Diese Information ist lediglich eine technische Beschreibung des Produktes. Sie stellt keine Beschaffenheits- oder Haltbarkeitsgarantie im Sinne des Paragraphen 443 BGB dar.
This information is exclusively a technical description of the products. It is not meant or intends to be a special guarantee for a particular quality or durability with regard to section 443 BGB (German Civil Code).

Document Name:	**Engine Management System**
Project:	**HMC Theta II ETC/BIN**
Engineering Change Number:	
Version:	20.0
Release Date:	2008-05-27

Version history

Document version	Previous version	Change description (including number)
20.0		ECM

Contents of All Chapters

0 Basic SW General Operation

Table of Contents

Index

0.1 Software version information for external equipment

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SW_NAME_0	V/O	0…FFh	0…255	1	-
SW version information part for 0 byte (Bit 0…15)					
SW_NAME_1	V/O	0…FFh	0…255	1	-
SW version information part for 1 byte (Bit 0…15)					
SW_NAME_2	V/O	0…FFh	0…255	1	-
SW version information part for 2 byte (Bit 0…15)					
SW_NAME_3	V/O	0…FFh	0…255	1	-
SW version information part for 3 byte (Bit 0…15)					
SW_NAME_4	V/O	0…FFh	0…255	1	-
SW version information part for 4 byte (Bit 0…15)					
SW_NAME_5	V/O	0…FFh	0…255	1	-
SW version information part for 5 byte (Bit 0…15)					

Input data:

FUNCTION DESCRIPTION:

General information:

In order to show software version information at external monitoring equipment (ex, VMU), version name values are defined. These values are prepared for A2L files of VMU. A2L file is modified as ASAP formation for SAM2000 values and 6 bytes can express the software information.

Software name can be composed of 6 byte and every one byte means one name of SW_NAME_x

Example) 670G30 = 36, 37, 30, 47, 33, 30 [hex] : SW_NAME_x

Formula section:

Activation: once at reset

0.2 End-Of-Line Trimming of analog MAF Signal Input

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MAF_TRIM	V/S	0…1H	0…1	1	-
flag to indicate a performed MAF Trimming					
STATE_MAF_TRIM	V/S	00H FFH	'Trim done' 'Trim not done'	1	-
Status Information for MAF Trimming from ECU Production Line (copied from address 6000h…6001h)					
V_MAF_TRIM	V/S	0…3FFH	0…4.995	0.00488	V
voltage of MAF signal input measured with C_V_MAF_TRIM_NOM input					
FAC_MAF_TRIM	V/O/S	0…FFH	0.9…1.1	0.00078	-
stored MAF Trimming Factor from ECU Production Line as ratio between measured and nominal voltage					

Input data:

FUNCTION DESCRIPTION:

General information:

In order to compensate the Voltage Tolerance of the ECU internal Reference Voltage Supply which is used to evaluate the MAF Sensor Signal Input, a Trimming Value is determined during the ECU Production Process at the Manufacturing End-of-Line Test.

A nominal Trimming Voltage C_V_MAF_TRIM_NOM (typically 4.000 V) is applied from an external Precision Device to the ECU MAF Signal Input and then measured inside the ECU.

The MAF-Signal Voltage inside the ECU is measured 10 times and then averaged. The averaged Value is stored in the non-volatile Memory and used to calculate the Trimming Value as Ratio between the measured Voltage and the nominal Voltage.

The Trimming Value is limited to +/- 10 % (FAC_MAF_TRIM = 0.9 … 1.1).

Measured Values which are outside the allowed tolerance window indicate a too high ECU Hardware Tolerance or a wrongly adjusted nominal Voltage and reject the ECU from the Production Line for Rework.

The Trimming Value is determined **only once in the ECU Lifetime** during the VXI Test. As the Trimming Value is unique to the ECU hardware, **it is never overwritten or erased**.

For Testing Purpose it is possible to switch from the trimmed value to the init value by a calibration constant. The trimmed value remains in the memory, but the read-out correction value is then changed to the initialization values.

Memory Address	**Description**
6000…6001 h	Trimming Status Information: 00FF h = Trimming Value available 0000 h = no Trimming Value available
6002…6003 h	V_MAF_TRIM Range: 0…3FF hex = 0…4.995 phys
6004…6005 h	Complement of V_MAF_TRIM

Application conditions:

Initialisation: the Initialization is performed only when no Trimming was done before or the Trimming has been reset by LC_RST_MAF_TRIM for testing.

If LV_MAF_TRIM = 0 from NVMY *(no Trimming Values available)*

then

if STATE_MAF_TRIM = 00 h *(read from Address 6000h…6001h)*
(to check if MAF Trimming was done at ECU Production Line)

and MAF Trimming Value from Address 6002h…6005h is valid

then V_MAF_TRIM = saved value from Address 6002h…6005h

FAC_MAF_TRIM = V_MAF_TRIM / C_V_MAF_TRIM_NOM
(calculation of Trimming Factor from saved Trimming Value)

LV_MAF_TRIM = 1 *(to indicate that Trimming is available)*

else V_MAF_TRIM = C_V_MAF_TRIM_NOM

FAC_MAF_TRIM = 1 *(to apply neutral init values)*

else *(Trimming Values are available)*

FAC_MAF_TRIM = FAC_MAF_TRIM from NVMY

V_MAF_TRIM = V_MAF_TRIM from NVMY

Recurrence: 1 sec

Activation: always active

Deactivation: no Deactivation

Formula section:

Reset of the Trimming Values for Testing Purpose:

If LC_RST_MAF_TRIM = 1 *(Reset of Trimming Value requested)*

then V_MAF_TRIM = C_V_MAF_TRIM_NOM *(to apply neutral init values)*

FAC_MAF_TRIM = 1 *(to apply neutral init values)*

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_MAF_TRIM_NOM	1	0…3FFH	0…4.995	0.00488	V
nominal Voltage from external device for MAF Signal Trimming					
LC_RST_MAF_TRIM	1	0…1H	0…1	1	-
for testing purpose the Trimming can be reset to init values					

0.3 Determination and Storage of ECU Reset Status

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ECU_RST_STATUS_SW_1	V/O/S	0…FFFFh	0…65535	1	-
ECU SW Reset Status (Bit 0…15)					
ECU_RST_STATUS_SW_1_HB	V/O/S	0…FFh	0…255	1	-
ECU SW Reset Status (Bit 8…15)					
ECU_RST_STATUS_SW_1_LB	V/O/S	0…FFh	0…255	1	-
ECU SW Reset Status (Bit 0…7)					
ECU_RST_STATUS_SW_2	V/O/S	0…FFFFh	0…65535	1	-
ECU SW Reset Status (Bit 16…31)					
ECU_RST_STATUS_SW_2_HB	V/O/S	0…FFh	0…255	1	-
ECU SW Reset Status (Bit 24…31)					
ECU_RST_STATUS_SW_2_LB	V/O/S	0…FFh	0…255	1	-
ECU SW Reset Status (Bit 16…23)					
ECU_RST_STATUS_HW	V/O/S	0…FFFFh	0…65535	1	-
ECU HW Reset Status (Bit 0…15)					
ECU_RST_STATUS_HW_HB	V/O/S	0…FFh	0…255	1	-
ECU HW Reset Status (Bit 8…15)					
ECU_RST_STATUS_HW_LB	V/O/S	0…FFh	0…255	1	-
ECU HW Reset Status (Bit 0…7)					

Input data:

BIOS_RST_GET_STATUS	BIOS_RST_IS_STATUS_VALID		

FUNCTION DESCRIPTION:

General information:

In order to support the root-cause identification after a non-power-on ECU Reset, the available background information from I/O Software is read out and stored to the non-volatile memory. The stored information is only written or overwritten by unexpected resets.

The BIOS driver for the Reset provides a basic interface BIOS_RST_GET_STATUS which describes the status of processor reset unit and cause of ECU reset, if one occurs.

There are two groups of possible reset causes:
- SW-caused resets (described by ECU_RST_STATUS_SW_1/2)
- HW-caused resets (described by ECU_RST_STATUS_HW)

Formula section:

Activation: once after any Reset

Deactivation: 1. when ECU is reset through KWP service request, ECU RESET

2. when Reprogramming Session gets started

If BIOS_RST_IS_STATUS_VALID *(to determine non-power-on Reset)*

then write data from BIOS_RST_GET_STATUS

into ECU_RST_STATUS_SW_1
and ECU_RST_STATUS_SW_2
and ECU_RST_STATUS_HW

SW Reset Status

Bit	Description	Value '1'	Value '0'
31	Machine check reset status	Reset has occurred	Reset has not occurred
30	Alignment reset status	Reset has occurred	Reset has not occurred
29	Program exception reset status	Reset has occurred	Reset has not occurred
28	Software emulation reset status	Reset has occurred	Reset has not occurred
27	Instruction protect reset status	Reset has occurred	Reset has not occurred
26	Data protect reset status	Reset has occurred	Reset has not occurred
25 24	Sys exception reserved	Reset has occurred	Reset has not occurred
23	Trace reset status	Reset has occurred	Reset has not occurred
22	Data break reset status	Reset has occurred	Reset has not occurred
21	Instruction break reset status	Reset has occurred	Reset has not occurred
20	Maskable break reset status	Reset has occurred	Reset has not occurred
19	NMI break reset status	Reset has occurred	Reset has not occurred
18 17 16	Debug exception reserved	Reset has occurred	Reset has not occurred
15	FP unavailable reset status	Reset has occurred	Reset has not occurred
14	FP assist reset status	Reset has occurred	Reset has not occurred
13 12 11 10 09 08	FP exception reserved	-	-
07	System call reset status	Reset has occurred	Reset has not occurred
06 05 04 03 02 01 00	System call exception reserved	-	-

HW Reset Status

Bit	Description	Value '1'	Value '0'
15	External hard reset status (EHRS)	Reset has occurred	Reset has not occurred
14	External soft reset status (ESRS)	Reset has occurred	Reset has not occurred
13	Loss of lock reset status (LLRS)	Reset has occurred	Reset has not occurred
12	Software watchdog reset status (SWRS)	Reset has occurred	Reset has not occurred
11	Checkstop reset status (CSRS)	Reset has occurred	Reset has not occurred
10	Debug port hard reset status (DBHRS)	Reset has occurred	Reset has not occurred
09	Debug port soft reset status (DBSRS)	Reset has occurred	Reset has not occurred
08	JTAG reset status (JTRS)	Reset has occurred	Reset has not occurred
07	On-chip clock switch (OCCS)	Reset has occurred	Reset has not occurred
06	Illegal bit change (ILBC)	Reset has occurred	Reset has not occurred
05	Glitch detected on PORESET pin (GPOR)	Reset has occurred	Reset has not occurred
04	Glitch detected on HRESET pin (GHRST)	Reset has occurred	Reset has not occurred
03	Glitch detected on SRESET pin (GSRST)	Reset has occurred	Reset has not occurred
02 01 00	Reserved	-	-

0.4 Siemens Logistic Information – Description

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
sam_name_ecu[8]	O	0...FFh	0...255	1	-
SAM name of ECU-SW					
sam_name[8]	O	0...FFh	0...255	1	-
SAM name of calibration					
NC_KWP_IO_RSCOTSN[10]	O	0...FFh	0...255	1	-
RepairShopCode or TesterSerialNumber					
NC_KWP_IO_PD[4]	O	0...FFh	0...255	1	-
Software ProgrammingDate					
NC_KWP_IO_CRSCOTSN[10]	O	0...FFh	0...255	1	-
Calibration RepairShopCode or TesterSerialNumber					
NC_KWP_IO_CPD[4]	O	0...FFh	0...255	1	-
Calibration ProgrammingDate					

0.5 Customer Logistic Information configuration

0.5.1 Customer logistic information fied

0.5.1.1 Calibration reprogramming history data

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAL_PROG_TESTER_NR_HIS_i[10]	O	0...FFH	0...255	1	-
Repair shop code or tester serial number used for calibration reprogramming					
CAL_PROG_DATE_HIS_i[4]	O	0...FFH	0...255	1	-
Calibration programming date (Format: YYYY/MM/DD)					
CAL_NAME_HIS_i[8]	O	0...FFH	0...255	1	-
ECU calibration identifier					

Input data:

NC_KWP_IO_CRSCOTSN[10]	NC_KWP_IO_CPD[4]	sam_name[8]	

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_ NR _PROG_HIS	-	0...255H	0...255	1	[-]
Size of reprogramming history memory					

0.5.2 Local configuration data:

Data	Value	
NC_NR_PROG_HIS	10	Size of reprogramming history memory

PHYSICAL IMPLEMENTATION

-

FUNCTION DESCRIPTION:

Any time the calibration data is programmed via KWP2000 service, the following data are stored in the logistic area of the Siemens Boot SW:

Description	Variables located in ASW logistic area (ALIF_SI and ALIF_CU)	Variables located in Siemens Boot Block (SBLIF_CU)
Tester serial Number or Repair shop code	NC_KWP_IO_CRSCOTSN	CAL_PROG_TESTER_NR_HIS_i
Calibration programming Date	NC_KWP_IO_CPD	CAL_PROG_DATE_HIS_i
ECU Calibration Identifier	sam_name	CAL_NAME_HIS_i

Application conditions:

Initialisation: at programming of SIEMENS Boot Block: all variables = 0xFFh

Recurrence: -

Activation: by KWP2000 service STRBLI with RELI 01h

Formula section:

The following action must be done **BEFORE** StartRoutineByLocalID service to erase data:

If (CAL_NAME_HIS_[NC_NR_PROG_HIS – 1][0..7] = 0xFF) /* Last entry for calibration reprogramming history not yet filled

and (SW_NAME_HIS_[NC_NR_PROG_HIS – 1][0..7] = 0xFF) /* Last entry for software reprogramming history not yet filled

Then Search next free entry (=x) in Calibration reprogramming history starting at 0

Copy actual data to position x in history:

CAL_PROG_TESTER_NR_HIS_x = NC_KWP_IO_CRSCOTSN

CAL_PROG_DATE_HIS_x = NC_KWP_IO_CPD

CAL_NAME_HIS_x = sam_name

Endif

0.5.2.1 Software reprogramming history data

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SW_PROG_TESTER_NR_HIS_i[10]	O	0...FFH	0...255	1	-
Repair shop code or tester serial number used for SW reprogramming					
SW_PROG_DATE_HIS_i[4]	O	0...FFH	0...255	1	-
Software programming date (Format: YYYY/MM/DD)					
SW_NAME_HIS_i[8]	O	0...FFH	0...255	1	-
ECU Software identifier					

Input data:

NC_KWP_IO_RSCOTSN	NC_KWP_IO_PD	sam_name_ecu	

PHYSICAL IMPLEMENTATION

-

FUNCTION DESCRIPTION:

Any time the software is programmed via KWP2000 service, the following data are stored in the logistic area of the Siemens Boot SW:

Description	Variables located in ASW logistic area (ALIF_SI and ALIF_CU)	Variables located in Siemens Boot Block (SBLIF_CU)
Tester serial Number or Repair shop code	NC_KWP_IO_RSCOTSN	SW_PROG_TESTER_NR_HIS_i
SW programming Date	NC_KWP_IO_PD	SW_PROG_DATE_HIS_i
ECU Software Identifier	sam_name_ecu	SW_NAME_HIS_i

Application conditions:

Initialisation: at programming of SIEMENS Boot Block: all variables = 0xFFh

Recurrence: -

Activation: by KWP2000 service STRBLI with RELI 00h

Formula section:

The following action must be done **BEFORE** StartRoutineByLocalID service to erase code:

If (CAL_NAME_HIS_[NC_NR_PROG_HIS – 1][0..7] = 0xFF) /* Last entry for calibration reprogramming history not yet filled

and (SW_NAME_HIS_[NC_NR_PROG_HIS – 1][0..7] = 0xFF) /* Last entry for software reprogramming history not yet filled

Then Search next free entry (=x) in Software reprogramming history starting at 0

Copy actual data to position x in history:

SW_PROG_TESTER_NR_HIS_x = NC_KWP_IO_RSCOTSN

SW_PROG_DATE_HIS_x = NC_KWP_IO_PD

SW_NAME_HIS_x = sam_name_ecu

Endif

0.5.2.2 Locked by timer information for immobilizer

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
BF_LOCKED_BY_TIMER[32]	O	0...FFh	0...255	1	-
Bitfield, that indicates number of times entered state LOCKED_BY_TIMER					

Application conditions:

Initialisation: at programming of SIEMENS Boot Block: all entries = 0xFFh

Formula section:

For more information see immobilizer specification.

0.6 Erasement of adaptation values

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_VB_OFF	V	0...1H	0...1	1	-
Battery disconnected.					
LV_VB_OFF_ACT	V/O	0...1H	0...1	1	-
Filtered battery disconnected information bit.					

Input data:

CONF_FMY_RST	CONF_AD_RST	ECU_RST_STATUS_SW_1	ECU_RST_STATUS_SW_1_HB
ECU_RST_STATUS_SW_1_LB	ECU_RST_STATUS_SW_2	ECU_RST_STATUS_SW_2_HB	ECU_RST_STATUS_SW_2_LB
ECU_RST_STATUS_HW	ECU_RST_STATUS_HW_LB	ECU_RST_STATUS_HW_HB	TPS_AD_MMV_IS
V_TPS_AD_LIH_1	C_V_TPS_SP_LIH_1	V_TPS_AD_LIH_2	C_V_TPS_SP_LIH_2
V_TPS_AD_BOL_1	C_V_TPS_AD_BOL_INI_1	V_TPS_AD_BOL_2	C_V_TPS_AD_BOL_INI_2
AR_RED_AD_ADD	AR_RED_AD_ADD_MMV	AR_RED_AD_FAC	AR_RED_AD_FAC_COR
CTR_AR_RED_AD_ADD_FAST	AMP_AD	CTR_AR_RED_AD_ADD_ENA	MAF_FAC_ALTI_MMV
PSN_EDGE_AD_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	PSN_EDGE_AD_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	TECU_MAX	DIST
TAM	TCO_STOP	T_AST_STOP	LV_STALL
IGA_AD1_KNK_x__N__MAF[8][6]	IGA_AD2_KNK	KNKS_CMD_GAIN_AD_x	MFF_ADD_LAM_AD[NC_CBK_EX_NR]
FAC_L_RNG_LAM_AD[NC_CBK_EX_NR]	FAC_H_RNG_LAM_AD[NC_CBK_EX_NR]	FAC_MFF_ADD_FAC_LAM_AD[NC_CBK_EX_NR]	CTR_COLD_ST_LS_UP
CTR_COLD_ST_LS_DOWN	EGY_DEW_INT_TUBE[NC_NR_TUBE_MDL]	EGY_DEW_INT_CAT_1[NC_NR_CAT_MDL]	EGY_DEW_INT_CAT_2[NC_NR_CAT_MDL]
EGY_DEW_INT_TUR[NC_NR_TUR_MDL]	TEMP_TUBE[NC_NR_TUBE_MDL]	TEMP_CAT	TEMP_TUR[NC_NR_TUR_MDL]
T_DLY_I_AD_LAM_ADJ[NC_CBK_EX_NR]	T_DLY_I_AD_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	TQ_DIF_IS_AD	TQ_DIF_IS_AD_INTER
TQ_DIF_IS_AD_CONV	TQ_DIF_IS_AD_ACC_1_INTER	TQ_DIF_IS_AD_CONV_INTER	TQ_DIF_IS_AD_ACC_CONV_INTER
TQ_DIF_IS_AD_ACC	CPPWM_AD_MEM	FLOW_COR_CPS_OLD	IVVTHPWM_AD_IN_i[12]
IVVTHPWM_AD_EX_i[12]	IVVTHPWM_AD_AS_IN_i[12]	IVVTHPWM_AD_AS_EX_i[12]	CAM_DIF_DIAG_STAT_SAE_IVVT_IN_i
CAM_DIF_DIAG_STAT_SAE_IVVT_EX_i	CAM_DIF_DIAG_DYN_SAE_IVVT_IN_i	CAM_DIF_DIAG_DYN_SAE_IVVT_EX_i	PUT_WG_OPEN_AD
PRS_DIF_WG_ACR_SP_AD	V_DIF_MAX_MWSS_DIAG	LV_ERR_TCO_PREL_DET	FTL_MMV_REF_INI
FTL_MMV_POS_DIAG	FTL_MMV_NEG_DIAG	CTR_FCO_FTL_STUCK	R_IT_THD_OBD_LSH_UP[NC_CBK_EX_NR]
R_IT_OBD_LSH_UP[NC_CBK_EX_NR]	LV_ERR_OBD_LSH_UP	FAC_LSL_GAIN_AD[NC_CBK_EX_NR]	IPLSL_MMV_VLD_FCUT[NC_CBK_EX_NR]
IPLSL_VARI_VLD_FCUT[NC_CBK_EX_NR]	FAC_DIAG_DYN_LSL_UP[NC_CBK_EX_NR]	FAC_MV_DIAG_DYN_LSL_UP[NC_CBK_EX_NR]	VLS_UP_DIAG_SAVE_PUC_LSL_UP[NC_CBK_EX_NR]
LAMB_THD_VPLSL_LIM[NC_CBK_EX_NR]	R_IT_THD_OBD_LSH_DOWN[NC_CBK_EX_NR]	R_IT_OBD_LSH_DOWN[NC_CBK_EX_NR]	DELTA_CRK_DIF_MAX_ER

CTR_SEG_AD_ER	LV_SEG_AD_AVL_ER	LV_SEG_AD_LIM_ER	SEG_AD_MMV_ER[NC_CYL_NR]
CTR_STOP_FSD	T_AST_ADD_OLD	TEMP_COR_DIAGCP_OLD	FAC_TI_ST_AD_DIAGCP_OLD
DTP_MOD_6	DTP_DIF_MOD_6	DTP_DIF_COR_MOD_6	DTP_DIF_ACT_MOD_6
DTP_DIF_FUC_MISS_DIAG_MOD_6	FAC_DIAM_DIAGCP_MOD_6	V_DTP_DIF_PLAUS_MOD_6	DTP_PLAUS_H_MOD_6
DTP_PLAUS_L_MOD_6	EFF_CAT_DIAG[NC_CBK_EX_NR]	CTR_CAT_DIAG_TOT[NC_CBK_EX_NR]	EFF_CAT_DIAG_OBD[NC_CBK_EX_NR]
EFF_CAT_MAX_DIAG_OBD[NC_CBK_EX_NR]	CTR_MIS _TOT_NVMY	CTR_MIS_TOT_DC	CTR_MIS_DC_CYL[NC_CYL_NR]
CTR_MIS_DC_MMV_CYL[NC_CYL_NR]	SUM_TQI_REQ_LIM	RATIO_TQI_BAS_MAX_STND_2	LV_PORT_AD_VLD
V_PORT_AD_TOL	C_V_PORT_AD_TOL	V_PORT_AD_BOL	C_V_PORT_AD_BOL
V_PORT_TOL	V_PORT_BOL	V_VIM_AD_LONG	V_VIM_AD_SHO
CPU_LOAD_MAX_TOT_DC	PV_AV_MAX_TOT_DC	PV_AV_GRD_MAX_TOT_DC	PUT_MES_MAX_TOT_DC
PUT_MES_MIN_TOT_DC	TCO_MAX_TOT_DC	TCO_MIN_TOT_DC	TOIL_MAX_TOT_DC
TOIL_MIN_TOT_DC	TIA_MAX_TOT_DC	TIA_MIN_TOT_DC	MAF_MAX_TOT_DC
MAF_MIN_TOT_DC	CTR_CST_LOW_ALTI	CTR_CST_HIGH_ALTI	CTR_WST
CTR_HST_LOW_ALTI	CTR_HST_HIGH_ALTI	T_ERU_IS	T_ERU_PL
T_ERU_FL	T_ALL_DIAG_READY	N_32_MAX_TOT_DC	TQ_AV_MAX_TOT_DC
V_FTL_SUB_REF_INI	V_FTL_SUB_POS_DIAG	CTR_FCO_FTL_SUB_STUCK	V_FTL_SUB_NEG_DIAG
CAM_INT_PAS_RTD_SAE_IVVT_IN_i	CAM_INT_PAS_ADC_SAE_IVVT_EX_i	LAMB_DELTA_AD_LAM_ADJ[NC_CBK_EX_NR]	TCO_MIN_LAM_AD_WUP[NC_CBK_EX_NR]
FAC_LAM_TCO_MIN[NC_CBK_EX_NR]	TCO_A_LAM_AD_WUP[NC_CBK_EX_NR]	FAC_LAM_TCO_A[NC_CBK_EX_NR]	TCO_B_LAM_AD_WUP[NC_CBK_EX_NR]
FAC_LAM_TCO_B[NC_CBK_EX_NR]	TCO_C_LAM_AD_WUP[NC_CBK_EX_NR]	FAC_LAM_TCO_C[NC_CBK_EX_NR]	TCO_D_LAM_AD_WUP[NC_CBK_EX_NR]
FAC_LAM_TCO_D[NC_CBK_EX_NR]	TCO_E_LAM_AD_WUP[NC_CBK_EX_NR]	FAC_LAM_TCO_E[NC_CBK_EX_NR]	SEG_AD_MMV_ER_L[NC_CYL_NR]
SEG_AD_MMV_ER_MID[NC_CYL_NR]	SEG_AD_MMV_ER_H[NC_CYL_NR]	FTL_MMV_OLD	T_AST_INIT_COR_OLD
T_INH_FTL_NEW_DIAGCP	LV_INH_T_ERU_DIAGCP_OLD	LV_INH_FTL_NEW_DIAGCP	TAM_FUEL_TEMP
FTL_MMV	LV_VIM_AD_VLD	C_V_VIM_SHO_INI	C_V_VIM_LONG_INI
V_VIM_LONG	V_VIM_SHO	STATE_VIM_AD	LV_VIM_AD_REQ
LV_VIM_SP_AD	T_MIN_PUC_VIM_AD	C_T_MIN_PUC_VIM_AD	MAF_DIF_SQ_TOT_CAM_OFS_AD_RES[NC_NR_OPP_CAM_OFS_AD]
LV_IM_VB_OFF	LV_PORT_AD_BOL_VLD	LV_PORT_AD_TOL_VLD	FTL_SAVE_FQ_ST_AD
FAC_FQ_ST_AD_SAVE_RNG_1	FAC_FQ_ST_AD_SAVE_RNG_2	FAC_FQ_ST_AD_SAVE_RNG_3	CAM_OFS_IVVT_IN[NC_NR_OPP_CAM_OFS_AD]
CAM_OFS_IVVT_EX[NC_NR_OPP_CAM_OFS_AD]	LV_DET_FIRST_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	CTR_CAM_OFS_AD_RES_NOT_PLAUS[NC_NR_OPP_CAM_OFS_AD]	LV_CAM_OFS_AD_SYM_POS_ERR[NC_NR_OPP_CAM_OFS_AD]
T_CTR_AMP_AD_CMPL	C_FAC_FQ_ST_AD_SAVE_RNG_1	C_FAC_FQ_ST_AD_SAVE_RNG_2	C_FAC_FQ_ST_AD_SAVE_RNG_3
C_FTL_SAVE_FQ_ST_AD	DATA_SAVE_CASE	SAVE_DIST	SAVE_N[10]
SAVE_TCO[10]	SAVE_TPS[10]	SAVE_VB[10]	SAVE_TIA[10]
SAVE_VLS_UP[10]	SAVE_VLS_DOWN[10]	SAVE_FAC_LAM_COR_CMN[10]	SAVE_FAC_LAM_AD[10]
SAVE_MFF_ADD_LAM_AD[10]	SAVE_MAF[10]	SAVE_MAF_MES[10]	SAVE_MAP_MES_BAS[10]
SAVE_LV_RLY_EFP[10]	SAVE_VS[10]	SAVE_AMP_AD[10]	SAVE_IGA_AV_MV[10]

SAVE_TI_1_0[10]	SAVE_AR_RED_DIF_REL[10]	SAVE_CPPWM[10]	SAVE_CL_MMV[10]
SAVE_LV_CS[10]	SAVE_CTR_DMF[10]	SAVE_LV_ENG_OFF_DMF_IGN[10]	SAVE_LV_ENG_OFF_DMF_INJ[10]
SAVE_LV_ENG_OFF_DMF_TPS_ISA[10]	SAVE_ECU_RST_STATUS_HW[10]	SAVE_ECU_RST_STATUS_SW_1[10]	SAVE_ECU_RST_STATUS_SW_2[10]
SAVE_LV_IGN_INJ_LOCK_REQ[10]	SAVE_LV_IMOB_INJ_OFF[10]	SAVE_LV_IMOB_IGC_OFF[10]	SAVE_TQI_ASR_REQ[10]
SAVE_TQI_MSR_REQ[10]	SAVE_TQI_ASR_SLW_REQ[10]	SAVE_TQI_GS_REQ[10]	SAVE_SPK_RTD_TCU[10]
SAVE_OBD_TPS_AV_1[10]	SAVE_OBD_TPS_AV_2[10]	SAVE_OBD_PV_1[10]	SAVE_OBD_PV_2[10]
SAVE_ISAPWM_ISA[10]	SAVE_TPS_SP[10]	SAVE_LV_OFF_IV_MON[10]	SAVE_LAMB_LS_UP[10]
SAVE_ERR_INTM_DIAG_INST_ACT[10]	CAM_OFS_IVVT_IN_MAX_TOT_DC	CAM_OFS_IVVT_EX_MAX_TOT_DC	CAM_OFS_IVVT_IN_MIN_TOT_DC
CAM_OFS_IVVT_EX_MIN_TOT_DC	CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC	AR_RED_AD_CAM_AD_MAX_TOT_DC	AR_RED_AD_CAM_AD_MIN_TOT_DC
AR_RED_AD_ADD_DIF_MAX_TOT_DC	AR_RED_AD_ADD_DIF_MIN_TOT_DC	AR_RED_SUM_REL_MAX_TOT_DC	AR_RED_SUM_REL_MIN_TOT_DC
PUT_MDL_DIF_MMV_REL_MAX_TOT_DC	PUT_MDL_DIF_MMV_REL_MIN_TOT_DC	T_VIM_SWI_MEC_LONG_MAX_TOT_DC	T_VIM_SWI_MEC_SHO_MAX_TOT_DC
FAC_L_RNG_LAM_AD_MAX_TOT_DC	FAC_L_RNG_LAM_AD_MIN_TOT_DC	FAC_LAM_OUT_MAX_TOT_DC	FAC_LAM_OUT_MIN_TOT_DC
MFF_ADD_LAM_AD_MAX_TOT_DC	MFF_ADD_LAM_AD_MIN_TOT_DC	T_MAX_FSD_LAM_LIM_MAX_TOT_DC	T_MIN_FSD_LAM_LIM_MAX_TOT_DC
MFF_ADD_LAM_CYL_SEL_ADJ_H_RES_x	LAM_CYL_SEL_ADJ_FAC_x	TTIP_UP_MAX_TOT_DC[NC_CBK_EX_NR]	TTIP_UP_MIN_TOT_DC [NC_CBK_EX_NR]
TCC_ERR_OBD_LSH_DOWN_MAX_TOT_DC [NC_CBK_EX_NR]	EFF_CAT_DIAG_MAX_TOT_DC[NC_CBK_EX_NR]	FAC_MV_DIAG_DYN_MAX_TOT_DC[NC_CBK_EX_NR]	V_PORT_OPEN_MAX_TOT_DC
V_PORT_OPEN_MIN_TOT_DC	V_PORT_CLOSE_MAX_TOT_DC	V_PORT_CLOSE_MIN_TOT_DC	PORT_AV_GRD_OPEN_MAX_TOT_DC
PORT_AV_GRD_OPEN_MIN_TOT_DC	PORT_AV_GRD_CLOSE_MAX_TOT_DC	PORT_AV_GRD_CLOSE_MIN_TOT_DC	CAM_OFS_IVVT_IN_1
CAM_OFS_IVVT_EX_1	LV_CAM_OFS_AD_END	CAM_OFS_IVVT_IN_1_PREV	CAM_OFS_IVVT_EX_1_PREV
MAP_MAX_TOT_DC	MAP_MIN_TOT_DC	VS_MAX_TOT_DC	TOIL_GB_MAX_TOT_DC
FAC_DIAM_DIAGCP_MAX_TOT_DC	VLS_UP_MAX_TOT_DC	VLS_UP_MIN_TOT_DC	VLS_DOWN_MAX_TOT_DC
VLS_DOWN_MIN_TOT_DC	RATIO_MV_CYC_AFL_MAX_TOT_DC	RATIO_MV_CYC_AFR_MAX_TOT_DC	RATIO_MV_CYCNR_AFL_MAX_TOT_DC
RATIO_MV_CYCNR_AFR_MAX_TOT_DC	N_TCHA_MAX_TOT_DC		

General information:

The adaptation values can be erased in several ways:

- via KW2000-protocol
- after battery disconnection detected
- with application system by calibration

The adaptation values can be erased in SAM2000 by the calibration constant LC_AD_CLR (all adaptation values) or by the calibrations LC_AD_CLR_xxx (individual adaptation values).

The Reset is effective immediately and only at the transition from LC_AD_CLR(_xxx)= 0 to 1.

Note : additionally to the erasement of adpatation values, an erasement of the failure memory can be performed. More informatino available in chapter A.

Application recurrence: once after system reset.

0.6.1 Erasement after battery disconnection

A hardware with one bit-memory characteristic is provided which detects at system start up if the ECU or the battery was disconnected since the last housekeeping phase.

If a disconnection is configured and detected, and erasement of non volatile memory is configured the adaptation values will be reset like described below.

Formula section:

Detection of battery-off :

If CONF_FMY_RST = 1
or CONF_AD_RST > 0
then LV_VB_OFF_ACT = LV_VB_OFF *(One-Bit Memory is set)*
else LV_VB_OFF_ACT = 0 *(reset One-Bit Memory)*

Remark : The flag LV_VB_OFF is set to 1 if the battery has been disconnected since the last driving cycle : LV_VB_OFF = LV_IM_VB_OFF

Erasement of failure memory:

If LV_VB_OFF = 1
then **If** CONF_FMY_RST = 0 **then** the failure memory is **not** reset
If CONF_FMY_RST = 1 **then** the failure memory is reset

Erasement of adaptative values from the memory :

If LV_VB_OFF = 1
then **If** CONF_AD_RST = 0 **then** the adaptation values are **not** reset
If CONF_AD_RST = 1 **then all** the adaptation values are reset
If CONF_AD_RST = 2 **then** only **some** adaptation values are reset. See details in the table below.

Remark : After the first power latch phase after battery disconnection, it takes at least 50 ms until the One-Bit Memory is reset.

0.6.2 Adaptation values to be reset

	Category	Variable	reset value	LC_AD_CLR_xx	VB_OFF*)
0	ECU reset status	ECU_RST_STATUS_SW_1 ECU_RST_STATUS_SW_1_HB ECU_RST_STATUS_SW_1_LB ECU_RST_STATUS_SW_2 ECU_RST_STATUS_SW_2_HB ECU_RST_STATUS_SW_2_LB ECU_RST_STATUS_HW ECU_RST_STATUS_HW_HB ECU_RST_STATUS_HW_LB	0h 0h 0h 0h 0h 0h 0h 0h 0h	lc_clr_ecu_rst_status	no
0	Max CPU_LOAD	CPU_LOAD_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
4	TPS Adaptation	- For non-ETC system TPS_AD_MMV_IS - For ETC system **) V_TPS_AD_LIH_1 V_TPS_AD_LIH_2 V_TPS_AD_BOL_1 V_TPS_AD_BOL_2	see chap. 4 C_V_TPS_SP_LIH_1 C_V_TPS_SP_LIH_2 C_V_TPS_AD_BOL_INI_1 C_V_TPS_AD_BOL_INI_2	lc_ad_clr_tps	yes no
4	Reduced area controller	AR_RED_AD_ADD AR_RED_AD_ADD_MMV AR_RED_AD_FAC AR_RED_AD_FAC_COR CTR_AR_RED_AD_ADD_FAST	0h 0h 0h 0h 0h	lc_ad_clr_imm	yes
4	AMP Controller	AMP_AD CTR_AR_RED_AD_ADD_ENA T_CTR_AMP_AD_CMPL	c_amp_ini 0h 0h	lc_ad_clr_imm	yes
4	Altitude Correction	MAF_FAC_ALTI_MMV	80h	lc_ad_clr_maf_alti	yes
4	Camshaft Adapt.	PSN_EDGE_AD_CAM_IN_i PSN_EDGE_AD_CAM_EX_i	200h (0 phys.) 200h (0 phys.)	lc_ad_clr_ensd	yes
4	Camshaft Offset Adapt.	CAM_OFS_IVVT_IN[NC_NR_OPP_CAM_OFS_AD] CAM_OFS_IVVT_EX[NC_NR_OPP_CAM_OFS_AD] MAF_DIF_SQ_TOT_CAM_OFS_AD_RES[NC_NR_OPP_CAM_OFS_AD] LV_DET_FIRST_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD CTR_CAM_OFS_AD_RES_NOT_PLAUS[NC_NR_OPP_CAM_OFS_AD] LV_CAM_OFS_AD_SYM_POS_ERR[NC_NR_OPP_CAM_OFS_AD] CAM_OFS_IVVT_IN(EX)_1 LV_CAM_OFS_AD_END CAM_OFS_IVVT_IN(EX)_1_PREV	0h 0h 0h 0h 0h 0h 0h 0h 0h	lc_ad_clr_imm	yes
4	Max. ECU Temp.	TECU_MAX	0h	lc_ad_clr_tecu_max	no
4	Mileage Counter	DIST	0h	lc_ad_clr_sys	no
4	Ambient Air Temp.	TAM	see spec 30402402	lc_ad_clr_airt	no
4	Coolant Temp.	TCO_STOP	0h	lc_ad_clr_ente	no
4	Min/Max Coolant Temp	TCO_MAX_TOT_DC TCO_MIN_TOT_DC	0h FEh	lc_rst_fleet_stc	no
4	Min/Max Oil Temp	TOIL_MAX_TOT_DC TOIL_MIN_TOT_DC	0h C8h	lc_rst_fleet_stc	no
4	Min/Max Intake Air Temp	TIA_MAX_TOT_DC TIA_MIN_TOT_DC	0h FEh	lc_rst_fleet_stc	no

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
4	Min/Max Pedal value	PV_AV_MAX_TOT_DC PV_AV_GRD_MAX_TOT_DC	0h 0h	lc_rst_fleet_stc	no
4	Min/Max Boost pressure	PUT_MES_MAX_TOT_DC PUT_MES_MIN_TOT_DC	0h FFFFh	lc_rst_fleet_stc	no
4	Min/Max Air mass	MAF_MAX_TOT_DC MAF_MIN_TOT_DC	0h FFFFh	lc_rst_fleet_stc	no
4	Min/Max manifold Air pressure	MAP_MAX_TOT_DC MAP_MIN_TOT_DC	0h FFFFh	lc_rst_fleet_stc	no
4	Max Eng. Speed	N_32_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
4	Max Vehicle Speed	VS_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
4	IMM Fleet variables	CAM_OFS_IVVT_IN_MAX_TOT_DC CAM_OFS_IVVT_EX_MAX_TOT_DC CAM_OFS_IVVT_IN_MIN_TOT_DC CAM_OFS_IVVT_EX_MIN_TOT_DC CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC AR_RED_AD_CAM_AD_MAX_TOT_DC AR_RED_AD_CAM_AD_MIN_TOT_DC AR_RED_AD_ADD_DIF_MAX_TOT_DC AR_RED_AD_ADD_DIF_MIN_TOT_DC AR_RED_SUM_REL_MAX_TOT_DC AR_RED_SUM_REL_MIN_TOT_DC PUT_MDL_DIF_MMV_REL_MAX_TOT_DC PUT_MDL_DIF_MMV_REL_MIN_TOT_DC	8000H 8000H 7FFFH 7FFFH 0H 8000H 7FFFH 8000H 7FFFH 8000H 7FFFH 8000H 7FFFH	lc_rst_fleet_stc	no
4	CAN Messages	TOIL_GB_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
5	Auxilary Start Function	T_AST_STOP LV_STALL	0h 0h	lc_ad_clr_eng_au x	yes
5	Engine operating states	CTR_CST_LOW_ALTI CTR_CST_HIGH_ALTI CTR_WST CTR_HST_LOW_ALTI CTR_HST_HIGH_ALTI T_ERU_IS T_ERU_PL T_ERU_FL T_ALL_DIAG_READY	0h 0h 0h 0h 0h 0h 0h 0h 0h	lc_rst_fleet_stc	no
5	History data at engine stall or unexpected acceleration	DATA_SAVE_CASE SAVE_DIST SAVE_N SAVE_TCO SAVE_TPS SAVE_VB SAVE_TIA SAVE_VLS_UP SAVE_VLS_DOWN SAVE_FAC_LAM_COR_CMN SAVE_FAC_LAM_AD SAVE_MFF_ADD_LAM_AD SAVE_MAF SAVE_MAF_MES SAVE_MAP_MES_BAS SAVE_LV_RLY_EFP SAVE_VS SAVE_AMP_AD SAVE_IGA_AV_MV	0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h	lc_rst_his_spo	no

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
		SAVE_TI_1_0 SAVE_AR_RED_DIF_REL SAVE_CPPWM SAVE_CL_MMV SAVE_LV_CS SAVE_CTR_DMF SAVE_LV_ENG_OFF_DMF_IGN SAVE_LV_ENG_OFF_DMF_INJ SAVE_LV_ENG_OFF_DMF_TPS_ISA SAVE_ECU_RST_STATUS_HW SAVE_ECU_RST_STATUS_SW_1 SAVE_ECU_RST_STATUS_SW_2 SAVE_LV_IGN_INJ_LOCK_REQ SAVE_LV_IMOB_INJ_OFF SAVE_LV_IMOB_IGC_OFF SAVE_TQI_ASR_REQ SAVE_TQI_MSR_REQ SAVE_TQI_ASR_SLW_REQ SAVE_TQI_GS_REQ SAVE_SPK_RTD_TCU SAVE_OBD_TPS_AV_1 SAVE_OBD_TPS_AV_2 SAVE_OBD_PV_1 SAVE_OBD_PV_2 SAVE_ERR_INTM_DIAG_INST_ACT - For non-ETC system SAVE_ISAPWM_ISA - For ETC system SAVE_TPS_SP SAVE_LV_OFF_IV_MON - For lin. upstream O2-Sensor config.: SAVE_LAMB_LS_UP	0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h 0h		
6	Knock Adaptation	IGA_AD1_KNK_x__N__MAF IGA_AD2_KNK KNKS_CMD_GAIN_AD_x	80h 80h 00h	lc_ad_clr_knk	yes
7	Lambda Adaptation	MFF_ADD_LAM_AD[i] FAC_L_RNG_LAM_AD[i] FAC_H_RNG_LAM_AD[i] FAC_MFF_ADD_FAC_LAM_AD - For lin. upstream O2-Sensor config.: TCO_MIN_LAM_AD_WUP TCO_A_LAM_AD_WUP TCO_B_LAM_AD_WUP TCO_C_LAM_AD_WUP TCO_D_LAM_AD_WUP TCO_E_LAM_AD_WUP FAC_LAM_TCO_MIN FAC_LAM_TCO_A FAC_LAM_TCO_B FAC_LAM_TCO_C FAC_LAM_TCO_D FAC_LAM_TCO_E CTR_STOP_FSD	0h 0h 0h 0h 83h (50°C) C_TCO_A_LAM_AD_WUP C_TCO_B_LAM_AD_WUP C_TCO_C_LAM_AD_WUP C_TCO_D_LAM_AD_WUP C_TCO_E_LAM_AD_WUP 0h (0 %) 0h (0 %) 0h (0 %) 0h (0 %) 0h (0 %) 0h (0 %) 0h	lc_ad_clr_lam or LV_ERR_LOAD_TPS_PLAUS = 1	yes
7	EXTD	CTR_COLD_ST_LS_UP CTR_COLD_ST_LS_DOWN EGY_DEW_INT_TUBE[i]	0h 0h 0h	lc_ad_clr_extd	yes

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
		EGY_DEW_INT_CAT_1[i] EGY_DEW_INT_CAT_2[i] EGY_DEW_INT_TUR[i] TEMP_TUBE[i] TEMP_CAT[i] TEMP_TUR[i]	0h 0h 0h 1112h 1112h 1112h		
7	Fuel Trim Adapt.	- For bin upstream O2-Sensor config.: T_DLY_I_AD_LAM_ADJ T_DLY_I_AD_LAM_ADJ_CAT_DIAG - For lin. upstream O2-Sensor config.: LAMB_DELTA_AD_LAM_ADJ	 0h 0h 0h	lc_ad_clr_lam	no
7	Fuel Quality Adapt.	FAC_FQ_ST_AD_SAVE_RNG_1 FAC_FQ_ST_AD_SAVE_RNG_2 FAC_FQ_ST_AD_SAVE_RNG_3 FTL_SAVE_FQ_ST_AD	C_FAC_FQ_ST_AD_ SAVE_RNG_1 C_FAC_FQ_ST_AD_ SAVE_RNG_2 C_FAC_FQ_ST_AD_ SAVE_RNG_3 C_FTL_SAVE_FQ_ST_AD	lc_ad_clr_fq	yes
7	CILC	- For lin. upstream O2-Sensor config.: MFF_ADD_LAM_CYL_SEL_ADJ_H_RES_x LAM_CYL_SEL_ADJ_FAC_x	 0h 0h	lc_ad_clr_cilc	yes
8	Idle Speed Controller	TQ_DIF_IS_AD TQ_DIF_IS_AD_INTER TQ_DIF_IS_AD_CONV TQ_DIF_IS_AD_ACC_1_INTER TQ_DIF_IS_AD_CONV_INTER TQ_DIF_IS_AD_ACC_CONV_INTER TQ_DIF_IS_AD_ACC	0h 0h 0h 0h 0h 0h 0h	lc_ad_clr_tqlo	yes
9	CPS Adaptation	CPPWM_AD_MEM FLOW_COR_CPS_OLD	0h 0h	lc_ad_clr_cps	yes
9	IVVT	IVVTHPWM_AD_IN(_EX)_i[12] IVVTHPWM_AD_AS_IN(_EX)_i[12] CAM_DIF_DIAG_STAT_SAE_IVVT_IN_i CAM_DIF_DIAG_STAT_SAE_IVVT_EX_i CAM_DIF_DIAG_DYN_SAE_IVVT_IN_i CAM_DIF_DIAG_DYN_SAE_IVVT_EX_i CAM_INT_PAS_RTD_SAE_IVVT_IN_i CAM_INT_PAS_ADC_SAE_IVVT_EX_i	0 % 0 % Attention 0 % <> 0h 0h 0h 0h 0h 0h 0h	lc_ad_clr_ivvt	yes
9	CHRG	- For TCI-System (691SW): PUT_WG_OPEN_AD PRS_DIF_WG_ACR_SP_AD	 7FFFh (phys.: 0 hPa) 7FFFh (phys.: 0 hPa)	see spec 30408C01 see spec 30908D01	yes
9	PORT	LV_PORT_AD_VLD LV_PORT_AD_BOL_VLD LV_PORT_AD_TOL_VLD V_PORT_AD_TOL V_PORT_AD_BOL V_PORT_TOL V_PORT_BOL	0h 0h 0h C_V_PORT_AD_TOL C_V_PORT_AD_BOL C_V_PORT_AD_TOL C_V_PORT_AD_BOL	lc_ad_clr_port	no

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
9	VIM	V_VIM_AD_LONG V_VIM_AD_SHO V_VIM_LONG V_VIM_SHO STATE_VIM_AD T_MIN_PUC_VIM_AD LV_VIM_AD_REQ LV_VIM_SP_AD LV_VIM_AD_VLD	C_V_VIM_LONG_INI C_V_VIM_SHO_INI C_V_VIM_LONG_INI C_V_VIM_SHO_INI 0h (VIM_AD_PUC_TIMER) C_T_MIN_PUC_VIM_AD 1h 0h 0h	lc_ad_clr_vim	no
9	PORT Fleet variables	V_PORT_OPEN_MAX_TOT_DC V_PORT_OPEN_MIN_TOT_DC V_PORT_CLOSE_MAX_TOT_DC V_PORT_CLOSE_MIN_TOT_DC PORT_AV_GRD_OPEN_MAX_TOT_DC PORT_AV_GRD_OPEN_MIN_TOT_DC PORT_AV_GRD_CLOSE_MAX_TOT_DC PORT_AV_GRD_CLOSE_MIN_TOT_DC	0h 3FFh 0h 3FFh F000h 0h 0h 1000h	lc_rst_fleet_stc	no
A	MWSS Adaptation	V_DIF_MAX_MWSS_DIAG	0h	lc_ad_clr_mwss	yes
A	TCO Sensor Error	LV_ERR_TCO_PREL_DET	0h	lc_ad_clr_ente	no
A	Pedal Value Sensor Error	- For ETC-System: LV_PVS_H_R_1_SAVE LV_PVS_H_R_2_SAVE	 0h 0h	lc_ad_clr_pvs	no
A	Fuel level sensor	FTL_MMV_REF_INI FTL_MMV_POS_DIAG FTL_MMV_NEG_DIAG CTR_FCO_FTL_STUCK V_FTL_SUB_REF_INI V_FTL_SUB_POS_DIAG V_FTL_SUB_NEG_DIAG CTR_FCO_FTL_SUB_STUCK	0h 0h 0h 0h 0h 0h 0h 0h	lc_ad_clr_ftl	yes
A	VIM diagnosis	T_VIM_SWI_MEC_LONG_MAX_TOT_DC T_VIM_SWI_MEC_SHO_MAX_TOT_DC	0h 0h	lc_rst_fleet_stc	no
B	Upstream O2 sensor	- For bin. upstream O2-Sensor config: R_IT_THD_OBD_LSH_UP R_IT_OBD_LSH_UP LV_ERR_OBD_LSH_UP - For lin. upstream O2-Sensor config.: FAC_LSL_GAIN_AD IPLSL_MMV_VLD_FCUT IPLSL_VARI_VLD_FCUT FAC_DIAG_DYN_LSL_UP FAC_MV_DIAG_DYN_LSL_UP VLS_UP_DIAG_SAVE_PUC_LSL_UP LAMB_THD_VPLSL_LIM	 FFFFh FFFFh 0 0h c_iplsl_nom_purge 0h 0h 0h 0h 0h	lc_ad_clr_lam	no
		VLS_UP_MAX_TOT_DC VLS_UP_MIN_TOT_DC	0h 3FFh	lc_rst_fleet_stc	no
B	Downstream O2 sensor	R_IT_THD_OBD_LSH_DOWN R_IT_OBD_LSH_DOWN	FFFFh FFFFh	lc_ad_clr_lam	no
		VLS_DOWN_MAX_TOT_DC VLS_DOWN_MIN_TOT_DC	0h 3FFh	lc_rst_fleet_stc	no

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
B	Segment Time Adaptation	DELTA_CRK_DIF_MAX_ER	(c_crk_dif_max_er * NC_CYL_NR) << 2	lc_ad_clr_enrd	yes
		CTR_SEG_AD_ER	0h		
		LV_SEG_AD_AVL_ER	0h		
		LV_SEG_AD_LIM_ER	0h		
		SEG_AD_MMV_ER[NC_CYL_NR]	0h		
		SEG_AD_MMV_ER_L[NC_CYL_NR]	0h		
		SEG_AD_MMV_ER_MID[NC_CYL_NR]	0h		
		SEG_AD_MMV_ER_H[NC_CYL_NR]	0h		
B	FSD	T_AST_ADD_OLD	0h	lc_ad_clr_fsd	yes
		TEMP_COR_DIAGCP_OLD	40h (phys:0°C)		
		FAC_TI_ST_AD_DIAGCP_OLD	80h (phys: 1)		
		- For bin. upstream O2-Sensor config:			
		CTR_STOP_FSD	0h		
B	EVAP	DTP_MOD_6	0h	lc_ad_clr_evap	no
		DTP_DIF_MOD_6	0h		
		DTP_DIF_COR_MOD_6	0h		
		DTP_DIF_ACT_MOD_6	0h		
		DTP_DIF_FUC_MISS_DIAG_MOD_6	0h		
		FAC_DIAM_DIAGCP_MOD_6	0h		
		V_DTP_DIF_PLAUS_MOD_6	0h		
		DTP_PLAUS_H_MOD_6	0h		
		DTP_PLAUS_L_MOD_6	0h		
		FTL_MMV_OLD	FTL_MMV		
		TAM_FUEL_TEMP	0h		
		T_AST_INIT_COR_OLD	0h		
		T_INH_FTL_NEW_DIAGCP	0h		
		LV_INH_T_ERU_DIAGCP_OLD	0h		
		LV_INH_FTL_NEW_DIAGCP	0h		
		FAC_DIAM_DIAGCP_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
B	Catalyst Diag.	EFF_CAT_DIAG	0h	lc_ad_clr_cat_diag	no
		CTR_CAT_DIAG_TOT	0h		
		EFF_CAT_DIAG_OBD[i]	0h		
		EFF_CAT_MAX_DIAG_OBD[i]	0h		
B	Misfire Detection	CTR_MIS TOT_NVMY	0h	lc_ad_clr_misf	no
		CTR_MIS_TOT_DC	0h		
		CTR_MIS_DC_CYL	0h		
		CTR_MIS_DC_MMV_CYL	0h		
B	LACO Fleet variables	- For lin. upstream O2-Sensor config.:	0h	lc_rst_fleet_stc	no
		FAC_L_RNG_LAM_AD_MAX_TOT_DC	0h		
		FAC_L_RNG_LAM_AD_MIN_TOT_DC	0h		
		FAC_LAM_OUT_MAX_TOT_DC	0h		
		FAC_LAM_OUT_MIN_TOT_DC	0h		
		MFF_ADD_LAM_AD_MAX_TOT_DC	0h		
		MFF_ADD_LAM_AD_MIN_TOT_DC	0h		
		T_ MAX_FSD_LAM_LIM_MAX_TOT_DC	0h		
		T_ MIN_FSD_LAM_LIM_MAX_TOT_DC			
B	EGCP Fleet variables	- For lin. upstream O2-Sensor config.:		lc_rst_fleet_stc	no
		TTIP_UP_MAX_TOT_DC	8000h		
		TTIP_UP_MIN_TOT_DC	7FFFh		
		TCC_ERR_OBD_LSH_DOWN_MAX_TOT_DC	0h		
		FAC_MV_DIAG_DYN_MAX_TOT_DC	0h		
		- For bin. upstream O2-Sensor config.:			
		RATIO_MV_CYC_AFL_MAX_TOT_DC	0h		
		RATIO_MV_CYC_AFR_MAX_TOT_DC	0h		
		RATIO_MV_CYCNR_AFL_MAX_TOT_DC	0h		
		RATIO_MV_CYCNR_AFR_MAX_TOT_DC	0h		

	Category	Variable	reset value	LC_AD_CLR_xx	VB_ OFF*)
B	EGTR Fleet variables	- For lin. upstream O2-Sensor config.: EFF_CAT_DIAG_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
B	Turbo charger rotational speed	- For TCI config.(NC_CHRG_CONF = 1): N_TCHA_MAX_TOT_DC	0h	lc_rst_fleet_stc	no
D	Torque Limitation	SUM_TQI_REQ_LIM	0h	lc_ad_clr_tqsp	yes
4	Max Eng. Torque	TQ_AV_MAX_TOT_DC	8000h	lc_rst_fleet_stc	no
D	Torque reduction factor	RATIO_TQI_BAS_MAX_STND_2	8000h	lc_ad_clr_ratio_tqi	yes

*) VB_OFF : this column inidicates which adaptative should be reset in case of LV_VB_OFF = 1 and CONF_AD_RST = 2.

**) if lc_ad_clr_tps is set to 1 only the request for new adaptation is set (LV_TPS_AD_REQ); the values are initialised to its reset values at the beginning of the adaptation

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_AD_CLR	1	0...1H	0...1	1	-
Logical constant for initializing all adaption values					
LC_AD_CLR_SYS	1	0...1H	0...1	1	-
Logical constant for initializing DIST					
LC_AD_CLR_KNK	1	0...1H	0...1	1	-
Logical constant for initializing KNK adaption values					
LC_AD_CLR_LAM	1	0...1H	0...1	1	-
Logical constant for initializing LAMBDA adaption values					
LC_AD_CLR_MAF_ALTI	1	0...1H	0...1	1	-
Logical constant for initializing MAF_FAC_ALTI adaption value					
LC_AD_CLR_TPS	1	0...1H	0...1	1	-
Logical constant for initializing TPS adaption values					
LC_AD_CLR_IVVT	1	0...1H	0...1	1	-
Logical constant for initializing IVVT adaption values					
LC_AD_CLR_ENSD	1	0...1H	0...1	1	-
Logical constant for initializing ENSD adaption values					
LC_AD_CLR_IMM	1	0...1H	0...1	1	-
Logical constant for initializing IMM adaption values					
LC_AD_CLR_MWSS	1	0...1H	0...1	1	-
Logical constant for initializing MWSS Diagnosis					
LC_AD_CLR_EXTD	1	0…1H	0…1	1	-
Logical constant for initializing EXTD adaption values					
LC_AD_CLR_ENRD	1	0…1H	0…1	1	-
Logical constant for initializing ENRD adaption values					
LC_AD_CLR_AIRT	1	0…1H	0…1	1	-
Logical constant for initializing AIRT adaption values					
LC_AD_CLR_TQLO	1	0…1H	0…1	1	-
Logical constant for initializing TQLO adaption values					
LC_AD_CLR_TQSP	1	0…1H	0…1	1	-
Logical constant for initializing TQSP adaption values					
LC_AD_CLR_MISF	1	0…1H	0…1	1	-
Logical constant for initializing MISF adaption values					
LC_AD_CLR_ENG_AUX	1	0…1H	0…1	1	-
Logical constant for initializing ENG_AUX adaption values					
LC_AD_CLR_RATIO_TQI	1	0…1H	0…1	1	-
Logical constant for initializing RATIO_TQI_BAS_MAX_STND_2 adaption values					
LC_AD_CLR_CAT_DIAG	1	0…1H	0…1	1	-
Logical constant for initializing catalyst diagnosis					
LC_AD_CLR_ENTE	1	0...1H	0...1	1	-
Logical constant for initializing ENTE adaptation values					
LC_AD_CLR_FSD	1	0...1H	0...1	1	-
Logical constant for initializing fuel system diagnosis					
LC_AD_CLR_CPS	1	0...1H	0...1	1	-
Logical constant for initializing canister purge solenoid adaptaion values					
LC_AD_CLR_TECU_MAX	1	0...1H	0...1	1	-
Logical constant for initializing max. ECU temperature					
LC_AD_CLR_PVS	1	0...1H	0...1	1	-
Logical constant for initializing pedal value sensor diagnosis value					
LC_AD_CLR_FTL	1	0…1H	0…1	1	-
Logical constant for initializing fuel level sensor diagnosis value					
LC_CLR_ECU_RST_STATUS	1	0…1H	0…1	1	-
Logical constant for initializing ECU reset status					
LC_AD_CLR_EVAP	1	0...1H	0...1	1	-
Logical constant for initializing EVAP monitoring values					

LC_AD_CLR_PORT	1	0...1H	0...1	1	-
Logical constant for Initializing PORT Adaptation values					
LC_RST_FLEET_STC	1	0...1H	0...1	1	-
Logical constant to reset fleet statistic values					
LC_AD_CLR_VIM	1	0...1H	0...1	1	-
Logical constant for Initializing VIM Adaptation values					
LC_AD_CLR_FQ	1	0...1H	0...1	1	-
Logical constant for Initializing Fuel Quality adaptation values					
LC_AD_CLR_CILC	1	0...1H	0...1	1	-
Logical constant for Initializing CILC adaptation values					
LC_RST_HIS_SPO	1	0...1H	0...1	1	-
Logical constant for initializing saved history data at spontaneous engine/vehicle behaviour					

0.7 Control of data saving in Non Volatile Memory

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_RD_AR_NVMY	O	0...1H	0...1	1	-
Flag indicating Block restored has already been read NC_MAX_RD_AR_NVMY times					

General information:

The saving of datas to the Non Volatile Memory is only allowed at the end of Power latch before ECU switch off. The area for saved data is made of several blocks. At the end of the Power latch data are stored from RAM to the next free block of NVMY. A reset before end of Powerlatch prevents this saving and the last valid block is read out again at next initalisation. If same block of Reset values is already read out for NC_MAX_RD_NVMY times, the flag LV_ERR_RD_AR_NVMY is set.

Application conditions:

Initialisation: *(no Initialisation, because Activation during reset Phase)*

Recurrence: once per Reset

Activation: at Reset

Formula section:

If block n is valid, but this block has been read already NC_MAX_RD_AR_NVMY times
then LV_ERR_RD_AR_NVMY = 1
else LV_ERR_RD_AR_NVMY = 0
endif

NC_MAX_RD_AR_NVMY = 1

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_MAX_RD_AR_NVMY	-	0..10H	0...16	1	-
number of read outs of NVMY datas already read by previous store					

0.8 CPU Load Measurement

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CPU_LOAD	O/V	0...400H	0..100	0.0977	%
CPU load above measurement level					
CPU_LOAD_MAX	O/V	0...400H	0..100	0.0977	%
Maximum detected CPU load above measurement level					
CPU_LOAD_MAX_TOT_DC	O/V/S	0...400H	0..100	0.0977	%
Former / current driving cycle maximum CPU_LOAD					
CPU_LOAD_TSK_PER_MIN_SAVE	O/V/S	0...FFFFFFFFH	0...4294967295	1	Ticks
Saved Minimum Time between 2 Instances of TASK_BG_MES					
CPU_LOAD_TSK_PER_MIN	O/V	0...FFFFFFFFH	0... 4294967295	1	Ticks
Minimum Time between 2 Instances of TASK_BG_MES					
CPU_LOAD_TSK_PER	O/V	0...FFFFFFFFH	0... 4294967295	1	Ticks
Time between 2 Instances of TASK_BG_MES					
CPU_LOAD_TSK_PER_CTR	O/V	0...FFFFH	0..65535	1	-
Counter of TASK_BG_MES Instances					

FUNCTION DESCRIPTION:

General information:

The CPU load is defined as the processor activity above the background level.

The variable CPU_LOAD_MAX represents the maximum detected CPU load value. The calibration flag LC_CPU_LOAD_MAX_RST can be used in order to reset this variable.

The variable CPU_LOAD_MAX_TOT_DC represents the maximum CPU load value over several driving cycles since ECU programming.

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

CPU_LOAD_MAX_TOT_DC = 0

- otherwise: restored from non-volantile memory

Formula section:

IF CPU_LOAD_MAX > CPU_LOAD_MAX_TOT_DC

THEN CPU_LOAD_MAX_TOT_DC = CPU_LOAD_MAX

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CPU_LOAD_SES_T_BEGIN		0..FFFFH	[0…65535]	1	-
Beginning of measurement session					
C_CPU_LOAD_SES_T_END		0..FFFFH	[0…65535]	1	-
End of measurement session					
C_CPU_LOAD_PER		0..FFFFH	[0…65535	1	-
Time between 2 measurement sessions					
C_CPU_LOAD_TSK_MES_DLY		0...FFFFH	0...65535	1	-
Calibration for acting on measurement task period					
LC_CPU_LOAD_TSK_PER_MIN_UPD		0..1	0..1	-	-
Flag for authorizing update of variable CPU_LOAD_TSK_PER_MIN_SAVE with CPU_LOAD_TSK_PER_MIN					
LC_CPU_LOAD_TSK_PER_MIN_RST		0..1	0..1	-	-
Flag for reset of variable CPU_LOAD_TSK_PER_MIN					
LC_CPU_LOAD_MAX_RST		0..1	0..1	-	-
Flag for reset of variable CPU_LOAD_MAX					

0.9 Stack size monitoring of Ercosek 4.1

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
E_OS_STATE_STACK	V	0000H 0001H 0002H 0100H 0200H 0404H	STACK_OK USTACK_OVFL USTACK_NFULL ESTACK_OVFL ESTACK_NFULL OSTACK_OVFL	1	[-]
Stack status					
E_OS_USED_USER_STACK	V	0...FFFFFFFFH	0...17179869180	4	[Byte]
User stack consumption					
E_OS_USED_SYS_STACK	V	0...FFFFFFFFH	0...17179869180	4	[Byte]
System/Ercosek stack consumption					
E_OS_USED_USER_STACK_MAX	V/S	0...FFFFFFFFH	0...17179869180	4	[Byte]
Maximum user stack consumption (stored at shutdown)					
E_OS_USED_SYS_STACK_MAX	V/S	0...FFFFFFFFH	0...17179869180	4	[Byte]
Maximum system/Ercosek stack consumption (stored at shutdown)					

FUNCTION DESCRIPTION:

General information:

The stack monitor of Ercosek operating system is called to gain information about the stack consumption. The maximum values are stored at ECU shutdown to track the stack consumption over several driving cycles.

The calculations are done, when no other task is pending (background task).

The following variables are provided by the Ercosek operating system:

E_OS_STATE_STACK:

- STACK_OK: Stack consumption of user and system stack in normal ranges (0..75 %)
- USTACK_OVFL, ESTACK_OVFL: Overflow of user or Ercosek stack (> 100 %). This is extremely dangerous and therefore prompt actions to reach 75 % have to be taken.
- USTACK_NFULL, ESTACK_NFULL: User or Ercosek stack is nearly full (75..100 %). Actions to fulfill the requirement of max. 75 % have to be taken.

E_OS_USED_USER_STACK: Used user stack size in byte.

E_OS_USED_SYS_STACK: Used system/Ercosek stack size in byte.

Application conditions:

Initialisation: -

Recurrence: background

Activation: at every engine state

Deactivation: -

Formula section:

If E_OS_USED_USER_STACK > E_OS_USED_USER_STACK_MAX(n-1)
Then E_OS_USED_USER_STACK_MAX = E_OS_USED_USER_STACK

If E_OS_USED_SYS_STACK > E_OS_USED_SYS_STACK_MAX(n-1)
Then E_OS_USED_SYS_STACK_MAX = E_OS_USED_SYS_STACK

1 General

Table of Contents

Index

1.1 General

1.1.1 Application of the system for natural aspirateded system

1.1.1.1 Engine

Description	HMC Theta PI Engine Facelift
No. of cylinders	4
Displacement	1.8l, 2.0l, 2.4l
Charge cycle	DOHC 4 valves/cylinder
Valve Timing	continuously variable intake and exhaust
Intake Manifold	2-Step variable
Intake Port	Variable Charge Motion Flap
Emission Classification	S-ULEV, ULEV-LEVII, EU4, K-LEV, GEN, Leaded

1.1.1.2 Vehicle

Description	NF Facelift MY 2008 and others
Transmission	M5, A5
Power steering	Pressure Feedback via Pressure Switch or Pressure Transducer
Air conditioner	Pressure Feedback via Pressure Switch or Pressure Transducer
Cooling Fan	Relay Type or PWM type
Generator Load Signal	PWM Type

1.1.2 Application of the system for turbo charged system

1.1.2.1 Engine

Description	HMC Theta PI Turbo Engine
No. of cylinders	4
Displacement	2.0l
Charge cycle	DOHC 4 valves/cylinder
Valve Timing	continuously variable intake and exhaust
Emission Classification	ULEV-LEVII, EU4, K-LEV, GEN

1.1.2.2 Vehicle

Description	BK MY 2008 and others
Transmission	M5, A5
Power steering	Pressure Feedback via Pressure Switch or Pressure Transducer
Air conditioner	Pressure Feedback via Pressure Switch or Pressure Transducer
Cooling Fan	Relay Type or PWM type
Generator Load Signal	PWM Type

1.2 System Performance for C-Sample (preliminary)

* Manifold Absolute Pressure Sensor with TIA Acquisition
* Pressure upstream of throttle acqusition (TCI variant)
* Ambient pressure acquisition (integrated in ECU) (for TCI variant)
* Mass Air Flow Acquisition with Backflow-Compensation (alternative to MAP Input via configuration switch) and TIA Acquisition inside MAF Sensor
  (additionally MAF input instead of PUT-Sensor for Calibration purpose possible)
* Ignition (Single Ignition Coil)
* TCO Acquisition
* TOIL Acquisition
* Heated Oxygen Sensor; Linear upstream for S-ULEV
  Binary ZrO2 upstream for all other markets ;binary ZrO2 downstream
* Exhaust wastegate (variable PWM control) turbo charger bypass (TCI variant)
* Recirculation valve (on/off connected to vacuum MAP)–intake compressor bypass (TCI variant)
* Adaptive Lambda Control
* Evaporative Emission Control
* Adaptive Idle Control
* Throttle Position Sensor (non ETC-variant)
* Electronic Throttle Control by PWM H-bridge with 2-channel Throttle Position Sensors (ETC-variant)
* Pedal value Sensor 2 channel (ETC variant)
* Air Conditioner Compressor Control (Relay type)
* OBD-II Diagnosis and Emergency Operation
* Storage of Adaptation data and Failure Memory data in Flash Memory
* CAN Connection to other Vehicle Control Units*** (TCU; TCS; MSR; ESP)
* CAN Connection to Application System

*** **Remark**: The CAN specification is common for all HMC Projects with SV EMS

1.3 Basic System Definitions

1.3.1 Allocation of CRK / CAM angles

CRK disk:
The falling edge of the first tooth after the gap is 114 °CRK before the ignition TDC of the first or the fourth cylinder.

CAM disk:
The electrical falling edge of the Intake CAM signal is positioned at tooth #114 (564°CRK absolute), the electrical rising edge of the Intake CAM signal is located at tooth #54 (204°CRK abs.).
The distinction between tooth #1 and #61 is done depending on the Intake CAM signal level during the gap:
- Intake CAM signal level = high during the gap, then tooth #61;
- Intake CAM signal level = low during the gap, then tooth #1.

The electrical falling edge of the Exhaust CAM signal is positioned at tooth #75 (336°CRK abs.), the electrical rising edge of the Exhaust CAM signal is located at tooth #15 (696°CRK abs.)

A "true power-on" logic of the CAM-Sensor is mandatory for this setup.

The firing order is stored as follows: 1 - 3 - 4 - 2 .

The injection valves are controlled according to the firing order.

An angle at start of injection of 0° means: The injection valve begins to open at the previous ignition TDC of this cylinder. A negative SOI means that the injection will start between the previous TDC of this cylinder and the next TDC of this cylinder.

1.3.2 Cylinder numbering

In the software the cylinders are numbered logically with 0...3, whereas in the ECU-hardware they are numbered 1...4.

If in the specification a cylinder-specific-variable is mentioned with ...[CYL] then always the logic value is ment.

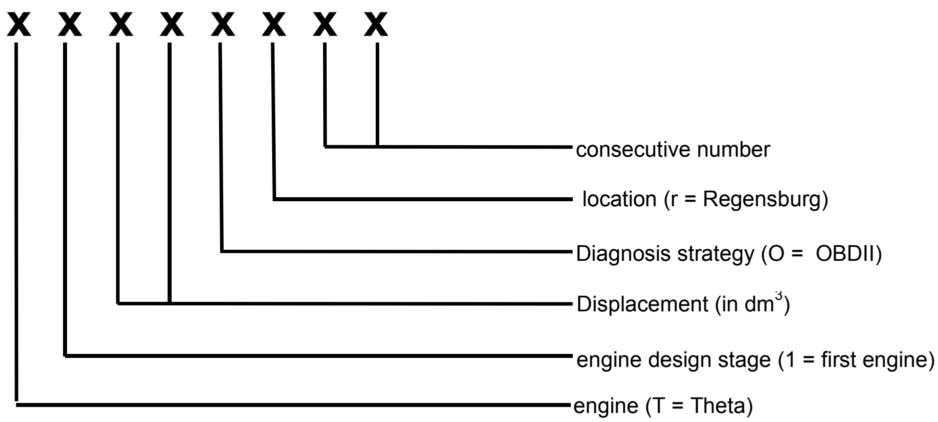
The firing order of the engine follows continuously from logic 0 to logic 3, whereas the physical firing order is 1 - 3 - 4 - 2.

The relation between logical and physical cylinder numbers is as follows:

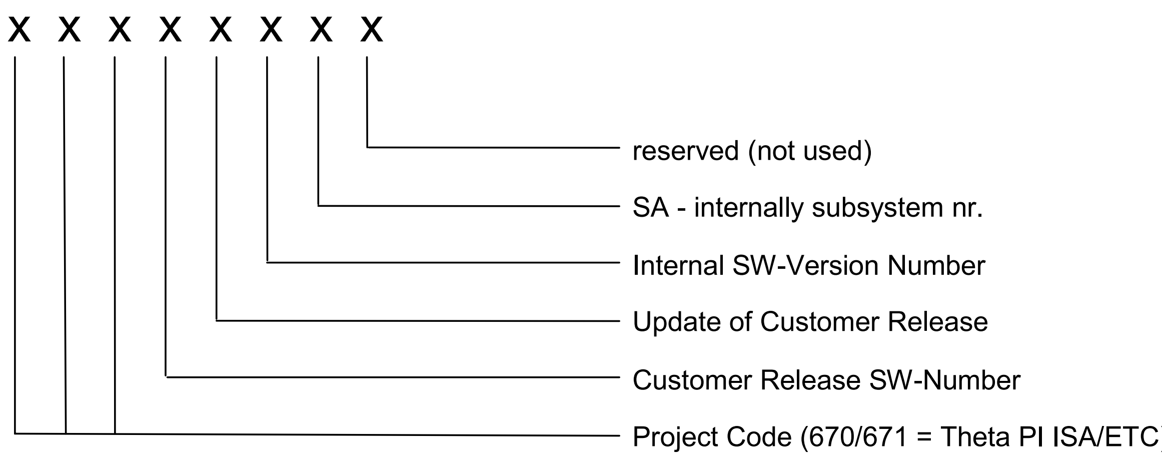
SW-Logic	HW-Logic	Physical
0	1	1
1	2	3
2	3	4
3	4	2

1.3.3 Nomenclature of Program and Data Updates

The data update is described as follows:



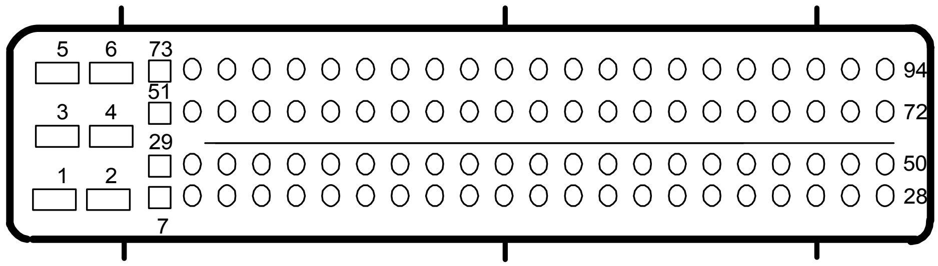
The program update is described as follows:



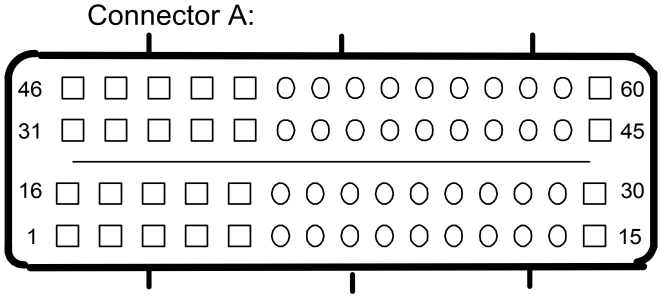
1.4 PIN-reservation of the connector (C-sample)

Figure of 154 - PIN Connector:

Connector K:



Connector A:



Connector A:

Pin	No	Parameter name	Signal name	I	O	Shield	GND
X1-1	A1	IGC1	Ignition 1 (Cyl1-Non Immo, Cyl4-Immo)		X		
X1-2	A2	SHIELD	Shield for ignition			X	
X1-3	A3	N.U.					
X1-4	A4	N.U.					
X1-5	A5	N.U.					
X1-6	A6	N.U.					
X1-7	A7	N.U.					
X1-8	A8	N.U.					
X1-9	A9	N.U.					
X1-10	A10	N.U.					
X1-11	A11	N.U.					
X1-12	A12	N.U.					
X1-13	A13	HDLP	Head Lamp Input (Digital)	X			
X1-14	A14	ELS	Electrical load signal	X			
X1-15	A15	CRUISE_GND	Cruise Control Switch GND				X
X1-16	A16	IGC2	Ignition 2 (Cyl3-Non Immo, Cyl2-Immo)		X		
X1-17	A17	N.U.					
X1-18	A18	N.U.					
X1-19	A19	N.U.					
X1-20	A20	N.U.					
X1-21	A21	N.U.					
X1-22	A22	N.U.					

X1-23	A23	N.U.					
X1-24	A24	N.U.					
X1-25	A25	N.U.					
X1-26	A26	N.U.					
X1-27	A27	N.U.					
X1-28	A28	START_SW	Start Switch (Optional)	X			
X1-29	A29	BLS	Brake Light Switch (ETC)	X			
X1-30	A30	CRUISE	Cruise Control Switch	X			
X1-31	A31	IGC3	Ignition 3 (Cyl4-Non Immo, Cyl1-Immo)		X		
X1-32	A32	N.U.					
X1-33	A33	N.U.					
X1-34	A34	N.U.					
X1-35	A35	N.U.					
X1-36	A36	N.U.					
X1-37	A37	N.U.					
X1-38	A38	N.U.					
X1-39	A39	N.U.					
X1-40	A40	N.U.					
X1-41	A41	N.U.					
X1-42	A42	ACCIN	A/Con Compressor Switch	X			
X1-43	A43	CLUTCH	Clutch Switch (ETC)	X			
X1-44	A44	BTS	Brake Test Switch (ETC)	X			
X1-45	A45	PORT1	Port Flap Actuator (1)		X		
X1-46	A46	IGC4	Ignition 4 (Cyl2-Non Immo, Cyl3-Immo)		X		
X1-47	A47	N.U.					
X1-48	A48	N.U.					
X1-49	A49	N.U.					
X1-50	A50	N.U.					
X1-51	A51	N.U.					
X1-52	A52	N.U.					
X1-53	A53	N.U.					
X1-54	A54	N.U.					
X1-55	A55	N.U.					
X1-56	A56	N.U.					
X1-57	A57	ACIN	A/Con Request	X			
X1-58	A58	PSTE_SW	Power Steering Switch	X			
X1-59	A59	PRS_ACC	A/Con pressure switch	X			
X1-60	A60	PORT2	Port Flap Actuator (2)		X		

Connector K:

Pin	No	Parameter name	Signal name	I	O	Shield	GND
X1-61	K1	PWR_GND	Power Ground				X
X1-62	K2	V_IGK	Battery Voltage After IG Key	X			
X1-63	K3	PWR_GND	Power Ground				X
X1-64	K4	V_EL	Supply after Main Relay	X			
X1-65	K5	ECU_GND	ECU Ground				X
X1-66	K6	VBD	Supply Battery	X			
X1-67	K7	WHEEL_IND+	Inductive Wheel Speed Sensor (+)	X			
X1-68	K8	FTL2	Fuel Tank Level Sensor second Input	X			
X1-69	K9	FTL	Fuel Tank Level Sensor	X			
X1-70	K10	MAP2 MAF_CAL CUR_BAT	Manifold Pressure Sensor 2 (TCI) Mass Air Flow Sensor for Calibration (Non TCI) Battery Charging Current Sensor (Analog BEM Sensor)	X			
X1-71	K11	MAP2_GND MAF_CAL_GND	Manifold Pressure Sensor 2 Ground (TCI) Mass Air Flow Sensor Ground for Calibration (Non TCI)				X

X1-72	K12	KNK	Knock Sensor	X			
X1-73	K13	PVS2_GND	Pedal Value Sensor2_GND (ETC)				X
X1-74	K14	TCO_GND	Coolant Temperature Sensor_GND				X
X1-75	K15	CAM2	Camshaft Position Sensor 2	X			
X1-76	K16	O2S_U_GND	Binary O2 Sensor Upstream Ground (Non SULEV)				X
X1-77	K17	CRK	Crankshaft Position Sensor	X			
X1-78	K18	VIP	Linear Lambda VIP (SULEV)		X		
X1-79	K19	VN	Linear Lambda VN (SULEV)	X			
X1-80	K20	VG	Linear Lambda VG (SULEV)				
X1-81	K21	TAM	Ambient Temperature Sensor (TCI)	X			
X1-82	K22	PORT_FB_AN	Port Flap Analog Feedback Sensor	X			
X1-83	K23	FCO	Fuel Consumption Signal		X		
X1-84	K24	RLY_STARTER	STARTER_RELAY (optional)		X		
X1-85	K25	IV1	Injector 1 (Cyl.1)		X		
X1-86	K26	IV2	Injector 2 (Cyl.3)		X		
X1-87	K27	IV3	Injector 3 (Cyl.4)		X		
X1-88	K28	IV4	Injector 4 (Cyl.2)		X		
X1-89	K29	WHEEL_IND-	Inductive Wheel Speed Sensor (-)	X			
X1-90	K30	MAP_VCC	Supply Manifold Pressure Sensor		X		
X1-91	K31	MAP MAF	Manifold Pressure Sensor Mass Airflow Sensor (alternative)	X			
X1-92	K32	TPS2	Throttle Position Sensor 2 (ETC)	X			
X1-93	K33	TCO	Coolant Temp Sensor	X			
X1-94	K34	KNK_GND	Knock Sensor Ground				X
X1-95	K35	PVS2	Pedal Value Sensor 2 (ETC)	X			
X1-96	K36	PVS2_VCC	Supply Pedal Value Sensor 2 (ETC)		X		
X1-97	K37	CAM2_GND	Camshaft Position Sensor 2 GND				X
X1-98	K38	O2S_U	Binary O2 Sensor Upstream (Non SULEV)	X			
X1-99	K39	CRK_GND	Crankshaft Position Sensor GND				X
X1-100	K40	VS	Vehicle Speed Sensor	X			
X1-101	K41	PSTE_AN	Power Steering Sensor	X			
X1-102	K42	VRC	Linear Lambda VRC (SULEV)				
X1-103	K43	SENS_SUB_VCC	Sensor Supply (DTP, APT, PSTE_AN, VCM_FB,VIS CUR_BAT)		X		
X1-104	K44	PORT_FB_GND/ TAM_GND	Port Flap Feedback Sensor GND Ambient Temp. Sensor Ground				X
X1-105	K45	TPPWM	Throttle Position Signal		X		
X1-106	K46	ALTPWM	Alternator Output (PWM) (only for BEM)		X		
X1-107	K47	WASTE_GATE	Waste Gate Valve Solenoid (TCI)		X		
X1-108	K48	BYPASS	Bypass Solenoid Valve (TCI)		X		
X1-109	K49	SPARE_OUT	Spare Output		X		
X1-110	K50	VIS	Variable Intake Solenoid		X		
X1-111	K51	V_EL	Supply after Main Relay	X			
X1-112	K52	DTP/NVLD_SW	Differential Tank Pressure Sensor NVLD-Switch (optional)	X			
X1-113	K53	TIA	Intake Air Temp Sensor	X			
X1-114	K54	APT	A/Con pressure transducer	X			
X1-115	K55	VIS_FB_GND	VIS Feedback Sensor Ground				X
X1-116	K56	VIS_FB PC_FAN_SW	VIS Feedback Sensor Passenger compartment fan switch	X			
X1-117	K57	APT_GND	A/Con pressure transducer GND				X
X1-118	K58	TOIL TBAT	Oil Temp Sensor Battery Temperature Sensor (only for BEM Analog Sens.)	X			
X1-119	K59	TPS_GND	Throttle Position Sensor Ground				X
X1-120	K60	PVS1_VCC	Supply Pedal Value Sensor 1(ETC)		X		
X1-121	K61	PVS1_GND	Pedal Value Sensor1_GND (ETC)				X
X1-122	K62	CAM1	Camshaft Position Sensor 1	X			
X1-123	K63	TPS_VCC	Supply Throttle Position Sensor		X		

X1-124	K64	RLY_MAIN	Main Power Relay		X		
X1-125	K65	CFA_PWM	PWM Fan Control		X		
X1-126	K66	IVVT1	Variable Valve Timing 1		X		
X1-127	K67	CPPWM	Canister Purge Solenoid		X		
X1-128	K68	IVVT 2	Variable Valve Timing 2		X		
X1-129	K69	IMOB_MIL	Immobilizer Malfunction indicating Lamp		X		
X1-130	K70	RLY_EFP	Electric Fuel Pump Relay		X		
X1-131	K71	ETC1	Electronic Throttle Control 1 (ETC)		X		
X1-132	K72	ETC2	Electronic Throttle Control 2 (ETC)		X		
X1-133	K73	DTP_GND	Differential Tank Pressure Sensor GND				X
X1-134	K74	MAF/MAP_TIA_GND	MAF/MAP_TIA_GND				X
X1-135	K75	IMOB_DL	Immobilizer Data line				
X1-136	K76	LIN	LIN_BUS				
X1-137	K77	CAN_H	CAN_H				
X1-138	K78	CAN_L	CAN_L				
X1-139	K79	TOIL_GND CUR_BAT_TBAT_GND	Oil Temperature Sensor Ground Battery Sensor Ground (only for BEM Analog Sensor)				X
X1-140	K80	TPS1	Throttle Position Sensor 1	X			
X1-141	K81	PSTE-GND	Power steering sensor GND				X
X1-142	K82	PVS1	Pedal Value Sensor 1 (ETC)	X			
X1-143	K83	CAM1_GND	Camshaft Sensor1 GND				X
X1-144	K84	O2S_D	Binary O2 Sensor Downstream	X			
X1-145	K85	O2S_D_GND	Binary O2 Sensor Downstream GND				X
X1-146	K86	ESS	Engine Speed Signal		X		
X1-147	K87	RLY_ACCOUT	A/Con Compressor Relay		X		
X1-148	K88	RLY_CFA_H	Cooling Fan Relay High		X		
X1-149	K89	ACC_MAIN_LAMP ISA1_PWM	Cruise Control Main Lamp (ETC) Idle Speed Actuator 1 (open) (ISA)		X		
X1-150	K90	ACC_SET_LAMP ISA2_PWM	Cruise Control Set Lamp (ETC) Idle Speed Actuator 2 (close) (ISA)		X		
X1-151	K91	SOV/NVLD_OUT	Shut Off Valve NVLD Out (optional)		X		
X1-152	K92	MIL	Malfunction Indicating Lamp		X		
X1-153	K93	O2SH_U	O2 Sensor Heater Upstream		X		
X1-154	K94	O2SH_D	O2 Sensor Heater Downstream		X		

1.5 List of Configuration Data

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ACTIVE_CRK_EDGE	1	0...1H	0...1	1	-
Active crankshaft edge definition - used in the s/w, but not in the spec					
NC_CBK_NR	1	0...2H	0...2	1	-
Remark : not used in the s/w - for spec interface only					
NC_CYC_TOOTH_NR	1	0...FFH	0...255	1	tooth
Number of crankshaft teeth per engine cycle including the missing teeth in the reference gap					
NC_FAC_MAF_INT	1	0...FFFFH	0...892,4	0,0136	ms
Factor for unit adaptation					
NC_FRQ_ASW	1	0...FFFFFFFFH	1...4295e6	1	Hz
Applicative software frequency (Remark : not used in the s/w - for spec interface only)					
NC_FRQ_ECU	1	0...FFFFFFFFH	1...4295e6	1	Hz
ECU processor real frequency					
NC_KNKS_CH1	1	0...FFH	0...255	1	-
Initialization value for the analogue channel of knock sensor					
NC_KNKS_CONF	1	0...FFH	0...255	1	-
Configuration : knock sensor number = f(cylinder number)					
NC_NR_TOOTH_TDC_REF	1	0..FFFF	0..65535	1	tooth
Number of teeth between first tooth and TDC synchro					
NC_T_BAS_FRQ_DIV	1	0...FFH	0...255	1	-
Divisor for calculation of timer frequency for basic time measurements					
NC_T_SEG_LOW_N	1				
Maximum segment duration for low engine speed range.					
NC_T_SEG_MAX	1				
Maximum segment duration for engine stalling detection.					
NC_T_TOOTH_MAX	1				
Maximum duration of normal tooth.					
NC_T_TOOTH_MIN	1				
Minimum duration of normal tooth.					
T1_FREQ	1				
Timer frequency					
T8_FREQ	1				
Timer frequency					
NC_TIPO_MIN	1				
Value of TIPO_MIN (minimum injection time for injection update)					
NC_TOOTH_GRD_MAX	1	0...1000H	0...16	0.0039	-
Maximum tooth gradient for valid or invalid tooth duration.					
NC_TOOTH_GRD_MAX_GAP	1	0...1000H	0...16	0.0039	-
Maximum tooth gradient for valid or invalid crankshaft gap detection.					
NC_TOOTH_GRD_MIN	1	0...1000H	0...16	0.0039	-
Minimum tooth gradient for valid or invalid tooth detection.					
NC_TOOTH_GRD_MIN_GAP	1	0...1000H	0...16	0.0039	-
Minimum tooth gradient for valid or invalid crankshaft gap detection.					
NC_USE_EXT_ADJ	1	0...1H	0...1	1	-
Configuration Constant, 1 = Service tool intervention enabled, 0 = Service tool intervention inhibited					
NC_LAM_SWI	1	0...1H	0...1	1	-
NC_SEG_TOOTH_RR	1	1H...FFH	0...255	1	-
Number of wheel speed signal teeth to build one segment, typical value = 4					
NC_MAX_T_AST_FIRST_EOL_FDOUT	1	0H...FFFFH	0...6553.5	0.1	sec
Maximum misfire detection inhibition time after first engine run					

Input data:

NC_CPPWM_CPS_BOL	NC_TD_LIM	NC_IGBT_CUT_OFF_T	NC_CYL_NR
NC_INI_CTR_DEAC	NC_KEY_OFF_NR	NC_KEY_ON_NR	NC_KNKWB_INI
NC_KNKWE_INI	NC_FAC_MAF_INT_100	NC_CPPWM_CPS_TOL	NC_PSD_DLY
NC_N_DIF_MIN_CRLC	NC_FAST_TRAN_SYN_ENG_IG K_OFF_PWL	NC_NR_TOOTH_GAP_TDC	NC_KEY_OFF_THR
NC_N_MIN	NC_T_SEG_FRQ	NC_LOW_N	NC_N_MAX

Remark : in some cases, the physical and hexadecimal limits don't appear because they are not explicit in the s/w. Only the value is used.

General information:

To satisfy the spec interface (data defined and used), the following convention is applied :

- if a NC value is used as an INPUT in a function, it is defined as a CONFIGURATION DATA in the present spec "List of Configuration Data", and its value is also affected in the present spec.

- if a NC value is defined in a function (in the field " CONFIGURATION DATA), then it is called as an INPUT in the present spec "List of Configuration Data", and its value is affected in this present spec (even if its value is already affected in the function)

Description:

Name	Value
NC_ACTIVE_CRK_EDGE	00H = 0 dec.
NC_CBK_NR	1
NC_CPPWM_CPS_BOL	024EH = 1.797%
NC_CPPWM_CPS_TOL	7DB1H = 98.189%
NC_CYC_TOOTH_NR	78H = 120
NC_FAC_MAF_INT	0198H = 408 dec.
NC_FAC_MAF_INT_100	07F8H = 2040 dec. (called NC_FAC_MAF_INT_100ms in the s/w)
NC_FRQ_ASW	16000000
NC_FRQ_ECU	24000000
NC_IGBT_CUT_OFF_T	4E2H -> 5 msec.
NC_INI_CTR_DEAC	96H -> 600 microsec.
NC_KEY_OFF_NR	32H = 50 dec. = 50 msec
NC_KEY_OFF_THR	157dec = 9DH
NC_KEY_ON_NR	6H = 6 dec. = 6 msec
NC_KNKS_CH1	E0H = 11100000b
NC_KNKS_CONF	0 (one knock sensor for all 4 cylinders)
NC_KNKWB_INI	20H
NC_KNKWE_INI	90H
NC_N_DIF_MIN_CRLC	-150 dec.
NC_NR_TOOTH_TDC_REF	NC_NR_TOOTH_GAP_TDC = 13H = 19 dec. = 19 Teeth
NC_T_BAS_FRQ_DIV	128
NC_T_SEG_FRQ	0x401CD0
NC_T_SEG_LOW_N	(T8_FREQ * NC_CYC_TOOTH_NR / NC_CYL_NR) / NC_LOW_N
NC_T_SEG_MAX	(T8_FREQ * NC_CYC_TOOTH_NR /

	NC_CYL_NR) / NC_N_MIN
NC_T_TOOTH_MAX	T8_FREQ / NC_N_MIN
NC_T_TOOTH_MIN	T8_FREQ / NC_N_MAX
NC_TD_LIM	5625
NC_TIPO_MIN	200 µs
NC_TOOTH_GRD_MAX	0200H = 2
NC_TOOTH_GRD_MAX_GAP	0380H = 3,5
NC_TOOTH_GRD_MIN	0080H = 0,5
NC_TOOTH_GRD_MIN_GAP	0280H = 2,5
NC_USE_EXT_ADJ	0
NC_USE_IV_DIAG_MIS	1
T1_FREQ	NC_FRQ_ECU / NC_T_BAS_FRQ_DIV = 187,5 kHz
T8_FREQ	NC_FRQ_ECU / NC_T_BAS_FRQ_DIV = 187,5 kHz
NC_LAM_SWI	0
NC_SEG_TOOTH_RR	4
NC_FAST_TRAN_SYN_ENG_IGK_OFF_PWL	FFFF = 655.35 s
NC_MAX_T_AST_FIRST_EOL_FDOUT	30 sec
NC_PSD_DLY	2

1.6 AIRT Aggregate general:

Target / Objectives of the function:

The goal of the AIRT is:

_ to realize the acquisition of the sensor voltage

_ to compute:

. the air ambient temperature TAM (ie: outside the intake manifold)
. the three air temperatures TIA_THR, TIA_IM and TIA_CYL (i.e.: inside the intake system) taking into account:
. External EGR version (or not)
. CHRG version (or not)
. One or several air temperature sensors at different locations on the intake system

_ to perform the diagnosis of the TIA sensor(s).

Requirements:

TIA sensor electrical diagnosis (OBD1)

TAM sensor (via CAN) out of range diagnosis (OBD1): via Project specific solutions
TIA and TAM sensor plausibility diagnoses (OBDII)

Physical background of the function:

The AIRT uses basically 2 Physical / Mathematical laws:

_ *The heat transfer law for the mix of two gas:*

Between fresh air and burned gases coming from EGR (we suppose that the isobaric heat capacities are the same for air and EGR gases).

$$m_{egr}C_p\left(T_{egr}-T\right)=m_{air}C_p\left(T-T_{air}\right)\Rightarrow T=T_{egr}\times ratio\_egr+T_{air}\times\left(1-ratio\_egr\right)$$

To estimate the air temperature, we use the following law :

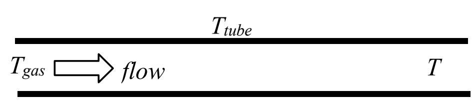
_ *The Reynolds law:*

It describes the heat transfers of a gas flowing into a tube. For gases at moderate temperature, the change of temperature between the final gas temperature and the initial one follows the law:

$$T-T_{gas}=\frac{K\left(L,D\right)\left[T_{tube}-T_{gas}\right]}{flow^{0.2}}$$

Where: K depends on the length of the tube (L) and its diameter (D).

flow is the mass flow of gas.



Overview picture:

General remark concerning all Signal Flow Diagrams presented below:

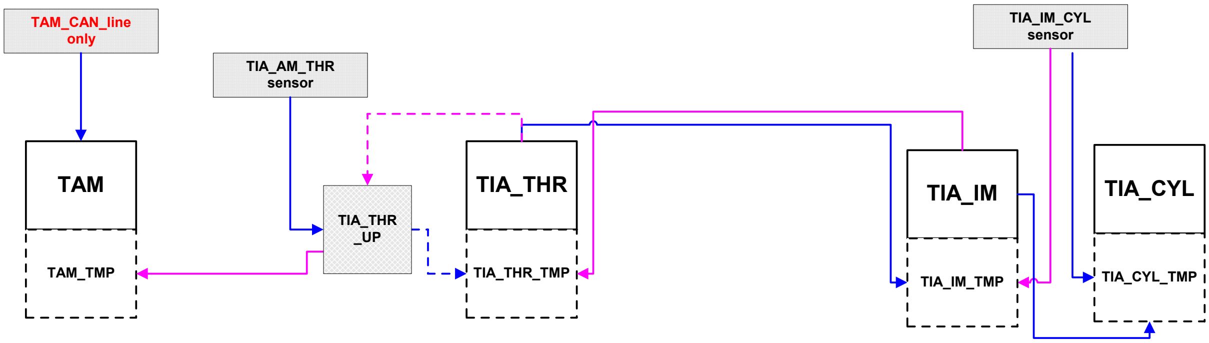
. in blue ink: DIRECT data flow / in pink ink: REVERSE data flow

. “TIA_x_TMP” is the main variable for TIA_x calculation:

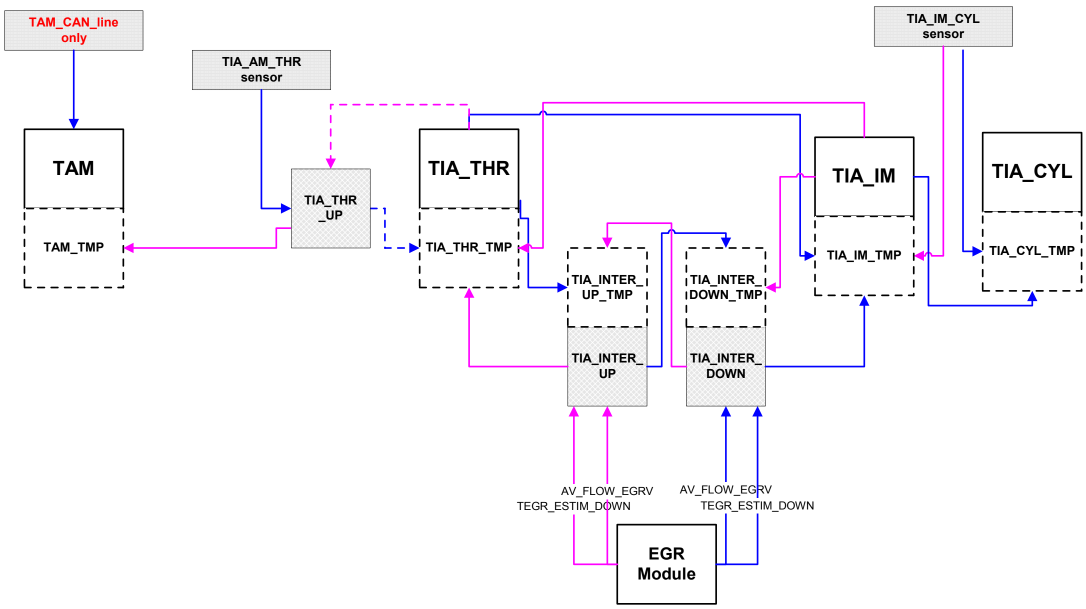
example: TIA_THR_TMP is the main input for TIA_THR calculation.

. EGR module [respectively CHRG module] are defined in EGRC Aggregate [respectively CHRG Aggregate]

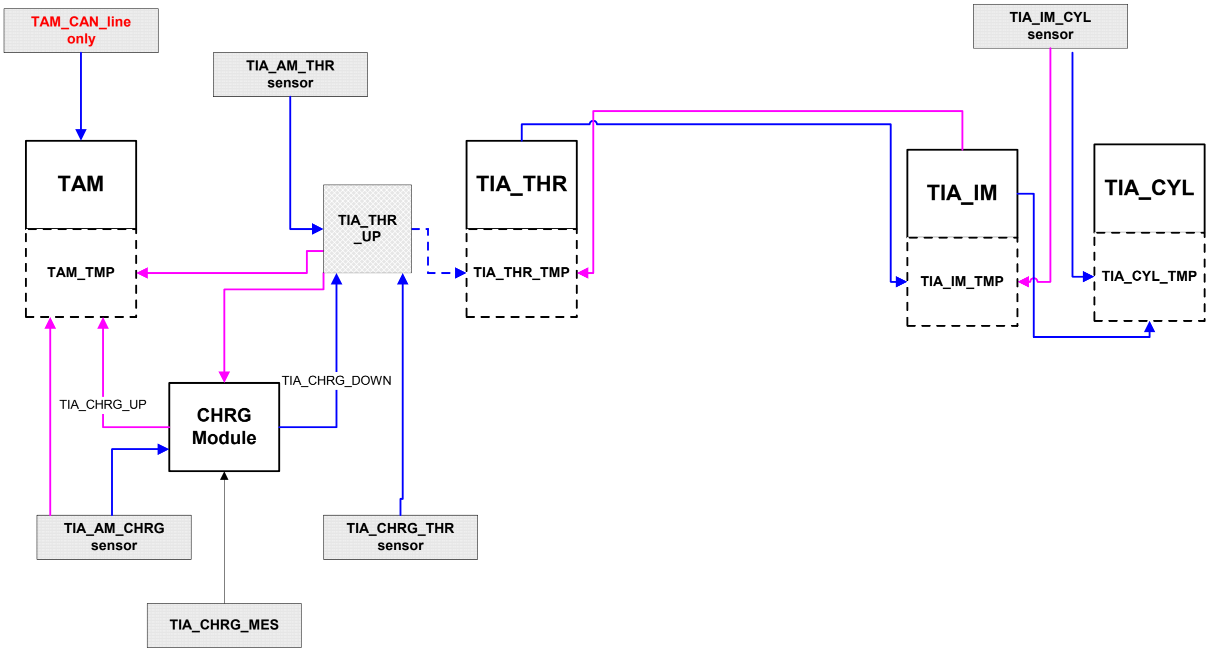
- Signal Flow Diagram for version: No CHRG + No External EGR (+ 2 possible TIA sensors positions)



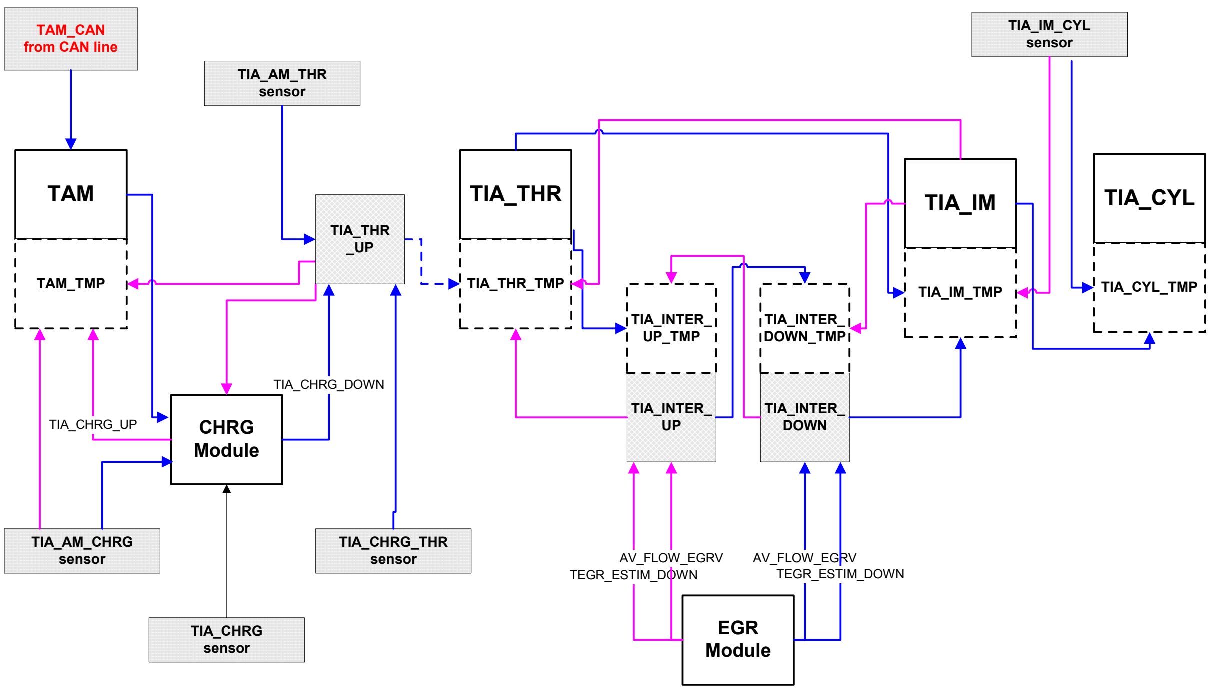
- Signal Flow Diagram for version: No CHRG + External EGR (+ 2 possible TIA sensors positions)



- Signal Flow Diagram for version: CHRG + No External EGR (+ 5 possible TIA sensors positions)



- Signal Flow Diagram for version: CHRG + External EGR (+ 5 possible TIA sensors positions)



1.7 General information

Target

The goal of the EGPR is to calculate the actual valid exhaust pressure in the exhaust manifold.

There is an increase of the pressure in the exhaust pipe in comparison to the ambient pressure. The increase of the pressure is higher with increasing mass flow in the exhaust pipe. The cylinder mass air flow is direct proportional to the exhaust mass flow. Therefore the pressure increase in the exhaust pipe is a function of the engine mass flow.

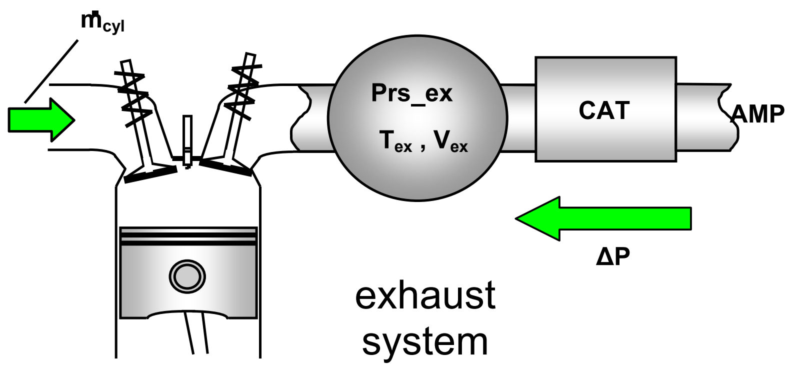
Dependence of IMM on reliable measurement signals

To reach always λ = 1, i.e. in steady state/ transient, cold/warm engine operating, a precise load signal is required. The IMM strongly depends on reliable measurement signals, for two main reasons:

_ In systems equipped with an air-mass flow sensor (HFM) during steady state engine operation, the measured air-mass flow signal (MAF) usually equals the air-mass flow into the cylinder, which is the base for injection time calculation. This is not true during transient engine operation, because of the filling characteristic of the manifold. The HFM is mounted in front of the intake manifold which behaves during transient as a mass storage system. To avoid λ deviations, the dynamic behaviour (in steady state and transient engine operating) of the intake manifold is calculated via the “IMM”.

_ Systems equipped with a manifold pressure (MAP) sensor do not have problems with the filling dynamics of the intake manifold. Due to the fact that the air-mass flow is not directly measured but only calculated out of the air density, therefore it strongly depends on the knowledge of the air-temperature in the cylinder. Any deviations from the temperatures valid during the calibration of the engine have to be compensated.

Overview



List /short description of the containing function

-Exhaust gas pressure determination

- The EGPR is based on an increase of the pressure in the exhaust pipe in comparison to the ambient pressure. This increase is proportional to the engine flow:

$$\dot{m}_{eng} = \dot{m}_{cyl} + \dot{m}_{ff\_sp} \qquad \textbf{(1)}$$

- The exhaust pressure is calculated as :

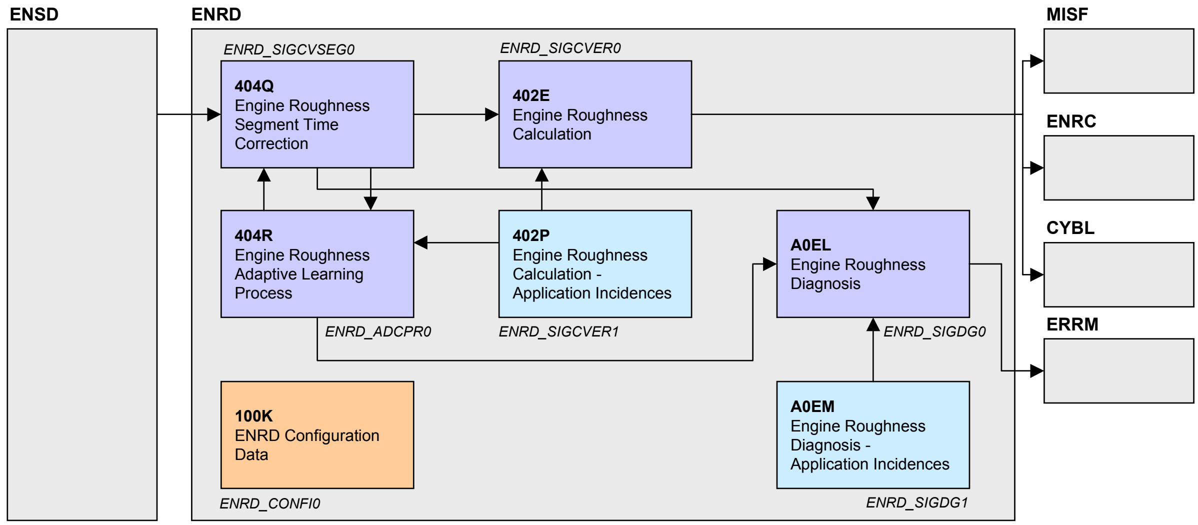
$$p_{ex} = amp + \Delta \cdot f\left(\dot{m}_{eng}\right) \qquad \textbf{(2)}$$

1.8 ENRD General

1.8.1 General description

The goal of the ENRD is to provide an engine roughness information to functions linked to combustion process like misfire monitoring, engine roughness control and cylinder balancing.

1.8.2 Architecture Overview



1.8.3 Description of the containing functions

1.8.3.1.1 404Q Engine roughness segment time correction

Segment times coming from ENSD are managed and corrected according adaptive learning values obtained in 404R module.

1.8.3.1.2 402P Engine roughness calculation - Application Incidences

Application specific requirements, fade-out requests are handled to act on the generic function core. (Template module, modified by the project according its integration environment)

1.8.3.1.3 402E Engine roughness calculation

Engine roughness indexes calculation is grounded on corrected segment time samples provided by 404Q module.

1.8.3.1.4 404R Engine roughness adaptive learning process

Engine roughness adaptive learning process provides adaptive values to 404Q module, to minimise noises produced by flywheel mechanical tolerances.

This module also consumes fade-out requests coming from 402P module.

1.8.3.1.5 A0EM Engine roughness diagnosis - Application Incidences

Application specific requirements, fade-out requests are handled to act on the generic engine roughness diagnosis function. (Template module, modified by the project according its integration environment)

1.8.3.1.6 A0EL Engine roughness diagnosis

This modules provides two diagnosis :

A diagnosis for the ER segment acquisition (thin synchronisation default criterion).

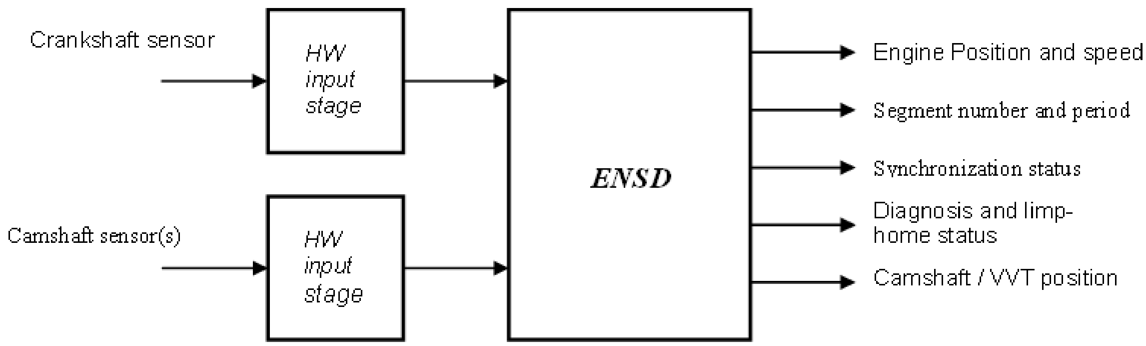
A check range diagnosis for the ER adaptive values obtained by the 404R module.

1.8.3.1.7 100K ENRD Configuration data

This module defines ENRD configuration data used during software integration & compilation. These configuration options defines mainly data and buffer size according the number of cylinders. This module is not attached to a software task.

1.9 ENSD General

1.9.1 General Description



Up to 4 camshaft sensors are supported.

The ENSD aggregate calculates all output data which depend on crankshaft and camshaft sensor information. This includes:

- engine position
- engine speed
- engine speed gradient
- segment number and period
- synchronization status
- diagnosis and limp-home flags
- camshaft / VVT position feedback
- min./max. engine position for pre-injection

At engine start, three synchronization modes are activated:

1) Crankshaft (self-)synchronization:

   The purpose of this synchronization mode is to identify the crankshaft position (0...360° CRK). The crankshaft position is clearly identified at the reference gap of the crankshaft target wheel.

2) Camshaft (self-)synchronization:

   The purpose of this synchronization mode is to identify the camshaft position (0..720° CRK). The camshaft position is clearly identified as soon as an unambigous camshaft edge pattern is found.

3) Camshaft/crankshaft synchronization:

   The purpose of this synchronization mode is to identify the engine position (0..720° CRK). The engine position is clearly identified as soon as an unambigous crankshaft to camshaft position is found.

Injection / Ignition can be enabled as soon as

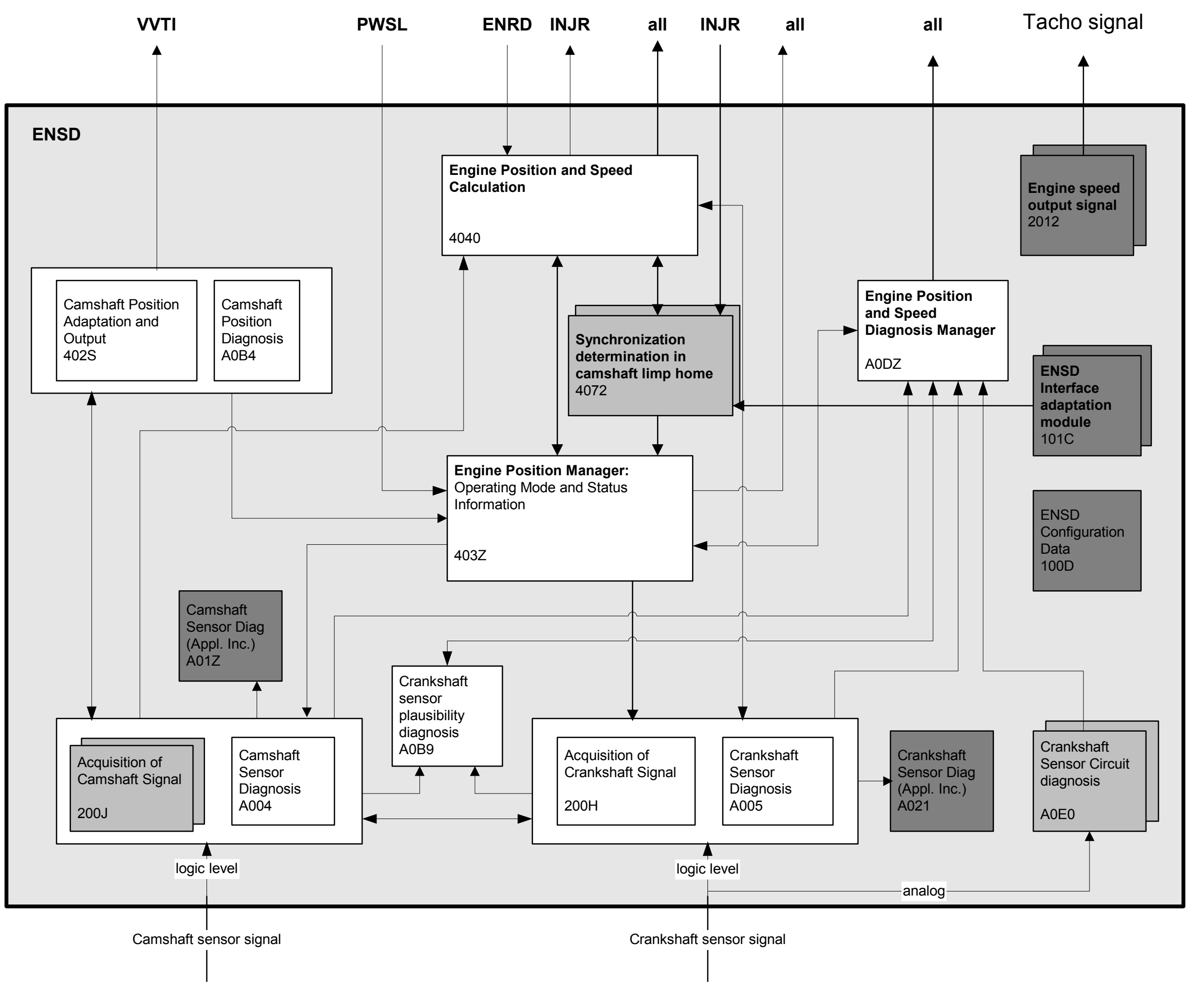
- engine position is identified (MPI engines)

- engine position is validated (DI / DS engines)

A corresponding status information (LV_SYN_ENG / LV_SYN_VLD) is produced and sent to the other aggregates.

1.9.2 Overview

The following figure shows the functional breakdown of the ENSD aggregate:



The light shaded blocs represent modules which are choosen in function of the actual system configuration (Hook modules).

The dark shaded blocs represent modules which have to be modified by the project („templates“ – Hook modules)

1.9.3 Short description of the containing functions

200H: Acquisition of crankshaft signal

Crankshaft synchronization, reference gap detection and crankshaft tooth validation. Measurements of tooth periods and segment periods

200J: Acquisition of camshaft signal

cam/crk synchronization (engine position offset determination), camshaft self-synchronization and engine position determination before engine synchronization (for pre-injection).

403Z: Engine position manager

Selection of operating modes (normal mode, camshaft or crankshaft limp-home mode) and generation of synchronization status information.

4040: Engine position and speed calculation

Calculation of engine position, speed, speed gradient, fast engine speed, segment period and number.

4072: Synchronization determination in camshaft limp-home

Engine synchronization determination in camshaft camshaft limp-home by specific injection pattern management

402S: Camshaft adaptation and position output

Camshaft edge position adaptation and camshaft position feedback for VVT-controller.

A0DZ: Engine position and speed diagnosis manager

Generation of diagnosis information and diagnosis inhibition flags.

A004: Camshaft sensor diagnosis

Camshaft segment period diagnosis and camshaft ratio check.

A0B4: Camshaft position diagnosis

Diagnosis of camshaft to crankshaft reference position.

A005: Crankshaft sensor diagnosis

Crankshaft tooth number and tooth period diagnosis.

A0B9: Crankshaft sensor plausibility diagnosis

Failure detection if crankshaft synchronization cannot be achieved, if camshaft signal is missing and if camshaft signal not valid for engine synchronization.

A021: Crankshaft sensor circuit diagnosis

Detection of open circuit and short circuit for magnetic crankshaft position sensor.

1.10 ERRM General

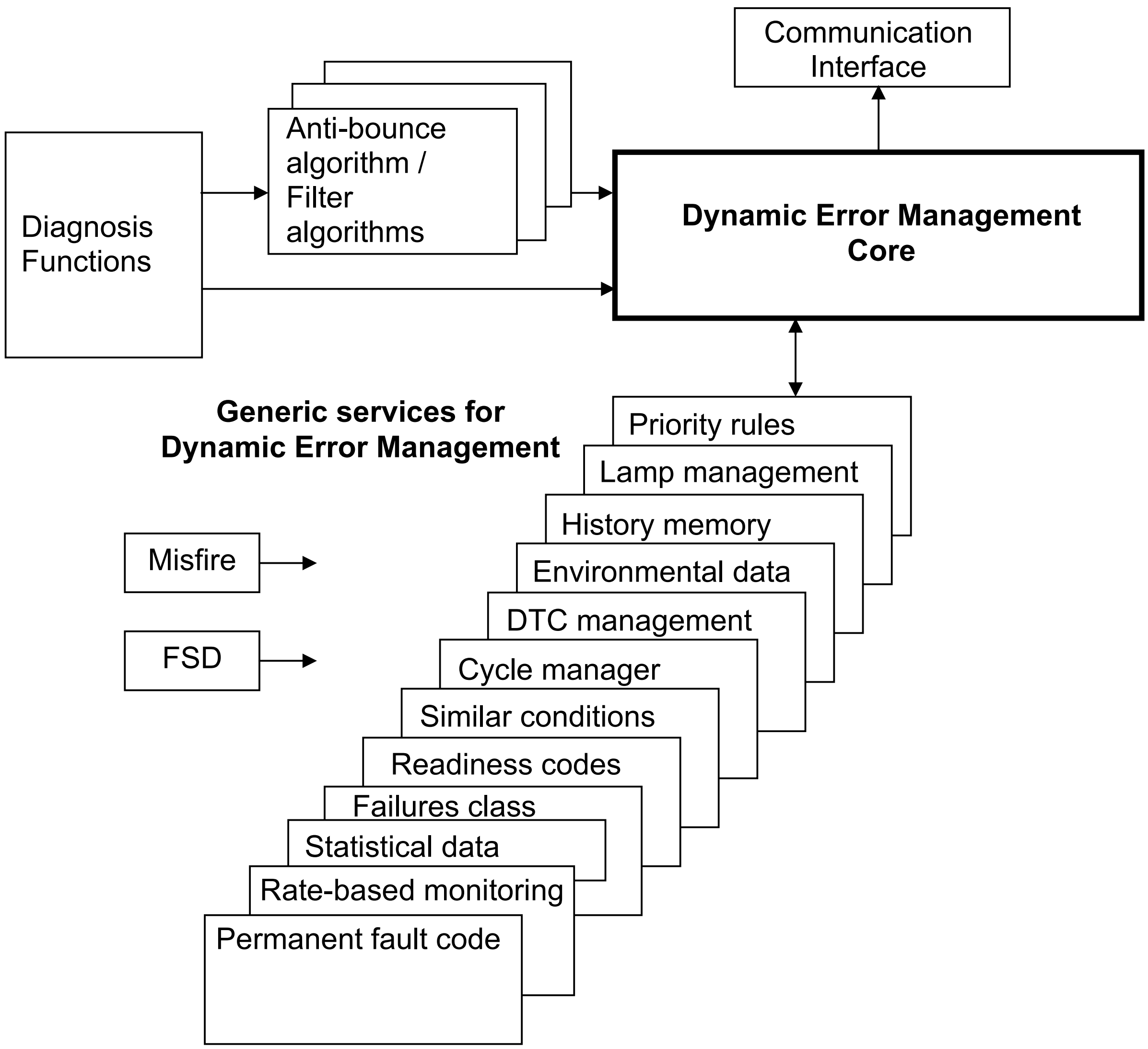
1.10.1 Target / Objectives of ERRM function

The purpose of the Error Management is :

- to collect, store and provide diagnostics information for repair and inspection programs
- to inform the driver via dashboard lamps from malfunctions which could affect emissions compliance with regulations (CARB for US market / EOBD for European market).

1.10.2 Architecture Overview

Principle of the dynamic error management is to receive diagnoses results (filtered or not by anti-bounce, muliti-condition, or statistical algorithms). These results are managed as diagnoses failures, according **CARB** and **EOBD** standards.



1.10.3 Description of the containing functions

- Dynamic Error Management core

The dynamic error management core manages different failures states thanks to the fault detection performed by the diagnoses : detected, present, temporary, pending, disappeared and confirmed. All failures are detected without any limitation on quantity. On the contrary, the number of present failures able to be stored into dynamic memory is limited. Management of this limitation is performed thanks to priority rules mecanism.
When a failure is stored into dynamic memory, the related environment data is stored, the lamp management may force lamp illumination (MIL, WAL,...), and DTCs management is done until failure is deleted. When deleted, failures are stored into history memory.

- History memory

History memory permits to trace historic of deleted failures. Some specific information are stored for each deleted failure. Only the more pertinent data linked to failure shall be stored.

- Debounce algorithm

The debounce algorithms is in charge of detected failures filtering according several available and predefined filters.

- Multi-conditions debounce algorithm

Debounce algorithm able to filter failures according a conditions per symptoms based algorithm.

- Cycle manager

Cycle manager computes driving cycle and warm-up cycle according to engine states and engine coolant temperature. Cycle manager permits to evaluate failure states.

- Failures class

Each failure can be configured to obtain a specific behaviour : emission relevant or not, can illuminate MIL or not ... This module contain to all the predefined behaviours each failure can be associated with. It permits to symplify greatly tuning of diagnosis by decreasing quantity of calibrations.

- DTC management

DTC management permits to catch some data when a failure becomes present and to generate a code identifier, called DTC. It permits to identify default of the function.

- Communication interface

This module allow to access failure memory data related to error management such as freeze frame, stored DTCs, readiness codes.

- Priority rules

This module manages dynamic memory size limitation. It defines criteria, called priority rules, to store or not a new failure into this dynamic memory.

- Environmental data

This module describes the structure of the freeze frame and its management : storage and delete.

- Similar conditions

Similar conditions are additional conditions based on engine and load status to erase a failure in memory (for Misfire and Fuel system failure only).

- Lamp management

Lamp management module can manage illumination of several warning lamps (MIL and other warning lamps) including pre-drive check.

- Readiness codes

Readiness codes allow to know if a diagnosis has been performed or not.

- Rate-based monitoring

Rate-base monitoring functionality : monitoring performances under real world conditions. Performs statistics calculation on diagnoses.

- Statistical data

Statistical data allows monitoring and storage of statistical data on error management data flow.

- Permanent fault code

This module manages the permanent fault codes into a new dedicated dynamic memory.

1.11 EXTC General

The aggregate EXTC –EXhaust system Temperature Control- concerns the component overheating prevention and catalyst heating.

1.11.1 Components overheating prevention functions

There are 3 measures to reduce the temperature in the exhaust line. With high temperature first *ignition retardation* shell be inhibited, because it causes high exhaust temperature due to worse combustion efficiency (used for catalyst heating). The 2nd measure is the enrichment of the air/fuel mixture, what makes the temperature decrease. The 3rd measure is the reduction of the indicated torque. With high catalyst temperature especially after a time with enriched mixture it is necessary to inhibit fuel cutoff in pull phase. This function is located in Appl. Inc. and can be modified by project in "Overheating prevention (Appl. Inc.)". In the figure below the overheating prevention interfaces are marked gray.

1.11.1.1 Minimum ignition limitation

The inhibition of late ignition angles is realized via limitation of the minimum ignition difference to reference ignition - IGA_DIF_MIN_TEG. In general this value does not limit the ignition itself, but reduces the torque reserve! For steady state environment this corresponds to an ignition advance. Dynamic torque reduction via ignition retardation below IGA_MIN_TEG is allowed!

For direct limitation of the ignition the value IGA_DIF_MIN_TEG_BAS has to limit IGA_DIF_MIN. This function has to be done by the project in "(TQM) Minimum ignition angle".

According specification: "IGA_MIN limitation for exhaust gas temperature protection"

1.11.1.2 Enrichment of air-fuel mixture

With richer mixture the exhaust temperature can be lowered. For catalyst overheating prevention there is an I–controller. The number of temperatures which are to be controlled and their location can be specified in Appl. Inc. and configuration. It is recommended to use for controlling static catalyst temperatures to realize a better control response. For turbine overheating prevention -if existent- there is a P-I--controller. The lambda-interface is LAMB_OHP and concerns the minimum of LAMB_COP (for catalyst prevention) and LAMB_TUR_OHP (for turbine prevention).

According specification: "Overheating prevention (Appl. Inc.)"; "Catalyst overheating prevention"; "Turbocharger overheating prevention"

1.11.1.3 Torque reduction

The reduction of the indicated torque shell normally not become effective – depends on catalyst design. The function can be activated if the Lambda-setpoint is on bottom limit and the temperature is still high.

According specification: "Torque based overheating prevention"; "Torque based turbine overheating prevention"

1.11.2 Catalyst heating functions

With EXTC you can impact on several setpoints to optimize them for quick light off of the catalyst and for minimum emissions. Catalyst heating can be activated for after start catalyst heating and for catalyst heating in low load to avoid temperatures below the catalyst working temperature.

The parameters for catalyst heating are:

- TQ_ADD_CH - torque reserve (results in air mass flow and ignition)
- LAMB_CH - air-fuel-mixture
- N_SP_IS_CH - idle speed setpoint
- FUP_RES_H_SP_CH - fuel pressure setpoint – only relevant HPDI engines
- CAM_SP_IVVT_CH_IN/EX - camshaft setpoints (for variable valve timing)
- SOI_1_MPLH_CH - start of 1st injection during MPLH *-only AGGR-Vers. 2*
- EOI_2_MPLH_CH - end of 2nd injection during MPLH *-only AGGR-Vers. 2*
- FAC_MFF_SP_1_MPLH_CH - Split up factor for MPLH *-only AGGR-Vers. 2*

These setpoints are inputs for other aggregates and are used if catalyst heating is active. Some of them are inputs in application incidences of other aggregates and can be used in different ways. The catalyst heating interfaces are marked black.

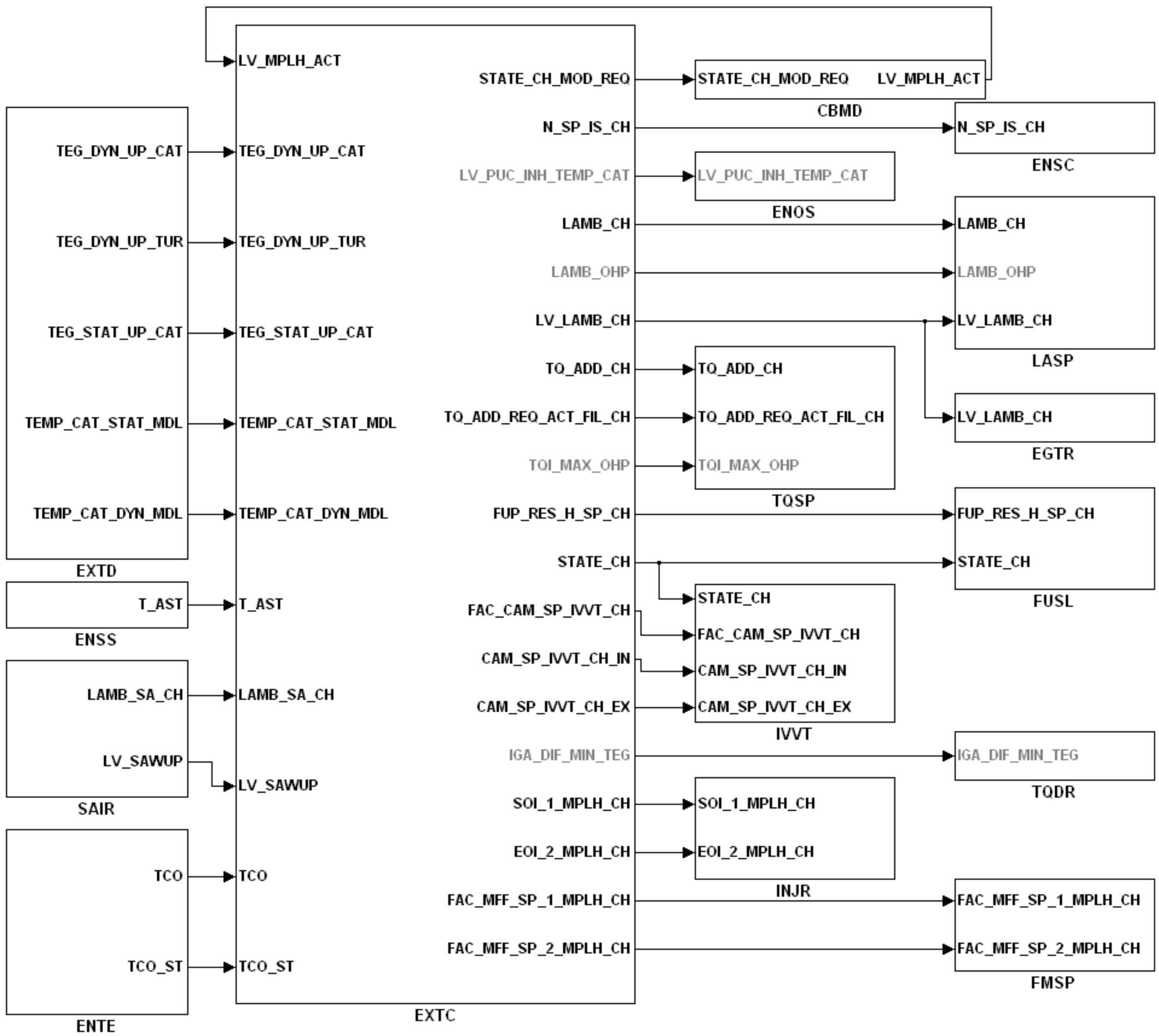


Figure: Main aggregate interfaces

1.12 EXTD General

EXTD (EXhaust system Temperature Determination) is the aggregate where all necessary temperatures in the exhaust line are determined. These temperatures are used for different purposes in different other functionalities:

1. Component overheating prevention: Turbine and Catalyst overheating prevention
2. Lambda sensor heater management
3. Catalyst efficiency diagnosis

Additionally EXTD is responsible for the dew point detection (or dew point passed detection) which is used for the full activation of the lambda sensor heater.

In general there are two ways to determine the temperatures in the exhaust line:

1. Measurement of the exhaust gas temperature
2. Model of the exhaust temperature dependent on the engine operating conditions

The model is always necessary to have a back up information in case of sensor damage and the sensor damage has also to be detected.

So EXTD can be split up in 4 parts:

1. Exhaust gas temperature sensor acquisition
2. Exhaust gas temperature sensor diagnosis
3. Exhaust gas temperature model
4. Dew point detection

1.12.1 Exhaust gas temperature sensor acquisition

Three specifications belong to this part:

1. **EXTD infrastructure requirement specification (IRS)**: Specifies all information concerning the hardware of the ECU and how often the sensor information is updated. If no sensor is available in the system (NC_TEG_SENS_TYP = No Sensor) this specification is skipped.
2. **Exhaust gas temperature sensors (Appl. Inc.)**: Project specific part to inhibit the sensor acquisition for different cases e.g.: If the sensor is only available for certain configurations: Turbo / non Turbo; US market / European market. If no sensor is available in the system (NC_TEG_SENS_TYP = No Sensor) this specification is skipped.
3. **Exhaust gas temperature sensors**: Deals with the sensor information acquisition which can be the raw value of the ADC converter or a PWM value in case of a sensor type with external evaluation control unit which communicated with the ECU by PWM information. Conversion of the raw value in a "raw" exhaust gas temperature and filtering of this "raw" exhaust gas temperature. If no sensor is available in the system (NC_TEG_SENS_TYP = No Sensor) TEG_MES is initialized by zero within this specification.

1.12.2 Exhaust gas temperature sensor diagnosis

This part consists out of two specifications:

1. **Exhaust gas temperature sensor diagnosis (Appl. Inc.)**: Project specific part where the conditions for the activation of the different diagnosis are specified and also the anti bounce behaviour of the different diagnosis are specified. If no sensor is available in the system (NC_TEG_SENS_TYP = No Sensor) this specification is skipped.

2. **Exhaust gas temperature sensor diagnosis**: 4 different types of diagnosis are carried out here: electrical diagnosis, cold start diagnosis, increment diagnosis and gradient diagnosis. If no sensor is available in the system (NC_TEG_SENS_TYP = No Sensor) the global error flag LV_ERR_TEG_MES is initialized by zero.

1.12.3 Exhaust gas temperature model

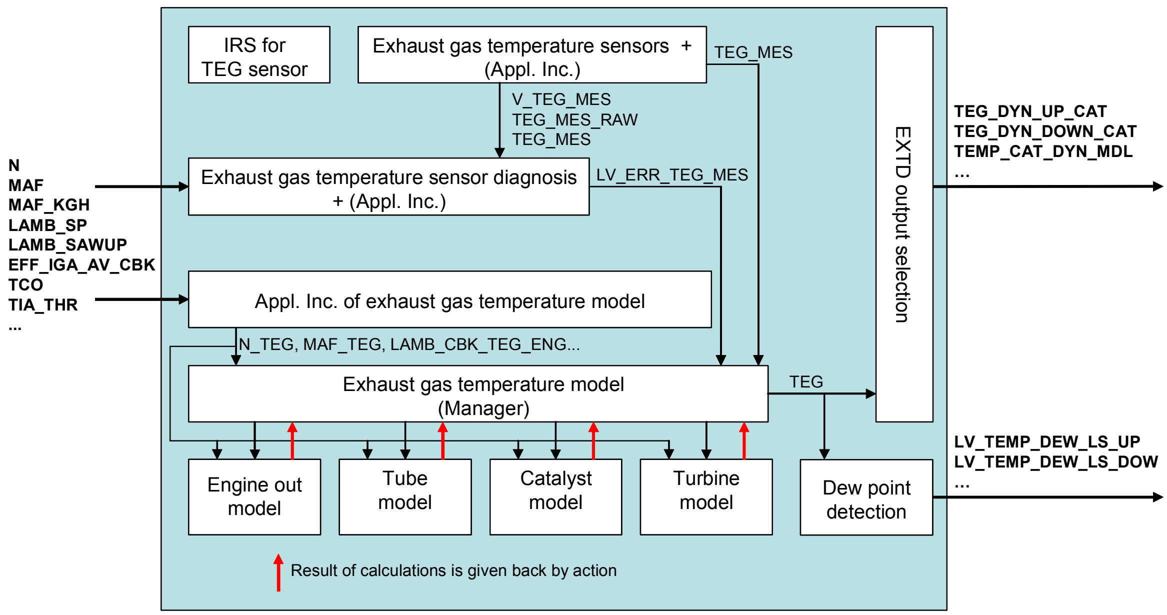
This part is the most important and also the biggest part of EXTD with the following 7 specifications:

1. **Exhaust gas temperature model (Appl. Inc.)**: In this project specific part almost all inputs of the exhaust gas temperature model are generated out of the available system information. E.g.: The engine speed for the exhaust gas temperature model N_TEG is generated out of the system variable N. This is done to deal with different customer request (e.g.: to have a filtered engine speed as input for the exhaust gas temperature model). Additionally several customer specific corrections (e.g.: TEG_ENG_OUT_ADD_CUS) for different purposes (e.g.: VVL and MPLH corrections) can be generated here.
2. **Exhaust gas temperature model**: This module is the manager of the exhaust gas temperature model. All component models (Engine out- / Tube- / Turbine- and Catalyst exhaust gas temperature model) are coordinated here. This means the manager calls the different component models in the right order with the correct update rate and writes the result calculated in the component model in the right position of the arrays (TEG_DYN_MDL, TEG, LV_TEMP_DEW_MDL...). It handles also at which position the sensor information is used instead of the model based value. There is only one calibration data in the specification LC_TEG_MDL_SENS_ENA which has to be set to one if the sensor information shall be used (if no sensor error detected). So for calibration this specification is not important.
3. **Engine out exhaust gas temperature model**: In this module all engine operating point dependencies (e.g.: engine speed, load, ignition angle ...) are used to determine the first temperature which is effective engine out exhaust gas temperature. Additional the engine out exhaust gas humidity is determined here.
4. **Tube exhaust gas temperature model**: This model part is used to simulate the cooling down of the exhaust gas and the influence on the dynamic of the exhaust gas temperature by the influence of the heat capacity of an exhaust pipe like the exhaust manifold and by the external conditions like ambient temperature and vehicle speed.
5. **Turbine exhaust gas temperature model**: This model part is used to simulate the cooling down of the exhaust gas due to the turbine power and the influence on the dynamic of the exhaust gas temperature by the influence of the heat capacity of the turbine housing and by the external conditions like ambient temperature and vehicle speed.
6. **Catalyst exhaust gas temperature model**: This model part is used to simulate the catalyst temperature and it's dynamic. Therefore the exothermic reactions are considered to calculate the increased temperature in the catalyst and also ambient temperature and vehicle speed are again used to calculate the exhaust gas temperature downstream catalyst.
7. **EXTD Output selection**: At the end of the calculation all exhaust gas temperatures are available in the array TEG[NC_NR_TEG_MDL] but the output of the aggregate are values with a certain naming (e.g.: TEG_DYN_UP_CAT). For each value with a certain naming a value out of TEG has to be selected. This mapping is done in this module.

1.12.4 Dew point detection

The dew point detection is a project specific specification because there are several strategies for this functionality. All information's out of the exhaust gas temperature model, like modelled tube wall temperatures and the internal dew point flags for each position in the exhaust line LV_TEMP_DEW_MDL[NC_NR_TEG_MDL], can be used to determine the "dew point passed flag" LV_TEMP_DEW_LS_UP/DOWN for the upstream and downstream sensor. But also mass air flow integral can be used to set the dew point flag. Of course also combinations out of these information's are possible.

Module overview:



1.12.5 Introduction

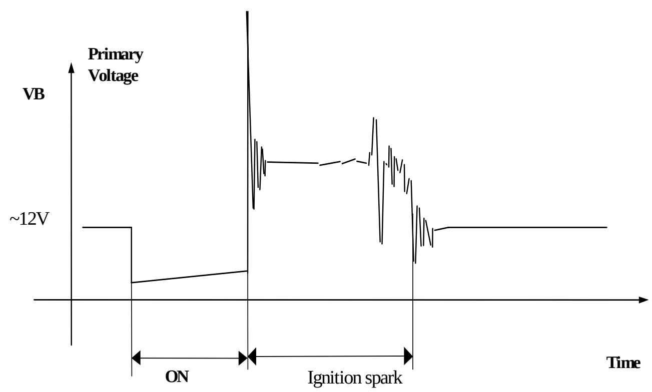
This document is the Aggregate Integration Document of the IGRE Aggregate. Its aim is to describe the integration constraints of this aggregate with the others. It aims with interfaces (imported and exported), architecture, and configuration. This document covers all versions of the IGRE aggregate.

1.12.6 Purpose

The IGRE Aggregate defines the calculation of the dwell time, the ignition realization and the ignition diagnosis functions.

Basically the ignition command function has to fulfill two main tasks; to ensure that there is enough energy in the ignition coil and to start the combustion at the requested ignition angle. Knowing the desired ignition advance, engine speed and several other parameters, this aggregate determines the correct timing for switching ON and OFF the power stage in order to bring the ignition coil to the necessary primary current value. The time during ON state (load phase) of the switch is often improperly called Dwell time (with reference to the dwell ratio in breaker systems).

At the end of this Dwell time the ignition stage is switched OFF and the ignition spark is created. The OFF event corresponds to the advance ignition angle (IGA_IGC).



The ignition command functions and the linked diagnostic functions are regrouped in the ignition realisation aggregate.

1.12.7 Function Description

This function sets the angular position of the dwell time turn on in order to have the time to set up the necessary current in the ignition coil.

The strategy is based on a priority of the ignition angle. If dwell time priority is requested the minimum dwell time is equal to maximum dwell time.

The strategy respects ignition coils with dual and single outputs. This function could be used independently of the number of cylinders (x) and the geometry of the crankshaft target wheel.

All of the functions described in this section operate over the entire engine speed range.

- Realization of minimum 1 ignition spark per cycle
- Realization of a number of multiple spark

- Physical limits of the ignition angle +60°CRK / -35,625°CRK
- In case of an acceleration of ±10 000 RPM/s the ignition spark output tolerance depending on engine speed.

1.13 IGSP General

1.13.1 General description

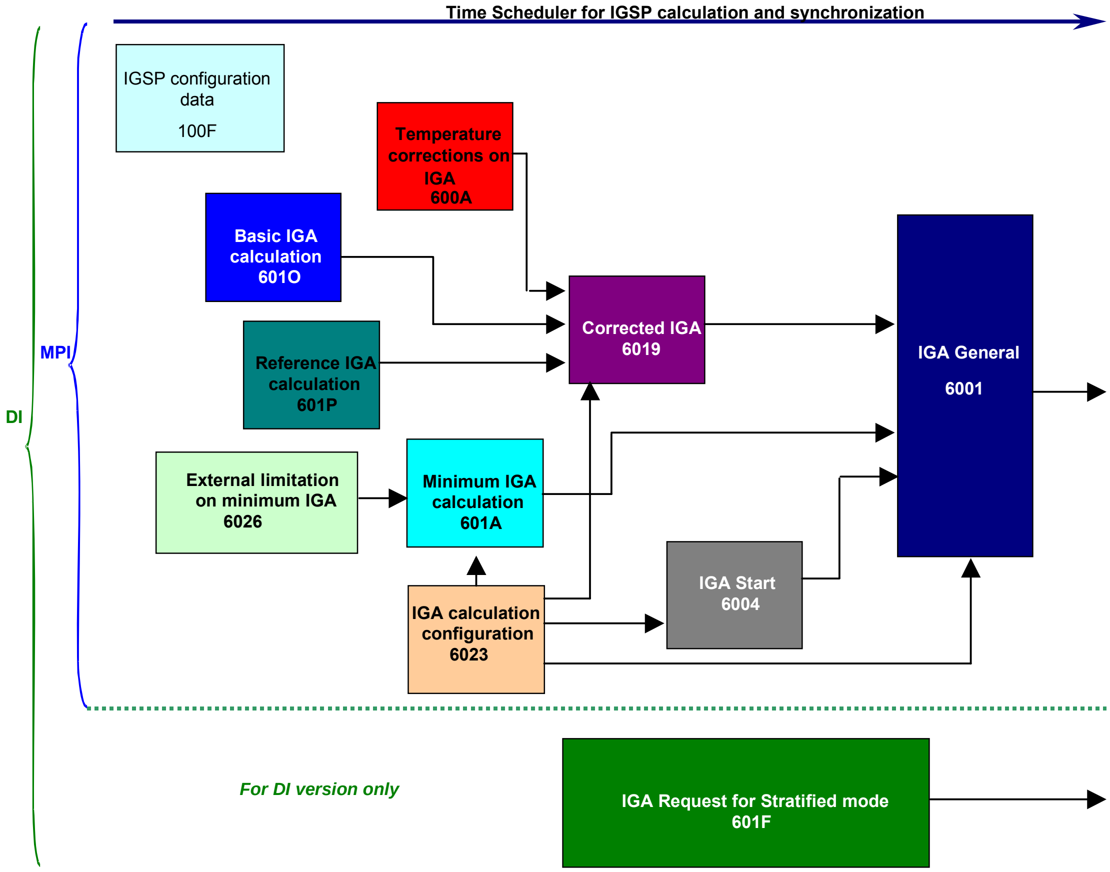
The IGSP Aggregate defines the calculation of the ignition angles to be applied on an engine with a TQ based EMS.

The Ignition functionality of an engine management system can be organised in two different parts. A first part is the determination of the ignition angle itself, and a second one is the realisation of this ignition angle with the management of the charge and the discharge of coils. This document is the “over head” for the first part, “Ignition Angle Calculation / Ignition Angle Setpoint”. The information of this function will thus been send to the coil for the proper application on the spark of the mandatory determined ignition angle.

The goal of this aggregate is to calculate the 3 ignition angles (BAS, REF and MIN) that describe the limits of engine behaviour and capacities and then to co-ordinate this basic structural angle with the ignition angle request from the Torque manager to produce the requested torque in the requested period, with the limitation to the engine possibilities. The ignition angle hence calculated will be provided to the second part in order to be realised as best by the ignition coil.

The Ignition structure gets an ignition angle request from torque structure (named IGA_DIF_TQ_REQ) which is the transformation in term of ignition angle of the setpoint for TQ. It provides representative ignition angles to the other functionality. With this information, the torque structure can interpret and define which ignition angle has to be realised. Then the ignition angle choice between TQ requests and engine possibilities with IGA is done to produce the best fitted (in term of TQ production and engine limits respect) ignition angle that will be applied to each cylinder.

1.13.2 Architecture Overview



1.13.3 Description of the containing functions

<u>Ignition angle calculation structure</u>

The calculation of the ignition angle to be applied is done by various functional calculations steps represented by the modules here listed:

- *IGSP Configuration data*
  Here are defined – by each project – the configurations values to be used for IGSP aggregate

- *IGA_BAS calculation*
  For the calculation of the basic ignition angle depending on the engine configuration Homogenous or Lean Burn and cylinder bank specific.

- *Reference IGA calculation*
  Here the reference ignition for best torque production is computed.

- *Temperature correction on ignition angle*
  Here the temperature corrections on IGA_BAS and IGA_REF are computed.

- *External limitation on minimum ignition angle*
  Some external limitations can be admitted on the minimum ignition in some particular engine running area.
  Each projects have to provide their specific considerations for this function.

- *Configuration of IGA calculation*
  Here are defined – by each project - the name of external corrections that will be applied to ignition

- *Minimum ignition angle*
  This module defines the minimum possible ignition angle for combustion stability.

- *Corrected IGA*
  Here the corrections on IGA_BAS and IGA_REF are performed: gathering and calculation of all corrections on the ignition angles.

- *Basic ignition angle at start*
  This module enables a specific management of IGA during start phase.

- *Ignition angle general*
  This block manages all the ignition requests – regarding torque request and engine capacities – and defines the IGA to be applied at the coil.

In case of switch to stratified mode only the module hereafter is calculated in the software.

- *IGA Request for stratified mode*
  This module co-ordinates the calculation of the ignition angle IGA_AV_S[x] which is applied to each different cylinder of the engine in Stratified combustion mode. This value is directly related and fitted for one particular cylinder (*cylinder x*).

1.14 General information

1.14.1 Target

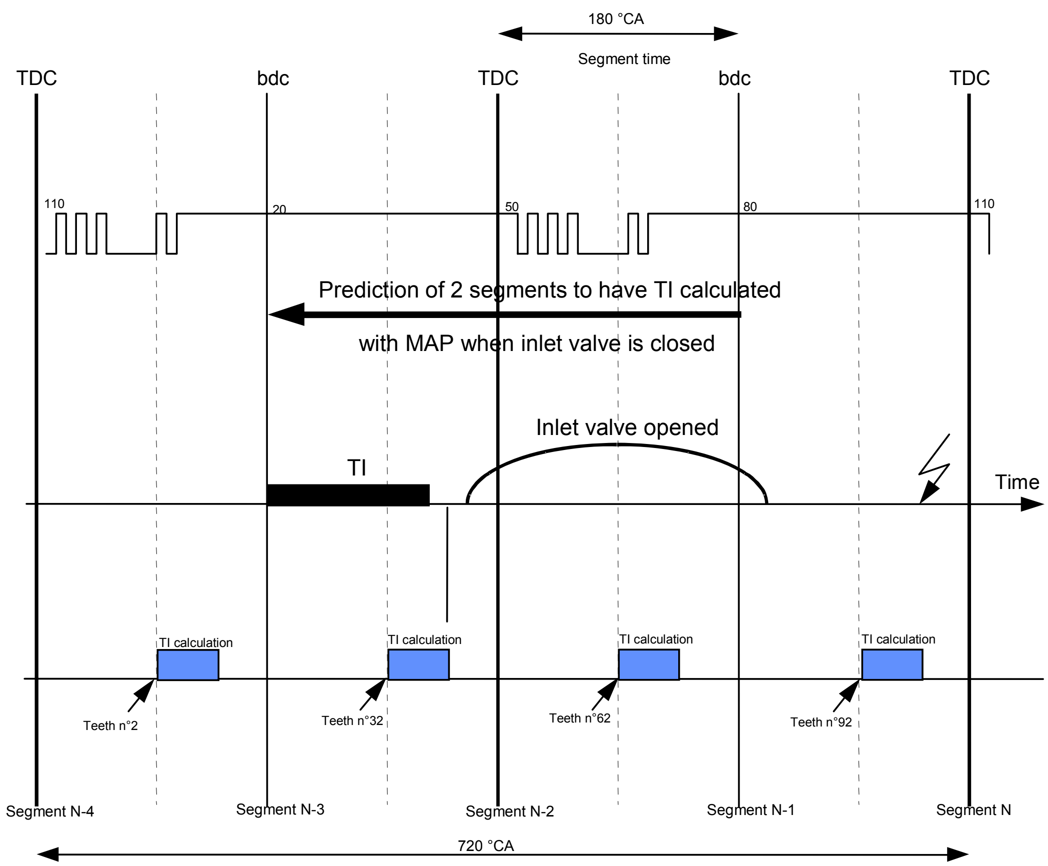
The purpose of the "**In**take **Sy**stem" (INSY) is to coordinate the air flow controlling components and to predict the air mass flow behavior. Needed is the prediction, as the injection time is calculated based on the real mass-flow, which is present in the cylinder for the considered combustion cycle. However, the injection time is calculated before the closure of the intake valves, and parameters like MAP or MAF are still not known.

All this modeling aims to evaluate, at the moment of the injection time calculation, the conditions in pressure and flow which are present at the closure of the intake valves for the concerned combustion cycle. Therefore, the predicted pressure is calculated. This predicted pressure is the estimation of the pressure value at the moment of the injection.

⇒ Calculation of MAF_THR, MAF_CYL and MAP in order to calculate MAP_PRED and MAF.

1.14.2 Scheme

Example for a 4 cylinder engine:



1.14.3 Dependence of IMM on reliable measurement signals

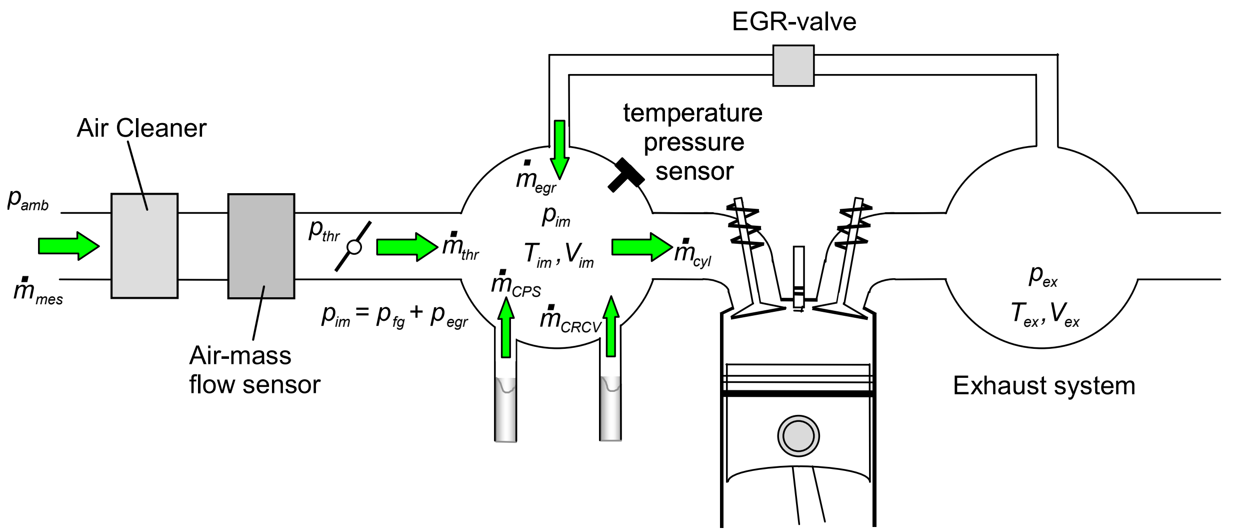
To reach always λ = 1, i.e. in steady state / transient, cold / warm engine operating a precise load signal is required. The IMM strongly depends on reliable measurement signals, for two main reasons:

In **systems equipped with an air-mass flow sensor (HFM)** during steady state engine operation, the measured air-mass flow signal (MAF_KGH) usually equals the air-mass flow into the cylinder, which is the base for injection time calculation. This is not true during transient engine operation, because of the filling characteristic of the manifold. The air-mass flow sensor is mounted in front of the intake manifold, which behaves during transient as a mass storage system. To avoid λ deviations, the dynamic behavior (in steady state and transient engine operating) of the intake manifold is calculated via the “INSY”.

Systems equipped with a manifold pressure (MAP) sensor do not have problems with the filling dynamics of the intake manifold. Because the air-mass flow is not directly measured but only calculated out of the air density, **it strongly depends on the knowledge of the air-temperature in the manifold**. Any deviations from the temperatures valid during the calibration of the engine have to be compensated.

1.15 Physical background

1.15.1 Scheme of the system used



Variables used:

p_{AMP}	AMP	Ambient pressure
p_{thr}	PUT	Pressure up stream the throttle
$\dot{m}_{THR}$	MAF_THR	Air-mass flow at the throttle
$\dot{m}_{CYL}$	MAF_CYL	Air-mass flow into the cylinder
$\dot{m}_{EGR}$	MAF_EGR	Exhaust air-mass flow
$\dot{m}_{CPS}$	FLOW_CPS	Air-mass flow through the canister purge solenoid
$\dot{m}_{CRCV}$	FLOW_CRCV	crank case air flow
T_{im}	TIA	Intake air temperature
V_{im}	VOL_IM	Volume of the intake manifold
p_{ex}	PRS_EX	Pressure in exhaust system
T_{ex}	TEG_EGR	Temperature of the exhaust side of EGR valve

1.15.2 Physical/ mathematical laws used in the intake manifold model

- The IMM is based on the **conservation of mass** over the intake manifold. Assuming that the air is an ideal gas this gives the basic equation:

$$\frac{\partial}{\partial t}\left(\frac{p_{im}V_{im}}{R_{air}T_{im}}\right) = \dot{m}_{in} - \dot{m}_{out} \qquad (3)$$

The inflowing air-mass flow is the sum of the throttle air-mass flow, the exhaust gas recirculation, the flow through the canister purge solenoid (CPS) and the crank case air flow (CRCV) $\dot{m}_{in} = \dot{m}_{thr} + \dot{m}_{egr} + \dot{m}_{cps} + \dot{m}_{crcv}$. The outflow is the air-mass flow into the cylinder.

Under the assumption that T_{im} and V_{im} *are* not time dependent (or at least quasi stationary) the **overall pressure change** can be modeled as:

$$\frac{\partial}{\partial t}p_{im} = \frac{R_{air}T_{im}}{V_{im}}\left(\dot{m}_{THR} + \dot{m}_{EGR} + \dot{m}_{CPS} + \dot{m}_{CRCV} - \dot{m}_{CYL}\right) \qquad (4)$$

- **The air-mass flow through the throttle** can be described with the flow of ideal gas through throttle. Throttling will be taken into account with the reduced area A_{red}

$$\dot{m}_{thr} = A_{red}\cdot\sqrt{\frac{2\cdot\chi}{\chi-1}\cdot\frac{1}{R_{air}\cdot T_{im}}}\cdot p_{thr}\cdot\Psi \qquad (5)$$

With

$$\Psi = \begin{cases} \sqrt{\left(\frac{p_{im}}{p_{thr}}\right)^{\frac{2}{\chi}} - \left(\frac{p_{im}}{p_{thr}}\right)^{\frac{\chi+1}{\chi}}} & \text{for undercritical pressure ratio } \left(\frac{p_{im}}{p_{thr}} > 0.53\right) \\ \Psi_{critical} = 0.2588 & \text{for overcritical pressure ratio } \left(\frac{p_{im}}{p_{thr}} < 0.53\right) \end{cases} \qquad (6)$$

The reduced area of the throttle A_{red} is a function of the throttle position TPS. The critical pressure ratio is the pressure ratio where sonic speed is reached at the throttle

$$\left.\frac{p_{im}}{p_{thr}}\right|_{critical} = \left(\frac{2}{\chi+1}\right)^{\frac{\chi}{\chi-1}} \approx 0.53 \qquad (7)$$

- The **flow into the cylinder** is modeled via the (linearized) volumetric efficiency

$$\dot{m}_{cyl} = \eta_{slop} \cdot p_{fg} + \eta_{slop} \cdot p_{EGR} - \eta_{ofs} \cdot \frac{p_{fg}}{p_{im}} - \eta_{ofs} \cdot \frac{p_{EGR}}{p_{im}} \tag{8}$$

where p_{fg} means the fresh gas partial pressure in the manifold (if no EGR available, it is the same as the manifold pressure). The volumetric efficiency is the suction air of the engine. It mainly depends on the engine speed, the exhaust gas pressure and engine hardware (e.g. variable valve timing, port flap position, variable intake manifold).

- The **intake manifold pressure** p_{im} is calculated by integration of eq. (2) with a trapezoidal algorithm

$$p_{im}^{N} = p_{im}^{N-1} + \frac{\Delta t^{N}}{2} \cdot \left(\dot{p}_{im}^{N-1} + \dot{p}_{im}^{N}\right) \tag{9}$$

- Similar to (2) the model for the exhaust **gas partial pressure** in the manifold can be derived (mass balance for exhaust gas)

$$\frac{\partial}{\partial t} p_{EGR} = \frac{R_{EGR} T_{im}}{V_{im}} \left(\dot{m}_{EGR} - \dot{m}_{cyl}^{EGR}\right) \tag{10}$$

To simplify differences between R_{air} and R_{EGR} at T_{im} are neglected (strictly, this is only valid for $\lambda = 1$).

- The **flow at the exhaust gas recirculation valve** can be modeled similar to that at the throttle

$$\dot{m}_{thr}^{EGR} = A_{red}^{EGR} \cdot \sqrt{\frac{2 \cdot \chi}{\chi - 1} \cdot \frac{1}{R_{air} \cdot T_{im}}} \cdot p_{ex} \cdot \Psi^{EGR} \tag{11}$$

with

$$\Psi^{EGR} = \begin{cases} \sqrt{\left(\frac{p_{im}}{p_{ex}}\right)^{\frac{2}{\chi}} - \left(\frac{p_{im}}{p_{ex}}\right)^{\frac{\chi+1}{\chi}}} & \textit{for}\ \text{undercritical}\ \textit{pressure ratio} \\ \Psi_{critical} = 0.2588 & \textit{for overcritical pressure ratio} \end{cases} \tag{12}$$

and p_{ex} meaning the pressure at the upstream side of the EGR valve (exhaust gas pressure).

The air-mass flow of exhaust gas into the cylinder can be calculated based on

$$\dot{m}_{cyl} = \dot{m}_{cyl}^{fg} + \dot{m}_{cyl}^{EGR} = \eta_{slop} \cdot p_{im} - \eta_{ofs} = \eta_{slop} \cdot (p_{fg} + p_{EGR}) - \eta_{ofs} \cdot \frac{(p_{fg} + p_{EGR})}{p_{im}} \tag{13}$$

as

$$\dot{m}_{cyl}^{EGR} = \eta_{slop} \cdot p_{EGR} - \eta_{ofs} \cdot \frac{p_{EGR}}{p_{im}} \tag{14}$$

The exhaust gas partial pressure is integrated with a rectangular algorithm

$$p_{EGR}^{N} = p_{EGR}^{N-1} + \Delta t^{N} \cdot \dot{p}_{EGR}^{N} \tag{15}$$

With the intake manifold pressure p_{im}^{N} (calculated with (2) - (7)) and the exhaust gas partial pressure p_{EGR}^{N} (calculated with (8) – (13)) it is possible to calculate the **fresh gas partial pressure**

$$p_{fg} = p_{im}^{N} - p_{EGR}^{N} \tag{16}$$

and then the fresh gas mass flow into the cylinder.

$$\dot{m}_{cyl,fg} = \eta_{slop} \cdot p_{fg} - \eta_{ofs} \cdot \frac{p_{fg}}{p_{im}} \tag{17}$$

In systems without EGR, the fresh-gas partial pressure equals the intake manifold pressure

$$p_{fg} = p_{im}^{N} \tag{18}$$

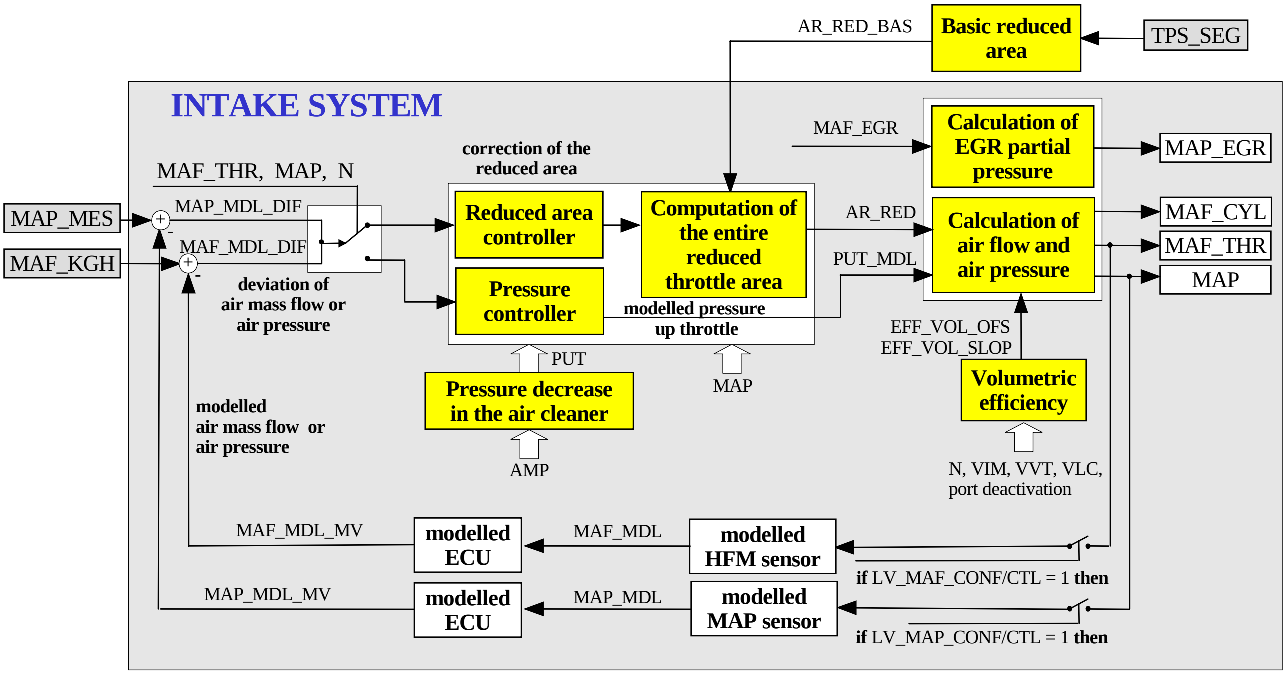
- Due to injection-timing constraints, it is necessary **to predict the pressure in the intake manifold** (usually about 2 segments in advance). The prediction ensures, that the air-mass flow used for the injection timing calculation is as close as possible to the pressure at closing the respective inlet valve. The prediction is done by extrapolation of the actual pressure with the mean value of the last two pressure gradients multiplied by a calibratable constant C

$$p_{fg}^{pred} = p_{fg} + \frac{(\dot{p}^{N} + \dot{p}^{N-1})}{2} \cdot C \cdot \Delta t^{N} \tag{19}$$

With the predicted pressure the (predicted) air flow into the cylinder (at closing the intake valve) is

$$\dot{m}_{cyl,fg}^{pred} = \eta_{slop} \cdot p_{fg}^{pred} - \eta_{ofs} \tag{20}$$

1.15.3 Algorithm for the complete Intake Manifold System



The complete calculation algorithm for the Intake manifold system is subdivided into six major separate modules:

- application incidence for project specific adaptation
- calculation of the air/EGR partial pressures and air mass flows
- INSY-controller with reduced area controller and pressure up throttle controller
- ambient pressure adaptation with ambient pressure learning
- reduced area adaptation
- calculation of the pressure decrease in the air filter
- volumetric efficiency calculation
- basic reduced area calculation

In addition, there are modules that strictly do not belong to the IMM, but usually are handled with the IMM:

- data acquisition for MAF values
- data acquisition for MAP values

The tuning of the complete IMM is therefore consequently also subdivided into the tuning of the separate depicted modules.

1.16 Measurement Set-up

1.16.1 Required sensor equipment

The following sensors are required for a calibration of the IMM:

- Pressure sensor in the intake manifold
- Linear lambda sensor
- Air-mass flow sensor and / or a fuel-consumption measuring device
- Barometer in the test-cell
- Coolant temperature sensor
- Intake air temperature sensor
- Pressure sensor between air filter and throttle
- Pressure sensor in the exhaust system (if EGR is configured)
- Pressure sensor just in front of the EGR-valve (if EGR is configured)
- Temperature sensor in front of the EGR-valve (if EGR is configured)

1.16.2 Recommendations on sensor equipment

It is recommended to use **calibrated** sensors. At least the HFM and all pressure sensors shall be calibrated. The HFM sensor should be calibrated completely mounted on the air-filter module on the flow-bench (required time: about ½ day). The resulting table should be used instead of the table in the ECU.

If possible, a pulsation compensated HFM sensor should be used. If the data acquisition is capable of measuring external analogue channels, it is not necessary to use the system HFM (i.e. the airflow sensor can be of any type). If no pulsation compensated HFM is used, it is important to mount the sensor at a location where no pulsation effects occur. E.g. an oil barrel, mounted **upstream** of the air-cleaner can be used as a low pass filter (do not mount it downstream of the throttle, the barrel will be damaged). If the airflow sensor is mounted upstream of the barrel, this ensures that no pulsation effects will occur. It is recommended to mount an air-intake in front of the sensor.

1.16.3 Data acquisition system

If the data is acquired with a calibration system, it should be capable of measuring external analogue channels. In addition, it should be possible to convert those measured voltages into physical units by interpolation with tables (not only linear functions).

It should be possible to export measured data into tables either in ASCII or Excel file format.

1.16.4 Test bench

<u>Duration of tests on engine dyno:</u> 2 days

<u>Duration of tests on chassis dyno:</u> 1 week

<u>Duration of tests on climatic chamber:</u> 1-2 weeks

<u>Duration of tests on altitude bench:</u> 1 week

For the calibration of the intake manifold model at normal engine operation temperature, there are no special requirements for the test bench.

If the IMM has to be calibrated during warm-up (temperature dependency) at the engine dyno it would be helpful if the coolant (and the intake air) can be adjusted to any desired temperature (e.g. use heat exchanger in coolant circuit).

If the dyno is equipped with an ACS (Automatic Calibration System), all steady state measurements can be done very fast (and unattended) using the ACS.

1.17 Calibration Method

1.17.1 Calibration preparation

The following points have to be done or checked before beginning with the calibration:

- Pulsation compensated air-mass flow sensor is used
- HFM is mounted at a position where no pulsation takes place
- Calibration of the air-mass flow sensor was done with complete air-cleaner module and resulting conversion table is used for conversion of voltage to air-mass flow
- Adaptation of the throttle position sensor stabilized
- Intake manifold has no leakage (especially required for MAF system)
- Throttle — and if used Idle speed actuator — should be nominal for principal calibration
- Run-in phase of the engine is finished
- All sensors (especially HFM and MAP) are calibrated and signals have been checked.

For a calibration of the temperature correction of the volumetric efficiency in addition the data acquisition system has to be prepared so that it can be started directly before starting the engine. Otherwise, data at low temperatures will be lost.

1.17.2 Points to be checked during the engine run

The following points have to be checked during the engine run:

- Engine is operating at nominal temperature (about 90°C)
- System intake air has desired value (usually about 25 °C)

1.17.3 Interfering functions

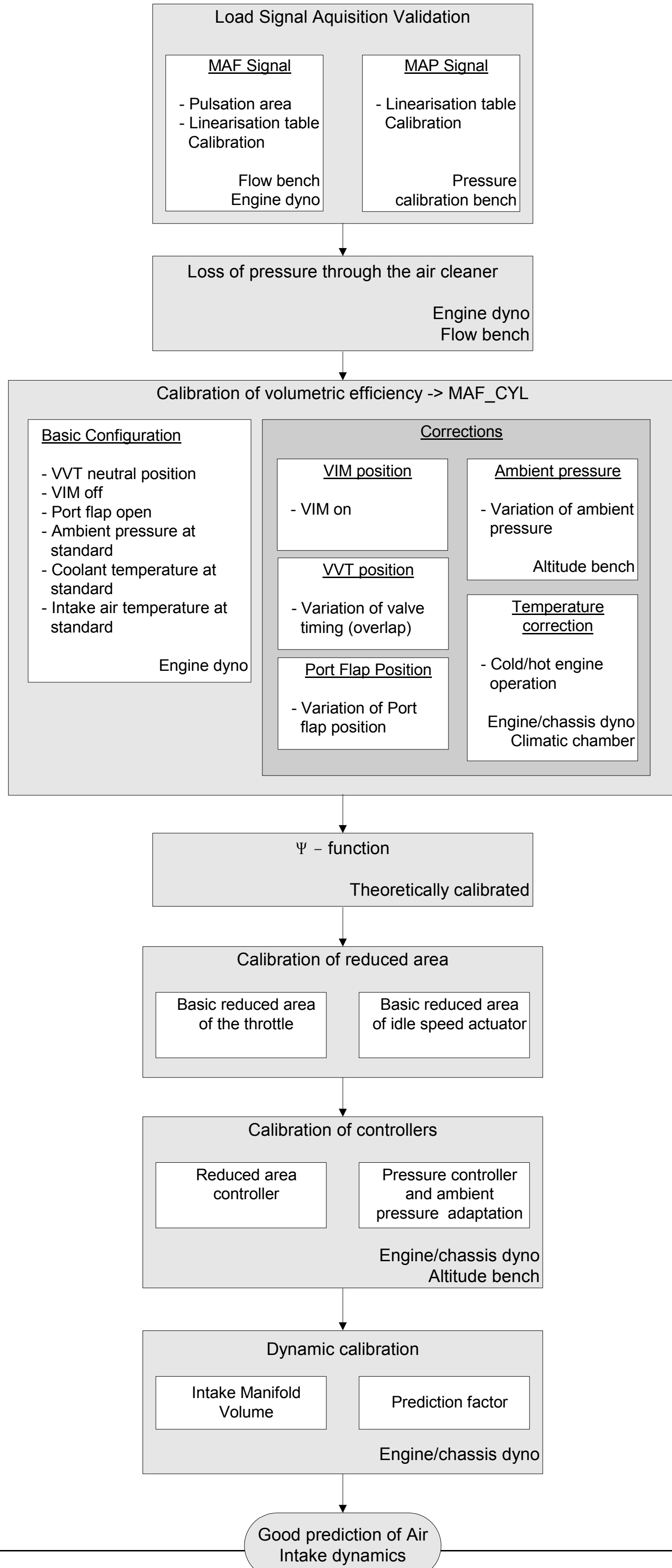
None

1.17.4 Test sequence

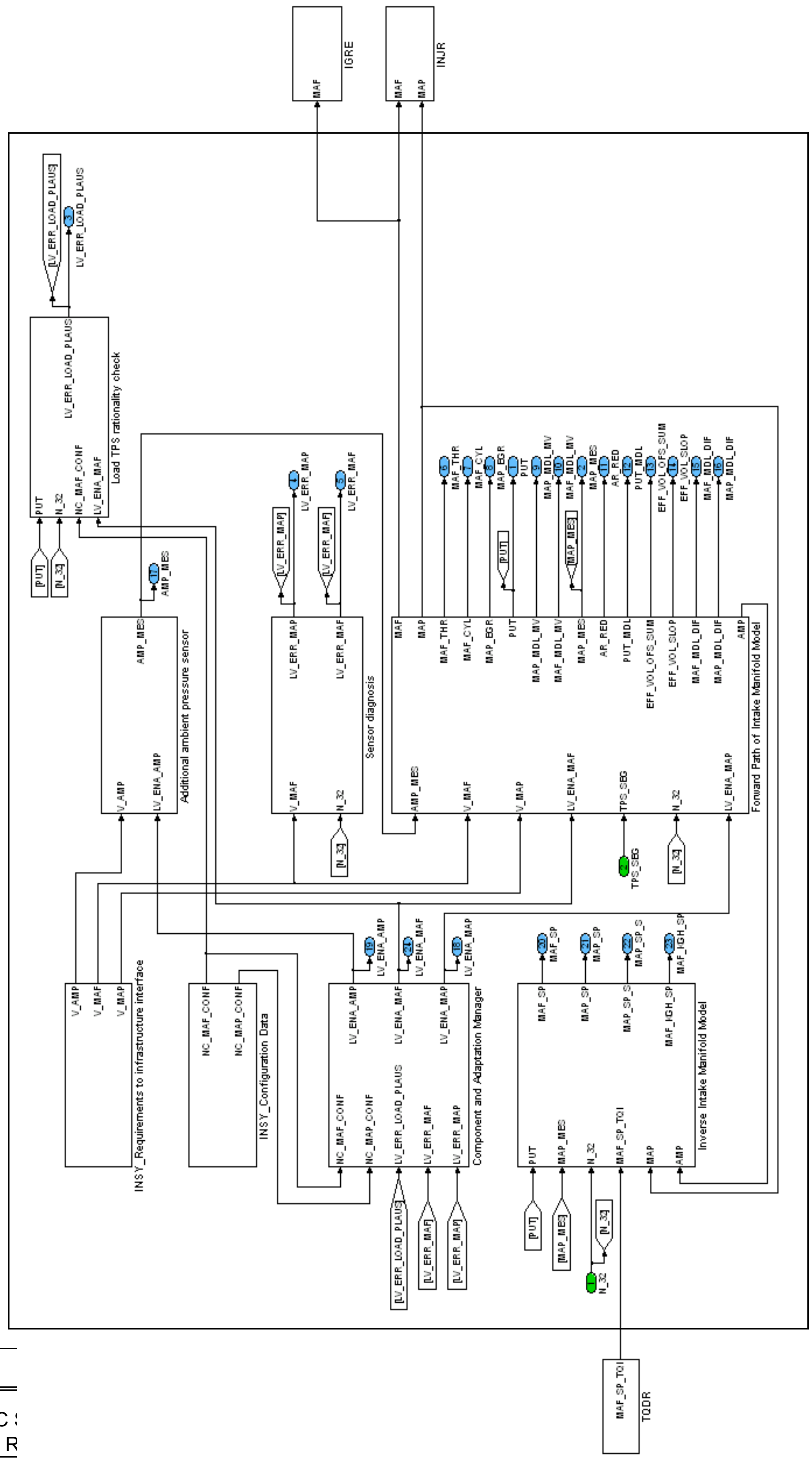
The steady state and transient calibrations are done on the engine test bench. The temperature corrections of the volumetric efficiency have to be done on a climatic chamber. Afterwards the steady state and transient calibrations have to be checked on a chassis dyno. Altitude corrections and pressure adaptation have to be calibrated either on an altitude test bench and/or on an altitude trip.

1.17.5 Calibration flow chart

general specification



1.17.6 Architecture Overview



1.18 MISF General

1.18.1 General description

The goal of the MISF is execute a misfire monitoring according legal requirements (conditions, engine area, errors & symptoms handling...) and provide to the ERRM aggregate following CARB misfire legal informations to manage the MIL and corresponding error Pcodes

- **CARB A misfire failure criterion:**

  Risk of catalyst damage, monitoring interval over 200 crankshaft during the driving cycle.

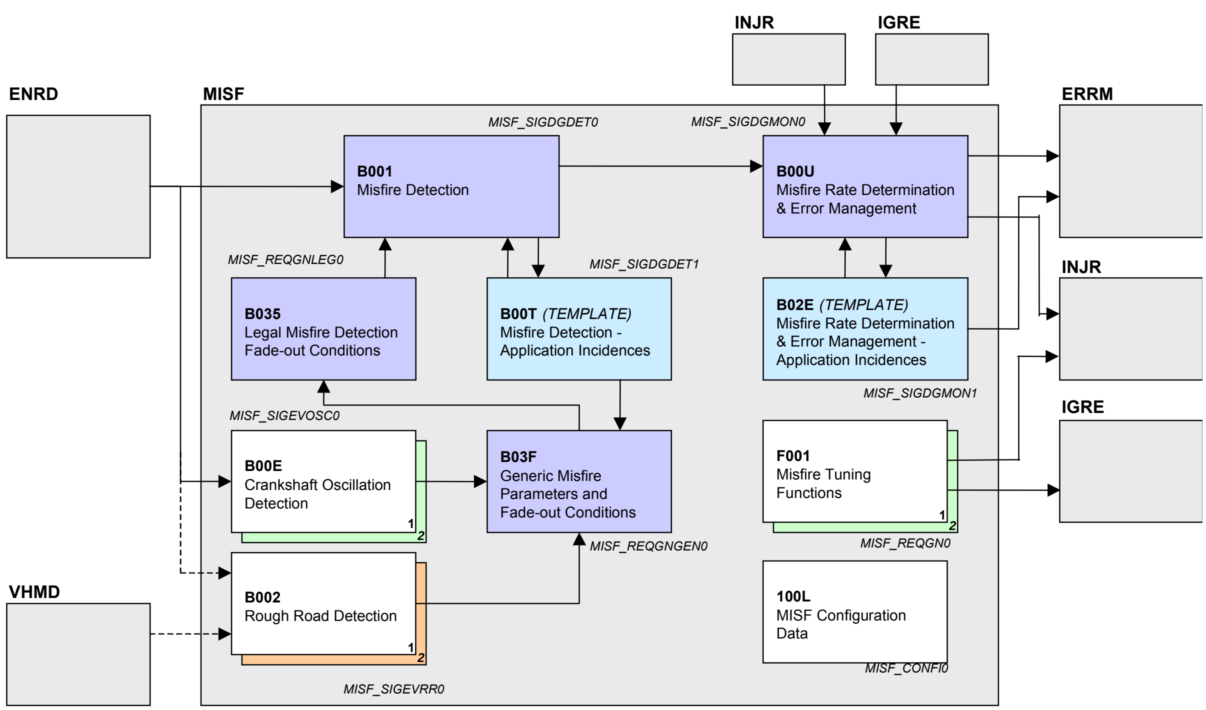
- **CARB B1 misfire failure criterion:**

  Emission increase, monitoring interval over the first 1000 crankshaft revolutions of the driving cycle.

- **CARB B4 misfire failure criterion:**

  Emission increase, monitoring interval over 1000 crankshaft revolutions. For error detection, misfire must be take place for 4 monitoring intervals (consecutive or not).

1.18.2 Architecture Overview



1.18.3 Description of the containing functions

1.18.3.1.1 B00E - Crankshaft oscillation detection

The crankshaft oscillation detection module uses engine roughness components to identify drivetrain oscillations than could cause wrong misfire detections in a defined engine operating area. If such oscillation occurs, this module triggers a fade-out.

1.18.3.1.2 B002 - Rough road detection

The rough road detection module uses informations coming from wheel speed sensor(s) or from ABS module (via harness or CAN) to identify transmission oscillations than could cause wrong misfire detections in a defined engine operating area. If such oscillation occurs, this module triggers a fade-out.

1.18.3.1.3 B03F - Generic misfire parameters & fade-out conditions

This module provides parameters needed for the misfire detection process (data delayed, data stacks, and zero load line...). It also managed all generic misfire fade-out conditions linked to the misfire detection method.

1.18.3.1.4 B035 - Legal misfire detection fade-out conditions

This module manages all legal misfire fade-out conditions defined in legal texts (US & EC).

1.18.3.1.5 B00T - Misfire detection - Application incidences

This module defines specific corrections for detection thresholds, application specific fade-out with standardised outputs to the generic modules.

1.18.3.1.6 B001 - Misfire detection

Detection core function based on engine roughness index provided by ENRD.

1.18.3.1.7 B00T - Misfire detection - Application incidences

Some functionality for calibration ease can be launched after detection.

1.18.3.1.8 B02E - Misfire rate determination & error management - Application Incidences

This module defines application specific fade-out & informations with standardised outputs to the generic misfire rate determination module. Diagnosis conditions are defined in this module.

Must be executed before B00U module.

1.18.3.1.9 B00U - Misfire rate determination & error management

This module defines the Misfire criterions according legal texts description *(MIS_A = misfire damage catalyst criterion, MIS_B1 = misfire emission criterion at engine warm-up & MIS_B4 = misfire emission criterion)* and identify the cylinders in failures

1.18.3.1.10 B02E - Misfire rate determination & error management - Application Incidences

This module defines misfire errors according the type chosen by NC_TREAT_DIAG_MIS: errors defined per misfire criterions or errors defined per cylinder.

Failures like Misfire with low fuel tank level, Multiple cylinder misfire, Random cylinder misfire are also managed in this module.

Must be executed after B00U module.

1.18.3.1.11 F001 - Misfire Tuning Functions

This module allows to generate misfire patterns (continuous or pseudo random) via injection and/or ignition shut-off interfaces.

general specification

This module is optionnal and can be integrated during validation and calibration stages. It is strongly recommended to remove this functionally on serial product software (integration choice via NC_USE_MIS_GEN compilation switch).

1.19 VVTI General

Objectives

The goal of the VVTI is to phase the opening period of the inlet and/or exhaust valves with regard to the crankschaft. In other words, it is possible to change the closing point of the inlet valves and/or the closing point of the exhaust valves.

This system provides the adjustment of a camshaft to a desired phasing ~ to a desired valve timing. The desired camshaft position is further called setpoint. The setpoint can be any position in the available adjustment range. Every project has to implement its determination of the setpoint as a function of the engine operating state. The setpoint can be adapted to the control algorithm slightly and the controller sets the camshaft to this setpoint.

Principle

The camshaft phasing is realised by means of an actuator that works with the engine oil. The actuator has two chambers that can be filled and emptied by the oil. Filling one chamber and emptying the second one leads to a change of the camshaft phasing. If the oil flow into and out of the chambers is stopped the actuator holds its position, i.e. the camshaft does not move (no additional movement to the basic rotary movement). The oil flow is controlled by means of a solenoid proportional oil control valve. The solenoid is energised with a duty cycle signal from 0 to 100 %. The piston of the valve is pressed against the solenoid by a string. If 0 % energisation is applied the valve piston is in the passive stop position. The oil flow causes that the actuator moves towards and remains in the actuator passive stop position. If 100 % energisation is applied the actuator moves towards and remains in the actuator active stop position. If an energisation is applied, in which the valve piston closes both chambers or opens both in the same level, the actuator holds its position. Such energisation is called holding energisation. It is approximately from 30 to 60 % duty cycle depending especially on the voltage and solenoid temperature.

The IVVT system is an optional part of the engine. Its implementation has many benefit impacts on the engine performance. The system allows to phase the opening period of the valves continuously in the adjustment range of the actuator. Influences of IVVT on the engine processes and how IVVT can improve the engine performance is briefly described in following paragraphs.

There are two physical effects regarding the gas flow in the engine manifold. The first one is the inertia of the gas flowing in a pipe (ram effect) and the second one is the wave effect.

The gas in the inlet pipe is accelerated during the induction stroke. The pressure in the combustion chamber rises at the end of the induction phase because the piston already moves towards the top dead centre. The kinetic energy of the flowing gas acts against the combustion chamber pressure and helps to improve the volumetric efficiency. The ram effect is very strong at high engine speeds. The inlet valve timing is to phase towards retard. The inlet valve timing is to phase towards advanced at low engine speeds in order to minimise the backflow due to the increased combustion chamber pressure.

The piston moving towards the bottom dead centre during the induction stroke creates a low pressure wave (compared with the mean inlet pressure) running away from the combustion chamber. It propagates with the velocity of sound in the pipe. This wave is reflected at the open end of the pipe (manifold collector) as a high pressure wave running towards the combustion chamber. If this high pressure wave reaches the inlet valve short before its closing, improvement of the volumetric efficiency can be obtained. The time of the wave arrival depends on the pipe length and time available, i.e., engine speed. The velocity of sound changes slightly with the air temperature.

These two effects are superposed. It is possible to increase the maximum torque at low speed and to keep the power output without increasing engine displacement by means of the infinitely variable inlet valve timing.

There are similar effects on the exhaust side. The gas flowing in a pipe tends to create vacuum if there is not enough gas supply from the combustion chamber, also "an anti-ram effect". High pressure waves are produced during the exhaust period. They are reflected as low pressure waves. It is possible to reduce the amount of the residual gases, and thus to manage more space for the fresh air. The phasing of the exhaust valve timing can contribute to improvement of the volumetric efficiency too. The high gas temperatures and high pressures, especially amplitudes, in the exhaust system make these effects very sensitive to changes of the engine load and speed.

The situation at full load has been described above. The IVVT system can improve the engine performance at part load too. The means is the internal exhaust gas recirculation. That is an intentional extension of the amount of the residual gases in the combustion chamber. Higher portion of the residual gases improves the thermodynamic efficiency of the cycle because of changed properties of the working fluids. Further, it is necessary to increase the inlet pressure in order to have the same power output (~ the same volumetric efficiency) at higher portion of the residual gases. This decreases the pumping losses in part load. It leads to the reduction of the fuel consumption. Higher portion of the residual gases dilutes the air-fuel mixture. The maximum combustion temperature is lowered and this reduces the production of oxides of nitrogen.

Higher portions of the residual gases are obtained by moving inlet towards advanced and exhaust towards retard. The exhaust gases flows into the inlet port at the opening of the inlet valve because the combustion chamber pressure is higher than the inlet system one. This leads to a decrease of the combustion chamber pressure, and thus, there can be a backflow of the exhaust gases from the exhaust port. The exhaust gases in the inlet port are then inducted into the combustion chamber during the main induction period.

The extension of the amount of the residual gases is limited. The ignition properties of the mixture get worse. It leads to higher engine roughness. The driver must not feel it especially in idle. The combustion duration becomes longer which acts against the fuel efficiency improvement. It is obvious that the ignition angle has to be fitted. On the other hand the IVVT system allows to decrease the engine roughness in idle in comparison with an engine with fixed timing. The fixed timing is a compromise for the maximum power output, maximum torque and idle stability. The variable timing can have the minimum valve overlap smaller, and thus less residual gases in idle and therefore better idle stability.

There are further effects of the IVVT system on the engine performance. It is possible to reduce the fuel film creation on the inlet port walls by means of the backflow of the hot gases from the combustion chamber. It can reduce the production of unburned hydrocarbons during the warm-up phase. The catalyst light-off steps and the catalyst overheating protection can be supported by IVVT. The variable valve timing influences the flow field within the combustion chamber which is a part of the combustion process in the gasoline direct injection engines.

Description

The IVVT system can be in active or inactive state. The inactive one is characterised by permanent low level energisation (~0-15 % PWM) of the solenoid valve. The active state means that the camshaft position is controlled to a desired setpoint. It is the normal prevailing state. The system is inactive when the engine is stopped. The following steps are done before reaching the active state: check min/max oil temperature and min/max battery voltage (continuously done), wait for enough oil pressure after the engine start, wait for the end of the reference position adaptation, active state is reached. If there is a failure the IVVT system is inactivated. In some cases the system remains partly active (some camshafts are active, some are inactive) in order to enable bank balancing (see bellow).

The reference position adaptation is done by the ENSD aggregate when enabled by the IVVT function. The aim of the adaptation is to eliminate inaccuracies of the camshaft position acquisition. The primary camshaft position from ENSD in its coordinate system is recalculated into an IVVT camshaft position. This position is related to the gas-exchange top dead centre.

The setpoint is calculated dependent on the engine operating state (combustion mode, load, speed, coolant temperature, etc.). The inlet setpoint can be prioritised to the exhaust one at low oil pressure. It means that the exhaust setpoint is kept untill inlet reaches its desired position and then the map exhaust setpoint is handed over.

The setpoint is worked up in order to facilitate the position control. The main operations are: limitation to the possible adjustment range and setpoint close to the reference position equals the reference position. If the setpoint is in the reference position for a longer time the reference position adaptation is enabled.

The camshaft position control is based on a superposition of the holding energisation and time limited low or high level energisations. Generally, the holding energisation is applied. The controller calculates an energisation duration and its level dependent on the setpoint course and the camshaft deviation from the setpoint. The holding energisation is replaced by the calculated level for the calculated time. This is realised at camshaft edges where the camshaft position is acquired.

The controller consists of a Pre-control and P-control. The Pre-control manages part of the setpoint change and reacts to setpoint changes quickly. The delay and the adjustment speed dependent on the engine speed and oil temperature are the Pre-control parameters. The P-control manages the residual part of the setpoint change and disturbances.

The primary holding duty cycle is a feed-forward control (map dependent on the battery voltage and modelled solenoid temperature). A correction is usually needed for every individual engine due to tolerances. This is realised by means of two adaptations (one of them is a kind of I-control) which evaluate the camshaft deviation and drift at steady setpoint conditions.

There is a balancing of the camshaft positions in the individual banks of a 2-bank system. Unequal adjustment speeds or a failure in one bank can cause different positions in the individual banks. An interbank controller calculates individual setpoints so that the differences between the banks are minimal.

There are some diagnoses in the IVVT system. The electrical failures "short circuit to plus", "short circuit to ground" and "open circuit" can be detected. A chain/belt jump is monitored. There is a diagnosis for the detection of a persistent camshaft deviation and/or slow responsing actuator. The system can be tested by means of a short trip test (special setpoint course) at EOL test or as a garage test.

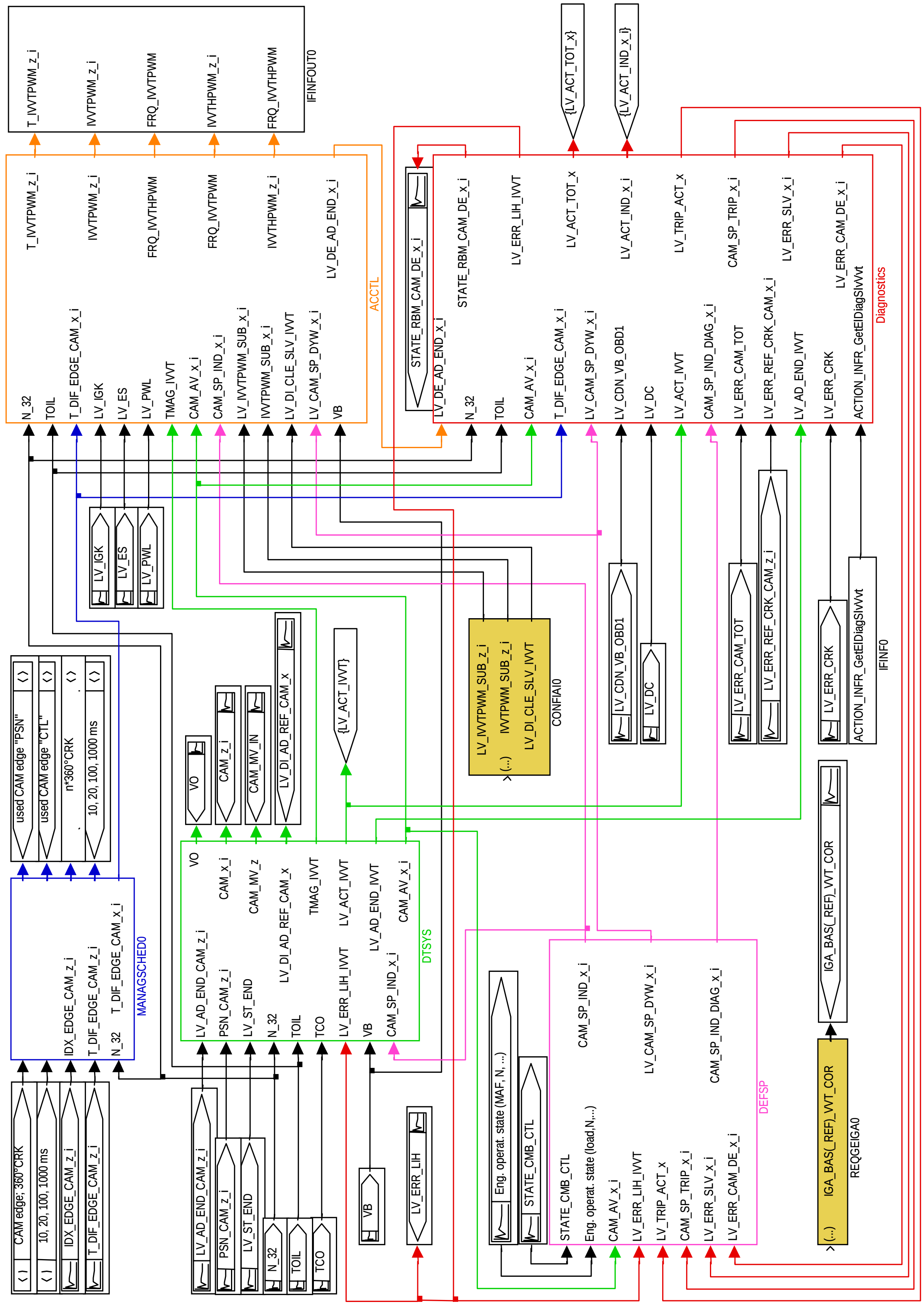
The IVVT system influences the gas-exchange process, especially the amount of the residual gases. The ignition angle has to be corrected. Ignition angle corrections (one for the basic and one for the reference ignition angle) are outputs from the VVTI aggregate.

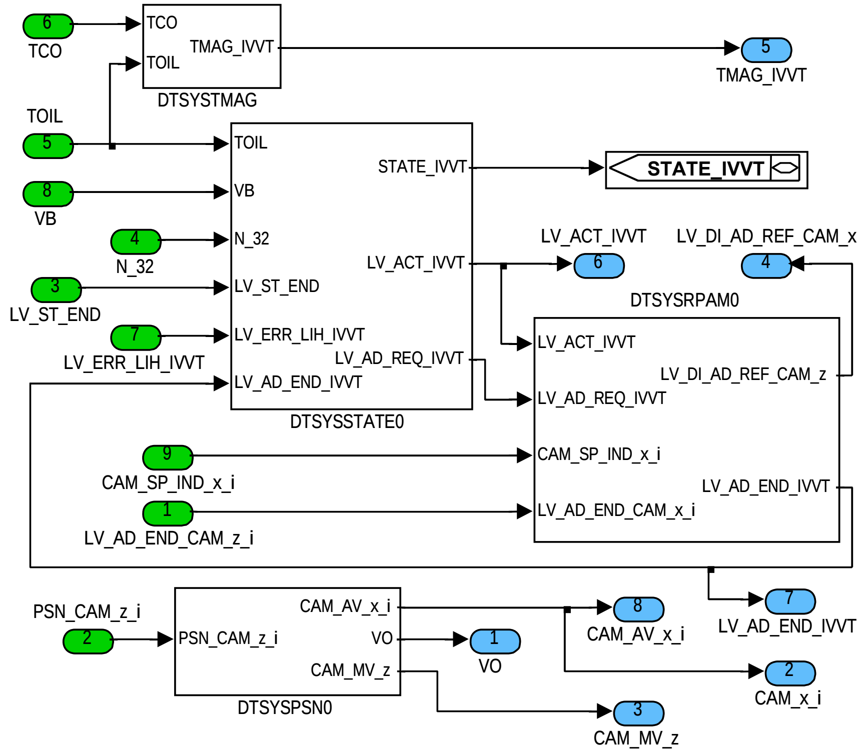
There is an important question how the engine reacts on an IVVT failure. It is possible to test it by means of a special algorithm. The operating point dependent setpoint is used in the diagnoses. The controller works with a special setpoint, e.g. a constant for a blocked actuator. The engine behavior, especially the emission production, can be observed.

A multi-segment camshaft target wheel can be used. It is not necessary to use all edges for the camshaft position control especially at high engine speeds. There can be harmful effects on the controll if the time between edges is shorter than the system delay. Additionally, there is a huge runtime demand if many edges are used. It is possible to strech the setpoint calcualtion to every n-th engine revolution (dependent on the engine speed) for runtime saving.

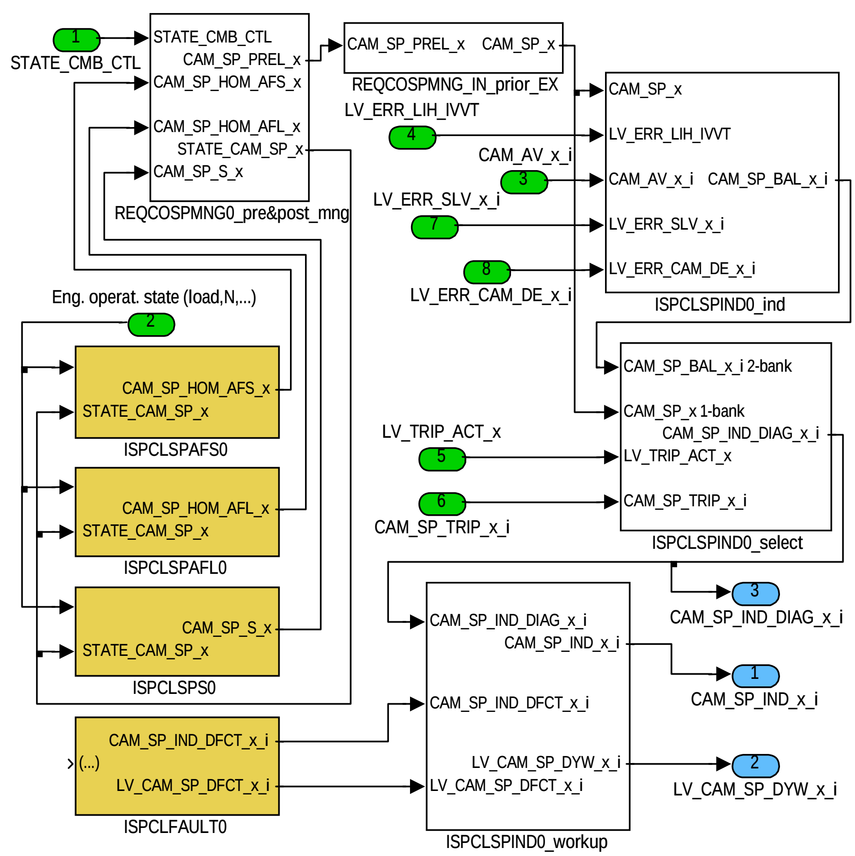
Flowchart

For the sake of clearness the basic flowchart (architecture) of the VVTI aggregate presented here is simplified. Only the main signals are shown. Further, names of some quantities were shortened. Following substitions were done: "x" = "IVVT_IN(_EX)" and "z" = "IN(_EX)".

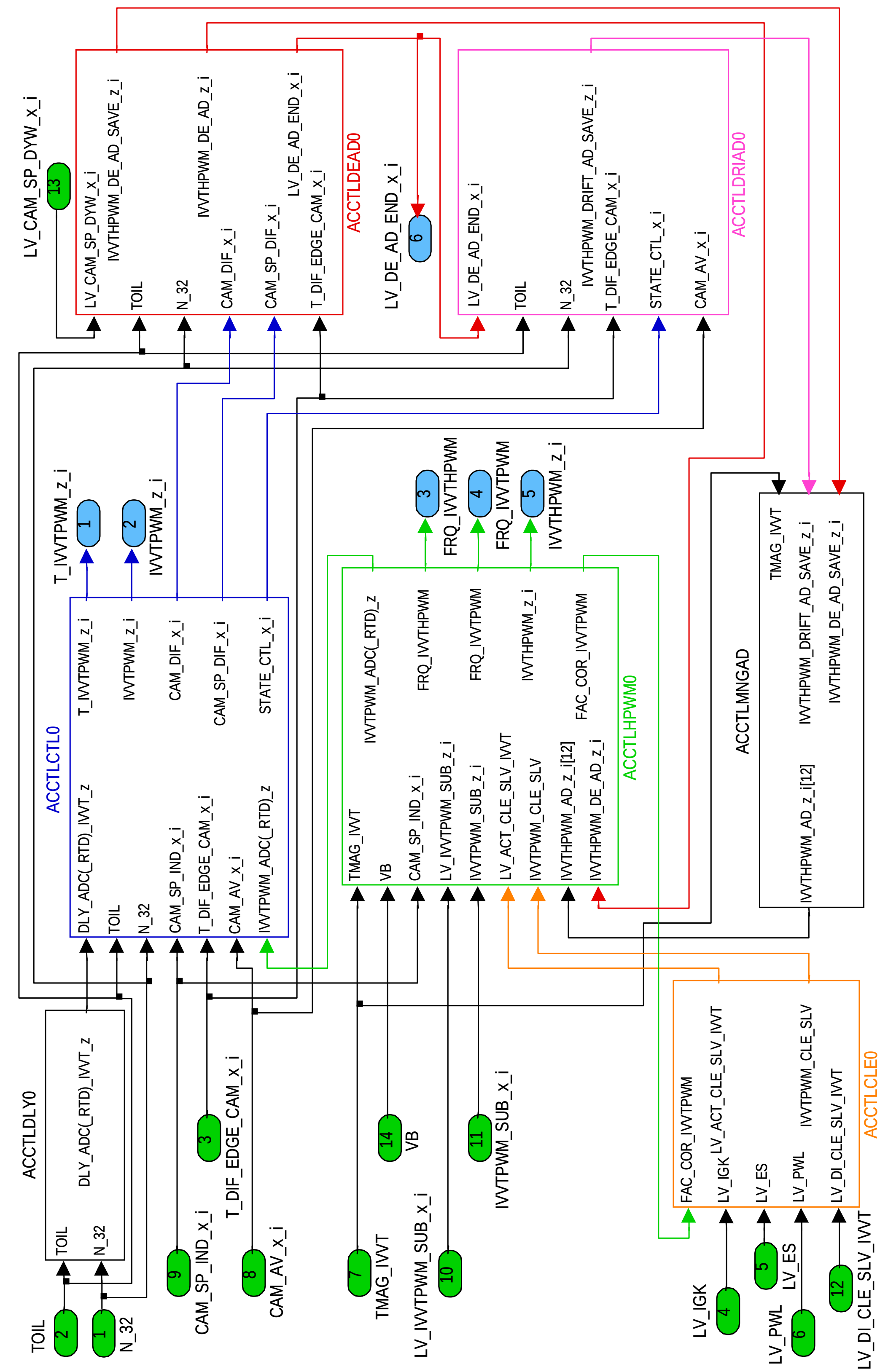

IFINFOUT0
ACCTL
Diagnostics
CONFIAI0
MANAGSCHED0
DTSYS
DEFSP
IFINF0
REQGEIGA0
{LV_ACT_TOT_x}
{LV_ACT_IND_x_i}
{LV_ACT_IVVT}
used CAM edge "PSN"
used CAM edge "CTL"
n*360°CRK
10, 20, 100, 1000 ms
CAM edge; 360°CRK
Eng. operat. state (MAF, N, ...)
Eng. operat. state (load,N,...)
IGA_BAS(_REF)_VVT_COR



Subsystem "DTSYS".



Subsystem "DEFSP".



Subsystem "ACCTL"

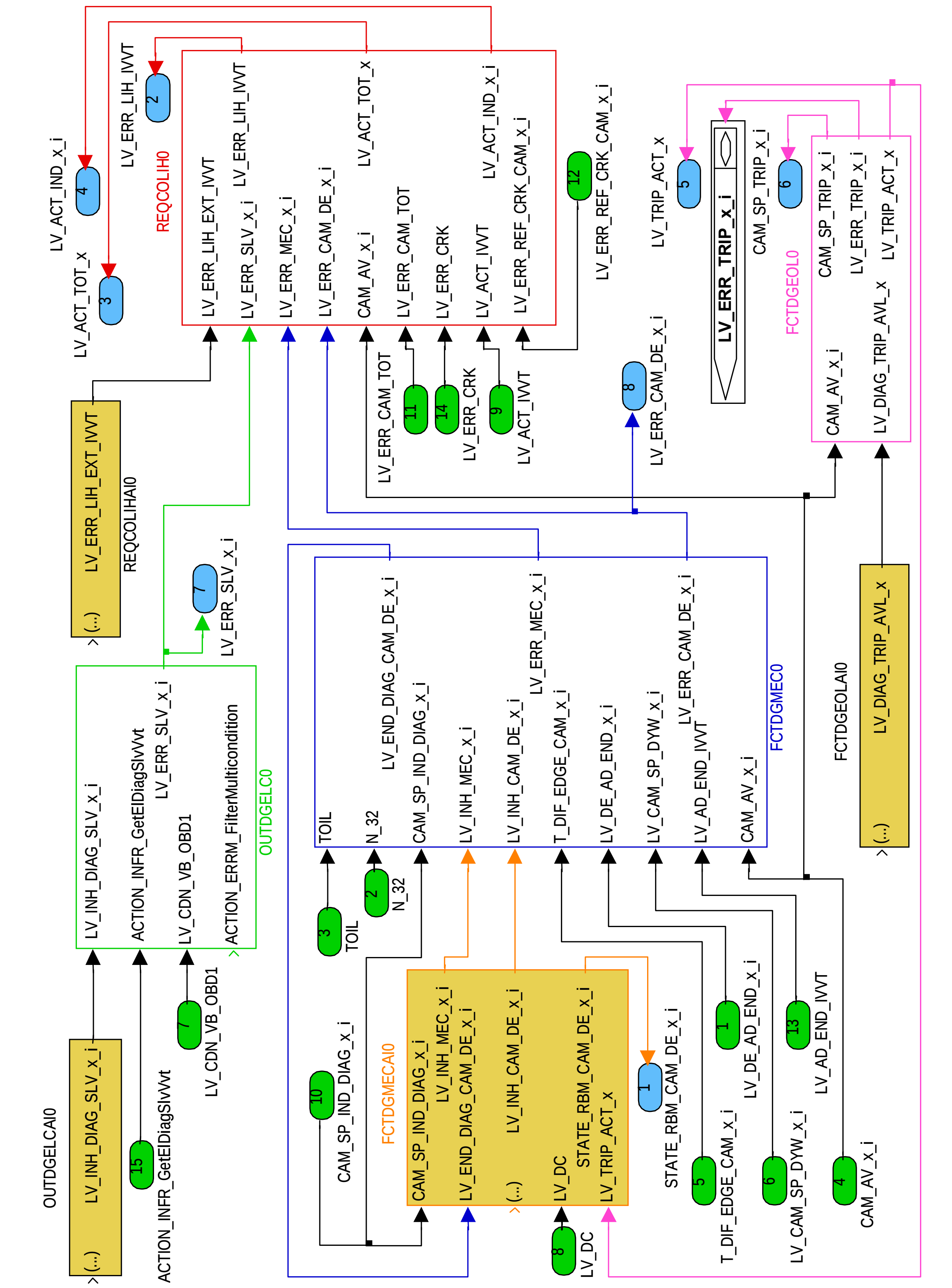
Chapter	Baseline	Include File
General	691F00	30101U02.00A

		Date	Department	Sign
Designed by	GC Shin	2008-05-27	SV P GS ES	
Released by	G. Raab	2008-05-27	SV P GS Sys2 PL	

Continental

Designation
Engine Management System HMC Theta II ETC/BIN

Document Key
E150-024.49.01 SPE 000 20.0



Subsystem "Diagnostics"

Aggregate functions

This aggregate version consists of the following functions:

- VVT State - Valve timing data are defined here. Camshaft position is calculated. IVVT system state is determined.
- VVT SP (setpoint) - Calculation of camshaft position setpoint as a function of engine state.
- VVT Control - Camshaft position control including feed-forward control, feed-back control and adaptations.
- VVT Interface BSW - Realisation of control signals. Ouput signal diagnosis information is acquired.
- VVT LIH (limp home) manager - Manages IVVT system state in case of a failure.
- VVT OBDI - Output signal diagnosis.
- VVT OBDII - Crankshaft to camshaft mechanics violation diagnosis (chain/belt jump). Camshaft position deviation diagnosis (actuator jammed, actuator slow responsing). Rate based monitoring interface.
- VVT EOL test - IVVT short trip (prescribed setpoint course is evaluated).
- VVT Ignition corr (correction) - Ignition angle correction as a function of a camshaft position not in reference position.

1.20 AIRT Configuration Data:

Input data:

NC_TIA_CONF	NC_TAM_CAN_USE	NC_TAM_CLC_BOL	NC_TAM_CLC_TOL

1.20.1 Local configuration data:

Here are listed the configuration data, which are internally (and only internally) used in AIRT.

Data	Value
NC_TAM_CLC_BOL (non calibreatable value)	-35°C (typical value for lower limit of TAM)
NC_TAM_CLC_TOL (non calibreatable value)	50°C (typical value for upper limit of TAM)
NC_DIAG_INTM_DYN_USE	1

1.20.2 Global configuration data:

Here are listed the configuration data, which can be used by other Aggregates.

Data	Value
NC_TIA_CONF (compile switch)	10 (one TIA Sensor in Intake Manifold)
NC_TAM_CAN_USE (compile switch)	1 (ambient Temperature via CAN available)

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_DIAG_INTM_DYN_USE	1	0H...1H	0...1	1	-
Choice to use the Intermittent and Stuck Signal Diagnosis (=1) else (=0)					

1.21 VVTI Configuration Data

Input data:

NC_NR_CBK_IVVT	NLC_IVVT_IN	NLC_IVVT_EX	NC_CAM_MDL_BUF_IVVT

General information:

The following describes the general rules for determination of the configuration data and their values.

1.21.1 Global configuration data

Here are listed the configuration data, which can be used in other aggregates :

Data	Value	Description
NC_NR_CBK_IVVT	1	Number of camshafts cylinder banks
NLC_IVVT_IN	1	inlet phasing present
NLC_IVVT_EX	1	exhaust: no phasing

1.21.2 Local Configuration Data

used only in the VVTI aggregate :

Data	Value	Description
NC_CAM_MDL_BUF_IVVT	50	buffer size for modelling camshaft position

1.22 EGPR Configuration Data

General information:

The constant NC_xxx_CONF describes whether a system is equipped with additional functionality or not.

The following describes the general rules for determination of the configuration data:

1.22.1 Global configuration data

Here are listed the configuration data, which can be used in other aggregates :

Data	Value
NC_PRS_EX_MES_CONF	0

1.22.2 Local configuration data

Here are listed the configuration data, which are used only in the EGPR aggregate.

Data	Value
NC_EF_CONF	0

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_PRS_EX_MES_CONF	1	0H...01H	0...1	1	-
System configuration flag (compiler) if an exhaust gas presure iis available (= 1) or not (= 0)					
NC_EF_CONF	1	0...1H	0...1	1	-
System configuration (compiler) flag to indicate if a system with exhaust flap is available (= 1) or not (= 0)					

1.23 EGRC configuration data

General information:

The constant NC_EGR_CONF describes whether a system is equipped with external Exhaust Gas Recirculation or not. NC_EGR_CONF is set to:

- 0, if external EGR is not available
- 1, if external EGR is available

Formula section:

NC_EGR_CONF = 0

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_EGR_CONF	1	0...1H	0...1	1	-
System configuration (compiler) flag to indicate whether external EGR is available (= 1) or not (= 0)					

1.24 FMSP Configuration Data

The following document describes the general rules for definition of the configuration data for FMSP Aggregate.

Input data:

NC_IN_REF	NC_LAMB_REF

General information:

The following document describes the general rules for definition of the configuration data for FMSP Aggregate

1.24.1 Global configuration data

Here are listed the configuration data, which are not only used in the modules of the FMSP aggregate.

Data	Value
NC_DUI_CONF	**0**: single injection
NC_USE_FQ	**1**: FQ is used
NC_WF_MFF_BAS	**0** : TI-based wallfilm compensation
NC_IN_REF	**00000000** (Pattern for allocation of physical cylinders to exhaust bank)
NC_LAMB_REF	**00000000** (Pattern for allocation of physical cylinders to intake bank)

1.24.2 Local configuration data

Here are listed the configuration data, which are used only in the FMSP aggregate.

Data	Value

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_DUI_CONF	1	0H 1H	single ínjection double ínjection	1	-
System flag to distinguish between Single- and Double Injection					
NC_USE_FQ	1	0H 1H	FQ is not used FQ is used	1	-
Configuration switch to indicate the usage of the generic fuel quality adaptation					
NC_WF_MFF_BAS	1	0H 1H	TI-based wallfilm compensation MFF-based wallfilm compensation	1	-
Configuration switch to indicate the usage of MFF- or TI based wallfilm compensation					

1.25 INJR Configuration Data

The following document describes the general rules for definition of the configuration data for INJR Aggregate.

Input data:

NC_CYL_NR			

General information:

The following document describes the general rules for definition of the configuration data for INJR Aggregate

1.25.1 Global configuration data

Here are listed the configuration data, which are not only used in the modules of the INJR aggregate.

Data	Value
NC_INJ_CONF	0: MPI
NC_NR_SYM_IV	Constant, which defines the available symptoms for injection valve diagnostic. ATIC 39: (SYM_SCB + SYM_SCG +SYM_OC) =7H

1.25.2 Local configuration data

Here are listed the configuration data, which are used only in the INJR aggregate.

Data	Value
NC_EOI_LIM	**End of phasing range relatet to NC_REF_EOI_LIM 540 °CRK (MPI-Engine)** **NC_REF_EOI_LIM = -360°CRK, defined in I/O SW, assigned to REF_EOI_LIM)** **(Refers to 30706C01 and 02)**
NC_INJ_INH_SWI_IV_SHIFT_NR	Constant which defines how many bits are used for the shut off sequence. **8 (Cf. 7018 Module)**
NC_USE_TI_EXT_ADJ	0

Data	Value
#if (NC_INJ_CONF = 0)	
NC_ECU_SAMPLE_FAST	**1ms (Specified fastest task recurrency)**
NC_EOI_CST_CYC_ST_ENA	Compiler switch to decide start strategy: cycle counter dependent : NC_EOI_CST_CYC_ST_ENA = 0 (Refers to 30706C02)
#elseif (NC_INJ_CONF = 1)	
NC_VBOOST_STAGE	Boost divisor bridge ratio for ATIC 21 HPDI Driver (Refers to 30402G01)
#endif	

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_INJ_CONF	1	0H 1H	MPI HPDI	1	-
Injection Mode					
NC_ECU_SAMPLE_FAST	1	-	1 ... 10	-	ms
Specified fastest task recurrency, recommendation 1ms					
NC_INJ_INH_SWI_IV_SHIFT_NR	1	3 ... 10 H	3 ... 16	1	-
Constant which defines how many bits are used for the shut off sequence					
NC_EOI_LIM	1	0 ... 780H	0 ... 720	0.375	°CRK
End of phasing range relatet to NC_REF_EOI_LIM					
NC_NR_SYM_IV	1	1..FH	1..15	1	-
Available symptoms for injection valve diagnostic					
NC_USET_ADJ	1	0 ... 1 H	0 ... 1	1	-
Compiler switch to enable external TI adjustment caused by service tool					
#if (NC_INJ_CONF = 0)					
NC_EOI_CST_CYC_ST_ENA	1	0 ... 1 H	0 ... 1	1	-
Compiler switch to decide start strategy: cycle counter dependent (1) or TCO dependent start (0)					
#elseif (NC_INJ_CONF = 1)					
NC_VBOOST_STAGE	1	0...1000H	0..25.6	0.00625	-
Boost divisor bridge ratio for ATIC 21 HPDI Driver					
#endif					

1.26 IGSP Configuration Data

Input data:

NC_INJ_CONF	NC_N_IGA_UPD_RATE	NC_CBK_EX_NR	NLC_USE_IGA_DUI_S
NC_IGSP_VERS			

General information:

The following document describes the general rules for definition of the configuration data for Ignition Angle Calculation Aggregate

1.26.1 Global configuration data

Data	Value
NC_INJ_CONF	MPI
NC_CBK_EX_NR	1

NC_INJ_CONF is used to integrate in the IGSP aggregate calculation and variables related to Stratified DI engines

NC_CBK_EX_NR is used to calculate the Mean Values of Ignition for determination of Ignition Efficiencies for Exhaust treatment

1.26.2 Local configuration data

Here are listed the configuration data, which are used only in the IGSP aggregate.

Data	Value
NC_IGSP_VERS	02
NC_N_IGA_UPD_RATE	8160rpm
NLC_USE_IGA_DUI_S	0

1.26.2.1 Cylinder bank definition for Ignition Angle definition

Due to the special needs of an engine with multiple cylinder banks each project have to defined the cylinders related to one cylinder bank for Ignition Angle calculation.

Normally, the cylinder bank definition for Ignition Angle corresponds to the one for Injection calculation.

_i represents the cylinder bank, defined in the table here after that will be used for Ignition Angle Calculation
_x designated the current cylinder

Firing order	1	3	4	2
Cylinder number _x	0	1	2	3
Cylinder bank _i (*IGA_BAS_i, …*)	1	1	1	1

1.27 IGRE Configuration Data

Input data:

NC_IGN_DIAG_TYP	NC_TD_LIM_MIN	NC_TD_LIM_MAX	NC_MPL_T_MAX
NC_MAX_IGN_MPL_NR	NC_IGC_DLY	NC_MPL_IGN_CRK_MAX	NC_IGC_CONF

General information :

The following describes the general rules for determination of the configuration data

1.27.1 Local configuration data

Here are listed the configuration data, which are used only in the IGRE aggregate.

Data	Value
NC_IGN_DIAG_TYP	ATM46
NC_TD_LIM_MIN	20 us
NC_TD_LIM_MAX	22.5 ms
NC_MAX_IGN_MPL_NR	7
NC_IGC_DLY	16
NC_MPL_T_MAX	60 ms
NC_MPL_IGN_CRK_MAX	12 °CRK)
NC_IGC_CONF	Full static

1.28 MISF Configuration Data

Input data:

NLC_USE_ER_STND_MIS	NLC_TREAT_DIAG_MIS	NLC_USE_MIS_GEN	SEG_NR_ER
NLC_USE_CRK_OSC_MIS	NC_SIZE_THD_ER_BUF	NC_CBK_EX_NR_MISF[NC_CYL_NR]	NC_CYL_NR
NLC_CONF_ER_MV	NLC_ENA_SCDN_NEW		

General information :

The following describes the general rules for determination of the configuration data

1.28.1 Global configuration data

Here are listed the configuration data, which can be used in other aggregates :

Data	Value
NLC_TREAT_DIAG_MIS	1
NLC_USE_CRK_OSC_MIS	1

1.28.2 Local configuration data

Here are listed the configuration data, which are used only in the MISF aggregate.

Data	Value
NC_MISF_VERS	1
NLC_USE_ER_STND_MIS	0
NC_SIZE_THD_ER_BUF	5
NC_CONF_RR_MIS	OTHER (Project Specific : 2KB00201,xxx)
NLC_USE_MIS_GEN	1
NC_CBK_EX_NR_MISF[NC_CYL_NR]	*see table below*
NLC_ENA_SCDN_NEW	1
NLC_CONF_ER_MV	0

Exhaust Configuration:

All cylinders are located on exhaust cylinder bank 0

SEG_NR_ER	0	1	2	3
NC_CBK_EX_NR_MISF	0	0	0	0

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_MISF_VERS	1	00H...FFH	0...255	1	-
MISF aggregate version					
NC_CONF_RR_MIS	1	0...FFH	0...255	1	-
Configuration of rough road detection module (None, WSS via Harness, CAN, Other)					

1.29 ENRD Configuration Data

Input data:

NC_CYL_NR	NC_SEG_DLY_ER_MES	NLC_CONF_GAIN_ADD_ER	NC_SIZE_SEG_T_COR_BUF

General information :

The following describes the general rules for determination of the configuration data

1.29.1 Local configuration data

Here are listed the configuration data, which are used only in the ENRD aggregate.

Data	**Value**
NC_ENRD_VERS	01
NC_SIZE_SEG_T_COR_BUF	9
NC_SEG_DLY_ER_MES	1
NLC_CONF_GAIN_ADD_ER	**1**

Default configurations:

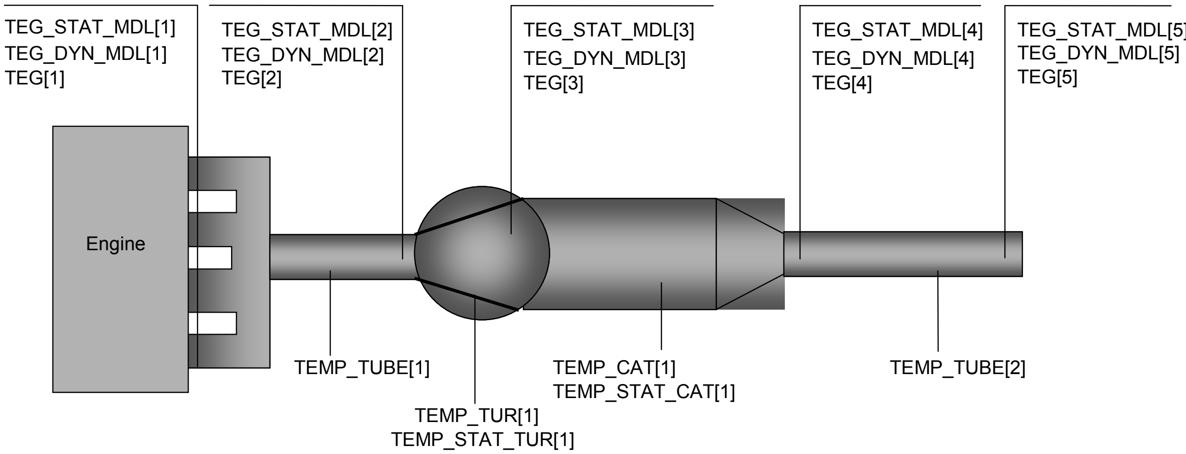
NC_CYL_NR	NC_SIZE_SEG_T_COR_BUF
4	9

Configuration data:

Name	**Dim**	**Hex. limits**	**Phys. limits**	**Resol.**	**Unit**
NC_ENRD_VERS	-	0...FFH	0...255	1	[-]
ENRD aggregate version					

1.30 EXTD Configuration data

Configuration data of EXTD.



1.30.1 Local configuration data

Here are listed the configuration data, which are used only in the EXTD aggregate.

Data	Value
NC_TEG_SENS_TYP	Without Sensor
NC_NR_TEG_SENS	1
NC_NR_TEG_MDL	5
NC_NR_ENG_OUT_MDL	1
NC_NR_TUBE_MDL	2
NC_NR_CAT_MDL	1
NC_NR_TUR_MDL	1
NC_TEG_MDL_TYP_CONF[NC_NR_TEG_MDL]	[0 1 3 2 1]
NC_TEG_MDL_INP_CONF[NC_NR_TEG_MDL]	[1 1 2 3 4]
NC_TEG_MDL_CBK_CONF[NC_NR_TEG_MDL]	[1 1 1 1 1]
NC_TEG_SENS_CONF[NC_NR_TEG_MDL]	[0 0 0 0 0]
NC_TEG_MDL_T_SAMPLE_CONF[NC_NR_TEG_MDL]	[20ms 20ms 100ms 200ms 200ms]
NC_TEG_MDL_SAMPLE_OFS_CONF[NC_NR_TEG_MDL]	[1 1 2 3 4]
NC_TEG_MDL_INST_CONF[NC_NR_TEG_MDL]	[1 1 1 1 2]

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_ENG_OUT_MDL	1	1...4H	1...4	1	[-]
number of engine out exhaust gas temperature model instances					
NC_NR_TUBE_MDL	1	1...6H	1...6	1	[-]
number of tube exhaust gas temperature model instances					
NC_NR_TUR_MDL	1	1...2H	1...2	1	[-]
number of turbine exhaust gas temperature model instances					
NC_NR_TEG_MDL	1	1...10H	1...16	1	[-]
number of overall exhaust gas temperature model instances					
NC_NR_TEG_SENS	1	1...6H	1...6	1	[-]
number of exhaust gas temperature sensor in system					
NC_TEG_MDL_CBK_CONF[NC_NR_TEG_MDL]	1	0...3H	1...4	1	[-]
determination of cylinder bank of each component model					
NC_TEG_MDL_INP_CONF[NC_NR_TEG_MDL]	1	0...FH	1...16	1	[-]
configuration of the inputs used in this model instance					
NC_TEG_MDL_TYP_CONF[NC_NR_TEG_MDL]	1	0...3H	0...3	1	[-]
determination of model typ for each mode instance					
NC_TEG_MDL_INST_CONF[NC_NR_TEG_MDL]	1	0...5H	1...6	1	[-]
configuration of the instance of the specified component model					
NC_TEG_MDL_T_SAMPLE_CONF[NC_NR_TEG_MDL]	1	1...64H	10...1000	10	[ms]
configuration of the sample time of the model instance					
NC_TEG_MDL_SAMPLE_OFS_CONF[NC_NR_TEG_MDL]	1	0...9H	1...10	1	[-]
configuration of the calculation delay for each model instance					
NC_TEG_SENS_CONF[NC_NR_TEG_MDL]	1	0...6H	0...6	1	[-]
configuration of sensor position					
NC_TEG_SENS_TYP	1	0H 1H 2H 3H	WITHOUT_SENSOR ADC PWM ADC_AND_PWM	1	[-]
Supported TEG-sensor types					
NC_NR_CAT_MDL	1	1...4H	1...4	1	[-]
number of catalyst exhaust gas temperature model instances					

1.31 EXTC Configuration Data

General information:

This module describes the configuration data that is defined in the aggregate EXTC and its setting (not calibratable) for this software release.

Description:

With NC_NR_COP_CTL it is possible to choose 1 or 2 COP-controller. In the module Overheating prevention (Appl. Inc.) it is choosen which temperature shell be used for activation (via TEMP_COP_ACT) and which for controlling (via TEMP_COP_CTL) of the temperature controlller for catalyst overheating prevention. For controlling static temperatures and for activation dynamic temperatures are used.

The catalyst overheating prevention is independent from aggregate versions.

Local configuration data:

Data	Value
NC_NR_COP_CTL	2 – two COP-controller, 1. controls exhaust temperature before catalyst, 2. controls temperature in catalyst

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_COP_CTL	1	1...2H	1...2	1	[-]
Number of temperature-controllers for catalyst overheating prevention					

1.32 ENSD Configuration Data

Input data:

NC_CYL_NR	NC_NR_EDGE_CAM_IN	NC_NR_EDGE_CAM_EX	NC_N_MAX
NC_N_MIN	NLC_CAM_IN	NLC_CAM_EX	NC_NR_CAM_CBK
NC_OFS_TDC0_REF_CRK	NC_PSN_SEG_TDC_REF	NC_PRI_LIH_CAM_IN	NC_PRI_LIH_CAM_CBK
NC_T_SEG_MAX_CAM_IN	NC_T_SEG_MAX_CAM_EX	NC_T_SEG_MIN_CAM_IN	NC_T_SEG_MIN_CAM_EX
NLC_LIH_CAM_IN	NLC_LIH_CAM_EX	NC_PRI_SYN_CAM_IN	NC_PRI_SYN_CAM_CBK
NC_PSN_EDGE_z_CAM_IN_i	NC_PSN_EDGE_z_CAM_EX_i	NC_NR_TOOTH_TOL_MISS	NC_NR_TOOTH_TOL_ADD
NC_ACT_CRK_EDGE	NC_NR_TOOTH	NC_NR_TOOTH_GAP	NC_NR_VLD_TOOTH
NC_PHA_SEG_ER_ENSD	NC_NR_TOOTH_STALL	NC_CRK_WIN_SEG_LEN	

General information :

The following describes the general rules for determination of the configuration data

1.32.1 Global configuration data

Here are listed the configuration data, which can be used in other aggregates :

Data	Value
NC_CYL_NR	4
NC_N_MIN	22
NC_N_MAX	8160
NLC_CAM_IN	1
NLC_CAM_EX	1
NC_NR_EDGE_CAM_IN	2
NC_NR_EDGE_CAM_EX	2
NC_NR_CAM_CBK	1

1.32.2 Local configuration data

Here are listed the configuration data, which are used only in the ENSD aggregate.

Data	Value
NLC_LIH_CAM_IN	1
NLC_LIH_CAM_EX	1
NC_PRI_LIH_CAM_IN	1
NC_PRI_LIH_CAM_CBK	1
NC_PRI_SYN_CAM_IN	1
NC_PRI_SYN_CAM_CBK	1
NC_T_SEG_MIN_CAM_IN	7,35*10-3 sec
NC_T_SEG_MIN_CAM_EX	7,35*10-3 sec
NC_T_SEG_MAX_CAM_IN	1,9965 sec
NC_T_SEG_MAX_CAM_EX	1,9965 sec

general specification

Data	Value
NC_CRK_SENS_TYP	0
NC_CAM_SENS_ TYP	2
NC_CAM_WHEEL_ TYP	0
NC_ACT_CRK_EDGE	0
NC_NR_TOOTH	60
NC_NR_TOOTH_GAP	2
NC_NR_TOOTH_TOL_ADD	2
NC_NR_TOOTH_TOL_MISS	2
NC_NR_VLD_TOOTH	4
NC_OFS_TDC0_REF_CRK	114
NC_PSN_SEG_TDC_REF	108
NC_PHA_SEG_ER_ENSD	42
NLC_ESS	1
NC_NR_CONF_CAM	1
NC_NR_TOOTH_STALL	3
NC_CRK_WIN_SEG_LEN	18°CRK

edge index z	1	2
NC_PSN_EDGE_z_CAM_IN_1	564	204
NC_PSN_EDGE_z_CAM_EX_1	336	696

To be defined by project with information from customer. Remember: edge #1 is the first falling edge after TDC0, positions are given for VVT passive.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CRK_SENS_TYP	1	0H 1H	ACPS MCPS	1	-
Crankshaft sensor technology					
NC_CAM_WHEEL_TYP	1	0H 1H	Single-tooth Multi-teeth	1	-
Camshaft target wheel type					
NC_CAM_SENS_TYP	1	0H 1H 2H	MCAM ACAM ACAM TPO	1	-
Camshaft sensor technology					
NC_NR_CONF_CAM	1	1H ... 2H	1 ... 2	1	-
Number of camshaft target wheel configurations					
NLC_ESS	1	0H 1H	not present present	1	-
Engine speed (Tacho) hardware output controlled by basic SW present					

1.33 ENSD interface adaptation module

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO_CMN	V/O	0...FFH	-50...205	1	[°C]
Coolant temperature (ENSD internal)					
VB_CMN	V/O	0...FFH	0...28.8	1.13E-01	[V]
Battery voltage (ENSD internal)					

Input data:

TCO	VB	NC_CAM_LIH_SWI	

FUNCTION DESCRIPTION:

General information:

The aim is to get a common definition for coolant temperature and battery voltage inside ENSD aggregate whatever their definition is in the rest of software.

Description:

TCO_CMN and VB_CMN are stubbed

Application conditions:

Initialisation: none

Recurrence: at reset

Activation: *NC_CAM_LIH_SWI ≠ “INJ”*

Deactivation: NC_CAM_LIH_SWI = “INJ”

Formula section:

TCO_CMN = 0
VB_CMN = 0

Remark: As this module is only a stub, TCO and VB are not used.

1.34 ENTE Configuration data

Input data:

NLC_TCO_2_CONF			

FUNCTION DESCRIPTION:

General information:

The logical constants NC(NLC)_xxx_CONF are necessary to adapt the Aggregate to a given system environment. The values describe either the variant of a used system component, or a vehicle component is available or not.

With use of the logical constant **NLC_ECT_CONF** the availability of an electronically controlled thermostat within the system is determined. In case of a vehicle configuration with an ECT, the component control and diagnosis is included within the aggregate version. (refer: ENTE A.I.D.)

With use of the logical constant **NLC_TCO_2_CONF** the availability of a coolant temperature sensor at radiator outlet is determined. In case of a vehicle configuration with a TCO_2 sensor, the sensor signal acquisition and diagnosis is included within the aggregate version. (refer: ENTE A.I.D.)

The configuration constant **NC_ECF_CONF** allows to determine the wanted fan control strategy. In principal it is possible to choose between a RLY- or a PWM-control architecture. A RLY/PWM fan control strategy in parallel is also existing. In this case the aggregate architecture allows switching either the control of RLY-fan(s) or the control of PWM–fan(s) during ECU runtime depending on the setting of a corresponding configuration bit. The control of both variants (RLY- and PWM-fan(s)) at the same time is not allowed and not supported within the aggregate.

Independent on the chosen fan control strategy (PWM/RLY), the configuration constant **NC_ECF_NR** describes the number of available cooling fans (hardware components) at the vehicle. In case of a PWM-fan configuration, the number of available PWM cooling fans at the vehicle is always equal to the number of PWM fan output stages provided by the ECU hardware.

Because one or more RLY-switch(es) may control only one cooling fan (hardware component) in case of a RLY-fan configuration, the number of RLY-switch(es) per cooling fan can be different. With use of the configuration constant **NC_ECF_RLY_NR**, the number of RLY output stages per cooling fan are determined.

With the configuration constant NC_NR_TCO_SENS the number of available coolant temperature sensors is determined, which provide the sensor voltage values to calculate TCO, TCO_2, TCO_n.

The following describes the general rules for determination of the configuration data.

1.34.1 Local configuration data

Here are listed the configuration data, which are used only in the ENTE aggregate.

Data	Value
NLC_ECT_CONF	[0] Electronically controlled thermostat (ECT) is not available
NLC_TCO_2_CONF	[0] Coolant temperature sensor at radiator outlet is not available
NC_ECF_CONF	[2] Control of RLY- or PWM-fan(s) requested (RLY or PWM fan(s))
NC_ECF_NR	[1] Number of available cooling fans (hardware components) at the vehicle
NC_ECF_RLY_NR	[2] Number of RLY output stages per cooling fan (hardware components)
NC_NR_TCO_SENS	[1] Number of available coolant temperature sensors (hardware components) at the vehicle

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_ECT_CONF	1	0...1H	0...1	1	-
System configuration flag (compiler) if an electronic controlled thermostat is available (=1) or not (=0)					
NC_ECF_CONF	1	0...3H	0...3	1	-
System configuration switch for cooling fan control strategy (RLY || PWM || RLY and PWM)					
NC_ECF_NR	1	1...FFH	1...255	1	-
Number of available cooling fans (hardware components) at the vehicle					
NC_ECF_RLY_NR	1	0...FFH	0...255	1	-
Number of RLY output stages per cooling fan (hardware components)					
NLC_TCO_2_CONF	1	0...1H	0...1	1	-
System configuration flag (compiler) if a coolant temperature sensor at radiator outlet is available (=1) or not (=0)					
NC_NR_TCO_SENS	1	1...FFH	1...255	1	-
Number of available coolant temperature sensors (hardware components) at the vehicle					

1.35 INSY Configuration Data

FUNCTION DESCRIPTION:

General information:

The constant NC_xxx_CONF describes whether a system is equipped with additional functionality or not.

NC_MAF_FAC_CYL is used as conversion factor between maf (kg/h) and maf (mg/stk) based on the cylinder number.

The variable NC_CBK_IN_NR defines the number of intake cylinder lines.

1.35.1 Global configuration data:

Data	Value
NC_ETC_CONF	1 *(ETC version)*
NC_AMP_CONF	1 *(with AMP-sensor)*
NC_CBK_IN_NR	1 *(one intake cylinder line)*
NC_MAF_FAC_CYL	8333.3 (4 cylinder engine)

1.35.2 Local configuration data:

Data	Value
NC_CRCV_CONF	0
NC_MAF_CONF	1 *(allow MAF-controll)*
NC_MAP_CONF	1 *(allow MAP-controll)*
NC_NR_STATE_ACR	2 (system with one additional actuator (VIM))
NC_NR_LEN_CAM_EX	6 = six break points for considering exhaust cam adjustment
NC_NR_LEN_CAM_IN	8 = eight break points for considering intake cam adjustment
NC_NR_N_EFF_VOL_BAS	16 = sixteen N breakpoints
NC_NR_MAP_SAMPLE_ACQ	4 = last four samples within one segment used for V_MAP_MV calulation
NC_NR_SP_REQ_CAM_OFS_AD	8 (8 inlet/exhaust camshaft positions are available for adaptation purposes)
NC_NR_OPP_CAM_OFS_AD	1 (camshaft-offset adaptation is done at one engine speed operating point)
NC_NR_MAP_OPP_CAM_OFS_AD	1 (camshaft-offset adaptation is done at one MAP_MES operating point)
NLC_CAM_OFS_AD	1 (camshaft offset adaptation functionality is activated)
NC_NR_LEN_IP_TIA_IM_STND	4 (calculation of TIA_IM_STND at 4 engine speed and MAF break points)
NC_EFF_VOL_ACR_COR_CONF	2 (VIM and Port correction of EFF_VOL done in basic volumetric efficiency structure and in additional maps)
NC_PUT_CONF	1 = sensor always used

Data	Value
NLC_CAM_OFS_AD_EXT_REQ	0 = No external activation of Camshaft-offset adaptation functionality.

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_AMP_CONF	-	0...2H	0...2	1	[-]
System configuration flag (compiler) if an ambient pressure sensor is available (= 1) or not (= 0)					
NC_CRCV_CONF	-	0...1H	0...1	1	[-]
System configuration (compiler) flag to indicate if Crank case ventilation flow is considered (=1) or not(=0)					
NC_ETC_CONF	-	0...1H	0...1	1	[-]
System configuration (compiler) flag to indicate if a system with ETC is available (= 1) or not (= 0)					
NC_MAF_CONF	-	0...2H	0...2	1	[-]
System configuration flag (compiler) if an air-flow sensor is available (= 1) or not (= 0)					
NC_MAP_CONF	-	0...2H	0...2	1	[-]
System configuration flag (compiler) if a MAP sensor is available (= 1) or not (= 0)					
NC_MAF_FAC_CYL	-		33333.3 / CYL_NR	0.1695447	[-]
	3 Cyl. 4 Cyl. 5 Cyl. 6 Cyl. 8 Cyl.	FFFFH C000H 9999H 8000H 6000H	11111.1 8333.3 6666.6 5555.5 4166.6		
conversion factor between MAF					
NC_CBK_IN_NR	-	1...4H	1...4	1	[-]
Number of intake cylinder banks					
NC_NR_STATE_ACR	-	0...FFH	0...255	1	[-]
Number of actuators which are influencing EFF_VOL; default value 1					
NC_NR_N_EFF_VOL_BAS	-	1...FFH	1...255	1	[-]
Number of N breakpoints					
NC_NR_LEN_CAM_EX	-	1...FFH	1...255	1	[-]
Number of CAM_EX break points; default value 1 (= no CAM_EX movement)					
NC_NR_LEN_CAM_IN	-	1...FFH	1...255	1	[-]
Number of CAM_IN break points; default value 1 (= no CAM_IN movement)					
NC_NR_MAP_SAMPLE_ACQ	V/O	1...5H	1...5	1	[-]
Number of used samples for V_MAP_MV calculation under transient conditions					
NC_NR_SP_REQ_CAM_OFS_AD	-	1...8H	1...8	1	[-]
Number of camshaft-setpoints available for camshaft-offset adaptation (default value=8)					
NC_NR_OPP_CAM_OFS_AD	-	1...4H	1...4	1	[-]
Number of N/MAP operating points at which the adaptation can be done					
NC_NR_MAP_OPP_CAM_OFS_AD	1	1...5H	1...5	1	[-]
Number of MAP_MES operating points where the Camshaft-offset adaptation can be done.					
NLC_CAM_OFS_AD	1	0...1H	0...1	1	[-]
Camshaft-offset adaptation functionality present(=1) or not present(=0) in system					
NC_NR_LEN_IP_TIA_IM_STND	1	1...6H	1...6	1	[-]
Number of axis breakpoints for calculation of TIA_IM_STND					
NC_EFF_VOL_ACR_COR_CONF	1	1...3H	1...3	1	[-]
System configuration flag to indicate where the VIM and PORT correction of EFF_VOL is done					
NC_PUT_CONF	1	0...2H	0...2	1	[-]
Pressure upstream throttle sensor configuration					
NLC_CAM_OFS_AD_EXT_REQ	1	0...1H	0...1	1	[-]
System configuration flag to indicate whether Camshaft-offset adaptation can be externally activated(=1 for e.g. through end-of-line test) or not(=0)					

1.36 CHRG Configuration Data

General information:

Every constant NC_xxx_CONF describes whether a system is equipped with additional functionality or not.

- Sensor-Configuration

For all sensors (NC_xxx_CONF) following possibilities are available:

0 = NO_SENSOR: sensor is not available, value never used
1 = SENSOR_ALWAYS_USED: sensor is always available and value is used
2 = SENSOR_SWITCHABLE: sensor-evaluation can be switched "on / off "during runtime (by LC_switch)

For information it's possible to show the usage later usage of the configuration: (TB) only for engine-testbench, (SER) also for series-production

- Variant-Selection

some NC-switches can be used to select between spec-versions, some can be used as normal compiler-switches (e.g. to select sensor-values for evaluation)

Type: C = Compilerswitch (NC-name usable within CHRG-Software-modules)

D = Spec-selection for different versions for DOKU

(NC-name only information, not usable)

Also a combination compilerswitch + Spec-Variant-selection "C/D" is allowed.

HINT:

M266 has 2 engine-variants: turbocharger and selfinduced system, each with a different sensor- and HW configurations. In order to have the same software for both variants, the sensor-configuration can be selected by LC_TCHA_CONF at power-up of the ECU.

Application conditions:

Initialisation: at ECU reset

Recurrence: none

1.36.1 Global configuration data

Here the configuration data are listed, which can be used in other aggregates :

Data	Value
NC_CHRG_CONF	1 = turbocharger
LC_TCHA_CONF	1 = turbocharger 0 = selfinduced engine (via data bin)

1.36.2 Local configuration data

Here the configuration data are listed, which are used only in the CHRG aggregate.

1.36.3 Turbine speed sensor (N_TCHA)

NC_N_TCHA_CONF	C/D	turbo charger speed sensor (TB) 0 = NO_SENSOR

1.36.4 Wastegate (WG) - location: exhaust path

Data	Type	Value
NC_PRS_WG_ACR_CONF	C/D	wastegate-actuator pressure sensor (TB) 0 = NO_SENSOR

1.36.5 Recirculation-system (RCL) - location: intake path

NC_PSN_RCL_CTL	C/D	recirculation-actuator position-control: 1 = PSN_RCL_DIG_CTL: digital control (ON/OFF)
NC_RCL_UP_ICO	C	position of recirculation pipe inlet 1 = upstream intercooler
NC_PSN_RCL_CONF	C/D	recirculation-actuator position sensor 0 = NO_SENSOR

1.36.6 general switches

NC_MAF_RNG_H_CHRG	C/D	Switch whether old/low or new/high MAF_KGH range is used 0 = USE_LOW_MAF_KGH_RANGE
NC_BOOST_MNG_CHRG	C/D	Switch whether Dynamic boost manager is supported 0 = DYNAMIC_BOOST MANAGER_NOT_SUPPORTED
NC_NR_TCHA	C	Number of turbo chargers in the system 1 = one turbo charger - Mono turbo system

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TCHA_CONF	1	0...1H	0...1	1	-
System configuration flag for turbo charged engine (=1), for naturally aspirated engine (=0)					

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_CHRG_CONF	1	0h..2h	0..2	1	-
Charger configuration					
NC_PSN_WG_CONF	1	0h..2h	0..2	1	-
Wastegate position sensor configuration					
NC_PSN_RCL_CONF	1	0h..2h	0..2	1	-
Recirculation actuator position sensor configuration					
NC_TIA_CHA_UP_CONF	1	0h..2h	0..2	1	-
Intake temperature upstream charger sensor configuration					
NC_TIA_CHA_DOWN_CONF	1	0h..2h	0..2	1	-
Intake temperature downstream charger sensor configuration					
NC_TEG_TUR_UP_CONF	1	0h..2h	0..2	1	-
Exhaust gas temperature upstream turbine sensor configuration					
NC_N_TCHA_CONF	1	0h..2h	0..2	1	-
Turbo charger speed sensor configuration					
NC_MAF_RCL_CONF	1	0h..2h	0..2	1	-
Recirculation air mass flow sensor configuration					
NC_PRS_WG_ACR_CONF	1	0h..2h	0..2	1	-
Wastegate actuator pressure configuration					
NC_PSN_WG_CTL	1	0h..1h	0..1	1	-
Wastegate position control configuration					
NC_RCL_UP_ICO	1	0h..1h	0..1	1	-
position of recirculation pipe inlet					
NC_PSN_WG_MDL	1	0h..1h	0..1	1	-
wastegate position model type					
NC_PRS_WG_ACR_MDL	1	0h..1h	0..1	1	-
wastegate actuator pressure model type					
NC_PSN_RCL_CTL	1	0h..2h	0..2	1	-
recirculation actuator position control					
NC_PRS_CHA_UP_CONF	1	0h..2h	0..2	1	-
Pressure upstream charger sensor configuration					
NC_MAF_RNG_H_CHRG	1	0h..1h	0..1	1	-
Switch whether old/low or new/high MAF_KGH range is used					
NC_BOOST_MNG_CHRG	1	0h..1h	0..1	1	-
Switch whether Dynamic boost manager is supported					
NC_NR_TCHA	1	0h..2h	0..2	1	-
Number of turbo chargers					
NC_TYP_SENS_TEG_TUR_UP	1	0h..1h	0..1	1	-
Type of exhaust gas temperature sensor (0 – analog voltage output, 1 – PWM output)					

1.37 TQDR Configuration Data

Input data:

NC_CYC_ADD_ACT			

General information :

The following describes the general rules for determination of the configuration data

1.37.1 Local configuration data

Here are listed the configuration data, which are used only in the TQDR aggregate.

Data	Value
NC_CYC_ADD_ACT	1

1.38 ERRM Configuration Data

Input data:

NC_ABC_CONF_FCT_DIAG_XX	NC_NR_DTC_FMT	NC_ERR_DTC_CONF
NC_NR_ENVD_CUS_SET_CMN	NC_MIL_CHK_TYP	NC_NR_ENVD_CUS_CMN
NC_NR_ENVD_PREV	NC_NR_ENVD_CUS_SET_SPC	NC_NR_ENVD_OBD
NC_NR_FRF_SET	NC_NR_ERR_DYN	NC_NR_ERR_HIS
NC_NR_WIN_SCDN	NC_NR_HIS	NC_NR_SYM_XX
NC_ENVD_CUS_CMN_UPD	NC_ERR_DTC_REQ_CUS	NC_ERR_DTC_REQ_OBD
NC_NR_ERR_INTM	NLC_ABC_INI_DC_END_DIAG	NC_NR_LAMP
NC_ERR_PRI_H	NLC_INC_ERR_PRI	NLC_OLD_ERR_PRI
NLC_OBD_FRF_PND	NLC_FMY_CONV_FOR_FRF	NC_NR_DIAG_RBM
NLC_BENCH_MODE	NC_LDP_1_DTC_TABLE_SIZE	NLC_MIL_ACT_REQ
NC_LDP_2_DTC_MIS_TABLE_SIZE	NC_LDP_2_DTC_TABLE_SIZE	NC_NR_ERR_PERM
NLC_OBD_DSL	NC_NR_DIAG_SYM_RBM	

General information :

The following describes the general rules for determination of the configuration data

1.38.1 Global configuration data

Data	Value	
	Typical values are listed below	
NLC_OBD_RBM_ENA	1	Rate base monitoring functionality present or not : 0 : Rate base monitoring disabled **1 : Rate base monitoring enabled**
NLC_ENA_SCDN_NEW	1	Old or new Similar conditions functionality used: 0: old SCDN functionality used 1: new SCDN functionality used
NLC_ENA_SCDN	1	Similar conditions present or not : 0 : Similar conditions disable, acceptable for Europe applications. 1 : Similar conditions enable, for US applications
NLC_CAN_LAMP_ACT	0	Warning lamps interface type 0: warning lamps hardwired to ECU 1: warning lamps status sended to CAN inter-system
NC_ERR_DET_UPD	01h	Configuration for all diagnoses : 00h: no sporadic error detection 01h: sporadic error detection 02h: sporadic error detection with extended information
NLC_ENA_MULTI_CDN	1	Activation/deactivation of multi-condition debounce algorithm 0 : Multi-condition debounce algorithm unused 1 : Multi-condition debounce algorithm used
NLC_OBD_READY_CAL	0	Activation/deactivation of calibratable readiness classes : 0 : readiness classes are not calibratable 1 : readiness classes are calibratable
NLC_ENA_SYM_DIAG_AD	0	Activation/deactivation of adaptations modules for symptom based diagnostics : 0 : adaptation not required 1 : adaptation used to connect symptom based diagnostics
NLC_ENA_RBM_DIAG_AD	0	Activation/deactivation of adaptations modules for symptom based diagnostics with rate-based monitoring (always <= NC_NR_DIAG_RBM) 0 : adaptation not required 1 : adaptation used to connect symptom based diagnostics to rate-based monitoring

1.38.2 Local configuration data

Here are listed the configuration data, which are used only in the ERRM aggregate.

Data	Value	
	Typical values are listed below	
NC_NR_ERR_DYN	10	Dynamic memory size, quantity of failures
NLC_OLD_ERR_PRI	1	Failure chronological priority order 0: priority to new failures **1: priority to old failures**
NLC_INC_ERR_PRI	1	Set this bit to 0 to disable increased failure priority **Set this bit to 1 to enable increased failure priority**
NC_ERR_PRI_H	6	failure priority level linked to NLC_INC_ERR_PRI
NC_NR_ENVD_OBD	33	Size of OBD environmental data, quantity of data byte
NC_NR_ENVD_CUS_CMN	0	Size of common environmental data, quantity of data byte
NC_NR_ENVD_CUS_SET_CMN	12	Size of environmental data set, quantity of data byte
NC_NR_ENVD_CUS_SET_SPC	4	Size of specific environmental data set, quantity of data byte
NC_NR_FRF_SET	1	Quantity of environmental data sets, quantity of sets
NC_NR_ENVD_PREV	2	Number of different failure instances xx using prestored freeze frame **00h:**No prestored freeze frame functiunality used **01h...FFh:**Quantity of failure instances XX using prestored freeze frame.
NC_ ENVD_CUS_CMN_UPD	1	Way to update environmental data 0: no update at failure status change **1: update at failure status change (same as obd)**
NC_NR_WIN_SCDN	xx	Quantity of diagnosis using similar conditions, quantity of diagnosis *For exact number see Chapter "Similar conditions (Appl. Inc.)" 5WB02U01.yyy*
NC_NR_ERR_HIS	10	Size of history memory. quantity of failures
NC_NR_HIS	22	Size of data stored for a single failure in history memory, quantity of data bytes
NC_ERR_DTC_REQ_OBD	0	Type of obd/carb DTCs returned by api **0 : detailed for each symptom** 1 : limited to the global failure
NC_ERR_DTC_REQ_CUS	0	Type of customer DTCs returned by api **0 : detailed for each symptom** 1 : limited to the global failure

Data	Value	
	Typical values are listed below	
NC_ERR_DTC_CONF	0	Selection of DTCs displaying strategies for ERR_DTC_IDX : 0 : One unique DTC which corresponds to the first symptom detected (ERR_SYM_MEM_IDX). The DTC is encoded according OBD J2012.1 : One unique global DTC whatever the symptom detected. The DTC is encoded according OBD J2012. 2 : One unique DTC which corresponds to the first symptom detected (ERR_SYM_MEM_IDX). The DTC is encoded according customer specific format. 3 : One unique global DTC whatever the symptom detected. The DTC is encoded according customer specific format. 4 : One unique DTC which corresponds to the last symptom detected (ERR_SYM_LST_IDX). The DTC is encoded according OBD J2012. 5 : One unique DTC which corresponds to the last symptom detected (ERR_SYM_LST_IDX). The DTC is encoded according customer specific format.
NC_NR_DTC_FMT	0	Selection of 6 or 10 DTCs configurations Defines 1 or 2 different DTC identifiers per symptom. Defintion with 1 DTC permits DTC usage in common for OBD and customer. 0 : a single DTC per symptom (a single one for OBD and cutomer) 1 : 2 DTCs defined per symptom (one for OBD, one for cutomer)

Data	Value	
	Typical values are listed below	
NC_MIL_CHK_TYP	3h	Pre-drive check strategy on stalling event : Bit0=0 : Disable pre-drive check after “stalling event/engine start” **Bit0=1 : Enable pre-drive check after “stalling event/engine start”** Display readiness status during pre-drive check (MIL blinking) : Bit1=0 : disable readiness status display during pre-drive check **Bit1=1 : enable readiness status display during pre-drive check**
NLC_OBD_FRF_PND	0	Defines the eobd/carb freeze frame strategy the following ways : **0 : eobd/carb freeze frame data of mode 2 will be returned after confirmed failure detection** 1 : eobd/carb freeze frame data of mode 2 will be returned after confirmed/emission relevant failure If confirmed in memory, after pending/emission relevant detection If not confirmed in memory.
NC_NR_DIAG_RBM	xx	Instance number (1...255) of monitors with real in-use performances tracked by the Rate-Based monitoring functionality. *For exact number see chapter* "Rate base monitoring (Appl. Inc.)" 5WB045xx.yyy
NC_NR_SYM_XX	 03h 07h	Configuration to define available symptom for each diagnosis XX : This constant is used in the multicondition filtering algorithm. 03h: SYM_0 & SYM_1 available 07h: SYM_0...SYM_2 available Please take note that this configuration shall be mentioned in “table of failure” module for each diagnosis
NC_ABC_CONF_FCT_DIAG_XX	 00h 03h 0Bh 0Eh 10h 13h	Configuration depends on each diagnosis XX behaviour : 00h: Standard configuration STD 03h: Standard configuration STD_INI with initialisation(only this config is available for multicondition) 0Bh: memory configuration MEM 0Eh: memory configuration MEM_INI 10h: decrement calibration configuration DEC_CAL 13h: statistical configuration STC Please take note that this configuration shall be mentioned in “table of failure” module for each diagnosis
NLC_ABC_INI_DC_END_DIAG	01h	Configuration for all diagnoses : 00h: diagnosis init managed at transition LV_IGK 0→1 01h: end of diagnosis init managed at transition LV_DC 1→0 (only this configuration is available for multicondition filtering)
NC_NR_ERR_INTM	10	Maximum number of diagnosis instances to be listed in ERRM_INTM_DIAG_INST[NC_NR_ERRM_INTM] array
NLC_BENCH_MODE	1	Configuration for all diagnoses, to allow/forbid the bench mode activation. The maximum CPU optimisation is reached when NLC_BENCH_MODE =0. 0 : bench mode can’t be activated whatever LC_ABC_BENCH value 1 : bench mode activation is possible when LC_ABC_BENCH = 1
NLC_FMY_CONV_FOR_FRF	1	Conversion of data before storage in the specific freeze frame 0: Data are not converted before the storage in the specific freeze frame 1: Store the converted data in the specific freeze frame

Data	Value	
	Typical values are listed below	
NC_LDP_2_DTC_MIS_TABLE_SIZE	6	Size of LDP_2_ID_ERR_DTC_MIS_SIZE axis (= NC_CYL_NR + 2)
NC_LDP_2_DTC_TABLE_SIZE	6	Size of LDP_2_ID_ERR_DTC_XX_SIZE axis (= 6 - NC_NR_DTC_FMT)
NC_LDP_1_DTC_TABLE_SIZE	1	Size of LDP_1_ID_ERR_DTC_XX_SIZE axis (= 1 + NC_NR_DTC_FMT)
NC_NR_LAMP	1	Number of warning lamps
NLC_MIL_ACT_REQ	1	External MIL request is taken or not into account for DIST_ACT_MIL and T_ACT_MIL calculations. 0 : External MIL request not taken into account 1 : External MIL request taken into account
NC_NR_ERR_PERM	4	Maximum number of failure defined in permanent failure memory structure (CARB required minimum value : 4)
NLC_OBD_DSL	0	Selection of engine type : gasoline (0) or diesel (1)
NC_NR_DIAG_SYM_RBM	0	Instance number of monitor designed for symptom based error management and connected to rate-based monitoring (Typical value : 0)

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_ENA_SCDN	1	0...1H	0...1	1	[-]
Boolean to indicate if similar conditions functionality is present (1) or not (0)					
NLC_ENA_SCDN_NEW	1	0...1H	0...1	1	[-]
Boolean to indicate if new similar conditions functionality is used (1) or not (0)					
NC_ERR_DET_UPD	1	0...2H	0...2	1	[-]
Activation of sporadic errors storage and visualisation (0: disabled, 1: enabled)					
NLC_OBD_RBM_ENA	1	0...1H	0...1	1	[-]
Rate-based monitoring functionality present or not (0: Rate-based monitoring disabled, 1: Rate-based monitoring enabled)					
NLC_CAN_LAMP_ACT	1	0...1H	0...1	1	[-]
Selection of the strategy to drive warning lamps (0: wire, 1: CAN)					
NLC_OBD_READY_CAL	1	0...1H	0...1	1	[-]
Activation/deactivation of calibratable readiness classes					
NLC_ENA_SYM_DIAG_AD	1	0...1H	0...1	1	[-]
Activation/deactivation of adaptations modules for symptom based diagnostics connection					
NLC_ENA_MULTI_CDN	1	0...1H	0...1	1	[-]
Activation/deactivation of multi-condition debounce algorithm (0: disabled, 1: enabled)					
NLC_ENA_RBM_DIAG_AD	1	0...1H	0...1	1	[-]
Activation/deactivation of adaptations modules for symptom based diagnostics with rate-based monitoring (always <= NC_NR_DIAG_RBM)					

1.39 EGCP Configuration Data

Input data:

NC_LSHPWM_BOL_LSH_DOWN	NC_LSHPWM_TOL_LSH_DOWN	NC_LSHPWM_BOL_LSH_UP	NC_LSHPWM_TOL_LSH_UP
NC_R_REF_LS_DOWN	NC_VLS_DOWN_CUR_PUMP_REF	NC_FAC_R_REF_LS_DOWN	NC_VLS_UP_CUR_PUMP_REF
NC_FAC_R_REF_LS_UP	NC_R_REF_LS_UP		

General information :

The following describes the general rules for determination of the configuration data

1.39.1 Global configuration data

Here are listed the configuration data, which can be used in other aggregates :

Data	**Value**
NC_CBK_EX_NR	1
NC_STATE_LSL_UP_IF	0
NC_STATE_VLS_UP_SIG_ACQ	0
NLC_LSH_RLY_EFP	0
NC_FRQ_LSHPWM_UP	-10
NC_FRQ_LSHPWM_DOWN	7,6
NC_FAC_R_REF_LS_UP	0.118316 (=5.62/47.5)
NC_R_REF_LS_UP	5620
NC_VLS_UP_CUR_PUMP_REF	4.82
NC_FAC_R_REF_LS_DOWN	0.118316 (=5.62/47.5)
NC_R_REF_LS_DOWN	5620
NC_VLS_DOWN_CUR_PUMP_REF	4.82

1.39.2 Local configuration data

Here are listed the configuration data, which are used only in the EGCP aggregate.

Data	**Value**
NC_LSHPWM_BOL_LSH_DOWN	1 %
NC_LSHPWM_TOL_LSH_DOWN	99 %
NC_LSHPWM_BOL_LSH_UP	1 %
NC_LSHPWM_TOL_LSH_UP	99 %

To be defined by project with information from customer.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CBK_EX_NR	1	1...4H	1...4	1	-
Number of exhaust cylinder banks					
NC_STATE_VLS_UP_SIG_ACQ	1	0...7H	0...7	1	-
Configuration switch for optional signal acquisition					
NLC_LSH_RLY_EFP	1	0...1H	0...1	1	-
Configuration switch to indicate whether lambda sensor heater is connected to EFP or main relay					
NC_STATE_LSL_UP_IF	1	0...7H	0...7	1	-
Interface to lambda sensor linear upstream					
NC_FRQ_LSHPWM_UP	1	0...FFFFH	0...4095.9375	0.0625	Hz
PWM frequency for the upstream oxygen sensor heater					
NC_FRQ_LSHPWM_DOWN	1	0...FFFFH	0... 4095.9375	0.0625	Hz
PWM frequency for the downstream oxygen sensor heater					

Notes:

- *The configuration switch NC_STATE_VLS_UP_SIG_ACQ denotes whether signal filtering shall be applied to the linear lambda sensor's raw signal provided by the BSW prior to further signal processing. If signal filtering shall be applied NC_STATE_VLS_UP_SIG_ACQ shall be set to 1. If not or if binary upstream sensor is used NC_STATE_VLS_UP_SIG_ACQ shall be set to 0.*

- *The PWM frequency for the oxygen sensor heater must be adjusted to values larger than 5Hz, i.e.* NC_FRQ_LSHPWM_UP >= 5Hz and NC_FRQ_LSHPWM_DOWN >= 5Hz. *Usually no values larger than 100 Hz are applied, the exact value has to be derived from the corresponding sensor specification.*

1.40 Variant and Version Coding

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CONF_TCU	V/O/S	0...0FH	0...15	1	-
Status information for gearbox type					
CONF_ACC	V/O/S	0...1H	0...1	1	-
Status information for climate control					
CTR_ACC_LEARN	V	0...FFH	0...255	1	-
Counter for learning of air condition control					
CONF_LAM	V/O	0...1H	0...1	1	-
Status information for oxygen sensor configuration					
CONF_CAT_EFF	V/O	0...1H	0...1	1	-
Status information for catalyst efficiency diagnosis					
CONF_MIL_FMY	O	0...5H	0...5	1	-
Status information for malfunction indicator lamp and failure memory management					
CONF_IMOB	V/O	0...1H	0...1	1	-
Status information for immobilizer					
CONF_VS	V/O/S	0H 1H 2H 3H 4H	GEAR_BOX ABS WHEEL LEARNING TCU_CAN	1	-
Status information for vehicle speed acquisition					
CONF_CP	V/O	0...2H	0...2	1	-
Status information for canister purge activation					
CONF_DIAGCP	V/O	0...2H	0...2	1	-
Status information for canister purge diagnosis activation					
CONF_DIAGCP_VOL	V/O	0... 2H	0...2	1	-
Status information for fuel tank level acquisition for canister purge diagnosis					
CONF_CFA	V/O	0...1H	0...1	1	-
Status information for selection of fan control method					
CONF_ACP	V/O	0...1H	0...1	1	-
Status information for AC pressure acqusition					
CONF_PSTE	V/O	0...1H	0...1	1	-
Status information for power steering pressure acqusition					
CONF_TOIL_MDL	V/O	0...2H	0...2	1	-
Status information for TOIL acquisition					
VS_ABS	-	0...FFH	0...255	1	km/h
Vehicle speed signal from ABS-input					
VS_WHEEL_INDU	-	0...FFH	0...255	1	km/h
Vehicle speed signal from inductive wheel speed sensor-input					
LV_AT	V/O	0...1H	0...1	1	-
= 1 in case of Automatic gear box					
CONF_AD_RST	V/O	0...2H	0...2	1	-
Status information for erasement of adaptative values					
CONF_FMY_RST	V/O	0...1H	0...1	1	-
Status information for erasement of failure memory					
CONF_CRU	V/O/S	0...1H	0...1	1	-
Status information for cruise control function activation (only available for ETC Configuration)					
CONF_GEN_LOAD	V/O	0...1H	0...1	1	-
Status information for GEN_LOAD function activation					
CONF_VIM	V/O	0...2H	0...2	1	-
Status information for Variable Intake Manifold Configuration					
CONF_ASR_SLOW	V/O	0...1H	0...1	1	-
Status Information for Slow ASR Request from TCS Control Unit available (only available for ETC Configuration)					
CONF_KOBD	V/O	0...1H	0...1	1	-
Status information for supporting KOBD specific requirement					
CONF_CAM_VVT_EX	V/O	0...1H	0...1	1	-

Status Information for exhaust camshaft configuration					
CONF_PORT	V/O	0...1H	0...1	1	-
Status Information for Port Flap configuration					
CONF_MAF	V/O	0...1H	0...1	1	-
Status information of the system load sensor					
CONF_TAM	V/O/S	0...2H	0...2	1	-
Status information for ambient temperature configuration					
T_TAM_LEARN	-	0...FFH	0...255	1	s
Time for TAM learning					
CONF_TCS	O	0...1H	0...1	1	-
Status information of the traction control unit					
LV_TEACH_CAN_FINISHED	O	0...1H	0...1	1	-
Status information of the traction control unit					
CONF_TCU_LEARN	O	0...1H	0...1	1	-
Status information of the traction control unit					
T_TEACH_CONF	-	0...FFH	0...255	1	s
Timer to limit teach time for TCU/TCS learning					
CONF_YEAR	V/O	30...53H	2000...2035	1	-
Model Year information for VIN coding					
CONF_BAT	V/O	0...1H	0...1	1	-
Status information for Battery Management					
CONF_BAT_LIN	V/O	0...2H	0...2	1	-
Status information for Sensor type used for Battery management					
CONF_PC_FAN	V/O	0... 2H	0...2	1	-
Status information for Passenger compartment fan					
CONF_ETL_TCU	V/O	0...2H	0...2	1	-
Status information if TCU is supporting engine torque limitation (slow path)					
CONF_ACC_FATC	V/O/S	0...1H	0...1	1	-
Status information for air con control by external unit					
T_ACC_FATC_LEARN	-	0...FFH	0...255	1	s
Timer to limit teach time for ACC_FATC learning					

Input data:

LV_CDN_VB_MIN_DIAG	VS	TCU_TYPE	LV_IM_ACIN
LC_TCHA_CONF	LV_ERR_CAN_BUS_OFF	LV_TAM_CAN_FIRST_VLD	LV_TCS1_CAN_VLD
LV_CRU_MAIN_SWI	LV_IM_ACIN_CAN	LV_ACC_FATC_CAN_FIRST_VLD	NC_ETC_CONF

Import actions:

ACTION_INFR_GetLvImAcin(OUT < lv_im_acin >**)**

FUNCTION DESCRIPTION:

General information:

In order to minimize the necessary amount of different ECU variants by different equipment in the vehicles under mass production conditions, it is necessary to build a structure into the ECU software that allows to select different control algorithm and access to different calibration data maps and constants by setting of configuration bytes.

It is possible to change the variants during the development process by SAM 2000 whereas the versions could be selected by KW 2000 (SAM-Dias or service tester) at the end of line-programming or at the service. The underlined variant respectively version is the default one.

Remark: A change of a version or variant will only be valid after system reset (that means when the power latch phase has expired).
This applies also to the outputs CONF_xxx and CONF_xxx_CAN.

Exception: CONF_CRU is learnt after Key-On without Powerlatch, it is effective immediately.

1.40.1 Variant coding

Oxygen sensor configuration :

CONF_LAM = C_CONF_LAM

C_CONF_LAM = 0 Open loop

C_CONF_LAM = 1 Lambda control

Vehicle speed acquisition :

C_CONF_VS = GEAR_BOX Vehicle speed signal from gear box (non-OBD II variant)

C_CONF_VS = ABS Vehicle speed signal from inductive wheel speed sensor or from ABS control unit (OBD II variant)

C_CONF_VS = TCU_CAN Vehicle speed signal from TCU via CAN (non-OBD II variant)

C_CONF_VS = 0	CONF_VS = GEAR_BOX	Vehicle speed signal from gear box
C_CONF_VS = 1	CONF_VS = ABS	Vehicle speed signal from ABS control unit
	CONF_VS = WHEEL	Vehicle speed signal from inductive wheel sensor
	CONF_VS = LEARNING	Vehicle speed signal learning currently active
C_CONF_VS = 2	CONF_VS = TCU_CAN	Vehicle speed signal from TCU via CAN

Immobilizer :

CONF_IMOB = C_CONF_IMOB

C_CONF_IMOB = 0 Passive

C_CONF_IMOB = 1 Active

Once the CONF_IMOB has been set for an ECU, it can be reset only by the service tester (not anymore by SAM2000).

Evaporative system :

C_CONF_CP = 0 Passive

C_CONF_CP = 1 Active with open loop lambda control

C_CONF_CP = 2 Active with closed loop lambda control

	CONF_LAM = 0	CONF_LAM = 1
C_CONF_CP = 0	CONF_CP = 0 (passive)	CONF_CP = 0 (passive)
C_CONF_CP = 1	CONF_CP = 1 (open loop)	CONF_CP = 1 (open loop)
C_CONF_CP = 2	CONF_CP = 1 (open loop)	CONF_CP = 2 (closed loop)

Evaporative system diagnosis :

C_CONF_DIAGCP = 0 Passive

C_CONF_DIAGCP = 1 Active : single run per driving cycle

C_CONF_DIAGCP = 2 Active : continuous run

	CONF_CP = 0	CONF_CP = 1	CONF_CP = 2
C_CONF_DIAGCP = 0 passive	CONF_DIAGCP = 0 passive	CONF_DIAGCP = 0 passive	CONF_DIAGCP = 0 passive
C_CONF_DIAGCP = 1 single run	CONF_DIAGCP = 0 passive	CONF_DIAGCP = 0 Passive	CONF_DIAGCP = 1 active
C_CONF_DIAGCP = 2 continuous	CONF_DIAGCP = 0 passive	CONF_DIAGCP = 0 Passive	CONF_DIAGCP = 2 active

Fuel tank level acquisition for EVAP-diagnosis

CONF_DIAGCP_VOL = C_CONF_DIAGCP_VOL

C_CONF_DIAGCP_VOL = 0 fuel tank level estimation by DTP signal gradient

C_CONF_DIAGCP_VOL = 1 fuel tank level acquisition by single fuel gauge

C_CONF_DIAGCP_VOL = 2 fuel tank level acquisition by dual fuel gauge

Failure management and MIL Handling Information:

CONF_MIL_FMY = C_CONF_MIL_FMY

CONF_MIL_FMY = 0	no MIL, Failure Memory Management OBD-I
CONF_MIL_FMY = 1	MIL treatment OBD-I, Failure Memory Management OBD-I
CONF_MIL_FMY = 2	MIL treatment OBD-II, Failure Memory Management OBD-II *Exception :* *MIL is turned on in first driving cycle for comprehensive components (CARB_COMP_MIL)*
CONF_MIL_FMY = 3	MIL treatment OBD-II, Failure Memory Management OBD-II
CONF_MIL_FMY = 4	MIL treatment E-OBD, Failure Memory Management E-OBD
CONF_MIL_FMY = 5	MIL treatment KOBD, Failure Memory Management KOBD

Variable valve timing (only exhaust camshaft)

CONF_CAM_VVT_EX = C_CONF_CAM_VVT_EX

C_CONF_CAM_VVT_EX = 0	exhaust camshaft without adjustment
C_CONF_CAM_VVT_EX = 1	exhaust camshaft with adjustment

Erasement of adaptative values :

CONF_AD_RST = C_CONF_AD_RST

C_CONF_AD_RST = 0	the adaptation values are **not** reset
C_CONF_AD_RST = 1	**all** the adaptation values are reset
C_CONF_AD_RST = 2	only **some** adapt. values are reset. See details in chap. 0

Erasement of failure memory :

CONF_FMY_RST = C_CONF_FMY_RST

C_CONF_FMY_RST = 0	then the failure memory is **not** reset in case of Battery has been disconnected since last driving cycle (LV_VB_OFF=1)
C_CONF_FMY_RST = 1	then the failure memory is reset in case of Battery has been disconnected since last driving cycle (LV_VB_OFF=1)

Status of selection of fan control method:

CONF_CFA = C_CONF_CFA

C_CONF_CFA = 0	controlled by RLY_CFA_HIGH/LOW
C_CONF_CFA = 1	controlled by CFAPWM

Acquisition of AC pressure:

CONF_ACP = C_CONF_ACP

C_CONF_ACP = 0	PRS_ACC from Triple Switch
C_CONF_ACP = 1	ACP from A/C Pressure Transducer

Acquisition of Power steering pressure:

CONF_PSTE = C_CONF_PSTE

C_CONF_PSTE = 0	Pressure Switch
C_CONF_PSTE = 1	Pressure from Pressure Transducer

Acquisition of TOIL:

CONF_TOIL_MDL = C_CONF_TOIL_MDL

C_CONF_TOIL_MDL = 0	TOIL from oil temperature sensor (if error free)
C_CONF_TOIL_MDL = 1	TOIL calculated from Toil model for Fuel system diagnosis
C_CONF_TOIL_MDL = 2	TOIL calculated from Toil model (TOIL_MDL)

Acquisition of Cruise Signal: (only available for ETC Configuration)

if	CONF_CRU = 0		Non-Cruise Vehicle or Cruise not yet learnt
then	**if**	LV_CRU_MAIN_SWI = 1	Cruise Control Main Switch after filtering
	then	CONF_CRU = 1	Cruise Control function learnt

Rermarks:

- LV_CRU_MAIN_SWI is calculated each 10 msec, therefore CONF_CRU is also calculated with 10 msec recurrence. Once CONF_CRU is learnt, it is effective immediately w/o PWL.
- CONF_CRU can be reset from 1 to 0 by KWP2000 service (see Keyword Spec).

Acquisition of GEN_LOAD:

CONF_GEN_LOAD = C_CONF_GEN_LOAD

C_CONF_GEN_LOAD = 0	GEN_LOAD is not available from alternator.
C_CONF_GEN_LOAD = 1	GEN_LOAD is available from alternator.

Catalyst efficiency diagnosis :

CONF_CAT_EFF = C_CONF_CAT_EFF

C_CONF_CAT_EFF = 0	Passive
C_CONF_CAT_EFF = 1	Active

Status of Variable Intake Manifold:

CONF_VIM = C_CONF_VIM

C_CONF_VIM = 0 no variable Intake Manifold available

C_CONF_VIM = 1 variable Intake Manifold available without position feedback

C_CONF_VIM = 2 variable Intake Manifold available with position feedback

Support for KOBD specific requirement:

CONF_KOBD = C_CONF_KOBD

C_CONF_KOBD = 0 Not support for KOBD specific requirement

C_CONF_KOBD = 1 Support for KOBD specific requirement

Status of Slow ASR Request Availability from TCS Control Unit:
(only available for ETC Configuration)

CONF_ASR_SLOW = C_CONF_ASR_SLOW

C_CONF_ASR_SLOW = 0 Slow Torque Request from TCS Control Unit not available

C_CONF_ASR_SLOW = 1 Slow Torque Request from TCS Control Unit available

Status of Port Flap:

CONF_PORT = C_CONF_PORT

C_CONF_PORT = 0 no Port Flap (TCI)

C_CONF_PORT = 1 Port Flap (Facelift, SULEV)

System Load Sensor:

CONF_MAF = C_CONF_MAF

C_CONF_MAF = 0 A MAP-Sensor is used as Load Sensor

C_CONF_MAF = 1 A MAF-Sensor is used as Load Sensor

Model Year information for VIN coding with sample ECU :

The main purpose of this configuration data is to avoid VIN failure message during J1699 test with sample ECU during development period.
With this year information, VIN code is to be automatically created.
Of course, this function is disabled at production SW.

CONF_YEAR = C_CONF_YEAR ⇒ Just write a model year.

TCU Learning:

CONF_TCU_LEARN = C_CONF_TCU_LEARN

C_CONF_TCU_LEARN = 0 no TCU learning via CAN

C_CONF_TCU_LEARN = 1 TCU learning via CAN activated

Status of Battery Sensor type (only for ETC versions):

CONF_BAT_LIN = C_CONF_BAT_LIN

Remark:

CONF_BAT_LIN = 0 (Battery Sensor sending Signal via Analog Inputs)

CONF_BAT_LIN = 1 (Communication with SEMI-Smart Battery Sensor via LIN BUS)

CONF_BAT_LIN = 2 (Communication with Smart Battery Sensor via LIN BUS)

Hint: As the Temperature Input of Analog BEM Sensor is same as OIL Temperature Sensor it is not possible to choose CONF_TOIL_MDL = 0 and CONF_BAT_LIN = 0, If this configuration is choosen, then CONF_BAT is set to = 0

Status of Battery Management (only for ETC versions):

If (CONF_TOIL_MDL = 0 and CONF_BAT_LIN = 0)

Same input PIN is used for TOIL-Sensor and analog TBAT-Sensor

Then CONF_BAT = 0

Else CONF_BAT = C_CONF_BAT

Remark:

CONF_BAT = 0 (no Battery Management)

CONF_BAT = 1 (Battery Management)

Status of Passenger Compartment digital input (only for ETC versions):

IF CONF_BAT = 0

Then CONF_PC_FAN = 0

Else CONF_PC_FAN = C_CONF_PC_FAN

Endif

C_CONF_PC_FAN = 0 (No Passenger Comp. Blower switch Input)

C_CONF_PC_FAN = 1 (Passenger Comp. Blower switch information via Analog Input)

C_CONF_PC_FAN = 2 (Passenger Comp. Blower switch information via CAN message)

Status of Engine torque limitation(slow path) from TCU

CONF_ETL_TCU = C_CONF_ETL_TCU

C_CONF_ETL_TCU = 0 (No slow intervention)

C_CONF_ETL_TCU = 1 (Intervention with TQI_GS_SLOW_REQ)

C_CONF_ETL_TCU = 2 (Proportion of fast signal TQI_GS_REQ is used)

1.40.2 Version Learning

Update rate : 1000ms

1.40.2.1 Gearbox Type

The presence of an automatic gearbox is learnt depending if there is a signal from the TCU_TYPE from TCU on the CAN-bus. With C_CONF_TCU_LEARN = 0 also a learnt TCU can be reset from CONF_TCU = 00H or 0AH to CONF_TCU = 0FH.

The learning strategy includes the following items:
- A virgin ECU is configured to MT.
- If the configuration of the ECU is MT then the ECU monitors the CAN concerning the related messages during 10s after key on.
- At a learnt ECU the normal CAN-timeout-diagnosis is active.
- It is possible to set back an already learnt TCU by diagnosis tool.

Remark:
It is necessary to have a different calibration of some maps depending on AT- or MT-gearbox-type. Due to the setting of the gearbox-type some maps are existing double. The maps for the AT are signed with '_AT' at the end of the map. If CONF_TCU = 0Fh (MT) then the map without '_AT' is choosen and vice versa. In the specification a double existing map is listed once but marked with (_AT) at the end of the map.

At the moment separated maps for CVT are not available.

```
If      LV_TEACH_CAN_FINISHED = 0
Then If     CONF_TCU_LEARN = 1
     Then If     CONF_TCU = 0Fh (MT/initial value)
          Then If     LV_TCU1_CAN_VLD = 1
               Then             If     TCU_TYPE = 01h
                    Then CONF_TCU = 00h        Automatic transmission (Step shift)
                    Else If     TCU_TYPE = 02h
                         Then CONF_TCU =  0Ah    Automatic transmission (CVT)
                         Else CONF_TCU = 0Fh (MT/initial value)Manual transmission
                         Endif
                    Endif
               Endif (no valid TCU1 message on CAN)
          Endif  (already learnt)
     Else   (no TCU learning)
            CONF_TCU = 0Fh (MT/initial value)Manual transmission
            LV_AT = 0
     Endif
     If     T_TEACH_CONF < 10
     Then   Increment T_TEACH_CONF by 1
     Else   LV_TEACH_CAN_FINISHED = 1
Else (no more teaching after end of teach time)
Endif
```

If CONF_TCU = 00h or CONF_TCU = 0Ah

then LV_AT = 1

else LV_AT = 0

Restart TCU learning:

TCU learning can be restartet using KWP2000 Service IOCtlByLocalId. After this Service call the following actions will be done:

CONF_TCU = 0Fh (MT/ initial value)

LV_AT = 0

LV_TEACH_CAN_FINISHED = 0

T_TEACH_CONF = 0

1.40.2.2 Traction Control system

The presence of a TCS is learnt depending if there is a signal from the TCS on the CAN-bus.

The learning strategy includes the following items:

- A virgin ECU is configured to "no TCS" (CONF_TCS = 0).
- If the configuration of the ECU is "no TCS" (CONF_TCS = 0) then the ECU monitors the CAN concerning the related messages during 10s after key on.
- It is possible to set back an already learnt TCS by diagnosis tool.

If LV_TEACH_CAN_FINISHED = 0

Then If CONF_TCS = 0

  Then If LV_TCS1_CAN_VLD = 1

    Then CONF_TCS = 1

    Endif

  Endif

Else no more teaching after end of teach time

Endif

Restart TCS learning:

TCS learning can be restartet using KWP2000 Service IOCtlByLocalId. After this Service call the following actions will be done:

CONF_TCS = 0

LV_TEACH_CAN_FINISHED = 0

T_TEACH_CONF = 0

1.40.2.3 Acquisition of the ambient air temperature

For turbo charged engines there are two possibilities to recieve TAM in formation: From the CAN-bus or from an analog sensor value.

The learning strategy for turbo charged engines (LC_TCHA_CONF = 1) includes the following items:

- A virgin ECU is configured to CONF_TAM = 0 (no sensor value available).
- The ECU monitors the CAN concerning the related messages during a certain time (C_T_TAM_LEARN) after reset. If the respective CAN messages were received CONF_TAM is set to 1 (TAM via CAN available).
- If the respective CAN messages were not received it is assumed that an analog signal is available and CONF_TAM is set to 2.
- It is possible to set back an already learnt TAM signal by diagnosis tool.

For non charged engines (LC_TCHA_CONF = 0) CONF_TAM is always set to 0.

CONF_TAM = 0: no TAM sensor available or not yet learnt
CONF_TAM = 1: TAM via CAN (only for LC_TCHA_CONF = 1)
CONF_TAM = 2: TAM via analog sensor signal (only for LC_TCHA_CONF = 1)

Initialization: - at first power up or non-volatile memory lost:
CONF_TAM = 0
- otherwise: restored from non-volatile memory
- at reset: T_TAM_LEARN = C_T_TAM_LEARN

Recurrence: 1000ms

Formula section:

```
If    CONF_TAM = 0
      And LC_TCHA_CONF = 1
      And LV_ERR_CAN_BUS_OFF = 0
Then        If     T_TAM_LEARN <> 0
            Then   T_TAM_LEARN = T_TAM_LEARN – 1
                   If     LV_TAM_CAN_FIRST_VLD = 1
                   Then   CONF_TAM = 1
                          T_TAM_LEARN = 0
                   Endif
            Else   CONF_TAM = 2
            Endif
Else        If     LC_TCHA_CONF = 0
            Then   CONF_TAM = 0
            Endif
Endif
```

1.40.2.4 Air Condition Control

The presence of an air condition control is learnt depending if there is once the signal for "air condition control selected" recognized. In order to prevent signal disturbance being interpreted as ACIN, the LV_IM_ACIN has to be present during 3 following recurrences.

<u>Application conditions:</u>

Initialisation: at reset:

CONF_ACC: restored from NVMY; 0 in case of 1st power up or NVMY lost

CTR_ACC_LEARN = 0

Recurrence: *1000 ms*

ACTION_INFR_GetLvImAcin(OUT < *lv_im_acin* >***)***

If CONF_ACC = 0 *(no air condition control present)*

then **if** LV_IM_ACIN = 1 *(ACIN selected by dashboard switch)*

or LV_IM_ACIN_CAN = 1 *(ACIN selected by CAN input)_* **then**

$CTR\_ACC\_LEARN_{(n)} = CTR\_ACC\_LEARN_{(n-1)} + 1$

if CTR_ACC_LEARN = 3 *(ACIN signal present during 3 sec)*

then CONF_ACC = 1 *(air condition control present /learnt)*

endif

else CTR_ACC_LEARN = 0 *(reset ACIN learning counter)*

endif

1.40.2.5 Vehicle Speed Input

For non-OBD II-variants the vehicle speed input is coming only from the gearbox. In this case CONF_VS = 0, as the variant-coding is C_CONF_VS = 0 (see chapter “Variant coding”).
For OBD II-variants (C_CONF_VS = 1) it is possible to obtain the input for the vehicle speed from an inductive wheel speed sensor or from ABS. The ECU is able to distinguish the two inputs by comparing the two different signals and to <u>learn</u> the actual configuration.

<u>Learning strategy:</u>

The learning is performed only once for virgin ECU and C_CONF_VS = 1 and CONF_VS ≠ 1 or 2 hex. During learning CONF_VS is 3, that means the learning is active and the result of the learning is not known yet.

One of the configurations (ABS or inductive wheel speed sensor) is considered as learned, if during C_T_VS_MIN_LEARN seconds VS_ABS or VS_WHEEL_INDU is greater than C_VS_MIN_LEARN and the VS-change is smaller than C_VS_DIF_MAX_LEARN. If the conditions are fullfilled for both inputs at the same time the ABS-variant is learned (as the signal coming from ABS is considered more stable):

If LV_CDN_VB_MIN_DIAG = 1

then a timer is started for C_T_VS_MIN_LEARN seconds

If (VS_WHEEL_INDU > C_VS_MIN_LEARN

and $|VS\_WHEEL\_INDU_n - VS\_WHEEL\_INDU_{n-1}| < C\_VS\_DIF\_MAX\_LEARN)$

then CONF_VS = 2 (Vehicle speed signal acquisition from inductive sensor)

If (VS_ABS > C_VS_MIN_LEARN

and $|VS\_ABS_n - VS\_ABS_{n-1}| < C\_VS\_DIF\_MAX\_LEARN)$

then CONF_VS = 1 (Vehicle speed signal acquisition from ABS)

Vehicle speed during learning:

During learning (CONF_VS = 3) the vehicle speed is determined as follows :

a). if VS_ABS = 0 and VS_WHEEL_INDU = 0 then VS = 0

b). if VS_ABS > 0 and VS_WHEEL_INDU = 0 then VS = VS_ABS

c). if VS_ABS = 0 and VS_WHEEL_INDU > 0 then VS = VS_WHEEL_INDU

d). if VS_ABS > 0 and VS_WHEEL_INDU > 0 then VS = VS_ABS
(as the signal coming from ABS-input is considered more stable)

Remark:

The learned configuration is stored in non-volatile memory.
It is possible to change the variant by SAM 2000 and therefore a new learning is performed after next engine start. This happens also after an erasement of the learned configuration via KW 2000 (SAM-S or service tester) at the end of line-programming or at the service.

1.40.2.6 Acquisition of the aircon control unit configuration

The presence of a external control unit is learnt depending if there is a signal from the FATC on the CAN-bus.

The learning strategy includes the following items:
- A virgin ECU is configured to CONF_ACC_FATC = 0
- The ECU monitors the CAN concerning the related messages during a certain time (C_T_ACC_FATC_LEARN) after reset. If the respective CAN messages were received CONF_ACC_FATC is set to 1 (ACC message via CAN available).
- If the respective CAN messages were not received it is assumed that no presence of FATC
- It is possible to set back an already learnt FATC configuration by diagnosis tool.

Initialization: - at first power up or non-volatile memory lost:
CONF_ACC_FATC = 0
- otherwise: restored from non-volatile memory
- at reset: T_ACC_FATC_LEARN = C_T_ACC_FATC_LEARN

Recurrence: 1000ms

Formula section:

```
If      CONF_ACC_FATC = 0
    And NC_ETC_CONF = 1
    And LV_ERR_CAN_BUS_OFF = 0
Then    If      T_ACC_FATC_LEARN ≠ 0
        Then    T_ACC_FATC_LEARN = T_ACC_FATC_LEARN – 1
                If      LV_ACC_FATC_CAN_FIRST_VLD = 1
                        Or      LC_ACC_FATC = 1
                Then    CONF_ACC_FATC = 1
                        T_ACC_FATC_LEARN = 0
                Endif
        Endif
Endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CONF_LAM	1	0...1H	0...1	1	-
Configuration byte for oxygen sensor configuration					
C_CONF_CAT_EFF	1	0...1H	0...1	1	-
Configuration byte for catalyst efficiency diagnosis					
C_CONF_CP	1	0...2H	0...2	1	-
Configuration byte for evaporative system					
C_CONF_DIAGCP	1	0...2H	0...2	1	-
Configuration byte for evaporative system diagnosis					
C_CONF_DIAGCP_VOL	1	0...2H	0...2	1	-
Configuration byte for fuel tank level acquisition for canister purge diagnosis					
C_CONF_MIL_FMY	1	0...5H	0...5	1	-
Configuration byte for malfunction indicator light and failure memory management					
C_CONF_VS	1	0...2H	0...2	1	-
Configuration byte for vehicle speed acquisition					
C_CONF_IMOB	1	0...1H	0...1	1	-
Configuration byte for immobilizer					
C_CONF_CFA	1	0...1H	0...1	1	-
Configuration byte for selection of fan control method					
C_CONF_ACP	1	0...1H	0...1	1	-
Configuration byte for AC pressure signal					
C_CONF_PSTE	1	0...1H	0...1	1	-
Configuration byte for Power steering pressure signal					
C_CONF_TOIL_MDL	1	0...2H	0...2	1	-
Configuration byte for TOIL acquisition					
C_CONF_TCU_LEARN	1	0...1H	0...1	1	-
Configuration byte for TCU type set Function via CAN					
C_VS_MIN_LEARN	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for vehicle speed input learning					
C_VS_DIF_MAX_LEARN	1	0...FFH	0...255	1	km/h
Maximum vehicle speed difference for vehicle speed input learning					
C_T_VS_MIN_LEARN	1	0...FFH	0...255	1	sec
Minimum time duration for vehicle speed input learning					
C_CONF_AD_RST	1	0...2H	0...2	1	-
Configuration byte for erasement of adaptative values					
C_CONF_FMY_RST	1	0...1H	0...1	1	-
Configuration byte for erasement of failure memory					
C_CONF_GEN_LOAD	1	0...1H	0...1	1	-
Cofiguration byte for GEN_LOAD					
C_CONF_VIM	1	0...2H	0...2	1	-
Configuration byte for variable intake manifold					
C_CONF_ASR_SLOW	1	0...1H	0...1	1	-
Configuration byte for availability of Slow Path Torque Reduction Request from TCS Control Unit (only available for ETC Configuration)					
C_CONF_KOBD	1	0...1H	0...1	1	-
Status information for supporting KOBD specific requirement					
C_CONF_CAM_VVT_EX	1	0...1H	0...1	1	-
Configuration byte for exhaust camshaft					
C_CONF_PORT	1	0...1H	0...1	1	-
Configuration byte for Port Flap					
C_CONF_MAF	1	0...1H	0...1	1	-
Configuration byte for Load Sensor					
C_T_TAM_LEARN	1	0...FFH	0...255	1	s
Duration for TAM learning					
C_CONF_YEAR	1	30...53H	2000...2035	1	-
Model Year information for automatic VIN coding in development ECU					

C_CONF_BAT	1	0...1H	0...1	1	-
Configuration byte for Battery Management					
C_CONF_BAT_LIN	1	0...2H	0...2	1	-
Configuration byte for Battery Management Sensor type					
C_CONF_PC_FAN	1	0... 2H	0...2	1	-
Configuration byte for Passenger Compartment Fan information					
C_CONF_ETL_TCU	1	0... 2H	0...2	1	-
Configuration byte for TCU supporting engine torque limitation (slow path)					
C_T_ACC_FATC_LEARN	1	0...FFH	0...255	1	s
Duration for ACC_FATC learning					
LC_ACC_FATC	1	0...1H	0...1	1	-
Logical byte for ACC_FATC presence					

1.41 Adjustments via serial communication line

1.41.1 Idle speed CO-trim

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_CO_IS	V/S/O	8000...7FFFH	-694.510597 ...694.489403	0,021194	Mg/stk
MFF correction for idle speed CO-adjustment					

Input data:

LV_IS	VS	TCO	LV_ERR_MAF
LV_ERR_VS	LV_ERR_TCO	LV_ERR_TPS	LV_ERR_ISA_i
LV_ERR_IV[NC_CYL_NR]	LV_ERR_MAP		

FUNCTION DESCRIPTION:

General information:

For the idle speed CO-trim, the diagnostic tester is using the KWP2000-protocol service Input Output Control By Local Identifier (IOCBLI). With its Input Output Control Parameter Short Term Adjustment (STA), the CO-trim is adjusted stepwise, that means each time the button “up” or “down” is pressed on the tester, the CO-trim value MFF_CO_IS is incremented or decremented by one step. If the CO-trim is correct, the value is stored if the service Long Term Adjustment (LTA) is performed. Temporary changes can be reset by the service Return Control To ECU (RCTECU).

1.41.1.1 Temporary change of MFF_CO_IS

The CO-trim value MFF_CO_IS is changed temporarily using the KWP2000 service IOCBLI-STA. If ignition is switched off, the changes are reset.

Application conditions:

The Adjustment can be performed if the following conditions are fullfilled:

If LV_IS = 1
and VS = 0
and TCO > C_TCO_MIN_CO_IS
and LV_ERR_TCO = 0
and LV_ERR_TPS = 0
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
and LV_ERR_ISA_i = 0
and LV_ERR_IV[NC_CYL_NR] = 0
and LV_ERR_VS = 0

If this service is performed, MFF_CO_IS is incremented or decremented by one step, depending on the control option byte of this service.
The condition LV_ERR_ISA_i is only for ISA system.

1.41.1.2 Long term storage of MFF_CO_IS

The CO-trim value MFF_CO_IS is stored in the flash memory using the KWP2000 service IOCBLI-LTA.

1.41.1.3 Reset changes of MFF_CO_IS to previous MFF_CO_IS

The CO-trim value MFF_CO_IS is reset to the MFF_CO_IS value which was permanently stored in the ECU, using the KWP2000 service IOCBLI-RCTECU.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_CO_IS	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature threshold for idle speed CO-adjustment.					

1.41.2 Nominal idle speed adjustment

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_SP_IS_ADJ_ASA	V/S	0...FFH	0...255	1	Rpm
Tuning correction for nominal idle speed					

Input data:

LV_IS	TCO	N_SP_IS	

FUNCTION DESCRIPTION:

Application conditions:

The Adjustment can be performed

If LV_IS = 1
and TCO > C_TCO_MIN_N_ASA

1.41.2.1 Idle speed-trim via Diagnostic tester

For the idle speed-trim, the diagnostic tester is using the KWP2000-protocol service Input Output Control By Local Identifier (IOCBLI). With its Input Output Control Parameter Short Term Adjustment (STA), the idle speed-trim is adjusted stepwise, that means each time the button “up” or “down” is pressed on the tester, the idle speed-trim value is incremented or decremented by one step. If the idle speed is correct, the value N_SP_IS_ADJ_ASA is stored if the service Long Term Adjustment (LTA) is performed. Temporary changes can be reset by the service Return Control To ECU (RCTECU).

1.41.2.2 Temporary change of N_SP_IS_ADJ_ASA

The idle speed-trim value N_SP_IS_ADJ_ASA is changed temporarily using the KWP2000 service IOCBLI-STA. If ignition is switched off, the changes are reset.

Application conditions:

LV_IS= 1

If this service is performed, N_SP_IS_ADJ_ASA is incremented or decremented with NC_N_SP_IS_ADJ_ASA by one step, depending on the direction of the correction (+/-) of this service but it is possible only to higher nominal idle speed values. A decrementation is possible down to the original uncorrected N_SP_IS.

1.41.2.3 Long term storage of N_SP_IS_ADJ_ASA

The idle speed-trim value N_SP_IS_ADJ_ASA is stored in the flash memory using the KWP2000 service IOCBLI-LTA.

1.41.2.4 Reset changes of N_SP_IS_ADJ_ASA to previous N_SP_IS_ADJ_ASA

The idle speed value N_SP_IS_ADJ_ASA is reset to the N_SP_IS_ADJ_ASA value which was permanently stored in the ECU, using the KWP2000 service IOCBLI-RCTECU.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_N_ASA	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature threshold for nominal idle speed correction.					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_SP_IS_ADJ_ASA	1	0...FFH	0...255	1	Rpm
Tuning correction for nominal idle speed - Non adjustable					

Applicative values:

NC_N_SP_IS_ADJ_ASA = 1

1.42 Actuator test

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_EXT_ADJ_MIL	V/O	0...1H	0...1	1	-
Flag to activate MIL Actuator Test					
LV_ACT_EXT_ADJ_MIL	V/O	0...1H	0...1	1	-
Adjustment value for MIL					
LV_EFP_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Fuel Pump Actuator Test					
LV_ACT_EFP_EXT_ADJ	V/O	0...1H	0...1	1	-
Adjustment value for Fuel Pump					
LV_ACCOUT_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate A/C Compressor Relay Actuator Test					
LV_ACT_ACCOUT_EXT_ADJ	V/O	0...1H	0...1	1	-
Adjustment value for A/C Compressor Relay					
LV_IMOB_MIL_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Immobilizer Lamp Actuator Test					
LV_ACT_IMOB_MIL_EXT_ADJ	V/O	0...1H	0...1	1	-
Adjustment value for Immobilizer Lamp					
LV_RLY_MAIN_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Main Relay Actuator Test					
LV_ACT_RLY_MAIN_EXT_ADJ	V/O	0...1H	0...1	1	-
Adjustment value for Main Relay					
LV_SOV_ACT_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate SOV Actuator Test					
LV_SOV_EXT_ADJ	V/O	0...1H	0...1	1	-
Adjustment value for SOV					
LV_CFA_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Cooling and Condenser Fans Actuator Test					
STATE_CFA_EXT_ADJ	V/O	0...3H	0...3	1	-
Adjustment value for Cooling and Condenser Fans state (0,1,2,3)					
CFAPWM_EXT_ADJ	V/O	0...FFH	0...99.61	0.39	%
Adjustment value Electric fan control pulse width modulation					
LV_CPPWM_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Canister Purge Valve Actuator Test					
CPPWM_EXT_ADJ	V/O	0...FFH	0...99.61	0.39	%
Adjustment value for Canister Purge Valve					
STATE_ACR_TEST_IVVT	V/O	0H 1H 2H 3H 4H	Passive IN_1 IN_2 EX_1 EX_2	-	-
State for IVVT Actuator Test					
INH_IV_EXT_ADJ	V/O	0...FFH	0...255	1	-
Inhibition of injection valve for Actuator Test					
LV_TPS_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate TPS Actuator Test					
TPS_EXT_ADJ	V/O	0...FFH	0...119.5	0.0073	°TPS
Adjustment value for TPS					
LSHPWM_EXT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFH	0...99.61	0.39	%
Adjustment value Downstream Sensor Heater					
LV_LSHPWM_DOWN_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate downstream O2-Sensor heater Actuator Test					
LSHPWM_EXT_LS_UP[NC_CBK_EX_NR]	V/O	0...FFH	0...99.61	0.39	%
Adjustment value Upstream Sensor Heater					
LV_LSHPWM_UP_EXT_ADJ	V/O	0...1H	0...1	1	-

Flag to activate upstream O2-Sensor heater Actuator Test					
LV_IGC_x_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Ignition Actuator Test (x = 0 to NC_CYL_NR-1)					
IGC_x_EXT_ADJ	V/O	0...1H	0...1	1	-
Boolean to trigger Ignition pulse during Actuator Test (x = 0 to NC_CYL_NR-1)					
LV_OPG_SP_ISA_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate Idle Speed Actuator Test					
OPG_SP_ISA_EXT_ADJ	V/O	0...FFH	0...99.61	0.39	%
Adjustment value for Idle Speed Actuator					
LV_EXT_ADJ_VIM	V/O	0...1H	0...1	1	-
Flag to activate the Variable Intake Manifold Switch					
LV_ACT_EXT_ADJ_VIM	V/O	0...1H	0...1	1	-
Adjustment value for Variable Intake Manifold Switch					
LV_EXT_ADJ_ACC_MAIN_LAMP	V/O	0...1H	0...1	1	-
Flag to activated ACC_MAIN_Lamp Actuator Test					
LV_ACT_EXT_ADJ_ACC_MAIN_LAMP	V/O	0...1H	0...1	1	-
Adjustment value for ACC_MAIN LAMP Actuator Test					
LV_EXT_ADJ_ACC_SET_LAMP	V/O	0...1H	0...1	1	-
Flag to activated ACC_SET_Lamp Actuator Test					
LV_ACT_EXT_ADJ_ACC_SET_LAMP	V/O	0...1H	0...1	1	-
Adjustment value for ACC_SET LAMP Actuator Test					
LV_PORT_SP_EXT_ADJ	V/0	0...1H	0...1	1	-
Flag to activate PORT_FLAP Actuator Test					
PORT_SP_EXT_ADJ	V/O	0...1H	0..1	1	-
Setpoint for Port Flap request by Service Tool (0 = close, 1= open)					
LV_PWM_WG_EXT_ADJ	V/0	0...1H	0...1	1	-
Flag to activate Wastegate Actuator Test					
PWM_WG_EXT_ADJ	V/O	0...FFFFH	0...99.9984	1.5258e-3	%
PWM for Wastegate requested by Service Tool					
LV_RCL_EXT_ADJ	V/O	0...1H	0...1	1	-
Flag to activate the Recirculation Valve Actuator test					
LV_ACT_RCL_EXT_ADJ	V/O	0...1H	0...1	1	-
Recirculation valve external Adjustment request by Service Tool					

Input data:

LV_ST_END	LV_IGK	LV_ES	LV_IS
ECU_STATE			

General information:

The test of the actuators is divided into several groups with different conditions. The adjustments are done via KWP2000 protocol by the service InputOutputControlByLocalIdentifier. For more detailed informations refer to KWP2000-specification. During test with engine stop, starting of the engine is possible only after ignition key off (ECU_STATE = ENG_LOCK)

1.42.1 Engine Lock

As long as any actuator test is activated the engine is locked (ECU_STATE = ENG_LOCK) with following exceptions (Actuator test allowed at LV_IS = 1)

- Injectors (INH_IV_x)
- O2 Heater Front (LSH_UP)
- O2 Heater Rear (LSH_DOWN)
- Variable Intake Manifold Switch (VIM)
- Recirculation valve (RCL)

1.42.2 On/Off Outputs

1.42.2.1 Actuator test is performed only with stopped engine

Conditions for activation: LV_IGK = 1
and LV_ES = 1

The component is switched on and off for each 1 sec

- Malfunction indicator lamp (MIL): on/off each 1 sec
- Fuel pump relay (EFP): on/off each 1 sec
- A/C compressor relay (ACC): on/off each 1 sec
- Imobilizer malfunction indicator light (IMOB_MIL) on/off each 1 sec
- Shut off valve (SOV) open/close each 1 sec
- Main relay (MAIN): on/off each 1 sec

(all other tests are switched off before testing main relay)

only for ETC version:

- Adaptive Cruise Control Main Lamp on/off each 1 sec
- Adaptive Cruise Control Set Lamp on/off each 1 sec

The component is switched on and off for each 5 sec

- CFA relay low speed (RLY_FAN_L) on/off each 5 sec
- CFA relay high speed (RLY_FAN_H) on/off each 5 sec

The component is permanently switched on/off by request

- Shut off valve (SOV): action 00H permanently open
  action FFH permanently closed

1.42.2.2 Actuator test activation at idle or stopped engine

Conditions for Activation: LV_IGK = 1
and LV_IS = 1 **or** LV_ES = 1

The component is switched on and off for each 1 sec

- Variable Intake Manifold Switch (VIM) on/off each 1 sec

only for TCI version:

- Recirculation Valve (RCL) on/off each 1 sec

1.42.3 PWM outputs

1.42.3.1 Actuator test activation only at stopped engine

Conditions for activation: LV_IGK = 1
and LV_ES = 1

- Oil control valve (OCV) Intake side 0%/100% each 0,5 sec
  If IOL-ID 24h is send from service tool,
  then On/Off definition On: STATE_ACR_TEST_IVVT = 1h
  Off: STATE _ACR_TEST_IVVT = 0h

- Oil control valve (OCV) Exhaust side 0%/100% each 0,5 sec
  If IOL-ID 80h is send from service tool,
  then On/Off definition On: STATE_ACR_TEST_IVVT = 3h
  Off: STATE _ACR_TEST_IVVT = 0h

- Canister purge solenoid (CPS) : on/off each 1 sec
  On/Off definition : On: CPPWM_EXT_ADJ = C_MAX_PWM_CPS
  Off: CPPWM_EXT_ADJ = C_MIN_PWM_CPS

- Port Flap (VCM): on/off each 1 sec
  On/Off definition : On: PORT_SP_EXT_ADJ = 1 (close)
  Off: PORT_SP_EXT_ADJ = 0 (open)

- Cooling Fan (CFA) on/off each 1 sec
  On/Off definition: On: CFA_PWM_EXT_ADJ = C_MAX_PWM_CFA
  Off: CFA_PWM_EXT_ADJ = C_MIN_PWM_CFA

- ETC Motor (MTC) (**only for ETC version**) on/off each 1 sec
  On/Off definition : On: TPS_EXT_ADJ = C_MAX_ACR_MTC
  Off: TPS_EXT_ADJ = C_MIN_ACR_MTC

- Idle speed actuator (ISA): (**only for ISA version**) on/off each 1 sec
  On/Off definition : On: OPG_ISA_EXT_ADJ = C_MAX_PWM_ISA
  Off: OPG_ISA_EXT_ADJ = C_MIN_PWM_ISA

- Wastegate (WG) (**only for TCI version**) on/off each 1sec
  On/Off definition : On: PWM_WG_EXT_ADJ = C_MAX_PWM_WG
  Off: PWM_WG_EXT_ADJ = C_MIN_PWM_WG

1.42.3.2 Actuator test activation only after Start

Conditions for activation: LV_IGK= 1
and LV_ST_END = 1

- O2 Heater Front (LSH_UP) on/off each 1 sec
  On/Off definition:
  On: LSHPWM_EXT_LS_UP [NC_CBK_EX_NR] = C_MAX_PWM_LSH_UP
  Off: LSHPWM_EXT_LS_UP [NC_CBK_EX_NR] = C_MIN_PWM_LSH_UP

- O2 Heater Rear (LSH_DOWN) on/off each 1 sec
  On/Off definition:
  On: LSHPWM_EXT_LS_DOWN [NC_CBK_EX_NR] = C_MAX_PWM_LSH_DOWN
  Off: LSHPWM_EXT_LS_DOWN [NC_CBK_EX_NR] = C_MIN_PWM_LSH_DOWN

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MIN_PWM_CPS	1	0...FFH	0...99,609	0,391	%
CPS low value (= OFF state) for Actuator test					
C_MAX_PWM_CPS	1	0...FFH	0...99,609	0,391	%
CPS high value (= ON state) of CPS for Actuator test					
C_MIN_PWM_CFA	1	0...FFH	0...99.609	0.391	%
CFA low value (= OFF state) for Actuator test					
C_MAX_PWM_CFA	1	0...FFH	0...99.609	0.391	%
CFA high value (= ON state) for Actuator test					
C_MIN_PWM_LSH_UP	1	0...FFH	0...99.609	0.391	%
LSH_UP low value (= OFF state) for Actuator test					
C_MAX_PWM_LSH_UP	1	0...FFH	0...99.609	0.391	%
LSH_UP high value (= ON state) for Actuator test					
C_MIN_PWM_LSH_DOWN	1	0...FFH	0...99.609	0.391	%
LSH_DOWN low value (= OFF state) for Actuator test					
C_MAX_PWM_LSH_DOWN	1	0...FFH	0...99.609	0.391	%
LSH_DOWN high value (= ON state) for Actuator test					
C_MIN_ACR_MTC	1	0...FFH	0...119.5	0.4686	°
ETC Position low value (= OFF state) for Actuator test					
C_MAX_ACR_MTC	1	0...FFH	0...119.5	0.4686	°
ETC Position high value (= ON state) for Actuator test					
C_MIN_PWM_ISA	1	0...FFH	0...99.609	0.391	%
ISA low value (= OFF state) for Actuator test					
C_MAX_PWM_ISA	1	0...FFH	0...99.609	0.391	%
ISA high value (= ON state) for Actuator test					
C_MIN_PWM_WG	1	0...FFFFH	0...99.9984	1.5258e-3	%
Wastegate low value (= OFF state) for Actuator test					
C_MAX_PWM_WG	1	0...FFFFH	0...99.9984	1.5258e-3	%
Wastegate high value (=ON state) for Actuator test					

1.42.4 Injectors

Conditions for activation: LV_IGK = 1

and LV_IS = 1

Each several injector can be selected for a test. After activation of the test, the corresponding injector is disabled, until the test is stopped.

- **If** IOL 39h is recieved from service tool **then** switch On/OFF log. Injector 0 (phys: IVCyl 1)
- **If** IOL 3Ah **then** log Injector 3 (phys: IV Cyl 2)
- **If** IOL 3Bh **then** log.Injector 1 (phys: IV Cyl 3)
- **If** IOL 3Ch **then** log.Injector 2 (phys: IV Cyl 4)

INH_IV_EXT_ADJ = x x x x 0 0 1 0 BIN

- logical cylinder 0 enabled
- logical cylinder 1 disabled
- logical cylinder 2 disabled
- logical cylinder 3 disabled

1.42.5 Ignition

Each ignition coil can be separately selected for Actuator Test. When Actuator Test is activated, a spark is generated with normal Dwell Time every second.

Application conditions :

Initialization : LV_IGC_x_EXT_ADJ = IGC_x_EXT_ADJ = 0

Activation : LV_IGK = 1

and LV_ES = 1

Deactivation : LV_IGK = 0

or LV_ES = 0

If conditions for activation are fullfilled and Ignition Actuator test is activated via KW2000, then LV_IGC_x_EXT_ADJ is set to 1 and IGC_x_EXT_ADJ is toggling with 1 Hz frequency and 50% PWM ratio (trigger point to generate ignition pulse), else both LV_IGC_x_EXT_ADJ and IGC_x_EXT_ADJ are set to 0.

If IOL 31h is send by service tool, **then** activate test for IGC_0_EXT_ADJ (phys: IGC Cyl 1)

If IOL 32h **then** IGC_3_EXT_ADJ (phys: IGC Cyl 2)

If IOL 33h **then** IGC_1_EXT_ADJ (phys: IGC Cyl 3)

If IOL 34h **then** IGC_2_EXT_ADJ (phys: IGC Cyl 4)

1.43 Initialization of variables from non implemented functions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAM_AV_MV_IN	O	8000...7FFFH	-768...767.976	0.0234375	[°CRK]
average value inlet-CAM position					
ENG_STATE	O/V	0H 1H 2H 3H 4H 5H	"ES" "ST" "IS" "PL" "PU" "PUC"	0	[-]
Engine operating state					
ERR_SYM_ISA	O	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	[-]
Dummy: Detected error symptom ISA diagnosis					
ERR_SYM_ISA_1	O	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	[-]
Dummy: Detected error symptom ISA_1 diagnosis					
ERR_SYM_ISA_2	O	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	[-]
Dummy: Detected error symptom ISA_2 diagnosis					
ERR_SYM_PUT_PLAUS_ES	O	0H 8H	NO_SYM SYM_3	1	[-]
Detected symptom pressure up throttle plausibility diagnosis while standing vehicle					
ERR_SYM_PUT_PLAUS_FL	O	0H 8H	NO_SYM SYM_3	1	[-]
Detected symptom pressure up throttle plausibility diagnosis in full load					
FAC_PORT_DEAC_MV	O	0...FFFFH	0...0.99998	0.0153e-3	[-]
Average of FAC_PORT_DEAC[NC_PORT_NR]					
FAC_TQ_REQ_VSL	O	0...FFFFH	0...1.99999	0.0000305	[%]
IGA_EOLP	O	0...FFH	-36.625...60	0.375	[°CRK]
IGA_WOUT_KNK	O	0...FFH	-35.625...60	0.375	[°CRK]
actual ignition angle (incl. application corrections without knock interception)					
ISAPWM	O/V	0...FFFFH	0...99.998	0.001526	[%]
Old ISA actuator setpoint, now replaced by OPG_AV_ISA.					
LAM_ADJ_i	O	8000...7FFFH	-50...49.9992	0.0015	[%]
External lambda control offset					
LAMB_i	O	0...7FFFH	0...31.99902	0.00098	[-]
Actual Lambda					
LAMB_RGN[NC_CBK_EX_NR]	O	0...7FFFH	0...31.99902	0.00098	[-]
lambda setpoint for catalyst regeneration					
LAMB_SA_CH	O/V	0...7FFFH	0...31.99902	0.00098	[-]
Lambda for secondary air catalyst heating					
LAMB_SO2P	O	0...7FFFH	0...31.99	0.000977	[-]
Lambda setpoint for desulfation					
LAMB_SP_DIAG_LS_UP_DOWN	O	0...7FFFH	0...31.99902	0.00098	[-]
Lambda setpoint request from now deleted CHK_LS_DOWN diagnosis.					
LAMB_SP_REQ	O	0...7FFFH	0...31.99902	0.00098	[-]
Lambda setpoint request during combustion mode change in homogeneous					
LV_ACT_CRU	O	0...1H	0...1	1	[-]

Intervention of cruise control					
LV_ACT_VSL	O	0...1H	0...1	1	[-]
LV_CAM_POS_DIAG_EXT_REQ	O	0...1H	0...1	1	[-]
Logical bit indicating the activation of Camshaft offset adaptation through short trip test					
LV_CLU_OPEN	O	0...1H	0...1	1	[-]
Clutch Switch OPEN information					
LV_CLU_SWI	O	0...1H	0...1	1	[-]
Clutch switch information					
LV_CP_RAMP_OPEN_ACT	O	0...1H	0...1	1	[-]
Logical variable CP ramp open active					
LV_EFP	O	0...1H	0...1	1	[-]
bit for relay electrical fuel pump activated					
LV_ENA_PRS_CHA_UP	O	0...1H	0...1	1	[-]
Enable flag for pressure sensor upstream charger					
LV_ENA_PRS_WG_ACR	O	0...1H	0...1	1	[-]
Enable flag for wastegate actuator pressure sensor					
LV_ENA_PSN_RCL	O	0...1H	0...1	1	[-]
Enable flag for recirculation actutator position sensor					
LV_END_PSN_ACT_DIAG_IVVT_EX_i	O	0...1H	0...1	1	[]
Flag for mechenical endposition range advanced exhaust camshaft					
LV_END_PSN_ACT_DIAG_IVVT_IN_i	O	0...1H	0...1	1	[]
Flag for mechenical endposition range advanced intake camshaft					
LV_ENG_RUN_DIAG_SAVE	O/V	0...1H	0...1	1	[-]
Flag indicating that the engine has rotated at least once during this driving cycle.					
LV_EOL_OBD	O	0...1H	0...1	1	[-]
LV_EOL_OBD_DC	O	0...1H	0...1	1	[-]
Boolean for end-of-line test active in current driving cycle					
LV_ERR_CAN_ICL_TAM	O	0...1H	0...1	1	[-]
Present error CAN_ICL_TAM					
LV_ERR_EL_CPS	O	0...1H	0...1	1	[-]
Boolean for electrical error currently present on canister purge solenoid					
LV_ERR_ISA_1	O	0...1H	0...1	1	[-]
LV_ERR_ISA_2	O	0...1H	0...1	1	[-]
LV_ERR_LAM_LIM_i	O	0...1H	0...1	1	[-]
LV_ERR_MAF_DET	O	0...1H	0...1	1	[-]
LV_ERR_SA_SAP	O	0...1H	0...1	1	[-]
Present failure: Minimum secondary air flow rate failure after filtering					
LV_ERR_SA_SAV	O	0...1H	0...1	1	[-]
Present failure: Mechanically jammed SAV after filtering					
LV_ERR_SAP	O	0...1H	0...1	1	[-]
Boolean for error currently present on secondary air pump command signal.					
LV_ERR_SAV	O	0...1H	0...1	1	[-]
Boolean for error currently present on secondary air valve command signal.					
LV_ERR_TPS_DET	O	0...1H	0...1	1	[-]
LV_ERR_TQ_CONV_ETCU	O	0...1H	0...1	1	[-]
Error bit for converter torque signal error from TCU					
LV_ERR_V_VB_2_MIN	O	0...1H	0...1	1	[-]
LV_ERR_VCC_MAP	O	0...1H	0...1	1	[-]
Failure state currently present on MAP-sensor supply voltage					
LV_FCUT_TRA	O	0...1H	0...1	1	[-]
Logical variable for authorizing fuel cut-off pattern at transient torque at trailing throttle					

LV_HOM_ACT	O	0...1H	0...1	1	[-]
Homogeneous ignition and injection parameters are applied to the engine					
LV_HOM_AFL_ACT	O	0...1H	0...1	1	[-]
Homogeneous air fuel lean mode active					
LV_HOM_AFS_ACT	O	0...1H	0...1	1	[-]
Logical variable indicates active homogeneous stoichiometric state					
LV_HOM_RUN	O	0...1H	0...1	1	[-]
Flag to activate homogeneous calculations					
LV_IGA_AND_INJ_SWI	O	...H	0...0	1	[-]
Logical control variable for igintion and injection switch					
LV_IGA_EOL_ACT	O	0...1H	0...1	1	[-]
LV_ISC_INH_EXT_ADJ	O	0...1H	0...1	1	[-]
LV_LAM_NOT_STAT_CDN	O	0...1H	0...1	1	[-]
Flag for stop mode activation of lambda controller by wall film					
LV_LAMB_SP_REQ_ACT	O/V	0...1H	0...0	1	[-]
Logical control bit for activating LAMB_SP_REQ					
LV_LAMB_SP_REQ_DIAG_ACT	O/V	0...1H	0...0	1	[-]
Indicates LAMB_SP request from now deleted CHK_LS_DOWN diagnosis.					
LV_LS_DOWN_OBD_2_MAN_DEAC	O	0...1H	0...1	1	[-]
LV_MAX_ERR_FSD_1	O	0...1H	0...1	1	[-]
LV_PC_FAN	O	0...1H	0...1	1	[-]
Status bit for Passenger Compartment Fan					
LV_POW_AC_TRV	O	0...1H	0...1	1	[-]
Status bit for torque intervention from a centrifugal force – RCL input					
LV_PUC_LOCK_TNT	O	0...1H	0...1	1	[-]
LV_PUC_SA_INH	O	0...1H	0...1	1	[-]
LV_PUT_ADD_TCHA_BOOST_REQ	O	0...1H	0...1	1	[-]
Request boost pressure reserve for Turbo Charger boost					
LV_PV_SPT_ACT	O	0...1H	0...1	1	[-]
LV_RGN_NT_REQ	O	0...1H	0...1	1	[-]
LV_RLY_ST	O	0...1H	0...1	1	[-]
LV_RNG_L_REQ	O/V	0...1H	0...1	1	[-]
Activation condition for low-range-mode					
LV_S_ACT	O	0...1H	0...1	1	[-]
Stratified ignition and injection parameters are applied to the engine					
LV_S_CLC	O	0...1H	0...1	1	[-]
Torque intervention due MSR					
LV_S_RUN	O	0...1H	0...1	1	[-]
LV_SAP	O	00...01H	0...1	1	[-]
Activation of secondary air pump					
LV_SAV	O	00...01H	0...1	1	[-]
Activation of secondary air valve					
LV_SAWUP	O/V	0...1H	0...1	0	[-]
Auxiliary function „Secondary air mixture warm-up“					
LV_TPS_AD_ACT_EXT_ADJ	O	0...1H	0...1	1	[-]
LV_TPS_LIH_ACT	O	0...1H	0...1	1	[-]
MAP_SP_PQ_MAX	O/V	0...FFFFH	0...5434	0.0829175	[hPa]

Minimum mainfold air pressure setpoint for unthrottled engine operation for charged engine					
MFF_ADD_REAC	O	0...FFFFH	0...1389	0.02119	[mg/stk]
Reactivation enrichment					
MFF_SP_S_i	O	0...FFFFH	0...1389	0.02119	[mg/tdc]
Fuel mass in stratified combustion.					
NC_USE_SEG_HALF_AVL	O	0...1H	0...1	1	[-]
Compiler switch to indicate, that a half segment trigger is available					
OPM_AV	O	0H 1H 2H 3H 8H	- S AFS AFL LIH	1	[-]
Combustion Mode Actual Mode					
SAF_KGH	O	0...FFFFH	0...1023.984	0.015625	[kg/h]
Corrected value of secondary air mass flow					
STATE_CMB_CTL	O	0...8H	0...8	1	[-]
States of the combustion management					
STATE_RBM_DYN_TIA	O/V	0...7H	0...0	1	[-]
Interface of monitor of TIA Signal Stuck diagnosis for the rate based monitoring statistics for AIRT					
STATE_SA	O	0...8H	0...0	1	[-]
State of Secondary Air Function					
STATE_SAV_DIAG	O	0...1H	0...1	1	[-]
not used – only to satisfy Evap control interface					
TAM_BAS	O	0...03FFH	0...4.9951	0	[V]
Ambient Temperature Sensor Raw Aquisition					
TEG_DYN	O/V	0...7FF0H	0...2047	0.0625	[°C]
Modelled exhaust gas temperature upstream for interfaces					
TIA_DIF_PLAUS_TIA_THD	O	80...7FH	-96...95.25	0.75	[°C]
TIA variation threshold value for TIA plausibility symptom determination					
TQ_ADD_HEAT_ACC	O/V	8000...7FFFH	-1024...1023.97	0.03125	[Nm]
Torque reserve for switching on additional heating by ACC					
TQ_ADD_SO2P	O/V	8000...7FFFH	0...0	0.03125	[Nm]
torque reserve requested for desulfation catalyst heating					
TQ_CONV_ETCU	O	8000...7FFFH	-1024...1023.97	0.03125	[Nm]
Converter torque information from TCU					
VLFT_AV	O	0...FFFFH	0...65.535	0.001	[mm]
Actual value of maximum valve lift					
VP_MAP_MV_MAX_DIAG	O	0...7FFFFH	0...4.9998	1.525E-4	[V]
Threshold value of VP_MAP_MV to detect short circuit in signal wire to VB					
VP_MAP_MV_MIN_DIAG	O	0...7FFFFH	0...4.9998	1.525E-4	[V]
Threshold value of VP_MAP_MV to detect short circuit in signal wire to ground or wire break					
VS_FIL	O	0...FFFFH	0...511.99218	0.0078125	[km/h]
Filtered vehicle speed					
VS_STATE	O/V	0...2H	0...2	1	[-]
Vehicle speed state for cooling fans.					
LV_FUP_LAMP_ON	O	0...1H	0...1	1	[-]
Indicating lamp "ON" state when fuel pressure built-up					

Input data:

CAM_MV_IN	LV_ERR_CPS	LV_CRU_ACT	TEG_DYN_LS_UP[NC_CBK_EX_NR]
LV_ERR_VB	LV_CS	LV_RLY_EFP	LV_ERR_FSD_LAM_LIM_i
N_32	LV_FIRST_VLD_TOOTH	OPG_AV_ISA	LAMB_SP[NC_CBK_EX_NR]
VS_STATE_CFA	STATE_ENG	STATE_RBM_TIA_DYN	LV_ERR_FSD_i
VS	VP_TAM	LV_ERR_VCC_PVS_2	VP_MAP_MAX_DIAG
	VP_MAP_MIN_DIAG		

General information:

This chapter contains data, which are used in generic modules, but not defined in other modules.

Formula section

CAM_AV_MV_IN	=	CAM_MV_IN	
ENG_STATE	=	STATE_ENG	
ERR_SYM_ISA	=	0H	
ERR_SYM_ISA_1	=	0H	
ERR_SYM_ISA_2	=	0H	
ERR_SYM_PUT_PLAUS_ES	=	0H	
ERR_SYM_PUT_PLAUS_FL	=	0H	
FAC_PORT_DEAC_MV	=	0	
FAC_TQ_REQ_VSL	=	0	
IGA_EOLP	=	0	(= 5FH)
IGA_WOUT_KNK	=	0	
ISAPWM	=	OPG_AV_ISA	
LAM_ADJ_i	=	0	
LAMB_i	=	LAMB_SP[NC_CBK_EX_NR]	
LAMB_RGN[NC_CBK_EX_NR]	=	31,999	(= 7FFFH)
LAMB_SA_CH	=	1	
LAMB_SO2P	=	31,999	(= 7FFFH)
LAMB_SP_DIAG_LS_UP_DOWN	=	1	(= 400H)
LAMB_SP_REQ	=	1	(= 400H)
LV_ACT_CRU	=	LV_CRU_ACT	
LV_ACT_VSL	=	0	
LV_CAM_POS_DIAG_EXT_REQ	=	0	
LV_CLU_SWI	=	LV_CS	
LV_CLU_OPEN	=	LV_CS	
LV_CP_RAMP_OPEN_ACT	=	0	
LV_ENA_PRS_CHA_UP	=	0	
LV_ENA_PRS_WG_ACR	=	0	
LV_ENA_PSN_RCL	=	0	

LV_EFP = LV_RLY_EFP

LV_END_PSN_ACT_DIAG_IVVT_IN(EX)_i = 0

IF LV_FIRST_VLD_TOOTH = 1 **THEN** LV_ENG_RUN_DIAG_SAVE = 1 *(set once and only reset at ignition off)*

LV_EOL_OBD = 0

LV_EOL_OBD_DC = 0

LV_ERR_CAN_ICL_TAM = 0

LV_ERR_EL_CPS = LV_ERR_CPS

LV_ERR_ISA_1 = 0

LV_ERR_ISA_2 = 0

LV_ERR_LAM_LIM_i = LV_ERR_FSD_LAM_LIM_i

LV_ERR_MAF_DET = 0

LV_ERR_MAX_FSD = LV_ERR_FSD_1 (deviation Rich)

LV_ERR_SA_SAP = 0

LV_ERR_SA_SAV = 0

LV_ERR_SAP = 0

LV_ERR_SAV = 0

LV_ERR_TPS_DET = 0

LV_ERR_TQ_CONV_ETCU = 0

LV_ERR_VCC_MAP = LV_ERR_VCC_PVS_2

LV_ERR_V_VB_2_MIN = LV_ERR_VB

LV_FCUT_TRA = 0

LV_HOM_ACT = 1

LV_HOM_AFL_ACT = 0

LV_HOM_AFS_ACT = 0

LV_HOM_RUN = 1

LV_IGA_AND_INJ_SWI = 1

LV_IGA_EOL_ACT = 0

LV_ISC_INH_EXT_ADJ = 0

LV_LAM_NOT_STAT_CDN = 0

LV_LAMB_SP_REQ_ACT = 0

LV_LAMB_SP_REQ_DIAG_ACT = 0

LV_LS_DOWN_OBD_2_MAN_DEAC[NC_CBK_EX_NR] = 0

LV_MAX_ERR_FSD_1 = 1

LV_PC_FAN = 0

LV_POW_AC_TRV = 0

LV_PUC_LOCK_TNT = 0

LV_PUC_SA_INH = 0

LV_PUT_ADD_TCHA_BOOST_REQ = 0

LV_PV_SPT_ACT	=	0
LV_RGN_NT_REQ	=	0
LV_RLY_ST	=	0
LV_RNG_L_REQ	=	0
LV_S_ACT	=	0
LV_S_CLC	=	0
LV_S_RUN	=	0
LV_SAP	=	0
LV_SAV	=	0
LV_SAWUP	=	0
LV_TPS_AD_ACT_EXT_ADJ	=	0
LV_TPS_LIH_ACT	=	0
MAP_SP_PQ_MAX	=	IP_MAP_SP_PQ_MAX(N_32)
MFF_ADD_REAC	=	0
MFF_SP_S	=	0
NC_USE_SEG_HALF_AVL	=	0
OPM_AV	=	2H (AFS)
SAF_KGH	=	0 (kg/h)
STATE_CMB_CTL	=	0
STATE_RBM_DYN_TIA	=	STATE_RBM_TIA_DYN
STATE_SA	=	0 (no state)
STATE_SAV_DIAG	=	0 (not active)
TAM_BAS	=	VP_TAM
TEG_DYN	=	TEG_DYN_LS_UP[0] – different range !
TIA_DIF_PLAUS_TIA_THD	=	0
TQ_ADD_HEAT_ACC	=	0
TQ_ADD_SO2P	=	0
TQ_CONV_ETCU	=	0
VLFT_AV	=	0
VP_MAP_MV_MAX_DIAG	=	VP_MAP_MAX_DIAG
VP_MAP_MV_MIN_DIAG	=	VP_MAP_MIN_DIAG
VS_FIL	=	VS (different resolution!)
VS_STATE	=	VS_STATE_CFA
LV_FUP_LAMP_ON	=	0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAP_SP_PQ_MAX	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDP_N_32_IP_MAP_SP_PQ_MAX	6	0...FFH	0...8.16E+3	32	rpm
Minimum MAP_SP for unthrottled engine operation for charged engine					

1.44 Specification coherency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ACCOUT_RLY	O	0...1H	0...1	1	-
Dummy variable, for spec coherency only					
LV_CRK_ERR	O	0...1H	0 ... 1	-	-
converted information about crank signal error					
T_BAT_LIN_FIRST_VLD	O	0...FEH	-48....142.5	1	-
Dummy variable, for spec coherency only					

Input data:

LV_RLY_ACCOUT	LV_ERR_CRK	TBAT_LIN_FIRST_VLD	

Formula section:

LV_ACCOUT_RLY = LV_RLY_ACCOUT

LV_CRK_ERR = LV_ERR_CRK

T_BAT_LIN_FIRST_VLD = TBAT_LIN_FIRST_VLD

1.45 INTC - Introduction

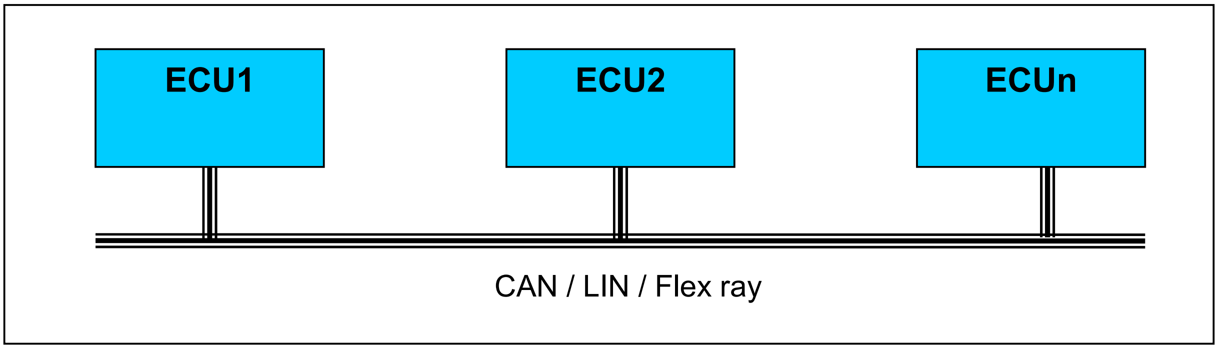
FUNCTION DESCRIPTION:

General information:

Intersystem communication stands for the communication across different control units (ECU) for various applications. The exchange of application data takes place in the form of messages. Data links like CAN, LIN or FlexRay are used for this communication purpose.

This document provides the brief introduction of these datalink layer protocols.

Signal flow diagram:



Description:

LIN
LIN stands for Local Interconnect Network.
LIN is a single wire bus system.
LIN network is a single master system with a maximum of 15 slave nodes.
LIN was designed to link switches, actuators and sensors in to a sub-bus that connects to the main bus, usually via a CAN.
LIN implementation is based on common UART/SCI interface hardware.
LIN is an inexpensive, low speed (from 1 to 20KiloBitsPerSecond), serial multiplexing protocol.

Most of the times the LIN master node implementation is done in the ECU. The LIN master node implementation in an ECU involves three different layers

- Application software
- LIN data link layer
- Low level Asynchronous (ASY) driver

The LIN master node functionality is explained in the below mentioned specification.
LIN: Master Node Data Link Layer
LIN: Master Node Data Link Layer (Appl. Inc.)

Normative reference for LIN

This implementation and the description of LIN data link layer is based on :
LIN Specification Package Revision 2.0 LIN Consortium 2003.

2 Basic SW Inputs and Outputs

Table of Contents

Index

A

B

C

E

F

G

I

P

2.1 Analog Inputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VP_MAF_CAL	O	0...3FFH	0...4.9951	5 / 1024	Volt
Mass Air Flow sensor for Calibration purpose raw acquisition.					
VB_BAS	O	0...3FFH	0... 4.9951	5 /1024	Volt
Battery voltage raw acquisition.					
V_IGK_BAS	O	0...3FFH	0...4.9951	5 / 1024	Volt
Battery voltage raw acquisition.					
VCC_SENS_SUB_DIAG_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage supply for DTP, APT, PORT Poti, VIM Poti, MAP 2 and PSTE_AN					
FTL_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Main Fuel tank level sensor raw acquisition					
TOIL_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Oil temperature sensor raw acquisition.					
V_DTP	V/O	0...03FFH	0...4.9951	5 / 1024	Volt
Differential fuel tank pressure sensor raw acquisition.					
ACP_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Air conditioning pressure transducer voltage raw acquisition.					
MWSS_DIAG_BAS	O	0...3FFH	0...4.9951	5 / 1024	Volt
Magnetic wheel speed signal sensor raw acquisition.					
V_TECU	O	0...3FFH	0...4.9951	5 / 1024	Volt
ECU/PCU hardware temperature sensor voltage					
VCC_TPS_DIAG_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage supply for throttle position sensor 1/2 raw acquisition					
VCC_PVS1_DIAG_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage supply for pedal value sensor 1 raw acquisition					
VCC_PVS2_DIAG_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage supply for pedal value sensor 2 and MAP1 raw acquisition					
PSP_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Power steering pressure transducer voltage raw acquisition.					
V_VIM	V/O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage value for position feedback sensor at VIM					
PORT_FB_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage value for position feedback Sensor Port Flap					
CRU_BAS	V/O	0...03FFH	0...4.9951	5 / 1024	Volt
Cruise switch signal					
TPS_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Throttle Position Sensor raw acquisition					
VP_TAM	V/O	0...03FFH	0...4.9951	5 / 1024	Volt
Ambient Temperature Sensor Raw Aquisition					
FTL_BAS_SUB	O	0...03FFH	0...4.9951	5 / 1024	Volt
Sub fuel tank level sensor raw acquisition					
TBAT_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Battery temperature sensor raw acquisition.					
CUR_BAT_BAS	O	0...03FFH	0...4.9951	5 / 1024	Volt
Battery Sensor Charging Current Raw Aquisition					
V_PC_FAN	O	0...03FFH	0...4.9951	5 / 1024	Volt
Voltage value from Passenger compartment blower switch					

FUNCTION DESCRIPTION:

Direct analog Inputs from A/D Converter

Name	Acquisition rate (msec.)	Port - Channel	Application recurrence (msec.)	Remarks
VCC_TPS_DIAG_BAS	1	A_AN0/PQB0	1	
PSP_BAS	1	A_AN1/PQB1	10	Power Steering Pressure
V_TPS_1_BAS	*	A_AN2/PQB2	**	Defined in xx2043yy.zzz (Only ETC)
TPS_BAS	1		1	Only ISA
V_PVS_1_BAS	*	A_AN3/PQB3	**	Defined in xx203Eyy.zzz (Only ETC)
VP_TIA	*	A_AN48/PQB4	**	Defined in xx 203Kyy.zzz
TOIL_BAS TBAT_BAS	1	A_AN49/PQB5	10	Depending on CONF_TOIL_MDL and CONF_BAT
KNKS	*	A_AN50/PQB6	**	Defined in xx2007yy.zzz
V(IP)	*	A_AN51/PQB7	*	Defined in XX702Yyy.zzz (only for Linear Lambda)
ACP_BAS	1	A_AN52/PQA0	100	APT Sensor value (same port channel is used alternative for PRS_ACC)
CRU BAS	1	A_AN53/PQA1	10	Only ETC
VLS_UP_RAW[1]	*	A_AN54/PQA2	**	Defined in xx2041yy.zzz (only for Binary Lambda)
PORT_FB_BAS	1	A_AN55/PQA3	20	Only non TCI
VLS_DOWN[1]	*	A_AN56/PQA4	**	Defined in xx2041yy.zzz
V(RI)	*	A_AN57/PQA5	**	Defined in xx702Yyy.zzz (only for Linear Lambda)
FTL_BAS	1	A_AN58/PQA6	100	Main_FTL
VP_TCO_SENS	*	A_AN59/PQA7	**	Defined in xx203Myy.zzz
V_PVS_2_BAS	*	B_AN0/PQB0	**	Defined in xx203Eyy.zzz (Only ETC)
V_VIM V_PC_FAN	1	B_AN1/PQB1	20	deact. by C_CONF_VIM <> 2 Only BEM C_CONF_BAT = 1
V_TPS_2_BAS	*	B_AN2/PQB2	**	Defined in xx2043yy.zzz (Only ETC)
MWSS_DIAG_BAS	1	B_AN3/PQB3	10	Wheel Diag

VP_TAM	1	B_AN48/PQB4	20	Only TCI
VCC_PVS1_DIAG_BAS	1	B_AN49/PQB5	1	Only ETC
V_TECU	1	B_AN50/PQB6	1000	
Name	Acquisition rate (msec.)	Port - Channel	Application recurrence (msec.)	Remarks
VCC_SENS_SUB _DIAG_BAS	1	B_AN51/PQB7	1	
VB_BAS	1	B_AN52/PQA0	10	V_EL
V_DTP	1	B_AN53/PQA1	10	
VP_AMP	*	B_AN54/PQA2	**	Defined in xx203Hyy.zzz (Only TCI)
V_IGK_BAS	1	B_AN55/PQA3	10	
VP_MAP VP_MAF	* *	B_AN56/PQA4	** **	Defined in xx203Hyy.zzz Defined in xx203Hyy.zzz
FTL_BAS_SUB	1	B_AN57/PQA5	100	Sub_FTL
CUR_BAT_BAS	1	B_AN58/PQA6	10	Only for Analog BEM Sensor
VP_PUT VP_MAF_CAL	* 1	B_AN58/PQA6	** 10 / SEG	Defined in xx203Hyy.zzz (TCI) Only for Non TCI
VCC_PVS2_DIAG_BAS	1	B_AN59/PQA7	1	shared sensor supply for PVS2 and MAP1

*) Aquisition rate defined in related module where variable is defined (see remark)
**) Application recurrence defined in related module where variable is defined (see remark)

2.2 Acquisition of after relay battery voltage

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VB_MES	O	0...3FFH	0 ... 4,9951	0,0049	V
Computed battery voltage					

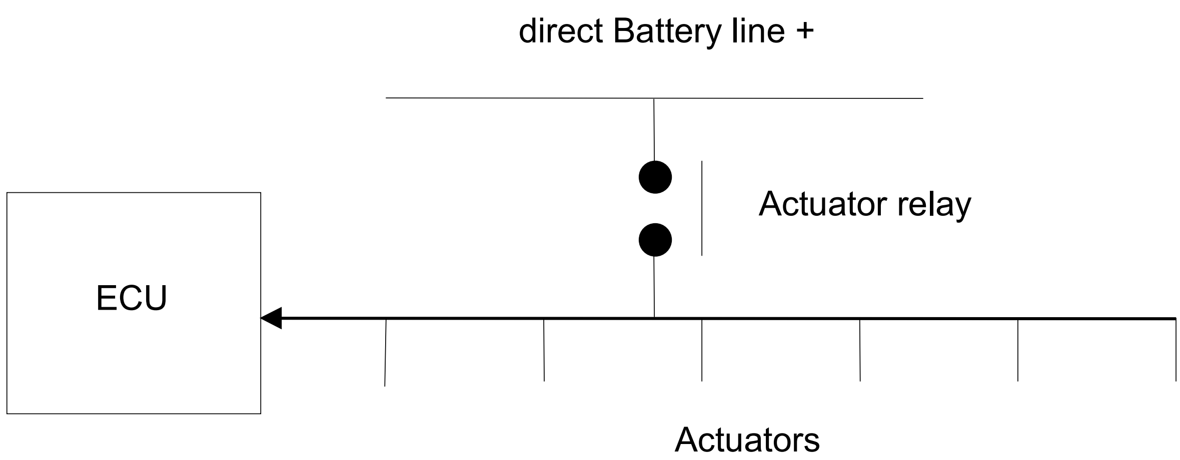
Input data:

VB_BAS			

FUNCTION DESCRIPTION:

General information:

The acquisition of the raw values is performed every 1 ms with the resolution of 10 bits.



Application conditions:

Activation: at reset

Initialization: VB_MES = VB_BAS.

Recurrence : 10 ms

System Description:

The range of VB_BAS and VB_MES depends to the voltage divider (means 0 - 5V at ADC) at the input of the µC. The input is the ADC-value of the V_EL-Pin.

Formula section:

$$VB\_MES = (1/NC\_VB\_SAMPLE\_NR) * \sum_{n=1}^{NC\_VB\_SAMPLE\_NR} VB\_BAS_n$$

NC_VB_SAMPLE_NR = 2^n ; 1 =< n =< 6 (NC_VB_SAMPLE_NR is initialized with 8)

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_VB_SAMPLE_NR	1	0...40H	0...64	1	-
number of acquisition for filtering					

2.3 Acquisition of raw after key battery voltage

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_IGK_MES	O	0...3FFH	0...4,9951	0,0049	V
Computed ignition key voltage					

Input data:

V_IGK_BAS			

FUNCTION DESCRIPTION:

General information:

The acquisition of the raw values is performed every 1 ms with the resolution of 10 bits.
V_IGK_BAS is the same ADC-value which is used in Key on/Key off detection.

Application conditions:

Activation: at reset

Initialization: V_IGK_MES = V_IGK_BAS

Recurrence : 10 ms

Formula section:

$$V\_IGK\_MES = (1/NC\_V\_IGK\_SAMPLE\_NR) * \sum_{n=1}^{NC\_V\_IGK\_SAMPLE\_NR} V\_IGK\_BAS_n$$

NC_V_IGK_SAMPLE_NR = 2^n ; 1 =< n =< 6 (NC_V_IGK_SAMPLE_NR is initialized with 8)

Configuration data:

Name	DIM	Hex. limits	Phys. limits	Resol.	Unit
NC_V_IGK_SAMPLE_NR	1	0...40H	0...64	1	-
number of acquisition for filtering					

2.4 BIOS Knock function

Input data:

KNKS_CMF_FIL_x	KNKS_CMD_INT_x	KNKS_CMD_GAIN_x	KNKS_CMD_CONF_x
KNKWB_x	KNKWE_x		

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNKS_x	O	0 ... FFFFH	0 ... 4.99992	$7.63*10^{-5}$	V
Knock noise raw signal for cylinder x					

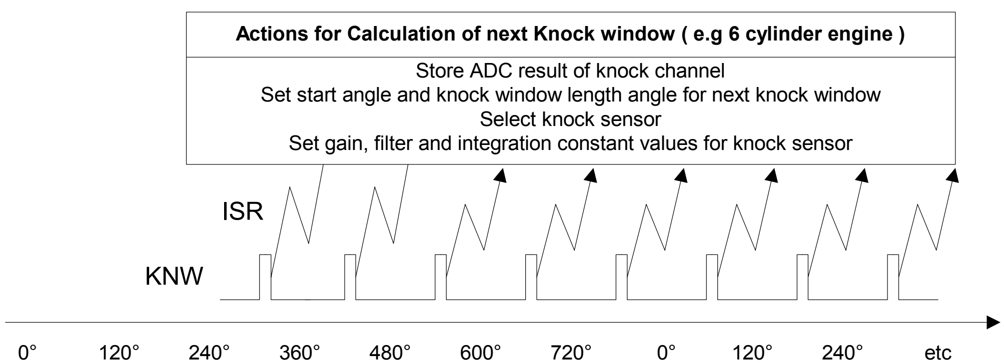
General information:

Activation : at every engine state except engine stop; calculate KNKS_CMD_CONF_x, KNKWB_x, KNKWE_x, KNKS_CMD_FIL_x, KNKS_CMD_INT_X AND KNKS_GAIN_x.
Deactivation : T_KNK_DISABLE

Initialization : calculate KNKS_CMD_CONF_x, KNKWB_x, KNKWE_x, KNKS_CMD_FIL_x, KNKS_CMD_INT_X AND KNKS_GAIN_x T_KNK_INIT
Update rate : every segment recalculate KNKWB_x, KNKWE_x, KNKS_CMD_FIL_x, KNKS_CMD_INT_x and KNKS_GAIN_x

The knock detection is performed for cylinder-individually defined measurement windows since knock-typical oscillations only occur for certain crankshaft angles. For the time of the knock window the knock signal is integrated. At the end of the knock window the integrated signal is sampled and converted to the digital value KNKSx.

The knock window for cylinder x+1 is reprogrammed at the end of the knock window for cylinder x. The parameters KNKS_CMD_FIL_x, KNKS_CMD_INT_x, KNKS_CMD_GAIN_x, KNKS_CMD_CONF_x, KNKWB and KNKWE have to be stable at this time. The knock channel adc is configured for triggering on the end of the knock window. The end of conversion generates an interrupt in which the parameters for the next knock window are set. The following picture illustrates this.



The knock function may be activated by T_KNK_INIT or T_KNK_ENABLE and deactivated by T_KNK_DISABLE.

2.5 Logic Inputs

2.5.1 Basic software

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_IM_ACIN	-	0...01H	0...1	1	-
Air Condition selected.					
LV_IM_ACCIN	-	0...01H	0...1	1	-
Air Condition requested.					
LV_IM_PRS_ACC	-	0...01H	0...1	1	-
Pressure switch for Air Condition circuit.					
LV_IM_VB_OFF	-	0...01H	0...1	1	-
Battery disconnected detection.					
LV_IM_CS_PN	O	0...01H	0...1	1	-
Clutch switch					
LV_PIN_BLS	O	0...01H	0...1	1	-
Brake light switch					
LV_PIN_BTS	O	0...01H	0...1	1	-
Brake test switch					
LV_IM_PSTE	-	0...01H	0...1	1	-
Power Steering Switch					
LV_IM_HDLP	-	0...1H	0...1	1	-
Head Lamp switch					

FUNCTION DESCRIPTION:

List of logic Inputs

Name	Port - Channel	Logic state	Remarks
LV_IM_ACIN	MDA30	Positive Logic	
LV_IM_ACCIN	MDA31	Positive Logic	
L_IM_PRS_ACC	A_AN52/PQA0	Positive Logic	
LV_IM_VB_OFF	MDA14	Positive Logic	
LV_IM_CS_PN	MPI032B11	Positive Logic	Only ETC
LV_PIN_BLS	MDA15	Positive Logic	Only ETC
LV_PIN_BTS	MDA12	Negative Logic	Only ETC
LV_IM_PSTE	SPGIOC7/IRQOUT/LWP0	Negative Logic	
LV_IM_HDLP	B_TPUCH_15	Positive Logic	Only ETC

2.5.2 Applicative software

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DRI	V/O	0...01H	0...1	1	-
Boolean for DRIVE or REVERSE engaged.					
LV_ACIN	V/O	0...01H	0...1	1	-
Air condition selected.					
LV_ACCIN	V/O	0...01H	0...1	1	-
Air condition requested.					
LV_PRS_ACC	V/O	0...01H	0...1	1	-
Pressure switch for Air Condition circuit.					
LV_PSTE	V	0...01H	0...1	1	-
Pressure switch for power steering circuit.					
LV_HDLP	V/O	0...01H	0...1	1	-
Head Lamp switch					

Input data:

CONF_ACC	LV_IM_PRS_ACC	LV_IM_ACIN	TAR_GC
PSP	CONF_TCU	LV_IM_ACCIN	CONF_PSTE
LV_IM_PSTE	LV_IM_HDLP	CONF_BAT	
LV_IM_ACCIN_CAN			

2.5.2.1 Air-condition system :

```
If      CONF_ACC = 1
then
        if      CONF_ACC_FATC = 0
        then    LV_ACIN = LV_IM_ACIN
        else    LV_ACIN = LV_IM_ACIN_CAN
        endif

if  CONF_ACC_FATC = 0              then    LV_ACCIN = LV_IM_ACCIN
        else    if      LC_TQ_LOSS_ACC_CLULESS = 0
                then    LV_ACCIN = LV_IM_ACCIN_CAN
                else    if      TQ_LOSS_ACC_CAN < C_TQ_LOSS_ACCIN_CAN
                        then            LV_ACCIN = 1
                        else            LV_ACCIN = 0
                        endif
                endif
        endif

    LV_PRS_ACC = LV_IM_PRS_ACC
```

else LV_ACIN = 0
and LV_ACCIN = 0
and LV_PRS_ACC = 0

2.5.2.2 Automatic gear-box system :

If CONF_TCU = 1
and TAR_GC > 0
then LV_DRI = 1
else LV_DRI = 0

Remark: For detailed information refer to CAN - Specification.

2.5.2.3 Power steering system :

If CONF_PSTE = 0
then LV_PSTE = LV_IM_PSTE

2.5.2.4 Battery and Energy Management

If CONF_BAT = 1
then LV_HDLP = LV_IM_HDLP
endif

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TQ_LOSS_ACCIN_CAN	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
ACC torque loss for LV_ACCIN					
LC_TQ_LOSS_ACC_CLULESS	1	0...1H	0...1	1	[-]
Boolean to clear clutchless external aircon control unit					

2.6 Acquisition of Crankshaft Signal

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ST_CRK_SYN	-	0...7H	0...7	1	[-]
Crankshaft synchronisation state					
LV_CRK_STOP	O	0...1H	0...1	1	[-]
Flag for Stop Engine request					
LV_CRK_RUN	O	0...1H	0...1	1	[-]
Flag for Running engine					
LV_ORNG_TOOTH_PER_CRK	V/O	0...1H	0...1	1	[-]
Invalid crankshaft tooth period					
LV_ORNG_NR_TOOTH_CRK	V/O	0...1H	0...1	1	[-]
Crankshaft tooth count incorrect at reference gap					
LV_LOST_SYN_CRK	V/O	0...1H	0...1	1	[-]
Crankshaft synchronization lost					
CRK_CTR	-	0...FFH	0...255	1	[-]
Crankshaft counter					
CRK_ADD_CTR	-	0...FFH	0...255	1	[-]
Crankshaft additional teeth counter					
CRK_MISS_CTR	-	0...FFH	0...255	1	[-]
Crankshaft missing teeth counter					
LV_CRK_FIRST_VLD_TOOTH	O	0...1H	0...1	1	[-]
Flag for first valid crankshaft tooth detected ready to synchronize					
LV_CRK_SYN	V/O	0...1H	0...1	1	[-]
Flag for crankshaft acquisition synchronized					
T_TOOTH	O	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Crankshaft tooth period					
T_SEG_ENSD	O	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Fine resolution segment period					
T_SEG_HALF_ENSD	O	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Fine resolution half-segment period					
T_SEG_ER	V/O	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Crankshaft segment period for misfire detection (ER algorithm)					
T_CRK_WIN_ENSD	V/O	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Crankshaft tooth window at end of segment					
LV_FIRST_REF_GAP	-	0...1H	0...1	1	[-]
First Reference gap is found					
LV_REF_GAP	-	0...1H	0...1	1	[-]
Second Reference gap is found					
LV_CRK_MISS_TOOTH	V/O	0...1H	0...1	1	[-]
At least one missing tooth has been simulated					

Input data:

N_32	LV_ACT_CRK	NC_CYL_NR	NC_NR_GAP

FUNCTION DESCRIPTION:

2.6.1 Crankshaft Synchronization

General information:

The crankshaft signal is generated by the crankshaft sensor in conjunction with the crankshaft target wheel. The target wheel has a theoretical number of NC_NR_TOOTH (typically 60) teeth, and NC_NR_TOOTH_GAP missing teeth (typically 2) forming the reference gap (“60-2”). The crankshaft sensor converts the mechanical shape into an electrical signal.

Magnetic Crankshaft Position Sensors (MCPS) deliver an AC signal with an amplitude depending primarily on crankshaft speed, and sensor air gap. The signal is shaped into a rectangular signal by the ECU input circuitry. Each edge of that signal represents a tooth or a gap of the target wheel. Only signal edges generated from teeth will be processed by software (active signal edge definition). These may be falling or rising edges, depending on sensor polarity.

Active Crankshaft Position Sensors (ACPS) switch an open collector output on and off with every tooth or gap passing. Connecting the sensor output to a pull-up resistor inside the ECU generates a signal with rectangular shape. Each edge of that signal represents a tooth or a gap of the target wheel. Only falling signal edges will be processed by software (active signal edge definition), because they are faster. Design and installation of the sensor must ensure that falling signal edges are generated from teeth.

The corresponding configuration data is NC_ACT_CRK_EDGE.

The choice is taken in a way not to use the unprecise signal edge generated in the crankshaft reference gap.

It may be 0 (falling) or 1 (rising) with MCPS, depending on sensor polarity.

With ACPS it should always be 0, because the falling edge is faster due to the sensor’s open collector interface. The unprecise signal edge in the reference gap is avoided by choosing the correct installation of the sensor relativ to the target wheel motion direction.

Application conditions:

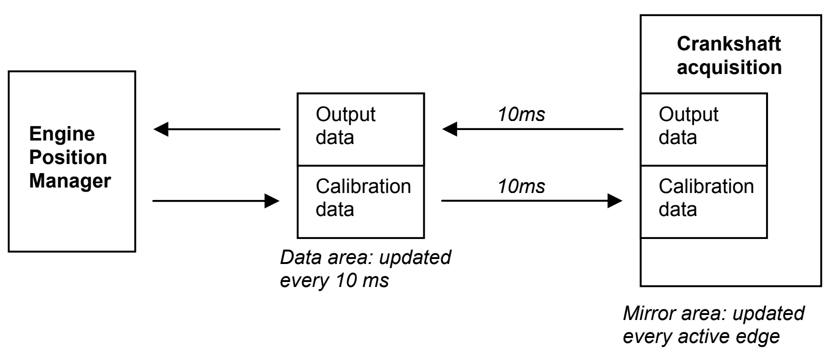
Recurrence:

For set of the RAM as flags, a mirror area is defined which contains a copy of the output and calibration data in the basic software area (see figure below).

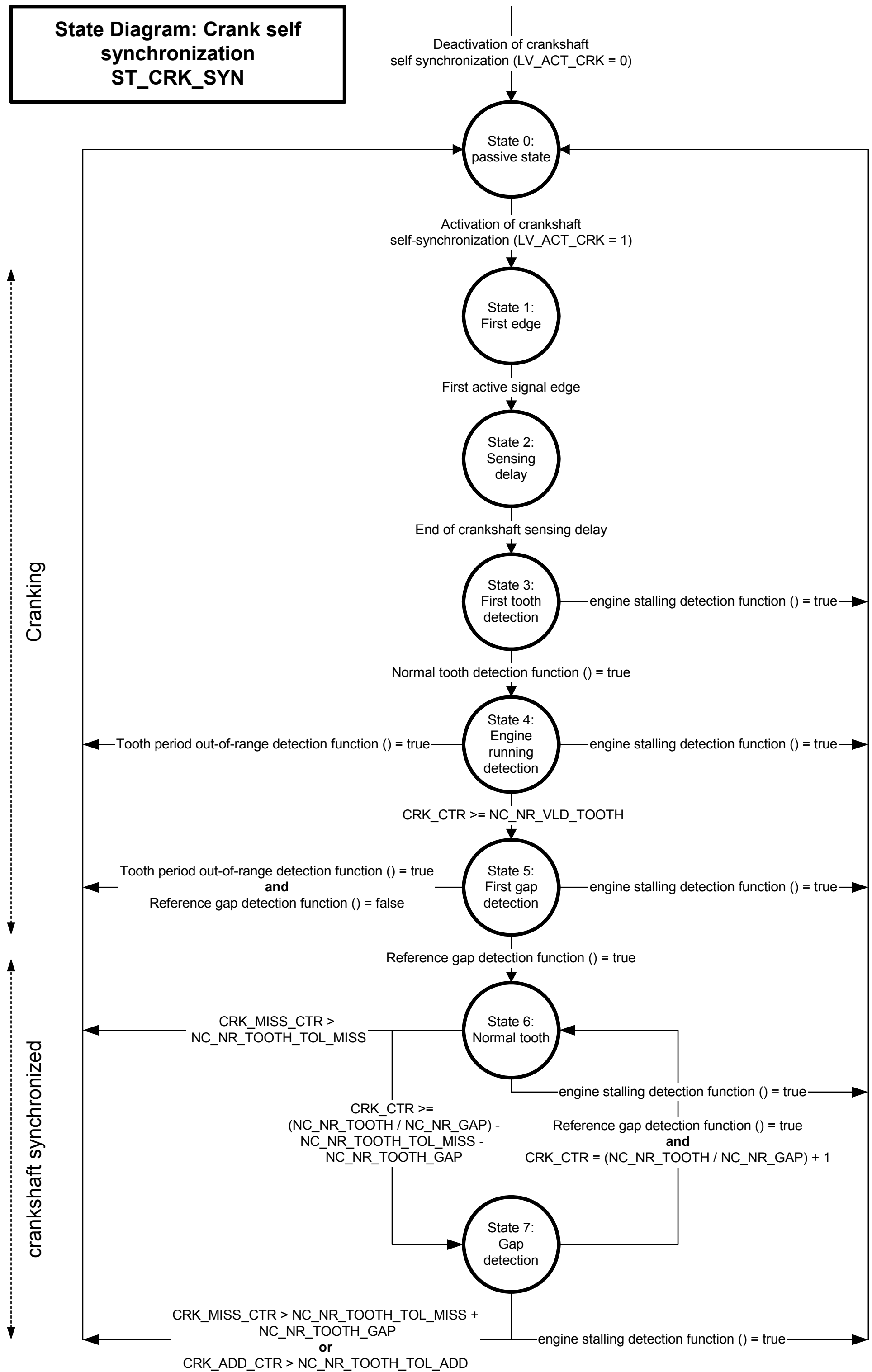
The mirror area of the output data is updated by the lower layer at every active signal edge.

The mirror area of the calibration data is updated by the upper layer each 10ms.

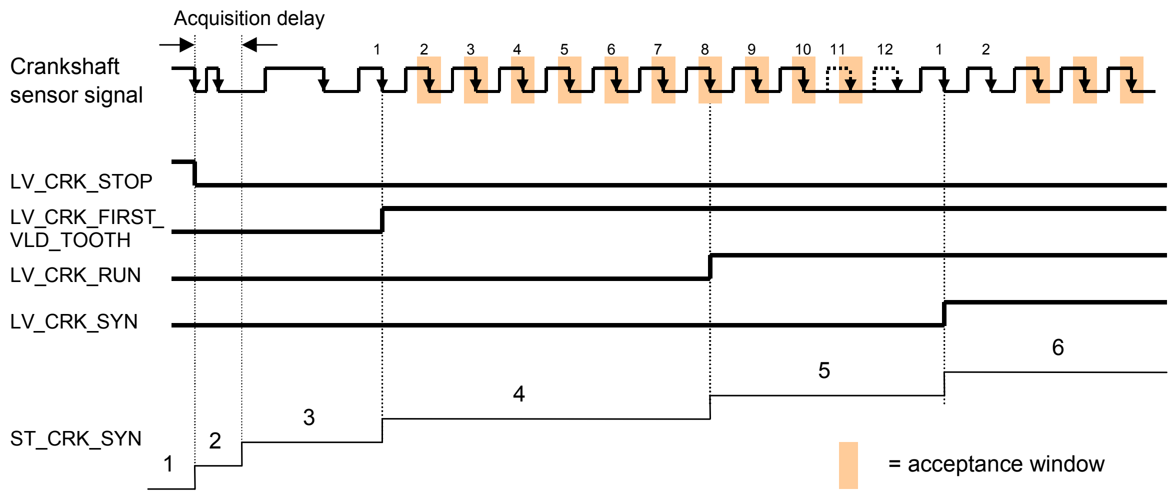
The output data is copied from the mirror area every 10 ms



State Flow Diagram:



Timing Diagram (example):



Formula section:

State 0: Crankshaft passive state

Input condition :

From state 4,5:	Tooth period out-of-range detection function () = true
From state 6:	CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS
From state 7:	CRK_ADD_CTR > NC_NR_TOOTH_TOL_ADD
	Or
	CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS + NC_NR_TOOTH_GAP
From state 3,4,5,6,7:	Engine stalling detection function () = true
From EPM :	LV_ACT_CRK = 0

Output condition :

To State 1: Activation of crk self-synchronization (LV_ACT_CRK = 1)

Action in the state :

LV_CRK_FIRST_VLD_TOOTH = 0

LV_CRK_SYN = 0

CRK_CTR = 0

LV_CRK_STOP = 1

LV_CRK_RUN = 0

CRK_MISS_CTR = 0

CRK_ADD_CTR = 0

LV_FIRST_REF_GAP = 0

LV_REF_GAP = 0

Transient action :

None

State 1: First signal edge

Input condition :

From state 0: Activation of crk self-synchronization (LV_ACT_CRK = 1)

Output condition :

To state 2: First active signal edge

Action in the state :

Wait for first active signal edge

LV_CRK_STOP = 0

Transient action :

No actions

State 2: Crankshaft sensing delay

Input condition :

From state 1: First active signal edge detected

Output condition :

To state 3: End of crankshaft acquisition delay time

Action in the state :

Crankshaft signal acquisition inactive during a time interval C_T_CRK_DLY.

Transient action :

To state 3: LV_ORNG_TOOTH_PER_CRK = 0

LV_ORNG_NR_TOOTH_CRK = 0

LV_LOST_SYN_CRK =0

State 3: First tooth detection

Input condition:

From state 2 : End of crankshaft acquistion delay time

Output condition:

State 4: Normal tooth detection function () = true

State 0: Engine stalling detection function () = true

Action in the state

Wait two next active signal edges in order to compute the first period

Activation of Normal tooth detection function ()

T_TOOTH n-1 = T_TOOTH n

Activation of Tooth period out-of-range detection function ()

Activation of Engine stalling detection function ()

Transient action :

To state 4 : LV_CRK_FIRST_VLD_TOOTH = 1

Increment CRK_CTR

State 4: Engine running detection

Input condition:

From state 3: Normal tooth detection function () = true

Output condition:

State 5: CRK_CTR >= NC_NR_VLD_TOOTH

State 0: Engine stalling detection function () = true

Or

Tooth period out-of-range detection function () = true

Action in the state:

If Normal tooth detection function () = true

And
Tooth period out-of-range detection function () = false
Then Increment CRK_CTR
Endif

If Tooth period out-of-range detection function () = true
Then LV_ORNG_TOOTH_PER_CRK = 1
Endif

Action in transient:

To state 5: LV_CRK_RUN = 1 (Engine speed can be computed)

<u>State 5: First gap detection</u>

Input condition :

From state 4: CRK_CTR >= NC_NR_VLD_TOOTH

Output conditions :

To state 6: Reference gap detection function () = true

To State 0: Engine stalling detection function () = true
Or
(Tooth period out-of-range detection function () = true
And
Reference gap detection function () = false)

Action in the state :

Activation of Reference gap detection function ()
Activation of Missing tooth simulation in gap position function ()

If Normal tooth detection function () = true
And
Tooth period out-of-range detection function () = false
Then Increment CRK_CTR
Endif

If Tooth period out-of-range detection function () = true
And
Reference gap detection function () = false

Then LV_ORNG_TOOTH_PER_CRK = 1

Endif

Transient action :

To state 6: LV_CRK_SYN =1

CRK_CTR = 1

LV_FIRST_REF_GAP = 1

State 6: Normal tooth

Input conditions :

From State 5: Reference gap detection function () = true

From State 7: Reference gap detection function () = true

And

CRK_CTR = (NC_NR_TOOTH / NC_NR_GAP) + 1

Output conditions :

To State 7: CRK_CTR >= (NC_NR_TOOTH / NC_NR_GAP) –
NC_NR_TOOTH_GAP - NC_NR_TOOTH_TOL_MISS

To State 0: CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS

Or

engine stalling detection function () = True

Action in the state :

Deactivation of Missing tooth simulation in gap position function ()

Activation of Missing tooth simulation function ()

if Normal tooth detection function ()= true

And

Tooth period out-of-range detection function () = false

Then Increment CRK_CTR

Endif

If Tooth period out-of-range detection function () = true

Then LV_ORNG_TOOTH_PER_CRK = 1

Endif

Transient action :

To State 0: **If** CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS
Then LV_LOST_SYN_CRK = 1
Endif

State 7 : Gap detection

Input condition :

From state 6: CRK_CTR >= (NC_NR_TOOTH / NC_NR_GAP) – NC_NR_TOOTH_GAP - NC_NR_TOOTH_TOL_MISS

Output condition :

To State 6: Reference gap detection function () = true
And
CRK_CTR = (NC_NR_TOOTH / NC_NR_GAP) + 1

To State 0: CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS + NC_NR_TOOTH_GAP
Or
CRK_ADD_CTR > NC_NR_TOOTH_TOL_ADD
Or
Engine stalling detection function () = True

Action in the state :

Deactivation of Missing tooth simulation function ()
Activation of Missing tooth simulation in gap position function ()

If Normal tooth detection function ()= true
And
Tooth period out-of-range detection function () = false
Then Increment CRK_CTR
If CRK_CTR > (NC_NR_TOOTH / NC_NR_GAP) – NC_NR_TOOTH_GAP
Then Increment CRK_ADD_CTR
Endif
Endif

	If	Tooth period out-of-range detection function () = true
		And
		Reference gap detection function () = false
	Then	LV_ORNG_TOOTH_PER_CRK = 1
	Endif	

Transient action :

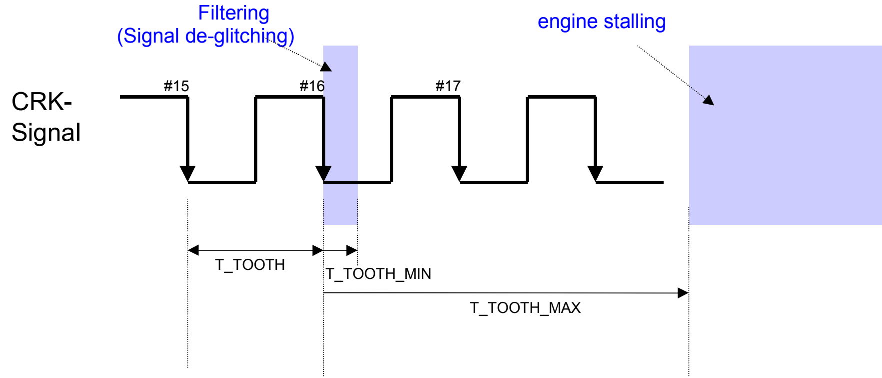
To state 6:	**If**	CRK_MISS_CTR > 0
		Or
		CRK_ADD_CTR > 0
	Then	LV_ORNG_NR_TOOTH_CRK = 1
	Endif	
	CRK_CTR = 1	
	CRK_MISS_CTR = 0	
	CRK_ADD_CTR = 0	
	LV_REF_GAP =1	
To State 0:	**If**	CRK_MISS_CTR > NC_NR_TOOTH_TOL_MISS + NC_NR_TOOTH_GAP
		Or
		CRK_ADD_CTR > NC_NR_TOOTH_TOL_ADD
	Then	LV_LOST_SYN_CRK = 1
	Endif	

2.6.2 Definition of the sub function task

2.6.2.1 Normal tooth detection function ()

General information:

The purpose of this function is to validate the tooth period of a normal tooth. The function returns false when the tooth period is too small (Signal edges detected before a delay time of T_TOOTH_MIN are not valid).



Formula section:

$$T\_TOOTH\_MIN = \frac{60}{NC\_NR\_TOOTH * NC\_N\_MAX}$$

If	T_TOOTH => T_TOOTH_MIN
Then	return (true)
Else	return (false)
Endif	

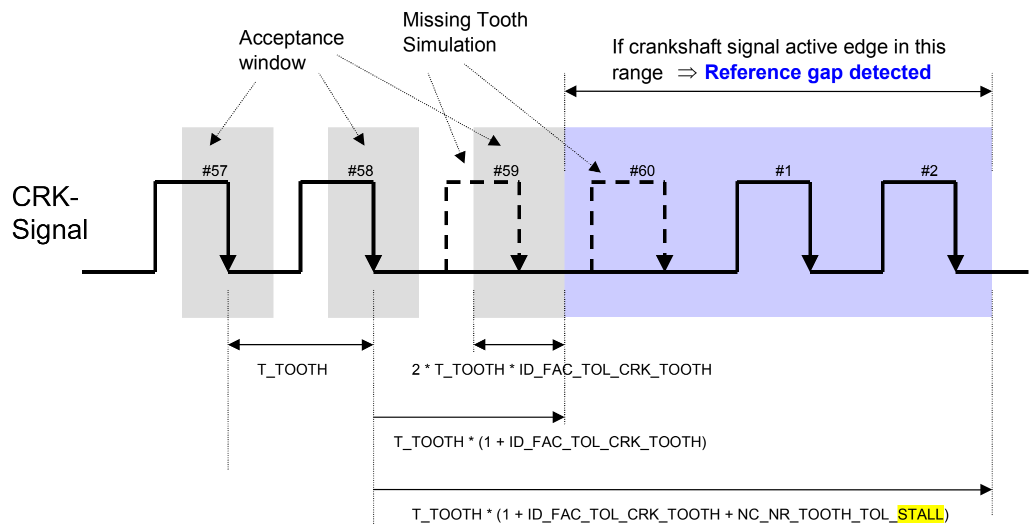
2.6.2.2 Reference gap detection function ()

General information:

The purpose of this function is to detect a reference gap.

A reference gap is detected if no signal edge was detected inside the crankshaft acceptance window and at least 1 (after synchronization) or NC_NR_TOOTH_GAP (first gap detection) teeth have been completely simulated.

The follow figure shows the reference gap detection after synchronization:



Formula section:

If LV_CRK_SYN = 1

Then **If** Number of simulated missing teeth in gap position >=1

Then **If** (NC_NR_TOOTH / NC_NR_GAP) – CRK_MISS_CTR <=

CRK_CTR <=

(NC_NR_TOOTH / NC_NR_GAP) + CRK_ADD_CTR

Then (*correct tooth counter)*

CRK_CTR = CRK_CTR + CRK_MISS_CTR – CRK_ADD_CTR

return (true)

Else return (false)

Endif

Else **If** Number of simulated missing teeth in gap position >=

NC_NR_TOOTH_GAP

Then return (true)

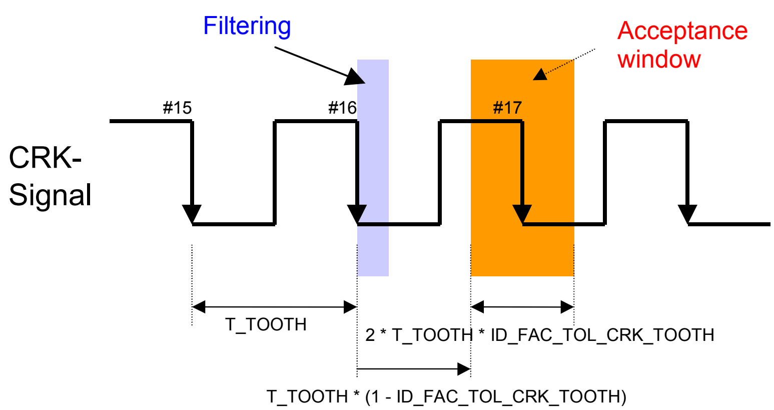
Endif

Endif

2.6.2.3 Tooth period out-of-range detection function ()

General information:

The purpose of this function is to check if the tooth period is within a defined window area. The function returns true if the tooth period is outside the acceptance window.



Formula section:

Before first reference gap:

If T_TOOTH n-1 * (1 – ID_FAC_TOL_CRK_TOOTH) ≤ T_TOOTH n ≤ T_TOOTH n-1 * (1 + ID_FAC_TOL_CRK_TOOTH)

Then return (false)

Else return (true)

After first reference gap:

If T_TOOTH n-1 * (1 – ID_FAC_TOL_CRK_TOOTH) ≤ T_TOOTH n

Then return (false)

Else return (true)

Remark: If T_TOOTH > T_TOOTH n-1 * (1 + ID_FAC_TOL_CRK_TOOTH) after first reference gap, then a tooth is simulated (see below "Missing Tooth Simulation Function").

The acceptance window is disabled

- on the first and second crankshaft signal edge after a reference gap detection
- on the first and second crankshaft signal edge after simulation of a missing tooth

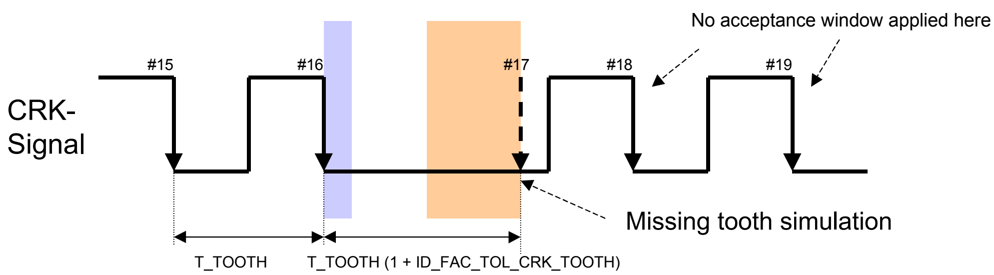
2.6.2.4 Missing Tooth Simulation Function ()

General information:

Simulate a tooth with tooth period T_TOOTH n-1 (last measured tooth period).

All measurements shall be based on the simulated tooth not on real edges.

Simulation starts at the last (real or simulated) signal edge if no signal edge is detected inside the acceptance window defined by ID_FAC_TOL_CRK_TOOTH.



Formula section:

Increment CRK_CTR.

Increment CRK_MISS_CTR.

LV_CRK_MISS_TOOTH = 1

2.6.2.5 Missing Tooth Simulation in Gap Position Function ()

General information:

Simulate a tooth with tooth period T_TOOTH n-1 (last measured tooth period).

Simulation starts at the last (real or simulated) signal edge if no signal edge is detected inside the acceptance window defined by ID_FAC_TOL_CRK_TOOTH.

Simulation stops when a edge is detected after the first simulated missing tooth.

All measurements shall be based on the simulated tooth not on real edges.

Formula section:

Increment CRK_CTR.

If number of simulated teeth < NC_NR_TOOTH_GAP

And

Missing tooth simulation stopped

Then Increment CRK_MISS_CTR

High acceleration during reference gap: only one tooth could be simulated

Endif

If number of simulated teeth > NC_NR_TOOTH_GAP

Then Increment CRK_ADD_CTR

High deceleration during reference gap: more than NC_NR_TOOTH_GAP teeth simulated

Endif

2.6.2.6 Engine stalling detection function ()

General information:

Engine stalling is detected when the number of consecutive simulated teeth exceeds NC_NR_TOOTH_STALL or when the tooth period is greater than T_TOOTH_MAX.

Formula section:

$$T\_TOOTH\_MAX = \frac{60}{NC\_NR\_TOOTH * NC\_N\_MIN}$$

If more than NC_NR_TOOTH_ STALL consecutive teeth are simulated

Or

T_TOOTH (1 + ID_FAC_TOL_CRK_TOOTH) >= T_TOOTH_MAX

Then return (true)

Else return (false)

Endif

2.6.3 Requirements to infrastructure

2.6.3.1 Crankshaft Tooth Time Measurement (T_TOOTH)

General information:

System request accuracy <= 4 us

The crankshaft tooth period T_TOOTH is the measured time between two consecutive valid active crankshaft signal edges (falling or rising according to configuration).

If a missing tooth is simulated, T_TOOTH represents the time period between the last two real valid signal edges. T_TOOTH keeps the latest value until a new measured value is available.

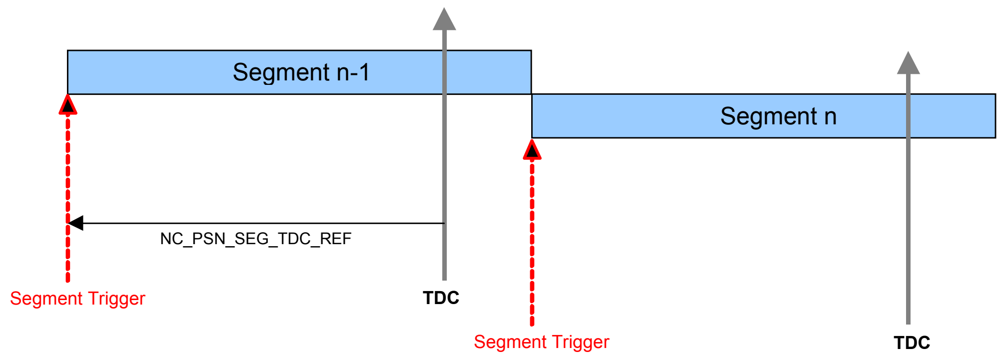
2.6.3.2 Segment Period Measurement (T_SEG_ENSD) and Trigger Generation

General information:

System request accuracy <= 1 us

At Segment event, a trigger is defined and a control signal is generated to trigger execution of segment synchronous tasks.

The corresponding time intervals T_SEG_ENSD are measured between tooth events located NC_PSN_SEG_TDC_REF degrees before each TDC.



Application conditions:

Initialisation:	T_SEG_ENSD is set to the maximum value
Recurrence :	before synchronization: once (when LV_CRK_RUN is set) after synchronization: NC_PSN_SEG_TDC_REF degrees before each TDC (segment trigger)
Activation :	LV_CRK_RUN = 1
Deactivation :	LV_CRK_STOP = 1

Formula section:

Before synchronization, one segment time is output at the moment when crankshaft rotation is validated. The first segment period is calculated from the most recent tooth period:

T_SEG_ENSD = (2 * T_TOOTH * NC_NR_TOOTH) / NC_CYL_NR

The same calculation applies to the first segment after synchronization.

After synchronization:

T_SEG_ENSD = time interval between tooth events located NC_PSN_SEG_TDC_REF degrees before each TDC

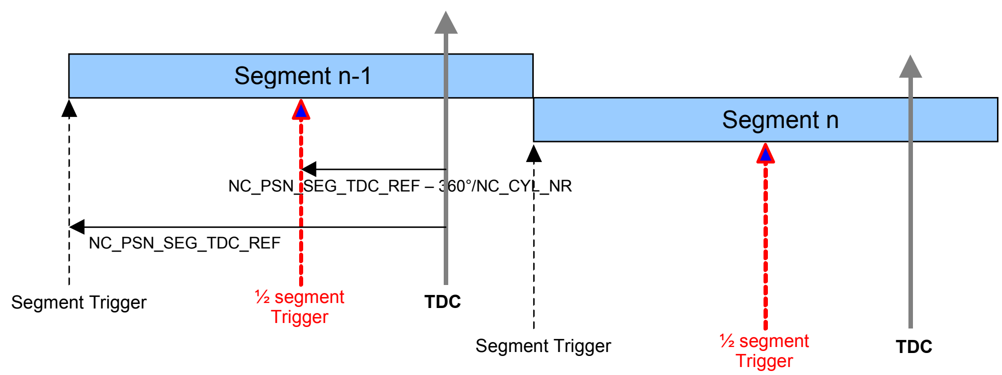
2.6.3.3 Half-Segment Period Measurement (T_SEG_HALF_ENSD) and Trigger Generation

General information:

System request accuracy <= 1 us

At the half of a segment (in the middle of two segment events) a half-segment event is generated and an half-segment trigger is defined and send to the ASW if the engine speed is lower than a threshold defined by NC_N_SEG_HALF_END.

This function provides a output value for the half-segment period. This measured time is send to the ASW.



Application conditions:

Initialisation: T_SEG_HALF_ENSD is set to the maximum value

Recurrence : before synchronization: once (when LV_CRK_RUN is set)
after synchronization:

NC_PSN_SEG_TDC_REF degrees before each TDC (segment trigger)

and

NC_PSN_SEG_TDC_REF – 360° CRK / NC_CYL_NR degrees before each TDC (half-segment trigger) **if** T_SEG_HALF_ENSD >= 60 / (NC_CYL_NR * NC_N_SEG_HALF_END)

Activation : LV_CRK_RUN = 1

Deactivation : LV_CRK_STOP = 1

Formula section:

The first half segment period is calculated from the most recent tooth period:

T_SEG_HALF_ENSD = (T_TOOTH * NC_NR_TOOTH) / NC_CYL_NR

After synchronization:

T_SEG_HALF_ENSD = time elapsed between

- a segment event and a half-segment event or
- a half-segment event and a segment event

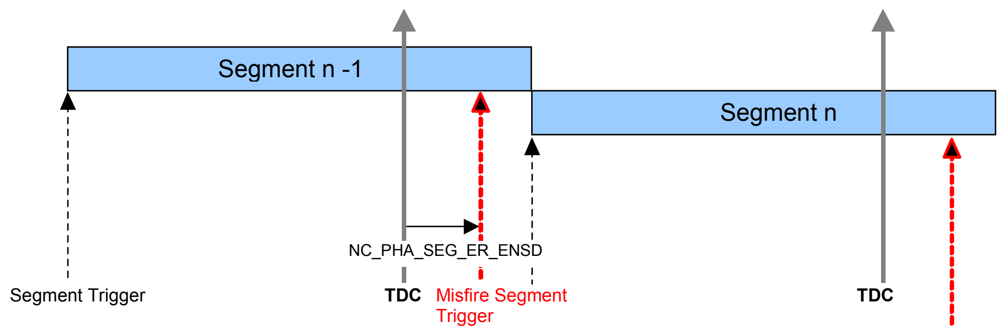
2.6.3.4 Misfire Segment Period Measurement (T_SEG_ER)

General information:

System request accuracy <= 1 us

The function provides output values for the misfire segment period.

The misfire segment period T_SEG_ER corresponds to the TDC period, measured at the tooth events, which are closest to NC_PHA_SEG_ER_ENSD after each TDC.



Application conditions:

Initialisation: T_SEG_ER is set to the maximum value

Recurrence : every misfire segment

Formula section:

T_SEG_ER = time interval between misfire segment tooth events.

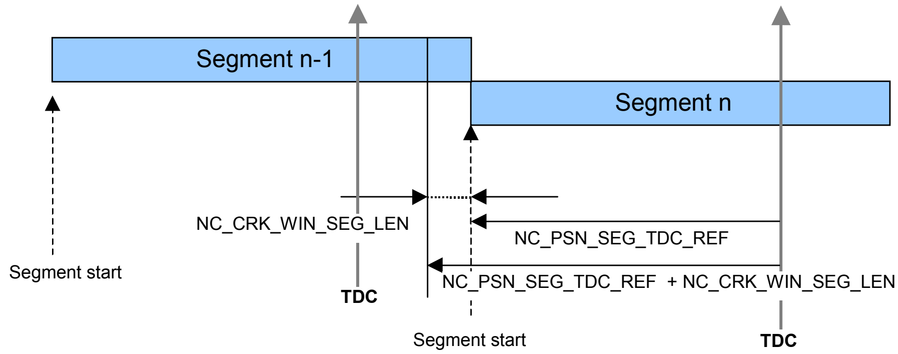
2.6.3.5 Crankshaft tooth window measurement (T_CRK_WIN_ENSD)

General information:

System request accuracy <= 1 us

The time interval elapsed between tooth events located between NC_CRK_WIN_SEG_LEN degrees before the end of each segment and the segment trigger is measured and send to the ASW segment synchronous

It provides and measures the time of the last NC_CRK_WIN_SEG_LEN teeth period before segment event.



Application conditions:

Initialisation: T_CRK_WIN_ENSD is set to the maximum value

Recurrence : every segment

Activation : LV_CRK_SYN = 1

Deactivation : LV_CRK_STOP = 1

Formula section:

If LV_CRK_SYN is set to 1 between NC_CRK_WIN_SEG_LEN degrees before the end of the segment and the segment trigger event

Then /* Recalculation from last available tooth period */

T_CRK_WIN = (T_TOOTH * NC_NR_TOOTH) / (360°CRK / NC_CRK_WIN_SEG_LEN)

Else /* Normal calculation */

T_CRK_WIN_ENSD = time interval between tooth events located between NC_CRK_WIN_SEG_LEN degrees before the end of each segment and the segment triger

Endif

2.6.3.6 ID_FAC_TOL_CRK_TOOTH calculation

General information:

ID_FAC_TOL_CRK_TOOTH[0rpm] is used until crankshaft synchronization (i.e. until first gap is detected, LV_CRK_SYN is set).

Then after crankshaft synchronization, ID_FAC_TOL_CRK_TOOTH is calculated depending on engine speed value (N_32).

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_T_CRK_DLY	1	0...29083H	0...40000	0.2380003	[µs]
Sensing delay time after first active crankshaft signal edge					
ID_FAC_TOL_CRK_TOOTH	6	0...FH	0...0.9375	0.0625	[-]
LDP_N_32_ID_FAC_TOL_CRK_TOOTH	6	0...FFH	0...8160	32	[rpm]
Factor for calculation of expected tooth period					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_SEG_HALF_END	-	0...1FE0H	0...8160	1	[rpm]
Engine speed threshold for half-segment trigger					
NC_CRK_WIN_SEG_LEN	-	0...FFH	0...95.625	0.375	[°CRK]
Width of crankshaft tooth window in °CRK at end of segment					
NC_PSN_SEG_TDC_REF	-	0...780H	0...720	0.375	[°CRK]
Segment start in °CRK before TDC					
NC_OFS_TDC0_REF_CRK	-	0...780H	0...720	0.375	[°CRK]
Reference gap position in °CRK before TDC0					
NC_PHA_SEG_ER_ENSD	-	0...FFH	0...95.625	0.375	[°CRK]
Misfire segment start in °CRK after TDC					
NC_NR_TOOTH_TOL_ADD	-	1...2H	1...2	1	[-]
Number of additional teeth tolerated between two reference gap occurrences					
NC_NR_TOOTH_TOL_MISS	-	1...2H	1...2	1	[-]
Number of missing teeth tolerated between two reference gap occurrences					
NC_NR_VLD_TOOTH	-	0...FH	0...15	1	[-]
Number of valid teeth necessary for validation of crankshaft rotation					
NC_NR_TOOTH	-	1...FFH	1...255	1	[-]
Theoretical number of teeth per crankshaft revolution					
NC_NR_TOOTH_GAP	-	1...2H	1...2	1	[-]
Number of missing teeth forming the reference gap					
NC_NR_TOOTH_STALL	-	2...7H	2...7	1	[-]
Number of consecutive missing teeth for engine stalling detection					
NC_N_MIN	-	0...1FE0H	0...8160	1	[rpm]
Minimum engine speed					
NC_N_MAX	-	0...3FC0H	0...16320	1	[rpm]
Maximum engine speed					
NC_ACT_CRK_EDGE	-	0H 1H	FALLING RISING	1	[-]
Active edge of crankshaft signal					

2.6.4 System requirements to sensor signal

2.6.4.1 General

With both sensor technologies, there is an angular offset between the mechanical reference (usually the trailing edge of a tooth) and the timing reference of the sensor (zero crossing of the MCPS signal, output switching of the ACPS) for an active edge. This offset will further on be referred to as phase angle. The phase angle depends strongly on sensor tolerances, installation tolerances (air gap), and operating conditions (speed, temperature).

The tolerances given in the following paragraphs refer to the angular position of the crankshaft target wheel when an active signal edge is recognized by the input port of the µP, or to the rotation angle between such events, respectively.

2.6.4.2 Relative phase angle tolerances

The usual synchronization algorithms can accept ±3% tolerance of the target wheel rotation angle between consecutive active signal edges.

e.g. for a "60-2" target wheel, this is 6° (18°) ±0.2°

The sensor/target wheel combination has to fulfill this requirement at all operating conditions.

The permissible accumulation of tooth angle tolerances over a TDC segment is ±0.25% of the angle for one TDC segment (720°/ NC_CYL_NR) at segment reference conditions.

e.g. for four-cylinder engines 180° ±0.45°

e.g. for six-cylinder engines 120° ±0.3°

Segment reference conditions are: 2000rpm, 80°C.

Please note that for engines with more than 8 cylinders, this requirement is more stringent than the requirement for consecutive signal edges.

2.6.4.3 Relative phase angle stability

The angle for one TDC period measured under segment reference conditions must not have a drift > 0.045° versus speed variation from 2000 rpm...7000 rpm and temperature variation within the complete range. This requirement has to be fulfilled for all tolerances and all possible displacements of the target wheel relative to the sensor (air gap, radial run-out, crankshaft flexion, bearing clearance...).

In case of MCPS sensors, phase angle stability must be measured with the ECU hardware stage.

2.6.4.4 Repeatability

Two different definitions of repeatability are functionally equal. The choice depends on the sensor test bench and measurement equipment.

1. Phase angle repeatability

(mechanical edge to signal edge for the same tooth) ±0.025°

2. Interval repeatability

(signal edge to signal edge for the same pair of teeth) ±0.05°

In both cases the repeatability is determined from the maximum and minimum value out of a large number of measurements under constant conditions.

2.6.5 Hardware and HAL requirements

2.6.5.1 EMI

Any signal disturbance must be suppressed as far as possible. Supplemental trigger events must not be generated due to such disturbance, particularly not at the slow zero crossing in the reference gap.

2.6.5.2 Phase shift repeatability

At constant operating conditions, the phase shift introduced by the input circuitry must be constant within ±0.1 µs

2.7 Acquisition of Camshaft Signal for single-tooth Target Wheel

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_SEG_CAM_IN_i	O/V	0...7FFFFFH	0...1.9965	2.38e-7	s
Camshaft segment period					
T_SEG_CAM_EX_i	O/V	0...7FFFFFH	0...1.9965	2.38e-7	s
Camshaft segment period					
REL_ANG_CAM_REF_GAP	-	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Measured angular distance between reference gap and camshaft edge					
REL_ANG_CRK_CAM	-	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Measured angular distance between consecutive active camshaft edges					
PSN_TOOTH1_CAM	-	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Theoretical angular distance between crankshaft tooth #1 and camshaft edge					
PSN_CAM_CAM	-	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Theoretical angular distance between consecutive active camshaft edges					
ST_CAM_SS	-	0..05H	0..5	1	-
Camshaft self synchronization state					
ST_CAM_CRK_S	-	0..04H	0..4	1	-
Cam crank synchronization state					
ST_CAM_PRE_INJ_S	-	0..03H	0..3	1	-
Cam crank synchronization state					
LV_SYN_CAM_IN_i	O/V	0...FFH	0...255	1	-
Intake camshaft synchronized					
LV_SYN_CAM_EX_i	O/V	0...FFH	0...255	1	-
Exhaust camshaft synchronized					
CTR_EDGE_CAM_IN_i	O/V	0...FFH	0...255	1	-
Continuous camshaft signal edge counter					
CTR_EDGE_CAM_EX_i	O/V	0...FFH	0...255	1	-
Continuous camshaft signal edge counter					
IDX_EDGE_CAM_IN_i	O/V	0...FH	1...16	1	-
Index of the last camshaft signal edge					
IDX_EDGE_CAM_EX_i	O/V	0...FH	1...16	1	-
Index of the last camshaft signal edge					
RATIO_PER_CAM_IN_i	-	0...FFFFH	0.0039...255.97	0.0039	-
Camshaft period ratio for synchronization					
RATIO_PER_CAM_EX_i	-	0...FFFFH	0.0039...255.97	0.0039	-
Camshaft period ratio for synchronization					
LV_ORNG_PER_CAM_IN_i	O/V	0...1H	0...1	1	-
Intake camshaft segment period out of range					
LV_ORNG_PER_CAM_EX_i	O/V	0...1H	0...1	1	-
Exhaust camshaft segment period out of range					
LV_ORNG_RATIO_CAM_IN_i	O/V	0...1H	0...1	1	-
Intake camshaft segment ratio out of range					
LV_ORNG_RATIO_CAM_EX_i	O/V	0...1H	0...1	1	-
Exhaust camshaft segment ratio out of range					
LV_VLD_PSN_CAM_IN_i	O/V	0..1H	0...1	1	-
Intake camshaft position measurement valid					
LV_VLD_PSN_CAM_EX_i	O/V	0...1H	0...1	1	-
Exhaust camshaft position measurement valid					
LV_CAM_STOP_IN_i	O	0..1H	0...1	1	-
Self synchronization on Intake camshaft i is stopped					
LV_CAM_STOP_EX_i	O	0...1H	0...1	1	-
Self synchronization on Exhaust camshaft i is stopped					
PSN_ENG_CRK_OFS	O	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Engine position offset for initialization at crankshaft synchronization.					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CAM_SYN_CRK	O/V	0...1H	0...1	1	-
Camshaft acquisition ready for crankshaft synchronization.					
LV_ORNG_CAM_SYN_CRK	O/V	0...1H	0...1	1	-
Camshaft signal for crankshaft synchronization out of range					
RATIO_PSN_EDGE_z_CAM_IN_i	-	0...FFFFH	0.0039...255.97	0.0039	-
Theoretical intake camshaft i period ratio at edge z					
RATIO_PSN_EDGE_z_CAM_EX_i	-	0...FFFFH	0.0039...255.97	0.0039	-
Theoretical exhaust camshaft i period ratio at edge z					
PSN_ENG_SYN_CAM_MIN	O	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Minimum engine position during crankshaft synchronization phase.					
PSN_ENG_SYN_CAM_MAX	O	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Maximum engine position during crankshaft synchronization phase.					
CAM_DYW_SYN_IN	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window for angle between intake camshaft signal edges in crankshaft synchronization mode					
CAM_DYW_SYN_EX	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window for angle between exhaust camshaft signal edges in crankshaft synchronization mode					
CAM_DYW_CRK_SYN_ADC_IN	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window advance for intake camshaft to crankshaft reference in crankshaft synchronization mode					
CAM_DYW_CRK_SYN_ADC_EX	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window advance for exhaust camshaft to crankshaft reference in crankshaft synchronization mode					
CAM_DYW_CRK_SYN_RTD_IN	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window retard for intake camshaft to crankshaft reference in crankshaft synchronization mode					
CAM_DYW_CRK_SYN_RTD_EX	-	0...7FFH	0...127.9375	0.0625	[°CRK]
Tolerance window retard for exhaust camshaft to crankshaft reference in crankshaft synchronization mode					

Input data:

LV_CRK_FIRST_VLD_TOOTH	LV_CRK_SYN	NC_NR_TOOTH	PSN_ENG_CRK
NC_OFS_TDC0_REF_CRK	N_32	LV_ACT_SYN_CRK_CAM_IN_i	LV_ACT_SYN_CRK_CAM_EX_i
LV_ACT_CAM_IN_i	LV_ACT_CAM_EX_i	NC_NR_GAP	NC_CAM_SENS_TYP
LV_CAM_LOCK_IVVT_IN_i	LV_CAM_LOCK_IVVT_EX_i	NLC_IVVT_IN	NLC_IVVT_EX
LV_STOP_ENG			CONF_CAM_VVT_EX

Import actions:

ACTION_ENSD_*GetDigCAMINLevel* (IN <STATE_CAM_IN>)
ACTION_ENSD_*GetDigCAMEXLevel* (IN <STATE_CAM_EX>)

General information:

This specification applies for cam/crk configurations with:

- up to 4 camshaft sensors
- active camshaft sensors (ACAM) with TPO or without TPO
- magnetic camshaft sensors (MCAM)
- halfmoon or pin-type camshaft target wheels
- one reference gaps per engine revolution (NC_NR_GAP = 1)

The active camshaft signal edge(s) for synchronization can be selected by configuration data NC_ACT_CAM_EDGE_SYN:

- NC_ACT_CAM_EDGE_SYN = 1 or 2: no level detection is performed, only camshaft edges (falling or rising) are evaluated. This configuration shall be selected when a ACAM non – TPO or a MCAM sensor is used.
- NC_ACT_CAM_EDGE_SYN = 3: level detection at engine start, falling and rising camshaft edges are evaluated. This configuration should be selected when a ACAM TPO sensor is used.

A diagnostic output is delivered to allow detection of camshaft signal failure.

The signal is used for:

1. Self synchronization of all camshafts : timing validation of the active edges of the signal in correspondance with its theoretic position (used for VVT controler, for detection of crankshaft signal failure, and for limp-home in case of crankshaft signal failure).
2. Camshaft / crankshaft synchronization of the selected camshaft : determination of the offset for the engine position calculation
3. Engine position interface for pre-injection (with the selected camshaft used for cam/crank synchronization)

These three functions are described with three states diagrams.

Application conditions:

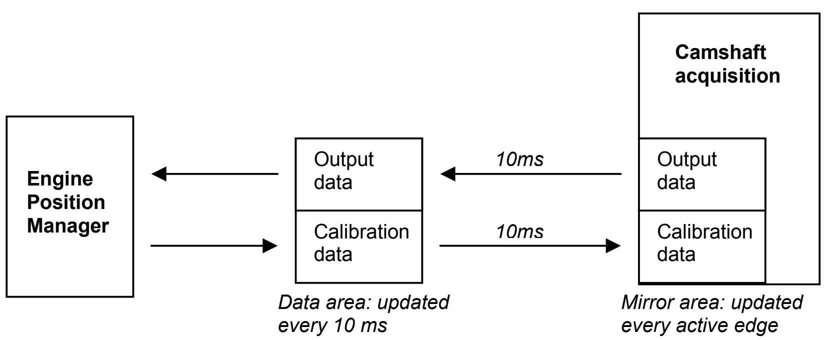
Recurrence:

For set of the RAM as flags, a mirror area is defined which contains a copy of the output and calibration data in the basic software area (see figure below).

The mirror area of the output data is updated by the lower layer at every active signal edge.

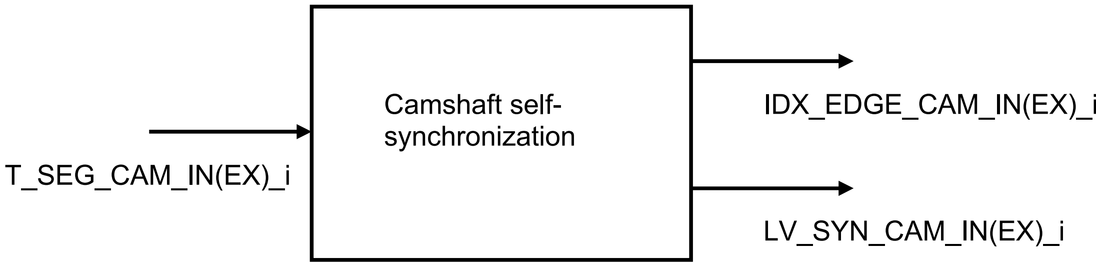
The mirror area of the calibration data is updated by the upper layer each 10ms.

The output data is copied from the mirror area every 10 ms



2.7.1 Camshaft Self Synchronization

General information:



Self-synchronization is performed on every available camshaft sensor. It's purpose is to determine the current camshaft signal index IDX_EDGE_CAM_IN(EX)_i based on measured camshaft segment periods T_SEG_CAM_IN(EX)_i (see requirements to infrastructure).

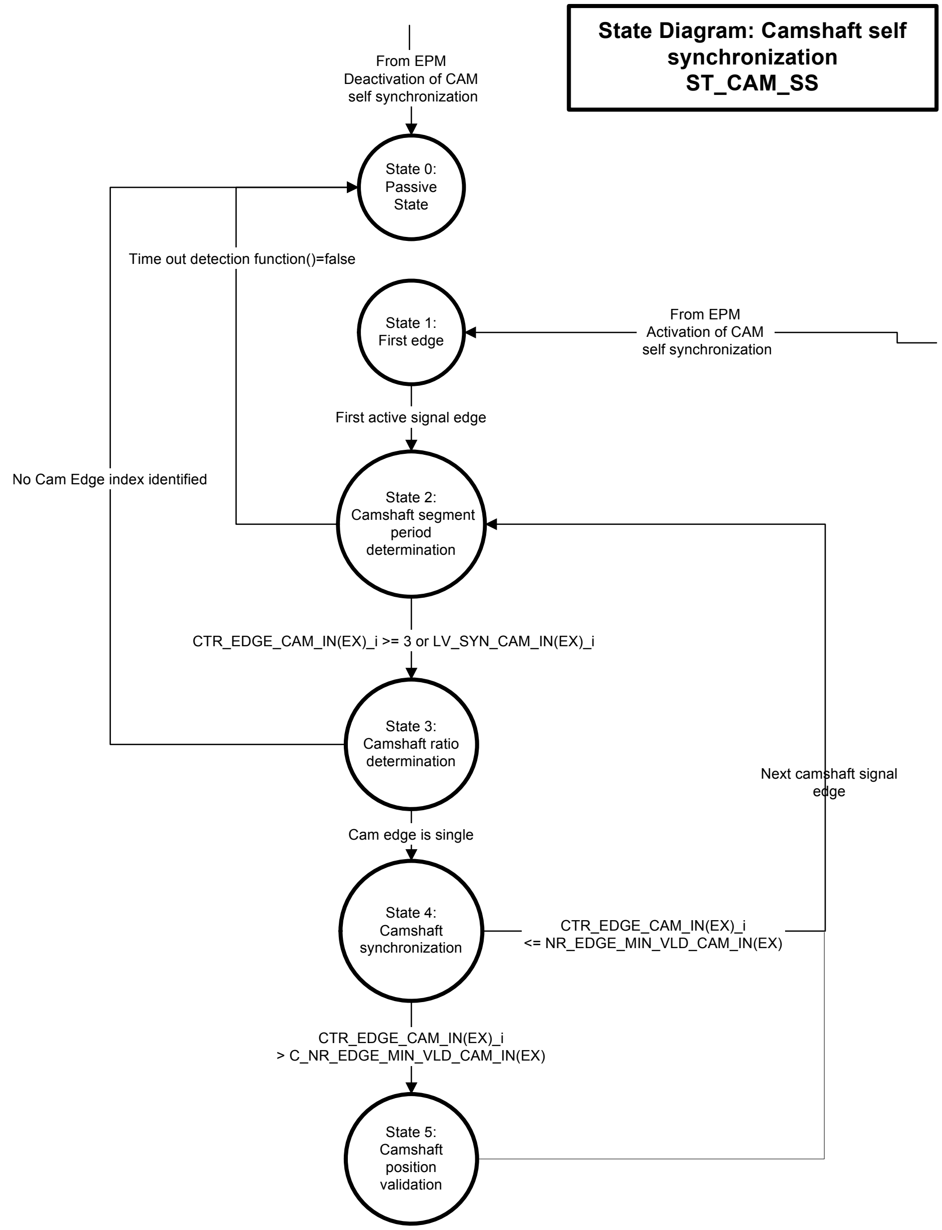
Note that there are two lists of possible camshaft signal indexes:

- For self-synchronization (described in this chapter)
- For cam/crk synchronization and engine position determination for pre-injection

Application conditions:

Recurrence: every camshaft signal edge (falling and rising)

Signal flow diagram ST_CAM_SS



Formula section:

Passive state : state 0

Input condition :

From EPM :	Desactivation of CAM self synchronization
From state 2 :	Time out detection function () = False
From state 3 :	No cam edge index identified

Output condition :

No output conditions

Action in the state :

LV_CAM_STOP_IN_i = LV_CAM_STOP_EX_i = 1
T_SEG_CAM_IN_i = NC_T_SEG_MAX_CAM_IN
T_SEG_CAM_EX_i = NC_T_SEG_MAX_CAM_EX
LV_SYN_CAM_IN_i = LV_SYN_CAM_EX_i = 0
LV_VLD_PSN_CAM_IN_i = LV_VLD_PSN_CAM_EX_i = 0
CTR_EDGE_CAM_IN_i = CTR_EDGE_CAM_EX_i= 0
IDX_EDGE_CAM_IN_i = IDX_EDGE_CAM_EX_i = 1
RATIO_PER_CAM_IN_(EX)_i = 0
RATIO_PSN_EDGE_z_CAM_IN(EX)_i= 0

Action in transient:

No actions

First edge : state 1

Input condition:

From EPM: Activation of CAM self synchronization

Output condition:

To state 2 First active signal edge

Action in the state:

LV_CAM_STOP_IN(EX)_i = 0
LV_ORNG_PER_CAM_IN(EX)_i = 0
LV_ORNG_RATIO_CAM_IN(EX)_i = 0

The index of the camshaft edge leading to the camshaft signal level will be determined (p.e. if the signal level is high, index correponding to a rising signal edges).

Wait for first active signal edge

Action in transient:

No actions

Camshaft segment period determination : State 2

Input condition :

From state 1: First active signal edge occured

From state 4: CTR_EDGE_CAM_IN(EX)_i <= C_NR_EDGE_MIN_VLD_CAM_IN(EX)

And next camshaft active edge

From state 5: next camshaft active edge

Output condition :

To state 0: Time out detection function () = False

To state 3: CTR_EDGE_CAM_IN(EX)_i >= 3
Or LV_SYN_CAM_IN(EX)_i = 1

Action in the state:

If Signal de-glitching () = true

Then LV_ORNG_PER_CAM_IN(EX)_i = 1

Else Activation of Time out detection function ()

Increment CTR_EDGE_CAM_IN(EX)_i

Index of the possible cam edges table is incremented by one modulo 2.

Endif

Action in transient:

No actions

<u>Camshaft ratio determination : State 3</u>

Input condition :

From state 2: CTR_EDGE_CAM_IN(EX)_i >= 3

Or

LV_SYN_CAM_IN(EX)_i = 1

Output condition :

To state 0: No Cam Edge index identified

To state 4: Cam Edge index is single

Action in the state :

Activation of Camshaft segment ratio calculation function ()

Activation of Camshaft edge recognition function ()

Action in transient :

To state 0: LV_ORNG_RATIO_CAM_IN(EX)_i = 1

To state 4: LV_SYN_CAM_IN(EX)_i = 1

<u>Camshaft synchronization : State 4</u>

Input condition :

From state 3: Cam Edge index is single

Output condition :

To state 2: CTR_EDGE_CAM_IN(EX)_i <= C_NR_EDGE_MIN_VLD_CAM_IN(EX)

To state 5: CTR_EDGE_CAM_IN(EX)_i > C_NR_EDGE_MIN_VLD_CAM_IN(EX)

Action in the state: IDX_EDGE_CAM_IN(EX)_i = Cam Edge index

Wait next camshaft active edge

Action in transient:

To state 5: LV_VLD_PSN_CAM_IN(EX)_i = 1

<u>Validation of camshaft position : State 5</u>

Input condition:

From state 4: CTR_EDGE_CAM_IN(EX)_i > C_NR_EDGE_MIN_VLD_CAM_IN(EX)

Output condition:

To state 2: Next camshaft active edge

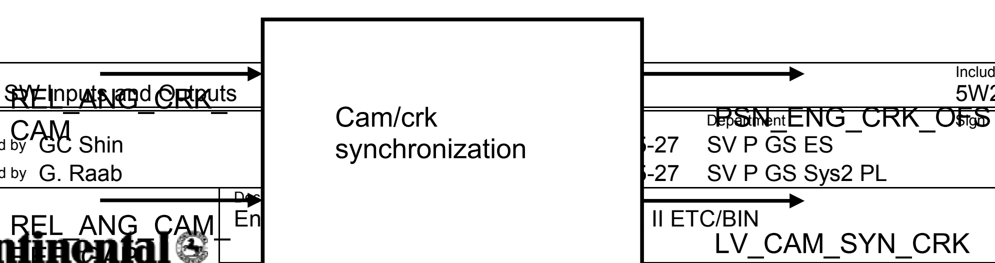
Action in the state: Wait next camshaft active edge

Action in transient:

None

2.7.2 Camshaft/Crankshaft Synchronization

<u>General information:</u>



Cam/crk-synchronization is performed on the camshaft sensor which is selected for synchronization by the engine position manager. It's purpose is to determine the engine position offset PSN_ENG_CRK_OFS based on measured angular distances REL_ANG_CRK_CAM and REL_ANG_CAM_REF_GAP (see requirements to infrastructure).

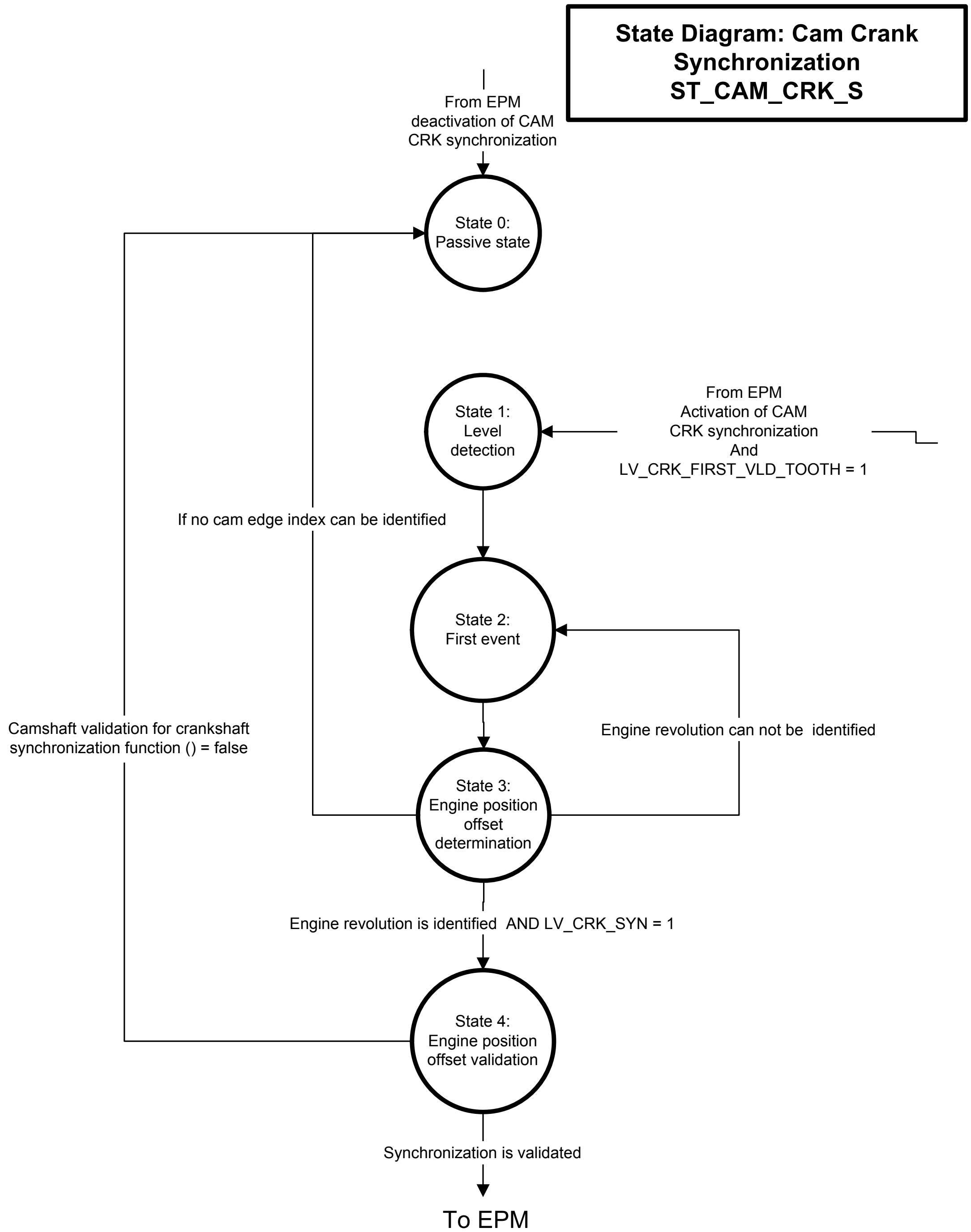
The engine position offset is

- 360° or 720° for NC_NR_GAP = 1

Note that there are two lists of possible camshaft signal indexes:

- For self-synchronization
- For cam/crk synchronization and engine position determination for pre-injection (described in this chapter)

Signal flow diagram:



Application conditions:

Recurrence: every falling camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 1

every rising camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 2

every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

Formula section:

Passive state : state 0

Input condition :

From EPM : Deactivation of CAM CRK synchronization

Output condition :

No output conditions

Action in the state : PSN_ENG_CRK_OFS = 0

LV_CAM_SYN_CRK = 0

PSN_ENG_SYN_CAM_MIN = 0°

PSN_ENG_SYN_CAM_MAX = 720°

Activation of VVT lock check function ()

Action in transient :

No actions

Level detection : state 1

Input condition:

From EPM : Activation of CAM CRK synchronization

And

LV_CRK_FIRST_VLD_TOOTH = 1

Output condition :

To state 2: At the end of the action in the state

Action in the state :

LV_ORNG_CAM_SYN_CRK = 0

If NC_ACT_CAM_EDGE_SYN = 1
Then Keep falling signal edge: Index list = {1}
Elseif NC_ACT_CAM_EDGE_SYN = 2
Then Keep rising signal edge: Index list = {2}
Else Level detection:
  If camshaft signal level is high
  Then Keep rising signal edge: Index list = {2}
  Else Keep falling signal edge: Index list = {1}
  Endif
Endif

Action in transient :

No actions

First event : state 2

Input condition :

From state 1: At the end of the action in the state 1

From state 3: Engine revolution can not be identified

Output condition :

To state 3: Every camshaft edge or every crankshaft reference gap.

Action in the state :

Wait cam edge or crankshaft reference gap

Action in transient :

No actions

Engine position offset determination : state 3

Input condition :

From state 2 : Camshaft edge or crankshaft reference gap

Output condition :

To state 4 : Engine revolution is identified AND LV_CRK_SYN = 1.

To state 2 : Engine revolution can not be identified

To state 0 : If no cam edge index can be identified.

Action in the state :

Camshaft edge index determination function ()

Action in transient :

To state 4 : LV_CAM_SYN_CRK = 1

To state 0 : LV_ORNG_CAM_SYN_CRK = 1

<u>Engine position offset validation : state 4</u>

Input condition :

From state 3 : Engine revolution is identified AND LV_CRK_SYN = 1

Output condition :

To state 0 : Camshaft validation for crankshaft synchronization function () = false

To EPM: Synchronization is validated by EPM

Action in the state :

Activation of Camshaft validation for crankshaft synchronization function ()

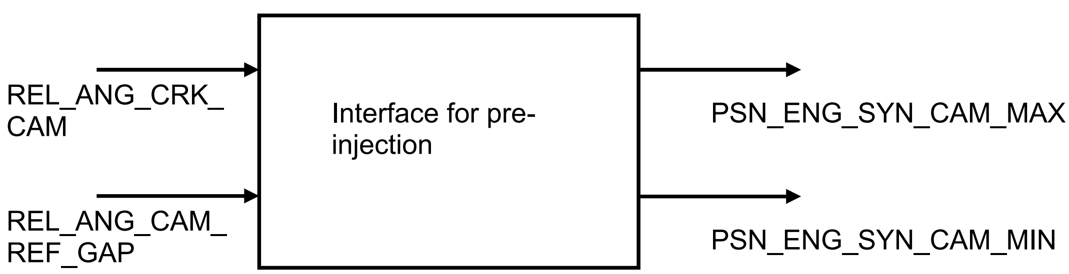
Action in transient :

To state 0 : LV_CAM_SYN_CRK is reset

LV_ORNG_CAM_SYN_CRK = 1

2.7.3 Engine Position Interface for Pre-Injection

<u>General information:</u>



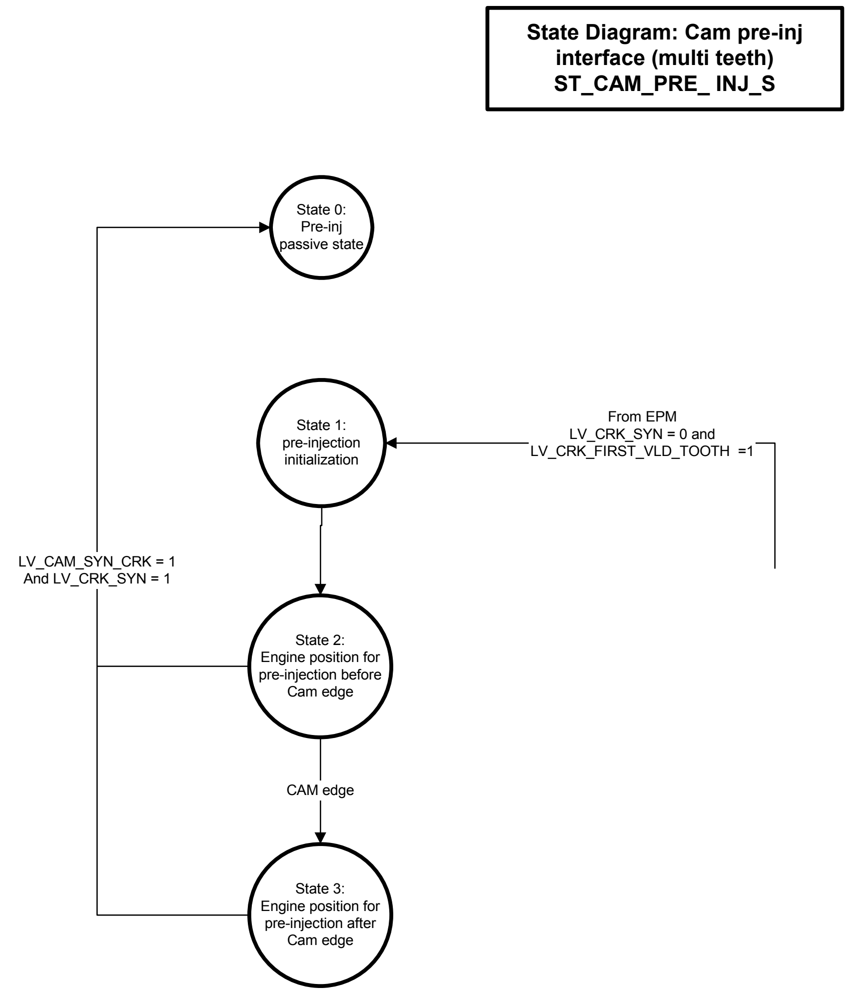
For rapid start of port-injection engines it is useful to start injection before synchronization on crankshaft signal is achieved. An approximative information about engine position is necessary for phasing the injection in a way to avoid emissions increase.

For this purpose, a minimum and the maximum engine position is determined according to the actual possible camshaft signal edge indexes found by camshaft signal acquisition for crankshaft synchronization. The minimum and the maximum position will converge gradually during the synchronization process.

Application conditions:

Recurrence:	every falling camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 1
	every rising camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 2
	every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

Signal flow diagram:



Formula section:

Engine position for pre-injection passive state : State 0

Input condition :

From state 2,3 : LV_CRK_SYN = 1 and LV_CAM_SYN_CRK = 1

Output condition :

No conditions

Action in the state :

No actions

Action in transient :

No actions

Engine position for pre-injection initialization state : State 1

Input condition :

From EPM : LV_CRK_SYN = 0 and LV_CRK_FIRST_VLD_TOOTH =1

Output condition :

To state 2: End of the state 1

Action in the state :

If NC_ACT_CAM_EDGE_SYN = 1

Then Keep initialization values for
PSN_ENG_SYN_CAM_MIN(MAX)

Elseif NC_ACT_CAM_EDGE_SYN = 2

Then Keep initialization values for
PSN_ENG_SYN_CAM_MIN(MAX)

Else

If CONF_CAM_VVT_EX = 0

PSN_ENG_SYN_CAM_MIN =

NC_PSN_EDGE_z_CAM_IN(EX)_i –

CAM_DYW_SYN_IN(EX)

for the lowest possible signal edge index z

PSN_ENG_SYN_CAM_MAX =

NC_PSN_EDGE_z+1_CAM_IN(EX)_i +

CAM_DYW_SYN_IN(EX)

for the next highest possible signal edge index z+1

Elseif CONF_CAM_VVT_EX = 1

ACTION_ENSD_GetCAMINLevel(IN<STATE_CAM_IN>)

ACTION_ENSD_GetCAMEXLevel(IN<STATE_CAM_EX>)

If [(STATE_CAM_IN = 0 **and** STATE_CAM_EX = 1)

or

(STATE_CAM_IN = 1 **and** STATE_CAM_EX = 0)**]**

PSN_ENG_SYN_CAM_MIN =
NC_PSN_EDGE_z_CAM_EX_i –
CAM_DYW_SYN_EX

for the lowest possible signal edge index z

PSN_ENG_SYN_CAM_MAX =
NC_PSN_EDGE_z+1_CAM_IN_i +
CAM_DYW_SYN_IN

for the next highest possible signal edge index z+1

Endif

If (STATE_CAM_IN = 1 **and** STATE_CAM_EX = 1)

or

If (STATE_CAM_IN = 0 **and** STATE_CAM_EX = 0)

PSN_ENG_SYN_CAM_MIN =
NC_PSN_EDGE_z_CAM_IN_i – CAM_DYW_SYN_IN

for the lowest possible signal edge index z

PSN_ENG_SYN_CAM_MAX =
NC_PSN_EDGE_z+1_CAM_EX_i +
CAM_DYW_SYN_EX

for the next highest possible signal edge index z+1

Endif

Endif

Endif

Action in transient :

LV_ANG_EVT = 0

Engine position for pre-injection before cam edge state : State 2

Input condition :

From state 1: End of the state 1

Output condition :

To state 3 : First cam edge found

To state 0 : LV_CRK_SYN = 1 and LV_CAM_SYN_CRK = 1

Action in the state :

Activation of Pre-injection interface computation before cam edge function ()

Action in transient :

To state 0 : PSN_ENG_SYN_CAM_MIN= PSN_ENG_SYN_CAM_MAX = PSN_ENG_CRK

Engine position for pre-injection after cam edge state : State 3

Input condition :

From state 2: First cam edge found

Output condition :

To state 0 : LV_CRK_SYN = 1 and LV_CAM_SYN_CRK = 1

Action in the state :

Activation of Pre-injection interface computation after Cam edge function ()

Action in transient :

To state 0 : PSN_ENG_SYN_CAM_MIN= PSN_ENG_SYN_CAM_MAX = PSN_ENG_CRK

2.7.4 Definition of the sub function task

2.7.4.1 Time out detection function ()

General information:

Time-out is detected when the segment period exceeds NC_T_SEG_MAX_CAM_IN(EX).
Time out detection is used for engine stalling detection during crankshaft limp-home.

Formula section:

If T_SEG_CAM_IN(EX)_i <= NC_T_SEG_MAX_CAM_IN(EX)

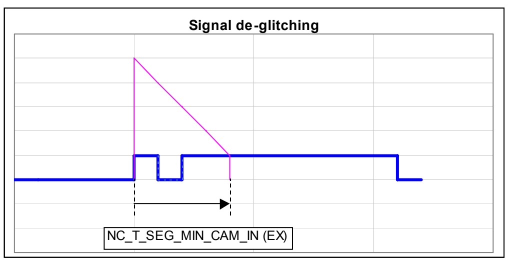
Then return (true)

Else return(false)

Endif

2.7.4.2 Signal de-glitching ()

Any signal edge detected before a delay time NC_T_SEG_MIN_CAM_IN respectively NC_T_SEG_MIN_CAM_EX following the previous signal edge will not produce any output.



Formula section:

If a glitch occurs during the delay NC_T_SEG_MIN_CAM_IN(EX)

Then Return (true)

Else Return (false)

Endif

2.7.4.3 Camshaft segment ratio calculation function ()

General information:

The computation of the ratio will be compared to the theoric ratio.

Application conditions:

Recurrence : every camshaft active edge.

Measured ratio :

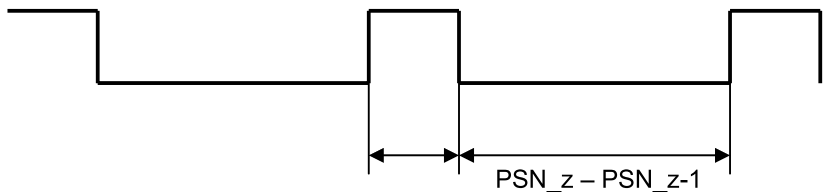
RATIO_PER_CAM_IN(EX)_i =

T_SEG_CAM_IN(EX)_i_n / T_SEG_CAM_IN(EX)_i_{n-1}

Theoric ratio : for all remaining indexes z

RATIO_PSN_EDGE_z_CAM_IN(EX)_i =

(NC_PSN_EDGE_z_CAM_IN(EX)_i - NC_PSN_EDGE_(z-1)_CAM_IN(EX)_i) /
(NC_PSN_EDGE_(z-1)_CAM_IN(EX)_i - NC_PSN_EDGE_(z-2)_CAM_IN(EX)_i)



2.7.4.4 Camshaft edge recognition ()

General information:

The algorithm has to determine the index of the actual signal edge corresponding to the position of the camshaft by checking the measured ratio and the theoric ratio.

Application conditions:

Recurrence: every camshaft active edge.

Formula section:

For all remaining indexes z:

If RATIO_PSN_EDGE_z_CAM_IN(EX)_i * ID_FAC_CAM_IN(EX) > RATIO_PER_CAM_IN(EX)_i > RATIO_PSN_EDGE_z_CAM_IN(EX)_i / ID_FAC_CAM_IN(EX)

Then keep index z in the table

Else eliminate index z from the table

Endif

2.7.4.5 Camshaft edge index determination function ()

General information:

This function determines the possible camshaft signal indexes and returns the engine position offset (0° or 360°) when index is identified.

The range of the local variables alpha and beta is:

Alpha: -720° CRK to +720° CRK

Beta: 0° CRK to +720° CRK

Limit: 0° CRK to +720° CRK

Application conditions:

Recurrence: every falling camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 1

every rising camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 2

every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

Formula section:

Calculate theoretical angular distance between two successive active camshaft edges:

If NC_ACT_CAM_EDGE_SYN = 3

Then F*alling and rising camshaft edges*

PSN_CAM_CAM = NC_PSN_EDGE_z_CAM_IN(EX)_i –

NC_PSN_EDGE_z-1_CAM_IN(EX)_i

Else *Only Falling or only rising camshaft edges*

PSN_CAM_CAM = NC_PSN_EDGE_z_CAM_IN(EX)_i –

NC_PSN_EDGE_z-2_CAM_IN(EX)_i = 720° CRK

Endif

Tests (depending on actual event and history) are done for each index z of the possible indexes table. If the test is false then the corresponding index is eliminated.

The following table shows the actions to perform in function of the current (n) and last event (n-1).

The event is a detection of Camshaft edge (CAM) or a detection of reference gap (GAP).

Tests are done with the current value of REL_ANG_CRK_CAM.

The EPM initialize PSN_ENG_CRK with 360° - NC_OFS_TDC0_REF_CRK (engine revolution 1)

Event (n)	CAM	CAM	GAP	GAP	CAM	GAP
Event (n-1)	None	CAM	None	CAM	GAP	GAP
Action / Test	Action 1 Test 1	Action 1 Test 2	Tests 3	Test 4	Action 1 Test 5	Test 3

Action 1:

This action should be done before the test.

If NC_ACT_CAM_EDGE_SYN = 3

Then Increment indexes in the list of possible camshaft signal index by 1 modulo 2.

Else (no operation)

Endif

Test 1:

Beta = PSN_CAM_CAM
+ CAM_DYW_SYN_IN(EX)

If Beta < 0°

Then Beta = Beta + 720°

Endif

Keep all indexes for which the following condition is true :

REL_ANG_CRK_CAM < Beta

Test 2:

Alpha = PSN_CAM_CAM - CAM_DYW_SYN_IN(EX)

Beta = PSN_CAM_CAM + CAM_DYW_SYN_IN(EX)

If Beta < 0°

Then Alpha = Alpha + 720°
Beta = Beta + 720°

Endif

If Alpha < 0°

Then (no operation)

Endif

Keep all indexes for which the following condition is true:

Alpha < REL_ANG_CRK_CAM < Beta

Test 3:

Test 3A:

Beta = 360° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i
+ CAM_DYW_CRK_SYN_ADC_IN(EX)

If Beta < 0°

Then Beta = Beta + 720°

Endif

If REL_ANG_CRK_CAM < Beta < Limit

Then PSN_ENG_CRK_OFS_TMP=0

Endif

Test 3B:

Beta = 720° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i
+ CAM_DYW_CRK_SYN_ADC_IN(EX)

If Beta < 0°

Then Beta = Beta + 720°

Endif

If **REL_ANG_CRK_CAM** < Beta < Limit

Then PSN_ENG_CRK_OFS_TMP= 360

Endif

Limit = Maximum angle between two active camshaft edges:

NC_ACT_CAM_EDGE_SYN = 3 and half-moon target wheel:

Limit = 360° CRK

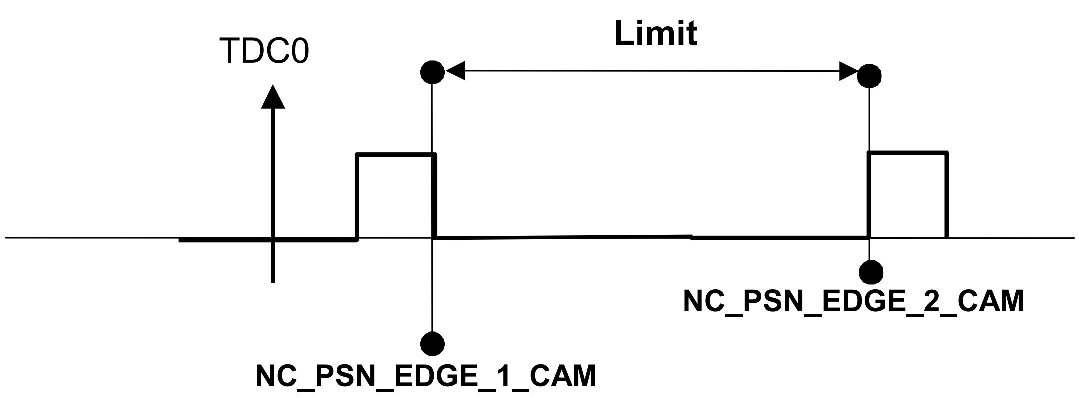
NC_ACT_CAM_EDGE_SYN = 3 and pin-type target wheel:

360° CRK < Limit < 720° CRK

NC_ACT_CAM_EDGE_SYN = 1 or 2 (non-TPO sensors):

Limit = 720° CRK

For example: pin-type target wheel, NC_ACT_CAM_EDGE_SYN=3:



If **Test 3A** And **Test 3B** are true

Or

NC_CAM_SENS_TYP = 0 (MCAM sensor)

Then The engine revolution can not be identified

*(comment: identical camshaft signal level on two successive reference gaps **and** measured ang. distance < theoretical ang. distance between camshaft edge #z and reference gap **or** MCAM sensor)*

Else **if** **Test 3A** And **Test 3B** are false

Then Index z should be eliminated

*(comment: different camshaft signal level on two successive reference gaps **and** measured ang. distance > theoretical ang. distance between camshaft edge #z and reference gap)*

Else PSN_ENG_CRK_OFS = PSN_ENG_CRK_OFS_TMP

The engine revolution is identified

*(comment: different camshaft signal level on two successive reference gaps **and** measured ang. distance < theoretical ang. distance between camshaft edge #z and reference gap **or** identical camshaft signal level on two successive reference gaps **and** measured ang. distance > theoretical ang. distance between camshaft edge #z and reference gap)*

Endif

Endif

Test 4:

Alpha = 360° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i -CAM_DYW_CRK_SYN_RTD_IN(EX)

Beta = 360° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i + CAM_DYW_CRK_SYN_ADC_IN(EX)

If Beta < 0°

Then Alpha = Alpha + 720°

Beta = Beta + 720°

Endif

If Alpha < 0°

Then (no operation)

Endif

If Alpha < REL_ANG_CRK_CAM < Beta

Then PSN_ENG_CRK_OFS = 0°

Engine revolution identified

Else

Alpha = 720° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i

-CAM_DYW_CRK_SYN_RTD_IN(EX)

Beta = 720° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i

+ CAM_DYW_CRK_SYN_ADC_IN(EX)

If Beta < 0°

Then Alpha = Alpha + 720°

Beta = Beta + 720°

Endif

If Alpha < 0°

Then (no operation)

Endif

if Alpha < REL_ANG_CRK_CAM < Beta

Then PSN_ENG_CRK_OFS = 360°

Engine revolution identified

Else Index z should be eliminated

Endif

Endif

Test 5:

Alpha = NC_PSN_EDGE_z_CAM_IN(EX)_i - (360° - NC_OFS_TDC0_REF_CRK)

- CAM_DYW_CRK_SYN_ADC_IN(EX)

Beta = NC_PSN_EDGE_z_CAM_IN(EX)_i - (360° - NC_OFS_TDC0_REF_CRK)

+ CAM_DYW_CRK_SYN_RTD_IN(EX)

If Beta < 0°

Then Alpha = Alpha + 720°

Beta = Beta + 720°

Endif

If Alpha < 0°

Then (no operation)

Endif

If Alpha < REL_ANG_CAM_REF_GAP < Beta

Then PSN_ENG_CRK_OFS = 0°

Engine revolution identified

Else

Alpha = NC_PSN_EDGE_z_CAM_IN(EX)_i - (720° - NC_OFS_TDC0_REF_CRK)

- CAM_DYW_CRK_SYN_ADC_IN(EX)

Beta = NC_PSN_EDGE_z_CAM_IN(EX)_i - (720° - NC_OFS_TDC0_REF_CRK)
+ CAM_DYW_CRK_SYN_RTD_IN(EX))

If Beta < 0°
Then Alpha = Alpha + 720°
Beta = Beta + 720°
Endif
If Alpha < 0°
Then (no operation)
Endif

if Alpha < REL_ANG_CAM_REF_GAP < Beta
Then PSN_ENG_CRK_OFS = 360°
Engine revolution identified
Else Index z should be eliminated
Endif

Endif

2.7.4.6 Camshaft validation for crankshaft synchronization function ()

General information:

The purpose of this function is to check the angular distance between cam/crk events. The function returns false is the active camshaft signal edge z is outside the defined tolerance window.

Application conditions:

Recurrence: every falling camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 1

every rising camshaft signal edge and every reference gap if NC_ACT_CAM_EDGE_SYN = 2

every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

Formula section:

Calculate theoretical angular distance between two successive active camshaft edges:

If NC_ACT_CAM_EDGE_SYN = 3
Then F*alling and rising camshaft edges*
PSN_CAM_CAM = NC_PSN_EDGE_z_CAM_IN(EX)_i –
NC_PSN_EDGE_z-1_CAM_IN(EX)_i
Else *Only Falling or only rising camshaft edges*

PSN_CAM_CAM = NC_PSN_EDGE_z_CAM_IN(EX)_i –

NC_PSN_EDGE_z-2_CAM_IN(EX)_i = 720° CRK

Endif

The following table shows the actions to perform in function of the current (n) and last event (n-1).

The event is a detection of Camshaft edge (CAM) or a detection of reference gap (GAP).

Tests are done with the current value of REL_ANG_CRK_CAM.

Event (n)	CAM	GAP	CAM	GAP	GAP	GAP
Event (n-1)	CAM	CAM	GAP	GAP	GAP	GAP
Event (n-2)				GAP	None	CAM
Action / Test	Action 1 Test 1	Test 2	Action 1 Test 3	Test 4	None	none

Action 1

This action should be done before the test.

If NC_ACT_CAM_EDGE_SYN = 3

Then Increment indexes in the list of possible camshaft signal index by 1 modulo 2.

Else (no operation)

Endif

Test 1

Alpha = PSN_CAM_CAM - CAM_DYW_SYN_IN(EX)

Beta = PSN_CAM_CAM + CAM_DYW_SYN_IN(EX)

If Beta < 0°

Then Alpha = Alpha + 720°

Beta = Beta + 720°

Endif

If Alpha < 0°

Then (no operation)

Endif

If Alpha < REL_ANG_CRK_CAM < Beta

Then return (true)

```
        Else        return (false)
        Endif
```

Test 2

```
    Alpha = 360° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i
                        - CAM_DYW_CRK_SYN_RTD_IN(EX)
    Beta = 360° - NC_OFS_TDC0_REF_CRK  - NC_PSN_EDGE_z_CAM_IN(EX)_i
                        + CAM_DYW_CRK_SYN_ADC_IN(EX)
    If          Beta < 0°
    Then        Alpha = Alpha + 720°
                Beta = Beta + 720°
    Endif
    If          Alpha < 0°
    Then        (no operation)
    Endif

    If      PSN_ENG_CRK = 360° - NC_OFS_TDC0_REF_CRK (engine revolution 1)
    Then
            If      Alpha < REL_ANG_CRK_CAM < Beta
            Then    return (true)
            Else    return (false)
            Endif
    Else
            Alpha = 720° - NC_OFS_TDC0_REF_CRK - NC_PSN_EDGE_z_CAM_IN(EX)_i
                        - CAM_DYW_CRK_SYN_RTD_IN(EX)
            Beta = 720° - NC_OFS_TDC0_REF_CRK  - NC_PSN_EDGE_z_CAM_IN(EX)_i
                        + CAM_DYW_CRK_SYN_ADC_IN(EX)
            If          Beta < 0°
            Then        Alpha = Alpha + 720°
                        Beta = Beta + 720°
            Endif
            If          Alpha < 0°
            Then        (no operation)
            Endif

            if      PSN_ENG_CRK = 720° - NC_OFS_TDC0_REF_CRK (engine revolution 0)
```

```
        Then
                If      Alpha < REL_ANG_CRK_CAM < Beta
                Then    return (true)
                Else    return (false)
                Endif
        Endif
    Endif
```

Test 3

```
    Alpha = NC_PSN_EDGE_z_CAM_IN(EX)_i - (360° - NC_OFS_TDC0_REF_CRK )
                    - CAM_DYW_CRK_SYN_ADC_IN(EX)
    Beta = NC_PSN_EDGE_z_CAM_IN(EX)_i - (360° - NC_OFS_TDC0_REF_CRK )
                    + CAM_DYW_CRK_SYN_RTD_IN(EX)
    If          Beta < 0°
    Then        Alpha = Alpha + 720°
                Beta = Beta + 720°
    Endif
    If          Alpha < 0°
    Then        (no operation)
    Endif

    If          360° - NC_OFS_TDC0_REF_CRK < PSN_ENG_CRK <
                    720° - NC_OFS_TDC0_REF_CRK
    Then        (engine revolution 1)
                If          Alpha < REL_ANG_CAM_REF_GAP < Beta
                Then        return (true)
                Else        return (false)
                Endif
    Else
        (engine revolution 0)
        Alpha = NC_PSN_EDGE_z_CAM_IN(EX)_i - (720° - NC_OFS_TDC0_REF_CRK)
                    - CAM_DYW_CRK_SYN_ADC_IN(EX)
        Beta = NC_PSN_EDGE_z_CAM_IN(EX)_i - (720° - NC_OFS_TDC0_REF_CRK)
                    + CAM_DYW_CRK_SYN_RTD_IN(EX)
        If          Beta < 0°
        Then        Alpha = Alpha + 720°
                    Beta = Beta + 720°
```

Endif

If Alpha < 0°

Then (no operation)

Endif

if Alpha < REL_ANG_CAM_REF_GAP < Beta

Then return (true)

Else return (false)

Endif

Endif

Test 4

No camshaft edge occured during one engine cycle

return (false)

2.7.4.7 Pre-injection interface computation before Cam edge function ()

General information:

A truth table is used in order to define the possible engine position.

Application conditions:

Recurrence : On each crankshaft active edge (CRK), reference gap (GAP) or camshaft edge (CAM)

Formula section:

Event (n)	CRK	GAP
Pre-inj Action	1	2

Pre-inj Action 1

If CONF_CAM_VVT_EX = 0

If REL_ANG_CRK_CAM > C_CRK_ANG_DLY

AND LV_ANG_EVT = 0

Then **PSN_ENG_SYN_CAM_MIN** n= **PSN_ENG_SYN_CAM_MIN** n-1 + C_PSN_MIN_OFS

PSN_ENG_SYN_CAM_MAX n= **PSN_ENG_SYN_CAM_MAX** n-1 - C_PSN_MAX_OFS

LV_ANG_EVT = 1

Else

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 + 360/NC_NR_TOOTH

PSN_ENG_SYN_CAM_MAX n= **PSN_ENG_SYN_CAM_MAX** n-1

Endif

Elseif CONF_CAM_VVT_EX = 1

If REL_ANG_CRK_CAM > C_CRK_ANG_DLY_1

AND LV_ANG_EVT = 0

ACTION_ENSD_GetCAMINLevel(IN<STATE_CAM_IN>)

ACTION_ENSD_GetCAMEXLevel(IN<STATE_CAM_EX>)

If [(STATE_CAM_IN = 0 **and** STATE_CAM_EX = 1) **or** (STATE_CAM_IN = 1 **and** STATE_CAM_EX = 0)]

Then **PSN_ENG_SYN_CAM_MIN** n= **PSN_ENG_SYN_CAM_MIN** n-1 + C_PSN_MIN_OFS_1

PSN_ENG_SYN_CAM_MAX n= **PSN_ENG_SYN_CAM_MAX** n-1 - C_PSN_MAX_OFS

Elseif [(STATE_CAM_IN = 1 **and** STATE_CAM_EX = 1) **or** (STATE_CAM_IN = 0 **and** STATE_CAM_EX = 0)]

Then **PSN_ENG_SYN_CAM_MIN** n= **PSN_ENG_SYN_CAM_MIN** n-1 + C_PSN_MIN_OFS_2

PSN_ENG_SYN_CAM_MAX n= **PSN_ENG_SYN_CAM_MAX** n-1 - C_PSN_MAX_OFS

Endif

Else

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 + 360/NC_NR_TOOTH

PSN_ENG_SYN_CAM_MAX n= **PSN_ENG_SYN_CAM_MAX** n-1

Endif

Endif

<u>Pre-inj Action 2</u>

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 +

(NC_NR_TOOTH_GAP + 1) * 360/NC_NR_TOOTH

PSN_ENG_SYN_CAM_MAX n = **PSN_ENG_SYN_CAM_MAX** n-1

2.7.4.8 *Pre-injection interface computation after Cam edge function ()*

<u>General information:</u>

A truth table is used in order to define the possible engine position.

<u>Application conditions:</u>

Recurrence : On each crankshaft active edge (CRK), reference gap (GAP) or camshaft edge (CAM)

<u>Formula section:</u>

Event (n)	CRK	CRK	CAM	GAP	GAP
Event (n-1)	CRK	CAM	CRK	CRK	CAM
Pre-inj Action	1	1	2	3	4

Pre-inj Action 1

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 + 360°/NC_NR_TOOTH

PSN_ENG_SYN_CAM_MAX n = **PSN_ENG_SYN_CAM_MAX** n-1 + 360°/NC_NR_TOOTH

Pre-inj Action 2

PSN_ENG_SYN_CAM_MIN = NC_PSN_EDGE_z_CAM_IN(EX)_i – CAM_DYW_SYN_IN(EX)

for the lowest possible signal edge index z

PSN_ENG_SYN_CAM_MAX = NC_PSN_EDGE_z_CAM_IN(EX)_i + CAM_DYW_SYN_IN(EX)

for the highest possible signal edge index z

Note: The possible signal edges z are determined by checking the conditions of "Camshaft Edge Index determination function()". This function must be executed before action 2.

Pre-inj Action 3

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 + (NC_NR_TOOTH_GAP+1) * 360°/NC_NR_TOOTH

PSN_ENG_SYN_CAM_MAX n = **PSN_ENG_SYN_CAM_MAX** n-1 + (NC_NR_TOOTH_GAP+1) * 360°/NC_NR_TOOTH

Pre-inj Action 4

A camshaft edge has occured during the reference gap (when the camshaft edge is detected Pre-inj Action 2 is performed and when the first crankshaft active edge after the reference is detected Pre-inj Action 4 is performed)

PSN_ENG_SYN_CAM_MIN n = **PSN_ENG_SYN_CAM_MIN** n-1 + **REL_ANG_CRK_CAM**

PSN_ENG_SYN_CAM_MAX n = **PSN_ENG_SYN_CAM_MAX** n-1 + **REL_ANG_CRK_CAM**

2.7.4.9 VVT lock check function ()

General information:

The purpose of this function is to check the VVT lock position flag and to extend the cam/crk tolerance windows if the VVT is not in locked position at engine start.

Application conditions:

Recurrence: at transition of LV_STOP_ENG from 0 to 1

Or

Reset ECU

Formula section:

If LV_CAM_LOCK_IVVT_IN(EX)_i = 0 for the camshaft sensor selected for cam/crk synchronization

And

NLC_IVVT_IN(EX) = 1

Then VVT not in locked position at engine start

CAM_DYW_SYN_IN = C_DYW_CAM_SYN_IN + C_CAM_ADJ_VVT_SYN_IN

CAM_DYW_SYN_EX = C_DYW_CAM_SYN_EX + C_CAM_ADJ_VVT_SYN_EX

CAM_DYW_CRK_SYN_ADC_IN = C_DYW_CAM_CRK_SYN_ADC_IN + C_CAM_ADJ_VVT_SYN_CRK_ADC_IN

CAM_DYW_CRK_SYN_ADC_EX = C_DYW_CAM_CRK_SYN_ADC_EX

CAM_DYW_CRK_SYN_RTD_IN = C_DYW_CAM_CRK_SYN_RTD_IN

CAM_DYW_CRK_SYN_RTD_EX = C_DYW_CAM_CRK_SYN_RTD_EX + C_CAM_ADJ_VVT_SYN_CRK_RTD_EX

Else

VVT in locked position at engine start or no VVT at all

CAM_DYW_SYN_IN = C_DYW_CAM_SYN_IN

CAM_DYW_SYN_EX = C_DYW_CAM_SYN_EX

CAM_DYW_CRK_SYN_ADC_IN = C_DYW_CAM_CRK_SYN_ADC_IN

CAM_DYW_CRK_SYN_ADC_EX = C_DYW_CAM_CRK_SYN_ADC_EX

CAM_DYW_CRK_SYN_RTD_IN = C_DYW_CAM_CRK_SYN_RTD_IN

CAM_DYW_CRK_SYN_RTD_EX = C_DYW_CAM_CRK_SYN_RTD_EX

Endif

2.7.5 Requirements to Infrastructure

2.7.5.1 Acquisition and pre-filtering

General information:

The Cam acquisition will be available for every active edge of the signal. A pre-filtering is used in order to remove the glitch.

Application conditions:

Recurrence: every active edge

Formula section:

System request accuracy <= 4 us

Activation of Signal de-glitching function () at each active edge.

Calculation of camshaft tooth period: The camshaft tooth period T_SEG_CAM_IN(EX)_i is the time between two consecutive camshaft signal edges.

2.7.5.2 Relative cam crank angle position

General information:

The purpose of this function is to provide a measurement for

- the angular distance between the first valid crankshaft teeth and the first active camshaft edge
- the angular distance between two consecutive valid active camshaft signal edges

Only active camshaft signal edges shall be processed. The active camshaft edge for synchronization is defined by NC_ACT_CAM_EDGE_SYN.

Application conditions:

Recurrence: every crankshaft or camshaft active edge

Formula section:

```
If      LV_CRK_FIRST_VLD TOOTH = 0
Then    REL_ANG_CRK_CAM = 0
Else    If      active cam edge
        Then    REL_ANG_CRK_CAM = 0
        Else    If      active crank edge or after missing tooth simulation
                Then    REL_ANG_CRK_CAM = REL_ANG_CRK_CAM +
                                360°crk /NC_NR_TOOTH
                Else    REL_ANG_CRK_CAM n is frozen
                Endif
        Endif
Endif
```

2.7.5.3 Relative cam angle position from reference gap

General information:

The purpose of this function is to provide a measurement for

- the angular distance between a crankshaft reference gap and a active camshaft signal edge

This counter will be reset at every reference gap.

Application conditions:

Recurrence: every crankshaft active edge

Formula section:

```
If      LV_CRK_SYN = 0
Then    REL_ANG_CAM_REF_GAP = 0
Else    if      Reference Gap is detected
        Then    REL_ANG_CAM_REF_GAP = 0
        Else    if      crank active edge
                Then    REL_ANG_CAM_REF_GAP = REL_ANG_CAM_REF_GAP +
                                360°crk /NC_NR_TOOTH
                Else    REL_ANG_CAM_REF_GAP is frozen
                Endif
        Endif
Endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_NR_EDGE_MIN_VLD_CAM_IN	1	0...FFH	0...255	1	-
Number of intake camshaft signal edges for valid position output					
C_NR_EDGE_MIN_VLD_CAM_EX	1	0...FFH	0...255	1	-
Number of exhaust camshaft signal edges for valid position output					
C_DYW_CAM_SYN_IN	1	0...1FFH	0...31.9375	0.0625	°CRK
Tolerance window for angle between intake camshaft signal edges in crankshaft synchronization mode					
C_DYW_CAM_SYN_EX	1	0...1FFH	0...31.9375	0.0625	°CRK
Tolerance window for angle between exhaust camshaft signal edges in crankshaft synchronization mode					
C_DYW_CAM_CRK_SYN_ADC_IN	1	0...7FFH	0...127.9375	0.0625	°CRK
Tolerance window advance for intake camshaft to crankshaft reference in crankshaft synchronization mode					
C_DYW_CAM_CRK_SYN_ADC_EX	1	0...7FFH	0...127.9375	0.0625	°CRK
Tolerance window advance for exhaust camshaft to crankshaft reference in crankshaft synchronization mode					
C_DYW_CAM_CRK_SYN_RTD_IN	1	0...7FFH	0...127.9375	0.0625	°CRK
Tolerance window retard for intake camshaft to crankshaft reference in crankshaft synchronization mode					
C_DYW_CAM_CRK_SYN_RTD_EX	1	0...7FFH	0...127.9375	0.0625	°CRK
Tolerance window retard for exhaust camshaft to crankshaft reference in crankshaft synchronization mode					
ID_FAC_CAM_IN	16	10...FFH	1...15.9375	0.0625	-
LDPM_N_32_1_ENSD	16	0...FFH	0...8160	32	rpm
Period ratio tolerance factor for intake camshaft					
ID_FAC_CAM_EX	16	10...FFH	1...15.9375	0.0625	-
LDPM_N_32_1_ENSD	16	0...FFH	0...8160	32	Rpm
Period ratio tolerance factor for exhaust camshaft					
C_CRK_ANG_DLY	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Crankshaft angle threshold after start of cranking to reduce the possible engine position range for pre-injection					
C_PSN_MIN_OFS	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Offset angle on minimum engine position for pre-injection when Crankshaft angle threshold is reached					
C_PSN_MAX_OFS	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Offset angle on maximum engine position for pre-injection when Crankshaft angle threshold is reached					
C_CAM_ADJ_VVT_SYN_IN	1	0...7FFH	0...127.9375	0.0625	°CRK
Extension of tolerance window CAM_DYW_SYN_IN when VVT is not in locked position at start					
C_CAM_ADJ_VVT_SYN_EX	1	0...7FFH	0...127.9375	0.0625	°CRK
Extension of tolerance window CAM_DYW_SYN_EX when VVT is not in locked position at start					
C_CAM_ADJ_VVT_SYN_CRK_ADC_IN	1	0...7FFH	0...127.9375	0.0625	°CRK
Extension of tolerance window CAM_DYW_CRK_SYN_ADC_IN when VVT is not in locked position at start					
C_CAM_ADJ_VVT_SYN_CRK_RTD_EX	1	0...7FFH	0...127.9375	0.0625	°CRK
Extension of tolerance window CAM_DYW_CRK_SYN_RTD_EX when VVT is not in locked position at start					
C_CRK_ANG_DLY_1	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Crankshaft angle threshold after start of cranking to reduce the possible engine position range for pre-injection when both camshaft sensors are available					
C_PSN_MIN_OFS_1	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Offset angle on minimum engine position for pre-injection when Crankshaft angle threshold is reached when the intake and exhaust camshaft signals are high and low respectively or vice-versa					
C_PSN_MIN_OFS_2	1	0000...2CFFH	0...719.9375	0.0625	°CRK
Offset angle on minimum engine position for pre-injection when Crankshaft angle threshold is reached when the intake and exhaust camshaft signals are either both high or both low					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_GAP	1	1...2H	1..2	1	-
Number of reference gaps per engine revolution					
NC_ACT_CAM_EDGE_SYN	1	1...3H	1..3	1	-
Active edge of camshaft signal for cam/crk synchronization					
NC_ACT_CAM_EDGE_LIH	1	1...3H	1..3	1	-
Active edge of camshaft signal for crankshaft limp-home					
NC_T_SEG_MIN_CAM_IN	1	0...FFFFFFH	0...3.9929	2.38e-7	s

Minimum time between camshaft signal edges					
NC_T_SEG_MAX_CAM_IN	1	0...FFFFFFH	0...3.9929	2.38e-7	s
Maximum time between camshaft signal edges					
NC_T_SEG_MIN_CAM_EX	1	0...FFFFFFH	0...3.9929	2.38e-7	s
Minimum time between camshaft signal edges					
NC_T_SEG_MAX_CAM_EX	1	0...FFFFFFH	0...3.9929	2.38e-7	s
Maximum time between camshaft signal edges					
NC_NR_EDGE_CAM_IN	1	1...FH	1...15	1	-
Number of signal edges per intake camshaft revolution					
NC_NR_EDGE_CAM_EX	1	1...FH	1...15	1	-
Number of signal edges per exhaust camshaft revolution					
NC_PSN_EDGE_z_CAM_IN_i	1...2	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Ideal engine position relative to TDC0 for intake camshaft i signal edge z; Dimension = NC_NR_EDGE_CAM_IN					
NC_PSN_EDGE_z_CAM_EX_i	1...2	0000 ... 2CFFH	0...719.9375	0.0625	°CRK
Ideal engine position relative to TDC0 for exhaust camshaft i signal edge z; Dim. = NC_NR_EDGE_CAM_EX					

2.8 IVVT Output - Requirements to Infrastructure

Export actions:

ACTION_INFR_SetHoldIvvt(IN <Ivvt>, IN <Ivvthpwm>, IN <Frq_Ivvthpwm>)
This action sets a holding pulse width modulation energisation for the camshaft Ivvt.
ACTION_INFR_StartPulseIvvt(IN <Ivvt>, IN <Ivvtpwm>, IN <Frq_Ivvtpwm>, IN <T_Ivvtpwm>)
This action starts a time limited pulse width modulation energisation for the camshaft Ivvt.
ACTION_INFR_StopPulseIvvt(IN <Ivvt>)
This action stops the time limited pulse width modulation energisation for the camshaft Ivvt.

Description for actions:

ACTION_INFR_SetHoldIvvt(<Ivvt>, <Ivvthpwm>, <Frq_Ivvthpwm>)
This action sets a holding pulse width modulation energisation for the camshaft Ivvt.
Default SYNCHRONIZATION option when calling this Action is ASYNCHRONOUS.

Parameter	Type	Hex. Limits	Phys. Limits	Resol.	Unit
Ivvt	IN	0H 1H 2H 3H	IVVT_IN_1 IVVT_EX_1 IVVT_IN_2 IVVT_EX_2	1	-
Camshaft specification for setting energisation					
Ivvthpwm	IN	0...FFFFH	0...99.99847	0.001526	%
Holding pulse width modulation					
Frq_Ivvthpwm	IN	32H...1F4H	50...500	1	Hz
Frequency of holding pulse width modulation					

ACTION_INFR_StartPulseIvvt(<Ivvt>, <Ivvtpwm>, <Frq_Ivvtpwm>, <T_Ivvtpwm>)
This action starts a time limited pulse width modulation energisation for the camshaft Ivvt.
Default SYNCHRONIZATION option when calling this Action is ASYNCHRONOUS.

Parameter	Type	Hex. Limits	Phys. Limits	Resol.	Unit
Ivvt	IN	0H 1H 2H 3H	IVVT_IN_1 IVVT_EX_1 IVVT_IN_2 IVVT_EX_2	1	-
Camshaft specification for setting energisation					
Ivvtpwm	IN	0...FFFFH	0...99.99847	0.001526	%
Pulse width modulation for adjustment energisation					
Frq_Ivvtpwm	IN	32H...1F4H	50...500	1	Hz
Frequency of pulse width modulation for adjustment energisation					
T_Ivvtpwm	IN	0...FFFFFFFFH	0...17179.87	4e-6	s
Duration of adjustment energisation					

ACTION_INFR_StopPulseIvvt(<Ivvt>)
This action stops the time limited pulse width modulation energisation for the camshaft Ivvt.
Default SYNCHRONIZATION option when calling this Action is ASYNCHRONOUS.

Parameter	Type	Hex. Limits	Phys. Limits	Resol.	Unit
Ivvt	IN	0H 1H 2H 3H	IVVT_IN_1 IVVT_EX_1 IVVT_IN_2 IVVT_EX_2	1	-
Camshaft specification for setting energisation					

FUNCTION DESCRIPTION:

General information:

The IVVT control consists of the merging of the holding pulse width modulation energisation and time limited low or high level adjustment pulses. The holding energisation is a system parameter, which is adapted slightly due to e.g. tolerances. It can be the low level energisation in some special cases. The adjustment energisations are calculated by the controller and lead to camshaft movement. Some solenoid valve energisation cases are shown in figure "Solenoid valve energisation".

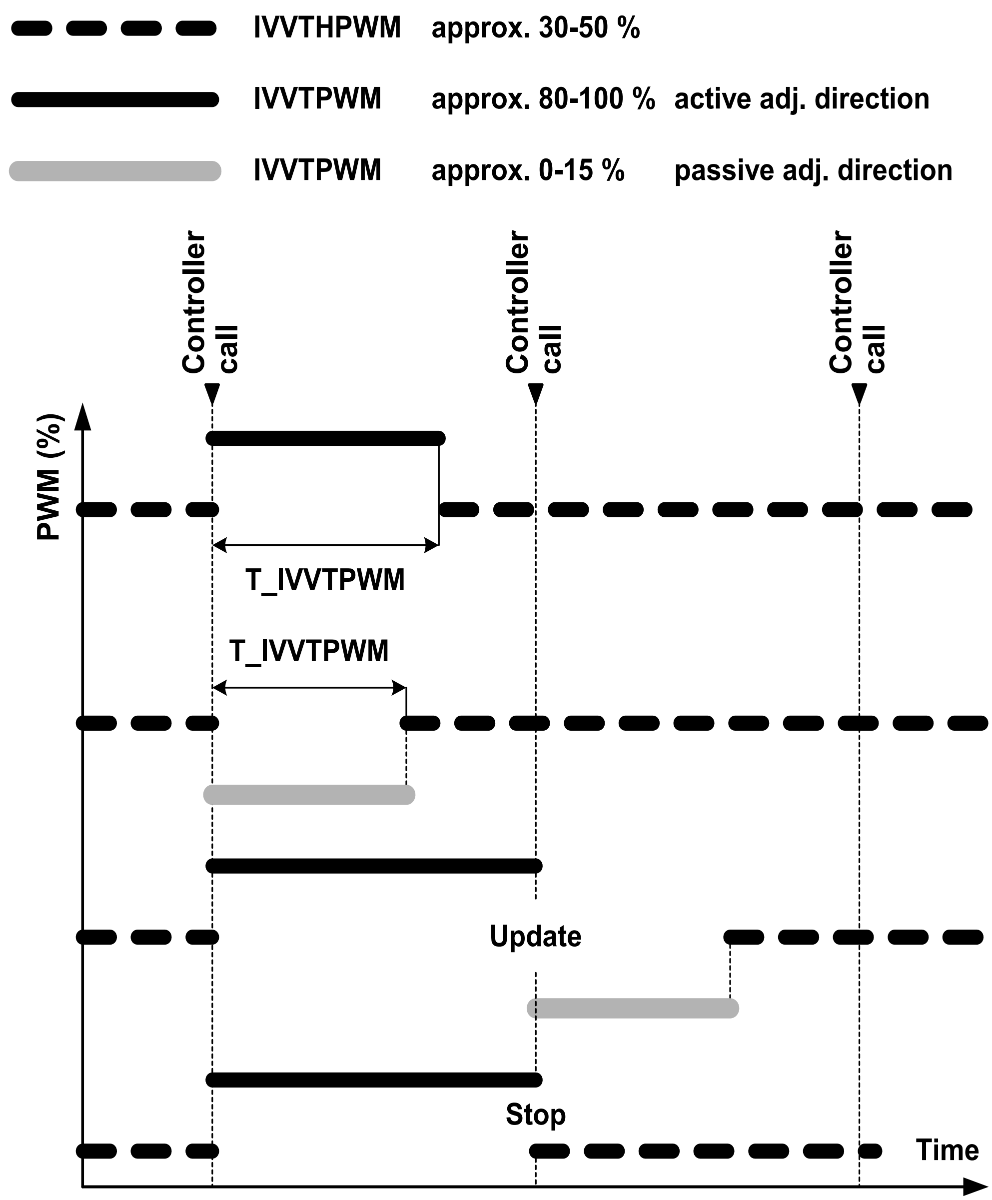


Figure: "Solenoid valve energisation".

Requirements for ACTION_INFR_SetHoldIvvt:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Ivvt	1	-	-	-	Set energisation for related camshaft: IVVT_IN_1, IVVT_IN_2, IVVT_EX_1, IVVT_EX_2
Ivvthpwm	0.025	-	-	-	If FFFFH is set then real output has to be 100 %
Frq_Ivvthpwm	1	-	-	-	

Requirements for ACTION_INFR_StartPulseIvvt:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Ivvt	1	-	-	-	Set energisation for related camshaft: IVVT_IN_1, IVVT_IN_2, IVVT_EX_1, IVVT_EX_2
Ivvtpwm	0.025	-	-	-	If FFFFH is set then real output has to be 100 %
Frq_Ivvtpwm	1	-	-	-	
T_Ivvtpwm	1e-4	-	-	-	

Requirements for ACTION_INFR_StopPulseIvvt:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Ivvt	1	-	-	-	Set energisation for related camshaft: IVVT_IN_1, IVVT_IN_2, IVVT_EX_1, IVVT_EX_2

Diagnosis: -

Coincidence requirements:

ACTION_INFR_SetHoldIvvt starts the holding energisation.

ACTION_INFR_StartPulseIvvt starts a time limited adjustment energisation. This energisation replaces the holding energisation.

If the adjustment energisation is over or is stopped by ACTION_INFR_StopPulseIvvt the holding energisation starts again.

An active adjustment energisation can be updated by a new one without a switch to the holding energisation. The update means a start of a new adjustment energisation.

There is no effect if ACTION_INFR_StopPulseIvvt is called and there is currently no active adjustment energisation.

2.9 Acquisition of vehicle speed

Output data:

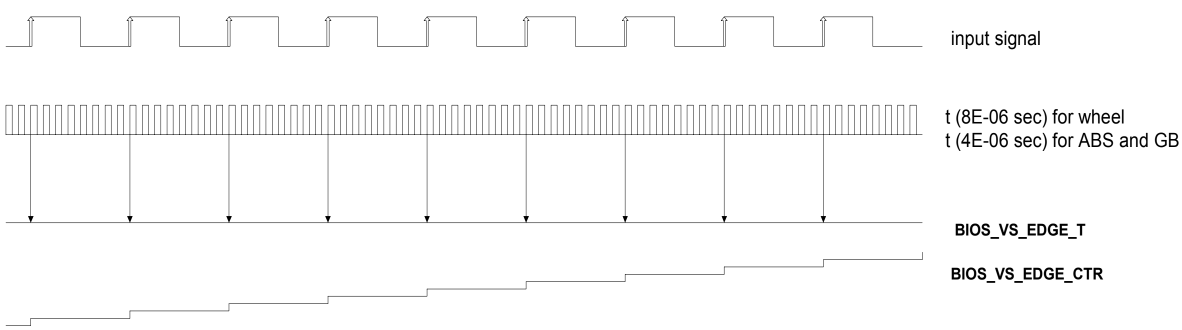
Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
BIOS_VS_EDGE_CTR	O	0 ... FFFFH	0 ... 65565	1	-
Vehicle speed sensor tooth counter (free running counter)					
BIOS_VS_EDGE_T	O	0 ... FFFFH	0 … 1,718 E10	4	µs
timestamp of last tooth (free running timer)					

FUNCTION DESCRIPTION:

General information:

The signal from wheel sensor or ABS control unit or gearbox sensor (high and low) is recorded by incrementation of the free running edge counter BIOS_VS_EDGE_CTR with one digit at every change from low to high. The value is stored as BIOS_VS_EDGE_CTR in a free running counter too.

Signal flow diagram:



2.9.1 Segment period acquisition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_T_MES_0_RR_BAS	V/O	0...FFFFH	RES_ECU_ SEG_T_MES_RR	LIM_PHY_ECU_ SEG_T_MES_RR	s
Actual segment time measurement from the wheel speed signal – BSW acquisition					
SEG_T_MES_1_RR_BAS	V/O	0...FFFFH	RES_ECU_ SEG_T_MES_RR	LIM_PHY_ECU_ SEG_T_MES_RR	s
Previous segment time measurement from the wheel speed signal – BSW acquisition					

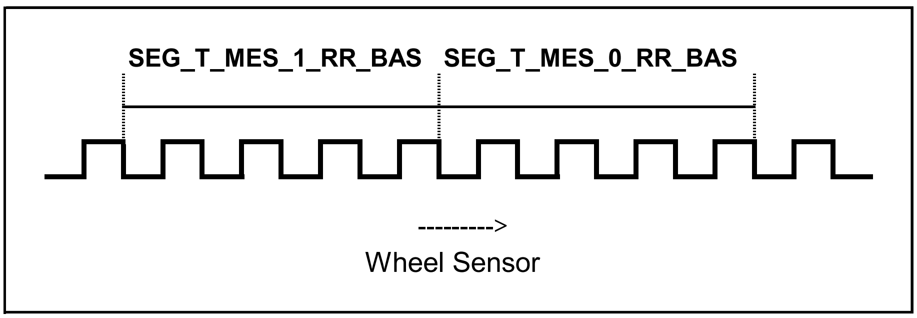
Input data:

NC_T_BAS_FRQ_DIV	NC_FRQ_ECU		

General information:

SEG_T_MES_0_RR_BAS and SEG_T_MES_0_RR_BAS are the Basic s/w acquisitions of wheel speed signal. More informations are available in chapter B, "Rough Road detection".

Description:



Remark : SEG_T_MES_1_RR is the segment period preceding SEG_T_MES_0_RR.

2.9.1.1 Definition of output variable depending on ECU and ASW frequency

As the function is used with different ECU and ASW quartz frequencies, the output resolution and the physical limits are defined as variables.

The resolution of the output data depending on the quartz frequency is defined as:

RES_ECU_SEG_T_MES_RR = NC_T_BAS_FRQ_DIV / NC_FRQ_ECU

The HEX-limits don’t change.

The physical limits are :

LIM_PHY_ECU_SEG_T_MES_RR = LIM_HEX (dec) * RES_ECU_SEG_T_MES_RR

2.10 Acquisition of electronically controlled cooling fan signal

General information:

The configuration constant NC_ECF_CONF allows to determine the wanted fan control strategy. In principal it is possible to choose between a RLY- or a PWM-control architecture. A RLY/PWM fan control strategy in parallel is also existing. In this case the aggregate architecture allows to switch either the control of RLY-fan(s) or the control of PWM–fan(s) during ECU runtime depending on the setting of the configuration bit CONF_CFA. The control of both variants (RLY- and PWM-fan(s)) in parallel at the same time is not allowed.

The following fan control architecture is adjusted:

NC_ECF_CONF = 2 ➔ control of cooling fan(s) by Relay(s) **or** PWM(s)
(CONF_CFA = 0 for RLY- control)
(CONF_CFA = 1 for PWM- control)

The configuration bit CONF_CFA is representing the activated control algorithm. Either the functions for RLY-fan(s) or the functions for PWM-fan(s) are activated. Both control functions are never activated in at the same time.

For each fan stage as well as each PWM value, a signal at the ECU output stage has to be generated by the infrastructure to guarantee the link between the calculated signal to the available cooling fan(s) (hardware component) at the vehicle.

2.10.1 Acquisition of cooling fan signal (RLY version)

Input data:

LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR]	NC_ECF_RLY_NR	NC_ECF_NR	

FUNCTION DESCRIPTION:

General information:

In case of a enabled RLY-fan control strategy (refer chapter: “ENTE Configuration data”), a Boolean is calculated (LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR) to switch on / off the requested cooling fan stage(s). For each fan stage a signal at the ECU output has to be generated by the infrastructure to guarantee the link between the calculated signal to the available cooling fan(s) (hardware component) at the vehicle.

Name	Frequency	Notes
LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR] (LV_RYL_FAN_L) (LV_RYL_FAN_H)	40 ms	Logical signal to switch on / off the requested cooling fan stage(s)

Application conditions:

Activation: at every engine operating state

Deactivation: -

(see: “ENTE scheduler”)

Formula section:

In case of a RLY/PWM-fan configuration (NC_ECF_CONF=2), when the RLY output stage(s) for the control of the cooling fan(s) at the vehicle are activated, the PWM output stage(s) are permanent disabled (no PWM output) to prevent an unwanted fan control behaviour.

2.10.2 Acquisition of cooling fan signal (PWM version)

Input data:

ECFPWM[NC_ECF_NR]	NC_ECF_NR		

FUNCTION DESCRIPTION:

General information:

In case of an enabled PWM-fan control strategy (refer chapter: “ENTE Configuration data), a pulse width modulated value is calculated (ECFPWM[NC_ECF_NR]) to control the requested cooling fan(s). For each PWM value, a signal at the ECU output stage has to be generated by the infrastructure to guarantee the link between the calculated signal to the available cooling fan(s) (hardware component) at the vehicle.

PWM output name	Frequency	Updating of duty cycle	Range of duty cycle
ECFPWM[NC_ECF_NR] (CFAPWM_CFA)	300 Hz	200 ms	0 -100%

Application conditions:

Activation: at every engine operating state

Deactivation: -

(see: “ENTE scheduler”)

Formula section:

In case of a RLY/PWM-fan configuration (NC_ECF_CONF=2), when the PWM output stage(s) for the control of the cooling fan(s) at the vehicle are activated, the RLY output stage(s) are permanent disabled (no RLY output) to prevent an unwanted fan control behaviour.

2.11 Acquisition of throttle position

2.11.1 Definition of TPSPWM

Output data:

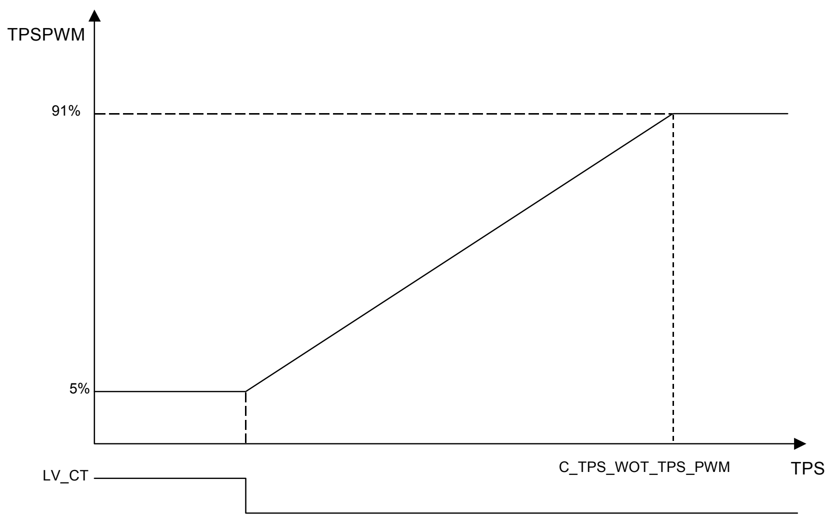
Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TPSPWM	V/O	0...FFFFH	0...100	1,53E-3	%
Relative throttle angle distributed to TCU					

Input data:

TPS	C_TPS_IS_OUT	LV_ERR_TPS	LV_ERR_TPS_PLAUS
LV_CT			

If LV_ERR_TPS = 1
or LV_ERR_TPS_PLAUS = 1
then TPSPWM = 3% *(Error Identifier)*
else **if** LV_CT = 1
then TPSPWM = 5%
else **if** TPS > C_TPS_WOT_TPS_PWM
then TPSPWM = 91%
else TPSPWM =

$$\frac{91\% - 5\%}{C\_TPS\_WOT\_TPSPWM - C\_TPS\_IS\_OUT} * (TPS - C\_TPS\_IS\_OUT) + 5\%$$



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TPS_WOT_TPSPWM	1	0...FFH	0...119,5	0,47	°TPS
Throttle angle which corresponds with 91% TPSPWM					

2.12 Acquisition of fuel tank level voltage

2.12.1 Basic software

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_FTL	V/O	0...FFH	0...4,985	0.0195	Volt
Main Fuel tank level voltage measured					
V_FTL_SUB	V/O	0...FFH	0...4,985	0.0195	Volt
Sub Fuel tank level voltage measured					

Input data:

FTL_BAS	FTL_BAS_SUB	CONF_DIAGCP_VOL	

FUNCTION DESCRIPTION:

General information:

The raw value of fuel tank level voltage (FTL_BAS) is measured by continuous conversion (10 bits) every **0.1 sec**.

The sub value of fuel tank level voltage (FTL_BAS_SUB) is measured by continuous conversion (10 bits) every **0.1 sec**, only if dual FTL sensor is available.

The corresponding value of the first measurement after hardware reset is used for initialisation.

Formula section:

```
If        CONF_DIAGCP_VOL = 0                     (no FTL sensor)
then            V_FTL       = 0
          and   V_FTL_SUB   = 0

else if   CONF_DIAGCP_VOL = 1                     (single FTL sensor)
then            V_FTL       = FTL_BAS / 4         (8 bits)
          and   V_FTL_SUB   = 0

else if   CONF_DIAGCP_VOL = 2                     (dual FTL sensor)
                V_FTL       = FTL_BAS / 4         (8 bits) : main FTL
          and   V_FTL_SUB   = FTL_BAS_SUB / 4     (8 bits) : sub FTL
endif
```

2.13 Input system event : Key off recognition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_KEY_OFF	V/O	0...1H	0...1	1	No
Key off flag					

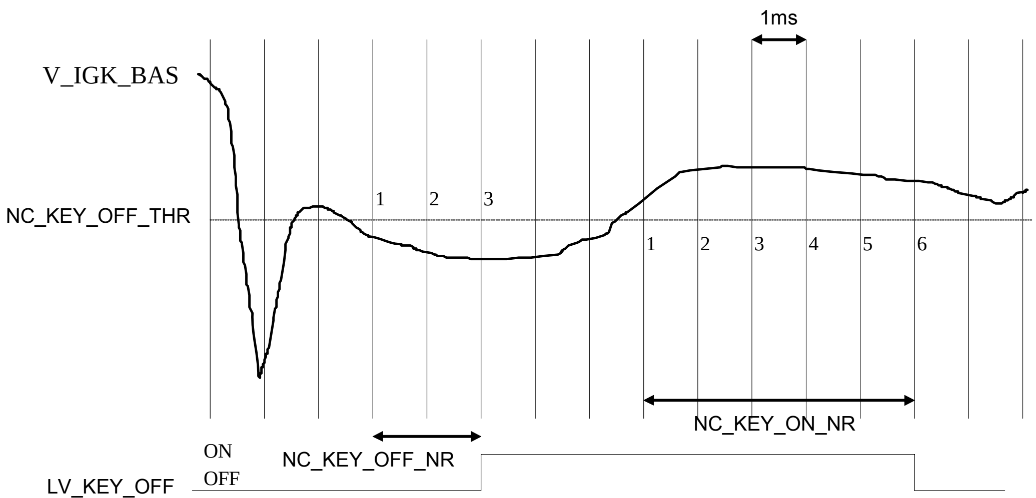
Input data:

V_IGK_BAS			

FUNCTION DESCRIPTION:

General information:

According to the battery voltage raw value (V_IGK_BAS) after ignition key, the ignition key OFF recognition is performed.



See on Analogic Acquisition Sampling chapter the recurrence of V_IGK_BAS acquisition.

Application conditions:

Activation: at reset

Initialization: LV_KEY_OFF = first valid value **after** RESET

Recurrency: 1ms

Formula section:

Ignition key ON recognised :

If V_IGK_BAS **>** NC_KEY_OFF_THR

for more than NC_KEY_ON_NR number of successive samples (* 1 msec.)

then LV_KEY_OFF **= 0** (ON)

End IF

Ignition key OFF recognised :

If V_IGK_BAS **<** NC_KEY_OFF_THR

for more than NC_KEY_OFF_NR number of successive samples (* 1 msec.)

then LV_KEY_OFF **= 1** (OFF)

End IF

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_KEY_OFF_THR	1	0...3FFH	0...NC_V_IGK_MAX	NC_V_IGK_MAX / 1024	V
Threshold for key off detection (typical value : 142 = 4 Volts)					
NC_KEY_OFF_NR	1	0...FFH	0...255	1	No
Samples filter for key off detection (typical value : 3)					
NC_KEY_ON_NR	1	0...FFH	0...255	1	No
Samples filter for key on detection (typical value : 6)					

2.14 Logic Outputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ACC_MAIN_LAMP	-	0...1H	0...1	1	-
Cruise control main lamp signal from ECU					
ACC_SET_LAMP	-	0...1H	0...1	1	-
Cruise control activ lamp signal from ECU					
LV_RLY_MAIN	V	0...01H	0...1	1	-
Main relay activation					

Input data:

LV_EXT_ADJ_ACC_MAIN_LAMP	LV_CRU_MAIN_SWI	LV_CRU_ACT	LV_CMD_RCL_OPEN
LV_ACT_EXT_ADJ_ACC_MAIN_LAMP	LV_EXT_ADJ_MIL	LV_MIL	LC_TCHA_CONF
LV_EXT_ADJ_ACC_SET_LAMP	LV_VIM_SP	ENG_STATE	LV_ACT_EXT_ADJ_MIL
LV_ACT_EXT_ADJ_ACC_SET_LAMP	NC_ETC_CONF		

FUNCTION DESCRIPTION:

List of logic Outputs:

Name	Port – Channel	Logic state
ACC_MAIN_LAMP	MPWM1	positive
ACC_SET_LAMP	MPWM3	positive
LV_RLY_MAIN	MPIO32B5	negative
LV_VIM_SP	A_TPUCH6	negative
LV_CMD_RCL_OPEN	A_TPUCH12	positive
LV_RLY_ST	SGPIOC6	negative

2.14.1 Cruise main lamp activation (NC_ETC_CONF = 1)

Initialisation: At EEProm Error or LV_IGK=0 to 1 then ACC_MAIN_LAMP = 0

if LV_EXT_ADJ_ACC_MAIN_LAMP = 1

then ACC_MAIN_LAMP = LV_ACT_EXT_ADJ_ACC_MAIN_LAMP

else if LV_CRU_MAIN_SWI = 1

then ACC_MAIN_LAMP = 1

else ACC_MAIN_LAMP = 0

2.14.2 Cruise set lamp activation (NC_ETC_CONF = 1)

if LV_EXT_ADJ_ACC_SET_LAMP = 1

then ACC_SET_LAMP = LV_ACT_EXT_ADJ_ACC_SET_LAMP

else if LV_CRU_ACT = 1

then ACC_SET_LAMP = 1

else ACC_SET_LAMP = 0

2.14.3 MIL activation

if ENG_STATE = ES

and LV_EXT_ADJ_MIL = 1

then MIL output is switched according to LV_ACT_EXT_ADJ_MIL

else MIL output is switched according to LV_MIL

2.14.4 Main relay activation

The main relay is controlled by ECU and the LV_RLY_MAIN shows the status of main relay from ECU output channel.

2.14.5 VIM activation

if LV_VIM_SP = 1

then activate VIM output with 99.4 % duty @ 20 Hz (=diagnosis pulse time 0.30ms)

else deactivate VIM output with 1% duty @ 20 Hz (=diagnosis pulse time 0.50ms)

2.14.6 RCL (Bypass valve) activation (LC_TCHA_CONF = 1)

If LV_CMD_RCL_OPEN = 1

then activate RCL output (open RCL valve)

else deactivate RCL output (close RCL valve)

2.14.7 Starter Relay activation

Initialisation: according to Spec: 5W901501.00X " In Reset ECU output is pulled to ground" (i.e. LV_RLY_ST = 1)

If LV_RLY_ST = 1

then output switched to GND (as negative logic)

else output switched to VB (as negative logic)

2.15 PWM Outputs and Inputs

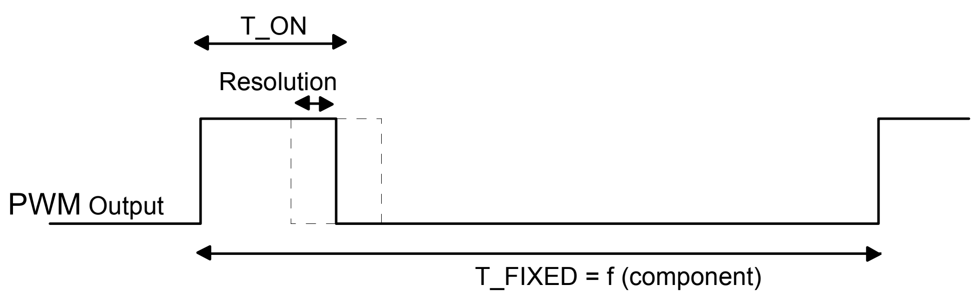
Input data:

CPPWM_CPS	LSHPWM_UP[NC_CBK_EX_NR]	LSHPWM_DOWN[NC_CBK_EX_NR]	IVVTPWM_IN_i
IVVTPWM_EX_i	TPSPWM	CFAPWM_CFA	ISAPWM_ISA
PWM_WG	LV_SOV	PORTPWM	FRQ_PORTPWM
GEN_LOAD	MTCPWM	ALTPWM	CONF_BAT

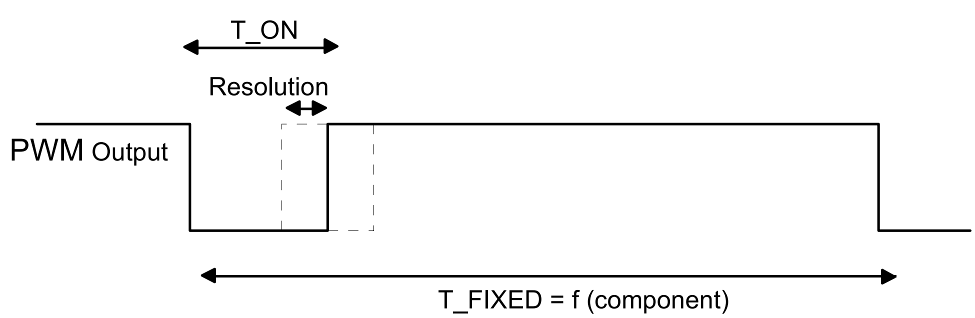
FUNCTION DESCRIPTION:

2.15.1 PWM management principle :

“ Positive “ PWM : Duty cycle = T_ON / T_FIXED



“ Negative “ PWM : Duty cycle = T_ON / T_FIXED



In case of PWM input signal, the PWM value is automatically calculated by basic software.

2.15.2 List of PWM Outputs:

PWM Output Name	Frequency (Hz)	Attached Component	Updating of Duty Cycle (ms)	Range of Duty Cycle (%)	PWM State	Remarks
CPPWM_CPS	16		20	0...99,9	negative	
LSHPWM_UP	10	Zirconium NTK	1000	2,3...97,3	negative	binary
	30			4...96		linear
LSHPWM_DOWN	7,6	Zirconium NTK	1000	2,3...97,3	negative	
IVVTPWM_IN	50...500 *)	Denso Actuator	10	0...99,9	negative	
IVVTPWM_EX	50...500 *)	Denso Actuator	10	0...99,9	negative	
TPSPWM	100	TCU	10	5...91	positive	
CFAPWM_CFA	300		200	0...99.61	positive	
MTCPWM	1000	ETC	2	-100...99.9	for -100...0%: ETC1: 0V ETC2: positive for 0...99.9%: ETC1: VB ETC2: negative ***)	Only ETC
ISAPWM_ISA	250		10	0...99.9	ISA1:negative ISA2:positive	Only ISA
PWM_WG	30**)	Wastegate	10	0...100	negative	Only TCI
LV_SOV	7	SOV		0.003/100	negative	ON/OFF control
PORTPWM	FRQ_PORTPWM*) 1000…8000	DC-Motor Port Actuator	10	-100..99.9	for -100...0%: PORT1: 0V PORT2: positive for 0...99.9%: PORT1: VB PORT2: negative	
ALTPWM	125	Alternator regulator	100	0...99.9	negative	Only for ETC

*) the frequency is managed in the function itself. Please see these function for more details.

**) after ECU reset PWM_WG frequency is switched to C_PWM_WG_FRQ (see function for more details)

***) for the PWM state see datailed information in "THRO-Requirements to infrastructure interface (ETC)" –module (xx2043yy.zzz)

2.15.3 List of PWM Inputs:

PWM Input Name	Frequency (Hz)	Attached Component	Updating of Duty Cycle (ms)	Range of Duty Cycle (%)	PWM State
GEN_LOAD	60	Valeo Alternator (CONF_BAT = 0)	10	0...100	Negative
	167	Denso Alternator (CONF_BAT_= 0)			Negative
	250	Valeo Alternator for Battery Management (CONF_BAT = 1)			Positive

2.16 Fuel Injection – Requirements to I/O SW

2.16.1 General Requirements (MPI, HPDI)

Output data:(general)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_INJ_StateOfPrevInj_x	O	-	0...1	1	-
Cylinder individuel flag which indicates whether last injection was activated or deactivated					
TRL_SOI_1_MES_x	O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the first injection pulse, estimated					
TRL_EOI_1_ MES_x	O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the first injection pulse, estimated					
TRL_SOI_2_ MES_x	O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the second injection pulse, estimated					
TRL_EOI_2_ MES_x	O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the second injection pulse, estimated					
TRL_TI_1_ MES_x	O	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, first pulse					
TRL_TI_2_ MES_x	O	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, second pulse					
TRL_TRIG_EOI_LIM(CYL_AV)	O	-	-	-	-
Trigger at every EOI_LIM_x					
CYL_AV	O	0 ... 7 H	0 ... 7	1	-
Function parameter of TRL_TRIG_EOI_LIM, which indicate the actual logical cylinder number for that cylinder which EOI_LIM was reached.					
TRL_NC_REF_EOI_LIM	O	**0 ... FC40 H**	**0 ... -360**	**0.375**	°CRK
Offset of the EOI_LIM reference point after TDC					

Input data: (general)

NC_CYL_NR	TRL_INJ_MOD_GLOBAL	TRL_NC_EOI_LIM	TRL_TI_PLS_1_x
TRL_EOI_1_x	TRL_INJ_MOD_x	TRL_SOI_MAX	TRL_TI_PLS_2_ x
TRL_EOI_2_x	TRL_LV_ADD_PULSE_x	TRL_TI_ADD_DLY	TRL_TRIG_INJ_UPD
TRL_EOI_MIN_x	TRL_LV_INH_INJ_x	TRL_TI_ADD_PULSE_MIN	TRL_TRIG_INJ_UPD_x
TRL_FAC_ADD_PULSE			

Detailed input data definition (general):

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_TI_PLS_1_x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, first pulse					
TRL_TI_PLS_2_ x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, second pulse					
TRL_EOI_1_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Cylinder individual end of injection, first pulse					
TRL_EOI_2_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Cylinder individual end of injection, second pulse					
TRL_INJ_MOD_GLOBAL	-	0 ... FFH	0 ... 255	1	-
Global injection mode for all cylinders					
TRL_INJ_MOD_x	-	0 ... FFH	0 ... 255	1	-
Cylinder individual injection mode					
TRL_EOI_MIN_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest possible EOI relatet to EOI_LIM					
TRL_LV_ADD_PULSE_x	-	0 ... 1H	0 ... 1	1	-
Switch to enable additional injection pulse at homogeneous mode, single injection mode (injection update at transient operation)					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_LV_INH_INJ_x	-	0 ... 1H	0 ... 1	1	-
Flag that indicates if cylinder shut off is active or not					
TRL_SOI_MAX	-	0 ... 780H	0 ... 720	0.375	°CRK
Earliest possible SOI					
TRL_TI_ADD_DLY	-	8000 ... 7FFFH	-131.07 ... 131.03	0.004	ms
Injector dead time correction, homogeneous mode 1. pulse					
TRL_TI_ADD_PULSE_MIN	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time for additive injection pulse					
TRL_FAC_ADD_PULSE	-	0 ... FFH	0 ... 0.996094	0.0039	-
Weighting factor for injection time update at transient conditions (additive pulse)					
TRL_TRIG_INJ_UPD	-	-	-	-	-
Trigger for update of calculated injection data					
TRL_TRIG_INJ_UPD_x	-	-	-	-	-
Trigger for update of calculated injection data, cylinder individual					
TRL_NC_EOI_LIM	-	0 ... 780H	0 ... 720	0.375	°CRK
End of phasing range relatet to NC_REF_EOI_LIM					
NC_CYL_NR	-	3...8H	3...8	1	-
Number of cylinders					

Note: The prefix TRL to a data xxx (e.g. TI_1_x) indicates a I/O SW function interface for that data. In the following specification only the data labels (e.g. TI_1_x) is used to describe the required functionality.

General information:

The I/O software is responsible for opening and closing the injectors in all injection modes according to the inputs supplied. It is is applicable for HPDI and MPI engines with 3 to 8 cylinders (logical cylinder numbers 0, 1, 2, ...). It must work in an engine speed range from 30 to 8200 rpm.

Two injection pulses per cylinder and working cycle (720 °crk) should be handled.

2.16.1.1 Reference point and injection phase

The fuel can be injected over a range of 720 °CRK. The cylinder individual zero position of this range is defined via the constants NC_REF_EOI_LIM (defined in I/O SW) and NC_EOI_LIM (import data) with reference to the ignition TDC of cylinder 0, see figure below.

Related to tdc cylinder 0:

EOI_LIM_x = NC_REF_EOI_LIM + NC_EOI_LIM - x * (720 / NC_CYL_NR)

SOI_LIM_x = EOI_LIM_x + 720 °CRK

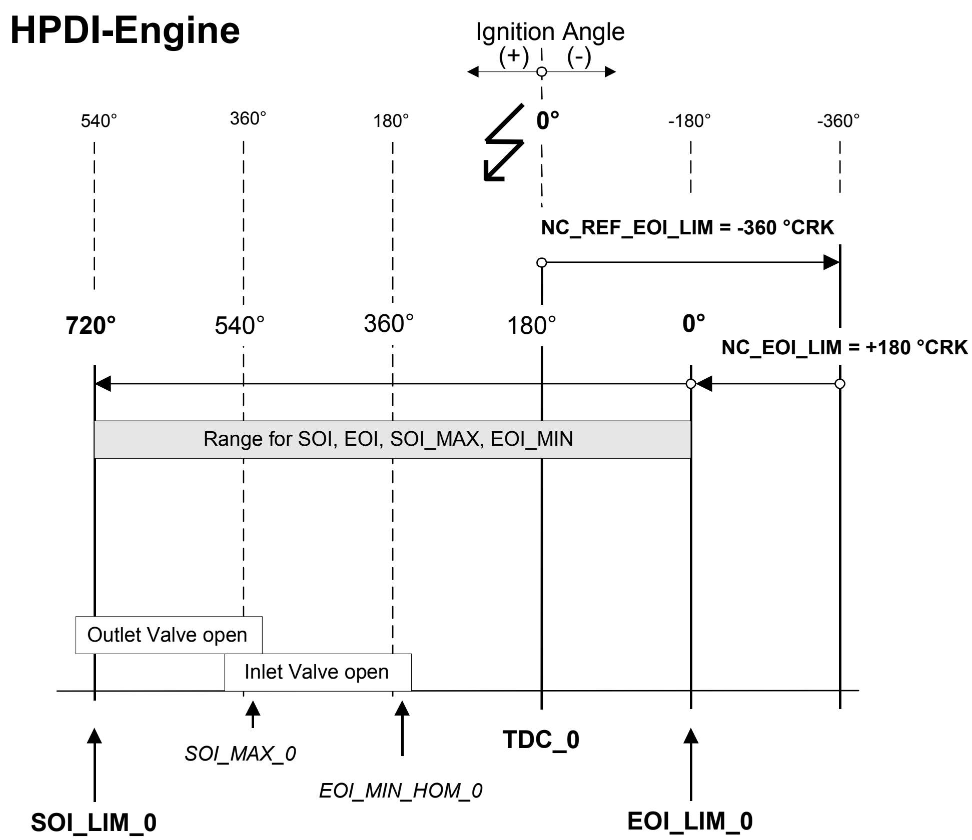


Figure 1: Example for reference point and injection phase, HPDI engine

Definition of the non calibrateable constants HPDI:

NC_REF_EOI_LIM = -360°CRK

NC_EOI_LIM = 180 °CRK

Definition of the non calibrateable constants MPI:

NC_REF_EOI_LIM = -360°CRK

NC_EOI_LIM = 540 °CRK

2.16.1.2 Calculation of Start of Injection SOI

The start of injection angles SOI_1_x, SOI_2_x, SOI_POST_INJ_x are calculated using the end of injection angles EOI_1_x, EOI_2_x, EOI_POST_INJ_x and the cylinder specific injection times TI_1_x, TI_2_x, TI_POST_INJ_x.

EOI_1_x, EOI_2_x and EOI_POST_INJ_x are counted in degrees with reference to EOI_LIM_x.

The latest engine speed information is used for the transformation between time and angle.

2.16.1.3 Injection and Calculation Updates

Injection is started at the angle SOI_1_x, SOI_2_x and SOI_POST_INJ_x, respectively.

As soon as the trigger TRIG_INJ_UPD / TRIG_INJ_UPD_x is set, the input data are updated (once per segment).

The trigger TRIG_INJ_UPD / TRIG_INJ_UPD_x is set every time EOI_LIM_x is reached.

Every injection starts earliest at SOI_MAX_x and ends latest at EOI_HOM_x.

Between different injection modes it will be switched with interface INJ_MOD_GLOBAL. It switches all cylinders at same time to a new injection mode. The INJ_MOD_x interface is able to switch between single and double injection mode for a single selected logical cylinder x. These rules will be excepted by pre injection mode, which changes automatically the cylinder individual pre injection mode to sequential mode with single injection cylinder individual. – for detail look at chapter 2.16.2.1 Pre injection.

The following injection modes are valid for interface INJ_MOD_GLOBAL:

1. Injection disable mode
2. Sequential injection mode with single injection
3. Sequential injection mode with double injection
4. Pre injection mode

The following injection modes are valid for interface INJ_MOD_x:

2. Sequential injection mode with single injection
3. Sequential injection mode with double injection

2.16.1.3.1 Single Injection

⇒ If the calculation is updated before the injector is opened and the new SOI_1_x has already passed, the injection is started at once and will be continued for TI_1_x.

(see Figure 2, Injection Update 1)

⇒ If the calculation is updated while the injector is open and LV_ADD_PULSE_x = 1, the injection time should be increased if

$TI\_1\_x_{n+1} > TI\_1\_x_{n}$

and should be decreased if

$TI\_1\_x_{n+1} < TI\_1\_x_{n}$

If the new TI of the pulse has already elapsed the injection should be stopped immediately.

(see Figure 2, Injection Update 2)

⇒ If the calculation is updated after the previous calculated fuel amount is already injected (the injector is already closed) an additional injection pulse should be performed immediately after the injection update. But only

IF
LV_ADD_PULSE_x = 1
AND
$(TI\_1\_x_{n+2} - TI\_1\_x_{n+1}) * FAC\_ADD\_PULSE + TI\_ADD\_DLY > TI\_ADD\_PULSE\_MIN$
AND
NC_T_MIN_INJ_INH has elapsed after the actual EOI_1_x
THEN
$TI\_x_{add.\ Puls} = (TI\_1\_x_{n+2} - TI\_1\_x_{n+1}) * FAC\_ADD\_PULSE + TI\_ADD\_DLY$
ELSE
Additional injection pulse is not performed
ENDIF

Remark: $TI\_1\_x_{n+1}$ is the actual performed injection time.

(see Figure 2, Injection Update 3)

⇒ If the calculation is updated while the additional pulse is performed the injection time should be increased if

$TI\_x_{add.\ Puls} < (TI\_1\_x_{n+3} - TI\_1\_x_{n+2}) * FAC\_ADD\_PULSE + TI\_ADD\_DLY$

The additional pulse should be stopped immediately at EOI_MIN.

The injection time should be decreased if

$TI\_x_{add.\ Puls} > (TI\_1\_x_{n+3} - TI\_1\_x_{n+2}) * FAC\_ADD\_PULSE + TI\_ADD\_DLY$

If the new TI of the pulse has already elapsed the injection should be stopped immediately.

(see Figure 2, Injection Update 4)

Note: In stratified mode LV_ADD_PULSE_x is always zero.Therefore neither in/decrease of the injection time nor the additional pulse is performed. In homogeneous mode LV_ADD_PULSE_x can be set by the application system.

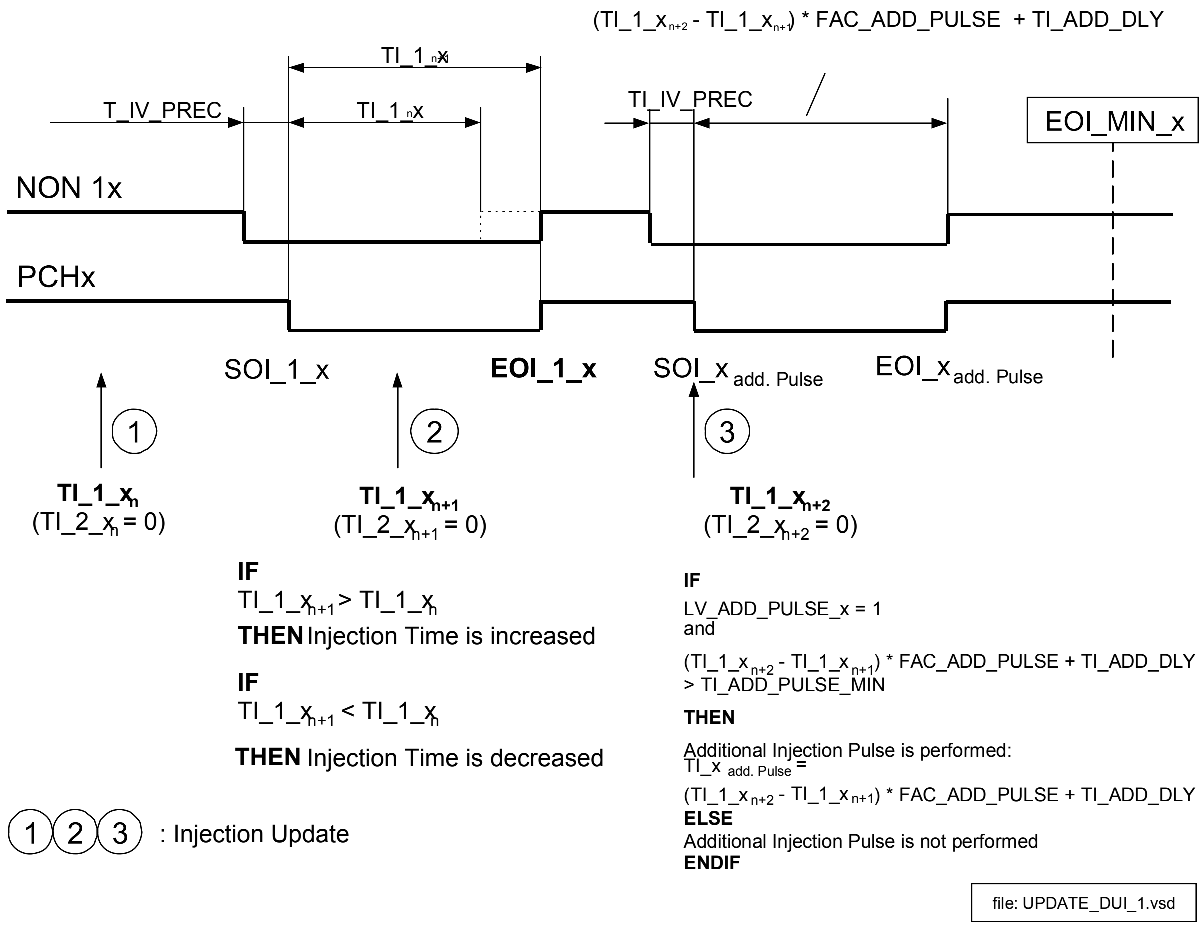


Figure 2: Injection update at single injection (HPDI Engine).
(Please Note: TRL_TI_PLS_1 = TI_1_x + T_IV_PREC for HPDI engine
TRL_TI_PLS_1 = TI_1_x for MPI engine)

2.16.1.3.2 Double Injection

⇒ If the calculation is updated befor the injector is opened for the first pulse and the new SOI_1_x has already passed, the injection is started at once and will be continued for TI_1_x.

(see Figure 3, Injection Update 1)

⇒ If the calculation is updated while the injector is open and LV_ADD_PULSE_x = 1, for the first pulse the injection time should be increased if

$TI\_1\_x_{n+1} > TI\_1\_x_{n}$

and should be decreased if

$TI\_1\_x_{n+1} < TI\_1\_x_{n}$

If the new TI of the pulse has already elapsed the injection should be stopped immediately.

(see Figure 3, Injection Update 2)

⇒ If the calculation is updated after the first pulse is performed (the injector is already closed) the second injection pulse should be performed with EOI_2_x $_{n+2}$.
But only

IF
LV_ADD_PULSE_x = 1
THEN
FAC_ADD_PULSE_CLC_x = FAC_ADD_PULSE
(means a pulse update of the preceding pulse is performed)
ELSE
FAC_ADD_PULSE_CLC_x = 0
(means a pulse update of the preceding pulse is not performed)
ENDIF

IF
(TI_1_x $_{n+2}$ – TI_1_x $_{n+1}$) * FAC_ADD_PULSE_CLC_x + TI_2_x > TI_ADD_PULSE_MIN
AND
NC_T_MIN_INJ_INH has elapsed after the actual EOI_1_x
THEN
TI_2_x = (TI_1_x $_{n+2}$ – TI_1_x $_{n+1}$) * FAC_ADD_PULSE_CLC_x + TI_2_x
ELSE
Second injection pulse is not performed
ENDIF

(see Figure 3, Injection Update 3)

If the new SOI_2_x has already passed, the injection is started at once and will be continued for TI_2_x.

The second pulse should be stopped immediately at EOI_MIN.

⇒ If the calculation is updated while the injector is open for the second pulse the injection time should be increased if
TI_2_x < (TI_1_x $_{n+3}$ – TI_1_x $_{n+1}$) * FAC_ADD_PULSE_CLC_x + TI_2_x

The second pulse should be stopped immediately at EOI_MIN.

The injection time should be decreased if
TI_2_x > (TI_1_x $_{n+3}$ – TI_1_x $_{n+1}$) * FAC_ADD_PULSE_CLC_x + TI_2_x

If the new TI of the pulse has already elapsed the injection should be stopped immediately.

(see Figure 3, Injection Update 4)

Note: In stratified mode LV_ADD_PULSE_x is always zero. In homogeneous mode LV_ADD_PULSE_x can be set by the application system.

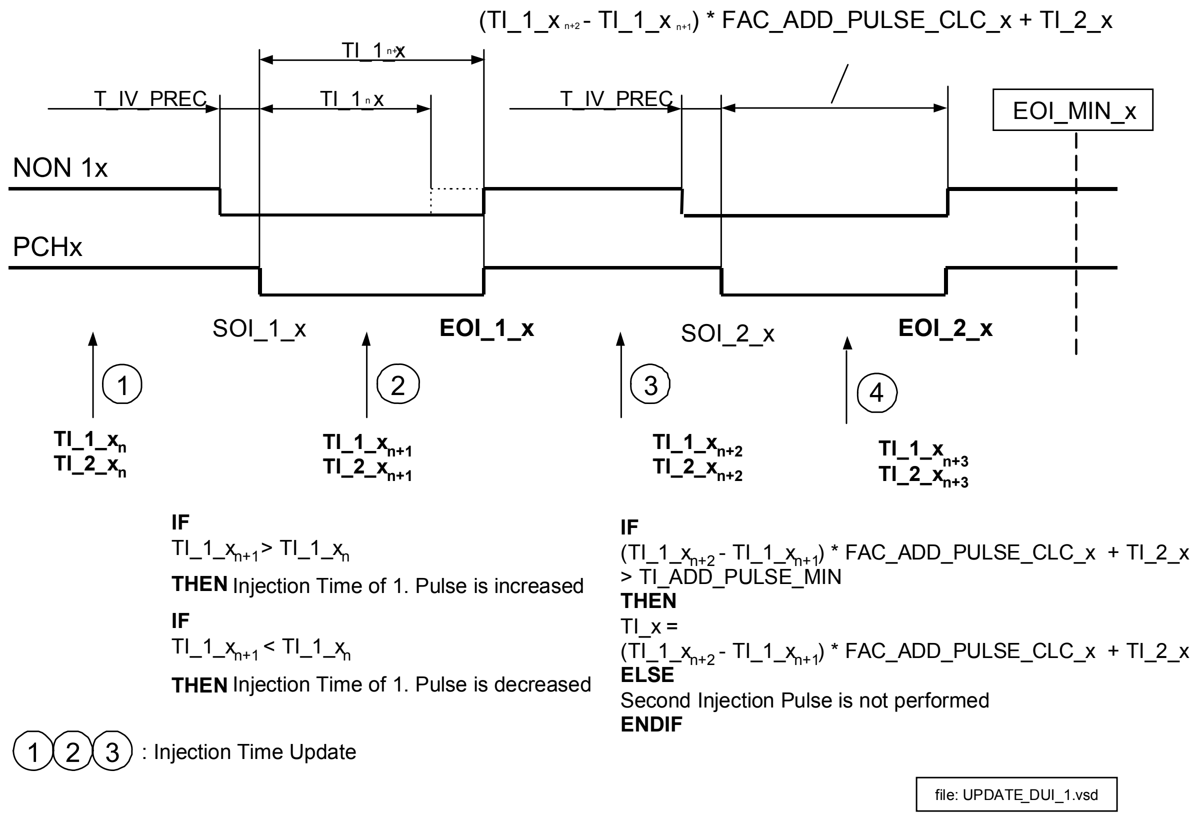


Figure 3: Injection update at double injection (HPDI Engine).
(Please Note: TRL_TI_PLS_1 = TI_1_x + T_IV_PREC for HPDI engine
TRL_TI_PLS_1 = TI_1_x for MPI engine
TRL_TI_PLS_2 = TI_2_x + T_IV_PREC for HPDI engine
TRL_TI_PLS_2 = TI_2_x for MPI engine)

Application hint:

Minimum off-time between two injections NC_T_MIN_INJ_INH is 0.1 ms. For diagnosis the minimum off-time between two injections has to be 0.25 ms).

2.16.1.4 Priorities for timing of injection

If the different requirements for the injection can not all be fulfilled simultaneously, the priorities are defined as:

1. The injection shall not start before the beginning of the injection phase (SOI_LIM).
2. The injection shall not proceed beyond the end of the injection phase (EOI_LIM).

3. The injection shall not start until the minimum delay NC_T_MIN_INJ_INH has passed after a previous injection.
4. The actual injection time shall be as close as possible to the desired injection time.
5. The actual end of injection shall be as close as possible to the desired end of injection.

2.16.1.5 Deactivation of cylinders

All cylinders can be deactivated independently. The information is contained in the variable LV_INH_IV_x which is input for this functionality. The variable is checked at every SOI_x.

If LV_INH_IV_x = 1 the injection is not performed.

At EOI_LIM_x, the deactivation status of the cylinder during the previous injection phase (0 ... 720 °CRK) is indicated at the output interface by the variable INJ_StateOfPrevInj_x.

INJ_StateOfPrevInj_x = 1 .... logical cylinder x was injected

INJ_StateOfPrevInj_x = 0 .... logical cylinder x was not injected

In case of double injection: If the first injection is deactivated by LV_INH_IV_x then the second injection is not performed.

2.16.1.6 Information about actual performed SOI, EOI and TI at the Output Interface

After an Injection was performed, the (cylinder individual) SOI, EOI and TI of that injection is indicated at the output interface as:

SOI_1_MES_x ... actual performed SOI of the first injection pulse (estimated)

EOI_1_MES_x ... actual performed EOI of the first injection pulse (estimated)

SOI_2_MES_x ... actual performed SOI of the second injection pulse (estimated)

EOI_2_MES_x ... actual performed EOI of the second injection pulse (estimated)

TI_1_MES_x ... actual performed injection time of the first injection pulse

TI_2_MES_x ... actual performed injection time of the second injection pulse

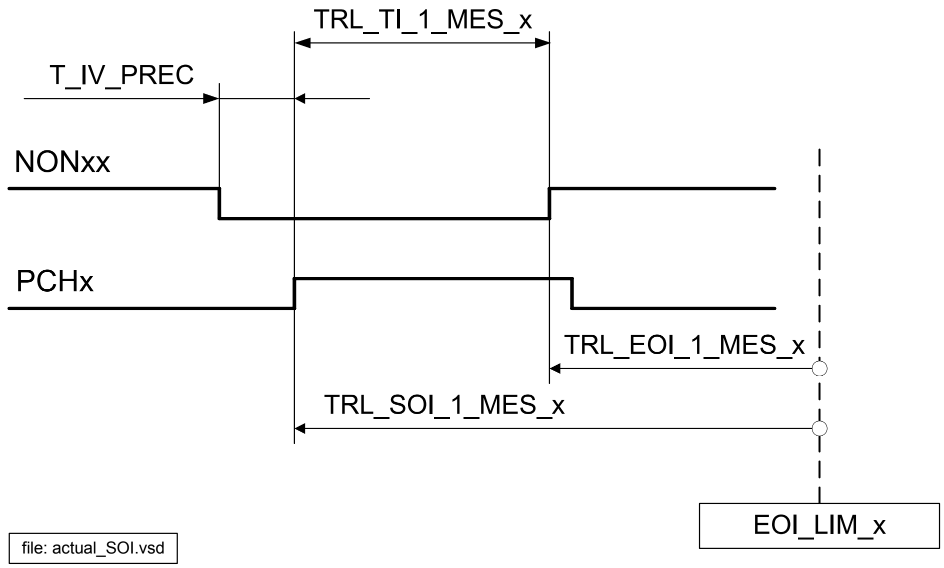


Figure 3: Actual performed SOI and EOI for the first injection pulse, example for HPDI-Engines.

(SOI is defined by the engine rotation position of the rising edge of the current controller signal PCHx. EOI is defined by the engine rotation position of the rising edge of the current controller signal NONxx.)

2.16.1.7 Dynamic angular error of SOI and EOI under transient engine operation

The maximum angular error under transient engine operation is defined as follows:

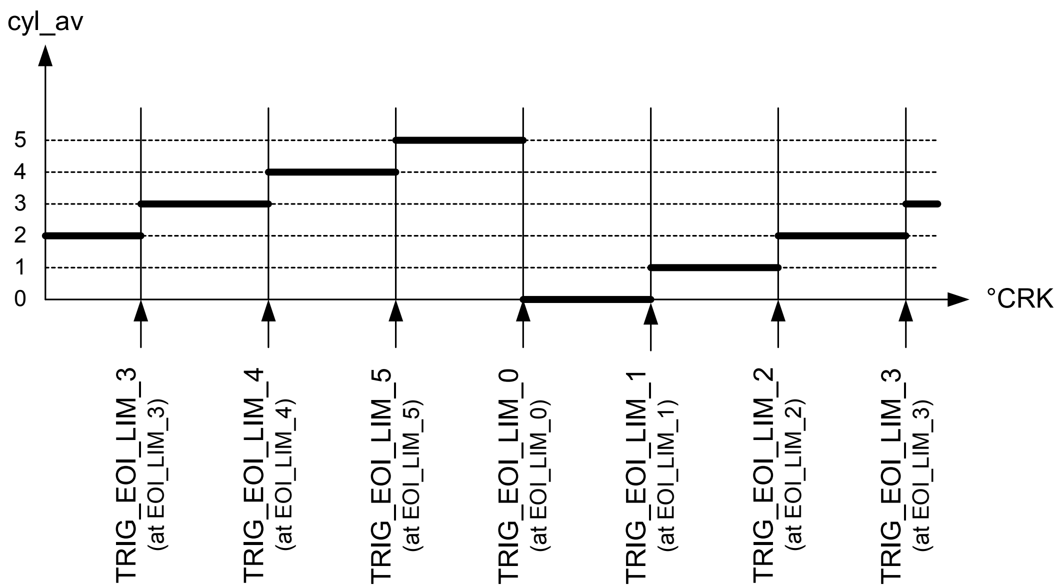
Maximum angular error = f(N, Gradient of N)

The error is measured assuming the engine speed is constant and begins to accelerate or decelerate at a constant rate. A positive error is defined if the actual angular is smaller than expected (acceleration error) while a negative error is defined if the actual angular is bigger than expected (deceleration error).

2.16.1.8 Indication of the actual number of TRIG_EOI_LIM_x

2.16.1.9 At every EOI_LIM_x a trigger TRIG_EOI_LIM_x is required.

2.16.1.10 The actual number of the TRIG_EOI_LIM_x is indicated by the parameter CYL_AV, see figure below.



File: cyl_av.vsd

2.16.1.11 Diagnosis for ATIC21 and ATIC 39

This section describes the requirements needed for the diagnosis of the injectors and injector driver.

Output data: (Diagnosis)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_LV_SCG_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual short cut to ground information					
TRL_LV_SCB_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual short cut to battery information					
TRL_LV_OC_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual open circuit information					
TRL_LV_SCG_VLD_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual information if diagnosis is valid or not (short cut to ground information)					
TRL_LV_SCB_VLD_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual information if diagnosis is valid or not (short cut to battery information)					
TRL_LV_OC_VLD_IV_x	O	0 ... 1H	0 ... 1	1	-
Cylinder individual information if diagnosis is valid or not (open circuit information)					

2.16.1.11.1 Recurrence of the output data every 10 ms

⇒ Error informations are gathered (or-ed symptom specific) until read out by the application software ASW.

At the moment when ASW is reading data from I/O software, the I/O software data will be reset. (It is possibel to have more than one electrical failure symtom reported to ASW.)

⇒ The cylinder specific variable LV_SCG_VLD_IV_x, LV_SCB_VLD_IV_x, LV_OC_VLD_IV_x are set to one (valid) if new diagnosis data from hardware are available.

⇒ If more the one error symptom is detected during the 10 ms readout period, the priority of the errors reported is:

1. short cut to battery (scb)
2. short cut to ground (scg)
3. open curcuit (oc)

⇒ SCB will shut off the device and the error will be kept by the device until a “off-slope” occurs.

⇒ SCG, OC: read out of the device erases error status and the device is not shut off.

Application hint:

The minimum injection time TI_MIN has to be calibrated to ensure proper diagnosis (for diagnosis the minimum injection time is 0.4 ms and the minimum off-time between two injections is 0.25 ms).

Configuration data: (chapter 1.1)

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_T_MIN_INJ_INH	1	0 ... 64H	0.1... 0.5	0.004	ms
Minimum off time between two injection pulses					
NC_REF_EOI_LIM	1	FC40 ... 0 H	-360 ... 0	0.375	°CRK
Offset of the EOI_LIM reference point after TDC					

2.16.2 Additional Requirements for MPI engines

2.16.2.1 Pre injection

Input data: (Pre injection)

TRL_EOI_PRE_INJ [NC_CYL_NR]	TRL_TI_PRE_INJ [NC_CYL_NR]	TRL_EOI_MIN_PRE_INJ

Detailed input data definition (Pre injection):

TRL_EOI_PRE_INJ [NC_CYL_NR]	-	0 ... 780H	0 ... 720	0.375	°CRK
End of injection for preinjection					
TRL_TI_PRE_INJ [NC_CYL_NR]	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Pre injection time					
TRL_EOI_MIN_PRE_INJ	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest end of injection angle for pre injection					

FUNCTION DESCRIPTION:

The aim is to **pre-synchronise** and start the engine as fast as possible. The experience has shown, that when the engines was switched off, it will stop in the most of the cases at the same specific engine positions. This information is used together with the information of relative engine positions PSN_ENG_SYN_MIN and PSN_ENG_SYN_MAX to pre-synchronise the engine. In the case that the difference of PSN_ENG_SYN_MIN and PSN_ENG_SYN_MAX is lower than the threshold PSN_DIF_ENG_SYN_THD then the pre injection will be enabled in the ASW.

Injection phasing for preinjection:

The cylinder individual end-of-preinjection-angle EOI_PRE_INJ [NC_CYL_NR] is calculated in the ASW and send to the IO SW. All angles are related to engine position. In addition to that, the angle EOI_MIN_PRE_INJ is used to look for the first cylinder and for the first fuel injection.

The start of injection angles for pre injection are calculated using the end of injection angles EOI_PRE_INJ [NC_CYL_NR] and the injection times TI_PRE_INJ [NC_CYL_NR] analog to chapter “Calculation of Start of Injection SOI”.

The preinjection is carried out by the IO SW in the case that the injection mode is set to pre injection.

In the case that the injection mode sequential is selected, then the fuel injection will start in sequential mode directly (this occurs when crankshaft synchronisation is detected by the ASW). The IO SW is using the standard data set for injection (TI_1_x, ...).

Actualisation of the engine position during the pre synchronisation phase

During the pre synchronisation phase it is possible, that the actual engine position can jump due to hardware events i.e. CAM edge or CRK GAP. The IO- injection – SW has to react on such changing engine positions and check if any already started injections can be finished successfully at the latest possible angle EOI_MIN_PRE_INJ. See the four cases in figure 4.

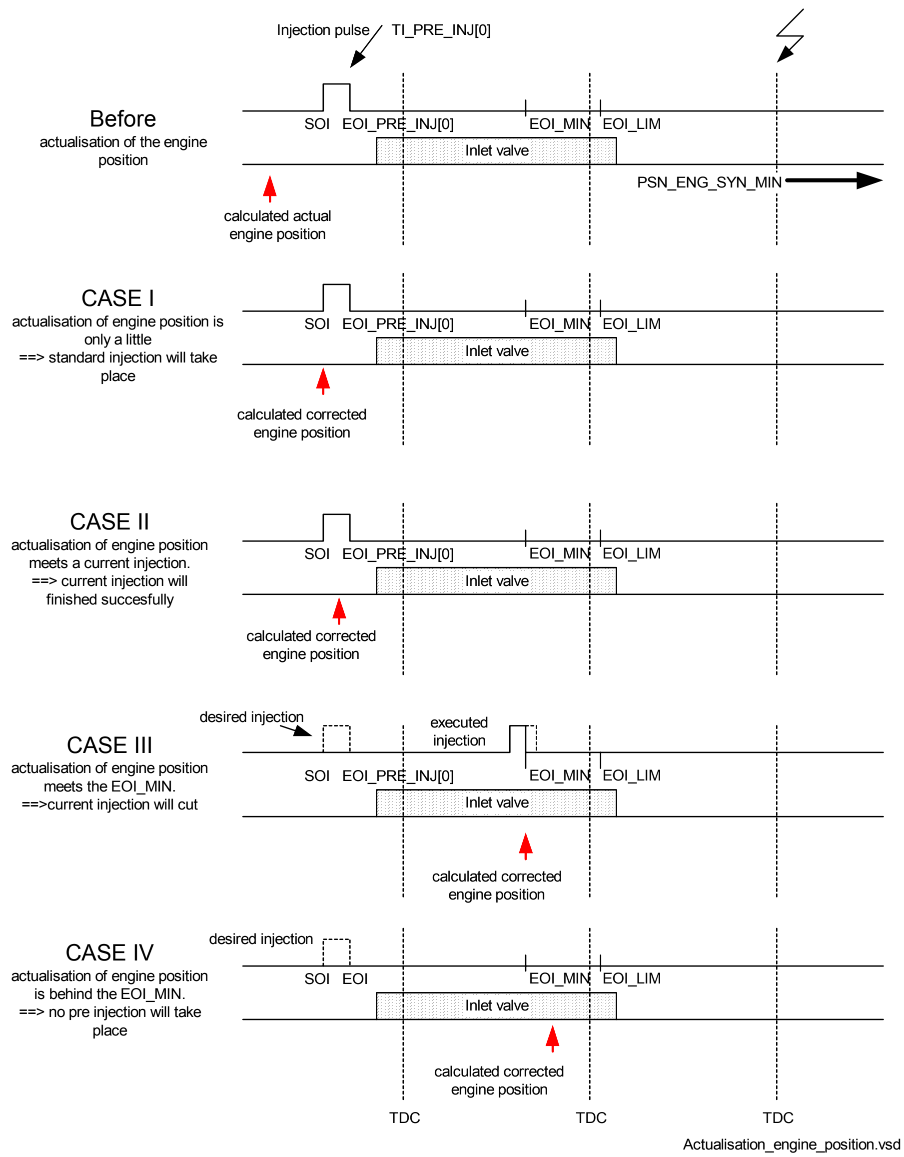


Figure 4: Injection behaviour at engine position actualisation

Formula section:

IF Injection mode pre injection is selected

THEN

the following actions has to be executed:

- First injection TI_PRE_INJ [NC_CYL_NR] will be applied with the phasing of EOI_PRE_INJ[NC_CYL_NR].
- As basis for the EOI calculation the angles are referenced to engine position.
- The latest allowed end of injection for the first pulse is EOI_MIN_PRE_INJ.
- All cylinders will be checked if TI_PRE_INJ [0] can be finished until EOI_MIN_PRE_INJ. Then choose the **earliest** cylinder, who's individual pulse position is before EOI_MIN_PRE_INJ, For this check it is assumed that the engine speed will not change during the injection.
- The following cylinders do not need the check of EOI_PRE_INJ because their pulse positions will be later.
- They are performed with TI_PRE_INJ [1 ... NC_CYL_NR –1] and EOI_PRE_INJ [1 ... NC_CYL_NR –1] in their logical order.
- For each of the NC_CYL_NR preinjections:
    - it is executed immediately if it's start of injection SOI is already past **or** it will be started at it's determined SOI,
    - it will be cut off at EOI_MIN_PRE_INJ at latest.
- After each injection the INJ_MOD_x is changed by the IO-SW to sequential mode with single injection cylinder individual.
- If an engine position actualisation occurs due to a position jump then the four following cases has to be checked corresponding to figure 1:

Case I: Injection was not started up to now.
Start with pre-injection at planned reference position.
Stop injection at EOI_MIN_PRE_INJ position or earlier.

Case II: Injection has been started.
Stop injection at EOI_MIN_PRE_INJ position or earlier.

Case III: Injection should be started during singular step.
Start injection immediately and cut at EOI_MIN_PRE_INJ or earlier.

Case IV: Injection happened during singular step.

Start injection immediately and stop at EOI_MIN_PRE_INJ or earlier.
If the actualised position is behind the EOI_MIN_PRE_INJ, skip the pre-injection and continue with standard sequential injection, i.e. pre-injection on this cylinder is missing!

ENDIF

IF the injection mode sequential is selected,

THEN

the fuel injection will start in sequential mode directly (this occurs when crankshaft synchronisation is detected by the ASW). The IO SW is using the standard data set for injection (TI_1_x, ...).

ENDIF

2.16.3 Additional Requirements for HPDI engines

2.16.3.1 Injection pulse lock mechanism for ATIC 21 driver

For the ATIC 21 driver a lock mechanism is needed to prevent overlapped pulses on the two injectors, which are connected to the same high side driver. After start of a pulse sequence on one of the two injectors the other injection driver has to be locked for injection pulse generation. The lock mechanism has to be switched to the other injector, if one of the following conditions occurs:

- Immediately after finishing the latest possible pulse of the current unlocked injector plus minimum off-time between to injections (NC_T_MIN_INJ_INH)
- Immediately after reaching the EOI_MIN of the latest pulse for the current selected cylinder, independent from having carried out or having finished the pulse under consideration of NC_T_MIN_INJ_INH

After the cylinders are switched the previous cylinder has to be locked – no injection pulses for the previous cylinder are allowed anymore.

2.16.3.2 Post Pulses (Injection after ignition TDC, e.g. for catalyst heating)

Input data: (Post Pulses)

TRL_EOI_POST_INJ_x	TRL_TI_POST_INJ_x	TRL_LV _POST_INJ_CH_x	TRL_EOI_MIN_POST _x
TRL_SOI_MAX_POST_x			

Detailed input data definition (Post Pulses):

TRL_EOI_POST_INJ_x	-	0 ... 780H	0 ... 720	0.375	°CRK
End of the post injection					
TRL_TI_POST_INJ_x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Injection time of the post injection					
TRL_LV _POST_INJ_CH_x	-	0 ... 1H	0 ... 1	1	-
Logical variable for enabling post injection					
TRL_EOI_MIN_POST _x	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest possible EOI for post injection relatet to EOI_LIM					
TRL_SOI_MAX_POST_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Earliest possible SOI for post injection related to EOI_LIM					

Post pulses are performed if LV_POST_INJ_x = 1 with the injection time TI_POST_INJ_x and with the end of injection EOI_POST_INJ_x.

LV_POST_INJ_x is checked at every TRIG_INJ_UPD_x and every EOI of the pulse before start of injection of a post pulse (SOI_POST_INJ_x).

Post pulses are not started before SOI_MAX_POST_x and are stopped immediately at EOI_MIN_POST_x.

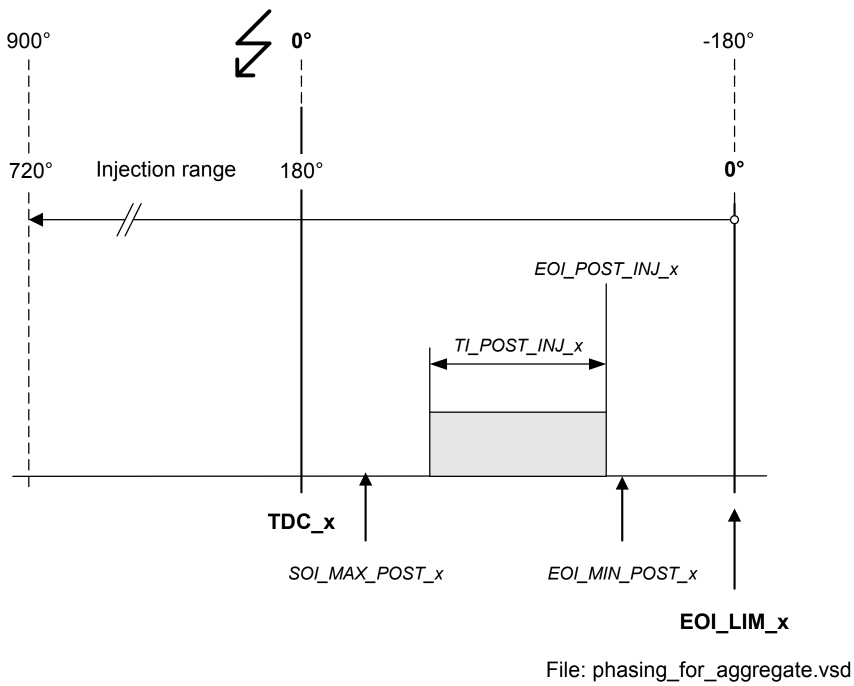


Figure 4: Phasing of post pulses

2.16.3.3 Injector current control (ATIC 21)

Output data: (Injector current control)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_VBOOST_BAS	O	0 ... 3FFH	0 ... 4.999	5/1023	V
Boost voltage					

Input data: (Injector current control)

TRL_T_IV_PREC	TRL_T_HLD_IVP	TRL_T_END_IVP	

Detailed input data definition (Injector current control):

TRL_T_IV_PREC	-	0...2FFH	0...3	0.004	ms
Injector current precharge time					
TRL_T_HLD_IVP	-	0...7H	0.192..0.416	0.032	ms
Injector current Peak Hold Time					
TRL_T_END_IVP	-	0..1FH	0..0.062	0.002	ms
Injector current Fast Decrease Time					

The specification refers to the injector driver ATIC 21 in combination with solenoid injectors. The current to control the injector is variable and can be influenced via the application software. The current profile is described by

- the **"Precharge Time" T_IV_PREC**, necessary to limit the energy which comes from the DC/DC-Converter (ECU integrated) and is needed during the "Peak Hold Time".
- the **"Peak Hold Time" T_HLD_IVP** (or boost phase) to guarantee a safe opening of the injector up to a defined fuel pressure,
- the **"Fast Decrease Time" T_END_IVP**, which enables a fast reduction of the energy stored in solenoid coil down to the level of the Hold Current. This guarantees almost a constant injector closing time (for a certain fuel pressure) down to very short injection times (~ 0.4 ms) and therefore a better Linerar Flow Range,
- the **"Hold Phase"** with a constant current, high enough to keep the injector open,

see Figure 5.

The boost voltage is the voltage from the DC/DC-Converter which is active at the injector from start of injection until the injector current reaches its peak value, see figure 5.

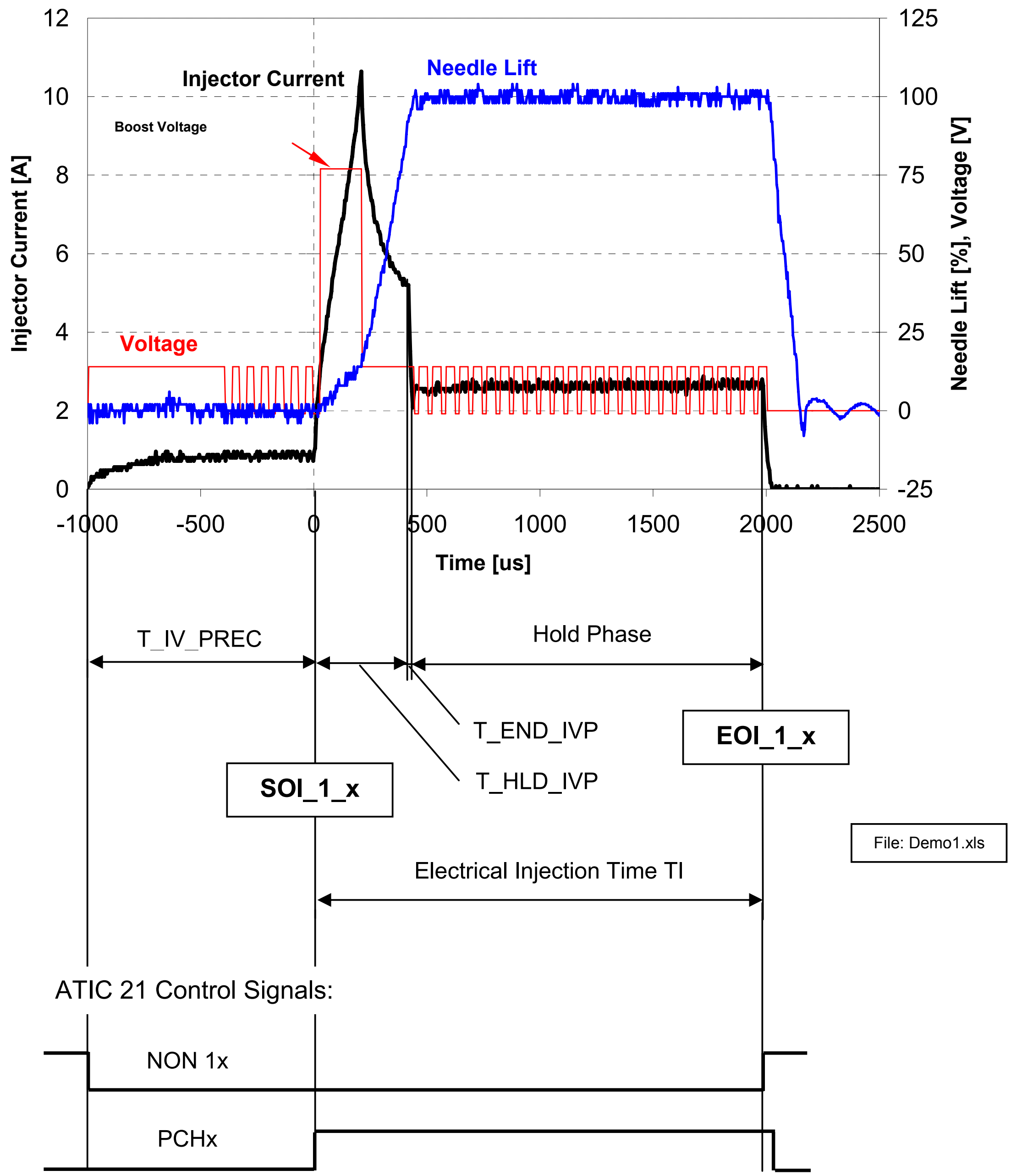


Figure 5: Current, voltage and needle lift at the injector

2.17 Spark Advance and Dwell Output

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TD_FAC_MAX	O/V	1...FFH	1/128...255/128	1/128	-
Maximal dwell time control					
TD_FAC_MIN	O/V	1...FFH	1/128...255/128	1/128	-
Minimal dwell time control					

Input data:

LV_ ERR_CAM	VB	LV_ST	N_32
LDPM_N_32_1_IGRE	LV_SYN_ENG	NC_CYL_NR	

Import actions:

ACTION_INFR_SetIgnCtl(IN <>)

FUNCTION DESCRIPTION:

This function sets the angular position of the dwell time turn on in order to have the time to set up the necessary current in the ignition coil.

The strategy is based on a priority of the ignition angle. If dwell time priority is requested the minimum dwell time is equal to maximum dwell time.

The strategy respects ignition coils with dual and single outputs. This function could be used independently of the number of cylinders (n) and the geometry of the crankshaft target wheel.

This function could be used for homogeneous combustion mode and for stratified combustion mode. The calculation of the spark advance is in both cases the same.

All of the functions described in this section operate over the entire engine speed range.

This specification includes two mode following the IGRE version: PI or PI + DI

Application conditions:

Initialisation: at reset

Activation: LV_SYN_ENG = 1 Engine synchronised

2.17.1 Minimum / Maximum dwell time calculation

FUNCTION DESCRIPTION:

The two values are calculated independently of all environment factors. The aim is to have a larger window at start and low engine speed to obtain a better flexibility to apply the requested ignition angle.

These 2 values are expressed in term of min/max factor to be applied to the nominal dwell time and transferred to the Basic SW

The cylinder individual nominal Dwell Time to be applied requested by ASW function is defined in TD_IGC[x] taking into account the combustion mode switch

Application conditions:

Initialisation: TD_FAC_MIN = TD_FAC_MAX = 1 *at reset*

Recurrence: 10 ms

Formula section:

If LV_ST = 1

Then *(Start mode)*

If VB > C_TD_VB_MAX

Then TD_FAC_MAX = 1

TD_FAC_MIN = Min[C_TD_FAC_ST_MIN , 1]

Else TD_FAC_MAX = Max[C_TD_FAC_ST_MAX , 1]

TD_FAC_MIN = Min[C_TD_FAC_ST_MIN , 1]

Else *(run mode)*

If VB > C_TD_VB_MAX

Then TD_FAC_MAX = 1

TD_FAC_MIN = Min[IP_TD_FAC_MIN , 1]

Else TD_FAC_MAX = Max[IP_TD_FAC_MAX , 1]

TD_FAC_MIN = Min[IP_TD_FAC_MIN , 1]

Endif

In every case: TD_IGC[x] * TD_FAC_MAX <= NC_TD_LIM_MAX

TD_IGC[x] * TD_FAC_MIN >= NC_TD_LIM_MIN

NC_TD_LIM_MIN is defined for dwell time protection in case of advance ignition changes during the dwell time application. If the dwell time is equal to this minimal limit value, the advance ignition could be not guaranteed

NC_TD_LIM_MAX is defined for the maximum dwell time in case of advance ignition changes during the dwell time application. If the dwell time is equal to this maximal limit value, the advance ignition could be not guaranteed

Note: In any case, TD_FAC_MIN has to be smaller than 1 and TD_FAC_MAX greater than one to be sure to respect the requested dwell and avoid inconsistencies
care has to be taken with the setting of NC_TD_LIM_MAX in order to avoid Dwell Time / coil charging recovery:
Dwell time can start at the earliest at 288°CRK before the TDC.
NC_TD_LIM_MAX should be tuned in consequence to respect coil charging time no recovery and coil capacity

2.17.2 Half static of full static

Recurrence: engine stop to engine run

Formula section:

Following NC_IGC_CONF = Half or full, the ignition is half static controlled or full static controlled

ACTION_INFR_SetIgnCtl(NC_IGC_CONF)

2.17.3 Limp home without cam signal

Recurrence: engine stop to engine run

Formula section:

In case of CAM failure, the cylinder in combustion cannot be unequivocally identified. Engine phasing might be wrong. In such a case, a limp home mode function on Ignition is provided in order to have the possibility to drive the engine. Nevertheless, in this case, since the right cylinder phasing is not known, Ignition coil have to be driven in Half-Static mode to be sure to execute at least an ignition on the cylinder in combustion.

Decision to use such Limp Home functionality will depend on the setting of LC_IGC_LIH_CONF. If LC_IGC_LIH_CONF = 0, no Limp Home is provided in CAM failure and Ignition is stopped. If LC_IGC_LIH_CONF = 1, then Half-Static Ignition mode is done in case of CAM failure

If LV_ERR_CAM is active, the ignition coil control will be executed in half-static ignition-coil control mode if selected by the settings of LC_IGC_LIH_CONF

In such a case (*half-static mode activation in CAM failure*) the same Ignition and Dwell Time will be applied to the 2 Crank synchronous cylinders (*eg. on a 4 cyl engine, cyl. 1 and 4 and cyl. 2 and 3 will be fed with the same Ignition and Dwell Time*)

The limp home function could be applied only if the number of cylinders is even (2, 4, 6, ...)

If NC_IGC_CONF = full

Then **If** LV_ERR_CAM = 1 And LC_IGC_LIH_CONF = 1

Then **ACTION_INFR_SetIgnCtl**(HALF_STATIC)

Else **ACTION_INFR_SetIgnCtl**(FULL_STATIC)

EndIf

Else **ACTION_INFR_SetIgnCtl**(HALF_STATIC)

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_TD_ FAC_ST_ MIN	1	1...FFH	1/128...255/128	1/128	-
Factor for minimum dwell time					
C_TD_ FAC_ST_ MAX	1	1...FFH	1/128...255/128	1/128	-
Factor for maximum dwell time					
C_TD_VB_MAX	1	0…FFH	0...26	0,102	V
Maximum battery voltage to activate MPL (typical value : 16V)					
IP_TD_FAC_MAX	8	1...FF	1/128...255/128	1/128	-
LDPM_N_32_1_IGRE	8	0...FF	0...8160	32	rpm
Max dwell time factor					
IP_TD_FAC_MIN	8	1...FF	1/128...255/128	1/128	-
LDPM_N_32_1_IGRE	8	0...FF	0...8160	32	rpm
Min dwell time factor					
LC_IGC_LIH_CONF	1	0...1	0...1	1	-
Configuration Choice for Half static Mode in case of CAM limp home: 0 = no Limp Home - 1 = Half Static					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_TD_LIM_MIN	1	0...FFFFH	0...262,14	0.004	ms
NC_TD_LIM_MIN (Typical value 1 ms)					
NC_TD_LIM_MAX	1	0...FFFFH	0...262,14	0.004	ms
NC_TD_LIM_MAX - Typical value 16 ms					
NC_IGC_CONF	1	1...2	1...2	1	-
Half static or full static (Typical value for debug = full static)					

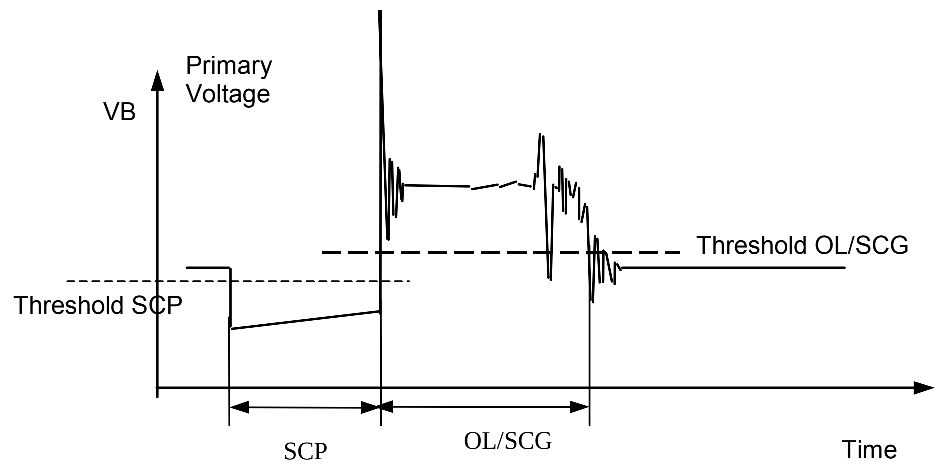
2.18 Ignition output diagnosis (for ATM 46 / Combined "ASIC")

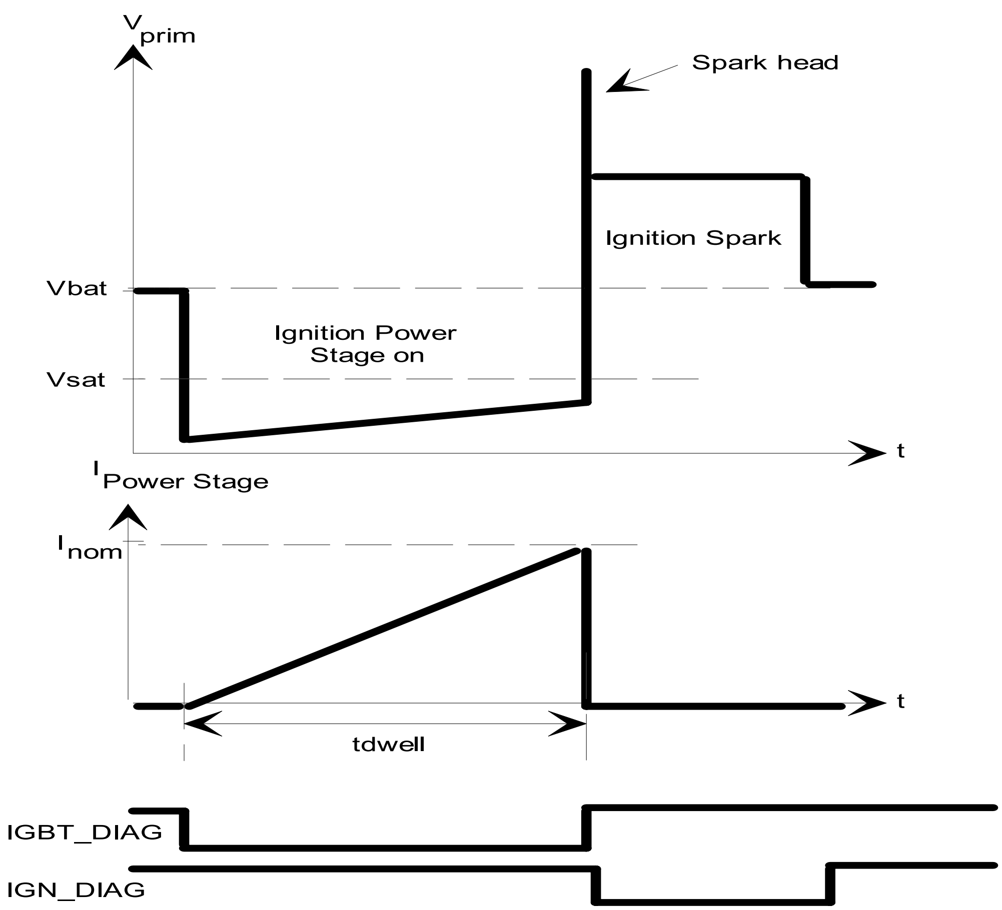
Each ATM46 could analyze 4 ignition coils. If diagnostics for more the than 4 coils are necessary, a parallel ATM46 has to be used.

The diagnosis is based on ignition coil primary over voltage measurement.

The input signal of the ATM 46 is the IGBT collector voltage.

In case of an ignition system with single output coils, the function does not work in camshaft limp home mode.





2.18.1 IGBT Protection (with “ATM46” / Combined “ASIC”)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_SCP_IGC[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Diagnostic of primary short circuit to battery (cylinder selected)					

Input data:

LV_SYN_ENG	NC_CYL_NR	SEG_NR	LV_CDN_DIAG_IGC_SCP[NC_CYL_NR]

Import actions:

ACTION_INFR_GetIgnScpDiagAtm46Atic71(IN <>,OUT<>)

FUNCTION DESCRIPTION:

The aim of the present function is to protect the ignition power stage against short circuit to battery (SCP).

The purpose of ignition power stage diagnostic is to detect a short circuit to battery voltage condition (SCP). At each Dwell Time On event, the ignition power stage is switched on. In normal condition, the current increase in the ignition power stage is limited by the inductance of the ignition coil solenoid.
In case of a short circuit to battery, the coil external device does not limit the current in the ignition power stage. In this situation the current is only limited by the current capability of the ignition power stage. This power dissipation would destroy the power stage. From the Dwell Time ON event in SCP condition, the IGBT’s can withstand this overload current only for a very short time (*750µs for IGBT without shunt*). So, it is necessary to switch off this IGBT transistor for safety and protection reasons by software (*IGBT deactivation by switching off the charging command in SW*) in case of an overload condition and after a certain time delay has elapsed (NC_INI_CTR_DEAC). This time delay is necessary in order to filter parasitic spikes. On the other hand this time delay must not exceed the maximum rating of the IGBT’s.

For ignition power stage diagnostic, the voltage across the ignition power stage is compared to a fixed threshold voltage. This threshold voltage is chosen to be higher than the maximum saturation voltage of the ignition power stage during a dwell time in normal operation. If there is no short to battery voltage condition, the voltage across the ignition power stage drops below this fixed threshold voltage at each dwell time. In case of a short to battery voltage failure, this voltage is not (or only for a short time) passed. A micro controller monitoring the ignition power stage diagnostic output is able to detect this fault.

This ignition power stage diagnostic circuit is designed for an engine management system including four ignition power stages. This four diagnostic information can be evaluated at one output by having a logic OR function of them.

After each deactivation from this protection function, the cut off IGBT will be reactivated again in the next ignition cycle

The function could be used for ignition systems with single-output coils and double-output coils. The IGBT protection function is cylinder selectively deactivated in case of single-output coil and linked dual cylinders for ½ static coils.

Application conditions:

Activation: *LV_SYN_ENG = 1*

Formula section:

Initialization: at reset
LV_SCP_IGC[x] = 0

Update rate: segment

System description:

The ATM46 is used to generate a signal evaluated by the SW for overload condition detection.

To perform the IGBT protection and SCP diagnostic, the BSW provides one functional service (BSW driver – one driver for one ATM46 protection and SCP diagnostic line)

As only one IGBT diagnosis output signal from the "ATM 46" exists (all channels are linked by a wired "OR") and only one BSW functional service (*this BSW functional service used is described in the following paragraph*) is used in connection to this diagnosis output signal line, the function is limited to non-overlapping dwell time pulses. In case of overlapping Dwell Time, the last TD ON event that occurs will re-trigger the call of this function and thus be considered for diagnosis purpose.

Formula section:

ACTION_INFR_GetIgnScpDiagAtm46Atic71(SEG_NR -2,LV_SCP_IGC[SEG_NR-2])

2.18.1.1 Ignition Actuator Tests Diagnosis

Formula section:

Initialization: LV_SCP_IGC[x] = 0 at reset

System description:

For ignition coil actuator tests purpose, the SCP diagnostic function has to be enabled and activated upon the tests requests

The same case and comments (see before) apply here also.

Formula section:

For x = 0 to NC_CYL_NR – 1
 If LV_CDN_DIAG_IGC_SCP[x] = 1
 Then **ACTION_INFR_GetIgnScpDiagAtm46Atic71**(x,LV_SCP_IGC[x])
 EndIf
EndFor

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_INI_CTR_DEAC	1	0...FFFFH	0...262,14	0,004	ms
Time delay after IGBT threshold response before switching OFF the IGBT - typical value 600 us					
NC_IGBT_CUT_OFF_T	1	0...FFFFH	0...262,14	0,004	ms
Inhibition of "IGBT protection cut off function"					

2.18.1.2 OL/ SCG (Burn Time measurement) - Acquisition of the burn time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_OL_IGC[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Diagnostic of primary open load (cylinder selected)					
LV_SCG_IGC[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Diagnostic of primary short circuit to ground (cylinder selected)					
V_DUR_IGC[NC_CYL_NR]	V	0...FFFFH	0...262,14	0,004	ms
Burn time duration for diagnosis validation					

Input data:

LV_SYN_ENG	N_32	MAF	NC_CYL_NR

Import actions:

ACTION_INFR_GetIgnScgDiagAtm46Atic71 (IN <>,IN<>,OUT<>,OUT<>)

Description:

This ignition diagnosis function evaluates the primary over voltage duration **V_DUR_IGC[x]** provided after the ignition coil cut off.

The aim of this spark burning diagnostic function is to measure the spark duration by detecting the presence of a spark on an ignition plug. The over-voltage at the primary coil of an ignition solenoid is equal to the voltage across the spark gap of the ignition plug divided by the transmission ratio of the ignition solenoid and superimposed on the battery voltage. The time where the voltage at the primary coil of the ignition solenoid is higher than a battery voltage dependent threshold voltage (V_{BD_TH}) gives the spark duration.

In the moment, when the ignition power stage (IGBT) is switched OFF *(ie. at TD OFF event occurrence)*, the voltage V_C at the collector of the IGBT rises, due to self-induction, and ignition occurs. As long as the ignition spark is burning, the voltage is well exceeding the battery voltage.

By means of a hardware defined threshold (V_{BD_TH}; typical value: V_{Bat} + 3V) the burn condition is detected and a matching signal is generated, *eg. Here by the combined ASIC ATM46*. This signal is used to measure the overall burning duration.

The Ignition Burning diagnostic signal IGN_DIAG is generated by this combined ASIC. This time for the burning spark IGN_DIAG is calculated via the duration of the diagnostic signal staying above the threshold of V_{BD_TH}.

This ATM46 ignition diagnostic circuit is designed for an engine management system including up to 4 ignition coils/solenoids. This four diagnostic information are then evaluated on one output by applying a logic NOR function on them. The output signal IGN_DIAG of the ignition diagnostic circuit is processed by the µP.

Due to combustion chamber turbulence the spark of an ignition plug is not stable. The result is an unstable primary voltage at the ignition solenoid. Therefore a filter in the output signal IGN_DIAG is necessary to prevent the µP from high frequency inputs.

The ignition coil primary over voltage on the ignition output is roughly equal to the voltage across the spark gap divided by the ignition coil transformation ratio, superimposed on the battery voltage.

The Burn Time measurement is used today only for Spark / Combustion Diagnosis Purpose. It is a rough measurement of the effective Burn Time.

Application conditions:

Activation: LV_SYN_ENG = 1

Initialisation: LV_OL_IGC[x] = 0

LV_SCG_IGC[x] = 0

at reset

V_DUR_IGC[x] = 0

Recurrence: segment

Formula section:

ACTION_INFR_GetIgnScgDiagAtm46Atic71(SEG_NR-2,ID_V_DUR_IGC_MIN, V_DUR_IGC[SEG_NR-2], LV_SCG_IGC[SEG_NR-2])
LV_OL_IGC[SEG_NR-2] = LV_SCG_IGC[SEG_NR-2]

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
ID_V_DUR_IGC_MIN	8x6	0...FFFFH	0...262,14	0,004	ms
LDP_N_32_ID_V_DUR_IGC_MIN	8	0...FFH	0...8160	32	rpm
LDP_MAF_ID_V_DUR_IGC_MIN	6	0...FFH	0...1389	5.44	mg/stk
Minimum duration of the Burntime to detect ignition errors					

2.19 Fuel consumption signal output FCO

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FCO	V/O	0...FFFFH	0...8388,6	0,128	µl
Fuel consumption quantity for fuel consumption signal output.					
FCO_NR	-	0...5H	0...5	1	1/10msec
Number of pulses for fuel consumption determination.					

Input data:

FCO_SUM			

FUNCTION DESCRIPTION:

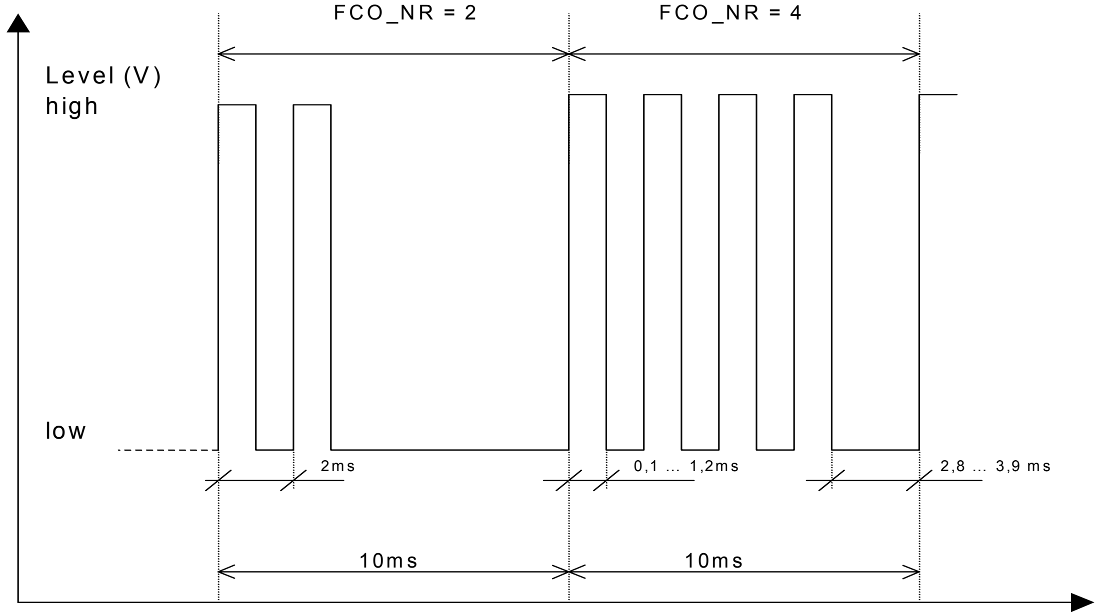
General information:

Application recurrence **: 10 ms**

The ECU generates a fuel consumption output signal (FCO) to be dispatched to other ECUs (Transmission Control Unit, Dashboard ...).

The information about the actual fuel consumption is passed to the other ECUs every 10 ms via a pulse code. Each pulse represents a fuel quantity of 80 µl. The pulse is active at level „high“. The high phase can last between 0,1 and 1,2 msec.
The period of one pulse lasts 2 ms. For each 10 ms duration the maximum possible number of pulses FCO_NR is 5.
As soon as all injectors are switched off (PUC..) the level of the signal is set to „low“.

The remaining injection quantity, which is not covered by the current pulses (during the 10 ms), is memorized for the following calculation.



Initialisation :

The output signal is set to “low” at key on after ECU initialisation.

The signal is deactivated when the house keeping phase has elapsed.

Formula section:

$$FCO = (FCO\_SUM_{n} - FCO\_SUM_{n-1})$$

$$FCO\_NR = FCO / 80\ \mu l$$

FCO_NR is limited to $0 \leq FCO\_NR \leq 5$.
The remaining fuel quantity is memorized for the next calculation (n+1).

2.19.1.1 Engine speed signal output ESS

2.19.1.1.1 Basic software

FUNCTION DESCRIPTION:

General information:

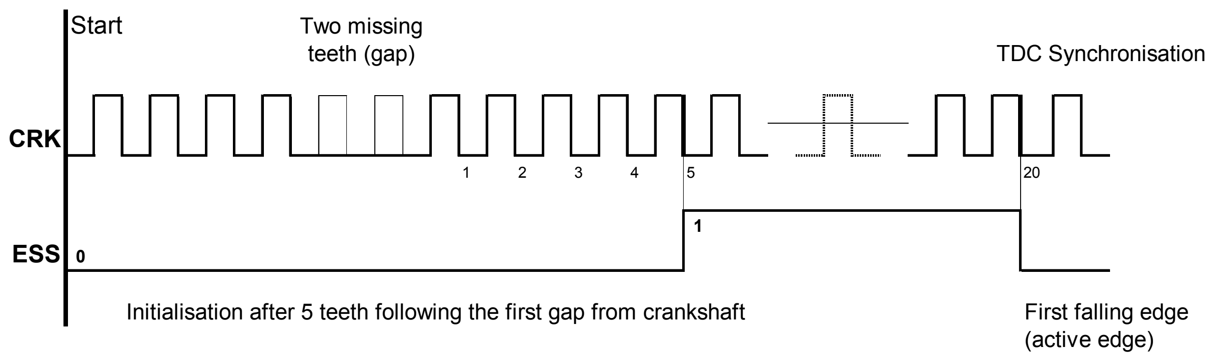
This specification is applicable for a 4 cylinders engine.

The ECU generates an engine speed signal (ESS) to be dispatched to other ECUs (Transmission Control Unit, Dashboard ...)

This output is only activated when the engine is running and the signal is representative of engine speed. It is performed with a fixed logic state ratio and a variable frequency proportional to the engine speed.

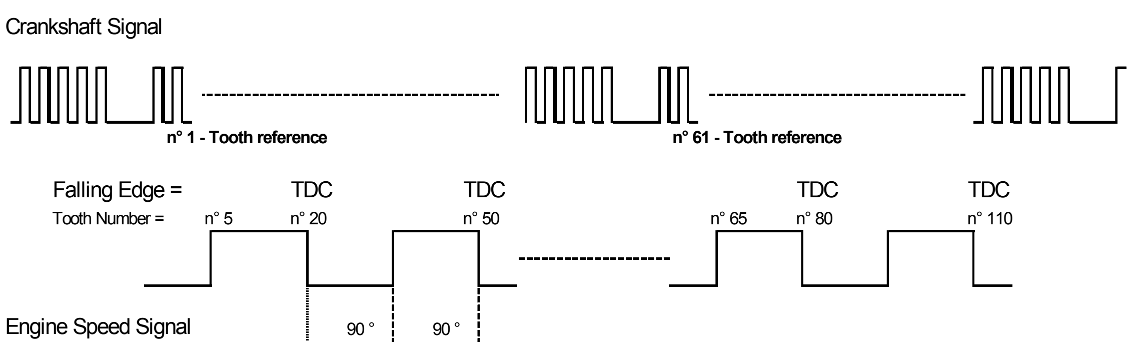
Initialisation :

The output signal is initialised with a low logic state (0).



Formula section:

The active edge on the engine speed signal is the falling edge, corresponding to the TDC location.



2.20 Acquisition of Air conditioning Pressure Signal

2.20.1 Air conditioning Pressure Voltage Signal

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_ACP_MES	V/O	0...3FFH	0...4,995	4.88e-3	V
Air conditioning Pressure Voltage Signal (before electrical Diagnosis)					

Input data:

ACP_BAS			

FUNCTION DESCRIPTION:

General information:

The raw value for air conditioning pressure voltage signal (ACP_BAS) is measured by continuous conversion (10 bits) every 10 msec.

The value of the numeric conversion is adapted to take into account of the electronic component drift. The result of this adaptation must be linearized according to the pressure sensor response.

The corresponding value of the first measurement after hardware reset is used for initialization.

Formula section:

The voltage from the air conditioning pressure voltage signal is converted to the measured AC pressure V_ACP_MES.

$$V\_ACP\_MES = (1/10) * \sum_{n=1}^{n=10} ACP\_BAS_n$$

2.20.2 Supply Voltage signal for Air conditioning Pressure Tranducer

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VCC_SENS_SUB_MES	V/O	0...7FFH	0...9.9951	4.88e-3	V
Supply voltage to calculate ACP_RATIO					

Input data:

VCC_SENS_SUB_BAS			

FUNCTION DESCRIPTION:

General information:

The raw value for APT supply voltage signal (VCC_SENS_SUB_BAS) is measured by continuous conversion (10 bits) every 10 msec.

Formula section:

The voltage signal is converted to the VCC_SENS_SUB_MES.

VCC_SENS_SUB_BAS delivered from HW is half of the real supply voltage.

$$VCC\_SENS\_SUB\_MES = [\,(1/10) * \sum_{n=1}^{n=10} VCC\_SENS\_SUB\_BAS_n\,] * 2$$

2.21 Get SLV VVT Electric Diagnosis Result - Requirements to Infrastructure

Export actions:

ACTION_INFR_GetElDiagSlvVvt(IN <Vvt>, OUT <Cdn_Diag>, OUT <Err_Diag>)
This action reads the failure and condition information for a symptom of the SLV VVT power stage of camshaft Vvt

Description for actions:

<table>
<tr><th colspan="6">ACTION_INFR_GetElDiagSlvVvt(<Vvt>, <Cdn_Diag>, <Err_Diag>)</th></tr>
<tr><td colspan="6">This action reads the failure and condition information for a symptom of the SLV VVT power stage of camshaft Vvt.
The readout of the power stage is performed autonomous and the information is gathered.
When calling this Action the information inside the Infrastructure is reset after returning the OUT parameters.</td></tr>
<tr><th>Parameter</th><th>Type</th><th>Hex. Limits</th><th>Phys. Limits</th><th>Resol.</th><th>Unit</th></tr>
<tr><td>Vvt</td><td>IN</td><td>0H
1H
2H
3H</td><td>VVT_IN_1
VVT_EX_1
VVT_IN_2
VVT_EX_2</td><td>1</td><td>-</td></tr>
<tr><td colspan="6">Camshaft specification for reading out the electric diagnosis</td></tr>
<tr><td>Cdn_Diag</td><td>OUT</td><td>0...7H</td><td>0...7</td><td>1</td><td>-</td></tr>
<tr><td colspan="6">Diagnosis condition for symptom:
Bit 0: Set, if diagnosis condition for symptom SCP (SYM_0) is fulfilled
Bit 1: Set, if diagnosis condition for symptom SCG (SYM_1) is fulfilled
Bit 2: Set, if diagnosis condition for symptom OC (SYM_2) is fulfilled</td></tr>
<tr><td>Err_Diag</td><td>OUT</td><td>0H
1H
2H
4H</td><td>NO_SYM
SYM_0
SYM_1
SYM_2</td><td>1</td><td>-</td></tr>
<tr><td colspan="6">Raw value of error symptom.
The relevant bit is set, if the error has beed detected.</td></tr>
</table>

FUNCTION DESCRIPTION:

General information:

This Action returns the result of the electric diagnosis of the VVT solenoid control valve power stage dedicated to the camshaft given by the parameter Vvt.

- The device readout is performed autonomous by the Infrastructure each 10 ms.
- The error informations are gathered in the Infrastructure (or-ed symptom, camshaft specific) until the Application reads out the information by calling ACTION_INFR_GetElDiagSlvVvt().
- After having read out the information by calling ACTION_INFR_GetElDiagSlvVvt(), the data inside the Infrastructure are reset. Resetting of Cdn_Diag avoids unambigious results in case of too short calling reccurrence of ACTION_INFR_GetElDiagSlvVvt(): Reset Cdn_Diag indicates, that the gathering of the information is not completely finished.

Requirements for ACTION_INFR_GetElDiagSlvVvt:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Vvt	1	1	1		Return parameters Err_Diag and Cdn_Diag

					belonging to electrical diagnosis of related camshaft: VVT_IN_1, VVT_IN_2, VVT_EX_1, VVT_EX_2
Cdn_Diag	-	-	<bit coded>	Err_Diag	Diagnosis condition for each symptom **Bit 0:** diagnosis condition for symptom SCP (SYM_0) **Bit 1:** diagnosis condition for symptom SCG (SYM_1) **Bit 2:** diagnosis condition for symptom OC (SYM_2) The relevant bit is set, if the condition for a valid diagnosis is fulfilled.
Err_Diag	-	-	<bit coded>	Cdn_Diag	Bit coded result of each symptom (SYM_0=SCP, SYM_1=SCG, SYM_2=OC) **0H** = NO_SYM **1H** = SYM_0 **2H** = SYM_1 **4H** = SYM_2 The relevant bit is set, if the error has beed detected.

Diagnosis: ACTION_INFR_GetElDiagSlvVvt() returns the electric diagnosis

Coincidence requirements: No coincidence requirements to other events

2.22 Acquisition of Oil Temperature

2.22.1 Basic Software

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOIL_MES	V/O	0...C8H	-40...160	1	°C
Oil temperature signal measured value					
V_TOIL	V/O	0...FFH	0...4.9805	5 / 256	Volt
Oil temperature voltage raw value (8 bits).					

Input data:

TOIL_BAS			

FUNCTION DESCRIPTION:

General information:

This function is applicable with a NTC type of sensor.

The raw value of the oil temperature (TOIL_BAS) is measured by continuous conversion (10 bits) every 1 msec.

The value of the numeric conversion is adapted to take into account of the electronic component drift. The result of this adaptation must be linearized according to the temperature sensor response.

The corresponding value of the first measurement after hardware reset is used for initialization.

Formula section:

The voltage from the oil temperature sensor is converted to the measured oil temperature TOIL_MES using the map IP_TOIL__V_TOIL.

V_TOIL ← TOIL_BAS / 4 (8 bits)

TOIL_MES = IP_TOIL__V_TOIL

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TOIL__V_TOIL	16	0...C8H	-40...160	1	°C
LDP_V_TOIL__IP_TOIL	16	0...3FFH	0...5	0,0049	V
Conversion table for linearization of TOIL value acquisition					

2.23 Acquisition of PVS

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_PVS_1	O/V	0 ... 3FFH	0 ... 4.995117	0.004883	V
Voltage of pedal value sensor 1					
V_PVS_2	O/V	0 ... 3FFH	0 ... 4.995117	0.004883	V
Voltage of pedal value sensor 2					

Input data:

V_PVS_1_BAS	V_PVS_2_BAS		

FUNCTION DESCRIPTION:

General information:

The Signal Acquisition and Conversion is done in the Basic Software.

A Filtering to reduce noise is done in the Application Software, refer to the Subchapter "PVS Variables" in the Chapter "System Variables".

Application conditions:

Activation: allways

Deactivation: -

Initialisation: at reset

V_PVS_1 = V_PVS_1_BAS

V_PVS_2 = V_PVS_2_BAS

Update Rate: 10 ms

Formula section:

V_PVS_1 = V_PVS_1_BAS

V_PVS_2 = V_PVS_2_BAS

2.24 INSY Requirements Infrastructure Interface

Overview

Those actions read out the dedicated digitalized voltages.

The sensor values retrieved by the actions have to be valid at reset.

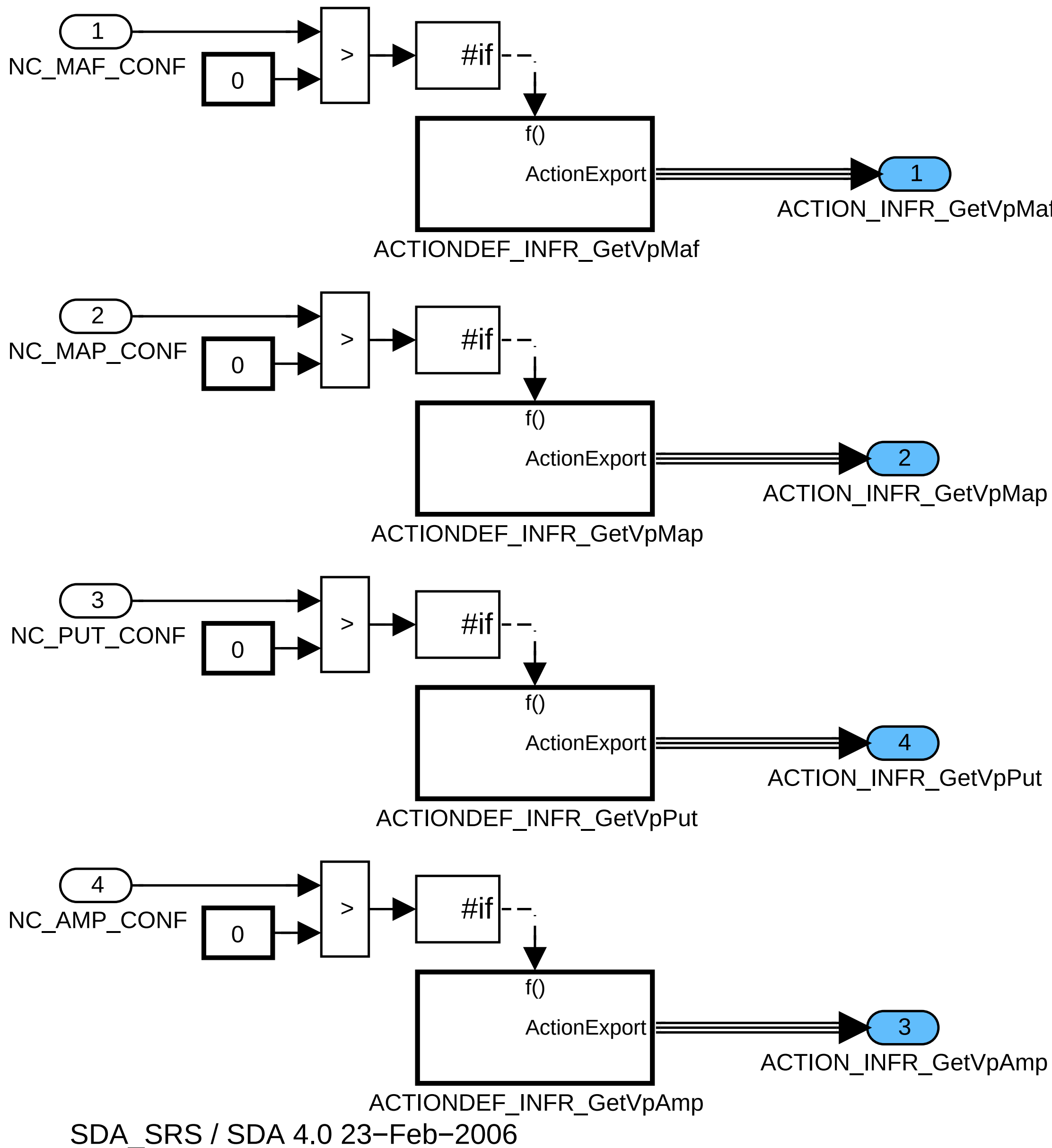


Figure 1 INSY_IFINF0

Input data:

NC_MAF_CONF	NC_MAP_CONF	NC_PUT_CONF	NC_AMP_CONF

Export actions:

ACTION_INFR_GetVpAmp(OUT <VP_AMP>)
This action returns the digitalized voltage of ambient pressure data acquisition.
ACTION_INFR_GetVpMaf(OUT <VP_MAF>)
This action returns the digitalized voltage of mass air flow data acquisition.
ACTION_INFR_GetVpMap(OUT <VP_MAP>)
This action returns the digitalized voltage of manifold air pressure data acquisition.
ACTION_INFR_GetVpPut(OUT <VP_PUT>)
This action returns the digitalized voltage of pressure upstream throttle data acquisition.

Description for Actions

ACTION_INFR_GetVpAmp(OUT <VP_AMP>)

This action returns the digitalized voltage of ambient pressure data acquisition. The digitalization is done autonomous by the Infrastructure.

Parameter	Type	Hex. limits	Phys. limits	Resol.	Unit
VP_AMP	OUT	0...7FFFH	0...4.999847	1.52588 E-4	V
This parameter returns the digitalized voltage					

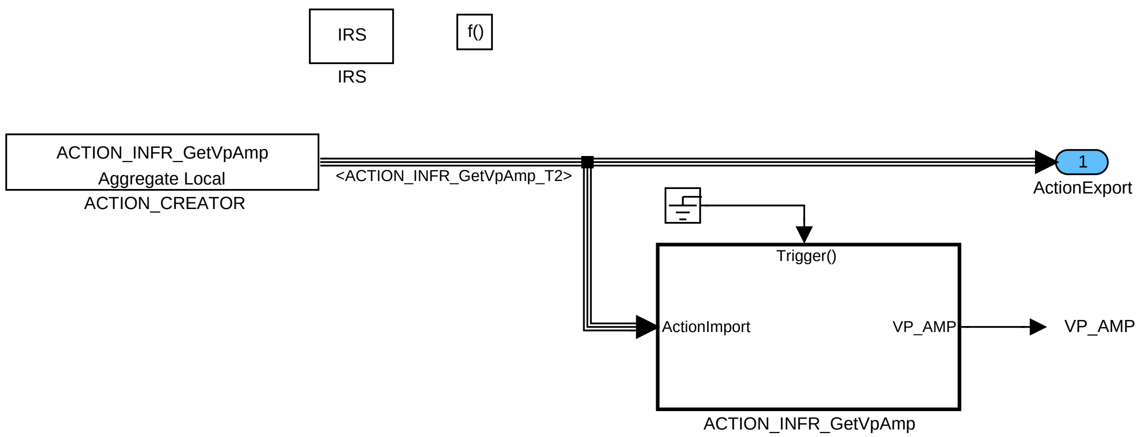


Figure 2 INSY_IFINF0/ ACTIONDEF_INFR_GetVpAmp

ACTION_INFR_GetVpMaf(OUT <VP_MAF>)

This action returns the digitalized voltage of mass air flow data acquisition. The digitalization is done autonomous by the Infrastructure.

Parameter	Type	Hex. limits	Phys. limits	Resol.	Unit
VP_MAF	OUT	0...7FFFH	0...4.999847	1.52588 E-4	V
This parameter returns the digitalized voltage					

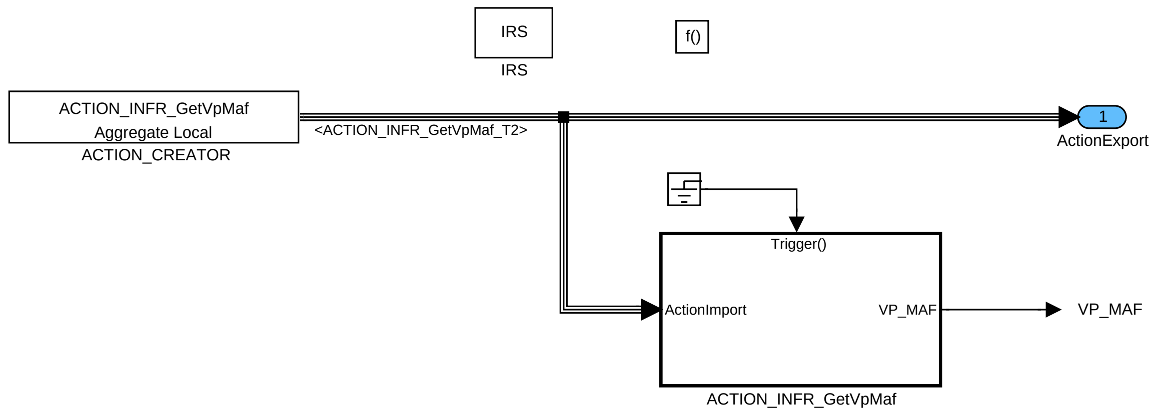


Figure 3 INSY_IFINF0/ ACTIONDEF_INFR_GetVpMaf

ACTION_INFR_GetVpMap(OUT <VP_MAP>)					
This action returns the digitalized voltage of manifold air pressure data acquisition. The digitalization is done autonomous by the Infrastructure.					
Parameter	**Type**	**Hex. limits**	**Phys. limits**	**Resol.**	**Unit**
VP_MAP	OUT	0...7FFFH	0...4.999847	1.52588 E-4	V
This parameter returns the digitalized voltage					

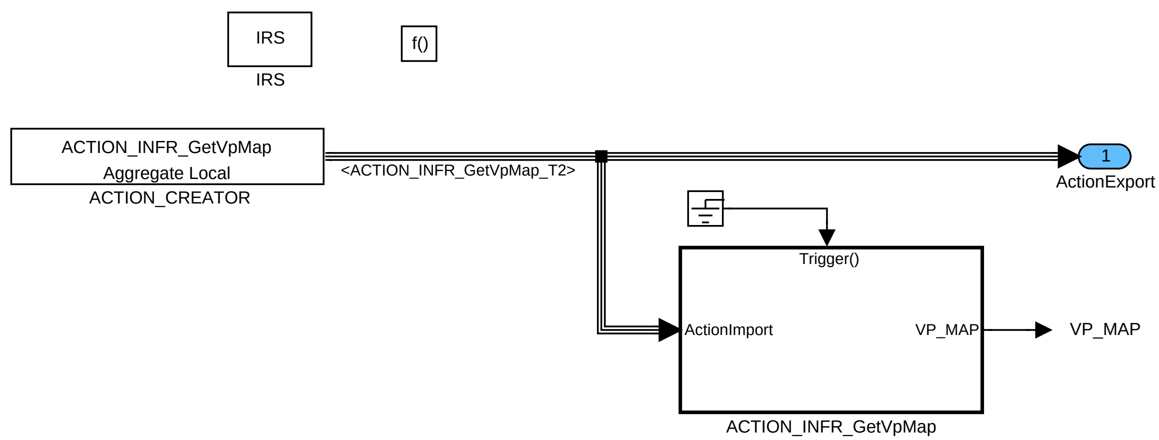


Figure 4 INSY_IFINF0/ ACTIONDEF_INFR_GetVpMap

ACTION_INFR_GetVpPut(OUT <VP_PUT>)					
This action returns the digitalized voltage of pressure upstream throttle data acquisition. The digitalization is done autonomous by the Infrastructure.					
Parameter	**Type**	**Hex. limits**	**Phys. limits**	**Resol.**	**Unit**
VP_PUT	OUT	0...7FFFH	0...4.99984741	1.52588 E-4	V
This parameter returns the digitalized voltage					

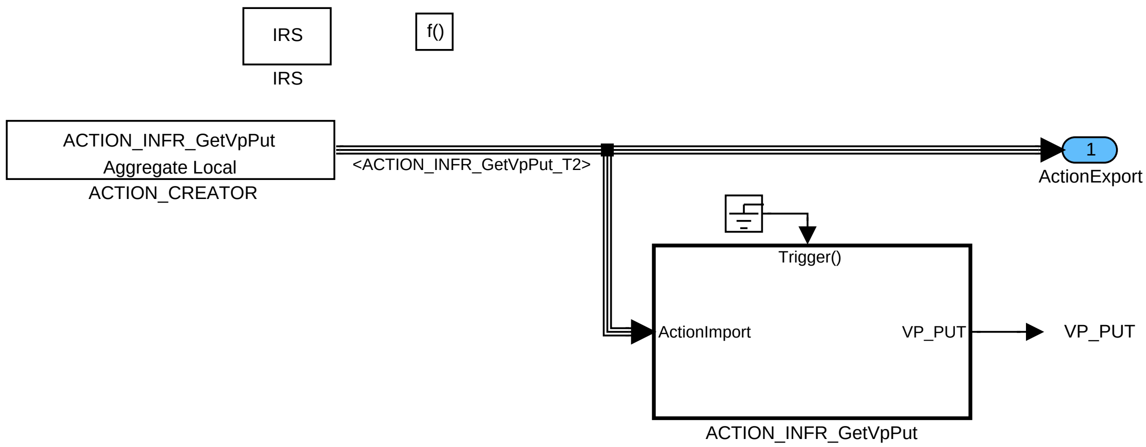


Figure 5 INSY_IFINF0/ ACTIONDEF_INFR_GetVpPut

Function Description

2.24.1 Requirements for ACTION_INFR_GetVpMaf:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
VP_MAF		< 1% of reference supply voltage for sensor	ADC: 10 bit value: 15 bit	Not relevant	VP_MAF is converted into a MAF signal. To receive a certain absolute MAF precision, the given precision of VP_MAF is needed.

Diagnosis: A diagnosis is performed based on the returned signal elsewhere

Coincidence requirements: The sensor value retrieved by the action has to be valid at reset.

When calling the Action, the returned voltage is not older than 1ms.

2.24.2 Requirements for ACTION_INFR_GetVpMap:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
VP_MAP		< 1% of reference supply voltage for sensor	ADC: 10 bit value: 15 bit	Not relevant	VP_MAP is converted into a MAP signal. To receive a certain absolute MAP precision, the given precision of VP_MAP is needed.

Diagnosis: A diagnosis is performed based on the returned signal elsewhere

Coincidence requirements: The sensor value retrieved by the action has to be valid at reset.

When calling the Action, the returned voltage is not older than 1ms.

2.24.3 Requirements for ACTION_INFR_GetVpPut:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
VP_PUT		< 1% of reference supply voltage for sensor	ADC: 10 bit value: 15 bit	Not relevant	VP_PUT is converted into a PUT signal. To receive a certain absolute PUT precision, the given precision of VP_PUT is needed.

Diagnosis: A diagnosis is performed based on the returned signal elsewhere

Coincidence requirements: The sensor value retrieved by the action has to be valid at reset.

When calling the Action, the returned voltage is not older than 1ms.

2.24.4 Requirements for ACTION_INFR_GetVpAmp:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
VP_AMP		< 1% of reference supply voltage for sensor	ADC: 10 bit value: 15 bit	Not relevant	VP_AMP is converted into a AMP signal. To receive a certain absolute AMP precision, the given precision of VP_AMP is needed.

Diagnosis: A diagnosis is performed based on the returned signal elsewhere

Coincidence requirements: The sensor value retrieved by the action has to be valid at reset.

When calling the Action, the returned voltage is not older than 100ms.

2.25 Acquisition of sensor voltages within INSY

Overview

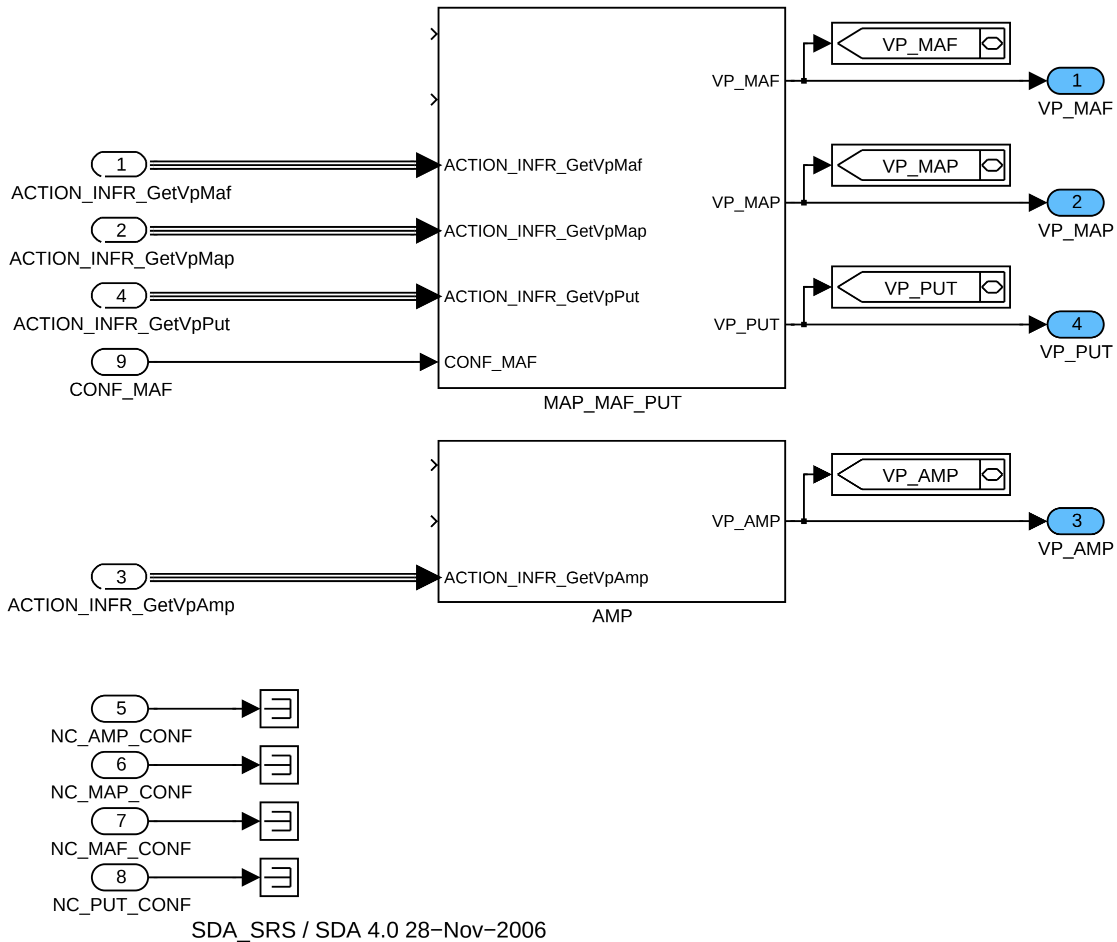


Figure 6 INSY_SIGCVSENS0

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VP_MAF	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
Mass air flow sensor raw acquisition.					
VP_MAP	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
Voltage of the intake manifold pressure sensor (for diagnosis)					
VP_AMP	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
measured sensor voltage at ECU input pin for AMP					
VP_PUT	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
Pressure upstream throttle sensor voltage					

Input data:

NC_AMP_CONF	NC_MAP_CONF	NC_MAF_CONF	NC_PUT_CONF
CONF_MAF			

Import actions:

ACTION_INFR_GetVpAmp(OUT <VP_AMP>)
This action returns the digitalized voltage of ambient pressure data acquisition
ACTION_INFR_GetVpMaf(OUT <VP_MAF>)
This action returns the digitalized voltage of mass air flow data acquisition
ACTION_INFR_GetVpMap(OUT <VP_MAP>)
This action returns the digitalized voltage of manifold air pressure data acquisition
ACTION_INFR_GetVpPut(OUT <VP_PUT>)
This action returns the digitalized voltage of pressure upstream throttle data acquisition

2.25.1 Mass air flow, Manifold air pressure and Pressure upstream throttle raw acquisitions

The purpose of this module is to make a first level of filtering of raw acquisition mass air flow and / or raw acquisition of manifold air pressure and / or raw acquisition of pressure upstream throttle.

Function Description

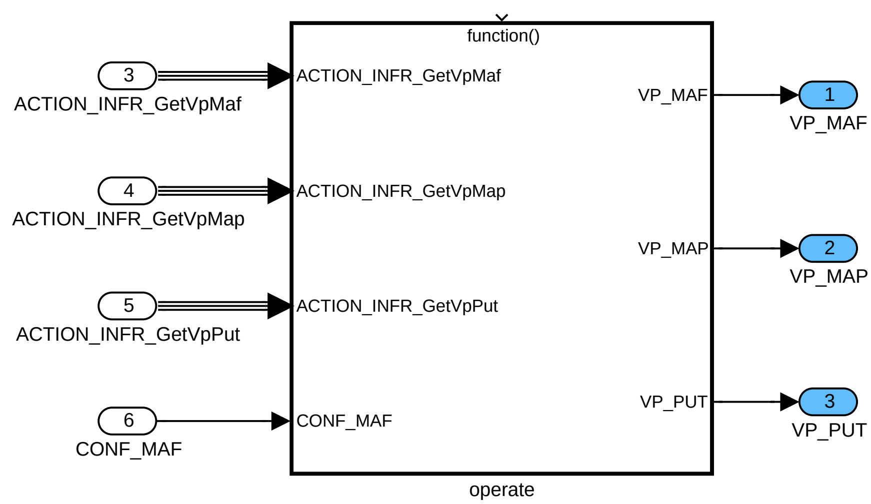


Figure 7 INSY_SIGCVSENS0/ MAP_MAF_PUT

2.25.1.1 Initialization at reset and operate_1ms:

Filtering of raw values is done in infrastructure. The values are returned by the action ACTION_INFR_GetVpMaf, ACTION_INFR_GetVpMap and ACTION_INFR_GetVpPut.

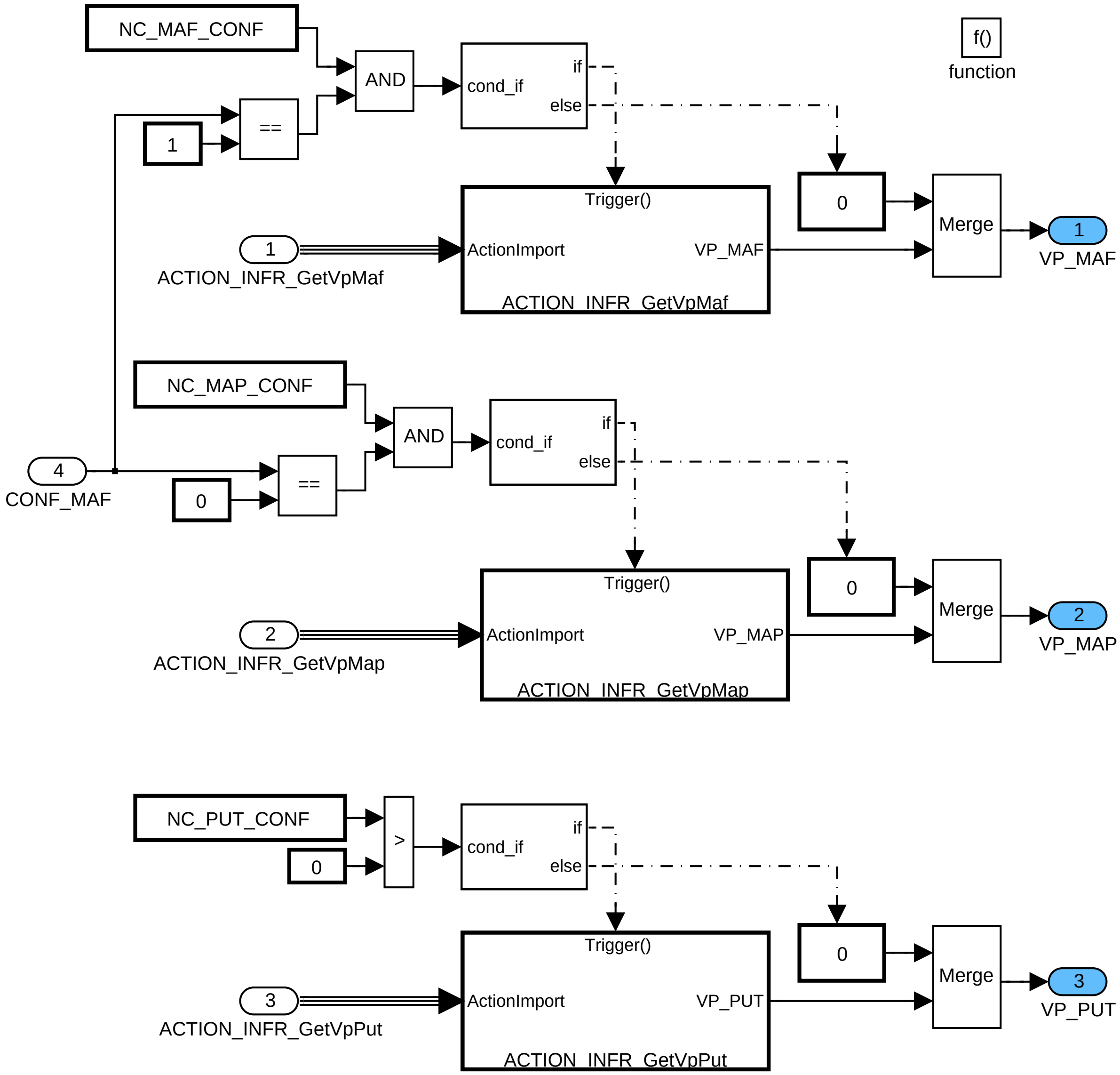


Figure 8 INSY_SIGCVSENS0/ MAP_MAF_PUT/ operate

2.25.2 Ambient pressure raw acquisition

The purpose of this module is to make a first level of filtering of raw acquisition of ambient pressure.

Function Description

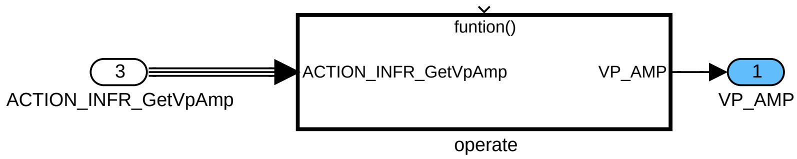


Figure 9 INSY_SIGCVSENS0/ AMP

2.25.2.1 Initialization at reset and operate_100ms:

Filtering of raw values is done in infrastructure. The value is returned by the action ACTION_INFR_GetVpAmp.

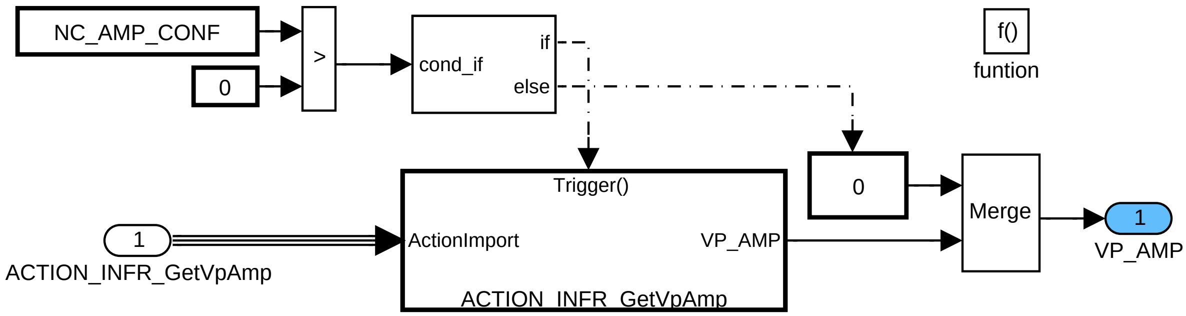


Figure 10 INSY_SIGCVSENS0/ AMP/ operate

2.26 THRO - Requirements for infrastructur (ETC)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_TPS_1_BAS	V/O	0h ... 3FFh	0 ... 4.9951	0.0049	V
Raw value throttle position sensor 1 (not amplified)					
V_TPS_GAIN_1	V/O	0h ... 3FFh	0 ... 4.9951	0.0049	V
Dummy value for throttle position sensor 1 (amplified)					
V_TPS_2_BAS	V/O	0h ... 3FFh	0 ... 4.9951	0.0049	V
Raw value throttle position sensor 2 (not amplified)					

Input data:

MTCPWM	LV_PRDR_ACT		

FUNCTION DESCRIPTION:

General information:

This module describes the interfaces to the basic software (BSW) and the time behaviour between throttle position signal aquisition, digital position controller calculation and the PWM output generation (power stage).

2.26.1 Throttle position signal aquisition

Description:

The two throttle position sensor signals are determined every millisecond. V_TPS_1_BAS and V_TPS_2_BAS are converted in the standard queue.

Name	Aquisition recurrence	Acquisition precision	Notes
V_TPS_1_BAS	1 ms	10 bit	Raw value of TPS 1
V_TPS_2_BAS	1 ms	10 bit	Raw value of TPS 2

Application conditions:

Initialisation: *at reset:*

V_TPS_x_BAS = 0 V

V_TPS_GAIN_1 = 5 V *{ Is intialisied but isn't determined in DC ! }*

Recurrence: 1 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

Raw signal processing: -

2.26.2 PWM output generation

Description:

The power at the H–bridge outputs is managed by a PWM signal and the current direction is adjusted by a direction request (a second power stage input pin) with a update rate of 2 ms.

Name	Frequency (Hz)	Updating of duty cycle	Range of duty cycle	Init.	Notes
MTCPWM	1000	2 ms	-100 % ... 99.997 %	0 %	Output position controller

Application conditions:

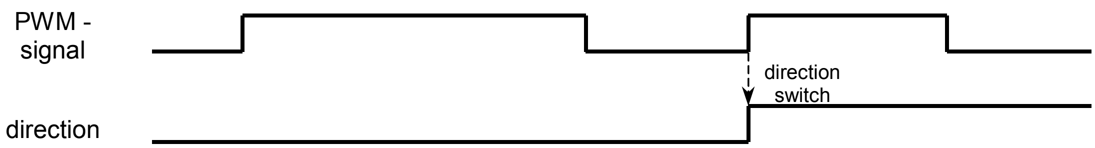
Initialisation: *at reset inactive, currentless*

Recurrence: 2 ms

Activation: at every engine operating state

Deactivation: -

Note: The direction of the H-bridge can be inverted only synchronised to the PWM:



2.26.3 Time behaviour

General information:

This chapter describes the time behaviour between throttle position signal aquisition, digital position controller calculation and the PWM output generation (power stage).
The aim is to synchronise the TPS aquisition with the controller calculation and the PWM output generation for minimal time delay.

Description:

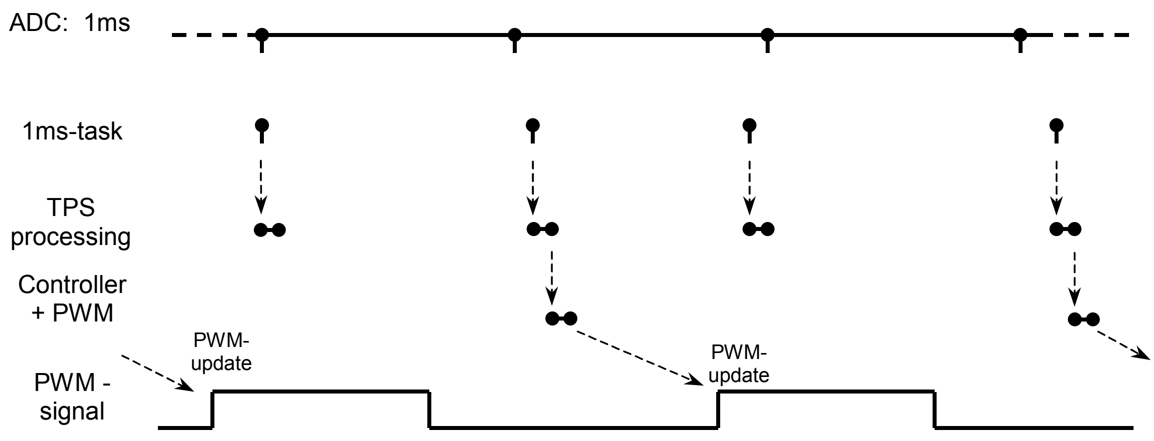
The TPS signal determination and the digital position controller are calculated in the 1 ms ASW task. The AD conversion process (after every ms) and the PWM generation are executed asynchronously to the 1 ms ASW task. The maximum time delay between AD conversion and PWM generation depend on time jitter between:

- AD conversion and TPS signal determination ($0 \leq T_1 \leq 1$ ms)
- Position controller calculation and PWM update ($0 \leq T_2 \leq 1$ / frequency)

The maximum time delay is the summation of T_1 and T_2 !

($0 \leq T_{delay} \leq 3$ ms for a frequency of 500 Hz, $0 \leq T_{delay} \leq 2$ ms for a frequency of 1000 Hz)

Signal flow diagram:



2.26.4 Power stage activation

General information:

The ETC power stage is equipped with two disable lines to switch-on or off. Disable line one is connected with the main controller and disable line two with the monitoring unit, both processor units are able to switch-off the power stage. (To the activation of the power stage both disable lines have to be enable.)

Description:

The BSW provides a rountine to switch on or off the power stage via disable line.

Application conditions:

Initialisation: *at reset:*

{ Consider requirements of Monitoring Concept (Level 3 – PREDRIVE check) }

if LV_PRDR_ACT == 0

then *{ disable power stages }*

CALL BSW routine to disable H-bridge via disable line

else *{ enable power stages }*

CALL BSW routine to enable H-bridge via disable line

endif

Recurrence: -

Activation: *at every engine operating state*

Deactivation: -

2.26.5 Power stage error detection

General information:

The power stage (H – bridge IC) for electronical throttle control is equipped with internal diagnosis functions. The following errors can be indicated but can not be distinguished by the software per power stage status flag line.

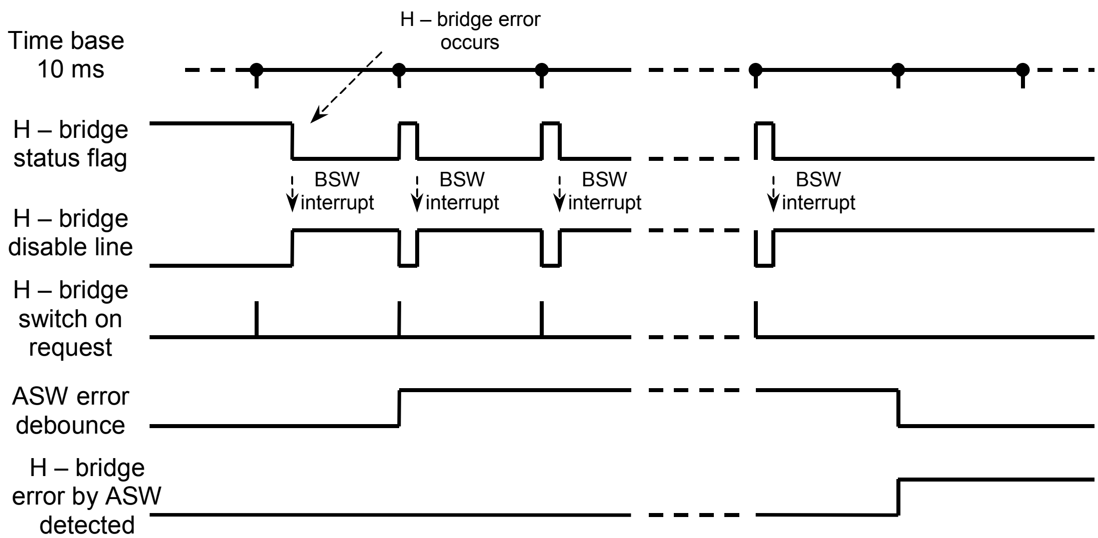
- Undervoltage / Overcurrent / Overtemperature / H-bridge hardware disable

Description:

The following error detection are done by BSW and ASW if an error occurs. In the case of errors, the status flag line of the power stage changes his logical level and triggers an BSW interrupt. The interrupt set a mark for the ASW “ power stage diagnosis “ and switch-off the power stage. The ASW reads and resets the mark, starts the error debounce and switch-on the power stage. The error detection is started newly.

After the error debounce, an error is set and the power stage is switched off by the ASW. The respective ASW functions are executed in a time base of 10 ms.

Signal flow diagram:



2.27 ETC Monitoring ECM2 - Requirements to BSW

2.27.1 Requirements concerning ‘Monitoring of engine speed using a software timer’

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_SEG_SW	O	0 ... FFFFFFFFH	0 ... 2.6844e8	1	µs
segment time determined by means of a software timer					

FUNCTION DESCRIPTION:

General information:

The segment time T_SEG_SW is generated by a software timer solution in the BSW. The software timer is triggered by crankshaft segments and produces time stamps for the trigger events. The maximum delay between event and time stamp must be less than 300 µs (split into an invariable part of 50 µs and a variable part of 250µs). From the time stamps the signal T_SEG_SW is determined; its resolution is 1 µs.

The segment time T_SEG_SW is necessary to generate an engine speed signal that is independent from N_32 (generated from T_SEG). It is used in the module ‚Monitoring of engine speed using a software timer‘ of the aggregate ECM2.

Note: Since for the event <Engine Stop> the engine speed N_32 is set to zero in the function level (and T_SEG is set to its maximum value, respectively), the segment time T_SEG_SW has to be set to its maximum value for <Engine Stop> as well. However, this action must not be triggered using the variable LV_ES=0, but must be initialized by the BSW trigger that is set when the BSW is recognizing the event <Engine Stop>.

Application conditions:

Initialisation: max. value

at 1. initialization after <Power On> and

at transition BSW event <Engine Running> ⇒ <Engine Stop>

Recurrence: segment - synchronous

Activation: during every engine state

Deactivation: -

2.27.2 Requirements for pedal value signals and throttle position sensor signals

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_PVS_1_BAS	O	0...3FFH	0 ... 4.995117	0.004888	V
Voltage PVS channel 1					
V_PVS_2_BAS	O	0...3FFH	0 ... 4.995117	0.004888	V
Voltage PVS channel 2					
V_TPS_1_BAS	O	0...3FFH	0 ... 4.995117	0.004888	V
Voltage TPS channel 1					
V_TPS_2_BAS	O	0...3FFH	0 ... 4.995117	0.004888	V
Voltage TPS channel 2					

FUNCTION DESCRIPTION:

General information:

The pedal value signals, channel 1 and 2, as well as the throttle position signals, channel 1 and 2, are monitored in the *Process Monitoring*, see "Monitoring of pedal value signals" and "Monitoring or mass air flow signals". The according modules are running with a recurrence of 40ms, see "Process Monitoring". To define the scope of the *Process Monitoring* as wide as possible, the voltages should be a direct output of the ADCs.

Note: Here only the requirements regarding the *Process Monitoring* are given. No functional aspects of the functionalities primarily processing these signals are listed here. Therefore especially the recurrence may occure with a higher frequency, but the range and resolution have to be the same.

Application conditions:

Recurrence: 40ms

Activation: for conditions see 'Application Incidences and configuration of Process Monitoring'

2.28 Exhaust gas composition – Requirements to I/O BSW

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_UP_RAW[NC_CBK_EX_NR]	O/V	0...3FFH	0...4.995	0.0049	Volt
Raw value of upstream m oxygen sensor voltage					
VLS_DOWN[NC_CBK_EX_NR]	O/V	0...3FFH	0...4.995	0.0049	Volt
Downstream oxygen sensor voltage					
LV_LSH_SCP_LSH_UP[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					
LV_LSH_SCG_LSH_UP[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					
LV_LSH_OC_LSH_UP[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					
LV_LSH_SCP_LSH_DOWN[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					
LV_LSH_SCG_LSH_DOWN[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					
LV_LSH_OC_LSH_DOWN[NC_CBK_EX_NR]	O	0...1H	0...1	1	-
Logical value for Hardware-based LSH error detection					

Input data:

NC_CBK_EX_NR	LV_R_IT_REQ_LS_UP[NC_CBK_EX_NR]	LV_R_IT_REQ_LS_DOWN[NC_CBK_EX_NR]	NC_STATE_VLS_UP_SIG_ACQ
NC_STATE_LSL_UP_IF			

FUNCTION DESCRIPTION:

General information:

Name	Acquisition recurrence	Acquisition precision	Notes
VLS_UP_RAW[i]	10ms	10bit	NC_STATE_VLS_UP_SIG_ACQ = 0
	1ms	10bit	NC_STATE_VLS_UP_SIG_ACQ = 1
VLS_DOWN[i]	10ms	10bit	-

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank.

If the lambda sensor signal shall be filtered prior to further processing NC_STATE_VLS_UP_SIG_ACQ has to be set to 1, otherwise to 0.

Description:

Voltages of lambda sensors offered by Basic SW are transformed to application system specific variables.

The error detection is effected via the ECU hardware.
Error informations are gathered (or-ed symptom specific) until read out by the application software ASW.

At the moment when ASW is reading data from I/O software, the I/O software data will be reset. (It is possibel to have more than one electrical failure symtom reported to ASW.)

The variables LV_LSH_SCP/SCG/OC_LSH_UP/DOWN[i] are set to one (valid) if new diagnosis data from hardware are available.

If more than one error symptom is detected during the readout period, the priority of the errors reported is:

4. short cut to battery (scp)
5. short cut to ground (scg)
6. open curcuit (ol)

⇒ SCP will shut off the device and the error will be kept by the device until a "off-slope" occurs.

⇒ SCG, OL: read out of the device erases error status and the device is not shut off.

Application conditions:

Activation:	at every engine operating state	
Deactivation:	-	
Update rate:		
VLS_UP_RAW[i]:	10 ms for NC_STATE_VLS_UP_SIG_ACQ = 0	
	1 ms for NC_STATE_VLS_UP_SIG_ACQ = 1	
VLS_DOWN[i] :	10 ms	
Initialization:	at reset:	VLS_UP_RAW[i] = 0
		VLS_DOWN[i]= 0

Formula section:

 # If NC_STATE_LSL_UP_IF = 0 *% Masking shall only be done for binary sensor*
 # then **If** LV_R_IT_REQ_LS_UP[i] = 0
then *acquire* VLS_UP_RAW[i]
else $VLS\_UP\_RAW[i]_N = VLS\_UP\_RAW[i]_{N-1}$
endif
 # else *acquire* VLS_UP_RAW[i]
 # endif

If LV_R_IT_REQ_LS_DOWN[i] = 0

then *acquire* VLS_DOWN[i]

else $VLS\_DOWN[i]_{N} = VLS\_DOWN[i]_{N-1}$

endif

2.29 ENTE – Requirements to infrastructure interface

Input data:

NC_NR_TCO_SENS			

Export actions:

ACTION_INFR_GetVpTco (OUT <>)
This action provides the digitized voltage value of the available coolant temperature sensors in an array.

Description for actions:

ACTION_INFR_GetVpTco (OUT <>)					
This action provides the digitized voltage value of the available coolant temperature sensors in an array delivered from the infrastructure. The AD conversion is performed autonomously by the infrastructure. When the action is called the gathered information will be provided to the application software level. The dimension of the parameter *Vp_tco_sens* is given by configuration input NC_NR_TCO_SENS. The index range is 0 … NC_NR_TCO_SENS-1.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Vp_tco_sens [NC_NR_TCO_SENS]	OUT	0...7FFFH	0...4.999847412	1.53E-04	V
Coolant temperature sensor voltage value					

FUNCTION DESCRIPTION:

General information:

The following action is used for the acquisition of the digitized voltage value of the adressed coolant temperature sensor in an array. The action delivers the coolant temperature sensor voltage value to the application software level from the standard AD converter queue.

- NC_NR_TCO_SENS is valid from 1...255.

  The voltage belonging to the first sensor is returned in Vp_tco_sens[0].

  The voltage belonging to the *n* sensor is returned in Vp_tco_sens[n-1].
- The AD conversion is performed autonomously by the infrastructure, the returned value is not older than 10 ms.
- The voltage value is gathered in the infrastructure until the application reads out the information by calling the action, old values are replaced by new values.

Requirements for ACTION_INFR_GetVpTco:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Vp_tco_sens [NC_NR_TCO_SENS]			10 Bit	The following coinicidence requirement has to be fulfilled in case of NC_NR_TCO_SENS > 1: Vp_tco_sens[k] is digitized at point in time t_k. For the returned array the maximum allowed time periode $\| t_k - t_i \|$ must less than 10ms.	No comment

Diagnosis: no electrical diagnosis done here

Coincidence requirements: When calling this action, all returned voltages must be within the last 100 ms.

2.30 ENSD - Requirements to infrastructure interface

Export actions:

ACTION_ENSD_GetDigCAMINLevel (OUT <STATE_CAM_IN>)
This action reads whether the intake camshaft signal is high or low
ACTION_ENSD_GetDigCAMEXLevel (OUT <STATE_CAM_EX>)
This action reads whether the exhaust camshaft signal is high or low

FUNCTION DESCRIPTION:

General information:

The minimum and maximum engine positions are determined according to the actual possible camshaft signal edge indexes found by the camshaft signal acquisition for crankshaft synchronisation. In the presence of both intake and exhaust camshaft signals the engine position is known with better precision. This is done by initialising the minimum and maximum engine positions based on the combinations of the intake and exhaust camshaft signals. And for this the state of the camshaft signals (HIGH or LOW) are required.

Activation: CONF_CAM_VVT_EX = 1

Deactivation: LV_CRK_SYN = 1

Description for actions:

ACTION_ENSD_GetDigCAMINLevel (OUT<STATE_CAM_IN>)					
This action reads whether the intake camshaft signal is high or low					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
STATE_CAM_IN	OUT	0H 1H	LOW HIGH	1	-
State variable indicating intake camshaft signal is high or low					
ACTION_ENSD_GetDigCAMEXLevel (OUT <STATE_CAM_EX>)					
This action reads whether the exhaust camshaft signal is high or low					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
STATE_CAM_EX	OUT	0H 1H	LOW HIGH	1	-
State variable indicating exhaust camshaft signal is high or low					

2.30.1 Detailed description for Action: ACTION_ENSD_GetDigCAMINLevel

Requirements for ACTION_ENSD_GetDigCAMINLevel:

Data acquisition					
Parameter	Absolute precision	Relaitve precision	Resolution	Coincidence with parameter	Comment
STATE_CAM_IN	Not relevant	Not relevant	1	Not relevant	Not relevant

Diagnosis: no electrical diagnosis done here
Coincidence requirements: Every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

2.30.2 Detailed description for Action: ACTION_ENSD_GetDigCAMEXLevel

Requirements for ACTION_ENSD_GetDigCAMEXLevel:

Data acquisition					
Parameter	Absolute precision	Relaitve precision	Resolution	Coincidence with parameter	Comment
STATE_CAM_EX	Not relevant	Not relevant	1	Not relevant	Not relevant

Diagnosis: no electrical diagnosis done here
Coincidence requirements: Every camshaft signal edge (rising and falling) and every reference gap if NC_ACT_CAM_EDGE_SYN = 3

2.31 ENSS - Requirements to infrastructure interface

Output data:

Input data:

Export actions:

ACTION_INFR_EONVStartTimer ( )
Start the timer of the PIC
ACTION_INFR_EONVStopTimer ( )
Stop the timer of the PIC
ACTION_INFR_EONVResetTimer ( )
Reset the basic timer of the PIC
ACTION_INFR_EONVWarmReset ( )
Reset PIC micro controller
ACTION_INFR_EONVCompleteReset( )
Reset PIC micro controller and clear all data
ACTION_INFR_EONVReadTimer (IN<Lv_cmd>**, OUT<**T_es> **)**
Engine Off timer is read in from PIC

FUNCTION DESCRIPTION:

General information:

The Engine off timer function is deployed to two separate processors – the main microcontroller and an external 8-bit microcontroller. The two controllers are linked via a dedicated SPI (serial peripheral interface) bus.

Within the main microcontroller, the BIOS driver EMUSPI (emulated SPI) is used to control communication with the 8-bit microcontroller. Consult the specification "BIOS SW Specification EMUSPI" (bemuspi.doc), which is maintained by the software team, for more information. The application software interfaces with the IO software through the TRL (transformation layer) via the TRL EMUSPI interface. Consult the specification "TRL SW Specification EMUSPI" (temuspi.doc), which is also maintained by the software team, for more information.

Description:

The raw engine off duration, t_es, is produced from a basic timer that is capable of running while the main CPU is powered-off. The basic timer is deployed on the external 8-bit microcontroller and can be started, stopped, and reset upon request from the application.

Application conditions:

Recurrence: Upon request.

Activation: Upon request.

Deactivation: Upon request

Description for actions:

The general mechanism for reading values is to invoke an interface to read the values from the external microcontroller into the EMUSPI shift register, and then invoke another interface to read the values from the EMUSPI shift register into the variable. These interfaces are called upon request from the application software; however, the timing constraints given in the specifications (bemuspi.doc and temuspi.doc) must be respected.

--- Interfaces involved in getting data ---
Note: * Interface need to be called two times consecutively with Lv_cmd = 0 and Lv_cmd =1, respectively. Minimum recurrence of these consecutive calls is 5ms. Any other interfaces must not be called between these two calls.

ACTION_INFR_EONVStartTimer ( )
This action start the basic timer inside PIC

ACTION_INFR_EONVStopTimer ( )
This action stop the basic timer inside PIC

ACTION_INFR_EONVResetTimer ( )
Reset the basic timer

ACTION_INFR_EONVWarmReset ( )
Reset PIC controller

ACTION_INFR_EONVCompleteReset( )
Reset the PIC controller and all data are clear

***ACTION_INFR_EONVReadTimer (** Lv_cmd**,** T_es**)**					
The engine of timer is read in over the PIC controller					
Name	Type	Hex. Limits	Phys. Limits	Resol.	Unit
Lv_cmd	IN	0..1H	0..1	1	-
T_es	OUT	0...FFFFFFH	0...16777216	1	sec

Information for ACTION_INFR_EONV_ReadTimer:

T_es = Data shifted right by 8

Diagnosis: no diagnosis

2.32 AIRT Temperature Sensor(s): Infrastructure Requirement

Overview

FUNCTION DESCRIPTION:

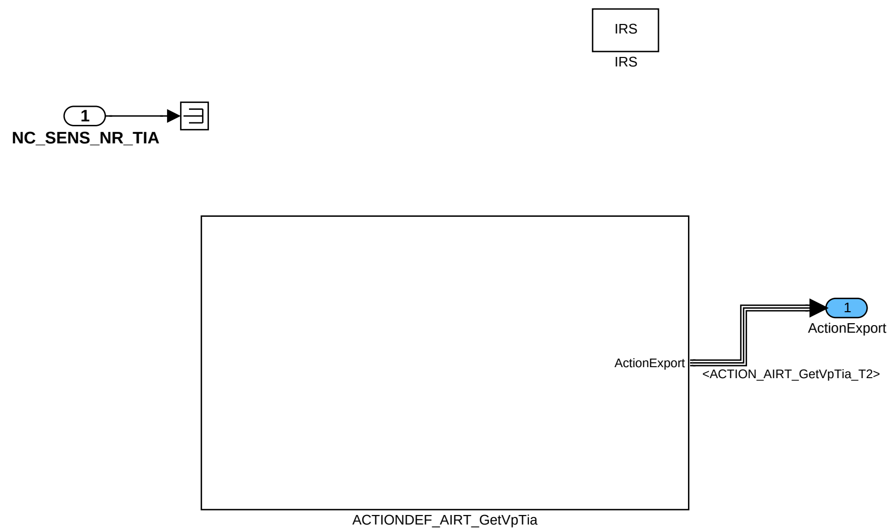
The sensor value retrieved by the action has to be valid at reset

This function is applicable with a NTC type sensor.

The data of the ACTION call have the following meaning.

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
CtrTiaSens	**1**			**VpTia is the digitalised voltage of the sensor with the index CtrTiaSens**	
VpTia	**Not relevant**	**< 1% of reference supply voltage for sensor**	**ADC: 10bit** **Value: 15bit**	**Recurrence 100ms**	

(*) Depending on the Project, one or several TIA sensors can be used. See Specification "**Air Temperature Sensor(s) Configuration - 30404X01"** where the Sensors Configuration is defined from NC_TIA_CONF value.



SDA_SRS / SDA 3.1 31−Mar−2004

Figure 11 AIRT_IFINF0

Input data:

NC_SENS_NR_TIA

Export actions:

ACTION_AIRT_GetVpTia(IN <CtrTiaSens>, OUT <VpTia>)
Action to read the sensor voltage of the different intake air temperature sensors (NC_SENS_NR_TIA sensors)

Description for Actions

ACTION_AIRT_GetVpTia(IN <CtrTiaSens>, OUT <VpTia>)

The input <CtrTiaSens> gives the basic software the information which of the NC_SENS_NR_TIA sensor has to be provided to the application software.
Action to read the sensor voltage of intake air temperature sensors (NC_SENS_NR_TIA sensors)

Parameter	Type	Hex. limits	Phys. limits	Resol.	Unit
CtrTiaSens	IN	0...3H	1...4	1	-
Array index of TIA sensor					
VpTia	OUT	0...7FFFH	0...4.999847	1.52588 E-4	V
voltage of the intake air temperature sensor					

2.33 CHRG - Requirements to Infrastructure Interface

2.33.1 Analog inputs from hardware to ECU

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VP_N_TCHA	V/O	0...7FFFH	0... 4.999847	1.5259E-4	V
Turbo charger speed sensor raw signal					

FUNCTION DESCRIPTION:

General information:

Name	Acquisition recurrence	Acquisition precision	Initialization at ECU reset	Notes (Initialization with values which stand for a critical engine operation)
VP_N_TCHA	10 ms	10 bit	4.999847	

Diagnosis capabilities:

Electrical diagnosis will be done by using converted values within the Application.

Consistency requirements regarding AD convertion:

2.33.2 Get EL_RCL_ACR electrical diagonis result - Requirements to Infrastructure

Input data:

Export actions:

ACTION_INFR_GetElDiagEl_rcl_acr(OUT <Cdn_diag>, OUT<Err_diag>**)**
This action reads the failure and condition information for a symptom of the RCL_ACR power stage

Description for actions:

ACTION_INFR_GetElDiagEl_rcl_acr(OUT <Cdn_diag>, OUT<Err_diag>**)** This action reads the failure and condition information for a symptom of the RCL_ACR power stage. The readout of the power stage is performed autonomous and the information is gathered. When calling this Action the information inside the Infrastructure is reset after returning the OUT parameters.					
Name	Type	Hex. Limits	Phys. Limits	Resol.	Unit

Cdn_diag	OUT	0 ... 7H	0 ... 7	1	-
Diagnosis condition for symptom: Bit 0: Set, if diagnosis condition for symptom SCP (SYM_0) is fulfilled Bit 1: Set, if diagnosis condition for symptom SCG (SYM_1) is fulfilled Bit 2: Set, if diagnosis condition for symptom OC (SYM_2) is fulfilled					
Err_diag	OUT	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Raw value of error symptom. The relevant bit is set, if the error has beed detected.					

FUNCTION DESCRIPTION:

General information:

This Action returns the result of the electrical diagnosis of the RCL_ACR power stage.

_ The device readout is performed autonomous by the Infrastructure each 10ms.
_ The error informations are gathered in the Infrastructure (or-ed symptom, cylinder specific) until the Application reads out the information by calling ACTION_INFR_GetElDiagEl_rcl_acr().
_ After having read out the information by calling ACTION_INFR_GetElDiagEl_rcl_acr(), the data inside the Infrastructure are reset. Resetting of *Cdn_diag* avoids unambigious results in case of too short calling reccurrence of ACTION_INFR_GetElDiagEl_rcl_acr(): Reset *Cdn_diag* indicates, that the gathering of the information is not completely finished.

Requirements for ACTION_AGGR_Command:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Cdn_diag	-	-	<bit coded>	Err_diag	Diagnosis condition for each symptom **bit 0:** diagnosis condition for symptom SCP (SYM_0) **bit 1:** diagnosis condition for symptom SCG (SYM_1) **bit 2:** diagnosis condition for symptom OC (SYM_2) The relevant bit is set, if the condition for a valid diagnosis is fulfilled
Err_diag	-	-	<bit coded>	Cdn_diag	Bitcoded result of each symptom (SYM_0=SCP, SYM_1=SCG, SYM_2=OC) **0h** = NO_SYM **1h** = SYM_0 **2h** = SYM_1 **4h** = SYM_2 The relevant bit is set, if the error has beed detected.

Diagnosis: ACTION_INFR_GetElDiagEl_rcl_acr returns the electric diagnosis

Coincidence requirements: no coincidence requirements to other events

2.33.3 Get WG electrical diagonis result - Requirements to Infrastructure

Input data:

Export actions:

ACTION_INFR_GetElDiagWg(OUT <Cdn_diag>, OUT<Err_diag>**)**
This action reads the failure and condition information for a symptom of the WG power stage

Description for actions:

ACTION_INFR_GetElDiagWg(OUT <Cdn_diag>, OUT<Err_diag>**)**					
This action reads the failure and condition information for a symptom of the WG power stage. The readout of the power stage is performed autonomous and the information is gathered. When calling this Action the information inside the Infrastructure is reset after returning the OUT parameters.					
Name	Type	Hex. Limits	Phys. Limits	Resol.	Unit
Cdn_diag	OUT	0 ... 7H	0 ... 7	1	-
Diagnosis condition for symptom: Bit 0: Set, if diagnosis condition for symptom SCP (SYM_0) is fulfilled Bit 1: Set, if diagnosis condition for symptom SCG (SYM_1) is fulfilled Bit 2: Set, if diagnosis condition for symptom OC (SYM_2) is fulfilled					
Err_diag	OUT	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Raw value of error symptom. The relevant bit is set, if the error has beed detected.					

FUNCTION DESCRIPTION:

General information:

This Action returns the result of the electrical diagnosis of the wastegate (Wg) power stage.

_ The device readout is performed autonomous by the Infrastructure each 10ms.
_ The error informations are gathered in the Infrastructure (or-ed symptom, cylinder specific) until the Application reads out the information by calling ACTION_INFR_GetElDiagWg().
_ After having read out the information by calling ACTION_INFR_GetElDiagWg(), the data inside the Infrastructure are reset. Resetting of *Cdn_diag* avoids unambigious results in case of too short calling reccurrence of ACTION_INFR_GetElDiagWg(): Reset *Cdn_diag* indicates, that the gathering of the information is not completely finished.

Requirements for ACTION_AGGR_Command:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**

Cdn_diag	-	-	<bit coded>	Err_diag	Diagnosis condition for each symptom **bit 0:** diagnosis condition for symptom SCP (SYM_0) **bit 1:** diagnosis condition for symptom SCG (SYM_1) **bit 2:** diagnosis condition for symptom OC (SYM_2) The relevant bit is set, if the condition for a valid diagnosis is fulfilled
Err_diag	-	-	<bit coded>	Cdn_diag	Bitcoded result of each symptom (SYM_0=SCP, SYM_1=SCG, SYM_2=OC) **0h** = NO_SYM **1h** = SYM_0 **2h** = SYM_1 **4h** = SYM_2 The relevant bit is set, if the error has beed detected.

Diagnosis: ACTION_INFR_GetElDiagWg returns the electric diagnosis

Coincidence requirements: no coincidence requirements to other events

2.33.4 Digital Outputs from ECU to hardware

Output data:

Name	Frequency	Notes
#if NC_PSN_RCL_CTL = 1 **#then**		
LV_CMD_RCL_OPEN	-	Control signal for the digital recirculation actuator 1 = open, 0 = close
#end		

Input data:

LV_CMD_RCL_OPEN	NC_PSN_RCL_CTL		

2.33.5 Pulse width modulation Outputs from ECU to hardware

Output data:

PWM output name	Frequency	Updating of duty cycle	Range of duty cycle
#if NC_PSN_RCL_CTL = 2 **#then**			
PWM_RCL	30 Hz	10 ms	0 -100%
#end			
PWM_WG	30 Hz	10 ms	0 -100%
after ECU-reset PWM_WG-frequency is switched to C_PWM_WG_FRQ (see 30404A01.xxx)			

Input data:

PWM_RCL	PWM_WG	NC_PSN_RCL_CTL	

2.34 IGRE – Requirements to infrastucture interface

Input data:

NC_MPL_T_MAX	NC_MPL_IGN_CRK_MAX	NC_INI_CTR_DEAC	NC_IGBT_CUT_OFF_T

Detailed input data definition:

Name	Hex. limits	Phys. limits	Resol.	Unit
NC_MPL_T_MAX	0...FFFFH	0...262.14	0.004	ms
Maximum time duration of multiple spark (normal value 60ms)				
NC_MPL_IGN_CRK_MAX	0...FFH	0...95.625	0.375	°CRK
Maximum ignition angle after TDC to start TD_MPL (normal value 12 to 18°CRK)				
NC_INI_CTR_DEAC	0...FFFFH	0...262.14	0.004ms	ms
Time delay after IGBT response before switching OFF the IGBT – typical value 600µs				
NC_IGBT_CUT_OFF_T	0...FFFFH	0...262.14	0.004ms	ms
Inhibition of "IGBT protection cut off function"				

Export actions:

ACTION_INFR_SetIgnEnable(IN <Cyl>, IN <Inh_igc>)
Enables (inh_igc=1) or disables (inh_igc=0) the ignition for cylinder Cyl.
ACTION_INFR_SetIgnAngle(IN <Cyl>, IN <Iga_igc>)
Sets the ignition angle to apply for cylinder Cyl
ACTION_INFR_SetIgnDwell(IN <Cyl>, IN<Td_fac_min>, IN <Td_igc>, IN <Td_fac_max>)
Sets the dwell time duration and the factors for maximal and minimum dwell time for cylinder Cyl
ACTION_INFR_SetIgcDwellTest(IN <Cyl>, IN <Td_igc>)
Sets the dwell time duration for cylinder Cyl and activate coil charging accordingly
ACTION_INFR_SetIgnMplDwell(IN <Td_mpl>)
Sets dwell time for multiple spark
ACTION_INFR_SetIgnMplDly(IN <Td_mpl_dly>)
Sets the dwell time interruption in multiple sparks
ACTION_INFR_SetIgnMplNr(IN <Ign_mpl_nr>)
Sets the number of ignition sparks in multiple spark
ACTION_INFR_SetIgnCtl(IN <Ign_ctl_mod>)
Sets the ignition in full static or half static mode
ACTION_INFR_GetIgnScpDiagAtm46Atic71(IN <Cyl>, OUT<Diag_scp>)
Gets the diagnosis output for SCP detection
ACTION_INFR_GetIgnScgDiagAtm46Atic71(IN <Cyl>, IN <Min_burn_time>, OUT <V_dur_igc>, OUT<Diag_scg>)
Gets the diagnosis output for SCG or OL detection and the burn time duration

Description for actions:

ACTION_INFR_ SetIgnEnable(IN <Cyl>, IN <Inh_igc>)					
Enables (inh_igc=1) or disables (inh_igc=0) the ignition for cylinder Cyl.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number affected by the inhibition of ignition system					
Inh_igc	IN	0 ... 1	0 ... 1	1	1
Flag indicating inhibition of ignition system					

ACTION_INFR_SetIgnAngle(IN <Cyl>, IN <Iga_igc>)					
Sets the ignition angle to apply for cylinder Cyl					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number on which the ignition angle is applied					
Iga_igc	IN	0 ... FF	-35.625 ... 60	0.375	°CRK
Ignition angle to be applied on cylinder Cyl					

ACTION_INFR_SetIgnDwell(IN <Cyl>, In <Td_fac_min>, IN <Td_igc>, IN <Td_fac_max>)					
Sets the dwell time duration and the factors for maximal and minimum dwell time					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number on which the dwell time is set					
Td_fac_min	IN	1 ... FF	1/128 ... 255/128	1/128	-
Factor to calculate minimum dwell time					
Td_igc	IN	0 ... FFFF	0 ... 262.14	0.004	ms
Dwell time duration					
Td_fac_max	IN	1 ... FF	1/128 ... 255/128	1/128	-
Factor to calculate maximum dwell time					

ACTION_INFR_SetIgcDwellTest(IN <Cyl>, IN <Td_igc>)					
Sets the dwell time duration for cylinder Cyl and activate coil charging accordingly					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number on which the dwell time is set					
Td_igc	IN	0 ... FFFF	0 ... 262.14	0.004	ms
Dwell time duration					

ACTION_INFR_SetIgnMplDwell (IN <Td_mpl>)					
Sets dwell time for multiple spark					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Td_mpl	IN	0 ... FFFF	0 ... 262.14	0.004	ms
Dwell time for multiple spark					

ACTION_INFR_SetIgnMplDly (IN <Td_mpl_dly>)					
Sets the dwell time interruption in multiple sparks					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Td_mpl_dly	IN	0 ... FFFF	0 ... 262.14	0.004	ms
Dwell time interruption					

ACTION_INFR_SetIgnMplNr(IN <Ign_mpl_nr>)					
Sets the number of ignition sparks in multiple spark					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Ign_mpl_nr	IN	0 ... FF	0 ... 255	1	-
Number of ignition sparks					

ACTION_INFR_SetIgnCtl(IN <Ign_ctl_mod>)					
Sets the ignition in full static or half static mode					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Ign_ctl_mod	IN	1 ... 2	1 HALF_STATIC 2 FULL_STATIC		
Control mode for ignition are : HALF_STATIC or FULL_STATIC					

ACTION_INFR_GetIgnScpDiagAtm46Atic71(IN <Cyl>, OUT<Diag_scp>)					
Gets the diagnosis output for SCP detection					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number on which the diagnosis for SCP is get					
Diag_scp	OUT	0 ... 1	0 ... 1	1	-
Diagnostic of primary short circuit to battery					

ACTION_INFR_GetIgnScgDiagAtm46Atic71(IN <Cyl>, IN <min_burn_time>, OUT <V_dur_igc>, OUT<Diag_scg>)					
Gets the diagnosis output for SCG or OL detection and the burn time duration					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
Cyl	IN	0 ... 7	0 ... 7	1	-
Cylinder number on which the diagnosis for SCG or OL is get					
Min_burn_time	IN	0 ... FFFF	0 ... 262.14	0.004	ms
Minimum duration for burn time to detect ignition errors					
V_dur_igc	OUT	0 ... FFFF	0 ... 262.14	0.004	ms
Burn time duration for diagnosis validation					
Diag_scg	OUT	0 ... 1	0 ... 1	1	-
Diagnostic for primary short circuit to ground or open load					

FUNCTION DESCRIPTION:

General information:

Requirements for ACTION_INFR_ SetIgnEnable:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Inh_igc	1	-	1		0: ignition inhibited 1 : ignition enabled

Coincidence requirements:

The decision to inhibit an ignition stage is taken for each output before the turn on at TD_ON[x]. This means that no Disable / cut off can be set on a coil while charging.

Management of Ignition Cut-Off: Enable / Disable mode of Ignition Power stages

For normal Dwell Time and coil charging management:

a. No Ignition cut is allowed on a currently charging coil, *ie.* no Disable Command of a power stage is considered after the TD_ON event
   Nevertheless, if the Disable request still be present, it will be applied on the next coming to fire cylinder
b. Enable command request coming to a power stage in Disable mode will be only performed if TD_IGC[x] time to IGA_IGC point can be respected/realised.

In other case, this Enable request will be performed only on the next engine cycle (720°CRK or 360°CRK depending of the Ignition Coil mode - full or half static) if it still be present

c. No Re-Enable command can - neither will - be done on a power stage which is yet already in Enable mode

In Multiple Spark phase:

d. As for the nominal phase, no Ignition-Cut can be allowed on a currently charging coil even in Multiple Spark phase, *ie.* no Disable Command of power stage is considered after during the TD_MPL charge: if one TD_MPL is on the way, no Ignition cut is possible.
See dwell comments
The Disable request could be done/happened anywhere but it will only be executed, can only occur and be taken into account only at the end of the charging of this TD_MPL

Requirements for ACTION INFR SetIgnAngle:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Iga_igc	0.375°CRK		0.375°CRK		

Diagnosis:

Coincidence requirements:

Requirements for ACTION INFR SetIgnDwell:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Td_fac_min			1/128		
Td_igc			0.004ms		
Td_fac_max			1/128		

Diagnosis:

Coincidence requirements:

The normal request dwell time to be programmed and applied for charging the ignition coil is and has to be TD_IGC[x]
The coil starts charging with the target of achieving TD_IGC[x] at the IGA_IGC[x] angle. An update occuring for the Ignition Spark Angle will be taken into account as long as TD_MIN is not reached.
This Ignition Spark Angle change obtained when reaching TD_MIN will be taken into account and respected by a corresponding change of the coil charging time (the Dwell Time).
But, such a change of the Dwell Time to be thus and then applied can only be processed within the limits defined by TD MIN and TD MAX limits excursion range (*see drawing here after*)

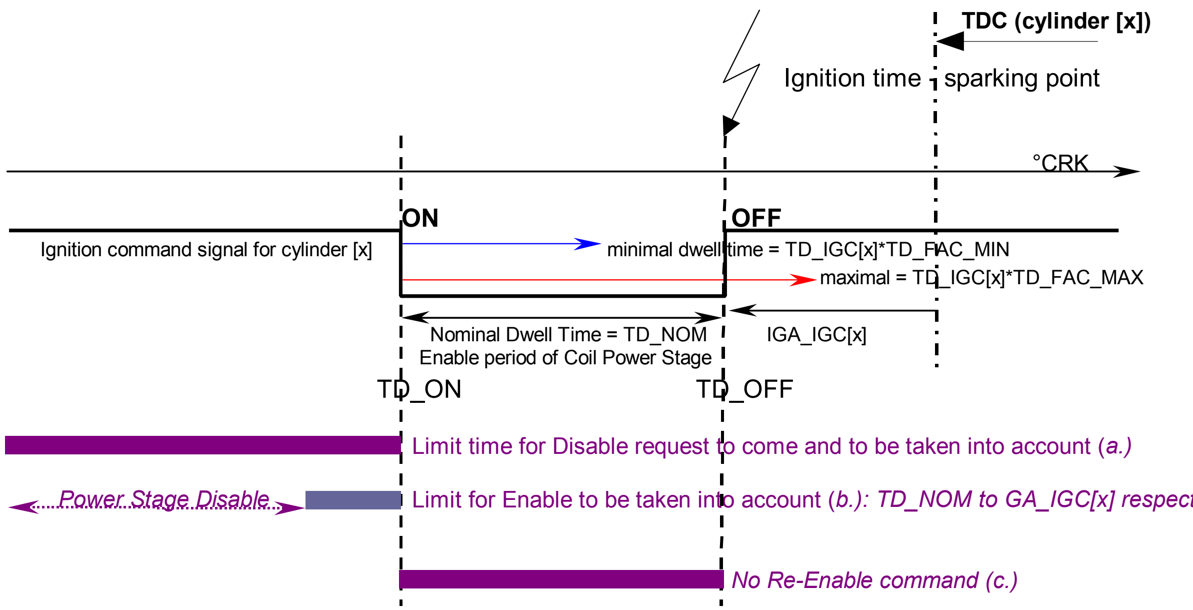
For the definition and calculation of those TD MIN and TD MAX:

TD MAX = TD_IGC[x] * TD_FAC_MAX but in any cases TD MAX <= NC_TD_LIM_MAX

*TD MAX = min(TD_IGC[x] * TD_FAC_MAX , NC_TD_LIM_MAX)*

TD MIN = TD_IGC[x] * TD_FAC_MIN but in any cases TD MIN >= NC_TD_LIM_MIN

*TD MIN = max(TD_IGC[x] * TD_FAC_MIN, NC_TD_LIM_MIN)*



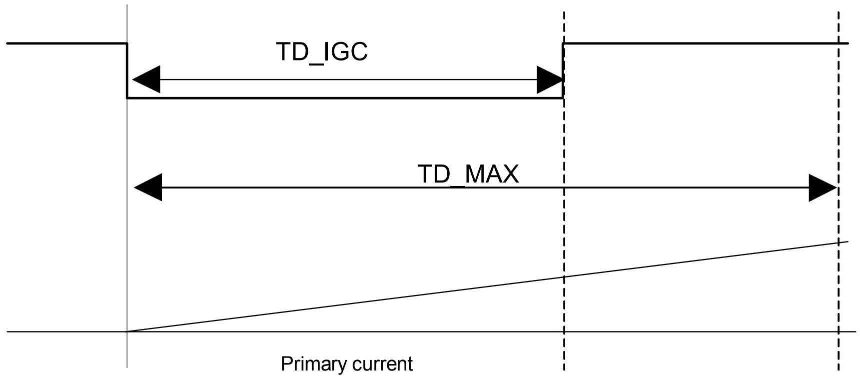
Calculations after starting the dwell time output at TD_ON

For the ignition time there are three main requests for the dwell time output.

1. A minimum dwell time TD_IGC[x] * TD_FAC_MIN has to be guaranteed to ensure ignition.
2. The dwell time has to be limited by a maximum dwell time which is defined as a TD_IGC[x] * TD_FAC_MAX.

   To prevent all cases of big negative engine speed gradients which could destroy the ignition system, by too long conduction between conduction angle calculation and spark, a timer limits the resulting conduction time (ignition power stage protection against high current).

   The ignition output is turned off immediately at TD_IGC[x] * TD_FAC_MAX (IGA_IGC[x] could not be respected).
3. In these limits (TD_IGC[x] * TD_FAC_MIN, TD_IGC[x] * TD_FAC_MAX) we have an absolute ignition angle priority.



Requirements for ACTION_INFR_SetIgcDwellTest:

Data acquisition					
Parameter	Absolute precision	Relative precision	Resolution	Coincidence with parameter	Comment
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Td_igc			0.004ms		

Diagnosis:

Coincidence requirements:

The normal request dwell time to be programmed and applied for charging the ignition coil is and has to be TD_IGC[x]
On Actuator test request activation, the coil starts charging with the target of achieving TD_IGC[x]

The coils start charging upon actuator test request pattern with the dwell time defined.

The pulse is generated once and immediately after calling the action. The Dwell time could not be updated during the pulse generation, excepting if it is set to 0. in this case, the current pulse should be stopped immediately.

Detection of SCP failure must be enabled

When a pulse is active on a cylinder no pulse could be generated on an other cylinder connected on the same ignition driver (ATIC71 or ATM46).

Requirements for ACTION_INFR_SetIgnMplDwell

Data acquisition					
Parameter	Absolute precision	Relative precision	Resolution	Coincidence with parameter	Comment
Td_mpl			0.004ms		

Diagnosis:

Coincidence requirements:

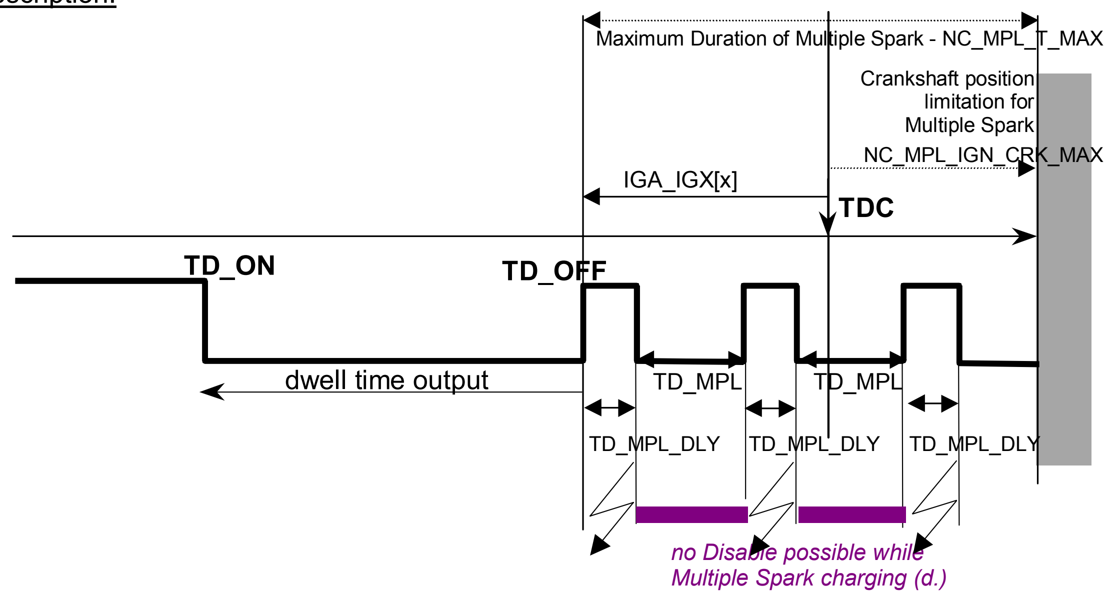
The value TD_MPL determines the conduction time of the ignition coil and the time delay TD_MPL_DLY specifies a time-out when the ignition coil is turned on for the next spark.

The number of successive sparks is defined as IGN_MPL_NR and checked at every TD_OFF.

The absolute duration of multiple sparks is limited to a maximum time defined as NC_MPL_T_MAX.

The maximum ignition angle to start a TD_MPL is defined as NC_MPL_IGN_CRK_MAX after TDC.

Description:



Requirements for ACTION_INFR_SetIgnMplDly:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Td_mpl_dly			0.004ms		

Diagnosis:

Coincidence requirements:

Requirements for ACTION_INFR_SetIgnMplNr:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Ign_mpl_nr			1		

Diagnosis:

Coincidence requirements:

Requirements for ACTION_INFR_ SetIgnCtl:

Data acquisition					
Parameter	Absolute precision	Relative precision	Resolution	Coincidence with parameter	Comment
Ign_ctl_mod					

Diagnosis:

Coincidence requirements:

This action musn't be performed after the synchronization.

Requirements for ACTION_INFR_GetIgnScpDiagAtm46Atic71:

Data acquisition					
Parameter	Absolute precision	Relative precision	Resolution	Coincidence with parameter	Comment
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Diag_scp			1		

Diagnosis:

Coincidence requirements:

1. A Dwell Time Pulse is considered as a main or a multiple dwell pulse. Thus there is no distinction in the treatment of “Dwell Time On Events” from main and multiple pulses.
2. Consequently, there is no distinction in the treatment of “Spark Events” from main and/or multiple Dwell Time pulses.
3. A “Dwell Time On Event” defines the start time of coil charging. The resolution and precision of this time is defined by the corresponding ignition channel
4. A “Spark Event” or “Dwell Time Off Event” defines the end time of coil charging. The resolution and precision of this time is defined by the corresponding ignition channel.

Each time a “Dwell Time On Event” occurs, the time during which the IGBT_DIAG Pin indicates a SCP condition has to be integrated within a time window (*defined below*). If this integrated time inside this window exceeds NC_INI_DEAC_CTR, the cylinder that has produced the most recent Dwell Time On event must be switch off as fast as possible for the current ignition cycle. The time window begins at any “Dwell Time On Event” and ends if either one of the following criteria is met:

cr.a) another “Dwell Time On Event” from any Cylinder occurs. This restarts the time window and thus the integration of the time for which IGBT_DIAG indicates SCP conditions cr.b) a “Spark Event” of the same Cylinder that started charging the coil occurs. “Spark Events” of any other Cylinder are ignored.

cr.c) the integrated time for which the IGBT_DIAG pin indicates a SCP condition exceeds NC_INI_DEAC_CTR

cr.d) the time period back to the most recent “Dwell Time On Event” exceeds NC_IGBT_CUT_OFF_T.

Note that: * in case of overlapping of dwell time periods no short circuit detection can be guaranty due to HW design and limitation.

* If Dwell Time periods are overlapping, only the most recent one that has the latest started a coil charging (*last and most recent TD ON event, whatever it is*) will trigger the activation of this protection (BSW service) function. Hence, the last occurring TD ON event will define the coil to be considered by this function and thus diagnosed and protected (cut off if needed, when SCP failure appears)

With overlapping Dwell Time periods, there is thus a risk to do not diagnose (and switch off) the correct IGBT in real failure conditions

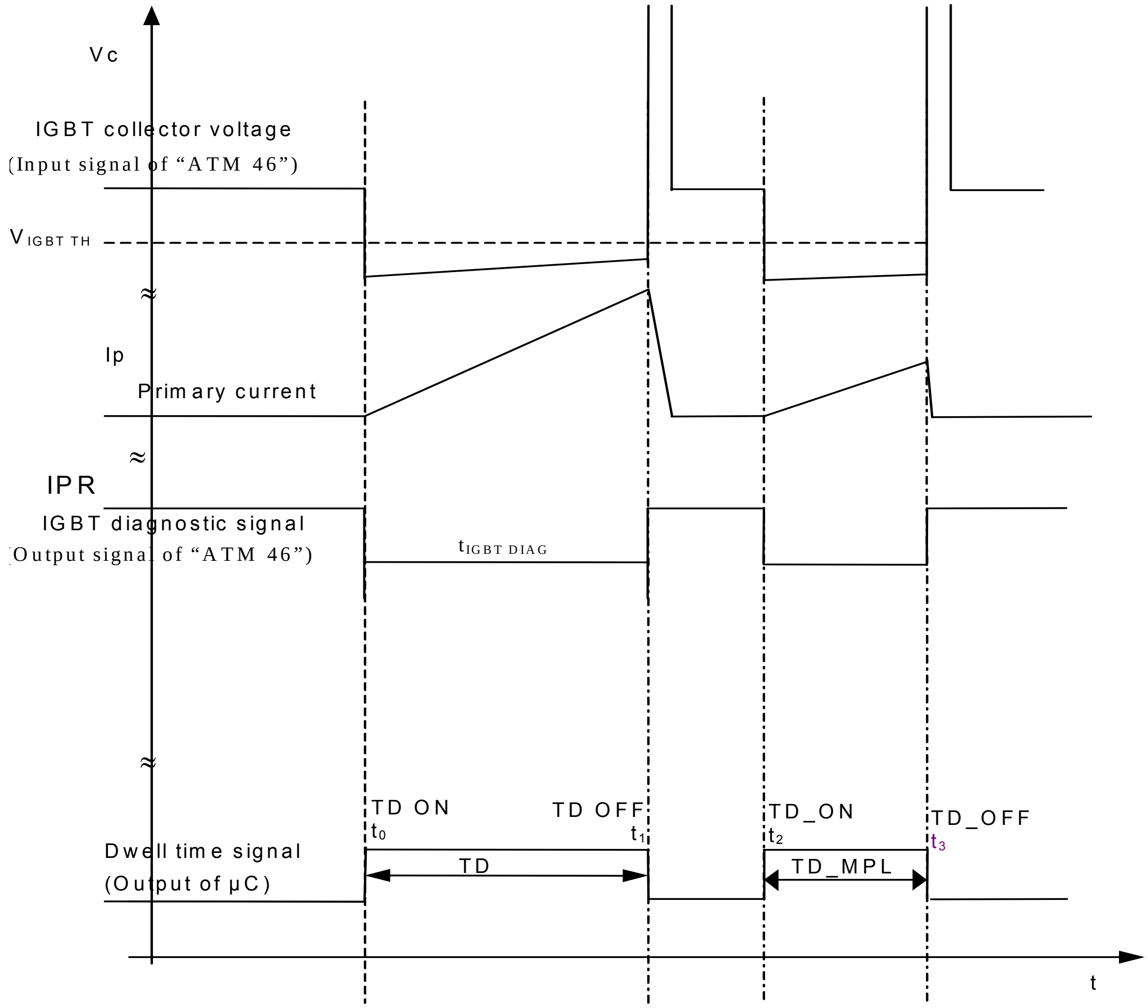
Dwell Times overlapping conditions may occur, in normal running, with too long settings of Dwell time in half and full static ignition modes.

Dwell Times overlapping conditions will occur, in Ignition Limp Home running (*ie. ½ static mode usage for coils charging*).

One ATM 46 could handle only 4 ignition coils. If there are more than 4 ignition coils 2 ASIC have to be used. To reduce the possibility of overlapping we have to use them in a crossed order.

Timing diagram:

Normal Condition:

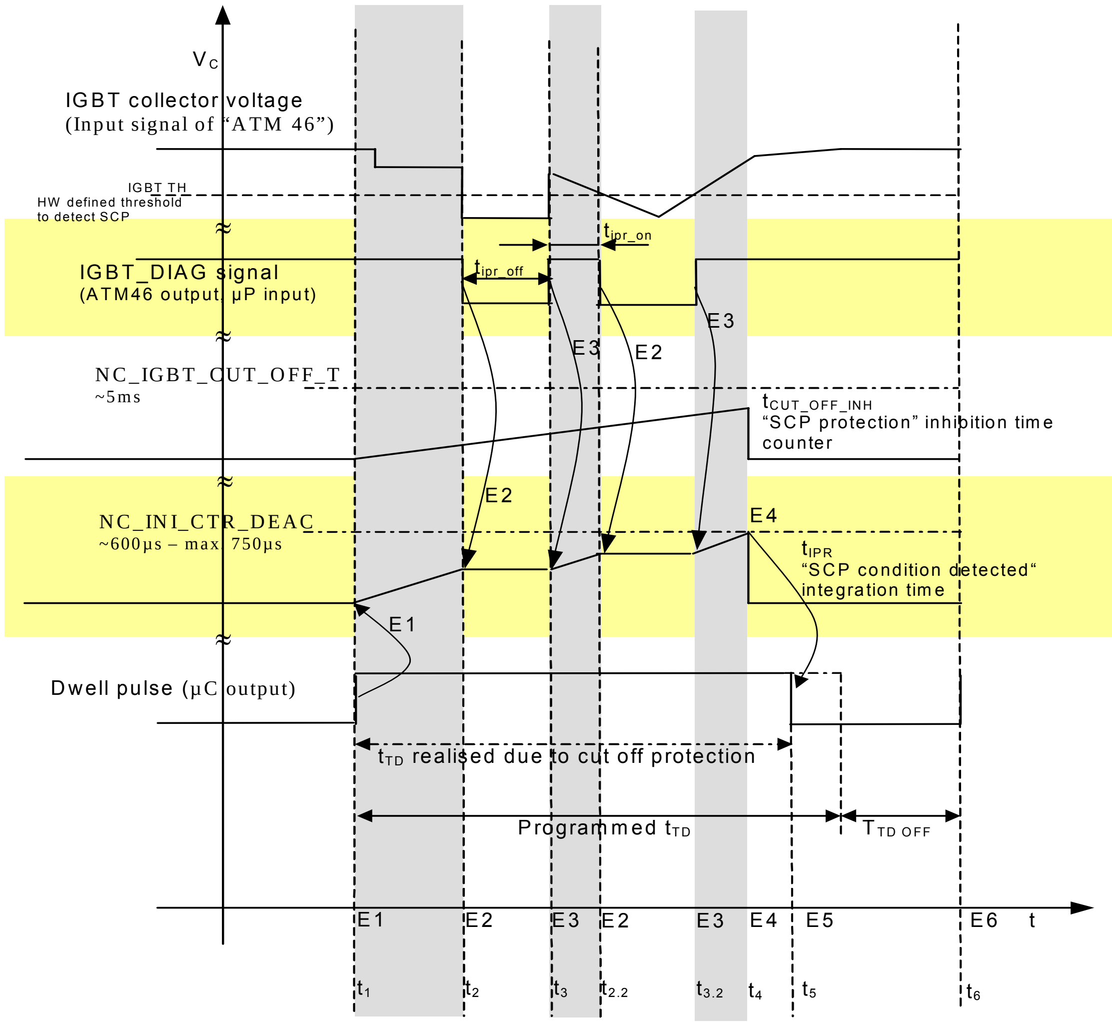


V_{IGBT_TH} is the threshold voltage of the failure detection of IGBT protection. It is fixed by the component ("ATM 46") at ≈3V.

The input signal of the "ATM 46" is the IGBT collector voltage.

The output signal of "ATM 46" is low as long as the battery voltage is less than the threshold of $V_{IGBT\ TH}$ if no failure is detected.

Cut Off Condition and integration of SCP condition for Dwell pulse:



A dwell pulse is considered as a main or multiple dwell pulse, no distinctions are done as both can be critics in case of SCP failure.

When the collector voltage is higher than the threshold (V_{IGBT_TH}), it indicates a SCP condition on IGBT. In this case, the IGBT_DIAG signal from the ASIC is high (in SCP level).

a) Events and Actions definition

E1: Dwell Time On Event – Start the charging of the coil
When this event occurs at t_1:
- Clear the SCP pulse periods measured up to now (*ie. the "SCP condition detected" integration time*)
– start / enable to integrate the time for which the IGBT_DIAG Pin indicates a SCP condition

E2: IGBT output without SCP condition (t_2) or transition to No SCP ($t_{2.2}$ - *SCP disappears*)
When E2 occurs, the *"SCP condition detected" integration time* is stopped (*no or no more SCP conditions seen*)

E3: IGBT output in SCP conditions (t_3) or transition to SCP ($t_{3.2}$ - *SCP occurs again*)
When E3 occurs, the *"SCP condition detected" integration time* is started or re-started (*SCP conditions seen or re-seen again*)

E4: Cut Event from the IGBT SCP Protection function

E4 will not occur in normal condition when the coil encounters no SCP conditions (no or too short E3 is met, only E2)

t_{IPR} designates the *“SCP condition detected“ integration time* acquired (*see after*)
t_{IPR} is permanently checked when this protection function is active

IGBT Cut Off in case of SCP detection validated:

If t_{IPR} > NC_INI_CTR_DEAC

And $(t_4 – t_1)$ < NC_IBGT_CUT_OFF_T

Then Switch off the ignition channel that has produced the most recent Dwell Time On Event (*most recent or latest E1 event*) as fast as possible

Dwell Time applied in this case will be “t_{TD} realized due to cut off protection” function and not the “Programmed t_{TD}”

Else No action on ignition channel

EndIf

E5: Spark Event at the cutting of Ignition Coil charging

E5 occurs at t_5. E5 / t_5 can be either:
- the normal “*Programmed t_{TD}*” if this SCP protection function has not been reached (no E4) or
- the Coil charging Time corresponding to *“t_{TD} realized due to cut off protection”* function if this SCP protection function has been reached (at E4)

When E5 occurs: the *“SCP condition detected“ integration time* (t_{IPR}) is stopped

E6: Identical to Dwell On Event E1 for the next following coil coming to charging mode on the same ASIC diagnostic line

b) Times definition

t_{ipr_on} = time period that indicates a SCP conditions (*ie. V_C > IGBT_TH, the IGBT_DIAG signal from the ASIC indicates the SCP*). During this time, the *“SCP condition detected” integration time* – t_{IPR} – is enabled.

t_{ipr_off} = time period where no SCP conditions exist (*ie. V_C < IGBT_TH*). This can happen also in between IGBT SCP diagnosis pulses detection

t_{TD} = time period of any dwell pulse. It correspond normally to the programmed Dwell Time period for charging the Ignition coil. Here, it can be shortened (*cut off of the Coil Charging*) by the IGBT SCP protection function.

$t_{TD\ OFF}$ =time period between any successive dwell pulses

$t_{CUT\_OFF\_INH}$ = For the Ignition Coil under SCP Protection diagnostic, the “Cut Off” functionality is deactivated after a time longer than
$t_{CUT\_OFF\_INH}$ > NC_IGBT_CUT_OFF_T
If such a time out occurs, the dwell time is not interrupted cause it corresponds to case where IGBT handles over a situation with a high gradient of battery tension and not a short circuit to battery

t_{IPR} = *“SCP condition detected“ integration time* acquired
This Pulse Measurement Method has to be active – which is equivalent to: the IGBT_DIAG input to the µP at SCP level has to be measured – between t_0 and either criteria cr.a), cr.b), cr.c) or cr.d) is met **whichever** comes first.

*The minimum length of IGBT_DIAG pulses to be detected should be as short as possible (minimal requirement: 8µs = 2*4µs)*

Warning: due different HW and/or µP architecture and implementation constraint limitations

1. Every events E1 to E6 must be processed within a Maximum tolerable time delay smaller than 50µs
2. The Dwell Time and Dwell Charging period succession should be limited to:
   $100µs \le t_{TD} \le 60ms$ and $100µs \le t_{TD\ OFF}$
3. Event E4 = "Switch off the ignition channel that has produced the most recent Dwell Time On Event" should only occur within 50µs after SCP detection confirmation
4. IGBT_DIAG pulses shorter than 50µs have a risk of not being processed

Requirements for ACTION_INFR_GetIgnScgDiagAtm46Atic71:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
Cyl	1	-	1		Index range: 0 ... NC_CYL_NR-1
Min_burn_time			0.004ms		
V_dur_igc			0.004ms		
Diag_scg			1		

Diagnosis:

Coincidence requirements:

For the Spark Burn Time Duration measurement, the functionality is:

- To ensure a correct measurement in case of spark blow out, only the Active Level of Burn Time measurement signal from the ATM46 (IGN_DIAG) is integrated in the time window defined by two consecutive main spark events on the same ASIC
- Active Level of Burn Time measurement signal can be "High" or "Low" depending on HW configuration
- Minimum length of Active Level pulses to be detected is 50µs and Minimum inter-space between Active Level pulses to be detected is 50µs. Consequently, IGN_DIAG pulses shorter than 50µs have a risk of not being processed.

Sparking events from multiple spark pulses do not restart the time window for measurement. Indeed, the Burning effect that can be created by any Multiple spark pulses is *de facto* integrated and cumulated with the Burning Duration from the preceding Main Spark pulse.

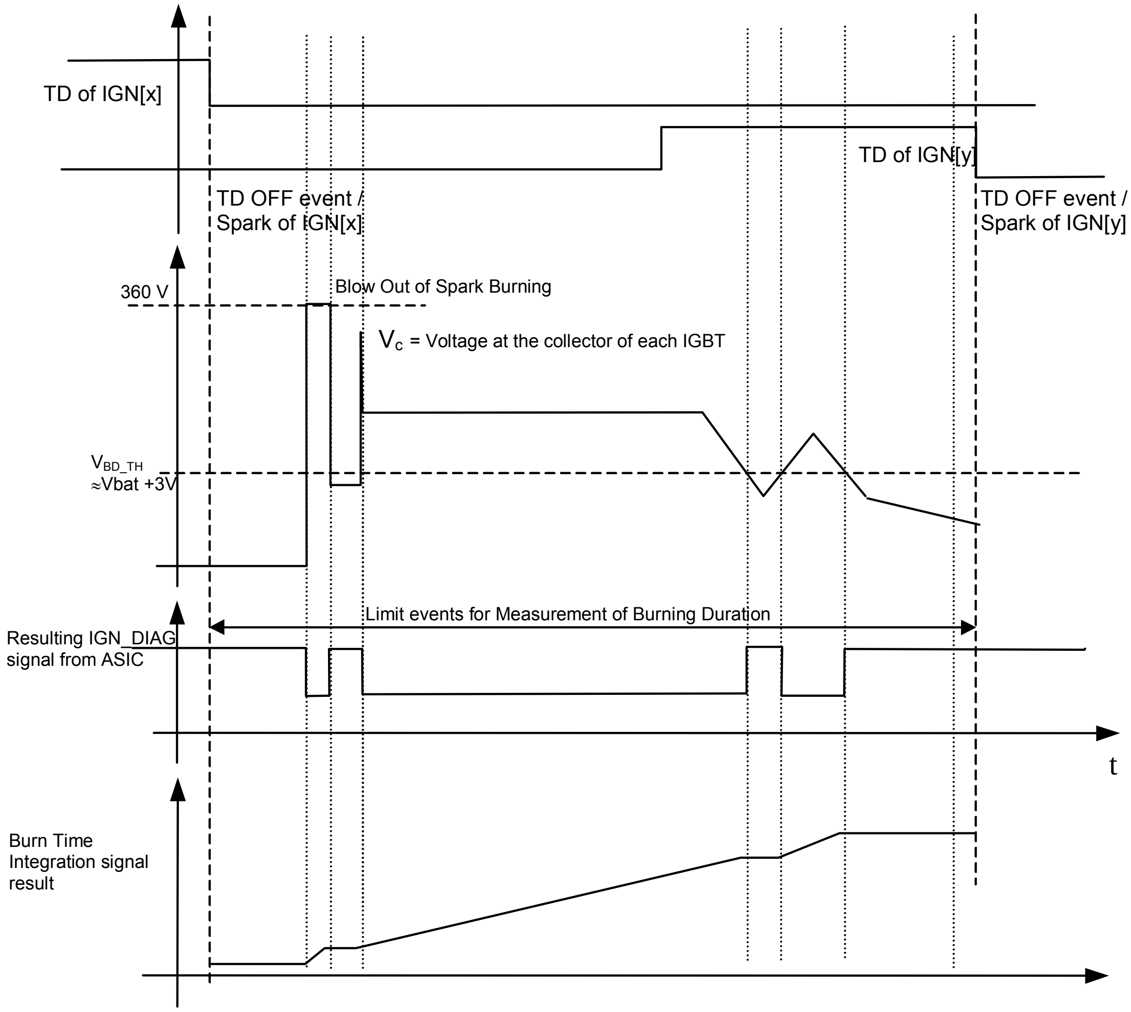
At the end of the time measurement window started by cylinder[x], the measured burning duration is read out and written into the variable V_DUR_IGC[x] for the corresponding logical cylinder [x]. The time for Burn Time measurement is then cleared and prepared for the next measurement.

Consider that only one IGN_DIAG signal exists (*this IGN_DIAG signal represents a wired "OR" of all the presents coils channels*) which will limit the measurement range.

In Limp Home and ½ static modes, 2 main spark events – on one ATM46 diagnostic signal line – occur at the same time. This means that the time windows for measurement of (*at least*) two cylinders overlap. In such cases, due to the "OR" wiring on the signal information, only the cylinder on which the most recent main spark event (*TD OFF*) is produced will define and will be identified for the time window measurement.
Therefore, the reported burning duration in the interface variable V_DUR_IGC[x] may not correspond to right real burning cylinder [x] in case of Limp Home mode and ½ static coils.

Each ATM46 could analyze 4 ignition coils. If diagnostics for more the than 4 coils are necessary, a parallel ATM46 have to be used.

Timing diagram:



2.35 Acquisition of sensor voltages within AIRT:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VP_TIA[NC_SENS_NR_TIA]	O/V	0...7FFFH	0...4.999847	1.52588E-4	V
Air intake temperature raw value: one value for each TIA sensor of AIRT					

Input data:

NC_SENS_NR_TIA			

Import actions:

ACTION_AIRT_GetVpTia(IN <CtrTiaSens>, OUT <VpTia>)
Action to read the sensor voltage of different TIA sensors

2.35.1 General:

General information:

The purpose of this module is to provide the air sensor voltage out from the basic SW value and provide a consistent value for one calculation run.

Application Condition

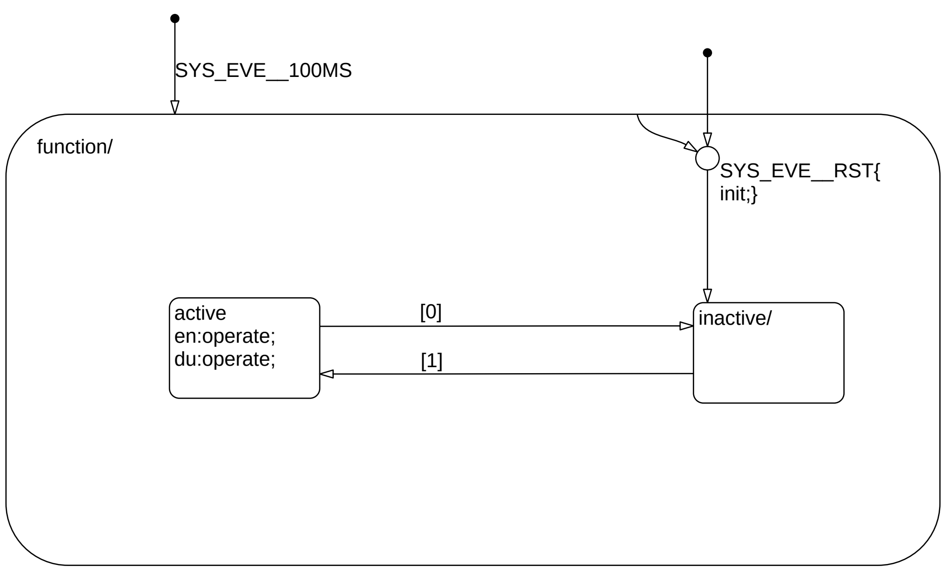
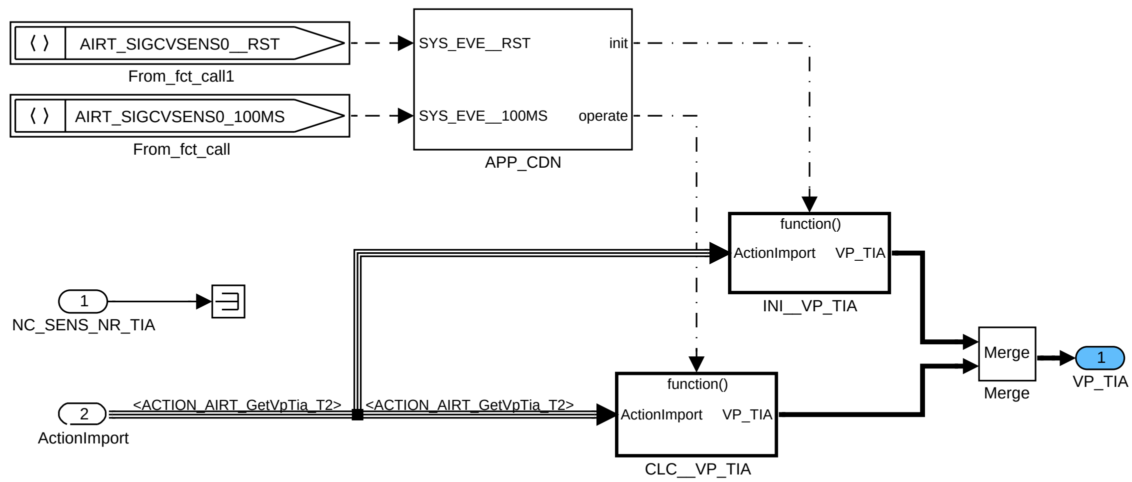


Figure 12 AIRT_SIGCVSENS0/ APP_CDN/ APP_CDN

Function Description



SDA_SRS / SDA 4.0 01−Apr−2004

Figure 13 AIRT_SIGCVSENS0

2.35.1.1 SUBFUNCTION: CLC__VP_TIA

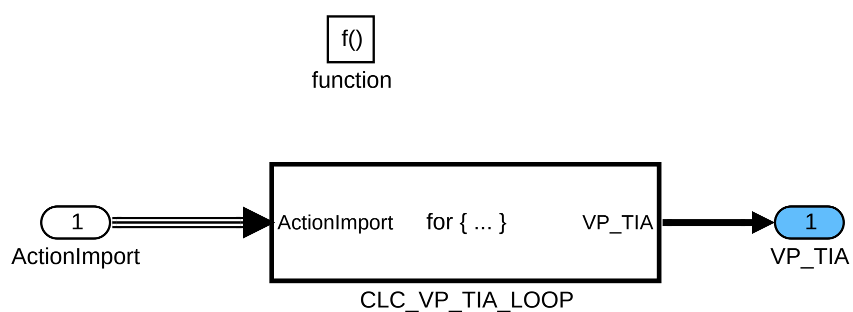


Figure 14 AIRT_SIGCVSENS0/ CLC__VP_TIA

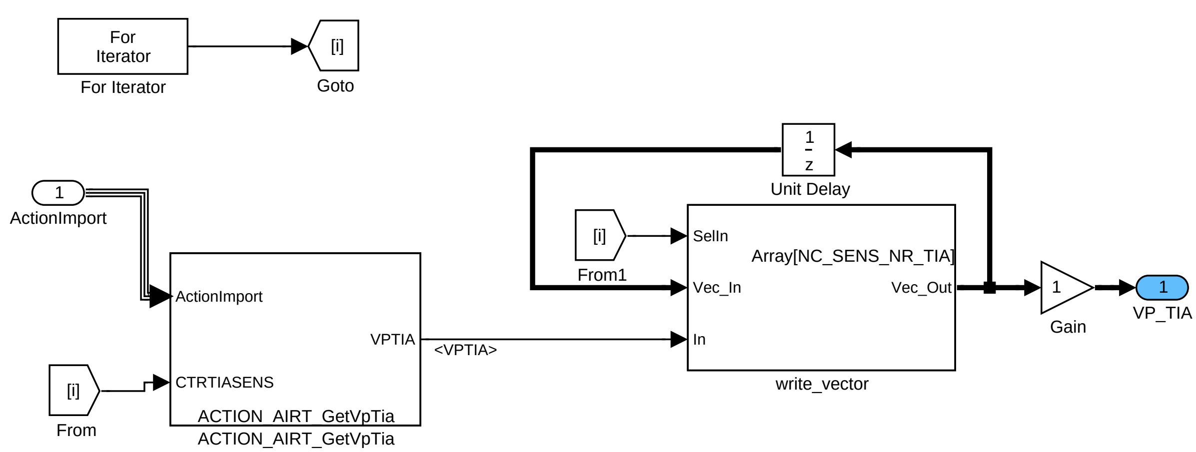


Figure 15 AIRT_SIGCVSENS0/ CLC__VP_TIA/ CLC_VP_TIA_LOOP

2.35.1.2 SUBFUNCTION: INI__VP_TIA

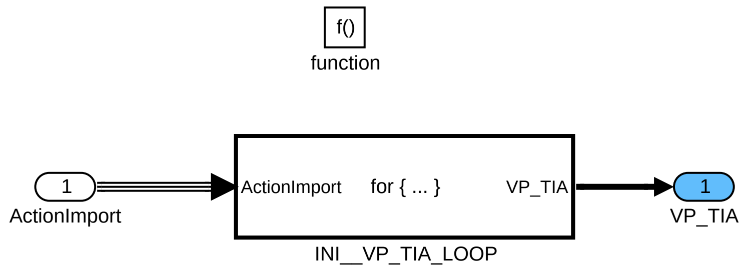


Figure 16 AIRT_SIGCVSENS0/ INI__VP_TIA

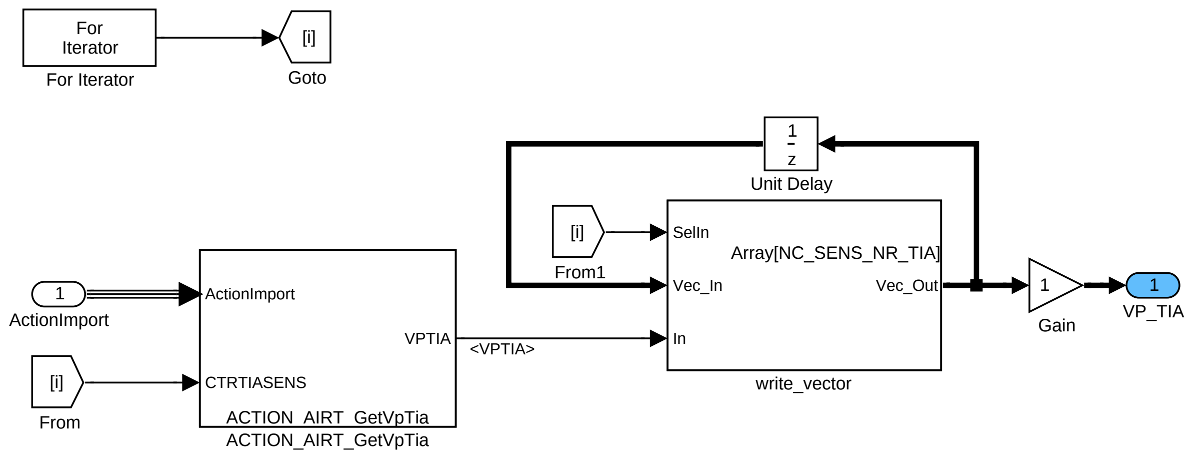


Figure 17 AIRT_SIGCVSENS0/ INI__VP_TIA/ INI__VP_TIA_LOOP

2.36 HVAC - Requirements to Infrastructure interface

Export actions:

ACTION_INFR_GetLvImAcin(OUT < lv_im_acin >**)**
This action reads the information for air condition selected.

Description for actions:

ACTION_INFR_GetLvImAcin (OUT < lv_im_acin >**)**					
This action reads the information for air condition selected					
Name	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
lv_im_acin	OUT	0...01H	0...1	1	-
Air condition selected					

FUNCTION DESCRIPTION:

General information:

ACTION_INFR_GetLvImAcin:

This action reads out "air condition selected" information with a defined coherency/coincidence.

Requirements for ACTION_INFR_GetLvImAcin:

Data acquisition					
Parameter	**Absolute precision**	**Relative precision**	**Resolution**	**Coincidence with parameter**	**Comment**
lv_im_acin	-	-	<bit coded>		Flag for air condition selected: 0 = not selected 1 = selected

Diagnosis: no electrical diagnosis done here

Coincidence requirements: no coincidence requirements to other events

2.37 Acquisition of battery data

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
CUR_BAT_EFF	O/V	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Effective battery current					
CUR_BAT_EFF_MMV	O/V	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Effective mean moving value of the battery current					
CUR_BAT_INI	O/V	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Initial batter current					
CUR_BAT_MES	V	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Battery Charging Current					
CUR_BAT_SD	V	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Self discharging Current of Battery					
LV_CUR_BAT	V	0... 1H	0... 1	1	[-]
Flag showing the battery is charged currently					
LV_VB_SENS_LIN_INI_END	O/V	0... 1H	0... 1	1	[-]
End of battery sensor data initialization					
T_TBAT_MES_MMV	V	0... FFFFH	0... 65535	1	[-]
Sample number value for TBAT_MES_MMV calculation					
TBAT_CLC	O/V/S	0... FEH	-48... 142.5	0.75	[°C]
Calculated Battery Temperature (if LC_TBAT_CLC = 1 at Battery Fluid, otherwise equal TBAT_MES_MMV)					
TBAT_CLC_INI	V	0... FEH	-48... 142.5	0.75	[°C]
Calculated Battery Temperature at Initialisation (at Battery Fluid)					
TBAT_INI	O/V	0... FEH	-48... 142.5	0.75	[°C]
Measured Battery Temperature at Initialisation					
TBAT_MES	V	0... FEH	-48... 142.5	0.75	[°C]
Measured Battery Temperature					
TBAT_MES_MMV	V	0... FEH	-48... 142.5	0.75	[°C]
Measured Battery Temperature (mean value)					
TBAT_SUM	V	0... FDFF02H	-3145680... 9338738	0.75	[°C]
Accumulated Battery Temperature (for Mean value calculation)					
V_CUR_BAT	V	0... 3FFH	0... 4.9951171875	4.88281e-3	[V]
Battery Current Input signal in Voltage					
V_TBAT	V	0... 3FFH	0... 4.9951171875	4.88281e-3	[V]
Battery Temperature Input signal in Voltage					
VB_H	V	0... 3FFH	0... 26	0.0254154	[V]
Battery voltage with High resolution					
VB_H_INI	O/V	0... 3FFH	0... 26	0.0254154	[V]
Initial high Battery voltage					
VB_H_MMV	O/V	0... 3FFH	0... 26	0.0254154	[V]
High Mean Moving Voltage value					

Input Data:

CUR_BAT_BAS	CUR_BAT_LIN	CONF_BAT	CONF_BAT_LIN
LV_SENS_BAT_LIN_VLD	NC_FAC_VB_RATIO	TBAT_BAS	TBAT_LIN
T_ES	VB	VB_LIN	VB_MES

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_CUR_BAT_EFF	1	0... FFH	0... 0.99	3.88235e-3	[-]

Filter constant for Battery temperature calculation					
C_CRLC_TBAT	1	0... FFH	0... 0.99	3.88235e-3	[-]
Filter constant for Battery temperature calculation					
C_CRLC_VB_H	1	0... FFH	0... 0.99	3.88235e-3	[-]
Calibration for high resolution battery voltage computation					
C_CUR_BAT_EFF_MMV_SUB	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Substitute value for CUR_BAT_EFF_MMV					
C_CUR_BAT_EFF_SUB	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Substitute value for CUR_BAT_EFF					
C_CUR_BAT_INI_SUB	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Substitute value for CUR_BAT_INI					
C_T_TBAT_MES_MMV	1	0... FFFFH	0... 65535	1	[-]
Sample number for calculation of mean value for TBAT_MES					
C_TBAT_CLC_SUB	1	0... FEH	-48... 142.5	0.75	[°C]
Substitute value for TBAT_CLC					
C_TBAT_INI_SUB	1	0... FEH	-48... 142.5	0.75	[°C]
Substitute value for TBAT_INI data					
C_VB_H_INI_SUB	1	0... 3FFH	0... 26	0.0254154	[V]
Substitute value for VB_H_INI					
C_VB_H_MMV_SUB	1	0... 3FFH	0... 26	0.0254154	[V]
Substitute value for VB_H_MMV					
IP_CUR_BAT_MES	16	0... FFFFH	-120 ...120	3.66217e-3	[A]
LDP_V_CUR_BAT_CUR_BAT_MES	16	0... 3FFH	0... 4.9951171875	4.88281e-3	[V]
Battery charging current linear table					
IP_CUR_BAT_SD	12*12	0... FFFFH	-120 ...120	3.66217e-3	[A]
LDP_VB_CUR_BAT_SD	12	0... FFH	0... 26	0.1019608	[V]
LDP_TBAT_CLC_CUR_BAT_SD	12	0... FEH	-48... 142.5	0.75	[°C]
Battery Selfdischarging current					
IP_FAC_TBAT_CLC_INI	12	0... FFH	0... 0.99	3.88235e-3	[-]
LDP_T_ES_FAC_TBAT_CLC_INI	12	0... FFFFH	0... 65535	1	[min]
Weigthening Factor for TBAT_CLC_INI calculation					
IP_TBAT_MES	12	0... FEH	-48... 142.5	0.75	[°C]
LDP_V_TBAT_TBAT_MES	12	0... 3FFH	0... 4.9951171875	4.88281e-3	[V]
Battery Temperature linear table					
LC_VB_SENS_ACQ_SUB	1	0... 1H	0... 1	1	[-]
Usage of substitute calibration value (typical value 0 : not actif)					

General Information

For the Acquisition of Battery Temperature and Battery Current there are two possible Sensors. The Analog Sensor (CONF_BAT_LIN = 0) send the raw values for Battery Sensor (Current and Temperature) via Analog INPUT. These signals are measured by continuous conversion (10bits) every 10msec.
The Battery Sensor is a Hall Effect Sensor with a NTC Temperature element. The Hall sensor provides a voltage Signal, which is directly proportional to the current, which is going in or out of the battery. According to the characteristic line of the Sensor small voltages shows a negative current (Battery is discharged) while high Voltage shows positive current (Battery is charged)

The Semi Smart Battery Sensor (CONF_BAT_LIN = 1) send the signals via LIN BUS communication. The Temperature Signal is already based on a model, which gives the Fluid temperature inside the battery

To take into consideration the current losses of the battery, the Self discharging current is subtract from the measured current.

general specification

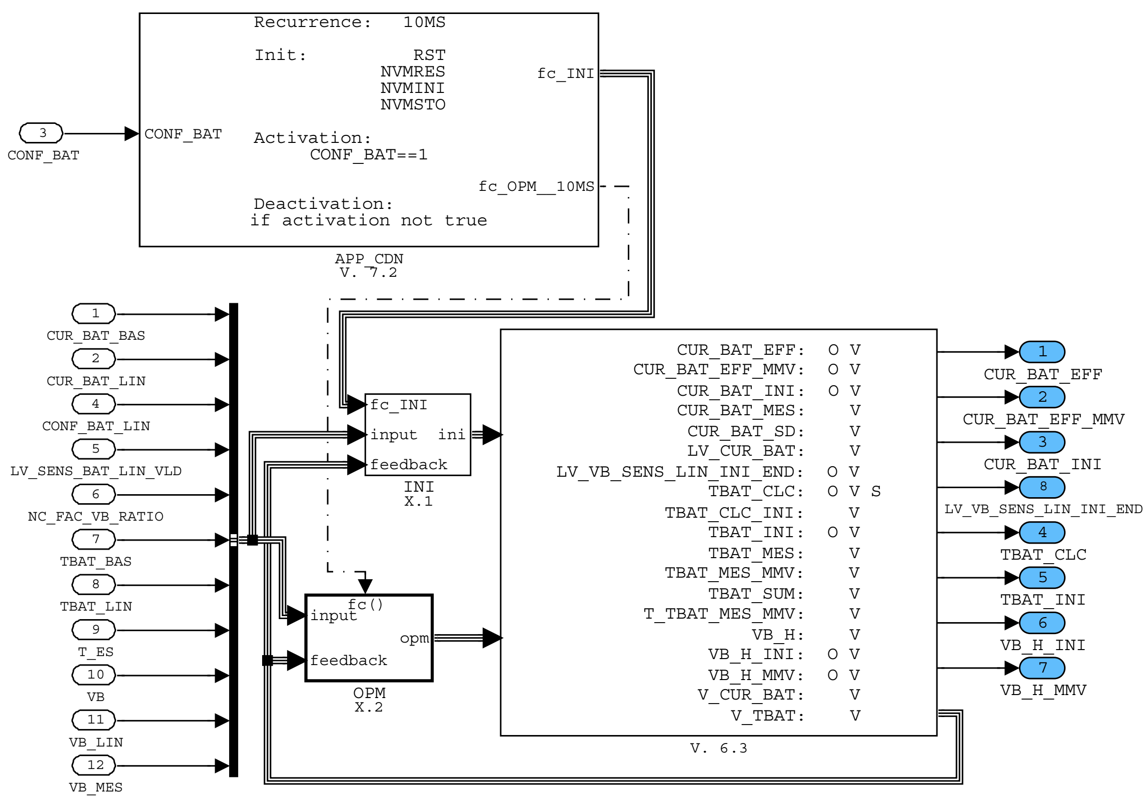
The Analog Temperature Sensor measures the Temperature at the Sensor location. As the Actual Battery Temperature increased quicker than the Battery fluid temperature a Low Pass Filter is used. The Initial value is calculated depending on Engine off time, calculated value at engine stop and Sensor value at start.

For Battery Management it is necessary to know the Battery voltage with higher resolution. This can be done either with Battery Sensor via LIN BUS communication or via Analog Input signal from VB_MES.

Application Conditions

Initialization: RST, NVMRES, NVMINI, NVMSTO
Recurrence: 10MS
Activation: CONF_BAT==1
Deactivation: if activation not true

Function description



SDA_SRS / SDA V 5.3.2 / 05−Mar−2008

Figure 18:

2.37.1 Initialization

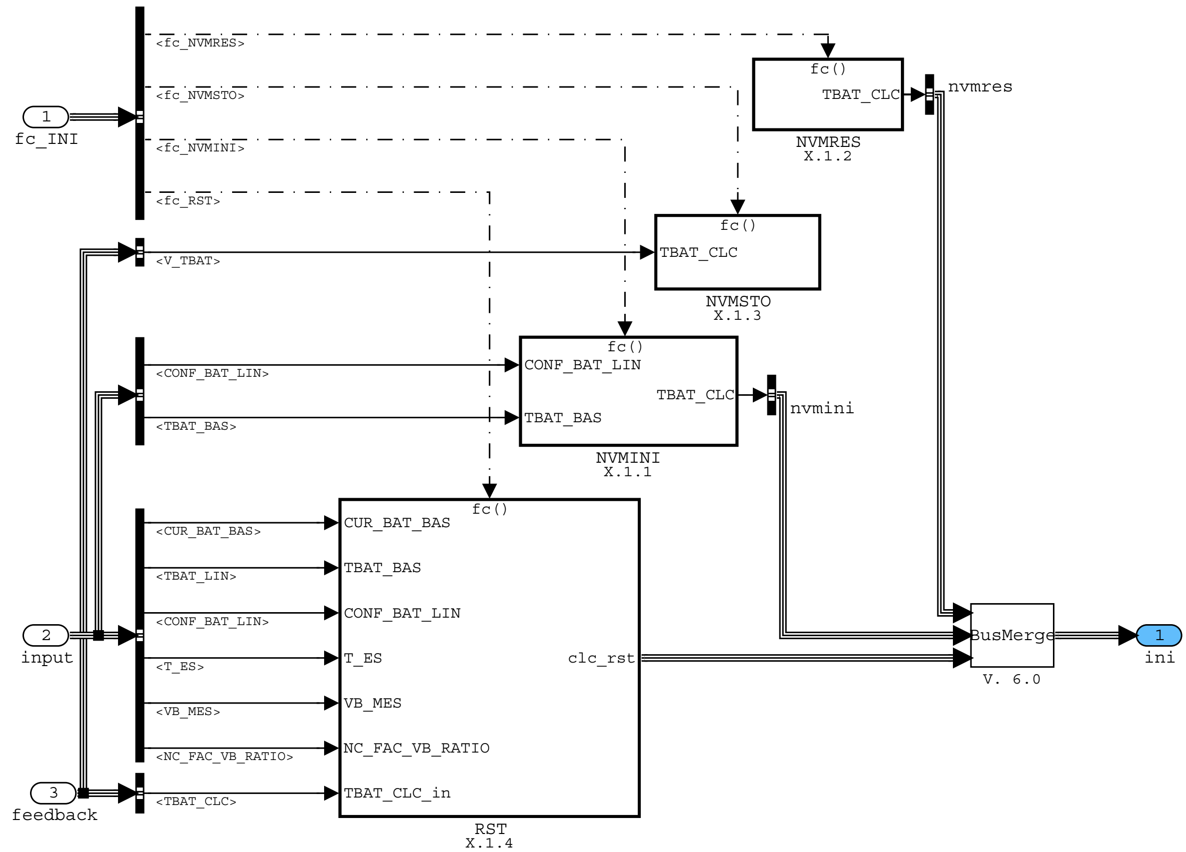


Figure 19:

2.37.1.1 Initialization for Non-volitile memory variable

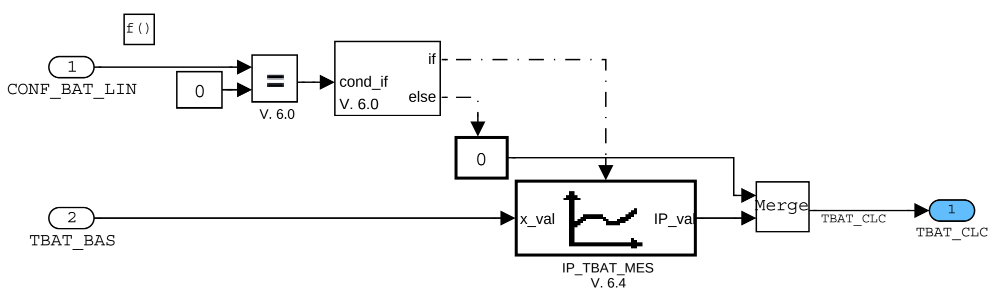


Figure 20:

2.37.1.2 Restore value for Non-volitile memory variable

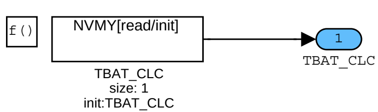


Figure 21:

2.37.1.3 Calculation and Store the value for Non-volitle memory variable

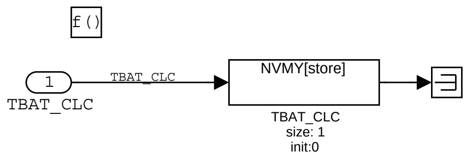


Figure 22:

2.37.1.4 Initialization at Reset

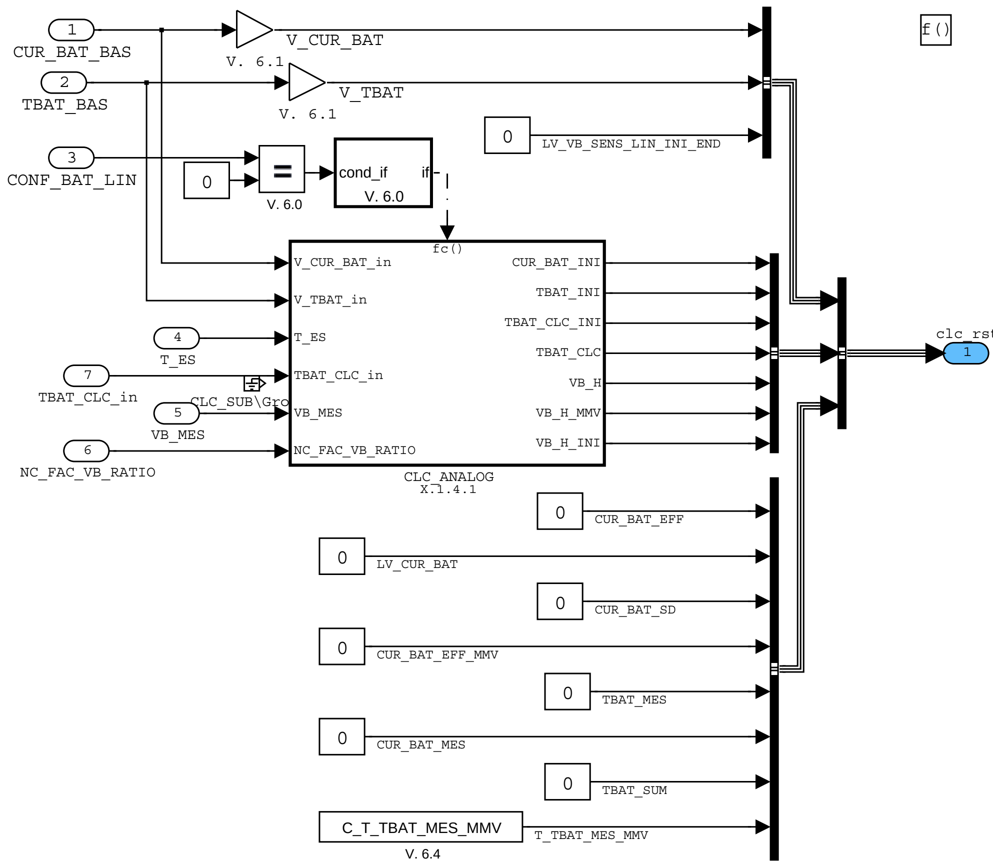


Figure 23:

2.37.1.4.1 Analog battery sensor is enable

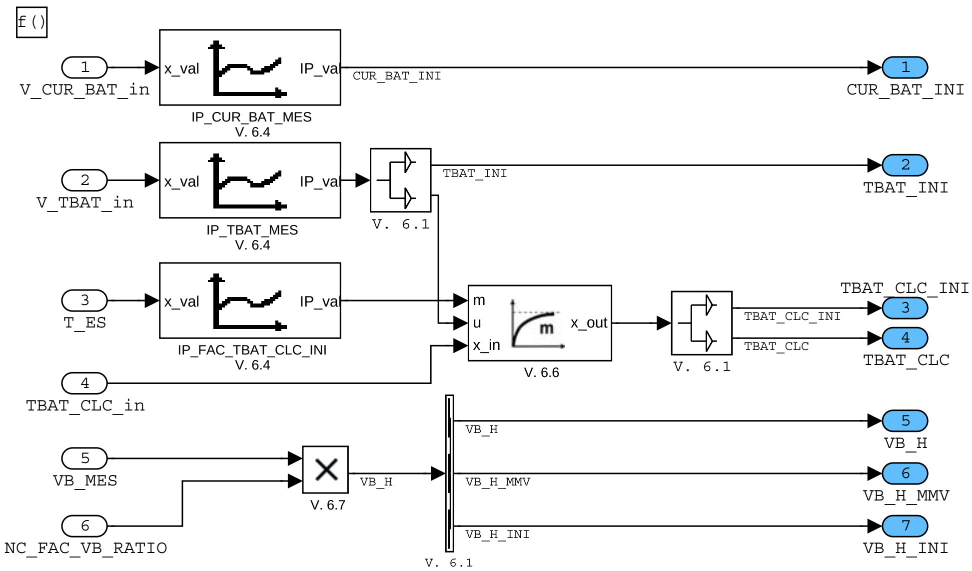


Figure 24:

2.37.2 Formula Section

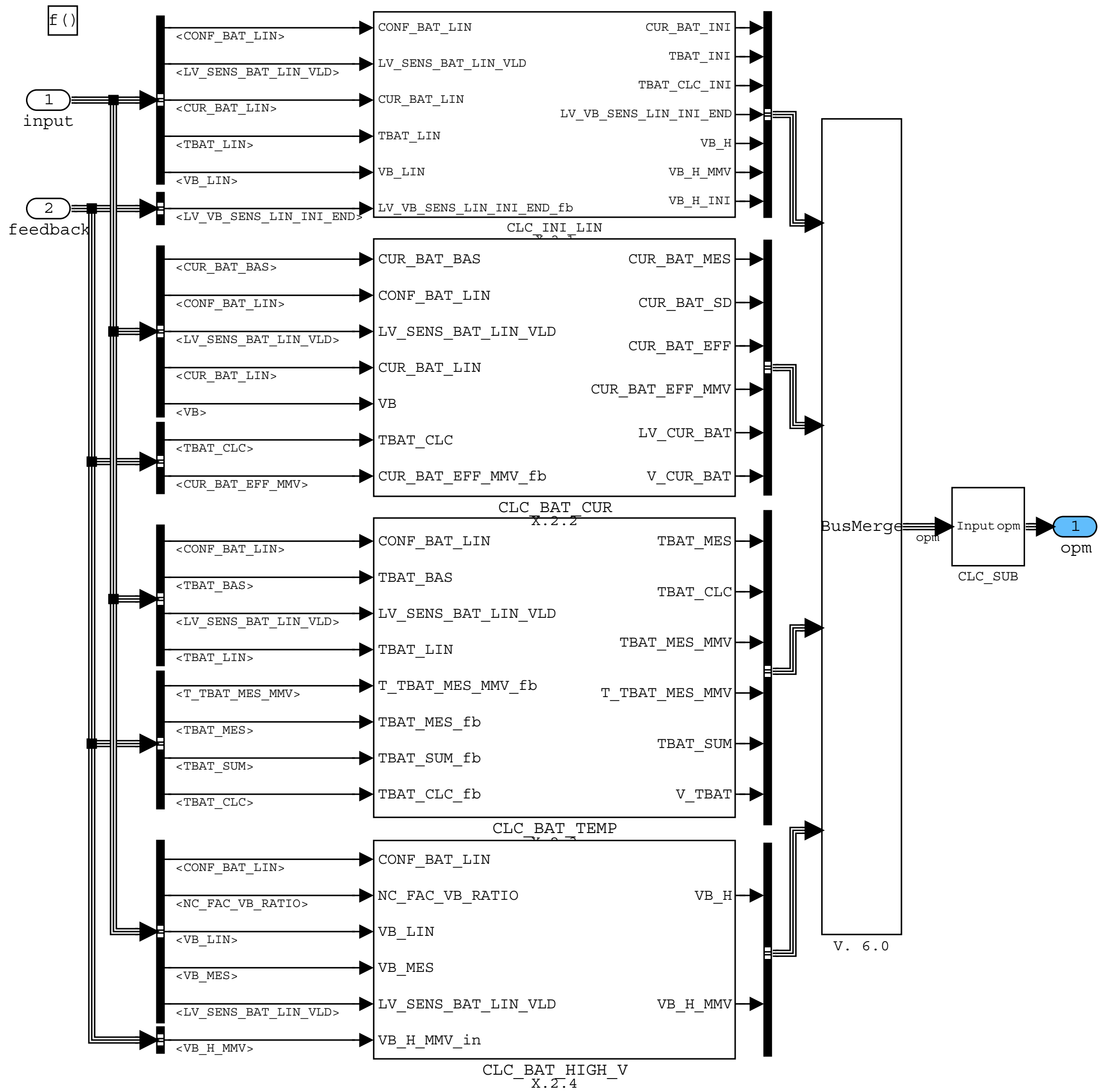


Figure 25:

2.37.2.1 Check if the calculation at initialization for LIN sensor is requested

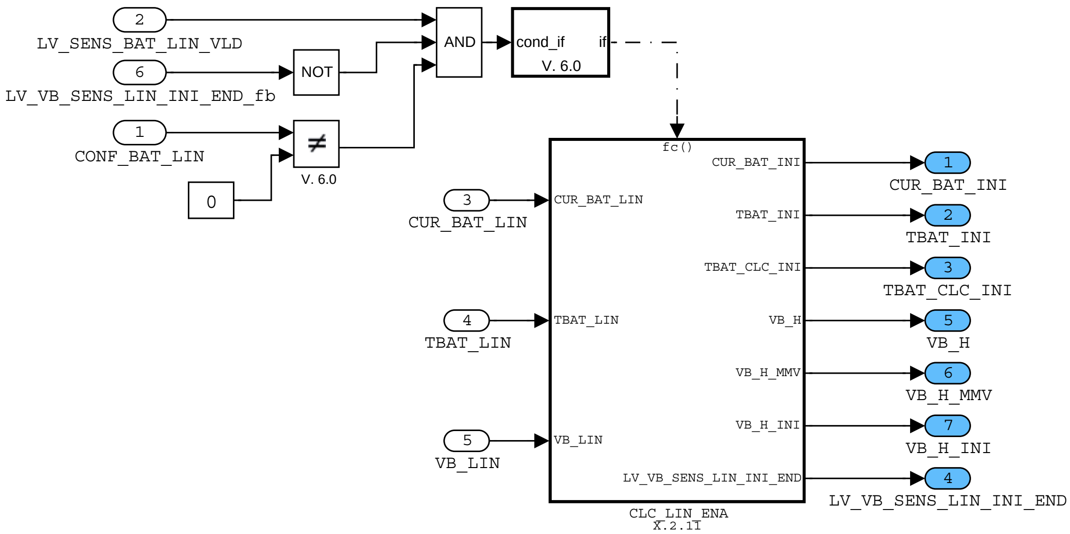


Figure 26:

2.37.2.1.1 Calculation of battery sensor current and temperature at initialization

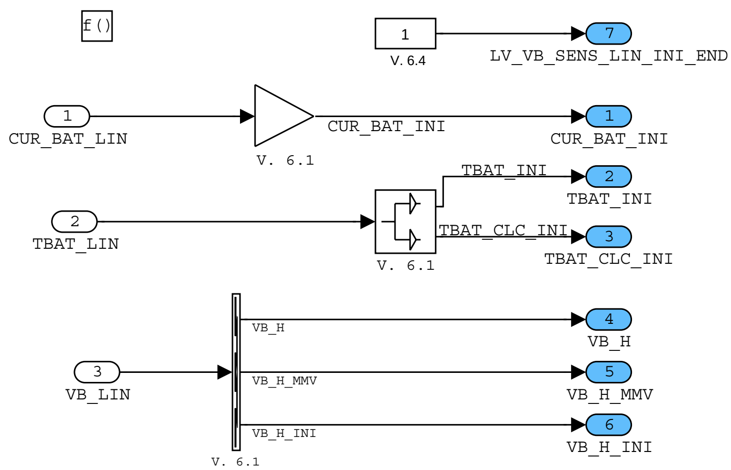


Figure 27:

2.37.2.2 Calculation of battery current

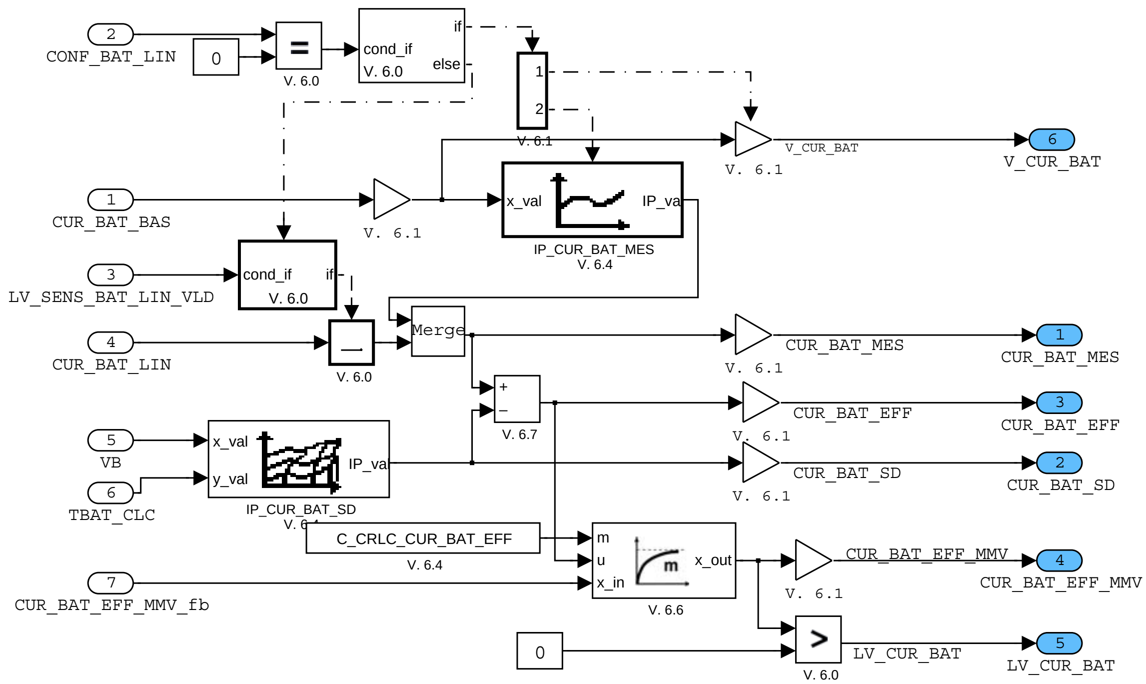


Figure 28:

2.37.2.3 Calculation of battery temperature:

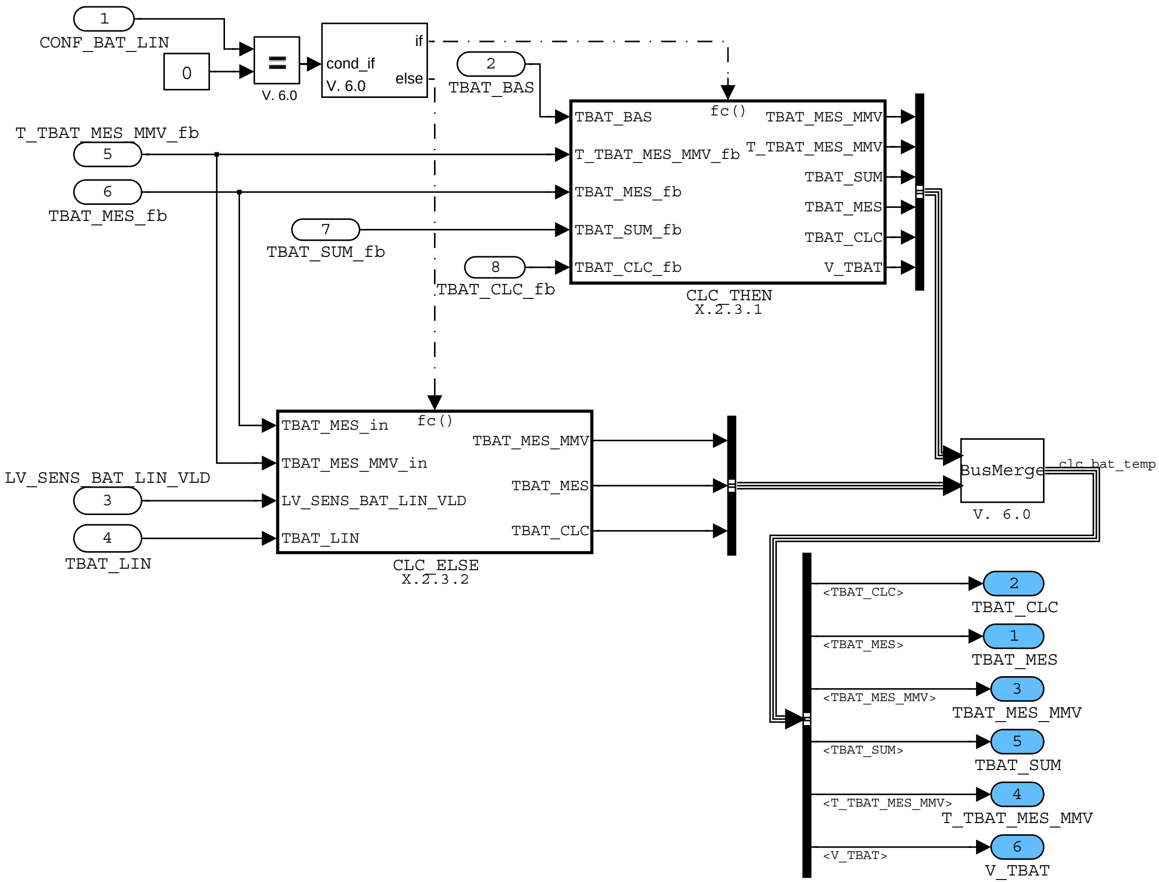


Figure 29:

2.37.2.3.1 Linear battery sensor is enable

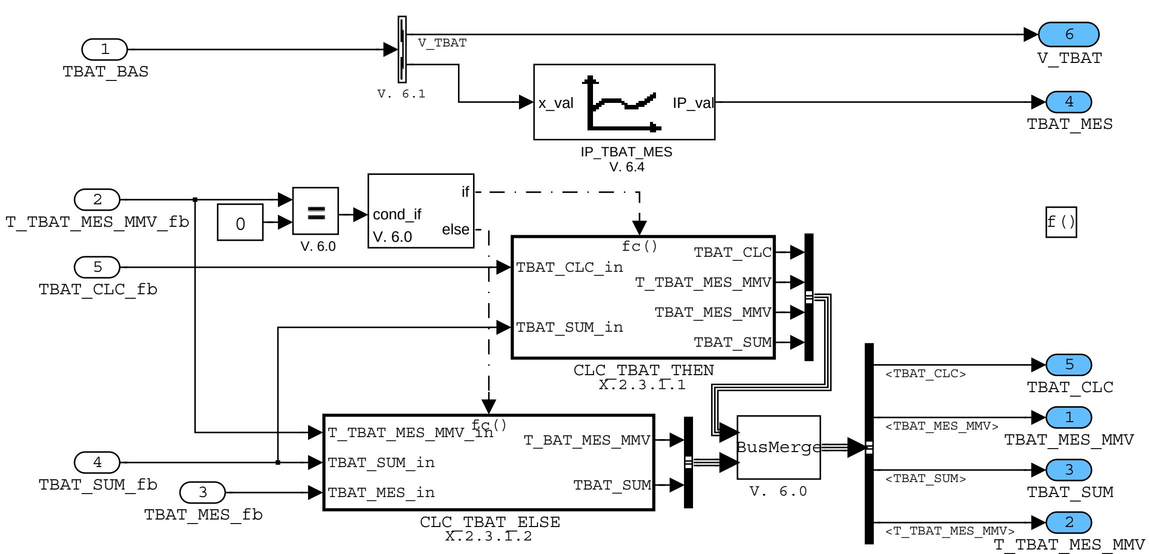


Figure 30:

2.37.2.3.1.1 Calculation of measured battery temperature with respect to Sample number value for TBAT_MES_MMV

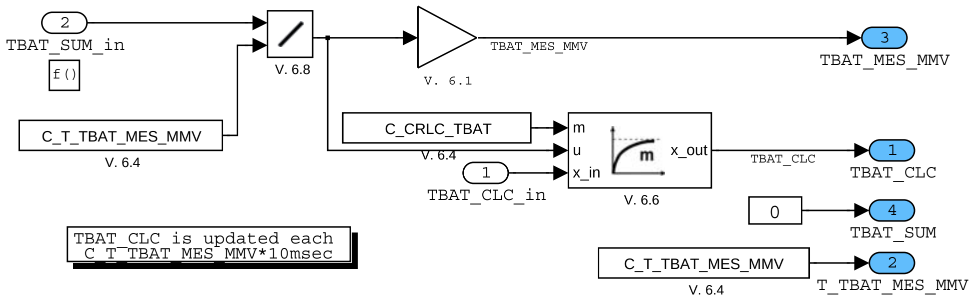


Figure 31:

2.37.2.3.1.2 Condition false

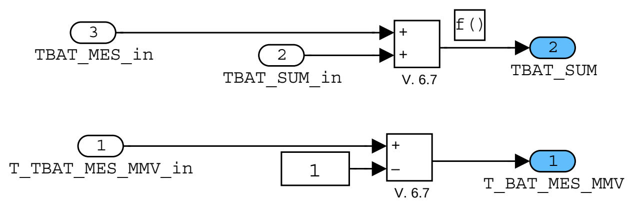


Figure 32:

2.37.2.3.2 Analog battery sensor is enable

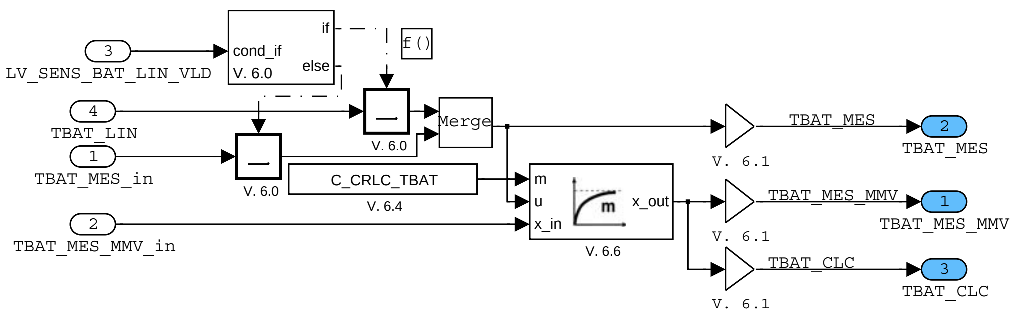


Figure 33:

2.37.2.4 Calculation of battery high resolution voltage

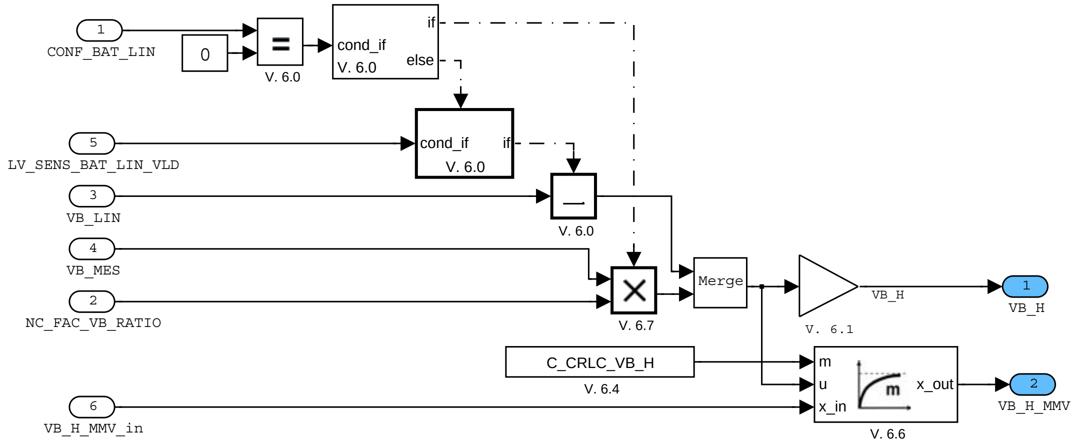


Figure 34:

3 AGGR adaptation modules

Table of Contents

Index

3.1 AGGR adaptation: INJR

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TFU	V/O	0 ... FEH	-48...142.5	0.75	°C
Fuel temperature					
LV_ST_INJ_AUTH	O	0 ... 1H	0...1	1	-
LV_AFL_CLC	O	0 ... 1H	0...1	1	-
LV_IND_FCUT	O	0 ... 1H	0...1	1	-
TI_ADD_x	O	8000...7FFFH	-131.04...131.1	0.004	ms
Total additive correction from engine roughness					
TI_FAC_x	O/V	0...FFFFH	0...1.999969	0.00003	-
Total multiplicative correction value for cylinder balancing					

Input data:

LV_FCUT_IND	AMP	LAMB_SP_1	LV_ST_END
TI_IS_MMV			

FUNCTION DESCRIPTION:

General information:

This chapter describes the in- and output changes in reason of the existing project environment and follower AGGR’es.
Additionally the initialisation of AGGR’es input value what is not described in the AGGR structure will be set.

Formula section:

LV_ST_INJ_AUTH = 1

LV_AFL_CLC = 0

TFU = 20,3

TI_ADD_x = 0

LV_IND_FCUT = LV_FCUT_IND

Recurrence: segment synchronous
Initialisation: At reset etc

TI_FAC_x = 1

Formula Section:

IF LV_ST_END = 1

THEN TI_FAC_x = IP_TI_FAC + TI_IS_MMV

ELSE TI_FAC_x = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TI_FAC	6 * 6	0 ... FFFF H	0 ... 1.999969	0.00003	-
LDPM_AMP_1_INJR	6	0 ... FFFF H	0 ... 5434	0.083	hPa
LDPM_LAMB_SP_1_INJR	6	0 ... 7FFF H	0 ... 31.99	9.7656E-4	-
Map enabling the correction of injection time versus AMP and LAMB_SP					

3.2 EVAM adaptation module

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CL_MMV_CLC_END	V/O	0...FFFFH	0...8	0.000122	-
Canister load at the end of MAX_PURGE operation					
MFF_ADD_CP_KGH	V/O	0...FFFFH	0...8	0.000122	kg/h
Additive adaptive fuel flow from the ACF for the injection fuel mass correction					
FLOW_DLY_MMV_CP	V/O	0...FFFFH	0...8	0.000122	kg/h
Lowpass filter: filtered value of FLOW_DLY_CP					
LV_CP_SET_CLOSE	V/O	0...1H	0...1	1	-
Dummy value, lambda sensor close flag not used					
LV_DIAGCP_ACT_EXT_ADJ	V/O	0...1H	0...1	1	-
Dummy value, no external tool					
LV_VB_CDN_OBD_1	V/O	0...1H	0...1	1	-
Flag previously not available, set to LV_CDN_VB_OBD1					
FLOW_SP_CPS_EVAP	V/O	0...FFFFH	0...8	0.000122	Kg/hr
Flow setpoint for FLOW_CPS at evacuation for CL_gradient check					
LV_ACT_CP	V/O	0...1H	0...1	1	-
Flag indicating CP not active and DIAGCP active (1) or inactive (0)					

Input data:

C_FUEL_FAC_CP	CL_MMV_NORM_PURGE_END	LV_CDN_VB_OBD1	MAF_CPS_DLY_2_MMV
C_CONF_DIAGCP	FUEL_FLOW_ADD_AD_CP	LV_DIAGCPS_ACT	FLOW_SP_CPS_VAP_CHK

FUNCTION DESCRIPTION:

General information:

This chapter describes the in- and output changes in order to adapt EVAM solution in HMC theta project.

Description:

Application conditions (general):

Initialisation: at ECU reset: *All variables set to 0h*

Activation: in all engine operating states

Deactivation: -

Recurrence: 10 ms

Formula section :

MFF_ADD_CP_KGH = FUEL_FLOW_ADD_AD_CP

CL_MMV_CLC_END = CL_MMV_NORM_PURGE_END

FLOW_DLY_MMV_CP = MAF_CPS_DLY_2_MMV

C_RAF_CLC_CP = C_FUEL_FAC_CP

LV_VB_CDN_OBD_1 = LV_CDN_VB_OBD1

LV_ACT_CP = LV_DIAGCPS_ACT

FLOW_SP_CPS_EVAP = FLOW_SP_CPS_VAP_CHK

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_RAF_CLC_CP	1	0...FFFFH	0...256	0.0039	-
stoiciometric fuel constant: ca. 14.7					

3.2.1 Naming Convention

Several variable names have been changed in this integration. The following list provides an overview of the changes for reference.

The list includes any name changes (inputs, outputs, calibrations).

Old Name	New Name
C_CL_MMV_NORM_DIAGCP	C_CL_MMV_CLC_DIAGCP
C_DIAGCP_DIA_MAX	C_DIAM_MAX_DIAGCP
C_DIAGCP_DIA_MAX_VAP_CHK	C_DIAM_MAX_DIAGCP_VAP_CHK
C_DIAGCP_DIA_MIN	C_DIAM_MIN_DIAGCP
C_DIAGCP_DIA_MIN_VAP_CHK	C_DIAM_MIN_DIAGCP_VAP_CHK
C_DIAGCP_MDL_BAS_MAX	C_FAC_DIAGCP_MDL_BAS_MAX
C_DIAGCP_MDL_BAS_MAX_VAP_CHK	C_FAC_DIAGCP_MAX_VAP_CHK
C_DTP_EVAC_MMV_CRLC	C_DTP_MMV_CRLC
C_DTP_EVAC_SLOP_MMV_CRLC	C_DTP_SLOP_MMV_CRLC
C_FUEL_FAC_CP	C_RAF_CLC_CP
C_HYS_NEG_DTP_DIAGCP	C_DTP_HYS_NEG
C_HYS_POS_DTP_DIAGCP	C_DTP_HYS_POS
C_LAM_COR_MAX_VAP_CHK	C_FAC_LAM_COR_MAX_VAP_CHK
C_LAM_COR_MIN_VAP_CHK	C_FAC_LAM_COR_MIN_VAP_CHK
C_LAM_MAX_DIAGCP	C_FAC_LAM_MAX_DIAGCP
C_LAM_MAX_VAP_CHK	C_FAC_LAM_MAX_VAP_CHK
C_LAM_MIN_DIAGCP	C_FAC_LAM_MIN_DIAGCP
C_LAM_MIN_VAP_CHK	C_FAC_LAM_MIN_VAP_CHK
C_MAF_CPS_MAX_FUC_MISS_DIAG	C_FLOW_CPS_MAX_FUC_MISS_DIAG
C_MAF_CPS_MIN_FUC_MISS_DIAG	C_FLOW_CPS_MIN_FUC_MISS_DIAG
C_MAF_CPS_SP_CLOSE_DIAGCP	C_FLOW_CPS_SP_CLOSE_DIAGCP
C_MAX_DIF_SLOP_MMV	C_DTP_MAX_DIF_SLOP_MMV
C_MAX_DTP_REC_DIAGCP	C_DTP_MAX_REC_DIAGCP
C_MAX_PRS_DYN_DIAGCP	C_PRS_MAX_DYN_DIAGCP
C_MIN_PRS_DYN_DIAGCP	C_PRS_MIN_DYN_DIAGCP
C_REL_FLOW_CPS_MIN_CP_DIAGCP	C_FLOW_CPS_MIN_CP_DIAGCP
Old Name	**New Name**
C_SUM_CYC_DIAGCP_MAX	C_NR_CYC_DIAGCP_MAX
C_SUM_CYC_FUC_MISS_DIAG_MAX	C_NR_CYC_FUC_MISS_DIAG_MAX
C_SUM_CYC_INTR_DIAGCP_MAX	C_NR_CYC_INTR_DIAGCP_MAX
C_SUM_CYC_INTR_FUC_MISS_DIAG	C_NR_CYC_INTR_FUC_MISS_DIAG

C_T_AST_INIT_DIAGCP	C_T_AST_INI_DIAGCP
C_T_AST_MIN_DIAGCP_05MM	C_T_AST_MIN_DIAGCP_1
C_T_INH_DIAGCP_AMP_GRD	C_T_INH_AMP_GRD_DIAGCP
C_T_NORM_PURGE_END_MAX	C_T_CLC_PURGE_END_MAX
C_T_VS_MIN_INIT_DIAGCP	C_T_VS_MIN_INI_DIAGCP
C_T_VS_MIN_TAM_FUEL_TEMP	C_T_VS_MIN_TAM_COR_TFU
C_TCO_ST_MAX_DIAGCP_05MM	C_TCO_ST_MAX_DIAGCP_1
C_TCO_ST_MIN_DIAGCP_05MM	C_TCO_ST_MIN_DIAGCP_1
C_TIA_MAX_DIAGCP	C_TAM_MAX_DIAGCP
C_TIA_MIN_DIAGCP	C_TIA_MIN_DIAGCP
C_VS_MIN_INIT_DIAGCP	C_VS_MIN_INI_DIAGCP
CDN_DIAG_SOV	STATE_CDN_DIAG_SOV
CL_MMV_NORM_PURGE_END	CL_MMV_CLC_END
CP_STATE	STATE_CP
DIAGCP_AR	AR_DIAGCP
DIAGCP_DIA	FAC_DIAM_DIAGCP
DIAGCP_DIA_MOD_6	FAC_DIAM_DIAGCP_MOD_6
DIAGCP_MDL_BAS	FAC_DIAGCP_MDL_BAS
DIAGCP_MDL_DIAG	FAC_DIAGCP_MDL_DIAG
DIAGCP_MDL_END	FAC_DIAGCP_MDL_END
DIAGCP_MDL_ESTIM	FAC_DIAGCP_MDL_ESTIM
DIAGCP_VOL_COR	FTV_COR_DIAGCP
DTP_EVAC_BEG_DIAGCP	DTP_BEG_DIAGCP
DTP_EVAC_MMV	DTP_MMV
DTP_EVAC_SLOP	DTP_SLOP
DTP_EVAC_SLOP_MMV	DTP_SLOP_MMV
DTP_EVAC_SLOP_MMV_DIF	DTP_SLOP_MMV_DIF
DTP_EVAC_SLOP_MMV_MAX	DTP_SLOP_MMV_MAX
DTP_EVAC_SLOP_MMV_MIN	DTP_SLOP_MMV_MIN
ERR_SYM_LEAK_LARGE	ERR_SYM_VAP_LEAK_10
ERR_SYM_LEAK_SMALL_05MM	ERR_SYM_VAP_LEAK_1
ERR_SYM_LEAK_SMALL_1MM	ERR_SYM_VAP_LEAK_2
FUEL_FLOW_ADD_AD_DLY	MFF_ADD_CP_KGH
FUEL_FLOW_CP_VAP_CHK	MFF_VAP_CHK_DIAGCP
ID_T_AST_MIN_DIAGCP__TCO_ST	ID_T_AST_MIN_DIAGCP
ID_T_AST_MIN_DIAGCP_2__TCO_ST	ID_T_AST_MIN_DIAGCP_2
INTR_STATE_DIAGCP	STATE_INTR_DIAGCP
IP_DIAGCP_MDL_DTP	IP_DTP_MDL_DIAGCP
IP_DIAGCP_MDL_ESTIM	IP_FAC_DIAGCP_MDL_ESTIM
IP_DIAGCP_VOL_COR__CAN_FTL	IP_FTV_COR_DIAGCP
IP_FAC_SQRT_DIAGCP_AR	IP_FAC_SQRT_AR_DIAGCP
LAM_0_CP_VAP_CHK	FAC_LAM_0_CP_VAP_CHK
LAM_DIF_CP_VAP_CHK	FAC_LAM_DIF_CP_VAP_CHK
LAM_MV_DIAGCP	FAC_LAM_DIAGCP
LC_INH_VAP_CHK	LC_VAP_CHK_INH
LC_INH_VAP_CHK	LC_VAP_CHK_INH
LDP_AMP	LDP_AMP_ID_AMP_TEMP_FUEL
LDP_AMP__SQRT_AMP	LDP_AMP_IP_AMP_SQRT

Old Name	New Name
LDP_DIAGCP_AR_IP_FAC_SQRT	LDP_AR_DIAGCP_IP_FAC_SQRT
LDP_DIAGCP_MDL_BAS_IP_DIAGCP	LDP_FAC_DIAGCP_MDL_BAS_IP_DIAGCP
LDP_DTP__DIAGCP_MDL_DTP	LDP_DTP_IP_DTP_MDL_DIAGCP
LDP_TAM_FUEL_TEMP_IP_T_AST_COR	LDP_TAM_COR_TFU_IP_T_AST_COR

LDP_TAM_FUEL_TEMP_IP_TEMP_FUEL	LDP_TAM_COR_TFU_IP_TEMP_FUEL
LDP_TCO_ST__T_AST_MIN_DIAGCP	LDP_TCO_ST_IP_T_AST_MIN_DIAGCP
LDP_TCO_ST__T_AST_MIN_DIAGCP_2	LDP_TCO_ST_IP_T_AST_MIN_DIAGCP_2
LDPM_VS_ID_CRLC_TEMP_FUEL	LDPM_VS_EVAM_1
LV_ACT_DIAGCP	LV_DIAGCP_ACT
LV_ACT_DIAGCPS	LV_DIAGCPS_ACT
LV_CDN_DIAG_LEAK_LARGE	LV_CDN_DIAG_VAP_LEAK_10
LV_CDN_DIAG_LEAK_SMALL_05MM	LV_CDN_DIAG_VAP_LEAK_1
LV_CDN_DIAG_LEAK_SMALL_1MM	LV_CDN_DIAG_VAP_LEAK_2
LV_CDN_DIAGCP	LV_STATE_CDN_DIAGCP
LV_CDN_FUC_MISS_DIAG	LV_FUC_MISS_DIAG_CDN
LV_CDN_VB_OBD1	LV_VB_CDN_OBD_1
LV_CHK_MPL_DIAGCP	LV_DIAGCP_CHK_MPL
LV_CHK_MPL_FUC_MISS_DIAG	LV_DIAG_CHK_MPL_FUC_MISS
LV_DTP_EVAC_SLOP	LV_DTP_SLOP
LV_END_DIAG_LEAK_LARGE	LV_END_DIAG_VAP_LEAK_10
LV_END_DIAG_LEAK_SMALL_05MM	LV_END_DIAG_VAP_LEAK_1
LV_END_DIAG_LEAK_SMALL_1MM	LV_END_DIAG_VAP_LEAK_2
LV_END_VAP_CHK	LV_VAP_CHK_END
LV_ERR_LEAK_LARGE	LV_ERR_VAP_LEAK_10
LV_ERR_LEAK_SMALL_05MM	LV_ERR_VAP_LEAK_1
LV_ERR_LEAK_SMALL_1MM	LV_ERR_VAP_LEAK_2
LV_INH_DIAGCP_AMP_GRD	LV_AMP_GRD_DIAGCP_INH
LV_INH_DIAGCP_T_ERU	LV_INH_T_ERU_DIAGCP
LV_INH_DIAGCP_T_ERU_OLD	LV_INH_T_ERU_DIAGCP_OLD
LV_LIM_DYN_PRS_DIAGCP	LV_PRS_LIM_DYN_DIAGCP
LV_MDL_DIAGCP	LV_FAC_DIAGCP_MDL
LV_OFF_MTC_MON	LV_MTC_MON_OFF
LV_RST_NORM_PURGE_END	LV_STATE_RST_CLC_END
MAF_CPS	FLOW_CPS
MAF_CPS_DLY_2_MMV	MAF_DLY_2_MMV_CP
MAF_DLY_MMV	MAF_TOT_CP_DLY_MMV
REL_FLOW_CPS	FLOW_CPS
SUM_CYC_DIAGCP	NR_CYC_DIAGCP
SUM_CYC_FUC_MISS_DIAG	NR_CYC_FUC_MISS_DIAG
SUM_CYC_INTR_DIAGCP	NR_CYC_INTR_DIAGCP
SUM_CYC_INTR_FUC_MISS_DIAG	NR_CYC_INTR_FUC_MISS_DIAG
T_INH_DIAGCP_AMP_GRD	T_INH_AMP_GRD_DIAGCP
T_NORM_PURGE_END	T_CLC_END
T_VS_MIN_INIT_DIAGCP	T_VS_MIN_INI_DIAGCP
T_VS_MIN_TAM_FUEL_TEMP	T_VS_MIN_TAM_COR_TFU
TAM_FUEL_TEMP	TAM_COR_TFU
TIA	TAM
TOT_DIF_SIG_BAS	DTP_TOT_DIF_BAS
TOT_DIF_SIG_DIAG	DTP_TOT_DIF_DIAG
TOT_SUM_DTP	DTP_TOT_SUM

3.3 AGGR adaptation: MISF

3.3.1 Rough road acquisition adaptation (contents of B002)

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
T_SEG_RR_0	O/V	0H...FFFFH	0...262140	4	µs
Actual segment time measurement from the wheel speed signal.					
T_SEG_RR_1	O/V	0H...FFFFH	0...262140	4	µs
Previous segment time measurement from the wheel speed signal.					

Input data:

BIOS_VS_EDGE_CTR	BIOS_VS_EDGE_T		

Application conditions:

Initialisation : all other outputs and local variables are set to 0 at ECU-reset and at function deactivation

Recurrence : 10 msec of rough road event task

Activation : After VS signal learning is finished

Formula section:

Segment period acquisition

The function uses consecutive segment times for calculation of a wheel speed gradient. The segments are built with a minimum number of NC_SEG_TOOTH_RR edges and afterwards normalized to a length of 4 teeth.

If BIOS_VS_EDGE_CTR - TMP_BIOS_VS_EDGE_CTR_OLD >= NC_SEG_TOOTH_RR
then *// If condition reversed, Then and else position interchanged*

(calculation of T_SEG_RR_0)

T_SEG_RR_1=T_SEG_RR_0

$$T\_SEG\_RR\_0 = \frac{(BIOS\_VS\_EDGE\_T - TMP\_BIOS\_VS\_EDGE\_T\_OLD)*4}{BIOS\_VS\_EDGE\_CTR - TMP\_BIOS\_VS\_EDGE\_CTR\_OLD}$$

TMP_BIOS_VS_EDGE_T_OLD=BIOS_VS_EDGE_T

TMP_BIOS_VS_EDGE_CTR_OLD=BIOS_VS_EDGE_CTR

else (no seg.period acquisition)

No Action

T_SEG_RR_1 is the segment period preceding T_SEG_RR_0. TMP_BIOS_VS_EDGE_T_OLD & TMP_BIOS_VS_EDGE_CTR_OLD are temporary local variables .

Configuration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
NC_SEG_TOOTH_RR	1	1H...FFH	0...255	1	-
Number of wheel speed signal teeth to build one segment , typical value = 4					

3.3.2 Configuration Data of AGGR CBMD, exported to other aggr.

Configuration data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
NC_CMB_CONF	1	0H 1H 2H 3H	AFS AFS_S AFS_AFL AFS_AFL_S	1	[-]
Engine combustion modes target					

Description:

Name	Value
NC_CMB_CONF	0H (AFS)

3.4 Transfer module for aggregate EGCP

3.4.1 EGCP Outputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_LSH_DOWN_1	V/O	0H 1H 2H 3H 4H 5H 6H	LSH_OFF LSH_POW_RISE LSH_POW_RED LSH_POW_FALL LSH_POW_CTL LSH_VB_PROT LSH_TEMP_PROT	1	-
State of the downstream oxygen sensor heater					
VLS_DOWN_1	V/O	0...3FFh	0...4.995	4.89E-3	V
Downstream sensor voltage measured					
LV_ERR_VLS_UP_1	V/O	0...1H	0...1	1	-
Upstream oxygen sensor signal electrical fault present					
LV_ERR_VLS_DOWN_1	V/O	0...1H	0...1	1	-
Downstream oxygen sensor signal electrical fault present					
LV_ERR_LS_UP_FRQ_1	V/O	0...1H	Passive;Active	1	-
status diagnostic result oxygen sensor frequency monitoring					
LV_ERR_LS_UP_SWT_1	V/O	0...1H	Passive;Active	1	-
status diagnostic result of switch-time monitoring					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]	LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	STATE_LSH_DOWN[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

The HMC Theta PI project has one cylinder bank and exhaust system.

The variable assignment must be executed after the EGCP modules execution.

Application conditions:

Initialisation: at ECU reset all variables are initialised with 0

Recurrence: 10ms

Activation: at all engine operating states

Deactivation: -

Formula section:

LV_ERR_LS_UP_FRQ_1 = LV_ERR_FRQ_LS_UP[1]

LV_ERR_LS_UP_SWT_1 = LV_ERR_SWT_LS_UP[1]

STATE_LSH_DOWN_1 = STATE_LSH_DOWN[1]

VLS_DOWN_1 = VLS_DOWN[1]

LV_ERR_VLS_UP_1 = 0

LV_ERR_VLS_DOWN_1 = 0

3.4.2 EGCP Inputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_VB_CDN_OBD_2	V/O	0...1H	0...1	1	-
Boolean for battery voltage condition fulfilled for OBD-II diagnosis					
LV_ERR_MEC_OPEN_CPS	V/O	0...1H	0...1	1	-
Boolean for mechanical CPS error					
LV_ET_ACT	V/O	0…1H	0…1	1	-
Flag indicating emission check active					
LV_CP_CLOSE_ACT	V/O	0...1H	0...1	1	-
Logical value for CPS valve closed					
LV_TEG_MIN_THD[i]	V/O	0...1H	0...1	1	-
Dew Point reached at both banks					
LV_DIAG_EOL_REQ_OBD_LSH_DOWN	V/O	0...1H	0...1	1	-
Flag indicating a request for a LSH_DOWN OBD 2 end of line test.					

Input data:

	LV_CDN_VB_OBD2	LV_CLOSE_ACT_CP	LV_ERR_MEC_CPS
LV_TEMP_DEW_LS_UP[NC_CBK_EX_NR]			

FUNCTION DESCRIPTION:

General information:

The HMC Theta PI project has one cylinder bank and exhaust system.

This function must be executed before the EGCP modules are executed.

Application conditions:

Initialisation: at ECU reset all variables are initialised with 0

Recurrence: *T_SAMPLE = 10 ms*

Activation: *at all engine operating states*

Deactivation: -

Formula section:

LV_ET_ACT = 0

LV_VB_CDN_OBD_2 = LV_CDN_VB_OBD2

LV_ERR_MEC_OPEN_CPS = LV_ERR_MEC_CPS

LV_CP_CLOSE_ACT = LV_CLOSE_ACT_CP

LV_DIAG_EOL_REQ_OBD_LSH_DOWN = 0

LV_TEG_MIN_THD[i] = LV_TEMP_DEW_LS_UP[i]

3.5 Transfer module for aggregate LACO

3.5.1 LACO Outputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LSCL_i	V/O	0...1H	0...1	1	-
lambda controller activation signal					
LAM_MV_i	V/O	8000...7FFFh	-50...49.9985	1.5259E-3	%
mean value of lambda controller output					
LAM_MV_LPF_1	V/O	8000...7FFFh	-50...49.9985	1.5259E-3	%
moving mean value of LAM_MV_1					
LAM_MV_LPF_CP_1	V/O	8000...7FFFh	-50...49.9985	1.5259E-3	%
Mean value of the controller output, used by canister purge					
LV_LAM_LIM_MAX_i	V/O	0...1H	0...1	1	-
Upper threshold of controller output exceeded					
LV_LAM_LIM_MIN_i	V/O	0...1H	0...1	1	-
bottom threshold of controller output exceeded					
LV_MFF_AD_END	V/O	0...1H	0...1	1	-
Logical value signals the temporary end of lambda adaptation					
TI_LAM_COR_i	V/O	8000...7FFFH	-50...49.9985	1.5259E-3	%
Lambda controller injection time value					
LAM_P_CTR	V/O	0...FFH	0...255	1	-
P jump counter (artifical P jump) (CAN overflow)					
T_MFF_AD_MIN	V/O	0...FFFFH	0...6553.5	0.1	s
Time indicating the time for next requested lambda adaptation					
LV_MFF_AD_CDN	V/O	0...1H	0...1	1	-
Boolean for time scheduler indicating good conditions for lambda adaptation					
CAT_DIAG_i	V/O	0...7FH	0...1.984	1.56E-2	-
Catalyst diagnosis value in OSC range					
LV_MFF_AD_FAC_L_LIM_MAX_i	V/O	0...1H	0...1	1	-
upper limitation of multiplicative adapted factor of lower area					
LV_MFF_AD_FAC_L_LIM_MIN_i	V/O	0...1H	0...1	1	-
bottom limitation of multiplicative adapted factor of lower area					
LV_MFF_AD_FAC_H_LIM_MAX_i	V/O	0...1H	0...1	1	-
upper limitation of multiplicative adapted factor of upper area					
LV_MFF_AD_FAC_H_LIM_MIN_i	V/O	0...1H	0...1	1	-
bottom limitation of multiplicative adapted factor of upper area					
LV_MFF_AD_ADD_LIM_MAX_i	V/O	0...1H	0...1	1	-
upper limitation of additive adapted factor					
LV_MFF_AD_ADD_LIM_MIN_i	V/O	0...1H	0...1	1	-
bottom limitation of additive adapted factor					
LV_MFF_AD_FAC_H_i	V/O	0...1H	0...1	1	-
flag indicating that adaptation in upper multiplicative area is active					
LV_MFF_AD_FAC_L_i	V/O	0...1H	0...1	1	-
flag indicating that adaptation in lower multiplicative area is active					
LV_MFF_AD_ADD_i	V/O	0...1H	0...1	1	-
flag indicating that adaptation in additive area is active					

Input data:

LV_LAM_LSCL[NC_CBK_EX_NR]	FAC_LAM_MV[NC_CBK_EX_NR]
FAC_LAM_MV_MMV[NC_CBK_EX_NR]	FAC_LAM_MV_MMV_CP[NC_CBK_EX_NR]
LV_FAC_LAM_LIM_MAX[NC_CBK_EX_NR]	LV_FAC_LAM_LIM_MIN[NC_CBK_EX_NR]
LV_LAM_AD_END	LV_FAC_H_RNG_LIM_MAX_LAM_AD[NC_CBK_EX_NR]
LV_LAM_AD_CDN	LV_FAC_H_RNG_LIM_MIN_LAM_AD[NC_CBK_EX_NR]
EFF_CAT_DIAG_i	LV_MFF_ADD_LIM_MAX_LAM_AD[NC_CBK_EX_NR]

T_PRI_TOT_LAM_AD	LV_MFF_ADD_LIM_MIN_LAM_AD[NC_CBK_EX_NR]
CTR_RAF_CHG[NC_CBK_EX_NR]	LV_FAC_H_RNG_LAM_AD[NC_CBK_EX_NR]
FAC_LAM_LIM[NC_CBK_EX_NR]	LV_FAC_L_RNG_LAM_AD[NC_CBK_EX_NR]
LV_MFF_ADD_RNG_LAM_AD[NC_CBK_EX_NR]	LV_FAC_L_RNG_LIM_MIN_LAM_AD[NC_CBK_EX_NR]
	LV_FAC_L_RNG_LIM_MAX_LAM_AD[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

The variable assignment must be executed after the LACO modules execution.

Application conditions (general):

Initialisation: *at ECU reset:*
T_MFF_AD_MIN = 6553.5 s
CAT_DIAG_1 = EFF_CAT_DIAG[1]

Activation: in all engine operating states

Deactivation: -

Application conditions (1):

Recurrence: T_SAMPLE = 10 ms

Formula section (1):

LV_LSCL_1 = LV_LAM_LSCL[NC_CBK_EX_NR]

LAM_MV_1 = FAC_LAM_MV[NC_CBK_EX_NR]

LAM_MV_LPF_1 = FAC_LAM_MV_MMV[NC_CBK_EX_NR]

LAM_MV_LPF_CP_1 = FAC_LAM_MV_MMV_CP[NC_CBK_EX_NR]

LV_LAM_LIM_MAX_1 = LV_FAC_LAM_LIM_MAX[NC_CBK_EX_NR]

LV_LAM_LIM_MIN_1 = LV_FAC_LAM_LIM_MIN[NC_CBK_EX_NR]

CAT_DIAG_1 = EFF_CAT_DIAG_i *(/* variable definitions are not equal */)*

LAM_P_CTR = CTR_RAF_CHG[NC_CBK_EX_NR]

TI_LAM_COR_i = FAC_LAM_LIM[NC_CBK_EX_NR]

Application conditions (2):

Recurrence: T_SAMPLE = 20 ms

Formula section (2):

LV_MFF_AD_END = LV_LAM_AD_END

T_MFF_AD_MIN = T_PRI_TOT_LAM_AD

LV_MFF_AD_CDN = LV_LAM_AD_CDN

LV_MFF_AD_FAC_L_LIM_MAX_i = LV_FAC_L_RNG_LIM_MAX_LAM_AD[i]

LV_MFF_AD_FAC_L_LIM_MIN_i = LV_FAC_L_RNG_LIM_MIN_LAM_AD[i]

LV_MFF_AD_FAC_H_LIM_MAX_i = LV_FAC_H_RNG_LIM_MAX_LAM_AD[i]

LV_MFF_AD_FAC_H_LIM_MIN_i = LV_FAC_H_RNG_LIM_MIN_LAM_AD[i]

LV_MFF_AD_ADD_LIM_MAX_i = LV_MFF_ADD_LIM_MAX_LAM_AD[i]

LV_MFF_AD_ADD_LIM_MIN_i = LV_MFF_ADD_LIM_MIN_LAM_AD[i]

LV_MFF_AD_FAC_H_i = LV_FAC_H_RNG_LAM_AD[i]

LV_MFF_AD_FAC_L_i = LV_FAC_L_RNG_LAM_AD[i]

LV_MFF_AD_ADD_i = LV_MFF_ADD_RNG_LAM_AD[NC_CBK_EX_NR]

3.5.2 LACO Inputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FAC_LAM_SHIFT_CP	V/O	0...1H	0...1	1	-
logical value for lambda control correction during RAMP_OPEN operation					
FAC_LAM_SHIFT_CP	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
correction value for lambda control correction during RAMP_OPEN operation					
LV_CAT_PURGE_ACT[1]	V/O	0...1H	0...1	1	-
Status of catalyst purge activation					
LV_LAM_AD_ENA	V/O	0...FFH	0...1.992188	7.8125E-3	-
Lambda adaptation enabled					
T_DLY_CAT_PURGE[1]	V/O	0...FFFFH	0...1279.980469	1.9531E-2	ms
LAM-P-jump delay time of the catalyst purge function					
CRLC_FAC_LAM_MV_MMV_CP	V/O	0...FFH	0...0.996094	3.9063E-3	-
correlation constant for calculation of filtered lambda controller output mean value for CP					
PV	V/O	0...3FFH	0...99.902344	0.097656	%
Global degree of activation of the accelerator pedal (high resolution)					
LV_LAM_LIM_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Request for forced lambda adaptation					

Input data:

LV_LAM_ADJ_CP	LAM_ADJ_CP
	LV_FAC_LAM_SHIFT_CP_END[NC_CBK_EX_NR]
LV_LAM_LIM_MFF_AD_i	FLOW_SP_CPS_EVAP
PV_AV	
	LV_TI_AD_CP_INH

FUNCTION DESCRIPTION:

General information:

This function must be executed before the LACO modules are executed.

Application conditions (general):

Initialisation: at ECU reset:

CRLC_FAC_LAM_MV_MMV_CP = 1

Activation: in all engine operating states

Deactivation: -

Application conditions:

Recurrence: T_SAMPLE = 10 ms

Formula section:

IF (LV_FAC_LAM_SHIFT_CP_END[NC_CBK_EX_NR] = 1)
THEN

set LV_LAM_ADJ_CP to 0 by using service of EVAC function

ENDIF

LV_FAC_LAM_SHIFT_CP = LV_LAM_ADJ_CP

FAC_LAM_SHIFT_CP = LAM_ADJ_CP

LV_CAT_PURGE_ACT[1] = LV_ACT_INT_PUC_1

LV_LAM_AD_ENA = **NOT** (LV_TI_AD_CP_INH)

T_DLY_CAT_PURGE[1] = 0

CRLC_FAC_LAM_MV_MMV_CP = IP_CRLC_FAC_LAM_MV_MMV_CP

LV_LAM_LIM_LAM_AD[i] = LV_LAM_LIM_MFF_AD_i

PV = PV_AV

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CRLC_FAC_LAM_MV_MMV_CP	6	0...FFH	0...0.996094	3.9063E-3	-
LDP_MAF_CYL_IP_CRLC_FAC_LAM_MMV	6	0...FFFFH	0...2047.96875	0.03125	kg/h
correlation constant for calculation of filtered lambda controller output mean value for CP					

3.6 EXTD aggregate adaptation module

3.6.1 EXTD Outputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TEG_CAT_DOWN_MIN_THD[NC_CBK_EX_NR]	O	0h ...1h	0 ... 1	1	-
Flag indicating that dew point is passed at lambda sensor down catalyst					
TEMP_CAT	O	0 ... FFFFh	-33 ... 990	0.0156	°C
modeled catalyst monolith temperature					
TEG_CAT_UP_MDL[NC_CBK_EX_NR]	O	0...7FF0H	0...2047	0.0625	[°C]
Modelled exhaust gas temperature; engine out (bank selective)					
TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]	O	0...7FF0H	0...2047	0.0625	[°C]
Modelled exhaust gas temperature; engine out (bank selective)					
TEG_CAT_UP_MDL_MAX	O	0...7FF0H	0...2047	0.0625	[°C]
Maximum out of TEG_CAT_UP_MDL					

Input data:

TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	LV_TEMP_DEW_LS_DOWN[NC_CBK_EX_NR]	TEG_DYN_UP_CAT[NC_CBK_EX_NR]	TEG_DYN_DOWN_CAT[NC_CBK_EX_NR]

Formula section:

Recurrence: 100 ms

This variable will be directly mapped on the new name.

LV_TEG_CAT_DOWN_MIN_THD[i] = LV_TEMP_DEW_LS_DOWN[i]

Formula section:

Recurrence: 100 ms

TEMP_CAT = TEMP_CAT_DYN_MDL[1]

TEG_CAT_UP_MDL[1] = TEG_DYN_UP_CAT[1]

TEG_CAT_UP_MDL_MAX = TEG_DYN_UP_CAT[1]

TEG_CAT_DOWN_MDL[1] = TEG_DYN_DOWN_CAT[1]

3.6.2 EXTD Inputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_PUC	V/O	0 ... FFFFh	0 ... 655.35	0.01	s
PUC duration time					

Input data:

LV_PUC			

3.6.2.1 Calculation of T_PUC

Application conditions:

Initialisation: At reset T_PUC = 0

Recurrence: 10ms

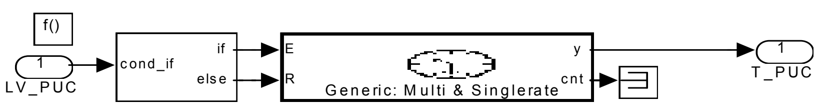
Activation: LV_PUC = 1

Then increment T_PUC

Deactivation: LV_PUC = 0

Then T_PUC = 0

Formula section:



3.7 AGGR adaptation: ERRM

Export actions:

ACTION_ERRM_TrigFarm(IN <XX>)
This action indicates if a failure status has changed to inform FARM

Formula section:

ACTION_ERRM_TrigFarm(XX) do nothing !

4 System variables

Table of Contents

Index

D

G

I

L

M

N

O

P

T

4.1 ECU state

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ECU_STATE	O/V	0H 1H 2H 3H 4H 5H	ENG_STOP RUN_ENG SYN_ENG_IGK_ON SYN_ENG_IGK_OFF PWL ENG_LOCK	-	-
Information on ECU state					

Input data:

LV_IGK	LV_ES	LV_MU_INH_PWL_TRAN_ES_EL	

FUNCTION DESCRIPTION:

General information:

The function ECU-state manages detection of the operating states of the ECU and transitions between them referring to the main inputs key detection and crankshaft synchronization.

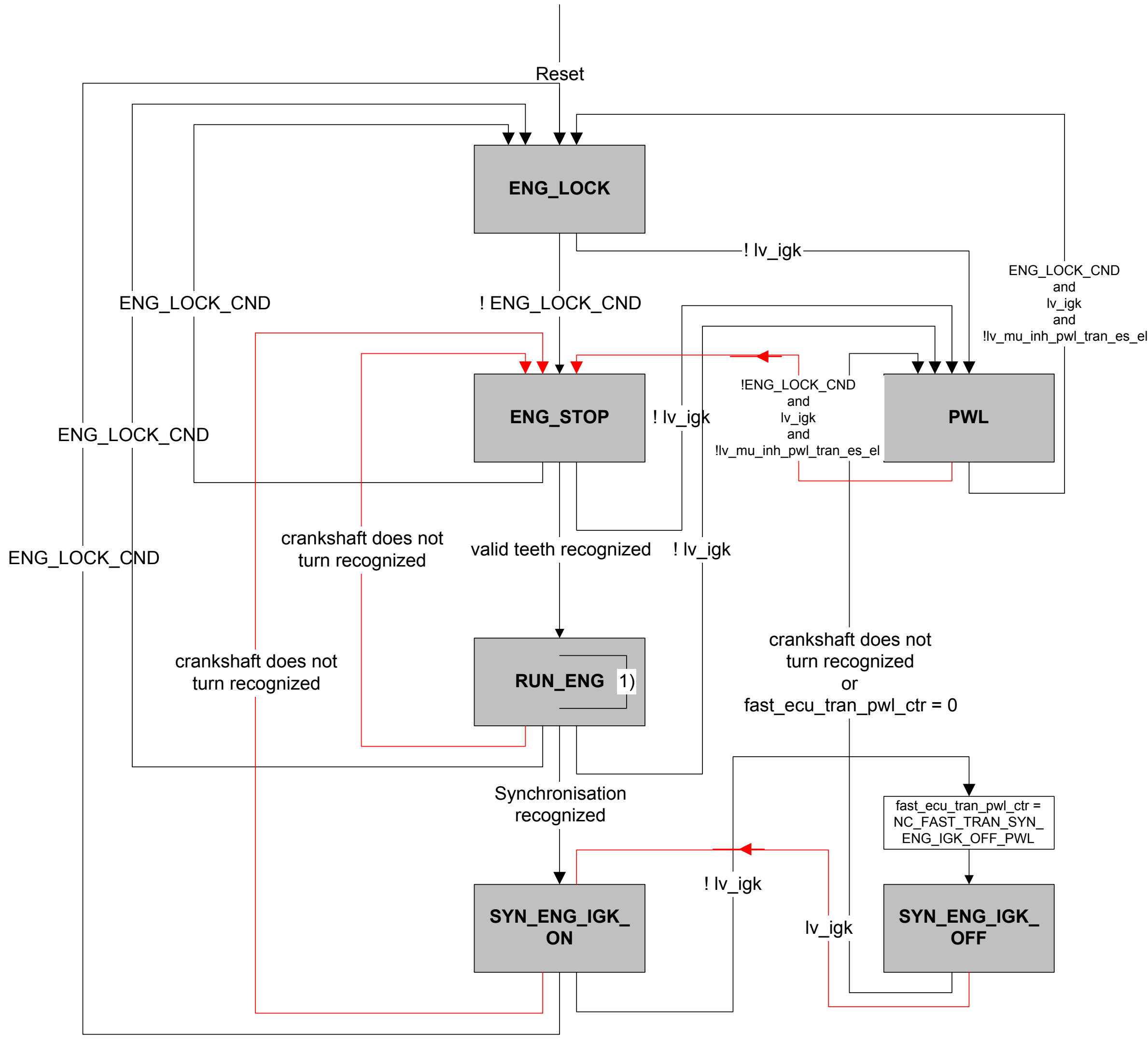
The ECU-state offers system events corresponding to those transitions that can be used by other functions (e. g. ignition function that asks for ignition reset on the transition engine running to engine stop).

Those actions are described in the specification chapter of the corresponding functions.

An important value to determine the ECU-state is the value ENG_LOCK_CDN. Here are stored bit-wise conditions which prevent leaving the ECU-state ENG_LOCK. If anyone of those bits becomes 1, the ECU-state will become ENG_LOCK. These conditions are:

- cross check of software and calibration data reference

ECU-State
Activation: every engine state Deactivation: - Initialisation: ECU_STATE = ENG_LOCK Update rate: 10ms



1) Transition ENG_STOP -> RUN_ENG -> SYN_ENG_IGK_ON in one cycle if Synchronisation recognized (by BSW)

4.1.1 ENGINE LOCK (ENG_LOCK)

This state is dedicated to a special operating state where all essential outputs are inhibited.

An interface for project specific activities on this state (ENG_LOCK) does exist.

The ECU state is initialized with this state after reset.

4.1.1.1 ENGINE LOCK => POWERLATCH (PWL)

IF LV_IGK = 0
THEN offer an interface for project specific activities on this state transition
ECU_STATE = PWL
ENDIF

4.1.1.2 ENGINE LOCK => ENGINE STOP (ENG_STOP)

IF LV_IGK **AND** (ENG_LOCK_CDN = 0)
THEN offer an interface for project specific activities on this
state transition,
ECU_STATE = ENG_STOP
ENDIF

4.1.2 ENGINE STOP

The crankshaft does not turn

An interface for project specific activities on this state (ENG_STOP) does exist.

4.1.2.1 ENGINE STOP => POWERLATCH

IF LV_IGK = 0
THEN offer an interface for project specific activities
ECU_STATE = PWL
ENDIF

4.1.2.2 ENGINE STOP => ENGINE LOCK

IF (ENG_LOCK_CDN ≠ 0) **AND** (LV_IGK)
THEN offer an interface for project specific activities,
ECU_STATE = ENG_LOCK
ENDIF

4.1.2.3 ENGINE STOP => ENGINE STOP

IF LV_ES **AND** (LV_IGK) **AND** (ENG_LOCK_CDN = 0)

THEN offer an interface for project specific activities

ENDIF

4.1.2.4 ENGINE STOP => RUNNING ENGINE (RUN_ENG)

IF (STATE_CRK_SYNC < BIOS_CRK_VLD_TEETH)
AND (LV_IGK) **AND** (ENG_LOCK_CDN = 0)

THEN offer an interface for project specific activities

ECU_STATE = RUN_ENG

ENDIF

4.1.2.5 ENGINE STOP => SYNCHRONOUS ENGINE IGNITION KEY ON (SYN_ENG_IGK_ON)

IF LV_SYN_ENG **AND** LV_RUN_ENG

THEN offer an interface for project specific activities for transition of state RUN_ENG,

offer a second interface for project specific activities on the state transition in state SYN_ENG_IGK_ON.

ECU_STATE = SYN_ENG_IGK_ON

ENDIF

4.1.3 RUNNING ENGINE (RUN_ENG)

This state corresponds to a turning engine without synchronization on crankshaft.

An interface for project specific activities on this state (RUN_ENG) does exist.

4.1.3.1 RUNNING ENGINE => POWERLATCH

IF LV_IGK = 0

THEN offer an interface for project specific activities on this state transition.

ECU_STATE = PWL

ENDIF

4.1.3.2 RUNNING ENGINE => ENGINE LOCK

IF ENG_LOCK_CDN ≠ 0

THEN offer an interface for projekt specific activities,

ECU_STATE = ENG_LOCK

ENDIF

4.1.3.3 RUNNING ENGINE => ENGINE STOP

IF STATE_CRK_SYNC = BIOS_EPO_CRK_PAS

THEN offer an interface for project specific activities,

ECU_STATE = ENG_STOP

ENDIF

4.1.3.4 RUNNING ENGINE => SYNCHRON ENGINE IGNITION KEY ON

IF STATE_CRK_SYNC = BIOS_EPO_CRK_SYNC

THEN offer an interface for project specific activities

ECU_STATE = SYN_ENG_IGK_OFF

ENDIF

4.1.4 SYNCHRON ENGINE: IGNITION KEY ON (SYN_ENG_IGK_ON)

This state corresponds to a rotating crankshaft after having synchronized on camshaft combined with ignition key on.

4.1.4.1 SYN_ENG_IGK_ON => ENGINE LOCK

IF ENG_LOCK_CDN ≠ 0

THEN offer an interface for project specific activities

ECU_STATE = ENG_LOCK

ENDIF

4.1.4.2 SYN_ENG_IGK_ON => SYN_ENG_IGK_OFF

IF (LV_IGK = 0) **AND** (ENG_LOCK_CDN = 0)

THEN offer an interface for project specific activities,

FAST_ECU_TRAN_PWL_CTR=
NC_FAST_TRAN_SYN_ENG_IGK_OFF_PWL

ECU_STATE = SYN_ENG_IGK_OFF

ENDIF

4.1.4.3 SYN_ENG_IGK_ON => ENGINE STOP

IF (STATE_CRK_SYNC = BIOS_EPO_CRK_PAS) **AND** (ENG_LOCK_CDN = 0)

THEN offer an interface for project specific activities

ECU_STATE = ENG_STOP

ENDIF

4.1.5 SYNCHRON ENGINE: IGNITION KEY OFF (SYN_ENG_IGK_OFF)

This state corresponds to a turning crankshaft after having synchronized on crankshaft combined with ignition key off.

4.1.5.1 SYN_ENG_IGK_OFF => SYN_ENG_IGK_ON

IF LV_IGK = 1
THEN offer an interface for project specific activities
ECU_STATE = SYN_ENG_IGK_ON
ENDIF

4.1.5.2 SYN_ENG_IGK_OFF => POWERLATCH

IF (LV_IGK = 0)
THEN FAST_ECU_TRAN_PWL_CTR = FAST_ECU_TRAN_PWL_CTR - 1
IF LV_STOP_ENG **OR** (FAST_ECU_TRAN_PWL_CTR = 0)
THEN offer an interface for project specific activities
ECU_STATE = PWL
ENDIF
ENDIF

4.1.6 POWER LATCH

Engine stalled, data stored, closedown ahead

An interface for project specific activities on this state (POWER LATCH) does exist

4.1.6.1 PWL => ENGINE LOCK

IF LV_IGK **AND** (LV_MU_INH_PWL_TRAN_ES_EL=0) **AND**
(ENG_LOCK_CDN ≠ 0)
THEN ECU_STATE = ENG_LOCK
ENDIF

4.1.6.2 PWL => ENGINE STOP

IF LV_IGK **AND** (LV_MU_INH_PWL_TRAN_ES_EL=0) **AND**
(ENG_LOCK_CDN = 0)
THEN ECU_STATE = ENG_STOP
ENDIF

4.1.6.3 PWL

IF (LV_ IGK=0) **OR** (LV_MU_INH_PWL_TRAN_ES_EL = 1)
THEN The condition lv_mu_inh_pwl_tran_es_el = 1 indicates that the monitoring unit disabled the powerstages of MTC and injection outputs. IGK ON recognition is not accepted without reset.

ENDIF

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_FAST_TRAN_SYN_ENG_IGK_OFF_PWL	1	0 ... FFFFH	0 ... 655.35	0.01	s
Maximum delay for transition from SYN_ENG_IGK_OFF to PWL despite engine running					

4.2 Engine speed setpoint calculation

4.2.1 Nominal idle speed N_SP_IS

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_SP_IS	V / O	0...1FE0H	0...8160	1	rpm
Nominal idle speed					
N_SP_IS_1	V	0...1FE0H	0...8160	1	rpm
intermediate value : basic value versus TCO_ST, TCO, ACIN, Gearbox					
N_SP_IS_2	V	0...1FE0H	0...8160	1	rpm
intermediate value : basic value + catalyst heating or high transmission oil temperature offset					
N_SP_IS_3	V	0...1FE0H	0...8160	1	Rpm
intermediate value : minimum engine speed in case of low battery voltage					
N_SP_IS_4	V	0...1FE0H	0...8160	1	Rpm
intermediate value : minimum engine speed in case of vehicule rolling					
N_SP_IS_5	V	0...1FE0H	0...8160	1	Rpm
intermediate value : minimum engine speed at brake switch off					
N_SP_IS_6	V	0...1FE0H	0...8160	1	Rpm
intermediate value : Power compensation with cooling fan high active					
N_SP_IS_7	V	0...1FE0H	0...8160	1	rpm
intermediate value : high power steering pressure					
N_SP_IS_8	V	0...1FE0H	0...8160	1	rpm
intermediate value : high oil temperature					
N_SP_IS_9	V	0...1FE0H	0...8160	1	rpm
intermediate value : speed increase request from TCU					
N_SP_IS_GEN_LOAD	V/O	0...1FE0H	0...8160	1	Rpm
Nominal idle speed to compensate electric load signal from alternator					
T_DLY_INC_N_SP_IS_GEN_LOAD	V	0..FFFFH	0...655,35	0.01	Sec
Delay time to increase N_SP_IS_GEN_LOAD in case of LV_DRI = 0					
T_DLY_INC_N_SP_IS_GEN_LOAD_DRI	V	0..FFFFH	0...655,35	0.01	Sec
Delay time to increase N_SP_IS_GEN_LOAD in case of LV_DRI = 1					
T_DLY_DEC_N_SP_IS_GEN_LOAD	V	0..FFFFH	0...655,35	0.01	Sec
Delay time to decrease N_SP_IS_GEN_LOAD					
T_DLY_N_SP_IS_GEN_LOAD	V	0..FFFFH	0...655,35	0.01	Sec
Delay time to enable GEN_LOAD compensation function					
CTR_N_SP_IS_GEN_LOAD_INI	-	0..FFH	0...255	1	-
Counter to make initialize N_SP_IS_GEN_LOAD					
LV_ERR_LIH_SCG_IVVT_IN	V / O	0...1H	0...1	1	-
Boolean for limp home of IVVT short circuit to ground error (Inlet)					
LV_ERR_LIH_SCG_IVVT_EX	V / O	0...1H	0...1	1	-
Boolean for limp home of IVVT short circuit to ground error (Exhaust)					
N_SP_IS_SLV_IVVT_SCG	V	0...1FE0H	0...8160	1	rpm
intermediate value : wtih IVVT SCG error condition					
N_SP_IS_DIF	V	8000...7FFFH	-32768...32767	1	rpm
Nominal Idle Speed difference					
N_SP_IS_EOL	V / O	0...1FE0H	0...8160	1	rpm
intermediate value : speed increase request during EOL EVAP & IVVT					
N_SP_IS_OFS_AMP	V	8000...7FFFH	-32768...32767	1	rpm
Engine speed setpoint offset due to atmospheric pressure					

Input data:

TCO	LV_DLY_N_SP_IS	LV_ACIN	LV_CH
T_AST	VB_MMV	LV_DRI	VS
LV_ST	GEN_LOAD_MMV_FIL	LV_ERR_MWSS	LV_AT
LV_ERR_GEN_LOAD	TCO_ST	AMP	VB
LV_ES	TOIL_GB	LV_PUC	ERR_SYM_SLV_IVVT_IN(_EX)_1

N_SP_IS_ADJ_ASA	CONF_GEN_LOAD	LV_DRI	CTR_ABC_SLV_IVVT_IN(_EX)_1
LV_AT	NC_ETC_CONF	LV_BLS	LV_BRAKE_OFF
STATE_CFA	LV_ERR_PSP	PSP	LV_ERR_PSP_SWI_PLAUS
TOIL	LV_PSTE	LV_ERR_PSP_PLAUS	LV_N_INC_TCU_ACT
N_SP_IS_TCU	EOL_STATE_DIAGCP	LV_TRIP_ACT_IVVT_IN	LV_TRIP_ACT_IVVT_EX

FUNCTION DESCRIPTION:

General information:

The nominal idle speed in the engine operating state Idle (LV_IS) depends on the additional loads, coolant temperature TCO and cranking coolant temperature TCO_ST. It is initialized with the table corresponding to the conditions at the initialization (additional loads).

The input value LV_DLY_N_SP_IS (output from module Converter Torque) is derived from LV_DRI including a tuneable delay time. If drive is engaged the converter torque is build up after a short delay time (about 400 ms) if the change in engine speed setpoint for drive engaged also is delayed the new setpoint can be adjusted more stable.

Following corrections can be done:
- Nominal Idle speed increase for fast Catalyst Heating : If Catalyst Heating function is active, nominal idle speed offset depends on coolant temperatue (TCO), ambient pressure (AMP), time after Start (T_AST), and gearbox (A/T, M/T)
- Nominal idle speed increase in case of vehicle rolling (VS > 0) : separate correction for A/T and M/T
- Nominal idle speed increase in case of high transmission oil temperature for A/T vehicle
- Nominal Idle speed increase in case of low Battery Voltage : correction in Drive for low Vehicle speed or for M/T vehicle (always active)

In general the transition from a nominal idle speed to another is performed with an adjustable change limitation IP_N_SP_LGRD.

Exception: no change limitation is applied during catalyst heating activation – see below

Application conditions:

Activation: N <= C_N_MAX_IDLE_CALC

Deactivation: N > C_N_MAX_IDLE_CALC

$N\_SP\_IS_{k} = N\_SP\_IS_{k-1}$

Initialization: N_SP_IS = N_SP_IS_1 = N_SP_IS_2 = IP_N_SP_IS_AT or IP_N_SP_IS_MT depending on LV_AT

N_SP_IS_3 = N_SP_IS_4 = N_SP_IS_5 = N_SP_IS_6 = 0

N_SP_IS_7 = N_SP_IS_8 = = N_SP_IS_9 = N_SP_IS_EOL = 0 at reset

Recurrence: 10 ms

Basic values N_SP_IS_1

If LV_ACIN = 1 *(ACC active)*

then **if** LV_AT = 0

then **if** VS = 0

then N_SP_IS_1 = IP_ACIN_N_SP_IS_MT__TCO__TCO_ST

```
                else    N_SP_IS_1 = IP_ACIN_VS_N_SP_IS_MT__TCO__TCO_ST
        else    if      LV_DLY_N_SP_IS = 1                              /     *)
                then    N_SP_IS_1 = IP_DRI_ACIN_N_SP_IS__TCO__TCO_ST
                else    N_SP_IS_1 = IP_ACIN_N_SP_IS_AT__TCO__TCO_ST
else            (LV_ACIN = 0)           (ACC inactive)
        if      LV_AT = 0
        then    if      VS = 0
                then    N_SP_IS_1 = IP_N_SP_IS_MT__TCO__TCO_ST
                else    N_SP_IS_1 = IP_VS_N_SP_IS_MT__TCO__TCO_ST
        else    if      LV_DLY_N_SP_IS = 1                              /    *)
                then    N_SP_IS_1 = IP_DRI_N_SP_IS__TCO__TCO_ST
                else    N_SP_IS_1 = IP_N_SP_IS_AT__TCO__TCO_ST
```

**) LV_DLY_N_SP_IS : time delay for changing to new idle speed setpoint ended (in case of LV_DRI = 1); this bit is set in the converter torque module*

Catalyst heating N_SP_IS_2

```
If              LV_CH = 1               (Catalyst Heating offset)
then
        if      LV_AT = 1
        then    if      LV_DLY_N_SP_IS = 1
                then    N_SP_IS_2 = N_SP_IS_1 + IP_N_SP_ADD_CAT_AT_DRI
                else    N_SP_IS_2 = N_SP_IS_1 + IP_N_SP_ADD_CAT_AT *
                                        IP_FAC_N_SP_ADD_CAT__T_AST
        else    N_SP_IS_2 = N_SP_IS_1 + IP_N_SP_ADD_CAT_MT *
                                        IP_FAC_N_SP_ADD_CAT__T_AST
else            (LV_CH = 0)
        if      LV_AT = 1               (Offset in case of high transmission oil temperature)
        then    if      TOIL_GB >= C_TOIL_GB_MIN_N_SP_IS
                then    N_SP_IS_2 = N_SP_IS_1 + IP_N_SP_ADD_TOIL_GB
                else    if      TOIL_GB <= C_TOIL_GB_MIN_N_SP_IS
                                        - C_TOIL_GB_HYS_N_SP_IS
                        then    N_SP_IS_2 = N_SP_IS_1
        else    N_SP_IS_2 = N_SP_IS_1
```

Battery Voltage Correction N_SP_IS_3

```
if              LV_ES = 0 and LV_ST = 0
```

then **if** VB < C_VB_N_MIN_ADD_ON for more than C_T_N_MIN_ADD_ON
then LV_VB_N_MIN_ADD = 1
else **if** VB > C_VB_N_MIN_ADD_OFF for more than C_T_N_MIN_ADD_OFF
then LV_VB_N_MIN_ADD = 0
else LV_VB_N_MIN_ADD remains unchanged
else LV_VB_N_MIN_ADD remains unchanged
if LV_VB_N_MIN_ADD = 1
then **If** LV_DRI = 1 **and** VS < C_DRI_VS_MIN_VB
then N_SP_IS_3 = IP_DRI_N_SP_MIN_VB__VB_MMV
else **if** LV_DRI = 0
then N_SP_IS_3 = IP_N_SP_MIN_VB__VB_MMV
else N_SP_IS_3 = 0
else N_SP_IS_3 = 0

Minimum idle speed at rolling vehicle N_SP_IS_4

If VS > 0 **and** LV_ERR_MWSS = 0
then **If** LV_AT = 0
then N_SP_IS_4 = C_N_SP_IS_MIN_VS_MT
else **if** LV_DRI = 1
then N_SP_IS_4 = IP_N_SP_IS_MIN_VS_AT
else N_SP_IS_4 = C_N_SP_IS_MIN_VS_AT
else N_SP_IS_4 = 0

Minimum idle speed at Brake switch off N_SP_IS_5

```
If      LV_AT = 1
then    If      NC_ETC_CONF = 0
        then    If              LV_BRAKE_OFF = 1
                        and     VS < C_VS_MIN_BLS
                        and     LV_DRI = 1
                then            N_SP_IS_5 = C_N_SP_IS_BRK_OFF_AT
                else            N_SP_IS_5 = 0
        else    if              LV_BLS = 0
                        and     VS < C_VS_MIN_BLS
                        and     LV_DRI = 1
                then            N_SP_IS_5 = C_N_SP_IS_BRK_OFF_AT
                else            N_SP_IS_5 = 0
else    N_SP_IS_5 = 0
```

Cooling fan correction N_SP_IS_6

Power compensation with cooling fan high active.

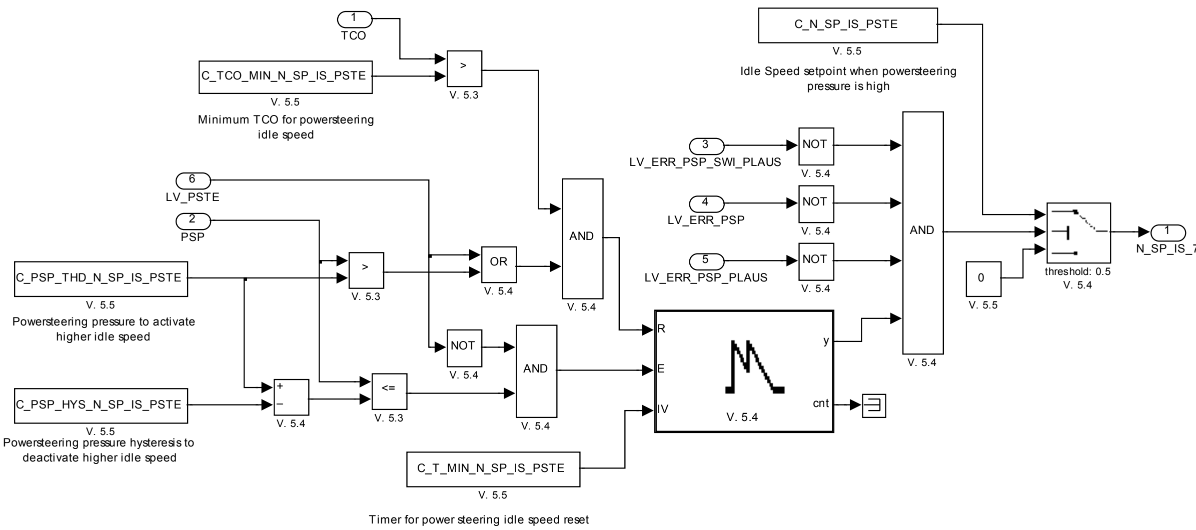
```
If      STATE_CFA = 3
then    If      LV_ACIN = 1
        then    N_SP_IS_6 = C_N_SP_ADD_CFA_ACIN
        else    N_SP_IS_6 = C_N_SP_ADD_CFA
else    N_SP_IS_6 = 0
```

Power steering correction N_SP_IS_7

Increased idle speed when power steering active.

If the system is error free with engine temperature above a threshold and the power steering pressure exceeds a threshold, or the power steering pressure switch is activated, an idle speed setpoint request is activated.

The request is deactivated when the power steering pressure has been below the hysterisis or the pressure switch is deactivated for a calibrateable time.



High oil temperature correction N_SP_IS_8

N_SP_IS_8 = IP_N_SP_IS_TOIL

Engine speed increase request from TCU N_SP_IS_9

If LV_N_INC_TCU_ACT = 1

then N_SP_IS_9 = N_SP_IS_TCU

  if N_SP_IS_9 >= C_N_INC_TCU_REQ_MAX

  then N_SP_IS_9 = C_N_INC_TCU_REQ_MAX

  endif

  if N_SP_IS_9 <= C_N_INC_TCU_REQ_MIN

  then N_SP_IS_9 = C_N_INC_TCU_REQ_MIN

  endif

else N_SP_IS_9 = 0

endif

Electrical load compensation N_SP_IS_GEN_LOAD

Activation : All engine operating state

If CONF_GEN_LOAD = 0
or TCO <= C_TCO_MIN_N_SP_IS_GEN_LOAD
or LV_ERR_GEN_LOAD = 1
then N_SP_IS_GEN_LOAD = 0
else N_SP_IS_GEN_LOAD is calculated as below description after time delay C_T_DLY_N_SP_IS_GEN_LOAD:

The N_SP_IS_GEN_LOAD is performed with an adjustable change limitation C_N_SP_LGRD_IS_GEN_LOAD.

If LV_ERR_GEN_LOAD is set to 1, the N_SP_IS_GEN_LOAD is initialized to C_N_SP_IS_GEN_LOAD_INI with an adjustable change limitation C_N_SP_LGRD_IS_GEN_LOAD.

If GEN_LOAD_MMV_FIL > C_GEN_LOAD_N_SP_IS_GEN_LOAD_ACT
then T_DLY_DEC_N_SP_IS_GEN_LOAD is initialized to C_T_DLY_DEC_N_SP_IS_GEN_LOAD
  If LV_DRI = 0
  then T_DLY_INC_N_SP_IS_GEN_LOAD_DRI is initialized to C_T_DLY_INC_N_SP_IS_GEN_LOAD_DRI
    and T_DLY_INC_N_SP_IS_GEN_LOAD is decreased by 10ms step.
  else T_DLY_INC_N_SP_IS_GEN_LOAD is initialized to C_T_DLY_INC_N_SP_IS_GEN_LOAD
    and T_DLY_INC_N_SP_IS_GEN_LOAD_DRI is decreased by 10ms step.
  endif
  if T_DLY_INC_N_SP_IS_GEN_LOAD = 0
    or T_DLY_INC_N_SP_IS_GEN_LOAD_DRI = 0
  then $N\_SP\_IS\_GEN\_LOAD_n = N\_SP\_IS\_GEN\_LOAD_{n-1}$ + ID_N_SP_IS_INC_GEN_LOAD_MMV_FIL
    and T_DLY_INC_N_SP_IS_GEN_LOAD = C_T_DLY_INC_N_SP_IS_GEN_LOAD
    and T_DLY_INC_N_SP_IS_GEN_LOAD_DRI = C_T_DLY_INC_N_SP_IS_GEN_LOAD_DRI
  endif
else **If** GEN_LOAD_MMV_FIL <= C_GEN_LOAD_N_SP_IS_GEN_LOAD_ACT – C_GEN_LOAD_HYS
  then T_DLY_INC_N_SP_IS_GEN_LOAD is initialized to C_T_DLY_N_SP_IS_GEN_LOAD
    and T_DLY_INC_N_SP_IS_GEN_LOAD_DRI is initialized to

C_T_DLY_N_SP_IS_GEN_LOAD_DRI

and T_DLY_DEC_N_SP_IS_GEN_LOAD is decreased by 10ms step.

endif

if T_DLY_DEC_N_SP_IS_GEN_LOAD = 0

then $N\_SP\_IS\_GEN\_LOAD_n = N\_SP\_IS\_GEN\_LOAD_{n-1}$ - ID_N_SP_IS_DEC_GEN_LOAD_MMV_FIL

and T_DLY_DEC_N_SP_IS_GEN_LOAD = C_T_ DLY_DEC_N_SP_IS_GEN_LOAD

endif

endif

N_SP_IS_GEN_LOAD is limited for each engine condition as below (depending on M/T or A/T):

	Minimum	Maximum
LV_DLY_N_SP_IS = 0 **and** LV_ACIN = 0	IP_N_SP_IS_MT(_AT)	IP_N_SP_IS_MT(_AT) + C_N_SP_IS_GEN_LOAD_MAX
LV_DLY_N_SP_IS = 0 **and** LV_ACIN = 1	IP_ACIN_N_SP_IS_MT(_AT)	IP_ACIN_N_SP_IS_MT(_AT) + C_ACIN_N_SP_IS_GEN_LOAD_MAX
LV_DLY_N_SP_IS = 1 **and** LV_ACIN = 0	IP_DRI_N_SP_IS	IP_DRI_N_SP_IS + C_DRI_N_SP_IS_GEN_LOAD_MAX
LV_DLY_N_SP_IS = 1 **and** LV_ACIN = 1	IP_DRI_ACIN_N_SP_IS	IP_DRI_ACIN_N_SP_IS + C_DRI_ACIN_N_SP_IS_GEN_LOAD_MAX

When T_DLY_DEC_N_SP_IS_GEN_LOAD is reached to 0, if N_SP_IS_GEN_LOAD is not changed compared to previous value after reaching its minimum value, then CTR_N_SP_IS_ GEN_LOAD_INI is increased by 1 step.

If CTR_N_SP_IS_GEN_LOAD_INI is higher than C_CTR_N_SP_IS_GEN_LOAD_INI, N_SP_IS_GEN_LOAD is set to C_N_SP_IS_GEN_LOAD_INI.
(In this case, N_SP_IS_GEN_LOAD could be lower than its minimum value.)

End of line test idle N_SP_IS_EOL

Activation : Only under EOL test request

EOL (End of line) test should be done at high enough rpm to prevent engine stall and to increase passing rate at production line. IVVT EOL needs high oil pressure to activate IVVT contoller and this can be realized only at high rpm.

If EOL_STATE_DIAGCP = 1h (EOL_ACT)

then N_SP_IS_EOL = C_N_SP_IS_EOL_DIAGCP

else **if** LV_TRIP_ACT_IVVT_IN = 1 **or** LV_TRIP_ACT_IVVT_EX = 1

then N_SP_IS_EOL = C_N_SP_IS_EOL_IVVT

else N_SP_IS_EOL = 0 [rpm]

endif

endif

Offset due to atmospheric pressure

In order to assure manifold vacuum for brake performance in all atmospheric conditions e.g. at high altitude, it is possible to calibrate an offset to the target engine idle speed dependant on atmospheric pressure. This offset may be positive or negative. It is added to the final coordinated idle speed setpoint.

Due to the fact that this offset is applied in all conditions, care must be taken to ensure that all engine functional requirements are fufilled even with the offset applied.

N_SP_IS_OFS_AMP = IP_N_SP_IS_OFS_AMP(AMP)

Final value

For driving off assistance, the following offset, which is determined in the module “Driving off assistance”, has to be considered:
And also IVVT short circuit to ground error must be considered for engine stall prevention purpose.

If LV_DRI = 0

N_SP_IS_SLV_IVVT_SCG = IP_N_SP_IS_SLV_IVVT_SCG__TCO

else N_SP_IS_SLV_IVVT_SCG = IP_N_SP_IS_SLV_IVVT_SCG_DRV__TCO

If ERR_SYM_SLV_IVVT_IN(_EX)_1 = SYM_1 *(IVVT Short to Ground)*

and CTR_ABC_SLV_IVVT_IN_(EX)_1 >= C_ABC_MIN_SLV_IVVT_SCG_LIH_IN(_EX)

and TCO >= C_TCO_MIN_SLV_IVVT_SCG_LIH

and VS <= C_VS_MAX_SLV_IVVT_SCG_LIH

then N_SP_IS is increased with C_N_SP_IS_TRA_IVVT_SCG_0_1

N_SP_IS = max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5, N_SP_IS_6, N_SP_IS_7, N_SP_IS_8, N_SP_IS_9, N_SP_IS_GEN_LOAD, N_SP_IS_SLV_IVVT_SCG, N_SP_IS_EOL)

+ N_SP_IS_ADJ_ASA * IP_FAC_N_SP_IS__TCO

+ N_SP_IS_OFS_AMP

and LV_ERR_LIH_SCG_IVVT_IN(_EX) = 1

else *(No error or Short to Battery or Open circuit cases)*

N_SP_IS = max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5, N_SP_IS_6,
N_SP_IS_7, N_SP_IS_8, N_SP_IS_9, N_SP_IS_GEN_LOAD,
N_SP_IS_EOL)

+ N_SP_IS_ADJ_ASA * IP_FAC_N_SP_IS__TCO

+ N_SP_IS_OFS_AMP

and LV_ERR_LIH_SCG_IVVT_IN(_EX) = 0

Change limitation IP_N_SP_LGRD_IS is always applied on N_SP_IS except at transition LV_CH = 0 --> 1 (when Catalyst Heating correction is set) or LV_N_INC_TCU_ACT = 1 (when engine speed increase request from TCU is set)

N_SP_IS_DIF calculation

if max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5,N_SP_IS_6, N_SP_IS_7, N_SP_IS_8,
N_SP_IS_9, N_SP_IS_GEN_LOAD)$_{N}$

<> max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5, N_SP_IS_6, N_SP_IS_7,
N_SP_IS_8, N_SP_IS_9, N_SP_IS_GEN_LOAD)$_{N-1}$

then N_SP_IS_DIF$_{N}$ = max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5, N_SP_IS_6,
N_SP_IS_7, N_SP_IS_8, N_SP_IS_9, N_SP_IS_GEN_LOAD)$_{N}$

– max (N_SP_IS_2, N_SP_IS_3, N_SP_IS_4, N_SP_IS_5, N_SP_IS_6,
N_SP_IS_7, N_SP_IS_8, N_SP_IS_9, N_SP_IS_GEN_LOAD)$_{N-1}$

else N_SP_IS_DIF$_{N}$ = N_SP_IS_DIF$_{N-1}$

NB N_SP_IS_OFS_AMP is not included in N_SP_IS_DIF, as the value is expected to change only slowly. N_SP_IS_DIF reflects only changes in the nominal engine idle speed.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_N_SP_IS_MT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed without additional load on the engine (M/T)					
IP_VS_N_SP_IS_MT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed without additional load on the engine and vehicle rolling (M/T)					
IP_N_SP_IS_AT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed without additional load on the engine (A/T)					
IP_DRI_N_SP_IS	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with DRIVE (A/T) engaged					
IP_ACIN_N_SP_IS_MT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with air conditioner switched on (M/T)					
IP_ACIN_VS_N_SP_IS_MT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with air conditioner switched on and vehicle rolling (M/T)					
IP_ACIN_N_SP_IS_AT	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with air conditioner switched on (A/T)					
IP_DRI_ACIN_N_SP_IS	6*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
LDPM_TCO_ST_N_SP_IS	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with DRIVE (A/T) engaged and air conditioner switched on					
IP_N_SP_ADD_CAT_AT_DRI	5*5	0000...1FE0H	0...8160	1	rpm
LDPM_TCO__N_SP_ADD_CAT	5	0000...FE H	-48...142.5	0.75	°C
LDP_T_AST__N_SP_ADD_CAT	5	0000...FFFF H	0...6553.5	0.1	sec
Additive term on nominal idle speed when Catalyst Heating function is active for AT if LV_DRI = 1					
IP_N_SP_ADD_CAT_AT	5*6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO__N_SP_ADD_CAT	5	0000...FE H	-48...142.5	0.75	°C
LDPM_AMP__N_SP_ADD_CAT	6	0000...FFFF H	0...5432	0,083	Hpa
Additive term on nominal idle speed when Catalyst Heating function is active for AT if LV_DRI = 0					
IP_N_SP_ADD_CAT_MT	5*6	0..1FE0H	0...8160	1	rpm
LDPM_TCO__N_SP_ADD_CAT	5	0000...FE H	-48...142.5	0.75	°C
LDPM_AMP__N_SP_ADD_CAT	6	0...FFFFH	0...5432	0,083	Hpa
Additive term on nominal idle speed when Catalyst Heating function is active for MT					
IP_FAC_N_SP_ADD_CAT__T_AST	5	0000...FFH	0...0.996	1/256	-
LDP_T_AST__FAC_N_SP_ADD_CAT	5	0000...FFFFH	0.6553.5	0.1	sec
Correction factor on norminal idle speed depending on time after start					
IP_DRI_N_SP_MIN_VB__VB_MMV	6	0...1FE0H	0...8160	1	Rpm
LDPM_VB_MMV_N_SP_MIN_VB	6	0...FFH	0...26	0.102	V
Minimum idle speed for low battery voltage in case LV_DRI =1					
IP_N_SP_MIN_VB__VB_MMV	6	0...1FE0H	0...8160	1	Rpm
LDPM_VB_MMV_N_SP_MIN_VB	6	0...FFH	0...26	0.102	V
Minimum idle speed for low battery voltage in case LV_DRI = 0					
IP_N_SP_ADD_TOIL_GB	4	0..1FE0H	0...8160	1	rpm
LDP_TOIL_GB_N_SP_ADD	4	0...FEH	-40...214	1	°C
Nominal Idle Speed offset in case of high oil temperature of Automatic gearbox					
IP_N_SP_LGRD__N_SP_IS_DIF	8	1...FFH	0.25...63.75	0.25	rpm
LDP_N_SP_IS_DIF	8	0...FFFFH	-32768...32767	1	rpm

Nominal Idle Speed change limitation					
C_TOIL_GB_MIN_N_SP_IS	1	0...FEH	-40...214	1	°C
Minimum gearbox oil temperature to activate Nominal Idle Speed correction					
C_TOIL_GB_HYS_N_SP_IS	1	0...FEH	-40...214	1	°C
Hysteresis for minimum gearbox oil temperature to activate Nominal Idle Speed correction					
C_TCO_MIN_N_SP_IS_GEN_LOAD	1	0...FEH	-48...142.5	0.75	°C
Minimum Coolant temperature to activate N_SP_IS_GEN_LOAD					
C_GEN_LOAD_N_SP_IS_GEN_LOAD_ACT	1	0..FFH	0...99,6	0,4	%
Minimum electric load PWM signal to increase N_SP_IS_GEN_LOAD					
C_GEN_LOAD_HYS	1	0..FFH	0...99,6	0,4	%
Hysterisis of electric load PWM signal to decrease N_SP_IS_GEN_LOAD					
C_T_DLY_INC_N_SP_IS_GEN_LOAD	1	0..FFFFH	0...655,35	0.01	sec
Delay time to increase N_SP_IS_GEN_LOAD in case of LV_DRI = 0					
C_T_DLY_INC_N_SP_IS_GEN_LOAD_DRI	1	0..FFFFH	0...655,35	0.01	sec
Delay time to increase N_SP_IS_GEN_LOAD in case of LV_DRI = 1					
C_T_DLY_DEC_N_SP_IS_GEN_LOAD	1	0..FFFFH	0...655,35	0.01	sec
Delay time to decrease N_SP_IS_GEN_LOAD					
C_N_SP_IS_GEN_LOAD_MAX	1	0000...1FE0H	0...8160	1	rpm
Maximum engine speed increment by GEN_LOAD in case of LV_DRI = 0 & LV_ACIN = 0					
C_ACIN_N_SP_IS_GEN_LOAD_MAX	1	0000...1FE0H	0...8160	1	rpm
Maximum engine speed increment by GEN_LOAD in case of LV_DRI = 0 & LV_ACIN = 1					
C_DRI_N_SP_IS_GEN_LOAD_MAX	1	0000...1FE0H	0...8160	1	rpm
Maximum engine speed increment by GEN_LOAD in case of LV_DRI = 1 & LV_ACIN = 0					
C_DRI_ACIN_N_SP_IS_GEN_LOAD_MAX	1	0000...1FE0H	0...8160	1	rpm
Maximum engine speed increment by GEN_LOAD in case of LV_DRI = 1 & LV_ACIN = 1					
ID_N_SP_IS_INC_GEN_LOAD_MMV_FIL	8	0...7F80H	0...8160	0.25	rpm
LDPM_GEN_LOAD_MMV_FIL	8	0..FFFFH	0...99,998	0,00153	%
idle speed increasement depending on PWM electric load signal					
ID_N_SP_IS_DEC_GEN_LOAD_MMV_FIL	8	0...7F80H	0...8160	0.25	rpm
LDPM_GEN_LOAD_MMV_FIL	8	0..FFFFH	0...99,998	0,00153	%
idle speed decreasement depending on PWM electric load signal					
C_N_SP_LGRD_IS_GEN_LOAD	1	1...FFH	0.25...63.75	0.25	Rpm/ sample
N_SP_IS_GEN_LOAD change limitation					
C_CTR_N_SP_IS_GEN_LOAD_INI	1	0...FFH	0...255	1	-
Counter threshold to make initialize N_SP_IS_GEN_LOAD					
C_N_SP_IS_GEN_LOAD_INI	1	0...1FE0H	0...8160	1	rpm
N_SP_IS_GEN_LOAD if it is confirmed low elecric load.					
C_DRI_VS_MIN_VB	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for activating battery charging in case LV_DRI = 1					
C_N_MAX_IDLE_CALC	1	0...1FE0H	0...8160	1	rpm
Maximum threshold of engine speed for calculation					
C_VB_N_MIN_ADD_ON	1	0...FFH	0...25.8984	26/256	V
Voltage threshold to enable low battery voltage compensation function					
C_VB_N_MIN_ADD_OFF	1	0...FFH	0... 25.8984	26/256	V
Voltage threshold to disable low battery voltage compensation function					
C_T_N_MIN_ADD_ON	1	0...FFH	0...255	1	s
Time delay to activate low battery voltage compensation					
C_T_N_MIN_ADD_OFF	1	0...FFH	0...255	1	s
Time delay to deactivate low battery voltae compensation					
C_N_SP_IS_MIN_VS_MT	1	0000...1FE0H	0...8160	1	rpm
Idle speed for rolling vehicle – MT car					
C_N_SP_IS_MIN_VS_AT	1	0000...1FE0H	0...8160	1	rpm
Idle speed for rolling vehicle – AT car					
C_N_SP_IS_TRA_IVVT_SCG_0_1	1	0...FFH	0...511	2	RPM
Nominal idle speed change limitation for during _SLV_IVVT_SCG_LIH = 0 → 1					
C_ABC_MIN_SLV_IVVT_SCG_LIH_IN	1	0...FFH	0...255	1	-
Minimum Anti - bounce counter threshold for limp home activation in case of short circuit ground error (Inlet)					
C_ABC_MIN_SLV_IVVT_SCG_LIH_EX	1	0...FFH	0...255	1	-
Minimum Anti - bounce counter threshold for limp home activation in case of short circuit ground error (Exhaust)					
C_TCO_MIN_SLV_IVVT_SCG_LIH	1	0...FEH	-48...142.5	0.75	°C

Minimum coolant threshold for fast limp home activation in case of short circuit ground error					
C_VS_MAX_SLV_IVVT_SCG_LIH	1	0...FFH	0...255	1	km/h
Maximum VS threshold for fast limp home activation in case of short circuit ground error					
IP_FAC_N_SP_IS__TCO	9	0...FFH	0...0.9961	1/256	-
Correction factor for nominal idle speed adjustment.					
IP_N_SP_IS_SLV_IVVT_SCG	6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with IVVT SCG error at vehicle stop condition (LV_DRI = 0)					
IP_N_SP_IS_SLV_IVVT_SCG_DRV	6	0000...1FE0H	0...8160	1	rpm
LDPM_TCO_4	6	0000...FE H	-48...142.5	0.75	°C
Nominal idle speed with IVVT SCG error at driving condition (LV_DRI = 1)					
C_VS_MIN_BLS	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for N_SP_IS at brake switch off					
C_N_SP_IS_BRK_OFF_AT	1	0000...1FE0H	0...8160	1	rpm
Idle speed at brake switch off – AT car					
C_N_SP_ADD_CFA	1	0000...1FE0H	0...8160	1	rpm
Idle speed correction for high speed cooling fan active in case LV_ACIN = 0					
C_N_SP_ADD_CFA_ACIN	1	0000...1FE0H	0...8160	1	rpm
Idle speed correction for high speed cooling fan active in case LV_ACIN = 1					
C_T_DLY_N_SP_IS_GEN_LOAD	1	0..FFFFH	0...655,35	0.01	Sec
Delay time to enable GEN_LOAD compensation function					
IP_N_SP_IS_MIN_VS_AT	6	0000...1FE0H	0...8160	1	rpm
LDP_VS	6	0...FFH	0...255	1	km/h
Nominal idle speed with vehicle rolling at AT vehicle (LV_DRI = 1)					
C_N_SP_IS_PSTE	1	0000...1FE0H	0...8160	1	rpm
Idle speed setpoint with high power steering pressure					
C_TCO_MIN_N_SP_IS_PSTE	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature threshold for activation of power steering idle increase					
C_PSP_THD_N_SP_IS_PSTE		0..FFH	0...25.5	0.1	MPa
Power steering pressure threshold for activating idle speed increase					
C_PSP_HYS_N_SP_IS_PSTE		0..FFH	0...25.5	0.1	MPa
Power steering pressure hysterisis for deactivating idle speed increase					
C_T_MIN_N_SP_IS_PSTE		1..FFFFH	0.01...655.35	0.01	s
Time for power steering idle speed reset					
IP_N_SP_IS_TOIL	4	0000...1FE0H	0...8160	1	rpm
LDP_TOIL_N_SP_IS_TOIL	4	00...C8 H	-40...160	1	°C
Minimum idle speed request dependent on oil temperature					
C_N_INC_TCU_REQ_MAX	1	0000...1FE0H	0...8160	1	rpm
Maximum engine speed rise request from TCU					
C_N_INC_TCU_REQ_MIN	1	0000...1FE0H	0...8160	1	rpm
Minimum engine speed rise request from TCU					
C_N_SP_IS_EOL_DIAGCP	1	0000...05DCH	0...1500	1	rpm
Idle speed for EVAP EOL diagnosis					
C_N_SP_IS_EOL_IVVT	1	0000...05DCH	0...1500	1	rpm
Idle speed for IVVT EOL diagnosis					
IP_N_SP_IS_OFS_AMP	4	0000...FFFFH	-32768...32767	1	rpm
LDP_AMP_N_SP_IS_OFS	4	0000...FFFF H	0...5434	0,083	hPa
Engine speed setpoint offset due to atmospheric pressure					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_SP_MAX_IS	1	00...1FE0H	0...8160	1	rpm
Max value of N_SP_IS (see chap. 2 system configuration)					

4.2.2 Idle Speed Setpoint Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_SP_IS_COR	V / O	0...1FE0H	0...8160	1	rpm
Corrected idle speed setpoint					
N_SP_IS_COR_GRD	V / O	80...7FH	-4096...4064	32	rpm/sec.
Corrected idle speed setpoint gradient					
N_SP_IS_COR_1	V	0...1FE0H	0...8160	1	rpm
Corrected idle speed setpoint, intermediate value					
N_SP_IS_COR_2	V	0...1FE0H	0...8160	1	rpm
Speed leaded idle speed setpoint					
N_SP_IS_COR_3	V	0...1FE0H	0...8160	1	rpm
Low pass filtered idle speed setpoint					
N_SP_IS_COR_ADD	V	0...1FE0H	0...8160	1	rpm
Additive correction, input low pass filter, intermediate value					
N_SP_IS_COR_ADD_LP_INP	V	0...1FE0H	0...8160	1	rpm
Additive correction, input low pass filter					
N_SP_IS_COR_ADD_MMV	V	0...1FE0H	0...8160	1	rpm
Additive correction, output low pass filter					

Input data:

N	N_SP_IS	LV_CT	C_N_MAX_IDLE_CALC

FUNCTION DESCRIPTION:

General information:

When speed is falling with high negative gradients at the actual engine speed N with N_DIF < 0 (overspeed), an interception function should be active to prevent transient effects round the idle speed setpoint N_SP_IS. These transient effects are normally caused by high integral part values, which may result from activating the idle speed controller at high speed deviations. In order to prevent such situations, a corrected setpoint N_SP_IS_COR is calculated, which intercepts speed when falling into idle range and which leads to small deviations in speed from the (corrected) setpoint and so to small controller outputs. This interception function is only active in case of overspeed.

If this function is activated (LC_N_SP_IS_COR_CALC = 1), the N_DIF_COR calculation, which is described in the following chapter, will be done by the equation N_DIF_COR = N_SP_IS_COR – N. So, N_DIF_COR will not be calculated by N_DIF-N_DIF_MMV·C_N_DIF_FAC in this case! This means, N_DIF_COR will then not include the DT_1-filtering! See also chapter "Engine speed deviations N_DIF, N_DIF_MMV, N_DIF_COR", "N_DIF_COR calculation".

The calculation of N_SP_IS_COR is depending on the actual value N, figure 1.

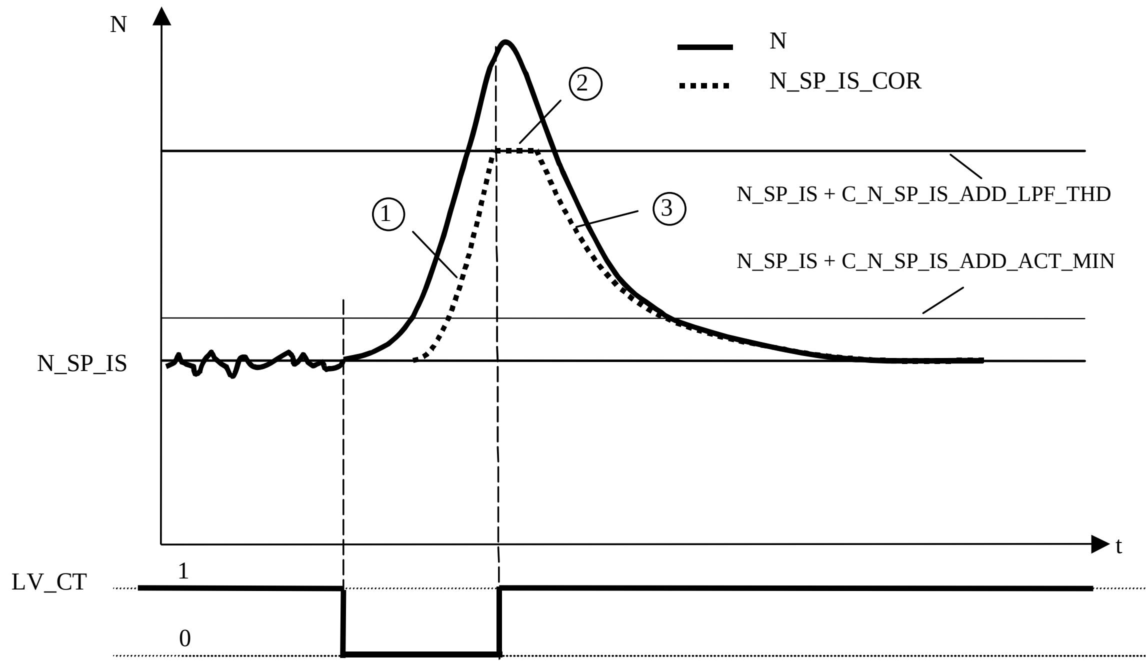


Figure 1: *Calculation of N_SP_IS_COR, part I*

If the driver hits the gas pedal (LV_CT=0) **OR** speed is above the threshold N_SP_IS + C_N_SP_IS_ADD_LPF_THD **AND** speed is above the offset N_SP_IS + C_N_SP_IS_ADD_ACT_MIN, the corrected idle speed setpoint N_SP_IS_COR is calculated as a part of the actual speed, case 1 in figure 1:

IF **[** N > N_SP_IS + C_N_SP_IS_ADD_ACT_MIN **]** **AND** **[{** LV_CT = 0 **}** **OR**
{ N_SP_IS + C_N_SP_IS_ADD_LPF_THD < N_SP_IS_COR_2 **}]**
THEN
N_SP_IS_COR = min { C_FAC_N_SP_IS_COR, 1 } · N,
with 0 <= C_FAC_N_SP_IS_COR <= 1
ENDIF;

(1)

If N_SP_IS_COR reaches N_SP_IS + C_N_SP_IS_ADD_LPF_THD or becomes greater, this upper limit value is held, case 2 in figure 1. When the driver releases the gas pedal (LV_CT=1) and speed is falling again, a low pass filter is activated in case

N_SP_IS_COR_2=C_FAC_N_SP_IS_COR · N = N_SP_IS + C_N_SP_IS_ADD_LPF_THD, (2)

i.e. N reaches

$$N = \frac{N\_SP\_IS + C\_N\_SP\_IS\_ADD\_LPF\_THD}{C\_FAC\_N\_SP\_IS\_COR} \quad (3)$$

Then the corrected setpoint N_SP_IS_COR is lead on an e-function down to the setpoint N_SP_IS. If in this situation the driver hits the gas pedal, N_SP_IS_COR is calculated by eq.(1) again.

The following picture 2 shows a special situation.

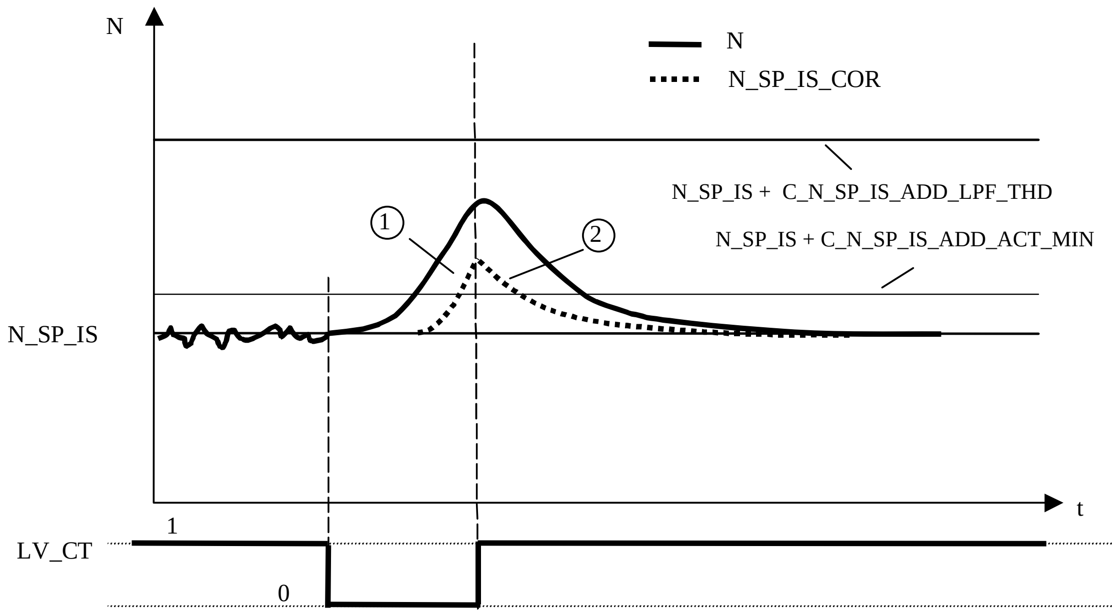


***Figure 2:** Calculation of N_SP_IS_COR, part II*

When the driver releases the gas pedal in the range

N_SP_IS + C_N_SP_IS_ADD_ACT_MIN <= N_SP_IS_COR <= N_SP_IS + C_N_SP_IS_ADD_LPF_THD

again, then the low pass filter starts from the last value N_SP_IS_COR.

Application conditions:

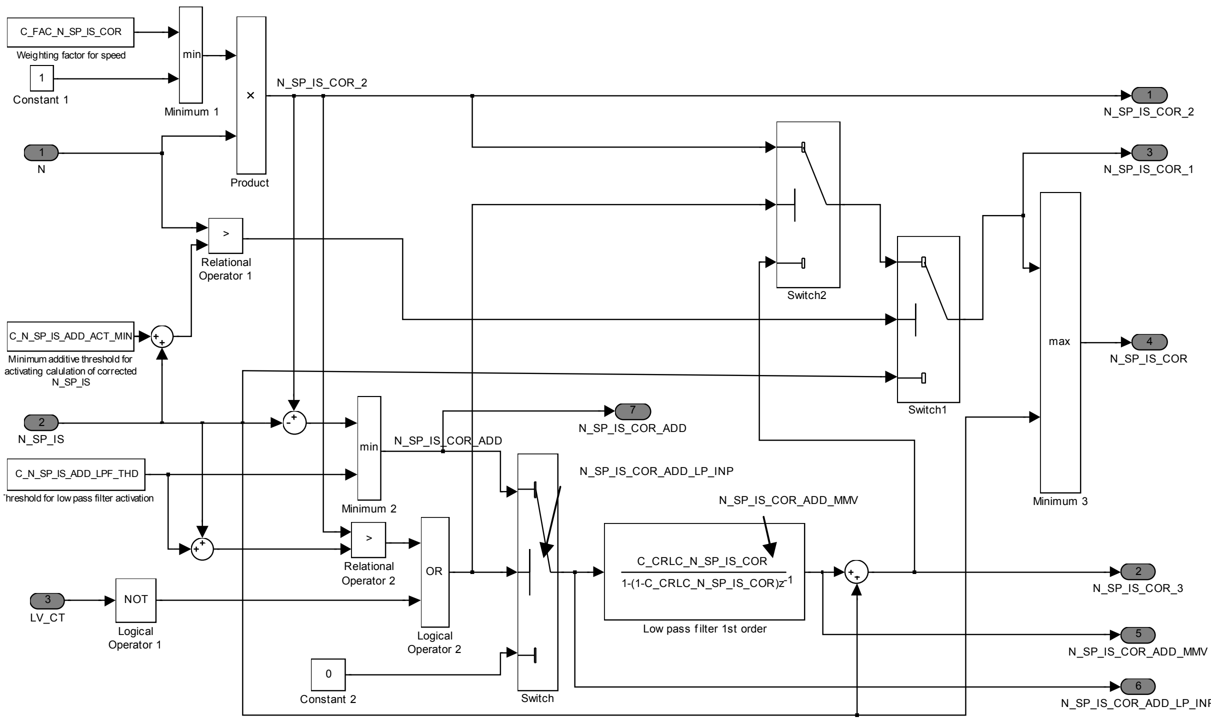
Activation: **(** LC_N_SP_IS_COR_CALC = 1 **) AND (** N <= C_N_MAX_IDLE_CALC **)**

Deactivation: **(** LC_N_SP_IS_COR_CALC = 0 **) OR (** N > C_N_MAX_IDLE_CALC **)**

$N\_SP\_IS\_COR_k = N\_SP\_IS\_COR_{k-1}$

Initialization: N_SP_IS_COR = N_SP_IS, N_SP_IS_COR_GRD = 0 at reset

Update rate: 10 ms



Formula section:

Figure 3: *Structure calculation of N_SP_IS_COR*

Initialization

N_SP_IS_COR_2 = **min {** C_FAC_N_SP_IS_COR, 1 **}** * N

N_SP_IS_COR_ADD = **min {**N_SP_IS_COR_2 - N_SP_IS, C_N_SP_IS_COR_LPF_THD**}**

Initialization low pass filter

IF **(**LV_CT = 0**) OR (**N_SP_IS + C_N_SP_IS_ADD_LPF_THD < N_SP_IS_COR_2**)**

THEN N_SP_IS_COR_ADD_LP_INP = N_SP_IS_COR_ADD;

ELSE N_SP_IS_COR_ADD_LP_INP = 0;

Low pass filter

$$N\_SP\_IS\_COR\_ADD\_MMV_k = N\_SP\_IS\_COR\_ADD\_MMV_{k-1} + C\_CRLC\_N\_SP\_IS\_COR \cdot (N\_SP\_IS\_COR\_ADD\_LP\_INP_k - N\_SP\_IS\_COR\_ADD\_MMV_{k-1});$$

Determination N_SP_IS_COR

IF **[** N > N_SP_IS + C_N_SP_IS_ADD_ACT_MIN **]**

IF **{** LV_CT = 0 **}** **OR**

{ N_SP_IS + C_N_SP_IS_ADD_LPF_THD < N_SP_IS_COR_2 **}**

THEN N_SP_IS_COR_1 = N_SP_IS_COR_2;

ELSE N_SP_IS_COR_1 = N_SP_IS_COR_3 =

N_SP_IS + N_SP_IS_COR_ADD_MMV$_k$;

ELSE N_SP_IS_COR_1 = N_SP_IS;

ENDIF;

N_SP_IS_COR = **max [** N_SP_IS_COR_1, N_SP_IS **]**;

Determination of N_SP_IS_COR gradient :

N_SP_IS_COR gradient is used to correct engine speed gradient used as input of Idle speed P-D controller. It is N_SP_IS_COR variation over 100 msec. :

N_SP_IS_COR_GRD = (N_SP_IS_COR$_{(n)}$ – N_SP_IS_COR$_{(n-10)}$) / 0.1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_N_SP_IS_COR	1	0 ... FF H	0 ... 1.992	0.0078	-
Weighting factor for N_SP_IS_COR					
C_N_SP_IS_ADD_LPF_THD	1	0...1FE0H	0...8160	1	rpm
Threshold for filtering N_SP_IS_COR					
C_CRLC_N_SP_IS_COR	1	0 ... FF H	0 ... 0.9984	0.0039	-
Filter constant for N_SP_IS_COR					
C_N_SP_IS_ADD_ACT_MIN	1	0...1FE0H	0...8160	1	rpm
Minimum additive threshold for activation calculation of corrected N_SP_IS					
C_N_SP_IS_COR_LPF_THD	1	0...1FE0H	0...8160	1	rpm
Threshold for low pass filter activation					
LC_N_SP_IS_COR_CALC	1	0...1	0...1H	1	-
Logical bit for activating the calculation of N_SP_IS_COR					

4.2.3 Engine speed deviations N_DIF, N_DIF_MMV, N_DIF_COR

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_DIF	V/O	8000...7FFFH	-32768...32767	1	rpm
Engine speed deviation N_SP_IS-N					
N_DIF_MMV	V/O	8000...7FFFH	-32768...32767	1	rpm
N_DIF moving mean value					
N_DIF_COR	V/O	8000...7FFFH	-32768...32767	1	rpm
Idle speed control variable					
N_DIF_1	V	8000...7FFFH	-32768...32767	1	rpm
Engine speed deviation 1 N_SP_IS-N					
N_DIF_2	V	8000...7FFFH	-32768...32767	1	rpm
Engine speed deviation 2 N_SP_IS-N					
N_DIF_3	V/O	8000...7FFFH	-32768...32767	1	rpm
Engine speed deviation 3 N_SP_IS-N					

Input data:

N	N_SP_IS	N_SP_IS_COR	LV_IS
LC_N_SP_IS_COR_CALC	C_N_MAX_IDLE_CALC		

FUNCTION DESCRIPTION:

General information:

Application conditions:

Activation: N <= C_N_MAX_IDLE_CALC

Deactivation: N > C_N_MAX_IDLE_CALC

$N\_DIF_k = N\_DIF_{k-1}$

$N\_DIF\_COR_k = N\_DIF\_COR_{k-1}$

$N\_DIF\_3_k = N\_DIF\_3_{k-1}$

Initialization:

N_DIF	= 0	at reset
N_DIF_MMV	= 0	at reset
N_DIF_COR	= 0	at reset

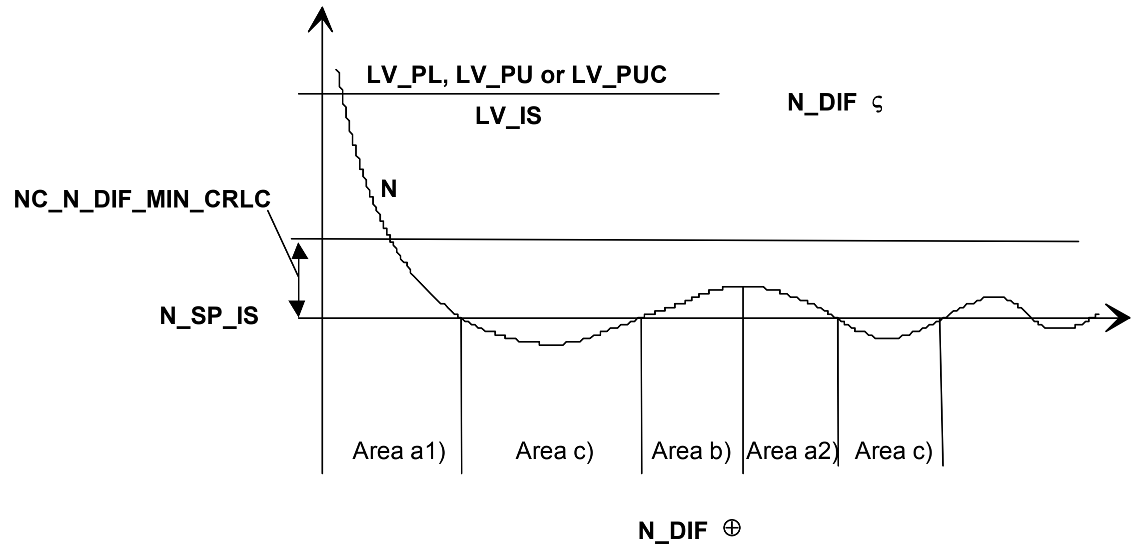
Update rate: 10 ms

Formula section:

N_DIF calculation:

The deviation N_DIF from the nominal idle speed is defined as:

N_DIF = N_SP_IS - N



N_DIF_MMV calculation:

After entry in Idle (LV_IS=1), N_DIF_MMV is set to N_DIF taking into account the limits defined below. Then the moving mean value is determined:

N_DIF_MMV (n) = N_DIF_MMV (n-1) + (N_DIF - N_DIF_MMV(n-1)) * N_DIF_CRLC

The correlation constant C_N_DIF_CRLC may assume 3 discrete values:

a1) Decreasing engine speed above nominal idle speed after entry in Idle (LV_IS=1) and conditions b.) and c.) not previously met (or N_DIF < NC_N_DIF_MIN_CRLC):

(N_DIF (n-3) < N_DIF (n) and N_DIF (n) < 0)

N_DIF_CRLC = **C_N_DIF_CRLC** (per adjustment)

a2) Decreasing engine speed above nominal idle speed

and N_DIF > **NC_N_DIF_MIN_CRLC**

and condition b.) previously fulfilled

N_DIF_CRLC = 1

b) Increasing engine speed above nominal idle speed

((N_DIF (n-3) >= N_DIF (n) and N_DIF (n) < 0)

N_DIF_CRLC = 1

c) Engine speed below nominal idle speed after entry at idle and is still below idle speed

(N_DIF (n) >= 0)

N_DIF_CRLC = 0

Limits of N_DIF_MMV:

C_N_DIF_MIN_MMV <= N_DIF_MMV <= 0

Description:

N_DIF_COR calculation:

N_DIF_COR is relevant for the idle controller (idle-charge actuator and ignition timing):

IF **{** LC_N_SP_IS_COR_CALC = 0 **}** `% this variable is calibratible in`
`% "Engine Speed Setpoint Calculation"`

N_DIF_COR = N_DIF - N_DIF_MMV * C_N_DIF_FAC

ELSE

N_DIF_COR = N_SP_IS_COR – N;

ENDIF;

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DIF_CRLC	1	0...FFH	0...0.996	0.0039	-
Correlation factor for N_DIF_MMV calculation					
C_N_DIF_FAC	1	0...FFH	0...0.996	0.0039	-
Multiplicative factor for N_DIF_COR calculation					
C_N_DIF_MIN_MMV	1	F010...0H	-4080...0	1	rpm
Minimum filtering constant					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_DIF_MIN_CRLC	1	F010...0H	-4080...0	1	rpm
Non calibrated correlation factor for N_DIF_MMV calculation					

4.2.4 Start End Engine Speed

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_MAX_TOL_ST	V / O	0 ... FF H	0 ... 8160	32	rpm
Top level end of start engine speed					

Input data:

TCO	C_N_MAX_IDLE_CALC		

General information:

Application conditions:

Activation: N <= C_N_MAX_IDLE_CALC

Deactivation: N > C_N_MAX_IDLE_CALC

$N\_MAX\_TOL\_ST_k = N\_MAX\_TOL\_ST_{k-1}$

Initialization: N_MAX_TOL_ST = IP_N_MAX_TOL_ST at reset

Update rate: 1 s

Formula section:

N_MAX_TOL_ST = IP_N_MAX_TOL_ST

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_N_MAX_TOL_ST	9	0...FF H	0...8160	32	rpm
LDPM_TCO_9	9	0...FE H	-48 ...142.5	0.75	°C
Top level end of start engine speed					

4.2.5 Engine speed deviation N_DIF_ST

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_DIF_ST	V / O	8000 ... 7FFF H	-32768 ... 32767	1	rpm
Engine speed deviation N_MAX_TOL_ST - N					

Input data:

N	N_MAX_TOL_ST	C_N_MAX_IDLE_CALC	

General information:

The engine speed deviation N_DIF_ST is used as an input variable in module “Torque control at start” for calculation TQ_ST.

Description:

Application conditions:

Activation: N <= C_N_MAX_IDLE_CALC

Deactivation: N > C_N_MAX_IDLE_CALC

$N\_DIF\_ST_{k} = N\_DIF\_ST_{k-1}$

Initialization: N_DIF_ST = conversion(N_MAX_TOL_ST) at reset

Update rate: 10 ms

Formula section:

N_DIF_ST = {conversion(N_MAX_TOL_ST)} - N

conversion means shift from 32 rpm to 1 rpm resolution

4.2.6 Engine speed setpoint ratio

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_SP_IS_RATIO	V / O	0...FFFFH	0... 7,999	122E-6	-
Ratio actual engine speed versus engine speed setpoint at idle					
N_SP_IS_RATIO_ASR	V / O	0...FFFFH	0... 7,999	122E-6	-
Ratio actual engine speed versus engine speed setpoint in case of ASR request					

Input data:

N	N_SP_IS	C_N_MAX_IDLE_CALC	

General information:

Application conditions:

Activation: N <= C_N_MAX_IDLE_CALC

Deactivation: N > C_N_MAX_IDLE_CALC

$N\_SP\_IS\_RATIO_{k} = N\_SP\_IS\_RATIO_{k-1}$

$N\_SP\_IS\_RATIO\_ASR_{k} = N\_SP\_IS\_RATIO\_ASR_{k-1}$

Initialization: N_SP_IS_RATIO = N_SP_IS_RATIO_ASR = 0 at reset

Updat Rate: 10 ms

Formula section:

N_SP_IS_RATIO = N / N_SP_IS

N_SP_IS_RATIO_ASR = N / (N_SP_IS + C_N_ADD_ASR_SP)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_ADD_ASR_SP	1	0...1FE0H	0...8160	1	Rpm
Engine Speed set-point offset in case of low ASR Request					

4.3 Engine speed setpoint for transmission

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_N_INC_TCU_ACT	V/O	0...1H	0...1	1	-
Engine speed increasing activation for transmission request					
LV_T_MAX_N_INC_ACT_NOT_PLAUS	V	0...1H	0...1	1	-
Engine speed increasing activation plausibility					
T_MAX_N_INC_ACT	V	0...FFFFH	0 ... 655.35	0.01	s
Engine speed increasing duration time					
N_INC_OLD	V	0...1FE0H	0 ... 8160	1	rpm
Engine speed when engine speed increasing request is set					

Input data:

N_SP_IS_TCU	N	VS	LV_AT
LV_N_INC_TCU_REQ	LV_N_LIM_ETC_LIH	LV_ERR_TPS	LV_ERR_ISA_1
LV_ERR_ISA_2	LV_ERR_PVS	LV_CT	LV_MSR_ACT
LV_IGK	LV_ERR_CAN_BUS_OFF	LV_ERR_TIMEOUT_TCU1	LV_ERR_CAN_FRF
LV_ERR_CAN	LV_ERR_VS	N_TUR_CONV	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_SP_IS_TCU_MIN	1	0...1FE0H	0...8160	1	rpm
Minimum N_SP_IS_TCU to enable engine speed incrasing request					
C_VS_MIN_N_INC_TCU_ACT	1	0...FFH	0...255	1	Km/h
Minimum vehicle speed to enable engine speed incrasing request					
IP_N_SP_IS_TCU_MAX	8*7	0...1FE0H	0...8160	1	rpm
LDPM_VS__N_SP_IS_TCU	8	0...FFH	0...255	1	km/h
LDPM_GEAR__N_SP_IS_TCU	7	0...6H	0...6	1	-
Maximum N_SP_IS_TCU request for AT versions					
IP_N_INC_TCU_REL_MAX	8*7	0...1FE0H	0...8160	1	rpm
LDPM_VS__N_SP_IS_TCU	8	0...FFH	0...255	1	km/h
LDPM_GEAR__N_SP_IS_TCU	7	0...6H	0...6	1	-
Maximum deviation of engine speed between request value from TCU and saved value at LV_N_INC_TCU_REQ = 0 -> 1					
C_N_INC_TCU_REL_TUR_CONV_MAX	1	0...1FE0H	0 ... 8160	1	rpm
Maximum deviation of engine speed between request value from TCU and torque converter turbine speed					
C_T_MAX_N_INC_NOT_PLAUS	1	0...FFFFH	0 ... 655.35	0.01	s
Maximum time of engine speed increasing activation after activation of the function					
LC_N_INC_TCU_REQ	1	0...1H	0 ... 1	1	-
logical constant to allow engine speed incrasing request					

FUNCTION DESCRIPTION:

General information:

The engine speed setpoint request for AT is calculated in this module, after plausibility check of N_SP_IS_TCU = engine speed setpoint request from TCU.
Monitoring is done in "Transmission control unit diagnosis".

Application conditions:

Initialisation: at reset or N = 0 or LV_IGK = 0-> 1

LV_N_INC_TCU_ACT = 0
LV_T_MAX_N_INC_ACT_NOT_PLAUS = 0
N_INC_OLD = 0

Recurrence: 10ms

Activation:

Enable conditions

LV_AT = 1

Activation of the timer T_MAX_N_INC_ACT

Recurrence: 10 ms

If LV_N_INC_TCU_ACT = 1
Then T_MAX_N_INC_ACT incremented until FFFF H
Else T_MAX_N_INC_ACT is initialized with 0

Maximum Requested GS Intervention Time

Maximum requested time for the engine speed controller to be active. If this time is exceeded the request is set as not plausible.

If T_MAX_N_INC_ACT > C_T_MAX_N_INC_NOT_PLAUS
Then LV_T_MAX_N_INC_ACT_NOT_PLAUS = 1
Endif

Save engine speed at LV_N_INC_TCU_REQ set

If LV_N_INC_TCU_REQ = 0 -> 1
Then N_INC_OLD = N
Endif

Check of the activation condition of the rpm controller

If LC_N_INC_TCU_REQ = 1
and VS > C_VS_MIN_N_INC_TCU_ACT
and N_SP_IS_TCU > C_N_SP_IS_TCU_MIN
and N_SP_IS_TCU < IP_N_SP_IS_TCU_MAX
and N_SP_IS_TCU – N_INC_OLD < IP_N_INC_TCU_REL_MAX
and N_SP_IS_TCU – N_TUR_CONV < C_N_INC_TCU_REL_TUR_CONV_MAX
and LV_N_INC_TCU_REQ = 1
and LV_ERR_CAN_BUS_OFF = 0
and LV_ERR_TIMEOUT_TCU1 = 0
and LV_ERR_CAN_FRF = 0
and LV_ERR_CAN = 0

```
        and    LV_ERR_VS = 0
        and    LV_ERR_TPS = 0
        and    LV_CT = 1
        and    LV_MSR_ACT = 0
        and    LV_T_MAX_N_INC_ACT_NOT_PLAUS = 0
        and    ( LV_N_LIM_ETC_LIH = 0    and    LV_ERR_PVS = 0 )
                                                : NC_ETC_CONF = 1 case
        and    ( LV_ERR_ISA_1 = 0        and    LV_ERR_ISA_2 = 0 )
                                                : NC_ETC_CONF = 0 case
then    LV_N_INC_TCU_ACT = 1
else    LV_N_INC_TCU_ACT = 0
endif
```

4.4 Engine Speed Limit Coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_MAX_THD	V/O	0...1FE0H	0...8160	1	rpm
Actual engine speed limit					
N_MAX_VS_0	V	0...FFH	0...8160	32	rpm
Engine speed limit for stopped vehicle					
LV_N_MAX_VS_0	V/O	0...1H	0...1	-	-
engine speed limitation for stopped vehicle active					
T_N_MAX_VS_0_1	V	0...FFFFH	0...6553.5	0.1	sec.
Timer to detect LV_N_MAX_VS_0					
T_N_MAX_VS_0_2	V	0...FFFFH	0...6553.5	0.1	sec.
Timer for engine speed limitation					
N_MAX_GR	-	0...FFH	0...8160	32	rpm
Engine speed limit at high gear ratio					
LV_N_MAX_GR	V	0...1H	0...1	-	-
engine speed limitation for at high gear ratio active					
N_MAX_VS_ERR	-	0...FFH	0...8160	32	rpm
Engine speed limit with present vehicle speed signal error					
LV_N_MAX_VS_ERR	V	0...1H	0...1	-	-
engine speed limitation with present VS error					
N_MAX_ETC_LIH	-	0...FFH	0...8160	32	rpm
Engine speed limit for ETC limp home					
N_MAX_CRK_LIH	-	0...FFH	0...8160	32	rpm
Engine speed limit in case of crank error with limp home activated					
N_MAX_VS_MAX	V	0...1FE0H	0...8160	1	rpm
Engine speed limit in case of vehicle speed limitation					
VS_RATIO	V	0...FFFFH	0...32	4.883E-4	-
Ratio between vehicle speed limit and actual vehicle speed					
LV_VS_MAX	V	0...1H	0...1	-	-
Boolean signals an active vehicle speed limitation					
N_MAX_CAM_LIH	-	0...FFH	0...8160	32	rpm
Engine speed limit in case of cam error with limp home activated					
N_MAX_ONE_TOOTH_OFF	V	0…FFH	0…8160	32	Rpm
Engine speed limit in case of CAM chain ONE_TOOTH_OFF detection					
N_MAX_PVS_LIH	-	0…FFH	0…8160	32	Rpm
Engine speed limit for PVS limp home					
LV_N_MAX_PVS_LIH	V	0…1H	0…1	-	-
Engine speed limitation for PVS limp home active					
N_MAX_NEUTRAL	V	0…FFH	0…8160	32	rpm
Engine speed limit in case of gearbox in neutral					
N_MAX_TOIL	V	0…FFH	0…8160	32	rpm
Engine speed limit due to high engine oil temperature					
LV_N_MAX_TOIL	V	0...1H	0...1	-	-
Active engine speed limitation due to high oil temperature					
N_MAX_BAS	V	0…1FE0H	0…8160	1	rpm
Basic engine speed limitation (gradient limited)					
N_MAX_RAW	V	0…FFH	0…8160	32	rpm
Basic engine speed limitation (no gradient limitation)					
LV_N_MAX_RED_ACT	V	0...1H	0...1	-	-
Active reduced engine speed limitation due to sustained engine speed limitation					
CTR_N_MAX	V	0...FFFFH	0...65535	1	-
Static engine speed limitation counter					

Input data:

VS	GR_AT	GR_MT	LV_AT
LV_ACT_LIH_CRK_CAM_IN_1	GEAR	LV_ERR_VS	LV_CT
LV_DRI	TCO	N_32	LV_ERR_CRK
LV_ERR_TCO	LV_ERR_TCO_STUCK	LV_ERR_TCO_GRD	LV_ERR_TCO_PLAUS
LV_ERR_MWSS	LV_ERR_TPS	LV_ERR_TPS_PLAUS	LV_ERR_MAF
LV_ERR_LOAD_PLAUS	TPS	VS_MAX	N
LV_LIH_ERR_CAM	LV_ERR_CAM_IN_i	LV_ERR_TOOTH_OFF_IN[NC_NR_CAM_CBK]	LV_ACT_PVS_LIH
TOIL	LV_N_LIM_REQ_MON	LV_N_MAX	

FUNCTION DESCRIPTION:

General information:

Several conditions can activate an engine speed limit. The actual engine speed limit is calculated as minimal selection of all requests.

Several conditions can activate an engine speed limit:

a1) Engine speed limit for stopped vehicle :

a2) Engine speed limit for highest gear ratio in A/T or M/T-vehicle:

a3) Engine speed limit in case of vehicle speed signal error

a4) Engine speed limit in case of throttle position signal error, mass air flow sensor error or in case of engine speed limitation required by monitoring concept

a5) Engine speed limitation for vehicle speed limitation:

a6) Engine speed limitation in case of CRK error with limp-home activated

a7) Engine speed limitation in case of CAM error with limp-home activated

a8) Engine speed limitation in case of CAM chain ONE_TOOTH_OFF

a9) Engine speed limitation in case of PVS limp-home activated

a10) Engine speed limitation in case of gearbox in neutral

a11) Engine speed limitation in case of too high engine oil temperature

a12) Engine speed limitation in case of sustained engine speed limiter driving

Time recurrence: 100 ms

Initialization (at engine start):

N_MAX_THD	=	NC_N_REF_MAX
N_MAX_NEUTRAL	=	NC_N_REF_MAX
N_MAX_TOIL	=	NC_N_REF_MAX
LV_N_MAX_TOIL	=	0
LV_N_MAX_VS_0	=	0
LV_N_MAX_GR	=	0
LV_N_MAX_VS_ERR	=	0
LV_VS_MAX	=	0

T_N_MAX_VS_0_1	=	0
T_N_MAX_VS_0_2	=	0
LV_N_MAX_PVS_LIH	=	0
LV_N_MAX_RED_ACT	=	0
N_MAX_BAS	=	C_N_MAX
CTR_N_MAX	=	0

Formula section:

a1) Engine speed limit for stopped vehicle :

In order to prevent the catalyst system from over-heating, it is detected if the engine speed stays high with vehicle stopped during a pre-defined time.

if VS = 0
and LV_CT = 0
and LV_DRI = 0
and TCO > C_TCO_N_MAX_VS_0
and **(** N_32 > C_N_MAX_VS_0 **or** LV_N_MAX_VS_0 = 1 **)** [1])
and LV_ERR_CRK = 0
and LV_ERR_CAM_IN_i = 0
and LV_ERR_TCO = 0
and LV_ERR_TCO_STUCK = 0
and LV_ERR_TCO_GRD = 0
and LV_ERR_TCO_PLAUS = 0
and LV_ERR_VS = 0
and LV_ERR_MWSS = 0
then increment T_N_MAX_VS_0_1 (Do NOT reset value if saturation is reached)
if T_N_MAX_VS_0_1 > C_T_N_MAX_VS_0
then LV_N_MAX_VS_0 = 1
and increment T_N_MAX_VS_0_2 (Do NOT reset value if saturation is reached)
and N_MAX_VS_0 = IP_N_MAX_VS_0
else LV_N_MAX_VS_0 = 0
and reset T_N_MAX_VS_0 _1, reset T_N_MAX_VS_0_2
and N_MAX_VS_0 = NC_N_REF_MAX

[1]) This special condition permits to check the N threshold only for activation of LV_N_MAX_VS_0. Once LV_N_MAX_VS_0 is set, the N condition must not be checked anymore.

a2) Engine speed limit for highest gear ratio in A/T or M/T-vehicle:

With highest gear an engine speed limitation can be activated to protect the cardan joints.

```
if      LV_ERR_VS = 0
  and   LV_ERR_MWSS = 0
  and   {     ( LV_AT = 0   and   GR_MT >= C_GR_MT )
        or    ( LV_AT = 1   and   GR_AT >= C_GR_AT ) }
then    N_MAX_GR          = C_N_MAX_GR
  and   LV_N_MAX_GR       = 1
else    N_MAX_GR          = NC_N_REF_MAX
  and   LV_N_MAX_GR       = 0
```

a3) Engine speed limit in case of vehicle speed signal error

```
if      ( LV_ERR_VS = 1    or    LV_ERR_MWSS = 1 )
and     { (LV_AT = 1  and   GR_AT >= C_GR_AT )
        or    LV_AT = 0 }
then    N_MAX_VS_ERR = C_N_MAX_VS_ERR
        and   LV_N_MAX_VS_ERR = 1
else    N_MAX_VS_ERR = NC_N_REF_MAX
        LV_N_MAX_VS_ERR = 0
```

a4) Engine speed limitation request by the ETC limp home management or Monitoring Concept

```
if      STATE_N_LIM_ETC_REQ = N_LIM_CON or LV_N_LIM_REQ_MON = 1
then    N_MAX_ETC_LIH = C_N_MAX_ETC_LIH
else    if    STATE_N_LIM_ETC_REQ = N_LIM_PVS
        then  N_MAX_ETC_LIH = IP_N_MAX_ETC_LIH (PV_AV)
        else  N_MAX_ETC_LIH     = NC_N_REF_MAX
        endif
endif
```

a5) Engine speed limitation for vehicle speed limitation:

In case of vehicle speed limitation the engine speed limitation will be activated.

VS_LIM = max**{** VS, (VS_MAX/32) **}**
(to avoid division by zero and to ensure no overflow of VS_RATIO)

VS_RATIO = VS_MAX / VS_LIM

N_MAX_VS_MAX = N * VS_RATIO

if VS_RATIO <= 1.0
then LV_VS_MAX = 1
else LV_VS_MAX = 0
endif

a6) Engine speed limitation in case of CRK error with limp-home activated

if LV_ACT_LIH_CRK_CAM_IN_1 = 1

then N_MAX_CRK_LIH = C_N_MAX_CRK_LIH

else N_MAX_CRK_LIH = NC_N_REF_MAX

a7) Engine speed limitation in case of CRK error with limp-home activated

if LV_LIH_ERR_CAM = 1

then N_MAX_CAM_LIH = C_N_MAX_CAM_LIH

else N_MAX_CAM_LIH = NC_N_REF_MAX

a8) Engine speed limitation in case of CAM chain ONE_TOOTH_OFF

if LV_ERR_TOOTH_OFF_IN[NC_NR_CAM_CBK] = 1

then N_MAX_ONE_TOOTH_OFF = C_N_MAX_ONE_TOOTH_OFF

else N_MAX_ONE_TOOTH_OFF = NC_N_REF_MAX

endIf

a9) Engine speed limitation in case of PVS limp-home activated

if LV_ACT_PVS_LIH = 1

then N_MAX_PVS_LIH = C_N_MAX_PVS_LIH

LV_N_MAX_PVS_LIH = 1

else N_MAX_PVS_LIH = NC_N_REF_MAX

LV_N_MAX_PVS_LIH = 0

endIf

a10) Engine speed limitation in case of gearbox in neutral

if	LV_AT = 1 **and** LV_DRI = 0
then	N_MAX_NEUTRAL = C_N_MAX_NEUTRAL_AT
elseif	LV_AT = 0 **and** GEAR = 0
then	N_MAX_NEUTRAL = C_N_MAX_NEUTRAL_MT
else	N_MAX_NEUTRAL = NC_N_REF_MAX

a11) Engine speed limitation in case of too high engine oil temperature

if	TOIL ≥ C_TOIL_THD_N_MAX **and** LV_N_MAX_TOIL = 0	
then	LV_N_MAX_TOIL = 1(keep 1 until below 'else' conditions are fulfilled.)	
else	**if**	TOIL < C_TOIL_THD_N_MAX – C_TOIL_HYS_N_MAX
	and	LV_N_MAX_TOIL = 1
	then	LV_N_MAX_TOIL = 0

If	LV_N_MAX_TOIL = 1	
then	**if** LV_AT = 0	
	then	N_MAX_TOIL = IP_N_MAX_TOIL_MT
	else	N_MAX_TOIL = IP_N_MAX_TOIL_AT
else	N_MAX_TOIL = NC_N_REF_MAX	

a12) Engine speed limitation in case of sustained engine speed limiter driving

- *A counter counts up when the engine speed limit is above the threshold*
- *When the counter reaches a threshold the engine speed limit is ramped down*
- *The counter is decremented when the engine limiter is not active*
- *When the counter reaches zero, the higher engine speed limit is allowed again*

if	N_32 > C_N_MAX_RED **and** LV_N_MAX_RED_ACT = 0	
then	CTR_N_MAX = CTR_N_MAX$_{(n-1)}$ + C_CTR_N_MAX_INC	
else		
	if	LV_N_MAX = 0 **and** N_32 <= C_N_MAX_RED
	then	CTR_N_MAX = CTR_N_MAX$_{(n-1)}$ - C_CTR_N_MAX_DEC
		CTR_N_MAX = max(0,CTR_N_MAX) *counter may not go below zero*
	endif	
endif		

Flip Flop construction for LV_N_MAX_RED_ACT

```
if      CTR_N_MAX >= C_CTR_N_MAX_THD
then    LV_N_MAX_RED_ACT = 1
else
        if      CTR_N_MAX = 0
        then    LV_N_MAX_RED_ACT = 0
        endif
endif
```

Selection of raw engine speed limit

```
if      LV_N_MAX_RED_ACT = 1
then    N_MAX_RAW = C_N_MAX_RED
else    N_MAX_RAW = C_N_MAX
endif
```

Gradient limitation of threshold

N_MAX_BAS = N_MAX_RAW (resolution change and gradient limitation)

N_MAX_BAS is gradient limited with C_N_MAX_LGRD in both direction

b) Selection of the currently active engine speed limit N_MAX_THD:

The lowest active engine speed limitation is selected as input for the controller:

```
N_MAX_THD   = MAX {   C_N_MAX_THD_MIN,
                      MIN (  N_MAX_VS_0,
                             N_MAX_GR,
                             N_MAX_VS_ERR,
                             N_MAX_ETC_LIH,
                             N_MAX_VS_MAX,
                             N_MAX_CRK_LIH,
                             N_MAX_CAM_LIH,
                             N_MAX_ONE_TOOTH_OFF,
                             N_MAX_PVS_LIH,
                             N_MAX_NEUTRAL,
                             N_MAX_TOIL
                             N_MAX_BAS        )     }
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_N_MAX_VS_0	-	00...FEH	-48...142.5	0.75	°C
Coolant temperature threshold to detect LV_N_MAX_VS_0					
C_T_N_MAX_VS_0	-	0...FFFFH	0...6553.5	0.1	sec.
Timer to detect LV_N_MAX_VS_0					
C_N_MAX_VS_0	1	0...FFH	0...8160	32	rpm
Engine speed limit for stopped vehicle					
C_CTR_N_MAX_DEC	1	0...FFFFH	0....65535	1	-
Static engine speed limitation counter decrement					
C_CTR_N_MAX_INC	1	0...FFFFH	0....65535	1	-
Static engine speed limitation counter increment					
C_CTR_N_MAX_THD	1	1...FFFFH	1....65535	1	-
Static engine speed limitation counter threshold					
C_GR_MT	1	0...FFH	0....255	1	-
Gear threshold for MT-vehicle engine speed limitation in high gear					
C_GR_AT	1	0...FFH	0....255	1	-
Gear threshold for AT-vehicle engine speed limitation in high gear					
C_N_MAX_GR	1	0...FFH	0...8160	32	rpm
Engine speed limit at high gear ratio					
C_N_MAX_VS_ERR	1	0...FFH	0...8160	32	rpm
Engine speed limit with VS error					
C_N_MAX_ETC_LIH	1	0...FFH	0...8160	32	rpm
Engine speed limit for ETC limp home					
IP_N_MAX_ETC_LIH	1x4	0 ... FFH	0 ... 8160	32	rpm
LDP_PV_AV_N_MAX_ETC_LIH	4	0 ... FFH	0 ... 99.6	0.39	%
Engine speed limit for ETC limp home					
C_N_MAX_CRK_LIH	1	0...FFH	0...8160	32	rpm
Engine speed limit in case of crank error with limp-home activated					
C_N_MAX_LGRD	1	0...1FE0H	0...8160	1	rpm
Limitation gradient for basic engine speed limitation					
C_N_MAX	1	0...FFH	0...8160	32	rpm
General engine speed threshold for engine speed limitation					
C_N_MAX_RED	1	0...FFH	0...8160	32	rpm
Engine speed threshold for continual engine speed limitation					
C_N_MAX_THD_MIN	1	0...FFH	0...8160	32	rpm
Minimum allowed engine speed threshold					
C_N_MAX_CAM_LIH	1	0...FFH	0...8160	32	rpm
Engine speed limit in case of cam error with limp-home activated					
C_N_MAX_ONE_TOOTH_OFF	1	0…FFH	0…8160	32	Rpm
Engine speed limit in case of CAM chain ONE_TOOTH_OFF detection					
C_N_MAX_PVS_LIH	1	0…FFH	0…8160	32	Rpm
Engine speed limit for PVS limp home					
C_N_MAX_NEUTRAL_AT	1	0...FFH	0...8160	32	rpm
Engine speed limitation in neutral (AT)					
C_N_MAX_NEUTRAL_MT	1	0...FFH	0...8160	32	rpm
Engine speed limitation in neutral (MT)					
IP_N_MAX_VS_0	6	0...FFH	0...8160	32	rpm
LDP_T_N_MAX_VS_0_2	6	0...FFFFH	0...6553.5	0.1	sec.
Engine speed limit for stopped vehicle					
C_TOIL_THD_N_MAX	1	0...C8H	0...200	1	°C
Oil temperature threshold for engine speed limitation at high oil temperatures					
C_TOIL_HYS_N_MAX	1	0...28H	0...40	1	°C
Oil temperature hysterisis for engine speed limitation at high oil temperatures					
IP_N_MAX_TOIL_MT	6	0...FFH	0...8160	32	[rpm]
LDP_TOIL_IP_N_MAX_TOIL_MT	6	0...C8H	-40...160	1	[°C]
Engine speed limitation depending on oil temperature for MT					

IP_N_MAX_TOIL_AT	6	0...FFH	0...8160	32	[rpm]
LDP_TOIL_IP_N_MAX_TOIL_AT	6	0...C8H	-40...160	1	[°C]
Engine speed limitation depending on oil temperature for AT					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_REF_MAX	1	0...FFH	0...8160	32	rpm
Threshold for not active engine speed limitation = 8160rpm					

4.5 Mass air flow variables

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
MAF_DIF	O/V	8000... 7FFFH	-694.5106... 694.4894	0.0211948	[mg/stk]
Mass air flow difference per segment					
MAF_KGH	O/V	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Air-mass flow per segment in kg/h					
MAF_MMV	O/V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
MAF moving mean value					
MAF_MMV_DIF	O/V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Difference between filtered MAF and instantaneous MAF moving mean value					

Input Data:

MAF_THR	MAF_KGH_MES	LV_MAF_CTL	MAF
SEG_INC	NC_MAF_CONF		

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_MAF_MMV_DIF	1	0... FFH	0... 0.9960938	3.9063e-3	[-]
Filtering constant for MAF moving mean value calculation					
C_MAF_MMV_HYS	1	0... 7FFFH	0... 694.4788	0.0211945	[mg/stk]
Threshold for detecting raising MAF_MMV					

General Information

This module is calculated segment synchronously or two segment synchronously.

The air mass flow MAF_KGH is determined depending on the presence of an air mass flow meter in the system.

- If no air mass flow meter is present, the intake manifold model based MAF_THR is copied to MAF_KGH.

- In a MAF controlled system the measured air mass flow is copied to MAF_KGH.

MAF_DIF is the increase of MAF over the previous segment. In case of two-segment synchronous operation it is half the increase since the previous calculation.

In the case of rapidly falling load, the transition to (LV_PUC) can be accelerated in order to avoid an engine speed run-up, which is undesirable. To trigger this function (see the following chapters: ignition, ignition angle correction for trailing throttle) some load conditions have to be fulfilled.

Hint: The calculation of this module is done synchronously to multiples of a segment.

Function description

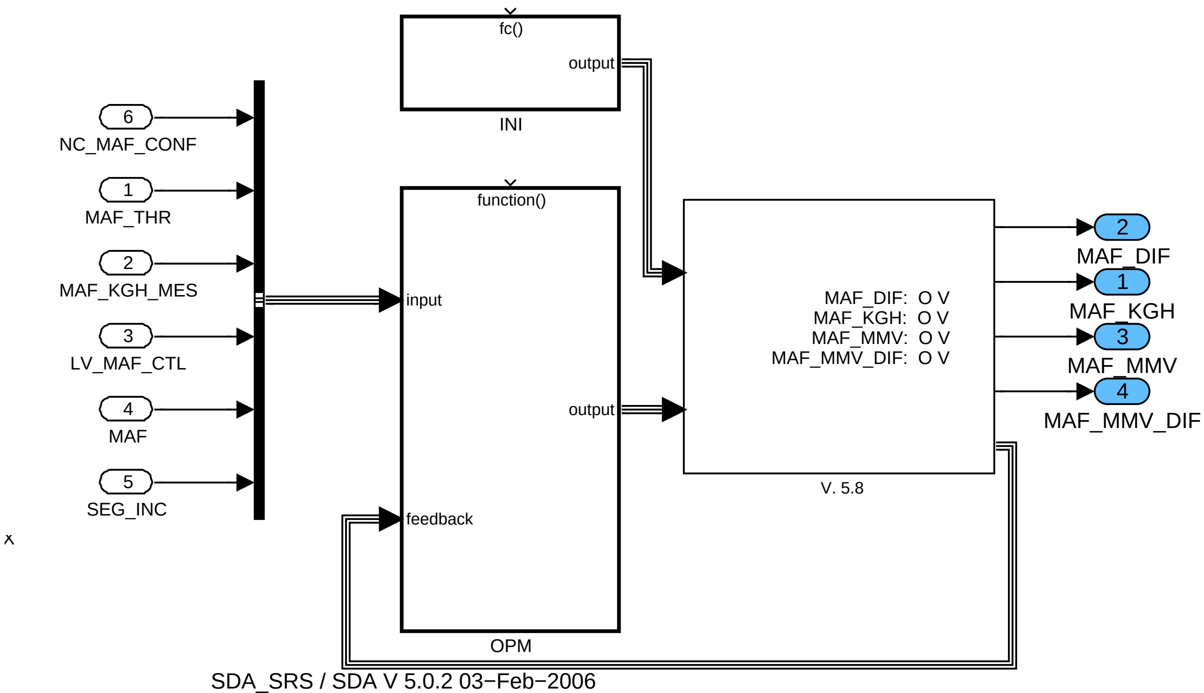


Figure 1:
Path: INSY_SELCTMAF0

4.5.1 Initialization:

4.5.1.1 Initialization at reset and ERU2ES:

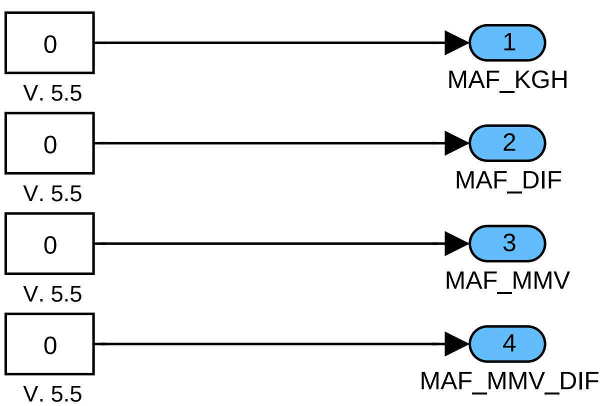


Figure 2:
Path: INSY_SELCTMAF0/INI/INI_RST_ERU2ES

4.5.2 operate:

4.5.2.1 operate_SEG:

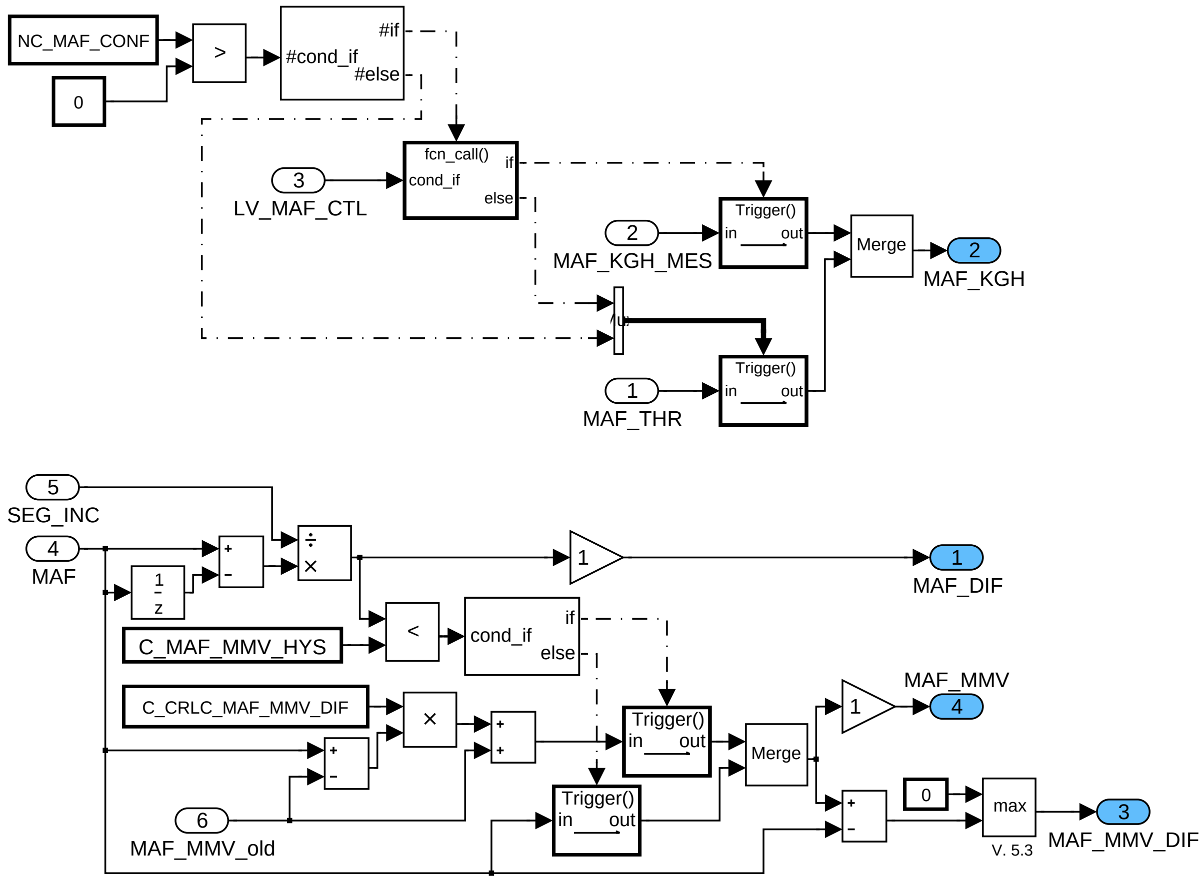


Figure 3:
Path: INSY_SELCTMAF0/OPM/OPM_SEG

4.6 Acquisition of mass air flow

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_KGH_MES_BAS	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Raw air-mass flow measured (without correction, to be used for diagnosis only)					
MAF_MAX	O/V	0...FFFFH	0...1.02398E+3	0.015625	kg/h
Maximum value per segment for MAF acquisisition which is converted by the table ID_MAF_TAB					
MAF_MIN	O/V	0...FFFFH	0...1.02398E+3	0.015625	kg/h
Minimum value per segment for MAF acquisisition which is converted by the table ID_MAF_TAB					

Input data:

VP_MAF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_MAF_TAB	256	0...FFFFH	0...1.02398E+3	0.015625	kg/h
LDP_VP_MAF_ID_MAF_TAB	256	0...7FFFH	0...4.99984741	1.52588E-4	V
conversion table for MAF value					

4.6.1 INSY_SIGCVMAF0.

General information:

The raw value for MAF is measured by continous conversion every 1 ms. The raw values (8 bit) are converted by a table ID_MAF_TAB (size: 256 words) into values with the unit [kg/h] and summed up in alternating two buffers, MAF_SUM and MAF_SUM_1. The number of values is counted in MAF_CTR or MAF_CTR_1.

Depending on the logical value LV_MAF_SEG one of the two buffers — MAF_SUM or MAF_SUM_1 — and the respective counter — MAF_CTR or MAF_CTR_1 — are selected for writing, the contents of the other buffer (counter) is used for diagnosis and calculations.

After having read out one buffer — MAF_SUM or MAF_SUM_1 — the buffer has to be cleared as well as the corresponding counter — MAF_CTR or MAF_CTR_1.

A two-buffer-system is used to avoid incorrect MAF calculation. The buffers alternate with each change of segment, i. e. LV_MAF_SEG is toggled each segment always on fixed tooth event before setting crankshaft trigger.

Application Condition

Activation: At every engine state

Deactivation: -

Initialization: After HW reset, the buffers - MAF_SUM and MAF_SUM_1 - and the corresponding counters - MAF_CTR and MAF_CTR_1 - have to be cleared and the pointer has to be initialized with the address of ID_MAF_TAB.

LV_MAF_SEG=0

Calculation recurrence:

Segment synchronous: Calculation of MAF_KGH_MES_BAS

1 ms: Calculation of MAF_SUM and MAF_SUM_1

Function Description

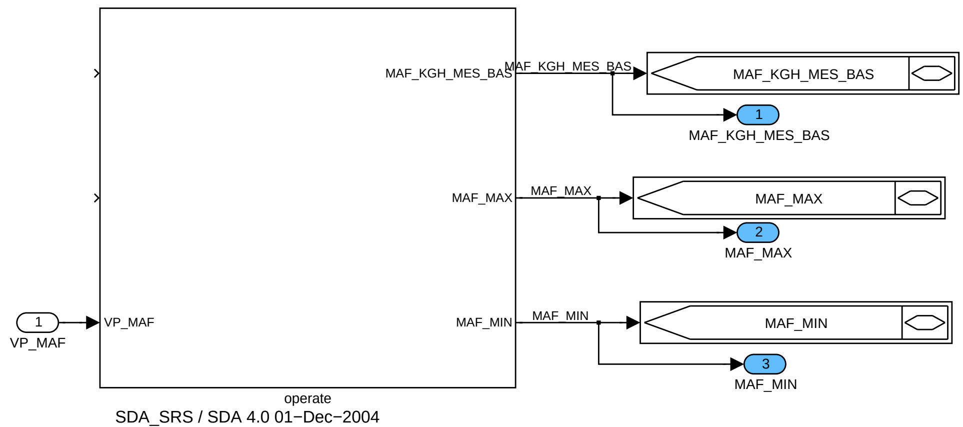


Figure 4 INSY_SIGCVMAF0

4.6.1.1 SUBFUNCTION: operate

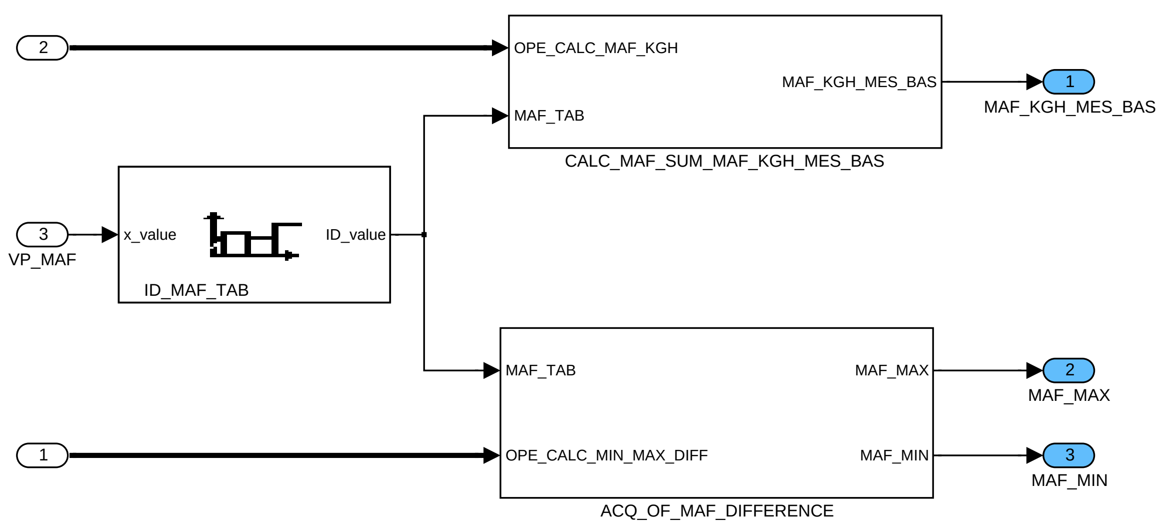


Figure 5 INSY_SIGCVMAF0/ operate

Calculation of MAF_SUM and MAF_KGH_MES_BAS

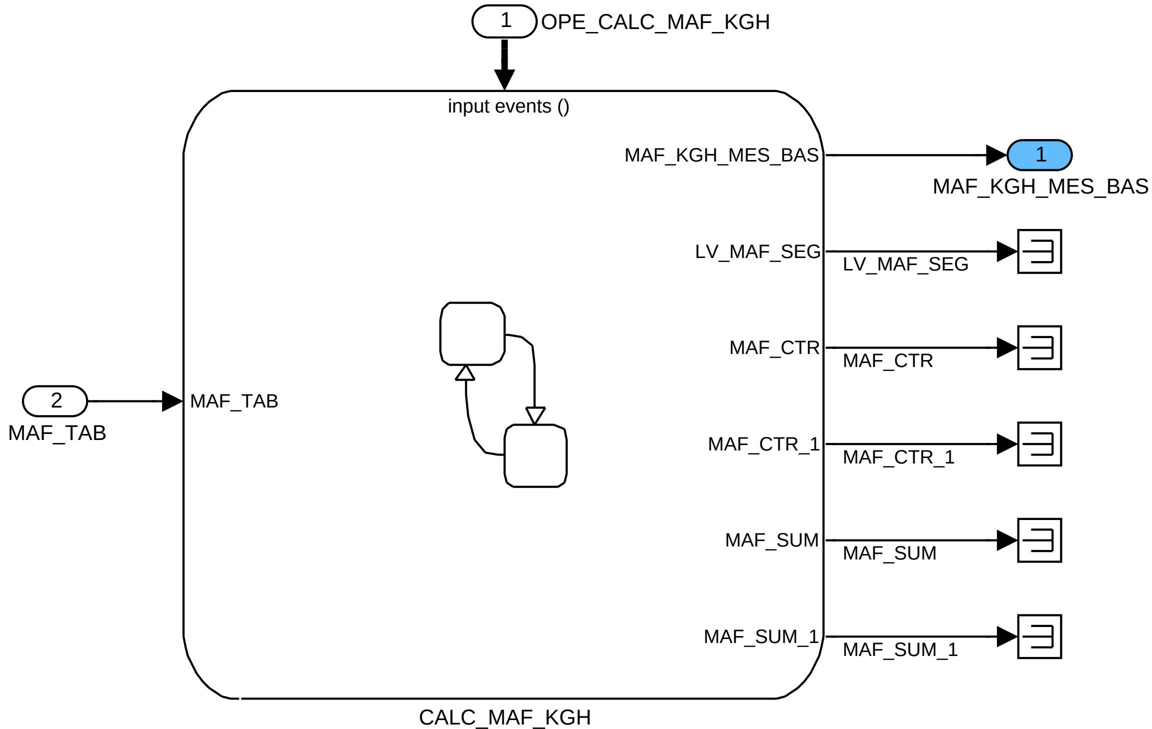


Figure 6 INSY_SIGCVMAF0/ operate/ CALC_MAF_SUM_MAF_KGH_MES_BAS

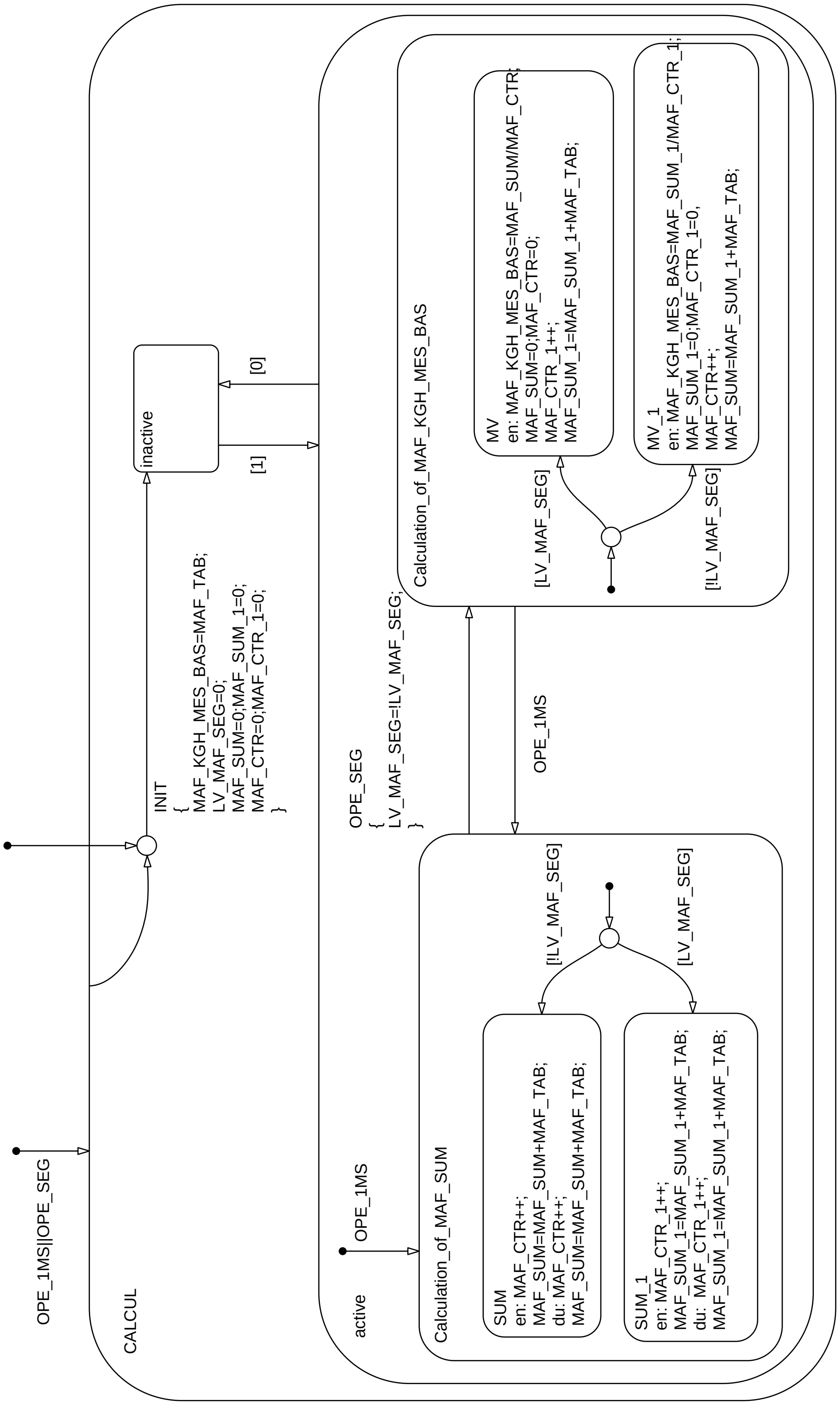


Figure 7 INSY_SIGCVMAF0/ operate/ CALC_MAF_SUM_MAF_KGH_MES_BAS/ CALC_MAF_KGH

Acquisition of mass air flow difference

General information:

For correction of oscillating MAF, the system needs the mass air flow differential that exists through a segment. To obtain this, the maximum and the minimum value - MAF_MAX and MAF_MIN - of MAF, converted by the table ID_MAF_TAB, during the actual segment will be stored in order to calculated the difference (module 30401E01).

Application Condition

Activation:	At every engine state
Deactivation:	-
Initialization:	At reset MAF_MAX=MAF_MAX_intern=0000h MAF_MIN=MAF_MIN_intern=0FFFFh
Calculation recurrence:	1ms If the engine is running on a fixed tooth each segment.

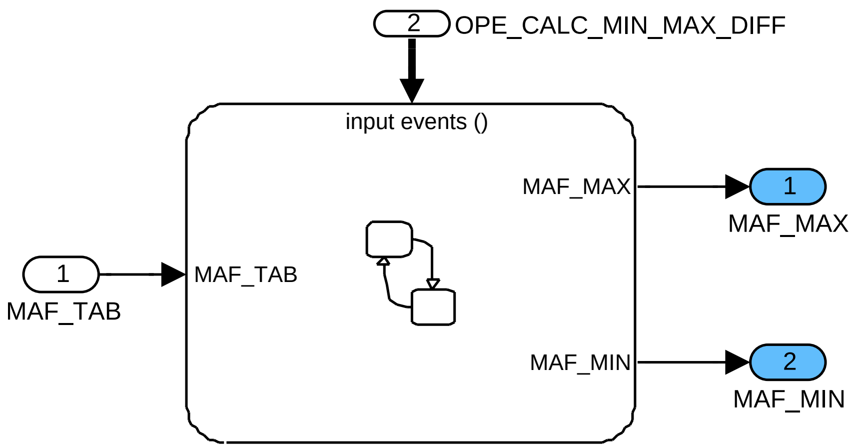


Figure 8 INSY_SIGCVMAF0/ operate/ ACQ_OF_MAF_DIFFERENCE

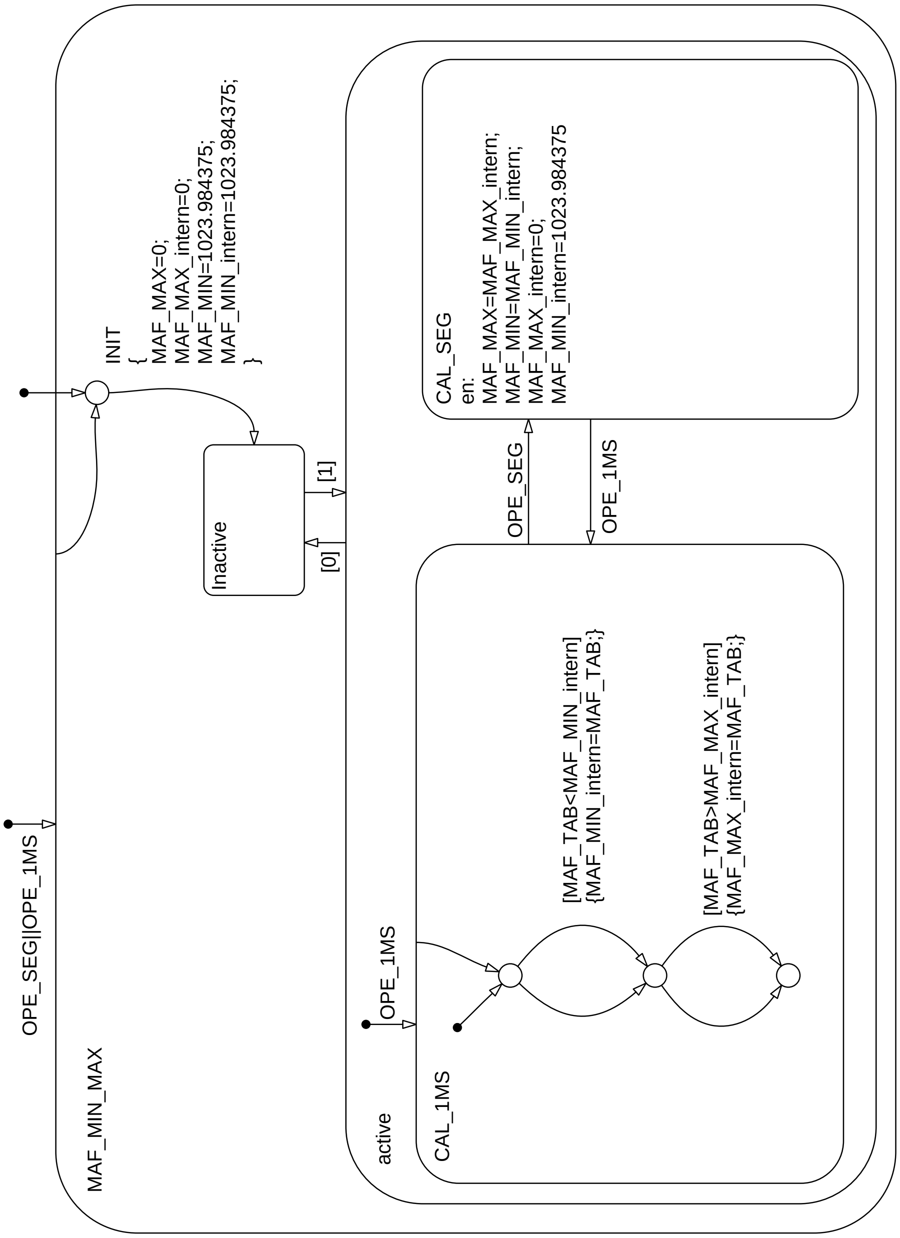


Figure 9 INSY_SIGCVMAF0/ operate/ ACQ_OF_MAF_DIFFERENCE/ CALC_MIN_MAX_DIFF

4.7 Determination of air pressures

Overview

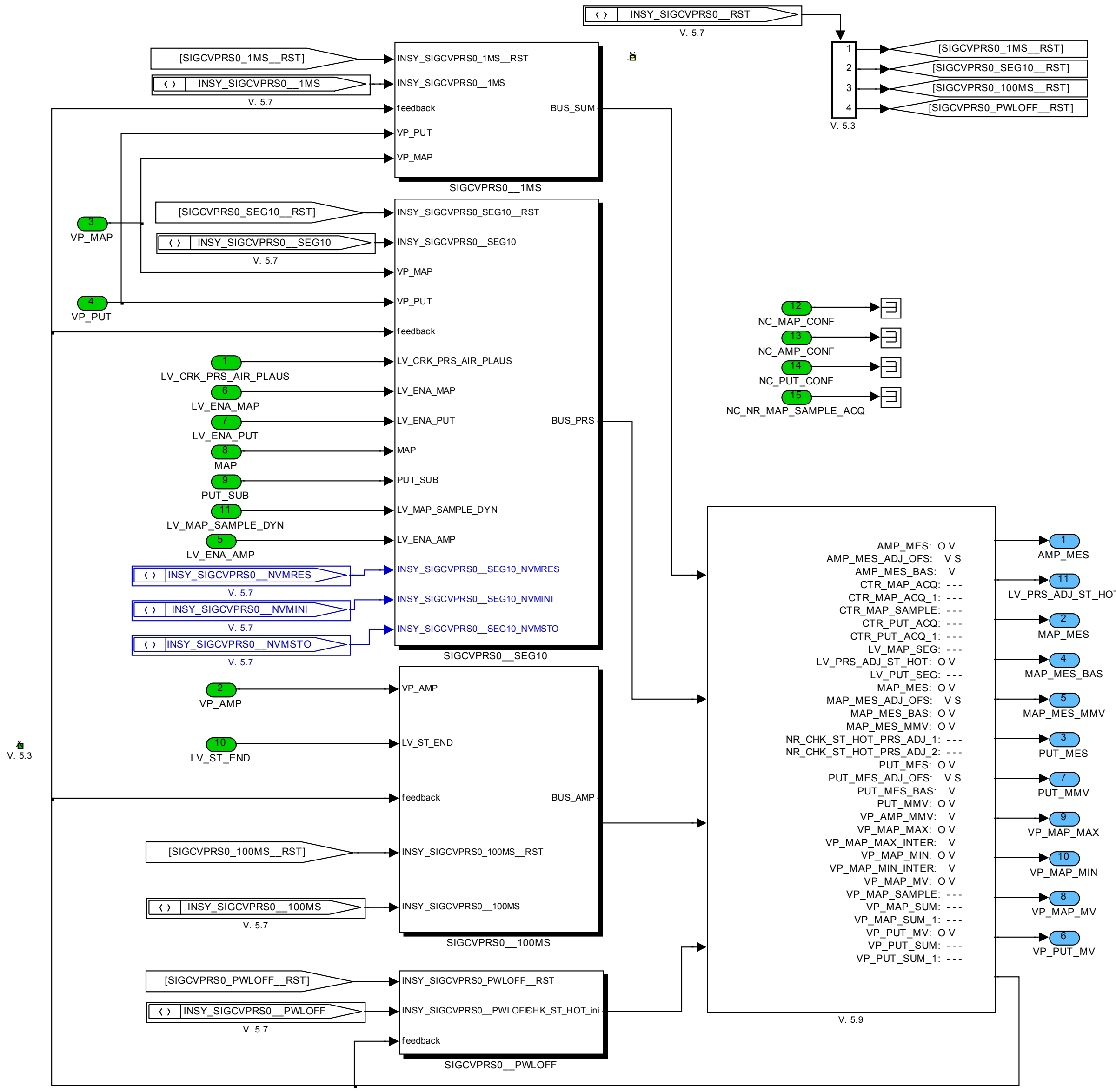


Figure 10:
Path: INSY_SIGCVPRS0

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
AMP_MES	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Measured ambient air pressure					
AMP_MES_ADJ_OFS	V/S	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Offset of AMP_MES to the pressure mean value at start (adjustment of the air pressures)					
AMP_MES_BAS	V	0... FFFFH	0... 5434	0.0829175	[hPa]
Measured ambient air pressure (raw value)					
LV_PRS_ADJ_ST_HOT	O/V	0... 1H	0... 1	1	[-]
Flag which shows whether a hot start was done during one ECU run					
MAP_MES	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Intake manifold pressure per segment measured					
MAP_MES_ADJ_OFS	V/S	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Offset of MAP_MES to the pressure mean value at start (adjustment of the air pressures)					
MAP_MES_BAS	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Intake manifold pressure per segment measured (raw value)					
MAP_MES_MMV	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Filtered MAP_MES					
PUT_MES	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
pressure upstream throttle per segment measured					
PUT_MES_ADJ_OFS	V/S	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Offset of PUT_MES to the pressure mean value at start (adjustment of the air pressures)					
PUT_MES_BAS	V	0... FFFFH	0... 5434	0.0829175	[hPa]
Intake manifold pressure up throttle per segment measured					
PUT_MMV	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
filtered value of pressure upstream throttle					
VP_AMP_MMV	V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Filtered VP_AMP					
VP_MAP_MAX	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Maximum manifold air pressure per segment in voltage					
VP_MAP_MAX_INTER	V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Internal maximum manifold air pressure per segment in voltage					
VP_MAP_MIN	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Minimum manifold air pressure per segment in voltage					
VP_MAP_MIN_INTER	V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Internal minimum manifold air pressure per segment in voltage					
VP_MAP_MV	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Mean value Voltage of the intake manifold pressure sensor (for diagnosis)					
VP_PUT_MV	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Mean value Voltage of the pressure upstream throttle					

Input Data:

LV_CRK_PRS_AIR_PLAUS	VP_AMP	VP_MAP	VP_PUT
LV_ENA_AMP	LV_ENA_MAP	LV_ENA_PUT	NC_NR_MAP_SAMPLE_AC Q
MAP	PUT_SUB	LV_ST_END	LV_MAP_SAMPLE_DYN

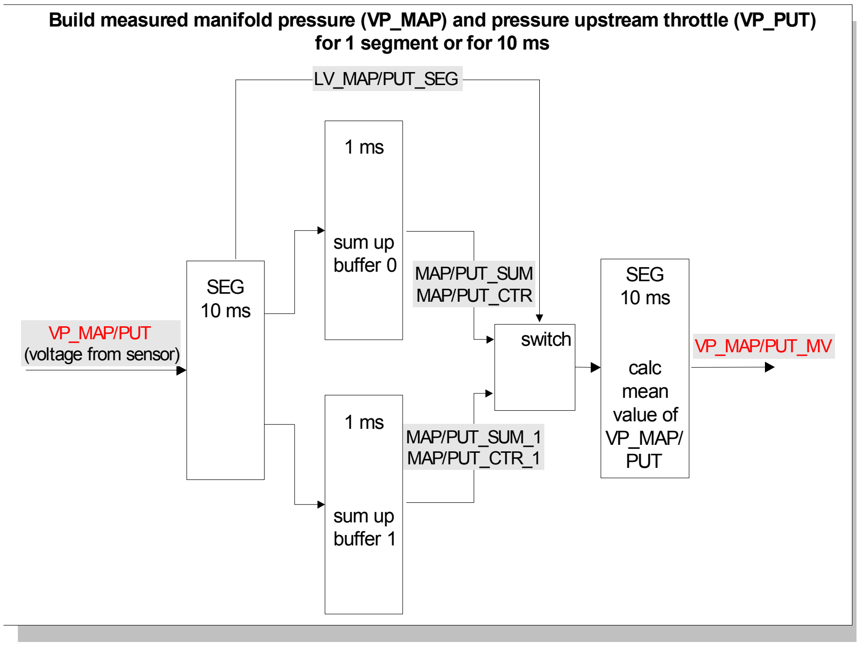
NC_MAP_CONF	NC_AMP_CONF	NC_PUT_CONF	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_AMP_LGRD_MAX	1	0... FFH	0... 21.144	0.0829176	[hPa/100ms]
Maximum change limitation of the Ambient pressure					
C_CRLC_MAP_MES_MMV	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Correlation constant for MAP_MES_MMV					
C_CRLC_PUT_MMV	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Filter constant for pressure upstream throttle calculation					
C_CRLC_VP_AMP_MMV	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Correlation constant for VP_AMP_MMV calculation					
IP_AMP_MES	2	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_VP_AMP_IP_AMP_MES	2	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Conversion characteristic for AMP_MES					
IP_MAP_MES	2	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_VP_MAP_MV_IP_MAP_MES	2	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Conversion characteristic for MAP_MES					
IP_PUT_MES	2	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_VP_PUT_MV_IP_PUT_MES	2	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Conversion characteristic for PUT_MES					
C_FAC_MAP_MES_ADJ_OFS	1	0... 78H	0... 120	1	[-]
Weighting factor for MAP sensor at pressure sensor adjustment					
C_FAC_AMP_MES_ADJ_OFS	1	0... 78H	0... 120	1	[-]
Weighting factor for AMP sensor at pressure sensor adjustment					
C_FAC_PUT_MES_ADJ_OFS	1	0... 78H	0... 120	1	[-]
Weighting factor for PUT sensor at pressure sensor adjustment					

General Information

This module includes the signal conversion of the sensor voltage into the measured pressure signal for all air pressure sensors. Also a pressure adjustment between all the used pressure sensors is done at engine standing before start.



The raw value (voltage) for MAP/PUT is measured by continuous conversion every 1ms. The values (10 bit) are summed up in two alternating buffers, MAP/PUT_SUM or MAP/PUT_SUM_1. The numbers of values are counted in comparable buffers MAP/PUT_CTR or MAP/PUT_CTR_1. This mechanism is necessary to synchronize the measurement and the calculation of MAP/PUT_MES (build mean value with a standardized range)
Depending on the logical variable LV_MAP/PUT_SEG one of the two buffers MAP/PUT_SUM or MAP/PUT_SUM_1 and the respective counter MAP/PUT_CTR or MAP/PUT_CTR_1 are selected for writing, the contents of the other buffer (counter) is used for calculations and diagnosis (after this buffer and the corresponding counter have been read they are cleared).
After having read out one buffer MAP/PUT_SUM or MAP/PUT_SUM_1 the buffer has to be cleared as well as the corresponding counter MAP/PUT_CTR or MAP/PUT_CTR_1.
A twobuffersystem is used to avoid incorrect MAP/PUT calculation.

Note: It must be made sure by SW that the 1ms task is called before the reset.

4.7.1 operate_1ms:

Function description

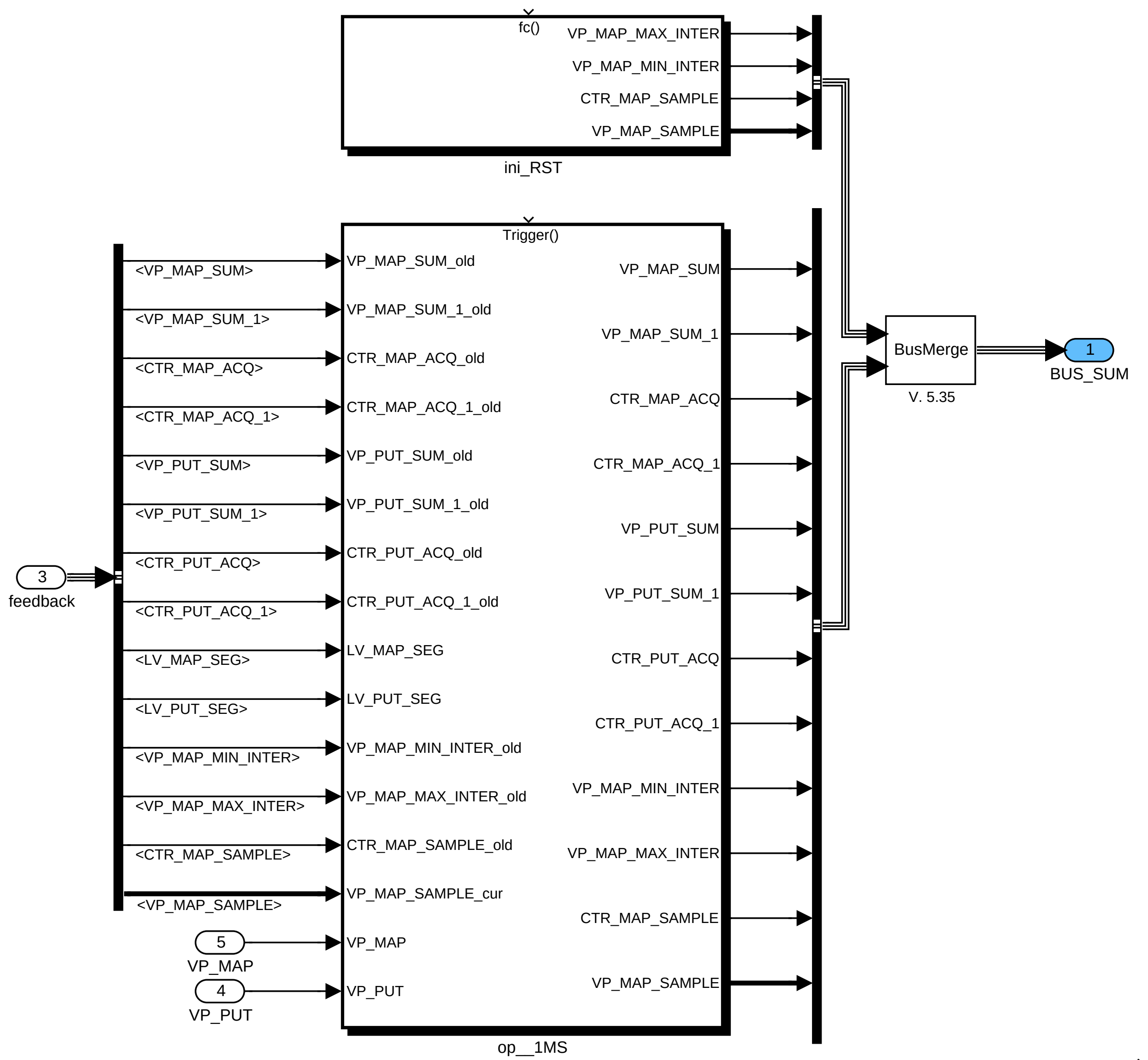


Figure 11:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS

4.7.1.1 Initialization at reset:

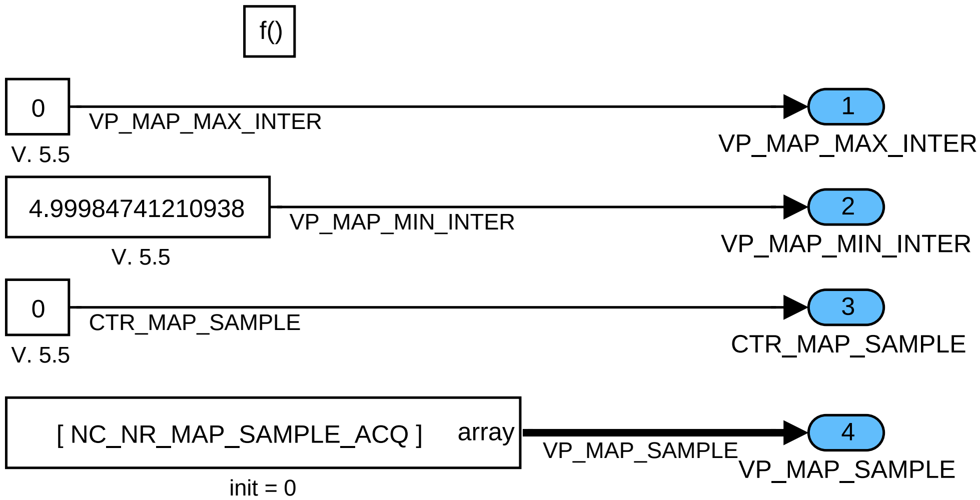


Figure 12:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/ini_RST

4.7.1.2 Calculation of VP_MAP_SUM and VP_PUT_SUM depending on system configuration:

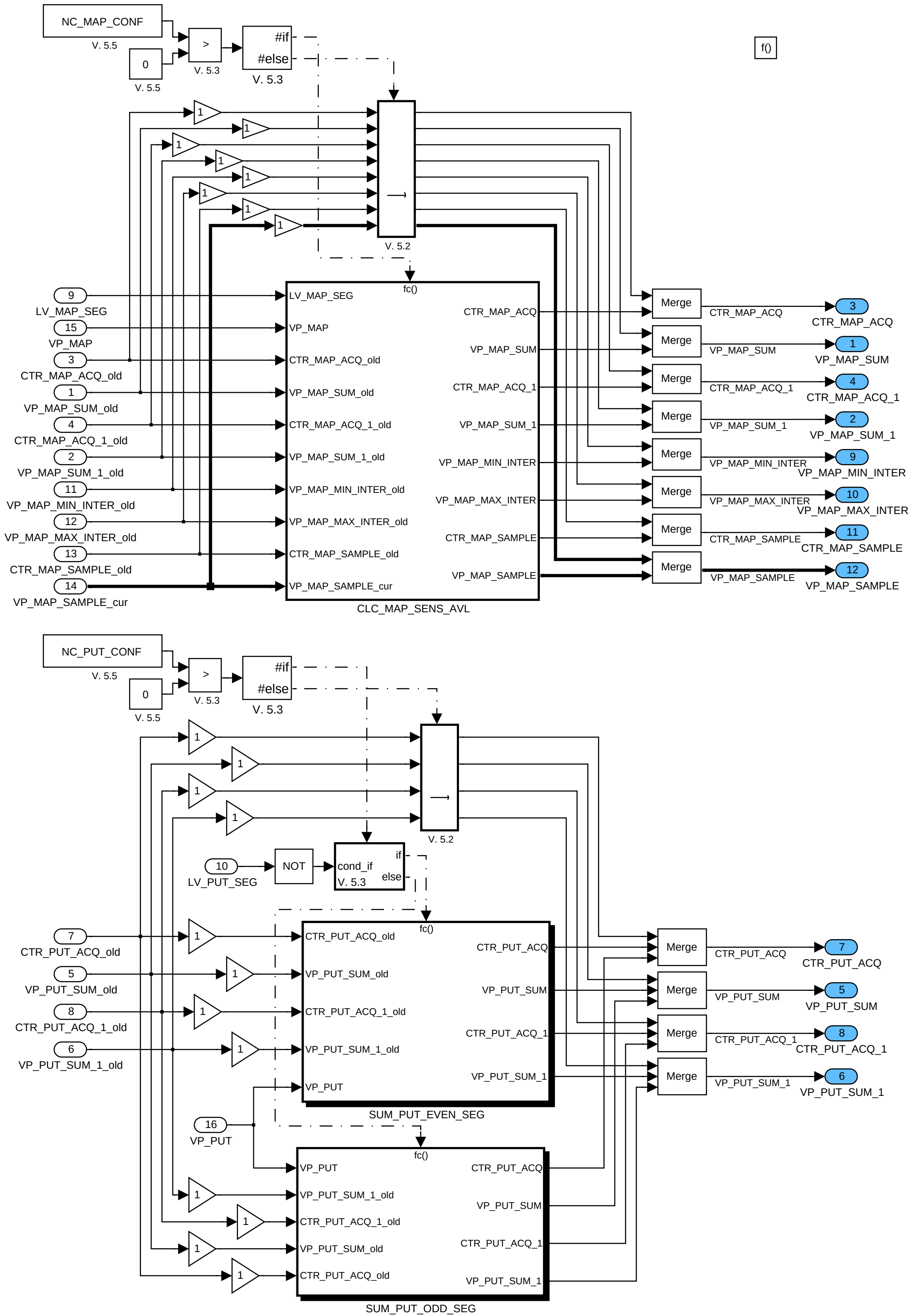


Figure 13:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS

4.7.1.2.1 Calculation of VP_MAP_SUM and CTR_MAP_ACQ for two alternating buffers:

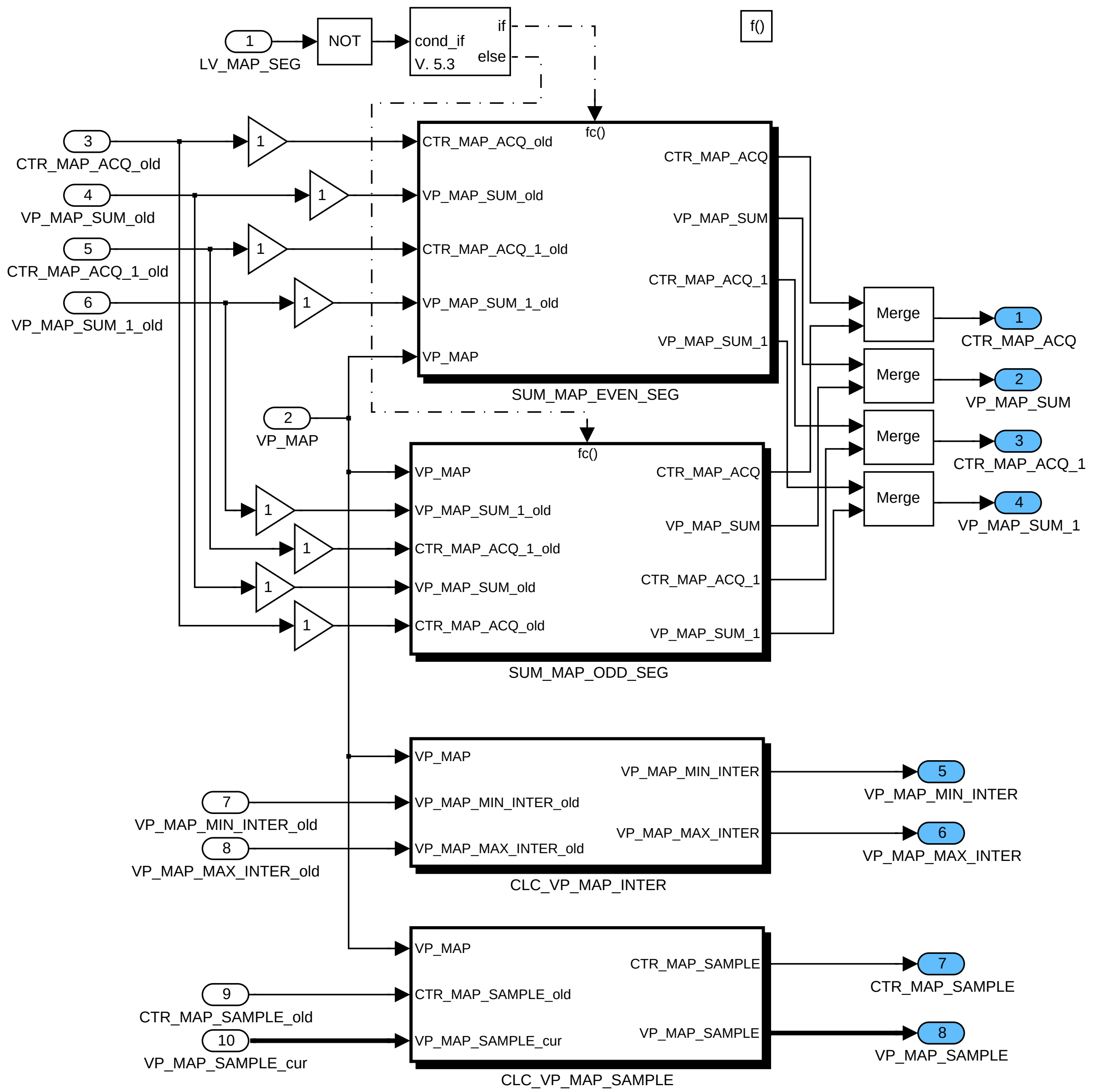


Figure 14:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/CLC_MAP_SENS_AVL

4.7.1.2.1.1 Calculation of VP_MAP_SUM and CTR_MAP_ACQ in buffer 1:

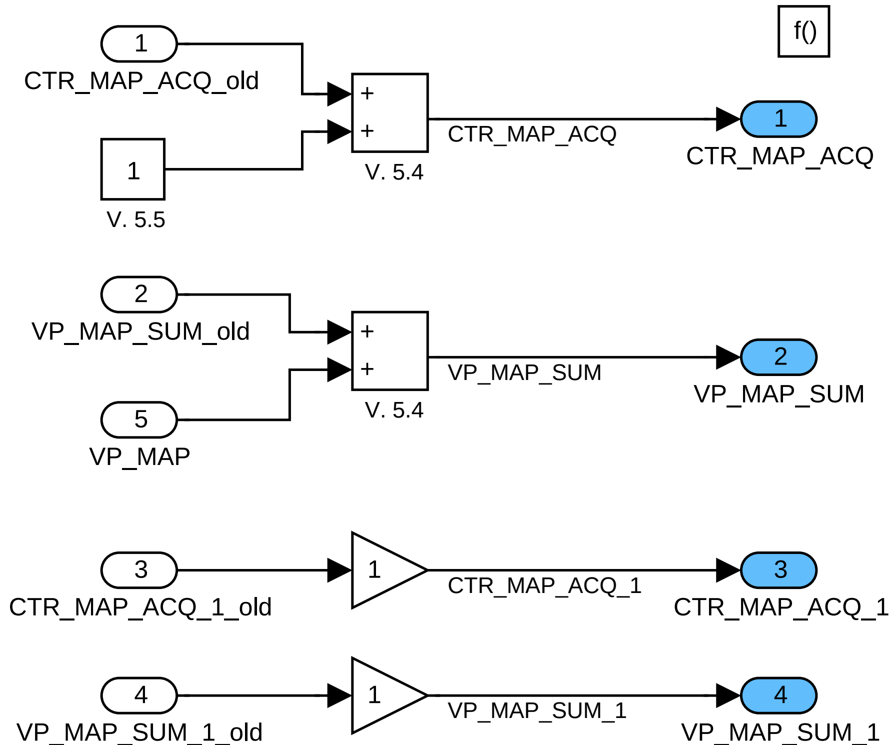


Figure 15:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/CLC_MAP_SENS_AVL/SUM_MAP_EVEN_SEG

4.7.1.2.1.2 Calculation of VP_MAP_SUM_1 and CTR_MAP_ACQ_1 in buffer 2:

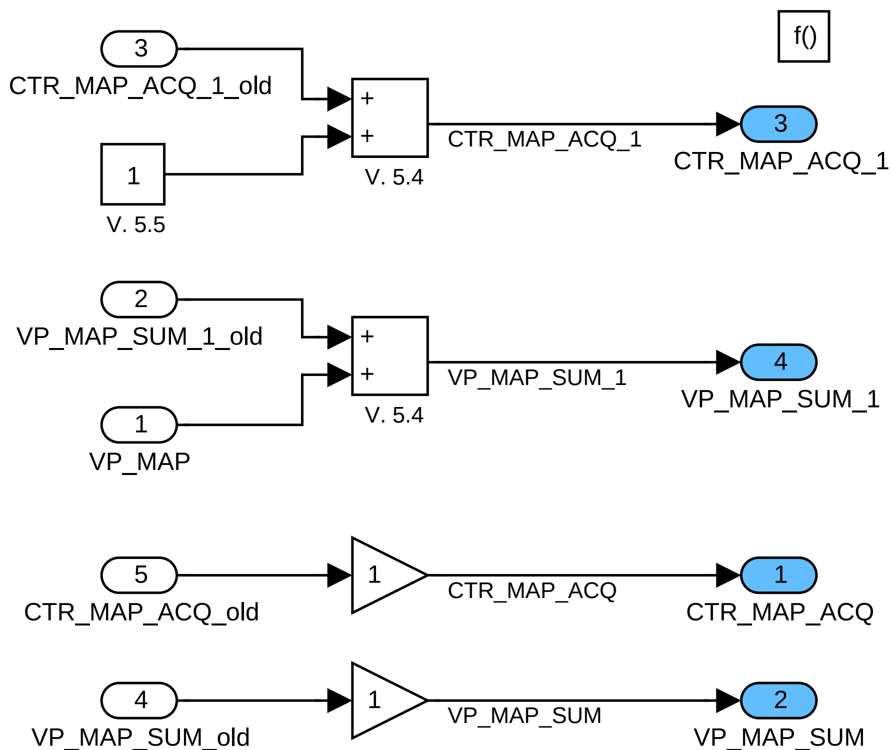


Figure 16:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/CLC_MAP_SENS_AVL/SUM_MAP_ODD_SEG

4.7.1.2.1.3 Calculation of VP_MAP_MAX/MIN_INTER:

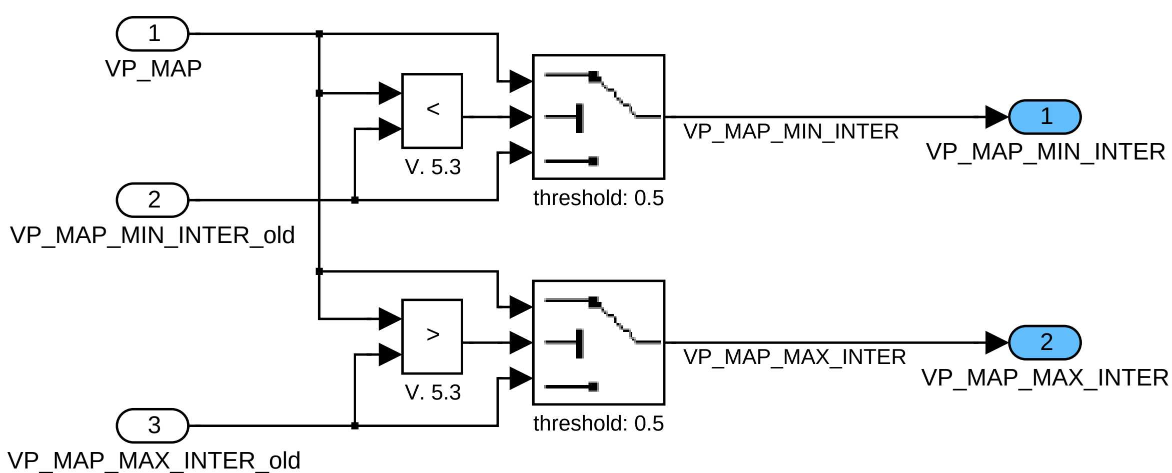


Figure 17:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/CLC_MAP_SENS_AVL/CLC_VP_MAP_INTER

4.7.1.2.1.4 Calculation of VP_MAP_SAMPLE:

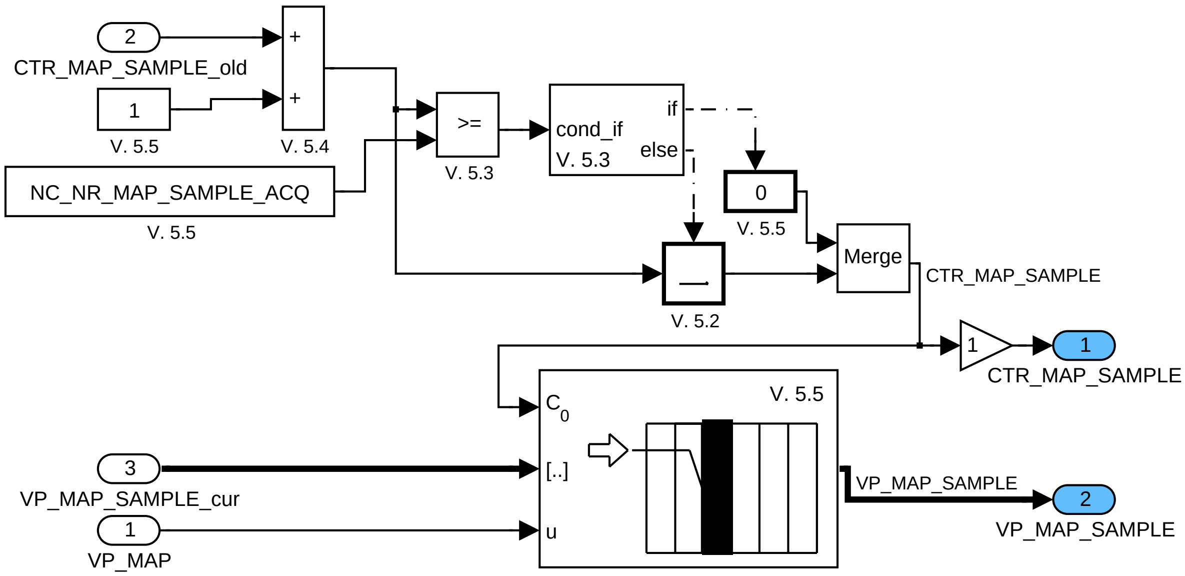


Figure 18:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/CLC_MAP_SENS_AVL/CLC_VP_MAP_SAMPLE

4.7.1.2.2 Calculation of VP_PUT_SUM and CTR_PUT_ACQ in buffer 1:

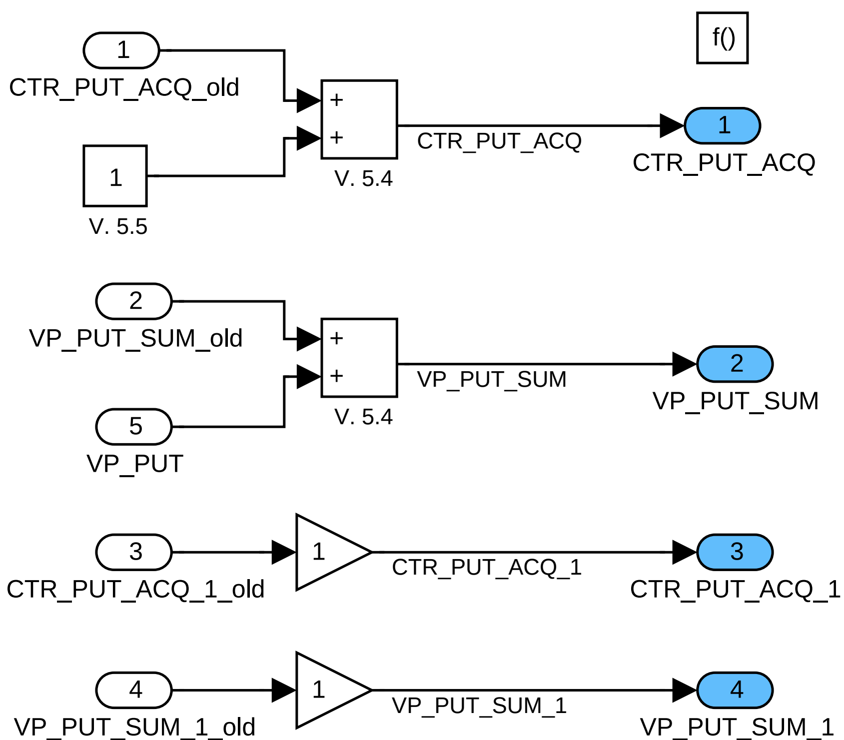


Figure 19:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/SUM_PUT_EVEN_SEG

4.7.1.2.3 Calculation of VP_PUT_SUM_1 and CTR_PUT_ACQ_1 in buffer 2:

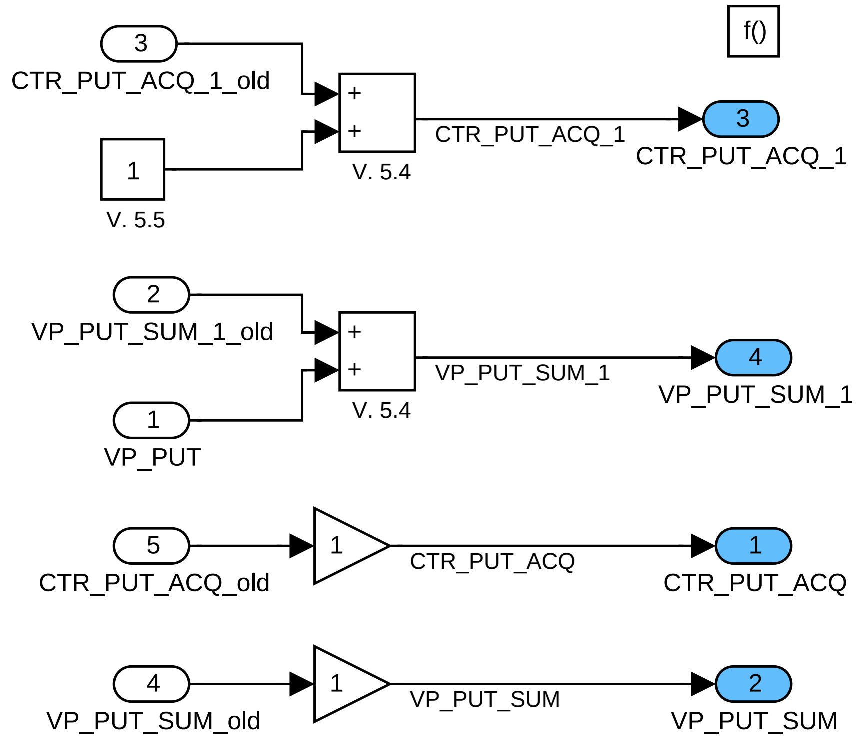


Figure 20:
Path: INSY_SIGCVPRS0/SIGCVPRS0__1MS/op__1MS/SUM_PUT_ODD_SEG

4.7.2 operate_10ms (LV_ES = 1) and operate_SEG (LV_ES = 0):

Function description

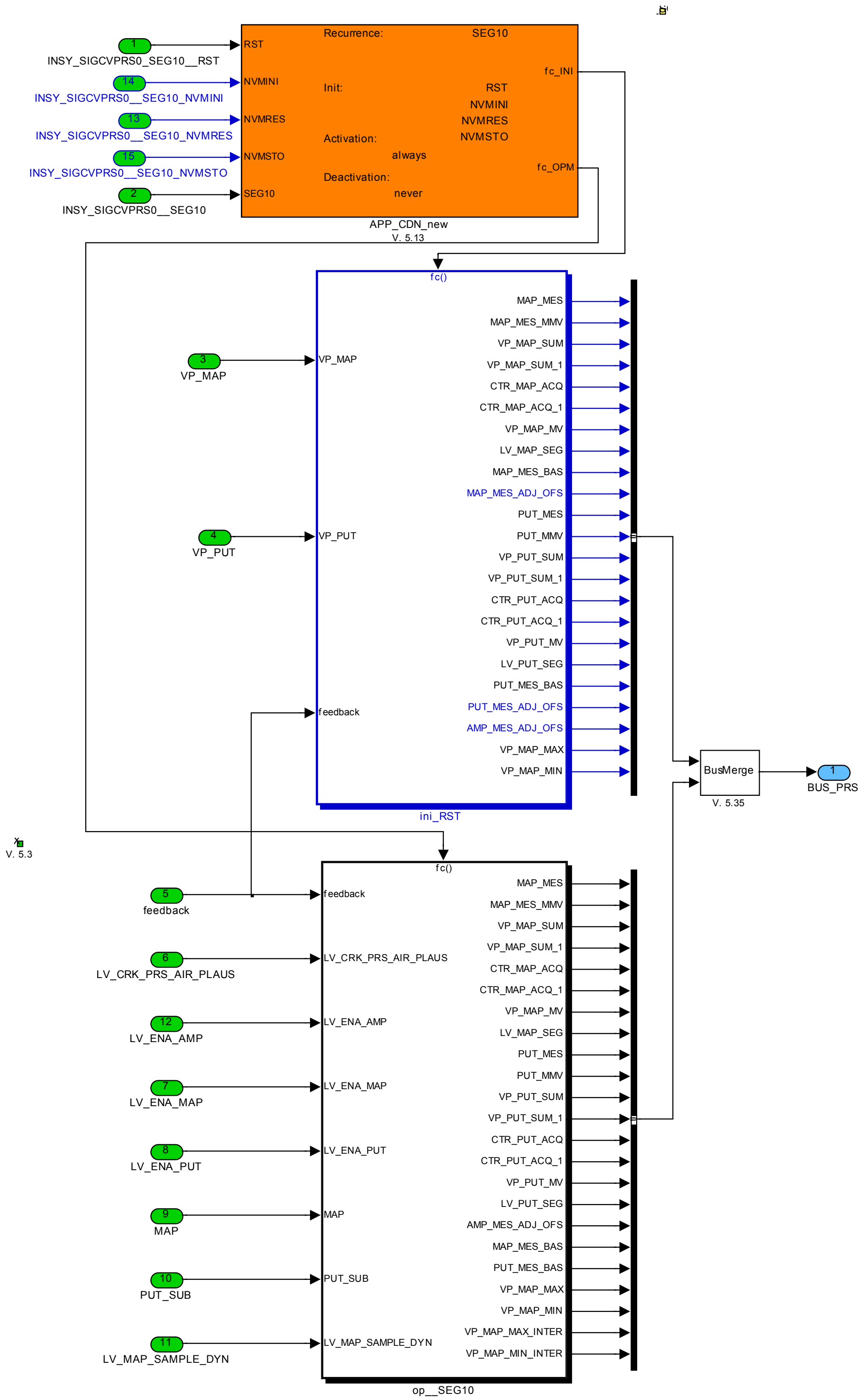


Figure 21:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10

4.7.2.1 Initialization:

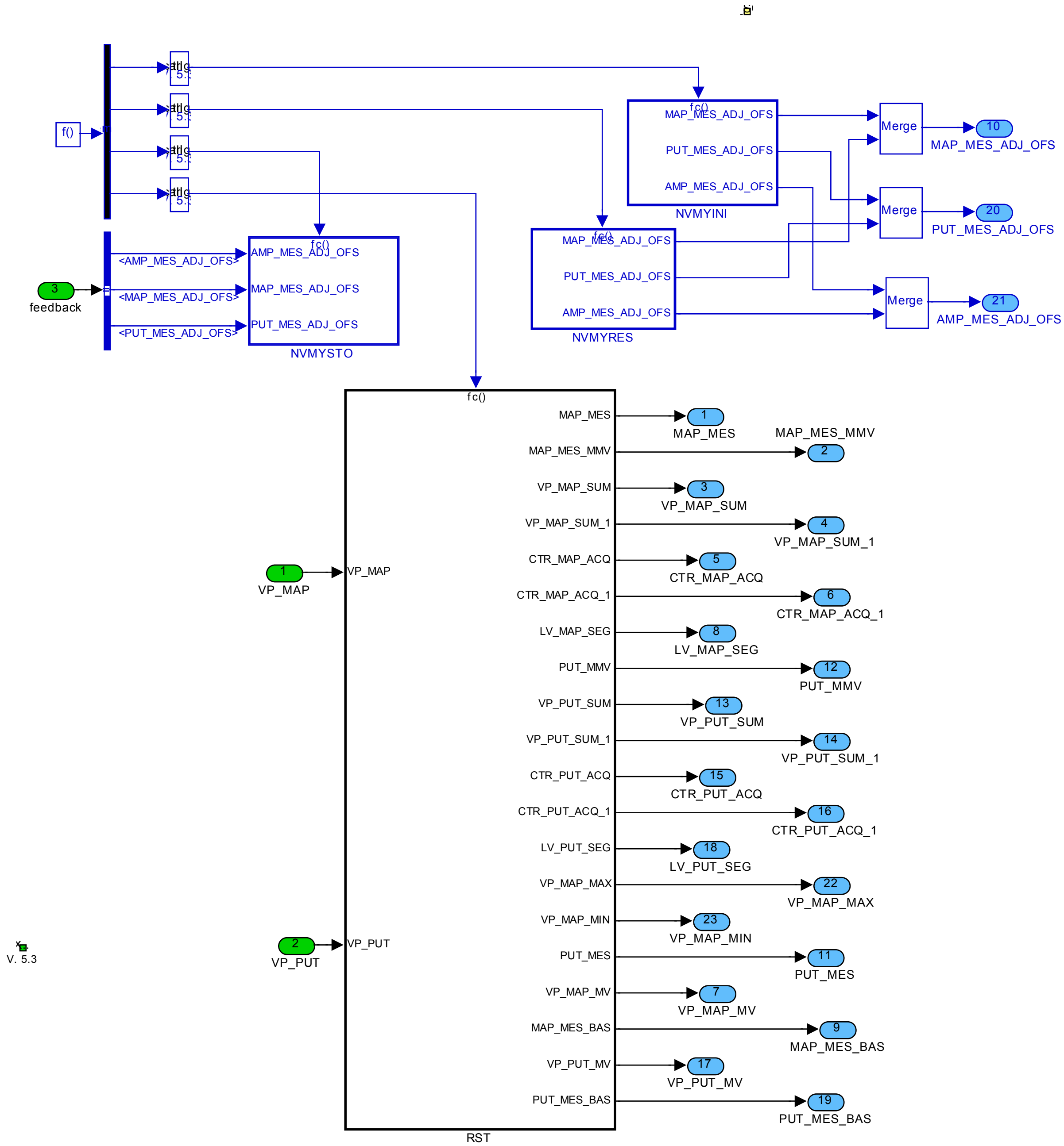


Figure 22:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/ini_RST

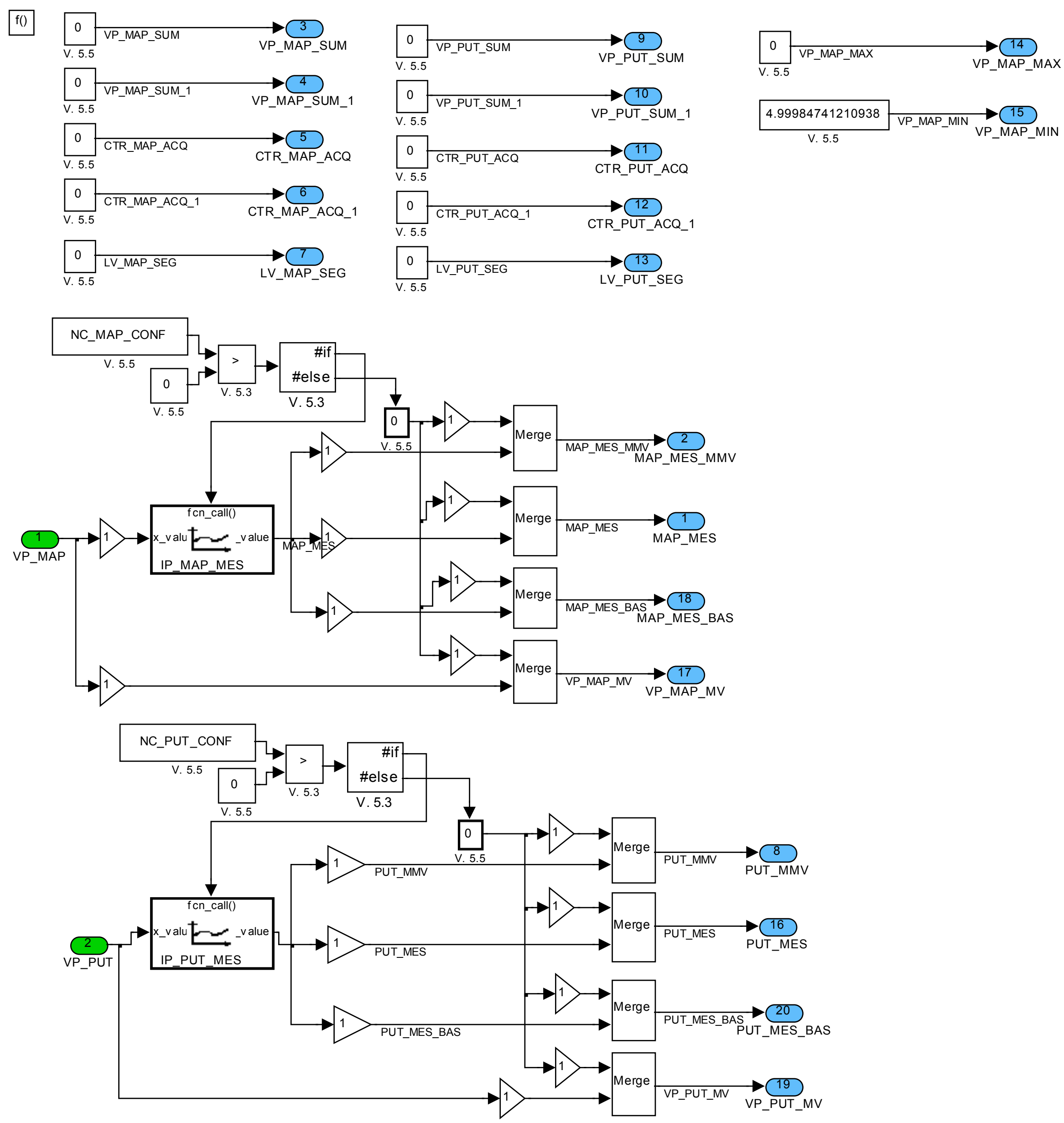


Figure 23:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/ini_RST/RST

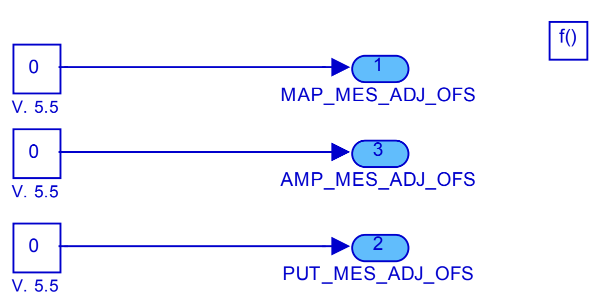


Figure 24:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/ini_RST/NVMYINI

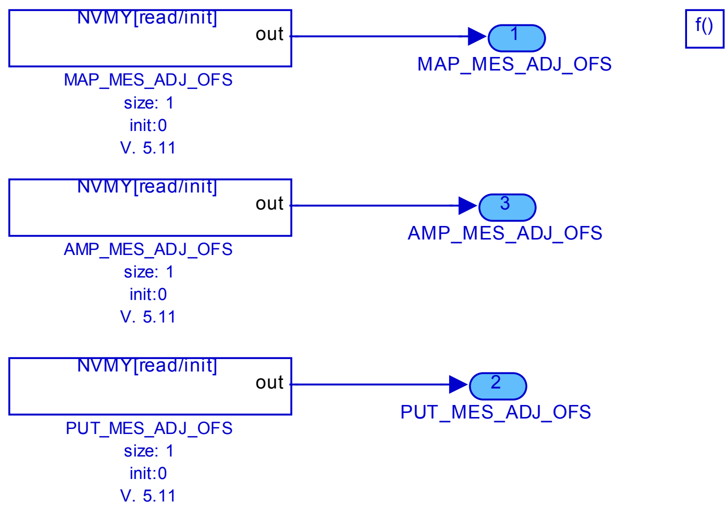


Figure 25:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/ini_RST/NVMYRES

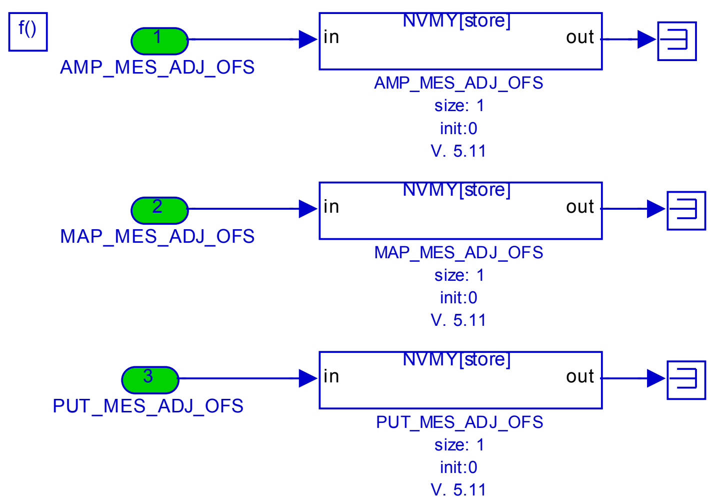


Figure 26:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/ini_RST/NVMYSTO

4.7.2.2 operate_SEG10:

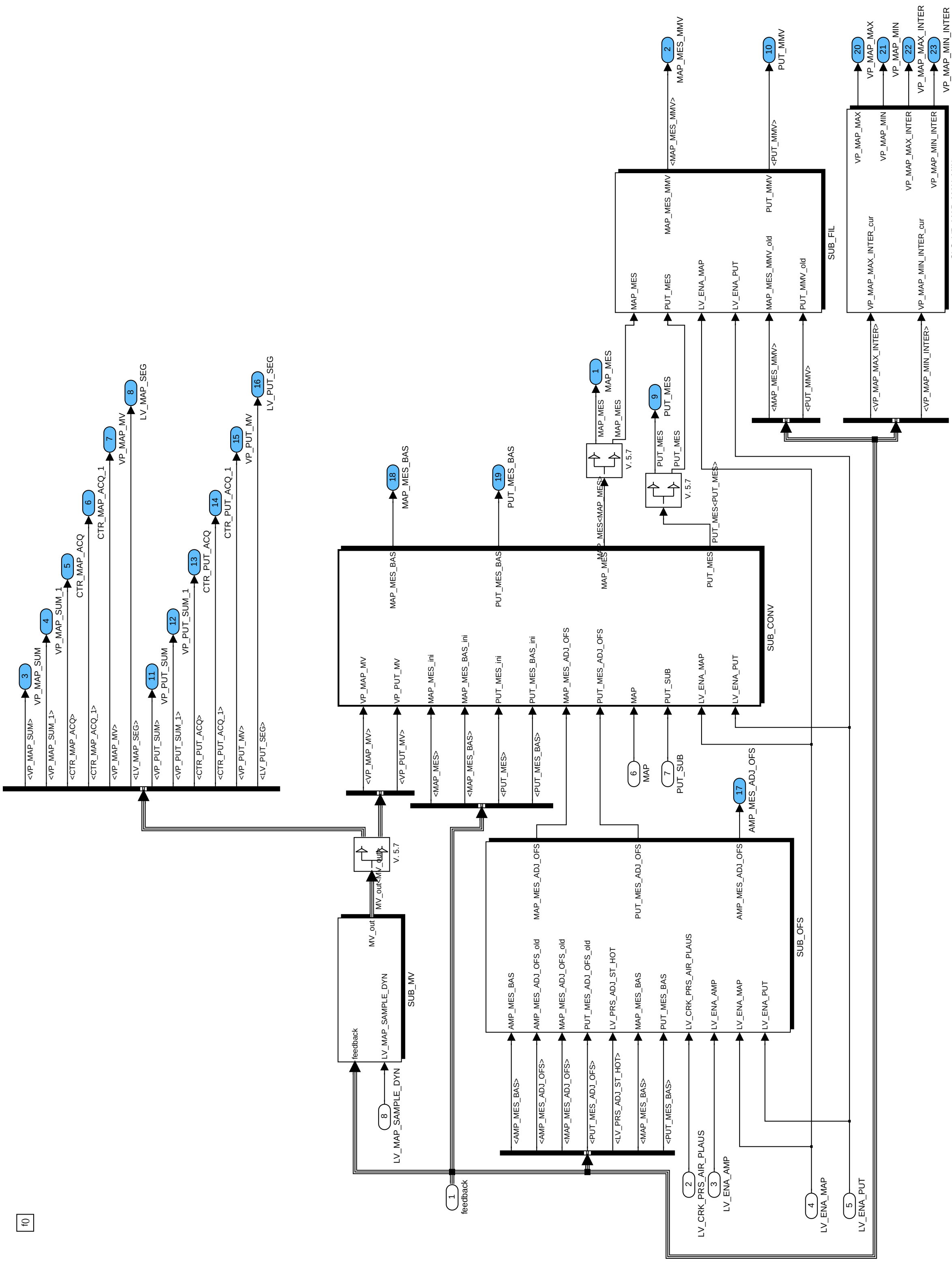


Figure 27:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10

4.7.2.2.1 Calculation of voltage mean value over 10ms (LV_ES = 1) or one SEG (LV_ES = 0):

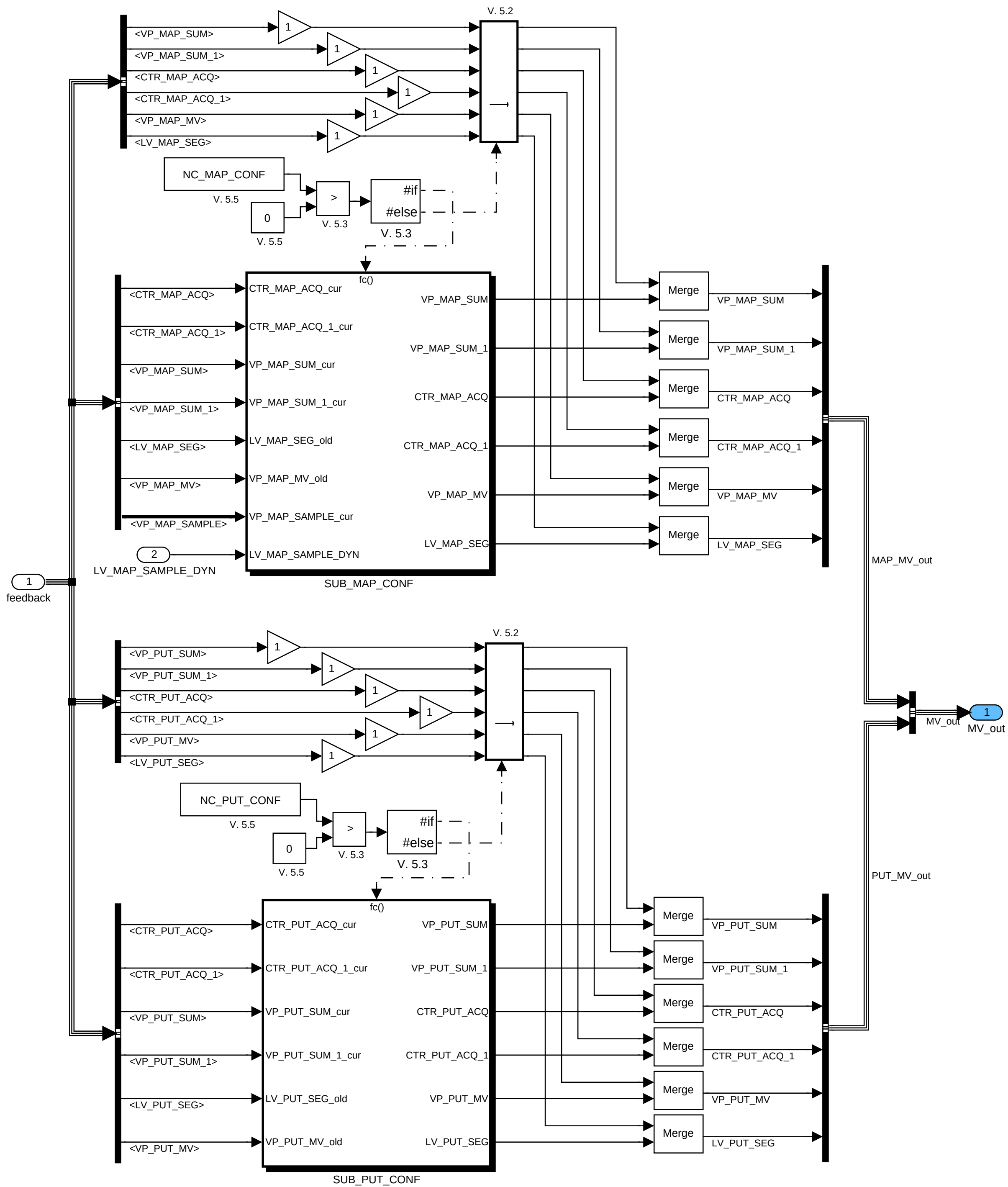


Figure 28:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV

4.7.2.2.1.1 Calculation of VP_MAP_MV:

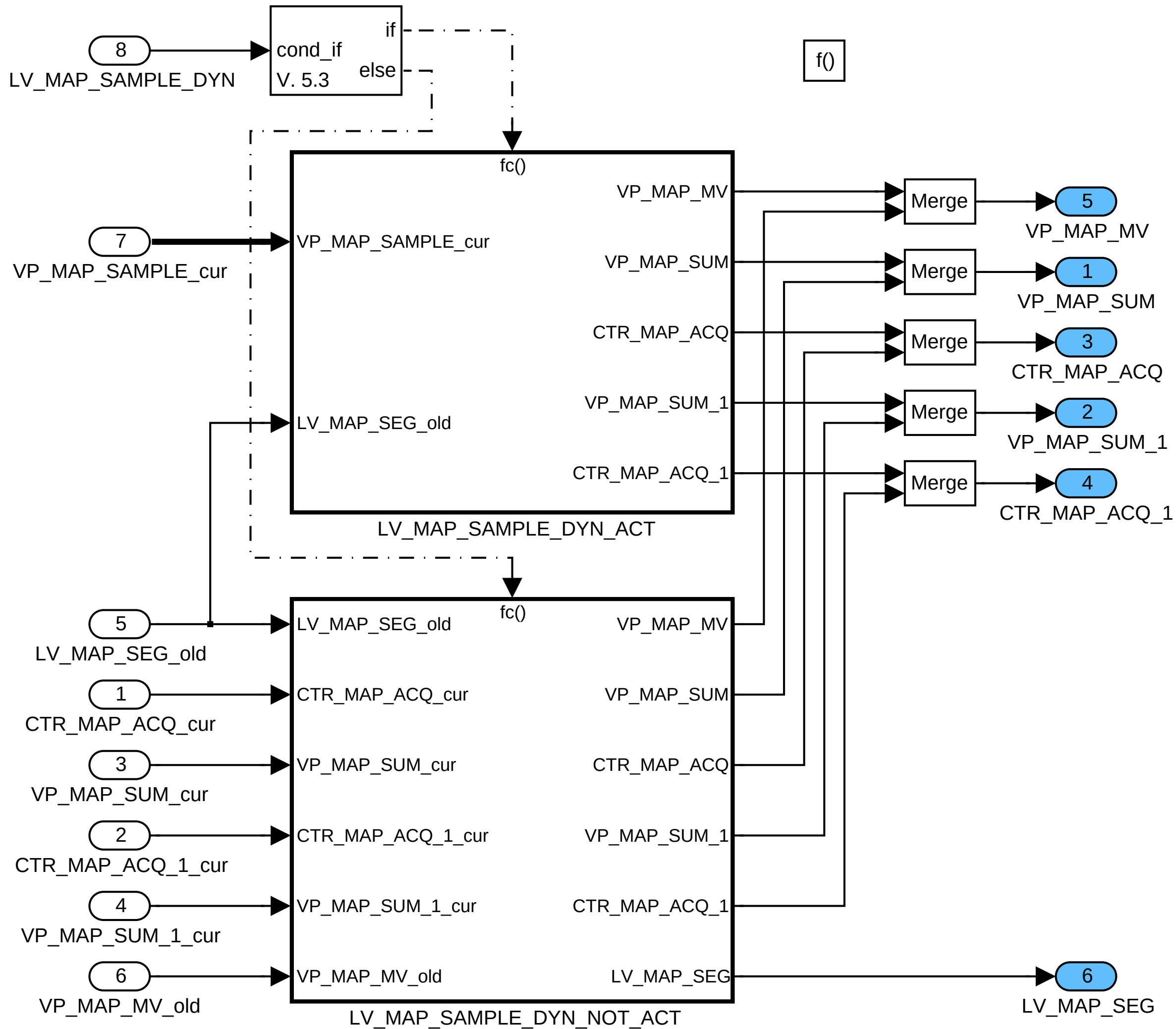


Figure 29:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_MAP_CONF

4.7.2.2.1.1.1 Calculation of VP_MAP_MV if LV_MAP_SAMPLE_DYN = 1:

It is possible for transient operating points only to use the last defined number (= NC_NR_MAP_SAMPLE_ACQ) of samples. Through this the pressure change is included more quickly and the dynamics of the intake manifold model increases.

This mechanism is activated only at transient engine operation. The controlbit LV_MAP_SAMPLE_DYN is set in the Appl.Inc.

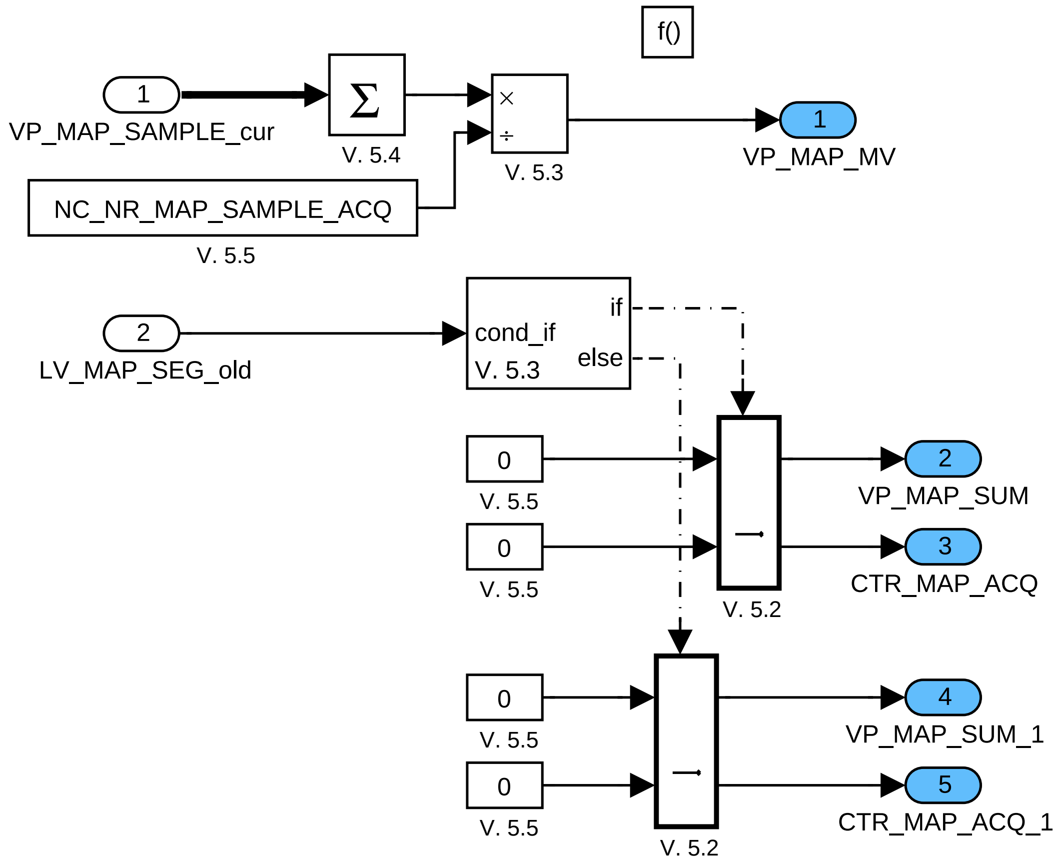


Figure 30:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_MAP_CONF/LV_MAP_SAMPLE_DYN_ACT

4.7.2.2.1.1.2 Calculation of VP_MAP_MV if LV_MAP_SAMPLE_DYN = 0:

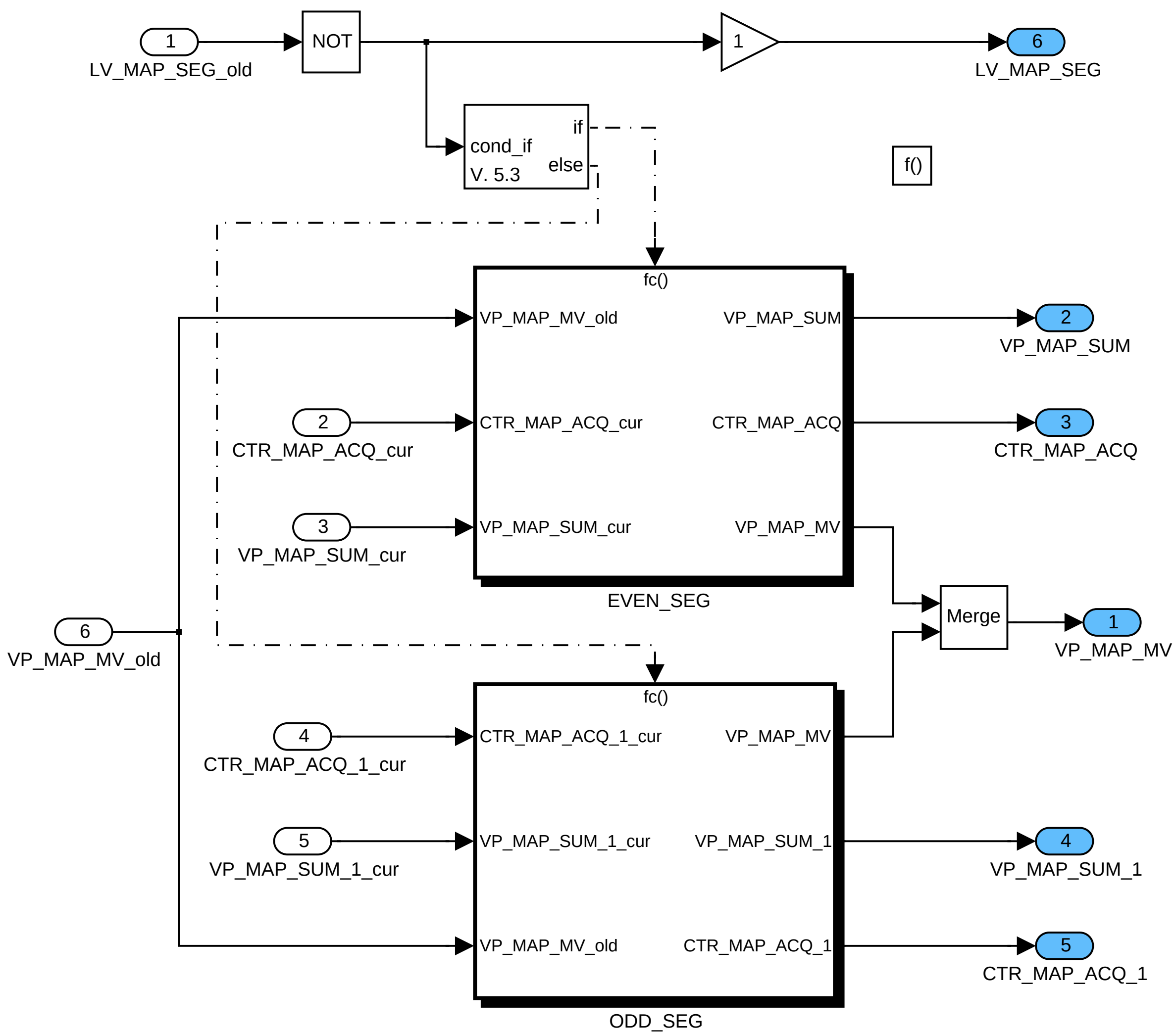


Figure 31:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_MAP_CONF/LV_MAP_SAMPLE_DYN_NOT_ACT

4.7.2.2.1.1.2.1 Calculation of VP_MAP_MV - read out buffer 1:

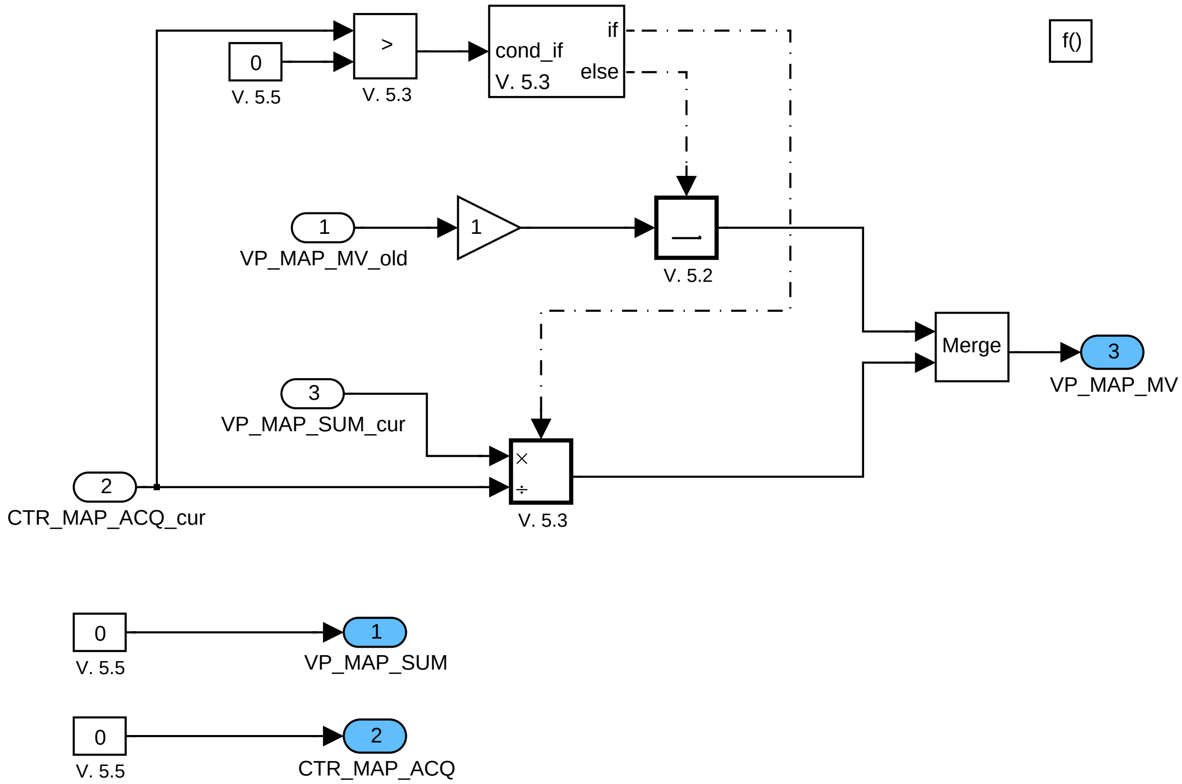


Figure 32:
Path:
INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_MAP_CONF/LV_MAP_SAMPLE_DYN_NOT_ACT/EVEN_SEG

4.7.2.2.1.1.2.2 Calculation of VP_MAP_MV - read out buffer 2:

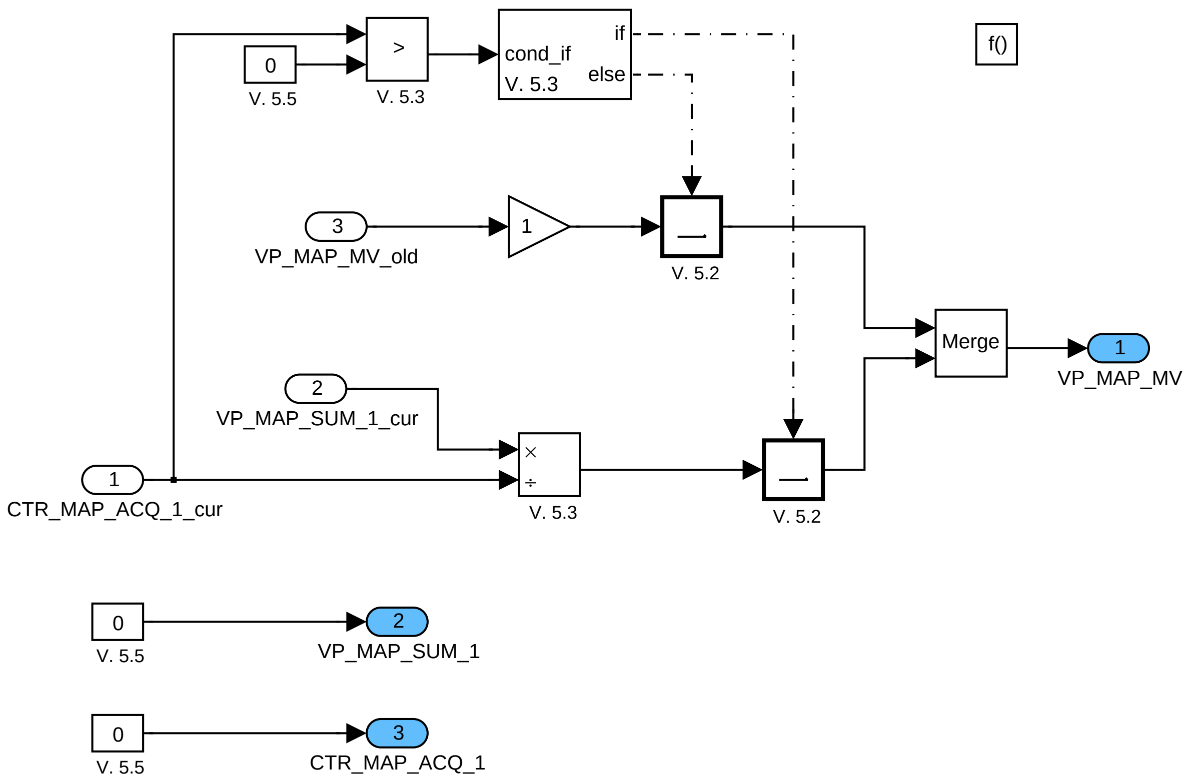


Figure 33:
Path:
INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_MAP_CONF/LV_MAP_SAMPLE_DYN_NOT_ACT/ODD_SEG

4.7.2.2.1.2 Calculation of VP_PUT_MV:

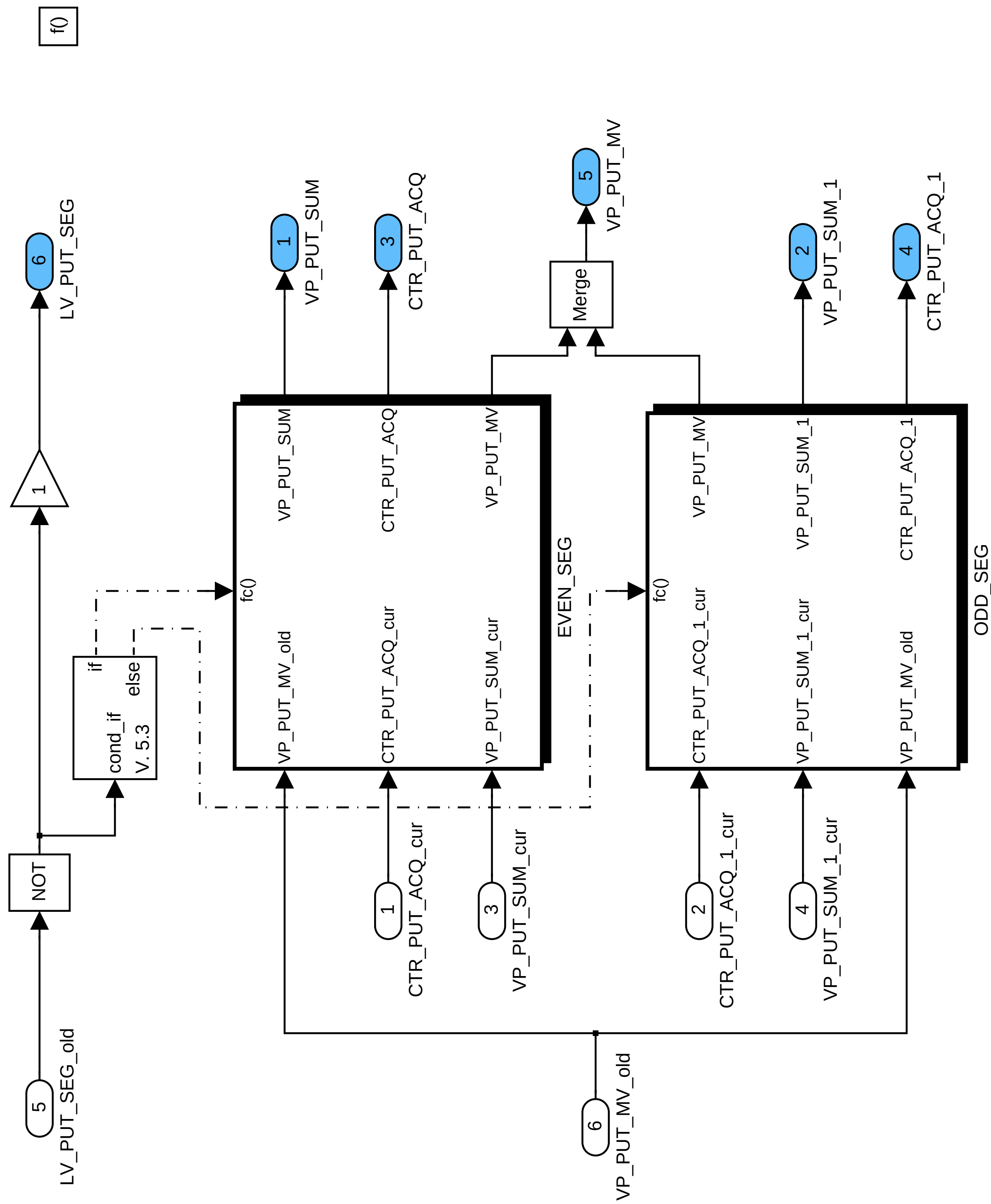


Figure 34:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_PUT_CONF

4.7.2.2.1.2.1 Calculation of VP_PUT_MV - read out buffer 1:

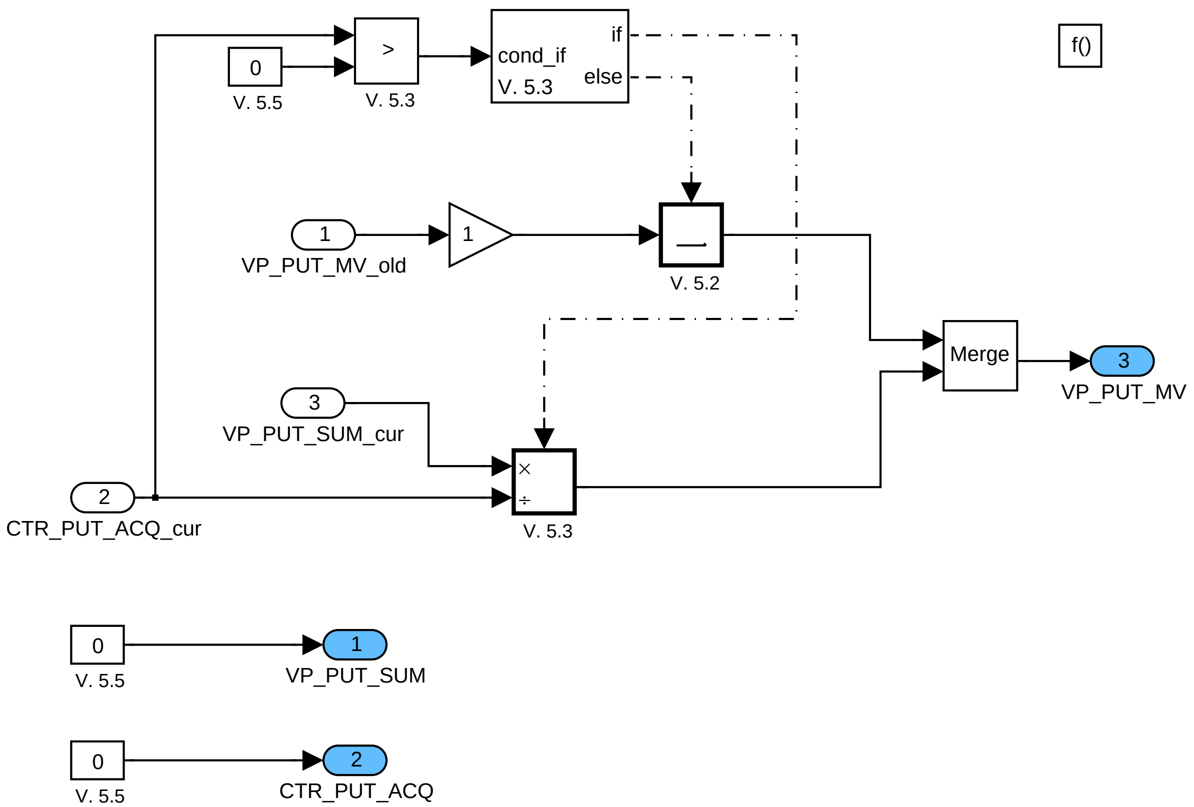


Figure 35:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_PUT_CONF/EVEN_SEG

4.7.2.2.1.2.2 Calculation of VP_PUT_MV - read out buffer 2:

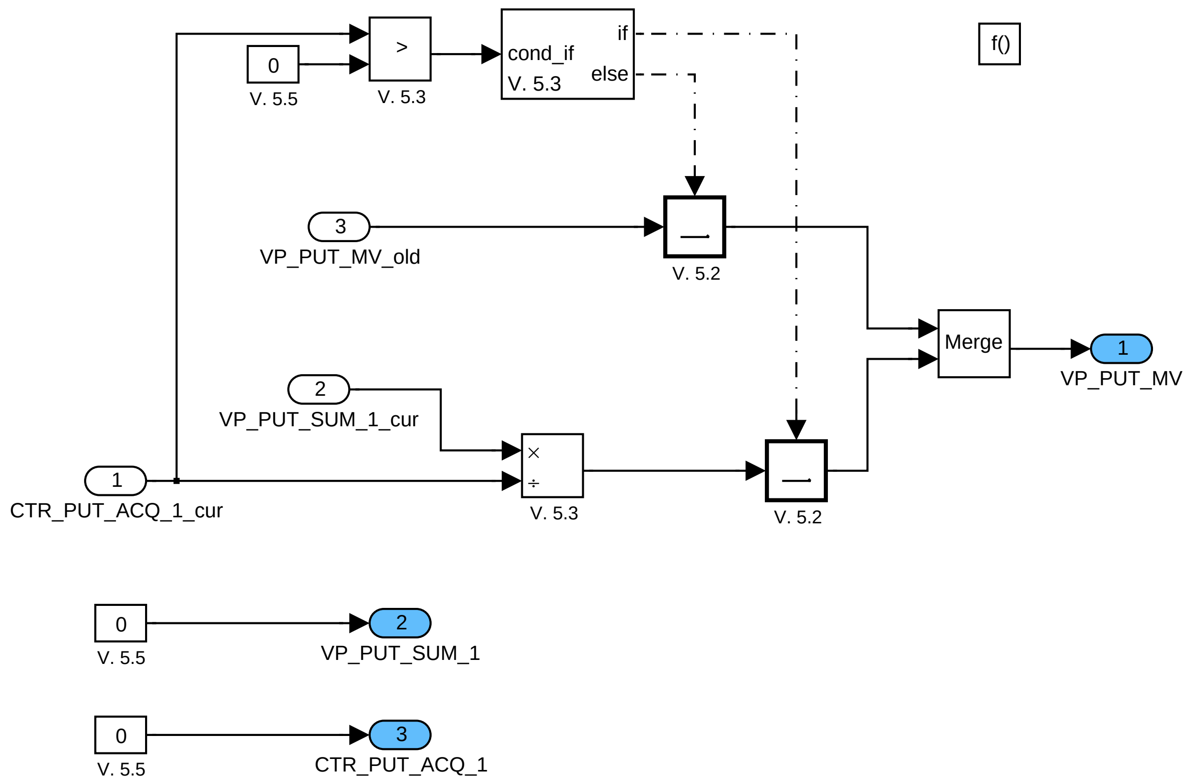


Figure 36:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_MV/SUB_PUT_CONF/ODD_SEG

4.7.2.2.2 Calculation of the adjustment offset for each pressure sensor:

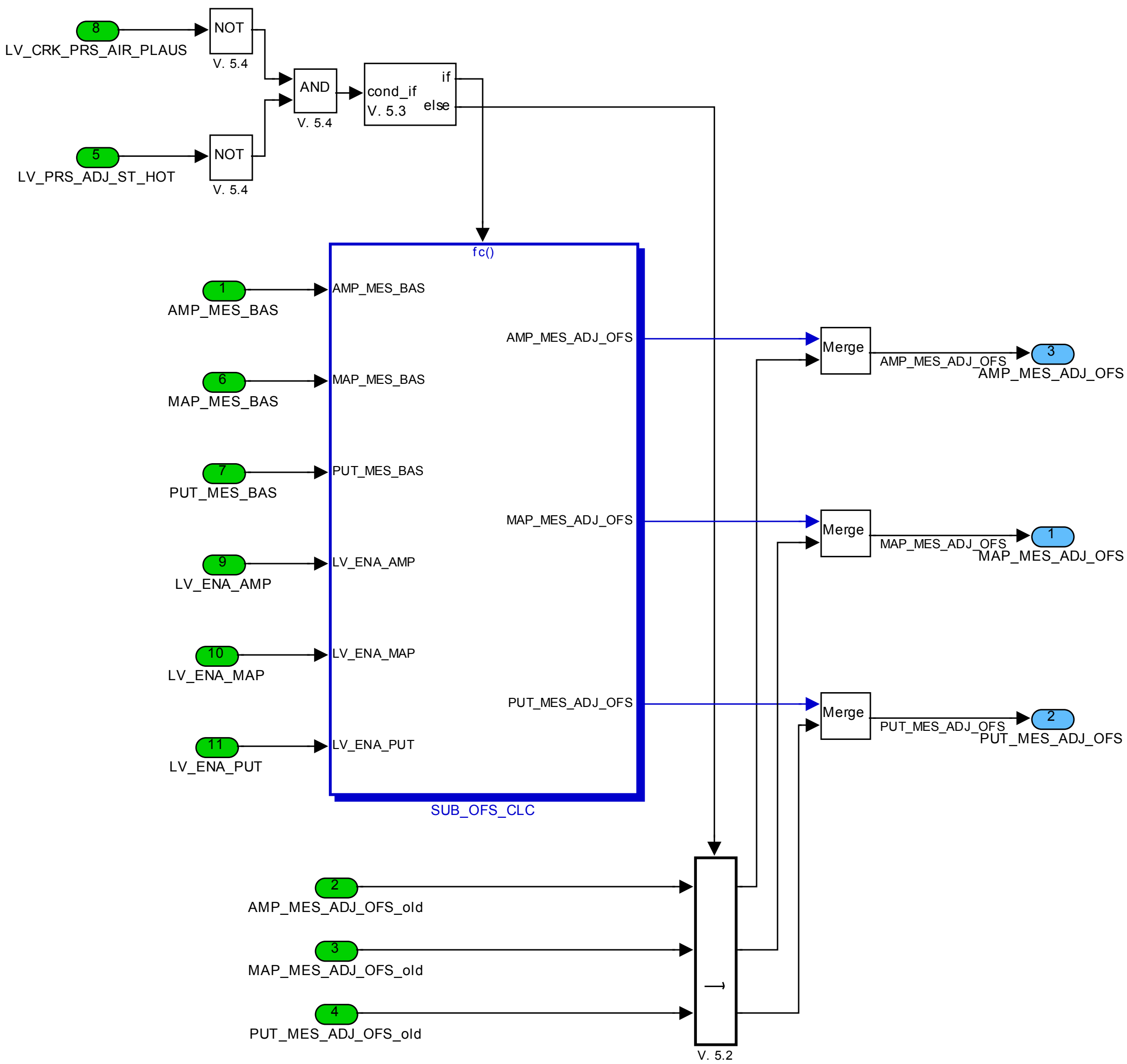


Figure 37:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_OFS

4.7.2.2.2.1 Calculation of the adjustment offset for each pressure sensor - depending on system configuration:

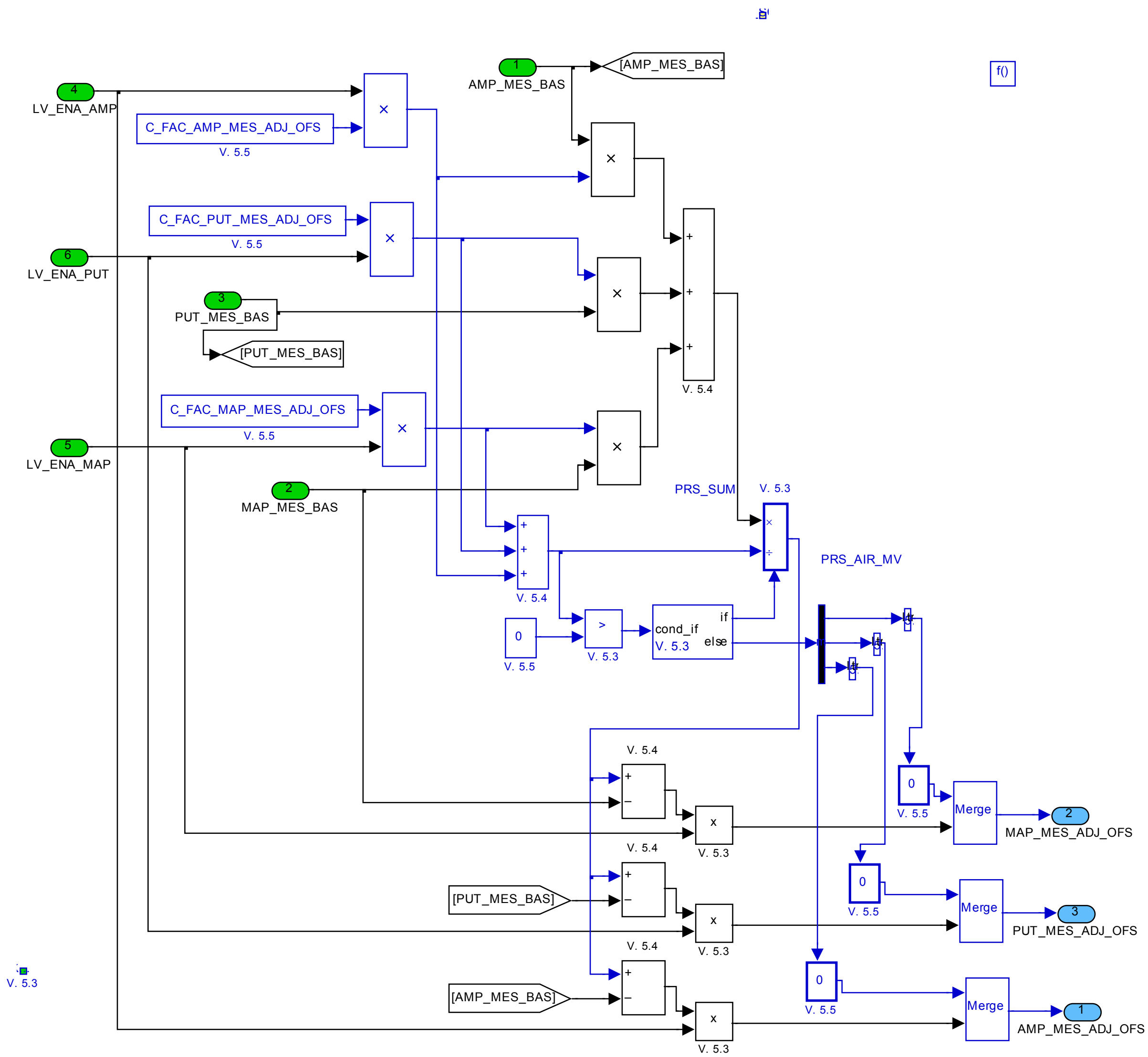
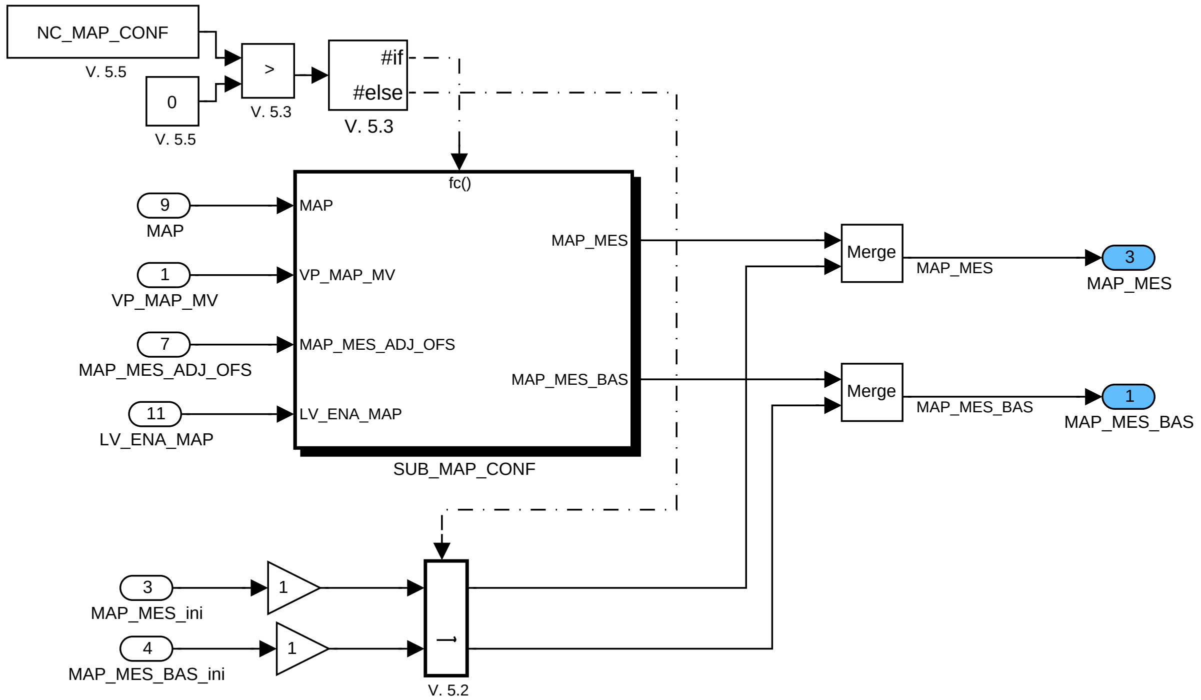


Figure 38:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_OFS/SUB_OFS_CLC

4.7.2.2.3 Conversion of sensor voltage into a pressure signal:



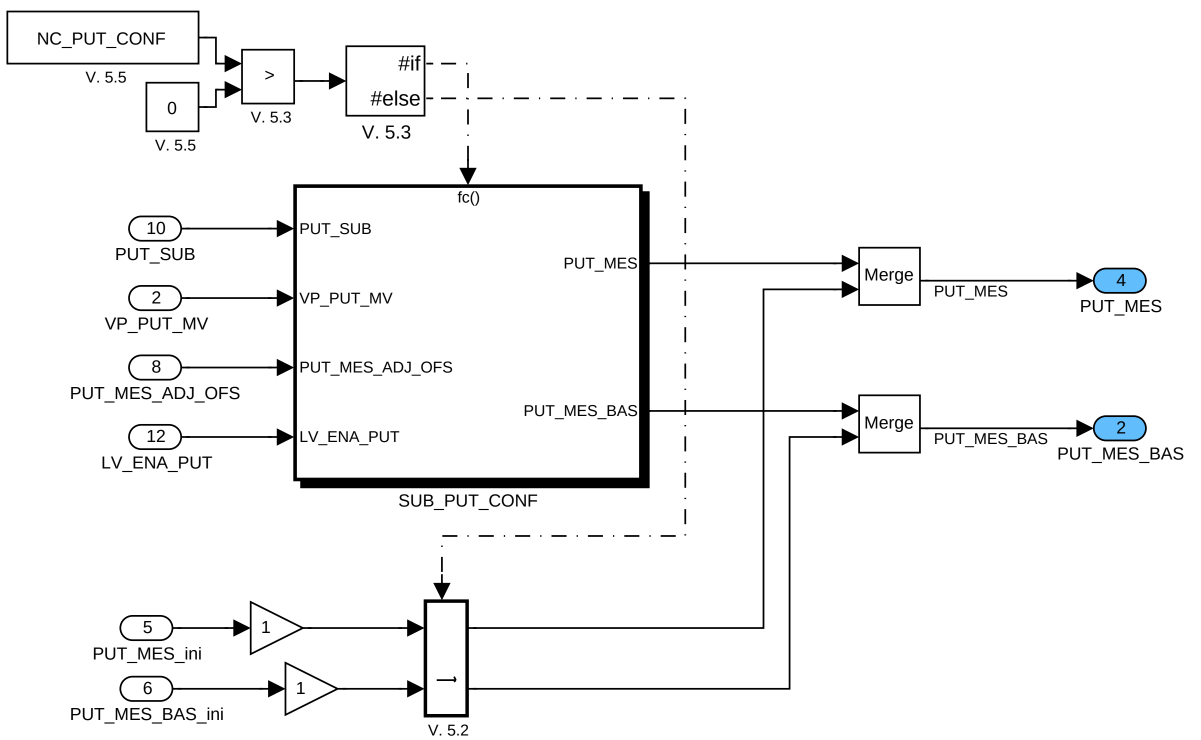


Figure 39:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_CONV

4.7.2.2.3.1 Conversion of the MAP sensor voltage into MAP_MES:

The manifold air pressure sensor voltage VP_MAP_MV is converted into a pressure signal MAP_MES.

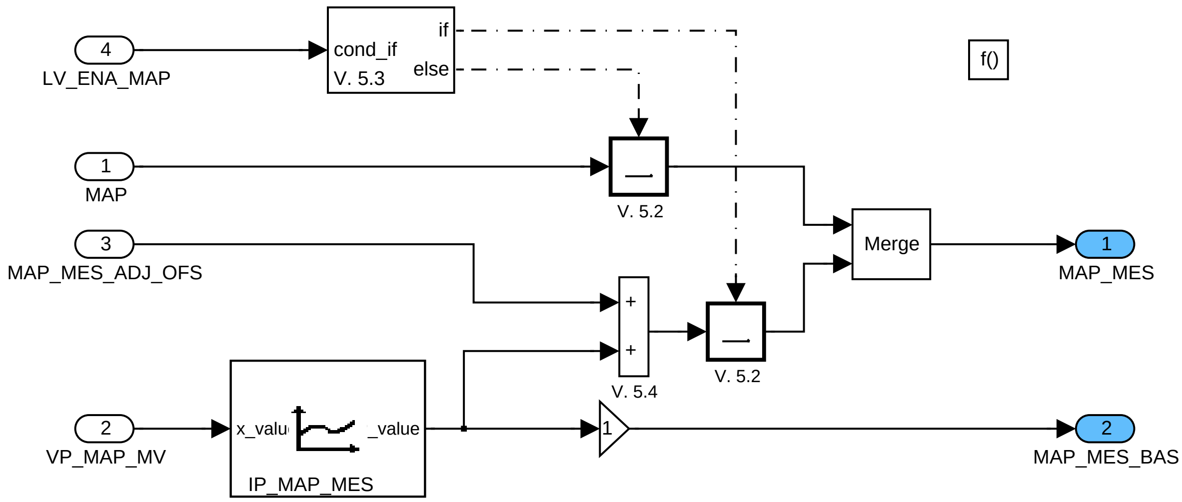


Figure 40:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_CONV/SUB_MAP_CONF

4.7.2.2.3.2 Conversion of the PUT sensor voltage into PUT_MES:

The pressure upstream throttle sensor voltage VP_PUT_MV is converted into a pressure signal PUT_MES.

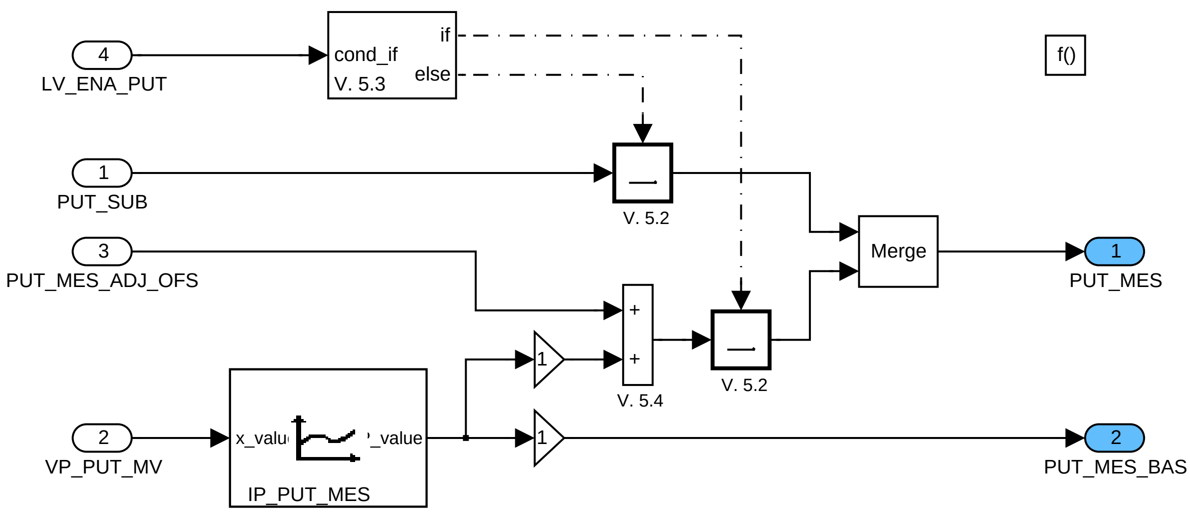


Figure 41:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_CONV/SUB_PUT_CONF

4.7.2.2.4 Calculation of filtered pressure signals:

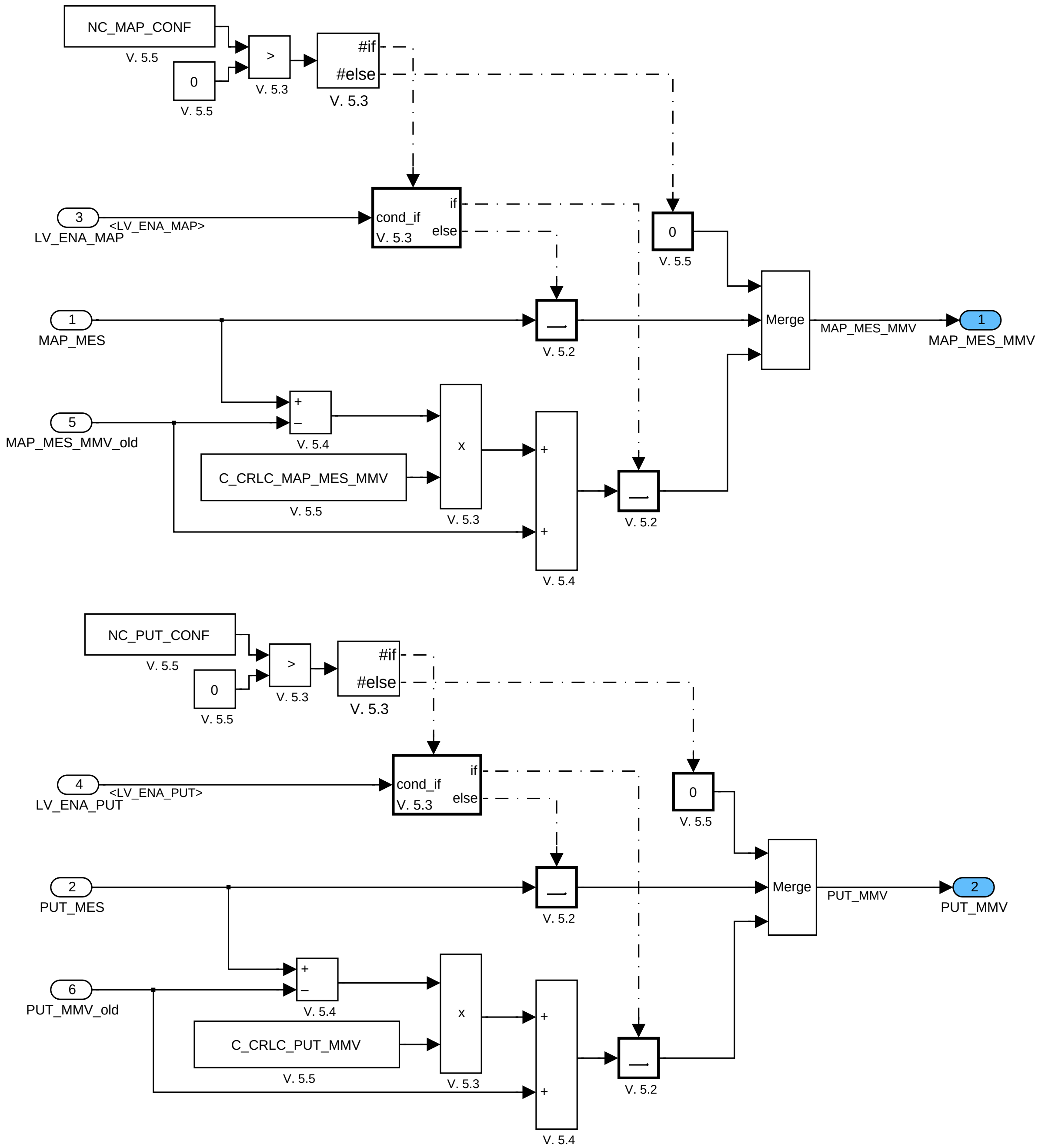


Figure 42:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/SUB_FIL

4.7.2.2.5 Calculation of VP_MAP_MAX/MIN:

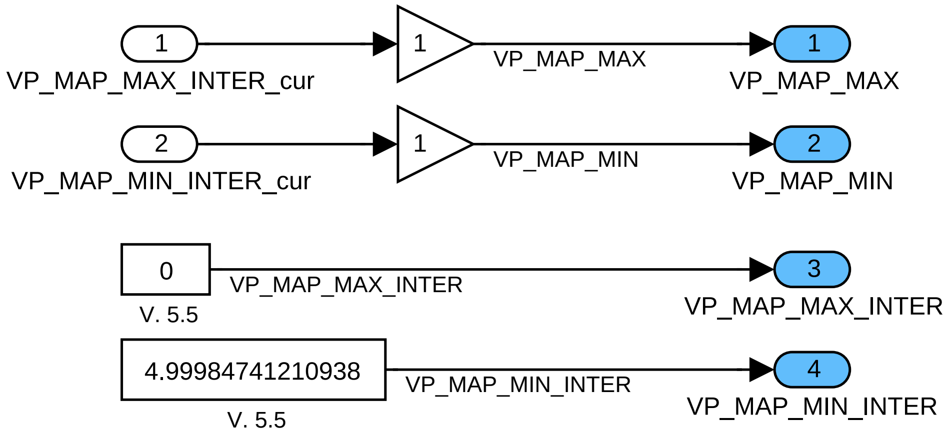


Figure 43:
Path: INSY_SIGCVPRS0/SIGCVPRS0__SEG10/op__SEG10/CLC_VP_MAP_MAX_MIN

4.7.3 operate_100ms:

Function description

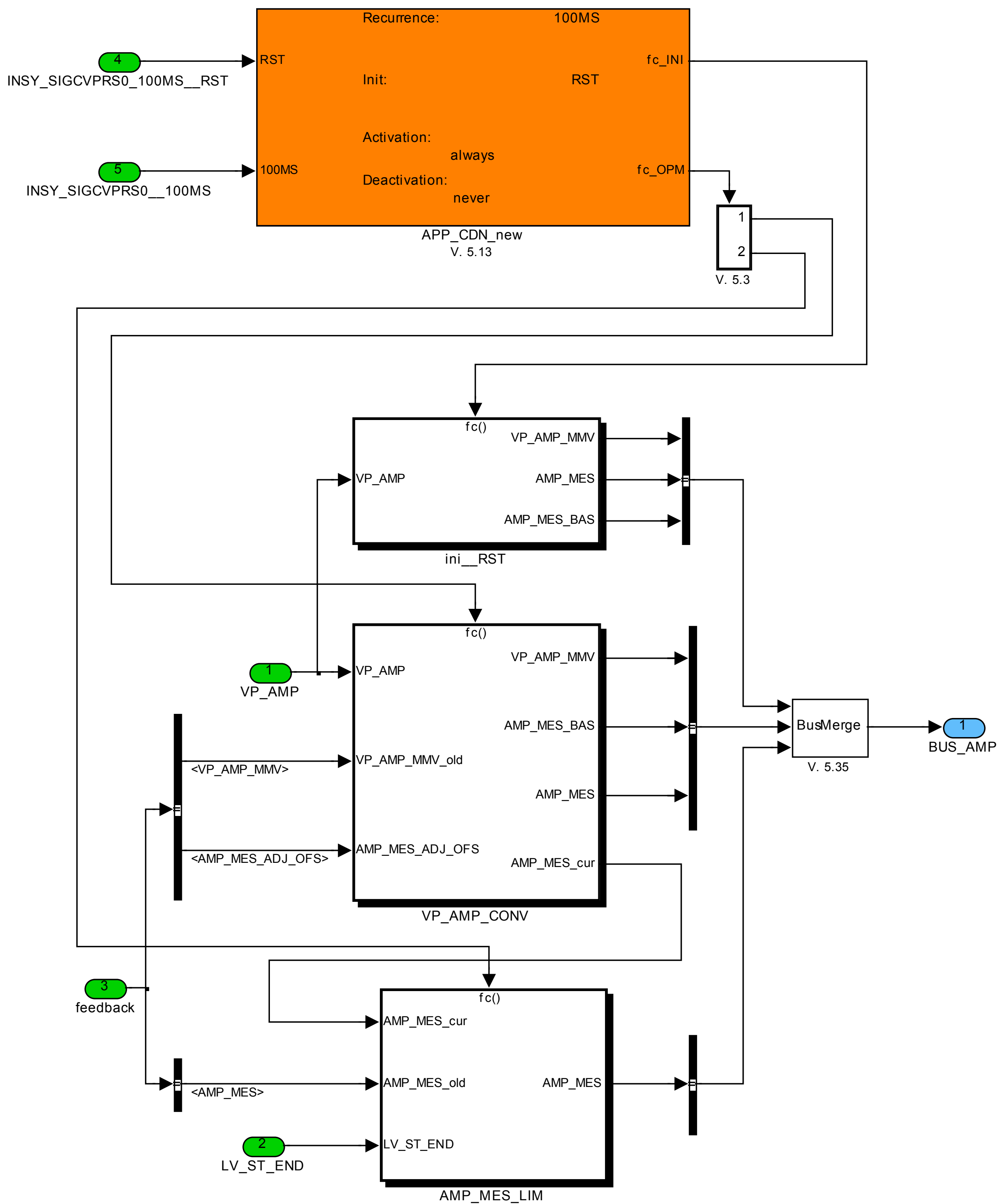


Figure 44:
Path: INSY_SIGCVPRS0/SIGCVPRS0__100MS

4.7.3.1 Initialization at reset:

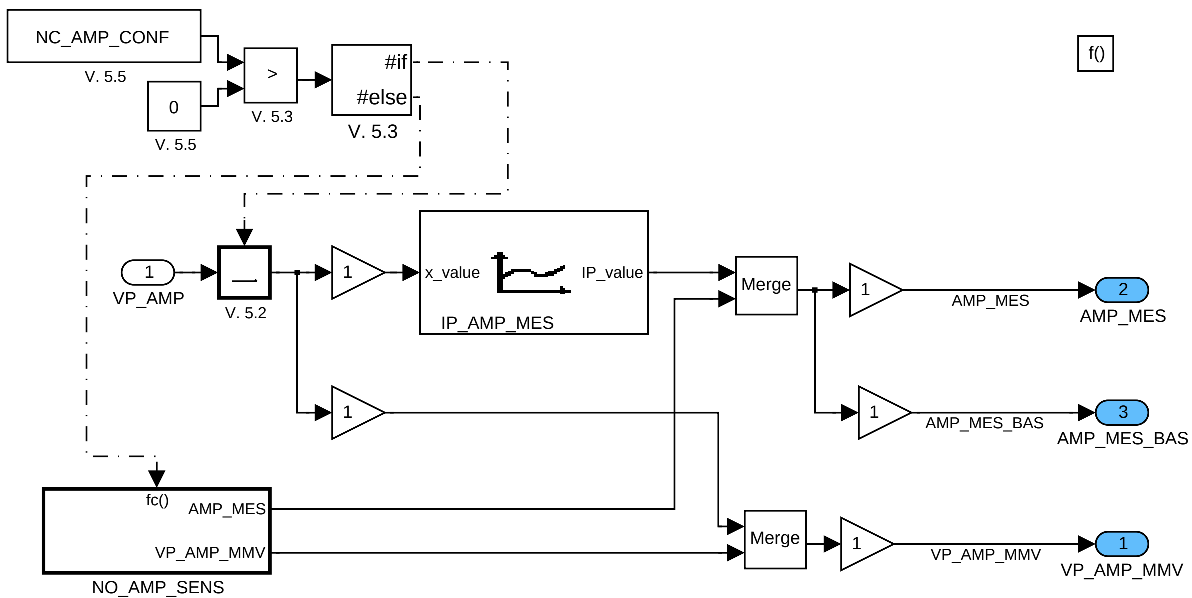


Figure 45:
Path: INSY_SIGCVPRS0/SIGCVPRS0__100MS/ini__RST

4.7.3.1.1 Initialization at reset if no AMP sensor is available:

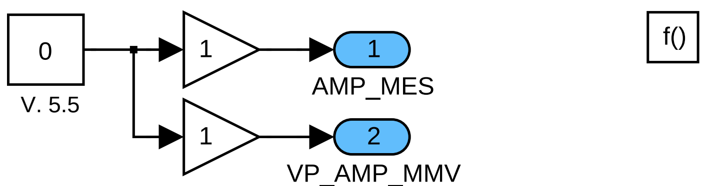


Figure 46:
Path: INSY_SIGCVPRS0/SIGCVPRS0__100MS/ini__RST/NO_AMP_SENS

4.7.3.2 Conversion of the AMP sensor voltage into AMP_MES:

The ambient pressure sensor voltage VP_AMP is converted into a pressure signal AMP_MES.

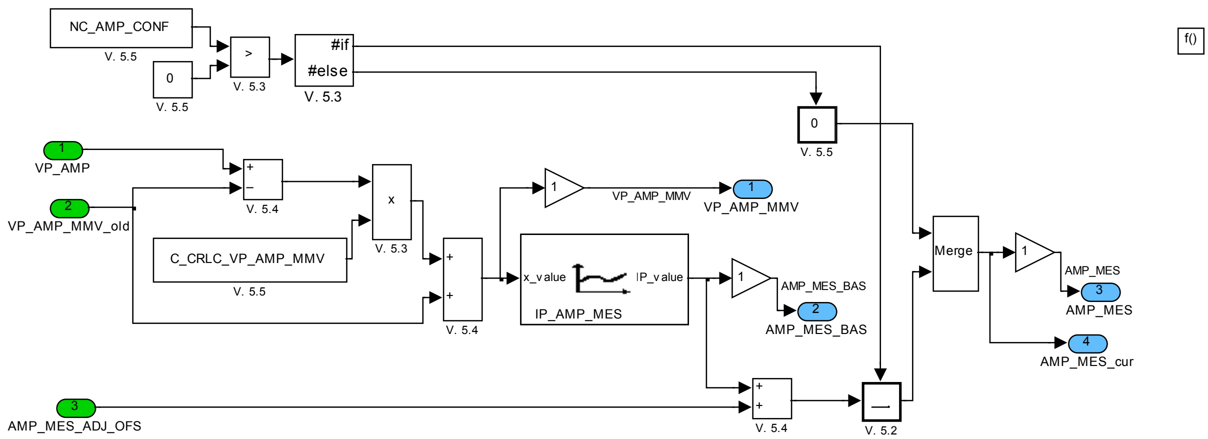


Figure 47:
Path: INSY_SIGCVPRS0/SIGCVPRS0__100MS/VP_AMP_CONV

4.7.3.3 Change limitation of AMP_MES:

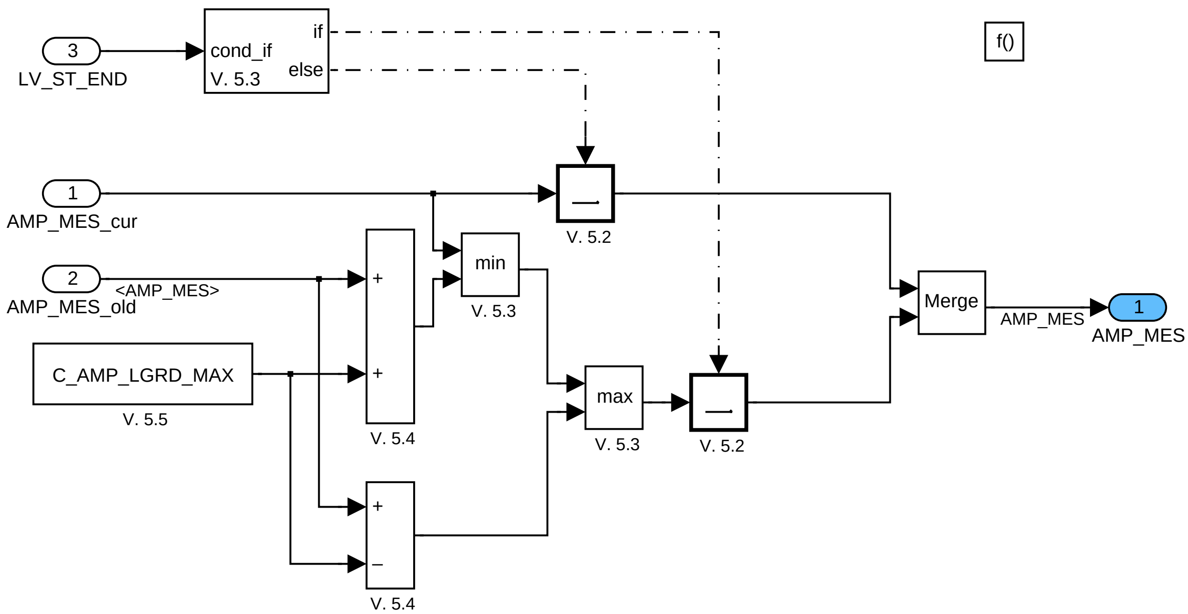


Figure 48:
Path: INSY_SIGCVPRS0/SIGCVPRS0__100MS/AMP_MES_LIM

4.7.4 operate_PWLOFF:

Function description

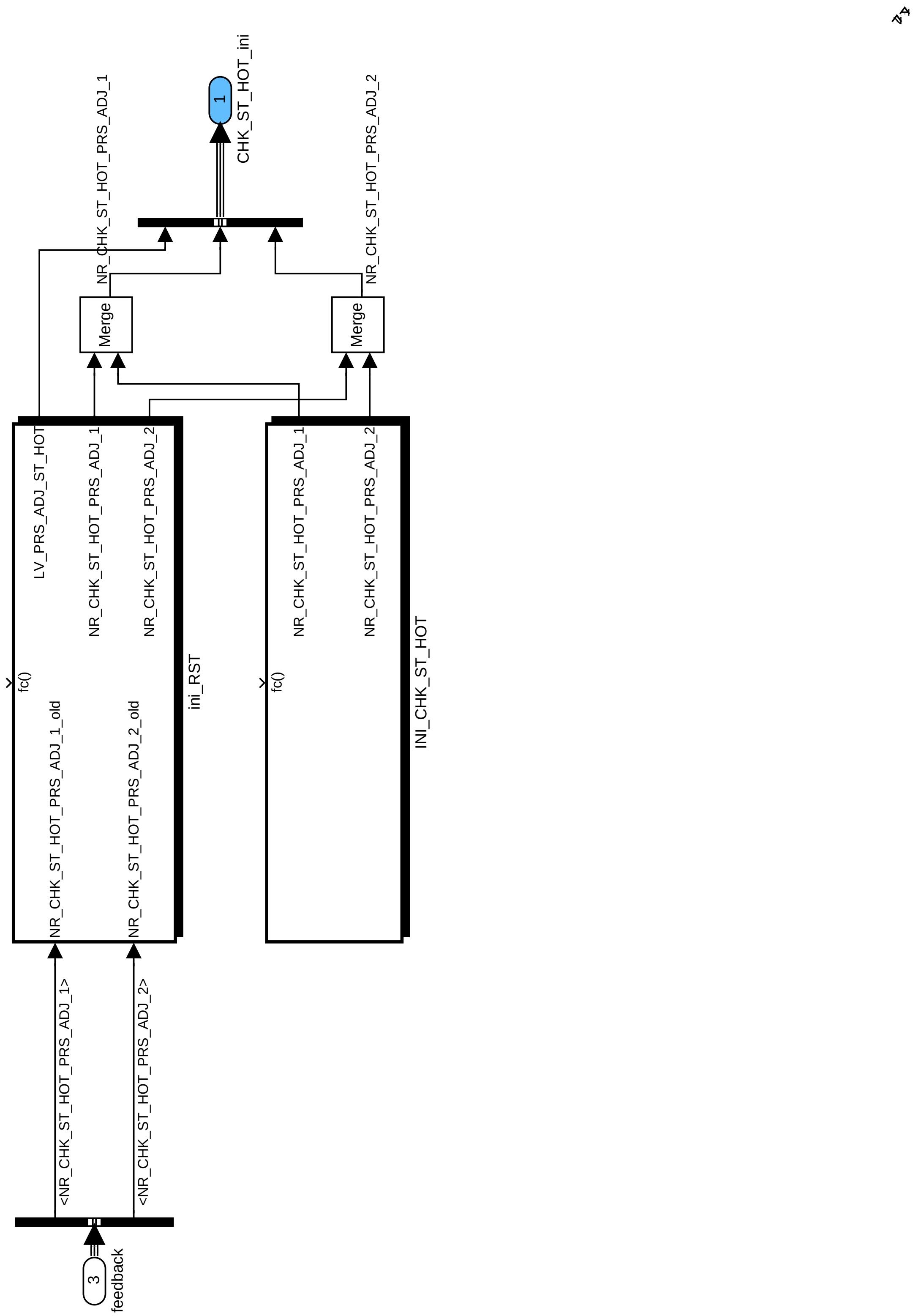


Figure 49:
Path: INSY_SIGCVPRS0/SIGCVPRS0__PWLOFF

4.7.4.1 Initialization at reset:

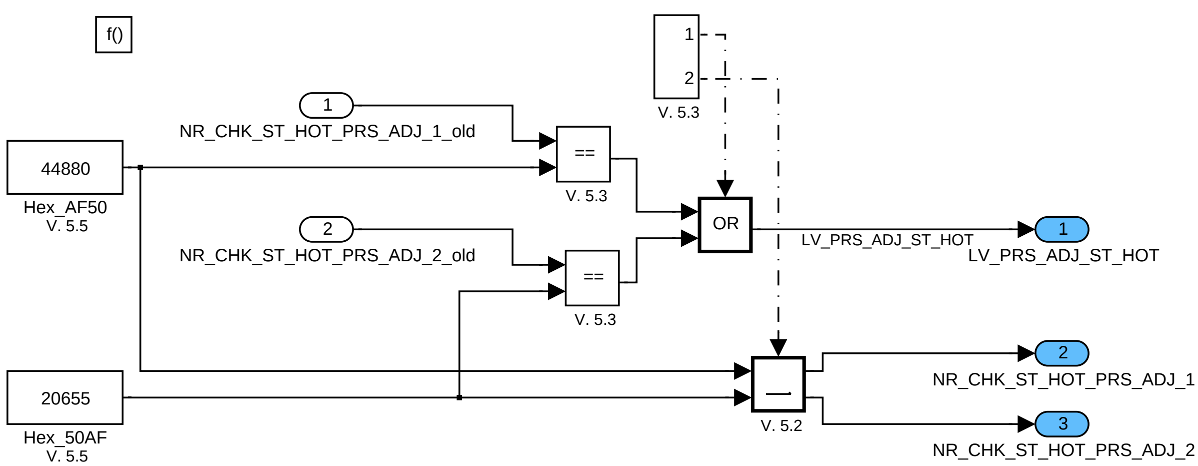


Figure 50:
Path: INSY_SIGCVPRS0/SIGCVPRS0__PWLOFF/ini_RST

4.7.4.2 Initialization at PWLOFF of variables for checking whether there was a hot start:

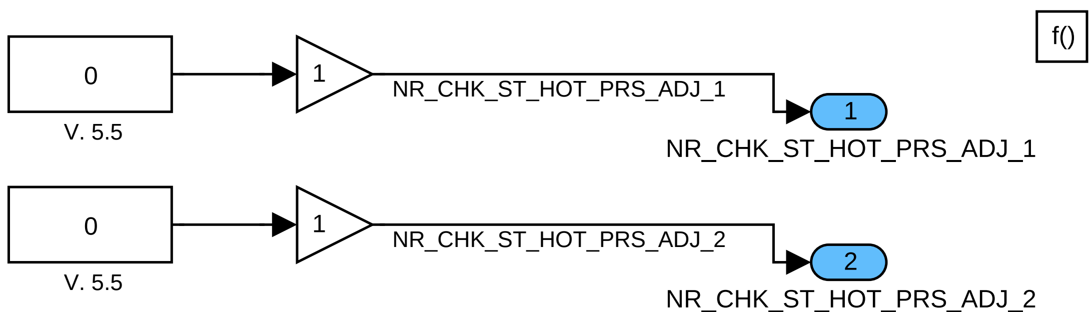


Figure 51:
Path: INSY_SIGCVPRS0/SIGCVPRS0__PWLOFF/INI_CHK_ST_HOT

4.8 Pressure Variables (only in case of MAP Sensor)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAP_MDL_MV	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Mean value of the model manifold pressure					
MAP_MDL_DIF	O/V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Deviation between measured and calculated manifold pressure					
LV_MAP_PLS	O/V	0...1H	0...1	1	-
Boolean for MAP pulsation present					
MAP_MDL	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Model manifold pressure					
MAP_CLC_DIF	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Manifold air pressure difference per segment					
CTR_MAP_PLS	V	0...FFH	0...255	1	-
Counter for MAP pulsation					
MAP_PLS	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manifold pressure pulsation					

Input data:

AMP	LV_ES	LV_ENA_MAP	MAP
LV_MAP_CTL	VP_MAP_MIN	VP_MAP_MAX	SEG_INC
N_32	TPS	MAP_MES	IP_MAP_MES

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_MAP_MDL	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for model manifold pressure					
C_CTR_MAP_PLS	1	0...FFH	0...255	1	-
Counter for detecting manifold air pressure pulsation and back					
C_MAP_DIF_MAX	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
MAP_DIF threshold for pulsation correction					
C_MAP_PLS_HYS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
MAP_PLS hysteresis for IP_MAP_PLS_MAX threshold					
C_N_MAX_MAP_PLS	1	0...FFH	0...8.16E+3	32	rpm
Max. engine speed threshold for MAP pulsation					
C_TPS_MIN_MAP_PLS	1	0...FFH	0...119.040469	0.46682537	°TPS
Minimum TPS threshold for MAP pulsation correction					
IP_MAP_PLS_MAX	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_N_32_1_INSY	6	0...FFH	0...8.16E+3	32	rpm
MAP_PLS threshold for pulsation correction					

4.8.1 General information:

The model mean value MAP_MDL_MV has to be calculated. The sensor behaviour and the data-acquisition of the ECU is integrated in the model to minimize the deviations under transient engine operating. In the case of a bad sensor signal (LV_MAP_CTL=0) this value will be used as substitution for the measured manifold pressure sensor value.

Function Description

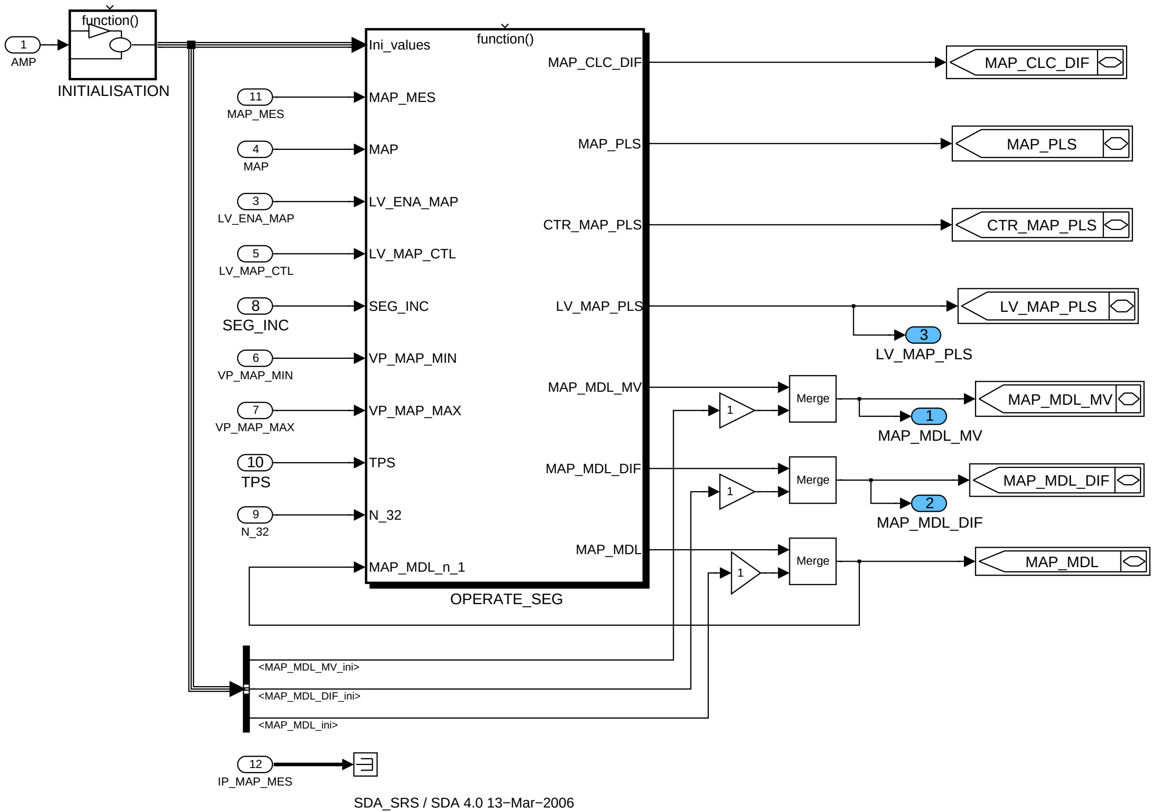


Figure 52 INSY_MDLADMAP0

4.8.1.1 Initialization:

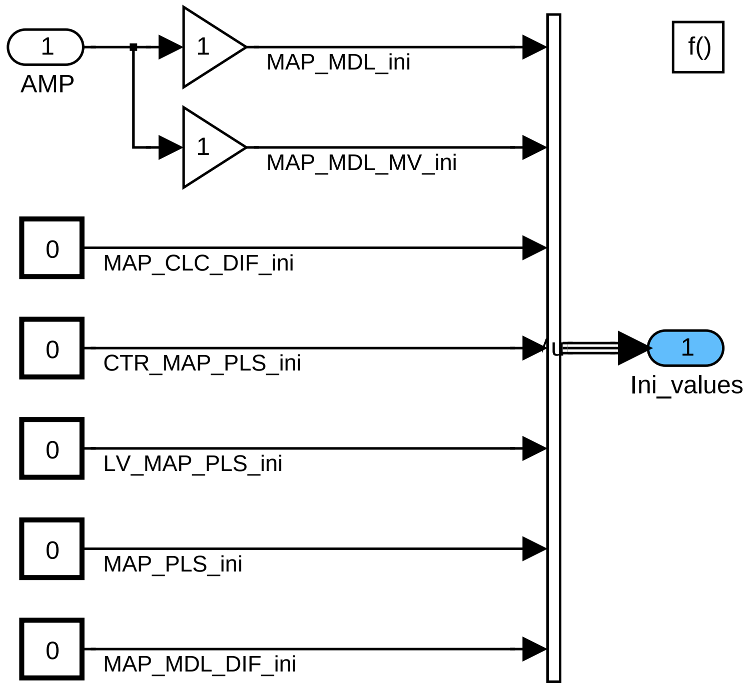


Figure 53 INSY_MDLADMAP0/ INITIALISATION

4.8.1.2 Calculation of model deviation MAP_MDL_DIF

The deviation between the model manifold pressure and the measured manifold pressure will be calculated only if the sensor is diagnosed as properly working.

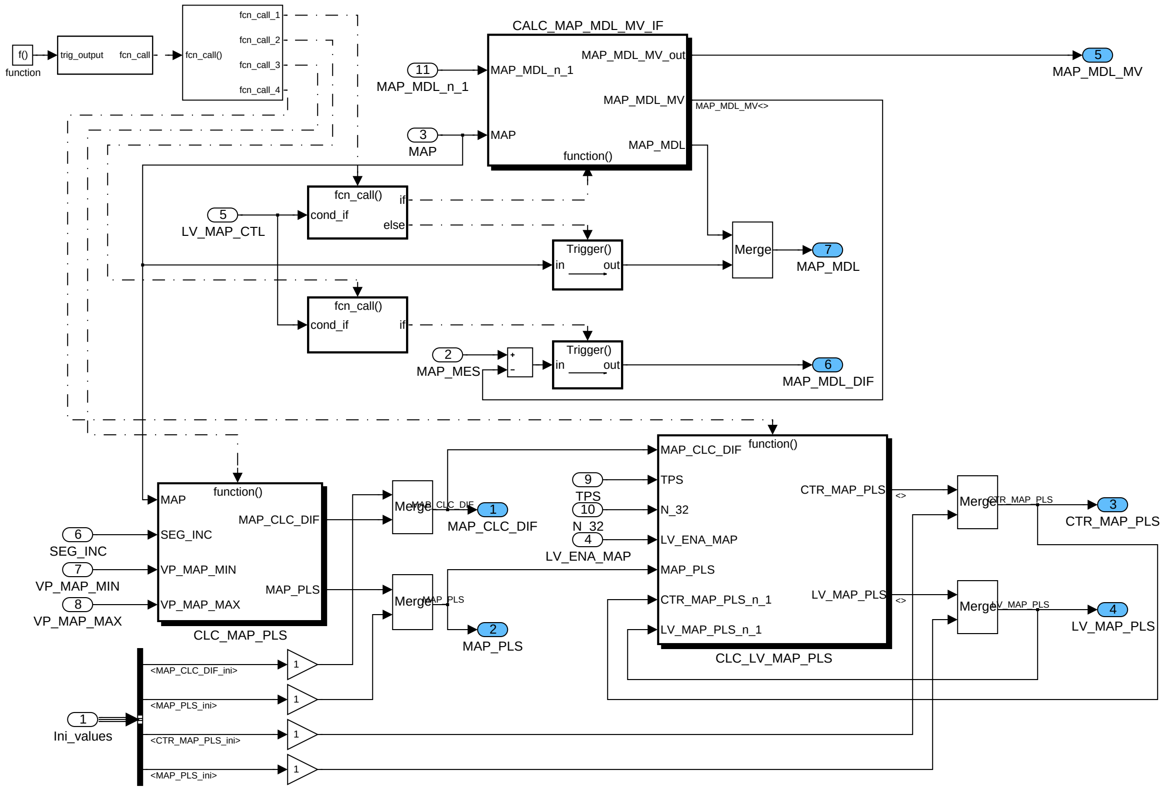


Figure 54 INSY_MDLADMAP0/ OPERATE_SEG

Calculation of model manifold pressure mean value MAP_MDL_MV

The scanning of the manifold pressure by the ECU is described by the mean value between the two previous segments. This calculation only takes place if the sensor is properly working.

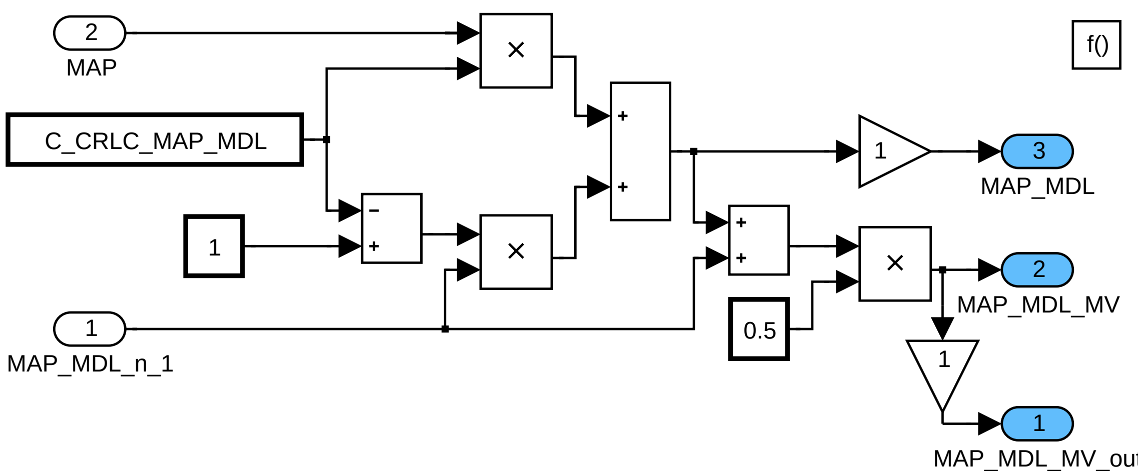


Figure 55 INSY_MDLADMAP0/ OPERATE_SEG/ CALC_MAP_MDL_MV_IF

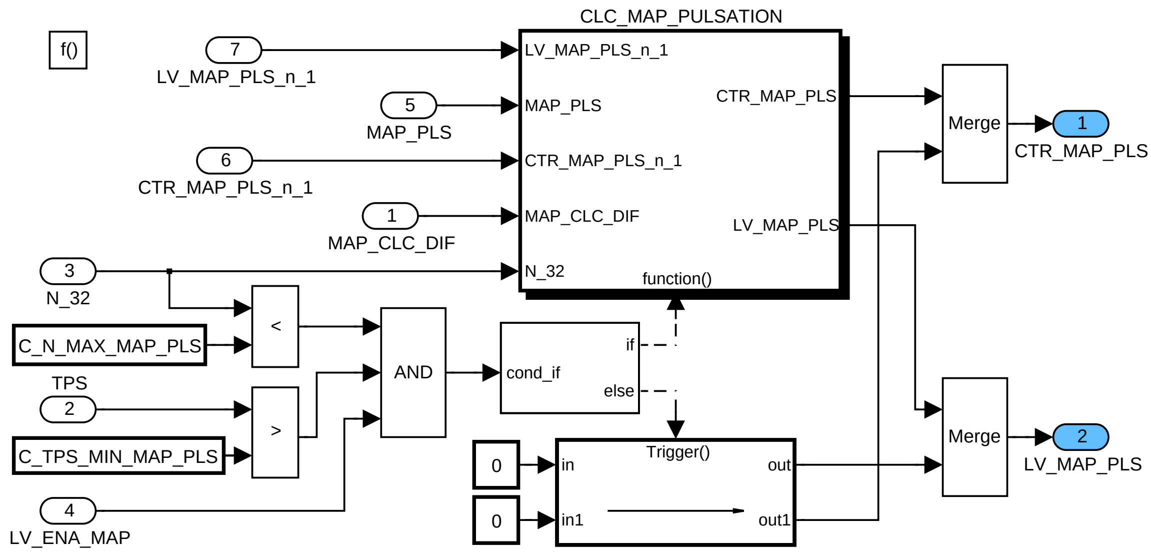


Figure 56 INSY_MDLADMAP0/ OPERATE_SEG/ CLC_LV_MAP_PLS

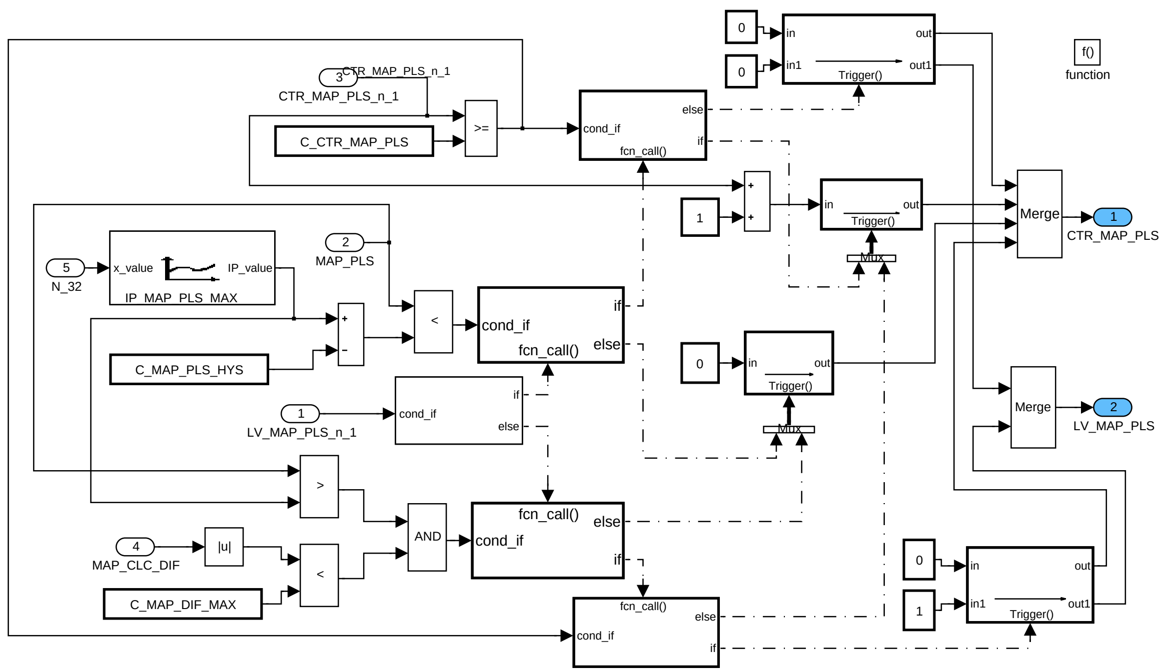


Figure 57 INSY_MDLADMAP0/ OPERATE_SEG/ CLC_LV_MAP_PLS/ CLC_MAP_PULSATION

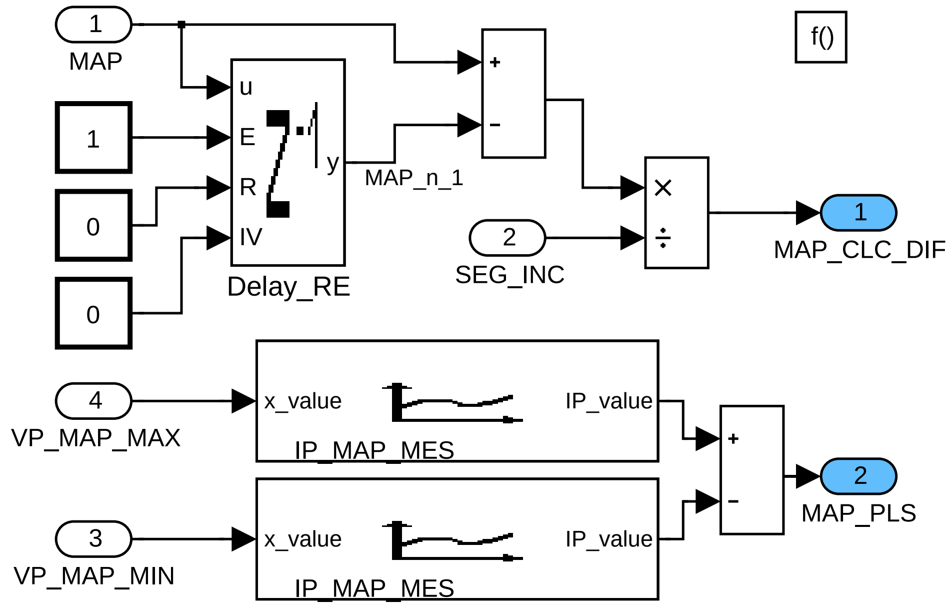


Figure 58 INSY_MDLADMAP0/ OPERATE_SEG/ CLC_MAP_PLS

4.9 Sensor specific Air Mass Flow Variables (only in case of MAF Sensor)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_KGH_MES	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Corrected mass air flow, measured per segment					
MAF_MDL_DIF	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
Deviation of model air mass flow					
MAF_MDL_MV	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Model air mass flow mean value [kg/h]					
LV_MAF_PULS	O/V	0...1H	0...1	1	-
Boolean for pulsation present					
MAF_MES	V	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Air-mass flow per segment					
MAF_PULS	V	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Air- mass flow pulsation amplitude					
T_SEG_T_1_HFM_RATIO	V	0...FFFFH	0...7.99987793	1.2207E-4	-
time constant for modeling HFM					
MAF_MDL	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
modeled mass air flow at sensor					
CTR_MAF_PULS	V	0...FFH	0...255	1	-
Counter for MAF_PULS					

Input data:

NC_MAF_FAC_CYL	MAF_KGH_MES_BAS	MAF_KGH_MES_FAC	N
LV_MAF_CTL	T_SEG_AV	MAF_THR	MAF_MAX
MAF_MIN	TPS	N_32	LV_ERR_MAF
MAF_DIF			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_MAF_PULS	1	1...10H	1...16	1	-
Counter for detection pulsation and back					
C_MAF_DIF_MAX	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
MAF_DIF threshold for pulsation correction					
C_MAF_PULS_HYS	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
MAF_PULS hysteresis for IP_MAF_PULS_MAX__N threshold					
C_N_MAX_PULS	1	0...FFH	0...8.16E+3	32	rpm
Max. engine speed threshold for pulsation					
C_TPS_MIN_PULS	1	0...FFH	0...119.040469	0.46682537	°TPS
Minimum TPS threshold for pulsation correction					
IP_CRLC_HFM	6	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_T_SEG_T_1_HFM_RATIO_IP_CRLC	6	0...FFFFH	0...7.99987793	1.2207E-4	-
Correlation constant for the first order time delay of the calculated air mass flow at the throttle					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_PULS_MAX	6	0...FFFFH	0...1.389E+3	0.0211947 8	mg/stk
LDPM_N_32_1_INSY	6	0...FFH	0...8.16E+3	32	rpm
MAF_PULS threshold for pulsation correction					
IP_T_1_HFM	4	0...FFFFH	0...65.535	0.001	ms
LDPM_MAF_KGH_1_INSY	4	0...FFFFH	0...2.04797E+3	0.03125	kg/h
First time constant hot film air mass meter					

4.9.1 INSY_MDLADMAF0

MAF_KGH_MES is equivalent to MAF_KGH_MES_BAS corrected by the factor MAF_KGH_MES_FAC.

The air-mass MAF_MES in [mg/stroke] is calculated from the measured air-mass flow MAF_KGH_MES and engine speed N. NC_MAF_FAC_CYL is inversely proportional to the number of cylinders, so the output MAF_MES depends on the number of cylinders.

Maximum and minimum air mass values are determined via a segment, from which results the differential air mass value. The values MAF_MAX and MAF_MIN are the maximum and minimum air mass flow values in [kg/h], they are converted in [mg/stroke] via the engine speed and the number of cylinders.

The model mean value MAF_MDL_MV has to be calculated. The sensor behavior and the data-acquisition of the ECU are integrated in the model to minimize the deviations under transient engine operating. In the case of a bad sensor signal (LV_MAF_CTL = 0) this value will be used as substitution for the measured air-mass flow value.

Function Description

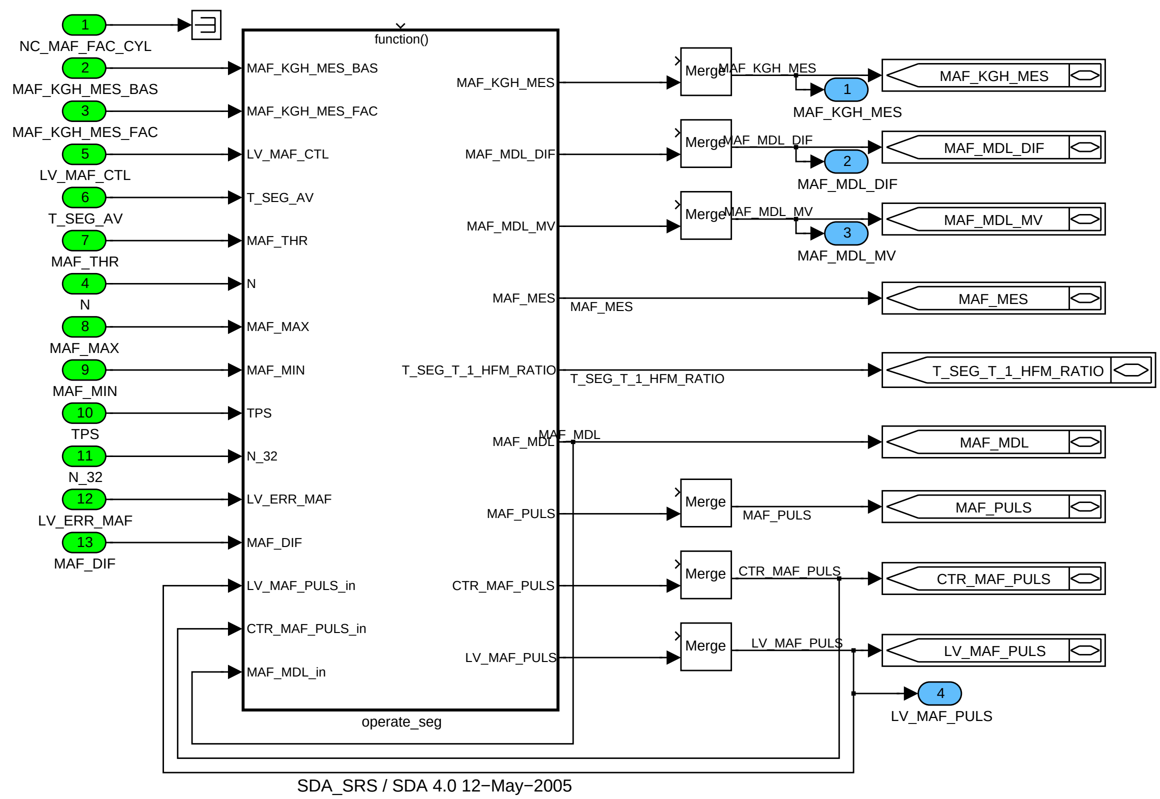


Figure 59 INSY_MDLADMAF0

4.9.1.1 INSY_MDLADMAF0/OPERATE_SEG

MAF_KGH_MES and MAF_MES are calculated seperately before the intake manifold model. The remaining calculations are done after the intake manifold model.

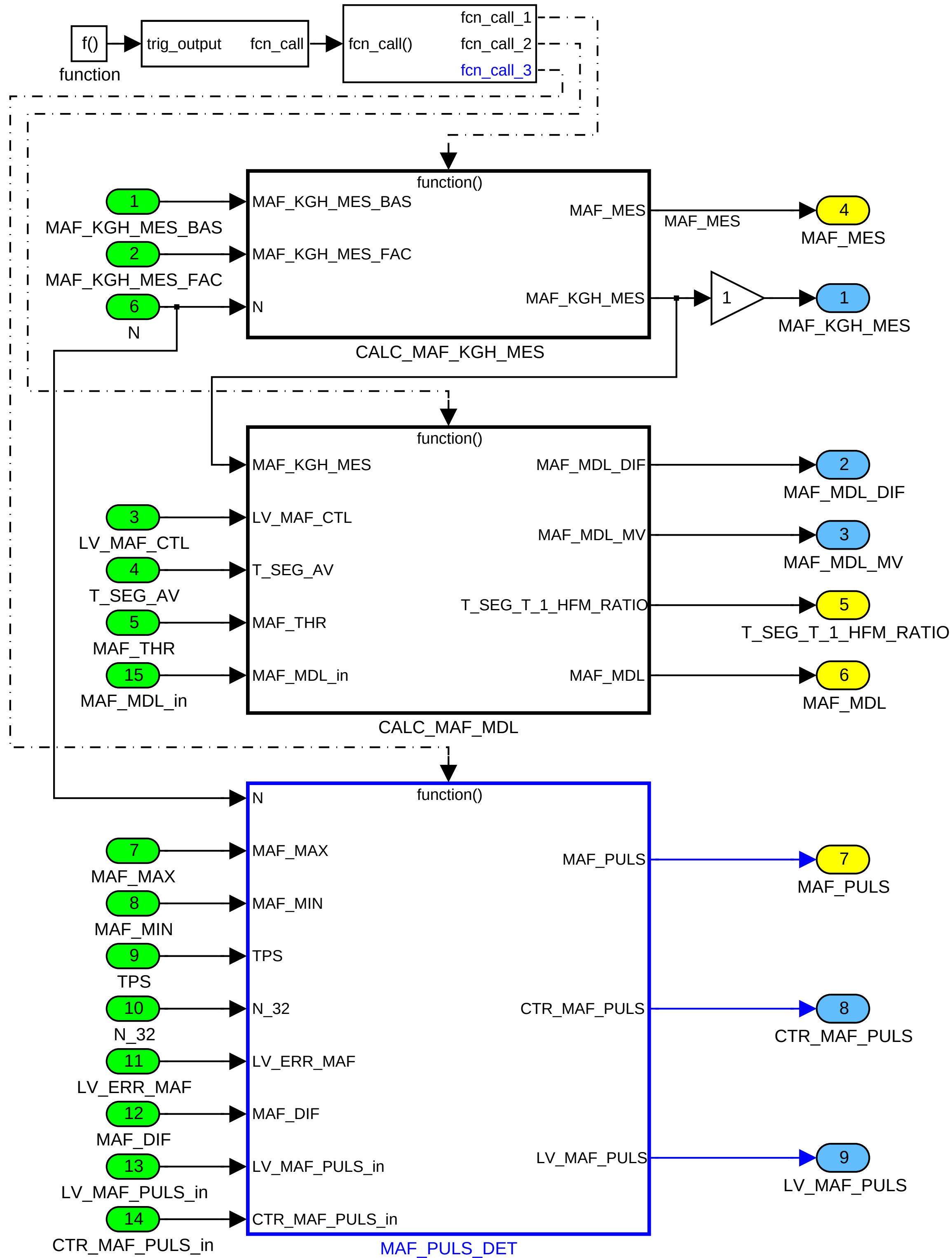


Figure 60 INSY_MDLADMAF0/ operate_seg

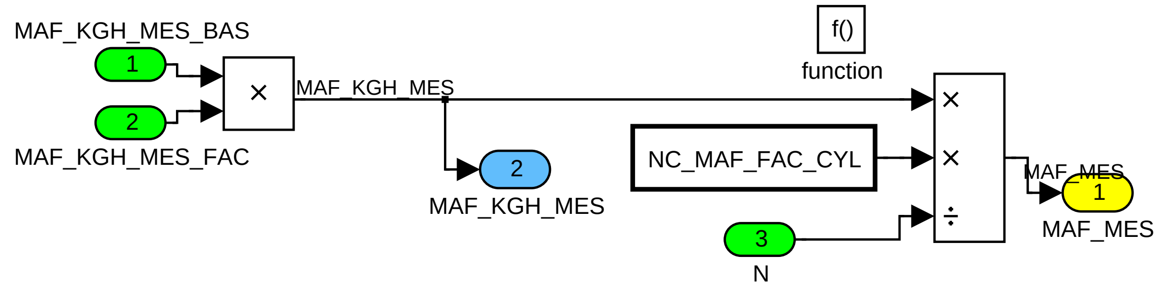


Figure 61 INSY_MDLADMAF0/ operate_seg/ CALC_MAF_KGH_MES

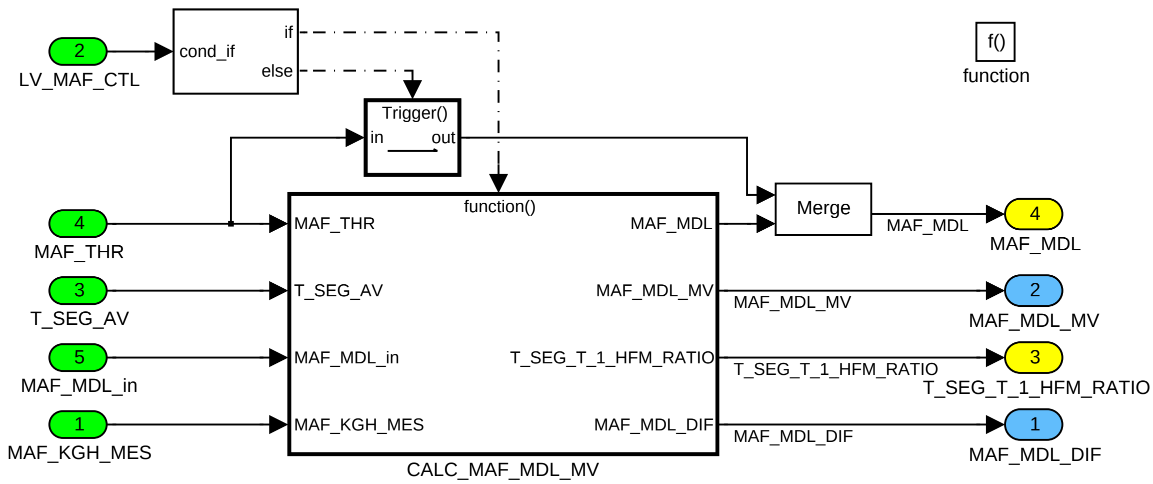


Figure 62 INSY_MDLADMAF0/ operate_seg/ CALC_MAF_MDL

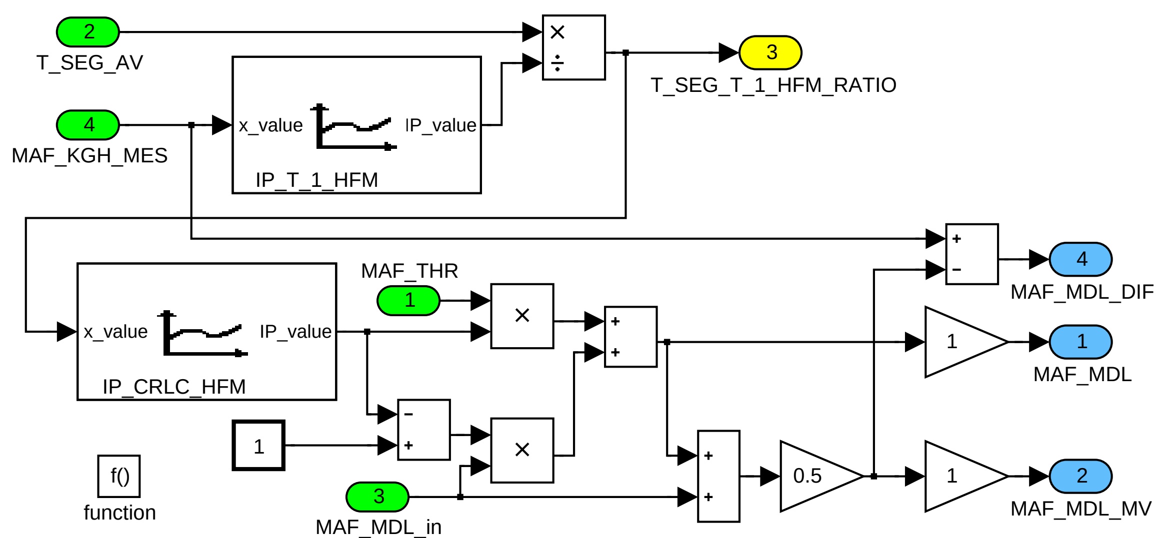


Figure 63 INSY_MDLADMAF0/ operate_seg/ CALC_MAF_MDL/ CALC_MAF_MDL_MV

INSY_MDLADMAF0/OPERATE_SEG/MAF_PULS_DET

If the difference of maximum and minimum air mass flow during a segment MAF_PULS is greater than a threshold MAF_PULS_MAX LV_CTR_SET_MAF_PLS is set. If MAF_PULS is smaller than MAF_PULS_MAX minus a hysteresis LV_CTR_RST_MAF_PLS is set. When any of these bit toggles then the counter CTR_MAF_PULS is initialized with C_CTR_MAF_PULS; otherwise CTR_MAF_PULS is decreased. If LV_CTR_SET_MAF_PLS has been set for C_CTR_MAF_PULS consecutive cycles, then MAF pulsation is detected (LV_MAF_PULS = 1). If LV_CTR_RST_MAF_PLS has been set for C_CTR_MAF_PULS consecutive cycles, then no MAF pulsation is detected (LV_MAF_PULS = 0).

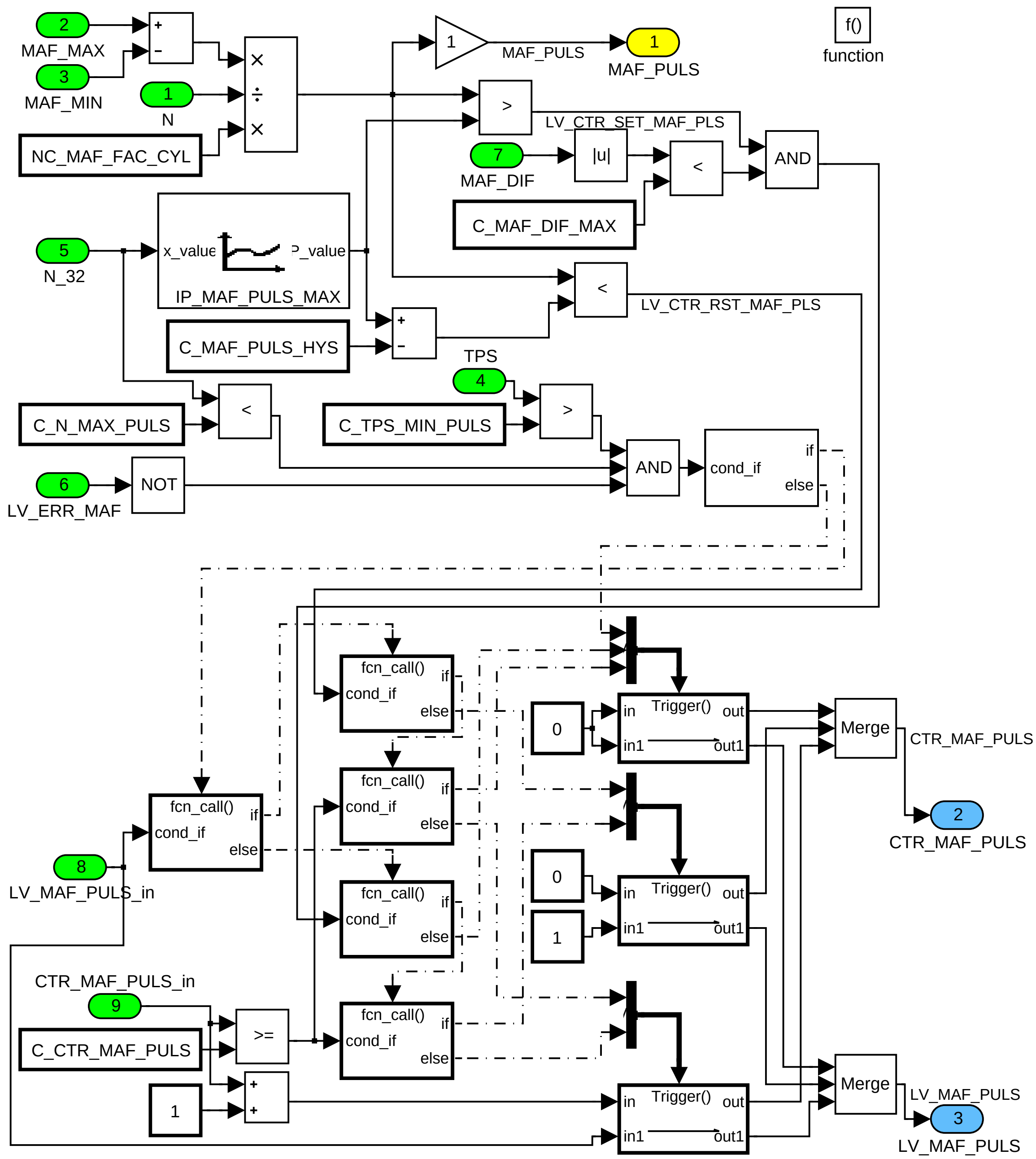


Figure 64 INSY_MDLADMAF0/ operate_seg/ MAF_PULS_DET

4.10 Intake manifold model

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
EGR_RATIO	O/V	0... FFFFH	0... 99.9984741211	1.5259e-3	[%]
Air mass flow at EGR valve / air mass flow to cylinder					
EGR_RATIO_CYL	O/V	0... FFFFH	0... 99.9984741211	1.5259e-3	[%]
Ratio of exhaust gas flow and complete gas flow into cylinder					
FAC_MAP_CLC_A	O	0... FFFFH	0... 12.0599926417	184.02e-6	[kg/(h*h Pa)]
Temporary variable for MAP calculation					
FAC_MAP_CLC_B	O	0... FFFFH	0... 32767.5	0.5	[kg/h]
Temporary variable for MAP calculation					
FAC_MAP_CLC_B0	O	0... FFFFH	0... 12.0599926417	184.02e-6	[kg/(h*h Pa)]
Temporary variable for MAP calculation					
FAC_MAP_CLC_C	O	0... FFFFH	0... 31.9995117188	488.28e-6	[-]
Temporary variable for MAP calculation					
LV_RA_VOL_IM_POS	V	0... 1H	0... 1	1	[-]
Flag to indicate switching IP_RA_VOL_IM_DYN_POS/NEG					
MAF	O/V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Air mass (predicted) for calculation of basic fuel injection					
MAF_CYL	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Air mass flow to cylinder [kg/h]					
MAF_CYL_STK	O/V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Air mass flow to cylinder [mg/stk]					
MAF_FAC_OFS	O/V	0... FFFFH	0... 8.88033	135.51e-6	[-]
Offset of the air flow equation for the throttle air mass flow					
MAF_FAC_SLOP	O/V	0... FFFFH	0... 8.88033	135.51e-6	[-]
Slope of the air flow equation for the throttle air mass flow					
MAF_FG_CYL	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Fresh air mass flow in the cylinder					
MAF_FG_CYL_STK	O/V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Fresh air mass flow in the cylinder per stroke					
MAF_HB	O/V	0... FFH	0... 1389	5.4470588	[mg/stk]
same as MAF but with a lower resolution (HB means High Byte)					
MAF_KGH_FG_PRED	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Predicted fresh gas out of the manifold					
MAF_MDL_CON_0	O/V	0... 7FFFH	0... 58299605.539173	1779.2171	[1/(s²m)]
Specific gas constant of air RA * Intake manifold air temperature TIA_IM / intake manifold volume VOL_IM					
MAF_MDL_CON_1	O/V	0... FFFFH	0... 0.0206019	314.36e-9	[s/m]
sqrt(2x/((x 1)*RA*TIA)) with x = kappa = 1.4					
MAF_MDL_CON_11	O/V	0... FFFFH	0... 120.7e-6	1.8418e-9	[m*s]
MAF_MDL_CON_1 * AR_RED					
MAF_THR	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Air mass flow at the throttle					
MAP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Manifold air pressure					
MAP_DRV1	O/V	8000... 7FFFH	-82.9175249866 ...82.9149945446	2.5304e-3	[hPa/ms]
First derivative of intake manifold pressure					
MAP_EGR	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]

Residual gas partial pressure in the manifold					
MAP_EGR_DRV1	V	8000... 7FFFH	-82.9175249866 ...82.9149945446	2.5304e-3	[hPa/ms]
First derivative of intake manifold residual gas pressure					
MAP_EGR_PRED	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Predicted residual gas partial pressure in the manifold					
MAP_EGR_RATIO	O/V	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Ratio from MAP_EGR and MAP					
MAP_ESTIM	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
estimated manifold pressure					
MAP_FG	V	0... FFFFH	0... 5434	0.0829175	[hPa]
Fresh gas partial pressure in the manifold					
MAP_FG_PRED	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Predicted fresh gas partial manifold air pressure					
MAP_PRED	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Predicted manifold air pressure					
PQ	O/V	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Pressure quotient at the throttle					
PQ_ESTIM	V	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Estimated pressure quotient at the throttle					
PQ_SP_MAP_DRV1_SWI	V	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Filtered pressure quotient for switching MAF_THR and MAP_DRV1 calculation					

Input Data:

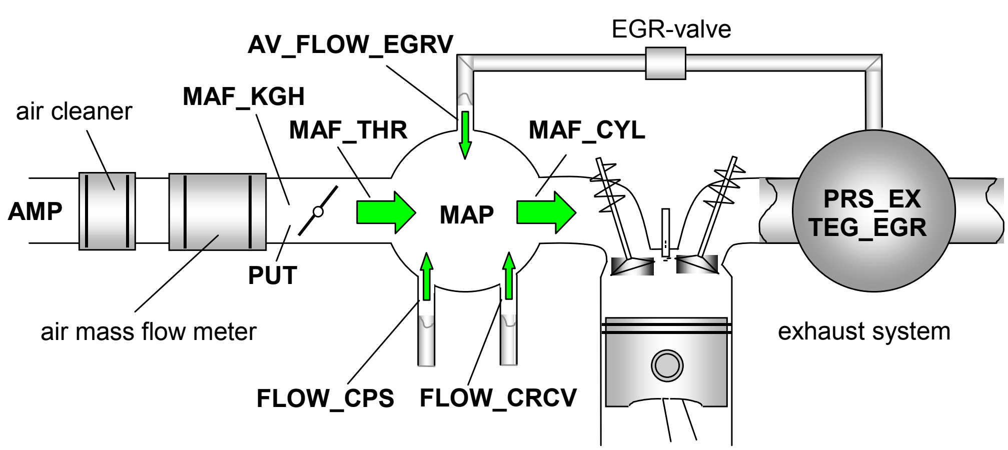
AMP	AR_RED	AR_RED_BAS	AV_FLOW_EGRV
EFF_VOL_OFS_SUM	EFF_VOL_SLOP	FLOW_CPS	FLOW_CRCV
LV_LIH_MAF	LV_ST_END	MAF_KGH_SUB	MAP_MES
N	NC_CHRG_CONF	NC_CYL_NR	NC_EGR_CONF
NC_MAF_FAC_CYL	NC_MAP_CONF	N_32	PUT_MDL
TIA_IM	TIA_THR	T_SEG_AV	T_SEG_AV_IMM
PUT	LV_MAP_PLS	LV_ERR_LOAD_PLAUS	LV_ENA_MAP
LV_MAP_CTL	CAM_MV_IN	PQ_SP	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_PQ_MAP_DRV1_SWI	1	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Correlation constant for PQ_SP_MAP_DRV1_SWI calculation					
C_DPL	1	0... 7FH	0... 12.7	0.1	[l]
Displacement of the engine					
C_FAC_MAP_PRED_TOL	1	0... FFH	0... 1.9921875	7.8125e-3	[-]
Factor to limit predicted manifold pressure					
C_MAF_CYL_COR	1	0... FFFFH	0... 61.035e-6	9.3133e-10	[(h/kg)²]
Correction factor for MAF_CYL calculation					
C_MAP_DRV1_HYS	1	8000... 0H	-82.9175249866 ...0	2.5304e-3	[hPa/ms]
Hysteresis for switching IP_RA_VOL_DYN_POS/NEG					
C_PQ_SP_MAP_DRV1_SWI	1	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Hysteresis considered for switching MAP_DRV1 calculation					

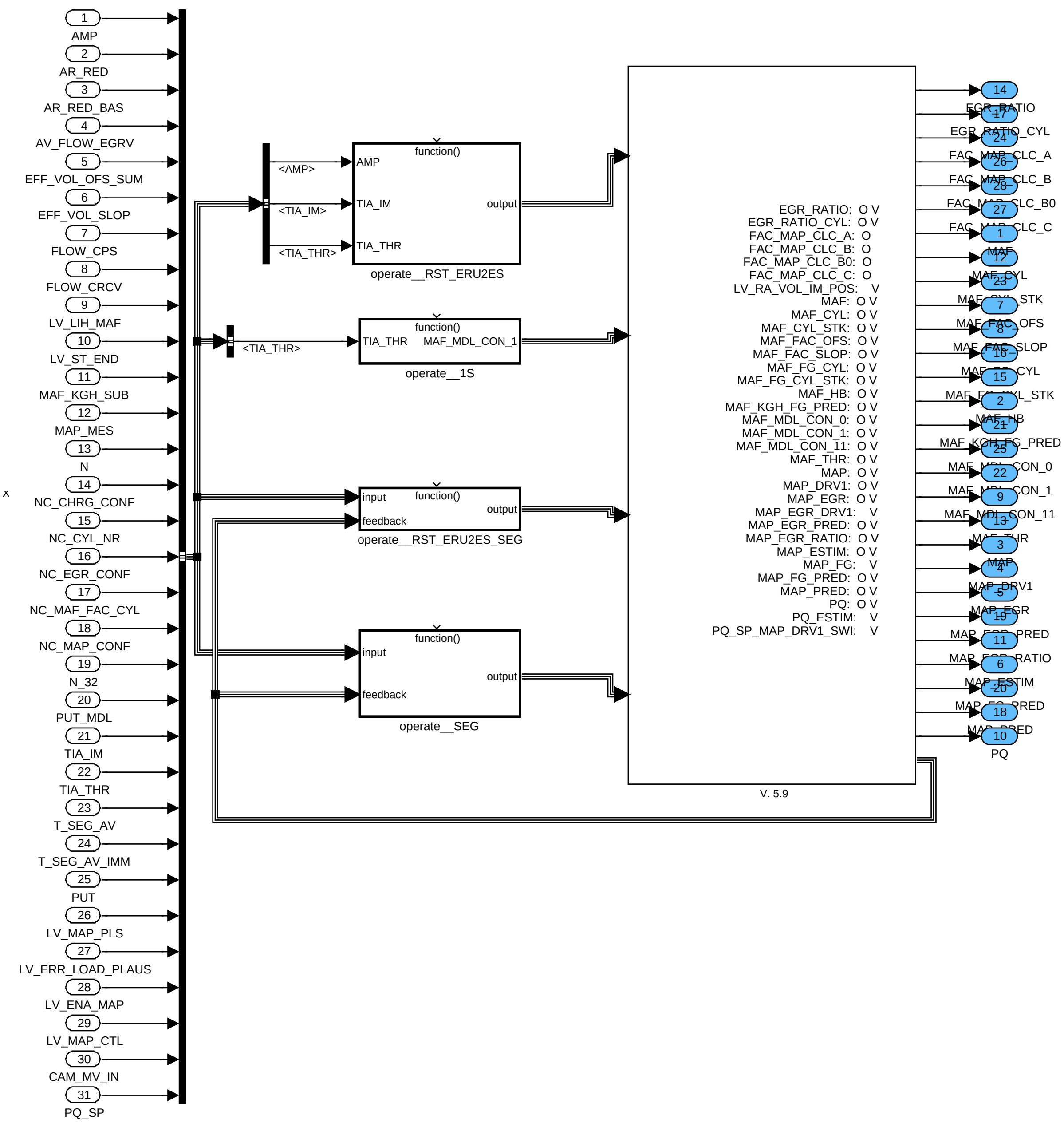
C_PQ_THD	1	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Calculation threshold for manifold pressure. Typical value: 0.9900 (FD70H)					
ID_MAF_FAC_OFS	16	0... FFFFH	0... 8.88033	135.51e-6	[-]
LDPM_PQ_ESTIM_1_INSY	16	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Offset of linearized air flow function					
ID_MAF_FAC_SLOP	16	0... FFFFH	0... 8.88033	135.51e-6	[-]
LDPM_PQ_ESTIM_1_INSY	16	0... FFFFH	0... 0.99998474121	15.259e-6	[-]
Slope of linearized air flow function					
IP_MAF_MDL_CON_1	8	0... FFFFH	0... 0.0206019	314.36e-9	[s/m]
LDP_TIA_THR_IP_MAF_MDL_CON_1	8	0... FEH	-48... 142.5	0.75	[°C]
Temperature influence on air flow function at the throttle					
IP_MAP_PRED_FAC	6*4	0... FFH	0... 7.96875	0.03125	[-]
LDPM_N_32_1_INSY	6	0... FFH	0... 8160	32	[rpm]
LDP_CAM_MV_IN_IP_MAP_PRED_FAC	4	0... FFH	60... 155.625	0.375	[°CRK]
Prediction factor for MAP					
IP_RA_VOL_IM_DYN_NEG	4*6	0... FFFFH	0... 303648.412228	4.6333778	[Pa/(kg*K)]
LDP_N_32_IP_RA_VOL_IM_DYN_NEG	6	0... FFH	0... 8160	32	[rpm]
LDP_MAP_IP_RA_VOL_IM_DYN_NEG	4	0... FFFFH	0... 5434	0.0829175	[hPa]
Specific gas constant of air divided by intake manifold volume(including engine speed and map decrease corrections)					
IP_RA_VOL_IM_DYN_POS	4*6	0... FFFFH	0... 303648.412228	4.6333778	[Pa/(kg*K)]
LDP_N_32_IP_RA_VOL_IM_DYN_POS	6	0... FFH	0... 8160	32	[rpm]
LDP_MAP_IP_RA_VOL_IM_DYN_POS	4	0... FFFFH	0... 5434	0.0829175	[hPa]
Specific gas constant of air divided by intake manifold volume(including engine speed and map increase corrections)					
LC_MAP_MES_ACT	1	0... 1H	0... 1	1	[-]
Switch whether MAP_MES(=1) or MAP(=0) is active for MAP_PRED calculation					
LC_PQ_ESTIM_SWI	1	0... 1H	0... 1	1	[-]
Switch whether PQ_SP is considered (= 1) or not (= 0) for PQ_ESTIM calculation					

General Information



The intake manifold model calculates the air mass flow at the throttle, the air mass flow into the cylinder and the pressure in the intake manifold. In case of exhaust gas recirculation (NC_EGR_CONF = 1) the exhaust gas flow will be considered. If the system is not equipped with an EGR then some of the calculations need not to be done (reduction of calculation time).

Function description



SDA_SRS / SDA V 5.0.2.2 21−Mar−2006

Figure 65:
Path: INSY_DTSYSIMM0

4.10.1 Initialization: operate__RST_ERU2ES

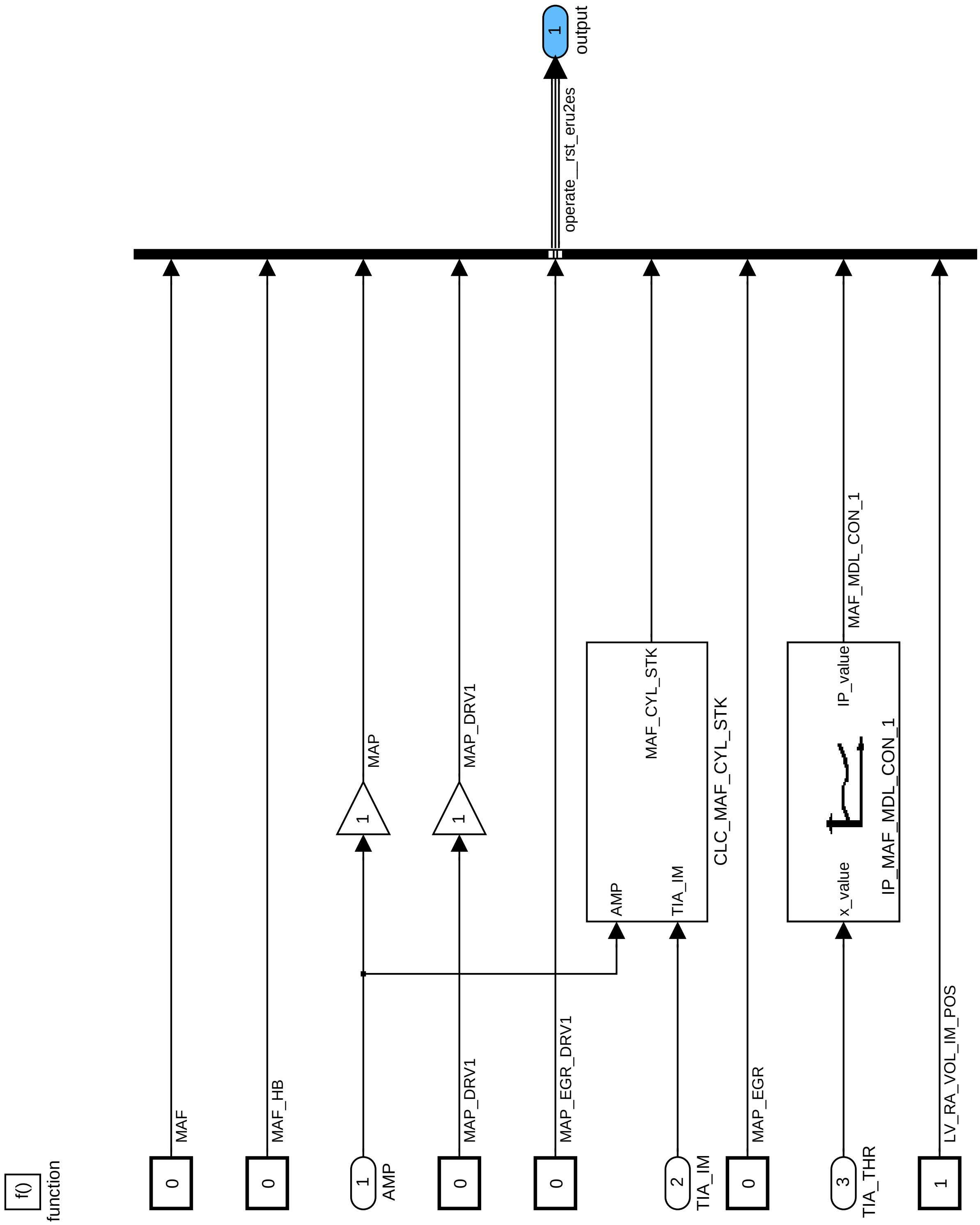


Figure 66:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES

4.10.1.1 Calculation of MAF_CYL_STK:

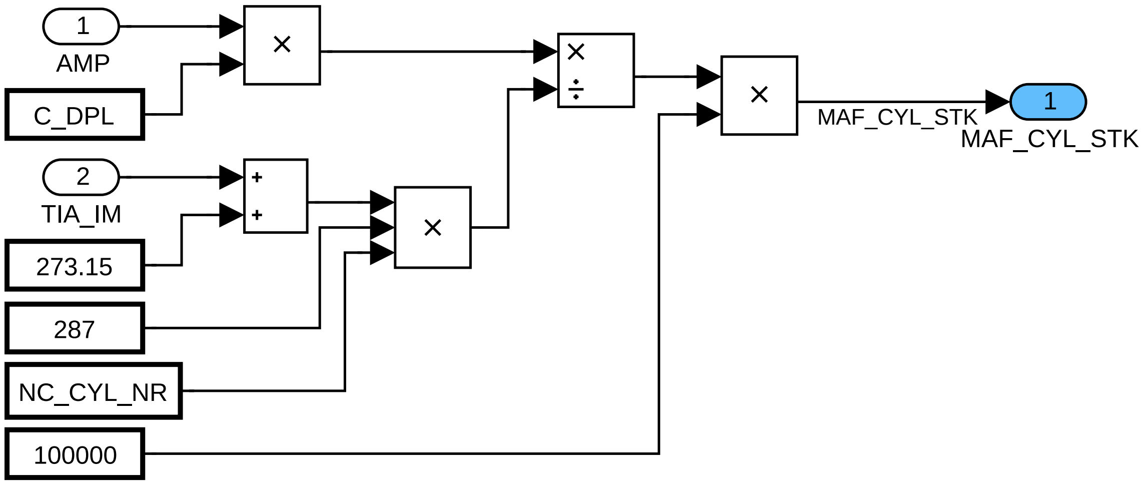


Figure 67:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES/CLC_MAF_CYL_STK

4.10.2 Calculation of MAF_MDL_CON_1: operate__1s

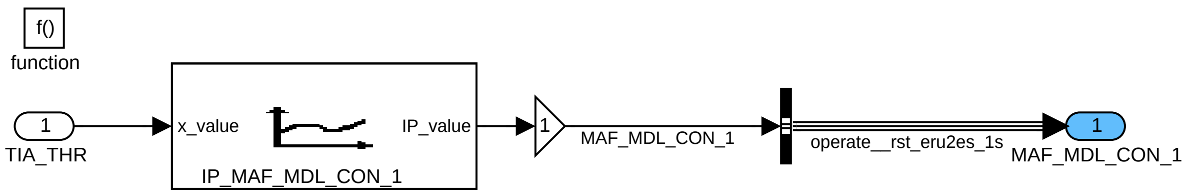


Figure 68:
Path: INSY_DTSYSIMM0/operate__1S

4.10.3 Calculation of Constants

During one loop some variables are used twice or more. These constants are prepaed at the beginning of the calculation process. The intake air temperature is scanned once a second and for the multiplication converted into the unit [°Kelvin].
The variable RA_VOL_IM represents the specific gas constant of air (RA = 287 [J/(kgK)]) devided by the intake manifold volume (m3]). Due to the influence of engine speed and load change it is defined as a map and not as a single constant.
Note: RA_VOL_IM and MAF_MDL_CON_0 have to be calculated at reset for the first time. Then the calculations run synchronously to multiples of a segment.

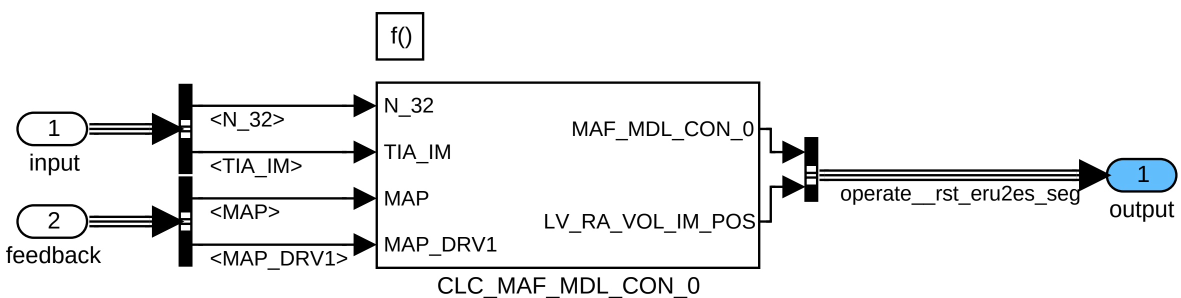


Figure 69:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES_SEG

4.10.3.1 Calculation of MAF_MDL_CON_0 and RA_VOL_IM:

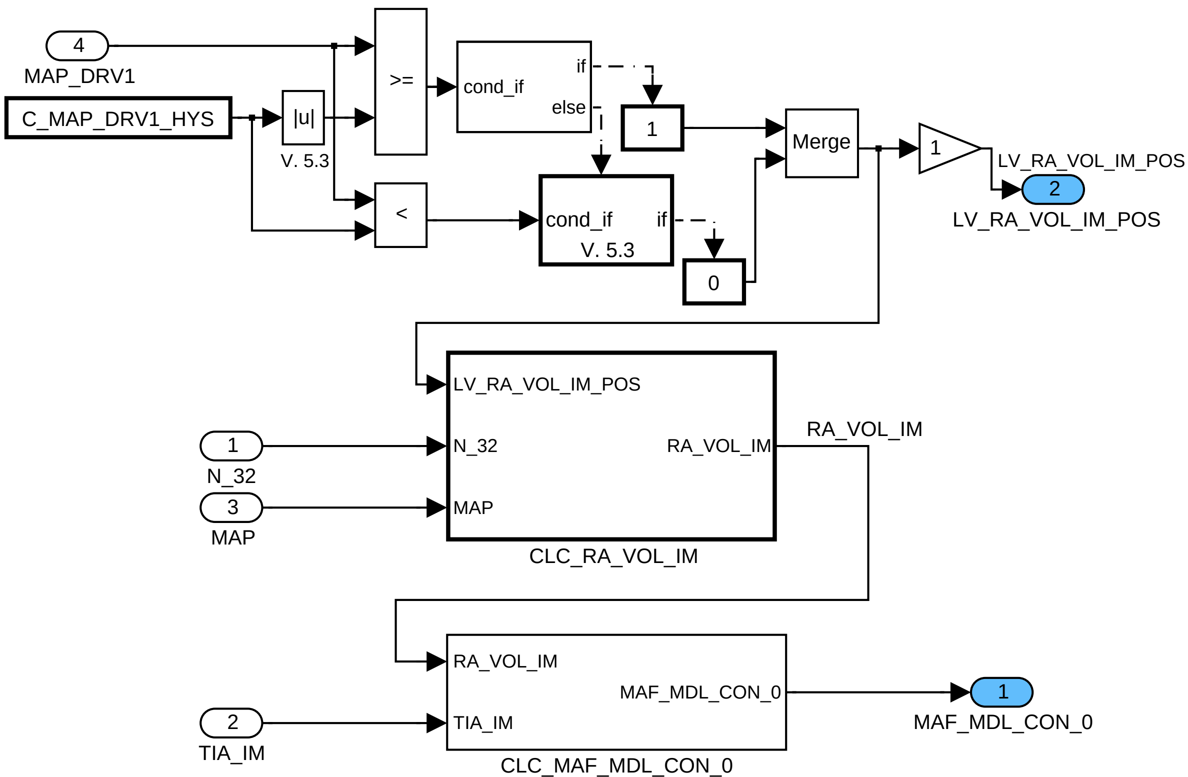


Figure 70:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES_SEG/CLC_MAF_MDL_CON_0

4.10.3.1.1 Calculation of RA_VOL_IM:

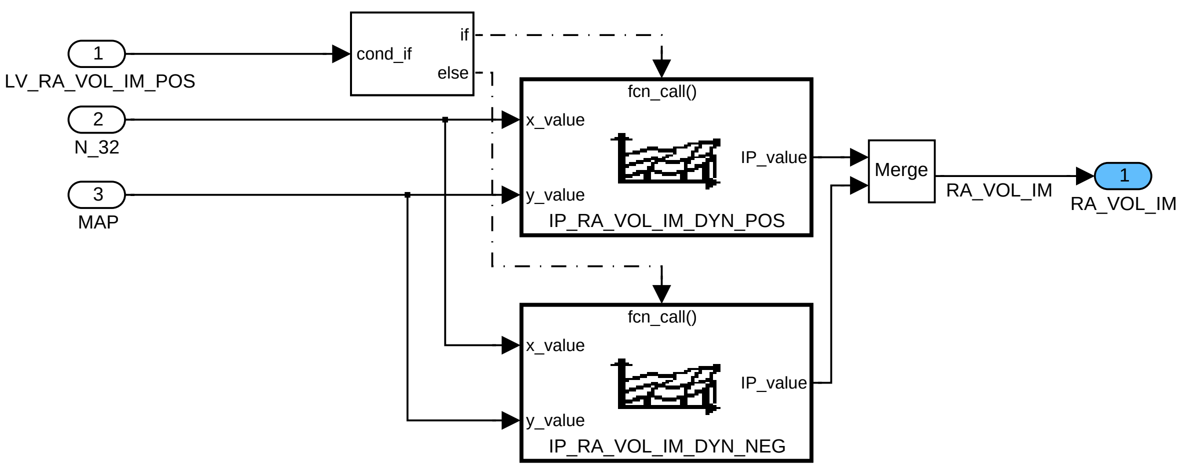


Figure 71:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES_SEG/CLC_MAF_MDL_CON_0/CLC_RA_VOL_IM

4.10.3.1.2 Calculation of MAF_MDL_CON_0:

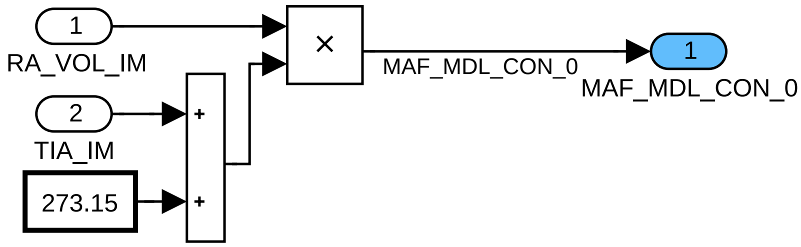


Figure 72:
Path: INSY_DTSYSIMM0/operate__RST_ERU2ES_SEG/CLC_MAF_MDL_CON_0/CLC_MAF_MDL_CON_0

4.10.4 Calculation of segment task

Note: The calculation is done synchronously to multiples of a segment.

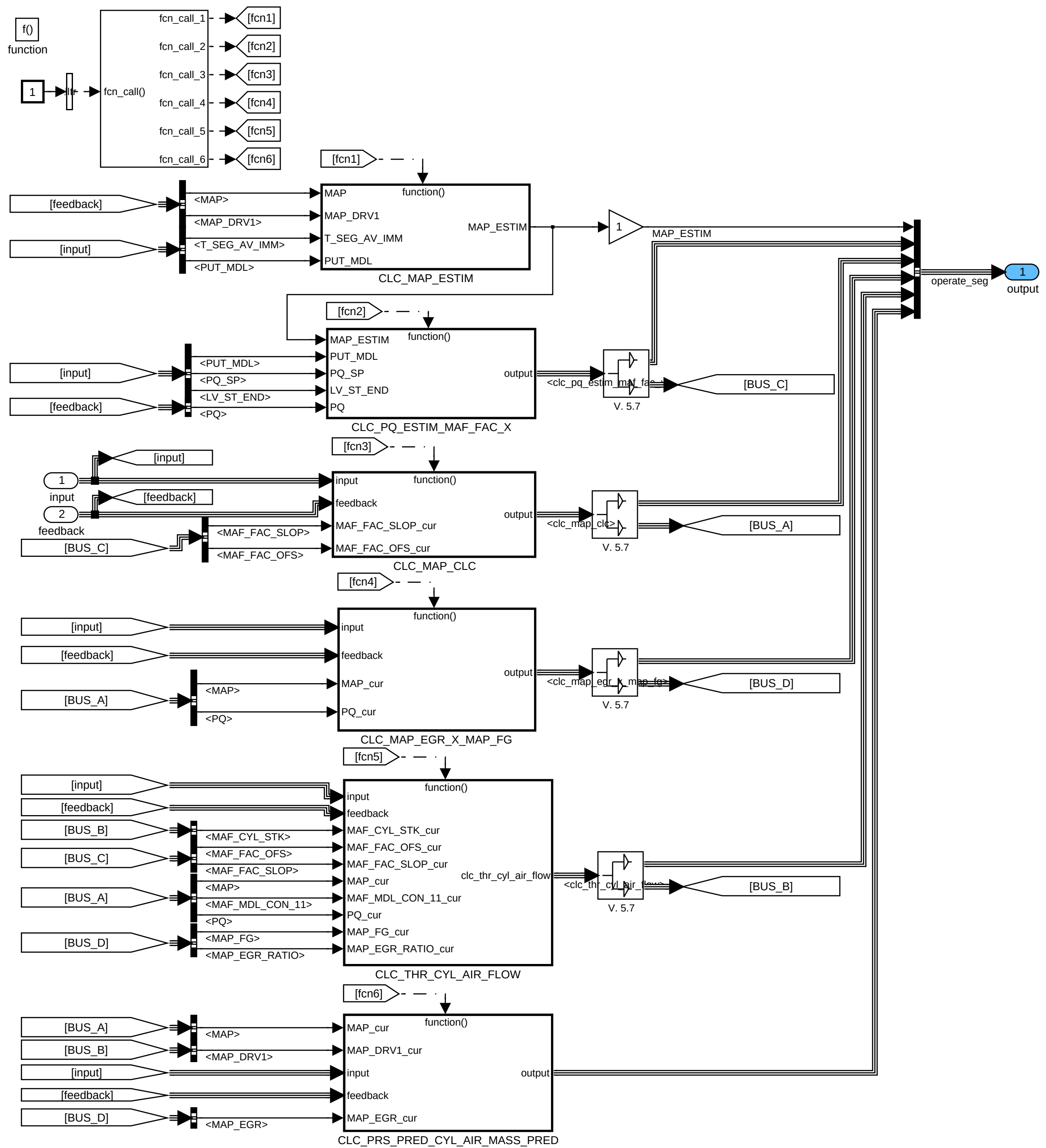


Figure 73:
Path: INSY_DTSYSIMM0/operate__SEG

4.10.4.1 Calculation of MAP_ESTIM:

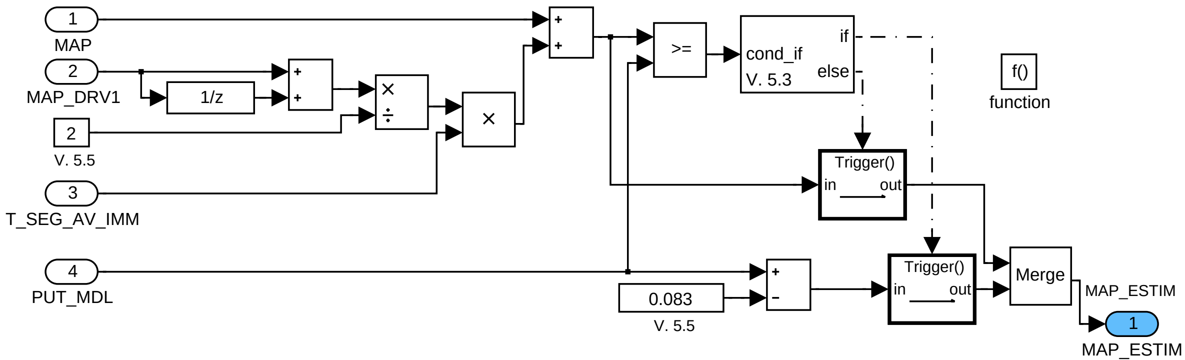


Figure 74:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_ESTIM

4.10.4.2 Estimation of the pressure quotient of PQ_ESTIM:

For the calculation of the manifold air pressure the slope MAF_FAC_SLOP and the offset MAF_FAC_OFS of the linearized air-flow function are necessary. An estimation of the future pressure quotient is done. It is necessary to make a estimation for the pressure quotient at the throttle. MAF_FAC_SLOP and MAF_FAC_OFS are read out of two interpolation table.
Slope MAF_FAC_SLOP and offset MAF_FAC_OFS of the throttle equation are read synchronously to multiples of a segment from the tables ID_MAF_FAC_OFS and ID_MAF_FAC_SLOP. The two maps have the same list of data points (LDPM_PQ_ESTIM_1_INSY).

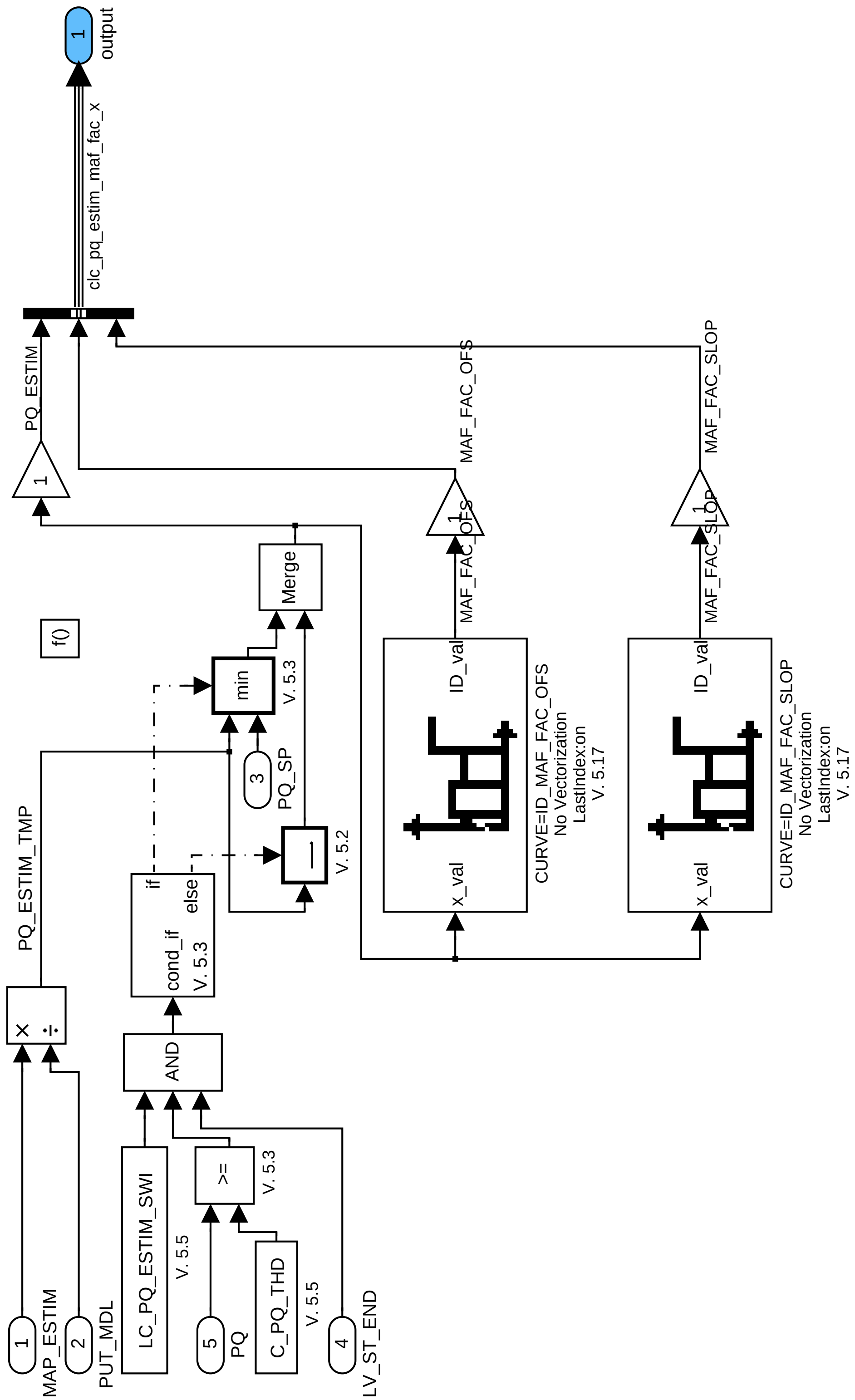


Figure 75:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PQ_ESTIM_MAF_FAC_X

4.10.4.3 Calculation of the manifold pressure MAP

The term of the air flow calculation at the throttle depending on the intake air temperature is stored in the map IP_MAF_MDL_CON_1. This interpolation can be done in the time schedule (once a second).
Note: MAF_MDL_CON_1 has to be calculated at reset for the first time.

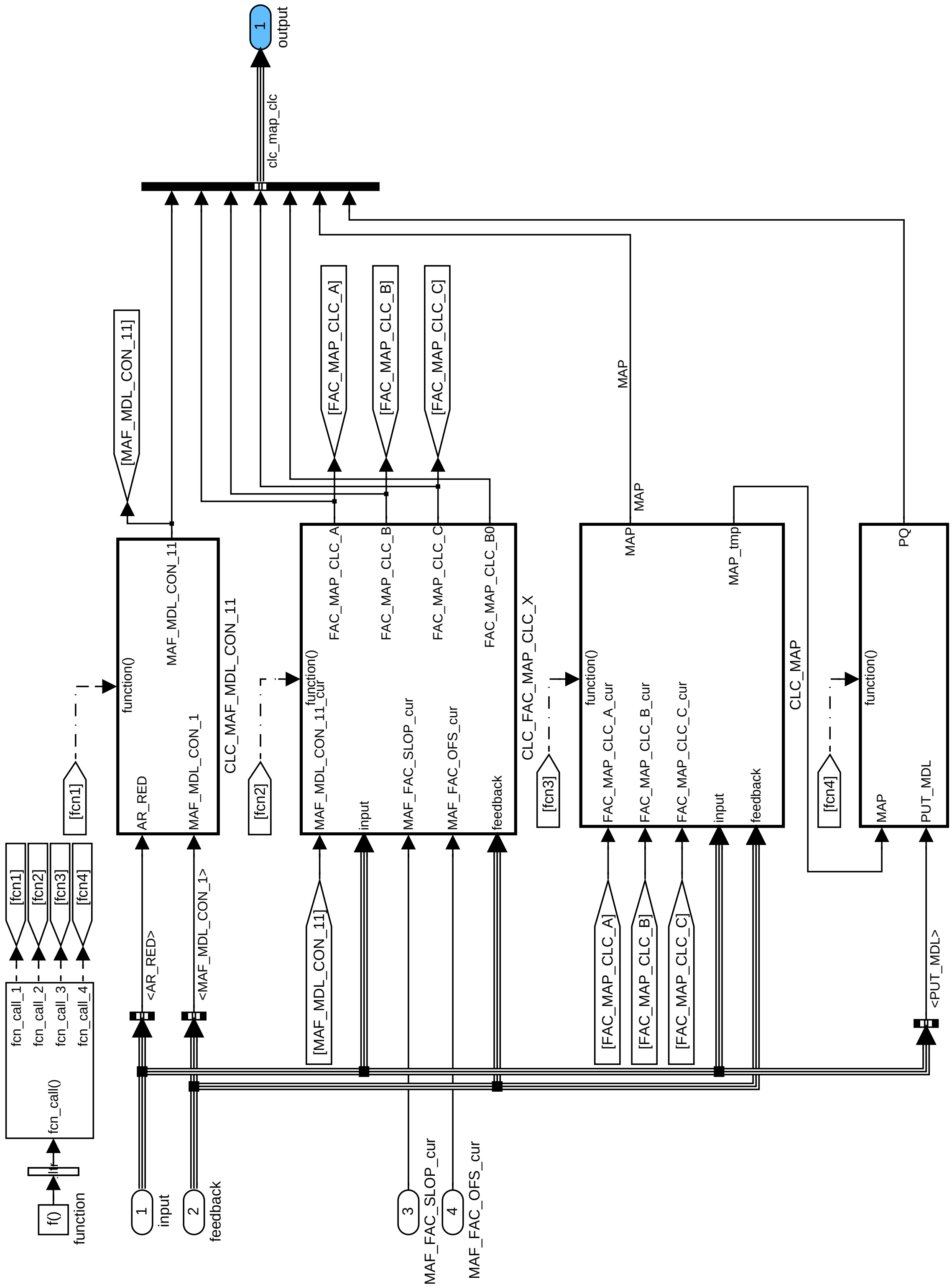


Figure 76:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC

4.10.4.3.1 Calculation of MAF_MDL_CON_11

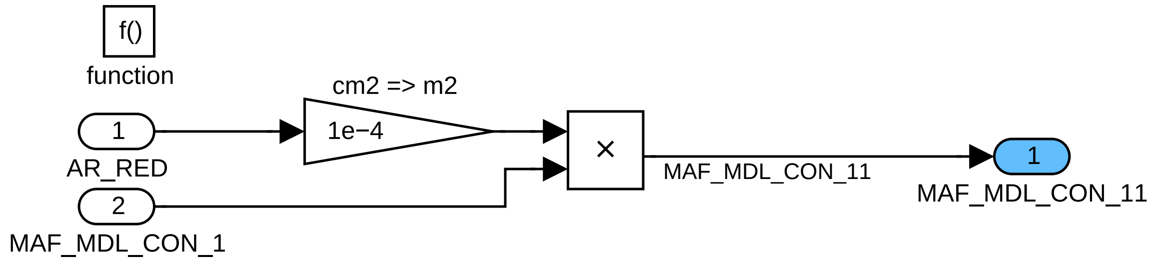


Figure 77:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_MAF_MDL_CON_11

4.10.4.3.2 Calculation of temporary variables FAC_MAF_X

Calculation of the temporary variables MAF_FAC_CLC_A, MAF_FAC_CLC_B, MAF_FAC_CLC_B0 and MAF_FAC_CLC_C.

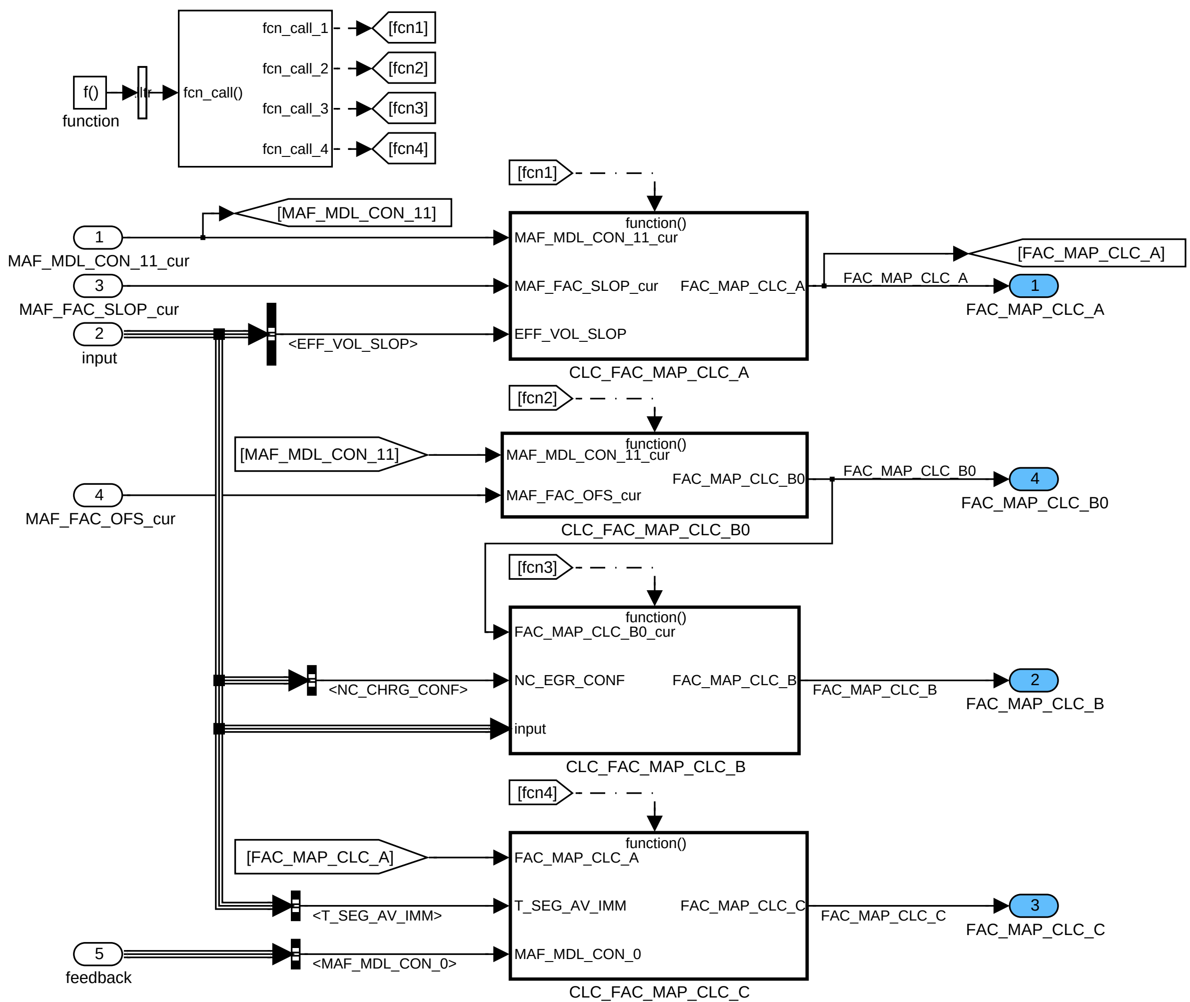


Figure 78:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X

4.10.4.3.2.1 Calculation of FAC_MAP_CLC_A

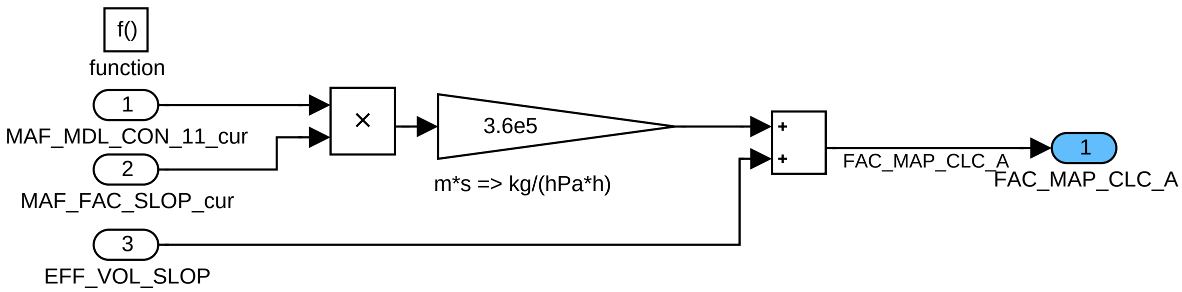


Figure 79:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_A

4.10.4.3.2.2 Calculation of FAC_MAP_CLC_B0

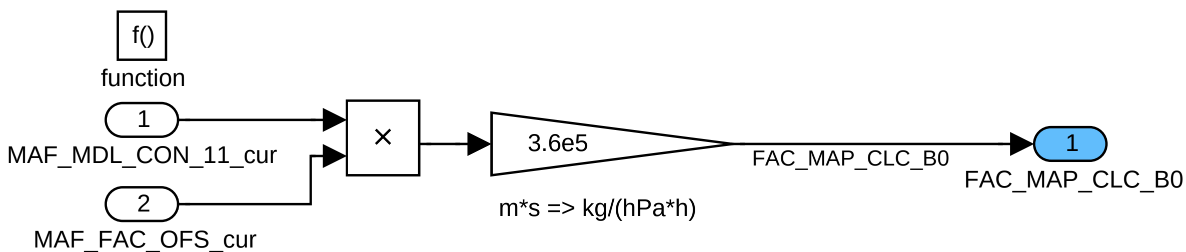


Figure 80:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_B0

4.10.4.3.2.3 Calculation of FAC_MAP_CLC_B

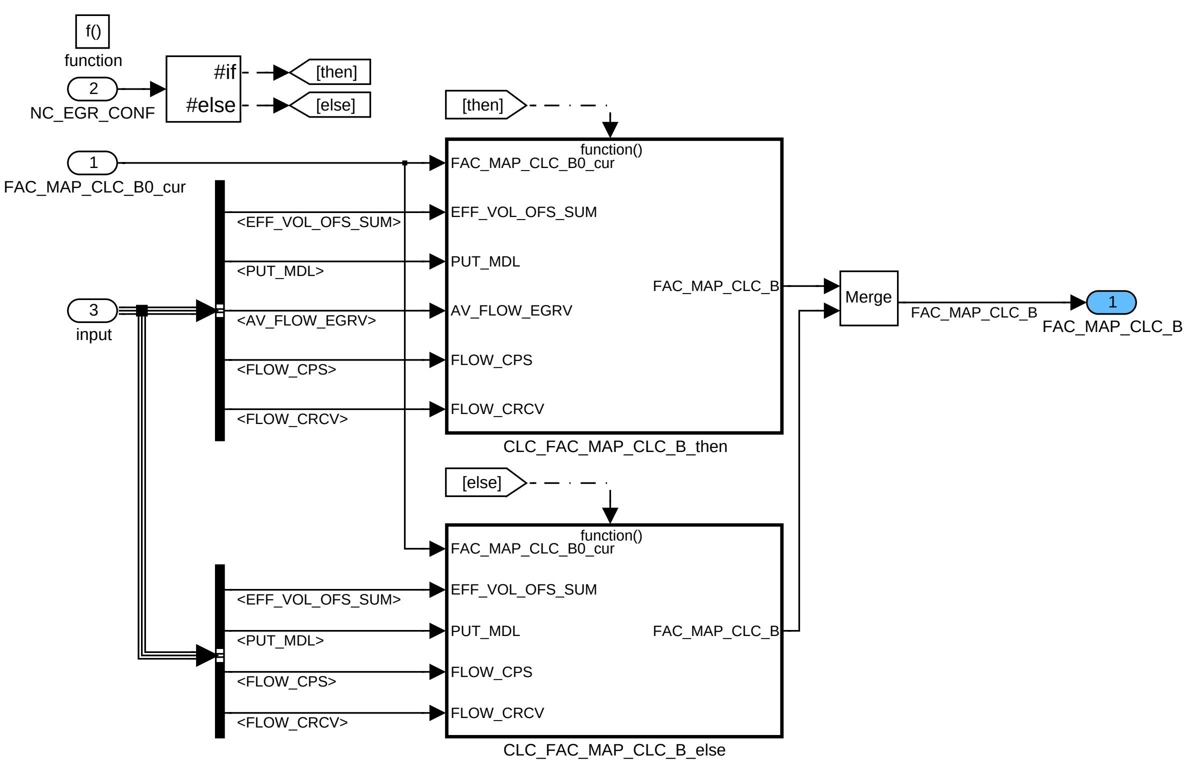


Figure 81:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_B

4.10.4.3.2.3.1 Calculation of FAC_MAP_CLC_B - then branch

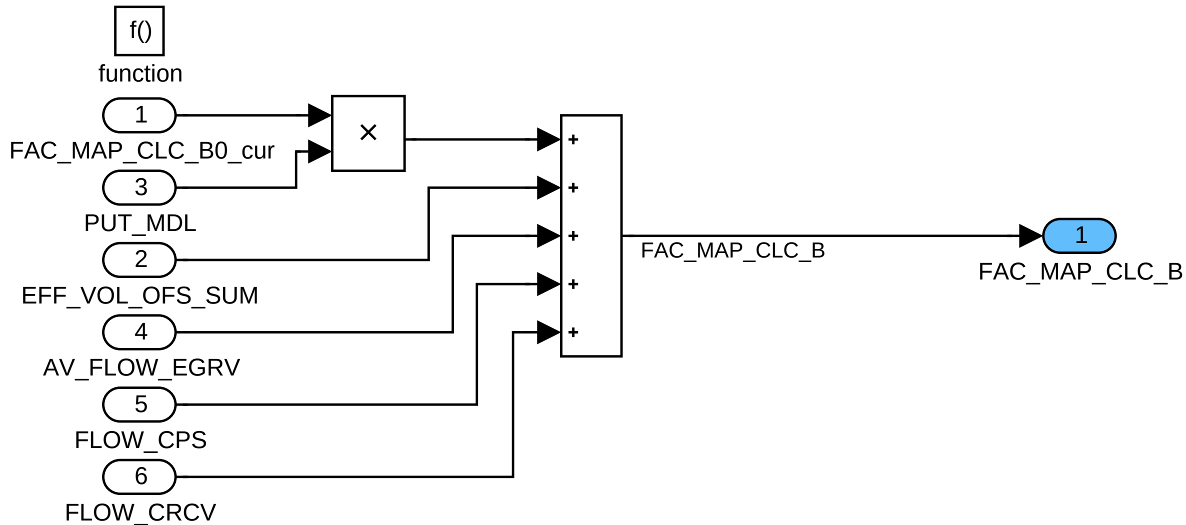


Figure 82:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_B/CLC_FAC_MAP_CLC_B_then

4.10.4.3.2.3.2 Calculation of FAC_MAP_CLC_B - else branch

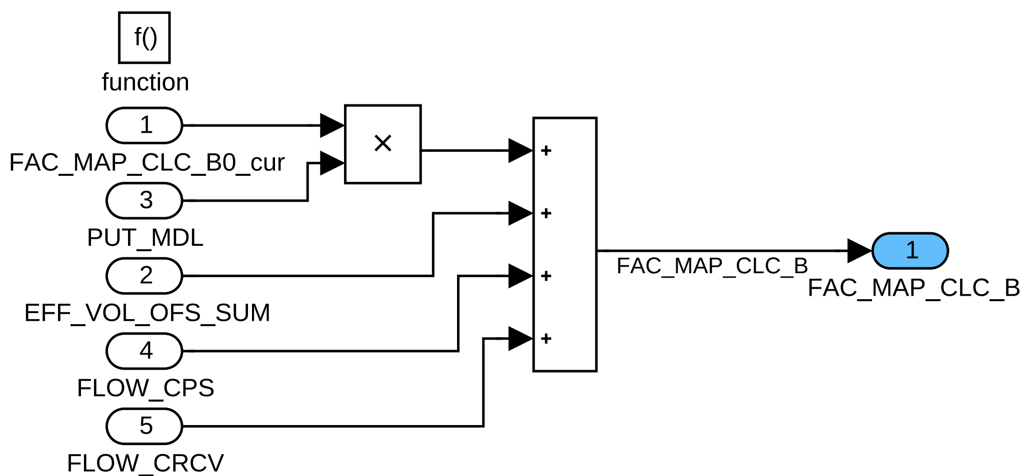


Figure 83:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_B/CLC_FAC_MAP_CLC_B_else

4.10.4.3.2.4 Calculation of FAC_MAP_CLC_C:

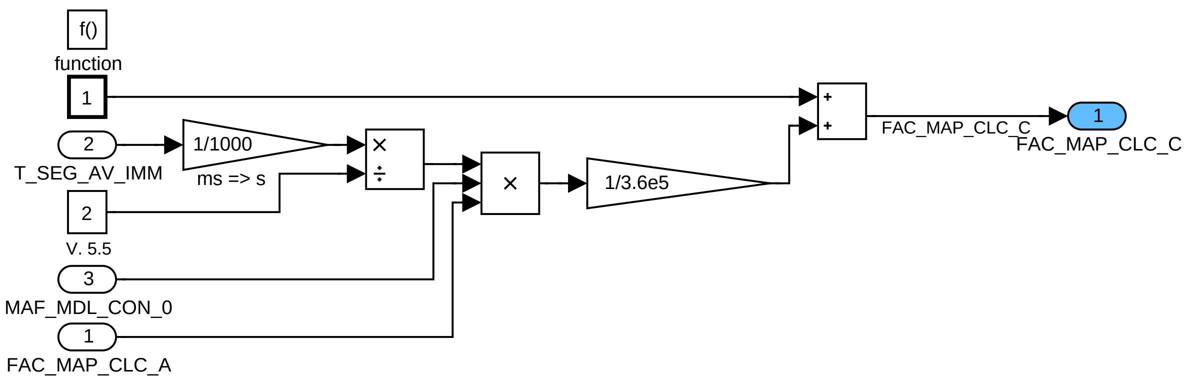


Figure 84:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_FAC_MAP_CLC_X/CLC_FAC_MAP_CLC_C

4.10.4.3.3 Calculation of MAP - overview:

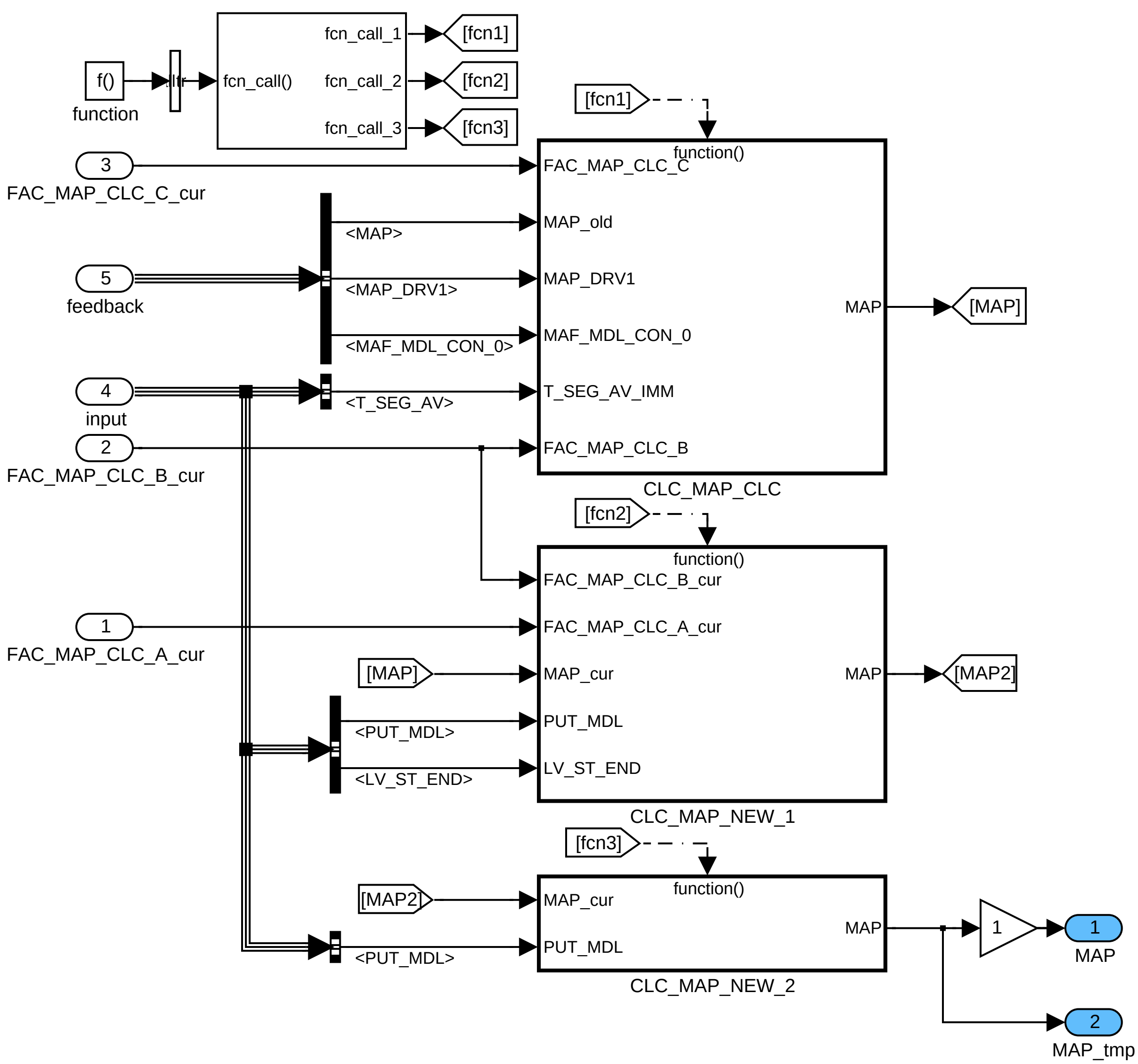


Figure 85:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_MAP

4.10.4.3.3.1 Calculation of MAP:

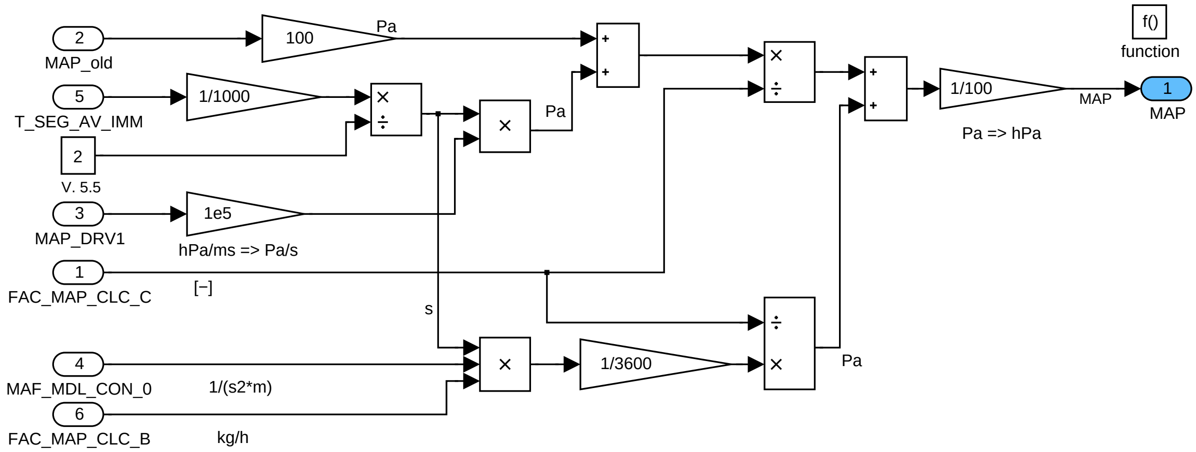


Figure 86:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_MAP/CLC_MAP_CLC

4.10.4.3.3.2 Calculation of MAP_NEW_1

If the engine's state is not "engine start" then during full load MAP is calculated as followed:

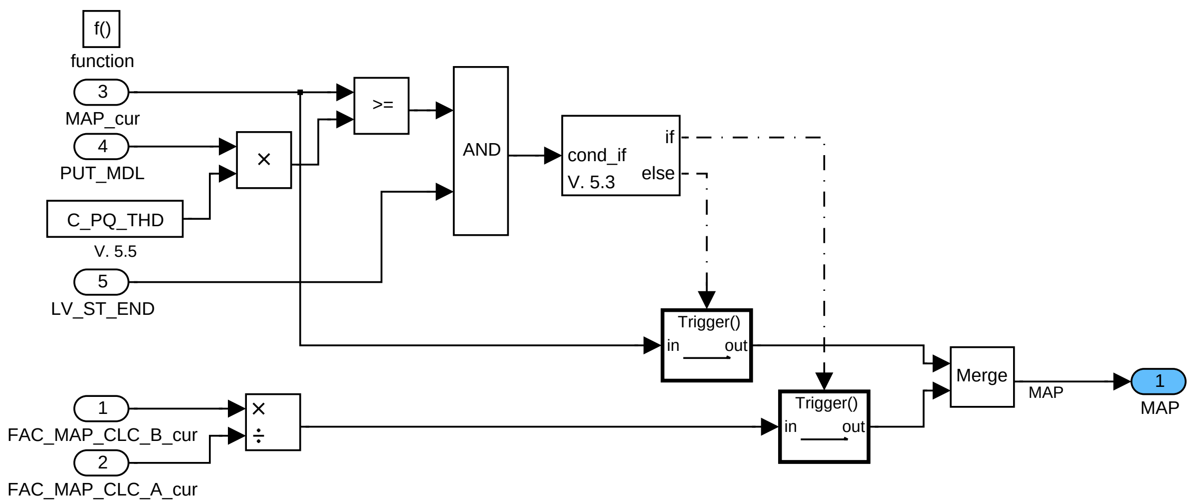


Figure 87:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_MAP/CLC_MAP_NEW_1

4.10.4.3.3.3 Calculation of MAP_NEW_2

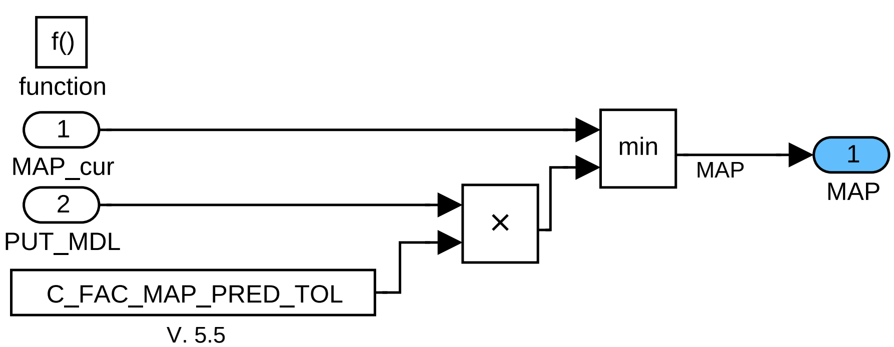


Figure 88:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_MAP/CLC_MAP_NEW_2

4.10.4.3.4 Calculation of PQ

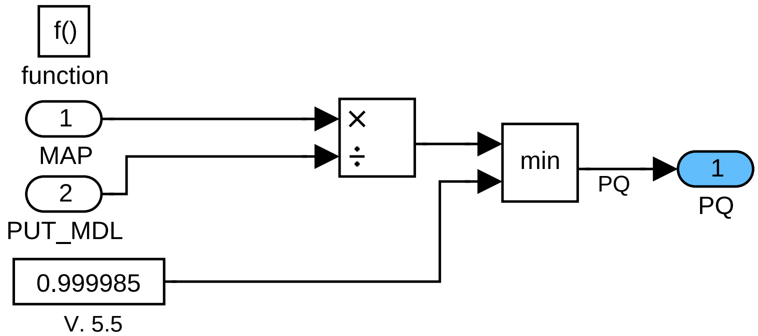


Figure 89:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_CLC/CLC_PQ

4.10.4.4 Calculation of MAP_EGR, MAP_EGR_DRV1, MAP_FG, and MAP_EGR_RATIO

Most parts of this chapter have only to calculated if NC_EGR_CONF = 1. As the total pressure in the manifold is known, it is possible to calculate the mass flow at the EGR valve and therefore the residual gas pressure in the intake manifold. With the residual gas pressure and the total manifold pressure the fresh gas partial pressure can be calculated.

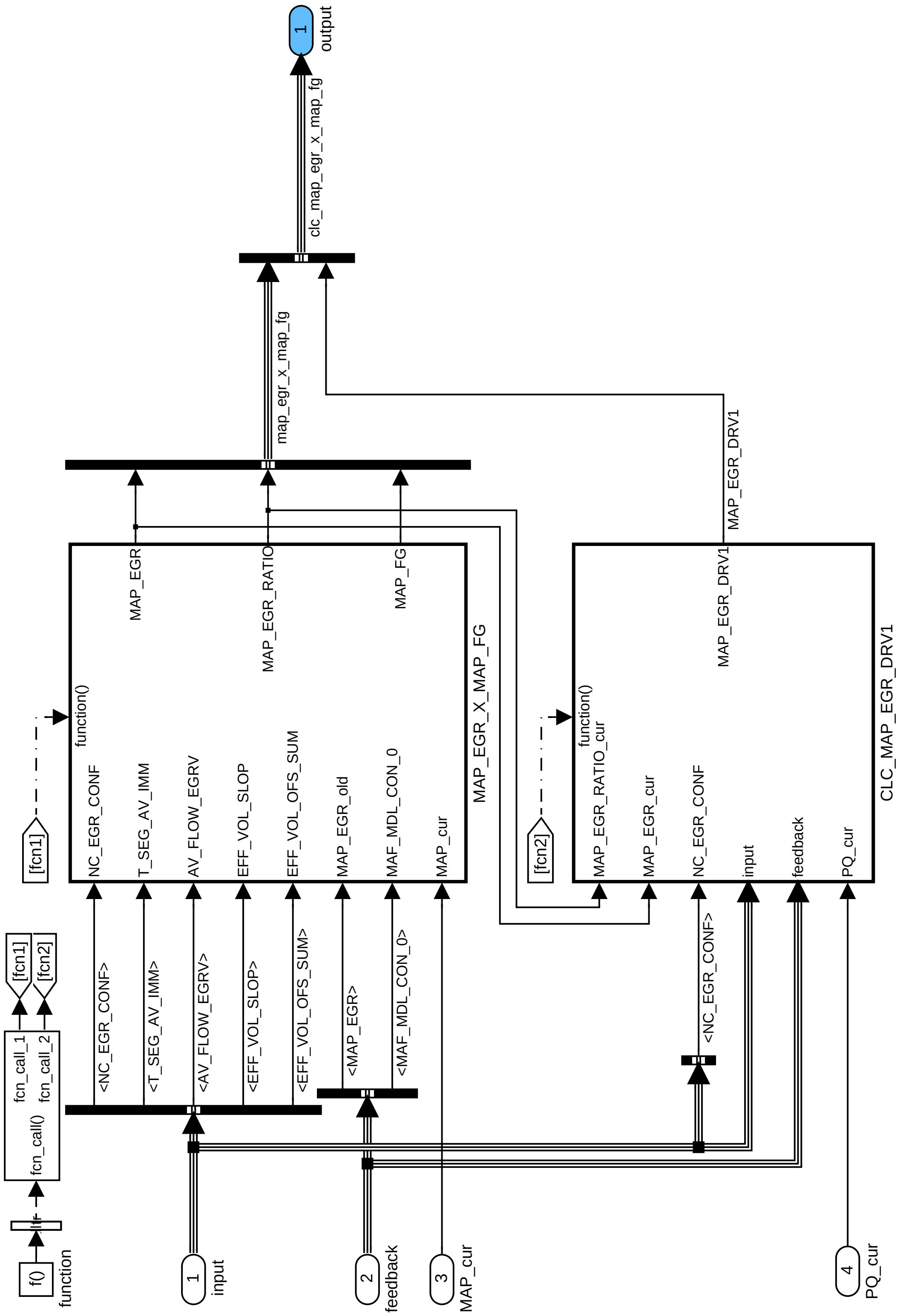


Figure 90:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_EGR_X_MAP_FG

4.10.4.4.1 Calculation of MAP_EGR_X_MAP_FG

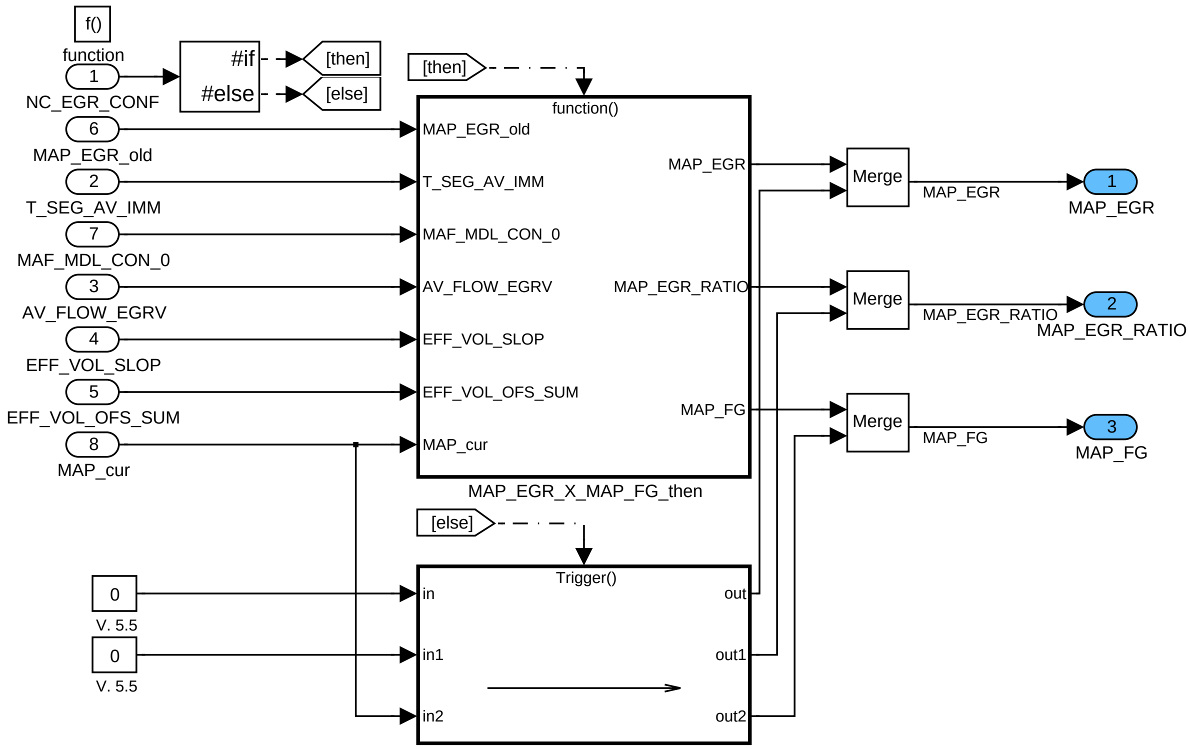


Figure 91:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_EGR_X_MAP_FG/MAP_EGR_X_MAP_FG

4.10.4.4.1.1 Calculation of MAP_EGR_X_MAP_FG - then branch

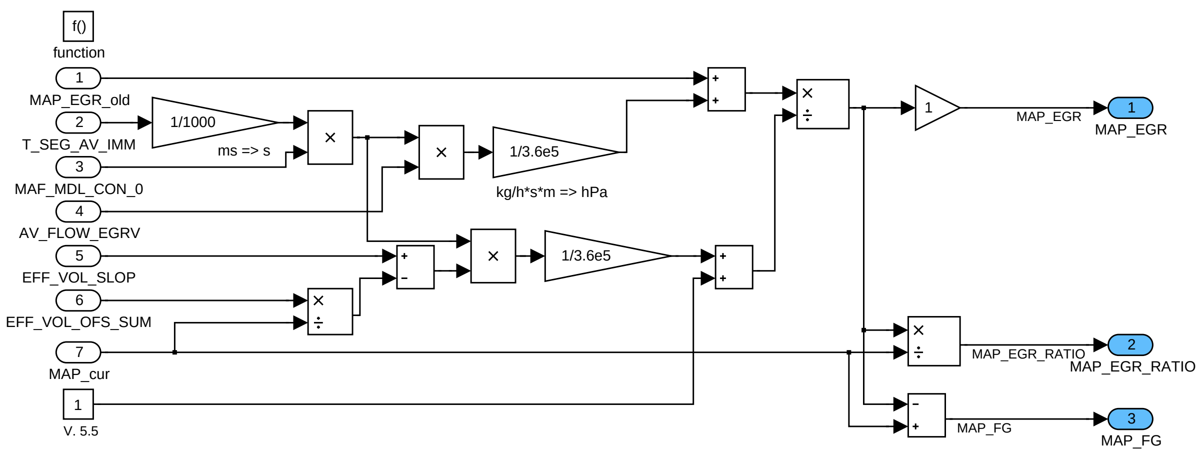


Figure 92:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_EGR_X_MAP_FG/MAP_EGR_X_MAP_FG/MAP_EGR_X_MAP_FG_then

4.10.4.4.2 Calculation of CLC_MAP_EGR_DRV1

The first derivative of the residual gas partial pressure calculated as follows:

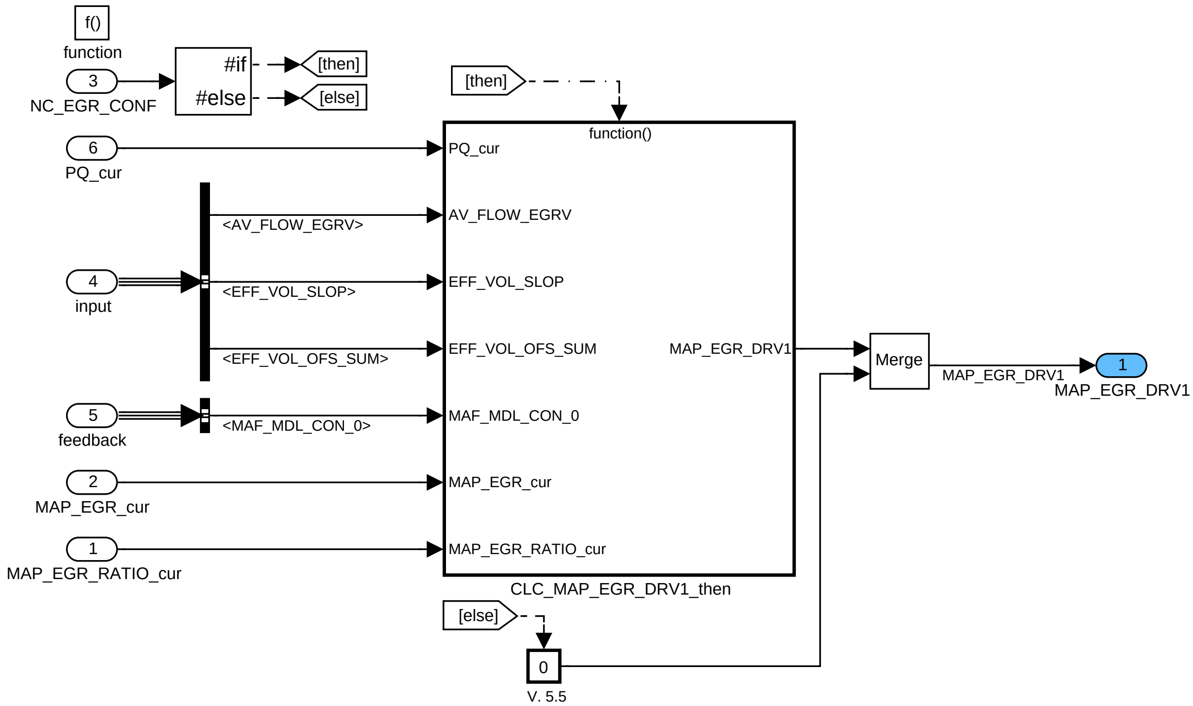


Figure 93:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_EGR_X_MAP_FG/CLC_MAP_EGR_DRV1

4.10.4.4.2.1 Calculation of MAP_EGR_DRV1 - then branch

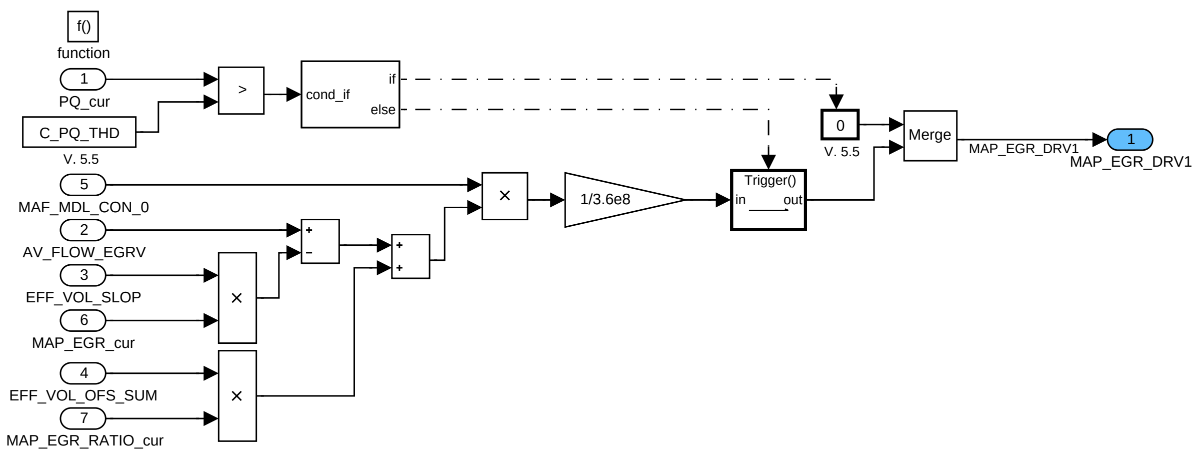


Figure 94:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_MAP_EGR_X_MAP_FG/CLC_MAP_EGR_DRV1/CLC_MAP_EGR_DRV1_then

4.10.4.5 Calculation of the throttle and cylinder air flows

With the new manifold pressure it is possible to calculate the actual gas flows into and out of the manifold. Further for the next step the change of manifold pressure is calculated. The complete gas flow into the cylinder is:

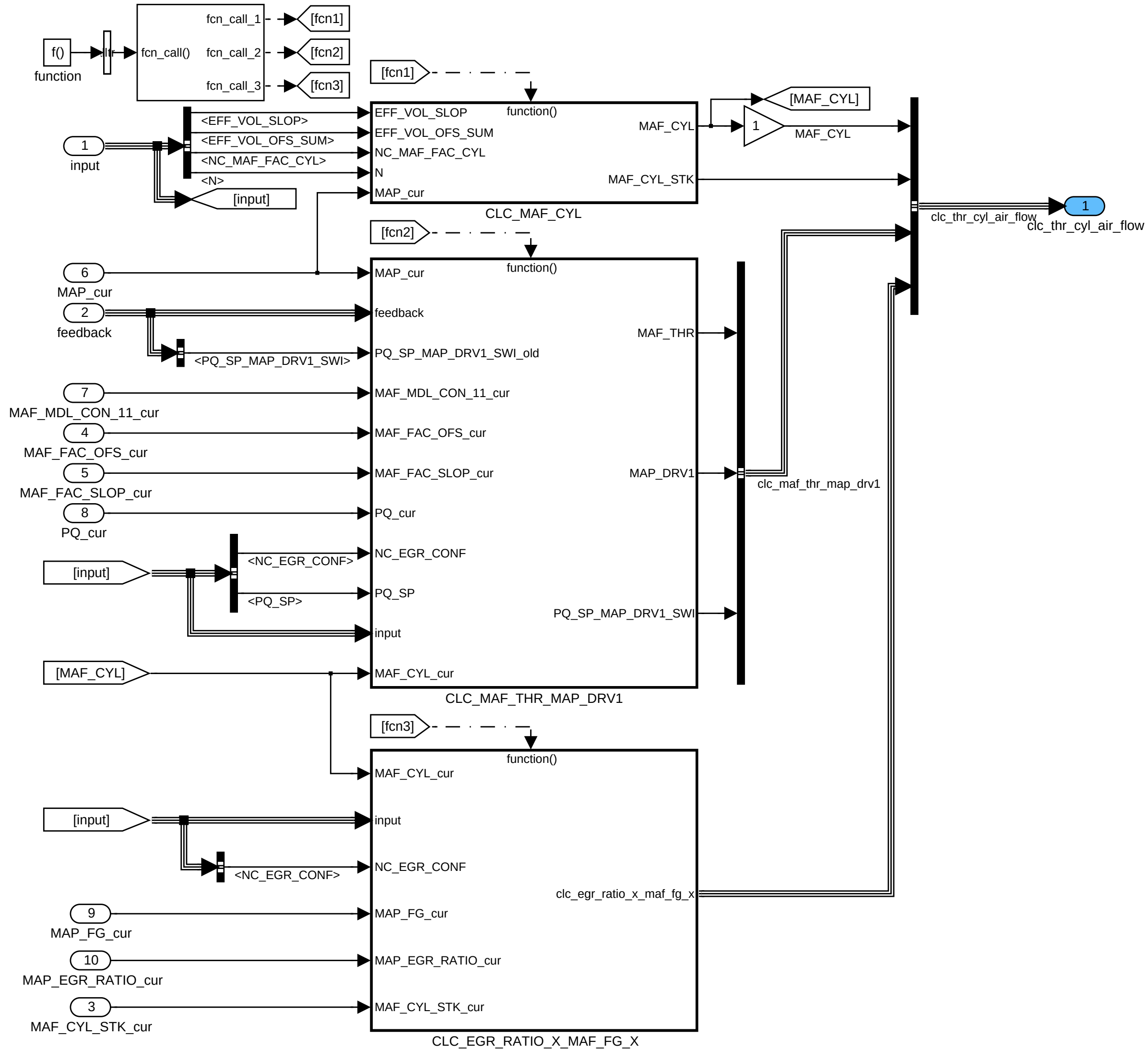


Figure 95:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW

4.10.4.5.1 Calculation of MAF_CYL

For taking into account pumping losses of the engine the segment synchronous variable MAF_CYL_STK is needed (in module TQ_LOSS).

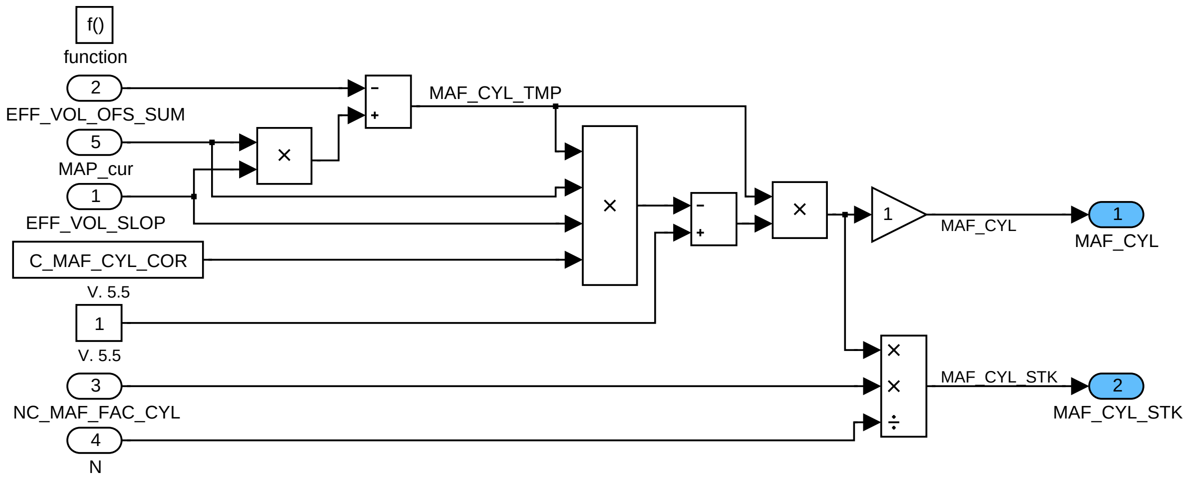


Figure 96:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_CYL

4.10.4.5.2 Calculation of MAF_THR and MAP_DRV1

The air-flow at the throttle and the pressure change in the manifold are calculated as:

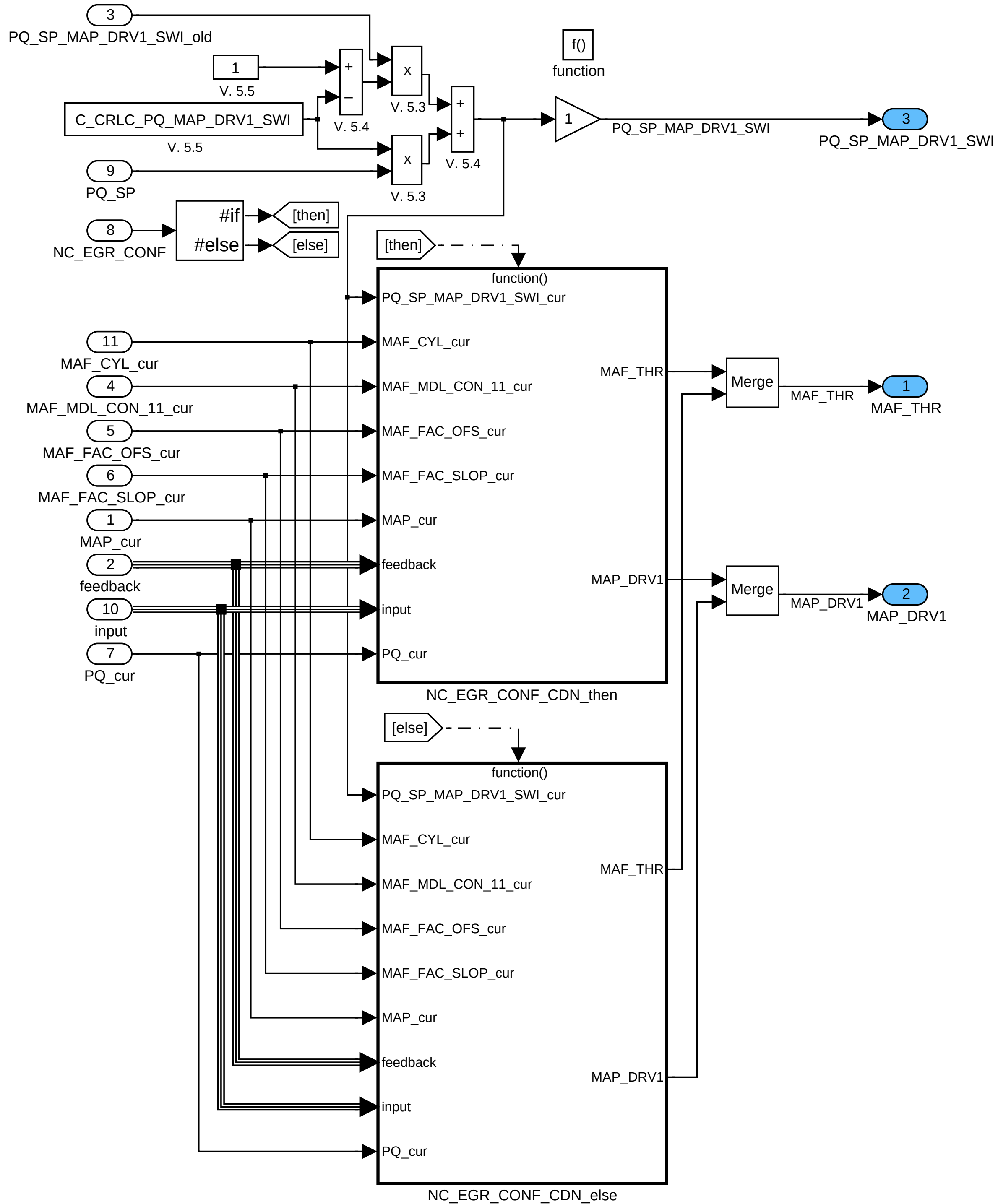


Figure 97:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1

4.10.4.5.2.1 Calculation of MAF_THR and MAP_DRV1 - then branch

For systems with EGR.

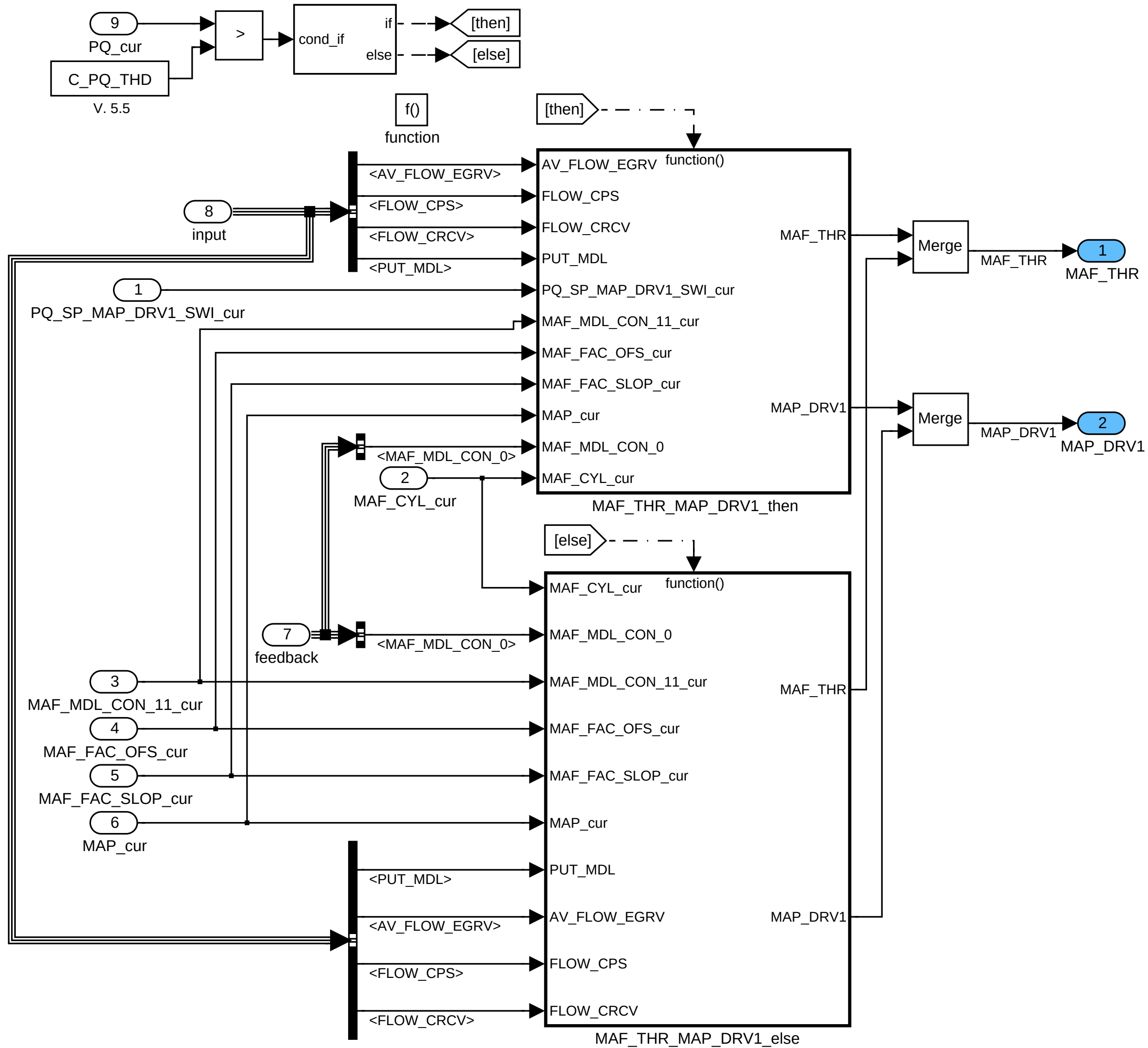


Figure 98:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_then

4.10.4.5.2.1.1 Calculation of MAF_THR and MAP_DRV1 if PQ is above a certain threshold:

Calibration hint: C_PQ_SP_MAP_DRV1_SWI should be calibrated so that its value is C_PQ_SP_MAX minus a hysteresis.

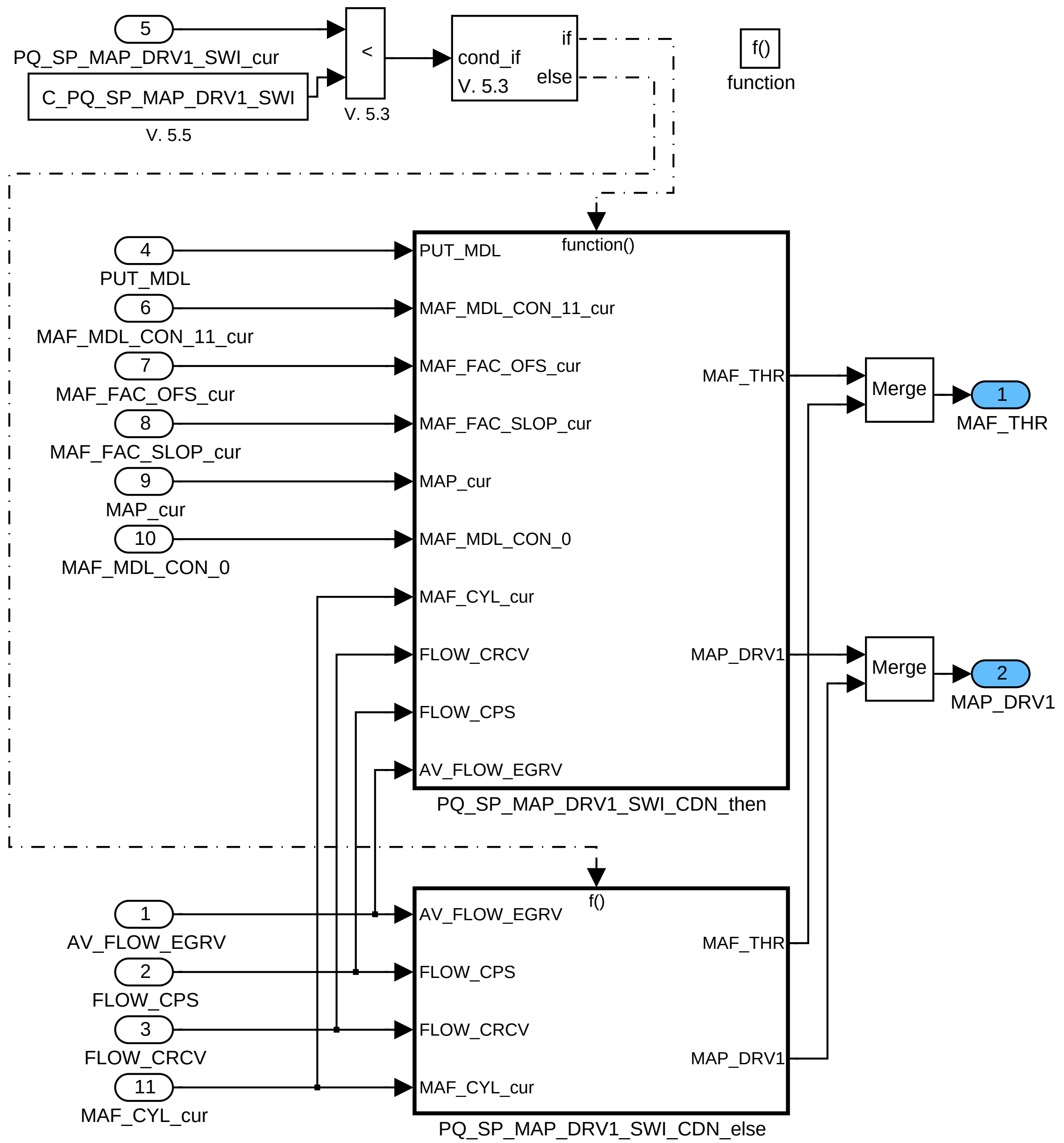


Figure 99:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_then/MAF_THR_MAP_DRV1_then

4.10.4.5.2.1.1.1 Calculation of MAF_THR and MAP_DRV1 if PQ_SP_MAP_DRV1_SWI condition is active:

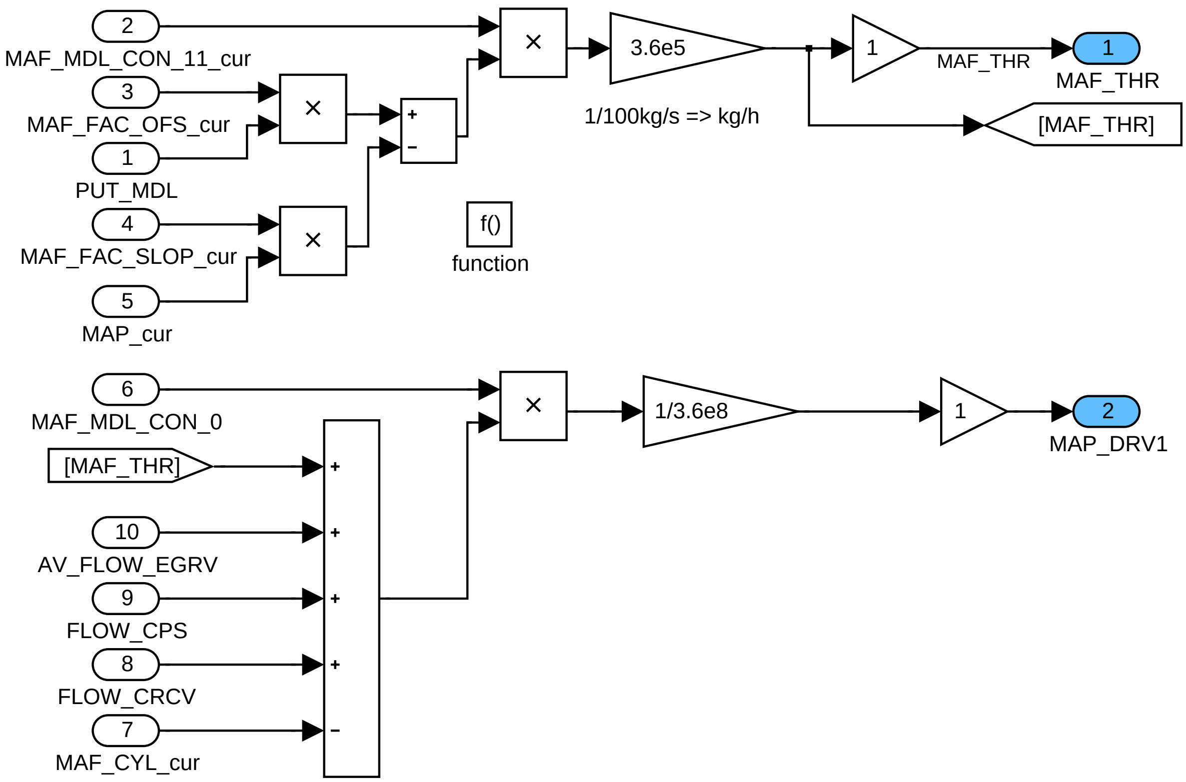


Figure 100:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_then/MAF_THR_MAP_DRV1_then/PQ_SP_MAP_DRV1_SWI_CDN_then

4.10.4.5.2.1.1.2 Calculation of MAF_THR and MAP_DRV1 if PQ_SP_MAP_DRV1_SWI condition is not active:

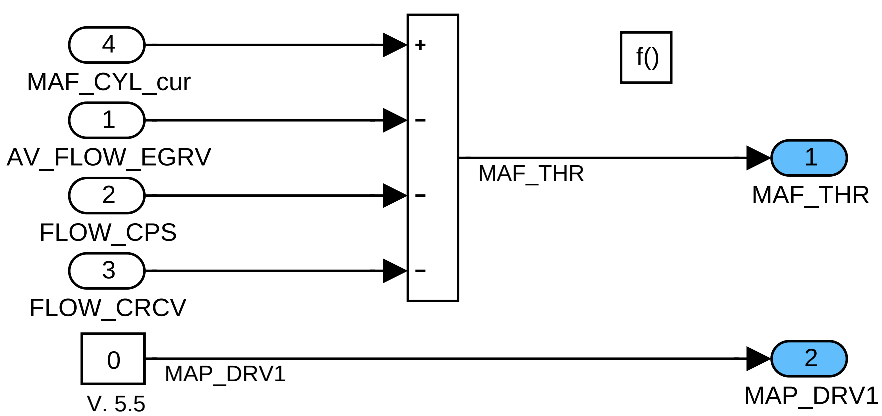


Figure 101:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_then/MAF_THR_MAP_DRV1_then/PQ_SP_MAP_DRV1_SWI_CDN_else

4.10.4.5.2.1.2 Calculation of MAF_THR and MAP_DRV1 if PQ is under or equal to a certain threshold:

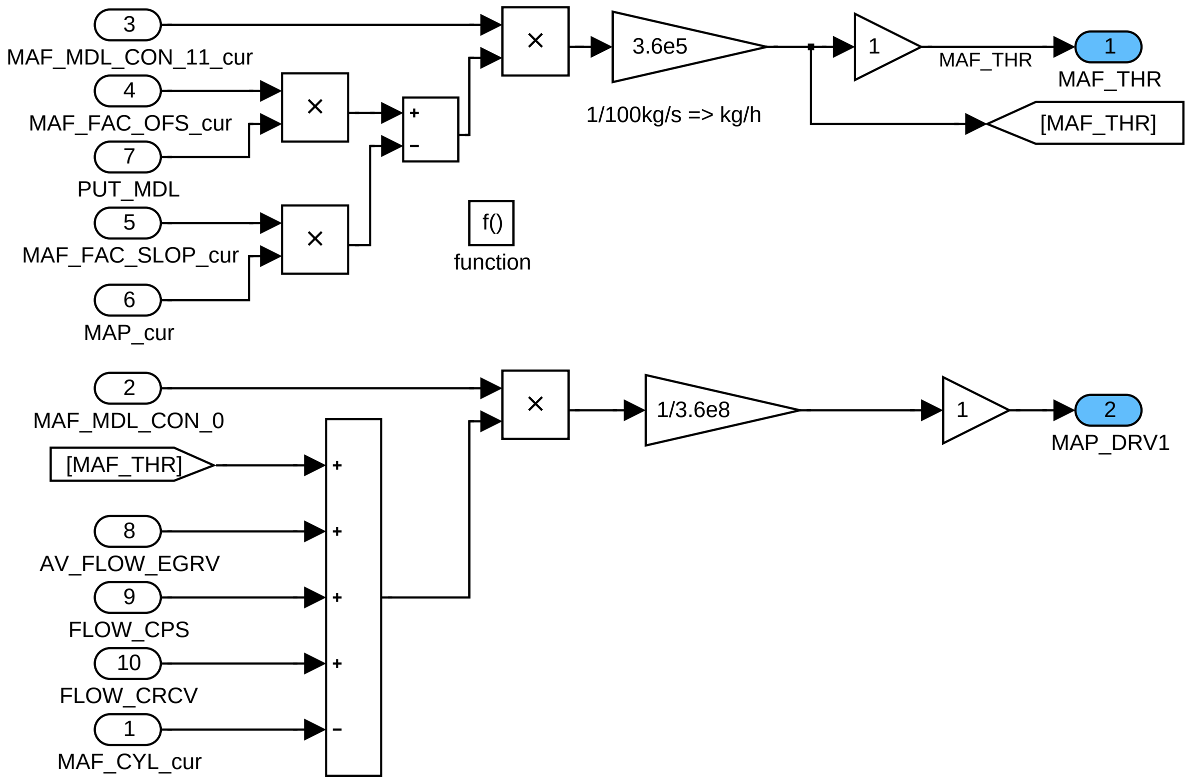


Figure 102:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_then/MAF_THR_MAP_DRV1_else

4.10.4.5.2.2 Calculation of MAF_THR and MAP_DRV1 - else branch

For systems without EGR.

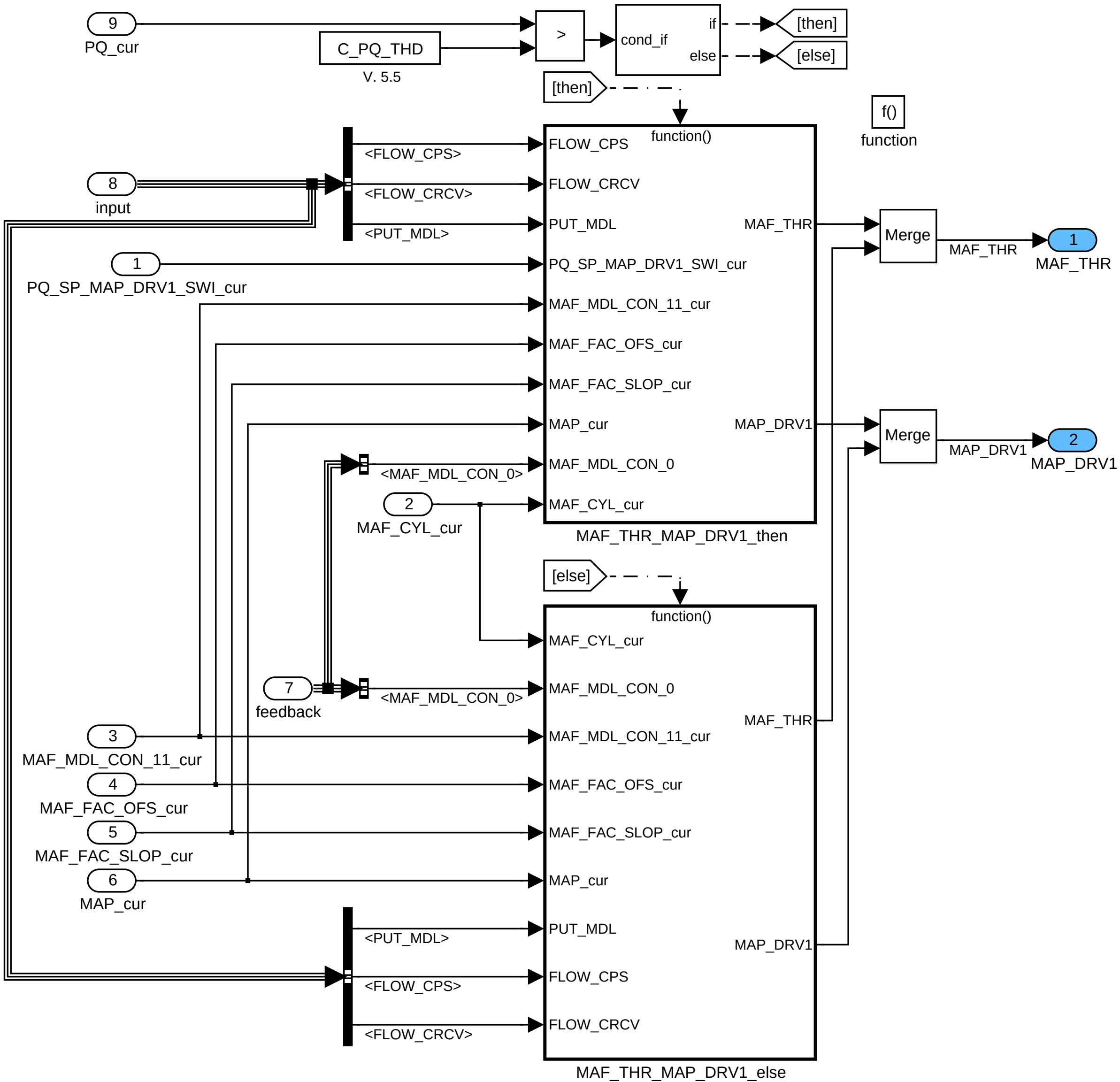


Figure 103:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_else

4.10.4.5.2.2.1 Calculation of MAF_THR and MAP_DRV1 if PQ is above a certain threshold:

Calibration hint: C_PQ_SP_MAP_DRV1_SWI should be calibrated so that its value is C_PQ_SP_MAX minus a hysteresis.

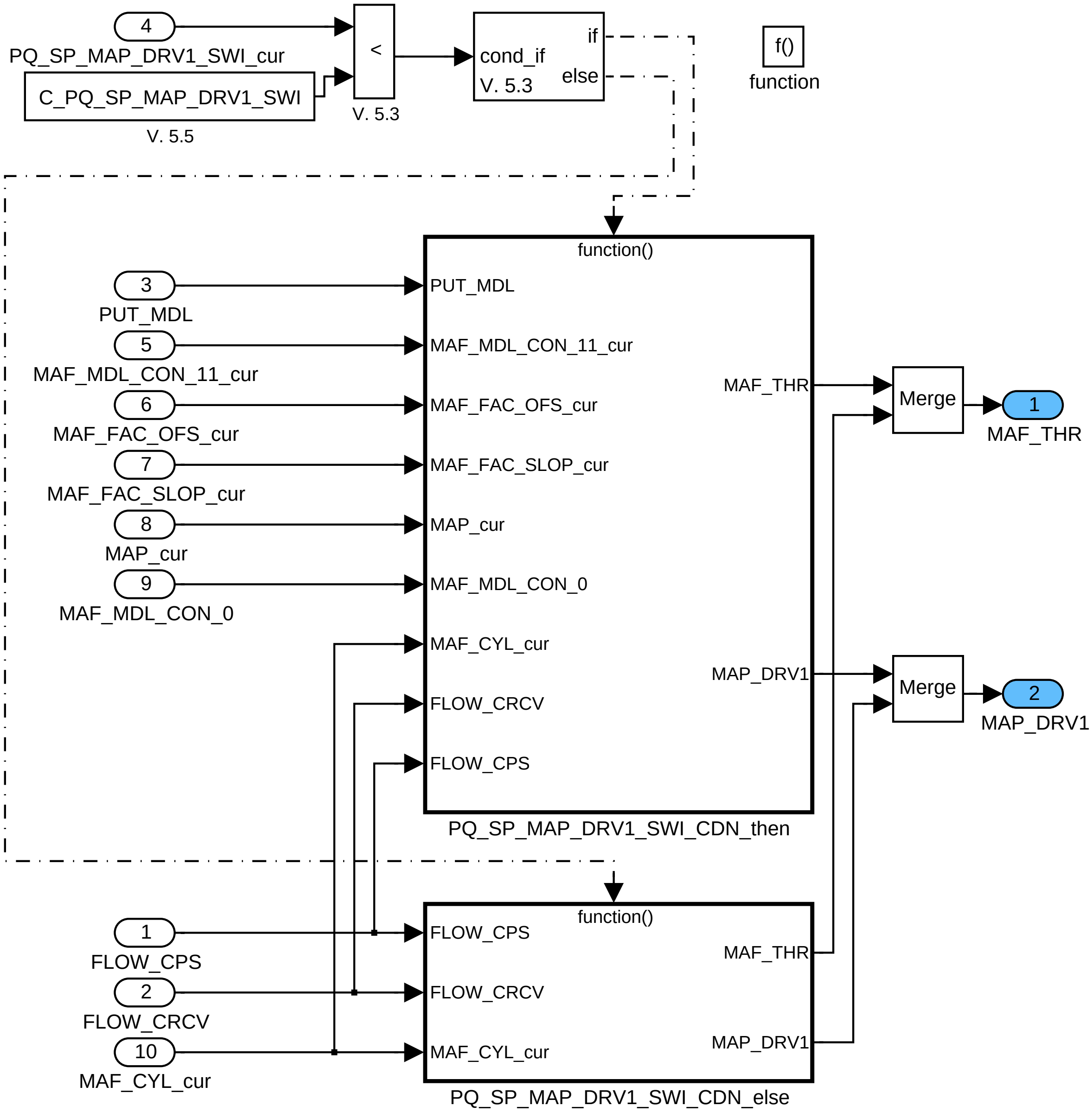


Figure 104:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_else/MAF_THR_MAP_DRV1_then

4.10.4.5.2.2.1.1 Calculation of MAF_THR and MAP_DRV1 if PQ_SP_MAP_DRV1_SWI condition is active:

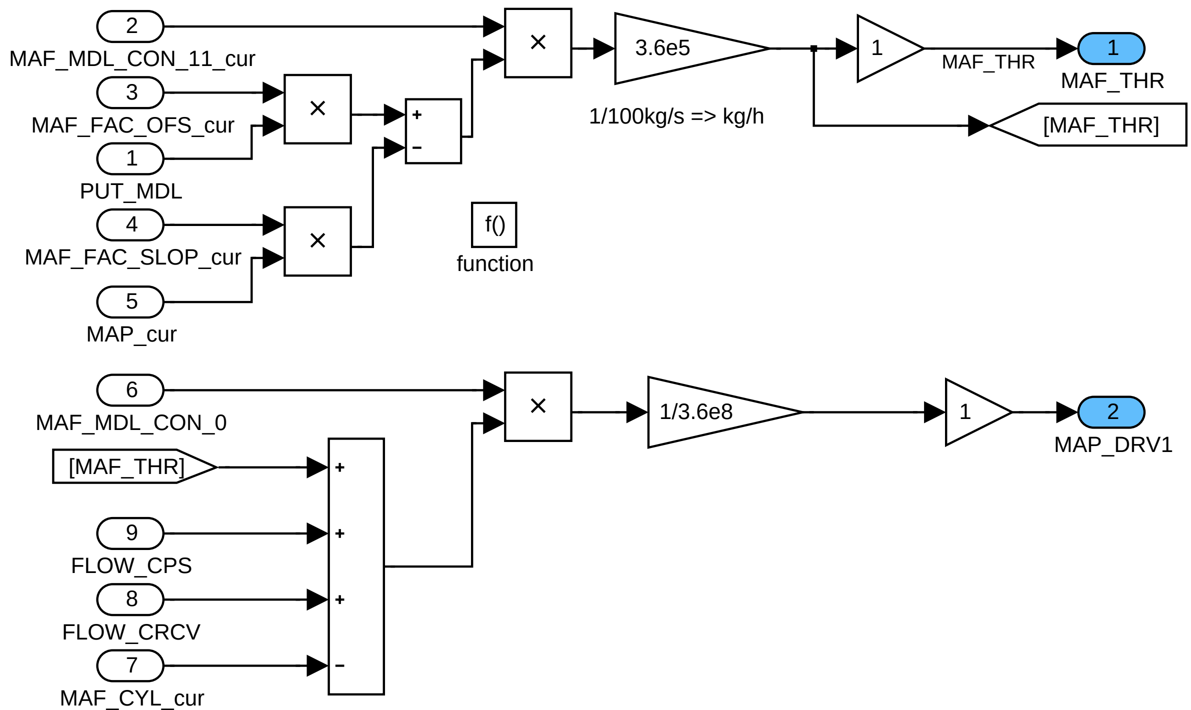


Figure 105:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_else/MAF_THR_MAP_DRV1_then/PQ_SP_MAP_DRV1_SWI_CDN_then

4.10.4.5.2.2.1.2 Calculation of MAF_THR and MAP_DRV1 if PQ_SP_MAP_DRV1_SWI condition is not active:

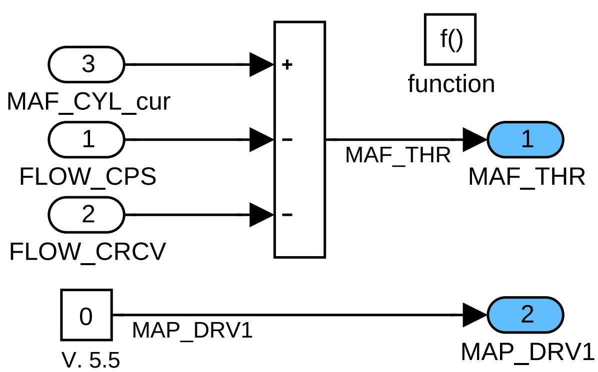


Figure 106:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_else/MAF_THR_MAP_DRV1_then/PQ_SP_MAP_DRV1_SWI_CDN_else

4.10.4.5.2.2.2 Calculation of MAF_THR and MAP_DRV1 if PQ is under or equal to a certain threshold:

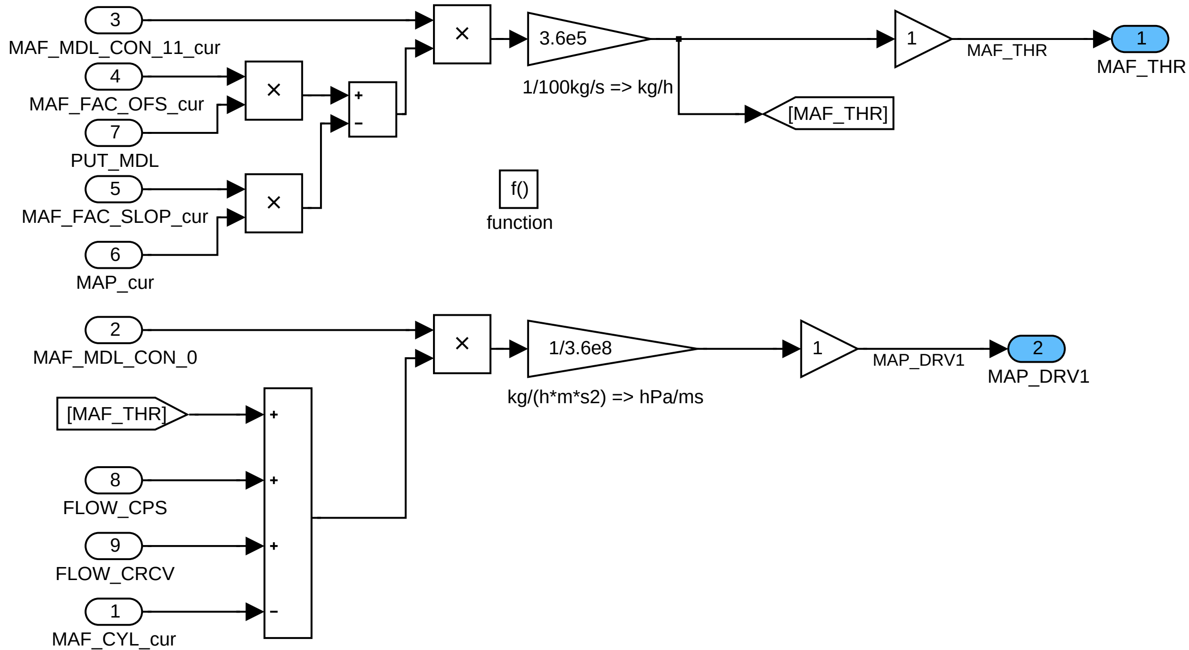


Figure 107:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_MAF_THR_MAP_DRV1/NC_EGR_CONF_CDN_else/MAF_THR_MAP_DRV1_else

4.10.4.5.3 Calculation of EGR_RATIO_X and MAF_FG_X

It is possible to calculate the EGR-rate EGR_RATIO out of the total air mass flow into the cylinder MAF_CYL and the air mass flow at the EGR valve.

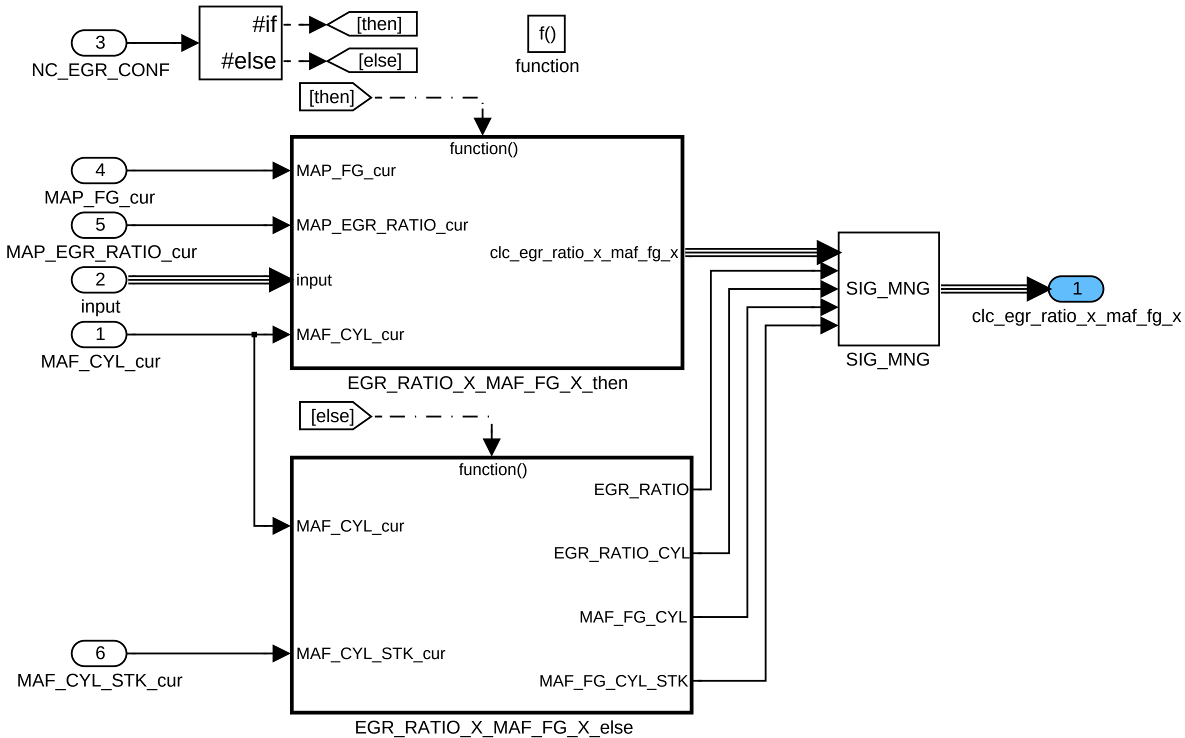


Figure 108:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X

4.10.4.5.3.1 EGR_RATIO_X and MAF_FG_X - then branch

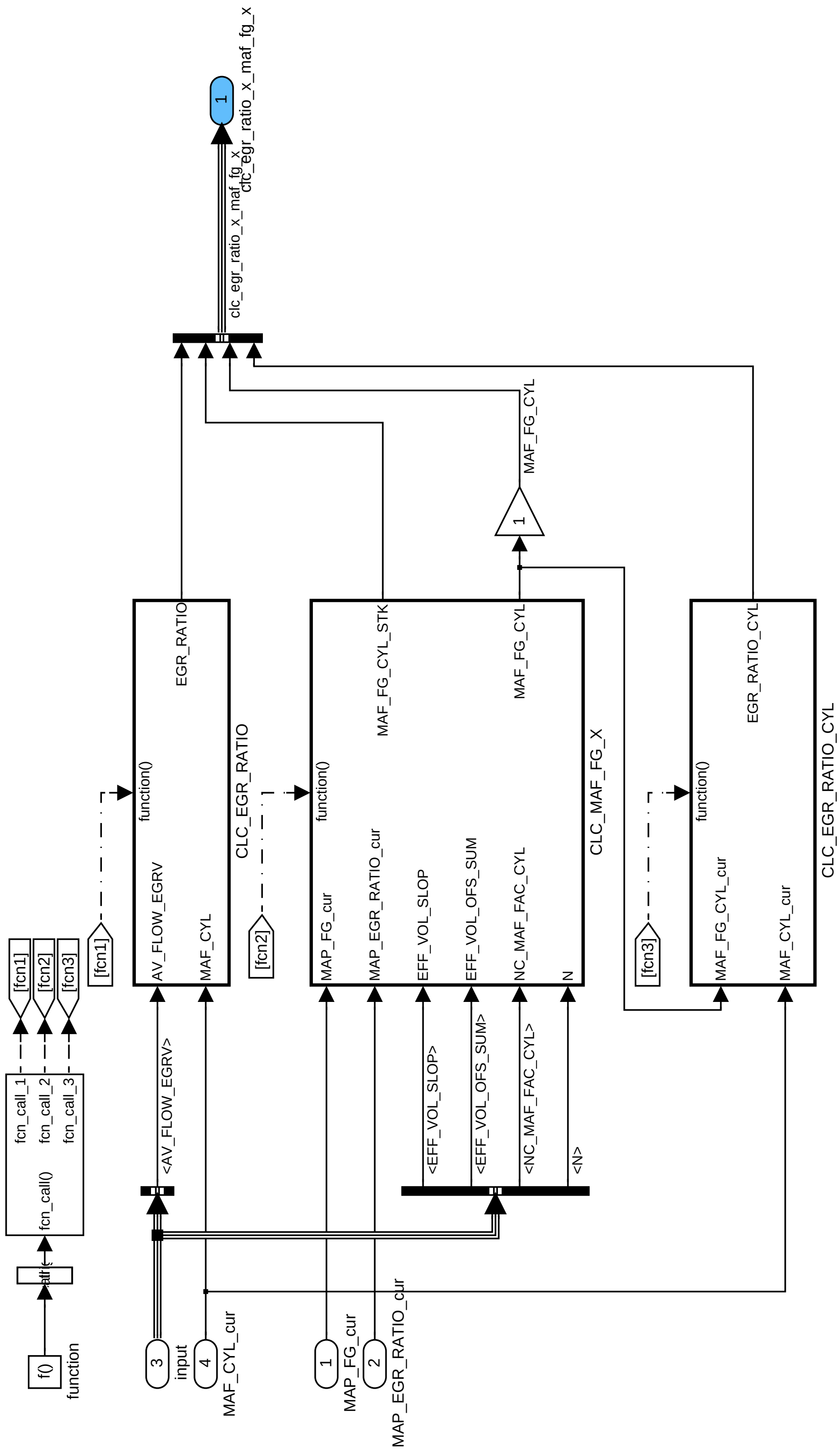


Figure 109:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X/EGR_RATIO_X_MAF_FG_X_then

4.10.4.5.3.1.1 Calculation of EGR_RATIO

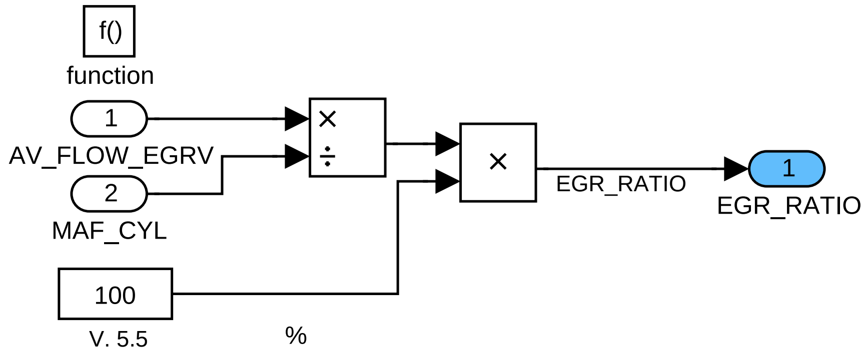


Figure 110:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X/EGR_RATIO_X_MAF_FG_X_then/CLC_EGR_RATIO

4.10.4.5.3.1.2 Calculation of MAF_FG_CYL and MAF_FG_CYL_STK

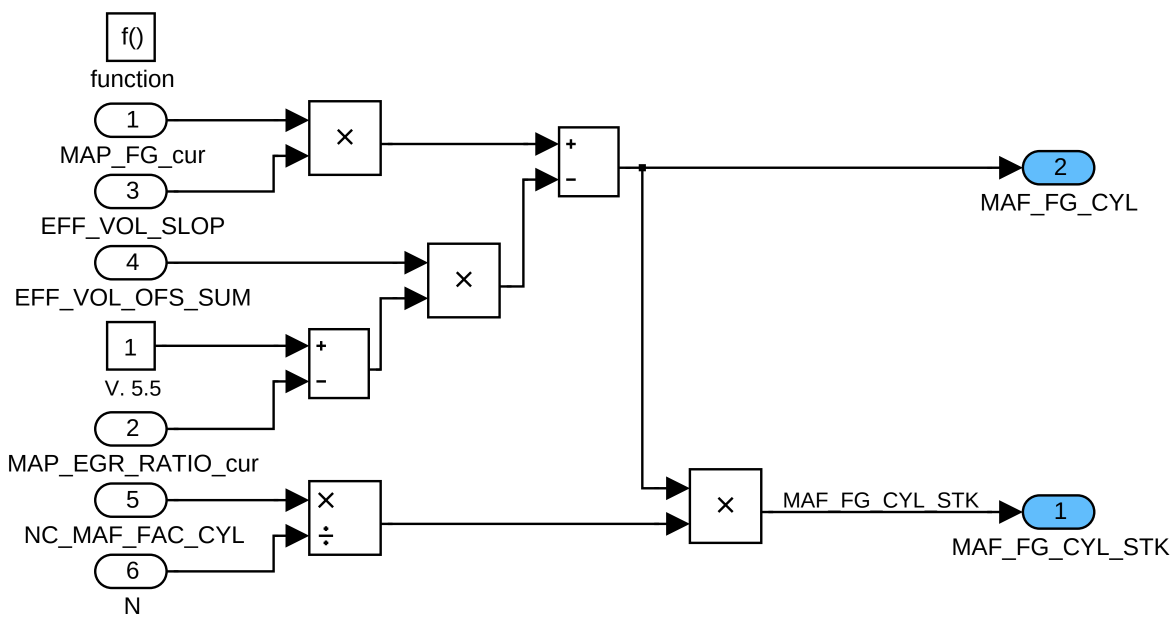


Figure 111:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X/EGR_RATIO_X_MAF_FG_X_then/CLC_MAF_FG_X

4.10.4.5.3.1.3 Calculation of EGR_RATIO_CYL

The ratio of exhaust gas flow and complete gas flow into cylinder EGR_RATIO_CYL can be calculated out of the total air mass flow into cylinder MAF_KGH_ENG_1 and the difference of the total air mass flow and the fresh gas flow into the cylinder:

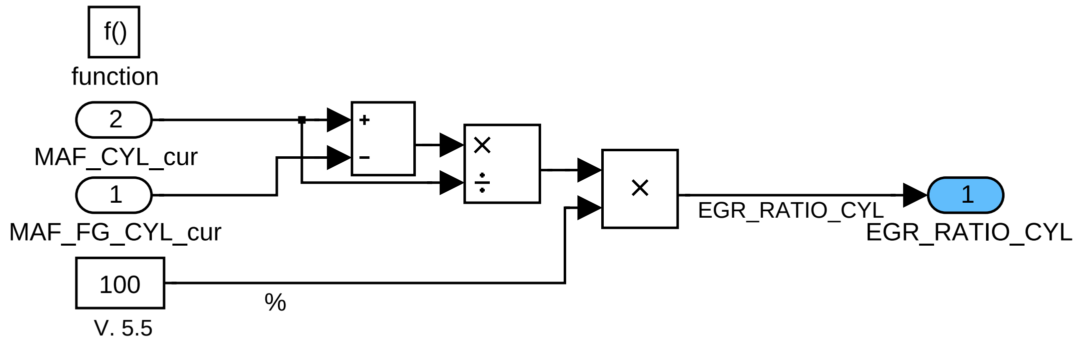


Figure 112:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X/EGR_RATIO_X_MAF_FG_X_then/CLC_EGR_RATIO_CYL

4.10.4.5.3.2 EGR_RATIO_X and MAF_FG_X - else branch

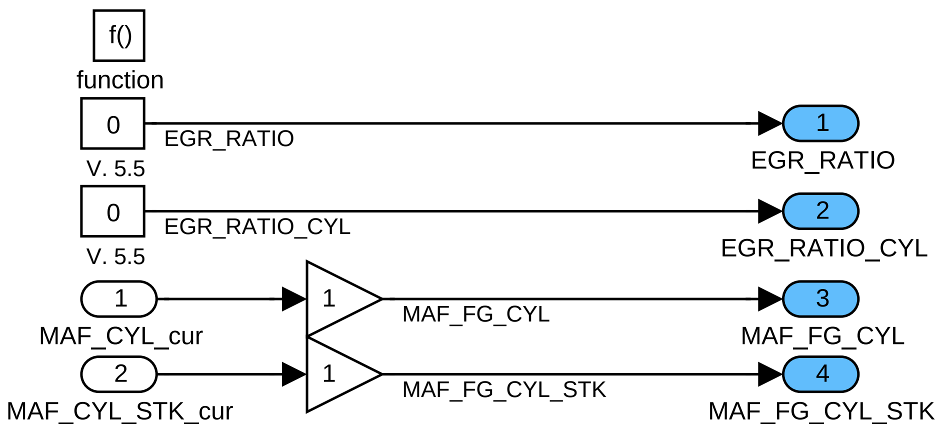


Figure 113:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_THR_CYL_AIR_FLOW/CLC_EGR_RATIO_X_MAF_FG_X/EGR_RATIO_X_MAF_FG_X_else

4.10.4.6 Pressure predictions and the cylinder air mass prediction

The change of pressure in the actual and previous segment is known. So it is possible to predict the manifold air pressure for a various number of segments. The number of segments for the prediction (prediction horizon) is a function of engine speed.

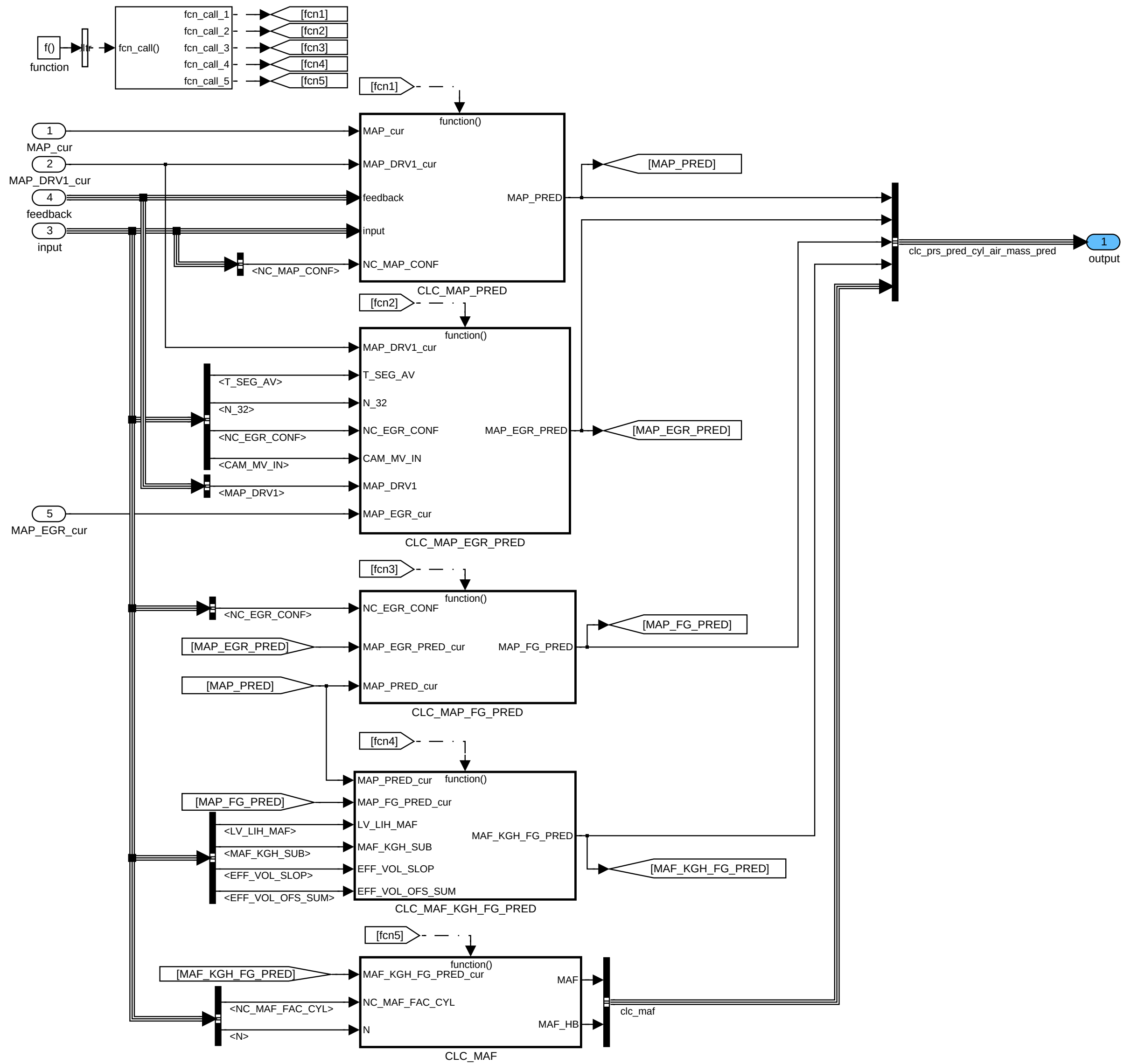


Figure 114:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED

4.10.4.6.1 Calculation of MAP_PRED

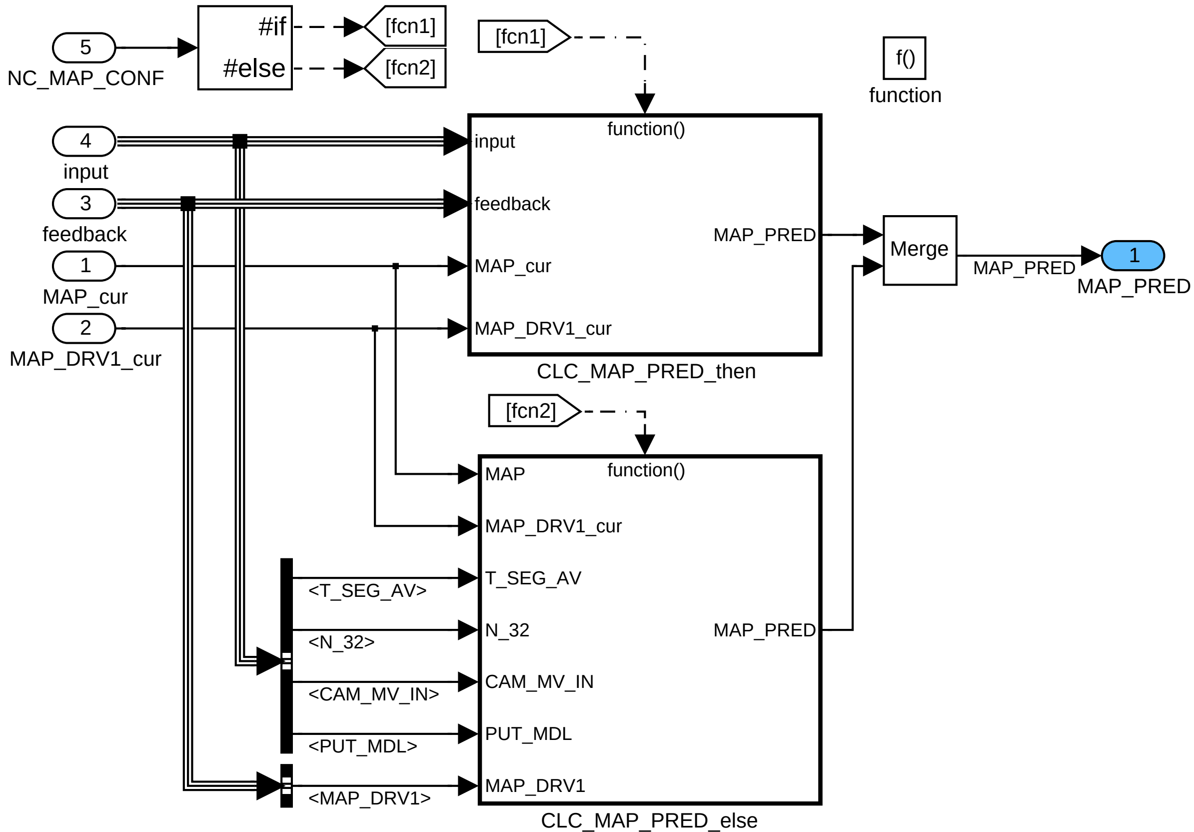


Figure 115:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_PRED

4.10.4.6.1.1 Calculation of MAP_PRED - then branch

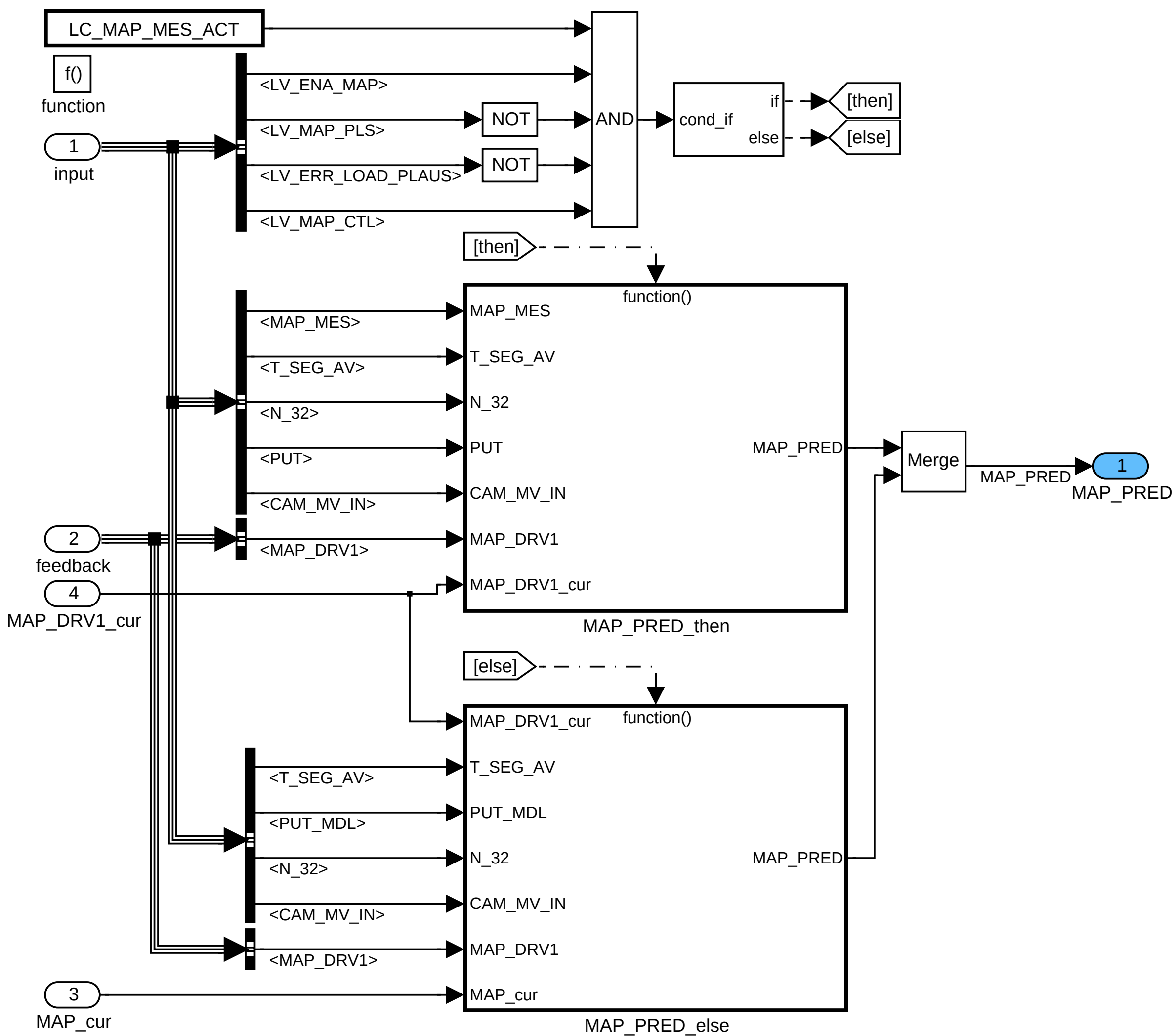


Figure 116:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_PRED/CLC_MAP_PRED_then

4.10.4.6.1.1.1 MAP_PRED - then branch

If the predicted manifold pressure exceeds the pressure upstream to the throttle it is limited to C_FAC_MAP_PRED_TOL * PUT.

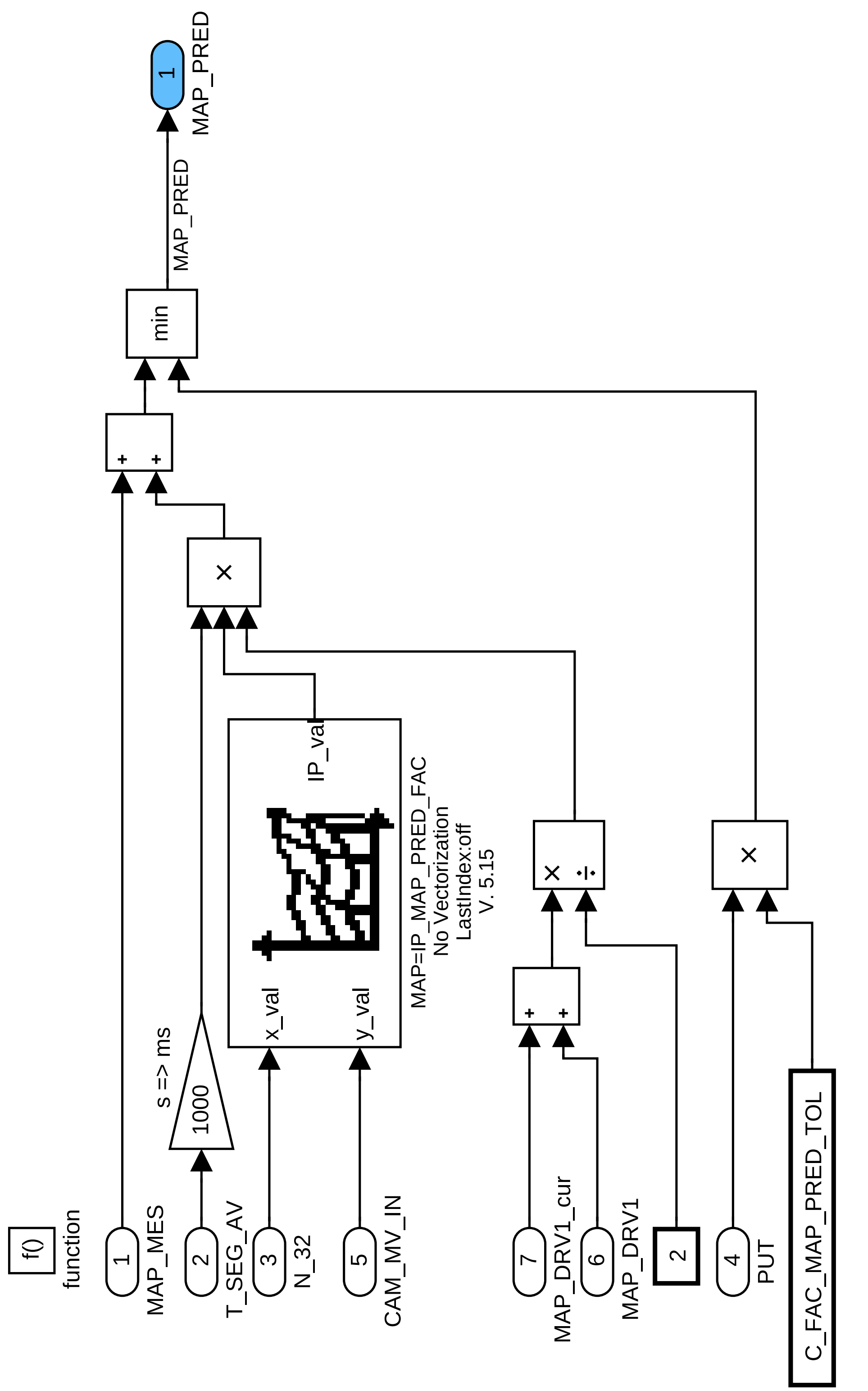


Figure 117:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_PRED/CLC_MAP_PRED_then/MAP_PRED_then

4.10.4.6.1.1.2 MAP_PRED - else branch

If the predicted manifold pressure exceeds the pressure upstream to the throttle (PUT_MDL) it is limited to C_FAC_MAP_PRED_TOL * PUT_MDL.

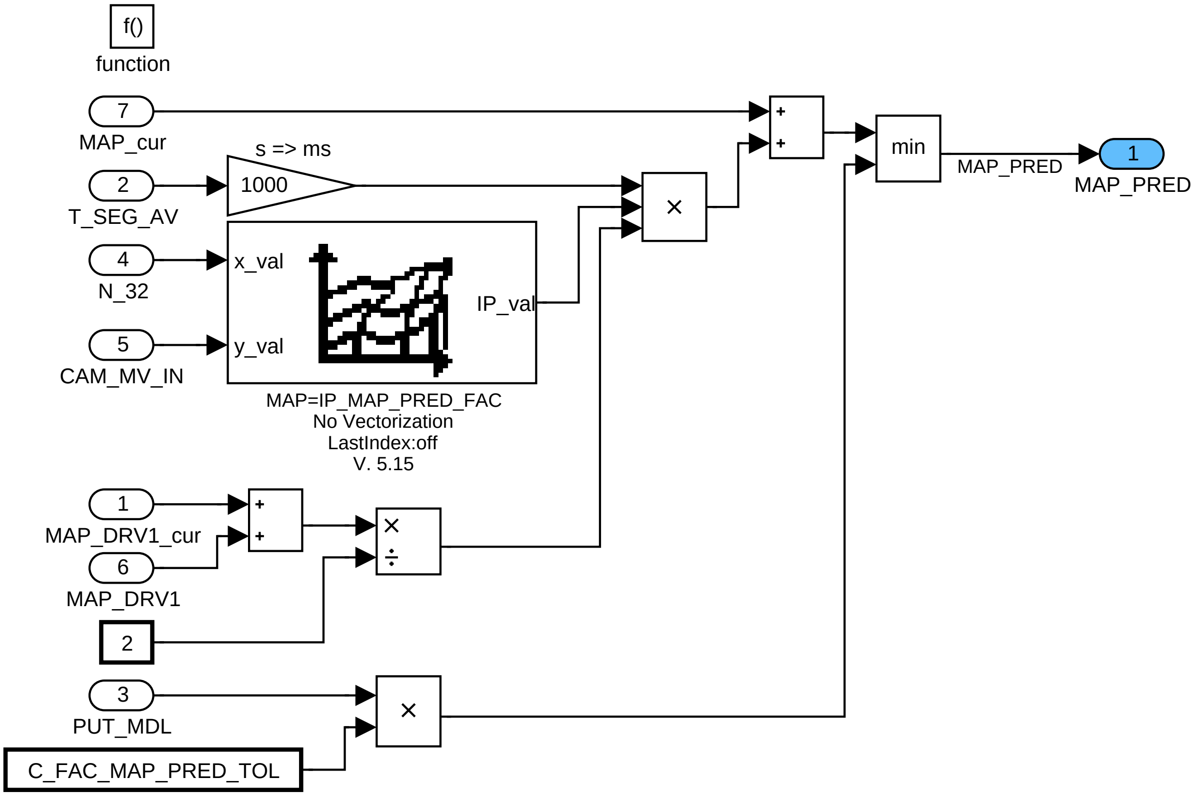


Figure 118:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_PRED/CLC_MAP_PRED_then/MAP_PRED_else

4.10.4.6.1.2 Calculation of MAP_PRED - else branch

If the predicted manifold pressure exceeds the pressure upstream to the throttle (PUT_MDL) it is limited to C_FAC_MAP_PRED_TOL * PUT_MDL.

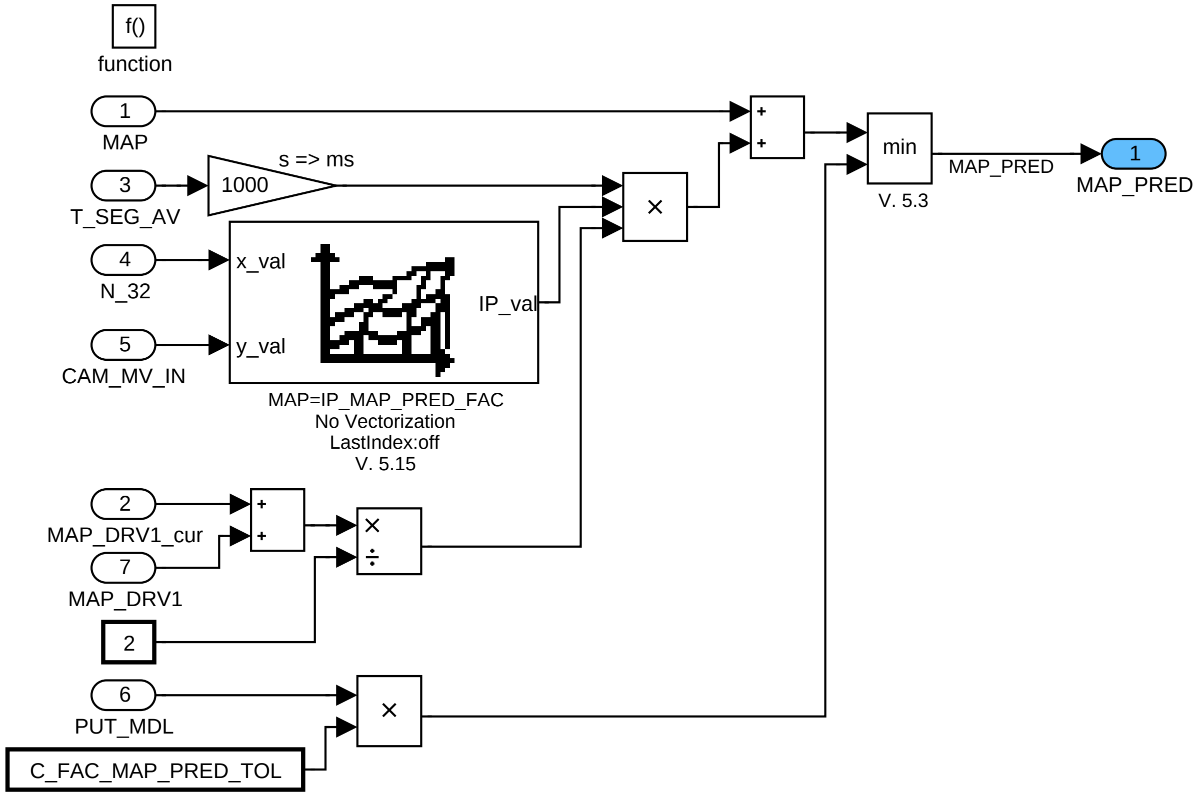


Figure 119:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_PRED/CLC_MAP_PRED_else

4.10.4.6.2 Calculation of MAP_EGR_PRED

The predicted residual gas partial pressure in the manifold is given by:

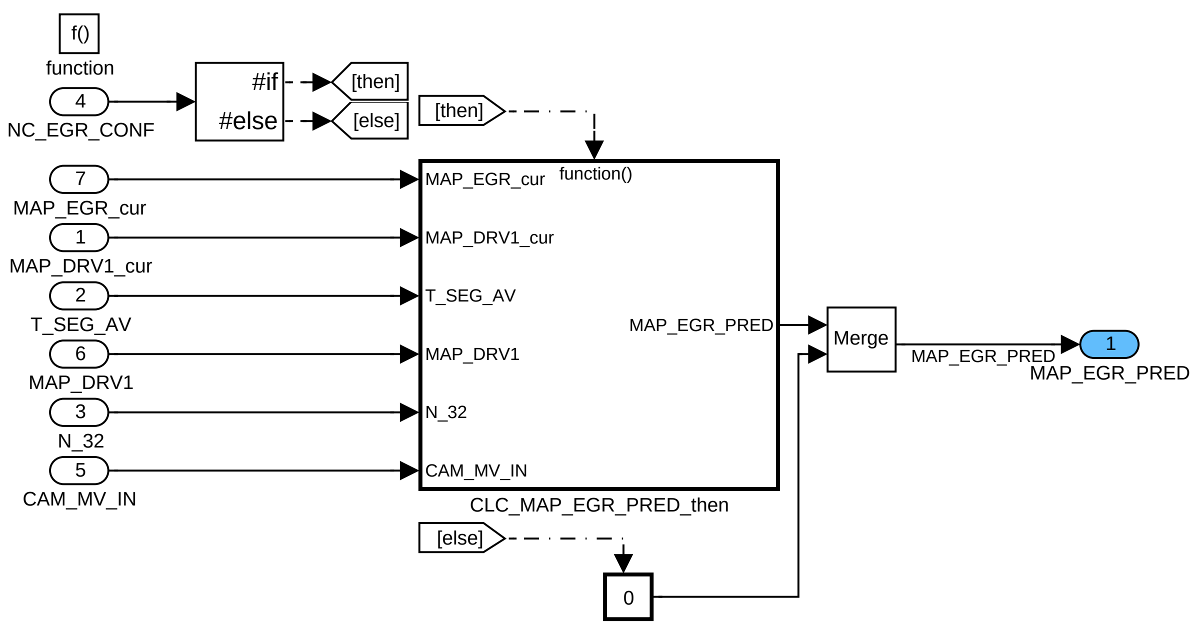


Figure 120:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_EGR_PRED

4.10.4.6.2.1 Calculation of MAP_EGR_PRED - then branch

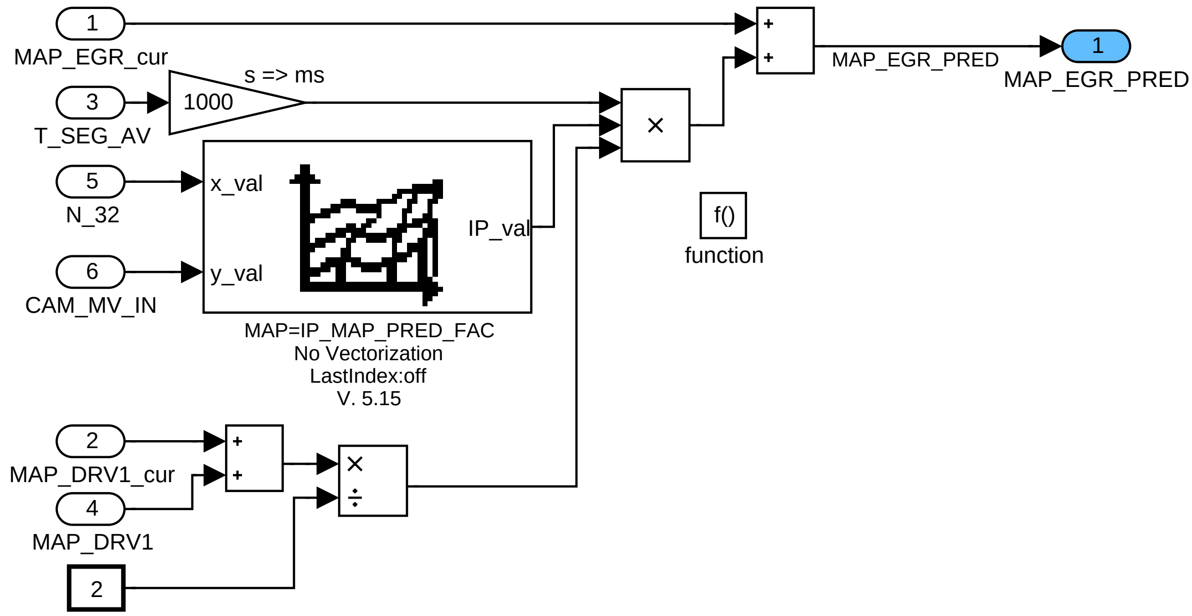


Figure 121:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_EGR_PRED/CLC_MAP_EGR_PRED_then

4.10.4.6.3 Calculation of MAP_FG_PRED

With the predicted manifold pressure and the residual gas pressure it is possible to calculate the fresh air partial pressure and the predicted mass of fresh air into the cylinder. In case of a TPS error non ETC systems use the measured mass air flow as an substitute for the air mass flow model value.
The predicted fresh gas partial pressure in the manifold is given by:

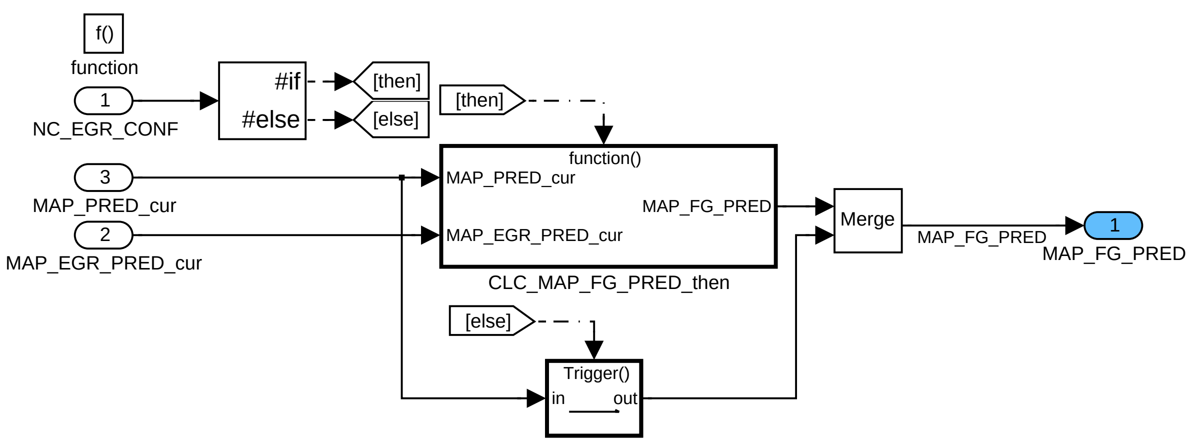


Figure 122:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_FG_PRED

4.10.4.6.3.1 Calculation of MAP_FG_PRED - then branch

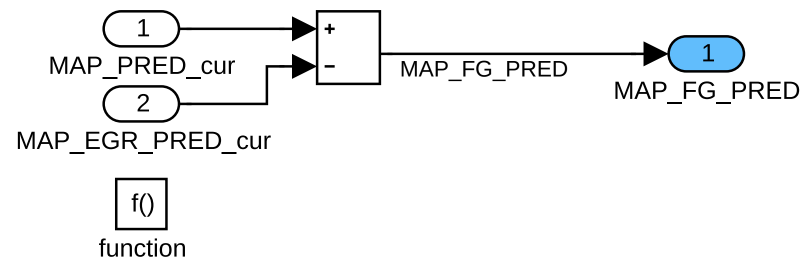


Figure 123:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAP_FG_PRED/CLC_MAP_FG_PRED_then

4.10.4.6.4 Calculation of MAF_KGH_FG_PRED:

It is necessary to consider two different errors (they are defined in the Intake Manifold Model Appl. Inc.) because if one of these errors occurs then we have to calculate the predicted fresh gas out of the manifold in the following way (using a map or using the measured value).

The predicted fresh gas mass flow out of the manifold is given by:

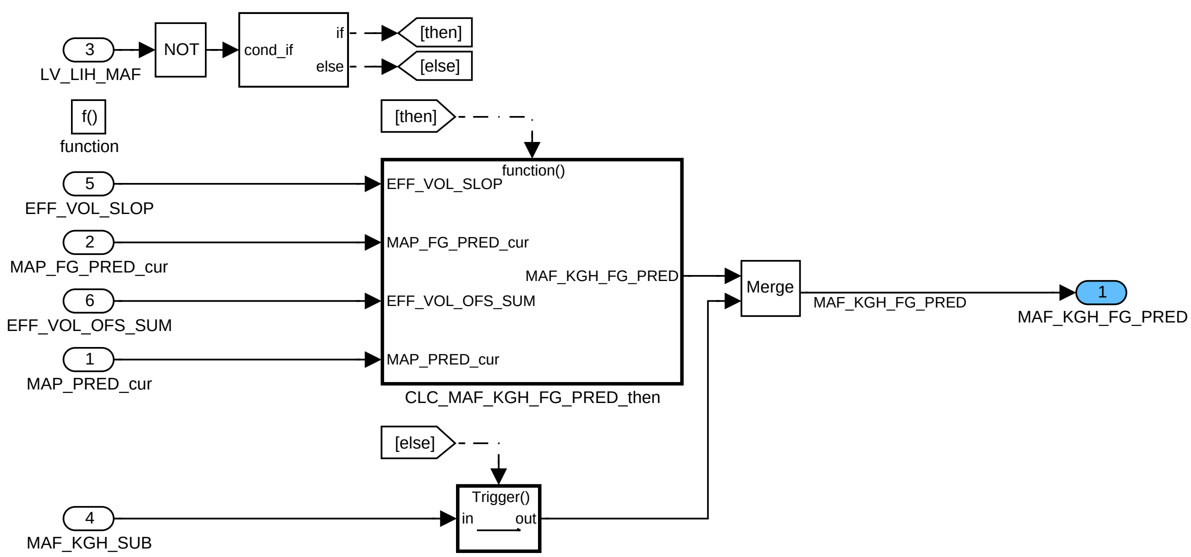


Figure 124:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAF_KGH_FG_PRED

4.10.4.6.4.1 Calculation of MAF_KGH_FG_PRED - then branch:

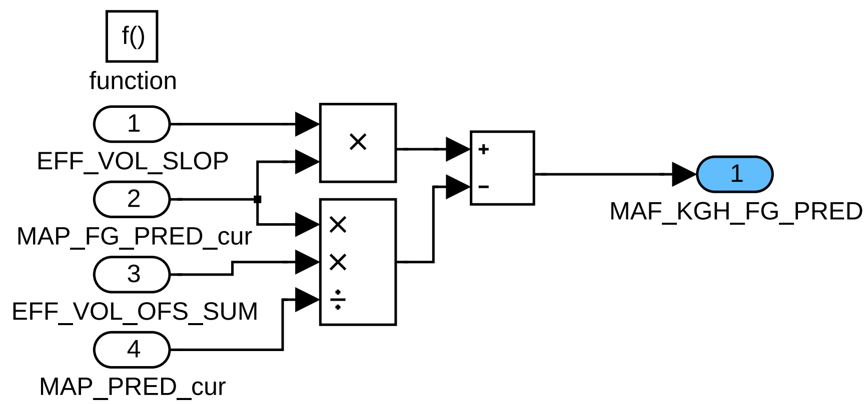


Figure 125:
Path:
INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAF_KGH_FG_PRED/CLC_MAF_KGH_FG_PRED_then

4.10.4.6.5 Calculation of MAF:

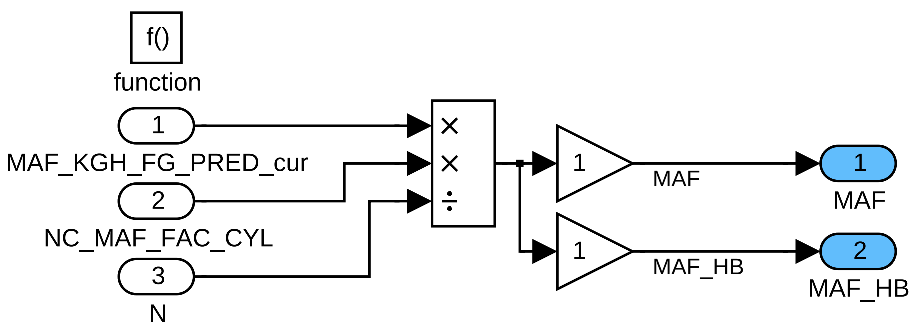


Figure 126:
Path: INSY_DTSYSIMM0/operate__SEG/CLC_PRS_PRED_CYL_AIR_MASS_PRED/CLC_MAF

4.11 Application Incidence Intake Manifold Model (MAF/MAP-System)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FLOW_CRCV	O	0H...FFFFH	0...2047.96875	0.03125	kg/h
Calculated crank case air flow (if NC_CRCV_CONF = 0 this is a dummy variable)					
LV_AMP_AD_RST	O/V	0H ..01H	0...1	1	-
Adaptation of ambient pressure is reseted to "0"					
LV_AMP_AD_STOP	O/V	0H ..01H	0...1	1	-
Adaptation of ambient pressure is stopped					
LV_AMP_SWI	O/V	0H...01H	0..1	1	-
Ambient pressure sensor is switched off (=0) or not (=1)					
LV_AR_RED_AD_RST	O/V	0H ..01H	0...1	1	-
Adaptation of reduced area is reseted to "0"					
LV_AR_RED_AD_STOP	O/V	0H ..01H	0...1	1	-
Adaptation of reduced area is stopped					
LV_AR_RED_CTL_STOP	O/V	0H ..01H	0...1	1	-
Reduced area controller is stopped					
LV_MAF_CTL	O/V	0H ..01H	0...1	1	-
MAF controlled system					
LV_MAF_SWI	O/V	0H...01H	0..1	1	-
MAF sensor is switched off (=0) or not (=1)					
LV_PUT_CTL_STOP	O/V	0H ..01H	0...1	1	-
Pressure controller is stopped					
LV_MAP_CTL	O/V	0H...01H	0...1	1	-
MAP controlled system (always 0 because no MAP sensor available)					
LV_MAP_SWI	O/V	0H...01H	0..1	1	-
Manifold pressure sensor is switched off (=0) or not (=1)					
N_MAF_MES_FAC	O/V	0H...1FE0H	0...8160	1	Rpm
Corrected engine speed for correction of MAF signal					
MAF_KGH_MES_FAC	O/V	0H...FFH	0...1.992	0.0078	-
Correction factor for MAF_KGH_MES_BAS					
FAC_AR_RED_MAF	O/V	0H...FFFFH	0...1.99997	3.05e-5	-
Correction factor for basic reduced area					
SEG_INC	O/V	0H...FFH	0...255	1	-
Segment counter for INSY					
T_SEG_AV_IMM	O/V	0H...FFFFH	0...262.14	0.004	ms
Segment time for INSY					
LV_ERR_LOAD_TPS_PLAUS_FAST	V	0...1	0...1	1	-
Unplausible Load or TPS ratio detected					
LV_TRA_STATE	V	0...0H 1...1H	"A" "D"	-	-
Detection of increasing or decreasing throttle position (Acceleration or Deceleration)					
LV_INH_AR_RED_AD	O/V	0..1H	0..1	1	-
Bit for inhibit the adaptation of the reduced area at the throttle					
LV_INH_AMP_AD	O/V	0..1H	0..1	1	-
Bit for inhibit the adaptation of ambient pressure at wide open throttle					
MAF_KGH_SUB	O/V	0...FFFFH	0...2047.96875	0.03125	kg/h
Substitude air mass for limp home					
OPG_SP_LIM_ISA	V/O	0...FFFFH	0...99.9985	0.0015	%
Idle Speed Actuator opening limitation in case of suspected TPS Ratio failure					
LV_LIH_MAF	V/O	0...1H	0...1	1	-
Limp home flag for error on TPS or/and MAF signals					
CYCNR_OPG_LIM_ISA	-	0...FFH	0...255	1	-
Counter of number of segments to activate ISA opening limitation in case of suspected TPS failure					
FLOW_CPS	V/O	0...FFFF H	0...8	0.000122	kg/h
flow from tank-system into manifold through CPS					
LV_LOAD_TPS_PLAUS_ACT_FAST	V	0...1	0...1	1	-
MAP / TPS Ratio check function active					

CTR_SEG_LIH_MAF	V	0...FFH	0...255	1	-
Counter to set LV_LIH_MAF = 1 with NC_ETC CONF = 1					
LV_LIH_MAF_TMP	V	0...1	0...1	1	-
Temporary flag to adjust STATE_TPS_DIAG (10ms) to LV_LIH_MAF (segment synchronous) update rate					
LV_TPS_GRD_CTL_STOP	V	0...1	0...1	1	-
Flag indicating that Reduced Area / Ambient Pressure controllers must be stopped due to high TPS_GRD					
CYC_TPS_GRD_CTL_STOP	V	0...FFH	0...255	1	-
Number of cycles to stop Reduced Area / Ambient Pressure controllers in case of high TPS_GRD					
MAF_MES_FIL	O/V	0...FFFFH	0...1389	0.0212	mg/stk
Filtered Measured Air-mass flow per segment					
STATE_ACR_EFF_VOL_CLC	O/V	0...FFH	0...255	1	-
Detected state of actuators					
LV_MAP_SAMPLE_DYN	O/V	0...1H	0...1	1	-
Activation bit to speed up MAP acquisition					
AMP_SUB	O/V	0...FFFFH	0...5434	0.083	hPa
AMP substitude value for LC_TCHA_CONF = 1 and LV_ENA_AMP = 0 (charged system)					
LV_TPS_LIH_LOAD_TPS_PLAUS	V/O	0...1H	0...1	1	-
Flag to request TPS limp home due to unplausible Load/TPS signal					
FAC_VIM_TMP	V/O	0...FFH	0...0.99609	3.9063e-3	[-]
Value of FAC_VIM used only for the calculation of Volumetric efficiency					
FAC_PORT_DEAC_MV_TMP	V/O	0...FFFFH	0...0.99998	0.0153e-3	[-]
Value of FAC_PORT_DEAC_MV used only for the calculation of Volumetric efficiency					
LV_PUT_SWI	O/V	0H...01H	0..1	1	-
Switching flag for pressure upstream throttle sensor					
CAM_OFS_IVVT_IN_1	O/V/S	8000...7FFFH	-48...47.9985352	0.0014648 4	°CRK
Offset for inlet camshaft position which is used as an input for the volumetric efficiency calculation					
CAM_OFS_IVVT_EX_1	O/V/S	8000...7FFFH	-48...47.9985352	0.0014648 4	°CRK
Offset for exhaust camshaft position which is used as an input for the volumetric efficiency calculation					
MAF_MIN_TOT_DC	V/O/S	0...FFFFH	0...1.389E+3	0.0211947 8	mg/stk
Former / current driving cycle minimum MAF					
MAF_MAX_TOT_DC	V/O/S	0...FFFFH	0...1.389E+3	0.0211947 8	mg/stk
Former / current driving cycle maximum MAF					
MAP_MIN_TOT_DC	V/O/S	0...FFFFH	0...5434	0.0829175	hPa
Former / current driving cycle minimum MAP					
MAP_MAX_TOT_DC	V/O/S	0...FFFFH	0... 5434	0.0829175	hPa
Former / current driving cycle maximum MAP					
PUT_MES_MAX_TOT_DC	O/V/S	0...FFFFH	0...5.434E+3	0.0829175	hPa
Former / current driving cycle maximum pressure upstream throttle per segment measured					
PUT_MES_MIN_TOT_DC	O/V/S	0...FFFFH	0...5.434E+3	0.0829175	hPa
Former / current driving cycle minimum pressure upstream throttle per segment measured					
LV_MAF_CAL_SWI	V	0...1H	0...1	1	-
MAF sensor for Calibration purpose is active					
MAF_KGH_MES_BAS_CAL	V	0...FFFFH	0...2.04797e +3	0.03125	Kg/h
Raw air mass flow measured for Calibration Purpose (without correction)					
MAF_KGH_MES_CAL	V	0...FFFFH	0...2.04797e +3	0.03125	Kg/h
Corrected mass air flow for Calibration Purpose					
MAF_MES_CAL	V	0...FFFFH	0...1.389e +3	0.0211947 8	mg/stk
Mass air flow for Calibration Purpose					
LV_INH_MAP_CTL	O/V	0H ..01H	0...1	1	-
Flag to indicate that a MAP-sensor-failure is assumed in case of a Load-TPS-error					
LAMB_CTL_SUM_INH_MAP[NC_CBK_EX_NR]	V	8000...7FFFH	-50 ... 49.99847	1.52e-3	%
relative LAM correction (included LAM adaptation and LAM controller)					
LAMB_CTL_SUM_INH_MAP_MMV[NC_CBK_EX_NR]	V	8000...7FFFH	-50 ... 49.99847	1.52e-3	%
Filtered relative LAM correction (included LAM adaptation and LAM controller)					

CAM_OFS_IVVT_IN_MAX_TOT_DC	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Former / current driving cycle maximum offset for inlet camshaft position					
CAM_OFS_IVVT_IN_MIN_TOT_DC	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Former / current driving cycle minimum offset for inlet camshaft position					
CAM_OFS_IVVT_EX_MAX_TOT_DC	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Former / current driving cycle maximum offset for exhaust camshaft position					
CAM_OFS_IVVT_EX_MIN_TOT_DC	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Former / current driving cycle minimum offset for exhaust camshaft position					
CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC	O/V/S	0...CH	0...12	1	-
Former / current driving cycle maximum counter over number of camshaft-positions being used for adaptation					
AR_RED_AD_CAM_AD_MAX_TOT_DC	O/V/S	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Former / current driving cycle maximum result of AR_RED_AD calculation for CAM_OFS_AD functionality purpose					
AR_RED_AD_CAM_AD_MIN_TOT_DC	O/V/S	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Former / current driving cycle minimum result of AR_RED_AD calculation for CAM_OFS_AD functionality purpose					
AR_RED_AD_ADD_DIF_MAX_TOT_DC	O/V/S	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Former / current driving cycle maximum difference of AR_RED_AD_ADD and AR_RED_AD_ADD_MMV					
AR_RED_AD_ADD_DIF_MIN_TOT_DC	O/V/S	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Former / current driving cycle minimum difference of AR_RED_AD_ADD and AR_RED_AD_ADD_MMV					
AR_RED_SUM_REL_MAX_TOT_DC	O/V/S	8000...7FFFH	-100...99.996948	3.0518e-3	%
Former / current driving cycle maximum filtered reduced area controller excitation (without adaptation values)					
AR_RED_SUM_REL_MIN_TOT_DC	O/V/S	8000...7FFFH	-100...99.996948	3.0518e-3	%
Former / current driving cycle minimum filtered reduced area controller excitation (without adaptation values)					
PUT_MDL_DIF_MMV_REL_MAX_TOT_DC	O/V/S	8000...7FFFH	-50 ... 49.99847	1.52e-3	%
Former / current driving cycle maximum filtered pressure controller devided by ampient pressure					
PUT_MDL_DIF_MMV_REL_MIN_TOT_DC	O/V/S	8000...7FFFH	-50 ... 49.99847	1.52e-3	%
Former / current driving cycle minimum filtered pressure controller devided by ampient pressure					
LV_CAM_OFS_AD_END	O/V/S	0H ..01H	0...1	1	-
Flag to indicate that the camshaft offset adaptation is finished					
CAM_OFS_IVVT_IN_1_PREV	V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Offset for inlet camshaft position from previous adaptation					
CAM_OFS_IVVT_EX_1_PREV	V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Offset for exhaust camshaft position from previous adaptation					
CAM_OFS_IVVT_IN_TMP	-	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Temporary offset for inlet camshaft position					
CAM_OFS_IVVT_EX_TMP	-	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Temporary offset for exhaust camshaft position					

Input data:

LV_CAM_SP_ADJ_CAM_OFS_AD	T_AST	TCO_ST	FAC_MAF_TRIM
NC_CRCV_CONF	T_SEG_AV	NC_ETC_CONF	STATE_TPS_DIAG
N_32	TPS_GRD	LV_ERR_TCO_GRD	ERR_SYM_MAF_SCG_OC
LV_ERR_ISA_i	MAF_CPS	LV_ERR_TCO	LV_ERR_TPS_PLAUS
ERR_SYM_MAF_SCP	LV_TPS_LIH_ACT	AR_RED_DIF_REL	TIA
N_SP_IS	TPS_SEG	LV_ERR_TCO_PLAUS	LV_ERR_LOAD_TPS_PLAUS
N	LV_TPS_GRD_UP	T_SEG_HALF_AV	LV_ERR_TCO_STUCK
MAF_MAX_TOT	MAF_MDL_DIF	LV_MAF_PULS	LV_ERR_MEC_OPEN_CPS
MAP	LV_END_AMP_AD	STATE_CLC_RED	STATE_CLC_SEG_INSY
LV_ERR_CPS	ERR_SYM_ISA	AMP	LV_TQ_ADD_ENG_STALL_ACT
LV_ERR_TIA	ERR_SYM_TPS_1	ERR_SYM_TPS_2	FAC_VIM_REQ_CAM_OFS_AD
NC_MAP_CONF	LV_ERR_LOAD_PLAUS	FAC_VIM	C_AMP_INI
VP_MAF	LV_ERR_MAP	LV_ST_END	ERR_SYM_EL_MAP
EFF_VOL_SLOP	MAP_MES	EFF_VOL_OFS_SUM	NC_MAF_FAC_CYL
NC_STATE_CLC_RED_SEG_1	LV_ERR_TPS_DET	NC_STATE_CLC_RED_MASK_1	LV_CDN_VB_OBD2
LV_ERR_MAP_PLAUS_DIAG	LC_TCHA_CONF	LV_EFF_VOL_CLC_CAM_OFS_AD	NLC_IVVT_IN
NLC_IVVT_EX	STATE_CAM_OFS_AD	LV_ES	LV_ST
MAF_CYL_STK	MAF_MES	CONF_MAF	LV_DET_FIRST_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]
CAM_OFS_IVVT_IN[NC_NR_OPP_CAM_OFS_AD]	CAM_OFS_IVVT_EX[NC_NR_OPP_CAM_OFS_AD]	PUT_MES	FAC_PORT_DEAC
VP_MAF_CAL	MAF	ID_MAF_TAB	LV_IGK
LV_ERR_FSD_LAM_LIM_i	LV_ERR_TPS	MAP_MES_BAS	FAC_LAM_LIM[NC_CBK_EX_NR]
FAC_LAM_AD_OUT[NC_CBK_EX_NR]	FAC_MFF_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	CTR_SP_REQ_CAM_SP_ADJ	AR_RED_AD_ADD_REQ_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]
STATE_AR_RED_AD_ADD	CTR_AR_RED_AD_ADD_FAST	AR_RED_AD_ADD	AR_RED_AD_ADD_MMV
AR_RED_SUM_COR_ACT_REL	PUT_MDL_DIF_MMV_REL		

FUNCTION DESCRIPTION:

General information:

The configuration of the system is done with the non-calibratable constants. Those constants are also used as compiler switches in some cases.

Application conditions:

Initialization: The initialization at reset is split up:

- part 1: only "Setting of sensor configuration" (chapter 1.1.1)
- part 2: all other "initialization at reset" - tasks

Activation:	at every engine state
Deactivation:	-
Calculation recurrence:	10 ms
Segment synchronous	1.1.3 TPS error handling. To adjust STATE_TPS_DIAG (10 ms) to LV_LIH_MAF (segment synchronous) just "If-Then-Else" loop No.:(2) (Definition of LV_LIH_MAF) is calculated every 10 ms. 1.1.5 ISA opening limitation in case of suspected TPS Ratio failure LV_TPS_GRD_CTL_STOP calculation

4.11.1 Adjustment of Intake System

4.11.1.1 System load sensor (MAF/MAP-System)

Via CONF_MAF it is possible to switch between MAF- and MAP controlled System. The MAF, MAP or AMP sensor can switched off via the variables LV_MAF_SWI, LV_MAP_SWI and LV_AMP_SWI.

Additionally the MAF signal can be calculated via second Input (alternative to PUT-Sensor Input for Non-Turbo versions). This is determined only for Calibration Purpose.

Initialization: At Reset – Setting of Sensor configuration (Initialization at reset – part 1)

If CONF_MAF = 1

Then LV_MAP_SWI = 0

LV_MAF_SWI = 1

Else LV_MAP_SWI = 1

LV_MAF_SWI = 0

Endif

If LC_TCHA_CONF = 1

Then LV_AMP_SWI = 1

LV_PUT_SWI = 1

LV_MAF_CAL_SWI = 0

Else LV_AMP_SWI = 0

LV_PUT_SWI = 0

LV_MAF_CAL_SWI = 1

Endif

The correction value MAF_KGH_MES_FAC is included to correct the MAF_KGH_MES (measured air-mass flow) in case of strong pulsation. As the engine speed where pulsation occurs is depending on air temperature (air velocity change), a corrected engine speed is used to interpolate IP_MAF_KGH_MES_FAC. Offset engine speed is typically positive for high air temperature and negative for low air temperature.

Furthermore a Trimming Factor is applied to compensate EMS MAF Signal Input Circuit Tolerance. The Trimming Factor is determined at the EMS Production Line and then stored in the non-volatile Memory. The Trimming Factor is corrected by an additional Compensation Factor depending on VP_MAF to compensate the non-linear MAF Linearization Characteristic.

Initialization at reset: MAF_KGH_MES_FAC = 1

Segment synchronuos update:

N_MAF_MES_FAC = N + IP_N_MAF_MES_FAC_OFS (TIA)

<u>Remark</u> : if the result of N + IP_N_MAF_MES_FAC_OFS (TIA) is below 0 or above 8160 (min. / max. limits of engine speed), then N_MAF_MES_FAC is set to 0 or 8160 accordingly.

MAF_KGH_MES_FAC = IP_MAF_KGH_MES_FAC (N_MAF_MES_FAC, TPS_SEG)
* (1 + IP_FAC_MAF_TRIM_COR * (1 - FAC_MAF_TRIM))

4.11.1.2 Closed Crank case ventilation

No <u>additional</u> CRCV flow is considered (NC_CRCV_CONF = 0)

Initialization at reset: FLOW_CRCV = 0

4.11.1.3 TPS error handling

In Case of a TPS error there is a special handling of the MAF signal in the Intake Manifold Model for non-ETC systems.

In case of a non-ETC system a double error has to be detected, an error present on the TPS acquisition and an error present on the MAF sensor (both are coming from OBDI diagnosis).

If an ETC-System is used, some related variables must be defined as dummy to satisfy the corresponding interfaces.

Initialization at reset: LV_LIH_MAF = 0
MAF_KGH_SUB = 0
MAF_MES_FIL = 0

For ETC-System (NC_ETC_CONF = 1):
Initialization at reset : LV_LIH_MAF_TMP = 0
CTR_SEG_LIH_MAF = 0
LV_TPS_LIH_LOAD_TPS_PLAUS = 0 *(never changed)*
LV_ERR_LOAD_TPS_PLAUS_FAST = 0 *(never changed)*
LV_LOAD_TPS_PLAUS_ACT_FAST = 0 *(never changed)*

For non ETC-System (NC_ETC_CONF = 0):
Initialization at reset: LV_LOAD_TPS_PLAUS_ACT_FAST = 0
LV_ERR_LOAD_TPS_PLAUS_FAST = 0
LV_TPS_LIH_LOAD_TPS_PLAUS = 0
LV_LIH_MAF_TMP = 0 *(never changed)*
CTR_SEG_LIH_MAF = 0 *(never changed)*

- *Definition of MAF_MES_FIL :*

If (LV_ST_END = 1)

Then $MAF\_MES\_FIL_{N} = MAF\_MES\_FIL_{N-1}$

$+ C\_CRLC\_MAF\_MES\_FIL * (MAF\_MES\_{}_{N} – MAF\_MES\_FIL_{N-1})$

Else $MAF\_MES\_FIL_{N} = C\_MAF\_MES\_FIL\_ST$

Endif

- *Detection of “Acceleration” or “Deceleration” state :*

Initialization : LV_TRA_STATE = “D” at a reset of ECU or at the transition LV_IGK = 0->1

#If(1) NC_ETC_CONF = 0

#Then(1) **If(2)** TPS_GRD > C_TPS_GRD_TRA
And LV_TPS_GRD_UP = 1
And LV_TRA_STATE = “D”
Then(2) LV_TRA_STATE = “A”
Endif(2)
If(3) TPS_GRD > C_TPS_GRD_TRA
And LV_TPS_GRD_UP = 0
And LV_TRA_STATE = “A”
Then(3) LV_TRA_STATE = “D”
Endif(3)

#Endif(1)

- *Diagnosis activation conditions :*

#If(1) NC_ETC_CONF = 0

#Then(1) **If(2)** LV_CDN_VB_OBD2 = 1
And LV_ERR_CPS = 0
And LV_ERR_MEC_OPEN_CPS = 0
And LV_ERR_TPS_DET = 0
And ERR_SYM_EL_MAP = 0
And ERR_SYM_MAF_SCG_OC = 0
And ERR_SYM_MAF_SCP = 0
And LV_ERR_ISA_i = 0
And LV_ERR_TIA = 0
And LV_ERR_TCO = 0

```
                And   LV_ERR_TCO_STUCK = 0
                And   LV_ERR_TCO_GRD = 0
                And   LV_ERR_TCO_PLAUS = 0
                And   N_32 > C_N_MIN_RATIO_CHK
        Then(2)       LV_LOAD_TPS_PLAUS_ACT_FAST = 1
        Else(2)       LV_LOAD_TPS_PLAUS_ACT_FAST = 0
        Endif(2)
#Endif(1)
```

- *Definition of LV_ERR_LOAD_TPS_PLAUS_FAST :*

```
#If(1)  NC_ETC_CONF = 0
#Then(1)
    If(2)   LV_LOAD_TPS_PLAUS_ACT_FAST = 1
    Then(2)
        If(3)   ( {LV_TRA_STATE = "A" Or TPS_SEG > IP_TPS_SEG_LOAD_MAX}
                And
                ({LV_MAP_SWI = 1
                and   MAP > C_FAC_RATIO_MAX * AMP
                and   MAP_MES < C_FAC_RATIO_MAX_DIAG * AMP}
                or
                {LV_MAF_SWI = 1
                and   MAF_CYL_STK > C_FAC_RATIO_MAX * MAF_MAX_TOT
                and   MAF_MES < C_FAC_RATIO_MAX_DIAG * MAF_MAX_TOT}) )
                Or
                ( LV_TRA_STATE = "D"
                And
                ({LV_MAP_SWI = 1
                and   MAP < C_FAC_RATIO_MIN * AMP
                and   MAP_MES > C_FAC_RATIO_MIN_DIAG * AMP}
                or
                {LV_MAF_SWI = 1
                and   MAF_CYL_STK < C_FAC_RATIO_MIN * MAF_MAX_TOT
                and   MAF_MES > C_FAC_RATIO_MIN_DIAG * MAF_MAX_TOT}) )
        Then(3)       LV_ERR_LOAD_TPS_PLAUS_FAST = 1
        Else(3)       LV_ERR_LOAD_TPS_PLAUS_FAST = 0
```

```
            Endif(3)
      Else(2)      LV_ERR_LOAD_TPS_PLAUS_FAST = 0
      Endif(2)
#Endif(1)
```

- *Definition of LV_LIH_MAF:*

```
#If(1)      NC_ETC_CONF = 1
#Then(1) (ETC-System)
      If(2)   LV_ES = 0 And LV_ST = 0       Note :Recurrence of
                                            loop (2) and (3) is 10ms.
      Then(2)
            If(3)       STATE_TPS_DIAG = TPS_MAX_ACT
                        Or LV_TQ_ADD_ENG_STALL_ACT = 1
                        Or ( ERR_SYM_TPS_1 <> "NO_SYM"
                           And  ERR_SYM_TPS_2 <> "NO_SYM" )
            Then(3)     LV_LIH_MAF_TMP = 1
            Endif(3)
      Endif(2)
      If(4)       LV_LIH_MAF_TMP = 1
      Then(4)     LV_LIH_MAF = 1
                  LV_LIH_MAF_TMP = 0
                  CTR_SEG_LIH_MAF = C_CTR_SEG_LIH_MAF
      Else(4)     IF(5)       CTR_SEG_LIH_MAF = 0
                  Then(5)     LV_LIH_MAF = 0
                  Else(5)     CTR_SEG_LIH_MAF = CTR_SEG_LIH_MAF - 1
                  Endif(5)
      Endif(4)

#Else(1) (non ETC-System)
            If(6)       LV_ERR_TPS_PLAUS = 1
                        Or LV_ERR_LOAD_TPS_PLAUS_FAST = 1
            Then(6)     LV_TPS_LIH_LOAD_TPS_PLAUS = 1   (Request TPS limp home)
            Else(6)     LV_TPS_LIH_LOAD_TPS_PLAUS = 0
            IF(7)       LV_TPS_LIH_LOAD_TPS_PLAUS = 1
                        Or LV_TPS_LIH_ACT = 1
                        Or ERR_SYM_ISA ≠ 0
```

Then(7) LV_LIH_MAF =1

Else(7) LV_LIH_MAF = 0

Endif(7)

#Endif(1)

- *Definition of MAF_KGH_SUB:*

If(1) LV_MAF_SWI = 1

Then(1) **If(2)** ERR_SYM_MAF_SCG_OC ≠ 0

Or ERR_SYM_MAF_SCP ≠ 0

Then(2) MAF_KGH_SUB = IP_MAF_TPS_DIAG

Else(2)

MAF_KGH_SUB = MAF_MES_FIL * N / NC_MAF_FAC_CYL

Endif(2)

Endif(1)

If(3) LV_MAP_SWI = 1

Then(3) **If(4)** ERR_SYM_EL_MAP ≠ 0 **Or** LV_INH_MAP_CTL = 1

Then(4) MAF_KGH_SUB = IP_MAF_TPS_DIAG

Else(4) MAF_KGH_SUB = EFF_VOL_SLOP * MAP_MES - EFF_VOL_OFS_SUM

Endif(4)

Endif(3)

4.11.1.4 Correction of the basic reduced throttle area

FAC_AR_RED_MAF is a correction factor for the calculation of AR_RED_BAS(forward path) and AR_RED_SP_MMV(inverse path).

Initialization at reset: FAC_AR_RED_MAF = 1

Formula section:

FAC_AR_RED_MAF = IP_FAC_AR_RED_THR_COR (N_32, TPS_SEG)

4.11.1.5 ISA opening limitation in case of suspected TPS Ratio failure : OPG_SP_LIM_ISA

In order to avoid unexpected high Idle Speed Actuator opening in case of too high TPS signal, ISA opening is limited via OPG_SP_LIM_ISA if TPS input is suspected to be unplausible : that means high negative reduced area controller or high negative MAP_MDL_DIF.

***Initialization* :**

#If NC_ETC_CONF = 1

Then OPG_SP_LIM_ISA =0 (never changed)

#Else OPG_SP_LIM_ISA = 99.9985%

#Endif

Formula section:

#If NC_ETC_CONF = 0

#Then

If(1) (LV_MAP_SWI =1

And MAP_MES_BAS - MAP < IP_MAP_MDL_DIF_OPG_LIM_ISA)

Or

(LV_MAF_SWI =1

And MAF_MDL_DIF < IP_MAF_MDL_DIF_OPG_LIM_ISA)

Then(1) CYCNR_OPG_LIM_ISA = CYCNR_OPG_LIM_ISA + 1

Else(1) CYCNR_OPG_LIM_ISA = 0

Endif(1)

If(2) AR_RED_DIF_REL < C_AR_RED_DIF_REL_OPG_LIM_ISA

Or (CYCNR_OPG_LIM_ISA >= C_CYCNR_OPG_LIM_ISA)

Then(2)

If(3) N > N_SP_IS

Then(3) OPG_SP_LIM_ISA = IP_OPG_SP_LIM_ISA (TPS_AV, TCO)

Else(3) OPG_SP_LIM_ISA = 99,9985%

Endif(3)

Else(2) OPG_SP_LIM_ISA = 99,9985%

Endif(2)

#Endif

4.11.2 Check for MAP-Sensor-failure in case of a Load-TPS-plaus error

For MAP-Systems it is assumed that the Load-sensor is faulty if a Load-TPS-error is detected and Lambda is out of range.

If a MAP-Failure is detected (LV_INH_MAP_CTL = 1) the IMM-controller, reduced area adaptation and AMP- adaptation is stopped and reset for this DC. Also LV_MAP_CTL is set to 0 to not use MAP_MES for MAF-calculation.

Initialisation at reset: LV_INH_MAP_CTL = 0

LAMB_CTL_SUM_INH_MAP_i = 0

LAMB_CTL_SUM_INH_MAP_MMV_i = 0

Recurrence: 10 ms

Formula section:

LAMB_CTL_SUM_INH_MAP_i = FAC_LAM_LIM_i+ FAC_LAM_AD_OUT_i+
FAC_MFF_ADD_LAM_AD_OUT_i

LAMB_CTL_SUM_INH_MAP_MMV_i = (LAMB_CTL_SUM_INH_MAP_MMV_i(N-1)
* (1 – C_CRLC_LAMB_CTL_SUM_INH_MAP))
+ (LAMB_CTL_SUM_INH_MAP_i *
C_CRLC_LAMB_CTL_SUM_INH_MAP)

If LV_MAP_SWI = 1
And LV_ERR_LOAD_TPS_PLAUS = 1
And (LAMB_CTL_SUM_INH_MAP_MMV_i > C_LAMB_CTL_INH_MAP_MAX
Or LAMB_CTL_SUM_INH_MAP_MMV_i < C_LAMB_CTL_INH_MAP_MIN**)**
Then LV_INH_MAP_CTL = 1
Endif

4.11.3 Check for IMM closed loop condition

If LV_MAF_MAP_REQ = 0 and no MAP-sensor error is detected, the adjustment of the Model is done with the MAP-sensor. If LV_MAF_MAP_REQ = 1 and no pulsation takes place and no mass air flow sensor error is detected, the adjustment of the model is done with the mass air flow meter. If the sensor is faulty, the model is working as a throttle / engine-speed system without adjustment (open loop).

To stop or to reset to “0” the adaptation or the controller values of the ambient pressure or the reduced area a cross lock matrix is used. This matrix describes all errors and their implications.

Initialisation at reset: LV_MAF_CTL = 0
LV_MAP_CTL = 0

Formula section:

Recurrence: 10 ms

Coordination between MAF- and MAP-controlled:

If(1) CONF_MAF = 1
Then(1) LV_MAP_CTL = 0
If(2) ERR_SYM_MAF_SCG_OC = 0
And ERR_SYM_MAF_SCP = 0
And LV_MAF_PULS = 0
Then(2) LV_MAF_CTL = 1 *(MAF-control)*
Else(2) LV_MAF_CTL = 0 *(α-/N-control)*
Endif(2)
Else(1) LV_MAF_CTL = 0
If(3) LV_ERR_MAP = 0

And ERR_SYM_EL_MAP = 0

And LV_ERR_LOAD_PLAUS = 0

And LV_ERR_MAP_PLAUS_DIAG = 0

And LV_INH_MAP_CTL = 0

Then(3) LV_MAP_CTL = 1 *(MAP-control)*

Else(3) LV_MAP_CTL = 0 *(α-/N-control)*

Endif(3)

Endif(1)

4.11.4 Check for Volumetric Efficiency Calculation

Explanation: FAC_VIM_TMP is always calculated segment synchronous (with FAC_VIM_TMP = FAC_VIM). If a calculation for the camshaft offset adaptation is required it is possible that FAC_VIM_TMP is calculated additionally with a recurrence of 20 ms (with FAC_VIM_TMP = (FAC_VIM or FAC_VIM_REQ_CAM_OFS_AD)).

4.11.4.1 Calculation of variables for Camshaft Offset Adaptation – Seg. Syn. & 20ms

Application conditions:

Activation: **If** multi segment synchronous

Then recurrence multiples of a segment

Endif

#If [(NLC_IVVT_IN = 1) **or** (NLC_IVVT_EX = 1)]

#Then

If (LV_CAM_SP_ADJ_CAM_OFS_AD = 1 **and**
LV_EFF_VOL_CLC_CAM_OFS_AD = 0 **and**
STATE_CAM_OFS_AD = 3)

Then recurrence every 20 ms

Endif

#Endif

Deactivation: -

Initialization: -

Recurrence: The calculation is done synchronously to **multiples of a segment or in 20ms recurrency**

Formula section:

FAC_VIM_TMP = FAC_VIM

FAC_PORT_DEAC_MV_TMP = FAC_PORT_DEAC

4.11.4.2 Calculation of variables for Camshaft Offset Adaptation – 20ms

Application conditions:

Activation: **#If** [(NLC_IVVT_IN = 1) **or** (NLC_IVVT_EX = 1) **and** (LV_EFF_VOL_CLC_CAM_OFS_AD = 1)]

#Then

every 20 ms

#Endif

Formula section:

FAC_VIM_TMP = FAC_VIM_REQ_CAM_OFS_AD

4.11.4.3 Calculation of variables for Camshaft Offset Adaptation – interface to basic volumetric efficiency

The Camshaft-offset adaptation functionality is used to learn the inlet and exhaust camshaft error and other errors leading to a wrong calculation of the real air-mass flow into the cylinder. The result of this adaptation(CAM_OFS_IVVT_IN and CAM_OFS_IVVT_EX) is an offset to the inlet and exhaust position which is used as an input for the basic volumetric efficiency calculation.

Application conditions:

Activation: LC_EFF_VOL_INP_CAM_OFS_INH = 0

And STATE_CAM_OFS_AD = 5 -> 0

And LV_DET_VLD_RES_CAM_OFS_AD[1] = 1

And LV_CAM_OFS_AD_END = 0

Deactivation: -

Initialization: - at first ECU power up and non-volantile memory lost:

CAM_OFS_IVVT_IN(EX)_1 = 0

LV_CAM_OFS_AD_END = 0

CAM_OFS_IVVT_IN(EX)_1_PREV = 0

- otherwise: restored from non-volantile memory

at reset: CAM_OFS_IVVT_IN(EX)_TMP = 0

Recurrence: 20ms

Formula section:

If |CAM_OFS_IVVT_IN[1]| > C_CAM_OFS_MIN

or |CAM_OFS_IVVT_EX[1]| > C_CAM_OFS_MIN

Then CAM_OFS_IVVT_IN_TMP = CAM_OFS_IVVT_IN[1]

CAM_OFS_IVVT_EX_TMP = CAM_OFS_IVVT_EX[1]

```
        If      CAM_OFS_IVVT_IN[1] > C_CAM_OFS_MAX
        Then    CAM_OFS_IVVT_IN_TMP = C_CAM_OFS_MAX
        Endif
        If      CAM_OFS_IVVT_IN[1] < -C_CAM_OFS_MAX
        Then    CAM_OFS_IVVT_IN_TMP = -C_CAM_OFS_MAX
        Endif
        If      CAM_OFS_IVVT_EX[1] > C_CAM_OFS_MAX
        Then    CAM_OFS_IVVT_EX_TMP = C_CAM_OFS_MAX
        Endif
        If      CAM_OFS_IVVT_EX[1] < -C_CAM_OFS_MAX
        Then    CAM_OFS_IVVT_EX_TMP = -C_CAM_OFS_MAX
        Endif
        If      CAM_OFS_IVVT_IN_1_PREV = 0
                And CAM_OFS_IVVT_EX_1_PREV = 0     (No previous adaptation)
        Then    CAM_OFS_IVVT_IN_1_PREV = CAM_OFS_IVVT_IN_TMP
                CAM_OFS_IVVT_EX_1_PREV = CAM_OFS_IVVT_EX_TMP
        Else    CAM_OFS_IVVT_IN_1 = (CAM_OFS_IVVT_IN_1_PREV +
                                              CAM_OFS_IVVT_IN_TMP) / 2
                CAM_OFS_IVVT_EX_1 = (CAM_OFS_IVVT_EX_1_PREV +
                                              CAM_OFS_IVVT_EX_TMP) / 2
                LV_CAM_OFS_AD_END = 1
        Endif
Else    CAM_OFS_IVVT_IN_1 = 0
        CAM_OFS_IVVT_EX_1 = 0
        LV_CAM_OFS_AD_END = 1
Endif
```

4.11.4.4 Selection of STATE_ACR_EFF_VOL_CLC for the Basic Volumetric Efficiency Calculation

Application conditions:

Activation: **if** multi segment synchronous

then recurrence multiples of a segment

endif

#if [(NLC_IVVT_IN = 1) **or** (NLC_IVVT_EX = 1)]

#then

if (LV_EFF_VOL_CLC_CAM_OFS_AD = 1)

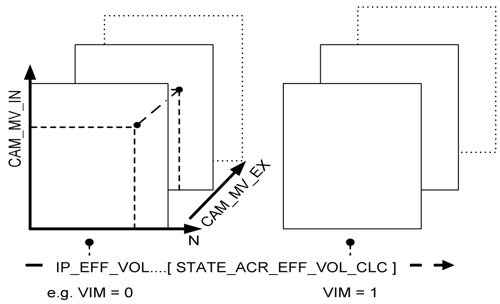
then recurrence every 20 ms

endif

#endif

General information:

With the index STATE_ACR_EFF_VOL_CLC can be switched between different sets of volumetric efficiency maps, depending on the project-configuration (set in the INSY Configuration data).
Example: influence of actuators VIM.



Formula section:

If (FAC_VIM_TMP < C_FAC_VIM_MIN_EFF_VOL_CLC)

Then STATE_ACR_EFF_VOL_CLC = 0

Else STATE_ACR_EFF_VOL_CLC = 1

Endif

4.11.5 Speeded up MAP-acquisition

When the systems runs MAP-controlled (LV_MAP_CTL = 1), it is possible to speed up the load acquisition during transient operating conditions.

At reset: LV_MAP_SAMPLE_DYN = 0

4.11.6 Coordination AMP substitude value for INSY

For naturally aspirated engines:

At reset: AMP_SUB = C_AMP_INI

4.11.7 Cross lock matrix definition

All possible errors and their implications are defined in the cross lock matrix. The output of this matrix results in the settings of the logical variables to stop or to reset to “0” the adaptation or the controller values of the ambient pressure and/or the reduced area.

cross lock matrix ⇒		
	LV_AMP_AD_RST/_STOP	= 0 / 1
	LV_PUT_CTL_STOP	= 0 / 1
	LV_AR_RED_AD_RST/_STOP	= 0 / 1
	LV_AR_RED_CTL_STOP	= 0 / 1

If (LV_ERR_LOAD_TPS_PLAUS = 1 **And** LV_MAF_SWI = 1)
Or LV_INH_MAP_CTL = 1
Or [NC_ETC_CONF == 0 **And** (LV_ERR_TPS_DET = 1 **Or** LV_ERR_ISA_i = 1)]
Then LV_AMP_AD_STOP = 1
LV_AMP_AD_RST = 1
LV_AR_RED_AD_STOP = 1
LV_AR_RED_AD_RST = 1
Else **If** LV_ERR_TPS = 1
Or LV_ERR_LOAD_TPS_PLAUS = 1
Or LV_ERR_LOAD_TPS_PLAUS_FAST = 1
Or (OPG_SP_LIM_ISA < 99,9985% **And** NC_ETC_CONF = 0)
Then LV_AMP_AD_STOP = 1
Else LV_AMP_AD_STOP = 0
Endif
LV_AMP_AD_RST = 0
LV_AR_RED_AD_STOP = 0
LV_AR_RED_AD_RST = 0
Endif

<u>*Determination of LV_TPS_GRD_CTL_STOP*</u> *:*

Target of this flag is to indicate that Reduced Area / Ambient Pressure controllers must be stopped due to transient conditions (high throttle position gradient) during a pre-defined number of cycles.

Following calculation is segment synchronous:

If (TPS_GRD > C_TPS_GRD_CTL_STOP_MAX **And** LV_TPS_GRD_UP =1)
Or (TPS_GRD > C_TPS_GRD_CTL_STOP_MIN **And** LV_TPS_GRD_UP = 0)

```
Then        LV_TPS_GRD_CTL_STOP = 1
            CYC_TPS_GRD_CTL_STOP = C_CYC_TPS_GRD_CTL_STOP
Else        If     LV_TPS_GRD_CTL_STOP = 1
            Then   If     CYC_TPS_GRD_CTL_STOP = 0
                   Then   LV_TPS_GRD_CTL_STOP = 0
                   Else   CYC_TPS_GRD_CTL_STOP = CYC_TPS_GRD_CTL_STOP - 1
                   Endif
            Endif
Endif
```

Conditions to stop Reduced Area / Ambient Pressure controllers :

```
If          LV_TPS_GRD_CTL_STOP = 1
            Or (LV_ERR_LOAD_TPS_PLAUS = 1 And LV_MAF_SWI = 1)
            Or [NC_ETC_CONF == 0 And (LV_ERR_TPS_DET = 1 Or LV_ERR_ISA_i = 1)]
            Or T_AST < ID_T_AST_INSY_CTL_DEAC
Then        LV_AR_RED_CTL_STOP = 1
            LV_PUT_CTL_STOP = 1
Else        LV_AR_RED_CTL_STOP = 0
            LV_PUT_CTL_STOP = 0
Endif
```

4.11.8 Calculation of the relevant segment time and segment counter

Application conditions:

Activation:	multi segment synchronous
Deactivation:	-
Initialization:	SEG_INC = 1
Calculation recurrence:	The calculation is done -depending on the activation condition synchronously to **multiples of a segment**.

General information:

If INSY - depending on the engine speed - is not activated every segment, the segment time T_SEG_AV and the increment of the segment counter have to be adapted.

Formula section:

If (STATE_CLC_SEG_INSY = INSY_SEG_HALF)

Then T_SEG_AV_IMM = T_SEG_HALF_AV

SEG_INC_N = 1 *(only used by modules running segment / double segment synchronous)*

Else **If** ($STATE\_CLC\_RED_N$ <> $STATE\_CLC\_RED_{N-1}$)

Then SEG_INC = (STATE_CLC_RED and NC_STATE_CLC_RED_MASK_1) /NC_STATE_CLC_RED_SEG_1

Else SEG_INC_N = SEG_INC_{N-1} (remains unchanged)

T_SEG_AV_IMM = T_SEG_AV * SEG_INC

Endif

Endif

4.11.9 Flag to inhibit the adaptation of AMP with open throttle (PQ >= C_PQ_AMP_AD)

The ambient pressure adaptation could be inhibited under certain conditions; no such conditions are yet defined, so

Initialization at reset: LV_INH_AMP_AD = 0 (never changed)

4.11.10 Flag to inhibit the adaptation of the additive correction of the reduced area at the throttle

In order to inhibit the adaptation of the reduced area at the throttle a flag is defined that will be set each time the AMP adaptatio is learning during a period of time.

If LV_END_AMP_AD = 1 **And** LV_INH_AMP_AD = 0

Then LV_INH_AR_RED_AD = 0

Else LV_INH_AR_RED_AD = 1

Endif

4.11.11 Variables for fleet monitoring

- Monitoring of MAF:

Activation: LV_IGK = 1 and LV_ST_END = 1

Deactivation: -

Initialization: *at first ECU powver up and non-volatile memory lost:*

MAF_MIN_TOT_DC = FFFFH (1,389E+03 mg/stk)

MAF_MAX_TOT_DC = 0H (0 mg/stk)

-otherwise: restore from non-volatile memory

Calculation recurrence: The calculation is done synchronously to **multiples of a segment**.

Formula section:

If MAF < MAF_MIN_TOT_DC
Then MAF_MIN_TOT_DC = MAF
Endif
If MAF > MAF_MAX_TOT_DC
Then MAF_MAX_TOT_DC = MAF
Endif

4.11.12 - Monitoring of MAP:

Initialisation: - at first ECU power up and non-volantile memory lost:
MAP_MAX_TOT_DC = 0H
MAP_MIN_TOT_DC = FFFFH
- otherwise: restored from non-volantile memory

Recurrence: *The calculation is done synchronously to **multiples of a segment**.*

Activation: *LV_IGK = 1 and LV_ST_END = 1*

Deactivation: -

Formula section:

If MAP > MAP_MAX_TOT_DC
Then MAP_MAX_TOT_DC = MAP
Endif
If MAP < MAP_MIN_TOT_DC
Then MAP_MIN_TOT_DC = MAP
Endif

4.11.13 - Monitoring of PUT_MES:

Initialisation: - at first ECU power up and non-volantile memory lost:
PUT_MES_MAX_TOT_DC = 0H
PUT_MES_MIN_TOT_DC = FFFFH
- otherwise: restored from non-volantile memory

Recurrence: *10 ms*

Activation: *LV_ES = 0 and LV_PUT_SWI = 1*

Deactivation: -

Formula section:

If PUT_MES > PUT_MES_MAX_TOT_DC

Then PUT_MES_MAX_TOT_DC = PUT_MES

Endif

If PUT_MES < PUT_MES_MIN_TOT_DC

Then PUT_MES_MIN_TOT_DC = PUT_MES

Endif

4.11.14 - Monitoring of CAM_OFS_IVVT_IN(_EX):

Initialisation: - at first ECU power up and non-volantile memory lost:
CAM_OFS_IVVT_IN(_EX)_MAX_TOT_DC = 8000H
CAM_OFS_IVVT_IN(_EX)_MIN_TOT_DC = 7FFFH
- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *STATE_CAM_OFS_AD = 5H -> 0H*

Deactivation: -

Formula section:

If CAM_OFS_IVVT_IN(_EX)[1] > CAM_OFS_IVVT_IN(_EX)_MAX_TOT_DC

Then CAM_OFS_IVVT_IN(_EX)_MAX_TOT_DC = CAM_OFS_IVVT_IN(_EX)[1]

Endif

If CAM_OFS_IVVT_IN(_EX)[1] < CAM_OFS_IVVT_IN(_EX)_MIN_TOT_DC

Then CAM_OFS_IVVT_IN(_EX)_MIN_TOT_DC = CAM_OFS_IVVT_IN(_EX)[1]

Endif

4.11.15 - Monitoring of CTR_SP_REQ_CAM_SP_ADJ:

Initialisation: - at first ECU power up and non-volantile memory lost:
CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC = 0H
- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *STATE_CAM_OFS_AD = 3H (Adaptation active)*

Deactivation: -

Formula section:

If CTR_SP_REQ_CAM_SP_ADJ > CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC

Then CTR_SP_REQ_CAM_ADJ_MAX_TOT_DC = CTR_SP_REQ_CAM_SP_ADJ

Endif

4.11.16 - Monitoring of AR_RED_AD_ADD_REQ_CAM_OFS_AD:

Initialisation: - at first ECU power up and non-volantile memory lost:

AR_RED_AD_CAM_AD_MAX_TOT_DC = 8000H

AR_RED_AD_CAM_AD_MIN_TOT_DC = 7FFFH

- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *STATE_CAM_OFS_AD > 1H*

Deactivation: -

Formula section:

If AR_RED_AD_ADD_REQ_CAM_OFS_AD[1] >
AR_RED_AD_CAM_AD_MAX_TOT_DC

Then AR_RED_AD_CAM_AD_MAX_TOT_DC =
AR_RED_AD_ADD_REQ_CAM_OFS_AD[1]

Endif

If AR_RED_AD_ADD_REQ_CAM_OFS_AD[1] <
AR_RED_AD_CAM_AD_MIN_TOT_DC

Then AR_RED_AD_CAM_AD_MIN_TOT_DC =
AR_RED_AD_ADD_REQ_CAM_OFS_AD[1]

Endif

4.11.17 Monitoring of AR_RED_AD_ADD – AR_RED_AD_ADD_MMV:

Initialisation: - at first ECU power up and non-volantile memory lost:
AR_RED_AD_ADD_DIF_MAX_TOT_DC = 8000H
AR_RED_AD_ADD_DIF_MIN_TOT_DC = 7FFFH
- otherwise: restored from non-volantile memory

Recurrence: *2 s*

Activation: *STATE_AR_RED_AD_ADD = 2H* ***AND*** *CTR_AR_RED_AD_ADD_FAST = 0*

Deactivation: -

Formula section:

If AR_RED_AD_ADD – AR_RED_AD_ADD_MMV >
AR_RED_AD_ADD_DIF_MAX_TOT_DC

Then AR_RED_AD_ADD_DIF_MAX_TOT_DC = AR_RED_AD_ADD –
AR_RED_AD_ADD_MMV

Endif

If AR_RED_AD_ADD – AR_RED_AD_ADD_MMV <
AR_RED_AD_ADD_DIF_MIN_TOT_DC

Then AR_RED_AD_ADD_DIF_MIN_TOT_DC = AR_RED_AD_ADD –
AR_RED_AD_ADD_MMV

Endif

4.11.18 - Monitoring of AR_RED_SUM_COR_ACT_REL:

Initialisation: - at first ECU power up and non-volantile memory lost:
AR_RED_SUM_REL_MAX_TOT_DC = 8000H
AR_RED_SUM_REL_MIN_TOT_DC = 7FFFH
- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *LV_IGK = 1*

Deactivation: -

Formula section:

If AR_RED_SUM_COR_ACT_REL > AR_RED_SUM_REL_MAX_TOT_DC

Then AR_RED_SUM_REL_MAX_TOT_DC = AR_RED_SUM_COR_ACT_REL

Endif

If AR_RED_SUM_COR_ACT_REL < AR_RED_SUM_REL_MIN_TOT_DC

Then AR_RED_SUM_REL_MIN_TOT_DC = AR_RED_SUM_COR_ACT_REL

Endif

4.11.19 - Monitoring of PUT_MDL_DIF_MMV_REL:

Initialisation: - at first ECU power up and non-volantile memory lost:
PUT_MDL_DIF_MMV_REL_MAX_TOT_DC = 8000H
PUT_MDL_DIF_MMV_REL_MIN_TOT_DC = 7FFFH
- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *LV_IGK = 1*

Deactivation: -

Formula section:

If PUT_MDL_DIF_MMV_REL > PUT_MDL_DIF_MMV_REL_MAX_TOT_DC

Then PUT_MDL_DIF_MMV_REL_MAX_TOT_DC = PUT_MDL_DIF_MMV_REL

Endif

If PUT_MDL_DIF_MMV_REL < PUT_MDL_DIF_MMV_REL_MIN_TOT_DC

Then PUT_MDL_DIF_MMV_REL_MIN_TOT_DC = PUT_MDL_DIF_MMV_REL

Endif

4.11.20 Calculation of MAF- Signal for Calibration Purpose

General information:

The calculation of the MAF Signal for calibration purpose is done same way as the MAF Signal for System. The factor for MAF trimming value is not included, as no Trimming will be done for this input.

Activation: LV_MAF_CAL_SWI = 1

Deactivation: LV_MAF_CAL_SWI = 0

Initialization: After HW reset, the buffers - MAF_SUM_CAL and MAF_SUM_1_CAL - and the corresponding counters - MAF_CTR_CAL and MAF_CTR_1_CAL - have to be cleared and the pointer has to be initialized with the address of ID_MAF_TAB.

LV_MAF_SEG_CAL=0

Calculation recurrence:

Segment synchr: Calculation of MAF_KGH_MES_BAS_CAL

1ms Calculation of MAF_SUM_CAL and MAF_SUM_1_CAL

Formula section:

if LV_MAF_SEG_CAL = 0

then MAF_CTR_CAL = MAF_CTR_CAL + 1

MAF_SUM_CAL = MAF_SUM_CAL + ID_MAF_TAB(VP_MAF_CAL)

else **if** LV_MAF_SEG_CAL == 1

then MAF_CTR_1_CAL = MAF_CTR_1_CAL + 1

MAF_SUM_1_CAL = MAF_SUM_1_CAL + ID_MAF_TAB(VP_MAF_CAL),

endif

endif

Calculation of MAF_KGH_MES_BAS_CAL:

Calculation recurrence: If the engine is running on a fixed tooth each segment.

LV_MAF_SEG_CAL = ! LV_MAF_SEG_CAL

if LV_MAF_SEG_CAL = 1

and MAF_CTR_CAL > 0

then

$$\mathrm{MAF\_KGH\_MES\_BAS\_CAL} = \frac{\mathrm{MAF\_SUM\_CAL}}{2 * \mathrm{MAF\_CTR\_CAL}}$$

MAF_SUM_CAL = MAF_CTR_CAL = 0

else

if (LV_MAF_SEG_CAL == 0)

and (MAF_CTR_1_CAL > 0)

then

$$\mathrm{MAF\_KGH\_MES\_BAS\_CAL} = \frac{\mathrm{MAF\_SUM\_1\_CAL}}{2 * \mathrm{MAF\_CTR\_1\_CAL}}$$

```
            MAF_SUM_1_CAL = MAF_CTR_1_CAL = 0
        endif
endif
```

If the selected counter is zero MAF_KGH_MES_BAS_CAL stay unchanged.

Calculation of MAF_KGH_MES_CAL and MAF_MES_CAL:

MAF_KGH_MES_CAL = MAF_KGH_MES_BAS_CAL * IP_MAF_KGH_MES_FAC

MAF_MES_CAL = MAF_KGH_MES_CAL / N * NC_MAF_FAC_CYL

4.11.21 Interface to other functions:

- **flow from canisterpurge-solenoid:**

FLOW_CPS = MAF_CPS

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TPS_GRD_CTL_STOP_MAX	1	0...FFH	0...2987.5	11.72	°TPS/sec
High TPS_GRD threshold to stop Reduced Area controller					
C_TPS_GRD_CTL_STOP_MIN	1	0...FFH	0...2987.5	11.72	°TPS/sec.
Low TPS_GRD threshold to stop Reduced Area controller					
IP_N_MAF_MES_FAC_OFS	6	FC18...03E8H	-1000...+1000	1	Rpm
LDP_TIA_N_MAF_MES_FAC_OFS	6	0...FEH	-48...142,5	0,75	°C
Correction of measured Mass Air Flow					
IP_MAF_KGH_MES_FAC	24*8	0...FFH	0...1.992	0.0078	-
LDP_N_MAF_KGH_MES_FAC	24	0...1FE0H	0...8160	1	Rpm
LDP_TPS_SEG_MAF_KGH_MES_FAC	8	0...3FFFH	0...119.5	0.0073	°TPS
Correction of measured Mass Air Flow					
C_FAC_RATIO_MAX	1	0...FFH	0...0.996	0.00392	-
MAF_CYL/MAP thd. based on MAF_MAX_TOT/AMP to detect high TPS failure, only for NC_ETC_CONF = 0					
C_FAC_RATIO_MAX_DIAG	1	0...FFH	0...0.996	0.00392	-
MAF_MES/MAP_MES thd. based on MAF_MAX_TOT/AMP to detect high TPS failure, only for NC_ETC_CONF = 0					
C_FAC_RATIO_MIN	1	0...FFH	0...0.996	0.00392	-
MAF_CYL/MAP thd. based on MAF_MAX_TOT/AMP to detect low TPS failure, only for NC_ETC_CONF = 0					
C_FAC_RATIO_MIN_DIAG	1	0...FFH	0...0.996	0.00392	-
MAF_MES/MAP_MES thd. based on MAF_MAX_TOT/AMP to detect low TPS failure, only for NC_ETC_CONF=0					
IP_TPS_SEG_LOAD_MAX	8	0...3FFFH	0...119.5	0.0073	°TPS
LDPM_N_32_TPS_SEG_LOAD_MAX	8	0...FFH	0...8160	32	Rpm
TPS_SEG position to have MAF/MAP equal to Full Load MAF/MAP, only for NC_ETC_CONF = 0					
C_TPS_GRD_TRA	1	0...FFH	0...2987.5	11.72	°TPS/sec.
Throttle gradient thd. to detect acceleration or deceleration, only for NC_ETC_CONF = 0					
IP_OPG_SP_LIM_ISA	6 * 4	0...FFFFH	0...99.9985	0.0015	%
LDP_TPS_AV_IP_OPG_SP_LIM_ISA	6	0...3FFFH	0...119.5	0.0073	°TPS
LDP_TCO_IP_OPG_SP_LIM_ISA	4	0...FEH	-48...142.5	0.75	°C
Idle Speed Actuator opening limitation in case of suspected TPS failure, only for NC_ETC_CONF = 0					
C_AR_RED_DIF_REL_OPG_LIM_ISA	1	8000...7FFFH	-50...49.998	0.0015	%
Reduced Area controller deviation to activate ISA opening limitation in case of suspected TPS failure, only for NC_ETC_CONF = 0					
C_CYCNR_OPG_LIM_ISA	1	1...FFH	1...255	1	-
Minimum number of segments to activate ISA opening limitation in case of suspected TPS failure, only for NC_ETC_CONF = 0					
IP_MAP_MDL_DIF_OPG_LIM_ISA	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hpa
LDPM_N_32_TPS_SEG_LOAD_MAX	8	0...FFH	0...8160	32	rpm
MAP_MDL_DIF deviation to activate ISA opening limitation in case of suspected TPS failure, only for NC_ETC_CONF = 0					
C_N_MIN_RATIO_CHK	1	00...FFH	0...8160	32	Rpm
Minimum engine speed to enable MAF/TPS Ratio check, only for NC_ETC_CONF = 0					
IP_MAF_TPS_DIAG	1*6	00...FFFFH	0...2047.96875	0.03125	kg/h
LDPM_N_32_1_INSY	6	00...FFH	0...8160	32	rpm
MAF in case of TPS and MAF failure at the same time					

IP_FAC_MAF_TRIM_COR	1*4	00...FFH	0... 4.9805	0.0195	-
LDP_VP_MAF_IP_FAC_MAF_TRIM_COR	4	00...3FFH	0...4,995	0,00488	V
Correction of MAF Signal Trimming Factor depending on MAF Signal Voltage					
C_CTR_SEG_LIH_MAF	1	0...FFH	0...255	1	-
Max value for CTR_SEG_LIH_MAF, only for NC_ETC_CONF = 1					
C_CYC_TPS_GRD_CTL_STOP	1	0...FFH	0...255	1	-
Number of cycles to stop Reduced Area / Ambient Pressure controller in case of high TPS gradient					
ID_T_AST_INSY_CTL_DEAC	4	0…FFFFH	0…6553.5	0.1	Sec.
LDP_TCO_ST_INSY_CTL_DEAC	4	0...FEH	-48...142.5	0.75	°C
Minimum time after Start to activate Intake Manifold Model controllers					
Switch to use mass airflow sensor for calibration purposes					
C_MAF_MES_FIL_ST	1	0...FFFFH	0...1389	0.0212	mg/stk
MAF_MES_FIL initialisation in Start					
C_FAC_VIM_MIN_EFF_VOL_CLC	1	0...FFH	0...0.996	0.0039	-
Value to switch on/off variable intake manifold correction for volumetric efficiency					
C_CRLC_MAF_MES_FIL	1	0...FFH	0...0.996	0.039	-
Filter constant for MAF_MES_FIL calculation					
IP_MAF_MDL_DIF_OPG_LIM_ISA	8	0...FFFFH	-1024...1023.97	0.03125	Kg/h
LDPM_N_32_TPS_SEG_LOAD_MAX	8	0...FFH	0...8160	32	rpm
MAF_MDL_DIF deviation to activate ISA opening limitation in case of suspected TPS failure, only for NC_ETC_CONF = 0					
LC_EFF_VOL_INP_CAM_OFS_INH	1	0...1H	0...1	1	[-]
Logical bit to inhibit the influence of CAM_OFS_IVVT_IN/EX on the input variables used for EFF_VOL calculation					
IP_FAC_AR_RED_THR_COR	12*16	0...FFFFH	0...1.99996	0.0305e-3	[-]
LDP_N_AR_RED_THR_COR	12	0...FFH	0...8160	32	[rpm]
LDP_TPS_SEG_AR_RED_THR_COR	16	0...3FFFH	0...119.5	7.2941e-3	[°TPS]
correction of AR_RED_BAS, depending on N_32 and TPS_SEG					
C_CAM_OFS_MAX	1	0...7FFFH	0...47.9985352	0.00146484	°CRK
Maximum value of adaptation results					
C_CAM_OFS_MIN	1	0...7FFFH	0...47.9985352	0.00146484	°CRK
Maximum value of adaptation results					
C_LAMB_CTL_INH_MAP_MAX	1	8000...7FFFH	-50...49.9984741	0.00152588	%
Upper threshold for identification of a MAP-sensor-failure in case of a Load-TPS-error					
C_LAMB_CTL_INH_MAP_MIN	1	8000...7FFFH	-50...49.9984741	0.00152588	%
Lower threshold for identification of a MAP-sensor-failure in case of a Load-TPS-error					
C_CRLC_LAMB_CTL_SUM_INH_MAP	1	0...FFH	0...0.99609375	0.00390625	-
Constant for LAMB_CTL_SUM_INH_MAP filtering					

4.12 Air mass limits

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_AMP_TIA_COR	O/V	0...FFFFH	0...1.999969	3.052E-5	-
Air density correction					
FAC_MAF_MAX	O/V	0...FFFFH	0...1.999969	3.052E-5	-
Ratio between MAF and MAF_MAX_COR					
MAF_MAX_COR	O/V	0...FFFFH	0...1.389E+3	0.0212	mg/stk
Maximum possible fresh air entering in the cylinder (open throttle)					
MAF_MAX_TOT	O	0...FFFFH	0...1.389E+3	0.0212	mg/stk
Total maximum of the fresh air going into the cylinder (for current engine speed and wide open throttle)					
MAF_MIN_COR	O	0...FFFFH	0...1.389E+3	0.0212	mg/stk
Minimum available fresh air going into the cylinder					

Input data:

EFF_VOL_OFS_SUM_MAX_COR	EFF_VOL_SLOP_MAX_COR	MAF	N
PUT	PUT_MAX	TIA_THR	AMP
NC_MAF_FAC_CYL	PV_AV_H		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_FAC_AMP_TIA_COR_SWI	1	0...1H	0...1	1	-
Switch for FAC_AMP_TIA_COR calculation for charger (0: using maps, 1: proportional to PUT divided by TIA)					
LC_MAF_MAX_SWI	1	0...1H	0...1	1	-
Switch for MAF_MAX calculation (0: calculation with full load map, 1: calculation with volumetric efficiency)					
IP_MAF_MAX_STND	8	0...FFFFH	0...1.389E+3	0.0212	mg/stk
LDPM_N_4_INSY	8	0...1FE0H	0...8.16E+3	1	rpm
Maximum intake air of the engine at standardized ambient pressure					
IP_FAC_AMP_TQ	12x12	0...FFFFH	0...1.999969	3.052E-5	-
LDP_AMP_IP_FAC_AMP_TQ	12	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDP_PV_AV_H_IP_FAC_AMP_TQ	12	0...3FFH	0...99.9	0.09765396	%
Ambient pressure correction					
IP_FAC_MAF_MAX_AMP	8x6	0...FFFFH	0...1.999969	3.052E-5	-
LDPM_N_4_INSY	8	0...1FE0H	0...8.16E+3	1	rpm
LDP_AMP_IP_FAC_MAF_MAX_AMP	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
Correction factor for the maximum intake air depending on the ambient pressure					
IP_FAC_MAF_MAX_TIA	8x6	0...FFFFH	0...1.999969	3.052E-5	-
LDPM_N_4_INSY	8	0...1FE0H	0...8.16E+3	1	rpm
LDP_TIA_IP_FAC_MAF_MAX_TIA	6	0...FEH	-48...142.5	0.75	°C
Correction factor for the maximum intake air depending on the intake air temperature					

4.12.1 INSY_M405W

General information:

This module calculates the maximum possible air mass flow that is available to the Engine Management system.

The maximum possible air-mass flow into the cylinder will be reached with wide open throttle (i.e. MAP = PUT). For the calculation it is assumed, that at open throttle there will be no EGR-ratio active.

Application Condition

Activation: At every engine state

Deactivation: -

Initialization: -

Recurrence: 40 ms

Application Condition

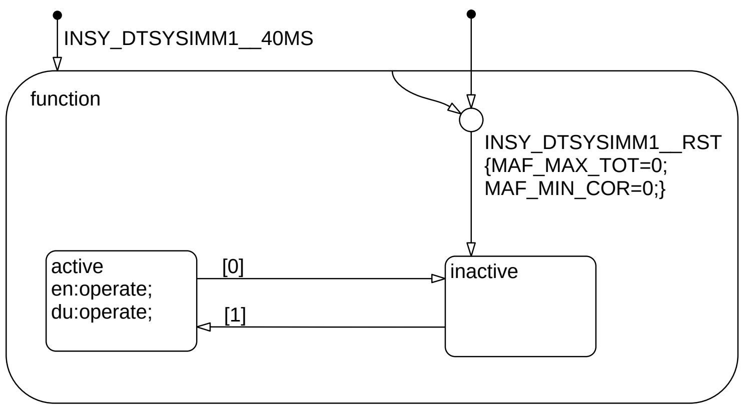


Figure 127 INSY_M405W/ APP_CDN/ Chart

Function Description

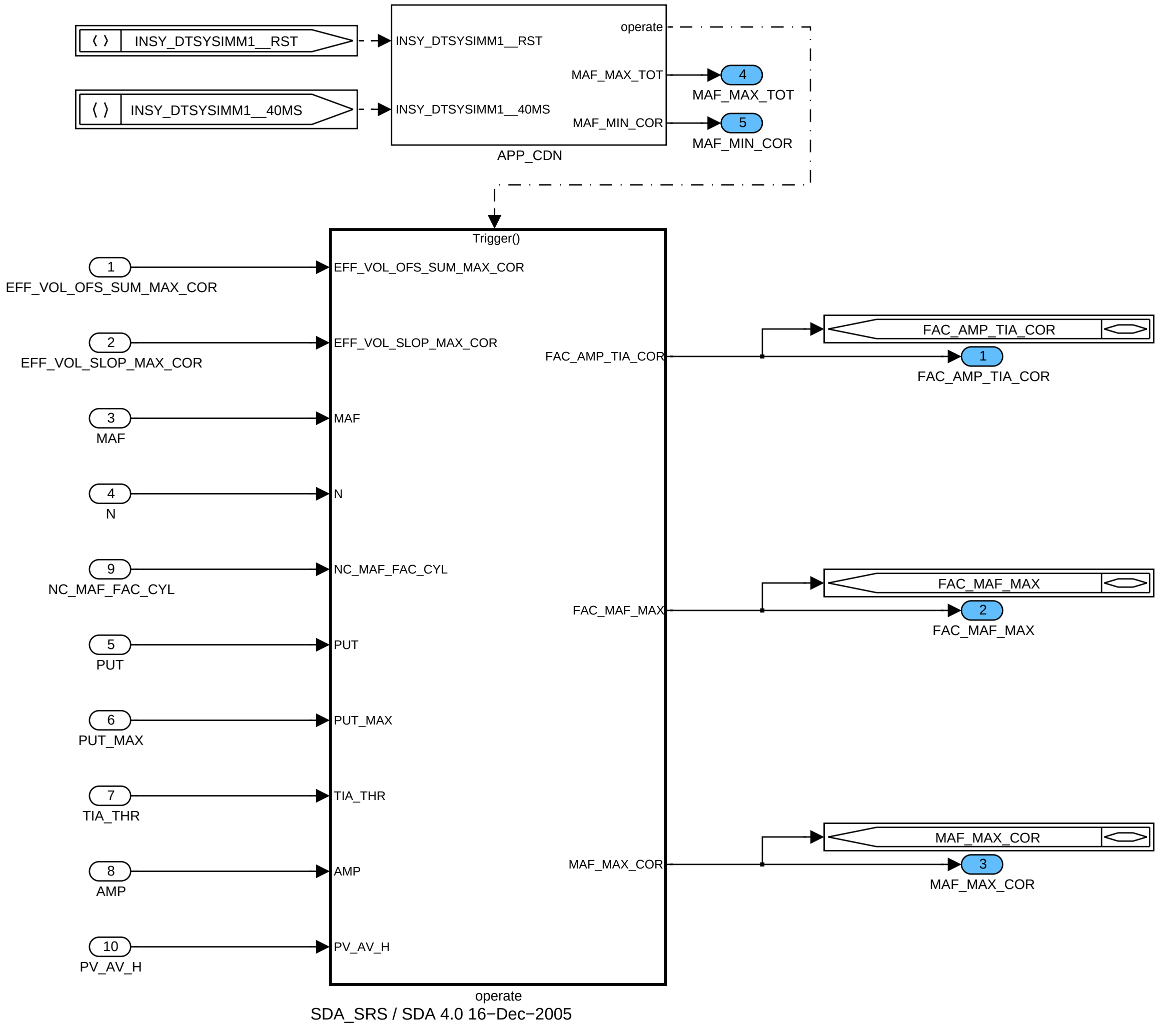


Figure 128 INSY_M405W

4.12.1.1 Calculation of maximum possible mass air flow and the ratio between actual and maximum mass air flow

The maximum air mass flow going into the cylinder at the current engine speed will be reached with wide open throttle. To calculate this air mass, the manifold pressure is assumed to be equal to the pressure up throttle PUT.

It is possible to switch between two algorithms:

If LC_MAF_MAX_SWI = 0 the calculation is done with a full load map. The standard value from this map is corrected with a factor depending on ambient pressure and intake air temperature. For naturally aspirated engines this factor is proportional to the ambient pressure divided by the intake air teperature in Kelvin. For charged engines an alternative calculation with two interpolation maps can be activated with LC_FAC_AMP_TIA_COR_SWI = 0.

If LC_MAF_MAX_SWI = 1 the calculation is done with a maximum slope and offset and transformed into the air mass MAF_MAX_COR [mg/stk] by dividing by the engine speed and applying the factor NC_MAF_FAC_CYL (depending on the number of cylinders).

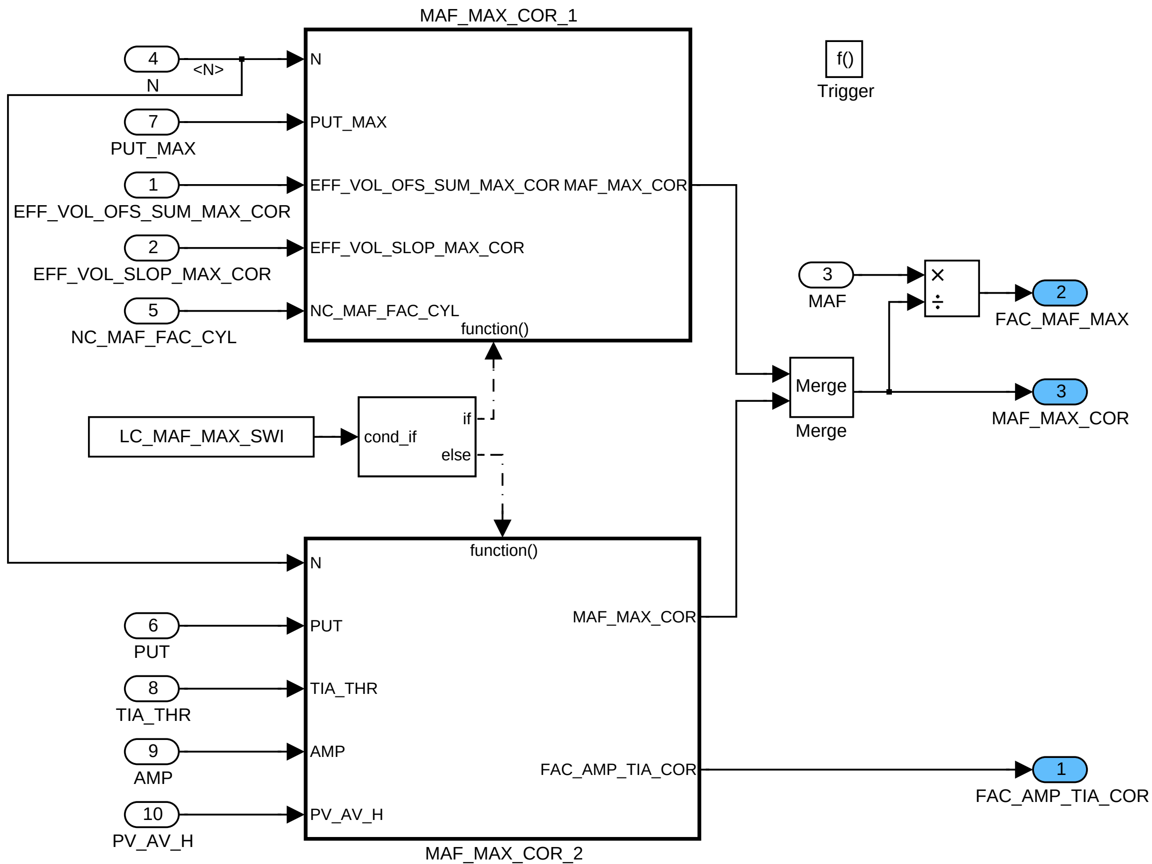


Figure 129 INSY_M405W/ operate

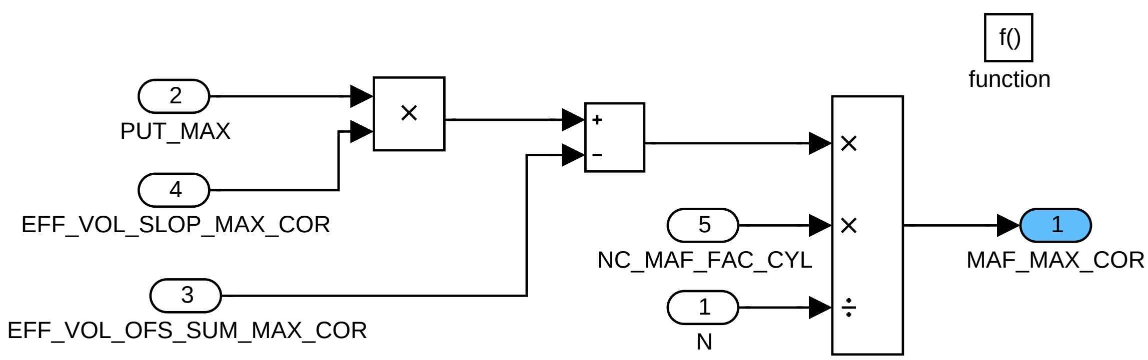


Figure 130 INSY_M405W/ operate/ MAF_MAX_COR_1

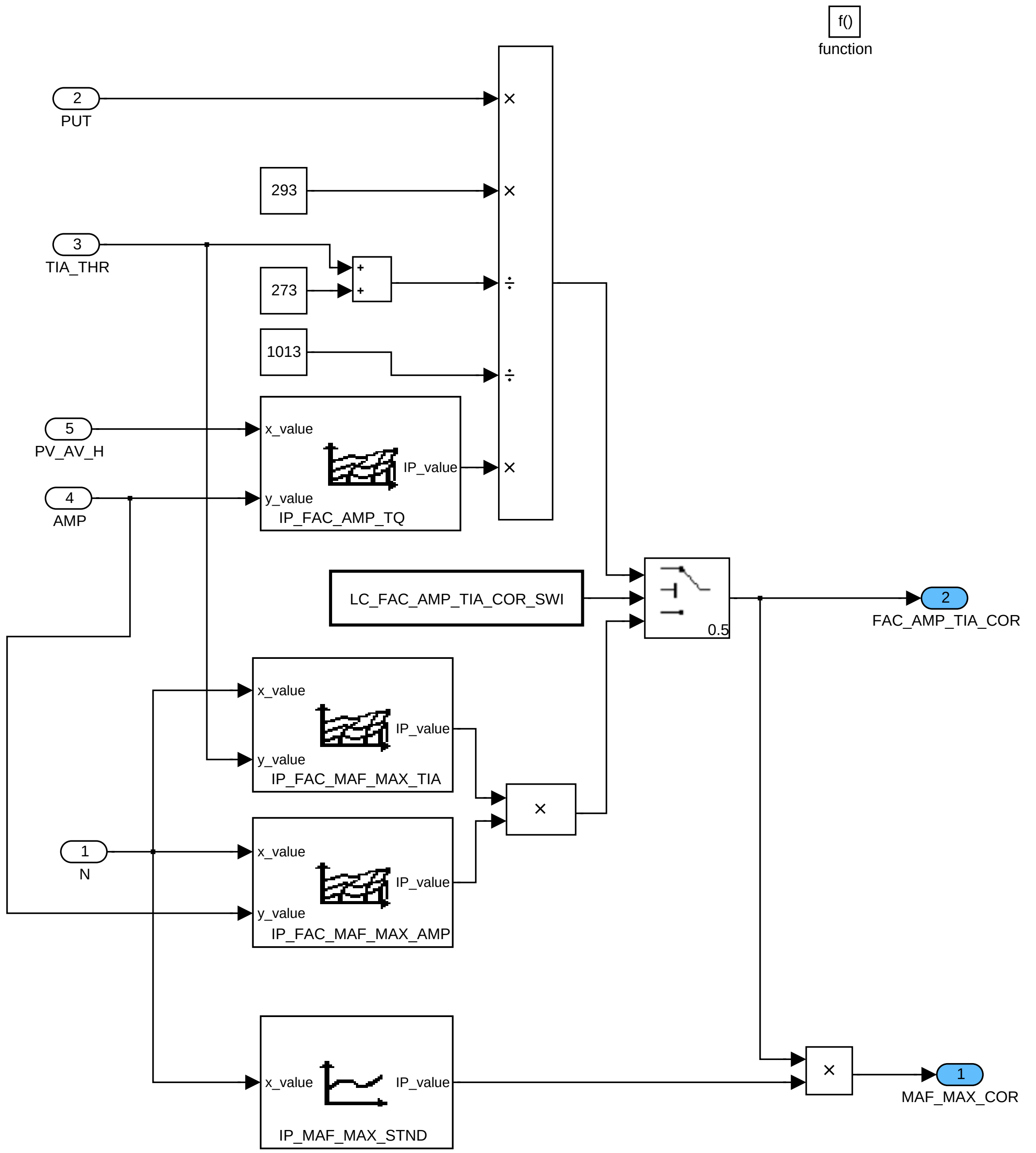


Figure 131 INSY_M405W/ operate/ MAF_MAX_COR_2

4.13 Air mass flow integral calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_INT_PUC_ACT	O/V	0...FFFFH	0...2.91267E+3	0.04444444	g
air mass flow integral during pull cut off phase					
MAF_INT_PUC_NOT_ACT	O/V	0...FFFFH	0...2.91267E+3	0.04444444	g
air mass flow integral out of pull cut off phase					

Input data:

LV_ST_END	LV_PUC	MAF_CYL	

4.13.1 INSY_REQCOMAFIN0

For calatyst enrichment function and for plausibility checks of lambda sensor diangosis functions air mass flow integrals during and out of trailing throttle fuel cut-off phase must be calculated.

Function Description

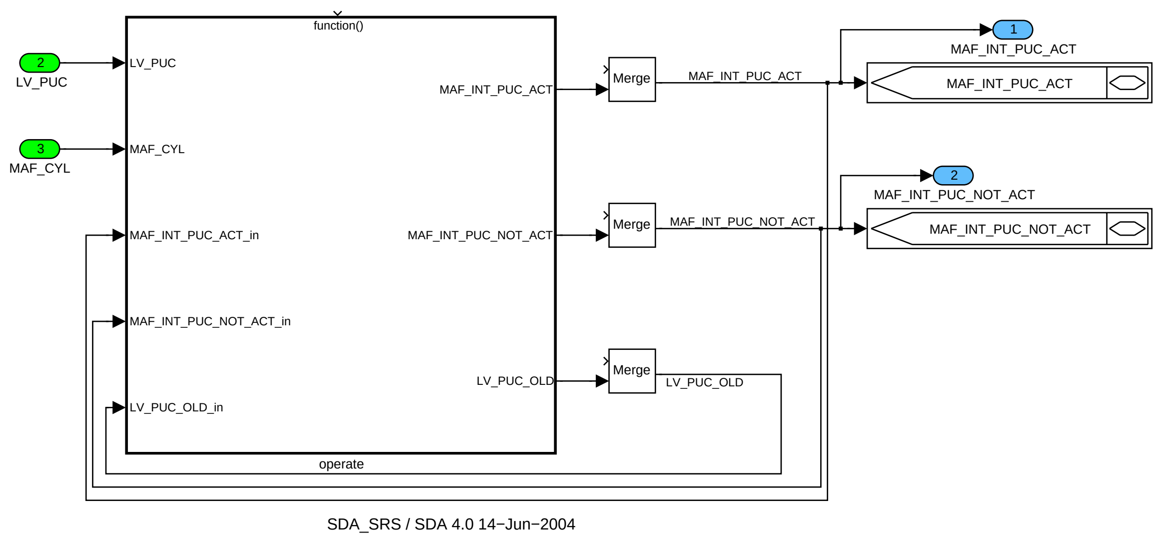


Figure 132 INSY_REQCOMAFIN0

4.13.1.1 operate

During trailing throttle fuel cut-off phase (LV_PUC = 1) the air mass flow integral MAF_INT_PUC_ACT is calculated. The integral is reset to 0 at the rising edge of LV_PUC.

Out of trailing throttle fuel cut-off phase the integral MAF_INT_PUC_NOT_ACT is calculated. It is reset to 0 at the falling edge of LV_PUC.

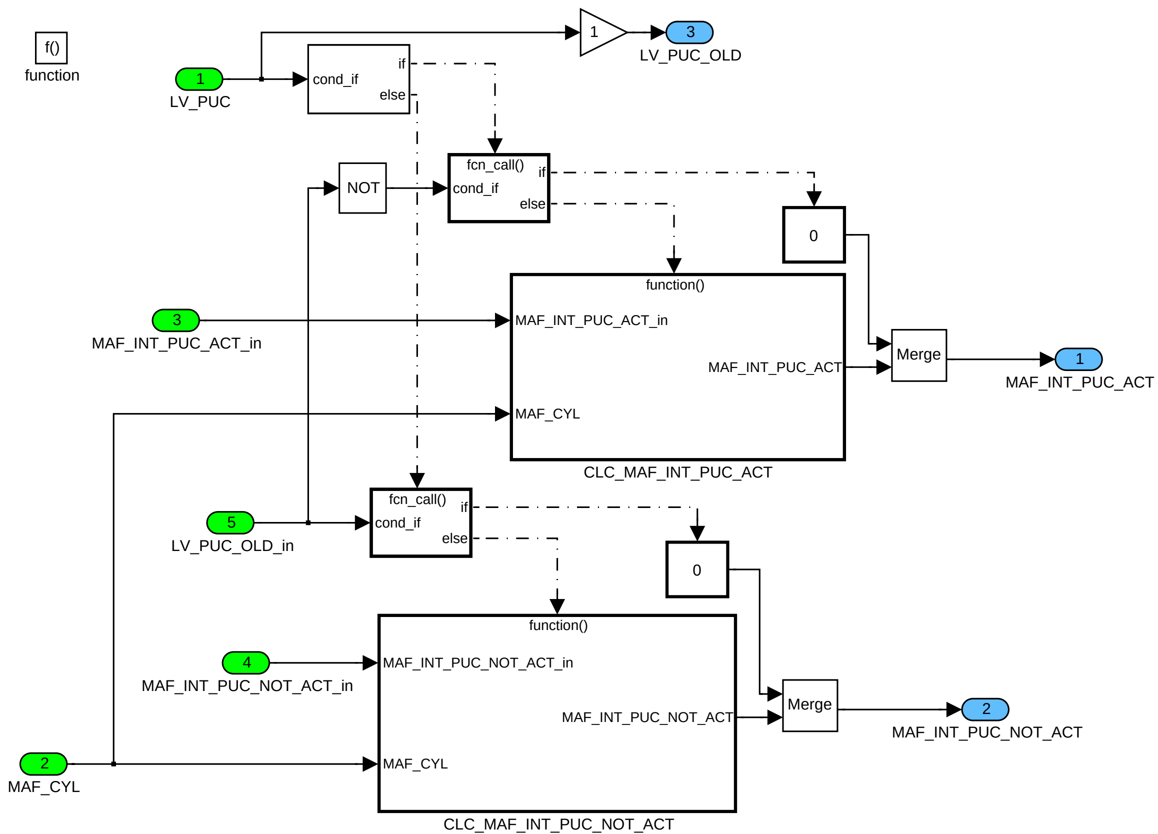


Figure 133 INSY_REQCOMAFIN0/ operate

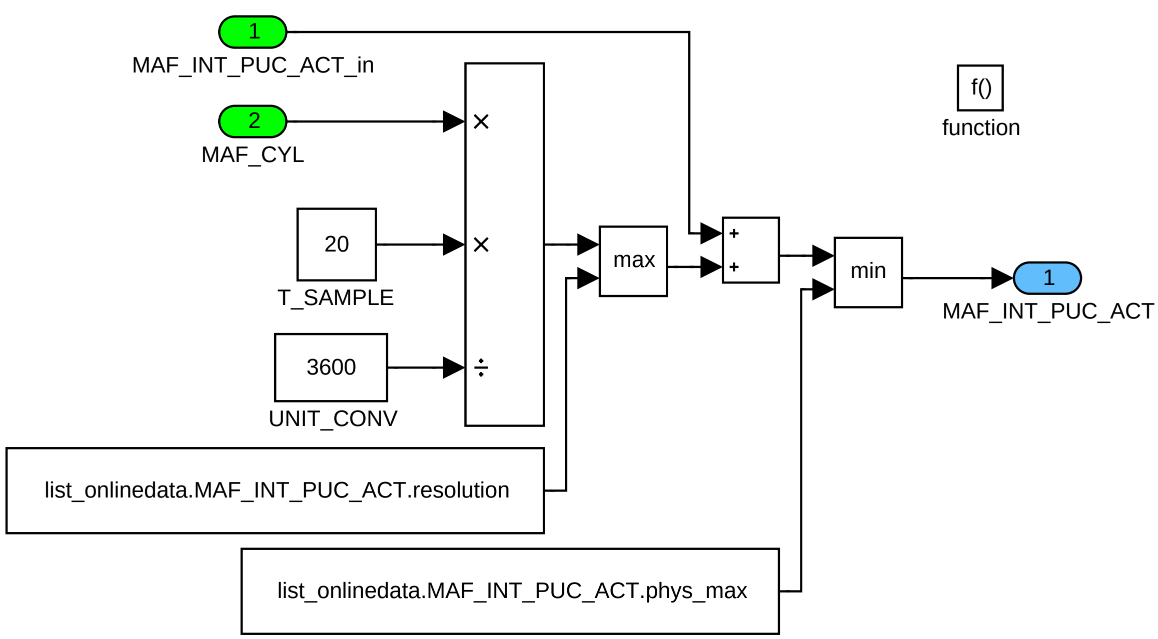


Figure 134 INSY_REQCOMAFIN0/ operate/ CLC_MAF_INT_PUC_ACT

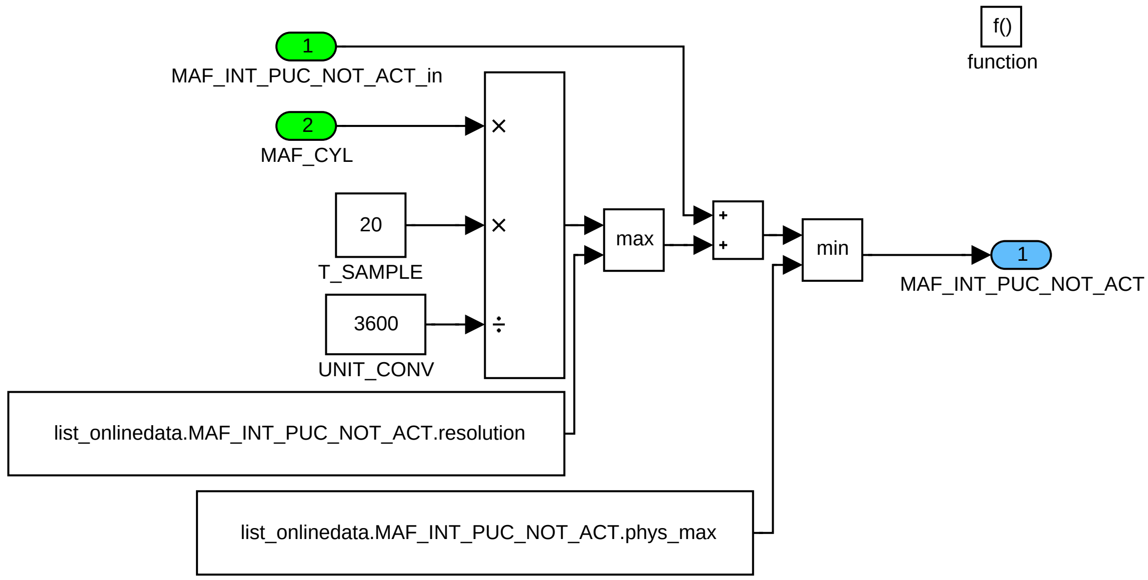


Figure 135 INSY_REQCOMAFIN0/ operate/ CLC_MAF_INT_PUC_NOT_ACT

4.14 INSY-Controller (Reduced Area and Pressure upstream throttle controller)

Overview

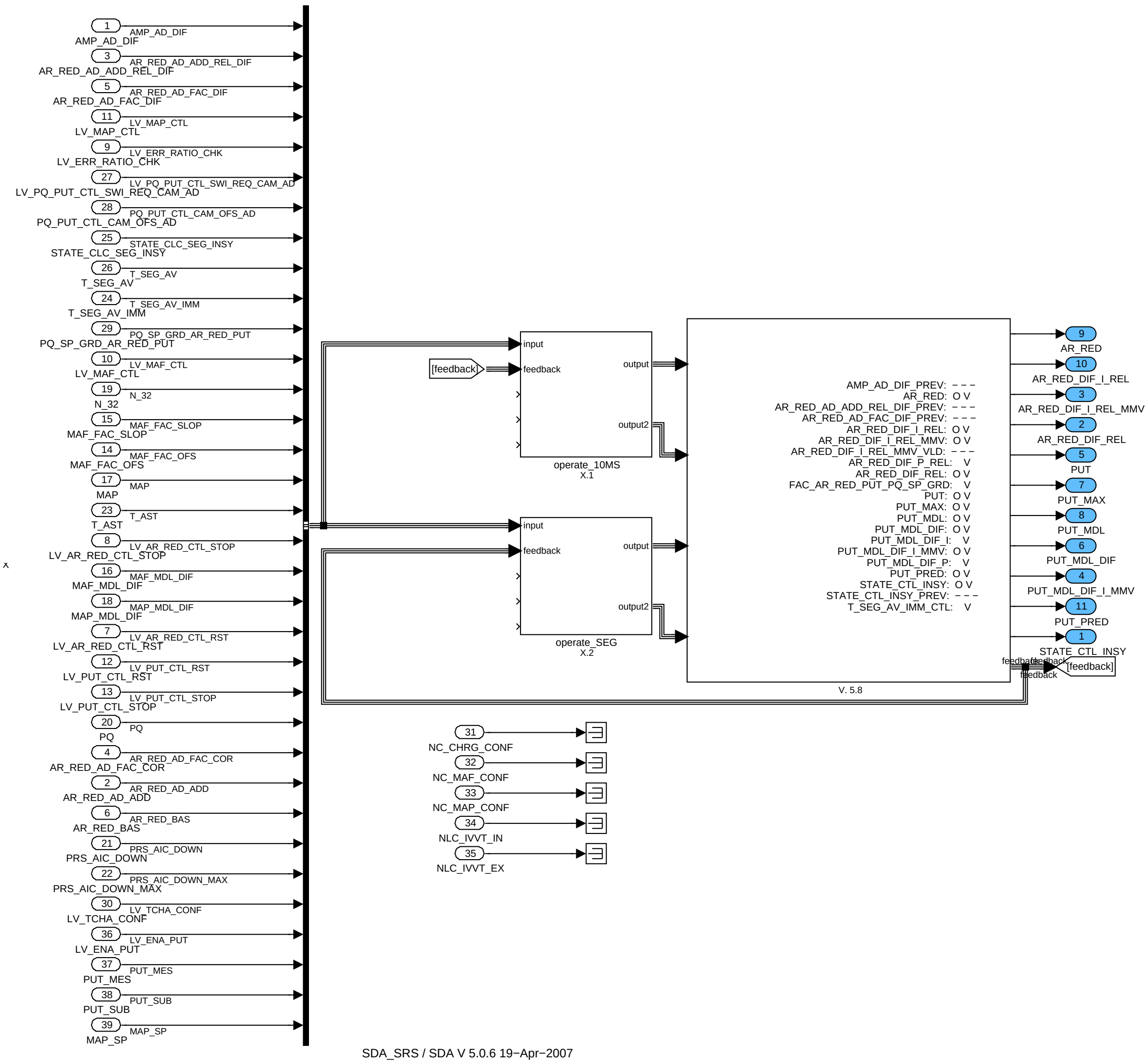


Figure 136:
Path: INSY_M404T

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
AMP_AD_DIF_PREV	-	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Previous AMP_AD_DIF = change of the adapted ambient pressure					
AR_RED	O/V	0... FFFFH	0... 58.592855	894.07e-6	[cm²]
Reduced area at the throttle					
AR_RED_AD_ADD_REL_DIF_PREV	-	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Previous AR_RED_AD_ADD_REL_DIF = Increment of additive adaptive parameter (relative)					
AR_RED_AD_FAC_DIF_PREV	-	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Previous AR_RED_AD_FAC_DIF = Increment of multiplicative adaptive parameter					
AR_RED_DIF_I_REL	O/V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
I-value (relative) of the closed loop control reduced throttle area					
AR_RED_DIF_I_REL_MMV	O/V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Moving mean I-value (relative) of the closed loop control reduced throttle area					
AR_RED_DIF_I_REL_MMV_VLD	-	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Last valid moving average of I-part (relative) of the closed loop control reduced throttle area					
AR_RED_DIF_P_REL	V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
P-value (relative) of the closed loop control reduced throttle area					
AR_RED_DIF_REL	O/V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Correction of reduced throttle area = PI-controller output					
AR_RED_I_REL	-	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
I-value increment (relative) of the closed loop control reduced throttle area					
FAC_AR_RED_PUT_PQ_SP_GRD	V	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
Ramping factor for INSY controller I share for PQ_SP jumps					
PQ_PUT_CTL	-	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Pressure quotient to switch from area to pressure control					
PQ_TMP	-	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure quotient at the throttle for INSY controller					
PUT	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure upstream throttle					
PUT_MAX	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Maximum possible pressure upstream throttle					
PUT_MDL	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Modeled pressure upstream throttle					
PUT_MDL_DIF	O/V	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Correction of pressure upstream throttle = PI-controller output					
PUT_MDL_DIF_I	V	8000... 7FFFH	-339.630182345 ...339.619817655	0.0103647	[hPa]
I-value of the closed loop control of pressure upstream throttle					
PUT_MDL_DIF_I_MMV	O/V	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Moving mean I-value of the closed loop control of pressure upstream throttle					
PUT_MDL_DIF_I_TMP	-	8000... 7FFFH	-339.630182345 ...339.619817655	0.0103647	[hPa]
I-value increment of the closed loop control of pressure upstream throttle					

PUT_MDL_DIF_P	V	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
P-value of the closed loop control of pressure upstream throttle					
PUT_PRED	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Predicted pressure upstream throttle					
STATE_CTL_INSY	O/V	0 1 2 3 4	INSY_MAF_AR_CTL INSY_MAF_PUT_CTL INSY_MAP_AR_CTL INSY_MAP_PUT_CTL INSY_OPL_CTL	-	[-]
State of intake manifold model adaptation					
STATE_CTL_INSY_PREV	-	0 1 2 3 4	INSY_MAF_AR_CTL INSY_MAF_PUT_CTL INSY_MAP_AR_CTL INSY_MAP_PUT_CTL INSY_OPL_CTL	-	[-]
Previous state of intake manifold model adaptation					
T_SEG_AV_IMM_CTL	V	0... FFFFH	0... 262.14	4e-3	[ms]
Segment time for INSY controller only					

Input Data:

AMP_AD_DIF	AR_RED_AD_ADD	AR_RED_AD_ADD_REL_DIF	AR_RED_AD_FAC_COR
AR_RED_AD_FAC_DIF	AR_RED_BAS	LV_AR_RED_CTL_RST	LV_AR_RED_CTL_STOP
LV_ERR_RATIO_CHK	LV_MAF_CTL	LV_MAP_CTL	LV_PUT_CTL_RST
LV_PUT_CTL_STOP	MAF_FAC_OFS	MAF_FAC_SLOP	MAF_MDL_DIF
MAP	MAP_MDL_DIF	N_32	PQ
PRS_AIC_DOWN	PRS_AIC_DOWN_MAX	T_AST	T_SEG_AV_IMM
STATE_CLC_SEG_INSY	T_SEG_AV	LV_PQ_PUT_CTL_SWI_REQ_CAM_AD	PQ_PUT_CTL_CAM_OFS_AD
PQ_SP_GRD_AR_RED_PUT	LV_TCHA_CONF	NC_CHRG_CONF	NC_MAF_CONF
NC_MAP_CONF	NLC_IVVT_IN	NLC_IVVT_EX	LV_ENA_PUT
PUT_MES	PUT_SUB	MAP_SP	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_AR_RED_DIF_I_REL_MAX	1	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Maximum value for AR_RED_DIF_I_REL					
C_AR_RED_DIF_I_REL_MIN	1	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Minimum value for AR_RED_DIF_I_REL					
C_CRLC_AR_RED_DIF_I_REL_MMV	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Correlation constant for the moving men value calculation of AR_RED_DIF_I_REL					
C_CRLC_PUT_MDL_DIF_I_MMV	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Correlation constant for moving mean value calculation of PUT_MDL_DIF_I					
C_FAC_AR_RED_DIF_I_REL_INI	1	0... FFH	0... 1.9921875	7.8125e-3	[-]

Scaling factor for part of PUT_MDL_DIF_I wich is transformed to AR_RED_DIF_I_REL_INI					
C_LGRD_AR_RED_DIF_TRA	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Linear negative gradient of reduced area control during pressure control					
C_LGRD_PUT_MDL_DIF_I_TRA	1	0... FFFFH	0... 679.25	0.0103647	[hPa]
Linear negative gradient of PUT_MDL_DIF_I during reduced area control					
C_LGRD_PUT_MDL_DIF_TRA	1	0... FFFFH	0... 5434	0.0829175	[hPa]
Linear negative gradient of pressure control during reduced area control					
C_PQ_PUT_CTL	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Pressure quotient to switch from area to pressure control					
C_PQ_PUT_CTL_HYS	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Hysteresis around pressure ratio to switch from area to pressure control					
C_PUT_MDL_DIF_I_MAX	1	8000... 7FFFH	-339.630182345 ...339.619817655	0.0103647	[hPa]
Upper threshold of integral part of pressure controller					
C_PUT_MDL_DIF_I_MIN	1	8000... 7FFFH	-339.630182345 ...339.619817655	0.0103647	[hPa]
Lower threshold of integral part of pressure controller					
C_T_AST_INSY_CTL_DEAC	1	0... FFFFH	0... 6553.5	0.1	[s]
Time to deactivate INSY control after engine start					
IP_AR_RED_I_REL_1	6*4	0... FFFFH	0... 171660	2.6193637	[%/kg]
LDPM_N_32_1_INSY	6	0... FFH	0... 8160	32	[rpm]
LDPM_PQ_1_INSY	4	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
I-value (relative) reduced throttle area (only with MAF-controlled: MAF_CONF=1)					
IP_AR_RED_I_REL_2	6*4	0... FFFFH	0... 2.24636	34.2773e-6	[%/(hPa*s)]
LDPM_N_32_1_INSY	6	0... FFH	0... 8160	32	[rpm]
LDPM_PQ_1_INSY	4	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
I-value (relative) reduced throttle area (only with MAP-controlled: MAP_CONF=1)					
IP_AR_RED_P_REL_1	6*4	0... FFFFH	0... 12.5005	190.745e-6	[(%*h)/kg]
LDPM_N_32_1_INSY	6	0... FFH	0... 8160	32	[rpm]
LDPM_PQ_1_INSY	4	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
P-value reduced throttle area (only with MAF-controlled: MAF_CONF=1)					
IP_AR_RED_P_REL_2	6*4	0... FFFFH	0... 0.58886	8.98543e-6	[%/hPa]
LDPM_N_32_1_INSY	6	0... FFH	0... 8160	32	[rpm]
LDPM_PQ_1_INSY	4	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
P-value reduced throttle area (only with MAP-controlled: MAP_CONF=1)					
IP_FAC_AR_RED_PUT_PQ_SP_GRD	4*6	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_PQ_SP_GRD_IP_FAC_AR_RED_PUT	6	0... FFFFH	-1... 0.99996948	30.5176e-6	[-]
LDP_PQ_IP_FAC_AR_RED_PUT_PQ_GRD	4	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Ramping factor for INSY controller I share for PQ_SP jumps					
IP_FAC_PRED_PUT	6*6	0... FFFFH	0... 8192	0.1250019	[-]
LDPM_N_32_4_CHRG	6	0... FFH	0... 8160	32	[rpm]
LDPM_MAP_SP_PRS_CHA_UP_1_CHRG	6	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Prediction horizon of PUT_MMV for PUT calculation					
IP_FAC_PRED_PUT_RAMP	4	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_PUT_SUB_DIF_IP_FAC_PRED_PUT	4	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]

Factor to ramp down PUT prediction horizon near basic charge pressure					
IP_FAC_PUT_SUB_PUT	6*6	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDPM_N_32_4_CHRG	6	0... FFH	0... 8160	32	[rpm]
LDPM_MAP_SP_PRS_CHA_UP_1_CHRG	6	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Weighting factor of PUT_SUB for PUT calculation					
IP_FAC_PUT_SUB_PUT_2	4	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_PUT_SUB_DIF_IP_FAC_PUT_SUB	4	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Weighting factor of PUT_SUB for PUT calculation					
IP_PUT_I_1	6*6	0... FFFFH	0... 7293750	111.29549	[Pa/kg]
LDPM_PQ_TMP_1_INSY	6	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_MAF_MDL_DIF_1_INSY	6	0... FFFFH	-1024... 1023.96875	0.03125	[kg/h]
INSY controller I gain for pressure control (LV_MAF_CTL = 1)					
IP_PUT_I_2	6*6	0... FFFFH	0... 61.034223	931.323e-6	[1/s]
LDPM_PQ_TMP_1_INSY	6	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_MAP_MDL_DIF_1_INSY	6	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
INSY controller I gain for pressure control (LV_MAP_CTL = 1)					
IP_PUT_P_1	6*6	0... FFFFH	0... 42.51956	648.807e-6	[(hPa*h)/kg]
LDPM_PQ_TMP_1_INSY	6	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_MAF_MDL_DIF_1_INSY	6	0... FFFFH	-1024... 1023.96875	0.03125	[kg/h]
INSY controller P share for pressure control (LV_MAF_CTL = 1)					
IP_PUT_P_2	6*6	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
LDPM_PQ_TMP_1_INSY	6	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_MAP_MDL_DIF_1_INSY	6	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
INSY controller P share for pressure control (LV_MAP_CTL = 1)					
LC_AR_RED_PUT_P_INI	1	0... 1H	0... 1	1	[-]
Switch whether the P shares of the INSY controller are initialized (= 1) or ramped to 0 (= 0)					

General Information

The strategy of the closed loop control depends on the available sensors:

- If only a mass air flow sensor is available the loop will be controlled with MAF (LV_MAF_CTL = 1).

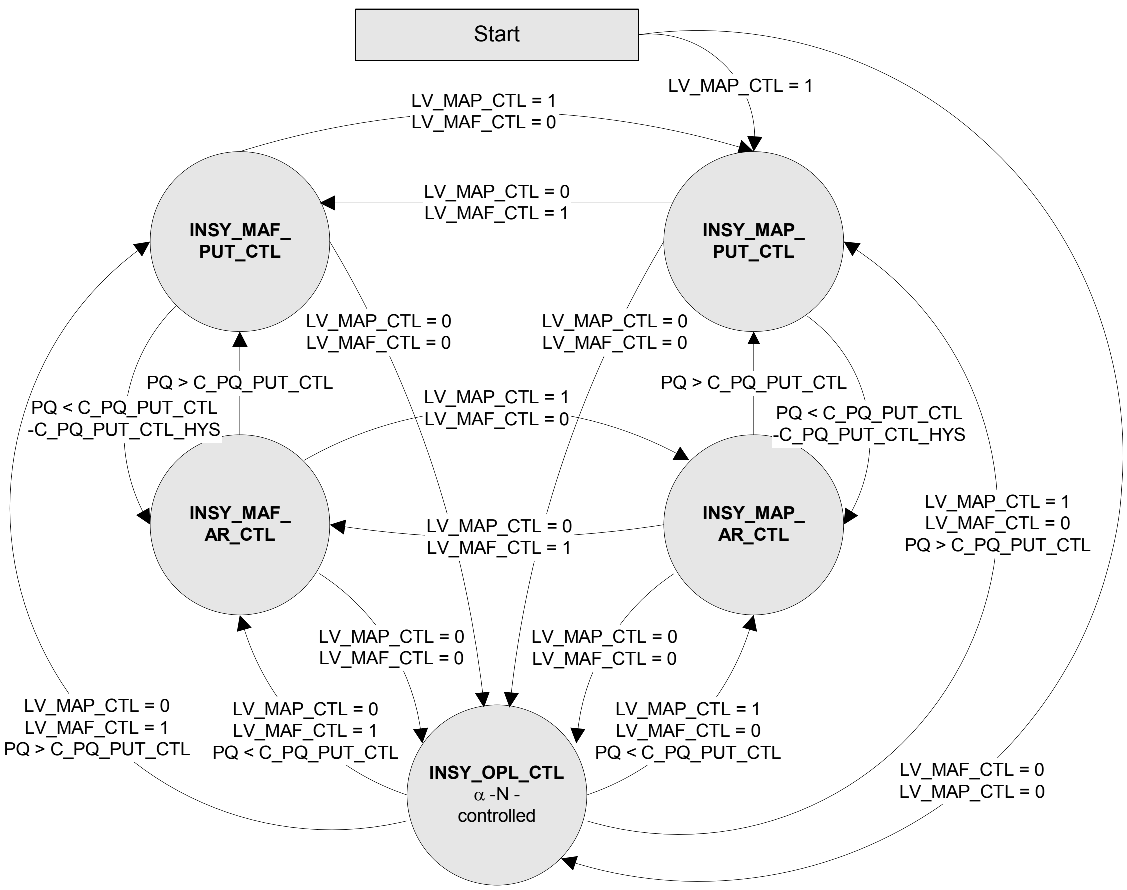
- If only a manifold pressure sensor is available the loop will be controlled with MAP (LV_MAP_CTL = 1).

- If both sensors are available the control depends on the pressure ratio at the throttle PQ. If PQ is smaller than the calibratable threshold C_PQ_MAF_CTL_MAX, considering a hysteresis C_PQ_MAF_CTL_MAX_HYS, the control will be done with the mass air flow. If the mass air flow sensor gives no reasonable signal (LV_MAF_CTL = 0) then the loop will be controlled with MAP (LV_MAP_CTL = 1). If PQ is greater than the calibratable threshold C_PQ_MAF_CTL_MAX the control will be done with MAP, except the MAP sensor gives no reasonable signal (LV_MAP_CTL = 0). Then the loop will be controlled with MAF (LV_MAF_CTL = 1).

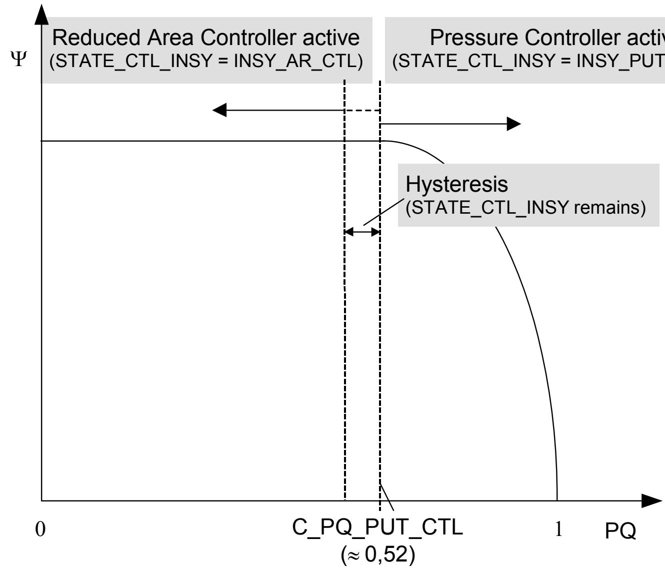
- If there is no reasonable signal for the closed loop control (LV_MAP_CTL = 0 and LV_MAF_CTL = 0) the system will be N controlled.

As long as the pressure ratio at the throttle d s not reach the calibratable threshold C_PQ_PUT_CTL, considering a hysteresis C_PQ_PUT_CTL_HYS, the deviations between intake manifold model values and measurement values are minimized with a PI controller adapting the reduced area (STATE_CTL_INSY = INSY_xx_AR_CTL). If the pressure ratio exceeds C_PQ_PUT_CTL the deviations are minimized with a PI controller adapting the modeled pressure upstream throttle PUT_MDL (STATE_CTL_INSY = INSY_xx_PUT_CTL).

In the state INSY_xx_AR_CTL the correction AR_RED_DIF_REL is calculated out of the model deviation MAF_MDL_DIF or MAP_MDL_DIF. If the manifold pressure sensor and the mass air flow sensor are defect the model g s into open loop INSY_CTL_OPL and AR_RED_DIF_x are identical to zero.



Picture 1: Control strategy, STATE_CTL_INSY transitions for different sensor configurations (Remark: This diagram is meant as an illustration only, in case of any doubt the written transition conditions are valid)



Picture 2: Working range of Reduced Area Controller and Pressure Controller

The reduced area controller can be stopped or reset to 0 by LV_AR_RED_CTL_STOP / RST. Correspondingly the controller for the pressure upstream the throttle can be stopped or reset to 0 by LV_PUT_CTL_STOP / RST.

4.14.1 Recurrence: 10ms:

Function description

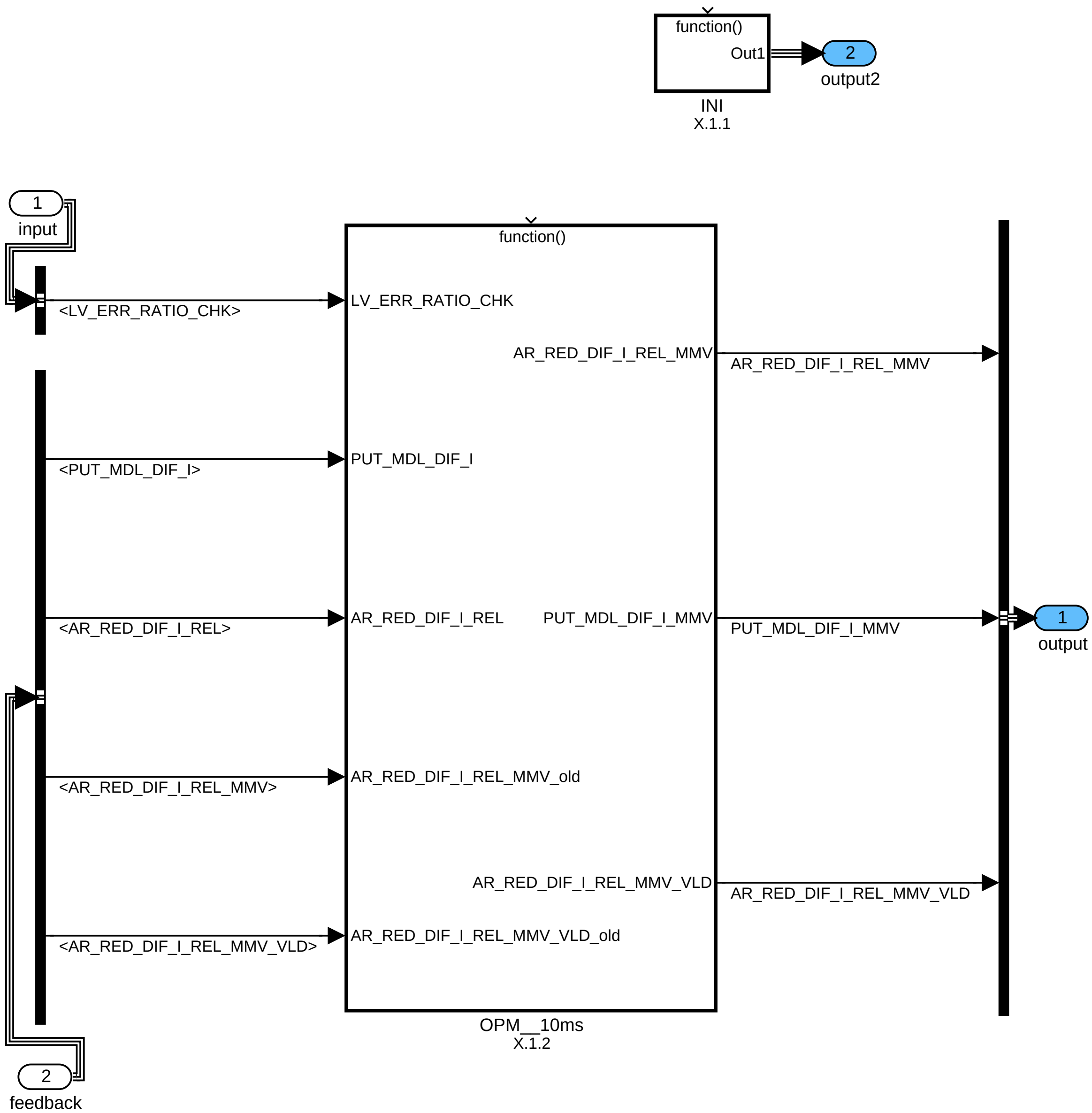


Figure 137:
Path: INSY_M404T/operate_10MS

4.14.1.1 Initialization (10ms - task):

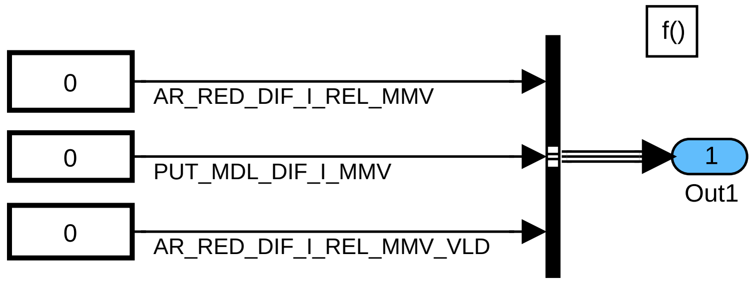
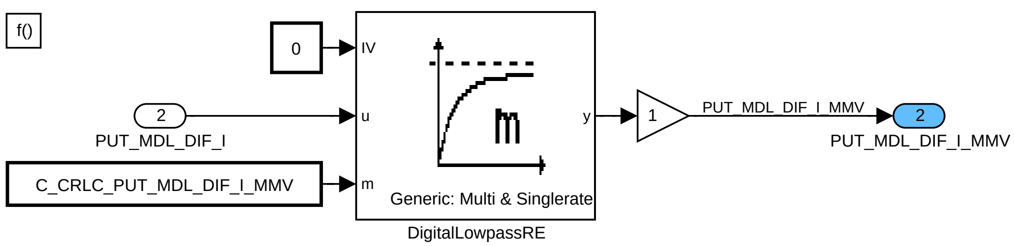


Figure 138:
Path: INSY_M404T/operate_10MS/INI

4.14.1.2 Calculation of AR_RED_DIF_I_REL_MMV:

If an error of MAP or TPS is present (LV_ERR_RATIO_CHK = 1) a moving average of the I-part is calculated in two different ways depending on the sign of AR_RED_DIF_I_REL. For the correction of steady state errors the integral part of the model pressure is delayed with a low pass filter.



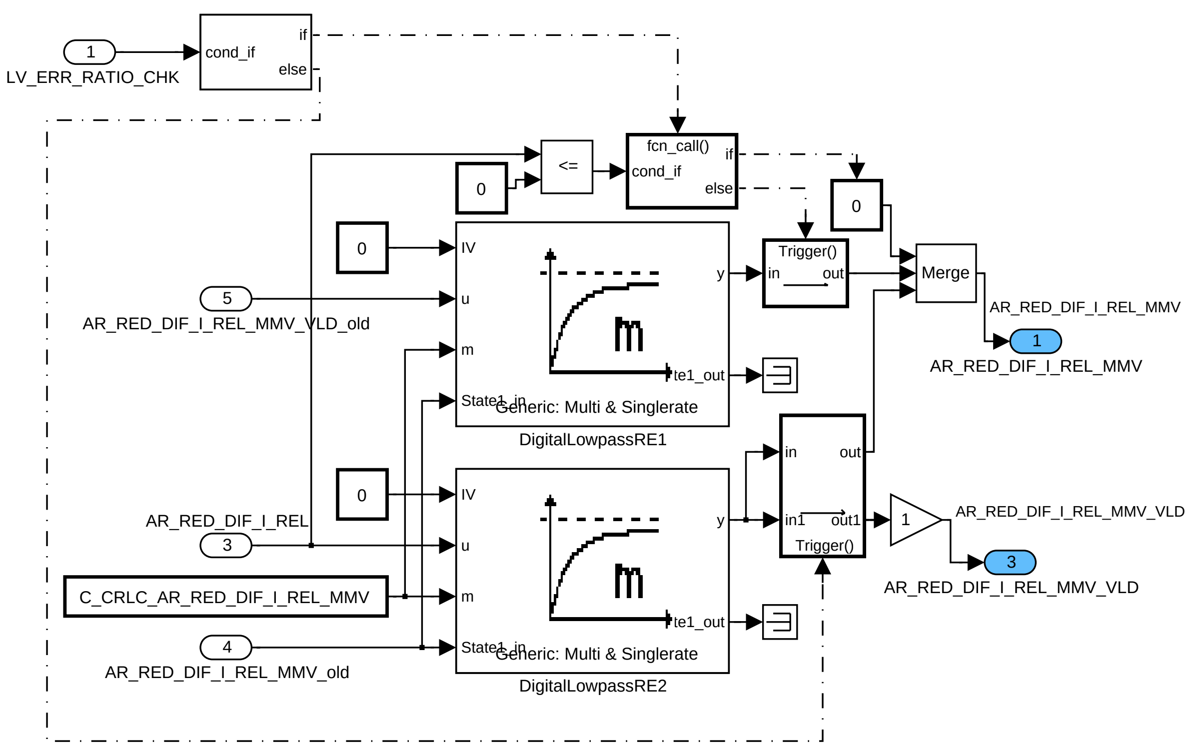


Figure 139:
Path: INSY_M404T/operate_10MS/OPM__10ms

4.14.2 Recurrence: Segment synchronous

The segment synchronous tasks are activated multisegment synchronously, i. e. every half segment, every segment or every other segment according to the Administration of calculation optimization.

Function description

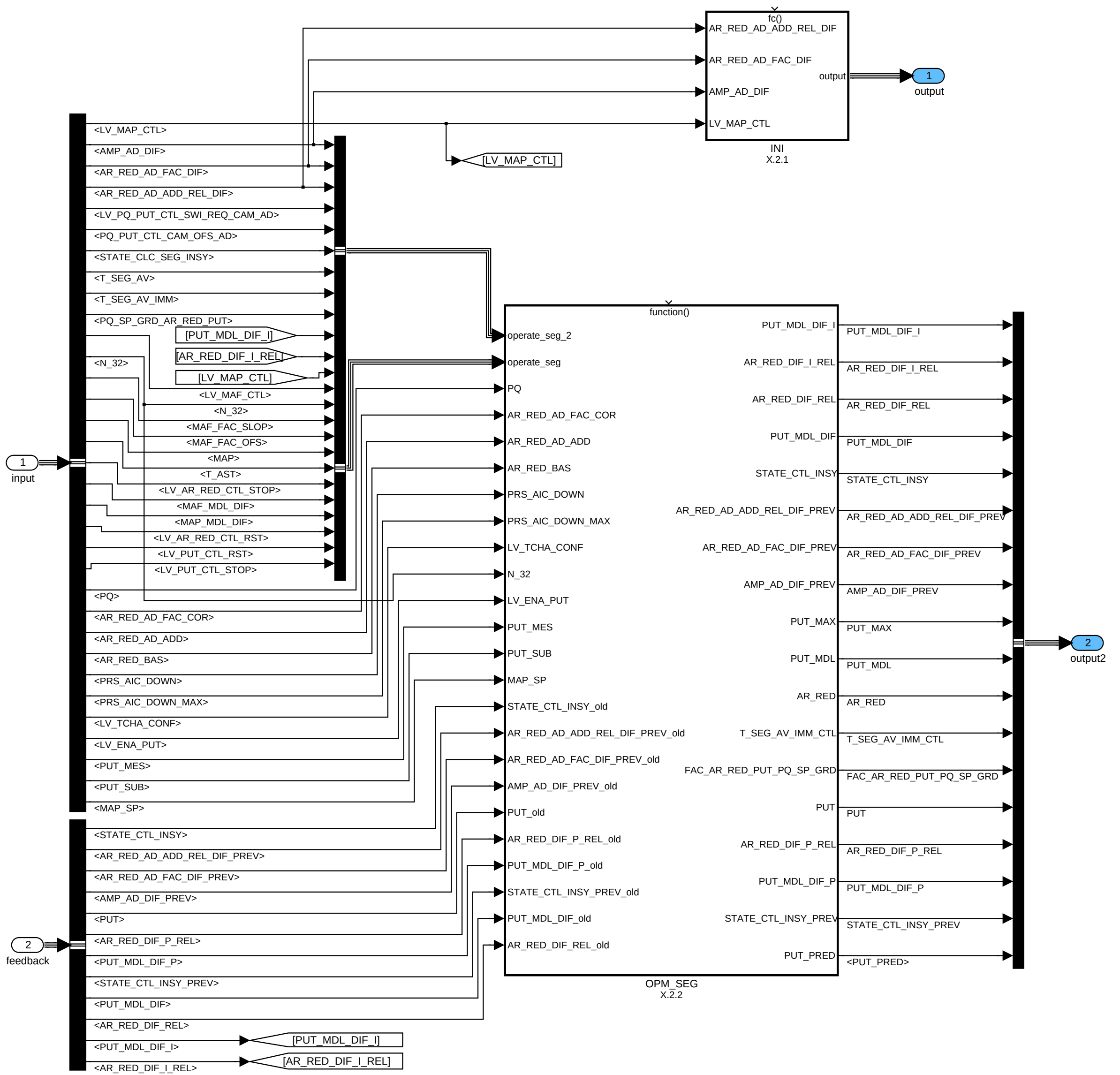


Figure 140:
Path: INSY_M404T/operate_SEG

4.14.2.1 Initialization (SEG - task):

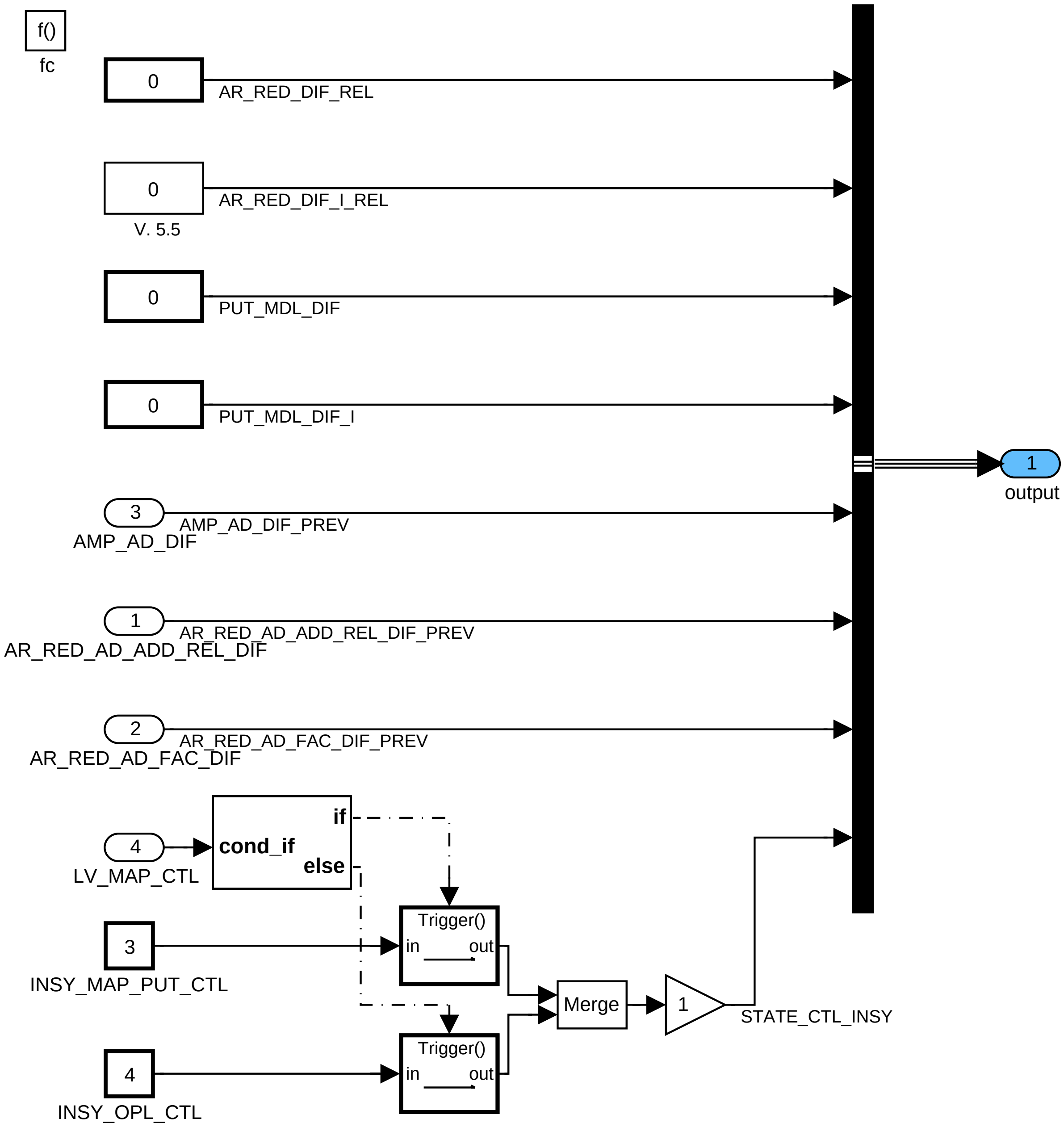


Figure 141:
Path: INSY_M404T/operate_SEG/INI

4.14.2.2 Overview over all possible states of intake manifold model adaptation:

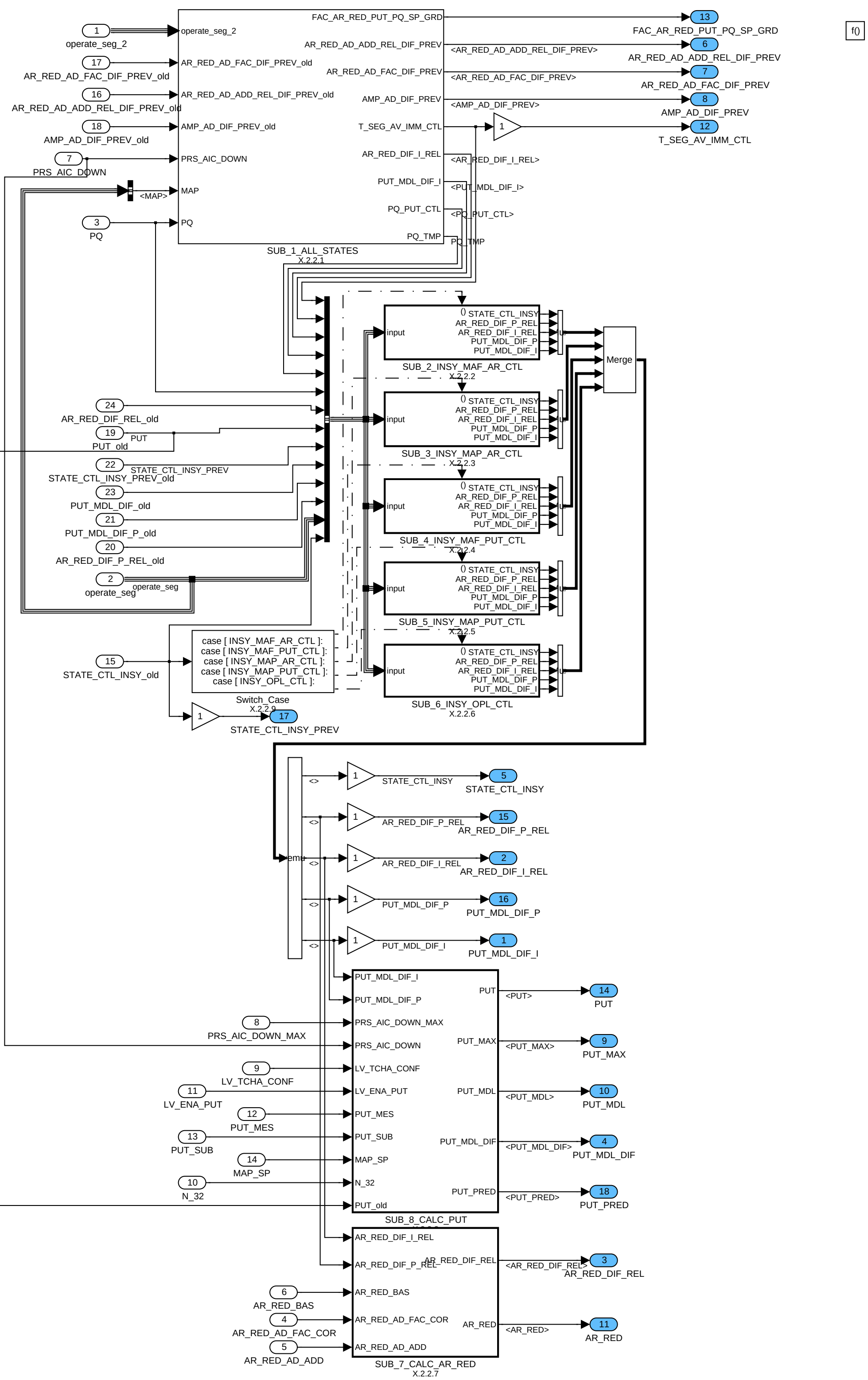


Figure 142:
Path: INSY_M404T/operate_SEG/OPM_SEG

4.14.2.2.1 SUB_1: Calculation of values which are used in all states (previous calculations):

When an reduced area adaptation has been performed, i.e. when a new AR_RED_AD_ADD or AR_RED_AD_FAC has been calculated by the Reduced Area Adaptation module, the reduced area controller output and its integral part are corrected with the learned adaptation values differences in order to get a continuous AR_RED. If an execution of the Reduced Area Adaptation module performs no adaptation, i.e. delivers no significant AR_RED_AD_ADD_REL_DIF or AR_RED_AD_FAC_DIF respectively, the Adaptation module sets AR_RED_AD_ADD_REL_DIF or AR_RED_AD_FAC_DIF to 0.
The variables with the ending PREV are defined to synchronize the different recurrences of the reduced area adaptation and the INSY-controller.

With the map IP_FAC_AR_RED_PUT_PQ_SP_GRD the I shares of the controller can be ramped down to zero depending on the rate of change of PQ_SP.

When an ambient pressure adaptation has been performed, i.e. when a new AMP_AD_DIF has been calculated by the Ambient Pressure Adaptation module, the pressure controller output and its integral part are corrected with the learned ambient pressure values in order to get a continuous PUT_MDL. If an execution of the Ambient Pressure Adaptation module performs no adaptation, i.e. delivers no significant AMP_AD_DIF, it sets AMP_AD_DIF = 0.

If a request from CAM_OFS_AD is valid, PQ_PUT_CTL has to be calculated in a different way.

The segment time for the INSY controller is chosen here, normal or double segment time are possible. As the INSY controller will not be calculated with half segment then the internal segment time has to be the normal segment time T_SEG_AV in case of INSY_SEG_HALF.

PQ_TMP is built for internal use.

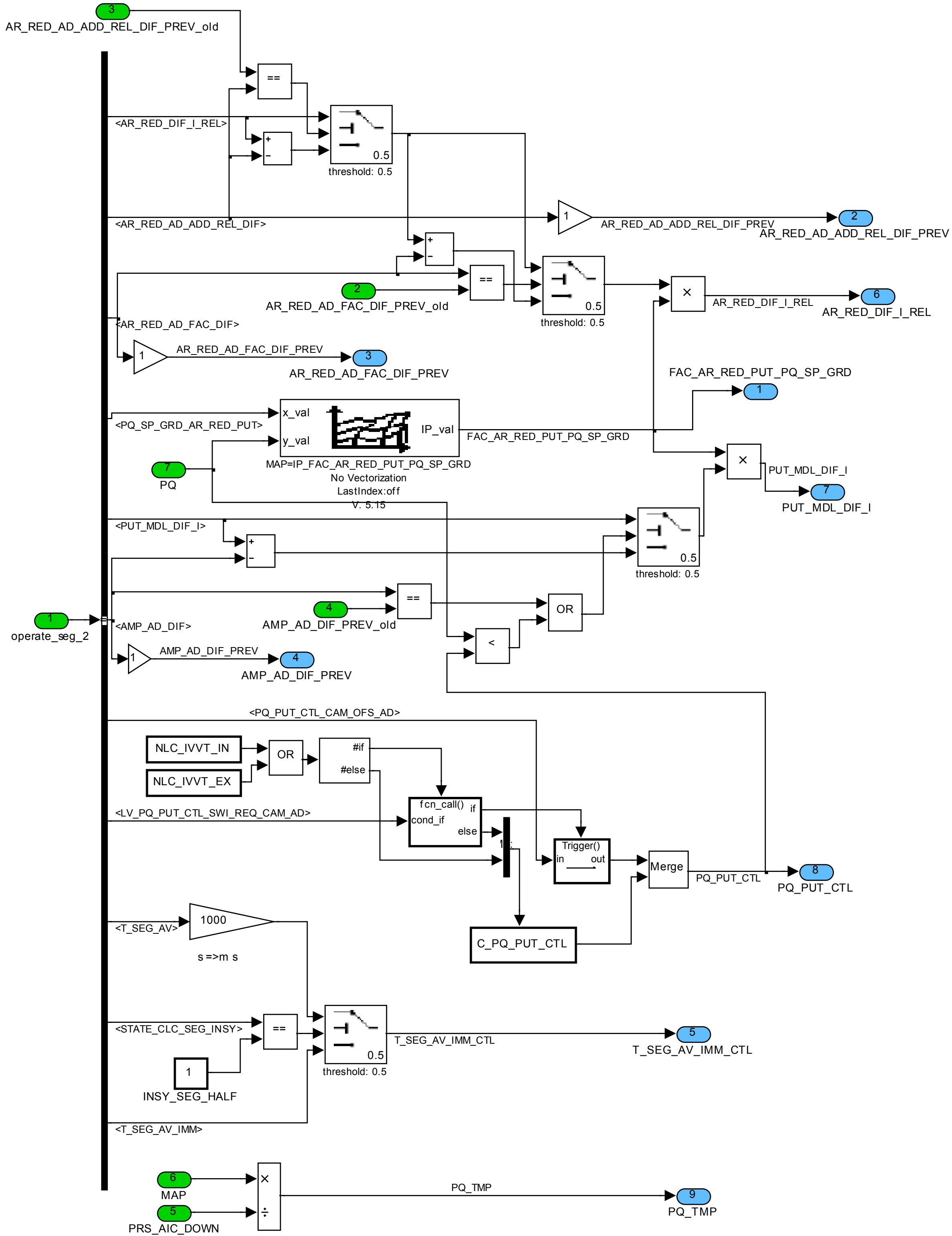


Figure 143:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_1_ALL_STATES

4.14.2.2.2 SUB_2: MAF sensor based intake manifold model adjustment via reduced area (INSY_MAF_AR_CTL):

4.14.2.2.2.1 SUB_21: MAF sensor based intake manifold model adjustment via reduced area (INSY_MAF_AR_CTL)

The Reduced Area controller works based on the air-mass flow deviation MAF_MDL_DIF. The Pressure Controller is switched off. PUT_MDL_DIF_I and PUT_MDL_DIF_P are not calculated as long as the Reduced Area controller is active. PUT_MDL_DIF_P is ramped down to 0

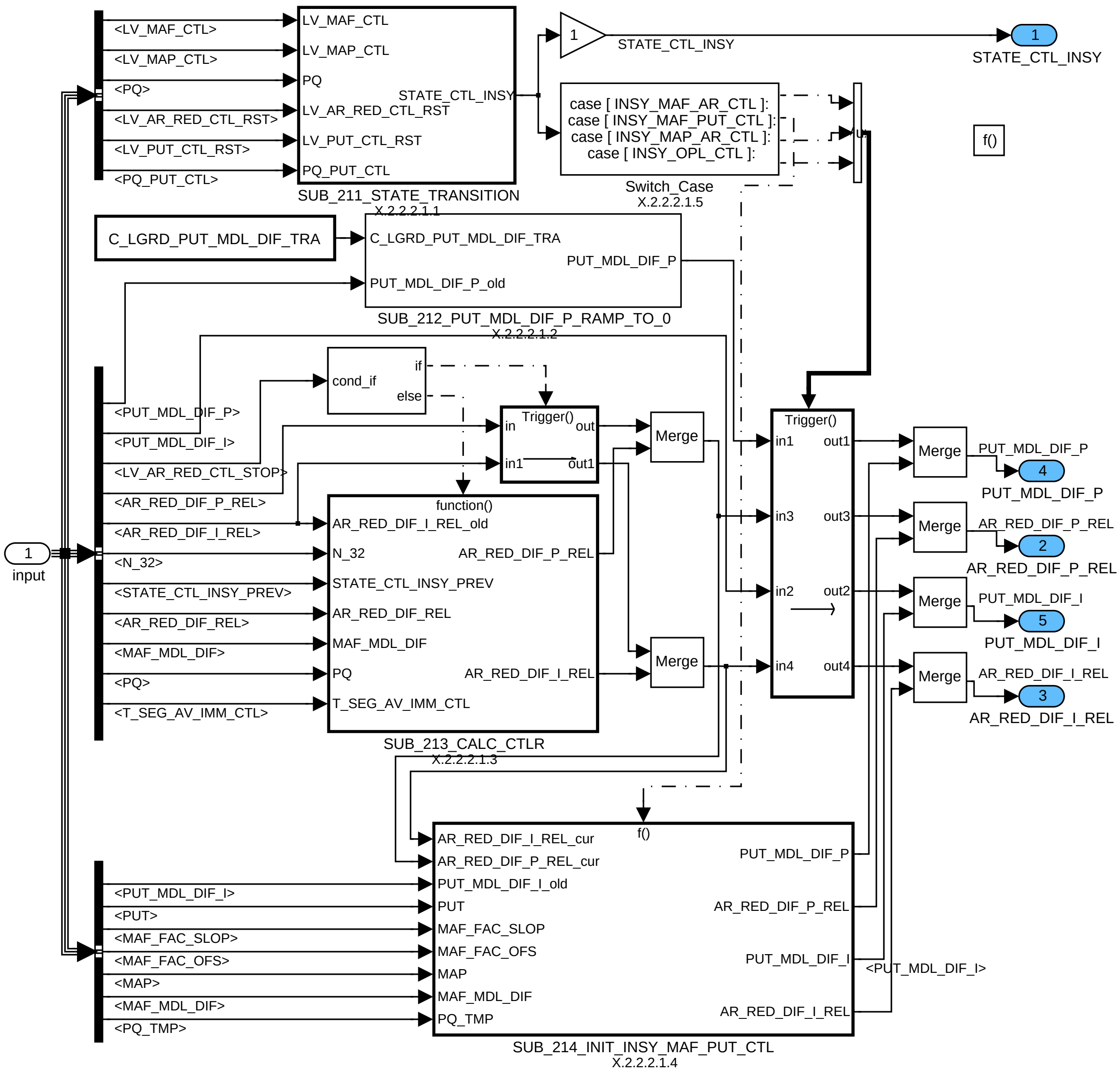


Figure 144:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL

4.14.2.2.2.1.1 SUB_211: Check if a state transition is requested:

The INSY controller can be deactivated (switched to open loop) by the logical variables LV_AR_RED_CTL_RST and LV_PUT_CTL_RST.

From INSY_MAF_AR_CTL a switch over to INSY_OPL_CTL, INSY_MAF_PUT_CTL and INSY_MAP_AR_CTL is possible. If a switch to INSY_MAP_PUT_CTL is required, which can happen if C_PQ_PUT_CTL is close to C_PQ_MAF_CTL_MAX, the system is switched over to INSY_MAF_PUT_CTL first. The switch to INSY_MAP_PUT_CTL is performed at the next execution cycle of the INSY controller. To avoid an unsteady controller output signal when switching over to the pressure controller, its integral part is accordingly initialized.

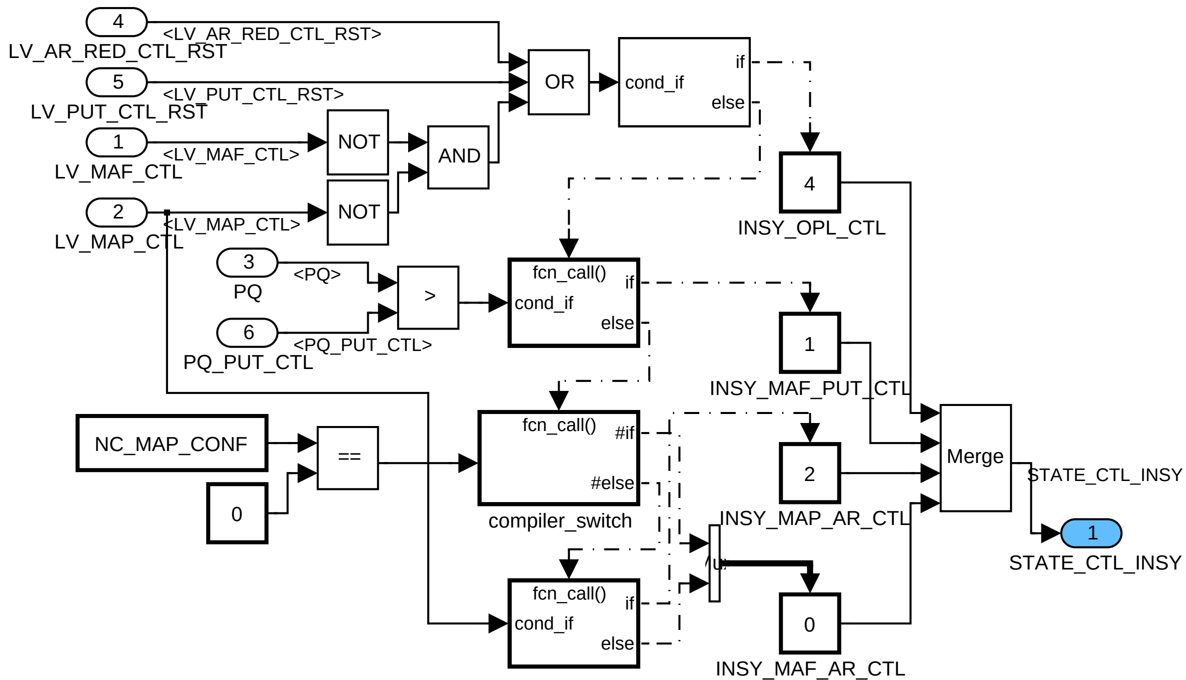


Figure 145:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_211_STATE_TRANSITION

4.14.2.2.2.1.2 SUB_212: PUT_MDL_DIF_P is ramped down to 0:

PUT_MDL_DIF_P is ramped down to 0 with a gradient of C_LGRD_PUT_MDL_DIF_TRA beginning with the STATE_CTL_INSY transition.

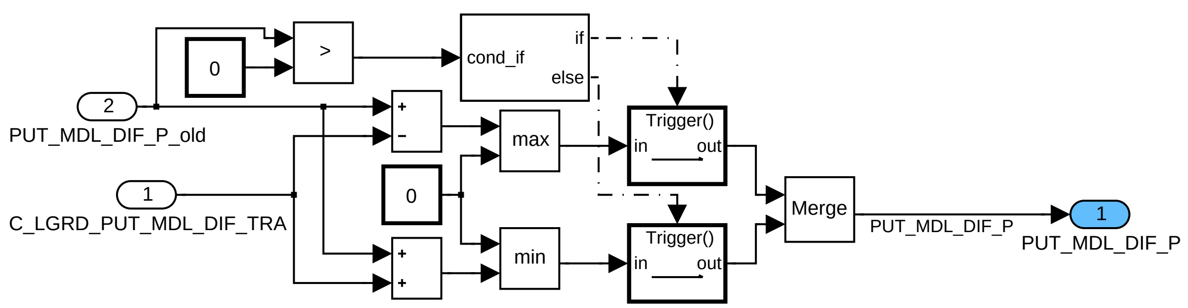


Figure 146:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_212_PUT_MDL_DIF_P_RAMP_TO_0

4.14.2.2.2.1.3 SUB_213: Calculations of controller outputs (PI):

Controller initialization for smooth transition from INSY_MAP_AR_CTL if STATE_CTL_INSY_old = 2 or normal controller calculation.

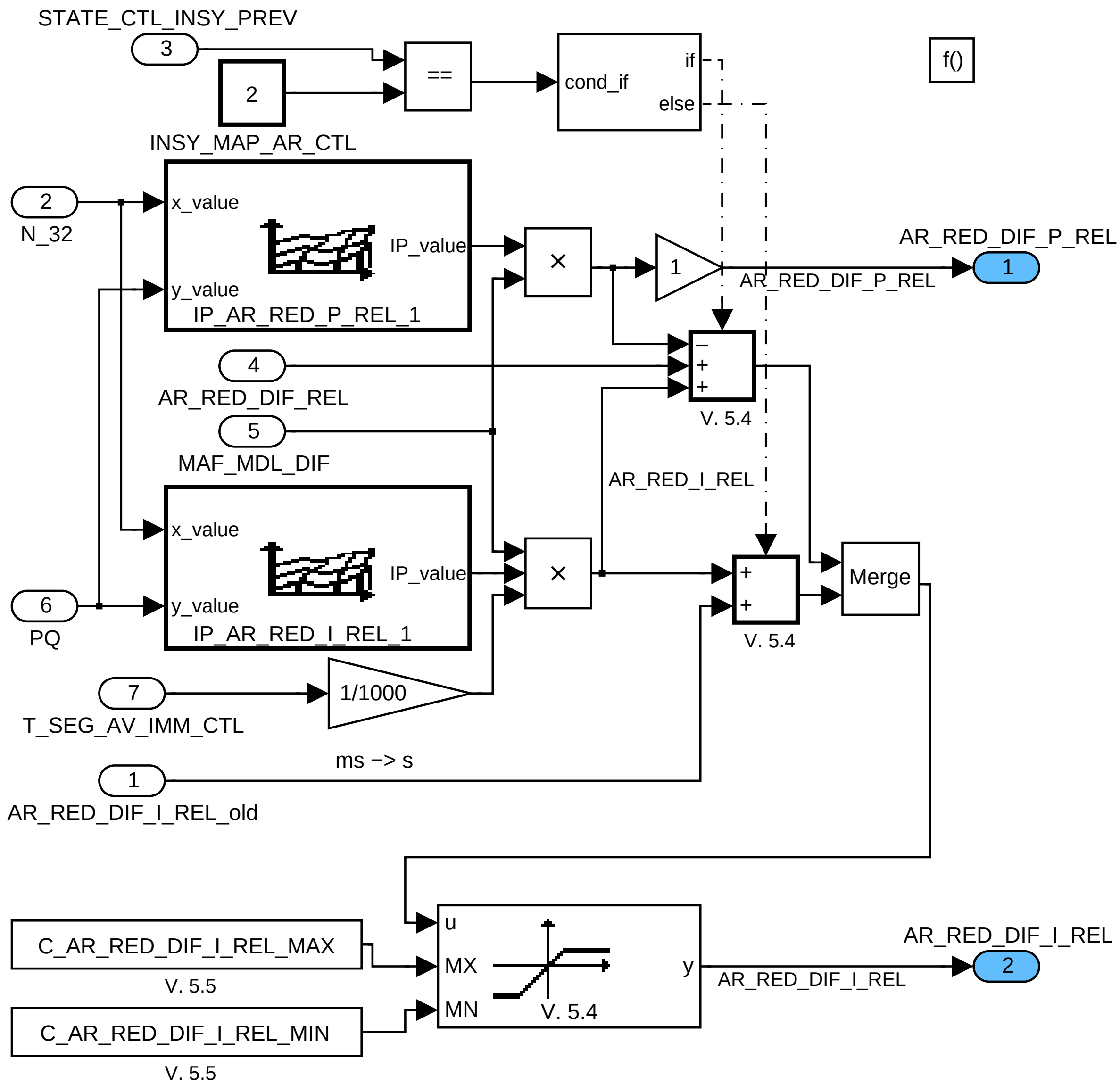


Figure 147:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_213_CALC_CTLR

4.14.2.2.2.1.4 SUB_214: INSY_MAF_AR_CTL:Initialization of controller outputs at transition to PUT-controlled (INSY_MAF_PUT_CTL):

At transition between AR_RED and PUT controlled states, two initialization modes are possible depending of LC_AR_RED_PUT_P_INI.

Controller initialization for smooth transition to INSY_MAF_PUT_CTL: PUT_MDL_DIF_I can have been changed since the last activation of the pressure controller via ambient pressure adaptation, so PUT_MDL_DIF_I_old is not necessarily 0.

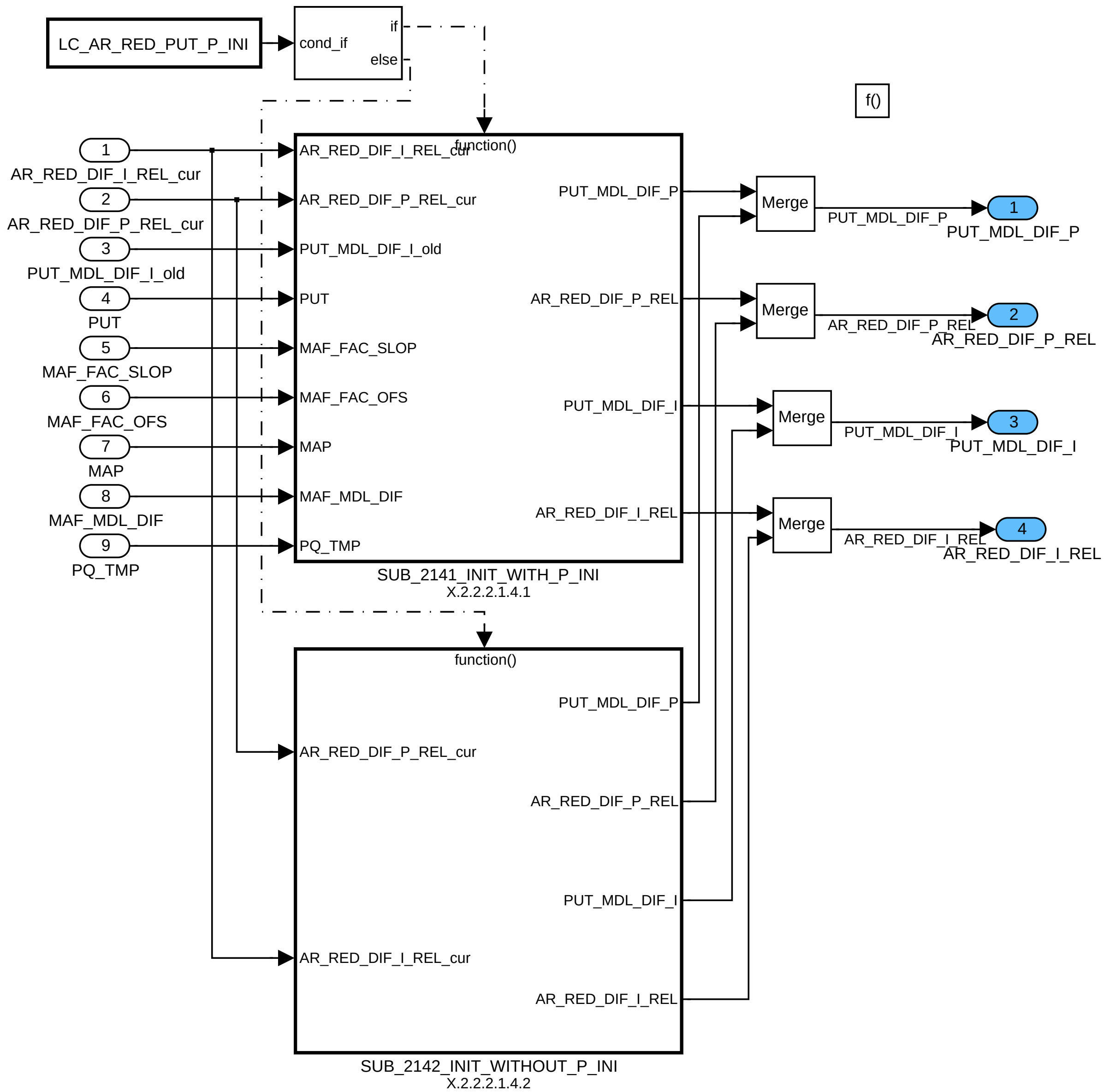


Figure 148:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_214_INIT_INSY_MAF_PUT_CTL

4.14.2.2.2.1.4.1 SUB_2141: Initialization with the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 1):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is also initialized. The P share of the old state is set to 0.

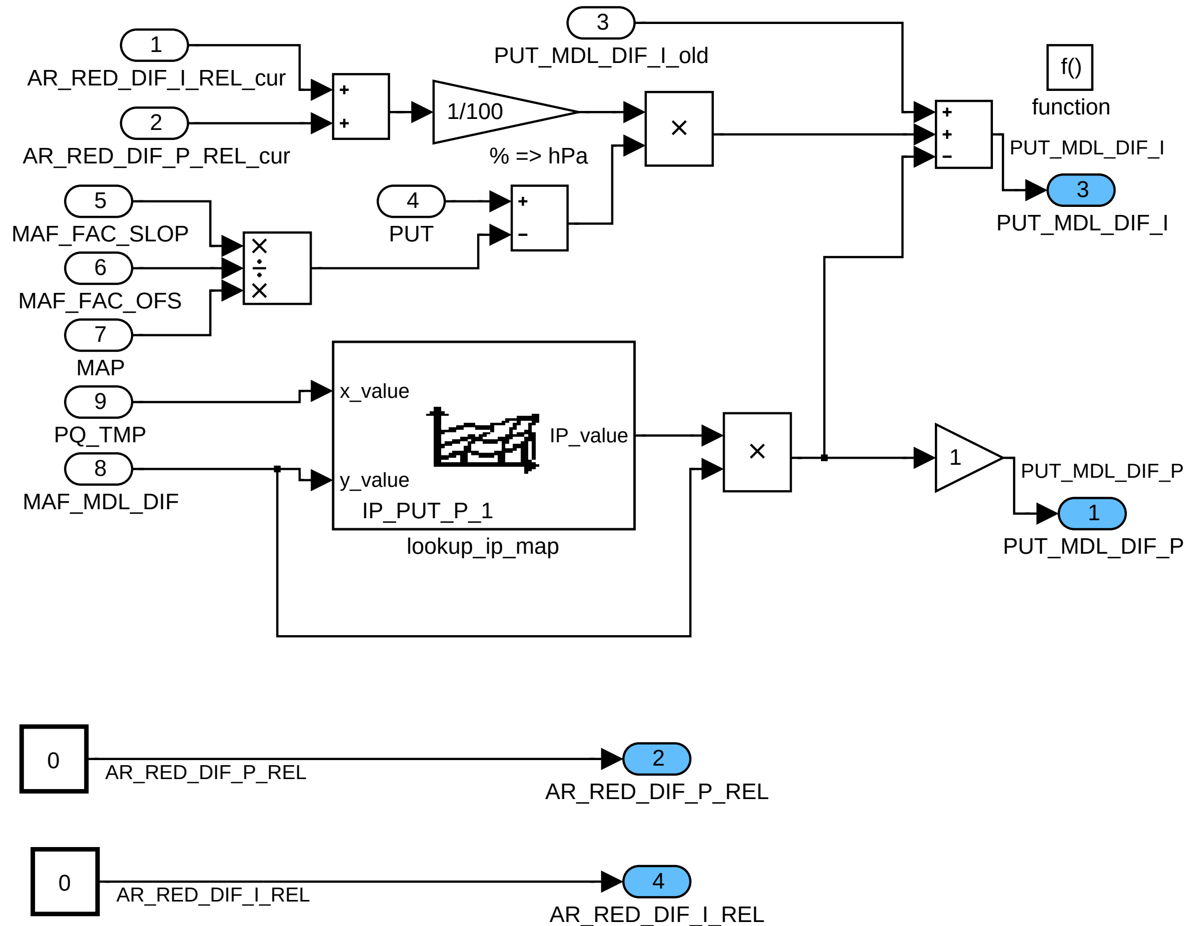


Figure 149:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_214_INIT_INSY_MAF_PUT_CTL/SUB_2141_INIT_WITH_P_INI

4.14.2.2.2.1.4.2 SUB_2142: Initialization without the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 0):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is not initialized. The P share of the old state is ramped down to 0.

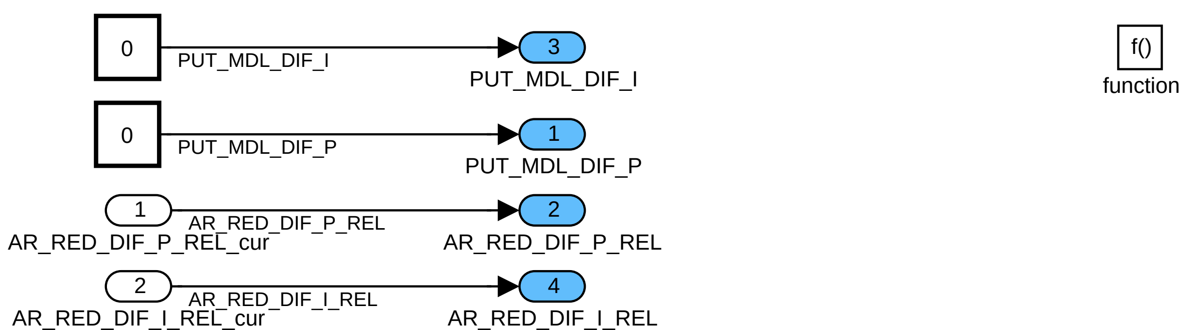


Figure 150:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_2_INSY_MAF_AR_CTL/SUB_21_INSY_MAF_AR_CTL/SUB_214_INIT_INSY_MAF_PUT_CTL/SUB_2142_INIT_WITHOUT_P_INI

4.14.2.2.3 SUB_3: MAP sensor based intake manifold model adjustment via reduced area (INSY_MAP_AR_CTL):

4.14.2.2.3.1 SUB_31: MAP sensor based intake manifold model adjustment via reduced area (INSY_MAP_AR_CTL)

The Reduced Area controller works based on the manifold air pressure deviation MAP_MDL_DIF. The Pressure Controller is switched off. PUT_MDL_DIF_I and PUT_MDL_DIF_P are not calculated as long as the Reduced Area controller is active. PUT_MDL_DIF_P is ramped down to 0.

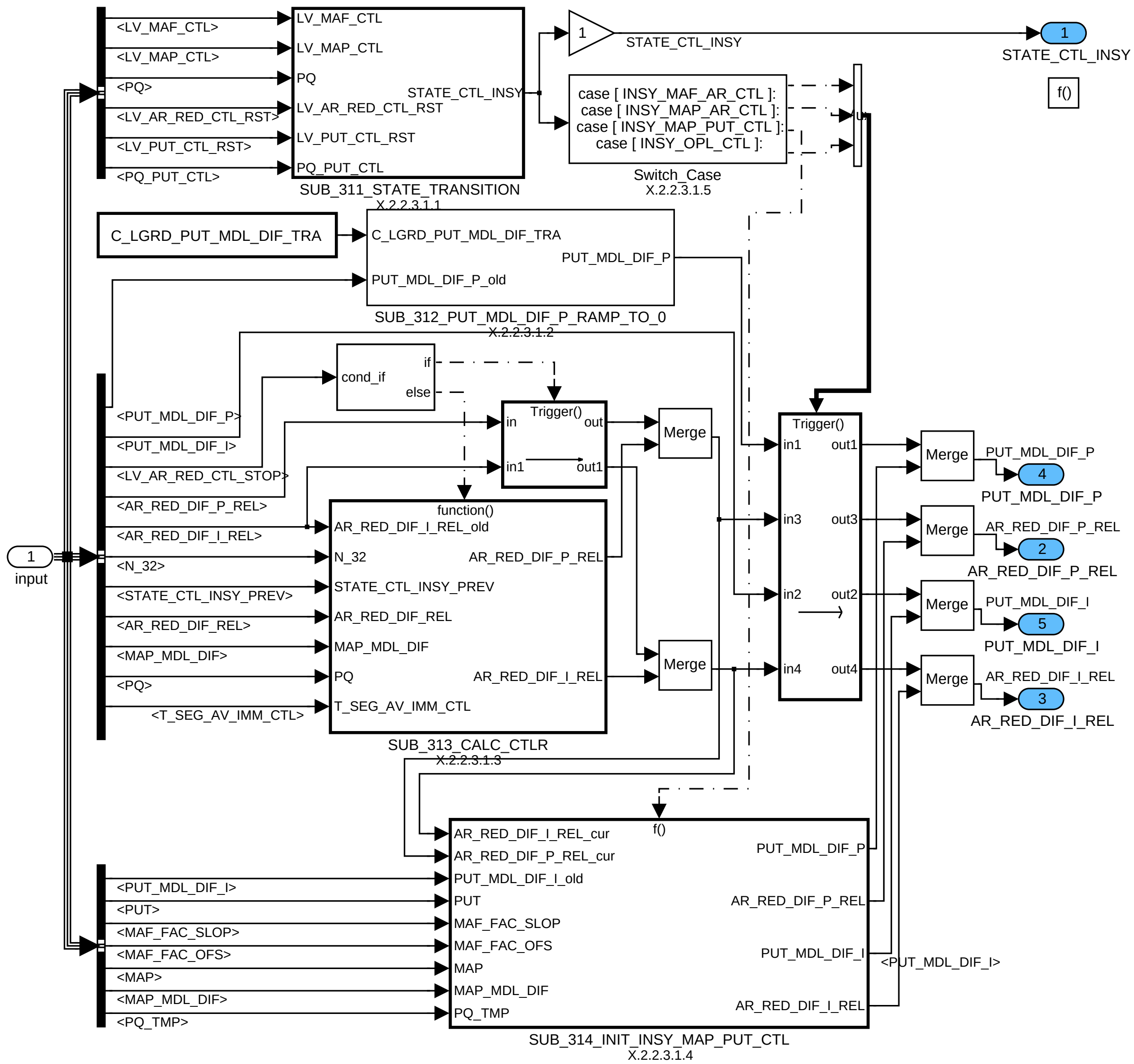


Figure 151:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL

4.14.2.2.3.1.1 SUB_311: Check if a state transition is requested:

The INSY controller can be deactivated (switched to open loop) by the logical variables LV_AR_RED_CTL_RST and LV_PUT_CTL_RST.

From INSY_MAP_AR_CTL a switch over to INSY_OPL_CTL, INSY_MAF_AR_CTL and INSY_MAP_PUT_CTL is possible. If a switch to INSY_MAF_PUT_CTL is required, which can happen if C_PQ_PUT_CTL is close to C_PQ_MAF_CTL_MAX, the system is switched over to INSY_MAP_PUT_CTL first. The switch to INSY_MAF_PUT_CTL is performed at the next execution cycle of the INSY controller. To avoid an unsteady controller output signal when switching over to the pressure controller, its integral part is accordingly initialized.

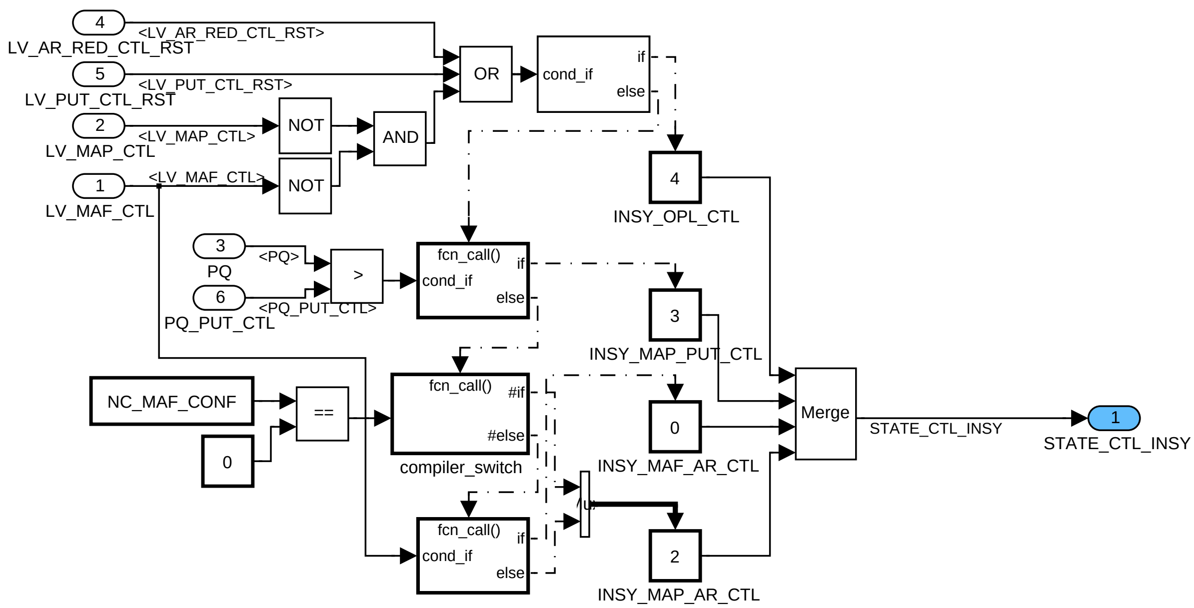


Figure 152:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_311_STATE_TRANSITION

4.14.2.2.3.1.2 SUB_312: PUT_MDL_DIF_P is ramped down to 0:

PUT_MDL_DIF_P is ramped down to 0 with a gradient of C_LGRD_PUT_MDL_DIF_TRA beginning with the STATE_CTL_INSY transition

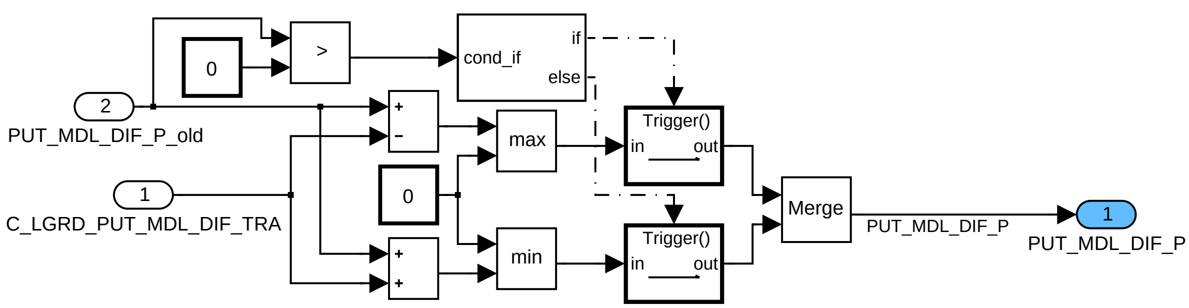


Figure 153:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_312_PUT_MDL_DIF_P_RAMP_TO_0

4.14.2.2.3.1.3 SUB_313: Calculations of controller outputs (PI):

Controller initialization for smooth transition from INSY_MAF_AR_CTL if STATE_CTL_INSY_old = 0 or normal controller calculation.

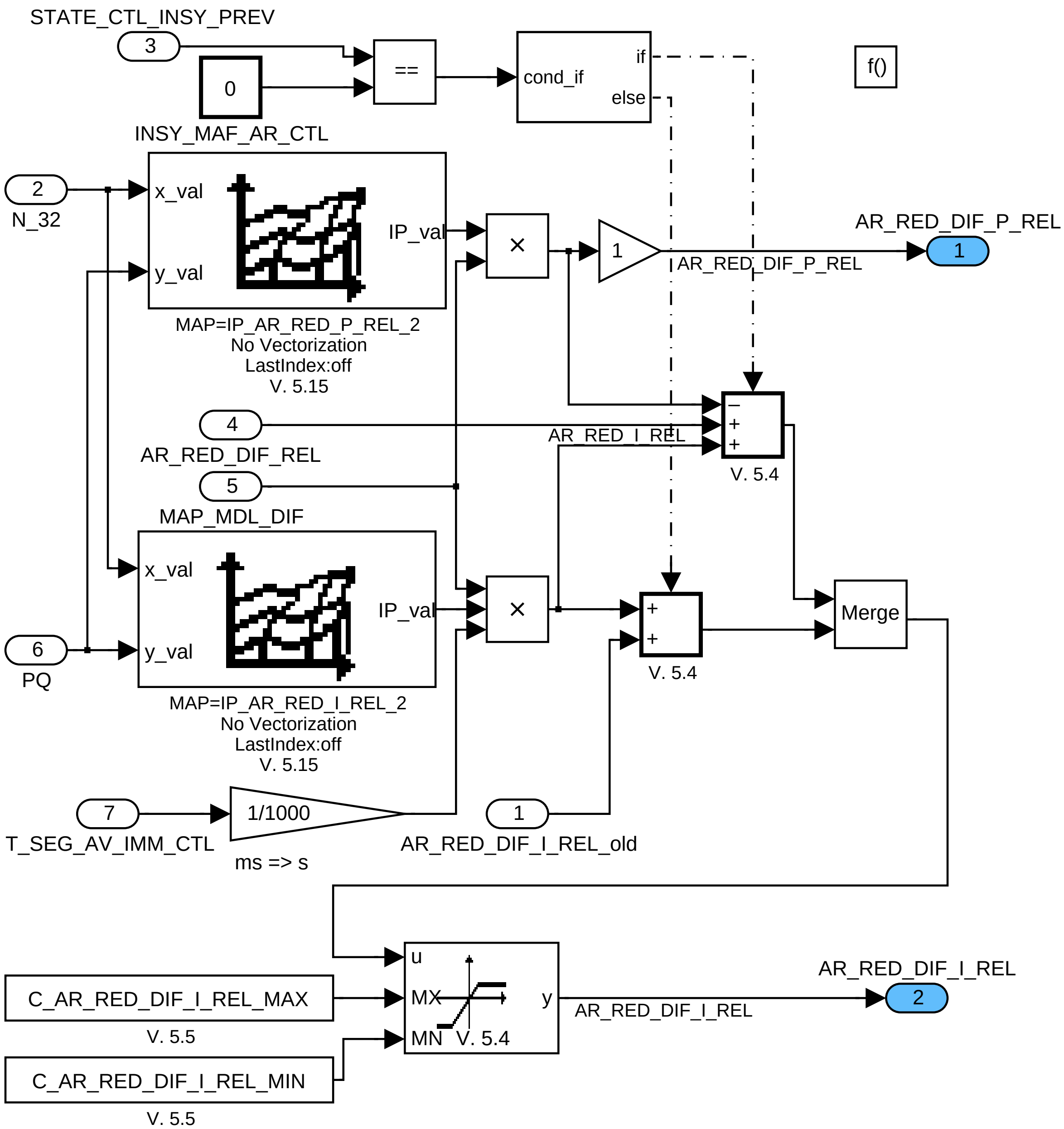


Figure 154:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_313_CALC_CTLR

4.14.2.2.3.1.4 SUB_314: INSY_MAP_AR_CTL:Initialization of controller outputs at transition to PUT-controlled (INSY_MAP_PUT_CTL):

At transition between AR_RED and PUT controlled states, two initialization modes are possible depending of LC_AR_RED_PUT_P_INI.

Controller initialization for smooth transition to INSY_MAP_PUT_CTL: PUT_MDL_DIF_I can have been changed since the last activation of the pressure controller via ambient pressure adaptation, so PUT_MDL_DIF_I_old is not necessarily 0.

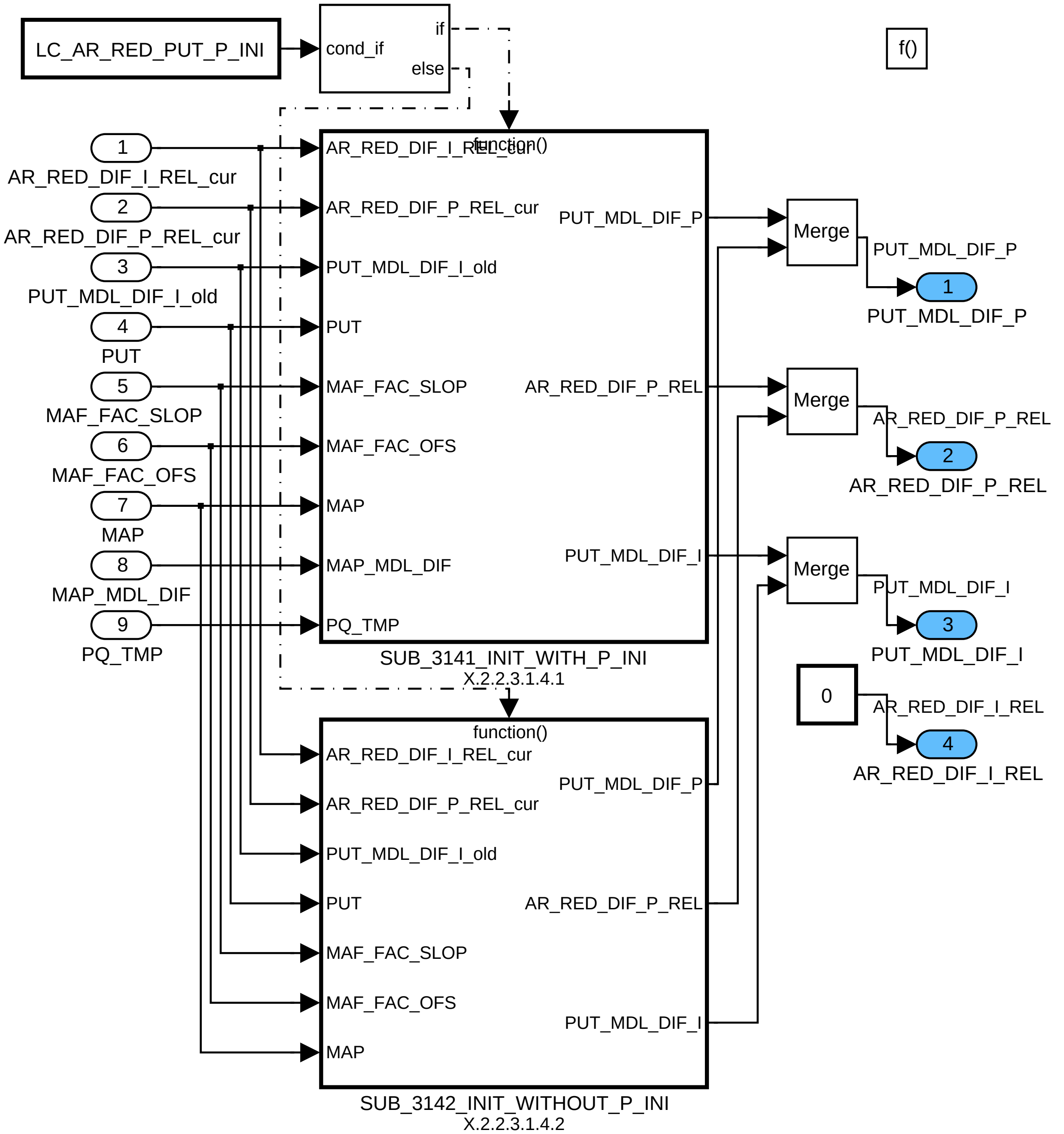


Figure 155:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_314_INIT_INSY_MAP_PUT_CTL

4.14.2.2.3.1.4.1 SUB_3141: Initialization with the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 1):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is also initialized. The P share of the old state is set to 0.

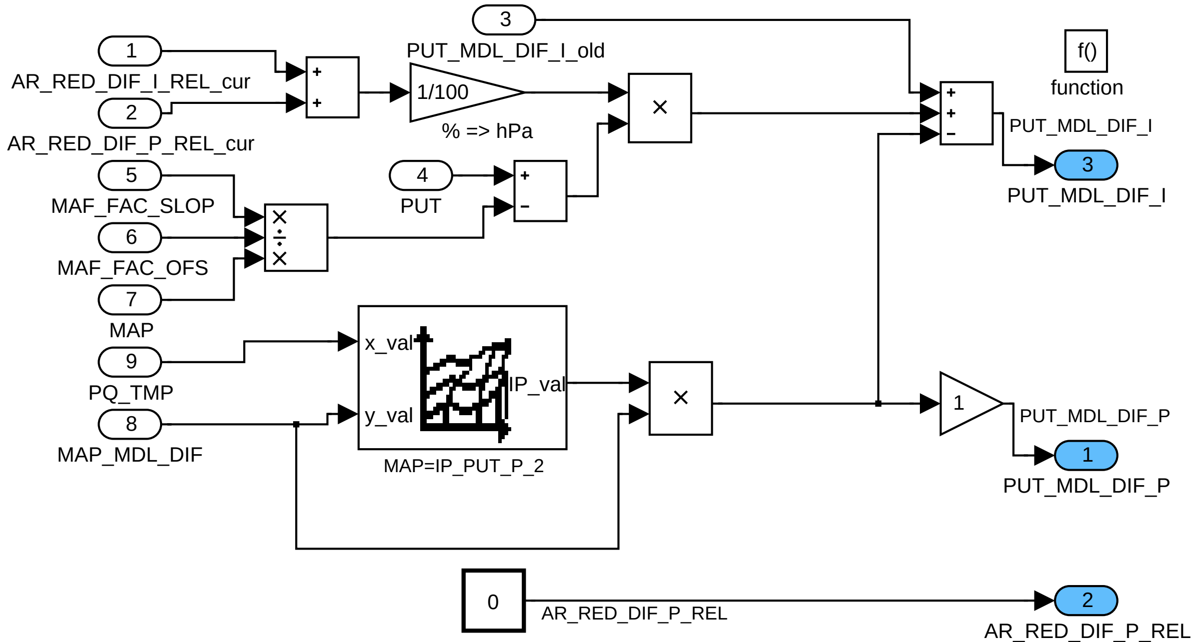


Figure 156:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_314_INIT_INSY_MAP_PUT_CTL/SUB_3141_INIT_WITH_P_INI

4.14.2.2.3.1.4.2 SUB_3142: Initialization without the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 0):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is not initialized. The P share of the old state is ramped down to 0.

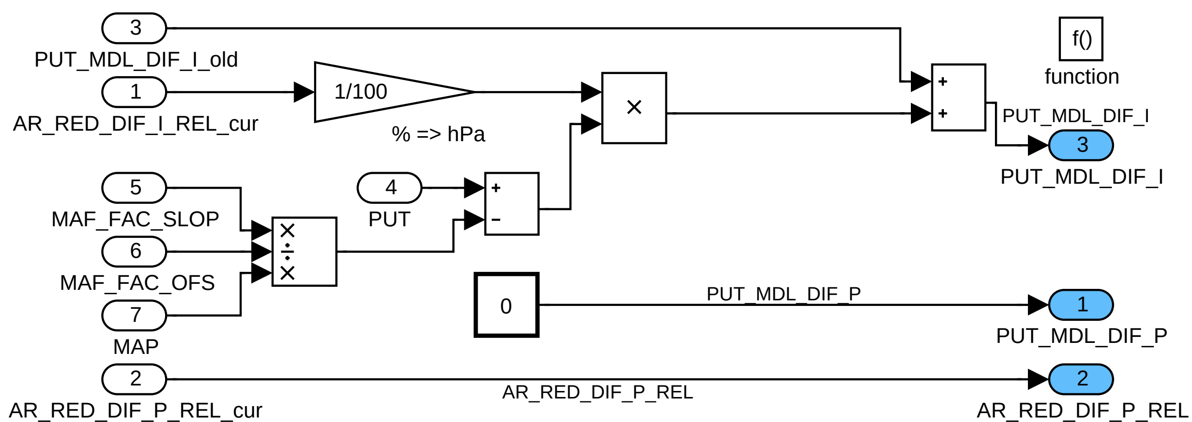


Figure 157:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_3_INSY_MAP_AR_CTL/SUB_31_INSY_MAP_AR_CTL/SUB_314_INIT_INSY_MAP_PUT_CTL/SUB_3142_INIT_WITHOUT_P_INI

4.14.2.2.4 SUB_4: MAF sensor based intake manifold model adjustment via pressure upstream throttle (INSY_MAF_PUT_CTL):

4.14.2.2.4.1 SUB_41: MAF sensor based intake manifold model adjustment via pressure upstream throttle (INSY_MAF_PUT_CTL)

The Pressure Controller works based on the air-mass flow deviation MAF_MDL_DIF. The Reduced Area Controller is switched off. AR_RED_DIF_P_REL and AR_RED_DIF_I_REL are not calculated as long as the pressure controller is active. AR_RED_DIF_P_REL and AR_RED_DIF_I_REL are ramped down to 0.

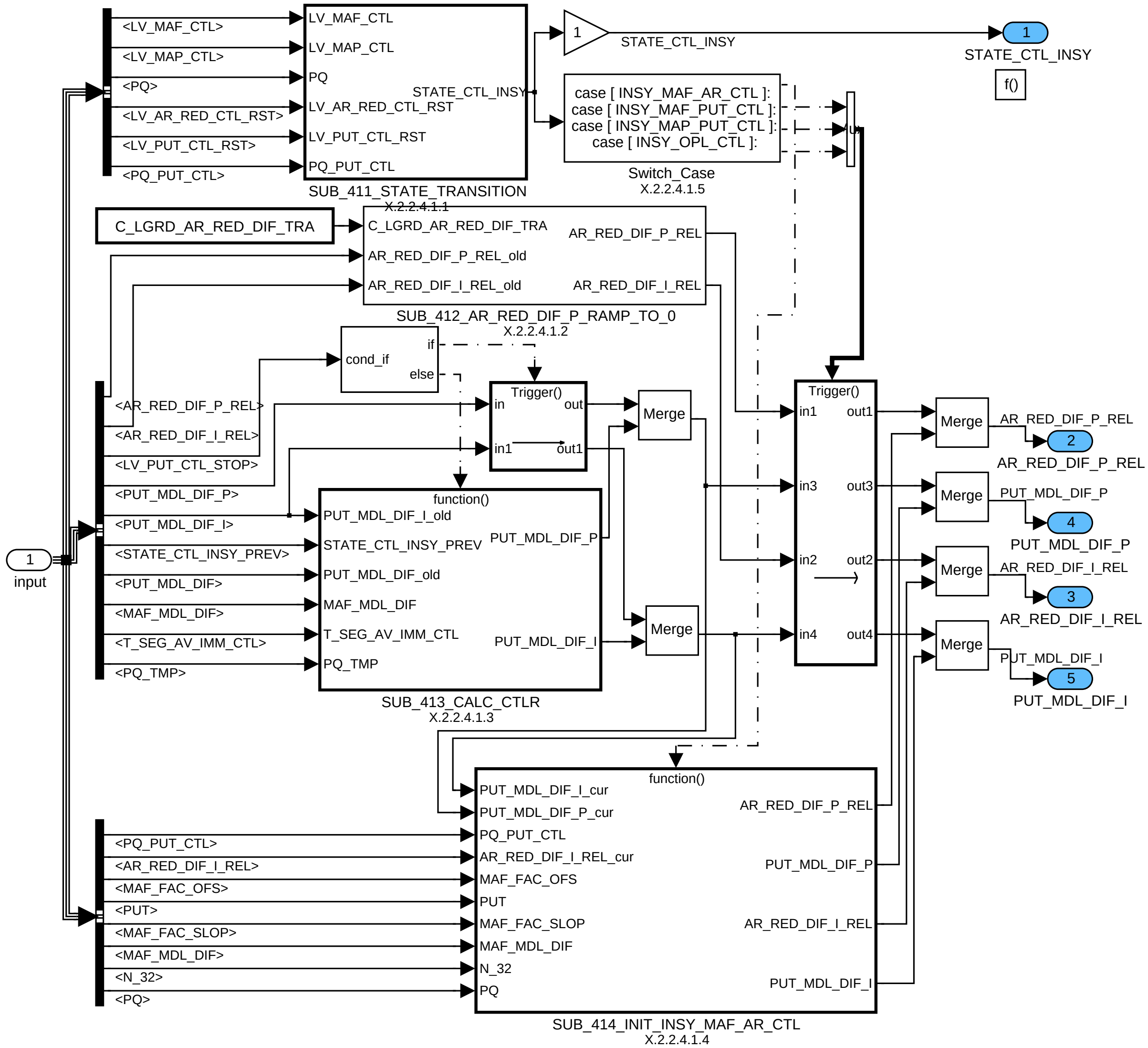


Figure 158:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL

4.14.2.2.4.1.1 SUB_411: Check if a state transition is requested:

The INSY controller can be deactivated (switched to open loop) by the logical variables LV_AR_RED_CTL_RST and LV_PUT_CTL_RST.

From INSY_MAF_PUT_CTL a switch over to INSY_OPL_CTL, INSY_MAF_AR_CTL and INSY_MAP_PUT_CTL is possible. If a switch to INSY_MAP_AR_CTL is required, which can happen if C_PQ_PUT_CTL is close to C_PQ_MAF_CTL_MAX, the system is switched over to INSY_MAF_AR_CTL first. The switch to INSY_MAP_AR_CTL is performed at the next execution cycle of the INSY controller. To avoid an unsteady controller output signal when switching over to the Reduced Area controller, its integral part is accordingly initialized.

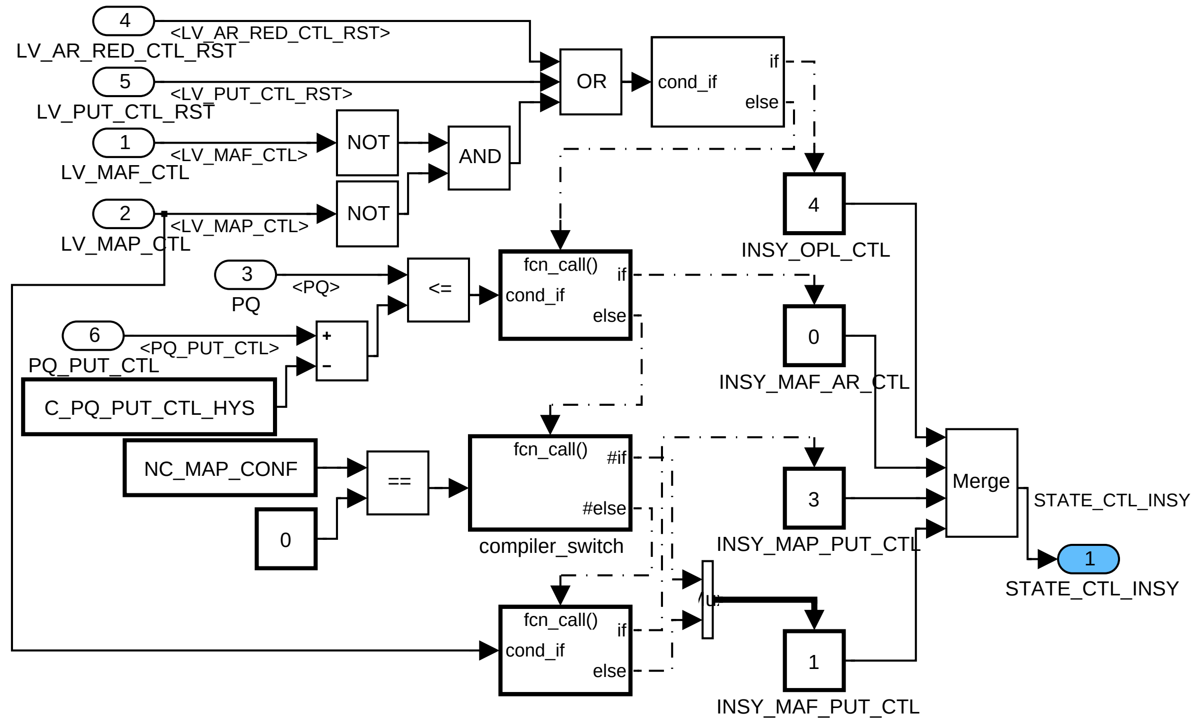


Figure 159:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_411_STATE_TRANSITIO
N

4.14.2.2.4.1.2 SUB_412: AR_RED_DIF_P_REL is ramped down to 0:

AR_RED_DIF_P_REL and AR_RED_DIF_I_REL are ramped down to 0 with a gradient of C_LGRD_AR_RED_DIF_TRA beginning with the STATE_CTL_INSY transition.

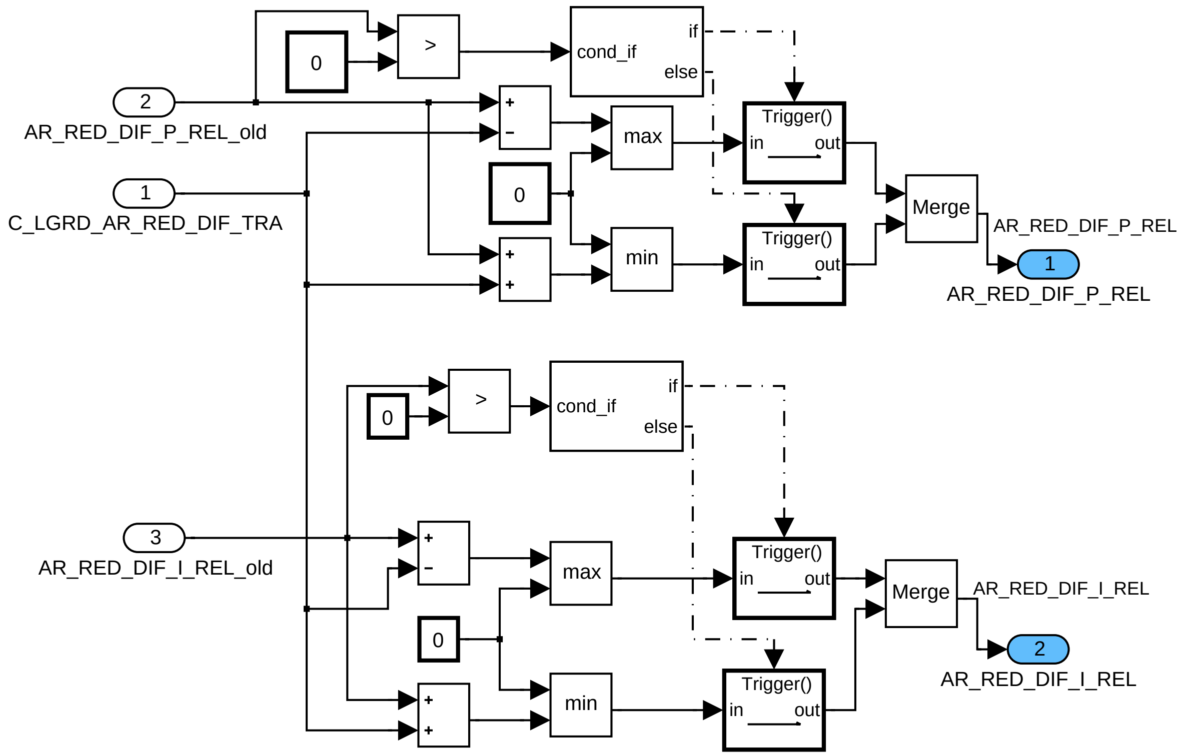


Figure 160:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_412_AR_RED_DIF_P_RAMP_TO_0

4.14.2.2.4.1.3 SUB_413: Calculations of controller outputs (PI):

Controller initialization for smooth transition from INSY_MAP_PUT_CTL if STATE_CTL_INSY_old = 3 or normal controller calculation otherwise.

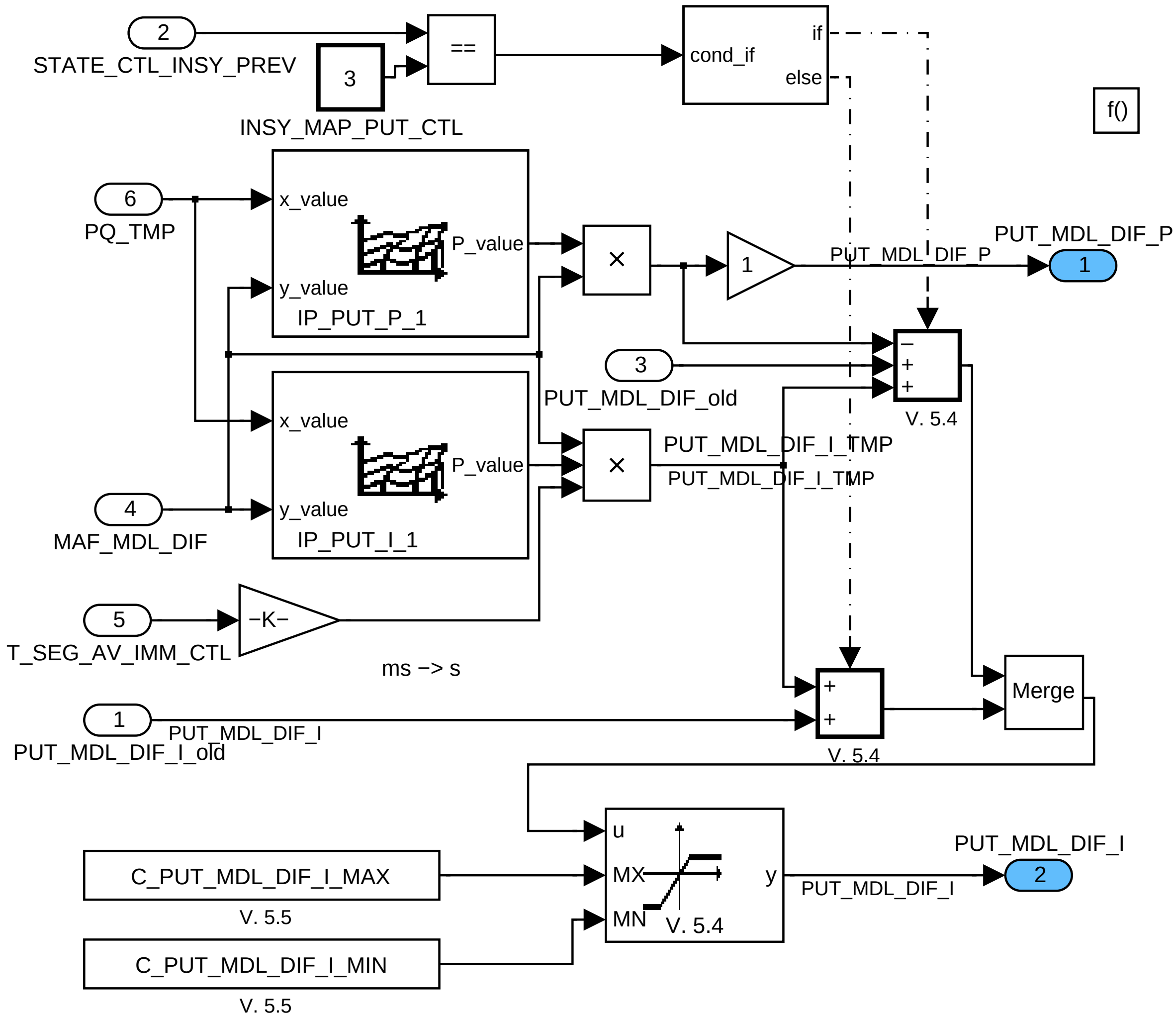


Figure 161:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_413_CALC_CTLR

4.14.2.2.4.1.4 SUB_414: INSY_MAF_PUT_CTL:Initialization of controller outputs at transition to reduced throttle area controlled (INSY_MAF_AR_CTL):

At transition between PUT and AR_RED controlled states, two initialization modes are possible depending of LC_AR_RED_PUT_P_INI.
Controller initialization for smooth transition to INSY_MAF_AR_CTL.
AR_RED_DIF_I_REL can have been changed since the last activation of the reduced area controller via the reduced area adaptation, so AR_RED_DIF_I_REL_old is not necessarily 0.

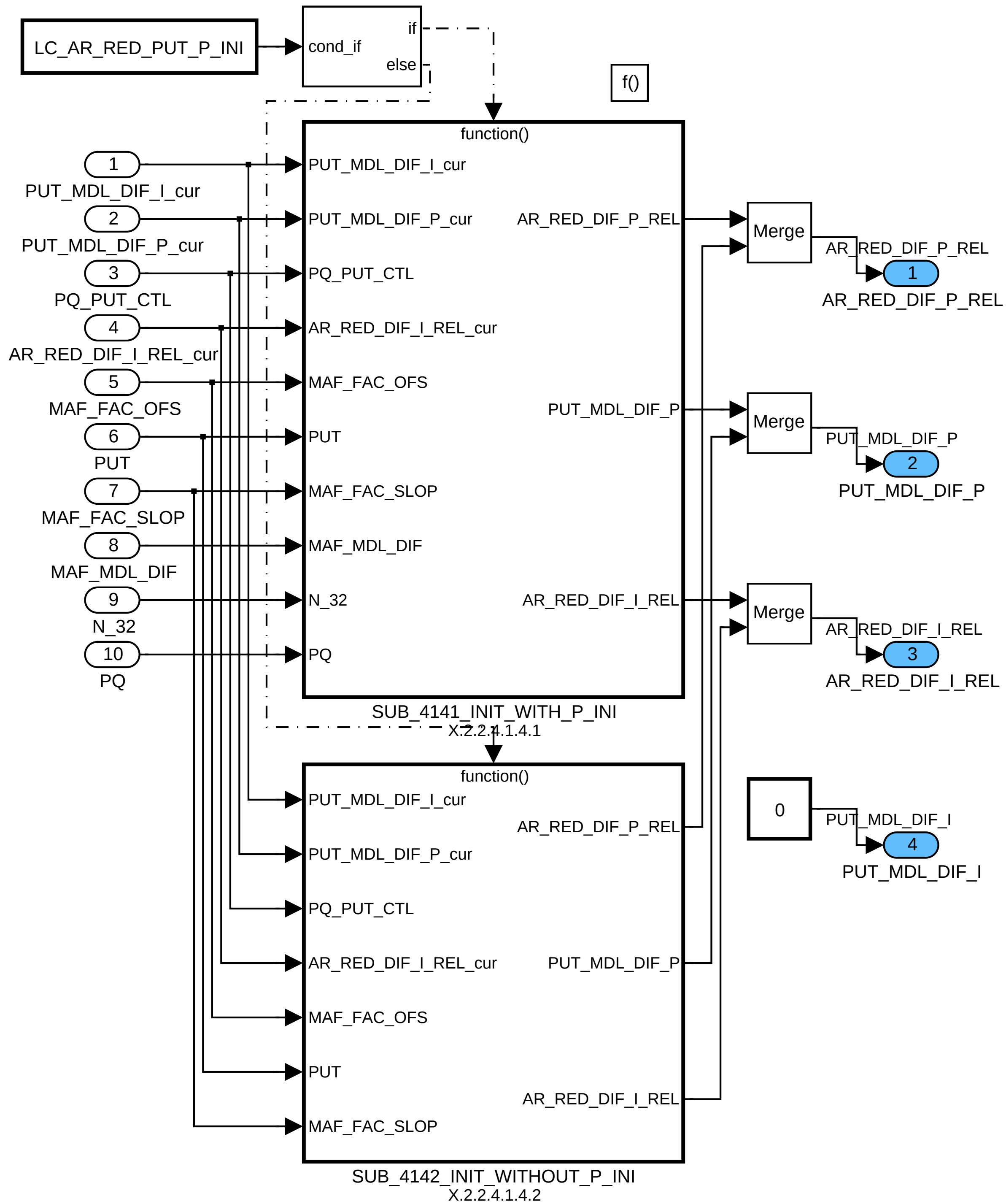


Figure 162:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_414_INIT_INSY_MAF_AR_CTL

4.14.2.2.4.1.4.1 SUB_4141: Initialization with the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 1):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is also initialized. The P share of the old state is set to 0.

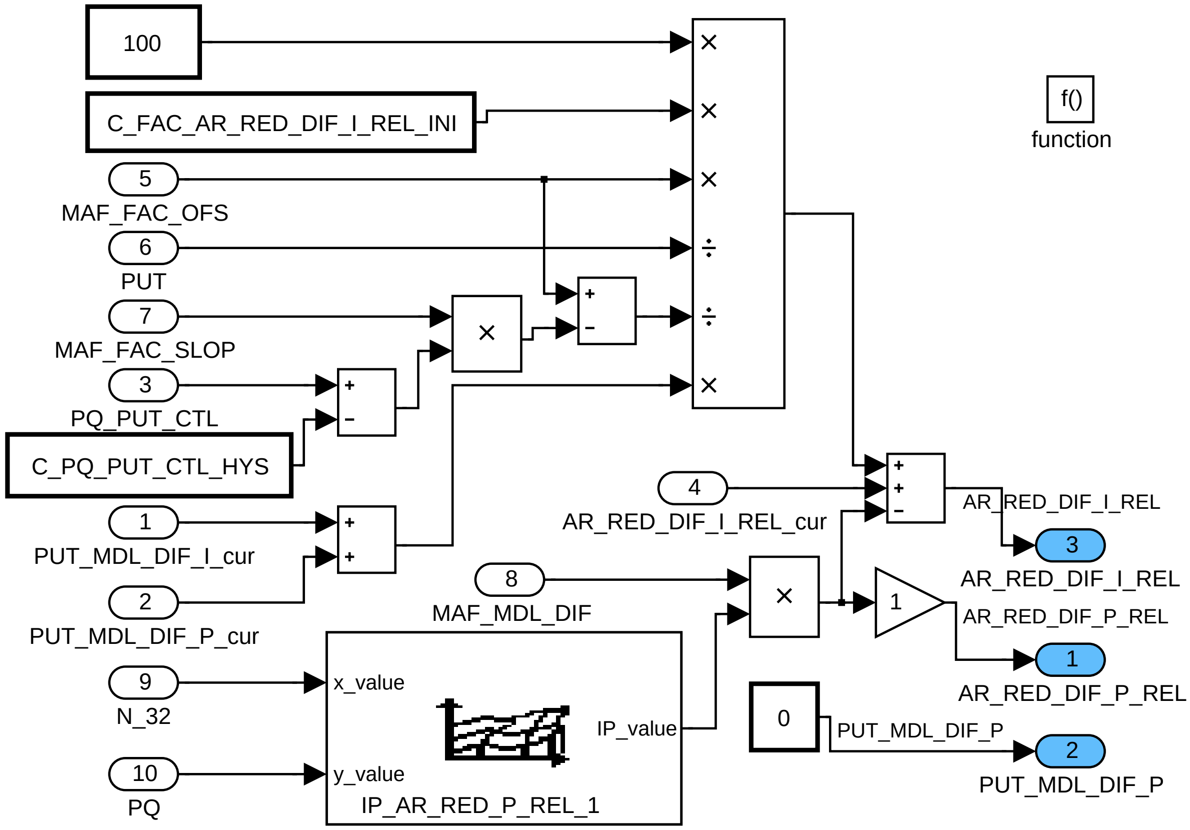


Figure 163:

Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_414_INIT_INSY_MAF_AR_CTL/SUB_4141_INIT_WITH_P_INI

4.14.2.2.4.1.4.2 SUB_4142: Initialization without the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 0):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is not initialized. The P share of the old state is ramped down to 0.

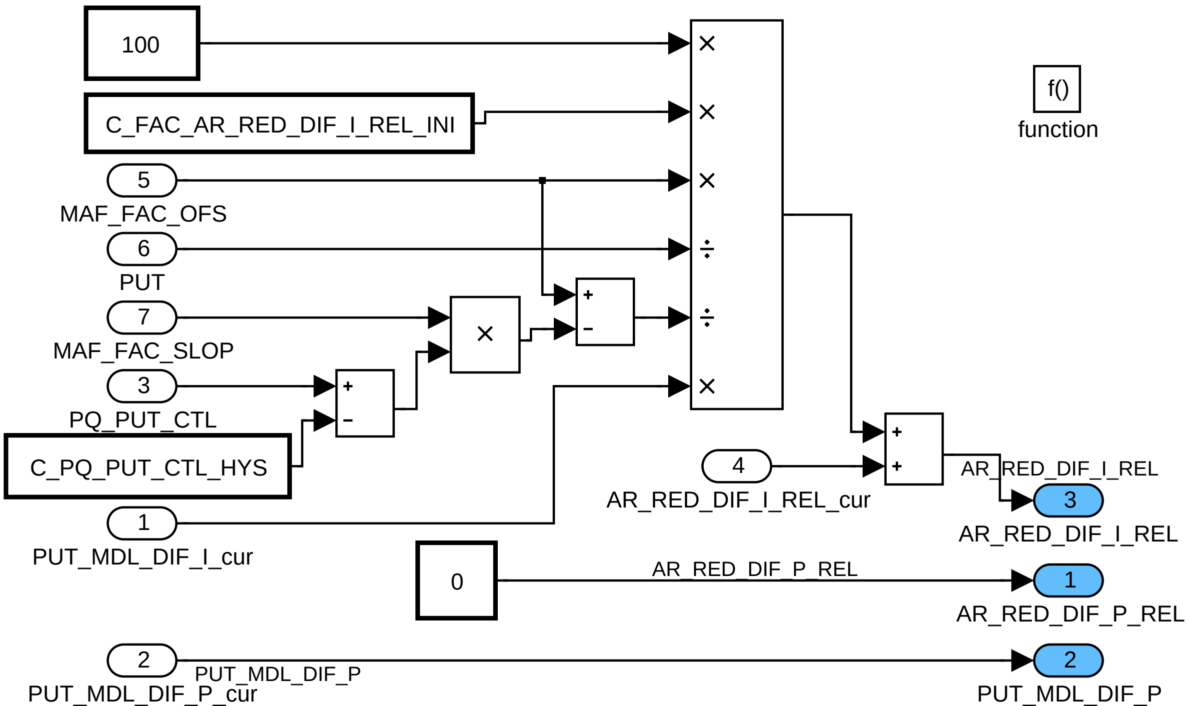


Figure 164:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_4_INSY_MAF_PUT_CTL/SUB_41_INSY_MAF_PUT_CTL/SUB_414_INIT_INSY_MAF_AR_CTL/SUB_4142_INIT_WITHOUT_P_INI

4.14.2.2.5 SUB_5: MAP sensor based intake manifold model adjustment via pressure upstream throttle (INSY_MAP_PUT_CTL):

4.14.2.2.5.1 SUB_51: MAP sensor based intake manifold model adjustment via pressure upstream throttle (INSY_MAP_PUT_CTL)

The Pressure Controller works based on the manifold pressure deviation MAP_MDL_DIF. The Reduced Area Controller is switched off. AR_RED_DIF_P_REL and AR_RED_DIF_I_REL are not calculated as long as the pressure controller is active. AR_RED_DIF_P_REL is ramped down to 0.

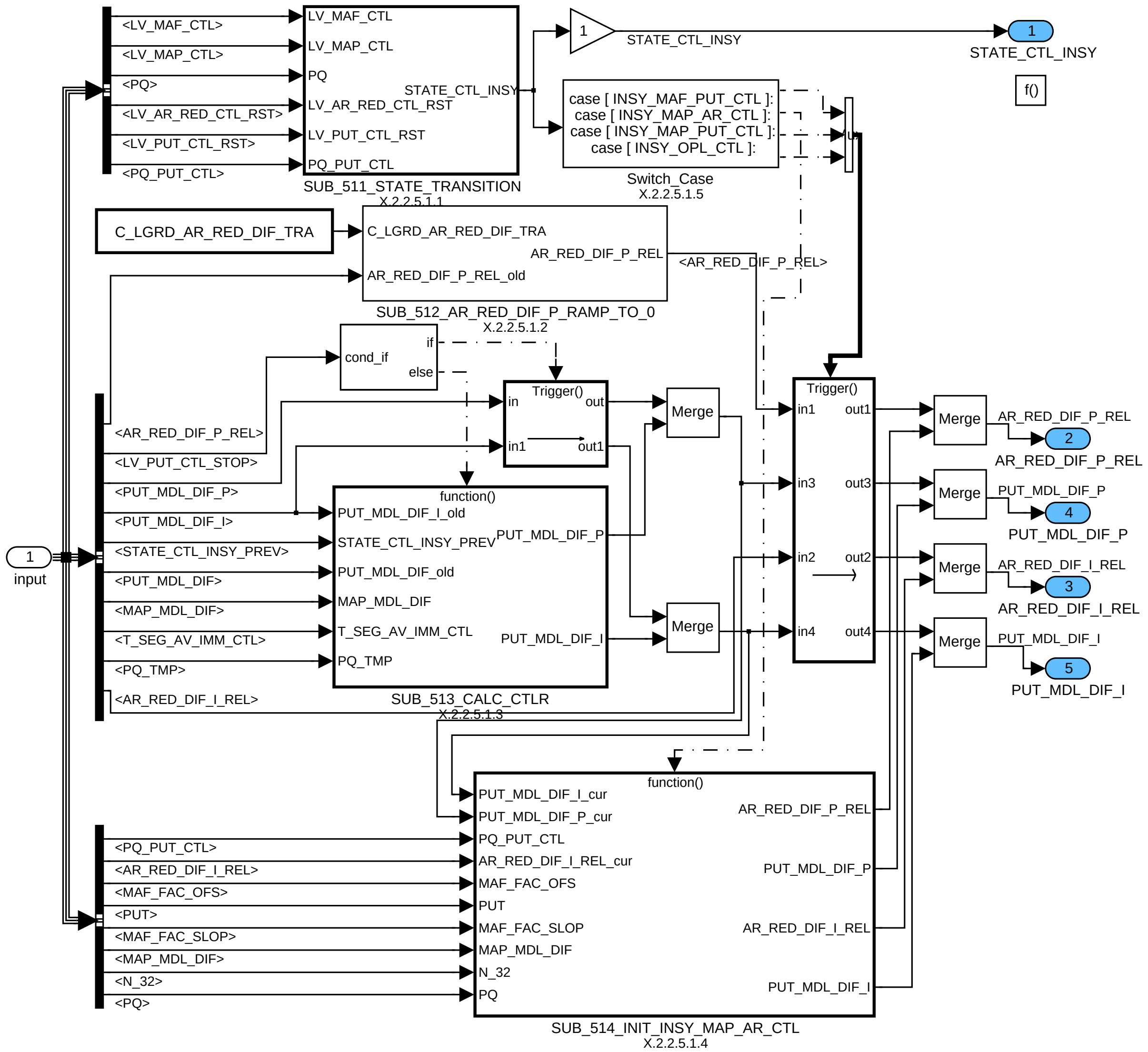


Figure 165:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL

4.14.2.2.5.1.1 SUB_511: Check if a state transition is requested:

The INSY controller can be deactivated (switched to open loop) by the logical variables LV_AR_RED_CTL_RST and LV_PUT_CTL_RST.

From INSY_MAP_PUT_CTL a switch over to INSY_OPL_CTL, INSY_MAF_PUT_CTL and INSY_MAP_AR_CTL is possible. If a switch to INSY_MAF_AR_CTL is required, which can happen if C_PQ_PUT_CTL is close to C_PQ_MAF_CTL_MAX, the system is switched over to INSY_MAP_AR_CTL first. The switch to INSY_MAF_AR_CTL is performed at the next execution cycle of the INSY controller. To avoid an unsteady controller output signal when switching over to the Reduced Area controller, its integral part is accordingly initialized.

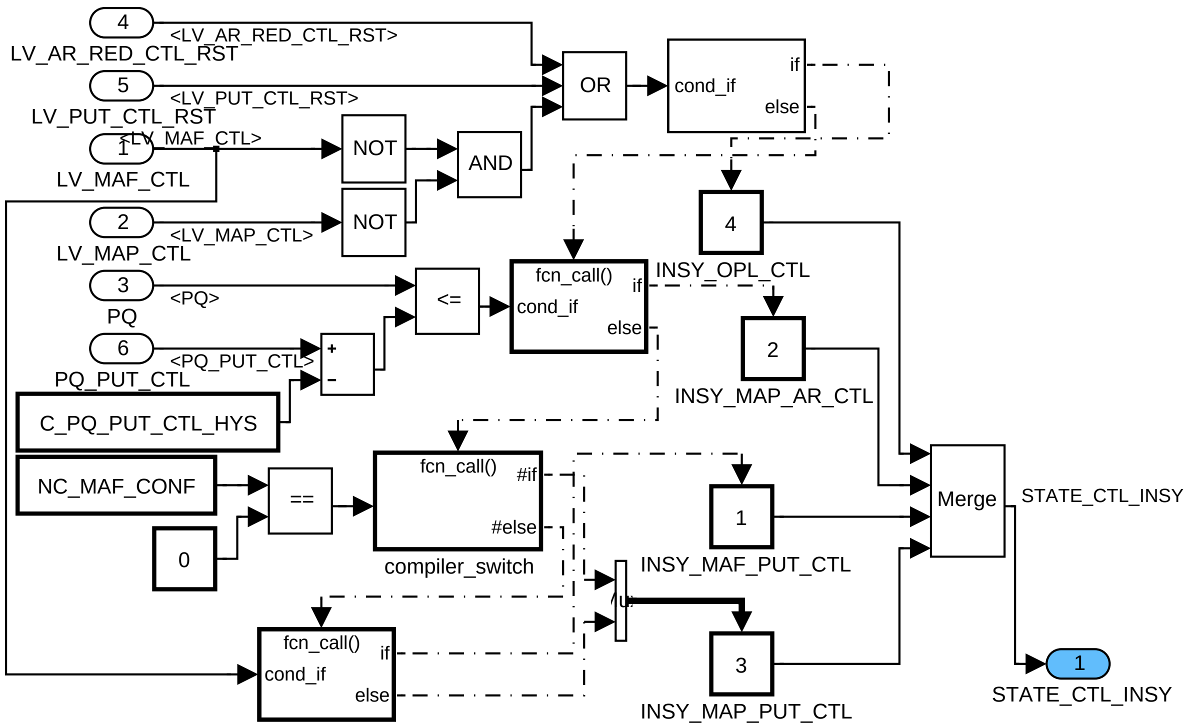


Figure 166:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_511_STATE_TRANSITIO
N

4.14.2.2.5.1.2 SUB_512: AR_RED_DIF_P_REL is ramped down to 0:

AR_RED_DIF_P_REL is ramped down to 0 with a gradient of C_LGRD_AR_RED_DIF_TRA beginning with the STATE_CTL_INSY transition.

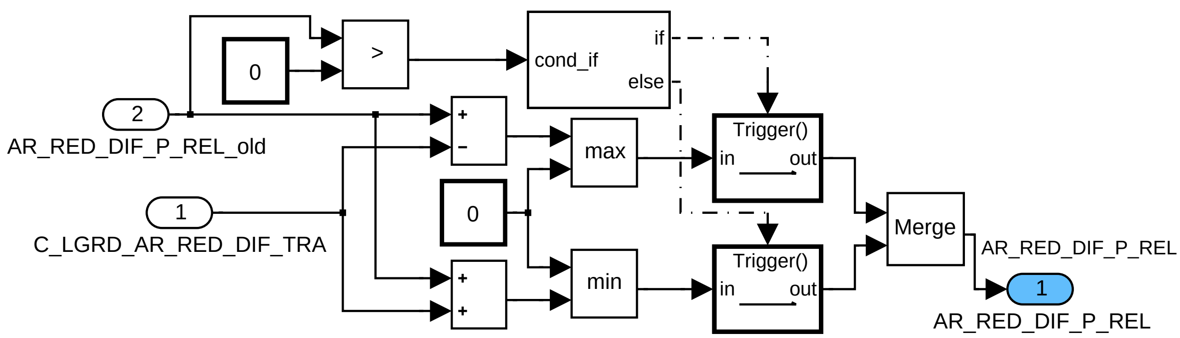


Figure 167:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_512_AR_RED_DIF_P_R
AMP_TO_0

4.14.2.2.5.1.3 SUB_513: Calculations of controller outputs (PI):

Controller initialization for smooth transition from INSY_MAF_PUT_CTL if STATE_CTL_INSY_old = 1 or normal controller calculation otherwise.

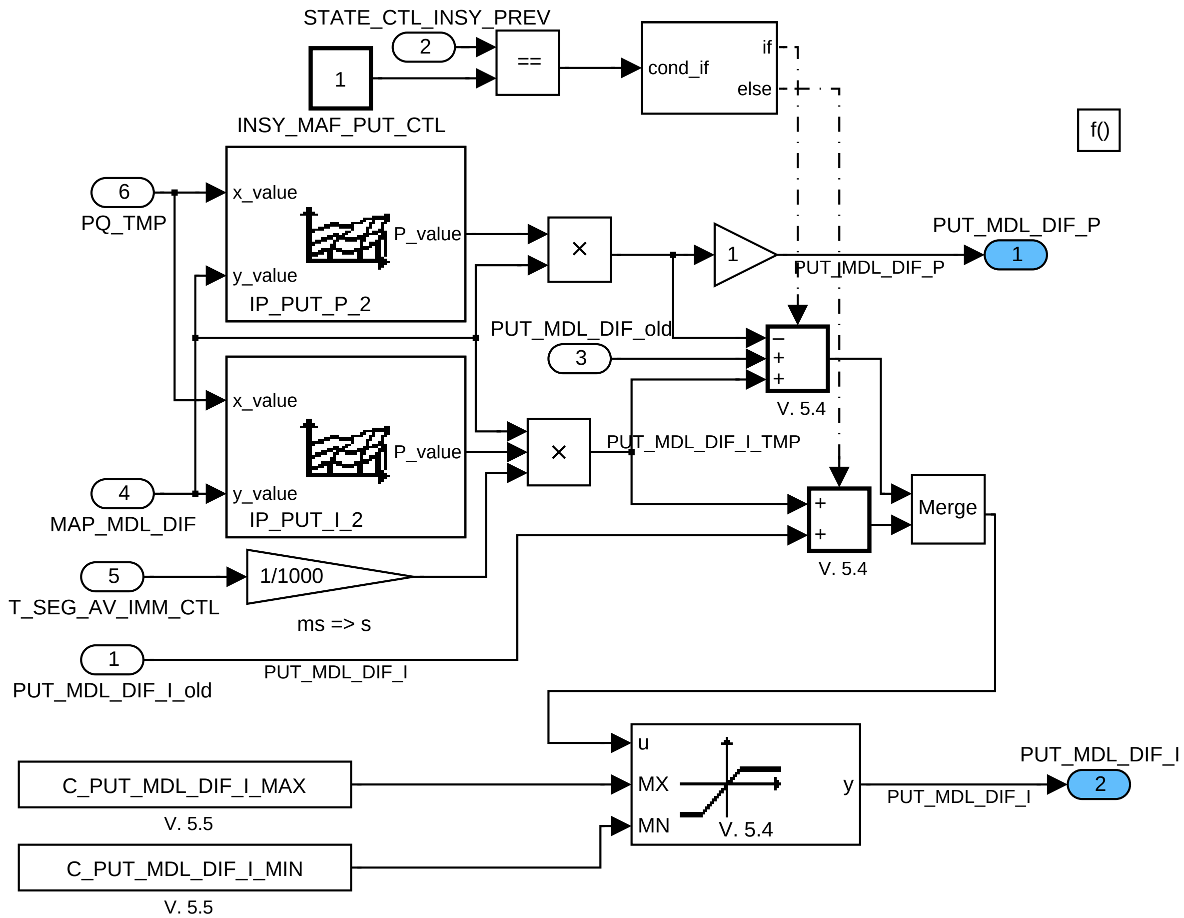


Figure 168:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_513_CALC_CTLR

4.14.2.2.5.1.4 SUB_514: INSY_MAP_PUT_CTL:Initialization of controller outputs at transition to reduced throttle area controlled (INSY_MAP_AR_CTL):

At transition between PUT and AR_RED controlled states, two initialization modes are possible depending of LC_AR_RED_PUT_P_INI.

Controller initialization for smooth transition to INSY_MAP_AR_CTL: AR_RED_DIF_I_REL can have been changed since the last activation of the reduced area controller via the reduced area adaptation, so AR_RED_DIF_I_REL_n-1 is not necessarily 0.

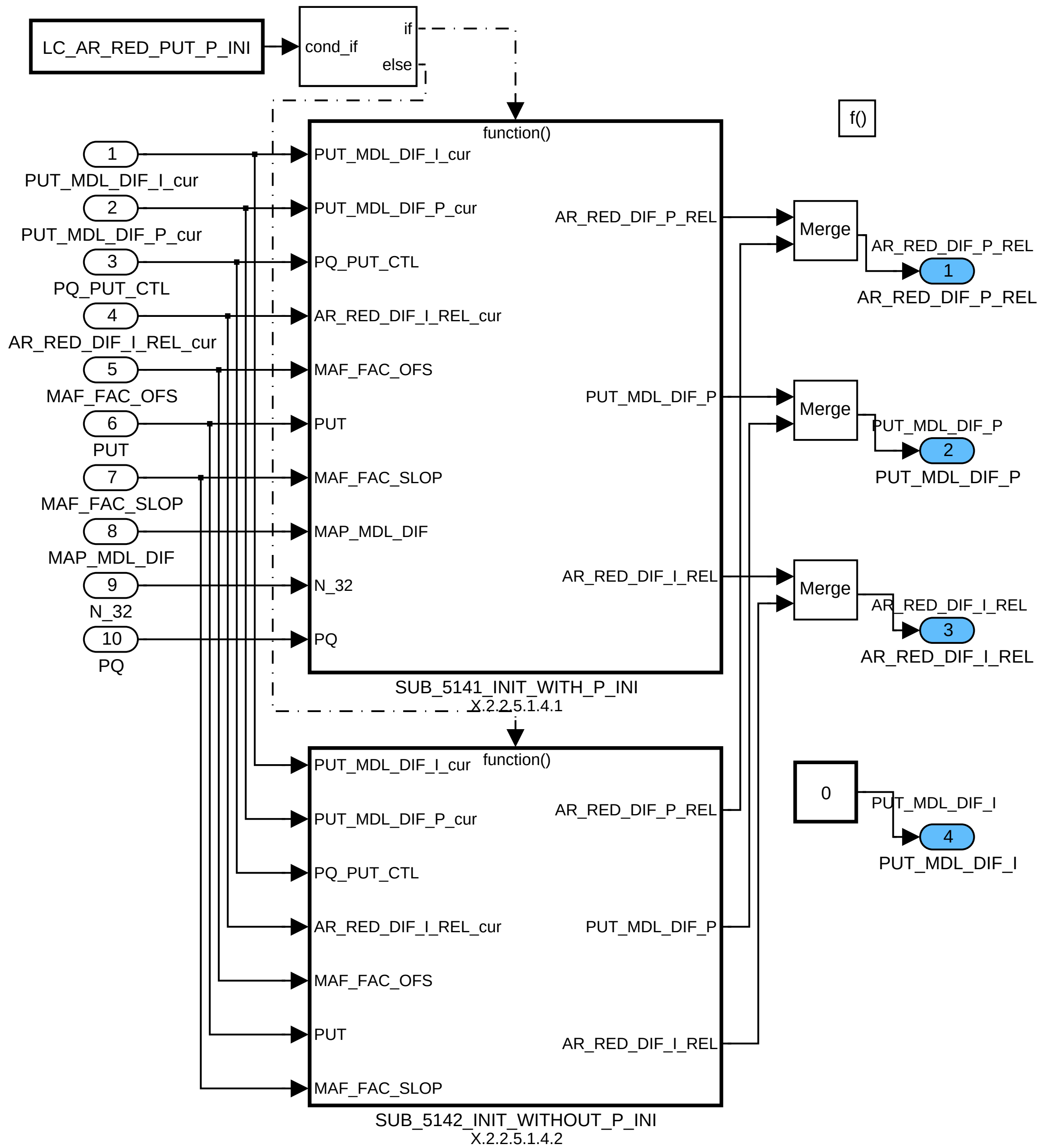


Figure 169:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_514_INIT_INSY_MAP_A R_CTL

4.14.2.2.5.1.4.1 SUB_5141: Initialization with the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 1):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is also initialized. The P share of the old state is set to 0.

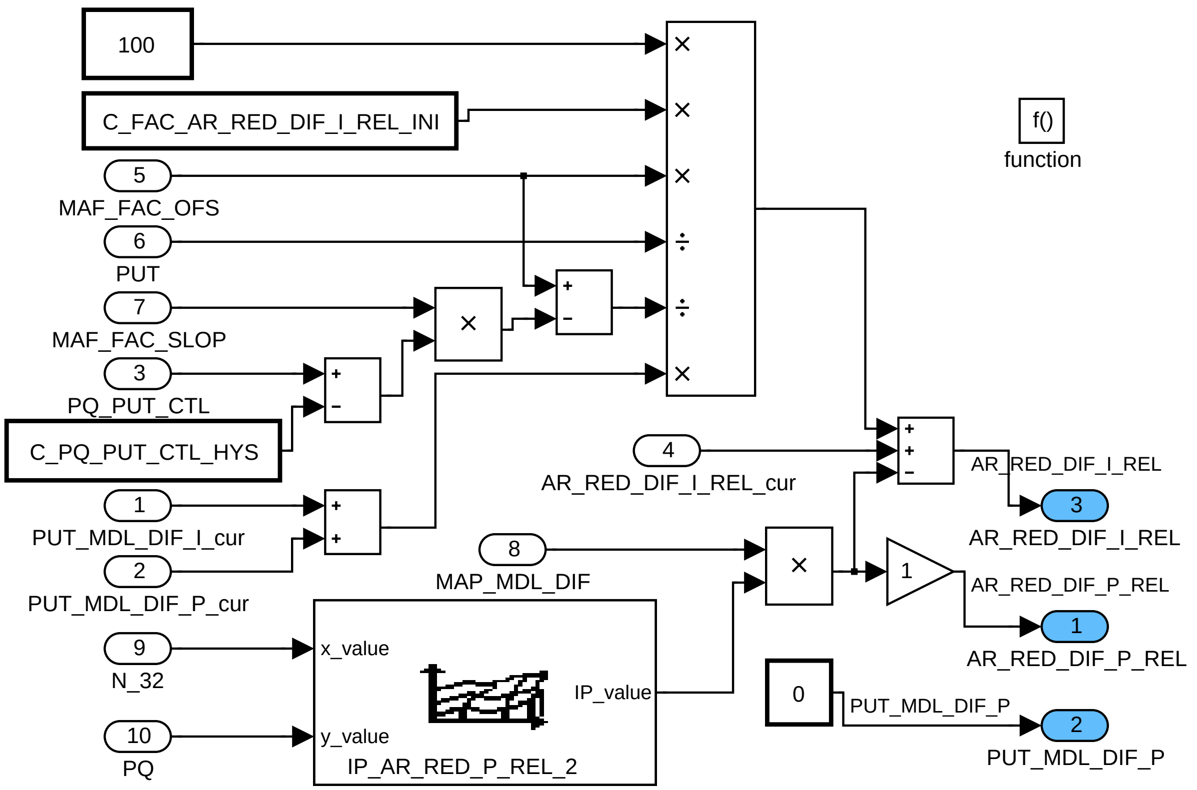


Figure 170:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_514_INIT_INSY_MAP_AR_CTL/SUB_5141_INIT_WITH_P_INI

4.14.2.2.5.1.4.2 SUB_5142: Initialization without the initialization of the P-share of the new state (LC_AR_RED_PUT_P_INI = 0):

The I share for the new state is initialized based on the I share of the old state. The P share of the new state is not initialized. The P share of the old state is ramped down to 0.

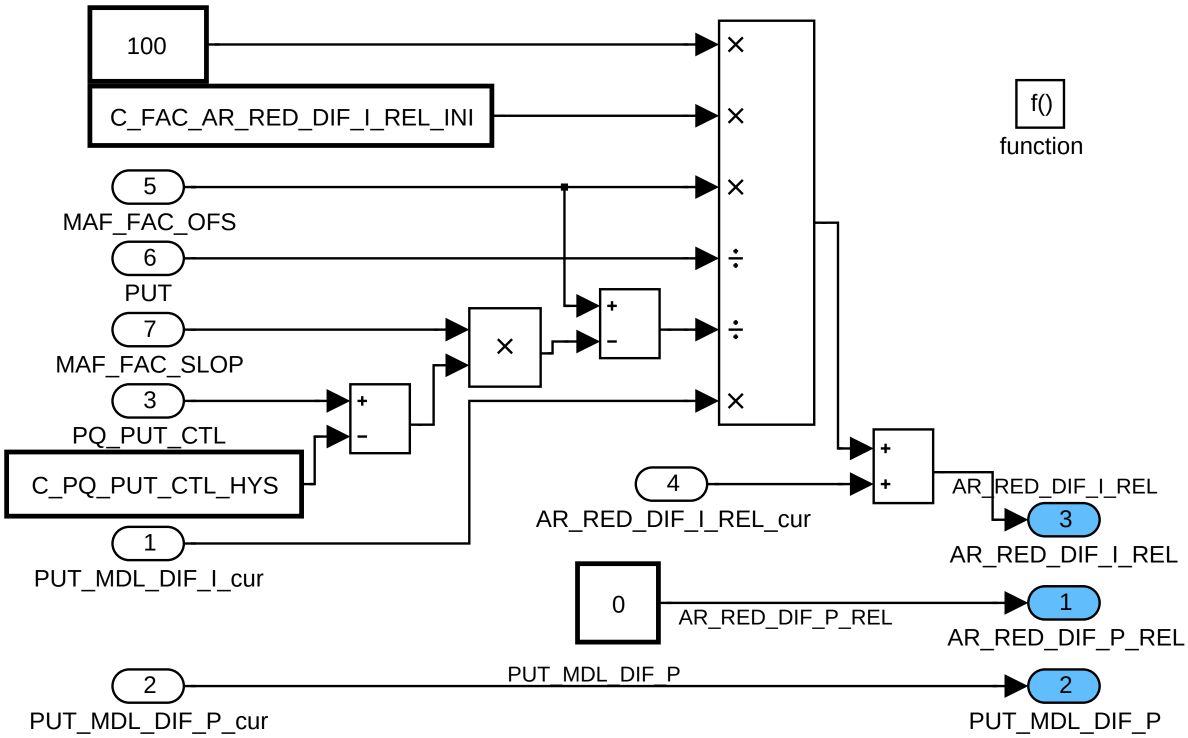


Figure 171:
Path:
INSY_M404T/operate_SEG/OPM_SEG/SUB_5_INSY_MAP_PUT_CTL/SUB_51_INSY_MAP_PUT_CTL/SUB_514_INIT_INSY_MAP_AR_CTL/SUB_5142_INIT_WITHOUT_P_INI

4.14.2.2.6 SUB_6: No intake manifold model adjustment to any load sensor:

Both Reduced Area Controller and Pressure controller are inactive in INSY_OPL_CTL state. Both P and I share of both controllers are ramped to 0 with the gradients C_LGRD_AR_RED_DIF_TRA and C_LGRD_PUT_MDL_DIF_TRA and C_LGRD_PUT_MDL_DIF_I_TRA to avoid jumps in AR_RED and PUT_MDL when switching over to STATE_INSY_CTL.

When in INSY_OPL_CTL occurs either LV_MAF_CTL = 1 or LV_MAP_CTL = 1, then depending on the current PQ the system switches to one of the four closed loop control states INSY_MAF_AR_CTL, INSY_MAP_AR_CTL, INSY_MAF_PUT_CTL, INSY_MAP_PUT_CTL. It has to be ensured by the Intake Manifold Model application incidences that LV_MAF_CTL and LV_MAP_CTL never are set at the same time.

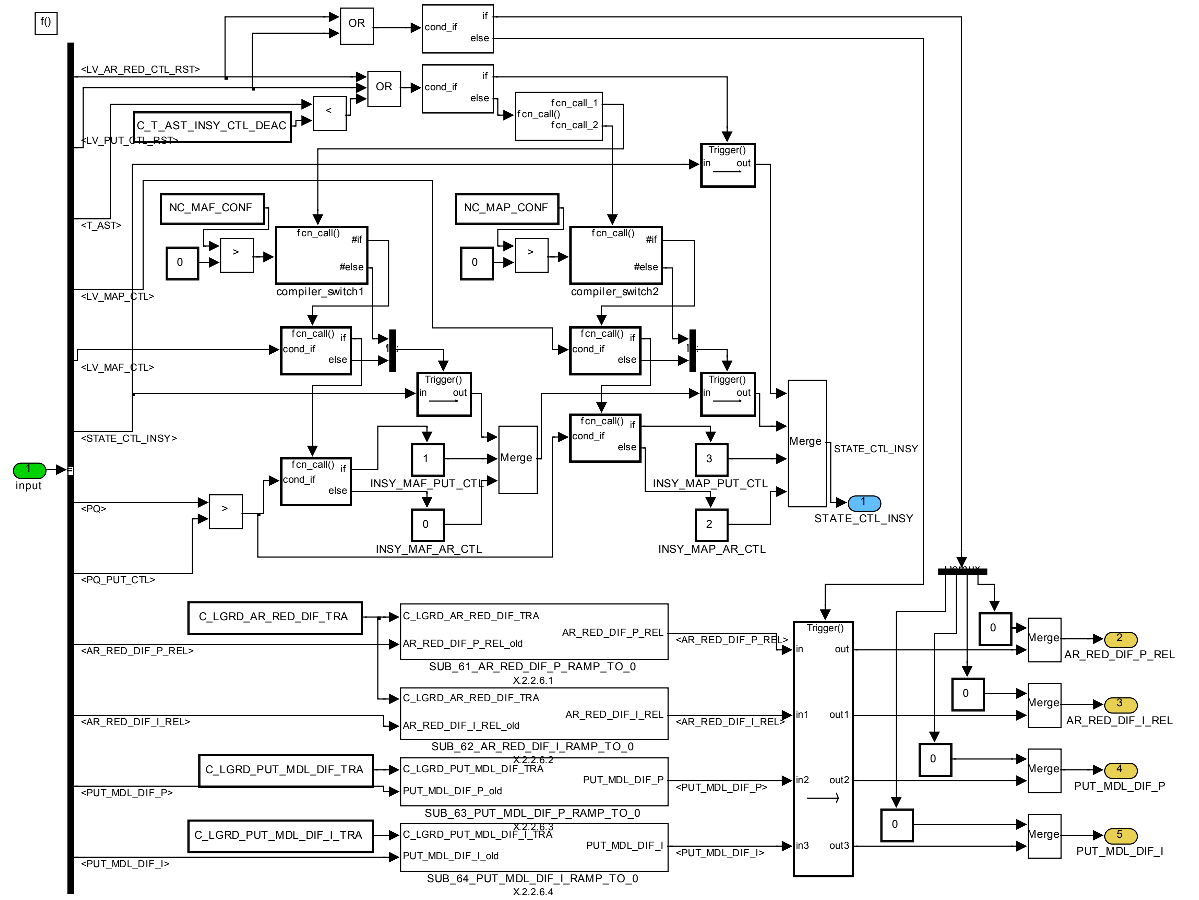


Figure 172:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_6_INSY_OPL_CTL

4.14.2.2.6.1 SUB_61: AR_RED_DIF_P_REL is ramped down to 0:

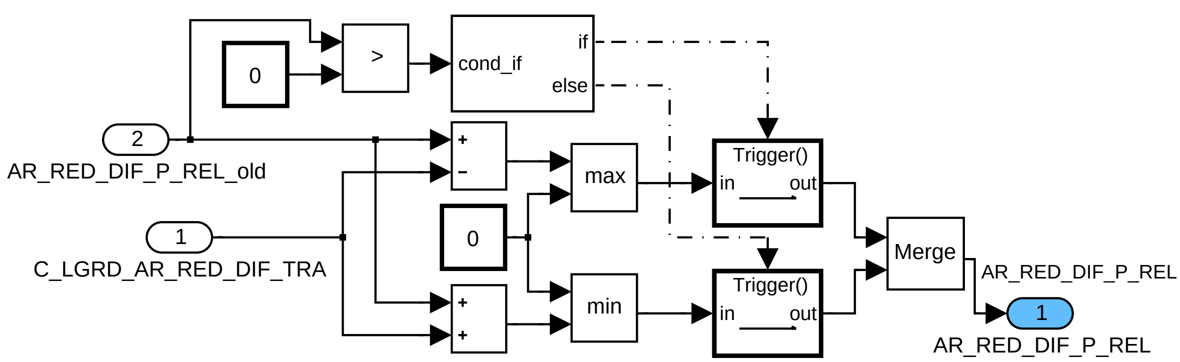


Figure 173:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_6_INSY_OPL_CTL/SUB_61_AR_RED_DIF_P_RAMP_TO_0

4.14.2.2.6.2 SUB_62: AR_RED_DIF_I_REL is ramped down to 0:

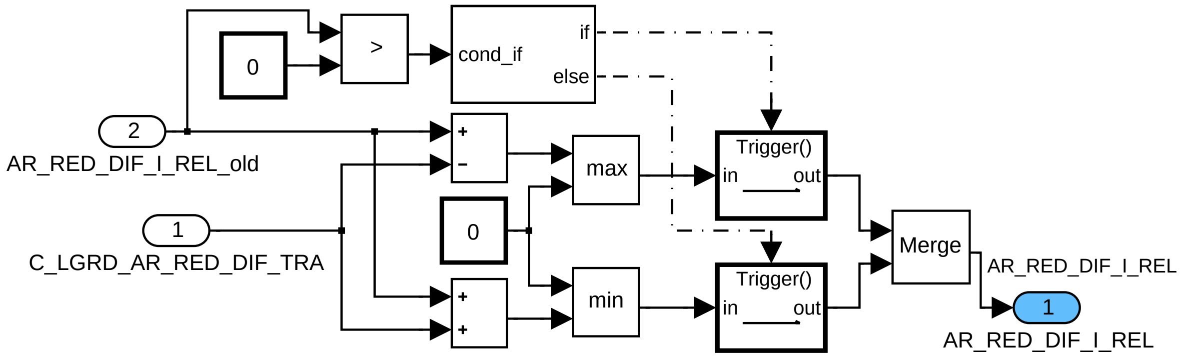


Figure 174:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_6_INSY_OPL_CTL/SUB_62_AR_RED_DIF_I_RAMP_TO_0

4.14.2.2.6.3 SUB_63: PUT_MDL_DIF_P is ramped down to 0:

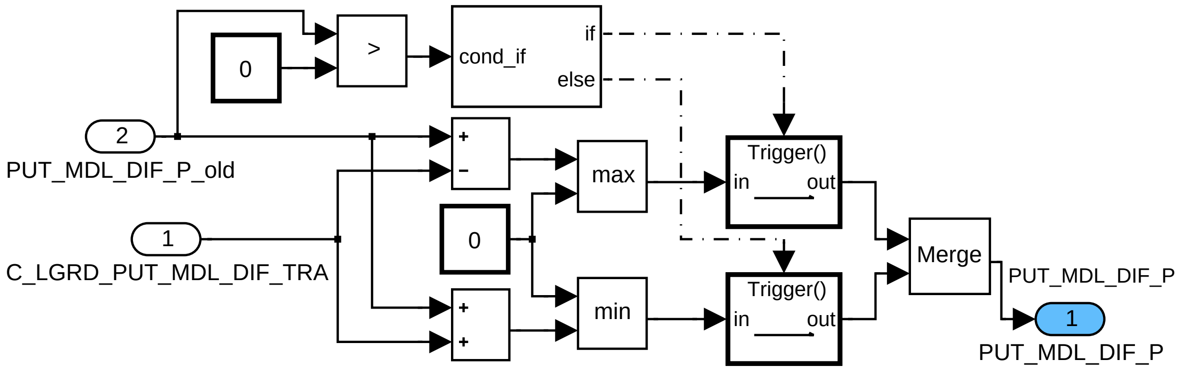


Figure 175:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_6_INSY_OPL_CTL/SUB_63_PUT_MDL_DIF_P_RAMP_TO_0

4.14.2.2.6.4 SUB_64: PUT_MDL_DIF_I is ramped down to 0:

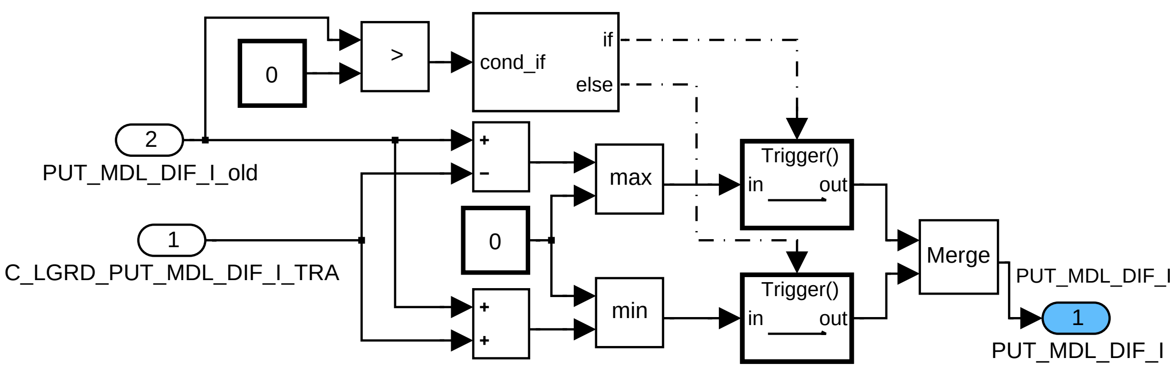


Figure 176:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_6_INSY_OPL_CTL/SUB_64_PUT_MDL_DIF_I_RAMP_TO_0

4.14.2.2.7 SUB_7: Calculation of the reduced area controller output AR_RED_DIF_REL:

Reduced area controller output AR_RED_DIF_REL is calculated. The final reduced area AR_RED includes basic reduced area AR_RED_BAS based on TPS, the controller output AR_RED_DIF_REL and the adaptive corrections.

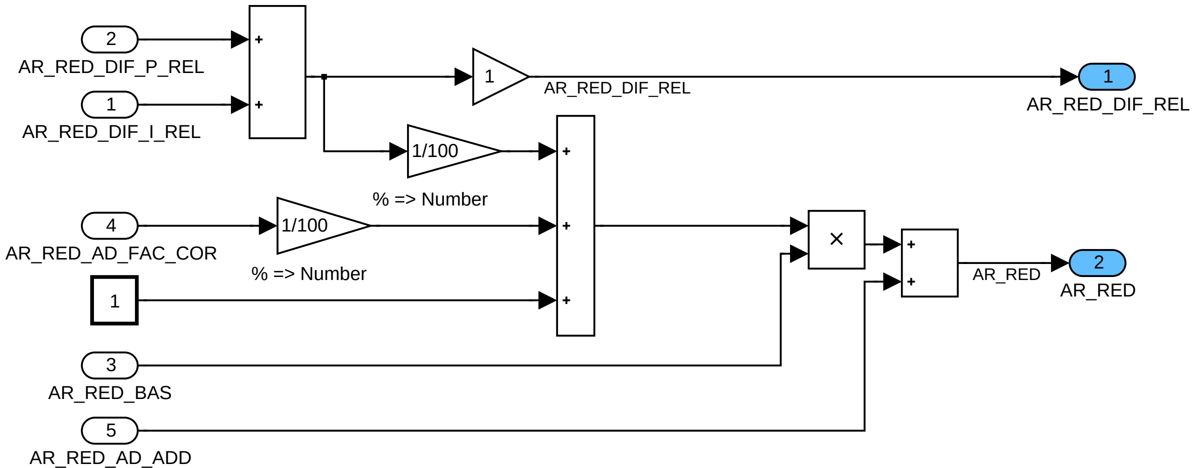


Figure 177:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_7_CALC_AR_RED

4.14.2.2.8 SUB_8: Calculation of the pressure upstream throttle:

For engines with turbocharger or supercharger the pressure upstream throttle PUT is equal to a pressure upstream throttle substitute value if a PUT sensor value is not available or defective. If a PUT sensor is available and o.k. then the measured PUT value is used.

For engines without charger PUT is equal to the pressure downstream air cleaner PRS_AIC_DOWN.

The P- and I-share of the pressure controller are summed up to PUT_MDL_DIF.

For turbocharged engines and engines without charger PUT and PUT_MDL_DIF are summed up to PUT_MDL, for supercharged engines PUT_MDL_DIF is not taken into consideration.

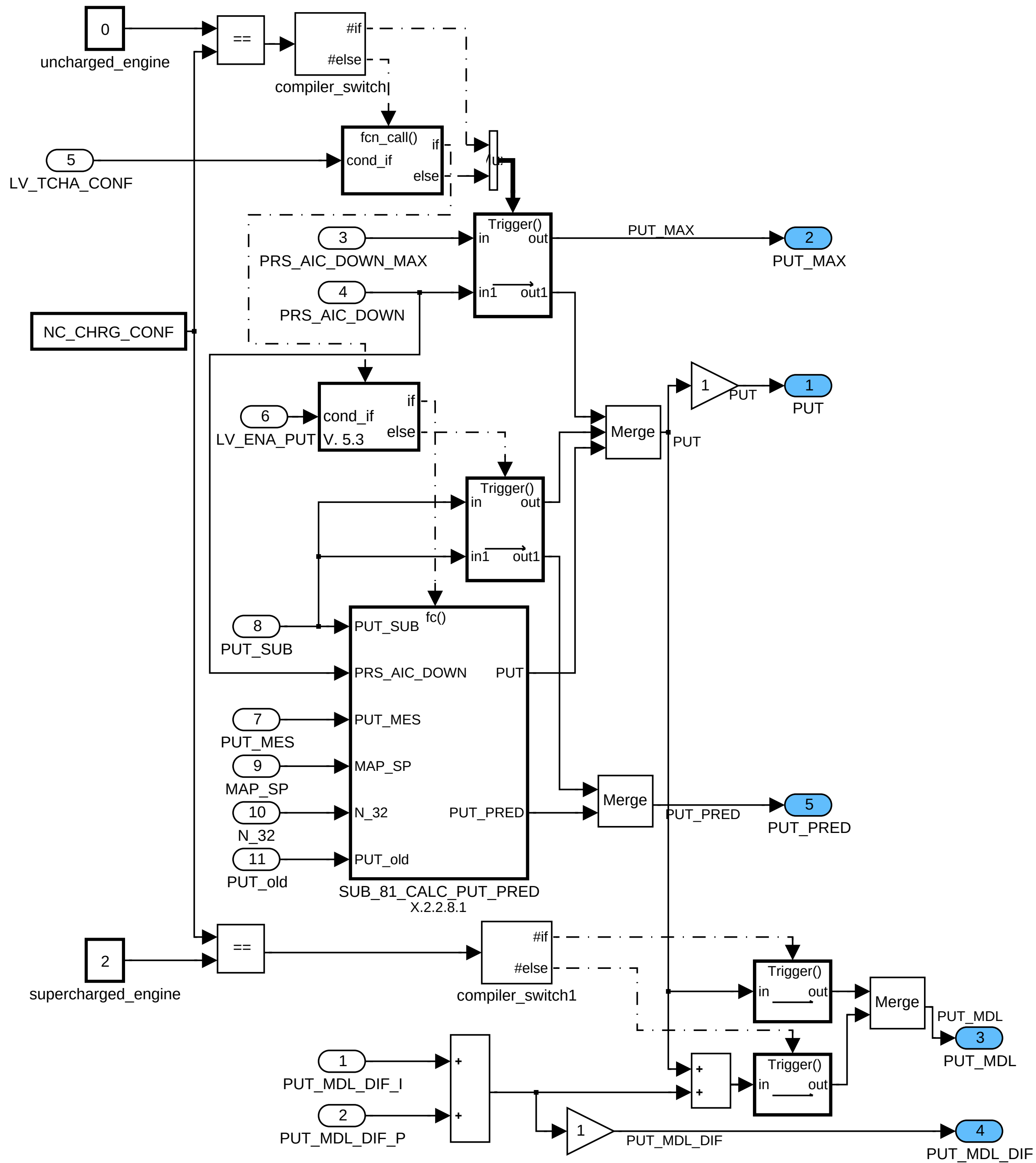


Figure 178:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_8_CALC_PUT

4.14.2.2.8.1 SUB_81: Calculation of the predicted pressure upstream throttle:

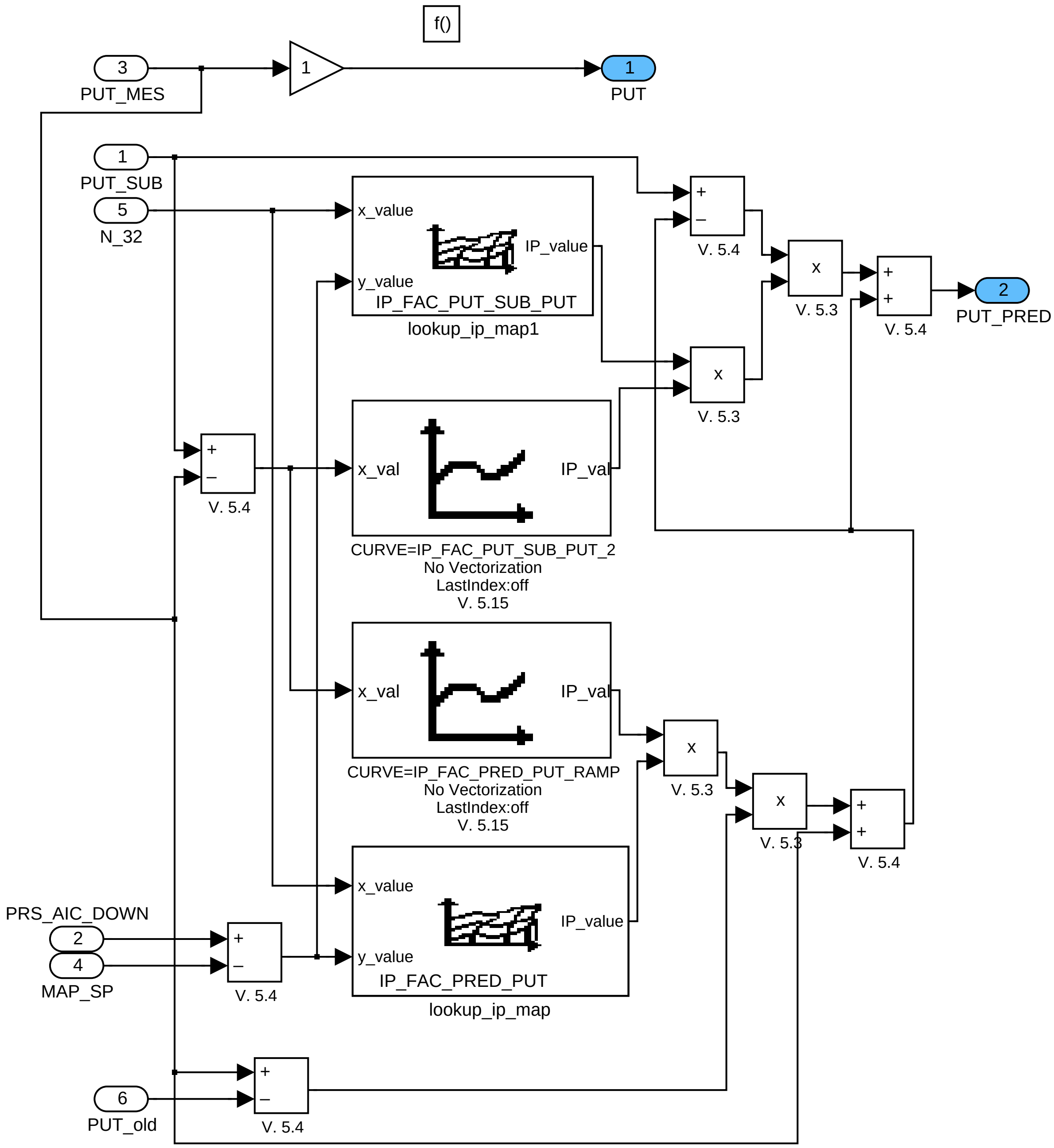


Figure 179:
Path: INSY_M404T/operate_SEG/OPM_SEG/SUB_8_CALC_PUT/SUB_81_CALC_PUT_PRED

4.15 Reduced area adaptation

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
AR_RED_AD_ADD	O/V/S	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Additive adaptive parameter					
AR_RED_AD_ADD_DIF	V	0... FFFFH	0... 58.592855	894.07e-6	[cm²]
Adaptation demand during a driving cycle					
AR_RED_AD_ADD_MMV	O/V/S	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Moving mean value of the additive adaptive parameter					
AR_RED_AD_ADD_REL_DIF	O/V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Increment of additive adaptive parameter (relative)					
AR_RED_AD_ADD_TMP_1	V	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Temporary variable					
AR_RED_AD_ADD_TMP_SAVE	V	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Temporary variable					
AR_RED_AD_FAC	O/V/S	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Multiplicative adaptive parameter					
AR_RED_AD_FAC_COR	O/V/S	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Correction factor for entire adaptive parameter					
AR_RED_AD_FAC_DIF	O/V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Increment of multiplicative adaptive parameter					
AR_RED_AD_FAC_TMP_SAVE	V	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Temporary variable					
CTR_AR_RED_AD_ADD_FAST	O/V/S	0... FFH	0... 255	1	[-]
Counter to control the fast adaptation					
LV_ACT_AR_RED_AD_ADD	V	0... 1H	0... 1	1	[-]
Flag of conformed conditions for the adaptation of the reduced throttle area					
STATE_AR_RED_AD_ADD	O/V	0 1 2 3	PASSIVE RST_OR_STOPPED ACTIVE_NORMAL ACTIVE_FOR_CAM_OFS_AD	-	[-]
State of additive reduced area(AR_RED) adaptation					

Input Data:

AR_RED_BAS	AR_RED_DIF_I_REL_MMV	LV_AR_RED_AD_RST	LV_AR_RED_AD_STOP
LV_ENA_AR_RED_AD	LV_INH_AR_RED_AD	LV_TPS_AD_REQ	N_32
PQ	NLC_CAM_OFS_AD	AR_RED_AD_FAC_REQ_CAM_INTER	AR_RED_AD_ADD_REQ_CAM_INTER
LV_AR_RED_AD_ADD_REQ_CAM_TMP	LV_REQ_ISC	AR_RED_DIF_I_REL	C_AR_RED_DIF_I_REL_MAX
C_AR_RED_DIF_I_REL_MIN			

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_AR_RED_AD_ADD_MAX	1	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Maximum allowed value for additive reduced area adaptation					
C_AR_RED_AD_ADD_MIN	1	8000... 7FFFH	-29.2968745348 ...29.2959804652	894.07e-6	[cm²]
Minimal allowed value for additive reduced area adaptation					
C_AR_RED_AD_COR_CRLC	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Coefficient for the normal adaptation of the reduced throttle area					
C_AR_RED_AD_COR_CRLC_FAST	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Coefficient for the fast adaptation of the reduced are of the throttle (demand adaptation was activated)					
C_AR_RED_AD_FAC_MAX	1	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Maximum allowed value for multiplicative reduced area adaptation					
C_AR_RED_AD_FAC_MIN	1	8000... 7FFFH	-50... 49.9984741211	1.52588e-3	[%]
Minimal allowed value for multiplicative reduced area adaptation					
C_CRLC_AR_RED_AD_FAC	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Factor for the multiplicative reduced area adaptation					
C_CTR_AR_RED_AD_ADD_FAST	1	0... FFH	0... 255	1	[-]
Counter for the control of the fast adaptation of the reduced throttle area					
C_N_AR_RED_AD_ADD_MAX	1	0... FFH	0... 8160	32	[rpm]
Maximum engine speed for learning of additive adaptation value					
C_N_AR_RED_AD_ADD_MIN	1	0... FFH	0... 8160	32	[rpm]
Minimum engine speed for learning of additive adaptation value					
C_N_AR_RED_AD_FAC_MAX	1	0... FFH	0... 8160	32	[rpm]
Maximum engine speed for learning of multiplicative adaptation value					
C_N_AR_RED_AD_FAC_MIN	1	0... FFH	0... 8160	32	[rpm]
Minimum engine speed for learning of multiplicative adaptation value					
C_PQ_AR_RED_AD_ADD_MAX	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Maximum pressure quotient for learning of additive adaptation value (Initial value: 0.35)					
C_PQ_AR_RED_AD_FAC_MAX	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Maximum pressure quotient for learning of multiplicative adaptation value (Initial value: 0.52)					
C_PQ_AR_RED_AD_FAC_MIN	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Minimum pressure quotient for learning of multiplicative adaptation value (Initial value: 0.4)					
IP_CRLC_AR_RED_AD_ADD_MMV	4	0... FFH	0... 0.99609375	3.90625e-3	[-]
LDP_AR_RED_AD_ADD_DIF_IP_CRLC	4	0... FFFFH	0... 58.592855	894.07e-6	[cm²]
Averaging constant for the calculation of the floating mean value AR_RED_AD_ADD_MMV					
IP_FAC_AR_RED_COR	6	0... FFH	0... 0.99609375	3.90625e-3	[-]
LDP_PQ_IP_FAC_AR_RED_COR	6	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Weighting function for decreasing multiplicative learning controller value during PQ>C_PQ_AMP_CTL					

General Information

General information:
At certain operating points depending on engine speed and the pressure ratio a learning algorithm for the adaptation values of the throttle area is implemented. In a brand new ECU the adaptation values AR_RED_AD_ADD and AR_RED_AD_FAC are initialized to 0.
Additional the adaptation can be stopped or reset to 0 by logical variables.

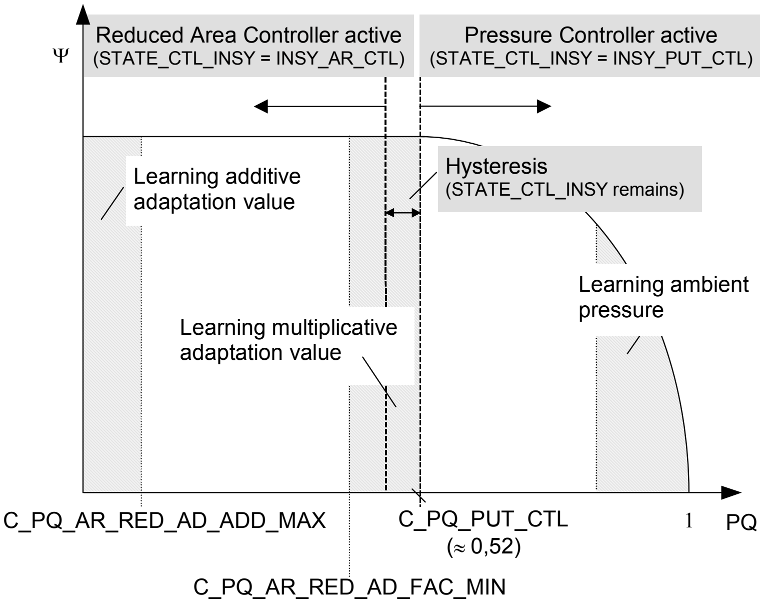


Figure 1: Learning of reduced area adaptation constants

When the adaptation from the throttle is done than the moving mean value (AR_RED_AD_ADD_MMV) has to be set as null.

Reduced area adaptation learning:

In order to compensate serial production tolerances an adaptive correction of the reduced area is learned. The additive and multiplicative adaptation corrections AR_RED_AD_ADD and AR_RED_AD_FAC are used in calculating the reduced area of the throttle AR_RED.

For calculation of the new adaptation value the filtered integral part of the controller is used.

The additive value AR_RED_AD_ADD (offset in air-mass flow through blow-by) is only learned at closed throttle and if the engine speed is inside a calibratable range.

The adaptation is only allowed, when LV_ENA_AR_RED_AD has been set by the Component and Adaptation Manager module. This flag also considers that the reduced area adaptation is inhibited if the MAF/MAP-TPS plausibility check detects implausibilities between the MAF/MAP and the TPS signal (LV_ERR_RATIO_CHK = 1). Additionally, the adaptation is enforced when the I part of the reduced area controller is out of range (in idle) in order to make the LoadTPS ratio check more sensitive. In case of the AMP adaptation is not learned the reduced area adaptation will be inhibited.(LV_INH_AR_RED_AD).

If the actual and the last learned value of AR_RED_AD_ADD differ, then the integral part AR_RED_DIF_I_REL of the INSY controller is corrected.

The learned values are limited to calibratable minimum and maximum values.

When the conditions of the reduced area adaptation are confirmed (LV_ACT_AR_RED_AD_ADD = 1) the first calculation from the additive value will be not perform before the process of a whole interval of calculation is done (2000ms). The verification of the conditions of activation takes place every 1000ms.

The multiplicative value (e.g. HFM value influenced by humidity) is only learned in a calibratable range of PQ near 0.52 (transition from overcritical to undercritical flow through the throttle) and if in addition the engine speed is inside a specified range.

Application Conditions

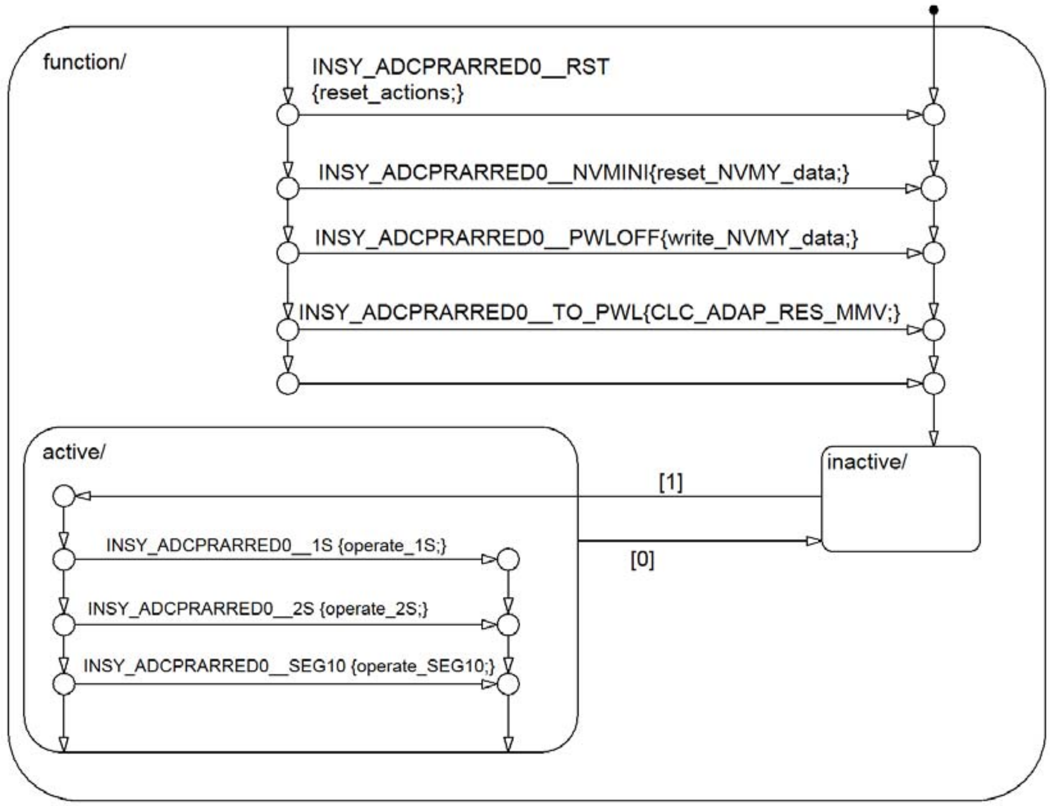


Figure 180:
Path: INSY_ADCPRARRED0/APP_CDN/Chart

Function description

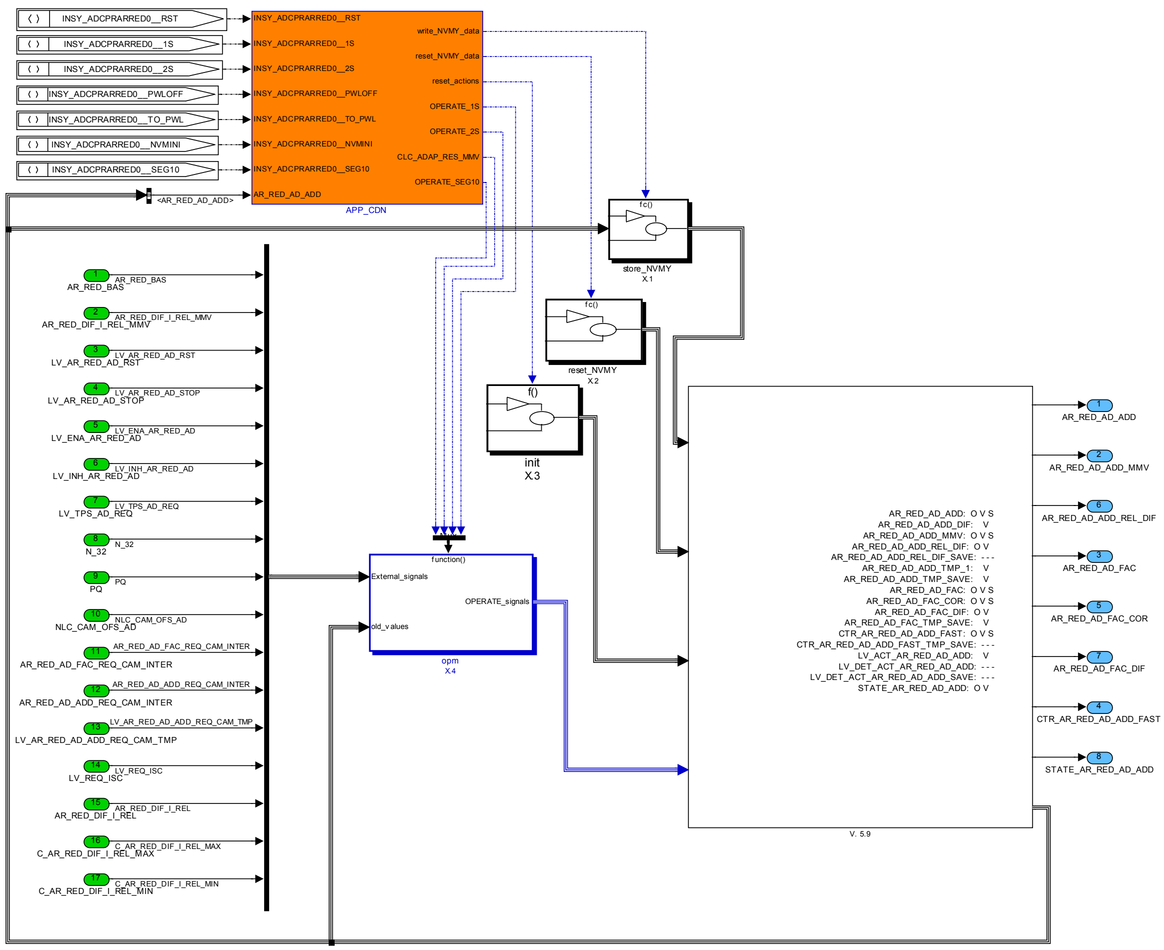


Figure 181:
Path: INSY_ADCPRARRED0

4.15.1 STORE_NVMY

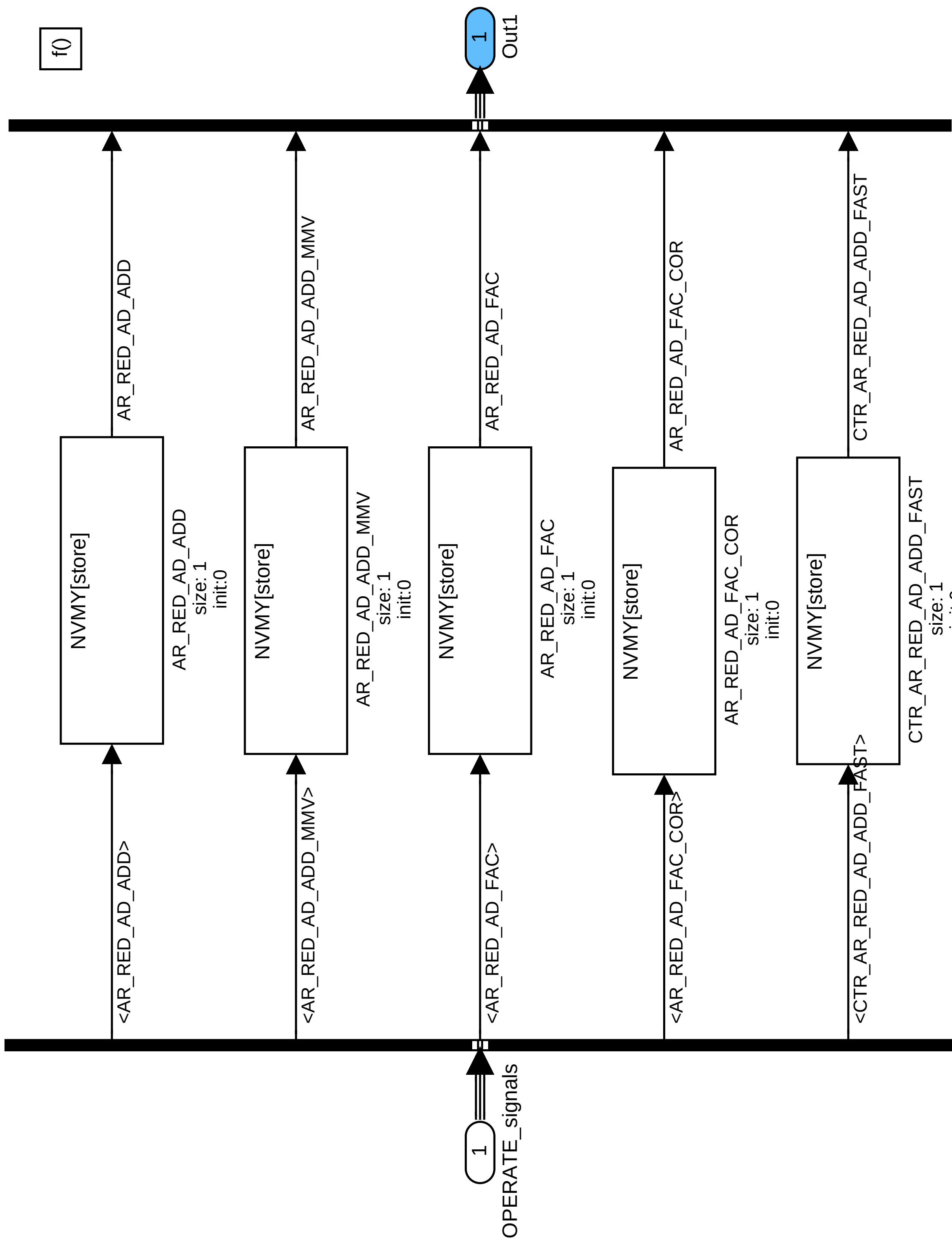


Figure 182:
Path: INSY_ADCPRARRED0/store_NVMY

4.15.2 RESET_NVMY

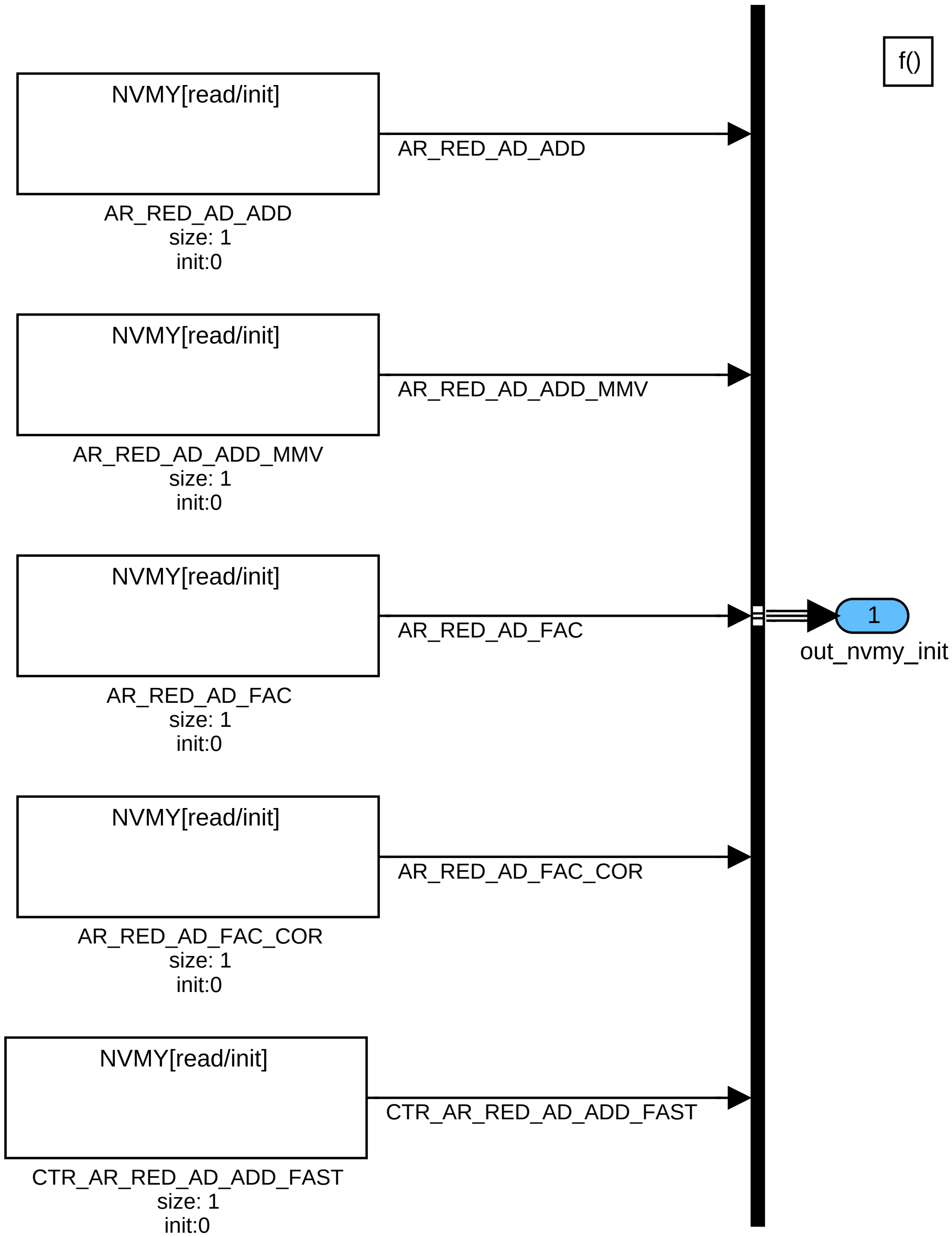


Figure 183:
Path: INSY_ADCPRARRED0/reset_NVMY

4.15.3 INITIALIZATION

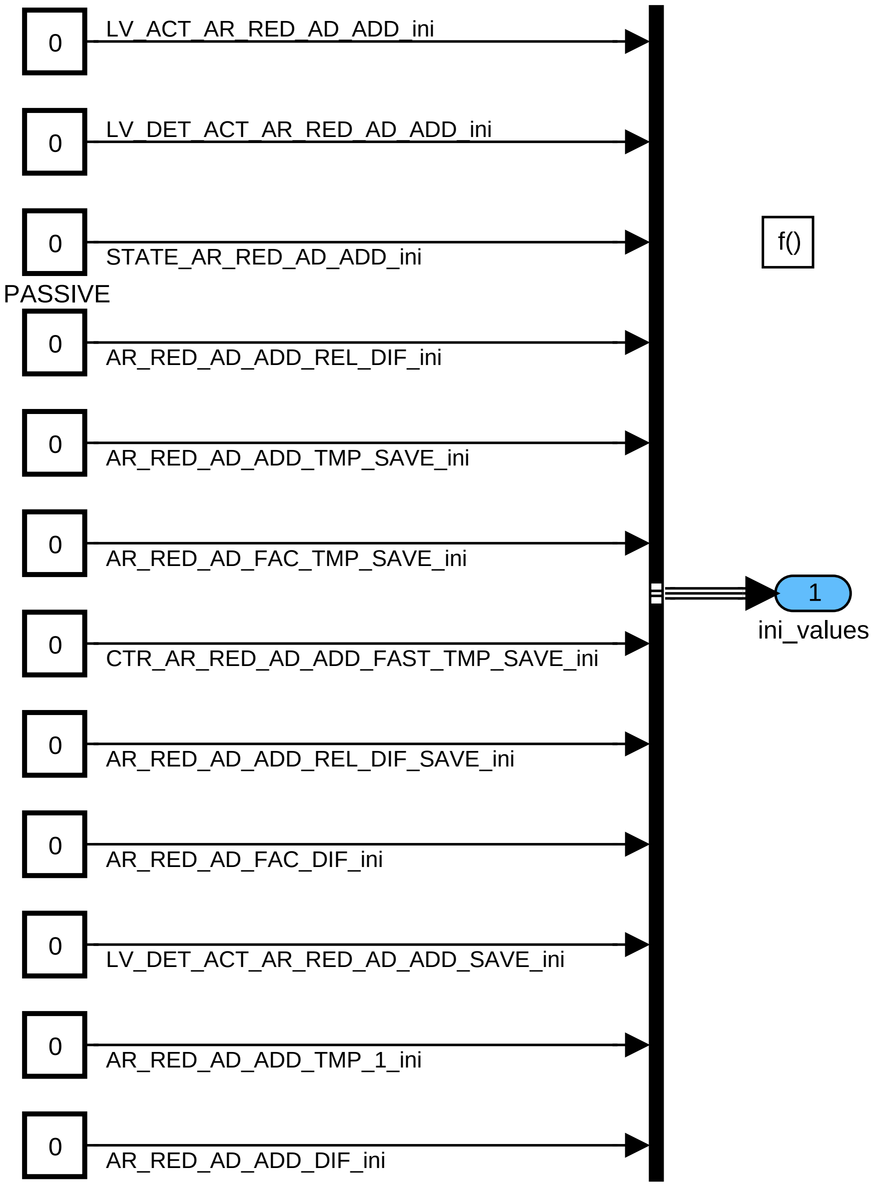


Figure 184:
Path: INSY_ADCPRARRED0/init

4.15.4 Operate:

The variable AR_RED_AD_FAC_COR is calculated every 10ms if LV_ES = 1 or it is calculated every Segment if LV_ES = 0 (SEG10).

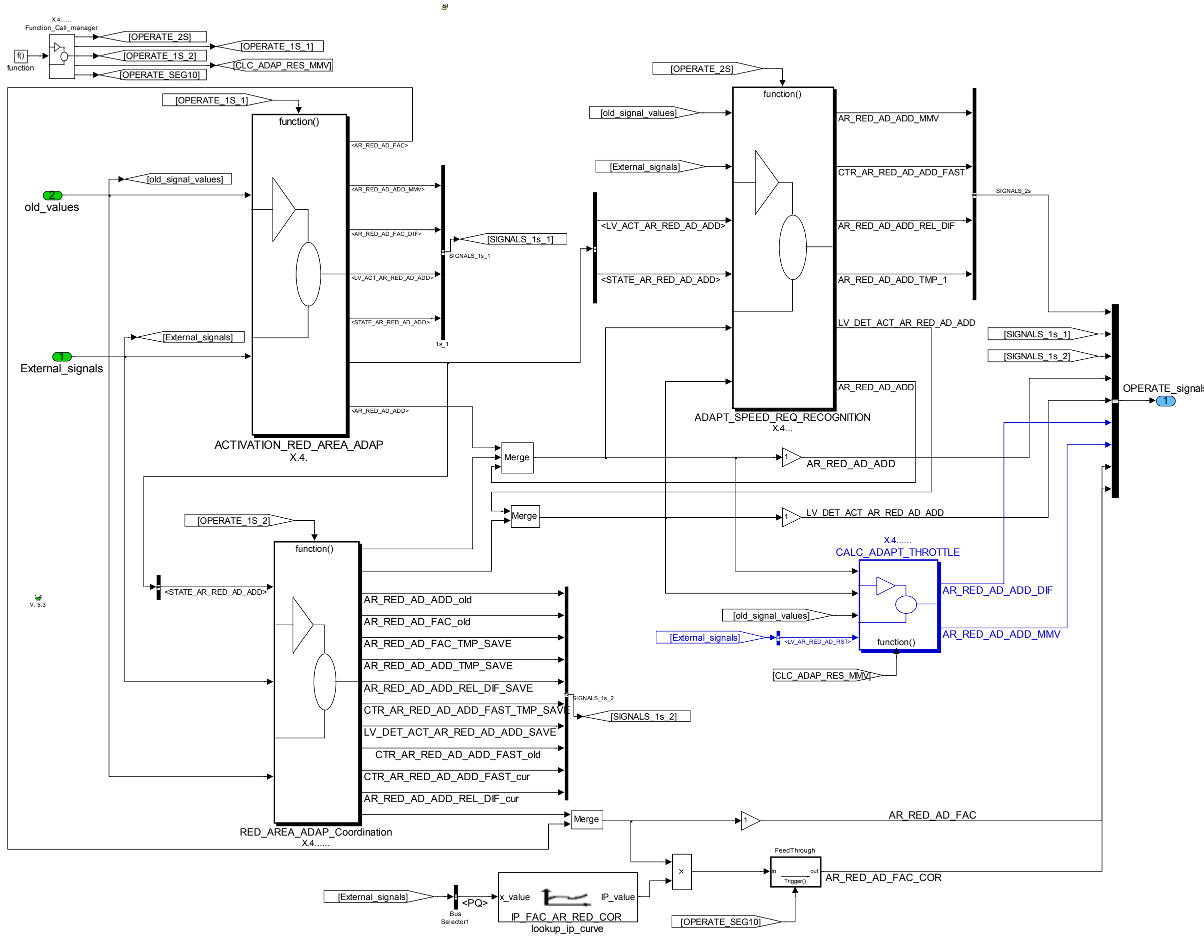


Figure 185:
Path: INSY_ADCPRARRED0/opm

4.15.4.1 FUNCTION_CALL_MANAGER

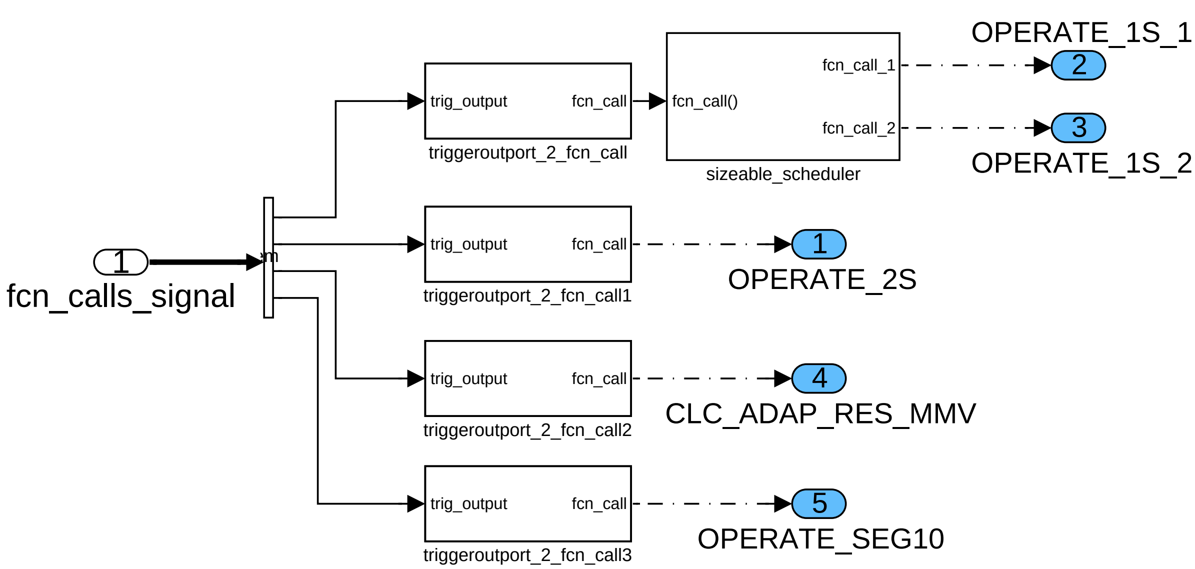


Figure 186:
Path: INSY_ADCPRARRED0/opm/Function_Call_manager

4.15.4.2 ACTIVATION_RED_AREA_ADAP

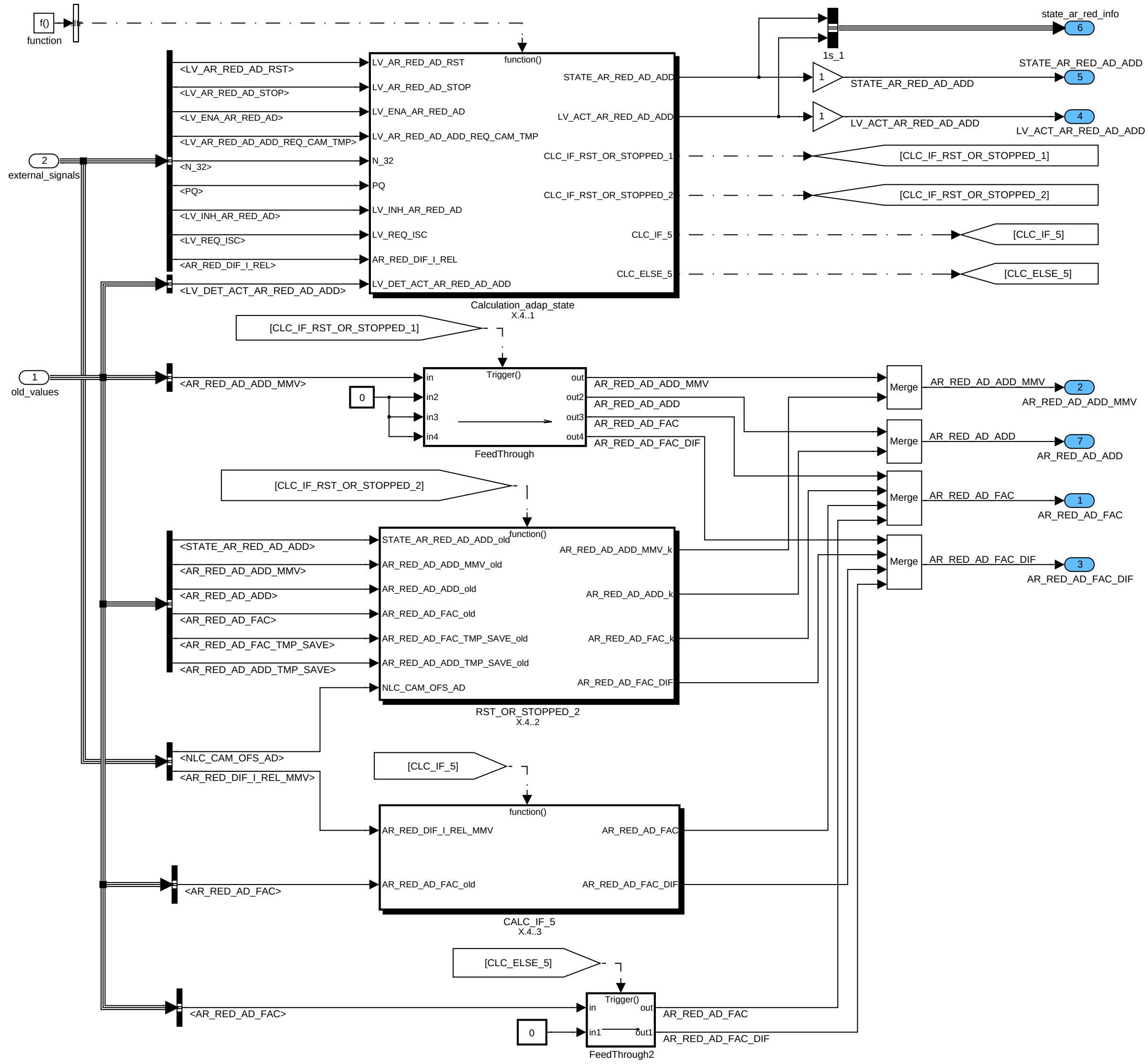


Figure 187:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP

4.15.4.2.1 CALCULATION_ADAP_STATE

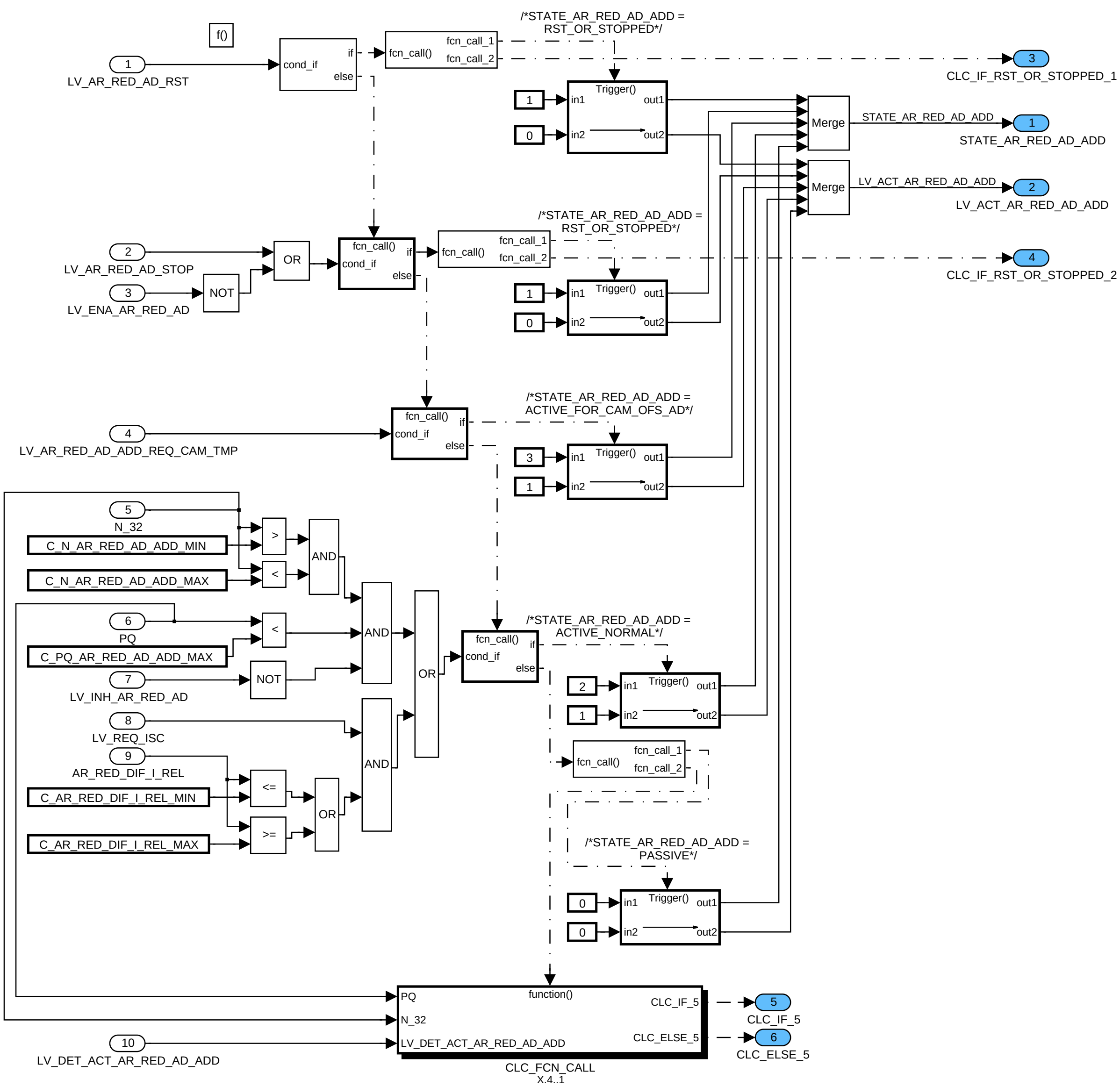


Figure 188:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/Calculation_adap_state

4.15.4.2.1.1 CLC_FCN_CALL

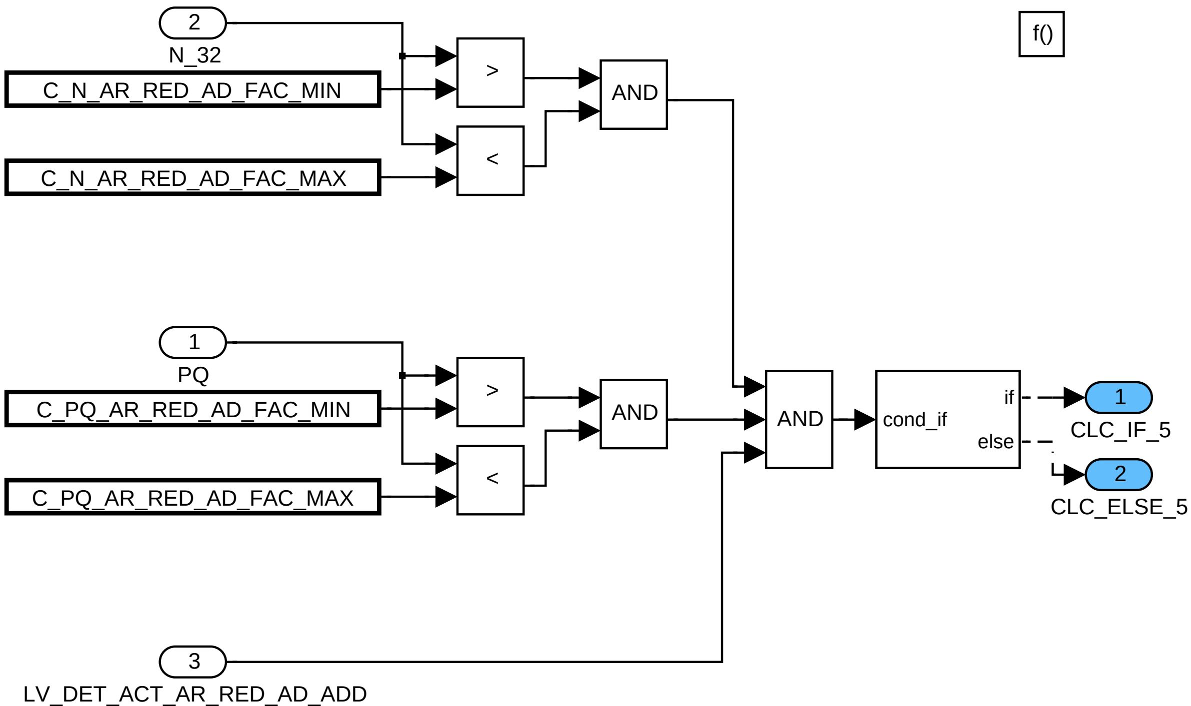


Figure 189:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/Calculation_adap_state/CLC_FCN_CALL

4.15.4.2.2 Coordination of reduced-area adaptation request for CAM_OFS_AD functionality - 1000ms

The bit LV_AR_RED_AD_ADD_REQ_CAM_TMP is used to indicate the status of the adaptation request by the CAM_OFS_AD functionality. Before the adaptation is then actually carried out, the results of the NORMAL Reduced-area adaptation(that is used by the other INSY modules) are saved in temporary variables. When the adaptation for the CAM_OFS_AD functionality has been completed, the results of the previous NORMAL adaptation cycle(saved in the temporary variables) are copied back into the corresponding variables.

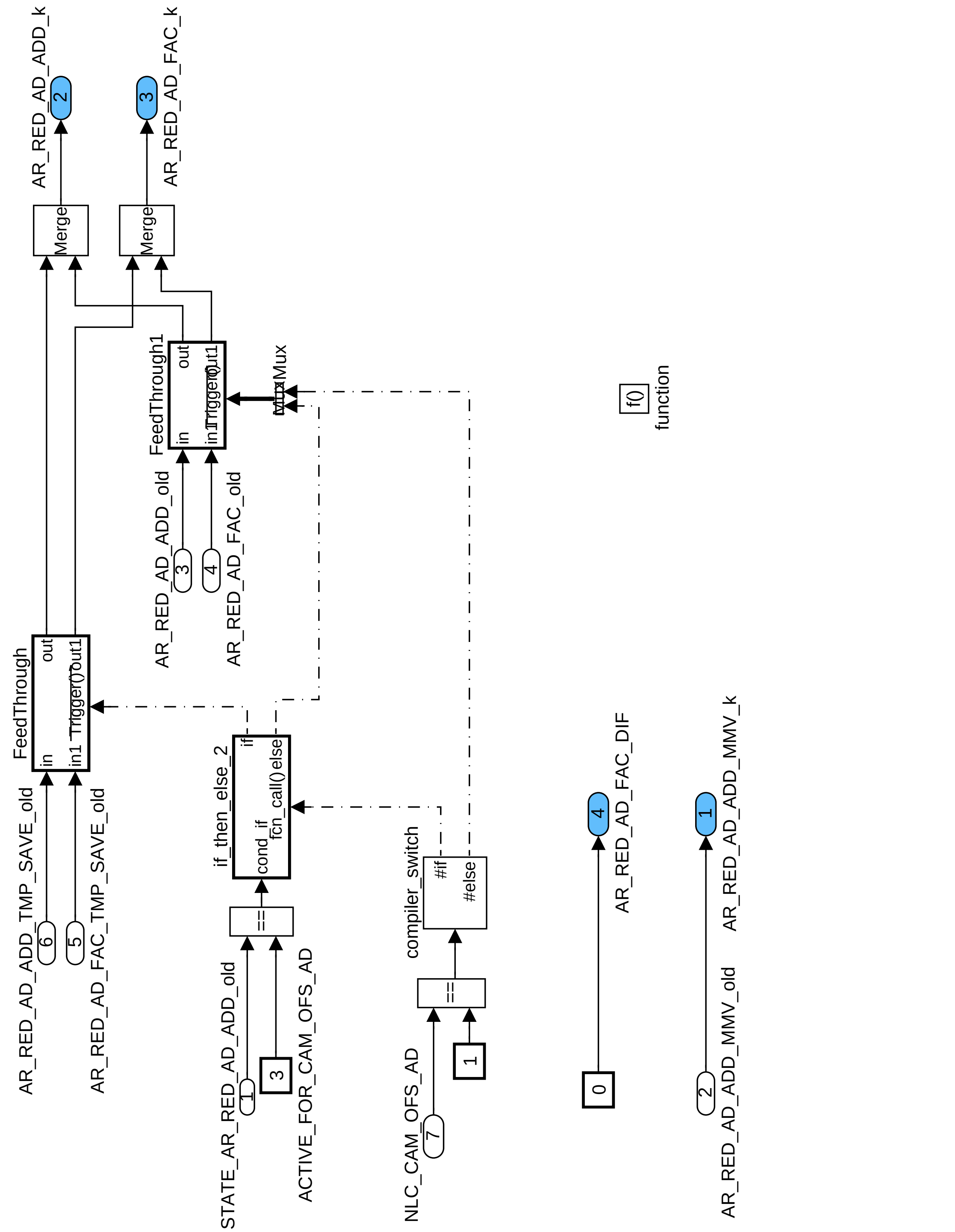


Figure 190:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/RST_OR_STOPPED_2

4.15.4.2.3 CALC_IF_5

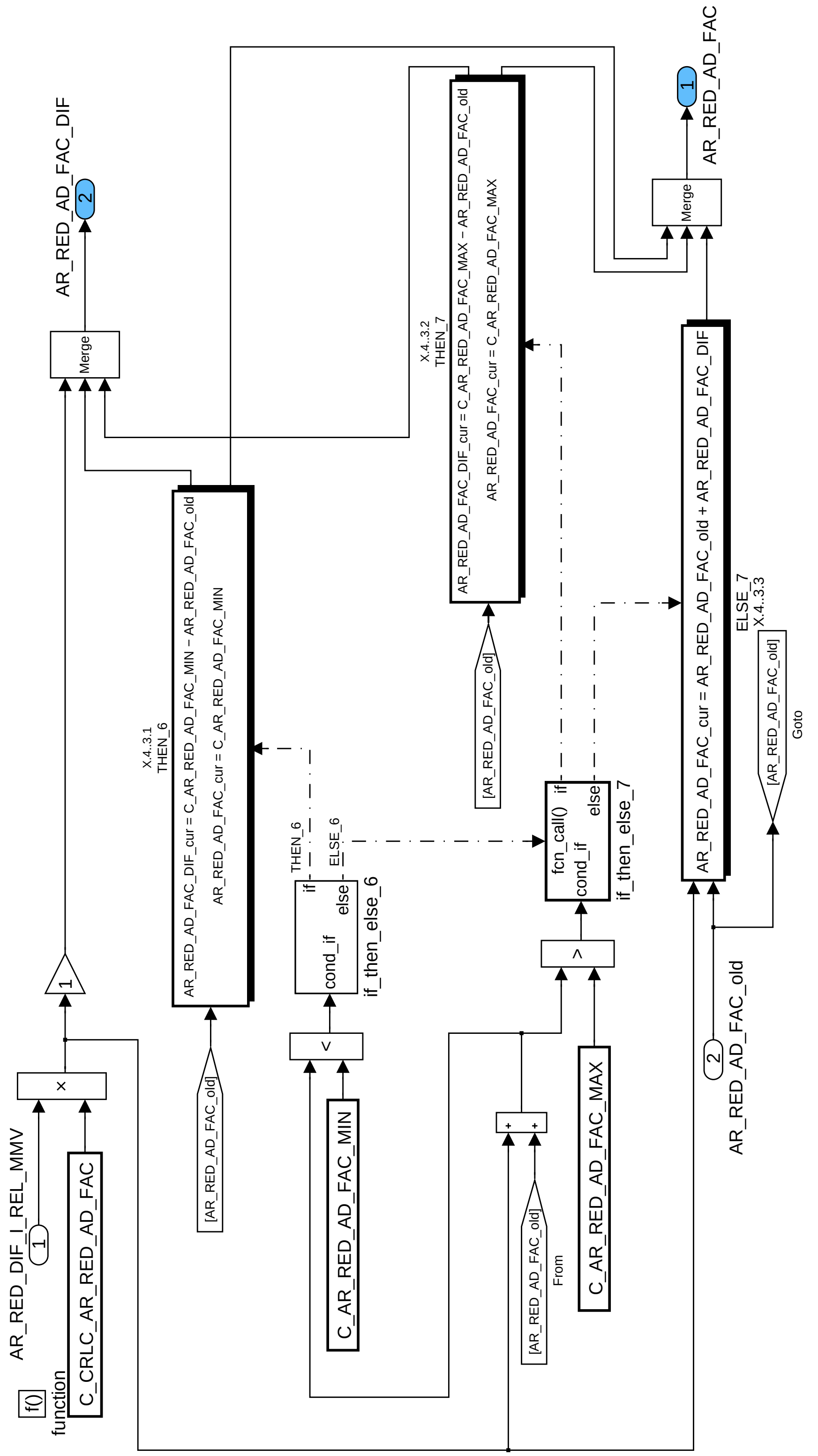


Figure 191:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/CALC_IF_5

4.15.4.2.3.1 THEN_6

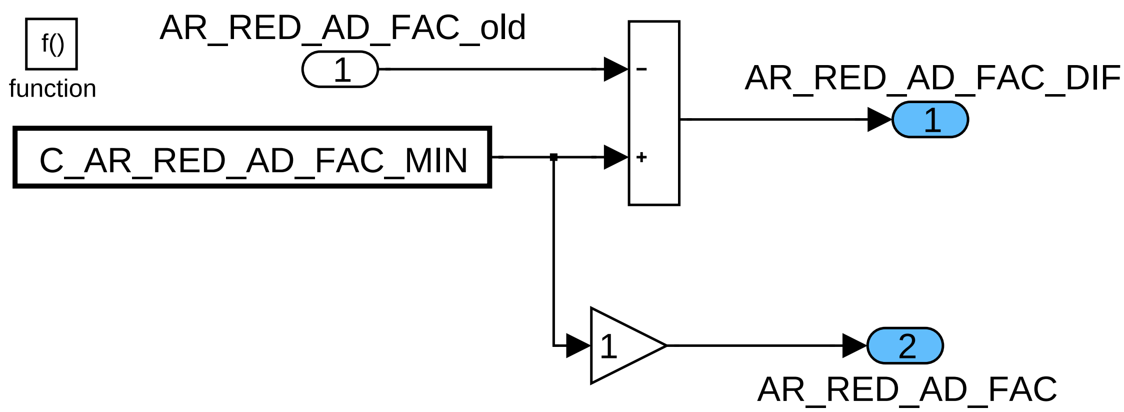


Figure 192:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/CALC_IF_5/THEN_6

4.15.4.2.3.2 THEN_7

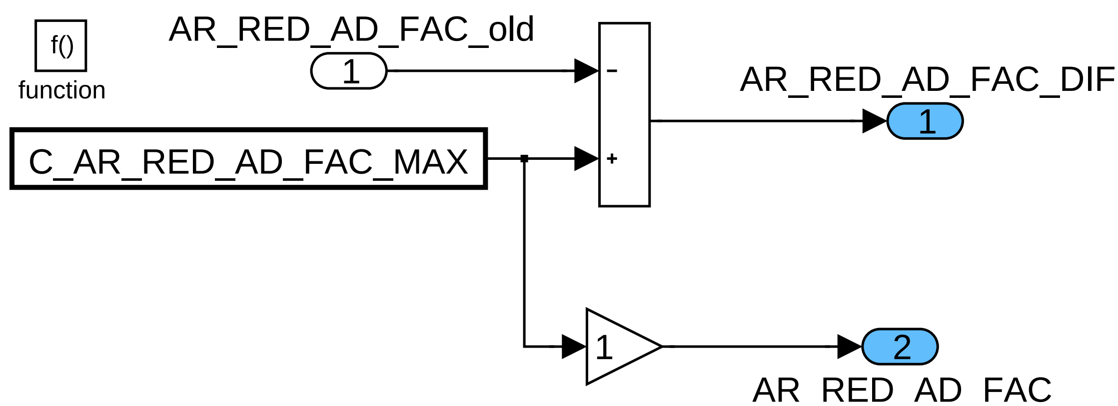


Figure 193:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/CALC_IF_5/THEN_7

4.15.4.2.3.3 ELSE_7

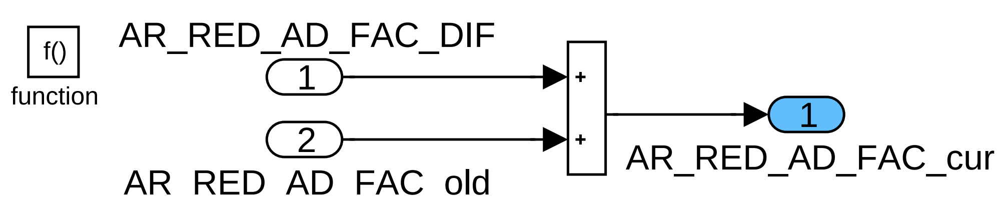


Figure 194:
Path: INSY_ADCPRARRED0/opm/ACTIVATION_RED_AREA_ADAP/CALC_IF_5/ELSE_7

4.15.4.3 Coordination of reduced-area adaptation request for CAM_OFS_AD functionality - 1000ms

The bit LV_AR_RED_AD_ADD_REQ_CAM_TMP is used to indicate the status of the adaptation request by the CAM_OFS_AD functionality. Before the adaptation is then actually carried out, the results of the NORMAL Reduced-area adaptation(that is used by the other INSY modules) are saved in temporary variables. When the adaptation for the CAM_OFS_AD functionality has been completed, the results of the previous NORMAL adaptation cycle(saved in the temporary variables) are copied back into the corresponding variables.

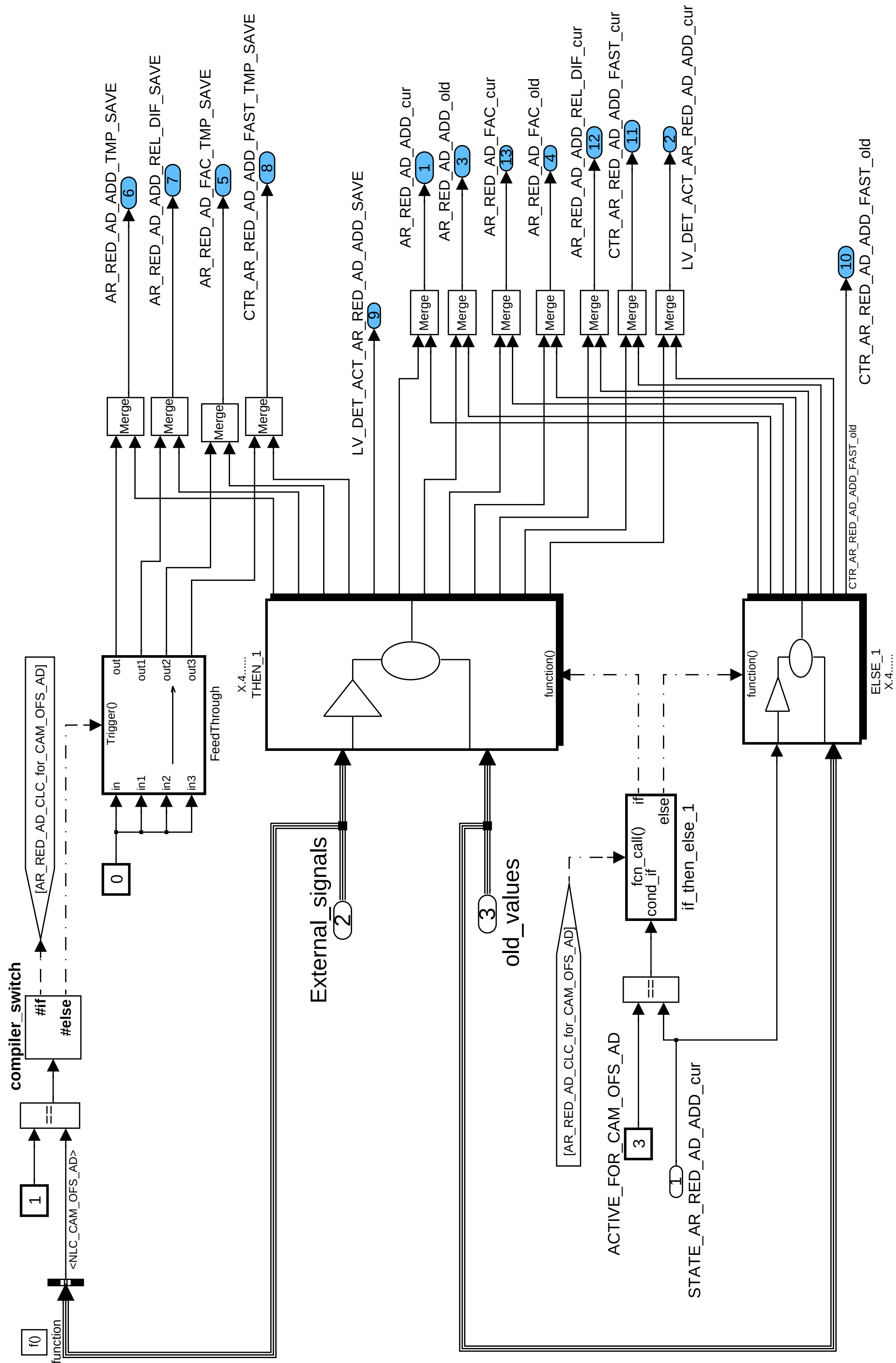


Figure 195:
Path: INSY_ADCPRARRED0/opm/RED_AREA_ADAP_Coordination

4.15.4.3.1 THEN_1

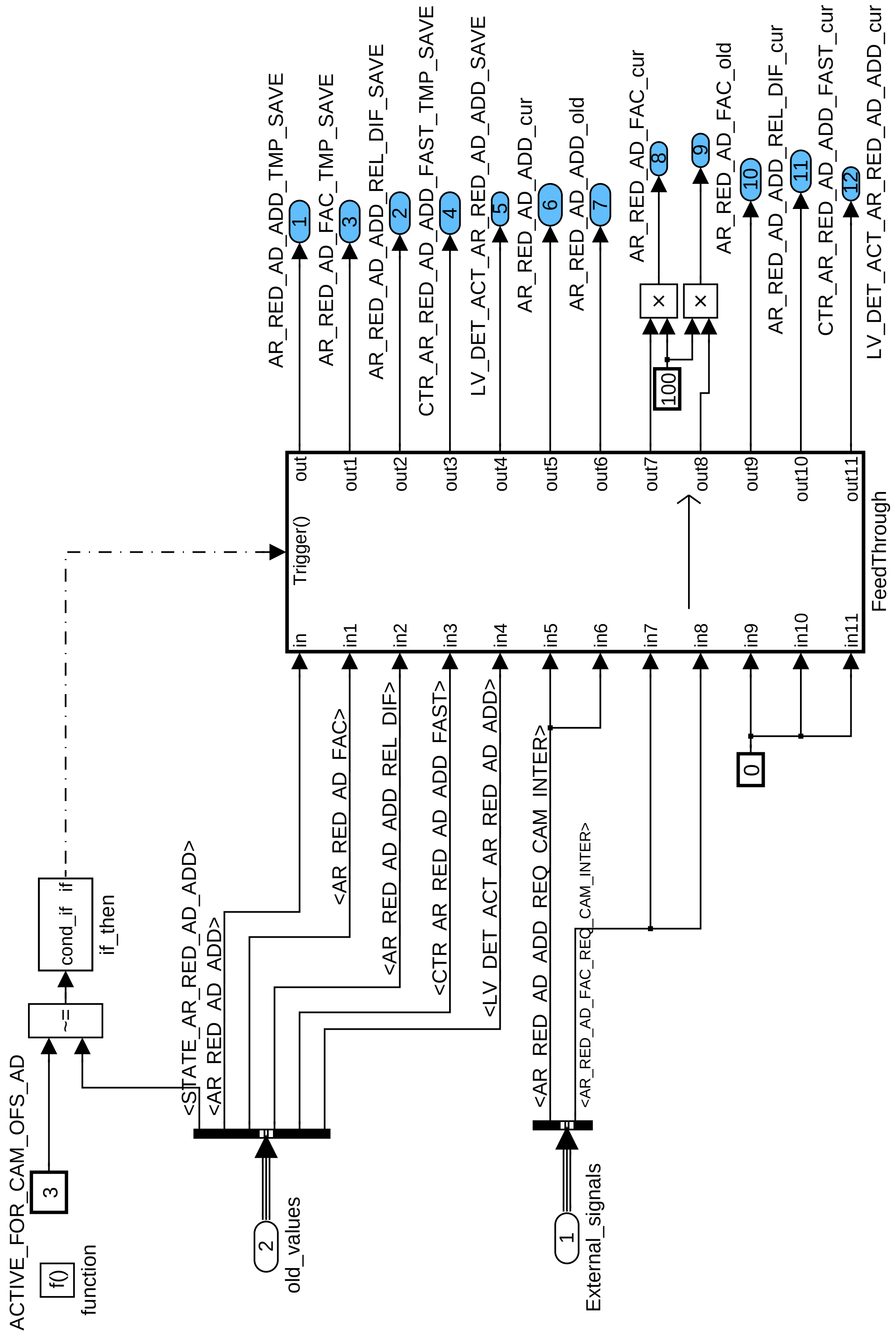


Figure 196:
Path: INSY_ADCPRARRED0/opm/RED_AREA_ADAP_Coordination/THEN_1

4.15.4.3.2 ELSE_1

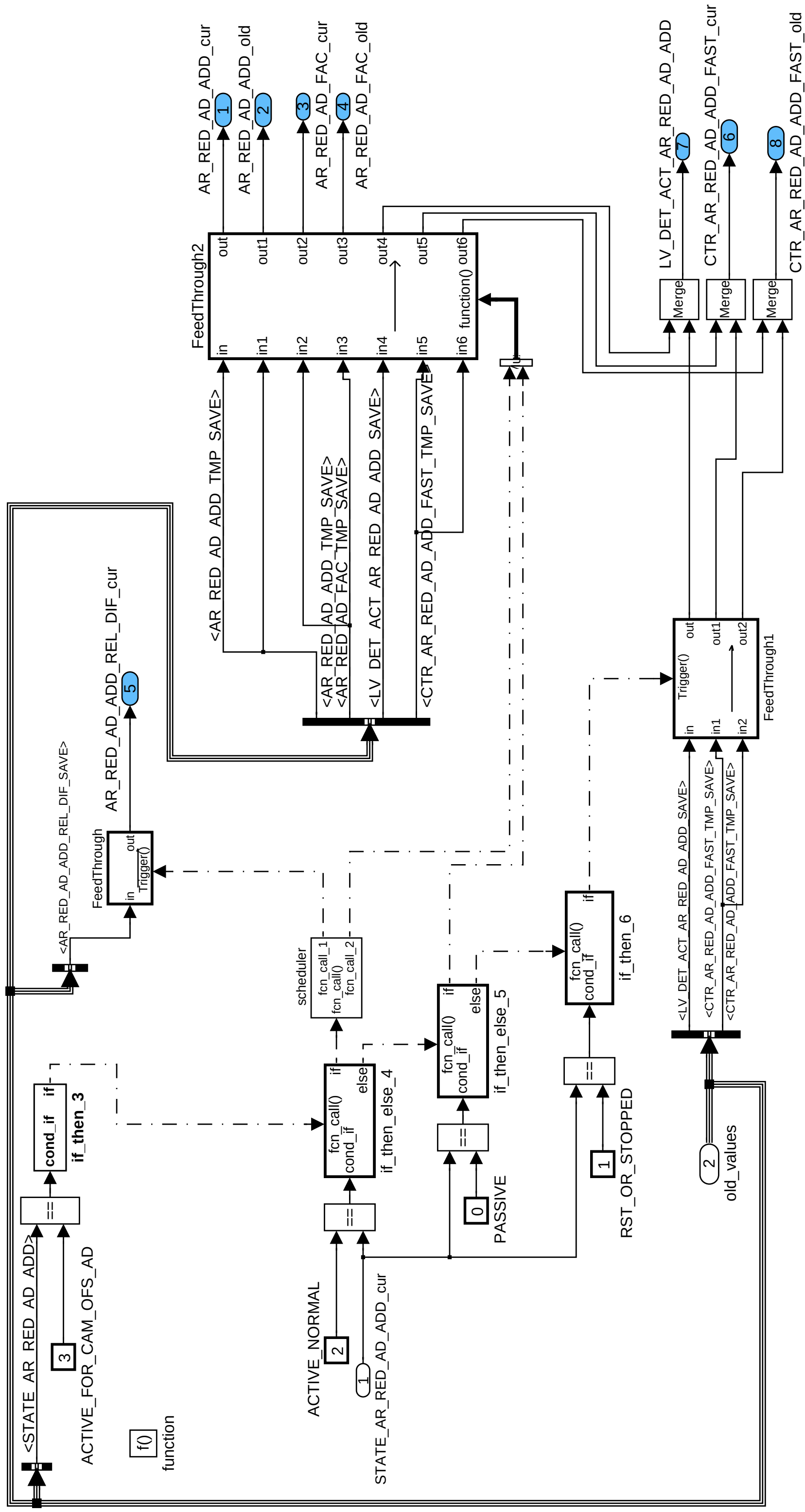


Figure 197:

Path: INSY_ADCPRARRED0/opm/RED_AREA_ADAP_Coordination/ELSE_1

4.15.4.4 Adaptation speed request recognition

The adaptation of the reduced throttle area can be performed at two different speed levels, depending on the flag LV_TPS_AD_REQ, indicating a throttle adaptation request. If the throttle adaptation is requested (LV_TPS_AD_REQ = 1), the adaptation speed will be increased for a calibratable number of cycles (counter CTR_AR_RED_AD_ADD_FAST).
In the post-operating phase of the ECU, the counter CTR_AR_RED_AD_ADD_FAST is saved in the non-volatile RAM of the ECU. To detect that the fast adaptation of additive reduced area has been learnt, a flag is defined in order to fix the order to make the several adaptations and to prevent the multiplicative adaptation from taking place before the additive adaptation. It must be noted that the fast adaptation is not used when STATE_AR_RED_AD_ADD = ACTIVE_FOR_CAM_OFS_AD.
It is important to ensure, that AR_RED_AD_ADD_REL_DIF is set to 0 when no AR_RED_AD_ADD adaptation is performed!. When the counter CTR_AR_RED_ADD_FAST decreases to zero (only at the first time), the calculation of AR_RED_AD_ADD_REL_DIF will be switched over to the c fficient C_AR_RED_AD_COR_CRLC and at the same time the moving mean value AR_RED_AD_ADD_MMV will take the value from AR_RED_AD_ADD.

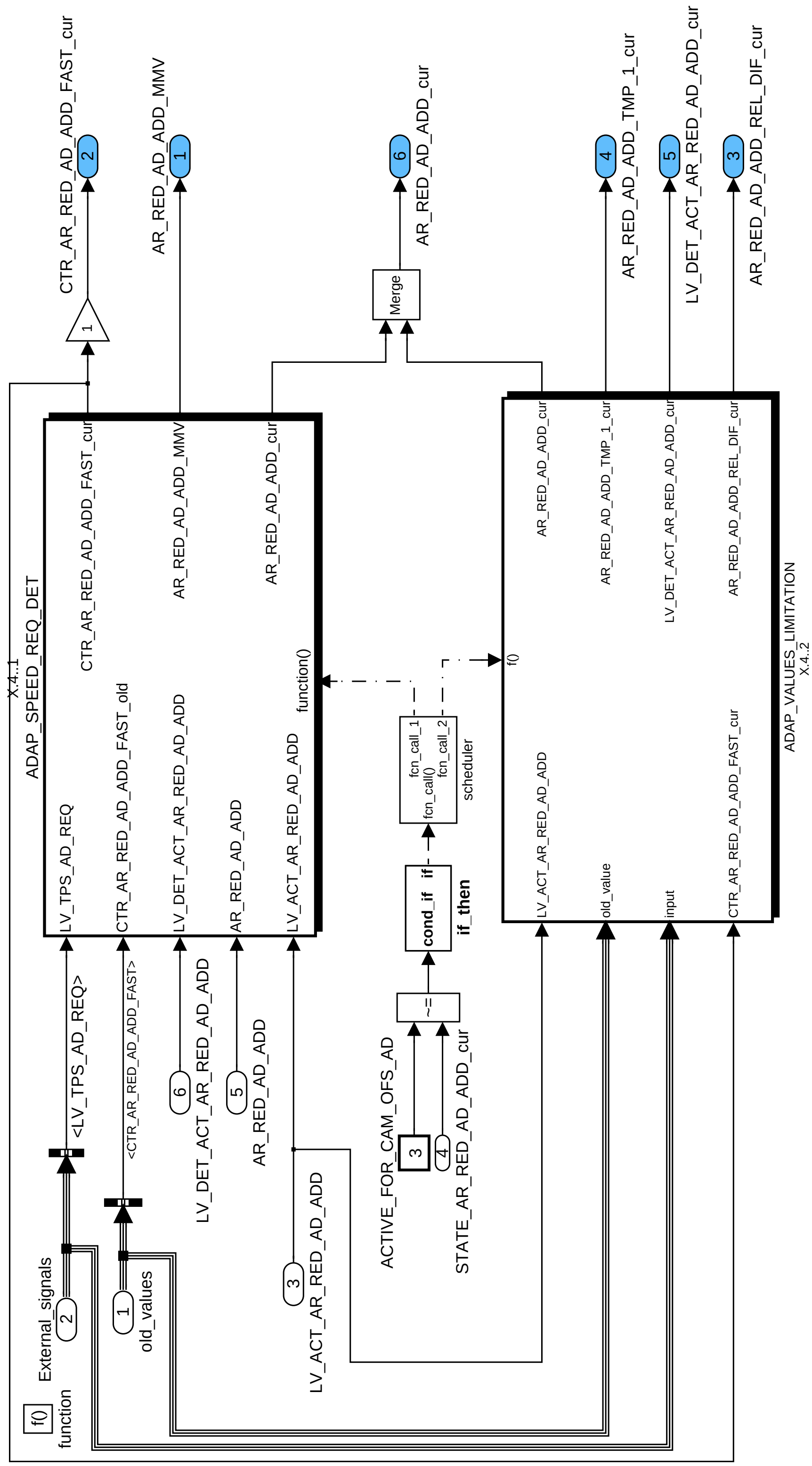


Figure 198:

Path: INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION

4.15.4.4.1 ADAP_SPEED_REQ_DET

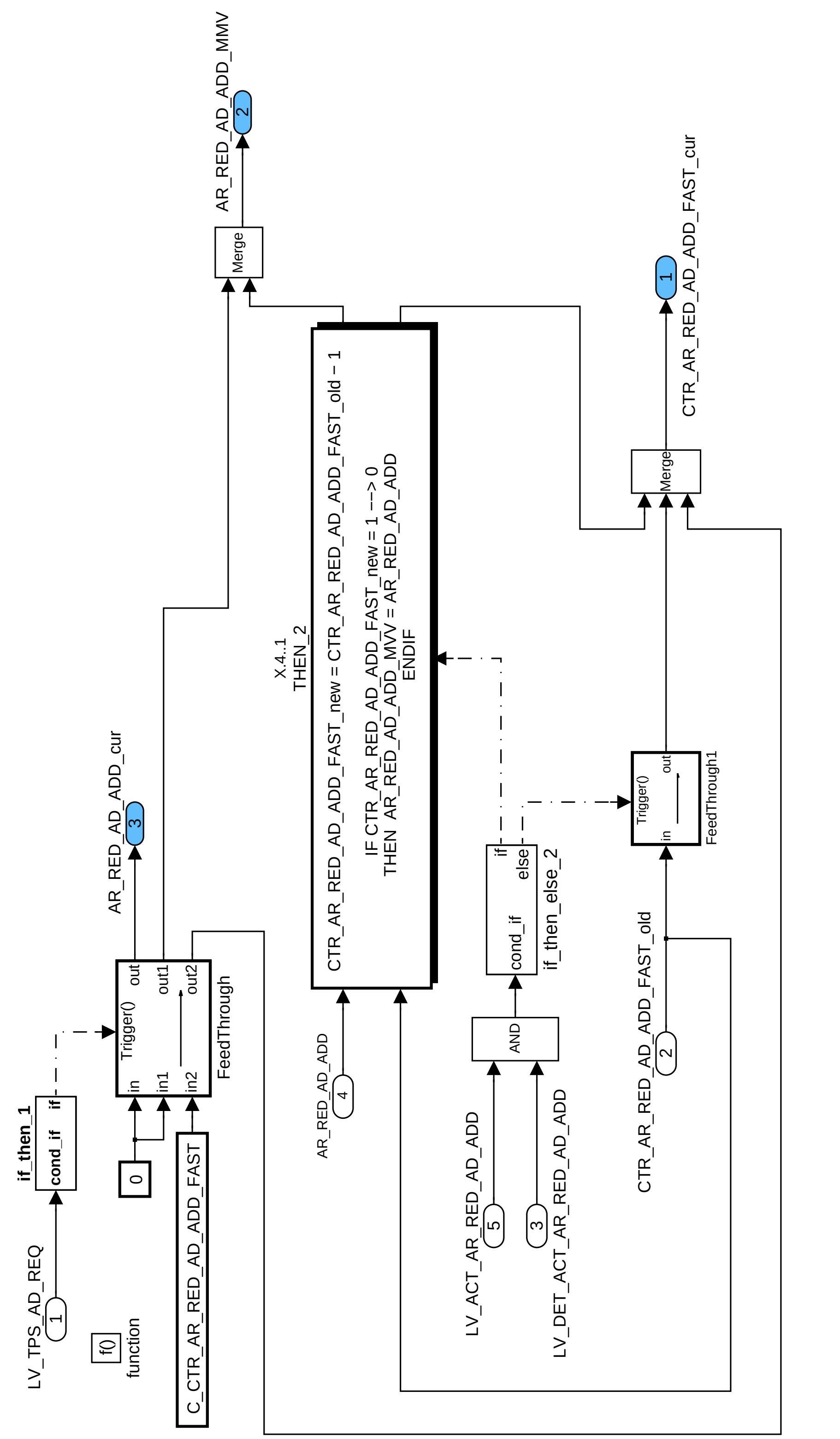


Figure 199:
Path: INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_SPEED_REQ_DET

4.15.4.4.1.1 THEN_2

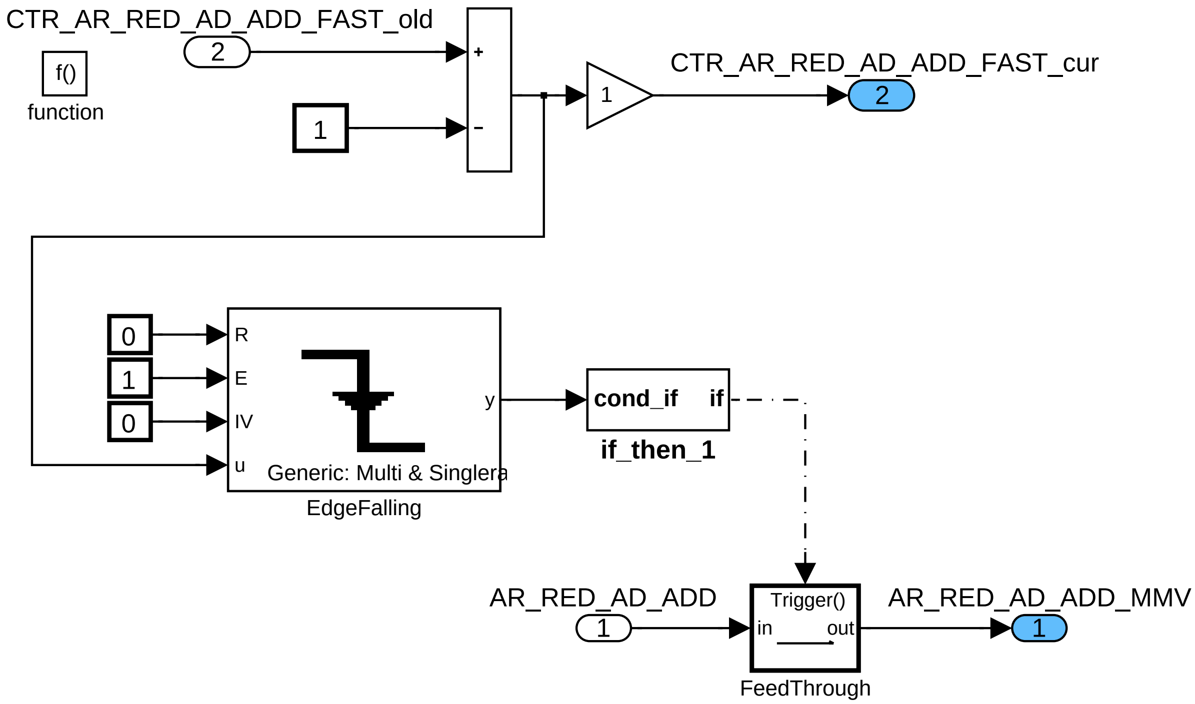


Figure 200:
Path: INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_SPEED_REQ_DET/THEN_2

4.15.4.4.2 ADAP_VALUES_LIMITATION

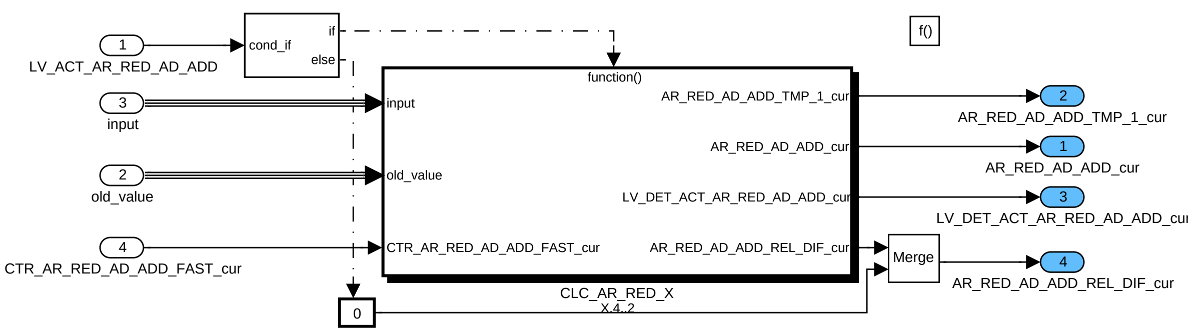


Figure 201:
Path: INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION

4.15.4.4.2.1 CLC_AR_RED_X

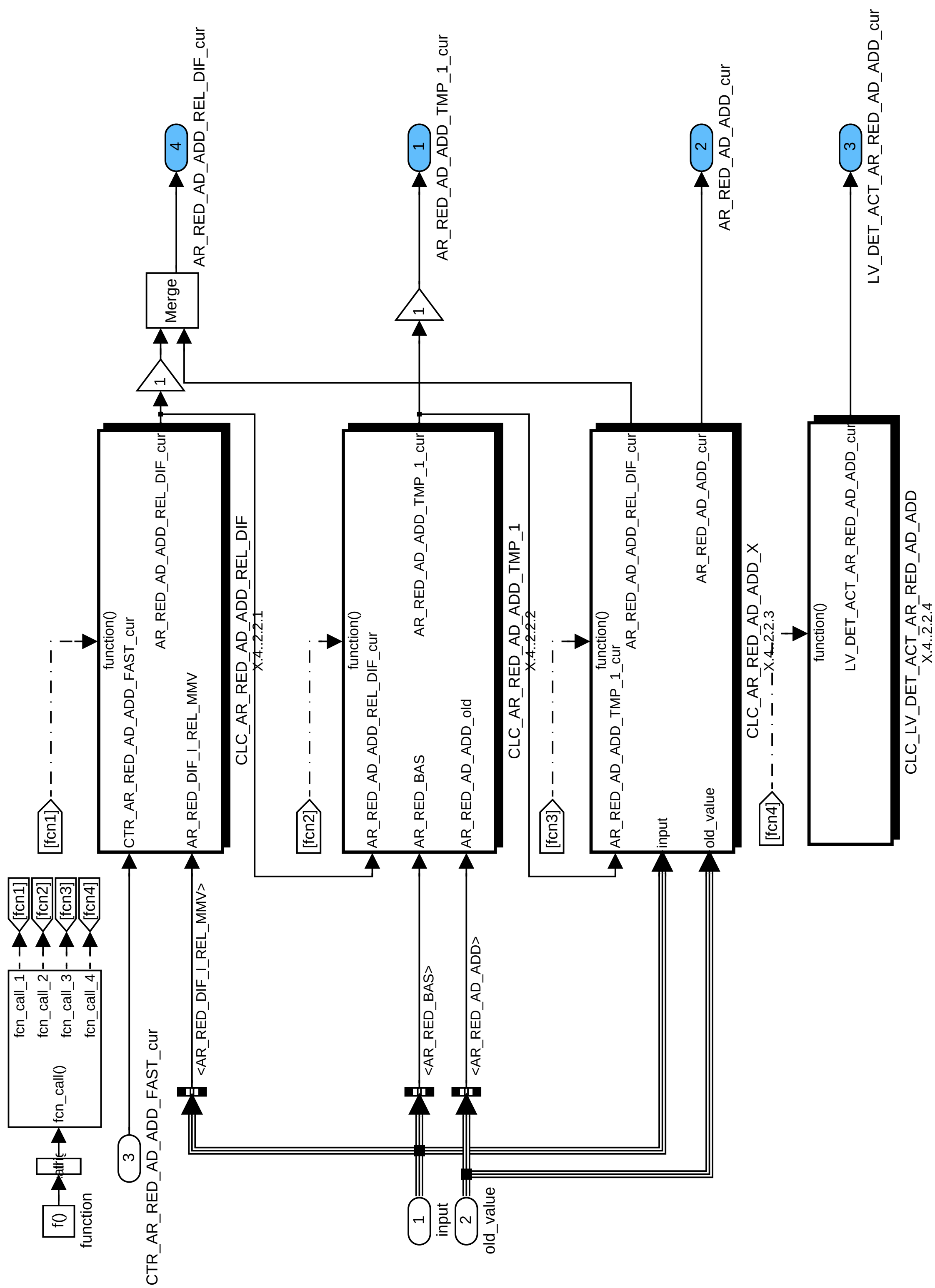


Figure 202:
Path: INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X

4.15.4.4.2.1.1 CLC_AR_RED_AD_ADD_REL_DIF

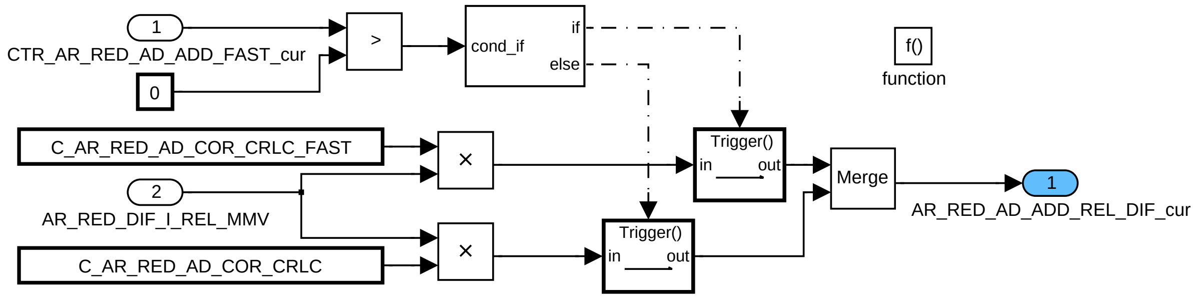


Figure 203:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_REL_DIF

4.15.4.4.2.1.2 CLC_AR_RED_AD_ADD_TMP_1

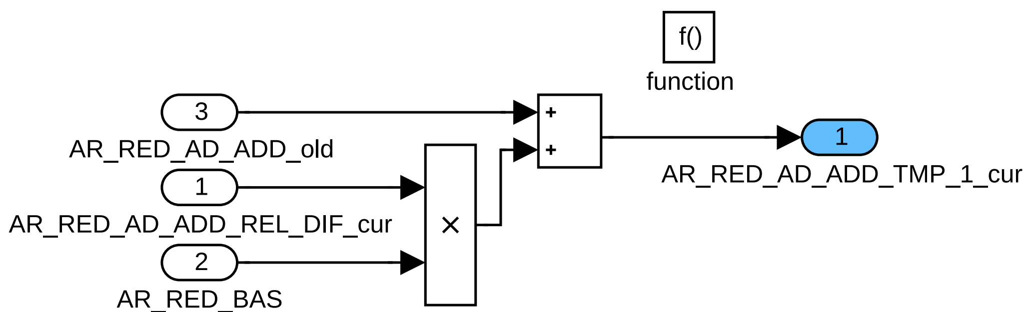


Figure 204:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_TMP_1

4.15.4.4.2.1.3 CLC_AR_RED_AD_ADD_X

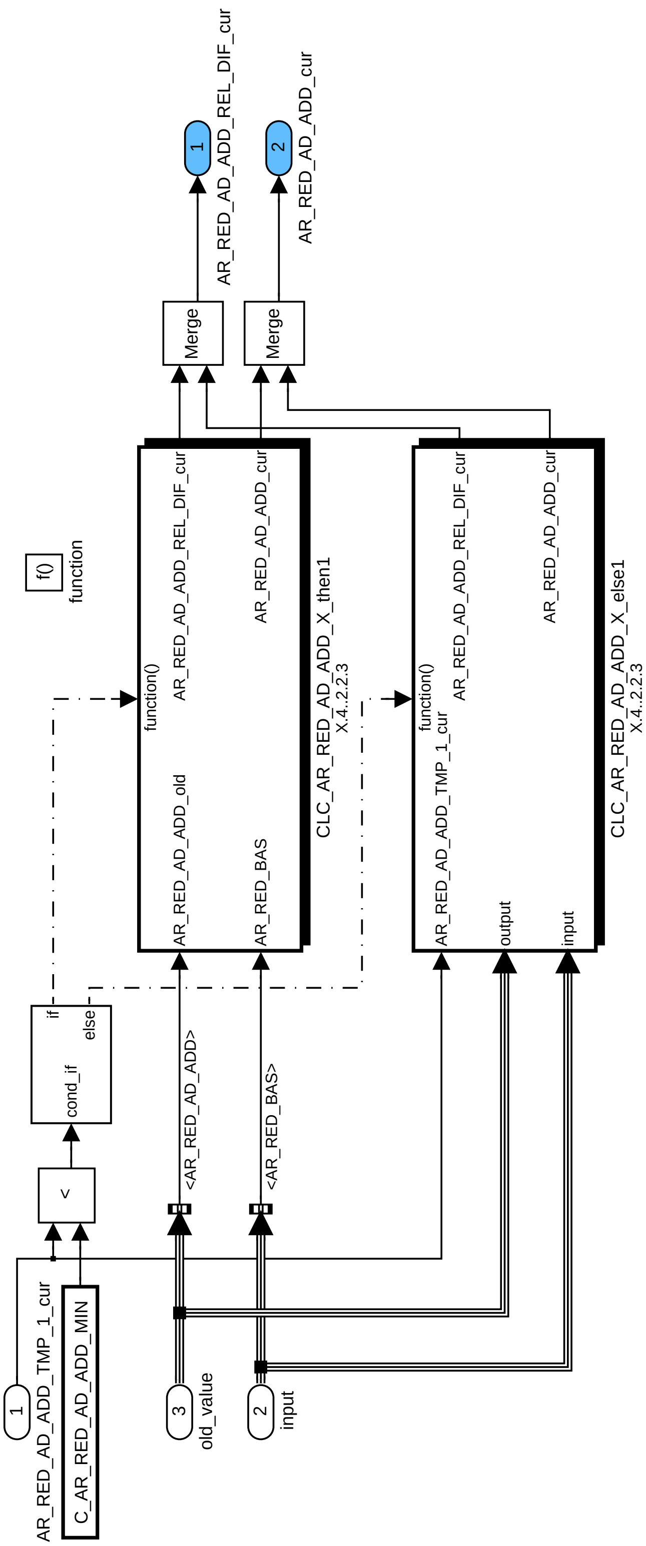


Figure 205:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_X

4.15.4.4.2.1.3.1 CLC_AR_RED_AD_ADD_X_THEN1

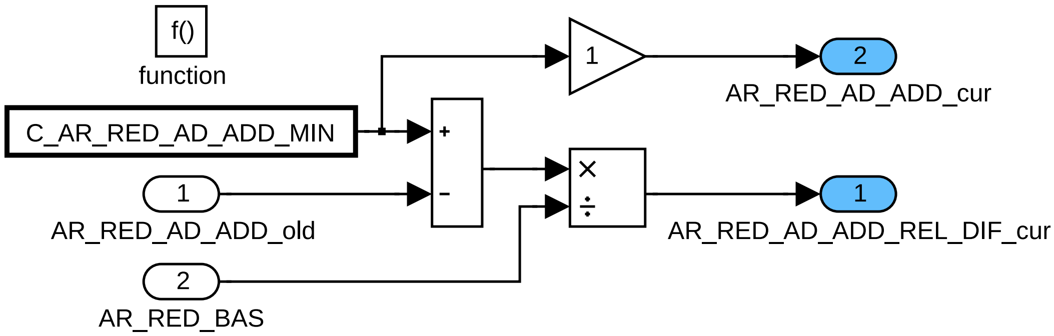


Figure 206:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_X/CLC_AR_RED_AD_ADD_X_then1

4.15.4.4.2.1.3.2 CLC_AR_RED_AD_ADD_X_ELSE1

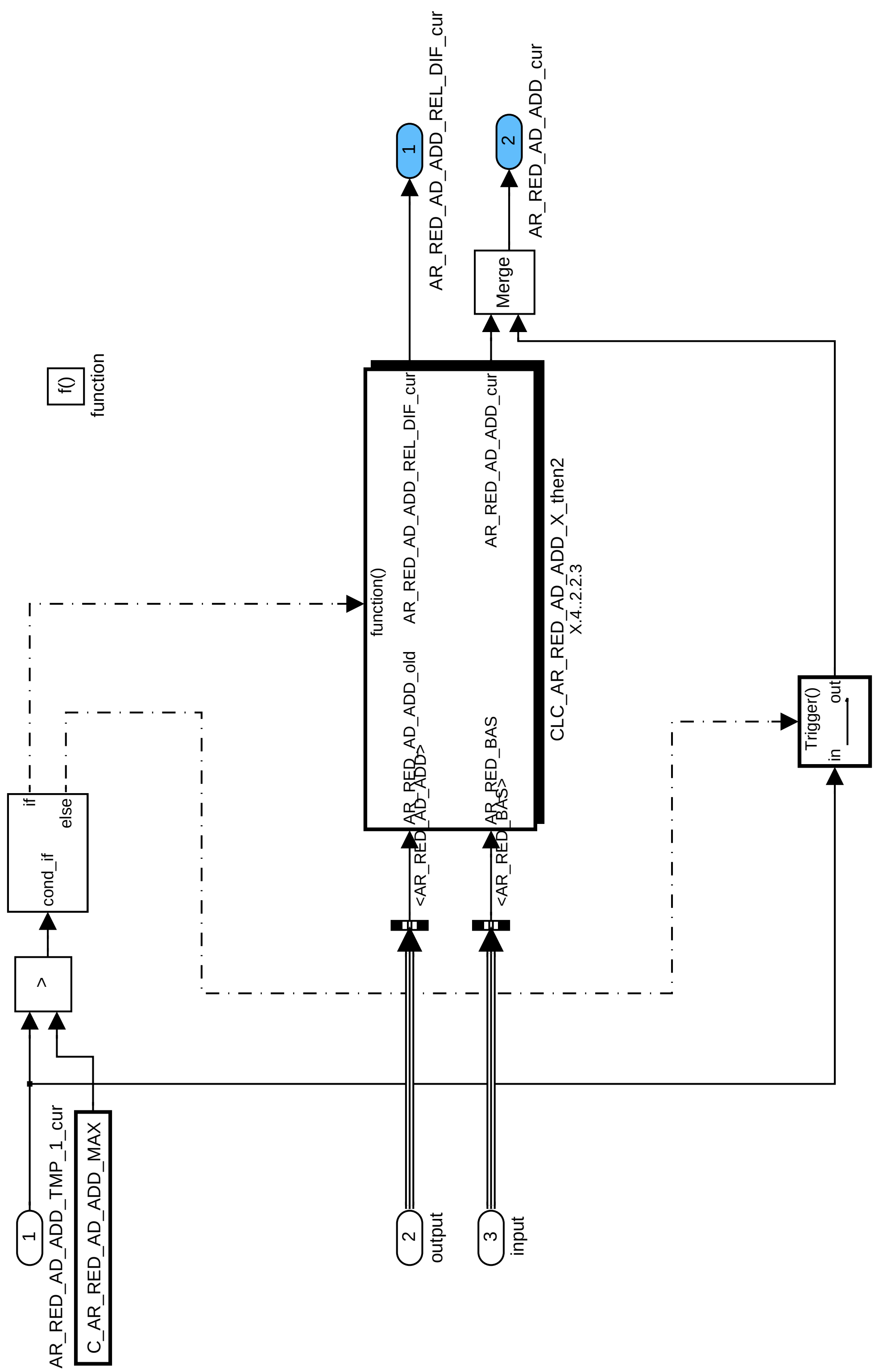


Figure 207:

Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_X/CLC_AR_RED_AD_ADD_X_else1

4.15.4.4.2.1.3.2.1 CLC_AR_RED_AD_ADD_X_THEN2

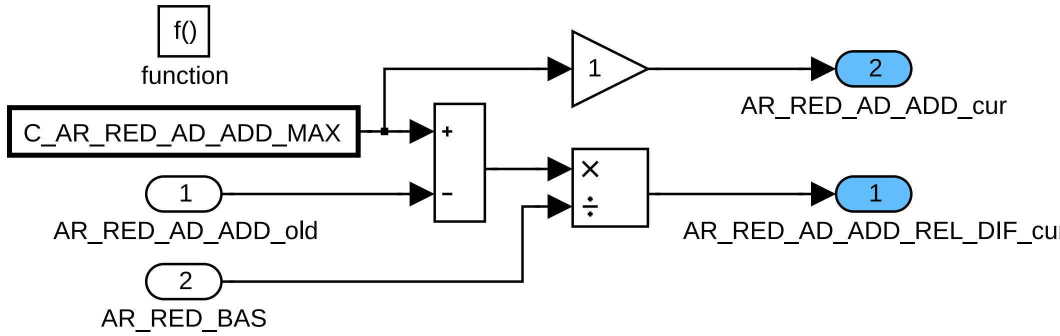


Figure 208:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_AR_RED_AD_ADD_X/CLC_AR_RED_AD_ADD_X_else1/CLC_AR_RED_AD_ADD_X_then2

4.15.4.4.2.1.4 CLC_LV_DET_ACT_AR_RED_AD_ADD

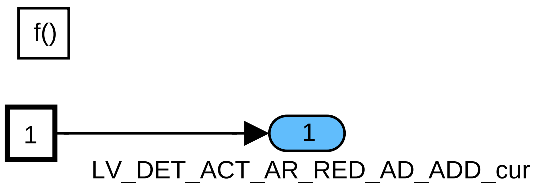


Figure 209:
Path:
INSY_ADCPRARRED0/opm/ADAPT_SPEED_REQ_RECOGNITION/ADAP_VALUES_LIMITATION/CLC_AR_RED_X/CLC_LV_DET_ACT_AR_RED_AD_ADD

4.15.4.5 Calculation of the moving mean value from the adaptation of the throttle

/*This chapter is only calculated in the power latch phase i.e. transition to PWL, called before adaptation values are stored in NVMY*/
The calculation of the moving mean value is only executed if at least one new adaptation value AR_RED_AD_ADD has been calculated in the current DC (LV_ACT_AR_RED_AD_ADD = 1). Otherwise the moving mean value stored in the EPROM remains unchanged. It has to be taken into account that in this case one has to abdicate the using of the congregating filter.
The averaging constant CRLC_AR_RED_AD_ADD_MMV to calculate the mean value depends on the adaptation demand AR_RED_AD_ADD_DIF from a driving cycle. Therefore the value AR_RED_AD_ADD_MMV in case of natural changes (pollution) will follow very fast the value AR_RED_AD_ADD (leakage) and in case of fast changes the value of the adaptation will be followed slowly.This adaptation is learned if at least once the adaptation of the ambient pressure at open throttle is learned
In case of erasing the adaptation values it is necessary to put the values from AR_RED_AD_ADD_MMV and from AR_RED_AD_ADD to zero and the counter CTR_AR_RED_AD_ADD_FAST has to be initialized with the value of the c fficient C_CTR_AR_RED_AD_ADD_FAST. It must be noted here that this erase-function is specific for each project.

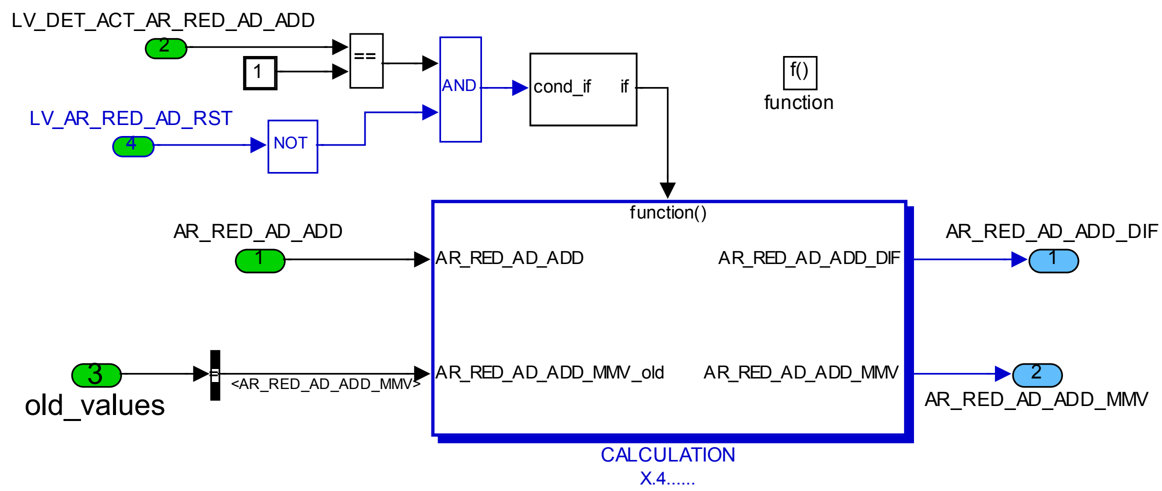


Figure 210:
Path: INSY_ADCPRARRED0/opm/CALC_ADAPT_THROTTLE

4.15.4.5.1 CALCULATION

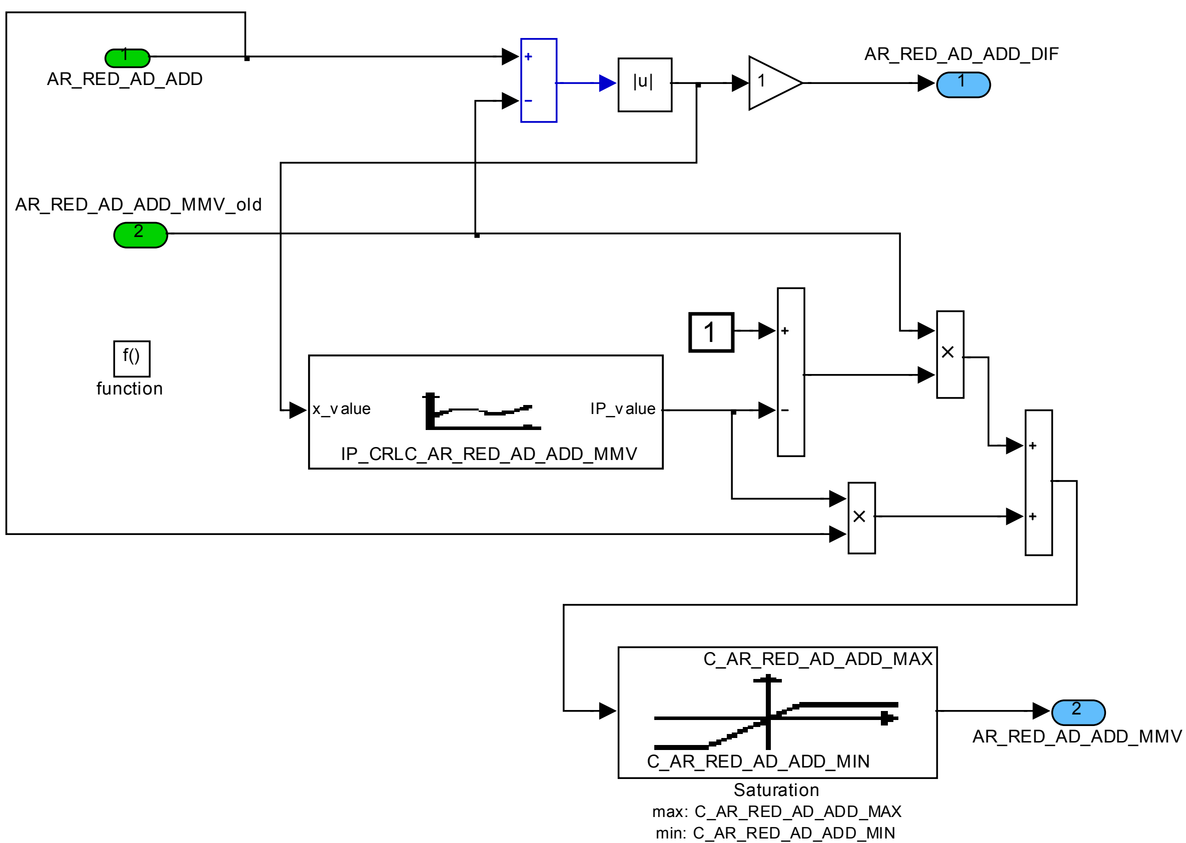


Figure 211:
Path: INSY_ADCPRARRED0/opm/CALC_ADAPT_THROTTLE/CALCULATION

4.16 Ambient Pressure Adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AMP	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Ambient pressure (measured or adapted)					
LV_END_AMP_AD	O/V	0...1H	0...1	1	-
End of ambient pressure adaptation at wide open throttle (if a load sensor is available)					
AMP_AD	O/V/S	0...FFFFH	0...5.434E+3	0.08291752	hPa
Ambient pressure (adapted)					
AMP_AD_DIF	O/V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Change of the adapted ambient pressure					
STATE_AMP_AD	O/V	0H 1H 2H 3H	NO_AMP_AD AMP_AD_VIA_MODEL WAITING_FOR_AMP_AD_VIA_MAP_MES AMP_AD_VIA_MAP_MES	1	-
State of ambient pressure adaptation					
CTR_DLY_AMP_AD	O/V	0...FFH	0...255	1	s
Time delay to inhibit AMP learning after ES					
T_CTR_AMP_AD_CMPL	O/V/S	0...FFFFH	0...6.5535E+4	1	s
Time counter indicating when the last succesfull completion of the adpatation was detected					
CTR_AR_RED_AD_ADD_ENA	V/S	0...FFH	0...255	1	-
Counter to allow the reduced area adaptation not before AMP is adapted					
T_DLY_AMP_AD_MAP_MES	V	0...FFH	0...25.5	0.1	s
Delay time counter for ambient pressure adaptation via MAP_MES					
VS_TMP_AMP_AD_PL	V	0...FFH	0...255	1	km/h
Temporary vehicle speed to detect decelerating vehicle					
LV_AMP_AD_MAP_MES_ENA_RST	V	0...1H	0...1	1	-
Logical variable to enable AMP adaptation via MAP_MES at first valid tooth					
T_DLY_AMP_AD_PUT_MES	V	0...FFH	0...25.5	0.1	s
Delay time counter for ambient pressure adaptation via PUT_MES					
SEG_CTR_AMP_AD_PL	V	0...FFFFH	0...6.5535E+4	1	-
Segment counter for ambient pressure adaptation during part load					
SEG_CTR_AMP_AD	V	0...FFFFH	0...6.5535E+4	1	-
Segment counter for ambient pressure adaptation					
MAF_KGH_MMV	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Moving mean value of the mass air flow					
AR_RED_BAS_MMV	V	0...FFFFH	0...58.592855	8.9407E-4	cm²
Moving mean value of the basic reduced area					
AMP_ESTIM	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Estimated ambient pressure					

Input data:

LV_ST	AMP_MES	AR_RED_AD_ADD	AR_RED_AD_FAC
AR_RED_BAS	FAC_LAM_MV_MMV[NC_CBK_EX_NR]	LV_AMP_AD_RST	LV_AMP_AD_STOP

LV_ENA_AMP	LV_ENA_AMP_AD_LAMB	LV_ENA_AMP_AD_PQ	LV_MAF_CTL
LV_MAP_CTL	LV_PUC	MAF_FAC_OFS	MAF_FAC_SLOP
MAF_KGH	MAF_MDL_CON_1	MAP	MAP_DRV1
MAP_MES	MAP_MES_MMV	N_32	NC_CBK_EX_NR
TCO	PQ	PUT_MDL_DIF_I_MMV	SEG_INC
TPS_AV	LV_INH_AMP_AD	LV_ENA_MAP	AMP_DEC
LV_ENA_AMP_AD_MAP_MES	LV_ENA_PUT	LC_TCHA_CONF	NC_CHRG_CONF
VS	LV_IS	LV_AT	LV_PU
FLOW_ENG	PUT_MES		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_AD_COR_CRLC	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for ambient pressure learning					
C_AMP_AD_DIF_NEG_AT_IS	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Negative correction of the ambient pressure adaptation during part load for A/T for idle speed					
C_AMP_AD_DIF_NEG_AT_PUC	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Negative correction of the ambient pressure adaptation during part load for A/T for pull fuel cutoff (PUC)					
C_AMP_AD_DIF_NEG_MT	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Negative correction of the ambient pressure adaptation during part load for M/T					
C_AMP_AD_DIF_POS_AT_IS	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Positive correction of the ambient pressure adaptation during part load for A/T for idle speed					
C_AMP_AD_DIF_POS_AT_PUC	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Positive correction of the ambient pressure adaptation during part load for A/T for pull fuel cutoff (PUC)					
C_AMP_AD_DIF_POS_MT	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Positive correction of the ambient pressure adaptation during part load for M/T					
C_AMP_AD_PL_HYS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Hysteresis for ambient pressure adaptation during part load					
C_AMP_DIF_TI_LAM_NEG	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Negative correction of the ambient pressure adaptation with lambda controller					
C_AMP_DIF_TI_LAM_POS	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Positive correction of the ambient pressure adaptation with lambda controller					
C_AMP_HYS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Hysteresis for ambient pressure adaptation via MAP_MES					
C_AMP_INI	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Initial value for the ambient pressure					
C_AMP_MAX	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum value for ambient pressure					
C_AMP_MIN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum value for ambient pressure					
C_CRLC_AMP_AD	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for ambient pressure adaptation					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_AMP_AD_MAP_MES	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant ambient pressure adaptation via map sensor					
C_CRLC_AR_RED_BAS_AMP_AD_PL	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for moving mean value calculation of AR_RED_BAS					
C_CRLC_MAF_KGH_AMP_AD_PL	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for moving mean value calculation of MAF_KGH					
C_CTR_AR_RED_AD_ADD_ENA	1	0...FFH	0...255	1	-
Counter to allow the adaptation of the aditive reduced area after learning the ambient pressure					
C_CTR_DLY_AMP_AD	1	1...FFH	1...255	1	s
Time delay to inhibit AMP learning after ES					
C_MAF_ENG_AMP_AD_MAX	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Maximum threshold to inhibit ambient pressure adaptation via PUT_MES					
C_MAP_DRV1_MAX_AMP_AD_PL	1	0...7FFFH	0...82.9149945	0.00253044	hPa/ms
Maximum value of the first derivative of manifold pressure for ambient pressure adaptation part load					
C_N_AMP_AD_MAX	1	0...FFH	0...8.16E+3	32	rpm
Maximum engine speed for ambient pressure learning					
C_N_AMP_AD_MIN	1	0...FFH	0...8.16E+3	32	rpm
Minimum engine speed for ambient pressure learning					
C_N_MIN_AMP_AD_IS	1	0...FFH	0...8.16E+3	32	rpm
Minimum value of the engine speed for ambient pressure adaptation in IS					
C_N_MIN_AMP_AD_PUC	1	0...FFH	0...8.16E+3	32	rpm
Minimum value of the engine speed for ambient pressure adaptation in PUC					
C_PQ_AMP_AD	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Pressure quotient for adaptation ambient pressure (Initial value: 0.80)					
C_PQ_AMP_AD_PL	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Pressure quotient for adaptation ambient pressure during part load (Initial value: 0.35)					
C_SEG_CTR_AMP_AD	1	0...FFFFH	0...6.5535E+4	1	-
Segment counter of waiting time before allowing adaptation of ambient pressure (typical value (200)					
C_SEG_CTR_AMP_AD_PL	1	0...FFFFH	0...6.5535E+4	1	-
Segment counter for moving mean value calculation of AR_RED_BAS. MAF_KGH_MMV					
C_TCO_THD_AMP_AD	1	0...FEH	-48...142.5	0.75	°C
Threshold of TCO in case of no working load sensor					
C_TI_LAM_MAF_PULS_MAX	1	8000...7FFFH	-50...49.9984741	0.00152588	%
Positive correction of the ambient pressure adaptation during part load					
C_TI_LAM_MAF_PULS_MIN	1	8000...7FFFH	-50...49.9984741	0.00152588	%
Negative correction of the ambient pressure adaptation during part load					
C_TPS_MIN_AMP_AD_PL	1	0...3FFFH	0...119.5	0.00729415	°TPS
Minimum value of the throttle position for ambient pressure adaptation part load					
C_T_DLY_AMP_AD_MAP_MES_MIN	1	0...FFH	0...25.5	0.1	s
Delay time to allow the ambient pressure adaptation via MAP sensor					
C_T_DLY_AMP_AD_PUT_MES	1	0...FFH	0...25.5	0.1	s
Delay time to allow ambient pressure adaptation via PUT_MES					
C_VS_DIF_AMP_AD_PL	1	0...FFH	0...255	1	km/h
Vehicle speed decrease to detect a deceleration					
C_VS_MIN_AMP_AD_PL	1	0...FFH	0...255	1	km/h
Minimum vehicle speed to learn ambient pressure in IS					
LC_AR_RED_AD_ENA	1	0...1H	0...1	1	-
Logical variable enabled for permission of adaptive values for PUC					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PRS_INC_PUT_MES	6	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDP_FLOW_ENG_PRS_INC_PUT_MES	6	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Pressure correction factor for learning ambient pressure via PUT sensor					
IP_TPS_AMP_AD_MAP_MES_MIN	8	0...3FFFH	0...119.5	0.0072941 5	°TPS
LDP_N_32_IP_TPS_AMP_AD_MAP_MES	8	0...FFH	0...8.16E+3	32	rpm
Minimum open throttle to allow ambient pressure adaptation via MAP sensor					

4.16.1 INSY_ADCPRAMP0

General information:

Usually the ambient pressure is not measured with a dedicated sensor. When the engine is not running, then typically after 1-3 seconds the pressure in the manifold becomes equal to the ambient pressure of the environment. If a manifold pressure sensor is available (LV_ENA_MAP = 1) AMP_AD can be initialized with the measured MAP then.

Additionally the adaptation of the ambient pressure can be stopped by logical variables.

If the pressure ratio at the throttle exceeds the calibratable threshold C_PQ_AMP_AD and the engine speed is inside a calibratable range then the ambient pressure AMP_AD is learned (**Figure Learning of ambient pressure**).

Under the engine-state PUC a correction of the ambient pressure is made by a comparison between the measured air-mass flow and the calculated air-mass flow at the throttle.

For a configuration with charger an ambient pressure sensor is needed.

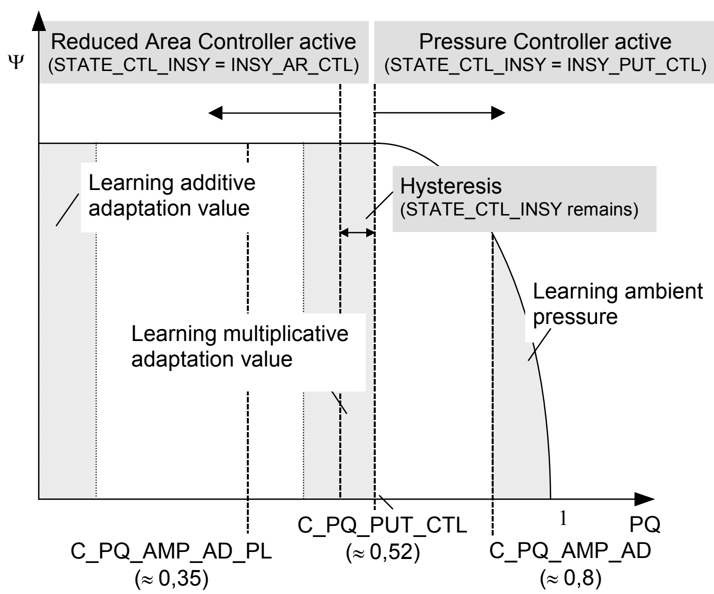


Figure: **Learning of ambient pressure**

Application Condition

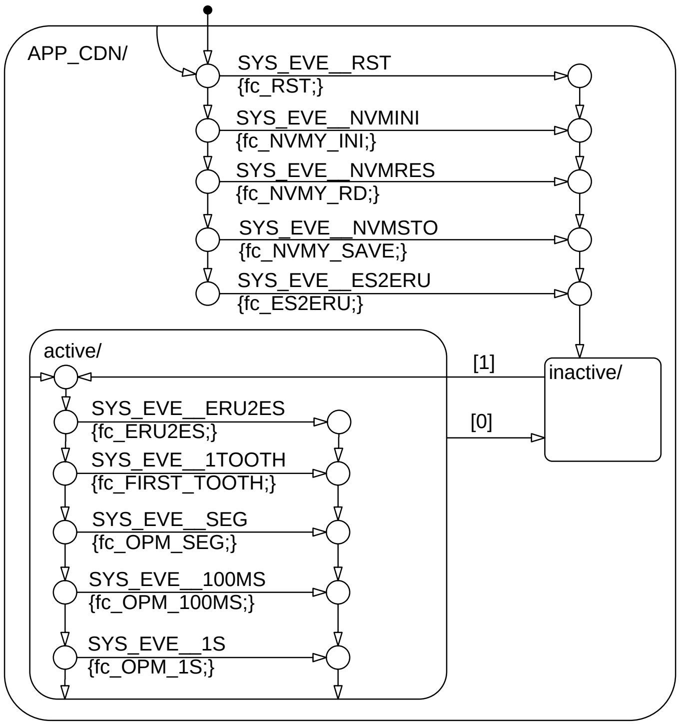


Figure: INSY_ADCPRAMP0/ APP_CDN/ Chart

Function Description

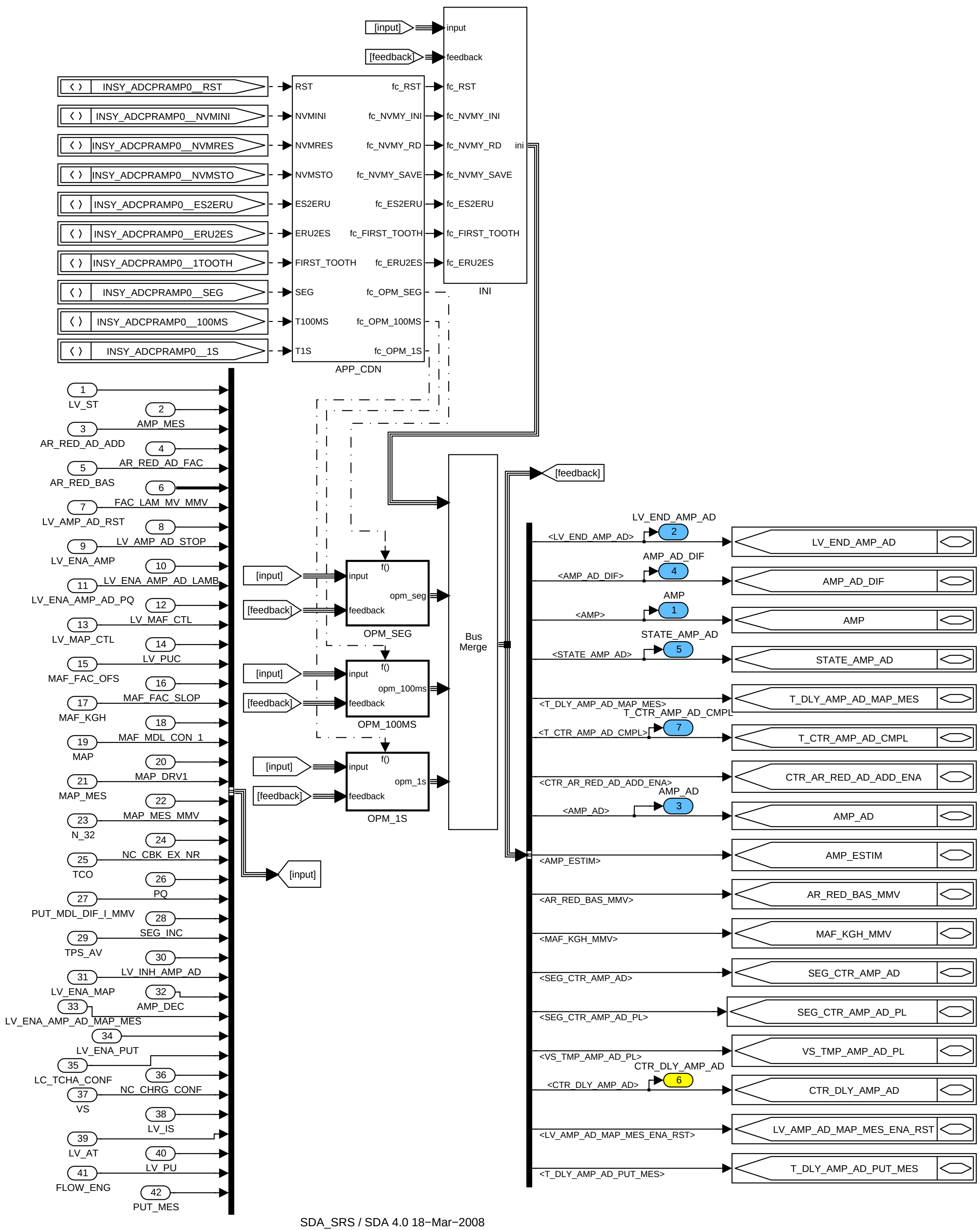


Figure 212 INSY_ADCPRAMP0

4.16.1.1 Bus merge of function:

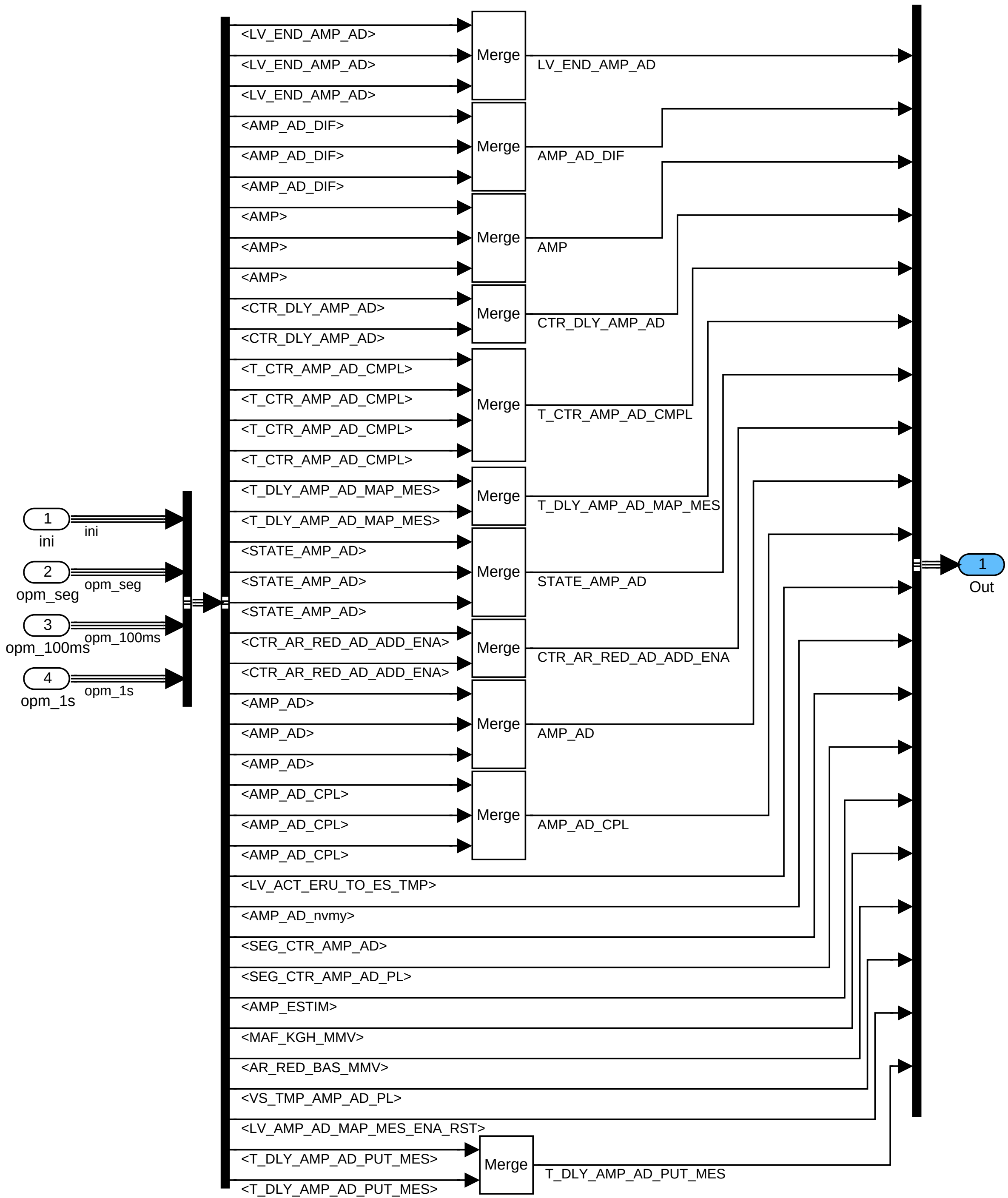


Figure 213 INSY_ADCPRAMP0/ BusMerge

4.16.1.2 Initialization of ambient pressure AMP_AD if no ambient pressure sensor is available

Initialization of ambient pressure AMP_AD if no ambient pressure sensor is available

Usually the ambient pressure is not measured with a dedicated sensor (LV_ENA_AMP = 0).

The adapted ambient pressure AMP_AD is initialized with its value stored in the none volatile memory. In case of a checksum error, AMP_AD is initialized with C_AMP_INI. During the initialization of the ECU, MAP and AMP are initialized with this AMP_AD. If the system is equipped with a properly working manifold pressure sensor, AMP_AD is overwritten during the detection of the first valid tooth (see Figure Initialization path for ambient pressure variable AMP).

The measurement of the ambient pressure AMP is done with the manifold pressure sensor. It must therefore be ensured that during the ambient pressure data acquisition the engine is not running. For this purpose, the first valid tooth will be the trigger i.e. the motor is nearly standing. If this tooth is detected, the current buffer content is used for sensor diagnosis and for calculating MAP_MES. If MAP_MES is validated by the sensor diagnosis, AMP and AMP_AD are set to MAP_MES.

Information to detect the first ambient pressure adaptation within the engine run must be stored in the memory area that is not cleared by the RAM test. This ensures that measurement of ambient pressure with the MAP-sensor will not be allowed in case of ECU-resets, resynchronization during running engine or during a restart with not finished pressure equalization.

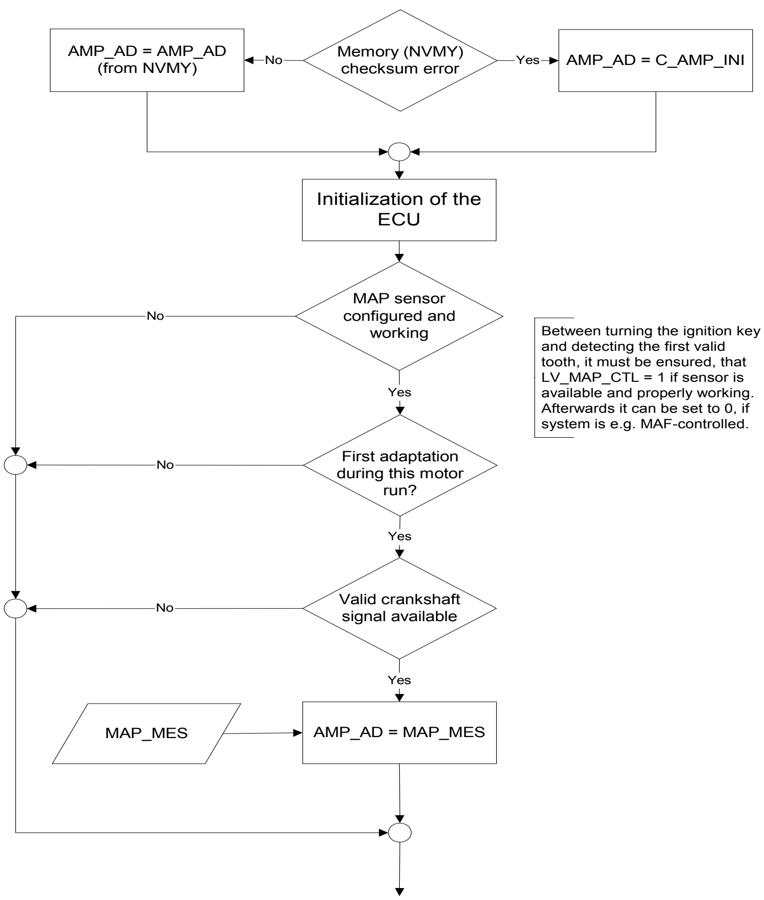


Figure: Initialization path for ambient pressure variable AMP

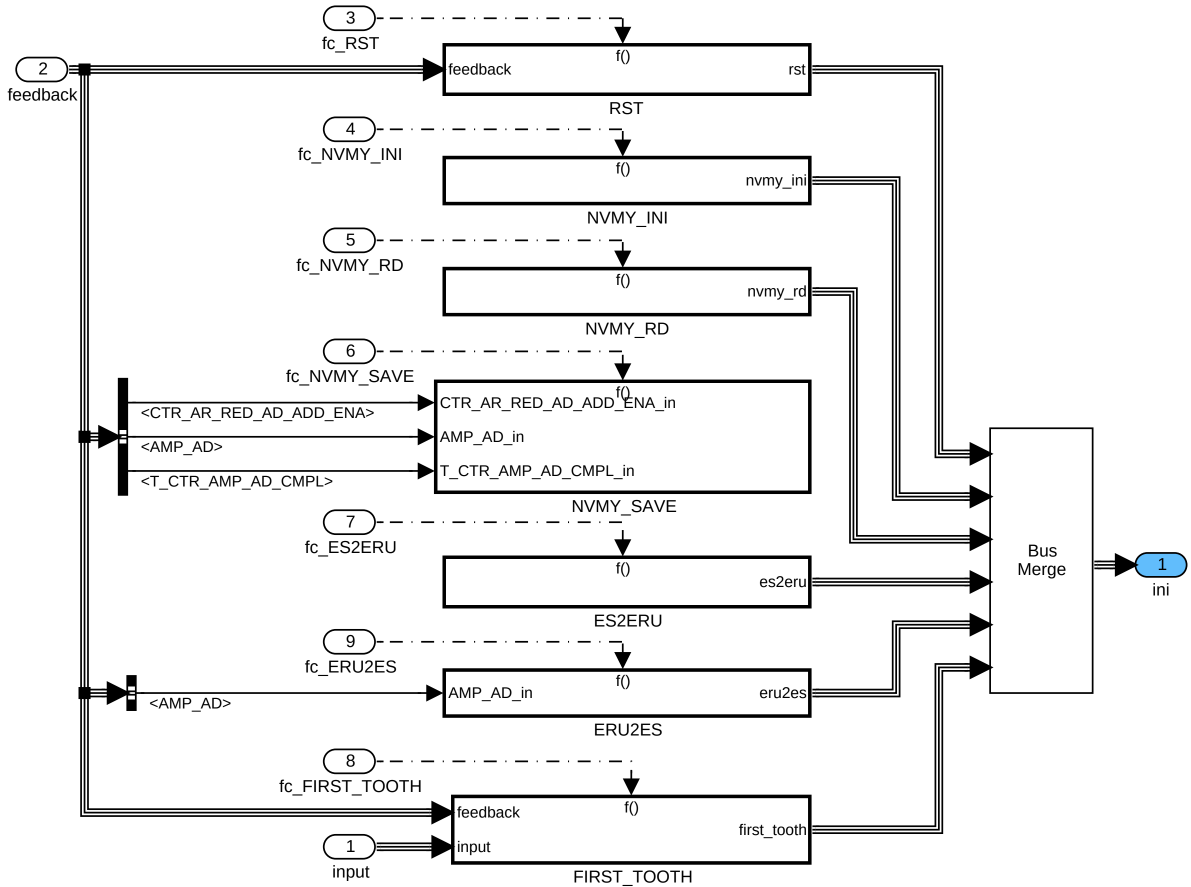


Figure 214 INSY_ADCPRAMP0/ INI

Bus merge for initialization task:

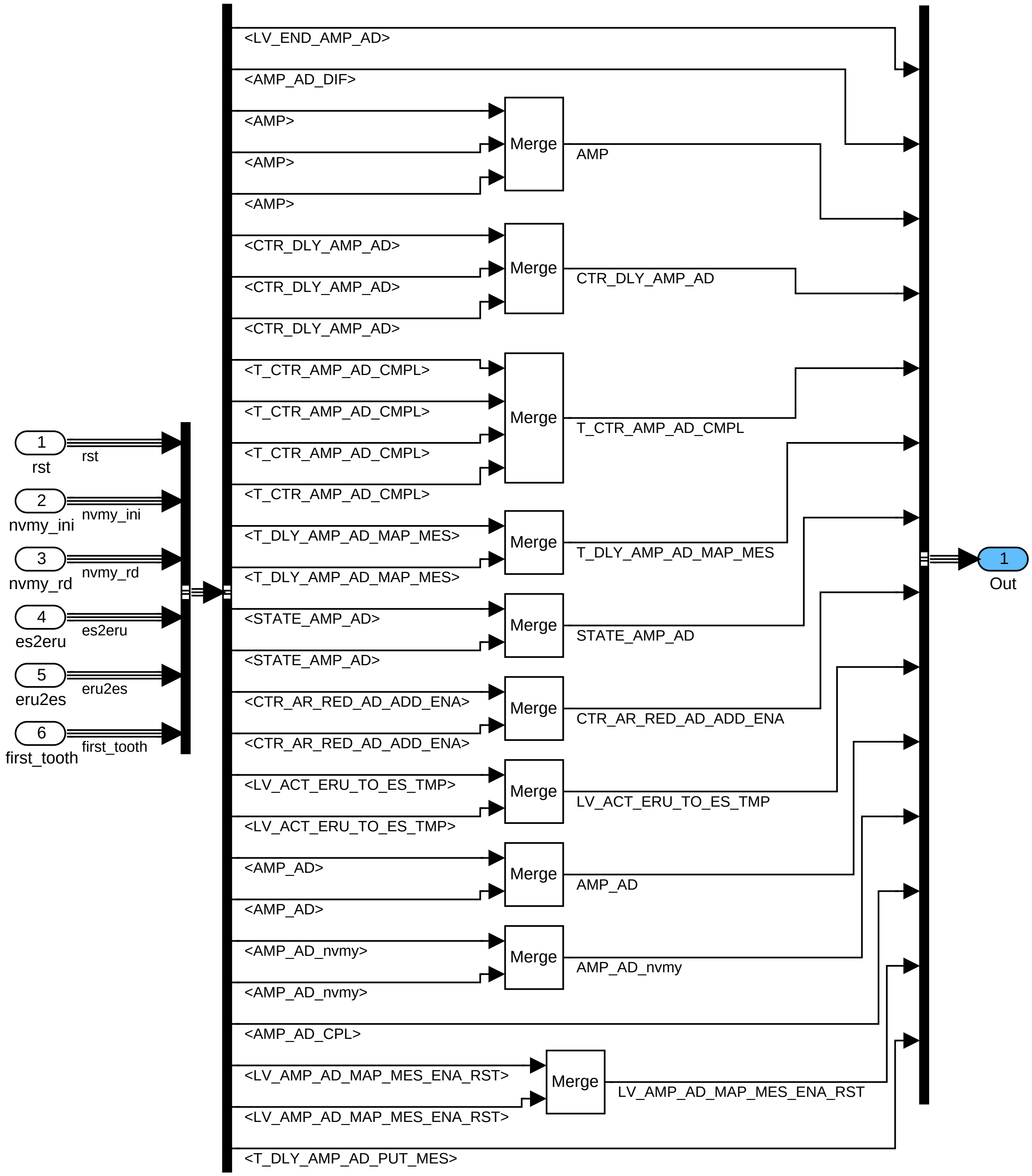


Figure 215 INSY_ADCPRAMP0/ INI/ BusMerge

general specification

Initialization at engine run to engine stop:

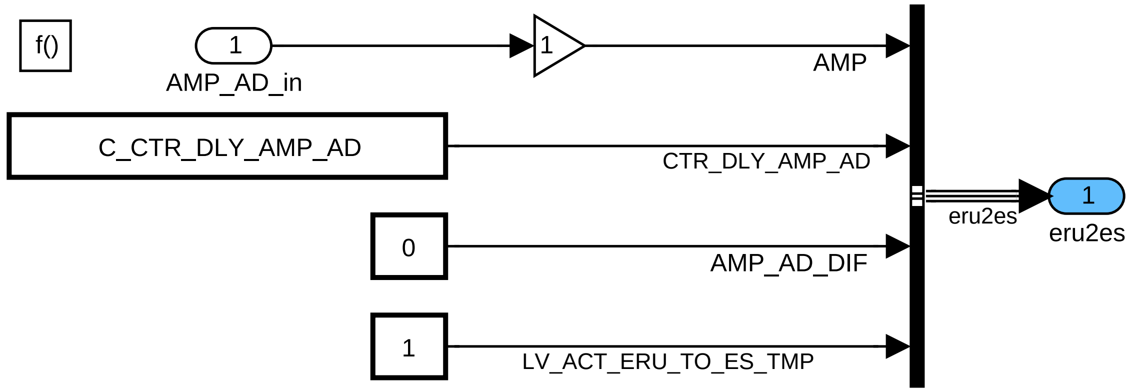


Figure 216 INSY_ADCPRAMP0/ INI/ ERU2ES

Initialization at engine stop to engine run:

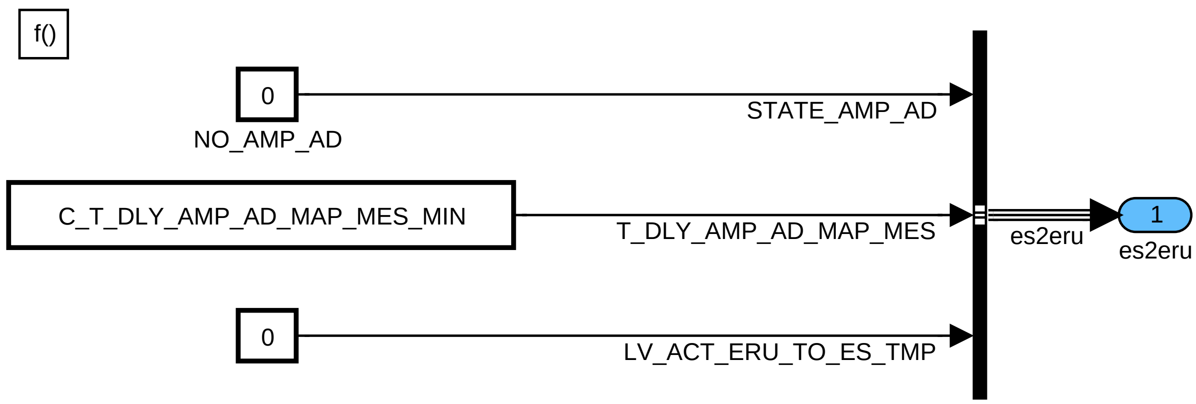


Figure 217 INSY_ADCPRAMP0/ INI/ ES2ERU

Initialization at detection of the first valid tooth:

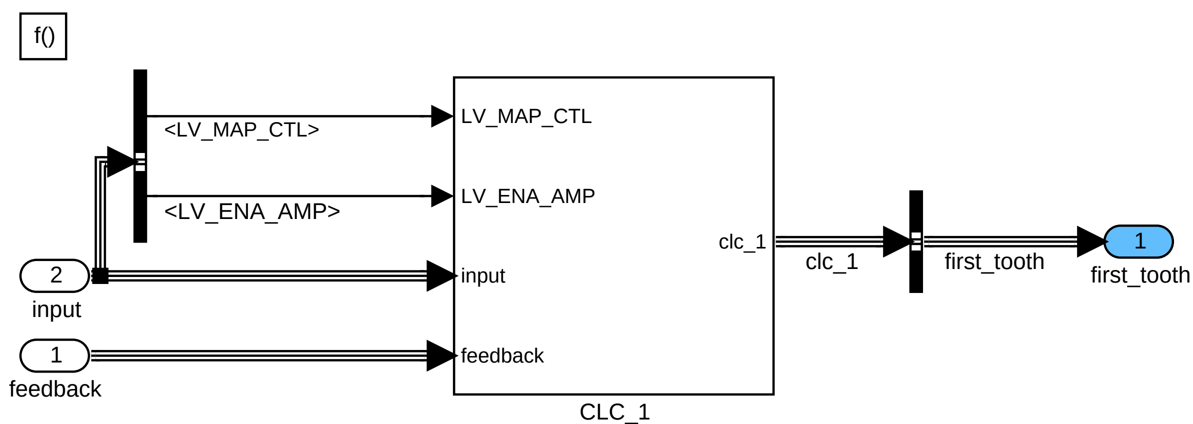


Figure 218 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH

Initialization at detection of the first valid tooth:

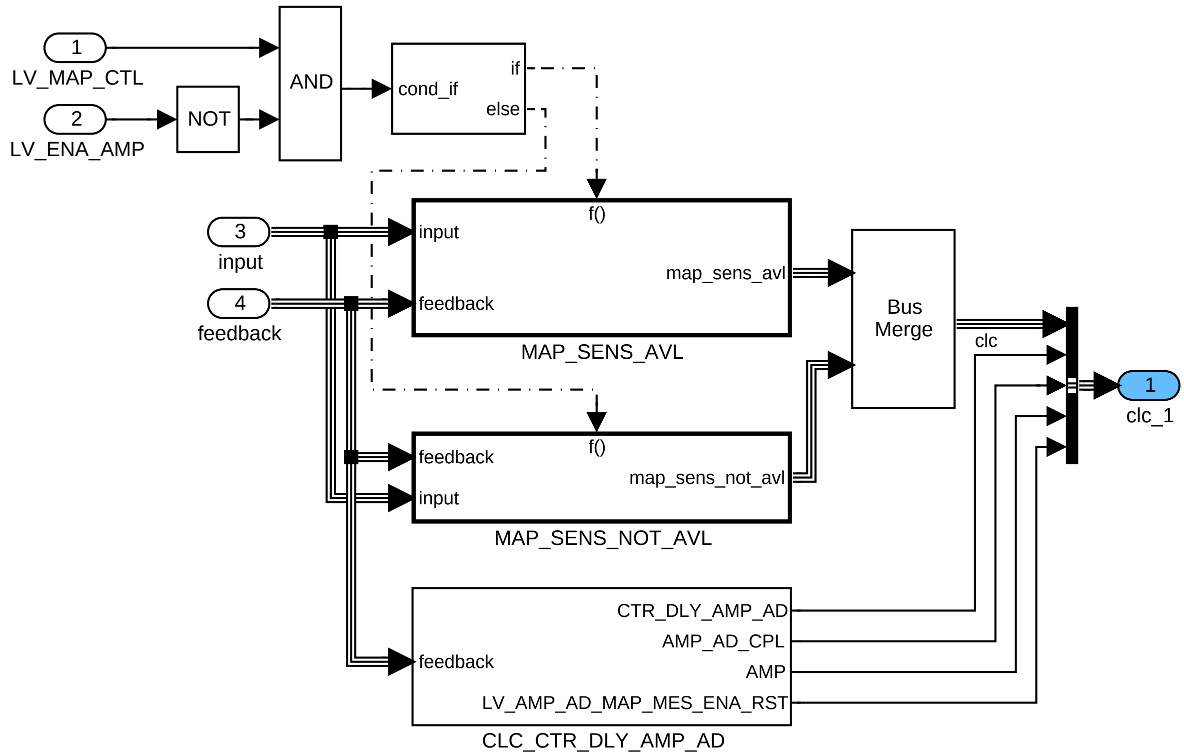


Figure 219 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1

Bus merge for initialization at first valid tooth:

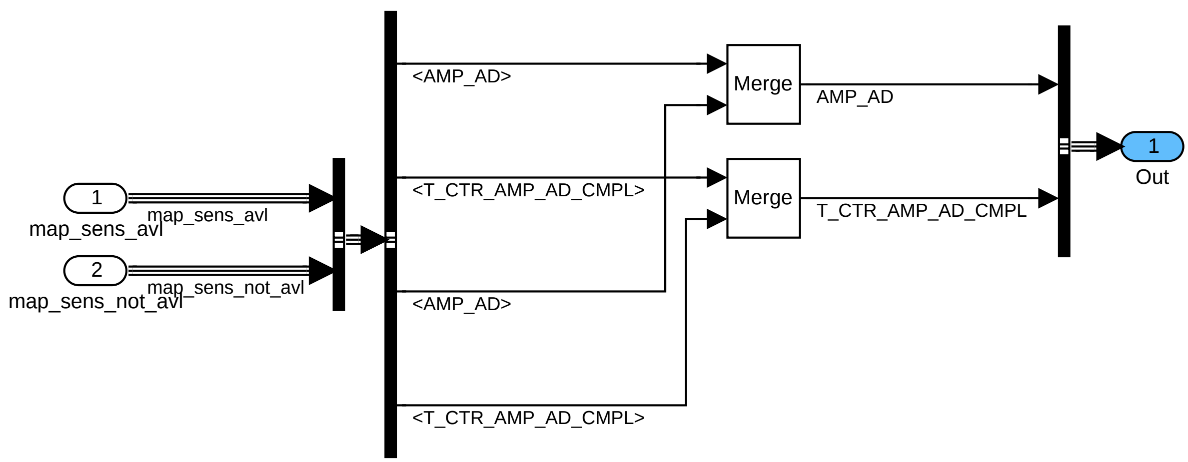


Figure 220 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ BusMerge

Calculation of CTR_DLY_AMP_AD and AMP_AD_CPL at first valid tooth:

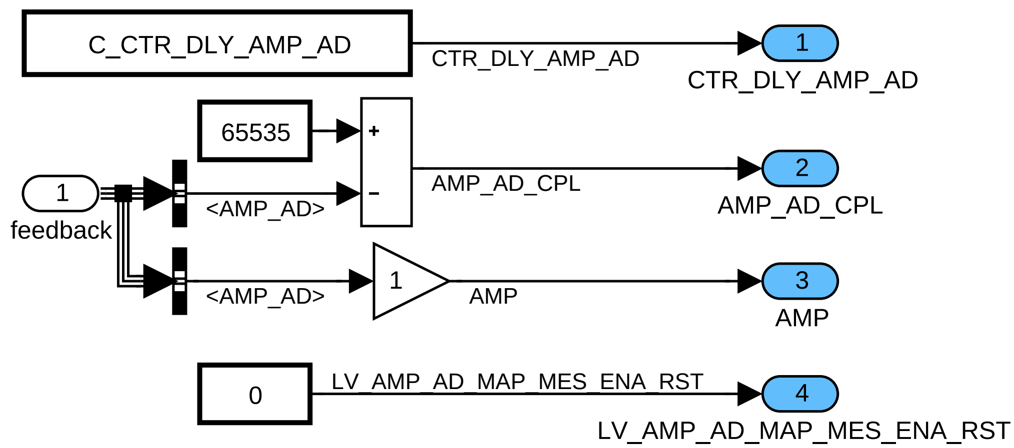


Figure 221 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ CLC_CTR_DLY_AMP_AD

Initialization at first valid tooth if MAP sensor available and no AMP sensor available:

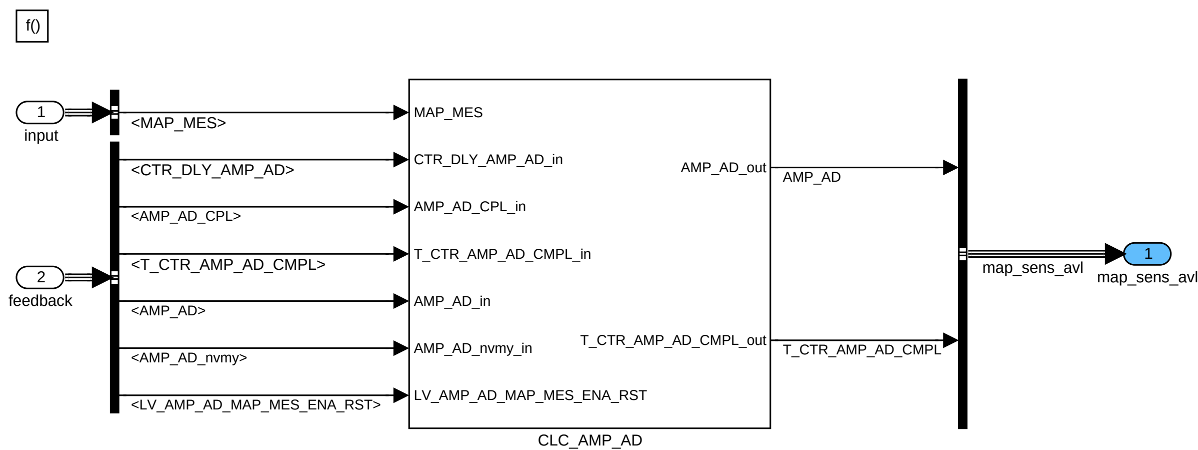


Figure 222 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ MAP_SENS_AVL

Initialization at first valid tooth if MAP sensor available and no AMP sensor available:

When the engine is not running, after a certain time (typically after 1 3 seconds) when the pressure in the intake manifold is equal to the ambient pressure or no warm reset was detected then AMP and AMP_AD are initialized with the measured MAP. If a warm reset was detected then AMP_AD is initialized with the so far adapted AMP value. At a cold reset with MAP_MES out of his plausible range, AMP_AD is initialized with the AMP_AD value read out from non volatile memory.

A warm reset can be detected by comparing the AMP_AD value with a calculated complementary value of AMP_AD.

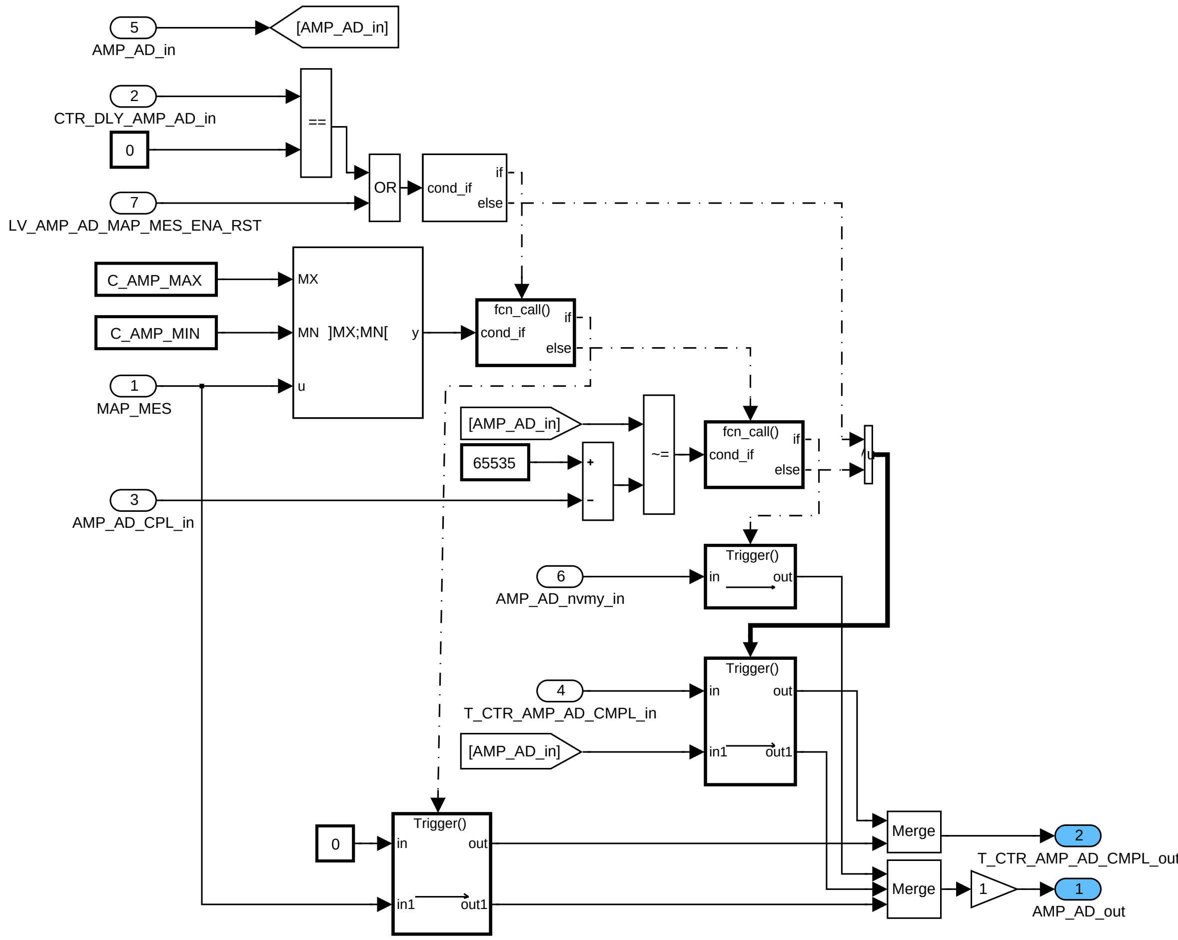


Figure 223 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ MAP_SENS_AVL/ CLC_AMP_AD

<u>Initialization at first valid tooth if MAP sensor not available or AMP sensor available:</u>

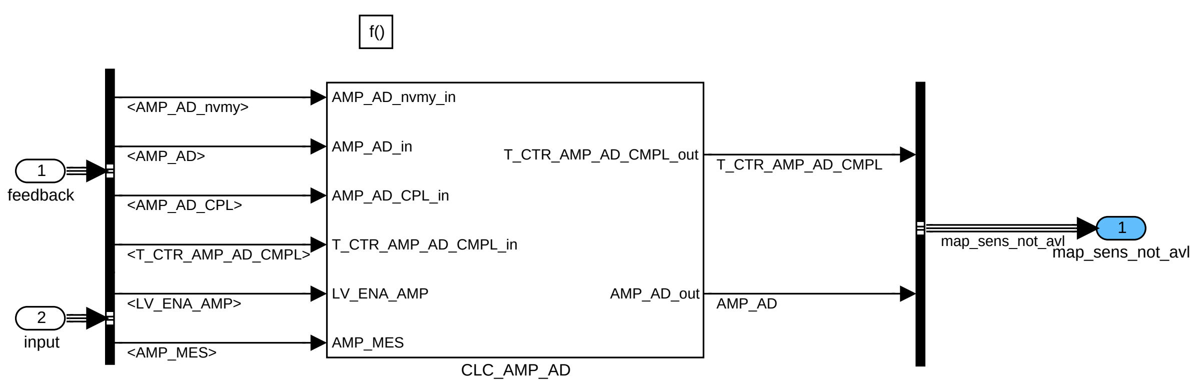


Figure 224 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ MAP_SENS_NOT_AVL

<u>Initialization at first valid tooth if MAP sensor not available or AMP sensor available:</u>

If a warm reset was detected then AMP_AD is initialized with the so far adapted AMP value. At a cold reset with an AMP-sensor available, AMP_AD is initialized with the measured AMP value. If no AMP-sensor is available then AMP_AD is initialized with the AMP_AD value read out from non volatile memory.

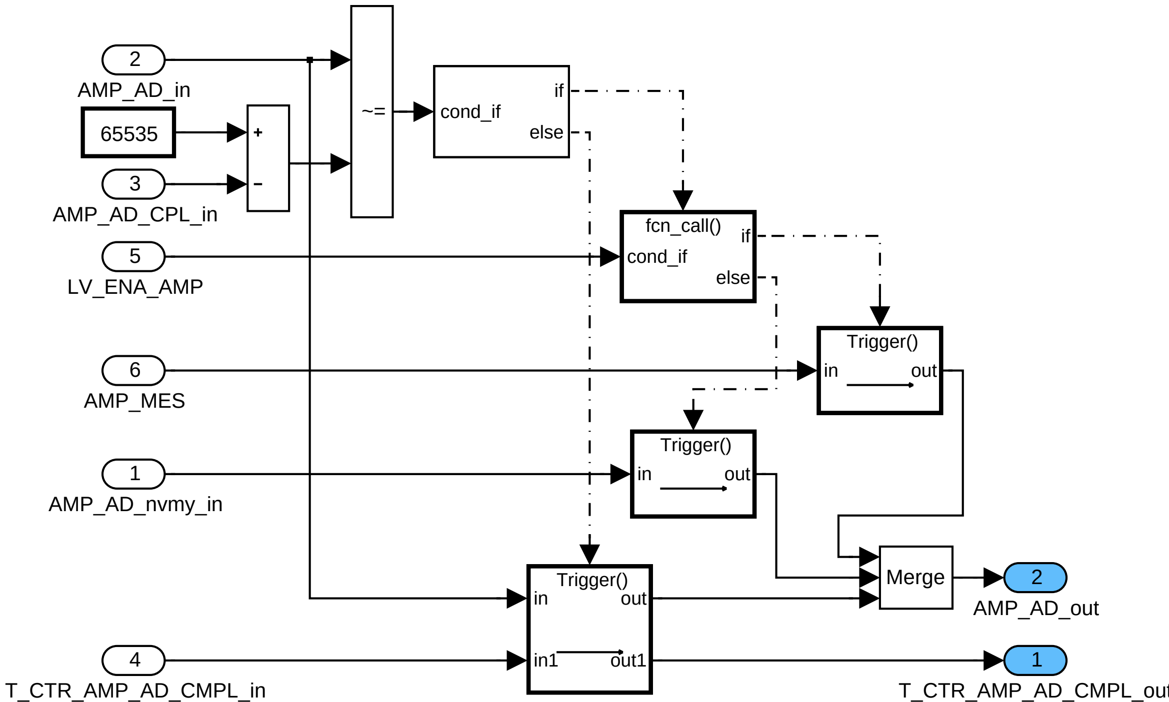


Figure 225 INSY_ADCPRAMP0/ INI/ FIRST_TOOTH/ CLC_1/ MAP_SENS_NOT_AVL/ CLC_AMP_AD

Initialization of variables which are stored in non volatile memory (only for brand new ECU):

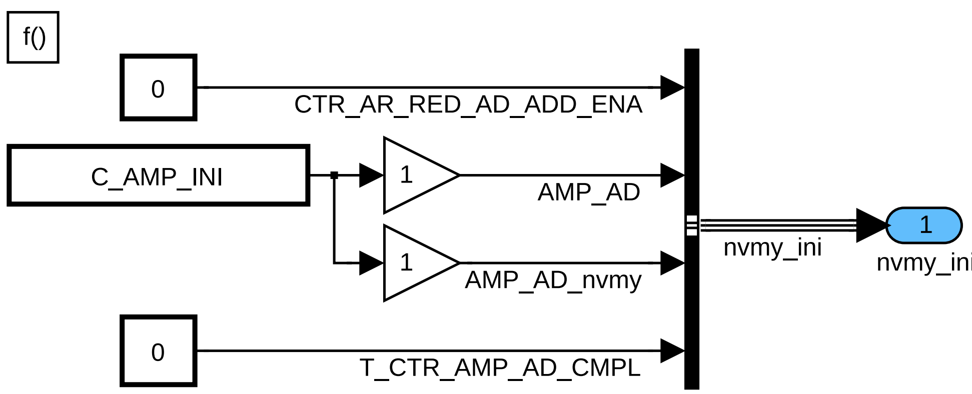


Figure 226 INSY_ADCPRAMP0/ INI/ NVMY_INI

Read out variables from non volatile memory at reset:

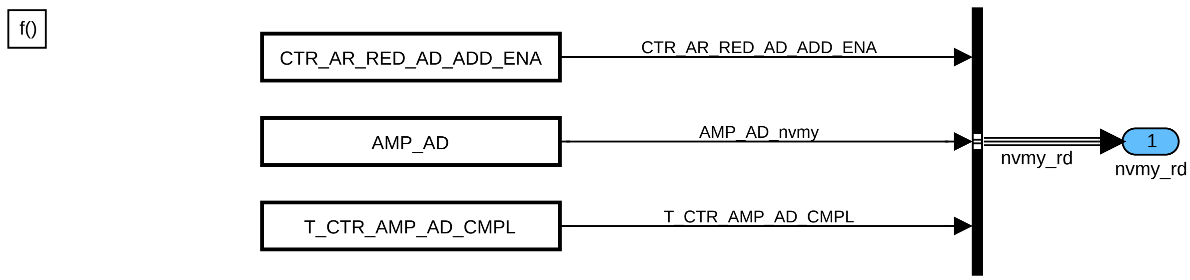


Figure 227 INSY_ADCPRAMP0/ INI/ NVMY_RD

Write variables to non volatile memory at powerlatch:

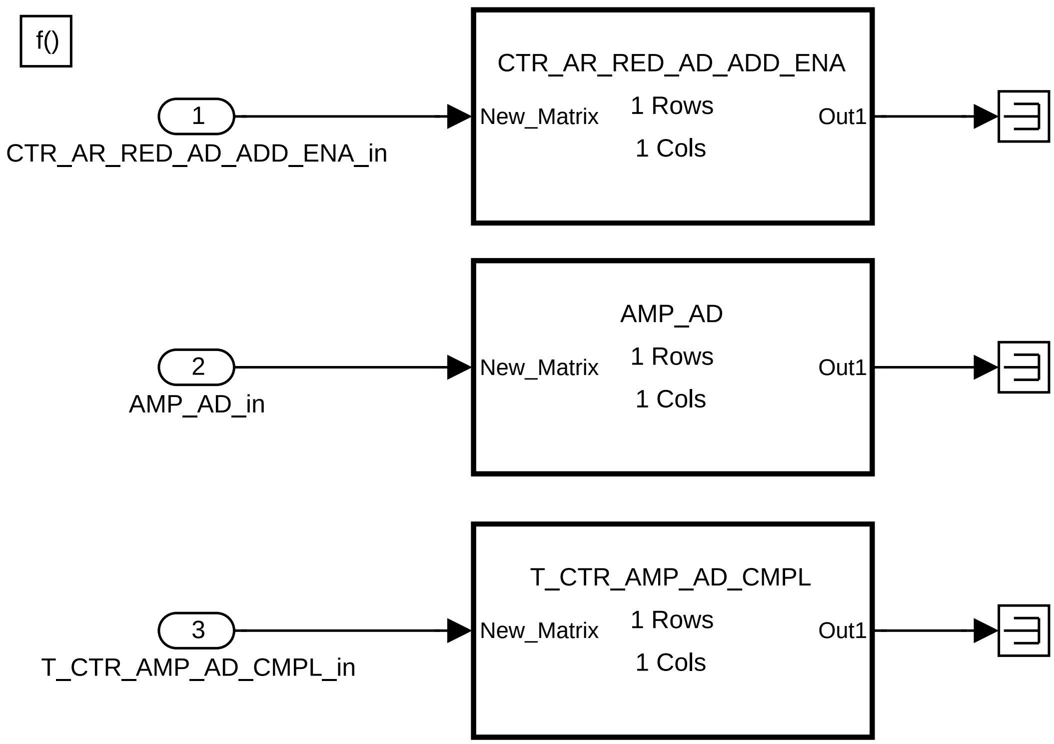


Figure 228 INSY_ADCPRAMP0/ INI/ NVMY_SAVE

Initialisation at reset

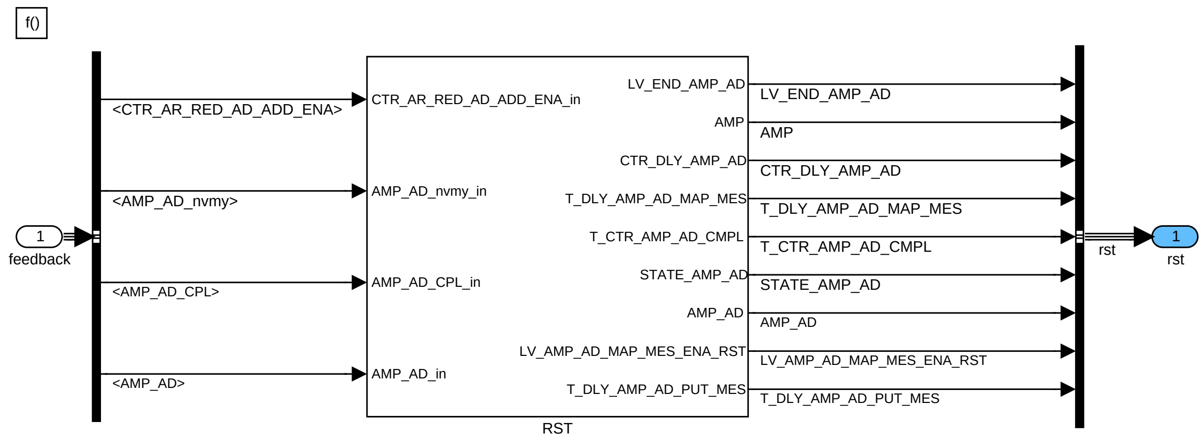


Figure 229 INSY_ADCPRAMP0/ INI/ RST

Initialisation at reset

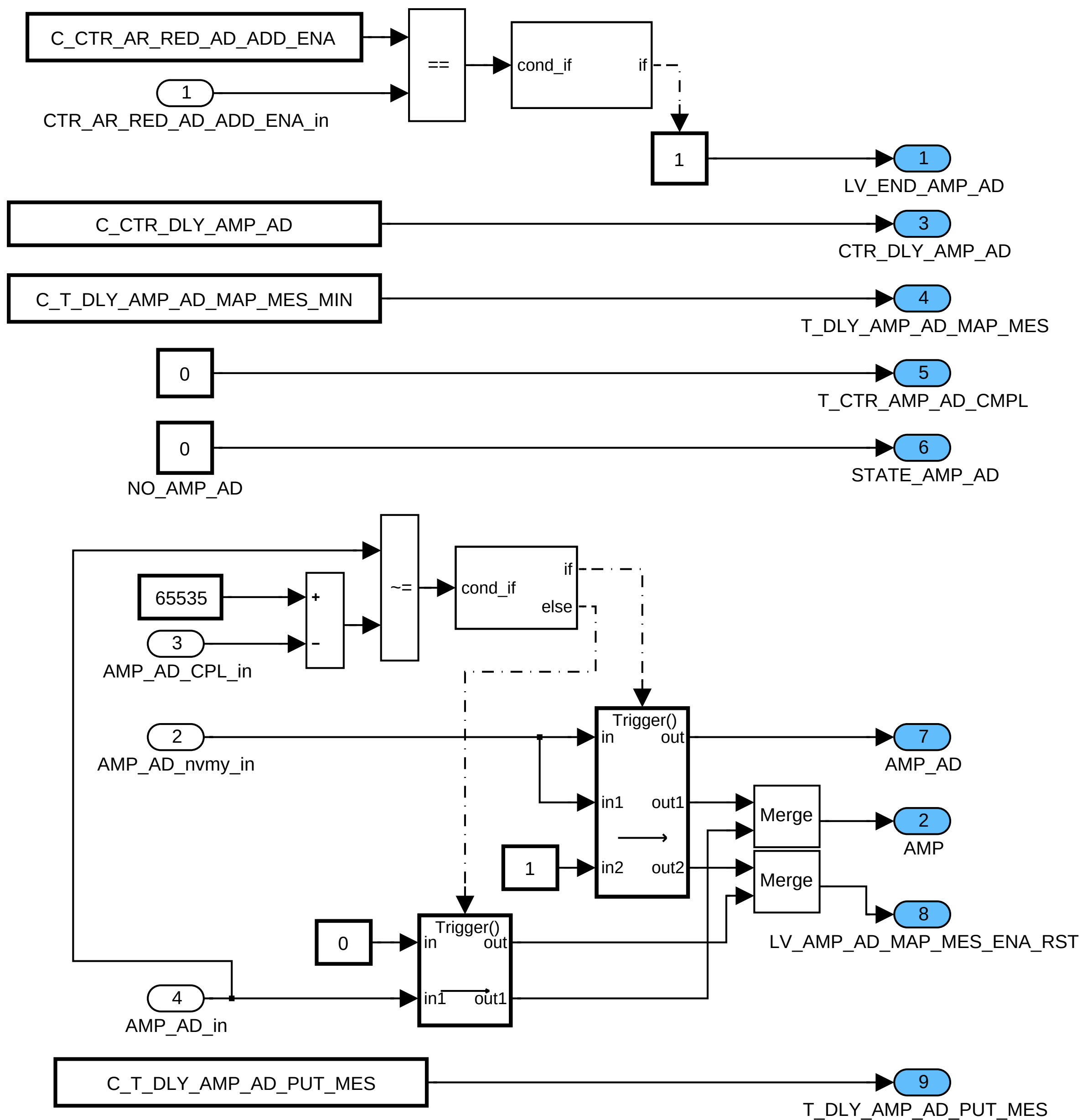


Figure 230 INSY_ADCPRAMP0/ INI/ RST/ RST

4.16.1.3 Learning of ambient pressure at full load (takeover of MAP_MES) – recurrence 100ms:

If the System is equipped with a MAP sensor, the adaptation of the Ambient pressure can be done at full load using the load sensor signal. Adaptation is only allowed, when the enable bit LV_ENA_AMP_MAP_MES has been set by the Component and Adaptation Manager module. This enable flag takes into account the inhibition of Pressure learning if the load/throttle position rationality check detects implausibilities (LV_ERR_RATIO_CHK =1). The adaptation is only performed if the throttle opening is above a calibratable angle.

Ambient pressure should only be learned when a stable system is reached (e.g. no adaptation during short tip-in). Therefore a counter T_DLY_AMP_AD_MAP_MES is initialized with C_T_DLY_AMP_AD_MAP_MES at the transition from (TPS_AV > IP_TPS_AMP_AD_MAP_MES_MIN) to (TPS_AV <= IP_TPS_AMP_AD_MAP_MES_MIN) and then decreased by 100 ms every calculation call until it reaches zero. Adaptation is only allowed if this counter has reached zero. If one of the conditions are not fulfilled T_DLY_AMP_AD_MAP_MES is set back to C_T_DLY_AMP_AD_MAP_MES.

If T_DLY_AMP_AD_MAP_MES reaches zero, then the measured manifold pressure MAP_MES will be stored. With this stored value – regardless of whether the conditions are fulfilled or not – AMP_AD_DIF will be calculated. This is done till PUT is within a calibratable range around the stored manifold pressure.

Are the conditions of this algorithm fulfilled then the ambient pressure adaptation via model will be stopped and the ambient pressure will be adapted using the measured manifold pressure.

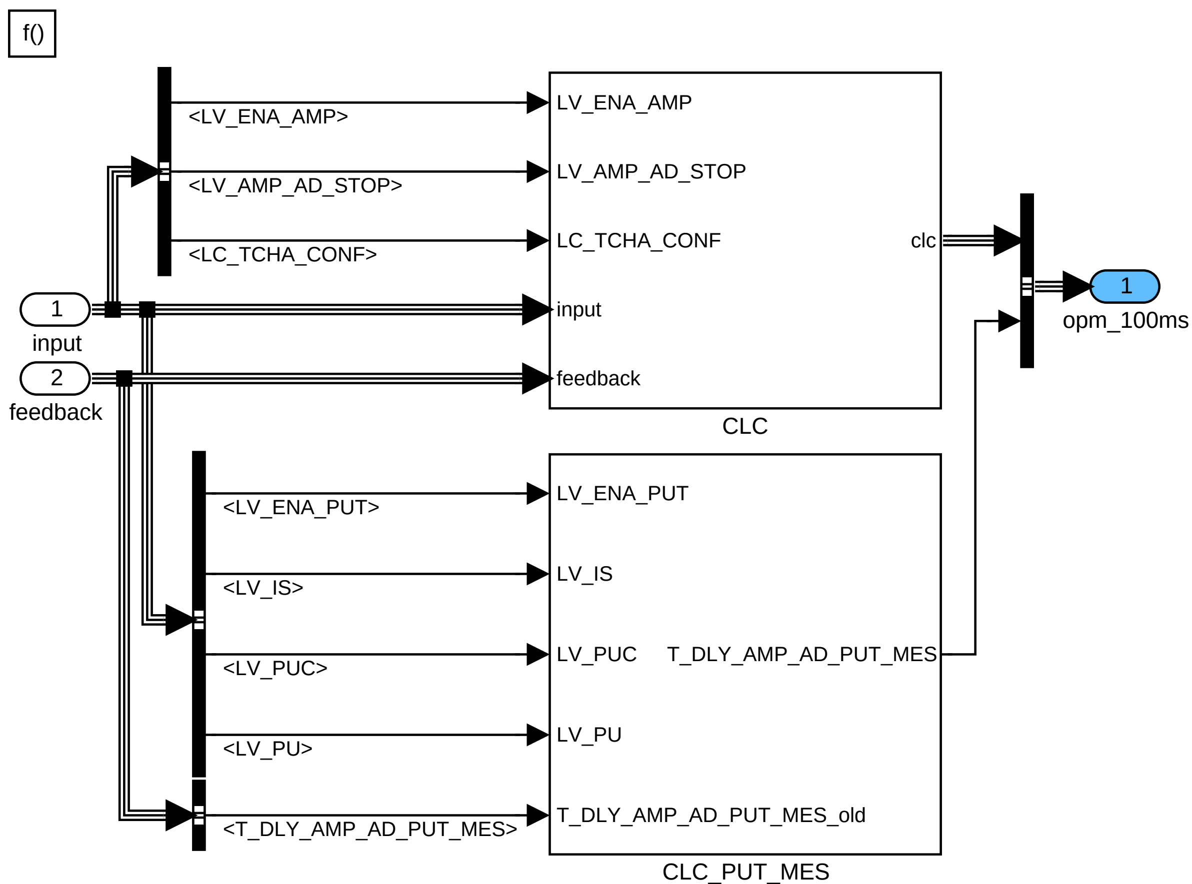


Figure 231 INSY_ADCPRAMP0/ OPM_100MS

Activation of learning of ambient pressure at full-load (takeover of MAP_MES):

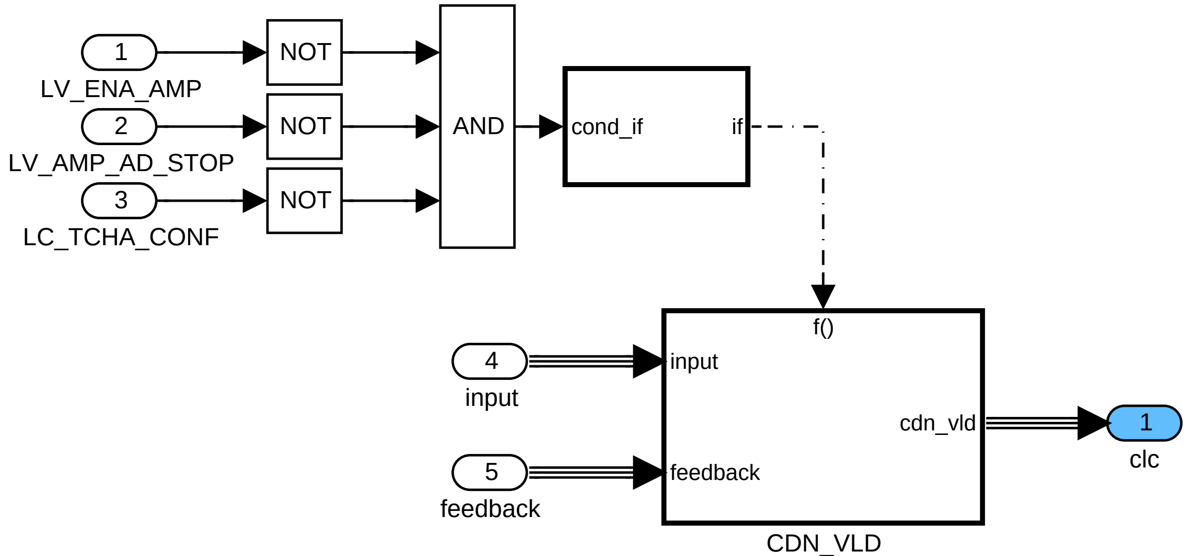


Figure 232 INSY_ADCPRAMP0/ OPM_100MS/ CLC

Overview over calculations for learning of ambient pressure at full-load (takeover of MAP_MES):

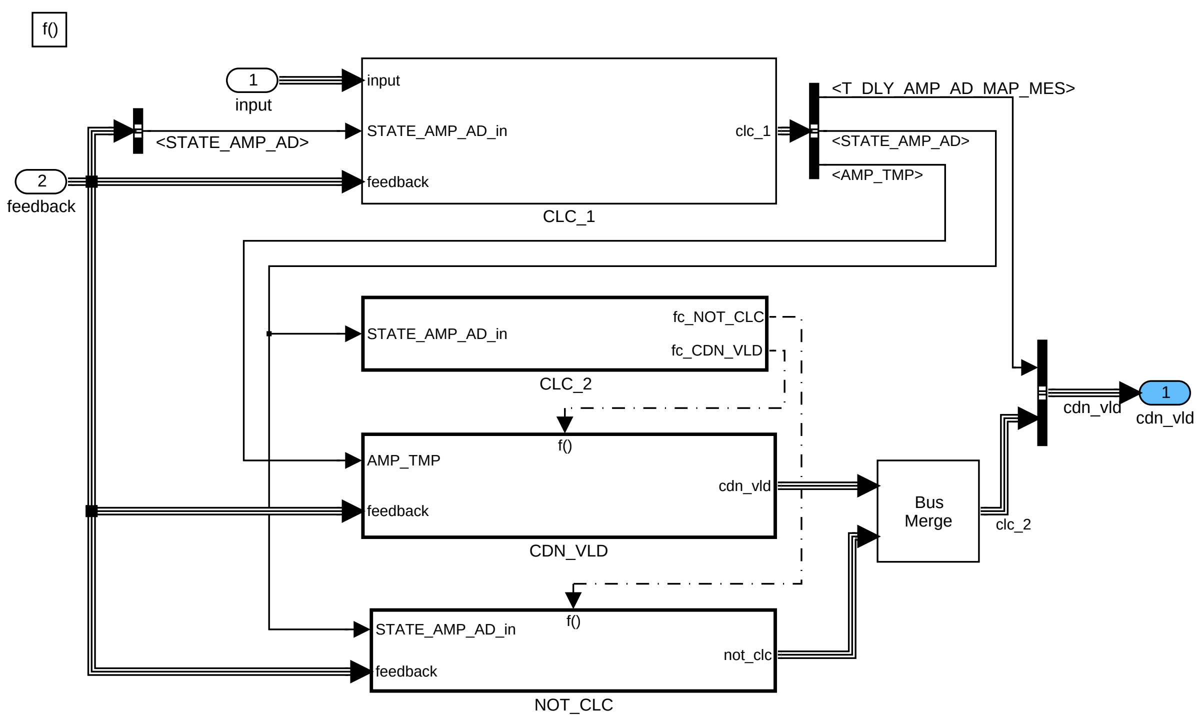


Figure 233 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD

Bus merge for chapter "Learning of ambient pressure at full-load (takeover of MAP_MES)":

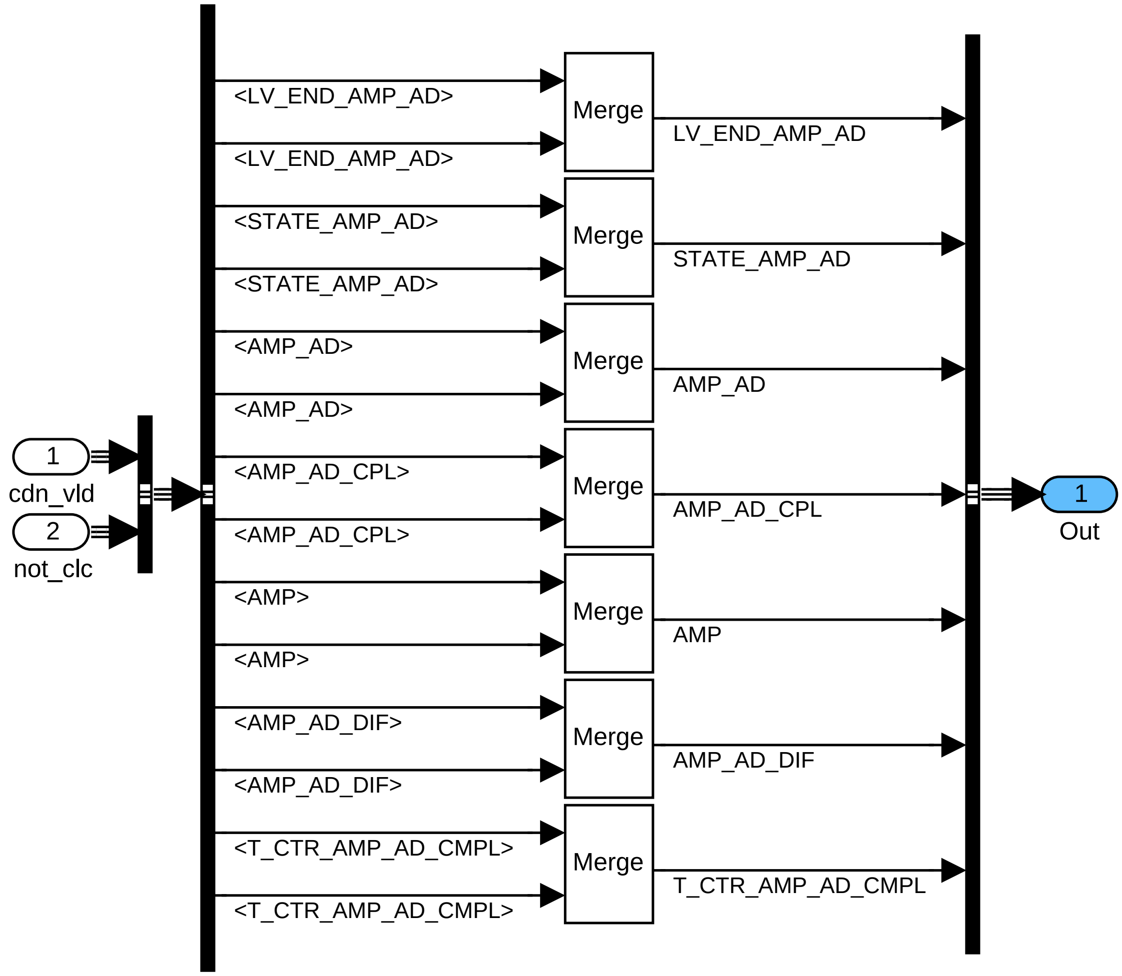


Figure 234 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ BusMerge

Ambient pressure was learned via MAP_MES: Calculation of AMP_AD_DIF and AMP_AD and limitation of AMP_AD

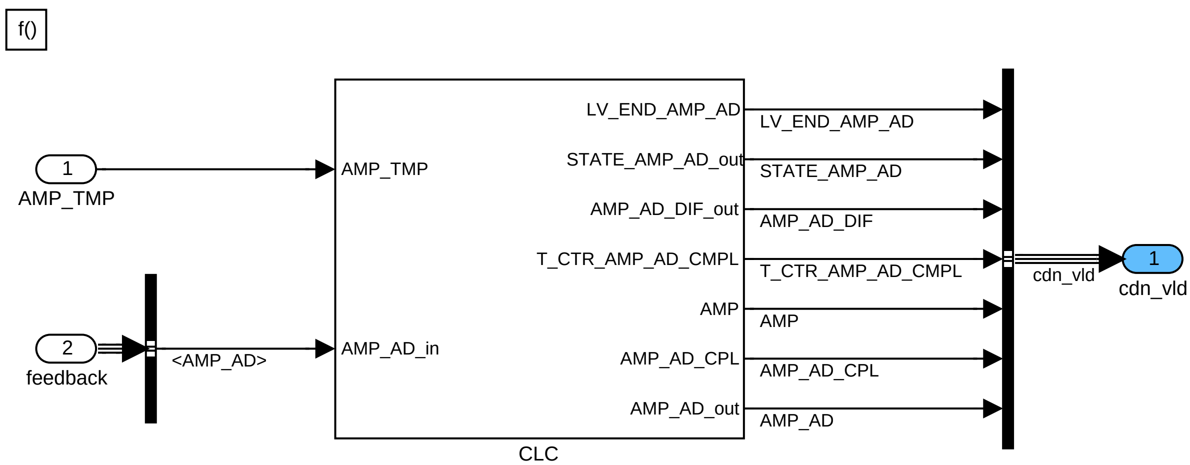


Figure 235 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CDN_VLD

Ambient pressure was learned via MAP_MES: Calculation of AMP_AD_DIF and AMP_AD and limitation of AMP_AD

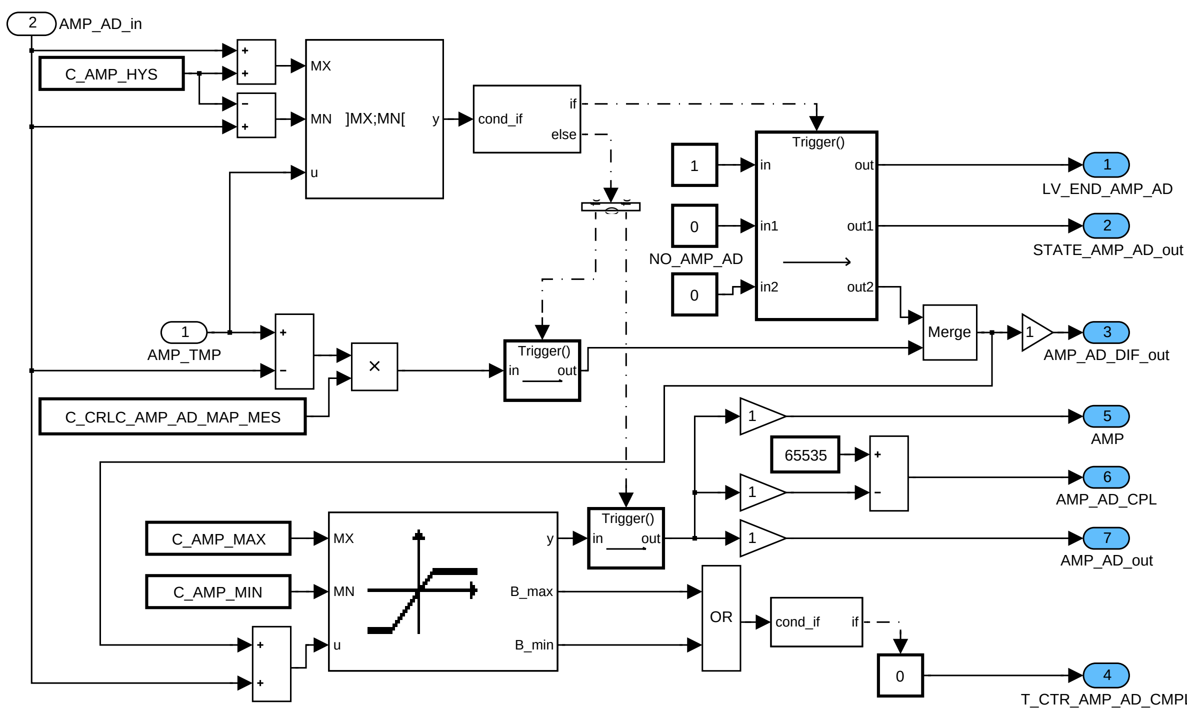


Figure 236 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CDN_VLD/ CLC

Checking whether a stable system at full load is reached for ambient pressure learning via MAP_MES:

Learning of ambient pressure via the measured MAP value should only be done after a stable system at full load is reached. To ensure stable conditions TPS_AV must be higher than a threshold (IP_TPS_AMP_AD_MAP_MES_MIN) for a certain time

(C_T_DLY_AMP_AD_MAP_MES_MIN). Only after reaching this stable system at full load, STATE_AMP_AD is set to 3 (AMP_AD_VIA_MAP_MES).

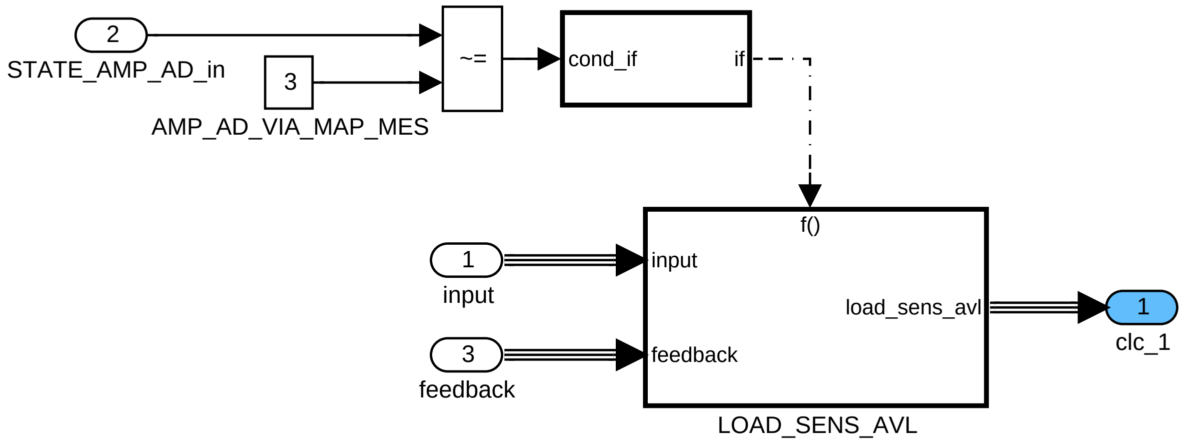


Figure 237 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_1

<u>Checking whether a stable system at full load is reached for ambient pressure learning via MAP_MES:</u>

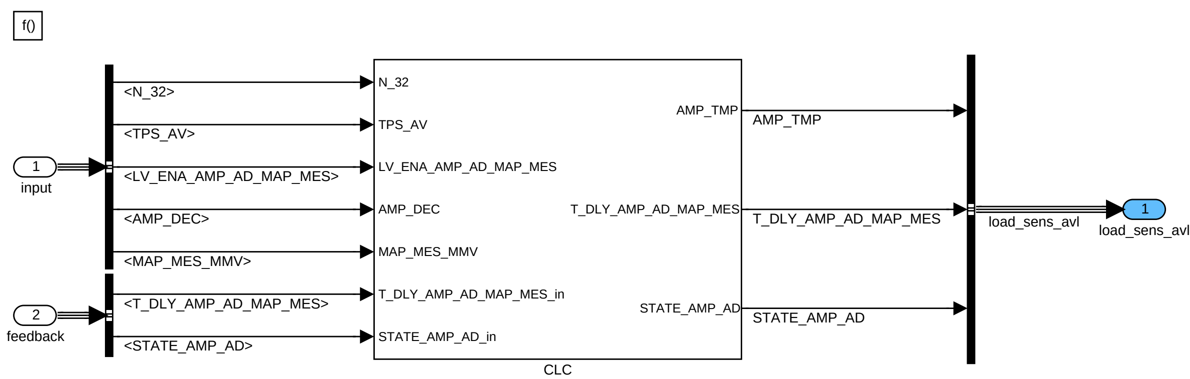


Figure 238 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_1/ LOAD_SENS_AVL

Checking whether full load is reached for ambient pressure learning via MAP_MES:

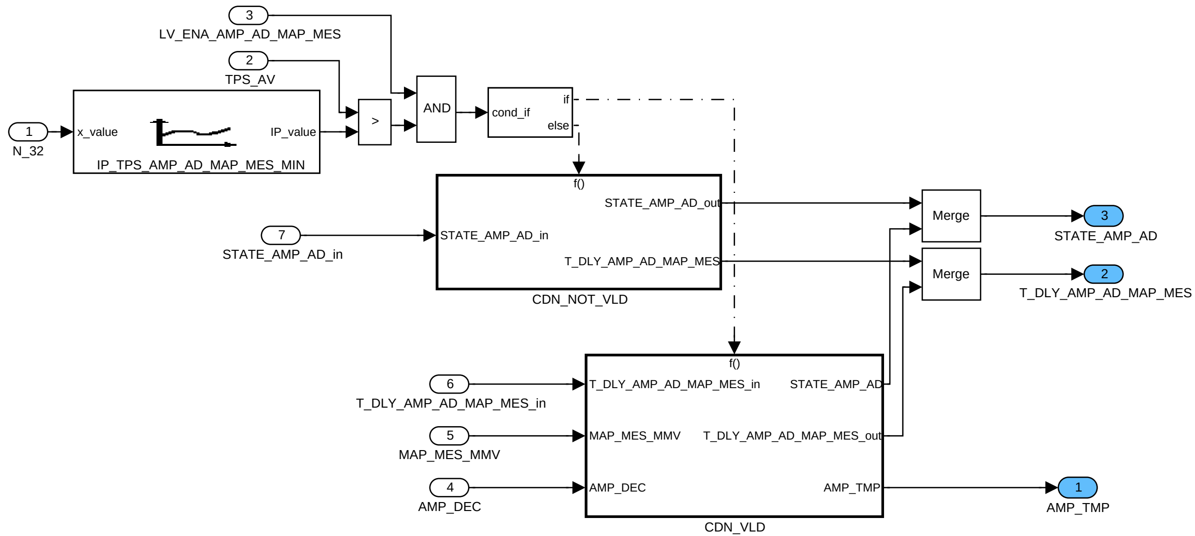


Figure 239 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_1/ LOAD_SENS_AVL/ CLC

Full load condition not fulfilled:

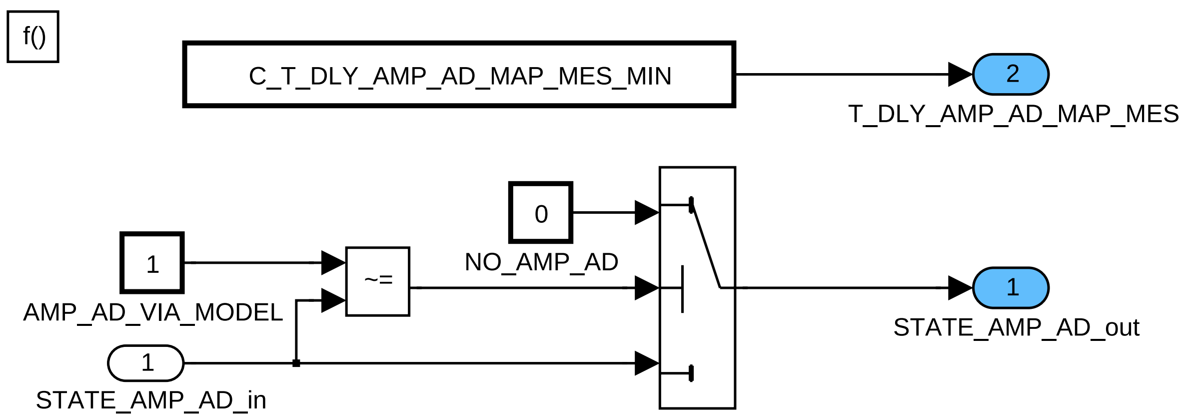


Figure 240 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_1/ LOAD_SENS_AVL/ CLC/ CDN_NOT_VLD

Full load condition fulfilled: Checking whether a stable system is reached and, if condition fulfilled, ambient pressure is learned via MAP_MES:

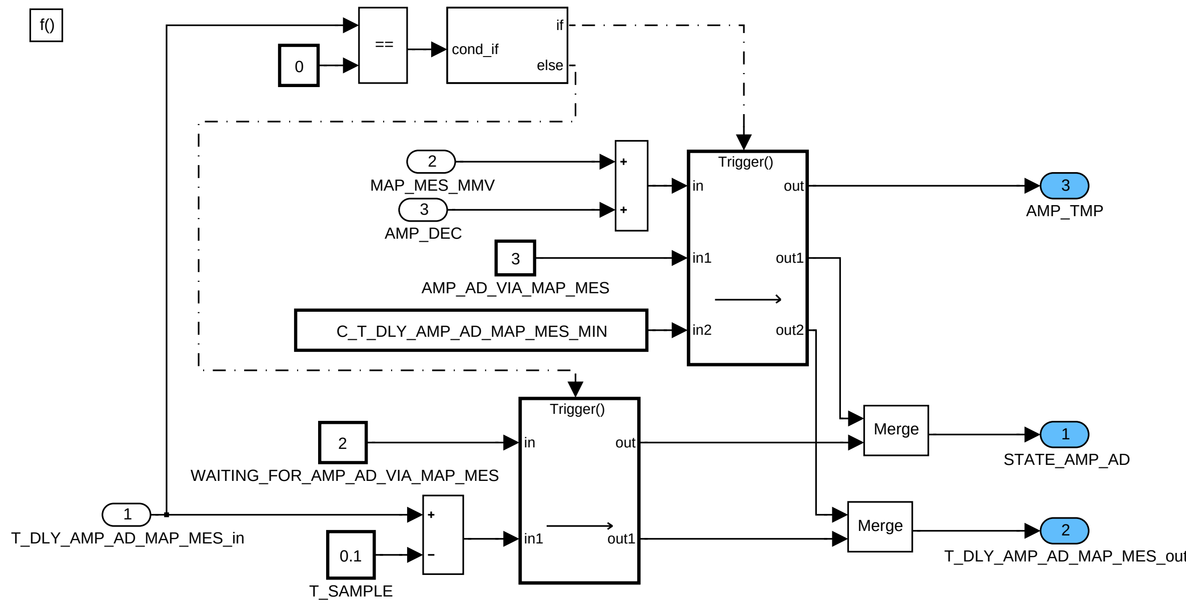


Figure 241 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_1/ LOAD_SENS_AVL/ CLC/ CDN_VLD

Checking whether a stable system at full load was reached and ambient pressure was learned via MAP_MES:

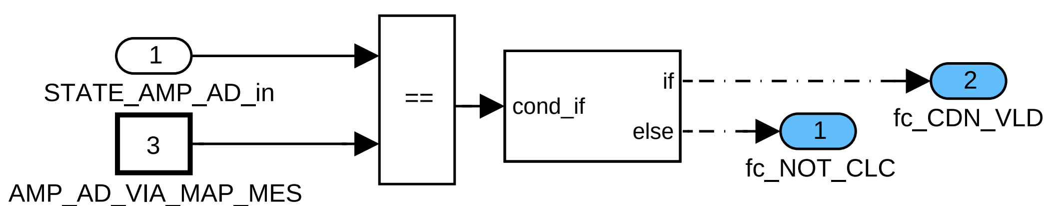


Figure 242 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ CLC_2

A stable system at full load was not reached and ambient pressure was not learned via MAP_MES:

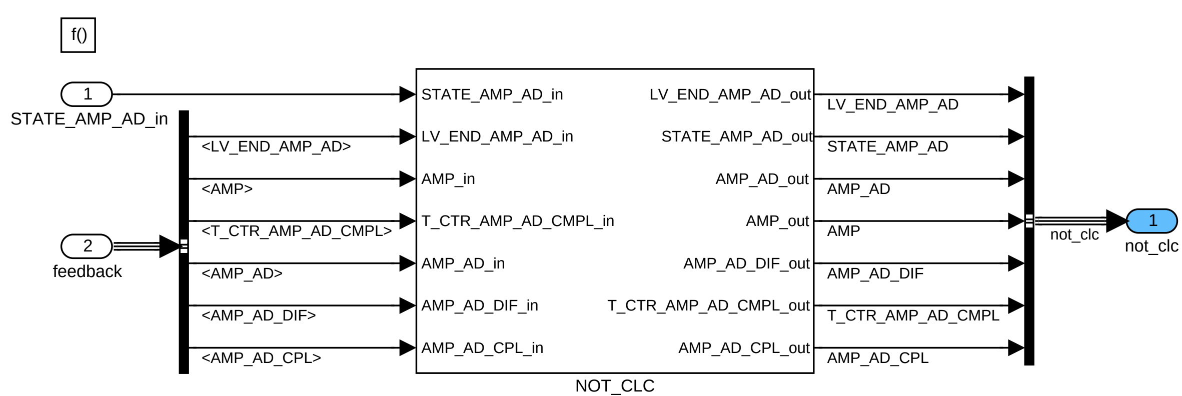


Figure 243 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ NOT_CLC

A stable system at full load was not reached and ambient pressure was not learned via MAP_MES:

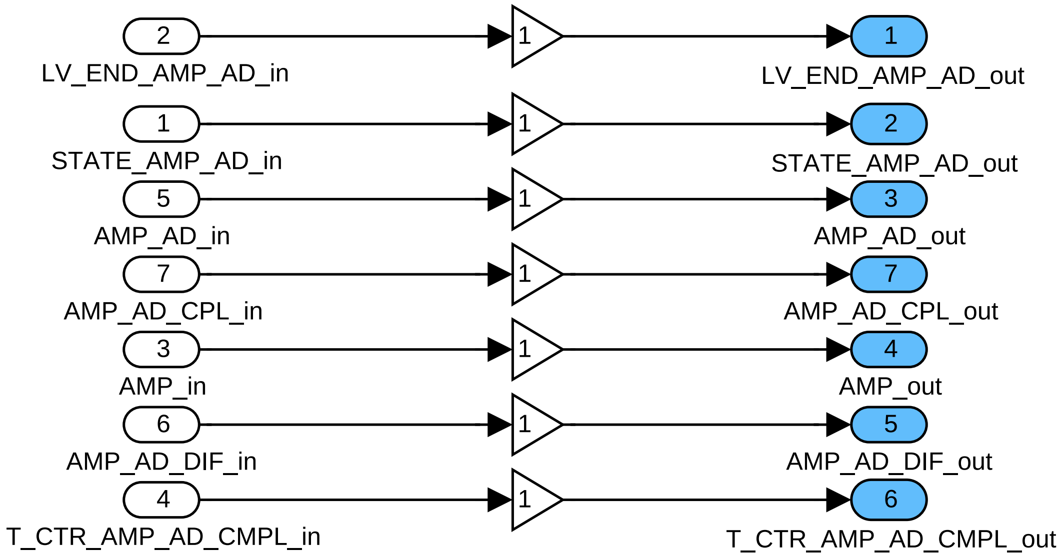


Figure 244 INSY_ADCPRAMP0/ OPM_100MS/ CLC/ CDN_VLD/ NOT_CLC/ NOT_CLC

Learning of ambient pressure Via PUT Sensor.

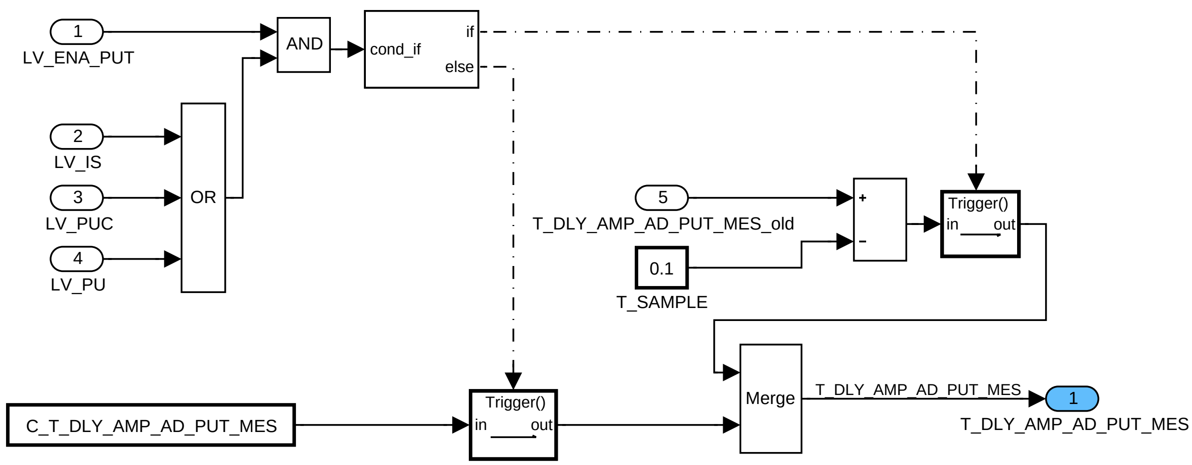


Figure 245 INSY_ADCPRAMP0/ OPM_100MS/ CLC_PUT_MES

4.16.1.4 Recurrence: 1s

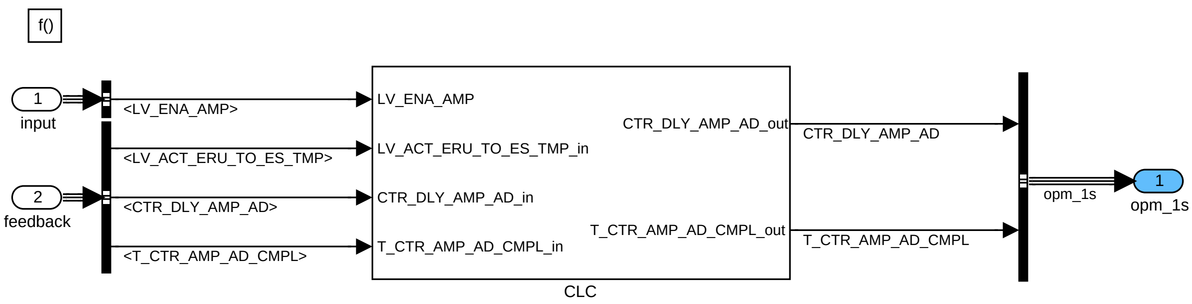


Figure 246 INSY_ADCPRAMP0/ OPM_1S

<u>Calculation of CTR_DLY_AMP_AD and T_CTR_AMP_AD_CMPL:</u>

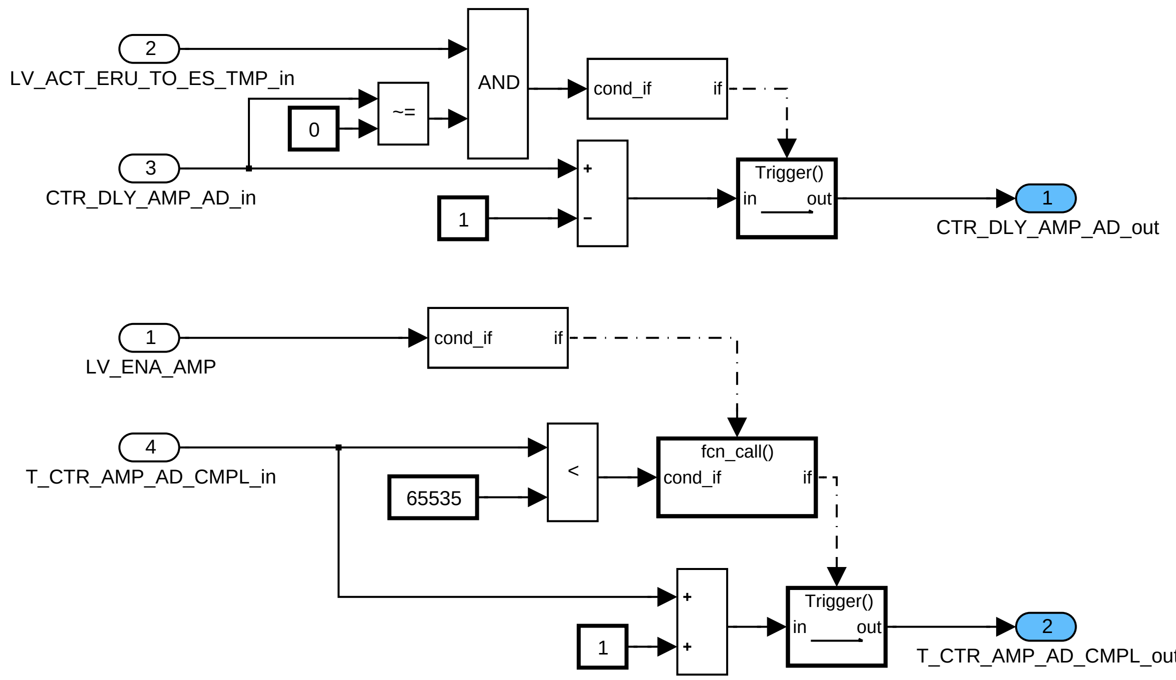


Figure 247 INSY_ADCPRAMP0/ OPM_1S/ CLC

4.16.1.5 Recurrence: Segment synchronous

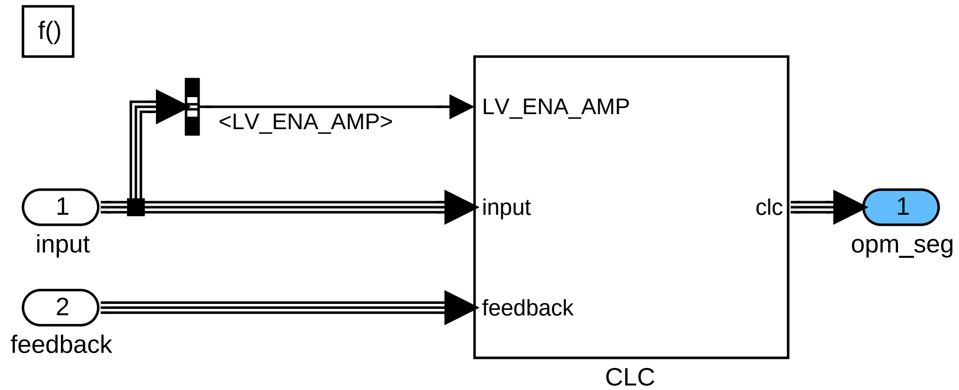


Figure 248 INSY_ADCPRAMP0/ OPM_SEG

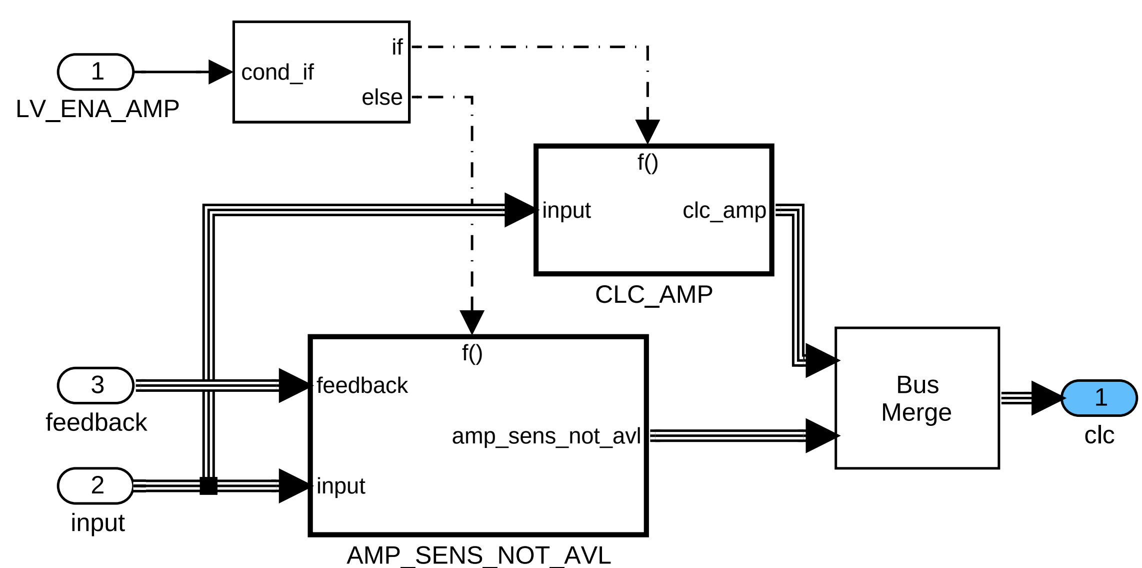


Figure 249 INSY_ADCPRAMP0/ OPM_SEG/ CLC

AMP sensor not available: Learn AMP

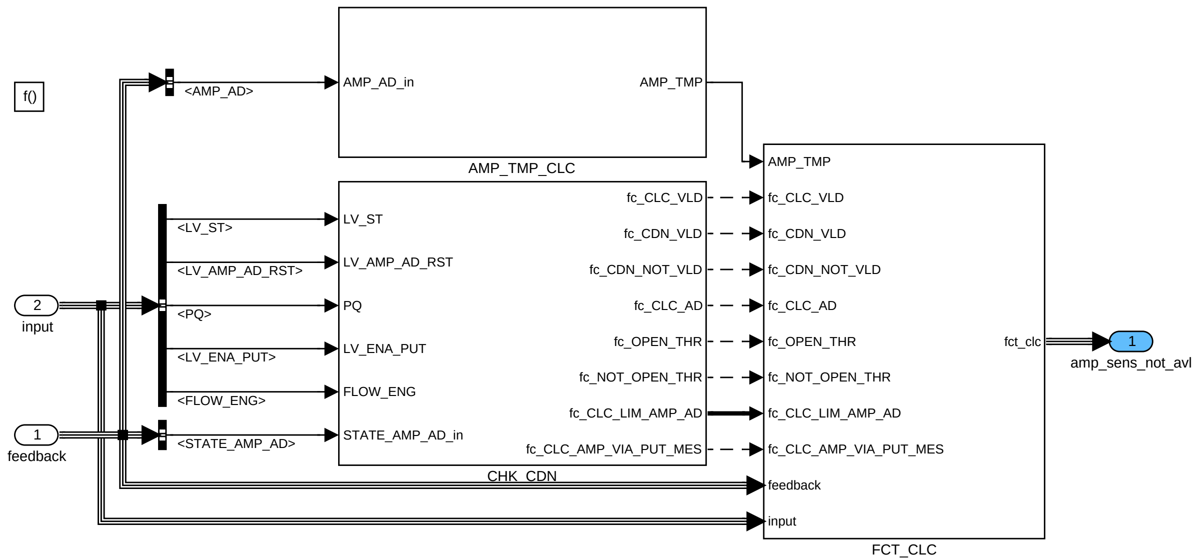


Figure 250 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL

Learn AMP: Calculation of AMP_TMP

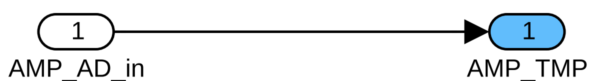


Figure 251 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ AMP_TMP_CLC

Learn AMP: Checking conditions for activation of different possibilities to learn AMP

If the system is equipped with a PUT sensor, the adaptation of the Ambient Pressure can be done using its signal at the Engine State IS, PUC or PU.

To ensure that ambient Pressure is learned at Stable system a delay timer is used in the System

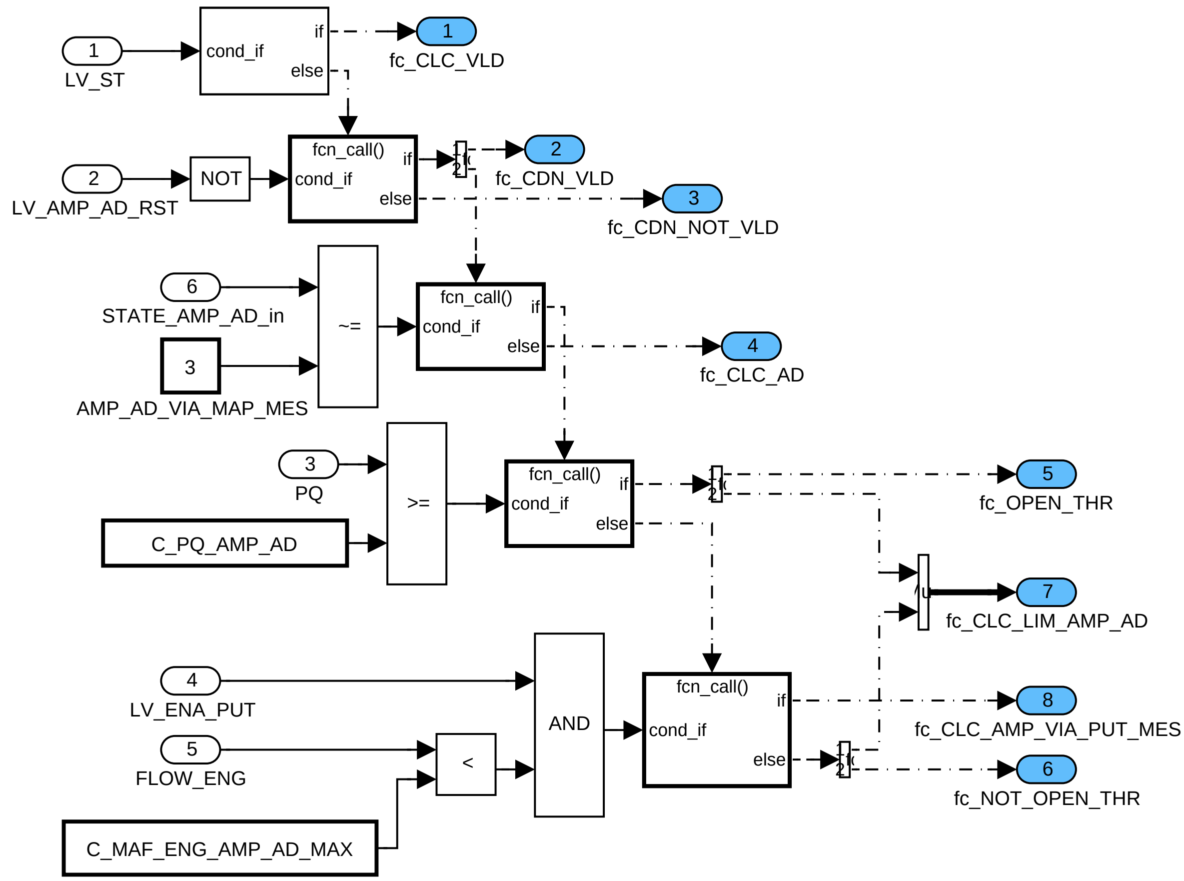


Figure 252 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ CHK_CDN

Learn AMP: Overview over different possibilities to learn AMP

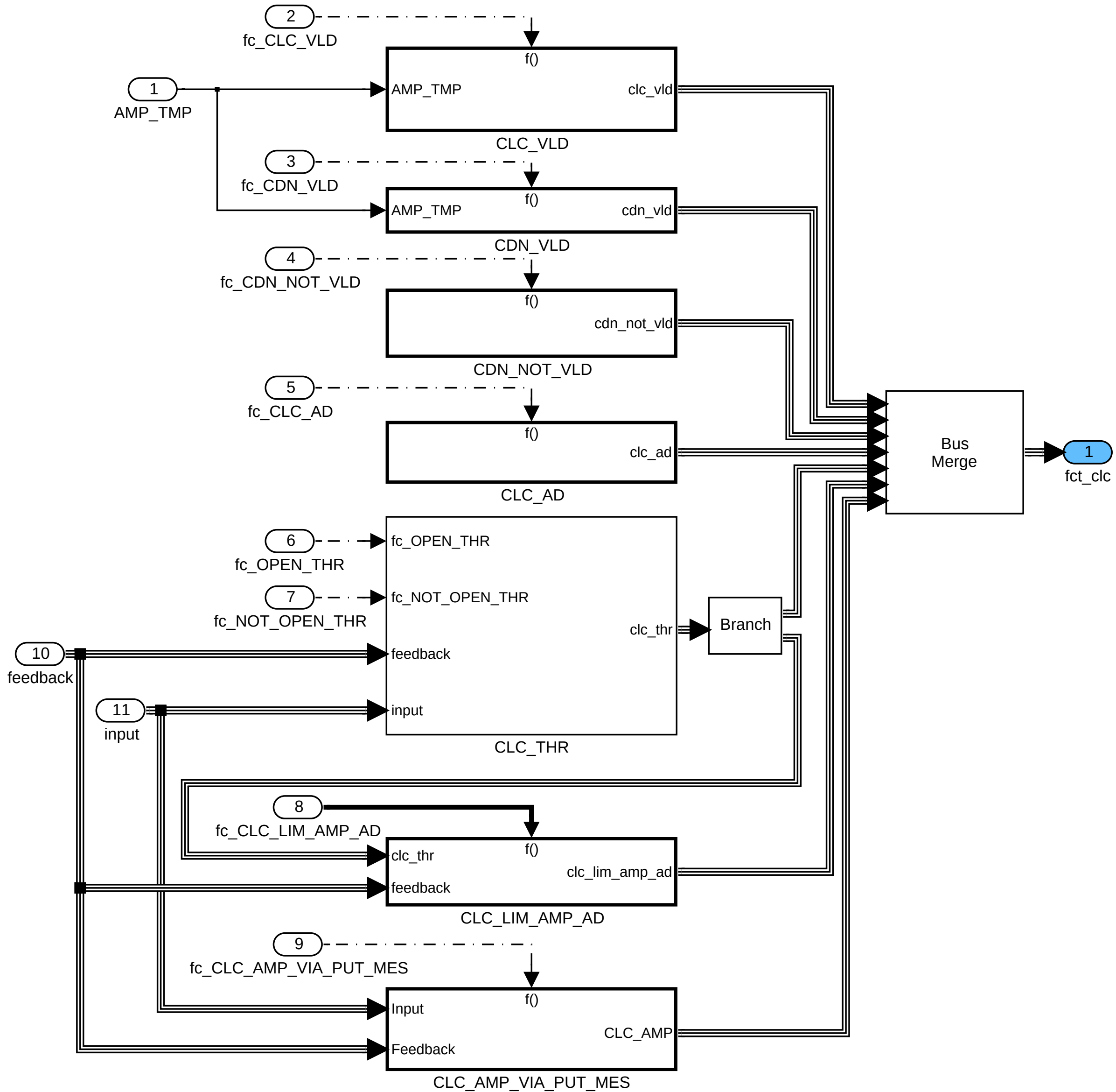


Figure 253 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC

Branch block for chapter "AMP sensor not available: Learn AMP":

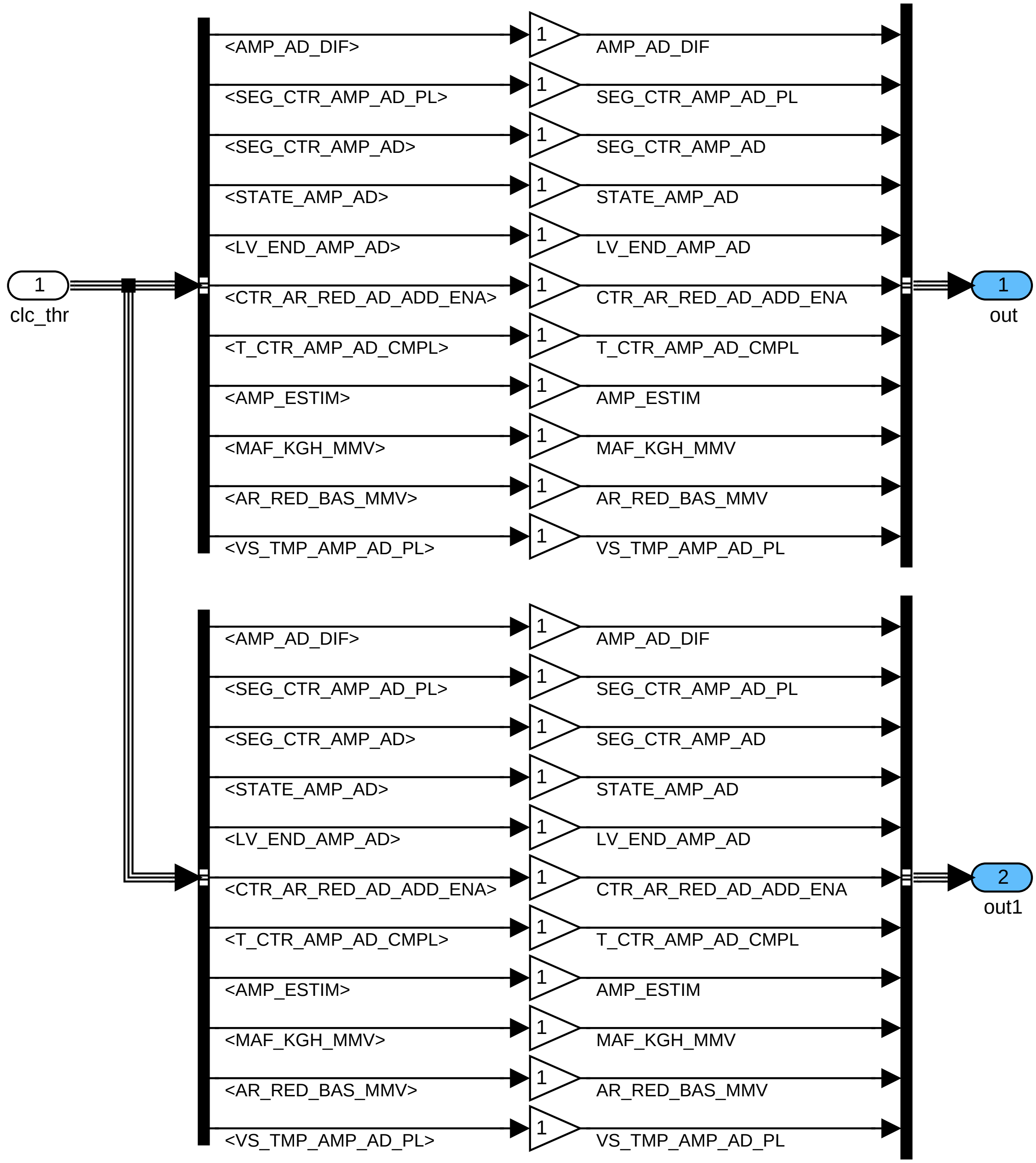


Figure 254 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ Branch

Bus merge for chapter "AMP sensor not available: Learn AMP":

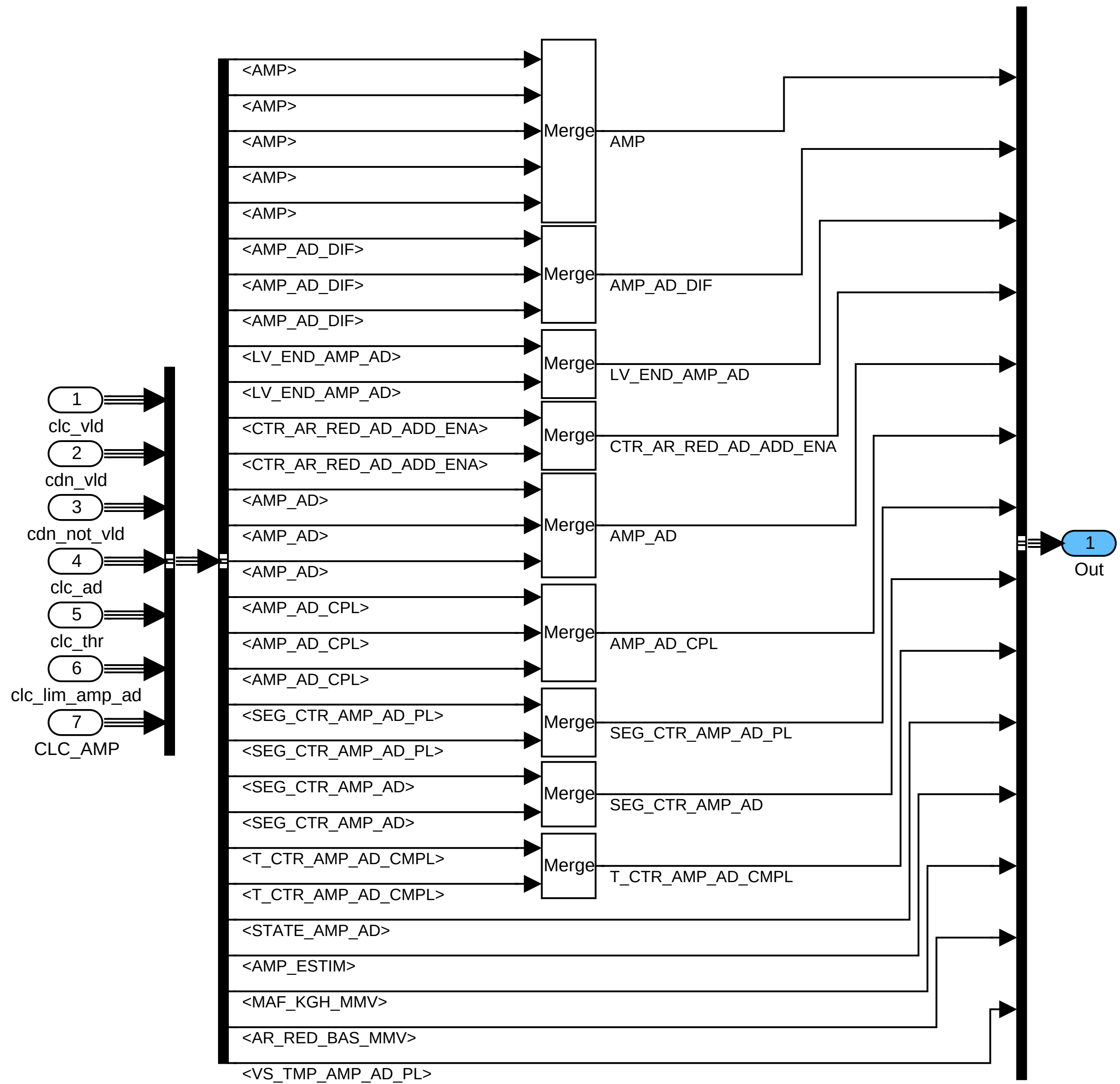


Figure 255 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ BusMerge

Learn AMP: Initialization of AMP (LV_ST = 0 and LV_AMP_AD_RST = 1)

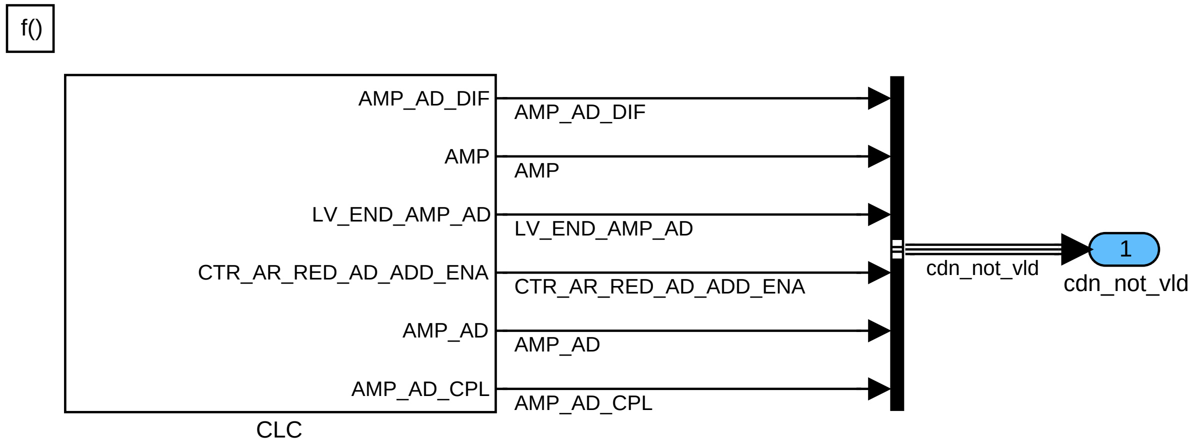


Figure 256 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CDN_NOT_VLD

Learn AMP: Initialization of AMP (LV_ST = 0 and LV_AMP_AD_RST = 1)

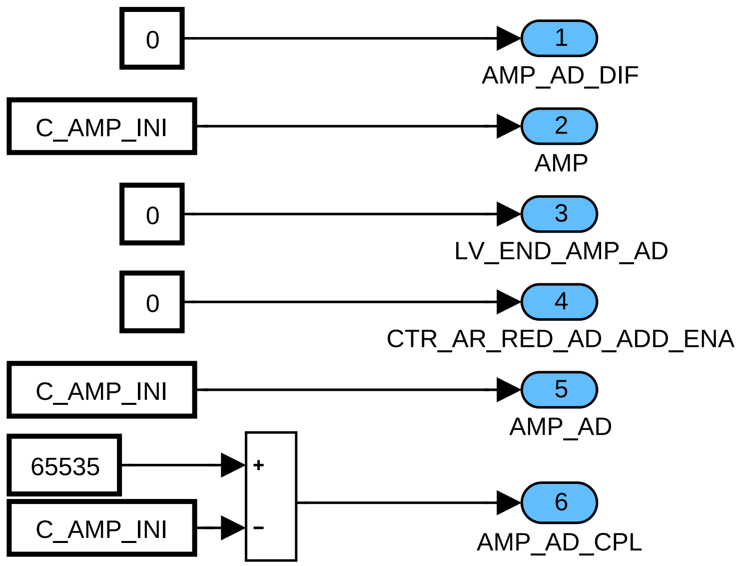


Figure 257 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CDN_NOT_VLD/ CLC

Learn AMP: Calculation of AMP via AMP_TMP (LV_ST = 0 and LV_AMP_AD_RST = 0)

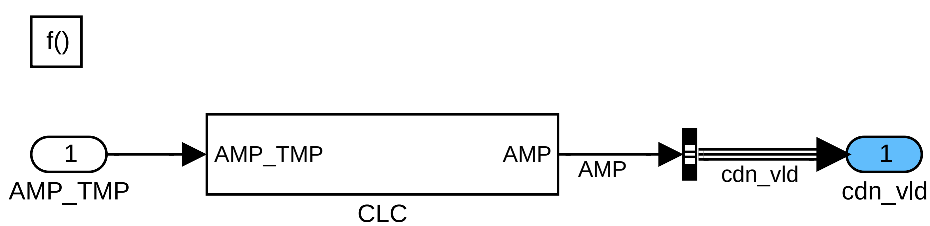


Figure 258 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CDN_VLD

Learn AMP: Calculation of AMP via AMP_TMP (LV_ST = 0 and LV_AMP_AD_RST = 0)

AMP = AMP_AD or AMP = AMP_SUB. This depends on the configuration: charged engine or not.

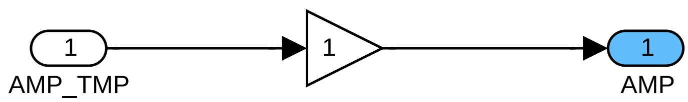


Figure 259 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CDN_VLD/ CLC

Learn AMP: Initialization of SEG_CTR_AMP_AD (LV_ST = 0 and LV_AMP_AD_RST = 0 and STATE_AMP_AD = 3)

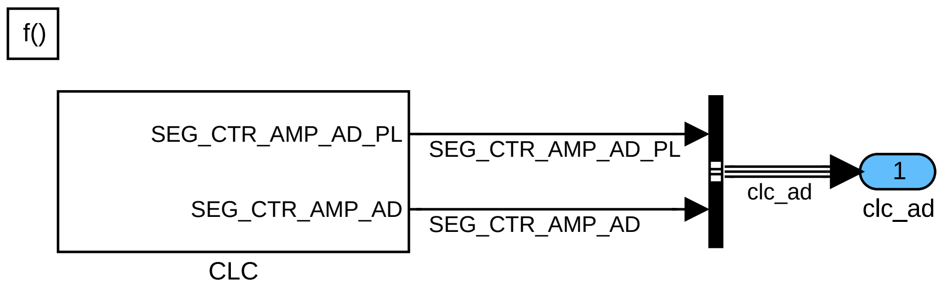


Figure 260 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_AD

Learn AMP: Initialization of SEG_CTR_AMP_AD (LV_ST = 0 and LV_AMP_AD_RST = 0 and STATE_AMP_AD = 3)

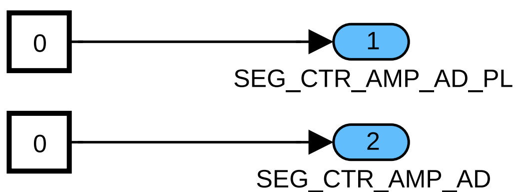


Figure 261 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_AD/ CLC

Learn AMP: Calculation of AMP via PUT_MES

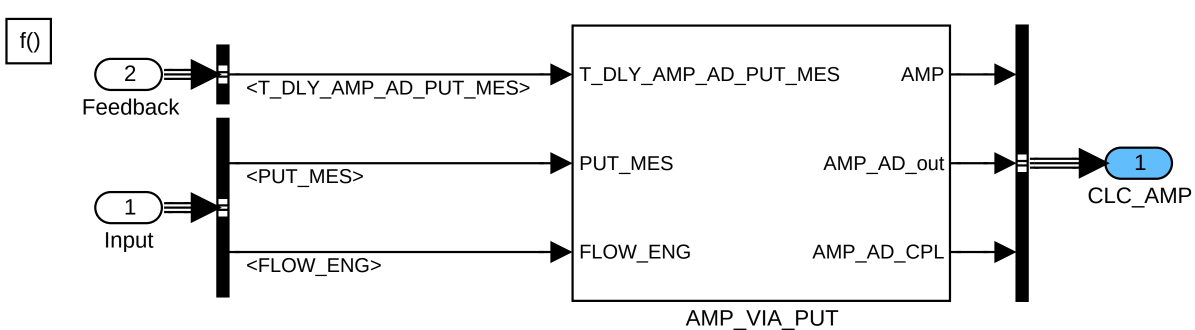


Figure 262 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_AMP_VIA_PUT_MES

Learn AMP: Calculation of AMP via PUT_MES

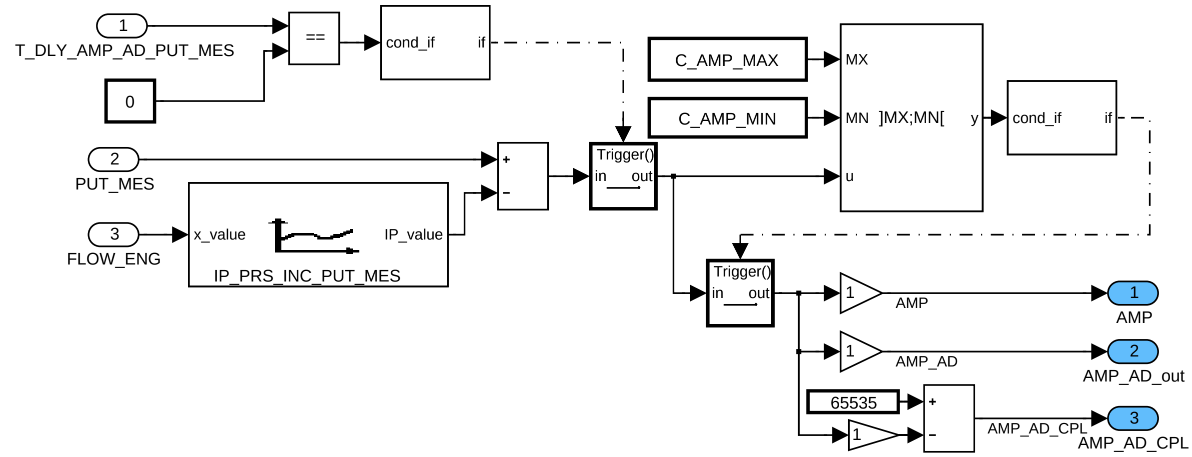


Figure 263 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_AMP_VIA_PUT_MES/ AMP_VIA_PUT

Calculation of AMP_AD = AMP_AD_old + AMP_AD_DIF and limitation of AMP_AD

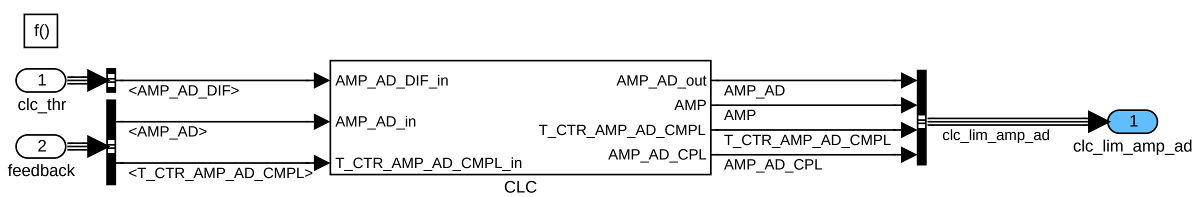


Figure 264 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_LIM_AMP_AD

Calculation of AMP_AD = AMP_AD_old + AMP_AD_DIF and limitation of AMP_AD

This subsystem is calculated always after the calculation of chapter "Learn AMP at open throttle" if it was activated and also after the calculation of chapter "Learn AMP at not open throttle" if it was activated.

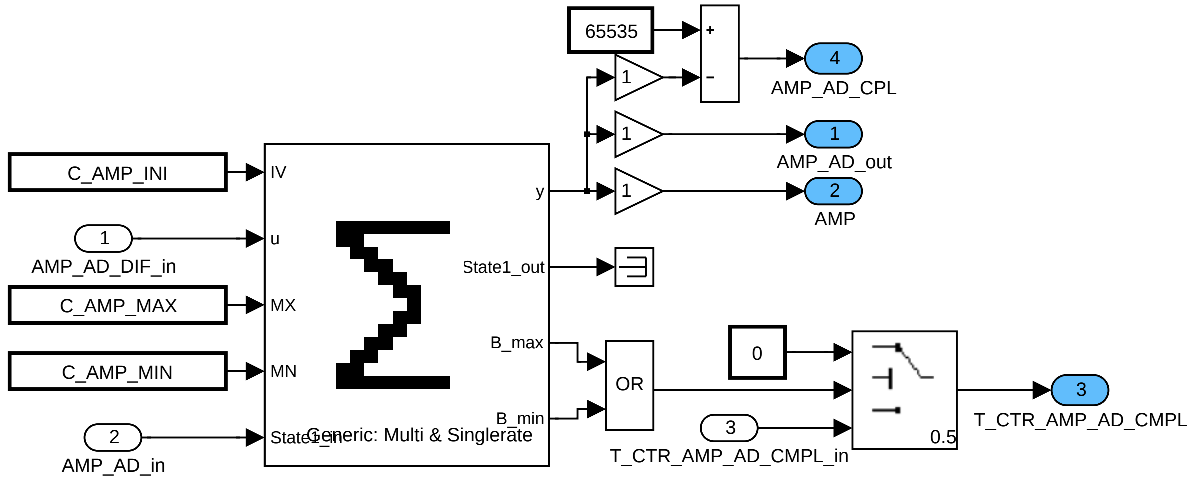


Figure 265 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_LIM_AMP_AD/ CLC

Learn AMP at open throttle or at not open throttle:

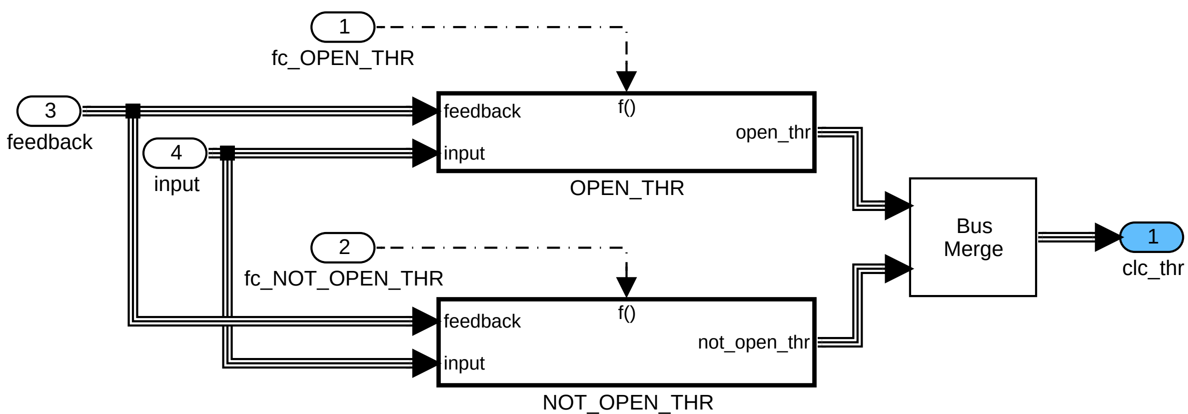


Figure 266 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR

Bus merge for chapter "Learn AMP at open throttle or at not open throttle":

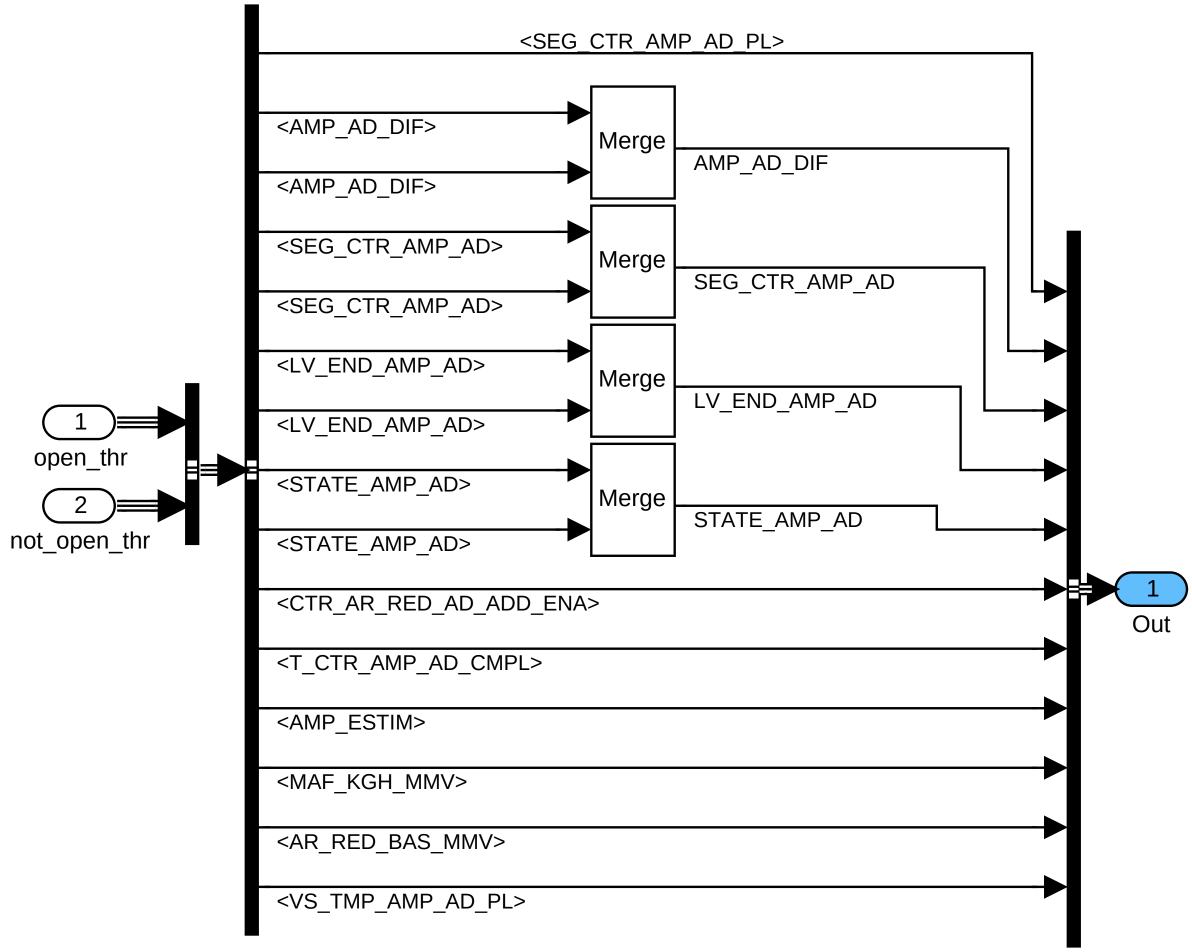


Figure 267 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ BusMerge1

Learn AMP at not open throttle:

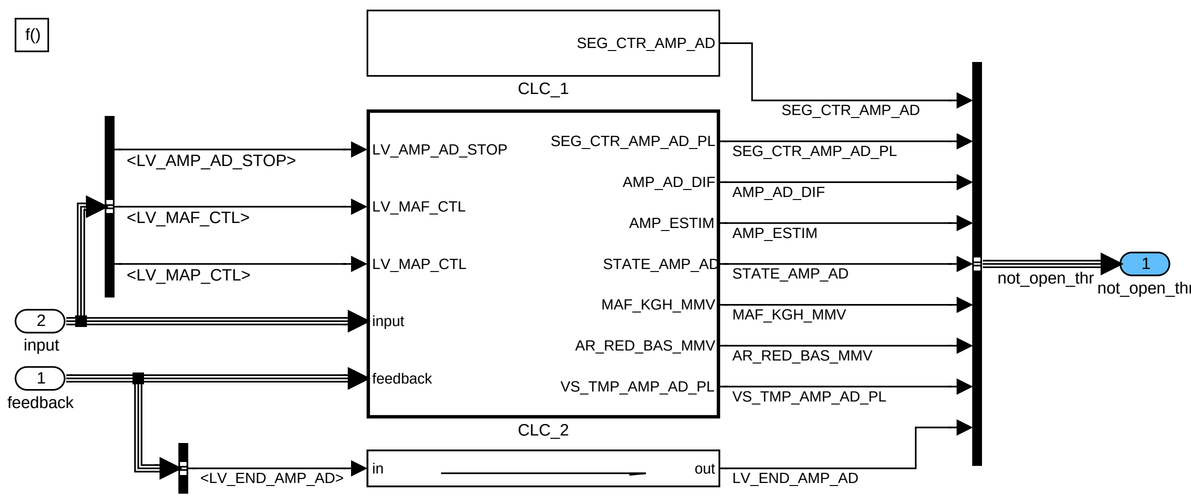


Figure 268 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR

Learn AMP at not open throttle: Initialization of SEG_CTR_AMP_AD

Figure 269 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_1

Learn AMP at not open throttle: Calculation of AMP_AD_DIF

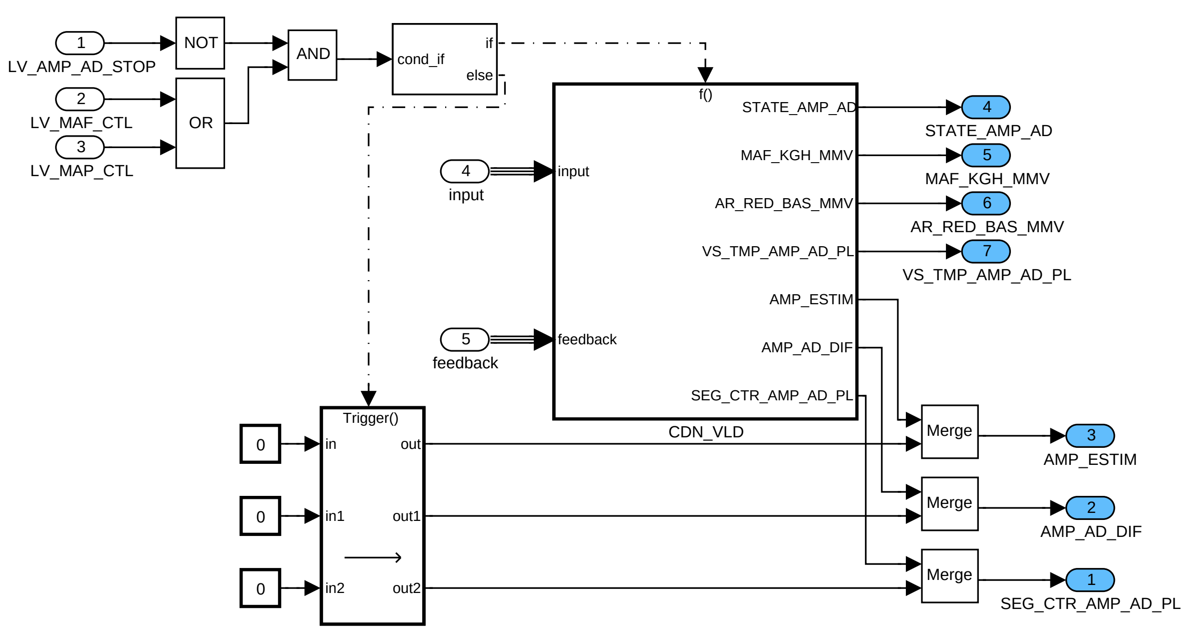


Figure 270 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2

Learn AMP at not open throttle: Overview over calculations for AMP_AD_DIF

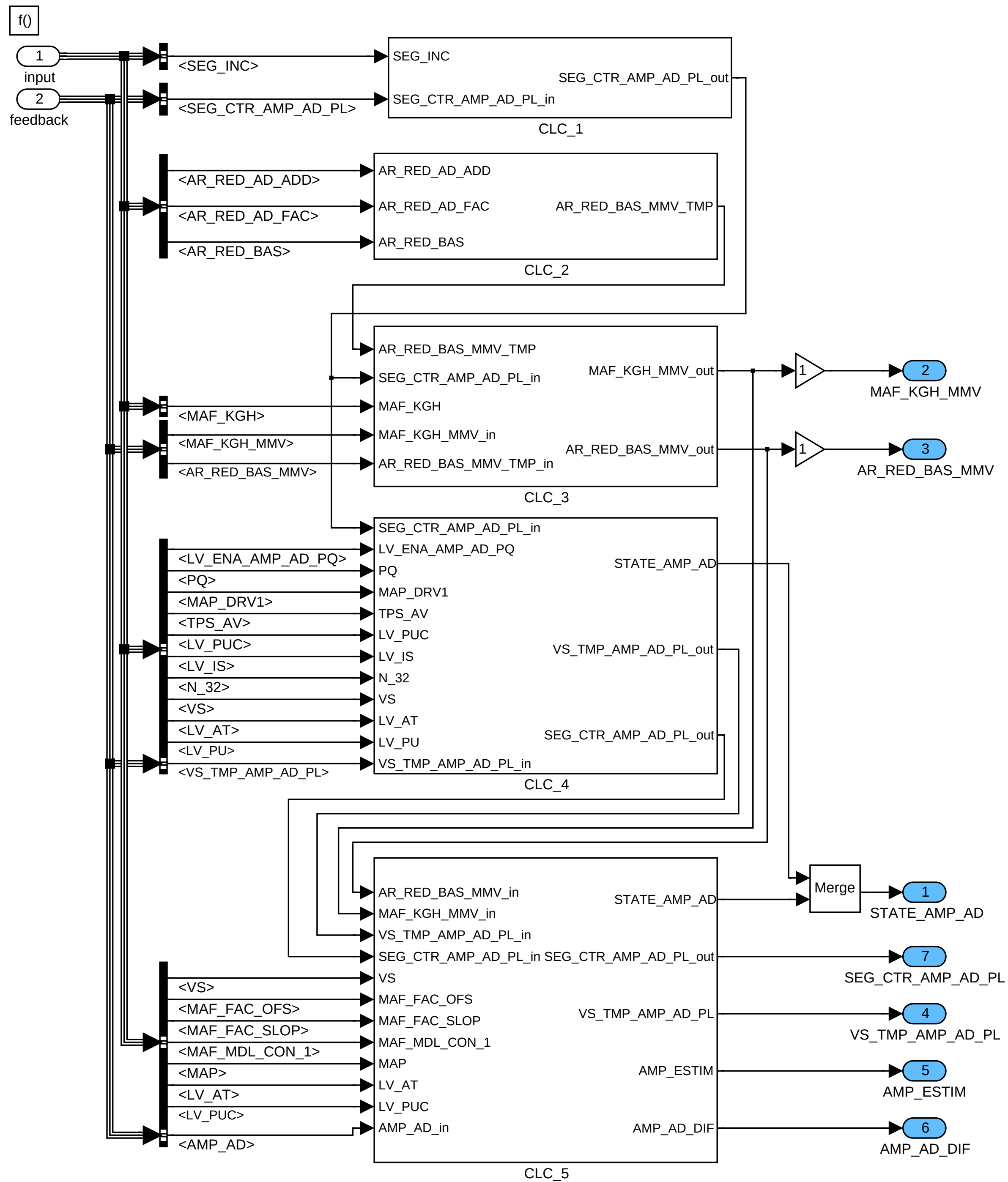


Figure 271 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD

Learn AMP at not open throttle: Incrementing of counter SEG_CTR_AMP_AD_PL

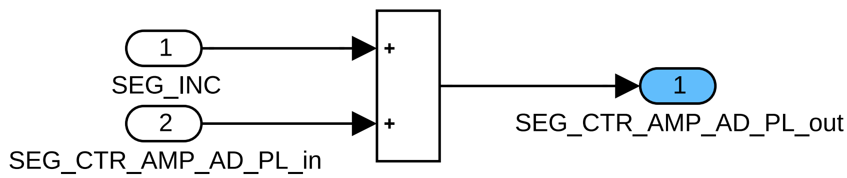


Figure 272 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_1

Learn AMP at not open throttle: Correction of AR_RED_BAS: AR_RED_BAS_MMV_TMP

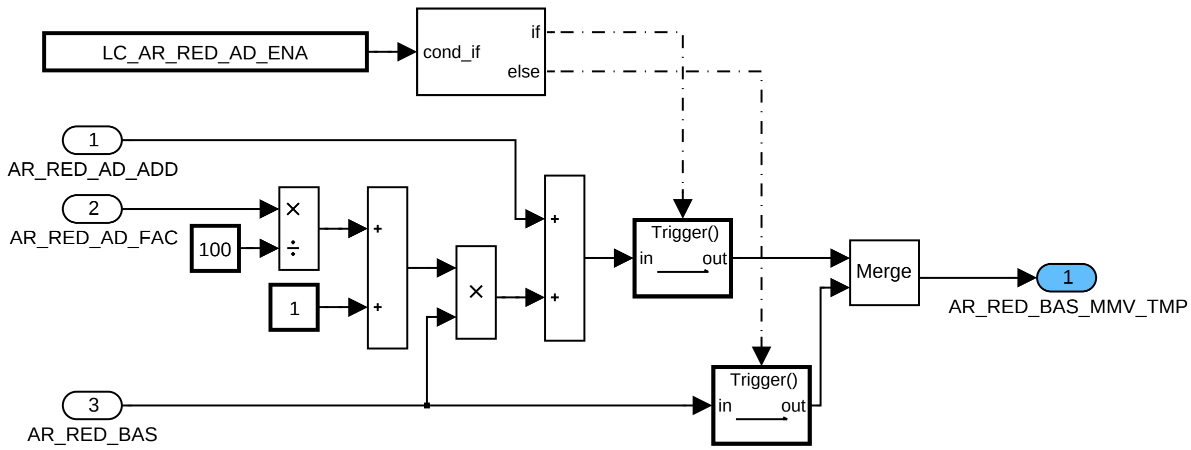


Figure 273 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_2

Learn AMP at not open throttle: Calculation of MAF_KGH_MMV and AR_RED_BAS_MMV

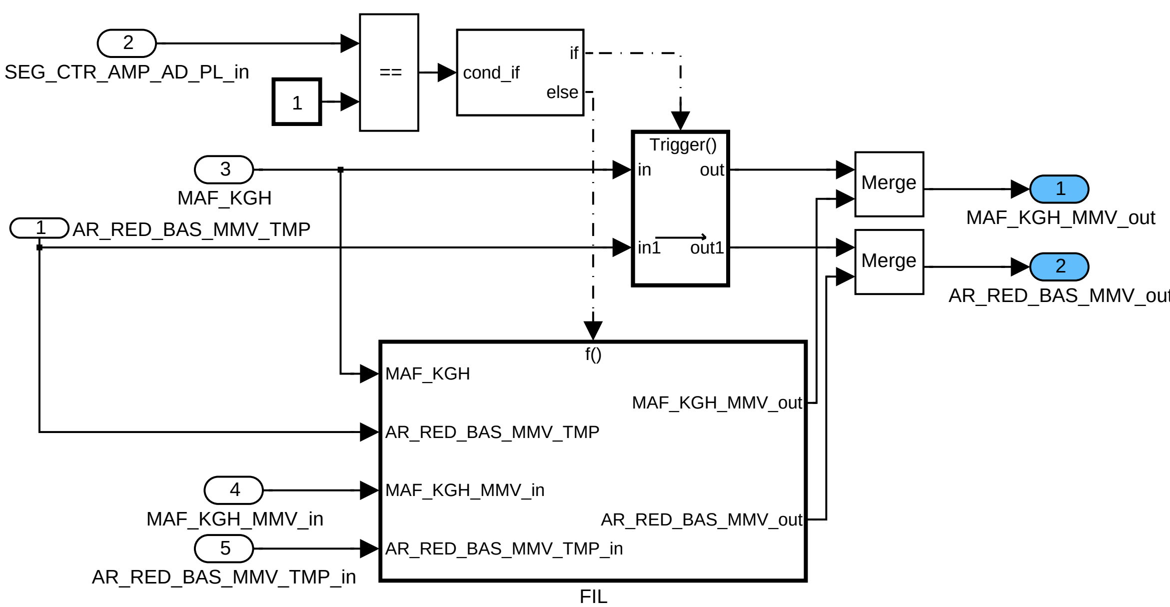


Figure 274 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_3

Learn AMP at not open throttle: Filtering of MAF_KGH and AR_RED_BAS_MMV_TMP

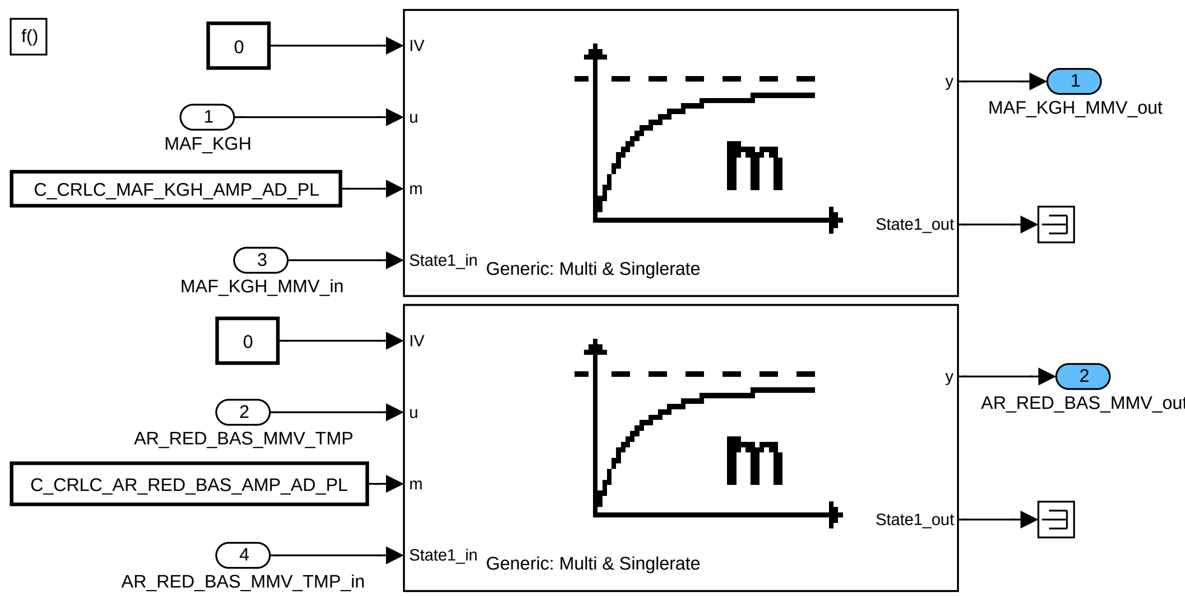


Figure 275 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_3/ FIL

Learn AMP at not open throttle: Check condition whether you are in engine state PUC or IS to enable AMP adaptation via model

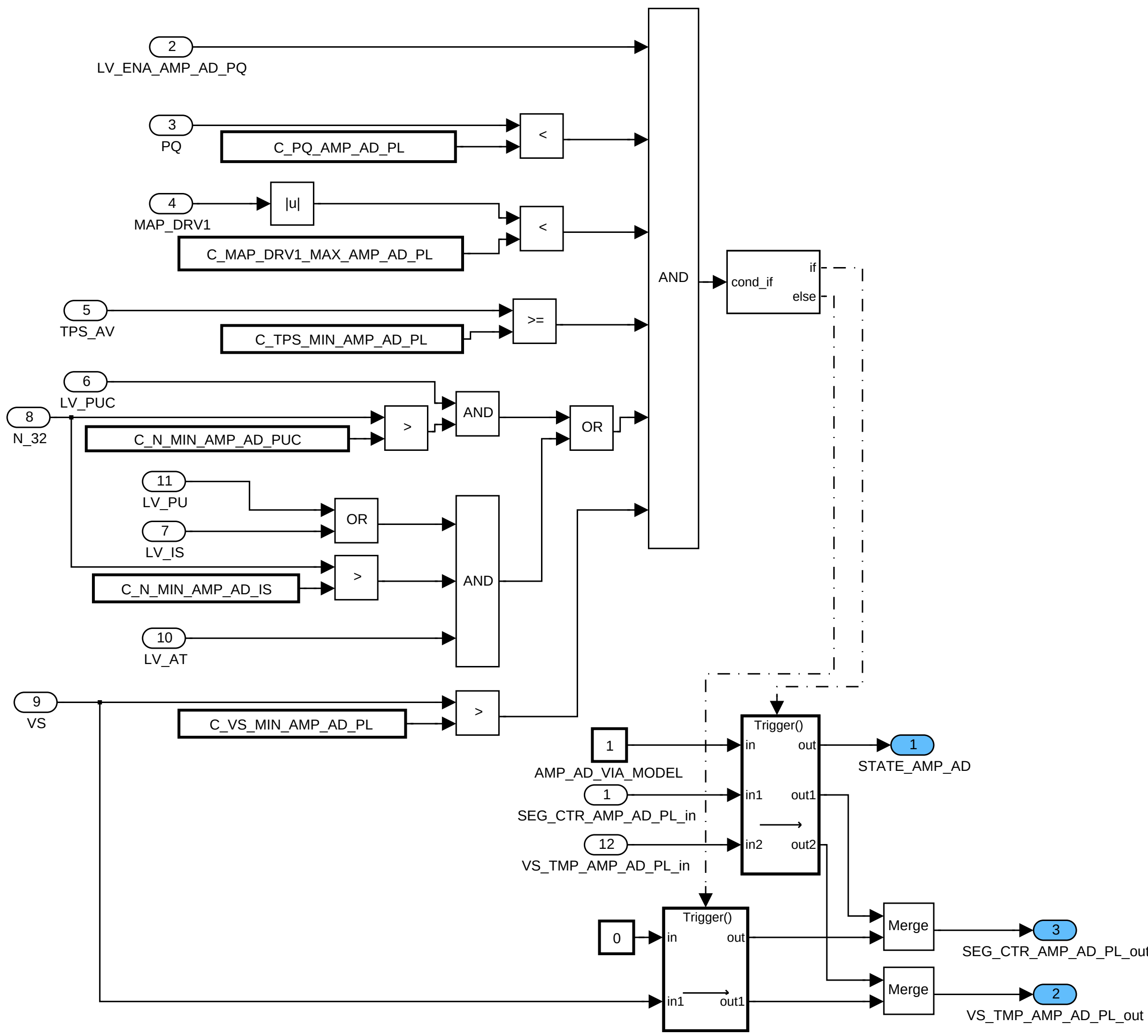


Figure 276 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_4

Learn AMP at not open throttle: AMP adaptation in engine state PUC or IS if SEG_CTR_AMP_AD_PL = C_SEG_CTR_AMP_AD_PL

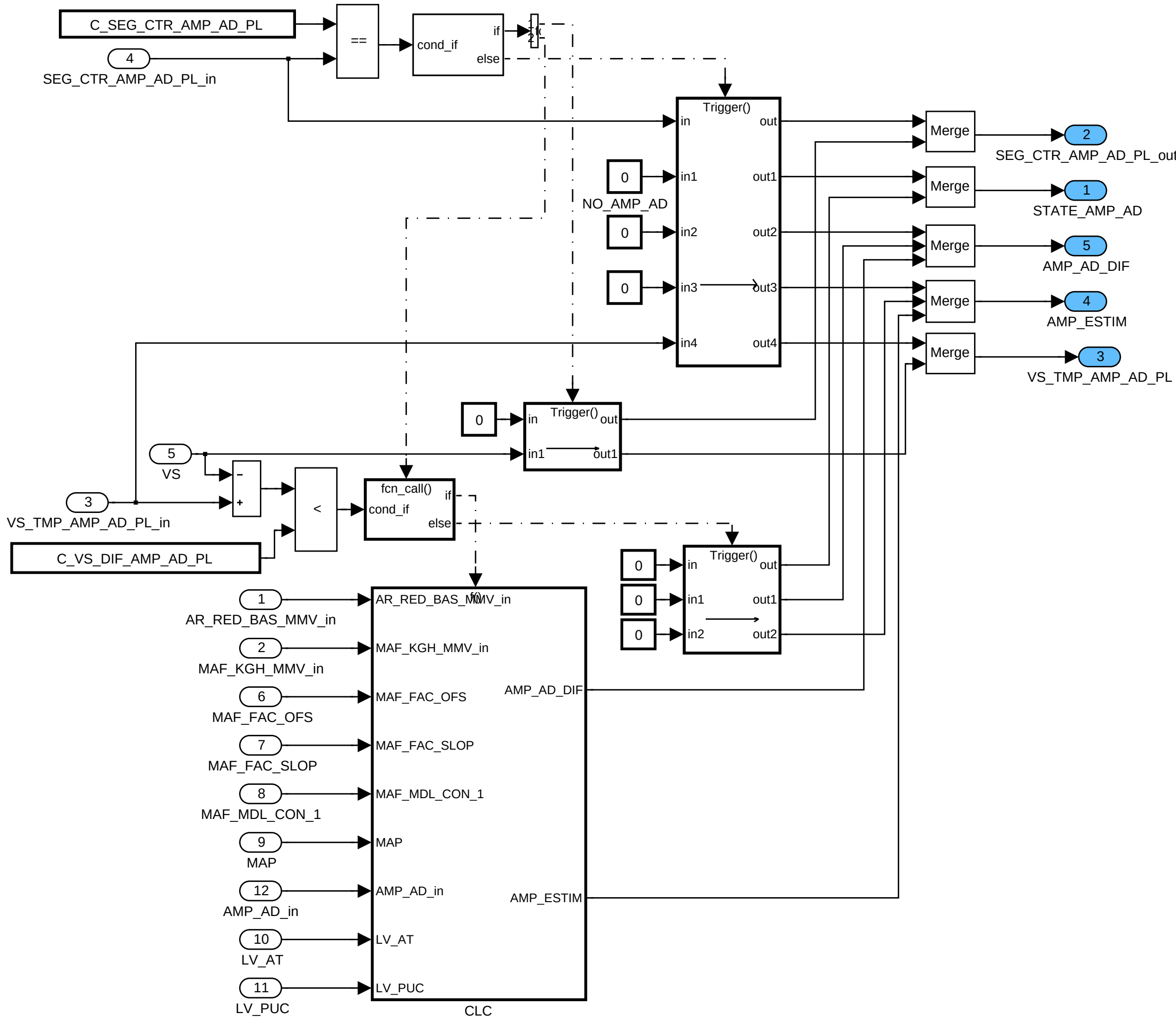


Figure 277 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_5

Learn AMP at not open throttle: Calculation of AMP_AD_DIF in engine state PUC or IS and calculation of AMP_ESTIM

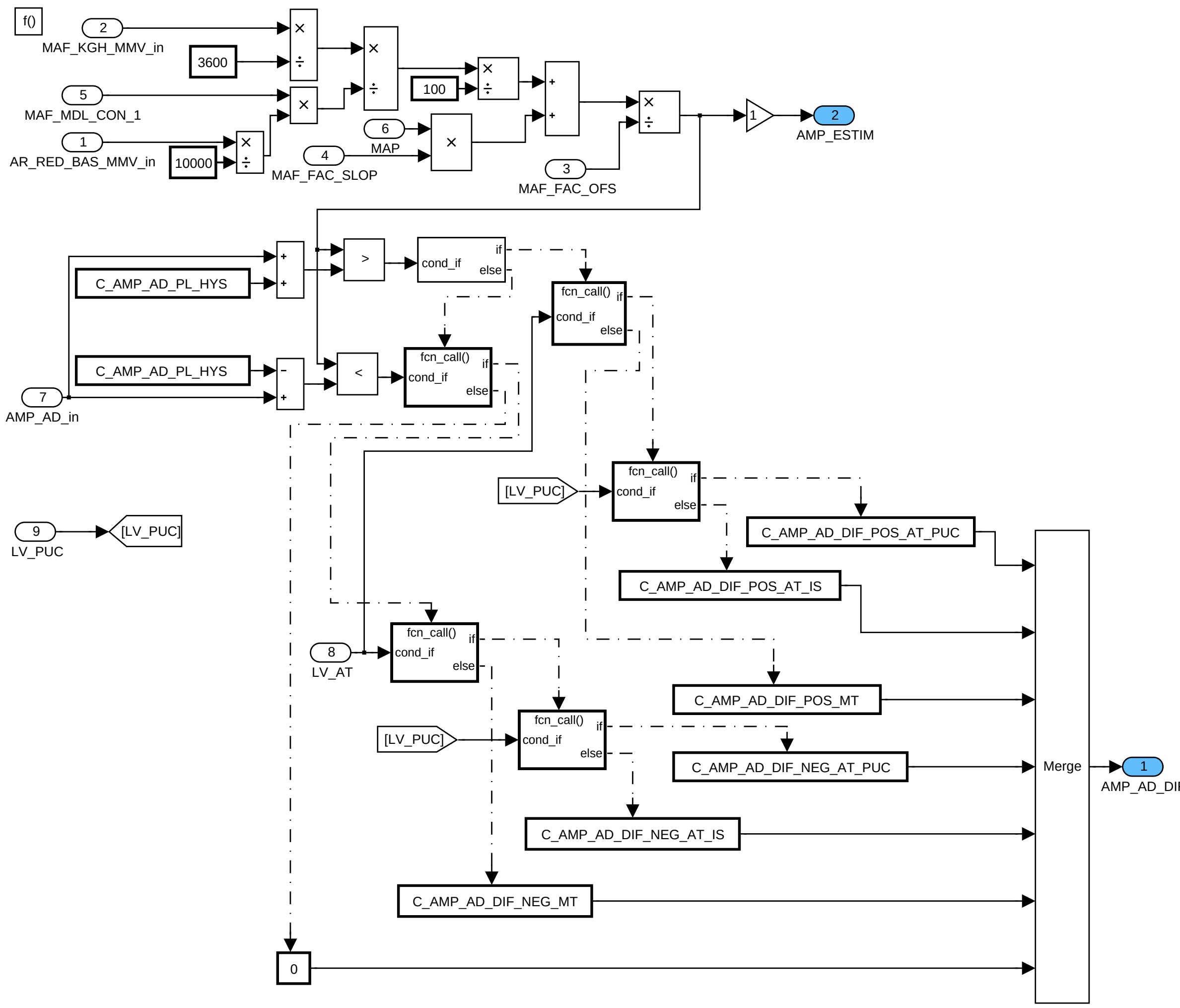


Figure 278 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ NOT_OPEN_THR/ CLC_2/ CDN_VLD/ CLC_5/ CLC

Learn AMP at open throttle (LV_ST = 0 and LV_AMP_AD_RST = 0 and STATE_AMP_AD != 3 and PQ >= C_PQ_AMP_AD):

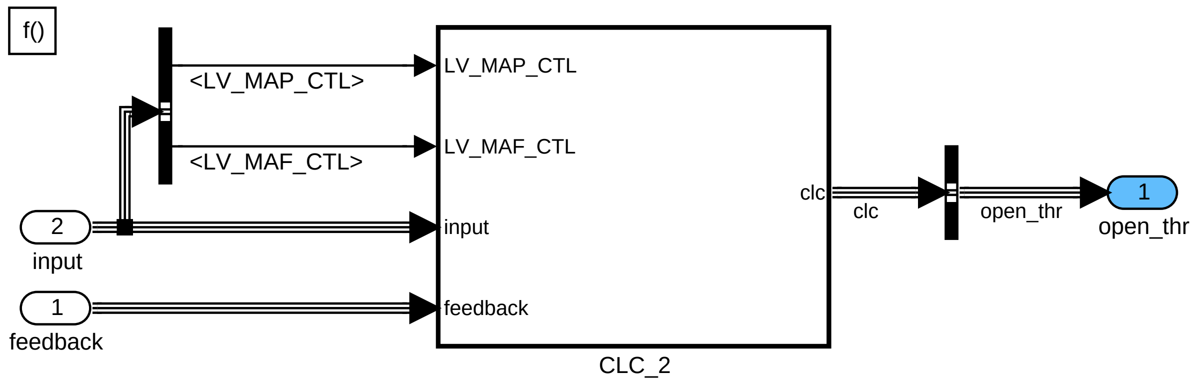


Figure 279 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR

Learn AMP at open throttle (LV_ST = 0 and LV_AMP_AD_RST = 0 and STATE_AMP_AD != 3 and PQ >= C_PQ_AMP_AD):

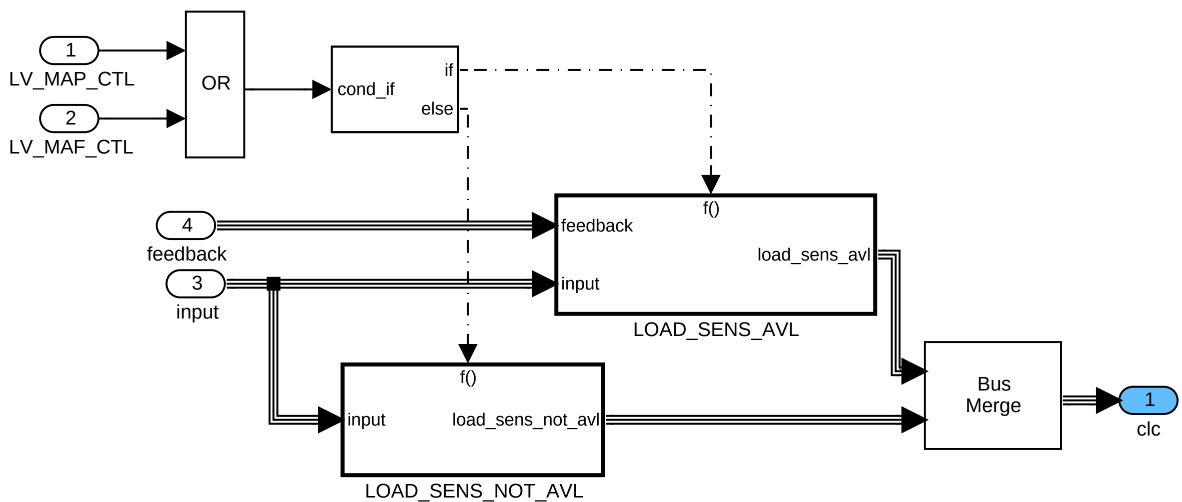


Figure 280 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2

Bus merge for chapter "Learn AMP at open throttle":

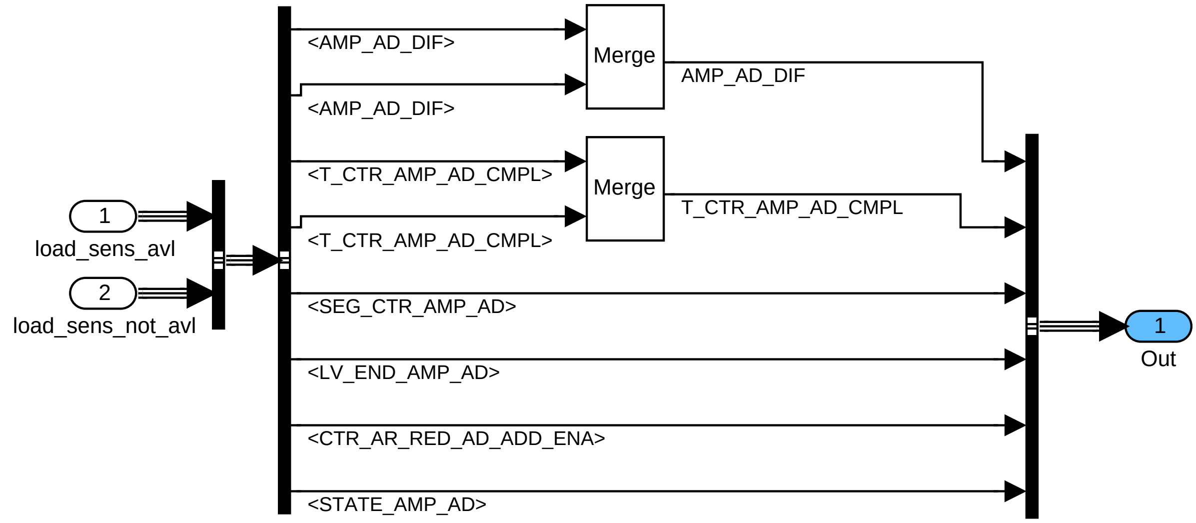


Figure 281 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ BusMerge

Learn AMP at open throttle with load sensor available:

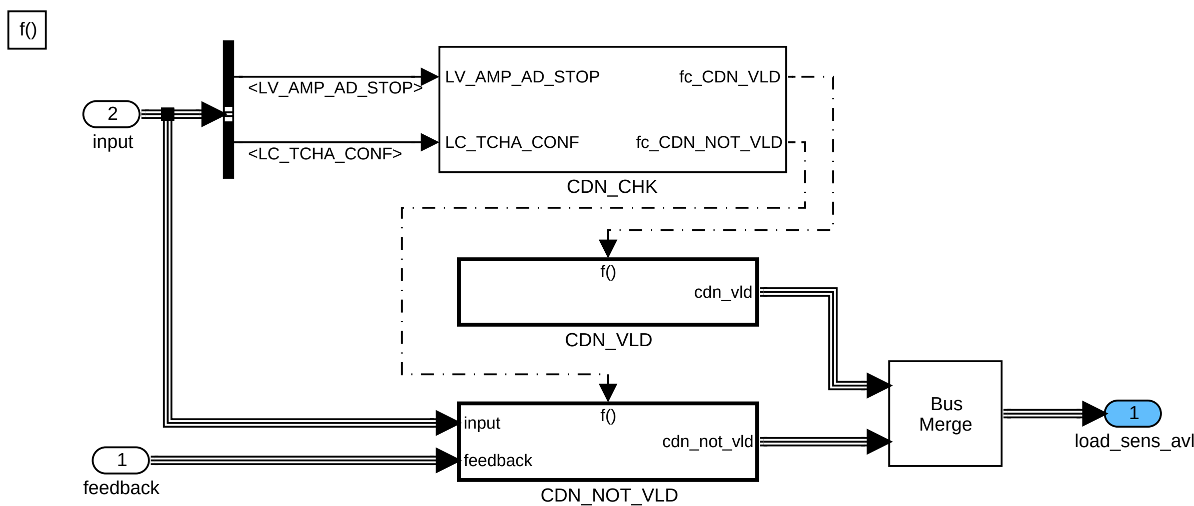


Figure 282 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL

Bus merge for chapter "Learn AMP at open throttle with load sensor available":

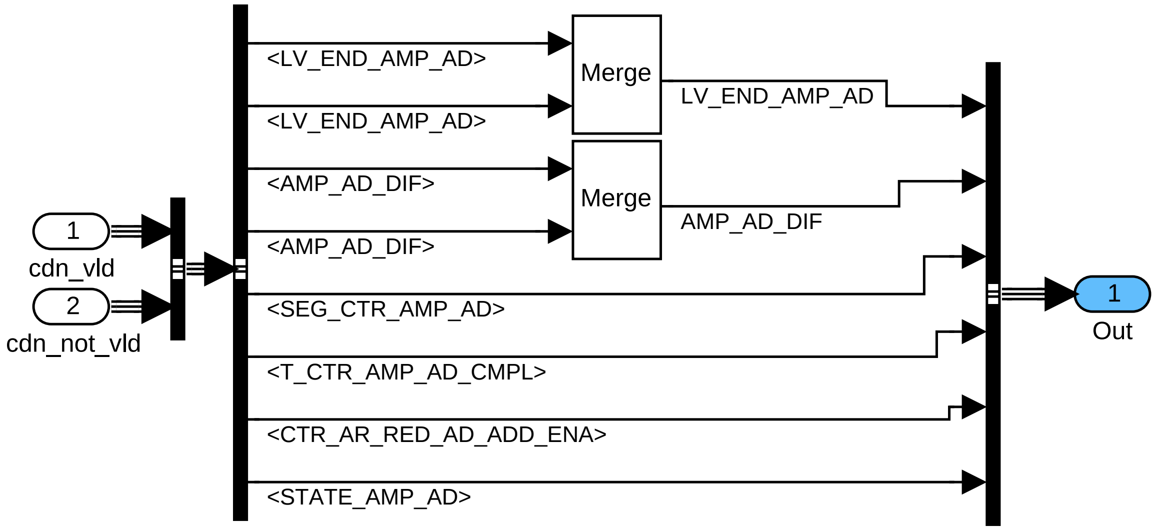


Figure 283 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ BusMerge

Learn AMP at open throttle with load sensor available: Check condition to initialize or to calculate AMP_AD_DIF and LV_END_AMP_AD

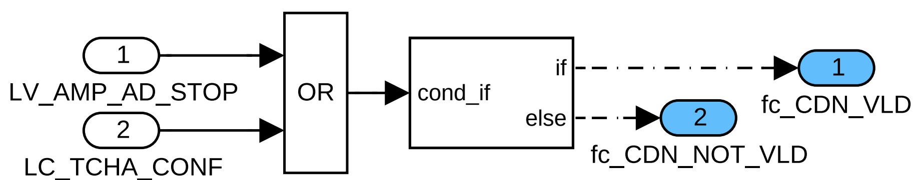


Figure 284 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_CHK

Learn AMP at open throttle with load sensor available: Calculation of AMP_AD_DIF and LV_END_AMP_AD

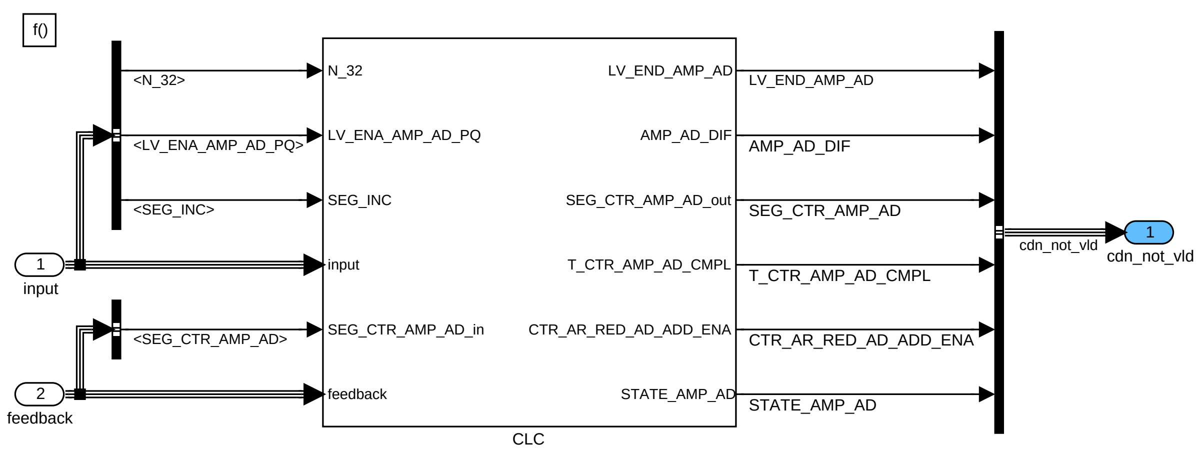


Figure 285 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD

Learn AMP at open throttle with load sensor available: Calculation of AMP_AD_DIF and LV_END_AMP_AD

The calculation of AMP_AD_DIF and LV_END_AMP_AD depends on the variable LV_ENA_AMP_AD_PQ and N_32, otherwise AMP_AD_DIF is initialized with 0.

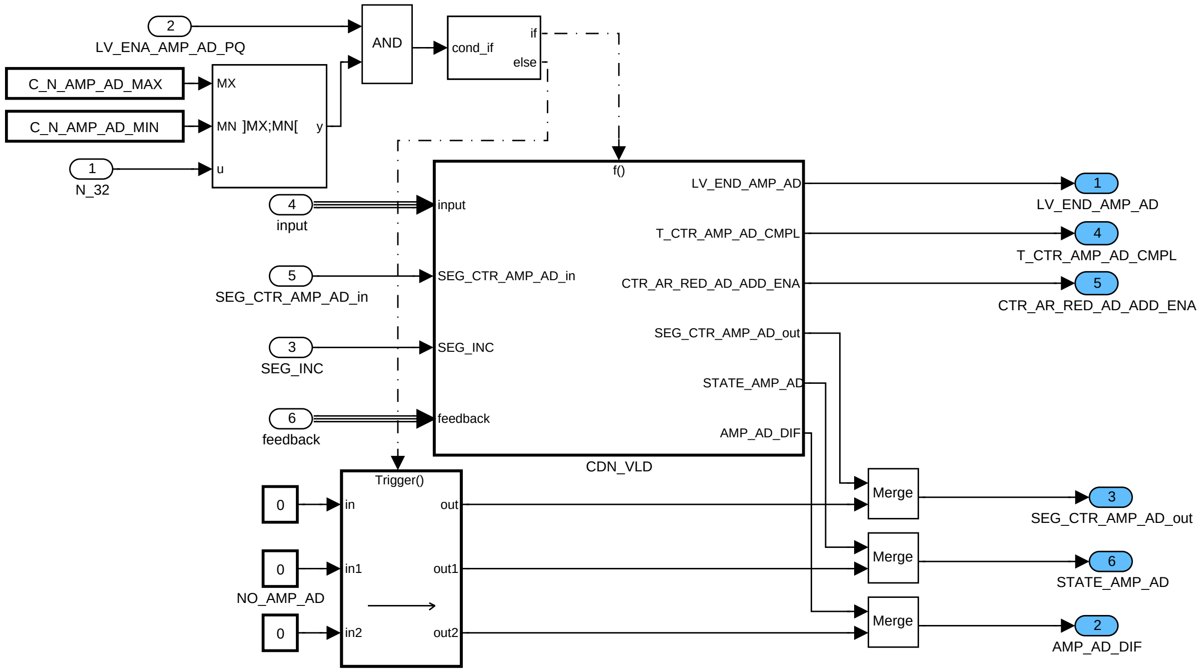


Figure 286 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD/ CLC

Learn AMP at open throttle with load sensor available: Calculation of AMP_AD_DIF and LV_END_AMP_AD depending on SEG_CTR_AMP_AD

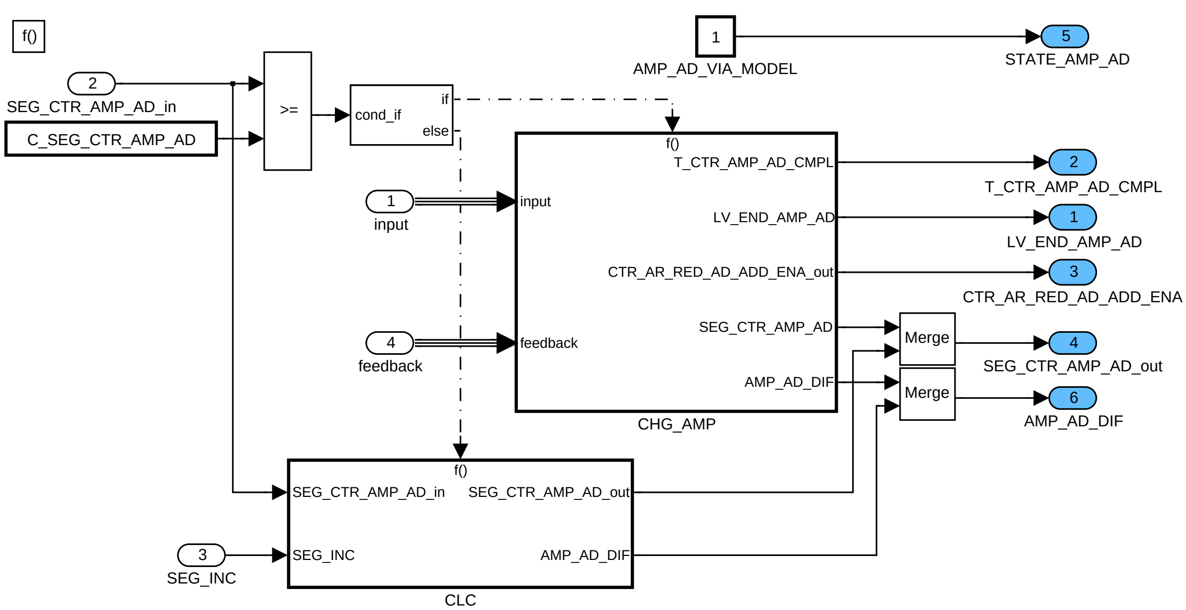


Figure 287 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD/ CLC/ CDN_VLD

Learn AMP at open throttle with load sensor available: Condition fulfilled for calculation of AMP_AD_DIF and LV_END_AMP_AD (SEG_CTR_AMP_AD >= C_SEG_CTR_AMP_AD)

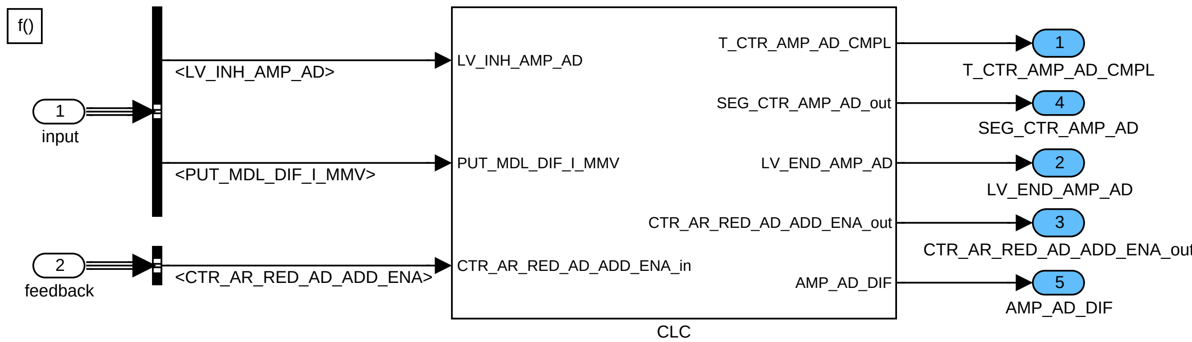


Figure 288 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD/ CLC/ CDN_VLD/ CHG_AMP

Learn AMP at open throttle with load sensor available: Condition fulfilled for calculation of AMP_AD_DIF and LV_END_AMP_AD (SEG_CTR_AMP_AD >= C_SEG_CTR_AMP_AD)

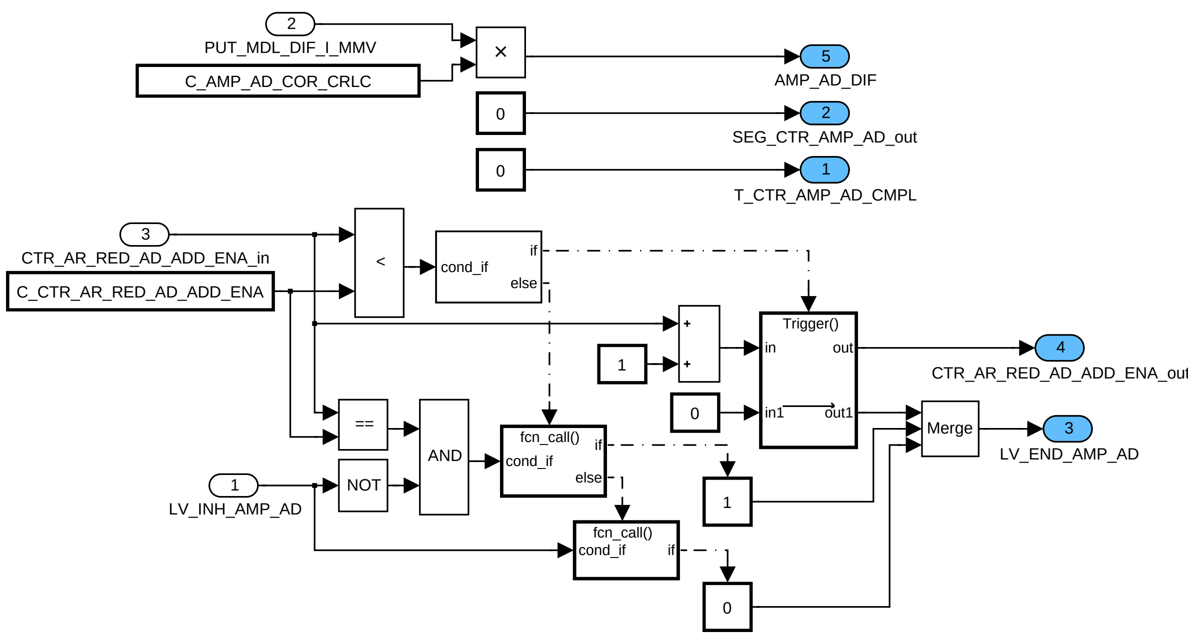


Figure 289 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD/ CLC/ CDN_VLD/ CHG_AMP/ CLC

Learn AMP at open throttle with load sensor available: Initialization of AMP_AD_DIF and incrementing of counter SEG_CTR_AMP_AD (SEG_CTR_AMP_AD < C_SEG_CTR_AMP_AD)

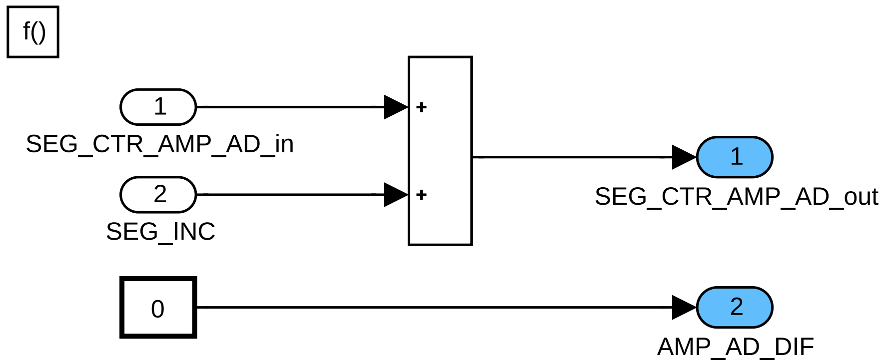


Figure 290 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_NOT_VLD/ CLC/ CDN_VLD/ CLC

Learn AMP at open throttle with load sensor available: Initialization of AMP_AD_DIF and LV_END_AMP_AD

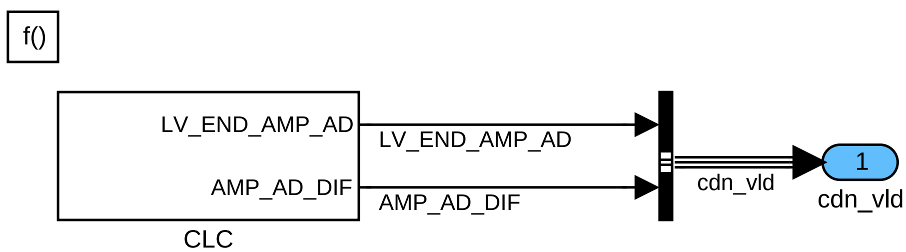


Figure 291 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_VLD

Learn AMP at open throttle with load sensor available: Initialization of AMP_AD_DIF and LV_END_AMP_AD

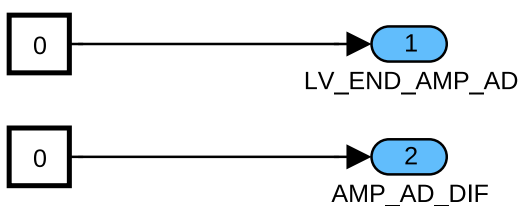


Figure 292 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_AVL/ CDN_VLD/ CLC

Learn AMP at open throttle with no load sensor available:

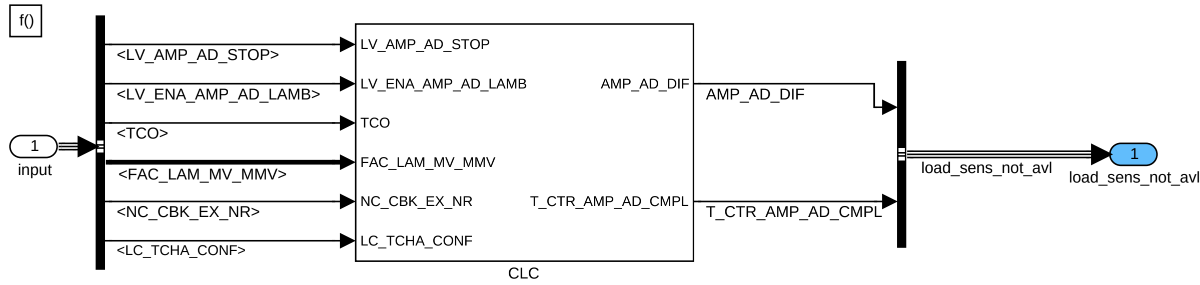


Figure 293 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_NOT_AVL

Learn AMP at open throttle with no load sensor available: Calculation of AMP_AD_DIF

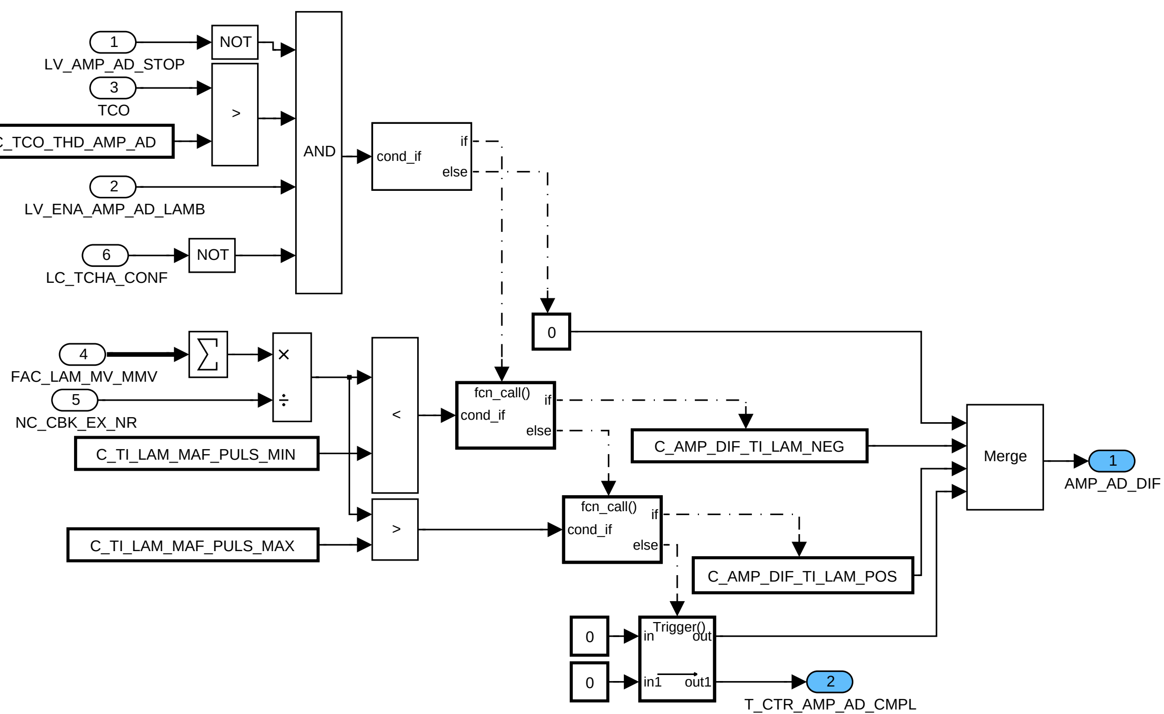


Figure 294 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_THR/ OPEN_THR/ CLC_2/ LOAD_SENS_NOT_AVL/ CLC

Learn AMP: Calculation of AMP via AMP_TMP (LV_ST = 1)

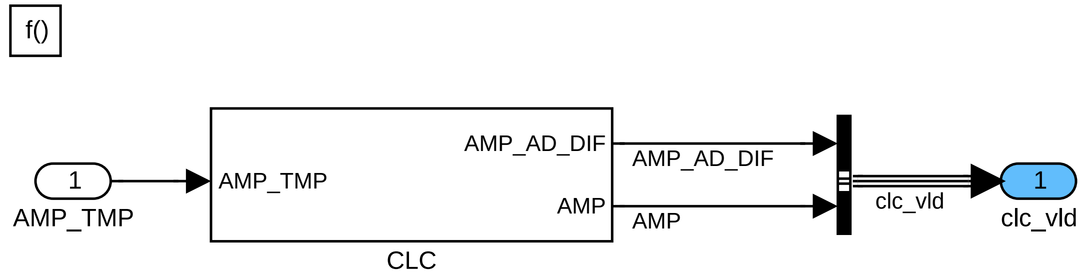


Figure 295 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_VLD

Learn AMP: Calculation of AMP via AMP_TMP (LV_ST = 1)

AMP = AMP_AD or AMP = AMP_SUB. This depends on the configuration: charged engine or not.

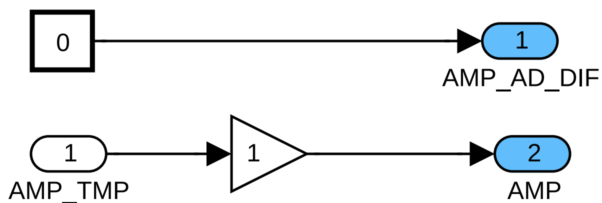


Figure 296 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ AMP_SENS_NOT_AVL/ FCT_CLC/ CLC_VLD/ CLC

Bus merge for chapter "AMP sensor available or not":

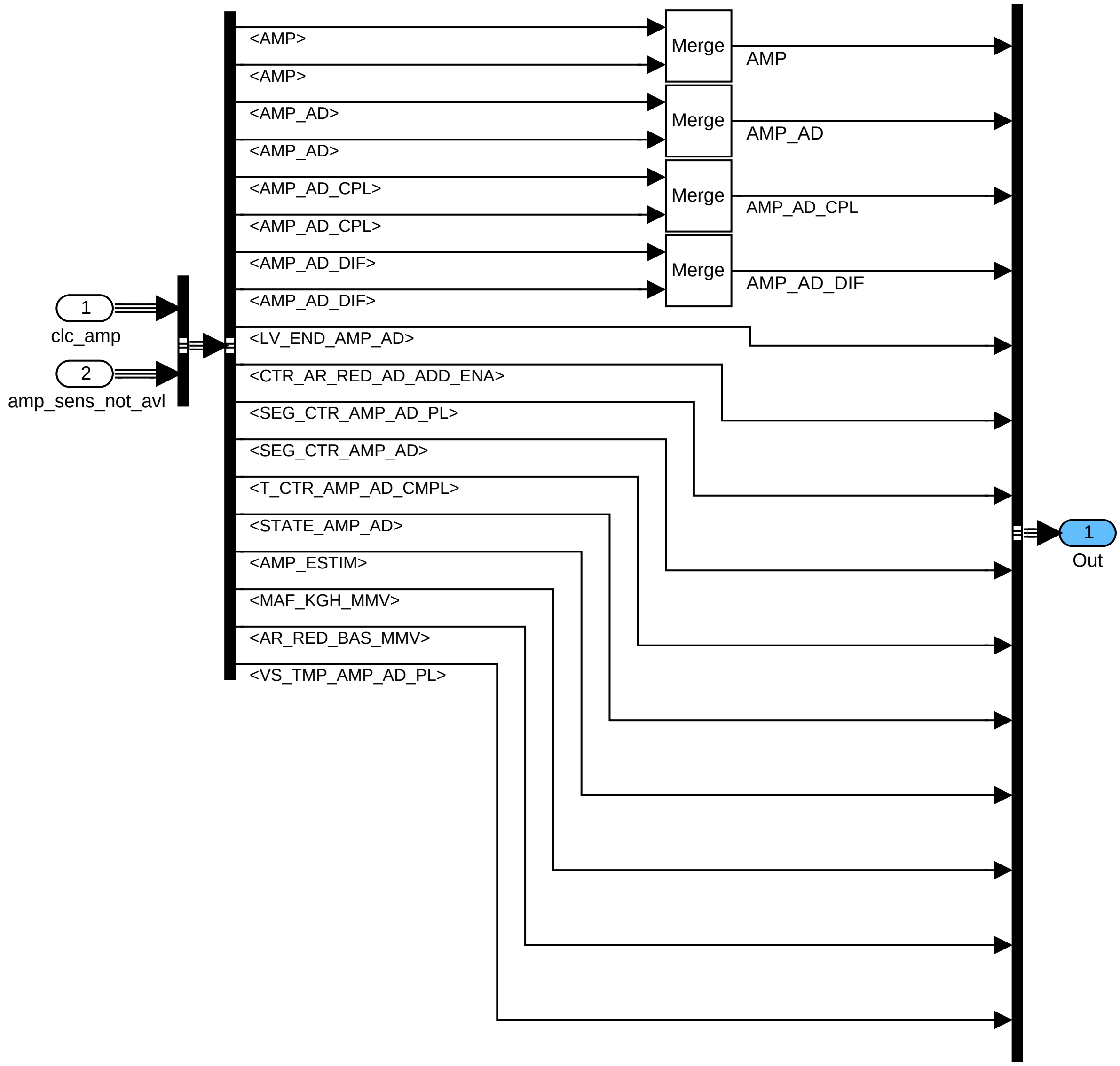


Figure 297 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ BusMerge

AMP sensor available:

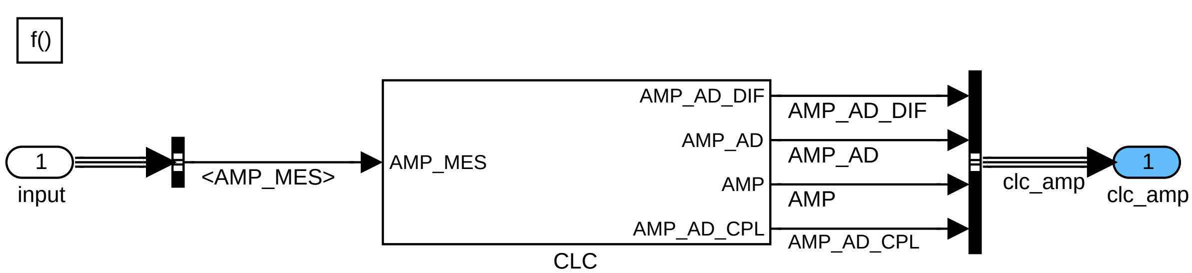


Figure 298 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ CLC_AMP

AMP sensor available: Calculation of AMP, AMP_AD via measured AMP:

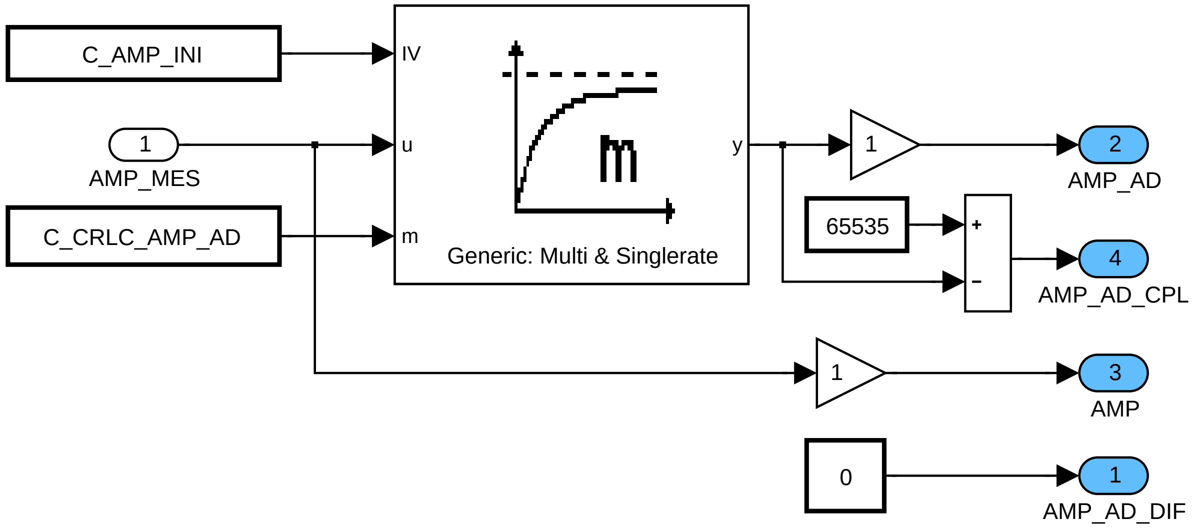


Figure 299 INSY_ADCPRAMP0/ OPM_SEG/ CLC/ CLC_AMP/ CLC

4.17 Pressure decrease through the air cleaner

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
AMP_DEC	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure decrease through the air cleaner					
PRS_AIC_DOWN	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure downstream the air cleaner					
PRS_AIC_DOWN_MAX	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Maximum pressure downstream the air cleaner					

Input Data:

MAF_THR	AMP	MAF_CYL	VS
PQ	PQ_SP		

Calibration Data:

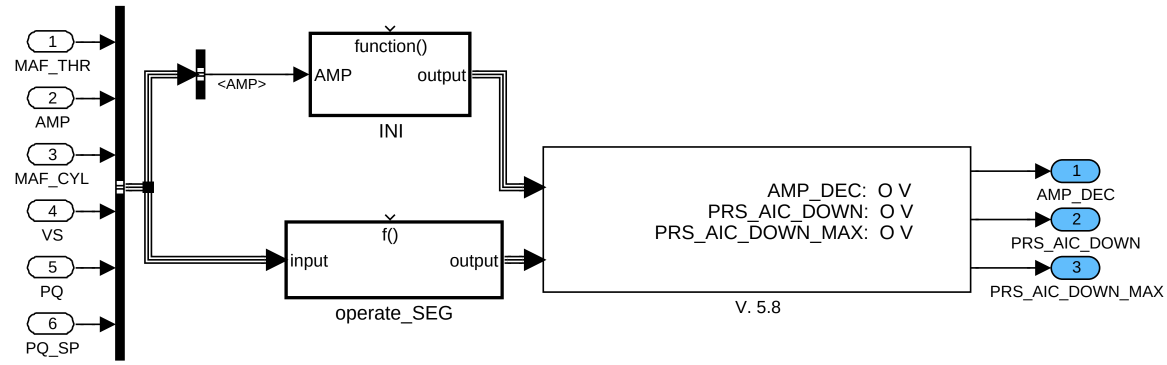
Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_MAF_DIF_MAX_PRS_AIC_DOWN_SWI	1	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Maximum threshold for stationary state determination depending of the difference between MAF_THR and MAF_CYL					
C_PQ_PRS_LOSS_AIC_SWI_HYS	1	0... FFFFH	0... 0.9999847	15.259e-6	[-]
Pressure quotient hysteresis for switching back to default input value of IP_PRS_LOSS_AIC					
C_PQ_PRS_LOSS_AIC_SWI_THD	1	0... FFFFH	0... 0.9999847	15.259e-6	[-]
Pressure quotient threshold for switching between two input values of IP_PRS_LOSS_AIC					
IP_PRS_LOSS_AIC	4x4	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_MAF_KGH_IP_PRS_LOSS_AIC	4	0... FFFFH	0... 2047.969	0.03125	[kg/h]
LDP_VS_IP_PRS_LOSS_AIC	4	0... FFH	0... 255	1	[km/h]
Pressure loss through the air cleaner					

General Information

General information:
With increasing air mass flow through the air cleaner there is a drop of pressure in the air cleaner in comparison to the ambient pressure.

The calculation is done depending on the activation condition synchronously to multiples of segment.

Function description



SDA_SRS / SDA V 5.0.2 27−Jan−2006

Figure 300:
Path: INSY_MDLADPRSDA0

4.17.1 SUBFUNCTION: INI

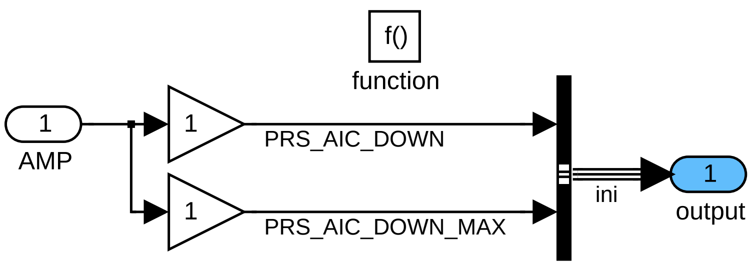


Figure 301:
Path: INSY_MDLADPRSDA0/INI

4.17.2 Calculation of segment task

4.17.2.1 Calculation of PRS_AIC_DOWN, PRS_AIC_DOWN_MAX and AMP_DEC

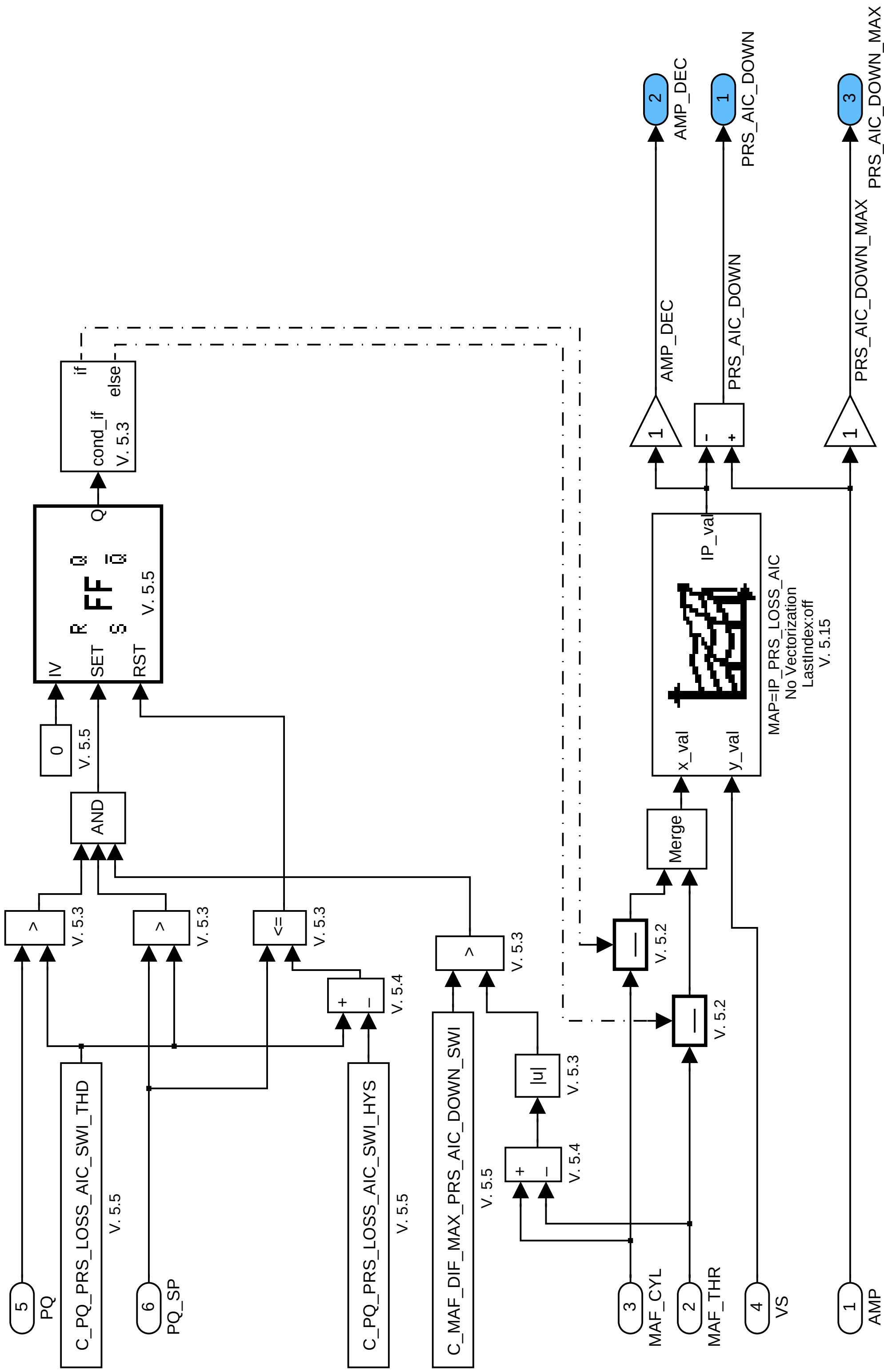


Figure 302:
Path: INSY_MDLADPRSDA0/operate_SEG/CLC_PRS_AIC

4.18 Basic Volumetric Efficiency for IVVT at Inlet and Outlet

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_OFS_BAS	O/V	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Offset of the basic volumetric efficiency					
EFF_VOL_OFS_BAS_REQ_CAM_OFS_AD[3]	O/V	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Offset of the basic volumetric efficiency - valid only for CAM_OFS_AD functionality					
EFF_VOL_SLOP_BAS	O/V	0...FFFFH	0...12.05999	0.184e-3	[kg/(h*hPa)]
Slope of the basic volumetric efficiency					
EFF_VOL_SLOP_BAS_REQ_CAM_OFS_AD[3]	O/V	0...FFFFH	0...12.05999	0.184e-3	[kg/(h*hPa)]
Slope of the basic volumetric efficiency - valid only for CAM_OFS_AD functionality					
IDX_CAM_EX	V	0...FFH	0...255	1	[-]
Detected index of CAM_EX					
CAM_MV_IN_1	V	0...FFH	60...155.625	0.375	[°CRK]
Inlet camshaft position which is used as input for the volumetric efficiency calculation					
CAM_MV_EX_1	V	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Exhaust camshaft position which is used as input for the volumetric efficiency calculation					

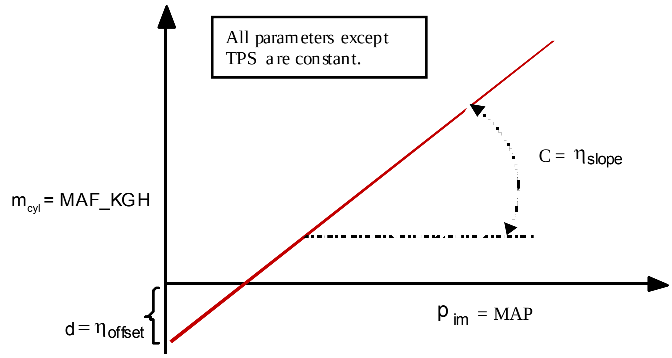
Input data:

NLC_IVVT_IN	CAM_MV_IN	LV_EFF_VOL_CLC_CAM_OFS_AD	STATE_CLC_RED
C_CAM_INI_EX	N_REQ_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	NC_NR_N_EFF_VOL_BAS
NLC_IVVT_EX	NC_NR_LEN_CAM_EX	STATE_ACR_EFF_VOL_CLC	NC_NR_LEN_CAM_IN
C_CAM_INI_IN	NC_NR_STATE_ACR	NC_STATE_CLC_RED_CLC_DEAC	CAM_MV_EX
N	STATE_CLC_RED	LV_CAM_SP_ADJ_CAM_OFS_AD	CAM_OFS_IVVT_IN_1
		CAM_MV_IN_REQ_CAM_OFS_AD[3]	CAM_OFS_IVVT_EX_1
		CAM_MV_EX_REQ_CAM_OFS_AD[3]	

FUNCTION DESCRIPTION:

General information:

At a given engine speed N, the air-mass flow is modelled as a linear function (volumetric efficiency) depending on the manifold pressure MAP. The influence of the engine specific actuators is modelled as variations of the slope and offset. This approach ensures a basically linear model.



With the index STATE_ACR_EFF_VOL_CLC (built in the Appl.Inc.), it can be switched between different sets of volumetric efficiency maps, depending on the project-configuration (see INSY Configuration data).
Example: influence of actuators VIM, PORT, VVL etc., or others like SECONDARY-AIR.

No interpolation between these different IP-set [x] and IP-set [y] will be done.

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_EFF_VOL_OFS_2[NC_NR_STATE_ACR][NC_NR_LEN_CAM_EX]	NC_NR_N_EFF_VOL_BAS*NC_NR_LEN_CAM_IN	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDPM_N_3_INSY	NC_NR_N_EFF_VOL_BAS	0...1FE0H	0...8160	1	[rpm]
LDPM_CAM_MV_IN_1_INSY	NC_NR_LEN_CAM_IN	0...FFH	60...155.625	0.375	[°CRK]
Offset of the cylinder air-mass flow					
IP_EFF_VOL_SLOP_2[NC_NR_STATE_ACR][NC_NR_LEN_CAM_EX]	NC_NR_N_EFF_VOL_BAS*NC_NR_LEN_CAM_IN	0...FFFFH	0...12.05999	0.184e-3	[kg/(h*hPa)]
LDPM_N_3_INSY	NC_NR_N_EFF_VOL_BAS	0...1FE0H	0...8160	1	[rpm]
LDPM_CAM_MV_IN_1_INSY	NC_NR_LEN_CAM_IN	0...FFH	60...155.625	0.375	[°CRK]
Slope of the cylinder air-mass flow					

Hint:

IP_EFF_VOL_RAM	NC_NR_LEN_CAM_EX	0...FFFFH	-	-	-
LDPM_CAM_MV_EX_INSY	NC_NR_LEN_CAM_EX	0...FFH	-40.125...-135.75	0.375	°CRK
intermediate array (defined in the same way like a map), contains temporary offset/slope of the cylinder air-mass flow; IP_EFF_VOL_RAM has to be placed in RAM; not visible in application-system!! LDPM_CAM_MV_EX_INSY can be calibrated normally					

4.18.1 Standard EFF_VOL calculation (Seg-Syn.), using camshaft position mean value

Recurrence: multiples of a segment

Activation: **if** (STATE_CLC_RED **and** NC_STATE_CLC_RED_CLC_DEAC) = 0
(masking = bitwise AND!)

Formula:

#if (NLC_IVVT_IN = 1)
#then CAM_MV_IN_1 = $CAM\_MV\_IN_k$ + CAM_OFS_IVVT_IN_1
#else CAM_MV_IN_1 = C_CAM_INI_IN
#endif

#if (NLC_IVVT_EX = 1)
#then CAM_MV_EX_1 = $CAM\_MV\_EX_k$ + CAM_OFS_IVVT_EX_1
#else CAM_MV_EX_1 = C_CAM_INI_EX
#endif

subfunction-CALL: Calculate EFF_VOL_SLOP/OFS_BAS – Chapter 1.3
with input values: N_TMP = N
CAM_MV_IN_TMP = CAM_MV_IN_1
CAM_MV_EX_TMP = CAM_MV_EX_1

# Result: EFF_VOL_SLOP_BAS = EFF_VOL_SLOP_BAS_TMP
EFF_VOL_OFS_BAS = EFF_VOL_OFS_BAS_TMP

4.18.2 EFF_VOL-REQ-calculation for CAM_OFS-adaptation (20ms), when requested

Recurrence: *every 20 ms*

Activation: LV_EFF_VOL_CLC_CAM_OFS_AD = 1

Formula:

for i = 0 **to** 2 **FOR-loop below to be completed within one recurrence**

#if (NLC_IVVT_IN = 1)
#then CAM_MV_IN_2 = CAM_MV_IN_REQ_CAM_OFS_AD[i]
#else CAM_MV_IN_2 = 0hex
#endif

#if (NLC_IVVT_EX = 1)
#then CAM_MV_EX_2 = CAM_MV_EX_REQ_CAM_OFS_AD[i]
#else CAM_MV_EX_2 = 0hex
#endif

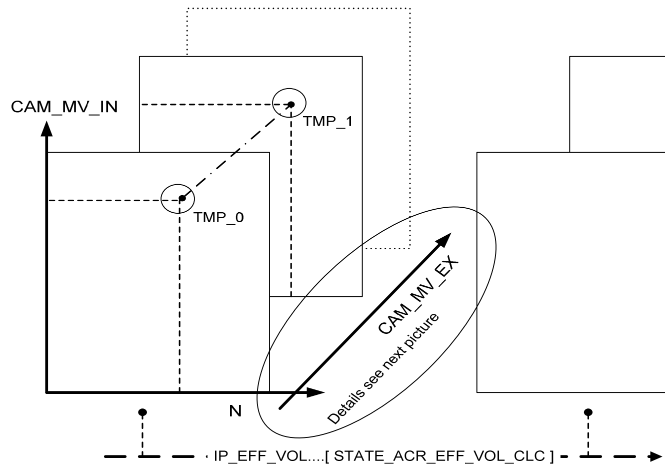
subfunction-CALL: Calculate EFF_VOL_SLOP/OFS_BAS – Chapter 1.3
with input values: N_TMP = N_REQ_CAM_OFS_AD
CAM_MV_IN_TMP = CAM_MV_IN_2
CAM_MV_EX_TMP = CAM_MV_EX_2

#Result: EFF_VOL_SLOP_BAS_REQ_CAM_OFS_AD[i] = EFF_VOL_SLOP_BAS_TMP
EFF_VOL_OFS_BAS_REQ_CAM_OFS_AD[i] = EFF_VOL_OFS_BAS_TMP

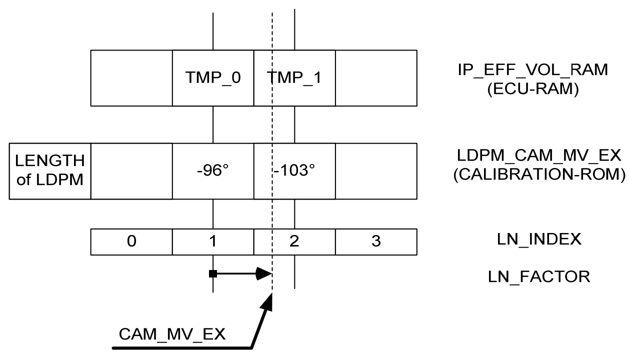
endfor

4.18.3 Main algorithm for the Basic Volumetric Efficiency - slope and offset

This subfunction is called twice with different input-parameters (see chapters above):
Param1 = N_TMP, Param2 = CAM_MV_IN_TMP, Param3 = CAM_MV_EX_TMP



Picture 1: overview of interpolation with 3 dimensions, and configurable sets of these 3D-maps



Picture 2: interpolation of 3rd dimension, with the help of standard library-routines

Example: input: CAM_MV_EX = -101.5°
output: LN_INDEX = 1
LN_FACTOR = (-101.5°-(-96°)) / (-103°-(-96°)) = 0.7857

LN_INDEX array-position [0...FF] of LDPM-search (range: 0...NC_NR_LEN_CAM_EX-1)
LN_FACTOR physical distance [0..0.9999][0..FFFF] between 2 interpolation-fields
in case of direct hit of the online-value CAM_MV_EX with the calibrated LDP-value, the internal variable LN_FACTOR is set to 0.

Note: The Hex. limits and Phys. limits of LDPM_CAM_MV_EX run against each other -> calibration of LDPM_CAM_MV_EX has to be done in ascending order for the Hex. limits.

Hint for better readability
"STATE_ACR" is used in following text equivalent to "STATE_ACR_EFF_VOL_CLC"

Formula section:

1. index-search for CAM_MV_EX-interpolation

* do LDPM-search of IP_EFF_VOL_RAM (CAM_MV_EX_TMP)

* save internal results to be used below:
  IDX_CAM_EX = LN_INDEX
  DELTA_CAM_EX_TMP = LN_FACTOR

2. Read out corresponding IP..., depending on STATE_ACR and CAM_MV_EX

TMP_0_OFS = IP_EFF_VOL_OFS_2 [STATE_ACR][IDX_CAM_EX]
(N_TMP, CAM_MV_IN_TMP)

TMP_0_SLOP = IP_EFF_VOL_SLOP_2 [STATE_ACR][IDX_CAM_EX]
(N_TMP, CAM_MV_IN_TMP)

if(1) *"no direct hit of LDPM" (DELTA_CAM_EX_TMP ≠ 0)*

then(1) *interpolation between 2 map-results:*

TMP_1_OFS = IP_EFF_VOL_OFS_2 [STATE_ACR][IDX_CAM_EX + 1]
(N_TMP, CAM_MV_IN_TMP)

TMP_1_SLOP = IP_EFF_VOL_SLOP_2 [STATE_ACR][IDX_CAM_EX + 1]
(N_TMP, CAM_MV_IN_TMP)

3. Store both interpolation-results into 2 fields of dummy-map (located in ECU-RAM)

offset:

IP_EFF_VOL_RAM [IDX_CAM_EX] = TMP_0_OFS

IP_EFF_VOL_RAM [IDX_CAM_EX + 1] = TMP_1_OFS

EFF_VOL_OFS_BAS_TMP = IP_EFF_VOL_RAM

Do interpolation with calculated index IDX_CAM_EX and DELTA_CAM_EX_TMP (from LDPM-search in(1.)) !

slope:

IP_EFF_VOL_RAM [IDX_CAM_EX] = TMP_0_SLOP

IP_EFF_VOL_RAM [IDX_CAM_EX + 1] = TMP_1_SLOP

EFF_VOL_SLOP_BAS_TMP = IP_EFF_VOL_RAM

Do interpolation with calculated index IDX_CAM_EX and DELTA_CAM_EX_TMP (from LDPM-search in(1.)) !

else(1) *"direct hit of LDP" : no additional interpolation necessary*

offset:

EFF_VOL_OFS_BAS_TMP = TMP_0_OFS

slope:

EFF_VOL_SLOP_BAS_TMP = TMP_0_SLOP

endif(1)

4.19 Calculation of VIM influence for volumetric efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_SLOP_VIM_COR	V/O	8000...7FFFH	-6.029...6.029	1.84e-4	kg/(hPa*h)
VIM influence for slope of volumetric efficiency					
EFF_VOL_OFS_VIM_COR	V/O	8000...7FFFH	-1024...1023.97	0.03125	kg/h
VIM influence for offset of volumetric efficiency					
FAC_VIM	V/O	0...FFH	0...0.996	0.0039	-
Correction factor of EFF_VOL_SLOP_VIM_COR and EFF_VOL_OFS_VIM_COR					
VIM_AV_FIL	V/O	0...FFH	0...0.996	0.0039	-
Filtered value of VIM_AV					
VIM_SP_FIL	V/O	0...FFH	0...0.996	0.0039	-
Filtered value of VIM_SP					

Input data:

PQ	LV_VIM_SP	VIM_AV	CONF_VIM
LV_ERR_VIM_FB_EL		N	VO

FUNCTION DESCRIPTION:

General information:

The influence of the switching intake manifold is taken into account via the correction coefficients EFF_VOL_SLOP_VIM_COR and EFF_VOL_OFS_VIM_COR.

In case of available VIM position feedback signal (CONF_VIM = 2), this measured signal VIM_AV is filtered and the correction of gas flow is based on this information VIM_AV_FIL.
If no feedback signal available (CONF_VIM = 1) or the diagnosis detects an error on the feedback signal (LV_ERR_VIM_FB_EL = 1), then the correction of gas flow is calculated via the filtered VIM setpoint VIM_SP_FIL.

The basic maps for the determination of the influence are depending on engine speed and valve overlap. The correction for slope/ offset of the volumetric efficiency is calculated with the filtered activation value of VIM.

Application conditions:

Initialisation: at reset

EFF_VOL_SLOP_VIM_COR = 0 kg(hPa * h) (0H)

EFF_VOL_OFS_VIM_COR = 0 kg/h (0H)

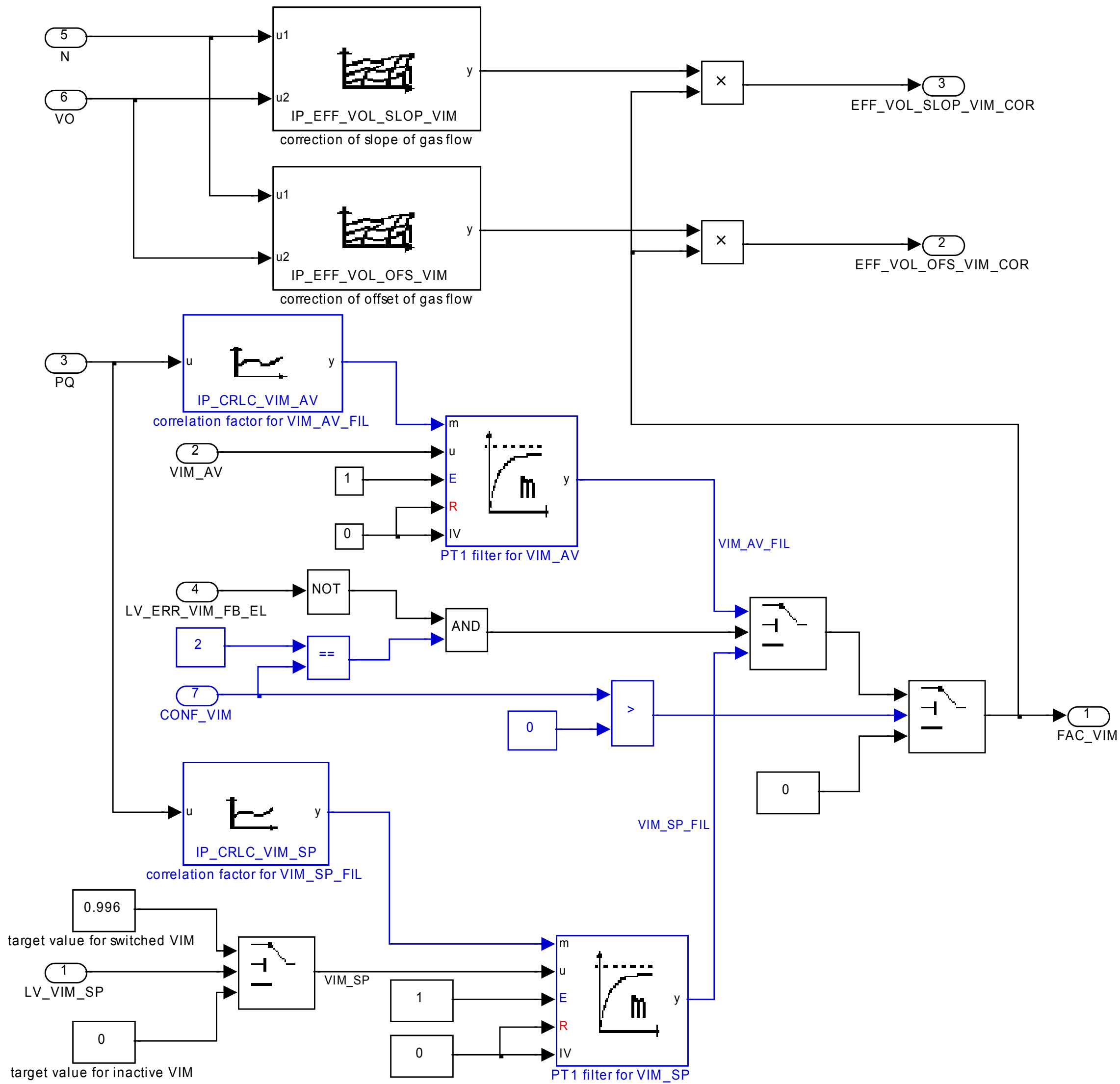
FAC_VIM = 0 (0H)

Recurrence: segment synchronous

Activation: at every engine state

Deactivation: none

Signal flow diagram:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_EFF_VOL_SLOP_VIM	12x12	0...FFFFH	-6.029...6.029	1.84e-4	kg/(hPa•h)
LDPM_N_1_INSY	12	0...1FE0H	0...8160	1	rpm
LDPM_VO_1_INSY	12	0...1C7H	0...170.625	0.375	°CRK
VIM influence for volumetric efficiency (slope of gas flow)					
IP_EFF_VOL_OFS_VIM	12x12	0...FFFFH	-1024...1023.969	0.03125	kg/h
LDPM_N_1_INSY	12	0...1FE0H	0...8160	1	rpm
LDPM_VO_1_INSY	12	0...1C7H	0...170.625	0.375	°CRK
VIM influence for volumetric efficiency (offset of gas flow)					
IP_CRLC_VIM_AV	1x4	0...FFH	0...0.996	0.0039	-
LPDM_PQ_FAC_VIM_AV	4	0...FFFFH	0...0.99998	1.525e-5	-
correlation constant for PT1 filter of VIM_AV					
IP_CRLC_VIM_SP	1x4	0...FFH	0...0.996	0.0039	-
LDPM_PQ_FAC_VIM_AV	4	0...FFFFH	0.99998	1.525e-5	-
correlation constant for PT1 filter of VIM_SP					

4.20 Calculation of PORT influence for volumetric efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_SLOP_PORT_COR	V/O	8000...7FFFH	-6.0301...6.0299	0.184e-3	kg/(hPa*h)
PORT influence for slope of volumetric efficiency					
EFF_VOL_OFS_PORT_COR	V/O	8000...7FFFH	-1024...1023.969	0.03125	kg/h
PORT influence for offset of volumetric efficiency					

Input data:

N	VO	FAC_PORT_DEAC	CONF_PORT

FUNCTION DESCRIPTION:

General information:

The influence of Port Flap for the volumetric efficiency is taken into account via the correction coefficients EFF_VOL_SLOPE_PORT_COR and EFF_VOL_OFS_PORT_COR.

The basic maps for the determination of the influence are depending on engine speed and valve overlap. The correction for slope/ offset of the volumetric efficiency is weighted with the actual value of the Port flap position.

Application conditions:

Initialisation: EFF_VOL_SLOP_PORT_COR = 0 kg(hPa * h) (0H)

EFF_VOL_OFS_PORT_COR = 0 kg/h (0H)

Recurrence: segment synchronous

Activation: CONF_PORT = 1

Deactivation:

Formula section:

EFF_VOL_SLOP_PORT_COR = FAC_PORT_DEAC * IP_EFF_VOL_SLOP_PORT

EFF_VOL_OFS_PORT_COR = FAC_PORT_DEAC * IP_EFF_VOL_OFS_PORT

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_EFF_VOL_SLOP_PORT	12*12	0...FFFFH	-6.0301...6.0299	0.184e-3	kg/(hPa*h)
LDPM_N_1_INSY	12	0...1FE0H	0...8160	1	[rpm]
LDPM_VO_1_INSY	12	0...1C7H	0...170.625	0.375	[°CRK]
Port influence for volumetric efficiency (slope of gas flow)					
IP_EFF_VOL_OFS_PORT	12*12	0...FFFFH	-1024...1023.969	0.03125	[kg/h]
LDPM_N_1_INSY	12	0...1FE0H	0...8160	1	[rpm]
LDPM_VO_1_INSY	12	0...1C7H	0...170.625	0.375	[°CRK]
Port influence for volumetric efficiency (offset of gas flow)					

4.21 Volumetric efficiency correction coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_OFS_SUM	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
Final offset of volumetric efficiency					
EFF_VOL_OFS_SUM_MAX_COR	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
Maximum possible offset of volumetric efficiency (temperature and altitude corrected)					
EFF_VOL_SLOP	O/V	0...FFFFH	0...12.0599926	1.84024E-4	kg/(h*hPa)
Final slope of volumetric efficiency					
EFF_VOL_SLOP_MAX_COR	O/V	0...FFFFH	0...12.0599926	1.84024E-4	kg/(h*hPa)
Maximum possible slope of volumetric efficiency (temperature and altitude corrected)					

Input data:

EFF_VOL_FLOW_COR	EFF_VOL_OFS_AMP_COR	EFF_VOL_OFS_BAS	EFF_VOL_SLOP_AMP_COR
EFF_VOL_SLOP_BAS	EFF_VOL_TEMP_COR_MMV	EFF_VOL_OFS_PORT_COR	EFF_VOL_OFS_VIM_COR
EFF_VOL_SLOP_PORT_COR	EFF_VOL_SLOP_VIM_COR	N	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_EFF_VOL_OFS_SUM_MAX	12	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
LDPM_N_1_INSY	12	0...1FE0H	0...8.16E+3	1	rpm
Offset of gas flow into cylinder at WOT and standard conditions (incl. all actuator influences, e.g. IVVT, VIM)					
IP_EFF_VOL_SLOP_MAX	12	0...FFFFH	0...12.0599926	1.84024E-4	kg/(h*hPa)
LDPM_N_1_INSY	12	0...1FE0H	0...8.16E+3	1	rpm
Slope of gas flow into cylinder at WOT and standard conditions (incl. all actuator influences, e.g. IVVT, VIM)					

4.21.1 General information:

Volumetric efficiency adaption

In systems with both MAF and MAP sensor one of measured values MAF_KGH and MAP_MES delivered by these load sensors is always used in the INSY controller to tune the model of the throttle flow. The present module uses the other load sensor to adapt the volumetric efficiency of the engine.

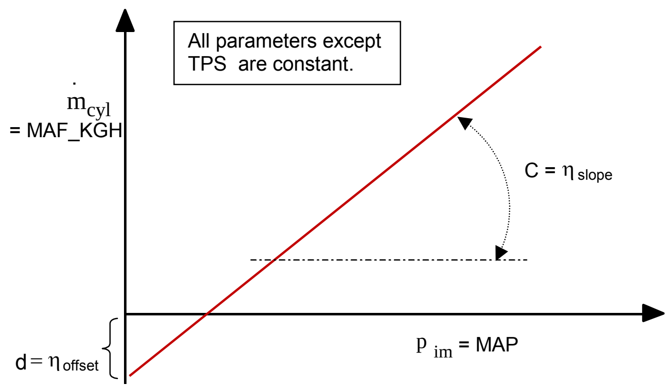
At a given engine speed N the air-mass flow is modeled as a linear function (volumetric efficiency) depending on the manifold pressure MAP.

Variations of the slope and offset model the influence of the engine specific actuators. The total volumetric efficiency can be build out of a basic value and a lot of engine specific corrections (e.g. VIM or port flap influences). Corrective values of these slope and offset EFF_VOL_SLOP_DIF and EFF_VOL_OFS_DIF are calculated here. EFF_VOL_OFS_DIF is calculated at low MAP, EFF_VOL_SLOP_DIF at high MAP.

Calculation of final volumetric efficency

general specification

At a given engine speed N the air-mass flow is modeled as a linear function (volumetric efficiency) depending on the manifold pressure MAP.



Variations of the slope and offset model the influence of the engine specific actuators. This approach ensures a basically linear model.

The total volumetric efficiency can be build out of a basic value and a lot of engine specific corrections (e.g. VIM or port flap influences). Both the basic efficiency and all the engine specific influences are calculated in the corresponding external modules. This module only coordinates all the corrections and calculates the final volumetric efficiency. The complete structure is depicted below.

Function Description

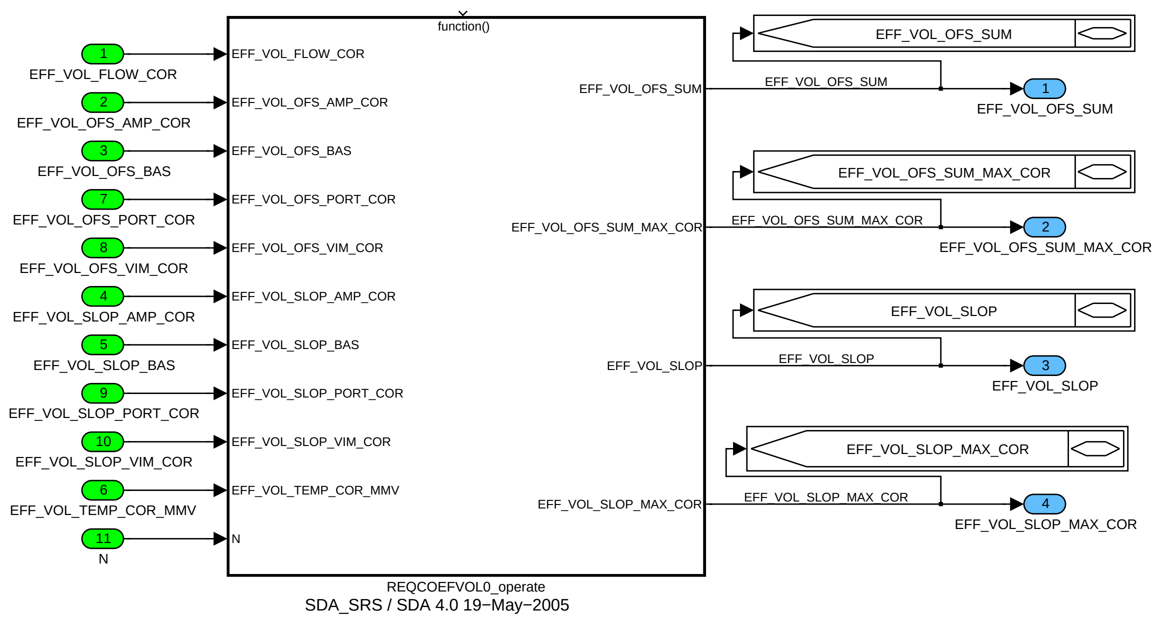


Figure 303 INSY_M403R

4.21.1.1 Calculation of final volumetric efficiency

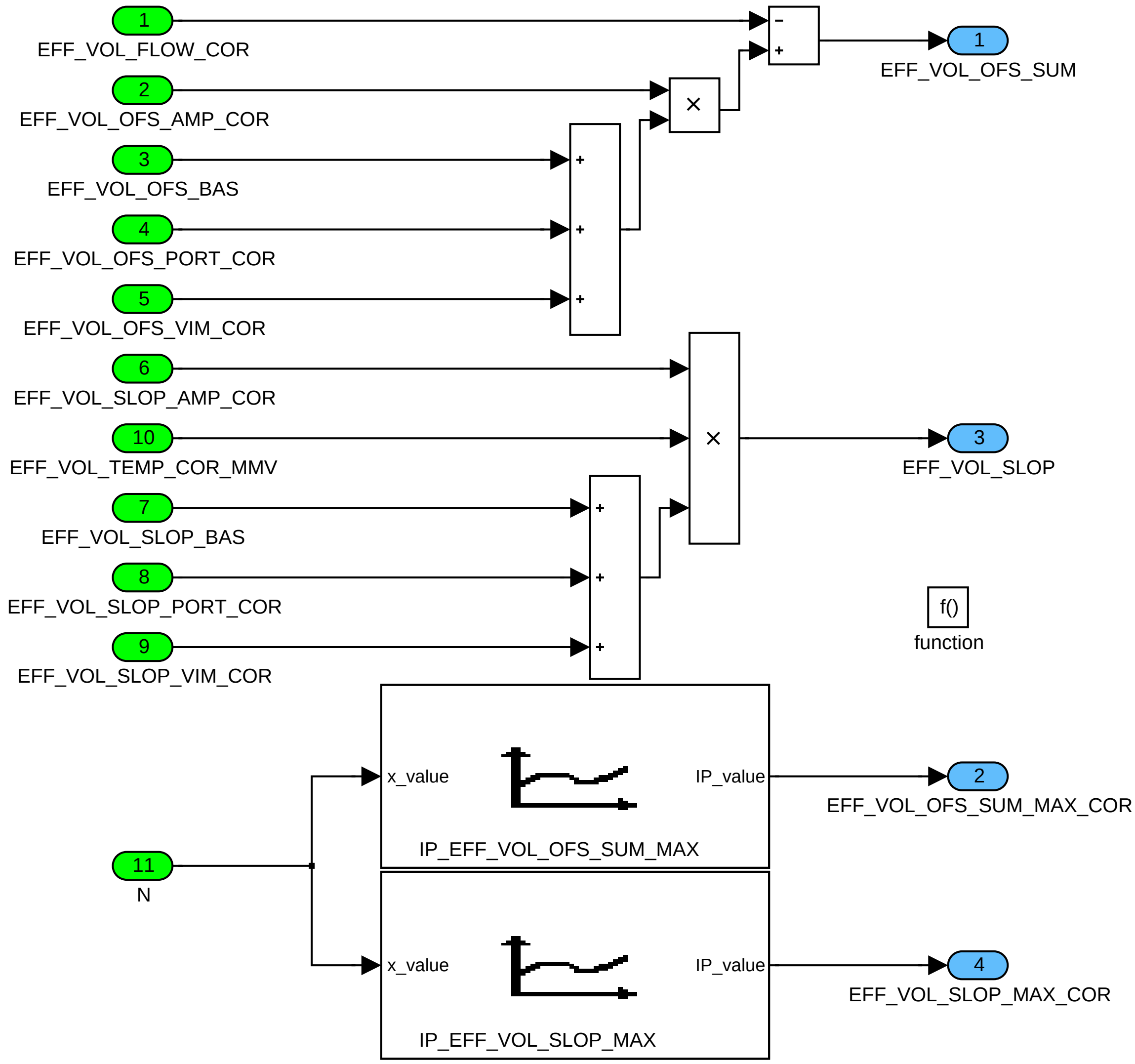


Figure 304 INSY_M403R/ REQCOEFVOL0_operate

4.22 Altitude Correction of Volumetric Efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_OFS_AMP_COR	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Altitude correction of basic volumetric efficiency offset					
EFF_VOL_SLOP_AMP_COR	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Altitude correction of basic volumetric efficiency slope					

Input data:

AMP

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_EFF_VOL_OFS_AMP	6	0...FFFFH	0...1.99996948	3.05176E-5	-
LDPM_AMP_1_INSY	6	0...FFFFH	0...5.434E+3	0.08292	hPa
Ambient pressure correction of the volumetric efficiency offset					
IP_EFF_VOL_SLOP_AMP	6	0...FFFFH	0...1.99996948	3.05176E-5	-
LDPM_AMP_1_INSY	6	0...FFFFH	0...5.434E+3	0.08292	hPa
Ambient pressure correction of the volumetric efficiency slope					

4.22.1 General Information:

The effect of changig the exhaust pressure caused by variation in ambient pressure (.e.g. due to changes in the altitude) is taken into account on both, offset and slope of the volumetric efficiency. The correction is a factor, that the basic slope (and correspondingly the offset) has to be multiplied with.

Function Description

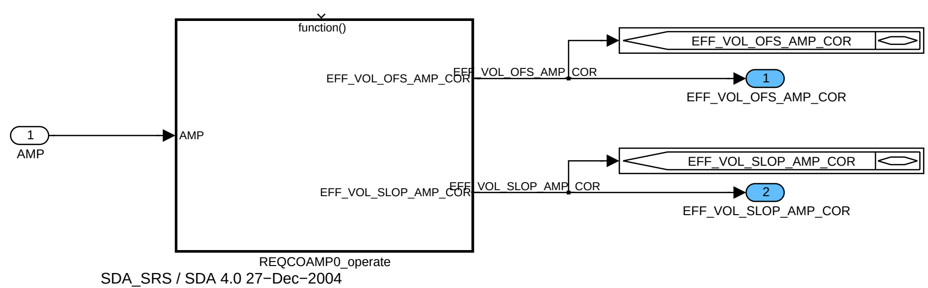


Figure 305 INSY_REQCOAMP0

4.22.1.1 SUBFUNCTION: REQCOAMP0_operate

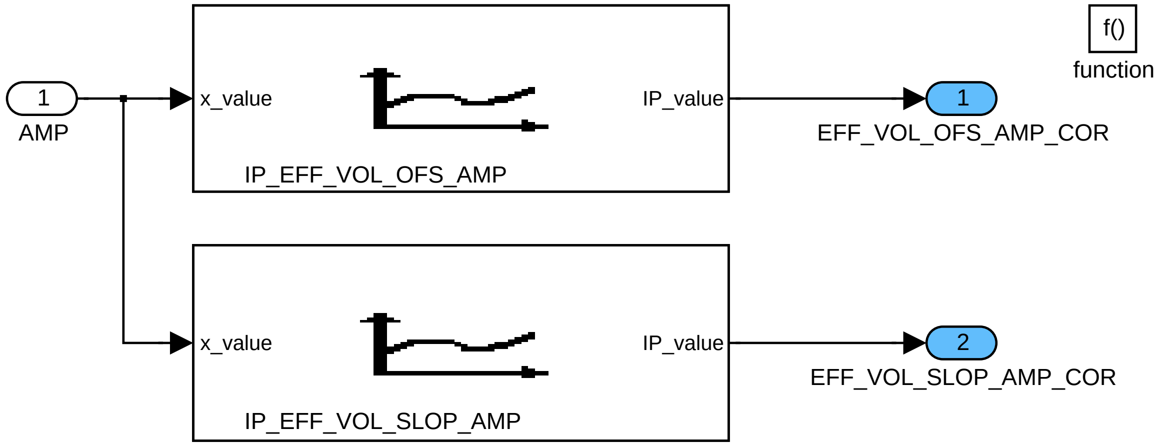


Figure 306 INSY_REQCOAMP0/ REQCOAMP0_operate

4.23 Temperature correction of Volumetric Efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_TEMP_COR_MMV	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Filtered volumetric efficiency correction factor due to temperature effects					
EFF_VOL_TEMP_COR	V	0...FFFFH	0...1.99996948	3.05176E-5	-
Volumetric efficiency correction factor due to temperature effects					

Input data:

LV_ST_END	N_32	TCO	TIA_IM
	MAF	NC_NR_LEN_IP_TIA_IM_STND	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_STND	1	0...FEH	-48...142.5	0.75	°C
Standard coolant temperature					
C_TCYL_STND	1	0...FEH	-48...142.5	0.75	°C
Standard temperature in the cylinder					
IP_CRLC_EFF_VOL_TEMP_COR	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_N_32_2_INSY	8	0...FFH	0...8.16E+3	32	rpm
Filter constant for volumetric efficiency correction					
IP_EFF_TCO_FAC	8x8	0...FFH	0...0.99609375	0.00390625	-
LDPM_N_32_2_INSY	8	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_1_INSY	8	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Volumetric efficiency weighting factor versus cooling temperature					
IP_EFF_TIA_IM_FAC	8x8	0...FFH	0...0.99609375	0.00390625	-
LDPM_N_32_2_INSY	8	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_1_INSY	8	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Volumetric efficiency weighting factor versus intake manifold gas temperature					
IP_TIA_IM_STND	NC_NR_LEN_IP_TIA_IM_STND xNC_NR_LEN_IP_TIA_IM_STND	0...FEH	-48...142.5	0.75	°C
LDP_N_32_IP_TIA_IM_STND	NC_NR_LEN_IP_TIA_IM_STND	0...FFH	0...8.16E+3	32	rpm
LDP_MAF_IP_TIA_IM_STND	NC_NR_LEN_IP_TIA_IM_STND	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Standard intake air temperature					

4.23.1 General Information

Target of the temperature correction function is, to compensate the temperature influences onto the volumetric efficiency calculation. The air-mass flow into the cylinder is the product of the volume flow and the density of the air $\dot{m} = A \cdot w \cdot \rho(T) = \dot{V} \cdot \rho(T)$, where the density depends on the air-temperature. At low temperatures the air-density is much higher than at standard temperatures, therefore the air flowing into the cylinder at the same manifold pressure will increase. The basic volumetric efficiency usually is calibrated at standard temperatures of about 25 °C for intake air and about 90 °C for coolant temperature. To take into account the change in density with temperature, the volumetric efficiency has to be corrected.

Especially at speed-density systems, this temperature correction is essential, because the density of the air and therefore the air-mass flow strongly depend on the air temperature. This means all deviations from the temperatures at which the volumetric efficiency was calibrated will lead to a wrong air-mass flow value into the cylinder (e.g. wrong MAF value at cold starts without temperature corrections for a MAP system).

At systems equipped with an HFM sensor, the temperature correction is not as essential as for speed-density systems. Nevertheless, it is used to consider the heating of the air due to heat transfer from the warm intake duct to the air.The compensation function is capable of taking into account deviations of coolant temperature and intake air temperature separately

$$EFF\_VOL\_COR = \frac{273 + C\_TCYL\_STND}{273 + C\_TCYL\_STND + IP\_EFF\_TCO\_FAC \cdot (TCO - C\_TCO\_STND) + IP\_EFF\_TIA\_IM\_FAC \cdot (TIA\_IM - C\_TIA\_STND)}$$

Application Condition

Activation:	At every engine state
Deactivation:	-
Initialization:	*at reset and engine running to stopped engine*
	EFF_VOL_TEMP_COR=EFF_VOL_TEMP_COR_MMV=1
Calculation recurrence:	The calculation is done depending on the activation condition- synchronously to **multiples of a segment**

Application Condition

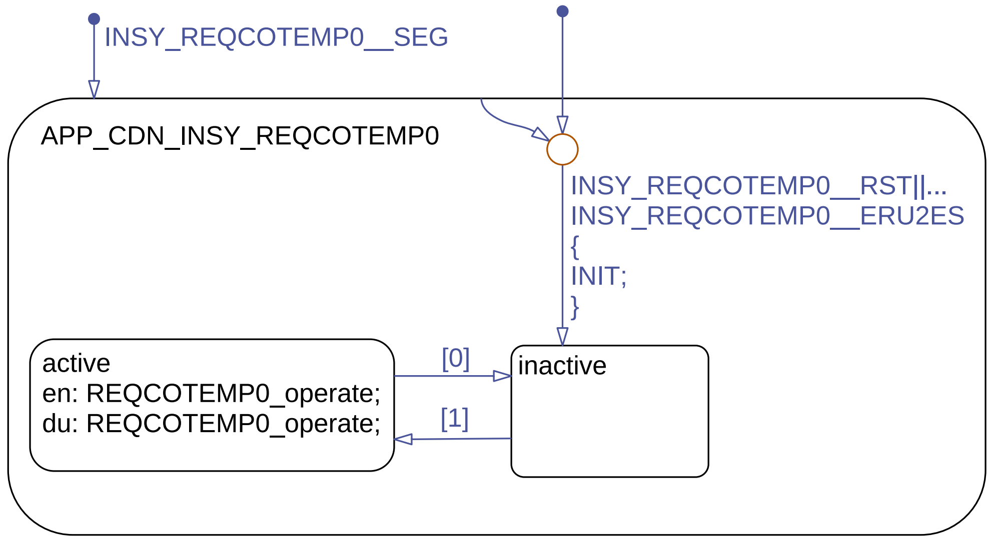


Figure 307 INSY_REQCOTEMP0/ APP_CDN/ APP_CDN_Chart

Function Description

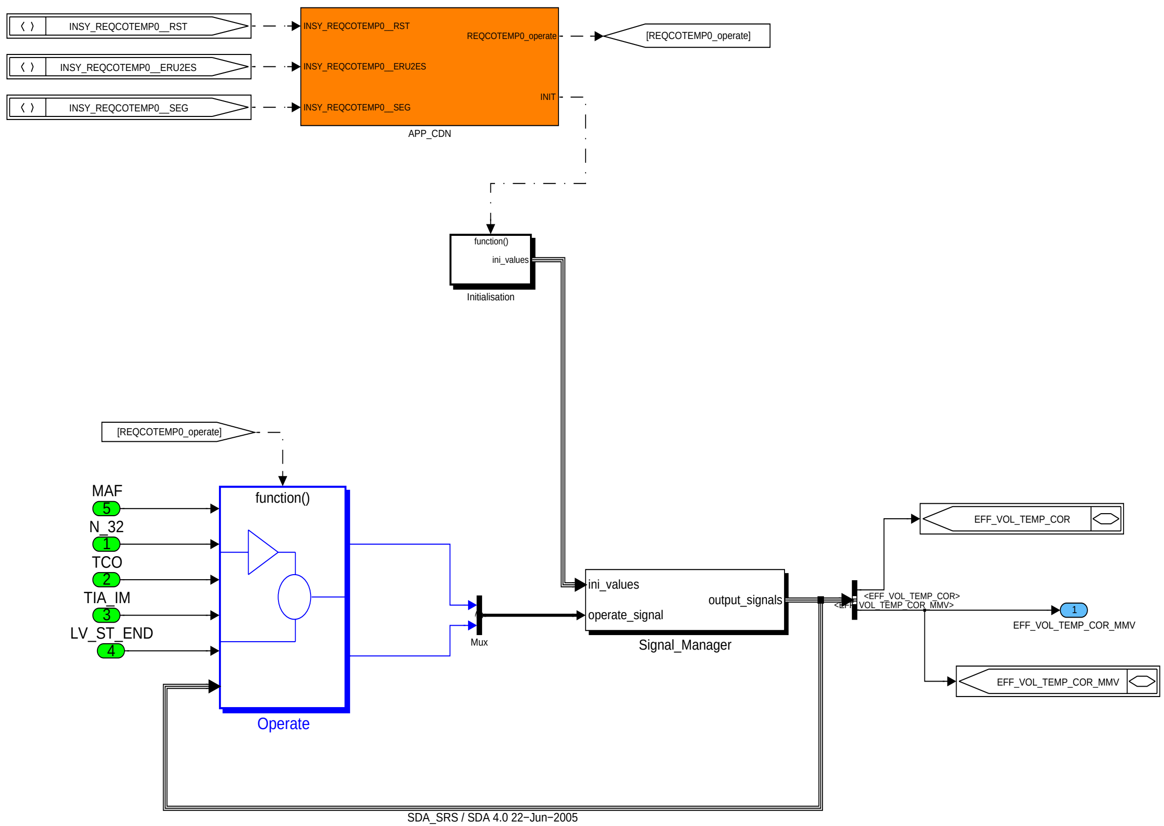


Figure 308 INSY_REQCOTEMP0

4.23.1.1 Initialization

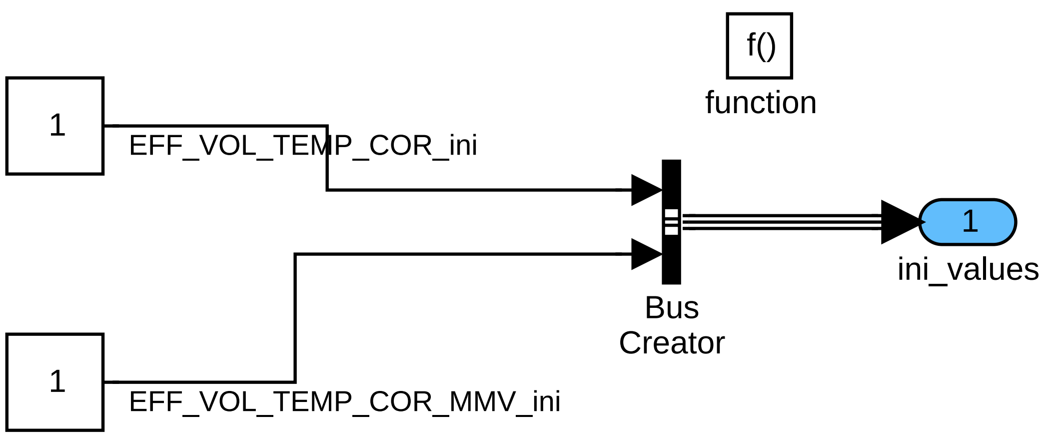


Figure 309 INSY_REQCOTEMP0/ Initialisation

Description:

The temperature correction factor EFF_VOL_TEMP_COR is the ratio between air temperature in the cylinder for standard conditions TCYL compared to air temperature in the cylinder for current TCO, TIA_IM and operating point. TCO and TIA_IM are compared to standard conditions and weighted by operating point.

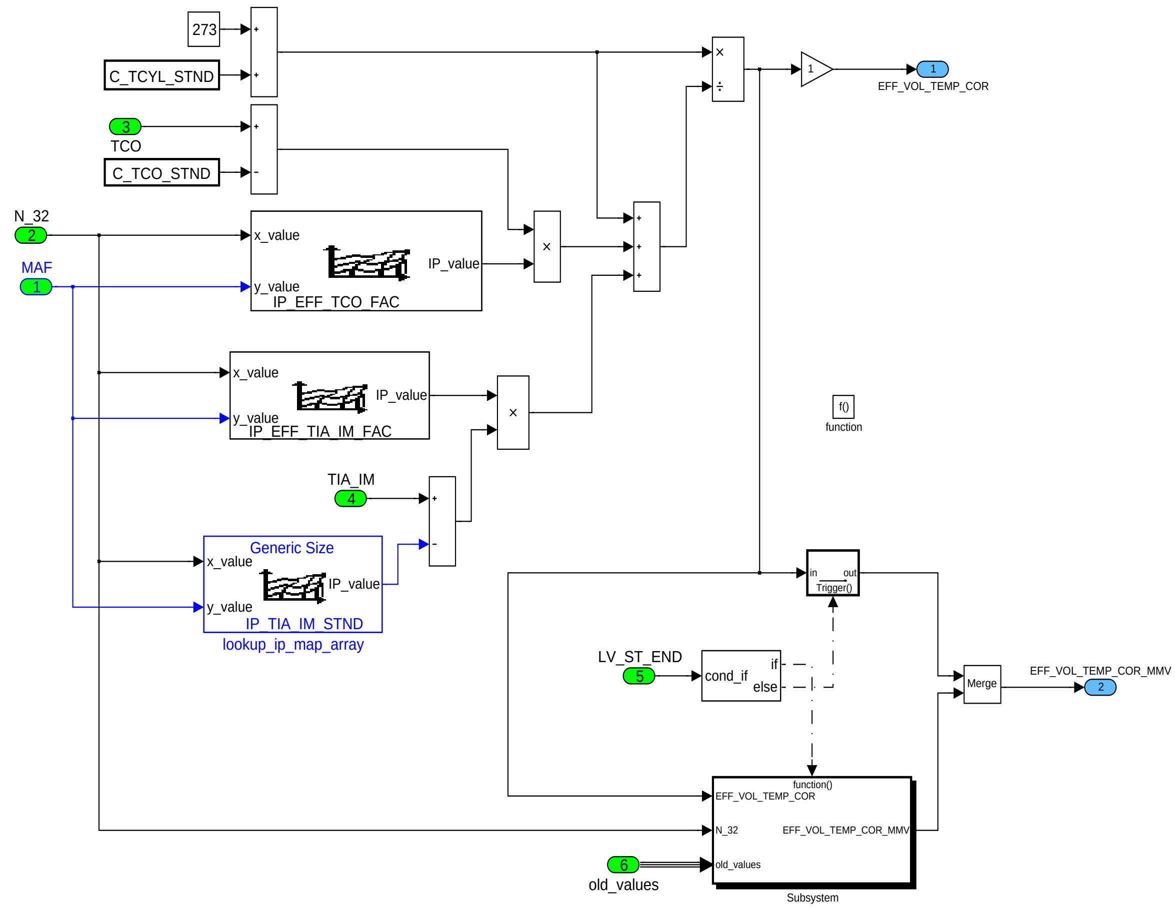


Figure 310 INSY_REQCOTEMP0/ Operate

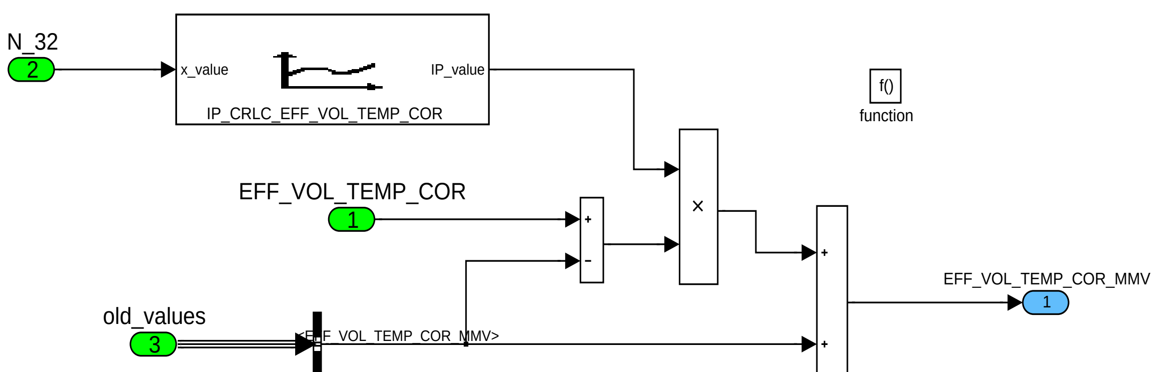


Figure 311 INSY_REQCOTEMP0/ Operate/ Subsystem

4.24 Flow Correction for Volumetric Efficiency Calculation

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
EFF_VOL_FLOW_COR	O/V	8000... 7FFFH	-1024... 1023.96875	0.03125	[kg/h]
Flow correction for volumetric efficiency					
EFF_VOL_PRS_EX_COR	V	8000... 7FFFH	-100... 99.9969482422	3.0518e-3	[kg/h]
Exhaust gas back pressure correction for volumetric efficiency					
EFF_VOL_VO_COR_1	V	8000... 7FFFH	-100... 99.9969482422	3.0518e-3	[-]
Valve overlap correction for volumetric efficiency with extended physical range					
PQ_MAP_PRS_EX	V	0... FFFFH	0... 1.99996948242	30.518e-6	[-]
Pressure quotient between intake manifold and exhaust pipe					

Input Data:

MAP_ESTIM	MAF_KGH	N	PRS_EX
VO	NC_NR_N_EFF_VOL_BAS	STATE_ACR_EFF_VOL_CLC	NC_NR_STATE_ACR

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
IP_EFF_VOL_PRS_EX_COR_1 [NC_NR_STATE_ACR]	NC_NR_N_EFF_VOL_BAS*12	0... FFFFH	-100... 99.9969482422	3.0518e-3	[kg/h]
LDPM_N_3_INSY	NC_NR_N_EFF_VOL_BAS	0... 1FE0H	0... 8160	1	[rpm]
LDPM_PQ_MAP_PRS_EX_1_INSY	12	0... FFFFH	0... 1.99996948242	30.518e-6	[-]
Exhaust gas back pressure correction for volumetric efficiency with NC_NR_STATE_ACR dependency					
IP_EFF_VOL_VO_COR_1 [NC_NR_STATE_ACR]	NC_NR_N_EFF_VOL_BAS*12	0... FFFFH	-100... 99.9969482422	3.0518e-3	[-]
LDPM_N_3_INSY	NC_NR_N_EFF_VOL_BAS	0... 1FE0H	0... 8160	1	[rpm]
LDPM_VO_1_INSY	12	0... 1C7H	0... 170.625	0.375	[°CRK]
Valve overlap correction for volumetric efficiency with NC_NR_STATE_ACR dependency					
IP_PRS_EX	12	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_MAF_KGH_IP_PRS_EX	12	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Exhaust gas back pressure (for tests)					
LC_PRS_EX_SWI	1	0... 4H	0... 4	1	[-]
Switch for PRS_EX calculation (0: calculation with model, 1: calculation with map, 3 and 4 not supported)					

General Information

general specification

Target of this correction function is, to consider the influence of cam phasing and exhaust gas back pressure on volumetric efficiency.

While the load exchange process with opening of the intake valves a back flow of burnt load from the combustion chamber to the intake manifold occurs. Beside engine speed, the amount of this back flow depends on the ratio between the exhaust gas back pressure and the intake manifold pressure and on the valve opening timing. This leads to a reduction of the volumetric efficiency.

Description:
The mass flow from the combustion chamber to the intake manifold or the exhaust pipe depends at any given engine speed on the one hand on the ratio between exhaust gas back pressure and intake-manifold pressure and on the other hand on the valve timing.

The number of engine speed breakpoints of the maps IP_EFF_VOL_PRS_EX_COR_1 and IP_EFF_VOL_VO_COR_1 are defined by NC_NR_N_EFF_VOL_BAS (defined in INSY configuration data). The same breakpoints are used for the basic volumetric efficiency maps (IP_EFF_VOL_OFS/SLOP).

Hint for calibration data: Dimension N = NC_NR_N_EFF_VOL_BAS for LDPM_N_3_INSY

With the index STATE_ACR_EFF_VOL_CLC (built in the Appl. Inc.) it can be switched between different maps. Between how many maps it can be switched depends on the number of actuators (project configuration) which are influencing EFF_VOL (e.g. VIM, PORT, etc.) (like in the basic volumetric efficiency).

Application Conditions:

Activation:	always
Deactivation:	-
Initialization:	At a reset of the ECU or at the transition from engine running to stopped engine the following values are initialized PQ_MAP_PRS_EX=1
Calculation recurrence:	The calculation is done –depending on the activation condition- synchronously to **multiples of a segment**

Function description

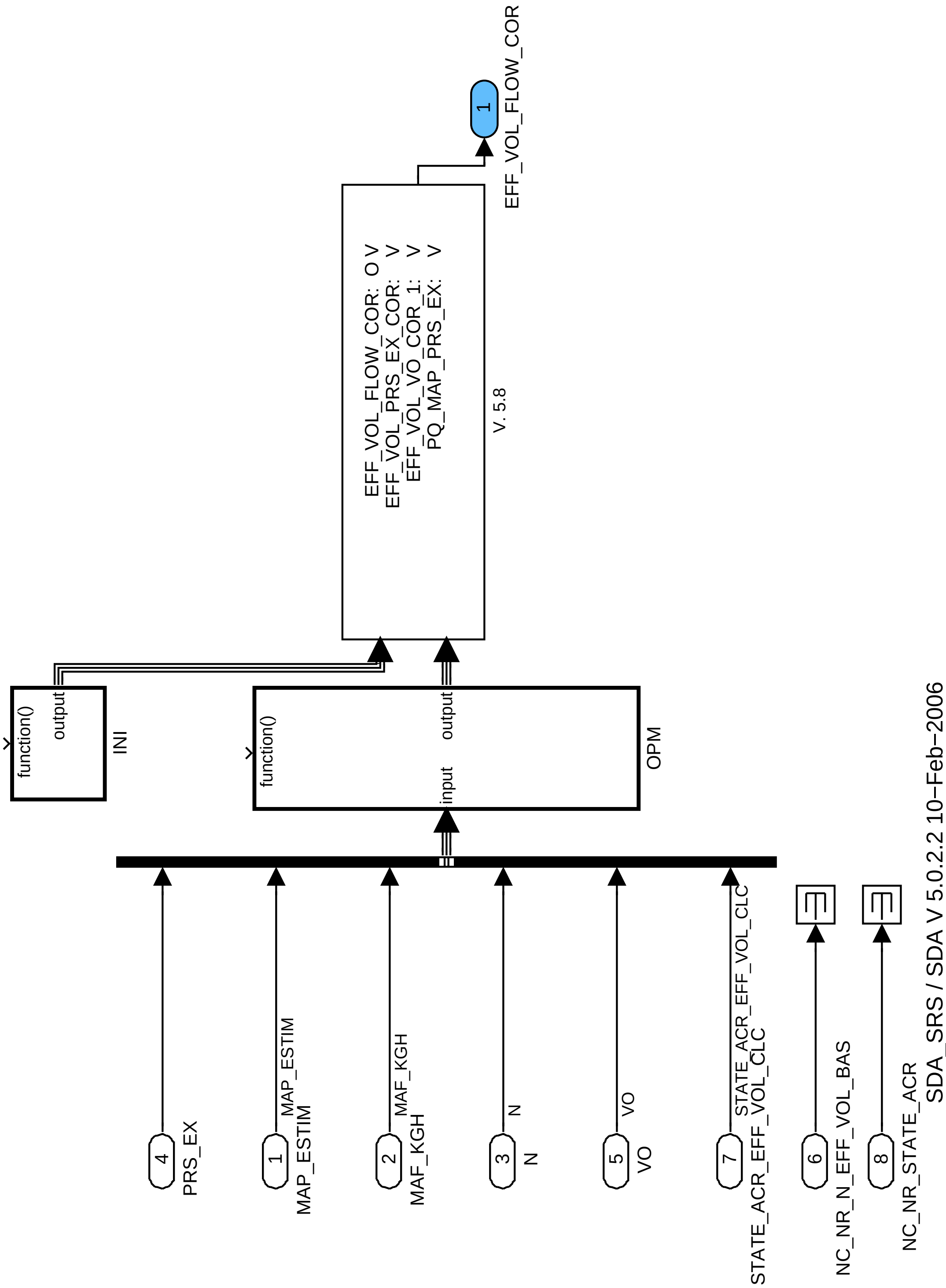


Figure 312:
Path: INSY_REQCOEFVOL1

4.24.1 SUBFUNCTION: INI

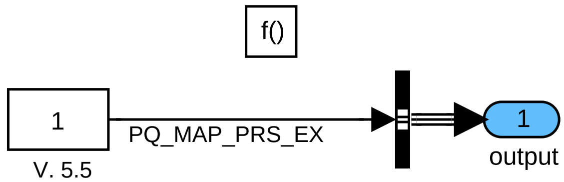


Figure 313:
Path: INSY_REQCOEFVOL1/INI

4.24.2 INSY_REQCOEFVOL1/OPM

4.24.2.1 INSY_REQCOEFVOL1/OPM/OPM_SEG

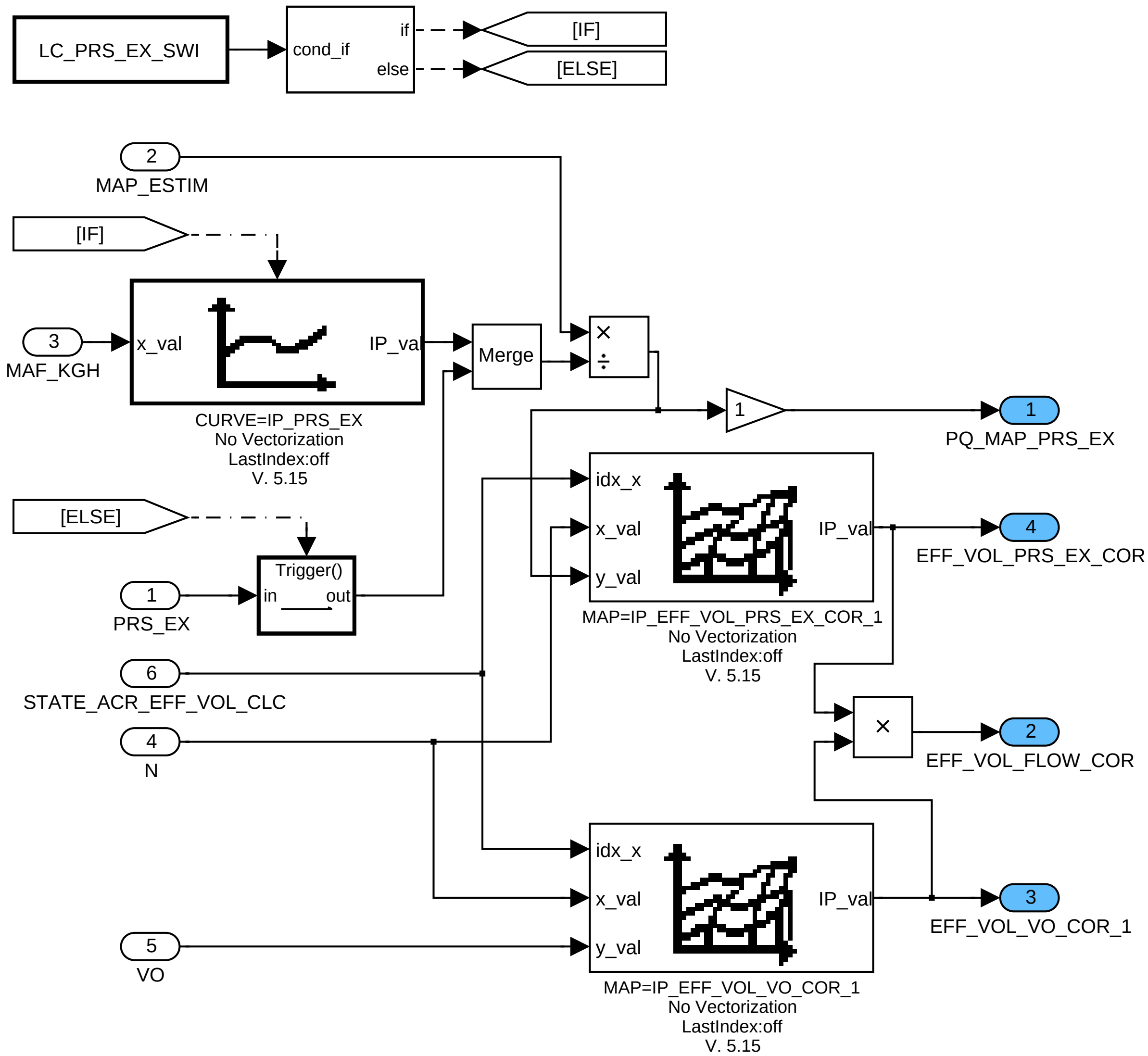
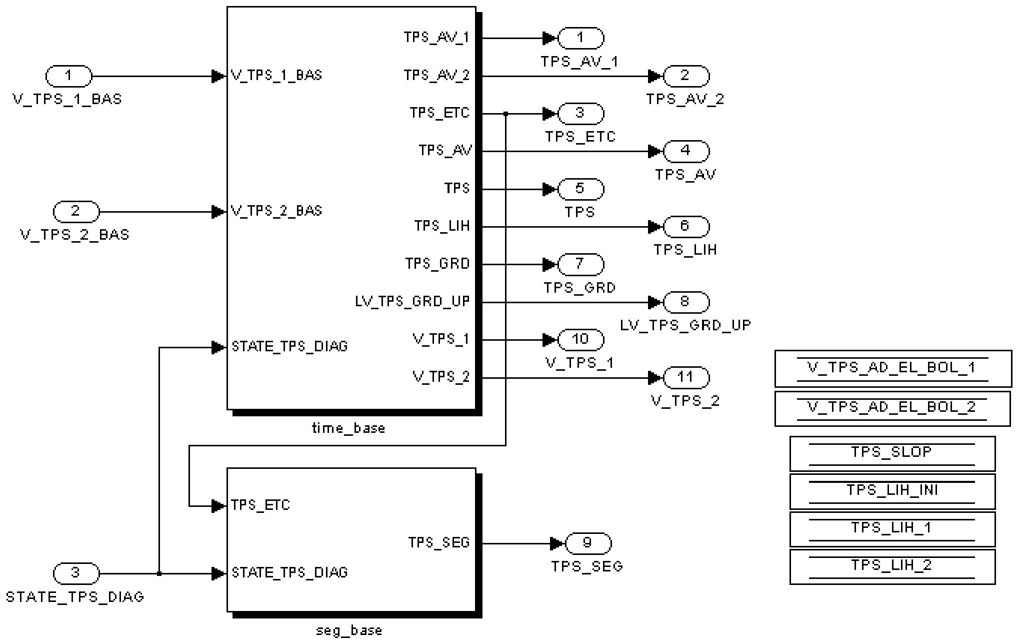


Figure 314:
Path: INSY_REQCOEFVOL1/OPM/OPM_SEG

4.25 Determination of the Throttle Position

General Information:

The throttle position is determined by a two-channel sensor system. Both of the supply voltage channels delivers inverse dispersing voltage signals. With the help of the determined adaptation values, the TPS-Position is standardized using the standard slope of the throttle position sensors.
V_TPS_1 has to be increasing and V_TPS_2 has to be decreasing with rising throttle.



A Plausibility Unit monitors both sensor signals, as well as the corresponding supply voltage and from them establishes the system state of the TPS-Position Acquisition (Undisturbed, Disturbed, TPS-Position not recognizable). From the voltage value of both TPS-Channels, the existing TPS-Position for the channels is determined, taking into consideration, the adaptation value for the lower stop. Based on the determined state of the system, the resulting TPS-Position is determined by the individual values. In case of a disturbance of the position acquisition, the selection of the position is conducted from one of the two DK-Channels- respectively- by a complete collapse, or the Shutdown of the TPS_control and with it the Limp Home.

Output data:

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
TPS_ETC	V/O	C000h...3FFFh	-119.51...119.5	0.0073	°TPS
Opening angle of the throttle valve for electronic throttle control					
TPS_AV	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Opening angle of the throttle valve					
TPS	V/O	0...FFh	0...119.5	0.47	°TPS
Opening angle of the throttle valve, low resolution					
TPS_AV_1	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Opening angle of the throttle valve of TPS-Poti 1					
TPS_AV_2	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Opening angle of the throttle valve of TPS-Poti 2					
TPS_LIH	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Limp home position of the throttle valve					
TPS_GRD	V/O	0...FFh	0...2987.5	11.72	°TPS/s
Throttle position gradient, absolute value					
LV_TPS_GRD_UP	V/O	0...1h	0...1	1	-
Logical variable for throttle position gradient direction					
TPS_SEG	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Segment synchronous measured throttle valve position					
V_TPS_1	V/O	0...3FFh	0... 4.9951	0.0049	V
Throttle valve voltage TPS – channel 1					
V_TPS_2	V/O	0...3FFh	0...4.9951	0.0049	V
Throttle valve voltage TPS – channel 2					

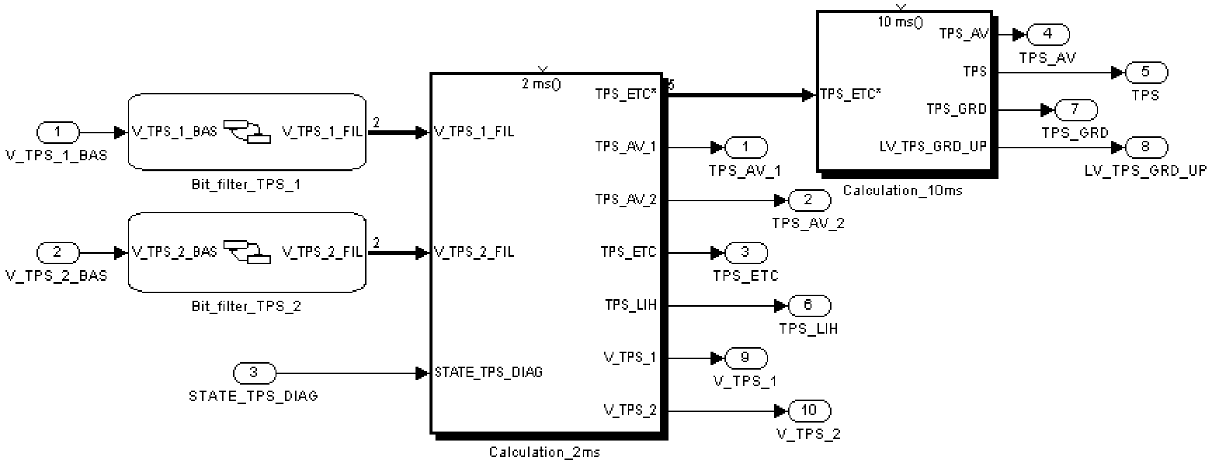
Input data:

V_TPS_1_BAS	V_TPS_2_BAS	V_TPS_AD_EL_BOL_1	V_TPS_AD_EL_BOL_2
TPS_SLOP	TPS_LIH_INI	TPS_LIH_1	TPS_LIH_2
STATE_TPS_DIAG			

4.25.1 Time synchronous throttle valve position

Description:

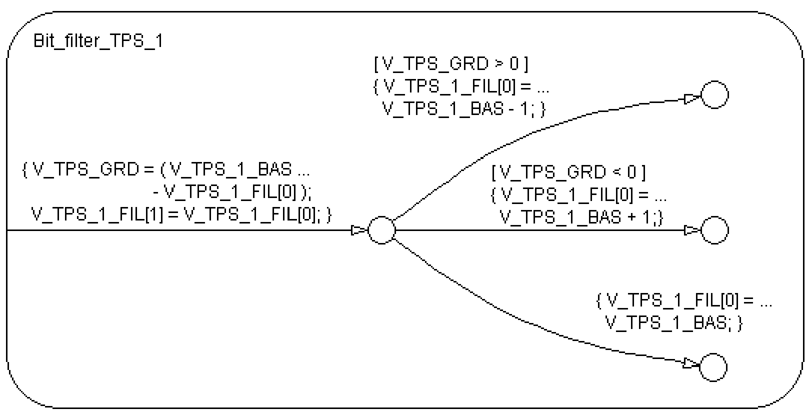
The time synchronous throttle position value TPS_ETC is needed as input for the electronic throttle control. The calculation of TPS_ETC is carried out every 2 ms.



4.25.1.1 TPS signal filter

Description:

The AD values of the throttle position sensors have a resolution of 10 bit (1024 increments) according to the AD converter. The smallest signification bit, one increment, is the fault of the AD conversion process and this bit can alternate from conversion to conversion. To compensate this fault a bit filter is used, see below.



Bit filter, general functionality (throttle position sensor 1)

Application conditions:

Initialisation: v_tps_x_fil[0] = v_tps_x_fil[1] = 0;

Recurrence: 1 ms

Activation: at every operating state

Formula section:

{ bit filter for TPS 1}
v_tps_grd_1 = V_TPS_1_BAS – v_tps_1_fil[0];

v_tps_1_fil[1] = v_tps_1_fil[0];

if v_tps_grd_1 > 0

then v_tps_1_fil[0] = V_TPS_1_BAS – 1

else **if** v_tps_grd_1 < 0

then v_tps_1_fil[0] = V_TPS_1_BAS + 1

else v_tps_1_fil[0] = V_TPS_1_BAS

endif

{ bit filter for TPS 2 }
v_tps_grd_2 = V_TPS_2_BAS – v_tps_2_fil[0];

v_tps_2_fil[1] = v_tps_2_fil[0];

if v_tps_grd_2 > 0

then v_tps_2_fil[0] = V_TPS_2_BAS – 1

else **if** v_tps_grd_2 < 0

then v_tps_2_fil[0] = V_TPS_2_BAS + 1

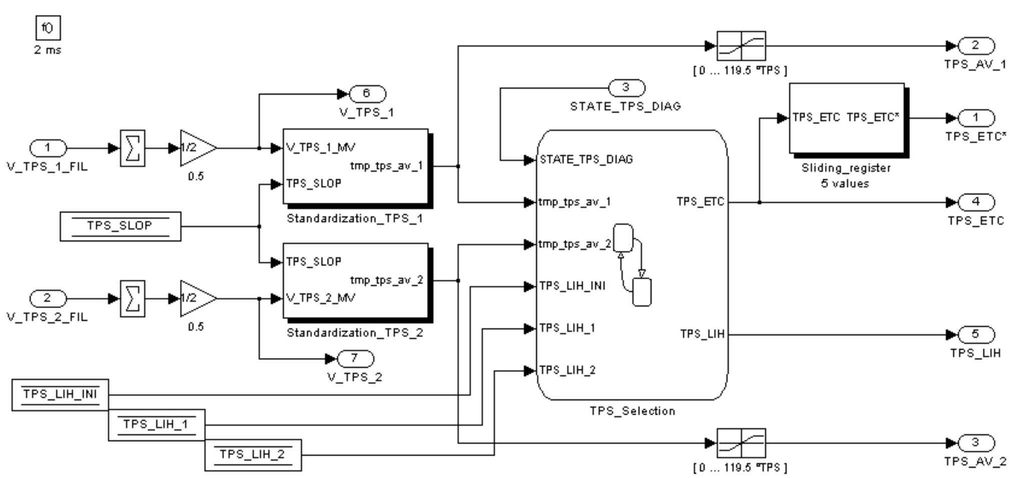
else v_tps_2_fil[0] = V_TPS_2_BAS

endif

4.25.1.2 Standardization of the TPS-Voltage

Description:

With the help of the determined adaptation values, the TPS-Position is standardized using the standard slope TPS_SLOP of the throttle position sensors.



Application conditions:

Initialisation: TPS_AV_x = C_TPS_MAX

Recurrence: 2 ms

Activation: at every operating state

Formula section:

{ The values ..._mv are meant to be local variables with a resolution 0,00244 V instead of 0,00488 V for V_TPS_x. These values are introduced to do the calculation below with higher resolution. }

V_TPS_x = v_tps_x_mv = 0.5 * (v_tps_x_fil[0] + v_tps_x_fil[1])

{ We introduce logical variables tmp_tps_av_x with physical limits of -119.51 ... 119.5 °TPS which are also needed in the section " Selection of throttle control value TPS_ETC " . }

actual value TPS 1 *{ Calculation with a higher resolution than V_TPS_1 }*

tmp_tps_av_1 = (v_tps_1_mv – V_TPS_AD_EL_BOL_1) * TPS_SLOP

actual value TPS 2 *{ Calculation with a higher resolution than V_TPS_2 }*

tmp_tps_av_2 = (V_TPS_AD_EL_BOL_2 – v_tps_2_mv) * TPS_SLOP

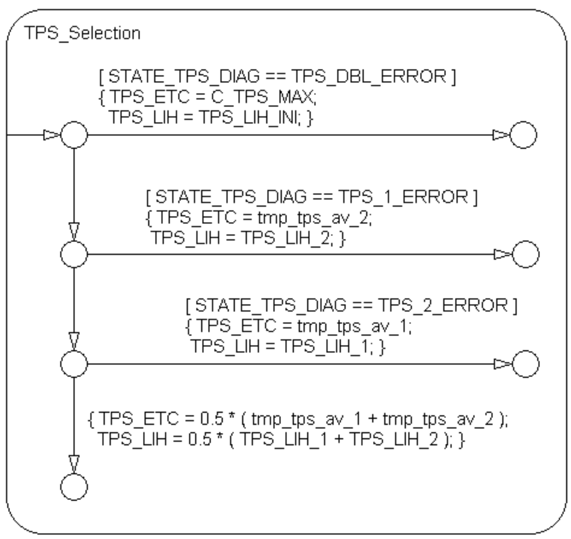
{ The actual values are limited to values greater than zero }

TPS_AV_x = MAX(tmp_tps_av_x, 0)

4.25.1.3 Selection of throttle control value TPS_ETC

Description:

The selection of electronic throttle control value TPS_ETC and the limp home position TPS_LIH depends on the status of TPS diagnosis.



Application conditions:

Initialisation: TPS_LIH = TPS_LIH_INI
TPS_ETC = C_TPS_MAX

Recurrence: 2 ms

Activation: at every operating state

Formula section:

```
{ TPS1 & TPS2 erroneous }
if    STATE_TPS_DIAG == TPS_DBL_ERROR
then        TPS_ETC = C_TPS_MAX
            TPS_LIH  = TPS_LIH_INI
{ TPS 1 erroneous }
else  if    STATE_TPS_DIAG == TPS_1_ERROR
      then        TPS_ETC = tmp_tps_av_2
                  TPS_LIH = TPS_LIH_2
      { TPS 2 erroneous }
      else  if    STATE_TPS_DIAG == TPS_2_ERROR
            then  TPS_ETC = tmp_tps_av_1
                  TPS_LIH = TPS_LIH_1
            { TPS1 & TPS2 errorfree, no TPS plausibility error }
            else  TPS_ETC = 0.5 * ( tmp_tps_av_1 + tmp_tps_av_2 )
                  TPS_LIH = 0.5 * ( TPS_LIH_1 + TPS_LIH_2 )
            endif
         endif
endif
```

4.25.1.4 Calculation of throttle position value TPS_AV/TPS

Description:

The TPS_ETC values are written into a buffer with 5 values, all 10 ms the mean value TPS_AV is calculated. TPS is the throttle position value TPS_AV with low resolution.

Application conditions:

Initialisation: TPS_AV = TPS = TPS_ETC(k-n) = C_TPS_MAX

Recurrence: 10ms

Activation: at every operating state

Formula section:

```
TPS_AV  = 0.2 * [ TPS_ETC(k) + TPS_ETC(k-1) + TPS_ETC(k-2) ...
                + TPS_ETC(k-3) + TPS_ETC(k-4) ]
TPS     = TPS_AV
```

4.25.1.5 Calculation of throttle position gradient: (TPS_GRD)

Description:

The gradient is calculated as (absolute value of) difference from the actual throttle value and the throttle value k - 4 (that is the throttle value which was actual 40ms before).

Application conditions:

Initialisation: TPS_GRD = 0

LV_TPS_GRD_UP = 0

Recurrence: 10ms

Activation: at every operating state

Formula section:

TPS_GRD = | [TPS_AV(k) – TPS_AV(k-4)] * 25 |

The throttle gradient direction is calculated as follows

If [TPS_AV(k) – TPS_AV(k-4)] < 0

then LV_TPS_GRD_UP = 0 (negative throttle gradient)

else LV_TPS_GRD_UP = 1 (positive throttle gradient)

endif

4.25.2 Segment synchronous throttle valve position: (TPS_SEG)

Description:

The segment synchronous throttle valve position TPS_SEG is needed as input for the intake manifold model. The current throttle valve position TPS_ETC is copied in TPS_SEG to every segment interrupt, in case of TPS double error, TPS_SEG ist set to TPS_LIH_INI.

Application conditions:

Initialisation: TPS_SEG = TPS_LIH_INI

Recurrence: segment synchronous

Activation: at every operating state

Formula section:

{ TPS1 & TPS2 erroneous }

if STATE_TPS_DIAG == TPS_DBL_ERROR

then TPS_SEG = TPS_LIH_INI

else TPS_SEG = TPS_ETC

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TPS_MAX	1	0...3FFFh	0...119,5	0,0073	°TPS
Maximum mechanical opening angle of the throttle valve in degree					

4.26 Component and Adaptation Manager - INSY

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_AMP_AD_REQ_LAMB	O/V	0... 1H	0... 1	1	[-]
Flag to indicate that ambient pressure adaptation via lambda is requested (interface to EVAC)					
LV_ENA_AMP	O/V	0... 1H	0... 1	1	[-]
Enable flag for ambient pressure sensor					
LV_ENA_AMP_AD_LAMB	O/V	0... 1H	0... 1	1	[-]
Enable flag for ambient pressure adaptation (lambda based)					
LV_ENA_AMP_AD_MAP_MES	O/V	0... 1H	0... 1	1	[-]
Flag to indicate that ambient pressure adaptation via MAP_MES is requested					
LV_ENA_AMP_AD_PQ	O/V	0... 1H	0... 1	1	[-]
Enable flag for ambient pressure adaptation (at full and part load)					
LV_ENA_AR_RED_AD	O/V	0... 1H	0... 1	1	[-]
Enable flag for reduced area adaptation (additive and multiplicative parameter)					
LV_ENA_MAF	O/V	0... 1H	0... 1	1	[-]
Enable flag for mass air flow sensor					
LV_ENA_MAP	O/V	0... 1H	0... 1	1	[-]
Enable flag for manifold air pressure sensor					
LV_ENA_PUT	O/V	0... 1H	0... 1	1	[-]
Enable flag for pressure upstream throttle sensor					

Input Data:

NC_MAP_CONF	LV_MAP_SWI	ERR_SYM_EL_MAP	NC_AMP_CONF
LV_AMP_SWI	LV_END_DIAG_EL_MAP	LV_ERR_MAP_PLAUS_DIAG	LV_ERR_MAP
NC_MAF_CONF	LV_MAF_SWI	LV_ERR_MAF	LV_ERR_AMP
LV_ERR_AMP_PLAUS_DIAG	LC_TCHA_CONF	NC_CHRG_CONF	LV_CMD_EGRV_EGR_DIAG
LV_ERR_EGR_2	LV_ERR_LOAD_PLAUS	T_AST	NC_CBK_EX_NR
LV_ERR_FSD [NC_CBK_EX_NR]	LV_LAM_LSCL [NC_CBK_EX_NR]	LV_LDC_LAM_AD [NC_CBK_EX_NR]	LV_CP_CLOSE_ACT
LV_CP_ENA	LV_CDN_DIAG_EL_MAP	LV_END_DIAG_AMP	LV_CDN_DIAG_AMP
ERR_SYM_AMP	NC_PUT_CONF	LV_PUT_SWI	ERR_SYM_EL_PUT
LV_CDN_DIAG_EL_PUT	LV_END_DIAG_EL_PUT	LV_ERR_PUT_PLAUS	LV_ERR_PUT

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_T_AST_AMP_AD_MAP_MES	1	0... FFFFH	0... 6553.5	0.1	[s]
Time after start which must have passed so that AMP adaptation is allowed by means of MAP_MES					
LC_AMP_AD_MAP_MES_SWI	1	0... 1H	0... 1	1	[-]
Switch whether ambient pressure adaptation is done via adaptation algorithm (= 0) or via MAP_MES (= 1)					

General Information

There are several optional components that are not mandatory for every system configuration. These are mainly sensors, but components like charger and EGR are optional as well. The mandatory air path functions are designed to work both with or without and with several combinations of these optional components. Some optional components can be switched on/off if they are present (e.g. EGR), other cannot be switched off if they are present (e.g. charger).

general specification

In dependence of the presence and recognized malfunction of the components several adaptations related to these components can be allowed or inhibited.
The function of the present Component and Adaptation Manager module is:
- to collect configuration information (information about the presence of components in the system),
- to collect error state information (information about malfunction of existing components),
- to enable or disable the usage of these components,
- to allow or inhibit adaptations related to these components.

Function description

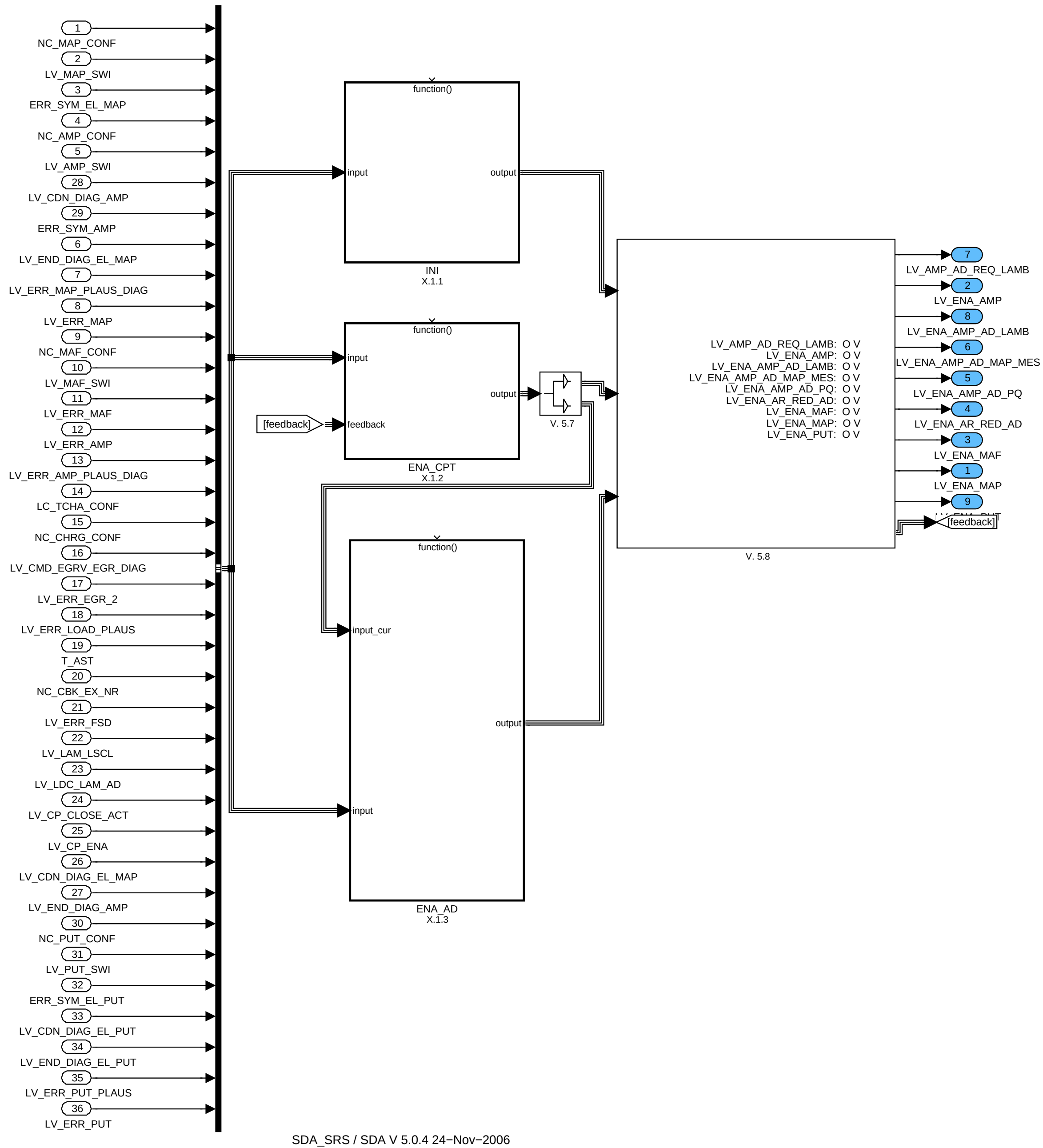


Figure 315:
Path: INSY_MANAG0

4.26.1 Sensor check at events RST, IGKON and CLFMY:

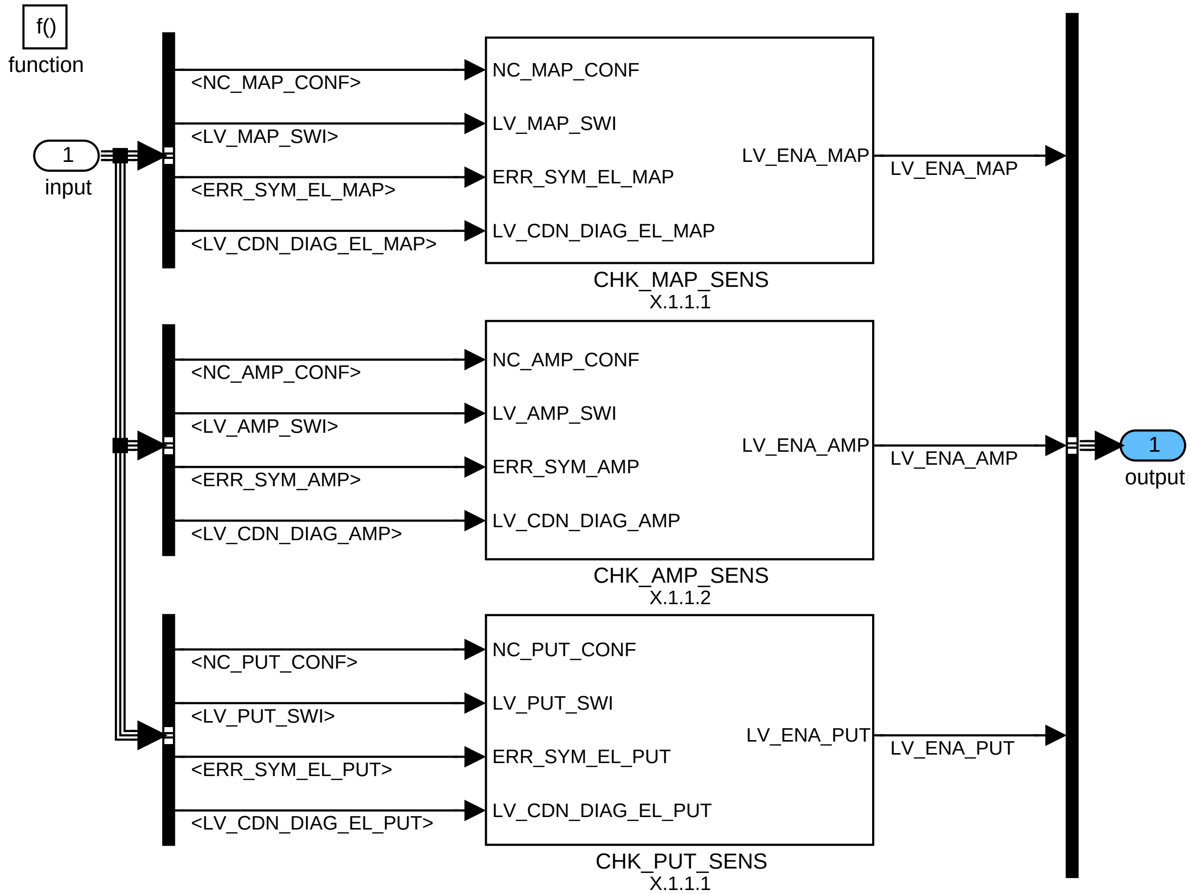


Figure 316:
Path: INSY_MANAG0/INI

4.26.1.1 Check MAP sensor

If a MAP sensor is present and no MAP sensor malfunction has been recognized, the use of the MAP sensor is enabled. This has to be checked at reset for the first time and at IGKON and CLFMY.

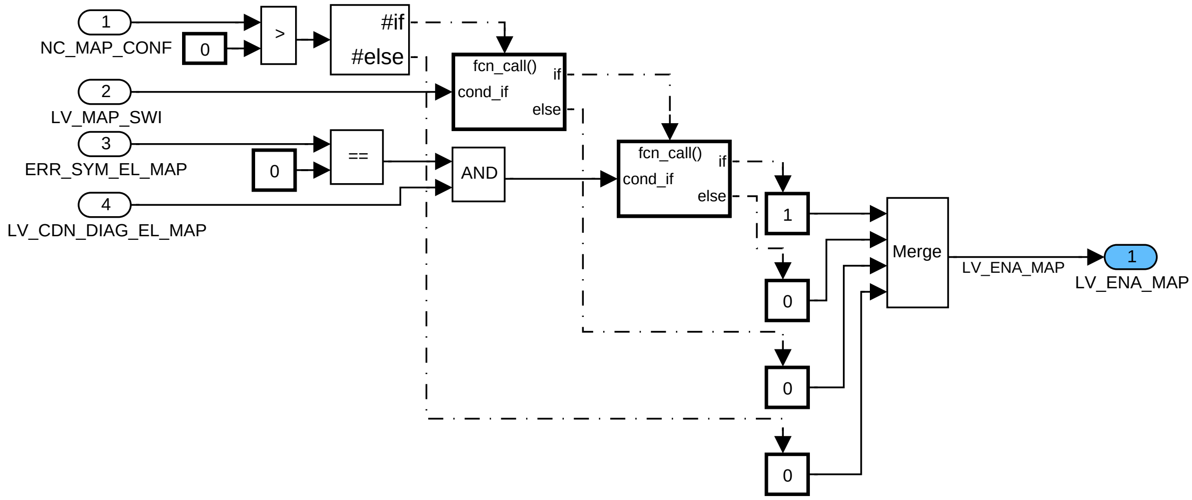


Figure 317:
Path: INSY_MANAG0/INI/CHK_MAP_SENS

4.26.1.2 Check AMP sensor

If an AMP sensor is present and no AMP sensor malfunction has been recognized, the use of the AMP sensor is enabled. This has to be checked at reset for the first time and at IGKON and CLFMY.

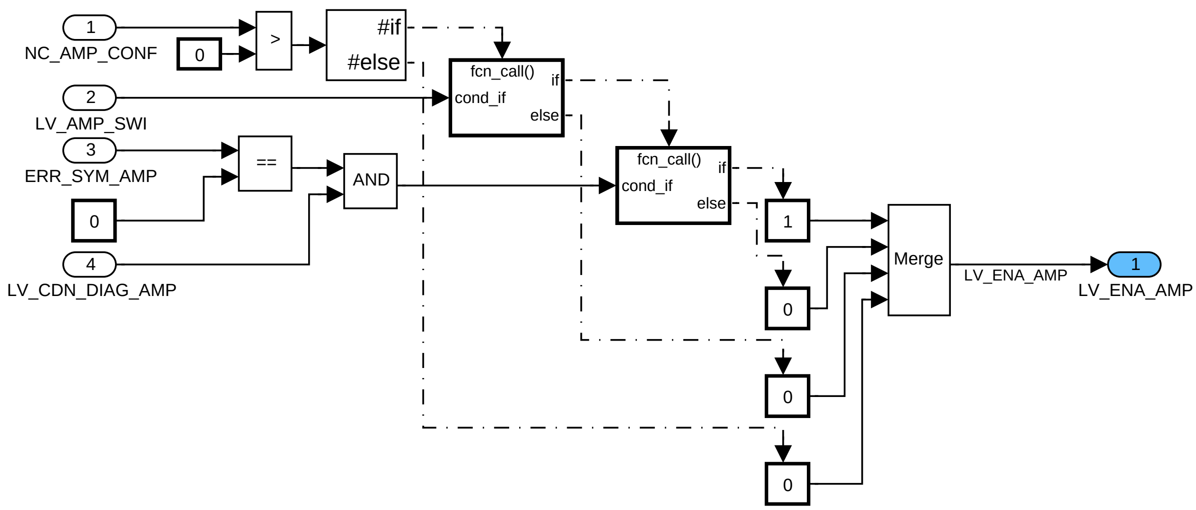


Figure 318:
Path: INSY_MANAG0/INI/CHK_AMP_SENS

4.26.1.3 Check PUT sensor

If a PUT sensor is present and no PUT sensor malfunction has been recognized, the use of the PUT sensor is enabled. This has to be checked at reset for the first time and at IGKON and CLFMY.

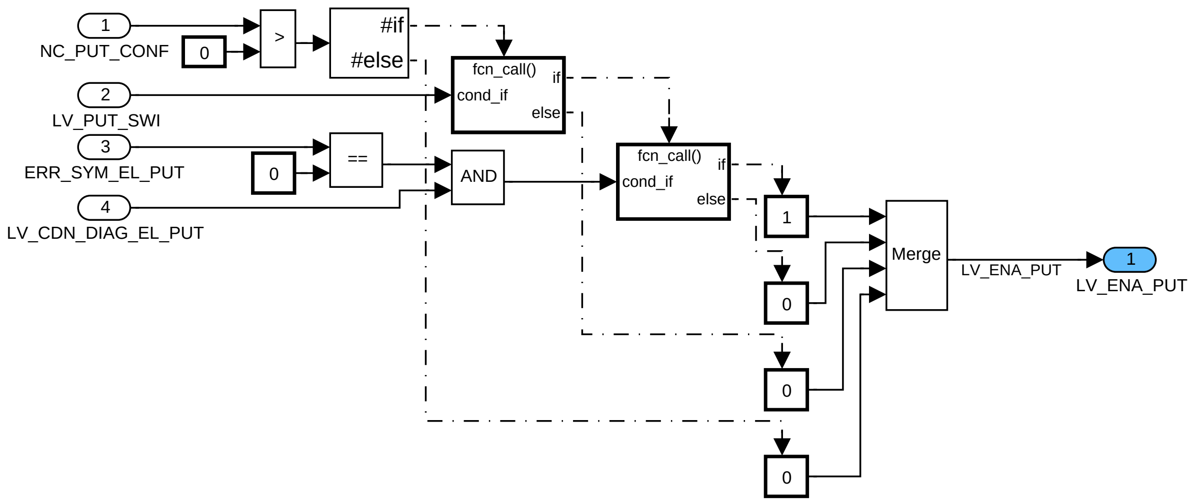


Figure 319:
Path: INSY_MANAG0/INI/CHK_PUT_SENS

4.26.2 Enabling components

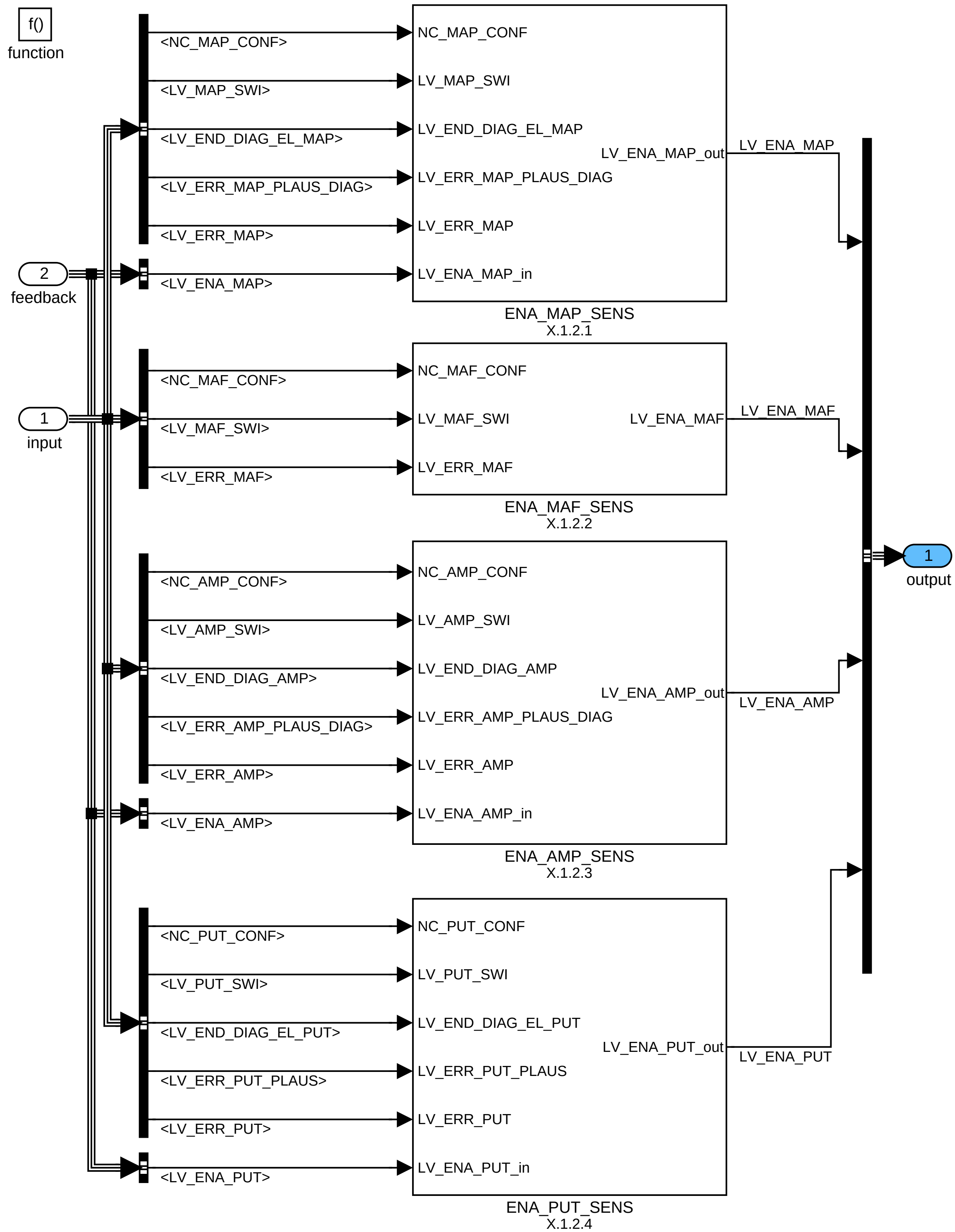


Figure 320:
Path: INSY_MANAG0/ENA_CPT

4.26.2.1 Enabling usage of manifold air pressure sensor

If a MAP sensor is present and no MAP sensor malfunction has been recognized, the usage of the MAP sensor is enabled.

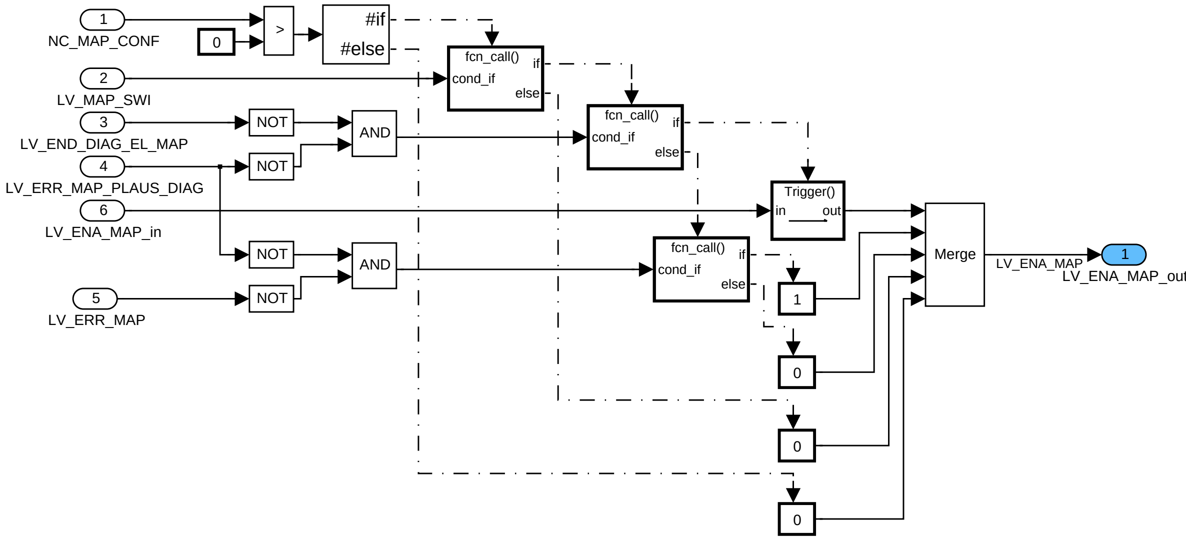


Figure 321:
Path: INSY_MANAG0/ENA_CPT/ENA_MAP_SENS

4.26.2.2 Enabling usage of mass air flow sensor

If a MAF sensor is present and no MAF sensor malfunction has been recognized, the usage of the MAF sensor is enabled.

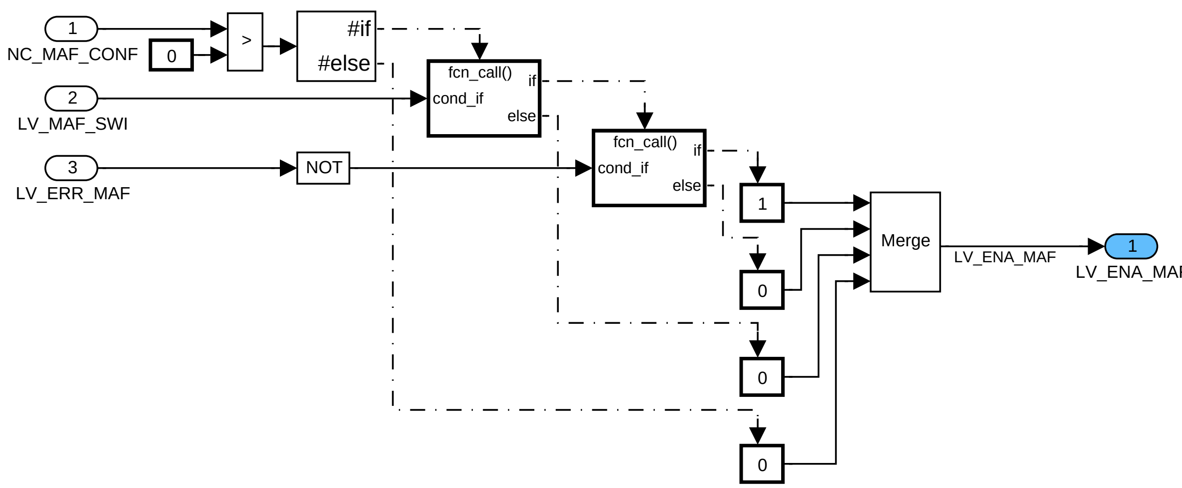


Figure 322:
Path: INSY_MANAG0/ENA_CPT/ENA_MAF_SENS

4.26.2.3 Enabling usage of ambient pressure sensor

If an AMP sensor is present and no AMP sensor malfunction has been recognized, the usage of the AMP sensor is enabled.

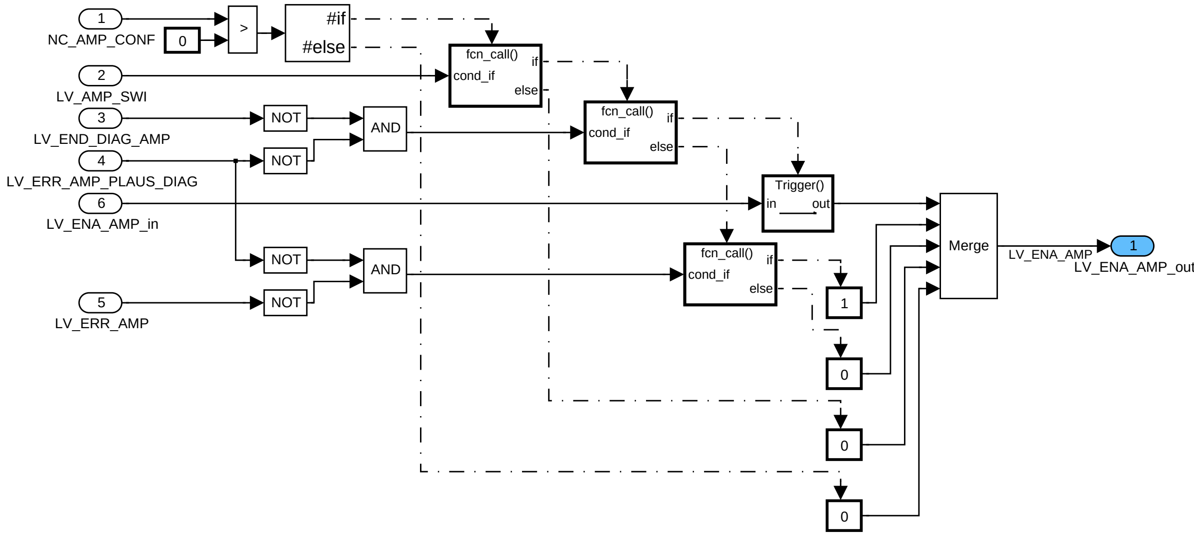


Figure 323:
Path: INSY_MANAG0/ENA_CPT/ENA_AMP_SENS

4.26.2.4 Enabling usage of pressure upstream of throttle sensor

If a PUT sensor is present and no PUT sensor malfunction has been recognized, the usage of the PUT sensor is enabled.

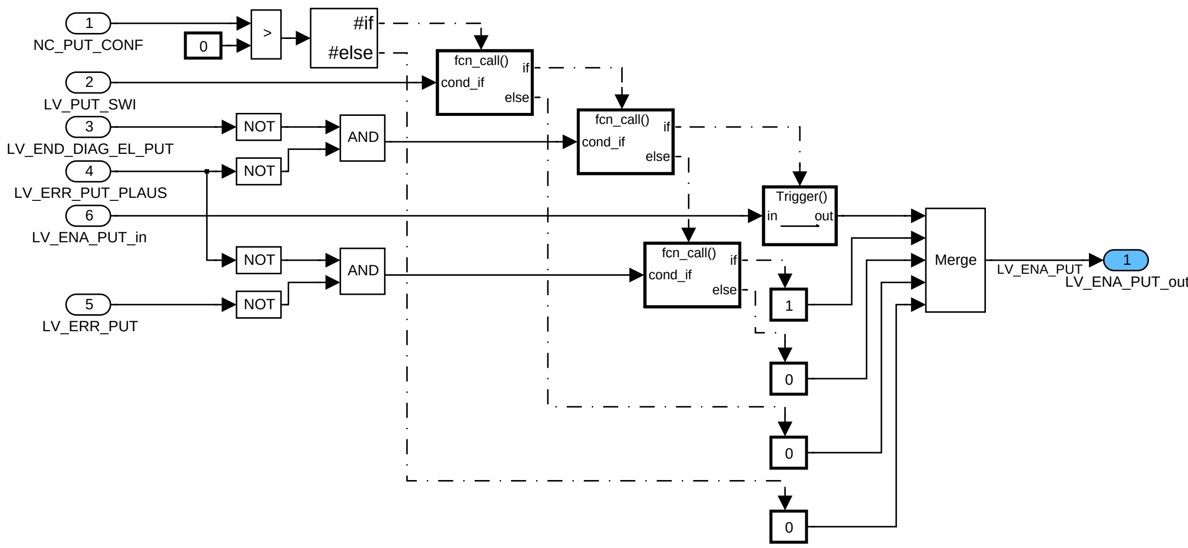


Figure 324:
Path: INSY_MANAG0/ENA_CPT/ENA_PUT_SENS

4.26.3 Enabling adaptations

Besides from specific enabling conditions the presence of a faulty load sensor (LV_ERR_LOAD_PLAUS) or a stuck EGR valve (LV_ERR_EGR_2 = 1) inhibits any adaptation.

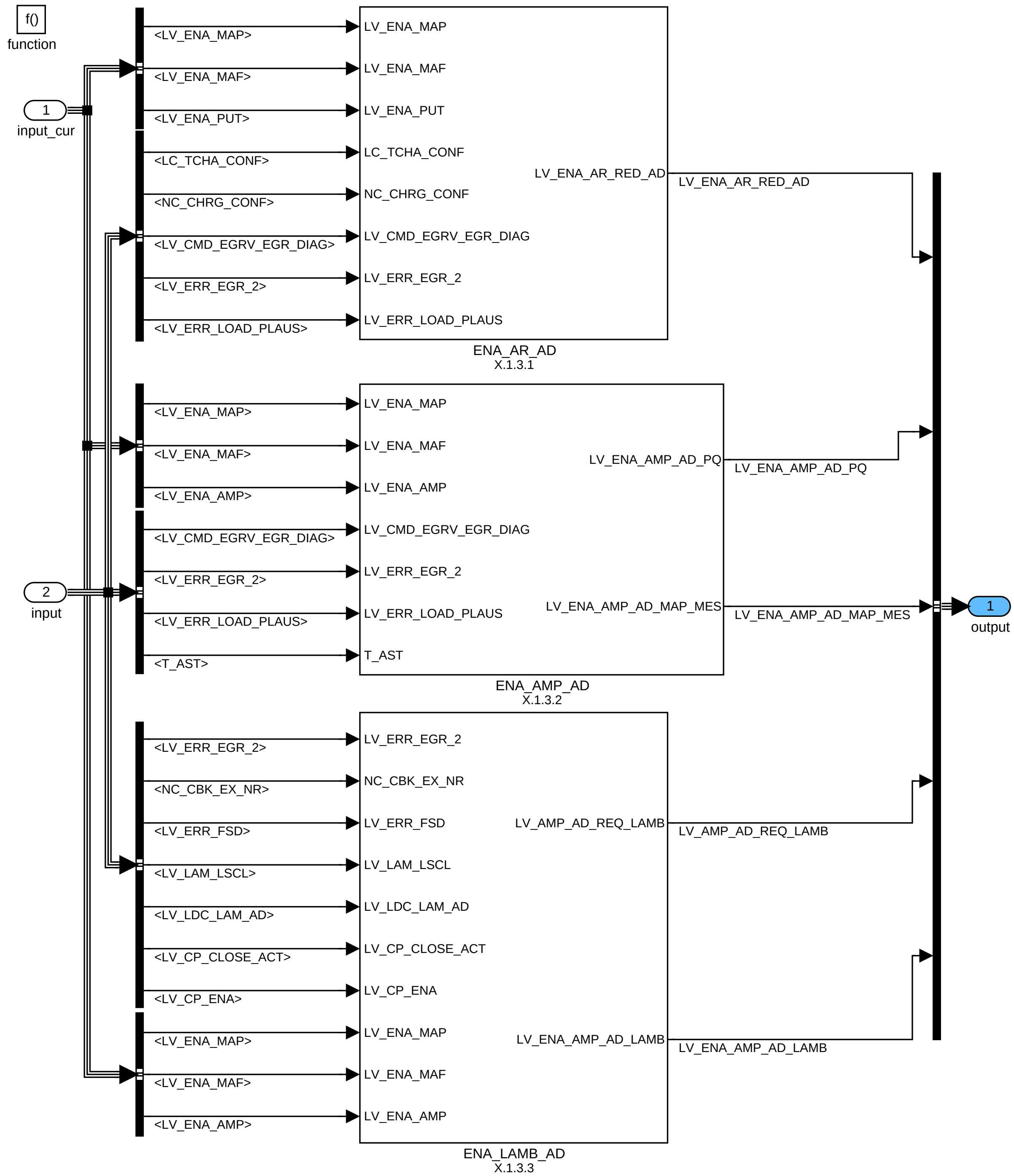


Figure 325:
Path: INSY_MANAG0/ENA_AD

4.26.3.1 Enabling reduced area adaptation

Reduced area adaptation is only possible
- with one working load sensor and
- the EGR adaptation that has higher priority is not active (LV_CMD_EGRV_EGR_DIAG = 0).

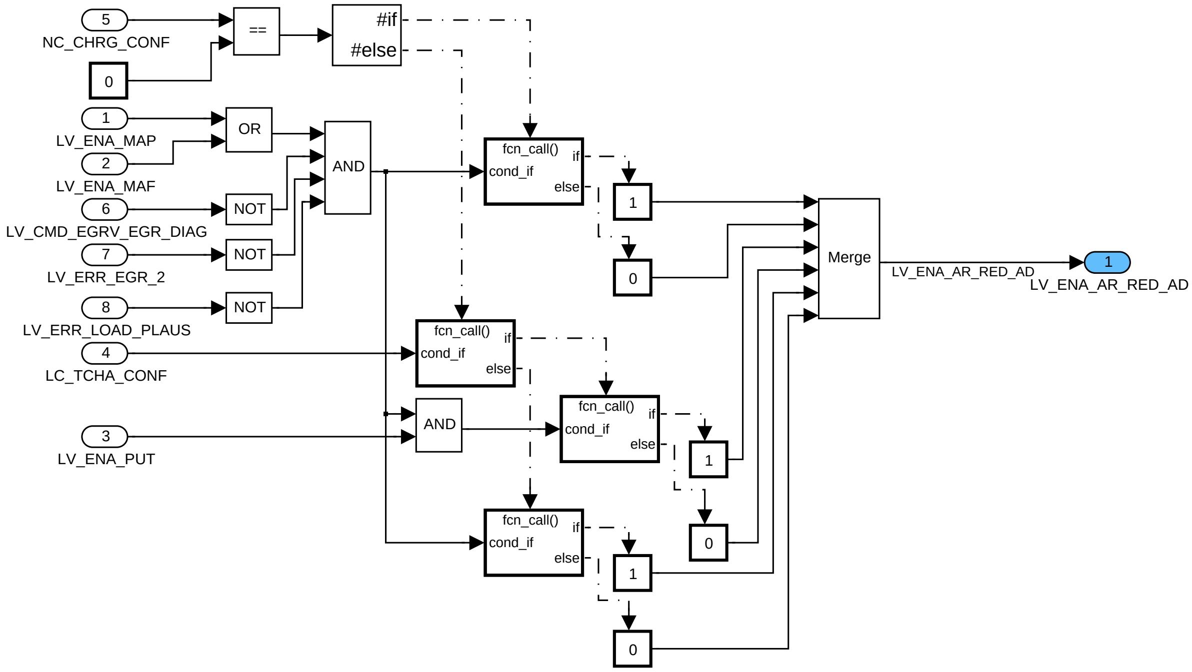


Figure 326:
Path: INSY_MANAG0/ENA_AD/ENA_AR_AD

4.26.3.2 Enabling ambient pressure adaptation with load sensor

Ambient pressure adaptation with load sensor is only possible
- without working ambient pressure sensor,
- with one working load sensor,
- The EGR adaptation that has higher priority is not active (LV_CMD_EGRV_EGR_DIAG = 0)

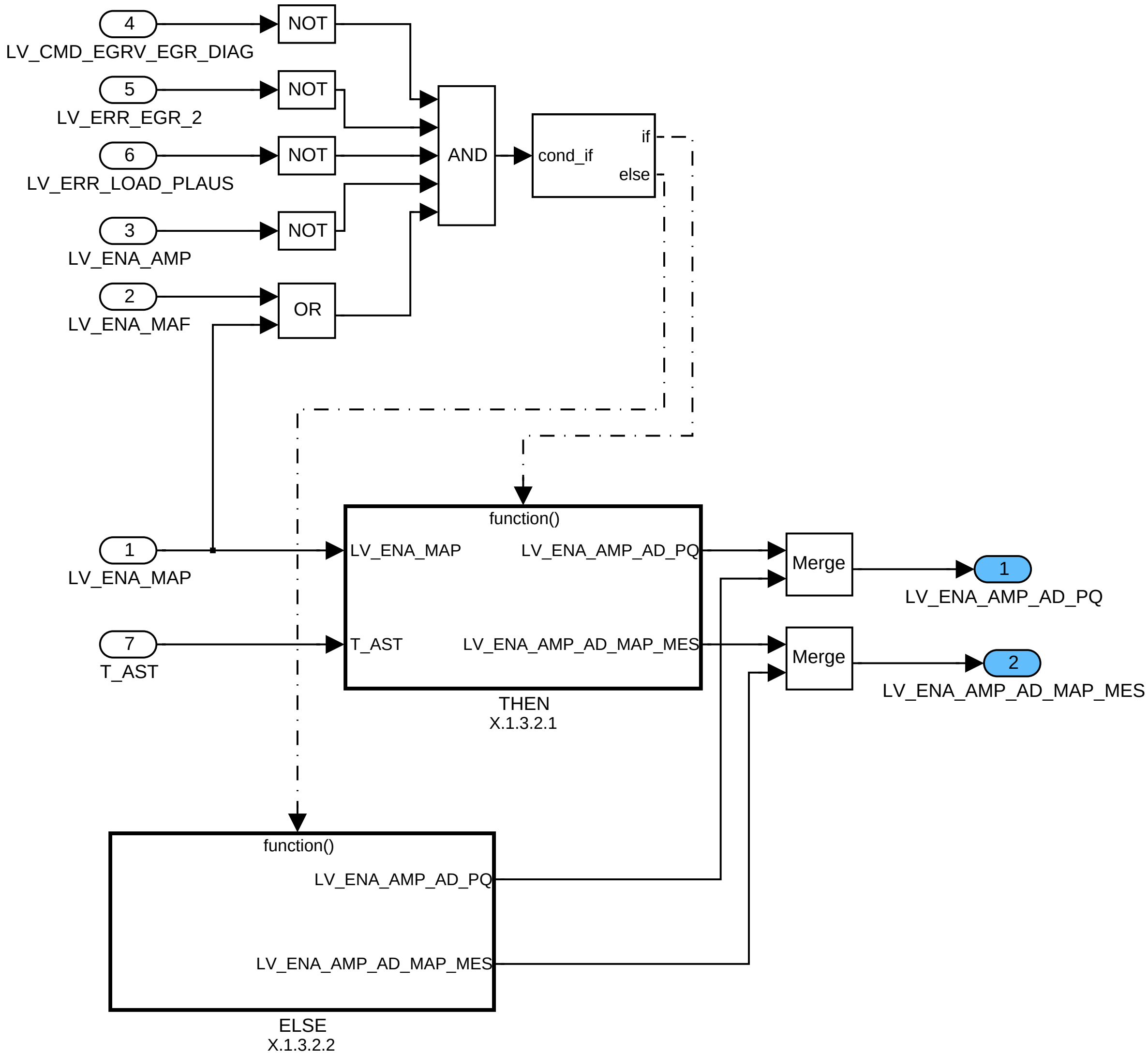


Figure 327:
Path: INSY_MANAG0/ENA_AD/ENA_AMP_AD

4.26.3.2.1 Enabling ambient pressure adaptation with load sensor - then branch:

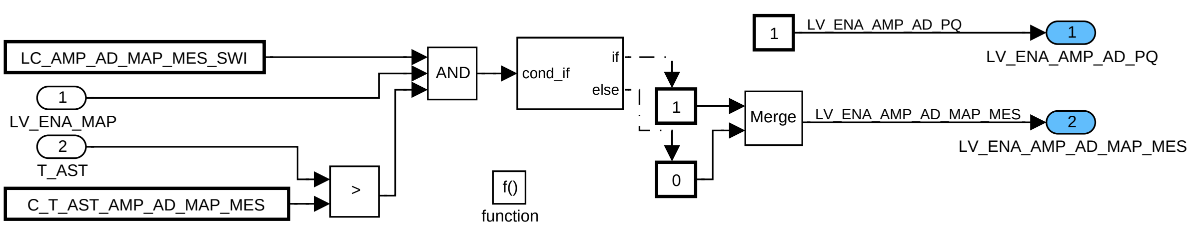


Figure 328:
Path: INSY_MANAG0/ENA_AD/ENA_AMP_AD/THEN

4.26.3.2.2 Enabling ambient pressure adaptation with load sensor - else branch:

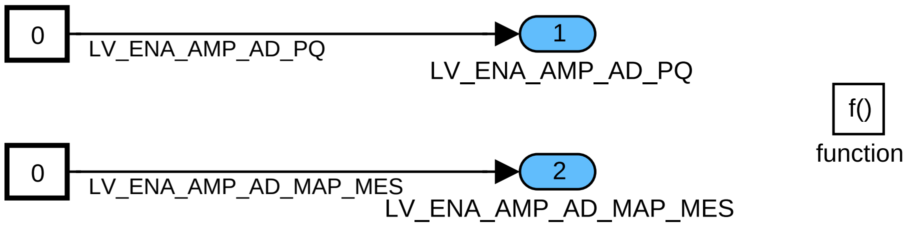


Figure 329:
Path: INSY_MANAG0/ENA_AD/ENA_AMP_AD/ELSE

4.26.3.3 Enabling lambda based ambient pressure adaptation

Lambda based ambient pressure adaptation is only possible
- without working ambient pressure sensor,
- without working load sensors

The calculation shall be done for all exhaust cylinder banks:
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

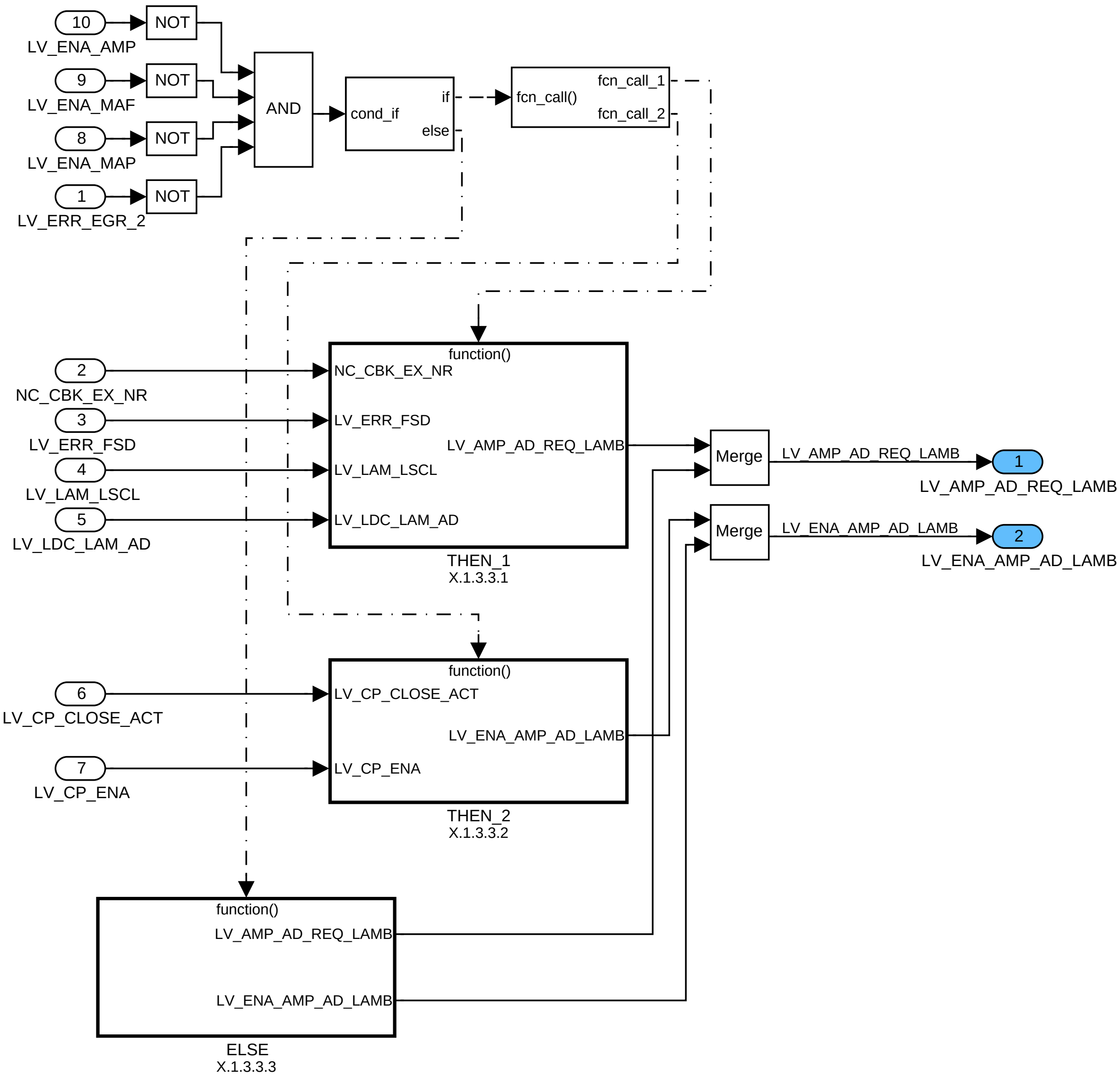


Figure 330:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD

4.26.3.3.1 Enabling lambda based ambient pressure adaptation - then 1 branch:

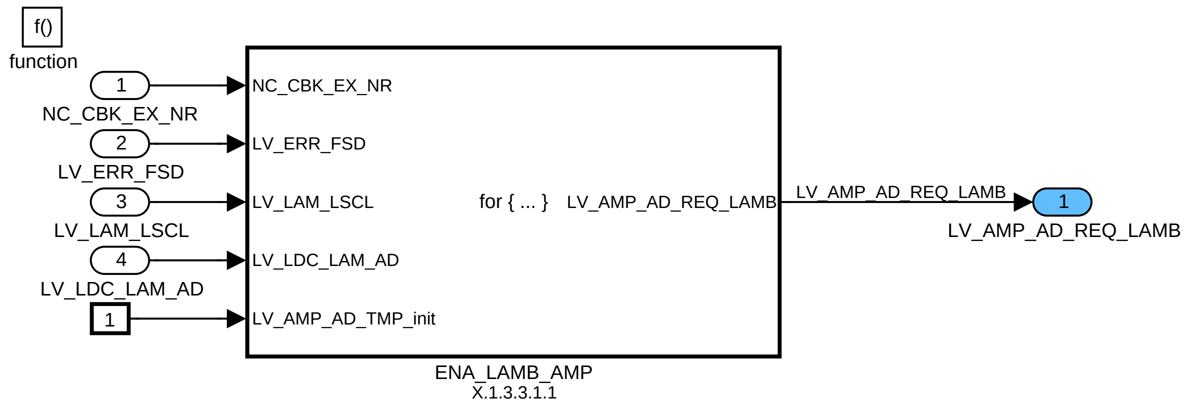


Figure 331:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD/THEN_1

4.26.3.3.1.1 Enabling lambda based ambient pressure adaptation - then 1 branch: for loop

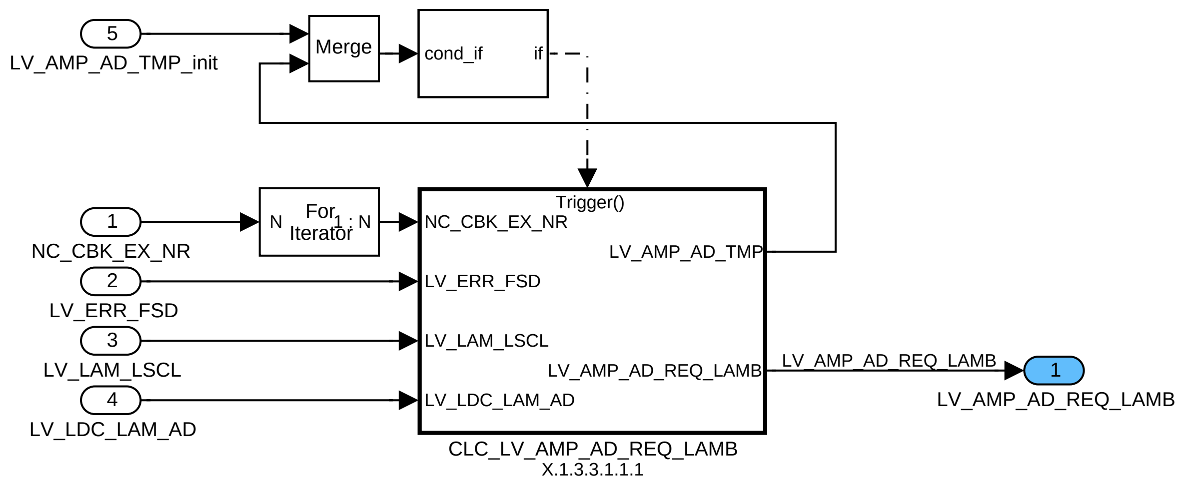


Figure 332:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD/THEN_1/ENA_LAMB_AMP

4.26.3.3.1.1.1 Enabling lambda based ambient pressure adaptation - then 1 branch: Calculation of LV_AMP_AD_REQ_LAMB

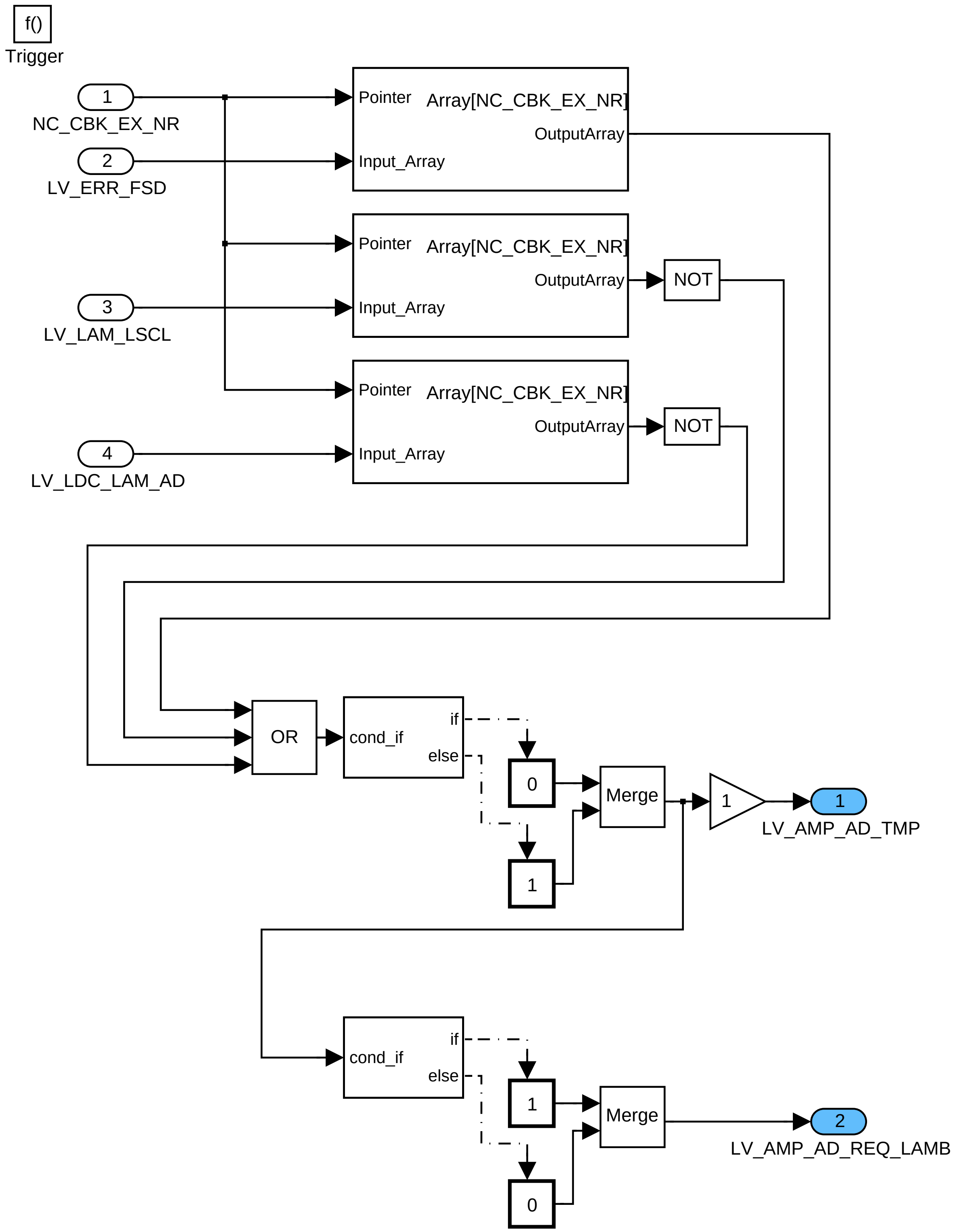


Figure 333:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD/THEN_1/ENA_LAMB_AMP/CLC_LV_AMP_AD_REQ_LAMB

4.26.3.3.2 Enabling lambda based ambient pressure adaptation - then 2 branch:

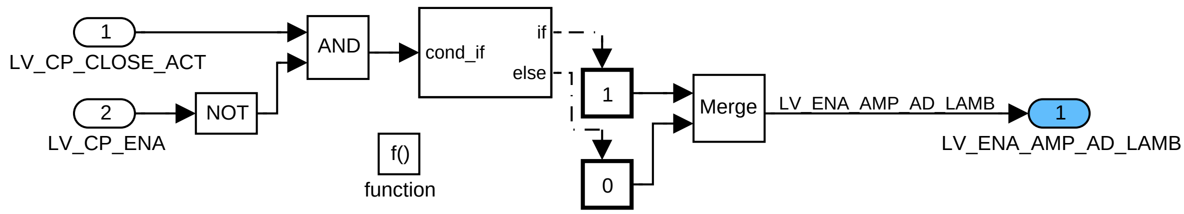


Figure 334:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD/THEN_2

4.26.3.3.3 Enabling lambda based ambient pressure adaptation - else branch:

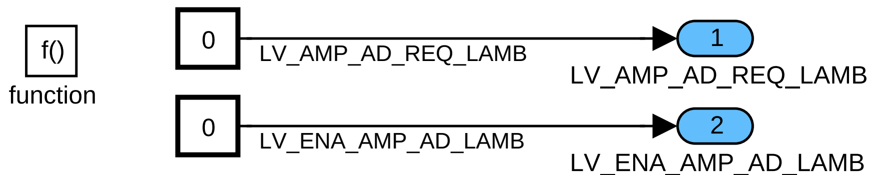


Figure 335:
Path: INSY_MANAG0/ENA_AD/ENA_LAMB_AD/ELSE

4.27 Vehicle Speed Variables

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS	V/O	0...FFH	0...255	1	km/h
vehicle speed					
VS_MAX	O	0...FFH	0...255	1	km/h
Maximum vehicle speed.					
VS_FAC	O	0 … 1FFFFFFFH	0 … 0,582542	1,736109 *10E-8	m
Factor to calculate vehicle speed in km/h from signal pulses					
LV_VS_RUN	V/O	0...1H	0...1	1	-
Boolean for running vehicle					
VS_WORD	V/O	0...FFFFH	0...511.992	0.0078125	km/h
Vehicle speed (16 bits)					
VS_MAX_DC	V/O	0...FFH	0...255	1	km/h
Maximum vehicle speed during current driving cycle					

Input data:

VS_SENS_XX	CONF_VS	VS_TCU_CAN	

FUNCTION DESCRIPTION:

General information:

Running vehicle is detected if vehicle speed VS >= C_VS_MIN_RUN.

Activation : at every engine operating state

Deactivation : -

Initialization : VS = 0

LV_VS_RUN = 0

VS_MAX_DC = 0

Update rate : 10ms

With the configuration byte CONF_VS, different possibilities of vehicle speed acquisition are available (gear box, inductive wheel speed sensor, ABS control unit, TCU via CAN).

The following variables are automatically adjusted according to CONF_VS:

- VS_MAX : maximum vehicle speed
- VS_FAC : factor to make the conversion from pulse to distance(m)

Filtering function:

To prevent for sudden VS signal jumps, a function is used to filter implausible changes of the vehicle speed signal. If the difference between 2 following values is grater than a calibratable constant, the previous value is kept.

If more than 5 following values have been determined as implausible the latest value is used.

Formula section:

VS_MIN, VS_MAX, VS_FAC calculation:

```
If      CONF_VS = "GEAR_BOX" (0 hex)
then    VS_MAX  =  C_VS_MAX_GB
        VS_FAC  =  C_VS_FAC_GB
else    VS_FAC = C_VS_FAC_ABS_WHEEL
        if      CONF_VS = "TCU_CAN" (4 hex)
        then    VS_MAX = C_VS_MAX_TCU_CAN
        else    VS_MAX =  C_VS_MAX_ABS_WHEEL
        endif
endif
```

The calculation is performed once after system reset.

VS calculation :

```
If      CONF_VS = "WHEEL"
then    VS = VS_SENS_WHEEL
else    if      CONF_VS = "TCU_CAN"
        then    VS = VS_TCU_CAN
        else    VS = VS_SENS_ABS_GB
        endif
endif
```

VS_MAX_DC calculation:

```
If      VS > VS_MAX_DC
then    VS_MAX_DC = VS
endif
```

VS_WORD calculation (high resolution – 16 bits) :

```
If      CONF_VS = "WHEEL"
then    VS = VS_SENS_WHEEL
else    if      CONF_VS = "TCU_CAN"
        then    VS = VS_TCU_CAN
        else    VS = VS_SENS_ABS_GB
        endif
endif
```

VS Filter:

If $| VS_{(n)} – VS_{(n-1)} | > C\_VS\_DIF\_MAX\_FIL$
and CTR_VS_FIL < 5
then $VS = VS_{(n-1)}$ *(previous value is kept)*
and CTR_VS_FIL = CTR_VS_FIL + 1
else $VS = VS_{(n)}$
and CTR_VS_FIL = 0

The Filter can be enabled by setting LC_VS_FIL_ENA = 1

LV_VS_RUN calculation :

If VS >= C_VS_MIN_RUN
then LV_VS_RUN = 1
else LV_VS_RUN = 0
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MIN_RUN	1	0...FFH	0...255	1	km/h
vehicle speed threshold for running vehicle detection					
C_VS_MAX_GB	1	0...FFH	0...255	1	km/h
Maximum Vehicle speed if CONF_VS = "GEAR_BOX"					
C_VS_MAX_ABS_WHEEL	1	0...FFH	0...255	1	km/h
Maximum Vehicle speed if CONF_VS = "ABS" or "WHEEL"					
C_VS_MAX_TCU_CAN	1	0...FFH	0...255	1	km/h
Maximum Vehicle speed if CONF_VS = "TCU_CAN"					
C_VS_FAC_GB	1	0 … 1FFFFFFH	0 … 0,582542	1,736109 *10E-8	m
Vehicle distance pulse if CONF_VS = "GEAR_BOX"					
C_VS_FAC_ABS_WHEEL	1	0 … 1FFFFFFH	0 … 0,582542	1,736109 *10E-8	m
Vehicle distance per pulse CONF_VS = "ABS" or "WHEEL"					
C_VS_DIF_MAX_FIL	1	0 … FFH	0 … 255	1	km/h
Filtering threshold for VS changes					
LC_VS_FIL_ENA	1	0 … 1H	0 … 1	1	-
Activation constant for VS filtering function					

4.27.1 Internal vehicle speed signal

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_SENS_XX	V/O	0 ... FFFFH	0...1023,984375	0,015625	km/h
vehicle speed calculated from corresponding input signal					
VS_SENS_GRD_XX	V	8000 ... 7FFFH	-256 ... 255,9921875	0,0078125	km/h / 10ms
Vehicle speed gradient calculated from VS_SENS_XX					
VS_EDGE_CTR_AV_XX	V	0 ... FFFFh	0 ... 65565	1	-
actual value of vehicle speed tooth counter (free running counter)					
VS_EDGE_T_AV_XX	V	0 ... FFFFFFFFh	0 ... 17179869	0,004	ms
actual timestamp of last tooth (free running timer)					
LV_VST	V	0 ... 1H	0 ... 1	1	-
boolean for stopped vehicle					

Input data:

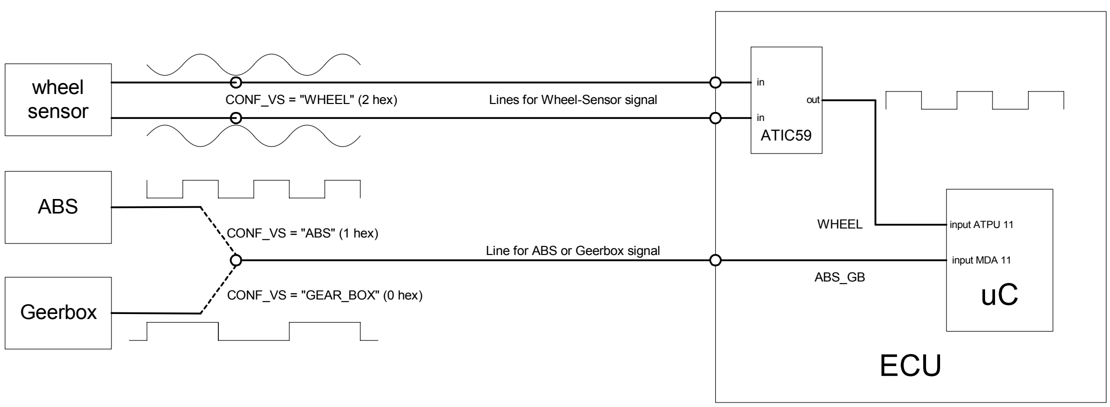
BIOS_VS_EDGE_CTR	BIOS_VS_EDGE_T	VS_FAC	

FUNCTION DESCRIPTION:

General information:

The source of the signal for the VS acquisition is defined by CONF_VS. The ECU has two different input lines: one for the signal from the inductive wheel speed sensor and one as common line for signal from ABS control unit or gear box sensor.
The data processing for each input line is done separately.

The two different input lines can be distinguished by the suffix XX:
with XX= "ABS_GB" for input from ABS contol unit or gear box sensor
"WHEEL" for input from inductive wheel speed sensor



Every 10 ms, VS_EDGE_CTR_AV_XX and VS_EDGE_T_AV_XX are updated based on BIOS_VS_EDGE_CTR and BIOS_VS_EDGE_T.

The edge counter is checked every 10ms and if new edge is detected ,

⇒ The logical variable for stopped vehicle LV_VST is set to 0.

⇒ If ($VS\_EDGE\_CTR\_AV\_XX_{n}$ - $VS\_EDGE\_CTR\_AV\_XX_{n-x}$) > C_VS_EDGE_SUM_MIN a new vehicle speed calculation is performed. VS_SENS_XX as well as VS_SENS_GRD_XX are calculated as described in formula section.

If no new edge counter is detected during the time C_T_VST_XX,
⇒ The logical variable for stopped vehicle LV_VST is set to 1.
⇒ VS_SENS_XX as well as VS_SENS_GRD_XX are set to 0.

Application conditions:

Activation : at every engine operating state

Deactivation : -

Initialization : VS_SENS_XX = 0
VS_SENS_GRD_XX = 0

LV_VST = 1

Update rate : 10ms

Formula section:

$$VS\_SENS\_XX = VS\_FAC * (VS\_EDGE\_CTR\_AV\_XX_{n} - VS\_EDGE\_CTR\_AV\_XX_{n-x}) / (VS\_EDGE\_T\_AV\_XX_{n} - VS\_EDGE\_T\_AV\_XX_{n-x})$$

$$VS\_SENS\_GRD\_XX = (VS\_SENS\_XX_{n} – VS\_SENS\_XX_{n-x}) / (VS\_EDGE\_T\_AV\_XX_{n} - VS\_EDGE\_T\_AV\_XX_{n-x})$$

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_T_VST_GB	1	1 … FFH	10 … 2550	10	ms
Time delay for detection of vehicle stop in case C_CONF_VS =0					
C_T_VST_ABS_WHEEL	1	1 … FFH	10 … 2550	10	ms
Time delay for detection of vehicle stop in case C_CONF_VS = 1					
C_VS_EDGE_SUM_MIN	1	1 … FFH	1 … 255	1	-
minimum number of wheel segments for evaluation of vehicle speed					

4.27.2 Vehicle speed state VS_STATE_CFA

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_STATE_CFA	V/O	0...2H	0...2	1	-
Vehicle speed state for cooling fans.					

Input data:

VS	LV_ACCIN	LV_ERR_VS	LV_ERR_MWSS

FUNCTION DESCRIPTION:

General information:

The fans are controlled according to the vehicle speed. For this purpose, a vehicle speed state is determined. The vehicle speed thresholds are different if the air conditioner is selected or not.

Formula section:

Initialization:

At ECU initialization, VS_STATE_CFA is set to 0.

Calculation :

* LV_ACCIN = 0

If (VS_STATE_CFA = 0 **and** VS **>** C_VS_2_CFA)

then VS_STATE_CFA **= 1**

If (VS_STATE_CFA = 1 **and** VS **<** C_VS_2_CFA - C_VS_HYS_1_CFA)

then VS_STATE_CFA **= 0**

* LV_ACCIN = 1

If (VS_STATE_CFA = 0 **and** VS **>** C_VS_1_CFA)

then VS_STATE_CFA **= 1**

If (VS_STATE_CFA = 1 **and** VS **<** C_VS_1_CFA - C_VS_HYS_1_CFA)

then VS_STATE_CFA **= 0**

* Independently of LV_ACCIN

If (VS_STATE_CFA = 1 **and** VS **>** C_VS_3_CFA)

then VS_STATE_CFA **= 2**

If (VS_STATE_CFA = 2 **and** VS **<** C_VS_3_CFA - C_VS_HYS_2_CFA)

then VS_STATE_CFA **= 1**

Remark : C_VS_1_CFA **<** C_VS_2_CFA **<** C_VS_3_CFA

Limp Home :

If LV_ERR_VS **=** 1 *(Error currently present on vehicle speed sensor)*

or LV_ERR_MWSS = 1 *(Error currently present on MWS-sensor)*

then VS_STATE_CFA **=** 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_1_CFA	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for fans management.					
C_VS_2_CFA	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for fans management.					
C_VS_3_CFA	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for fans management.					
C_VS_HYS_1_CFA	1	0...FFH	0...255	1	km/h
Vehicle speed hysteresis for VS_STATE_CFA transition 1 -> 0.					
C_VS_HYS_2_CFA	1	0...FFH	0...255	1	km/h
Vehicle speed hysteresis for VS_STATE_CFA transition 2 -> 1.					

4.28 Vehicle Speed (Appl. Inc.)

4.28.1 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_MAX_TOT_DC	V/O/S	0...FFH	0...255	1	km/h
Former / current driving cycle maximum vehicle speed					

Input data:

VS	LV_IGK	ERR_SYM_VS	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

VS_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: *10 ms*

Activation: *LV_IGK = 1*

Deactivation: -

Formula section:

If ERR_SYM_VS = 0

Then If VS > VS_MAX_TOT_DC

Then VS_MAX_TOT_DC = VS

Endif

Endif

4.29 Mileage counter

4.29.1 Distance accumulator

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DIST	S/O/V	0...FFFFFFFFH	0...429496729,5	0,1	km
Distance accumulator [0,1 km]					
DIST_TMP	-	0...FFFFFFFFH	0...38177,487	8,88889E-06	m
Internal distance counter					
DIST_DC	O/V	0...FFFFFFFFH	0...429496729,5	0,1	km
Distance at current driving cycle					

Input data:

CONF_VS	VS_EDGE_CTR_AV_XX	VS_FAC	

FUNCTION DESCRIPTION:

General information:

This Function is to accumulate the covered distance since the first vehicle start in resolution of 100m for EOBD purpose.

Application conditions:

Activation : at every engine operating state

and if the VS signal is determinate (CONF_VS **is not** "LEARNING")

Deactivation : -

Initialization : at the first power on: DIST = 0

*at transition LV_IGK 0->*1 *or reset*: DIST_DC = 0

Update rate : 1s

In dependence of the configuration byte CONF_VS the pulse counter VS_EDGE_CTR_AV_XX is used for distance calculation.

With XX = 'ABS_GB' if ABS control unit or gear box sensor is used

'WHEEL' if inductive wheel speed sensor is used.

Formula section:

$$DIST\_TMP_{(n)} = DIST\_TMP_{(n-1)} + (VS\_FAC/512 * (VS\_EDGE\_CTR\_AV\_XX_{(n)} - VS\_EDGE\_CTR\_AV\_XX_{(n-1)}))$$

If DIST_TMP >= 100m

then DIST = DIST + 0,1 km

DIST_DC = DIST_DC + 0,1 km

DIST_TMP = DIST_TMP –100m

EndIf

4.29.2 Distance for fuel consuption rate

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DIST_FCO	O/V	0...FFFFFFH	0...16777.215	0,001	km
Distance for fuel consumption rate [0,001 km]					

Application conditions:

Initialisation: At ECU reset :

DIST_FCO = 0

Recurrence: 1s

Activation: at every engine operating state

and if the VS signal is determinate (CONF_VS **is not** “LEARNING”)

Deactivation: -

Formula section:

If $DIST\_FCO_{(n-1)}$ < 0xFFFFFF

Then $DIST\_FCO_{(n)} = DIST\_FCO_{(n-1)} + (DIST\_TMP_{(n)} – DIST\_TMP_{(n-1)}) * 0.001$

Else $DIST\_FCO_{(n)} = DIST\_FCO_{(n-1)}$

EndIf

Remark : After calculating difference of DIST_TMP, the distance which is smaller than 1m should be kept and added to next DIST_FCO(n+1) calculation.

4.29.3 Distance at last time failure meory cleared

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DIST_FMY	-/S	0...FFFFFFFFH	0...4294967295	1	km
Distance at last time failure memory cleared					

FUNCTION DESCRIPTION:

General information:

Purpose of this strategy is to memorize the distance of the last time the failure memory was cleared.

Application conditions:

Initialisation: on failure memory clear service received

DIST_FMY = DIST

Recurrence: -

Activation: -

Deactivation: -

4.30 Battery voltage VB

4.30.1 Computation of VB

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VB	V/O	0...FFH	0...26	0.102	V
Relay - battery voltage					
V_IGK	V/O	0...FFH	0...26	0.102	V
Key - battery voltage					

Input data:

VB_MES	V_IGK_MES	LV_ERR_VB_OC	

FUNCTION DESCRIPTION:

General information:

The battery voltage (V_IGK_MES and VB_MES (10 bit)) is measured up to 30V, 28V or 26V (means 0-5V at ADC), depending on the project-specific solution, but the range of VB (8 bit) remains 0-26 V. For the computation of VB in hex the correction factor NC_FAC_VB_RATIO is necessary; its value depends on the project-specific battery voltage range. NC_FAC_VB_RATIO is definded in the "List of Configuration Datas". The following table shows how NC_FAC_VB_RATIO is formed.

Measured battery voltage range (for example)	NC_FAC_VB_RATIO
0 ..... 26 V	26V / 26V
0 .. 28,8 V	28,8V / 26V
0 ..... 30 V	30V / 26V

Application conditions:

Activation: at reset

Initialisation: VB = 0

V_IGK = 0

Recurrence: 10 ms

Formula section:

HEX calculation:

VB = (VB_MES / 4) * NC_FAC_VB_RATIO

V_IGK = (V_IGK_MES / 4) * NC_FAC_VB_RATIO

VB relay:

If LV_ERR_VB_OC = 1
Then VB = V_IGK
Else **If** VB_MES <= 26 Volt
Then VB = (VB_MES / 4) * NC_FAC_VB_RATIO
Else VB = 26 Volt (FF hex)
Endif
Endif

VB key:

If V_IGK_MES <= 26 Volt
Then V_IGK = (V_IGK_MES / 4) * NC_FAC_VB_RATIO
Else V_IGK = 26 Volt (FF hex)
Endif

4.30.2 Secured battery voltage computation

Output data:

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
VB_SECU	V	0...FFH	0...26	0,102	V
Battery Voltage secured for dwell computation					

Input data:

VB			

Description:

A secured battery voltage value is a mix of VB and V_IGK values which provides a safer battery voltage for functions especially in case of VB - relay wire to the ECU disconnection.

The securate battery voltage value can be used for exemple for dwell computation in order to protect ignition coil and ECU ignition output stage against too long dwell time.

VB_SECU = MAX(VB, V_IGK - NC_VB_SECU)

with VB_SECU - NC_VB_SECU satured to zero

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_VB_SECU	1	0...FFH	0...26	0,102	V
Battery voltage offset (typical value = 1,02V)					
NC_FAC_VB_RATIO	1	0..FFH	0 ... 1,9922	0,0078125	-
Factor to adapt the HEX-range to the range of 0 to 26 V					

4.30.3 Battery voltage ranges

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_VB_JUMP	V/O	0...1H	0...1	1	-
Logical value for detected overvoltage (jump start)					
LV_CDN_VB_OBD1	V/O	0...1H	0...1	1	-
Boolean for battery voltage condition fulfilled for OBD diagnosis					
LV_CDN_VB_OBD2	V/O	0...1H	0...1	1	-
Boolean for battery voltage condition fulfilled for OBD-II diagnosis					
LV_CDN_VB_MIN_DIAG	V/O	0...1H	0...1	1	-
Boolean for battery voltage condition fulfilled for remaining diagnosis					
LV_CDN_VB_CAN_DIAG	V/O	0...1H	0...1	1	-
Boolean for battery voltage condition fulfilled for CAN diagnosis					

Input data:

VB	LV_IGK	C_VB_MIN_CAN_DIAG	

FUNCTION DESCRIPTION:

General information:

Depending on VB there is a detection:

- Jump start
- Valid range for OBD1 diagnosis
- Valid range for OBD2 diagnosis
- Valid range for CAN diagnosis
- Valid range for remaining customer diagnosis

Application conditions:

Initialisation: all 0

Recurrence: 10ms

Activation: at reset

4.30.3.1 Detection of overvoltage (Jump start)

Description:

The bit LV_VB_JUMP is used by functions which must not run when overvoltage is detected for example secondary air injection to prevent damages to the components.

Formula section:

Jump start detection:

If VB > C_VB_MIN_JUMP
Then LV_VB_JUMP = 1
Else LV_VB_JUMP = 0
Endif

4.30.3.2 Battery voltage valid for OBD diagnosis

Description:

Depending of OBD1/OBD2 there is a different flag for the valid VB range

Formula section:

Valid range for OBD1 diagnosis:

If VB > C_VB_MIN_OBD1 **and** VB < C_VB_MAX_OBD1
Then LV_CDN_VB_OBD1 = 1
Else LV_CDN_VB_OBD1 = 0
Endif

Valid range for OBD2 diagnosis:

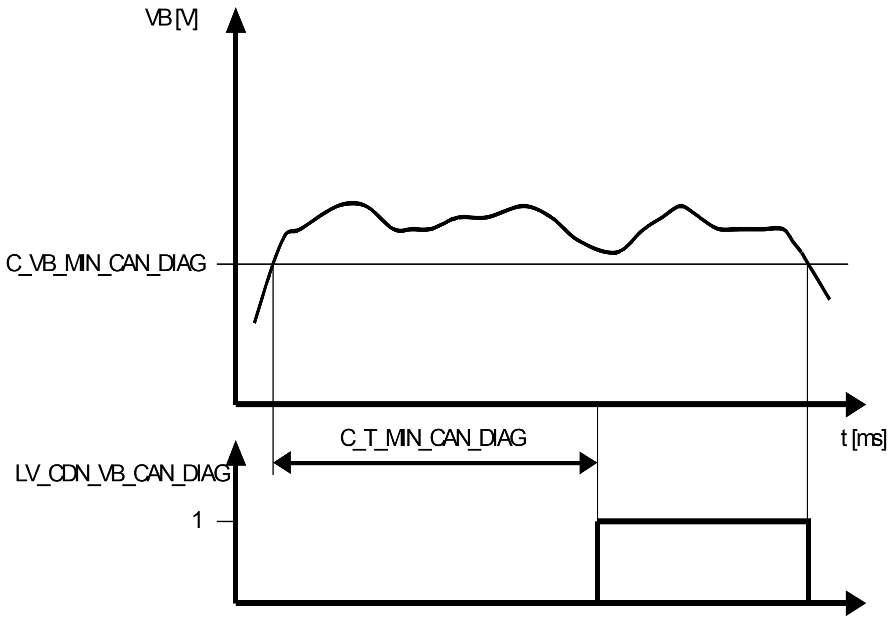
If VB > C_VB_MIN_OBD2 **and** VB < C_VB_MAX_OBD2
Then LV_CDN_VB_OBD2 = 1
Else LV_CDN_VB_OBD2 = 0
Endif

4.30.3.3 Minimum battery voltage detection for CAN diagnosis

Description:

Due to low battery voltage CAN messages can be mutilated. So CAN timeout diagnosis is only done if battery voltage is valid for a minimum duration time.

Signal flow diagram:



Formula section:

Valid range for CAN diagnosis:

If LV_IGK = 1 **and**

VB > C_VB_MIN_CAN_DIAG **for**

time > C_T_MIN_CAN_DIAG

(timer always started at every transition VB > C_VB_CAN_DIAG)

Then LV_CDN_VB_CAN_DIAG = 1

Else LV_CDN_VB_CAN_DIAG = 0

Endif

4.30.3.4 Minimum Battery Voltage Detection for non OBD Diagnosis

Description:

The remaining diagnosis functions which are not covered by the above mentioned conditions are disabled below a separate threshold, which should be lower as OBD1/2 and CAN.

Formula section:

Valid range for non OBD diagnosis:

If VB < C_VB_MIN_DIAG **or**
VB > C_VB_MAX_DIAG **or**
LV_IGK = 0
Then LV_CDN_VB_MIN_DIAG = 0
Else LV_CDN_VB_MIN_DIAG = 1
Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VB_MIN_JUMP	1	0...FFH	0...26	0.102	V
VB -threshold for maximum permissible battery voltage					
C_VB_MIN_OBD1	1	0...FFH	0...26	0.102	V
VB –threshold for minimum permissible battery voltage for OBD diagnosis					
C_VB_MAX_OBD1	1	0...FFH	0...26	0.102	V
VB -threshold for maximum permissible battery voltage for OBD diagnosis					
C_VB_MIN_OBD2	1	0...FFH	0...26	0.102	V
VB -threshold for minimum permissible battery voltage for OBD2 diagnosis					
C_VB_MAX_OBD2	1	0...FFH	0...26	0.102	V
VB -threshold for maximum permissible battery voltage for OBD2 diagnosis					
C_VB_MIN_DIAG	1	0...FFH	0...26	0,102	V
Battery voltage threshold for low voltage detection for remaining diagnosis					
C_VB_MAX_DIAG	1	0...FFH	0...26	0,102	V
Battery voltage threshold for high voltage detection for remaining diagnosis					
C_T_MIN_CAN_DIAG	1	0...FFH	0...2550	0.01	s
VB-Time-condition for CAN diagnosis					

4.30.4 Moving average value for battery voltage

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VB_MMV	V/O	0...FFH	0...26	0.102	V
Moving mean value of battery voltage VB					

Input data:

VB			

A moving average value VB_MMV is used in the chapter “Engine speed setpoint calculation” to define a minimum N_SP_IS in dependance of the battery voltage:

Application conditions:

Activation: *at reset*

Initialization: VB_MMV = 14 V

Recurrence: 10 ms

Formula section:

$VB\_MMV_k = VB\_MMV_{k-1} + (VB_k - VB\_MMV_{k-1}) \cdot C\_CRLC\_VB.$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_VB	1	0...FFH	0...0.996	0.0039	-
Low pass filter correlation constant for VB_MMV					

4.31 TIA_THR_UP, TIA_TMP and TAM temperatures

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_TMP	V/O	0...FEH	-48...142.5	0.75	[°C]
Interface value of Air temperature used as an input for mappings					
TIA_THR_UP	V/O	0...FEH	-48...142.5	0.75	[°C]
Air interface temperature up throttle					
TAM	V/O/S	0...FEH	-48...142.5	0.75	[°C]
Ambient temperature					
TAM_ST	V/O	0...FEH	-48...142.5	0.75	[°C]
Ambient air temperature at Start					
T_TAM1	V	0...FFH	0...255	1	[s]
First timer for TAM model					
T_TAM2	V	0...FFH	0...255	1	[s]
Second timer for TAM model					
TAM_MDL	V	0...FEH	-48...142.5	0.75	[°C]
Modeled ambient temperature					
LV_TAM_VLD_DIAGCP	V/O	0...1H	0...1	1	-
Flag to indicate that TAM is valid for EVAP mon. : TAM_MDL updated at least one time since engine Start					
T_TAM_VLD_DIAGCP	V	0...FFFFH	0...65535	1	Sec.
Timer to reset LV_TAM_VLD_DIAGCP if no TAM_MDL learning take place (to ensure regular TAM_MDL update)					

Input data:

TIA[NC_SENS_NR_TIA]	LV_ERR_TAM_CAN	VS	LV_ST_END
ERR_SYM_TAM_CAN	TIA_CHRG_UP	TIA_CHRG_DOWN	NC_IDX_TIA_IM_CYL
NC_IDX_TIA_CHRG_THR	NC_TIA_CONF	NC_IDX_TIA_AM_CHRG	CONF_TAM
NC_SENS_NR_TIA	LV_ERR_ORNG_TAM	LV_ERR_PLAUS_TAM	ERR_SYM_ORNG_TAM
LV_TAM_CAN_FIRST_VLD	STATE_ENG	ERR_SYM_PLAUS_TAM	TAM_MES_MMV
ERR_SYM_EL_TAM	MAF_KGH	CONF_MAF	LV_T_ES_NOT_PLAUS
T_ES	TIA_THR		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TAM_INI_MIN	1	0...FEH	-48...142.5	0.75	[°C]
TAM value at initialization cnnot be lower than C_TAM_INI_MIN					
C_TAM_AS	1	0...FEH	-48...142.5	0.75	[°C]
Application system calibration for TAM					
C_TAM_OFS	1	0...FEH	0...190.5	0.75	[°C]
Temperature offset between TAM_TMP and TAM					
C_T_TAM1	1	0...FFH	0...255	1	[s]
Time threshold to correct TAM					
C_T_TAM2	1	0...FFH	0...255	1	[s]
Time threshold vehicle speed conditon can be violated without reset of timer					
C_VS_TAM_MIN	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed to calculate TAM					
LC_ERR_TAM_CAN_AS	1	0...1H	0...1	1	[-]
Application system switch for TAM subsitute formula validation: default value = 0					
LC_TAM_AS	1	0...1H	0...1	1	[-]
Application system value for TAM: TAM = C_TAM_AS if LC_TAM_AS = 1					
C_MAF_KGH_TAM_MIN	O/V	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Minimum air-mass flow to calculate TAM					
C_T_ES_TEMP_MIN	1	0... FFFFH	0...6.5535E+4	1	min
Minimum necessary engine shut off time duration (cold start condition reached)					

C_T_TAM_VLD_DIAGCP	1	0... FFFFH	0...65535	1	Sec
Time interval to reset LV_TAM_VLD_DIAGCP					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_TAM_CLC_BOL	-	0...FEH	-48...142.5	0.75	[°C]
Lower limit of TAM: typical value = -35°C					
NC_TAM_CLC_TOL	-	0...FEH	-48...142.5	0.75	[°C]
Upper limit of TAM: typical value = 50°C					

4.31.1 General Information [Version for : NC_CHRG_CONF <> 0 And NC_TIA_CONF = 10,11,12,13,21,22,23,24,30]

TIA_TMP:

TIA_TMP is defined as one of the TIA sensor values. In case of one TIA sensor only, then TIA_TMP equals this only sensor value. It will be used as a reference input for calibration tables.

TIA_THR_UP:

TIA_THR_UP is an interface variable. Its purpose is to link this specification to the “TIA_THR, TIA_IM, TIA_CYL calculation” specification. Indeed, TIA_THR_UP is used for TIA_THR and/or for TAM calculation.

TAM:

The ambient temperature TAM can be measured by a sensor (CONF_TAM = 2) **or** can be delivered via the CAN (CONF_TAM = 1) **or** determined from a model based on the temporary variable TAM_TMP. TAM_TMP is defined below.

Basically the algorithm of the model consists of two timers T_TAM1 and T_TAM2. Timer T_TAM1 represents the time the vehicle speed exceeds the threshold C_VS_TAM_MIN. When that conditon is fullfilled for a defined time (T_TAM1 > C_T_TAM1) a new TAM value is calculated based on TAM_TMP and a constant temperature offset C_TAM_OFS.

T_TAM2 is the timer that counts the time the vehicle speed condition is violated. It is permissible to violate the vehicle speed conditon for a certain time (T_TAM2 < C_T_TAM2) without a reset of timer T_TAM1.

The resulting TAM value is limited to a range between NC_TAM_CLC_BOL and NC_TAM_CLC_TOL.

LV_TAM_VLD_DIAGCP / T_TAM_VLD_DIAGCP:

To ensure accurate EVAP monitoring result, accurate TAM_MDL must be calculated. Flag LV_TAM_VLS_DIAGCP shall indicate that TAM_MDL was updated since engine start or within the last C_T_TAM_VLD_DIAGCP seconds. Flag LV_TAM_VLD_DIAGCP is reseted every C_T_TAM_VLD_DIAGCP seconds if no TAM_MDL take place (due to MAF_KGH condition to learn TAM_MDL, it can happen that TAM_MDL is not updated during long down-hill driving -> need to disable EVAP monitoring until next TAM_MDL update by reseting LV_TAM_VLD_DIAGCP).

TAM_ST:

Ambient temperature at start TAM_ST

Application Conditions:

Initialization: **At** Reset

- TIA_TMP:

NC_TIA_CONF	10	11, 22, 23, 30
TIA_TMP	TIA[NC_IDX_TIA_IM_CYL]	TIA[NC_IDX_TIA_AM_CHRG]

NC_TIA_CONF	13, 21	12, 24
TIA_TMP	TIA[NC_IDX_TIA_CHRG_THR]	TIA_CHRG_DOWN

- TIA_THR_UP and TAM_TMP:

TAM_TMP = TIA_THR_UP = TIA_TMP

- TAM:

If T_ES > C_T_ES_TEMP_MIN **and** LV_T_ES_NOT_PLAUS = 0
Then TAM = TIA_TMP
Else **If** there is no previously TAM value saved in the NVMY
Or the EEPROM does not work
Then TAM = 20.25°C
Else TAM = TAM saved in NVMY
Endif
TAM = Max[min(TAM; TIA_TMP), C_TAM_INI_MIN]
Endif

- TAM_ST = TAM
- TAM_MDL = TAM
- LV_TAM_VLD_DIAGCP = 0
- T_TAM_VLD_DIAGCP = 0

Recurrence: **see below**

Activation: At every Engine States

Deactivation: -

Formula section:

4.31.2 TIA_TMP definition:

Recurrence: **100 ms**

The aim of the table here below is to define the values of TIA_TMP depending on the Hardware Configuration (see NC_TIA_CONF definition in Chapter Sensor(s) Configuration).

NC_TIA_CONF	10	11, 22, 23, 30
TIA_TMP	TIA[NC_IDX_TIA_IM_CYL]	TIA[NC_IDX_TIA_AM_CHRG]

NC_TIA_CONF	13, 21	12, 24
TIA_TMP	TIA[NC_IDX_TIA_CHRG_THR]	TIA_CHRG_DOWN

eg: for NC_TIA_CONF = 23: TIA_TMP = TIA[NC_IDX_TIA_AM_CHRG]

4.31.3 TIA_THR_UP definition:

Recurrence: **100 ms**

In the same way, the table here below defines the value of TIA_THR_UP, depending on the configuration of the load sensor – either for MAF (CONF_MAF = 1) or MAP sensor (CONF_MAF = 0).

	CONF_MAF = 1	CONF_MAF = 0
TIA_THR_UP	TIA_TMP	TIA_THR

eg: for NC_TIA_CONF = 23: TIA_THR_UP = TIA_CHRG_DOWN

(note: TIA_THR_UP is not defined for NC_TIA_CONF = 10 as it is not used.)

4.31.4 TAM_TMP definition:

Recurrence: **1 s**

	CONF_TAM <> 0 **And** LV_ERR_TAM = 0 **And** LC_ERR_TAM_CAN_AS = 0	CONF_TAM = 0 **Or** LV_ERR_TAM = 1 **OR** LC_ERR_TAM_CAN_AS = 1	
NC_TIA_CONF	**-**	**10,** 12, 13, 21, 24	11, 22, 23, 30
TAM_TMP	(Not Calculated)	**TIA_CHRG_UP**	TIA[NC_IDX_TIA_AM_CHRG]

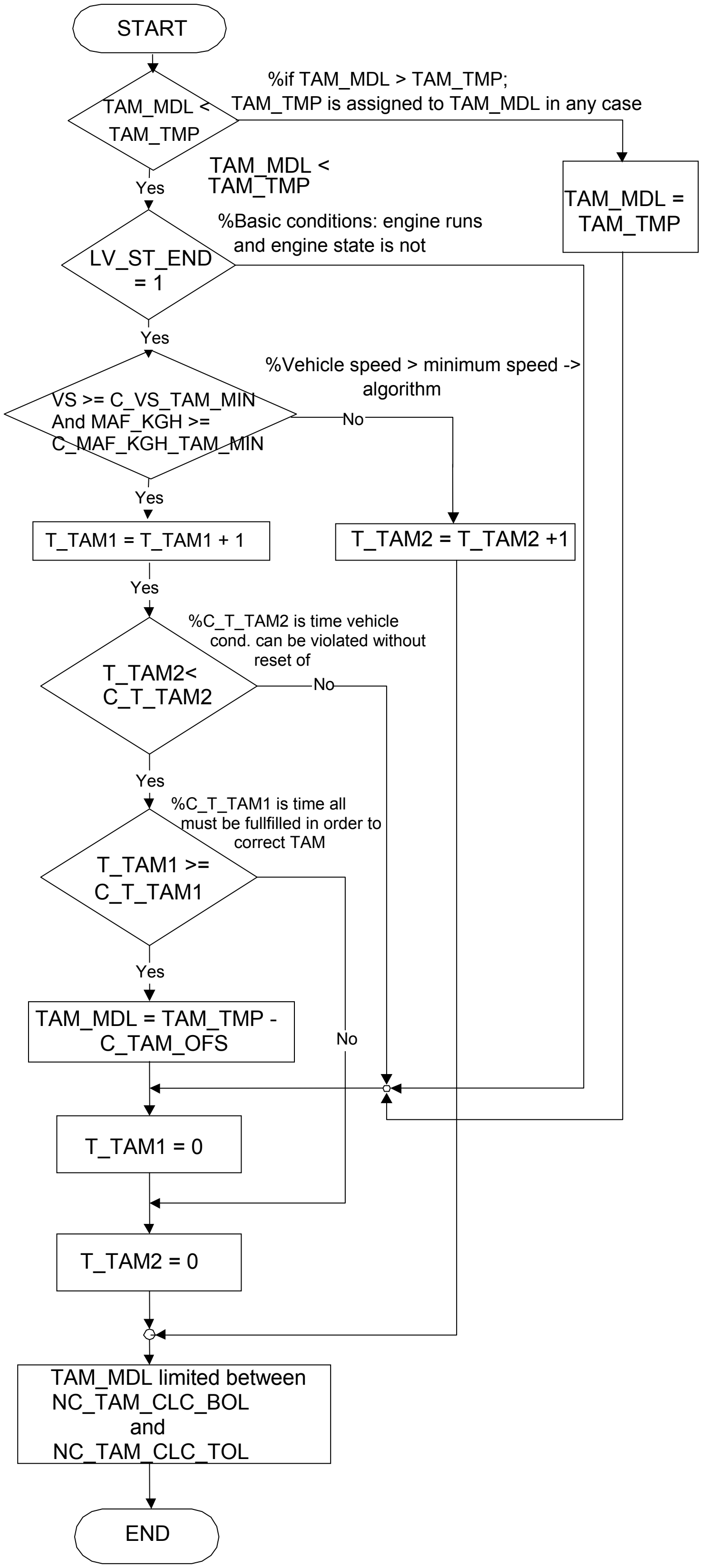
4.31.5 TAM_MDL,LV_TAM_VLD_DIAGCP and T_TAM_VLD_DIAGCP calculation:

Recurrence: **1 s**

```
If(1) TAM_MDL < TAM_TMP
Then(1)
   If(2) LV_ST_END = 1 (ie: Basic conditions: engine runs and engine state is not start)
        Then(2)
            If(3)   VS >= C_VS_TAM_MIN
                    And MAF_KGH >= C_MAF_KGH_TAM_MIN
            Then(3)
                    T_TAM1 = T_TAM1 + 1 (ie:1 s)
                    If(4) T_TAM2 < C_T_TAM2
                    (With C_T_TAM2 the time the vehicle speed condition can be violated
                    without a reset of timer T_TAM1)
                    Then(4)
                        If(5) T_TAM1 >= C_T_TAM1
                        With C_T_TAM1 the time all conditons have to be fullfilled in
                        order to enable a correction of TAM_MDL
                        Then(5)
                            TAM_MDL = TAM_TMP - C_TAM_OFS
                            T_TAM1 = 0
                            T_TAM2 = 0
                            LV_TAM_VLD_DIAGCP = 1
                            T_TAM_VLD_DIAGCP = C_T_TAM_VLD_DIAGCP
                        Else(5)
                            T_TAM2 = 0
                        Endif(5)
                    Else(4)
                        T_TAM1 = 0
                        T_TAM2 = 0
                    Endif(4)
            Else(3)
                    T_TAM2 = T_TAM2 + 1
            Endif(3)
        Else(2)
            T_TAM1 = 0
            T_TAM2 = 0
        Endif(2)
Else(1)
   TAM_MDL = TAM_TMP
   T_TAM1 = 0
   T_TAM2 = 0
   LV_TAM_VLD_DIAGCP = 1
   T_TAM_VLD_DIAGCP = C_T_TAM_VLD_DIAGCP
Endif(1)
```

If(1) T_TAM_VLD_DIAGCP = 0
Then(1) LV_TAM_VLD_DIAGCP = 0
Else(1) $T\_TAM\_VLD\_DIAGCP_{(n)} = T\_TAM\_VLD\_DIAGCP_{(n-1)} - 1$

TAM_MDL is limited between upper and lower limit:
NC_TAM_CLC_BOL <= TAM_MDL <= NC_TAM_CLC_TOL
Flow chart for the calculation of TAM_MDL (for 2nd case: ie: substitute formula):


START
TAM_MDL < TAM_TMP
%if TAM_MDL > TAM_TMP;
TAM_TMP is assigned to TAM_MDL in any case
TAM_MDL = TAM_TMP
Yes
TAM_MDL < TAM_TMP
%Basic conditions: engine runs and engine state is not
LV_ST_END = 1
Yes
%Vehicle speed > minimum speed -> algorithm
VS >= C_VS_TAM_MIN And MAF_KGH >= C_MAF_KGH_TAM_MIN
No
Yes
T_TAM1 = T_TAM1 + 1
T_TAM2 = T_TAM2 +1
Yes
%C_T_TAM2 is time vehicle cond. can be violated without reset of
T_TAM2< C_T_TAM2
No
Yes
%C_T_TAM1 is time all must be fullfilled in order to correct TAM
T_TAM1 >= C_T_TAM1
Yes
TAM_MDL = TAM_TMP - C_TAM_OFS
No
T_TAM1 = 0
T_TAM2 = 0
TAM_MDL limited between NC_TAM_CLC_BOL and NC_TAM_CLC_TOL
END

4.31.6 TAM calculation:

Recurrence: **1 s**

- There are 3 possibilities to calculate TAM: from the CAN, from an analog sensor value or from a model (then use of TAM sustitute formula)
- It is also possible to fix TAM by the mean of an application system calibration (if LC_TAM_AS=1).
- LC_ERR_TAM_CAN_AS can be set to 1 to easier software validation (otherwise, it must always be set to 0).

If LC_TAM_AS = 0

Then

If CONF_TAM = 0 *(no TAM sensor available or learnt)*

Or LV_ERR_TAM = 1

Or LC_ERR_TAM_CAN_AS = 1

Then TAM = TAM_MDL

Else

If ERR_SYM_ORGN_TAM <> 0

Or ERR_SYM_EL_TAM <> 0

Or ERR_SYM_TAM_CAN <> 0

Or ERR_SYM_PLAUS_TAM <> 0

Or (CONF_TAM = 1 **And** LV_TAM_CAN_FIRST_VLD = 0)

Then $TAM_{(n)} = TAM_{(n-1)}$

Else TAM = TAM_MES_MMV

Endif

Endif

Else TAM = C_TAM_AS

Endif

4.31.7 TAM_ST:

If STATE_ENG = "ES"
Then TAM_ST = TAM
Endif *(TAM_ST value remains unchaged out of ES)*

4.32 Key on recognition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_IGK	V/O	0...1H	0...1	1	[-]
Key on flag					

Input data:

LV_KEY_OFF	LV_MU_INH_PWL_TRAN_ES_EL		

FUNCTION DESCRIPTION:

General information:

LV_IGK is the boolean for ignition key status (on or off).

Application conditions:

Initialisation: **at reset**:

If LV_KEY_OFF = 1

Then init LV_IGK to 0 (off)

Else init LV_IGK to 1 (on)

Endif

Recurrence: 10 ms

Activation: at reset

Formula section:

If LV_KEY_OFF = 0

and LV_MU_INH_PWL_TRAN_ES_EL = 0

Then LV_IGK = 1 (on)

Else LV_IGK = 0 (off)

Endif

4.33 Coolant temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO	V/O	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature					
TCO_MES	V/O	0...FEH	-48...142.5	0.75	[°C]
Measured coolant temperature					
TCO_ST	V/O	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature at start					
TCO_STOP	V/O/S	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature at transition to engine stop (ES)					
LV_CDN_INI_TCO	V	0...1H	0...1	1	[-]
Boolean for coolant temperature initialisation conditions					
VP_TCO[NC_NR_TCO_SENS]	V/O	0...7FFFH	0...4.999847412	1.53E-04	[V]
Sensor raw acquisition of all available coolant temperature sensors					

Input data:

TCO_SUB	LV_ERR_TCO	LV_ERR_TCO_PREL	TIA_IM
LV_ERR_TIA_IM	LV_ES	NC_NR_TCO_SENS	T_ES

Import actions:

ACTION_INFR_GetVpTco (OUT <>)

FUNCTION DESCRIPTION:

General information:

The coolant temperature raw sensor voltage is measured with use of an A/D converter by the ECU hardware. The software then is generating a modified sensor signal (VP_TCO[0]), which is requested by the application software for further execution.

The modified sensor signal (VP_TCO[0]) is converted into a measured temperature value (TCO_MES) with use of a one-dimension interpolation table (IP_TCO_MES).

Application conditions:

See separate chapters:

4.33.1 Coolant temperature gradient monitoring

FUNCTION DESCRIPTION:

General information:

The measured coolant temperature (TCO_MES) is monitored for not plausible gradients. It is checked if the temperature difference between the old and the new measured coolant temperature (TCO_MES) exceeds the permissible gradient C_TCO_MES_GRD_MAX. In this case the old measured value remains unchanged. If the temperature difference between the measured coolant temperature values is exceeding the permissible gradient again during the next measurement, the new measured value is transferred into the working memory. A measured value can be inhibited only if the gradient exceeded for one time (recurrence).

The monitoring of the coolant temperature gradient does not lead to failure entries. The only purpose is to extract implausible measured values (noise).

Application conditions:

Initialisation at Reset:

FOR k = 0 to NC_NR_TCO_SENS -1
    ACTION_INFR_GetVpTco(VP_TCO[k])
END FOR

// all sensor values are imported and provided as output

$TCO\_MES_{(n)} = TCO\_MES_{(n-1)}$ = IP_TCO_MES

// acquisition of the main coolant temperature measurement signal

Recurrence: 100 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

// all sensor values are imported and provided as output

FOR k = 0 to NC_NR_TCO_SENS - 1

ACTION_INFR_GetVpTco(VP_TCO[k])

END FOR

// VP_TCO[0] is used as input signal for the interpolation table IP_TCO_MES

TCO_MES = IP_TCO_MES

// acquisition of the main coolant temperature measurement signal

If $| TCO\_MES_{(n-1)} - TCO\_MES_{(n)} |$ > C_TCO_MES_GRD_MAX

then $TCO\_MES_{(n)} = TCO\_MES_{(n-1)}$ (only one time!)

else $TCO\_MES_{(n)} = TCO\_MES_{(n)}$

endif

4.33.2 Coolant temperature acquisition

FUNCTION DESCRIPTION:

General information:

The actualisation of the coolant temperature value (TCO) is only executed, if no coolant temperature error / preliminary error is detected. In case of a detected error, the TCO value will be set to the calculated coolant temperature substitute value (TCO_SUB).

The initialisation of the coolant temperature value (TCO) is only performed once per engine run as long as the engine operating state “Engine stop” is active. If the engine stalls and in this case the engine operating state is changing to “Engine stop” again, a new initialization of the coolant temperature will not be done. Power latch must always be finished before a new initialization process is performed as long as no ECU reset occurs. An ECU reset is handled like the beginning of a new engine run and an initialization of the TCO value is also done in this case for at least one recurrence.

Application conditions:

Initialisation at Reset:

LV_CDN_INI_TCO = 1

TCO = TCO_MES

Initialisation at Engine stop to Engine run (ES_to_ERU):

LV_CDN_INI_TCO = 0

Activation: at every engine operating state

Deactivation: -

4.33.2.1 Coolant temperature acquisition (at ES after RESET)

Application conditions:

Recurrence: 10 ms

Activation: LV_CDN_INI_TCO = 1

Deactivation: LV_CDN_INI_TCO = 0

Formula section:

```
if (1)          LV_ERR_TCO = 0                          and
                LV_ERR_TCO_PREL = 0
then (1)        TCO = TCO_MES
else (1)        (TCO Sensor Error Case)
        if (2)  LV_ERR_TIA_IM = 0
        then (2) (reliable TIA signal available)
                if (3)      T_ES < IP_T_ES_MIN_TCO_SUB_INI
                then (3)    (short T_ES, no complete cooling of engine)
                            TCO = TCO_STOP – IP_OFS_TCO_SUB_INI
                else (3)    (long enough T_ES for complete cooling down)
                            TCO = IP_TCO_INI_TIA_IM
                endif (3)
        else (2) (no reliable TIA signal available)
                TCO = C_TCO_INI_TIA_IM
        endif (2)
endif (1)
```

4.33.2.2 Coolant temperature acquisition (at ERU / at ES after ERU_to_ES)

Recurrence: 100 ms

Activation: LV_CDN_INI_TCO = 0

Deactivation: LV_CDN_INI_TCO = 1

Formula section:

if (1) LV_ERR_TCO = 0

then (1)

  if (2) LV_ERR_TCO_PREL = 0

  then (2) TCO = TCO_MES

  else (2) $TCO_{(n)} = TCO_{(n-1)}$

  endif (2)

else (1) TCO = TCO_SUB

endif (1)

4.33.3 Coolant temperature at engine start

FUNCTION DESCRIPTION:

General information:

The coolant temperature at engine start (TCO_ST) is determined as long as the engine operating state “Engine stop” is detected. During this time, TCO_ST is set with the actual coolant temperature value TCO.

Application conditions:

Initialisation at Reset:

TCO_ST = TCO_MES

Recurrence: 10 ms

Activation: LV_ES = 1

Deactivation: LV_ES = 0

Formula section:

TCO_ST = TCO

4.33.4 Coolant Temperature at engine stop

FUNCTION DESCRIPTION:

General information:

The coolant temperature at engine stop (TCO_STOP) is determined at each transition to the engine operating state “Engine stop”. TCO_STOP will be stored in the non-volatile memory during the ECU’s self-holding phase.

Application conditions:

Initialisation at Reset:

TCO_STOP = TCO_STOP

(stored in the non-volatile memory)

Initialization at Engine run to Engine stop (ERU_to_ES):

TCO_STOP = TCO

Activation: at every engine operating state

Deactivation: -

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TCO_MES	16	0...FEH	-48...142.5	0.75	[°C]
LDP_VP_TCO_IP_TCO_MES	16	0...7FFFH	0...4.999847412	1.53E-04	[V]
Linearization of the coolant temperature sensor voltage (VP_TCO[0]) for temperature acquisition					
IP_TCO_INI_TIA_IM	6	0...FEH	-48...142.5	0.75	[°C]
LDP_TIA_IM_IP_TCO_INI_TIA_IM	6	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature initialization value in case of a coolant temperature failure					
C_TCO_INI_TIA_IM	1	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature initialization value in case of a coolant- and intake air- temperature failure					
C_TCO_MES_GRD_MAX	1	0...FEH	0...190.5	0.75	[°C]
Maximum permissible temperature gradient for coolant temperature gradient monitoring					
IP_T_ES_MIN_TCO_SUB_INI	6	0...FFFFH	0...65535	1	[min]
LDP_TCO_STOP_IP_T_ES_MIN_TCO	6	0...FEH	-48...142.5	0.75	[°C]
Engine Off Timer Threshold for consideration of TCO_STOP and T_ES for TCO initialization					
IP_OFS_TCO_SUB_INI	6	0...FEH	0...190.5	0.75	[°C]
LDP_T_ES_IP_OFS_TCO_SUB_INI	6	0...FFFFH	0...65535	1	[min]
Offset on TCO initialization value depending on T_ES					

4.34 Coolant temperature (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO_MIN_TOT_DC	V/O/S	0...FEH	-48...142.5	0.75	°C
Former / current driving cycle minimum TCO					
TCO_MAX_TOT_DC	V/O/S	0...FEH	-48...142.5	0.75	°C
Former / current driving cycle maximum TCO					

Input data:

TCO	LV_IGK	ERR_SYM_TCO_EL	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

TCO_MIN_TOT_DC = FEH (142.5°C)

TCO_MAX_TOT_DC = 0H (-48°C)

- otherwise: restored from non-volantile memory

Recurrence: 1s

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If ERR_SYM_TCO_EL = 0

Then If TCO < TCO_MIN_TOT_DC

Then TCO_MIN_TOT_DC = TCO

Endif

Endif

If ERR_SYM_TCO_EL = 0

Then If TCO > TCO_MAX_TOT_DC

Then TCO_MAX_TOT_DC = TCO

Endif

Endif

4.35 Coolant temperature substitute model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO_SUB	V/O	0...FE00H	-48...142.5	2.93E-03	[°C]
Coolant temperature substitute value					
TCO_SUB_INC	V	0...FE00H	-48...142.5	2.93E-03	[°C]
Coolant temperature substitute increment value					
FAC_TCO_SUB_COR	V	0...FFH	0...1.99218	0.0078125	[-]
Coolant temperature substitute value correction factor					
T_DLY_TCO_SUB_ST	V	0...FFFFH	0...65535	1	[s]
Delay time counter for coolant temperature substitute value calculation after engine start					
T_DLY_TCO_SUB	V	0...FFFFH	0...65535	1	[s]
Delay time counter for coolant temperature substitute value calculation during pull fuel cutoff or idle speed					
LV_CDN_ENA_TCO_SUB	V	0...1H	0...1	1	[-]
Boolean for coolant temperature substitute model initialisation condition					
TCO_SUB_INC_TMP	V	0...FE00H	0...190.5	2.93E-03	[°C]
Temporary Increment value for the coolant temperature substitute value calculation considering starting condition					
FAC_TCO_SUB_INC_TMP	V	0...FFH	0...0.99609	3.91E-03	[-]
Temporary Increment factor for the coolant temperature substitute value calculation considering starting condition					
TCO_SUB_DEC_TMP	V	0...FE00H	0...190.5	2.93E-03	[°C]
Temporary Decrement value for the coolant temperature substitute value calculation considering starting condition					
LV_TCO_SUB_INC_FAST	V	0...1H	0...1	1	-
Flag for fast TCO_SUB increment according to engine starting condition (1: "FAST" , 0 : "SLOW")					

Input data:

LV_ST	LV_IS	LV_PL	LV_PUC
LV_PU	TCO	TCO_ST	LV_ERR_TCO
TAM	MAF_KGH	TIA_ST	T_AST
LV_ERR_TCO_STUCK_RNG	IP_TCO_INI_TIA_IM	TIA_IM	

FUNCTION DESCRIPTION:

General information:

The coolant temperature substitute model (TCO_SUB) is initialized with the coolant temperature at engine start (TCO_ST) and an offset value (IP_TCO_SUB_INI_OFS) as soon as the system event “Engine stop” to “Engine run” is detected. The substitute model increment values are dependent on the actual engine performance (equivalent to mass air flow MAF_KGH). To take the smaller incrementation values at higher coolant temperatures into consideration, the increment values are weighted. TCO_SUB calculation shall be divided into two modes (fast/slow) depending on coolant temperature and intake air temperature.

During “Idle speed” and “Pull fuel cutoff” phases the calculation of the coolant temperature substitute value remains activated.

A delay time after engine start to activate the incrementation of the temperature substitute value is incremented, to simulate the warm up of the real coolant temperature very efficient.

In case of a coolant temperature acquisition failure (LV_ERR_TCO = 1), the coolant temperature substitute value is reinitialized with the latest valid coolant temperature value. In this case the model incrementation starts from the initialization value on.

Application conditions:

Initialisation at Engine stop to Engine run (ES_to_ERU):

T_DLY_TCO_SUB_ST = 0

T_DLY_TCO_SUB = 0

$TCO\_SUB_{(n)} = TCO\_SUB_{(n-1)}$ = TCO_ST + IP_TCO_SUB_INI_OFS

LV_CDN_ENA_TCO_SUB = 1

Initialisation at Exit start (EXIT_ST):

If TIA_ST > C_TCO_ST_MIN_TCO_SUB

And abs(TCO_ST - TIA_ST) < C_DIF_TCO_TIA_ST_MIN_TCO_SUB

Then LV_TCO_SUB_INC_FAST = 1

Else LV_TCO_SUB_INC_FAST = 0

Initialisation at Engine run to Engine stop (ERU_to_ES):

LV_CDN_ENA_TCO_SUB = 0

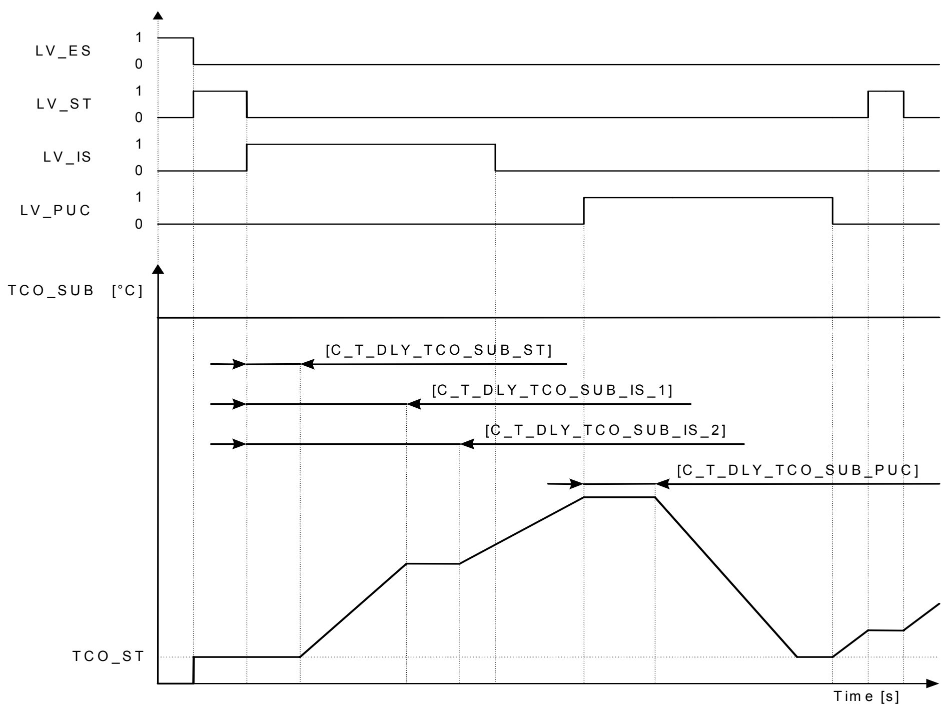
Recurrence: 1000 ms

(100ms for TCO sub re-initialisastion at TCO error detection)

Activation: LV_CDN_ENA_TCO_SUB = 1

Deactivation: LV_CDN_ENA_TCO_SUB = 0

Signal flow diagram:



Formula section:

Calculation of the substitute model increment value during “Engine start”:

If LV_ST = 1

then TCO_SUB_INC = 0

Switching of LV_TCO_SUB_INC_FAST:

If LV_TCO_SUB_INC_FAST = 1 // Fast TCO_SUB increment

Then **If** TCO_SUB > C_MAX_TCO_SUB_FAST_INC

Then LV_TCO_SUB_INC_FAST = 0 // Slow TCO_SUB increment

∗ Remark: LV_TCO_SUB_INC_FAST may not be switched from "FAST" to "SLOW" by setting of C_MAX_TCO_SUB_FAST_INC to be maximum value.

Determination of temporary model inc./dec. value and factor according to starting condition:

If LV_TCO_SUB_INC_FAST = 1

Then TCO_SUB_INC_TMP = IP_TCO_SUB_INC_FAST (MAF_KGH)

FAC_TCO_SUB_INC_TMP = IP_FAC_TCO_SUB_INC_FAST (MAF_KGH ,TCO_SUB)

TCO_SUB_DEC_TMP = IP_TCO_SUB_DEC_FAST (MAF_KGH)

Else TCO_SUB_INC_TMP = IP_TCO_SUB_INC_SLOW (MAF_KGH)

FAC_TCO_SUB_INC_TMP = IP_FAC_TCO_SUB_INC_SLOW (MAF_KGH,TCO_SUB)

TCO_SUB_DEC_TMP = IP_TCO_SUB_DEC_SLOW (MAF_KGH)

Calculation of the substitute model increment value during “PL” or “PU”:

If LV_PL = 1 **or** LV_PU = 1

then TCO_SUB_INC = TCO_SUB_INC_TMP

* FAC_TCO_SUB_INC_TMP

* FAC_TCO_SUB_COR

Calculation of the substitute model increment value during “PUC”:

If Transition: LV_PUC = 0 -> LV_PUC = 1

then T_DLY_TCO_SUB = 0

if LV_PUC = 1

then **if** T_DLY_TCO_SUB < C_T_DLY_TCO_SUB_PUC

then TCO_SUB_INC = 0

else TCO_SUB_INC = -**(** TCO_SUB_DEC_TMP **)**

***** FAC_TCO_SUB_INC_TMP

T_DLY_TCO_SUB = T_DLY_TCO_SUB + 1 s

Calculation of the substitute model increment value during "Idle speed":

```
If      Transition:    LV_IS = 0 -> LV_IS = 1
then    T_DLY_TCO_SUB = 0
if      LV_IS = 1
then    if      T_DLY_TCO_SUB < C_T_DLY_TCO_SUB_IS_1
          or    T_DLY_TCO_SUB > C_T_DLY_TCO_SUB_IS_2
        then    TCO_SUB_INC  =    TCO_SUB_INC_TMP
                                  * FAC_TCO_SUB_INC_TMP
                                  * FAC_TCO_SUB_COR
        else    TCO_SUB_INC  = 0
        T_DLY_TCO_SUB = T_DLY_TCO_SUB + 1 s
```

Calculation of the coolant temperature substitute model value:

```
If      LV_ERR_TCO = 0
then    FAC_TCO_SUB_COR = 1
else    FAC_TCO_SUB_COR = C_FAC_TCO_SUB_COR
endif

If      T_DLY_TCO_SUB_ST < C_T_DLY_TCO_SUB_ST
then    if      Transition: LV_ERR_TCO_STUCK_RNG  = 0 --> 1
        then  TCO_SUB = IP_TCO_INI_TIA_IM
        else   TCO_SUB(n) = TCO_SUB(n-1)
                T_DLY_TCO_SUB_ST = T_DLY_TCO_SUB_ST + 1 s
        endif
else    if      Transition: LV_ERR_TCO = 0 --> 1
        then  TCO_SUB(n) = TCO(n-1) (re-initialization of TCO_SUB with last plausible TCO value)
                (This is done as 100msec task)
        endif
        TCO_SUB(n) = TCO_SUB(n-1) + TCO_SUB_INC

endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TCO_SUB_INC_FAST	1*6	0...FE00H	0...190.5	2.93E-03	[°C]
LDP_MAF_KGH_IP_TCO_SUB_INC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Fast Increment value for the coolant temperature substitute value calculation					
IP_TCO_SUB_INC_SLOW	1*6	0...FE00H	0...190.5	2.93E-03	[°C]
LDP_MAF_KGH_IP_TCO_SUB_INC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Slow Increment value for the coolant temperature substitute value calculation					
IP_FAC_TCO_SUB_INC_FAST	6*6	0...FFH	0...0.99609	3.91E-03	[-]
LDP_MAF_KGH_IP_FAC_TCO_SUB_INC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDP_TCO_SUB_IP_FAC_TCO_SUB_INC	6	0...FE00H	-48...142.5	2.93E-03	[°C]
Fast Increment factor for the coolant temperature substitute value calculation					
IP_FAC_TCO_SUB_INC_SLOW	6*6	0...FFH	0...0.99609	3.91E-03	[-]
LDP_MAF_KGH_IP_FAC_TCO_SUB_INC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDP_TCO_SUB_IP_FAC_TCO_SUB_INC	6	0...FE00H	-48...142.5	2.93E-03	[°C]
Slow Increment factor for the coolant temperature substitute value calculation					
IP_TCO_SUB_DEC_FAST	1*6	0...FE00H	0...190.5	2.93E-03	[°C]
LDP_MAF_KGH_IP_TCO_SUB_DEC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Fast Decrement value for the coolant temperature substitute value calculation					
IP_TCO_SUB_DEC_SLOW	1*6	0...FE00H	0...190.5	2.93E-03	[°C]
LDP_MAF_KGH_IP_TCO_SUB_DEC	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Slow Decrement value for the coolant temperature substitute value calculation					
IP_TCO_SUB_INI_OFS	2*2	0...FE00H	-48...142.5	2.93E-03	[°C]
LDP_TCO_ST_IP_TCO_SUB_INI_OFS	2	0...FEH	-48...142.5	0.75	[°C]
LDP_TAM_IP_TCO_SUB_INI_OFS	2	0...FEH	-48...142.5	0.75	[°C]
Initialization offset value for the coolant temperature substitute value calculation					
C_T_DLY_TCO_SUB_ST	1	0...FFFFH	0...65535	1	[s]
Delay time after start to activate the coolant temperature substitute value calculation					
C_T_DLY_TCO_SUB_PUC	1	0...FFFFH	0...65535	1	[s]
Delay time after pull fuel cutoff to activate the coolant temperature substitute value calculation					
C_T_DLY_TCO_SUB_IS_1	1	0...FFFFH	0...65535	1	[s]
Delay time after idle speed to stop the coolant temperature substitute value calculation					
C_T_DLY_TCO_SUB_IS_2	1	0...FFFFH	0...65535	1	[s]
Delay time after idle speed to activate the coolant temperature substitute value calculation					
C_FAC_TCO_SUB_COR	1	0...FFH	0...1.99218	0.0078125	[-]
Multiple Coolant temperature substitute value correction factor					
C_TCO_ST_MIN_TCO_SUB	1	0...FEH	-48...142.5	0.75	[°C]
Minimum TCO_ST for fast TCO_SUB calculation					
C_DIF_TCO_TIA_ST_MIN_TCO_SUB	1	0...FEH	0...190.5	0.75	[°C]
Constant to distinguish partial cool-down for fast TCO_SUB calculation					
C_MAX_TCO_SUB_FAST_INC	1	0...FEH	-48...142.5	0.75	[°C]
Maximum TCO_SUB to activate fast TCO_SUB increment					

4.36 AIRT Integration module to HMC Theta Project

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TAM_MES	V/O	0H...FEH	-48...142.5	0.75	°C
Ambient air temperature, raw measured value					
TAM_MES_MMV	V/O	0H...FEH	-48...142.5	0.75	°C
Ambient air temperature, filtered value					
TIA	V/O	0H...FEH	-48...142.5	0.75	°C
Air intake temperature					
TIA_ST	V/O	0H...FEH	-48...142.5	0.75	°C
Air intake temperature at start					
LV_ERR_TIA	V/O	0H ... 1H	0 ...1	1	-
Boolean for Electrical error currently present on TIA sensor (after debounce)					
LV_END_DIAG_TIA	V/O	0...1H	0...1	1	-
Result of TIA electrical diagnosis					

Input data:

LV_IGK	TIA_TMP	LV_ES	NC_IDX_TIA_AM_THR
TIA_IM	TIA_THR	LV_ERR_TIA_IM	LV_END_DIAG_EL_TIA[NC_IDX_TIA_AM_THR]
VP_TAM	TAM_CAN	CONF_TAM	CONF_MAF
LV_ERR_TIA_THR	VS	MAF_CYL	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CRLC_TAM_MES	4x4	0...FFH	0...0.99609307	0.003906251	-
LDP_VS_IP_CRLC_TAM_MES	4	0...FFH	0...255	1	km/h
LDP_MAF_CYL_IP_CRLC_TAM_MES	4	0...FFFFH	0...2047.969	0.03125	kg/h
Map for TAM_MMV filtering factor depending on the actual vehicle speed and air mass flow					
IP_TAM_MES	16	0...FEH	-48...142.5	0.75	°C
LDPM_VP_TAM_IP_TAM_MES	16	0...7FFFH	0...4.999847	1.52588E-4	V
Ambient temperature linear table					

4.36.1 Definition of TIA

FUNCTION DESCRIPTION:

General information:

Basically the HMC Theta projects uses the TIA sensor located in the manifold pressure sensor (named “TIA_IM_CYL”). However it is possible, for vehicle types equipped with a mass air flow sensor (MAF), to switch to a TIA sensor position located at the MAF (upstream the throttle body). By CONF_MAF the system switches between MAF position (CONF_MAF = 1) and MAP position (CONF_MAF = 0).

Application conditions:

Initialisation: at reset or transition LV_IGK = 0 → 1

TIA = TIA_TMP

Recurrence: 100 ms (note: TIA_IM is calculated every 100 ms)

Activation: every engine state

Deactivation:

Formula section:

If CONF_MAF = 0 *(system with MAP sensor - TIA sensor located at MAP sensor)*

Then TIA = TIA_IM

Elseif CONF_MAF = 1 *(system with MAF sensor - TIA sensor located at MAF sensor)*

Then TIA = TIA_THR

4.36.2 Definition of TIA_ST

FUNCTION DESCRIPTION:

General information:

AIRT produces TIA_TMP as reference for intake air temperature from sensor. Also TIA_ST will be defined equal to TIA_TMP.

Application conditions:

Initialisation: at reset and LV_IGK = 0→1

TIA_ST = TIA_TMP

Recurrence: 100 ms (note: TIA_TMP is calculated every 100 ms)

Activation: every engine state

Deactivation:

Formula section:

TIA_ST = TIA_TMPevery 100 ms as long as LV_ES = 1.

4.36.3 Definition of LV_ERR_TIA

FUNCTION DESCRIPTION:

General information:

LV_ERR_TIA is defined as the currently general error in HMC Theta project. LV_ERR_TIA will be defined equal to LV_ERR_TIA_IM (for systems equipped with a MAP sensor – combined TMAP sensor, located in the intake manifold) or equal to LV_ERR_TIA_THR (for systems equipped with a MAF sensor – temperature value coming from MAF sensor, located upstream the throttle).

Application conditions:

Initialisation:

Recurrence: 100 ms

(note: LV_ERR_TIA_IM is calculated every 100 ms)

Activation: *every engine state*

Deactivation:

Formula section:

If CONF_MAF = 0 *(system with MAP sensor - TIA sensor located in the intake manifold)*

Then LV_ERR_TIA = LV_ERR_TIA_IM

Elseif CONF_MAF = 1 *(system with MAF sensor - TIA sensor located upstream the throttle)*

Then LV_ERR_TIA = LV_ERR_TIA_THR

4.36.4 Definition of LV_END_DIAG_TIA

FUNCTION DESCRIPTION:

General information:

LV_END_DIAG_TIA is defined equal to LV_END_DIAG_EL_TIA[NC_IDX_TIA_IM_CYL].

Application conditions:

Initialisation:

Recurrence: 1000 ms

(note: LV_END_DIAG_EL_TIA[NC_IDX_TIA_IM_CYL] is calculated every 100 ms)

Activation: every engine state

Deactivation:

Formula section:

LV_END_DIAG_TIA = LV_END_DIAG_EL_TIA[NC_IDX_TIA_IM_CYL]

4.36.5 Calculation of TAM_MES and TAM_MES_MMV

FUNCTION DESCRIPTION:

General information:

For turbocharged (TCI) version the ambient temperature (TAM) is measured by an analog temperature sensor or is available via CAN message.

Application conditions:

Initialisation: at reset TAM_MES = TAM_MES_MMV = TAM
(Important: TAM has to be initialised previously!)

Recurrence: 100 ms

Activation: *every engine state*

Formula section:

If CONF_TAM = 0 *(no TAM sensor available or learned)*

Then TAM_MES = -48°C

Elseif CONF_TAM = 1 *(TAM value via CAN)*

Then TAM_MES = TAM_CAN

Else *(CONF_TAM = 2 TAM value via analog sensor)*

TAM_MES = IP_TAM_MES (VP_TAM)

End

If $TAM\_MES_{(n)} <= TAM\_MES\_MMV_{(n-1)}$

Then TAM_MES_MMV = TAM_MES

Else

$$TAM\_MES\_MMV_{(n)} = TAM\_MES\_MMV_{(n-1)} + (TAM\_MES - TAM\_MES\_MMV_{(n-1)}) * IP\_CRLC\_TAM\_MES(VS, MAF\_CYL)$$

Endif

4.37 Calculation of the basic reduced area

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AR_RED_BAS	O/V	0...FFFFH	0...58.592855	8.9407E-4	cm²
Basic reduced opening area					

Input data:

FAC_AR_RED_MAF	TPS_SEG	LV_AT	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_AR_RED_THR_AT	16	0...FFFFH	0...58.592855	8.9407E-4	cm²
LDPM_TPS_SEG_IP_AR_RED_THR	16	0...3FFFH	0...119.5	0.00729415	°TPS
Reduced Area at the throttle for A/T					
IP_AR_RED_THR_MT	16	0...FFFFH	0...58.592855	8.9407E-4	cm²
LDPM_TPS_SEG_IP_AR_RED_THR	16	0...3FFFH	0...119.5	0.00729415	°TPS
Reduced Area at the throttle for M/T					

In this the effective opening area for the air – mass flow into the take manifold is calculated. The reduced opening area of the throttle depend on its actual opening angle (segment synchronous TPS_SEG)

Application Condition

Function Description

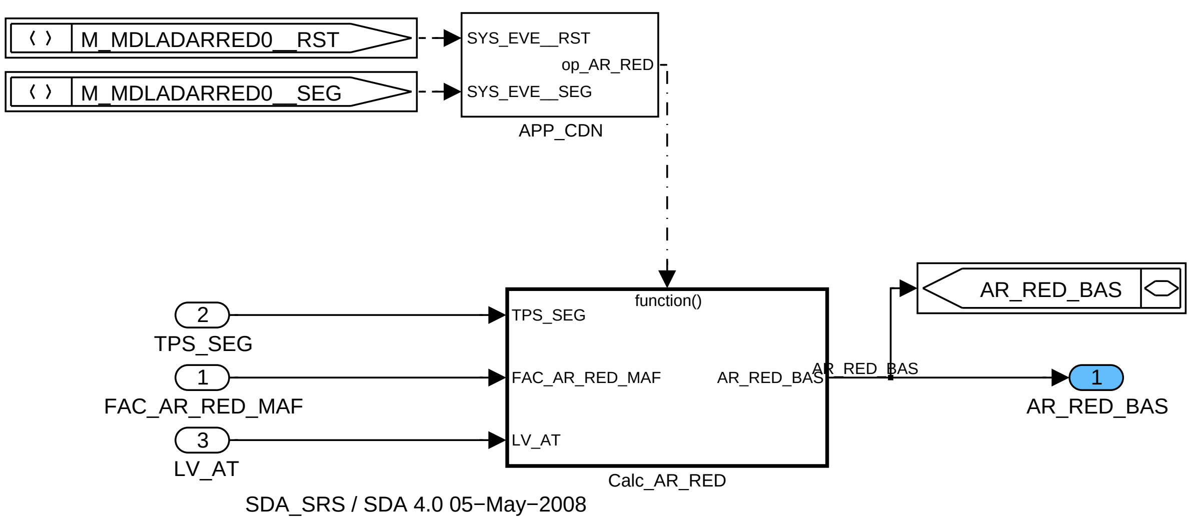


Figure 336 INSY_MDLADARRED0

4.37.1.1 SUBFUNCTION: Calc_AR_RED

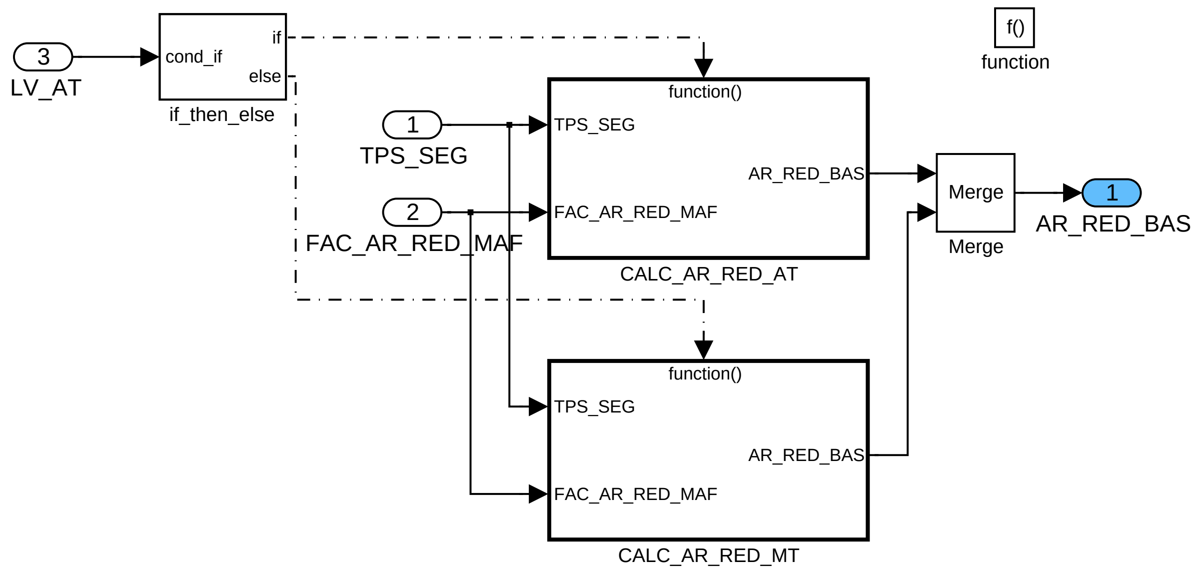


Figure 337 INSY_MDLADARRED0/ Calc_AR_RED

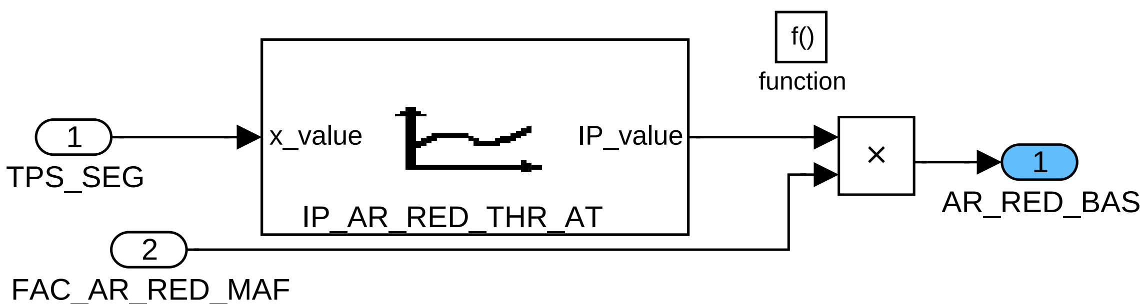


Figure 338 INSY_MDLADARRED0/ Calc_AR_RED/ CALC_AR_RED_AT

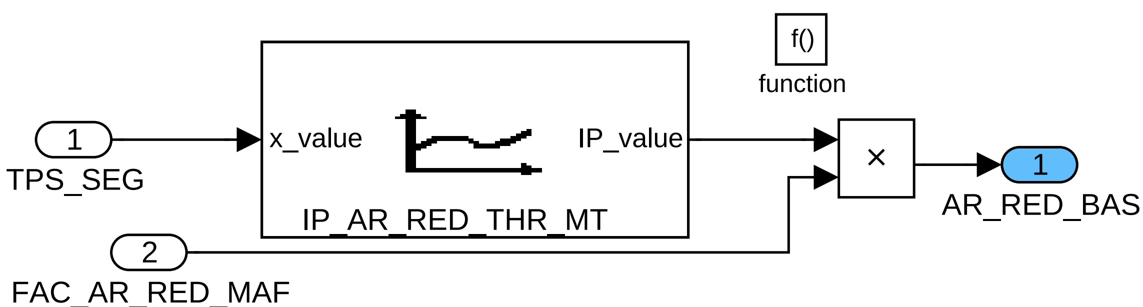


Figure 339 INSY_MDLADARRED0/ Calc_AR_RED/ CALC_AR_RED_MT

4.38 AIRT Air Temperature Sensor(s) Configuration:

4.38.1 General:

The configuration of the hardware definition is done with the Non-Calibratable constants NC_TIA_CONF which is here also used as Compiler Switch.

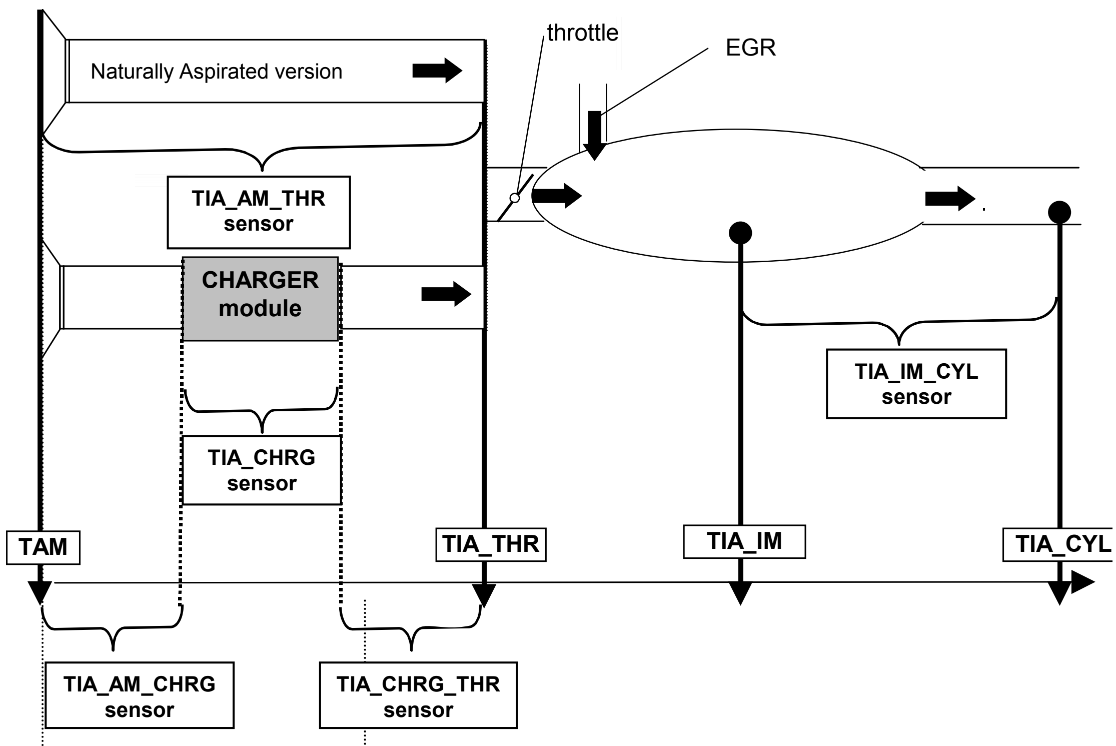
Hypothesis: At least one TIA system sensor is used in the complete intake system (ie: AIRT + CHRG Aggregates).

4.38.2 Air Temperature Sensor Configuration Variables:

FUNCTION DESCRIPTION:

General information:

Definition of the possible positions for the TIA sensor(s):
The drawing above describes the 5 **possible** positions inside the Intake System for the TIA sensors:



- *Whatever version (CHRG or No CHRG):*

TIA_IM_CYL sensor is located:
. After the Throttle in case of no EGR version
. After the EGR Inlet in case of EGR version

- *In case of No CHRG version **only**:*

TIA_AM_THR sensor is located before the Throttle (ie: located in the Intake System only; TIA_AM_THR cannot be an Ambiant temperature sensor)

- *In case of CHRG version **only**:*

_ TIA_AM_CHRG sensor is located in the intake system before the Charger Module **or** can be an Ambiant Temperature sensor.
_ TIA_CHRG_THR sensor is located between the Charger Module and the Throttle

remark: TIA_CHRG sensor, located inside the CHRG Module, is defined in the CHRG Aggregate.

Application conditions:

Activation: at all Engine States

Deactivation: -

Initialization: **at** Reset / initialized with the values of the table below

Recurrence: **no** (the outputs are just calculated once)

Formula section:

4.38.2.1 NC_TIA_CONF definition:

Depending on the number of TIA sensors and of their location(s), NC_TIA_CONF has to be set with the value given in the Table below.

NC_TIA_CONF defines the TIA sensor(s) Intake System hardware

Engine version	Possible Sensors	1					2					3
Naturally aspirated only	TIA_AM_THR sensor	-	-	-	-	Av.	Av.	-	-	-	-	-
Charged only	TIA_AM_CHRG sensor	-	Av.	-	-	-	-	-	Av.	Av.	-	Av.
Charged only	TIA_CHRG_THR sensor	-	-	-	Av.	-	-	Av.	Av.	-	-	Av.
Naturally asp. or Charged	TIA_IM_CYL sensor	Av.	-	-	-	-	Av.	Av.	-	Av.	Av.	Av.
Charged only	*TIA_CHRG sensor*	-	-	Av.	-	-	-	-	-	-	Av.	-
	NC_TIA_CONF	**10**	**11**	**12**	**13**	**14**	**20**	**21**	**22**	**23**	**24**	**30**
	NC_SENS_NR_TIA	**1**	**1**	**0 (*)**	**1**	**1**	**2**	**2**	**2**	**2**	**1 (*)**	**3**
	NC_IDX_TIA_AM_THR	0	0	0	0	**1**	**1**	0	0	0	0	0
	NC_IDX_TIA_AM_CHRG	0	**1**	0	0	0	0	0	**1**	**1**	0	**1**
	NC_IDX_TIA_CHRG_THR	0	0	0	**1**	0	0	**1**	**2**	0	0	**2**
	NC_IDX_TIA_IM_CYL	**1**	0	0	0	0	**2**	**2**	0	**2**	**1**	**3**

eg: if the 2 sensors TIA_AM_THR and TIA_IM_CYL are available (notation “Av.” In the Table for “Available”) then NC_TIA_CONF has to be set to = 20

Remark1:
the first digit of NC_TIA_CONF fixes the number of sensor ie:
. definition with **1** sensor for **1**0 ≤ NC_TIA_CONF ≤ **1**4
. definition with **2** sensor for **2**0 ≤ NC_TIA_CONF ≤ **2**4
. definition with **3** sensor for NC_TIA_CONF = **3**0
Remark2:
NC_TIA_CONF is the only value which has to be set by the Application Engineer.

4.38.2.2 NC_SENS_NR_TIA Definition:

See same Table above.

From NC_TIA_CONF is automatically defined the number of sensors used **for the AIRT. The eventual sensor from the Charger Module is excluded (*).**

eg: . for NC_TIA_CONF = 20 then NC_SENS_NR_TIA = 2
. for NC_TIA_CONF = 24 then NC_SENS_NR_TIA = 1

4.38.2.3 NC_IDX_TIA_[AM_THR ,IM_CYL, AM_CHRG, CHRG_THR] definitions:

Explanation with an example with the variable TIA_MES and for NC_TIA_CONF = 20:

_ On the Application Tool:

The value of TIA_MES:

- for the sensor TIA_AM_THR will be displayed as: TIA_MES[NC_IDX_TIA_AM_THR]
- for the sensor TIA_IM_CYL will be displayed as: TIA_MES[NC_IDX_TIA_IM_CYL]

_ For software coding:

It is necessary to define the correspondance between the Index (NC_IDX) and the physical position of the sensor (function of NC_TIA_CONF). A value from 0 to 3 is allocated to each index (See same Table above).

- NC_IDX_TIA_AM_THR = 1
- NC_IDX_TIA_IM_CYL = 2

which means:

- TIA_MES[1] corresponds to the temperature value from the sensor TIA_AM_THR
- TIA_MES[2] corresponds to the temperature value from the sensor TIA_IM_CYL

in the same way for every variable “VARIABLE”:

- VARIABLE[1] is relative to sensor TIA_AM_THR
- VARIABLE[2] is relative to sensor TIA_IM_CYL

Remark: When a sensor is not available then the corresponding index value is set to 0.
Eg: for NC_TIA_CONF = 20 then NC_IDX_TIA_AM_CHRG = 0

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_TIA_CONF	1	0...28H	0...40	1	[-]
TIA sensor(s) intake system hardware definition (compile switch)					
NC_SENS_NR_TIA	1	0...3H	0...3	1	[-]
Number of TIA sensors for AIRT Aggregate only (ie: excluding CHRG Aggregate)					
NC_IDX_TIA_AM_THR	1	0...3H	0...3	1	[-]
Index value corresponding to TIA_AM_THR temperature					
NC_IDX_TIA_AM_CHRG	1	0...3H	0...3	1	[-]
Index value corresponding to TIA_AM_CHRG temperature					
NC_IDX_TIA_CHRG_THR	1	0...3H	0...3	1	[-]
Index value corresponding to TIA_CHRG_THR temperature					
NC_IDX_TIA_IM_CYL	1	0...3H	0...3	1	[-]
Index value corresponding to TIA_IM_CYL temperature					

Important:

- NC_TIA_CONF has to be set by the Project

- NC_SENS_NR_TIA, NC_IDX_TIA_[AM_THR, AM_CHRG, CHRG_THR, IM_CYL] values are automatically configurated from NC_TIA_CONF (ie: no manual intervention from the Project is needed).

4.38.3 TAM via CAN Configuration: NC_TAM_CAN_USE

FUNCTION DESCRIPTION:

General information:

If NC_TAM_CAN_USE is set to 1 then the value delivered via the TAM_CAN Line is assigned to TAM. In case of an error on the CAN Line a model is used.
Else (ie: NC_TAM_CAN_USE = 0) then TAM will be calculated from a model based on a TIA sensor value.

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_TAM_CAN_USE	1	0...1H	0...1	1	[-]
TAM is computed from to the value delivered by the TAM_CAN Line: 1 = Yes / 0 = No					

4.39 TIA_THR, TIA_IM and TIA_CYL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_CYL	O/V	0...FEH	-48...142.5	0.75	°C
Intake air temperature at cylinder inlet					
TIA_IM	O/V	0...FEH	-48...142.5	0.75	°C
Air temperature in the intake manifold					
TIA_THR	O/V	0...FEH	-48...142.5	0.75	°C
Air temperature at the throttle body					
TIA_CYL_BAS	V	0...FEH	-48...142.5	0.75	°C
Intermediate value of Air temperature used for TIA_CYL calculation					
TIA_IM_BAS	V	0...FEH	-48...142.5	0.75	°C
Intermediate value of Air temperature used for TIA_IM calculation					
TIA_THR_BAS	V	0...FEH	-48...142.5	0.75	°C
Intermediate value of Air temperature used for TIA_THR calculation					

Input data:

MAF_THR	N_32	TCO	TIA_TMP
TIA_THR_UP	TIA[NC_SENS_NR_TIA]	CONF_MAF	NC_SENS_NR_TIA
NC_IDX_TIA_IM_CYL	TOIL		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_TIA_CYL	1	0...FFH	0...0.99609307	0.00390625	-
TIA_CYL mean value filtering factor: Typical value: 0.996					
C_CRLC_TIA_IM	1	0...FFH	0...0.99609307	0.00390625	-
TIA_IM mean value filtering factor: Typical value: 0.996					
C_CRLC_TIA_THR	1	0...FFH	0...0.99609307	0.00390625	-
TIA_THR mean value filtering factor: Typical value: 0.996					
C_TIA_CYL_AS	1	0...FEH	-48...142.5	0.75	°C
Application system calibration for TIA_CYL					
C_TIA_IM_AS	1	0...FEH	-48...142.5	0.75	°C
Application system calibration for TIA_IM					
C_TIA_THR_AS	1	0...FEH	-48...142.5	0.75	°C
Application system calibration for TIA_THR					
LC_TIA_CYL_AS	1	0...1H	0...1	1	-
Application system calibration switch for TIA_CYL: 1 = use of application system value C_TIA_CYL_AS					
LC_TIA_IM_AS	1	0...1H	0...1	1	-
Application system calibration switch for TIA_IM: 1 = use of application system value C_TIA_IM_AS					
LC_TIA_THR_AS	1	0...1H	0...1	1	-
Application system calibration switch for TIA_THR: 1 = use of application system value C_TIA_THR_AS					
IP_FAC_TIA_CYL	8x8	0...FFH	0...3.984375	0.015625	-
LDPM_TIA_TMP_1_AIRT	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_1_AIRT	8	0...FEH	-48...142.5	0.75	°C
Correction factor mapping for TIA_CYL calculation. Typical value: 1					
IP_FAC_TIA_IM	8x8	0...FFH	0...3.984375	0.015625	-
LDPM_TIA_TMP_1_AIRT	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_1_AIRT	8	0...FEH	-48...142.5	0.75	°C
Correction factor mapping for TIA_IM calculation. Typical value: 1					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TIA_THR	8x8	0...FFH	0...3.984375	0.015625	-
LDPM_TIA_TMP_1_AIRT	8	0...FEH	-48...142.5	0.75	°C
LDP_TOIL_1_AIRT	8	0...C8H	-40...160	1	°C
Correction factor mapping for TIA_THR calculation. Typical value: 1					
IP_TIA_CYL	8x8	0...FFH	-96...95.25	0.75	°C
LDPM_MAF_THR_1_AIRT	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
LDPM_N_32_1_AIRT	8	0...FFH	0...8.16E+3	32	rpm
basic mapping for TIA_CYL calculation. Typical value: 0					
IP_TIA_IM	8x8	0...FFH	-96...95.25	0.75	°C
LDPM_MAF_THR_1_AIRT	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
LDPM_N_32_1_AIRT	8	0...FFH	0...8.16E+3	32	rpm
Basic mapping for TIA_IM calculation. Typical value: 0					
IP_TIA_THR	8x8	0...FFH	-96...95.25	0.75	°C
LDPM_MAF_THR_1_AIRT	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
LDPM_N_32_1_AIRT	8	0...FFH	0...8.16E+3	32	rpm
basic mapping for TIA_THR calculation. Typical value: 0					

4.39.1 General :

Version for NC_EGR_CONF = 0 and for systems with different TIA sensor locations

Objective is to estimate the air temperature at the throttle (TIA_THR), in the manifold (TIA_IM) and before the inlet valve (TIA_CYL) taking into account the different hardware configuration – depending on the load sensor (in case of MAP sensor it is located in the intake manifold, in case of MAF sensor the location is upstream the throttle).

Model is based on the following hypothesis:

The flow from the Canister Purge is neglected: also, MAF_THR will be used (and not MAF_CYL).

To estimate the air temperature, we use the following law :

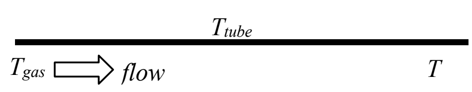
The Reynolds law:

It describes the heat transfers of a gas flowing into a tube. For gases at moderate temperature, the change of temperature between the final gas temperature and the initial one follows the law:

$$T - T_{gas} = \frac{K(L,D)\left[T_{tube} - T_{gas}\right]}{flow^{0.2}}$$

Where: K depends on the length of the tube (L) and its diameter (D).

flow is the mass flow of gas.



It is very difficult to define properly the temperature of the tube for all the configurations but we can approach this law using two mappings. One is depending on MAF_THR and N_32, the second one on TIA_TMP and TCO.

Air flow transients management:

The temperature dynamic is lower than the flow one (MAF_THR). To take into account the geometry differences (L and D) between the different sections, we will apply differents filtering factors (C_CRLC_TIA_THR, C_CRLC_TIA_IM and C_CRLC_TIA_CYL) on the temperatures values.

Three other temperature variables („temporary" variables) are created to avoid too many redundancies in the specification

. TIA_THR_TMP: used to calculate TIA_THR

. TIA_IM_TMP: used to calculate TIA_IM

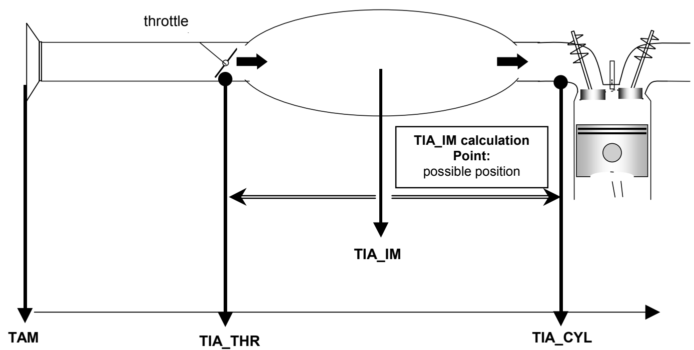
. TIA_CYL_TMP: used to calculate TIA_CYL

Model for temperature calculation will differ depending on:

_ how many TIA sensors are used

_ their respective location(s) on the intake system – depending on load sensor configuration, it differs between positioning upstream the throttle (in case of MAF sensor) and in the intake manifold (in case of MAP sensor),

Definition of TIA_IM calculation point position:



Function Description

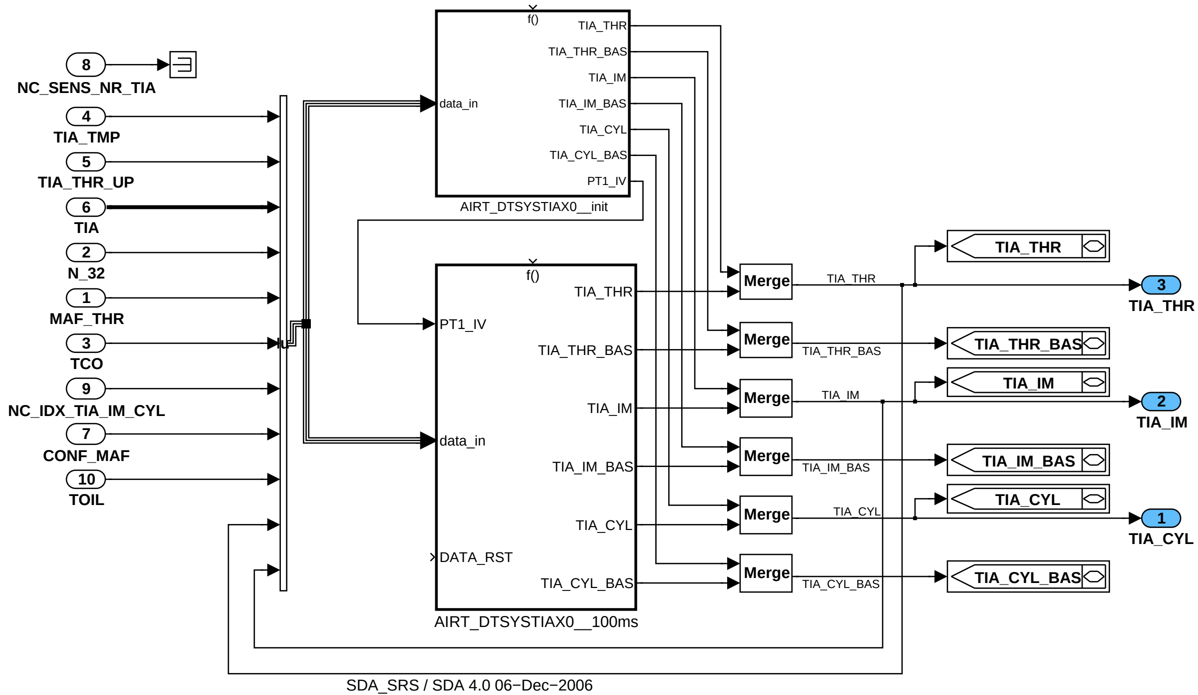


Figure 340 AIRT_DTSYSTIAX0

4.39.1.1 SUBFUNCTION: AIRT_DTSYSTIAX0__init

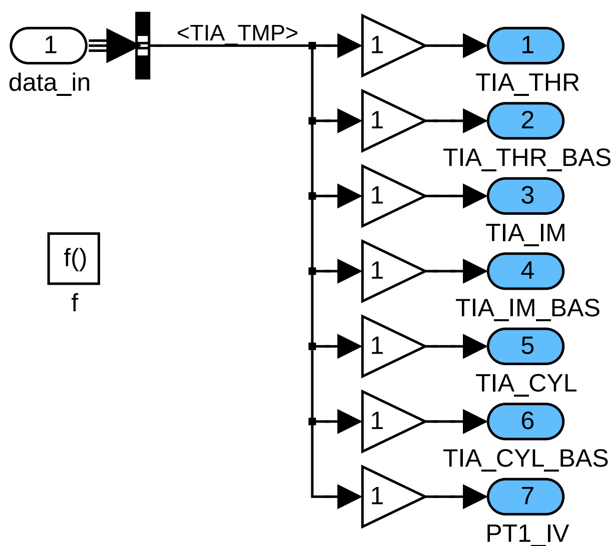


Figure 341 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__init

4.39.1.2 SUBFUNCTION: AIRT_DTSYSTIAX0__100ms

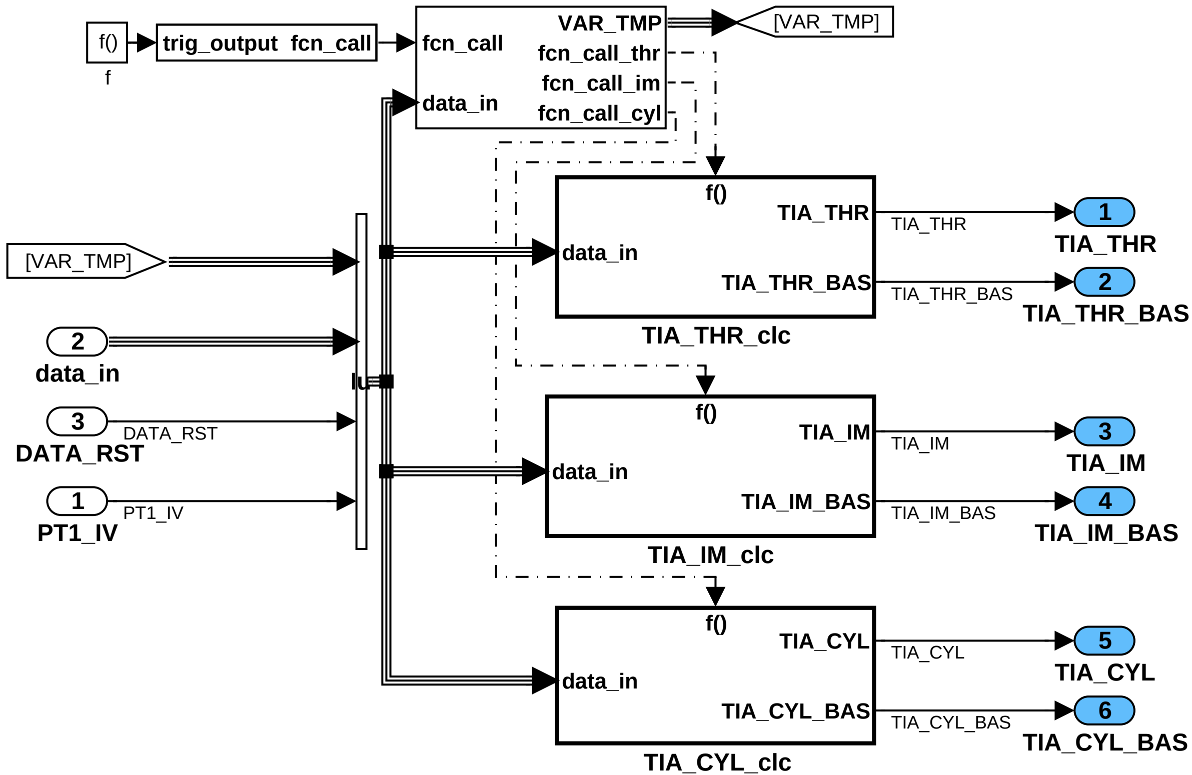


Figure 342 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms

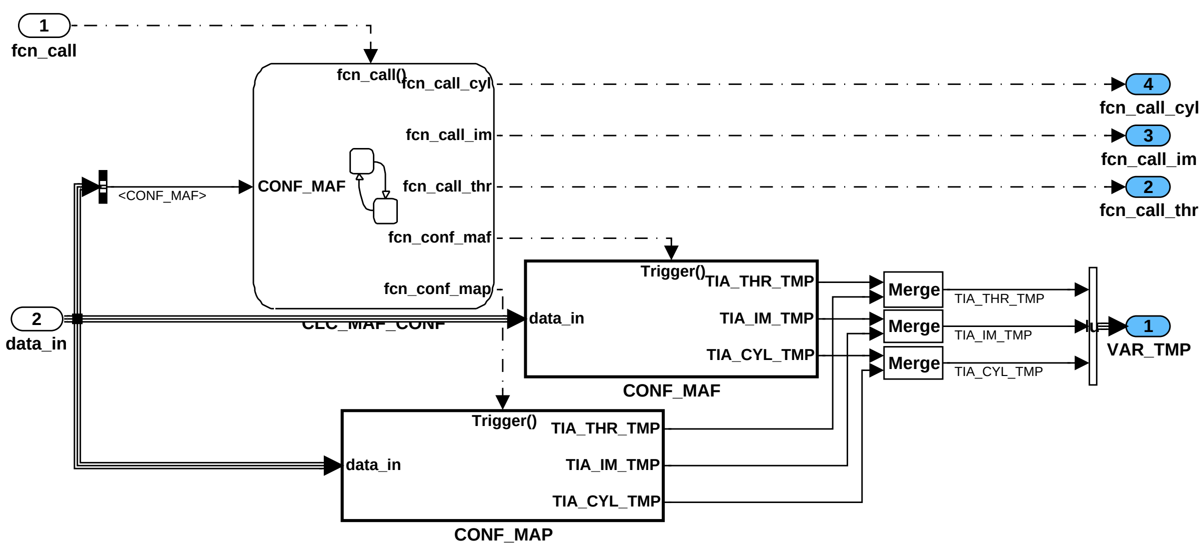


Figure 343 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ scheduler

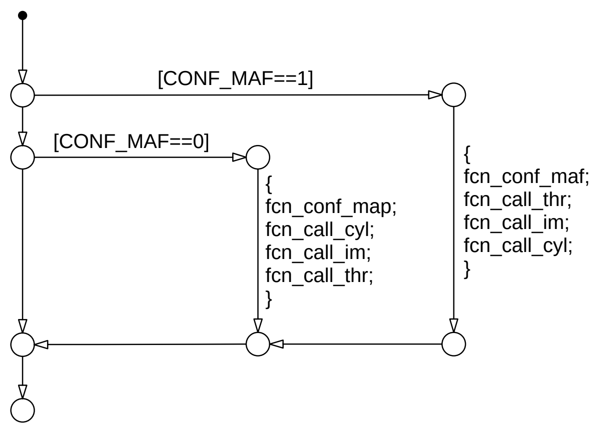


Figure 344 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ scheduler/ CLC_MAF_CONF

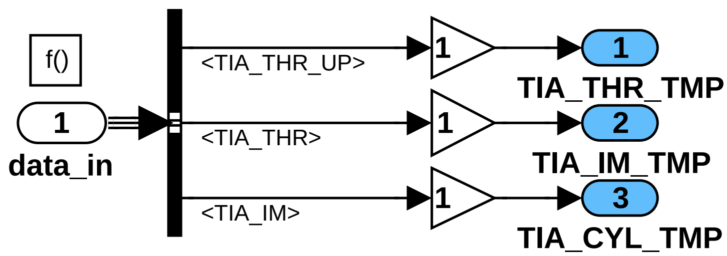


Figure 345 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ scheduler/ CONF_MAF

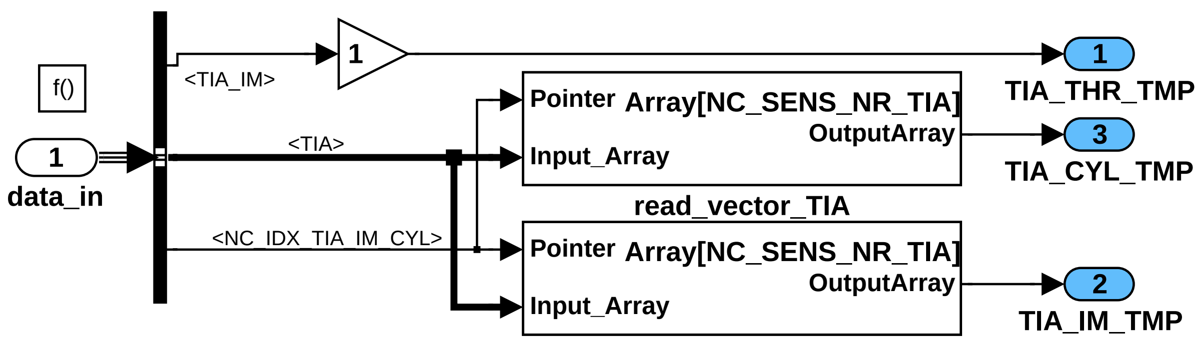


Figure 346 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ scheduler/ CONF_MAP

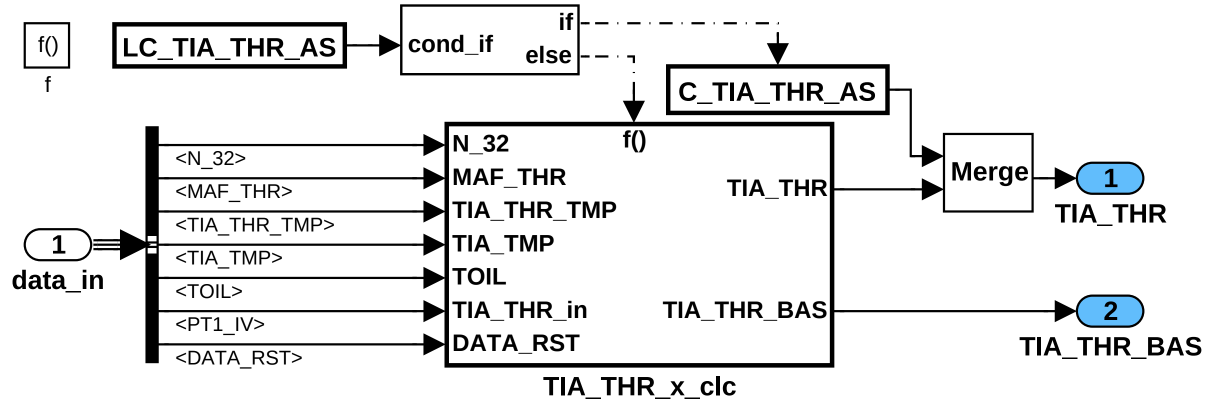


Figure 347 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_THR_clc

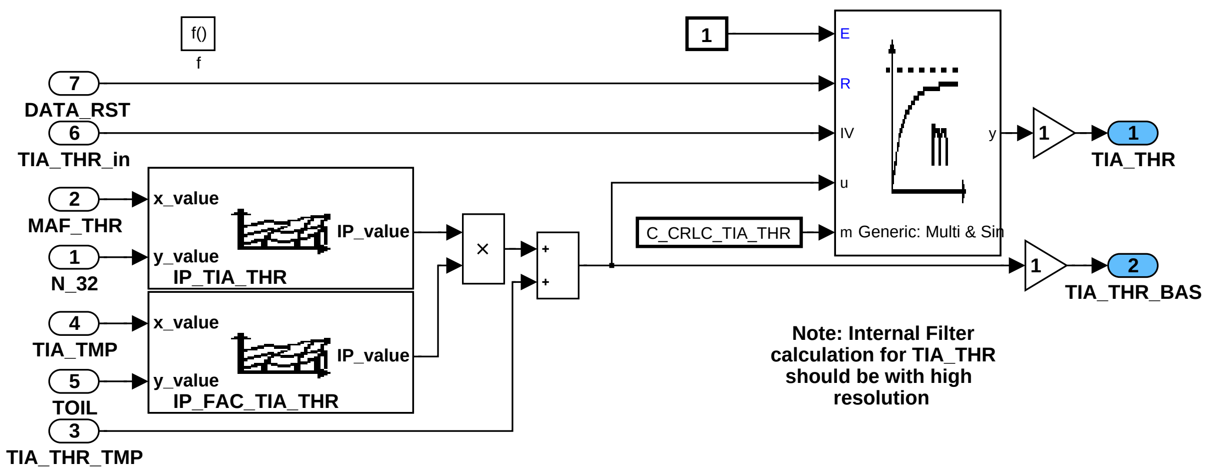


Figure 348 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_THR_clc/ TIA_THR_x_clc

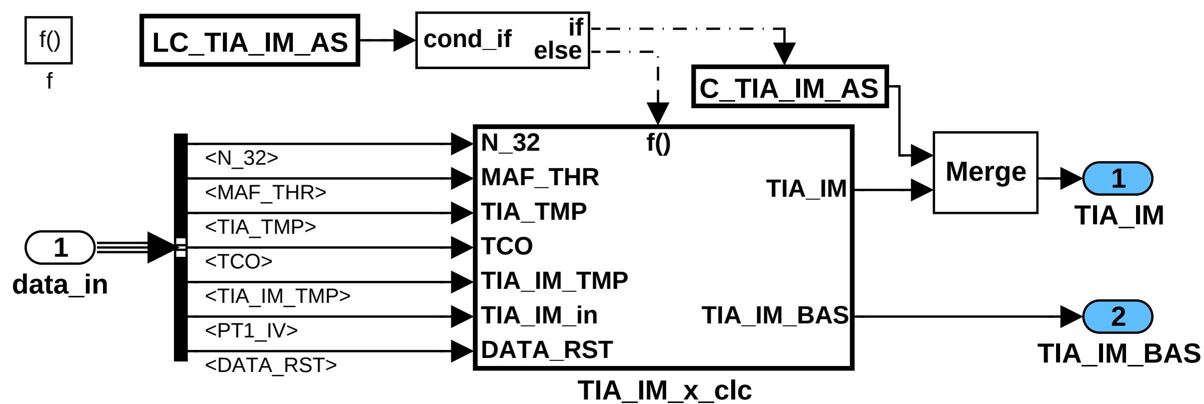


Figure 349 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_IM_clc

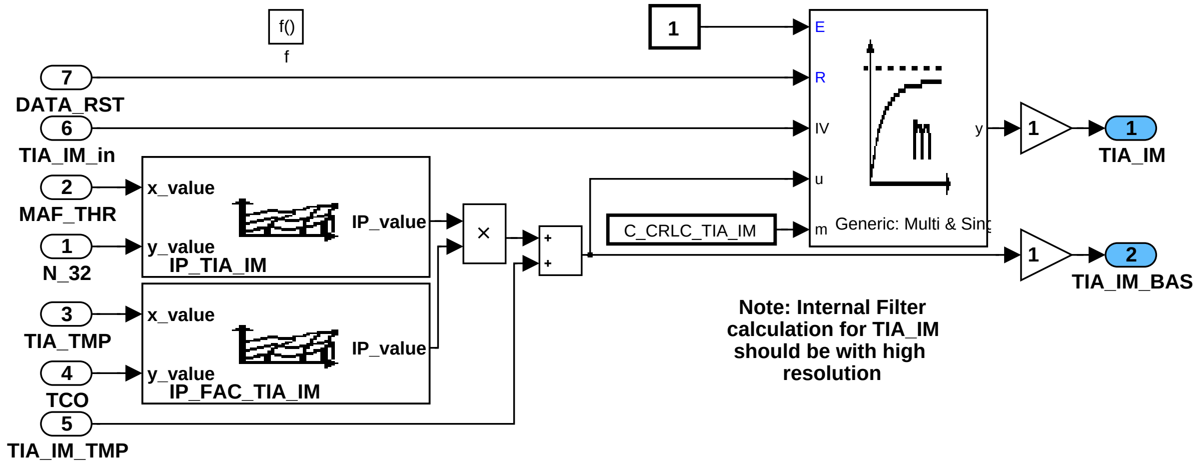


Figure 350 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_IM_clc/ TIA_IM_x_clc

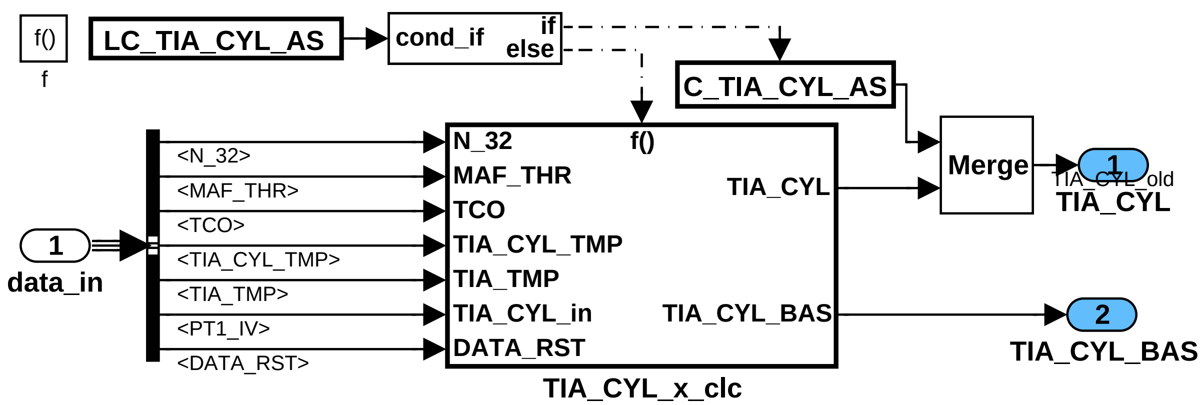


Figure 351 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_CYL_clc

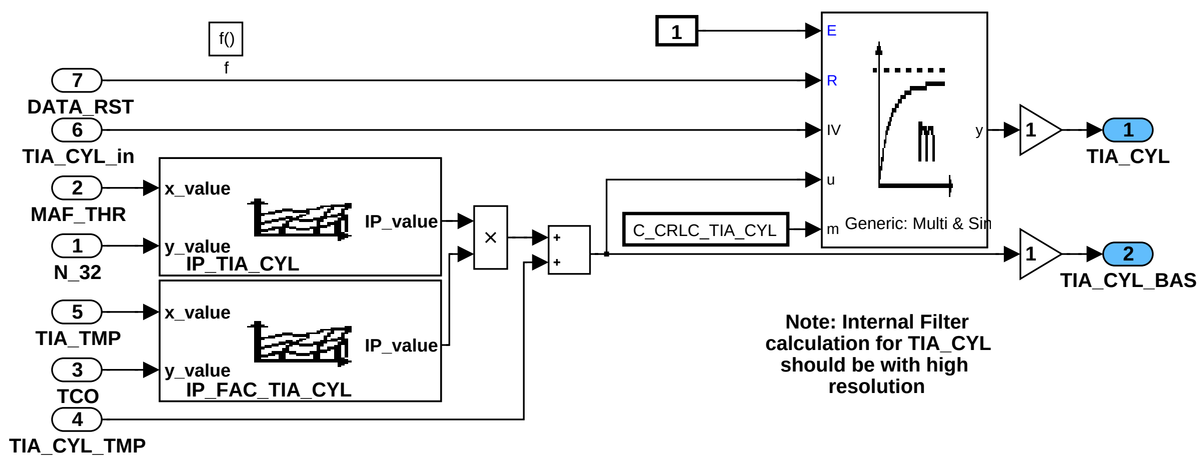


Figure 352 AIRT_DTSYSTIAX0/ AIRT_DTSYSTIAX0__100ms/ TIA_CYL_clc/ TIA_CYL_x_clc

4.40 Air Temperatures at Throttle, Intake Manifold and Cyl. (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_MIN_TOT_DC	V/O/S	0...FEH	-48...142.5	0.75	°C
Former / current driving cycle minimum TIA					
TIA_MAX_TOT_DC	V/O/S	0...FEH	-48...142.5	0.75	°C
Former / current driving cycle maximum TIA					

Input data:

TIA	LV_IGK	ERR_SYM_EL_TIA [NC_SENS_NR_TIA]	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

TIA_MIN_TOT_DC = FEH (142.5 °C)

TIA_MAX_TOT_DC = 00H (-48 °C)

- otherwise: restored from non-volantile memory

Recurrence: 1s

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If ERR_SYM_EL_TIA = NO_SYM
Then If TIA < TIA_MIN_TOT_DC
Then TIA_MIN_TOT_DC = TIA
Endif
Endif

If ERR_SYM_EL_TIA = NO_SYM
Then If TIA > TIA_MAX_TOT_DC
Then TIA_MAX_TOT_DC = TIA
Endif
Endif

4.41 AIRT Air Temperature Variables

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA[NC_SENS_NR_TIA]	O/V	0...FEH	-48...142.5	0.75	°C
Intake air temperature: one value for each TIA sensor of AIRT					
TIA_MES_ST[NC_SENS_NR_TIA]	O/V	0...FEH	-48...142.5	0.75	°C
Intake air temperature at Start: one value for each TIA sensor of AIRT					
TIA_MES_MMV[NC_SENS_NR_TIA]	O/V	0...FEH	-48...142.5	0.75	°C
Filtered air intake temperature: one value for each TIA sensor of AIRT					
TIA_THR_ST	O/V	0...FEH	-48...142.5	0.75	°C
Air temperature at the throttle body at start					
TIA_MES[NC_SENS_NR_TIA]	O/V	0...FEH	-48...142.5	0.75	°C
Intake air raw measured value: one value for each TIA sensor of AIRT					

Input data:

VP_TIA[NC_SENS_NR_TIA]	TIA_TMP	TIA_THR	TIA_SUB[NC_SENS_NR_TIA]
ERR_SYM_EL_TIA[NC_SENS_NR_TIA]	LV_ERR_EL_TIA[NC_SENS_NR_TIA]	ERR_SYM_INTM_TIA[NC_SENS_NR_TIA]	LV_ERR_INTM_TIA[NC_SENS_NR_TIA]
ERR_SYM_DYN_TIA[NC_SENS_NR_TIA]	LV_ERR_DYN_TIA[NC_SENS_NR_TIA]	ERR_SYM_PLAUS_TIA[NC_SENS_NR_TIA]	LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA]
LV_IGK	LV_ES	NC_SENS_NR_TIA	C_VP_TIA_MAX_DIAG_EL[NC_SENS_NR_TIA]
C_VP_TIA_MIN_DIAG_EL[NC_SENS_NR_TIA]			

Calibration data:

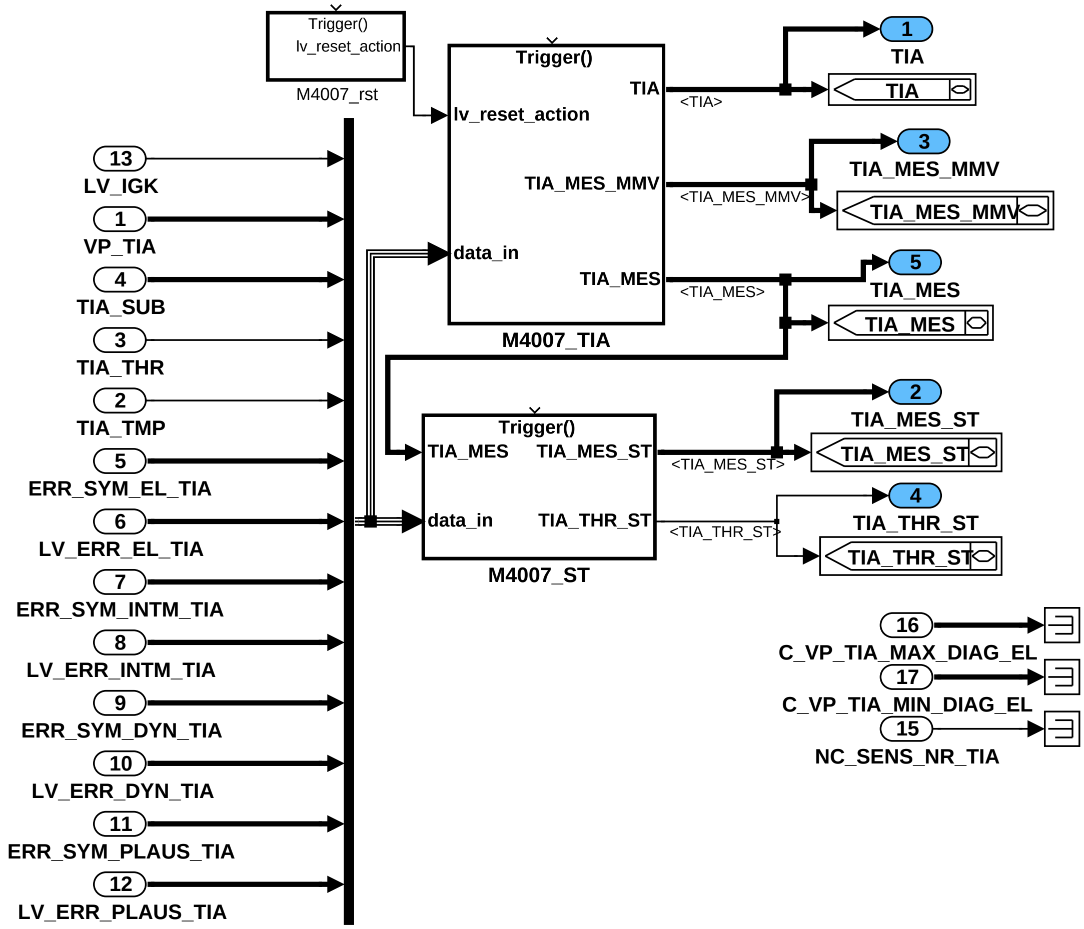
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_TIA_MES	1	0...FFH	0...0.99609307	0.003906251	-
TIA_MMV filtering factor					
IP_TIA_MES	16	0...FEH	-48...142.5	0.75	°C
LDPM_VP_TIA_1_AIRT	16	0...7FFFH	0...4.999847	1.52588E-4	V
Intake temperature linear table					

4.41.1 General

NC_SENS_NR_TIA has been defined depending on the TIA hardware definition (NC_TIA_CONF value).

The **Air temperature Variables** have to be determined for the NC_SENS_NR_TIA sensor(s) (See “Loop For m = 1 to NC_SENS_NR_TIA” below).

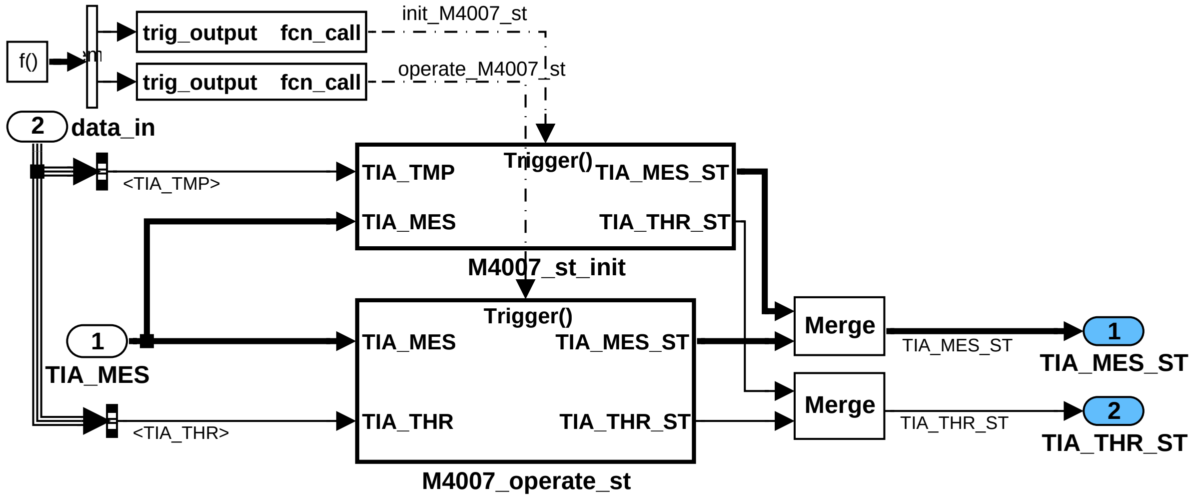
Function Description



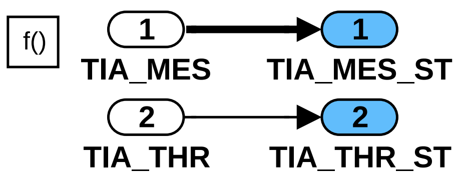
SDA_SRS / SDA 3.1 20−Apr−2004

AIRT_SIGCV0

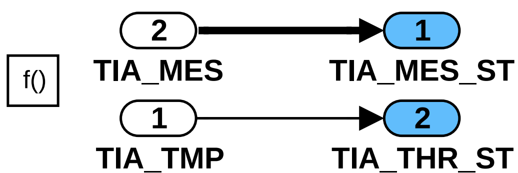
4.41.1.1 SUBFUNCTION: M4007_ST



AIRT_SIGCV0/M4007_ST

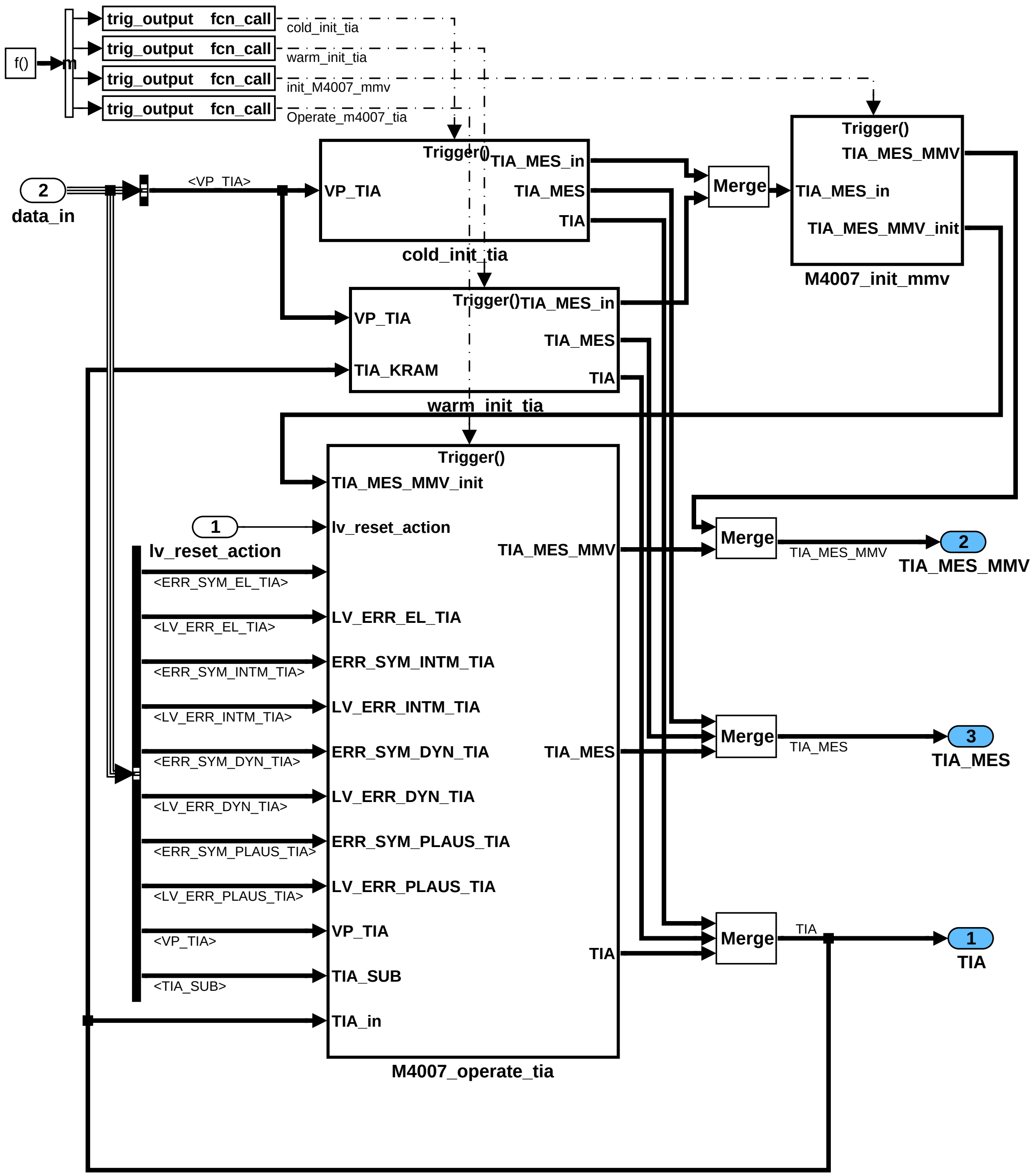


AIRT_SIGCV0/M4007_ST/M4007_operate_st



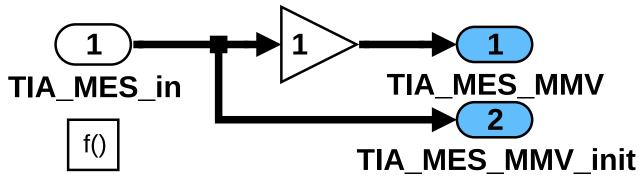
AIRT_SIGCV0/M4007_ST/M4007_st_init

4.41.1.2 SUBFUNCTION: M4007_TIA

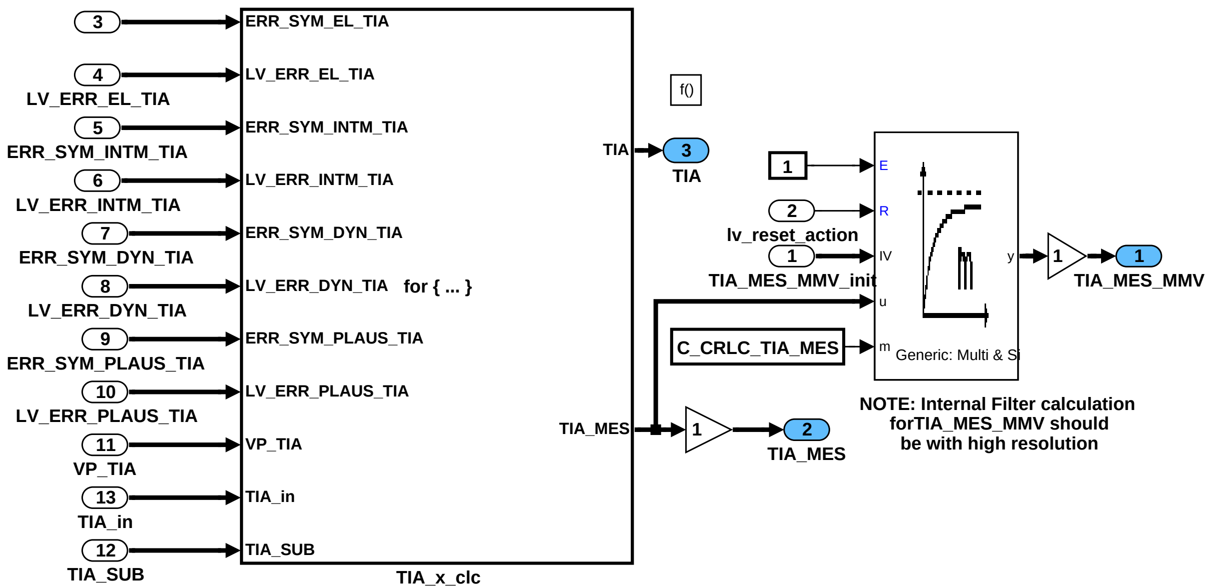


AIRT_SIGCV0/M4007_TIA

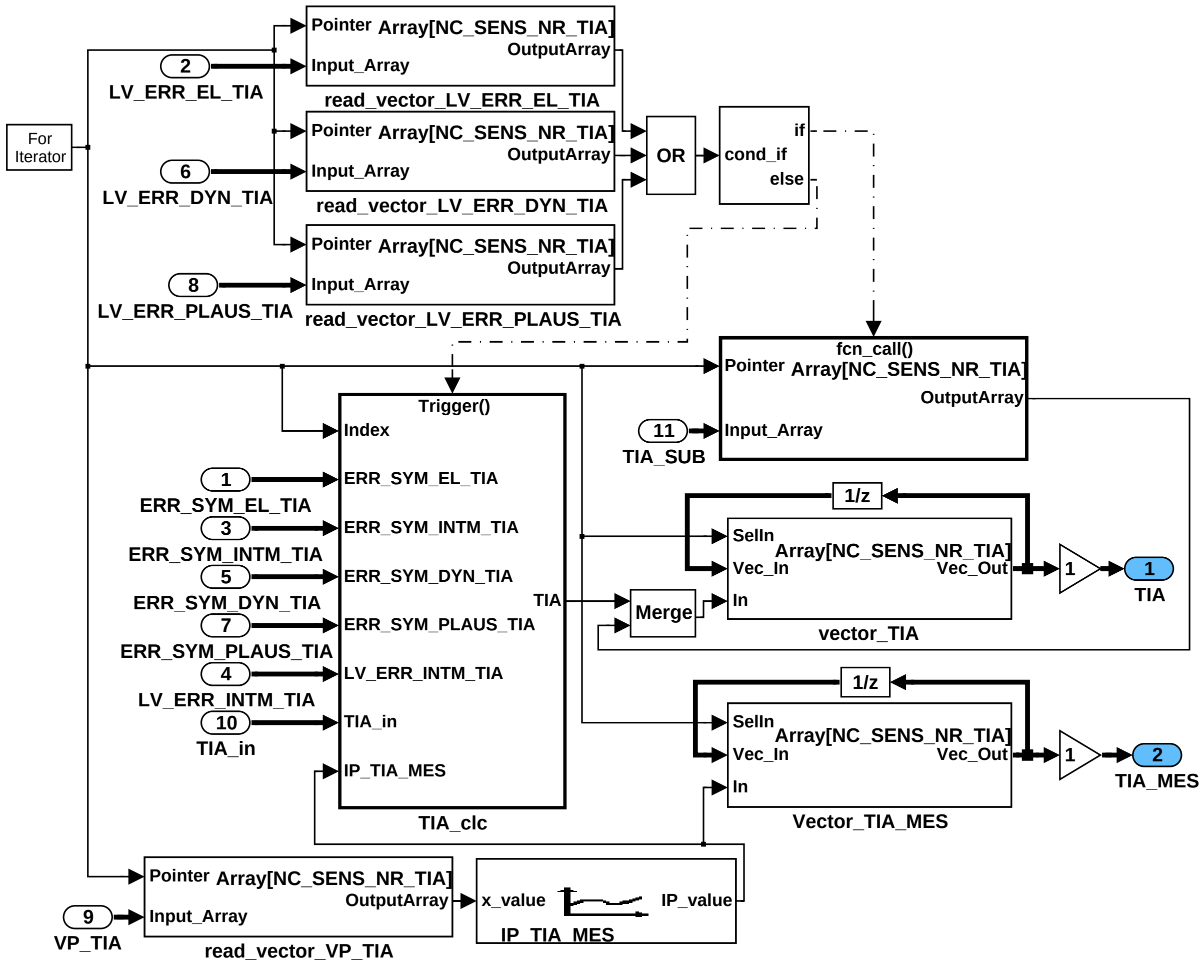
general specification



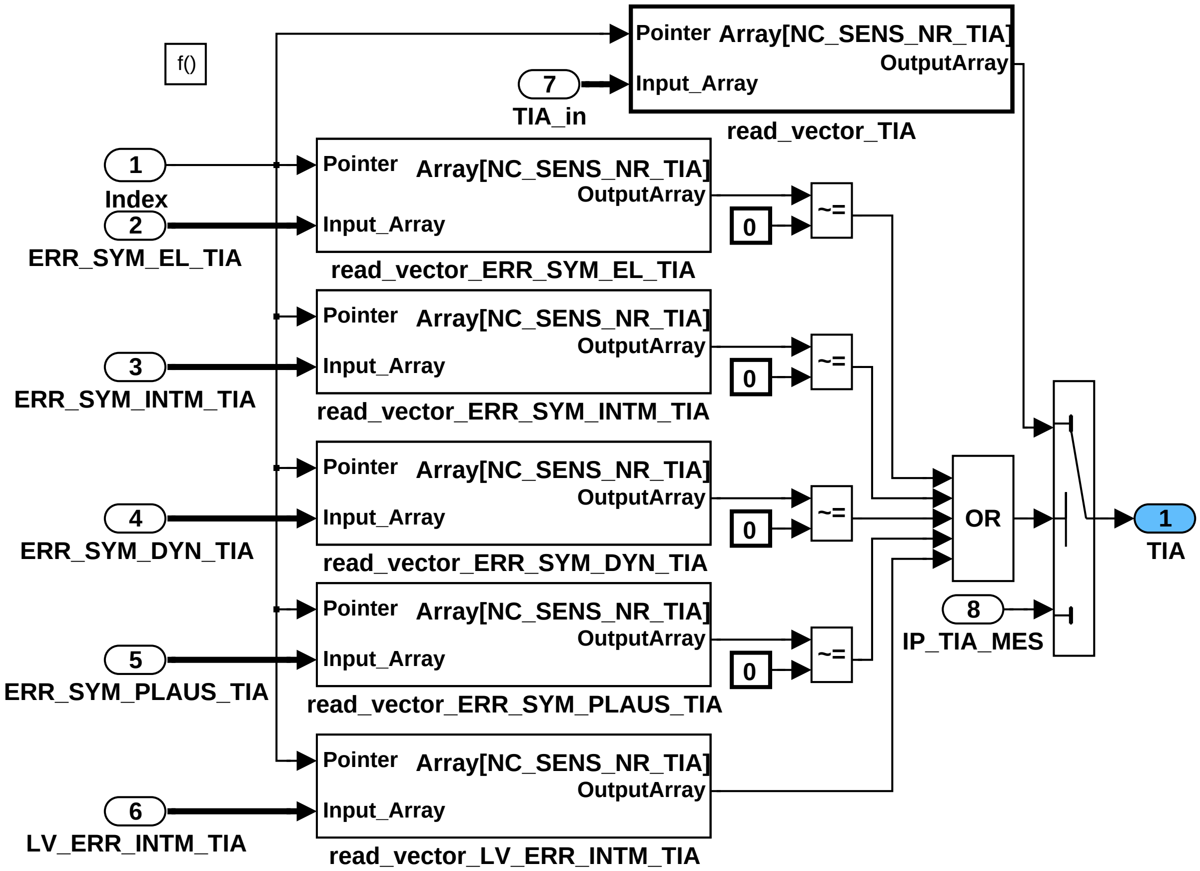
AIRT_SIGCV0/M4007_TIA/M4007_init_mmv



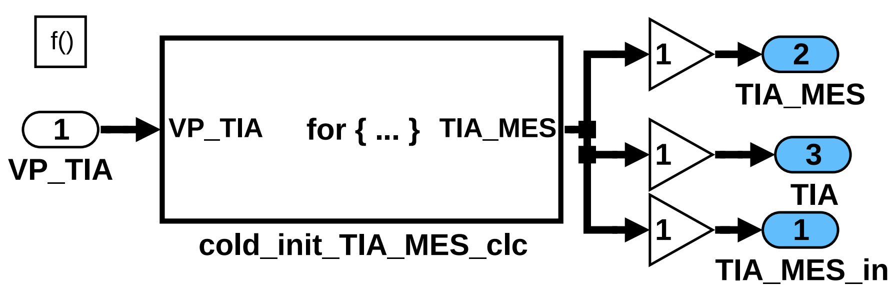
AIRT_SIGCV0/M4007_TIA/M4007_operate_tia



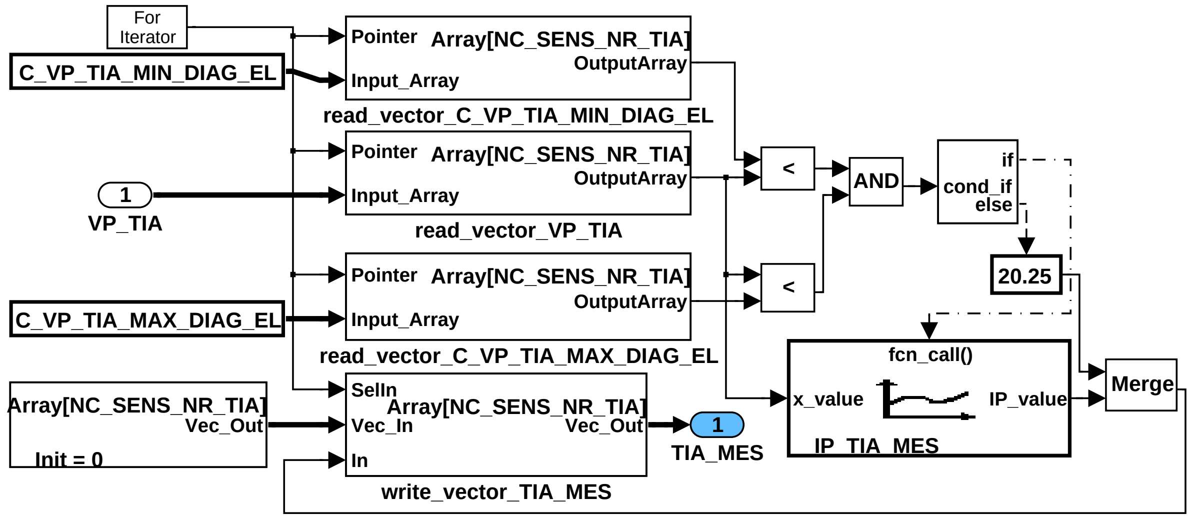
AIRT_SIGCV0/M4007_TIA/M4007_operate_tia/TIA_x_clc



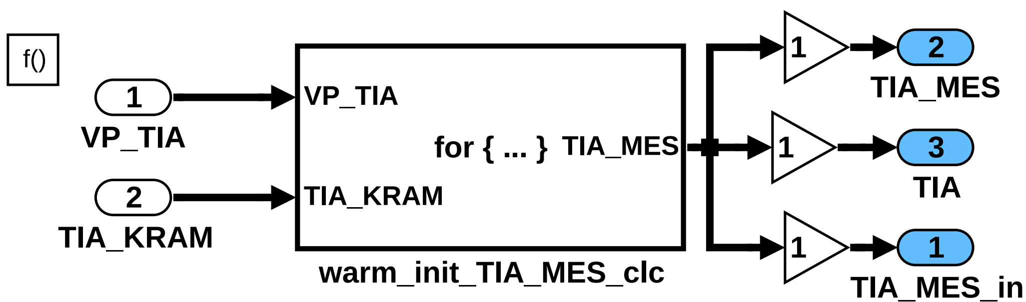
AIRT_SIGCV0/M4007_TIA/M4007_operate_tia/TIA_x_clc/TIA_clc



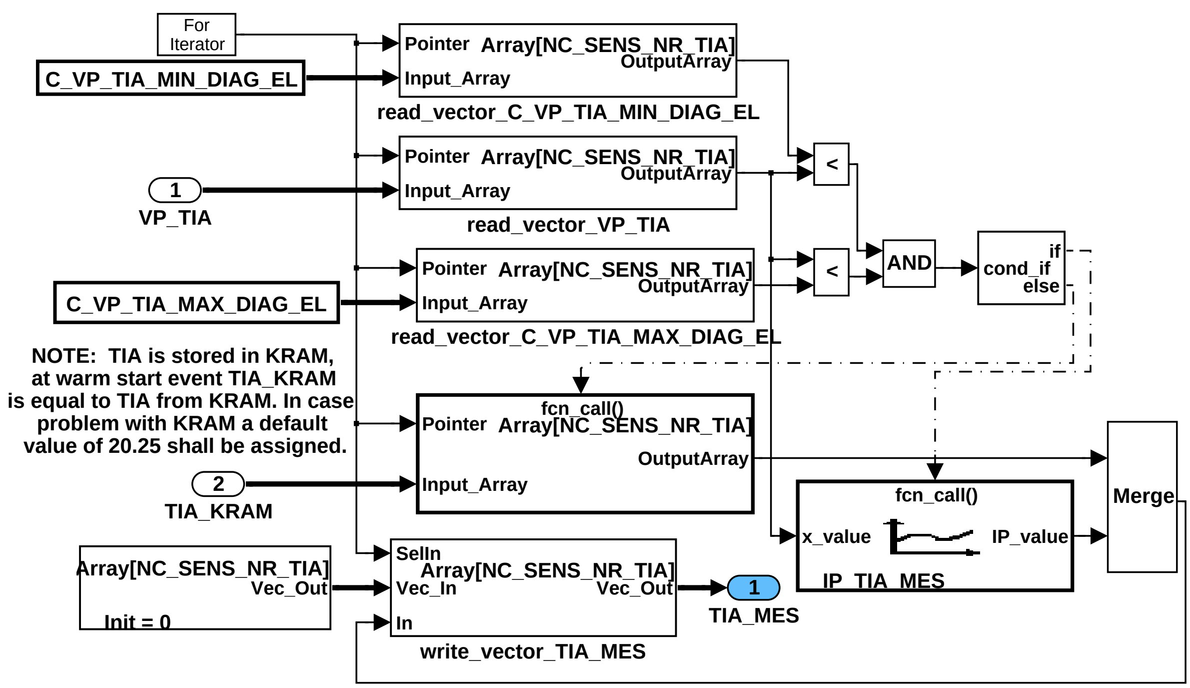
AIRT_SIGCV0/M4007_TIA/cold_init_tia



AIRT_SIGCV0/M4007_TIA/cold_init_tia/cold_init_TIA_MES_clc

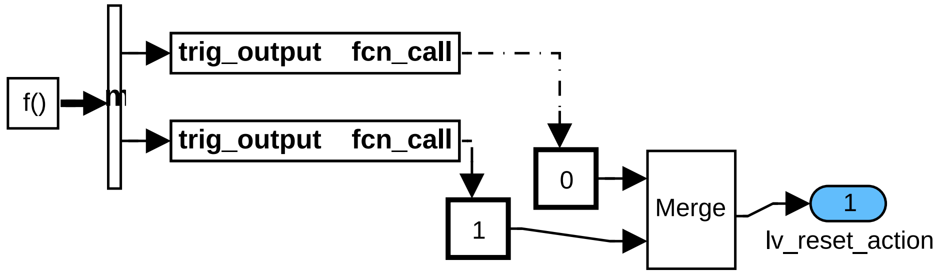


AIRT_SIGCV0/M4007_TIA/warm_init_tia



AIRT_SIGCV0/M4007_TIA/warm_init_tia/warm_init_TIA_MES_clc

4.41.1.3 SUBFUNCTION: M4007_RST



AIRT_SIGCV0/M4007_rst

4.42 Charge Air Temperatures

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_ABSV_CHA_UP	V/O	0...FFFFH	0 ... 832.051666	0.0127	K
Absolute (ie: KELVIN) Air Temperature upstream the Charger					
TIA_CHRG_DOWN	V/O	0...FEH	- 48 ... 142.5	0.75	°C
Air Temperature downstream the CHRG module					
TIA_CHRG_UP	V/O	0...FEH	- 48 ... 142.5	0.75	°C
Air Temperature upstream the CHRG module					
FAC_TIA_COR_TEMP	V/O	0...FFFFH	0 ... 3.999939	6.104e-5	-
Warm up temperature correction factor					
TIA_CHA_DOWN [NC_NR_TCHA]	V/O	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature downstream the Charger (inlet branch specific)					
TIA_CHA_DOWN_BAS	-	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TIA_CHA_DOWN					
TIA_CHA_UP	V/**O**	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature upstream the Charger					
TIA_CHA_UP_BAS	-	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TIA_CHA_UP					
TIA_RCL_DOWN	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature downstream the Recirculation Actuator					
TIA_RCL_DOWN_BAS	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TIA_RCL_DOWN					
TIA_AIC_DOWN	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature downstream the Air Cleaner casing					
TIA_AIC_DOWN_SUB	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Substitute value of TIA_AIC_DOWN					
TIA_AIC_DOWN_BAS	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TIA_AIC_DOWN					
TIA_THR_TAM_DIF	V	80...7FH	-96...95.25	0.75	°C
Air Temperature Difference between TIA_THR and TAM					
TIA_RCL_AIC_DOWN_DIF	-	8000...7FFFH	-192...191.9941	5.8594e-3	°C
Temperature difference between TIA_RCL_DOWN$_{(n-1)}$ and TIA_AIC_DOWN for TIA_RCL_DOWN computation					
CTR_TIA_RCL_DOWN	V	0...FFH	0...255	1	-
Counter used for TIA_RCL_DOWN determination during RCL valve closing phase					
FAC_TIA_CHA_DOWN [NC_NR_TCHA]	V	0...FFFFH	0 ... 4	6.104e-5	-
Global factor for TIA_CHA_DOWN calculation					
TIA_AIC_DOWN_DIF	-	8000...7FFFH	-192...191.9941	5.8594e-3	°C
TIA_AIC_DOWN temperature increase					
TIA_RCL_DOWN_DIF	-	8000...7FFFH	-192...191.9941	5.8594e-3	°C
Air temperature difference between TIA_RCL_DOWN_BAS and TIA_RCL_DOWN$_{(n-1)}$					
TIA_CHA_DOWN_DIF	-	8000...7FFFH	-192...191.9941	5.8594e-3	°C
Air temperature difference between TIA_CHA_DOWN_BAS and TIA_CHA_DOWN$_{(n-1)}$					
TIA_CHA_UP_DIF	V	8000...7FFFH	-192...191.9941	5.8594e-3	°C
Air temperature difference between TIA_CHA_UP_BAS and TIA_CHA_UP$_{(n-1)}$					
TAM_CHA_SUB_BAS	V	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TAM_CHA_SUB					
TIA_RCL_UP	O	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature upstream the Recirculation Actuator					
TEMP_HOT	-	0...FEH	-48...142.5	0.75	°C
Reference hot source air temperature used for warm up correction determination					
TEMP_COLD	-	0...FEH	-48...142.5	0.75	°C
Reference cold source air temperature used for warm up correction determination					
TIA_AIC_DOWN_SUB_BAS	**-**	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Raw value of TIA_AIC_DOWN_SUB					
TAM_CHA_SUB	-	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Substitute value of TAM_CHA (modeled ambient air temperature substitute value – for charger engine)					

FLOW_RCL_RATIO	V	0...FFFFH	0 ... 1.99996948	3.0518e-5	-
Flow ratio between MAF_RCL and (MAF_AIC+MAF_RCL)					
LV_TAM_MES_VLD_TCHA	-	0...1H	0...1	1	-
Bit indicating that there is a valid measured ambient temperature signal for CHRG calculations					
TIA_CHA_DOWN_MV	V/O	8000...7FFFH	-48...335.99414	5.8594e-3	°C
Air Temperature downstream the Charger (inlet branch average)					

Input data:

TIA_ICO_UP [NC_NR_TCHA]	TIA_THR	TAM	MAF_KGH_AIC_TCHA [NC_NR_TCHA]
PQ_CHA [NC_NR_TCHA]	LV_ERR_TAM	N_CHA [NC_NR_TCHA]	NC_RCL_UP_ICO
MAF_RCL [NC_NR_TCHA]	LC_TCHA_CONF	NC_TIA_CONF	NC_TAM_CAN_USE
TCO	NC_NR_TCHA		

FUNCTION DESCRIPTION:

General information:

Version valid for : NC_CHRG_CONF <> 0
And (NC_TIA_CONF = 10,13 And NC_TAM_CAN_USE = 1
Or NC_TIA_CONF = 22,23)

Two basic informations are needed from AIRT Aggregate to compute the charger intake air temperatures models:
_ an intake air temperature information at the throttle (or downstream the throttle) and
_ an ambient air temperature information (TAM_CAN **or/and** the information from the sensor situated closed to the air filter case – called "TIA_AM_CHRG" in AIRT)
remark: configurations set with NC_TIA_CONF = 12, 24 (TIA sensor "TIA_CHRG" inside the CHRG Aggregate) are not covered as no TIA diagnosis is available inside CHRG Aggregate currently.

The aim of this module is to define:
_ the relative value (ie: °C) of the temperatures:
. downstream the air cleaner casing
. upstream / downstream the charger
. upstream / downstream the recirculation actuator
and the relevant substitute values in case of unvalid TAM value
_ the absolute value (ie: Kelvin) of the temperature upstream charger
_ finally, the temperature interfaces with AIRT Aggregate.

4.42.1 Intake air temperatures in CHRG:

Description:

Application conditions:

Initialisation: At ECU Reset

TEMP_COLD, TEMP_HOT, FAC_TIA_COR_TEMP: using formula section

TIA_ABSV_CHA_UP = **TAM** + 273

TIA_AIC_DOWN = TIA_CHA_UP = TIA_RCL_UP = TIA_RCL_DOWN_BAS =
TIA_RCL_DOWN = **TAM**

for i = 1 to NC_NR_TCHA **do**

TIA_CHA_DOWN [i] = **TAM**

endfor

TIA_CHA_DOWN_MV = TAM

TAM_CHA_SUB_BAS = TIA_AIC_DOWN_SUB_BAS = TIA_AIC_DOWN_SUB = **TIA_THR**

Other variables listed in Output Data list and refering to this chapter = 0H

Recurrence: **100 ms**

Activation: LC_TCHA_CONF = 1 ***(important: valid for Chap 1.1 only !!)***

Deactivation: LC_TCHA_CONF = 0 ***(important: valid for Chap 1.1 only !!)***

Formula section:

Remark: Every calculation input must be in [°C]

if NC_NR_TCHA = 1 **then**

MAF_RCL_MV_TMP = MAF_RCL [1]

MAF_KGH_AIC_TCHA_MV_TMP = MAF_KGH_AIC_TCHA [1]

else

MAF_RCL_MV_TMP = (MAF_RCL [1] + MAF_RCL [2]) / 2

MAF_KGH_AIC_TCHA_MV_TMP =
(MAF_KGH_AIC_TCHA [1] + MAF_KGH_AIC_TCHA [2]) / 2

endif

Substitute TAM calculation based on downstream temperature (TIA_THR):

TIA_THR_TAM_DIF = TIA_THR - TAM

TAM_CHA_SUB_BAS = TIA_AIC_DOWN_SUB $_{N-1}$ -
IP_TIA_AIC_DOWN_OFS (MAF_KGH_AIC_TCHA_MV_TMP, TIA_THR_TAM_DIF)

TAM_CHA_SUB $_{N}$ = TAM_CHA_SUB $_{N-1}$ +
C_CRLC_TAM_CHA_SUB * (TAM_CHA_SUB_BAS $_{N}$ - TAM_CHA_SUB $_{N-1}$)

if TAM_CHA_SUB <= TIA_AIC_DOWN_SUB $_{N-1}$ **then**

TAM_CHA_SUB = TIA_AIC_DOWN_SUB $_{N-1}$

endif

FAC_TIA_COR_TEMP calculation:

Definition of the reference cold and hot sources (via C_TEMP_COLD_HOT_SRC)

if C_TEMP_COLD_HOT_SRC = "TIA_THR_TCO" **then**

TEMP_COLD = TIA_THR

TEMP_HOT = TCO

endif

if C_TEMP_COLD_HOT_SRC = "TAM_TCO" **and**
(NC_TAM_CAN_USE = 1 **or** NC_TIA_CONF = 22 **or** 23) **then**

TEMP_COLD = TAM

TEMP_HOT = TCO

endif

Definition of the warm_up correction (used commonly for high temperature offsets)

FAC_TIA_COR_TEMP = IP_FAC_TIA_COR_TEMP (TEMP_COLD, TEMP_HOT)

<u>LV_TAM_MES_VLD_TCHA calculation:</u>

if LV_ERR_TAM = 1 **and** LC_ERR_TAM_SUPP = 0 **then**

LV_TAM_MES_VLD_TCHA = 0 // TAM info not valid: calculations are based on TIA_THR

else

LV_TAM_MES_VLD_TCHA = 1 // TAM info valid: calculations are based on TAM

endif

<u>TIA_AIC_DOWN calculation:</u>

// TIA_AIC_DOWN_SUB can be used as input in TAM substitute formula (in AIRT)
// via TIA_CHRG_UP (see TIA_CHRG_UP definition in Chap1.2)

if MAF_RCL_MV_TMP = 0 **then**

TIA_AIC_DOWN_SUB_BAS = $TIA\_CHA\_UP_{N-1}$ -
IP_TIA_CHA_UP_OFS (MAF_KGH_AIC_TCHA_MV_TMP, TIA_THR_TAM_DIF)

else // TIA_AIC_DOWN_SUB_BAS remains unchanged

endif

$TIA\_AIC\_DOWN\_SUB_{N}$ = $TIA\_AIC\_DOWN\_SUB_{N-1}$ + C_CRLC_TIA_AIC_DOWN_SUB *
(TIA_AIC_DOWN_SUB_BAS - $TIA\_AIC\_DOWN\_SUB_{N-1}$)

// Temperature always increases between air cleaner and charger

if TIA_AIC_DOWN_SUB >= $TIA\_CHA\_UP_{N-1}$ **then**

TIA_AIC_DOWN_SUB = $TIA\_CHA\_UP_{N-1}$

endif

if LV_TAM_MES_VLD_TCHA = 0 **then** // calculations based on TIA_THR

TIA_AIC_DOWN = TIA_AIC_DOWN_SUB

else // calculations based on TAM

TIA_AIC_DOWN_BAS = TAM +
IP_TIA_AIC_DOWN_OFS (MAF_KGH_AIC_TCHA_MV_TMP, TIA_THR_TAM_DIF)

TIA_AIC_DOWN_DIF = TIA_AIC_DOWN_BAS - $TIA\_AIC\_DOWN_{N-1}$

$TIA\_AIC\_DOWN_{N}$ = $TIA\_AIC\_DOWN_{N-1}$ +
IP_CRLC_TIA_AIC_DOWN_RGL (TIA_AIC_DOWN_DIF) *
(TIA_AIC_DOWN_BAS - $TIA\_AIC\_DOWN_{N-1}$)

endif

for i = 1 to NC_NR_TCHA **do**

TIA_CHA_DOWN calculation:

FAC_TIA_CHA_DOWN [i] = 1 + ((IP_FAC_TIA_CHA_DOWN_LOAD (PQ_CHA [i], N_CHA [i]) - 1) * FAC_TIA_COR_TEMP)

if LC_TIA_CHA_DOWN_AS = 1 **then**

TIA_CHA_DOWN [i] = C_TIA_CHA_DOWN_AS

else

if LV_TAM_MES_VLD_TCHA = 0 **then** // calculations based on TIA_THR

TIA_CHA_DOWN [i] = TIA_ICO_UP [i]

else // calculations based on TAM

TIA_CHA_DOWN_BAS = ((TIA_CHA_UP $_{N-1}$ + 273) * FAC_TIA_CHA_DOWN [i]) - 273

TIA_CHA_DOWN_DIF = TIA_CHA_DOWN_BAS - TIA_CHA_DOWN [i] $_{N-1}$

TIA_CHA_DOWN [i] $_{N}$ = TIA_CHA_DOWN [i] $_{N-1}$ + TIA_CHA_DOWN_DIF * IP_CRLC_TIA_CHA_DOWN_RGL (TIA_CHA_DOWN_DIF)

endif

endfor

if NC_NR_TCHA = 1 **then**

TIA_CHA_DOWN_MV = TIA_CHA_DOWN [1] $_{N-1}$

FAC_TIA_CHA_DOWN_MV_TMP = FAC_TIA_CHA_DOWN [1] $_{N-1}$

else

TIA_CHA_DOWN_MV = (TIA_CHA_DOWN [1] $_{N-1}$ + TIA_CHA_DOWN [2] $_{N-1}$) / 2

FAC_TIA_CHA_DOWN_MV_TMP = (FAC_TIA_CHA_DOWN [1] $_{N-1}$ + FAC_TIA_CHA_DOWN [2] $_{N-1}$) / 2

endif

TIA_RCL_UP calculation:

// independend of whether there is a valid TAM, The temperature upstream the recirculation actuator is depending on the recirculation pipe inlet position (ie: upstream or downstream the intercooler)

#if NC_RCL_UP_ICO = 1 **then**

TIA_RCL_UP = TIA_CHA_DOWN_MV

#else

TIA_RCL_UP = TIA_THR

#endif

TIA_RCL_DOWN calculation:

TIA_RCL_AIC_DOWN_DIF = TIA_RCL_DOWN $_{N-1}$ - TIA_AIC_DOWN

if MAF_RCL_MV_TMP >= C_MAF_RCL_TIA_MIN **then**

CTR_TIA_RCL_DOWN = 0 (counter reset)

TIA_RCL_DOWN_BAS = TIA_RCL_UP –
(IP_TIA_RCL_DOWN_OFS (MAF_RCL_MV_TMP) * FAC_TIA_COR_TEMP)

else

CTR_TIA_RCL_DOWN = CTR_TIA_RCL_DOWN + 1

if CTR_TIA_RCL_DOWN >=
IP_CTR_TIA_RCL_DOWN (TIA_RCL_AIC_DOWN_DIF) **then**

TIA_RCL_DOWN_BAS $_N$ = TIA_RCL_DOWN_BAS $_{N-1}$ -
IP_TIA_RCL_DOWN_DEC (TIA_RCL_AIC_DOWN_DIF)

CTR_TIA_RCL_DOWN = 0 (counter reset)

endif

endif

TIA_RCL_DOWN_DIF = TIA_RCL_DOWN_BAS - TIA_RCL_DOWN $_{N-1}$

TIA_RCL_DOWN $_N$ = TIA_RCL_DOWN $_{N-1}$ + (TIA_RCL_DOWN_DIF *
IP_CRLC_TIA_RCL_DOWN (TIA_RCL_DOWN_DIF))

if TIA_RCL_DOWN <= TIA_AIC_DOWN **then**

TIA_RCL_DOWN = TIA_AIC_DOWN

endif

TIA_CHA_UP calculation:

if LV_TAM_MES_VLD_TCHA = 0 **then** // calculations based on TIA_THR

TIA_CHA_UP_BAS = (TIA_CHA_DOWN_MV + 273) /
FAC_TIA_CHA_DOWN_MV_TMP - 273

TIA_CHA_UP $_N$ = TIA_CHA_UP $_{N-1}$ + C_CRLC_TIA_CHA_UP_SUB *
(TIA_CHA_UP_BAS - TIA_CHA_UP $_{N-1}$)

TIA_ABSV_CHA_UP = TIA_THR + 273

// In case of TAM failure, TAM substitute value (model value calculated in AIRT Agg.) can be underestimated in case of the vehicle is not started for a long period of time. This leads to a underestimation of TIA_ABSV_CHA_UP and of the charger speed N_CHA. Also, in case of TAM failure, TIA_ABSV_CHA_UP is approached by TIA_THR (as TIA is always higher than TAM). In this way, we insure that the charger is always adequately protected from overspeed.

else // calculations based on TAM

FLOW_RCL_RATIO = min (1, (MAF_RCL_MV_TMP / (MAF_RCL_MV_TMP +
MAF_KGH_AIC_TCHA_MV_TMP)) * C_FAC_FLOW_RCL_RATIO)

// FLOW_RCL_RATIO limited to maximum 1

TIA_CHA_UP_BAS = (TIA_RCL_DOWN * FLOW_RCL_RATIO) +

((1 – FLOW_RCL_RATIO) * TIA_AIC_DOWN) +
IP_TIA_CHA_UP_OFS (MAF_KGH_AIC_TCHA_MV_TMP, TIA_THR_TAM_DIF)

TIA_CHA_UP_DIF = TIA_CHA_UP_BAS - TIA_CHA_UP $_{N-1}$

TIA_CHA_UP $_N$ = TIA_CHA_UP $_{N-1}$ + TIA_CHA_UP_DIF *
IP_CRLC_TIA_CHA_UP (MAF_KGH_AIC_TCHA_MV_TMP, TIA_CHA_UP_DIF)

TIA_ABSV_CHA_UP = TIA_CHA_UP + 273

endif

4.42.2 Intake air temperature interfaces between CHRG and AIRT:

Description:

Application conditions:

Initialisation: **At** Reset

TIA_CHRG_UP = TIA_CHRG_DOWN = TIA_THR

Recurrence: **100 ms**

Activation: for all Engine States

Deactivation: -

Formula section:

Remark: Every calculation input must be in [°C]

TIA_CHRG_UP:

#If NC_TIA_CONF = **10** Or **13**
#Then **If** LC_TCHA_CONF = 0
Or (LC_TCHA_CONF =1 **And** LC_TIA_CHRG_UP_CLC = 0)
Then TIA_CHRG_UP = TIA_THR
Else TIA_CHRG_UP = TAM_CHA_SUB
Endif
#Endif *(TIA_CHRG_UP not needed for AIRT: TIA_CHRG_UP keeps init value)*

TIA_CHRG_DOWN :

(TIA_CHRG_DOWN not needed for AIRT: TIA_CHRG_UP keeps init value)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEMP_COLD_HOT_SRC	1	0...1H 0H 1H	0 ... 1 0 = TIA_THR_TCO 1 = TAM_TCO	1	-
Choice of cold source temperature					
C_FAC_FLOW_RCL_RATIO	1	0...FFH	0 ... 1.992	0.0078	-
Multiplicative corrective factor on FLOW_RCL_RATIO (typical value=1)					
C_TIA_CHA_DOWN_AS	1	8000...7FFFH	-192...191.9941	5.8594e-3	°C
Application system value for TIA_CHA_DOWN calculation					

C_MAF_RCL_TIA_MIN	1	0...FFFFH	0 ... 2047.96875	0.03125	Kg/h
Minimum MAF_RCL threshold value for TIA_RCL_DOWN determination					
C_CRLC_TAM_CHA_SUB	1	0...FFFFH	0 ... 0.999984	0.15259e-4	-
TAM_CHA_SUB mean value filtering factor					
C_CRLC_TIA_CHA_UP_SUB	1	0...FFFFH	0 ... 0.999984	0.15259e-4	-
TIA_CHA_UP_SUB mean value filtering factor					
C_CRLC_TIA_AIC_DOWN_SUB	1	0...FFFFH	0 ... 0.999984	0.15259e-4	-
TIA_AIC_DOWN_SUB mean value filtering factor					
IP_CRLC_TIA_RCL_DOWN	1*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDPM_ TIA_ 1_CHRG	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
TIA_RCL_DOWN mean value filtering factor during RCL opening phase					
IP_CRLC_TIA_CHA_UP	6*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDPM_MAF_AIC_IP_TIA	6	0... FFFFH	0 ... 2047.96875	0.03125	Kg/h
LDPM_ TIA_ 1_CHRG	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
TIA_CHA_UP mean value filtering factor					
IP_CRLC_TIA_CHA_DOWN_RGL	1*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDPM_ TIA_ 1_CHRG	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
TIA_CHA_DOWN mean value filtering factor					
IP_CRLC_TIA_AIC_DOWN_RGL	1*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDPM_ TIA_ 1_CHRG	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
TIA_AIC_DOWN_RGL mean value filtering factor					
IP_TIA_CHA_UP_OFS	6*6	0...FFFFH	-192...191.9941	5.8594e-3	°C
LDPM_MAF_AIC_IP_TIA	6	0... FFFFH	0 ... 2047.96875	0.03125	Kg/h
LDPM_TIA_THR_TAM_DIF_1_CHRG	6	0...FFH	-96...95.25	0.75	°C
Load and temperature influences on TIA_CHA_UP determination					
IP_TIA_AIC_DOWN_OFS	6*6	0...FFFFH	-192...191.9941	5.8594e-3	°C
LDPM_MAF_AIC_IP_TIA	6	0... FFFFH	0 ... 2047.96875	0.03125	Kg/h
LDPM_TIA_THR_TAM_DIF_1_CHRG	6	0...FFH	-96...95.25	0.75	°C
Load and temperature influences on TIA_AIC_DOWN determination					
IP_CTR_TIA_RCL_DOWN	1*6	0...FFH	0...255	1	-
LDPM_TIA_RCL_AIC_DOWN_DIF	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
CTR_TIA_RCL_DOWN counter value for decrementation of TIA_RCL_DOWN during RCL closing phase					
IP_TIA_RCL_DOWN_DEC	1*6	0...FFFFH	-192...191.9941	5.8594e-3	°C
LDPM_TIA_RCL_AIC_DOWN_DIF	6	0...FFFFH	-192...191.9941	5.8594e-3	°C
Decrement for TIA_RCL_DOWN computation for RCL closing phase					
IP_TIA_RCL_DOWN_OFS	1*4	0...FFFFH	-192...191.9941	5.8594e-3	°C
LDP_MAF_RCL_IP_TIA_RCL_DOWN	4	0...FFFFH	0 ... 2047.96875	0.03125	Kg/h
Air Temperature drop between upstream and downstream Recirculation actuator (positive values)					
IP_FAC_TIA_CHA_DOWN_LOAD	8*12	0...FFFFH	0 ... 4	6.104e-5	-
LDP_PQ_CHA_IP_FAC_TIA_CHA_DOWN	8	0...FFFFH	0 ..15.999756	244.14e-6	-
LDP_N_CHA_IP_FAC_TIA_CHA_DOWN	12	0...FFFFH	0...400 000	6.1035	1/min
Air Temperature increase factor through the Charger (typical value=1)					
IP_FAC_TIA_COR_TEMP	8*8	0...FFFFH	0 ... 3.999939	6.104e-5	-
LDP_TEMP_COLD_IP_FAC_TIA_COR	8	0...FEH	-48...142.5	0.75	°C
LDP_TEMP_HOT_IP_FAC_TIA_COR	8	0...FEH	-48...142.5	0.75	°C
Temperature Correction factor mapping (typical value=1)					
LC_TIA_CHA_DOWN_AS	1	0...1H	0 ... 1	1	-
Application System value selection for TIA_CHA_DOWN definition					
LC_TIA_CHRG_UP_CLC	1	0...1H	0 ... 1	1	-
Choice of TIA_CHRG_UP formula					
LC_ERR_TAM_SUPP	1	0... 1H	0 ... 1	1	-
Switch to suppress temperature susbstitute value selection					

4.43 Turbo charger rotational speed

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_TCHA_GRD	O/V	8000...7FFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm/segment
Gradient of the turbo charger speed of segment					
N_TCHA_MES_MMV	O/V	0...FFFFH	0...4E+5	6.10360876	rpm
Turbo charger rotational speed moving mean value (measured value)					
N_TCHA	O/V	0...FFFFH	0...4E+5	6.10360876	rpm
Turbo charger rotational speed					
N_CHA[NC_NR_TCHA]	O/V	0...FFFFH	0...4E+5	6.10360876	rpm
Turbo charger speed					
N_TCHA_MES	V	0...FFFFH	0...4E+5	6.10360876	rpm
Turbo charger rotational speed (measured value)					

Input data:

VP_N_TCHA	N_CHA_CLC[NC_NR_TCHA]	LC_TCHA_CONF	NC_N_TCHA_CONF
NC_NR_TCHA			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_N_TCHA_GRD	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Correlation constant for charger speed gradient filtering					
C_CRLC_N_TCHA_MES	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Correlation constant for smoothing of measured turbo charger speed					
C_DELTA_MAX_FAC_NEG_N_TCHA	1	0...FFFFH	0...99.9984741	0.00152588	%
factor for maximum of negative delta of N_TCHA					
C_DELTA_MAX_FAC_POS_N_TCHA	1	0...FFFFH	0...99.9984741	0.00152588	%
factor for maximum of positive delta of N_TCHA					
C_DELTA_MAX_OFS_NEG_N_TCHA	1	0...FFFFH	0...4E+5	6.10360876	rpm
offset for maximum of negative delta of N_TCHA					
C_DELTA_MAX_OFS_POS_N_TCHA	1	0...FFFFH	0...4E+5	6.10360876	rpm
offset for maximum of positive delta of N_TCHA					
C_N_TCHA_SENS_SRC	1	0...2H	0...2	1	-
Indicator which turbo charger speed is measured (0 - none, 1 - inlet branch 1, 2 - inlet branch 2)					
IP_N_TCHA_MES	4	0...FFFFH	0...4E+5	6.10360876	rpm
LDP_VP_N_TCHA_IP_N_TCHA_MES	4	0...7FFFH	0...4.99984741	1.52588E-4	V
Table for conversion of turbo charger rotational speed sensor voltage to speed signal					

4.43.1 FUNCTION PART: CHRG_SIGCVNTCHA0

Function Description

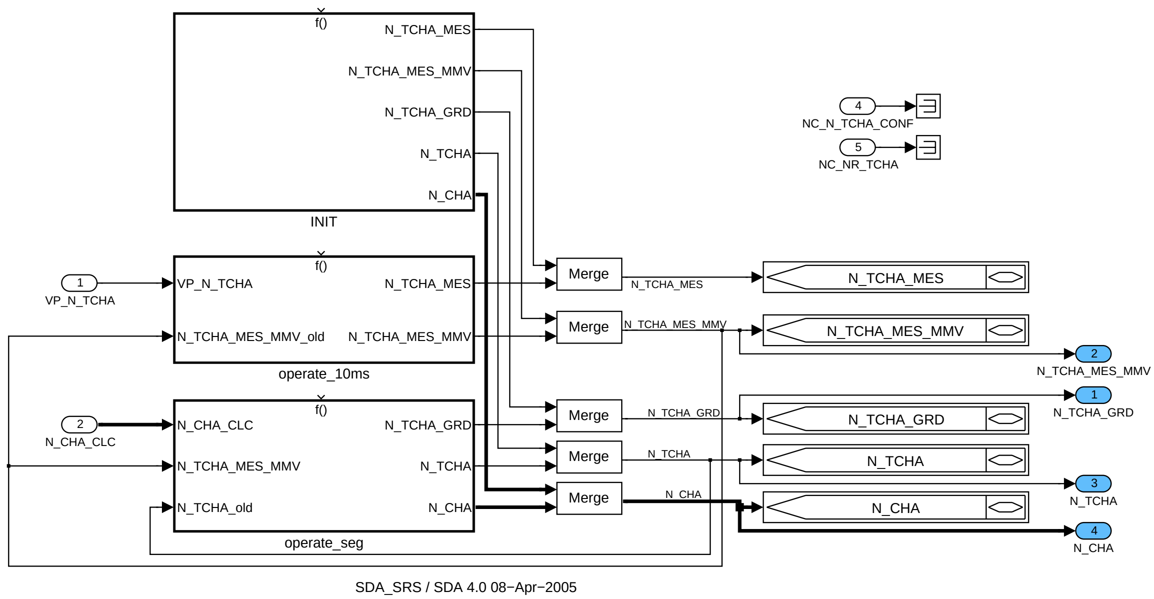


Figure 353 CHRG_SIGCVNTCHA0

4.43.1.1 SUBFUNCTION: INIT

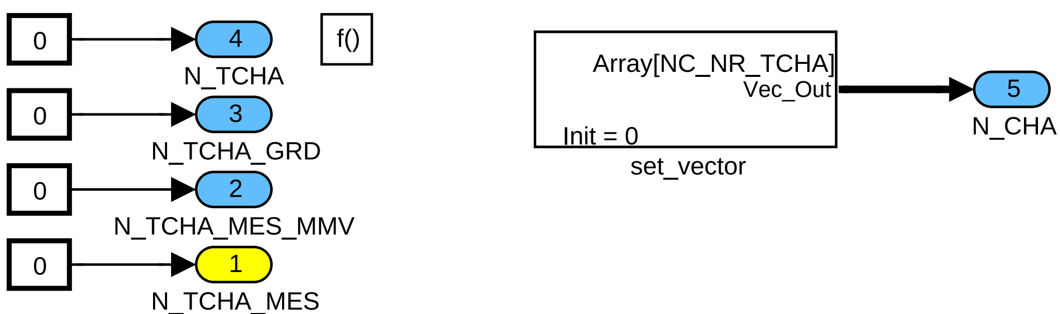


Figure 354 CHRG_SIGCVNTCHA0/ INIT

4.43.1.2 CHRG_SIGCVNTCHA0/OPERATE_10MS

Turbo Charger Rotational Speed (measured value)

This function calculates the measured turbo charger speed.

The module Turbo charger rational speed used the voltage of a turbo charger rational speed sensor. To avoid usage of signal noise and in case of short signal interruption the last value is used. To know if the last value is valid or not valid, the algorithm calculates the present step of the turbo charger sensor signal and compares it with the possible positve or negative deviations. Due to the fact of oscillations, for the value N_TCHA_MES_MMV a moving mean value calculation is used.

general specification

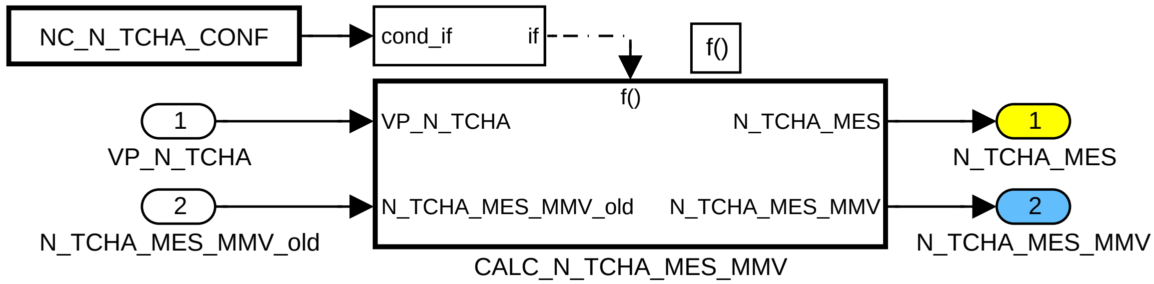


Figure 355 CHRG_SIGCVNTCHA0/ operate_10ms

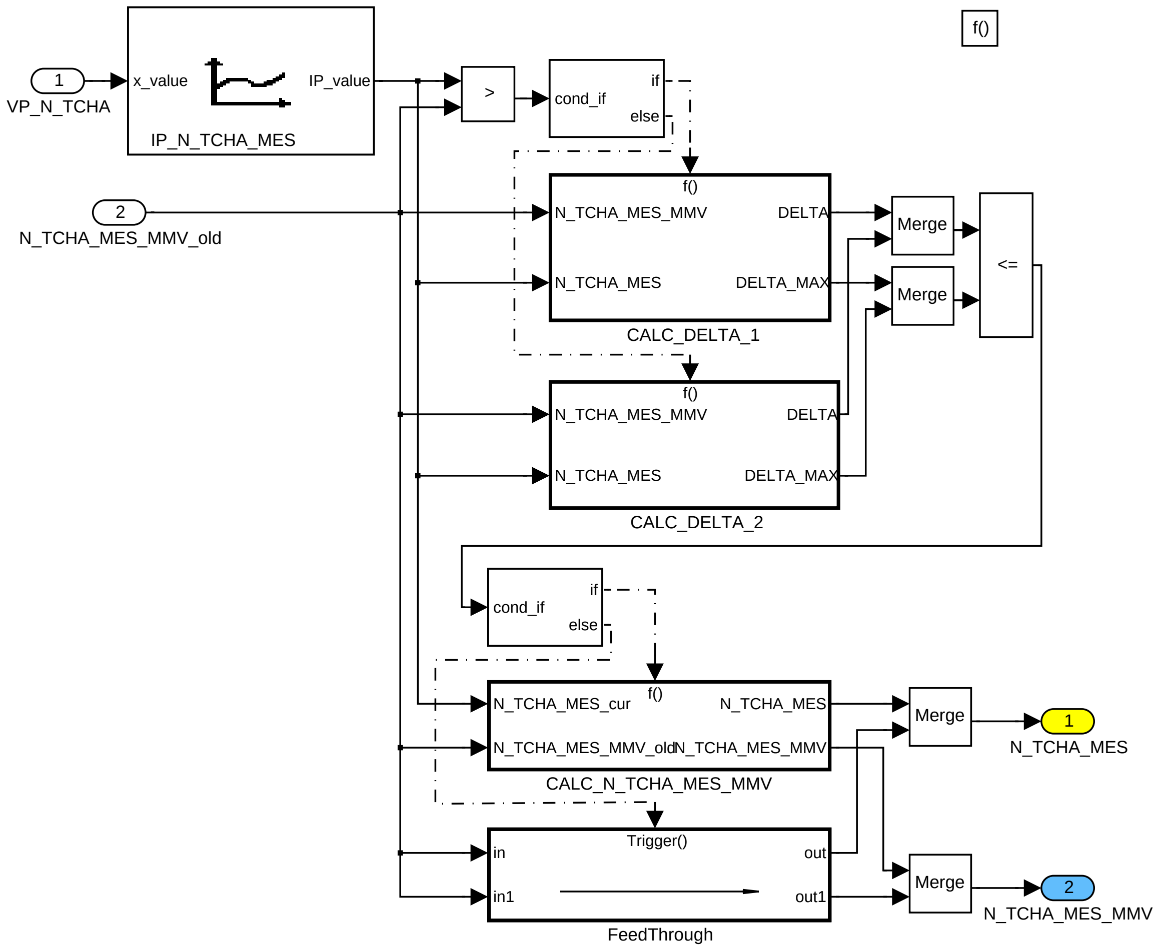


Figure 356 CHRG_SIGCVNTCHA0/ operate_10ms/ CALC_N_TCHA_MES_MMV

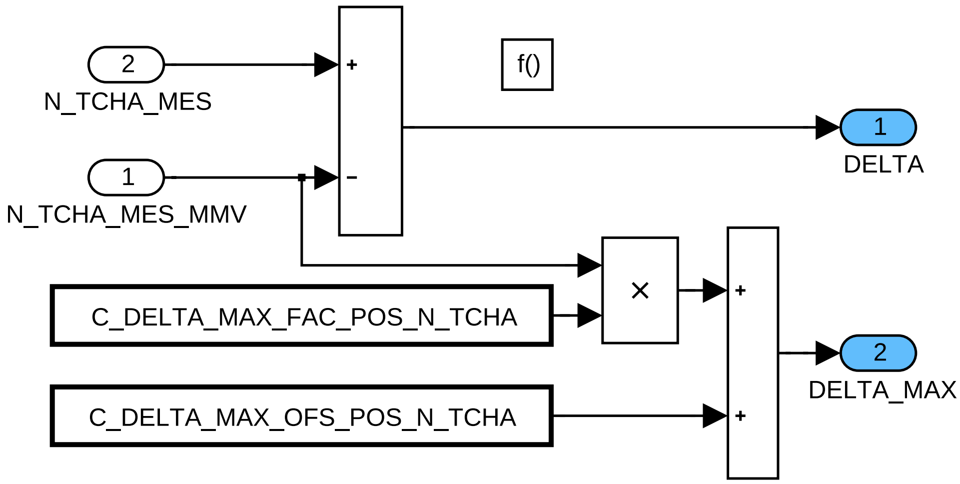


Figure 357 CHRG_SIGCVNTCHA0/ operate_10ms/ CALC_N_TCHA_MES_MMV/ CALC_DELTA_1

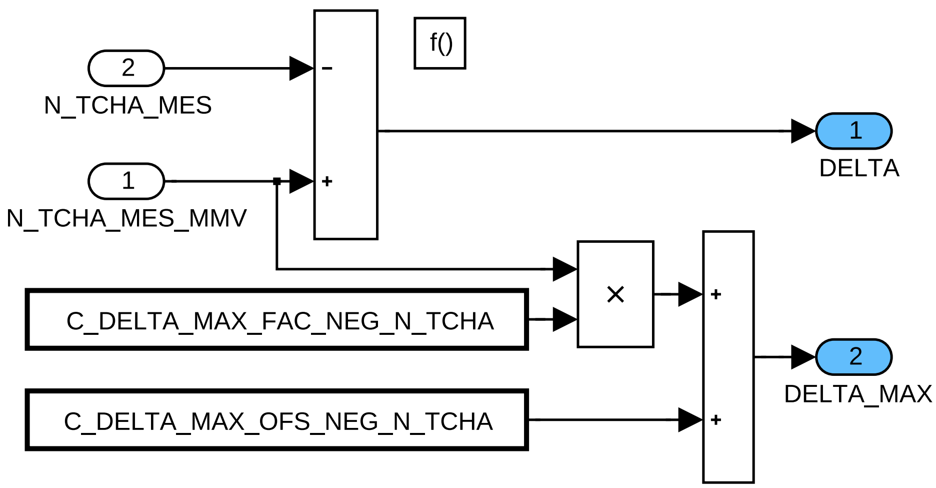


Figure 358 CHRG_SIGCVNTCHA0/ operate_10ms/ CALC_N_TCHA_MES_MMV/ CALC_DELTA_2

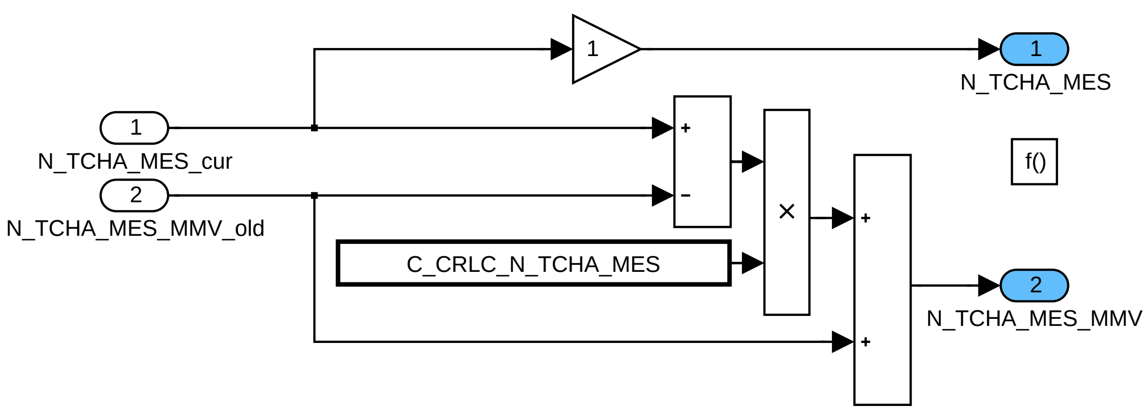


Figure 359 CHRG_SIGCVNTCHA0/ operate_10ms/ CALC_N_TCHA_MES_MMV/ CALC_N_TCHA_MES_MMV

4.43.1.3 CHRG_SIGCVNTCHA0/OPERATE_SEG

Turbo charger rotational speed – interface values

N_TCHA is the interface, which should be used by other modules. Via LC_N_CHA_CLC_ACT is selected, whether the model based charger speed N_CHA_CLC or the measured and filtered turbocharger speed N_TCHA_MES_MMV is used. Afterwards the gradient of the turbo charger speed over the segment time is calculated

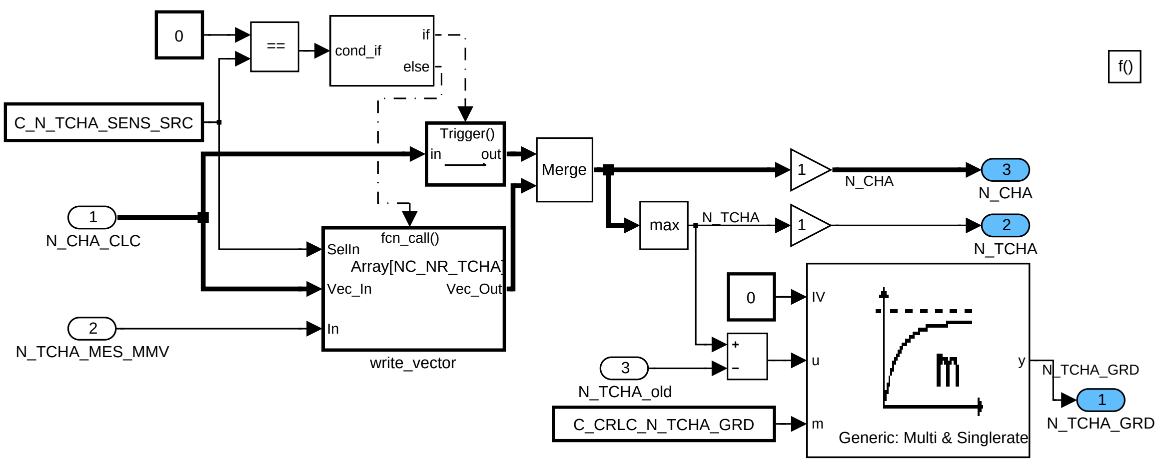


Figure 360 CHRG_SIGCVNTCHA0/ operate_seg

4.44 Recirculation Mass Air Flow determination: MAF_RCL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_RCL[NC_NR_TCHA]	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Recirculation air mass flow (inlet branch specific)					
PQ_RCL[NC_NR_TCHA]	O/V	0...FFFFH	0...15.9997559	2.44141E-4	-
Recirculation actuator pressure quotient (inlet branch specific)					
MAF_MDL_CON_1_RCL	O/V	0...FFFFH	0...0.0206019	3.14365E-7	s/m
sqrt(2x/((x-1)*RA*(TIA_RCL_UP+273))) with x = kappa = 1.4					
PRS_RCL_UP[NC_NR_TCHA]	O/V	0...7FFFH	0...2.71696E+3	0.08291752	hPa
Pressure upstream recirculation actuator (inlet branch specific)					
PRS_RCL_UP_MV	O/V	0...7FFFH	0...2.71696E+3	0.08291752	hPa
pressure upstream recirculation actuator (inlet branch average)					
MAF_RCL_MV	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Recirculation air mass flow (inlet branch average)					
AR_RED_RCL[NC_NR_TCHA]	V	0...FFFFH	0...58.592855	8.9407E-4	cm²
Reduced area for recirculation actuator (inlet branch specific)					

Input data:

NC_PSN_RCL_CTL	AR_RED_RCL_MDL[NC_NR_TCHA]	PSN_RCL[NC_NR_TCHA]	C_PSN_RCL_THD_DIG_CTL
LV_ERR_PLAUS_CLOSE_RCL	LC_ERR_PLAUS_RCL_ACR_SUPP	NC_RCL_UP_ICO	PRS_CHA_DOWN[NC_NR_TCHA]
PRS_CHRG_DOWN	PRS_CHA_UP[NC_NR_TCHA]	TIA_RCL_UP	LV_ERR_PLAUS_OPEN_RCL
LC_TCHA_CONF	TIA_THR	AMP	NC_NR_TCHA

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AR_RED_RCL_MAX	1	0...FFFFH	0...58.592855	8.9407E-4	cm²
Maximum value for reduced area of the recirculation actuator (used If Digital controller only)					
C_AR_RED_RCL_MIN	1	0...FFFFH	0...58.592855	8.9407E-4	cm²
Minimum value for reduced area of the recirculation actuator (used If Digital controller only)					
C_CRLC_MAF_RCL_CLC	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Filter coefficient for low pass filtering of MAF_RCL_CLC (typical value=1)					
C_MAF_RCL_AS	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
MAF_RCL value fixed via Application System					
LC_AR_RED_RCL	1	0...1H	0...1	1	-
Selection switch for AR_RED_RCL calculation (used if Digital controller only)					
LC_MAF_RCL_AS	1	0...1H	0...1	1	-
Selection for MAF_RCL value fixed via Application sytem					
IP_AR_RED_RCL	12	0...FFFFH	0...58.592855	8.9407E-4	cm²
LDP_PSN_RCL_IP_AR_RED_RCL	12	0...FFFFH	0...99.9984741	0.00152588	%
Reduced area of the recirculation actuator					
IP_MAF_MDL_CON_1_RCL	8	0...FFFFH	0...0.0206019	3.14365E-7	s/m
LDP_TIA_RCL_UP_IP_MAF_MDL_CON_1	8	0...FFFFH	-48...335.994141	0.00585938	°C
Temperature influence on air flow function at the recirculation actuator					
IP_PSI_RCL	16	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_PQ_RCL_IP_PSI_RCL	16	0...FFFFH	0...15.9997559	2.44141E-4	-
psi-function for RCL					

4.44.1 CHRG_MDLADMAFRC0

General information:

The objective is to determine the air mass flow through the Recirculation actuator.

The air mass flow is function of the pressure upstream the recirculation actuator and this pressure is defined according the position of the recirculation actuator (see NC_RCL_UP_ICO value).

A digital actuator, depending on its design, can show a true digital or a pseudo-continuous behaviour. Both possibilities are taken into account.

Function Description

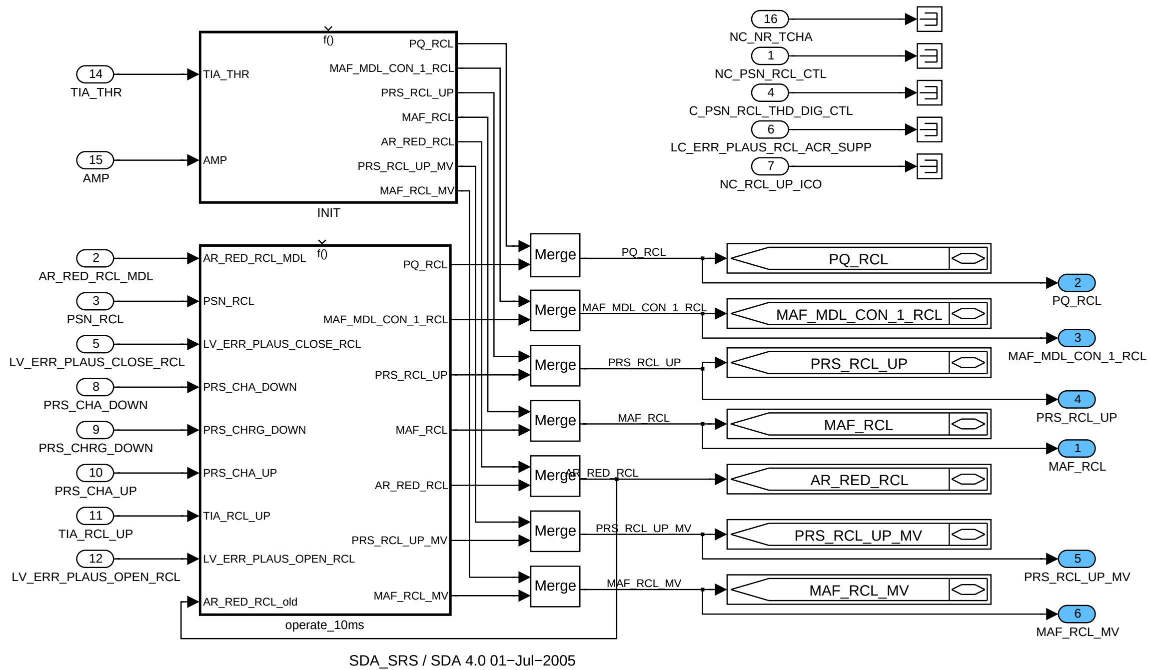


Figure 361 CHRG_MDLADMAFRC0

4.44.1.1 SUBFUNCTION: INIT

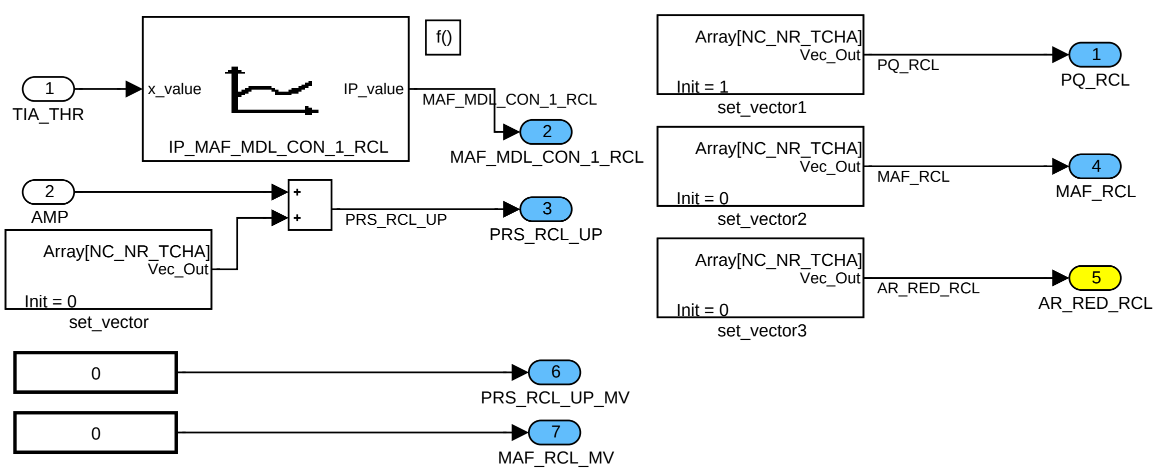


Figure 362 CHRG_MDLADMAFRC0/ INIT

4.44.1.2 SUBFUNCTION: operate_10ms

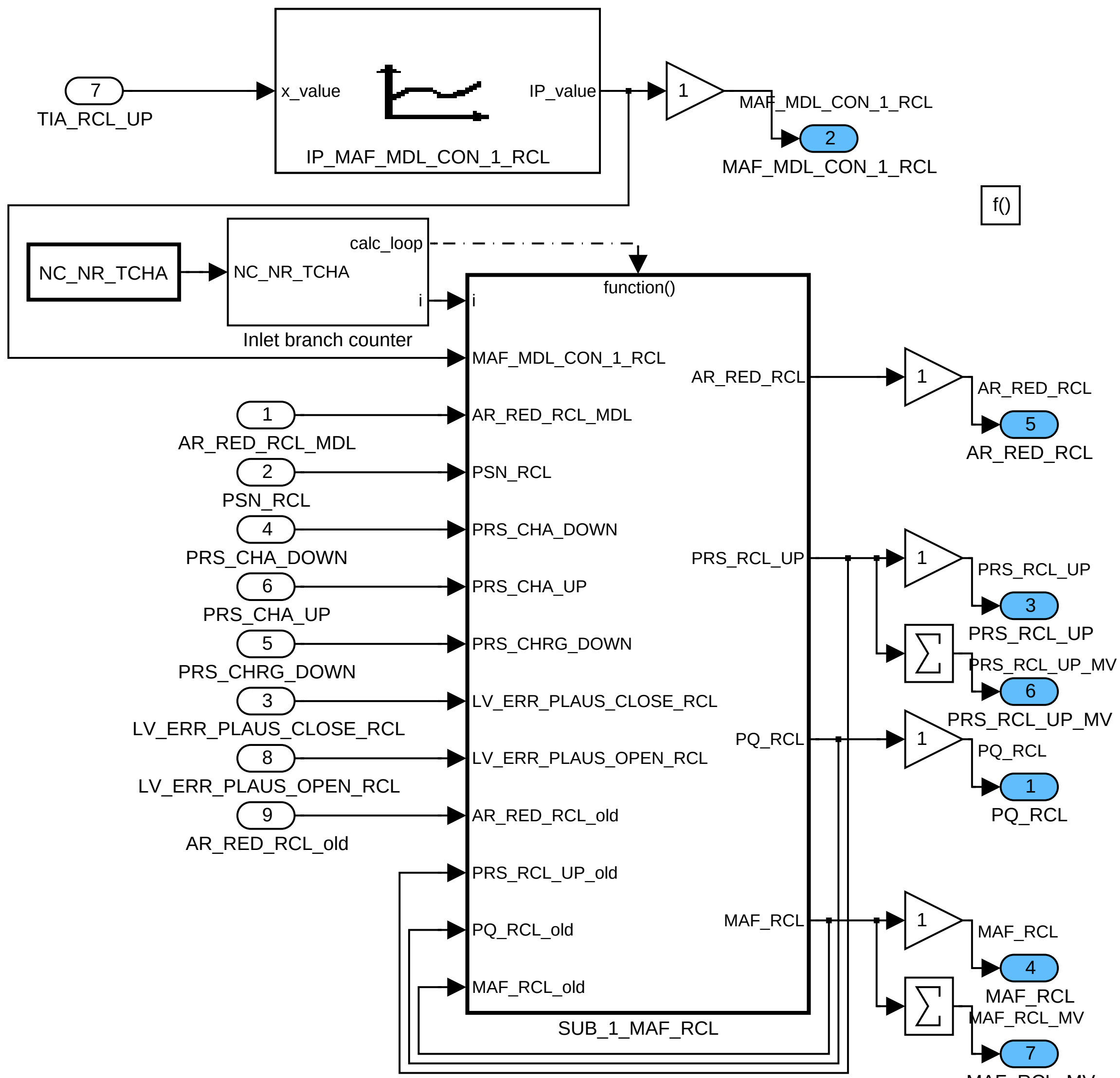


Figure 363 CHRG_MDLADMAFRC0/ operate_10ms

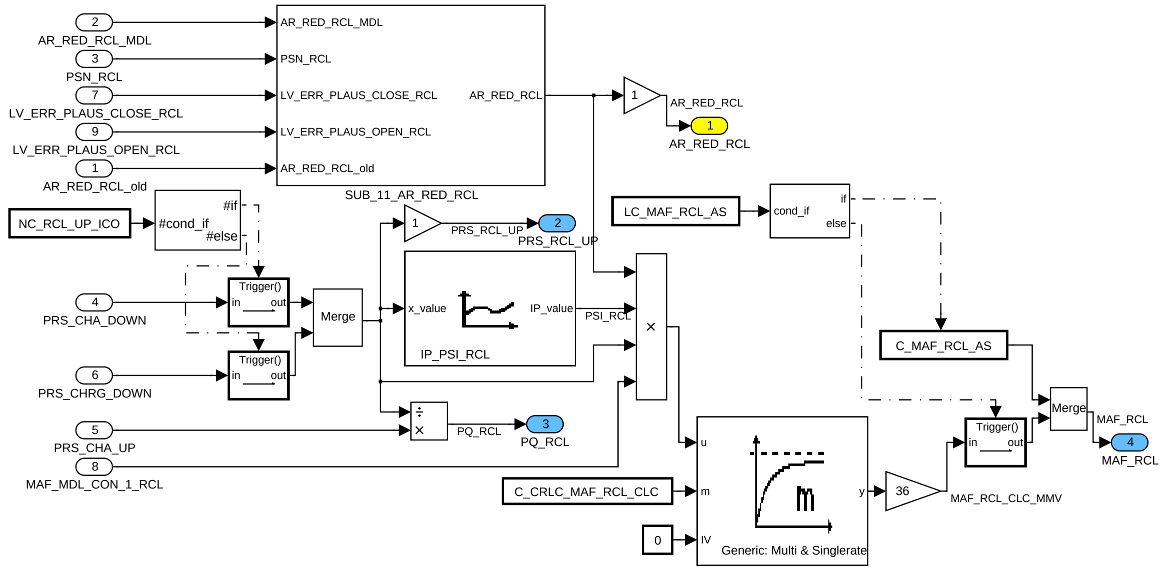


Figure 364 CHRG_MDLADMAFRC0/ operate_10ms/ SUB_1_MAF_RCL/ SUB_1_MAF_RCL

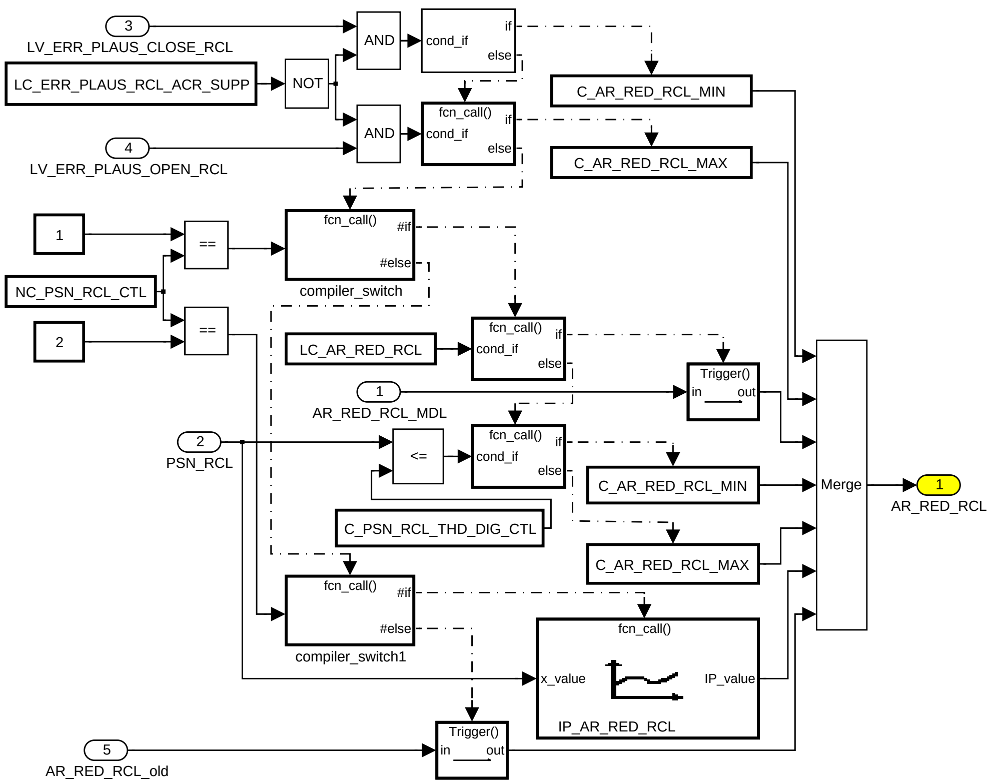


Figure 365 CHRG_MDLADMAFRC0/ operate_10ms/ SUB_1_MAF_RCL/ SUB_1_MAF_RCL/ SUB_11_AR_RED_RCL

4.45 Recirculation Actuator Pressure

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PRS_EPC_RCL_DOWN	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure downstream RCL EPC					
PRS_VAC_ACR_DIF_MAP	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Pressure difference between PRS_EPC_RCL_UP_ACT and MAP					
PRS_VAC_ACR_DIF_AMP	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Pressure difference between PRS_EPC_RCL_UP_ACT and AMP					
PRS_MAP_RCL_MMV_DIF	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Difference between MAP_RCL_MMV and PRS_EPC_RCL_UP_ACT for CRLC_MAP_RCL determinaiton					
PRS_EPC_RCL_DOWN_BAS	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Raw value of PRS_EPC_RCL_DOWN					
CRLC_MAP_RCL	V	0...FFH	0...0.99609375	0.00390625	-
Variation of PRS_EPC_RCL_DOWN_BAS					
MAP_RCL_MMV	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Variation of PRS_EPC_RCL_DOWN_BAS					
PRS_EPC_RCL_UP_ACT_DIF	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Pressure upstream RCL EPC variation					
PRS_EPC_RCL_UP_ACT	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure upstream the RCL EPC: PRS_EPC_RCL_DOWN = PRS_EPC_RCL_UP_ACT when EPC activated					
PRS_EPC_RCL_UP_PAS	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure upstream the RCL EPC: PRS_EPC_RCL_DOWN = PRS_EPC_RCL_UP_PAS when EPC passive					
CTR_TRAN_RCL	V	0...2H	0...2	1	-
Count of number of transition PSN_RCL_SP = 0 to 99.998474%					
CTR_LEAK_RCL	V	0...FFFFH	0...6.5535E+4	1	-
Counter for PRS_EPC_RCL_UP_ACT_DIF determination in case of leak only					

Input data:

AMP	MAP	PRS_CHRG_DOWN	PRS_CHA_DOWN[NC_NR_TCHA]
PSN_RCL_SP	LV_ERR_EL_RCL_ACR	ERR_SYM_EL_RCL_ACR	LC_TCHA_CONF
TAM	NC_NR_TCHA		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_MAP_RCL_RISE	1	0...FFH	0...0.99609375	0.00390625	-
Filter for MAP_RCL_MMV determination in case of rising MAP values					
C_CRLC_PRS_EPC_RCL_DOWN_FALL	1	0...FFH	0...0.99609375	0.00390625	-
Filter for PRS_EPC_RCL_DOWN determination in case of falling PRS_EPC_RCL_DOWN values					
C_CRLC_PRS_EPC_RCL_DOWN_RISE	1	0...FFH	0...0.99609375	0.00390625	-
Filter for PRS_EPC_RCL_DOWN determination in case of rising PRS_EPC_RCL_DOWN values					
C_CTR_LEAK_RCL	1	0...FFFFH	0...6.5535E+4	1	-
Counter for PRS_EPC_RCL_UP_ACT_DIF detemination in case of leak only					
C_FAC_PRS_RCL_UP_ACT_TRAN	1	0...FFH	0...1.9921875	0.0078125	-
Factor on PRS_EPC_RCL_UP_ACT_TRAN (every 10ms) for 1st transition PSN_RCL_SP=0 to 99.998474%					
C_PRS_EPC_RCL_DOWN_AS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Application System value for PRS_EPC_RCL_DOWN					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PRS_RCL_DIF_THD	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Threshold pressure value used for pressure determination in CHRG Aggregate					
C_SRC_PRS_EPC_RCL_UP_PAS	1	1H 2H 3H 4H 5H	AMP PRS_CHRG_DOWN PRS_CHA_DOWN_1 MAP PRS_CHA_DOWN_2	1	-
Source of pressure on passive RCL EPC input (PRS_EPC_RCL_UP_PAS value)					
LC_ERR_EL_RCL_ACR_SUPP	1	0...1H	0...1	1	-
Suppression of LV_ERR_EL_RCL_ACR_SUPP effect on PRS_EPC_RCL_DOWN_BAS					
LC_PRS_EPC_RCL_DOWN_AS	1	0...1H	0...1	1	-
Selection for PRS_EPC_RCL_DOWN fixed via Application System					
IP_CRLC_MAP_RCL_FALL	6	0...FFH	0...0.99609375	0.00390625	-
LDP_PRS_MAP_RCL_MMV_DIF_IP_CRLC	6	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Filter for MAP_RCL_MMV determination in case of falling MAP values					
IP_FAC_PRS_EPC_RCL_UP_LEAK_MAP	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_PRS_DIF_1_CHRG	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Multiplicative correction of PRS_EPC_RCL_UP_ACT variation (due to leakage only) versus AMP					
IP_FAC_PRS_EPC_RCL_UP_LEAK_TAM	6	0...FFH	0...1.9921875	0.0078125	-
LDPM_TAM_1_CHRG	6	0...FEH	-48...142.5	0.75	°C
Multiplicative correction of PRS_EPC_RCL_UP_ACT variation (due to leakage only) versus TAM					
IP_FAC_PRS_EPC_RCL_UP_TRAN_TAM	6	0...FFH	0...1.9921875	0.0078125	-
LDPM_TAM_1_CHRG	6	0...FEH	-48...142.5	0.75	°C
Multiplicative correction of PRS_EPC_RCL_UP_ACT variation (if PSN_RCL_SP=0 to 99.998474%) versus TAM					
IP_PRS_EPC_RCL_UP_LEAK_MAP	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_PRS_DIF_1_CHRG	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_UP_ACT variation (every C_CTR_LEAK_RCL * 10ms) due to leakage only: MAP dependent term					
IP_PRS_EPC_RCL_UP_TRAN	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_PRS_DIF_1_CHRG	8	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_UP_ACT variation (10ms) for the 2nd and next transitions PSN_RCL_SP=0 to 99.998474%					

4.45.1 CHRG_MDLADPRCAC0

Description:

PRS_EPC_RCL_UP_ACT consists of modeling the pressure variation due to:

- leakage in the complete vacuum system (vacuum reservoir, connecting hoses and vacuum valve). PRS_EPC_RCL_UP_ACT will increase [or decrease] slowly depending on the pressure difference between the Manifold and PRS_EPC_RCL_UP_ACT.

- EPC activation: each activation of the RCL valve leads to vacuum consumption (independently of the effective opening of the RCL valve).

The Electro pneumatic converter (EPC) is connected to three pressures:

- the outlet pressure downstream EPC:

EPC: PRS_EPC_RCL_DOWN

- the 2 inlet pressures upstream EPC:

PRS_EPC_RCL_UP_PAS and PRS_EPC_RCL_UP_ACT defined as follows:

. PRS_EPC_RCL_DOWN equal PRS_EPC_RCL_UP_PAS when EPC is passive,

. PRS_EPC_RCL_DOWN equal PRS_EPC_RCL_UP_ACT when EPC is active.

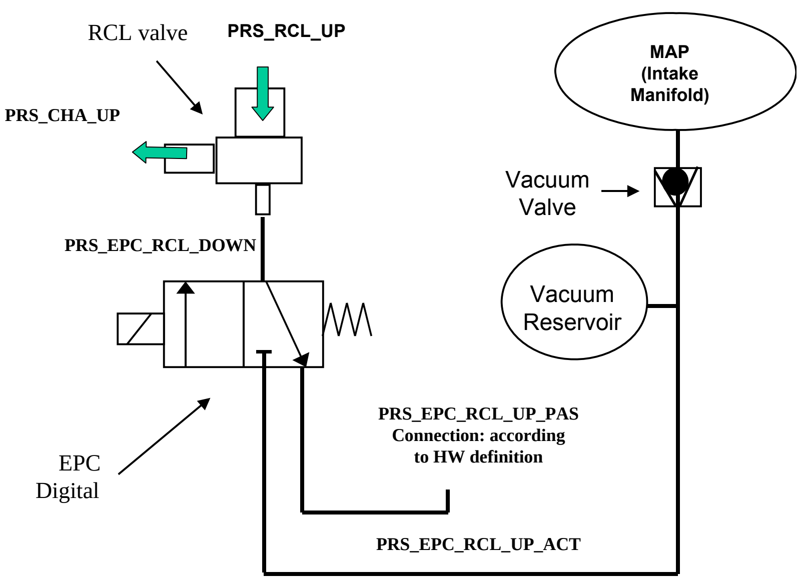
Hardware configuration:

. PRS_EPC_RCL_UP_PAS is configurable: it can be connected to the ambiance (AMP) or to the pressure upstream the throttle (PRS_CHRG_DOWN) or to the pressure downstream the charger (PRS_CHA_DOWN) or to manifold pressure (MAP).

. PRS_EPC_RCL_UP_ACT must be connected to Manifold (Vacuum) (MAP)

The RCL valve can be activated by the mean of vacuum pressure (PRS_EPC_RCL_DOWN). When the EPC is passive (no activated) ambient pressure or overpressure (according to hardware definition) is applied on the RCL valve: The RCL valve remains closed. When the EPC is activated the vacuum pressure is applied on the RCL valve. The source of vacuum is the pressure from the Intake Manifold. If this vacuum pressure decreases sufficiently (example: in Idle state) the Vacuum Valve opens and the vacuum pressure spreads in the Vacuum reservoir. The Vacuum valve closes as soon as the pressure increases in the Manifold. In this way, the Vacuum pressure is stored in the Vacuum Reservoir and the valve can be activated even if the pressure in the Manifold increased in the mean time.

example of RCL hardware



Function Description

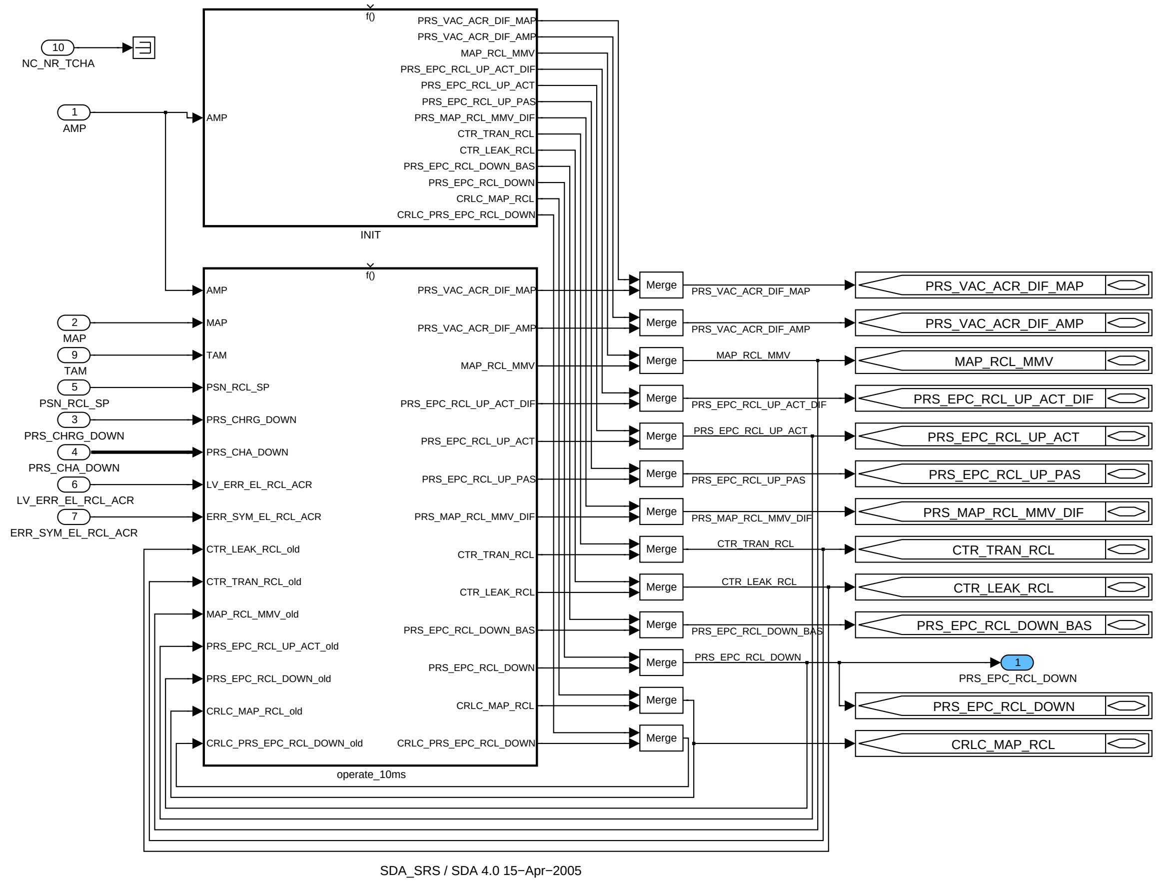


Figure 366 CHRG_MDLADPRCAC0

4.45.1.1 SUBFUNCTION: INIT

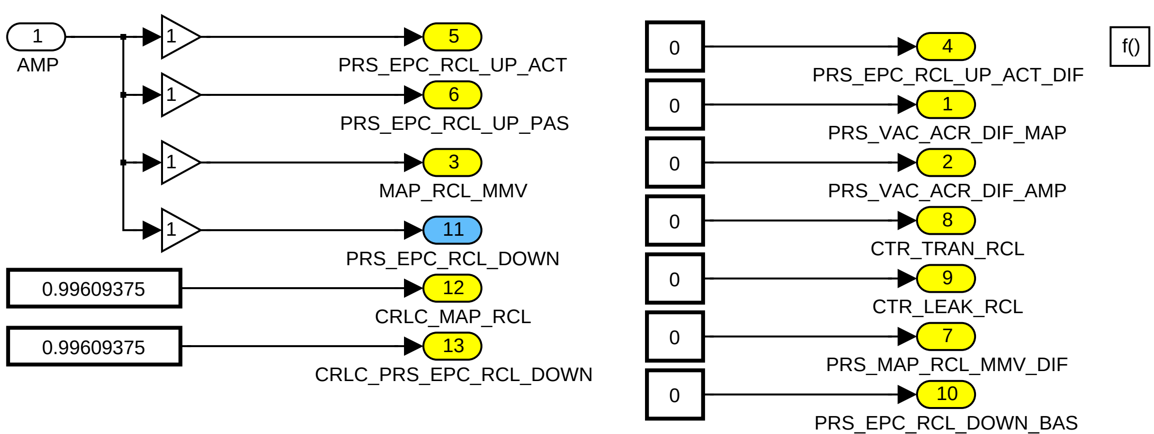


Figure 367 CHRG_MDLADPRCAC0/ INIT

4.45.1.2 SUBFUNCTION: operate_10ms

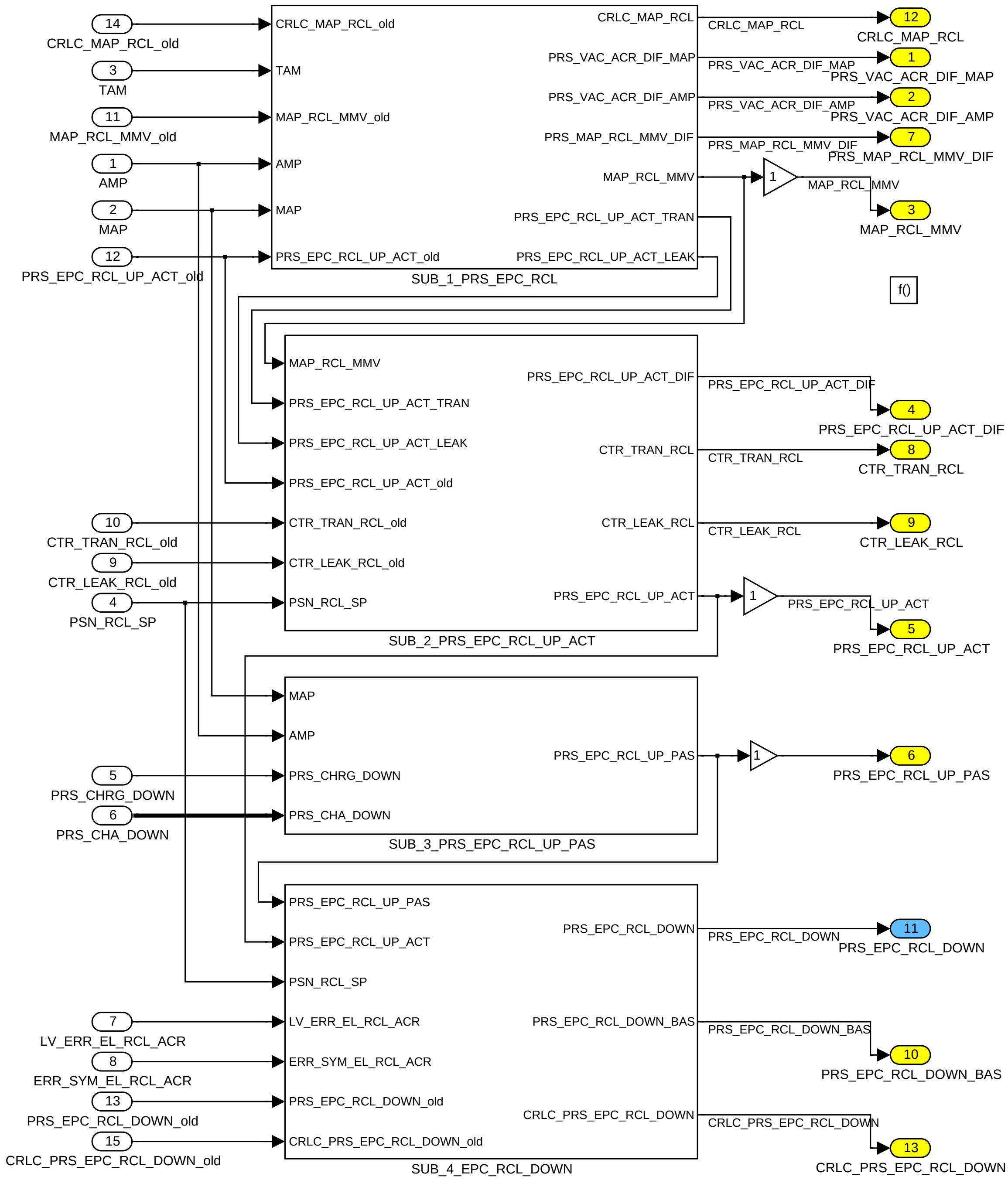


Figure 368 CHRG_MDLADPRCAC0/ operate_10ms

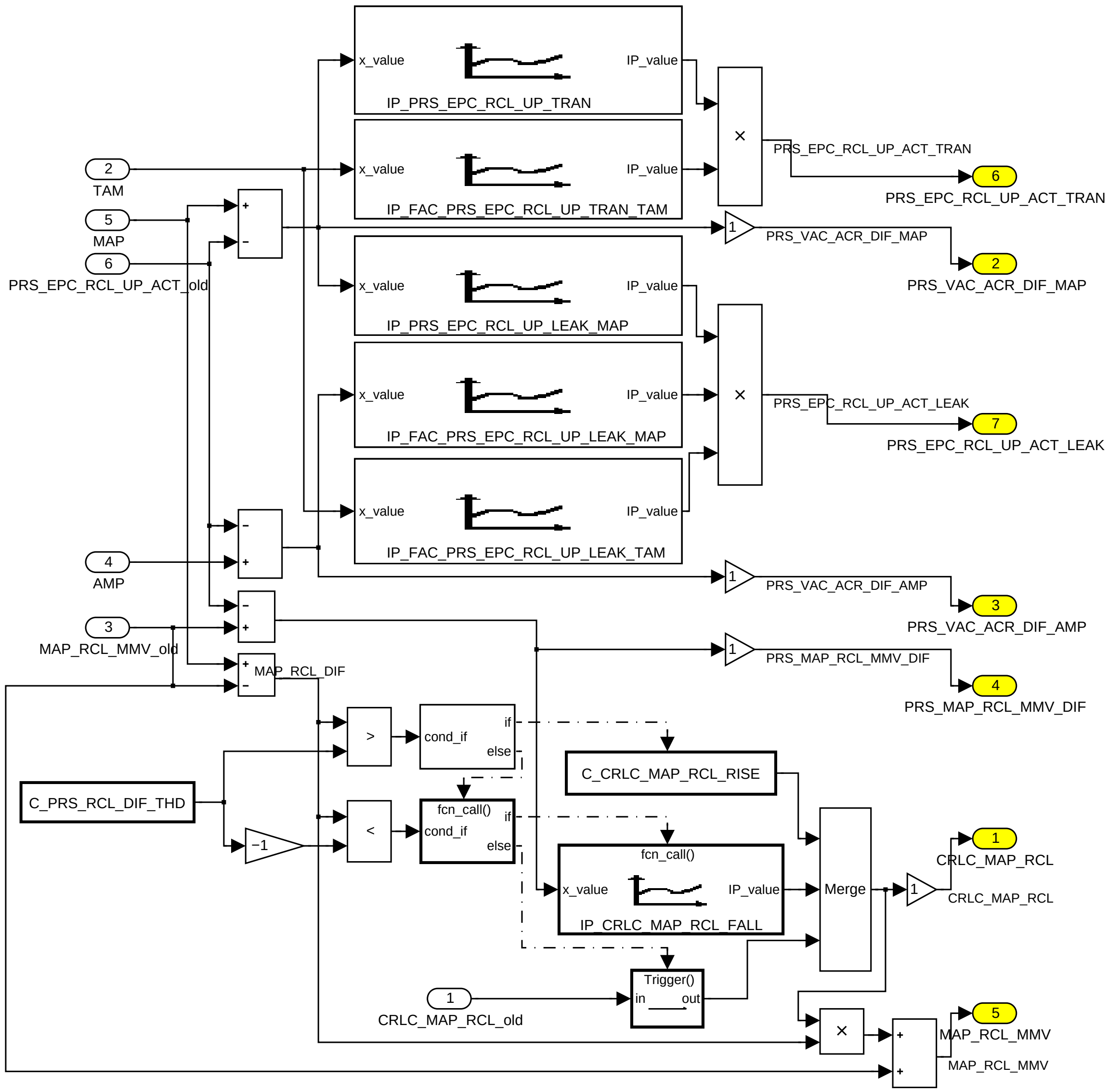


Figure 369 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_1_PRS_EPC_RCL

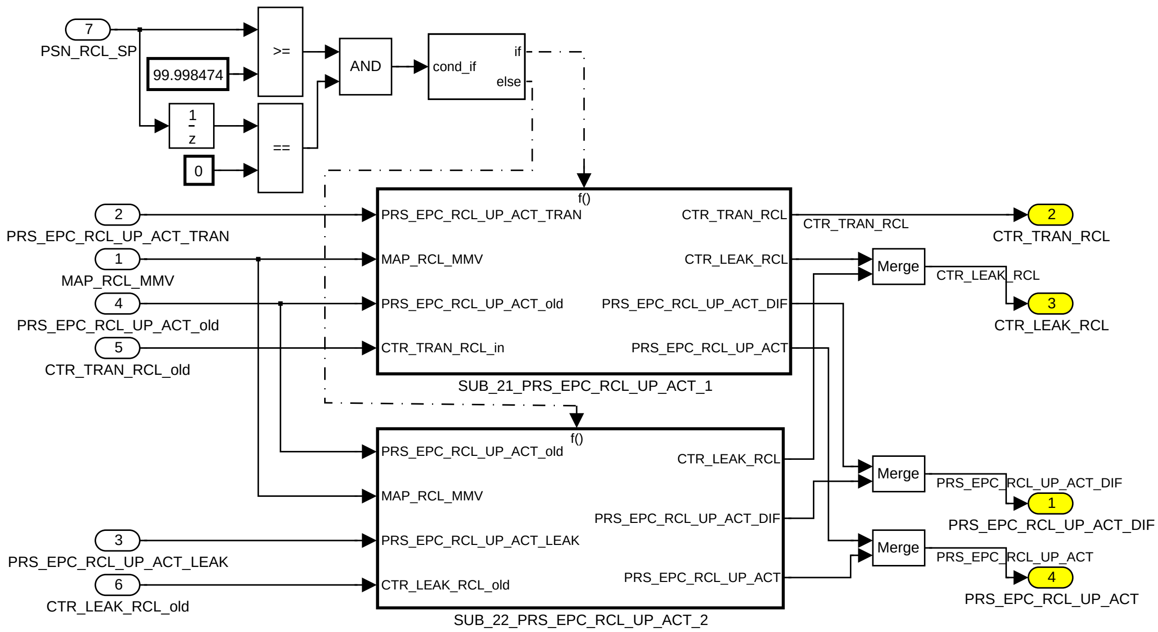


Figure 370 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_2_PRS_EPC_RCL_UP_ACT

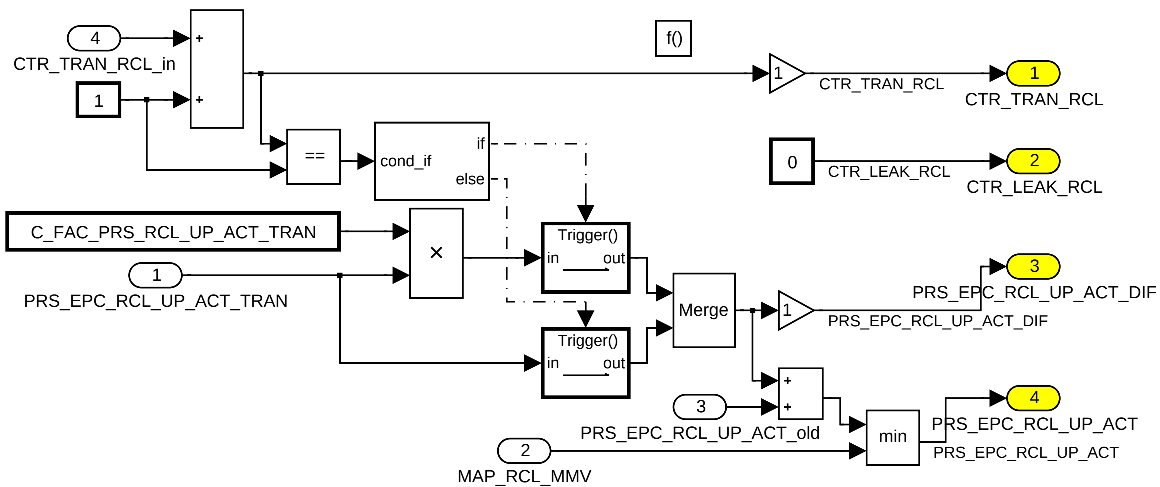


Figure 371 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_2_PRS_EPC_RCL_UP_ACT/ SUB_21_PRS_EPC_RCL_UP_ACT_1

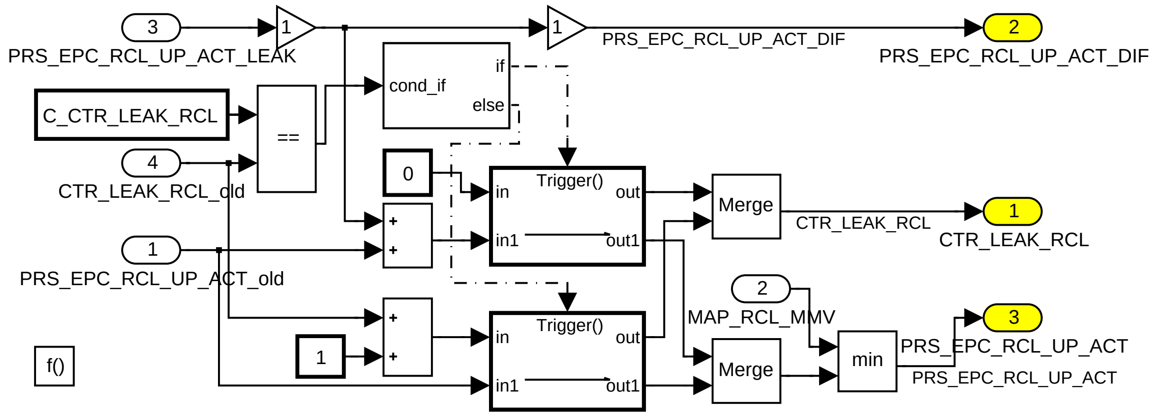


Figure 372 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_2_PRS_EPC_RCL_UP_ACT/ SUB_22_PRS_EPC_RCL_UP_ACT_2

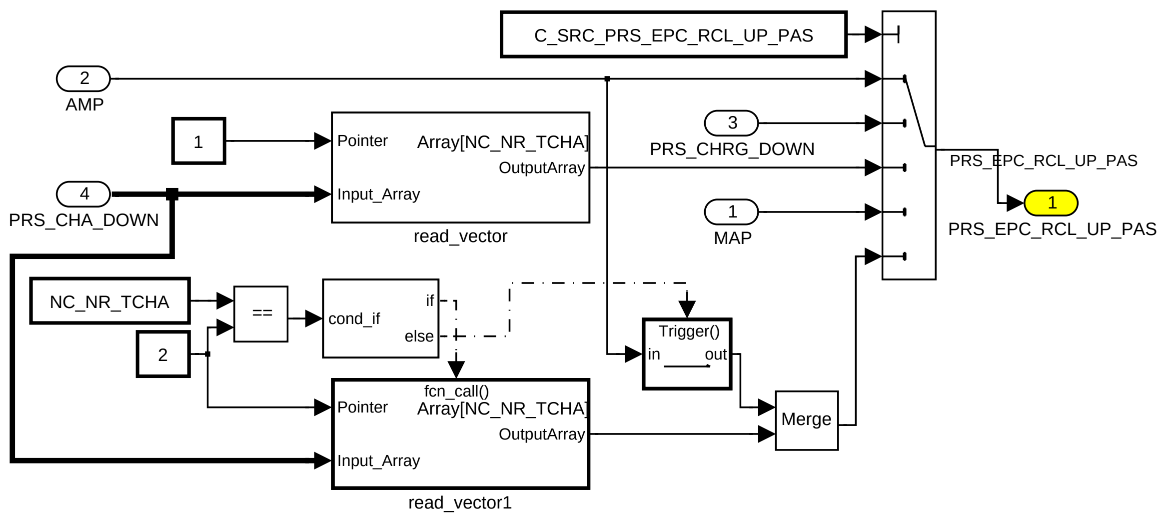


Figure 373 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_3_PRS_EPC_RCL_UP_PAS

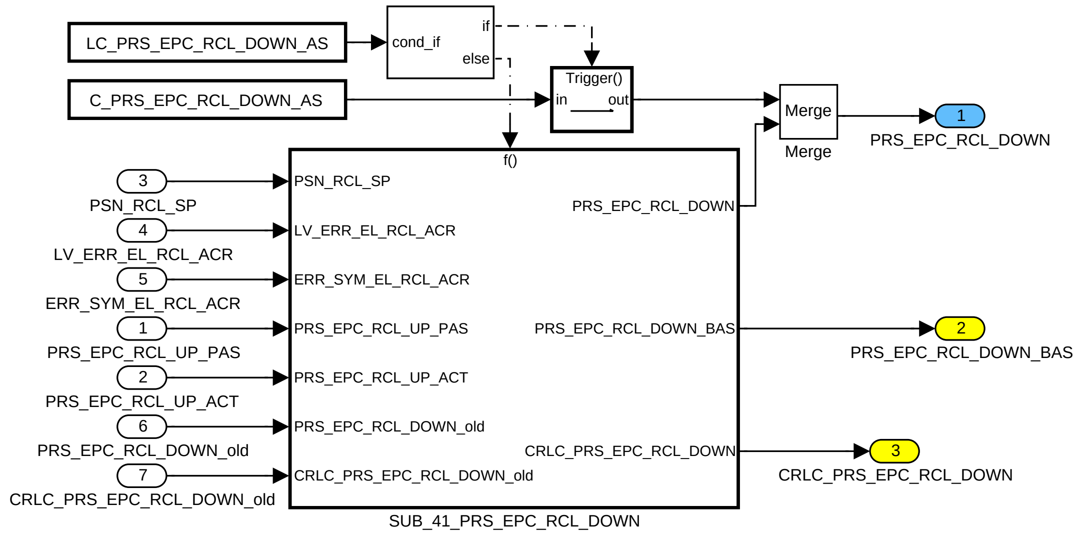


Figure 374 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_4_EPC_RCL_DOWN

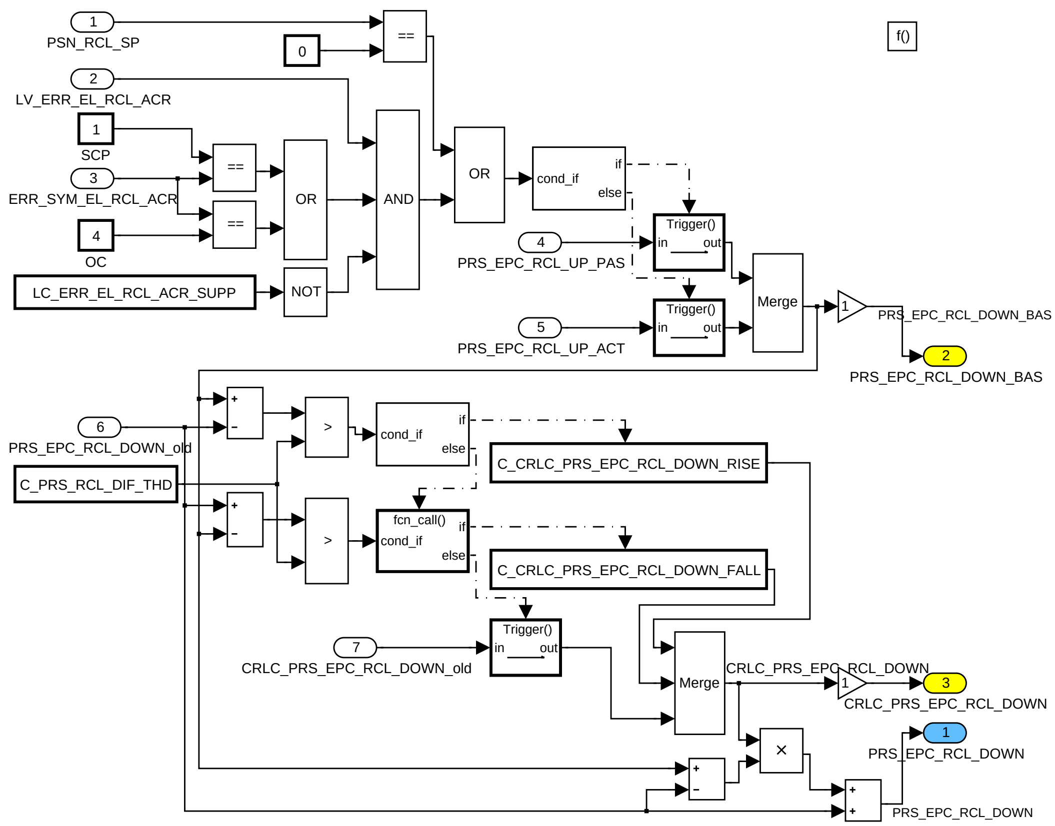


Figure 375 CHRG_MDLADPRCAC0/ operate_10ms/ SUB_4_EPC_RCL_DOWN/ SUB_41_PRS_EPC_RCL_DOWN

4.46 Intercooler Variables

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PRS_LOSS_ICO [NC_NR_TCHA]	V/O	0 ... FFFFH	0 ... 5434	0.083	hPa
Pressure drop through the Intercooler (Inlet branch specific)					
TIA_ICO_UP [NC_NR_TCHA]	V/O	0...FFFFH	-48 ... 335.995	0.75 / 128	°C
Air Temperature upstream intercooler					
TIA_ICO_UP_BAS [NC_NR_TCHA]	V	0...FFFFH	-48 ... 335.995	0.75 / 128	°C
Raw value of Air Temperature upstream intercooler					
TIA_ICO_UP_DIF	-	8000...7FFFH	-192 ... 192	0.75 / 128	°C
Upstream intercooler temperature variation					
TIA_ICO_UP_DIF_LOAD	-	0 ... FEH	0 ... 190.5	0.75	°C
Basic term (load dependant) of TIA_ICO_UP_BAS_DIF					
FAC_TIA_ICO_UP_DIF	-	0...FFH	0...3.984	0.0157	-
TIA_ICO_UP_BAS_DIF multiplicative correction versus TEMP_ICO_COOL					
TIA_ICO_UP_BAS_DIF	-	0 ... FEH	0 ... 190.5	0.75	°C
Temperature difference between upstream and downstream Intercooler					
MAF_ICO [NC_NR_TCHA]	-	0 ... FFFFH	0 ... 2047.96875	0.03125	Kg/h
Air mass flow through the Intercooler					
PRS_LOSS_ICO_MV	V/O	0 ... FFFFH	0 ... 5434	0.083	hPa
Pressure drop through the Intercooler (Inlet branch average)					

Input data:

MAF_THR_TCHA	TIA_THR	LC_TCHA_CONF	NC_RCL_UP_ICO
MAF_CHA [NC_NR_TCHA]	TAM	FAC_TIA_COR_TEMP	VS
NC_NR_TCHA	MAF_KGH_AIC_TCHA [NC_NR_TCHA]		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TIA_ICO_UP_DIF	8 * 6	0...FEH	0 ... 190.5	0.75	°C
LDPM_MAF_ICO_1_CHRG	8	0...FFFFH	0 ... 2047.96875	0.03125	kg/h
LDP_VS_IP_TIA_ICO_UP_DIF	6	0...FFH	0 ... 255	1	km/h
Basic term for TIA_ICO_UP_BAS_DIF determination (positive values)					
IP_FAC_TIA_ICO_UP_DIF_TEMP_ICO	1*8	0...FFH	0...3.984	0.0157	-
LDP_TEMP_ICO_COOL_IP_FAC_TIA	8	0...FEH	- 48 ... 142.5	0.75	°C
Multiplicative correction versus TEMP_ICO_COOL for TIA_ICO_UP_BAS_DIF determination					
IP_CRLC_TIA_ICO_UP	1*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDP_TIA_CHA_DIF_IP_CRLC_TIA_ICO	6	0...FFFFH	-192 ... 192	0.75 / 128	°C
Filter for TIA_ICO_UP calculation					
IP_PRS_LOSS_ICO	8*6	0...FFFFH	0...5434	0.083	hPa
LDPM_MAF_ICO_1_CHRG	8	0...FFFFH	0 ... 2047.96875	0.03125	kg/h
LDPM_TAM_1_CHRG	6	0...FEH	-48...142.5	0.75	°C
Air Pressure Loss throught the Intercooler (positive values)					
C_TEMP_ICO_COOL	1	0...FEH	- 48...142.5	0.75	°C
Manual value for Intercooler coolant fluid temperature reference					
LC_TEMP_ICO_COOL	1	0...1H	0...1	1	-
Intercooler configuration: 0 = air/air Intercooler, 1 = air/water Intercooler					

4.46.1 CHRG_MDLADICO0

The pressure loss in the intercooler is defined as a function of:

_ the flow passing through the Intercooler (MAF_ICO) (1^{st} order)

_ ambiant air temperature (TAM) (2nd order)

The cooling fluid temperature reference can be fixed by the mean of the switch LC_TEMP_ICO_COOL.

Function Description

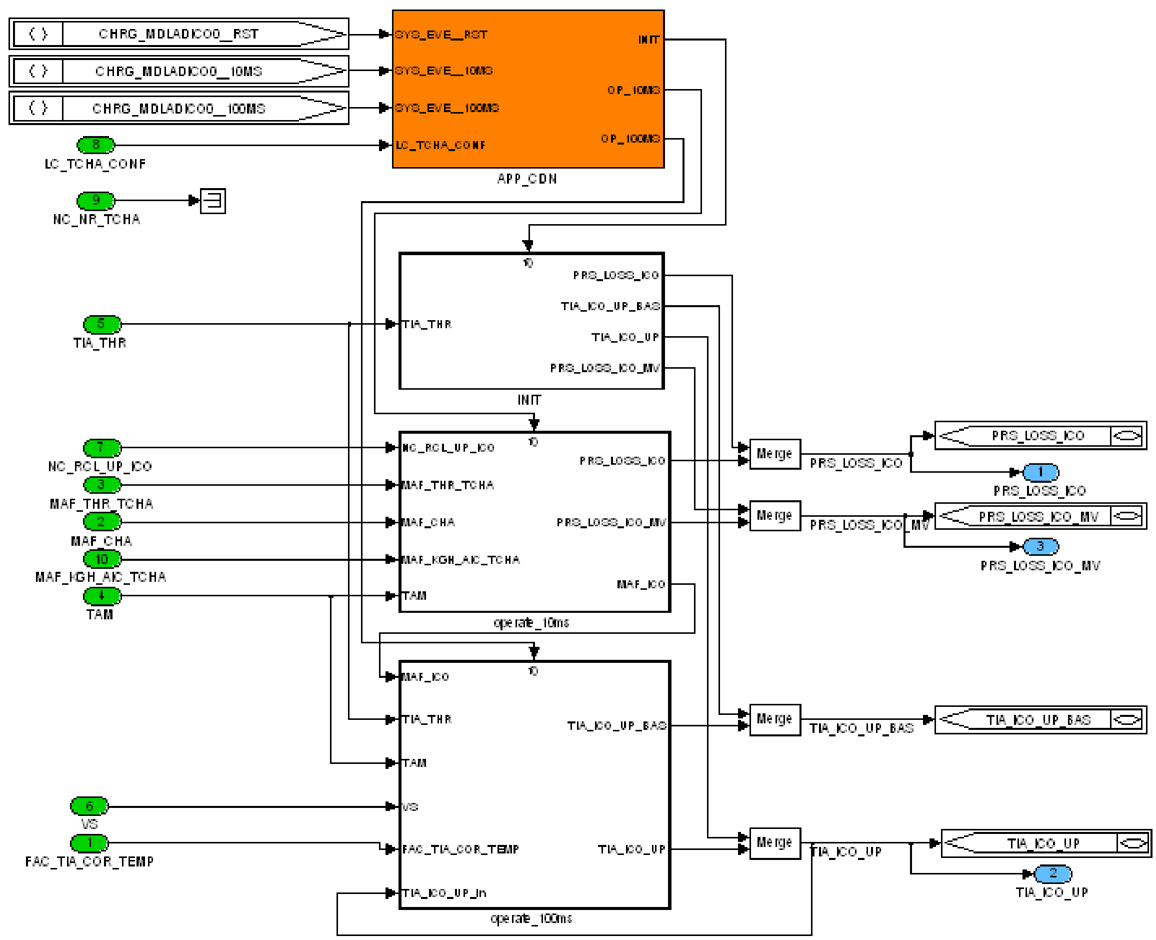


Figure 376 CHRG_MDLADICO0

4.46.1.1 SUBFUNCTION: INIT

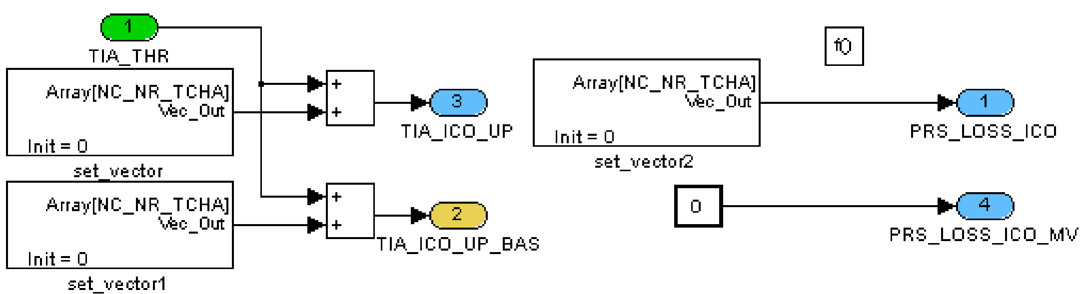


Figure 377 CHRG_MDLADICO0/ INIT

4.46.1.2 SUBFUNCTION: operate_100ms

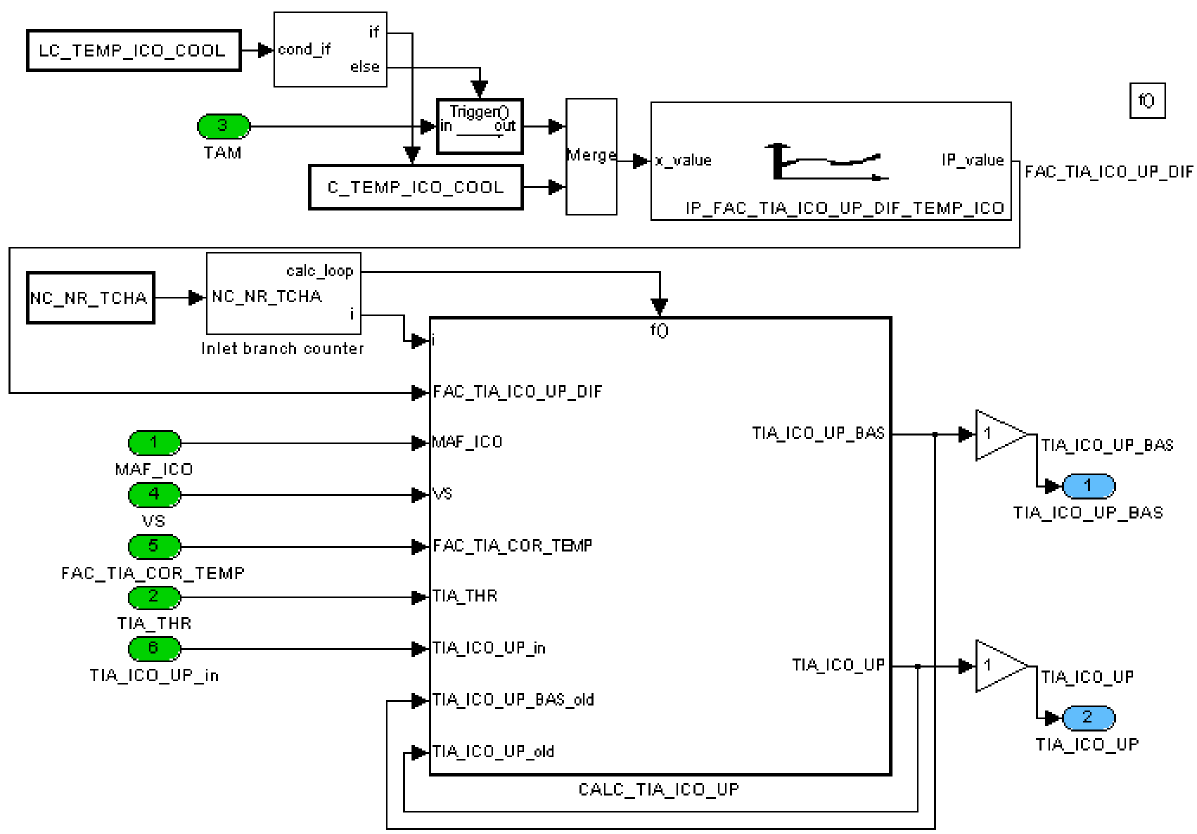


Figure 378 CHRG_MDLADICO0/ operate_100ms

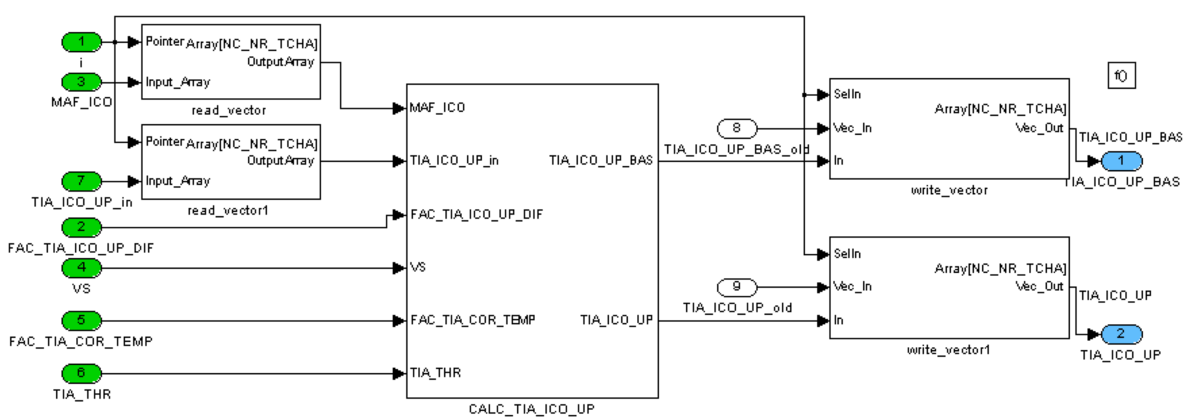


Figure 379 CHRG_MDLADICO0/ operate_100ms/ CALC_TIA_ICO_UP

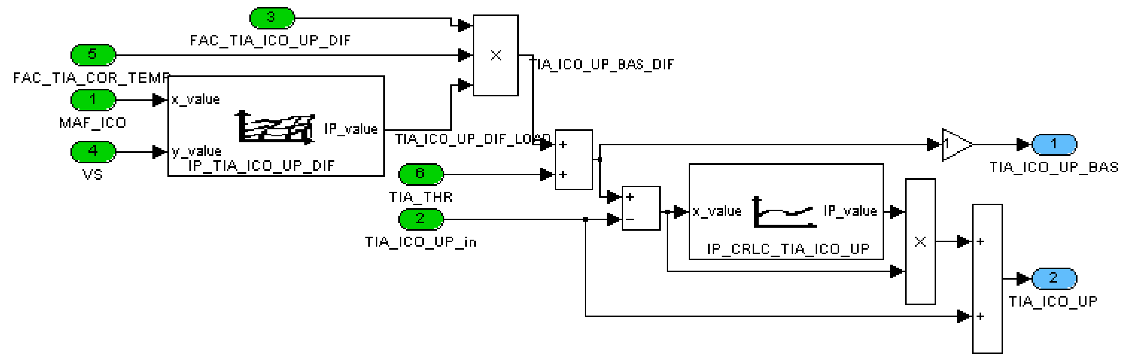


Figure 380 CHRG_MDLADICO0/ operate_100ms/ CALC_TIA_ICO_UP/ CALC_TIA_ICO_UP

4.46.1.3 SUBFUNCTION: operate_10ms

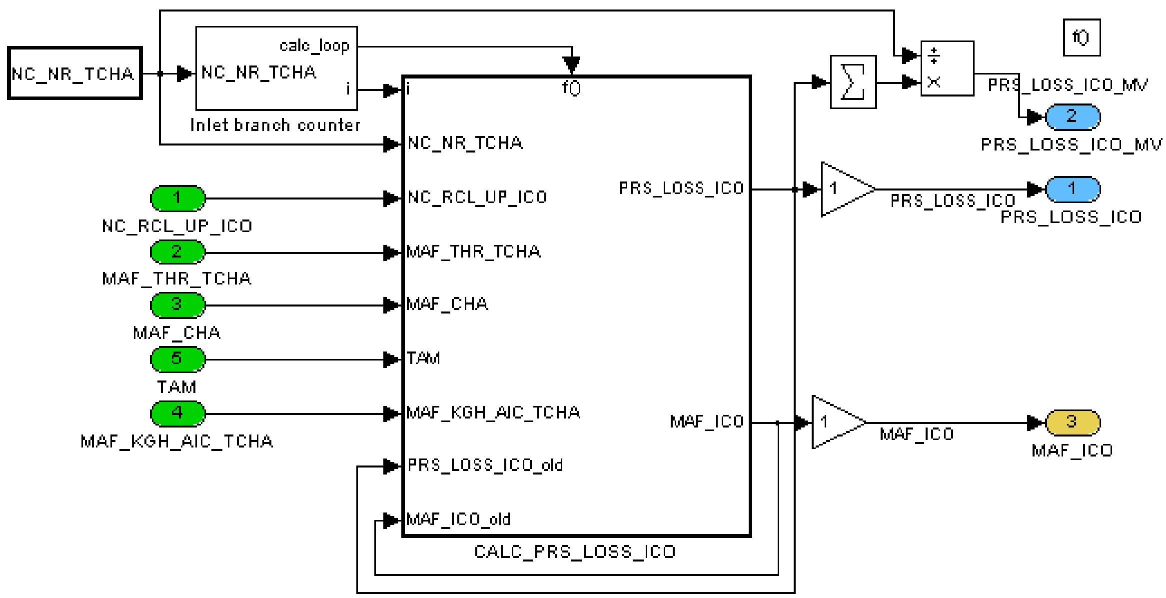


Figure 381 CHRG_MDLADICO0/ operate_10ms

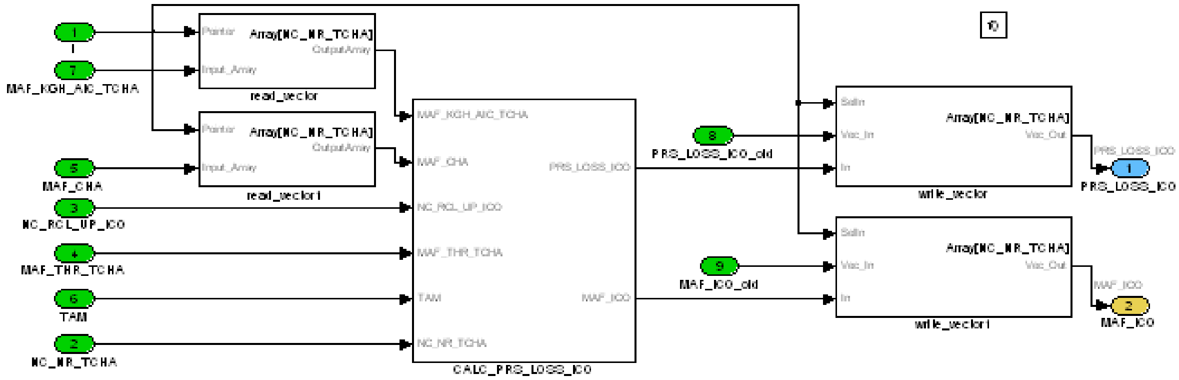


Figure 382 CHRG_MDLADICO0/ operate_10ms/ CALC_PRS_LOSS_ICO

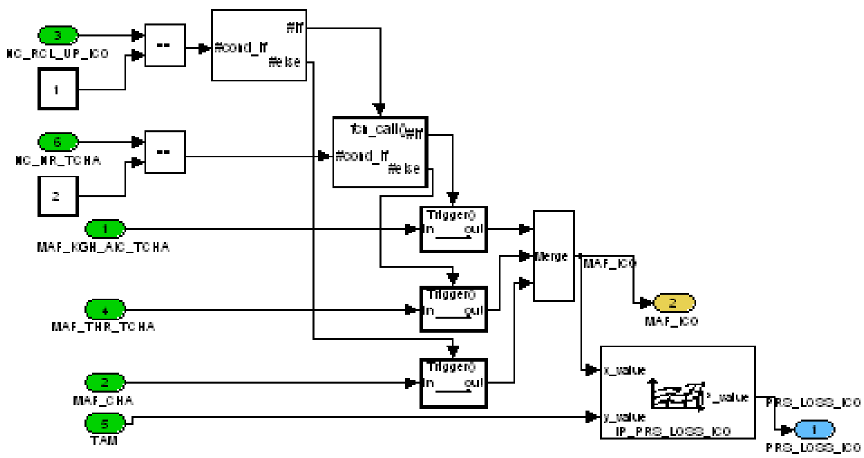


Figure 383 CHRG_MDLADICO0/ operate_10ms/ CALC_PRS_LOSS_ICO/ CALC_PRS_LOSS_ICO

4.47 Exhaust Gas Temperature upstream Turbine Turbo Charger

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEG_TUR_UP	O/V	0...FFE0H	-273.15 ... 1.77385E+3	0.03125	°C
Exhaust gas temperature upstream turbine of turbo charger					
TEG_TUR_UP_FIL	O/V	0...FFE0H	-273.15 ... 1.77385E+3	0.03125	°C
Filtered value of exhaust gas temperature upstream turbine of turbo charger					
TEG_TUR_UP_ABS	O/V	0...FFE0H	0...2.047E+3	0.03125	K
Exhaust gas temperature upstream turbine of turbo charger in Kelvin					
TEG_TUR_UP_ABS_SQRT	O/V	0...FFFFH	0...63.9	9.75051E-4	K^0.5
Square root of exhaust gas temperature upstream turbine of turbo charger in Kelvin					
TEG_TUR_UP_GRD	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	°C
Exhaust gas temperature upstream turbine gradient					

Input data:

LC_TCHA_CONF	NC_CBK_EX_NR	TEG_DYN_UP_TUR[NC_CBK_EX_NR]	LV_IGK

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEG_TUR_UP_FIL_HYS	1	0...FFE0H	-273.15 ... 1.77385E+3	0.03125	°C
Hysteresis for TEG_TUR_UP_FIL determination (TV:-8°C < -6.4°C = -1 digit around TEG_TUR_UP = 950°C)					
IP_TEG_TUR_UP_ABS_SQRT	16	0...FC00H	0...63	9.76563E-4	K^0.5
LDP_TEG_TUR_UP_ABS_TEG_TUR_UP	16	0...FFE0H	0...2.047E+3	0.03125	K
Square root of exhaust gas temperature upstream turbine in Kelvin					

4.47.1 General information

The function is calculated if LC_TCHA_CONF and LV_IGK are both equal one.

The exhaust gas temperature is filtered and its square root is calculated. The filter works only for decreasing exhaust gas temperature, negative changes smaller than C_TEG_TUR_UP_FIL_HYS are ignored.

Function Description

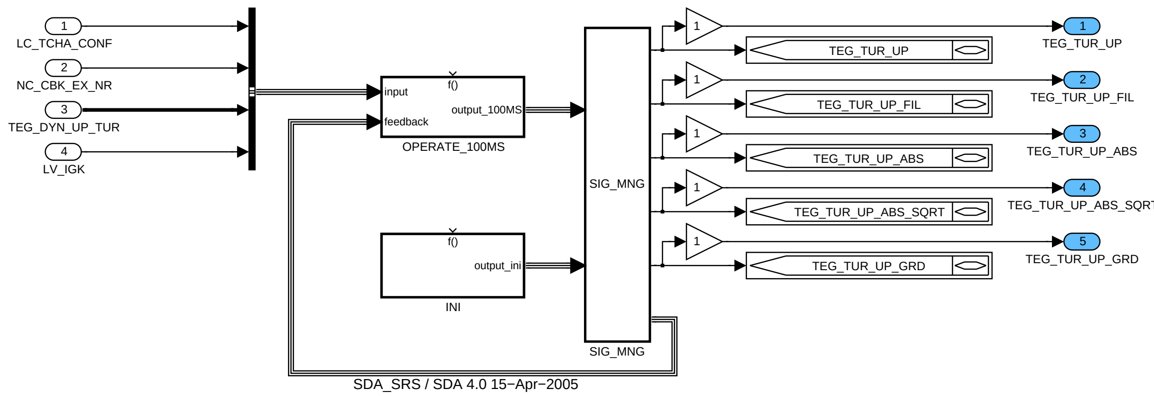


Figure 384 EXTD_MDLADad0

4.47.1.1 SUBFUNCTION: INI

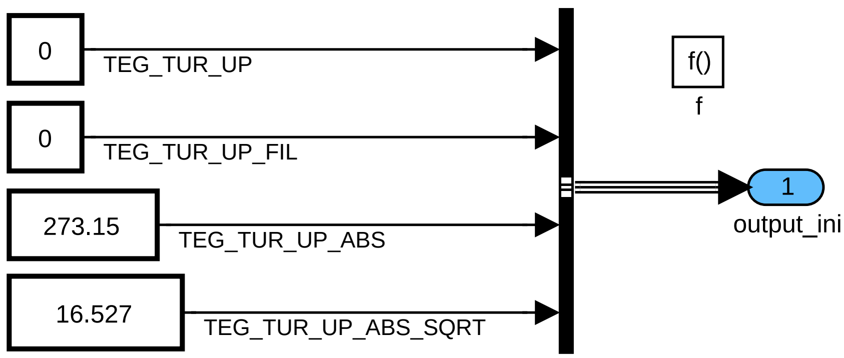


Figure 385 EXTD_MDLADad0/ INI

4.47.1.2 SUBFUNCTION: OPERATE_100MS

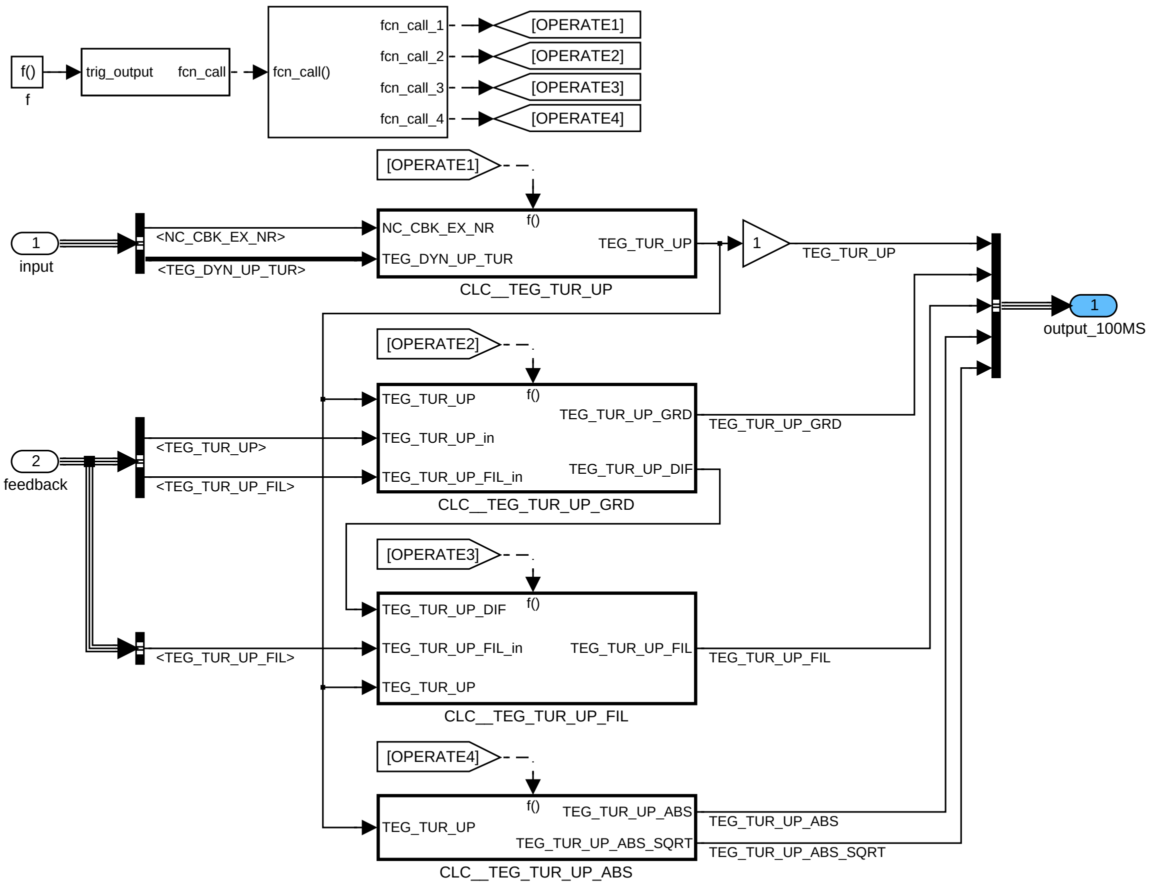


Figure 386 EXTD_MDLADad0/ OPERATE_100MS

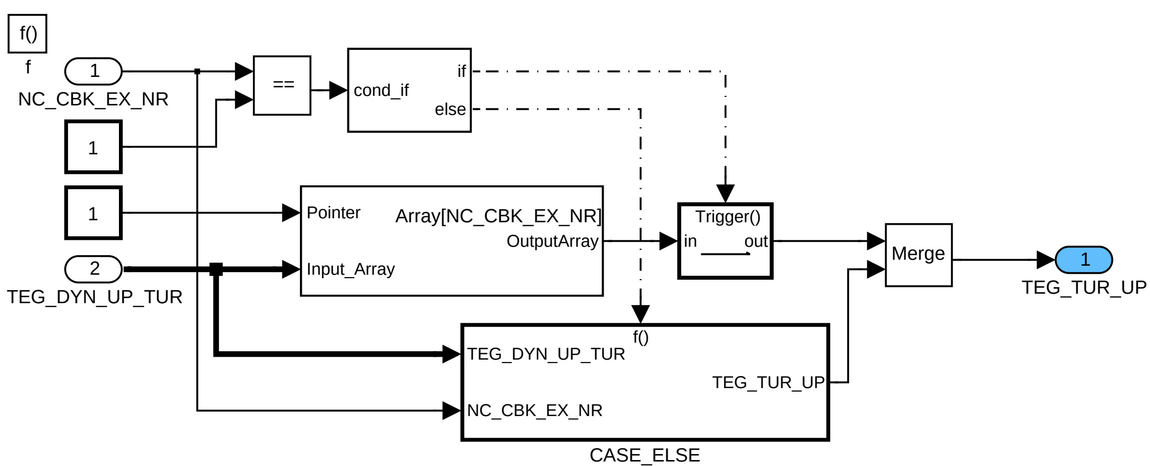


Figure 387 EXTD_MDLADad0/ OPERATE_100MS/ CLC__TEG_TUR_UP

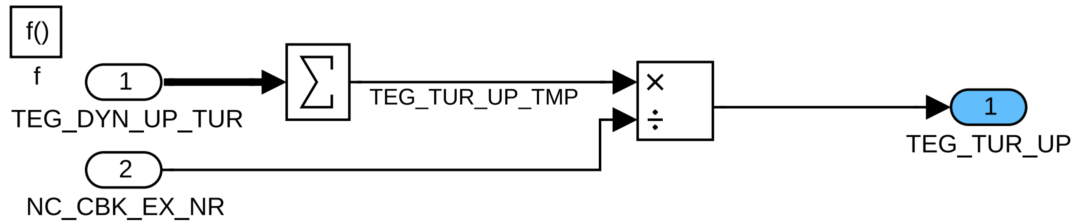


Figure 388 EXTD_MDLADad0/ OPERATE_100MS/ CLC__TEG_TUR_UP/ CASE_ELSE

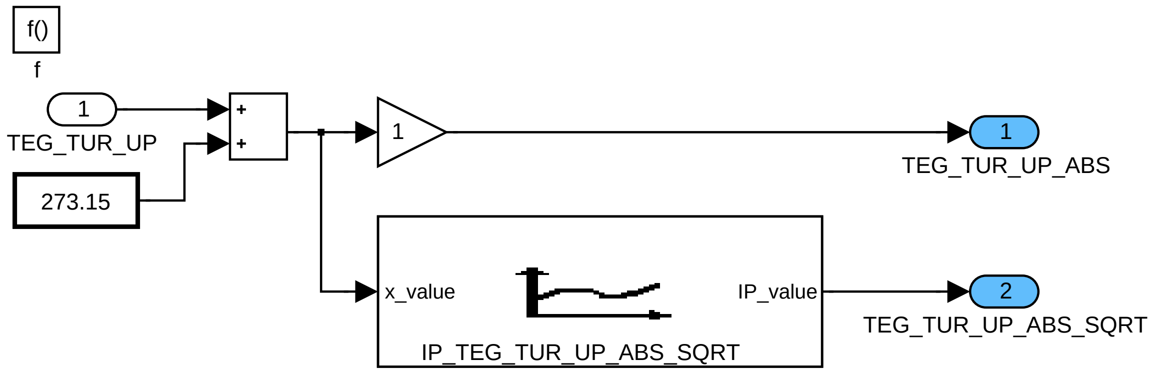


Figure 389 EXTD_MDLADad0/ OPERATE_100MS/ CLC__TEG_TUR_UP_ABS

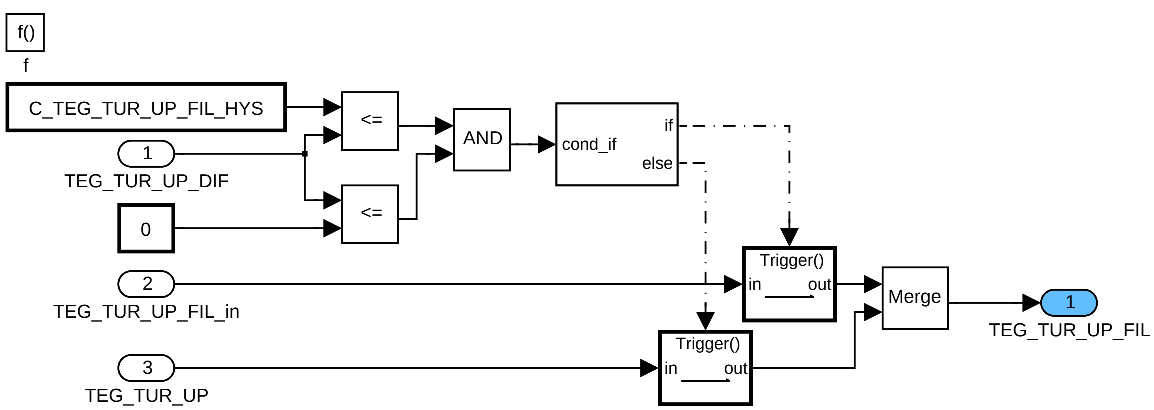


Figure 390 EXTD_MDLADad0/ OPERATE_100MS/ CLC__TEG_TUR_UP_FIL

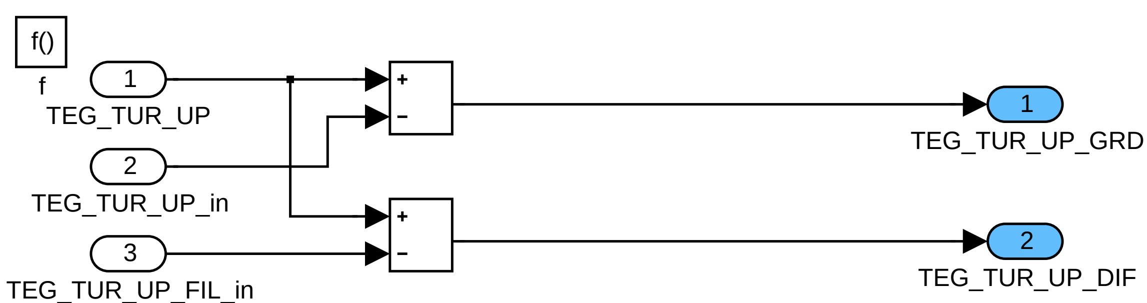


Figure 391 EXTD_MDLADad0/ OPERATE_100MS/ CLC__TEG_TUR_UP_GRD

4.47.1.3 SUBFUNCTION: SIG_MNG

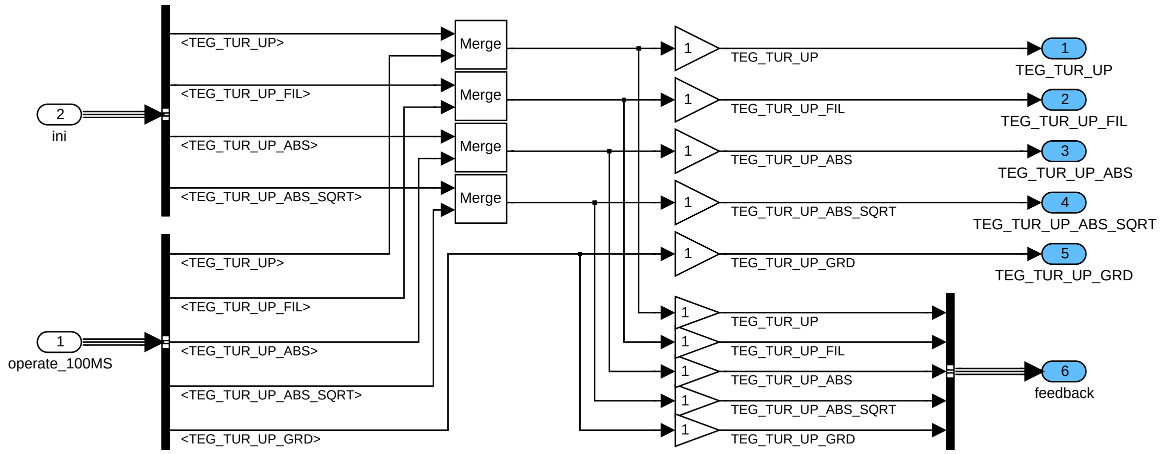


Figure 392 EXTD_MDLADad0/ SIG_MNG

4.48 Exhaust gas temperature sensors

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEG_MES[NC_NR_TEG_SENS]	O	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
measured exhaust gas temperatures					

Input data:

NC_NR_TEG_SENS

4.48.1 EXTD_SIGCV0

This module is just to define TEG_MES even if no sensor is available because TEG_MES is used as input in other functions:

TEG_MES[NC_NR_TEG_SENS] = 0°C (1112H)

Function Description

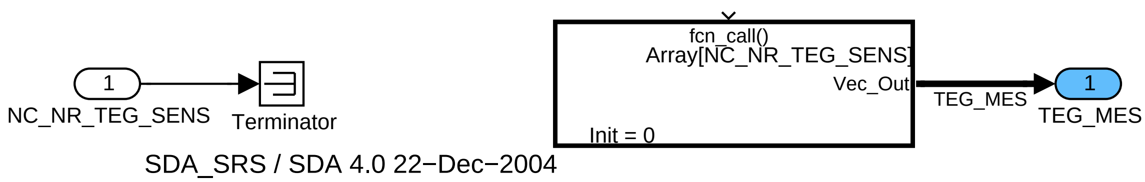


Figure 393 EXTD_SIGCV0

4.49 ECU temperature TECU

4.49.1 ECU Temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TECU	V/O	0...FE H	-48...142.5	0.75	°C
ECU/PCU hardware temperature					
TECU_MAX	V/O/S	0...FE H	-48...142.5	0.75	°C
Maximum measured ECU/PCU hardware temperature					

Input data:

V_TECU			

FUNCTION DESCRIPTION:

General information:

TECU represents the ECU/PCU internal temperature from ECU/PCU hardware.

TECU_MAX represents the max. measured ECU/PCU internal temperature since last programming.

Application conditions:

Recurrence: 1000 ms.

Initialisation: TECU = -48 °C (0H)

First Initialisation: TECU_MAX = -48 °C (0H)

Initialisation (after reset): TECU_MAX = TECU_MAX from previous driving cycle.

Formula section:

TECU = IP_TECU__V_TECU

If TECU > TECU_MAX

then TECU_MAX = TECU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TECU__V_TECU	8	0...FE H	-48...142.5	0.75	°C
LDP_V_TECU_IP_TECU	8	0...3FF H	0...4.995	4.88e-3	V
ECU temperature conversion table					

4.49.2 Monitoring of the ECU – Temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_STC_TECU_1	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 1 for TECU monitoring					
CTR_STC_TECU_2	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 2 for TECU monitoring					
CTR_STC_TECU_3	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 3 for TECU monitoring					
CTR_STC_TECU_4	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 4 for TECU monitoring					
CTR_STC_TECU_5	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 5 for TECU monitoring					
CTR_STC_TECU_6	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 6 for TECU monitoring					
CTR_STC_TECU_7	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 7 for TECU monitoring					
CTR_STC_TECU_8	V/S	0...FFFFFFFFH	0...4294967295	1	[-]
statistic counter 8 for TECU monitoring					
TRT_OLD	V	0...FFFFFFFFH	0...119304.6470	2.78E-05	[h]
TRT timestamp at last TECU monitoring classification					

Input data:

TECU	LV_ES	LV_IGK	TRT

General information:

This function should provide a statistical statement, how long the ECU was operated in a certain temperature range. After a calibratable time the actual ECU temperature is classified to a temperature range and the corresponding counter is incremented.

Application conditions:

Initialisation: restored from NVMY

Recurrence: 1 s

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If LV_IGK = 1 **and** LV_ES = 1 → 0
Then TRT_OLD = TRT
Else TRT_OLD = TRT_OLD
Endif

If(1) (LV_IGK = 1 **and** LV_ES = 0)

Then(1)**If**(2) TRT – TRT_OLD >= C_TRT_DIF_TECU

Then(2)TRT_OLD = TRT

If(3) TECU > C_TECU_THD_7

Then(3)$CTR\_STC\_TECU\_8_{(n)} = CTR\_STC\_TECU\_8_{(n-1)}+1$

Else(3) **If**(4) TECU > C_TECU_THD_6

Then(4)$CTR\_STC\_TECU\_7_{(n)} = CTR\_STC\_TECU\_7_{(n-1)}+1$

Else(4) **If**(5) TECU > C_TECU_THD_5

Then(5)$CTR\_STC\_TECU\_6_{(n)} = CTR\_STC\_TECU\_6_{(n-1)}+1$

Else(5) **If**(6) TECU > C_TECU_THD_4

Then(6)$CTR\_STC\_TECU\_5_{(n)} = CTR\_STC\_TECU\_5_{(n-1)}+1$

Else(6) **If**(7) TECU > C_TECU_THD_3

Then(7)$CTR\_STC\_TECU\_4_{(n)} = CTR\_STC\_TECU\_4_{(n-1)}+1$

Else(7) **If**(8) TECU > C_TECU_THD_2

Then(8) $CTR\_STC\_TECU\_3_{(n)} = CTR\_STC\_TECU\_3_{(n-1)}+1$

Else(8) **If**(9) TECU > C_TECU_THD_1

Then(9) $CTR\_STC\_TECU\_2_{(n)} = CTR\_STC\_TECU\_2_{(n-1)}+1$

Else(9) $CTR\_STC\_TECU\_1_{(n)} = CTR\_STC\_TECU\_1_{(n-1)}+1$

Endif(9)

Endif(8)

Endif(7)

Endif(6)

Endif(5)

Endif(4)

Endif(3)

Else(2) do nothing

Endif(2)

Else(1) do nothing

Endif(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TECU_THD_1	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 1 (init. Value: 50°C)					
C_TECU_THD_2	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 2 (init. Value: 60°C)					
C_TECU_THD_3	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 3 (init. Value: 70°C)					
C_TECU_THD_4	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 4 (init. Value: 85°C)					
C_TECU_THD_5	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 5 (init. Value: 100°C)					
C_TECU_THD_6	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 6 (init. Value: 110°C)					
C_TECU_THD_7	1	0...FEH	-48...142.5	0.75	[°C]
TECU threshold 7 (init. Value: 120°C)					
C_TRT_DIF_TECU	1	0...FFFFFFFFH	0... 119304.64708	0.0278e-3	[h]
TRT difference for ETCU statistic (init. value: 0,08333h)					

4.50 Determination of accelerator pedal value (PVS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_PVS_FIL_1	O/V	0...3FFH	0...4.995117	0.004883	V
Filtered voltage of Pedal Value Sensor 1					
V_PVS_FIL_2	O/V	0...3FFH	0...4.995117	0.004883	V
Filtered voltage of Pedal Value Sensor 2					
PV_AV_1_H	O/V	0...3FFH	0...99.9	0.09765	%
Degree of activation of the accelerator pedal from PVS-channel 1 (high resolution)					
PV_AV_2_H	O/V	0...3FFH	0...99.9	0.09765	%
Degree of activation of the accelerator pedal from PVS-channel 2 (high resolution)					
PV_AV_H	O/V	0...3FFH	0...99.9	0.09765	%
Global degree of activation of the accelerator pedal (high resolution)					
PV_AV	O/V	0...FFH	0...99.6	0.3906	%
Global degree of activation of the accelerator pedal (low resolution)					
PV_AV_SEL	O/V	0...3FFH	0...99.9	0.09765	%
Degree of activation of accelerator pedal raw value (high resolution)					
LV_PV_AV_SEL_NULL	O/V	0…1H	0…1	1	-
Logical Variable to indicate PV_AV_SEL set to zero as Forced Idle Limp Home Reaction					
LV_ERR_PVS_SNG	O/V	0…1H	0…1	1	-
only fault on one PVS channel and/or Ratio-error					
LV_ERR_PVS_MPL	O/V	0…1H	0…1	1	-
Fault on both PVS channels					

Input data:

V_PVS_1	V_PVS_2	LV_ERR_PVS_PRED	LV_ERR_PVS
LV_ERR_PVS_1	LV_ERR_PVS_2	LV_ERR_PVS_RATIO	LV_LIH_PVS_MOVE
LV_ERR_VCC_PVS_1	LV_ERR_VCC_PVS_2	LV_PVS_BLS_BTS_NOT_PLAUS	LV_LIH_BLS_BTS
LV_BLS	LV_BTS	LV_ERR_PVS_RATIO_PRED	C_CTR_MAX_PVS_MOVE

FUNCTION DESCRIPTION:

General information:

The accelerator pedal value is determined by means of a two-channel system (PVS: pedal value sensor).

The gradient of the raw values is limited. The values are filtered and normalized to a range between 0 and 99,9%.

The calculation of the pedal value depends on the error state. Normally the pedal value is calculated by channel 1. In case of a disturbed PVS-signal, a selection of one of both PVS-channels and a limitation is done, or the output is set to 0%.

Since it is a two-channel system, the letter ‘X’ is used for either channel 1 or channel 2, in order to simplify the description.

General Application condition:

Activation: always

Deactivation: -

Initialization: for additional conditions see ‘Appl. Incidences of PVS Determination’. Initialization shall be done according to the project philosophy.

V_PVS_1/2$_{(K-1)}$_INI= 5V

V_PVS_FIL_1/2$_{(K-1)}$_INI = 0V

LV_ERR_PVS_SUB$_{(K)}$ = 0

CTR_PVS_THD_IS_L_1 = C_CTR_MAX_PVS_MOVE

CTR_IDLE = C_CTR_MAX_PVS_IS_DET

LV_PVS_THD_IS_H_INI = 1

PV_AV_1/2_H = 0

PV_AV_H = 0

PV_AV = 0

PV_AV_SEL = 0

LV_PV_AV_SEL_NULL = 0

Recurrence: every 10 ms

Signal flow diagram:

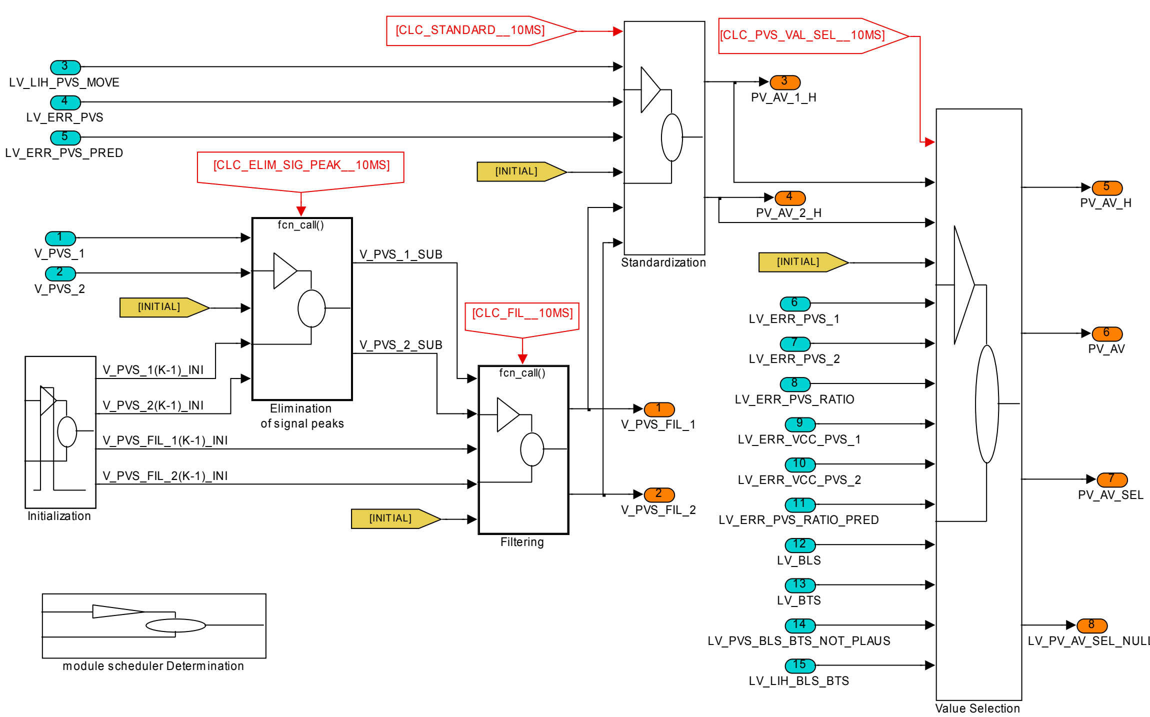


Figure 1: Determination of PVS

Description:

The module “Determination of accelerator pedal value (PVS)” is executed at function call CLC_PVS_SIG_CONV__10MS and initialized at CLC_PVS_SIG_CONV__RST.
Each function part is calculated according to the corresponding function call, which are organized in the module scheduler as shown below.

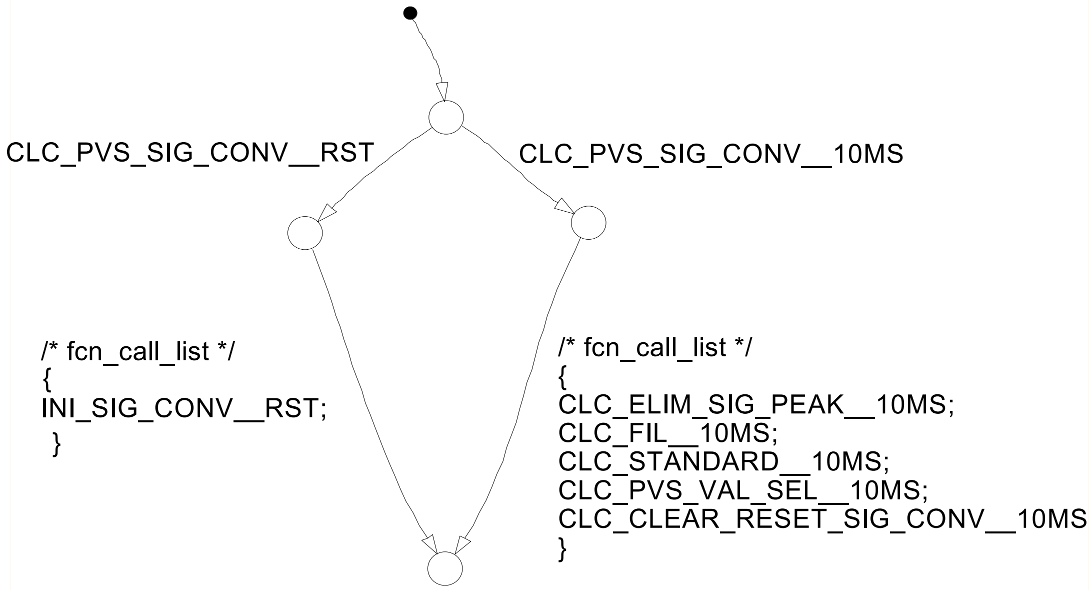


Figure 2: module scheduler Determination

Calculation order:

At initialization (CLC_PVS_SIG_CONV__RST):

1.	INI_SIG_CONV__RST	=>	INITIALIZATION

Calculation regular:

1.	CLC_ELIM_SIG_PEAK__10MS	=>	Elimination of Signal Peaks
2.	CLC_FIL__10MS	=>	Filtration
3.	CLC_STANDARD__10MS	=>	Standardization
4.	CLC_PVS_VAL_SEL	=>	Value Selection
5.	CLC_CLEAR_RESET_SIG_CONV	=>	INITIALIZATION

Initialization:

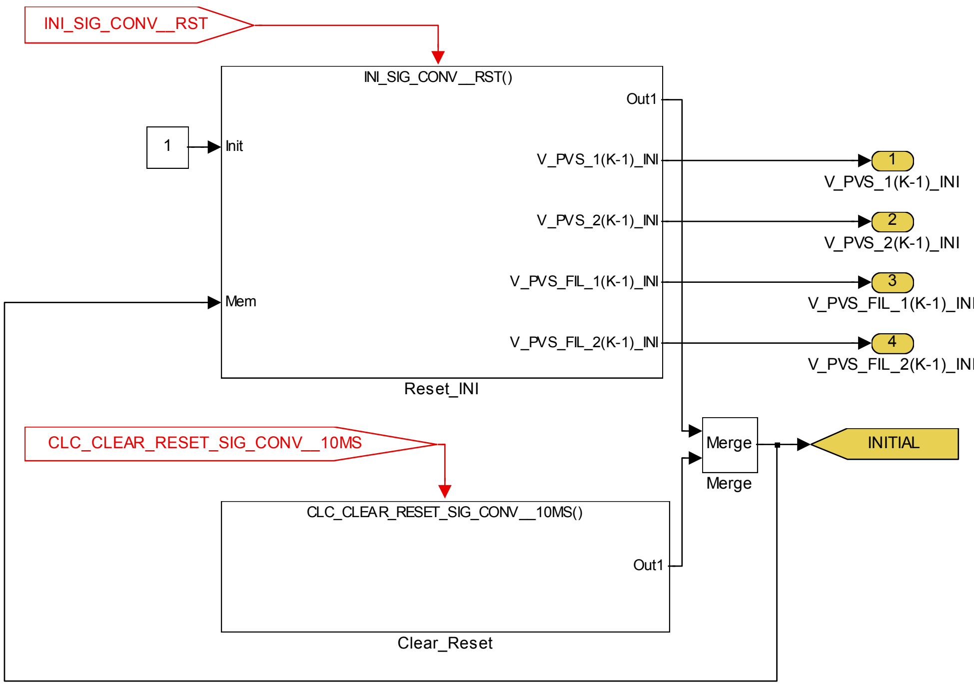


Figure 3: Initialization

At function call **INI_SIG_CONV__RST** initialization is done and “INITIAL” is set **active**.
It is set to **passive** again at the end of the next regular calculation cycle at function call **CLC_CLEAR_RESET_SIG_CONV__10MS**.

4.50.1 Elimination of signal peaks

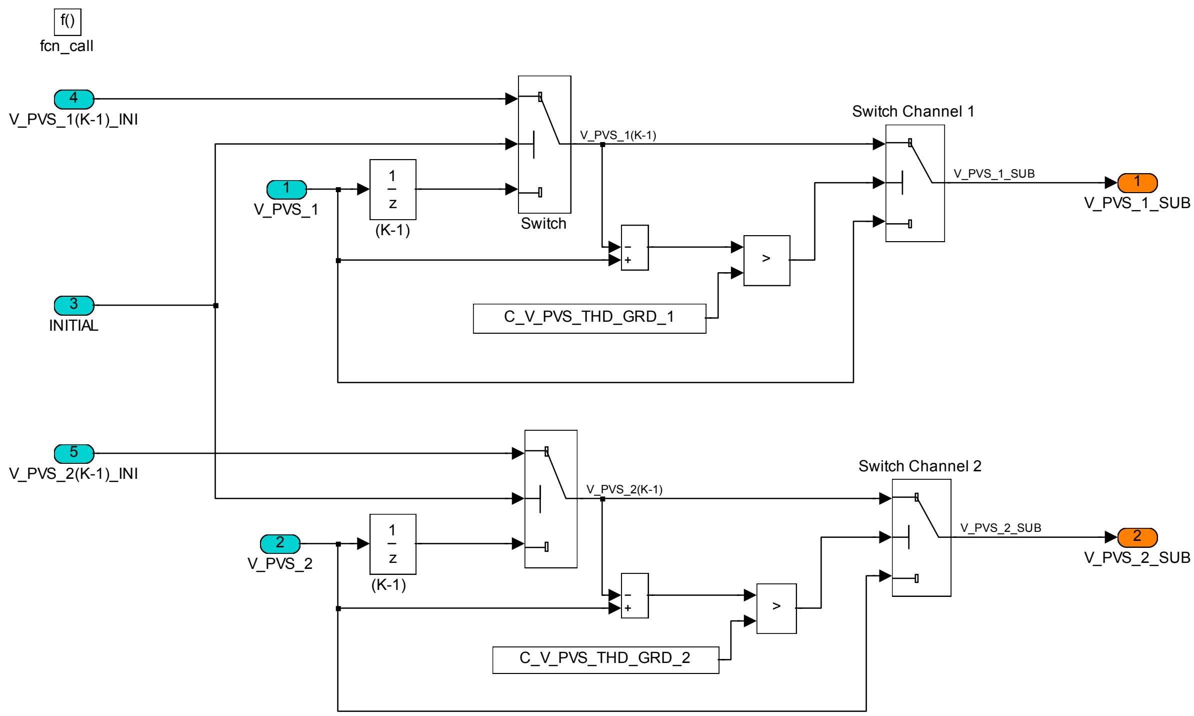


Figure 4: Elimination of signal peaks

Description:

To avoid surge voltage, the gradients of the PVS-signals are limited. If the positive signal gradient of channel X is larger than C_V_PVS_THD_GRD_X, the last value will be kept for one sampling period.

Formula section

Switch Channel 1:

IF $V\_PVS\_1_{(K)} - V\_PVS\_1_{(K-1)} > C\_V\_PVS\_THD\_GRD\_1$

{intermediate results are limited to pos. values}

THEN $V\_PVS\_1\_SUB = V\_PVS\_1_{(K-1)}$

ELSE $V\_PVS\_1\_SUB = V\_PVS\_1_{(K)}$

Switch Channel 2:

IF $V\_PVS\_2_{(K)} - V\_PVS\_2_{(K-1)} > C\_V\_PVS\_THD\_GRD\_2$

{intermediate results are limited to pos. values}

THEN $V\_PVS\_2\_SUB = V\_PVS\_2_{(K-1)}$

ELSE $V\_PVS\_2\_SUB = V\_PVS\_2_{(K)}$

4.50.2 Filtering of the PVS-Values

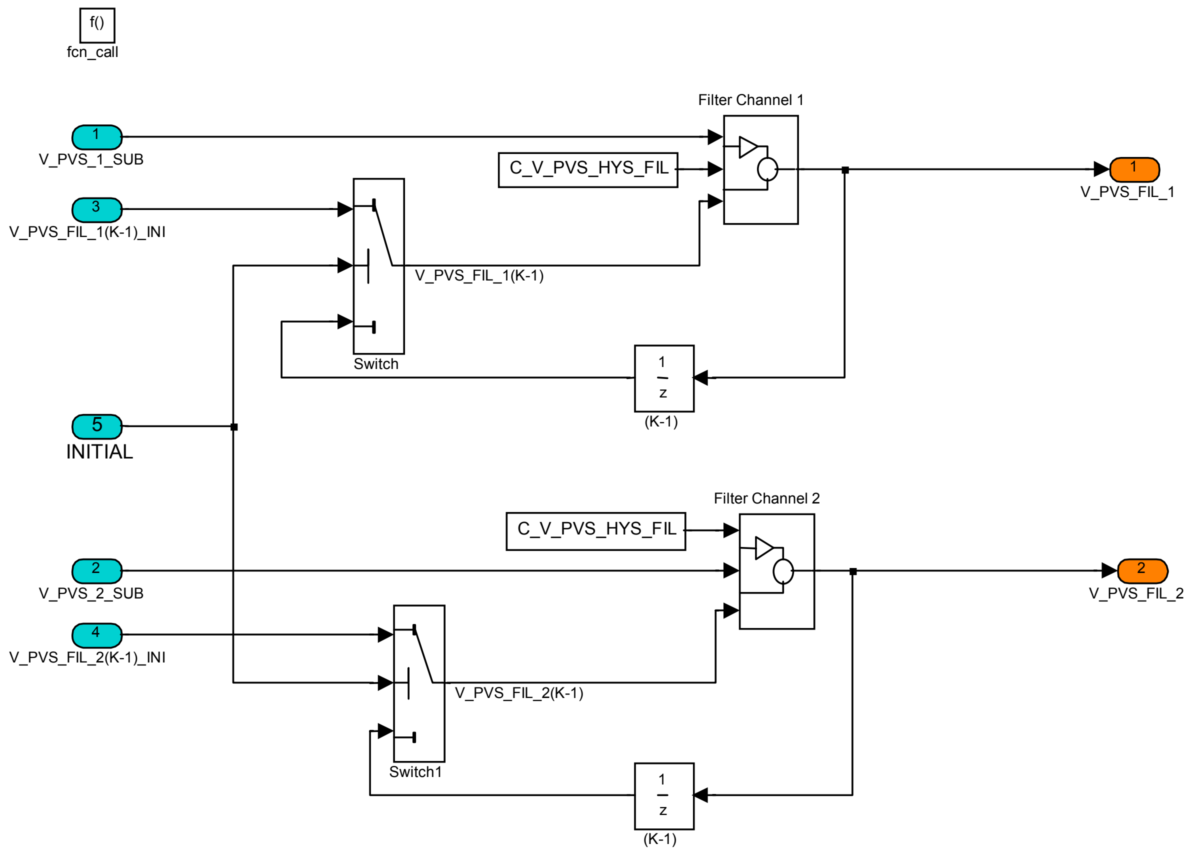


Figure 5: Filtering

Description:

To maintain a stabile driver demand, the PVS-Voltage values need to be filtered.
To avoid a detoriation of the signal accuracy, an asymmetrical filter is used.
The hysteresis C_V_PVS_HYS_FIL is used for both PVS-signals. For increasing signal voltage V_PVS_FIL_X = V_PVS_X_SUB – C_V_PVS_HYS_FIL.
This means a continuously increasing without big incremental steps.
For decreasing signal voltage V_PVS_FIL_X follows the raw value.

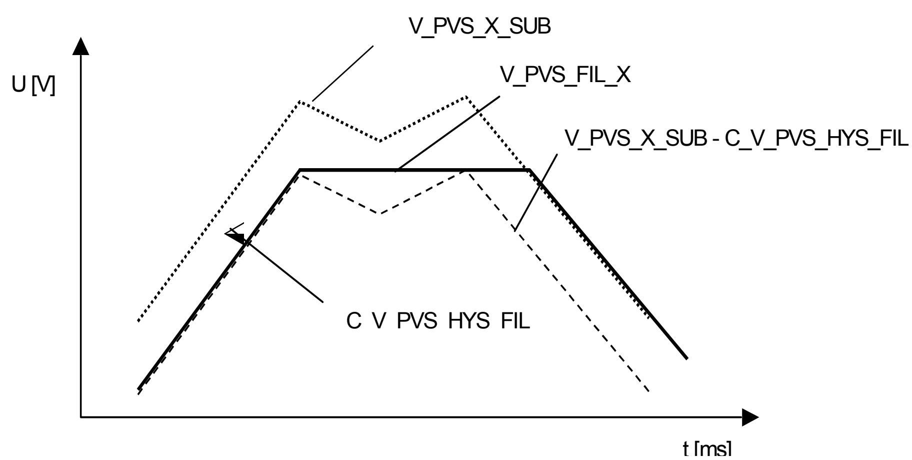


Figure 6: asymmetrical filter

Formula section:

Filter 'Channel_1':

IF V_PVS_FIL_1$_{(K-1)}$ < (V_PVS_1_SUB - C_V_PVS_HYS_FIL)
THEN V_PVS_FIL_1 = V_PVS_1_SUB - C_V_PVS_HYS_FIL
ELSE
  IF V_PVS_FIL_1$_{(K-1)}$ > V_PVS_1_SUB
  THEN V_PVS_FIL_1=V_PVS_1_SUB
  ELSE V_PVS_FIL_1=V_PVS_FIL_1$_{(K-1)}$
  ENDIF
ENDIF

Filter 'Channel_2':

IF V_PVS_FIL_2$_{(K-1)}$ < (V_PVS_2_SUB - C_V_PVS_HYS_FIL)
THEN V_PVS_FIL_2 = V_PVS_2_SUB - C_V_PVS_HYS_FIL
ELSE
  IF V_PVS_FIL_2$_{(K-1)}$ > V_PVS_2_SUB
  THEN V_PVS_FIL_2=V_PVS_2_SUB
  ELSE V_PVS_FIL_2=V_PVS_FIL_2$_{(K-1)}$
  ENDIF
ENDIF

4.50.3 Standardization of PVS

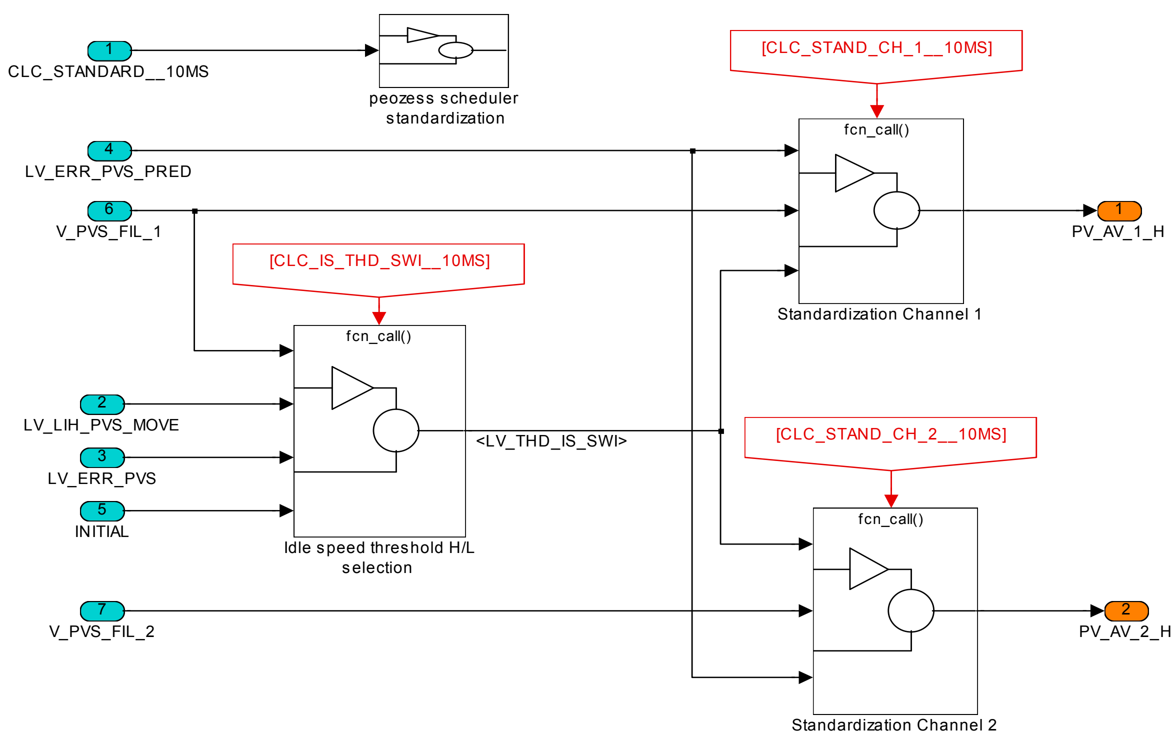


Figure 7: Standardization of PVS

Description:

The sensor voltages are normalized to a range between 0 and 99,9% (pedal value). The calculation is done with high or low thresholds for idle speed recognition depending on PVS error state and Initialization (idle speed threshold H/L selection).

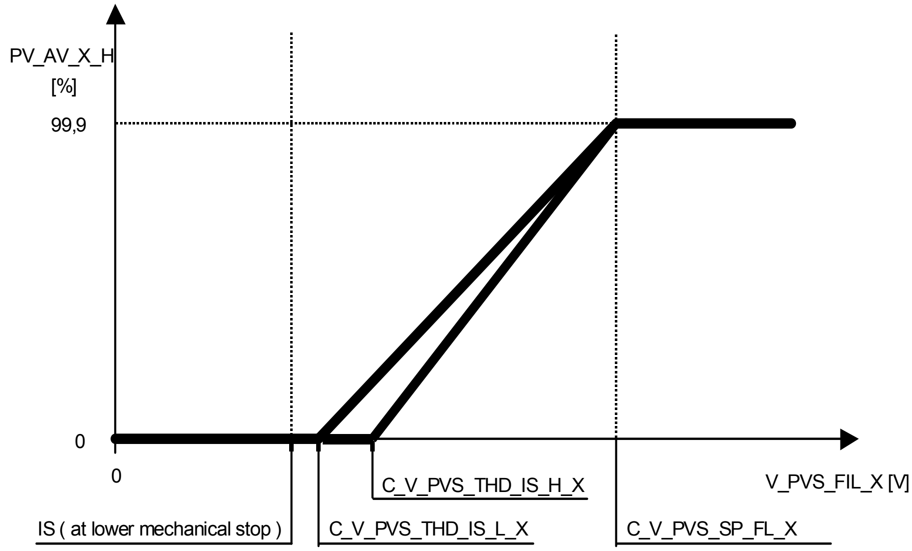


Figure 8: Standardization with high or low IS-threshold

Calculation order:

1. CLC_IS_THD_SWI__10MS => idle speed threshold H/L selection
2. CLC_STAND_CH_1__10MS => Standardization Channel 1
3. CLC_STAND_CH_2__10MS => Standardization Channel 2

4.50.3.1 idle speed threshold H/L selection

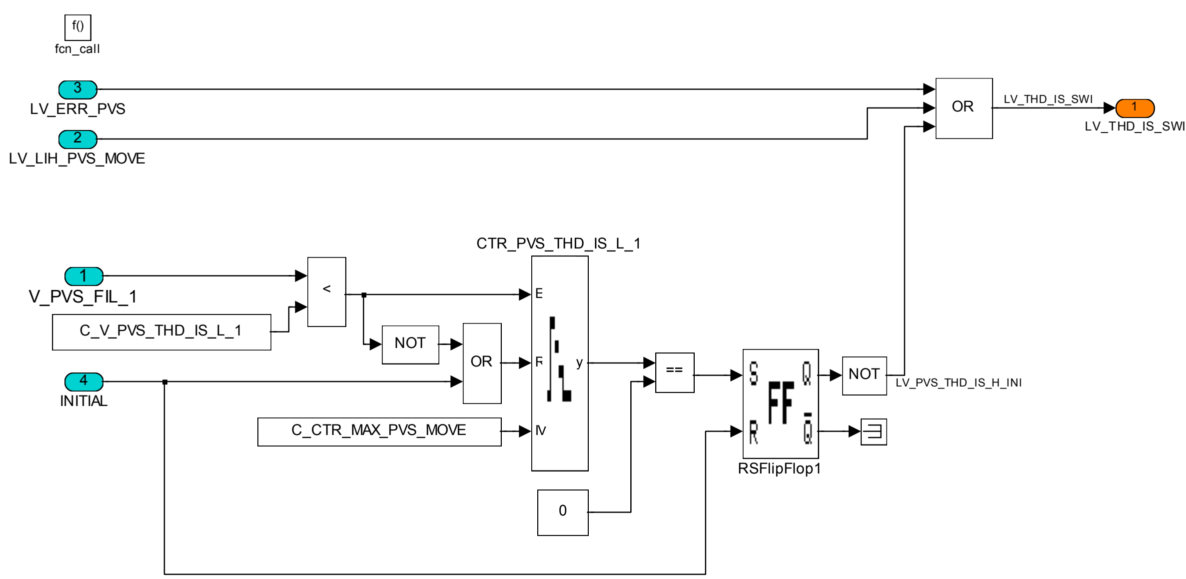


Figure 9: idle speed threshold H/L selection

Description:

The calculation of the pedal values with high idle speed thresholds is done (LV_THD_IS_SWI = 1), if

- a PVS error has been detected LV_ERR_PVS = 1, or
- or a Move error is recognized LV_LIH_PVS_MOVE = 1, or
- as long as V_PVS_FIL_1 has not been below C_V_PVS_THD_IS_L_1 for C_CTR_MAX_PVS_MOVE after initialization LV_PVS_THD_IS_H_INI = 1

Otherwize (LV_SWI_IS_THD = 0) the low thresholds are used.

For functionality it is essentially that the recognition time is the same as in the Move Check (Diagnosis of PVS), therefore C_CTR_MAX_PVS_MOVE is used here, too. It is defined in PVS Diagnosis.

Formula section:

```
IF          NOT V_PVS_FIL_1 < C_V_PVS_THD_IS_L_1                         OR
            at Initialization
THEN        CTR_PVS_THD_IS_L_1(K) = C_CTR_MAX_PVS_MOVE
ELSE
    IF      CTR_PVS_THD_IS_L_1(K) >0                                     AND
            V_PVS_FIL_1 < C_V_PVS_THD_IS_L_1
    THEN    CTR_PVS_THD_IS_L_1(K) = CTR_PVS_THD_IS_L_1(K-1) – [ update rate]
    ENDIF
ENDIF

IF          CTR_PVS_THD_IS_L_1(K) = 0
THEN        LV_PVS_THD_IS_H_INI = 0
ENDIF
{ at Initialization LV_PVS_THD_IS_H_INI = 1}

IF          LV_ERR_PVS = 1                                               OR
            LV_LIH_PVS_MOVE = 1                                          OR
            LV_PVS_THD_IS_H_INI = 1
THEN        LV_THD_IS_SWI = 1
ELSE        LV_THD_IS_SWI = 0
ENDIF
```

4.50.3.2 Standardization Channel 1

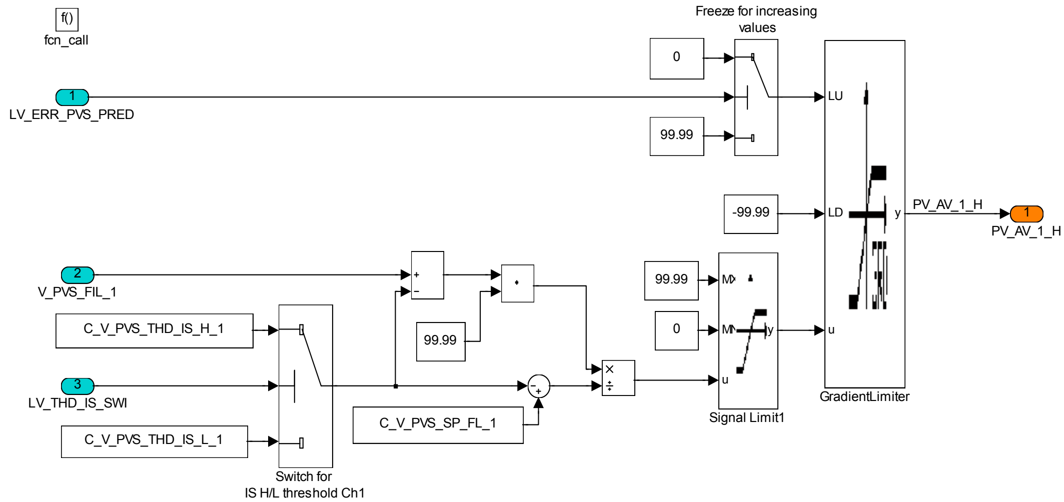


Figure 10: Standardization Channel 1

Description:

If an error is predicted but not debounced yet (LV_ERR_PVS_PRED), PV_AV_1_H will be frozen for $PV\_AV\_1\_H_{(K)} > PV\_AV\_1\_H_{(K-1)}$.

Depending on LV_THD_IS_SWI the high or low thresholds for idle speed recognition are selected for calculation.

If LV_THD_IS_SWI = 0:

$$PV\_AV\_1\_H = \frac{(V\_PVS\_FIL\_1 - C\_V\_PVS\_THD\_IS\_L\_1) \cdot 99{,}9\%}{C\_V\_PVS\_SP\_FL\_1 - C\_V\_PVS\_THD\_IS\_L\_1}$$

If LV_THD_IS_SWI = 1:

$$PV\_AV\_1\_H = \frac{(V\_PVS\_FIL\_1 - C\_V\_PVS\_THD\_IS\_H\_1) \cdot 99{,}9\%}{C\_V\_PVS\_SP\_FL\_1 - C\_V\_PVS\_THD\_IS\_H\_1}$$

4.50.3.3 Standardization Channel 2

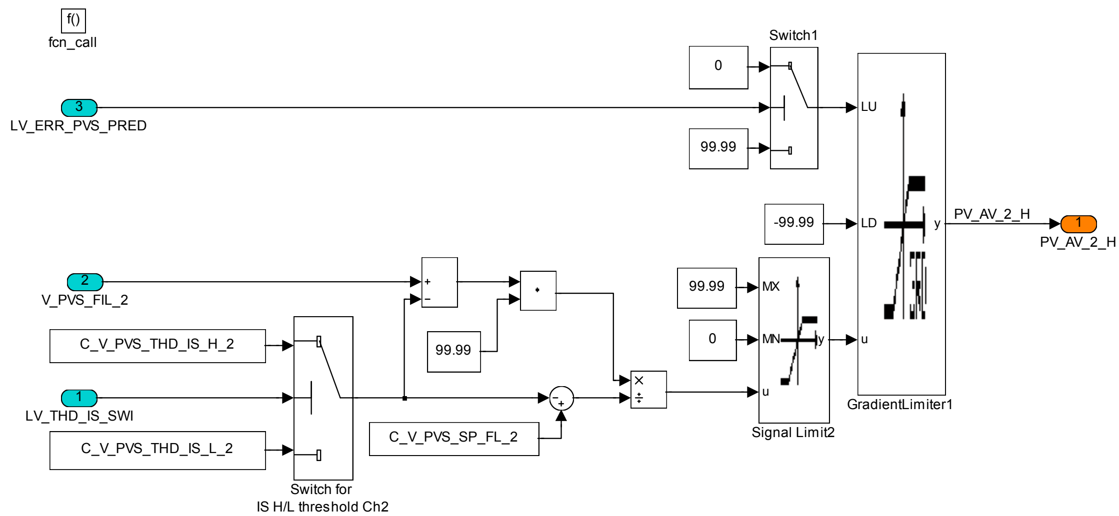


Figure 11: Standardization Channel 2

Description:

If an error is predicted but not debounced yet (LV_ERR_PVS_PRED), PV_AV_2_H will be frozen for $PV\_AV\_2\_H_{(K)} > PV\_AV\_2\_H_{(K-1)}$.

Depending on LV_THD_IS_SWI the high or low thresholds for idle speed recognition are selected for calculation.

If LV_THD_IS_SWI = 0:

$$PV\_AV\_2\_H = \frac{(V\_PVS\_FIL\_2 - C\_V\_PVS\_THD\_IS\_L\_2) \cdot 99{,}9\%}{C\_V\_PVS\_SP\_FL\_2 - C\_V\_PVS\_THD\_IS\_L\_2}$$

If LV_THD_IS_SWI = 1:

$$PV\_AV\_2\_H = \frac{(V\_PVS\_FIL\_2 - C\_V\_PVS\_THD\_IS\_H\_2) \cdot 99{,}9\%}{C\_V\_PVS\_SP\_FL\_2 - C\_V\_PVS\_THD\_IS\_H\_2}$$

.

4.50.4 Value Selection

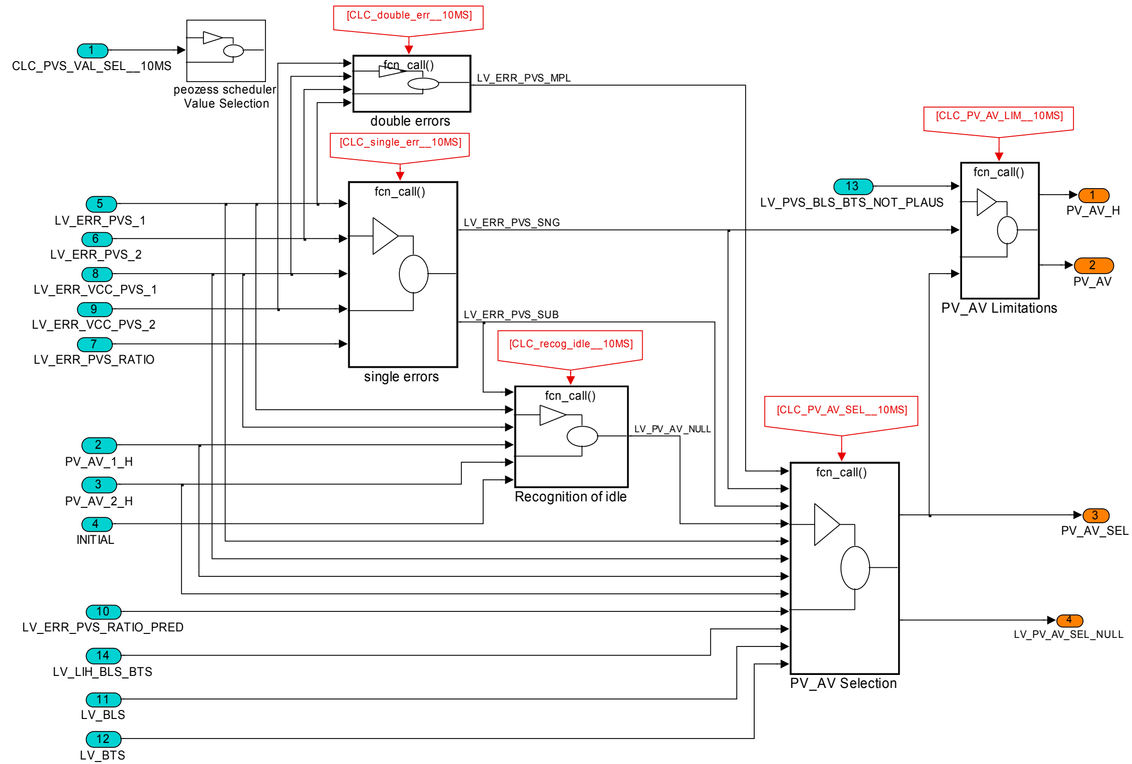


Figure 12: Value Selection

Description:

Depending on PVS error state the final pedal value (PV_AV_H) is calculated. A channel selection and a limitation is done based on PVS diagnosis and brake information.

Calculation order:

1.	CLC_single_err__10MS	=>	single errors
2.	CLC_double_err__10MS	=>	double errors
3.	CLC_recog_idle__10MS	=>	Recognition of idle
4.	CLC_PV_AV_SEL__10MS	=>	PV_AV Selection
5.	CLC_PV_AV_LIM__10MS	=>	PV_AV Limitation

4.50.4.1 single errors

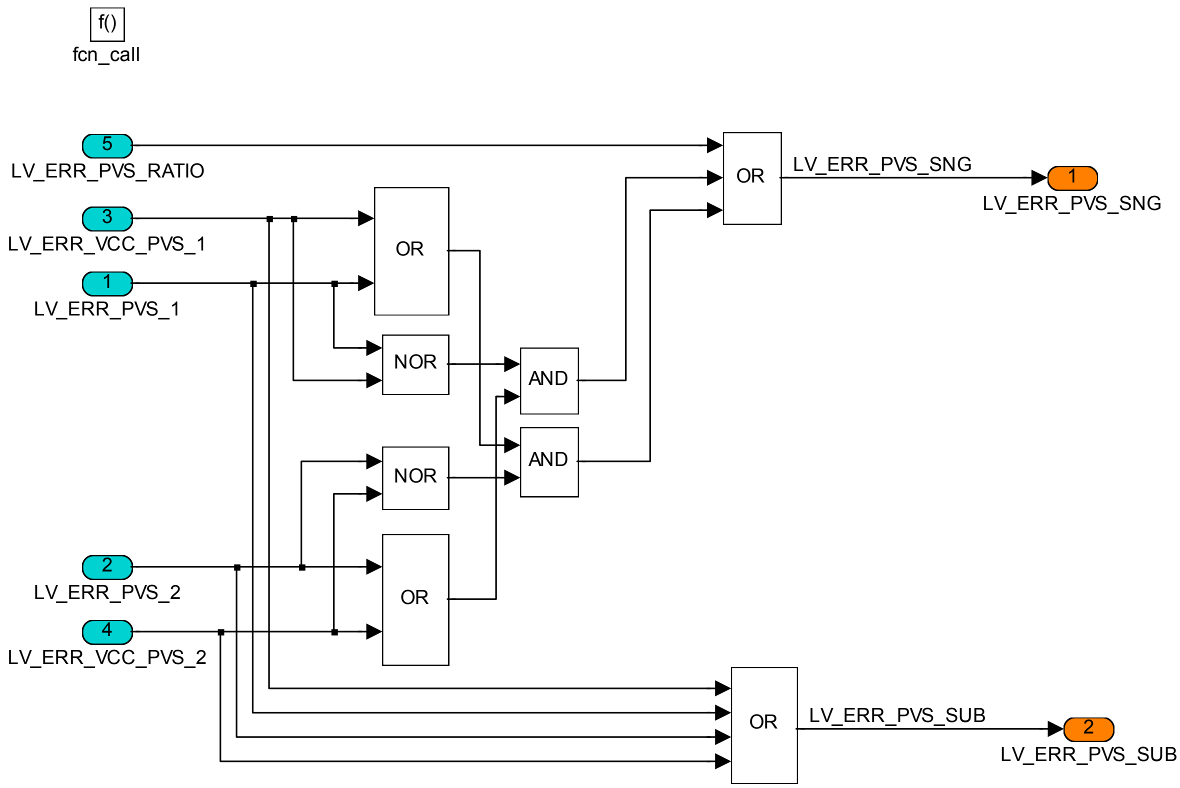


Figure 13: single errors

Formula section:

IF	LV_ERR_PVS_RATIO == 1	**OR**
	{ (LV_ERR_PVS_1==1 or LV_ERR_VCC_PVS_1==1) **and** NOT(LV_ERR_PVS_2==1 or LV_ERR_VCC_PVS_2==1) **}**	**OR**
	{ (LV_ERR_PVS_2==1 or LV_ERR_VCC_PVS_2==1) **and** NOT(LV_ERR_PVS_1==1 or LV_ERR_VCC_PVS_1==1) **}**	
THEN	LV_ERR_PVS_SNG = 1 (only fault on one channel)	
ELSE	LV_ERR_PVS_SNG = 0	
ENDIF		

IF	LV_ERR_PVS_1==1	**OR**
	LV_ERR_VCC_PVS_1==1	**OR**
	LV_ERR_PVS_2==1	**OR**
	LV_ERR_VCC_PVS_2==1	**OR**
THEN	LV_ERR_PVS_SUB = 1 (at least one error is detected)	
ELSE	LV_ERR_PVS_SUB = 0	
ENDIF		

4.50.4.2 double errors

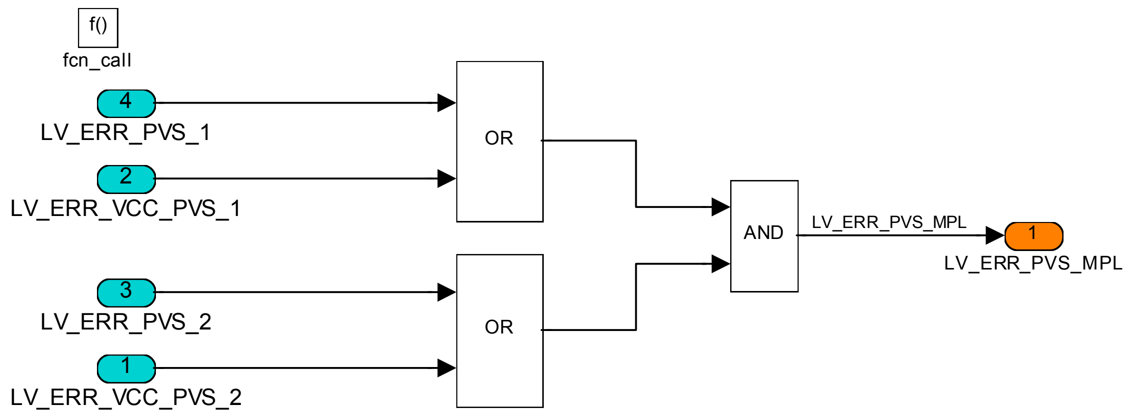


Figure 14: double errors

Formula section:

If	(LV_ERR_PVS_1 == 1 **or** LV_ERR_VCC_PVS_1 == 1)
and	(LV_ERR_PVS_2 == 1 **or** LV_ERR_VCC_PVS_2 == 1)
then	LV_ERR_PVS_MPL = 1 (both channels faulty)
else	LV_ERR_PVS_MPL = 0

4.50.4.3 Recognition of idle

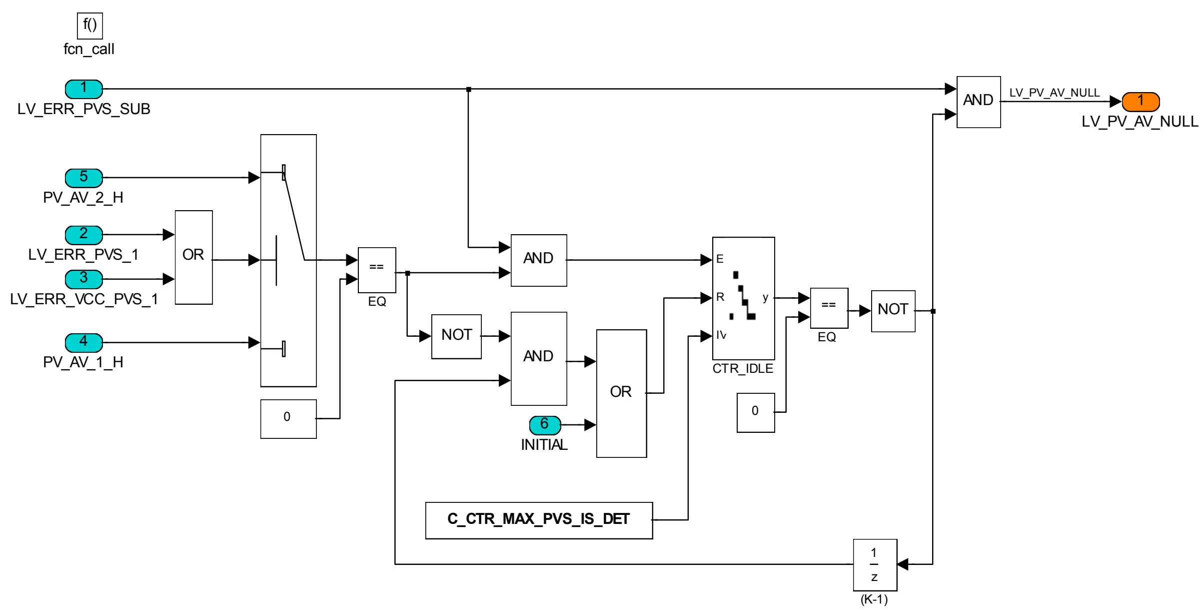


Figure 15: Recognition of PV_AV_X_H at idle

Description:

If channel x is recognized as faulty, the pedal value PV_AV_SEL will be set to 0% directly. The other channel can be selected as the leading channel if driver demand has been recognized as idle (PV_AV_y_H = 0 %) for C_CTR_MAX_PVS_IS_DET * update rate without interruption. As long as this condition is not fulfilled, PV_AV_SEL remains set to 0 % (LV_PV_AV_NULL = 1).

4.50.4.4 PV_AV Selection

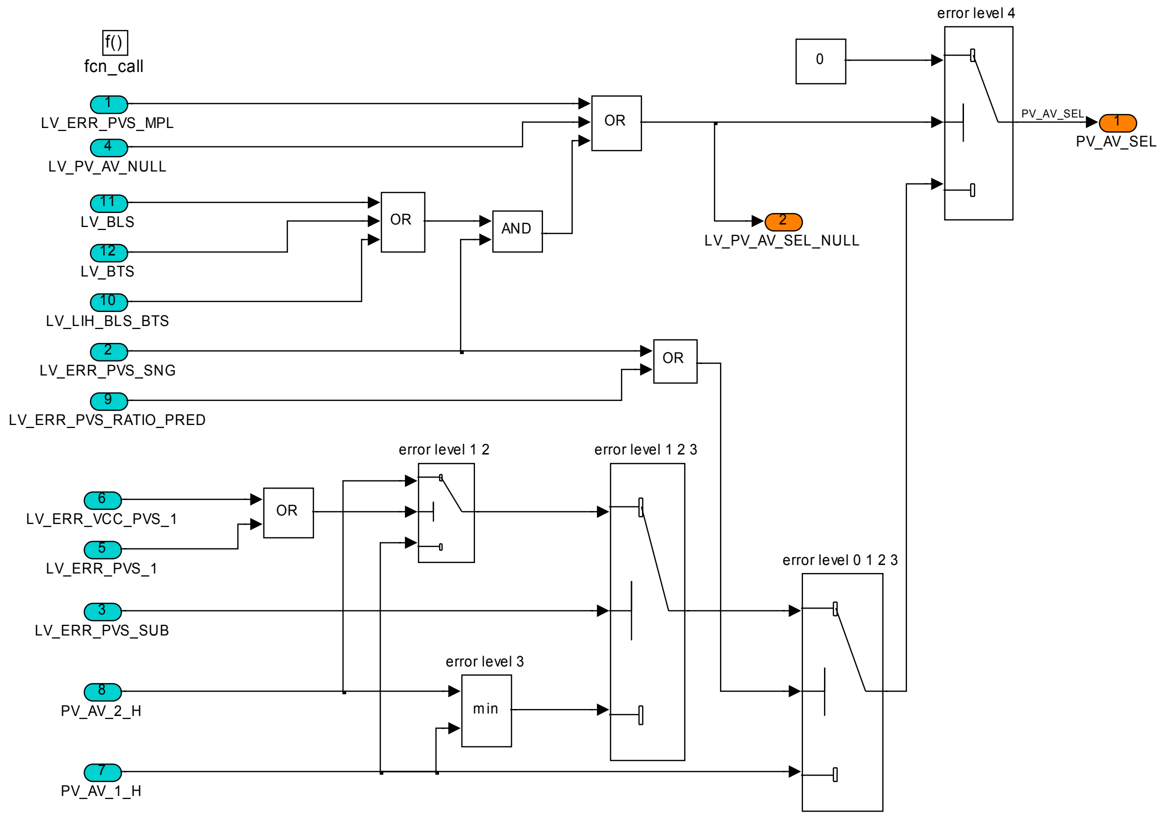


Figure 16: PV_AV Selection

Description:

The pedal value is selected depending on the PVS error state (error level) as follows:

Error level	Error bits		Error handling
0	 **and** **and**	LV_ERR_PVS_1/2 = 0 LV_ERR_VCC_PVS_1/2 = 0 LV_ERR_PVS_RATIO = 0	PV_AV_SEL= PV_AV_1_H no error: → Channel 1 is leading
1	 **and**	(LV_ERR_PVS_1=1 or LV_ERR_VCC_PVS_1=1) (LV_ERR_PVS_2=0 and LV_ERR_VCC_PVS_2=0)	PV_AV_SEL = PV_AV_2_H error on channel 1: → switch to channel 2
2	 **and**	(LV_ERR_PVS_2=1 or LV_ERR_VCC_PVS_2=1) (LV_ERR_PVS_1=0 and LV_ERR_VCC_PVS_1=0)	PV_AV _SEL = PV_AV_1_H error on channel 2: → switch to channel 1
3	 **or** **and** **and** **and** **and**	(LV_ERR_PVS_RATIO = 1 LV_ERR_PVS_RATIO_PRED = 1) LV_ERR_PVS_1 = 0 LV_ERR_PVS_2 = 0 LV_ERR_VCC_PVS_1 = 0 LV_ERR_VCC_PVS_2 = 0	PV_AV_SEL = min (PV_AV_1_H, PV_AV_2_H) ratio error: → min selection
4	 **and**	(LV_ERR_PVS_1=1 or LV_ERR_VCC_PVS_1=1) (LV_ERR_PVS_2=1 or LV_ERR_VCC_PVS_2=1)	PV_AV _SEL = 0 errors on both channels: → pedal value is set to 0%

Additionally to PVS double error (Error Level 4), PV_AV_SEL can be set to 0 % at following conditions:

– A ORNG- or VCC-error on channel x has been detected and the condition for channel switch is not fulfilled , LV_PV_AV_NULL = 1

– As in Error Level 1,2 and 3 only a one-driver demand is available, the break is used as redundant driver demand.
  If LV_ERR_PVS_SNG = 1 (any SRC-, VCC- or Ratio-error is set) and
  - a break detection (LV_BLS = 1 or LV_BTS = 1) **or**
  - a break-error LV_LIH_BLS_BTS = 1

  is present, PV_AV_SEL is set to 0 %.

4.50.4.5 PV_AV Limitations

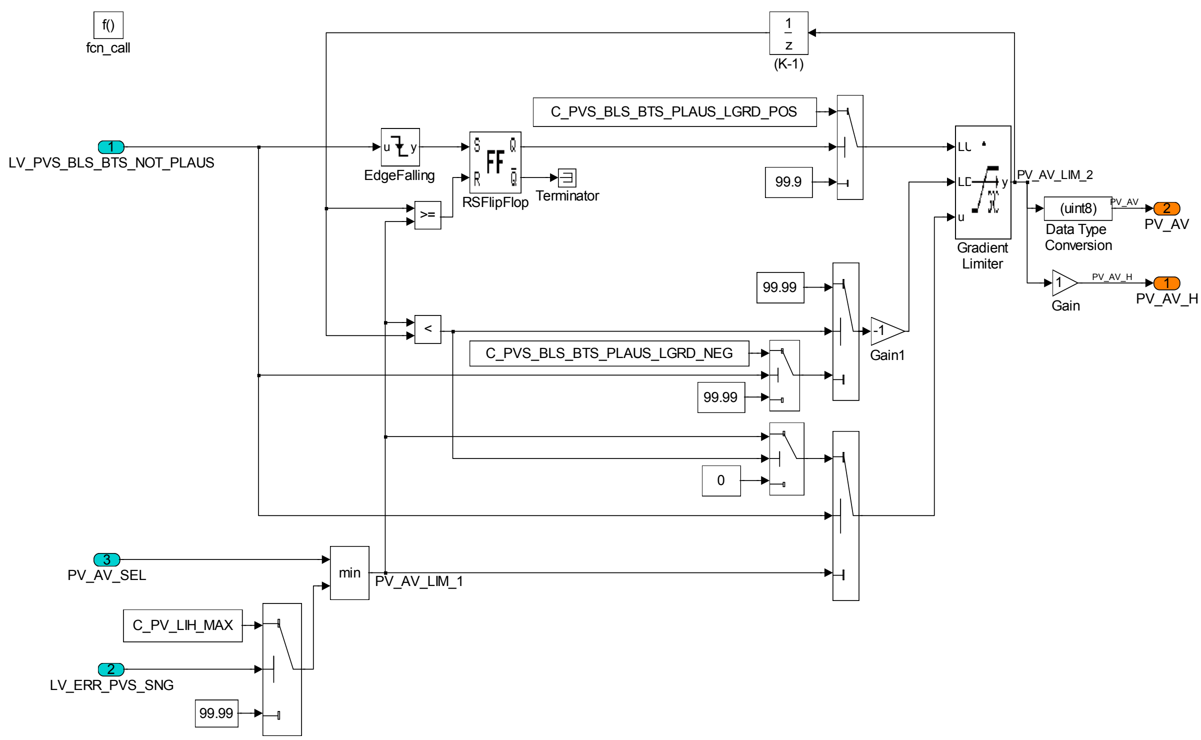


Figure 17: PV_AV Limitations

Description:

Limitations due to error level 1,2 or 3:

If LV_ERR_PVS_SNG = 1 (means only fault on one channel and/or Ratio-error) the absolute value of PV_AV_H is limited to C_PV_LIH_MAX.

Limitations due to plausibility PVS / brake

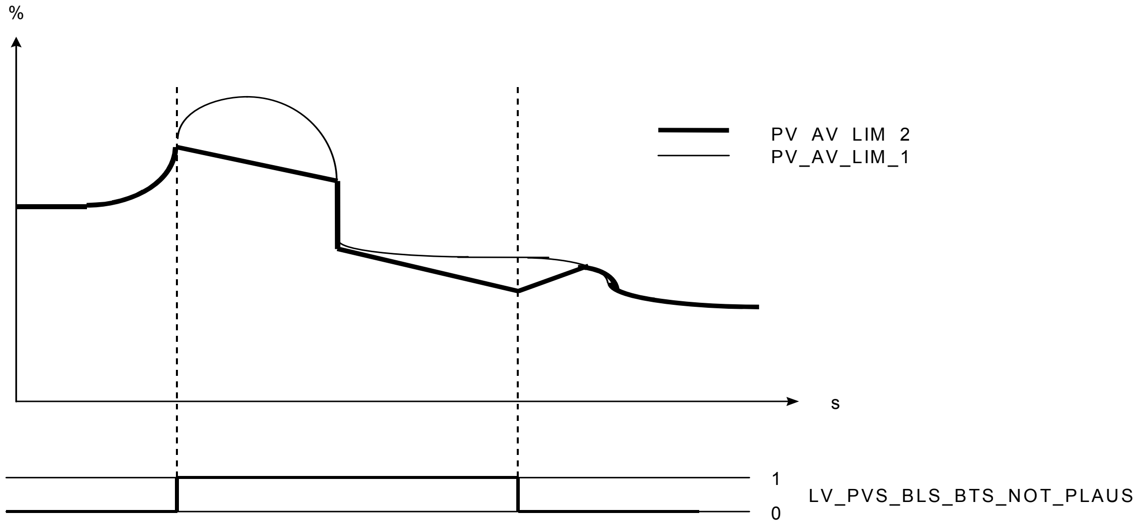


Figure 18: ramp

If an implausible state of brake and gas-pedal (LV_PVS_BLS_BTS_NOT_PLAUS=1) is recognized, the pedal value (PV_AV_LIM_2) will be decreased with C_PVS_BLS_BTS_PLAUS_LGRD_NEG by limiter. To avoid that PV_AV_LIM_2 may exceed PV_AV_LIM_1, PV_AV_LIM_2 is bound to PV_AV_LIM_1 if it drops quickly (faster than decreasing by ramp shown in Figure 18 ramp). After PVS_BLS_BTS_NOT_PLAUS = 0, PV_AV_LIM_2 will be increased with C_PVS_BLS_BTS_PLAUS_LGRD_POS by limiter up to PV_AV_LIM_1.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_PVS_THD_IS_L_1	1	0...3FFH	0...4.995117	0.004883	V
Threshold for idle speed recognition (low) of channel 1					
C_V_PVS_SP_FL_1	1	0...3FFH	0...4.995117	0.004883	V
Threshold for full load recognition of channel 1					
C_V_PVS_THD_IS_H_1	1	0...3FFH	0...4.995117	0.004883	V
Threshold for idle speed recognition (high) of channel 1					
C_V_PVS_THD_IS_L_2	1	0...3FFH	0...4.995117	0.004883	V
Threshold for idle speed recognition (low) of channel 2					
C_V_PVS_THD_IS_H_2	1	0...3FFH	0...4.995117	0.004883	V
Threshold for idle speed recognition (high) of channel 2					
C_V_PVS_SP_FL_2	1	0...3FFH	0...4.995117	0.004883	V
Threshold for full load recognition of channel 2					
C_PV_LIH_MAX	1	0...3FFH	0...99.902344	0.09766	%
Maximum value of PV_AV in case of failure					
C_PVS_BLS_BTS_PLAUS_LGRD_NEG	1	0...3FFH	0...99.902344	0.09766	%/10ms
negative gradient limitation of PV_AV at PVS_BLS_BTS_PLAUS-check					
C_PVS_BLS_BTS_PLAUS_LGRD_POS	1	0...3FFH	0...99.902344	0.09766	%/10ms
positive gradient limitation of PV_AV at PVS_BLS_BTS_PLAUS-check					
C_CTR_MAX_PVS_IS_DET	1	0...FFH	0...2.55	0.01	s
counter maximum value for idle recognition					
C_V_PVS_THD_GRD_1	1	0...3FFH	0...4.995117	0.004883	V
Threshold for gradient of V_PVS_1					
C_V_PVS_THD_GRD_2	1	0...3FFH	0...4.995117	0.004883	V
Threshold for gradient of V_PVS_2					
C_V_PVS_HYS_FIL	1	0...3FFH	0...4.995117	0.004883	V
Band width of dejitter filter					

4.51 Determination of accelerator pedal value (Appl. Inc.)

4.51.1 Application Incidences

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LIH_BLS_BTS	O/V	0 ... 1H	0 ... 1	1	-
limp home flag for error on BRAKE or CAN					

Input data:

LV_ERR_BLS_BTS			

General Information:

The application incidences describe several value- and bit-settings used as Inputs in ‘Determination of accelerator pedal value (PVS)’

Application conditions:

Activation: same than module ‘Determination of accelerator pedal value (PVS)’

Deactivation: same than module ‘Determination of accelerator pedal value (PVS)’

Initialization: LV_LIH_BLS_BTS = 0

Recurrence: same than module ‘Determination of accelerator pedal value (PVS)’

General Description:

The application incidences are executed at function call CLC_PVS_DETERM_ APPL_INC__10MS and initialized at CLC_PVS_SIG_CONV_APPL_INC__RST.

4.51.1.1 Initialization conditions

Description:

The Initialization conditions for the modules
- Determination of accelerator pedal value
- Determinations of accelerator pedal value (Appl. Inc.)

are commonly defined here

Initialization is done at following conditions

- at reset
- at transition from LV_IGK == 0 to 1
- at clearing the failure memory

Formula section:

If LV_ERR_BLS_BTS = 1

THEN LV_LIH_BLS_BTS = 1

ELSE LV_LIH_BLS_BTS = 0

4.52 Inverse accelerator pedal value (PV_CRU)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PV_CRU	O/V	0...FFH	0...99.6	0.3906	%
Inverse accelerator pedal					

Input data:

FAC_TQ_REQ	N	LV_CRU_ACT	FAC_TQ_REQ_DRIV
FAC_TQ_REQ_CRU			

FUNCTION DESCRIPTION:

General information:

During the cruise control function active, this inverse pedal value is calculated and this information could be used instead of PV_AV to TPS_SP_MDL_MAX calculation and auto transmission control.

The IP_PV_CRU is just inverted table of IP_FAC_TQ_REQ_DRIV.

General Application condition:

Activation: *always*

Deactivation: -

Initialization: PV_CRU = PV_AV

Recurrence: every 10 ms

Formula section:

If LV_CRU_ACT = 0
or FAC_TQ_REQ_DRIV > FAC_TQ_REQ_CRU
Then PV_CRU = PV_AV
Else PV_CRU = IP_PV_CRU__FAC_TQ_REQ__N

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PV_CRU	12*12	0...FFH	0...99.6	0.3906	%
LDP_FAC_TQ_REQ__PV_CRU	12	0 ... FFFFH	0 ... 1.999969	3.052E-5	-
LDP_N__PV_CRU	12	0 ... 1FE0H	0 ... 8160	1	rpm
Inverse Pedal value map					

4.53 Definition of PV_AV_CAN

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PV_AV_CAN	V/O	00...FFH	0...99.6	0,3906	%
PV_AV distributed via CAN.					

Input data:

PV_AV	LV_ERR_PVS_1	LV_ERR_VCC_PVS_1	LV_ERR_PVS_2
LV_ERR_VCC_PVS_2	PV_CRU	LV_CRU_ACT	LV_CRU_OVER_ACT
LV_MTC_CUR_OFF_REQ	LV_OFF_MTC_MON		

FUNCTION DESCRIPTION:

General information:

The PV-value is distributed to other control units via CAN.

If the EMS-ECU recognises that the accelerator pedal is not activated, then the value 00H is carried over on the CAN. If the EMS-ECU detects that the accelerator pedal is fully activated, then FEH is carried over on the CAN bus. In case of a PV-error is detected at both PV channels (PV_AV_1 and PV_AV_2) then the identifier FFH is distributed as an error identifier.

If ETC limphome is active and the throttle body is switched of (LV_MTC_CUR_OFF_REQ = 1) or ETC powerstage is disabled due to a monitoring error, PV_AV_CAN is set to FFH.

For cruise controlled driving, PV_AV_CAN is set to PV_CRU.

Formula section:

If ((LV_ERR_PVS_1 = 1 **or**
LV_ERR_VCC_PVS_1 = 1)
and (LV_ERR_PVS_2 = 1 **or**
LV_ERR_VCC_PVS_2 = 1))
or LV_MTC_CUR_OFF_REQ = 1
or LV_OFF_MTC_MON = 1

then PV_AV_CAN = FFH
else **if** (LV_CRU_ACT = 1 **and** LV_CRU_OVER_ACT = 0)
then PV_AV_CAN = PV_CRU
else **If** PV_AV > FE H
then PV_AV_CAN = FE H
else PV_AV_CAN = PV_AV

4.54 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PV_AV_MAX_TOT_DC	O/V/S	0...FFH	0...99.6	0.3906	%
Former / current driving cycle maximum accel. pedal position					
PV_AV_GRD	O/V	0...FFH	0...255	1	%/100ms
Pedal value gradient					
PV_AV_GRD_MAX_TOT_DC	O/V/S	0...FFH	0...255	1	%/100ms
Former / current driving cycle maximum gradient of accel. pedal position					

Input data:

PV_AV	LV_ES		

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

PV_AV_MAX_TOT_DC = 0

PV_AV_GRD_MAX_TOT_DC = 0

- otherwise: restored from non-volantile memory

- at LV_IGK 0->1 and reset:

PV_AV_GRD = 0

Recurrence: 10 ms

Activation: LV_ES = 0

Deactivation: -

Formula section:

If PV_AV > PV_AV_MAX_TOT_DC

Then PV_AV_MAX_TOT_DC = PV_AV

Endif

If $PV\_AV_{(n)} > PV\_AV_{(n-1)}$ \\ only for acceleration

Then $PV\_AV\_GRD = (PV\_AV_{(n)} – PV\_AV_{(n-1)}) * 10$

Else PV_AV_GRD = 0

Endif

If PV_AV_GRD > PV_AV_GRD_MAX_TOT_DC

Then PV_AV_GRD_MAX_TOT_DC = PV_AV_GRD

Endif

4.55 Definition of PV_GRD

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PV_GRD	V/O	FC00...3FFH	-100...99.902	0.0976	% / 40ms
Pedal value gradient					

Input data:

PV_AV			

Application conditions:

Initialisation: PV_GRD = 0

Recurrence: 10 ms

Activation: LV_ES = 0

Deactivation: -

FUNCTION DESCRIPTION:

General information:

The PV_GRD value shows the gradient of the accelerator pedal. It can have positive or negative values

Formula section:

$PV\_GRD = PV\_AV_{(n)} - PV\_AV_{(n-4)}$

4.56 PVS error limitations

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_MAX_PVS_LIH	O/V	0...FFFFH	0...1.999969	3.052E-5	-
Scaling factor for requested torque at clutch by pedal at PVS error					
FAC_TQ_MAX_PVS_BLS_BTS_LIM	O/V	0...FFFFH	0...1.999969	3.052E-5	-
Scaling factor for driver requested torque at clutch by pedal at PVS / Brake state detection					
LV_ACT_PVS_BLS_BTS_TQ_LIM	O/V	0...1H	0...1	1	-
Flag for activation limitation of driver requested torque at clutch (Brake / PVS)					

Input data:

TQ_MIN_CLU	TQ_MAX_CLU	LV_ACT_PVS_LIH	LV_IGK
VS	LV_BLS	LV_BTS	PV_AV_SEL
LV_AT	FAC_TQ_REQ_PV	LV_ERR_VS	

FUNCTION DESCRIPTION:

General information:

Due to Information of pedal value sensor additionally constrains can be performed to restrict vehicle dynamic:
- torque limitation due to PVS errors
- torque limitation due to PVS / Brake state (for AT vehicles only)

General Application conditions:

Activation: LV_IGK = 1

Deactivation: -

Initialization: FAC_TQ_MAX_PVS_LIH = 0

FAC_TQ_MAX_PVS_BLS_BTS_LIM = 0

LV_ACT_PVS_BLS_BTS_TQ_LIM = 0

Initialization conditions: same as in module ‘Diagnosis of Pedal value sensor (Application Incidences)’

Recurrence: 10 ms

Signal flow diagram:

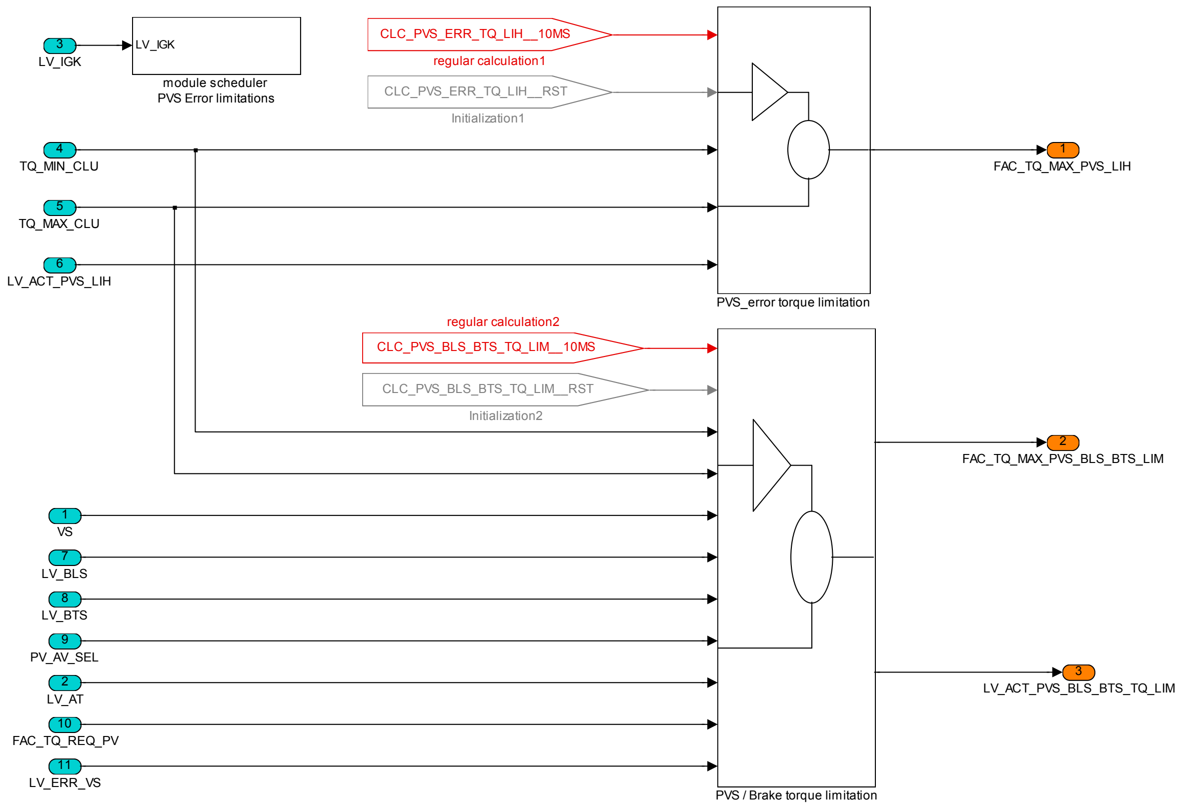


Figure 1: PVS error limitations

The module PVS error limitation is executed at function-call CLC_PVS_ERR_LIM__10MS.
Initialization is done at CLC_PVS_ERR_LIM__RST.

Calculation order (regular calculation):

1. PVS error torque limitation — on fcn-call CLC_PVS_ERR_TQ_LIH__10MS
2. PVS / Brake torque limitation — on fcn-call CLC_PVS_BLS_BTS_TQ_LIM__10MS

Initialization order:

1. PVS error torque limitation — on fcn-call CLC_PVS_ERR_TQ_LIH__RST
2. PVS / Brake torque limitation — on fcn-call CLC_PVS_BLS_BTS_TQ_LIM__RST

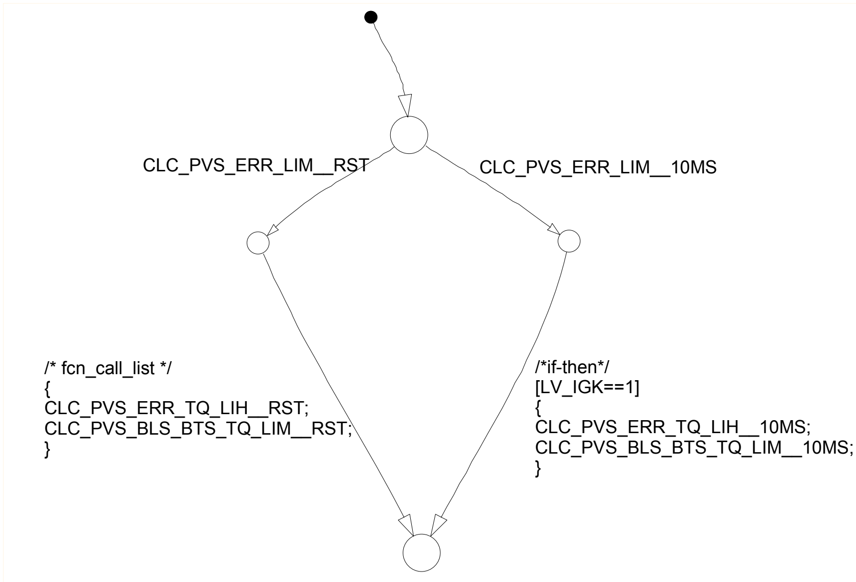


Figure 2: module scheduler “PVS error limitations”

4.56.1 PVS error torque limitation

General information:

As a consequence of a PVS error there may be a requirement to restrict the vehicle dynamic behavior. Therefore a limited scaling factor is calculated to restrict the driver requested torque. Additionally the positive gradient of scaling factor is limited to restrict vehicle acceleration.

Application conditions:

Activation: LV_IGK = 1

Deactivation: -

Signal flow diagram:

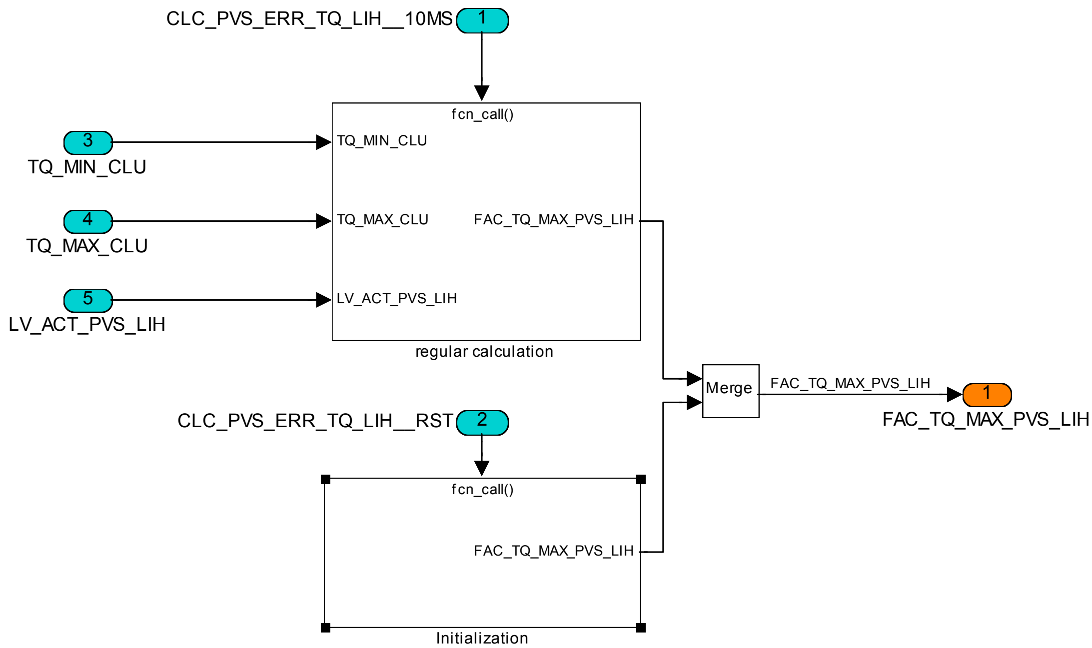


Figure 2: PVS error torque limitation

Description:

Regular calculation is done at function call CLC_PVS_ERR_TQ_LIH__10MS. Initialization is done at CLC_PVS_ERR_TQ_LIH__RST.

4.56.1.1 Initialization

At Initialization FAC_TQ_MAX_PVS_LIH is set to 0.

4.56.1.2 Regular calculation

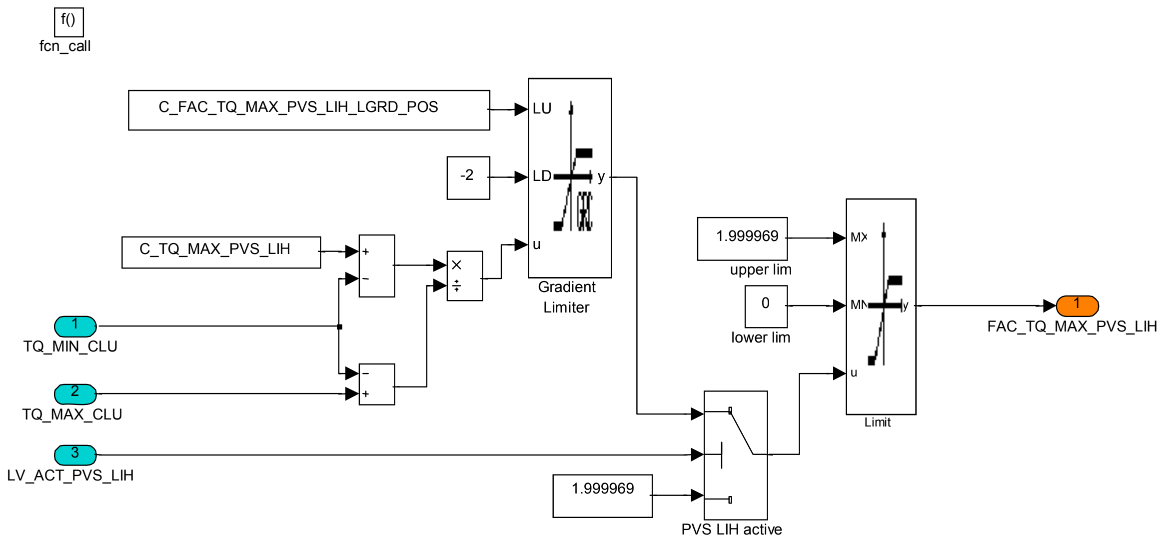


Figure 2: regular calculation

Description:

If a PVS error is debounced (LV_ACT_PVS_LIH = 1) the driver requested torque by pedal is limited to the maximum value C_TQ_MAX_PVS_LIH. As in torque management a scaling factor is required for calculation of engine torque, the limitation is done by the calculation of the scaling factor FAC_TQ_MAX_PVS_LIH. The positive gradient is limited to C_FAC_TQ_MAX_PVS_LIH_LGRD_POS.

If no PVS error is debounced (LV_ACT_PVS_LIH = 0), the factor is set to 1.999969.

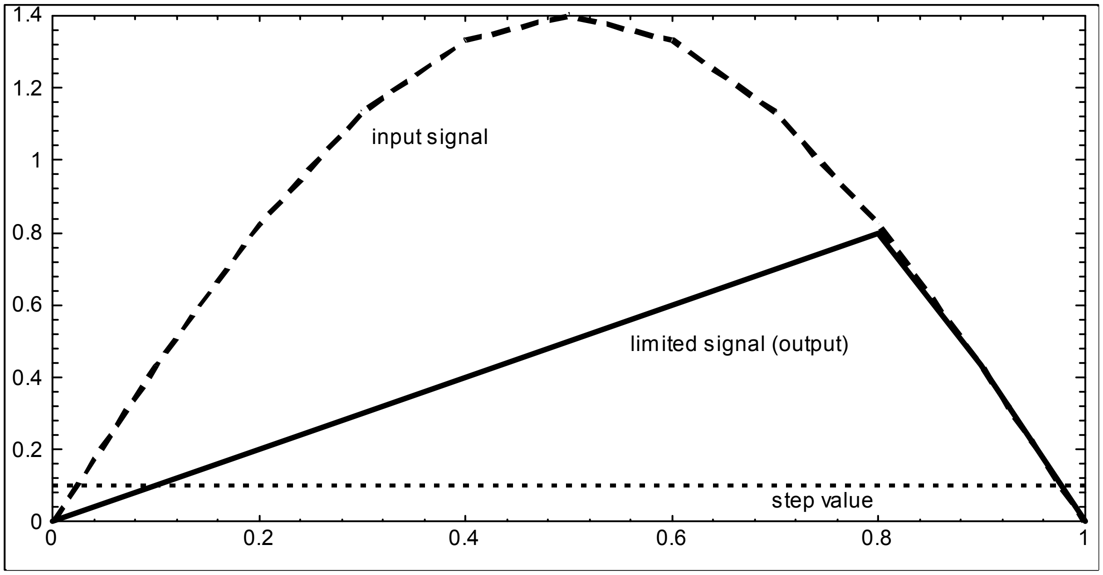
Formula section:

IF LV_ACT_PVS_LIH == 1
THEN FAC_TQ_MAX_PVS_LIH = (C_TQ_MAX_PVS_LIH – TQ_MIN_CLU) / (TQ_MAX_CLU -TQ_MIN_CLU)

{ The positive gradient is limited to C_FAC_TQ_MAX_PVS_LIH_LGRD_POS }

ELSE FAC_TQ_MAX_PVS_LIH = 1.999969
ENDIF

{ FAC_TQ_MAX_PVS_LIH is limited to max. 1.999969 and min. 0 } gradient limiter:



4.56.2 PVS / Brake torque limitation

General information:

To prevent a move away of the vehicle if both pedals (brake and gas pedal) are pressed, the driver requested torque can be limited to ensure that braking torque is higher than engine torque. Therefore a limited scaling factor is calculated depending on pedal activation.

General Application conditions:

Activation: LV_IGK = 1
LV_AT = 1

Deactivation: -

Signal flow diagram:

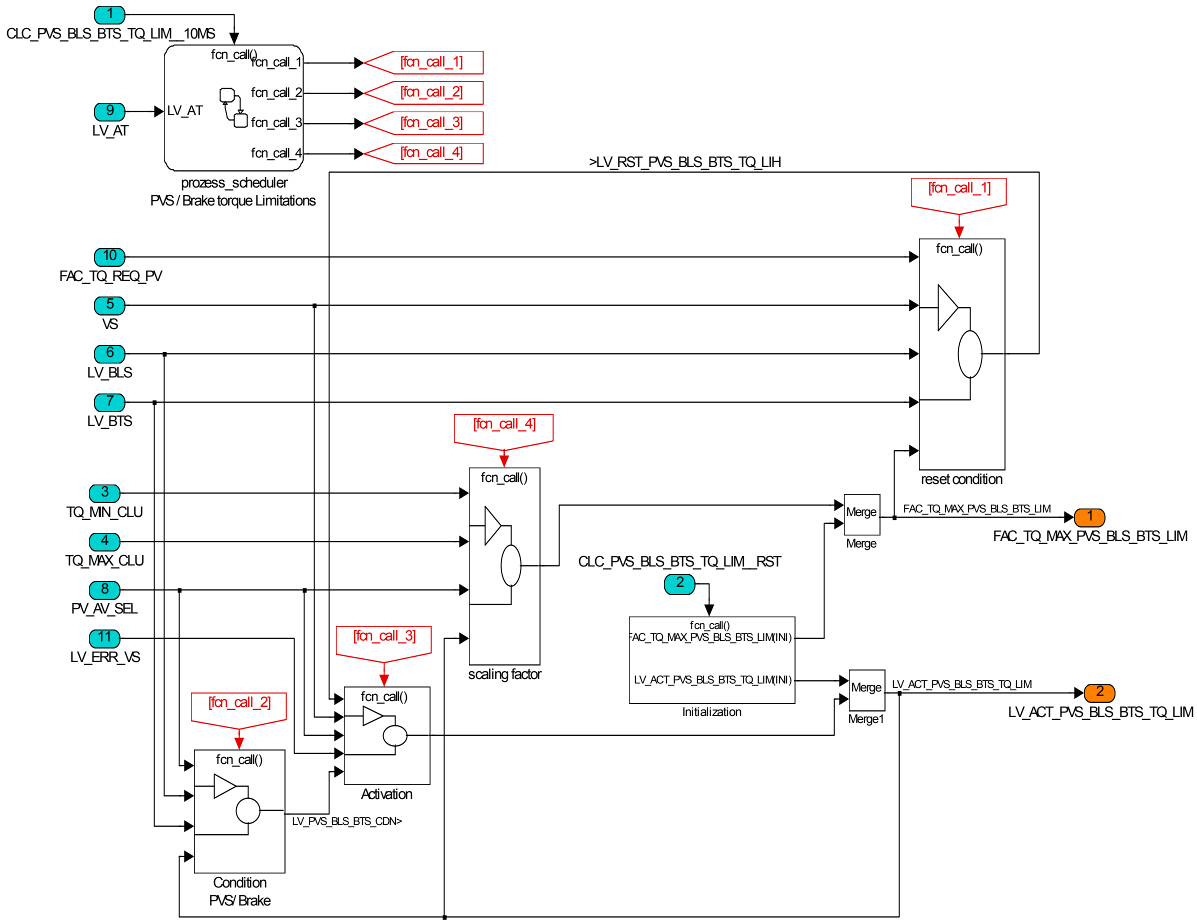


Figure 3: PVS / Brake torque limitation

Description:

Initialization:
Is done at function call CLC_PVS_BLS_BTS_TQ_LIM__RST

Calculation order:
(On function call CLC_PVS_BLS_BTS_TQ_LIM__10MS if LV_AT=1 only):

1. Reset condition on fcn_call_1
2. Condition PVS/ Brake on fcn_call_2
3. Activation on fcn_call_3
4. Scaling factor on fcn_call_4

4.56.2.1 Initialization

At Initialization FAC_TQ_MAX_PVS_BLS_BTS_LIM and LV_ACT_PVS_BLS_BTS_TQ_LIM are set = 0

4.56.2.2 Reset condition

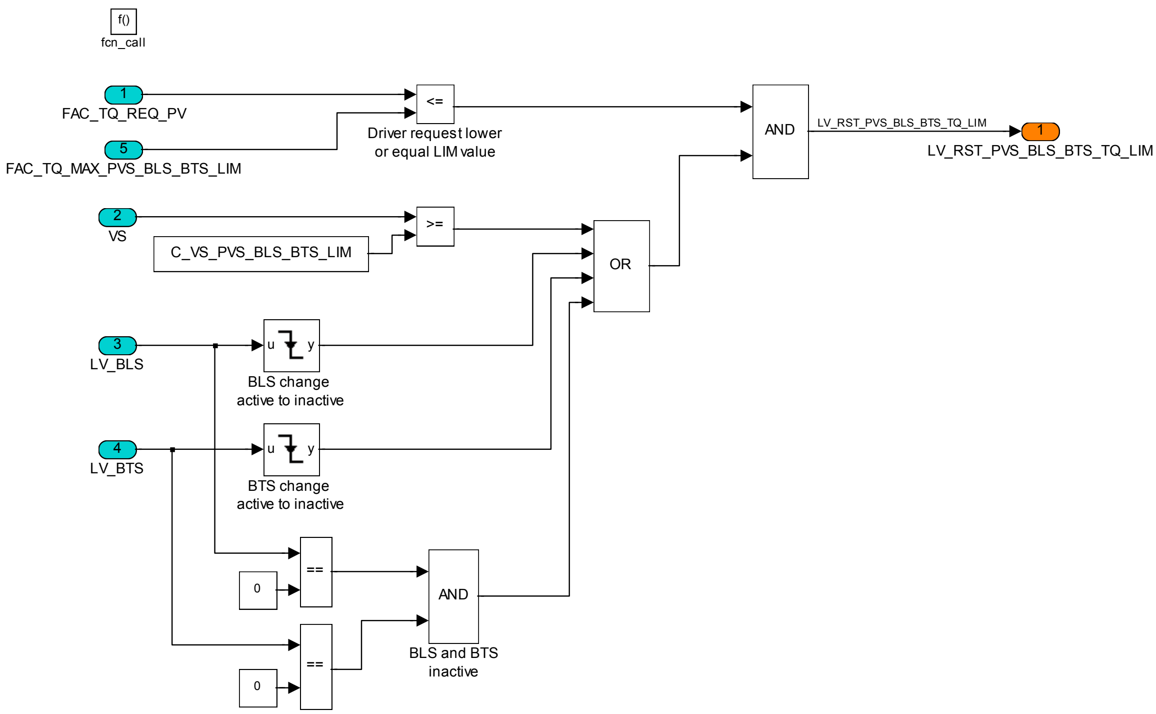


Figure 4: Reset condition

Description:

A limited scaling factor is calculated (after activation was done) till the reset conditions are fulfilled.

The reset flag LV_RST_PVS_BLS_BTS_TQ_LIM is set if the calculated scaling factor FAC_TQ_REQ_PV (real driver request) is lower or equal the limited scaling factor FAC_TQ_MAX_PVS_BLS_BTS_LIM **and**

-the vehicle speed is higher or equal C_VS_PVS_BLS_BTS_LIM or

-the brake changes to inactive (indicated by LV_BLS) or

-the brake changes to inactive (indicated by LV_BTS) or

-the brake is inactive (indicated if LV_BLS and LV_BTS are 0).

Formula section:

IF FAC_TQ_REQ_PV <= FAC_TQ_MAX_PVS_BLS_BTS_LIM **AND**

[VS >= C_VS_PVS_BLS_BTS_LIM **or**

($LV\_BLS_{(K-1)}$ == 1 and $LV\_BLS_{(K)}$ == 0 **)** **or**

($LV\_BTS_{(K-1)}$ == 1 and $LV\_BTS_{(K)}$ == 0 **)** **or**

(LV_BLS == 0 and LV_BTS == 0 **)** **]**

THEN LV_RST_PVS_BLS_BTS_TQ_LIM = 1

ELSE LV_RST_PVS_BLS_BTS_TQ_LIM = 0

ENDIF

4.56.2.3 Condition PVS / Brake

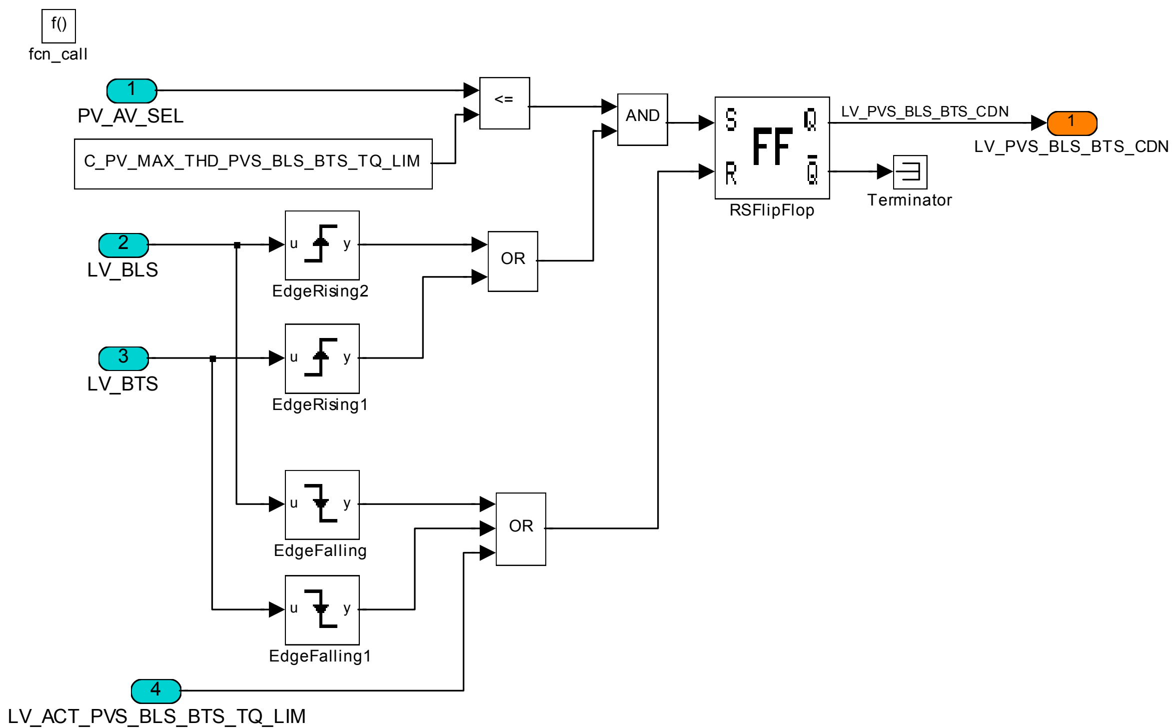


Figure 5: Condition PVS / Brake

Description:

The first condition which has to be fulfilled for a calculation of a limited scaling factor is the detection of brake activation (LV_BLS or LV_BTS) while the gas pedal is released (PV_AV_SEL <= C_PV_MAX_THD_PVS_BLS_BTS_TQ_LIM). The condition is reset if a transition of one brake bit from 1 to 0 is detected or the limitation has got active (LV_ACT_PVS_BLS_BTS_TQ_LIM = 1).

Formula section:

IF LV_ACT_PVS_BLS_BTS_TQ_LIM$_{(K-1)}$ == 1 **or**

(LV_BLS$_{(K-1)}$ == 1 and LV_BLS$_{(K)}$ == 0 **)** **or**

(LV_BTS$_{(K-1)}$ == 1 and LV_BTS$_{(K)}$ == 0 **)**

THEN LV_PVS_BLS_BTS_CDN = 0

ELSE

IF PV_AV_SEL <= C_PV_MAX_THD_PVS_BLS_BTS_TQ_LIM

AND

[**(** LV_BLS$_{(K-1)}$ == 0 and LV_BLS$_{(K)}$ == 1 **)** **or**

(LV_BTS$_{(K-1)}$ == 0 and LV_BTS$_{(K)}$ == 1 **)** **or** **]**

THEN LV_PVS_BLS_BTS_CDN = 1

ENDIF

ENDIF

4.56.2.4 Activation

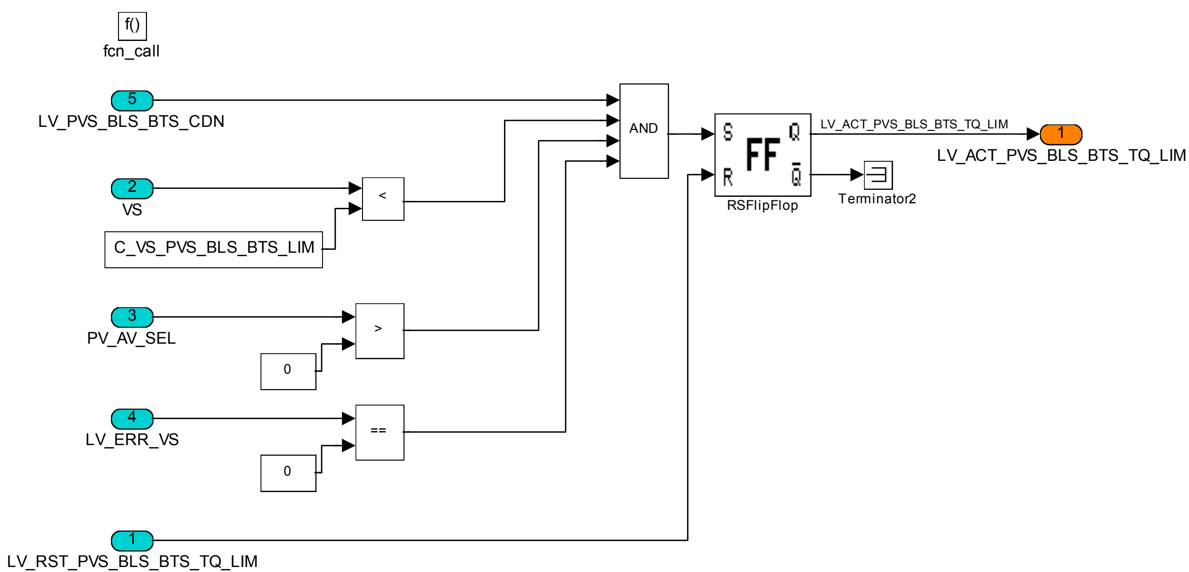


Figure 6: Activation

Description:

A limited scaling factor will be calculated if the flag LV_ACT_PVS_BLS_BTS_TQ_LIM is set.

The bit is set if the first condition has been fulfilled (LV_PVS_BLS_BTS_CDN = 1) and the vehicle speed is below C_VS_PVS_BLS_BTS_LIM threshold and no fault is detected on VS signal (LV_ERR_VS = 0) and the pedal value is higher 0% (PV_AV_SEL >0%).

The reset flag LV_RST_PVS_BLS_BTS_TQ_LIM resets the activation bit.

Formula section:

IF LV_RST_PVS_BLS_BTS_TQ_LIM == 1

THEN LV_ACT_PVS_BLS_BTS_TQ_LIM = 0

ELSE

IF LV_PVS_BLS_BTS_CDN == 1 **AND**

VS < C_VS_PVS_BLS_BTS_LIM **AND**

PV_AV_SEL > 0 **AND**

LV_ERR_VS == 0

THEN LV_ACT_PVS_BLS_BTS_TQ_LIM = 1

ENDIF

ENDIF

4.56.2.5 Scaling factor

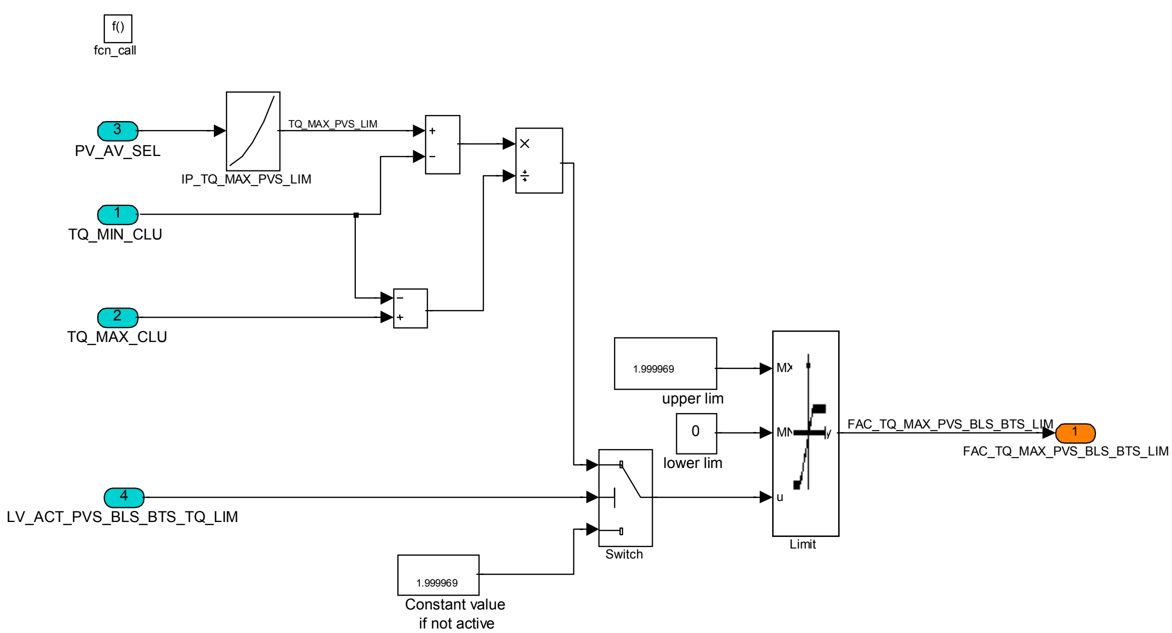


Figure 7: Scaling factor

Description:

If the activation conditions are fulfilled (LV_ACT_PVS_BLS_BTS_TQ_LIM ==1) the engine torque which can be requested by pedal is restricted to IP_TQ_MAX_PVS_LIM. As in torque management a scaling factor is required to calculate the engine torque, the scaling factor FAC_TQ_MAX_PVS_BLS_BTS_LIM is calculated for limitation.

If LV_ACT_PVS_BLS_BTS_TQ_LIM == 0 the factor is set to 1.999969.

Formula section:

TQ_MAX_PVS_LIM = IP_TQ_MAX_PVS_LIM (PV_AV_SEL)

IF LV_ACT_PVS_BLS_BTS_TQ_LIM ==1

THEN FAC_TQ_MAX_PVS_BLS_BTS_LIM = (TQ_MAX_PVS_LIM – TQ_MIN_CLU) / (TQ_MAX_CLU – TQ_MIN_CLU)

ELSE FAC_TQ_MAX_PVS_BLS_BTS_LIM = 1.999969

ENDIF

{ FAC_TQ_MAX_PVS_BLS_BTS_LIM is limited to max. 1.999969 and min. 0 }

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_TQ_MAX_PVS_LIH	1	8000...7FFFH	-1.024E3...1.02397E3	0.03125	Nm
Maximum engine torque requested by gas pedal at PVS error					
C_FAC_TQ_MAX_PVS_LIH_LGRD_POS	1	0...FFFFH	0...1.999969	3.052E-5	-
Positive gradient limitation of scaling factor					
C_VS_PVS_BLS_BTS_LIM	1	0...FFH	0...255	1	Km/h
VS threshold for PVS/brake torque limitation					
C_PV_MAX_THD_PVS_BLS_BTS_TQ_LIM	1	0...3FFH	0...99.9	0.09765	%
Maximum pedal value for limitation activation					
IP_TQ_MAX_PVS_LIM	6	0...FFFFH	-1.024E3...1.02397E3	0.03125	Nm
LDP_PV_AV_SEL_IP_TQ_MAX_PVS_LIM	6	0...3FFH	0...99.9	0.09765	%
Maximum allowed engine torque requested by gas pedal for PVS/brake torque limitation					

4.57 Kick down recognition / adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PVS_KD	O/V	0...1H	0...1	1	-
Flag for Kick-Down-Recognition (internally recognized from both PVS-Voltages)					
LV_KD	O/V	0...1H	0...1	1	-
Flag for Kick-Down-Recognition					
V_PVS_MAX_1	O/V	0...3FFH	0...4.995117	0.004883	V
Maximum voltage at upper mechanically stop PVS-channel 1					
V_PVS_MAX_2	O/V	0...3FFH	0...4.995117	0.004883	V
Maximum voltage at upper mechanically stop PVS-channel 2					

Input data:

V_PVS_FIL_1	V_PVS_FIL_2	LV_ERR_PVS_PRED	LV_ERR_PVS
LV_LIH_PVS_MOVE	LV_AT	LV_IGK	

FUNCTION DESCRIPTION:

General information:

The voltages of two PVS-channels depending on the PVS error system-status determine the Kick-Down operation.

General Application conditions:

Activation: LV_IGK == 1

LV_AT == 1

Deactivation: -

Initialization: LV_PVS_KD = 0

LV_KD = 0

V_PVS_MAX_1 = C_V_PVS_SP_MAX_1

V_PVS_MAX_2 = C_V_PVS_SP_MAX_2

Initialization conditions: same as in module ‘ Determination of accelerator pedal value (Application Incidences)

Recurrence: every 10 ms

Signal flow diagram:

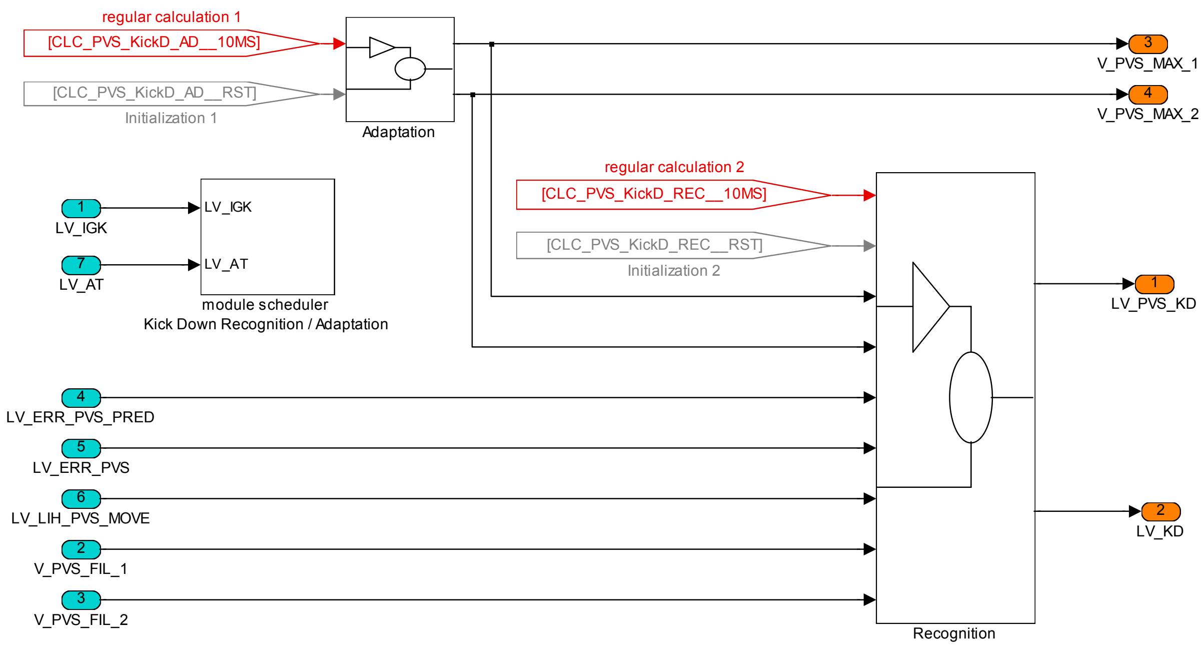


Figure 1: Kick Down Recognition / Adaptation

Description:

The module Kick Down Recognition / Adaptation is executed at function call CLC_PVS_KickD__10MS. Initialization is done at CLC_PVS_KickD__RST.

Calculation order (regular calculation):

1. Adaptation
2. Recognition

Initialization order:

1. Adaptation
2. Recognition

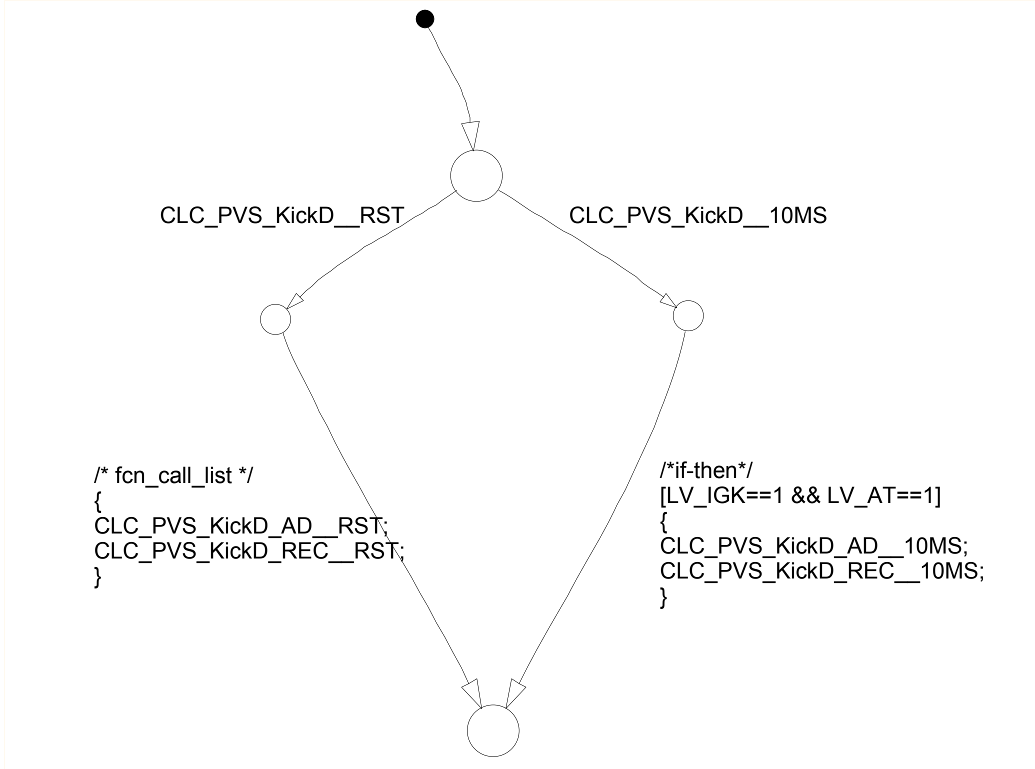


Figure 2: module scheduler “Kick Down Recognition / Adaptation”

4.57.1 Adaptation

General:

V_PVS_MAX_1 = C_V_PVS_SP_MAX_1 for channel 1 and
V_PVS_MAX_2 = C_V_PVS_SP_MAX_2 for channel 2.

In this case the calculation is only necessary at initialization and not every 10 ms.

4.57.2 Kick Down Recognition

Signal flow diagram:

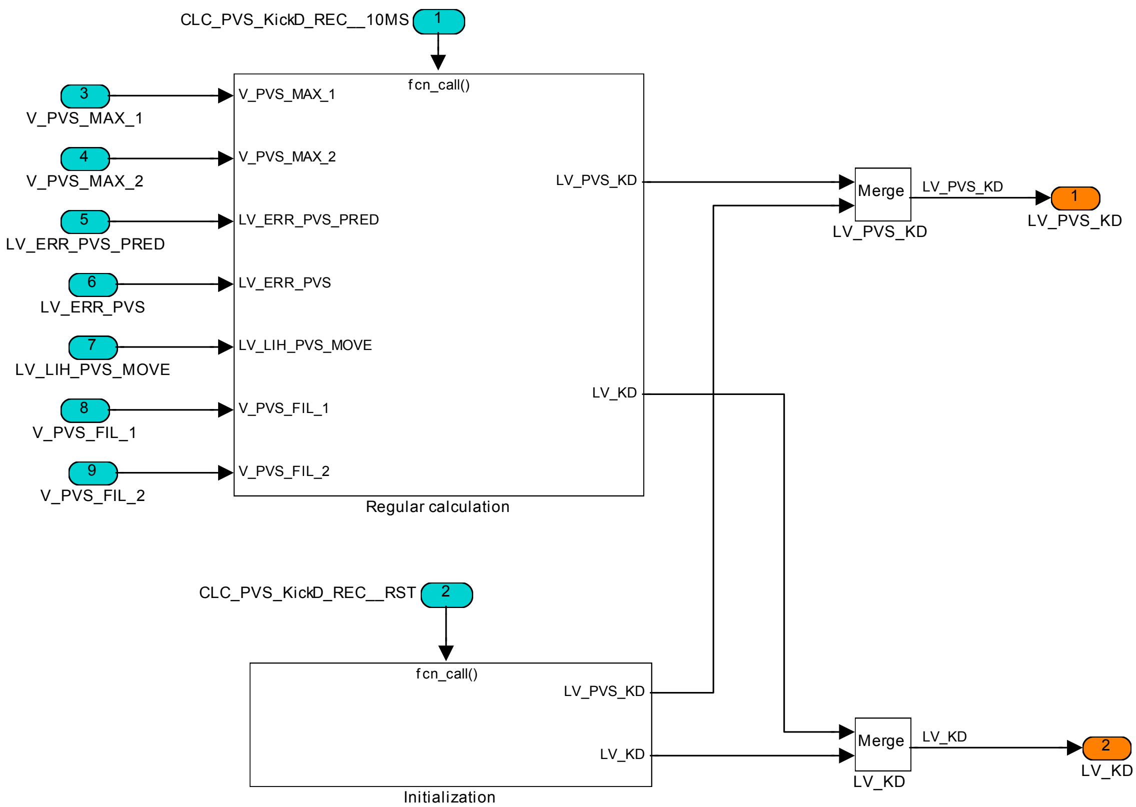


Figure 3: Kick Down Recognition

Description:

Regular calculation is done at function call CLC_PVS_KickD_REC__10MS. Initialization is done at CLC_PVS_KickD_REC__RST.

4.57.2.1 Initialization

At Initialization LV_PVS_KD and LV_KD are set to 0.

4.57.2.2 Regular calculation

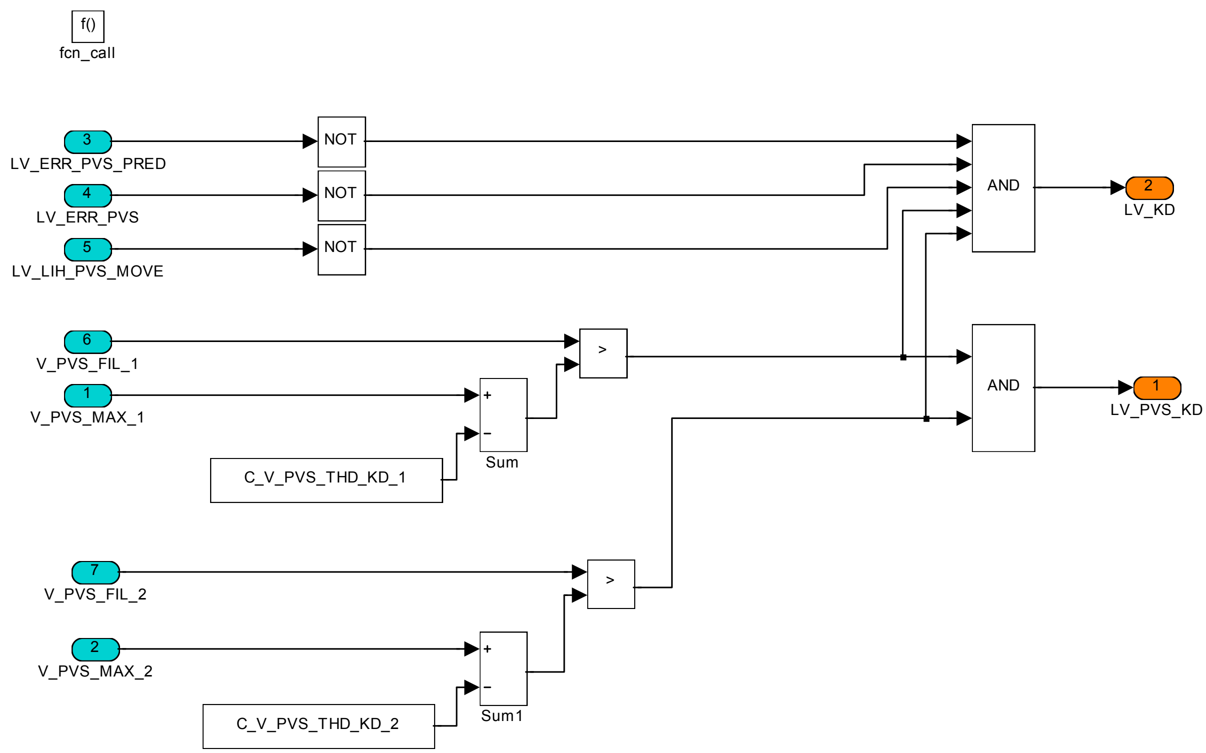


Figure 4: Regular calculation

Description:

Kick-Down recognition is done if there is no PVS-error predicted or present and the filtered signals of both channels exceed their thresholds.

Formula section:

IF	V_PVS_FIL_1 > (V_PVS_MAX_1 - C_V_PVS_THD_KD_1)	**AND**
	V_PVS_FIL_2 > (V_PVS_MAX_2 - C_V_PVS_THD_KD_2)	**AND**
	LV_AT = 1	**AND**
	LV_IGK = 1	
THEN	LV_PVS_KD = 1	
ELSE	LV_PVS_KD = 0	
ENDIF		

IF	NOT (LV_ERR_PVS_PRED == 1)	**AND**
	NOT (LV_ERR_PVS == 1)	**AND**
	NOT (LV_LIH_PVS_MOVE == 1)	**AND**
	LV_PVS_KD == 1	
THEN	LV_KD = 1	
ELSE	LV_KD = 0	
ENDIF		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_PVS_SP_MAX_1	1	0...3FFH	0...4.995117	0.004883	V
Maximum voltage of PVS-channel 1					
C_V_PVS_SP_MAX_2	1	0...3FFH	0...4.995117	0.004883	V
Maximum voltage of PVS-channel 2					
C_V_PVS_THD_KD_1	1	0...3FFH	0...4.995117	0.004883	V
Difference between V_PVS_MAX_1 and Kick-Down-Recognition (channel 1)					
C_V_PVS_THD_KD_2	1	0...3FFH	0...4.995117	0.004883	V
Difference between V_PVS_MAX_2 and Kick-Down-Recognition (channel 2)					

4.58 VIM variables

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VIM_AV	V/O	0...FFH	0...0.99609	3.9063e-3	[-]
Actual VIM position					
VIM_AV_TMP_1	-	0...FFH	0...0.99609	3.9063e-3	[-]
Actual VIM position (rising feedback characteristic)					
VIM_AV_TMP_2	-	0...FFH	0...0.99609	3.9063e-3	[-]
Actual VIM position (falling feedback characteristic)					

Input data:

LV_IGK	V_VIM	V_VIM_AD_LONG	V_VIM_AD_SHO
CONF_VIM			

FUNCTION DESCRIPTION:

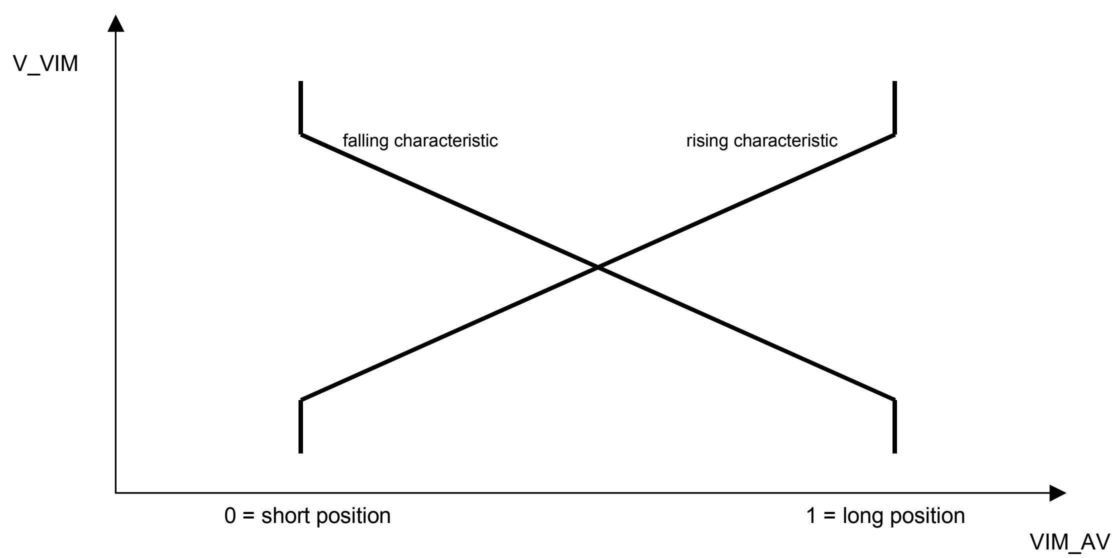
General information:

The calculation of the actual position of the variable intake manifold based on the feedback signal of the VIM actuator (V_VIM).

The values 0 and 1 point out the inactive (short intake length) or the switched (long intake length) position.

The measured voltage at the potentiometer is V_VIM. With the two adapted limitation stop values (V_VIM_AD_LONG/ SHO), V_VIM is transferred into VIM_AV.

The feedback characteristic (rising or falling characteristic) can be switched by a logical constant (LC_VIM_TYPE_SWI = 1 supports rising feedback characteristic).



Application conditions:

Initialisation: at LV_IGK = 0 → 1

VIM_AV = C_VIM_AV_SUB

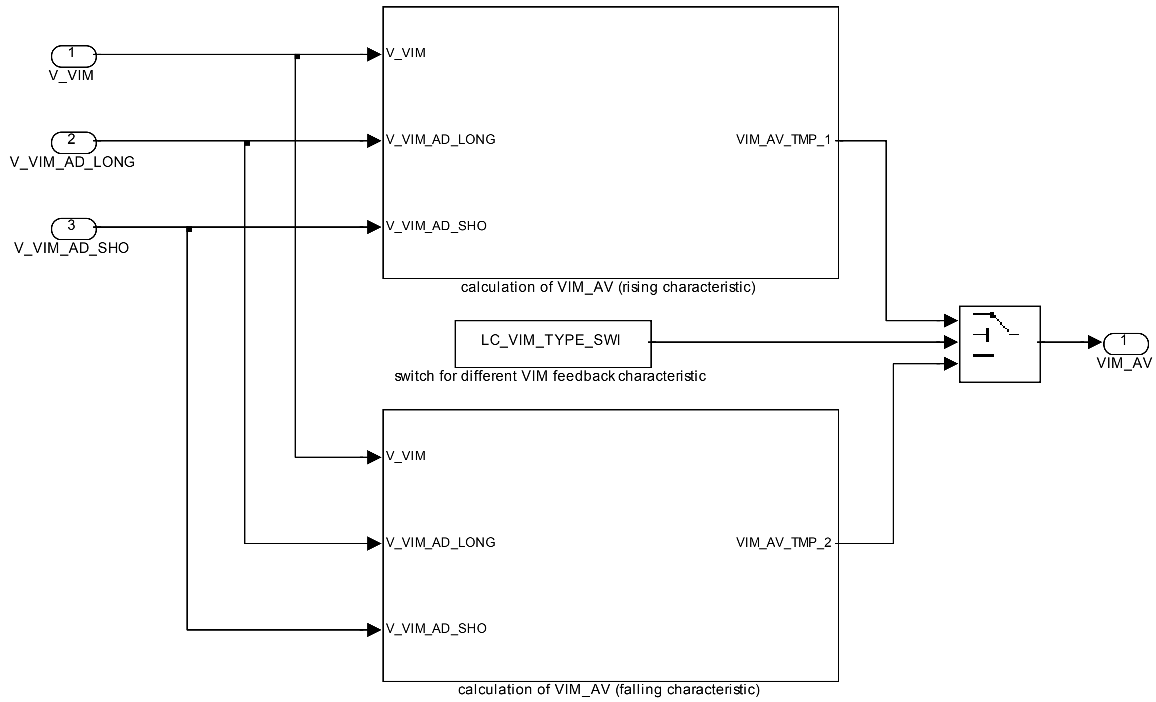
Recurrence: 20 ms

Activation: LV_IGK = 1 and CONF_VIM = 2

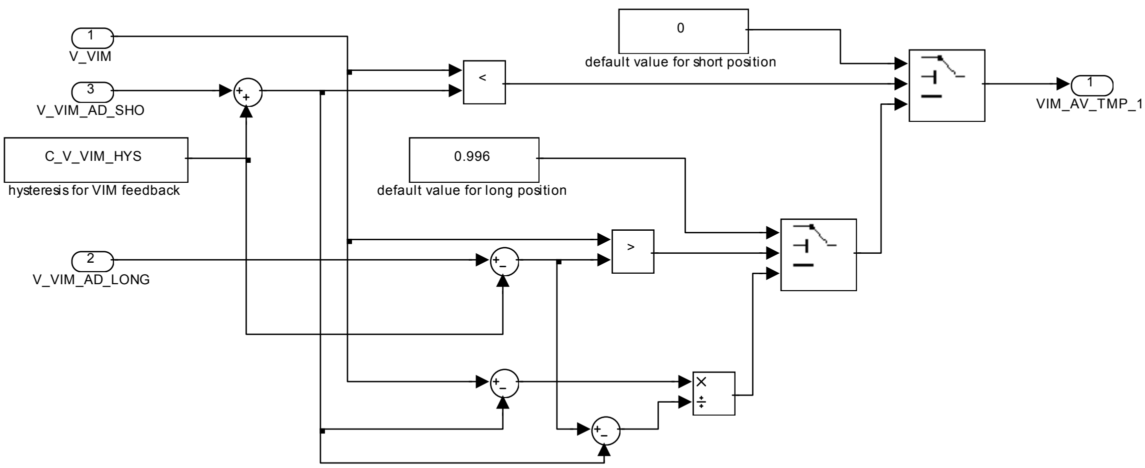
Deactivation: LV_IGK = 0

Signal flow diagram:

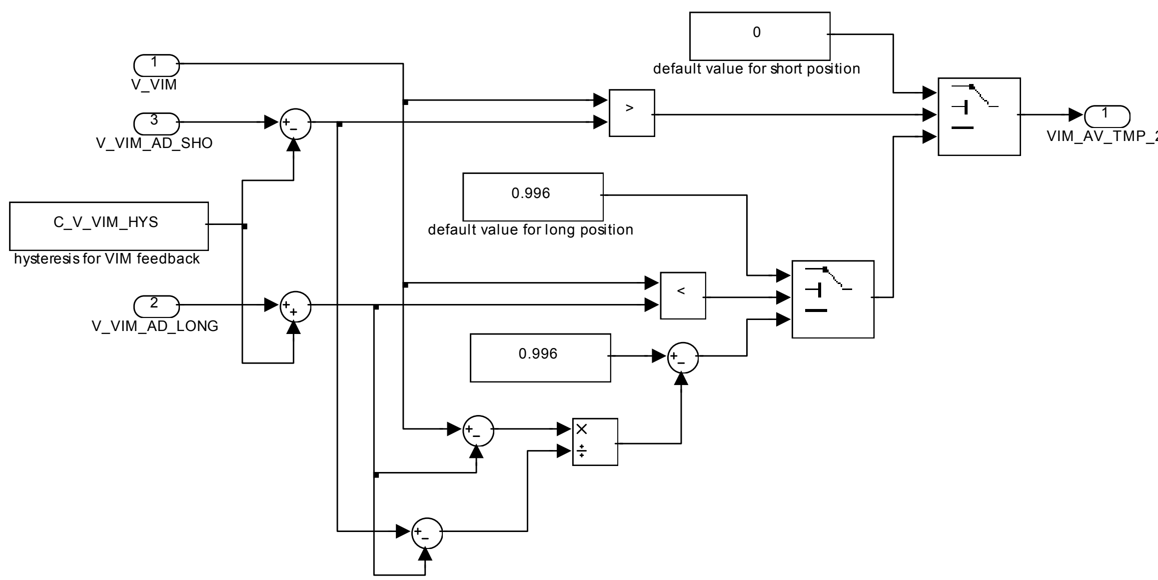
general overview:



calculation of VIM_AV with rising feedback characteristic:



calculation of VIM_AV with falling feedback characteristic:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_VIM_HYS	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Hysteresis for detection of long or short stop of VIM position					
C_VIM_AV_SUB	1	0...FFH	0...0.99609	3.9063e-3	[-]
Initialization value of VIM_AV					
LC_VIM_TYPE_SWI	1	0...1H	0...1	1	[-]
Switch for different VIM feedback characteristic					

4.59 Air Conditioning Pressure variables

4.59.1 Air Conditioning Pressure

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ACP	V/O	0...FFH	0...510	2	PSI
Air conditionning pressure					
ACP_RATIO	V	0...FFFFH	0...99.99847	0.001526	%
Air conditionning pressure ratio between V_ACP_MES and reference voltage					

Input data:

C_ACP_MIN_DIAG	V_ACP_MES	C_ACP_MAX_DIAG	VCC_SENS_SUB_MES

General information:

Acquisition reccurence : 100 ms

Formula section:

If V_ACP_MES >= C_ACP_MIN_DIAG

and V_ACP_MES <= C_ACP_MAX_DIAG

then ACP_RATIO = [V_ACP_MES / VCC_SENS_SUB_ MES] * 100

ACP = [ACP_RATIO – C_ACP_OFS] / C_ACP_SLOP

else ACP = 0

see Diagnosis and Limp home strategy in chapter A

endif

If the difference between the old and the new values of ACP exceeds the permissible gradient C_ACP_GRD_MAX, then the value remains unchanged. During the next measurement, if the maximum gradient is exceeded again, the new measured value will be stored in the RAM.

Errors are not stored with the monitoring of gradients. The only purpose is to extract invalid measured values (malfunctions).

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ACP_GRD_MAX	1	0...FFH	0...510	2	PSI
Maximum air conditioning pressure gradient					
C_ACP_OFS	1	0...FFFFH	0...99.99847	0,001526	%
Off value to calculate ACP					
C_ACP_SLOP	1	0...FFFFH	0...99.99847	0,001526	%/PSI
Gain value to calculate ACP					

4.59.2 AC pressure state STATE_ACP

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_ACP	V/O	0H 1H 2H 3H	ACP_L ACP_M_1 ACP_M_2 ACP_H	1	-
AC pressure state for cooling fans.					

Input data:

ACP	LV_ERR_ACP	LV_ERR_VCC_SENS_SUB	

FUNCTION DESCRIPTION:

General information:

The cooling fans are controlled according to the A/C pressure. For this purpose, the A/C pressure state is determined.

Formula section:

Initialization: At ECU initialization, STATE_ACP is set to ACP_L.

If (STATE_ACP = ACP_L **and** ACP **>** C_ACP_1_CFA)
then STATE_ACP **= ACP_M_1**

If (STATE_ACP = ACP_M_1 **and** ACP **<** C_ACP_1_CFA - C_ACP_HYS_1)
then STATE_ACP **= ACP_L**

If (STATE_ACP = ACP_M_2 **and** ACP **<** C_ACP_2_CFA - C_ACP_HYS_2)
then STATE_ACP **= ACP_M1**
If (STATE_ACP = ACP_M_1 **and** ACP **>** C_ACP_2_CFA)
then STATE_ACP **= ACP_M_2**

If (STATE_ACP = ACP_M_2 **and** ACP **>** C_ACP_3_CFA)
then STATE_ACP **= ACP_H**

If (STATE_ACP = ACP_H **and** ACP **<** C_ACP_3_CFA - C_ACP_HYS_3)
then STATE_ACP **= ACP_M_2**

Remark : C_ACP_1_CFA **<** C_ACP_2_CFA **<** C_ACP_3_CFA

Limp Home :

If LV_ERR_ACP = 1 *(Error currently present on AC pressure transducer)*

or LV_ERR_VCC_SENS_SUB = 1 *(Error currently present on voltage sypply for AC pressure transducer)*

then STATE_ACP = ACP_H

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ACP_1_CFA	1	0...FFH	0...510	2	PSI
AC pressure threshold for fans management.					
C_ACP_2_CFA	1	0...FFH	0...510	2	PSI
AC pressure threshold for fans management.					
C_ACP_3_CFA	1	0...FFH	0...510	2	PSI
AC pressure threshold for fans management.					
C_ACP_HYS_1	1	0...FFH	0...510	2	PSI
AC pressure hysteresis for STATE_ACP transition ACP_M_1 to ACP_L					
C_ACP_HYS_2	1	0...FFH	0...510	2	PSI
AC pressure hysteresis for STATE_ACP transition ACP_M_2 to ACP_M_1					
C_ACP_HYS_3	1	0...FFH	0...510	2	PSI
AC pressure hysteresis for STATE_ACP transition ACP_H to ACP_M_2					

4.60 Fuel pressure acquisition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FUP	V/O	0...FFFFH	0...173888	2.65	hPa
fuel pressure					

Input data:

AMP			

FUNCTION DESCRIPTION:

General information:

This file calculates the fuel pressure for port injection returnless fuel systems. The pressure is regulated by a mechanical pressure regulator. The reference pressure for that regulator is ambient pressure. Therefore the fuel pressure is an addition of the ambient pressure and the nominal feed pressure.

FUNCTION DESCRIPTION:

Initialization:

FUP = C_FUP_NOM + AMP

Recurrency: 100 ms

Formula section:

FUP = C_FUP_NOM + AMP

Calibration data:

Name	Dim	Hex.range	Phys.range	Res	Unit
C_FUP_NOM	1	0...FFFFH	0...5434.2598	0.08291	hPa
Nominal fuel pressure					

4.61 Gear ratio detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
GEAR	V/O	0...FFH	0...255	1	-
Common gear ratio independent from AT or MT					
GR_AT	V/O	0...FFH	0...255	1	-
Automatic transmission actual gear ratio					
GR_MT	V/O	0...FFH	0...255	1	-
Manual transmission gear ratio					
GR_SUB	V/O	0...FFH	0...255	1	-
Current gear ratio for cruise control without consideration of delay time					
LV_N_GRD_POS	V	0...1H	0...1	1	-
Bloolean for indicating the positive sign of the engine speed gradient					
LV_N_GRD_POS_PREV	V	0...1H	0...1	1	-
Bloolean for indicating the positive sign of the engine speed gradient one cycle before					
LV_WIDE_RNG_ACT_GR	V	0...1H	0...1	1	-
Bloolean for indicating wide limits active					
N_VS_RATIO	V	0...FFFFH	0...255.996	1/256	h/(km*min)
Engine speed / vehicle speed ratio					
N_VS_RATIO_FIL	V/O	0...FFFFH	0...255.996	1/256	h/(km*min)
Filtered engine speed / vehicle speed ratio					
T_GR_CHG_LIM	V	0...FFH	0...2.55	0.01	s
Delay timer to confirm a limit change from wide to narrow					
T_NOT_PLAUS_GR	V	0...FFH	0...2.55	0.01	s
Delay timer for resetting the gear ratio of the manual transmission					
T_PLAUS_GR	V	0...FFH	0...2.55	0.01	s
Delay timer for setting the gear ratio of the manual transmission					

Input data:

VS_WORD	N	LV_ES	LV_ST
LV_AT	LV_CS	N_32	LV _ERR_VS
LV_PL	GR_AT_INTER	N_GRD	LV_DT
LC_DT_CS_USE	PV_AV		

General information:

Gear ratio calculation is depending of transmission type (manual or automatic) .

If either manual or semisequential transmission is detected, gear ratio is detected by using engine speed/vehicle speed ratio.

In case of automatic transmission, gear ratio is detected by using the CAN - Information of EGS.

In order to avoid transient torque or transient ignition interventions at gear shifts with manual transmission, the gear variable GR_MT is set to 0 at active clutch switch LV_CS.

The variable GEAR is a common gear ratio independent from AT or MT.

Application conditions:

Initialisation: following initalisations are required at reset or ignition key off to on :

GR_AT = 0
GEAR = 0
GR_MT = 0
GR_MT_PREV = 0
GR_SUB = 0
GR_SUB_PREV = 0

N_VS_RATIO=0

LV_N_GRD_POS = 0
LV_N_GRD_POS_PREV = 0
LV_WIDE_RNG_ACT_GR = 1

Reset delay time T_GR_CHG_LIM = 0
Reset delay time T_PLAUS_GR = 0
Reset delay time T_NOT_PLAUS_GR = 0

Recurrence: 10 ms

Activation: *In all engine operating states except LV_ES and LV_ST .*

Deactivation: *In engine operating states LV_ES and LV_ST.*
action: GR_MT = 0
GR_AT = 0

4.61.1 Configuration

The project has the possibility to use or not the clutch switch information in order to detect the gear ratio. This choice is proposed through the configuration bit : LC_GR_MT_CS_USE

If LC_GR_MT_CS_USE =1 (use of CS information to detect gear ratio)
then Strategy 1 is applied
else Strategy 2 is applied
endif

4.61.2 Manual transmission (LV_AT=0)

Formula section:

Determination of engine speed / vehicle speed ratio

If the gear ratio value is given by the CAN see directly "CAN messages EMS" specification.

The aim of this strategy is to put gr_mt to 0 when clutch is engaged. When vs_word is different from 0, calculation of N_VS_RATIO is made.
In case of failure, the ratio is set at its maximum value.

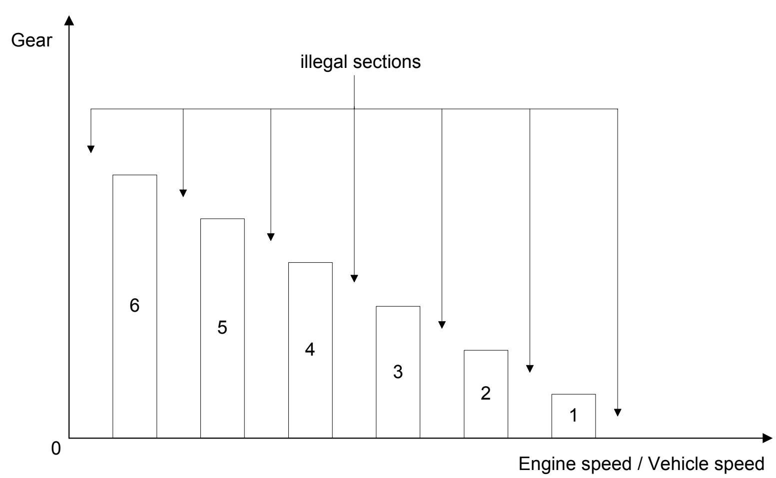
```
    If      (LV_AT = 0)
            If      (LV_ERR_VS = 0 and VS_WORD > 0)
then
                    N_VS_RATIO = N / VS_WORD
            else
                    N_VS_RATIO=FFFFH
                    GR_SUB_PREV = 0
            endif
    endif
```

4.61.2.1 Strategy 1: use of clutch switch information

Determination of gear position (variable GR_MT)

If a Manual Transmission is recognized the determination of the actual gear position is made by comparing the engine speed / vehicle speed ratio to the gear identification limits, during a stabilized duration C_GEAR_DLY_1.

The transition from a gear to gr_mt=0 must be confirmed during C_GEAR_DLY_2



Determination of gear box position (variable GR_MT)

The identification of a gear is made by engine speed to vehicle speed ratio.

```
If      LV_ES = 1 or LV_ST = 1 or LV_AT = 1 or
        N_VS_RATIO = FFFFH or LV_CS = 1
then    GR_MT = 0
else
        GR_SUB = ID_GR_MT (N_VS_RATIO)
        If      GR_SUB ≠ 0
        then    If      GR_SUB_PREV = 0
                then    T_NOT_PLAUS_GR = 0
                        Increment timer T_PLAUS_GR
                        GR_SUB_PREV = GR_SUB
                else    If      T_PLAUS_GR ≥ C_GEAR_DLY_1
                        then    GR_MT = GR_SUB
                                T_PLAUS_GR = 0
                        else    Increment timer T_PLAUS_GR
                        endif
                endif
        else    If      GR_SUB_PREV ≠ 0
                then    T_PLAUS_GR = 0
                        Increment timer T_NOT_PLAUS_GR
                        GR_SUB_PREV = GR_SUB
                else    If      T_NOT_PLAUS_GR ≥ C_GEAR_DLY_2
                        then    GR_MT = 0
                                T_NOT_PLAUS_GR = 0
                        else    Increment timer T_NOT_PLAUS_GR
                        endif
                endif
        endif
endif
```

4.61.3 Strategy 2: clutch switch information isn't used

Determination of gear box position (variable GR_MT)

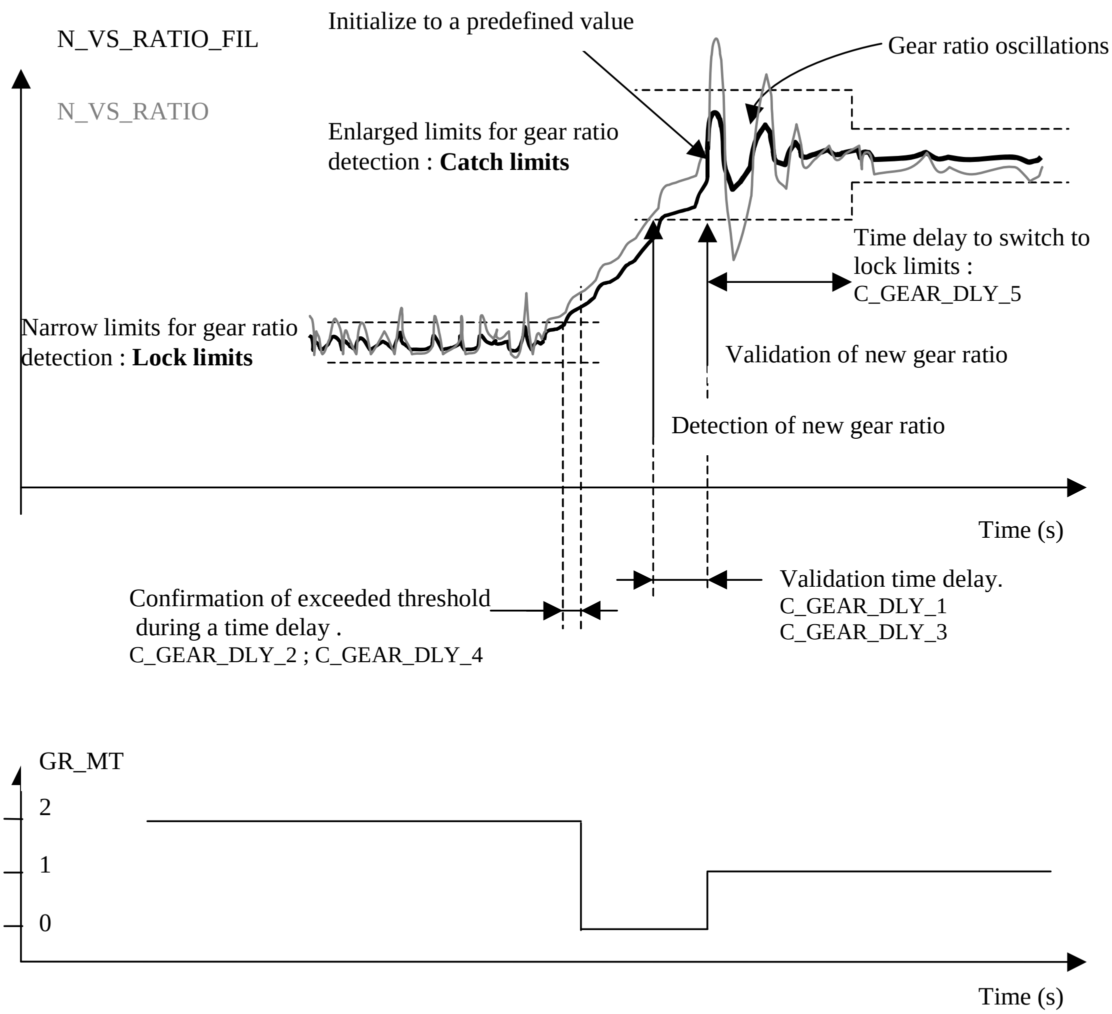
The aim of this strategy is to detect a gear change as fast as possible. It's based on a filtering which depend on present gear and engine speed in order to filter N_VS_RATIO oscillations (see gear ratio filtering). The gear identification is made by comparing the result of filtering (N_VS_RATIO_FIL) to predefined thresholds during a confirmation or validation time. We establish two kinds of limits:

Wide or catch limits: to detect a change from gear 0

Narrow or lock limits: to detect a change to gear 0.

The switch from the wide limits to the narrow limits is done after a time delay (C_GEAR_DLY_5).

Note: To avoid GR_MT oscillations, between gear 0 and the original gear as soon as the vehicle is declutched, we conserve the narrow limits only for the previous gear (see GR_MT_PREV). On the other hand, the change to the other gears is defined by the wide limits.



Determination of gear box position (variable GR_MT)

If a manual transmission is recognized, the identification of a gear is made by comparing the N_VS_RATIO_FIL to the gear identification limits, during a stabilized duration.

```
If          LV_ES = 1 or LV_ST = 1 or LV_AT = 1
then
            GR_MT = 0
            GR_MT_PREV = 5
            GR_SUB_PREV = 0
            N_VS_RATIO_FIL = 0
else        (Gear Ratio Filtering; Limits Determination)
    If      (LV_DT = 0 and LC_DT_CS_USE= 0 and PV_AV ≤ C_PV_AV_GEAR_MIN)
    then    GR_MT = 0
            GR_SUB = ID_GR_MT_RNG_1(N_VS_RATIO_FIL)
    else
            If      LV_WIDE_RNG_ACT_GR = 1 (Wide limit active)
            then    GR_SUB = ID_GR_MT_RNG_1 (N_VS_RATIO_FIL)
                     If     GR_SUB = GR_MT_PREV and GR_MT=0
                    then    (Narrow limit active)
                            GR_SUB = ID_GR_MT_RNG_2 (N_VS_RATIO_FIL)
                    endif
            else    (Narrow limit active)
                    GR_SUB = ID_GR_MT_RNG_2 (N_VS_RATIO_FIL)
            endif

            If      GR_SUB ≠ 0
            then    If      GR_SUB_PREV = 0
                    then    T_NOT_PLAUS_GR = 0
                            Increment timer T_PLAUS_GR
                            GR_SUB_PREV = GR_SUB
                    else    If      T_PLAUS_GR ≥ C_GEAR_DLY_3
                            then    GR_MT = GR_SUB
                                    GR_MT_PREV = GR_MT
                                    T_PLAUS_GR = 0
                            else    Increment timer T_PLAUS_GR
                            endif
                    endif
            else    If      GR_SUB_PREV ≠ 0
                    then    T_PLAUS_GR = 0
                            Increment timer T_NOT_PLAUS_GR
                            GR_SUB_PREV = GR_SUB
                    else    If      T_NOT_PLAUS_GR ≥ C_GEAR_DLY_4
                            then    GR_MT = 0
                                    T_NOT_PLAUS_GR = 0
                            else    Increment timer T_NOT_PLAUS_GR
                            endif
                    endif
            endif
    endif
endif
```

Reset of GR_MT_PREV at stopped vehicle:
```
If      VS_WORD = 0
then    GR_MT_PREV = 0
endif
```

Gear Ratio Filtering :

The aim is to filter engine speed oscillations which can appear during a gear change.

```
if      N_VS_RATIO(n) = FFFFH
then    N_VS_RATIO_FIL(n) = FFFFH
else    N_VS_RATIO_FIL(n) = N_VS_RATIO_FIL(n-1) + ID_CRLC_N_VS_RATIO_FIL *
        (N_VS_RATIO(n) -N_VS_RATIO_FIL(n-1))
endif
```

Limits Determination :

As long as the engine state is part load, we use the wide limits. On the other hand, for the other engine states, as soon as the engine speed gradient is lower than a threshold, we use the narrow limits after a time delay (C_GEAR_DLY_5).

```
If    GR_MT ≠ 0
then  If   LV_PL = 0      (tip in / out protection)
      then
           If    |N_GRD| ≤ C_N_GRD_GEAR_MAX
           then  Increment of T_GR_CHG_LIM
                 If |N_GRD| > C_N_GRD_GEAR_MIN & LV_WIDE_RNG_ACT_GR = 1
                 then
                       If     N_GRD ≥ 0
                       then   LV_N_GRD_POS = 1
                       else   LV_N_GRD_POS = 0
                       endif
                       If     LV_N_GRD_POS ≠ LV_N_GRD_POS_PREV
                       then   T_GR_CHG_LIM = 0
                       endif
                       LV_N_GRD_POS_PREV = LV_N_GRD_POS
                 endif
           else  T_GR_CHG_LIM = 0
           endif

           If    T_GR_CHG_LIM ≥ C_GEAR_DLY_5
           then  Stop increment of T_GR_CHG_LIM
                 Activation of narrow limits
                 LV_WIDE_RNG_ACT_GR = 0
           else  Activation of wide limits
                 LV_WIDE_RNG_ACT_GR = 1
           endif
      else       Activation of wide limits
                 LV_WIDE_RNG_ACT_GR = 1
                 T_GR_CHG_LIM = 0
      endif
else  Activation of wide limits
      LV_WIDE_RNG_ACT_GR = 1
T_GR_CHG_LIM = 0
```

endif

4.61.4 Automatic transmission

The GR_AT gear ratio is calculated via CAN information coming from ETCU.

If LV_AT = 0
then
GR_AT = 0
else
GR_AT = GR_AT_INTER
endif

4.61.5 GEAR -Coordination

If LV_AT =1
then
GEAR = GR_AT
else
GEAR = GR_MT
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_GR_MT	14	0...FFH	0...255	1	-
LDP_N_VS_RATIO_ ID_GR_MT	14	0...FFFFH	0...255.996	1/256	-
Index table for Gear Ratio detection (LV_VAR_TOT_RATIO = 0)					
ID_GR_MT_RNG_1	14	0...FFH	0...255	1	-
LDP_N_VS_RATIO_FIL_ ID_MT_RNG_1	14	0...FFFFH	0...255.996	1/256	-
Index table for Gear Ratio detection with wide limits (LV_VAR_TOT_RATIO = 0)					
ID_GR_MT_RNG_2	14	0...FFH	0...255	1	-
LDP_N_VS_RATIO_FIL_ ID_MT_RNG_2	14	0...FFFFH	0...255.996	1/256	-
Index table for Gear Ratio detection with narrow limits (LV_VAR_TOT_RATIO = 0)					
C_GEAR_DLY_1	1	0...FFH	0...2.55	0.01	s
Time needed for gear ratio detection (with Clutch configuration)					
C_GEAR_DLY_2	1	0...FFH	0...2.55	0.01	s
Time needed for GR_MT=0 detection (with Clutch configuration)					
C_GEAR_DLY_3	1	0...FFH	0...2.55	0.01	s
Time needed for gear ratio detection (without Clutch configuration)					
C_GEAR_DLY_4	1	0...FFH	0...2.55	0.01	s
Time needed for GR_MT=0 detection (without Clutch configuration)					
C_GEAR_DLY_5	1	0...FFH	0...2.55	0.01	s
Delay time to switch from wide limits to narrow limits					
ID_CRLC_N_VS_RATIO_FIL	6*8	0...FFFFH	0...0.999985	1.5259 e-5	-
LDP_N_32_ ID_CRLC_N_VS_RATIO	6	0...FFH	0...8160	32	rpm
LDP_ GR_MT_ ID_CRLC_N_VS_RATIO	8	0...FFH	0...255	1	-
N_VS_RATIO filter value depending on N_32 and gr_mt					
C_N_GRD_GEAR_MAX	1	0...7FH	0...4064	32	rpm/s
Maximum engine speed gradient to change gear limits					
C_N_GRD_GEAR_MIN	1	0...7FH	0...4064	32	rpm/s
Minimum engine speed gradient to change gear limits					
C_PV_AV_GEAR_MIN	1	0...FFH	0...99.6	0.39	%
Pedal value below which GR_MT is forced to zero when LV_DT = 0.					
LC_GR_MT_CS_USE	1	0...1H	0...1	1	-
Configuration flag for using or not LV_CS information					

4.62 Drive Train Engaged

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DT	V/O	0...1	0...1	1	-
Boolean for “closed DRIVE TRAIN”					
N_GEAR_SHIFT_BEG	V	0...1FE0H	0...8160	1	rpm
Engine speed at the begin of STATE_ENG = PU for the detection of overspeed.					
LV_N_RISE_GEAR_SHIFT	V	0...1H	0...1	1	-
Logical variable for the detection of the engine speed overshoot at gear shifts.					
T_N_RISE_GEAR_SHIFT	V	0...FFH	0...2.55	0.01	s
Timer for the detection of an open drive train during gear shifts.					
T_DT_TRANS	V	0...FFH	0...2.55	0.01	s
Timer for the detection of a closed drive train.					
T_DT_IS_SHIFT	V	0...FFFFH	0...655.35	0.01	s
Timer for the detection of a closed drive train in idle speed.					
LV_N_GEAR_SHIFT_BEG_AUTH	V	0...1H	0...1	1	-
Authirization for storing N_GEAR_SHIFT_BEG.					
T_PV_AV_GR_SUB	V	0...FFFFH	0...655.35	0.01	s
Timer for the detection of a closed drive train, if GR_SUB is <> 0					

Input data:

LV_CS	LV_AT	LV_DRI	VS
GR_MT	N_GRD	LV_ES	PV_AV
LV_PUC	N_DIF_REF_LPF_AJ	LV_IS	GR_SUB
LV_CLU_SWI	N	TQI_REQ_DRIV	N_VS_RATIO_FIL
MAF	PV_CRU	LV_PU	

General information:

Drive train engaged active (LV_DT=1) means drive train is closed, engine torque is transmitted to the wheels and the vehicle is running (VS >= C_VS_MIN_DT). For the anti - jerk function the DT detection is expanded. The drive train is engaged after Clutch switch is detected and a positive engine speed gradient is detected.

The dependabillity to the vehicle-speed has different reason for AT and MT vehicle.

AT:

With engaged gear it´s normally not necessary to have an agressive idle-speed control. The only exception is a sudden breaking at slowly moving vehicle with engaged gear. In this case it´s necessary to switch to the strong P_D - parameter map of the idle-speed control to avoid idle-speed break-down.

MT:

At start of the vehicle without pushing the accelerator pedal it´s necessary to switch as fast as possible to the smooth P_D idle-speed control map to avoid vehicle jerking. Depending on slip of the clutch it may last too long until a gear-position determination is possible. Therefore in this case the switching from agressive to the smooth P_D map can happen even at slow vehicle speed.

Nevertheless the actualisation of LV_DT is strongly depending on the update-rate of the input signal ‘VS’.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: at reset, LV_DT = 0; T_DT_TRANS = 0; T_DT_IS_SHIFT = 0

Update rate: 10 ms

4.62.1 Configuration

The project has the possibility to use or not the clutch switch information in order to detect the drive train. This choice is proposed through the configuration bit : LC_DT_CS_USE

If LC_DT_CS_USE =1 (use of CS information to detect drive train engaged)
then Strategy 1 is applied
else Strategy 2 is applied
endif

4.62.2 Manual transmission

Strategy 1 : use clutch switch information

Formula section:

```
If    LV_AT = 0
Then
      If      LV_ES = 0
      Then
              If      LV_CS change from 0 to 1 or
              (GR_MT = 0  and   VS <  C_VS_MIN_DT)
              Then  LV_DT = 0
              Else
                      If      LV_CS = 0 and N_GRD > C_N_GRD_MIN_DT
                      Then  LV_DT = 1
                      Endif
              Endif
      Else    LV_DT=0
      Endif
Endif
```

Strategy 2: the clutch switch information isn't reliable or available:

Formula section:

Detection of engine overshoot at gear shifts:

```
If    PV_AV                     = 0                          and
      VS                        > C_VS_N_RISE_GEAR_SHIFT     and
      N                         > C_N_RISE_GEAR_SHIFT        and
      LV_DT                     = 1
then
      if    LV_N_GEAR_SHIFT_BEG_AUTH = 1
      then
      N_GEAR_SHIFT_BEG = N
      T_N_RISE_GEAR_SHIFT  =  C_T_N_RISE_GEAR_SHIFT
      LV_N_GEAR_SHIFT_BEG_AUTH = 0
      endif
      if          N > N_GEAR_SHIFT_BEG + IP_N_GEAR_SHIFT_OFS        and
                  T_N_RISE_GEAR_SHIFT  > 0
      then        gear shifting is detected by the engine speed overshoot
                  LV_N_RISE_GEAR_SHIFT = 1
                  LV_N_GEAR_SHIFT_BEG_AUTH = 1
      else
                  LV_N_RISE_GEAR_SHIFT = 0
      endif
      decrement T_N_RISE_GEAR_SHIFT
else
      LV_ N_RISE_GEAR_SHIFT           = 0
      LV_N_GEAR_SHIFT_BEG_AUTH        = 1
endif
```

Detection of an engaged drivetrain:

$N\_DIF\_REF\_LPF\_AJ\_GRD = N\_DIF\_REF\_LPF\_AJ_{(N)} – N\_DIF\_REF\_LPF\_AJ_{(N-1)}$

If(1) LV_ES = 0
then(1)
If(2) LV_AT = 0
then(2)
If(3) VS>= C_VS_MIN_DT
then(3)

if(4) LV_IS = 1 **and** LC_CLU_SWI_HW = 1
then(4)
if(5) LV_CLU_SWI = 1

then(5) LV_DT = 0
else(5) LV_DT = 1, T_DT_IS_SHIFT = 0
endif(5)
else(4)
if(6) LV_DT = 1
then(6) (detection of disengaging the clutch)

if(7) {GR_MT_N <> GR_MT_{N-1} **and** GR_MT_{N-1} > 0
and (N > C_N_32_GR_MAX **or**
TQI_REQ_DRIV < C_TQI_REQ_DRIV_GR_MIN)**}**

or

{PV_AV < C_PV_AV_DT_CLOSE_MAX **and**
|N_DIF_REF_LPF_AJ| > IP_N_DIF_REF_LPF_AJ_MIN **}**

or

LV_N_RISE_GEAR_SHIFT = 1

then(7) LV_DT = 0
endif(7)
else(6)

if(8) LV_PUC = 1 **or** (LV_PU = 1 **and** LC_DT_PU_ENA = 1)
then(8) (reengage the clutch in PUC or PU condition)
if(9) N_GRD > C_N_GRD_MIN_DT_PUC
then(9)
reset T_DT_TRANS if
(transition from LV_PUC 0->1 **and** old LV_PU = 0 **and** LC_DT_PU_ENA = 1)
or (transition from LV_PU 0->1 **and** LC_DT_PU_ENA = 1 **and** old LV_PUC = 0)
or (transition from LV_PUC 0->1 **and** LC_DT_PUC_ENA = 0)
if(10) T_DT_TRANS >= C_T_DT_TRANS_PUC

then(10) LV_DT = 1, T_DT_IS_SHIFT = 0, T_DT_TRANS = 0,
T_PV_AV_GR_SUB = 0
else(10)
if(18) GR_SUB = 0

then(18) T_DT_TRANS = 0
else(18) increment T_DT_TRANS
endif(18)
endif(10)
else(9) T_DT_TRANS = 0
endif(9)
else(8) (standard engaging of the clutch in part load)

(standard) **if(11)** (PV_AV > C_PV_AV_DT_OPEN_MIN
or PV_CRU > C_PV_AV_DT_OPEN_MIN) **and**
N_VS_RATIO_FIL < C_N_VS_RATIO_FIL_MIN **and**
|N_DIF_REF_LPF_AJ| < IP_N_DIF_REF_LPF_AJ_MAX
then(11)
reset T_DT_TRANS if
(transition from LV_PUC 1->0 **and** new LV_PU=0 **and** LC_DT_PU_ENA = 1)
or (transition from LV_PU 1->0 **and** LC_DT_PU_ENA = 1 **and** new LV_PUC = 0)
or (transition from LV_PUC 0->1 **and** LC_DT_PUC_ENA = 0)
if(12) T_DT_TRANS >= C_T_DT_TRANS_MAX

then(12) LV_DT = 1, T_DT_IS_SHIFT = 0, T_DT_TRANS = 0,
T_PV_AV_GR_SUB = 0
else(12) increment T_DT_TRANS
endif(12)
else(11) T_DT_TRANS = 0
if(13) PV_AV > C_PV_AV_DT_OPEN_2_MIN
and

(fast shifting up) **{**[N_DIF_REF_LPF_AJ>IP_N_DIF_REF_LPF_AJ_MAX
and N_DIF_REF_LPF_AJ_GRD <
C_N_DIF_REF_LPF_AJ_GRD_MAX]

or

(fast shifting down) [N_DIF_REF_LPF_AJ < -IP_N_DIF_REF_LPF_AJ_MAX
and N_DIF_REF_LPF_AJ_GRD >
C_N_DIF_REF_LPF_AJ_GRD_MIN]**}**

then(13) LV_DT = 1, T_DT_IS_SHIFT = 0,
T_PV_AV_GR_SUB = 0
endif(13)

(shifting in idle) **if(14)** (GR_SUB ≠ 0 **and** LV_IS = 1)
then(14)
if(15) T_DT_IS_SHIFT > C_T_DT_IS_SHIFT

then(15) LV_DT = 1, T_DT_IS_SHIFT = 0,
T_PV_AV_GR_SUB = 0
else(15) increment T_DT_IS_SHIFT
endif(15)
else(14) T_DT_IS_SHIFT = 0
endif(14)
endif(11)

(backup condition, **if(16)** (PV_AV > C_PV_AV_GR_SUB
or PV_CRU > C_PV_AV_GR_SUB)
and GR_SUB ≠ 0

```
if GR_SUB is stable)    then(16)
                           if(17)       T_PV_AV_GR_SUB > C_T_PV_AV_GR_SUB
                           then(17)     LV_DT=1, T_PV_AV_GR_SUB = 0,
                                        T_DT_IS_SHIFT = 0, T_DT_TRANS = 0
                           else(17)     increment T_PV_AV_GR_SUB
                           endif(17)
                        else(16)        T_PV_AV_GR_SUB = 0
                        endif(16)
                     endif(8)
                 endif(6)
               endif(4)
      else(3)         LV_DT = 0
      endif(3)
   endif(2)
else(1) LV_DT = 0
endif(1)
```

4.62.3 Automatic transmission:

Formula section:

```
If       LV_ES = 0
Then     If     LV_AT = 1
         Then   If     (LV_DRI = 1   and   VS >=  C_VS_MIN_DT)
                Then   LV_DT = 1
                Else   LV_DT = 0
                Endif
         Endif
Else     LV_DT = 0
Endif
```

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_DT_CS_USE	1	0...1H	0...1	1	-
Configuration whether the clutch swich shall be used for DT detection.					
LC_CLU_SWI_HW	1	0...1H	0...1	1	-
Configuration whether a hardware clutch swich shall be used for DT detection.					
LC_DT_PU_ENA	1	0...1H	0...1	1	-
Configuration to allow LV_DT detection in PU					

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_N_DIF_REF_LPF_AJ_MAX	1*6	0... 7FFFH	0..511.98	0.0156	rpm
LDPM_VS_ IP_N_DIF_REF_LPF_AJ	6	0...FFH	0...255	1	Km/h
Maximum permitted deviation between reference N (AJ) and N to detect changes in DT (0->1)					
IP_N_DIF_REF_LPF_AJ_MIN	6*4	0... 7FFFH	0..511.98	0.0156	rpm
LDP_GEAR_IP_N_DIF_LPF_AJ_MIN	6	0...FFH	0...255	1	-
LDP_TOIL_IP_N_GEAR_SHIFT_OFS	4	0...FFH	-40...215	1	°C
Minimum permitted deviation between reference N (AJ) and N to detect changes in DT (1->0)					
IP_N_GEAR_SHIFT_OFS	4*6	0...1FE0H	0...8160	1	rpm
LDP_MAF_IP_N_GEAR_SHIFT_OFS	4	0...FFFFH	0...1389	0.0211	mg/str
LDP_GEAR_IP_N_DIF_LPF_AJ_MIN	6	0...FFH	0...255	1	-
Engine speed offset for the detection of overspeed during gear changes.					
C_VS_MIN_DT	1	0 ... FFH	0 ... 255	1	km/h
Threshold for vehicle running and drive-train closed detection					
C_N_GRD_MIN_DT	1	80 ... 7FH	-4096 ... 4064	32	rpm/s
Threshold for engine speed gradient for LV_DT detection					
C_VS_N_RISE_GEAR_SHIFT	1	0...FFH	0...255	1	Km/h
Minimum vehicle speed to detect an engine speed overshoot at gear shifts.					
C_N_RISE_GEAR_SHIFT	1	0...1FE0H	0...8160	1	rpm
Min. engine speed for the detection of overspeed during gear changes.					
C_N_32_GR_MAX	1	0 ... FFH	0 ... 8160	32	rpm
N threshold to inhibit the detection of LV_DT = 0 due to jerks in the power train					
C_TQI_REQ_DRIV_GR_MIN	1	0 ...7FFFH	0 ...1023,97	0.03125	Nm
PV_AV threshold to inhibit the detection of LV_DT = 0 due to jerks in the power train					
C_PV_AV_DT_CLOSE_MAX	1	0 ... FFH	0 ... 99.6	0.39	%
Maximum permitted PV_AV to detect open DT					
C_N_GRD_MIN_DT_PUC	1	80 ... 7FH	-4096 ... 4064	32	rpm/s
Minimum engine speed gradient to detect closed drive train in PUC					
C_PV_AV_DT_OPEN_MIN	1	0 ... FFH	0 ... 99.6	0.39	%
Minimum permitted PV_AV to detect close DT					
C_N_VS_RATIO_FIL_MIN	1	0...FFFFH	0...255.996	1/256	h/(km*min)
N_VS_RATIO_FIL threshold to detect LV_DT = 1 correct at starting the vehicle					
C_T_DT_TRANS_MAX	1	0 ... FFH	0 ... 2.55	0.01	s
Timer which runs after fulfilling all conditions before DT = 1 is allowed					
C_T_DT_TRANS_PUC	1	0 ... FFH	0 ... 2.55	0.01	s
Timer which runs after fulfilling all conditions before DT = 1 is allowed in PUC.					
C_PV_AV_DT_OPEN_2_MIN	1	0 ... FFH	0 ... 99.6	0.39	%
Minimum permitted PV_AV to detect close DT (second way)					
C_N_DIF_REF_LPF_AJ_GRD_MAX	1	8000...7FFFH	-512..511.98	0.0156	rpm/10ms
Maximum permitted gradient deviation between reference N (AJ) and N to detect changes in DT (0->1)					
C_N_DIF_REF_LPF_AJ_GRD_MIN	1	8000...7FFFH	-512..511.98	0.0156	rpm/10ms
Minimum permitted gradient deviation between reference N (AJ) and N to detect changes in DT (0->1)					
C_T_DT_IS_SHIFT	1	0...FFFFH	0...655.35	0.01	s
Timer for the detection of a closed drive train in idle speed.					
C_PV_AV_GR_SUB	1	0 ... FFH	0 ... 99.6	0.39	%
Minimum permitted pedal value for drive train detection when GR_SUB is already detected.					
C_T_PV_AV_GR_SUB	1	0...FFFFH	0...655.35	0.01	s
Timer threshold for the detection of a closed drive train when GR_SUB is already stable.					
C_T_N_RISE_GEAR_SHIFT	1	0 ... FFH	0 ... 2.55	0.01	s
Timer threshold for the detection of an open drive train during gear shifts.					

4.63 Oxygen Sensor Signal Processing Upstream

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_UP[NC_CBK_EX_NR]	O/V	0...3FFH	0...4.99512	0.0048828	V
Upstream Oxygen sensor voltage					
LV_VLS_UP_VLD[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Boolean flag indicating whether clipping of the WRAF sensor signal is taking place (1 - No clipping, 0 - clipping)					
VLS_UP_10_RAW[NC_CBK_EX_NR]	O/V	0...3FFH	0...4.99512	0.0048828	V
Raw value of WRAF sensor voltage signal - Recurrence 10.0 msec					
LV_VLS_UP_INIT[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Boolean flag indicating the buffer initialisation after switching between Gain 16 & Gain 8 values of WRAF Controller or after Offset adjustment of WRAF sensor					

Input data:

VLS_UP_RAW[NC_CBK_EX_NR]	NC_CBK_EX_NR		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VLS_UP_THD_BOL	1	0...3FFH	0...4.99512	0.0048828	V
Lower voltage threshold for WRAF sensor signal					
C_VLS_UP_THD_TOL	1	0...3FFH	0...4.99512	0.0048828	V
Upper voltage threshold for WRAF sensor signal					

4.63.1 EGCP_SIGCVLSL3

The raw output voltage representing lambda which is provided by BSW is processed within this module and mapped with the system variable VLS_UP[i].

The signal acquisition shall take place at BSW level and the value VLS_UP_RAW[i] will be updated every 10 ms. The output of this module VLS_UP[i] is same as the raw value delivered by BSW.

The signal is monitored for possible clipping due to saturation effects with signal levels close to the range limits of the output amplifier.

The bit LV_VLS_UP_INIT[i] is not applicable here and hence initialized to zero.

The calculation shall be done for all exhaust cylinder banks.

NC_CBK_EX_NR defines the number of exhaust banks. For vector elements, the variable extension „_i“ is used in the model instead of „[i]“ as found in the textual description.

Function Description

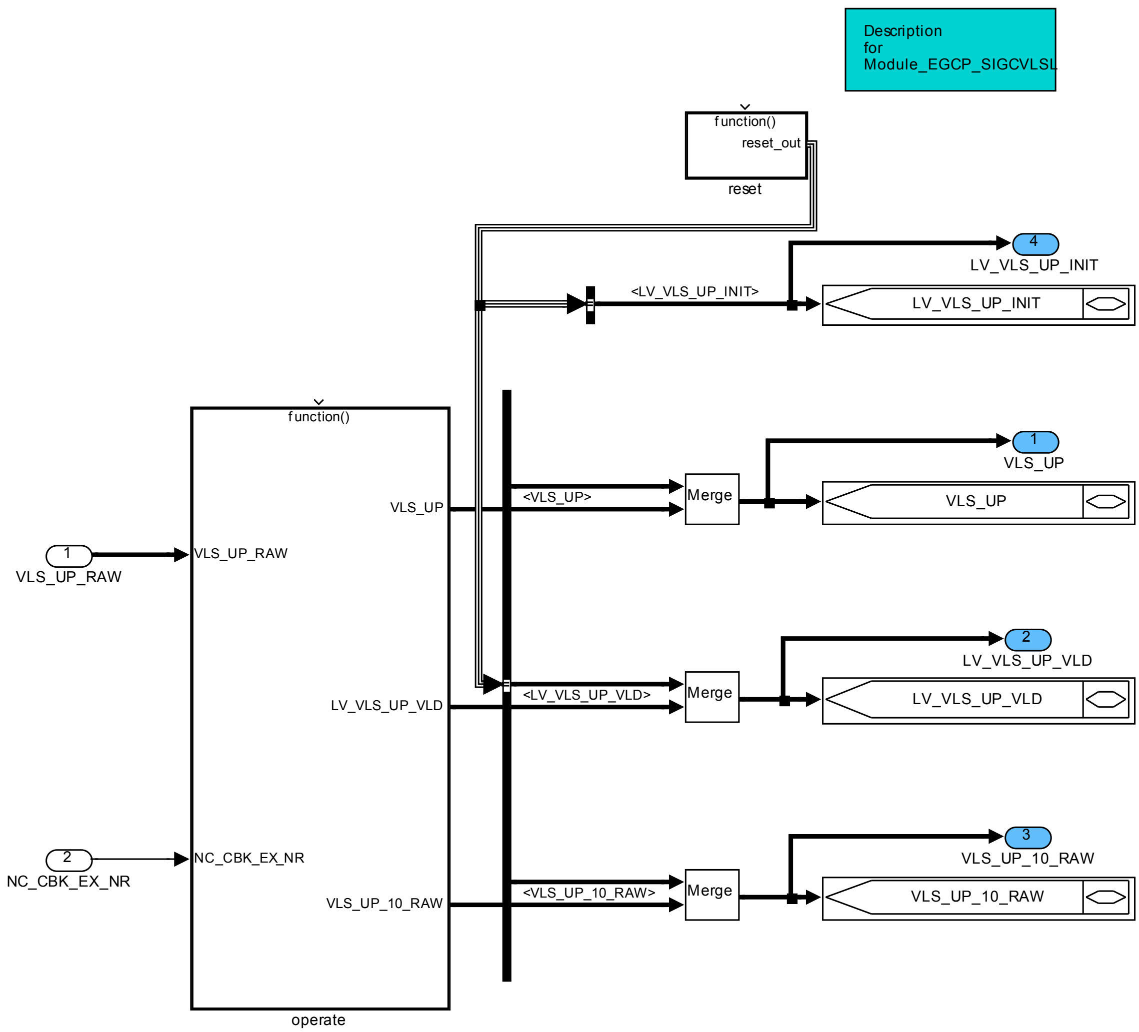


Figure 394 EGCP_SIGCVLSL3

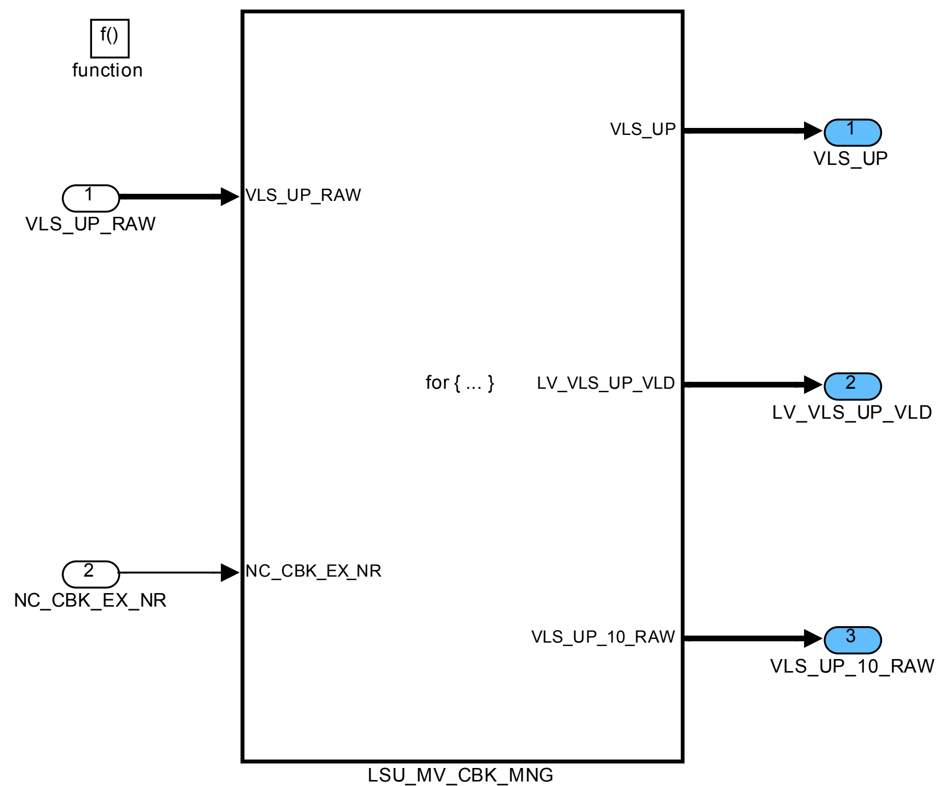


Figure 395 EGCP_SIGCVLSL3/ operate

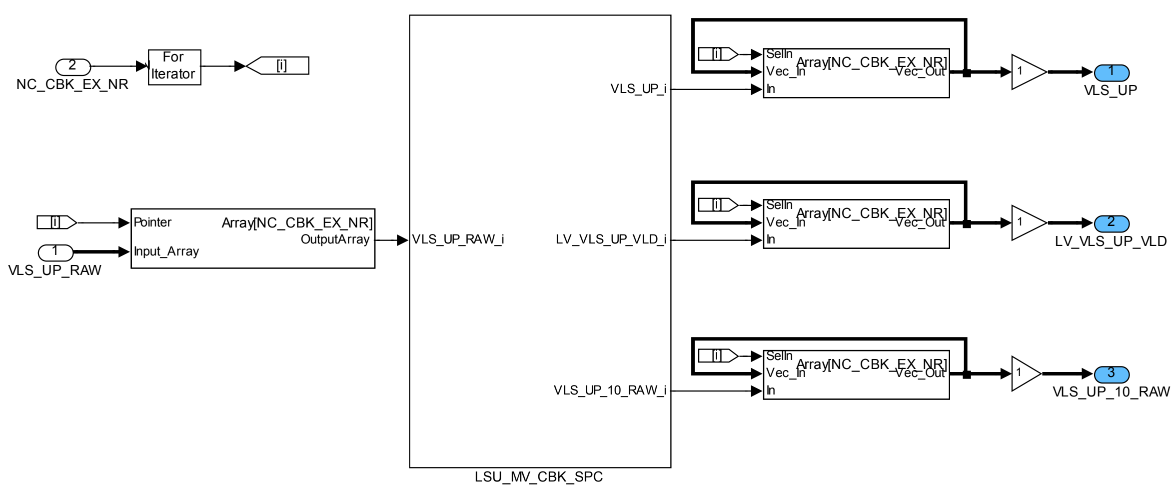


Figure 396 EGCP_SIGCVLSL3/ operate/ LSU_MV_CBK_MNG

Detection of voltage clipping

The raw signal voltage is always compared with limit values and checked for clipping of signal if any. Accordingly the flag LV_VLS_UP_VLD[i] will be set.

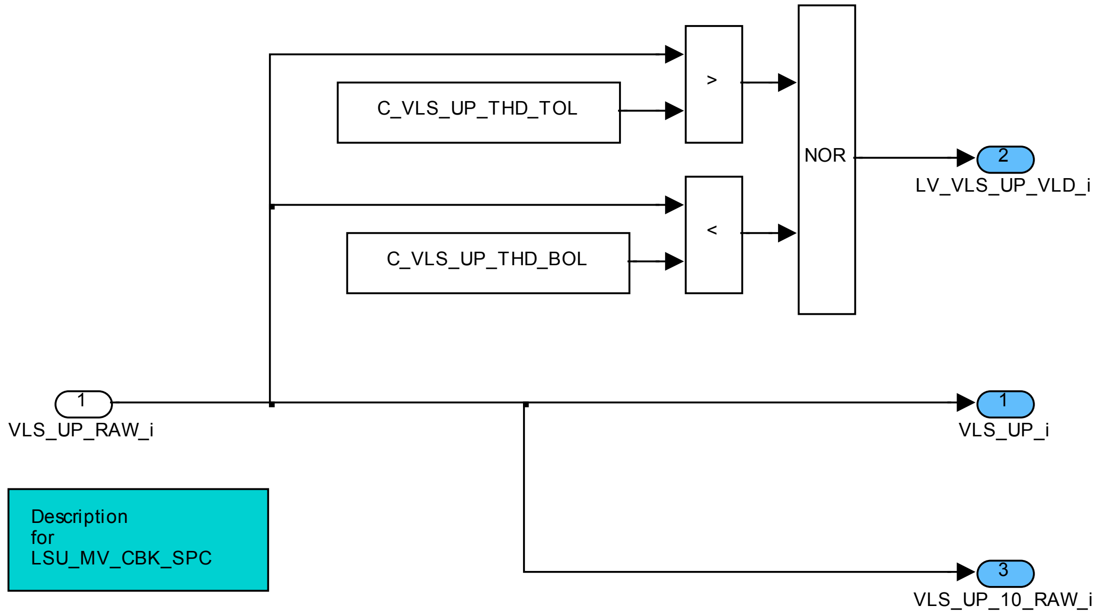


Figure 397 EGCP_SIGCVLSL3/ operate/ LSU_MV_CBK_MNG/ LSU_MV_CBK_SPC

4.64 Downstream oxygen sensor internal resistance determination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
R_IT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	Ω
Internal resistance of downstream oxygen sensor (Released)					
R_IT_MES_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω
Internal resistance of downstream oxygen sensor (Measured)					
R_IT_MDL_LS_DOWN_NEW[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	Ω
Internal resistance of downstream oxygen sensor (Modelled for new sensor)					
POW_MMV_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...84.498711	1.2837E-3	W
Moving mean value of downstream oxygen sensor heater power					
TTIP_REF_MDL_MMV_LS_DOWN[NC_CBK_EX_NR]	V	0...7FFFH	0...2047.938	62.5e-3	°C
Reference of downstream oxygen sensor ceramic temperature (Modelled for new sensor)					
VLS_DOWN_CUR_PUMP_ON[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828e-3	V
Downstream oxygen sensor voltage C_T_CUR_PUMP_ON_LS_DOWN after switching on pump current switch					
VLS_DOWN_CUR_PUMP_OFF[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828e-3	V
Downstream oxygen sensor voltage prior to switching on pump current switch					
CTR_CYCNR_R_IT_LS_DOWN_VLD[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	-
Handshake counter indicating the number of valid resistance values determined since initial activation					
LV_VLS_DOWN_DRV1_CDN_R_LS[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that low gradient condition met					
LV_VLS_DOWN_RNG_CDN_R_LS[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that VLS_DOWN at low gradient condition met					
LV_R_IT_REQ_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that conditions have been met and resistance determination requested					
LV_R_IT_DET_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that resistance has already been determined (lock out)					
VLS_DOWN_MMV_DRV1_THD_MIN[NC_CBK_EX_NR]	V	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Lower variable VLS_DOWN gradient threshold					
VLS_DOWN_MMV_DRV1_THD_MAX[NC_CBK_EX_NR]	V	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Upper variable VLS_DOWN gradient threshold					
T_AFL_CYC_HLD_R_IT_LS_DOWN[NC_CBK_EX_NR]	V	0...FFH	0...2.55	10e-3	s
Lean mixture cycle time (measured upstream), held until next threshold crossing downstream					
FAC_T_VLS_DOWN_MMV_DRV1[NC_CBK_EX_NR]	V	0...FFH	0...63.75	0.25	-
Multiplicative factor modifies DRV1_VLS_DOWN_MMV_MIN_THD dependent on T_VLS_CYC_AFL_HLD					
FAC_T_VLS_DOWN_MMV_DRV1_ABS[NC_CBK_EX_NR]	V	0...FFH	0...15.9375	62.5e-3	-
Multiplicative factor modifies C_VLS_DOWN_MMV_DRV1_ABS_MAX dependent on T_VLS_CYC_AFL_HLD					
TTIP_MES_LS_DOWN[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Downstream oxygen sensor ceramic temperature					
TTIP_MES_TOL_LS_DOWN[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C

Downstream oxygen sensor ceramic temperature, Upper limit, Unfiltered					
TTIP_MES_BOL_LS_DOWN[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Downstream oxygen sensor ceramic temperature, Lower limit, Unfiltered					
TTIP_MES_TOL_MMV_LS_DOWN[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Downstream oxygen sensor ceramic temperature, Upper limit, Filtered					
TTIP_MES_BOL_MMV_LS_DOWN[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Downstream oxygen sensor ceramic temperature, Lower limit, Filtered					
T_SAMPLE_R_IT_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	10e-3	s
Time measured between internal resistance samples					

Input data:

LV_ST_END	VLS_DOWN[NC_CBK_EX_NR]	VLS_DOWN_DRV1_MMV[NC_CBK_EX_NR]	LV_INH_R_IT_LS_DOWN[NC_CBK_EX_NR]
LV_ES	VLS_DOWN_MMV_MIN[NC_CBK_EX_NR]	VLS_DOWN_DRV1_MMV_MIN[NC_CBK_EX_NR]	LV_VLS_DOWN_MMV_LIM[NC_CBK_EX_NR]
TEMP_CAT	VLS_DOWN_MMV_MAX[NC_CBK_EX_NR]	VLS_DOWN_DRV1_ABS_MAX[NC_CBK_EX_NR]	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]
MAF_KGH	VLS_DOWN_BOL[NC_CBK_EX_NR]	T_AFL_CYC_HLD[NC_CBK_EX_NR]	STATE_LSH_DOWN[NC_CBK_EX_NR]
V_EFC_LSH_DOWN[NC_CBK_EX_NR]	VLS_DOWN_TOL[NC_CBK_EX_NR]	CTR_AFL_CYC[NC_CBK_EX_NR]	LV_VLS_DOWN_MMV_ACT
TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]	NC_CBK_EX_NR		

FUNCTION DESCRIPTION:

General information:

The function shall provide the output variable R_IT_LS_DOWN[i], the internal resistance of the oxygen sensor. The resistance is dependent on the operating temperature of the sensor and may be used for oxygen sensor diagnosis.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank.

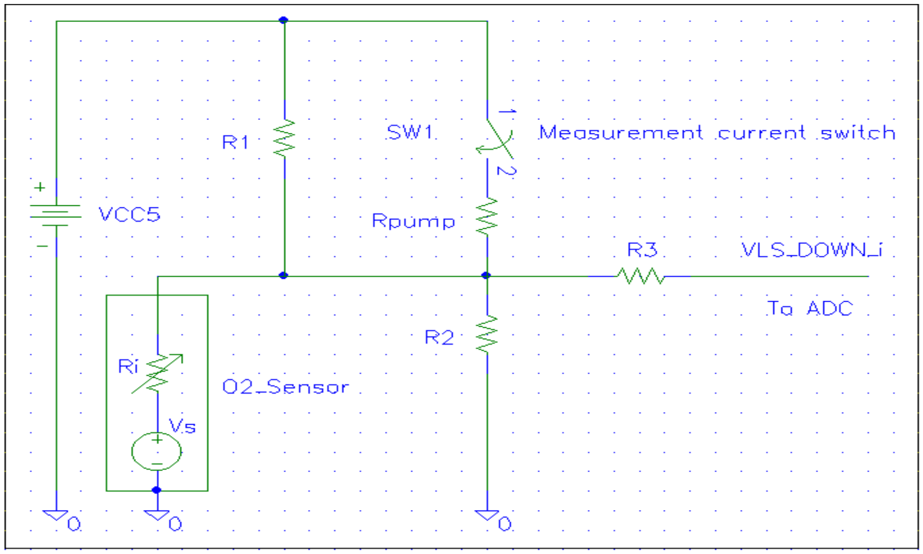


Figure 1 Simplified measurement circuit schematic, Internal resistance determination

Signal flow diagram:

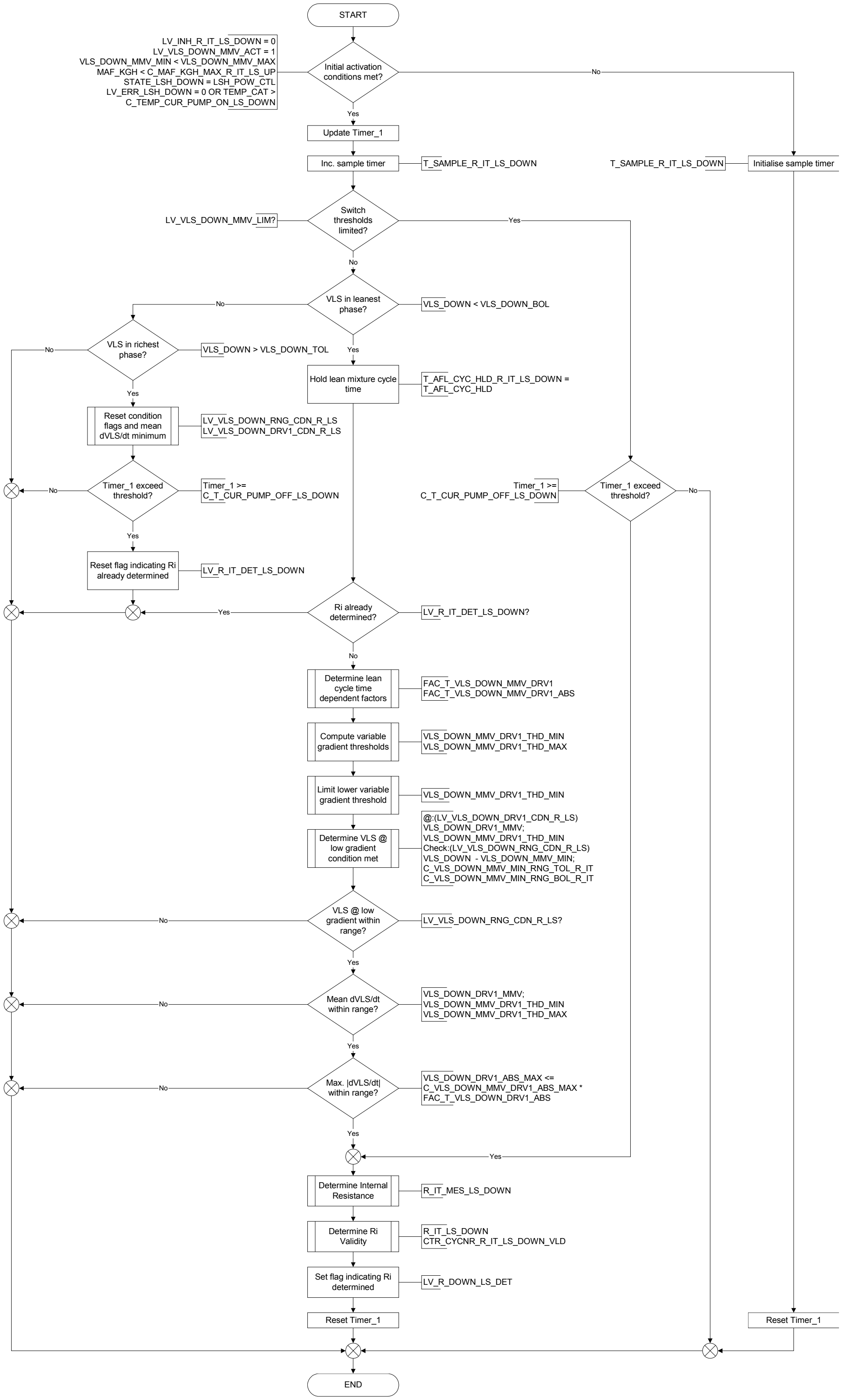


Figure 2 Function flow chart, Internal resistance determination

Description:

The internal resistance of the oxygen sensor shall be determined by use of the oxygen sensor interface circuit *Figure 1* (Capacitors not shown). The internal resistance provides information as to the tip temperature of the sensor and may be used for diagnosis purposes and operative readiness determination.

Under normal conditions, the switch SW1 shall be open and the oxygen sensor signal voltage (VLS_DOWN[i]) is measured by the microcontroller.

The potential divider (resistors R_1 & R_2) shall be permit the operation readiness of the oxygen sensor to be determined and enable electrical fault conditions to be detected.

Periodically, for a limited amount of time, the switch SW1 shall be closed and a voltage of approx. 4.8 V connected through resistor R_{pump} to the oxygen sensor signal connection. This shall cause current to flow into the sensor and through the internal resistance R_i, thereby “pumping” the sensor. The resultant voltage acquired at the microcontroller input shall permit the internal resistance of the sensor to be calculated by use of a formula shown in the formula section below.

Prior to permitting the determination of the internal resistance, a number of conditions shall be determined to be met:

The initial activation conditions shall require that: No function inhibit is requested (LV_INH_R_IT_LS_DOWN[i] = 0); Delay since deactivation since last overrun fuel cut-off passed (LV_VLS_DOWN_MMV_ACT); The initial post initialisation transients have passed for the VLS_DOWN_MMV_xxx[i] inputs (VLS_DOWN_MMV_MIN[i] < VLS_DOWN_MMV_MAX[i]); The engine load threshold has not been exceeded (MAF_KGH < C_MAF_KGH_MAX_R_IT_LS_DOWN); The oxygen sensor heater state LSH_POW_CTL has been reached; No heater OBDI error has been detected in the relevant heater driver (LV_ERR_LSH_DOWN[i]) or should a heater driver fault have been detected, the modelled catalyst temperature TEMP_CAT shall exceed the threshold C_TEMP_CUR_PUMP_ON_LS_DOWN

Once the above conditions have been met, an internal timer (*TIMER_1*) shall be started and timer T_SAMPLE_R_IT_LS_DOWN[i] shall be updated. The latter timer shall be initialised with its maximum value of 655.35 s. This ensures that the temperature limits for the validity function are correctly initialised. The further procedure shall depend on the status of the oxygen sensor signal, as determined by LV_VLS_DOWN_MMV_LIM. Should the VLS_DOWN_MMV_xxx[i] signals be limited, i.e. no or little oxygen sensor amplitude is present, the function shall determine the internal resistance every (C_T_CUR_PUMP_OFF_LS_DOWN + C_T_CUR_PUMP_ON_LS_DOWN + C_T_DLY_CUR_PUMP_LS_DOWN). Otherwise the oxygen sensor signal shall be determined to have significant amplitude and the internal resistance determination shall only be carried out in lean phases where the oxygen sensor signal is relatively flat and stable. This is achieved by the following conditions:

If the oxygen sensor signal is lean (VLS_DOWN[i] < VLS_DOWN_BOL[i]) and neither the resistance determination has been carried out yet in this lean phase nor the determination be locked out due to the minimum off time (LV_R_IT_DET_LS_DOWN[i] = 0) then the function shall determine the lean cycle time dependent factors FAC_T_VLS_DOWN_MMV_DRV1[i] & FAC_T_VLS_DOWN_MMV_DRV1_ABS[i] from the appropriate maps IP_FAC_T_DRV1_VLS_DOWN_MMV & IP_FAC_T_VLS_DOWN_MMV_DRV1_ABS respectively. These factors shall be dependent on the held lean cycle time from the previous lean cycle (T_AFL_CYC_HLD_R_IT_LS_DOWN[i]). Should the cycle time decrease then it is intended that the factors shall increase and vice versa.

The thresholds for the gradient condition shall then be computed. These thresholds (VLS_DOWN_MMV_DRV1_THD_MAX[i] & VLS_DOWN_MMV_DRV1_THD_MIN[i]) are directly proportional to the amplitude of the difference between VLS_DOWN_MMV_xxx[i] signals. Furthermore, VLS_DOWN_MMV_DRV1_THD_MIN[i] shall be modified by factor FAC_T_VLS_DOWN_MMV_DRV1[i]. The factor FAC_T_VLS_DOWN_MMV_DRV1[i] shall be set to 1 unless C_CTR_AFL_CYC_MIN_RI_IT_LS_DOWN AF cycles have passed since the last lambda controller activation, as counted by CTR_AFL_CYC[i]. This ensures that T_AFL_CYC_HLD_R_IT_LS_DOWN[i] is valid.

The variable threshold VLS_DOWN_MMV_DRV1_THD_MIN[i] shall be limited so that its value is not less than that of VLS_DOWN_DRV1_MMV_MIN[i] * C_FAC_VLS_DOWN_MMV_DRV1_THD_MIN.

The function shall then wait for the oxygen sensor signal gradient to exceed the minimum threshold (VLS_DOWN_DRV1_MMV[i] ≥ VLS_DOWN_MMV_DRV1_THD_MIN[i]) and then determine whether the absolute oxygen sensor signal voltage is within a particular range of the VLS_DOWN_MMV_MIN[i] signal (VLS_DOWN_MMV_MIN[i] - C_VLS_DOWN_MMV_MIN_RNG_BOL_R_IT ≤ VLS_DOWN[i] ≤ VLS_DOWN_MMV_MIN[i] + C_VLS_DOWN_MMV_MIN_RNG_TOL_R_IT). If this is the case, the VLS_DOWN[i] shall be considered to be consistent with previous lean phase amplitudes and resistance determination may be carried out. This condition shall only be carried out once per lean phase. i.e. at transition from rich to lean. The fulfilment of the gradient condition is indicated by LV_VLS_DOWN_DRV1_CDN_R_LS[i] and the fulfilment of the range condition by LV_VLS_DOWN_RNG_CDN_R_LS[i].

If the range condition has been met, the gradient is in a particular range (VLS_DOWN_MMV_DRV1_THD_MIN[i] ≤ VLS_DOWN_DRV1_MMV[i] ≤ VLS_DOWN_MMV_DRV1_THD_MAX[i]) and the oxygen sensor signal is determined to be stable in the short term (VLS_DOWN_DRV1_ABS_MAX[i] ≤ C_VLS_DOWN_MMV_DRV1_ABS_MAX * FAC_T_VLS_DOWN_MMV_DRV1_ABS[i]) then resistance determination is requested (LV_R_IT_REQ_LS_DOWN[i] = 1). The factor FAC_T_VLS_DOWN_MMV_DRV1_ABS[i] shall be set to 1 unless C_CTR_AFL_CYC_MIN_RI_IT_LS_DOWN AF cycles have passed since the last lambda controller activation, as counted by CTR_AFL_CYC[i]. This ensures that T_AFL_CYC_HLD_R_IT_LS_DOWN[i] is valid.

Once internal resistance determination has been requested, the Basic SW shall take a single VLS_DOWN[i] sample and store as VLS_DOWN_CUR_PUMP_OFF[i]. SW1 shall be switched ON, at time C_T_CUR_PUMP_ON_LS_DOWN[i] and the VLS_DOWN[i] shall be masked. A second oxygen sensor signal voltage sample shall be taken and stored as VLS_DOWN_CUR_PUMP_ON[i]. SW1 shall then be switched OFF and masking shall continue for a time C_T_DLY_CUR_PUMP_LS_DOWN, after which the VLS_DOWN[i] shall return to normal following of the oxygen sensor signal voltage. Masking shall be discontinued if the oxygen sensor signal voltage exceeds VLS_DOWN_CUR_PUMP_ON[i] + C_VLS_HYS_R_IT_LS_DOWN during the time C_T_DLY_CUR_PUMP_LS_DOWN after switching off SW1. Adding the voltage hysteresis shall prevent the function to discard lambda sensor voltage masking if the first VLS_DOWN[i] value after switching off the pump current is larger than the last VLS_DOWN[i] value while pumping due to signal noise.

The internal resistance R_IT_MES_LS_DOWN[i] shall be determined using the formula stated with the main variables being oxygen sensor voltage prior to pumping, VLS_DOWN_CUR_PUMP_OFF[i], and signal voltage during pumping, VLS_DOWN_CUR_PUMP_ON[i].

The internal resistance shall be calculated every time the oxygen sensor is pumped. For each new resistance value detected, a counter CTR_CYCNR_R_IT_LS_DOWN_VLD[i] shall be incremented to indicate to other functions that a new resistance value is available.

NOTE: During the time where SW1 is switched on, (LV_R_IT_REQ_LS_DOWN[i]=1) the output voltage of the oxygen sensor (VLS_DOWN[i]) does not represent the true A/F ratio and is therefore invalid and shall not be used as such. The basic software shall mask the response of the sensor to pumping by keeping the voltage constant during this time by using the last VLS_DOWN[i] sample prior to activating SW1.

The internal resistance shall be calculated every time the oxygen sensor is pumped. For each new resistance value detected, the validity of the resistance shall be checked:

Should the validity function not be inhibited (LC_R_IT_VLD_INH_LS_DOWN = 0), then the tip temperature of the ceramic shall be determined from a map. TTIP_MES_LS_DOWN[i] represents the nominal temperature, TTIP_MES_TOL_LS_DOWN[i] represents the maximum temperature (R_IT_MES_LS_DOWN[i] * (1 – C_FAC_R_IT_ERR_NEG_LS_DOWN)), i.e. temperature for resistance with negative measurement error and TTIP_MES_BOL_LS_DOWN[i] represents the minimum temperature (R_IT_MES_LS_DOWN[i] * (1 + C_FAC_R_IT_ERR_NEG_LS_DOWN)), i.e. temperature for resistance with positive measurement error. A low pass filter shall then be applied to the upper and lower temperature limits to represent the maximum possible temperature gradient between resistance samples. Dependent on the direction of change of resistance between two measurements, either constant C_CRLC_TTIP_RISE_R_IT_LS_DOWN (rising temp.) or C_CRLC_TTIP_FALL_R_IT_LS_DOWN (falling temp.) shall be applicable. The final filter constant used, limited to a maximum of 1, shall be the aforementioned constant multiplied by the time between resistance samples (T_SAMPLE_R_IT_LS_DOWN[i]). This shall be to compensate for the irregular sample time.

Should then the ceramic tip temperature fall within the temperature limit bounds, the measured resistance shall be considered to be valid, shall be passed to the output variable R_IT_LS_DOWN[i], the counter CTR_CYCNR_R_IT_LS_DOWN_VLD[i] shall be incremented to indicate to other functions that a new resistance value is available and the T_SAMPLE_R_IT_LS_DOWN[i] shall be reset. If the above temperature condition is not met, R_IT_LS_DOWN[i] shall be maintained at its last value and the filtered temperature limits shall be reset to their previous values, i.e. the resistance value shall be ignored.

Should the validity function be inhibited then for each new resistance the R_IT_MES_LS_DOWN[i] shall be passed to R_IT_LS_DOWN[i] and the counter CTR_CYCNR_R_IT_LS_DOWN_VLD[i] shall be incremented.

In the case of significant oxygen sensor amplitude (LV_VLS_DOWN_MMV_LIM = 0) and the sensor signal be in the rich phase (VLS_DOWN[i] > VLS_DOWN_TOL[i]), the flags LV_VLS_DOWN_RNG_CDN_R_LS[i] & LV_VLS_DOWN_DRV1_CDN_R_LS[i] shall be reset. Furthermore, if these conditions have been met and the timer *TIMER_1* exceeds C_T_CUR_PUMP_OFF_LS_DOWN then LV_R_IT_DET_LS_DOWN[i] shall be reset, thereby permitting the internal resistance to be determined in the next cycle should the appropriate conditions be met.

For reference reasons of OBD II diagnosis the value R_IT_MDL_LS_DOWN_NEW[i] is calculated out of heating power, exhaust gas temperature and MAF_KGH.

Application conditions:

Recurrence: T_SAMPLE_1 = 10 ms

The function shall be carried out once every 10 ms. The internal resistance value shall only be calculated after “pumping” the oxygen sensor. See recurrence there. Reference internal rsistance shall be calculated with recurrence 1 s.

Initialisation:

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop (LV_ES).

R_IT_LS_DOWN[i] = 65535
R_IT_MES_LS_DOWN[i] = 65535
R_IT_MDL_LS_DOWN_NEW[i] = 65535
POW_MMV_LSH_DOWN[i] = 0
VLS_DOWN_CUR_PUMP_ON[i] = 0
VLS_DOWN_CUR_PUMP_OFF[i] = 0
CTR_CYCNR_R_IT_LS_DOWN_VLD[i] = 0
LV_VLS_DOWN_DRV1_CDN_R_LS[i] = 0
LV_VLS_DOWN_RNG_CDN_R_LS[i] = 0
LV_R_IT_REQ_LS_DOWN[i] = 0
LV_R_IT_DET_LS_DOWN[i] = 0
VLS_DOWN_MMV_DRV1_THD_MIN[i] = 0
VLS_DOWN_MMV_DRV1_THD_MAX[i] = 0
T_AFL_CYC_HLD_R_IT_LS_DOWN[i] = 0
FAC_T_VLS_DOWN_MMV_DRV1[i] = 0
FAC_T_VLS_DOWN_MMV_DRV1_ABS[i] = 0
TTIP_REF_MDL_MMV_LS_DOWN[i] = 0
TTIP_MES_LS_DOWN[i] = 0
TTIP_MES_TOL_LS_DOWN[i] = 0
TTIP_MES_BOL_LS_DOWN[i] = 0
TTIP_MES_TOL_MMV_LS_DOWN[i] = 0
TTIP_MES_BOL_MMV_LS_DOWN[i] = 0
T_SAMPLE_R_IT_LS_DOWN[i] = 655.35

NOTE: All “N-1” variables shall also be initialised to the associated value specified above.

Activation / Deactivation:

If LV_ST_END = 1
then ***Function activated***
else ***Function deactivated***
endif

Formula section:

If LV_INH_R_IT_LS_DOWN[i] = 0
and LV_VLS_DOWN_MMV_ACT = 1
and VLS_DOWN_MMV_MIN[i] < VLS_DOWN_MMV_MAX[i]
and MAF_KGH < C_MAF_KGH_MAX_R_IT_LS_DOWN
and STATE_LSH_DOWN[i] = "LSH_POW_CTL"
and (LV_ERR_LSH_DOWN[i] = 0
or TEMP_CAT > C_TEMP_CUR_PUMP_ON_LS_DOWN)
then ***Update*** *TIMER_1*
Increment T_SAMPLE_R_IT_LS_DOWN[i]
If LV_R_IT_REQ_LS_DOWN[i] = 0
then **If** LV_VLS_DOWN_MMV_LIM[i] = 0
then **If** VLS_DOWN[i] < VLS_DOWN_BOL[i]
then T_AFL_CYC_HLD_R_IT_LS_DOWN[i] =

T_AFL_CYC_HLD[i]

If LV_R_IT_DET_LS_DOWN[i] = 0
then "*Determine lean cycle time dependent factors"*
"*Compute variable gradient thresholds"*
"Limit lower variable gradient threshold"
"Determine VLS_DOWN at low gradient condition met"
If LV_VLS_DOWN_RNG_CDN_R_LS[i] = 1
and VLS_DOWN_DRV1_MMV[i] ≥ VLS_DOWN_MMV_DRV1_THD_MIN[i]
and VLS_DOWN_DRV1_MMV[i] ≤ VLS_DOWN_MMV_DRV1_THD_MAX[i]
and VLS_DOWN_DRV1_ABS_MAX[i] ≤ C_VLS_DOWN_MMV_DRV1_ABS_MAX * FAC_T_VLS_DOWN_MMV_DRV1_ABS[i])
then LV_R_IT_REQ_LS_DOWN[i] = 1
VLS_DOWN_CUR_PUMP_OFF[i] = VLS_DOWN[i]
Switch SW1 ON
"Determine internal resistance" started
endif
endif
else **If** VLS_DOWN[i] > VLS_DOWN_TOL[i]
then LV_VLS_DOWN_RNG_CDN_R_LS[i] = 0
LV_VLS_DOWN_DRV1_CDN_R_LS[i] = 0
If *TIMER_1* ≥ C_T_CUR_PUMP_OFF_LS_DOWN
then LV_R_IT_DET_LS_DOWN[i] = 0
endif
endif
endif
else **If** (*TIMER_1* ≥ C_T_CUR_PUMP_OFF_LS_DOWN)
then LV_R_IT_REQ_LS_DOWN[i] = 1
VLS_DOWN_CUR_PUMP_OFF[i] = VLS_DOWN[i]
Switch SW1 ON
"Determine internal resistance" started
endif
endif

endif

If LV_R_IT_REQ_LS_DOWN[i] = 1 -> 0

then "*Calculation of Internal resistance*"

LV_R_IT_DET_LS_DOWN[i] = 1

Reset *TIMER_1*

endif

else ***Reset*** *TIMER_1*

T_SAMPLE_R_IT_LS_DOWN[i] = 655.35 (i.e. FFFFH)

LV_R_IT_DET_LS_DOWN[i] = 0

LV_VLS_DOWN_DRV1_CDN_R_LS[i] = 0

LV_VLS_DOWN_RNG_CDN_R_LS[i] = 0

endif

"Calculation of internal resistance reference"

NOTES:

1. ***TIMER_1 refers to a SW internal timer and not a visible output data variable.***
2. ***T_SAMPLE_R_LS_DOWN[i] shall not be permitted to overflow. When the specified maximum value has been reached, this shall be held until the timer is reset by the function.***

Calculation of internal resistance reference:

% This calculation is carried out with the recurrence: T_SAMPLE_2 = 1 s.

POW_MMV_LSH_DOWN$[i]_n$ = POW_MMV_LSH_DOWN$[i]_{n-1}$ *

(1 - C_CRLC_POW_MMV_LSH_DOWN) +
C_CRLC_POW_MMV_LSH_DOWN *

V_EFC_LSH_DOWN$[i]^2$ * T_SAMPLE_2 * [W/(s*$V^{2)}$]

TTIP_REF_MDL_MMV_LS_DOWN$[i]_n$ = TTIP_REF_MDL_MMV_LS_DOWN$[i]_{n-1}$ *

(1 - IP_CRLC_TTIP_REF_MDL_LS_DOWN) +

IP_CRLC_TTIP_REF_MDL_LS_DOWN * TEG_CAT_DOWN_MDL[i]

% IP (MAF_KGH)

R_IT_MDL_LS_DOWN_NEW[i] = IP_R_IT_MDL_LS_DOWN_NEW

% IP (POW_MMV_LSH_DOWN[i]; TTIP_REF_MDL_MMV_LS_DOWN[i])

Determine lean cycle time dependent factors:

If (CTR_AFL_CYC[i] ≥ C_CTR_AFL_CYC_MIN_RI_IT_LS_DOWN)

then FAC_T_VLS_DOWN_MMV_DRV1[i] =
IP_FAC_T_DRV1_VLS_DOWN_MMV (T_AFL_CYC_HLD_R_IT_LS_DOWN[i])

FAC_T_VLS_DOWN_MMV_DRV1_ABS[i] =
IP_FAC_T_VLS_DOWN_MMV_DRV1_ABS (T_AFL_CYC_HLD_R_IT_LS_DOWN[i])

else FAC_T_VLS_DOWN_MMV_DRV1[i] = 1
FAC_T_VLS_DOWN_MMV_DRV1_ABS[i] = 1

endif.

Compute variable gradient thresholds:

VLS_DOWN_MMV_DRV1_THD_MAX[i] = C_VLS_DOWN_MMV_DRV1_THD_MAX *
C_FAC_VLS_DOWN_MMV_DRV1_THD *
(VLS_DOWN_MMV_MAX[i] -
VLS_DOWN_MMV_MIN[i])

VLS_DOWN_MMV_DRV1_THD_MIN[i] = C_VLS_DOWN_MMV_DRV1_THD_MIN *
C_FAC_VLS_DOWN_MMV_DRV1_THD *
(VLS_DOWN_MMV_MAX[i] -
VLS_DOWN_MMV_MIN[i]) *
FAC_T_VLS_DOWN_MMV_DRV1[i]

Limit lower variable gradient threshold:

VLS_DOWN_MMV_DRV1_THD_MIN[i] = MAX(VLS_DOWN_MMV_DRV1_THD_MIN[i],
VLS_DOWN_DRV1_MMV_MIN[i] *
C_FAC_VLS_DOWN_MMV_DRV1_THD_MIN)

Determine VLS_DOWN at low gradient condition met:

If (LV_VLS_DOWN_DRV1_CDN_R_LS[i] = 0) **&**
(VLS_DOWN_DRV1_MMV[i] ≥ VLS_DOWN_MMV_DRV1_THD_MIN[i])

then LV_VLS_DOWN_DRV1_CDN_R_LS[i] = 1
If (VLS_DOWN[i] ≤
(VLS_DOWN_MMV_MIN[i] + C_VLS_DOWN_MMV_MIN_RNG_TOL_R_IT)) **&**
(VLS_DOWN[i] ≥
(VLS_DOWN_MMV_MIN[i] - C_VLS_DOWN_MMV_MIN_RNG_BOL_R_IT))

then LV_VLS_DOWN_RNG_CDN_R_LS[i] = 1

else LV_VLS_DOWN_RNG_CDN_R_LS[i] = 0

endif

endif.

NOTE: VLS_DOWN_MMV_MIN[i] + C_VLS_DOWN_MMV_MIN_RNG_TOL_R_IT shall not cause an overflow and VLS_DOWN_MMV_MIN[i] - C_VLS_DOWN_MMV_MIN_RNG_BOL_R_IT shall not cause an underflow.

Determine internal resistance:

Recurrence: T_SAMPLE_3 = 1 ms

If LV_R_IT_REQ_LS_DOWN[i] = 1

then

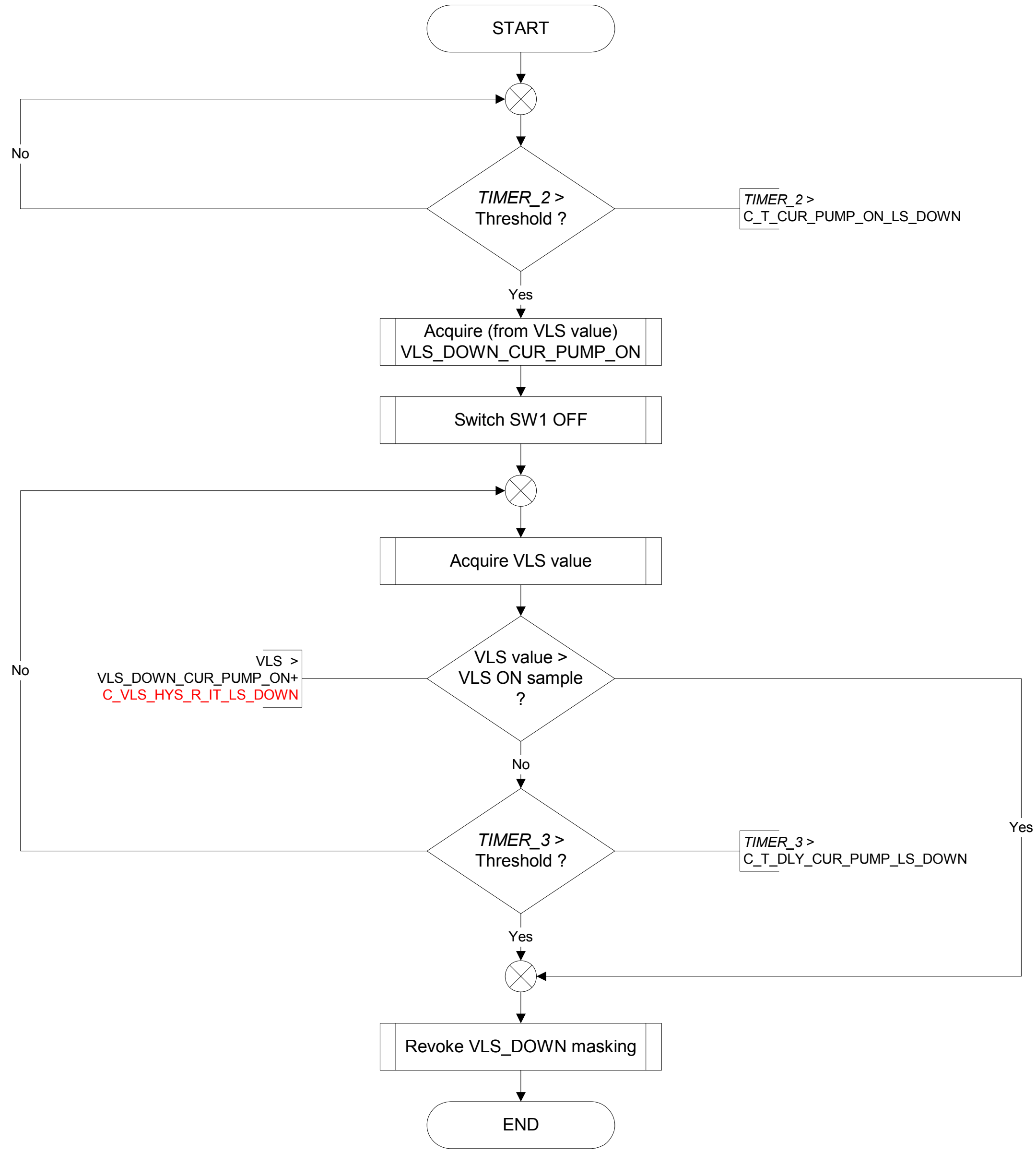


Figure 3 Function flow chart, Internal resistance determination

LV_R_IT_REQ_LS_DOWN[i] = 0

endif

Calculation of Internal resistance:

The internal resistance shall be determined as follows:

R_IT_MES_LS_DOWN[i] = NC_R_REF_LS_DOWN * (VLS_DOWN_CUR_PUMP_ON[i] - VLS_DOWN_CUR_PUMP_OFF[i]) / {NC_VLS_DOWN_CUR_PUMP_REF - (VLS_DOWN_CUR_PUMP_ON[i] - VLS_DOWN_CUR_PUMP_OFF[i])* NC_FAC_R_REF_LS_DOWN - VLS_DOWN_CUR_PUMP_ON[i]}

Determine internal resistance validity:

The following function shall be carried once out for every new resistance value determined.

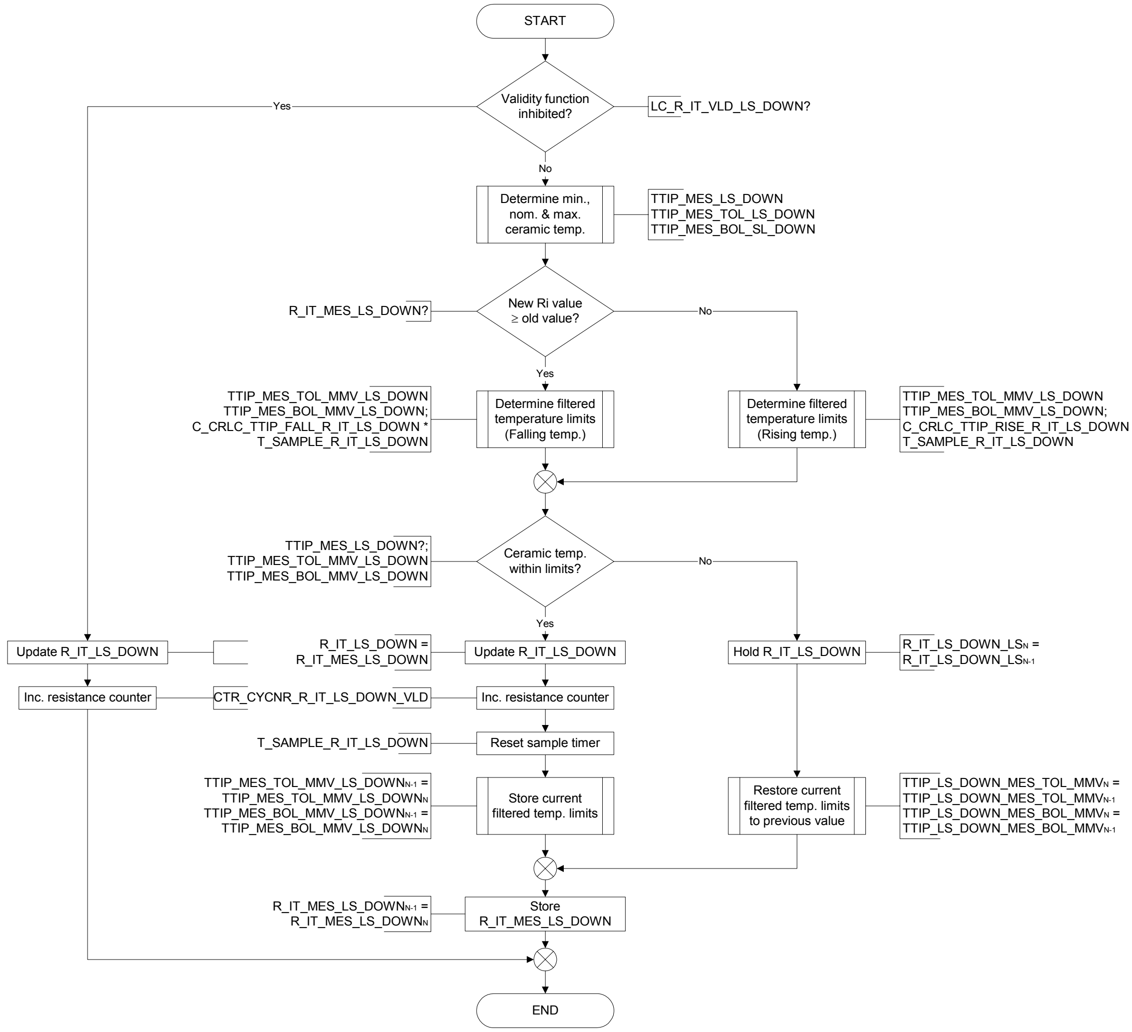


Figure 4 Function flow chart, Internal resistance validity determination

If (LC_R_IT_VLD_INH_LS_DOWN = 0)

then

TTIP_MES_LS_DOWN[i] = IP_TTIP_MES_LS_DOWN (R_IT_MES_LS_DOWN[i])
TTIP_MES_TOL_LS_DOWN[i] = IP_TTIP_MES_LS_DOWN (R_IT_MES_LS_DOWN[i] * (1 - C_FAC_R_IT_ERR_NEG_LS_DOWN))
TTIP_MES_BOL_LS_DOWN[i] = IP_TTIP_MES_LS_DOWN (R_IT_MES_LS_DOWN[i] * (1+ C_FAC_R_IT_ERR_POS_LS_DOWN))

If (R_IT_MES_LS_DOWN[i]$_{N}$ ≥ R_IT_MES_LS_DOWN[i]$_{N-1}$)

then

TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N}$ = TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$ + (TTIP_MES_TOL_LS_DOWN[i] - TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$) * MIN(1, C_CRLC_TTIP_FALL_R_IT_LS_DOWN * T_SAMPLE_R_IT_LS_DOWN[i])

TTIP_MES_BOL_MMV_LS_DOWN[i]$_N$ = TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$ +
(TTIP_MES_BOL_LS_DOWN[i] - TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_FALL_R_IT_LS_DOWN * T_SAMPLE_R_IT_LS_DOWN[i])

else
TTIP_MES_TOL_MMV_LS_DOWN[i]$_N$ = TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$ +
(TTIP_MES_TOL_LS_DOWN[i] - TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_RISE_R_IT_LS_DOWN * T_SAMPLE_R_IT_LS_DOWN[i])

TTIP_MES_BOL_MMV_LS_DOWN[i]$_N$ = TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$ +
(TTIP_MES_BOL_LS_DOWN[i] - TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_RISE_R_IT_LS_DOWN * T_SAMPLE_R_IT_LS_DOWN[i])

endif

If ((TTIP_MES_LS_DOWN[i] ≤ TTIP_MES_TOL_MMV_LS_DOWN[i]) **&**
(TTIP_MES_LS_DOWN[i] ≥ TTIP_MES_BOL_MMV_LS_DOWN[i]))

then
R_IT_LS_DOWN[i] = R_IT_MES_LS_DOWN[i]$_N$
Increment CTR_CYCNR_R_IT_LS_DOWN_VLD[i]
Reset T_SAMPLE_R_IT_LS_DOWN[i]
TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$ = TTIP_MES_TOL_MMV_LS_DOWN[i]$_N$
TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$ = TTIP_MES_BOL_MMV_LS_DOWN[i]$_N$

else
R_IT_LS_DOWN[i]$_N$ = R_IT_LS_DOWN[i]$_{N-1}$
TTIP_MES_TOL_MMV_LS_DOWN[i]$_N$ = TTIP_MES_TOL_MMV_LS_DOWN[i]$_{N-1}$
TTIP_MES_BOL_MMV_LS_DOWN[i]$_N$ = TTIP_MES_BOL_MMV_LS_DOWN[i]$_{N-1}$

endif

R_IT_MES_LS_DOWN[i]$_{N-1}$ = R_IT_MES_LS_DOWN[i]$_N$

else
R_IT_LS_DOWN[i] = R_IT_MES_LS_DOWN[i]$_N$
Increment CTR_CYCNR_R_IT_LS_DOWN_VLD[i]

endif.

NOTES:

1. ***The difference (VLS_DOWN_CUR_PUMP_ON[i] - VLS_DOWN_CUR_PUMP_OFF[i]) shall be limited to 0 for negative values as the voltage during pumping should be greater than that without.***
2. ***If the incrementation would cause the counter CTR_CYCNR_R_IT_LS_DOWN_VLD[i] to overflow then the counter shall be reset to 0.***
3. ***The basic software (BSW) shall mask the VLS_DOWN[i] signal during resistance determination by retaining the value of the last sample prior to activating the resistance determination. This shall be indicated by setting LV_R_IT_REQ_LS_DOWN[i]. Upon revoking the masking the last oxygen sensor signal voltage VLS value determined by the BSW shall be used for the VLS_DOWN[i] signal and LV_R_IT_REQ_LS_DOWN[i] shall be reset.***
4. ***Should C_T_CUR_PUMP_ON_LS_DOWN = 0, SW1 shall not be activated.***

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEMP_CUR_PUMP_ON_LS_DOWN	1	0...FFFFH	-33...990	0.0156	°C
Temperature threshold for the activation of sensor pumping					
C_T_CUR_PUMP_ON_LS_DOWN	1	0...08H	0...8e-3	1e-3	s
Duration for pump current active					
C_T_CUR_PUMP_OFF_LS_DOWN	1	0...FFFFH	0...655.35	10e-3	s
Minimum delay between consecutive pump current activation					
C_T_DLY_CUR_PUMP_LS_DOWN	1	0...FFH	0...255e-3	1e-3	s
Duration in which VLS_DOWN[NC_CBK_EX_NR] signal masked directly after Ri determination					
C_VLS_HYS_R_IT_LS_DOWN	1	0...3FFH	0...4.995117	4.8828e-3	V
Voltage hysteresis for evaluation if VLS_DOWN[i] shall be masked after Ri determination					
C_VLS_DOWN_MMV_MIN_RNG_TOL_R_IT	1	0...FFFFH	0...4.999665	7.629e-5	V
Permitted range of VLS_DOWN[NC_CBK_EX_NR] above VLS_DOWN_MMV_MIN[NC_CBK_EX_NR] at low gradient for Ri determination					
C_VLS_DOWN_MMV_MIN_RNG_BOL_R_IT	1	0...FFFFH	0...4.999665	7.629e-5	V
Permitted range of VLS_DOWN[NC_CBK_EX_NR] below VLS_DOWN_MMV_MIN[NC_CBK_EX_NR] at low gradient for Ri determination					
C_VLS_DOWN_MMV_DRV1_THD_MIN	1	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Minimum threshold for mean 1st derivative of VLS_DOWN[NC_CBK_EX_NR] signal for Ri determination					
C_VLS_DOWN_MMV_DRV1_THD_MAX	1	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Maximum threshold for mean 1st derivative of VLS_DOWN[NC_CBK_EX_NR] signal for Ri determination					
C_VLS_DOWN_MMV_DRV1_ABS_MAX	1	0...FFFFH	0...4.999665	7.629e-5	V/10ms
Threshold for absolute maximum 1st derivative of VLS_DOWN[NC_CBK_EX_NR] signal for Ri determination					
C_FAC_VLS_DOWN_MMV_DRV1_THD	1	0...FFH	0...1.992	7.81e-3	-
Multiplicative factor used to weight gradient thresholds dependent on VLS_DOWN_MMV_xxx[NC_CBK_EX_NR] amplitude					
C_FAC_VLS_DOWN_MMV_DRV1_THD_MIN	1	0...FFH	0...0.999600	3.92e-3	-
Hysteresis factor of DRV1_VLS_DOWN_MMV_MIN governing limiting of DRV1_VLS_DOWN_MMV_MIN_THD					
IP_FAC_T_DRV1_VLS_DOWN_MMV	1*8	0...FFH	0...63.75	0.25	-
LDPM_T_AFL_CYC_HLD_1_EGCP	8	0...FFH	0...2.55	10e-3	s
Multiplicative factor modifies DRV1_VLS_DOWN_MMV_MIN_THD dependent on T_VLS_CYC_AFL_HLD					
IP_CRLC_TTIP_REF_MDL_LS_DOWN	1*8	0...FFH	0...0.999600	3.92e-3	-
LDPM_MAF_KGH_1_EGCP	8	0...FFFFH	0...2047.9687	0.03125	kg/h
Correlation map for calculation of reference downstream tip temperature (modelled for new sensor)					
IP_R_IT_MDL_LS_DOWN_NEW	8*8	0...FFFFH	0...65535	1	Ω
LDP_POW_MMV_LSH_DOWN_IP_LS_DOWN	8	0...FFFFH	0...84.498711	1.2837E-3	W
LDP_TTIP_REF_MDL_MMV_LS_DOWN_R	8	0...7FFFH	0...2047.938	62.5e-3	°C
Internal resistance of a new downstream oxygen sensor (modelled)					
IP_FAC_T_VLS_DOWN_MMV_DRV1_ABS	1*8	0...FFH	0...15.9375	62.5e-3	-
LDPM_T_AFL_CYC_HLD_1_EGCP	8	0...FFH	0...2.55	10e-3	s
Multiplicative factor modifies C_VLS_DOWN_MMV_DRV1_ABS_MAX dependent on T_VLS_CYC_AFL_HLD					
C_CTR_AFL_CYC_MIN_RI_IT_LS_DOWN	1	0...FFFFH	0...65535	1	-
Min. no. AF cycles post lambda controller activation to permit use of T_VLS_CYC_AFL_HLD dependency					
C_MAF_KGH_MAX_R_IT_LS_DOWN	1	0...FFFFH	0...2047.96875	31.25e-3	kg/h
Maximum MAF_KGH threshold for activation of internal resistance determination					
C_FAC_R_IT_ERR_POS_LS_DOWN	1	0...FFH	0...0.999600	3.92e-3	-
Internal resistance measurement positive error (i.e. 0% to 100%)					
C_FAC_R_IT_ERR_NEG_LS_DOWN	1	0...FFH	0...0.999600	3.92e-3	-
Internal resistance measurement negative error (i.e. 100% to 0%)					
IP_TTIP_MES_LS_DOWN	1*8	0...7FFFH	0...2047.938	62.5e-3	°C
LDP_R_IT_MES_LS_DOWN_IP_LS_DOWN	8	0...FFFFH	0...65535	1	Ω
Oxygen sensor temperature vs. internal resistance characteristic					

C_CRLC_POW_MMV_LSH_DOWN	1	0...FFH	0...0.999600	3.92e-3	-
Correlation (filter) constant for moving mean value of downstream heater power					
C_CRLC_TTIP_FALL_R_IT_LS_DOWN	1	0...FFH	0...0.999600	3.92e-3	s^{-1}
Correlation (filter) constant for falling ceramic temperature, limit calculation					
C_CRLC_TTIP_RISE_R_IT_LS_DOWN	1	0...FFH	0...0.999600	3.92e-3	s^{-1}
Correlation (filter) constant for rising ceramic temperature, limit calculation					
LC_R_IT_VLD_INH_LS_DOWN	1	0...1H	0...1	1	-
Boolean flag inhibiting internal resistance determination validity function					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_R_REF_LS_DOWN	1	0...FFFFH	0...65535	1	Ω
Reference resistance for pump current (R_{Pump})					
NC_FAC_R_REF_LS_DOWN	1	0...FFFFH	0...3.9994	6.1035e-5	-
Pump current limiting to pull down resistance ratio (R_{Pump}/R_2)					
NC_VLS_DOWN_CUR_PUMP_REF	1	0...3FFH	0... 4.995117	4.8828e-3	V
Reference voltage value (4.82 V)					

4.64.1 Application specific conditions for oxygen sensor internal resistance

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_R_IT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that downstream sensor internal resistance determination shall be inhibited when set					

Input data:

NC_CBK_EX_NR	LV_ES		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

The Boolean flag LV_INH_R_IT_LS_DOWN[i] shall indicate whether the downstream oxygen sensor internal resistance determination shall be permitted to take place. When set, no resistance determination shall take place.

Description:

No application specific conditions have been defined to date. Internal resistance determination shall always take place (LV_INH_R_IT_LS_DOWN[i] shall be initialised to and remain 0).

Application conditions:

Initialisation:

The following initialisation shall be carried out upon leaving the engine state Engine Stop (LV_ES).

LV_INH_R_IT_LS_DOWN[i] = 0

Activation:

None defined at present.

Deactivation:

None defined at present.

Formula section:

None defined at present.

4.65 Upstream oxygen sensor internal resistance determination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
R_IT_LS_UP[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	Ω
Internal resistance of upstream oxygen sensor (Released)					
R_IT_MES_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω
Internal resistance of upstream oxygen sensor (Measured)					
R_IT_MDL_LS_UP_NEW[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	Ω
Internal resistance of upstream oxygen sensor (Modelled for new sensor)					
POW_MMV_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...84.498711	1.2837E-3	W
Moving mean value of upstream oxygen sensor heater power					
TTIP_REF_MDL_MMV_LS_UP[NC_CBK_EX_NR]	V	0...7FFFH	0...2047.938	62.5e-3	°C
Reference of upstream oxygen sensor ceramic temperature (Modelled for new sensor)					
VLS_UP_CUR_PUMP_ON[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828e-3	V
Upstream oxygen sensor voltage C_T_CUR_PUMP_ON_LS_UP after switching on pump current switch					
VLS_UP_CUR_PUMP_OFF[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828e-3	V
Upstream oxygen sensor voltage prior to switching on pump current switch					
CTR_CYCNR_R_IT_LS_UP_VLD[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	-
Handshake counter indicating the number of valid resistance values determined since initial activation					
LV_VLS_UP_DRV1_CDN_R_LS[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that low gradient condition met					
LV_VLS_UP_RNG_CDN_R_LS[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that VLS_UP at low gradient condition met					
LV_R_IT_REQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that conditions have been met and resistance determination requested					
LV_R_IT_DET_LS_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that resistance has already been determined (lock out)					
VLS_UP_MMV_DRV1_THD_MIN[NC_CBK_EX_NR]	V	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Lower variable VLS_UP gradient threshold					
VLS_UP_MMV_DRV1_THD_MAX[NC_CBK_EX_NR]	V	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Upper variable VLS_UP gradient threshold					
FAC_T_VLS_UP_MMV_DRV1[NC_CBK_EX_NR]	V	0...FFH	0...63.75	0.25	-
Multiplicative factor modifies DRV1_VLS_UP_MMV_MIN_THD dependent on T_VLS_CYC_AFL_HLD					
FAC_T_VLS_UP_MMV_DRV1_ABS[NC_CBK_EX_NR]	V	0...FFH	0...15.9375	62.5e-3	-
Multiplicative factor modifies C_VLS_UP_MMV_DRV1_ABS_MAX dependent on T_VLS_CYC_AFL_HLD					
TTIP_MES_LS_UP[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Upstream oxygen sensor ceramic temperature					
TTIP_MES_TOL_LS_UP[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Upstream oxygen sensor ceramic temperature, Upper limit, Unfiltered					
TTIP_MES_BOL_LS_UP[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Upstream oxygen sensor ceramic temperature, Lower limit, Unfiltered					
TTIP_MES_TOL_MMV_LS_UP[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Upstream oxygen sensor ceramic temperature, Upper limit, Filtered					

TTIP_MES_BOL_MMV_LS_UP[NC_CBK_EX_NR]	V	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Upstream oxygen sensor ceramic temperature, Lower limit, Filtered					
LV_V_REF_VLD_R_IT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating internal resistance determination circuit references valid					
LV_TTIP_MES_VLD_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating validity of oxygen sensor ceramic temperature TTIP_MES_LS_UP[NC_CBK_EX_NR]					
T_SAMPLE_R_IT_VLD_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	10e-3	s
Time measured between internal resistance samples to reset valid flag					
T_SAMPLE_R_IT_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	10e-3	s
Time measured between internal resistance samples					

Input data:

LV_ST_END	VLS_UP[NC_CBK_EX_NR]	VLS_UP_DRV1_MMV[NC_CBK_EX_NR]	LV_INH_R_IT_LS_UP[NC_CBK_EX_NR]
LV_ES	VLS_UP_MMV_MIN[NC_CBK_EX_NR]	VLS_UP_DRV1_MMV_MIN[NC_CBK_EX_NR]	LV_VLS_UP_MMV_LIM[NC_CBK_EX_NR]
TEG_DYN	VLS_UP_MMV_MAX[NC_CBK_EX_NR]	VLS_UP_DRV1_ABS_MAX[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]
MAF_KGH	VLS_UP_BOL[NC_CBK_EX_NR]	T_AFL_CYC_HLD[NC_CBK_EX_NR]	STATE_LSH_UP[NC_CBK_EX_NR]
V_EFC_LSH_UP[NC_CBK_EX_NR]	VLS_UP_TOL[NC_CBK_EX_NR]	CTR_AFL_CYC[NC_CBK_EX_NR]	LV_PUC
TEG_CAT_UP_MDL[NC_CBK_EX_NR]	NC_CBK_EX_NR		

FUNCTION DESCRIPTION:

General information:

The function shall provide the output variable R_IT_LS_UP[i], the internal resistance of the oxygen sensor. The resistance is dependent on the operating temperature of the sensor and may be used for oxygen sensor diagnosis.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

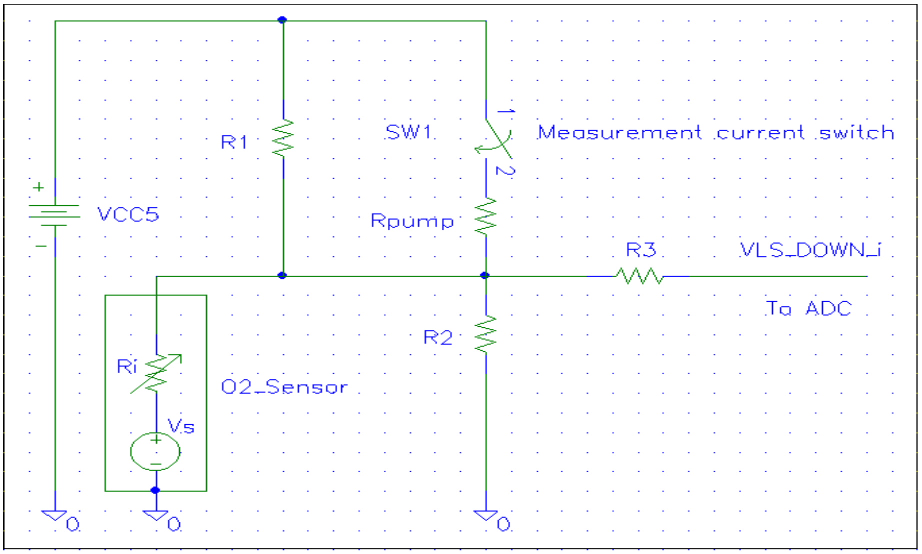


Figure 1 Simplified measurement circuit schematic, Internal resistance determination

Signal flow diagram:

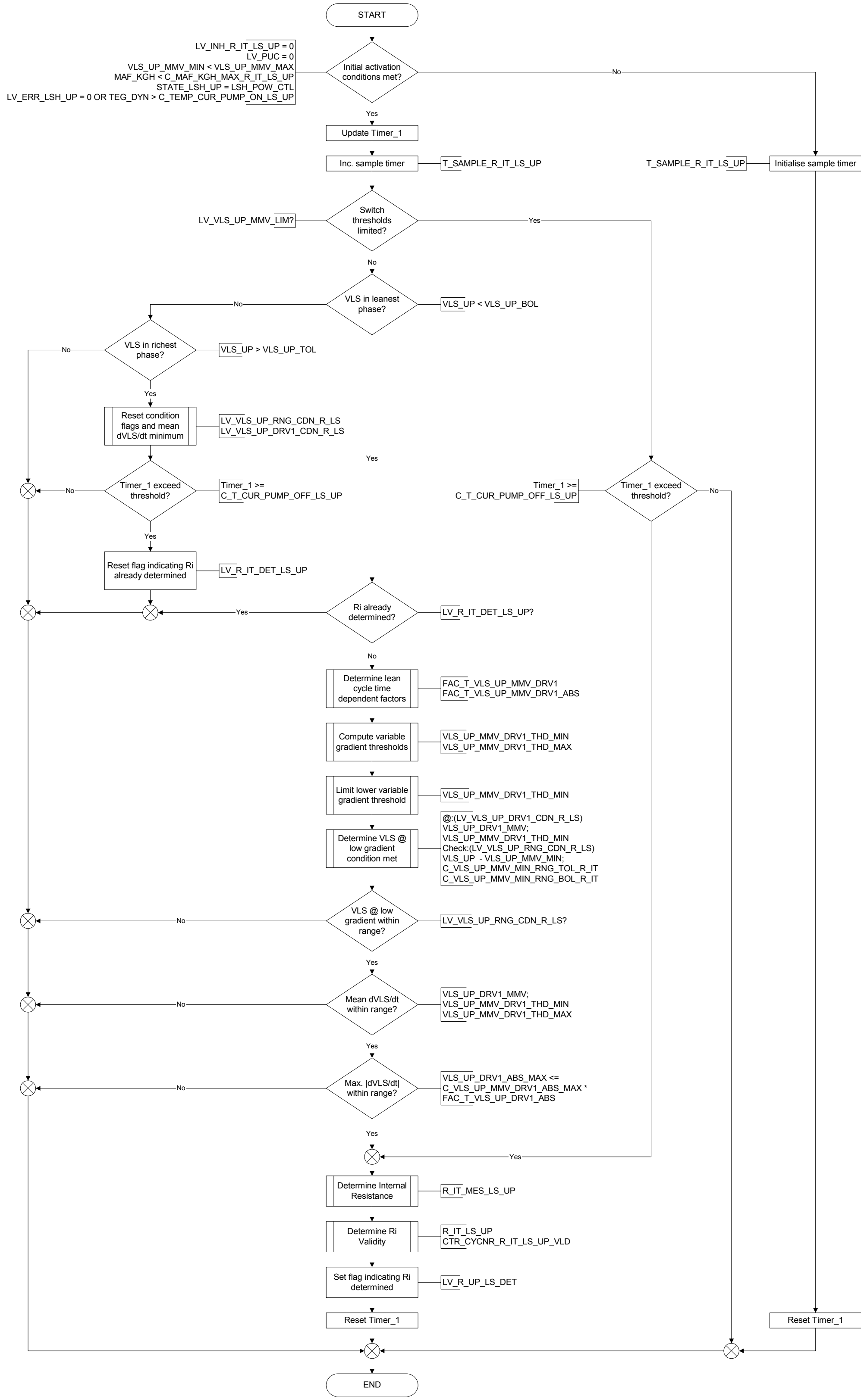


Figure 2 Function flow chart, Internal resistance determination

Description:

The internal resistance of the oxygen sensor shall be determined by use of the oxygen sensor interface circuit *Figure 1* (Capacitors not shown). The internal resistance provides information as to the tip temperature of the sensor and may be used for diagnosis purposes and operative readiness determination.

Under normal conditions, the switch SW1 shall be open and the oxygen sensor signal voltage (VLS_UP[i]) is measured by the microcontroller.

The potential divider (resistors R_1 & R_2) shall be permit the operation readiness of the oxygen sensor to be determined and enable electrical fault conditions to be detected.

Periodically, for a limited amount of time, the switch SW1 shall be closed and a voltage of approx. 4.8 V connected through resistor R_{pump} to the oxygen sensor signal connection. This shall cause current to flow into the sensor and through the internal resistance R_i, thereby "pumping" the sensor. The resultant voltage acquired at the microcontroller input shall permit the internal resistance of the sensor to be calculated by use of a formula shown in the formula section below.

Prior to permitting the determination of the internal resistance, a number of conditions shall be determined to be met:

The initial activation conditions shall require that: No function inhibit is requested (LV_INH_R_IT_LS_UP[i] = 0); Delay since deactivation since last overrun fuel cut-off passed (LV_PUC); The initial post initialisation transients have passed for the VLS_UP_MMV_xxx[i] inputs (VLS_UP_MMV_MIN[i] < VLS_UP_MMV_MAX[i]); The engine load threshold has not been exceeded (MAF_KGH < C_MAF_KGH_MAX_R_IT_LS_UP); The oxygen sensor heater state LSH_POW_CTL has been reached; No heater OBDI error has been detected in the relevant heater driver (LV_ERR_LSH_UP[i]) or should a heater driver fault have been detected, the modelled catalyst temperature TEG_DYN shall exceed the threshold C_TEMP_CUR_PUMP_ON_LS_UP

Once the above conditions have been met, an internal timer (*TIMER_1*) shall be started and timer T_SAMPLE_R_IT_LS_UP[i] shall be updated. The latter timer shall be initialised with its maximum value of 655.35 s. This ensures that the temperature limits for the validity function are correctly initialised. The further procedure shall depend on the status of the oxygen sensor signal, as determined by LV_VLS_UP_MMV_LIM. Should the VLS_UP_MMV_xxx[i] signals be limited, i.e. no or little oxygen sensor amplitude is present, the function shall determine the internal resistance every (C_T_CUR_PUMP_OFF_LS_UP + C_T_CUR_PUMP_ON_LS_UP + C_T_DLY_CUR_PUMP_LS_UP). Otherwise the oxygen sensor signal shall be determined to have significant amplitude and the internal resistance determination shall only be carried out in lean phases where the oxygen sensor signal is relatively flat and stable. This is achieved by the following conditions:

If the oxygen sensor signal is lean (VLS_UP[i] < VLS_UP_BOL[i]) and neither the resistance determination has been carried out yet in this lean phase nor the determination be locked out due to the minimum off time (LV_R_IT_DET_LS_UP[i] = 0) then the function shall determine the lean cycle time dependent factors FAC_T_VLS_UP_MMV_DRV1[i] & FAC_T_VLS_UP_MMV_DRV1_ABS[i] from the appropriate maps IP_FAC_T_DRV1_VLS_UP_MMV & IP_FAC_T_VLS_UP_MMV_DRV1_ABS respectively. These factors shall be dependent on the held lean cycle time from the previous lean cycle (T_AFL_CYC_HLD[i]). Should the cycle time decrease then it is intended that the factors shall increase and vice versa.

The thresholds for the gradient condition shall then be computed. These thresholds (VLS_UP_MMV_DRV1_THD_MAX[i] & VLS_UP_MMV_DRV1_THD_MIN[i]) are directly proportional to the amplitude of the difference between VLS_UP_MMV_xxx[i] signals. Furthermore, VLS_UP_MMV_DRV1_THD_MIN[i] shall be modified by factor

FAC_T_VLS_UP_MMV_DRV1[i]. The factor FAC_T_VLS_UP_MMV_DRV1[i] shall be set to 1 unless C_CTR_AFL_CYC_MIN_RI_IT_LS_UP AF cycles have passed since the last lambda controller activation, as counted by CTR_AFL_CYC[i]. This ensures that T_AFL_CYC_HLD[i] is valid.

The variable threshold VLS_UP_MMV_DRV1_THD_MIN[i] shall be limited so that its value is not less than that of VLS_UP_DRV1_MMV_MIN[i] * C_FAC_VLS_UP_MMV_DRV1_THD_MIN.

The function shall then wait for the oxygen sensor signal gradient to exceed the minimum threshold (VLS_UP_DRV1_MMV[i] ≥ VLS_UP_MMV_DRV1_THD_MIN[i]) and then determine whether the absolute oxygen sensor signal voltage is within a particular range of the VLS_UP_MMV_MIN[i] signal (VLS_UP_MMV_MIN[i] - C_VLS_UP_MMV_MIN_RNG_BOL_R_IT ≤ VLS_UP[i] ≤ VLS_UP_MMV_MIN[i] + C_VLS_UP_MMV_MIN_RNG_TOL_R_IT). If this is the case, the VLS_UP[i] shall be considered to be consistent with previous lean phase amplitudes and resistance determination may be carried out. This condition shall only be carried out once per lean phase. i.e. at transition from rich to lean. The fulfilment of the gradient condition is indicated by LV_VLS_UP_DRV1_CDN_R_LS[i] and the fulfilment of the range condition by LV_VLS_UP_RNG_CDN_R_LS[i].

If the range condition has been met, the gradient is in a particular range (VLS_UP_MMV_DRV1_THD_MIN[i] ≤ VLS_UP_DRV1_MMV[i] ≤ VLS_UP_MMV_DRV1_THD_MAX[i]) and the oxygen sensor signal is determined to be stable in the short term (VLS_UP_DRV1_ABS_MAX[i] ≤ C_VLS_UP_MMV_DRV1_ABS_MAX * FAC_T_VLS_UP_MMV_DRV1_ABS[i]) then resistance determination is requested (LV_R_IT_REQ_LS_UP[i] = 1). The factor FAC_T_VLS_UP_MMV_DRV1_ABS[i] shall be set to 1 unless C_CTR_AFL_CYC_MIN_RI_IT_LS_UP AF cycles have passed since the last lambda controller activation, as counted by CTR_AFL_CYC[i]. This ensures that T_AFL_CYC_HLD[i] is valid.

Once internal resistance determination has been requested, the Basic SW shall take a single VLS_UP[i] sample and store as VLS_UP_CUR_PUMP_OFF[i]. SW1 shall be switched ON, at time C_T_CUR_PUMP_ON_LS_UP[i] and the VLS_UP[i] shall be masked. A second oxygen sensor signal voltage sample shall be taken and stored as VLS_UP_CUR_PUMP_ON[i]. SW1 shall then be switched OFF and masking shall continue for a time C_T_DLY_CUR_PUMP_LS_UP, after which the VLS_UP[i] shall return to normal following of the oxygen sensor signal voltage. Masking shall be discontinued if the oxygen sensor signal voltage exceeds the VLS_UP_CUR_PUMP_ON[i] + C_VLS_HYS_R_IT_LS_UP value during the time C_T_DLY_CUR_PUMP_LS_UP after switching off SW1. Adding the voltage hysteresis shall prevent the function to discard lambda sensor voltage masking if the first VLS_UP[i] value after switching off the pump current is larger than the last VLS_UP[i] value while pumping due to signal noise.

The internal resistance R_IT_MES_LS_UP[i] shall be determined using the formula stated with the main variables being oxygen sensor voltage prior to pumping, VLS_UP_CUR_PUMP_OFF[i], and signal voltage during pumping, VLS_UP_CUR_PUMP_ON[i].

The internal resistance shall be calculated every time the oxygen sensor is pumped. For each new resistance value detected, a counter CTR_CYCNR_R_IT_LS_UP_VLD[i] shall be incremented to indicate to other functions that a new resistance value is available.

NOTE: During the time where SW1 is switched on, (LV_R_IT_REQ_LS_UP[i]=1) the output voltage of the oxygen sensor (VLS_UP[i]) does not represent the true A/F ratio and is therefore invalid and shall not be used as such. The basic software shall mask the response of the

sensor to pumping by keeping the voltage constant during this time by using the last VLS_UP[i] sample prior to activating SW1.

The internal resistance shall be calculated every time the oxygen sensor is pumped. For each new resistance value detected, the validity of the resistance shall be checked:

Should the validity function not be inhibited (LC_R_IT_VLD_INH_LS_UP = 0), then the tip temperature of the ceramic shall be determined from a map. TTIP_MES_LS_UP[i] represents the nominal temperature, TTIP_MES_TOL_LS_UP[i] represents the maximum temperature (R_IT_MES_LS_UP[i] * (1 – C_FAC_R_IT_ERR_NEG_LS_UP)), i.e. temperature for resistance with negative measurement error and TTIP_MES_BOL_LS_UP[i] represents the minimum temperature (R_IT_MES_LS_UP[i] * (1 + C_FAC_R_IT_ERR_NEG_LS_UP)), i.e. temperature for resistance with positive measurement error. A low pass filter shall then be applied to the upper and lower temperature limits to represent the maximum possible temperature gradient between resistance samples. Dependent on the direction of change of resistance between two measurements, either constant C_CRLC_TTIP_RISE_R_IT_LS_UP (rising temp.) or C_CRLC_TTIP_FALL_R_IT_LS_UP (falling temp.) shall be applicable. The final filter constant used, limited to a maximum of 1, shall be the aforementioned constant multiplied by the time between resistance samples (T_SAMPLE_R_IT_LS_UP[i]). This shall be to compensate for the irregular sample time.

Should then the ceramic tip temperature fall within the temperature limit bounds, the measured resistance shall be considered to be valid, shall be passed to the output variable R_IT_LS_UP[i], the counter CTR_CYCNR_R_IT_LS_UP_VLD[i] shall be incremented to indicate to other functions that a new resistance value is available and the T_SAMPLE_R_IT_LS_UP[i] shall be reset. If the above temperature condition is not met, R_IT_LS_UP[i] shall be maintained at its last value and the filtered temperature limits shall be reset to their previous values, i.e. the resistance value shall be ignored.

Should the validity function be inhibited then for each new resistance the R_IT_MES_LS_UP[i] shall be passed to R_IT_LS_UP[i] and the counter CTR_CYCNR_R_IT_LS_UP_VLD[i] shall be incremented.

In the case of significant oxygen sensor amplitude (LV_VLS_UP_MMV_LIM = 0) and the sensor signal be in the rich phase (VLS_UP[i] > VLS_UP_TOL[i]), the flags LV_VLS_UP_RNG_CDN_R_LS[i] & LV_VLS_UP_DRV1_CDN_R_LS[i] shall be reset. Furthermore, if these conditions have been met and the timer *TIMER_1* exceeds C_T_CUR_PUMP_OFF_LS_UP then LV_R_IT_DET_LS_UP[i] shall be reset, thereby permitting the internal resistance to be determined in the next cycle should the appropriate conditions be met.

For reference reasons of OBD II diagnosis the value R_IT_MDL_LS_UP_NEW[i] is calculated out of heating power, exhaust gas temperature and MAF_KGH.

Application conditions:

Recurrence: *T_SAMPLE_1 = 10 ms*

T_SAMPLE_2 = 1s

The function shall be carried out once every 10 ms. The internal resistance value shall only be calculated after “pumping” the oxygen sensor. See recurrence there. Reference internal rsistance shall be calculated with recurrence 1 s.

Initialisation:

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop (LV_ES).

R_IT_LS_UP[i] = 65535
R_IT_MES_LS_UP[i] = 65535
R_IT_MDL_LS_UP_NEW[i] = 65535
POW_MMV_LSH_UP[i] = 0
VLS_UP_CUR_PUMP_ON[i] = 0
VLS_UP_CUR_PUMP_OFF[i] = 0
CTR_CYCNR_R_IT_LS_UP_VLD[i] = 0
LV_VLS_UP_DRV1_CDN_R_LS[i] = 0
LV_VLS_UP_RNG_CDN_R_LS[i] = 0
LV_R_IT_REQ_LS_UP[i] = 0
LV_R_IT_DET_LS_UP[i] = 0
VLS_UP_MMV_DRV1_THD_MIN[i] = 0
VLS_UP_MMV_DRV1_THD_MAX[i] = 0
FAC_T_VLS_UP_MMV_DRV1[i] = 0
FAC_T_VLS_UP_MMV_DRV1_ABS[i] = 0
TTIP_REF_MDL_MMV_LS_UP[i] = 0
TTIP_MES_LS_UP[i] = 0
TTIP_MES_TOL_LS_UP[i] = 0
TTIP_MES_BOL_LS_UP[i] = 0
TTIP_MES_TOL_MMV_LS_UP[i] = 0
TTIP_MES_BOL_MMV_LS_UP[i] = 0
T_SAMPLE_R_IT_LS_UP[i] = 655.35
T_SAMPLE_R_IT_VLD_LS_UP[i] = 655.35
LV_V_REF_VLD_R_IT_LS_UP[i] = 0
LV_TTIP_MES_VLD_LS_UP[i] = 0

NOTE: All "N-1" variables shall also be initialised to the associated value specified above.

Activation / Deactivation:

If LV_ST_END = 1
then ***Function activated***
else ***Function deactivated***
endif

Formula section:

If LV_INH_R_IT_LS_UP[i] = 0
and LV_PUC = 0
and VLS_UP_MMV_MIN[i] < VLS_UP_MMV_MAX[i]
and MAF_KGH < C_MAF_KGH_MAX_R_IT_LS_UP
and STATE_LSH_UP[i] = "LSH_POW_CTL"
and (LV_ERR_LSH_UP[i] = 0
or TEG_DYN > C_TEMP_CUR_PUMP_ON_LS_UP)
then ***Update*** *TIMER_1*
Increment T_SAMPLE_R_IT_LS_UP[i]
If LV_R_IT_REQ_LS_UP[i] = 0
then **If** LV_VLS_UP_MMV_LIM[i] = 0
then **If** VLS_UP[i] < VLS_UP_BOL[i]

then **If** LV_R_IT_DET_LS_UP[i] = 0
then "*Determine lean cycle time dependent factors"*
"*Compute variable gradient thresholds"*
"Limit lower variable gradient threshold"
"Determine VLS_UP at low gradient condition met"
If LV_VLS_UP_RNG_CDN_R_LS[i] = 1
and VLS_UP_DRV1_MMV[i] ≥ VLS_UP_MMV_DRV1_THD_MIN[i]
and VLS_UP_DRV1_MMV[i] ≤ VLS_UP_MMV_DRV1_THD_MAX[i]
and VLS_UP_DRV1_ABS_MAX[i] ≤ C_VLS_UP_MMV_DRV1_ABS_MAX * FAC_T_VLS_UP_MMV_DRV1_ABS[i])
then LV_R_IT_REQ_LS_UP[i] = 1
VLS_UP_CUR_PUMP_OFF[i] = VLS_UP[i]
Switch SW1 ON
"Determine internal resistance" started
endif
endif
else **If** VLS_UP[i] > VLS_UP_TOL[i]
then LV_VLS_UP_RNG_CDN_R_LS[i] = 0
LV_VLS_UP_DRV1_CDN_R_LS[i] = 0
If *TIMER_1* ≥ C_T_CUR_PUMP_OFF_LS_UP
then LV_R_IT_DET_LS_UP[i] = 0
endif
endif
endif
else **If** (*TIMER_1* ≥ C_T_CUR_PUMP_OFF_LS_UP)
then LV_R_IT_REQ_LS_UP[i] = 1
VLS_UP_CUR_PUMP_OFF[i] = VLS_UP[i]
Switch SW1 ON
"Determine internal resistance" started
endif
endif
endif

If LV_R_IT_REQ_LS_UP[i] = 1 -> 0

then "*Calculation of Internal resistance*"

LV_R_IT_DET_LS_UP[i] = 1

Reset *TIMER_1*

endif

else ***Reset*** *TIMER_1*

T_SAMPLE_R_IT_LS_UP[i] = 655.35 (i.e. FFFFH)

LV_R_IT_DET_LS_UP[i] = 0

LV_VLS_UP_DRV1_CDN_R_LS[i] = 0

LV_VLS_UP_RNG_CDN_R_LS[i] = 0

endif

"Calculation of internal resistance reference"

"Calculation of internal resistance reference valid flag"

NOTES:

3. ***TIMER_1 refers to a SW internal timer and not a visible output data variable.***
4. ***T_SAMPLE_R_LS_UP[i] shall not be permitted to overflow. When the specified maximum value has been reached, this shall be held until the timer is reset by the function.***

Calculation of internal resistance reference:

% This calculation is carried out with the recurrence: T_SAMPLE_2 = 1 s.

POW_MMV_LSH_UP[i]$_n$ = POW_MMV_LSH_UP[i]$_{n-1}$ *

(1 - C_CRLC_POW_MMV_LSH_UP) + C_CRLC_POW_MMV_LSH_UP *

V_EFC_LSH_UP[i]^2 * T_SAMPLE_2 * [W/(s*V$^{2)}$]

TTIP_REF_MDL_MMV_LS_UP[i]$_n$ = TTIP_REF_MDL_MMV_LS_UP[i]$_{n-1}$ *

(1 - IP_CRLC_TTIP_REF_MDL_LS_UP) +

IP_CRLC_TTIP_REF_MDL_LS_UP * TEG_CAT_UP_MDL[i]

% IP (MAF_KGH)

R_IT_MDL_LS_UP_NEW[i] = IP_R_IT_MDL_LS_UP_NEW

% IP (POW_MMV_LSH_UP[i]; TTIP_REF_MDL_MMV_LS_UP[i])

Calculation of internal resistance reference valid flag:

% This calculation is carried out with the recurrence: T_SAMPLE_1

If CTR_CYCNR_R_IT_LS_UP_VLD[i]$_N$ = CTR_CYCNR_R_IT_LS_UP_VLD[i]$_{N-1}$

then **If** T_SAMPLE_R_IT_VLD_LS_UP[i] < C_T_SAMPLE_R_IT_VLD_LS_UP

then LV_TTIP_MES_VLD_LS_UP[i] = 1

T_SAMPLE_R_IT_VLD_LS_UP[i] =

T_SAMPLE_R_IT_VLD_LS_UP[i] + T_SAMPLE_1

else LV_TTIP_MES_VLD_LS_UP[i] = 0

endif

else LV_TTIP_MES_VLD_LS_UP[i] = 1

T_SAMPLE_R_IT_VLD_LS_UP[i] = 0

endif

Determine lean cycle time dependent factors:

If (CTR_AFL_CYC[i] ≥ C_CTR_AFL_CYC_MIN_RI_IT_LS_UP)

then FAC_T_VLS_UP_MMV_DRV1[i] = IP_FAC_T_DRV1_VLS_UP_MMV (T_AFL_CYC_HLD[i])

FAC_T_VLS_UP_MMV_DRV1_ABS[i] = IP_FAC_T_VLS_UP_MMV_DRV1_ABS (T_AFL_CYC_HLD[i])

else FAC_T_VLS_UP_MMV_DRV1[i] = 1
FAC_T_VLS_UP_MMV_DRV1_ABS[i] = 1

endif.

Compute variable gradient thresholds:

VLS_UP_MMV_DRV1_THD_MAX[i] = C_VLS_UP_MMV_DRV1_THD_MAX * C_FAC_VLS_UP_MMV_DRV1_THD * (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i])

VLS_UP_MMV_DRV1_THD_MIN[i] = C_VLS_UP_MMV_DRV1_THD_MIN * C_FAC_VLS_UP_MMV_DRV1_THD * (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) * FAC_T_VLS_UP_MMV_DRV1[i]

Limit lower variable gradient threshold:

VLS_UP_MMV_DRV1_THD_MIN[i] = MAX(VLS_UP_MMV_DRV1_THD_MIN[i], VLS_UP_DRV1_MMV_MIN[i] * C_FAC_VLS_UP_MMV_DRV1_THD_MIN)

Determine VLS_UP at low gradient condition met:

If (LV_VLS_UP_DRV1_CDN_R_LS[i] = 0) **&**
(VLS_UP_DRV1_MMV[i] ≥ VLS_UP_MMV_DRV1_THD_MIN[i])

then LV_VLS_UP_DRV1_CDN_R_LS[i] = 1

If (VLS_UP[i] ≤
(VLS_UP_MMV_MIN[i] + C_VLS_UP_MMV_MIN_RNG_TOL_R_IT)) **&**
(VLS_UP[i] ≥
(VLS_UP_MMV_MIN[i] - C_VLS_UP_MMV_MIN_RNG_BOL_R_IT))

then LV_VLS_UP_RNG_CDN_R_LS[i] = 1

else LV_VLS_UP_RNG_CDN_R_LS[i] = 0

endif

endif.

NOTE: VLS_UP_MMV_MIN[i] + C_VLS_UP_MMV_MIN_RNG_TOL_R_IT shall not cause an overflow and VLS_UP_MMV_MIN[i] - C_VLS_UP_MMV_MIN_RNG_BOL_R_IT shall not cause an underflow.

Determine internal resistance:

Recurrence: T_SAMPLE_3 = 1 ms

If LV_R_IT_REQ_LS_UP[i] = 1

then

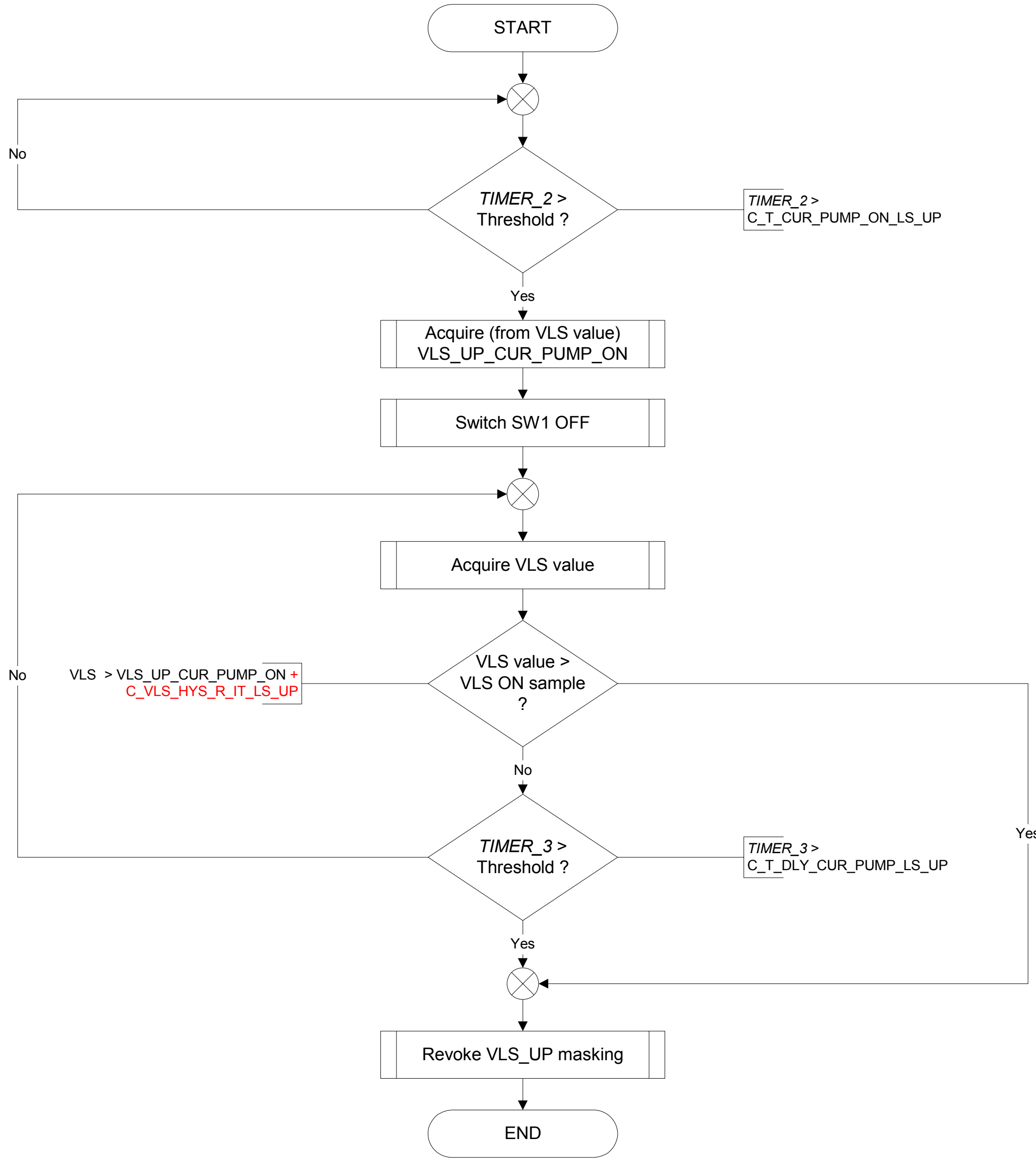


Figure 3 Function flow chart, Internal resistance determination

LV_R_IT_REQ_LS_UP[i] = 0

endif

Calculation of Internal resistance:

The internal resistance shall be determined as follows:

R_IT_MES_LS_UP[i] = NC_R_REF_LS_UP * (VLS_UP_CUR_PUMP_ON[i] -

VLS_UP_CUR_PUMP_OFF[i]) / {NC_VLS_UP_CUR_PUMP_REF -
(VLS_UP_CUR_PUMP_ON[i] - VLS_UP_CUR_PUMP_OFF[i])*
NC_FAC_R_REF_LS_UP - VLS_UP_CUR_PUMP_ON[i]}

Determine internal resistance validity:

The following function shall be carried once out for every new resistance value determined.

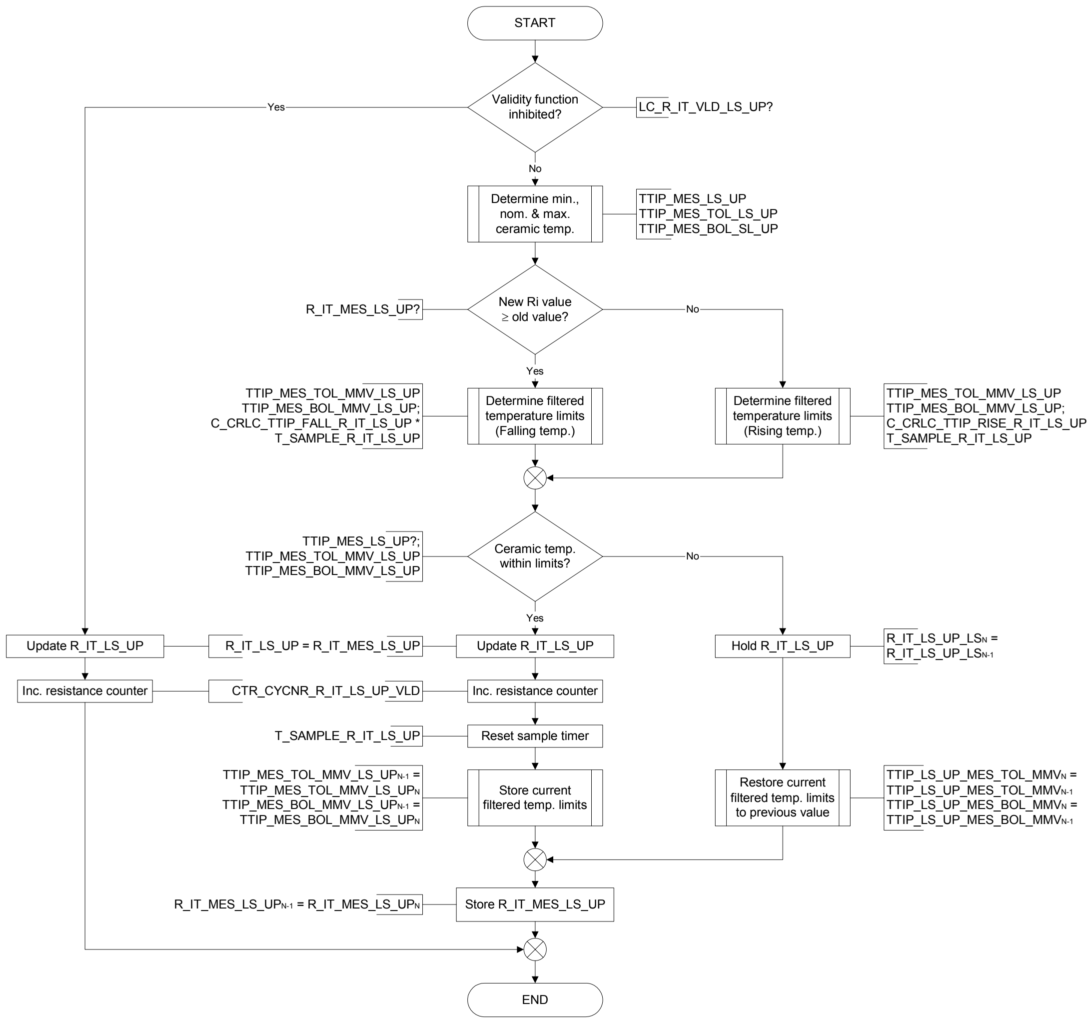


Figure 4 Function flow chart, Internal resistance validity determination

If (LC_R_IT_VLD_INH_LS_UP = 0)

then

TTIP_MES_LS_UP[i] = IP_TTIP_MES_LS_UP (R_IT_MES_LS_UP[i])
TTIP_MES_TOL_LS_UP[i] = IP_TTIP_MES_LS_UP (R_IT_MES_LS_UP[i] *
(1 - C_FAC_R_IT_ERR_NEG_LS_UP))
TTIP_MES_BOL_LS_UP[i] = IP_TTIP_MES_LS_UP (R_IT_MES_LS_UP[i] *
(1+ C_FAC_R_IT_ERR_POS_LS_UP))

If ($R\_IT\_MES\_LS\_UP[i]_N \geq R\_IT\_MES\_LS\_UP[i]_{N-1}$)

then

TTIP_MES_TOL_MMV_LS_UP[i]$_N$ = TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$ +
(TTIP_MES_TOL_LS_UP[i] - TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_FALL_R_IT_LS_UP * T_SAMPLE_R_IT_LS_UP[i])

TTIP_MES_BOL_MMV_LS_UP[i]$_N$ = TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$ +
(TTIP_MES_BOL_LS_UP[i] - TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_FALL_R_IT_LS_UP * T_SAMPLE_R_IT_LS_UP[i])

else

TTIP_MES_TOL_MMV_LS_UP[i]$_N$ = TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$ +
(TTIP_MES_TOL_LS_UP[i] - TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_RISE_R_IT_LS_UP * T_SAMPLE_R_IT_LS_UP[i])

TTIP_MES_BOL_MMV_LS_UP[i]$_N$ = TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$ +
(TTIP_MES_BOL_LS_UP[i] - TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$) *
MIN(1, C_CRLC_TTIP_RISE_R_IT_LS_UP * T_SAMPLE_R_IT_LS_UP[i])

endif

If ((TTIP_MES_LS_UP[i] ≤ TTIP_MES_TOL_MMV_LS_UP[i]) **&**
(TTIP_MES_LS_UP[i] ≥ TTIP_MES_BOL_MMV_LS_UP[i]))

then

R_IT_LS_UP[i] = R_IT_MES_LS_UP[i]$_N$
Increment CTR_CYCNR_R_IT_LS_UP_VLD[i]

LV_V_REF_VLD_R_IT_LS_UP[i] = 1
Reset T_SAMPLE_R_IT_LS_UP[i]

TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$ = TTIP_MES_TOL_MMV_LS_UP[i]$_N$
TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$ = TTIP_MES_BOL_MMV_LS_UP[i]$_N$

else

R_IT_LS_UP[i]$_N$ = R_IT_LS_UP[i]$_{N-1}$
TTIP_MES_TOL_MMV_LS_UP[i]$_N$ = TTIP_MES_TOL_MMV_LS_UP[i]$_{N-1}$
TTIP_MES_BOL_MMV_LS_UP[i]$_N$ = TTIP_MES_BOL_MMV_LS_UP[i]$_{N-1}$

endif

R_IT_MES_LS_UP[i]$_{N-1}$ = R_IT_MES_LS_UP[i]$_N$

else

R_IT_LS_UP[i] = R_IT_MES_LS_UP[i]$_N$
Increment CTR_CYCNR_R_IT_LS_UP_VLD[i]

endif.

NOTES:

5. ***The difference (VLS_UP_CUR_PUMP_ON[i] - VLS_UP_CUR_PUMP_OFF[i]) shall be limited to 0 for negative values as the voltage during pumping should be greater than that without.***
6. ***If the incrementation would cause the counter CTR_CYCNR_R_IT_LS_UP_VLD[i] to overflow then the counter shall be reset to 0.***
7. ***The basic software (BSW) shall mask the VLS_UP[i] signal during resistance determination by retaining the value of the last sample prior to activating the resistance determination. This shall be indicated by setting LV_R_IT_REQ_LS_UP[i]. Upon revoking the masking the last oxygen sensor signal voltage VLS value determined by the BSW shall be used for the VLS_UP[i] signal and LV_R_IT_REQ_LS_UP[i] shall be reset.***
8. ***Should C_T_CUR_PUMP_ON_LS_UP = 0, SW1 shall not be activated.***

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEMP_CUR_PUMP_ON_LS_UP	1	0...7FF0H	0...2047	0.0625	°C
Temperature threshold for the activation of sensor pumping					
C_T_CUR_PUMP_ON_LS_UP	1	0...08H	0...8e-3	1e-3	s
Duration for pump current active					
C_T_CUR_PUMP_OFF_LS_UP	1	0...FFFFH	0...655.35	10e-3	s
Minimum delay between consecutive pump current activation					
C_T_DLY_CUR_PUMP_LS_UP	1	0...FFH	0...255e-3	1e-3	s
Duration in which VLS_UP[NC_CBK_EX_NR] signal masked directly after Ri determination					
C_VLS_HYS_R_IT_LS_UP	1	0...3FFH	0...4.995117	4.8828e-3	V
Voltage hysteresis for evaluation if VLS_UP[i] shall be masked after Ri determination					
C_T_SAMPLE_R_IT_VLD_LS_UP	1	0...FFFFH	0...655.35	10e-3	s
Maximum time between internal resistance samples to reset valid flag					
C_VLS_UP_MMV_MIN_RNG_TOL_R_IT	1	0...FFFFH	0...4.999924	7.629e-5	V
Permitted range of VLS_UP[NC_CBK_EX_NR] above VLS_UP_MMV_MIN[NC_CBK_EX_NR] at low gradient for Ri determination					
C_VLS_UP_MMV_MIN_RNG_BOL_R_IT	1	0...FFFFH	0...4.999924	7.629e-5	V
Permitted range of VLS_UP[NC_CBK_EX_NR] below VLS_UP_MMV_MIN[NC_CBK_EX_NR] at low gradient for Ri determination					
C_VLS_UP_MMV_DRV1_THD_MIN	1	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Minimum threshold for mean 1st derivative of VLS_UP[NC_CBK_EX_NR] signal for Ri determination					
C_VLS_UP_MMV_DRV1_THD_MAX	1	8000...7FFFH	-2.5...2.4999	76.3e-6	V/10ms
Maximum threshold for mean 1st derivative of VLS_UP[NC_CBK_EX_NR] signal for Ri determination					
C_VLS_UP_MMV_DRV1_ABS_MAX	1	0...FFFFH	0...4.999665	7.629e-5	V/10ms
Threshold for absolute maximum 1st derivative of VLS_UP[NC_CBK_EX_NR] signal for Ri determination					
C_FAC_VLS_UP_MMV_DRV1_THD	1	0...FFH	0...1.992	7.81e-3	-
Multiplicative factor used to weight gradient thresholds dependent on VLS_UP_MMV_xxx[NC_CBK_EX_NR] amplitude					
C_FAC_VLS_UP_MMV_DRV1_THD_MIN	1	0...FFH	0...0.999600	3.92e-3	-
Hysteresis factor of DRV1_VLS_UP_MMV_MIN governing limiting of DRV1_VLS_UP_MMV_MIN_THD					
IP_FAC_T_DRV1_VLS_UP_MMV	1*8	0...FFH	0...63.75	0.25	-
LDPM_T_AFL_CYC_HLD_1_EGCP	8	0...FFH	0...2.55	10e-3	s
Multiplicative factor modifies DRV1_VLS_UP_MMV_MIN_THD dependent on T_VLS_CYC_AFL_HLD					
IP_CRLC_TTIP_REF_MDL_LS_UP	1*8	0...FFH	0...0.999600	3.92e-3	-
LDPM_MAF_KGH_1_EGCP	8	0...FFFFH	0...2047.9687	0.03125	kg/h
Correlation map for calculation of reference upstream tip temperature (modelled for new sensor)					
IP_R_IT_MDL_LS_UP_NEW	8*8	0...FFFFH	0...65535	1	Ω
LDP_POW_MMV_LSH_UP_IP_LS_UP	8	0...FFFFH	0...84.498711	1.2837E-3	W
LDP_TTIP_REF_MDL_MMV_LS_UP_R	8	0...7FFFH	0...2047.938	62.5e-3	°C
Internal resistance of a new upstream oxygen sensor (modelled)					
IP_FAC_T_VLS_UP_MMV_DRV1_ABS	1*8	0...FFH	0...15.9375	62.5e-3	-
LDPM_T_AFL_CYC_HLD_1_EGCP	8	0...FFH	0...2.55	10e-3	s
Multiplicative factor modifies C_VLS_UP_MMV_DRV1_ABS_MAX dependent on T_VLS_CYC_AFL_HLD					
C_CTR_AFL_CYC_MIN_RI_IT_LS_UP	1	0...FFFFH	0...65535	1	-
Min. no. AF cycles post lambda controller activation to permit use of T_VLS_CYC_AFL_HLD dependency					
C_MAF_KGH_MAX_R_IT_LS_UP	1	0...FFFFH	0...2047.96875	31.25e-3	kg/h
Maximum MAF_KGH threshold for activation of internal resistance determination					
C_FAC_R_IT_ERR_POS_LS_UP	1	0...FFH	0...0.999600	3.92e-3	-
Internal resistance measurement positive error (i.e. 0% to 100%)					
C_FAC_R_IT_ERR_NEG_LS_UP	1	0...FFH	0...0.999600	3.92e-3	-
Internal resistance measurement negative error (i.e. 100% to 0%)					
IP_TTIP_MES_LS_UP	1*8	0...7FFFH	0...2047.938	62.5e-3	°C
LDP_R_IT_MES_LS_UP_IP_R_LS_UP	8	0...FFFFH	0...65535	1	Ω
Oxygen sensor temperature vs. internal resistance characteristic					
C_CRLC_POW_MMV_LSH_UP	1	0...FFH	0...0.999600	3.92e-3	-

Correlation (filter) constant for moving mean value of upstream heater power					
C_CRLC_TTIP_FALL_R_IT_LS_UP	1	0...FFH	0...0.999600	3.92e-3	s^{-1}
Correlation (filter) constant for falling ceramic temperature, limit calculation					
C_CRLC_TTIP_RISE_R_IT_LS_UP	1	0...FFH	0...0.999600	3.92e-3	s^{-1}
Correlation (filter) constant for rising ceramic temperature, limit calculation					
LC_R_IT_VLD_INH_LS_UP	1	0...1H	0...1	1	-
Boolean flag inhibiting internal resistance determination validity function					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_R_REF_LS_UP	1	0...FFFFH	0...65535	1	Ω
Reference resistance for pump current (R_{Pump})					
NC_FAC_R_REF_LS_UP	1	0...FFFFH	0...3.9994	6.1035e-5	-
Pump current limiting to pull UP resistance ratio (R_{Pump}/R_2)					
NC_VLS_UP_CUR_PUMP_REF	1	0...3FFH	0... 4.995117	4.8828e-3	V
Reference voltage value (4.82 V)					

4.65.1 Application specific conditions for oxygen sensor internal resistance

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_R_IT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that upstream sensor internal resistance determination shall be inhibited when set					

Input data:

NC_CBK_EX_NR	LV_ES		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank.

The Boolean flag LV_INH_R_IT_LS_UP[i] shall indicate whether the upstream oxygen sensor internal resistance determination shall be permitted to take place. When set, no resistance determination shall take place.

Description:

No application specific conditions have been defined to date. Internal resistance determination shall always take place (LV_INH_R_IT_LS_UP[i] shall be initialised to and remain 0).

Application conditions:

Initialisation:

The following initialisation shall be carried out upon leaving the engine state Engine Stop (LV_ES).

LV_INH_R_LS_UP[i] = 0

Activation:

None defined at present.

Deactivation:

None defined at present.

Formula section:

None defined at present.

4.66 Clutch switch detection

4.66.1 Signal acquisition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CS	V/O	0...1H	0...1	1	-
Clutch switch					

Input data:

LV_AT	LV_IM_CS_PN		

Function description:

Initialisation:

at Reset:

IF LV_IM_CS_PN = 1 or
LV_AT = 1
THEN LV_CS = 0 (clutch pedal released)
ELSE LV_CS = 1 (clutch pedal actuated)

Recurrence: 10 ms

To detect if the clutch is depressed or not, LV_IM_CS_PN is evaluated.

Formula section:

IF LV_IM_CS_PN = 1 or
LV_AT = 1
THEN LV_CS = 0 (clutch pedal released)
ELSE LV_CS = 1 (clutch pedal actuated)

4.66.2 Clutch activation detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CS_ACT_DC	O/V	0...1	0...1	1	-
Logical Variable for detected clutch-pedal activation during present DC					

FUNCTION DESCRIPTION:

Function is used to detect if clutch pedal activation was recognized at least one time in present driving cycle.

Application conditions:

Initialisation: at reset: LV_CS_ACT_DC = 0

Recurrence: 100 ms

Activation: LV_IGK = 1

Formula section:

If LV_CS_ACT_DC = 0
Then **If** LV_CS 0->1
or
LV_CS 1->0
Then LV_CS_ACT_DC = 1
Endif
Endif

4.67 Oil Temperature TOIL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOIL	V/O	0...C8H	-40...160	1	°C
Oil temperature value after gradient limitation					
TOIL_STOP	V/O/S	0...C8H	-40...160	1	°C
Oil temperature at transition to engine stop (ES)					

Input data:

CONF_TOIL_MDL	LV_ERR_TOIL	TOIL_MDL_FSD	TOIL_MES
C_TOIL_MIN_DIAG	C_TOIL_MAX_DIAG	LV_ERR_TOIL_STUCK	LV_ERR_TOIL_PLAUS_L
LV_ERR_TOIL_PLAUS_H	C_TOIL_GRD_MAX	TOIL_MDL	

FUNCTION DESCRIPTION:

4.67.1 Oil temperature acquisition

General information:

The measured oil temperature TOIL_MES is monitored for implausible gradient by a gradient limitiation. The gradient limitation is not used as diagnostic function, it is only intended to fade-out implausible values (like noise).

CONF_TOIL_MDL shows if there is a manual request for using the TOIL model (from FSD) by switch C_CONF_TOIL_MDL = 1 or for using TOIL_MDL by switch C_CONF_TOIL_MDL = 2.

Application conditions:

Initialisation at Reset : TOIL = from calculation with TOIL_MES init value
(same calculation of TOIL like normal recurrence incl. range check,
as described in Formula Section)

Activation : At every engine operating state

The application recurrence is **1 sec.**

Formula section:

If (LV_ERR_TOIL = 0 *(No error at oil temperature signal)*
and LV_ERR_TOIL_STUCK = 0 *(No stuck Signal detected)*
and LV_ERR_TOIL_PLAUS_L = 0 *(No unplausible low value detected)*
and LV_ERR_TOIL_PLAUS_H = 0 *(No unplausible high value detected)*
and CONF_TOIL_MDL = 0) *(TOIL Sensor present)*
then
if TOIL_MES < C_TOIL_MIN_DIAG **or**
TOIL_MES > C_TOIL_MAX_DIAG
then $TOIL_n = TOIL_{n-1}$ *(keep previous value)*
else
if $TOIL\_MES_n < TOIL\_MES_{n-1} - C\_TOIL\_GRD\_MAX$
then $TOIL = TOIL\_MES_{n-1} - C\_TOIL\_GRD\_MAX$ *(limit TOIL by negative gradient)*
else
if $TOIL\_MES_n > TOIL\_MES_{n-1} + C\_TOIL\_GRD\_MAX$
then $TOIL = TOIL\_MES_{n-1} + C\_TOIL\_GRD\_MAX$
(limit TOIL by positive gradient)
else $TOIL = TOIL\_MES_n$ *(measured value from sensor)*
else
if CONF_TOIL_MDL = 2
then TOIL = TOIL_MDL *(calculated value)*
else TOIL = TOIL_MDL_FSD *(calculated value from Fuel System Diagnosis)*

4.67.2 Oil temperature at engine stop

General information:

The engine oil temperature at engine stop (TOIL_STOP) is determined at each transition to the engine operating state "Engine Stop". TOIL_STOP will be stored in the non-volatile memory durint the ECU'S self holding phase.

Application conditions:

Initialisation at Reset : TOIL_STOP = TOIL_STOP
(stored in the non-volatile memory)

Initialisation at Engine Run to Engine Stop (ERU_to_ES) :
TOIL_STOP = TOIL

Activation : At every engine operating state

4.68 Oil Temperature TOIL (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOIL_MIN_TOT_DC	V/O/S	0...C8H	-40...160	1	°C
Former / current driving cycle minimum TOIL					
TOIL_MAX_TOT_DC	V/O/S	0...C8H	-40...160	1	°C
Former / current driving cycle maximum TOIL					

Input data:

TOIL	LV_IGK	ERR_SYM_TOIL	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

TOIL_MIN_TOT_DC = C8H (160°C)

TOIL_MAX_TOT_DC = 0H (-40°C)

- otherwise: restored from non-volantile memory

Recurrence: 1s

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If ERR_SYM_TOIL = 0

Then If TOIL < TOIL_MIN_TOT_DC

Then TOIL_MIN_TOT_DC = TOIL

Endif

Endif

If ERR_SYM_TOIL = 0

Then If TOIL > TOIL_MAX_TOT_DC

Then TOIL_MAX_TOT_DC = TOIL

Endif

Endif

4.69 Oil temperatur model TOIL_MDL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOIL_MDL	V/O	0...C8H	-40...160	1	[°C]
TOIL calculated					
TOIL_MDL_RAW	V	0...C800H	-40...160	3.9063e-3	[°C]
TOIL Model calculated RAW-Value					
TOIL_MDL_GRD	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C/s]
Toil Model calculated gradient per second					
TOIL_MDL_1	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculation Step 1					
TOIL_MDL_2	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculation Step 2					
TOIL_MDL_3	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculation Step 3					
TOIL_MDL_4	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculation Step 4					
TOIL_MDL_OFS	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculated offset					
TOIL_MDL_0	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculation Step 0					
TOIL_MDL_H	V	0...C800H	-40...160	3.9063e-3	[°C]
TOIL_MDL calculated High-Part					
TOIL_MDL_L_GRD	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C/s]
TOIL_MDL temperature gradient					
TOIL_MDL_L_GRD_FIL	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C/s]
TOIL_MDL temperature gradient filtered					
TOIL_MDL_L	V	0...C8H	-40...160	1	[°C]
TOIL_MDL "Low Part"					
TOIL_MDL_MAX	V/O	0...C8H	-40...160	1	[°C]
Maximum Oil Temperature from the last engine running					
TOIL_MDL_OLD	V	0...C800H	-40...160	3.9063e-3	[°C]
TOIL_MDL (n-1)					
TOIL_MDL_CLC	V	0...C800H	-40...160	3.9063e-3	[°C]
TOIL_MDL calculated in high resolution					
TOIL_ST	V	0...C8H	-40...160	1	[°C]
TOIL started					
TOIL_OFS_ST	V	0...C8H	-40...160	1	[°C]
Additive offset at TOIL _ST calculation					
TOIL_MDL_OFS_COR_LOAD	V	8000...7FFFH	-128...127.99609	3.9063e-3	[°C]
TOIL_MDL calculated offset correction for load condition					

Input data:

N_32	TIA	MAF	TCO
VS	T_AST	TCO_ST	TCO_STOP
TAM	LV_ST_END	LV_IGK	LV_ERR_RD_AR_NVMY
TOIL_STOP	T_ES	LV_T_ES_NOT_PLAUS	LV_ES
LV_T_ES_RST	LV_PUC	T_PUC	

FUNCTION Description:

General Information:

The oiltemperature is calculated by a modell dependend on the surrounding and engine conditions regard to the thermo dynamic characteristics of the oil velocity.

Application condition :

Activation: LV_IGK = 1 **and** LV_ST_END = 1

Deactivation:: LV_IGK = 0 **or** LV_ST_END = 0

Initialisation: at 10ms after Reset

(Hint: It has to be ensured by SW, that the Initialisation of T_ES is done prior to TOIL_ST to get correct values of Engine off timer)

```
If      LV_ERR_RD_AR_NVMY = 0
        (TOIL_STOP properly stored after last engine stop)
   and  LV_T_ES_NOT_PLAUS = 0
   and  LV_T_ES_RST = 0
then    TOIL_OFS_ST = IP_TOIL_OFS_ST
   If      T_ES < IP_TOIL_MDL_ST
   Then    TOIL_ST =  TCO_ST + (TOIL_STOP – TCO_STOP) *
           IP_FAC_TOIL_ST
   else    TOIL_ST = TCO_ST + TOIL_OFS_ST
   endif
else    TOIL_ST = TCO_ST
endif
```

```
TOIL_MDL        = TOIL_ST
TOIL_MDL_RAW    = TOIL_ST
TOIL_MDL_OLD    = TOIL_ST
TOIL_MDL_MAX    = TOIL_ST
TOIL_MDL_CLC    = TOIL_ST
TOIL_MDL_GRD    = 0
TOIL_MDL_0      = 0
TOIL_MDL_1      = 0
TOIL_MDL_2      = 0
TOIL_MDL_3      = 0
TOIL_MDL_4      = 0
```

Recurrence: 1000 msec

Signal flow diagram:

30-NOV-99

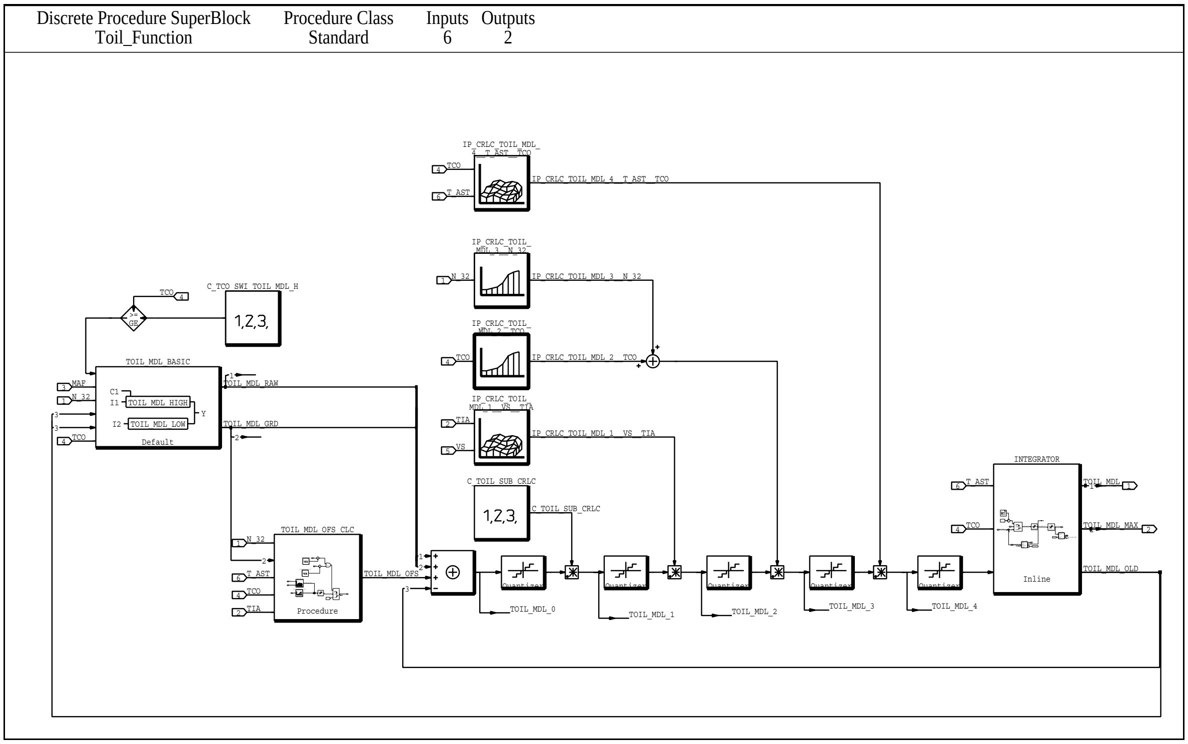


Figure 1: Toil_Function

Description:

TOIL_MDL Calculation:

The Oiltemperature calculation is splitted up in two „ways“. The "LOW-PART" based on the coolant temperature (TCO) and the TCO-gradients. In the high temperature range (TCO-temperature > IP_TCO_SWI_TOIL_MDL_H) the calculation of TOIL_MDL is done in the "HIGH-PART". This path based on the MAP (IP_TOIL_SUB__N32__MAF) dependend on the engine load and engine speed.

Both pathes of the TOIL calculation constitutes the BLOCK "TOIL_MDL_BASIC". The outputvalues are TOIL_MDL_RAW and TOIL_MDL_GRD. Both values are used for the multiplication with the weighting factors in order to get the calculated oil temperature (TOIL_MDL_CLC). The high resolution of TOIL_MDL_CLC (High-Byte) is transferred in the modeled oiltemperature TOIL_MDL (only the HIGH-Byte is used for TOIL_MDL with resolution 1°C).

The coolant temperature TCO is used for the initialization value for the "LOW-PART". The initialization for the toilmodel starting in the "HIGH-PART" is done at the first request of the function.

Formula section:

Calculation of TOIL_MDL_RAW, TOIL_MDL_GRD:

If (TCO >=IP_TCO_SWI_TOIL_MDL_H)

then TOIL_MDL_RAW =TOIL_MDL_H

TOIL_MDL_GRD = 0

else TOIL_MDL_RAW =TOIL_MDL_L

TOIL_MDL_GRD = TOIL_MDL_L_GRD

Endif

Calculation of TOIL_MDL:

TOIL_MDL_0 =		TOIL_MDL_RAW	(RAW-VALUE)
	+	TOIL_MDL_GRD	(Gradient)
	+	TOIL_MDL_OFS	(Offset)
	-	TOIL_MDL_OLD	(TOIL_MDLn-1)

TOIL_MDL_1 = TOIL_MDL_0 * C_TOIL_SUB_CRLC

TOIL_MDL_2 = TOIL_MDL_1 * IP_CRLC_TOIL_MDL_1__VS__TIA

TOIL_MDL_3 = TOIL_MDL_2 *(IP_CRLC_TOIL_MDL_2__TCO +
IP_CRLC_TOIL_MDL_3__N_32)

TOIL_MDL_4 = TOIL_MDL_3 *IP_CRLC_TOIL_MDL_4__T_AST__TCO

TOIL_MDL_CLC = TOIL_MDL_4 + TOIL_MDL_OLD

4.69.1 TOIL_MDL_HIGH

19-MAY-99

Discrete Procedure SuperBlock	Procedure Class	Inputs	Outputs
TOIL_MDL_HIGH	Standard	4	2

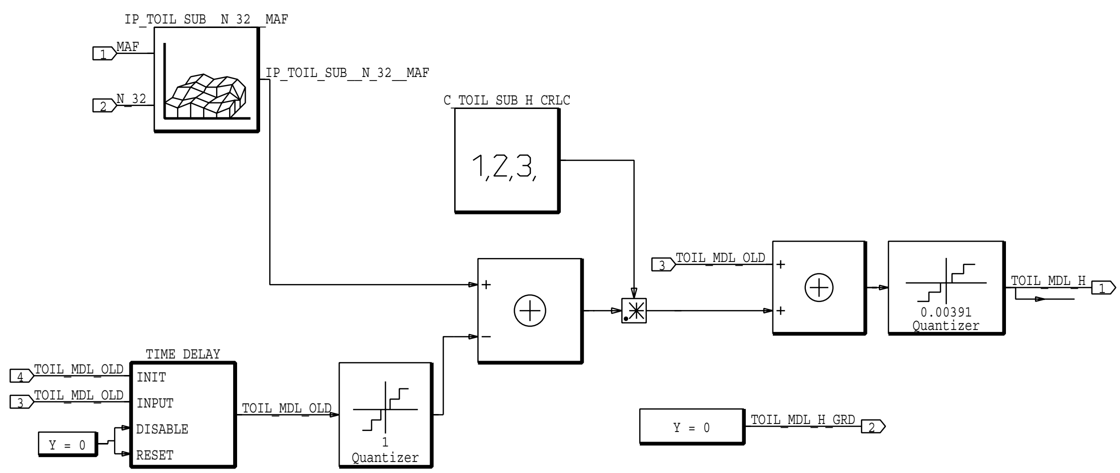


Figure 2: TOIL_MDL_HIGH

Description:

Calculation of TOIL_MDL_H

In the oiltemperature area the Modul TOIL_MDL_H is used for the calculation (TCO > IP_TCO_SWI_TOIL_MDL_H). In the high path of the calculation, no temperature gradients are used (TOIL_MDL_H_GRD =0). For the start during high temperatures, the model will be initialized by the first request of the function. The oiltemperature based on stationair temperature for steady states (engine speed and load N_32, MAF).

Initialization by first request of the function

TOIL_MDL_H = TOIL_MDL_CLC

Formula Section

** Initialisisation*

If TCO >= IP_TCO_SWI_TOIL_MDL_H

then TOIL_MDL_H = TOIL_MDL_CLC

Endif.

Calculation of TOIL_MDL_H, TOIL_MDL_MAX:

TOIL_MDL_Hn **=** TOIL_MDL_OLD **+** C_TOIL_SUB_H_CRLC
***** (IP_TOIL_SUB__N_32__MAF **–** TOIL_MDL_OLD)

TOIL_MDL_MAX **= max** (TOIL_MDL) for current driving cycle.

4.69.2 TOIL_MDL_LOW

19-MAY-99

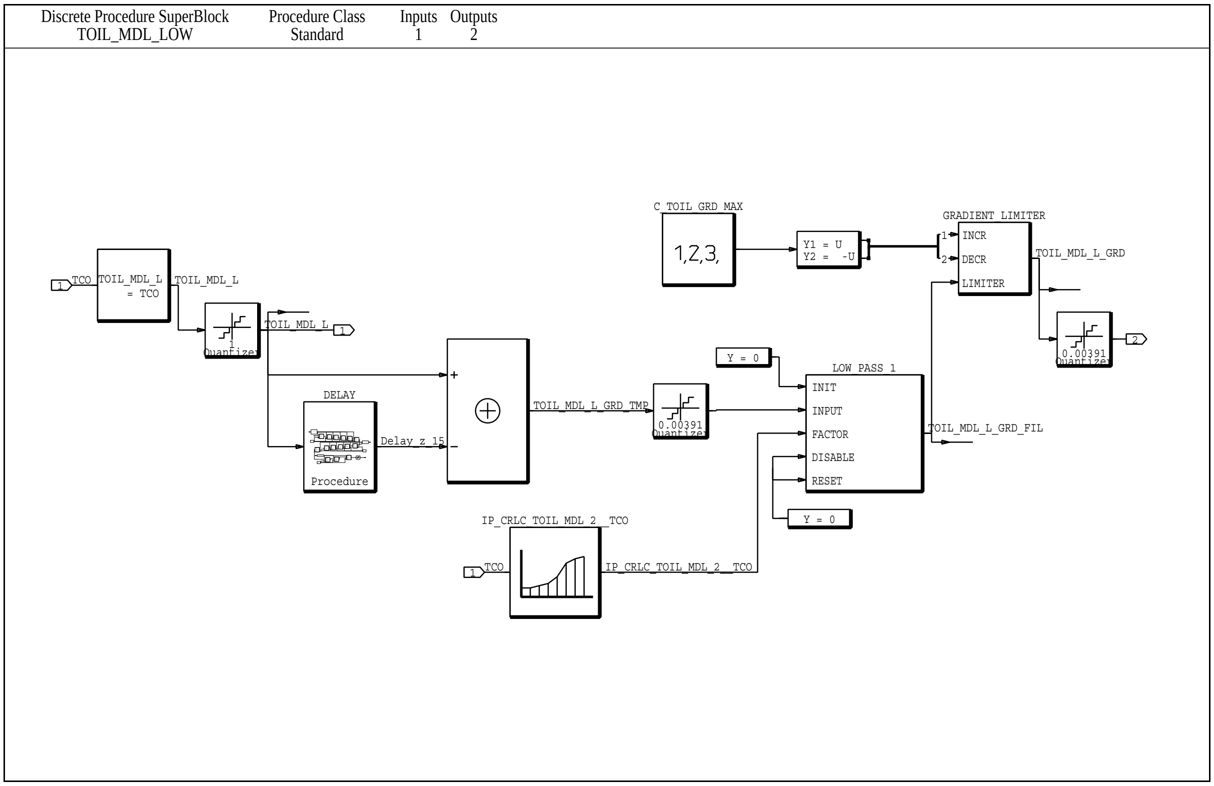


Figure 3: TOIL_MDL_LOW

Description:

Calculation of TOIL_MDL_L:

In the Oiltemperature range TCO < IP_TCO_SWI_TOIL_MDL_H the calculation will be done in the functionblock TOIL_MDL_L. The calculation based on the TCO behaviour in regard to the actuall engine state including weighting factors.

Initialization:

TOIL_MDL_L_GRD_FIL	= 0
TOIL_MDL_L_GRD	= 0
TOIL_MDL_L	= TCO
TOIL_MDL_(n-1)...TOIL_MDL_(n-15)	= TCO

Formula Section:

TOIL_MDL_L = TCO

TOIL_MDL_L_GRD_TMP = TOIL_MDL_L - TOIL_MDL_L(n-15)

TOIL_MDL_L_GRD_FIL = TOIL_MDL_L_GRD_FIL(n-1)
+ IP_CRLC_TOIL_MDL_2__TCO *
(TOIL_MDL_L_GRD_TMP - TOIL_MDL_L_GRD_FIL(n-1))

If TOIL_MDL_L_GRD_FIL >= C_TOIL_GRD_MAX

then TOIL_MDL_L_GRD = C_TOIL_GRD_MAX

else

if TOIL_MDL_L_GRD_FIL <= -C_TOIL_GRD_MAX

then TOIL_MDL_L_GRD = - C_TOIL_GRD_MAX

else

TOIL_MDL_L_GRD = TOIL_MDL_L_GRD_FIL

endif.

4.69.2.1 DELAY

19-MAY-99

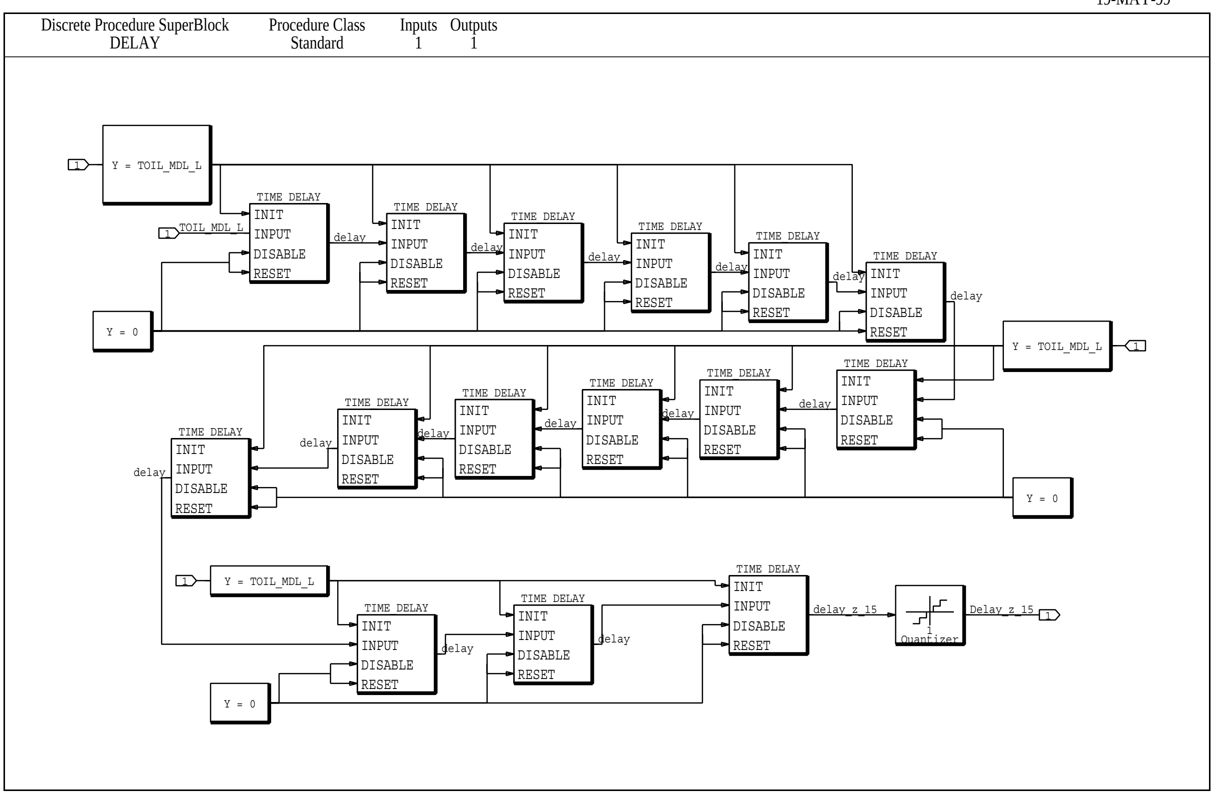


Figure 4: DELAY

Description:

The systemblock DELAY is a signaldelay of 15th order (z^{-15} = 15sec). The systemblock DELAY is initialized with TOIL_MDL_L.

4.69.3 Calculation of TOIL_MDL_OFS

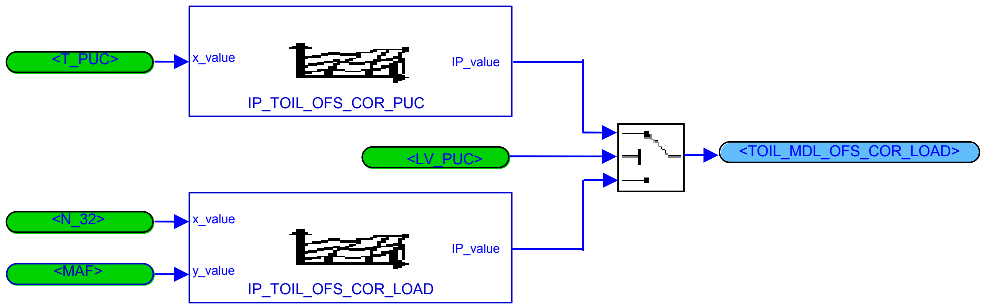


Figure 5: TOIL_MDL_OFS_COR_LOAD

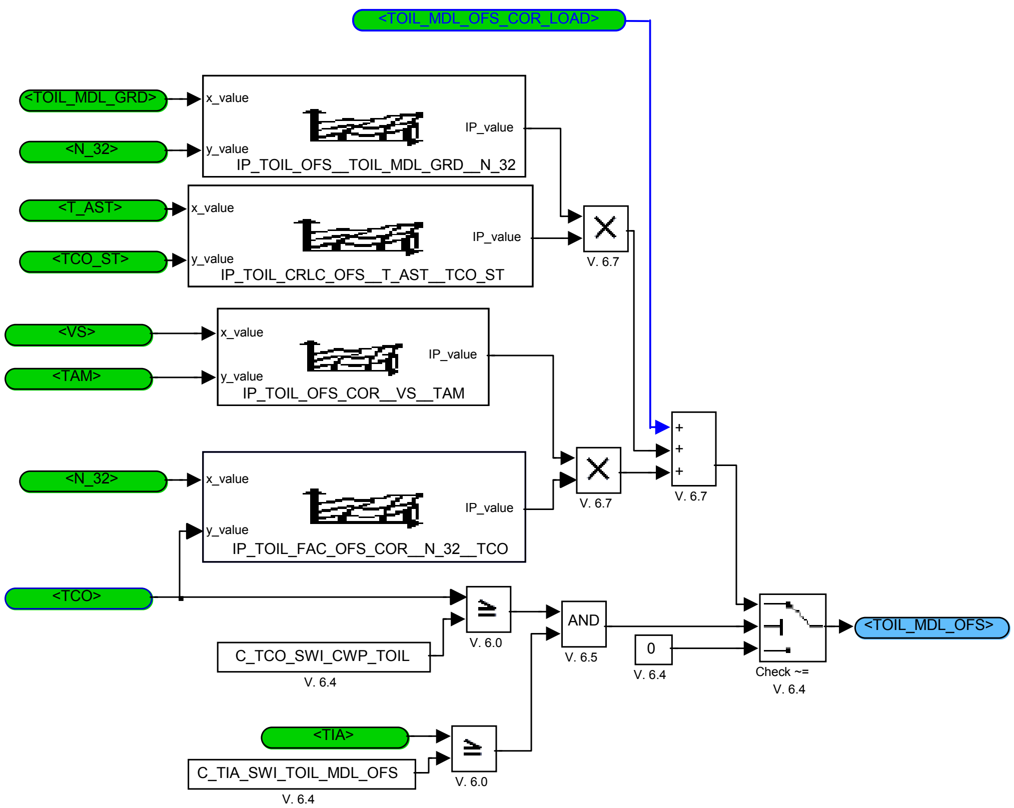


Figure 6: TOIL_MDL_OFS_CLC

Calculation of TOIL_MDL_OFS_COR_LOAD:

If LV_PUC = 1

Then TOIL_MDL_OFS_COR_LOAD = IP_TOIL_OFS_COR_PUC

Else TOIL_MDL_OFS_COR_LOAD = IP_TOIL_OFS_COR_LOAD

Endif

Calculation of TOIL_MDL_OFS:

If **(**(TCO >= C_TCO_SWI_CWP_TOIL) **and** (TIA >= C_TIA_SWI_TOIL_MDL_OFS)**)**

then

TOIL_MDL_OFS = IP_TOIL_OFS__TOIL_MDL_GRD__N_32 *
IP_TOIL_CRLC_OFS__T_AST__TCO_ST +
IP_TOIL_OFS_COR__VS__TAM *
IP_TOIL_FAC_OFS_COR__N_32__TCO +
TOIL_MDL_OFS_COR_LOAD

else TOIL_MDL_OFS = 0

endif

4.69.4 INTEGRATOR

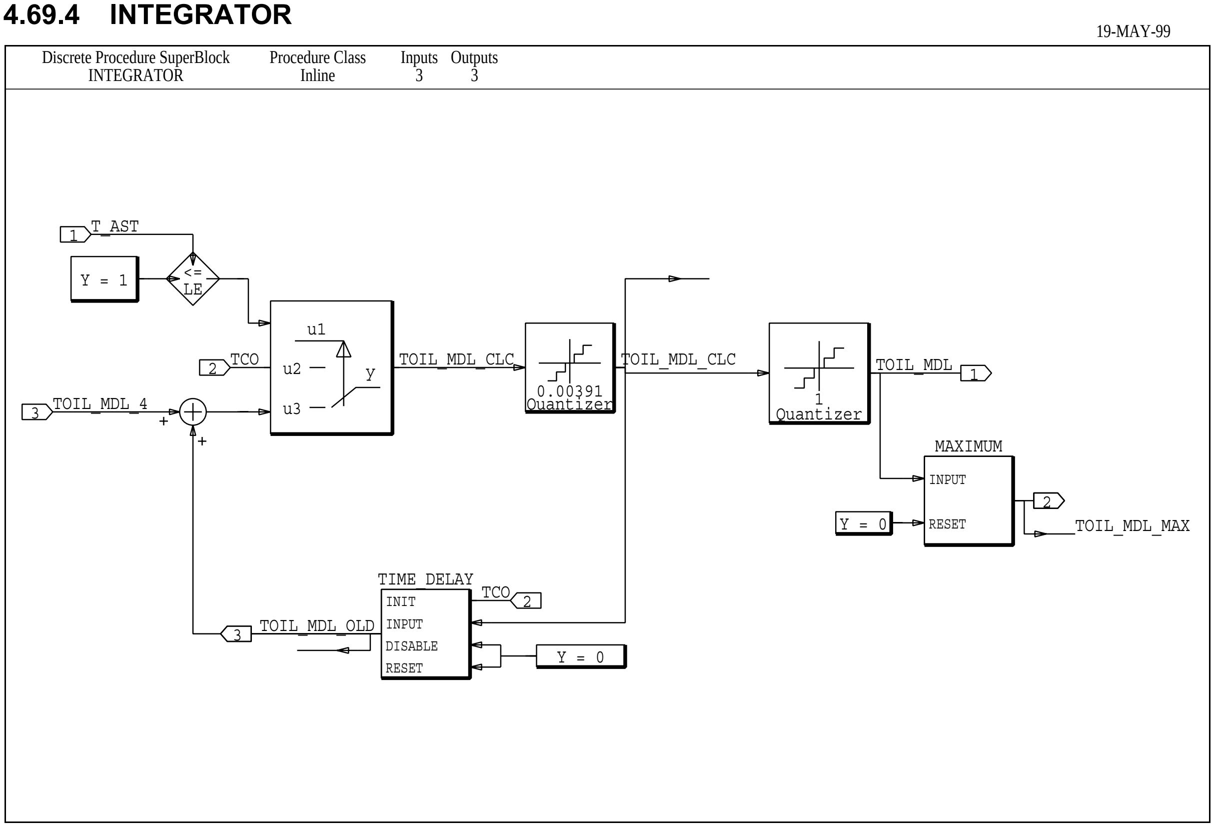


Figure 7: INTEGRATOR

Description:

The systemblock „Integrator“ is the last step of the oil temperature calculation. For internal oil temperature caluclation the high resolution of TOIL_MDL is used. The output value TOIL_MDL is the acutal calculated oil temperature with the resolution of 1°C. The calculated value TOIL_MDL_OLD is the calculated oil temperature 1 sec delayed. In case of ECU-RESET, the oil temperatures TOIL_MDL, TOIL_MDL_OLD; TOIL_MDL_MAX will be initialized with the coolant temperature TCO.

Formula Section:

Calculation of TOIL MDL_MAX:

TOIL_MDL_MAX = max (TOIL_MDL)   for current driving cycle

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Auflös.	Einh.
C_TOIL_SUB_CRLC	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Correlation constant for TOIL_MDL calculation					
C_TCO_SWI_CWP_TOIL	1	0...FEH	-48...142.5	0.75	[°C]
Switch for TOIL_MDL_OFS calculation of TOIL_MDL					
C_TIA_SWI_TOIL_MDL_OFS	1	0...FEH	-48...142.5	0.75	[°C]
TIA Condition Switch for TOIL_MDL_OFS calculation of TOIL_MDL					
IP_TCO_SWI_TOIL_MDL_H	6	0...FEH	-48...142.5	0.75	[°C]
LDP_N_32_IP_TCO_SWI_TOIL_MDL_H	6	0...FFH	0...8160	32	[rpm]
Switch for TOIL_MDL calculation High-Part					
C_TOIL_SUB_H_CRLC	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
correlation constant for TOIL-calculation					
C_TOIL_GRD_MAX	1	0...C8H	0...200	1	[°C/s]
max. Temperature gradient for TOIL-calculation					
IP_TOIL_CRLC_OFS	6*6	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_T_AST_IP_TOIL_CRLC_OFS	6	0...FFFFH	0...6553.5	0.1	[s]
LDP_TCO_ST_IP_TOIL_CRLC_OFS	6	0...FEH	-48...142.5	0.75	[°C]
Correlation constant for TOIL_MDL_OFS					
IP_TOIL_OFS	9*4	0...FFFFH	-128...127.99609	3.9063e-3	[°C]
LDP_TOIL_MDL_GRD_IP_TOIL_OFS	9	0...FFFFH	-128...127.99609	3.9063e-3	[°C]
LDPM_N_32_IP_TOIL_SUB	4	0...FFH	0...8160	32	[rpm]
additive offset for TOIL_MDL calculation					
IP_TOIL_OFS_COR	8*8	0...FFFFH	-128...127.99609	3.9063e-3	[°C]
LDP_VS_IP_TOIL_OFS_COR	8	0...FFH	0...255	1	[km/h]
LDP_TAM_IP_TOIL_OFS_COR	8	0...FEH	-48...142.5	0.75	[°C]
Corrective offset for TOIL_MDL calculation					
IP_TOIL_FAC_OFS_COR_	8*6	0...FFFFH	-1...0.99998	0.0306e-3	[-]
LDP_N_32_IP_FAC_OFS_COR	8	0...FFH	0...8160	32	[rpm]
LDP_TCO_IP_FAC_OFS_COR	6	0...FEH	-48...142.5	0.75	[°C]
correlation factor for corrective Offset					
IP_CRLC_TOIL_MDL_3	4	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDPM_N_32_IP_TOIL_SUB	4	0...FFH	0...8160	32	[rpm]
correlation factor for TOIL_MDL calculation					
IP_CRLC_TOIL_MDL_2	9	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_TCO_IP_CRLC_TOIL_MDL_2	9	0...FEH	-48...142.5	0.75	[°C]
correlation factor for TOIL_MDL calculation based on TCO					
IP_CRLC_TOIL_MDL_1	6*6	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_VS_IP_CRLC_TOIL_MDL_1	6	0...FFH	0...255	1	[km/h]
LDP_TIA_IP_CRLC_TOIL_MDL_1	6	0...FEH	-48...142.5	0.75	[°C]
correlation factor for TOIL_MDL calculation					
IP_CRLC_TOIL_MDL_4	6*4	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_T_AST_IP_CRLC_TOIL_MDL_4	6	0...FFFFH	0...6553.5	0.1	[s]
LDP_TCO_IP_CRLC_TOIL_MDL_4	4	0...FEH	-48...142.5	0.75	[°C]
Correlation factor for TOIL_MDL calculation					
IP_TOIL_SUB	4*4	0...C8H	-40...160	1	[°C]
LDPM_N_32_IP_TOIL_SUB	4	0...FFH	0...8160	32	[rpm]
LDP_MAF_IP_TOIL_SUB	4	0...FFH	0...1389	5.4470588	[mg/stk]
Basic temperature value for TOIL calculation at high TCO					
IP_FAC_TOIL_ST	8*8	0...FFH	-2...1.999939	0.015625	-
LDP_T_ES_IP_FAC_TOIL_ST	8	0...FFFFH	0...65535	1	[min]
LDP_TOIL_STOP_IP_FAC_TOIL_ST	8	0...C8H	-40...160	1	[°C]
Weithing Factor for TOIL_ST Temperature depending on Engine off time					
IP_TOIL_OFS_ST	8*8	0...C8H	-40...160	1	[°C]
LDP_T_ES_IP_TOIL_ST_OFS	8	0...FFFFH	0...65535	1	[min]
LDP_TCO_ST_IP_TOIL_ST_OFS	8	0...FEH	-48...142.5	0.75	[°C]
Additive offset TOIL for initialization depending on T_ES					
IP_TOIL_MDL_ST	8*8	0...FFFFH	0...65535	1	[min]
LDP_TCO_STOP_IP_TOIL_MDL_ST	8	0...FEH	-48...142.5	0.75	[°C]
LDP_TOIL_STOP_IP_TOIL_MDL_ST	8	0...C8H	-40...160	1	[°C]

Data to decide function which to be used for TOIL_ST					
IP_TOIL_OFS_COR_LOAD	8*8	0...FFFFH	-128...127.99609	3.9063e-3	[°C]
LDP_N_32_IP_TOIL_OFS_COR_LOAD	8	0...FFH	0...8160	32	[rpm]
LDP_MAF_IP_TOIL_OFS_COR_LOAD	8	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Corrective TOIL_MDL offset for high load driving					
IP_TOIL_OFS_COR_PUC	6	0...FFFFH	-128...127.99609	3.9063e-3	[°C]
LDP_T_PUC_IP_TOIL_OFS_COR_PUC	6	0...FFFFH	0...655.35	0.01	[sec]
Corrective TOIL_MDL offset for PUC					

4.70 Power Steering acquisition variable

(This functional will be activated only if CONF_PSTE = 1.)

4.70.1 Power steering Pressure Voltage Signal

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_PSP_MES	V/O	0...3FFH	0...4,995	4.88e-3	V
Power steering Pressure Voltage Signal (before electrical Diagnosis)					

Input data:

PSP_BAS			

FUNCTION DESCRIPTION:

General information:

The raw value for power steering pressure voltage signal (PSP_BAS) is measured by continuous conversion (10 bits) every 10 msec.

The value of the numeric conversion is adapted to take into account of the electronic component drift. The result of this adaptation must be linearized according to the pressure sensor response.

The corresponding value of the first measurement after hardware reset is used for initialization.

Formula section:

The voltage from the power steering pressure voltage signal is converted to the measured power steering pressure V_PSP_MES.

$$V\_PSP\_MES = (1/10) * \sum_{n=1}^{n=10} PSP\_BAS_n$$

4.70.2 Power steering pressure PSP

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PSP	V/O	0...FFH	0...25.5	0.1	Mpa
Power steering pressure					
PSP_RATIO	V	0...FFFFH	0...99.99847	0.001526	%
Power steering pressure ratio between V_PSP_MES and reference voltage					
LV_PSP	V/O	0...01H	0...1	1	-
Logic value for power steering activation					

Input data:

V_PSP_MES	VCC_SENS_SUB_MES		

General information:

Acquisition reccurence : 10 ms

Formula section:

If V_PSP_MES >= C_PSP_MIN_DIAG

and V_PSP_MES <= C_PSP_MAX_DIAG

then PSP_RATIO = [V_PSP_MES / VCC_SENS_SUB_MES] * 100

PSP = [PSP_RATIO – C_PSP_OFS] / C_PSP_SLOP

else see Diagnosis and Limp home strategy in chapter A

endif

If the difference between the old and the new values of PSP exceeds the permissible gradient C_PSP_GRD_MAX, then the value remains unchanged. During the next measurement, if the maximum gradient is exceeded again, the new measured value will be stored in the RAM.

Errors are not stored with the monitoring of gradients. The only purpose is to extract invalid measured values (malfunctions).

Definition of LV_PSP

If PSP > C_PSP_H_THD with LV_PSP = 0

then LV_PSP = 1

else **if** PSP < C_PSP_L_THD with LV_PSP = 1

then LV_PSP = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PSP_GRD_MAX	1	0...FFH	0...25.5	0.1	Mpa
Maximum Power steering pressure gradient					
C_PSP_OFS	1	0...FFFFH	0...99.99847	0,001526	%
Off value to calculate PSP					
C_PSP_SLOP	1	0...FFFFH	0...99.99847	0,001526	%/MPa
Gain value to calculate PSP					
C_PSP_H_THD	1	0...FFH	0...25.5	0.1	Mpa
Power steering pressure threshold to make LV_PSTE = 1					
C_PSP_L_THD	1	0...FFH	0...25.5	0.1	Mpa
Power steering pressure threshold to make LV_PSTE = 0					
C_PSP_MIN_DIAG	1	0...3FFH	0...5	0.0195	V
Power steering pressure transducer pressure raw acquisition Min. threshold					
C_PSP_MAX_DIAG	1	0...3FFH	0...5	0.0195	V
Power steering pressure transducer pressure raw acquisition Max. threshold					

4.70.3 Power steering pressure gradient PSP_GRD

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PSP_GRD	V	8000H...7FFFH	-1882...+1881,94	14.7/256	MPa/s
Power steering pressure position gradient					
PSP_GRD_MMV	V/O	8000H...7FFFH	-1882...+1881,94	14.7/256	MPa/s
Power steering pressure position gradient filtered value					

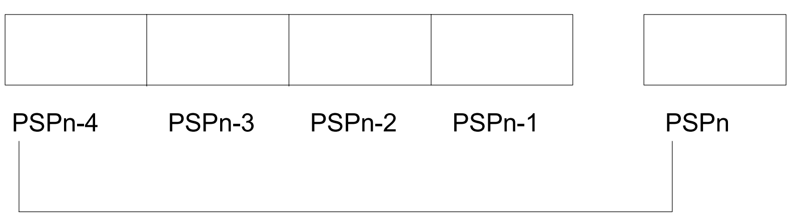
Input data:

PSP			

FUNCTION DESCRIPTION:

General information:

The power steering pressure gradient is calculated every 10 ms.
Therefore, the converted power steering pressure values are written into a ring buffer. This means that the last four power steering pressure values are available in addition to the current power steering pressure value.



The power steering pressure gradient is based on the change in power steering pressure angle over 40 ms.

Application conditions

Activation : at every engine state

Deactivation : -

Initialization : -

Formula section

$PSP\_GRD = PSP_{n} - PSP_{n-4}$

$PSP\_GRD\_MMV_{i} = PSP\_GRD\_MMV_{n-i} + C\_CRLC\_PSP * (PSP\_GRD - PSP\_GRD\_MMV_{i-1})$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PSP	1	0..FFFFH	0..1	1/65535	-
Filtering valur for Power steering gradiant with PSP_GRD					

4.71 Detection of Driver Request Passive

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CT	V/O	0...1H	0...1	1	-
Logical variable for detecting driver request passive					

Input data:

LV_MTC_CUR_OFF	FAC_TQ_REQ_CLU	N_32	

General information:

In this context 'driver request' stands for the driver wish itself, the cruise control or the vehicle speed limitation.
The detection of LV_CT = 0 (torque request from driver, cruise control or vehicle speed limiter) is done by evaluating the torque scaling factor FAC_TQ_REQ_CLU.
In case of LV_MTC_CUR_OFF = 1 (no more throttle control), the throttle will remain in the fixed limp home position, which is around 8° (depending on the used throttle); this is a part load position (LV_CT = 0). All torque requests can only be realized by spark advance interaction or single cylinder cut off coordination. For this purpose the driver passive detection is deactivated (LV_CT = 0).

Application conditions:

Recurrence: 10ms

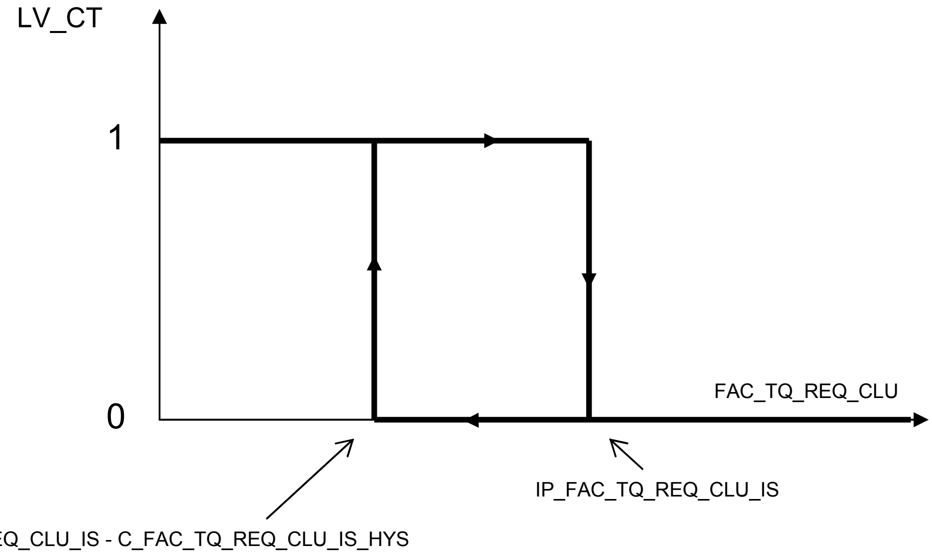
Activation: at every engine state

Deactivation: -

Initialization: at reset: LV_CT = 1

Formula section:

```
if      LV_MTC_CUR_OFF = 1
then    LV_CT = 0
else    if      LV_CT = 1
        then    if      FAC_TQ_REQ_CLU > IP_FAC_TQ_REQ_CLU_IS
                then    LV_CT = 0
                endif
        else    if      (FAC_TQ_REQ_CLU ≤ IP_FAC_TQ_REQ_CLU_IS
                                        - C_FAC_TQ_REQ_CLU_IS_HYS)
                then    LV_CT = 1
                endif
        endif
endif
```



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_REQ_CLU_IS	3	0...FFFFH	0...1.999969	3.052E-5	-
LDP_N_32_IP_FAC_TQ_REQ_CLU_IS	3	0...FFH	0...8160	32	rpm
Threshold on FAC_TQ_REQ_CLU for IS detection					
C_FAC_TQ_REQ_CLU_IS_HYS	1	0...FFFFH	0...1.999969	3.052E-5	-
Hysteresis on FAC_TQ_REQ_CLU for IS detection					

4.72 Brake switch variables

4.72.1 Signal acquisition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_BLS	V / O	0..1H	0..1	1	-
Brake Light Switch					
LV_BTS	V / O	0..1H	0..1	1	-
Brake Test Switch					

Input data:

LV_PIN_BLS	LV_PIN_BTS	LV_ES	LV_AT

FUNCTION DESCRIPTION:

General information:

The Bits LV_BLS and LV_BTS are set depending on evaluating the discrete Signals LV_PIN_BLS and LV_PIN_BTS.

Application conditions:

Initialisation: at reset

LV_BLS = 0

LV_BTS = 0

Recurrence: 10 ms

Activation: LV_IGK = 1

Deactivation: otherwise

Formula section:

If LV_ES = 0

Then LV_BLS = LV_PIN_BLS

LV_BTS = CPL(LV_PIN_BTS)

Else **If** LC_ENA_BLS_BTS_ENG_STOP = 1 **and** LV_AT = 1

(to allow LV_BLS/LV_BTS calculation at stopped engine for TCU only if required)

Then LV_BLS = LV_PIN_BLS

LV_BTS = CPL(LV_PIN_BTS)

Endif

Endif

4.72.2 Brake Toggling Detection (LV_BRAKE_TOG_EVE)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_BRAKE_TOG_EVE	V / O	0..1H	0..1	1	-
Logical Variable for detected brake toggling					

Input data:

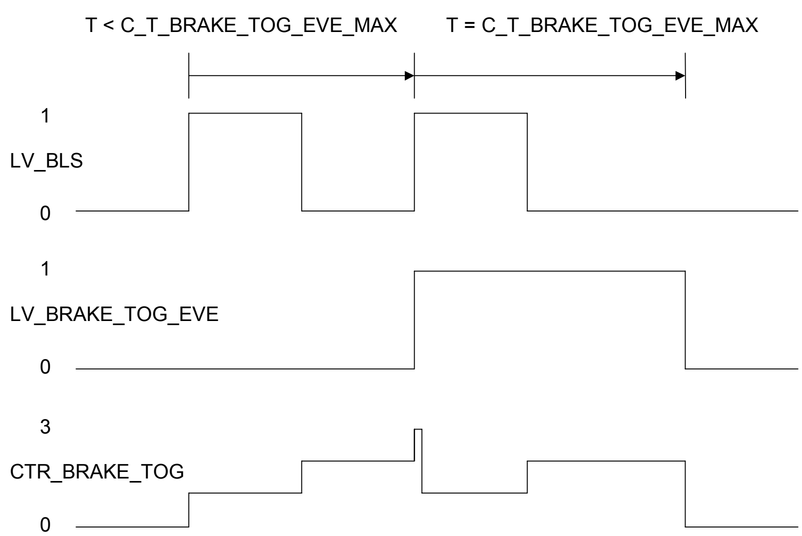
LV_BLS	LV_IGK	LV_ES	

FUNCTION DESCRIPTION:

General information:

If the driver pushes frequently the brake pedal, the brake booster can lead to pressure changes in the intake manifold, which then can affect other functions.
Therefore at detection of this brake toggle operation the bit LV_BRAKE_TOG_EVE = 1 is set.

If LV_BLS = 1, then the time C_T_BRAKE_TOG_EVE_MAX is started. If the transition from LV_BLS = 1 to LV_BLS = 0 and back to LV_BLS = 1 is detected before the time has elapsed, then the bit LV_BRAKE_TOG_EVE = 1 is set and the time C_T_BRAKE_TOG_EVE_MAX is restarted.
If the time elapses without detection of LV_BLS = 1 to LV_BLS= 0 to LV_BLS = 1, then LV_BRAKE_TOG_EVE is set back to 0.



Application conditions:

Activation:

LV_IGK = 1

and LV_ES = 0

Deactivation: otherwise

Initialisation: at reset

LV_BRAKE_TOG_EVE = 0

LV_BRAKE_TOG_EVE_CHK_ENA = 0

LV_BLS_TOG_OLD = 0

T_BRAKE_TOG = 0

CTR_BRAKE_TOG = 0

Update Rate: 100 ms

Formula section:

```
IF   LV_IGK = 1   AND
     LV_ES = 0
THEN
     IF   LV_BRAKE_TOG_EVE_CHK_ENA = 0
     THEN
          IF   LV_BLS_TOG_OLD = 0   AND
               LV_BLS = 1
          THEN
               CTR_BRAKE_TOG = 1
               T_BRAKE_TOG = C_T_BRAKE_TOG_EVE_MAX
               LV_BRAKE_TOG_EVE_CHK_ENA = 1
          END
     ELSE
          IF   T_BRAKE_TOG ≠ 0
          THEN
               T_BRAKE_TOG = T_BRAKE_TOG(k-1) - update rate
          ELSE
               CTR_BRAKE_TOG = 0
               LV_BRAKE_TOG_EVE = 0
               LV_BRAKE_TOG_EVE_CHK_ENA = 0
```

```
            END

            IF      T_BRAKE_TOG ≠ 0          AND
                    LV_BLS_TOG_OLD ≠ LV_BLS
            THEN
                    CTR_BRAKE_TOG = CTR_BRAKE_TOG(k-1) + 1

                    IF      CTR_BRAKE_TOG >= 3
                    THEN
                            CTR_BRAKE_TOG = 1
                            LV_BRAKE_TOG_EVE = 1
                            T_BRAKE_TOG = C_T_BRAKE_TOG_EVE_MAX
                    END
            END
    END

    LV_BLS_TOG_OLD = LV_BLS

ELSE
        LV_BLS_TOG_OLD = 0
        LV_BRAKE_TOG_EVE = 0
        LV_BRAKE_TOG_EVE_CHK_ENA = 0
        T_BRAKE_TOG = 0
        CTR_BRAKE_TOG = 0
END
```

4.72.3 Brake pedal activation detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_BLS_BTS_ACT_DC	O/V	0...1	0...1	1	-
Logical Variable for detected brake-pedal activation during present DC					

FUNCTION DESCRIPTION:

Function is used to detect if brake pedal activation was recognized at least one time in present driving cycle.

Application conditions:

Initialisation: at reset: LV_BLS_BTS_ACT_DC = 0

Recurrence: 100 ms

Activation: LV_IGK = 1

Formula section:

If LV_BLS_BTS_ACT_DC = 0
Then If LV_BLS = 1
and
LV_BTS = 1
Then LV_BLS_BTS_ACT_DC = 1
Endif
Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_BRAKE_TOG_EVE_MAX	1	0...FFH	0...25,5	0,1	s
maximum time to detect toggling brake operation					
LC_ENA_BLS_BTS_ENG_STOP	1	0...1	0...1	1	[-]
Logical constant to enable LV_BLS/LV_BTS calculation at stoped engine					

4.73 Electrical load signal from Alternator

Output data:

Name	Mod.	Hex. limits	Phys. limits	Resol.	Unit
GEN_LOAD	V/O	0...FFH	0...99.6	0.4	%
Electrical load PWM signal from Alternator					
GEN_LOAD_MMV	V/O	0...FFH	0...99.6	0.4	%
Filtered electrical load PWM signal from Alternator					
GEN_LOAD_MMV_FIL	V/O	0...FFFFH	0...99.6	0.0015	%
2nd Filtered electrical load PWM signal from Alternator for N_SP_IS calculation					
GEN_LOAD_GRD_MMV	V/O	8000 ... 7FFFH	-100 ... 99.996	3.051e-3	%/10ms
Filtered alternator load gradient signal					
GEN_LOAD_TMP	V	0...FFH	0...99.6	0.4	%
Electrical Load Signal from Alternator (Input signal from Basis SW)					

Input data:

LV_IGK	CONF_BAT		

General information:

The electrial load information is coming from alternator as PWM signal type.

The goal is to compute the alternator load for torque loss compensation to keep idle stability and to make battery charging by increment of engine speed in case of low battery voltage.

Application conditions:

Activation : LV_IGK = 1 **and** all engine operating state

Deactivation : -

Initialization : GEN_LOAD = GEN_LOAD_MMV = GEN_LOAD_MMV_FIL = 0 at reset

GEN_LOAD_GRD_MMV = 0 at reset

Recurrence: : 10ms

Formula section:

GEN_LOAD_TMP = Duty of GEN_LOAD PWM signal (INPUT from BSW)

if CONF_BAT = 0

then GEN_LOAD = GEN_LOAD_TMP

else GEN_LOAD = 100% - GEN_LOAD_TMP

to change the PWM state of Valeo regulator for BEM from positive to negative

Duty cycle

$$GEN\_LOAD\_MMV_n = GEN\_LOAD\_MMV_{n-1} + (GEN\_LOAD_n - GEN\_LOAD\_MMV_{n-1}) * C\_CRLC\_GEN\_LOAD$$

$GEN\_LOAD\_MMV\_FIL_n = GEN\_LOAD\_MMV\_FIL_{n-1} + (GEN\_LOAD\_MMV_n$

$- GEN\_LOAD\_MMV\_FIL_{n-1}) * C\_CRLC\_GEN\_LOAD\_MMV$

If there is no signal switching during the 21ms(specified inside SW), the PWM value is set to 99.6% if the GEN_LOAD level is low and the PWM value is set to 0% if the GEN_LOAD level is high.

$GEN\_LOAD\_GRD = GEN\_LOAD\_MMV_n – GEN\_LOAD\_MMV_{n-1}$

$GEN\_LOAD\_GRD\_MMV = GEN\_LOAD\_GRD\_MMV_{n-1} +$

$(GEN\_LOAD\_GRD_n - GEN\_LOAD\_GRD\_MMV_{n-1})$

$* C\_CRLC\_GEN\_LOAD\_GRD\_MMV$

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CRLC_GEN_LOAD	1	0...FFH	0...0.996	0.004	-
Filtering factor of GEN_LOAD					
C_CRLC_GEN_LOAD_MMV	1	0...FFFFH	0...0.996	0.000015	-
Filtering factor of GEN_LOAD_MMV for N_SP_IS calculation					
C_CRLC_GEN_LOAD_GRD_MMV	1	0...FFH	0...0.996	0.004	-
Filtering factor of GEN_LOAD_GRD					

4.74 Fuel Tank Level (FTL) acquisition variable

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FTL	V/O	0...FFH	0...100	0.39	%
Fuel tank level					
FTL_MES	V	0...FFH	0...100	0.39	%
Measured Fuel tank level					
FTL_VB_COR	V	0...FFFFH	0...100	0.0015	%
Fuel tank level corrected by battery voltage					
FTL_MMV	V	0...FFFFH	0...100	0.0015	%
Filtered Fuel tank level value					
V_FTL_SUB_MMV	V/O	0...FFH	0...4.985	0.0195	Volt
Filtered V_FTL_SUB value (from sub FTL sensor)					

Input data:

V_FTL	VB	LV_IGK	CONF_DIAGCP_VOL
V_FTL_SUB	C_FTL_LIH	LV_ERR_FTL_STUCK	LV_ERR_FTL_STUCK_H
LV_ERR_FTL_SUB_STUCK	LV_ERR_FTL		

FUNCTION DESCRIPTION:

The FTL information is issued from Fuel tank sensor.

This acquisition is done by wires. not by CAN.

4.74.1 FTL Hardware acquisition

General information:

The FTL is monitored for invalid gradient and values.

Acquisition recurrence : 0.1 s.

Initialisation:

FTL = FTL_MES (after Key on)

Formula section:

The FTL_MES is monitored for unplausible gradient.

FTL = FTL_MES
FTL_MES = IP_FTL_MES__V_FTL

except if :

If the difference between the old and the new FTL value exceeds the permissible gradient C_FTL_GRD_MAX. then the FTL value remains unchanged.

During the next measurement. if the maximum gradient is exceeded again. the new measured value will be stored in the RAM.

Errors are not stored with the monitoring of gradients. The only purpose is to extract invalid measured values (malfunctions).

Remark :

If the key off (LV_IGK 1->0) is detected, the previous FTL value is kept because the V_FTL value goes to 0 V.

4.74.2 Voltage battery correction

General information:

The FTL is corrected depending on battery voltage.

Recurrence : 0.1 s.

Formula section:

FTL_VB_COR = FTL * IP_FAC_FTL__VB__FTL

4.74.3 Filtered Fuel Tank Pressure FTL_MMV

General information:

The FTL_VB_COR is filtered to avoid big fluctuations due to fuel sloshes.

Recurrence : 0.1 s.

Initialisation:

FTL_MMV = FTL_VB_COR (after key on)

V_FTL_SUB_MMV = V_FTL_SUB (after key on)

Formula section:

If LV_ERR_FTL = 0
and LV_ERR_FTL_STUCK = 0
and LV_ERR_FTL_STUCK_H = 0
and LV_ERR_FTL_SUB_STUCK = 0
then
  If CONF_DIAGCP_VOL = 1 (with single FTL sensor)
  then $FTL\_MMV_i = FTL\_MMV_{i-1} + C\_FTL\_MMV\_GAIN * (FTL\_VB\_COR_i - FTL\_MMV_{i-1})$
  V_FTL_SUB_MMV = 0

  else if CONF_DIAGCP_VOL = 2 (with dual FTL sensor)
  then $FTL\_MMV_i = FTL\_MMV_{i-1} + C\_FTL\_MMV\_GAIN * (FTL\_VB\_COR_i - FTL\_MMV_{i-1})$
  $V\_FTL\_SUB\_MMV_i = V\_FTL\_SUB\_MMV_{i-1} + C\_FTL\_MMV\_GAIN * (V\_FTL\_SUB_i - V\_FTL\_SUB\_MMV_{i-1})$

  else if CONF_DIAGCP_VOL = 0 (without FTL sensor)
  then FTL_MMV = 49.99 [%]
  V_FTL_SUB_MMV = 0 [V]

```
        endif
else    FTL_MMV = C_FTL_LIH                        (irreversible)
        V_FTL_SUB_MMV = V_FTL_SUB_MMV              (irreversible)
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FTL_GRD_MAX	1	0...FFH	0...100	0.39	%
Maximum fuel tank level gradient					
IP_FTL_MES__V_FTL	12	0...FFH	0...100	0.39	%
LDP_V_FTL__IP_FTL_MES	12	0...FFH	0...4.985	0.0195	V
Fuel tank level from basic input					
IP_FAC_FTL__VB__FTL	8	0...FFH	0...1.999	0...0.0078	-
LDP_VB_IP_FAC_FTL	8	0...FFH	0...26	0...0.102	-
LDP_FTL__IP_FAC_FTL	8	0...FFH	0...100	0.39	%
Multiplicative VB correction for FTL					
C_FTL_MMV_GAIN	1	0..FFH	0...1	3.9e-3	-
Fuel tank level filter gain					

4.75 Acquisition of pressure at wastegate actuator

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PRS_WG_ACR_MES	O	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Measured pressure in wastegate actuator chamber					

4.75.1 CHRG_SIGCVPWGAC0

General information:

Dummy version for systems without wastegate actuator pressure sensor.

Function Description

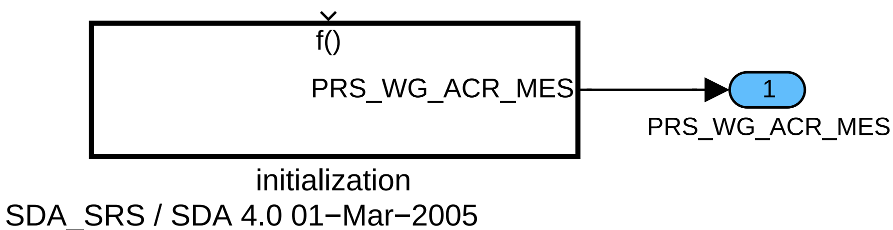


Figure 398 CHRG_SIGCVPWGAC0

4.75.1.1 SUBFUNCTION: initialization

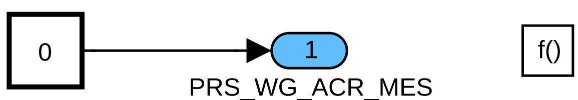


Figure 399 CHRG_SIGCVPWGAC0/ initialization

4.76 Engine off duration

4.76.1 Engine Off Timer

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_ES	V/O	0...FFFFH	0...65535	1	[min]
Engine off duration time					
T_ES_DIAG	V/O	0...FFFFFFH	0...16777215	1	[sec]
Engine off duration time, high resolution (for diagnosis, value not stopped at LV_ST_END)					
T_ES_IGK	V/O	0...FFFFFFH	0...16777215	1	[sec]
Engine off time at IGK_OFF --> ON (for diagnosis interface)					
T_IGK_ON	V/O	0...FFFFH	0... 65535	1	[sec]
Reference Timer started at IGK ON (for diagnosis interface)					
LV_T_ES_NOT_PLAUS	V/O	0...1H	0...1	1	-
Flag indication T_ES value is not plausible					
LV_T_ES_RST	V/O	0...1H	0...1	1	-
Flag indication T_ES value is currently resetted					
LV_T_ES_ACT	V/O/S	0...1H	0...1	1	-
Flag indication engine off timer is not yet activated for the first time					
T_ES_RST_ACT	V	0....FFFFH	0....65535	1	-
0H = no warm reset/ AF50H 0 warm reset – Word is stored every sec to reset resistant (RAM CHECK resistant) memory as value and complement					

Input data:

LV_ES	LV_IGK	LV_ST_END	LV_ERR_T_ES_PLAUS
LV_VB_OFF			

Import actions :

ACTION_INFR_EONVStartTimer ( )
Start the timer of the PIC
ACTION_INFR_EONVResetTimer ( )
Reset the basic timer of the PIC
ACTION_INFR_EONVReadTimer (Lv_cmd, T_es **)**
Engine Off timer is read in from PIC

FUNCTION DESCRIPTION:

General information:

The engine off duration time T_ES will be used mainly in temperature critical functions like injection and dew point recognition.

The engine off timer is running in a separate micro controller and is controlled by commands over a dedicated SPI line.

As soon as the condition LV_ES = 1 and LV_IGK = 1 is recognised the value of T_es is read from basic software and T_ES, T_ES_DIAG, T_ES_IGK are updated. The timer values must be upated before functions using this data are called (example, injection functions).

The engine off timer T_ES value is stopped after Start end detection, while the T_ES_DIAG timer for diagnosis is only resetted at transition to stopped engine.

At first Power on (new ECU) or after loss of NVMY or disconnected battery, the engine off timer is not started, as there is not yet a transition to stopped engine detected. Therefore the plausibility flag LV_T_ES_NOT_PLAUS = 1 is set.

For calibration purpose an add on value C_T_ES_MAN can be set to the Engine off timer value.

Due to communication delay with PIC controller a reset of the engine off timer takes a few 100 msec. The flag LV_T_ES_RST shows, that the engine off timer value could be delayed. This is important for other functionalities with higher recurrence using it.

To avoid a wrong Initialisation of values which are using T_ES after a warm reset (software reset), it has to be ensured, that LV_T_ES_NOT_PLAUS is set to 1 in case of warm reset. This is handled via the word T_ES_RST_ACT, which is saved in reset independent memory with value and complement value. This special part of the memory is not deleted in case of warm reset (Software reset). In case of cold reset (hardware reset) this information is lost.

Application conditions:

Initialisation:

at first power on or NVMY error:

LV_T_ES_ACT = 0

at reset:

if T_ES_RST_ACT = AF50h (**and** complement value = 50AFh)

then *warm reset (SW-reset) is detected*

LV_T_ES_NOT_PLAUS = 1

endif

Hint: To perform the battery off detection with engine off timer is has to be ensured by SW, that the reset of engine off timer has to be done prior to reset of LV_VB_OFF.

T_ES, T_ES_DIAG must be updated during initialization before being used in temperature critical functions like injection and dew point recognition.

if LC_T_ES_MAN = 0

then T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer

(information obtained from PIC)

else T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer + C_T_ES_MAN

(information obtained from PIC plus manually added time)

endif

if T_ES_DIAG > 3 932 100 sec (3BFFC4h)

then T_ES = 65535 min (FFFFh)

(to avoid T_ES counter overun)

else T_ES = T_ES_DIAG/60

endif

if LV_VB_OFF = 1

then LV_T_ES_ACT = 0

endif

if LV_T_ES_ACT = 0

or LV_ERR_T_ES_PLAUS = 1

then LV_T_ES_NOT_PLAUS = 1

endif

T_IGK_ON = 0

T_ES_IGK = T_ES_DIAG

LV_T_ES_RST = 0

at LV_IGK = 0-->1:

T_IGK_ON = 0

T_ES_IGK = T_ES_DIAG

at LV_IGK = 1--> 0:

T_ES_RST_ACT = 0h (and complementary value =FFFFH)

at LV_ES = 0-->1: (detection of stopped engine)

LV_T_ES_RST = 1

LV_T_ES_NOT_PLAUS = 1

function call: ACTION_INFR_EONVResetTimer ( )

(*send reset timer command)*

function call: ACTION_INFR_EONVStartTimer ( )

(*send start timer command)*

Hint: The command to reset timer has to be send allways before start timer command to ensure that the timer is resetted before starting

if LC_T_ES_MAN = 0

then T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer

(information obtained from PIC)

else T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer + C_T_ES_MAN

(information obtained from PIC plus manually added time)

endif

T_IGK_ON = 0

T_ES_IGK = 0

LV_T_ES_ACT = 1

Hint: The reset of the T_IGK_ON and T_ES_IGK has to be done after the T_ES_DIAG is obtained from PIC)

Recurrence: 1000ms

Activation:

Deactivation: *at "end power latch"*

Formula section:

Management of T_ES_RST_ACT:

If LV_IGK = 1

then T_ES_RST_ACT = AF50h (and complementary = 50AFh)

(to be written to reset resistant memory every sec)

endif

Management of engine off timer

if LC_T_ES_MAN = 0

then T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer

(information obtained from PIC)

else T_ES_DIAG = Result from ACTION_INFR_EONVReadTimer + C_T_ES_MAN

(information obtained from PIC plus manually added time)

endif

if LV_ST_END = 0

then **if** T_ES_DIAG > 3 932 100 sec (3BFFC4h)

then T_ES = 65535 min (FFFFh)

(to avoid T_ES counter overun)

else T_ES = T_ES_DIAG/60

endif

else T_ES (n) = T_ES (n-1)

(after Start end the T_ES value is not incremented anymore)

endif

Calculation of Main controller timer reference timer (for Diagnosis):

if LV_IGK = 1

then **if** T_IGK_ON < 65535 sec

then T_IGK_ON (n) = T_IGK_ON (n-1) + 1

else T_IGK_ON = 65535 sec

endif

Management of Plausibilty Flag:

if LV_T_ES_ACT = 0 *(battery off, loss of NVMY, first power on)*

or LV_ERR_T_ES_PLAUS = 1 (*Plausibilty Error)*

or LV_T_ES_RST = 1 (*engine off timer value not yet valid due to*

(PIC response delay)

then LV_T_ES_NOT_PLAUS = 1

else LV_T_ES_NOT_PLAUS = 0

Management of Reset flag:

if T_ES < = C_T_ES_RST

then LV_T_ES_RST = 0

T_ES is resetted, value can be used for other functionalities

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_T_ES_MAN	1	0...1H	0...1	1	[-]
switch for Manual T_ES value					
C_T_ES_MAN	1	0...FFFFFFH	0... 16777215	1	sec
Manual T_ES value					
C_T_ES_RST	1	FFFFH	0...65535	1	[min]
Threshold for engine off timer reset detection					

4.77 Engine roughness segment time correction

4.77.1 Phasing reference for the acquisition of engine roughness segment time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_NR_MES	V/O	0...BH	0...11	1	[-]
Phasing reference of ER segment acquisition					

Input data:

SEG_NR	NC_CYL_NR	LV_SYN_ENG	

FUNCTION DESCRIPTION:

The ER segments provided by the aggregate ENSD, are used to observe in-cylinder combustion effects on crankshaft. The cylinder x combustion is located between cylinder X tdc and cylinder X+1 tdc. Consequently, at X+1 tdc, the T_SEG_ER obtained is relative to cylinder X combustion, a delay of 1 tdc between SEG_NR value at cylinder observed *(NC_SEG_DLY_ER_MES=1)*.

Exceptionally, if the segment is located after ER segment, this delay must be cancelled *(NC_SEG_DLY_ER_MES=0)*, because in such case the phasing reference SEG_NR is updated after T_SEG_ER acquisition.

SEG_NR_MES, the ER segments phasing reference, will be used to identify which cylinder is observed with the current engine roughness segment, it's particularly useful for ER segment adaptive values identification.

Application conditions:

Initialisation: *at ECU reset, Engine Stop* ***Or*** *Deactivation event*

SEG_NR_MES = 0

Recurrence: Segment

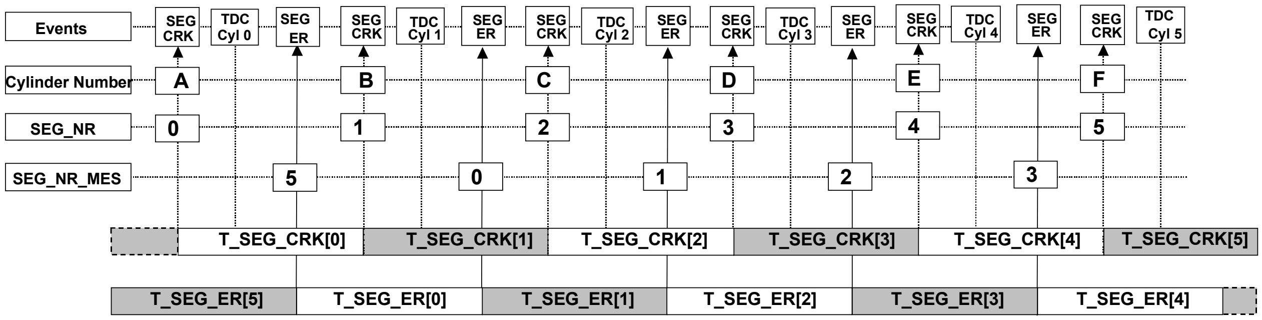
Activation: LV_SYN_ENG = 1

Deactivation: LV_SYN_ENG = 0

Formula section:

SEG_NR_MES = (SEG_NR – NC_SEG_DLY_ER_MES) % NC_CYL_NR

Implant scheme:



6 Cylinder Engine Example : *NC_SEG_DLY_ER_MES = 1*

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_SEG_DLY_ER_MES	-	0...2H	0...2	1	[-]
Delay between ER segment and Cylinder measured segment					

4.77.2 Engine roughness segment time acquisition and control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_T_MES	V/O	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Engine roughness raw segment time relative to former TDC					
SEG_T_MES_CYL[NC_CYL_NR]	V/O	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Buffer of cylinder engine roughness raw segment times					
LV_ENA_SEG_T_MES	V/O	0...1H	0...1	1	[-]
Engine roughness segment time validity flag					

Input data:

T_SEG_ER	LV_SYN_ENG	LV_LIH_ERR_CRK	LV_INH_MIS_CRK
NC_CYL_NR	LV_ERR_CRK		

General information:

Here's described the segment time buffers management & control

ER current raw segment time:

T_SEG_ER *(input from ENSD)*

Coming directly from the ENSD aggregate and update at each ER segment task according ER segment counter *(SEG_NR_MES)*.

Actual ER segment time sample is considered **unvalid** if :

- Engine synchronisation is disable (LV_SYN_ENG = 0)
- When on one crankshaft revolution at least one tooth less or more has been detected (LV_INH_MIS_CRK = 1 before antibounce counter),
- When the crahshaft error or limp home is activated (LV_ERR_CRK = 1 or LV_LIH_ERR_CRK = 1)
- SEG_T_MES obtained exceeds a defined valid range (C_SEG_T_MES_MIN & C_SEG_T_MES_MAX)

Cylinder ER raw segment times:

SEG_T_MES & SEG_T_MES_CYL[NC_CYL_NR] *(outputs)*

This is a buffer where T_SEG_ER values *(after data casting)* are stored according SEG_NR_MES counter. It allows having a memory of raw ER segments on a complete engine cycle. These values will be used by the ER adaptive learning process.

Application conditions:

Initialisation: *at ECU reset, Engine Stop* ***Or*** *Deactivation event*

SEG_T_MES = 0x7FFFFF

For x = 0 ... NC_CYL_NR-1

SEG_T_MES_CYL[x] = 0x7FFFFF

EndFor

LV_ENA_SEG_T_MES = 0

Recurrence: Segment

Activation: LV_SYN_ENG = 1

Deactivation: LV_SYN_ENG = 0

Formula section:

SEG_T_MES = T_SEG_ER *// physical equivalence conversion*

If LV_SYN_ENG = 0

Or SEG_T_MES > C_SEG_T_MES_MAX

Or SEG_T_MES < C_SEG_T_MES_MIN

Or LV_INH_MIS_CRK = 1

Or LV_ERR_CRK = 1

Or LV_LIH_ERR_CRK = 1

Then LV_ENA_SEG_T_MES = 0

SEG_T_MES = 0x7FFFFF

Else LV_ENA_SEG_T_MES = 1

EndIf

SEG_T_MES_CYL[SEG_NR_MES] = SEG_T_MES

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_SEG_T_MES_MAX	1	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
ER maximum valid segment time					
C_SEG_T_MES_MIN	1	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
ER minimum valid segment time					

4.77.2.1 Correction of the engine roughness segment adaptive values

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_AD_COR_ER	V	0...FFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER segment adaptive values correction versus engine speed (for each segment)					

Input data:

LV_AT	N_32	NC_CYL_NR	

FUNCTION DESCRIPTION:

General information:

Due to crankshaft flexion appearance at high engine speed that decreases the misfire detection efficiency, an open-loop adaptive correction value depending on engine speed for each cylinder can be introduced per type of transmission.

A correction can be applied only for each singular segment of the flywheel.

Application conditions:

Initialisation: *at ECU reset, Engine Stop* ***Or*** *Deactivation event*

SEG_AD_COR_ER = 0

Recurrence: Segment

Activation: LV_ENA_SEG_T_MES = 1

Deactivation: LV_ENA_SEG_T_MES = 0

Formula section:

SEG_AD_COR_ER is updated according the following table *(function of NC_CYL_NR and the current value of SEG_NR_MES)*

SEG_NR_MES	NC_CYL_NR = 3	NC_CYL_NR = 4	NC_CYL_NR = 5	NC_CYL_NR = 6	NC_CYL_NR = 8	NC_CYL_NR = 10	NC_CYL_NR = 12
0	SEG_AD_COR_ER = 0						
1	SEG_AD_COR_ER = IP_SEG_AD_COR_ER_xT_1(N_32)						
2	IP_SEG_AD_COR_ER_ xT_2(N_32)	0	IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)
3		IP_SEG_AD_COR_ER_ xT_1(N_32)	IP_SEG_AD_COR_ER_ xT_3(N_32)	0	IP_SEG_AD_COR_ER_ xT_3(N_32)	IP_SEG_AD_COR_ER_ xT_3(N_32)	IP_SEG_AD_COR_ER_ xT_3(N_32)
4			IP_SEG_AD_COR_ER_ xT_4(N_32)	IP_SEG_AD_COR_ER_ xT_1(N_32)	0	IP_SEG_AD_COR_ER_ xT_4(N_32)	IP_SEG_AD_COR_ER_ xT_4(N_32)
5				IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_1(N_32)	0	IP_SEG_AD_COR_ER_ xT_5(N_32)
6					IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_1(N_32)	0
7					IP_SEG_AD_COR_ER_ xT_3(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)	IP_SEG_AD_COR_ER_ xT_1(N_32)
8						IP_SEG_AD_COR_ER_ xT_3(N_32)	IP_SEG_AD_COR_ER_ xT_2(N_32)
9						IP_SEG_AD_COR_ER_ xT_4(N_32)	IP_SEG_AD_COR_ER_ xT_3(N_32)
10							IP_SEG_AD_COR_ER_ xT_4(N_32)
11							IP_SEG_AD_COR_ER_ xT_5(N_32)

_xT stands for _AT if LV_AT = 1 else stands for _MT

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
IP_SEG_AD_COR_ER_MT_1	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_1_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in MT for segment 1					
IP_SEG_AD_COR_ER_AT_1	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_6_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in AT for segment 1					
#IF NC_CYL_NR=3, 5, 6, 8,10 or 12					
IP_SEG_AD_COR_ER_MT_2	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_1_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in MT for segment 2					
IP_SEG_AD_COR_ER_AT_2	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_6_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in AT for segment 2					
#ENDIF					
#IF NC_CYL_NR=5,8,10,12					
IP_SEG_AD_COR_ER_MT_3	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_1_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in MT for segment 3					
IP_SEG_AD_COR_ER_AT_3	6	0...FFFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_6_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in AT for segment 3					
#ENDIF					
#IF NC_CYL_NR=5,10,12					

IP_SEG_AD_COR_ER_MT_4	6	0...FFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_1_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in MT for segment 4					
IP_SEG_AD_COR_ER_AT_4	6	0...FFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_6_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in AT for segment 4					
#ENDIF					
#IF NC_CYL_NR=12					
IP_SEG_AD_COR_ER_MT_5	6	0...FFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_1_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in MT for segment 5					
IP_SEG_AD_COR_ER_AT_5	6	0...FFFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
LDPM_N_32_6_ENRD	6	0...FFH	0...8160	32	[rpm]
Adaptive value correction versus engine speed in AT for segment 5					
#ENDIF					

4.77.3 Engine roughness segment time correction

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_T_COR_BUF[NC_SIZE_SEG_T_COR_BUF]	O	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
ER corrected segment times buffer (after adaptation)					
SEG_T_COR	V/O	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Current ER corrected segment time					

Input data:

LV_SYN_ENG	SEG_AD_MMV_ER[NC_CYL_NR]		

FUNCTION DESCRIPTION:

General information:

The prerequisite for a reliable engine roughness evaluation is an accurate ER segment time measurement.

As the crankshaft flywheel teeth are subject to manufacturing tolerances, geometrical run-out and off-center installation which result in angles modulation, consequently the measured raw segment periods aren't similar and have systematic inaccuracies regarding other segments.

As these inaccuracies are systematic, they can be "learned" by an adaptive process during fuel cut-off periods and used for ER segment time correction. This adaptive process is described in the module : Engine roughness adaptive learning process.

ER segment adaptive values: SEG_AD_MMV_ER[NC_CYL_NR] *(inputs)*

ER adaptive values are stored according SEG_NR_MES counter. These adaptive values are obtained and updated by the ER segment adaptive values learning process. There's one ER segment adaptive value per engine cylinder.

Remark: The SEG_AD_MMV_ER_x are stored in ECU non-volatile memory.

ER current corrected segment time: SEG_T_COR *(output)*

This value is the current result of the ER segment correction process obtained with T_SEG_ER as input.

<u>ER corrected segment times buffer:</u> SEG_T_COR_BUF[NC_SIZE_SEG_T_COR_BUF] *(outputs)*

This is a buffer where are stored SEG_T_COR values (ER corrected segment time current value) who are needed for ER calculation.

SEG_T_COR is stored according a FIFO management (first in, first out).

SEG_T_COR_BUF buffer size depends on engine type:

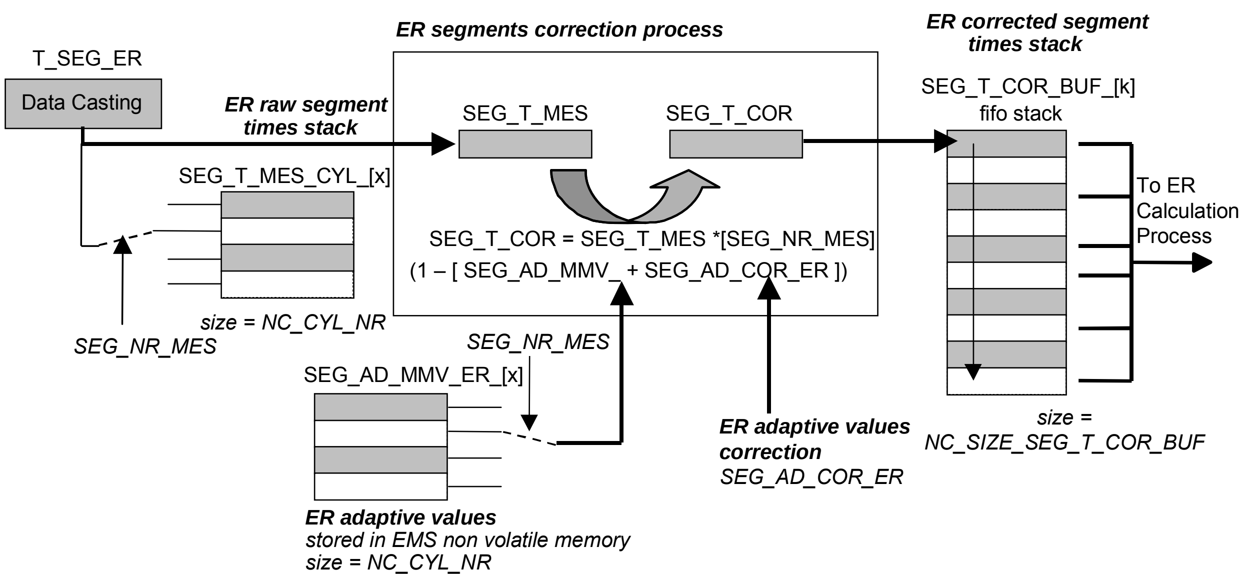
NC_SIZE_SEG_T_COR_BUF = *11 segment time samples for a 5 & 10 cylinder engine*

NC_SIZE_SEG_T_COR_BUF = *9 segment time samples for 4 & 8 cylinder engine*

NC_SIZE_SEG_T_COR_BUF = *7 segment time samples for 3 & 6 cylinder engine*

NC_SIZE_SEG_T_COR_BUF = *13 segment time samples for a 12 cylinder engine*

This size NC_SIZE_SEG_T_COR_BUF *is determined by the ER sample number required by ER calculation according engine type.*



<u>**Application conditions:**</u>

Initialisation: at ECU reset, Engine Stop ***Or*** *Deactivation event*

SEG_T_COR = 0x7FFFFF

For k = 0 to NC_SIZE_SEG_T_COR_BUF-1

SEG_T_COR_BUF[k] = 0x7FFFFF

EndFor

Recurrence: Segment

Activation: LV_SYN_ENG = 1

Deactivation: LV_SYN_ENG = 0

Formula section:

If(1) LV_ENA_SEG_T_MES = 0

Then(1) SEG_T_COR = 0x7FFFFF

For(2) k = 0 to NC_SIZE_SEG_T_COR_BUF-1

SEG_T_COR_BUF[k] = 0x7FFFFF

EndFor(2)

Else(1) SEG_T_COR =

SEG_T_MES * (1 – [SEG_AD_MMV_ER[SEG_NR_MES] + SEG_AD_COR_ER])

For(2) k : 1 to NC_SIZE_SEG_T_COR_BUF-1

SEG_T_COR_BUF[k] is managed as a FIFO stack *(SEG_T_COR_BUF last value is lost)*

EndFor(2)

SEG_T_COR_BUF[0] = SEG_T_COR

EndIf(1)

Where:

SEG_T_MES : *Current uncorrected ER segment time after data casting*

SEG_T_COR : *Current ER corrected segment time*

SEG_AD_MMV_ER[SEG_NR_MES] :

ER segment adaptive value corresponding to current T_SEG_ER phasing

SEG_AD_COR_ER :

Adaptive value correction versus engine speed corresponding to current T_SEG_ER phasing

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_SIZE_SEG_T_COR_BUF	-	0...FFH	0...255	1	[-]
Size of the array of engine roughness corrected segment times					

4.77.4 Engine roughness calculation

4.77.5 General information

The engine roughness index (ER) is a system variable relative to crankshaft segments behaviour. This index is mainly used by the misfire monitoring & algorithms dedicated to combustion process control.

Integration version for 3,4,5,6,8,10 & 12 cylinder engines.

The engine roughness system variable is calculated each ER window end occurrence, every time the engine is running & synchronised.

4.77.6 Engine roughness segment reference

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_NR_ER	V/O	0...BH	0...11	1	[-]
ER segment counter for segment identification (0 ... NC_CYL_NR - 1)					

Input data:

SEG_NR	NC_CYL_NR	LV_SYN_ENG	

Due to engine roughness definition, there is a delay of C_SEG_DLY_ER segments between ER actual value and the segment that was observed by this ER value.

SEG_NR_ER will be used to identify which cylinder is observed with the engine roughness system variable, it's particularly useful for misfire cylinder identification.

The segment counter SEG_NR_ER with the limit 0...(NC_CYL_NR - 1) is incremented in the same way as the common segment counter SEG_NR.

Application conditions:

Initialisation: *at ECU reset, Engine Stop* ***Or*** *Deactivation condition event*

SEG_NR_ER = 0

Recurrence: Segment

Activation: LV_SYN_ENG = 1

Deactivation: LV_SYN_ENG = 0

Formula section:

SEG_NR_ER = (SEG_NR + 2 * NC_CYL_NR – C_SEG_DLY_ER) % NC_CYL_NR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_SEG_DLY_ER	1	0...CH	0...12	1	[-]
Delay of ER segment reference					

4.77.7 Engine roughness segments control process

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ENA_ER	V	0...1H	0...1	1	[-]
Engine roughness valid calculation control flag					

Input data:

SEG_T_COR_BUF[NC_SIZE_SEG_T_COR_BUF]	LV_ENA_SEG_T_MES	N	LV_AT

Description:

All ER components calculations are based on corrected segment time acquisitions buffer values SEG_T_COR_BUF.

As soon as an invalid segment has been detected during acquisition process (LV_ENA_SEG_T_MES = 0) or segment adaptive calculation, ER calculation & misfire detection have to be disabled (LV_ENA_ER = 0).

Application conditions:

Initialisation: *at ECU reset, Engine Stop* ***Or*** *Deactivation condition event*

LV_ENA_ER = 0

Recurrence: Segment

Activation:

Engine roughness calculation is executed only if:

NC_SIZE_SEG_T_COR_BUF consecutive valid ER segment times have been acquired & stored in SEG_T_COR_BUF buffer.

"**Valid**" stands for:

LV_ENA_SEG_T_MES = 1 on NC_SIZE_SEG_T_COR_BUF consecutive ER segments

And no ER components calculation overflow on current segment *(DRV0, DRV1, DRV2 and ER components)*

And N <= C_N_MAX_ER

And [(N >= C_N_MIN_ER_MT **and** LV_AT = 0) **or**

(N >= C_N_MIN_ER_AT **and** LV_AT = 1)]

In this case:

ER components are calculated as described below.

LV_ENA_ER = 1 *(if no overflow occurs during ER components calculation)*

Deactivation:

Engine roughness calculation is stopped & initialised when at least one of the NC_SIZE_SEG_T_COR_BUF consecutive segment times have been acquired & stored in SEG_T_COR_BUF buffer have been detected unvalid.

"**Unvalid**" stands for:

LV_ENA_SEG_T_MES = 0 at least one time on NC_SIZE_SEG_T_COR_BUF consecutive ER segments

Or at least one calculation overflow occurs one of ER components *(DRV0, DRV1, DRV2 or ER component)* on current segment.

Or N > C_N_MAX_ER

Or (N < C_N_MIN_ER_MT **and** LV_AT = 0)

Or (N < C_N_MIN_ER_AT **and** LV_AT = 1)

In that case :

LV_ENA_ER = 0

ER components are initialised to 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_MIN_ER_MT	1	0...1FE0H	0...8160	1	[rpm]
minimum engine speed for engine roughness calculation for MT-vehicle					
C_N_MIN_ER_AT	1	0...1FE0H	0...8160	1	[rpm]
minimum engine speed for engine roughness calculation for AT-vehicle					
C_N_MAX_ER	1	0...1FE0H	0...8160	1	[rpm]
maximum engine speed for engine roughness calculation					

4.77.8 Cylinder specific engine roughness components correction

#IF NLC_CONF_GAIN_ADD_ER = 1

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_GAIN_ER[NC_CYL_NR]	V	0...1FFH	0...1.99609	3.9063e-3	[-]
ER multiplicative correction components					
FAC_ADD_ER[NC_CYL_NR]	V	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
ER additive correction components					

Input data:

LV_AT	LOAD_MIS	N_32	NC_CYL_NR

FUNCTION DESCRIPTION:

General information:

The misfire detection based on ER index can be complicated by the torsional/flexion vibrations especially with multiple cylinder engines with a long crankshaft. This affects on misfire detection efficiency, especially at high engine speed.

To balance this phenomenon, after segment timing correction with ER adaptive values *(see ER segment correction process)*, an irregular operation is determined from the measured cylinder segment times, cylinder specific additive and multiplicative corrections are applied on the raw engine roughness value *(ER_RAW)*.

FAC_ADD_ER[NC_CYL_NR] and FAC_GAIN_ER[NC_CYL_NR] correction coefficients are based on calibration tables specific to each cylinder and to transmission type (MT/AT).

Application conditions:

Initialisation: at ECU reset, at Engine Stop ***Or*** *on Deactivation condition event*

For x = 0 to NC_CYL_NR-1

FAC_GAIN_ER[x] = 1

FAC_ADD_ER[x] = 0

EndFor

Recurrence: *Segment*

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Formula section:

If LV_AT = 1

Then FAC_GAIN_ER[SEG_NR_ER] = IP_FAC_GAIN_ER_AT[SEG_NR_ER](N_32)

FAC_ADD_ER[SEG_NR_ER] =

IP_FAC_ADD_ER_AT[SEG_NR_ER](N_32,LOAD_MIS)

Else

FAC_GAIN_ER[SEG_NR_ER] = IP_FAC_GAIN_ER_MT[SEG_NR_ER](N_32)

FAC_ADD_ER[SEG_NR_ER] =

IP_FAC_ADD_ER_MT[SEG_NR_ER](N_32,LOAD_MIS)

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_GAIN_ER_MT[NC_CYL_NR]	6	0...1FFH	0...1.99609	3.9063e-3	[-]
LDPM_N_32_2_ENRD	6	0...FFH	0...8160	32	[rpm]
ER cylinder x multiplicative segment-dependent factor for torsion correction with MT					
IP_FAC_ADD_ER_MT[NC_CYL_NR]	6*6	0...FFFFFFH	-2000000... 1999999.7	0.2384186	[µs]
LDPM_N_32_3_ENRD	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_ADD_ER_MT	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER cylinder x additive segment-dependent factor for torsion correction with MT					
IP_FAC_GAIN_ER_AT[NC_CYL_NR]	6	0...1FFH	0...1.99609	3.9063e-3	[-]
LDPM_N_32_4_ENRD	6	0...FFH	0...8160	32	[rpm]
ER cylinder x multiplicative segment-dependent factor for torsion correction with AT					
IP_FAC_ADD_ER_AT[NC_CYL_NR]	6*6	0...FFFFFFH	-2000000... 1999999.7	0.2384186	[µs]
LDPM_N_32_5_ENRD	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_ADD_ER_AT	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER cylinder x additive segment-dependent factor for torsion correction with AT					

#ENDIF

Configuration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
NLC_CONF_GAIN_ADD_ER	-	0...1H	0...1	1	[-]
ER multiplicative and additive correction enable					

4.77.9 Engine roughness components calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DRV0_ER	V/O	80000000... 7FFFFFFFH	-512000000... 511999999.7	0.2384186	[µs]
Engine roughness static component					
DRV1_ER	V/O	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Engine roughness dynamic component					
DRV2_ER	V	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Engine roughness curvature component					
DRV2_MMV_ER	V	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Averaged engine roughness curvature component					
ER_RAW	V/O	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Engine roughness without curvature component					

Input data:

SEG_T_COR	SEG_T_COR_BUF[NC_SIZE_SEG_T_COR_BUF]	NC_CYL_NR	FAC_DRV2_ER
CRLC_DRV2_ER			

General information :

The engine roughness calculation needs three individual components: a static component (DRV0_ER), a dynamic component (DRV1_ER) and a curvature component (DRV2_ER). These component definitions are specific to the engine cylinder number (see following related subchapters).

DRV2_MMV_ER is the filtered curvature component correction (DRV2_ER) who allows adding an artificial positive offset on ER value in case of dynamic curvature effects. This feature had a safety effect against wrong detection in case of segment oscillations.

ER_RAW is the engine roughness value without curvature component (DRV2_MMV_ER), used by cylinder balancing and engine warm up monitoring modules.

Application conditions:

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *on Deactivation condition event*

DRV0_ER = DRV1_ER = DRV2_ER = DRV2_MMV_ER = ER_RAW = 0

Recurrence: Segment

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Formula section:

Engine roughness static component:

DRV0_ER(n) = NC_CYL_NR * [SEG_T_COR(n) - SEG_T_COR(n+1)]

Engine roughness dynamic and curvature components:

DRV1_ER and DRV2_ER components are engine type specific.

3 Cylinder Engine

#If NC_CYL_NR = 3

DRV1_ER(n) = SEG_T_COR(n-2) - SEG_T_COR(n+1)

DRV2_ER(n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-3) - SEG_T_COR(n)] - [SEG_T_COR(n) - SEG_T_COR(n+3)])

#EndIf

4 Cylinder Engine

#If NC_CYL_NR = 4

DRV1_ER (n) = SEG_T_COR(n-2) - SEG_T_COR(n+2)

DRV2_ER (n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-4)- SEG_T_COR(n)] - [SEG_T_COR(n) -SEG_T_COR(n+4)])

#EndIf

5 Cylinder Engine

#If NC_CYL_NR = 5

DRV1_ER (n) = SEG_T_COR(n-2) - SEG_T_COR(n+3)

DRV2_ER (n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-5)- SEG_T_COR(n)] - [SEG_T_COR(n)-SEG_T_COR(n+5)])

#EndIf

6 Cylinder Engine

#If NC_CYL_NR = 6

DRV1_ER (n) = SEG_T_COR(n-3) - SEG_T_COR(n+3)

DRV2_ER (n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-3)- SEG_T_COR(n)] - [SEG_T_COR(n)-SEG_T_COR(n+3)])

#EndIf

8 Cylinder Engine

#If NC_CYL_NR = 8

DRV1_ER (n) = SEG_T_COR(n-4) - SEG_T_COR(n+4)

DRV2_ER (n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-4)- SEG_T_COR(n)] - [SEG_T_COR(n) -SEG_T_COR(n+4)])

#EndIf

10 Cylinder Engine

#If NC_CYL_NR = 10

DRV1_ER (n) = SEG_T_COR(n-5) - SEG_T_COR(n+5)

DRV2_ER (n) = FAC_DRV2_ER *

max(0,[SEG_T_COR(n-5)- SEG_T_COR(n)] - [SEG_T_COR(n)-SEG_T_COR(n+5)])

#EndIf

12 Cylinder Engine

#If NC_CYL_NR = 12

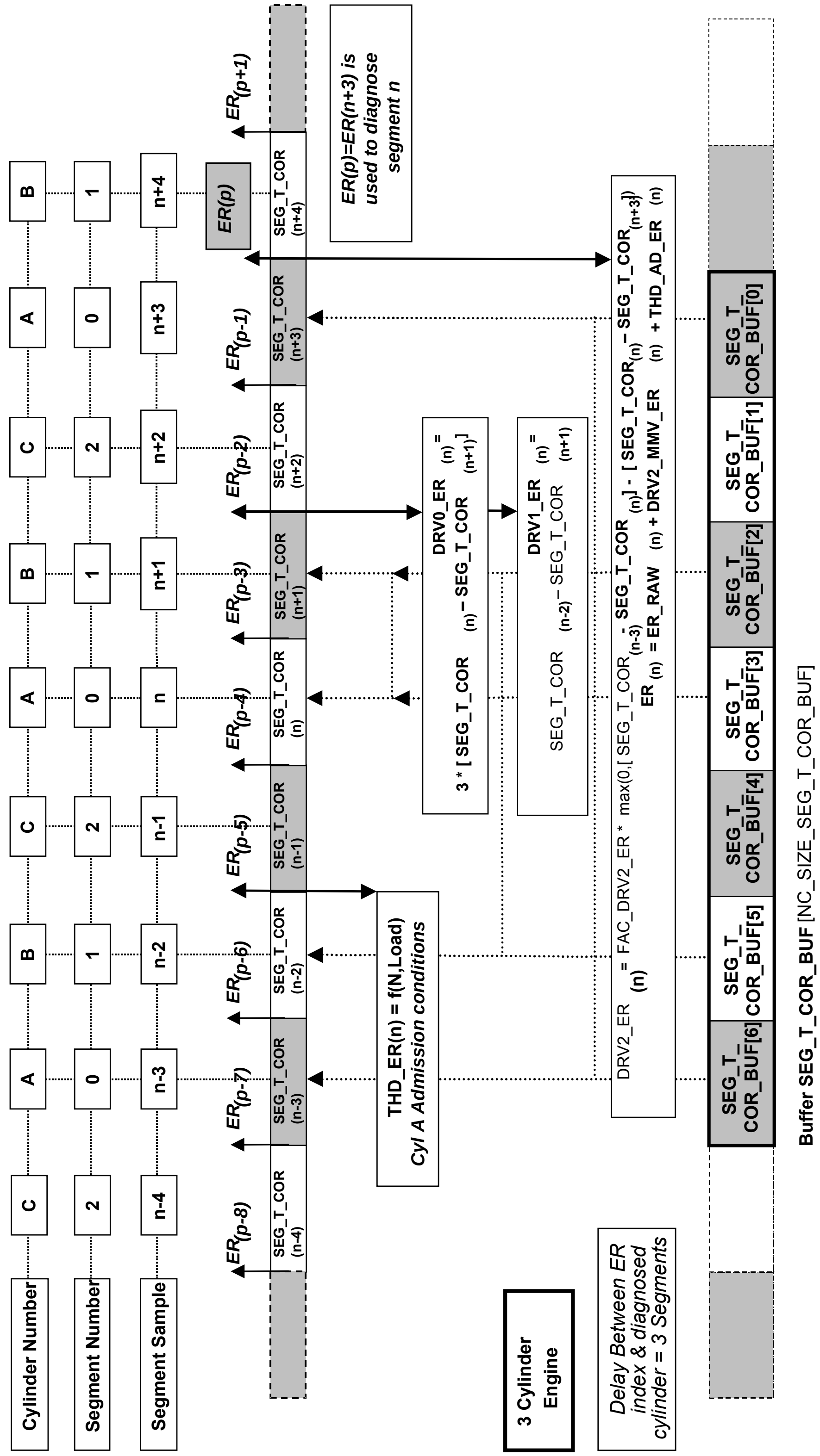
DRV1_ER (n) = SEG_T_COR(n-6) - SEG_T_COR(n+6)

DRV2_ER (n) = FAC_DRV2_ER *

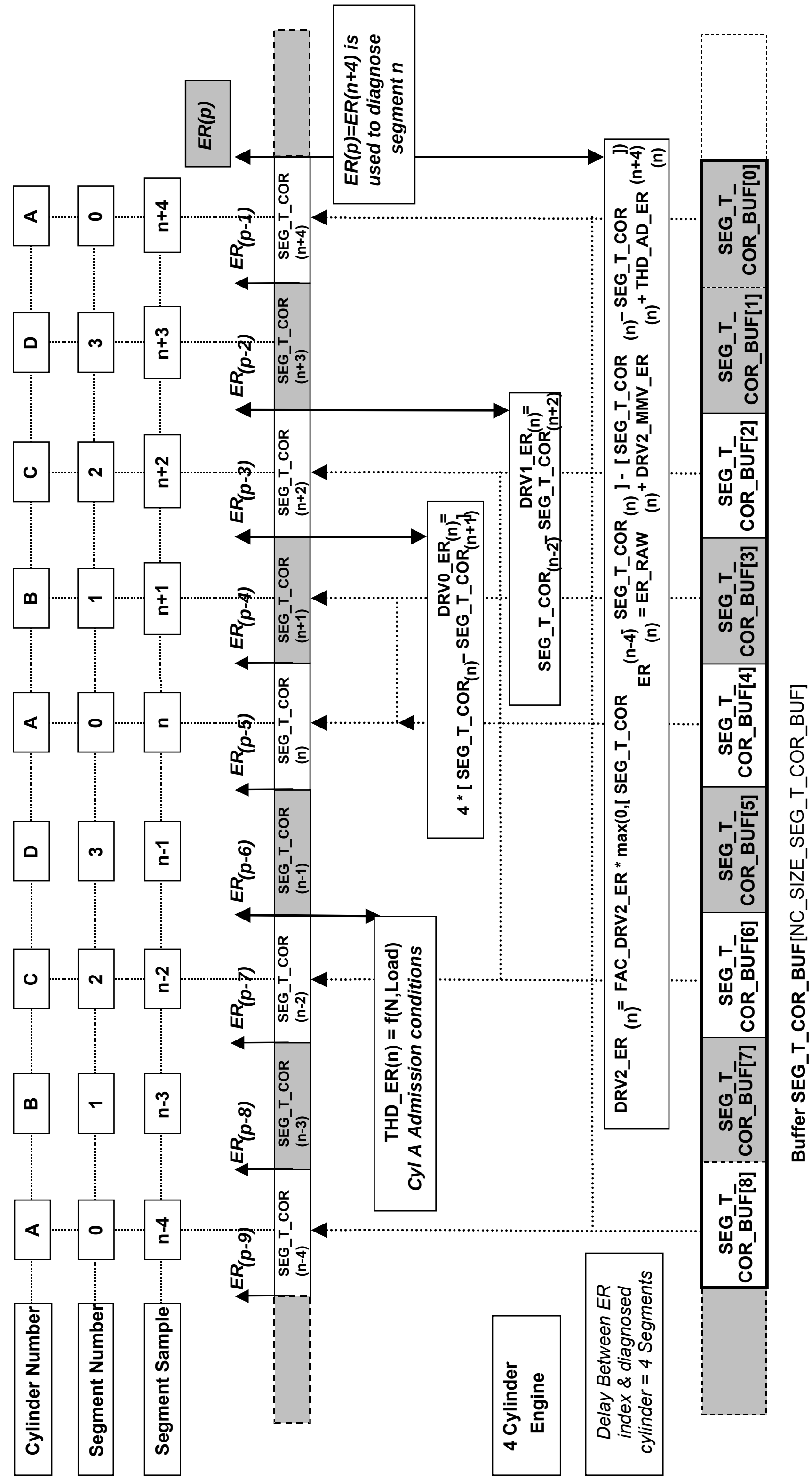
max(0,[SEG_T_COR(n-6)- SEG_T_COR(n)] - [SEG_T_COR(n)-SEG_T_COR(n+6)])

#EndIf

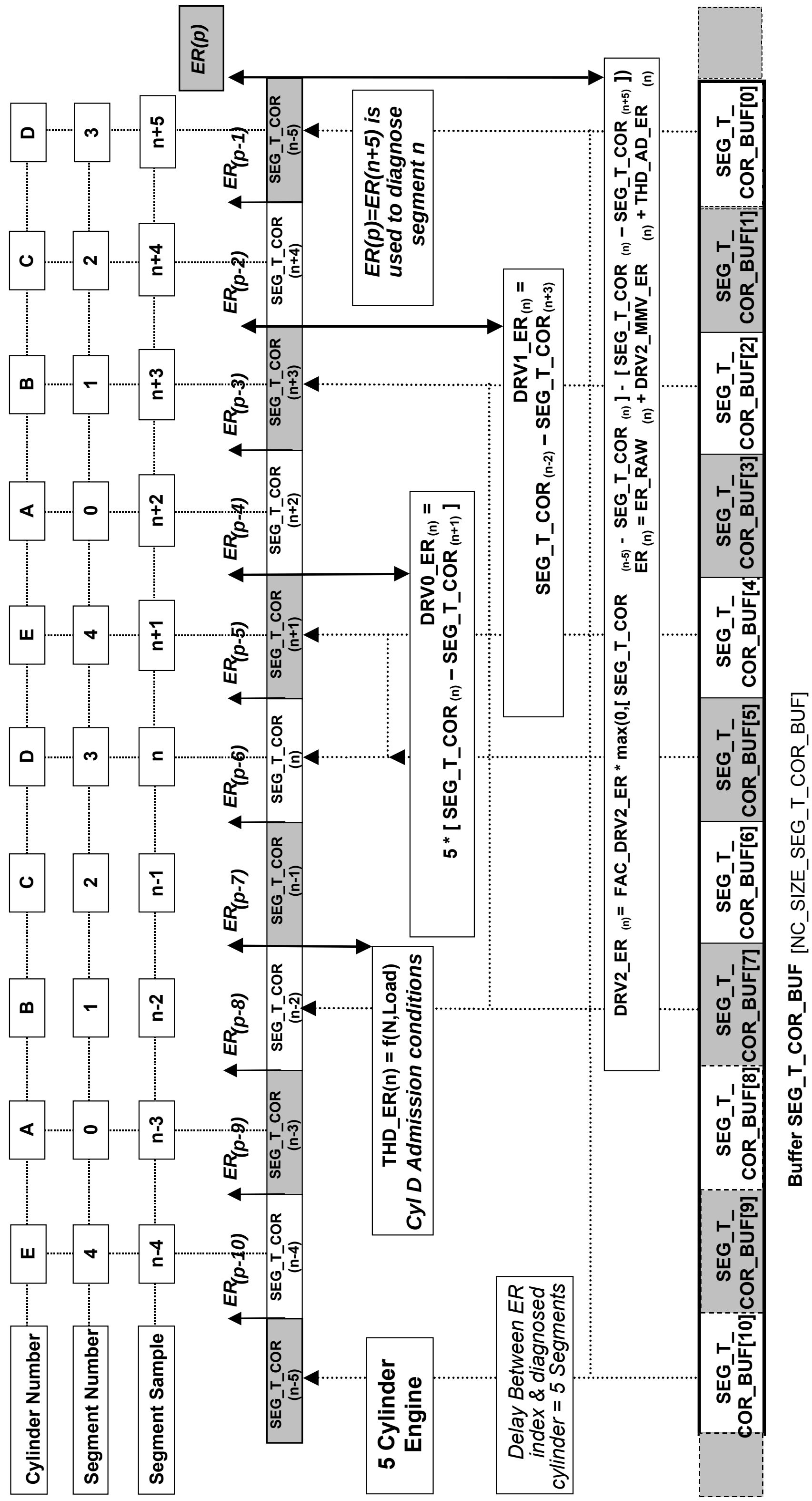
Implant scheme for 3 cylinder engine:



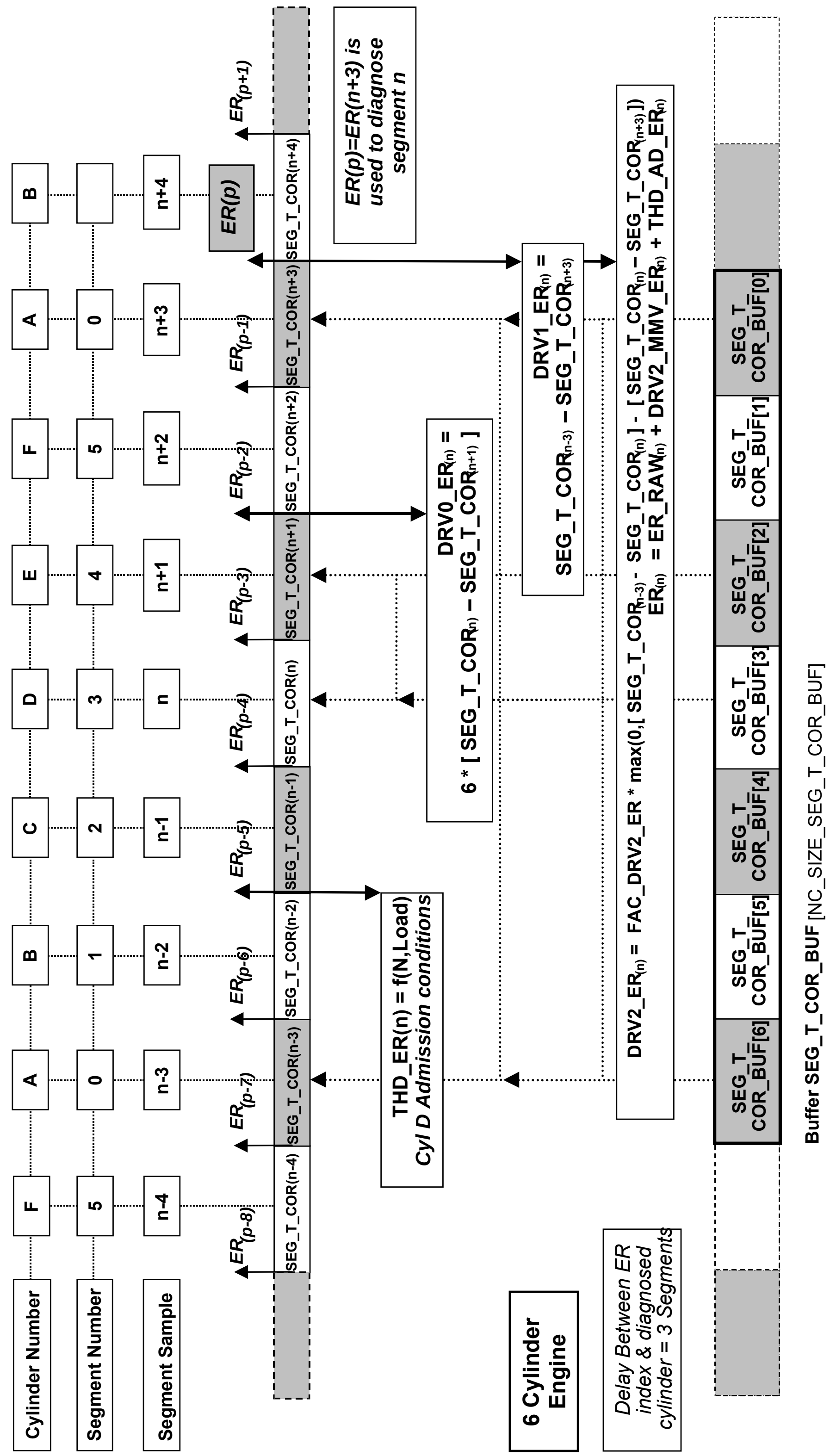
Implant scheme for 4 cylinder engine:



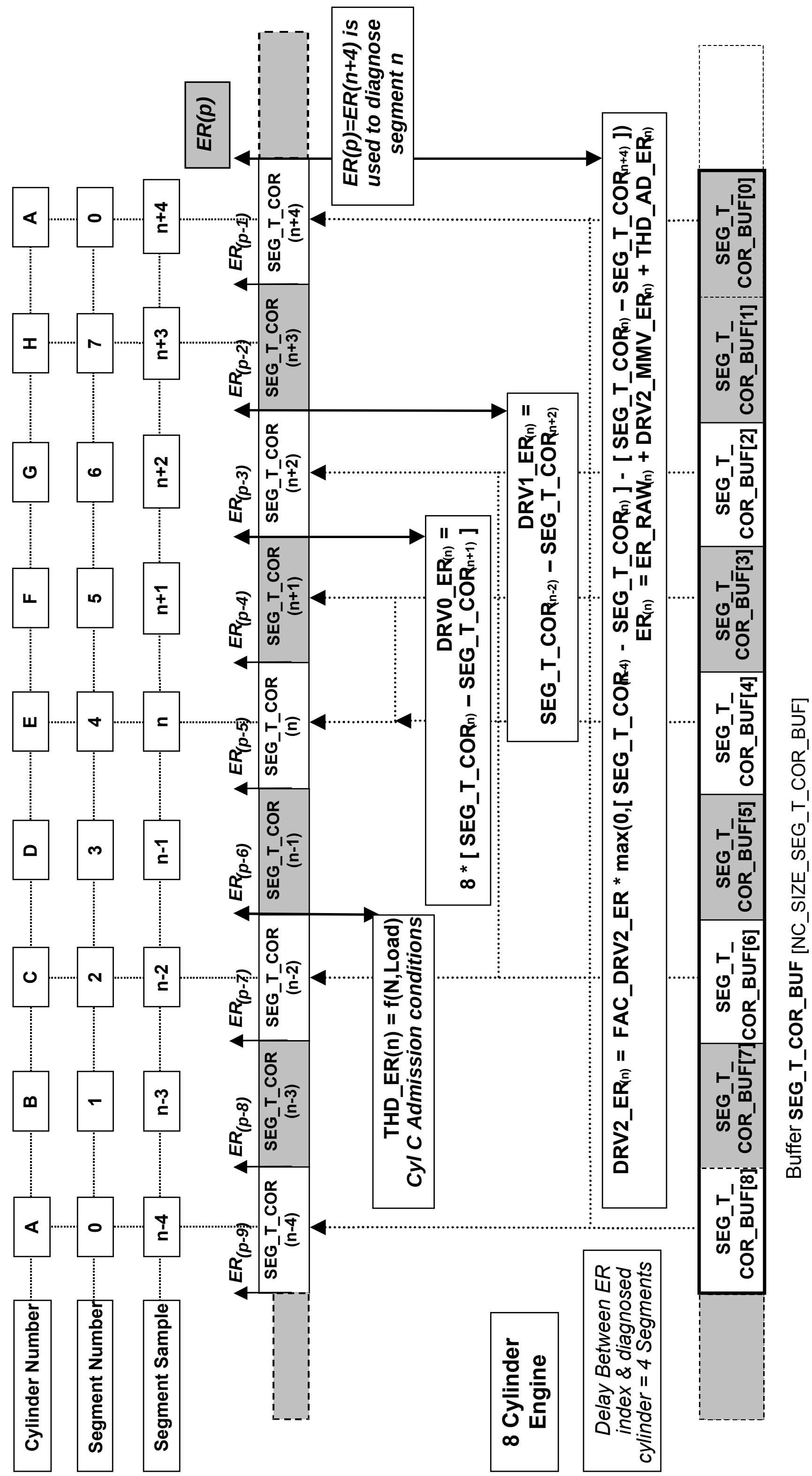
Implant scheme for 5 cylinder engine:



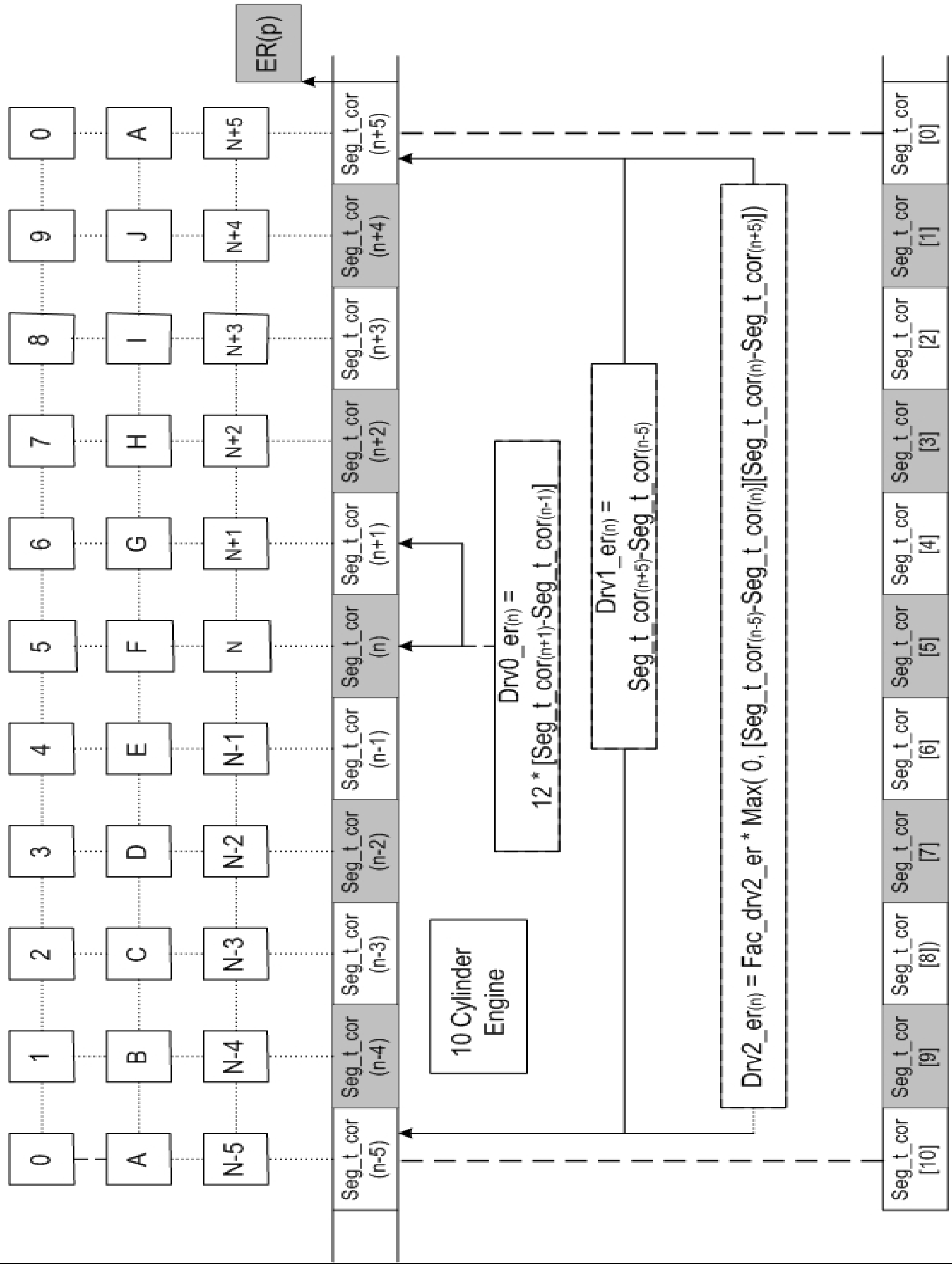
Implant scheme for 6 cylinder engine:



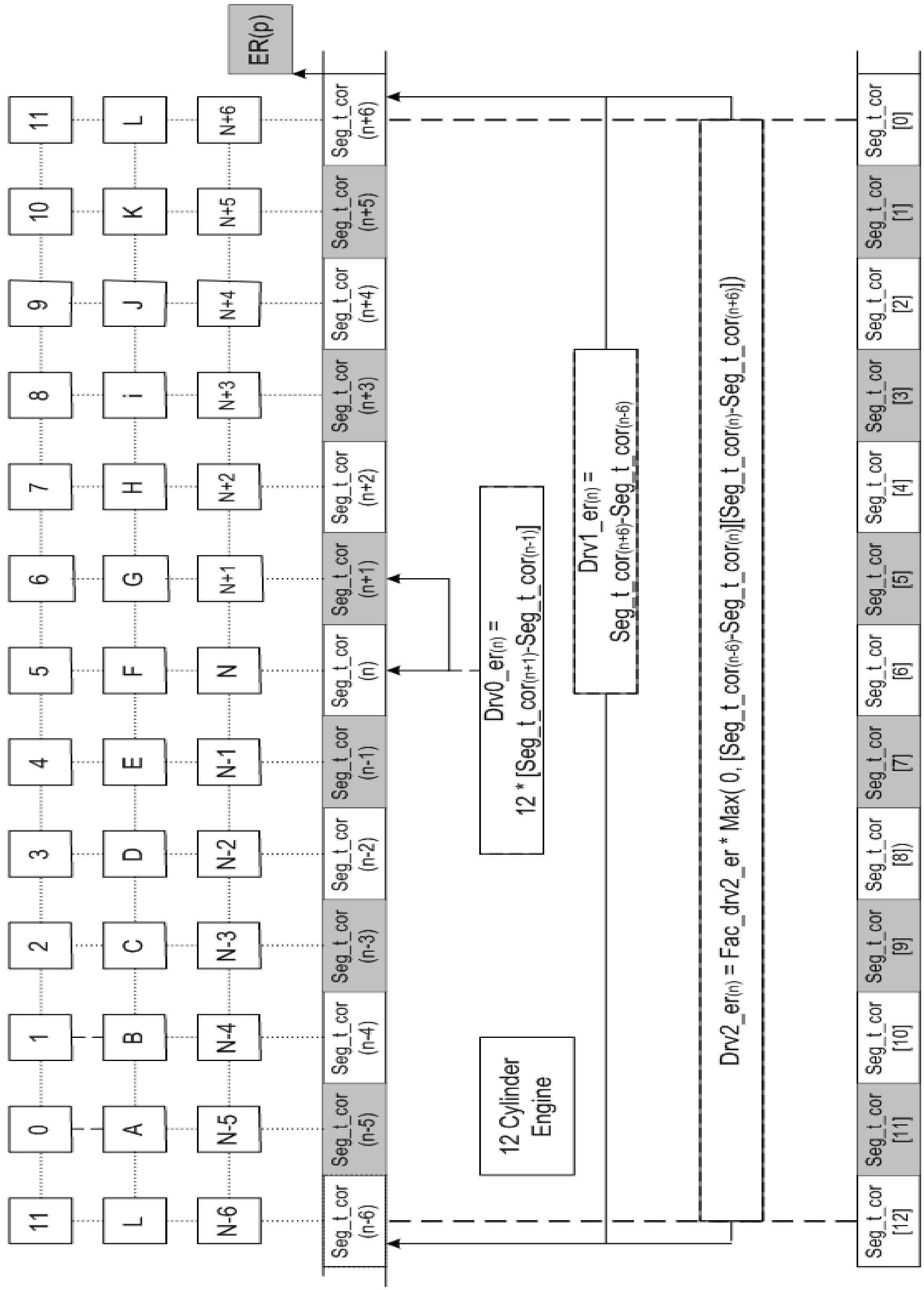
Implant scheme for 8 cylinder engine:



Implant scheme for 10 cylinder engine:



Implant scheme for 12 cylinder engine:



Engine roughness filtered curvature component:

$DRV2\_MMV\_ER_{(n)} = DRV2\_MMV\_ER_{(n-1)} + CRLC\_DRV2\_ER * (DRV2\_ER_{(n)} - DRV2\_MMV\_ER_{(n-1)})$

Engine roughness value without curvature component:

#If NLC_CONF_GAIN_ADD_ER = 1

ER_RAW = FAC_GAIN_ER[SEG_NR_ER] * (DRV0_ER - DRV1_ER - FAC_ADD_ER[SEG_NR_ER])

#Else

ER_RAW = DRV0_ER - DRV1_ER

#EndIf

4.77.10 Engine roughness values determination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ER	V/O	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Engine roughness					
ER_CYL[NC_CYL_NR]	V/O	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Cylinder normalised engine roughness values					
ER_STND	V/O	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Normalised engine roughness					
ER_STND_CYL[NC_CYL_NR]	V	FF800000 ...7FFFFFH	-2000000... 1999999.7	0.2384186	[µs]
Cylinder normalised engine roughness values					

Input data:

THD_AD_ER	SEG_T_COR_BUF[NC_SIZE_SEG_T_COR_BUF]	N	C_SEG_DLY_ER

Description:

ER is the complete engine roughness value including all ER components (DRV0_ER, DRV1_ER & DRV2_MMV_ER), specified & used by misfire detection module.

ER_STND is the normalised engine roughness value versus engine speed that could be amplified by a gain C_FAC_SCA_ER_STND. This is the nominal engine roughness value divided by the corresponding cubic corrected ER segment time $(SEG\_T\_COR\_BUF[C\_SEG\_DLY\_ER])^3$. The cubic segment time is calculated by using a mantisse and an exponent.

ER & ER_STND calculation formula are generic to all engine type.

A positive offset (THD_AD_ER) is added to ER components as long as the ER adaptive learning process hasn't been executed. This, in a way to avoid wrong misfire detection in engine operation areas where crankshaft decelerations due to a misfire are very close to ER noise due to the flywheel mechanical tolerances.

This offset decreases as the ER adaptive learning progress. At the end of the learning process, this offset is null *(see next chapter ER adaptive values learning process for more informations)*.

Once the ER fast adaptive learning stage has been executed, these flywheel mechanical tolerances are sufficiently reduced to able to detect misfire in full legal range without adding a positive offset.

ER_CYL_x and ER_STND_CYL_x are the engine roughness values obtained for corresponding cylinder x, these values are switched according SEG_NR_ER segment reference.

Application conditions:

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *on Deactivation condition event*

ER = 0

ER_STND = 0

For x = 0 to NC_CYL_NR-1

ER_CYL[x] = 0

ER_STND_CYL[x] = 0

EndFor

Recurrence: Segment

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Formula section:

Engine roughness:

ER = ER_RAW + DRV2_MMV_ER + THD_AD_ER

ER_CYL[SEG_NR_ER] = ER

Normalised Engine roughness versus engine speed:

If N < C_N_MAX_ER_STND

Then

ER_STND = 2 $^{\text{(C\_FAC\_SCA\_ER\_STND)}}$ * ER / (SEG_T_COR_BUF[C_SEG_DLY_ER])^3

// for the calculation, be carefull with the order of the calculation (risk to loose information with the division)

ER_STND_CYL[SEG_NR_ER] = ER_STND

Else

ER_STND = 0

ER_STND_CYL[SEG_NR_ER] = 0

EndIf

Note: if C_SEG_DLY_ER >= NC_SIZE_SEG_T_COR_BUF, SEG_T_COR_BUF[C_SEG_DLY_ER] data is limited to the last cell of the buffer.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_SCA_ER_STND	1	80...7FH	-128...127	1	[-]
Scaling factor used for normalised engine roughness calculation					
C_N_MAX_ER_STND	1	0...1FE0H	0...8160	1	[rpm]
Maximum engine speed to compute normalised engine roughness					

4.78 Engine roughness calculation - Application incidences

4.78.1 Inhibition of engine roughness adaptive process - Application specific

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_REQ_SEG_AD_MMV_ER_APP	V/O	0...1H	0...1	1	-
Flag indicating request to reload stored segment adaptive values					
SEG_AD_MMV_ER_APP[NC_CYL_NR]	V/O	8000H...7FFFH	-7.8125...7.8123	0.000238	‰
ER filtered adaptive values used for reload (for each segment) if segment adaptive process not valid					
LV_INH_APP_ER_AD	V/O	0...1H	0...1	1	-
ER segment adaptive process inhibition flag relative application specific conditions					

Input data:

LV_ENA_SEG_T_MES	LV_AT	GR_MT	GR_AT

FUNCTION DESCRIPTION:

Engine roughness segments adaptive process is inhibited when one of the following system conditions occurs. See application specific accessories that could have an impact on the transmission behaviour (CVT, Automatic gear shifting, specific gear ratios...)

Update rate: Engine roughness segment

Application conditions:

Activation: LV_ENA_SEG_T_MES = 1

Deactivation: LV_ENA_SEG_T_MES = 0

Initialisation: at ECU reset and Engine Stop

LV_INH_APP_ER_AD = 0

LV_REQ_SEG_AD_MMV_ER_APP = 0

SEG_AD_MMV_ER_APP[NC_CYL_NR] = 0

Formula section:

If (LV_AT = 1 **And** C_GR_INH_ER_AD_AT[GR_AT] = 1)

Or (LV_AT = 0 **And** C_GR_INH_ER_AD_MT[GR_MT] = 1)

Then LV_INH_APP_ER_AD = 1

Else LV_INH_APP_ER_AD = 0

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_GR_INH_ER_AD_AT	1	0...FF	0...255	1	-
Bitfield used to inhibit engine roughness adaptive according current AT gear ratio (Gear corresponding bit = 1, ER_AD inhibited)					
C_GR_INH_ER_AD_MT	1	0...FF	0...255	1	-
Bitfield used to inhabit engine roughness adaptive according current MT gear ratio (Gear corresponding bit = 1, ER_AD inhibited)					

4.78.2 Engine roughness adaptive process inhibition related to OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_OBD_ER_AD	V/O	0...1H	0...1	1	-
ER segment adaptive process inhibition flag linked to OBD I errors					

Input data:

LV_ERR_CRK	LV_ERR_CAM	LV_ERR_MAF	LV_ERR_TPS
LV_ENA_SEG_T_MES	LV_ERR_TOIL	LV_ERR_CPS	LV_ERR_IV[NC_CYL_NR]
LV_ERR_MEC_CPS			

FUNCTION DESCRIPTION:

Engine roughness segments adaptive process is inhibited when one of the following OBD I errors occurs.

Update rate: Engine roughness segment

Application conditions:

Activation: LV_ENA_SEG_T_MES = 1

Deactivation: LV_ENA_SEG_T_MES = 0

Initialisation: at ECU reset and Engine Stop

LV_INH_OBD_ER_AD = 0

Formula section:

```
If      LV_ERR_CRK = 1                  // Crankshaft signal error
   Or   LV_ERR_CAM = 1                  // Camshaft signal error
   Or   LV_ERR_TPS = 1                  // Throttle position sensor error
   Or   LV_ERR_MAF = 1                  // Air mass meter signal error
   Or   LV_ERR_MAP                      // Manifold absolute air pressure sensor
   Or   LV_ERR_TOIL        = 1          // oil temperature signal error
   Or   LV_ERR_CPS         = 1          // CPS electrical error
   Or   LV_ERR_IV[NC_CYL_NR] =1         // IV error
   Or   LV_ERR_MEC_CPS = 1              // CPS mechanical error

Then    LV_INH_OBD_ER_AD = 1
Else    LV_INH_OBD_ER_AD = 0
EndIf
```

4.78.3 End of line & After sale service request to reset ER segment adaptive values

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_SEG_AD_RST_ER_EOL	V/O	00H...01H	0...1	1	-
EOL specific request to reset ER segment adaptive values					

Input data:

LV_ENA_SEG_T_MES			

General information:

During end of line plant and/or for After sale services, a request is generated via communication protocol, to reset engine roughness segment adaptive values & learning process *(see project communication protocol).*

Update rate: Engine roughness segment

Application conditions:

Activation: LV_ENA_SEG_T_MES = 1

Deactivation: LV_ENA_SEG_T_MES = 0

Initialisation: at ECU reset and Engine Stop

LV_SEG_AD_RST_ER_EOL = 0

Description:

LV_SEG_AD_RST_ER_EOL = 0 *// No specific request*

4.78.4 Engine roughness DRV2 components filter & gain determination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CRLC_DRV2_ER	V/O	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation factor for engine roughness curvature component calculation					
FAC_DRV2_ER	V/O	0...3FCH	0...4	3.9216e-3	[-]
Weighting factor for engine roughness curvative component calculation (correction amount)					

Input data:

N_32	LV_AT	LOAD_MIS	CTR_T_ZDLY_MIS
TCO	LV_IS		

FUNCTION DESCRIPTION:

This function allows to determinate the coefficients relative to DRV2 components of the engine roughness, according different configurations:

If used *(see project/customer legal requirements concerning misfire monitoring after engine start)*, during zero delay misfire monitoring activation period, specific gain & filter coefficient can be apply to DRV2 component to be able to detect misfire just after engine start *(dynamic correction modified).*

Else, FAC_DRV2_ER & CRLC_DRV2_ER are defined according transmission type *(AT/MT)*.

Application conditions:

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *on Deactivation condition event*

FAC_DRV2_ER = IP_FAC_DRV2_ZDLY_ER(TCO)

CRLC_DRV2_ER = C_CRLC_DRV2_ZDLY_ER

Recurrence: Segment

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Formula section:

If(1) CTR_T_ZDLY_MIS > C_T_DRV2_ZDLY_ER

Then(1) *// Zero delay misfire monitoring activation period*

FAC_DRV2_ER = IP_FAC_DRV2_ZDLY_ER(TCO)

CRLC_DRV2_ER = C_CRLC_DRV2_ZDLY_ER

ElseIf(1) LV_AT = 1

// Automatic transmission vehicle

If(2) LV_IS = 1

And N_32 < C_N_32_MAX_DRV2_ER_IS

Then(2) FAC_DRV2_ER = IP_FAC_DRV2_ER_IS_AT(N, LOAD_MIS)

Else(2) FAC_DRV2_ER = IP_FAC_DRV2_ER_AT(N_32, LOAD_MIS)

EndIf(2)

CRLC_DRV2_ER = IP_CRLC_DRV2_ER_AT(N_32)

Else(1)

// Manual transmission vehicle

If(2) LV_IS = 1

And N_32 < C_N_32_MAX_DRV2_ER_IS

Then(2) FAC_DRV2_ER = IP_FAC_DRV2_ER_IS_MT(N, LOAD_MIS)

Else(2) FAC_DRV2_ER = IP_FAC_DRV2_ER_MT(N_32, LOAD_MIS)

EndIf(2)

CRLC_DRV2_ER = IP_CRLC_DRV2_ER_MT(N_32)

EndIf(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_DRV2_ZDLY_ER	1	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation factor for engine roughness curvature component calculation after start					
C_T_DRV2_ZDLY_ER	1	0...FFFFH	0...655.35	0.01	[s]
Delay for specific curvative component calculation during after start					
C_N_32_MAX_DRV2_ER_IS	1	0...FFH	0...8160	32	[rpm]
Delay for specific curvative component calculation during after start					
IP_FAC_DRV2_ER_AT	6*4	0...3FCH	0...4	3.9216e-3	[-]
LDP_N_32_IP_FAC_N_DRV2_ER_AT	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_MIS_IP_FAC_DRV2_ER_AT	4	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for DRV2 component calculation - AT vehicle					
IP_FAC_DRV2_ER_IS_AT	4*4	0...3FCH	0...4	3.9216e-3	[-]
LDP_N_IP_FAC_N_DRV2_ER_IS_AT	4	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_IP_FAC_DRV2_ER_IS_AT	4	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for DRV2 component calculation - AT vehicle in idle speed					
IP_CRLC_DRV2_ER_AT	9	0...FFH	0...0.99609	3.9063e-3	[-]
LDP_N_32_IP_CRLC_DRV2_ER_AT	9	0...FFH	0...8160	32	[rpm]
Correlation factor for engine roughness curvature component calculation (correction duration) - AT vehicle					
IP_FAC_DRV2_ER_MT	6*4	0...3FCH	0...4	3.9216e-3	[-]
LDP_N_32_IP_FAC_N_DRV2_ER_MT	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_MIS_IP_FAC_DRV2_ER_MT	4	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for DRV2 component calculation - MT vehicle					
IP_FAC_DRV2_ER_IS_MT	4*4	0...3FCH	0...4	3.9216e-3	[-]
LDP_N_IP_FAC_N_DRV2_ER_IS_MT	4	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_IP_FAC_DRV2_ER_IS_MT	4	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for DRV2 component calculation - MT vehicle in idle speed					
IP_CRLC_DRV2_ER_MT	9	0...FFH	0...0.99609	3.9063e-3	[-]
LDP_N_32_IP_CRLC_DRV2_ER_MT	9	0...FFH	0...8160	32	[rpm]
Correlation factor for engine roughness curvature component calculation (correction duration) - MT vehicle					
IP_FAC_DRV2_ZDLY_ER	6	0...3FCH	0...4	3.9216e-3	[-]
LDP_TCO_IP_FAC_DRV2_ZDLY_ER	6	0...FEH	-48...142.5	0.75	[°C]
Weighting factor for engine roughness curvative component calculation (correction amount) after start					

4.79 Engine roughness adaptive learning process

General information:

To reduce systematic noises generated by target wheel mechanical tolerances, an adaptive learning process is required to improve the signal noise ratio on engine roughness variable and to avoid wrong misfire detection at high engine speed / low load.

ER segment adaptive values learning process is based on ER segments timing observation according a geometrical reference (ER segment time 0, SEG_NR_MES = 0).

During an engine deceleration in fuel cut-off mode on all cylinders (LV_INJ_CUT, no engine combustion influences), due to crankshaft inertia, engine speed can be considered in a short horizon as constant, thus on one engine cycle (720°Crk) timing dissimilarities between segment reference and others will be observed. For longer horizon observation, a deceleration is applied according current segment position in the engine cycle.

When all adaptation conditions fulfilled (smooth deceleration in fuel cut-off conditions), an adaptive values learning process is performed every **720° crank angle**, starting with **segment 0** *(adaptive process reference).*

If there is no cylinder specific mechanical damage or crankshaft torsion/flexion, the adaptation values for cylinders with the same mechanical segment will be approximately identical :

3 cylinder engine	4 cylinder engine	5 cylinder engine	6 cylinder engine	8 cylinder engine	10 cylinder engine	12 cylinder engine
SEG_AD_ER_0 *(geometrically singular)*	SEG_AD_ER_0 ≅ SEG_AD_ER_2	SEG_AD_ER_0 *(geometrically singular)*	SEG_AD_ER_0 ≅ SEG_AD_ER_3	SEG_AD_ER_0 ≅ SEG_AD_ER_4	SEG_AD_ER_0 ≅ SEG_AD_ER_5	SEG_AD_ER_0 ≅ SEG_AD_ER_6
SEG_AD_ER_1 *(geometrically singular)*	SEG_AD_ER_1 ≅ SEG_AD_ER_3	SEG_AD_ER_1 *(geometrically singular)*	SEG_AD_ER_1 ≅ SEG_AD_ER_4	SEG_AD_ER_1 ≅ SEG_AD_ER_5	SEG_AD_ER_1 ≅ SEG_AD_ER_6	SEG_AD_ER_1 ≅ SEG_AD_ER_7
SEG_AD_ER_2 *(geometrically singular)*		SEG_AD_ER_2 *(geometrically singular)*	SEG_AD_ER_2 ≅ SEG_AD_ER_5	SEG_AD_ER_2 ≅ SEG_AD_ER_6	SEG_AD_ER_2 ≅ SEG_AD_ER_7	SEG_AD_ER_2 ≅ SEG_AD_ER_8
		SEG_AD_ER_3 *(geometrically singular)*		SEG_AD_ER_3 ≅ SEG_AD_ER_7	SEG_AD_ER_3 ≅ SEG_AD_ER_8	SEG_AD_ER_3 ≅ SEG_AD_ER_9
		SEG_AD_ER_4 *(geometrically singular)*			SEG_AD_ER_4 ≅ SEG_AD_ER_9	SEG_AD_ER_4 ≅ SEG_AD_ER_10
						SEG_AD_ER_5 ≅ SEG_AD_ER_11
Segments length : 240°Crk	*Segments length : 180°Crk*	*Segments length : 144°Crk*	*Segments length : 120°Crk*	*Segments length : 90°Crk*	*Segments length : 72°Crk*	*Segments length : 60°Crk*

4.79.1 Fade out conditions of engine roughness segment adaptive process

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
SEG_AD_FDOUT_ER_CDN	V	0...FFH	0...255	1	[-]
Segment adaptation fade out conditions carrier byte (before masking)					
SEG_AD_FDOUT_ER_CDN_NR	V	0...8H	0...8	1	[-]
Segment adaptation fade out conditions number (after masking)					
LV_SEG_AD_FDOUT_ER	V	0...1H	0...1	1	[-]
Boolean for segment adaptation fade out condition					
LV_INH_MAF_DIF_ER_AD	V	0...1H	0...1	1	[-]
ER adaptive process fade out request flag due to MAF transient condition					
LV_INH_LOAD_GRD_MAX_ER_AD	V	0...1H	0...1	1	[-]
ER adaptive process fade out request flag due to engine load transient condition					
LV_INH_TPS_GRD_ER_AD	V	0...1H	0...1	1	[-]
ER adaptive process fade out request flag due to TPS transient condition					
LV_INH_ACC_ER_AD	V	0...1H	0...1	1	[-]
ER adaptive process fade out request flag due to Air conditioner activation					

Input data:

LV_ACCOUT_RLY	MAF	LOAD_MIS	LV_ENA_SEG_T_MES
TPS_GRD	TPS_AV	LV_STATE_RR	LOAD_GRD_MIS
MAF_DIF	LV_INH_APP_ER_AD	LV_INH_OBD_ER_AD	

FUNCTION DESCRIPTION:

Description:

LV_SEG_AD_FDOUT_ER combines all system conditions to generate a fade out condition as input for the generic segment adaptation process.

Update rate: segment occurrence for conditions & data process

10ms for free running decounters

Application conditions:

Activation: *LV_ENA_SEG_T_MES = 1*

Deactivation: *LV_ENA_SEG_T_MES = 0*

Action on deactivation event ***Or*** *ECU reset:*

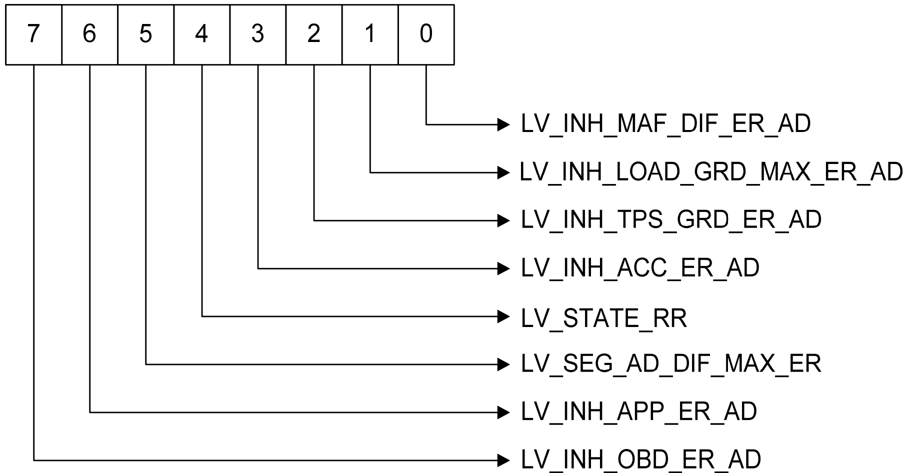
LV_SEG_AD_FDOUT_ER = 1

SEG_AD_FDOUT_ER_CDN = 0xFF

SEG_AD_FDOUT_ER_CDN_NR = 8

Definition of segment adaptation fade out conditions carrier:

SEG_AD_FDOUT_ER_CDN : *Carrier used for fade-out conditions merge*



4.79.1.1 Configurable Fade-out Management

Application conditions:

The Fade out management of the segment adaptive process can be configured with a bitfield mask calibration that allow to take in account or not some conditions set in the carrier.

If the corresponding bit in the SEG_AD_FDOUT_ER_CDN carrier structure is set to 0 in C_MASK_SEG_AD_FDOUT_ER calibration, then the corresponding condition will not fade-out the segment adaptive process via LV_SEG_AD_FDOUT_ER.

The number of fade-out conditions not masked by this calibration is set in SEG_AD_FDOUT_ER_CDN_NR.

Formula section:

SEG_AD_FDOUT_ER_CDN_NR =

sum (SEG_AD_FDOUT_ER_CDN **&** C_MASK_SEG_AD_FDOUT_ER)

! (bitfield operation) !

If SEG_AD_FDOUT_ER_CDN_NR ≠ 0

Then LV_SEG_AD_FDOUT_ER = 1

Else LV_SEG_AD_FDOUT_ER = 0

EndIf

4.79.1.2 Maximum air-mass gradient

Due to trailing throttle / acceleration transition problems it is necessary to disable ER segment learning process for a short period when the air-mass gradient per segment exceeds a calibration value.

Application conditions:

ER segment adaptation is disabled while the absolute of the air-mass gradient exceeds the value mentioned below:

After the last triggering of this segment adaptation is suppressed for a period of C_T_MAF_DIF_DLY_ER_AD.

Formula section:

If | MAF_DIF | > IP_MAF_DIF_MAX_ER_AD(MAF)

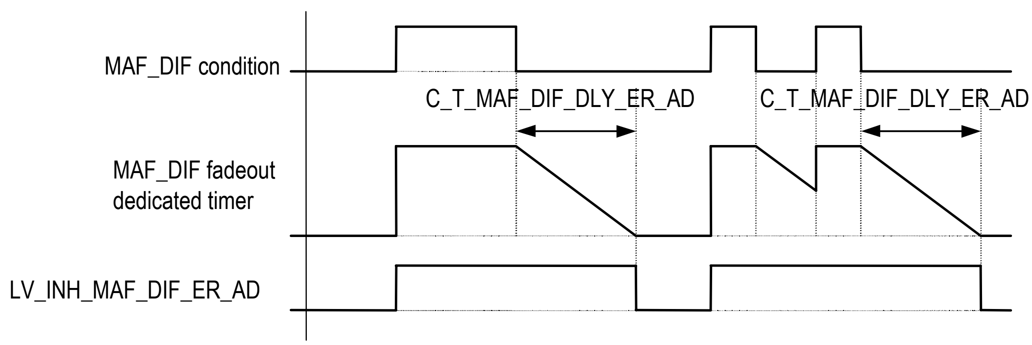
Then

After condition rising edge triggering, LV_INH_MAF_DIF_ER_AD flag is set to 1 as long as condition is true.

After condition falling edge triggering, LV_INH_MAF_DIF_ER_AD flag is hold to 1 for a period of C_T_MAF_DIF_DLY_ER_AD.

EndIf

Fade out behaviour summary:



4.79.1.3 Maximum actual load gradient

Due to general intervention of the torque model it's necessary to disable ER segment adaptation for a short period when the actual torque gradient exceeds a calibration value.

Application conditions:

ER segment adaptation is disabled while the absolute of the torque gradient – here LOAD_GRD_MIS the reference for Misfire detection process – exceeds the value mentioned below.

After the last triggering of this segment adaptation is suppressed for a period of C_T_LOAD_GRD_DLY_ER_AD.

Formula section:

If LOAD_GRD_MIS > IP_LOAD_GRD_ER_AD(LOAD_MIS)

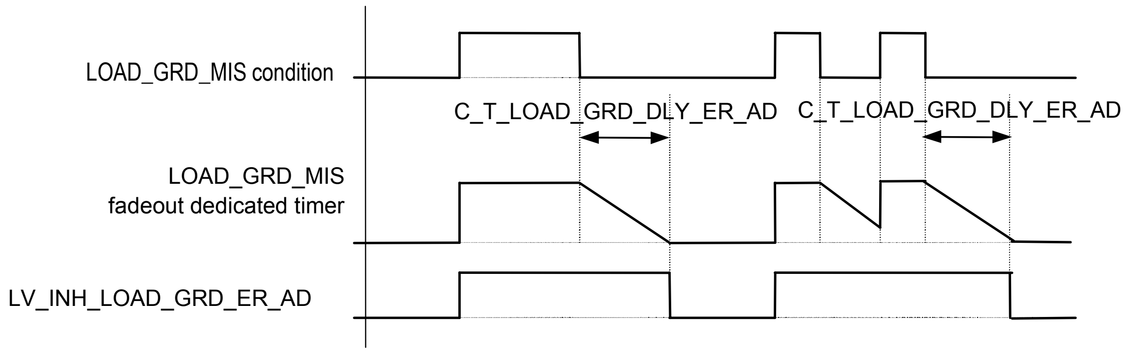
Then

After condition rising edge triggering, LV_INH_LOAD_GRD_MAX_ER_AD flag is set to 1 as long as condition is true.

After condition falling edge triggering, LV_INH_LOAD_GRD_MAX_ER_AD flag is hold to 1 for a period of C_T_LOAD_GRD_DLY_ER_AD.

EndIf

Fade out behaviour summary:



4.79.1.4 Maximum throttle gradient

Due to trailing throttle / acceleration transient problems at low load it is necessary to disable ER segment adaptation for a short period when the throttle gradient exceeds a calibration value.

Application conditions:

ER segment adaptation is disabled while the throttle position gradient exceeds the value mentioned below.

After the last triggering of this segment adaptation is suppressed for a period of C_T_TPS_GRD_DLY_ER_AD.

Formula section:

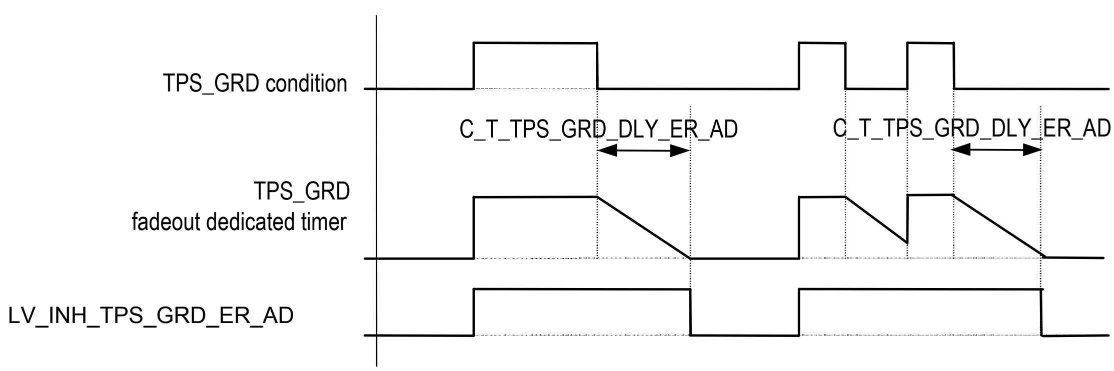
If TPS_GRD > IP_TPS_GRD_MAX_ER_AD(TP_AV)

Then

After condition rising edge triggering, LV_INH_TPS_GRD_ER_AD flag is set to 1 as long as condition is true.

After condition falling edge triggering, LV_INH_TPS_GRD_ER_AD flag is hold to 1 for a period of C_T_TPS_GRD_DLY_ER_AD.

EndIf



4.79.1.5 Air - conditioning compressor activation

When the air - conditioning compressor is switched on, an additional load is briefly applied to the engine.

This load jump can cause an ER segment period jump and crankshaft vibration.

Application conditions:

Engine roughness segment adaptation can be suppressed for the applicable constant period C_T_ACCOUT_DLY_ER_AD, starting if the air - conditioning compressor is switched on or off (LV_ACCOUT_RLY changes from 0 to 1 or from 1 to 0), during this period LV_INH_ACC_ER_AD is set to 1.

Formula section:

If LV_ACCOUT_RLY = 0 → 1
Or LV_ACCOUT_RLY = 1 → 0
Then LV_INH_ACC_ER_AD is set to 1 during C_T_ACCOUT_DLY_ER_AD
EndIf

4.79.1.6 Rough road condition active

Due to possible feedback from the driven wheels to the crankshaft it is necessary to fade out. ER segment adaptation when rough road condition is active (LV_STATE_RR = 1).

4.79.1.7 Engine roughness adaptive values on the same physical segment out of range

Due to possible feedback from Dual Mass Flywheel oscillations *(see transmission/crankshaft design)* in adaptive process engine speed range, fluctuations on ER adaptive values of identical physical segments can induce dissimilarity in ER segment correction. LV_SEG_AD_DIF_MAX_ER indicates that the engine roughness adaptive values difference on the same physical segment is out of range.

This flag is produced in the following chapter called: Engine roughness adaptive values difference on the same physical segment out of range.

4.79.1.8 Application specific inhibition request for engine roughness adaptive process

Due to possible accessories triggering specific to the application (CVT shifts...), an inhibition of the engine roughness adaptive process can be applied.

The inhibition is applied when LV_INH_APP_ER_AD = 1.

4.79.1.9 Inhibition request for engine roughness adaptive process linked to OBD errors

Due to possible troubles induced by OBD I sensors errors (CAM, CRK...), the engine roughness adaptive process can be inhibited by the flag LV_INH_OBD_ER_AD.

The inhibition is applied when LV_INH_OBD_ER_AD = 1.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_DIF_MAX_ER_AD	6	0...FFFFH	0...1389	0.0211948	[mg/stk]
LDP_MAF_IP_MAF_DIF_MAX_ER_AD	6	0...FFFFH	0...1389	0.0211948	[mg/stk]
Maximum air-mass gradient per SEG for fade out condition.					
IP_LOAD_GRD_ER_AD	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
LDP_LOAD_MIS_IP_LOAD_GRD_ER_AD	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Maximum actual torque gradient fade out condition.					
IP_TPS_GRD_MAX_ER_AD	6	0...FFH	0...2987.5	11.715686	[°TPS/s]
LDP_TPS_AV_IP_TPS_GRD_MAX_ER	6	0...3FFFH	0...119.5	7.2941e-3	[°TPS]
Maximum throttle gradient threshold for fade out condition.					
C_T_MAF_DIF_DLY_ER_AD	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum air-mass gradient has been detected.					
C_T_LOAD_GRD_DLY_ER_AD	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum actual torque gradient has been detected.					
C_T_TPS_GRD_DLY_ER_AD	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum throttle gradient has been detected.					
C_T_ACCOUT_DLY_ER_AD	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when air-conditioning compressor has been switched on or off					
C_MASK_SEG_AD_FDOUT_ER	1	0...FFH	0...255	1	[-]
Configuration mask for SEG_AD_FDOUT_ER_CDN fade out carrier structure					

4.79.2 End of line specific request for ER segment adaptive values learning process

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_SEG_AD_FDOUT_ER_EOL	V	0...1H	0...1	1	[-]
EOL specific request for ER segment adaptive learning process					

Input data:

LV_EOL_OBD	N_32	DRV1_ER	SEG_NR_MES
LV_INJ_CUT	LV_ENA_SEG_T_MES	LV_AT	

General information:

During end of line plant, specific conditions to activate ER adaptive process can be used. If OBD end of line process occurs, this request has priority on nominal conditions.

Update rate: ER segment occurrence

Application conditions:

Activation: *LV_ENA_SEG_T_MES = 1*

Deactivation: *LV_ENA_SEG_T_MES = 0*

*Action on deactivation event **Or** ECU reset:*

LV_SEG_AD_FDOUT_ER_EOL = 1

Description:

If LV_EOL_OBD = 1 *// OBD end of line process request*

And LV_INJ_CUT = 1 *(all cylinders shut-off)* during last C_NR_SEG_AD_ACT_MIN_ER_EOL segments since last SEG_NR_MES = 0

And { [LV_AT = 0 **And** N_32 **<** C_N_MAX_SEG_AD_ER_EOL_MT
And N_32 **>** C_N_MIN_SEG_AD_ER_EOL_MT]
Or [LV_AT = 1 **And** N_32 **<** C_N_MAX_SEG_AD_ER_EOL_AT
And N_32 **>** C_N_MIN_SEG_AD_ER_EOL_AT] }

And │ DRV1_ER │ **<** C_DRV1_MAX_SEG_AD_ER_EOL during C_NR_SEG_DRV1_MIN_ER_EOL segments

And LC_SEG_AD_ER_REQ_EOL = 1

Then *// Request in specific EOL range for segment adaptive process*

LV_SEG_AD_FDOUT_ER_EOL = 0

Else *// No specific request*

LV_SEG_AD_FDOUT_ER_EOL = 1

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_N_MAX_SEG_AD_ER_EOL_MT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning for EOL request mode in MT					
C_N_MIN_SEG_AD_ER_EOL_MT	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed threshold for crankshaft target wheel learning for EOL request mode in MT					
C_N_MAX_SEG_AD_ER_EOL_AT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning for EOL request mode in AT					
C_N_MIN_SEG_AD_ER_EOL_AT	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed threshold for crankshaft target wheel learning for EOL request mode in AT					
C_DRV1_MAX_SEG_AD_ER_EOL	1	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Maximum value of DRV1_ER (dynamic part of engine roughness) for crankshaft target wheel learning for EOL request mode					
C_NR_SEG_AD_ACT_MIN_ER_EOL	1	0...FFH	0...255	1	[-]
Minimum segments number in fuel cut-off mode before triggering ER segment adaptive learning process for EOL request mode					
C_NR_SEG_DRV1_MIN_ER_EOL	1	0...FFH	0...255	1	[-]
Minimum number of segments for engine roughness dynamic part for EOL request mode					
LC_SEG_AD_ER_REQ_EOL	1	0...1H	0...1	1	[-]
Specific EOL request enable flag for ER segment adaptive value learning process					

4.79.3 Engine roughness adaptive learning process management

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
THD_AD_ER	V/O	0...7FFFFFH	0...1999999.7	0.2384186	[µs]
Engine roughness threshold correction factor during segment adaptation learning process.					
DELTA_CRK_DIF_MAX_ER	V/S	0...FFFFH	0...999.98474	0.0152588	[°/oo]
Limitation of adaptation range (mech. Tolerance)					
CRLC_SEG_AD_ER	V	0...FFFFH	0...0.99998	0.0153e-3	[-]
ER segment adaptive values filtering coefficient					
CTR_SEG_AD_ER	V/S	0...FFFFH	0...65535	1	[-]
Counter of ER segment adaptation learning process					
LV_SEG_AD_AVL_ER	V/O/S	0...1H	0...1	1	[-]
Segment adaptation process state : no adaptive process executed (=0) fast adaptive process achieved (=1)					
LV_SEG_AD_RAW_ER	V	0...1H	0...1	1	[-]
ER segment adaptation process monitoring enable					
LV_SEG_AD_ER	V/O	0...1H	0...1	1	[-]
ER segment adaptation process enable					
LV_SEG_AD_N_VLD	V	0...1H	0...1	1	[-]
ER segment adaptation process enable vs N					

Input data:

LV_INH_OBD_ER_AD	LV_AT	N_32	DRV1_ER
LV_INJ_CUT	LV_ENA_SEG_T_MES	LV_ENA_ER	LV_SEG_AD_RST_ER_EOL
SEG_NR_MES	NC_CYL_NR	SEG_T_COR	

Description:

The adaptive process is managed in two phases:

- A fast adaptive process for end of line procedures (EOL), who allow a fast convergence to ER flywheel adaptive values.
- At the end of the fast adaptive process, a slow adaptive process who will monitor ER adaptive values during the engine lifetime.

An end of line specific filtering mode, to be used on EOL or special service request.

At the beginning of the segment adaptation learning process, it is necessary to upper the engine roughness value with an additional offset (THD_AD_ER), depending on the target wheel tolerances to be excepted.

The segment time adaptation is defined here on 3 different engine speed ranges (Low = _L, Middle = _MID and High = _H speed) to allow a specific crankshaft behaviour learning on each relevant engine speed area.

For instance, a flywheel with a ± 0.3°Crk mechanical tolerance on the ER segment corresponds to following segment drift:

Engine Type	ER Segment length	Induced Drift with ± 0.3°Crk tolerance on ER segment
3 cyl engine	240°Crk	2.5‰
4 cyl engine	180°Crk	3.33‰
5 cyl engine	144°Crk	4.17‰
6 cyl engine	120°Crk	5‰
8 cyl engine	90°Crk	6.68‰
10 cyl engine	72°Crk	8.33‰
12 cyl engine	60°Crk	10.00‰

The value by which the ER is increased at the beginning is gradually reduced to zero in a linear way until the 95% correction is obtained.

This number of steps is controlled by an internal adaptation counter CTR_SEG_AD_ER.

4.79.3.1 Engine roughness adaptive learning process

First time initialisation :

The ER segment adaptive values learning process is initialised when :

- ECU is initialised for the first time *(flash memory formatting)*
- A loss of adaptation values has been detected *(flash memory corrupted)*.
- A specific external tool request occurs *(use of **LV_SEG_AD_RST_ER_EOL**)*

// ER segment adaptive process control variables reseted

CTR_SEG_AD_ER = 0

DELTA_CRK_DIF_MAX_ER = NC_CYL_NR * 4 * C_CRK_DIF_MAX_ER

LV_SEG_AD_AVL_ER = 0

Update rate: ER segment task, every engine cycle when SEG_NR_MES = 0 ; SEG_T_MES_0 segment acquired.

Application conditions:

Activation/Deactivation: (based on LV_SEG_AD_RAW_ER evaluation)

// Raw adaptive value monitoring activation

If LV_INH_OBD_ER_AD = 0 *// OBDI inhibition conditions - See App. Inc.*
And LV_INJ_CUT = 1 *// All cylinders shut-off*
And N_32 < C_N_SEG_T_AD_RAW_MAX_ER
And N_32 > C_N_SEG_T_AD_RAW_MIN_ER
And LV_ENA_SEG_T_MES = 1
Then LV_SEG_AD_RAW_ER = 1
Else LV_SEG_AD_RAW_ER = 0
LV_SEG_AD_ER = 0

EndIf

Formula section:

// Adaptive value learning process activation condition vs N

If(1) **{** LV_SEG_AD_AVL_ER = 0
And [(LV_AT = 0 **And** N_32 **<** C_N_MAX_L_SEG_AD_ER_MT
And N_32 **>** C_N_MIN_L_SEG_AD_ER_MT **)**
Or (LV_AT = 1 **And** N_32 **<** C_N_MAX_L_SEG_AD_ER_AT
And N_32 **>** C_N_MIN_L_SEG_AD_ER_AT **)] }**

OR

{ LV_SEG_AD_AVL_ER = 1
And [(LV_AT = 0 **And** N_32 **<** C_N_MAX_H_SEG_AD_ER_MT
And N_32 **>** C_N_MIN_L_SEG_AD_ER_MT **)**
Or (LV_AT = 1 **And** N_32 **<** C_N_MAX_H_SEG_AD_ER_AT
And N_32 **>** C_N_MIN_L_SEG_AD_ER_AT **)] }**

Then(1) LV_SEG_AD_N_VLD =1
Else(1) LV_SEG_AD_N_VLD =0
EndIf(1)

// Adaptive value learning process activation

If(1) LV_SEG_AD_FDOUT_ER = 0
And [(LV_INJ_CUT = 1 *(all cylinders shut-off)* during last C_NR_SEG_AD_ACT_MIN_ER segments since last SEG_NR_MES = 0 **)**
And LV_SEG_AD_N_VLD = 1
And | DRV1_ER | **<** C_DRV1_MAX_SEG_AD_ER during C_NR_SEG_DRV1_MIN_ER segments **)**
Or LV_SEG_AD_FDOUT_ER_EOL = 0 **]**

Then(1) LV_SEG_AD_ER = 1 *// ER adaptive learning process enable*
If(2) LV_SEG_AD_AVL_ER = 0
Then (2) *// First adaptive process phase*
If(3) LV_SEG_AD_FDOUT_ER_EOL = 0
Then(3) CRLC_SEG_AD_ER = C_CRLC_SEG_AD_ER_EOL
Else(3) CRLC_SEG_AD_ER = C_CRLC_FAST_SEG_AD_ER
EndIf(3)

$$DELTA\_CRK\_DIF\_MAX\_ER_{(n)} = DELTA\_CRK\_DIF\_MAX\_ER_{(n-1)} - \frac{NC\_CYL\_NR * 4 * C\_CRK\_DIF\_MAX\_ER * CRLC\_SEG\_AD\_ER}{3}$$

CTR_SEG_AD_ER = CTR_SEG_AD_ER + 1 *(with saturation to max value)*

```
        If(3)   CTR_SEG_AD_ER >= C_NR_SEG_AD_AVL_ER
        Then(3)     LV_SEG_AD_AVL_ER = 1    // First adaptive process phase achieved
        EndIf(3)
    Else(2)     // Engine lifetime adaptive process phase
        CTR_SEG_AD_ER = CTR_SEG_AD_ER + 1     (with saturation to max value)
        If(3)   LV_SEG_AD_FDOUT_ER_EOL = 0
        Then(3)     CRLC_SEG_AD_ER = C_CRLC_SEG_AD_ER_EOL
        Else(3)     CRLC_SEG_AD_ER = C_CRLC_SEG_AD_ER
        EndIf(3)
    EndIf(2)
Else(1)     LV_SEG_AD_ER = 0                 // ER adaptive learning process disable
EndIf(1)
```

4.79.3.2 Engine roughness offset during first adaptive process phase

Update rate: ER segment task

Application conditions:

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Action on deactivation condition ***Or*** *ECU reset :* THD_AD_ER = 0

Formula section:

```
If      LV_SEG_AD_AVL_ER = 0
Then        // The ER offset factor THD_AD_ER is decreased via DELTA_CRK_DIF_MAX_ER value
            THD_AD_ER = DELTA_CRK_DIF_MAX_ER * SEG_T_COR
Else        THD_AD_ER = 0
EndIf
```

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_N_MIN_L_SEG_AD_ER_MT	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed threshold for crankshaft target wheel learning with MT					
C_N_MIN_L_SEG_AD_ER_AT	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed threshold for crankshaft target wheel learning with AT					
C_N_SEG_T_AD_RAW_MAX_ER	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel monitoring					
C_N_SEG_T_AD_RAW_MIN_ER	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed threshold for crankshaft target wheel monitoring					
C_CRK_DIF_MAX_ER	1	0...1FFH	0...7.79724	0.0152588	[°/oo]
Limitation of ER adaptation range (mechanical tolerance)					
C_DRV1_MAX_SEG_AD_ER	1	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Maximum value of DRV1_ER (dynamic part of engine roughness) for crankshaft target wheel learning.					
C_NR_SEG_AD_ACT_MIN_ER	1	0...FFH	0...255	1	[-]
Minimum segments number in fuel cut-off mode before triggering ER segment adaptive learning process					
C_NR_SEG_AD_AVL_ER	1	0...FFH	0...255	1	[-]
Segments number to achieve the first adaptive process phase					
C_NR_SEG_DRV1_MIN_ER	1	0...FFH	0...255	1	[-]
Minimum number of segments for engine roughness dynamic part					
C_CRLC_SEG_AD_ER	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Nominal ER segment adaptive values filtering coefficient					
C_CRLC_FAST_SEG_AD_ER	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Fast ER segment adaptive values filtering coefficient (fast learning phase)					
C_CRLC_SEG_AD_ER_EOL	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
EOL specific filtering coefficient for ER segment adaptive values learning process					
C_N_MAX_H_SEG_AD_ER_MT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in high range with MT					
C_N_MAX_MID_SEG_AD_ER_MT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in mid range with MT					
C_N_MAX_L_SEG_AD_ER_MT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in low range with MT					
C_N_MAX_H_SEG_AD_ER_AT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in high range with AT					
C_N_MAX_MID_SEG_AD_ER_AT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in mid range with AT					
C_N_MAX_L_SEG_AD_ER_AT	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed threshold for crankshaft target wheel learning in low range with AT					

4.79.4 Engine roughness adaptive values calculation & filtering

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
SEG_AD_RAW_MMV_ER[NC_CYL_NR]	V	8000...7FFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER filtered adaptive values used for learning process monitoring (for each segment)					
SEG_AD_MMV_ER_L[NC_CYL_NR]	V/O/S	8000...7FFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER filtered adaptive values used for learning process (for each segment) versus low engine speed					
SEG_AD_MMV_ER_MID[NC_CYL_NR]	V/O/S	8000...7FFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER filtered adaptive values used for learning process (for each segment) versus middle engine speed					
SEG_AD_MMV_ER_H[NC_CYL_NR]	V/O/S	8000...7FFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER filtered adaptive values used for learning process (for each segment) versus high engine speed					
LV_SEG_AD_LIM_ER	V/O/S	0...1H	0...1	1	[-]
ER filtered adaptive values out of range					

Input data:

SEG_AD_MMV_ER_APP[NC_CYL_NR]	LV_REQ_SEG_AD_MMV_ER_APP	SEG_NR_MES	LV_SEG_AD_RST_ER_EOL
SEG_AD_MMV_ER_APP[NC_CYL_NR]	SEG_T_MES_CYL[NC_CYL_NR]	C_N_MAX_H_SEG_AD_ER_MT	C_N_MAX_H_SEG_AD_ER_AT
C_N_MAX_MID_SEG_AD_ER_MT	C_N_MAX_MID_SEG_AD_ER_AT	C_N_MAX_L_SEG_AD_ER_MT	

FUNCTION DESCRIPTION:

Segment adaptation only makes sense in the engine operating state trailing throttle fuel cut-off (LV_PUC). It allows to determinate the coefficients for ER segment adaptive values filters used in the learning process.

Three different stages can be differentiate:

- Fast ER segment adaptive values filtering mode, used to quickly obtain valid adaptive values when the EMS is new.

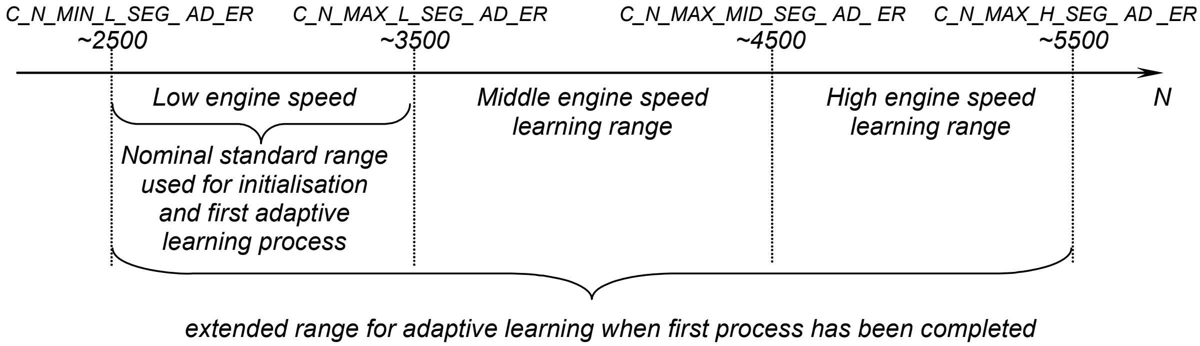
  Since Segment Time Adaptive Process is not completed *(ie. LV_SEG_AD_AVL_ER = 0*), the segment adaptation is only performed and learnt in the low engine speed range (*C_N_MIN_L _SEG_AD_ER* < *N_32* < *C_N_MAX_L_ SEG_AD_ER*).
  In this first case, only one global adaptation value is defined: the same value of segment adaptation will be calculated and applied for all engine speed range (*ie. SEG_AD_MMV_L_[SEG] with SEG_AD_MMV_H_[SEG] = SEG_AD_MMV_MID_[SEG] = SEG_AD_MMV_L_[SEG] in this case*).

- Nominal ER segment adaptive values filtering mode, used to monitor adaptive values during engine lifetime.

  When the first Adaptive Process has been completed *(ie. LV_SEG_AD_AVL_ER = 1*), then the Adaptive Process versus engine speed can take place
  In this case, the Engine Speed area for adaptation enabled will be enlarged (*C_N_MIN_L_SEG_AD_ER* < *N_32* < *C_N_MAX_H_SEG_AD_ER*) and the adaptation values will then be defined and performed according to the running engine speed range (*individual and separate calculation of SEG_AD_MMV_L_[SEG]*

for low N, SEG_AD_MMV_MID_[SEG] for mid N and SEG_AD_MMV_H_[SEG] for high N).

- EOL specific filtering mode, to be used on EOL or special service request.



LV_SEG_AD_LIM_ER is set if at least one SEG_AD_MMV_ER_L/MID/H[x] value exceeds C_SEG_AD_MAX_ER.

First time initialisation :

The ER segment adaptive values learning process is initialised when :

- ECU is initialised for the first time *(flash memory formatting)*
- A loss of adaptation values has been detected *(flash memory corrupted)*.
- A specific external tool request occurs *(use of LV_SEG_AD_RST_ER_EOL defined in Application Incidences file)*

// ER segment adaptive values reseted

For x = 0 ... [NC_CYL_NR-1]

SEG_AD_MMV_ER_L[x] = SEG_AD_MMV_ER_MID[x] = SEG_AD_MMV_ER_H[x] = 0

EndFor

LV_SEG_AD_LIM_ER = 0

Initialisation at ECU reset:

For x = 0...(NC_CYL_NR - 1)

SEG_AD_RAW_MMV_ER[x] = SEG_AD_MMV_ER_L[x]

EndFor

Update rate: ER segment occurrence

Application conditions:

This process is executed every engine cycle when SEG_T_MES_0 (reference segment) is acquired and when the monitoring of the adaptive learning process is active

Activation: LV_SEG_AD_RAW_ER = 1

Process on activation transient:

If LC_SEG_AD_LIM_REAC_ER = 1 *// reactivation of the adaptive values range check*

Then LV_SEG_AD_LIM_ER = 0

Else No Operation

EndIf

Deactivation: LV_SEG_AD_RAW_ER = 0

Formula section: *Temporary data are in italic*

For(1) x = 0 ... to ... [NC_CYL_NR-1]

Deceleration correction factor *(temporary data)*:

FAC_DECE_SEG_AD_ER[x] =

(x / NC_CYL_NR) * (SEG_T_MES_CYL[0] - *SEG_T_MES_CYL[0]_prev*)

Raw adaptive values:

SEG_AD_ER[x] =

[(SEG_T_MES_CYL[x] - *SEG_T_MES_CYL[0]_prev*) - *FAC_DECE_SEG_AD_ER[x]*] / *SEG_T_MES_CYL[0]_prev*

Engine roughness adaptive values filtering for monitoring and learning process:

$SEG\_AD\_RAW\_MMV\_ER[x]_i = SEG\_AD\_RAW\_MMV\_ER[x]_{i-1} +$

$C\_CRLC\_SEG\_AD\_RAW\_ER * (SEG\_AD\_ER[x]_i - SEG\_AD\_RAW\_MMV\_ER[x]_{i-1})$

If(2) LV_SEG_AD_ER = 1 **And** LV_SEG_AD_AVL_ER = 0

Then(2) $SEG\_AD\_MMV\_ER\_L[x]_i = SEG\_AD\_MMV\_ER\_L[x]_{i-1} +$

$CRLC\_SEG\_AD\_ER * [SEG\_AD\_ER[x]_i - SEG\_AD\_MMV\_ER\_L[x]_{i-1}]$

$SEG\_AD\_MMV\_ER\_MID[x]_i = SEG\_AD\_MMV\_ER\_L[x]$

$SEG\_AD\_MMV\_ER\_H[x]_i = SEG\_AD\_MMV\_ER\_L[x]$

Engine roughness adaptive values out of range check:

If(3) | SEG_AD_MMV_ER_L[x] | >= C_SEG_AD_MAX_ER

Then(3) LV_SEG_AD_LIM_ER = 1

EndIf(3)

EndIf(2)

If(2) LV_SEG_AD_ER = 1 **And** LV_SEG_AD_AVL_ER = 1

Then(2) **If**(3) **[** N_32 >= C_N_MAX_MID_SEG_AD_ER_MT **And** LV_AT = 0

Or N_32 >= C_N_MAX_MID_SEG_AD_ER_AT **And** LV_AT = 1 **]**

Then(3) $SEG\_AD\_MMV\_ER\_H[x]_i = SEG\_AD\_MMV\_ER\_H[x]_{i-1} +$

$CRLC\_SEG\_AD\_ER * [SEG\_AD\_ER[x]_i - SEG\_AD\_MMV\_ER\_H[x]_{i-1}]$

Engine roughness adaptive values out of range check:

If(4) | SEG_AD_MMV_ER_H[x] | >= C_SEG_AD_MAX_ER

Then(4) LV_SEG_AD_LIM_ER = 1

EndIf(4)

Else(3) **If**(4) **[** N_32 >= C_N_MAX_L_SEG_AD_ER_MT **And** LV_AT = 0
Or N_32 >= C_N_MAX_L_SEG_AD_ER_AT **And** LV_AT = 1 **]**
Then(4) $SEG\_AD\_MMV\_ER\_MID[x]_i = SEG\_AD\_MMV\_ER\_MID[x]_{i-1} + CRLC\_SEG\_AD\_ER * [SEG\_AD\_ER[x]_i - SEG\_AD\_MMV\_ER\_MID[x]_{i-1}]$
Engine roughness adaptive values out of range check:
If(5) | SEG_AD_MMV_ER_MID[x] | >= C_SEG_AD_MAX_ER
Then(5) LV_SEG_AD_LIM_ER = 1
EndIf(5)
Else(4) $SEG\_AD\_MMV\_ER\_L[x]_i = SEG\_AD\_MMV\_ER\_L[x]_{i-1} + CRLC\_SEG\_AD\_ER * [SEG\_AD\_ER[x]_i - SEG\_AD\_MMV\_ER\_L[x]_{i-1}]$
Engine roughness adaptive values out of range check:
If(5) | SEG_AD_MMV_ER_L[x] | >= C_SEG_AD_MAX_ER
Then(5) LV_SEG_AD_LIM_ER = 1
EndIf(5)
EndIf(4)
EndIf(3)
EndIf(2)

If(2) LV_SEG_AD_ER = 1 → 0
And LV_SEG_AD_AVL_ER = 1
//no risk because THD_AD_ER = 0, CTR_SEG_AD_ER no more used
And LV_REQ_SEG_AD_MMV_ER_APP = 1
Then(2) SEG_AD_MMV_ER_L[x] = SEG_AD_MMV_ER_APP[x]
SEG_AD_MMV_ER_MID[x] = SEG_AD_MMV_ER_APP[x]
SEG_AD_MMV_ER_H[x] = SEG_AD_MMV_ER_APP[x]
LV_SEG_AD_LIM_ER = 0
EndIf(2)

EndFor(1)

where:

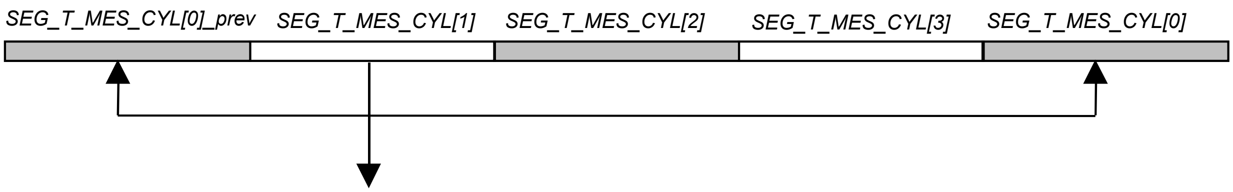
SEG_AD_ER[x] : *raw adaptive value for logical segment x, temporary data*
FAC_DECE_SEG_AD_ER[x] : *deceleration correction factor, temporary data*
SEG_AD_RAW_MMV_ER[x] : *fast filtered adaptive value for logical segment x*
SEG_AD_MMV_ER_L[x] : *filtered adaptive value for logical segment x low engine speed range*

SEG_AD_MMV_ER_MID[x] : *filtered adaptive value for logical segment x mid engine speed range*

SEG_AD_MMV_ER_H[x] : *filtered adaptive value for logical segment x high engine speed range*

SEG_T_MES_CYL[0]_prev : *raw segment period of reference segment (SEG_NR_MES=0), on previous engine cycle, <u>temporary data,</u> oldest value*

SEG_T_MES_CYL[0] : *raw segment period of reference segment (SEG_NR_MES=0) after NC_CYL_NR segments (720° crank angle), newest value*



FAC_DECE_SEG_AD_ER[1] =
*1/NC_CYL_NR * [SEG_T_MES_CYL[0] - SEG_T_MES_CYL[0]_prev]*

<u>Calibration data:</u>

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit

LC_SEG_AD_LIM_REAC_ER	1	0...1H	0...1	1	[-]
Reactivation of the ER segment adaptation range check at each learning phase					
C_CRLC_SEG_AD_RAW_ER	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Filter coefficient for ER segment adaptive values monitoring					
C_SEG_AD_MAX_ER	1	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
Limitation of ER segment adaptation range (mechanical tolerance).					

4.79.5 Engine roughness adaptive values difference on the same physical segment out of range

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_SEG_AD_DIF_MAX_ER	V/O	0...1H	0...1	1	[-]
Maximum ER adaptive values difference on same physical segment					
#IF NC_CYL_NR = 4, 6, 8, 10 Or 12					
SEG_AD_DIF_ER_0	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 0 & Identical)					
SEG_AD_DIF_ER_1	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 1 & Identical)					
#ENDIF					
#IF NC_CYL_NR = 6, 8, 10 Or 12					
SEG_AD_DIF_ER_2	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 2 & Identical)					
#ENDIF					
#IF NC_CYL_NR = 8, 10 Or 12					
SEG_AD_DIF_ER_3	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 3 & Identical)					
#ENDIF					
#IF NC_CYL_NR = 10 Or 12					
SEG_AD_DIF_ER_4	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 4 & Identical)					
#ENDIF					
#IF NC_CYL_NR = 12					
SEG_AD_DIF_ER_5	V	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
ER adaptive values difference on same physical segment (Segment 5 & Identical)					
#ENDIF					

Input data:

SEG_NR_MES	NC_CYL_NR		

General information:

Due to possible feedback from Dual Mass Flywheel oscillations *(see transmission/crankshaft design)* in adaptive process engine speed range, fluctuations on ER adaptive values of identical physical segments can induce dissimilarity in ER segment correction.

If difference between identical physical segments ER fast filtered adaptive values exceeds a calibratable gap, a fade out is applied.

ER identical physical segments adaptive value differences are evaluated each 720°Crk, for that this process is executed every engine cycle when SEG_T_MES_0 (*SEG_NR_MES = 0)* is acquired and when the adaptive monitoring process is active.

Update rate: *ER segment occurrence,*

Application conditions:

Activation: SEG_NR_MES = 0

And LV_SEG_AD_RAW_ER = 1

Deactivation: SEG_NR_MES != 0

Or LV_SEG_AD_RAW_ER = 0

Initialisation : *on ECU reset:*

LV_SEG_AD_DIF_MAX_ER = 0

#If NC_CYL_NR = 4, 6, 8, 10 or 12

SEG_AD_DIF_ER_0 = 0

SEG_AD_DIF_ER_1 = 0

#EndIf

#If NC_CYL_NR = 6, 8, 10 or 12

SEG_AD_DIF_ER_2 = 0

#EndIf

#If NC_CYL_NR = 8, 10 or 12

SEG_AD_DIF_ER_3 = 0

#EndIf

#If NC_CYL_NR = 10 or 12

SEG_AD_DIF_ER_4 = 0

#EndIf

#If NC_CYL_NR = 12

SEG_AD_DIF_ER_5 = 0

#EndIf

Formula section:

#If NC_CYL_NR = 3 Or NC_CYL_NR = 5

LV_SEG_AD_DIF_MAX_ER = 0

#EndIf

#If NC_CYL_NR = 4

SEG_AD_DIF_ER_0 = | SEG_AD_RAW_MMV_ER_0 - SEG_AD_RAW_MMV_ER_2 |

SEG_AD_DIF_ER_1 = | SEG_AD_RAW_MMV_ER_1 - SEG_AD_RAW_MMV_ER_3 |

If SEG_AD_DIF_ER_0 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_1 >= C_SEG_AD_DIF_MAX_ER

Then LV_SEG_AD_DIF_MAX_ER = 1

Else LV_SEG_AD_DIF_MAX_ER = 0

EndIf

#EndIf

#If NC_CYL_NR = 6

SEG_AD_DIF_ER_0 = | SEG_AD_RAW_MMV_ER_0 - SEG_AD_RAW_MMV_ER_3 |
SEG_AD_DIF_ER_1 = | SEG_AD_RAW_MMV_ER_1 - SEG_AD_RAW_MMV_ER_4 |
SEG_AD_DIF_ER_2 = | SEG_AD_RAW_MMV_ER_2 - SEG_AD_RAW_MMV_ER_5 |

If SEG_AD_DIF_ER_0 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_1 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_2 >= C_SEG_AD_DIF_MAX_ER
Then LV_SEG_AD_DIF_MAX_ER = 1
Else LV_SEG_AD_DIF_MAX_ER = 0
EndIf

#EndIf

#If NC_CYL_NR = 8

SEG_AD_DIF_ER_0 = | SEG_AD_RAW_MMV_ER_0 - SEG_AD_RAW_MMV_ER_4 |
SEG_AD_DIF_ER_1 = | SEG_AD_RAW_MMV_ER_1 - SEG_AD_RAW_MMV_ER_5 |
SEG_AD_DIF_ER_2 = | SEG_AD_RAW_MMV_ER_2 - SEG_AD_RAW_MMV_ER_6 |
SEG_AD_DIF_ER_3 = | SEG_AD_RAW_MMV_ER_3 - SEG_AD_RAW_MMV_ER_7 |

If SEG_AD_DIF_ER_0 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_1 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_2 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_3 >= C_SEG_AD_DIF_MAX_ER
Then LV_SEG_AD_DIF_MAX_ER = 1
Else LV_SEG_AD_DIF_MAX_ER = 0
EndIf

#EndIf

#If NC_CYL_NR = 10

SEG_AD_DIF_ER_0 = | SEG_AD_RAW_MMV_ER_0 - SEG_AD_RAW_MMV_ER_5 |
SEG_AD_DIF_ER_1 = | SEG_AD_RAW_MMV_ER_1 - SEG_AD_RAW_MMV_ER_6 |
SEG_AD_DIF_ER_2 = | SEG_AD_RAW_MMV_ER_2 - SEG_AD_RAW_MMV_ER_7 |
SEG_AD_DIF_ER_3 = | SEG_AD_RAW_MMV_ER_3 - SEG_AD_RAW_MMV_ER_8 |
SEG_AD_DIF_ER_4 = | SEG_AD_RAW_MMV_ER_4 - SEG_AD_RAW_MMV_ER_9 |

If SEG_AD_DIF_ER_0 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_1 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_2 >= C_SEG_AD_DIF_MAX_ER
Or SEG_AD_DIF_ER_3 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_4 >= C_SEG_AD_DIF_MAX_ER

Then LV_SEG_AD_DIF_MAX_ER = 1

Else LV_SEG_AD_DIF_MAX_ER = 0

EndIf

#EndIf

#If NC_CYL_NR = 12

SEG_AD_DIF_ER_0 = | SEG_AD_RAW_MMV_ER_0 - SEG_AD_RAW_MMV_ER_6 |

SEG_AD_DIF_ER_1 = | SEG_AD_RAW_MMV_ER_1 - SEG_AD_RAW_MMV_ER_7 |

SEG_AD_DIF_ER_2 = | SEG_AD_RAW_MMV_ER_2 - SEG_AD_RAW_MMV_ER_8 |

SEG_AD_DIF_ER_3 = | SEG_AD_RAW_MMV_ER_3 - SEG_AD_RAW_MMV_ER_9 |

SEG_AD_DIF_ER_4 = | SEG_AD_RAW_MMV_ER_4 - SEG_AD_RAW_MMV_ER_10 |

SEG_AD_DIF_ER_5 = | SEG_AD_RAW_MMV_ER_5 - SEG_AD_RAW_MMV_ER_11 |

If SEG_AD_DIF_ER_0 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_1 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_2 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_3 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_4 >= C_SEG_AD_DIF_MAX_ER

Or SEG_AD_DIF_ER_5 >= C_SEG_AD_DIF_MAX_ER

Then LV_SEG_AD_DIF_MAX_ER = 1

Else LV_SEG_AD_DIF_MAX_ER = 0

EndIf

#EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_SEG_AD_DIF_MAX_ER	1	0...7FFFH	0...7.81226	0.2384e-3	[°/oo]
Maximum range for difference between same physical segment adaptive values					

4.79.6 Actual adaptive values versus engine speed

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
SEG_AD_MMV_ER[NC_CYL_NR]	V/O/S	8000...7FFFH	-7.8125...7.81226	0.2384e-3	[°/oo]
ER filtered adaptive values used for learning process (for each segment)					

Input data:

LV_SYN_ENG	N_32	LV_AT	C_N_MAX_H_SEG_AD_ER_MT
C_N_MAX_L_SEG_AD_ER_MT	C_N_MAX_MID_SEG_AD_ER_AT	C_N_MAX_MID_SEG_AD_ER_MT	C_N_MAX_H_SEG_AD_ER_AT
C_N_MAX_L_SEG_AD_ER_AT			

FUNCTION DESCRIPTION:

The adaptative values from the three engine speed areas have be read and written into SEG_AD_MMV_ER[x] depending on engine speed.

For the definition of SEG_AD_MMV_ER[x], the following cases are considered:

For [x] = 0...NC_CYL_NR - 1

The calculation is done for the **actual active segment**
Update of SEG_AD_MMV_[x] occurs **when** segment **[x]** is active (*SEG_AD_MMV_1 is updated for x = 1, SEG_AD_MMV_2 is updated for x = 2, etc...*)

Update rate: ER segment occurrence

Application conditions:

Activation: LV_ENA_SEG_T_MES = 1

Deactivation: LV_ENA_SEG_T_MES = 0

Formula section:

```
If(1)     [ (C_N_MAX_L_SEG_AD_ER_MT = C_N_MAX_MID_SEG_AD_ER_MT =
            C_N_MAX_H_SEG_AD_ER_MT And LV_AT = 0)                          // MT-car
Or(1)      (C_N_MAX_L_SEG_AD_ER_AT = C_N_MAX_MID_SEG_AD_ER_AT =
            C_N_MAX_H_SEG_AD_ER_AT And LV_AT = 1)]                         // AT-car
Then(1)   SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_L[x]                            //low range
Else(1)
          If(2)   [ (C_N_MAX_MID_SEG_AD_ER_MT = C_N_MAX_H_SEG_AD_ER_MT
                     And LV_AT = 0)
          Or (2)    (C_N_MAX_MID_SEG_AD_ER_AT = C_N_MAX_H_SEG_AD_ER_AT
                     And LV_AT = 1)]
          Then(2)
                  If(3)   [(N_32 < C_N_MAX_L_SEG_AD_ER_MT And LV_AT = 0)
                  Or(3)    (N_32 < C_N_MAX_L_SEG_AD_ER_AT And LV_AT = 1)]
                  Then(3)   SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_L[x]          //low range
                  Else(3)   SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_MID[x]        //mid range
                  EndIf(3)
          Else(2)
                  If(4)    [ (N_32 < C_N_MAX_L_SEG_AD_ER_MT And LV_AT=0)   // MT-car
                  Or(4)      (N_32 < C_N_MAX_L_SEG_AD_ER_AT And LV_AT=1)]  // AT-car
                  Then(4)  SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_L[x]           //low range
                  Else(4)
                          If(5)  [(N_32 < C_N_MAX_MID_SEG_AD_ER_MT And LV_AT=0)
                          Or(5)  (N_32 < C_N_MAX_MID_SEG_AD_ER_AT And LV_AT=1)]
                          Then(5) SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_MID[x] //mid range
                          Else(5) SEG_AD_MMV_ER[x] = SEG_AD_MMV_ER_H[x] //high range
                          EndIf(5)
                  EndIf(4)
          EndIf(2)
EndIf(1)
```

4.80 Engine roughness signal preparation for cylinder balancing

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ER_STND_MMV_BAL[NC_CYL_NR]	O/V/S	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Filtered normalized cylinder selective engine roughness values					
ER_STD_MMV_BAL[NC_CYL_NR]	O/V/S	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Filtered standard deviation of the engine roughness					
ER_STND_MMV_DIF_BAL[NC_CYL_NR]	O/V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Difference of filtered normalized and average engine roughness value					
LV_DRV1_STND_BAL_FDOUT	O/V	0...1H	0...1	1	-
Flag for acceleration fade out of cylinder balancing					
ER_STND_MMV_STD_BAL[NC_CYL_NR]	O/V/S	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Filtered normalized engine roughness values for calculation of the standard deviation					
ER_STND_MMV_MV_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Mean value of all engine roughness values of the last [NC_CYL_NR] segments					
ER_STD_DIF_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Difference of the filtered normalized and normalized engine roughness values for calc. of the standard deviation					
ER_STND_FIL_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Normalized cylinder selective engine roughness values after median filter					
CTR_DLY_DRV1_STND_BAL	V	0...FFFFH	0...6.5535E+4	1	-
Segment delay counter for acceleration fade out of cylinder balancing					
ER_STD_BAL[NC_CYL_NR]	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Standard deviation of the engine roughness					
ER_STND_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Normalized cylinder selective engine roughness values					
DRV1_STND_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Standardized acceleration component of the ER value					
DLY_DRV1_STND_BAL	V	0...FFFFH	0...6.5535E+4	1	-
Segment delay for acceleration fade out of cylinder balancing					

Input data:

LV_ER_STND_ER_BAL_ACT	ER	SEG_NR_ER	N_32
LV_DRV1_ER_BAL_ACT	DRV1_ER	NC_CYL_NR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_ER_STD_BAL	1	0...FFH	0...0.99609307	0.00390625	-
Correlation constant for ER_STND_FIL_BAL value filter for standard deviation					
C_CRLC_ER_STD_MMV_BAL	1	0...FFH	0...0.99609307	0.00390625	-
Correlation constant for ER_STD_BAL value filter					

general specification

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_ER_STND_BAL	1	0...FFH	0...0.99609307	0.00390625	-
Correlation constant for ER_STND_FIL_BAL value filter					
C_CTR_DLY_DRV1_STND_BAL_DEC	1	0...FFH	0...255	1	-
Segment delay counter decrement value					
C_FAC_ER_STD_NEG_BAL	1	0...FFH	0...0.99609307	0.00390625	-
Weighting factor for negative ER_STD_DIF_BAL values					
C_FAC_ER_STD_POS_BAL	1	0...FFH	0...0.99609307	0.00390625	-
Weighting factor for positive ER_STD_DIF_BAL values					
IP_DLY_DRV1_STND_BAL	8	0...FFFFH	0...6.5535E+4	1	-
LDP_DRV1_STND_BAL_IP_DLY_BAL	8	0...FFFFH	-325.78...325.77	0.0099313	1/s²
Segment delay counter start value					

4.80.1 General information:

Attention - This specification version (402U02) is only valid for software variants which are providing the input variables ER and DRV1_ER as 32 bit values.

The engine roughness signal ER as well as the dynamic part of the engine roughness signal DRV1_ER are used as basis for the calculation of the input signal for the cylinder balancing control functions. The calculations are kept cylinder individual corresponding on the actual segment number SEG_NR_ER.

The module "Engine roughness signal preparation for cylinder balancing" is divided in two separate chapters. Within the first part the calculation of the input signals for the cylinder balancing controller is done, while the second part is providing fade out conditions while cylinder balancing has to be interrupted.

Application Condition

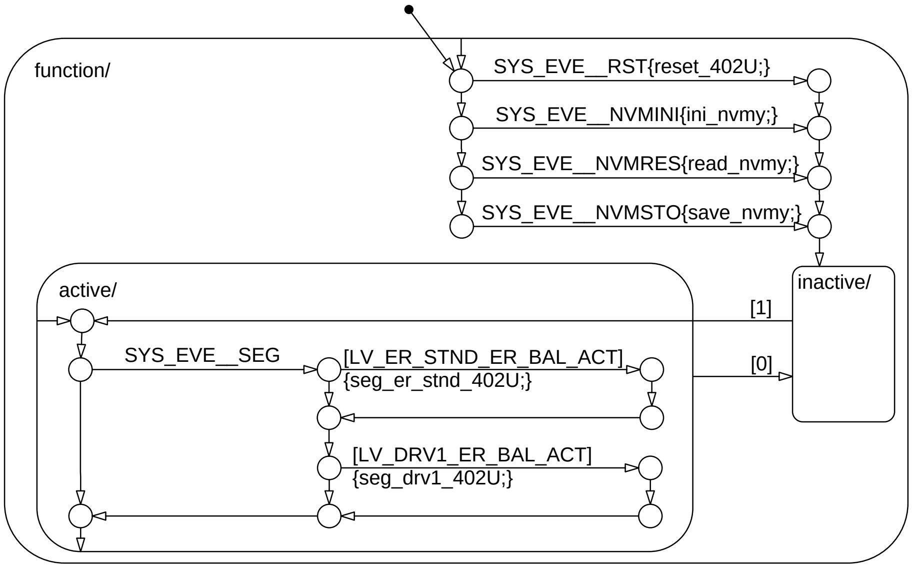


Figure 400 CYBL_SIGNAL/ APP_CDN/ Chart

Function Description

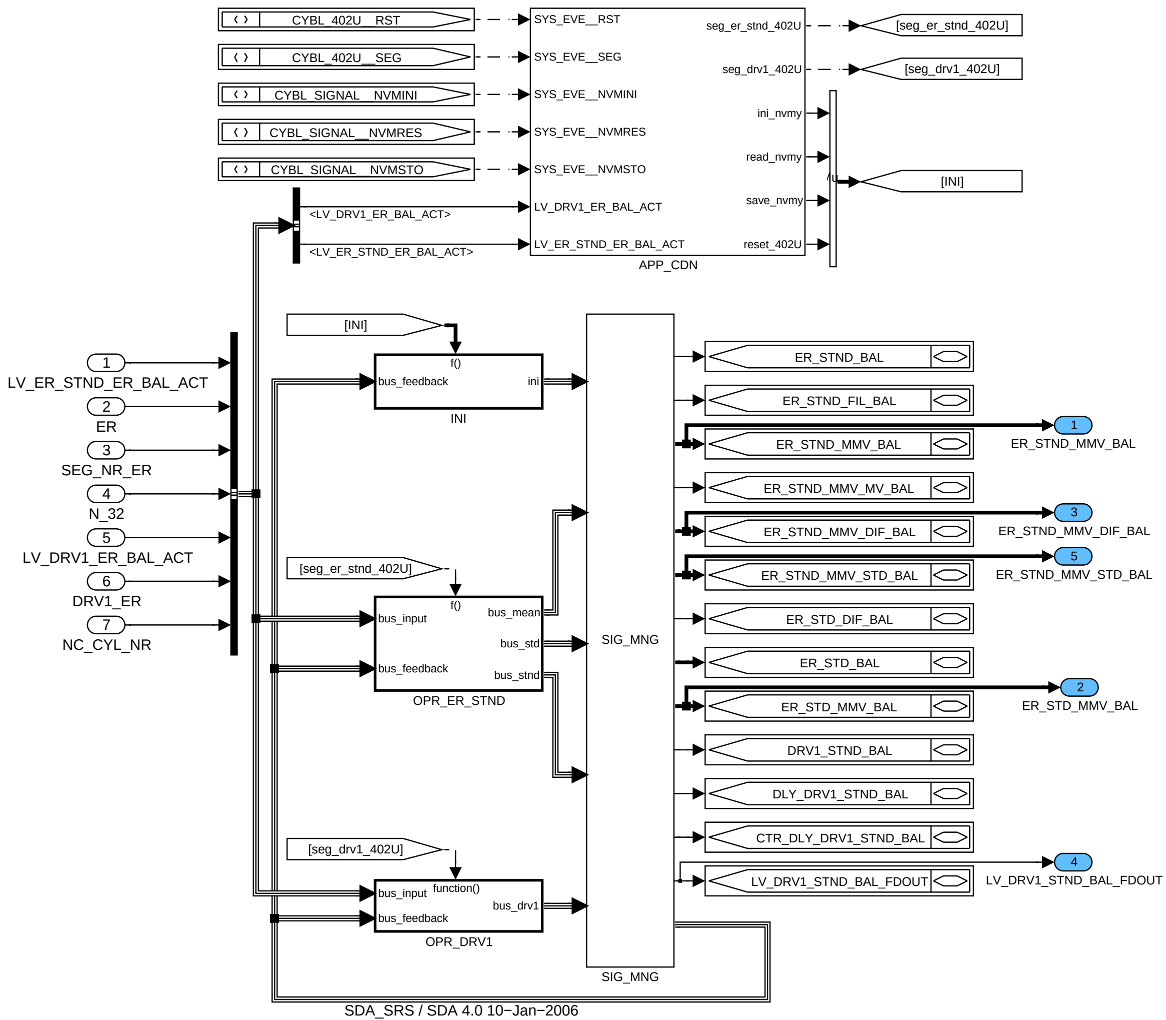


Figure 401 CYBL_SIGNAL

4.80.1.1 SUBFUNCTION: INI

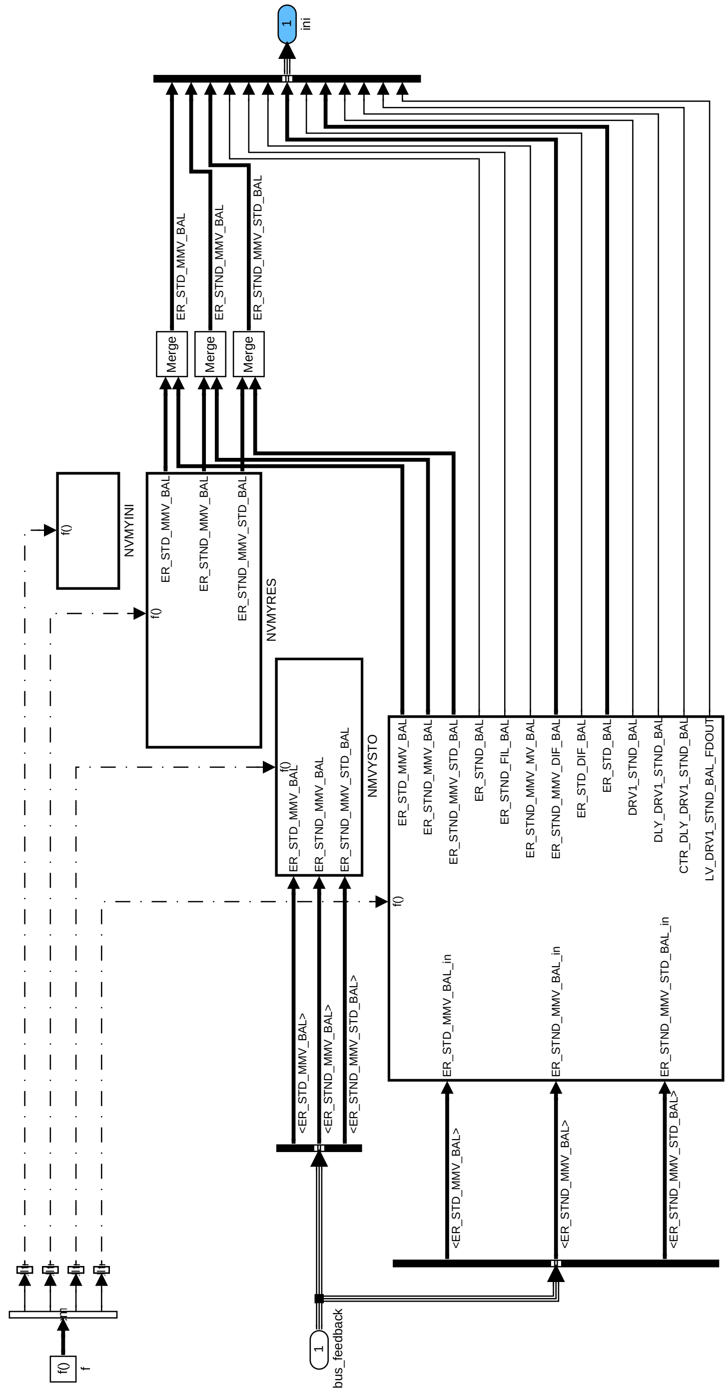


Figure 402 CYBL_SIGNAL/ INI

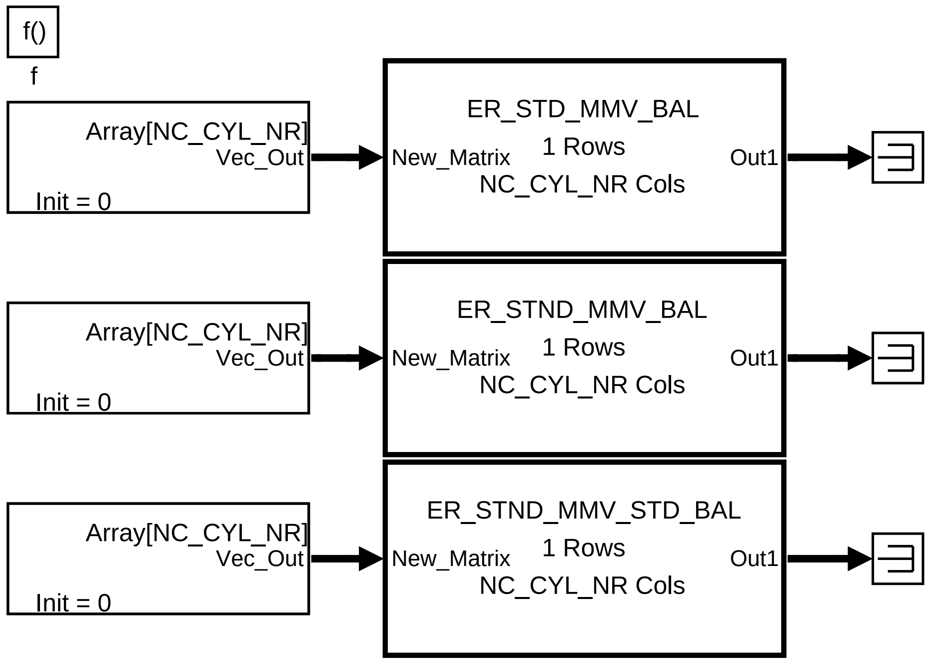


Figure 403 CYBL_SIGNAL/ INI/ NVMYINI

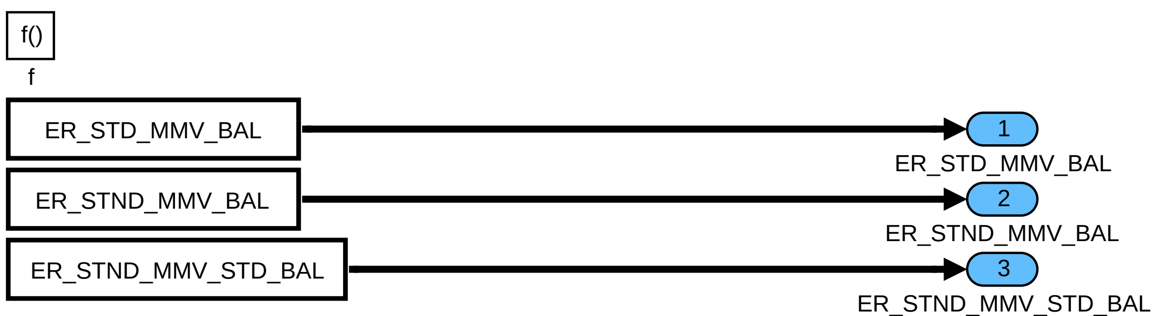


Figure 404 CYBL_SIGNAL/ INI/ NVMYRES

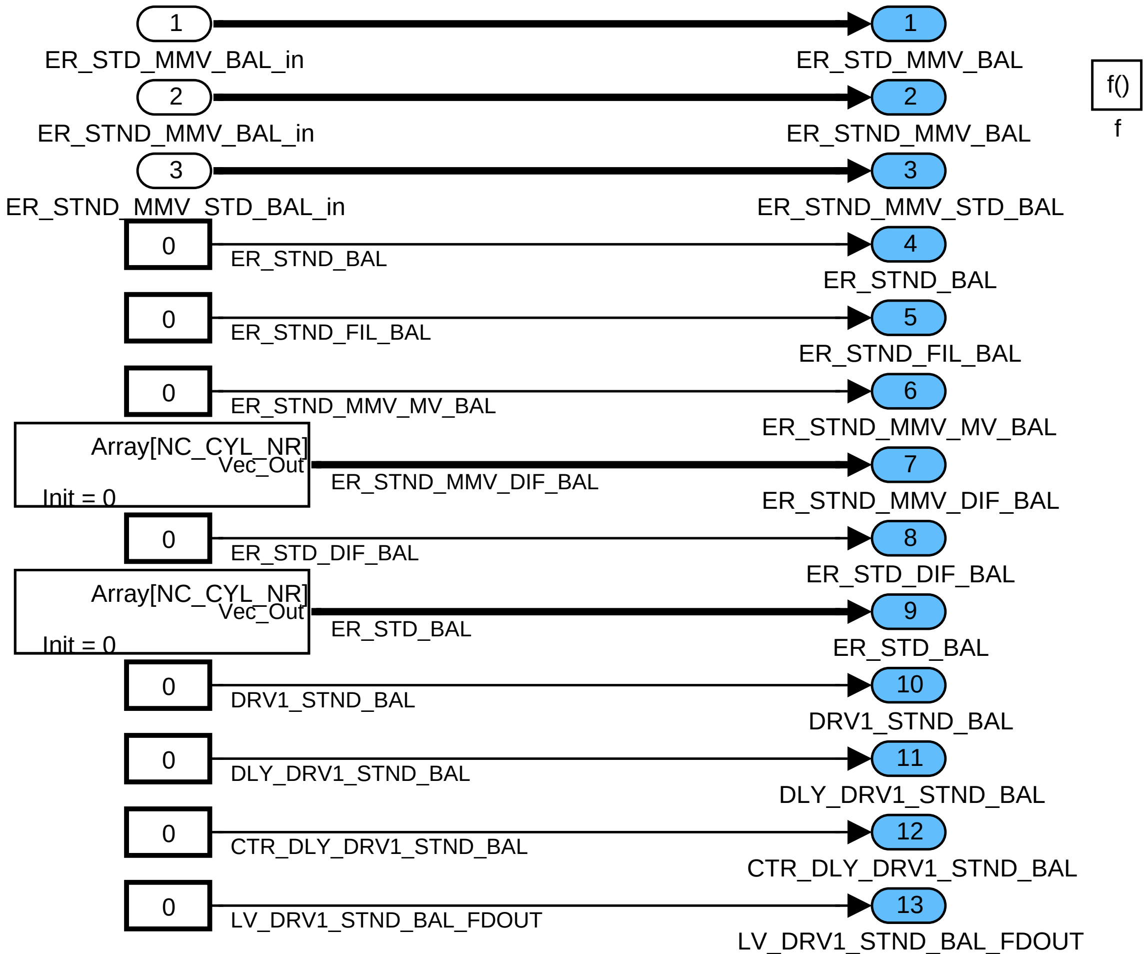


Figure 405 CYBL_SIGNAL/ INI/ RST

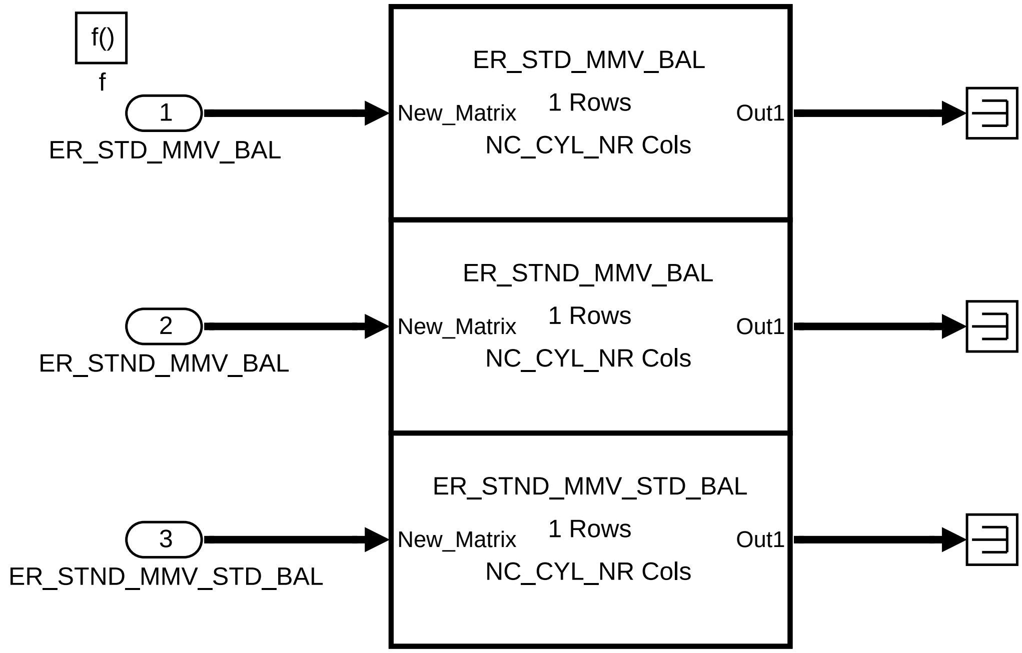


Figure 406 CYBL_SIGNAL/ INI/ NMVYSTO

4.80.1.2 Input signals for cylinder balancing adaptation functions

The ER values are used to calculate normalized cylinder selective engine roughness values ER_STND_BAL. After normalization (multiplication by N_32^3), the cylinder selective values are filtered by a Median Filter and a PT1-Filter.

In addition, moving mean values are built (ER_STND_MMV_STD_BAL[SEG_NR_ER]) for the calculation of cylinder specific standard deviations (ER_STD_BAL[SEG_NR_ER]).

The standard deviation ER_STD_DIF_BAL is calculated by subtracting the moving mean values ER_STND_MMV_STD_BAL[SEG_NR_ER] from the ER_STND_BAL values.

The resulting difference ER_STD_DIF_BAL is multiplied with a weighting factor for negative and positive values (C_FAC_ER_STD_NEG/POS_BAL). Afterwards a moving mean value is calculated based on this weighted signal.

To allow a cylinder specific adaptation via the engine roughness, a specific signal has to be created for each cylinder. Therefore, from the variable ER_STND_MMV_BAL[SEG_NR_ER] the value ER_STND_MMV_MV_BAL (average value of all cylinder specific moving mean values) is subtracted.

The values ER_STD_MMV_BAL[SEG_NR_ER], ER_STND_MMV_BAL[SEG_NR_ER] and ER_STND_MMV_STD_BAL[SEG_NR_ER] are stored in the non-volatile memory (NVMY) and used for initialization issues at next engine run.

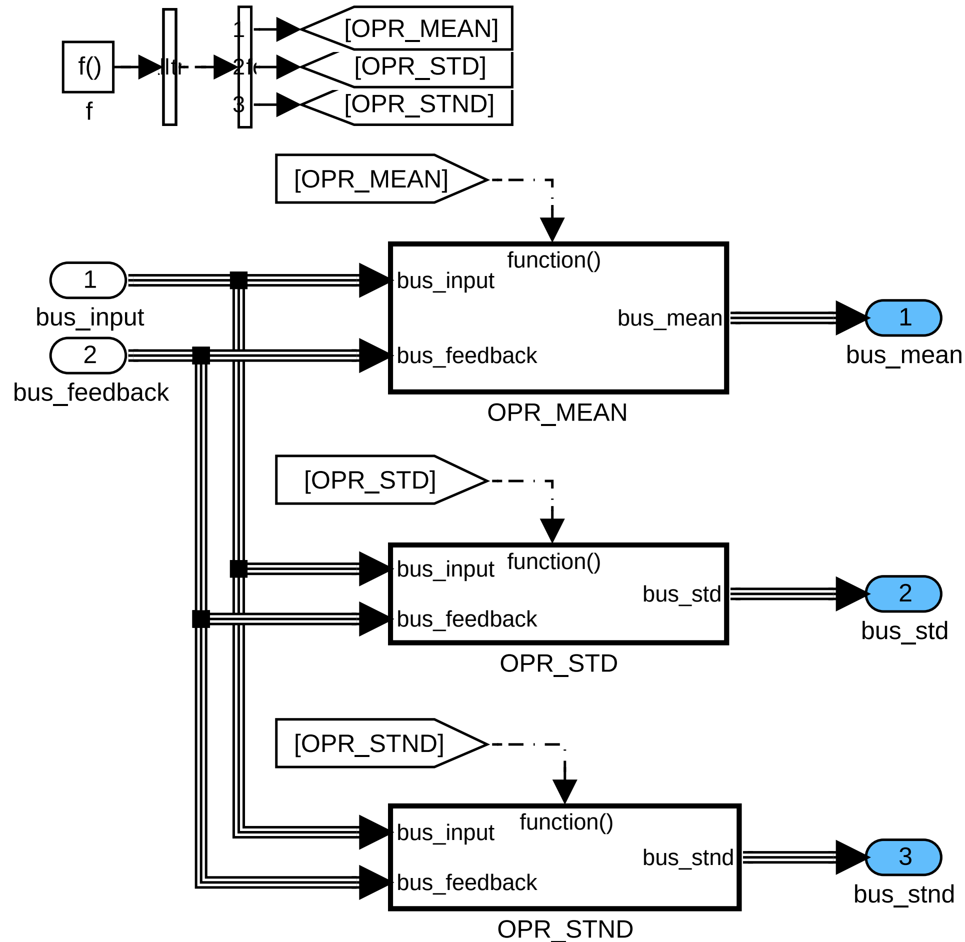


Figure 407 CYBL_SIGNAL/ OPR_ER_STND

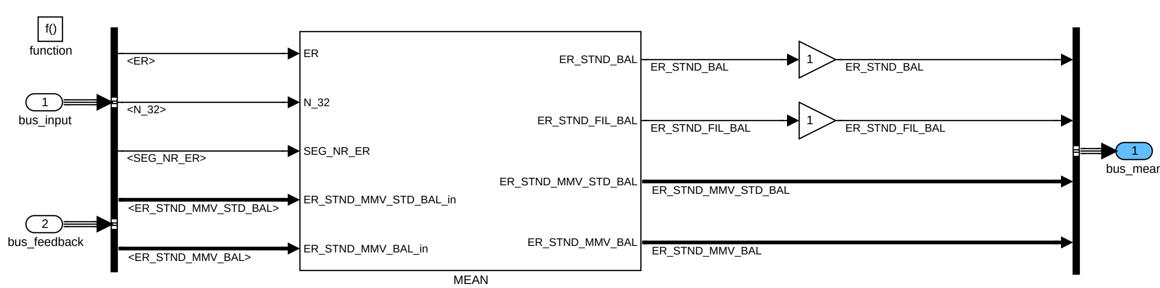


Figure 408 CYBL_SIGNAL/ OPR_ER_STND/ OPR_MEAN

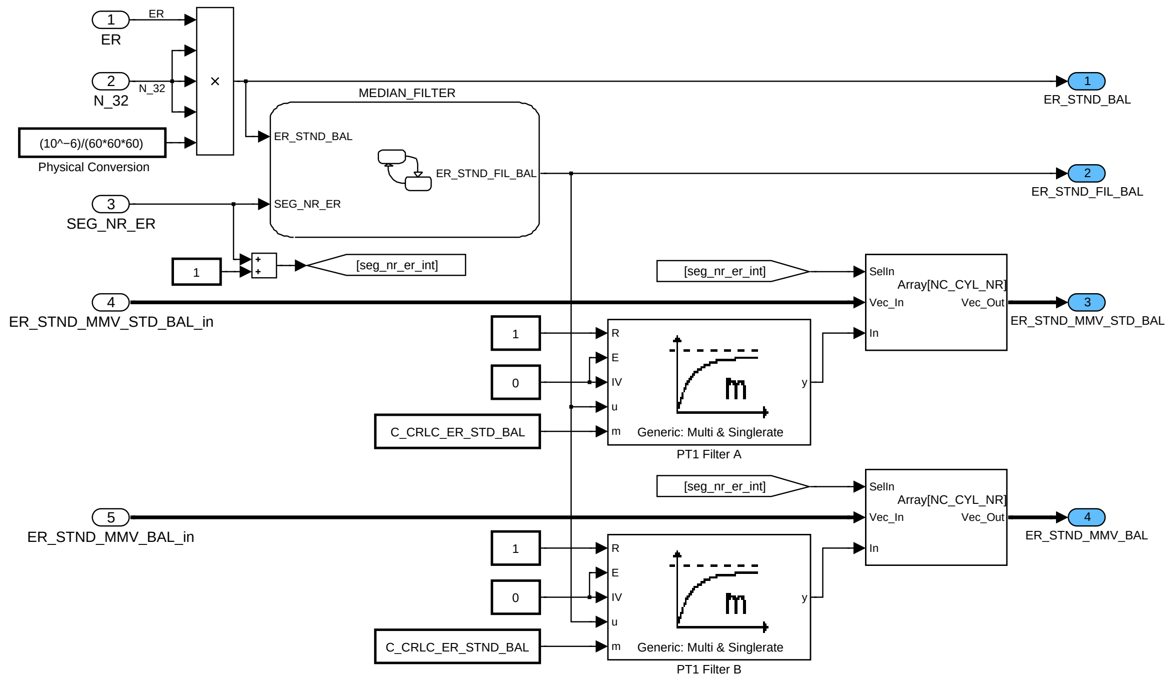


Figure 409 CYBL_SIGNAL/ OPR_ER_STND/ OPR_MEAN/ MEAN

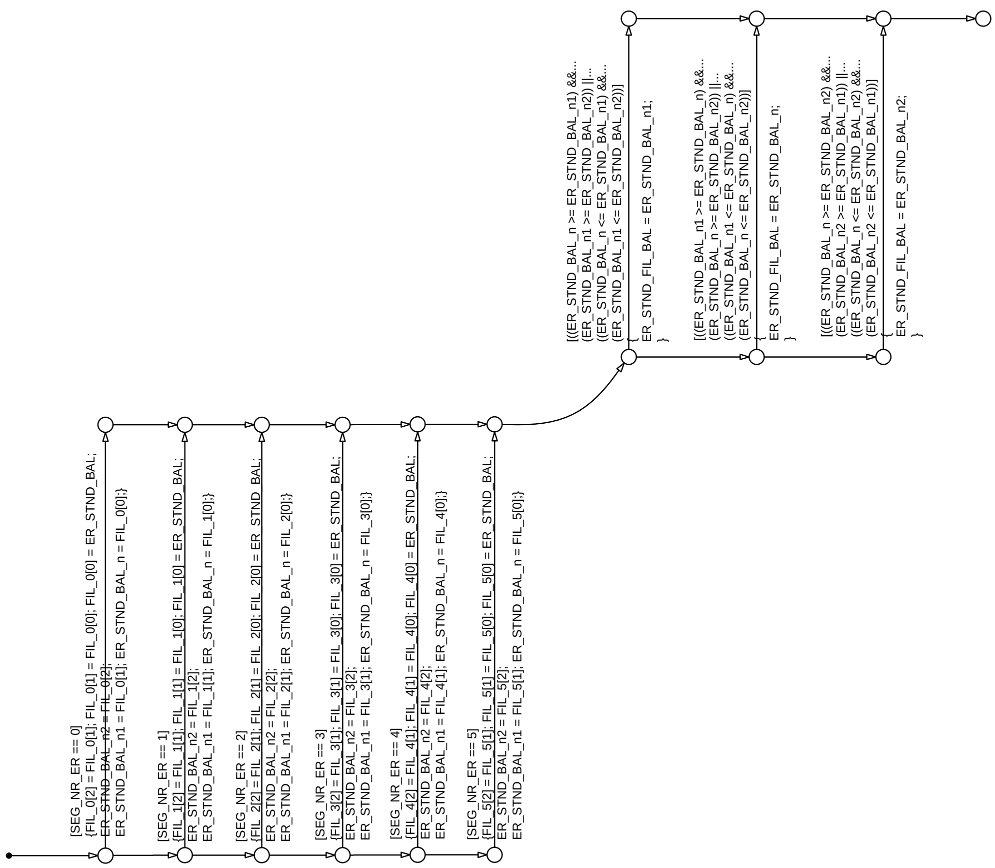


Figure 410 CYBL_SIGNAL/ OPR_ER_STND/ OPR_MEAN/ MEAN/ MEDIAN_FILTER

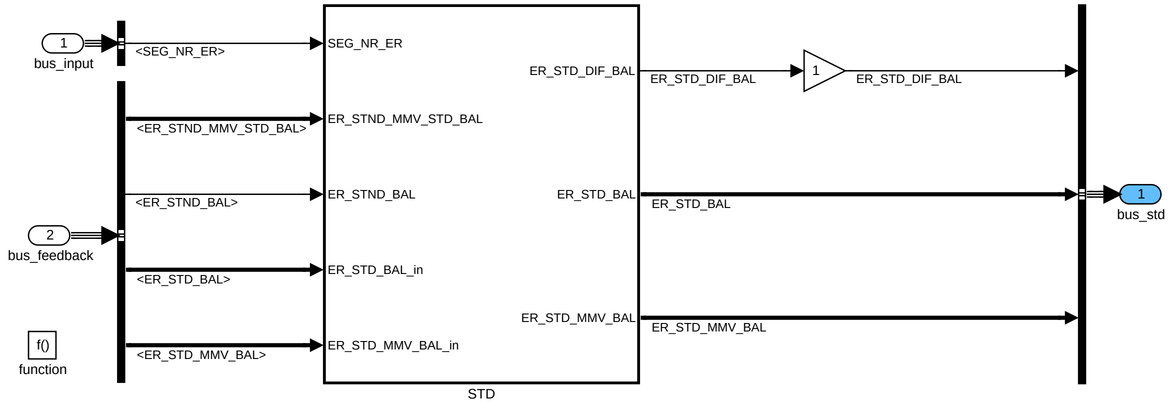


Figure 411 CYBL_SIGNAL/ OPR_ER_STND/ OPR_STD

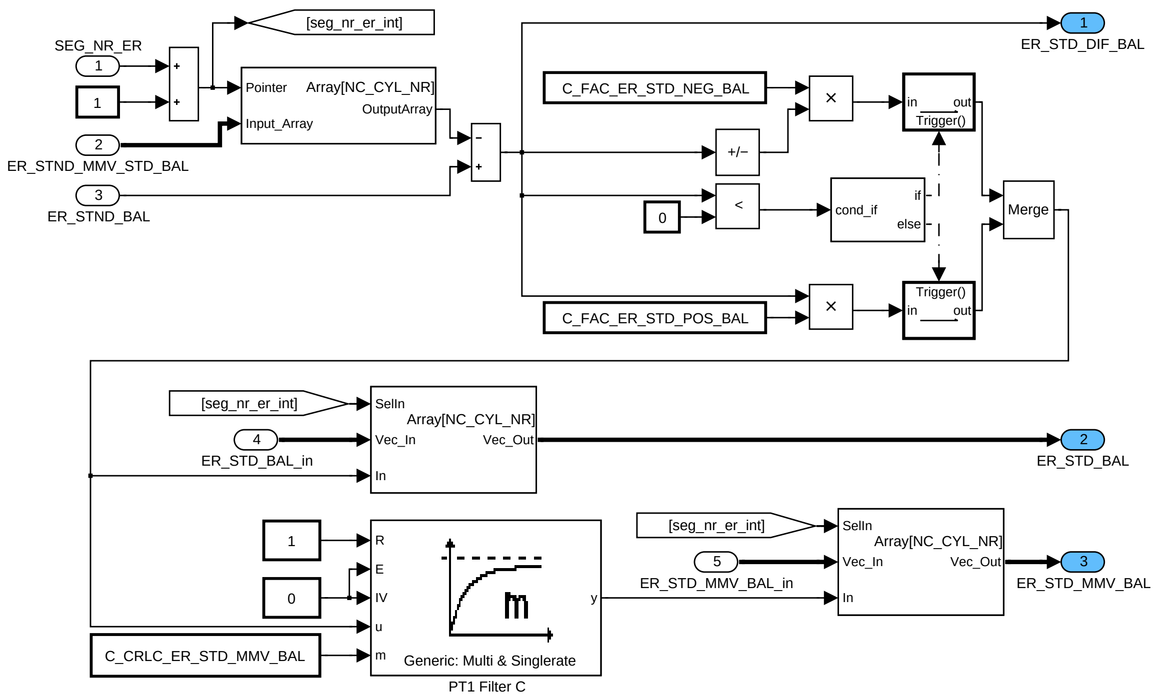


Figure 412 CYBL_SIGNAL/ OPR_ER_STND/ OPR_STD/ STD

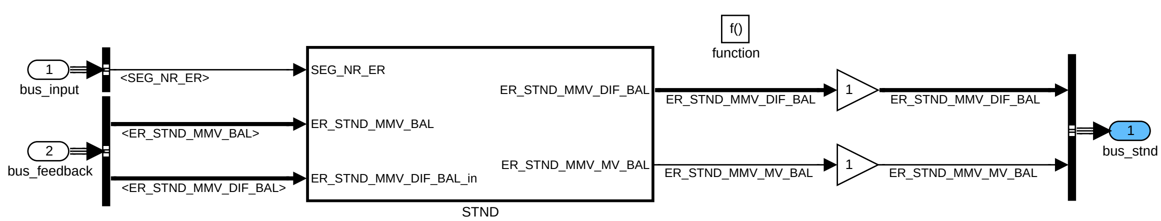


Figure 413 CYBL_SIGNAL/ OPR_ER_STND/ OPR_STND

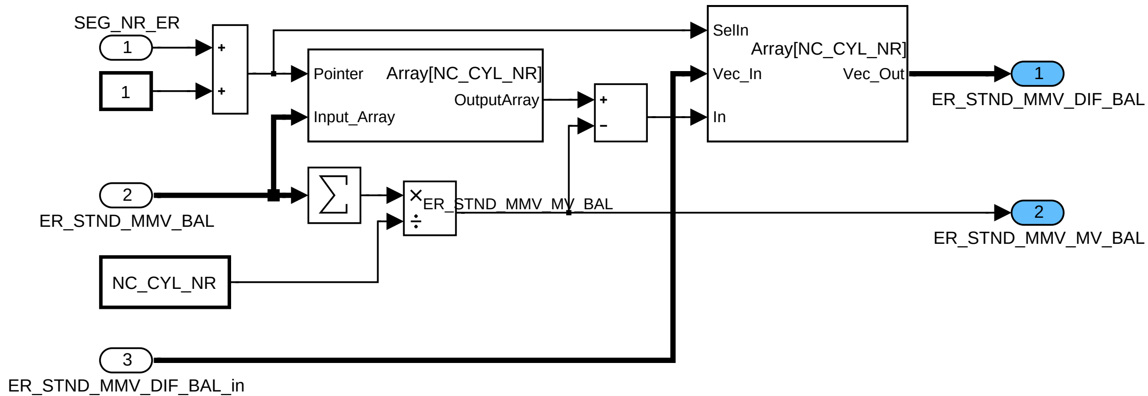


Figure 414 CYBL_SIGNAL/ OPR_ER_STND/ OPR_STND/ STND

4.80.1.3 Calculation of cylinder balancing acceleration fade out conditions

Multiplying the dynamic component (DRV1_ER) part of the engine roughness calculation with the engine speed by the power of three normalizes the signal. The defined signal is reproducing the engine acceleration in numbers. If the acceleration is quite high or low, in case of speeding up or down, the cylinder balancing adaptation is stopped for a certain number of segments. The number of segments are stored in a counter, which is counting back to zero in certain steps as mentioned in the signal flow diagram.

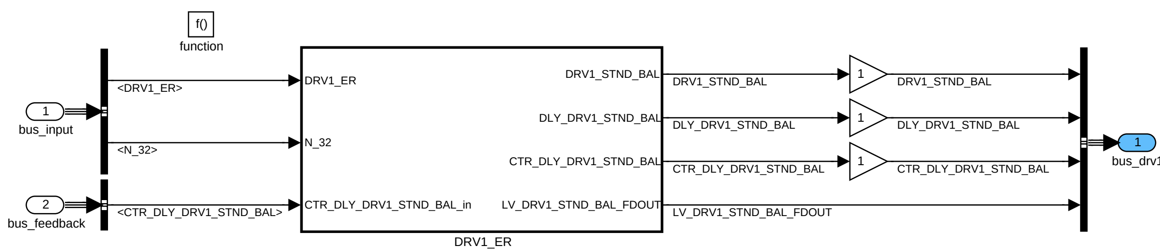


Figure 415 CYBL_SIGNAL/ OPR_DRV1

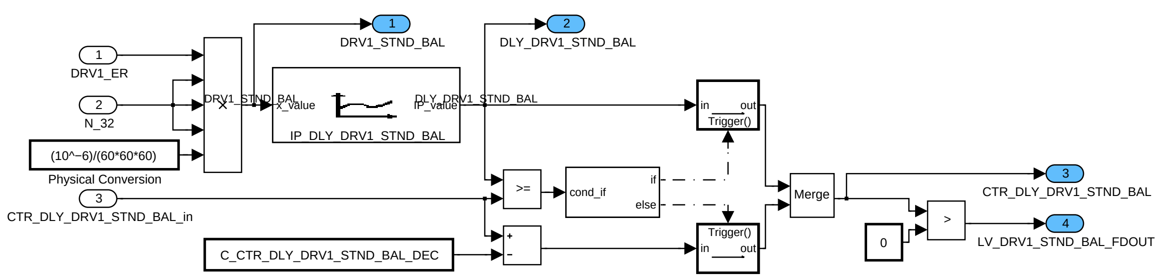


Figure 416 CYBL_SIGNAL/ OPR_DRV1/ DRV1_ER

4.80.1.4 SUBFUNCTION: SIG_MNG

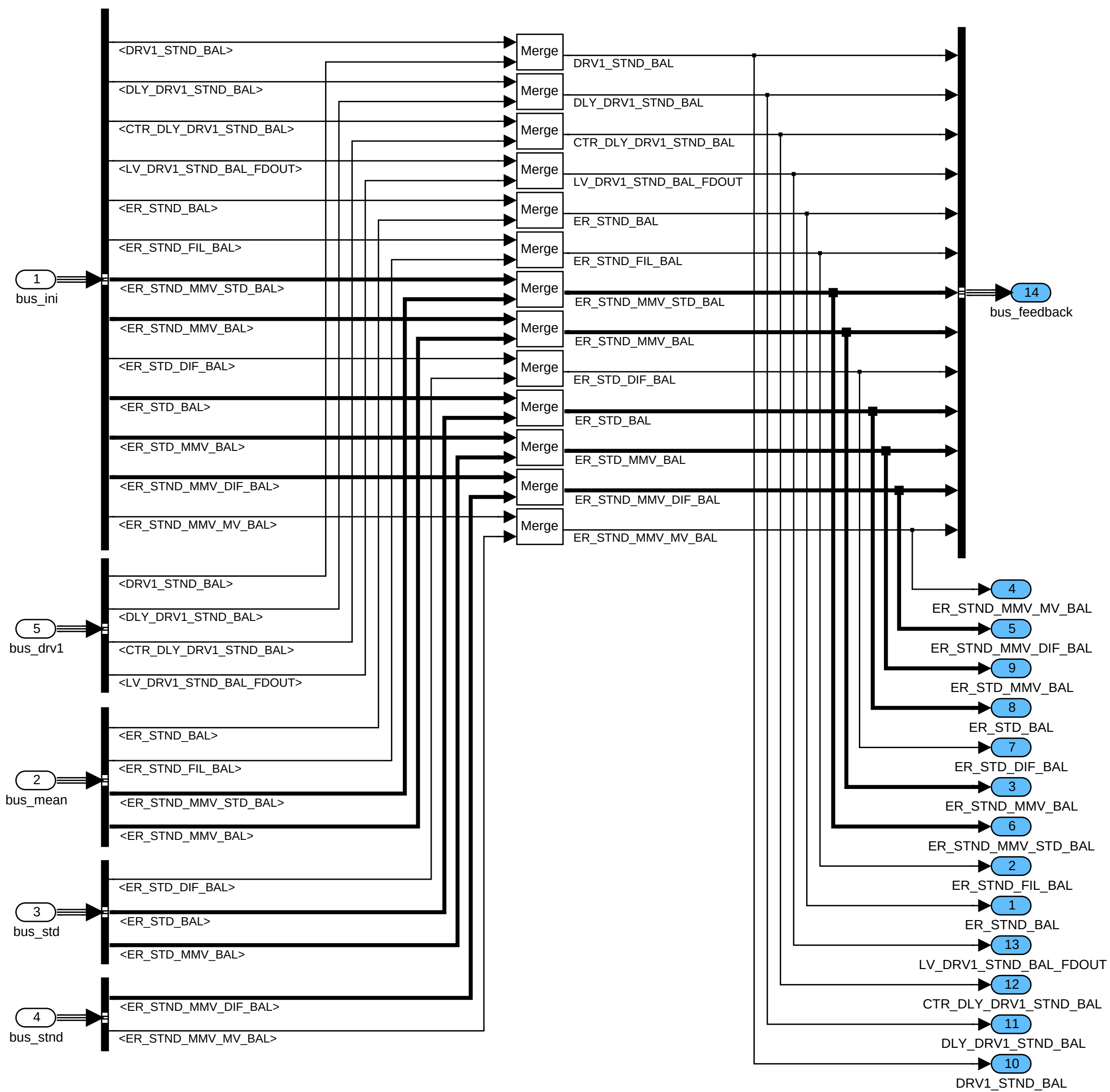


Figure 417 CYBL_SIGNAL/ SIG_MNG

4.81 Mass Air Flow variables

4.81.1 Altitude correction MAF_FAC_ALTI_MMV

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_FAC_ALTI_MMV	V/O	0...FFH	0...1.992	7.78 E-3	-
Mass air flow correction factor for altitude.					

Input data:

AMP_AD			

FUNCTION DESCRIPTION:

Formula section:

In order to take in account different atmospheric pressures at different altitudes, an altitude correction factor MAF_FAC_ALTI_MMV is calculated from division of AMP_AD by 1013 mbar.
The altitude correction factor is applied to TIPR_CST, TI_CST and ISAPWM_INI_I_ST during start and is taken into account as activation condition for evaporative emission control diagnosis and fuel system diagnosis.

Calculation of MAF_FAC_ALTI_MMV:

MAF_FAC_ALTI_MMV is calculated in all engine operation states:

MAF_FAC_ALTI_MMV = AMP_AD / 1013

Application recurrence : **1 sec.**

MAF_FAC_ALTI_MMV is stored in non-volatile memory. For a new car MAF_FAC_ALTI_MMV is initialised to 1.

4.82 Total running time

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
TRT	V/O/S	0...FFFFFFFFH	0...119304.6470 8	2.78E-05	[h]
Total running time					
TRT_FMY	-/S	0...FFFFFFFFH	0...119304.6470 8	2.78E-05	[h]
Total running time at failure memory cleared					

Input data:

LV_IGK	LV_ES		

Function description:

The total running time is incremented by 100ms each if the conditions "ignition key ON" and "engine running" has been detected.

The total running time is saved in the non-volatile memory at the end of the driving cycle.

Formula section:

IF LV_IGK = 1 AND LV_ES = 0

THEN TRT = TRT + 2,7778e-5h

ELSE TRT = TRT

Total running time at last time failure memory cleared

Purpose of this stragey is to memorize the total running time at the last time the failure memory was cleared.

Application conditions:

Initialisation: on failure memory clear service received

TRT_FMY = TRT

Recurrence: -

Activation: -

Deactivation: -

4.83 Engine Position Manager

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ST_EPM	-	0...8H	0...8	1	[-]
Engine position manager state					
LV_STOP_ENG	O	0...1H	0...1	1	[-]
Boolean information for engine stop request					
LV_FIRST_VLD_TOOTH	V/O	0...1H	0...1	1	[-]
First tooth detected after engine stop					
LV_RUN_ENG	V/O	0...1H	0...1	1	[-]
Engine running					
LV_SYN_ENG	V/O	0...1H	0...1	1	[-]
Boolean information for engine synchronization completed.					
LV_SYN_VLD	V/O	0...1H	0...1	1	[-]
Engine synchronization validated					
PSN_ENG_CRK	O	0...2CFFH	0...719.9375	0.0625	[°CRK]
Engine position from crankshaft					
PSN_ENG_CAM_LIH_CRK	O	0...2CFFH	0...719.9375	0.0625	[°CRK]
Engine position in crankshaft limp-home mode					
LV_LIH_ERR_CRK	V/O	0...1H	0...1	1	[-]
Crankshaft limp-home mode (crankshaft sensor failure)					
LV_LIH_ERR_CAM	V	0...1H	0...1	1	[-]
Camshaft limp-home mode (failure on all camshaft sensors)					
LV_ACT_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Selfsynchronisation activated on intake camshaft i					
LV_ACT_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Selfsynchronisation activated on exhaust camshaft i					
LV_ACT_LIH_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0...1H	0...1	1	[-]
Crankshaft limp-home operating mode activated on intake camshaft i					
LV_ACT_LIH_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0...1H	0...1	1	[-]
Crankshaft limp-home operating mode activated on exhaust camshaft i					
LV_ACT_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Crankshaft synchronization operating mode activated on intake camshaft i					
LV_ACT_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Crankshaft synchronization operating mode activated on exhaust camshaft i					
LV_ACT_CRK	V/O	0...1H	0...1	1	[-]
Crankshaft acquisition active					
LV_ERR_CAM	V/O	0...1H	0...1	1	[-]
Failure on all camshaft sensor signals in normal mode					
LV_ERR_LIH_CRK_CAM	V	0...1H	0...1	1	[-]
No more camshaft sensor signals available for crankshaft limp-home mode					
CTR_VLD_CAM_SYN_CRK	V	0...FFH	0...255	1	[-]
Counter for validation of camshaft signal for crankshaft synchronization					
LV_ENG_BACK_DET	V/O	0...1H	0...1	1	[-]
Engine backwards rotation detected					
LV_ENG_BACK_CFM	V/O	0...1H	0...1	1	[-]
Engine backwards rotation confirmed					
LV_CRK_MISS_RUN_ENG	V/O	0...1H	0...1	1	[-]
Boolean information engine running after engine stop request					
CTR_TEST_ENG_BACK_RST	V	0...FFH	0...255	1	[-]
Counter of successful tests to reset backwards rotation detection					

Input data:

LV_SYN_CAM_IN[NC_NR_CAM_CBK]	LV_IGK		LV_CRK_FIRST_VLD_TOOTH
LV_SYN_CAM_EX[NC_NR_CAM_CBK]	LV_CRK_RUN	LV_ERR_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_CAM_EX[NC_NR_CAM_CBK]
LV_CAM_SYN_CRK	LV_CRK_SYN	LV_CRK_STOP	LV_CAM_STOP_IN[NC_NR_CAM_CBK]
LV_ORNG_CAM_SYN_CRK	PSN_ENG_SYN_CAM_MIN	PSN_ENG_CRK_OFS	LV_CAM_STOP_EX[NC_NR_CAM_CBK]
PSN_EDGE_AD_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	NC_OFS_TDC0_REF_CRK	NC_PSN_EDGE_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	NC_PSN_EDGE_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]
PSN_EDGE_AD_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	NC_NR_TOOTH	IDX_EDGE_CAM_IN[NC_NR_CAM_CBK]	IDX_EDGE_CAM_EX[NC_NR_CAM_CBK]
LV_ERR_CRK	LV_SYN_VLD_CAM_LIH	LV_SEG_NR_UPD_REQ	LV_CAM_LIH_EXT_ENA
NC_ACT_CAM_EDGE_SYN	NC_ACT_CAM_EDGE_LIH	LV_CRK_MISS_TOOTH	PSN_ENG_ENSD
C_DYW_CAM_CRK_SYN_RTD_IN	C_DYW_CAM_CRK_SYN_RTD_EX	C_DYW_CAM_CRK_SYN_ADC_IN	C_DYW_CAM_CRK_SYN_ADC_EX
LV_ORNG_RATIO_CAM_IN[NC_NR_CAM_CBK]	LV_ORNG_RATIO_CAM_EX[NC_NR_CAM_CBK]	LV_LOST_SYN_CRK	C_N_NOT_REST
LC_ENG_BACK_INH	N_TOOTH	NC_NR_EDGE_CAM_IN	NC_NR_EDGE_CAM_EX
CTR_EDGE_CAM_IN[NC_NR_CAM_CBK]	CTR_EDGE_CAM_EX[NC_NR_CAM_CBK]		

FUNCTION DESCRIPTION:

General information:

The Engine Position Manager provides a summary of camshaft and crankshaft synchronization status information to the system. It sets the operating modes of camshaft and crankshaft acquisition depending on synchronization status, availability, and detected failures. The signal source for engine position will be selected based on the activated operating mode.

Camshaft and crankshaft signal acquisition will be set passive for a calibrated delay after key-off has been detected at low engine speed, and for another calibrated delay after a loss of synchronization at low engine speed. This avoids wrong injection, ignition and diagnosis with backward rotation after rapid key-off/on or stalling.

i = 1...NC_NR_CAM_CBK   z = 0...NC_NR_EDGE_CAM_IN/EX

Description:

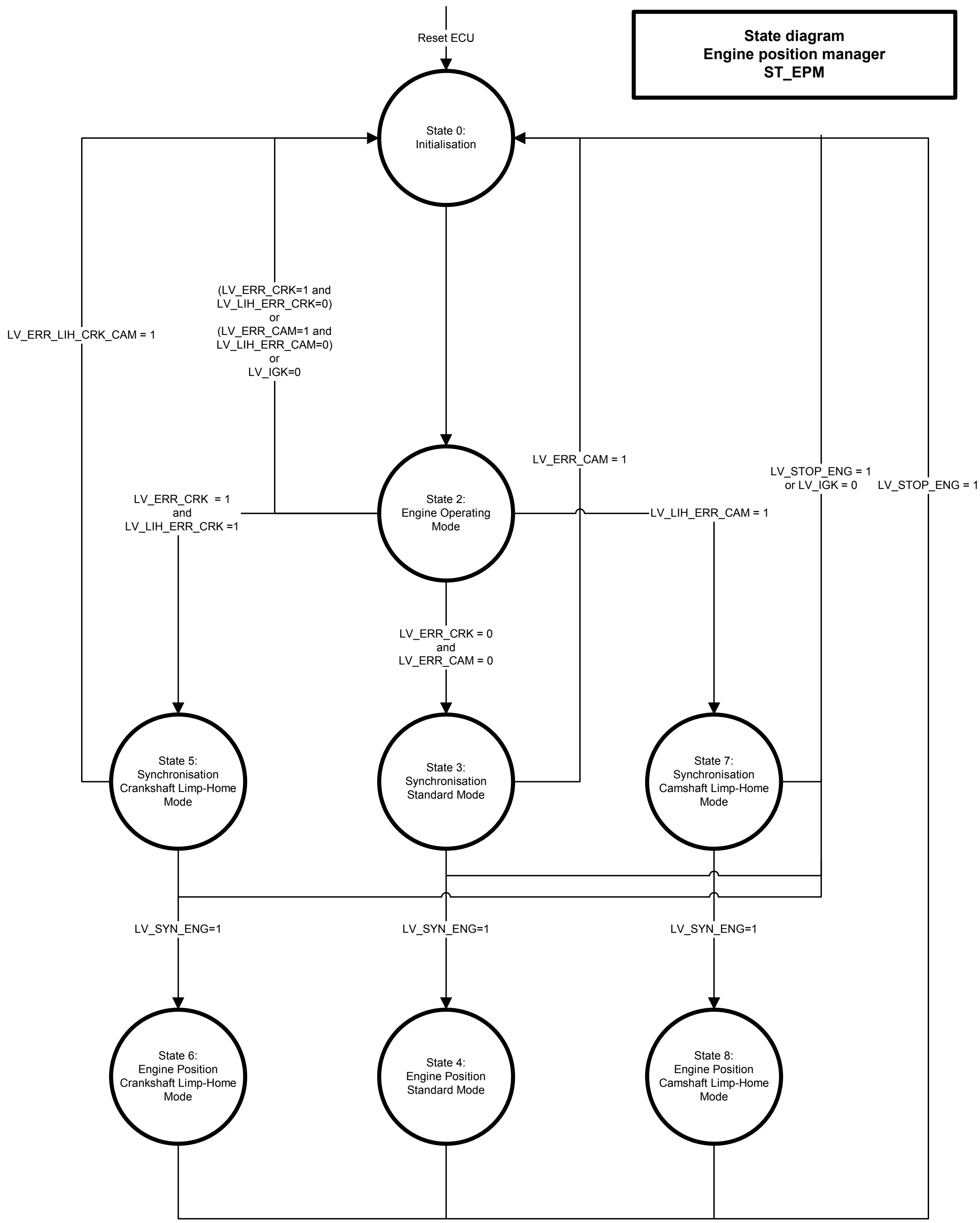
Engine synchronized (LV_SYN_ENG)	Yes (LV_SYN_ENG = 1)	No (LV_SYN_ENG = 0)
Crankshaft sensor available (LV_ERR_CRK = 0) At least one camshaft sensor available (LV_ERR_CAM = 0)	Standard Mode: Engine position and speed calculation with crankshaft sensor signal	Standard Mode Engine synchronisation with camshaft and crankshaft sensor information
Crankshaft sensor available (LV_ERR_CRK = 0) Failure on **all** Camshaft sensors (LV_ERR_CAM = 1)	Limp home Cam: Engine position and speed calculation with crankshaft sensor signal	Limp home Cam: Engine synchronisation without camshaft sensor information
Crankshaft sensor failure (LV_ERR_CRK = 1) At least one camshaft sensors available for crank limp-home (LV_ERR_LIH_CRK_CAM = 0)	Limp home Crank: Engine position and speed calculation without crankshaft sensor signal.	Limp home Crank: Engine synchronisation without crankshaft sensor information
Crankshaft sensor failure (LV_ERR_CRK = 1)		Engine synchronisation fiasco: Engine cannot be synchronized

Failure on **all** Camshaft sensors available for crank limp-home (LV_ERR_LIH_CRK_CAM = 1)		

Application conditions:

Recurrence: *10 ms*

4.83.1 State flow diagram



Initialisation:

LV_ERR_CAM is reset at 0 to 1 transition of LV_IGK

LV_CRK_MISS_RUN_ENG is reset and timer C_T_DLY_STOP_ENG stopped at 1 to 0 transition of LV_IGK

Formula section:

State 0 : Initialisation

Initialization of output data and deactivation of all state flow diagrams.

Input conditions:

External event:	ECU Reset
From state 2:	(LV_ERR_CRK = 1 And LV_LIH_ERR_CRK = 0)
	Or
	(LV_ERR_CAM = 1 And LV_LIH_ERR_CAM = 0)
	Or
	LV_IGK = 0
From state 3:	LV_ERR_CAM = 1
From state 5:	LV_ERR_LIH_CRK_CAM = 1
From state 3,5,7:	LV_STOP_ENG = 1
	Or
	LV_IGK = 0
From state 4,6,8:	LV_STOP_ENG = 1

Output conditions:

To State 2 :	End of initialisation

Action in the state:

LV_STOP_ENG = 1
LV_FIRST_VLD_TOOTH = 0
LV_RUN_ENG = 0
LV_SYN_ENG = 0
LV_SYN_VLD = 0
PSN_ENG_CRK = PSN_ENG_CAM_LIH_CRK = 0
CTR_VLD_CAM_SYN_CRK = 0
LV_LIH_ERR_CRK = 0
LV_LIH_ERR_CAM = 0
LV_ERR_LIH_CRK_CAM = 0

Deactivation of Crankshaft self-synchronisation:

LV_ACT_CRK = 0

Deactivation of Camshaft self-synchronisation for all camshaft signals:

LV_ACT_CAM_IN(EX)[i] = 0

Deactivation of Camshaft/Crankshaft synchronisation for all camshaft signals:

LV_ACT_SYN_CRK_CAM_IN(EX)[i] = 0

Deactivation of limp-home mode for all camshaft signals:

LV_ACT_LIH_CRK_CAM_IN(EX)[i] = 0

Deactivation of synchronization determination in camshaft limp home

LV_CAM_LIH_EXT_ENA = 0

LV_SYN_VLD_CAM_LIH = 0

LV_SEG_NR_UPD_REQ = 0

Deactivation of synchronization validation in camshaft limp home

Transition action:

None

State 2 : Engine Operating Mode

Select engine operating mode in function of diagnostic information and limp home status.

Input conditions:

From State 0: End of initialisation

Output conditions:

To state 0: (LV_ERR_CRK = 1 And

(LC_INH_LIH_CRK = 1 Or LV_ERR_LIH_CRK_CAM = 1))

Or

(LV_ERR_CAM = 1 And

LC_INH_LIH_CAM = 1 And LV_ERR_CRK = 0)

Or

LV_IGK = 0

To state 3: LV_ERR_CRK = 0 And LV_ERR_CAM = 0

To state 5: LV_LIH_ERR_CRK = 1 And LV_ERR_CRK = 1

To state 7: LV_LIH_ERR_CAM = 1

Action in the state:

If	LV_ERR_CRK = 1	
Then	**If**	LC_INH_LIH_CRK = 0
	And	LV_ERR_LIH_CRK_CAM = 0
	Then	*Select Crankshaft limp-home mode*
		LV_LIH_ERR_CRK = 1
	Else	*Crankshaft limp-home not possible*
	Endif	
Else	**If**	LV_ERR_CAM = 1
	And	LC_INH_LIH_CAM = 0
	Then	*Select Camshaft limp-home mode*
		LV_LIH_ERR_CAM = 1
	Else	*Camshaft limp-home not possible*
	Endif	
Endif		

Transition action:

To state 3,5,7: LV_STOP_ENG = 0

To state 3: Activation of Crankshaft self-synchronisation:

LV_ACT_CRK = 1

Activation of Camshaft self-synchronisation for all camshaft signals:

LV_ACT_CAM_IN(EX)[i] = 1

Activation of camshaft selection for synchronisation ()

Activation of Camshaft/Crankshaft synchronisation for the selected camshaft signal.

To state 5: Activation of Camshaft self-synchronisation for all the camshaft signals:

LV_ACT_CAM_IN(EX)[i] = 1

Activation of camshaft selection for limp-home ()

To state 7: Activation of Camshaft self-synchronisation for all the camshaft signals:

LV_ACT_CAM_IN(EX)[i] = 1

Activation of Crankshaft self-synchronisation:

LV_ACT_CRK = 1

State 3 : Synchronisation Standard Mode

Engine synchronisation with camshaft and crankshaft sensor information.

Input conditions:

From state 2:	LV_ERR_CRK = 0 And LV_ERR_CAM = 0

Output conditions:

To state 0:	LV_STOP_ENG = 1
	Or
	LV_IGK = 0
	Or
	LV_ERR_CAM = 1
To state 4:	LV_SYN_ENG = 1

Action in the state:

LV_FIRST_VLD_TOOTH = LV_CRK_FIRST_VLD_TOOTH
LV_RUN_ENG = LV_CRK_RUN
Activation of calculation of PSN_ENG_CRK in standard mode()
Activation of synchronization validation ()
Activation of engine position calculation standard mode ()
Activation of Camshaft self-synchronisation reactivation ()
Activation of engine stop detection ()

If LV_RUN_ENG = 1
Then Activation of engine speed calculation
Endif

On 0 to 1 transition of LV_CAM_SYN_CRK:

If LV_ERR_CAM_IN(EX)[i] = 0
For all available camshaft sensors
Then *Activation of fast synchronization ()*
Else *Activation of slow synchronization ()*
Endif

Transition action:

To state 4:	LV_CRK_MISS_RUN_ENG = 0
	Stop timer C_T_DLY_STOP_ENG

State 4 : Engine Position Standard Mode

Engine position and speed calculation with crankshaft sensor signal.

Input conditions:

From state 3: LV_SYN_ENG = 1

Output conditions:

To state 0: LV_STOP_ENG = 1

Action in the state:

Reactivation of Camshaft self-synchronisation ()

Activation of engine position calculation standard mode ()

Activation of segment period calculation ()

Activation of speed gradient calculation ()

Activation of crankshaft sensor phase angle correction ()

Activation of engine stop detection ()

If LV_SYN_VLD = 1

Then *Activation of Camshaft adaptation and position output ()*

Endif

Activation of Engine backwards rotation detection ()

Transition action:

To state 0: **If** C_T_DLY_STOP_ENG > 0

And LV_IGK = 1

And LV_CRK_MISS_RUN_ENG = 0

Then LV_CRK_MISS_RUN_ENG = 1

Start timer C_T_DLY_STOP_ENG

Reset LV_CRK_MISS_RUN_ENG after timer has elapsed

Endif

State 5 : Synchronisation Crankshaft Limp-Home Mode

Engine synchronisation without crankshaft sensor signal.

Input conditions:

From state 2: LV_LIH_ERR_CRK = 1

And

LV_ERR_CRK = 1

Output conditions:

To state 0: LV_STOP_ENG = 1

Or

LV_IGK = 0

Or

LV_ERR_LIH_CRK_CAM = 1

To state 6: LV_SYN_ENG=1

Action in the state:

Activation of crankshaft limp-home synchronization() for the selected camshaft signal

Activation of calculation of PSN_ENG_CAM_LIH_CRK()

Activation of engine position calculation crankshaft limp-home mode

Reactivation of Camshaft self-synchronisation ()

Activation of engine stop detection ()

If LV_RUN_ENG = 1

Then Activation of engine speed calculation

Endif

Transition action:

To state 6: LV_CRK_MISS_RUN_ENG = 0

Stop timer C_T_DLY_STOP_ENG

State 6 : Engine Position Crankshaft Limp-Home Mode

Engine position and speed calculation without crankshaft sensor signal.

Input conditions:

From state 5: LV_SYN_ENG = 1

Output conditions:

To state 0: LV_STOP_ENG = 1

Action in the state:

Activation of engine position calculation crankshaft limp-home mode

Activation of Camshaft self-synchronisation reactivation ()

Activation of segment period calculation

Activation of speed gradient calculation

Activation of engine stop detection ()

Transition action:

To state 0:

If C_T_DLY_STOP_ENG > 0

And LV_IGK = 1

And LV_CRK_MISS_RUN_ENG = 0

Then LV_CRK_MISS_RUN_ENG = 1

Start timer C_T_DLY_STOP_ENG

Reset LV_CRK_MISS_RUN_ENG after timer has elapsed

Endif

State 7 : Synchronisation Camshaft Limp-Home Mode

Engine synchronisation without camshaft sensor signal.

Input conditions:

From state 2: LV_LIH_ERR_CAM = 1

Output conditions:

To state 0: LV_STOP_ENG = 1

Or

LV_IGK = 0

To state 8: LV_SYN_ENG = 1

Action in the state:

Activation of camshaft limp-home synchronization()

LV_FIRST_VLD_TOOTH = LV_CRK_FIRST_VLD_TOOTH

LV_RUN_ENG = LV_CRK_RUN

Activation of calculation of PSN_ENG_CRK in camshaft limp-home mode()

Activation of engine position calculation standard mode

Activation of engine stop detection ()

If LV_RUN_ENG = 1

Then Activation of engine speed calculation

Endif

Transition action:

To state 8: LV_CRK_MISS_RUN_ENG = 0

Stop timer C_T_DLY_STOP_ENG

State 8 : Engine Position Camshaft Limp-Home Mode

Engine position and speed calculation without camshaft sensor signal.

Input conditions:

From state 7: LV_SYN_ENG = 1

Output conditions:

To state 0: LV_STOP_ENG = 1

Action in the state:

Activation of Synchronization determination in camshaft limp home()

Activation of Synchronization validation in camshaft limp home()

Reactivation of Camshaft self-synchronisation ()

Activation of PSN_ENG_CRK correction in camshaft limp home()

Activation of engine position calculation standard mode

Activation of segment period calculation

Activation of segment number correction in camshaft limp home()

Activation of speed gradient calculation

Activation of crankshaft sensor phase angle correction

Activation of engine stop detection ()

Activation of Engine backwards rotation detection ()

Transition action:

To state 0: **If** C_T_DLY_STOP_ENG > 0

And LV_IGK = 1

And LV_CRK_MISS_RUN_ENG = 0

Then LV_CRK_MISS_RUN_ENG = 1

Start timer C_T_DLY_STOP_ENG

Reset LV_CRK_MISS_RUN_ENG after timer has elapsed

Endif

4.83.2 Definition of the sub-function tasks

4.83.2.1 Fast Engine synchronization

General information:

Fast engine synchronization is in charge to synchronize as soon as the crankshaft signal acquisition is synchronized and a camshaft signal information is available on the selected camshaft.

Application conditions:

Recurrence: on 0 to 1 transition of LV_CAM_SYN_CRK

Formula section:

If LV_CAM_SYN_CRK = 1

Then LV_SYN_ENG = 1

Endif

4.83.2.2 Slow Engine synchronization

General information:

Slow engine synchronization is activated when a failure was detected on at least one camshaft sensor. The engine is synchronized as soon as a number of calibratable camshaft edges has occured (engine synchronized when synchronization is validated). This avoids multiple injection and ignition events with wrong synchronization.

Application conditions:

Recurrence: at every active camshaft edge and on 0 to 1 transition of LV_CAM_SYN_CRK

The active camshaft edge for synchronization is defined by NC_ACT_CAM_EDGE_SYN.

Formula section:

Increment CTR_VLD_CAM_SYN_CRK

If LV_CAM_SYN_CRK = 1

Then **If** CTR_VLD_CAM_SYN_CRK > C_NR_VLD_CAM_SYN_CRK

Then LV_SYN_ENG = 1

Endif

Endif

4.83.2.3 Crankshaft limp-home synchronization

General information:

Only the informations of the selected camshaft signal is used to perform the engine synchronization.
i = 1...NC_NR_CAM_CBK   z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Recurrence: at every camshaft edge

Formula section:

If   LV_SYN_CAM_IN(EX)[i] = 1 for the selected camshaft signal
Then   LV_FIRST_VLD_TOOTH = 1
    LV_RUN_ENG = 1
    LV_SYN_ENG = 1
Endif

4.83.2.4 Camshaft limp-home synchronization

General information:

No camshaft signals are available for the Crank/Cam synchronization so the engine synchronization is performed as soon as the crankshaft is synchronized (the engine revolution can not be identified so a default value is used).

Application conditions:

Recurrence: at every reference gap

Formula section:

If   LV_CRK_SYN = 1
Then   LV_SYN_ENG = 1
Endif

4.83.2.5 Synchronisation validation

General information:

Synchronisation is validated as soon as a number of calibratable camshaft edges has occured. During validation phase, the angular distance between cam/crk events is checked. Engine synchronisation is reset if one of the tests fails.
i = 1...NC_NR_CAM_CBK   z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Recurrence: *at every active camshaft edge and on 0 to 1 transition of LV_CAM_SYN_CRK as long as LV_SYN_VLD = 0*

The active camshaft edge for synchronization is defined by NC_ACT_CAM_EDGE_SYN.

Formula section:

Increment CTR_VLD_CAM_SYN_CRK

If LV_CAM_SYN_CRK = 1

Then **If** CTR_VLD_CAM_SYN_CRK > C_NR_VLD_CAM_SYN_CRK

Then LV_SYN_VLD = 1

Deactivation of Camshaft/Crankshaft synchronisation for the camshaft signal selected for this synchronization:

LV_ACT_SYN_CRK_CAM_IN(EX)[i] = 0

Endif

Endif

4.83.2.6 Synchronisation validation in camshaft limp home

General information:

Synchronization validation information is set as soon as it is requested by “Synchronization determination in camshaft limp home” module which is then deactivated.

Application conditions:

Initialisation: none

Recurrence: each segment event

Formula section:

If LV_SYN_VLD_CAM_LIH = 1
Then LV_SYN_VLD = 1
Deactivation of Synchronization determination in camshaft limp home()
End if

4.83.2.7 Camshaft Selection for synchronisation

General information:

Target wheels on intake and exhaust camshafts may be different, and all may be useable for synchronization. A camshaft signal can only be selected if no error is detected on the respective camshaft (LV_ERR_CAM_IN(EX)[i] = 0). For port-injection engines activation of synchronization mode on intake camshafts should be preferred. Pre-injection needs to be synchronized with intake timing, and the engine position for pre-injection is calculated from the synchronization signal.

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Formula section:

The selection order is given by the following table:

	NC_PRI_SYN_CAM_CBK = 1	NC_PRI_SYN_CAM_CBK = 2
NC_PRI_SYN_CAM_IN = 0	CAM_EX_1 CAM_EX_2 CAM_IN_1 CAM_IN_2	CAM_EX_2 CAM_EX_1 CAM_IN_2 CAM_IN_1
NC_PRI_SYN_CAM_IN = 1	CAM_IN_1 CAM_IN_2 CAM_EX_1 CAM_EX_2	CAM_IN_2 CAM_IN_1 CAM_EX_2 CAM_EX_1

The first camshaft sensor in the above list which fulfills the following conditions is selected:

- physically present in the system (configuration data NLC_CAM_IN, NLC_CAM_EX and NC_NR_CAM_CBK)
- no failure reported on it (LV_ERR_CAM_IN(EX)[i] = 0)

The crankshaft synchronization mode is activated on one camshaft signal by setting LV_ACT_SYN_CRK_CAM_IN(EX)[i] for the selected camshaft.

LV_ERR_CAM is set if all available camshaft sensors report and error.

4.83.2.8 Camshaft Selection for limp-home

General information:

Target wheels on intake and exhaust camshafts may be different, and not all may be useable for limp-home. The availability of camshaft sensors for limp-home is given by NLC_LIH_CAM_IN(EX) and NC_NR_CAM_CBK. A camshaft signal can only be selected if no error is detected on the respective camshaft (LV_ERR_CAM_IN(EX)[i] = 0).

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Formula section:

The selection order is given by the following table:

	NC_PRI_LIH_CAM_CBK = 1	NC_PRI_LIH_CAM_CBK = 2
NC_PRI_LIH_CAM_IN = 0	CAM_EX_1 CAM_EX_2	CAM_EX_2 CAM_EX_1

	CAM_IN_1 CAM_IN_2	CAM_IN_2 CAM_IN_1
NC_PRI_LIH_CAM_IN = 1	CAM_IN_1 CAM_IN_2 CAM_EX_1 CAM_EX_2	CAM_IN_2 CAM_IN_1 CAM_EX_2 CAM_EX_1

The first camshaft sensor in the above list which fulfills the following conditions is selected:

- physically present in the system (configuration data NLC_CAM_IN, NLC_CAM_EX_IN and NC_NR_CAM_CBK)
- no failure reported on it (LV_ERR_CAM_IN(EX)[i] = 0)
- available for crankshaft limp-home (NLC_LIH_CAM_IN = 1 or NLC_LIH_CAM_EX = 1)

The crankshaft limp-home mode is activated on one camshaft signal by setting LV_ACT_LIH_CRK_CAM_IN(EX)[i] for the selected camshaft.

LV_ERR_LIH_CRK_CAM is set if no more camshaft sensors is available.

4.83.2.9 Engine stop detection

Application conditions:

Recurrence: 10ms

Formula section:

LV_STOP_ENG is set if one of the conditions in the table below are met:

In Standard mode (States 3 and 4)	In Camshaft limp-home mode (States 7 and 8)	In Crankshaft limp-home mode (States 5 and 6)	In Engine position Standard and Camshaft limp-home mode (States 4 and 8)
LV_CRK_STOP = 1	LV_CRK_STOP = 1	LV_CAM_STOP_IN(EX)[i] = 1 for the selected camshaft signal	LV_ENG_BACK_CFM switches from 0 to 1
LV_ORNG_CAM_SYN_CRK = 1 (*)			
LV_ERR_CRK = 1 And LC_INH_LIH_CRK = 0			

(*) This information is available as long as LV_SYN_VLD = 0.

4.83.2.10 Camshaft self-synchronisation reactivation

General information:

When the self-synchronization of a camshaft signal goes to the fiasco state (LV_CAM_STOP_IN(EX)[i] = 1), the self-synchronization on this camshaft signal will be reactivated as soon as possible. If this camshaft is selected for the synchronization in crankshaft limp-home mode an engine stop is detected (by *engine stop detection()* sub-function task) and a new synchronization is restarted.

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Recurrence: 10ms

Formula section:

For all available camshaft signal sensors:

If LV_CAM_STOP_IN(EX)[i] = 1
And LV_ACT_LIH_CRK_CAM_IN(EX)[i] = 0
Then Activation of Camshaft self-synchronisation for this camshaft signal
Endif

4.83.2.11 *Calculation of PSN_ENG_CRK in standard mode()*

General information:

Formula section:

Before synchronization (LV_SYN_ENG = 0):

Truth table					
LV_CRK_SYN n	0	1			
LV_CRK_SYN n-1	0	0		1	
LV_CAM_SYN_CRK n	0	0	1	0	1
LV_CAM_SYN_CRK n-1	0	0	0	0	0
Engine position action	**1**	**2**	**3**	**4**	**5**

Action 1:

PSN_ENG_CRK = PSN_ENG_SYN_CAM_MIN

Action 2:

PSN_ENG_CRK = PSN_ENG_SYN_CAM_MIN

PSN_ENG_CRK_GAP = 360° - NC_OFS_TDC0_REF_CRK

Action 3:

PSN_ENG_CRK = 360° - NC_OFS_TDC0_REF_CRK + PSN_ENG_CRK_OFS

Action 4:

PSN_ENG_CRK = PSN_ENG_SYN_CAM_MIN

PSN_ENG_CRK_GAP n = PSN_ENG_CRK_GAP n-1 + (360°/NC_NR_TOOTH)

Action 5:

PSN_ENG_CRK = PSN_ENG_CRK_GAP + PSN_ENG_CRK_OFS

After the synchronization (LV_SYN_ENG = 1):

PSN_ENG_CRK is incremented on the tooth signal and jitter from 0 to 720°.

4.83.2.12 *Calculation of PSN_ENG_CRK in camshaft limp-home mode()*

General information:

Formula section:

Before synchronization (LV_SYN_ENG = 0):

Truth table		
LV_CRK_SYN n	0	1
LV_CRK_SYN n-1	0	0
Engine position action	**1**	**2**

Action 1:

PSN_ENG_CRK is incremented on the tooth signal and jitter from 0 to 360°

Action 2:

PSN_ENG_CRK = 720° - NC_OFS_TDC0_REF_CRK

After the synchronization (LV_SYN_ENG = 1):

PSN_ENG_CRK is incremented on the tooth signal and jitter from 0 to 720°

4.83.2.13 *PSN_ENG_CRK correction in camshaft limp-home()*

General information:

This function updates PSN_ENG_CRK depending on synchronization determination test result in camshaft limp home.

Application conditions:

Initialisation: none

Recurrence: none

Activation: LV_SEG_NR_UPD_REQ = 1

Deactivation: LV_SEG_NR_UPD_REQ = 0

Formula section:

PSN_ENG_CRK = PSN_ENG_CRK + (360°CRK / NC_NR_GAP)

4.83.2.14 *Calculation of PSN_ENG_CAM_LIH_CRK()*

General information:

The purpose of this function is to provide engine position during crankshaft limp-home. In crankshaft limp-home, the engine position is calculated from the camshaft sensor selected by LV_ACT_LIH_CRK_CAM_IN(EX)[i].

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Initialization: PSN_ENG_CAM_LIH_CRK = 0

Recurrence: every camshaft signal edge

Activation: LV_SYN_ENG = 1

Deactivation: LV_SYN_ENG = 0

Formula section:

For z = IDX_EDGE_CAM_IN(EX)[i] and the camshaft sensor selected for crankshaft limp home by LV_ACT_LIH_CRK_CAM_IN(EX)[i]:

If z is a falling camshaft signal edge

Then **If** NC_ACT_CAM_EDGE_LIH = 1

Or

NC_ACT_CAM_EDGE_LIH = 3

Then PSN_ENG_CAM_LIH_CRK = NC_PSN_EDGE_CAM_IN(EX)[z][i] +

Average value of PSN_EDGE_AD_CAM_IN(EX)[z][i] for all falling camshaft signal edges

Else *no update of PSN_ENG_CAM_LIH_CRK*

Endif

Else **If** NC_ACT_CAM_EDGE_LIH = 2

Or

NC_ACT_CAM_EDGE_LIH = 3

Then PSN_ENG_CAM_LIH_CRK = NC_PSN_EDGE_CAM_IN(EX)[z][i] +

Average value of PSN_EDGE_AD_CAM_IN(EX)[z][i] for all rising camshaft signal edges

Else *no update of PSN_ENG_CAM_LIH_CRK*

Endif

End

4.83.2.15 Engine backwards rotation detection ()

General information:

The aim of this function is to detect a change in the direction of engine rotation which would lead to a false crankshaft teeth counter value and thus to a wrong timing for injection and/or ignition events. This could occur at low engine speed with hard compression phases.
This functionality gets a lower performance if camshaft target wheel is symmetrical as only the tests on crankshaft signal can be done (Action 1 and Test 2).
i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Initialisation: at reset or engine stalling (LV_STOP_ENG transition from 0 to 1)
LV_ENG_BACK_DET = 0
LV_ENG_BACK_CFM = 0

Recurrence: 10ms and every camshaft edge (for all camshaft sensors) and every reference gap and at engine stalling (LV_STOP_ENG transition from 0 to 1)

Activation: LC_ENG_BACK_INH = 0

Deactivation: when application conditions are not fulfilled

Formula section:

If **(** N_TOOTH < C_N_NOT_REST **And** LV_CRK_MISS_TOOTH = 1 **)**

or LV_ENG_BACK_DET = 1

Then LV_ENG_BACK_DET = 1

Activation of Engine backwards rotation confirmation ()

Else LV_ENG_BACK_DET = 0

LV_ENG_BACK_CFM = 0

End

LV_CRK_MISS_TOOTH is reset after reading.

At transition from 0 to 1 of LV_ENG_BACK_DET, CTR_TEST_ENG_BACK_RST = C_NR_TEST_ENG_BACK_RST.

Engine backwards rotation confirmation ():

The following table shows the test or action to perform in function of the current event.The event is a detection of a camshaft edge (CAM) or a detection of reference gap (GAP) or an engine stall (Engine stall).

Event	GAP	Engine stall	CAM
Test / Action	Action 1	Test 2	Test 3

Action 1:

This event means that the reference gap has been detected at the expected position (the number of teeth between 2 gaps is correct) and that the number of camshaft edges over last engine cycle is correct for all camshaft sensors and thus that engine backwards rotation is not confirmed.

If abs(CTR_EDGE_CAM_IN(EX)[i] - CTR_EDGE_CAM_IN(EX)[i] at GAP(n-2)) = NC_NR_EDGE_CAM_IN(EX)

for all available CAM sensors with LV_ERR_CAM_IN(EX)[i] = 0

/* number of camshaft edges over last engine cycle equal to theoritical one */

Then CTR_TEST_ENG_BACK_RST = CTR_TEST_ENG_BACK_RST - 1

Endif

If CTR_TEST_ENG_BACK_RST = 0

Then LV_ENG_BACK_DET = 0

Endif

Test 2:

This event means that the number of teeth between 2 gaps is not correct due to engine backwards rotation which is thus confirmed.

If LV_LOST_SYN_CRK = 1

Then LV_ENG_BACK_CFM = 1

Test 3:

This test is only done on the camshaft signal that generated the event. It checks wether camshaft signal position is correct versus crankshaft signal or versus previous camshaft signal. If not, engine backwards rotation is confirmed.

If LV_ERR_CAM_IN(EX)[i] = 0

Then **if** **{**

(LV_SYN_CAM_IN(EX)[i] = 1 'z value, which represents camshaft edge number, is know as camshaft is self-synchronized

And

(PSN_ENG_ENSD ≤ NC_PSN_EDGE_CAM_IN(EX)[z][i] - C_DYW_CAM_CRK_SYN_ADC_IN(EX)

Or PSN_ENG_ENSD ≥ NC_PSN_EDGE_CAM_IN(EX)[z][i] + C_DYW_CAM_CRK_SYN_RTD_IN(EX)) **)**

Or

LV_ORNG_RATIO_CAM_IN(EX)[i] = 1 **}**

Then LV_ENG_BACK_CFM = 1

Else **if** LV_SYN_CAM_IN(EX)[i] = 1

Then LV_ENG_BACK_DET = 0

End

End

End

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_INH_LIH_CRK	1	0...1H	0...1	1	[-]
Inhibition of crankshaft limp-home mode					
LC_INH_LIH_CAM	1	0...1H	0...1	1	[-]
Inhibition of camshaft limp-home mode					
C_NR_VLD_CAM_SYN_CRK	1	0...FFH	0...255	1	[-]
Number of camshaft signal edges to validate cam/crk synchronization					
C_T_DLY_STOP_ENG	1	0...FFH	0...2550	10	[ms]
Delay time for engine running after engine stop request					
C_NR_TEST_ENG_BACK_RST	1	1...FFH	1...255	1	[-]
Number of successful tests to reset backwards rotation detection					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PRI_SYN_CAM_IN	1	0...1H	0...1	1	[-]
Priority for synchronization mode on intake camshaft sensor(s)					
NC_PRI_SYN_CAM_CBK	1	1...2H	1...2	1	[-]
Priority for synchronization on camshaft sensor(s) of cylinder bank 1 or 2					
NC_PRI_LIH_CAM_IN	1	0...1H	0...1	1	[-]
Priority for limp-home mode on intake camshaft sensor(s)					
NC_PRI_LIH_CAM_CBK	1	1...2H	1...2	1	[-]
Priority for limp-home mode on camshaft sensor(s) of cylinder bank 1 or 2					
NLC_CAM_IN	1	0...1H	0...1	1	[-]
Intake camshaft sensor(s) present in the system					
NLC_CAM_EX	1	0...1H	0...1	1	[-]
Exhaust camshaft sensor(s) present in the system					
NC_NR_CAM_CBK	1	1...2H	1...2	1	[-]
Camshaft sensor(s) present on one or two cylinder banks					
NLC_LIH_CAM_IN	1	0...1H	0...1	1	[-]
Intake camshaft sensor(s) available for crankshaft limp-home					
NLC_LIH_CAM_EX	1	0...1H	0...1	1	[-]
Exhaust camshaft sensor(s) available for crankshaft limp-home					

4.84 Engine Position and Speed Calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_32	V/O	0...FFH	0...8160	32	[rpm]
Engine speed -Resolution 32 rpm					
N	V/O	0...1FE0H	0...8160	1	[rpm]
Engine speed - resolution 1 rpm					
N_FAST	V/O	0...1FE0H	0...8160	1	[rpm]
Fast engine speed					
N_TOOTH	V/O	0...1FE0H	0...8160	1	[rpm]
Engine Speed calculated on last crankshaft tooth period (update rate 10ms)					
N_MMV	V/O	0...1FE0H	0...8160	1	[rpm]
Engine Speed ¿ moving mean value					
N_CRK_WIN	-	0...1FE0H	0...8160	1	[rpm]
Engine speed on crankshaft tooth window at end of segment					
N_GRD	V/O	80...7FH	-4096...4064	32	[rpm/s]
Engine speed gradient					
N_GRD_H_RES	V/O	F000...0FFFH	-4096...4095	1	[rpm/s]
High resolution engine speed gradient					
T_REV_AV	-	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Actual revolution time					
T_REV_PREV	-	0...7FFFFFH	0...1.9965	0.238e-6	[s]
Previous revolution time					
T_SEG_AV	V/O	0...FFFFH	0...0.26214	0.000004	[s]
Segment period (saturated at low engine speed).					
T_SEG_HALF_AV	V/O	0...FFFFH	0...0.26214	0.000004	[s]
Exported half-segment period					
SEG_CTR	V/O	0...FFFFH	0...65535	1	[-]
Continuous segment counter (saturated at maximum value)					
SEG_NR	V/O	0...7H	0...7	1	[-]
Actual segment number (from 0 to NC_CYL_NR-1)					
PSN_ENG_SYN_MIN	V/O	0...780H	0...720	0.375	[°CRK]
Minimum engine position during crankshaft synchronization phase.					
PSN_ENG_SYN_MAX	V/O	0...780H	0...720	0.375	[°CRK]
Maximum engine position during crankshaft synchronization phase.					
PSN_ENG_ENSD	O	0...2CFFH	0...719.9375	0.0625	[°CRK]
Actual engine position in fine resolution					
PSN_ENG	O	0...780H	0...720	0.375	[°CRK]
Exported engine position					

Input data:

LV_RUN_ENG	LV_CRK_SYN	NC_NR_TOOTH	PSN_ENG_CRK
PSN_ENG_SYN_CAM_MIN	PSN_ENG_SYN_CAM_MAX	PSN_ENG_CAM_LIH_CRK	T_TOOTH
T_SEG_HALF_ENSD	T_CRK_WIN_ENSD	LV_STOP_ENG	NC_NR_TOOTH
NC_PSN_SEG_TDC_REF	NC_CRK_WIN_SEG_LEN	T_SEG_ENSD	LV_SEG_NR_UPD_REQ

FUNCTION DESCRIPTION:

At engine stalling (LV_STOP_ENG 0 to 1), all output data are set to initialization value.

4.84.1 Engine Position Calculation Standard Mode

General information:

The function provides output values for engine position in standard operating mode.

Application conditions:

Initialisation: PSN_ENG = PSN_ENG_ENSD = 0

Recurrence: every update of the corresponding input values

Formula section:

PSN_ENG = PSN_ENG_ENSD (resolution conversion)

PSN_ENG_ENSD = PSN_ENG_CRK + ID_CRK_PHA_COR

PSN_ENG_SYN_MIN = PSN_ENG_SYN_CAM_MIN

PSN_ENG_SYN_MAX = PSN_ENG_SYN_CAM_MAX

The resolution of variables that are exported outside the aggregate is different from the resolution used internally!

4.84.2 Engine Position Calculation Crankshaft Limp-Home Mode

General information:

The function provides output values for engine position in crankshaft limp-home mode.

Application conditions:

Initialisation: PSN_ENG = PSN_ENG_ENSD = 0

Recurrence: every update of the corresponding input values

Formula section:

PSN_ENG = PSN_ENG_ENSD (resolution conversion)

PSN_ENG_ENSD = PSN_ENG_CAM_LIH_CRK

PSN_ENG_SYN_MIN = PSN_ENG_SYN_CAM_MIN

PSN_ENG_SYN_MAX = PSN_ENG_SYN_CAM_MAX

The resolution of variables that are exported outside the aggregate is different from the resolution used internally!

4.84.3 Segment Period Calculation

General information:

The function provides output values for the segment period.

The engine cycle is divided into a number of TDC periods corresponding to the number of cylinders NC_CYL_NR. The corresponding time intervals T_SEG_AV are measured between tooth events located NC_PSN_SEG_TDC_REF degrees before each TDC.

The segment number SEG_NR indicates the number of the cylinder that will be next in the firing order. A control signal is generated to trigger execution of segment synchronous tasks.

Application conditions:

Initialisation: T_SEG_AV is set to the maximum value

SEG_CTR = 0

SEG_NR = 0

Activation: LV_RUN_ENG = 1

Recurrence : when LV_RUN_ENG is set
And when LV_CRK_SYN is set
And every segment trigger

Formula section:

T_SEG_AV = T_SEG_ENSD saturated to maximum value at low speed.

At engine synchronization, SEG_NR is initialized with current segment number.

SEG_CTR is incremented at every segment.

4.84.4 Segment number correction in camshaft limp home

General information:

This function updates SEG_NR value in camshaft limp home mode depending on synchronization determination test result.

Application conditions:

Initialisation: none

Recurrence: none

Activation: LV_SEG_NR_UPD_REQ = 1

Deactivation: LV_SEG_NR_UPD_REQ = 0

Formula section:

SEG_NR = SEG_NR + NC_CYL_NR / (2 * NC_NR_GAP)

4.84.5 Engine speed N, N_32, N_MMV

General information:

The engine speed is determined in each segment from the segment period T_SEG_ENSD.

Application conditions:

Activation: LV_RUN_ENG = 1

Recurrence: when LV_RUN_ENG is set
And when LV_CRK_SYN is set
And every segment trigger

Formula section:

N = $\frac{2*60}{NC\_CYL\_NR*T\_SEG\_ENSD}$ for T_SEG_ENSD in s

N_32 = N in resolution 32 rpm

Before synchronization and at synchronization, one engine speed is output at the moment when crankshaft rotation is validated. The output value will be calculated from the most recent tooth period (see “Segment Period Calculation”)

N_MMV is the moving mean value calculated on the last 720° CRK:

N_MMV = $\frac{N_n + N_{n-1} + \ldots + N_{n-NC\_CYL\_NR}}{NC\_CYL\_NR}$

Before engine synchronisation N_MMV = N. After engine synchronisation and before the first engine cycle is completed, N_MMV is calculated as the mean value of the most recent segments of this cycle.

Set N_FAST to N if no fast engine speed calculation is required.

If C_N_FAST_SWI = 0

Then N_FAST = N

Endif

4.84.6 Fast Engine speed based on last tooth period

General information:

Provide engine speed calculated every 10ms on the last crankshaft tooth period during cranking, synchronisation and start phase. Calculation is done until speed threshold is reached.
If, during running, Engine speed becomes lower than this threshold, this calculation is started again
In the formula after, T_TOOTH in s is the time for the last crankshaft tooth period, acquired in BSW just before the 10ms trigger event used for update rate of N_TOOTH here.

Application conditions:

Initialisation: N_TOOTH = 0

Recurrence : 10ms

Activation: LV_RUN_ENG = 1

Deactivation: LV_RUN_ENG = 0

Formula section:

If N_32 <= C_N_TOOTH_END

Then N_TOOTH = 60 / (NC_NR_TOOTH * T_TOOTH)

Endif

If C_N_FAST_SWI = 1

Then N_FAST = N_TOOTH

Endif

4.84.7 Fast Engine Speed based on Half-Segment Period Calculation

General information:

This function provides fast engine speed based on half-segment period.

Application conditions:

Initialisation: T_SEG_HALF_AV is set to the maximum value

Recurrence: before synchronization: once at transition from LV_RUN_ENG from 0 to 1
after synchronization: every ½ segment trigger **and** segment trigger

Activation: LV_RUN_ENG = 1

Formula section:

If C_N_FAST_SWI = 2

Then N_FAST = $\frac{60}{NC\_CYL\_NR * T\_SEG\_HALF\_ENSD}$

Endif

T_SEG_HALF_AV = T_SEG_HALF_ENSD saturated to maximum value at low speed.

4.84.8 Fast Engine Speed based on Crankshaft Tooth Window

General information:

This function provides fast engine speed output based on a crankshaft window at the end of each segment. Updated every segment

Application conditions:

Initialisation: N_FAST = 0

Recurrence : every segment trigger

Activation: LV_RUN_ENG = 1
And
(C_N_FAST_SWI = 3 **Or** C_N_FAST_SWI = 4)

Deactivation: C_N_FAST_SWI <> 3 **And** C_N_FAST_SWI <> 4

Formula section:

N_CRK_WIN = $\frac{60}{T\_CRK\_WIN\_ENSD} \quad \frac{NC\_CRK\_WIN\_SEG\_LEN}{360° CRK}$

If C_N_FAST_SWI = 3

Then N_FAST = N + C_FAC_N_FAST * (N_CRK_WIN n – N_CRK_WIN n-1)

Endif

If C_N_FAST_SWI = 4

Then N_FAST = N + C_FAC_N_FAST * (N n – N_CRK_WIN n-1)

Endif

4.84.9 Engine speed gradient N_GRD, N_GRD_H_RES

General information:

The engine speed gradient is the acceleration of the crankshaft in rpm/s. For engins with an even number of cylinders (2...8), the gradient is calculated from the duration of the most recent engine revolution, and the duration of the previous engine revolution. With this calculation, the gradient is free from errors due to mechanical tolerances of the crankshaft target wheel.

For three-cylinder engins, the gradient is calculated from the most recent segment time, and the segment time before. Errors from mechanical tolerances of the crankshaft target wheel are not corrected, but are supposed to have a minor effect due to the greater angle of the segments.

Application recurrence: every segment trigger

Formula section:

For even cylinder numbers, the duration of the most recent engine revolution is the sum of the most recent NC_CYL_NR/2 segment times. For e.g. a four-cylinder engine, this is:

$$T\_REV\_AV = T\_SEG\_ENSD_{n} + T\_SEG\_ENSD_{n-1}$$

The duration of the previous engine revolution is for the example of a four-cylinder engine:

$$T\_REV\_PREV = T\_SEG\_ENSD_{n-2} + T\_SEG\_ENSD_{n-3}$$

N_GRD calculation:

N_GRD_H_RES =

$$\frac{60*(T\_REV\_PREV - T\_REV\_AV)}{T\_REV\_PREV * T\_REV\_AV * 0.5*(T\_REV\_PREV + T\_REV\_AV)}$$

for T_REV_AV and T_REV_PREV in s

N_GRD = N_GRD_H_RES in resolution 32 rpm/s.

The resolution of the segment time introduces an error into the calculation of the engine speed gradient, depending on engine speed.

For a timer resolution of 0.25µs, a difference of 1 timer tick between T_REV_AV and T_REV_PREV corresponds to:

0.015 rpm/s at 600 rpm

and 15 rpm/s at 6000 rpm

For three-cylinder engins, the gradient is calculated as follows:

N_GRD_H_RES =

$$\frac{2*60*(T\_SEG\_ENSD_{n-1} - T\_SEG\_ENSD_{n})}{3*T\_SEG\_ENSD_{n-1}*T\_SEG\_ENSD_{n}*0.5*(T\_SEG\_ENSD_{n-1} + T\_SEG\_ENSD_{n})}$$

N_GRD = N_GRD_H_RES in resolution 32 rpm/s.

The resolution of the segment time introduces an error into the calculation of the engine speed gradient, depending on engine speed.

A difference of 1 timer tick between $T\_SEG\_ENSD_{n}$ and $T\_SEG\_ENSD_{n-1}$ corresponds to:

0.51 rpm/s at 600 rpm

and 51 rpm/s at 6000 rpm

4.84.10 Crankshaft Sensor Phase Angle Correction

General information:

The sensor phase angle is defined as the rotation angle of the target wheel between the mechanical reference and the electrical reference for a tooth of the target wheel.

The mechanical reference in the target wheel drawing usually is the trailing edge of a tooth in the defined rotation direction. The mechanical reference position is reached when the center of the sensor is aligned to that angular reference.

The electrical reference may be on the falling or on the rising edge of the sensor output signal, see active edge definition in the acquisition of crankshaft signal. The reference position is reached when the sensor output signal switches (for active sensors) respectively has its zero crossing (for VR sensors).

A retard of the electrical reference relative to the mechanical edge is defined as a positive phase angle. With most sensors, the signal edge is generated at the center of each tooth at low speed. At high speed the signal edge may be retarded.

The typical phase angle ID_CRK_PHA_COR (at typical air gap and operating temperature) of the used sensor/target wheel combination is calibrated in a table versus engine speed. The value corresponding to the actual engine speed is added to the actual engine position delivered from crankshaft acquisition, in order to obtain a corrected engine position (see position calculation in the first chapter of this specification).

Application conditions:

Recurrence: every segment trigger

Activation: LV_CRK_SYN = 1

Deactivation: LV_CRK_SYN = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_TOOTH_END	1	0...FFH	0...8160	32	[rpm]
Engine speed threshold for N_TOOTH calculation at engine start					
C_N_FAST_SWI	1	0...4H	0...4	1	[-]
Switch case choice for N_FAST calculation mode					
C_FAC_N_FAST	1	0...FFH	0...0.99609	3.91E-03	[-]
Factor for fast engine speed prediction					
ID_CRK_PHA_COR	16	0...FFH	0...15.9375	0.0625	[°CRK]
LDP_N_32_ID_CRK_PHA_COR	16	0...FFH	0...8160	32	[rpm]
Table for crankshaft transmitter phase offset correction					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CYL_NR	1	1...8H	1...8	1	[-]
count of cylinder					

4.85 Engine Position and Speed Calculation (App. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_32_MAX_DC	V/O	0...FFH	0...8160	32	rpm
Maximum engine speed during actual driving cycle – Resolution 32 rpm					
N_32_MAX_TOT_DC	V/O/S	0...FFH	0...8160	32	rpm
Maximum engine speed during former / current driving cycle – Resolution 32 rpm					

Input data:

LV_RUN_ENG	N_32		

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

N_32_MAX_TOT_DC = 0H

- otherwise: N_32_MAX_TOT_DC restored form non-volantile memory
- at LV_IGK 0 ->1 and reset: N_32_MAX_DC = 0

Recurrence: segment synchronous

Activation: LV_RUN_ENG = 1

Deactivation: -

Formula section:

If N_32 > N_32_MAX_DC

Then N_32_MAX_DC = N_32

Endif

If N_32 > N_32_MAX_TOT_DC

Then N_32_MAX_TOT_DC = N_32

Endif

4.86 Synchronization determination in camshaft limp home

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CAM_LIH_EXT_ENA	V/O	0...1H	0...1	1	[-]
Camshaft limp home with external help enabled					
LV_SEG_NR_UPD_REQ	V/O	0...1H	0...1	1	[-]
SEG_NR update request					
LV_SYN_VLD_CAM_LIH	V/O	0...1H	0...1	1	[-]
Engine synchronization determined in camshaft limp home mode					

Input data:

N	N_GRD	SEG_CTR	LV_VS_RUN
INH_INJ	TCO_CMN	VB_CMN	SEG_NR
NC_CAM_LIH_SWI			

FUNCTION DESCRIPTION:

General information:

This functions aims at initializing the variables used by ENSD or others aggregates in case no camshaft limp home method has been selected.

Description:

Application conditions:

Initialisation: none

Recurrence: at reset

Activation: NC_CAM_LIH_SWI = “None”

Deactivation: NC_CAM_LIH_SWI ≠ “None”

Formula section:

LV_SYN_VLD_CAM_LIH = 0
LV_CAM_LIH_EXT_ENA = 0
LV_SEG_NR_UPD_REQ = 0

Remark: as this module is only a stub, N, N_GRD, SEG_CTR, LV_VS_RUN, INH_INJ, TCO_CMN, VB_CMN and SEG_NR are not used.

4.87 Camshaft adaptation and position output

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
PSN_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1FFH	-96...95.625	0.375	[°CRK]
Actual intake camshaft position relative to adapted passive position					
PSN_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1FFH	-96...95.625	0.375	[°CRK]
Actual exhaust camshaft position relative to adapted passive position					
PSN_EDGE_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	O	0...FFFH	-128...127.9375	0.0625	[°CRK]
Actual intake camshaft position relative to adapted passive position at signal edge index z					
PSN_EDGE_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	O	0...FFFH	-128...127.9375	0.0625	[°CRK]
Actual exhaust camshaft position relative to adapted passive position at signal edge index z					
PSN_EDGE_AD_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	V/O/S	0...3FFH	-32...31.9375	0.0625	[°CRK]
Adapted intake camshaft i signal edge z position					
PSN_EDGE_AD_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	V/O/S	0...3FFH	-32...31.9375	0.0625	[°CRK]
Adapted exhaust camshaft i signal edge z position					
PSN_AD_CAM_IN[NC_NR_CAM_CBK]	V/O	0...FFH	-48...47.625	0.375	[°CRK]
Adapted intake camshaft position relative to designed passive position					
PSN_AD_CAM_EX[NC_NR_CAM_CBK]	V/O	0...FFH	-48...47.625	0.375	[°CRK]
Adapted exhaust camshaft position relative to designed passive position					
PSN_DIF_EDGE_CAM_IN[NC_NR_CAM_CBK]	V/O	0...780H	0...720	0.375	[°CRK]
Crankshaft angle between the previous and the current intake camshaft position determination					
PSN_DIF_EDGE_CAM_EX[NC_NR_CAM_CBK]	V/O	0...780H	0...720	0.375	[°CRK]
Crankshaft angle between the previous and the current exhaust camshaft i position determination					
T_DIF_EDGE_CAM_IN[NC_NR_CAM_CBK]	V/O	0...FFFFFFFFH	0...17179.86918	0.000004	[s]
Time between the previous and current camshaft signal edge					
T_DIF_EDGE_CAM_EX[NC_NR_CAM_CBK]	V/O	0...FFFFFFFFH	0...17179.86918	0.000004	[s]
Time between the previous and current camshaft signal edge					
CTR_REV_AD_REF_CAM_IN[NC_NR_CAM_CBK]	V	0...FFH	0...255	1	[-]
Intake camshaft reference position adaptation revolution counter					
CTR_REV_AD_REF_CAM_EX[NC_NR_CAM_CBK]	V	0...FFH	0...255	1	[-]
Exhaust camshaft reference position adaptation revolution counter					
LV_AD_END_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Intake camshaft reference position adaptation successful					
LV_AD_END_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Exhaust camshaft reference position adaptation successful					
PSN_ENG_CAM_IN[NC_NR_CAM_CBK]	-	0...2CFFH	0...719.9375	0.0625	[°CRK]
Engine position at the last intake camshaft i edge					
PSN_ENG_CAM_EX[NC_NR_CAM_CBK]	-	0...2CFFH	0...719.9375	0.0625	[°CRK]
Engine position at the last exhaust camshaft i edge					
CAM_PSN_LST_REF_AD_IN[NC_NR_CAM_CBK]	V/O	0...3FFH	-32...31.9375	0.0625	[°CRK]
Adapted intake camshaft i signal position after last correct reference adaptation					
CAM_PSN_LST_REF_AD_EX[NC_NR_CAM_CBK]	V/O	0...3FFH	-32...31.9375	0.0625	[°CRK]
Adapted exhaust camshaft i signal position after last correct reference adaptation					
LV_CAM_AD_SAVE_IN[NC_NR_CAM_CBK]	V/O/S	0...1H	0...1	1	[-]
Confirmation that adaptative values at intake i have already been stored					

LV_CAM_AD_SAVE_EX[NC_NR_CAM_CBK]	V/O/S	0...1H	0...1	1	[-]
Confirmation that adaptative values at exhaust i have already been stored					
LV_CAM_AD_PWL_NOT_SAVE_IN[NC_NR_CAM_CBK]	-	0...1H	0...1	1	[-]
Camshaft adaptive values storage in NVMY not allowed, for intake i					
LV_CAM_AD_PWL_NOT_SAVE_EX[NC_NR_CAM_CBK]	-	0...1H	0...1	1	[-]
Camshaft adaptive values storage in NVMY not allowed, for exhaust i					
LV_TOOTH_OFF_DET_ENA_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Confirmation that a reference adaptation occurred after the very first one, to allow the one_tooth_off detection (intake i)					
LV_TOOTH_OFF_DET_ENA_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Confirmation that a reference adaptation occurred after the very first one, to allow the one_tooth_off detection (exhaust i)					

Input data:

LV_DI_AD_REF_CAM_IVVT_IN	LV_DI_AD_REF_CAM_IVVT_EX	IDX_EDGE_CAM_IN[NC_NR_CAM_CBK]	IDX_EDGE_CAM_EX[NC_NR_CAM_CBK]
NC_NR_EDGE_CAM_IN	NC_NR_EDGE_CAM_EX	LV_VLD_PSN_CAM_IN[NC_NR_CAM_CBK]	LV_VLD_PSN_CAM_EX[NC_NR_CAM_CBK]
NC_PSN_EDGE_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	NC_PSN_EDGE_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	PSN_ENG_ENSD	LV_SYN_VLD
LC_NOT_ADJ_CAM_IVVT_IN[NC_NR_CBK_IVVT]	LC_NOT_ADJ_CAM_IVVT_EX[NC_NR_CBK_IVVT]	NLC_IVVT_IN	NLC_IVVT_EX
T_SEG_CAM_IN[NC_NR_CAM_CBK]	T_SEG_CAM_EX[NC_NR_CAM_CBK]	N_32	
	C_TOL_REF_CRK_CAM_MIN_IN	C_TOL_REF_CRK_CAM_MIN_EX	C_TOL_REF_CRK_CAM_MAX_IN
C_TOL_REF_CRK_CAM_MAX_EX			

FUNCTION DESCRIPTION:

General information:

The function provides a feedback about the actual camshaft position relative to the engine position measured with the crankshaft sensor. The signals of one, two or four camshaft sensors are treated (intake and/or exhaust camshaft on one or two cylinder banks).

The actual engine position is captured with every camshaft signal edge. The camshaft position is calculated as the difference between the captured engine position, and the engine position measured on the respective camshaft. The difference may be due to tolerances, or due to an intentional displacement by a camshaft phasing system (VVTI).

Two different adaptive learning algorithms eliminate the tolerances, in order to measure exactly the VVTI displacement.

The reference position adaptation is done for all signal edges. It is performed on all intake camshafts as long as LV_DI_AD_REF_CAM_IVVT_IN = 0. It is performed on all exhaust camshafts as long as LV_DI_AD_REF_CAM__IVVT_EX = 0. The camshafts are then situated in their respective VVTI passive positions. After the completion of the adaptation LV_AD_END_CAM_IN(EX)[i] is set and PSN_AD_CAM_IN(EX)[i] is calculated as the mean value of all adaptated signal edges. The reference edge has the index 1. It is the first electrical falling edge of the camshaft signal after TDC0.

The position of other signal edges is adapted continuously relative to the reference signal edge, in order to always deliver a coherent position information.

Both adaptation algorithms are performed only at a limited dynamic variation of the camshaft position measured at the reference edge.

The logical calibration bit LC_NOT_ADJ_CAM_IVVT_IN(EX)[i] is set if the VVTI system is inhibited on the corresponding camshafts. The respective camshafts will be treated like being constantly in the VVTI passive position. If an engine has more camshaft sensors then VVTI actuators, then NLC_IVVT_IN(EX) = 0 indicates that VVTI is not existing on intake or on exhaust camshafts.

4.87.1 Engine Position Determination from Camshaft signal

General information:

As soon as camshaft is self-synchronized, the actual engine position is calculated at every camshaft signal edge from theorical position plus adaptation value for the actual signal edge index.

i = 1...NC_NR_CAM_CBK   z = 0...NC_NR_EDGE_CAM_IN/EX

Description:

Recurrence :   *every CAM active edge*

Formula section:

for z = IDX_EDGE_CAM_IN(EX)[i]

PSN_ENG_CAM_IN(EX)[i] (n) = NC_PSN_EDGE_CAM_IN(EX)[z][i] +
PSN_EDGE_AD_CAM_IN(EX)[z][i] (n-1) +
ID_CAM_PHA_COR

4.87.2 Camshaft Position Output

Application conditions:

Activation:

LV_VLD_PSN_CAM_IN(EX)[i] = 1 **And** LV_SYN_VLD = 1

Recurrence: *every update of IDX_EDGE_CAM_IN(EX)[i]*

Initialization:

PSN_CAM_IN(EX)[i] = 0
PSN_EDGE_CAM_IN(EX)[z][i] = 0

Formula section:

for z = IDX_EDGE_CAM_IN(EX)[i]

PSN_EDGE_CAM_IN(EX)[z][i] = PSN_ENG_ENSD - PSN_ENG_CAM_IN(EX)[i]

In the following calculations, the resolution has to be converted:

PSN_CAM_IN(EX)[i] = PSN_EDGE_CAM_IN(EX)[z][i]

PSN_DIF_EDGE_CAM_IN(EX)[i] = PSN_ENG_ENSD (n) - PSN_ENG_ENSD (n-1)

T_DIF_EDGE_CAM_IN(EX)[i] = T_SEG_CAM_IN(EX)[i]

4.87.3 Camshaft Position Adaptation

4.87.3.1 Reference Position Adaptation

General information:

The reference position adaptation is done whenever it is not disabled by the corresponding flag from VVTI, or if there is no VVTI on that camshaft, and if the condition for limited dynamic variation of the camshaft position is true. A flag is set to signal successful adaptation.

The flag is reset when the adaptation inhibit conditions become false.

If LV_DI_AD_REF_CAM_IVVT_IN(EX) = 1

Then LV_AD_END_CAM_IN(EX)[i] = 0

The camshaft position output value should be zero in VVTI passive position. At each ocurrence of the reference camshaft signal edge, a portion of the position deviation is added to the previous adaptation value, as long as adaptation conditions are true, this is done for all the camshaft signal edges.

If the conditions stay true for a number of revolutions, then the flag for successful end of adaptation is set (LV_AD_END_CAM_IN(EX)[i] = 1).

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Application conditions:

Activation:

LV_VLD_PSN_CAM_IN(EX)_I =1 **And** LV_SYN_VLD = 1

And

LV_DI_AD_REF_CAM_IVVT_IN(EX) = 0 **Or** LC_NOT_ADJ_CAM_IVVT_IN(EX)[i] = 1 **Or** NLC_IVVT_IN(EX) = 0

Recurrence: *every reference camshaft signal edge (IDX_EDGE_CAM_IN(EX)[i] = 1)*

Initialization:

LV_AD_END_CAM_IN(EX)[i] = 0

CTR_REV_AD_REF_CAM_IN(EX)[i] = 0

PSN_EDGE_AD_CAM_IN(EX)[z][i] from saved value in NVMY (if adaptation value has never been learned or if the stored value could not be read, PSN_EDGE_AD_CAM_IN(EX)[z][i] are initialized to 0)

PSN_AD_CAM_IN(EX)[i] = Sum **(** PSN_EDGE_AD_CAM_IN(EX)[z][i]

for z = 1…NC_NR_EDGE_CAM_IN(EX) **)** / NC_NR_EDGE_CAM_IN(EX)

(resolution conversion)

LV_CAM_AD_SAVE_IN(EX)[i] from saved value in NVMY (if adaptation value has never been learned or if the stored value could not be read, LV_CAM_AD_SAVE_IN(EX)[i] is initialized to 0)

PSN_EDGE_AD_CAM_IN(EX)[z][i] and LV_CAM_AD_SAVE_IN(EX)[i] shall be saved in NVMY only if: LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] =0 (otherwise the last stored values, which have been stored after last driving cycle without error, shall not be changed)

At reset: CAM_PSN_LST_REF_AD_IN(EX)[i] = PSN_EDGE_AD_CAM_IN(EX)[1][i]

(All these adaptive values has to be reset in case of cam/crk mechanical change (Chain change, camshaft sensor change…))

Formula section:

If |PSN_EDGE_CAM_IN(EX)[1][i] (n) – PSN_EDGE_CAM_IN(EX)[1][i] (n-1)| < C_GRD_AD_REF_MAX_CAM_IN(EX)

Then increment CTR_REV_AD_REF_CAM_IN(EX)[i]

The adaptation values are calculated for all indexes (for z = 1…NC_NR_EDGE_CAM_IN(EX))

PSN_EDGE_AD_CAM_IN(EX)[z][i] (n) = PSN_EDGE_AD_CAM_IN(EX)[z][i] (n-1) + C_CRLC_AD_REF_CAM_IN(EX) * PSN_EDGE_CAM_IN(EX)[z][i]

Else CTR_REV_AD_REF_CAM_IN(EX)[i] = 0

Endif

If CTR_REV_AD_REF_CAM_IN(EX)[i] >= C_NR_REV_AD_REF_CAM_IN(EX)

Then CTR_REV_AD_REF_CAM_IN(EX)[i] = 0

PSN_AD_CAM_IN(EX)[i] = Sum **(** PSN_EDGE_AD_CAM_IN(EX)[z][i]

for z = 1…NC_NR_EDGE_CAM_IN(EX) **)** / NC_NR_EDGE_CAM_IN(EX)

(resolution conversion)

If LV_CAM_AD_SAVE_IN(EX)[i] = 1

Then **If** |CAM_PSN_LST_REF_AD_IN(EX)[i] - PSN_EDGE_AD_CAM_IN(EX)[1][i]| < C_TOL_TOOTH_OFF_CAM_IN(EX)

Then CAM_PSN_LST_REF_AD_IN(EX)[i] = PSN_EDGE_AD_CAM_IN(EX)[1][i]

Else nothing

Endif

LV_TOOTH_OFF_DET_ENA_IN(EX)[i] = 1 *(to allow the ONE_TOOTH_OFF diagnosis only if a reference adaptation has been done after the very first one)*

Else **If** C_TOL_TOOTH_OFF_CAM_IN(EX) <> 0

Then CAM_PSN_LST_REF_AD_IN(EX)[i] = PSN_EDGE_AD_CAM_IN(EX)[1][i]

LV_CAM_AD_SAVE_IN(EX)[i] = 1 *(set at the end of the very first reference adaptation)*

Else nothing

Endif

Endif

LV_AD_END_CAM_IN(EX)[i] = 1

Else

LV_TOOTH_OFF_DET_ENA_IN(EX)[i] = 0

Endif

If LV_DI_AD_REF_CAM_IVVT_IN(EX) = 1

Then CTR_REV_AD_REF_CAM_IN(EX)[i] = 0

Endif

4.87.3.2 Continuous Edge Position Adaptation

The continuous position adaptation is enabled at a limited dynamic variation of the camshaft position measured at the reference signal edge.

The difference of the actual camshaft position measured at the reference edge, and the actual camshaft position measured at any other signal edge, is calculated once per camshaft revolution. That difference is multiplied with a cofficient (<<1) and added to the previous adaptation value to form the new adaptation value, which is applied in the following camshaft revolution. This way the camshaft position output value calculated at any signal edges will approach the camshaft position output value calculated at the reference signal edge.

Application conditions:

Activation:

LV_VLD_PSN_CAM_IN[i] = 1 **And** LV_SYN_VLD = 1

And

LV_DI_AD_REF_CAM_IVVT_IN(EX) = 1 **And** LC_NOT_ADJ_CAM_IVVT_IN(EX)[i] = 0 **And** NLC_IVVT_IN(EX) = 1

Recurrence: *every reference camshaft signal edge (IDX_EDGE_CAM_IN(EX)[i] = 1)*

Initialization:

PSN_EDGE_AD_CAM_IN(EX)[z][i] from saved value in NVMY (if adaptation value has never been learned or if the stored value could not be read, PSN_EDGE_AD_CAM_IN(EX)[z][i] is initialized to 0)

PSN_EDGE_AD_CAM_IN(EX)[z][i] and LV_CAM_AD_SAVE_IN(EX)[i] shall be saved in NVMY only if: LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] =0 (otherwise the last stored values, which have been stored after last driving cycle without error, shall not be changed)

Formula section:

For the following equations, the index z of the camshaft signal edges equals IDX_EDGE_CAM_IN(EX)[i] (z=1 for reference edge). The adaptation values are calculated for all indexes from 2 to NC_NR_EDGE_CAM_IN(EX).

If |PSN_EDGE_CAM_IN(EX)[1][i] (n) - PSN_EDGE_CAM_IN(EX)[1][i] (n-1)| < C_GRD_AD_MAX_CAM_IN(EX)

Then PSN_EDGE_AD_CAM_IN(EX)[z][i] (n) = PSN_EDGE_AD_CAM_IN(EX)[z][i] (n-1) + C_CRLC_AD_CAM_IN(EX) * (PSN_EDGE_CAM_IN(EX)[z][i] - PSN_EDGE_CAM_IN(EX)[1][i])

Endif

4.87.3.3 Reference adaptive values save authorization at Powerlatch

Before saving the adaptive values in NVMY during PWL, it is important to check if they are correct. Even if the REF_CRK_CAM error is not set, the calculated adaptive values can be wrong (for example, the error flag could have been reset on a quick key 0 to 1 transition).

Application conditions:

Activation:

ECU_STATE = PWL

Recurrence: *once during PWL, for all camshaft edges of all present camshaft sensors, before saving the NVMY adaptive values*

Initialization:

LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] =0

Formula section:

If PSN_EDGE_AD_CAM_IN(EX)[z][i] < C_TOL_REF_CRK_CAM_MIN_IN(EX)

or PSN_EDGE_AD_CAM_IN(EX)[z][i] > C_TOL_REF_CRK_CAM_MAX_IN(EX)

Then LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] = 1

Else

If LV_CAM_AD_SAVE_IN(EX)[i] = 1

And

CAM_PSN_LST_REF_AD_IN(EX)[i] <> PSN_EDGE_AD_CAM_IN(EX)[1][i]

And

C_TOL_TOOTH_OFF_CAM_IN(EX) <> 0

Then LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] = 1

Else *nothing, (if LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] has been set to 1 due to anyone of the camshaft edges, it must stay to 1 and the adaptive will not be saved. Contrary, if all the camshaft edges are correct, adaptive values can be saved, LV_CAM_AD_PWL_NOT_SAVE_IN(EX)[i] must stay at 0)*

Endif

Endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
ID_CAM_PHA_COR	16	0...5FFH	-48...47.9375	0.0625	[°CRK]
LDP_N_32_ID_CAM_PHA_COR	16	0...FFH	0...8160	32	[rpm]
Table for camshaft transmitter phase offset correction					
C_GRD_AD_MAX_CAM_IN	1	0...FFH	0...15.9375	0.0625	[°CRK]
Max. permissible drift of reference edge position for continuous adaptation					
C_GRD_AD_MAX_CAM_EX	1	0...FFH	0...15.9375	0.0625	[°CRK]
Max. permissible drift of reference edge position for continuous adaptation					
C_GRD_AD_REF_MAX_CAM_IN	1	0...FFH	0...15.9375	0.0625	[°CRK]
Max. permissible drift of reference edge position for reference position adaptation					
C_GRD_AD_REF_MAX_CAM_EX	1	0...FFH	0...15.9375	0.0625	[°CRK]
Max. permissible drift of reference edge position for reference position adaptation					
C_CRLC_AD_CAM_IN	1	0...FFH	0...0.99609	3.9063e-3	[-]
Averaging constant for continuous adaptation					
C_CRLC_AD_CAM_EX	1	0...FFH	0...0.99609	3.9063e-3	[-]
Averaging constant for continuous adaptation					
C_CRLC_AD_REF_CAM_IN	1	0...FFH	0...0.99609	3.9063e-3	[-]
Averaging constant for reference position adaptation					
C_CRLC_AD_REF_CAM_EX	1	0...FFH	0...0.99609	3.9063e-3	[-]
Averaging constant for reference position adaptation					
C_NR_REV_AD_REF_CAM_IN	1	0...FFH	0...255	1	[-]
Number of camshaft revolutions for reference position adaptation					
C_NR_REV_AD_REF_CAM_EX	1	0...FFH	0...255	1	[-]
Number of camshaft revolutions for reference position adaptation					
C_TOL_TOOTH_OFF_CAM_IN	1	0...1FFH	0...31.9375	0.0625	[°CRK]
Reference position tolerance for tooth off detection for intake camshaft					
C_TOL_TOOTH_OFF_CAM_EX	1	0...1FFH	0...31.9375	0.0625	[°CRK]
Reference position tolerance for tooth off detection for exhaust camshaft					

4.88 Charger environment, forward path

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FAC_TIA_CHA_UP	O/V	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
Temperature correction factor for PRS_CHRG_DOWN_MAX calculation					
MAF_CHA [NC_NR_TCHA]	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Mass air flow through charger					
MAF_CHA_DIF	V	0... FFFFH	-1024... 1023.96875	0.03125	[kg/h]
Mass air flow through charger difference					
PQ_CHA [NC_NR_TCHA]	O/V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio at charger (inlet branch specific)					
PQ_CHA_MV	O/V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio at charger (inlet branch average)					
PRS_CHA_DOWN [NC_NR_TCHA]	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure downstream charger					
PRS_CHA_DOWN_MV	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure down charger (Inlet branch average)					
PRS_CHA_UP [NC_NR_TCHA]	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure up charger					
PRS_CHA_UP_MV	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure up charger (inlet branch average)					
PRS_ICO_DIF	V	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Pressure drop at intercooler					
PUT_SUB	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure upstream throttle substitute in case of PUT sensor failure					
PUT_WG_OPEN	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure upstream throttle when WG opening requested					
VOL_FLOW_CHA_UP [NC_NR_TCHA]	O/V	0... FFFFH	0... 1.99996948242	30.5176e-6	[m3/s]
Volume flow up charger (inlet branch specific)					

Input Data:

TIA_ABSV_CHA_UP	N_32	PRS_LOSS_ICO [NC_NR_TCHA]	TIA_THR
TIA_CHA_DOWN_MV	T_SEG_AV	MAF_RCL [NC_NR_TCHA]	MAF_KGH_AIC_TCHA [NC_NR_TCHA]
MAF_THR_TCHA	AMP	LC_TCHA_CONF	LV_ENA_PUT
PUT_WG_OPEN_AD_ADD	NC_NR_TCHA	MAF_RCL_MV	PRS_LOSS_ICO_MV
MAF_KGH_AIC_MV_TCHA	PRS_CHRG_DOWN	PRS_AIC_DOWN_TCHA [NC_NR_TCHA]	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_MAF_CHA_DIF	1	0... FFFFH	0... 0.99998474	15.2588e-6	[-]
correlation constant for MAF_CHA calculation					
C_CRLC_MAF_KGH_PQ_CHA_PUT_SUB	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]

Correlation constant for substitute PUT calculation					
C_FIL_PRS_ICO_DIF_0	1	0... FFFFH	-1... 0.99996948	30.5176e-6	[-]
Filter constant 0 for PRS_ICO_DIF calculation					
C_FIL_PRS_ICO_DIF_1	1	0... FFFFH	-1... 0.99996948	30.5176e-6	[-]
Filter constant 1 for PRS_ICO_DIF calculation					
C_FIL_PRS_ICO_DIF_2	1	0... FFFFH	-1... 0.99996948	30.5176e-6	[-]
Filter constant 2 for PRS_ICO_DIF calculation					
C_PRS_CHA_DOWN_AS	1	0... FFFFH	0... 5434	0.0829175	[hPa]
Constant for manual pressure down charger					
C_PRS_CHA_UP_AS	1	0... FFFFH	0... 5434	0.0829175	[hPa]
Constant for use of manual pressure up charger					
C_RA_VOL_ICO	1	0... 764AH	0... 140307.6445	4.6333678	[Pa/(kg*K)]
Specific gas constant of air divided by intercooler volume					
IP_FAC_TIA_CHA_UP	7	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_TIA_ABSV_CHA_UP	7	0... FFFFH	0... 832.015666	0.0126957	[K]
Temperature correction factor for temperature upstream charger					
IP_PQ_CHA_PUT_SUB	6*6	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDP_N_32_IP_PQ_CHA_PUT_SUB	6	0... FFH	0... 8160	32	[rpm]
LDP_MAF_KGH_IP_PQ_CHA_PUT_SUB	6	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Pressure ratio at charger for substitute PUT calculation					
IP_PRS_CHA_DOWN_WG_OPEN	6*8	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_N_32_IP_PRS_CHA_DOWN	6	0... 1FE0H	0... 8160	1	[rpm]
LDP_PRS_CHA_UP_IP_PRS_CHA_DOWN	8	0... FFFFH	0... 5434	0.0829175	[hPa]
Model of PRS_CHA_DOWN when WG opening requested					
LC_AMP_PRS_CHA_DOWN_WG_OPEN	1	0... 1H	0... 1	1	[-]
Use AMP as Input of IP_PRS_CHA_DOWN_WG_OPEN					
LC_FAC_TIA_PRS_CHA_DOWN_WG_OPEN	1	0... 1H	0... 1	1	[-]
Switch whether PRS_CHA_DOWN_WG_OPEN is temperature corrected					
LC_MAF_CHA_CLC_PUT	1	0... 1H	0... 1	1	[-]
Switch whether MAF_CHA is calculated from PUT and MAF_THR (= 1) or from MAF_AIC (= 0)					
LC_PRS_CHA_DOWN_AS	1	0... 1H	0... 1	1	[-]
Switch for use of manual pressure down charger					
LC_PRS_CHA_UP_AS	1	0... 1H	0... 1	1	[-]
Switch for use of manual pressure up charger					

General Information

This module calculates the operating point for the charger, which is used to calculate the charger model.

Application Conditions

Initialization: RST, ERU2ES
Recurrence: SEG10
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

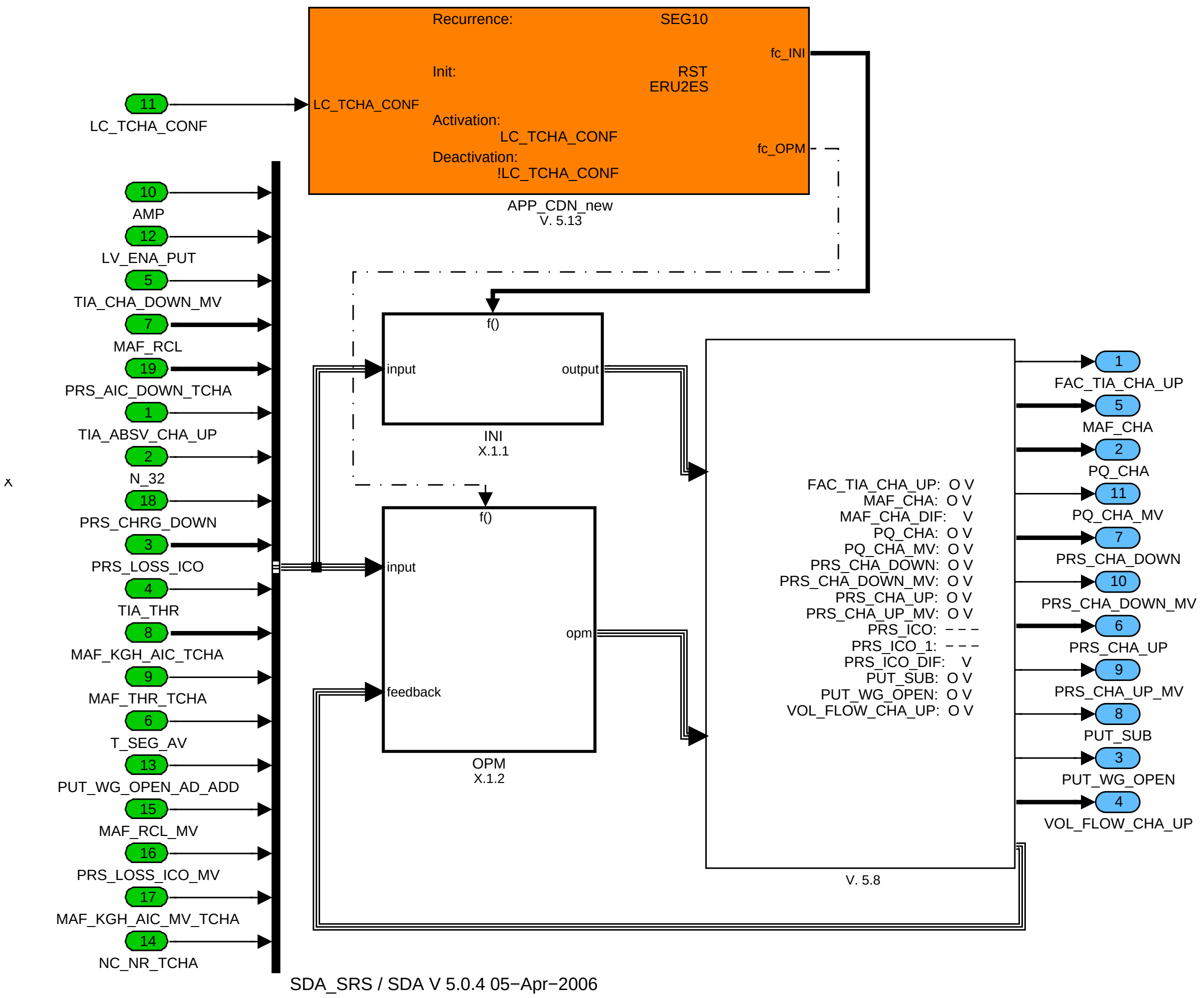


Figure 418:
Path: CHRG_MDLADCHAEN0

4.88.1 CHRG_MDLADCHAEN0/INI

CONTENT

4.88.1.1 CHRG_MDLADCHAEN0/INI/CLC_INI

CONTENT

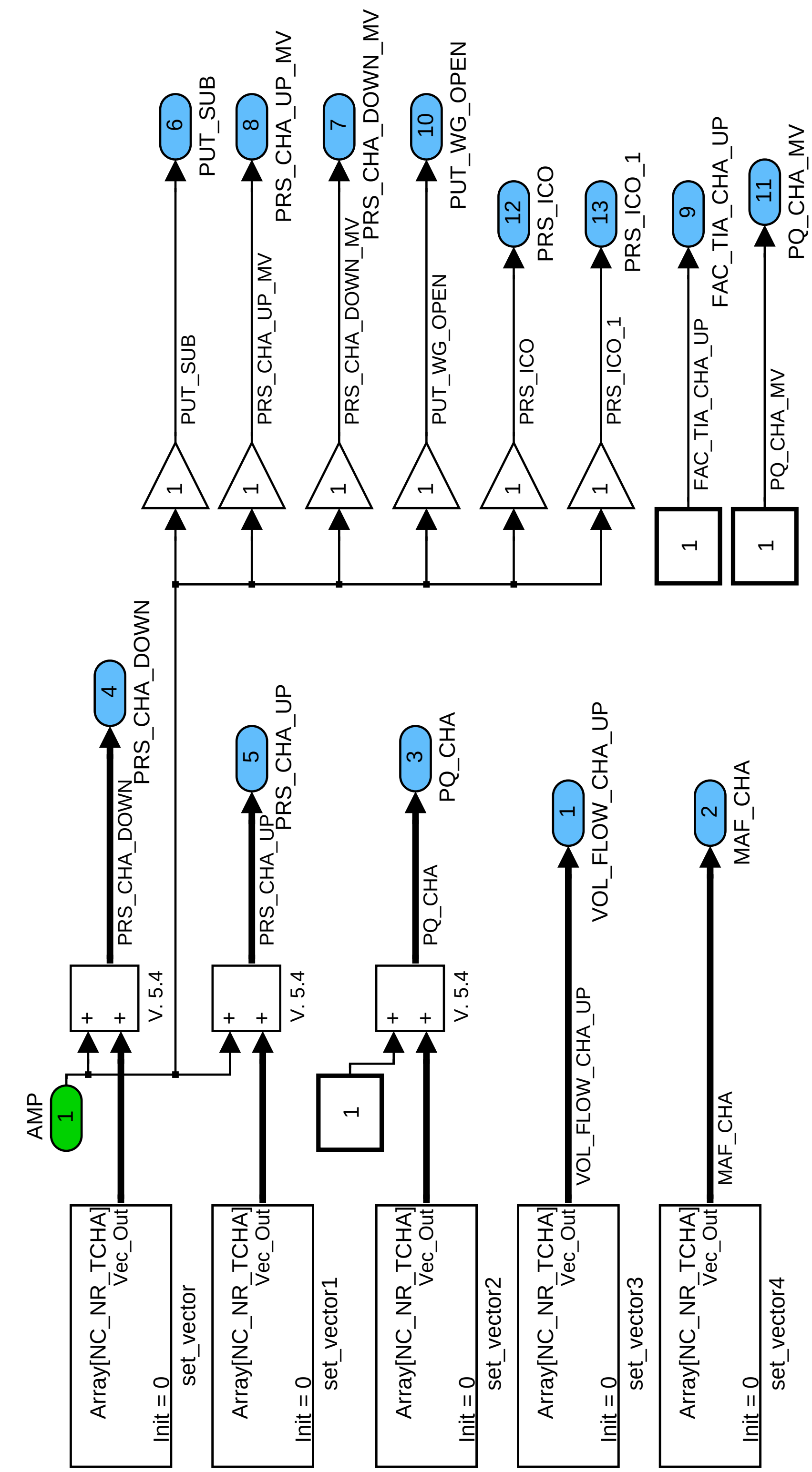


Figure 419:
Path: CHRG_MDLADCHAEN0/INI/CLC_INI

4.88.2 CHRG_MDLADCHAEN0/OPM

CONTENT

4.88.2.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM

CONTENT

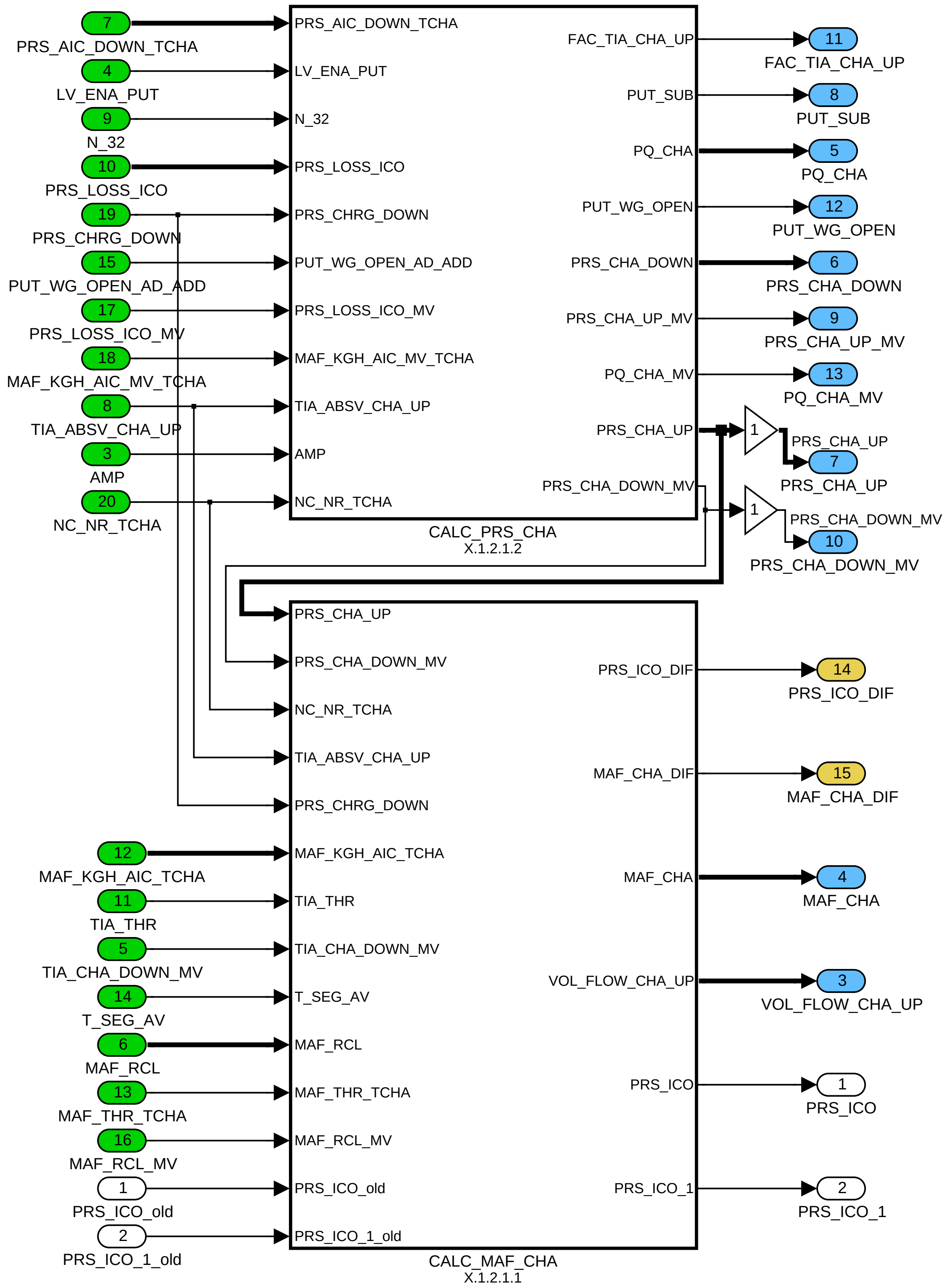


Figure 420:
Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM

4.88.2.1.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA

CONTENT

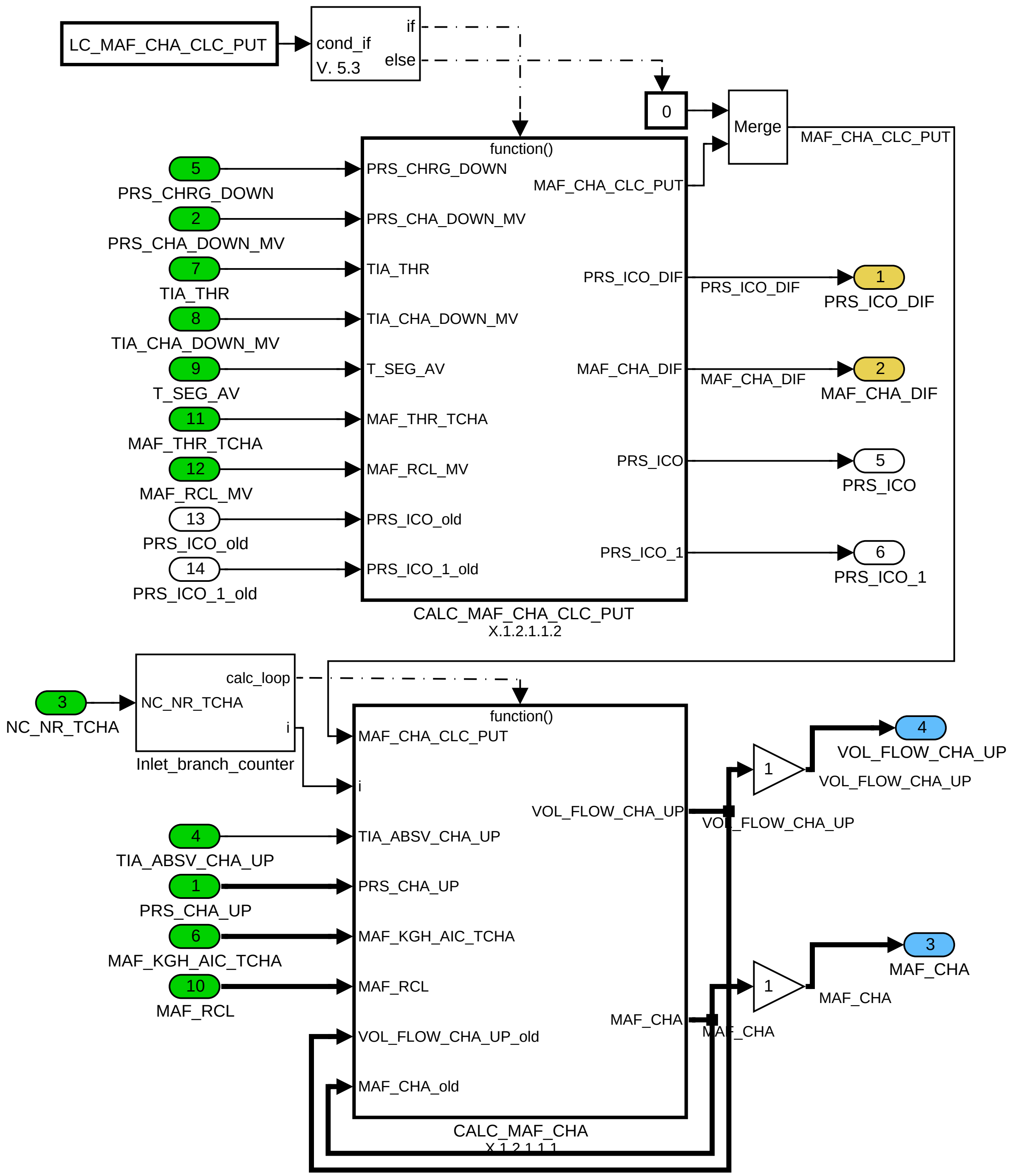


Figure 421:

Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA

4.88.2.1.1.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA/CALC_MAF_CHA

CONTENT

4.88.2.1.1.1.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA/CALC_MAF_CHA/CALC_MAF_CHA

CONTENT

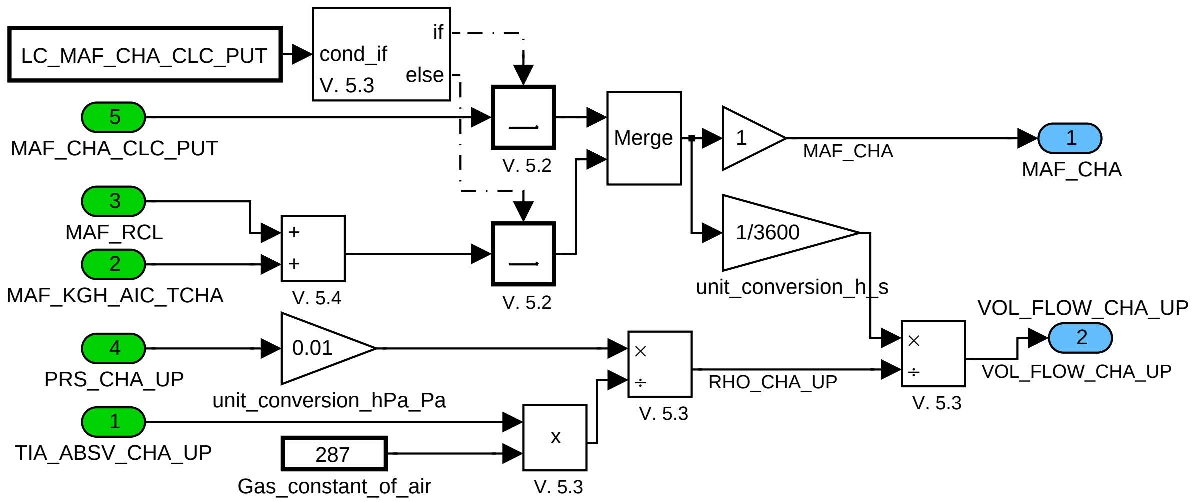


Figure 422:
Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA/CALC_MAF_CHA/CALC_MAF_CHA

4.88.2.1.1.2 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA/CALC_MAF_CHA_CLC_PUT

CONTENT

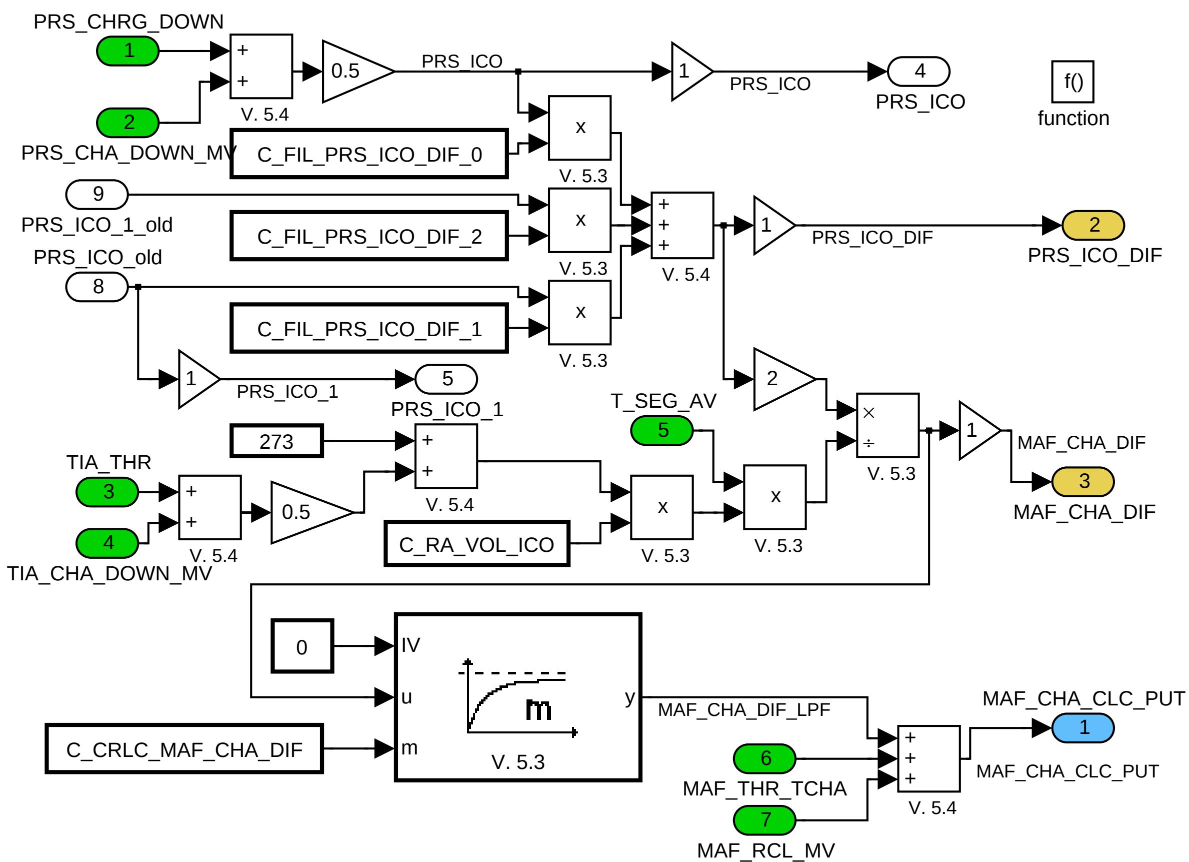


Figure 423:
Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_MAF_CHA/CALC_MAF_CHA_CLC_PUT

4.88.2.1.2 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA

CONTENT

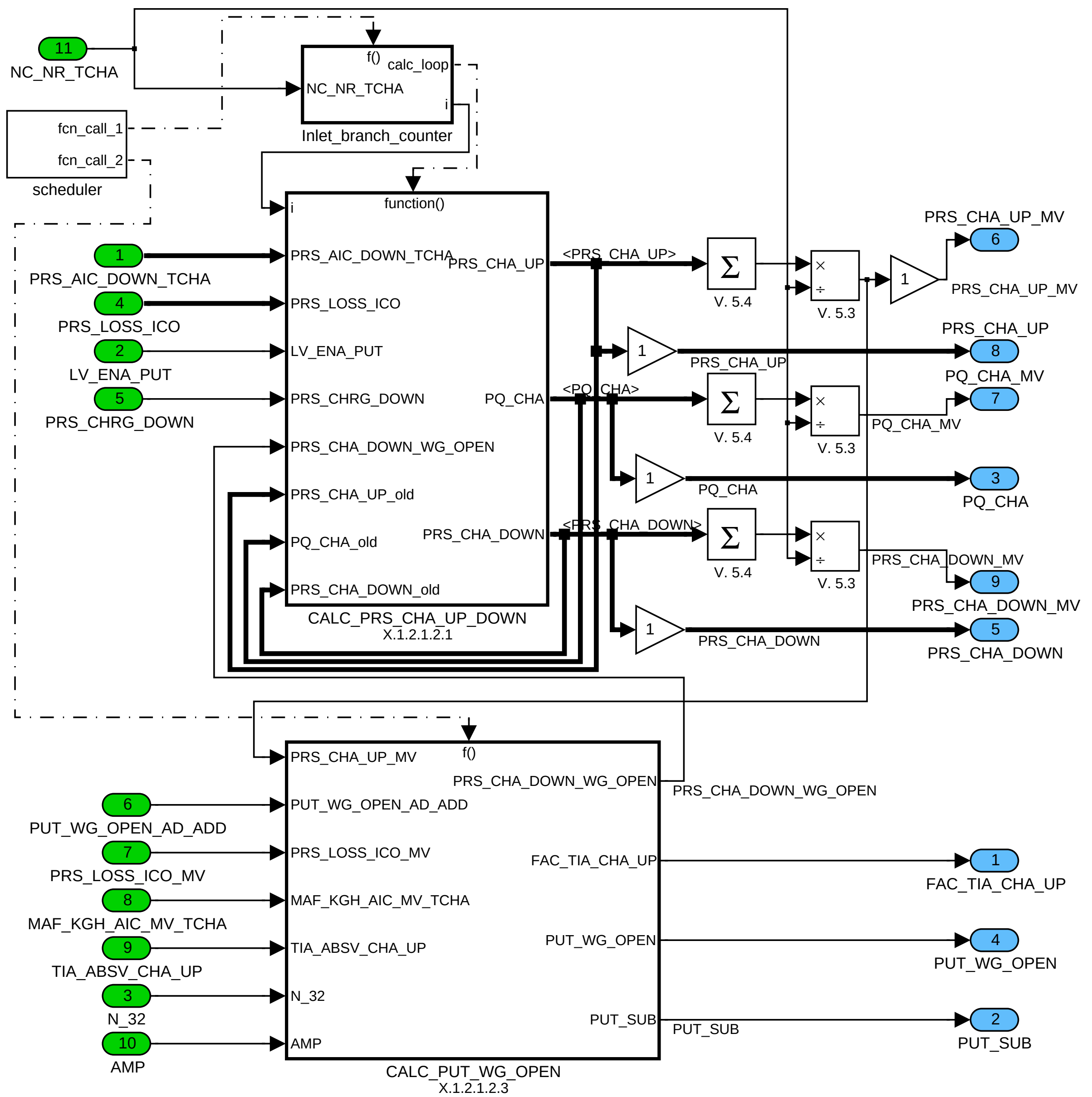


Figure 424:

Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA

4.88.2.1.2.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA/CALC_PRS_CHA_UP_DOWN

CONTENT

4.88.2.1.2.1.1 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA/CALC_PRS_CHA_UP_DOWN/CALC_PRS_CHA

CONTENT

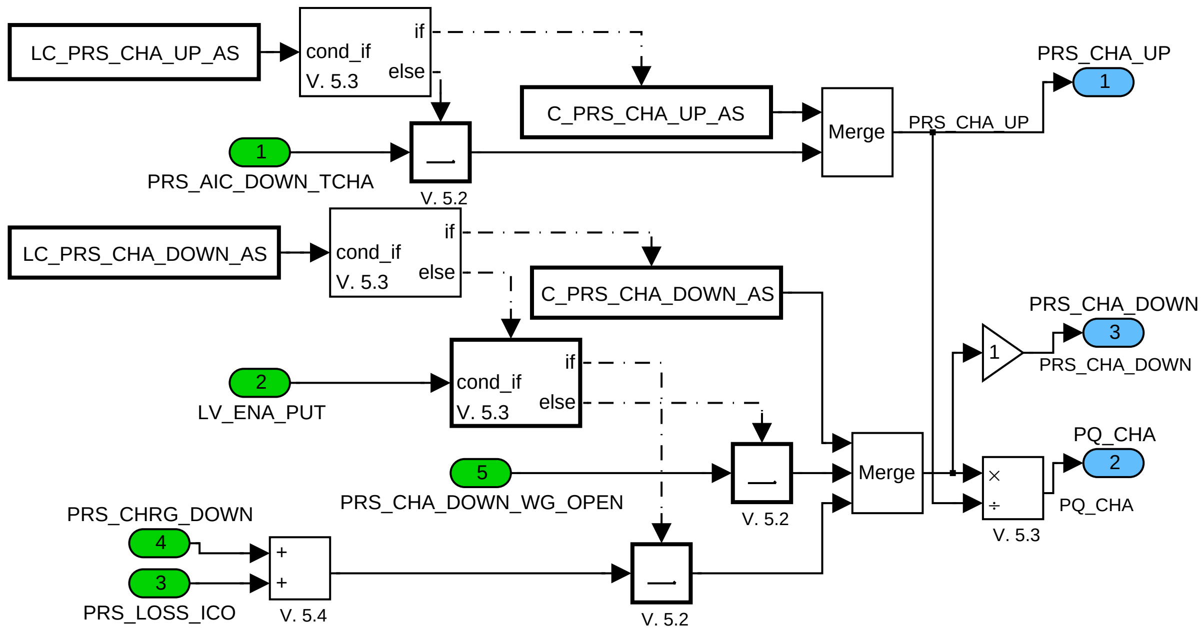


Figure 425:
Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA/CALC_PRS_CHA_UP_DOWN/CALC_PRS_CHA

4.88.2.1.2.2 CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA/CALC_PUT_WG_OPEN

CONTENT

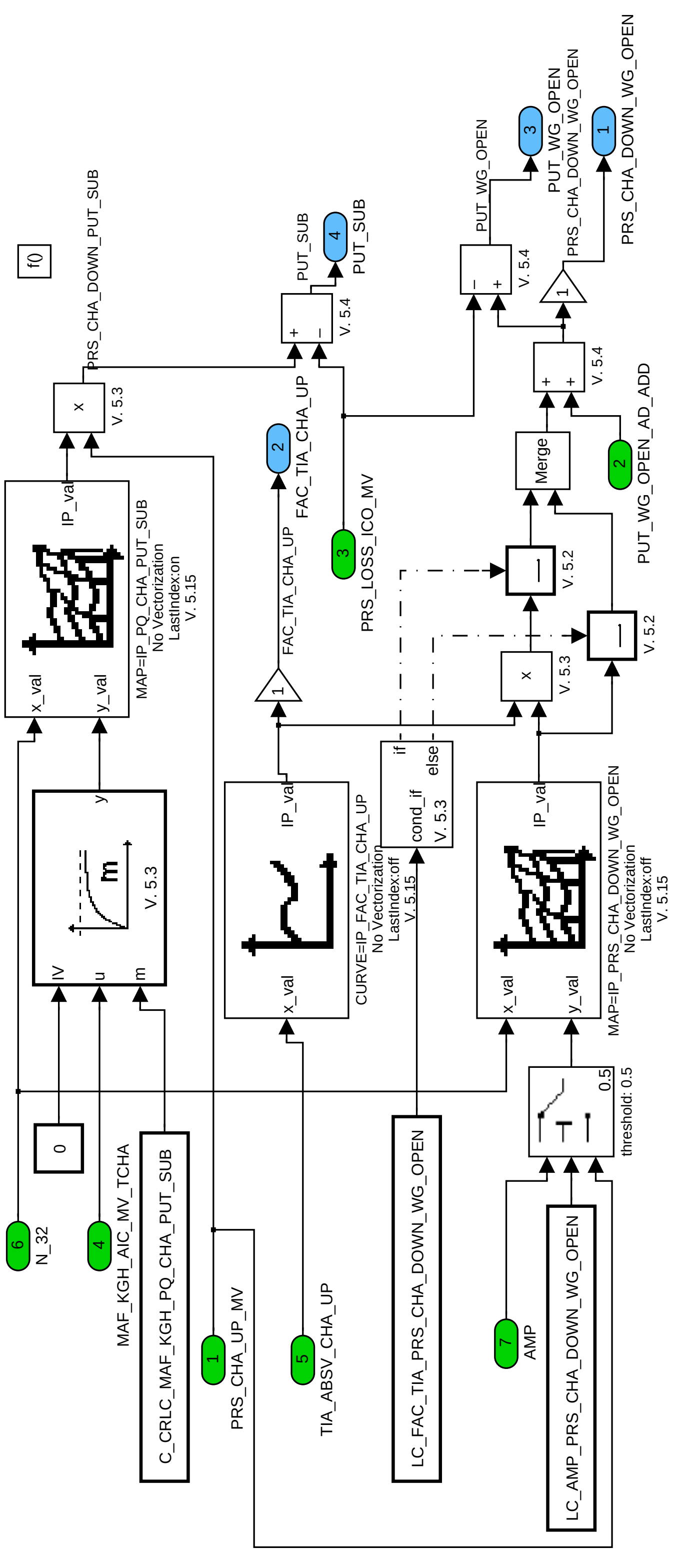


Figure 426:
Path: CHRG_MDLADCHAEN0/OPM/CLC_OPM/CALC_PRS_CHA/CALC_PUT_WG_OPEN

4.89 Basic charge air pressure adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DLY_PUT_WG_OPEN_AD	V	0...FFH	0...2.55	0.01	[s]
Delay time for basic charge air pressure adaptation					
PUT_WG_OPEN_AD_ADD	O/V	8000...7FFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Basic charge air pressure adaptation value					
IDX_N_PUT_WG_OPEN_AD	V	0...FFH	0...255	1	[-]
Engine speed index for basic charge air pressure adaptation					
PUT_WG_OPEN_AD[6]	O/V/S	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Map for adapted values for basic charge air pressure adaptation					
PUT_WG_OPEN_AD_AS[6]	V/S	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
For copy (for application purposes) of map for adapted values for basic charge air pressure adaptation					
LV_CDN_PUT_WG_OPEN_AD	V	0...1H	0...1	1	[-]
Conditions for basic charge air pressure fulfilled					
CTR_PUT_WG_OPEN_AD	V	0...FFFFH	0...65535	1	[-]
Basic charge air pressure adaptation counter					

Input data:

PWM_WG	LV_ST_END	MAP	PUT_MMV
PUT_WG_OPEN	N	LC_TCHA_CONF	LV_ENA_PUT

FUNCTION DESCRIPTION:

General information:

Basic charge air pressure PUT_WG_OPEN is defined as the stationary pressure upstream throttle if the throttle is completely open and the wastegate is controlled to be open. This basic charge air pressure varies between individual engines. These variations are learnt in the present module and stored in dependence of engine speed.

Application conditions:

Initialisation: at ECU reset

T_DLY_PUT_WG_OPEN_AD = C_T_DLY_PUT_WG_OPEN_AD_INI

LV_CDN_PUT_WG_OPEN_AD = 0

if (checksum error) **or** LC_AD_CLR_PUT_WG_OPEN **then**

PUT_WG_OPEN_AD [...] = 0

else

read PUT_WG_OPEN_AD from non volatile memory

endif

Recurrence: 10 ms

Activation: LC_TCHA_CONF = 1 **and** LV_ST_END = 1

Deactivation: LC_TCHA_CONF = 0 **or** LV_ST_END = 0

Formula section:

PUT_WG_OPEN_AD_ADD = PUT_WG_OPEN_AD (N)

LV_CDN_PUT_WG_OPEN_AD = 0

(1) if PWM_WG > C_PWM_BOL_PUT_WG_OPEN_AD **and**
PWM_WG < C_PWM_TOL_PUT_WG_OPEN_AD **and**
MAP / PUT_MMV > C_PQ_THD_PUT_WG_OPEN_AD **and** LV_ENA_PUT **then**
$T\_DLY\_PUT\_WG\_OPEN\_AD_{N} = T\_DLY\_PUT\_WG\_OPEN\_AD_{N-1}$ – 10 ms

(2) if T_DLY_PUT_WG_OPEN_AD == 0 **then** // delay time has elapsed

// search for current engine speed index of adaptation map

IDX_N_PUT_WG_OPEN_AD = FFh

// idx_sw is calculated by the software data point search algorithm for the axis
// LDPM_N_2_CHRG at N. The algorithm delivers indices starting from 0

(3) for idx_sw = 0 **to** (length (LDPM_N_2_CHRG)-1) **do**

(4) if (LDPM_N_2_CHRG [idx_sw] <= N **and**
N < LDPM_N_2_CHRG [idx_sw+1]) **then**

break

(4) endif

(3) endfor

// test whether current N is within one of the two possible adaptation windows

(5) if (ID_N_PUT_WG_OPEN_AD_MIN [idx_sw] <= N **and**
N < ID_N_PUT_WG_OPEN_AD_MAX [idx_sw]) **then**

IDX_N_PUT_WG_OPEN_AD = idx_sw

(5) endif

(6) if (idx_sw < (length (LDPM_N_2_CHRG)-1) **then**

(7) if (ID_N_PUT_WG_OPEN_AD_MN [idx_sw+1] <= N **and**
N <= ID_N_PUT_WG_OPEN_AD_MAX [idx_sw+1]) **then**

IDX_N_PUT_WG_OPEN_AD = idx_sw+1

(7) endif

(6) endif

PUT_WG_OPEN_DIF = PUT_MMV – PUT_WG_OPEN

(8) if IDX_N_PUT_WG_OPEN_AD < FFh **and**
abs (PUT_WG_OPEN_DIF) > C_PUT_WG_OPEN_AD_INC **then**
// a fitting window for adaptation has been found and adaptation input is greater
// than one increment of the adaptation map PRS_DIF_WG_ACR_SP_AD

(9) if PUT_WG_OPEN_DIF > 0 **then**

PUT_WG_OPEN_AD_TMP = PUT_WG_OPEN_AD

```
        (IDX_N_PUT_WG_OPEN_AD) + C_PUT_WG_OPEN_AD_INC
      (10) if PUT_WG_OPEN_AD_TMP < C_PUT_WG_OPEN_AD_TOL then
        // top limit of adaptation range not reached
        LV_CDN_PUT_WG_OPEN_AD = 1
      (10) endif
    (9) else
      PUT_WG_OPEN_AD_TMP = PUT_WG_OPEN_AD
        (IDX_N_PUT_WG_OPEN_AD) - C_PUT_WG_OPEN_AD_INC
      (11) if PUT_WG_OPEN_AD_TMP > C_PUT_WG_OPEN_AD_BOL then
        // bottom limit of adaptation range not reached
        LV_CDN_PUT_WG_OPEN_AD = 1
      (11) endif
    (9) endif
    (12) if LV_CDN_PUT_WG_OPEN_AD then
      CTR_PUT_WG_OPEN_AD  = CTR_PUT_WG_OPEN_AD + 1
      PUT_WG_OPEN_AD (IDX_N_PUT_WG_OPEN_AD) =
        PUT_WG_OPEN_AD_TMP         // modify adaptation map
    (12) endif
  (8) endif
  T_DLY_PUT_WG_OPEN_AD = C_T_DLY_PUT_WG_OPEN_AD_INI
 (2) endif
(1) else
 T_DLY_PUT_WG_OPEN_AD = C_T_DLY_PUT_WG_OPEN_AD_INI
(1) endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PWM_BOL_PUT_WG_OPEN_AD	1	0...FFFFH	0...99.99847	1.5259e-3	[%]
Bottom wastegate PWM limit for pressure upstream throttle at open wastegate adaptation					
C_PWM_TOL_PUT_WG_OPEN_AD	1	0...FFFFH	0...99.99847	1.5259e-3	[%]
Top wastegate PWM limit for pressure upstream throttle at open wastegate adaptation					
C_PQ_THD_PUT_WG_OPEN_AD	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Throttle pressure ratio threshold for pressure upstream throttle at open wastegate adaptation					
ID_N_PUT_WG_OPEN_AD_MAX	6	0...1FE0H	0...8160	1	[rpm]
LDPM_N_2_CHRG	6	0...1FE0H	0...8160	1	[rpm]
Maximum engine speed threshold for basic charge pressure adaptation					
ID_N_PUT_WG_OPEN_AD_MIN	6	0...1FE0H	0...8160	1	[rpm]
LDPM_N_2_CHRG	6	0...1FE0H	0...8160	1	[rpm]
Minimum engine speed threshold for basic charge pressure adaptation					
C_T_DLY_PUT_WG_OPEN_AD_INI	1	0...FFH	0...2.55	0.01	[s]
Delay time for basic charge air pressure adaption					
C_PUT_WG_OPEN_AD_INC	1	0...FFFFH	0...5434	0.0829175	[hPa]
Increment of PUTcontroller adaption map entries					
LC_AD_CLR_PUT_WG_OPEN	1	0...1H	0...1	1	[-]
Switch whether to clear basic charge pressure adaptation values after ECU reset					
C_PUT_WG_OPEN_AD_TOL	1	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Top limit of PUT controller adaptation map entries					
C_PUT_WG_OPEN_AD_BOL	1	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Bottom limit of PUT controller adaptation map entries					

4.90 Charger model for turbocharger

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VOL_FLOW_CHA_UP_RED[NC_NR_TCHA]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
Reduced volume flow up charger (inlet branch specific)					
N_CHA_CLC[NC_NR_TCHA]	O/V	0...FFFFH	0...4E+5	6.10360876	rpm
Calculated turbo charger speed (inlet branch specific)					
POW_CHA[NC_NR_TCHA]	O/V	0...FFFFH	0...65.535	0.001	kW
charger power (inlet branch specific)					
VOL_FLOW_CHA_UP_RED_MV	O/V	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
Reduced volume flow up charger (inlet branch average)					
POW_CHA_MV	O/V	0...FFFFH	0...65.535	0.001	kW
Charger power (inlet branch average)					
N_CHA_SP	O/V	0...FFFFH	0...4E+5	6.10360876	rpm
Turbo charger speed setpoint					
VOL_FLOW_CHA_UP_RED_SP	O/V	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
Reduced volume flow up charger					
POW_CHA_SP	O/V	0...FFFFH	0...65.535	0.001	kW
Charger power setpoint					
FAC_POW_CHA_SP_1	V	0...FFFFH	0...4.43036E+3	0.06760294	J/kg*K
Factor 1 for calculation of charger power					
FAC_POW_CHA_1[NC_NR_TCHA]	V	0...FFFFH	0...4.43036E+3	0.06760294	J/kg*K
Factor 1 for calculation of charger power (inlet branch specific)					
FAC_POW_CHA_2[NC_NR_TCHA]	V	0...FFFFH	0...1.99996948	3.05176E-5	-
Factor 2 for calculation of charger power (inlet branch specific)					
FAC_POW_CHA_SP_2	V	0...FFFFH	0...1.99996948	3.05176E-5	-
Factor 2 for calculation of charger power					
ANG_CHA_SP	V	0...FFH	0...89.25	0.35	°
Radius angle of charger map setpoint					
DIST_CHA_SP	V	0...FFFFH	0...1.41419198	2.15792E-5	-
Radius of charger map setpoint					
DIST_CHA_STND_SP	V	0...FFFFH	0...1.41419198	2.15792E-5	-
standardized radius of charger map					
ANG_CHA[NC_NR_TCHA]	V	0...FFH	0...89.25	0.35	°
Radius angle of charger map (inlet branch specific)					
DIST_CHA[NC_NR_TCHA]	V	0...FFFFH	0...1.41419198	2.15792E-5	-
Radius of charger map					
DIST_CHA_STND[NC_NR_TCHA]	V	0...FFFFH	0...1.41419198	2.15792E-5	-
standardized radius of charger map					
PQ_CHA_STND	-	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure ratio at charger standardized)					

Input data:

VOL_FLOW_CHA_UP[NC_NR_TCHA]	FAC_TIA_CHA_UP	PQ_CHA[NC_NR_TCHA]	MAF_CHA[NC_NR_TCHA]
TIA_ABSV_CHA_UP	LC_TCHA_CONF	NC_NR_TCHA	VOL_FLOW_CHA_UP_SP
PQ_CHA_SP	MAF_CHA_SP		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_STND_PQ_CHA	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Factor for pressure quotient at charger standardization					
C_FAC_STND_VOL_FLOW	1	0...FFFFH	0...15.9997559	2.44141E-4	m^3
Factor for volumetric flow at charger standardization					
C_N_CHA_COR_REF_VOL_FLOW	1	0...FFFFH	0...4E+5	6.10360876	rpm
corrected charger speed reference for volumetric flow standardization					
C_PQ_CHA_OFS	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Reference point offset for pressure quotient at charge					
C_VOL_FLOW_CHA_UP_RED_OFS	1	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
Reference point offset for volume flow at charger					
IP_ANG_CHA	16	0...FFH	0...89.25	0.35	°
LDP_FAC_SLOP_CHA_IP_ANG_CHA	16	0...FFFFH	0...63.9990234	9.76563E-4	-
Angle of radia					
IP_DIST_CHA	12	0...FFFFH	0...1.41419198	2.15792E-5	-
LDP_DIST_SQR_CHA_DIST_CHA	12	0...FFFFH	0...1.99996948	3.05176E-5	-
Standardized factor for square radius, charger map					
IP_DIST_CHA_OFS	6	0...FFFFH	0...1.41419198	2.15792E-5	-
LDPM_ANG_CHA_1	6	0...FFH	0...89.25	0.35	°
Radius offset					
IP_FAC_POW_CHA_2	16	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_PQ_CHA_POW_CHA	16	0...FFFFH	0...15.9997559	2.44141E-4	-
Factor 2 for calculation of charger power					
IP_FAC_STND_DIST_CHA	6	0...FFFFH	0...1.99996948	3.05176E-5	-
LDPM_ANG_CHA_1	6	0...FFH	0...89.25	0.35	°
factor for radius scaling					
IP_FAC_POW_CHA_1	9x16	0...FFFFH	0...4.43036E+3	0.06760294	J/kg*K
LDP_N_CHA_COR_POW_CHA	9	0...FFFFH	0...4E+5	6.10360876	rpm
LDP_VOL_FLOW_CHA_UP_RED_POW	16	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
Factor 1 for calculation of charger power					
IP_N_CHA_CLC	6x16	0...FFFFH	0...4E+5	6.10360876	rpm
LDPM_ANG_CHA_1	6	0...FFH	0...89.25	0.35	°
LDP_DIST_CHA_STND_IP_N_CHA_CLC	16	0...FFFFH	0...1.41419198	2.15792E-5	-
Calculated turbo charger speed					

4.90.1 CHRG_MDLADCHA0

General information:

This module calculates the turbocharger rotational speed with a special interpolation algorithm for the charger map.

Actual charger speed for NC_NR_TCHA chargers and charger speed setpoint are calculated using the same algorithm.

The square root calculation is based on a shifting algorithm implemented in order to improve calculation accuracy.

Function Description

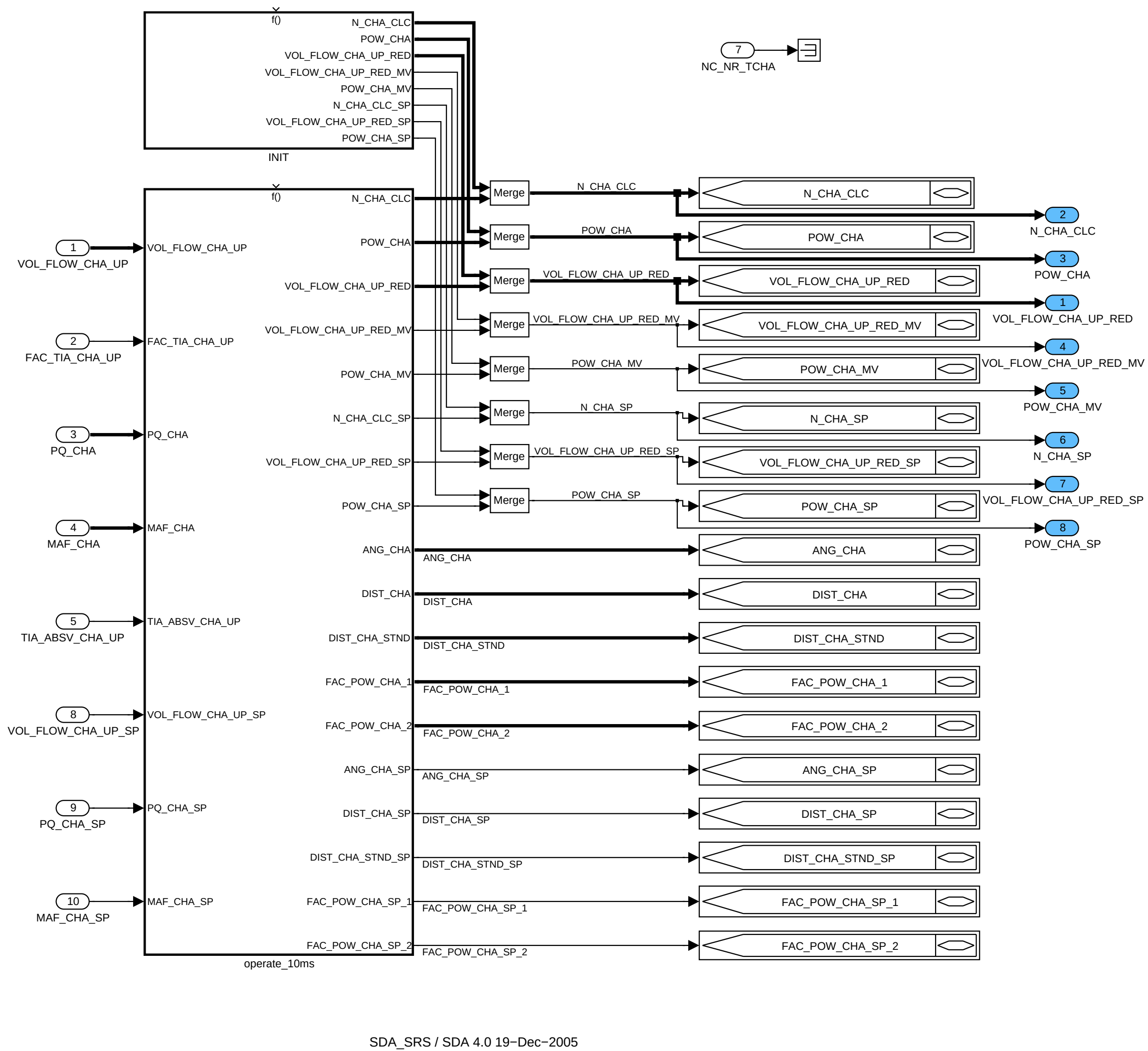


Figure 427 CHRG_MDLADCHA0

4.90.1.1 SUBFUNCTION: INIT

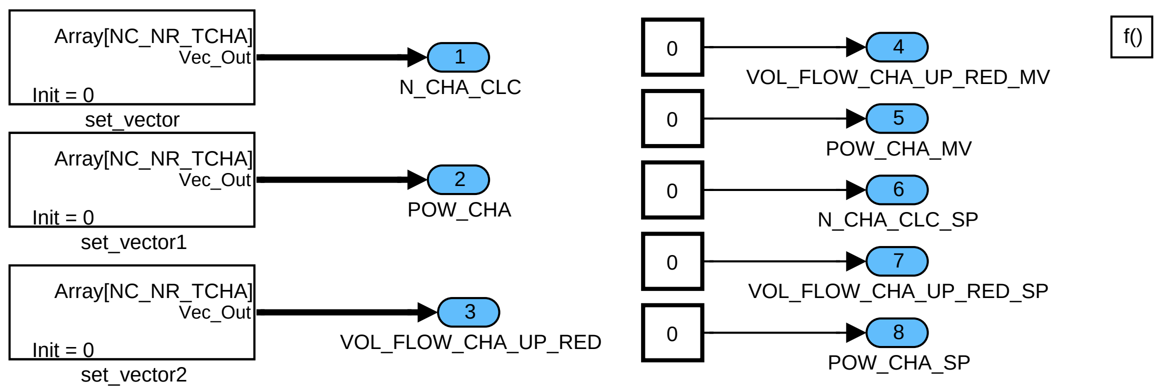


Figure 428 CHRG_MDLADCHA0/ INIT

4.90.1.2 SUBFUNCTION: operate_10ms

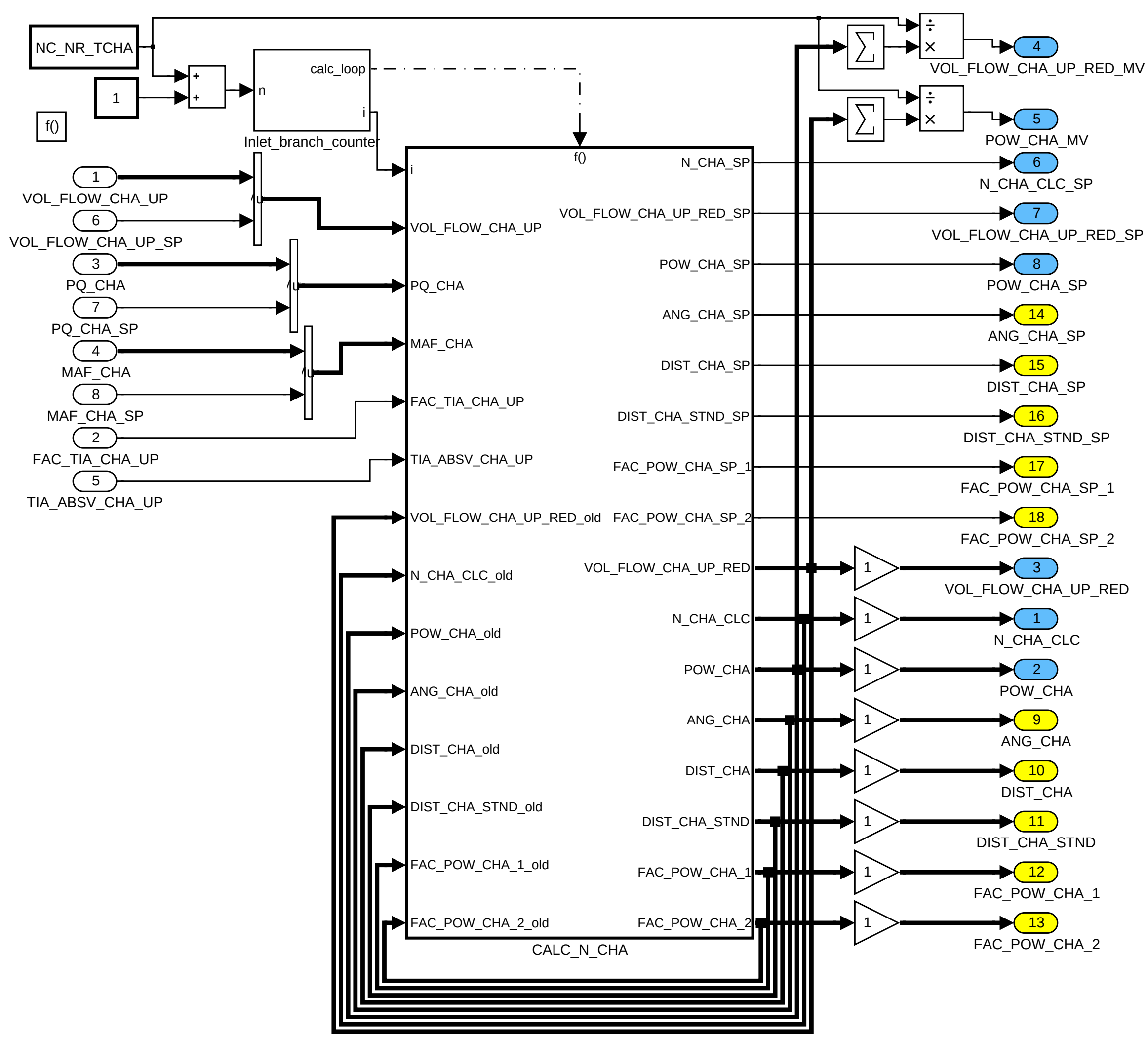


Figure 429 CHRG_MDLADCHA0/ operate_10ms

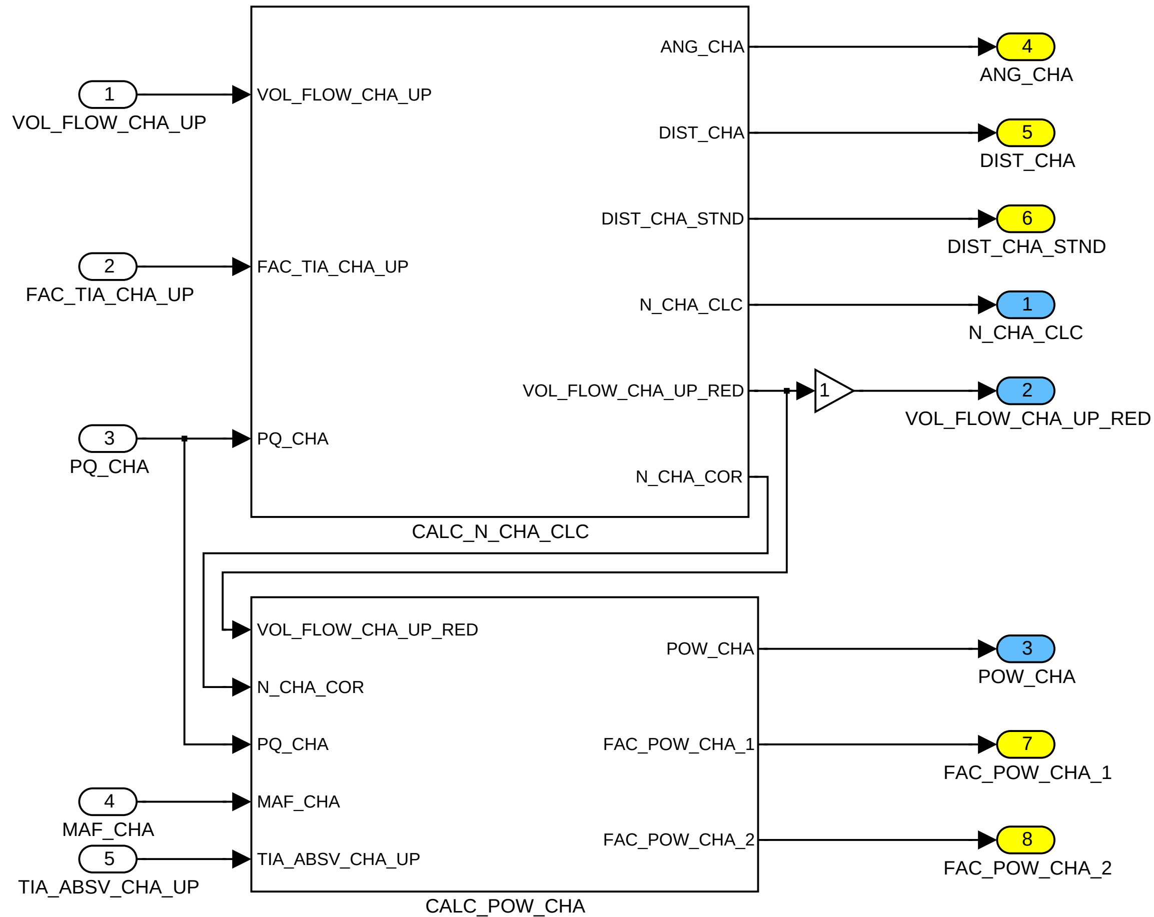


Figure 430 CHRG_MDLADCHA0/ operate_10ms/ CALC_N_CHA/ CALC_N_CHA

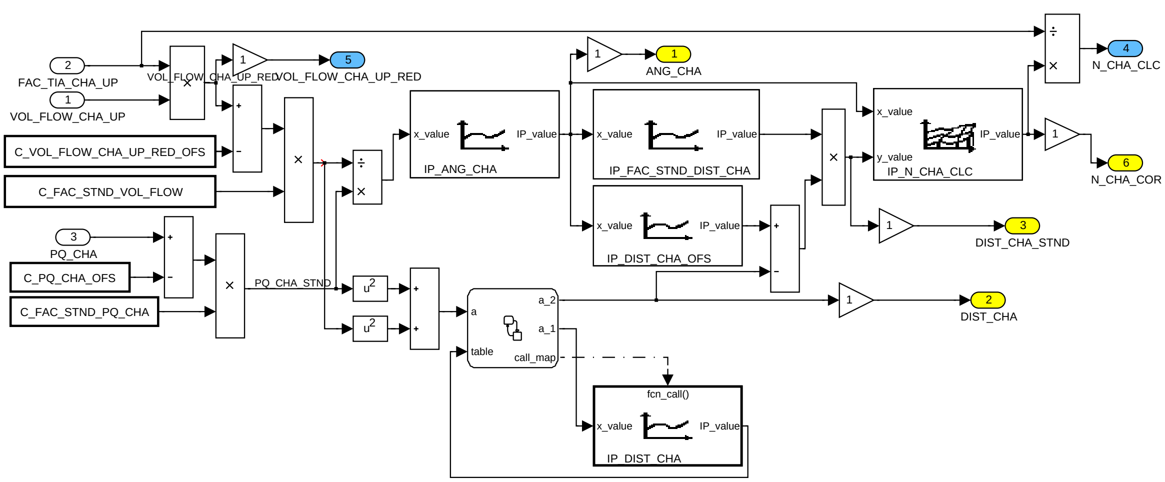


Figure 431 CHRG_MDLADCHA0/ operate_10ms/ CALC_N_CHA/ CALC_N_CHA/ CALC_N_CHA_CLC

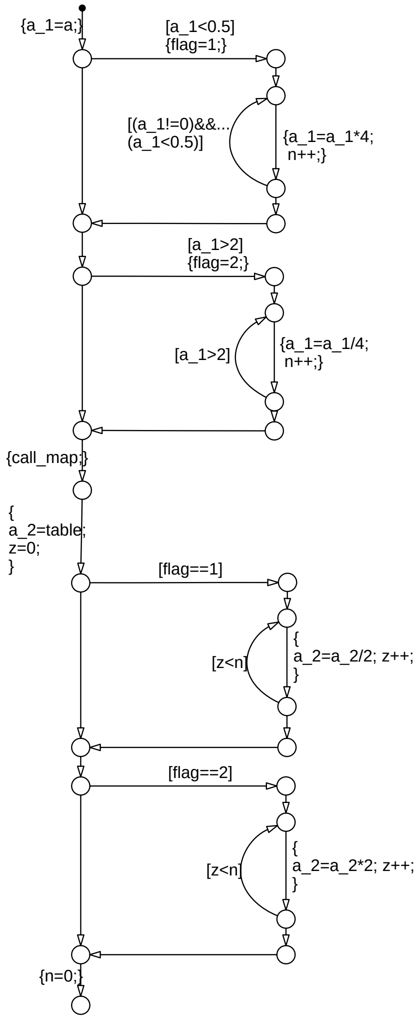


Figure 432 CHRG_MDLADCHA0/ operate_10ms/ CALC_N_CHA/ CALC_N_CHA/ CALC_N_CHA_CLC/ Chart

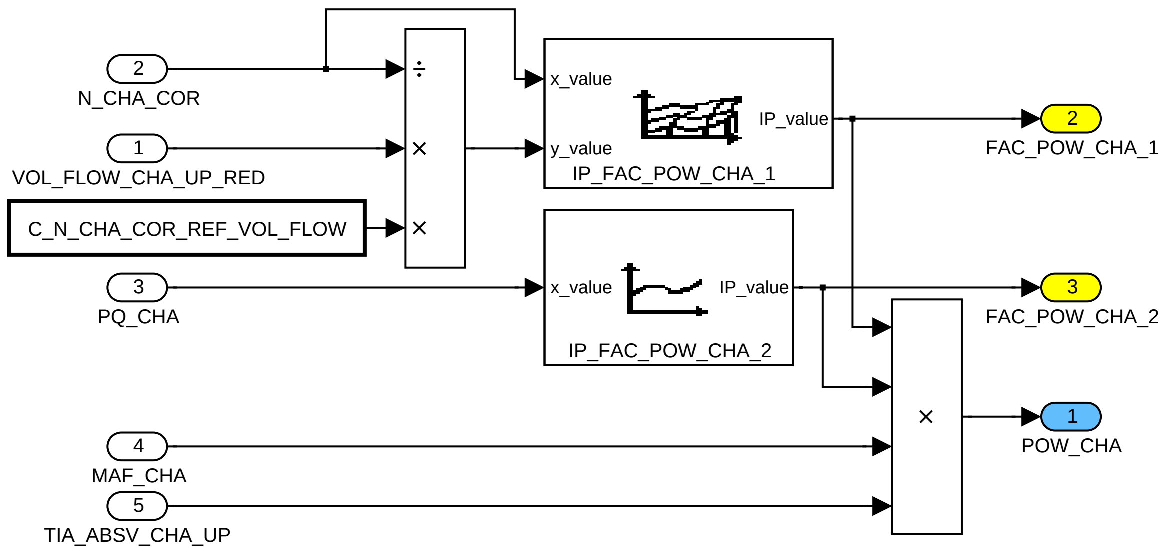


Figure 433 CHRG_MDLADCHA0/ operate_10ms/ CALC_N_CHA/ CALC_N_CHA/ CALC_POW_CHA

4.91 Exhaust gas pressure

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FAC_PRS_EX_N_TUR	V	0... FFH	0... 0.99609375	3.90625e-3	[-]
Factor for determination of the confidence range for the turbine speed for the exhaust pressure calculation					
PQ_EX	O/V	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
pressure quotient ambient / exhaust					
PRS_EX	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure in the exhaust system					
PRS_EX_PCAT_UP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Exhaust pressure pre catalyst					

Input Data:

AMP	N_TUR_COR	PRS_EX_TCHA	LC_TCHA_CONF
NC_CHRG_CONF	FLOW_CAT_MV	FLOW_ENG_MV	N_TCHA

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_PRS_EX_PCAT_UP	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Correlation constant for exhaust gas pressure pre catalyst					
IP_N_TUR_INTER	5	0... FFH	0... 0.99609375	3.90625e-3	[-]
LDP_N_TUR_IP_N_TUR_INTER	5	0... FFFFH	0... 400000	6.1036088	[rpm]
Setting confidence turbine speed for pressure ex calculation					
IP_PQ_EX_ESTIM	6*16	0... FFFFH	0... 3.99993896484	61.0352e-6	[-]
LDP_FLOW_ENG_MV_IP_PQ_EX_ESTIM	16	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
LDP_AMP_IP_PQ_EX_ESTIM	6	0... FFFFH	0... 5434	0.0829175	[hPa]
Map for the estimation of the pressure quotient turbine in case of low turbine speed or for a larger range					
IP_PRS_EX_PCAT_UP_INC	16	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_FLOW_CAT_MV_IP_PRS_EX_UP	16	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Pressure increase due to the catalyst					
LC_USE_N_TCHA_PRS_EX	1	0... 1H	0... 1	1	[-]
Switch for using N_TCHA for exhaust pressure calculation					

General Information

The aim of this module is the calculation of the exhaust gas pressure and the pressure increase due to catalyst.

Application Conditions

Initialization: RST
Recurrence: SEG
Activation: always
Deactivation: never

Function description

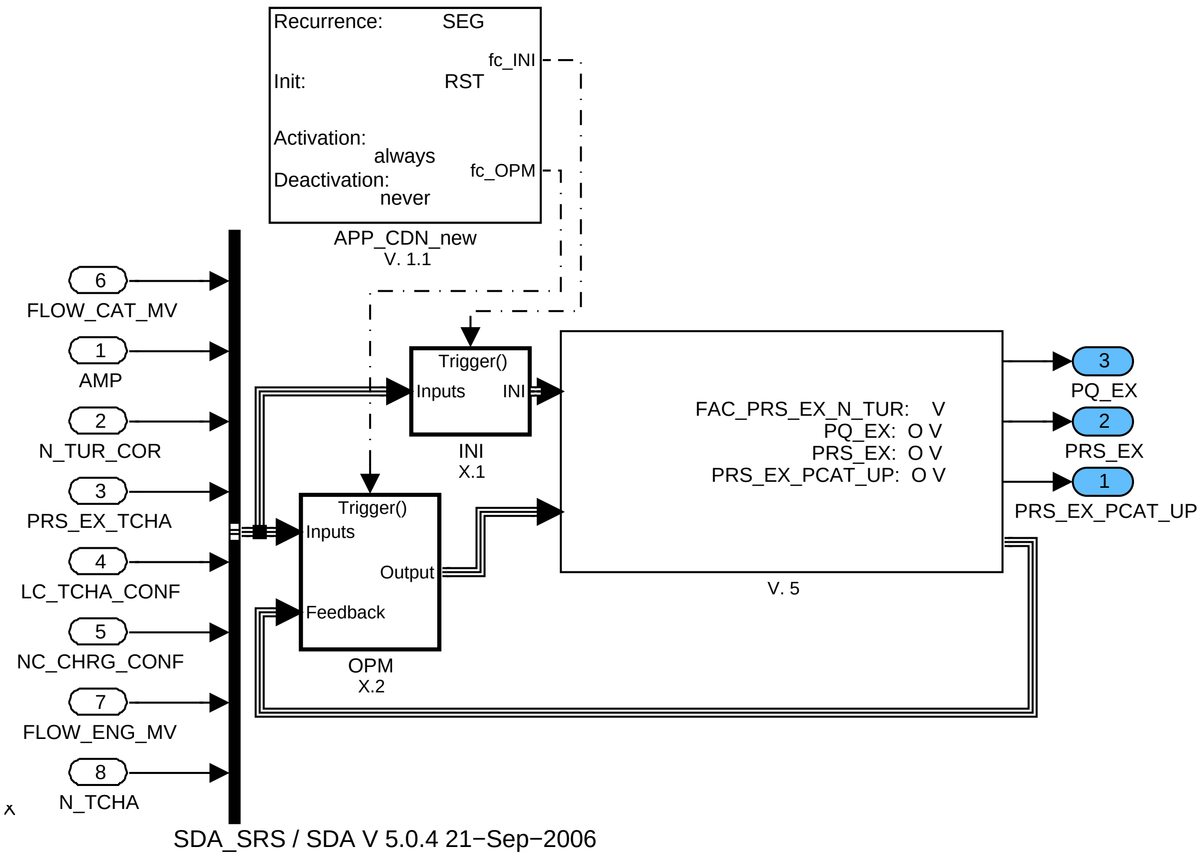


Figure 434:
Path: EGPR_DTSYSMODULE

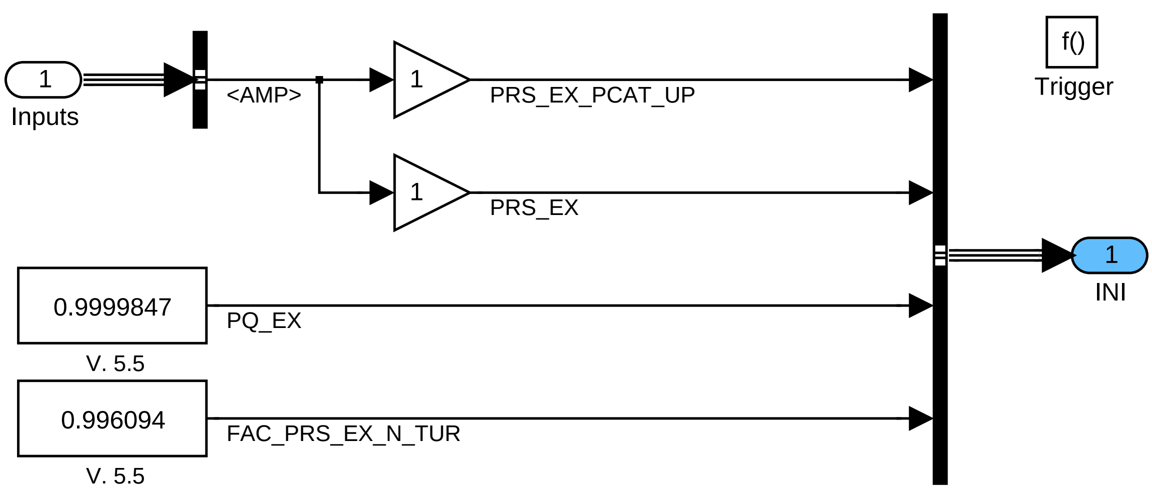


Figure 435:
Path: EGPR_DTSYSMODULE/INI

4.91.1 Calculation of the exhaust pressure

The calculation of the exhaust pressure is split into two parts. First is the pressure increase due to catalyst and second part is the pressure increase due to turbine (if available).

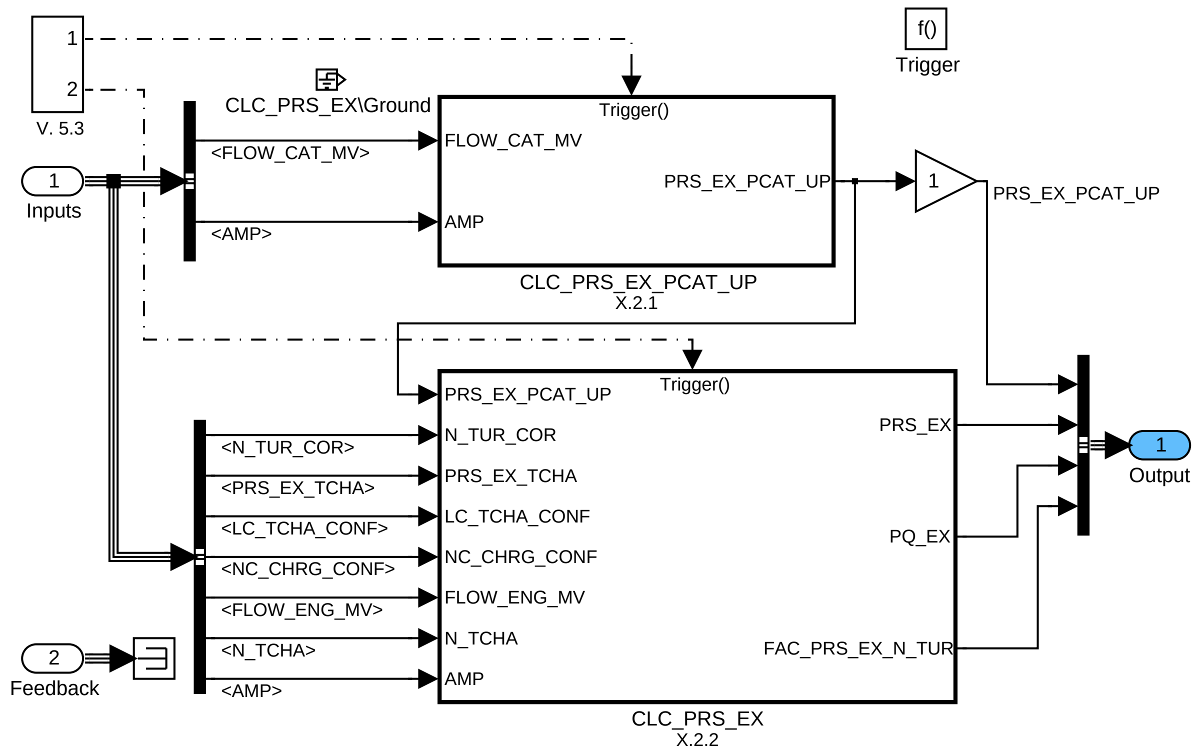


Figure 436:
Path: EGPR_DTSYSMODULE/OPM

4.91.1.1 Calculation of pressure increase due to catalyst

The pressure increase due to catalyst is mainly depending on the flow through the engine. The pressure increase itself is modelled by a simple map.

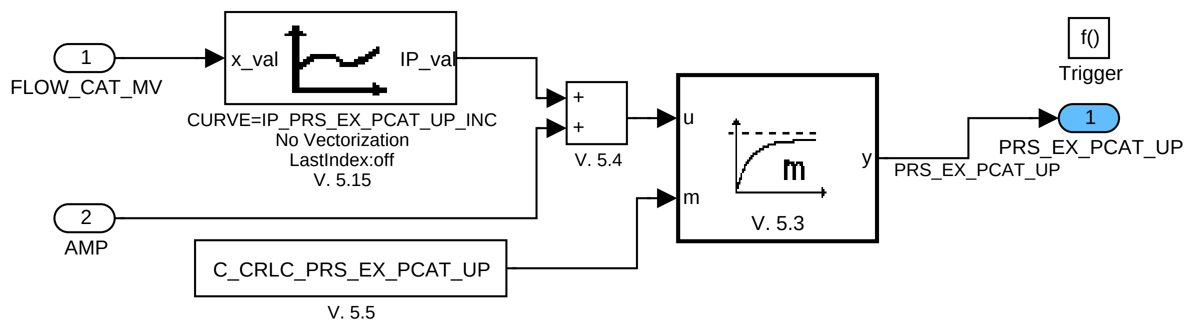


Figure 437:
Path: EGPR_DTSYSMODULE/OPM/CLC_PRS_EX_PCAT_UP

4.91.1.2 Calculation of pressure increase due to turbine

In case of turbo engine an additional pressure increase is present due to turbine. The main input for this calculation is coming from the CHRG aggregate.

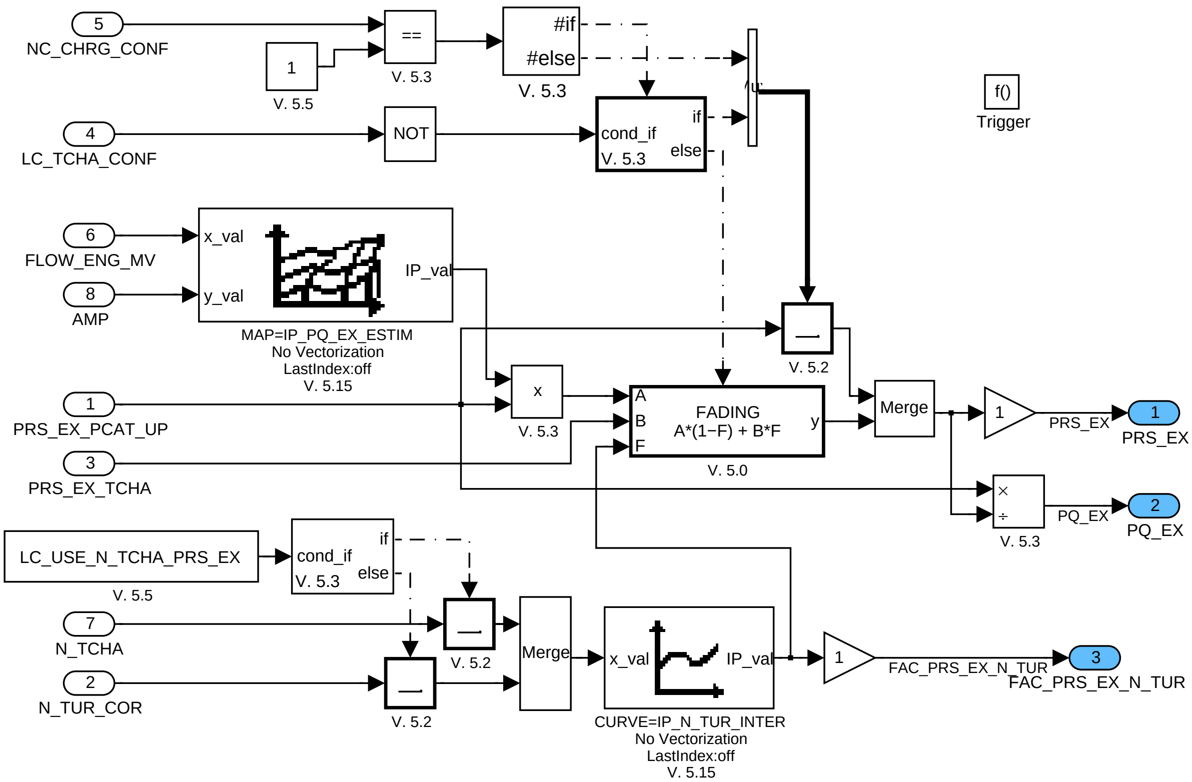


Figure 438:
Path: EGPR_DTSYSMODULE/OPM/CLC_PRS_EX

4.92 Exhaust pressure (Appl. Inc.)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FLOW_CAT_MV	O/V	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Mean value of flow through catalyst					
FLOW_ENG	O/V	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Flow engine					
FLOW_ENG_MV	O/V	0... FFFFH	0... 2047.969	0.03125	[kg/h]
Mean value of flow through engine of the exhaust banks					

Input Data:

N	MFF_SP_MV	NC_MAF_FAC_CYL	SUM_INH_INJ
NC_CYL_NR	MAF_CYL	NC_CBK_EX_NR	SAF_KGH
AV_FLOW_EGRV			

General Information

The aim of this module is the calculation of the relevant exhaust flows in order to determine the exhaust pressure.

Function description

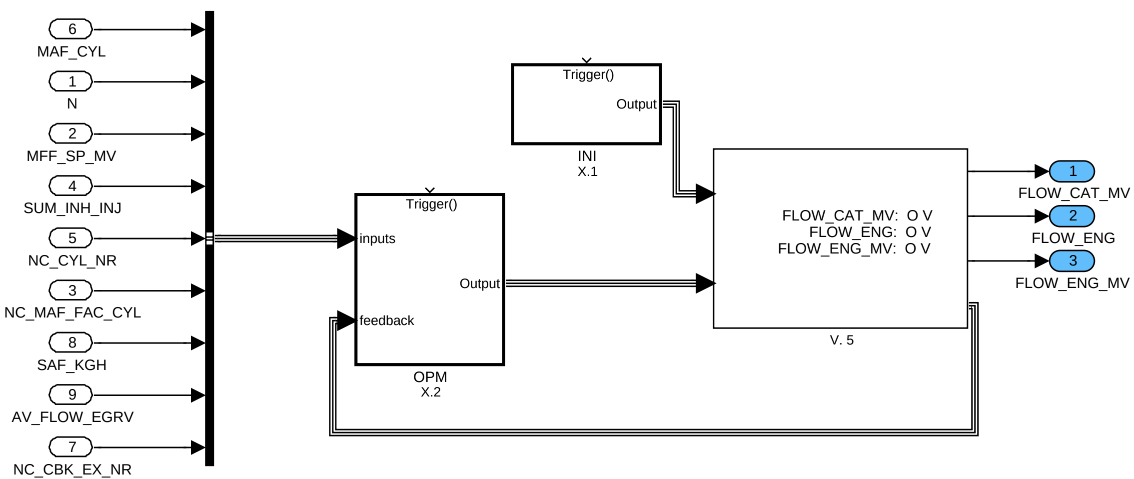


Figure 439:
Path: EGPR_MDLADNTCHAI0

4.92.1 SUBFUNCTION: INI

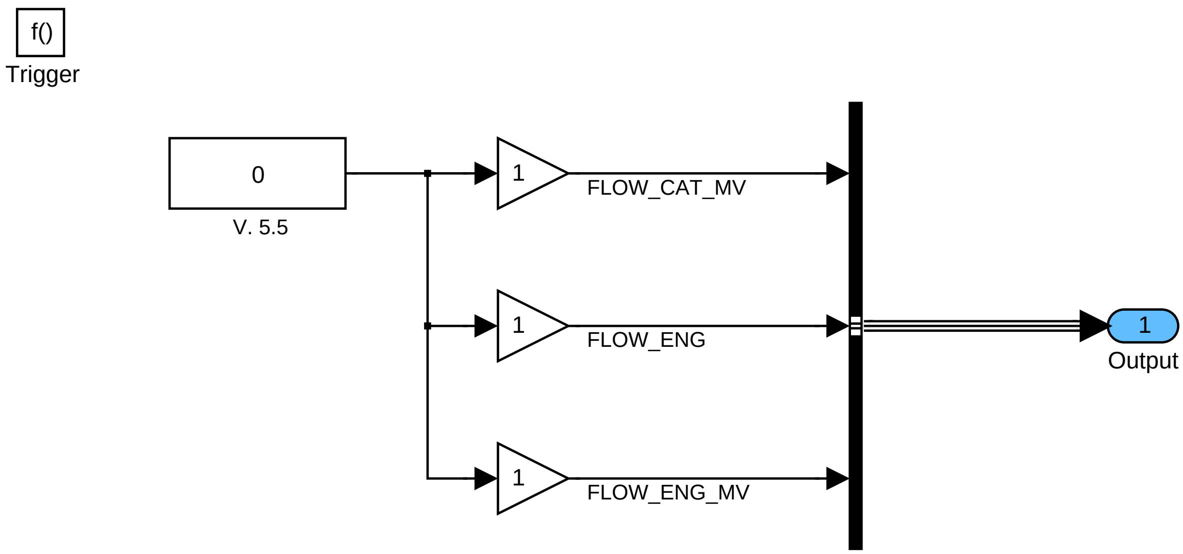


Figure 440:
Path: EGPR_MDLADNTCHAI0/INI

4.92.2 Main calculation for the flow through the catalyst

In this block the flow through the catalyst is calculated.

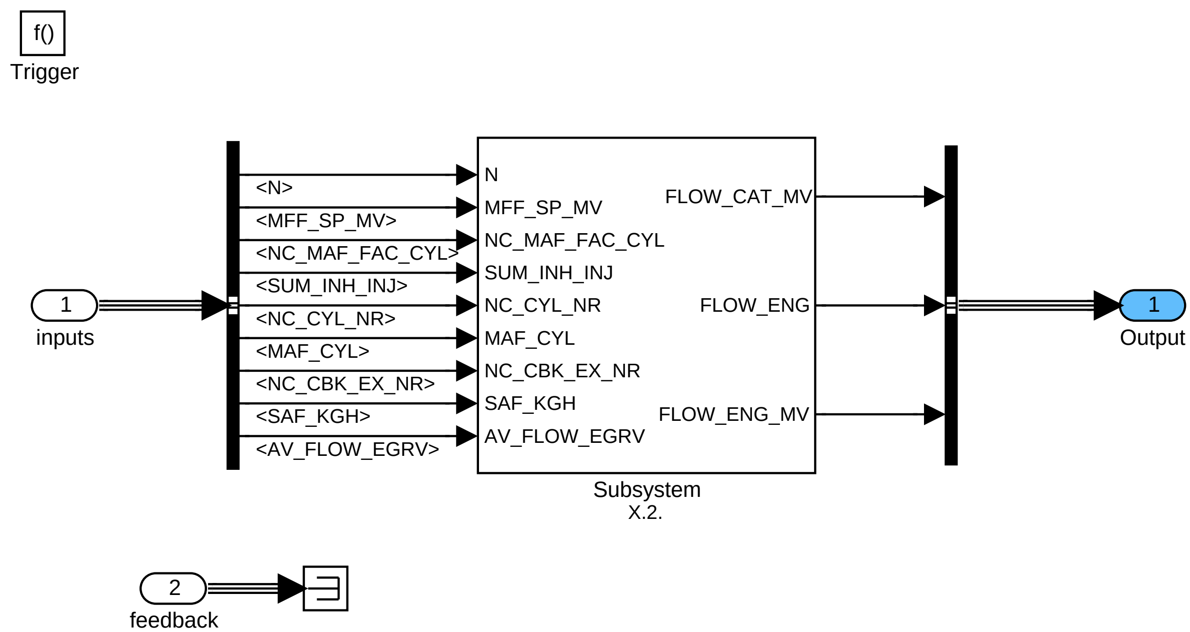


Figure 441:
Path: EGPR_MDLADNTCHAI0/OPM

4.92.2.1 Calculation of the flow through the catalyst

For the calculation of the exhaust pressure the flow through the catalyst has to be determined. This flow is consisting out of the mass air flow through into the engine and the injected fuel mass. Additionally the EGR and secondary air has to be considered. At least the mean value for the single bank has to be calculated.

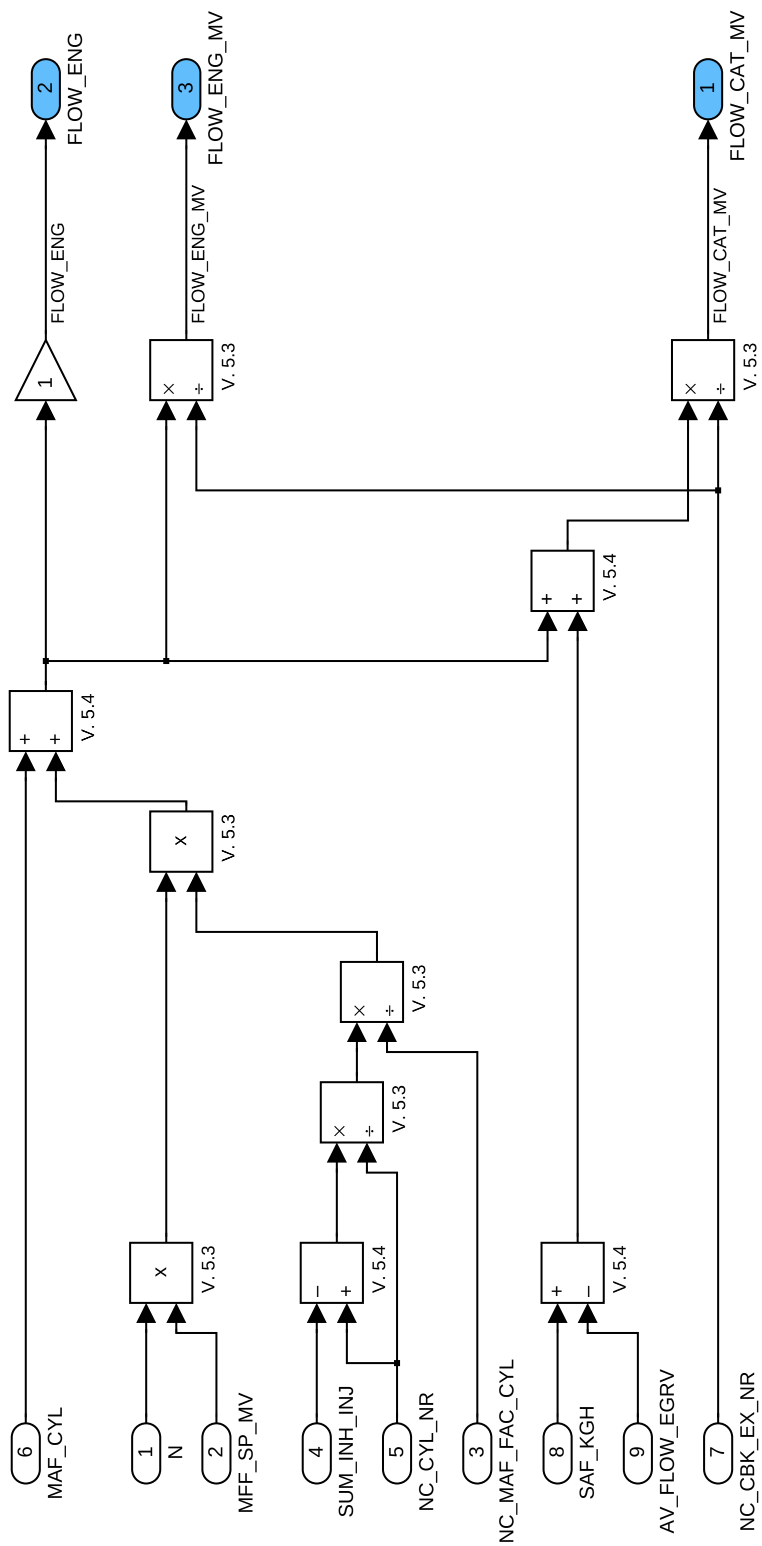


Figure 442:
Path: EGPR_MDLADNTCHAI0/OPM/Subsystem

4.93 Reduced area at waste gate

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AR_RED_WG	O/V	0...FFFFH	0...58.592855	8.9407E-4	cm²
Reduced area at wastegate					
FAC_AR_RED_WG	O/V	0...FFFFH	0...3.99993896	6.10352E-5	-
Correction factor for wastegate reduced area					

Input data:

TEG_TUR_UP_ABS	PSN_WG	LC_TCHA_CONF	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_AR_RED_WG	16	0...FFFFH	0...58.592855	8.9407E-4	cm²
LDP_PSN_WG_IP_AR_RED_WG	16	0...3FFFH	0...119.5	0.00729415	°TPS
Reduced area of waste gate					
IP_AR_RED_WG_COR	8	0...FFFFH	0...3.99993896	6.10352E-5	-
LDPM_TEG_TUR_UP_ABS_CHRG_1	8	0...FFE0H	0...2.047E+3	0.03125	K
Exhaust gas temperature correction of reduced area at the wastegate					

4.93.1 CHRG_MDLADARWG0

Function Description

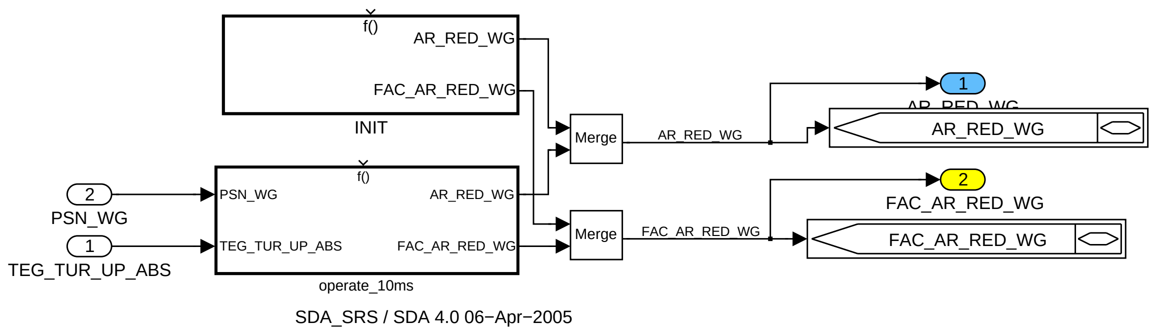


Figure 443 CHRG_MDLADARWG0

4.93.1.1 SUBFUNCTION: INIT

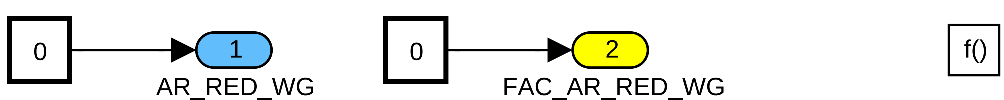


Figure 444 CHRG_MDLADARWG0/ INIT

4.93.1.2 SUBFUNCTION: operate_10ms

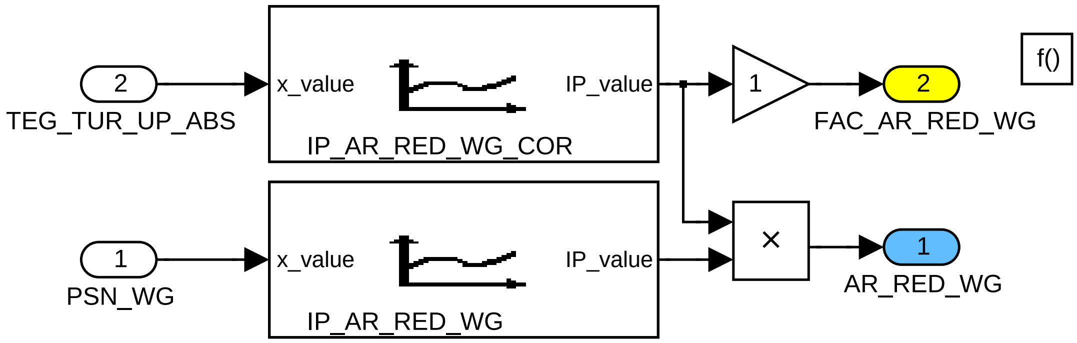


Figure 445 CHRG_MDLADARWG0/ operate_10ms

4.94 Wastegate actuator pressure

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PRS_WG_ACR	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure in wastegate actuator chamber					
PWM_WG_COR_WG_ACR	V	0...FFFFH	0...99.9984741	0.00152588	%
Temperature cirrected PWM at EPC for WG actuation					
PQ_EPC_IN	V	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure ratio at EPC input					
PQ_EPC_OUT	V	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure ratio at EPC output					

Input data:

LC_TCHA_CONF	LV_ENA_PRS_WG_ACR	PRS_WG_ACR_MES	PWM_WG
PRS_WG_ACR_EPC_H	PRS_WG_ACR_EPC_L	PRS_WG_ACR_NOT_CTL	PWM_WG_COR_OFS
PWM_WG_COR_SLOP	FAC_PWM_VB_EPC_WG		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PRS_WG_ACR	1	0...FFH	0...1	0.00392157	-
Filter coefficient for PRS_WG_ACR calculation					
C_PRS_WG_ACR_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value for wastegate actuator pressure					
C_TYP_EPC_WG_ACR	1	0...1H	0...1	1	-
Type of wastegate actuator EPC					
LC_PQ_EPC_OUT	1	0...1H	0...1	1	-
Switch whether PQ_EPC_OUT is calculated based on absolute (= 0) or relative pressure (= 1)					
LC_PRS_WG_ACR_MAN	1	0...1H	0...1	1	-
Switch whether manual value for wastegate actuator pressure is used					
IP_PRS_DIF_WG_ACR	6	0...FFFFH	-2.717E+3 ... 2.71692E+3	0.08291626	hPa
LDPM_PWM_WG_1_CHRG	6	0...FFFFH	0...99.9984741	0.00152588	%
Wastegate actuator pressure					
IP_PQ_EPC_OUT	6x8	0...FFFFH	0...15.9997559	2.44141E-4	-
LDPM_PWM_WG_1_CHRG	6	0...FFFFH	0...99.9984741	0.00152588	%
LDP_PQ_EPC_IP_PQ_EPC_OUT	8	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure ratio at the wastegate actuator EPC					

4.94.1 CHRG_MDLADPWGAC0

General information:

The wastgate actuator pressure PRS_WG_ACR can be measured or modelled. If a wastegate actuator pressure sensor is available (LV_ENA_PRS_WG_ACR = 1), the measured value PRS_WG_ACR_MES is used, otherwise a modelled value is used instead. The modelled or measured value can be overwritten by a manual value C_PRS_WG_ACR_MAN.

Two electropneumatic converter (EPC) types are covered. For C_TYP_EPC_WG_ACR = 0 both supply pressure and ambient pressure influence the wastegate actuator pressure. For C_TYP_EPC_WG_ACR = 1 only supply pressure influences the wastegate actuator pressure.

The behaviour of the EPC type 0 is modelled using two different algorithms switched by LC_PQ_EPC_OUT.

general specification

For LC_PQ_EPC_OUT = 0 IP_PQ_EPC_OUT describes the ratio between absolute pressures at the output and at the input of the EPC:

$$PQ\_EPC\_OUT = \frac{PRS\_WG\_ACR}{PRS\_WG\_ACR\_EPC\_L}$$

For LC_PQ_EPC_OUT = 1 IP_PQ_EPC_OUT describes the ratio between pressure differences at the output and at the input of the EPC:

$$PQ\_EPC\_OUT = \frac{PRS\_WG\_ACR - PRS\_WG\_ACR\_EPC\_L}{PRS\_WG\_ACR\_EPC\_H - PRS\_WG\_ACR\_EPC\_L}$$

To take into account the filling behaviour of the wastegate actuator the wastegate actuator pressure is low pass filtered.

Function Description

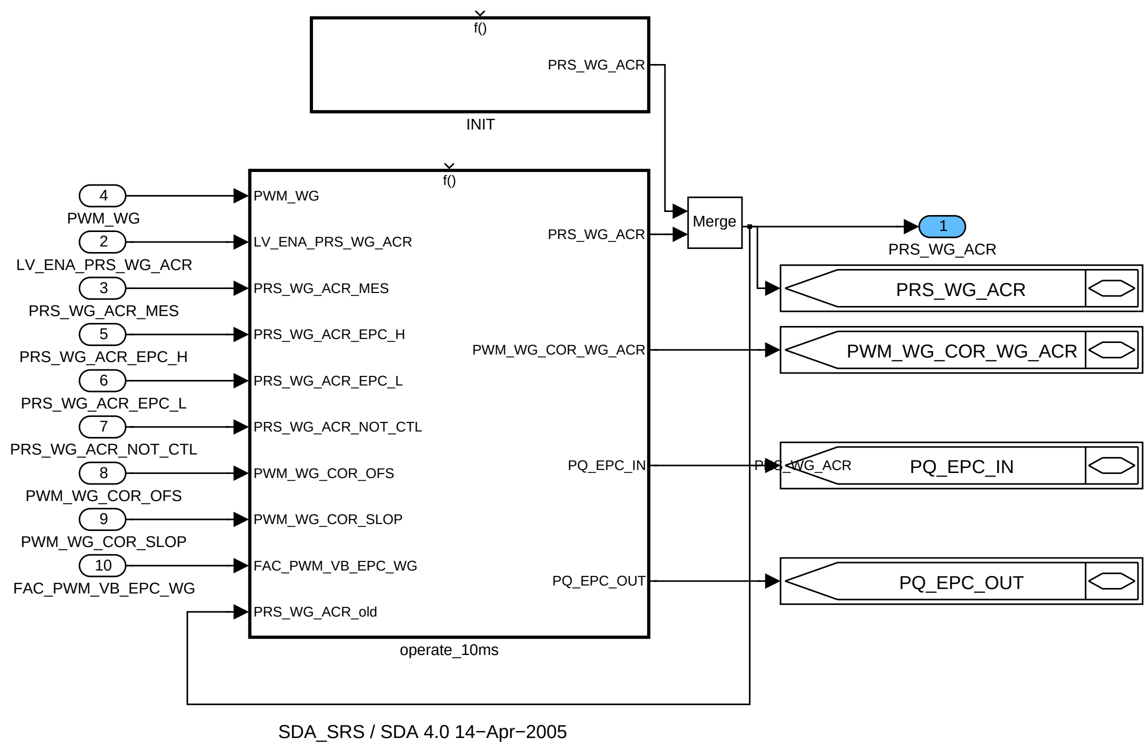


Figure 446 CHRG_MDLADPWGAC0

4.94.1.1 SUBFUNCTION: INIT

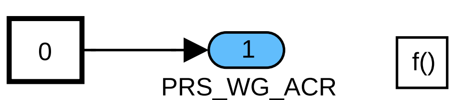


Figure 447 CHRG_MDLADPWGAC0/ INIT

4.94.1.2 SUBFUNCTION: operate_10ms

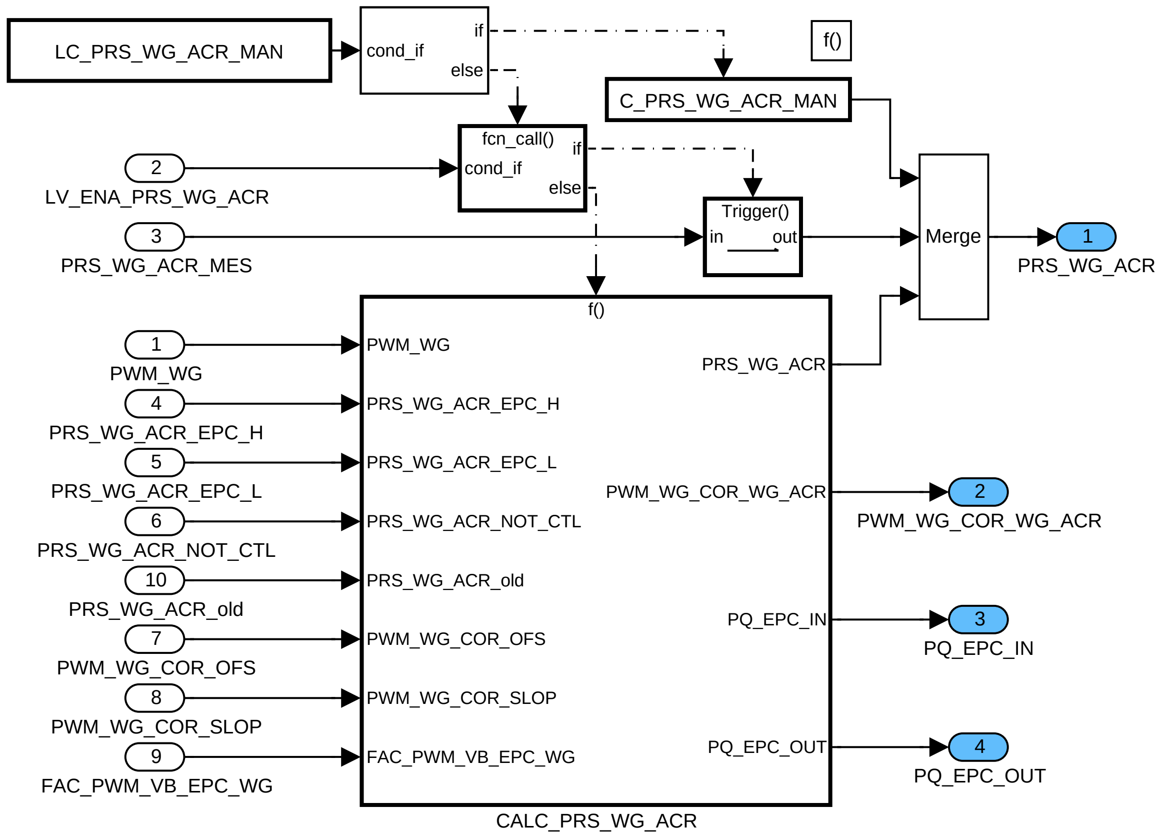


Figure 448 CHRG_MDLADPWGAC0/ operate_10ms

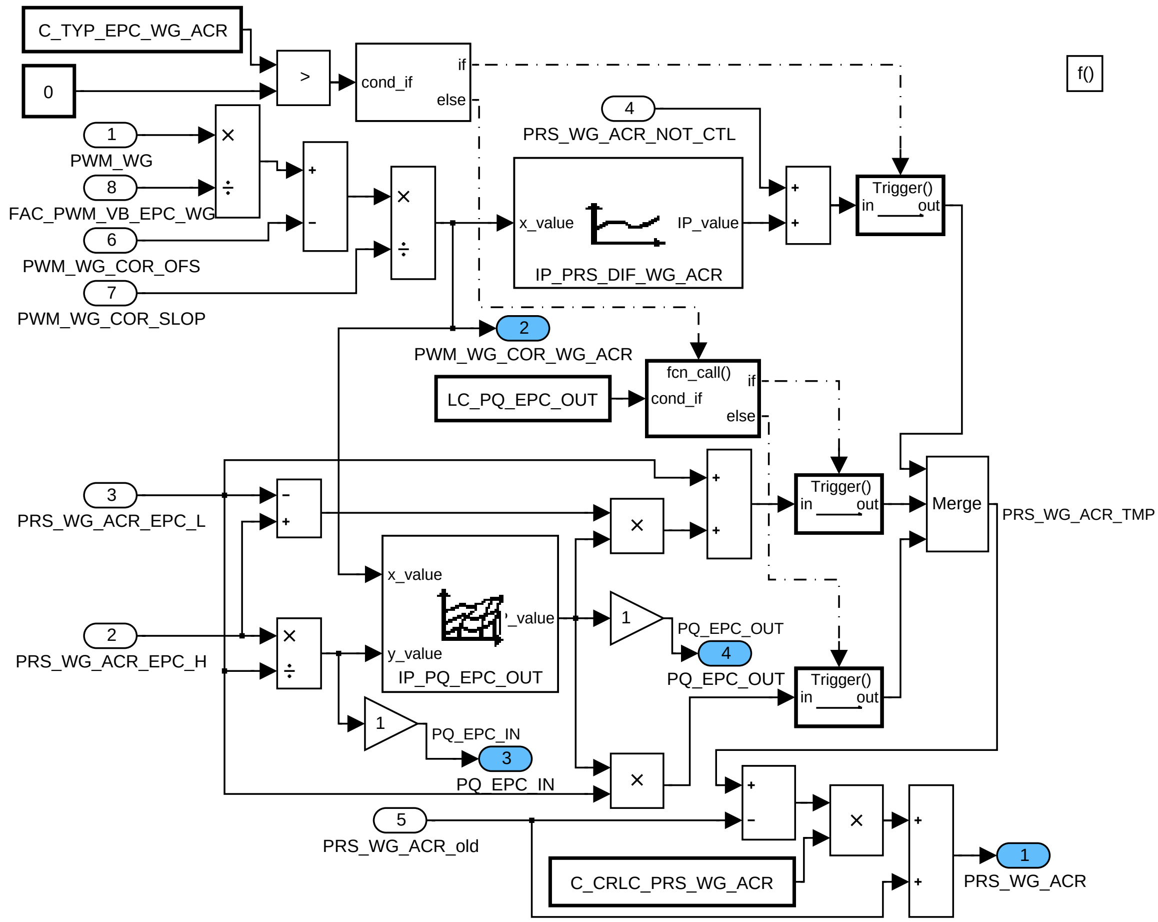


Figure 449 CHRG_MDLADPWGAC0/ operate_10ms/ CALC_PRS_WG_ACR

4.95 Wastegate position

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PSN_WG_ACR	O/V	0...FFFFH	0...63.999023	9.76562E-4	mm
Wastegate actuator position					
PSN_WG	O/V	0...3FFFH	0...119.5	0.00729415	°TPS
Wastegate position					
FOC_WG_ACR_SPR	V	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	N
Force produced by wastegate actuator spring					

Input data:

LC_TCHA_CONF	PRS_WG_ACR	PRS_WG_ACR_NOT_CTL	FLOW_ENG_TCHA
IP_FOC_WG_ACR_SPR			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PSN_WG_ACR	1	0...FFH	0...0.99609375	0.00390625	-
Correlation constant for wastegate actuator position (low pass filter)					
C_PSN_WG_MAN	1	0...3FFFH	0...119.5	0.00729415	°TPS
Manual value for wastegate position					
LC_PSN_WG_MAN	1	0...1H	0...1	1	-
Switch whether manual value for wastegate position is used					
IP_PSN_WG	16	0...3FFFH	0...119.5	0.00729415	°TPS
LDP_PSN_WG_ACR_IP_PSN_WG	16	0...FFFFH	0...63.999023	9.76562E-4	mm
Interpretation table of the waste gate position					
IP_PSN_WG_ACR_FLOW_ENG	12x14	0...FFFFH	0...63.999023	9.76562E-4	mm
LDP_FLOW_ENG_IP_PSN_WG_ACR	12	0...FFFFH	0...2.04797E+3	0.03125	kg/h
LDP_PRS_ACR_WG_IP_PSN_WG_ACR	14	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Position of wastegate actuator depending on engine flow and wastegate actuator pressure					

4.95.1 CHRG_MDLADPSNWG0

The wastegate position PSN_WG is modelled by means of a map depending on the flow engine and on pressure difference at the wastegate actuator PRS_DIF_WG_ACR. The modelled value can be overwritten by a manual value C_PSN_WG_MAN

The present specification describes a wastegate position model based on wastegate actuator pressure for a wastegate with one controlled pressure.

The pneumatic force at the actuator FOC_WG_ACR_SP has to compensate the sum of the spring force and exhaust gas pressure force. The force at wastegate actuator caused by spring is negative calibrated. It means, negative force opens the wastegate.

Function Description

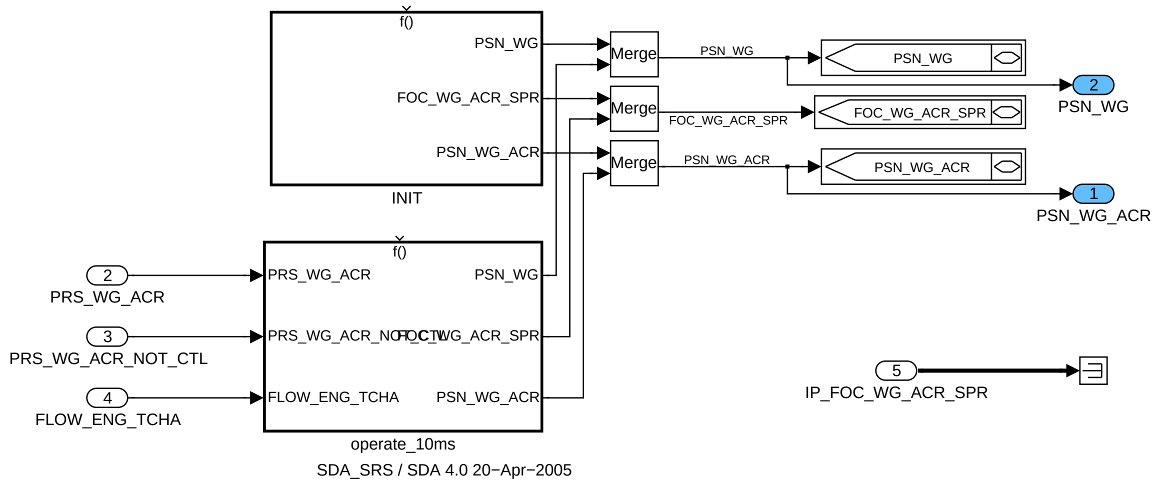


Figure 450 CHRG_MDLADPSNWG0

4.95.1.1 SUBFUNCTION: INIT

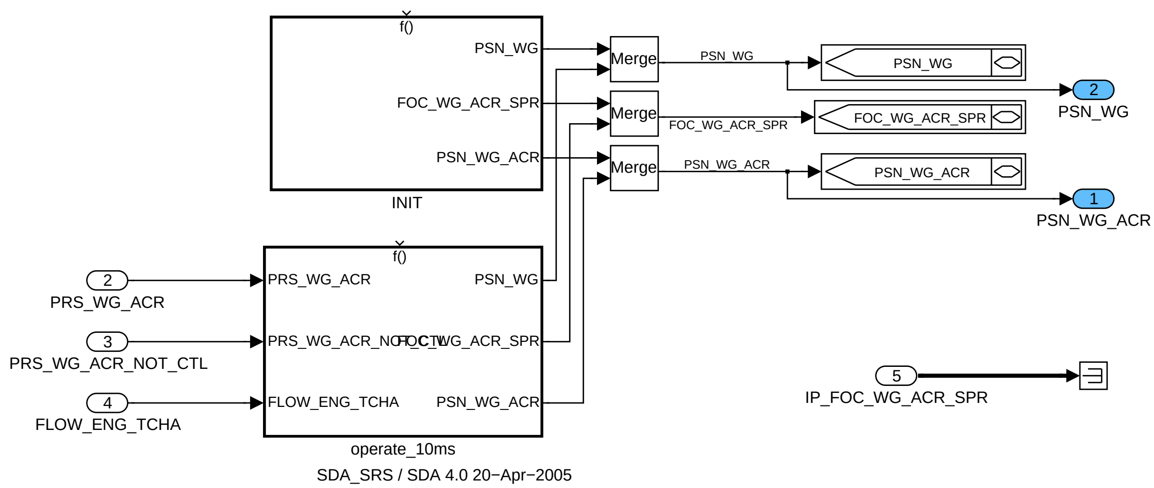


Figure 451 CHRG_MDLADPSNWG0/ INIT

4.95.1.2 SUBFUNCTION: operate_10ms

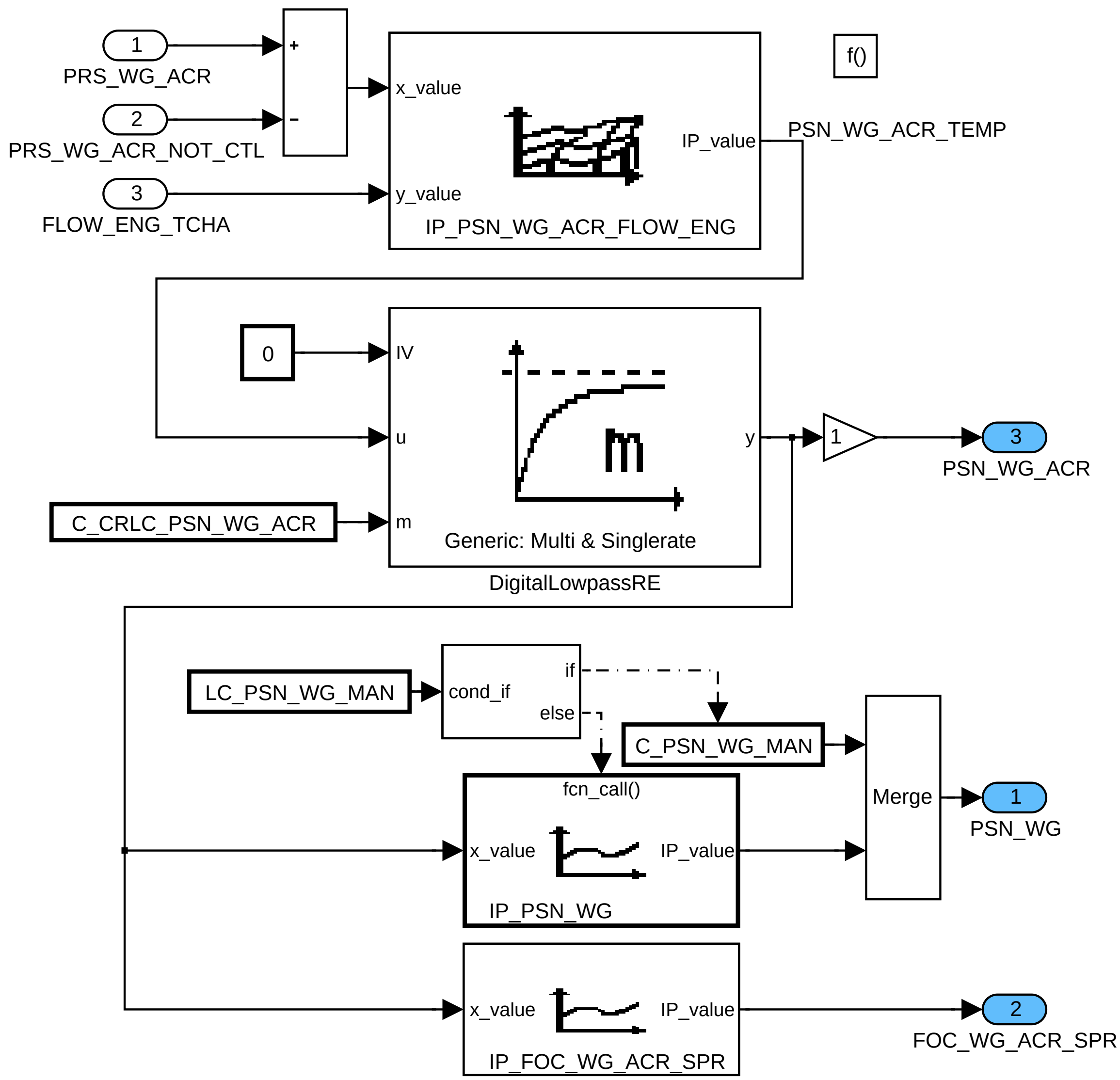


Figure 452 CHRG_MDLADPSNWG0/ operate_10ms

4.96 Recirculation actuator position

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AR_RED_RCL_MDL[NC_NR_TCHA]	O/V	0...FFFFH	0...58.592855	8.9407E-4	cm²
Modeled reduced area for recirculation actuator (inlet branch specific)					
PSN_RCL_MDL[NC_NR_TCHA]	O/V	0...FFFFH	0...99.9984741	0.00152588	%
Recirculation actuator position modelled value: closed = 0% / fully opened = 99.998474%					
PSN_RCL[NC_NR_TCHA]	O/V	0...FFFFH	0...99.9984741	0.00152588	%
Recirculation actuator position: closed = 0% / fully opened = 99.998474%					
PSN_RCL_MDL_MV	O/V	0...FFFFH	0...99.9984741	0.00152588	%
Recirculation actuator position modelled value: closed = 0% / fully opened = 99.998474% (inlet branch average)					
PRS_EPC_RCL_DOWN_FALL_CLOSE[NC_NR_TCHA]	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold value for fully close RCL valve in case of falling PRS_EPC_RCL_DOWN					
PRS_EPC_RCL_DOWN_FALL_OPEN	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold value for fully open RCL valve in case of falling PRS_EPC_RCL_DOWN					
PRS_EPC_RCL_DOWN_RISE_THD[NC_NR_TCHA]	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold to define "RCL valve close/open" in case of rising PRS_EPC_RCL_DOWN					
PRS_RCL_UP_COR[NC_NR_TCHA]	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
PRS_RCL_UP value corrected versus PRS_CHA_UP					
STATE_PRS_EPC_RCL_DOWN	V	0H 1H 2H	CON FALL RISE	1	-
Information byte for PRS_EPC_RCL_DOWN: Falling / Rising / Constant pressure (no state machine)					

Input data:

AMP	LC_TCHA_CONF	PSN_RCL_SP	PRS_CHA_UP[NC_NR_TCHA]
PRS_RCL_UP[NC_NR_TCHA]	PRS_EPC_RCL_DOWN	LV_ENA_PSN_RCL	PSN_RCL_MES
LV_ERR_PLAUS_CLOSE_RCL	LV_ERR_PLAUS_OPEN_RCL	C_AR_RED_RCL_MIN	C_AR_RED_RCL_MAX
NC_NR_TCHA			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PSN_RCL_MDL1	1	0...FFH	0...0.99609375	0.00390625	-
Filter coefficient for low pass filtering of PSN_RCL_SP for PSN_RCL_MDL1 calculation (typical value=1)					
C_CRLC_PSN_RCL_MDL2	1	0...FFH	0...0.99609375	0.00390625	-
Filter coefficient for low pass filtering of PSN_RCL_SP for PSN_RCL_MDL2 calculation (typical value=1)					
C_FAC_AR_RED_RCL_MAX	1	0...FFH	0...0.99609375	0.00390625	-
Factor on AR_RED_RCL_MAX for AR_RED characteristic definition (typical value = 1)					
C_FAC_PRS_CHA_UP	1	0...FFH	0...1.9921875	0.0078125	-
Factor on difference PRS_CHA_UP - C_FAC_PRS_CHA_UP (typical value = 1)					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PRS_CHA_UP_REF	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
PRS_CHA_UP reference value for AR_RED_RCL_MDL determination (typical value = 1000)					
C_PRS_EPC_RCL_DOWN_DIF_THD_FALL	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN difference threshold used when falling PRS_EPC_RCL_DOWN value (typical value=0)					
C_PRS_EPC_RCL_DOWN_DIF_THD_RISE	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN difference threshold used when rising PRS_EPC_RCL_DOWN value (typical value=0)					
C_PSN_RCL_MAN	1	0...FFFFH	0...99.9984741	0.00152588	%
Manual value for recirculation actuator position					
LC_AR_RED_RCL_MDL	1	0...1H	0...1	1	-
Switch to choose AR_RED_RCL model					
LC_ERR_PLAUS_RCL_ACR_SUPP	1	0...1H	0...1	1	-
Suppression of LV_ERR_PLAUS_RCL_ACR effect on PSN_RCL_MDL1, AR_RED_RCL_MDL, AR_RED_RCL					
LC_PSN_RCL_MAN	1	0...1H	0...1	1	-
Switch whether manual value for recirculation actuator position is used (typical value=0)					
LC_PSN_RCL_MDL	1	0...1H	0...1	1	-
Switch to choose the model for RCL actuator position					
LC_PSN_RCL_MDL2_BAS	1	0...1H	0...1	1	-
selection of PSN_RCL_MDL2_BAS definition (depending on if complete PSN characteristic available or not)					
IP_PRS_EPC_RCL_DOWN_ADD_COR	6	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_PRS_RCL_UP_2_CHRG	6	0...7FFFH	0...2.71696E+3	0.08291752	hPa
Additive correction on PRS_EPC_RCL_DOWN threshold values function of PRS_CHA_UP					
IP_PRS_EPC_RCL_DOWN_FALL_CLOSE	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_PRS_RCL_UP_2_CHRG	6	0...7FFFH	0...2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold value for fully close RCL valve in case of falling PRS_EPC_RCL_DOWN					
IP_PRS_EPC_RCL_DOWN_FALL_OPEN	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_PRS_RCL_UP_2_CHRG	6	0...7FFFH	0...2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold value for fully open RCL valve in case of falling PRS_EPC_RCL_DOWN					
IP_PRS_EPC_RCL_DOWN_RISE_THD	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_PRS_RCL_UP_2_CHRG	6	0...7FFFH	0...2.71696E+3	0.08291752	hPa
PRS_EPC_RCL_DOWN threshold to define "RCL valve close/open" in case of rising PRS_EPC_RCL_DOWN					
IP_PSN_RCL	12	0...FFFFH	0...99.9984741	0.00152588	%
LDP_AR_RED_RCL_MDL_IP_PSN_RCL	12	0...FFFFH	0...58.592855	8.9407E-4	cm²
RCL position versus modeled RCL reduced area (inverted characteristic of IP_AR_RED_RCL)					
IP_FAC_AR_RED_RCL_MDL	6x18	0...FFH	0...0.99609375	0.00390625	-
LDPM_PRS_RCL_UP_2_CHRG	6	0...7FFFH	0...2.71696E+3	0.08291752	hPa
LDP_PRS_EPC_RCL_DOWN_IP_FAC_AR	18	0...FFFFH	0...5.434E+3	0.08291752	hPa
Modeled reduced area of the recirculation actuator (used if LC_AR_RED_RCL_MDL = 1)					

4.96.1 CHRG_MDLADPSNRC0

The recirculation actuator position PSN_RCL can be measured or modelled. If a recirculation actuator position sensor is available and delivers a valid value (LV_ENA_PSN_RCL=1), the measured value PSN_RCL_MES is used, otherwise a modelled value is used instead. A manual value (_MAN) can overwrite both measured and model value.

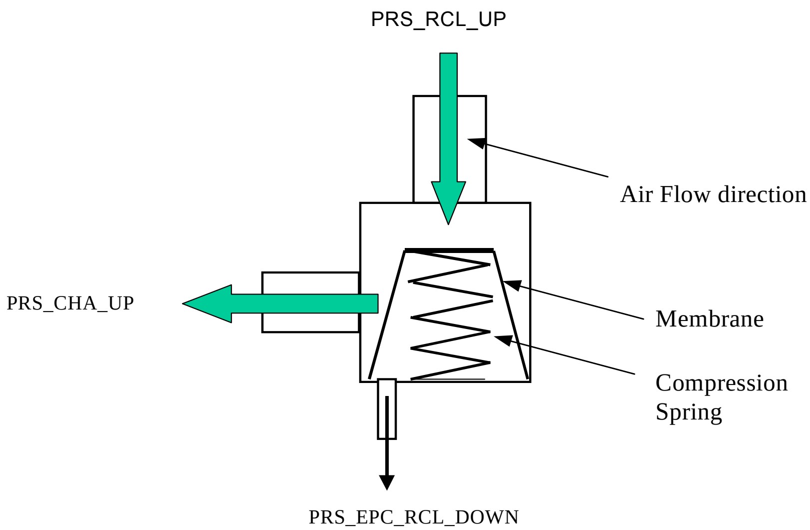
general specification

Two models are proposed:

_ simple model: the model value is a low pass filtered set point PSN_RCL_SP

_ more sophisticated models: the modeled reduced section of the RCL actuator AR_RED_RCL_MDL is determined based on the three pressures applied on the actuator: PRS_CHA_UP, PRS_RCL_UP and PRS_EPC_RCL_DOWN . The position of the actuator is defined based on AR_RED_RCL_MDL.

Example of RCL hardware



The position of the RCL valve depends on the balance forces of the compression spring and the three pressures applied on the membrane.

Application Condition

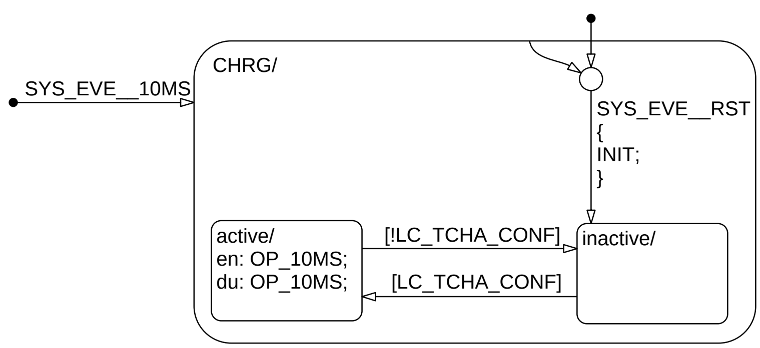


Figure 453 CHRG_MDLADPSNRC0/ APP_CDN/ Chart

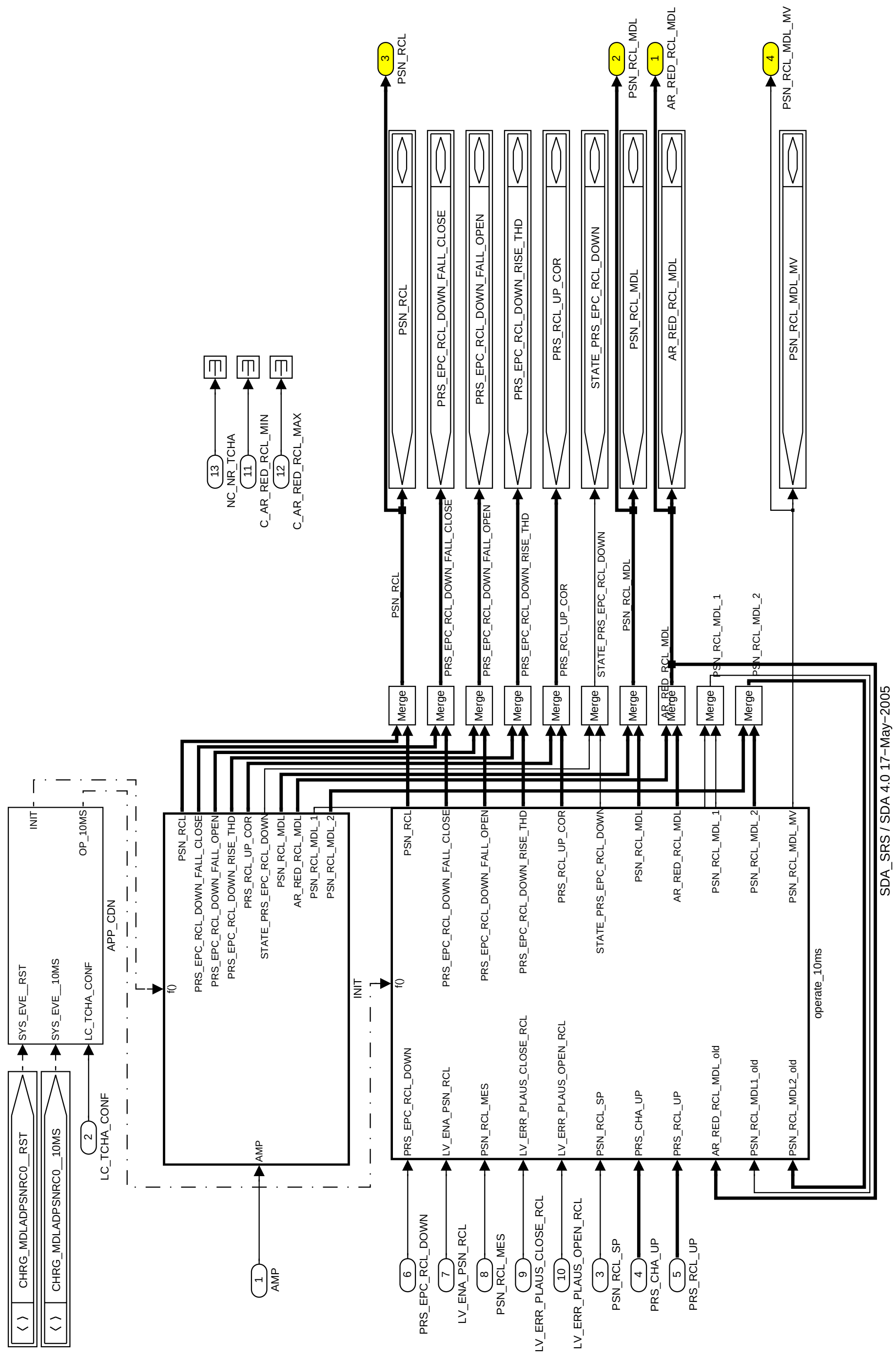


Figure 454 CHRG_MDLADPSNRC0

4.96.1.1 SUBFUNCTION: INIT

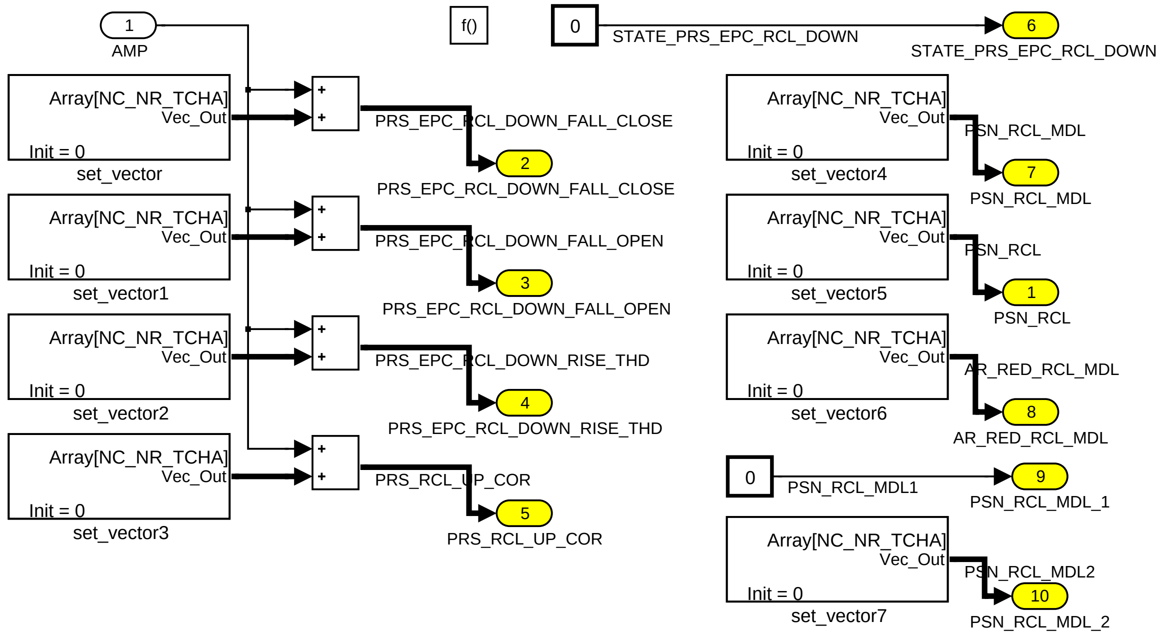


Figure 455 CHRG_MDLADPSNRC0/ INIT

4.96.1.2 SUBFUNCTION: operate_10ms

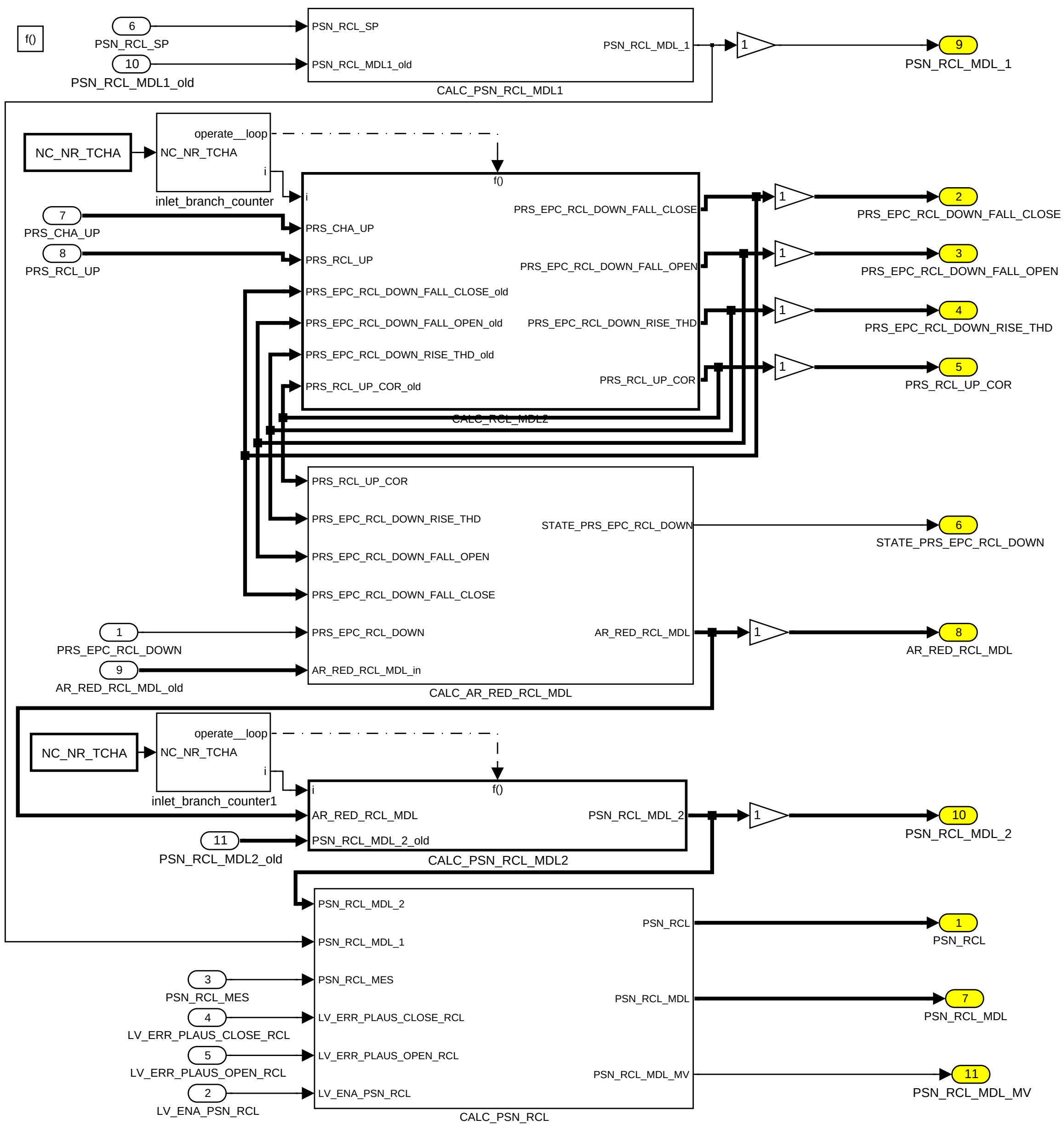


Figure 456 CHRG_MDLADPSNRC0/ operate_10ms

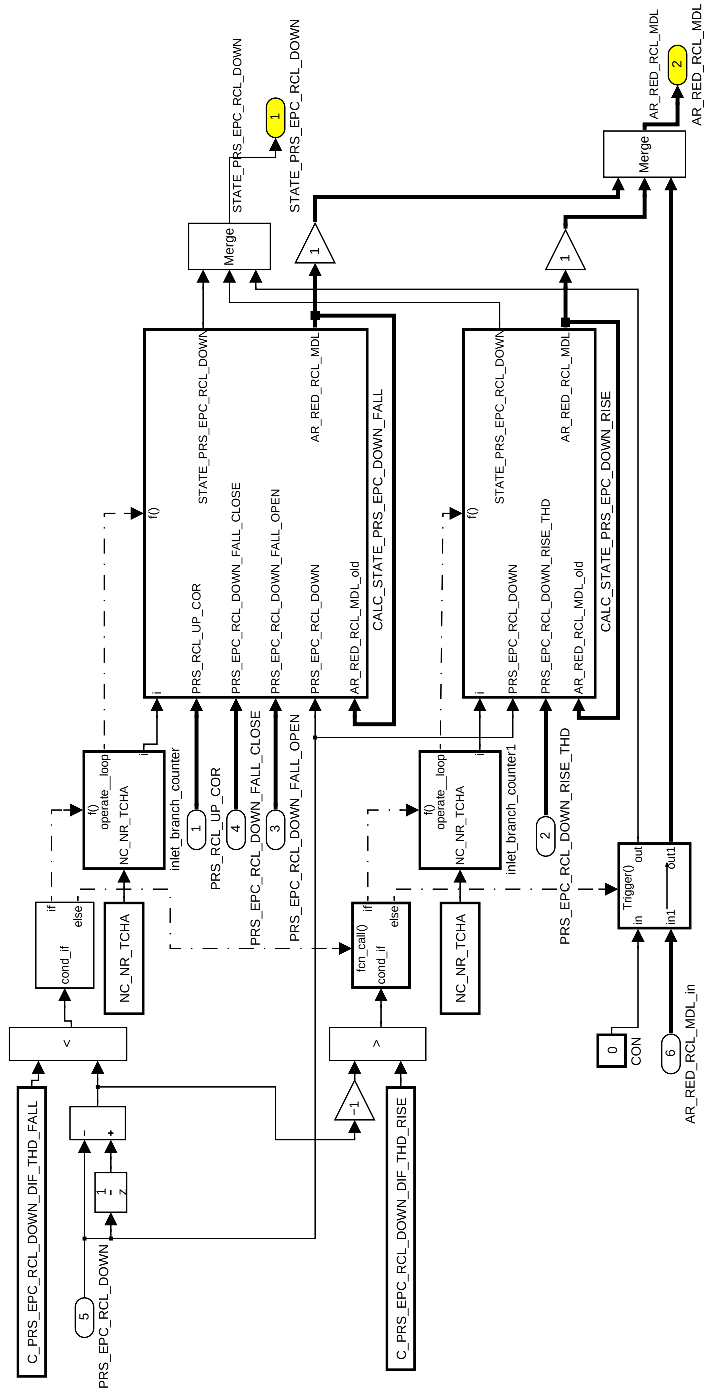


Figure 457 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL

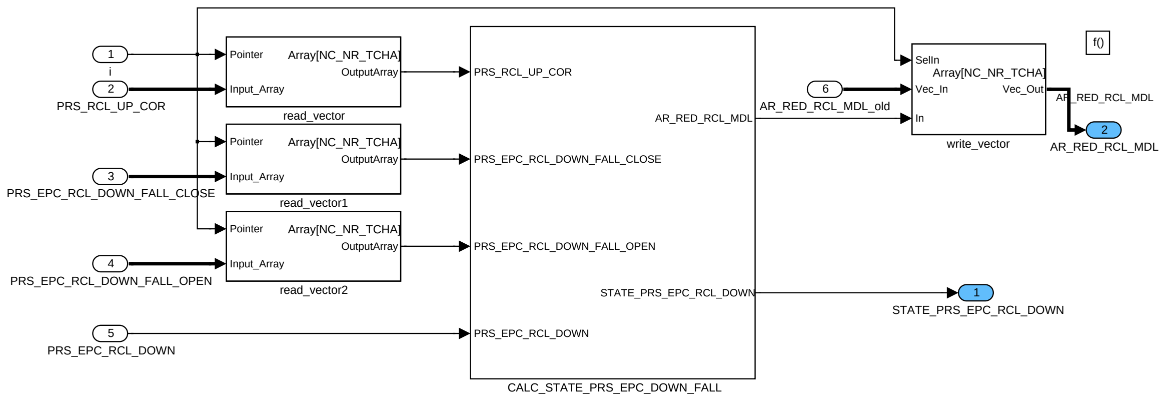


Figure 458 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ CALC_STATE_PRS_EPC_DOWN_FALL

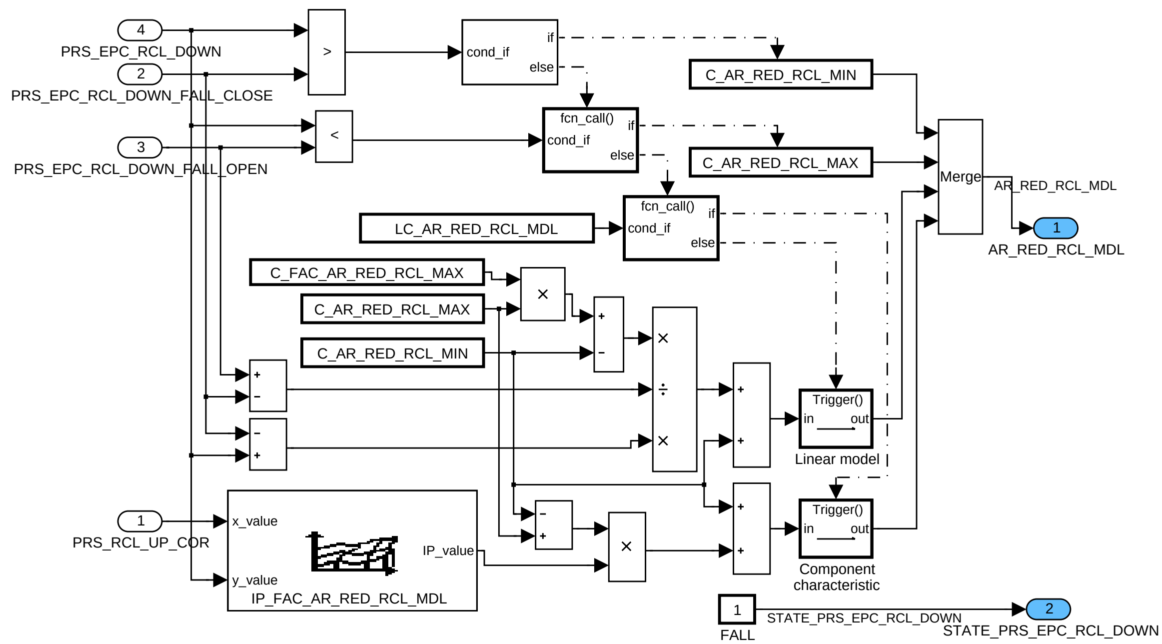


Figure 459 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ CALC_STATE_PRS_EPC_DOWN_FALL/ CALC_STATE_PRS_EPC_DOWN_FALL

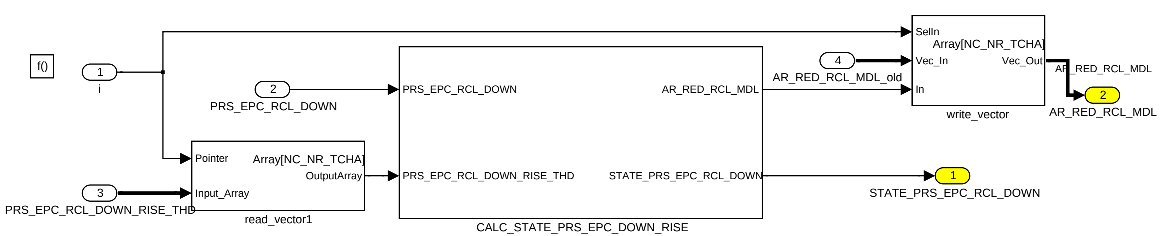


Figure 460 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ CALC_STATE_PRS_EPC_DOWN_RISE

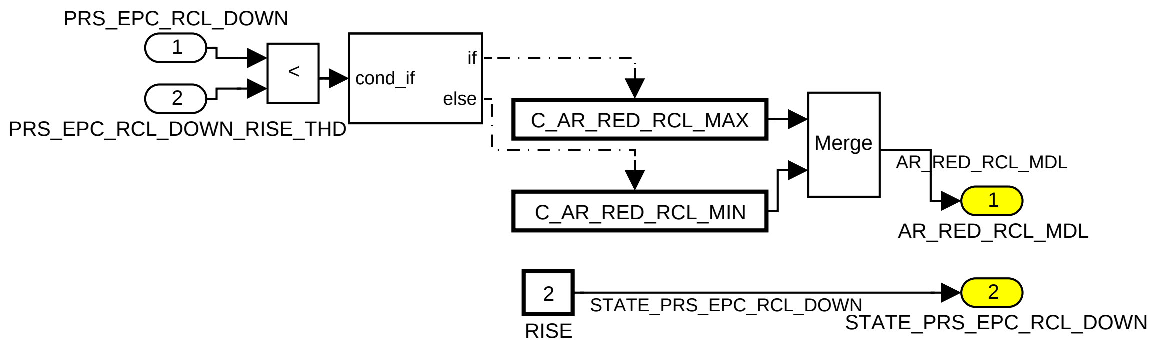


Figure 461 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ CALC_STATE_PRS_EPC_DOWN_RISE/ CALC_STATE_PRS_EPC_DOWN_RISE

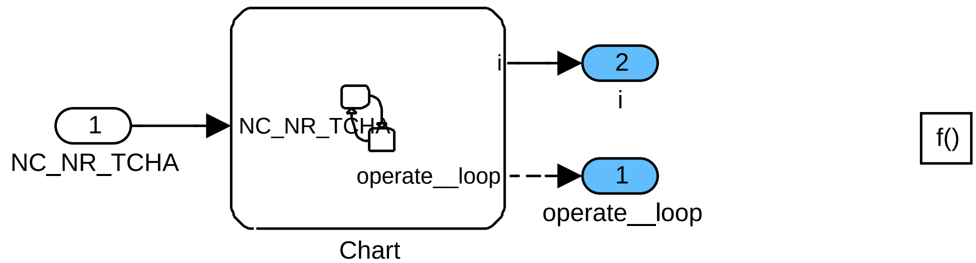


Figure 462 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ inlet_branch_counter

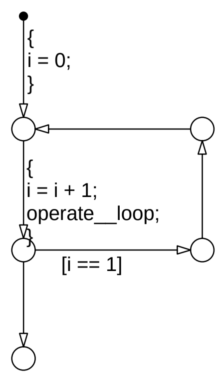


Figure 463 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ inlet_branch_counter/ Chart

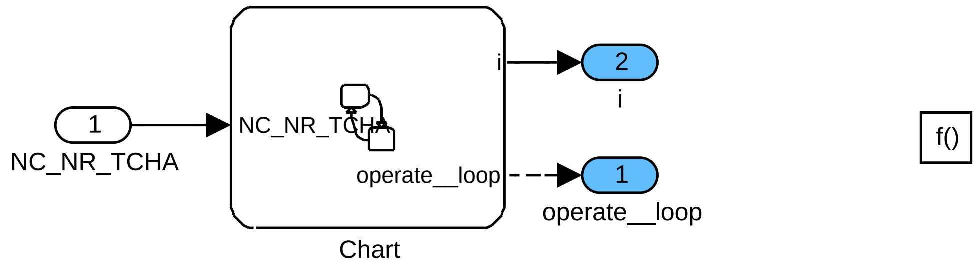


Figure 464 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ inlet_branch_counter1

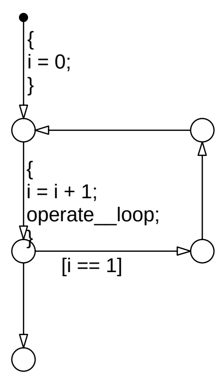


Figure 465 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_AR_RED_RCL_MDL/ inlet_branch_counter1/ Chart

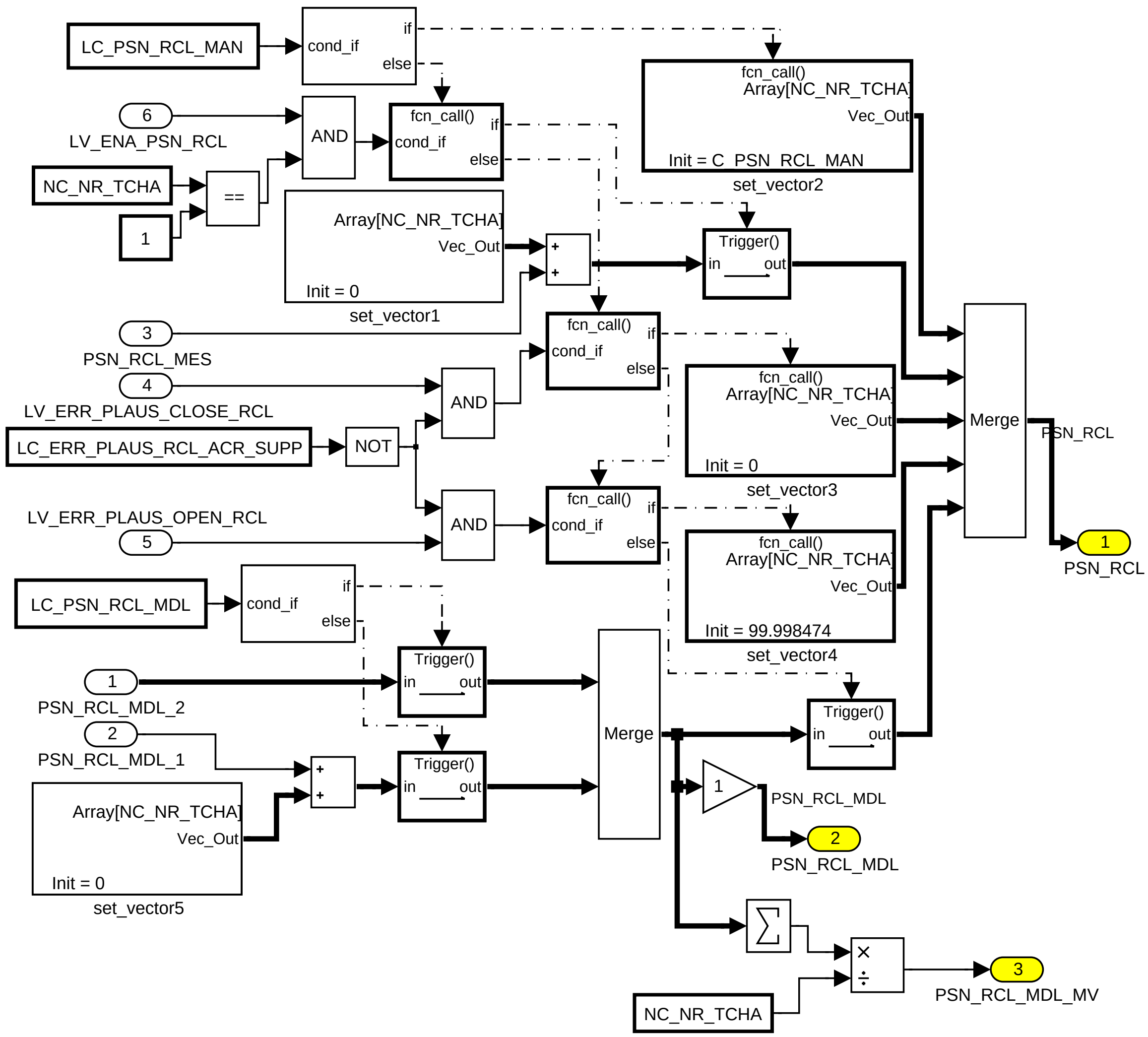


Figure 466 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_PSN_RCL

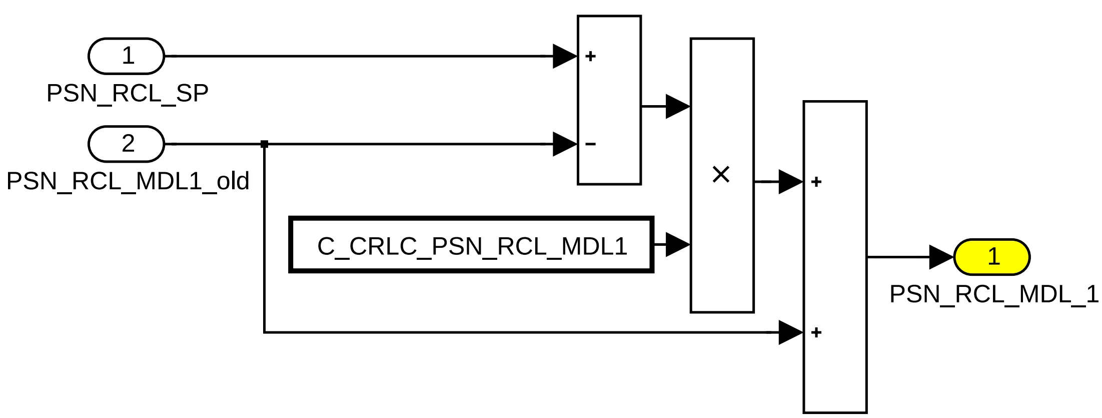


Figure 467 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_PSN_RCL_MDL1

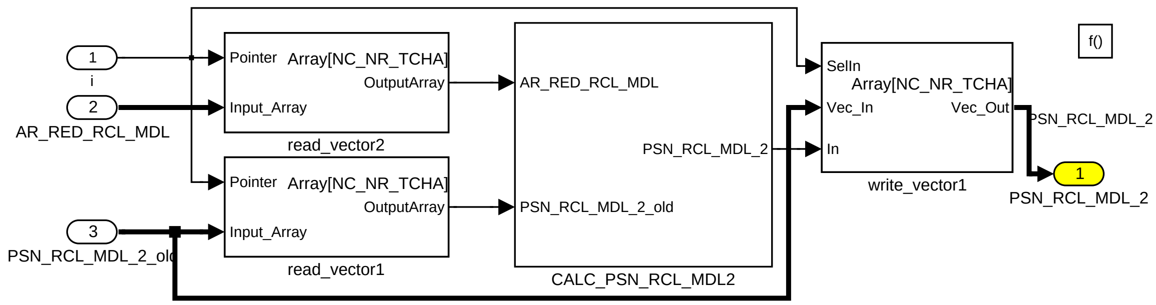


Figure 468 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_PSN_RCL_MDL2

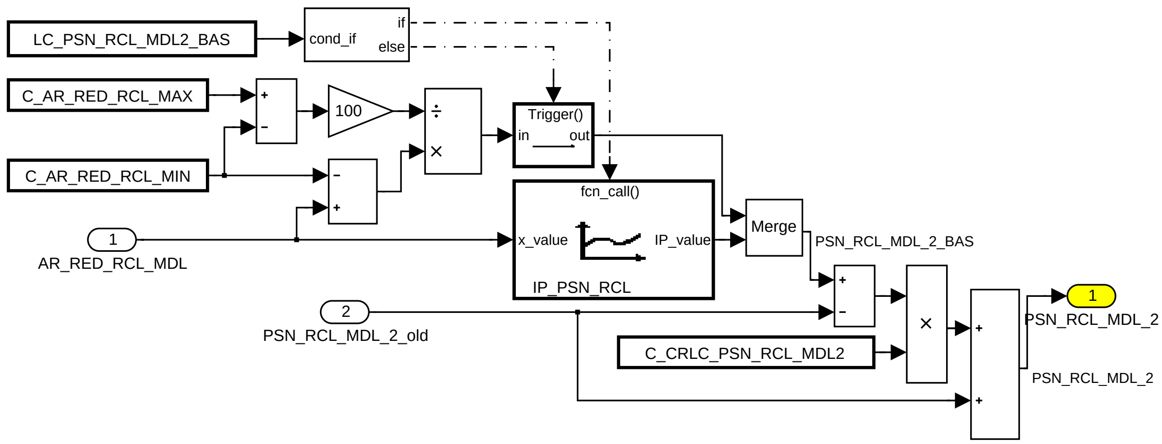


Figure 469 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_PSN_RCL_MDL2/ CALC_PSN_RCL_MDL2

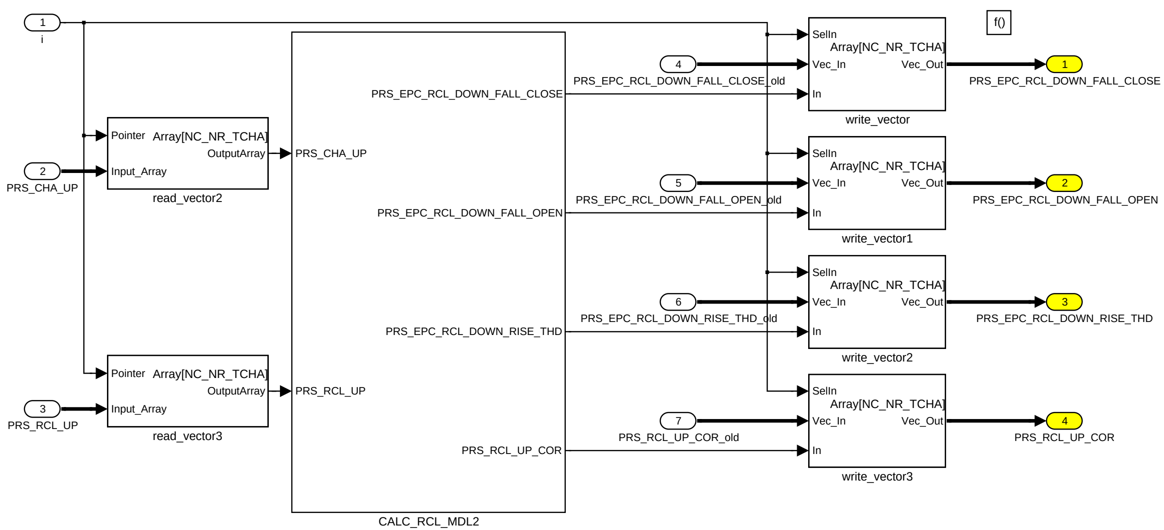


Figure 470 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_RCL_MDL2

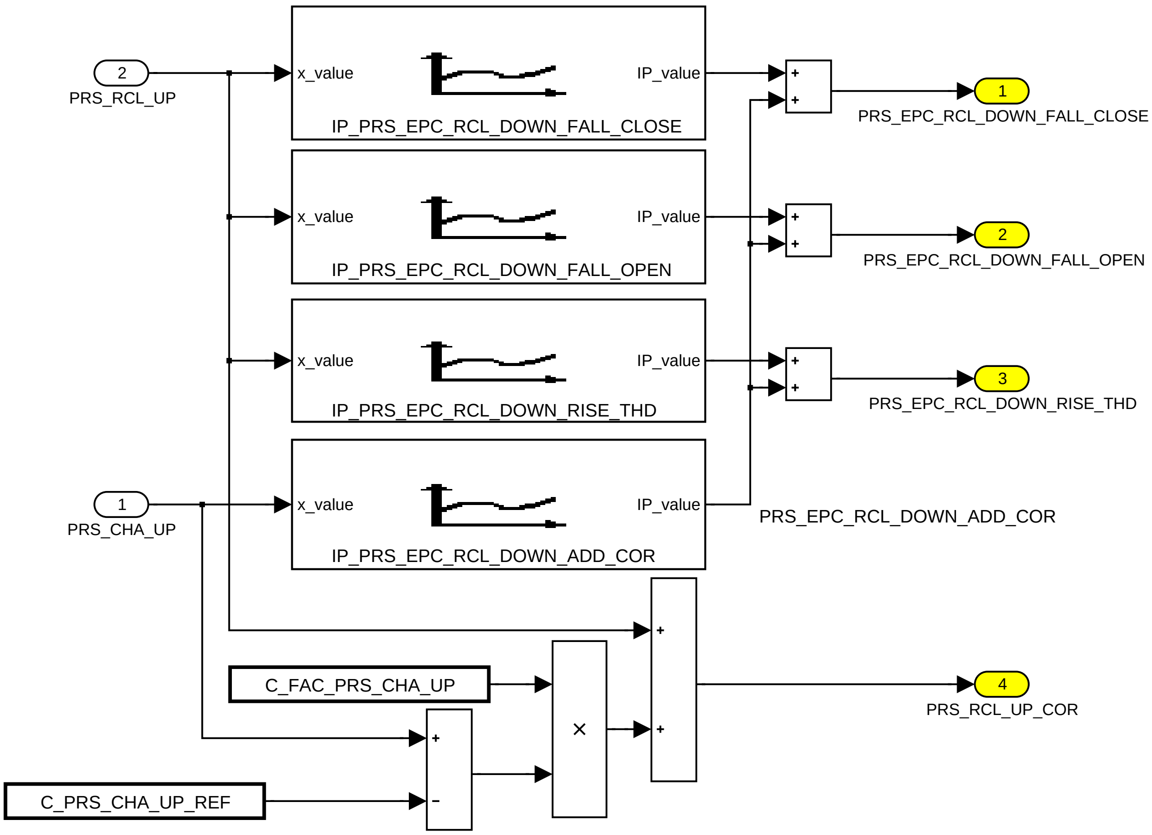


Figure 471 CHRG_MDLADPSNRC0/ operate_10ms/ CALC_RCL_MDL2/ CALC_RCL_MDL2

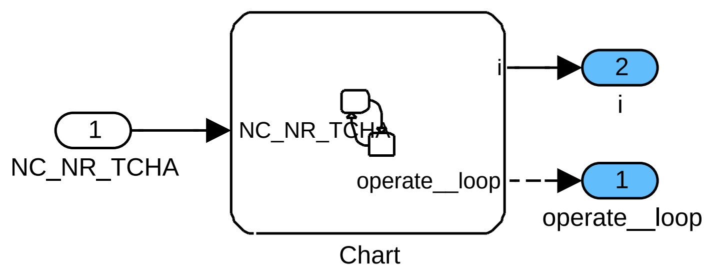


Figure 472 CHRG_MDLADPSNRC0/ operate_10ms/ inlet_branch_counter

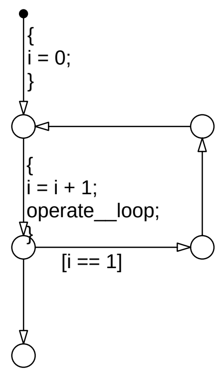


Figure 473 CHRG_MDLADPSNRC0/ operate_10ms/ inlet_branch_counter/ Chart

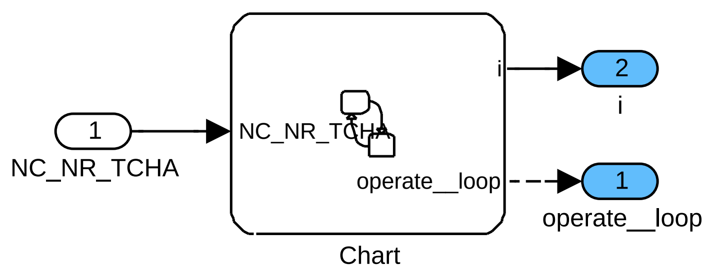


Figure 474 CHRG_MDLADPSNRC0/ operate_10ms/ inlet_branch_counter1

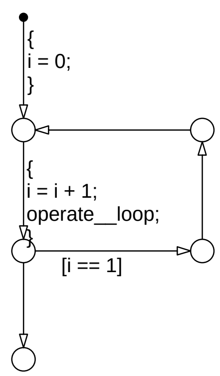


Figure 475 CHRG_MDLADPSNRC0/ operate_10ms/ inlet_branch_counter1/ Chart

4.97 Recirculation actuator position control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CMD_RCL_OPEN	O/V	0...1H	0...1	1	-
Digital control signal for opening of switching recirculation actuator (recirculation valve)					
PWM_RCL	O	0...FFFFH	0...99.998	0.00152587	%
Control signal for opening of continous recirculation actuator (recirculation flap or proportional valve)					

Input data:

PSN_RCL_SP	LC_TCHA_CONF		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PSN_RCL_THD_DIG_CTL	1	0...FFFFH	0...99.998474	0.00152588	%
Threshold for digital recirculation actuator position control					
LC_CMD_RCL_OPEN_MAN	1	0...1H	0...1	1	-
Digital control signal for recirculation valve given by application system					
LC_CMD_RCL_OPEN_MAN_ACT	1	0...1H	0...1	1	-
Activation of digital control signal for recirculation valve given by application system					

4.97.1 CHRG_ACCTLRCAC0

The present specification describes the simplest possible switching control of the recirculation actuator position (NC_PSN_CTL = 1). If the position setpoint PSN_RCL_SP is greater than a threshold C_PSN_RCL_THD_DIG_CTL then LV_CMD_RCL_OPEN is set to 1 to completely open the actuator. Otherwise LV_CMD_RCL_OPEN is set to 0 to completely close the actuator.

More sophisticated controllers can be specified as versions of this specification. In the present version PWM_RCL is a dummy output to have a common output interface with these other specification versions which are designed to control a continuous recirculation actuator (flap or proportional valve).

Function Description

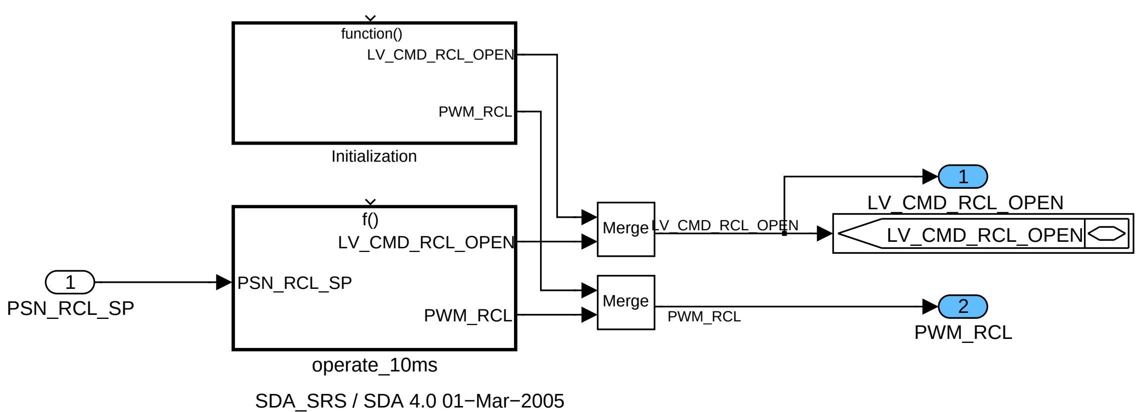


Figure 476 CHRG_ACCTLRCAC0

4.97.1.1 SUBFUNCTION: Initialization

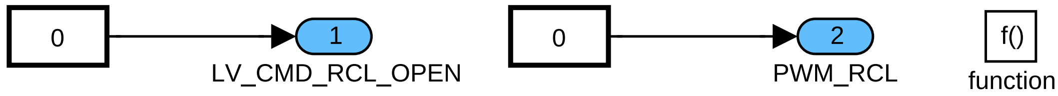


Figure 477 CHRG_ACCTLRCAC0/ Initialization

4.97.1.2 SUBFUNCTION: operate_10ms

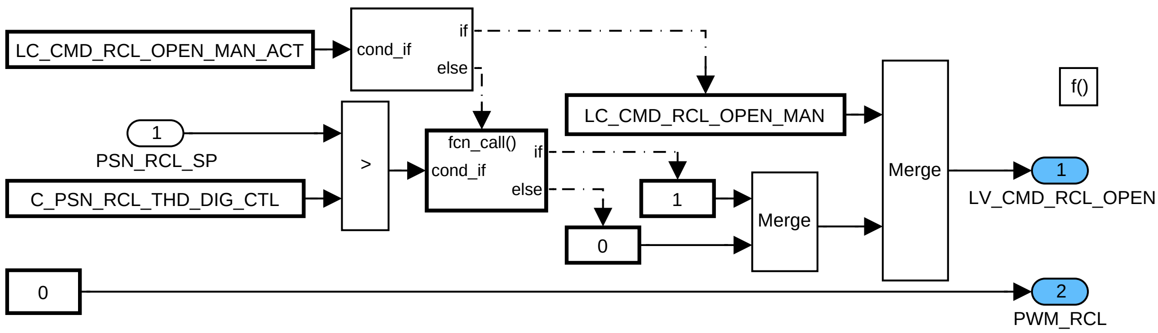


Figure 478 CHRG_ACCTLRCAC0/ operate_10ms

4.98 Flow Turbine Model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_TUR_COR	O/V	0...FFFFH	0...4E+5	6.1036087 6	rpm
Turbine speed related to exhaust gas temperature upstream turbine					
FLOW_TUR	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Turbine flow					
FAC_POW_TUR	O/V	0...FFFFH	0...3.99993896	6.10352E-5	W/(hPa*sqrt(K)
factor for calculation of turbine power					
POW_TUR	O/V	0...FFFFH	0...65.535	0.001	kW
turbine power delivered to charger					
FLOW_TUR_COR	V	0...FFFFH	0...7.9999	1.22071E-4	kg*K^-0.5 /s*bar
sum of slope and offset					
PQ_TUR	V	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure quotient at turbine (PRS_EX / PRS_EX_PCAT_UP)					

Input data:

N_TCHA	TEG_TUR_UP_ABS_SQRT	PQ_EX	PRS_EX
FLOW_WG	FLOW_ENG_TCHA	PRS_EX_PCAT_UP	LC_TCHA_CONF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEG_TUR_UP_ABS_REF_SQRT	1	0...FFFFH	0...45.254143	6.90534E-4	K^0.5
Square root of the reference exhaust gas temperature upstream turbine (constant)					
IP_FAC_POW_TUR	8x10	0...FFFFH	0...3.99993896	6.10352E-5	W/(hPa*sqrt(K)
LDPM_N_TUR_COR_1_CHRG	8	0...FFFFH	0...4E+5	6.1036087 6	rpm
LDPM_PQ_TUR_1_CHRG	10	0...FFFFH	0...15.9997559	2.44141E-4	-
Factor for calculation of turbine power					
IP_FLOW_TUR_COR	8x10	0...FFFFH	0...7.9999	1.22071E-4	kg*K^-0.5 /s*bar
LDPM_N_TUR_COR_1_CHRG	8	0...FFFFH	0...4E+5	6.1036087 6	rpm
LDPM_PQ_TUR_1_CHRG	10	0...FFFFH	0...15.9997559	2.44141E-4	-
Standard turbine flow : FLOW_TUR * TEG_TUR_UP_SQRT / PRS_EX					

4.98.1 CHRG_MDLADTUR0

The turbine model calculates the exhaust gas-mass flow through the turbine (internal calculations are done in kg/h for flow) and the turbine power delivered to the charger. All mechanical losses are calculated in the turbine model, even the mechanical losses of the charger.

The basic module for the turbine flow is a standard turbine map.

To make the map calculable it is necessary, to calculate PQ_TUR. The break points I is the turbine rotation speed and the pressure quotient at the turbine. The output FLOW_TUR_STND is multiplied by the last valid exhaust pressure PRS_EX and divided by the square root of exhaust gas temperature upstream turbine TEG_TUR_UP_ABS_SQRT.

Function Description

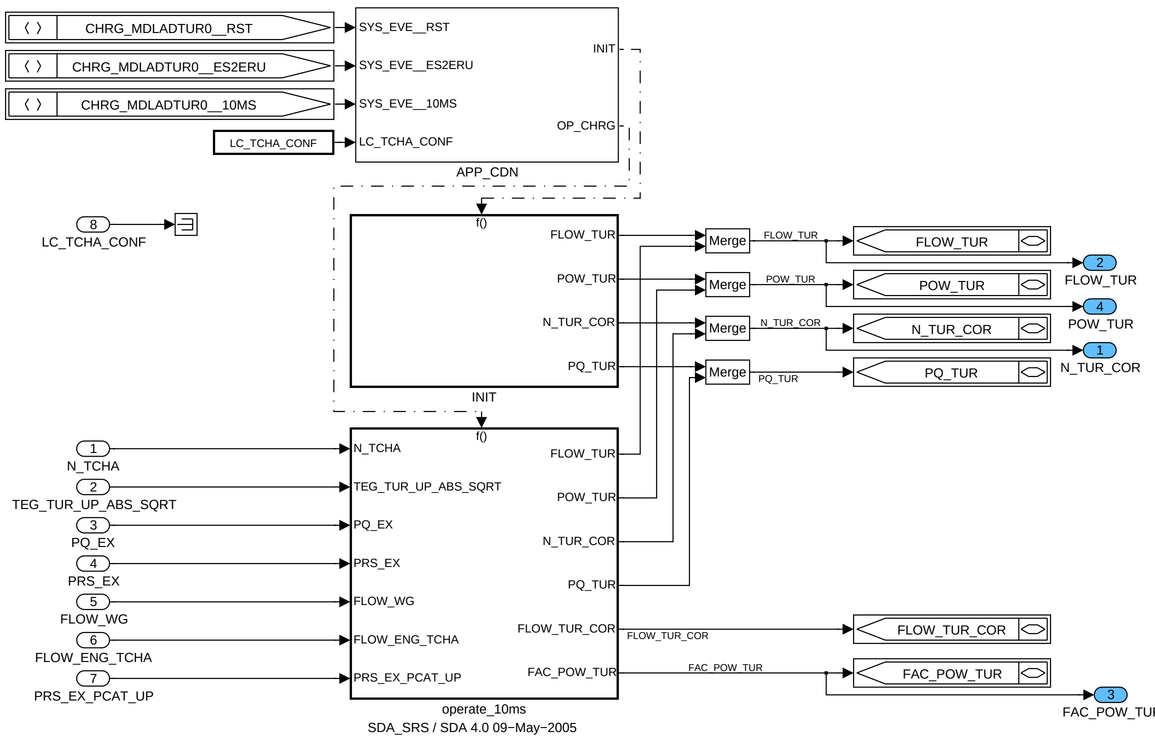


Figure 479 CHRG_MDLADTUR0

4.98.1.1 SUBFUNCTION: INIT

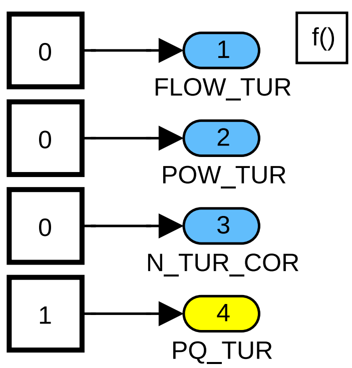


Figure 480 CHRG_MDLADTUR0/ INIT

4.98.1.2 SUBFUNCTION: operate_10ms

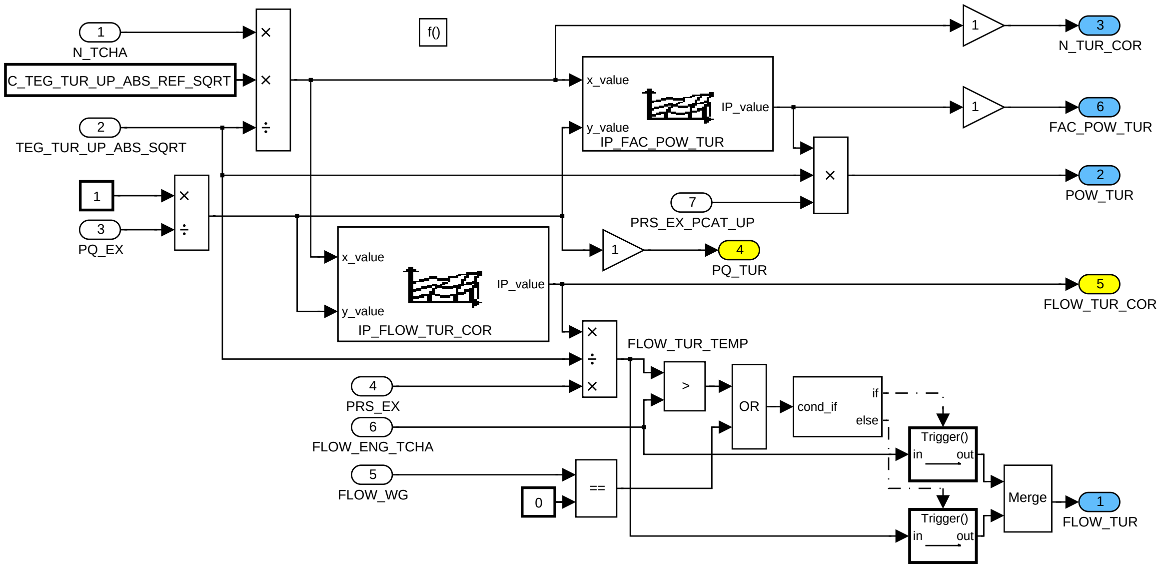


Figure 481 CHRG_MDLADTUR0/ operate_10ms

4.99 Wastegate Model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FLOW_WG_CON_1	O/V	0...FFFFH	0...0.0206019	3.14365E-7	s/m
sqrt(2x/((x-1)*C_RA_EX_VOL_EX* TEG_TUR_UP_ABS)) with x_ex = kappa_ex = 1.35					
FLOW_WG	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Waste gate flow					
PSI_WG	V	0...FFFFH	0...1.99996948	3.05176E-5	-
waste gate flow rate					
TQ_WG_PRS_EX	V	0...7FFFH	0...15.999756	4.88289E-4	Nm
torque caused by exhaust gas pressure forces at waste gate					

Input data:

TEG_TUR_UP_ABS	PQ_EX	FLOW_ENG_TCHA	FLOW_TUR
AR_RED_WG	PRS_EX	PSN_WG	PRS_EX_PCAT_UP
LC_TCHA_CONF	IP_FAC_TQ_WG_PRS_EX_SP		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_FLOW_WG_DIF	1	0...1H	0...1	1	-
Switch whether flow wastegate as flow difference					
IP_FLOW_WG_CON_1	8	0...FFFFH	0...0.02060191	3.14365E-7	s/m
LDPM_TEG_TUR_UP_ABS_CHRG_1	8	0...FFE0H	0...2.047E+3	0.03125	K
Exhaust gas temperature influence on exhaust gas function at the waste gate					
IP_PSI_WG	16	0...FFFFH	0...1.99996948	3.05176E-5	-
LDPM_PQ_EX_PSI_WG	16	0...FFFFH	0...0.99998474	1.52588E-5	-
PSI coefficient for waste gate flow					

4.99.1 CHRG_MDLADFLOWG0

The flow at the waste gate can be modelled und calculated similar to that at a throttle. The throttling will be taken into account with the reduced area at the waste gate The inputs are the pressure quotient PQ_EX (ambient pressure AMP divided by exhaust pressure PRS_EX), the exhaust pressure (PRS_EX), the reduced area of the waste gate (AR_RED_WG) and the absolute exhaust temperature.

The term of the flow calculation at the waste gate depending on the exhaust gas temperature is stored in the map IP_FLOW_WG _CON_1.

This module calculates also the torque at the waste gate caused by the exhaust gas pressure difference at the waste gate.

This calculation is based on a product of the reduced opening area of the wastegate, a factor depending on the exhaust gas constant, the adiabatic coefficient and the absolute exhaust temperature, the “Psi waste gate function (PSI_WG)” and the exhaust pressure.

Function Description

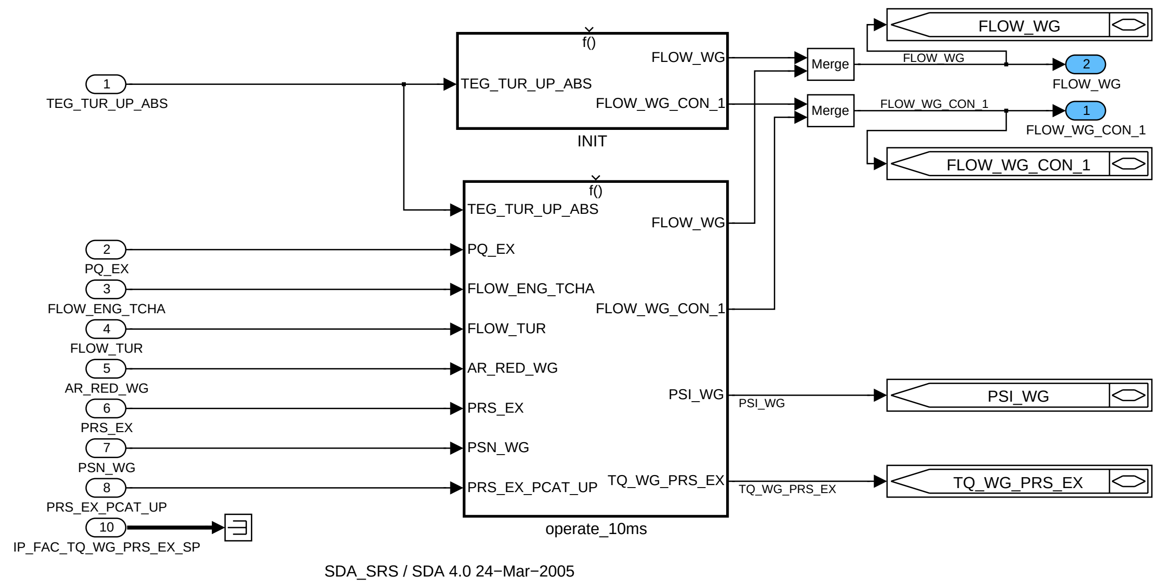


Figure 482 CHRG_MDLADFLOWG0

4.99.1.1 SUBFUNCTION: INIT

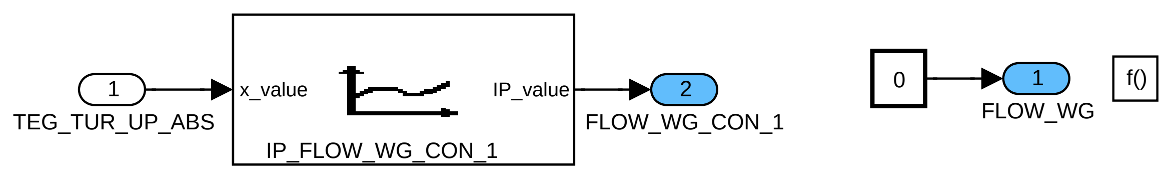


Figure 483 CHRG_MDLADFLOWG0/ INIT

4.99.1.2 SUBFUNCTION: operate_10ms

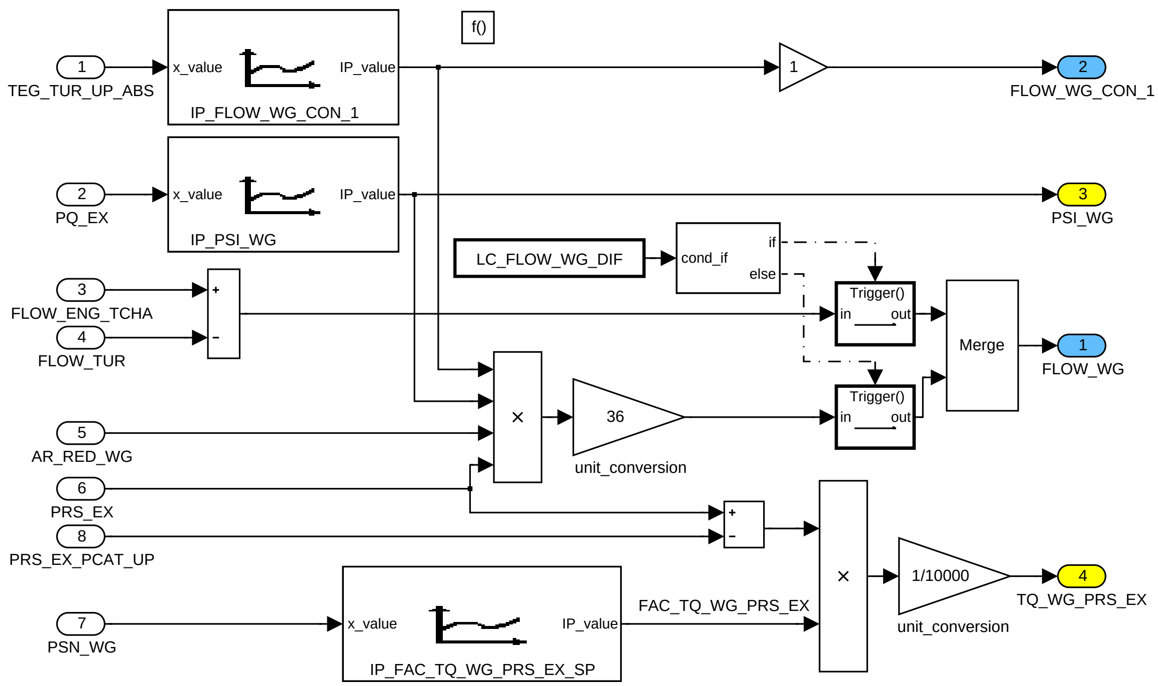


Figure 484 CHRG_MDLADFLOWG0/ operate_10ms

4.100 CAN interface

4.100.1 General information

The communication between the engine management system and the TCS control unit as well as the transmission control unit is done via the CAN bus.

This function description refers only to the torque management related CAN data. For a detailed description of the CAN functions please refer to the separate CAN specification.

To clearly distinguish between variables which are used for EMS internal calculation and variables that are available on the CAN all CAN-related variables are marked with the extension „*_CAN*“ at their name. However, in this chapter only the EMS internal variable names are used.

Recurrence : see CAN specification.

Reference : latest CAN specification version from HMC

4.100.2 CCP CAN Variables

CAN ID's:

Message Name	ID
CCP Receive	06A0H
CCP Transmit	06A1H
DAQ Segment	06A2H
DAQ Time	06A3H
DAQ Display	06A4H

Size multiplied by 7 is the number of bytes that can be displayed (e.g. 15*7 = 105 Bytes @ 10ms synchronous)

Variable Name	Value	Bytes
NC_ODT_SYNC_SIZE	10	70
NC_ODT_TIME_SIZE	10	70
NC_ODT_DISP_SIZE	20	140

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ODT_SYNC_SIZE	1	0 ... FFH	0 ... 255	1	-
size of synchronous ODT table					
NC_ODT_TIME_SIZE	1	0 ... FFH	0 ... 255	1	-
size of time (10ms) ODT table					
NC_ODT_DISP_SIZE	1	0 ... FFH	0 ... 255	1	-
size of display list table					

4.100.3 CAN variables

4.100.3.1 CAN variables from TCU1 Message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TAR_GC_CAN	-	0...7H	0...7	1	-
Target of gear change					
SWI_GS_CAN	V/O	0...1H	0...1	1	-
Gear shift active					
F_OBD	V	0...1H	0...1	1	-
OBD-relvant error in TCU					
TCU_STAT	V	0...1H	0...1	1	-
Status TCU					
SWI_CC_CAN	O	0...03H	0...3	1	-
Status for converter lockup clutch					
G_SEL_DISP_CAN	O	00...0FH	0...15	1	-
Gear selector display					
F_TCU	V/O	0…03H	0…3	1	-
TCU fault					
TCU_TYPE	O	0...03H	0...3	1	
Control unit type					
TCU_OBD_CAN	-	0...FH	0...15	1	-
OBD status of TCU					
TQI_TCU_CAN	V/O	0...FFH	0...99,61	0,39	%
TCU requested engine torque					
TEMP_AT_CAN	-	0...FEH	-40...214	1	°C
A/T fluid temperature					
N_TC	O	0...FFFFH	0...16383,75	0,25	Rpm
A/T Torque converter turbine speed – Filtered value					
TQI_TCU_MSR_CAN	V/O	0...FFH	0...99,61	0,39	%
TCU requested engine torque					
LV_TCU1_CAN_VLD	V/O	0...1H	0...1	1	-
Flag for detected valid TCU1 message					

Message Structure:

Message: TCU1	Identifier: 043FH

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
TAR_GC_CAN	Target of gear change	0	3	00H	-
SWI_GS_CAN	Gearchange active	3	1	00H	-
F_OBD	OBD-relevant error in TCU	4	1	00H	-
TCU_STAT	Status TCU	5	1	00H	-
SWI_CC_CAN	Converter lockup clutch	6	2	00H	-
G_SEL_DISP_CAN	Gear selector display	8	4	00H	0FH
F_TCU	TCU fault	12	2	00H	-
TCU_TYPE	Control unit type	14	2	00H	-
TCU_OBD_CAN	OBD status, transmissionn control	16	4	00H	-
GEAR_TYPE	Number of gear steps of A/T (not used forEMS)	20	4	00H	-
TQI_TCU_CAN	Torque intervention of TCU, referred to the indicated torque	24	8	FFH	-
TEMP_AT_CAN	A/T fluid temperature	32	8	FFH	FFH

N_TC	Torque converter turbine speed (low Byte)	40	8	00H	FEH
	Torque converter turbine speed (high Byte)	48	8	00H	FFH
TQI_TCU_MSR_CAN	TCU requested engine RPM increase	56	8	00H	-

After receiving a valid TCU1 message the flag LV_TCU1_CAN_VLD is set to 1. If this flag is 0 it indicates that the corrsponding message content contains default values.

4.100.3.2 Can variables from TCU2 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
BRAKE_ACT_TCU_CAN	O	0...03H	0...3	1	
Indication of brake switch on/off by TCU					
SPK_RTD_TCU_CAN	-	17H...67H	-15...+15	0.375	°CRK
Torque intervention during gear shift from TCU					
N_TC_RAW	O	0...FFFFH	0...16383,75	0,25	Rpm
Unfiltered Torque converter turbine speed					
LV_TCU2_CAN_VLD	O	0...1H	0...1	1	-
Flag for detected valid TCU2 message					
N_INC_TCU	O	0...1H	0...1	1	-
Engine speed increasing requirement flag					
ETL_TCU_CAN	V/O	0...FFH	0...510	2	Nm
Torque reduction requested for gear shift from TCU(slow path)					
VS_TCU	V	0...FFH	0...255	1	km/h
Vehicle speed calculated by TCU					

Message Structure:

Message: TCU2	Identifier: 0440H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
ETL_TCU_CAN	TCU requests engine torque limit	0	8	FFH	-
TQI_TCU_J	Torque intervention of TCU (not used for EMS)	8	8	FFH	-
VS_TCU	Vehicle speed calculated by TCU	16	8	00H	FFH
FAN_CTRL_TCU	Cooling fan control request by TCU (not used for EMS)	24	2	00H	-
BRAKE_ACT_TCU_CAN	Indication of brake switch on/off signal by TCU	26	2	01H	03H
FUEL_CUT_TCU	Fuel cut request during gear shift (not used for EMS)	28	1	00H	-
INH_FUEL_CUT	Inhibition of engine fuel cut off (not used for EMS)	29	1	00H	-
IDLE_UP_TCU	TCU requests engine idle RPM up (not used for EMS)	30	1	00H	-
N_INC_TCU	Engine speed increasing requirement flag	31	1	00H	-

SPK_RTD_TCU_CAN	Requested spark retard angle from TCU	32	8	FFH	-
N_TC_RAW	Unfiltered Torque converter turbine speed - low byte	40	8	00H	FFH
N_TC_RAW	Unfiltered Torque converter turbine speed - high byte	48	8	00H	FFH
VS_TCU_DECIMAL	The value below decimal point of vehicle speed (not used for EMS)	56	8	00H	-

After receiving a valid TCU2 message the flag LV_TCU2_CAN_VLD is set to 1. If this flag is 0 it indicates that the corrsponding message content contains default values.

4.100.3.3 CAN variables from TCU3 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_TGT_LUP	O	00...FEH	500...3040	10	rpm
Target engine speed used in lockup module					
LV_TCU3_CAN_VLD	O	0...1H	0...1	1	-
Flag for detected valid TCU3 message					

Message Structure:

Message: TCU3	Identifier: 0370H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
N_TGT_LUP	Target engine speed used in lock-up module	0	8	FFH	-
Free	Free	8	56	00H	-

After receiving a valid TCU3 message the flag LV_TCU3_CAN_VLD is set to 1. If this flag is 0 it indicates that the corrsponding message content contains default values.

4.100.3.4 Can Variables from TCS1 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCS_REQ_CAN	V/O	0...1H	0...1	1	-
TCS request (CAN variable name : TCS_REQ)					
MSR_C_REQ_CAN	V/O	0...1H	0...1	1	-
TCS request (CAN variable name : MSR_C_REQ)					
TQI_TCS_CAN	V/O	0...FFH	0...99,61	0,39	%
TCS requested engine torque (CAN variable name : TQI_TCS)					
TQI_MSR_CAN	V/O	0...FFH	0...99,61	0,39	%
TCS requested engine torque (CAN variable name : TQI_TCS)					
TQI_SLW_TCS_CAN	V/O	0...FFH	0...99,61	0,39	%
TCS requested engine torque (CAN variable name : TQI_SLW_TCS)					
TCS_PAS	-	0…1H	0…1	1	-
TCS "passive" indication					
TCS_GSC	-	0…1H	0…1	1	-
TCS gear shift characteristic					
ABS_DEF	-	0…1H	0…1	1	-
ABS "defective" indication					
LV_TCS1_CAN_VLD	O	0...1H	0...1	1	-
Flag for detected valid TCS1 message					

Message Structure:

Message: TCS1	Identifier: 0153H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
TCS_REQ_CAN	Request TCS	0	1	00H	-
MSR_C_REQ_CAN	Request for MSR functions	1	1	00H	-
TCS_PAS	TCS disabled by user (not used for EMS)	2	1	00H	-
TCS_GSC	TCS gear shift characteristic (not used for EMS)	3	1	00H	-
Reserved	Brake system definition (ABS/TCS/ESP/EHB) (not used for EMS)	4	2	00H	-
ABS_DIAG	ABS_TCS/ESP "diagnostic mode" indication (not used for EMS)	6	1	00H	-
ABS_DEF	ABS "defective" indication (not used for EMS)	7	1	00H	-
TCS_DEF	TCS "defective" indication (not used for EMS)	8	1	00H	-
TCS_CTL	TCS "control" indication (not used for EMS)	9	1	00H	-
ABS_ACT	ABS "control" indication (not used for EMS)	10	1	00H	-
EBD_DEF	EBD "defective" indication (not used for EMS)	11	1	00H	-
ESP_PAS	ESP disabled by user (not used for EMS)	12	1	00H	-
ESP_DEF	ESP "defective" indication (not used for EMS)	13	1	00H	-
ESP_CTL	ESP "control" indication (not used for EMS)	14	1	00H	-
TCS_MFRN	TCS Manufacturer Information (not used for EMS)	15	1	00H	-
Free	free	16	8	00H	-
TQI_TCS_CAN	Torque intervention for TCS functions,	24	8	FFH	-

	referred to the indicated engine torque				
TQI_MSR_CAN	Torque intervention for MSR functions, referred to the indicated engine torque	32	8	00H	-
TQI_SLW_TCS_CAN	Slow torque intervention for TCS functiuons, referred to the indicated engine torque	40	8	FFH	-
Free	Free	48	16	00H	-

After receiving a valid TCS1 message the flag LV_TCS1_CAN_VLD is set to 1. If this flag is 0 it indicates that the corrsponding message content contains default values.

4.100.3.5 Can Variables from TCS2 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TCS2_CAN_VLD	O	0...1H	0...1	1	-
Flag for detected valid TCS2 message					

Message Structure:

Message: TCS2	Identifier: 01F0H

Output period: 20ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
SA_COUNT	Streering Angle Count Value	0	16	4000H	FFFFH
SA_Z_COUNT	Streering Angle Count Value at Zero point	16	15	0000H	-
SA_Z_FLAG	Flag which indicates Zero point	31	1	00H	-
Free	Free	32	32	00H	-

After receiving a valid TCS2 message the flag LV_TCS2_CAN_VLD is set to 1. If this flag is 0 it indicates that the corrsponding message content contains default values.

4.100.3.6 Can Messages from FATC

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CR_Fatc_TqAcnOut	O	0 ... F9H	0 … 50.4	0.2	Nm
Aircon torque loss (CAN Value)					
CR_Fatc_AcnCltEnRq	O	0 ... 1H	0 … 1	1	
A/C(COMP) Clutch enable request (CAN Value)					
CF_Fatc_AcnRqSwi	O	0 ... 1H	0 … 1	1	
A/C selected (CAN Value)					
CF_Fatc_BlwrMax	O	0 ... 1H	0 … 1	1	
FATC Max Blower State (CAN Value)					

Message Structure:

Message: FATC	Identifier: 0350H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
CR_Fatc_TqAcnOut	Calculated A/C COMP torque	0	8	00H	
CF_Fatc_AcnRqSwi	A/C selected	8	1	00H	
CR_Fatc_AcnCltEnRq	A/C(COMP) Clutch enable request	9	1	00H	
CF_Fatc_BlwrMax	FATC Max Blower State	14	1	00H	

General information:

The variables defined above come on the CAN. Therefore, they are inputs for the EMS. Here are listed only the ones needed for the spec interface. More informations are available on the CAN. For more informations, please refer to the CAN spec.

4.100.3.7 Can Messages from FATC2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_OUT_CAN	V/O	0...C8H	-40...60	0.5	°C
Outside temperature (CAN Value)					

Message Structure:

Message: FATC2	Identifier: 0652H

Output period: 100ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
	Not used	0	1		
		...			

	Not used	48	8		
CR_Chs_OutTemp_C	Outside Temperature	56	8	00H	FDH

General information:

The variables defined above come on the CAN, from the TCU or the TCS. Therefore, they are inputs for the EMS. Here are listed only the ones needed for the spec interface. More informations are available on the CAN. For more informations, please refer to the CAN spec.

4.100.4 EMS Input Signals from the CAN

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_ASR_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque reduction requested for ASR from TCS control unit					
TQI_GS_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque reduction requested for gear shift from TCU					
TQI_GS_INC_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque increment requested for gear shift from TCU					
LV_ASR_REQ	V/O	0...1H	0...1	1	-
Boolean for ASR intervention requested by the TCS control unit					
LV_GS_ACT	V/O	0...1H	0...1	1	-
Boolean for GS intervention requested by the transmission control unit					
TOIL_GB	V/O	0...FEH	-40...214	1	°C
A/T fluid temperature					
LV_ERR_TOIL_GB_ETCU	V/O	0...1H	0...1	1	-
Error on TOIL_GB signalled by TCU					
TCU_OBD	V/O	0...FFH	0...255	1	-
OBD status of TCU					
TAR_GC	V/O	0...7H	0...7	1	-
Target of gear change					
G_SEL_DISP	V/O	00...0FH	0...15	1	-
Gear selector display					
GR_AT_INTER	O	0 ... FFH	0...255	1	-
Automatic transmission actual gear ratio – intermediate variable before GEAR calculation					
N_TUR_CONV	V/O	8000...7FFFH	-32768...32767	1	Rpm
Torque converter turbine speed – Filtered value					
N_TUR_CONV_RAW	V/O	8000...7FFFH	-32768...32767	1	Rpm
Torque converter turbine speed – Raw value					
LV_ERR_N_TUR_CONV_ETCU	V/O	0...1H	0...1	1	-
Error on N_TUR_CONV signalled by TCU					
TQI_ASR_SLW_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque reduction requested for ASR from TCS control unit – Slow path					
TQI_MSR_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque reduction requested for MSR from TCS control unit					
LV_MSR_REQ	V/O	0...1H	0...1	1	-
Boolean for MSR intervention requested by the TCS control unit					
LV_BRAKE_OFF	V/O	0...1H	0...1	1	-
Boolean for Brake switch switch off by the TCU control unit					
SPK_RTD_TCU	V/O	17H...67H	-15...+15	0.375	°CRK
Torque intervention during gear shift from TCU					
SWI_CC	V/O	0...03H	0...3	1	-
Status for converter lockup clutch					
LV_MIL_ACT_REQ	V/O	0...1H	0...1	1	-
Flag for MIL on request via CAN					
TAM_CAN	V/O	0...FEH	-48...142.5	0.75	°C
Ambient Air temperature (from CAN message)					
STATE_ERR_TAM_CAN	V/O	0H 1H 2H 3H	NO_SYM NOT_VLD SC OC	1	-
Error state of TAM sensor attached to FATC control unit					
LV_TAM_CAN_FIRST_VLD	V/O	0...1H	0...1	1	-
Indicates that the first valid TAM value (0...C8H) has been recieved from CAN					
LV_N_INC_TCU_REQ	V/O	0...1H	0...1	1	-
Engine speed increasing request flag from TCU					

N_SP_IS_TCU	V/O	0...1FE0H	0...8160	1	rpm
Target engine speed used in lock-up module					
TQI_GS_SLOW_REQ	V/O	0...7FFFH	0...1023.97	0,03125	Nm
Torque reduction requested for gear shift from TCU					
LV_ACC_FATC_CAN_FIRST_VLD	V/O	0...1H	0...1	1	-
Indicate that the first valid torque loss value or message(00 … FFH) has been received from CAN					
TQ_LOSS_ACC_CAN	V/O	8000 ... 0 H	-1024 … 0	0.03125	Nm
Air condition torque loss (CAN Value)					
LV_IM_ACCIN_CAN	V/O	0...1H	0...1	1	-
Air condition requested vis CAN					
LV_IM_ACIN_CAN	V/O	0...1H	0...1	1	-
Air selected via CAN					
LV_FATC_BLWR_MAX_CAN	V/O	0...1H	0...1	1	-
FATC Max Blower State via CAN					
VS_TCU_CAN	V/O	0...FFH	0...255	1	km/h
Vehicle speed calculated by TCU via CAN					

Input data:

TQI_TCS_CAN	TQI_TCU_CAN	TCS_REQ_CAN	SWI_GS_CAN
TEMP_AT_CAN	TCU_OBD_CAN	TAR_GC_CAN	N_TC
TQI_SLW_TCS_CAN	TQI_MSR_CAN	MSR_C_REQ_CAN	TQ_STND
LV_ASR_REQ	BRAKE_ACT_TCU_CAN	TQI_TCU_MSR_CAN	LV_MSR_REQ
SPK_RTD_TCU_CAN	IGA_BAS_COR[NC_CBK_IN_NR]	SWI_CC_CAN	N_TC_RAW
G_SEL_DISP_CAN	TEMP_OUT_CAN	N_INC_TCU	N_TGT_LUP
ETL_TCU_CAN	CR_FATC_TQACNOUT	CF_FATC_ACNCLTENRQ	CF_FATC_ACNRQSWI
VS_TCU	CF_FATC_BLWRMAX		

General information:

This paragraph shows some informations coming from the CAN (see paragraph above) and imported into EMS variables. For more informations, please refer to the CAN specification.

Formula section:

Input as CAN data (from the CAN)	Comment – Formula	Input to EMS internal function
TQI_TCS_CAN [%]	Torque reduction request for ASR from TCS unit TQI_ASR_REQ = (TQI_TCS_CAN*TQ_STND) / 100% *(this equation is valid with physical values only)*	**TQI_ASR_REQ [Nm]**
TQI_SLW_TCS_CAN [%]	Torque reduction request for ESP from TCS unit TQI_ASR_SLW_REQ = (TQI_SLW_TCS_CAN*TQ_STND) / 100% *(this equation is valid with physical values only)*	**TQI_ASR_SLW_REQ [Nm]**
TQI_MSR_CAN [%]	Torque reduction request for MSR from TCS unit TQI_MSR_REQ = (TQI_MSR_CAN*TQ_STND) / 100% *(this equation is valid with physical values only)*	**TQI_MSR_REQ [Nm]**
ETL_TCU_CAN [Nm]	Torque reduction request for Gear-Shift from TCU TQI_GS_SLOW_REQ=	**TQI_GS_SLOW_REQ [Nm]**

	(ETL_TCU_CAN)h * 64	
TQI_TCU_CAN [%]	Torque reduction request for Gear-Shift from TCU If SPK_RTD_TCU <> FFh and SPK_RTD_TCU < IGA_BAS_COR[NC_CBK_IN_NR] Then TQI_GS_REQ = 0 Else TQI_GS_REQ=(TQI_TCU_CAN*TQ_STND)/100% *(this equation is valid with physical values only)*	**TQI_GS_REQ [Nm]**
TQI_TCU_MSR_CAN	Torque increment request for Gear-Shift from TCU TQI_GS_INC_REQ =(TQI_TCU_MSR_CAN*TQ_STND) /100% *(this equation is valid with physical values only)*	**TQI_GS_INC_REQ**
*Note : the multiplications above must be done with the **physical** values only, because the hexadecimal limits are not the same for TQ_STND and TQ_xx_REQ.*		
TCS_REQ_CAN	Status-Bit = High, if ASR is requested	**LV_ASR_REQ**
MSR_C_REQ_CAN	Status-Bit = High, if MSR is requested	**LV_MSR_REQ**
SWI_GS_CAN	Status-Bit = High, if GS is requested	**LV_GS_ACT**
F_TCU	F_TCU = F_TCU	**F_TCU**
TEMP_AT_CAN	TOIL_GB = TEMP_AT_CAN	**TOIL_GB**
TEMP_AT_CAN	**If** TEMP_AT_CAN = FFH **then** LV_ERR_TOIL_GB_ETCU = 1 **else** LV_ERR_TOIL_GB_ETCU = 0	**LV_ERR_TOIL_GB _ETCU**
TCU_OBD_CAN	Bit 3 : MIL blinking request Bit 2 : MIL on request Bit 1 : freeze frame request Bit 0 : readiness info TCU_OBD = TCU_OBD_CAN	**TCU_OBD**
TAR_GC_CAN	TAR_GC = TAR_GC_CAN	**TAR_GC**
TAR_GC_CAN	*TAR_GC* *gear* *GR_AT_INTER* 000 Neutral / Parking 0 001 1st gear 1 010 2nd gear 2 011 3rd gear 3 100 4th gear 4 101 (5th gear) 5 110 (6th gear) 6 111 Reverse gear 7	**GR_AT_INTER**
G_SEL_DISP_CAN	*G_SEL_DISP_CAN* *Function* *G_SEL_DISP* 000 P 0 001 L 1 010 2 2 011 3 3 100 Not used 4 101 D 5 110 N 6 111 R 7 1000 sports mode / manual shift 8	**G_SEL_DISP**

	1111 Fault 15	
N_TC	N_TUR_CONV = N_TC (physically) with different resolution N_TC : 0,25 rpm N_TUR_CONV : 1 rpm	**N_TUR_CONV**
N_TC	**If** N_TC = FFFFH **then** LV_ERR_N_TUR_CONV_ETCU = 1 **else** LV_ERR_N_TUR_CONV_ETCU = 0	**LV_ERR_N_TUR_CONV_ETCU**
N_TC_RAW	N_TUR_CONV_RAW = N_TC_RAW (physically) with different resolution N_TC_RAW : 0,25 rpm N_TUR_CONV_RAW : 1 rpm	**N_TUR_CONV_RAW**
BRAKE_ACT_TCU_CAN	**If** BRAKE_ACT_TCU_CAN = 01H **then** LV_BRAKE_OFF = 1 **else** LV_BRAKE_OFF = 0 BRAKE_ACT_TCU_CAN = 00H : TMS does not support this function. = 01H : Brake switch is not pressed (OFF) = 02H : Brake switch is pressed (ON), = 03H : Brake switch failure.	**LV_BRAKE_OFF**
SPK_RTD_TCU_CAN	SPK_RTD_TCU = SPK_RTD_TCU_CAN	**SPK_RTD_TCU**
SWI_CC_CAN	SWI_CC = SWI_CC_CAN SWI_CC_CAN = 00H : No lock up control = 01H : Slip lock up = 02H : Fully lock up = 03 H : Slip lock up off	**SWI_CC**
TEMP_OUT_CAN	TAM_CAN = TEMP_OUT_CAN {Equation only valid with physical values} When TEMP_OUT_CAN indicates an error (value (FDH to FFH) the last valid value is output from this interface module.	**TAM_CAN**
TEMP_OUT_CAN	IF TEMP_OUT_CAN = FDH THEN STATE_ERR_TAM_CAN = 1H (NOT_VLD) ELSE IF TEMP_OUT_CAN = FEH THEN STATE_ERR_TAM_CAN = 2H (SC) ELSE IF TEMP_OUT_CAN = FFH THEN STATE_ERR_TAM_CAN = 3H(OC) ELSE STATE_ERR_TAM_CAN = 0H ENDIF ENDIF ENDIF	**STATE_ERR_TAM_CAN**
TEMP_OUT_CAN	LV_TAM_CAN_FIRST_VLD is set to 1 when the first valid TEMP_OUT_CAN (00H to C8H) value has been received via CAN.	**LV_TAM_CAN_FIRST_VLD**
N_INC_TCU	LV_N_INC_TCU_REQ = N_INC_TCU	**LV_N_INC_TCU_REQ**

N_TGT_LUP	N_SP_IS_TCU(dec) = [(N_TGT_LUP(dec)*10dec)]+500dec	**N_SP_IS_TCU**
CR_FATC_TQAC NOUT	LV_ACC_FATC_CAN_FIRST_VLD is set to 1 when the first valid torque loss value or message (00H to FFH) value has been received via CAN.	**LV_ACC_FATC_CAN_FIRST_VLD**
CR_FATC_TQAC NOUT	TQ_LOSS_ACC_CAN = CR_FATC_TQACNOUT (physically in absolute value) with different resolution and sign (TQ_LOSS_ACC_CAN in negative value) TQ_LOSS_ACC_CAN : 0.03125 Nm CR_FATC_TQACNOUT : 0.2 Nm	**TQ_LOSS_ACC_CAN**
CF_FATC_ACNC LTENRQ	LV_IM_ACCIN_CAN = CF_FATC_ACNCLTENRQ	**LV_IM_ACCIN_CAN**
CF_FATC_ACNR QSWI	LV_IM_ACIN_CAN = CF_FATC_ACNRQSWI	**LV_IM_ACIN_CAN**
CF_FATC_BLWR MAX	LV_FATC_BLWR_MAX_CAN = CF_FATC_BLWRMAX	**LV_FATC_BLWR_MAX_CAN**
VS_TCU	VS_TCU_CAN = VS_TCU	**VS_TCU_CAN**

If LC_INH_MSR is set to 1, input signals from TCS for MSR are ignored and, following EMS internal function are initialized as following,

If LC_INH_MSR = 1
Then LV_MSR_REQ = 0
and TQI_MSR_REQ =0 [Nm]

If LC_TQI_MSR_REQ_MAN is set to 1,TQI_MSR_REQ is just set to C_TQI_MSR_REQ_MAN regardless LC_INH_MSR and even there is no TCS unit.

If LC_TQI_MSR_REQ_MAN = 1
then LV_MSR_REQ = 1
and TQI_MSR_REQ = C_TQI_MSR_REQ_MAN
else TQI_MSR_REQ = (TQI_MSR_CAN*TQ_STND) / 100%

If LC_TQI_ASR_REQ_MAN is set to 1,TQI_ASR_REQ & TQI_ASR_SLW_REQ are just set to C_TQI_ASR_REQ_MAN & C_TQI_ASR_SLW_REQ_MAN.

If LC_TQI_ASR_REQ_MAN = 1
then LV_ASR_REQ = 1
and TQI_ASR_REQ = C_TQI_ASR_REQ_MAN
and TQI_ASR_SLW_REQ = C_TQI_ASR_SLW_REQ_MAN
else TQI_ASR_REQ = (TQI_TCS_CAN*TQ_STND) / 100%
and TQI_ASR_SLW_REQ=(TQI_SLW_TCS_CAN*TQ_STND) / 100%

if bit 2 of TCU_OBD = 1

then LV_MIL_ACT_REQ = 1

else LV_MIL_ACT_REQ = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_INH_MSR	1	0...1H	0 ... 1	1	
Inhibition bit to make disable torque request signal from MSR					
LC_TQI_MSR_REQ_MAN	1	0...1H	0 ... 1	1	
Manual MSR torque request enabled					
C_TQI_MSR_REQ_MAN	1	0...7FFFH	0...1023.97	0.03125	Nm
Manual applied Indicated MSR torque request					
LC_TQI_ASR_REQ_MAN	1	0...1H	0 ... 1	1	
Manual ASR torque request enabled					
C_TQI_ASR_REQ_MAN	1	0...7FFFH	0...1023.97	0.03125	Nm
Manual applied Indicated ASR torque request – Fast path					
C_TQI_ASR_SLW_REQ_MAN	1	0...7FFFH	0...1023.97	0.03125	Nm
Manual applied Indicated ASR slow torque request – Slow path					
LC_TCS1_OFF	1	0...1H	0 ... 1	1	-
Inhibition of TCS1 message receiving					

4.100.5 Timeout counter for CAN messages

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOUT_CTR_TCU1	V/O	0…FFH	0…2,55	0.01	s
Timeout counter TCU1 message					
TOUT_CTR_TCU2	V/O	0…FFH	0…2,55	0.01	s
Timeout counter TCU2 message					
TOUT_CTR_TCU3	V/O	0…FFH	0…2,55	0.01	s
Timeout counter TCU3 message					
TOUT_CTR_TCS1	V/O	0…FFH	0…2,55	0.01	s
Timeout counter TCS1 message					
TOUT_CTR_TCS2	V/O	0…FFH	0…5,1	0.02	s
Timeout counter TCS2 message					
TOUT_CTR_FATC2	V/O	0…FFH	0…25,5	0.1	s
Timeout counter FATC2 message					

Input data:

CONF_TCS	CONF_TCU		

If (CONF_TCU = 00H (AT) or CONF_TCU = 0AH (CVT), then a timeout counter TOUT_CTR_TCU1 (for TCU1 message) and if CONF_TCU = 00H (AT) then TOUT_CTR_TCU2 (for TCU2 message) and TOUT_CTR_TCU3 (for TCU3 message) is incremented every 10 ms. If a new valid TCU1/TCU2/TCU3 message is detected, then the related timeout counter TOUT_CTR_TCU1/2/3 is resetted to 0. After reaching NC_TOUT_MAX the values of the related TCU message are set to INIT value.

If TCS configuration is detected (CONF_TCS = 1), then a timeout counter TOUT_CTR_TCS1 (for TCS1 message) is incremented every 10 ms. If a new valid TCS1 message is detected, then the timeout counter TOUT_CTR_TCS1 is resetted to 0. After reaching NC_TOUT_MAX the values of the TCS1 message are set to INIT value.

If TCS configuration is detected (CONF_TCS = 1), then a timeout counter TOUT_CTR_TCS2 (for TCS2 message) is incremented every 20 ms. If a new valid TCS2 message is detected, then the timeout counter TOUT_CTR_TCS2 is resetted to 0. After reaching NC_TOUT_MAX_TCU2 the values of the TCS2 message are set to INIT value.

A timeout counter TOUT_CTR_FATC2 (for FATC2 message) is incremented every 100 ms. If a new valid FATC2 message is detected, then the timeout counter TOUT_CTR_ FATC2 is resetted to 0. After reaching NC_TOUT_MAX the values of the FATC2 message are set to INIT value.

NC_TOUT_MAX = 50

NC_TOUT_MAX_TCU2 = 25

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_TOUT_MAX	-	0...FFH	0...255	1	-
Maximum value for Message time out					
NC_TOUT_MAX_TCU2	-	0...FFH	0...255	1	-
Maximum value for Message time out of TCS2					

4.100.6 EMS Output Signals to the CAN

General information:

This paragraph shows some variables produced by the EMS exported to the CAN. For more informations, please refer to the CAN specification.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_CTR_STE_CAN	-	0...FFFFH	0...65535	10	ms
Counter for CAN power up delay					
LV_STE_CAN_OUT	-	0...1H	0...1	1	-
Flag indicating power up delay for output messages to CAN has passed					
LV_STE_TCU1_CAN_IN	O	0...1H	0...1	1	-
Flag indicating power up delay for input messages from CAN has passed					
EMS1_CAN[8]	-	0...255H	0...255	1	-
8 Byte CAN Frame for EMS1 message					
EMS2_CAN[8]	-	0...255H	0...255	1	-
8 Byte CAN Frame for EMS2 message					
EMS4_CAN[8]	-	0...255H	0...255	1	-
8 Byte CAN Frame for EMS4 message					

Application conditions:

Initialisation: at reset: all values = 0

Recurrence: 10 ms

Description:

After startup the ECU delayes sending EMS messages on CAN for 250ms:

Formula section:

```
If      T_CTR_STE_CAN < FFFFh
Then    Increment T_CTR_STE_CAN by 1
EndIf

If      LV_STE_CAN_OUT = 0
Then    If      T_CTR_STE_CAN >= 25
        Then    LV_STE_CAN_OUT = 1
        EndIf
EndIf

If      LV_STE_TCU1_CAN_IN = 0
        If      T_CTR_STE_CAN >= 25
        Then    LV_STE_TCU1_CAN_IN = 1
        EndIf
EndIf
```

Send CAN messages:

```
If      CCP CAN messages are NOT inhibited via KWP2000 Service
        "DisableEnableNormalMessageTransmission"
Then    If      LV_STE_CAN_OUT = 1
        Then    Fill Can Buffer EMSx_CAN with data
                Transmit EMSx Messages
        EndIf
EndIf
```

4.100.6.1 EMS1 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SWI_IGK_CAN	-	0...1H	0...1	1	-
Status of ignition lock					
LV_ACK_TCS	V	0...1H	0...1	1	-
Acknowledge bit from EMS					
ACK_TCS_CAN	-	0...1H	0...1	1	-
Acknowledge bit from EMS (CAN-value)					
PUC_STAT	-	0...1H	0...1	1	-
Engine in fuel cut-off					
TQ_COR_STAT_CAN	-	0...3H	0...3	1	-
Status of torque intervention from EMS (CAN-value)					
RLY_AC_CAN	-	0...1H	0...1	1	-
Activation of air conditioner compressor relay					
TQI_CAN	V	0...FFH	0...99.61	0.391	%
Indicated engine torque without torque intervention (CAN-value)					
F_SUB_TQI_CAN	-	0...1H	0...1	1	-
Error on load signal					
TQI_ACOR_CAN	V	0...FFH	0...99.61	0.391	%
Indicated engine torque during torque intervention (CAN-value)					
N_CAN	-	0...FFFFH	0... 16383,75	0.25	rpm
Engine speed					
F_N_ENG_CAN	-	0...1H	0...1	1	-
Error on engine speed signal					
TQFR_CAN	V	0...FFH	0...99.61	0.391	%
Engine friction torque (CAN_value)					
VS_CAN	-	0...FFH	0...255	1	km/h
Vehicle speed					
RATIO_TQI_BAS_MAX_STND_CAN	-	0...FFH	0...2	0,0078	-
Torque reduction factor sent to TCU – filtered value					

Input data:

LV_IGK	LV_ACK_TCS	LV_PUC	LV_RLY_ACCOUT
TQI_AV	LV_ERR_MAF	LV_ERR_LOAD_PLAUS	NC_ETC_CONF
N	LV_ERR_CRK	TQ_LOSS	VS
LV_ERR_VS	LV_ERR_MWSS	RATIO_TQI_BAS_MAX_STND	TQI_AV_EMS
LV_ERR_MAP			

Formula section:

EMS name	Comment - Formula	Output to CAN
LV_IGK	SWI_IGK_CAN = LV_IGK	**SWI_IGK_CAN**
LV_ACK_TCS	Monitoring of the message from the TCS control unit. The bit is set, if at least one new message from the TCS control unit has been transmitted within 500 ms and the following condition fulfilled. (LV_ASR_REQ=0 **and** LV_MSR_REQ=0 **and** TQI_TCS_CAN=FFh **and** TQI_MSR_CAN=0) **OR** (LV_ASR_REQ=0 **and** LV_MSR_REQ=1 **and** TQI_TCS_CAN=(bitwise inverted)TQI_MSR_CAN AND TQI_MSR_CAN > 0) **OR** (LV_ASR_REQ=1 **and** LV_MSR_REQ=0 **and** TQI_TCS_CAN<FFh **and** TQI_MSR_CAN=0) If the conditions above are not fulfilled LV_ACK_TCS = 0	**ACK_TCS_CAN**
LV_PUC	PUC_STAT = LV_PUC	**PUC_STAT**
STATE_TQ_INTV	Status of torque intervention. Feedback signal from the engine management system about to which extent a torque reduction is actually performed.	**TQ_COR_STAT_CAN**
LV_RLY_ACCOUT	RLY_AC_CAN = LV_RLY_ACCOUT	**RLY_AC_CAN**
TQI_AV_EMS	Indicated engine torque corresponding to the torque requested from the driver; including knock control, COP and other ignition angle corrections; torque interventions are not included. TQI_CAN = TQI_AV_EMS / TQ_STND * 100%	**TQI_CAN**
LV_ERR_MAF, LV_ERR_LOAD_PLAUS, LV_ERR_MAP	In case of MAF configuration : **If** LV_ERR_MAF = 1 **or** LV_ERR_LOAD_PLAUS = 1 **or** LV_ERR_MAP = 1 **then** F_SUB_TQI_CAN = 1	**F_SUB_TQI_CAN**
TQI_AV	Indicated engine torque during torqe intervention. TQI_ACOR_CAN = TQI_AV / TQ_STND * 100%	**TQI_ACOR_CAN**
N	N_CAN = N	**N_CAN**
LV_ERR_CRK	F_N_ENG_CAN = LV_ERR_CRK	**F_N_ENG_CAN**
TQ_LOSS	Engine friction torque including temperature and A/C correction TQFR_CAN = - TQ_LOSS / TQ_STND * 100%	**TQFR_CAN**
VS	**If** LV_ERR_VS = 1 **Or** LV_ERR_MWSS = 1 **Then** VS_CAN = FFh **Else** VS_CAN = VS	**VS_CAN**
RATIO_TQI_BAS_MAX_STND	RATIO_TQI_BAS_MAX_STND = RATIO_TQI_BAS_MAX_STND	**RATIO_TQI_BAS_MAX_STND_CAN**

Message Structure:

Message: EMS1	Identifier: 0316H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
SWI_IGK_CAN	Terminal 15 –KEY ON	0	1	00H	-
F_N_ENG_CAN	Error – engine speed signal	1	1	00H	-
ACK_TCS_CAN	Acknowledement TCS	2	1	00H	-
PUC_STAT	Engine in fuel cut off	3	1	00H	-
TQ_COR_STAT_CAN	Status, torque intervention	4	2	00H	-
RLY_AC_CAN	Activation, air conditioner compressor relay	6	1	00H	-
F_SUB_TQI_CAN	Error on torque measure or calculation	7	1	00H	-
TQI_ACOR_CAN	Indicated engine torque after torque interventions	8	8	00H	-
N_CAN	Engine speed (low Byte)	16	8	00H	-
	Engine speed (high Byte)	24	8	00H	-
TQI_CAN	Indicated engine torque	32	8	00H	-
TQFR_CAN	Frictional torque	40	8	00H	-
VS_CAN	Vehicle speed	48	8	00H	FFH
RATIO_TQI_BAS_ MAX_STND_CAN	Standard torque ratio	56	8	00H	-

4.100.6.2 EMS2 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAN_VERS_CAN	-	0...3FH	0...630	10	Nm
CAN version					
CONF_TCU_CAN	V	0...FH	0...15	1	-
A/T - M/T information					
OBD_FRF_ACK_CAN	-	0...3FH	0...630	10	Nm
OBD freeze frame					
TQ_STND_CAN	-	0...3FH	0...630	10	Nm
standard torque to which all torque reduction requests are related (CAN-value)					
TEMP_ENG_CAN	-	0...FFH	-48...142,5	0,75	°C
Engine coolant temperature					
MAF_FAC_ALTI_MMV_CAN	-	0...FFH	0...1.992	7.81 E-3	-
Mass air flow correction factor for altitude.					
VB_OFF_ACT_CAN	-	0...1H	0...1	1	-
Detection of battery disconnection					
ACK_ES_CAN	-	0...1H	0...1	1	-
Acknowledgement „engine stopped“					
CONF_MIL_FMY_CAN	-	0...7H	0...7	1	-
Configuration of MIL Handling and Failure Memory Management by ECU					
TPS_CAN	V/O	20...F5H	0...100	0,444	%
Relative throttle angle distributed via CAN					
ENG_CHR_CAN	-	0...FH	0...15	1	-
Engine Characteristic – Kind of fuel, ETS					
ENG_VOL_CAN	-	0...FFH	0...25,5	0,1	L
Engine Displacement					
BRAKE_ACT_CAN	V	0...03H	0...3	1	-
Indication of brake switch ON/OFF					
OD_OFF_REQ	-	0...1H	0...1	1	-
Over drive off request to TCU					
ACC_ACT	V	0...1H	0...1	1	-
Auto cruise control active					

Input data:

CONF_TCU	TCO	LV_ERR_TCO	MAF_FAC_ALTI_MMV
LV_VB_OFF	LV_ES	CONF_MIL_FMY	TPS
LV_ERR_TPS	LV_ERR_TPS_PLAUS	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK
ENG_CHR	ENG_VOL	LV_ERR_BLS_BTS	LV_ERR_MAF
LV_ERR_RATIO_CHK	LV_BLS	LV_ERR_MAP	LV_CRU_ACT
FAC_TQ_REQ_DRIV	FAC_TQ_REQ_CRU	LV_CT	NC_ETC_CONF
LV_REQ_DOWN_SHIFT			

Formula section:

EMS name	Comment – Formula	Output to CAN
	CAN version (actual Version of HMC CAN Spec: V1.4) CAN_VERS_CAN = 0CH	**CAN_VERS_CAN**
CONF_TCU	CONF_TCU_CAN = CONF_TCU	**CONF_TCU_CAN**
	OBD freeze frame See CAN specification for more details.	**OBD_FRF_ACK_CAN**

TQ_STND	Standard engine torque in Nm, to which the percentage torque quantities TQx_yyy are related. TQ_STD_CAN = TQ_STND	**TQ_STND_CAN**
TCO	**If** LV_ERR_TCO = 1 **or** LV_ERR_TCO_STUCK = 1 **or** LV_ERR_TCO_GRD = 1 **Then** TEMP_ENG_CAN = FFh **Else** TEMP_ENG_CAN = TCO	**TEMP_ENG_CAN**
MAF_FAC_ALTI_MMV	**If** LV_ERR_MAF = 1 **or** LV_ERR_RATIO_CHK = 1 **or** LV_ERR_MAP = 1 **Then** MAF_FAC_ALTI_MMV_CAN = 80h **Else** MAF_FAC_ALTI_MMV_CAN = MAF_FAC_ALTI_MMV	**MAF_FAC_ALTI_MMV_CAN**
LV_VB_OFF	VB_OFF_ACT_CAN = LV_VB_OFF	VB_OFF_ACT_CAN
LV_ES	ACK_ES_CAN = LV_ES	ACK_ES_CAN
CONF_MIL_FMY	CONF_MIL_FMY_CAN = CONF_MIL_FMY	**CONF_MIL_FMY_CAN**
TPS	**If** LV_ERR_TPS = 1 **or** LV_ERR_TPS_PLAUS = 1 **then** TPS_CAN = FFh **else If** TPS >= C_TPS_MAX_CAN **Then** TPS_CAN = F5h **Else If** LV_CT = 1 **Then** TPS_CAN = 20h **Else** TPS_CAN = $\left(\frac{(TPS - C\_TPS\_IS\_OUT)}{C\_TPS\_MAX\_CAN} * (F5h - 20h)\right) + 20h$	**TPS_CAN**
	In case of non-ETC system only : PV_AV_CAN = FFh In case of PV(both channel) or monitoring error : PV_AV_CAN = FFh In case of curise active : PV_AV_CAN = PV_CRU In normal case : PV_AV_CAN = PV_AV For more detailed information, please see module xx4031yy.zzz (Determination of accel. pedal value (appl. inc.)) in chapter 4 'Definition of PV_AV_CAN'.	**PV_AV_CAN**
ENG_CHR	ENG_CHR_CAN = ENG_CHR	**ENG_CHR_CAN**
ENG_VOL	ENG_VOL_CAN = ENG_VOL	**ENG_VOL_CAN**
LV_BLS	If NC_ETC_CONF = 0 Then BRAKE_ACT_CAN = 0h Else if LV_ERR_BLS_BTS = 01h Then BRAKE_ACT_CAN = 03h Else if LV_BLS = 0 Then BRAKE_ACT_CAN = 1h Else BRAKE_ACT_CAN = 2h	**BRAKE_ACT_CAN**

LV_CRU_ACT	If LV_CRU_ACT = 1 and FAC_TQ_REQ_DRIV <= FAC_TQ_REQ_CRU Then ACC_ACT = 1 Else ACC_ACT = 0	**ACC_ACT**
LV_REQ_DOWN_SHIFT	OD_OFF_REQ = LV_REQ_DOWN_SHIFT	**OD_OFF_REQ**

Message Structure:

Message: EMS2	Identifier: 0329H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
MUL_INFO	Multiplexed information - CAN Version (CAN_VERS_CAN), - TCU configuration (CONF_TCU_CAN) - OBD freeze frame (OBD_FRF_ACK_CAN) - Torque scaling factor (TQ_STND_CAN)	0	6	CAN_VERS	-
MUL_CODE	Identification of MUL INFO	6	2	00H	-
TEMP_ENG_CAN	Engine coolant temperature	8	8	FFH	FFH
MAF_FAC_ALTI_MMV_CAN	Mass air flow correction factor for altitude	16	8	80H	80H
VB_OFF_ACT_CAN	ECU adaptive values and failure memory erase after battery disconnection	24	1	00H	-
ACK_ES_CAN	Acknowledgement, engine stopped	25	1	00H	-
CONF_MIL_FMY_CAN	Configuration of MIL Handling and Failure Memory Management by ECU	26	3	03H	-
OD_OFF_REQ	Over drive off request to TCU	29	1	00H	-
ACC_ACT	Auto cruise control in activation	30	1	00H	-
CLU_ACK	Clutch operation acknowledge (not used)	31	1	00H	-
BRAKE_ACT_CAN	Indication of brake switch ON/OFF	32	2	01H	11B
ENG_CHR_CAN	Engine Characteristics – Kind of fuel, ETS	34	4	00H	-
GP_CTL	Glow plug control request (not used)	38	2	00H	-
TPS_CAN	Throttle Angle	40	8	20H	FFH
PV_AV_CAN	Accelerator pedal value	48	8	00H	FFH
ENG_VOL_CAN	Engine Displacement	56	8	00H	-

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TPS_IS_OUT	1	0...FFH	0...119.5	0,47	°TPS
Idle speed threshold for output signal formatting					
C_TPS_MAX_CAN	1	0...FFH	0...119.5	0,47	°TPS
Maximum TPS distributed via CAN.					

4.100.6.3 EMS4 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IM_AUTHEN_CAN	-	0...1H	0...1	1	-
Authetication "immobilizer"					
L_MIL_CAN	-	0...1H	0...1	1	-
Check Engine Lamp or Malfunction Indication Lamp (MIL for CARB/OBDII)					
IM_STAT_CAN	-	0...1H	0...1	1	-
AMP_CAN	V	0...1FH	458,98...792,78	10,73	mmHg
Atmospheric Pressure					
FCO_CAN	-	0...FFFFH	0...8388.6	0,128	µl
Fuel Consumption signal EMS					
VB_CAN	-	0...FFH	0...25.8984	0,1	V
Battery voltage					

Input data:

MAF_FAC_ALTI_MMV	FCO_SUM_DIF	VB	STATE_MIL
LV_ERR_CAN_BUS_OFF	CONF_IMOB	LV_ERR_MAF	LV_ERR_RATIO_CHK
LV_ERR_VB	LV_ERR_MAP	LV_LOCK_IMOB	LV_MIL_ACT_REQ
C_CONF_MIL	AMP_MES	LC_TCHA_CONF	

Formula section:

EMS name	Comment – Formula	Output to CAN
LV_LOCK_IMOB	IM_AUTHEN_CAN = LV_LOCK_IMOB	**IM_AUTHEN_CAN**
STATE_MIL LV_MIL_ACT_REQ C_CONF_MIL	**If** LV_ERR_CAN_BUS_OFF = 1 **or** (LV_ERR_CAN_BUS_OFF = 0 **and** STATE_MIL <> MIL_OFF**)** **or** (LV_MIL_ACT_REQ = 1 **and** bit1 of C_CONF_MIL =1**)** **then** L_MIL_CAN = 1 **else** L_MIL_CAN = 0 **endif**	**L_MIL_CAN**
CONF_IMOB	IM_STAT_CAN = CONF_IMOB	**IM_STAT_CAN**
MAF_FAC_ALTI_MMV	**If** LC_TCHA_CONF = 1 **and** LV_ERR_AMP = 0 **Then** AMP_CAN = (AMP_MES[dec] *0,005776) – 42,6256 **Else** **If** LV_ERR_MAF = 1 **or** LV_ERR_RATIO_CHK = 1 **or** LV_ERR_MAP = 1 **Then** AMP_CAN = 1Ch **Else** AMP_CAN [hex] = (MAF_FAC_ALTI_MMV[dec] - 77 [dec]) * 0,5535714 **Endif** **Endif**	**AMP_CAN**
FCO_SUM_DIF	FCO_CAN = FCO_SUM_DIF	**FCO_CAN**
VB	**If** LV_ERR_VB = 1	**VB_CAN**

	Then VB_CAN = FFh **Else** VB_CAN = VB	

Message Structure:

Message: EMS4	Identifier: 0545H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
IM_AUTHEN_CAN	Authetication "immobilzer"	0	1	00H	-
L_MIL_CAN	Lamp "check engine for OBD"	1	1	00H	-
IM_STAT_CAN	Status "immobilizer"	2	1	00H	-
AMP_CAN	Atmospheric Pressure	3	5	1CH	1CH
FCO_CAN	Fuel consumption (low Byte)	8	8	00H	-
	Fuel consumption (high Byte)	16	8	00H	-
VB_CAN	Battery Voltage	24	8	00H	FFH
TQI_ACOR_J	Flywheel torque after torque interventions (low Byte) (not relevant; only for RXC + RZD project)	32	8	00H	-
	Flywheel torque after torque interventions (high Byte) (not relevant; only for RXC + RZD project)	40	8	00H	-
TQI_J	Flywheel torque (low Byte) (not relevant; only for RXC + RZD project)	48	8	00H	-
	Flywheel torque (high Byte) (not relevant; only for RXC + RZD project)	56	8	00H	-

4.100.6.4 EMS5 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_CAN	-	0...FFH	-48...142,5	0,75	°C
Intake air temperature					
STATE_DC_OBD	V	0 ... 1FH	0 ... 31	1	-
Same to STATE_DC_RBM					
INH_DC_OBD	V	0 ... 1H	0 ... 1	1	-
Same to LV_INH_DC_RBM					
CTR_IG_CYC_OBD	V	0 ... FFFFH	0 ... 65535	1	-
Same to CTR_IGK_CYC_RBM					
CTR_CDN_OBD	V	0 ... FFFFH	0 ... 65535	1	-
Same to CTR_CDN_OBD_RBM					

Input data:

TIA	LV_INH_DC_RBM	STATE_DC_RBM	CTR_IGK_CYC_RBM
CTR_CDN_OBD_RBM			

Formula section:

EMS name	Comment – Formula	Output to CAN
TIA	**If** LV_ERR_TIA = 1 **Then** TIA_CAN = FFh **Else** TIA_CAN = TIA	TIA_CAN
STATE_DC_RBM	STATE_DC_OBD = STATE_DC_RBM	STATE_DC_OBD
LV_INH_DC_RBM	INH_DC_OBD = LV_INH_DC_RBM	INH_DC_OBD
CTR_IGK_CYC_RBM	CTR_IG_CYC_OBD = CTR_IGK_CYC_RBM	CTR_IG_CYC_OBD
CTR_CDN_OBD_RBM	CTR_CDN_OBD = CTR_CDN_OBD_RBM	CTR_CDN_OBD

Message Structure:

Message: EMS5	Identifier: 02A0H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
ECGPOvrd	Driver override (not relevant)	0	1	00H	-
QECACC	Failure in ACC-message detected by engine control (not relevant)	1	1	00H	-
ECFail	Function failure engine power control (not relevant)	2	1	00H	-
SwitchOffCondExt	ACC shut off condition detected (not relevant)	3	1	00H	-
BLCEFail	Failure brake light switch detected by engine control (not relevant)	4	1	00H	-
Free	Free	5	3	00H	-
FA_PV_CAN	Filtered Accelerator Pedal value(not relevant)	8	8	00H	FFH
TIA_CAN	Intake Air temperature	16	8	00H	FFH
STATE_DC_OBD	Status of rate-based monitoring	24	7	00H	-
INH_DC_RBM	Inhibition of rate based monitoring	31	1	00H	-
CTR_IG_CYC_OBD	Ignition cycle counter (low Byte)	32	8	00H	-
	Ignition cycle counter (high Byte)	40	8	00H	-
CTR_CDN_OBD	General denominator calculation (low Byte)	48	8	00H	-
	General denominator calculation (high Byte)	56	8	00H	-

4.100.6.5 EMS6 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_MIN_CAN	V	0...FFH	0...99.61	0.391	%
Minimum indicated engine torque					
TQI_TARGET	V	0...FFH	0...99.61	0.391	%
Indicated engine torque without torque intervention (CAN-value)					
CRUISE_LAMP_M	-	0...1	0...1	1	-
Cruise MAIN Switch indication lamp on request					
CRUISE_LAMP_S	-	0...1	0...1	1	-
Cruise SET Switch indication lamp on request					
ENG_STAT	V/O	0...7H	0...7	1	-
Engine status					
TQI_MAX_CAN	V	0...FFH	0...99.61	0.391	%
Maximum indicated engine torque (CAN-value)					
SPK_TIME_CUR	V	0...FFH	-35.625...60	0.375	°CRK
Current spark timing incl. requests by ESP, ACC and TCU					
SOAK_TIME_ERROR	-	0...1	0...1	1	-
Soak time error					
SOAK_TIME	V	0...FFH	0...255	1	min
Engine soaking time					

Input data:

TQI_REF_IGA_MIN_LAMB	TQI_CAN	TQ_REQ_CLU_1	TQ_LOSS
LV_CRU_MAIN_SWI	LV_CRU_ACT	TQI_REF_MAX	EFF_IGA_BAS_COR
STATE_ENG	LV_STALL	IGA_AV_MV	LV_ERR_T_ES_PLAUS
T_ES	LV_T_ES_NOT_PLAUS		

Formula section:

EMS name	Comment - Formula	Output to CAN
TQI_REF_IGA_MIN_LAMB	TQI_MIN_CAN = TQI_REF_IGA_MIN_LAMB / TQ_STND * 100%	TQI_MIN_CAN
	from EMS1 message	TQI_CAN
TQ_REQ_CLU_1 TQ_LOSS	TQI_TARGET = (TQ_REQ_CLU_1 – TQ_LOSS) / TQ_STND * 100%	TQI_TARGET
LV_CRU_MAIN_SWI	LV_CRU_MAIN_SWI = CRUISE_LAMP_M	CRUISE_LAMP_M
LV_CRU_ACT	LV_CRU_ACT = CRUISE_LAMP_S	CRUISE_LAMP_S
STATE_ENG LV_STALL	IF STATE_ENG = 0H (ES) THEN IF LV_STALL = 1 THEN ENG_STAT = 2H (Stalled) ELSE ENG_STAT = 0H (Engine Stop) ELSE IF STATE_ENG = 1H (ST) THEN ENG_STAT = 01H (Cranking) ELSE ENG_STAT = 03H (Runnung) ENDIF	ENG_STAT
LV_ERR_T_ES_PLAUS LV_T_ES_NOT_PLAUS	IF LV_ERR_T_ES_PLAUS = 1 or LV_T_ES_NOT_PLAUS = 1 THEN SOAK_TIME_ERROR = 1 ELSE SOAK_TIME_ERROR = 0	SOAK_TIME_ERROR

T_ES	IF T_ES <= FFH THEN SOAK_TIME = T_ES ELSE SOAK_TIME = FFH	SOAK_TIME
TQI_REF_MAX EFF_IGA_BAS_COR	TQI_MAX_CAN = (TQI_REF_MAX * EFF_IGA_BAS_COR) / TQ_STND * 100%	TQI_MAX_CAN
IGA_AV_MV	IGA_AV_MV = SPK_TIME_CUR	SPK_TIME_CUR

Message Structure:

Message: EMS6	Identifier: 0260H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
TQI_MIN_CAN	Minimum indicated engine torque	0	8	00H	-
TQI_CAN	Indicated engine torque	8	8	00H	-
TQI_TARGET	Target engine torque	16	8	00H	-
Glow_STAT	Glow lamp status (only for Diesel-ISG Project)	24	1	00H	-
CRUISE_LAMP_M	Cruise MAIN switch indication lamp on request	25	1	00H	-
CRUISE_LAMP_S	Cruise SET switch indication lamp on request	26	1	00H	-
Free	Free	27	1	00H	-
ENG_STAT_CAN	Engine Status	28	3	00H	07H
SOAK_TIME_ERROR	Soak time error	31	1	00H	-
SOAK_TIME	Engine soaking time	32	8	00H	-
TQI_MAX_CAN	Maximum indicated engine torque	40	8	00H	-
SPK_TIME_CUR	Current spark timing	48	8	00H	-
Free	Free	56	8	00H	-

4.100.6.6 EMS_H2 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
R_TQACNAPVC	V	0...FEH	0...50.8	0.2	Nm
Air conditioning torque loss					
R_PACNC	V	0...FEH	0...31750	125	hpa
Air conditioning pressure					

Input data:

TQ_MAX_ACC_FATC	ACP	LV_ERR_ACP	

Formula section:

EMS name	Comment - Formula	Output to CAN
TQ_MAX_ACC_FATC	R_TQACNAPVC = TQ_MAX_ACC_FATC(physically in absolute value) with different resolution and sign(R_TQACNAPVC in positive value) TQ_MAX_ACC_FATC : 0.03125 Nm R_TQACNAPVC : 0.2 Nm	R_TQACNAPVC
ACP	If LV_ERR_ACP = 1 Then R_PACNC = FFH Else R_PACNC(hpa) = ACP(psi) x 68.94(physically) with different resolution ACP : 2(psi) = 2 x 68.94 (hpa) R_PACNC : 125 (hpa)	R_PACNC

Message Structure:

Message: EMS_H2	Identifier: 018FH

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
R_TqAcnApvC	Max torque limitation of the A/C compressor (not supported)	0	8	00H	FFH
R_PAcnC	APT Sensor output value (not supported)	8	8	00H	FFH
TQI_B	Indicated engine torque (not supported)	16	8	00H	-
SLD_VS	Speed limiter vehicle speed (not supported)	24	8	00H	-
Reserved	CDA MODE Status flag (not supported)	32	3	00H	07H
Free	Free	35	29	00H	-

4.100.6.7 EngFrzFrm1 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PID_04h	-	0...FFH	0...100	100/255	%
Freeze Frame – Calculated LOAD value					
PID_05h	-	0...FFH	-40...215	1	°C
Freeze Frame – Engine coolant temperature					
PID_0Ch	-	0...FFFFH	0...16383.75	0.25	rpm
Freeze Frame – Engine RPM					
PID_0Dh	-	0...FFH	0...255	1	km/h
Freeze Frame – Vehicle speed sensor					
PID_11h	-	0...FFH	0...100	100/255	%
Freeze Frame – Absolute throttle position					
PID_03h	-	1H 2H 4H 8H 10H	OL_CDN CL OL_INTR OL_ERR CL_ERR	1	-
Freeze Frame – Fuel Control system Status					

Input data:

LOAD_CLC	OBD_TCO	OBD_N	VS
OBD_TPS_AV_1	STATE_LS[NC_CBK_EX_NR]		

Formula section:

EMS name	Comment - Formula	Output to CAN
LOAD_CLC	PID_04h = LOAD_CLC	PID_04h
OBD_TCO	PID_05h = OBD_TCO	PID_05h
OBD_N	PID_0Ch = OBD_N	PID_0Ch
VS	PID_0Dh = VS	PID_0Dh
OBD_TPS_AV_1	PID_11h = OBD_TPS_AV_1	PID_11h
STATE_LS	PID_03h = STATE_LS	PID_03h

Message Structure:

Message: EngFrzFrm1	Identifier: 00A0H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
PID_04h	Freeze Frame – Calculated LOAD value	0	8	00H	-
PID_05h	Freeze Frame – Engine coolant temperature	8	8	00H	-
PID_0Ch	Freeze Frame – Engine RPM (low Byte)	16	8	00H	-
	Freeze Frame – Engine RPM (high Byte)	24	8	00H	-
PID_0Dh	Freeze Frame – Vehicle speed sensor	32	8	00H	-
PID_11h	Freeze Frame – Absolute throttle position	40	8	00H	-
PID_03h	Freeze Frame – Fuel Control system Status (low Byte)	48	8	00H	-
	Freeze Frame – Fuel Control system Status (high Byte)	56	8	00H	-

4.100.6.8 EngFrzFrm2 message

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PID_06h	-	0...FFH	-100...99,22	100/128	%
Freeze Frame – Short Term Fuel Trim Bank1					
PID_07h	-	0...FFH	-100...99,22	100/128	%
Freeze Frame – Long Term Fuel Trim Bank1					
PID_0Bh	-	0...FFH	0...255	1	kPa
Freeze Frame – Manifold Absolute Pressure					

Input data:

OBD_LAM_COR_i	OBD_LAM_AD_i	OBD_MAP	

Formula section:

EMS name	Comment - Formula	Output to CAN
OBD_LAM_COR_1	PID_06h = OBD_LAM_COR_1	PID_06h
OBD_LAM_AD_1	PID_07h = OBD_LAM_AD_1	PID_07h
OBD_MAP	PID_0Bh = OBD_MAP	PID_0Bh

Message Structure:

Message: EngFrzFrm2	Identifier: 00A1H

Output period: 10ms

Signal Label	Signal designation	Bit add	Bit ind	Init value	Error ident.
PID_06h	Freeze Frame – Short Term Fuel Trim Bank1	0	8	80H	-
PID_07h	Freeze Frame – Long Term Fuel Trim Bank1	8	8	80H	-
PID_08h	Freeze Frame – Short Term Fuel Trim Bank2 (not supported)	16	8	00H	-
PID_09h	Freeze Frame – Long Term Fuel Trim Bank2 (not supported)	24	8	00H	-
PID_0Bh	Freeze Frame – Manifold Absolute Pressure	32	8	00H	-
PID_23h	Freeze Frame – Fuel Pressure (low Byte) (not supported)	40	8	00H	-
	Freeze Frame – Fuel Pressure (high Byte) (not supported)	48	8	00H	-
Free	Free	56	8	00H	-

4.100.7 Torque variable definitions

4.100.7.1 Standard torque (TQ_STND)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_STND	V/O	0 ... 3FH	0 ... 630	10	Nm
standard engine torque					

The standard engine torque **TQ_STND** is taken from C_TQ_STND and transmitted by the engine management system to the CAN stations (TCS control unit and TCU).

It is the reference torque to which the torque reduction quantities are related. The result are relative torque quantities as percentage which are put on the CAN.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_STND	1	0 ... 3FH	0 ... 630	10	Nm
standard engine torque					

4.100.7.2 Status of Torque Intervention (STATE_TQ_INTV)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_TQ_INTV	V/O	0...3H	0...3	1	
status torque intervention from EMS					
TQ_INTV_REL_ERR	V/O	0 ... FFH	0 ...0,996	0,0039	100%
relative error as difference between actual torque and requested torque related to the standard torque					

Input data:

TQI_AV	TQI_GS_REQ	TQI_ASR_REQ	TQ_STND
LV_AUTH_TQ_PAT	LV_TQ_IGA_ENA		

General information:

The engine management system uses the variable **STATE_TQ_INTV** to inform the transmission control unit and/or the traction system control unit about the state of the torque intervention. That means if and to which extent an ignition angle intervention and/or a cylinder shut-off intervention are admitted.

Status	Bit 5	Bit 4	Meaning
0	**0**	**0**	The desired intervention regarding ignition angle retardation and cylinder shut-off is executed. (Default value)
1	**0**	**1**	The desired intervention regarding ignition angle retardation and cylinder shut-off is executed; however, the requested target torque can not be adjusted precisely (torque steps)
2	**1**	**0**	The torque reduction regarding the ignition angle retardation cannot be completely executed. A cylinder shut-off is not possible at this time. Therefore a remaining torque (as difference between TQI_ASR/GS_REQ and TQI_AV) is present and cannot be reduced.
3	**1**	**1**	The desired torque intervention for TCS regarding the ignition angle and cylinder shut-off can no longer be executed. The torque intervention is terminated, the engine management system resets the requested engine torque to the TQI value using a ramp.

Activation of TQ_INTV_REL_ERR and STATE_TQ_INTV calculation:

TQI_TCS_CAN < FFh or TQI_TCU_CAN < FFh

Deactivation: TQI_TCS_CAN = FFh and TQI_TCU_CAN = FFh

Formula section:

Relative Error of torque reduction intervention

TQ_INTV_REL_ERR = (TQI_AV – TQI_xx_REQ) / TQ_STND

Status of torque intervention:

Status 0: **If** $|TQ\_INTV\_REL\_ERR| \le C\_TQ\_INTV\_REL\_ERR\_MAX$
then STATE_TQ_INTV = 0

Status 1: **If** $|TQ\_INTV\_REL\_ERR| > C\_TQ\_INTV\_REL\_ERR\_MAX$
and LV_AUTH_TQ_PAT = 1
then STATE_TQ_INTV = 1

Status 2: **If** $|TQ\_INTV\_REL\_ERR| > C\_TQ\_INTV\_REL\_ERR\_MAX$
and LV_AUTH_TQ_PAT = 0
then STATE_TQ_INTV = 2

Status 3: **If** no torque intervention is possible
LV_TQ_IGA_ENA = 0 **and** LV_AUTH_TQ_PAT = 0
then STATE_TQ_INTV = 3
and TQ_INTV_REL_ERR is set to 100%

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_INTV_REL_ERR_MAX	1	0 ... FFH	0 ...0,996	0,0039	100%
threshold for relative error of torque intervention for status output					

4.100.8 Definition of Engine Identification value

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ENG_CHR	V/O	0...FH	0...15	1	-
Engine Characteristic – Kind of fuel, ETS					
ENG_VOL	V/O	0...FFH	0...25,5	0,1	L
Engine Displacement					

General information:

This variable is to give an information for an engine identification(ex. Engine volume and engine type) for automatic variant coding.

Formula section:

1) Engine Characteristic (ENG_CHR)

- Initial value: 00H

ENG_CHR = C_ENG_CHR

Conversion:

ENG_CHR				Function
Reserv.	ETC	FUEL		
0	0	0	0	Fuel – Gasoline, non-ETC
0	0	0	1	Fuel – LPI(LPG), non-ETC
0	0	1	0	Fuel – Diesel, non-ETC–
0	1	0	0	Fuel – Gasoline, ETC – Applied
0	1	0	1	Fuel – LPI(LPG), ETC – Applied
0	1	1	0	Fuel – Diesel, ETC – Applied
X	X	1	1	Reserved - Not Used
1	X	X	X	Reserved - Not Used

2) Engine displacement

ENG_VOL = C_ENG_VOL

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ENG_CHR	1	0 ... 06H	0 ...6	1	-
Engine Characteristic – Kind of fuel, ETC					
C_ENG_VOL	1	0 ... FFH	0 ...25,5	0,1	L
Engine Displacement					

4.101 CAN Messages EMS (Appl. Inc.)

4.101.1 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TOIL_GB_MAX_TOT_DC	V/O/S	0...FEH	-40...214	1	°C
Former / current driving cycle maximum A/T fluid temperature					

Input data:

TOIL_GB	LV_IGK		

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

TOIL_GB_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: 10ms

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If TOIL_GB ≠ FFh

If TOIL_GB > TOIL_GB_MAX_TOT_DC

Then TOIL_GB_MAX_TOT_DC = TOIL_GB

Endif

Endif

4.102 Engine Efficiency state

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ENG_EFF_STATE	V/O	0H 1H 2H	LOW NORMAL HIGH	1	-
Engine Efficiency state					
T_DLY_ENG_EFF_STATE_SWI	V	0 ..FFH	0...25.5	1	s
Delay timer before a switch of engine efficency state is allowed after last switch					
LV_ENG_EFF_SWI_INH	V	0 ..1H	0...1	1	-
Flag indication a switch of engine efficiency state is inhibited					

Input data:

LV_BLS	LV_PU	LV_PUC	VS
N_32	LV_CS	CONF_BAT	LV_GS_ACT
LV_IS	TQ_REQ_CLU	PV_AV	GR_MT
CONF_CRU			

FUNCTION DESCRIPTION:

General information:

The engine efficiency is determined depending on the Torque request, the brake pedal value, the vehicle speed, engine speed and engine state. At high engine efficiency the battery is charged, at low the alternator load can be decreased.

Application conditions:

Initialisation: *at Reset:* ENG_EFF_STATE = NORMAL
T_DLY_ENG_EFF_STATE_SWI_=
C_T_DLY_ENG_EFF_STATE_SWI
LV_ENG_EFF_SWI_INH = 0

Recurrence: *100 ms*

Activation: CONF_BAT = 1

Formula section:

Calculation of Inhibit conditions for Engine efficiency state switch:

if LV_AT = 1
then **if** LV_GS_ACT = 1
gear change at Automatic Transmission vehicle
then LV_ENG_EFF_SWI_INH = 1
else LV_ENG_EFF_SWI_INH = 0

```
            endif
    else    if      CONF_CRU = 1
            then  if            LV_CS = 1
                        and     VS >= C_VS_ENG_EFF_INH
                                gear change at ManualTransmission vehicle (equipped with clutch
                                 switch)
                  then          LV_ENG_EFF_SWI_INH = 1
                  else          LV_ENG_EFF_SWI_INH = 0
                  endif
            else  if            GR_MT = 0
                       and      VS >= C_VS_ENG_EFF_INH
                                gear change at ManualTransmission vehicle (equipped without
                                 clutch switch)
                  then          LV_ENG_EFF_SWI_INH = 1
                  else          LV_ENG_EFF_SWI_INH = 0
                  endif
            endif
    endif
```

Calculation of engine effiency state:

```
    If      LV_ENG_EFF_SWI_INH = 0
      and   T_DLY_ENG_EFF_STATE_SWI = 0
    then      if     LV_BLS = 1
              or     LV_PU = 1
              or     LV_PUC = 1
              or     (TQ_REQ_CLU < IP_TQ_REQ_CLU_HIGH and LC_ENG_EFF_TQ = 1)
              or     (PV_AV < IP_PV_AV_ENG_EFF_HIGH and LC_ENG_EFF_TQ = 0)]
              and    VS >=  C_VS_ENG_EFF_HIGH
              then   ENG_EFF_STATE = HIGH
              else   if      (TQ_REQ_CLU > IP_TQ_REQ_CLU_LOW and
                             LC_ENG_EFF_TQ =1)
                     or      (PV_AV > IP_PV_AV_ENG_EFF_LOW and
                              LC_ENG_EFF_TQ = 0)
                     and      VS >=  C_VS_ENG_EFF_LOW
                     then    ENG_EFF_STATE = LOW
```

else ENG_EFF_STATE = NORMAL

endif

endif

endif

if ENG_EFF_STATE = ENG_EFF_STATE (n-1)

then T_DLY_ENG_EFF_STATE_SWI = T_DLY_ENG_EFF_STATE_SWI (n-1) -1 h

Decrease Switch delay time by 1h each 100msec

else T_DLY_ENG_EFF_STATE_SWI = C_T_DLY_ENG_EFF_STATE_SWI

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_REQ_CLU_HIGH	8	0...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_IP_TQ_REQ_CLU	8	0...FFH	0...8160	32	rpm
TQ_REQ_CLU Threshold for Engine efficiency HIGH					
IP_TQ_REQ_CLU_LOW	8	0...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_IP_TQ_REQ_CLU	8	0...FFH	0...8160	32	rpm
TQ_REQ_CLU Threshold for Engine efficiency LOW					
IP_PV_AV_ENG_EFF_HIGH	8	0...FFH	0...99.6	0.3906	%
LDPM_N_32_IP_PV_AV_ENG_EFF	8	0...FFH	0...8160	32	rpm
PV_AV Threshold for Engine efficiency HIGH					
IP_PV_AV_ENG_EFF_LOW	8	0...FFH	0...99.6	0.3906	%
LDPM_N_32_IP_PV_AV_ENG_EFF	8	0...FFH	0...8160	32	rpm
PV_AV Threshold for Engine efficiency LOW					
C_VS_ENG_EFF_INH	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for Engine efficiency change is inhibited					
C_VS_ENG_EFF_LOW	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for Engine efficiency LOW					
C_VS_ENG_EFF_HIGH	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for Engine efficiency HIGH					
C_T_DLY_ENG_EFF_STATE_SWI	1	0...FFH	0...25.5	1	Sec
Delay time before switch of Engine efficiency state is allowed after last switch					
LC_ENG_EFF_TQ	1	0...1H	0...1	1	-
Logical switch to change between TQ_REQ_CLU and PV_AV for engine efficiency calibration					

5 Engine operating states

Table of Contents

Index

T

5.1 General

A distinction is made on principle between the basic operating states and the auxiliary functions.

The basic operating states (e.g. **LV_ES**, **LV_ST**, **LV_IS**, **LV_PL**, **LV_PU** & **LV_PUC**) are independent basic functions which can only occur alternatively. Contrary to this, auxiliary functions can be simultaneously active and they always are superimposed on a basic operating state.

5.1.1 Engine operating state detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_ENG	V/O	0...0H 1...1H 2...2H 3...3H 4...4H 5...5H	"ES" "ST" "IS" "PL" "PU" "PUC"	-	-
Engine operating state					

Input data:

LV_ES	LV_ST	LV_IS	LV_PL
LV_PU	LV_PUC		

General information:

If the detection of an engine operating state influences several output signals, this engine operating state is described in the chapter ” Basic Operating States ”.

The engine operating state transitions are dependent on **throttle position** and **engine speed.**

The driver must be able to switch - off the engine by the ignition key in any operating state. To implement this, the battery voltage is measured.

For more explanations, refer to “ Acquisition of ignition key battery voltage “ in the major chapter “ Inputs “.

5.1.2 Function initialization

After initialization of the operating system and self - diagnosis of the control unit, the engine functions are assigned to their initial values.

The first measured values are diagnosed and backup values are made available as appropriate.

5.1.2.1 Reset with the system running

Input data:

N	NC_N_MIN	N_32	IP_N_MAX_TOL_ST__TCO
LV_ST			

FUNCTION DESCRIPTION:

General information:

If a reset occurs for N >= NC_N_MIN, initialization is performed as for an under-voltage reset.

In the case of an engine speed N_32 >= IP_N_MAX_TOL_ST__TCO the engine operating state start (LV_ST) is not executed.

Instead, the system immediately branches to the current engine operating state.

5.1.3 Engine operating state " Engine Stopped " (LV_ES)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ES	V/O	0...01H	0...1	1	-
Engine operating state " Engine Stopped " - Engine Operating State = ES.					

Input data:

LV_IGK	LV_ST	N	NC_N_MIN

FUNCTION DESCRIPTION:

General information:

The engine operating state engine stopped (LV_ES) is characterized by ignition key **on** (LV_IGK **= 1** (Active)) and N **<** NC_N_MIN.

In this engine operating state, the functions are assigned to initialization values.

The entire scope of diagnosis and actuator control is accessible.

Application conditions:

Deactivation:

Exit to **LV_ST :** Start

During the synchronisation phase, the tooth duration is taken into account to calculate the engine speed.

If N >= NC_N_MIN is detected in the operating state engine stopped (LV_ES)

then the fuel pump is switched on

(refer to chapter : " Relays control " in major chapter " Auxiliary Functions ")

and the first injection is performed

(refer to chapter : " Fuel injection output " in major chapter " Outputs ")

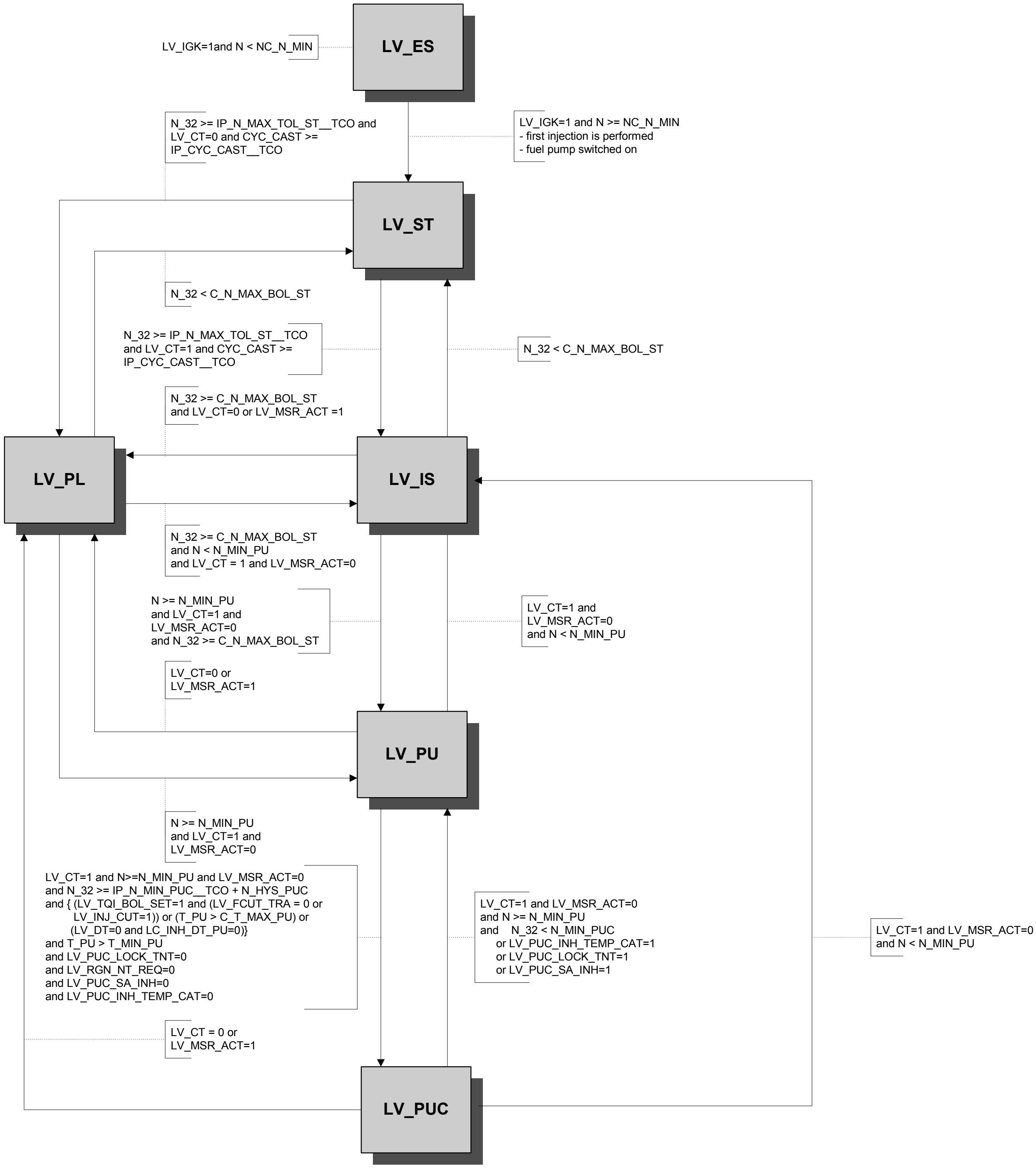
5.1.4 General Drawing

LV_ES : Engine Stopped

LV_PL : Part Load

LV_ST : Engine Start

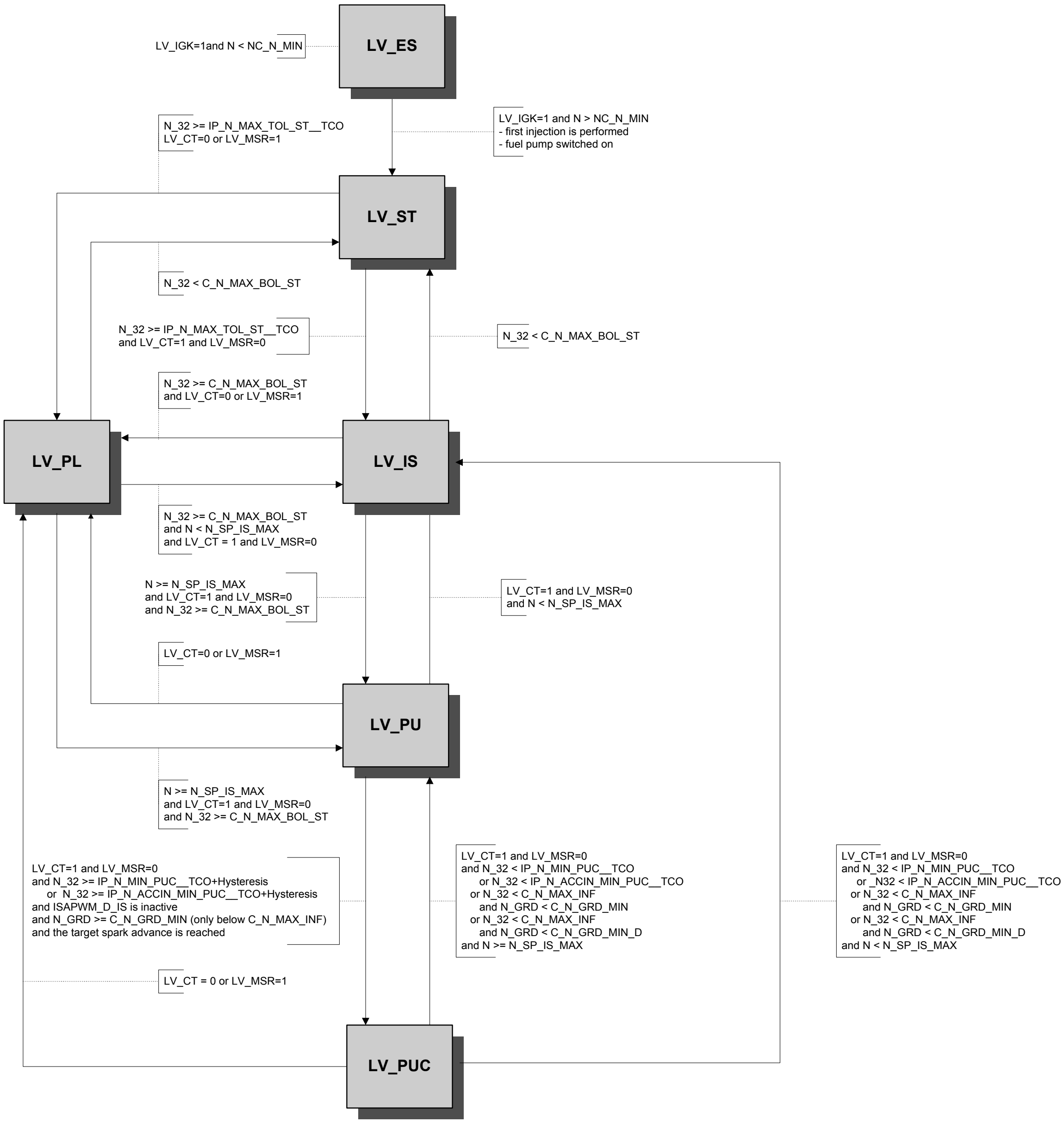
LV_PU : Trailing Throttle



LV_IS : Idle Speed

LV_PUC : Trailing Throttle Fuel Cut Off

5.1.5 General Drawing



LV_ES	:	Engine Stopped
LV_ST	:	Start
LV_IS	:	Idle Speed
LV_PL	:	Part Load
LV_PU	:	Trailing Throttle
LV_PUC	:	Trailing Throttle Fuel Cut Off
LV_FL	:	Full Load

5.2 Basic Operating States : LV_ST, LV_IS, LV_PL, LV_PU & LV_PUC

5.2.1 Engine operating state : " Start " (LV_ST)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ST	V / O	0...1H	0...1	-	-
Engine operating state "Start"					
LV_ST_END	V / O	0...1H	0...1	-	-
Engine operating state "Start" not active					

Input data:

LV_CT	N	TCO	LV_IS
LV_PL	N_MAX_TOL_ST	LV_ES	CYC_CAST

FUNCTION DESCRIPTION:

The engine operating state "Start" (**LV_ST**) is detected from part load and idle speed via means of engine speed.

Application conditions:

Recurrence: segment-synchronous; only once per 10 ms

Determination of LV_ST:

Deactivation:

1) **Exit to LV_IS :** Idle Speed
a) N >= N_MAX _TOL_ST
b) **and** LV_CT = 1
c) **and** CYC_CAST >= IP_CYC_CAST__TCO

2) **Exit to LV_PL :** Part Load
a) N >= N_MAX_TOL_ST
b) **and** LV_CT = 0
c) **and** CYC_CAST >= IP_CYC_CAST__TCO

Determination of LV_ST_END:

If LV_ST = 1 **or** LV_ES = 1
then LV_ST_END = 0
else LV_ST_END = 1

Note : N and N_MAX_TOL_ST have different resolution (1 rpm & 32 rpm)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CYC_CAST__TCO	9	0...FFFF H	0...65535	1	Seg
LDPM_TCO_9	9	0...FE H	-48...142.5	0.75	°C
Cycle counter to define the duration of the start phase charcterised by LV_ST					

5.2.2 Engine operating state : " Idle Speed " (LV_IS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_IS	V / O	0...1H	0...1	1	-
Engine operating state "Idle Speed"					
T_IS	V / O	0...FFFFH	0...655.35	0.01	s
Cumulated time in engine operating state Idle Speed and vehicle stopped					

Input data:

LV_CT	N_32	N	N_SP_IS
LV_MSR_ACT	LV_ST	LV_PL	LV_PU
N_MIN_PU	LV_N_INC_TCU_ACT		

Application conditions:

Recurrence: segment-synchronous; only once per 10 ms

Deactivation:

1) **Exit to LV_ST :** Start
a) N_32 < C_N_MAX _BOL_ST

2) **Exit to LV_PL :** Part Load
a) N_32 >= C_N_MAX_BOL_ST
b) **and** LV_CT = 0
or LV_MSR_ACT = 1

3) **Exit to LV_PU :** Trailing Throttle
a) N >= N_MIN_PU
b) **and** LV_CT = 1
c) **and** N_32 >= C_N_MAX_BOL_ST
d) **and** LV_MSR_ACT = 0
e) **and** LV_N_INC_TCU_ACT = 0

Calculation of the Timer T_IS (Cumulated time with LV_IS = 1 and VS = 0) **:**

If (LV_IS = 1 **and** Transition VS > 0 -> VS = 0)
or (VS = 0 **and** Transition LV_IS = 0 -> LV_IS = 1)
Then TI_IS = 0
Else **If** LV_IS = 1
and VS = 0
Then T_IS is incremented every 0.01s until it reaches max. value 655.35 s

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_MAX_BOL_ST	1	0...FFH	0...8160	32	rpm
Engine speed threshold to detect LV_ST or LV_PL from LV_IS					

5.2.3 Engine operating state : ” Part Load ” (LV_PL)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PL	V / O	0...1H	0...1	1	-
Engine operating state „Part Load“					

Input data:

LV_CT	N_32	N	N_SP_IS
LV_ST	LV_IS	LV_PU	LV_MSR_ACT
C_N_MAX_BOL_ST	N_MIN_PU		

Application conditions:

Recurrence: segment-synchronous; only once per 10 ms

Deactivation:

1) **Exit to LV_ST :** Start
a) N_32 < C_N_MAX _BOL_ST

2) **Exit to LV_IS :** Idle Speed
a) N_32 >= C_N_MAX_BOL_ST
b) **and** N < N_MIN_PU
c) **and** LV_CT = 1
d) **and** LV_MSR_ACT = 0

3) **Exit to LV_PU :** Trailing Throttle
a) N >= N_MIN_PU
b) **and** LV_CT = 1
c) **and** LV_MSR_ACT = 0

5.2.4 Engine operating state : ” Trailing Throttle ” (LV_PU)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PU	V / O	0...1H	0...1	-	-
Engine operating state „Trailing throttle“					
T_MIN_PU	V / O	0...FFFFH	0...655.35	0.01	s
Minimum duration of PU state					
T_PU	V / O	0...FFFFH	0...655.35	0.01	s
Time elapsed into LV_PU					
N_MIN_PU	V	0...1FE0H	0...8160	1	rpm
Minimum engine speed of PU state					
T_PU_TCU	V	0...FFFFH	0...655.35	0.01	s
Time elapsed into LV_PU after after entry of PU by TCU request					

Input data:

LV_CT	N_32	N_GRD	N_SP_IS
LV_AT	LV_PUC_INH_TEMP_CAT	LV_TQI_BOL_SET	LV_PL
LV_IS	LV_PUC	C_N_DIF_MAX_IS	N_MIN_PUC
N	N_HYS_PUC	LV_MSR_ACT	LV_DT
N_DIF_MMV	LV_PUC_LOCK_TNT	LV_RGN_NT_REQ	LV_PUC_SA_INH
C_N_DIF_FAC	LV_FCUT_TRA	LV_INJ_CUT	LC_INH_DT_PU
VS	GR_AT	LV_N_INC_TCU_ACT	SWI_CC

Application conditions:

Recurrence: segment-synchronous; only once per 10 ms

Deactivation:

1) **Exit to LV_PL :** Part Load
a) LV_CT = 0
or LV_MSR_ACT = 1

2) **Exit to LV_IS :** Idle Speed
a) LV_CT = 1
b) **and** **(** N < N_MIN_PU **or** LV_N_INC_TCU_ACT = 1 **)**
c) **and** LV_MSR_ACT = 0

3) **Exit to LV_PUC :** Trailing Throttle Fuel Cut Off

a) LV_CT = 1
b) **and** N >= N_MIN_PU
c) **and** LV_MSR_ACT = 0
d) **and** N_32 >= N_MIN_PUC + N_HYS_PUC
e) **and** **{**(LV_TQI_BOL_SET = 1)
and (LV_FCUT_TRA = 0 or LV_INJ_CUT=1)
or (T_PU > C_T_MAX_PU) ;maximum PU - time
or (LV_DT = 0 **and** LC_INH_DT_PU = 0)**}** ;open drivetrain
f) **and** **{** (T_PU_TCU = 0 **and** T_PU > T_MIN_PU) ;normal driving
or ($T\_PU\_TCU_{(n-1)}$ > 0 **and** $T\_PU\_TCU_{(n)}$ = 0) **}** ; N increased by TCU
g) **and** LV_PUC_LOCK_TNT = 0
h) **and** LV_RGN_NT_REQ = 0
i) **and** LV_PUC_SA_INH = 0

j) **and** LV_PUC_INH_TEMP_CAT = 0
k) **and** LV_N_INC_TCU_ACT = 0 ;to delay exit to PUC till N increment by TCU finished

Activation of the timer T_PU:

Recurrence: 10 ms

If LV_PU = 1

then T_PU incremented until FFFF H

else T_PU is initialized with 0

Activation of the timer T_PU_TCU:

Recurrence: 10 ms

If LV_N_INC_TCU_ACT = 1 -> 0

then T_PU_TCU is initialized with IP_T_MIN_PU_TCU

endif

if T_PU_TCU >0

and LV_N_INC_TCU_ACT = 0

then T_PU_TCU is decremented until 0

else T_PU_TCU = 0

endif

Note : T_PU_TCU is calculated to delay transition from PU to PUC when N increaed by TCU request

Definition of T_MIN_PU

If LV_AT = 0
Then **If** LV_DT = 0
Then T_MIN_PU = C_T_MIN_PU
Else T_MIN_PU = C_T_MIN_PU_DT
Else
If LV_DT = 0
Then T_MIN_PU = C_T_MIN_PU_AT
Else T_MIN_PU = IP_T_MIN_PU_AT_DT (N_32)

Note : T_MIN_PU is updated only at the entry in PU (transition XX --> PU).
If LV_DT state changes during PU, then T_MIN_PU is not updated.

Definition of N_MIN_PU

LV_AT = 0

In idle(LV_IS) :

N_MIN_PU = N_SP_IS + C_N_DIF_MAX_IS – N_DIF_MMV * C_N_DIF_FAC

Out of idle :

N_MIN_PU = N_SP_IS + C_N_DIF_MAX_IS

LV_AT = 1

In idle(LV_IS) :

If SWI_CC = 0

Then N_MIN_PU = N_SP_IS + IP_N_DIF_MAX_IS_0 – N_DIF_MMV * C_N_DIF_FAC

Else N_MIN_PU = N_SP_IS + IP_N_DIF_MAX_IS_1 – N_DIF_MMV * C_N_DIF_FAC_CC

Out of idle :

If SWI_CC = 0

Then N_MIN_PU = N_SP_IS + IP_N_DIF_MAX_IS_0

Else N_MIN_PU = N_SP_IS + IP_N_DIF_MAX_IS_1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_MIN_PU	1	0...FFFFH	0...655.35	0.01	s
Minimum time delay for PUC activation after entry in PU if LV_DT = 0 (MT gearbox)					
C_T_MIN_PU_AT	1	0...FFFFH	0...655.35	0.01	s
Minimum time delay for PUC activation after entry in PU if LV_DT = 0 (AT gearbox)					
C_T_MIN_PU_DT	1	0...FFFFH	0...655.35	0.01	s
Minimum time delay for PUC activation after entry in PU if LV_DT = 1 (MT gearbox)					
IP_T_MIN_PU_AT_DT	6	0...FFFFH	0...655.35	0.01	s
LDP_N_32_IP_T_MIN_PU_AT_DT	6	0...FFH	0...8160	32	rpm
Minimum time delay for PUC activation after entry in PU if LV_DT = 1 (AT gearbox)					
C_T_MAX_PU	1	0...FFFFH	0...655.35	0.01	s
Maximum delay time of PUC activation after entry of PU					
C_T_MIN_PU_DT	1	0...FFFFH	0...655.35	0.01	s
Minimum delay time of PUC activation after entry of PU for closed drivetrain					
C_N_DIF_MAX_IS	1	0...1FE0H	0...8160	1	rpm
Engine speed hysteresis before “Idle Speed“ for MT vehicle (LV_IS)					
IP_N_DIF_MAX_IS_0	8x7	0...1FE0H	0 ... 8160	1	rpm
LDPM_VS__IP_N_DIF_MAX_IS	8	0...FFH	0...255	1	Km/h
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis before "Idle Speed" for AT vehicle (LV_IS) : SWI_CC = 0					
IP_N_DIF_MAX_IS_1	8x7	0...1FE0H	0 ... 8160	1	rpm
LDPM_VS__IP_N_DIF_MAX_IS	8	0...FFH	0...255	1	Km/h
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis before "Idle Speed" for AT vehicle (LV_IS) : SWI_CC ≠ 0					
IP_T_MIN_PU_TCU	1	0...FFFFH	0...655.35	0.01	s
LDP_N_32__IP_T_MIN_PU_TCU	6	0...FFH	0...8160	32	rpm
Minimum delay time of PUC activation after entry of PU by TCU request					
C_N_DIF_FAC_CC	1	0...FFH	0...0.996	0.0039	-
Multiplicative factor for N_MIN_PU with converter lockup switch active					

5.2.5 Engine operating state : " Trailing Throttle Fuel Cut Off " (LV_PUC)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PUC	V / O	0...1H	0...1	-	-
engine operating state „trailing throttle fuel cut off "					
CYCNR_HYS_PUC	V	0...FFFFH	0...65535	1	Seg
cycle counter within PUC					
N_MIN_PUC	V/O	0...FFH	0...8160	32	rpm
actual valid minimum engine speed for PUC					
N_HYS_PUC	V/O	0...FFH	0...8160	32	rpm
actual valid engine speed hysteresis for PUC					

Input data:

LV_CT	N_32	LV_RLY_ACCOUT	ID_N_MIN_PUC__TCO__GR_MT
TCO	LV_AT	LV_IS	ID_N_MIN_PUC__TCO__GR_AT
LV_PU	LV_HOM_ACT	GR_AT	ID_N_ACCOUT_MIN_PUC__TCO__GR_AT
LV_PL	C_N_DIF_MAX_IS	N_MIN_PU	ID_N_ACCOUT_MIN_PUC__TCO__GR_MT
N	LV_MSR_ACT	GR_MT	LV_PUC_INH_TEMP_CAT
N_SP_IS	LV_PUC_LOCK_TNT	LV_PUC_SA_INH	LV_ERR_ISA_i
N_GRD	LV_DT	LV_BLS	NC_ETC_CONF
SWI_CC	LV_N_INC_TCU_ACT		

5.2.5.1 Minimum engine speed for LV_PUC detection

Application conditions:

Recurrence: segment syncronous; only once per 10 ms

Description:

The engine speed threshold for the detection of LV_PUC is derived from a characteristic as a function of coolant temperature (TCO) and gear (GR_MT; GR_AT):
ID_N_MIN_PUC__TCO__GR_MT/AT (LV_RLY_ACCOUT = 0, SWI_CC = 0)

During converter clutch lock up activation, the engine speed threshold for the detection of LV_PUC is derived from

ID_N_MIN_PUC_CC__TCO__GR_AT (LV_RLY_ACCOUT =0, SWI_CC <> 0)

There is a special characteristic for air conditioning compressor active (LV_RLY_ACCOUT = 1) and also function of coolant temperature (TCO) and gear (GR_MT; GR_AT):
ID_N_ACCOUT_MIN_PUC__TCO__GR_MT/AT (LV_RLY_ACCOUT =1, SWI_CC = 0)

During converter clutch lock up activation with air conditioning compressor active, the engine speed threshold for the detection of LV_PUC is derived from

ID_N_ACCOUT_MIN_PUC_CC__TCO__GR_AT (LV_RLY_ACCOUT =1, SWI_CC <> 0)

5.2.5.2 LV_PUC Hysteresis

Application conditions:

Recurrence: Segment

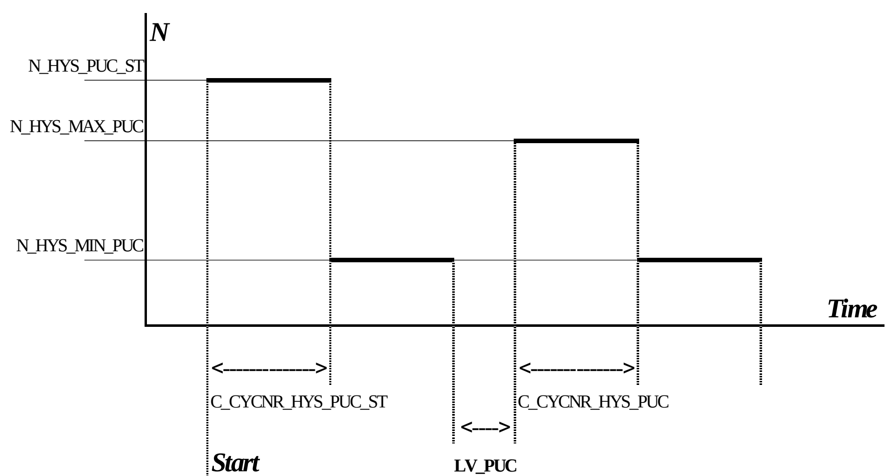
Description:

During Trailing Throttle Fuel Cut Off (LV_PUC = 1), the engine speed hysteresis is continuously calculated to a higher value **ID_N_HYS_MAX_PUC__GR_MT/AT (in case of SWI_CC = 0) or ID_N_HYS_MAX_PUC_LOCK_UP__GR_AT(in case of SWI_CC ≠ 0)** depending on status for converter lockup clutch(SWI_CC), and a cycle counter is loaded with an adjustable number, **C_CYCNR_HYS_PUC**.

The cycle counter is decremented at intervals of a crank angle segment (CRK) in any engine operating state except Trailing Throttle Fuel Cut Off (LV_PUC = 1). If LV_N_INC_TCU_ACT = 1 -> 0 or SWI_CC is converted to 0 from another state with cycle counter decremented, the cycle counter is set to 0 at once.

After the counter has reached its lower limit, the engine speed hysteresis is continuously calculated to the lower value **ID_N_HYS_MIN_PUC__GR_MT/AT(in case of SWI_CC = 0) or ID_N_HYS_MIN_PUC_LOCK_UP__GR_AT(in case of SWI_CC ≠ 0)** depending on status for converter lockup clutch(SWI_CC). This measure is to prevent the injection system from being switched on and off continuously, which is undesirable.

If the coolant temperature is below the threshold **C_TCO_MIN_PUC_ST** when the engine is started, then this engine speed hysteresis is increased to an adjustable value **ID_N_HYS_PUC_ST__GR_MT/AT**, for an adjustable number of cycles **C_CYCNR_HYS_PUC_ST** (This results in faster catalyst heat-up during trailing throttle operation). Otherwise the value ID_N_HYS_MAX_PUC__GR_MT/AT (in case of SWI_CC = 0) or ID_N_HYS_MAX_PUC_LOCK_UP__GR_AT (in case of SWI_CC ≠ 0) is used for C_CYCNR_HYS_PUC cycles



N_HYS_PUC calculation

(1) For engine state transition ES -> ST

```
If      TCO < C_TCO_MIN_PUC_ST
then
        CYCNR_HYS_PUC = C_CYCNR_HYS_PUC_ST
        if      LV_AT = 1
        then
                N_HYS_PUC = ID_N_HYS_PUC_ST__GR_AT
        else
                N_HYS_PUC = ID_N_HYS_PUC_ST__GR_MT
        endif
else
        CYCNR_HYS_PUC = C_CYCNR_HYS_PUC
        If      LV_AT = 1
        then
                If      SWI_CC = 0
                then
                        N_HYS_PUC = ID_N_HYS_MAX_PUC__GR_AT
                else
                        N_HYS_PUC = ID_N_HYS_MAX_PUC_LOCK_UP__GR_AT
                endif
        else
                N_HYS_PUC = ID_N_HYS_MAX_PUC__GR_MT
        endif
endif
```

(2) For every segment

```
If      LV_PUC = 1
then
        CYCNR_HYS_PUC = C_CYCNR_HYS_PUC
        If      LV_AT = 1
        then
                If      SWI_CC = 0
                then
                        N_HYS_PUC = ID_N_HYS_MAX_PUC__GR_AT
                else
                        N_HYS_PUC = ID_N_HYS_MAX_PUC_LOCK_UP__GR_AT
                endif
        else
                N_HYS_PUC = ID_N_HYS_MAX_PUC__GR_MT
        endif
else
        If      CYCNR_HYS_PUC = 0
        then
                if      LV_AT = 1
                then
                        If      SWI_CC = 0
                        then
```

N_HYS_PUC = ID_N_HYS_MIN_PUC__GR_AT
else
N_HYS_PUC = ID_N_HYS_MIN_PUC_LOCK_UP__GR_AT
endif
else
N_HYS_PUC = ID_N_HYS_MIN_PUC__GR_MT
endif
else
CYCNR_HYS_PUC is decremented.
If **(** SWI_CC$_{(n-1)}$ <> 0 **and** SWI_CC$_{(n)}$ = 0 **)**
; to update N_HYS_PUC for SWI_CC$_{(n)}$ *changing condition*
or **(** LV_N_INC_TCU_ACT$_{(n-1)}$=1 **and** LV_N_INC_TCU_ACT$_{(n)}$= 0 **)**
; to update N_HYS_PUC from SWI_CC <> map when N increased by TCU request
then
CYCNR_HYS_PUC = 0
else
no action
endif
endif
endif

5.2.5.3 Deactivation of LV_PUC

Recurrence: segment syncronous; only once per 10 ms

1) **Exit to LV_IS :** Idle Speed
a) **[** LV_CT = 1
b) **and** LV_MSR_ACT = 0
c) **and** N < N_MIN_PU
d) **and** **If** LV_ERR_ISA_1 = 1 or LV_ERR_ISA_2 = 1
or LV_ERR_TPS_PLAUS = 1 or LV_ERR_TPS = 1
then N_32 **< max. (**C_N_MIN_PUC_DIAG; ID_N_ACCOUT_MIN_PUC__TCO__GR_MT(or GR_AT)**)**
else N_32 **<** ID_N_MIN_PUC__TCO__GR_MT(or GR_AT)
(SWI_CC = 0 and LV_RLY_ACCOUT = 0)
or N_32 **<** ID_N_ACCOUT_MIN_PUC__TCO__GR_MT(or GR_AT)
(SWI_CC = 0 and LV_RLY_ACCOUT = 1)
or N_32 **<** ID_N_MIN_PUC_CC__TCO__GR_AT
(SWI_CC <> 0 and LV_RLY_ACCOUT = 0)
or N_32**<**ID_N_ACCOUT_MIN_PUC_CC__TCO__GR_AT
(SWI_CC <> 0 and LV_RLY_ACCOUT = 1)
e) **and** LV_N_INC_TCU_ACT = 0 **]**
f) **or** LV_N_INC_TCU_ACT = 1
(Deactivation of PUC to increase engine speed requested from TCU)

2) **Exit to LV_PU :** Trailing Throttle

a) LV_CT = 1
b) **and** LV_MSR_ACT = 0
c) **and** N >= N_MIN_PU
d) **and** d1) N_32 < N_MIN_PUC
d2) **or** LV_PUC_INH_TEMP_CAT = 1
d3) **or** LV_PUC_LOCK_TNT = 1
d4) **or** LV_PUC_SA_INH = 1
d5) **or** (N_32 **<** (N_MIN_PUC + min(N_HYS_PUC, IP_N_MIN_PUC_OFS)) **and** (LV_BLS = 1 **or** (**not** NC_ETC_CONF)) **and** LV_DT = 1)
e) **and** LV_N_INC_TCU_ACT = 0

3) **Exit to LV_PL:** Part Load

a) LV_CT = 0
or LV_MSR_ACT = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CYCNR_HYS_PUC	1	0...FFFFH	0...65535	1	Seg
Cycle counter to reset the LV_PUC engine speed hysteresis					
C_CYCNR_HYS_PUC_ST	1	0...FFFFH	0...65535	1	Seg
Cycle counter to reset the LV_PUC engine speed hysteresis in cold condition					
C_TCO_MIN_PUC_ST	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature to select the LV_PUC engine speed hysteresis					
ID_N_HYS_MAX_PUC__GR_AT	7	0...FFH	0...8160	32	rpm
LDPM_GR_AT__N_HYS_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis set entering LV_PUC					
ID_N_HYS_MAX_PUC__GR_MT	6	0...FFH	0...8160	32	rpm
LDPM_GR_MT__N_HYS_PUC	6	0...FFH	0...255	1	-
Engine speed hysteresis set entering LV_PUC					
ID_N_HYS_MAX_PUC_LOCK_UP__GR_AT	7	0...FFH	0...8160	32	rpm
LDPM_GR_AT__N_HYS_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis set entering LV_PUC (in case of SWI_CC ≠ 0)					
ID_N_HYS_MIN_PUC__GR_AT	7	0...FFH	0...8160	32	rpm
LDPM_GR_AT__N_HYS_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis reset when C_CYCNR_HYS_PUC_xx achieved					
ID_N_HYS_MIN_PUC__GR_MT	6	0...FFH	0...8160	32	rpm
LDPM_GR_MT__N_HYS_PUC	6	0...FFH	0...255	1	-
Engine speed hysteresis reset when C_CYCNR_HYS_PUC_xx achieved					
ID_N_HYS_MIN_PUC_LOCK_UP__GR_AT	7	0...FFH	0...8160	32	rpm
LDPM_GR_AT__N_HYS_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis reset when C_CYCNR_HYS_PUC_xx achieved(in case of SWI_CC ≠ 0)					
ID_N_HYS_PUC_ST__GR_AT	7	0...FFH	0...8160	32	rpm
LDPM_GR_AT__N_HYS_PUC	7	0...FFH	0...255	1	-
Engine speed hysteresis set entering LV_PUC in cold condition					
ID_N_HYS_PUC_ST__GR_MT	6	0...FFH	0...8160	32	rpm
LDPM_GR_MT__N_HYS_PUC	6	0...FFH	0...255	1	-
Engine speed hysteresis set entering LV_PUC in cold condition					
ID_N_MIN_PUC__TCO__GR_MT	6x6	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_MT__N_MIN_PUC	6	0...FFH	0...255	1	-
LV_PUC engine speed threshold					
ID_N_ACCOUT_MIN_PUC__TCO__GR_MT	6x6	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_MT__N_MIN_PUC	6	0...FFH	0...255	1	-
LV_PUC engine speed threshold with air conditioning compressor active					
ID_N_MIN_PUC__TCO__GR_AT	6x7	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
LV_PUC engine speed threshold					
ID_N_ACCOUT_MIN_PUC__TCO__GR_AT	6x7	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
LV_PUC engine speed threshold with air conditioned compressor active					
C_N_MIN_PUC_DIAG	1	0...FFH	0...8160	32	rpm
Engine speed threshold for trailing throttle fuel cut off limp home.					
IP_N_MIN_PUC_OFS	4	0...FFH	0...8160	32	rpm
LDP_N_GRD_N_MIN_PUC_OFS	4	0...FFH	-4096...4064	32	[rpm/s]
LV_PUC offset to avoid possible stall situation					
ID_N_MIN_PUC_CC__TCO__GR_AT	6x7	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
LV_PUC engine speed threshold with converter lockup switch active					

ID_N_ACCOUT_MIN_PUC_CC__TCO__GR _AT	6x7	0...FFH	0...8160	32	rpm
LDPM_TCO__N_MIN_PUC	6	0...FEH	-48...142.5	0.75	°C
LDPM_GR_AT__N_MIN_PUC	7	0...FFH	0...255	1	-
LV_PUC engine speed threshold with air conditioning compressor & converter lockup switch active					

5.2.6 Logical variable trailing trottle fuel cut-off request LV_PUC_REQ

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PUC_REQ	V / O	0...1 H	0...1	1	-
Logical variable trailing throttle fuel cut off request					

Input data:

LV_PU	LV_PUC		

General information:

LV_PUC_REQ is set at trailing throttle conditions. It is evaluated in module “Minimum torque at clutch”.

Activation: at every engine state

Recurrence: segment-syncronous; only once per 10 ms

Formula section:

LV_PUC_REQ = LV_PU or LV_PUC

5.3 Administration of calculation optimization

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_SEG_HALF	V/O	0 ... FH	0 ... 15	1	-
Half segment counter					
STATE_CLC_RED	V/O	0 ... FFH	0 ... 255	1	-
Calculation reduction state for runtime optimization					
CTR_CYL_NR_ST_CLC	V/O	0 ... 7H	0 ... 7	1	-
Number of first cylinder in calculation order					
CTR_CYL_NR_STOP_CLC	V/O	0 ... 7H	0 ... 7	1	-
Number of last cylinder in calculation order					
CTR_CBK_IN_NR_ST_CLC	V/O	0 ... 1H	1 ... 2	1	-
Start number of intake bank for calculations					
CTR_CBK_IN_NR_STOP_CLC	V/O	0 ... 1H	1 ... 2	1	-
Stop number of intake bank for calculations					
CTR_CBK_EX_NR_ST_CLC	V/O	0 ... 1H	1 ... 2	1	-
Start number of exhaust bank for calculations					
CTR_CBK_EX_NR_STOP_CLC	V/O	0 ... 1H	1 ... 2	1	-
Stop number of exhaust bank for calculations					

Input data:

NC_CYL_NR	LV_ST_END	N_32	SEG_NR
NC_IN_REF	NC_LAMB_REF	NC_CYL_NR	

FUNCTION DESCRIPTION:

General information:

This module is used for administration of runtime reduction and calculation optimization steps.

Pay Attention! This module has to be placed in the operating system immediately after the calculation of N_32 and SEG_NR, which are calculated by the ENSD aggregate. All aggregates, which are affected by calculation optimization, have to be calculated afterwards.

5.3.1 Half segment counter

Application conditions :

Activation : every engine state

Deactivation : -

Initialization: at reset or at every transition of LV_ES from 0 -> 1 do:
#if (NC_USE_SEG_HALF_AVL = 1)
#then CTR_SEG_HALF = (2 * NC_CYL_NR) - 1
#else CTR_SEG_HALF = (2 * NC_CYL_NR) - 2
#endif

Recurrence : #if (NC_USE_SEG_HALF_AVL = 1)
#then half segment synchronous
#else segment synchronous
#endif

This chapter describes the behavior of the half segment loop counter. If a half segment trigger is available by the ECU system the counter is incremented by one each half segment event. If only a segment trigger is available, the counter is incremented by two each segment event. Every 720° CRK the counter is set back to zero.

Application hint:

If a half segment trigger is available by the ECU system, set NC_USE_SEG_HALF_AVL to 1, else set it to 0 and insert the task in the segment synchronous environment.

Formula Section :

#if (NC_USE_SEG_HALF_AVL = 1)

#then

$$CTR\_SEG\_HALF_{(n)} = (CTR\_SEG\_HALF_{(n-1)} + 1)\ \mathbf{MODULO}\ (2 * NC\_CYL\_NR)$$

#else

$$CTR\_SEG\_HALF_{(n)} = (CTR\_SEG\_HALF_{(n-1)} + 2)\ \mathbf{MODULO}\ (2 * NC\_CYL\_NR)$$

#endif

5.3.2 Runtime optimization

Application conditions :

Activation : every engine state

Deactivation : -

Initialization: at reset or at every transition of LV_ES from 0 -> 1 do:

STATE_CLC_RED = NC_STATE_CLC_RED_SEG_1

CTR_CYL_NR_ST_CLC = 0

CTR_CYL_NR_STOP_CLC = (NC_CYL_NR - 1)

CTR_CBK_IN_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_IN_NR_STOP_CLC = NC_CBK_IN_NR

CTR_CBK_EX_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_EX_NR_STOP_CLC = NC_CBK_EX_NR

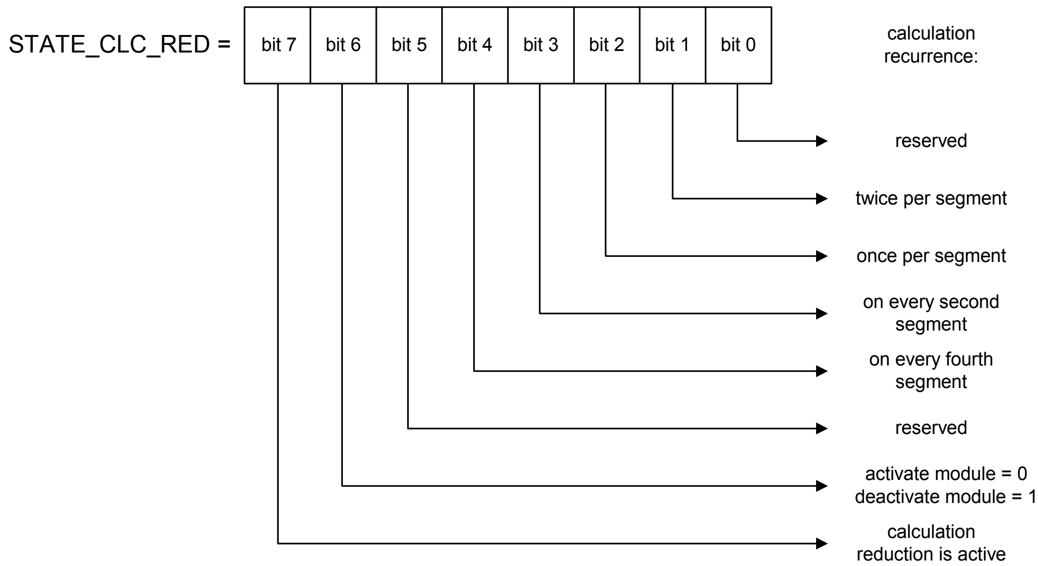
Recurrence : #if (NC_USE_SEG_HALF_AVL = 1)
#then half segment synchronous
#else segment synchronous
#endif

This chapter describes runtime optimization steps. First step is to reduce cylinder dependent calculations from all cylinders to one cylinder, dependent on the engine speed. An additional step is to reduce bank dependent calculations to the intake or exhaust bank, which is allocated to the related cylinder by NC_IN_REF and NC_LAMB_REF.

For engine charge optimization a half segment calculation recurrence is introduced, which depends on an engine speed threshold. Below this threshold MAF, MFF and TI calculation will be updated twice per segment (half segment calculation). Because of that a higher engine dynamic can be realized especially for low engine speeds.

Formula Section :

5.3.2.1 Definition of STATE_CLC_RED



5.3.2.1.1 Definitions of the non calibrateable constants to generate STATE_CLC_RED

NC_STATE_CLC_RED_SEG_HALF	= 0x02H
NC_STATE_CLC_RED_SEG_1	= 0x04H
NC_STATE_CLC_RED_SEG_2	= 0x08H
NC_STATE_CLC_RED_SEG_4	= 0x10H
NC_STATE_CLC_RED_CLC_DEAC	= 0x40H
NC_STATE_CLC_RED_ACT	= 0x80H
NC_STATE_CLC_RED_MASK_1	= 0x1EH
NC_STATE_CLC_RED_MASK_2	= 0xC0H

5.3.2.2 Check runtime optimization conditions

(1) IF SEG_NR = 0

(1) THEN (check for calculation recurrence change only at segment number 0)

(2) IF N_32 > C_N_32_THD_CLC_RED

(2) THEN (runtime reduction has to be activated)

STATE_CLC_RED = NC_STATE_CLC_RED_SEG_2

OR NC_STATE_CLC_RED_ACT ... bitwise

(2) ELSE

(3) IF N_32 < C_N_32_THD_CLC_INC

(3) THEN (half segment calculation has to be activated)
#if (NC_USE_SEG_HALF_AVL = 1)

#then

STATE_CLC_RED = NC_STATE_CLC_RED_SEG_HALF

#else

STATE_CLC_RED = NC_STATE_CLC_RED_SEG_1

#endif

(3) ELSE (check for deactivation conditions)

(4) IF

(((STATE_CLC_RED **AND** NC_STATE_CLC_RED_SEG_2) = NC_STATE_CLC_RED_SEG_2 ... bitwise

AND

N_32 < (C_N_32_THD_CLC_RED - C_N_32_HYS_CLC_RED)**)**

OR

((STATE_CLC_RED **AND** NC_STATE_CLC_RED_SEG_HALF) = NC_STATE_CLC_RED_SEG_HALF ... bitwise

AND

N_32 > (C_N_32_THD_CLC_INC + C_N_32_HYS_CLC_INC)**)**

)

(4) THEN

STATE_CLC_RED = NC_STATE_CLC_RED_SEG_1
(runtime reduction will be deactivated)

(4) ENDIF

(3) ENDIF

(2) ENDIF

(1) ENDIF

5.3.2.3 Determine cylinder and bank numbers for follow up calculations

(3) IF **(**STATE_CLC_RED **AND** NC_STATE_CLC_RED_SEG_2) = NC_STATE_CLC_RED_SEG_2 ... bitwise

(3) THEN

STATE_CLC_RED = STATE_CLC_RED
ExOR NC_STATE_CLC_RED_CLC_DEAC ... bitwise

CTR_CYL_NR_ST_CLC = ID_CTR_CYL_NR_CLC
(SEG_NR is input data for ID_CTR_CYL_NR_CLC)

CTR_CYL_NR_STOP_CLC = CTR_CYL_NR_ST_CLC

(4) IF Bit related to cylinder CTR_CYL_NR_ST_CLC of NC_IN_REF = 1

(4) THEN

CTR_CBK_IN_NR_ST_CLC = 2 (physical meaning)

CTR_CBK_IN_NR_STOP_CLC = 2 (physical meaning)

(4) ELSE

CTR_CBK_IN_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_IN_NR_STOP_CLC = 1 (physical meaning)

(4) ENDIF

(5) IF Bit related to cylinder CTR_CYL_NR_ST_CLC of NC_LAMB_REF = 1

(5) THEN

CTR_CBK_EX_NR_ST_CLC = 2 (physical meaning)

CTR_CBK_EX_NR_STOP_CLC = 2 (physical meaning)

(5) ELSE

CTR_CBK_EX_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_EX_NR_STOP_CLC = 1 (physical meaning)

(5) ENDIF

(3) ELSE

CTR_CYL_NR_ST_CLC = 0

CTR_CYL_NR_STOP_CLC = (NC_CYL_NR - 1)

CTR_CBK_IN_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_IN_NR_STOP_CLC = NC_CBK_IN_NR (physical meaning)

CTR_CBK_EX_NR_ST_CLC = 1 (physical meaning)

CTR_CBK_EX_NR_STOP_CLC = NC_CBK_EX_NR (physical meaning)

(3) ENDIF

Application hint

ID_CTR_CYL_NR_CLC describe the cylinder number, which will be calculated exclusive on the segment with the number SEG_NR, even if the runtime reduction is active. Therefor you have to specify the segment cylinder relation here proper to your engine type. For example (8 cyl. engine):

SEG_NR Segment number	ID_CTR_CYL_NR_CLC Exclusive calculated cylinder
0	3
1	4
2	5
3	6
4	7
5	0
6	1
7	2

Note that the related intake and exhaust banks will be calculated automatically by NC_IN_REF and NC_LAMB_REF.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_CTR_CYL_NR_CLC	NC_CYL_NR	0...7H	0...7	1	-
LDP_SEG_NR_ID_CTR_CYL_NR_CLC	NC_CYL_NR	0...7H	0...7	1	-
Cylinder and segment relation for runtime reduction					
C_N_32_THD_CLC_RED	1	0...FFH	0...8160	32	rpm
Engine speed threshold for activation of runtime reduction					
C_N_32_HYS_CLC_RED	1	0...3FH	0...2016	32	rpm
Engine speed hysteresis for deactivation of runtime reduction					
C_N_32_THD_CLC_INC	1	0...FFH	0...8160	32	rpm
Engine speed threshold for activation of half segment calculation					
C_N_32_HYS_CLC_INC	1	0...3FH	0...2016	32	rpm
Engine speed hysteresis for deactivation of half segment calculation					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_USE_SEG_HALF_AVL	1	0 ... 1H	0 .. 1	1	-
Compiler switch to indicate, that a half segment trigger is available					
NC_STATE_CLC_RED_SEG_HALF	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that the calculation of a module is done twice per segment					
NC_STATE_CLC_RED_SEG_1	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that the calculation of a module is done on every segment					
NC_STATE_CLC_RED_SEG_2	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that the calculation of a module is done on every second segment					
NC_STATE_CLC_RED_SEG_4	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that the calculation of a module is done on every fourth segment					
NC_STATE_CLC_RED_CLC_DEAC	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that the calculation of a module has not been done at next activation call					
NC_STATE_CLC_RED_ACT	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that calculation reduction is active					
NC_STATE_CLC_RED_MASK_1	1	0 ... FFH	0 ..255	1	-
Mask to isolate calculation recurrence information					
NC_STATE_CLC_RED_MASK_2	1	0 ... FFH	0 ..255	1	-
Mask to isolate activation information					

5.4 Full Load Detection

Output data:

Name	Mod.	Hex. Limit	Phys. Limit	Resol.	Unit
LV_FL	V/O	0...1H	0...1		-
Logical bit which indicates the full load request					
LV_FL_RAW	V/O	0...1H	0...1		-
Full load because of pedal value detected					

Input data:

PV_AV	N_32	TCO	VS
FAC_TQ_REQ_CLU	LV_FCUT_IND	LV_N_MAX	LV_LAM_LIM_MIN_i
LV_LAM_LIM_MAX_i			

Functions description:

General information:

The primary task of the module “Full load detection” is to detect the full load request of the driver. The indicator for the full load request is the logical variable LV_FL which sets the trigger for the mixture enrichment in module “Full load enrichment”.

The driver request for full load enrichment is represented by the logical variable LV_FL. If the actual pedal value PV_AV exceeds a certain calibrateable threshold, LV_FL_RAW is set to one. The threshold, which is derived by the map ID_PV_AV_FL, can vary with engine speed.

Certain conditions must be fullfilled to set the full load request LV_FL regardless of the drivers demand.

Application conditions:

Initialisierung: with reset LV_FL_RAW = 0
LV_FL = 0

Recurrence: 20 ms

Activation: all engine operating states

Formula:

If(1) PV_AV **>** ID_PV_AV_FL
or FAC_TQ_REQ_CLU > IP_FAC_TQ_REQ_CLU_FL
then(1) LV_FL_RAW = 1
else (1) **if(2)** PV_AV < ID_PV_AV_FL - C_PV_AV_HYS_FL
or FAC_TQ_REQ_CLU < IP_FAC_TQ_REQ_CLU_FL
-C_FAC_TQ_REQ_CLU_HYS_FL
then(2) LV_FL_RAW = 0
endif(2)
endif(1)

If(1) LV_FL_RAW = 1
and N_32 **>=** C_N_MIN_FL
and TCO **>=** C_TCO_MIN_FL
and VS **>=** C_VS_MIN_FL
and LV_FCUT_IND = 0
and LV_N_MAX = 0
and LV_LAM_LIM_MIN_i = 0
and LV_LAM_LIM_MAX_i = 0
then(1) LV_FL = 1
else(1) LV_FL = 0
endif(1)

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_PV_AV_HYS_FL	1	0...FFH	0...99.6%	0.3906	%
pedal value hysteresis for full load detection					
C_N_MIN_FL	1	0...FFH	0...8160	32	min^{-1}
minimal engine speed for FL					
C_TCO_MIN_FL	1	0...FEH	-48...142.5	0.75	°C
minimal cooling temperature for FL					
C_VS_MIN_FL	1	0...FFH	0...255	1	km/ h
minimal vehicle speed for FL					
ID_PV_AV_FL	8*1	0...FFH	0...99.6%	0.3906	%
LDP_N__PV_AV_FL	8	0...1FE0H	0...8160	1	1/min
pedal value threshold for the determination of LV_FL_RAW					
IP_FAC_TQ_REQ_CLU_FL	1x6	0...FFFFH	0...1.999969	3.052E-5	-
LDP_N_32__IP_FAC_TQ_REQ_CLU_FL	6	0...FFH	0...8160	32	rpm
Threshold on FAC_TQ_REQ_CLU for FL detection					
C_FAC_TQ_REQ_CLU_HYS_FL	1	0...FFFFH	0...1.999969	3.052E-5	-
Hysterisis for LV_FL detection for FAC_TQ_REQ_CLU parameter					

5.5 Auxilary functions for engine states

5.5.1 Statistical data for fleet monitoring

General information:

The variables defined in this module should give a general overview about the vehicle usage during a monitoring period.

5.5.1.1 Number of starts

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_CST_LOW_ALTI	O/V/S	0...FFH	0...255	1	[-]
Number cold starts at low altitude					
CTR_CST_HIGH_ALTI	O/V/S	0...FFH	0...255	1	[-]
Number of cold starts at high altitude					
CTR_WST	O/V/S	0...FFH	0...255	1	[-]
Number of warm starts					
CTR_HST_LOW_ALTI	O/V/S	0...FFH	0...255	1	[-]
Number of hot starts at low altitude					
CTR_HST_HIGH_ALTI	O/V/S	0...FFH	0...255	1	[-]
Number of hot starts at high altitude					

Input data:

LV_ST_END	TCO	AMP_AD	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volatile memory lost:

all Variables = 0

- otherwise: restored from non-volatile memory

Recurrence: *once at exit start (LV_ST_END = 0 -> 1)*

Activation: -

Deactivation: -

Formula section:

```
IF      TCO < C_TCO_MAX_CST
THEN IF     AMP_AD > C_AMP_AD_MAX_ST_HIGH_ALTI
        THEN IF     CTR_CST_LOW_ALTI < 255 (FFH)
                THEN CTR_CST_LOW_ALTI(n) = CTR_CST_LOW_ALTI(n-1) + 1
                ELSE CTR_CST_LOW_ALTI = 255 (FFH)
        ELSE IF     CTR_CST_HIGH_ALTI < 255 (FFH)
                THEN CTR_CST_HIGH_ALTI(n) = CTR_CST_HIGH_ALTI(n-1) + 1
                ELSE CTR_CST_HIGH_ALTI = 255 (FFH)
ELSE IF     TCO < C_TCO_MIN_HST
        THEN IF     CTR_WST < 255 (FFh)
                THEN CTR_WST(n) = CTR_WST(n-1) + 1
                ELSE CTR_WST = 255 (FFH)
        ELSE IF     AMP_AD > C_AMP_AD_MAX_ST_HIGH_ALTI
                THEN IF     CTR_HST_LOW_ALTI < 255 (FFH)
                        THEN CTR_HST_LOW_ALTI(n) = CTR_HST_LOW_ALTI(n-1) + 1
                        ELSE CTR_HST_LOW_ALTI = 255 (FFH)
                ELSE IF     CTR_HST_HIGH_ALTI < 255 (FFH)
                        THEN CTR_HST_HIGH_ALTI(n) = CTR_HST_HIGH_ALTI(n-1) + 1
                        ELSE CTR_HST_HIGH_ALTI = 255 (FFH)
        ENDIF
ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MAX_CST	1	0...FEH	-48...142.5	0.75	°C
Maximum TCO for cold start					
C_TCO_MIN_HST	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO for hot start					
C_AMP_AD_MAX_ST_HIGH_ALTI	1	0...FFH	0...5434	21.31	hPa
Maximum ambient pressure to detect start at high altitude					

5.5.1.2 Timer for engine run

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_ERU_IS_60	O/V	0...3CH	0...60	1	[s]
Timer for engine running in idle (for T_ERU_IS calculation)					
T_ERU_IS	O/V/S	0...FFFFH	0...65535	1	[min]
Time while engine was running in idle					
T_ERU_PL_60	O/V	0...3CH	0...60	1	[s]
Timer for engine running in part load (for T_ERU_PL calculation)					
T_ERU_PL	O/V/S	0...FFFFH	0...65535	1	[min]
Time while engine was running in part load					
T_ERU_FL_60	O/V	0...3CH	0...60	1	[s]
Timer for engine running in full load (for T_ERU_FL calculation)					
T_ERU_FL	O/V/S	0...FFFFH	0...65535	1	[min]
Time while engine was running in full load					

Input data:

LV_ES	LV_IS	LV_PL	LV_FL

Application conditions:

Initialisation: - at first ECU power up and non-volatile memory lost:

T_ERU_IS = T_ERU_PL_= T_ERU_FL = 0

- otherwise: restored from non-volatile memory

- at LV_IGK 0->1 and reset:

T_ERU_IS_60 = T_ERU_PL_60_= T_ERU_FL_60 = 0

Recurrence: *1s*

Activation: LV_ES = 0

Deactivation: -

Formula section:

Time engine run in idle

IF LV_IS = 1

THEN $T\_ERU\_IS\_60_{(n)} = T\_ERU\_IS\_60_{(n-1)} + 1$

IF T_ERU_IS_60 = 60

THEN T_ERU_IS_60 = 0

IF T_ERU_IS < 65535 (FFFFH)

THEN $T\_ERU\_IS_{(n)} = T\_ERU\_IS_{(n-1)} + 1$

ELSE T_ERU_IS = 65535 (FFFFH)

ENDIF

Time engine run in part load

IF LV_PL = 1 **and** LV_FL = 0

THEN $T\_ERU\_PL\_60_{(n)} = T\_ERU\_PL\_60_{(n-1)} +1$

  IF T_ERU_PL_60 = 60

  THEN T_ERU_PL_60 = 0

    IF T_ERU_PL < 65535 (FFFFH)

    THEN $T\_ERU\_PL_{(n)} = T\_ERU\_PL_{(n-1)} +1$

    ELSE T_ERU_PL = 65535 (FFFFH)

ENDIF

Time engine run in full load

IF LV_FL = 1

THEN $T\_ERU\_FL\_60_{(n)} = T\_ERU\_FL\_60_{(n-1)} +1$

  IF T_ERU_FL_60 = 60

  THEN T_ERU_FL_60 = 0

    IF T_ERU_FL < 65535 (FFFFH)

    THEN $T\_ERU\_FL_{(n)} = T\_ERU\_FL_{(n-1)} +1$

    ELSE T_ERU_FL = 65535 (FFFFH)

ENDIF

5.5.1.3 Timer for vehicle speed conditions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_VS_L_RNG_60	O/V	0...3CH	0...60	1	[s]
Timer for vehicle driven in low speed range (for T_VS_L_RNG calculation)					
T_ VS_L_RNG	O/V	0...FFFFH	0...65535	1	[min]
Time while vehicle was vehicle driven in low range					
T_VS_M_RNG_60	O/V	0...3CH	0...60	1	[s]
Timer for vehicle driven in medium speed range (for T_VS_M_RNG calculation)					
T_ VS_M_RNG	O/V	0...FFFFH	0...65535	1	[min]
Time while vehicle was vehicle driven in medium range					
T_VS_H_RNG_60	O/V	0...3CH	0...60	1	[s]
Timer for vehicle driven in high speed range (for T_VS_H_RNG calculation)					
T_ VS_H_RNG	O/V	0...FFFFH	0...65535	1	[min]
Time while vehicle was vehicle driven in high range					

Input data:

VS	LV_ERR_MWSS		

Application conditions:

Initialisation: at LV_IGK 0->1 and reset:

all variables = 0

Recurrence: 1s

Activation: VS > 0 km/h

Deactivation: LV_ERR_MWSS = 1

Formula section:

Time vehicle driven in low vehicle speed range

IF VS ≤ C_VS_L_RNG

THEN $T\_VS\_L\_RNG\_60_{(n)} = T\_VS\_L\_RNG\_60_{(n-1)} + 1$

IF T_VS_L_RNG_60 = 60

THEN T_VS_L_RNG_60 = 0

IF T_VS_L_RNG < 65535 (FFFFH)

THEN $T\_VS\_L\_RNG_{(n)} = T\_VS\_L\_RNG_{(n-1)} + 1$

ELSE T_VS_L_RNG = 65535 (FFFFH)

ENDIF

Time vehicle driven in medium vehicle speed range

IF C_VS_L_RNG < VS < C_VS_H_RNG

THEN $T\_VS\_M\_RNG\_60_{(n)} = T\_VS\_M\_RNG\_60_{(n-1)} + 1$

IF T_VS_M_RNG_60 = 60

THEN T_VS_M_RNG_60 = 0

IF T_VS_M_RNG < 65535 (FFFFH)

THEN $T\_VS\_M\_RNG_{(n)} = T\_VS\_M\_RNG_{(n-1)} + 1$

ELSE T_VS_M_RNG = 65535 (FFFFH)

ENDIF

Time vehicle driven in high vehicle speed range

IF VS >= C_VS_H_RNG

THEN $T\_VS\_H\_RNG\_60_{(n)} = T\_VS\_H\_RNG\_60_{(n-1)} + 1$

IF T_VS_H_RNG_60 = 60

THEN T_VS_H_RNG_60 = 0

IF T_VS_H_RNG < 65535 (FFFFH)

THEN $T\_VS\_H\_RNG_{(n)} = T\_VS\_H\_RNG_{(n-1)} +1$

ELSE T_VS_H_RNG = 65535 (FFFFH)

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_L_RNG	1	0...FFh	0...255	1	km/h
Vehicle speed threshold for low range detection					
C_VS_H_RNG	1	0...FFh	0...255	1	km/h
Vehicle speed threshold for high range detection					

5.5.1.4 Time after all readyness bits set

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_ALL_DIAG_READY	O/V/S	0...FFFFH	0...65535	1	[min]
Time since readyness is detected for all diagnosis					

Input data:

LV_READY_XX	C_ERR_CLAS_XX		

Application conditions:

Initialisation: - at first ECU power up, non-volantile memory lost and failure memory clear:

T_ALL_DIAG_READY = 0

- otherwise: restored from non-volantile memory

Recurrence: 60s

Activation: LV_DC = 1

Deactivation: -

Formula section:

For XX = all available Diagnosic instances

IF LV_READY_XX = 0 **or**

C_ERR_CLAS_XX = 00h

THEN **IF** T_ALL_DIAG_READY < 65535 (FFFFH)

THEN $T\_ALL\_DIAG\_READY_{(n)} = T\_ALL\_DIAG\_READY_{(n-1)} +1$

ELSE T_ALL_DIAG_READY = 65535 (FFFFH)

ENDIF

5.5.1.5 Time with no failure present

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_NO_ERR_PRES_60	O/V	0...3CH	0...60	1	s
Timer for engine operating with no error present (for T_NO_ERR_PRES calculation)					
T_NO_ERR_PRES	O/V	0...FFFFH	0...65535	1	min
Timer while engine was running with no error present					

Input data:

LV_ES	ERR_SYM_XX	LV_ERR_XX	

Application conditions:

Initialisation: at LV_IGK 0->1 and reset:

T_NO_ERR_PRES_60 = T_NO_ERR_PRES = 0

Recurrence: 1s

Activation: LV_ES = 0

Deactivation: -

Formula section:

For XX = all Diagnostic instances present in intermittent failure memory
(ERR_INTM_DIAG_INST)

IF LV_ERR_XX = 0 **and**
ERR_SYM_XX = 0

THEN $T\_NO\_ERR\_PRES\_60_{(n)} = T\_NO\_ERR\_PRES\_60_{(n-1)} + 1$

IF T_NO_ERR_PRES_60 = 60

THEN T_NO_ERR_PRES_60 = 0

IF T_NO_ERR_PRES < 65535 (FFFFH)

THEN $T\_NO\_ERR\_PRES_{(n)} = T\_NO\_ERR\_PRES_{(n-1)} + 1$

ELSE T_NO_ERR_PRES = 65535 (FFFFH)

ENDIF

5.5.2 Data Frame at engine stall event or unexpected accel.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ENG_STALL_SAVE_END	O/V	0...01H	0...1	1	-
Data storage after engine stall finished					
LV_AC_SPO_SAVE_END	O/V	0...01H	0...1	1	-
Data storage after unexpected acceleration finished					
LV_ENG_STALL	O/V	0...01H	0...1	1	-
Engine stall detected					
LV_AC_SPO	O/V	0...01H	0...1	1	-
Unexpected acceleration detected					
AC_VEH	O/V	8000...7FFFH	-15.72...15.72	0.00048	m/s^2
Vehicle acceleration					
T_INH_AST_ENG_STALL	O/V	0...FFH	0...51.0	0.2	s
Inhibition time after start for engine stall detection					
CTR_IDX_HIS	O	0...FFH	0...255	1	-
Index counter for data history					
LV_LST_DATA_SAVE	O	0...01H	0...1	1	-
Flag to indicate last data stored					
DATA_SAVE_CASE	O/V/S	0H 1H 2H	NO_SAVE ENG_STALL AC_SPO	-	-
Reason of data storage					
SAVE_DIST	O/V/S	0... FFFFFFFFH	0...429496729,5	0.1	km
Distance at data storage					
FAC_LAM_COR_CMN	-	8000...7FFFH	-100...100	0.00305	%
Common FAC_LAM_COR for lin and bin system					
HIS_N[8]	O/V	0...1FE0H	0...8160	1	rpm
History-Table of Engine speed					
HIS_TCO[8]	O/V	0...FEH	-48...142.5	0.75	°C
History-Table of Coolant temperature					
HIS_TPS[8]	O/V	0..FFH	0...119.5	0.47	°TPS
History-Table of opening angle of the throttle valve					
HIS_VB[8]	O/V	0...FFH	0...26V	0.102	V
History-Table of Battery voltage					
HIS_TIA[8]	O/V	0...FEH	-48...142.5	0.75	°C
History-Table of Air intake temperature					
HIS_VLS_UP[8]	O/V	0...3FFH	0...4.99512	0.004882	V
History-Table of Upstream oxygen sensor voltage					
HIS_VLS_DOWN[8]	O/V	0...3FFH	0...4.99512	0.004882	V
History-Table of Downstream oxygen sensor voltage					
HIS_FAC_LAM_COR_CMN[8]	O/V	8000...7FFFH	-100...100	0.00305	%
History-Tableof limited lambda controller output plus pre control correction					
HIS_FAC_LAM_AD[8]	O/V	8000...7FFFH	-50...49.99847	0001526	%
History-Table of fuel mass setpoint factor of lambda adaptation					
HIS_MFF_ADD_LAM_AD[8]	O/V	8000.7FFFH	-694.51...694.48	0.021195	mg/stk
History-Table of fuel mass setpoint offset					
HIS_MAF[8]	O/V	0...FFFFH	0...1389	0.021195	mg/stk
History-Table of Air mass (predicted)					
HIS_MAF_MES[8]	O/V	0...FFFFH	0...1389	0.021195	mg/stk
History-Table of Air mass flow					
HIS_MAP_MES_BAS[8]	O/V	0...FFFFH	0...5434	0.0829	hPa
History-Table of Intake manifold pressure per segment measured					
HIS_LV_RLY_EFP[8]	O/V	0...01H	0...1	-	-
History-Table of Fuel pump relay control					
HIS_VS[8]	O/V	0...FFH	0...255	1	km/h
History-Table of vehicle speed					

HIS_AMP_AD[8]	O/V	0...FFFFH	0...5434	0.00829	hPa
History-Table of Ambient pressure (measured or adapted)					
HIS_IGA_AV_MV[8]	O/V	0...FFH	-35.625...60	0.375	°CRK
History-Table of ignition angle on all cylinder					
HIS_TI_1_0[8]	O/V	0...FFFFH	0...262.14	0.004	ms
History-Table of cylinder individual injection time					
HIS_AR_RED_DIF_REL[8]	O/V	8000.7FFFH	-50...49.998	0.001525	%
History-Table of correction of reduced throttle area					
HIS_CPPWM[8]	O/V	0...7FFFH	0...99.99	0.0031	%
History-Table of PWM signal for CPS opening					
HIS_CL_MMV[8]	O/V	0...FFFFH	0...8	0.000122	-
History-Table of canister load					
HIS_LV_CS[8]	O/V	0...01H	0...1	-	-
History-Table of Clutch switch					
HIS_CTR_DMF[8]	O/V	0...FFH	0...255	1	-
History-Table of counter for engine speed oscillation detection					
HIS_LV_ENG_OFF_DMF_IGN[8]	O/V	0...01H	0...1	-	-
History-Table of ignition shut off request because of DMF oscillation					
HIS_LV_ENG_OFF_DMF_INJ[8]	O/V	0...01H	0...1	-	-
History-Table of injection shut off request because of DMF oscillation					
HIS_LV_ENG_OFF_DMF_TPS_ISA[8]	O/V	0...01H	0...1	-	-
History-Table of TPS/ISA shut off request because of DMF oscillation					
HIS_ECU_RST_STATUS_HW[8]	O/V	0...FFFFH	0...65535	-	-
History-Table of HW Reset Status					
HIS_ECU_RST_STATUS_SW_1[8]	O/V	0...FFFFH	0...65535	-	-
History-Table of SW Reset Status (Bit0..15)					
HIS_ECU_RST_STATUS_SW_2[8]	O/V	0...FFFFH	0...65535	-	-
History-Table of ECU Reset Status (Bit 16...31)					
HIS_LV_ IGN_INJ_LOCK_REQ[8]	O/V	0...01H	0...1	1	-
History-Table of Request to lock ignition and/or injection due to backwards rotation detection					
HIS_LV_IMOB_INJ_OFF[8]	O/V	0...01H	0...1	1	-
History-Table of prohibition of injection by Immoblizer					
HIS_LV_IMOB_IGC_OFF[8]	O/V	0...01H	0...1	1	-
History-Table of prohibition of ignition by Immoblizer					
HIS_TQI_ASR_REQ[8]	O/V	0...7FFFH	0...1023.97	0.03125	Nm
History-Table of Torque reduction request for ASR from TCS					
HIS_TQI_MSR_REQ[8]	O/V	0...7FFFH	0...1023.97	0.03125	Nm
History-Table of Torque reduction request for ASR from TCS					
HIS_TQI_ASR_SLW_REQ[8]	O/V	0...7FFFH	0...1023.97	0.03125	Nm
History-Table of Torque reduction request for ASR from TCS (slow path)					
HIS_TQI_GS_REQ[8]	O/V	0...7FFFH	0...1023.97	0.03125	Nm
History-Table of torque reduction request for gear-shift from TCU					
HIS_ SPK_RTD_TCU[8]	O/V	17H...67H	-15...+15	0.375	°CRK
History-Table of torque intervention during gear-shift from TCU					
HIS_OBD_TPS_AV_1[8]	O/V	0...FFH	0...100	0.392157	%
History-Table of throttle opening poti 1					
HIS_OBD_TPS_AV_2[8]	O/V	0...FFH	0...100	0.392157	%
History-Table of throttle opening poti 2					
HIS_OBD_PV_1[8]	O/V	0...FFH	0...100	0.392157	%
History-Table of pedal activation poti 1					
HIS_OBD_PV_2[8]	O/V	0...FFH	0...100	0.392157	%
History-Table of pedal activation poti 2					
HIS_ISAPWM_ISA[8]	O/V	0...FFFFH	0...99.9985	0.0015	%
History-Table of applied idle speed actuator duty cycle **(only available in case of NC_ETC_CONF = 0)**					
HIS_TPS_SP[8]	O/V	0...3FFFH	0...119.5	0.0073	°TPS
History-Table of throttle position setpoint **(only available in case of NC_ETC_CONF = 1)**					
HIS_LV_OFF_IV_MON[8]	O/V	0...01H	0...1	1	-
History-Table of request to disable IV power-stage**(only available in case of NC_ETC_CONF = 1)**					
HIS_LAMB_LS_UP[8]	O/V	0...7FFFH	0...31.99902	0.009766	-

History-Table of lambda signal of WRAF sensor **(only available in case of NC_STATE_LSL_UP_IF = 1)**					
HIS_ERR_INTM_DIAG_INST_ACT[8]	O/V	0... FFH	0..255	1	-
History-Table of latest entry of intermittent failure memory					
SAVE_N[10]	O/V/S	0...1FE0H	0...8160	1	rpm
Saved History-Table of Engine speed					
SAVE_TCO[10]	O/V/S	0...FEH	-48...142.5	0.75	°C
Saved History-Table of Coolant temperature					
SAVE_TPS[10]	O/V/S	0..FFH	0...119.5	0.47	°TPS
Saved History-Table of opening angle of the throttle valve					
SAVE_VB[10]	O/V/S	0...FFH	0...26V	0.102	V
Saved History-Table of Battery voltage					
SAVE_TIA[10]	O/V/S	0...FEH	-48...142.5	0.75	°C
Saved History-Table of Air intake temperature					
SAVE_VLS_UP[10]	O/V/S	0...3FFH	0...4.99512	0.004882	V
Saved History-Table of Upstream oxygen sensor voltage					
SAVE_VLS_DOWN[10]	O/V/S	0...3FFH	0...4.99512	0.004882	V
Saved History-Table of Downstream oxygen sensor voltage					
SAVE_FAC_LAM_COR_CMN[10]	O/V/S	8000...7FFFH	-100...100	0.00305	%
Saved History-Table of limited lambda controller output plus pre control correction					
SAVE_FAC_LAM_AD[10]	O/V/S	8000...7FFFH	-50...49.99847	0001526	%
Saved History-Table of fuel mass setpoint factor of lambda adaptation					
SAVE_MFF_ADD_LAM_AD[10]	O/V/S	8000.7FFFH	-694.51...694.48	0.021195	mg/stk
Saved History-Table of fuel mass setpoint offset					
SAVE_MAF[10]	O/V/S	0...FFFFH	0...1389	0.021195	mg/stk
Saved History-Table of Air mass (predicted)					
SAVE_MAF_MES[10]	O/V/S	0...FFFFH	0...1389	0.021195	mg/stk
Saved History-Table of Air mass flow					
SAVE_MAP_MES_BAS[10]	O/V/S	0...FFFFH	0...5434	0.0829	hPa
Saved History-Table of Intake manifold pressure per segment measured					
SAVE_LV_RLY_EFP[10]	O/V/S	0...01H	0...1	-	-
Saved History-Table of Fuel pump relay control					
SAVE_VS[10]	O/V/S	0...FFH	0...255	1	km/h
Saved History-Table of vehicle speed					
SAVE_AMP_AD[10]	O/V/S	0...FFFFH	0...5434	0.00829	hPa
Saved History-Table of Ambient pressure (measured or adapted)					
SAVE_IGA_AV_MV[10]	O/V/S	0...FFH	-35.625...60	0.375	°CRK
Saved History-Table of ignition angle on all cylinder					
SAVE_TI_1_0[10]	O/V/S	0...FFFFH	0...262.14	0.004	ms
Saved History-Table of cylinder individual injection time					
SAVE_AR_RED_DIF_REL[10]	O/V/S	8000.7FFFH	-50...49.998	0.001525	%
Saved History-Table of correction of reduced throttle area					
SAVE_CPPWM[10]	O/V/S	0...7FFFH	0...99.99	0.0031	%
Saved History-Table of PWM signal for CPS opening					
SAVE_CL_MMV[10]	O/V/S	0...FFFFH	0...8	0.000122	-
Saved History-Table of canister load					
SAVE_LV_CS[10]	O/V/S	0...01H	0...1	-	-
Saved History-Table of Clutch switch					
SAVE_CTR_DMF[10]	O/V/S	0...FFH	0...255	1	-
Saved History-Table of counter for engine speed oscillation detection					
SAVE_LV_ENG_OFF_DMF_IGN[10]	O/V/S	0...01H	0...1	-	-
Saved History-Table of ignition shut off request because of DMF oscillation					
SAVE_LV_ENG_OFF_DMF_INJ[10]	O/V/S	0...01H	0...1	-	-
Saved History-Table of injection shut off request because of DMF oscillation					
SAVE_LV_ENG_OFF_DMF_TPS_ISA[10]	O/V/S	0...01H	0...1	-	-
Saved History-Table of TPS/ISA shut off request because of DMF oscillation					
SAVE_ECU_RST_STATUS_HW[10]	O/V/S	0...FFFFH	0...65535	-	-
Saved History-Table of HW Reset Status					
SAVE_ECU_RST_STATUS_SW_1[10]	O/V/S	0...FFFFH	0...65535	-	-
Saved History-Table of SW Reset Status (Bit0..15)					

SAVE_ECU_RST_STATUS_SW_2[10]	O/V/S	0...FFFFH	0...65535	-	-
Saved History-Table of ECU Reset Status (Bit 16...31)					
SAVE_LV_IGN_INJ_LOCK_REQ[10]	O/V/S	0...01H	0...1	1	-
Saved History-Table of Request to lock ignition and/or injection due to backwards rotation detection					
SAVE_LV_IMOB_INJ_OFF[10]	O/V/S	0...01H	0...1	1	-
Saved History-Table of prohibition of injection by Immoblizer					
SAVE_LV_IMOB_IGC_OFF[10]	O/V/S	0...01H	0...1	1	-
Saved History-Table of prohibition of ignition by Immoblizer					
SAVE_TQI_ASR_REQ[10]	O/V/S	0...7FFFH	0...1023.97	0.03125	Nm
Saved History-Table of Torque reduction request for ASR from TCS					
SAVE_TQI_MSR_REQ[10]	O/V/S	0...7FFFH	0...1023.97	0.03125	Nm
Saved History-Table of Torque reduction request for ASR from TCS					
SAVE_TQI_ASR_SLW_REQ[10]	O/V/S	0...7FFFH	0...1023.97	0.03125	Nm
Saved History-Table of Torque reduction request for ASR from TCS (slow path)					
SAVE_TQI_GS_REQ[10]	O/V/S	0...7FFFH	0...1023.97	0.03125	Nm
Saved History-Table of torque reduction request for gear-shift from TCU					
SAVE_SPK_RTD_TCU[10]	O/V/S	17H...67H	-15...+15	0.375	°CRK
Saved History-Table of torque intervention during gear-shift from TCU					
SAVE_OBD_TPS_AV_1[10]	O/V/S	0...FFH	0...100	0.392157	%
Saved History-Table of throttle opening poti 1					
SAVE_OBD_TPS_AV_2[10]	O/V/S	0...FFH	0...100	0.392157	%
Saved History-Table of throttle opening poti 2					
SAVE_OBD_PV_1[10]	O/V/S	0...FFH	0...100	0.392157	%
Saved History-Table of pedal activation poti 1					
SAVE_OBD_PV_2[10]	O/V/S	0...FFH	0...100	0.392157	%
Saved History-Table of pedal activation poti 2					
SAVE_ISAPWM_ISA[10]	O/V/S	0...FFFFH	0...99.9985	0.0015	%
Saved History-Table of applied idle speed actuator duty cycle **(only available in case of NC_ETC_CONF = 0)**					
SAVE_TPS_SP[10]	O/V/S	0...3FFFH	0...119.5	0.0073	°TPS
Saved History-Table of throttle position setpoint **(only available in case of NC_ETC_CONF = 1)**					
SAVE_LV_OFF_IV_MON[10]	O/V/S	0...01H	0...1	1	-
Saved History-Table of request to disable IV power-stage **(only available in case of NC_ETC_CONF = 1)**					
SAVE_LAMB_LS_UP[10]	O/V/S	0...7FFFH	0...31.99902	0.009766	-
Saved History-Table of lambda signal of WRAF sensor **(only available in case of NC_STATE_LSL_UP_IF = 1)**					
SAVE_ERR_INTM_DIAG_INST_ACT[10]	O/V/S	0... FFH	0..255	1	-
Saved History-Table of latest entry of intermittent failure memory					

Hint: all History-Tables "HIS_xxx" and Saved History-Tables "SAVE_xxx" are visible as calibration-data in application tool

Input data:

LV_IGK	T_AST	N	N_GRD
NC_ETC_CONF	PV_AV	VS	TCO
TPS	VB	TIA	VLS_UP[NC_CBK_EX_NR]
VLS_DOWN[NC_CBK_EX_NR]	FAC_LAM_COR[NC_CBK_EX_NR]	FAC_LAM_AD[NC_CBK_EX_NR]	MFF_ADD_LAM_AD[NC_CBK_EX_NR]
MAF	MAF_MES	MAP_MES_BAS	LV_RLY_EFP
AMP_AD	IGA_AV_MV	TI_1_x	AR_RED_DIF_REL
CPPWM	CL_MMV	LV_CS	CTR_DMF
LV_ENG_OFF_DMF_IGN	LV_ENG_OFF_DMF_INJ	LV_ENG_OFF_DMF_TPS_ISA	ECU_RST_STATUS_HW
ECU_RST_STATUS_SW_1	ECU_RST_STATUS_SW_2	LV_IGN_INJ_LOCK_REQ	LV_IMOB_INJ_OFF
LV_IMOB_IGC_OFF	TQI_ASR_REQ	TQI_MSR_REQ	TQI_ASR_SLW_REQ
TQI_GS_REQ	SPK_RTD_TCU	OBD_TPS_AV_1	OBD_TPS_AV_2
OBD_PV_1	OBD_PV_2	ISAPWM_ISA	TPS_SP

LV_OFF_IV_MON	LAMB_LS_UP[NC_CBK_EX_NR]	ERR_INTM_DIAG_INST_ACT	NC_STATE_LSL_UP_IF
PV_CRU	LV_ST_END		

FUNCTION DESCRIPTION:

In case of an unexpected engine/vehicle behaviour (stall or self acceleration) environment data should be stored for root cause analysis; For that reason the 8 values before and 2 values after the unexpected event should be avaiable.

Application conditions:

Initialisation: - at first ECU power up or non-volatile memory lost:

all saved variables = 0

- otherwise: all saved variables restored from NVMY

- at reset or LV_IGK 0 ->1:

T_INH_AST_ENG_STALL = C_T_AST_MIN_ENG_STALL

all others: 0

Recurrence: 200 ms

Activation: LV_IGK = 1 **and**

LV_ENG_STALL_SAVE_END = 0 **and**

LV_AC_SPO_SAVE_END = 0

Deactivation: *otherwise*

Formula section:

If T_INH_AST_ENG_STALL > 0

Then **If** LV_ST_END = 1

Then $T\_INH\_AST\_ENG\_STALL_{(n)} = T\_INH\_AST\_ENG\_STALL_{(n-1)} – 0.2s$

Endif

Endif

IF T_INH_AST_ENG_STALL = 0

and N < C_N_MAX_ENG_STALL

THEN LV_ENG_STALL = 1

ENDIF

$AC\_VEH = AC\_VEH + (((VS_n – VS_{n-1}) / 200ms) – AC\_VEH) * C\_FAC\_AC\_VEH\_MMV$

IF N > C_N_MIN_AC_SPO

and N_GRD > C_N_GRD_MIN_AC_SPO

and ((NC_ETC_CONF = 0 **and** PV_AV < C_PV_AV_MAX_AC_SPO) **or**

(NC_ETC_CONF = 1 **and** PV_CRU < C_PV_AV_MAX_AC_SPO))

and AC_VEH > C_AC_MIN_AC_SPO

THEN LV_AC_SPO = 1

ELSE LV_AC_SPO = 0

IF NC_STATE_LSL_UP_IF = 1 *(linear upstream sensor system)*

THEN FAC_LAM_COR_CMN = FAC_LAM_COR

ELSE FAC_LAM_COR_CMN = FAC_LAM_COR (different phys. resolution!!!!)

ENDIF

Engine data history

For XXX = see table below

Dependance:

always	N	TCO	TPS	VB
	TIA	VLS_UP	VLS_DOWN	FAC_LAM_COR_CMN
	FAC_LAM_AD	MFF_ADD_LAM_AD	MAF	MAF_MES
	MAP_MES_BAS	LV_RLY_EFP	VS	AMP_AD
	IGA_AV_MV	TI_1_0	AR_RED_DIF_REL	CPPWM
	CL_MMV	LV_CS	CTR_DMF	LV_ENG_OFF_DMF_IGN
	LV_ENG_OFF_DMF_INJ	LV_ENG_OFF_DMF_TPS_ISA	ECU_RST_STATUS_HW	ECU_RST_STATUS_SW_1
	ECU_RST_STATUS_SW_2	LV_IGN_INJ_LOCK_REQ	LV_IMOB_INJ_OFF	LV_IMOB_IGC_OFF
	TQI_ASR_REQ	TQI_MSR_REQ	TQI_ASR_SLW_REQ	TQI_GS_REQ
	SPK_RTD_TCU	OBD_TPS_AV_1	OBD_TPS_AV_2	OBD_PV_1
	OBD_PV_2	ERR_INTM_DIAG_INST_ACT		
NC_ETC_CONF = 0	ISAPWM_ISA			
NC_ETC_CONF = 1	TPS_SP	LV_OFF_IV_MON		
NC_STATE_LSL_UP_IF = 1	LAMB_LS_UP			

IF LV_ENG_STALL = 0

and LV_AC_SPO = 0

THEN HIS_XXX[CTR_IDX_HIS] = XXX

CTR_IDX_HIS$_{(n)}$ = CTR_IDX$_{(n-1)}$ +1

IF CTR_IDX_HIS > 7

THEN CTR_IDX HIS = 0

ENDIF

ENDIF

Data storage:

(common for engine-stall and unexpected accleration; distinction possible by use of value "DATA_SAVE_CASE")

IF (LV_ENG_STALL = 1 **or** LV_AC_SPO = 1)

and LV_LST_DATA_SAVE = 0

THEN SAVE_XXX[8] = XXX

SAVE_DIST = DIST

LV_LST_DATA_SAVE = 1

FOR CTR_IDX_SAVE = 0 to 7 **DO**: (copy information)

SAVE_XXX[CTR_IDX_SAVE] = HIS_XXX[(CTR_IDX_HIS + CTR_IDX_SAVE) mod 8]

/ see information below for "mod" calculation

ENDIF

ELSE IF LV_ENG_STALL = 1 and LV_LST_DATA_SAVE = 1

THEN SAVE_XXX[9] = XXX

DATA_SAVE_CASE = 1h *(ENG_STALL)*

LV_ENG_STALL_SAVE_END = 1

ELSE IF LV_AC_SPO = 1 **and** LV_LST_DATA_SAVE = 1

THEN SAVE_XXX[9] = XXX

DATA_SAVE_CASE = 2h *(AC_SPO)*

LV_AC_SPO_SAVE_END = 1

ENDIF

In computing, the **modulo** operation finds the remainder of division of one number by another.

Given two numbers, *a* (the dividend) and *m* (the divisor), *a* modulo *m* (abbreviaties as *a* mod *m*) is the remainder, on division of *a* by *m*. For instance the expression "7 mod 3" would evaluate to 1.

$$(a \bmod m) = a - \left\lfloor \frac{a}{m} \right\rfloor \cdot m$$

The floor function returns the integer result of the division a/m.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_AST_MIN_ENG_STALL	1	0...FFH	51.0	0.2	s
Minimum time after start to detect engine stall					
C_N_MAX_ENG_STALL	1	1FE0H	0...8160	1	rpm
Threshold for engine stall detection					
C_FAC_AC_VEH_MMV	1	0...FFFFH	0...1	1.5E-5	-
Filtering factor for AC_VEH					
C_N_MIN_AC_SPO	1	0...1FE0H	0...8160	1	rpm
Minimum engine speed to detect unexpected acceleration					
C_N_GRD_MIN_AC_SPO	1	80..7FH	-4096...4064	32	rpm/s
Minimum engine speed gradient to detect unexpected acceleration					
C_PV_AV_MAX_AC_SPO	1	0...FFH	0...99.6	0.3906	%
Maximum pedal value to detect unexpected acceleration					
C_AC_MIN_AC_SPO	1	8000...7FFFH	-15.72...15.72	0.00048	m/s^2
Minimum acceleration to detect unexpected acceleration					
ID_HIS_N[8]	1x8	0...1FE0H	0...8160	1	rpm
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Engine speed					
ID_HIS_TCO[8]	1x8	0...FEH	-48...142.5	0.75	°C
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Coolant temperature					
ID_HIS_TPS[8]	1x8	0..FFH	0...119.5	0.47	°TPS
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of opening angle of the throttle valve					
ID_HIS_VB[8]	1x8	0...FFH	0...26V	0.102	V
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Battery voltage					
ID_HIS_TIA[8]	1x8	0...FEH	-48...142.5	0.75	°C
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Air intake temperature					
ID_HIS_VLS_UP[8]	1x8	0...3FFH	0...4.99512	0.004882	V
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Upstream oxygen sensor voltage					
ID_HIS_VLS_DOWN[8]	1x8	0...3FFH	0...4.99512	0.004882	V
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Downstream oxygen sensor voltage					
ID_HIS_FAC_LAM_COR_CMN[8]	1x8	8000...7FFFH	-100...100	0.00305	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of limited lambda controller output plus pre control correction					
ID_HIS_FAC_LAM_AD[8]	1x8	8000...7FFFH	-50...49.99847	0001526	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of fuel mass setpoint factor of lambda adaptation					
ID_HIS_MFF_ADD_LAM_AD[8]	1x8	8000.7FFFH	-694.51...694.48	0.021195	mg/stk
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of fuel mass setpoint offset					
ID_HIS_MAF[8]	1x8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Air mass (predicted)					
ID_HIS_MAF_MES[8]	1x8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Air mass flow					
ID_HIS_MAP_MES_BAS[8]	1x8	0...FFFFH	0...5434	0.0829	hPa
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Intake manifold pressure per segment measured					
ID_HIS_LV_RLY_EFP[8]	1x8	0...01H	0...1	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Fuel pump relay control					

ID_HIS_VS[8]	1x8	0...FFH	0...255	1	km/h
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of vehicle speed					
ID_HIS_AMP_AD[8]	1x8	0...FFFFH	0...5434	0.00829	hPa
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Ambient pressure (measured or adapted)					
ID_HIS_IGA_AV_MV[8]	1x8	0...FFH	-35.625...60	0.375	°CRK
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of ignition angle on all cylinder					
ID_HIS_TI_1_0[8]	1x8	0...FFFFH	0...262.14	0.004	ms
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of cylinder individual injection time					
ID_HIS_AR_RED_DIF_REL[8]	1x8	8000.7FFFH	-50...49.998	0.001525	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of correction of reduced throttle area					
ID_HIS_CPPWM[8]	1x8	0...7FFFH	0...99.99	0.0031	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of PWM signal for CPS opening					
ID_HIS_CL_MMV[8]	1x8	0...FFFFH	0...8	0.000122	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of canister load					
ID_HIS_LV_CS[8]	1x8	0...01H	0...1	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Clutch switch					
ID_HIS_CTR_DMF[8]	1x8	0...FFH	0...255	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of counter for engine speed oscillation detection					
ID_HIS_LV_ENG_OFF_DMF_IGN[8]	1x8	0...01H	0...1	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of ignition shut off request because of DMF oscillation					
ID_HIS_LV_ENG_OFF_DMF_INJ[8]	1x8	0...01H	0...1	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of injection shut off request because of DMF oscillation					
ID_HIS_LV_ENG_OFF_DMF_TPS_ISA[8]	1x8	0...01H	0...1	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of TPS/ISA shut off request because of DMF oscillation					
ID_HIS_ECU_RST_STATUS_HW[8]	1x8	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of HW Reset Status					
ID_HIS_ECU_RST_STATUS_SW_1[8]	1x8	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of SW Reset Status (Bit0..15)					
ID_HIS_ECU_RST_STATUS_SW_2[8]	1x8	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of ECU Reset Status (Bit 16...31)					
ID_HIS_LV_ IGN_INJ_LOCK_REQ[8]	1x8	0...01H	0...1	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Request to lock ignition and/or injection due to backwards rotation detection					
ID_HIS_LV_IMOB_INJ_OFF[8]	1x8	0...01H	0...1	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of prohibition of injection by Immoblizer					
ID_HIS_LV_IMOB_IGC_OFF[8]	1x8	0...01H	0...1	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of prohibition of ignition by Immoblizer					
ID_HIS_TQI_ASR_REQ[8]	1x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Torque reduction request for ASR from TCS					
ID_HIS_TQI_MSR_REQ[8]	1x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-

Dummy ID for specification of History-Table of Torque reduction request for ASR from TCS					
ID_HIS_TQI_ASR_SLW_REQ[8]	1x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of Torque reduction request for ASR from TCS (slow path)					
ID_HIS_TQI_GS_REQ[8]	1x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of torque reduction request for gear-shift from TCU					
ID_HIS_SPK_RTD_TCU[8]	1x8	17H...67H	-15...+15	0.375	°CRK
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of torque intervention during gear-shift from TCU					
ID_HIS_OBD_TPS_AV_1[8]	1x8	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of throttle opening poti 1					
ID_HIS_OBD_TPS_AV_2[8]	1x8	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of throttle opening poti 2					
ID_HIS_OBD_PV_1[8]	1x8	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of pedal activation poti 1					
ID_HIS_OBD_PV_2[8]	1x8	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of pedal activation poti 2					
ID_HIS_ISAPWM_ISA[8]	1x8	0...FFFFH	0...99.9985	0.0015	%
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of applied idle speed actuator duty cycle **(only available in case of NC_ETC_CONF = 0)**					
ID_HIS_TPS_SP[8]	1x8	0...3FFFH	0...119.5	0.0073	°TPS
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of throttle position setpoint **(only available in case of NC_ETC_CONF = 1)**					
ID_HIS_LV_OFF_IV_MON[8]	1x8	0...01H	0...1	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of request to disable IV power-stage **(only available in case of NC_ETC_CONF = 1)**					
ID_HIS_LAMB_LS_UP[8]	1x8	0...7FFFH	0...31.99902	0.009766	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of lambda signal of WRAF sensor **(only available in case of NC_STATE_LSL_UP_IF = 1)**					
ID_HIS_ERR_INTM_DIAG_INST_ACT[8]	1x8	0... FFH	0..255	1	-
LDPM_CTR_IDX_HIS	8	0...7H	0...7	1	-
Dummy ID for specification of History-Table of latest entry of intermittent failure memory					
ID_SAVE_N[10]	1X10	0...1FE0H	0...8160	1	rpm
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Engine speed					
ID_SAVE_TCO[10]	1X10	0...FEH	-48...142.5	0.75	°C
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Coolant temperature					
ID_SAVE_TPS[10]	1X10	0..FFH	0...119.5	0.47	°TPS
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of opening angle of the throttle valve					
ID_SAVE_VB[10]	1X10	0...FFH	0...26V	0.102	V
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Battery voltage					
ID_SAVE_TIA[10]	1X10	0...FEH	-48...142.5	0.75	°C
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Air intake temperature					
ID_SAVE_VLS_UP[10]	1X10	0...3FFH	0...4.99512	0.004882	V
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Upstream oxygen sensor voltage					
ID_SAVE_VLS_DOWN[10]	1X10	0...3FFH	0...4.99512	0.004882	V

LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Downstream oxygen sensor voltage					
ID_SAVE_FAC_LAM_COR_CMN[10]	1X10	8000...7FFFH	-100...100	0.00305	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of limited lambda controller output plus pre control correction					
ID_SAVE_FAC_LAM_AD[10]	1X10	8000...7FFFH	-50...49.99847	0001526	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of fuel mass setpoint factor of lambda adaptation					
ID_SAVE_MFF_ADD_LAM_AD[10]	1X10	8000.7FFFH	-694.51...694.48	0.021195	mg/stk
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of fuel mass setpoint offset					
ID_SAVE_MAF[10]	1X10	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Air mass (predicted)					
ID_SAVE_MAF_MES[10]	1X10	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Air mass flow					
ID_SAVE_MAP_MES_BAS[10]	1X10	0...FFFFH	0...5434	0.0829	hPa
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Intake manifold pressure per segment measured					
ID_SAVE_LV_RLY_EFP[10]	1X10	0...01H	0...1	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Fuel pump relay control					
ID_SAVE_VS[10]	1X10	0...FFH	0...255	1	km/h
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of vehicle speed					
ID_SAVE_AMP_AD[10]	1X10	0...FFFFH	0...5434	0.00829	hPa
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Ambient pressure (measured or adapted)					
ID_SAVE_IGA_AV_MV[10]	1X10	0...FFH	-35.625...60	0.375	°CRK
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of ignition angle on all cylinder					
ID_SAVE_TI_1_0[10]	1X10	0...FFFFH	0...262.14	0.004	ms
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of cylinder individual injection time					
ID_SAVE_AR_RED_DIF_REL[10]	1X10	8000.7FFFH	-50...49.998	0.001525	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of correction of reduced throttle area					
ID_SAVE_CPPWM[10]	1X10	0...7FFFH	0...99.99	0.0031	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of PWM signal for CPS opening					
ID_SAVE_CL_MMV[10]	1X10	0...FFFFH	0...8	0.000122	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of canister load					
ID_SAVE_LV_CS[10]	1X10	0...01H	0...1	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Clutch switch					
ID_SAVE_CTR_DMF[10]	1X10	0...FFH	0...255	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of counter for engine speed oscillation detection					
ID_SAVE_LV_ENG_OFF_DMF_IGN[10]	1X10	0...01H	0...1	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of ignition shut off request because of DMF oscillation					
ID_SAVE_LV_ENG_OFF_DMF_INJ[10]	1X10	0...01H	0...1	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of injection shut off request because of DMF oscillation					
ID_SAVE_LV_ENG_OFF_DMF_TPS_ISA[10]	1X10	0...01H	0...1	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of TPS/ISA shut off request because of DMF oscillation					

ID_SAVE_ECU_RST_STATUS_HW[10]	1X10	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of HW Reset Status					
ID_SAVE_ECU_RST_STATUS_SW_1[10]	1X10	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of SW Reset Status (Bit0..15)					
ID_SAVE_ECU_RST_STATUS_SW_2[10]	1X10	0...FFFFH	0...65535	-	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of ECU Reset Status (Bit 16...31)					
ID_SAVE_LV_IGN_INJ_LOCK_REQ[10]	1X10	0...01H	0...1	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Request to lock ignition and/or injection due to backwards rotation detection					
ID_SAVE_LV_IMOB_INJ_OFF[10]	1X10	0...01H	0...1	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of prohibition of injection by Immoblizer					
ID_SAVE_LV_IMOB_IGC_OFF[10]	1X10	0...01H	0...1	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of prohibition of ignition by Immoblizer					
ID_SAVE_TQI_ASR_REQ[10]	1X10	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Torque reduction request for ASR from TCS					
ID_SAVE_TQI_MSR_REQ[10]	1X10	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Torque reduction request for ASR from TCS					
ID_SAVE_TQI_ASR_SLW_REQ[10]	1X10	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of Torque reduction request for ASR from TCS (slow path)					
ID_SAVE_TQI_GS_REQ[10]	1X10	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of torque reduction request for gear-shift from TCU					
ID_SAVE_SPK_RTD_TCU[10]	1X10	17H...67H	-15...+15	0.375	°CRK
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of torque intervention during gear-shift from TCU					
ID_SAVE_OBD_TPS_AV_1[10]	1X10	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of throttle opening poti 1					
ID_SAVE_OBD_TPS_AV_2[10]	1X10	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of throttle opening poti 2					
ID_SAVE_OBD_PV_1[10]	1X10	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of pedal activation poti 1					
ID_SAVE_OBD_PV_2[10]	1X10	0...FFH	0...100	0.392157	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of pedal activation poti 2					
ID_SAVE_ISAPWM_ISA[10]	1X10	0...FFFFH	0...99.9985	0.0015	%
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of applied idle speed actuator duty cycle **(only available in case of NC_ETC_CONF = 0)**					
ID_SAVE_TPS_SP[10]	1X10	0...3FFFH	0...119.5	0.0073	°TPS
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of throttle position setpoint **(only available in case of NC_ETC_CONF = 1)**					
ID_SAVE_LV_OFF_IV_MON[10]	1X10	0...01H	0...1	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of request to disable IV power-stage **(only available in case of NC_ETC_CONF = 1)**					
ID_SAVE_LAMB_LS_UP[10]	1X10	0...7FFFH	0...31.99902	0.009766	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-

Dummy ID for specification of Saved History-Table of lambda signal of WRAF sensor **(only available in case of NC_STATE_LSL_UP_IF = 1)**					
ID_SAVE_ERR_INTM_DIAG_INST_ACT[10]	1X10	0... FFH	0..255	1	-
LDPM_CTR_IDX_SAVE	10	0...9H	0...9	1	-
Dummy ID for specification of Saved History-Table of latest entry of intermittent failure memory					

5.6 Auxiliary start function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CYC_ST	V/O	0...FFH	0...255	1	-
Cycle counter during start if the first injection is applied					
CYC_CAST	V/O	0...FFFFH	0...65535	1	-
Cycle counter during start and after start (reset at exit start to after start)					
T_AST	V/O	0...FFFFH	0...6553.5	0.1	s
Time after exit start to after start					
T_AST_STOP	V/O/S	0...FFFFH	0...6553.5	0.1	s
Time after exit start to after start saved in the non-volatile memory at engine shut down					
LV_STALL	V/O/S	0...1H	0...1	1	-
Flag indicating that the engine stalled with ignition key on					
LV_ST_ES	V/O	0...1H	0...1	1	-
Flag indicating that the engine did not exit start before the engine stopped					
LV_ST_VLD	-	0...1H	0...1	1	-
indicating that the last engine start was ok					

Input data:

LV_ST	LV_ST_END	LV_ES	SUM_INH_INJ
LV_RUN_ENG	LV_ST_INJ_AUTH		

FUNCTION DESCRIPTION:

General information:

The “Auxiliary start function” puts flags, time- and cycle-counters at the other function’s disposal. The flags are necessary to deal with the different ways to shut down the engine. The time- and cycle-counters are used very often, mainly in the warm up and catalyst heating functions but also in the restart function.

5.6.1 “Start break off” detection: LV_ST_ES

Description:

The “Start break off” (LV_ST_ES = 1) is detected if the engine stops without leaving the “Start”. Additionally the cycle counter CYC_ST has to reach a minimum number of cycles C_CYC_ST_MIN_ST_ES. The check of the minimum number of cycles is necessary to avoid a “start break off” detection in case of a to short activation of the starter.
In case of a “Start break off” detection the cycle counter CYC_ST is not reset to zero (see definition of CYC_ST).
The “Start break off”-Bit LV_ST_ES is reset at the next exit from “Start” to “Idle speed” or “Part load” and at reset.

Application conditions:

Initialisation: at transition LV_ST_END = 0 -> 1

(Systemevent: Exit “Start” to “Part load” or “Idle”):

LV_ST_VLD = 1

LV_ST_ES = 0

at transition LV_RUN_ENG = 0 -> 1

(Systemevent: “Engine stopped” to “Engine running”)

LV_ST_VLD = 0

Recurrence: once

Activation: at transition LV_ES = 0 -> 1

(Systemevent: “Engine running” to “Engine stopped”)

Formula section:

If LV_ST_VLD = 0 **and**
CYC_ST ≥ C_CYC_ST_MIN_ST_ES **then**

LV_ST_ES = 1

endif

5.6.2 “Engine stall” detection: LV_STALL

Description:

“Engine stall” is detected very time the engine stops after leaving the start without the request by means of injection cut off (SUM_INH_INJ number of cut of cylinders).
In case of a “Engine stall” detection the fuel mass for the next restart is handled differently (see “Restart function”).
The “Engine stall”-Bit LV_STALL is reset at the next exit from “Start” to “Idle speed” or “Part load” and is stored in the non-volatile memory for the next start.

Application conditions:

Initialisation: at transition LV_ST_END = 0 -> 1

(Systemevent: Exit “Start” to “Part load” or “Idle”):

LV_STALL = 0

Recurrence: once

Activation: at transition LV_ES = 0 -> 1

(Systemevent: “Engine running” to “Engine stopped”)

Formula section:

If LV_ST_VLD = 1 **and**

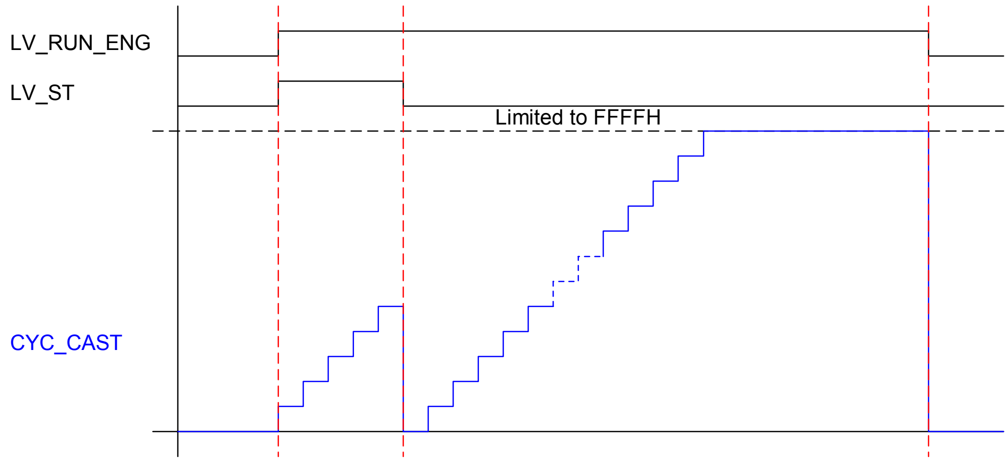
SUM_INH_INJ <= C_NR_MAX_INH_INJ_STALL **then**

LV_STALL = 1

endif

5.6.3 Cycle counter: CYC_CAST

Signal flow diagram:



Description:

The cycle counter CYC_CAST starts counting if the engine is running (LV_RUN_ENG = 1). At exit "Start" and if the engine stops the counter is rest to zero. It is limited to its maximum value.

Application conditions:

Initialisation:	at transition LV_ES = 0 -> 1	(engine stopped) **or**	
	at transition LV_ST = 1 -> 0	(exit start):	CYC_CAST = 0
Recurrence:	every segment		
Activation:	LV_RUN_ENG = 1		
Deactivation:	-		

Formula section:

If CYC_CAST < 65535 (FFFFH) **then**

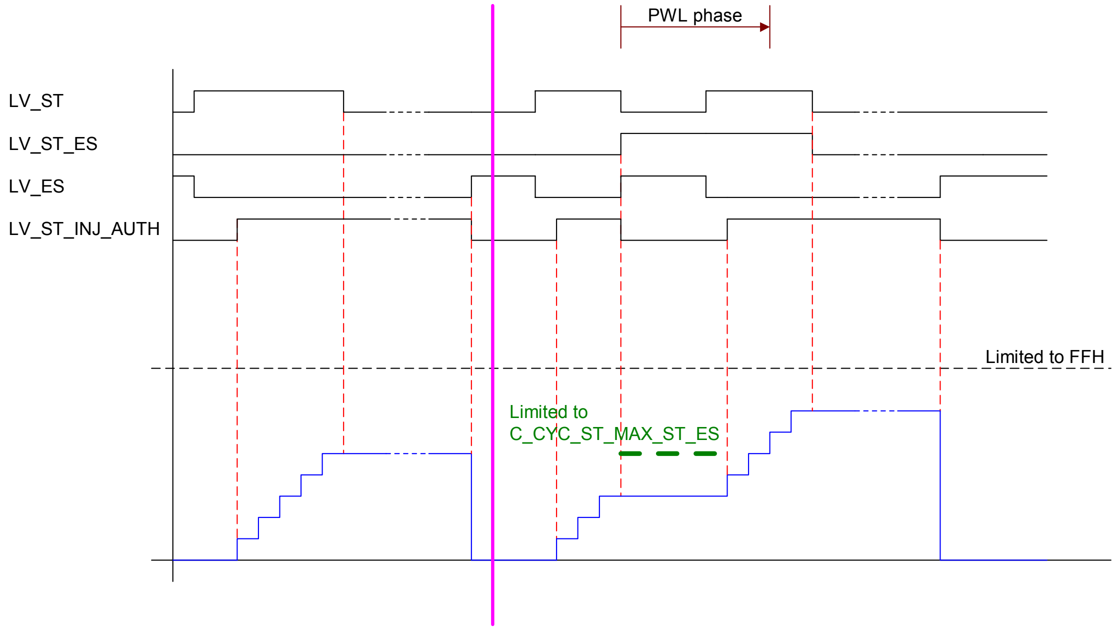
$CYC\_CAST_{n} = CYC\_CAST_{n-1} + 1$

else $CYC\_CAST_{n} = CYC\_CAST_{n-1}$

endif

5.6.4 Start cycle counter: CYC_ST

Signal flow diagram:



Description:

The cycle counter CYC_ST starts counting if the engine is in the start phase and the first injection is authorised (LV_ST_INJ_AUTH = 1). It stops counting at exit “Start”.
If the engines stops and no “Start break off” is detected (LV_ST_ES = 0) CYC_ST is reset to zero. Otherwise CYC_ST stays on its last value or is set to a maximum value C_CYC_ST_MAX_ST_ES if this maximum value is exceeded.

Application conditions:

Initialisation: at transition LV_ES = 0 -> 1

(Systemevent “Engine running” to “Engine stopped”):

If(1) LV_ST_ES = 1 **then**

If(2) CYC_ST > C_CYC_ST_MAX_ST_ES **then**

CYC_ST = C_CYC_ST_MAX_ST_ES

else(2) $CYC\_ST_{n} = CYC\_ST_{n-1}$

endif(2)

else(1) CYC_ST = 0

endif(1)

Recurrence: every segment

Activation: LV_ST = 1

Deactivation: -

Formula section:

If CYC_ST < 255 (FFH) **and**
LV_ST_INJ_AUTH = 1 **then**
$CYC\_ST_{n} = CYC\_ST_{n-1} + 1$
else $CYC\_ST_{n} = CYC\_ST_{n-1}$
endif

5.6.5 Time after start and the saved value after engine stopped: T_AST / T_AST_STOP

Description:

T_AST is a timer, which is started at exit “Start” to “Part load” or ”Idle speed”. It is limited to its maximum value.
If the engine stops the value of T_AST is stored in the non-volatile value T_AST_STOP. Afterwards T_AST is reset to zero.

Application conditions:

Initialisation: at transition LV_ST_END = 1 -> 0 (deactivation)
(“Engine running out of start” to “Engine stopped”):
T_AST_STOP = T_AST
T_AST = 0

Recurrence: 100 ms

Activation: LV_ST_END = 1

Deactivation: -

Formula section:

If T_AST < 6553.5s (FFFFH) **then**
$T\_AST_{n} = T\_AST_{n-1} + 0.1s$
else $T\_AST_{n} = T\_AST_{n-1}$

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CYC_ST_MIN_ST_ES	1	0...FFH	0...255	1	-
Minimum number of cycles necessary to detect “Start break off”					
C_CYC_ST_MAX_ST_ES	1	0...FFH	0...255	1	-
Maximum number of cycles CYC_ST is set to in case of a “Start break off” detection					
C_NR_MAX_INH_INJ_STALL	1	0...FFH	0...255	1	-
Maximum number of cut off cylinders to detect engine stall					

5.7 Catalyst heating coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_CH	O/V	0H 1H 2H	CH_OFF CH_AST CH_L_LOAD	1	-
State of catalyst heating					
TEMP_CAT_DIF_CH_L	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Difference between actual catalyst temperature and deactivation threshold for low load CH					
LV_CH	O/V	0...1H	0...1	1	-
Catalyst Heating Function general					
LV_CH_AST_CDN	V	0...1H	0...1	1	-
Indicates if conditions for after start catalyst heating are fulfilled					
LV_CH_L_CDN	V	0...1H	0...1	1	-
Indicates if conditions for low load catalyst heating are fulfilled					

Input data:

LV_CH_L_ENA	LV_CH_AST_ENA	T_AST_COR_CH	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]
TCO_ST	T_AST	STATE_ENG	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEMP_CAT_THD_CH_AST	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
catalyst temperature threshold for after start catalyst heating					
C_TEMP_CAT_THD_CH_L_OFF	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
catalyst temperature threshold for deactivation of low load CH					
C_TEMP_CAT_THD_CH_L_ON	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
catalyst temperature threshold for activation of low load CH					
C_T_AST_MIN_CH_L_LOAD	1	0...FFFFH	0...6.5535E+3	0.1	s
Minimum time after start for low load heating					
IP_T_AST_THD_CH_AST	8	0...FFFFH	0...6.5535E+3	0.1	s
LDPM_TCO_ST_1_EXTC	8	0...FEH	-48...142.5	0.75	°C
Time after start threshold for catalyst heating after start					

5.7.1 General information

This module activates the catalyst heating in general and makes a distinction between after start and low load catalyst heating.

Application Condition

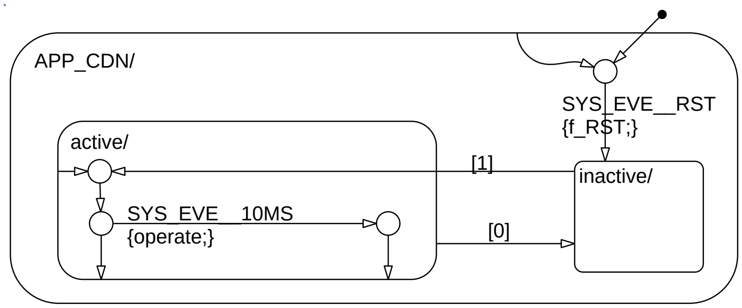


Figure 1 EXTC_ISPCLchc0/ APP_CDN/ Chart

Function Description

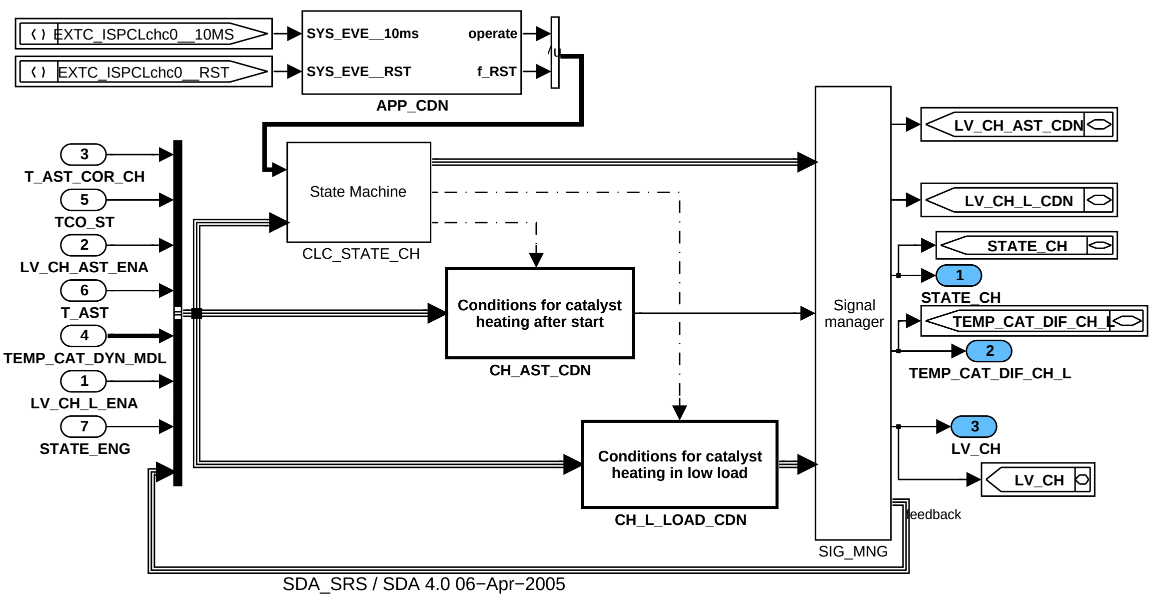


Figure 2 EXTC_ISPCLchc0

5.7.1.1 State Machine

In the passive state 'CH_OFF' first the conditions for after start cataylst heaitng ('CH_AST') are checked –via the function call f_CLC_CH_AST_CDN. If they are not true, conditions for low load heating ('CH_L_LOAD') are checked –via the function call f_CLC_CH_L_CDN. If 'CH_AST' of 'CH_L_LOAD' is active only the conditions for this state are checked.

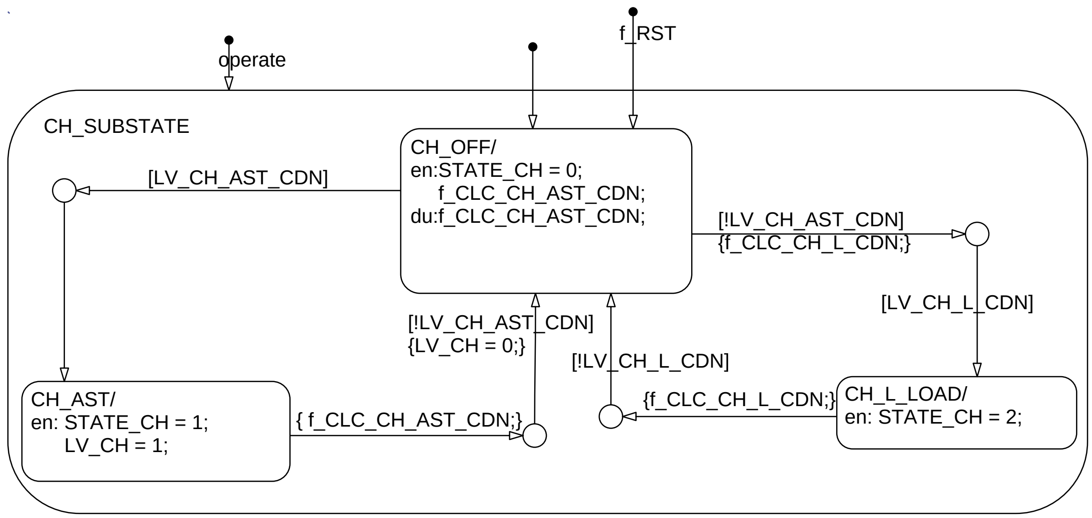


Figure 3 EXTC_ISPCLchc0/ CLC_STATE_CH/ SUB_2

5.7.1.2 Conditions for catalyst heating after start

The map-output is evaluated only as long as the engine is stopped. After that the output will not change anymore. TEMP_CAT_DYN_MDL can only be regarded after the engine runs because the temperature is initialised at system event "engine stopped to engine run".

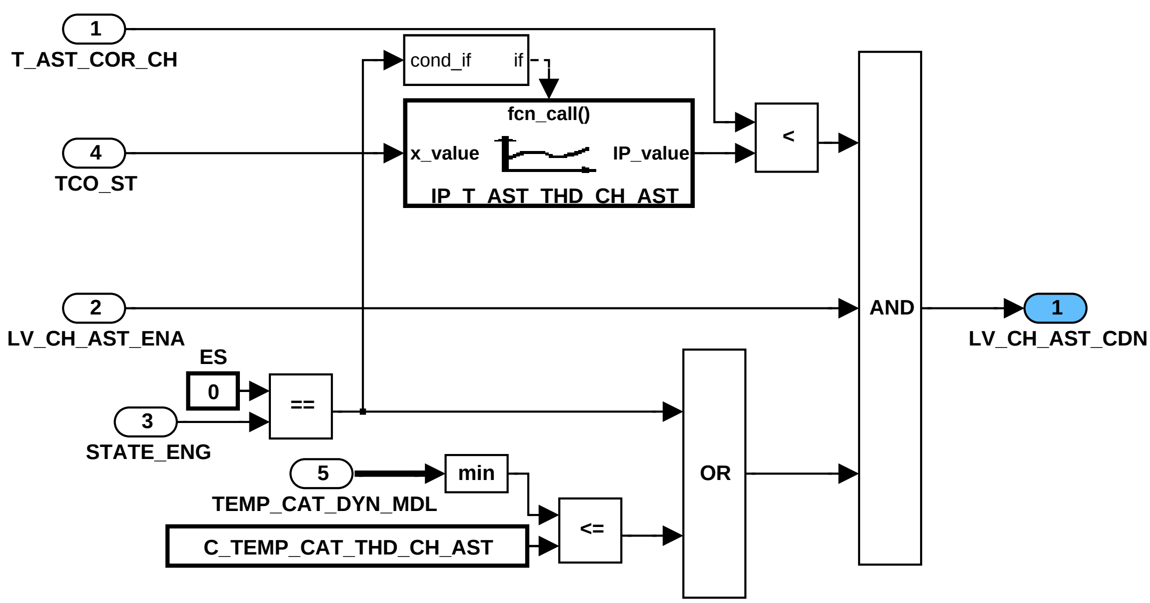


Figure 4 EXTC_ISPCLchc0/ CH_AST_CDN/ SUB

5.7.1.3 Conditions for catalyst heating in low load

The activation of low load catalyst heating depends on the catalyst temperature. TEMP_CAT_DIF_CH_L is the difference between the actual catalyst temperature and a calibratable temperature setpoint which is to be reached through heating measures. The value is negative if the catalyst has to be heated up. Via a T_AST –threshold it is possible to make after start catalyst heating higher prior.

general specification

The activation of low load catalyst heating depends on the catalyst temperature. TEMP_CAT_DIF_CH_L is the difference between the actual catalyst temperature and a calibratable temperature setpoint which is to be reached through heating measures. The value is negative if the catalyst has to be heated up. Via a T_AST –threshold it is possible to make after start catalyst heating higher prior.

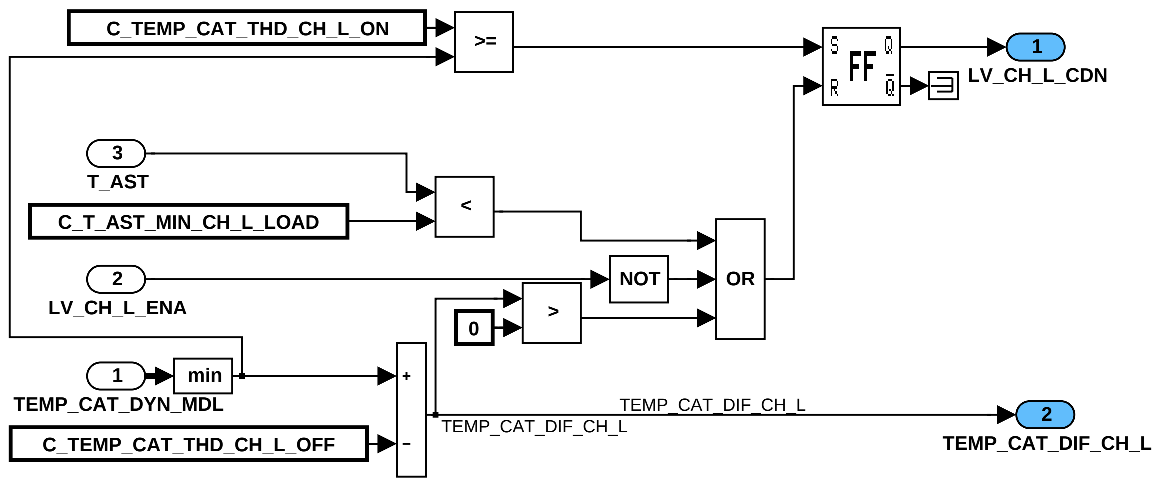


Figure 5 EXTC_ISPCLchc0/ CH_L_LOAD_CDN/ SDA48

5.8 Catalyst heating coordination (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TI_CH	V/O	0...1H	0...1	1	[-]
Catalyst Heating Function "lean injection"					
LV_CH_AST_ENA	V/O	0...1H	0...1	1	[-]
indicates if after start catalyst heating is enabled					
LV_CH_L_ENA	V/O	0...1H	0...1	1	[-]
Inhibition of catalyst heating in low load					
LV_CH_SET_OFF	V	0...1H	0...1	1	[-]
Logical variable cancelling catalyst heating					
MAF_INT_CH	V	0...FFFFH	0...9102.08	0.1388888	[g]
MAF-Integral for Catalyst heating and SAWUP					
T_AST_COR_CH	V/O	0...FFFFH	0...6553.5	0.1	[s]
Catalyst age depending correction of T_AST for after start catalyst heating					

Input data:

LV_ES	LV_DRI	LV_ACIN	VS
TCO	LV_ST	T_AST	T_ES

5.8.1 1s - Function

FUNCTION DESCRIPTION:

LV_CH_SET_OFF serves to cancel catalyst heating (LV_CH) by observing a catalyst temperature equivalent (MAF_INT_CH) and comparing it with a threshold. Additionally a TIA_ST and AMP_AD condition is to be fullfilled to serve drivability and break-booster requirements.

Recurrence: 1 s

Initialisation: MAF_INT_CH = 0 at LV_ES = 1 or LV_ST = 1

LV_CH_SET_OFF = 0 at reset

Application conditions:

Activation: In all engine states except LV_ST and LV_ES.

Deactivation: LV_CH_SET_OFF = 1

Formula section:

$MAF\_INT\_CH_{i} = MAF\_INT\_CH_{i-1} + MAF\_KGH$

Note: MAF_INT_CH and MAF_KGH have different resolutions!

5.8.2 100ms - Function

Recurrency: 100 ms

Initialisation: at transition from START(LV_ST) to PL or IS

LV_CH_AST_ENA = 1

LV_CH_L_ENA = 0

Application conditions:

Activation: all engine states

Formula section:

LV_TI_CH = LV_LAMB_CH

T_AST_COR_CH = T_AST *(function can be added to enlarge catalyst heating time due to catalyst aging)*

If MAF_INT_CH > C_MAF_INT_CH_MAX
or TIA_ST < C_TIA_ST_MIN_CH
or AMP_AD < C_AMP_AD_MIN_CH
then LV_CH_SET_OFF = 1

If LV_CH_AST_ENA = 1
Then **If** LV_CH_SET_OFF = 1
or VS > C_VS_MAX_CH
or (LV_DRI = 1 and TCO > C_TCO_DRI_MIN_CH)
or (LV_ACIN = 1 and TCO > C_TCO_ACIN_MIN_CH)
or T_ES < C_T_ES_THD_CH_AST
then LV_CH_AST_ENA = 0

**/ LV_CH_AST_ENA is irreversible until next RESET.*

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MAX_CH	1	0...FFH	0...255	1	[km/h]
Maximum vehicle speed for catalyst heating ignition angle correction					
C_TCO_DRI_MIN_CH	1	0...FEH	-48...142.5	0.75	[°C]
Minimum coolant temperature to stop Catalyst Heating corrections in case of LV_DRI = 1					
C_TCO_ACIN_MIN_CH	1	0...FEH	-48...142.5	0.75	[°C]
Minimum coolant temperature during switched on A/C for catalyst heating correction;Constant x					
C_MAF_INT_CH_MAX	1	0...FFFFH	0...9102.08	0.1388888	[g]
MAF_INT threshold for setting of LV_CH_SET_OFF					
C_TIA_ST_MIN_CH	1	0...FEH	-48...142.5	0.75	[°C]
Minimum TIA for CH activation					
C_T_ES_THD_CH_AST	1	0...FFFFH	0...65535	1	[min]
Engine off time threshold for catalyst heating after start					
C_AMP_AD_MIN_CH	1	0...FFFFH	0...5434	0.0829175	[hPa]
Ambient pressure threshold for catalyst heating break off					

5.9 Catalyst heating strategy

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_CH_MOD_REQ	O/V	0H 1H 2H	CH_MOD_OFF CH_MOD_HOM CH_MOD_MPLH	1	-
Requested CH-strategy (combustion mode) for catalyst heating					
STATE_CH_MOD	O/V	0H 1H 2H	CH_MOD_OFF CH_MOD_HOM CH_MOD_MPLH	1	-
Catalyst heating strategy					

Input data:

STATE_CH

5.9.1 General information

In Version 2 of the aggregate MPLH is not active. So STATE_CH_MOD is just set to CH_MOD_OFF of CH_MOD_HOM.

Application Condition

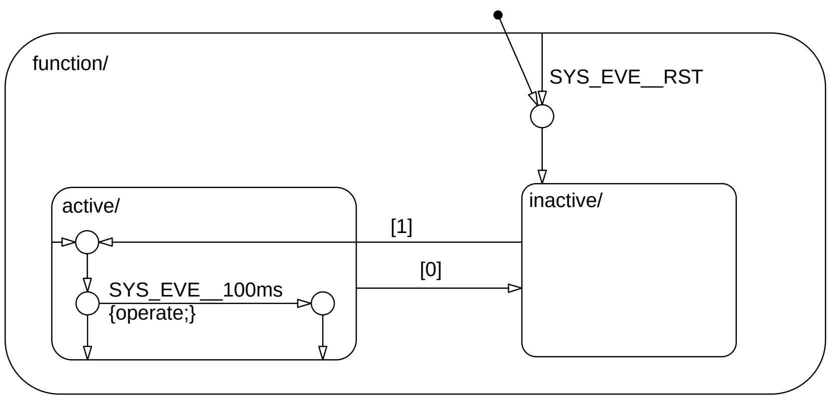


Figure 6 EXTC_REQGNchs0/ APP_CDN/ Chart1

Function Description

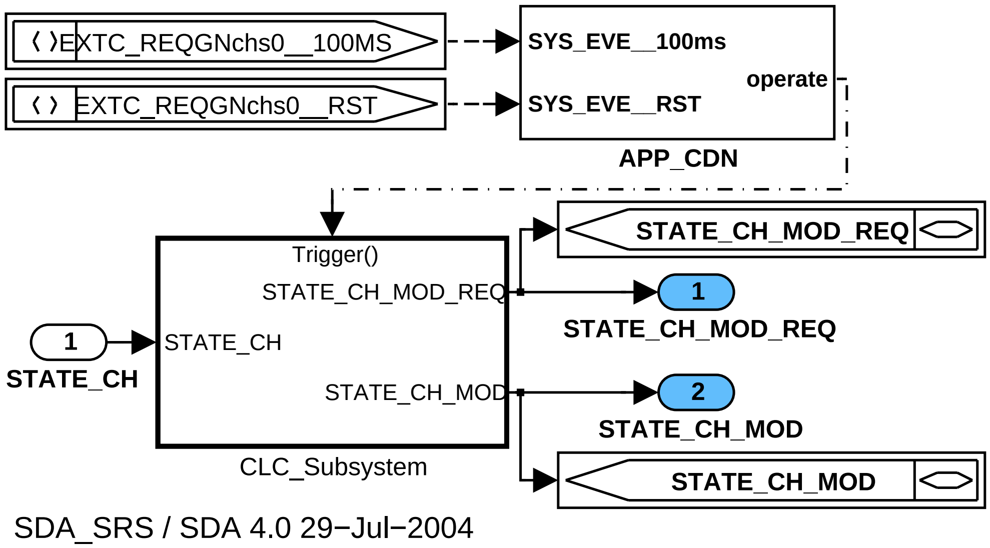


Figure 7 EXTC_REQGNchs0

5.9.1.1 SUBFUNCTION: CLC_Subsystem

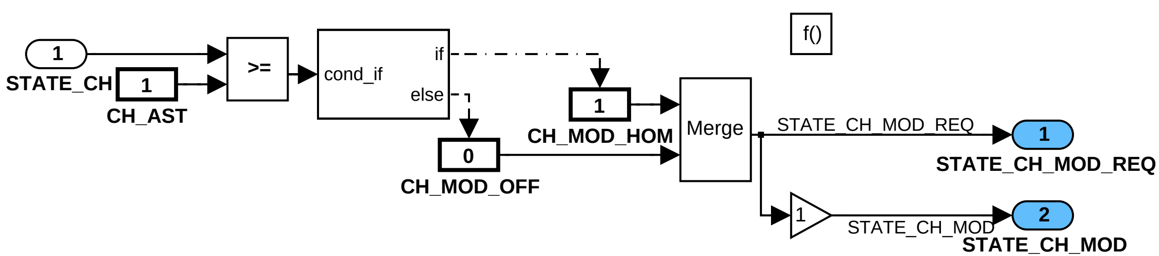


Figure 8 EXTC_REQGNchs0/ CLC_Subsystem

6 Ignition

Table of Contents

Index

A

C

M

N

6.1 General Ignition Angle

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_MIN	O/V	0...FFH	-35.625...60	0.375	°CRK
Minimum ignition angle					
IGA_SP	O/V	0...FFH	-35.625...60	0.375	°CRK
Setpoint ignition angle from torque managment					
LV_IGA_GRD_ACT	O/V	0...1H	0...1	1	-
Flag for ignition angle gradient limitation out of start active					
IGA_AV_MV	O/V	0...FFH	-35.625...60	0.375	°CRK
Mean value on all cylinder of the ignition angle AV					
IGA_BAS_COR_MV	O/V	0...FFH	-35.625...60	0.375	°CRK
Mean value of the bank selective corrected basic ignition angles					
IGA_AV_MV_CBK[NC_CBK_EX_NR]	O/V	0...FFH	-35.625...60	0.375	°CRK
Mean value on all cylinder of the ignition angle actual value - Exhaust cylinder bank selective					
IGA_AV[NC_CYL_NR]	O/V	0...FFH	-35.625...60	0.375	°CRK
Applied ignition angle; all corrections, knock control, torque intervention and start management included					
IGA_SP_MAX	V	0...FFH	-35.625...60	0.375	°CRK
Basic Ignition Angle, all corrections and application incidence choice included					
IGA[NC_CYL_NR]	V	0...FFH	-35.625...60	0.375	°CRK
Basic ignition angle, all corrections included (no torque and start management included)					
IGA_AV_1[NC_CYL_NR]	-	0...FFH	-35.625...60	0.375	°CRK
Actual value of ignition angle before after-start ignition management					

Input data:

IGA_ST	IGA_REF_COR	IGA_BAS_COR[NC_CBK_IN_NR]	LV_HOM_RUN
TCO	N_32	LV_IGA_EOL_ACT	LV_ST
LV_ES	IGA_CYL_KNK[NC_CYL_NR]	IGA_EOLP	NC_CYL_NR
NC_CBK_IN_NR	NC_CBK_EX_NR	IGA_DIF_TQ_REQ	IGA_DIF_MIN
IGA_EXT_COR[NC_CYL_NR]	CYC_CAST		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_IGA_ST_ENA	1	0...1H	0...1	1	
Logical flag for different ignition angle at start					
C_IGA_INI	1	0...FFH	-35.625...60	0.375	°CRK
calibrateable data ignition angle initialisation					
C_IGA_AS_CYL[NC_CYL_NR]	1	80...7FH	-48...47.625	0.375	°CRK
Spark retard by application system - cylinder individual					
C_IGA_AS	1	80...7FH	-48...47.625	0.375	-
Spark retard by application system					
LC_IGA_MAN_ACT	1	0...1H	0...1	1	-
logical constant for manual ignition angle setting					
C_IGA_MAN	1	0...FFH	-35.625...60	0.375	°CRK
Manual ignition angle value					
IP_IGA_LGRD_AST	8	0...FFH	0...95.625	0.375	°CRK
LDPM_TCO_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
Ignition step gradient limitation after start phase					
ID_CYC_DLY_IGA_GRD_ACT	4	0...4H	0...4	1	-
LDP_TCO_IGSP_IGA_GRD	4	0...FEH	-48...142.5	0.75	°C

Cycle delay to activate IGA_GRD after engine start

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_N_IGA_UPD_RATE	1	0...FFH	0...8.16E+3	32	rpm
Threshold on engine speed to change update rate of IGA calculation					

6.1.1 General information:

In this module, the spark advance that has to be applied to the ignition coil (IGA_AV[x]) is determined for each cylinder individually according to the actual engine capability and the actual torque request.

The ignition value IGA_AV[x] to apply on cylinder x is determined versus the fast torque intervention set point (IGA_SP), knock constraints (IGA_CYL_KNK[x]), IGA_MIN and IGA_BAS_COR[i].

Manual calibration of the actual ignition angle can be done if the flag LC_IGA_MAN_ACT = 1 For End of Line process, it is also possible to specify the ignition angle to be applied via the IGA_EOLP input by setting the flag LV_IGA_EOL_ACT to 1

IGA_DIF_TQ_REQ is the output from the torque management functions to realise the torque setpoint. This request from torque management is translated in IGA_SP within this module. IGA_SP is limited between IGA_MIN and IGA_BAS_COR[i] .

IGA_BAS_COR[i] is the corrected basic ignition angle which is realised if no other requests like knock control or torque requests (via IGA_SP) require a retarded ignition angle.

For engine having a multiple cylinder banks configuration (NC_CBK_IN_NR > 1) a cylinder bank specific Corrected Basic Ignition Angle IGA_BAS_COR[i] – with i designating the corresponding cylinder bank – is calculated.

The reference ignition angle IGA_REF_COR is a theoretical ignition angle at maximum engine torque. It is the reference value to all ignition angles for the calculation of all the ignition efficiencies.

According to application choice (*LC_IGA_ST_ENA = 1 or 0; read only after an ECU reset*), the ignition during start phase is not managed through the torque (via IGA_SP) but by using the input value IGA_ST.

At the end of start phase the transition from IGA_ST to the set point ignition IGA_AV_1[x] is done through a gradient limitation (*IP_IGA_LGRD_AST).*

Function Description

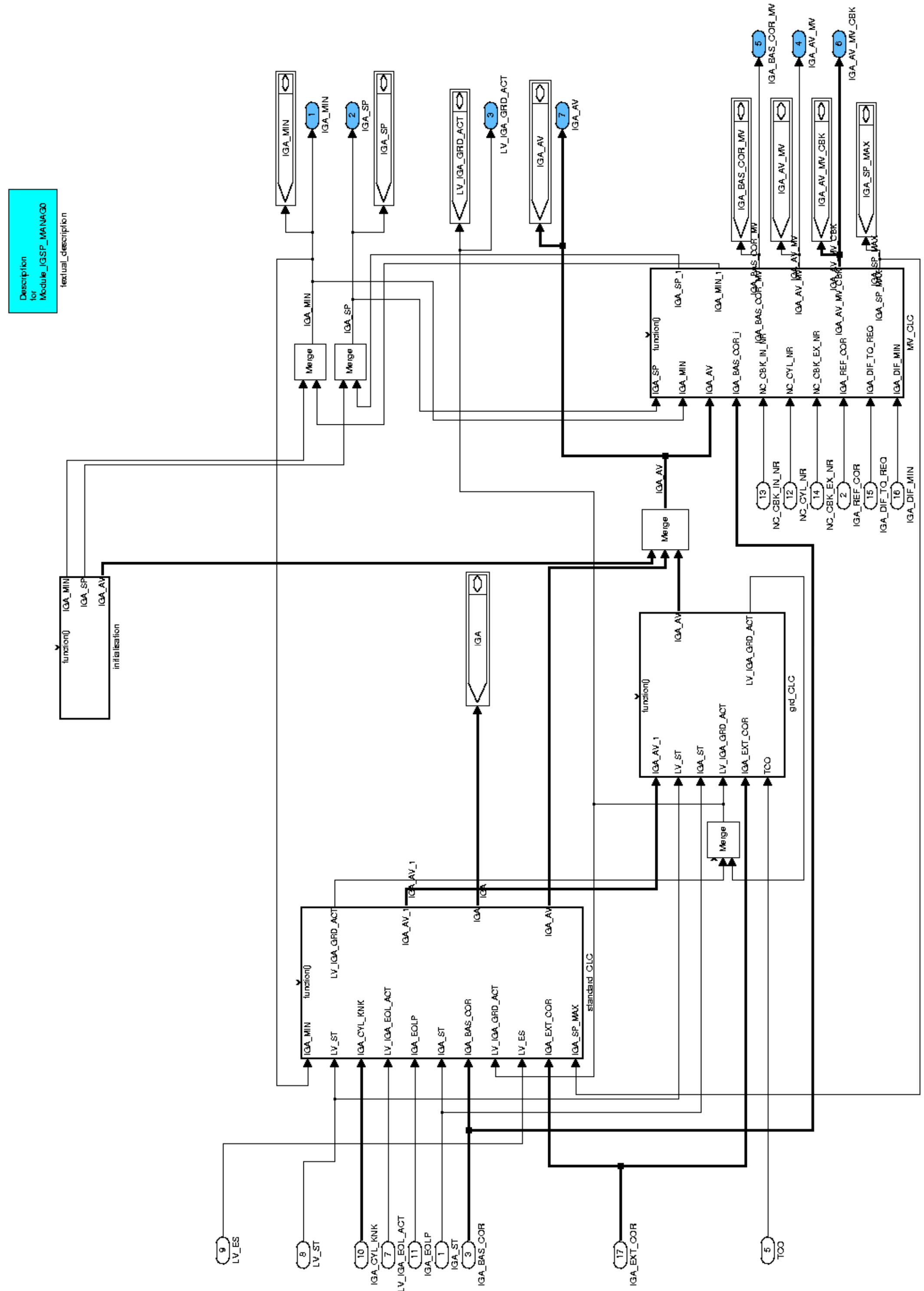


Figure 1 IGSP_MANAG0

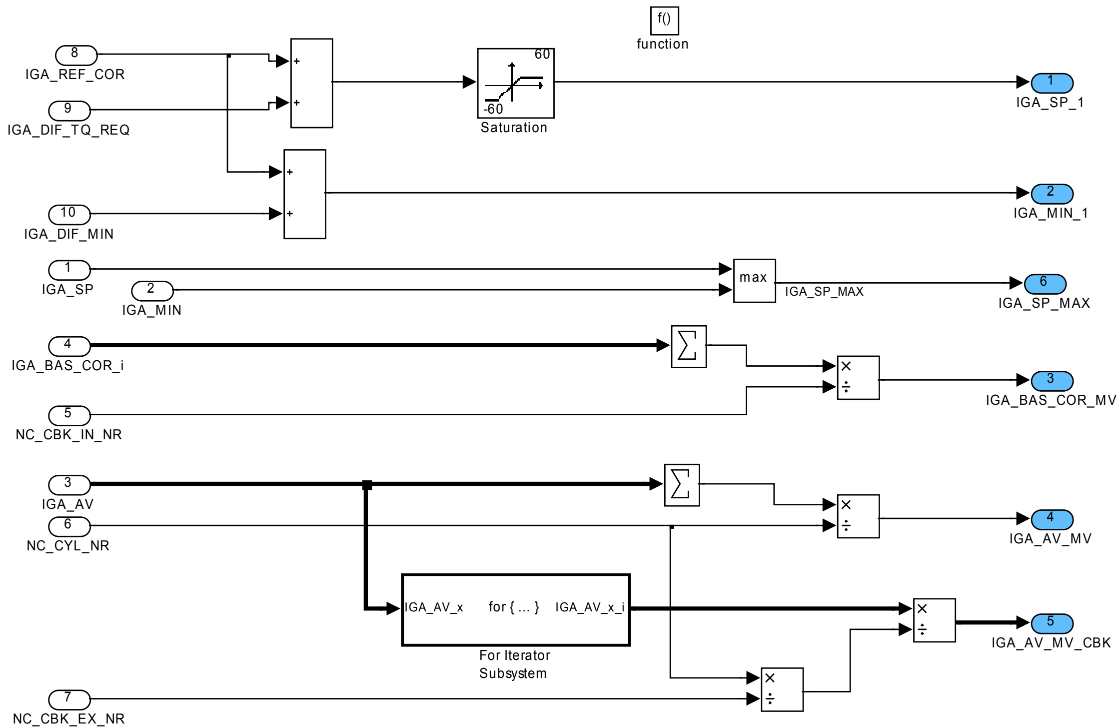


Figure 2 IGSP_MANAG0/ MV_CLC

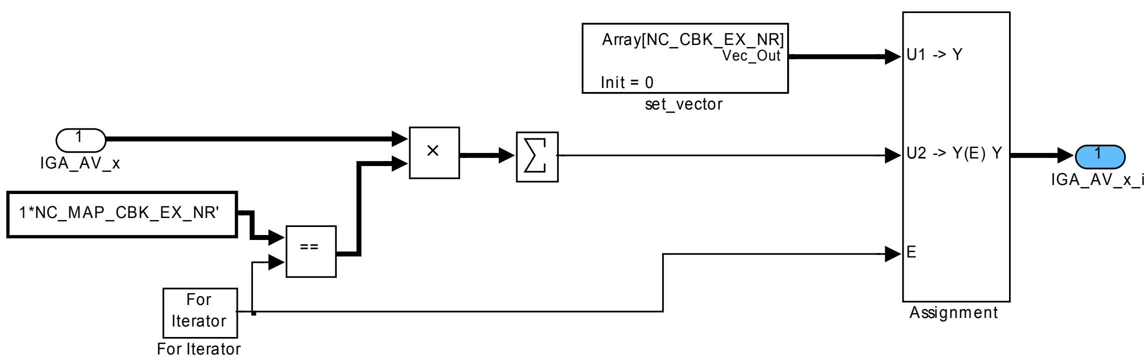


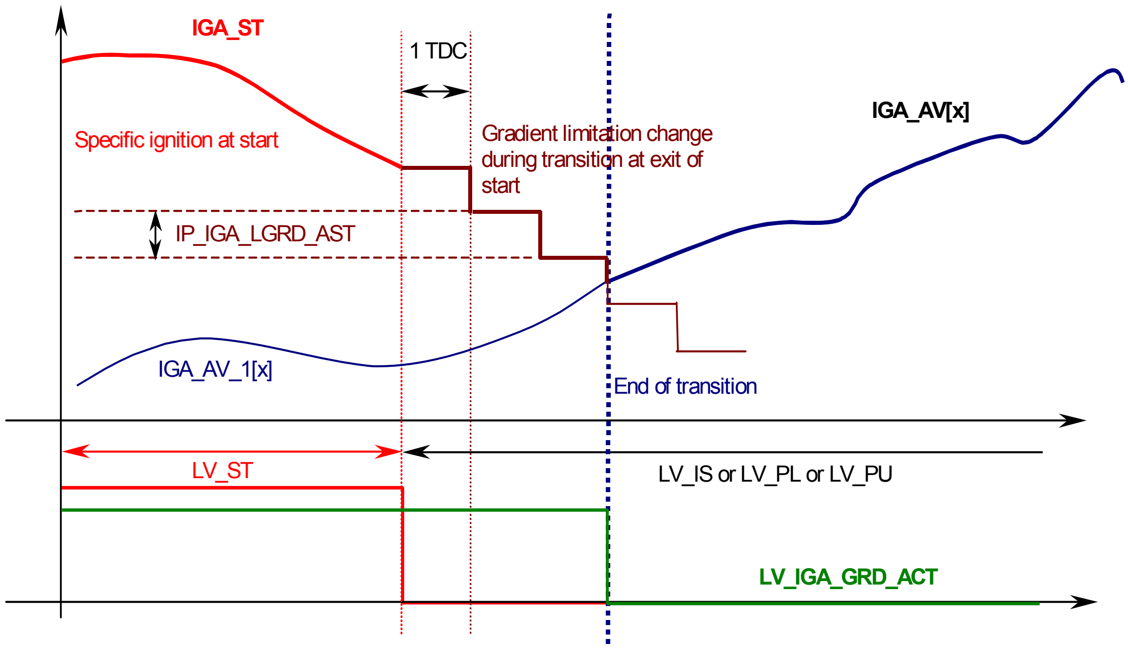
Figure 3IGSP_MANAG0/ MV_CLC/ For Iterator Subsystem

IGSP_MANAG0/GRD_CLC

IGA_ST is used during the starting phase.

At transition from LV_ST = 1 to 0, the ignition value at the end of start (IGA_ST) is returned to the ignition value out of start (IGA_AV_1[x]) using the change limitation IP_IGA_LGRD_AST. This gradient come-back on ignition angle out of start is applied every TDC.

The figure herafter gives an overview of the IGA calculated during and just after start phase if LC_IGA_ST_ENA is activated.



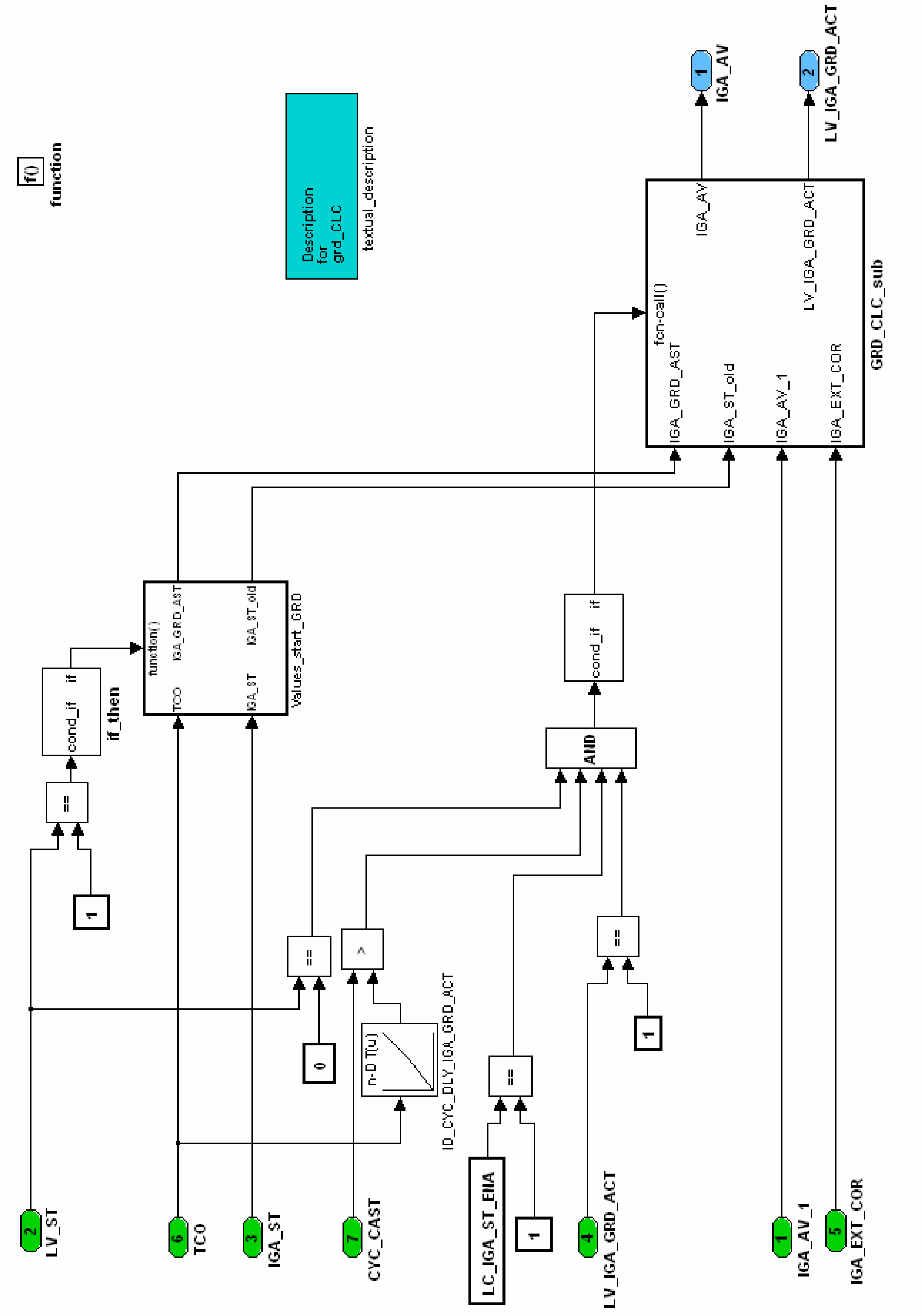


Figure 4 IGSP_MANAG0/ grd_CLC

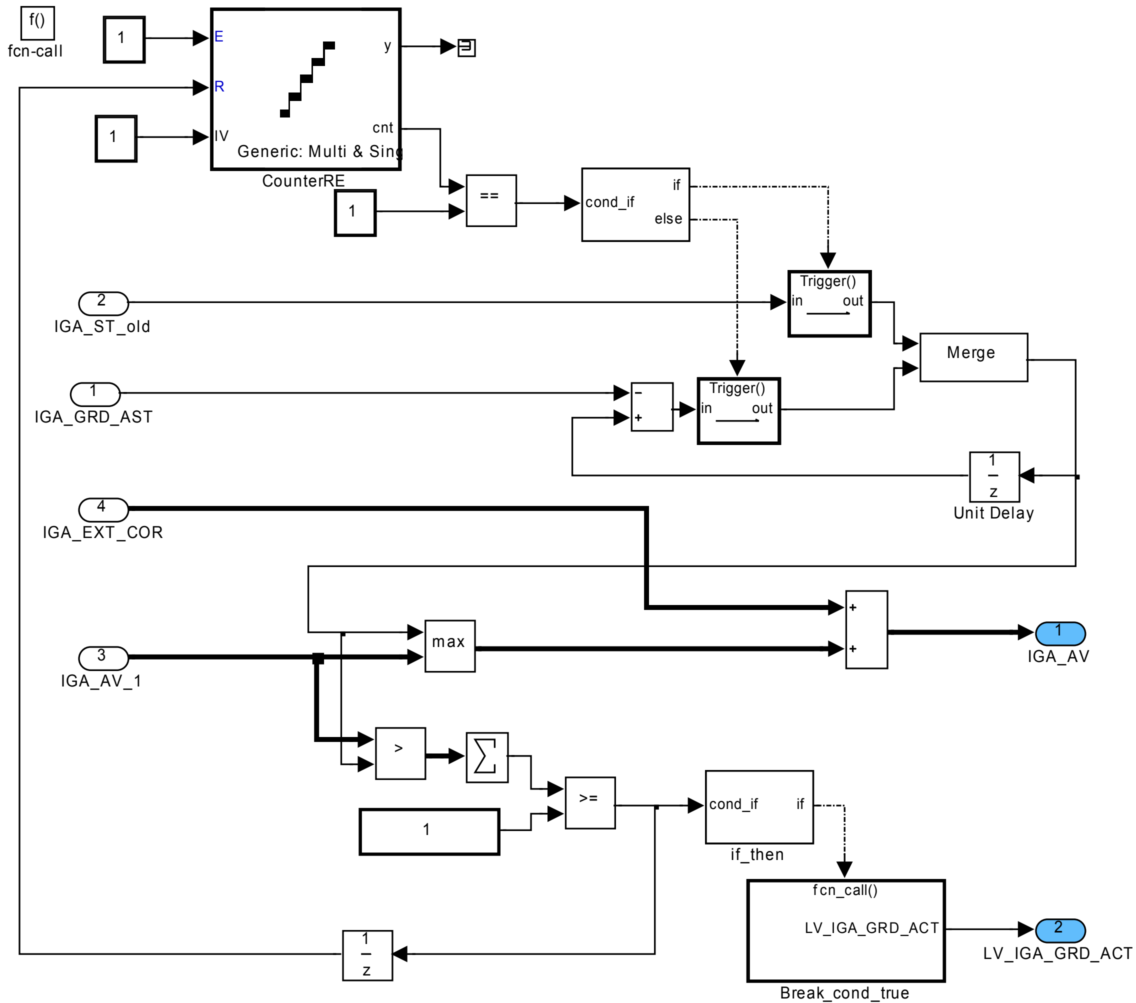


Figure 5 IGSP_MANAG0/ grd_CLC/ GRD_CLC_sub

Figure 6 IGSP_MANAG0/ grd_CLC/ GRD_CLC_sub/ Break_cond_true

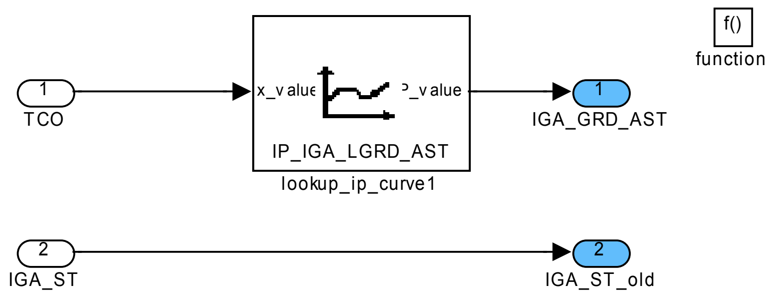


Figure 7 IGSP_MANAG0/ grd_CLC/ Values_start_GRD

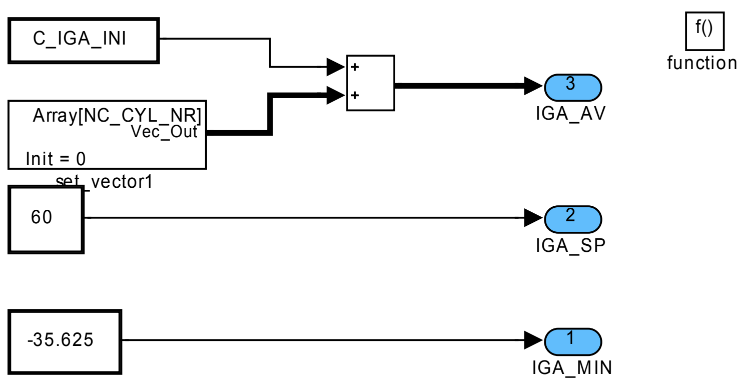


Figure 8 IGSP_MANAG0/ initialisation

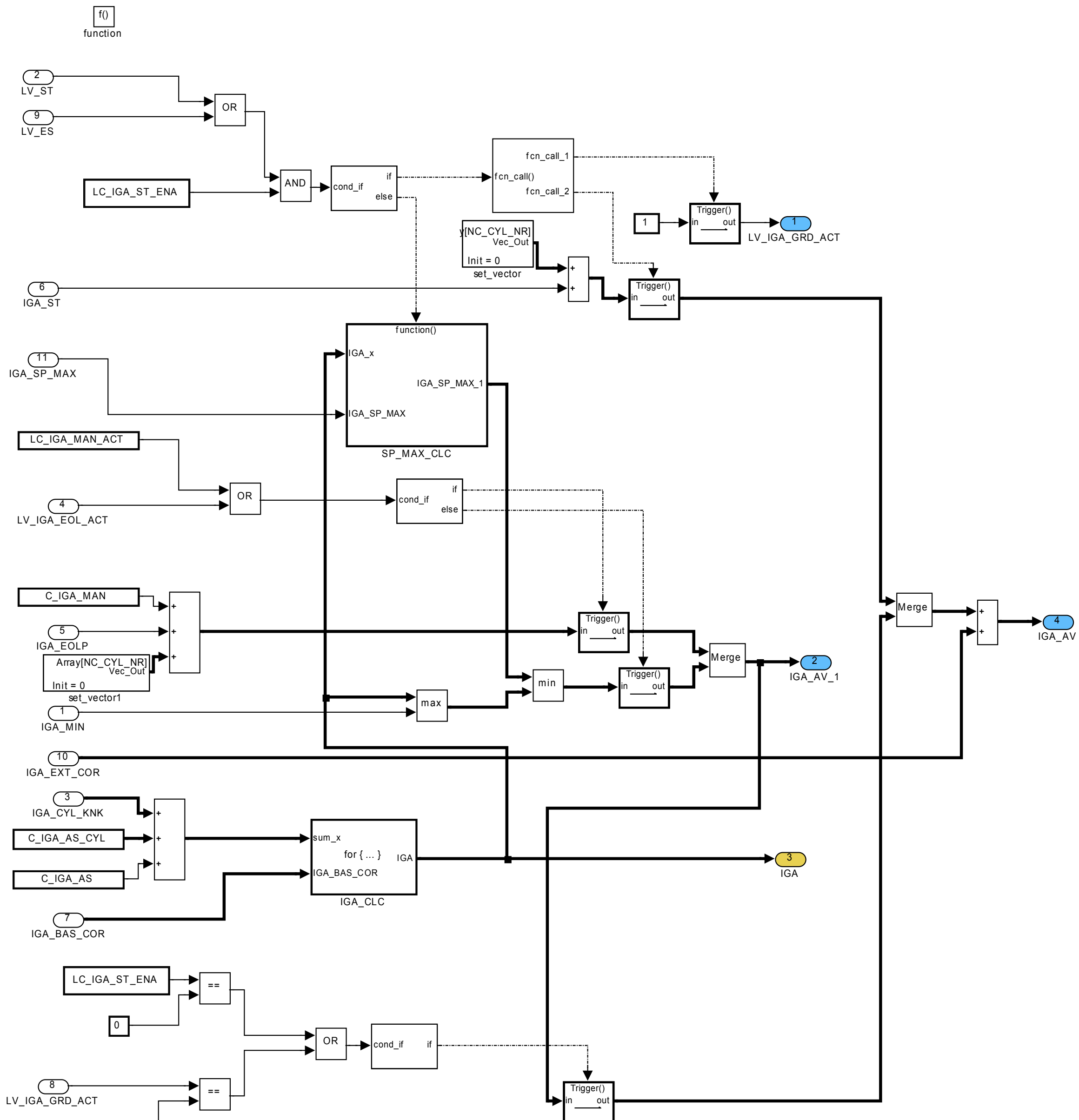


Figure 9 IGSP_MANAG0/ standard_CLC

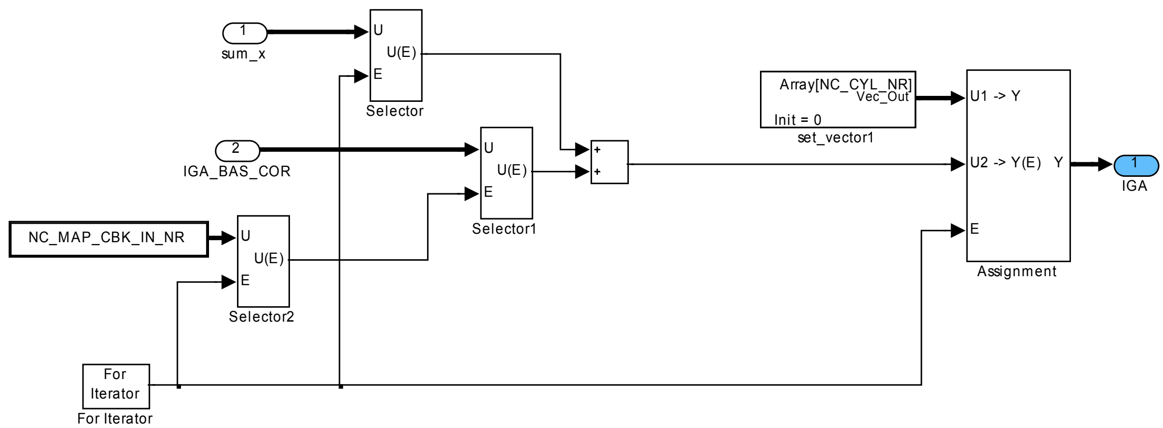


Figure 10 IGSP_MANAG0/ standard_CLC/ IGA_CLC

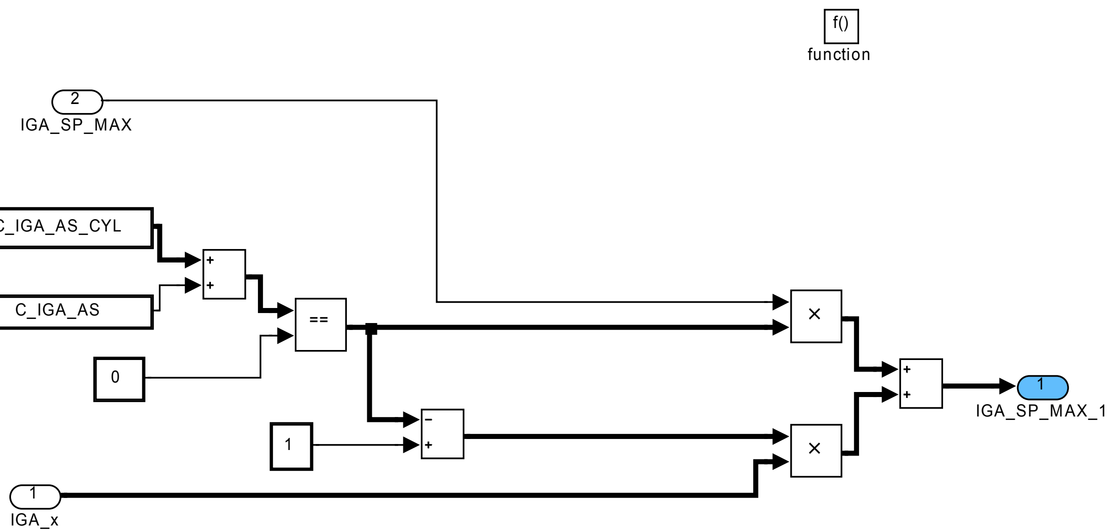


Figure 11 IGSP_MANAG0/ standard_CLC/ SP_MAX_CLC

6.2 Ignition Angle Efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_IGA_AV	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency actual ignition angle					
EFF_IGA_AV_CBK[NC_CBK_EX_NR]	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency actual ignition angle - Cylinder Bank selective					
EFF_IGA_BAS_COR	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency basic ignition angle, knock correction included					
EFF_IGA_BAS_COR_KNK_FIL	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency basic ignition angle, knock control filtered					
EFF_IGA_MIN	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency minimum ignition angle					
EFF_IGA_MIN_TEG	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Ignition efficiency minimum ignition angle with TEG correction					
IGA_DIF_BAS	O/V	0...FFH	0...-95.625	0.375	°CRK
Difference from basic to reference ignition angle					
IGA_DIF_AV_CBK[NC_CBK_EX_NR]	V	0...FFH	0...-95.625	0.375	°CRK
Difference from actual ignition angle to reference ignition angle – Cylinder Bank selective					
IGA_MV_KNK_FIL	V	80...7FH	-48...47.625	0.375	°CRK
Filtered value of knock mean correction					
IGA_BAS_COR_KNK_ADD	V	0...FFH	-35.625...60	0.375	°CRK
Corrected basic ignition angle, knock mean value included					
IGA_BAS_COR_KNK_FIL	V	0...FFH	-35.625...60	0.375	°CRK
Corrected basic ignition angle, knock control filtered					
IGA_DIF_AV	V	0...FFH	0...-95.625	0.375	°CRK
Difference from actual ignition angle to reference ignition angle					

Input data:

IGA_AV_MV	IGA_AV_MV_CBK[NC_CBK_EX_NR]	IGA_BAS_COR_MV	IGA_DIF_MIN
IGA_DIF_MIN_TEG	IGA_MV_KNK_ADD	IGA_REF_COR	LC_IGA_ST_ENA
LV_HOM_RUN	LV_ST	NC_CBK_EX_NR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_IGA_KNK	1	0...FFH	0...1	0.003922	-
Constant for knock correction filtering in efficiency calculation					
IP_EFF_IGA	16	0...FFFFH	0...1.9999	3.0517e-005	-
LDP_IGA_DIF_IP_EFF_IGA	16	0...FFH	0...-95.625	0.375	°CRK
Basic ignition angle efficiency					

6.2.1 General Information

To be consistent with the specific ignition management during start phase, the efficiency must be initialised during this phase. Otherwise, the efficiencies are calculated versus ignition deviations from the reference value.

In case of engine with multiple cylinder bank configuration (NC_CBK_EX_NR>1), for a more efficient modelling and control of the exhaust gas temperature, a cylinder bank specific actual ignition efficiency is determined for each bank.

In order to manage the knock effects on the torque co-ordination (unlocking of the knock control with the throttle opening), two different efficiencies are calculated on the corrected basic ignition angle. The first one include the mean value of the knock correction, the second one only takes care of a filtered value of this knock correction. In this way, the effect of the knock control on the torque management can be smoothened according to the filter constant (and even inhibited if the filter constant is set to 0).

Function Description

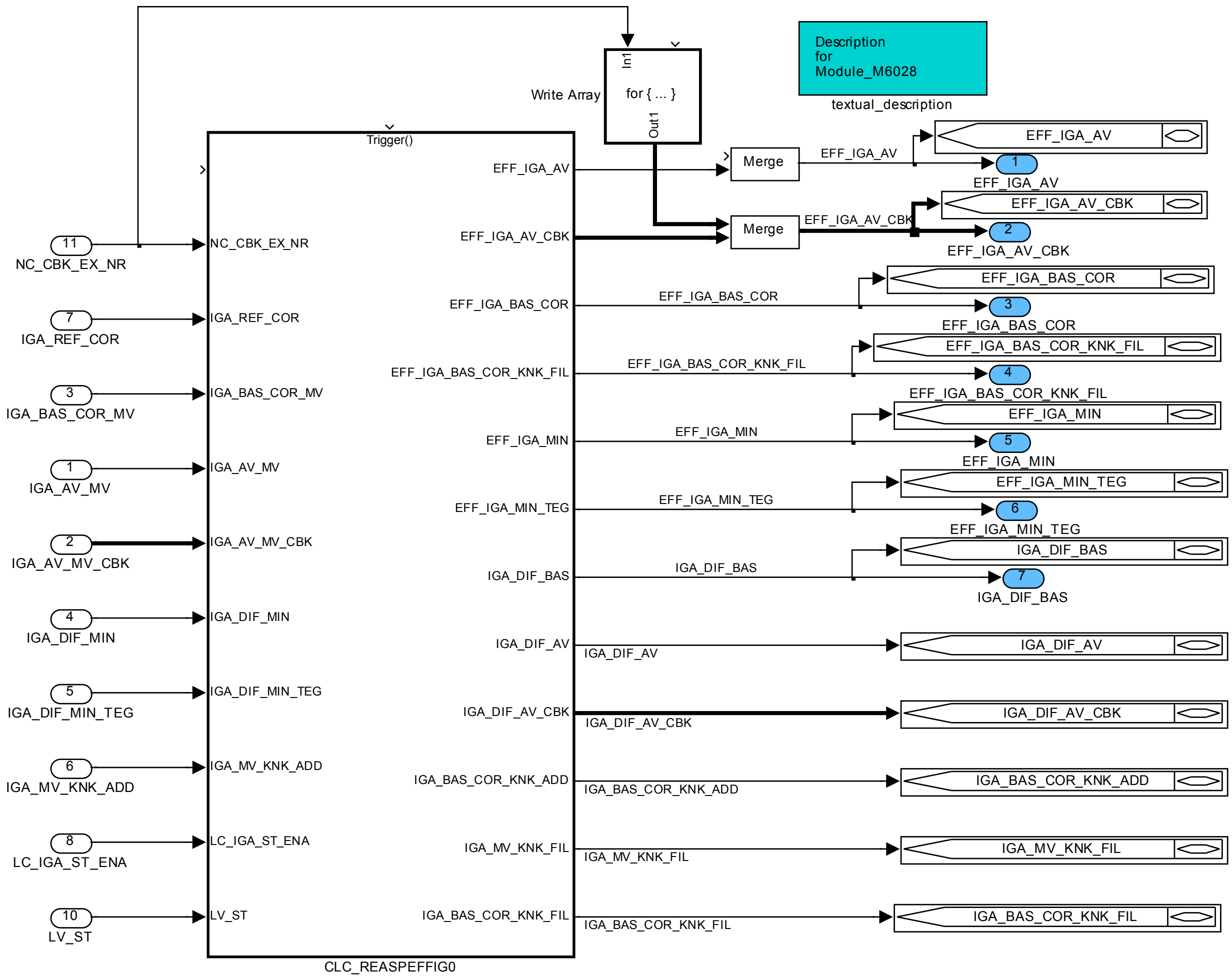


Figure 12 M6028

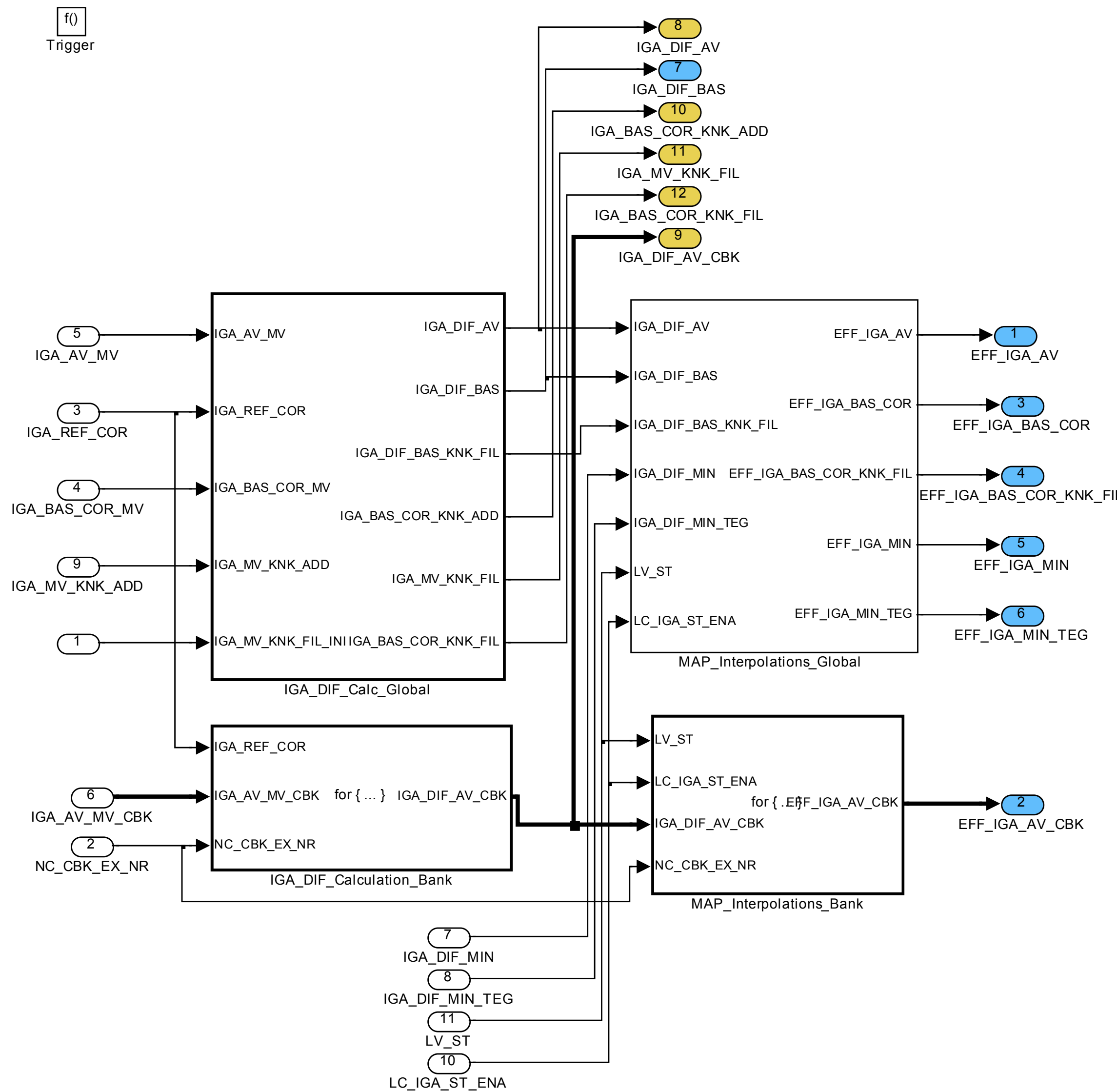


Figure 13 M6028/ CLC_REASPEFFIG0

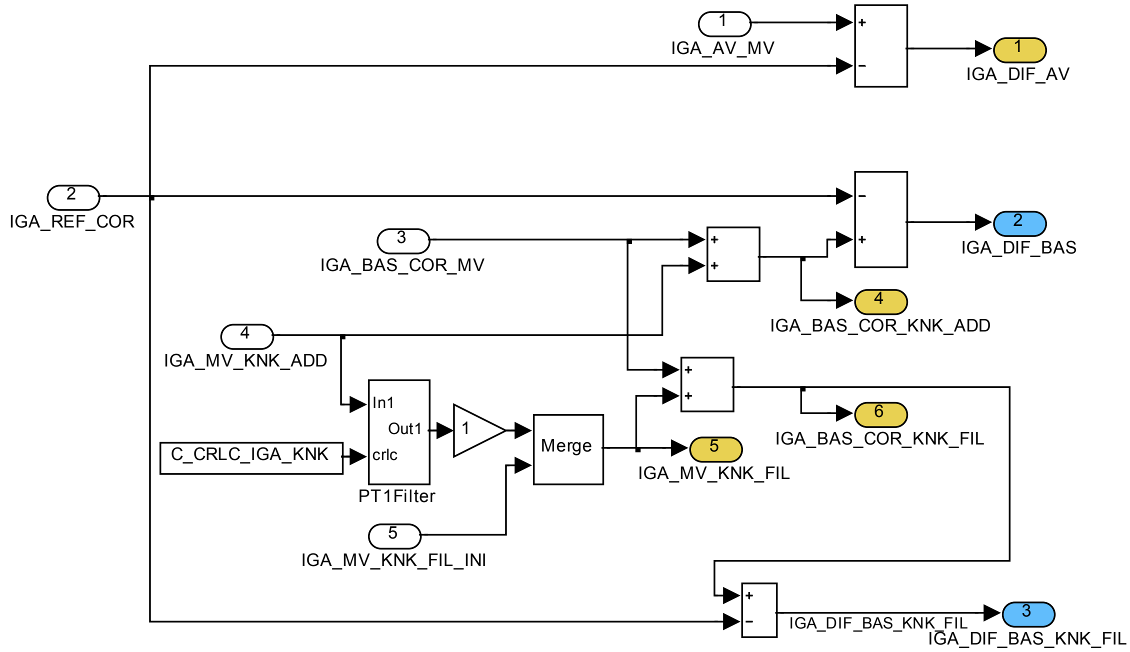


Figure 14 M6028/ CLC_REASPEFFIG0/ IGA_DIF_Calc_Global

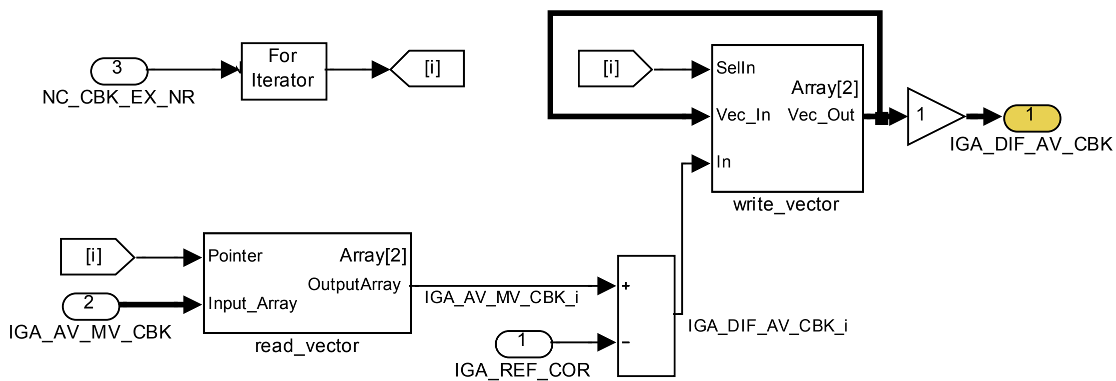


Figure 15 M6028/ CLC_REASPEFFIG0/ IGA_DIF_Calculation_Bank

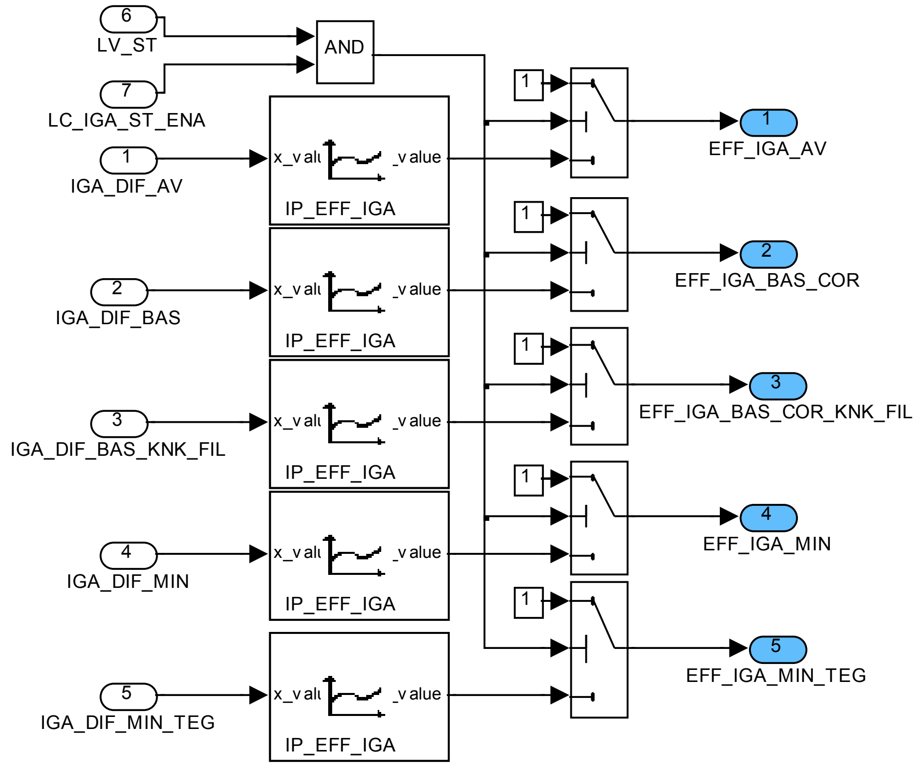


Figure 16 M6028/ CLC_REASPEFFIG0/ MAP_Interpolations_Global

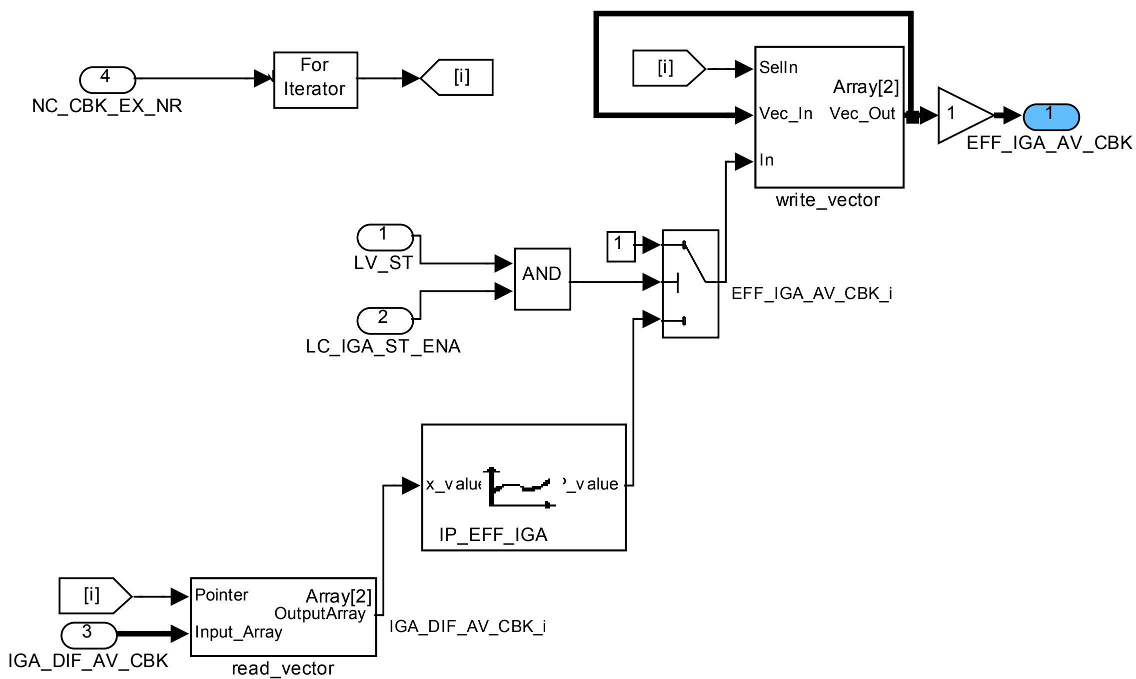


Figure 17 M6028/ CLC_REASPEFFIG0/ MAP_Interpolations_Bank

6.3 Configuration on IGA calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_REF_INTER_COR	O/V	80..7FH	-48..47.625	0.375	°CRK
Intermediate External corrections on Reference IGA					
IGA_BAS_INTER_COR[NC_CBK_IN_NR]	O/V	80..7FH	-48..47.625	0.375	°CRK
Intermediate External correction on Basic IGA – cylinder bank i related					
IGA_ST_COR	O/V	80..7FH	-48..47.625	0.375	°CRK
Additive External correction on IGA at Start					
IGA_EXT_COR[NC_CYL_NR]	O/V	80...7FH	-48...47.625	0.375	°CRK
Additive external ignition angle correction					

Input data:

LV_HOM_RUN	NC_CBK_NR	IGA_BAS_VVT_COR	IGA_REF_VVT_COR
IGA_BAS_AMP_COR	IGA_BAS_PORT_COR_i	IGA_REF_PORT_COR	IGA_ADD_ER_BAL[NC_CYL_NR]
LV_CH	LV_IS	TCO	

FUNCTION DESCRIPTION:

General information:

The basic and reference ignition angle have to be corrected due to influence of other external sub-systems(ex. VVTI) . These corrections are summed up to be added to the reference ignition angle within the module "General correction on Ignition angle".

Application conditions:

Activation: LV_HOM_RUN = 1

Deactivation: LV_HOM_RUN = 0

Initialisation: -

Update Rate: IGA_REF_INTER_COR => 10ms
IGA_BAS_INTER_COR_i => 10ms

Formula section:

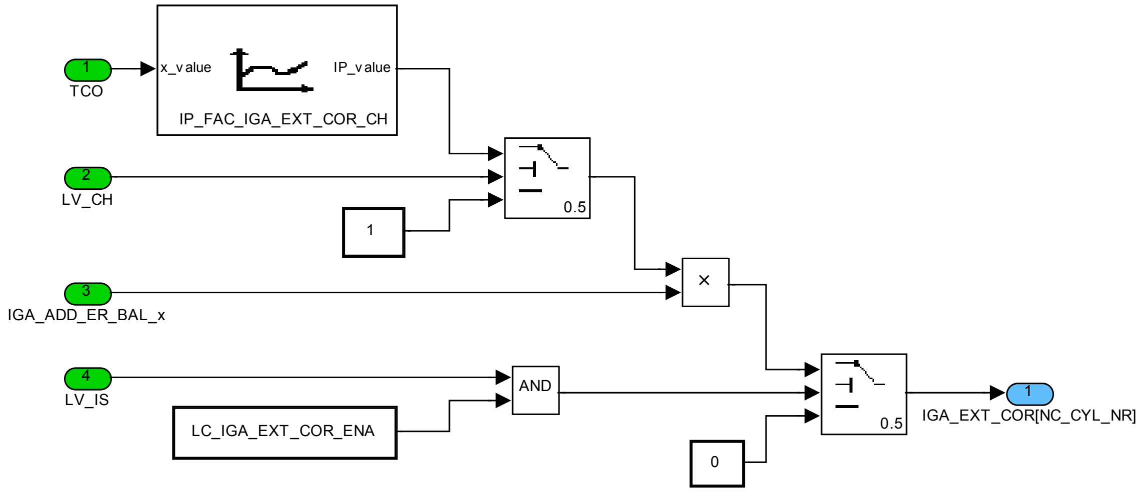
IGA_REF_INTER_COR = IGA_REF_VVT_COR
+ IGA_REF_PORT_COR

IGA_BAS_INTER_COR_i = IGA_BAS_VVT_COR+
+ IGA_BAS_PORT_COR_i

+ IGA_BAS_AMP_COR

IGA_ST_COR = 0

Correction of the basic ignition angle due to cylinder balancing though IGA intervention during idle speed



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_IGA_EXT_COR_ENA	1	0...1H	0...1	1	-
Logical switch to enable external ignition angle correction					
IP_FAC_IGA_EXT_COR_CH	1x6	0..FFH	0...1.992	0.0078	-
LDP_TCO_IP_FAC_IGA_EXT_COR_CH	6	0...FEH	-48...142.5	0.75	°C
Ignition angle correction factor for additive cylinder balancing value at catalyst heating					

6.4 Dwell time control (open loop)

6.4.1 Dwell time calculation

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TD	O/V	0...FFFFH	0....262,14	0,004	ms
Dwell time					

Input data:

TD_FAC	VB	N_32	LV_HOM_RUN

FUNCTION DESCRIPTION:

The dwell time raw value calculation is based on the simulation done by the HW-group. At any time an adaptation or correction factor has to be multiplied or added in this chapter.

The dwell time open-loop control value is stored in terms of a battery voltage and engine speed related characteristics.

Application conditions:

Initialisation: TD = C_TD_INI *at reset*

Recurrence: 10 ms

Activation: *at every engine state* **if** LV_HOM_RUN = 1

Formula section

TD = IP_TD(VB,N_32) * TD_FAC + C_TD_AS

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
IP_TD	8*8	0...FFFFH	0...262,14	0,004	ms
LDPM_VB_1_IGRE	8	0…FFH	0….26	0,102	V
LDPM_N_32_1_IGRE	8	0…FFH	0….8160	32	rpm
Dwell time correction versus battery voltage and engine speed					
C_TD_AS	1	F800...7F0H	-8,192..8,128	0,004	ms
Dwell time offset usual value: 0 ms					
C_TD_INI	1	0...FFFFH	0...262,14	0,004	ms
initial value for dwell time control (typ. Value 3ms for standart coils and <2ms for pencil coils)					

6.5 Dwell time factor calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TD_FAC	O/V	1...FFH	1/128....255/128	1/128	-
General dwell time control factor					

Input data:

TCO	N_32		

FUNCTION DESCRIPTION:

General information:

The dwell time factor is to increase the basic dwell time depending on coolant temperature.

This function is requested by HMC.

Initialisation : at reset

TD_FAC = IP_FAC_TD

Recurrence: 10 ms

Activation: all engine operating states

Formula section:

TD_FAC = IP_FAC_TD

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
IP_FAC_TD	8*8	1...FFH	1/128....255/128	1/128	-
LDP_TCO_IP_FAC_TD	8	0…FEH	-48….142.5	0,75	˚C
LDP_N_32_IP_FAC_TD	8	0…FFH	0….8160	32	rpm
Dwell time correction factor versus coolant temperature and engine speed					

6.6 Ignition Inhibition

6.6.1 Ignition inhibition at ignition key off

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_INH_IGC_IGK	V/O	0..1H	0..1	1	-
Flag to Inhibit ignition at IGK off					
CTR_CYC_IGK_OFF	V	0..FFH	0..255	1	-
Cycle counter with ignition off					

Input data:

LV_IGK	N_32	TCO	PREV_STATE_IV

Description:

If the the engine is running and the ignition key is turned off only the injection is inhibited but there is still a ignition spark. An effect of this is that despite zero injection there is combustion because of HCs from wall film, canister purge etc. As a result of this present HCs a combustion happens and the engine stalls very slow. This situation is worse at low coolant temperatures (TCO) because of the high wall film masses.
To guarantee a fast engine stall in case of ignition key off the ignition has to be inhibited too. To ensure that the injected fuel mass of all cylinders is ignited every cylinder has to be ignited once more after ignition key is turned off.

Application conditions:

Initialisation: at reset and deactivation

LV_INH_IGC_IGK = 0

CTR_CYC_IGK_OFF = 0

Recurrence: Segment syncronous
Activation: LV_IGK **=** 0
Deactivation: LV_IGK **=** 1

Formula section:

Increment counter CTR_CYC_IGK_OFF

If LV_IGK **=** 0
and N_32 **<** C_N_ES_IGC
and CTR_CYC_IGK_OFF > ID_CTR_CYC_IGK_OFF
and PREV_STATE_IV **=** 0
Then LV_INH_IGC_IGK **=** 1
Else LV_INH_IGC_IGK **=** 0

The ignition inhibition is done immediately

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_N_ES_IGC	1	0...FFH	0...8160	32	rpm
enigne speed threshold condition for ignition deactivation					
ID_CTR_CYC_IGK_OFF	6	0...FF	0...255	1	-
LDP_TCO_CTR_CYC_IGK_OFF	6	0..FE	-48...142.5	0.5	°C
Maximum ignition cycles with ignition switched off					

6.7 TD and IGA switches

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TD_IGC_x	O/V	0..FFFFH	0...262.14	0,004	ms
Dwell time duration					
IGA_IGC_x	O/V	0..FFH	-35.625...60	0,375	°CRK
Ignition angle of cylinder x					

Input data:

IGA_AV_x	TD	TD_FAC_MIN	TD_FAC_MAX
NC_CYL_NR	LV_ES	LV_IGK	LV_FIRST_VLD_TOOTH
LV_IGC_x_EXT_ADJ	IGC_x_EXT_ADJ	LV_INH_IGC[NC_CYL_NR]	

Import actions:

ACTION_INFR_SetIgnDwell(IN <>, IN <>, IN <>, IN <>)
ACTION_INFR_SetIgnAngle(IN <>, IN <>)
ACTION_INFR_SetIgcDwellTest(IN <>, IN <>)

FUNCTION DESCRIPTION:

This module should cover all necessary switches between the different dwell times and ignition angles to ensure two outputs, TD_IGC and IGA_IGC_x. This version is only used if there is only one combustion mode to rename the dwell time and the ignition angle to be coherent with the generic system structure.

Initialisation : at reset

TD_IGC_x = TD
IGA_IGC_x = IGA_AV_x

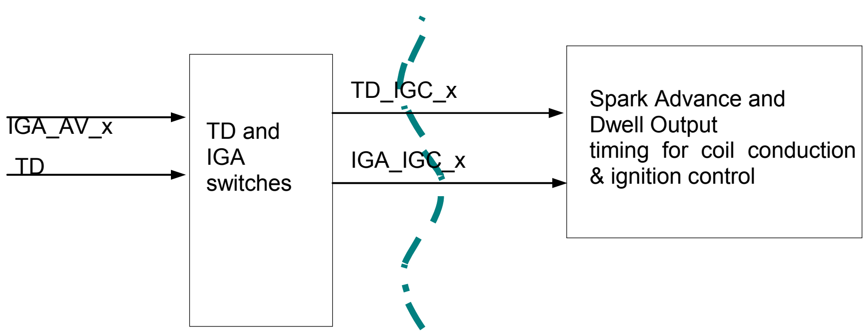
For x = 0 to NC_CYL_NR - 1

ACTION_INFR_SetIgnDwell(x,TD_FAC_MIN,TD_IGC[x],TD_FAC_MAX)

ACTION_INFR_SetIgnAngle(x,IGA_IGC[x])

Endfor

Signal flow diagram:



6.7.1 Dwell time switch

Application conditions:

Activation: Engine synchronized

Recurrence: 10ms

Formula section:

TD_IGC_x = TD

For x = 0 to NC_CYL_NR - 1

ACTION_INFR_SetIgnDwell(x,TD_FAC_MIN,TD_IGC[x],TD_FAC_MAX)

Endfor

6.7.2 Dwell time switch Ignition Actuator Tests Diagnosis

Application conditions:

Activation:	[	LV_ES = 1	*// Actuator tests activation*
	and	LV_IGK = 1	
	and	LV_FIRST_VLD_TOOTH = 0	
	and	LV_INH_IGC[x] = 0	
	and	at least one LV_IGC_x_EXT_ADJ = 1]	
Deactivation:	[	LV_ES = 0	*// Actuator tests deactivation*
	or	LV_IGK = 0	
	or	LV_FIRST_VLD_TOOTH = 1	
	or	LV_INH_IGC[x] = 1	
	or	at least one LV_IGC_x_EXT_ADJ = 0]	
Recurrence:	On transition of IGC_x_EXT_ADJ from 0 to 1		

System description:

For ignition coil actuator tests purpose, the Dwell Time has to be applied and transmitted to BSW via action. A specific service is used for those tests purposes.

Formula section:

TD_IGC_x = TD

```
For     x = 0 to NC_CYL_NR – 1
        If      LV_IGC_x_EXT_ADJ = 1
        Then    On transition of IGC_x_EXT_ADJ from 0 to 1        and
                LV_FIRST_VLD_TOOTH = 0
                ACTION_INFR_SetIgcDwellTest(x,TD_IGC[x])
        ElseIf  LV_FIRST_VLD_TOOTH = 1
                ACTION_INFR_SetIgcDwellTest(x,0)        // immediate cut on engine running
        EndIf
EndFor
```

6.7.3 Ignition angle switch

Application conditions:

Activation: Engine synchronized

Recurrence: 10ms + segment

Formula section:

IGA_IGC_x= IGA_AV_x

For x = 0 to NC_CYL_NR - 1

ACTION_INFR_SetIgnAngle(x,IGA_IGC[x])

Endfor

6.8 Ignition with multiple spark

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TD_MPL	V/O	0...FFFFH	0...262,14	0,004	ms
Dwell time for multiple spark					
TD_MPL_DLY	V/O	0...FFFFH	0...262,14	0,004	ms
Dwell time interruption					
IGN_MPL_NR	V/O	0...FFH	0…255	1	-
Number of ignition sparks					
T_CTR_TD_MPL	V	0...FFFFH	0...6553,5	0,1	s
Maintenance Time of Multiple Spark activation before Out Case					

Input data:

VB	TCO	LV_PUC	N_32
LV_IS	LV_ST	LV_ES	LV_FL
LDPM_VB_1_IGRE	LDPM_N_32_1_IGRE	LV_PL	LV_PU
LV_PUC	#IF(NC_INJ_CONF=HPDI		
	STATE_INJ_TYP_MEM_IV		
	#ENDIF		

Import actions:

ACTION_ INFR_SetIgnMplDwell (IN <>)
ACTION_ INFR_SetIgnMplDly (IN <>)
ACTION_ INFR_SetIgnMplNr (IN <>)

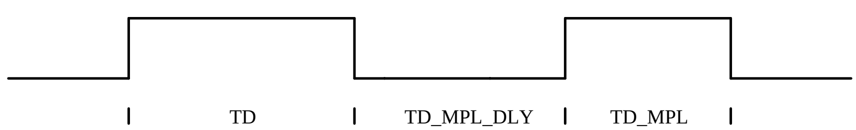
6.8.1 Calculation of parameters for multiple spark

FUNCTION DESCRIPTION:

With this function several subsequent sparks could be created. The delay time to create a spark and the time to reload the coil are defined in tables depending on battery voltage.

At start and for the following application conditions, the ignition is controlled by means of multiple sparks.

After an adjustable combustion period (C_TD_T_MPL), the ignition coil is loaded with a specific dwell time (IP_TD_MPL) to create successive sparks. The number of successive sparks is defined as IGN_MPL_NR.



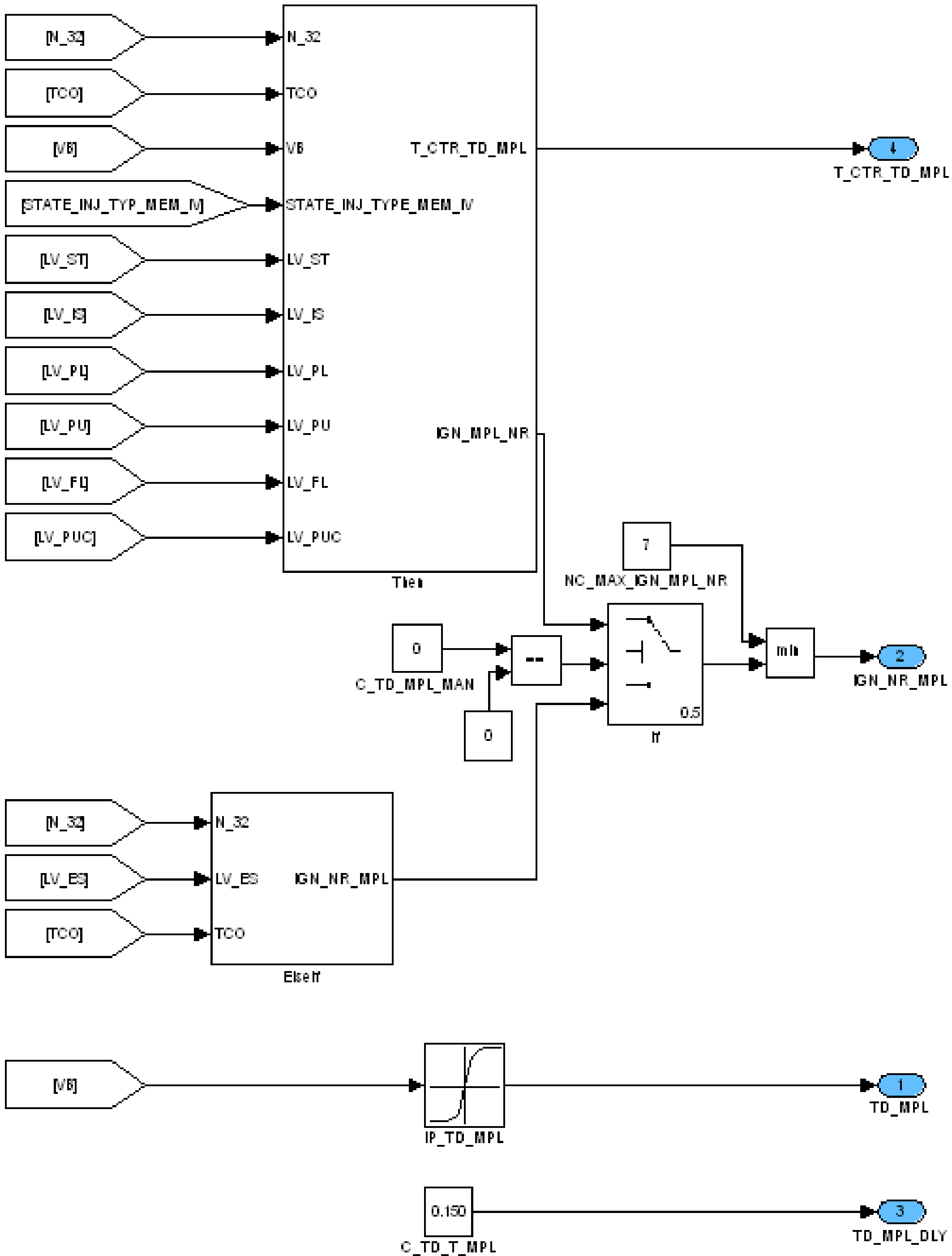
The MPL Sparks are maintained during a time (C_T_CTR_TD_MPL_AST or C_T_CTR_TD_MPL) after the end of the activation condition, before exiting this MPL procedure (*Application Multiple Spark Maintenance for a time before Exit Case*)

Application conditions:

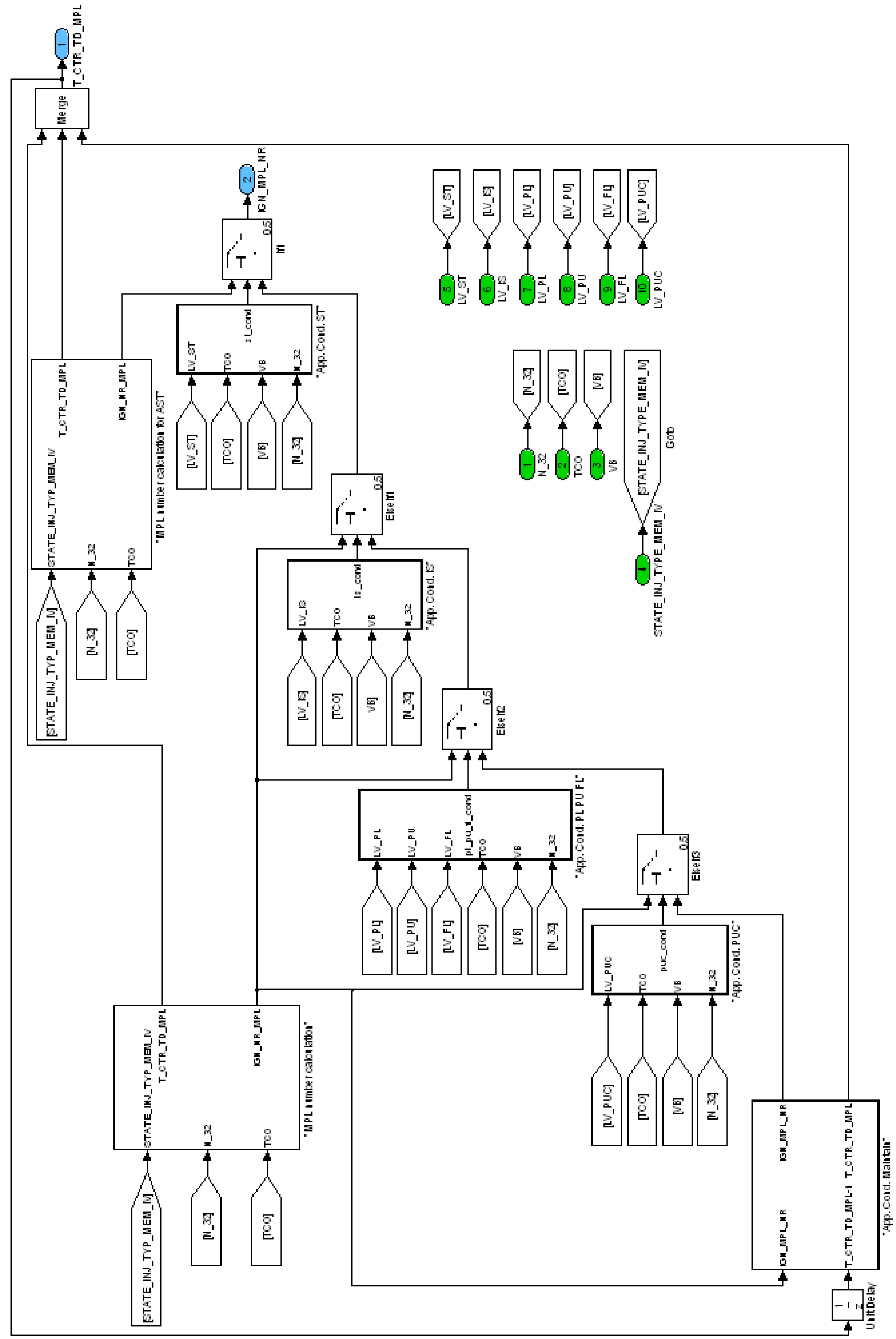
Initialisation:	IGN_MPL_NR	= 0	
	TD_MPL	= 0	
	TD_MPL_DLY	= 0	
	T_CTR_TD_MPL	= 0	*At reset*

Recurrence: 10 ms

Activation:



General Multiple Spark Activation and calculation



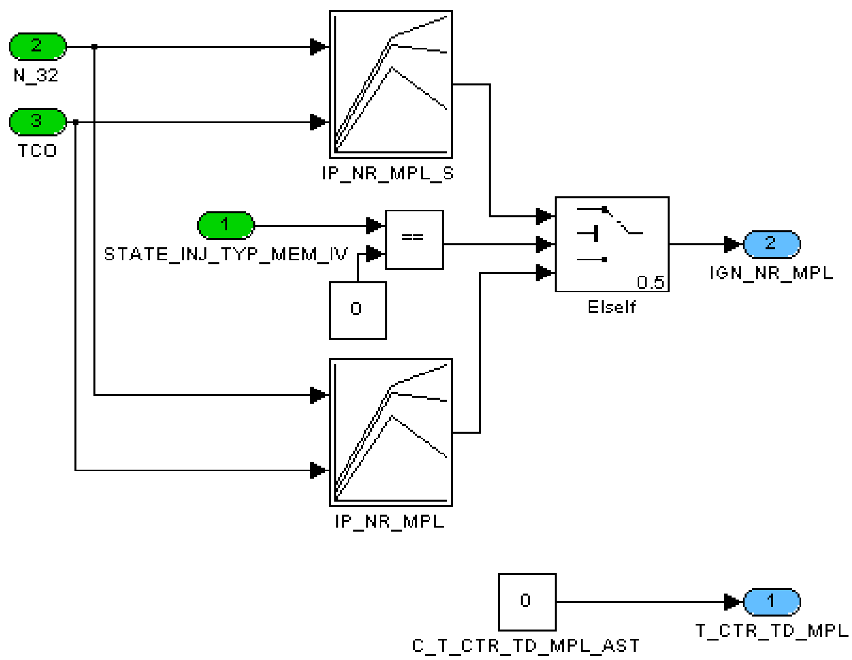
Case Then

Chapter	Baseline		Include File
Ignition	691F00		30600301.00E
	Date	Department	Sign
Designed by GC Shin	2008-05-27	SV P GS ES	
Released by G. Raab	2008-05-27	SV P GS Sys2 PL	

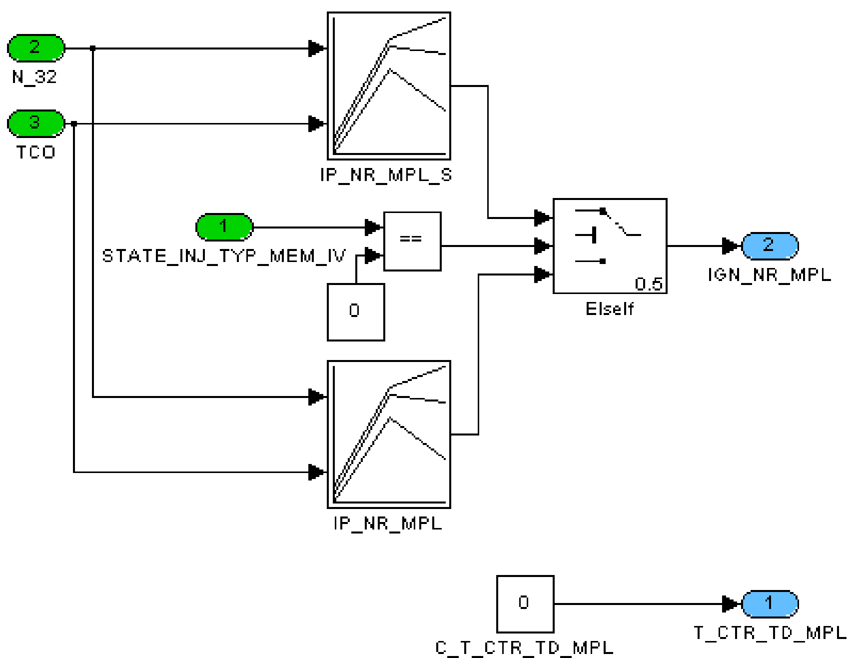
Continental

Designation
Engine Management System HMC Theta II ETC/BIN

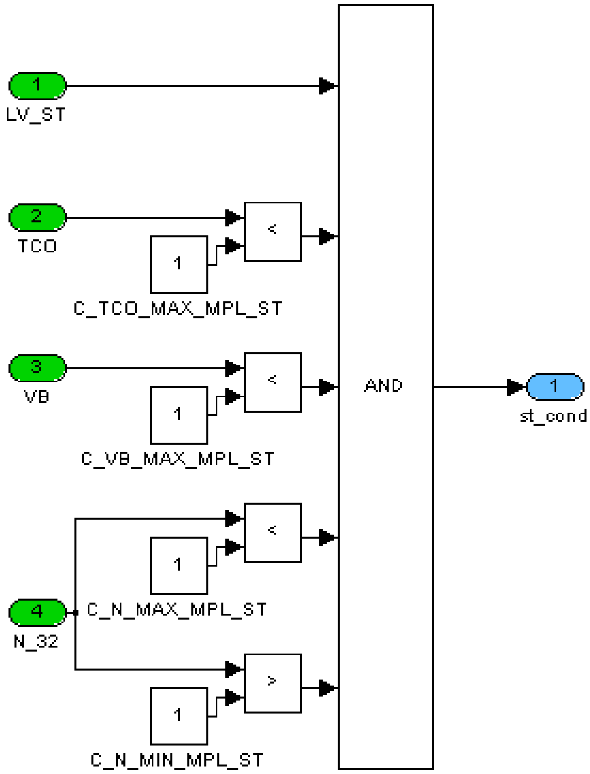
Document Key
E150-024.49.01 SPE 000 20.0



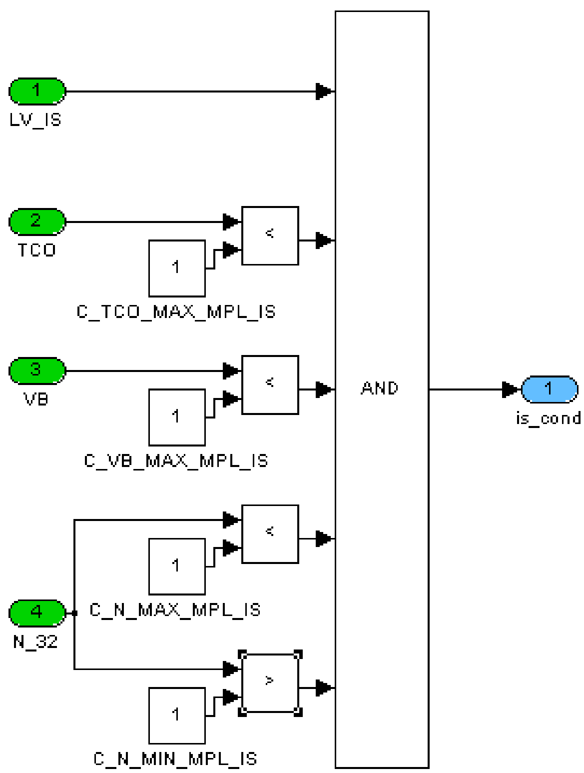
Multiple Spark Number Calculation for Start Case



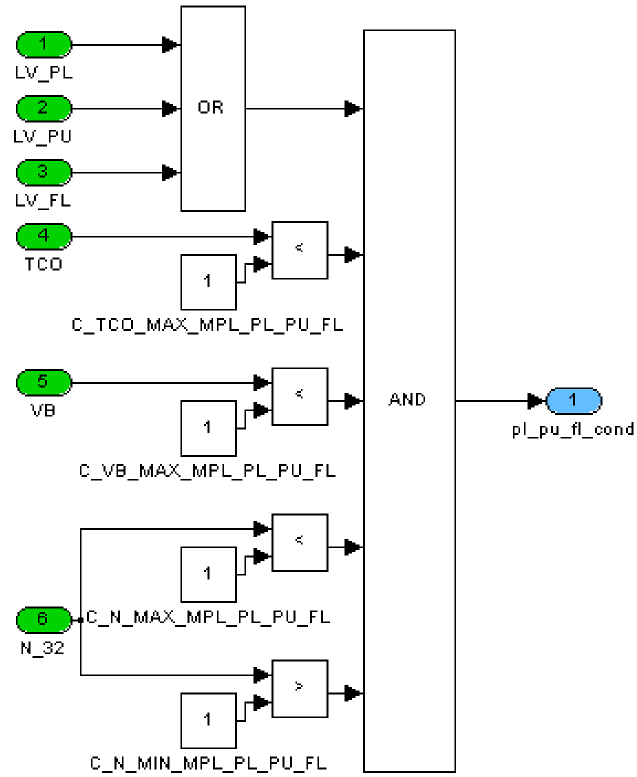
Multiple Spark Number Calculation for Other Cases



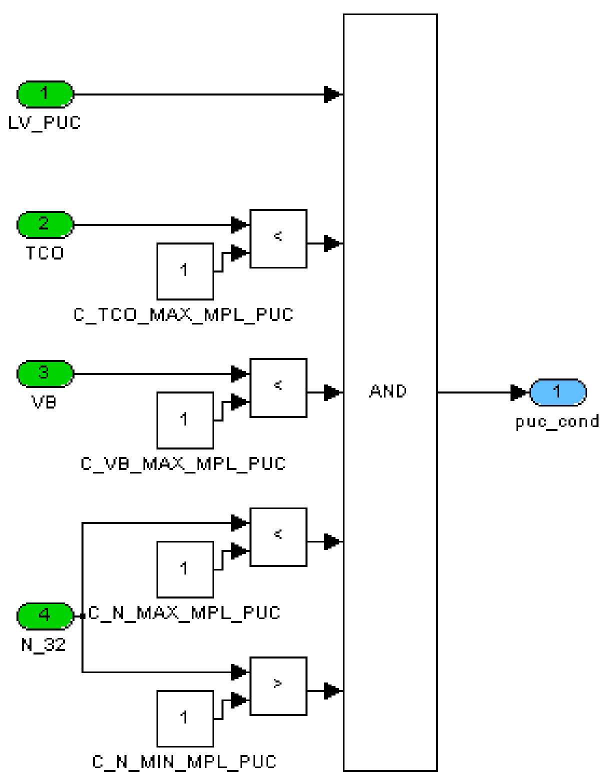
Activation Case: Start



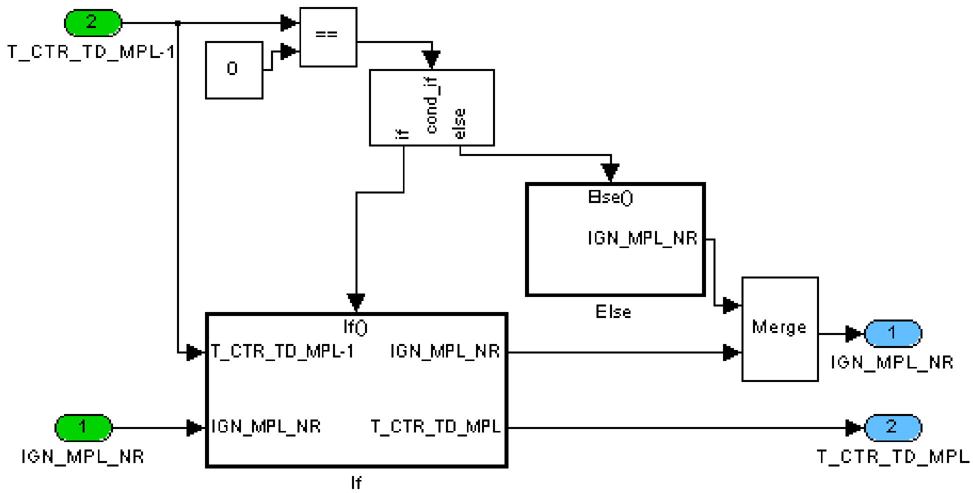
Activation Case: Idle



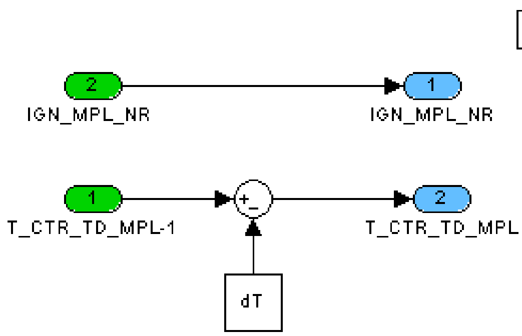
Activation Case: PL, PU, FL



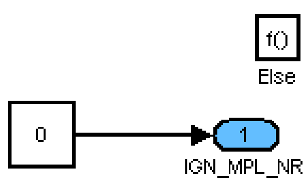
Activation Case: PUC



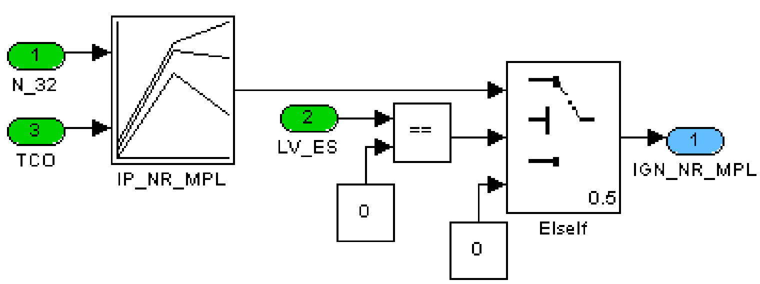
Activation Case: Maintenance time before exiting the Multiple Spark Activation



Activation Case: Maintenance time before exiting the Multiple Spark Activation - If



Activation Case: Maintenance time before exiting the Multiple Spark Activation - Else



Case: ElseIf

6.8.2 Settings of multiple spark parameters

Application conditions:

Initialisation: *At reset* **ACTION_INFR_SetIgnMplNr**(IGN_MPL_NR)

Recurrence: 10 ms

Formula section:

If IGN_MPL_NR = 0

Then **ACTION_INFR_SetIgnMplNr**(IGN_MPL_NR)

Else **ACTION_INFR_SetIgnMplDwell**(TD_MPL)

ACTION_INFR_SetIgnMplDly(TD_MPL_DLY)

ACTION_INFR_SetIgnMplNr(IGN_MPL_NR)

Endif

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TCO_MAX_MPL_ST	1	0...FEH	-48...142.5	0.75	°C
Max. Coolant Temp. to have multiple spark at start					
C_TCO_MAX_MPL_IS	1	0...FEH	-48...142.5	0.75	°C
Max. Coolant Temp. to have multiple spark at Idle speed					
C_TCO_MAX_MPL_PL_PU_FL	1	0...FEH	-48...142.5	0.75	°C
Max. Coolant Temp. to have multiple spark at Run					
C_TCO_MAX_MPL_PUC	1	0...FEH	-48...142.5	0.75	°C
Max. Coolant Temp. To have multiple spark at fuel cut off					
C_VB_MAX_MPL_ST	1	0...FFH	0...26	0.102	V
Max. Battery voltage to have multiple spark (requested value 16V) at start					
C_VB_MAX_MPL_IS	1	0...FFH	0...26	0.102	V
Max. Battery voltage to have multiple spark (requested value 16V) at idle speed					
C_VB_MAX_MPL_PL_PU_FL	1	0...FFH	0...26	0.102	V
Max. Battery voltage to have multiple spark (requested value 16V) at Run					
C_VB_MAX_MPL_PUC	1	0...FFH	0...26	0.102	V
Max. Battery voltage to have multiple spark (requested value 16V) at fuel cut off					
C_N_MIN_MPL_ST	1	0...FFH	0...8160	32	rpm
Engine speed min for ignition with multiple spark at Start					
C_N_MAX_MPL_ST	1	0...FFH	0...8160	32	rpm
Engine speed max for ignition with multiple spark at Start					
C_N_MIN_MPL_IS	1	0...FFH	0...8160	32	rpm
Engine speed min for ignition with multiple spark at Idle					
C_N_MAX_MPL_IS	1	0...FFH	0...8160	32	rpm
Engine speed max for ignition with multiple spark at idle					
C_N_MIN_MPL_PL_PU_FL	1	0...FFH	0...8160	32	rpm
Engine speed min for ignition with multiple spark at Run					
C_N_MAX_MPL_PL_PU_FL	1	0...FFH	0...8160	32	rpm
Engine speed max for ignition with multiple spark at Run					
C_N_MIN_MPL_PUC	1	0...FFH	0...8160	32	rpm
Engine speed min for ignition with multiple spark at fuel cut off					
C_N_MAX_MPL_PUC	1	0...FFH	0...8160	32	rpm
Engine speed max for ignition with multiple spark at Fuel cut off					
C_TD_T_MPL	1	0...FFFFH	0...262,14	0,004	ms
Minimum time between two ignition (normal value 150µs)					
IP_TD_MPL	8	0...FFFFH	0...262,14	0,004	ms
LDPM_VB_1_IGRE	8	0...FFH	0...26	0.102	V
Dwell time for multiple spark in homogeneous mode confirmed					

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_TD_MPL_S	8	0...FFFFH	0...262,14	0,004	ms
LDPM_VB_1_IGRE	8	0...FFH	0...26	0.102	V
Dwell time for multiple spark in stratified mode confirmed					
IP_NR_MPL	8*8	0…FFH	0…255	1	-
LDPM_N_32_1_IGRE	8	0...FFH	0...8160	32	rpm
LDP_TCO_IP_NR_MPL	8	0...FEH	-48...142.5	0.75	°C
Number of successive ignition sparks depending on engine speed and TCO in homogeneous mode					
IP_NR_MPL_S	8*8	0…FFH	0…255	1	-
LDPM_N_32_1_IGRE	8	0...FFH	0...8160	32	Rpm
LDP_TCO_IP_NR_MPL	8	0...FEH	-48...142.5	0.75	°C
Number of successive ignition sparks depending on engine speed and TCO in stratified mode					
C_TD_MPL_MAN	1	0...1H	0....1	1	-
Manual activation of the MPL					
C_T_CTR_TD_MPL_AST	1	0...FFFFH	0...655,35	0,01	s
Maintenance Time of Multiple Spark activation before Out Case and for the After Start phase					
C_T_CTR_TD_MPL	1	0...FFFFH	0...655,35	0,01	s
Maintenance Time of Multiple Spark activation before Out Case					

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_MAX_IGN_MPL_NR	1	0..15	0..F	1	-
Maximum multiple spark - Typical value for debug:7					
NC_MPL_T_MAX	1	0...FFFFH	0...262,14	0.004	ms
Maximum time duration of Multiple Spark (Normal value 60 ms)					
NC_MPL_IGN_CRK_MAX	1	0...FFH	0…95,625	0,375	°CRK
Maximum ignition angle after TDC to start TD_MPL (normal value 12 to 18 °CRK)					

6.9 Ignition Activation Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
INH_IGC	V/O	0...FFH	0...255	1	-
Inhibition of selected cylinder					

Input data:

LV_IMOB_IGC_OFF	LV_ENG_OFF_DMF_IGN	LV_IGN_INJ_LOCK_REQ	LV_INH_IGC_IGK

FUNCTION DESCRIPTION:

Application conditions:

Initialisation : at reset

Activation: engine synchronised

Output driver deactivation

General information

The decision to inhibit an ignition stage is taken for each output before the turn on at TD_ON_x.

Ignition shut off requested by Immobilizer:

if LV_IMOB_IGC_OFF = 1

then all ignition must be shut off: INH_IGC = xxxx1111

Ignition shut off requested by DMF Oscillation Detection:

if LV_ENG_OFF_DMF_IGN = 1

then all ignition must be shut off: INH_IGC = xxxx1111

Ignition shut off requested by Backward Engine Rotation Detection:

if LV_IGN_INJ_LOCK_REQ = 1

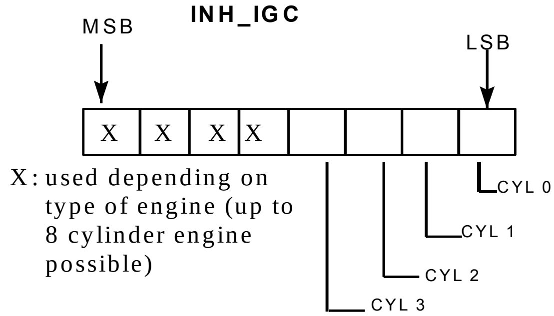
then all ignition must be shut off: INH_IGC = xxxx1111

Ignition shut off requested after ignition switch off:

if LV_INH_IGC_IGK = 1

then all ignition must be shut off: INH_IGC = xxxx1111

If one bit within INH_IGC is set to 1, then the corresponding ignition output is immediately inhibited. The ignition output remains inhibited while the INH_IGC is unchanged.


INH_IGC
MSB
LSB
X
X
X
X
X: used depending on type of engine (up to 8 cylinder engine possible)
CYL 0
CYL 1
CYL 2
CYL 3

6.10 Basic ignition angle at start (IGA_ST)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_ST	O/V	0..FFh	-35.625..60	0.375	°CRK
Basic ignition angle at start					

Input data:

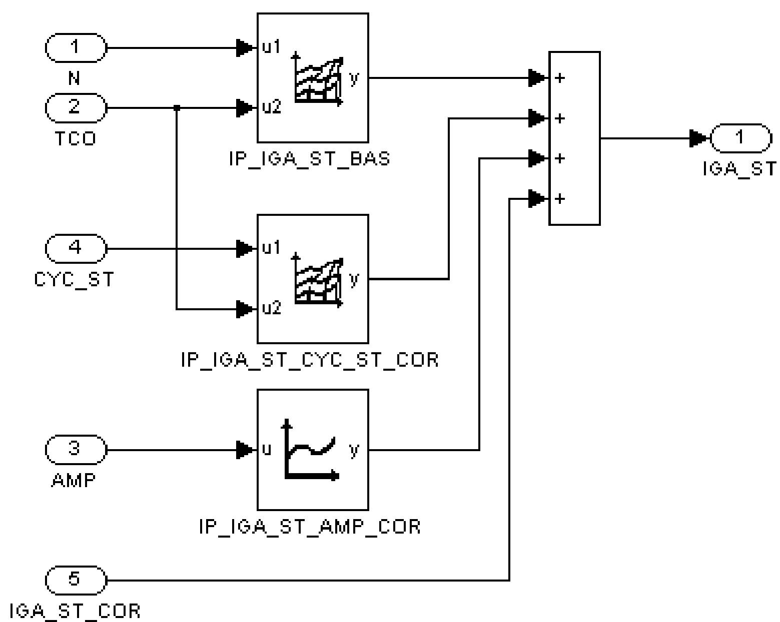
N	TCO	CYC_ST	AMP
LV_ST	IGA_ST_COR		

FUNCTION DESCRIPTION:

General information:

The ignition angle at start is basically dependent on the engine speed N and coolant temperature TCO determined by the map IP_IGA_ST_BAS. Additionally a map dependent on the ambient pressure AMP corrects this basic ignition angle. To avoid backfiring due to the fast increase of the combustion chamber temperature a second correction dependent on the cycle counter CYC_ST and the coolant temperature is added. IGA_ST_COR is added for project specific corrections.

Signal flow diagram:



Application conditions :

Activation : LV_ST = 1

Deactivation : LV_ST = 0

Initialisation : IGA_ST = 0°crk at reset

Update rate : every TDC and every 10 ms

Formula section :

IGA_ST = IP_IGA_ST_BAS
+ IP_IGA_ST_CYC_ST_COR
+ IP_IGA_ST_AMP_COR
+ IGA_ST_COR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_ST_BAS	6x8	0...FFH	-35.625...60	0.375	°CRK
LDP_N_IP_IGA_ST_BAS	6	0...1FE0H	0...8.16E3	1	rpm
LDPM_TCO_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
Basic ignition angle at start					
IP_IGA_ST_CYC_ST_COR	6x8	0...FFH	-48...47.625	0.375	°CRK
LDP_CYC_ST_IP_IGA_ST_CYC_ST_COR	6	0...FFH	0...255	1	-
LDPM_TCO_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
Correction of the basic ignition angle at start dependen on the start cycle counter					
IP_IGA_ST_AMP_COR	1x6	0...FFH	-48...47.625	0.375	°CRK
LDP_AMP_IP_IGA_ST_AMP_COR	6	0...FFFFH	0...5434	0.083	hPa
Ambient pressure correction of the basic ignition angle at start					

6.11 Ingition angle correction due to Port flap influence

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_BAS_PORT_COR_i	V/O	80...7FH	-48...47.6	0.375	°CRK
basic ignition angle correction with active port flap					
IGA_REF_PORT_COR	V/O	80...7FH	-48...47.6	0.375	°CRK
reference ignition angle correction with active port flap					

Input data:

FAC_PORT_DEAC	N_32	MAF_HB	LV_HOM_RUN
CONF_PORT	NC_CBK_NR		

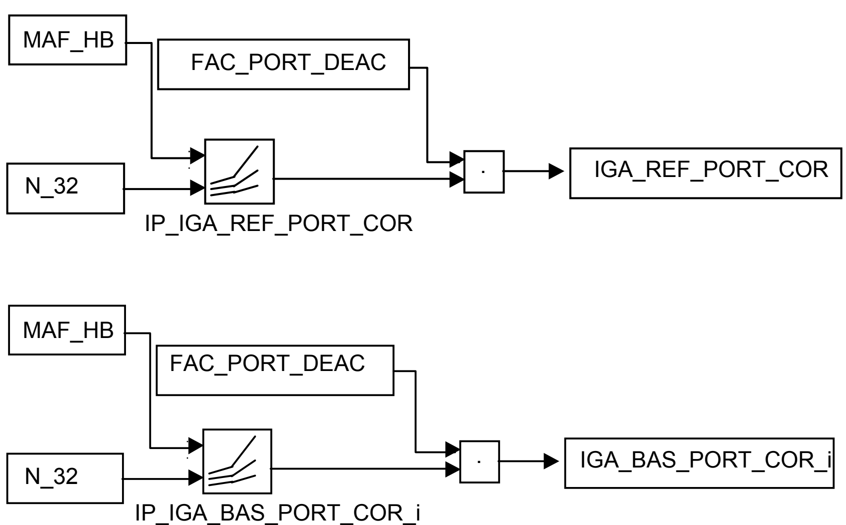
FUNCTION DESCRIPTION:

For engines equipped with a port flap it is necessary to have an ignition angle correction depending on the position of the port flap. This correction is an additive correction on the basic and the reference ignition angle (--> “corrected basic ignition angle” and “corrected reference ignition angle”).

Therefore the basic and reference ignition angle is to be calibrated with open (deactivated) port flap (FAC_PORT_DEAC = 0) and afterwards this correction has to be applied depending on load point (N, MAF) and port flap position (FAC_PORT_DEAC)

For systems with cylinder bank selective ignition angles (NC_CBK_NR = 2) this function is bank selective too. _i represents the cylinder bank. The ignition correction for the reference ignition angle (IGA_REF_PORT_COR) must not be cylinder bank selective, as the reference ignition angle is not bank selective. So IGA_REF_PORT_COR is calculated via FAC_PORT_COR_MV

Signal flow diagram:



Application conditions:

Initialisation: IGA_BAS_PORT_COR_i = IGA_REF_PORT_COR = 0
at reset and with LV_IGK = 0 --> 1

Recurrence: 10 ms

Activation: LV_HOM_RUN = 1
and CONF_PORT = 1

Deactivation: LV_HOM_RUN = 0
or CONF_PORT = 0

Formula section:

If N_32 ≤ C_N_THD_PORT_IGA_COR

then IGA_BAS_PORT_COR_1 = IP_IGA_BAS_PORT_COR_1 * FAC_PORT _DEAC

NC_CBK_NR = 2

IGA_BAS_PORT_COR_2 = IP_IGA_BAS_PORT_COR_2 * FAC_PORT _DEAC

end

IGA_REF_PORT_COR = IP_IGA_REF_PORT_COR * FAC_PORT_DEAC

else IGA_BAS_PORT_COR_1 = 0

NC_CBK_NR = 2

IGA_BAS_PORT_COR_2 = 0

end

IGA_REF_PORT_COR = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_BAS_PORT_COR_i	16*12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_32__IGA	16	0...FFH	0...8160	32	rpm
LDPM_MAF_HB__IGA	12	0...FFH	0...1389	5.447	mg/tdc
Basic Ignition angle correction for port flap activation (bank selective)					
IP_IGA_REF_PORT_COR	16*12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_32__IGA	16	0...FFH	0...8160	32	rpm
LDPM_MAF_HB__IGA	12	0...FFH	0...1389	5.447	mg/tdc
Reference Ignition angle correction for port flap activation					
C_N_THD_PORT_IGA_COR	1	0...FFH	0...8160	32	rpm
Engine speed threshold for activation					

6.12 Cylinder balancing via IGA intervention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_ADD_AD_ER_BAL[NC_CYL_NR]	O/V/S	80...7FH	-48...47.625	0.375	°CRK
Additive adaption value for cylinder balancing					
IGA_ADD_ER_BAL[NC_CYL_NR]	O/V	80...7FH	-48...47.625	0.375	°CRK
Additive correction value for cylinder balancing					
LV_IGA_ADD_ER_BAL_OBD_MAX_POS[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Flag for maximum OBD limit reached for cylinder balancing via IGA intervention					
LV_IGA_ADD_ER_BAL_OBD_MAX_NEG[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Flag for minimum OBD limit reached for cylinder balancing via IGA intervention					
ER_STND_IGA_ADD_ER_BAL	V	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Controller input value at additive path of cylinder balancing					
LV_ER_STND_IGA_ADD_ER_BAL	V	0...1H	0...1	1	-
Flag for controller input at additive path of cylinder balancing					
IGA_ADD_ER_BAL_CTL	V	80...7FH	-48...47.625	0.375	°CRK
Controller output value at additive path of cylinder balancing					
IGA_ADD_ER_BAL_COR[NC_CYL_NR]	V	80...7FH	-48...47.625	0.375	°CRK
Corrected additive correction value for cylinder balancing					
NR_CYC_IGA_ADD_ER_BAL_CTL_MAX	V	0...FFH	0...255	1	-
corresponding segment number of the controller that has reached the maximum limit (additive path)					
IGA_ADD_ER_BAL_LIM[NC_CYL_NR]	V	80...7FH	-48...47.625	0.375	°CRK
Limited controller output value at additive path of cylinder balancing					
IGA_ADD_ER_BAL_CTL_I[NC_CYL_NR]	V	80...7FH	-48...47.625	0.375	°CRK
I-part of the controller output value at additive path of cylinder balancing					
IGA_ADD_ER_BAL_CTL_MAX	V	0...FFFFH	0...6.5535E+4	1	-
Controller output value maximum limit reached at additive path of cylinder balancing					
IGA_ADD_ER_BAL_CTL_P	V	80...7FH	-48...47.625	0.375	°CRK
P-part of the controller output value for cylinder balancing via IGA intervention					

Input data:

ER_STND_MMV_DIF_BAL[NC_CYL_NR]	LV_IGA_ADD_ER_BAL_ENA	SEG_NR_ER	NC_CYL_NR
LV_IGA_ER_BAL_ACT	LV_IGA_ADD_AD_ER_BAL_ENA	LV_IGA_ADD_AD_ER_BAL_EXT_ADJ	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CYCNR_IGA_ADD_ER_BAL	1	1...FFH	1...255	1	-
Number of cycles for calculation of controller output values for cylinder balancing via IGA intervention					
C_ER_STND_THD_NEG_IGA_ADD_BAL	1	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Negative threshold for additive path of cylinder balancing					
C_ER_STND_THD_POS_IGA_ADD_BAL	1	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Positive threshold for additive path of cylinder balancing					
C_IGA_ADD_ER_BAL_CTL_MAX_NEG	1	80...7FH	-48...47.625	0.375	°CRK
Negative maximum limit for controller output value at additve path of cylinder balancing					
C_IGA_ADD_ER_BAL_CTL_MAX_POS	1	80...7FH	-48...47.625	0.375	°CRK
Positive maximum limit for controller output value at additve path of cylinder balancing					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_IGA_ADD_ER_BAL_OBD_MAX_NEG	1	80...7FH	-48...47.625	0.375	°CRK
Negative maximum OBD limit for correction value of cylinder balancing via IGA intervention					
C_IGA_ADD_ER_BAL_OBD_MAX_POS	1	80...7FH	-48...47.625	0.375	°CRK
Positive maximum OBD limit for correction value of cylinder balancing via IGA intervention					
C_IGA_ADD_ER_BAL_RST	1	0H 1H 2H	INI_ZERO INI_CTL INI_AD	1	-
Reset of additive correction and adaption values					
LC_IGA_ADD_AD_ER_BAL_RST_MAN	1	0...1H	0...1	1	-
Manual reset of the additive adaption values					
LC_IGA_ADD_ER_BAL_COR	1	0...1H	0...1	1	-
Manual switch for overall correction of controller output values for cylinder balancing via IGA intervention					
LC_IGA_ADD_ER_BAL_RST_MAN	1	0...1H	0...1	1	-
Manual reset of the additive correction values					
IP_IGA_ADD_I_ER_BAL	8	0...FFH	-48...47.625	0.375	°CRK
LDP_ER_STND_IGA_ADD_IP_I_ER_BAL	8	0...FFFFH	-325.78...325.77	0.0099420 2	1/s²
Basic index table for I-part of the controller at additive path of cylinder balancing					
IP_IGA_ADD_P_ER_BAL	8	0...FFH	-48...47.625	0.375	°CRK
LDP_ER_STND_IGA_ADD_IP_P_ER_BAL	8	0...FFFFH	-325.78...325.77	0.0099420 2	1/s²
Basic index table for P-part of the controller at additive path of cylinder balancing					

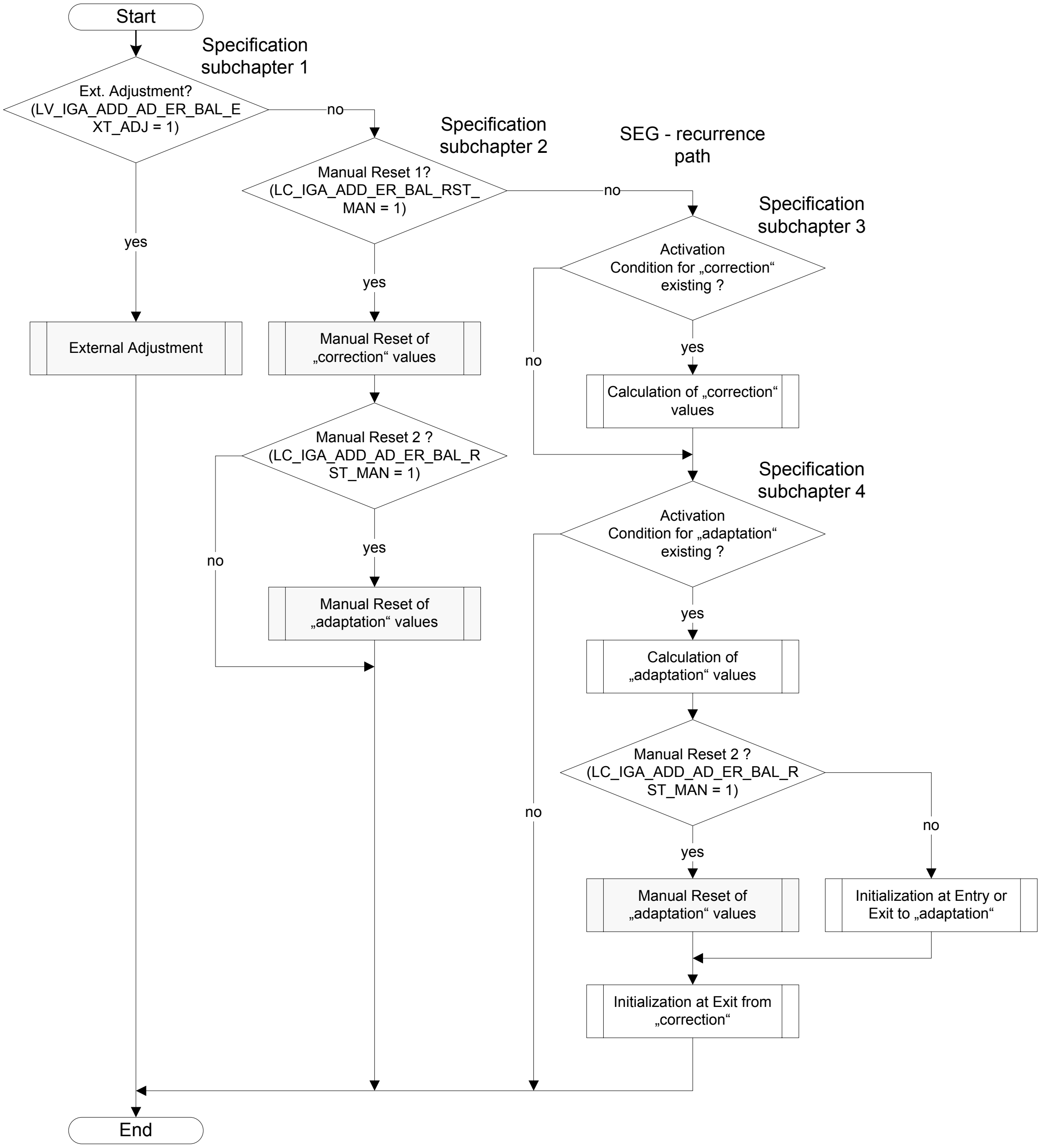
6.12.1 General information:

The calculation of the correction- or adaptation values is performed, depending on the settings of the corresponding logical activation variable. LV_IGA_ADD_ER_BAL_ENA is used to enable the correction path, while LV_IGA_ADD_AD_ER_BAL_ENA enables the adaptation path. As soon as the corresponding logical value is set to "1" in combination with additional activation conditions, the calculation is started.

The calculated additive adaptation value IGA_ADD_AD_ER_BAL[NC_CYL_NR] for each cylinder is stored in the non-volatile memory to be available for initialization issues at next engine run.

With use of the manual reset it is possible to initialize the additive correction- or adaptation values with zero at any time. In this case, the flags which indicate the exceeding of a controller output limitation are set to zero as well (refer: "Initialization at Manual Reset")

Signal flow diagram:


Start
Specification subchapter 1
Ext. Adjustment? (LV_IGA_ADD_AD_ER_BAL_EXT_ADJ = 1)
no
yes
External Adjustment
Specification subchapter 2
Manual Reset 1? (LC_IGA_ADD_ER_BAL_RST_MAN = 1)
yes
Manual Reset of „correction" values
Manual Reset 2 ? (LC_IGA_ADD_AD_ER_BAL_RST_MAN = 1)
no
yes
Manual Reset of „adaptation" values
SEG - recurrence path
Specification subchapter 3
Activation Condition for „correction" existing ?
yes
no
Calculation of „correction" values
Specification subchapter 4
Activation Condition for „adaptation" existing ?
yes
Calculation of „adaptation" values
Manual Reset 2 ? (LC_IGA_ADD_AD_ER_BAL_RST_MAN = 1)
yes
no
Manual Reset of „adaptation" values
Initialization at Entry or Exit to „adaptation"
Initialization at Exit from „correction"
End

Application Condition

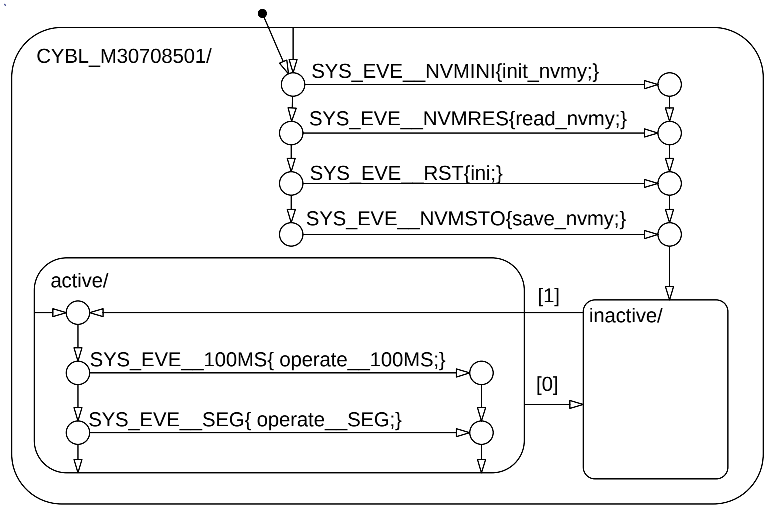


Figure 18 CYBL_M30602G02/ APP_CDN/ Chart

Function Description

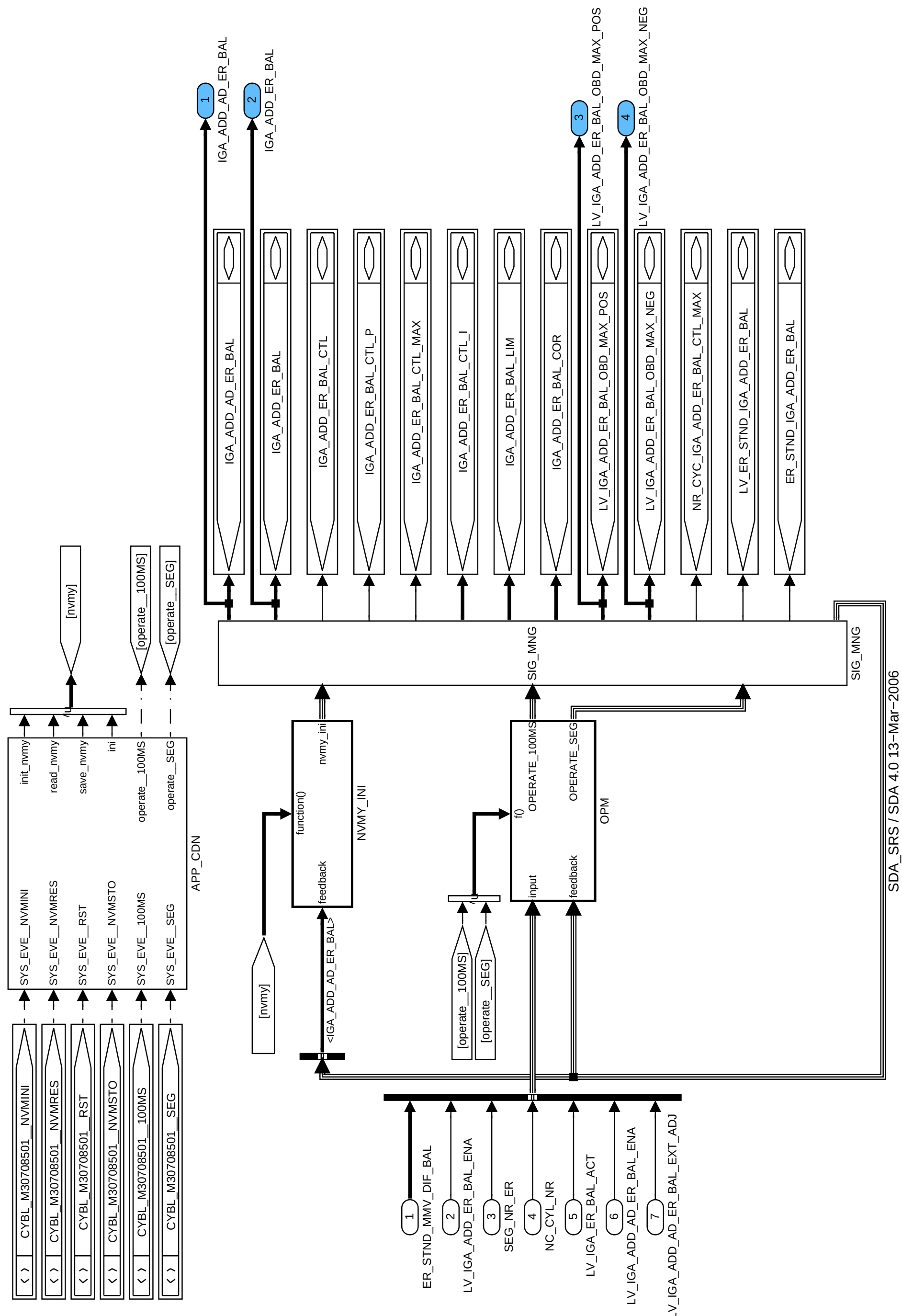


Figure 19 CYBL_M30602G02

6.12.1.1 Calculation of non volatile memory tasks

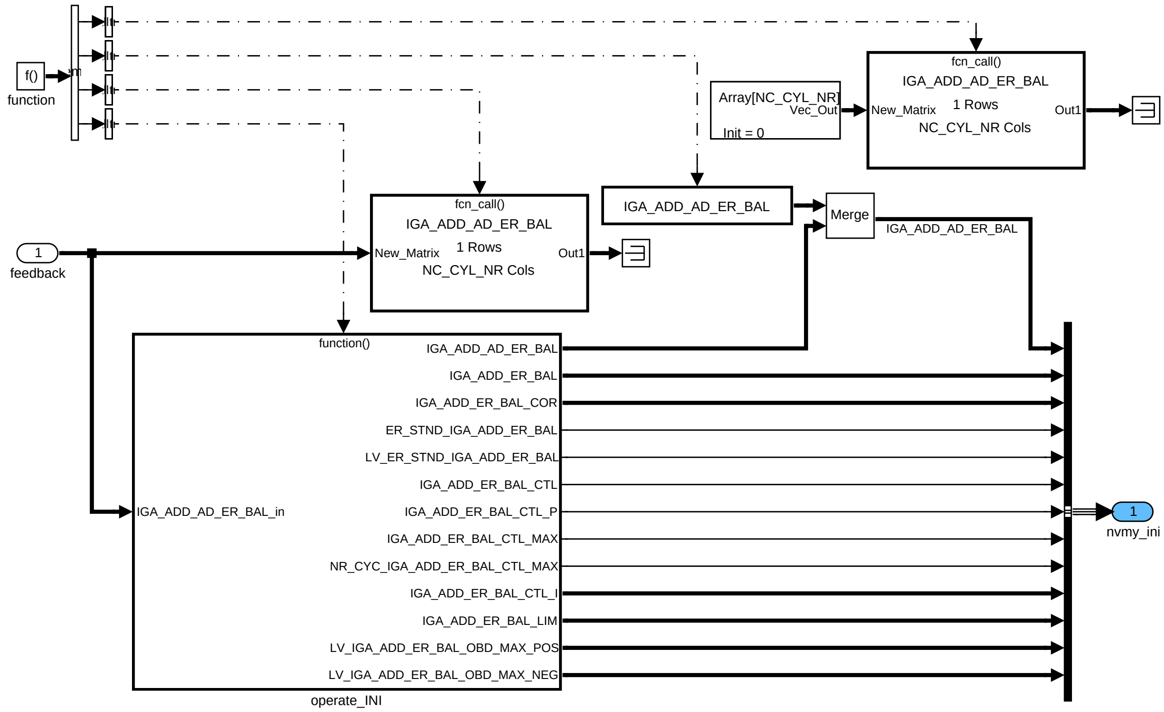


Figure 20 CYBL_M30602G02/ NVMY_INI

<u>Calculation of variables at reset task</u>

All other output data ("V" or "V/O") are set to "0" [phys].

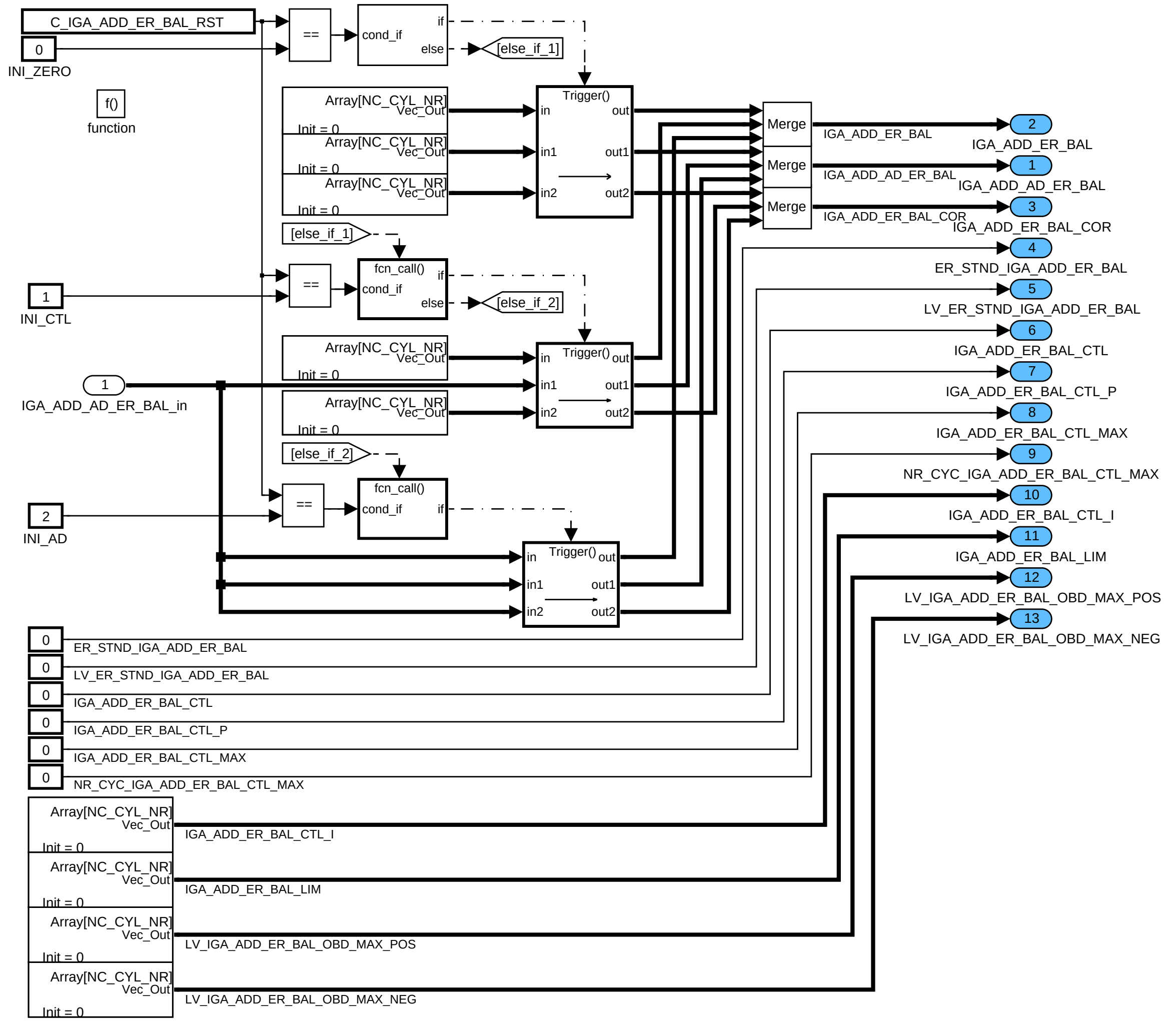


Figure 21 CYBL_M30602G02/ NVMY_INI/ operate_INI

6.12.1.2 Calculation of 100ms and segment synchronous tasks

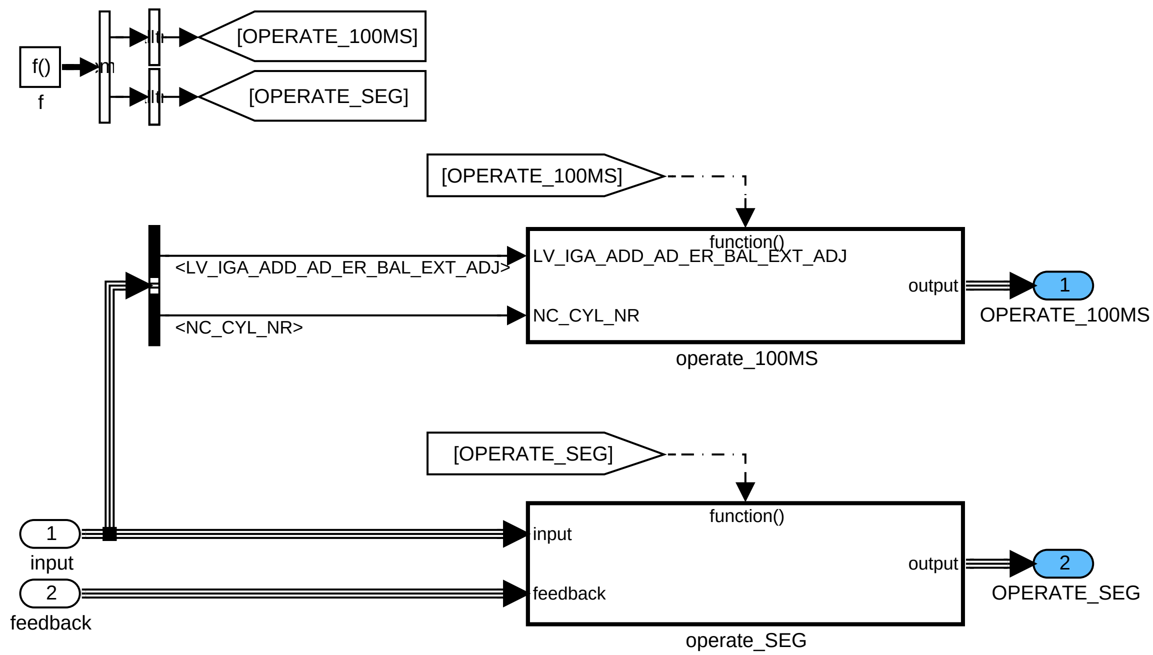


Figure 22 CYBL_M30602G02/ OPM

<u>External adjustment of correction values:</u>

In case of an external adjustment (request from key word protocol) the adaptation values for all cylinders are set to zero. While the external adjustment is activated, all following subchapters within the module "Cylinder balancing via IGA intervention" are deactivated and the corresponding "Initialization at Deactivation" process for this chapters is blocked during this period.

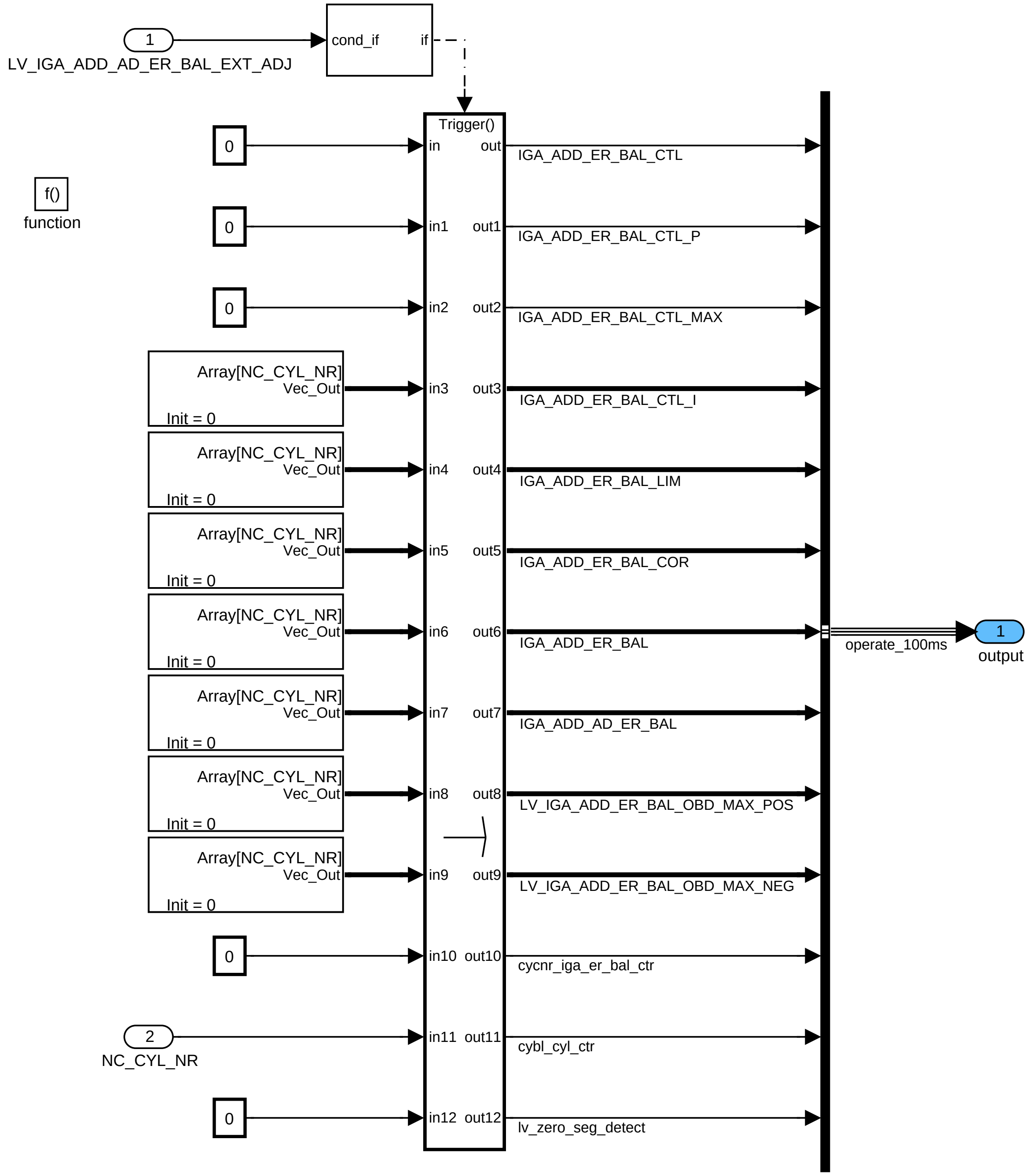


Figure 23 CYBL_M30602G02/ OPM/ operate_100MS

Calculation of segment tasks

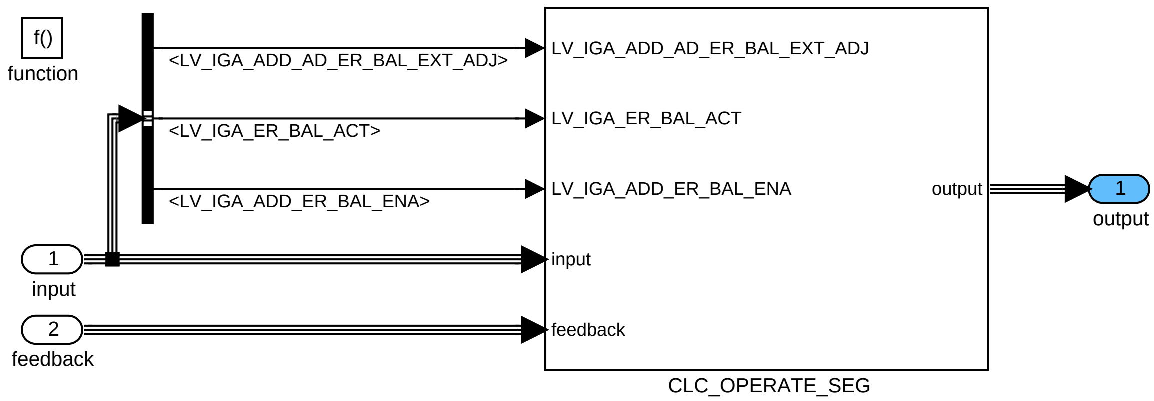


Figure 24 CYBL_M30602G02/ OPM/ operate_SEG

Calculation of segment tasks CLC_OPERATE_SEG

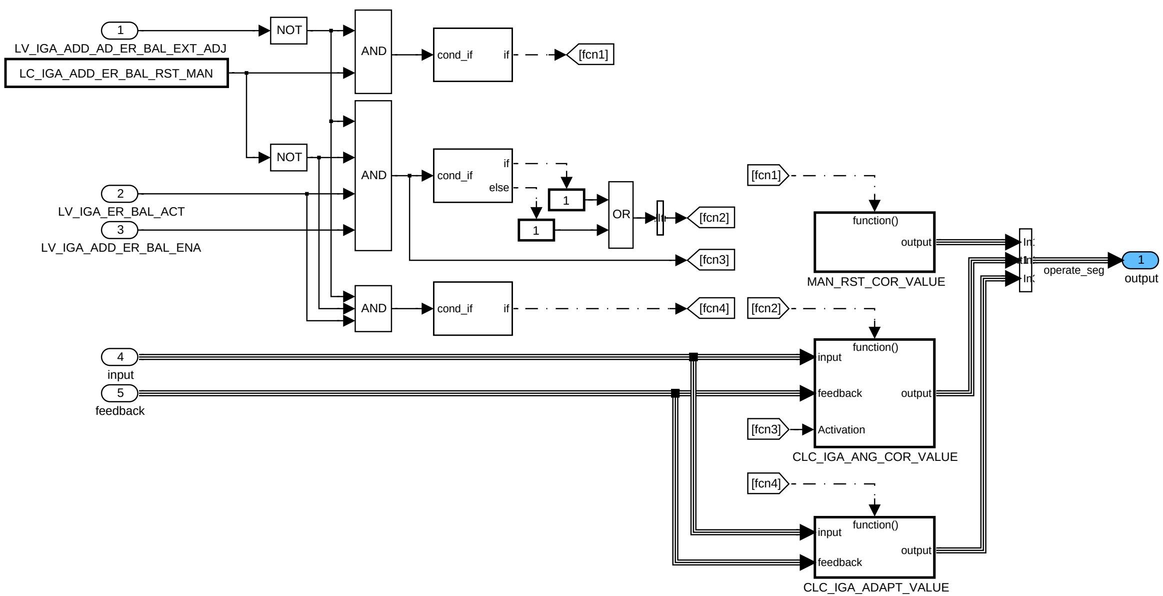


Figure 25 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG

Calculation of segment tasks CLC_OPERATE_SEG MAN_RST_COR_VALUE

In case of a manual reset it is possible to adjust the exported output data to a wanted value to make the function test and handling more easier. While the manual function reset is activated, all following subchapters within the module "Cylinder balancing via IGA intervention" are deactivated and the corresponding "Initialization at Deactivation" process for this chapters is blocked during this period.

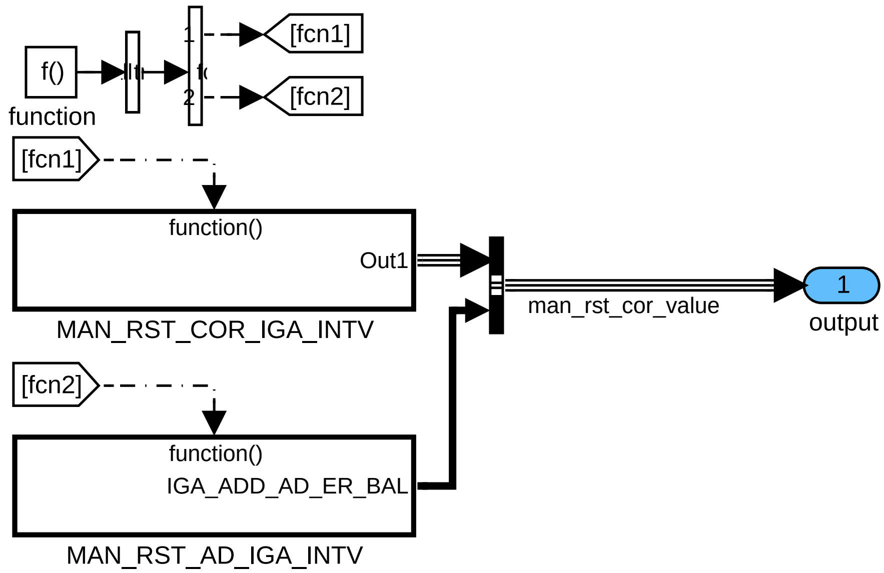


Figure 26 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ MAN_RST_COR_VALUE

Calculation of segment tasks CLC_OPERATE_SEG MAN_RST_COR_VALUE MAN_RST_COR_IGA_INTV

//Manual reset of correction values for cylinder balancing via IGA intervention

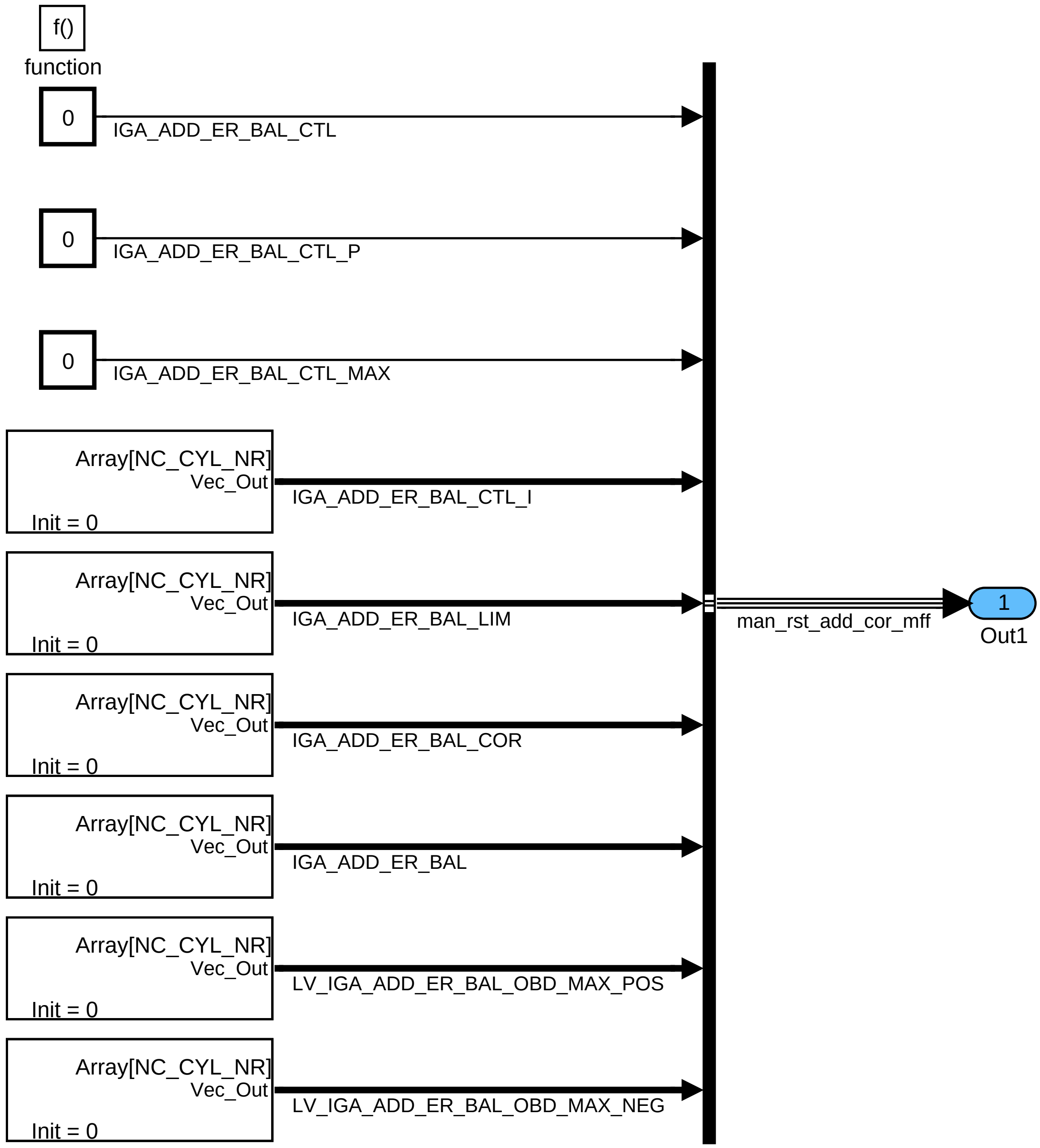


Figure 27 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ MAN_RST_COR_VALUE/ MAN_RST_COR_IGA_INTV

<u>Calculation of segment tasks CLC_OPERATE_SEG MAN_RST_COR_VALUE MAN_RST_AD_IGA_INTV</u>

//Manual reset of adaptation values for cylinder balancing via IGA intervention

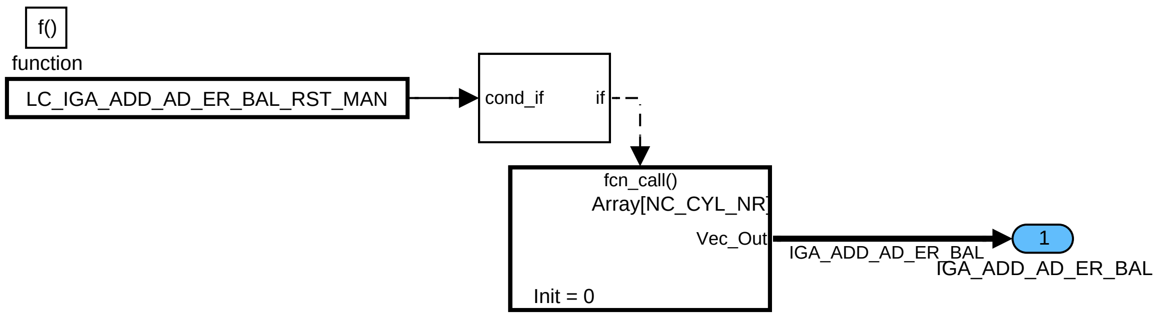


Figure 28 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ MAN_RST_COR_VALUE/ MAN_RST_AD_IGA_INTV

Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE

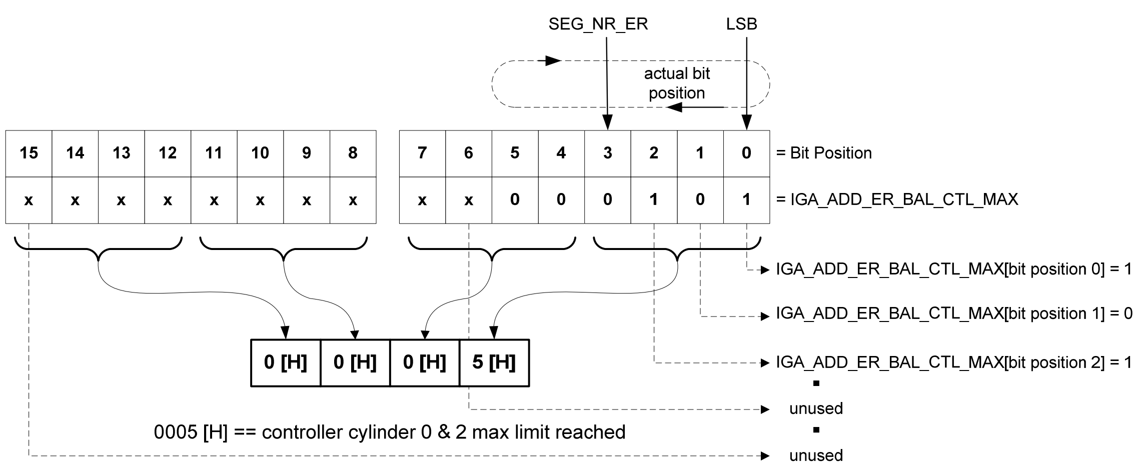
The calculation of the ignition angle correction values are only performed if the corresponding enable bit LV_IGA_ADD_ER_BAL_ENA is set to logical "1" in combination with additional activation conditions. In this case, a separate correction value for each cylinder is determined with use of a PI-controller.

As soon as a PI-controller of one cylinder is reaching its maximum, a corresponding value is set to "1" to indicate the related cylinder. To save memory resources, the information for all cylinders is located within one output data (WORD) is used to indicate one cylinder. It is possible to handle up to 16 cylinders with this method.

For a description more in detail please have a look to the signal flow diagram below. As example, a six-cylinder engine with two PI-controllers, which have reached the limits is shown.

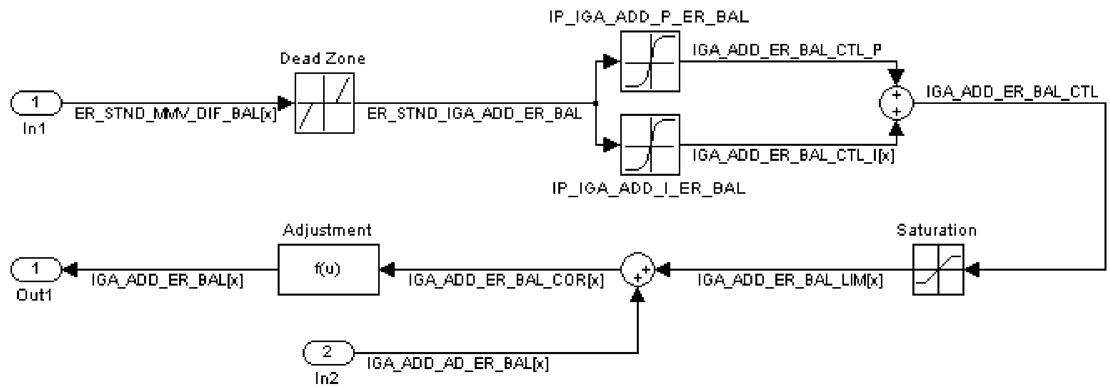
Signal flow diagram:

-Signalization of controller max.output value reached for ignition path



Note: always the complete WORD (16 BIT) is displayed as visible value

-Calculation of the ignition angle correction / adaptation values in general



Note: [_x] is representing [SEG_NR_ER] in the signal flow diagram above

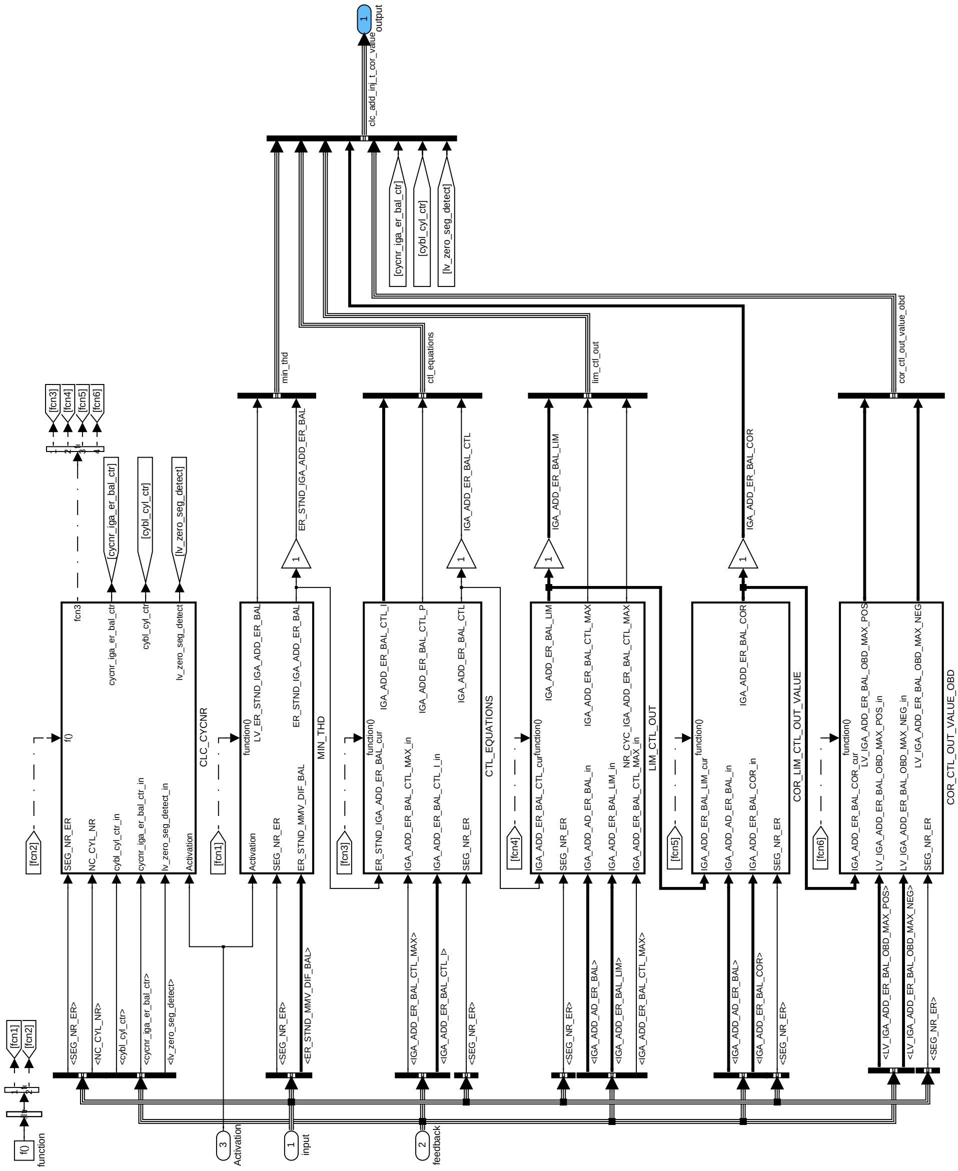


Figure 29 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE

<u>Minimum threshold:</u>

If the defined cylinder balancing input values ER_STND_IGA_DIF_BAL[SEG_NR_ER] are outside a calibrateable range, the controller is calculated. Otherwise the controller inputs are set to zero.

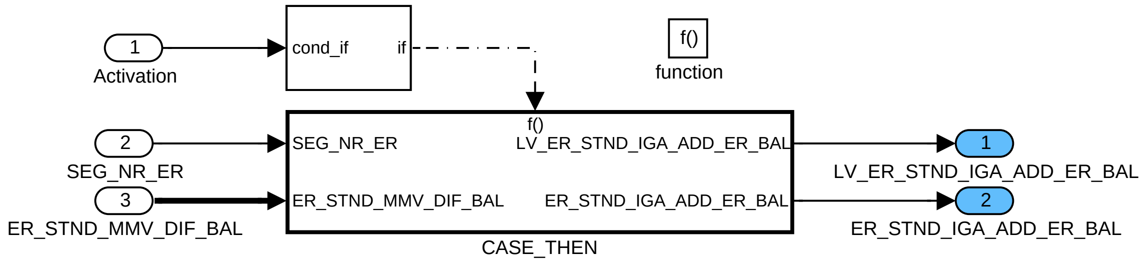


Figure 30 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ MIN_THD

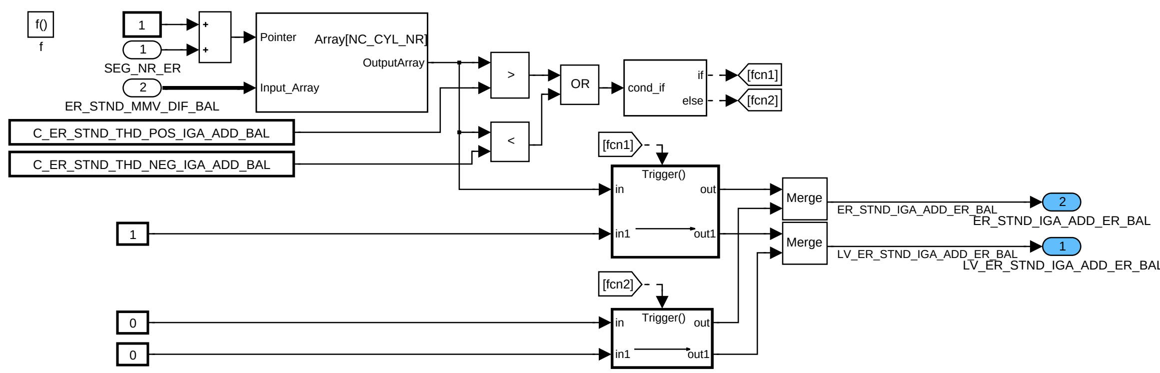


Figure 31 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ MIN_THD/ CASE_THEN

<u>Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE CLC_CYCNR</u>

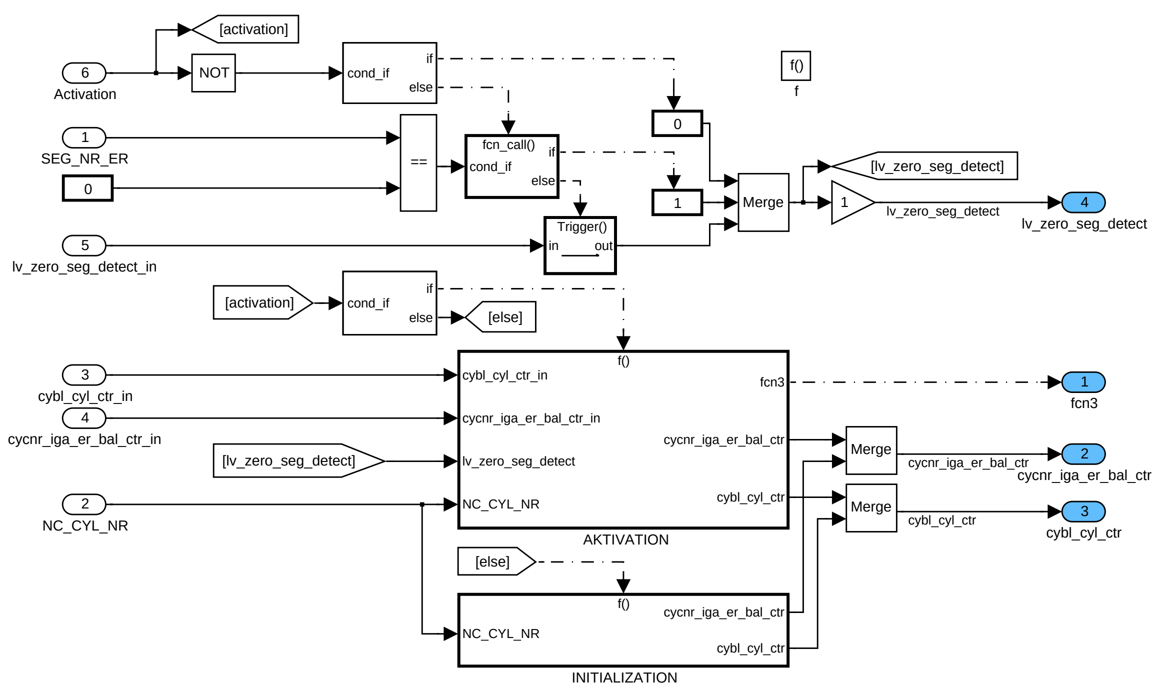


Figure 32 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CLC_CYCNR

Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE CLC_CYCNR AKTIVATION

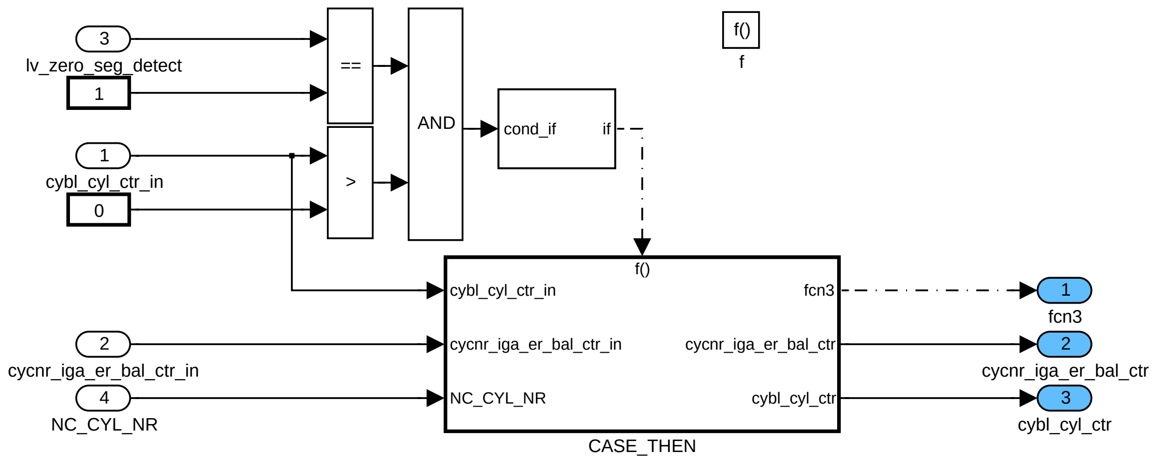


Figure 33 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CLC_CYCNR/ AKTIVATION

Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE CLC_CYCNR AKTIVATION CASE_THEN

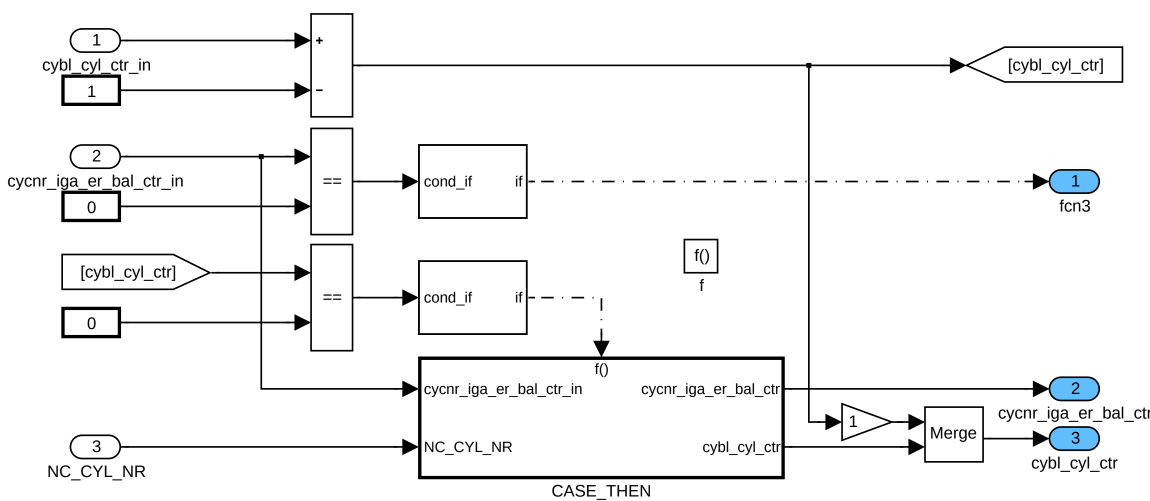


Figure 34 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CLC_CYCNR/ AKTIVATION/ CASE_THEN

Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE CLC_CYCNR AKTIVATION CASE_THEN CASE_THEN

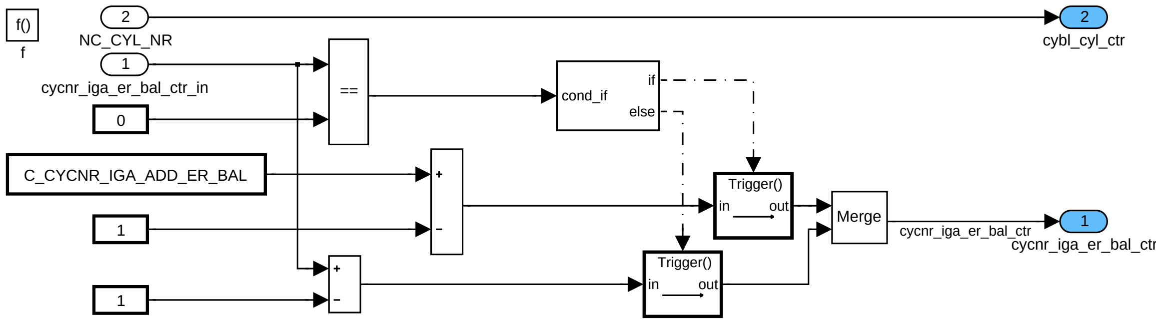


Figure 35 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CLC_CYCNR/ AKTIVATION/ CASE_THEN/ CASE_THEN

Calculation of segment tasks CLC_OPERATE_SEG CLC_IGA_ANG_COR_VALUE CLC_CYCNR INITIALIZATION

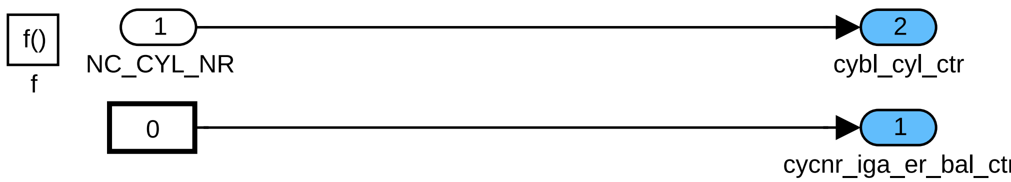


Figure 36 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CLC_CYCNR/ INITIALIZATION

Controller Equations

For the additive correction mode, a separate of NC_CYL_NR controllers exists operating in parallel (one controller for each cylinder).

The computation of the output values is carried out every C_CYCNR_IGA_ADD_ER_BAL engine cycles (= calibrateable number of engine cycles).

Only in case of C_CYCNR_IGA_ADD_ER_BAL = 1, the calculation of the additive path is done without interruption. At every segment the corresponding cylinder correction value is built.

If the computation is not done at every engine cycle (C_CYCNR_MFF_ADD_ER_BAL > 1), the trigger of the controller output calculation is always related to SEG_NR_ER = 0.

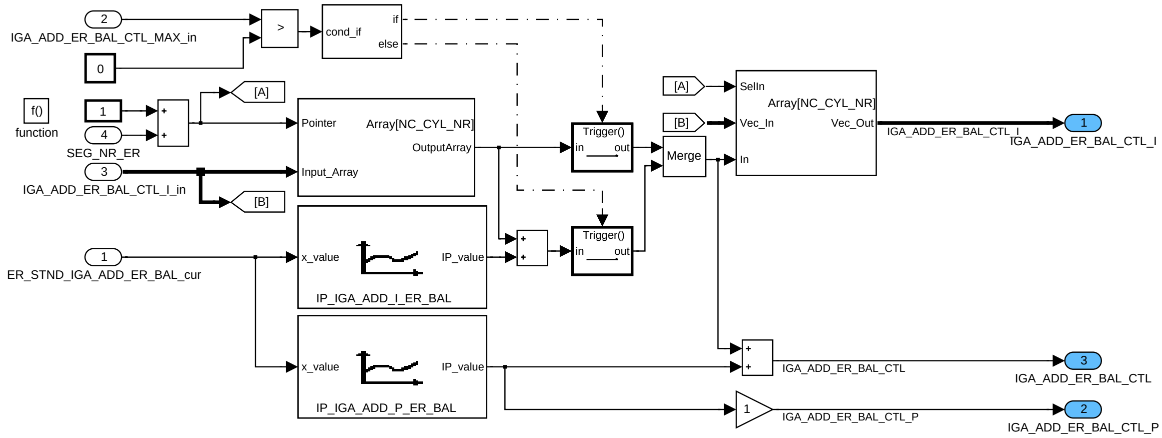


Figure 37 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ CTL_EQUATIONS

Limitation of the Controller Output

The outputs of the controllers are limited to a calibratable threshold. If one controller output has reached the limit, a flag is set to once and the corresponding segment number is displayed.

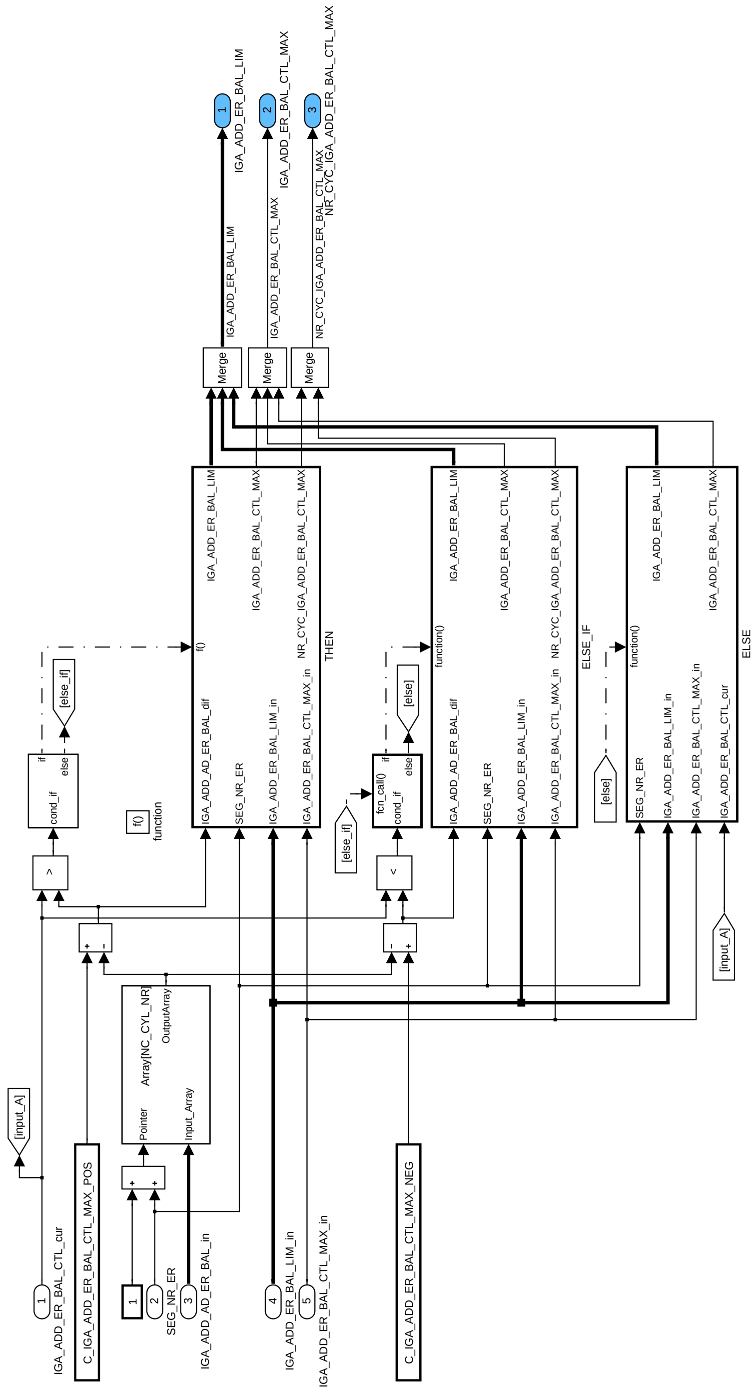


Figure 38 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ LIM_CTL_OUT

Limitation of the Controller Output THEN

(corresponding segment number where the controller has reached the limit)

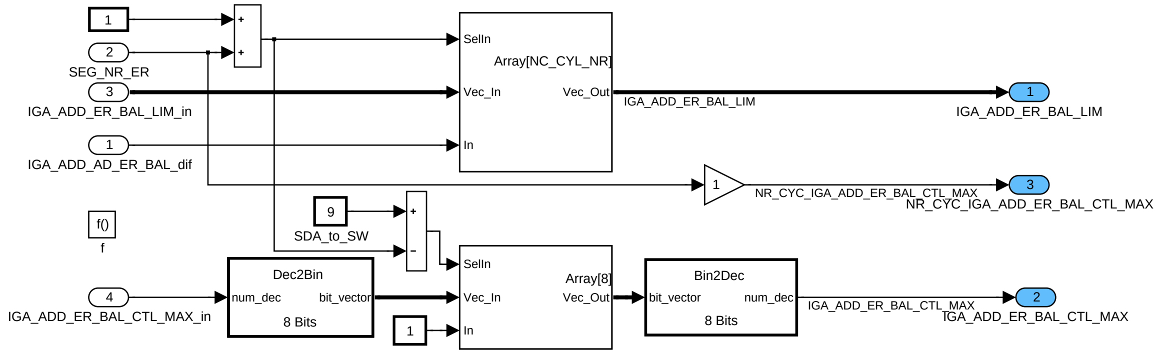


Figure 39 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ LIM_CTL_OUT/ THEN

Limitation of the Controller Output ELSE_IF

(corresponding segment number where the controller has reached the limit)

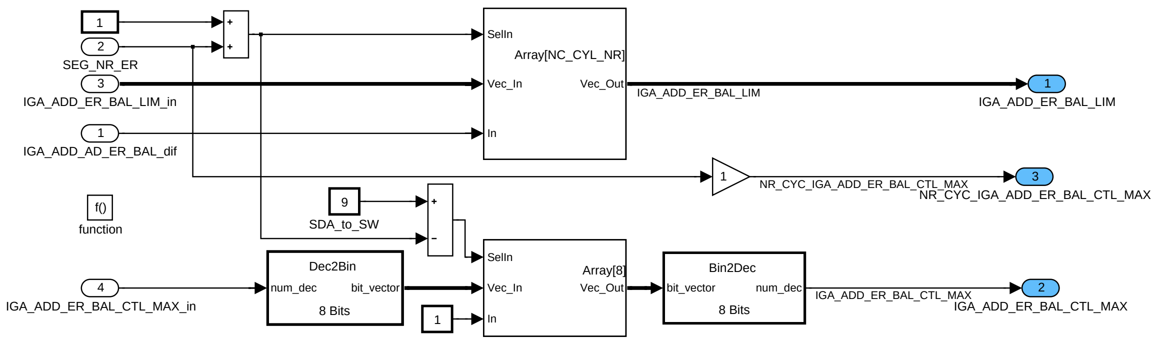


Figure 40 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ LIM_CTL_OUT/ ELSE_IF

Limitation of the Controller Output ELSE

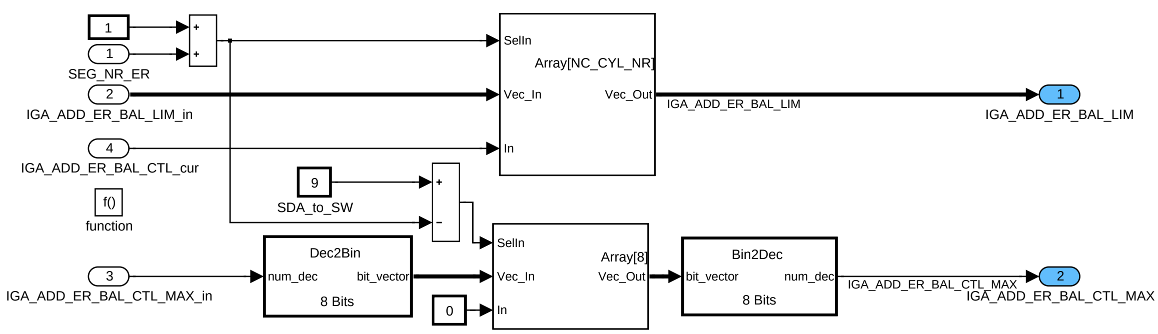


Figure 41 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ LIM_CTL_OUT/ ELSE

Correction of limited controller output values with adaptation values

The limited controller output values are corrected with the currently available adaptive correction values, which are saved in the NVMY and updated at Activation or Deactivation of the additive adaptation path (LV_IGA_ADD_AD_ER_BAL_ENA).

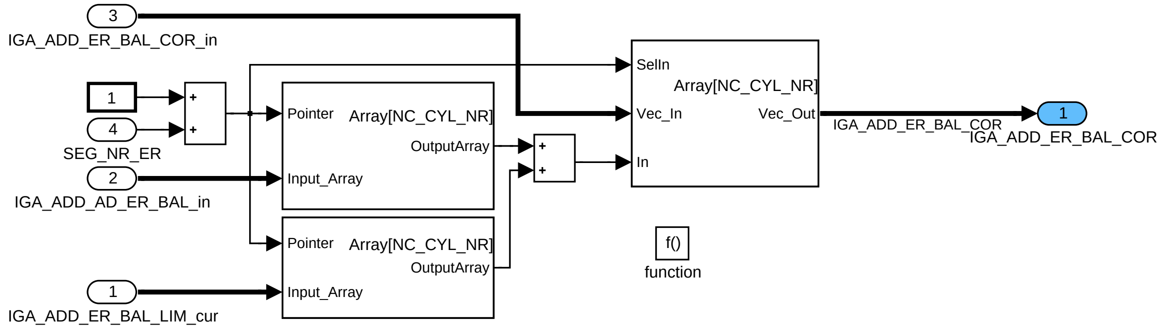


Figure 42 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ COR_LIM_CTL_OUT_VALUE

Observation of corrected controller output values for OBD issues

The additive correction values out of the correction path are observed for OBD purposes. As soon as the minimum maximum diagnosis threshold is exceeded, a corresponding flag for the related cylinder is set. In this case the OBD system is indicated to handle the further failure treatment. The diagnosis thresholds have always to be less or equal to the absolute minimum maximum controller output limits.

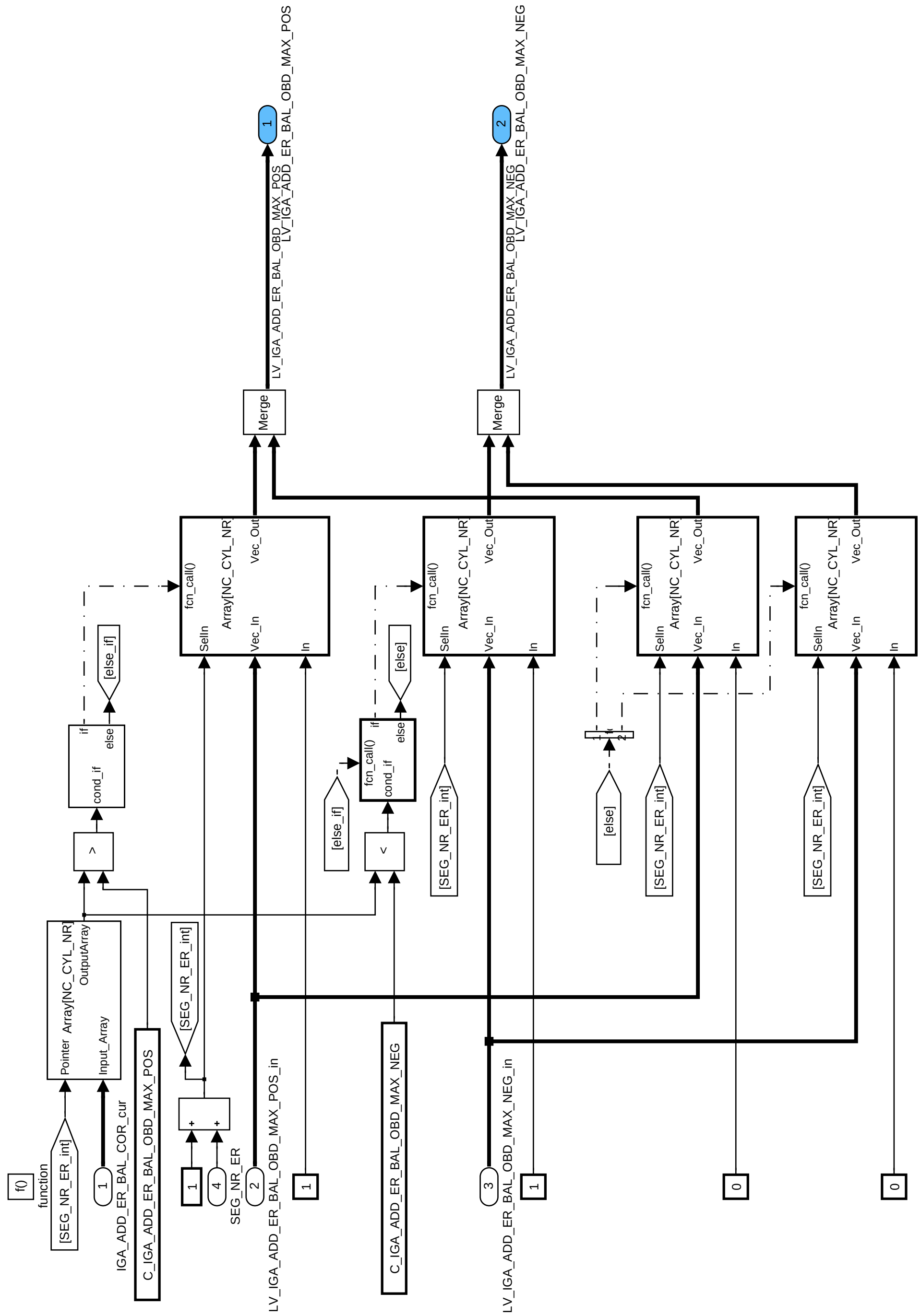


Figure 43 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ANG_COR_VALUE/ COR_CTL_OUT_VALUE_OBD

Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE

The corrected NC_CYL_NR controller output values are adjusted dependent on the corresponding segment number SEG_NR_ER. During the function is activated, at every new segment the corresponding IGA_ADD_ER_BAL value is updated. Due to this correction the sum of all adaptation output values remains zero to guarantee that the overall injection spreading stays unchanged.

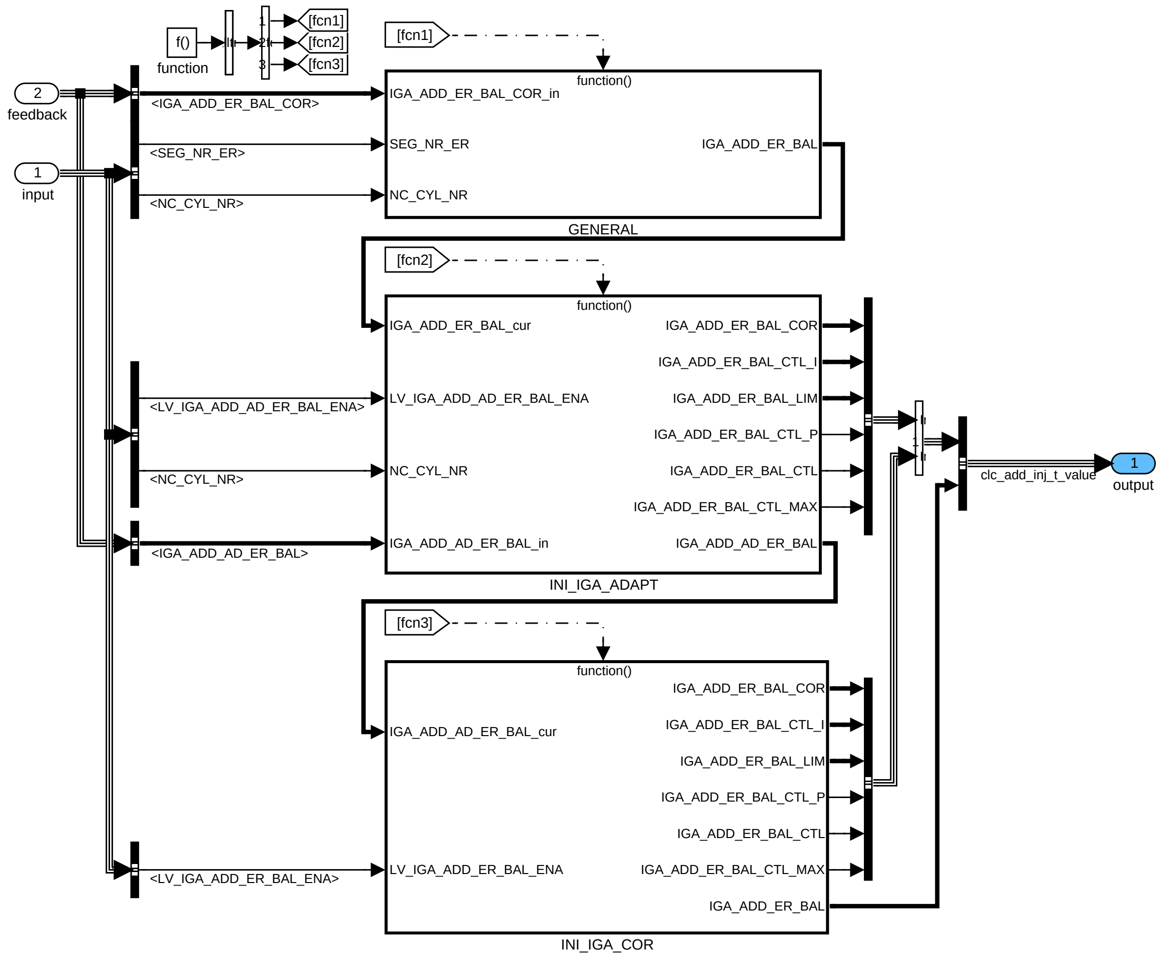


Figure 44 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE

<u>Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE GENERAL</u>

Sum of all NC_CYL_NR controller output values / number of cylinders.

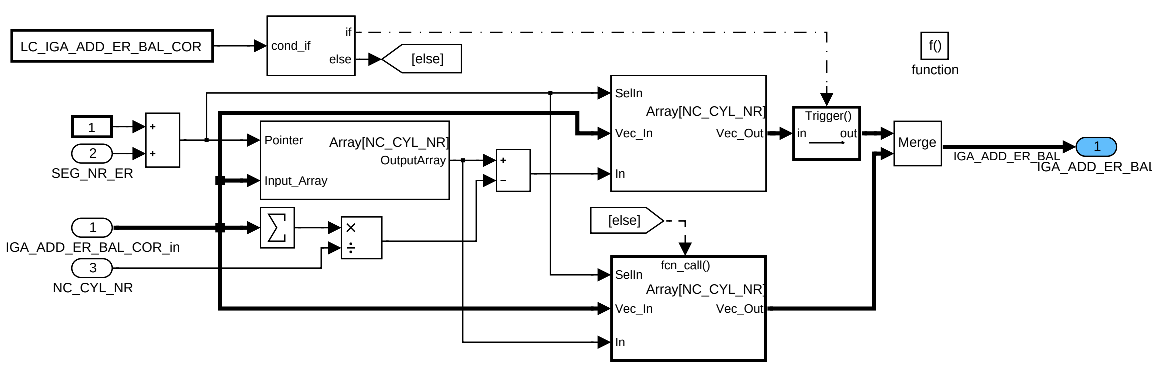


Figure 45 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ GENERAL

Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE INI_IGA_ADAPT

As soon as the ignition angle adaptation is enabled or disabled, the stored adaptive correction values are overwritten with the currently present global adaptive value and the controller output values are set to zero.

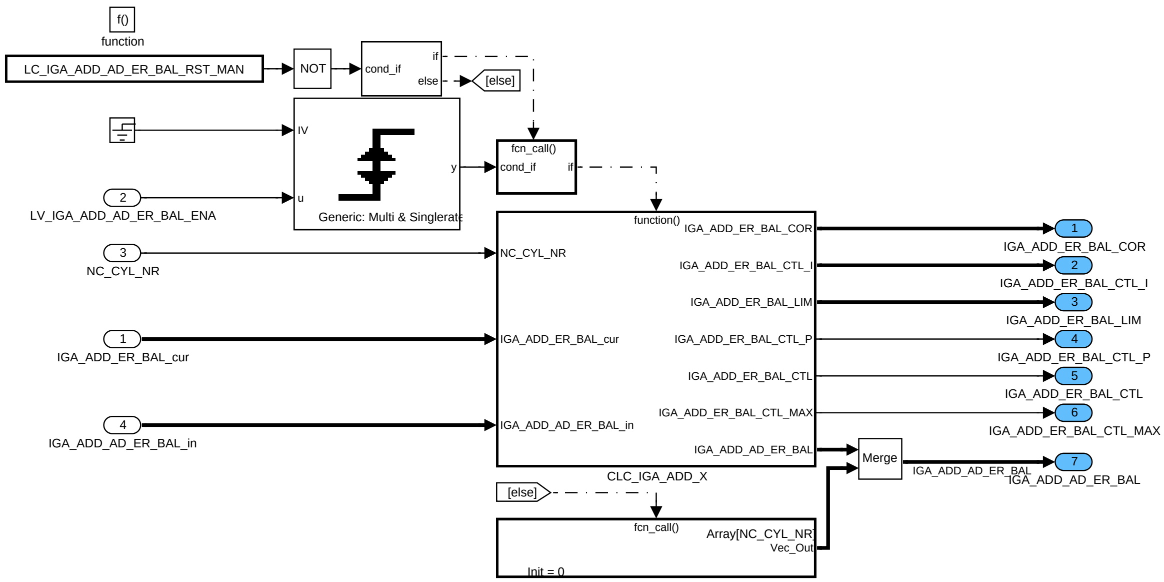


Figure 46 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ INI_IGA_ADAPT

Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE INI_IGA_ADAPT CLC_IGA_ADD_X

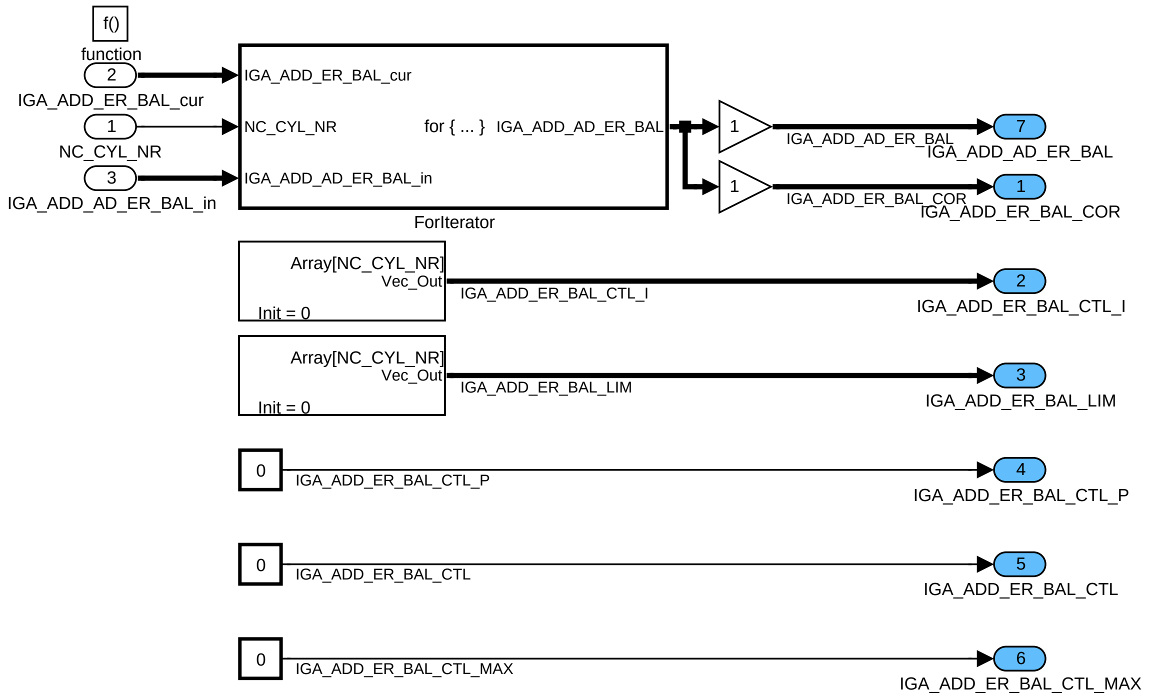


Figure 47 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ INI_IGA_ADAPT/ CLC_IGA_ADD_X

Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE INI_IGA_ADAPT CLC_IGA_ADD_X FORITERATOR

//in case of manual reset LC_IGA_ADD_AD_ER_BAL_RST_MAN = 1

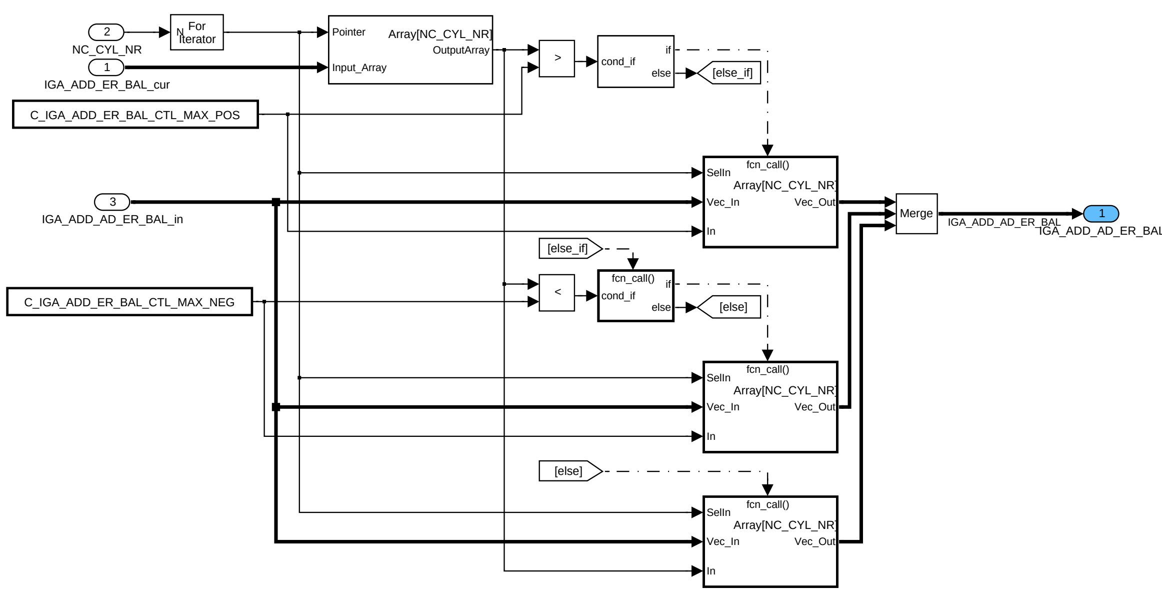


Figure 48 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ INI_IGA_ADAPT/ CLC_IGA_ADD_X/ ForIterator

Calculation of segment tasks CLC_OPERATE_SEG CLC_ADD_INJ_T_VALUE INI_IGA_COR

As soon as the ignition angle correction is disabled, the controller output values are set to zero. The initialization only takes place, if the logical constant for a manual reset of the controller values is set to zero (LC_IGA_ADD_ER_BAL_RST_MAN = 0)

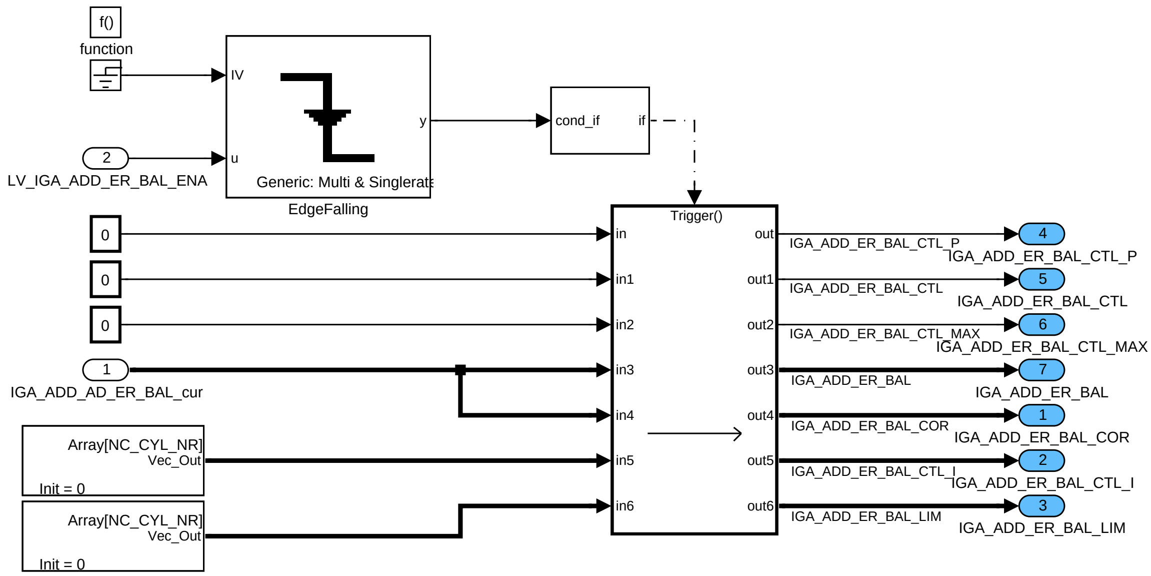


Figure 49 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ INI_IGA_COR

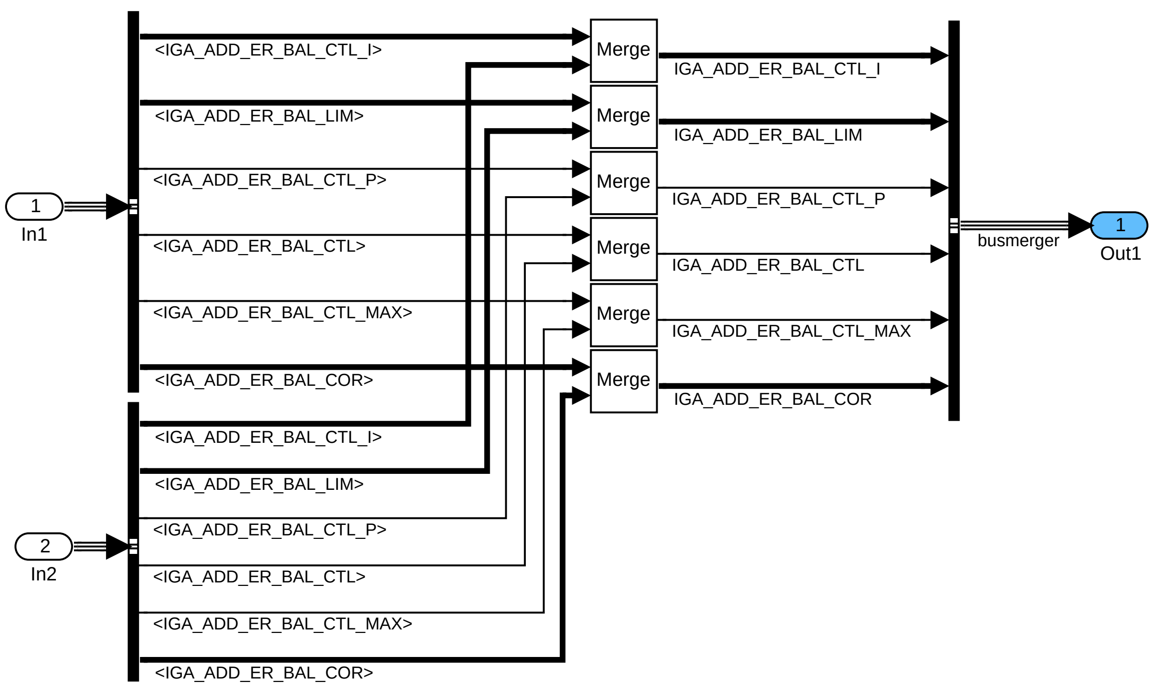


Figure 50 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ CLC_IGA_ADAPT_VALUE/ BusMerger

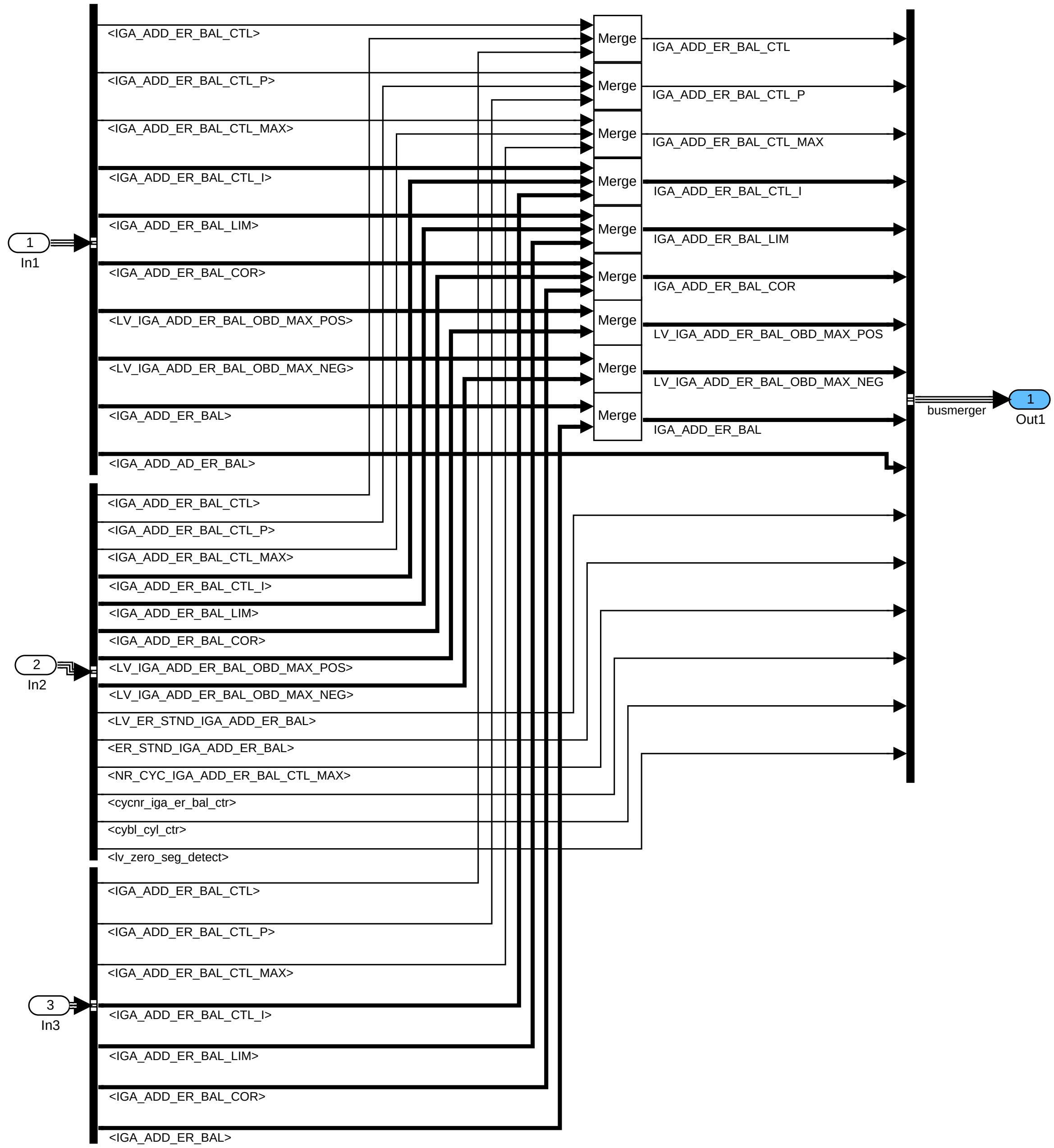


Figure 51 CYBL_M30602G02/ OPM/ operate_SEG/ CLC_OPERATE_SEG/ BusMerger

6.12.1.3 SUBFUNCTION: SIG_MNG

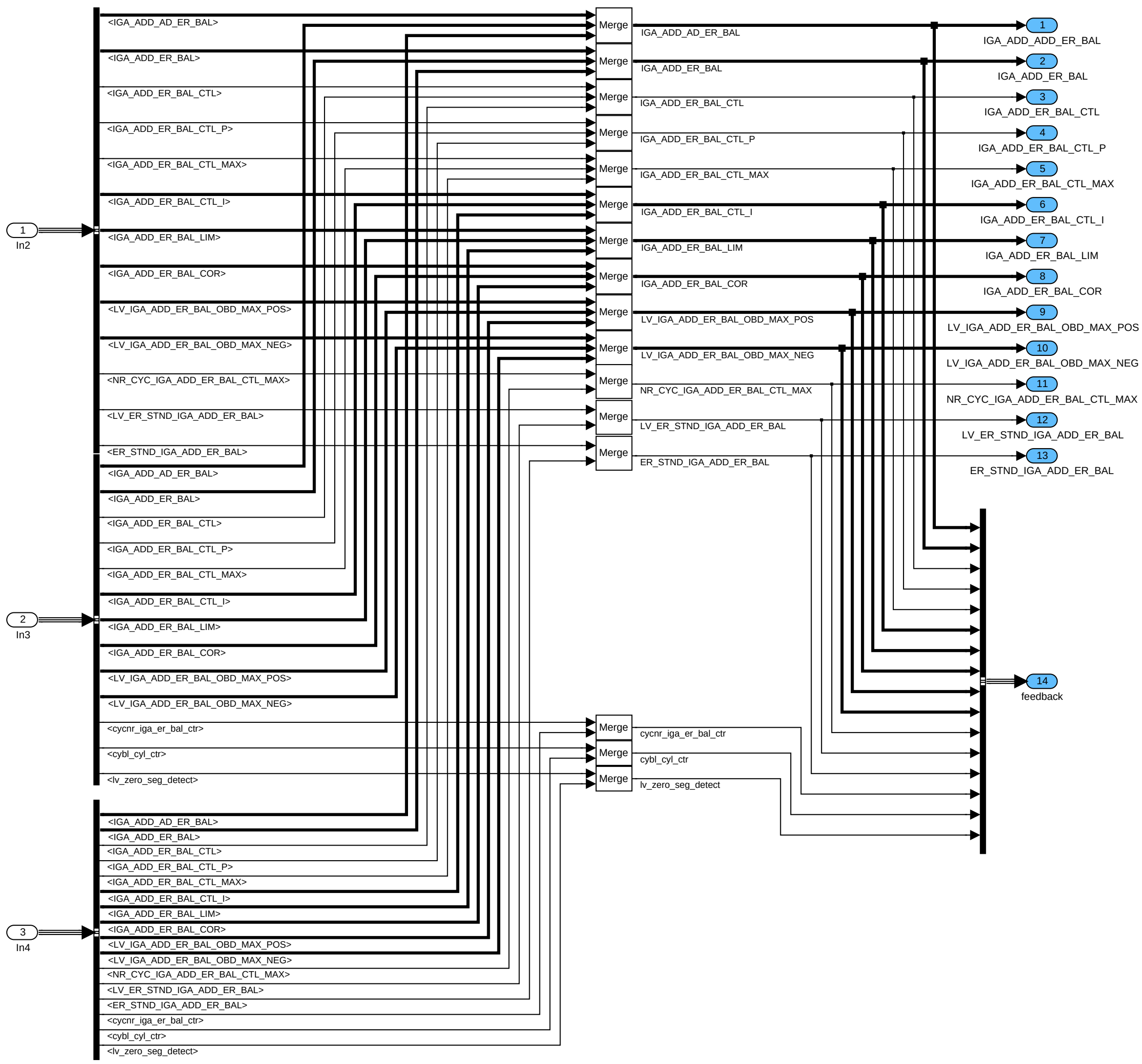


Figure 52 CYBL_M30602G02/ SIG_MNG

6.13 Instationary Knock Correction(IGA_TRA_KNK)

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
IGA_TRA_KNK	O/V	0 ... 80H	-48 ... 0	0.375	°CRK
Spark retard till high TPS_GRD or MAP_DRV1 applied to engnie					
IGA_TRA_KNK_INTER	V	0 ... 80H	-48 ... 0	0.375	°CRK
Spark retard till high TPS_GRD or MAP_DRV1 intermediate					
LV_IGA_TRA_KNK	O/V	0 ... 1H	0 ... 1	1	-
Flag for instationary correction avtive					
CTR_CYC_IGA_TRA_KNK_DLY	V	0 ... FFH	0 ... 255	1	-
Cycle counter for delay period before ignition intervention					
CTR_CYC_IGA_TRA_KNK_ACT	V	0 ... FFH	0 ... 255	1	-
Cycle counter during active ignition intervention					
CTR_CYC_IGA_TRA_KNK_WAIT	V	0 ... FFH	0 ... 255	1	-
Cycle counter for delay time before follow up ignition intervention					
STATE_IGA _TRA_KNK	V	0H 1H 2H 3H 4H 5H	INIT TRIGGER DELAY ACTIVE RAMP WAIT	1	-
State of ignition angle intervention for knock prevention					

Input data:

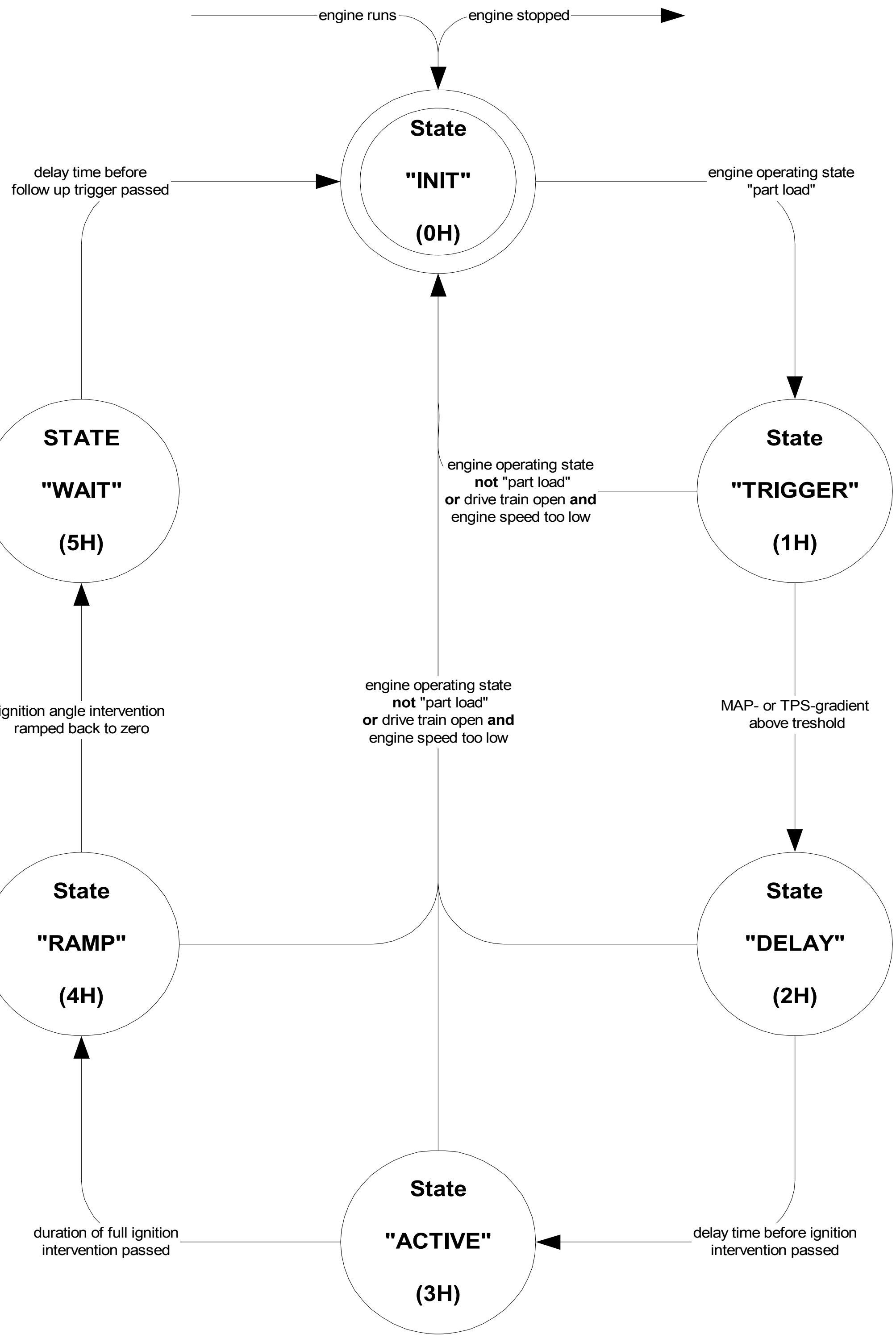
N	N_32	MAF	TPS_GRD
MAP_DRV1	TIA	VS	LV_DT
LV_PL	LV_AT	LV_TPS_GRD_UP	GEAR

Function description:

General Information:

In case the throttle position gradient or the intake manifold pressure gradient exceeds adjustable limits (“load dynamics”), an ignition angle intervention shall be performed in order to avoid knocking combustion.
After detection of load dynamics, the ignition intervention is not activated, before a calibratable number of segments has passed. During this delay the determined ignition adjustment (dependent on engine opereating point, vehicle speed and gear) is updated towards stronger interventions. At the end of the delay time - after the last update - the ignition intervention will be executed unchanged for a calibratable number of cycles.
After the end of this constant intervention phase, the ignition adjustment is reset to zero with calibratable incremet steps.
The ignition angle intervention of load dynamics is terminated whenever the drive train is detected open in combination with engine speed below a minimum threshold or in case engine operating state part load is left.
Since the trigger conditions (throttle- and pressure gradient) for ignition angle adjustment are linked by an "OR" operator, it might occur that both conditions are satisfied after a short time, thus causing a reiterated, undesired ignition angle intervention. To prevent this effect an adjustable waiting time before the next trigger of load dynamics can be applied.

Signal flow diagram:



In any state except RAMP and WAIT state, if ID_CYC_IGA_TRA_KNK_DLY = 0 and the triggering conditions are fulfilled, the spark timing retard(IGA_TRA_KNK) should be performed immediately without any delay.

Application conditions:

Activation:	LV_ES = 0
Deactivation:	LV_ES = 1
Recurrency:	every T.D.C.
Initialization:	STATE_IGA_TRA_KNK = “INIT”
	IGA_TRA_KNK = 0
	IGA_TRA_KNK_INTER = 0

Formula section:

State “*Init*ialization”: STATE_IGA_TRA_KNK = “INIT” (0H)

initialize all variables:

LV_IGA_TRA_KNK = 0

CTR_CYC_IGA_TRA_KNK_DLY = 0
CTR_CYC_IGA_TRA_KNK_ACT = 0
CTR_CYC_IGA_TRA_KNK_WAIT = 0

check for engine operating state part load:

if LV_PL = 1
and LV_TPS_GRD_UP = 1
then STATE_IGA_TRA_KNK = “TRIGGER”
no change of state:
else STATE_IGA_TRA_KNK = “INIT” (unchanged)
endif

State "Wait for *trigger*": STATE_IGA_TRA_KNK = "TRIGGER" (1H)

check termination conditions:

if LV_PL = 0
or **(** LV_DT = 0 **and** N < C_N_MIN_IGA_TRA_CS **)**
then STATE_IGA_TRA_KNK = "INIT"

check trigger conditions:

elseif **(**TPS_GRD >= ID_TPS_GRD_IGA_TRA_KNK(N_32)
and
LV_TPS_GRD_UP = 1**)**
or
(MAP_DRV1 >= ID_MAP_DRV1_IGA_TRA_KNK(N_32)
and
MAP_DRV1_{n} > MAP_DRV1_{n-1})
then STATE_IGA_TRA_KNK = "DELAY"

no change of state:

else STATE_IGA_TRA_KNK = "TRIGGER" (unchanged)
endif

State "*Delay* after load dynamics detection": STATE_IGA_TRA_KNK = "DELAY" (2H)

determine magnitude of ignition intervention for MT-vehicles:

if LV_AT = 0
then IGA_TRA_KNK_INTER = MIN **(** $IGA\_TRA\_KNK\_INTER_{n-1}$ **;**
IP_IGA_TRA_KNK_MT(N_32; MAF)
* IP_FAC_IGA_TRA(TIA)
* IP_FAC_IGA_TRA_GEAR_**MT**
(GEAR; VS) **)**

determine magnitude of ignition intervention for AT-vehicles:

else IGA_TRA_KNK_INTER = MIN **(** $IGA\_TRA\_KNK\_INTER_{n-1}$ **;**
IP_IGA_TRA_KNK_AT(N_32; MAF)
* IP_FAC_IGA_TRA(TIA)
* IP_FAC_IGA_TRA_GEAR_**AT**
(GEAR; VS) **)**

endif

increment cycle counter for delay before activation:

$CTR\_CYC\_IGA\_TRA\_KNK\_DLY_n = CTR\_CYC\_IGA\_TRA\_KNK\_DLY_{n-1} + 1$

check termination conditions:

if LV_PL = 0

or **(** LV_DT = 0 **and** N < C_N_MIN_IGA_TRA_CS **)**

then STATE_IGA_TRA_KNK = "INIT"

check duration of delay before activation:

elseif $CTR\_CYC\_IGA\_TRA\_KNK\_DLY_n$ >= ID_CYC_IGA_TRA_KNK_DLY(N_32)
(treshold only determined once at entry into state "DELAY")

then STATE_IGA_TRA_KNK = "ACTIVE"

no change of state:

else STATE_IGA_TRA_KNK = "DELAY " (unchanged)

endif

<u>State "Ignition intervention *active*":</u> STATE_IGA_TRA_KNK = "ACTIVE" (3H)

increment cycle counter for activation duration:

$CTR\_CYC\_IGA\_TRA\_KNK\_ACT_n = CTR\_CYC\_IGA\_TRA\_KNK\_ACT_{n-1} + 1$

copy magnitude of ignition intervention to execution variable:

IGA_TRA_KNK = IGA_TRA_KNK_INTER

LV_IGA_TRA_KNK = 1

check termination conditions:

if LV_PL = 0

or **(** LV_DT = 0 **and** N < C_N_MIN_IGA_TRA_CS **)then** STATE_IGA_TRA_KNK = "INIT"

check duration of activation:

elseif $CTR\_CYC\_IGA\_TRA\_KNK\_ACT_n$ >= ID_CYC_IGA_TRA_KNK_ACT(N_32)
(treshold only determined once at entry into state "ACTIVE")

then STATE_IGA_TRA_KNK = "RAMP"

no change of state:

else STATE_IGA_TRA_KNK = "ACTIVE" (unchanged)

endif

State "*Ramp* back ignition intervention": STATE_IGA_TRA_KNK = "RAMP" (4H)

ramp back ignition intervention:

$IGA\_TRA\_KNK_n = MIN\ (\ IGA\_TRA\_KNK_{n-1} + IP\_IGA\_TRA\_INC;\ \ 0\ )$

check termination conditions:

if LV_PL = 0
or **(** LV_DT = 0 **and** N < C_N_MIN_IGA_TRA_CS **)**
then STATE_IGA_TRA_KNK = "INIT"

check if ignition intervention ramped back completely:

elseif IGA_TRA_KNK = 0
then STATE_IGA_TRA_KNK = "WAIT"

no change of state:

else STATE_IGA_TRA_KNK = "RAMP" (unchanged)
endif

State "*Wait* after ignition intervention": STATE_IGA_TRA_KNK = "WAIT" (5H)

increment cycle counter for activation wait duration:

$CTR\_CYC\_IGA\_TRA\_KNK\_WAIT_n = CTR\_CYC\_IGA\_TRA\_KNK\_WAIT_{n-1} +1$

LV_IGA_TRA_KNK = 0

check duration of waitng time before release of follow up trigger:

if $CTR\_CYC\_IGA\_TRA\_KNK\_WAIT_n$ >= ID_CYC_IGA_TRA_KNK_WAIT(N_32)
(treshold only determined once at entry into state "WAIT")
then STATE_IGA_TRA_KNK = "INIT"

no change of state:

else STATE_IGA_TRA_KNK = "WAIT" (unchanged)
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
ID_TPS_GRD_IGA_TRA_KNK	1x4	0...FFH	0...2988	11.718	°TPS/s
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
TPS_GRD-threshold for activation of the IGA retard adjustment					
ID_MAP_DRV1_IGA_TRA_KNK	1x4	8000H...7FFFH	-82.9...82.9	0.00253	hPa/ms
LDPM_N_32_7	4	0...FFH	8160	32	rpm
MAP_DRV1-threshold for activation of the IGA retard adjustment					
IP_IGA_TRA_KNK_MT	4x4	0...80H	-48...0	0.375	°CRK
LDPM_MAF_IGA__IGA_TRA_KNK	4	0...FFH	0...1389	5.447	mg/stk
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
Ignition adjustment angle in retard direction					
IP_IGA_TRA_KNK_AT	4x4	0...80H	-48...0	0.375	°CRK
LDPM_MAF_IGA__IGA_TRA_KNK	4	0...FFH	0...1389	5.447	mg/stk
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
Ignition adjustment angle in retard direction					
IP_FAC_IGA_TRA	4	0...FFH	0...0.996	0.0039	-
LDP_TIA_FAC_IGA_TRA	4	0...FEH	-48...142.5	0.75	°C
TIA correction factor					
IP_FAC_IGA_TRA_GEAR_MT	6x6	0...FFH	0...0.996	0.0039	-
LDP_VS_FAC_IGA_TRA_GEAR_MT	6	0...FFH	0...255	1	km/h
LDPM_GEAR_1	6	0...7H	0...7	1	-
GEAR/VS correction factor					
IP_FAC_IGA_TRA_GEAR_AT	6x6	0...FFH	0...0.996	0.0039	-
LDP_VS_FAC_IGA_TRA_GEAR_AT	6	0...FFH	0...255	1	km/h
LDPM_GEAR_1	6	0...7H	0...7	1	-
GEAR/VS correction factor					
ID_CYC_IGA_TRA_KNK_DLY	1x4	0...FFH	0...255	1	-
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
Delay after the load dynamics has been triggered initially					
ID_CYC_IGA_TRA_KNK_ACT	1x4	0...FFH	0...255	1	-
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
Number of cycles for ignition angle adjustment active					
ID_CYC_IGA_TRA_KNK_WAIT	1x4	0...FFH	0...255	1	-
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
Delay between two ignition angle interventions due to load dynamics					
IP_IGA_TRA_INC	1x4	1...7FFFH	0.00146...47.99	0.00146	°CRK / wcyc
LDPM_N_32_7	4	0...FFH	0...8160	32	rpm
IGA limitation gradient for controlled reset of the ignition angle adjustment IP_IGA_TRA_KNK__N__MAF					
C_N_MIN_IGA_TRA_CS	1	0...1FE0H	0...8160	1	rpm
Engine speed threshold for load dynamics IGA					

6.13.1 Ignition angle correction in altitude at full-load (IGA_BAS_AMP_COR)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_BAS_AMP_COR	V/O	80...7FH	-48...47.625	0.375	°CRK
Ignition angle offset relative to ambient pressure.					

Input data:

LV_FL	N_32	AMP	

FUNCTION DESCRIPTION:

General information:

An ignition angle offset is applied in high altitude in case of full-load detection (LV_FL = 1). This offset depends on engine speed N and ambient pressure AMP.

Application conditions:

Activation: LV_HOM_RUN=1

Deactivation: LV_HOM_RUN=0

Initialisation: IGA_BAS_AMP_COR = 0

Update Rate: 10 ms.

Formula section:

if LV_FL = 1

then IGA_BAS_AMP_COR = IP_IGA_BAS_AMP_COR(N_32,AMP)

else IGA_BAS_AMP_COR = 0

endif

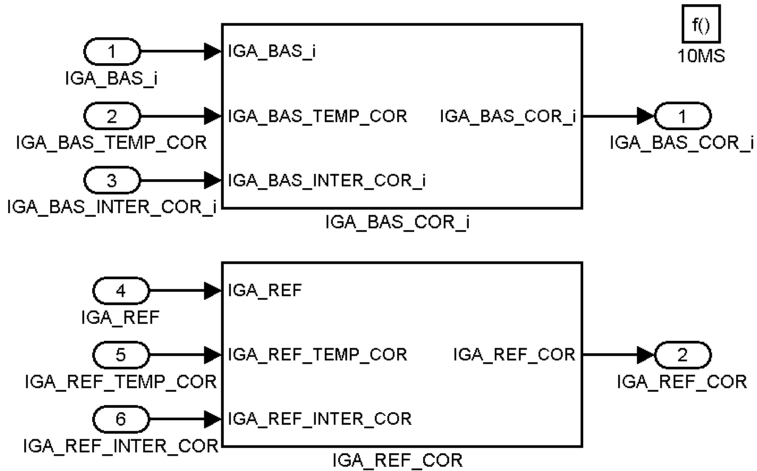
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_BAS_AMP_COR	8 x 4	80...7FH	-48...47.625	0.375	°CRK
LDP_N_32_IP_IGA_BAS_AMP_COR	8	0...FFH	0...8.16E3	32	rpm
LDP_AMP_IP_IGA_BAS_AMP_COR	4	0...FFFFH	0...5434	0.083	hPa
Ignition angle correction relative to ambient pressure.					

6.14 General correction on ignition angle

Overview:

This file enables each projects to make all the necessary corrections on ignition angle due to specific engine configuration and according to the external functions used by the project.



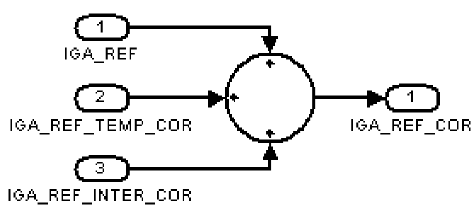
Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_REF_COR	O/V	0..FFh	-35.625..60	0.375	°CRK
Corrected reference ignition angle					
IGA_BAS_COR[NC_CBK_IN_NR]	O/V	0..FFH	-35.625..60	0.375	°CRK
Corrected basic ignition angle – for cylinder bank i					

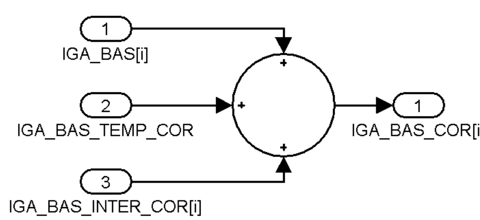
Input data:

LV_HOM_RUN	IGA_REF	IGA_BAS[NC_CBK_IN_NR]	IGA_BAS_TEMP_COR
IGA_REF_TEMP_COR	NC_CBK_IN_NR	IGA_REF_INTER_COR	IGA_BAS_INTER_COR[NC_CBK_IN_NR]

6.14.1 Total formula for IGA_REF_COR



6.14.2 Total formula for IGA_BAS_COR[i]



6.15 Minimum Ignition Angle

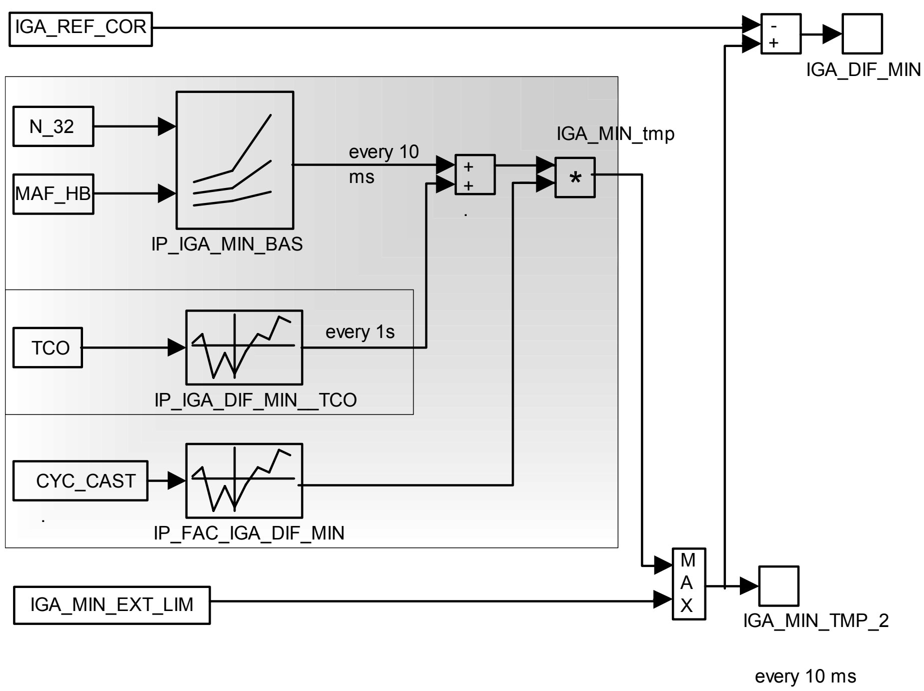
Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_DIF_MIN	O/V	0..FFH	0..-95.625	0.375	°CRK
Difference from minimum ignition angle to reference ignition angle					
IGA_MIN_TMP_2	V	0..FFH	-23.625..72	0.375	°CRK
minimum ignition angle internal value					

Input data:

LV_HOM_RUN	N_32	MAF_HB	TCO
IGA_REF_COR	CYC_CAST	IGA_MIN_EXT_LIM	

FUNCTION DESCRIPTION:



General information:

The minimum ignition angle IGA_MIN is the minimum allowable ignition angle which is possible to apply to the engine in hom. mode. The intension of this minimum ignition angle IGA_MIN is the absolut latest ignition angle, where the engine has complete combustion. Other aspects like exhaused gas temperature are taken into account somewhere else.
The IGA_MIN calculated from a basic value of the minimum ignition angle corrected with a temperature dependent additive term and a factor depending on engine cycles after start.

Application conditions:

Activation: LV_HOM_RUN = 1

Deactivation: LV_HOM_RUN = 0

Initialisation: IGA_MIN_TMP_2 = -23.625 °CRK at reset
IGA_DIF_MIN = 0

Update Rate: IP_IGA_DIF_MIN_BAS => 10 ms
IP_IGA_DIF_MIN_TCO => 1000 ms

Formula section:

Maximum possible spark retard:

In all practical cases IGA_DIF_MIN is a negative value which is the maximum allowable spark retard. It includes the coolant temperature correction and a cycle factor (for engine protection. pollution condition. etc…).

IGA_MIN_tmp = **(**IP_IGA_MIN_BAS (__N_32__MAF_HB) + IP_IGA_DIF_MIN_TCO**)**
***** IP_FAC_IGA_DIF_MIN_CYC_CAST

Minimum ignition angle (absolute value):

IGA_MIN_TMP_2 **= MAX (** IGA_MIN_tmp **;** IGA_MIN_EXT_LIM **)**

IGA_DIF_MIN **=** IGA_MIN_TMP_2 **–** IGA_REF_COR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_MIN_BAS	16*12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_32__IGA	16	0...FFH	0...8.16E3	32	rpm
LDP_MAF_HB__IGA_MIN_BAS	12	0...FFH	0...1.389E3	5.447	mg/tdc
Basic absolute minimum ignition angle difference value					
IP_IGA_DIF_MIN_TCO	8	0...FFH	-48...47.625	0.375	°CRK
LDPM_TCO__IGA_TEMP_FAC	8	0...FEH	-48...142.5	0.75	°C
TCO correction for minimum ignition angle					
IP_FAC_IGA_DIF_MIN_CYC_CAST	8	0...FFH	0..1.992	0.0078	-
LDP_CYC_CAST__FAC_IGA_DIF_MIN	8	0...FFFFH	0..65535	1	-
Combustion cycles after start correction for minimum ignition angle					

6.15.1 External Limitation on Minimum Ignition (version for absolute IGA_MIN appl.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DRI	V	0..FFFFH	655,35	0.01	Sec.
Time after Drive engaged to apply low Spark advance during Gear Shift					
IGA_MIN_EXT_LIM	V/O	0..FFH	-23.625..72	0.375	°CRK
minimum ignition angle during gear change of AMT					

Input data:

LV_HOM_RUN	LV_DRI	LV_GS_ENA	IGA[NC_CYL_NR]
SPK_RTD_TCU	TCO		

FUNCTION DESCRIPTION:

Depending on driving condition, Spark Advance is limited during Torque Reduction request for Gear Shift.

Application conditions:

Activation: LV_HOM_RUN = 1

Deactivation: LV_HOM_RUN = 0

Initialisation: IGA_MIN_EXT_LIM = -23.625 °CRK (0 hex)

T_DRI = 0

Recurrence: 10 ms

Formula section:

- *T_DRI definition* **:**

If LV_DRI = 0
Then T_DRI = 0
Else
  If Transition LV_DRI = 0 -> 1
  Then T_DRI is set to C_T_DRI
  Else T_DRI is decremented by 0,01 sec. Until it reaches 0

- *IGA_MIN_EXT_LIM definition* **:**

If LV_GS_ENA = 1
Then
  If SPK_RTD_TCU < FFH
  Then IGA_MIN_EXT_LIM = Min(SPK_RTD_TCU, IGA)

  Else **If** T_DRI > 0
    Then IGA_MIN_EXT_LIM = IP_IGA_GS_1
    Else IGA_MIN_EXT_LIM = C_IGA_GS_2

Else IGA_MIN_EXT_LIM = -23,625°CRK (0 hex)

- Remark:

When IGA_MIN_EXT_LIM is returned to -23.625°CRK (0 hex), there are 10ms delay and when IGA_MIN_EXT_LIM is returned to IP_IGA_GS_1 or C_IGA_GS_2, there are 30ms delay to avoid a sudden drop of TQI_AV by released IGA_AV limitation.

IGA is averaged value of IGA[NC_CYL_NR].

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
IP_IGA_GS_1	8	0...FFH	-23.625..72	0.375	°CRK
LDP_TCO__IGA_GS_1	8	0...FEH	-48..142.5	0.75	°C
Minimum spark advance during GS (short after transition LV_DRI = 0 -> 1)					
C_IGA_GS_2	1	0...FFH	-23.625..72	0.375	°CRK
Minimum Spark advance during Gear Shift					
C_T_DRI	1	0...FFFFH	655,35	0.01	Sec.
Time after Transition LV_DRI = 0 -> 1 to apply low Spark Advance during Gear Shift					

6.16 IGA_MIN limitation for exhaust gas temperature protection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_DIF_MIN_TEG	O/V	0...FFH	0...-95.625	-0.375	°CRK
Difference of minimum ignition angle with reference one - TEG correction included					
IGA_MIN_TEG	O/V	0...FFH	-35.625...60	0.375	°CRK
minimum ignition angle					
IGA_DIF_MIN_TEG_BAS	O/V	0...FFH	0...-95.625	-0.375	°CRK
Difference of minimum ignition angle with reference 1 - TEG correction included for IGA_MIN calculation					
TEG_HYS_IGA_DIF_MIN	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Temperature of exhaust gas with hysterisis for minimum IGA limitation					

Input data:

IGA_DIF_MIN	IGA_REF_COR	LV_ES	LV_HOM_RUN
LV_ST	TEG_DYN_UP_CAT[NC_CBK_EX_NR]	NC_CBK_EX_NR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEG_MAX_IGA	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Maximum allowable exhaust gas temperature for spark retard control					
C_TEG_MAX_IGA_HYS	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Hysteresis on exhaust gas temperature for spark retard control					
IP_IGA_DIF_MIN_TEG	16	0...FFH	0...-95.625	0.375	°CRK
LDP_TEG_IP_IGA_DIF_MIN_TEG	16	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Minimum ignition angle versus exhaust gas temperature					

6.16.1 General information

The minimum ignition angle IGA_DIF_MIN is the minimum allowable retard on ignition angle that is possible to apply to the engine. Hence, it only includes the combustion stability concern. It does not take into account any thermal effect. It should be used during the torque intervention phases in order to produce a fast reduction of the torque.

Applying this minimum ignition angle as actual ignition angle results in efficiency losses that leads to an exhaust gas overheating and/or engine damages.

In order to avoid overheating of the components in the exhaust-system branch, minimum spark retard has to be limited in accordance with the actual exhaust gas temperature TEG_DYN_UP_CAT[NC_CYL_CBK_NR].

Moreover to determine the maximum possible torque reserve for catalyst heating, the torque reserve co-ordination module needs the minimum ignition angle efficiency and the minimum ignition angle with exhaust branch protection efficiency in order to limit and choose the different torque reserve requests.

Hence another minimum spark retard difference have to be calculated IGA_DIF_MIN_TEG which takes into account the exhaust gas system protection. It is the maximum between the one calculated versus the maximal exhaust gases temperature of TEG_DYN_UP_CAT[NC_CYL_CBK_NR] and the basic one IGA_DIF_MIN.

Classically IGA_DIF_MIN is calculated from a basic value of the minimum ignition angle corrected with a temperature dependent additive term. The basic minimum spark advance is here extracted from a calibration map.

To calculate the minimum spark retard limitation according to the exhaust gas temperature the map IP_IGA_DIF_MIN_TEG is used. TEG_HYS_IGA_DIF_MIN is used as input for this map. This intermediate variable TEG_HYS_IGA_DIF_MINrepresents the exhaust gas temperature TEG_DYN_UP_CAT[NC_CYL_CBK_NR] increased by an hysterisis in order to anticipate the exhaust-system branch overheat.

So, in all practical cases IGA_DIF_MIN is a negative value that is the maximum allowable spark retard for engine stable running and pollutant conditions. It includes the coolant temperature correction. The exhaust gas temperature limitation is then taken into account in IGA_DIF_MIN_TEG.

The influence of the ignition angle on the exhaust gas temperature results from the ignition angle efficiency. Thus - in order to calibrate easily the map IP_IGA_DIF_MIN_TEG - one should use the efficiencies resulting from IGA_DIF_MIN values with the map IP_EFF_IGA and limit these efficiencies according to the actual exhaust gas temperature to avoid exhaust-system branch overheat. Then, thanks to the reverse map IP_IGA_DIF_EFF or thanks to the torque model for ignition angle efficiency, it is possible to calculate the value of the map IP_IGA_DIF_MIN_TEG.

The actual exhaust gas temperature TEG_DYN_UP_CAT[NC_CYL_CBK_NR] can be either measured by a sensor on the exhaust-system branch or calculated from an exhaust gas temperature model.

For engines with multiple cylinder banks, this temperature TEG_DYN_UP_CAT of exhaust gas before catalyst is defined and dedicated for each cylinder banks individually. Since here the limitation of ignition retard is just a protective function versus the catalyst overheating, only the maximal value from all those cylinder bank related TEG_DYN_UP_CAT temperature will be taken into account.

Note:

Two different update rates are used in the following calculations: for everything related to exhaust temperature only 1000ms rate is used to update calculation in synchronisation with the exhaust temperature updates (the one of TEG_DYN_UP_CAT[NC_CYL_CBK_NR]); for calculations related to ignition angle a 10 time faster update rate of 100ms.

These calculations are only performed when engine is running (not stalled) in homogenous mode after the end of start phase.

Application Condition

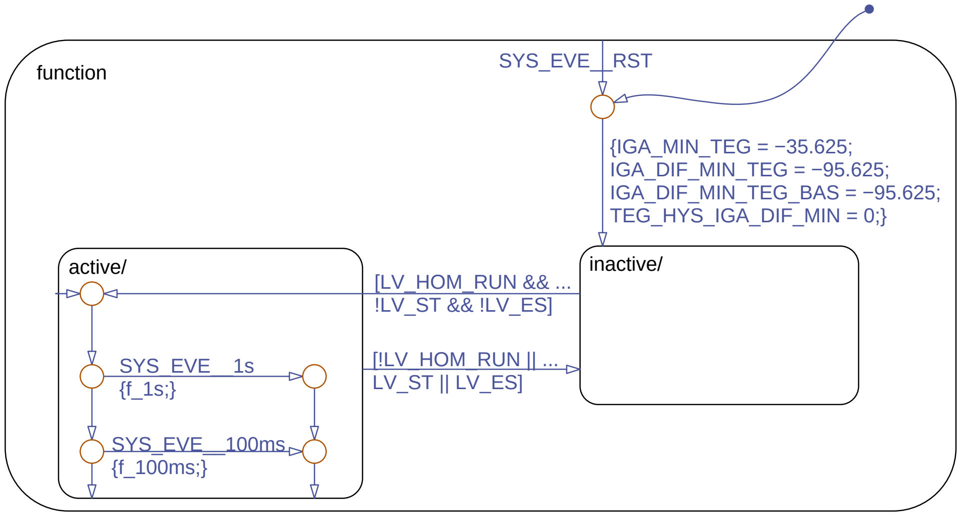


Figure 53 EXTC_REQGNiga0/ APP_CDN/ Chart

Function Description

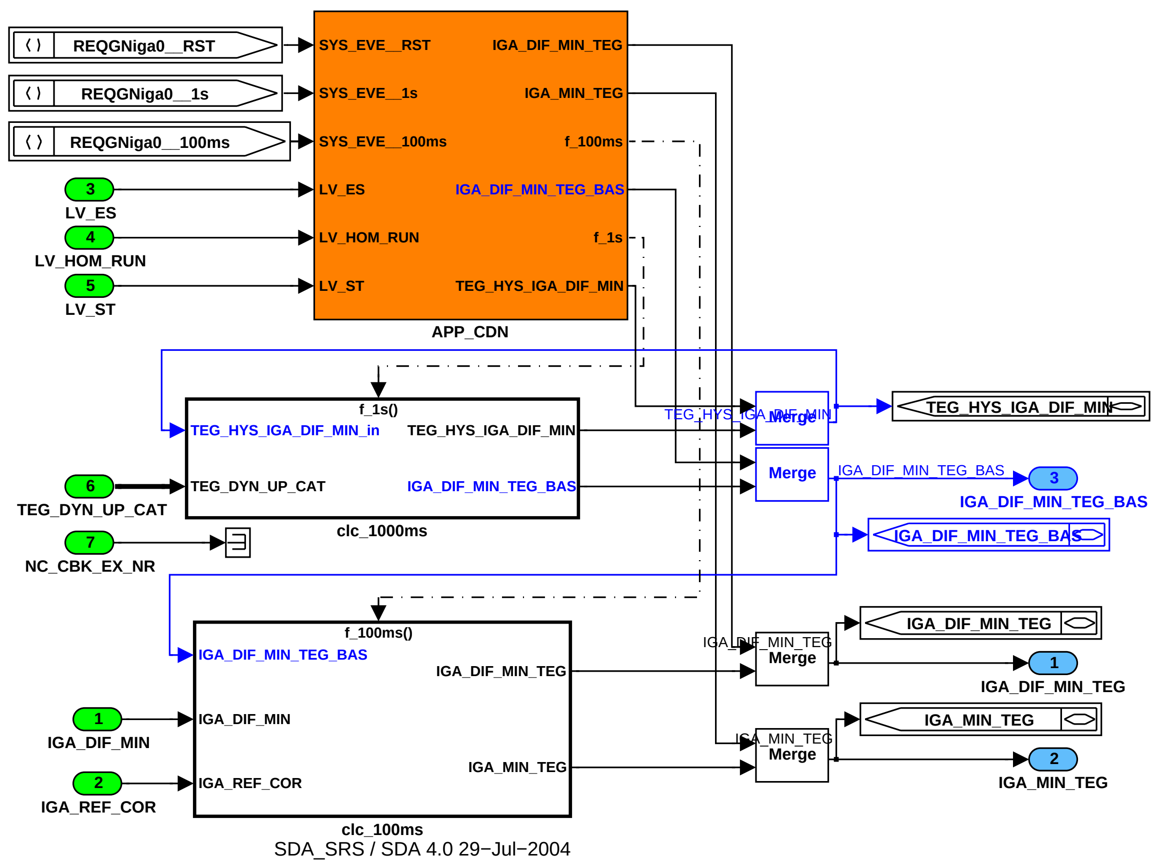


Figure 54 EXTC_REQGNiga0

6.16.1.1 SUBFUNCTION: clc_1000ms

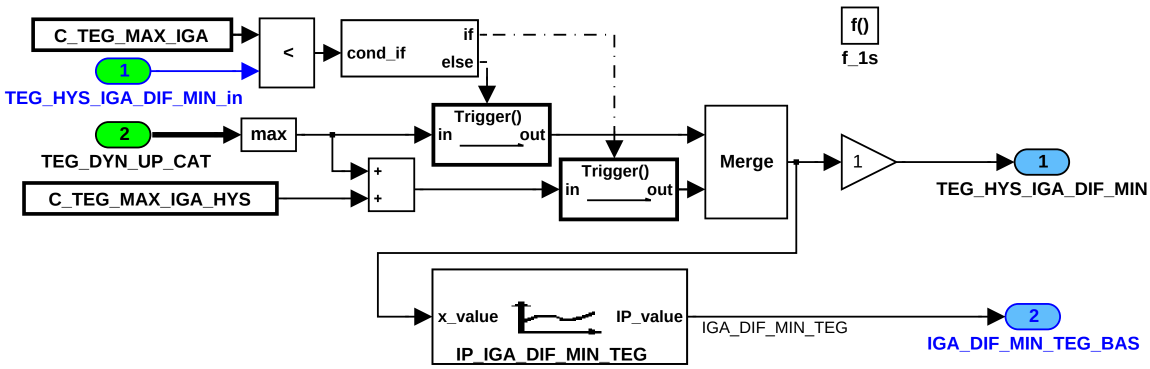


Figure 55 EXTC_REQGNiga0/ clc_1000ms

6.16.1.2 SUBFUNCTION: clc_100ms

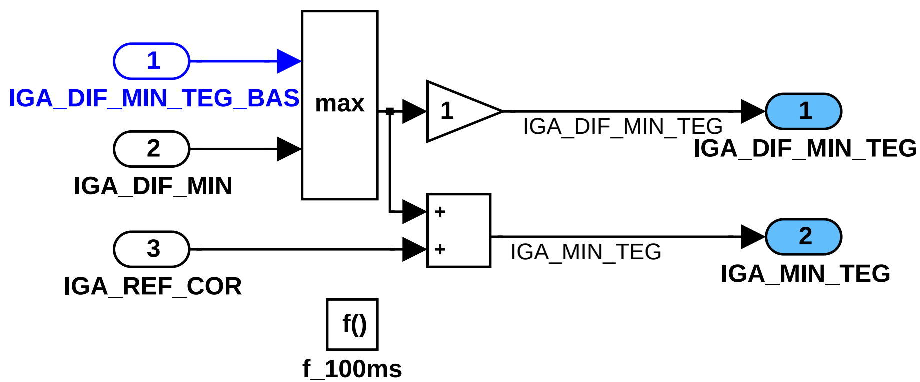


Figure 56 EXTC_REQGNiga0/ clc_100ms

6.17 (TQM) Setpoint Ignition Angle

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_IGA_SP	O/V	0...FFFFH	0...1.999969	3.05176E-5	-
Setpoint ignition efficiency					
IGA_DIF_TQ_REQ	O/V	FF01...FFH	95.625...-95.625	0.375	°CRK
Setpoint subtrahend ignition angle					
LV_TQ_IGA_ACT	O/V	0...1H	0...1	1	-
Logical variable: Torque intervention by spark retard is active					
EFF_IGA_SP_1	V	0...FFFFH	0...1.999969	3.05176E-5	-
Setpoint ignition efficiency					

Input data:

TQI_REF	EFF_EGR_HOM	EFF_LAMB_AV	EFF_SCC_AV
LV_HOM_RUN	LV_TQ_IGA_ENA	TQI_SP	LV_PUC
LV_IGK			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_EFF_MOD_SCC	1	0...1H	0...1	1	-
Mode switch for utilization of EFF_SCC_AV in transition out of PUC					
LC_TQ_IGA_INH	1	0...1H	0...1	1	-
Logical constant inhibition torque intervention by spark retard					
IP_IGA_DIF_TQ_REQ	16	0...FFH	0...-95.625	0.375	°CRK
LDP_EFF_IGA_SP_IP_IGA_DIF_TQ	16	0...FFFFH	0...1.9999	3.05165E-5	-
Setpoint spark retard for torque setpoint					

6.17.1 General Informationn

In homogeneous combustion mode (LV_HOM_RUN = 1), this module delivers the output IGA_DIF_TQ_REQ. The output IGA_DIF_TQ_REQ represents the ignition retard with reference to IGA_REF_COR that is necessary to generate the indicated engine torque TQI_SP. IGA_DIF_TQ_REQ is a function of the ignition angle efficiency setpoint EFF_IGA_SP, which, in turn, is calculated out of the Reference Indicated Torque TQI_REF and the actual efficiencies EFF_SCC_AV, EFF_EGR_HOM, and EFF_LAMB_AV.

During certain driving or operating modes (e.g. ‘Economy Mode’), the ignition intervention will be disabled by the Dynamic Torque Manager (i.e. LV_TQ_IGA_ACT = 0). In that case, IGA_DIF_TQ_REQ will be set to a passive value of 95.625 °Crk, resulting in a final ignition angle setpoint equal to IGA_BAS_COR.

In order to smooth the ignition angle setpoint behavior during transitions out of PUC, the efficiency EFF_SCC_AV might be replaced by a passive value of 1. This option can be chosen by the calibration switch LC_EFF_MOD_SCC (=0).

In order to ensure that the engine fires at IGA_MIN when the ignition is turned off (to avoid engine speed overshoots), the EFF_SCC_AV is set to 1 in this case.

Application Condition

Function Description

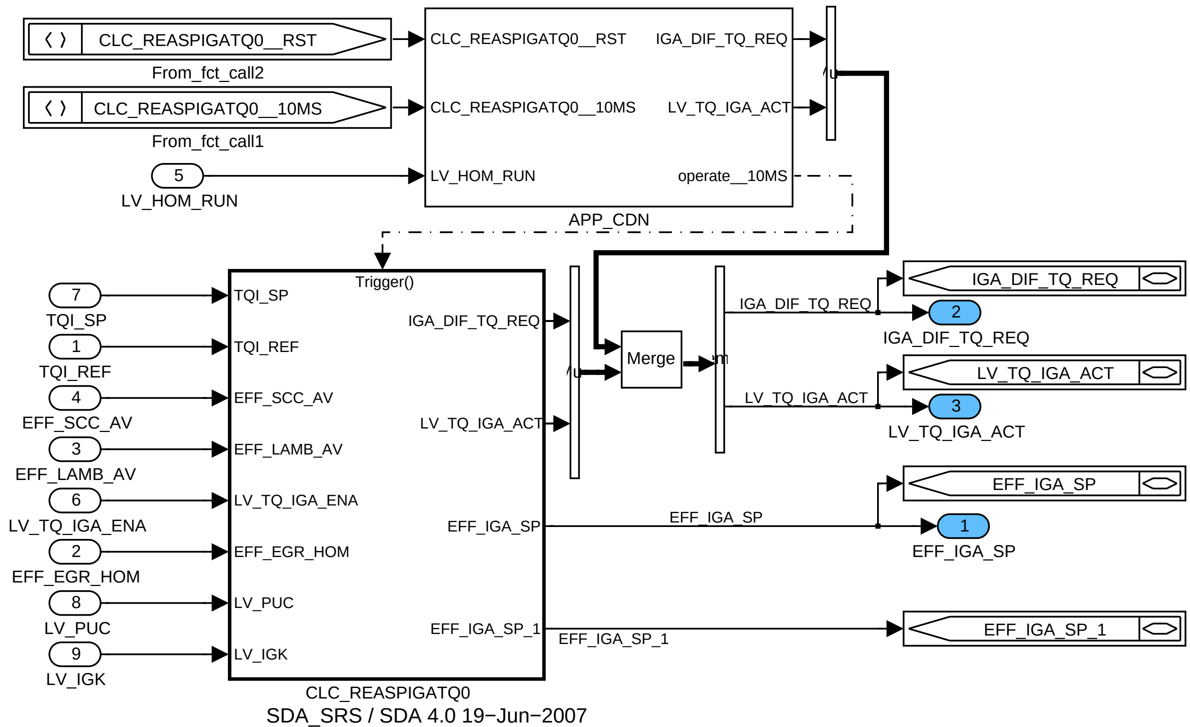


Figure 57 M601C

6.17.1.1 SUBFUNCTION: CLC_REASPIGATQ0

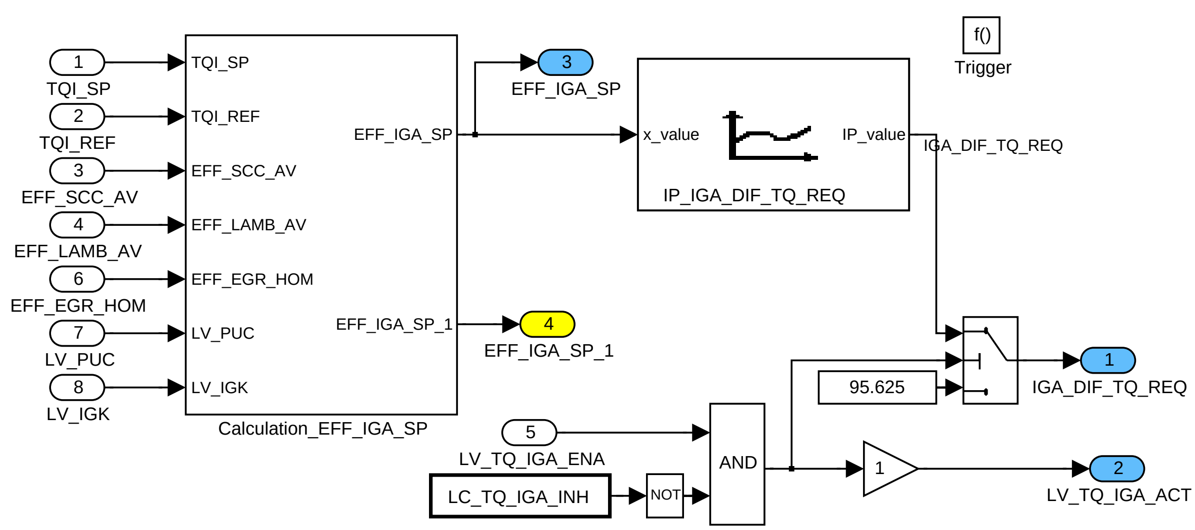


Figure 58 M601C/ CLC_REASPIGATQ0

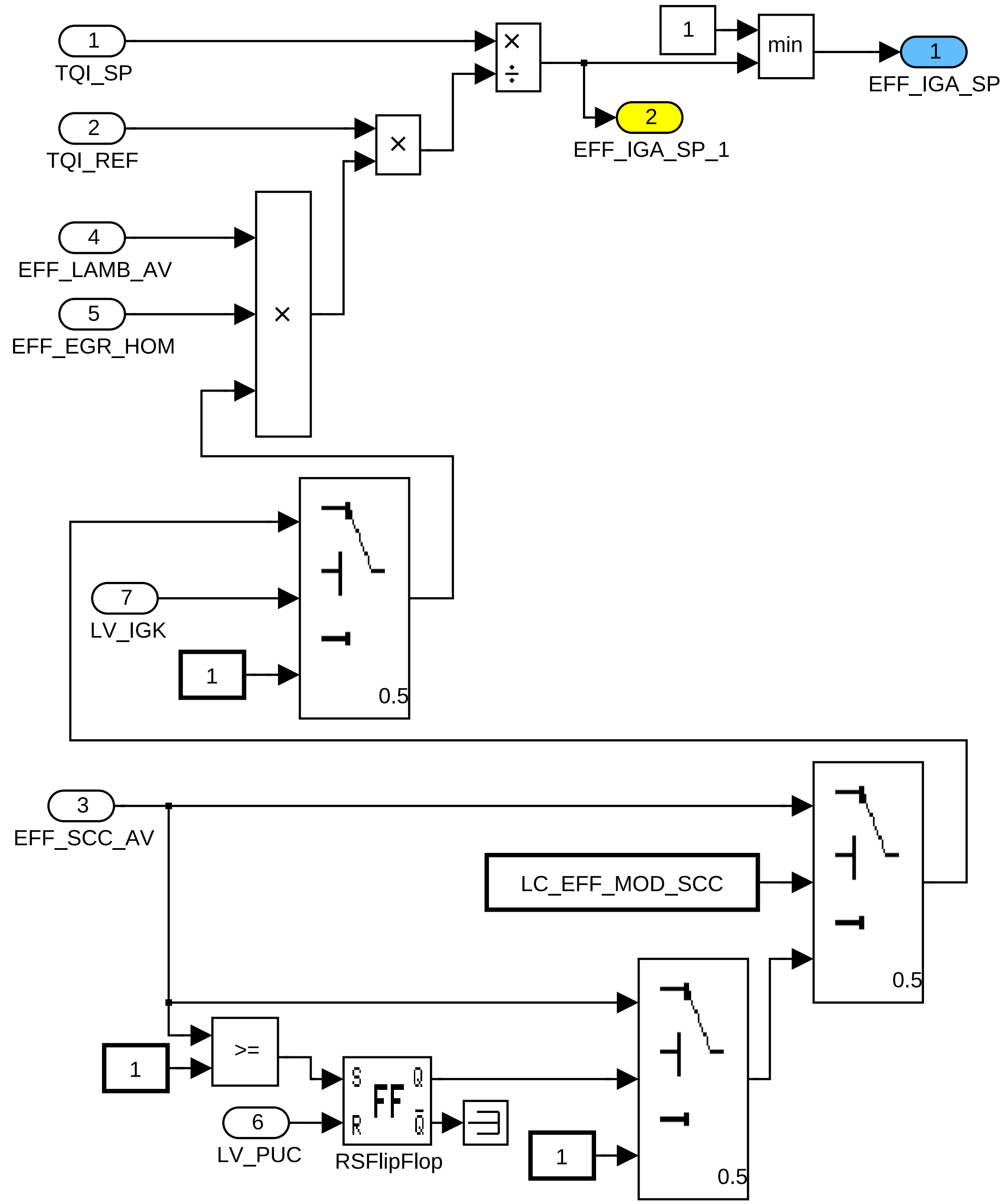


Figure 59 M601C/ CLC_REASPIGATQ0/ Calculation_EFF_IGA_SP

6.18 IGA_BAS[i] calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_BAS[NC_CBK_IN_NR]	O/V	0...FFH	-35.625...60	0.375	°CRK
Basic ignition angle from engine running point and lambda - for cylinder bank i					

Input data:

LV_HOM_RUN	N_32	MAF_HB	LAMB_SP_MV
NC_CBK_IN_NR			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_BAS[NC_CBK_IN_NR]	16x12	0...FFH	-35.625...60	0.375	°CRK
LDPM_N_32_1_IGSP	16	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_HB_1_IGSP	12	0...FFH	0...1.389E+3	5.44705882	mg/stk
Basic ignition angle					
IP_IGA_BAS_LAMB	8	0...FFH	-48...47.625	0.375	°CRK
LDPM_LAMB_SP_MV_1_IGSP	8	0...7FFFH	0...31.99	9.76287E-4	
Additive lambda setpoint correction on IGA_BAS - cylinder bank i					
IP_IGA_BAS_LAMB_FAC	8x6	0...FFH	0...3.984375	0.015625	-
LDPM_N_32_2_IGSP	8	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_HB_2_IGSP	6	0...FFH	0...1.389E+3	5.44705882	mg/stk
Weighting factor for LAMBDA correction versus N_32, MAF_HB					

6.18.1 General information:

For homogenous application, the basic ignition angle is calculated from a basis spark advance map depending on engine speed and load and includes the correction due to the actual value of richness.

To manage engine with an EMS defined for an ignition angle related to multiple cylinder banks (*NC_CBK_IN_NR*) this module can be used as many times as cylinder banks there are in the engine. Each time a different Basic Ignition Angle IGA_BAS[i] – *with i designating the corresponding cylinder banks considered* – is calculated to define and to allow a specific behaviour for each different cylinder bank.

In this aim different (*ie. specifically defined and related for each cylinder bank*) calibration maps IP_IGA_BAS[i] and IP_IGA_BAS_LAMB (*for correction versus lambda value*) can be used to produce different and cylinder bank specifically related Basic Ignition Angles.

In order to avoid determining a basic ignition angle IGA_BAS[i] greater than the reference one, care as to be taken during the calibration: One has to have a look on the calibration values written comparing each time the basic and reference ignition to ensure that basic ignition remains smaller than the reference one calibrated.

Function Description

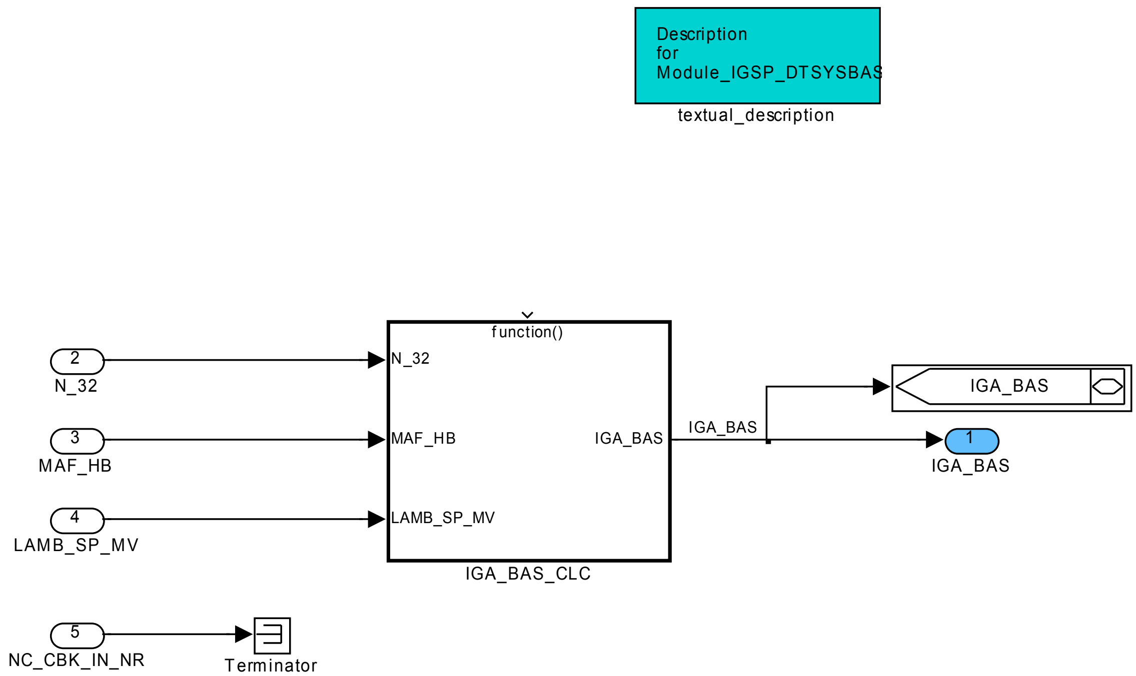


Figure 60 IGSP_DTSYSBAS0

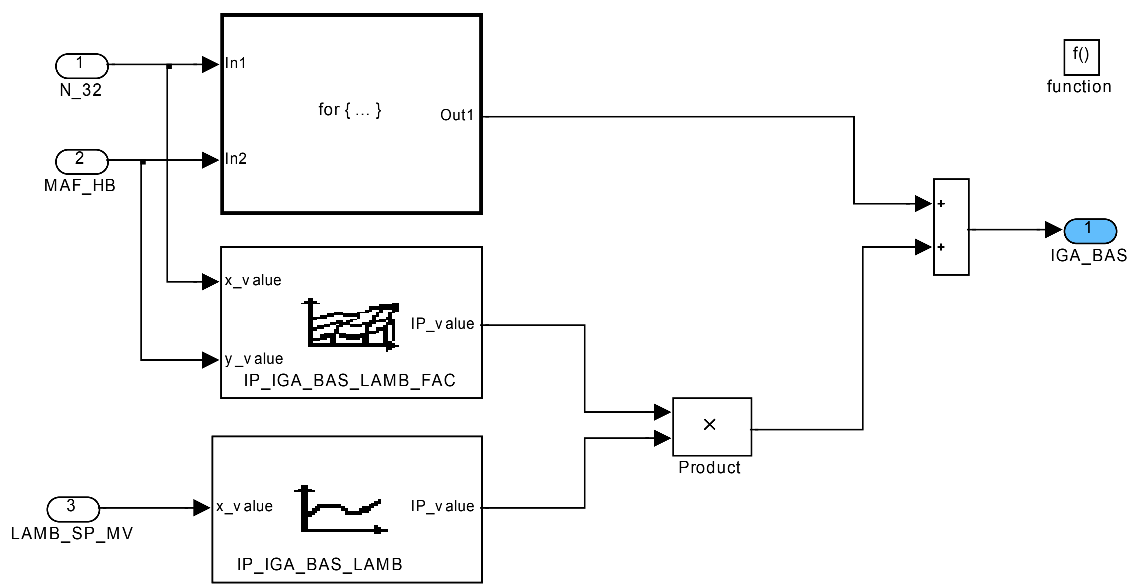


Figure 61 IGSP_DTSYSBAS0/ IGA_BAS_CLC

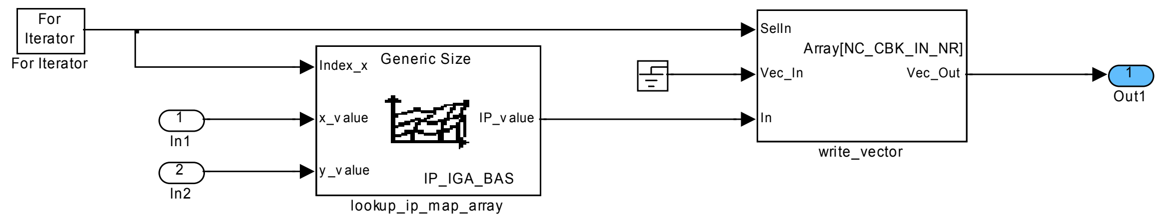
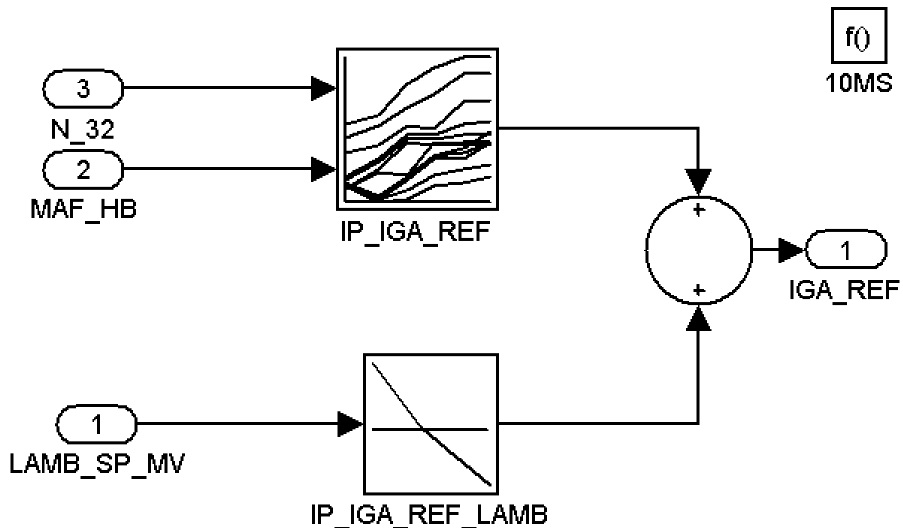


Figure 62 IGSP_DTSYSBAS0/ IGA_BAS_CLC/ Subsystem

6.19 Reference IGA calculation

Overview:

The reference ignition angle IGA_REF is a theoretical ignition angle at maximum engine torque.



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_REF	O/V	0..FFh	-35.625..60	0.375	°CRK
Reference ignition angle					

Input data:

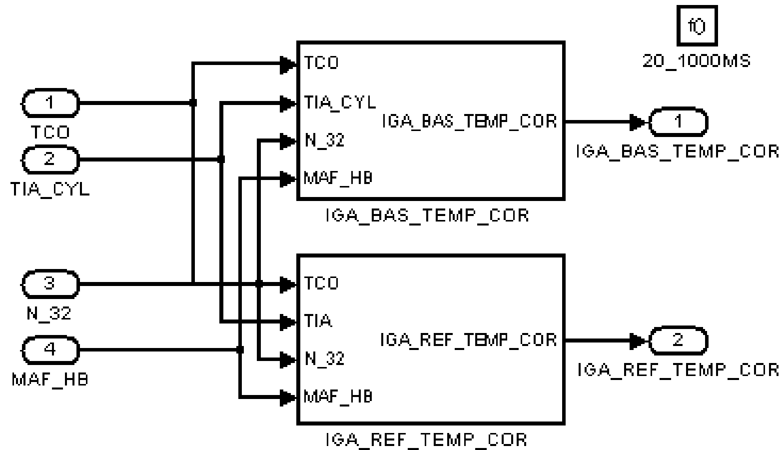
LV_HOM_RUN	N_32	MAF_HB	LAMB_SP_MV

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_IGA_REF_LAMB	8	0...FFH	-48...47.625	0.375	°CRK
LDPM_LAMB_SP_MV_1_IGSP	8	0...7FFFH	0...31.99	9.763E-4	-
additive lambda setpoint correction on IGA_REF					
IP_IGA_REF	16 x 12	0...FFH	-35.625...60	0.375	°CRK
LDPM_N_32_1_IGSP	16	0...FFH	0...8.16E3	32	rpm
LDPM_MAF_HB_1_IGSP	12	0...FFH	0...1.389E3	5.447	mg/stk
Reference ignition angle					

6.20 Temperature correction on ignition angle (TIA_CYL & TCO)

Overview:



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_BAS_TEMP_COR	O/V	80..7Fh	-48..47.625	0.375	°CRK
additive temperature correction of Basic IGA					
IGA_REF_TEMP_COR	O/V	80..7Fh	-48..47.625	0.375	°CRK
additive temperature correction of Reference IGA					

Input data:

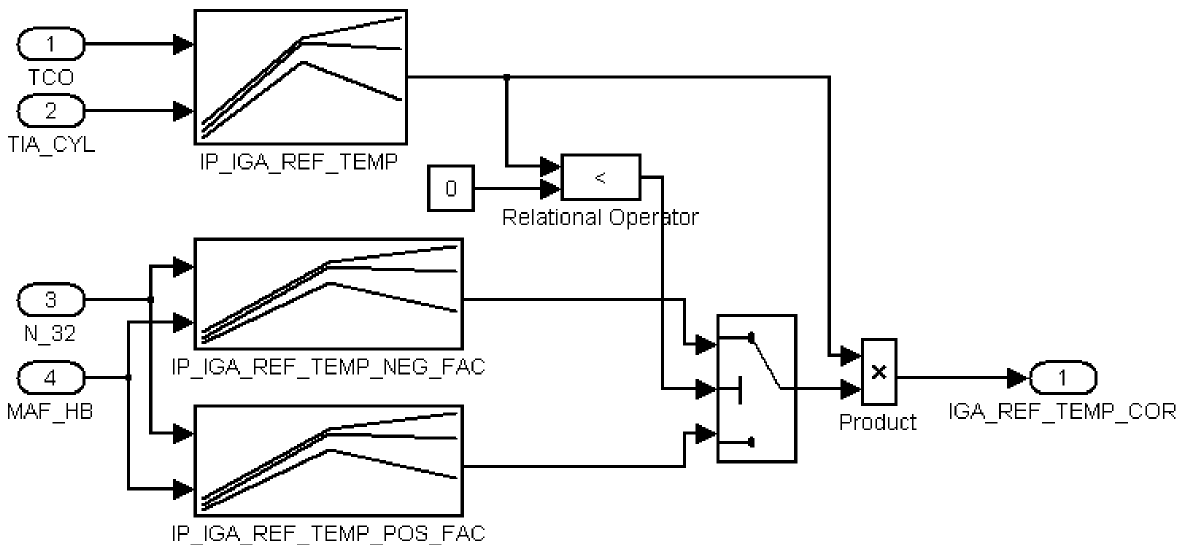
LV_HOM_RUN	N_32	MAF_HB	TCO
TIA_CYL			

Calibration data:

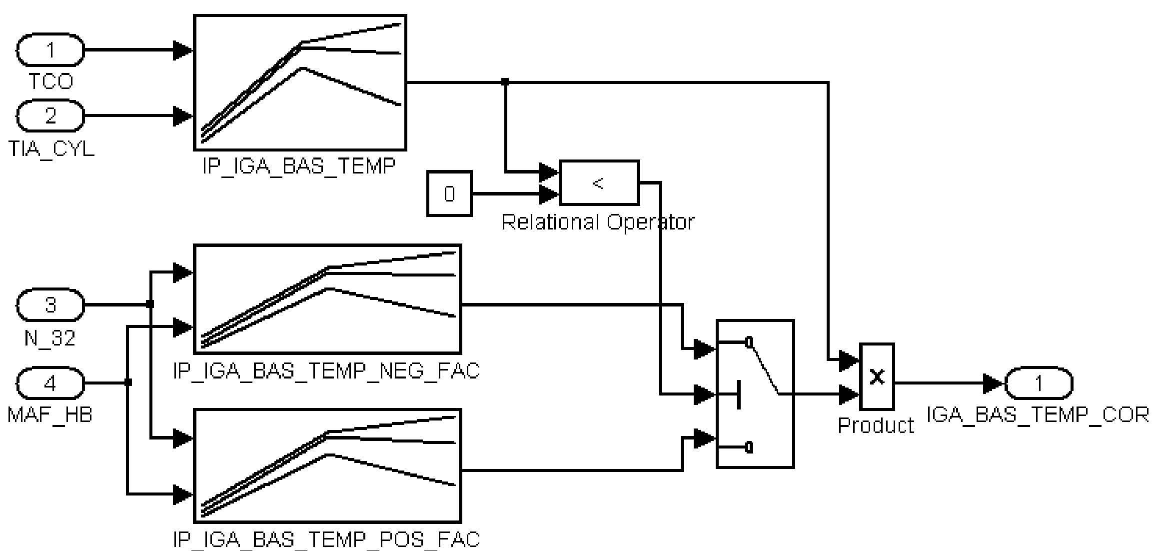
Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_IGA_REF_TEMP	8 x 8	0...FFH	-48...47.625	0.375	°CRK
LDPM_TCO_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
LDPM_TIA_CYL_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
Basis for temperature correction of Reference IGA versus temperatutre – Maximum corrective ignition angle					
IP_IGA_REF_TEMP_NEG_FAC	8 x 6	0...FFH	0...0,9960	3,90E-3	-
LDPM_N_32_2_IGSP	8	0...FFH	0...8.16E3	32	rpm
LDPM_MAF_HB_2_IGSP	6	0...FFH	0...1.389E3	5.447	mg/stk
Weighting factor for negative temperature correction of Reference IGA versus N_32, MAF_HB					
IP_IGA_REF_TEMP_POS_FAC	8 x 6	0...FFH	0...0,9960	3,90E-3	-
LDPM_N_32_2_IGSP	8	0...FFH	0...8.16E3	32	rpm
LDPM_MAF_HB_2_IGSP	6	0...FFH	0...1.389E3	5.447	mg/stk
Weighting factor for positive temperature correction of Reference IGA versus N_32, MAF_HB					
IP_IGA_BAS_TEMP	8 x 8	0...FFH	-48...47.625	0.375	°CRK
LDPM_TCO_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
LDPM_TIA_CYL_1_IGSP	8	0...FEH	-48...142.5	0.75	°C
Basis for temperature correction of Basic IGA versus temperature – Maximum corrective ignition angle					
IP_IGA_BAS_TEMP_NEG_FAC	8 x 6	0...FFH	0...0,9960	3,90E-3	-
LDPM_N_32_2_IGSP	8	0...FFH	0...8.16E3	32	rpm
LDPM_MAF_HB_2_IGSP	6	0...FFH	0...1.389E3	5.447	mg/stk
Weighting factor for negative temperature correction of Basic IGA versus N_32, MAF_HB					

IP_IGA_BAS_TEMP_POS_FAC	8 x 6	0...FFH	0...0,9960	3,90E-3	-
LDPM_N_32_2_IGSP	8	0...FFH	0...8.16E3	32	rpm
LDPM_MAF_HB_2_IGSP	6	0...FFH	0...1.389E3	5.447	mg/stk
Weighting factor for positive temperature correction of Basic IGA versus N_32, MAF_HB					

6.20.1 Temperature correction on Reference ignition angle



6.20.2 temperature correction on basic ignition angle



6.21 IVVT Ignition Angle Correction

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
IGA_BAS_VVT_COR	O/V	80...7FH	-48...47.625	0.375	°CRK
Additional basic ignition angle correction					
IGA_REF_VVT_COR	O/V	80...7FH	-48...47.625	0.375	°CRK
Additional reference ignition angle correction					
IGA_BAS_COR_IVVT_IN_i	V	80...7FH	-48...47.625	0.375	°CRK
Basic ignition angle correction; inlet contribution					
IGA_BAS_COR_IVVT_EX_i	V	80...7FH	-48...47.625	0.375	°CRK
Basic ignition angle correction; exhaust contribution					
FAC_IGA_BAS_COR_IVVT_IN_i	V	0...FFH	0...1.9921875	7.8125e-3	-
Correction factor for basic ignition angle correction; inlet					
FAC_IGA_BAS_COR_IVVT_EX_i	V	0...FFH	0...1.9921875	7.8125e-3	-
Correction factor for basic ignition angle correction; exhaust					
IGA_REF_COR_IVVT_IN_i	V	80...7FH	-48...47.625	0.375	°CRK
Reference ignition angle correction; inlet contribution					
IGA_REF_COR_IVVT_EX_i	V	80...7FH	-48...47.625	0.375	°CRK
Reference ignition angle correction; exhaust contribution					
FAC_IGA_REF_COR_IVVT_IN_i	V	0...FFH	0...1.9921875	7.8125e-3	-
Correction factor for reference ignition angle correction; inlet					
FAC_IGA_REF_COR_IVVT_EX_i	V	0...FFH	0...1.9921875	7.8125e-3	-
Correction factor for basic reference angle correction; exhaust					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	MAF	N
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i	VO_IND_i	LV_ES
NC_NR_CBK_IVVT	NLC_IVVT_IN	NLC_IVVT_EX	

FUNCTION DESCRIPTION:

General information:

The valve overlap influences the gas exchange process, and therefore, the amount of the residual gases in the combustion chamber. The portion of the residual gases influences the ignition delay and combustion velocity. Effects of the valve overlap on the ignition angle are taken into account. It is realized by means of correction values that are to add to the normal ignition angle.

Description:

The correction IGA_BAS_VVT_COR influences the actual ingnition angle, IGA_REF_VVT_COR is used for the reference ignition angle used in torque management. The algorithm for the basic and for the reference ignition angle correction is the same.

The primary ignition angle correction is calculated from the map IP_IGA_BAS(_REF)_COR_IVVT_IN(_EX) dependent on MAF and N. A correction value is to calibrate for such valve overlap that is prevailing in the corresponding operating point. There can be different valve overlap due to the camshaft position setpoint strategy or a failure. The primary correction is then corrected by means of IP_IGA_BAS(_REF)_COR_IVVT_IN(_EX) dependent on the actual camshaft position CAM_AV_IVVT_IN(_EX)_i and VO_IND_i. The correction for an individual camshaft is then IGA_BAS(_REF)_COR_IVVT_IN(_EX)_i.

The resulting ignition angle correction IGA_BAS(_REF)_VVT_COR depends on the configuration of the IVVT system. If there is only one camshaft effectively phased the resulting correction equals the individual camshaft correction. If both inlet and exhaust are phased the resulting correction is the sum of the inlet and of the exhaust individual correction. In case of the system with two camshaft banks (both cylinder lines effectively phased) the minimum individual correction from bank 1 and bank 2 is taken.

Application conditions:

Initialization:

At reset, at deactivation:

IGA_BAS_VVT_COR = 0

IGA_REF_VVT_COR = 0

Recurrence:

20 ms

Activation:

LV_ES = 0

Deactivation:

Not activation

Signal flow diagram:

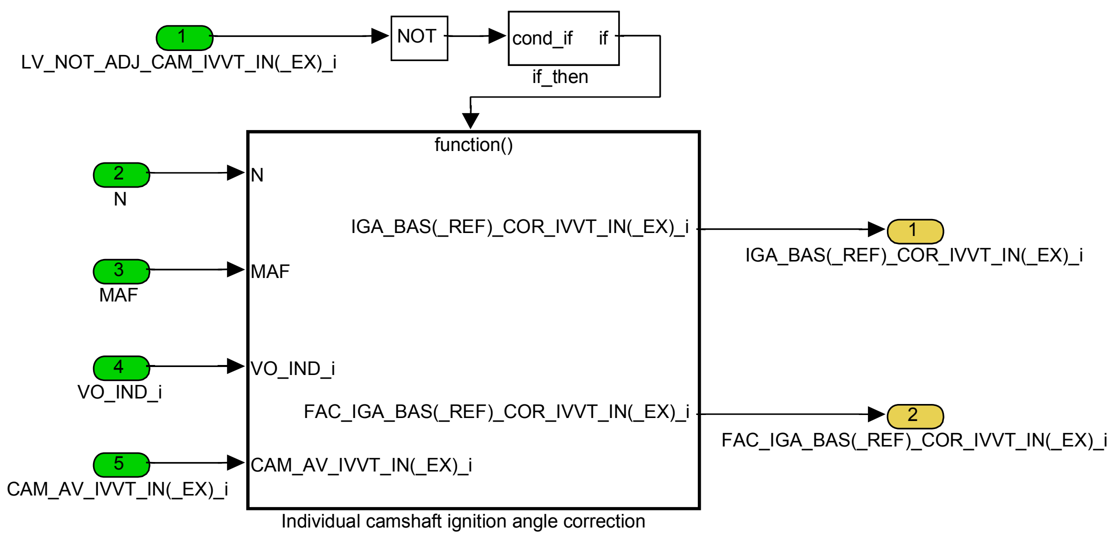


Figure: Ignition angle correction of individual camshaft.

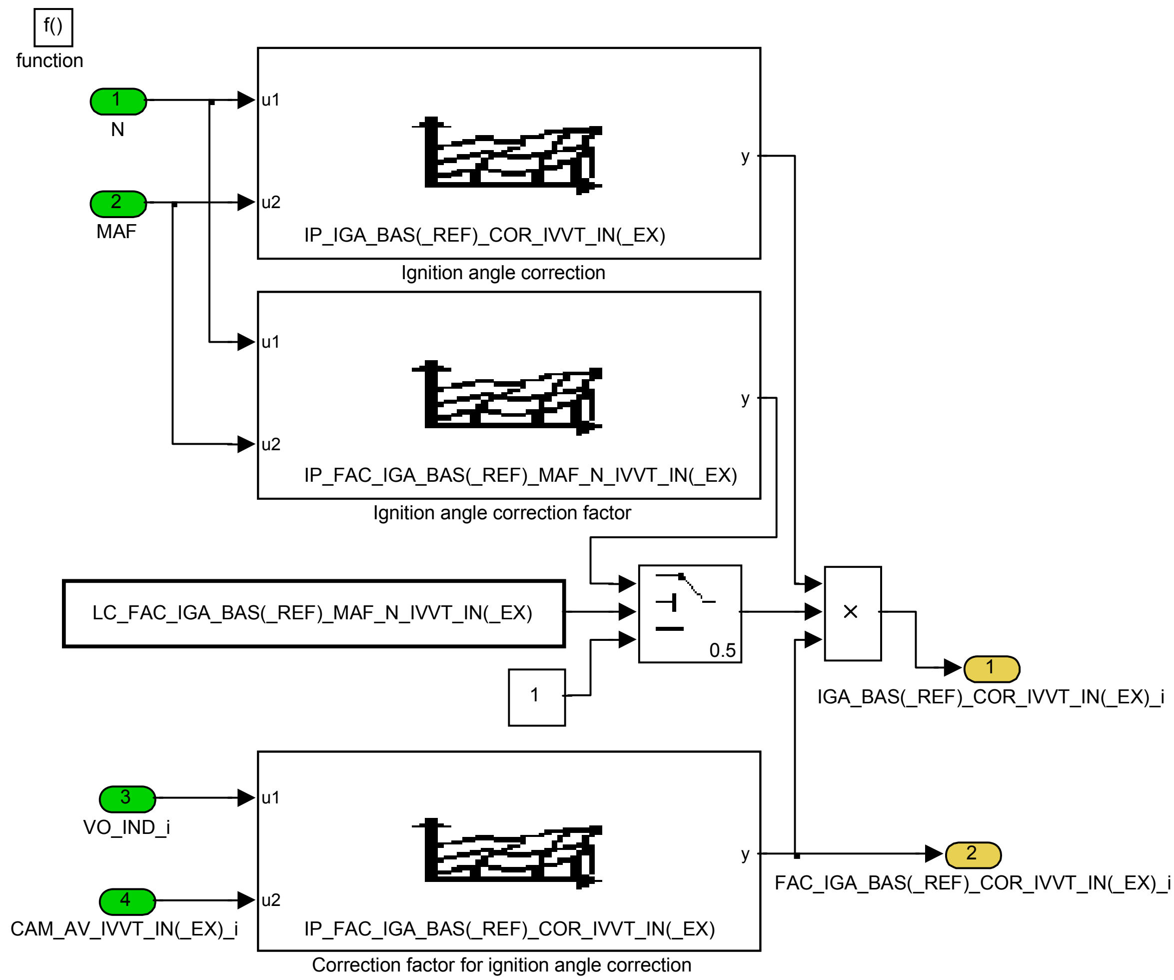


Figure: Calculation of ignition angle correction of individual camshaft.

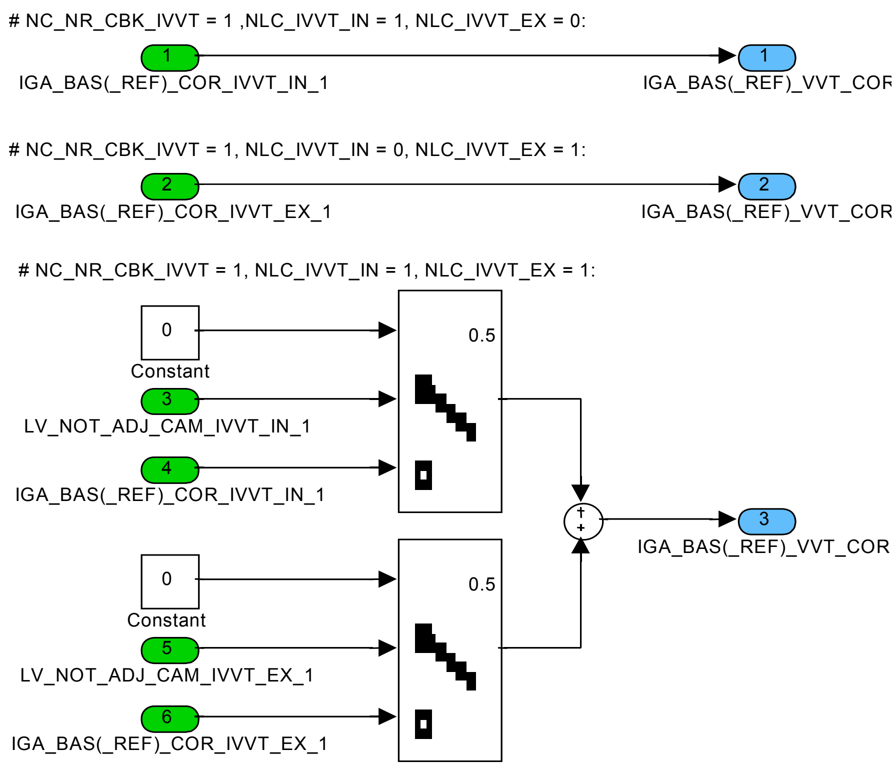


Figure: Resulting ignition angle correction for NC_NR_CBK_IVVT = 1.

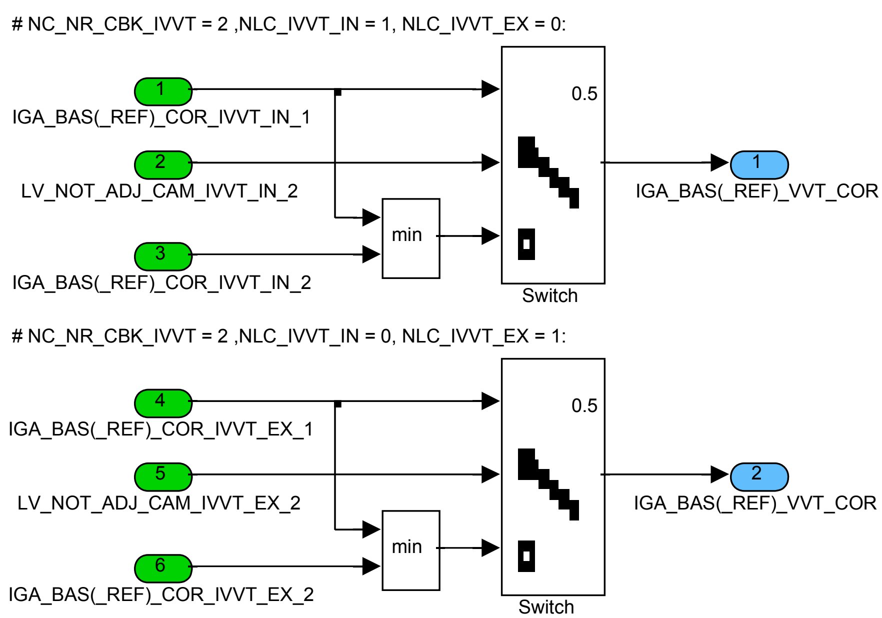


Figure: Resulting ignition angle correction for NC_NR_CBK_IVVT = 2 NLC_IVVT_IN = 1 OR NLC_IVVT_EX = 1.

NC_NR_CBK_IVVT = 2 ,NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

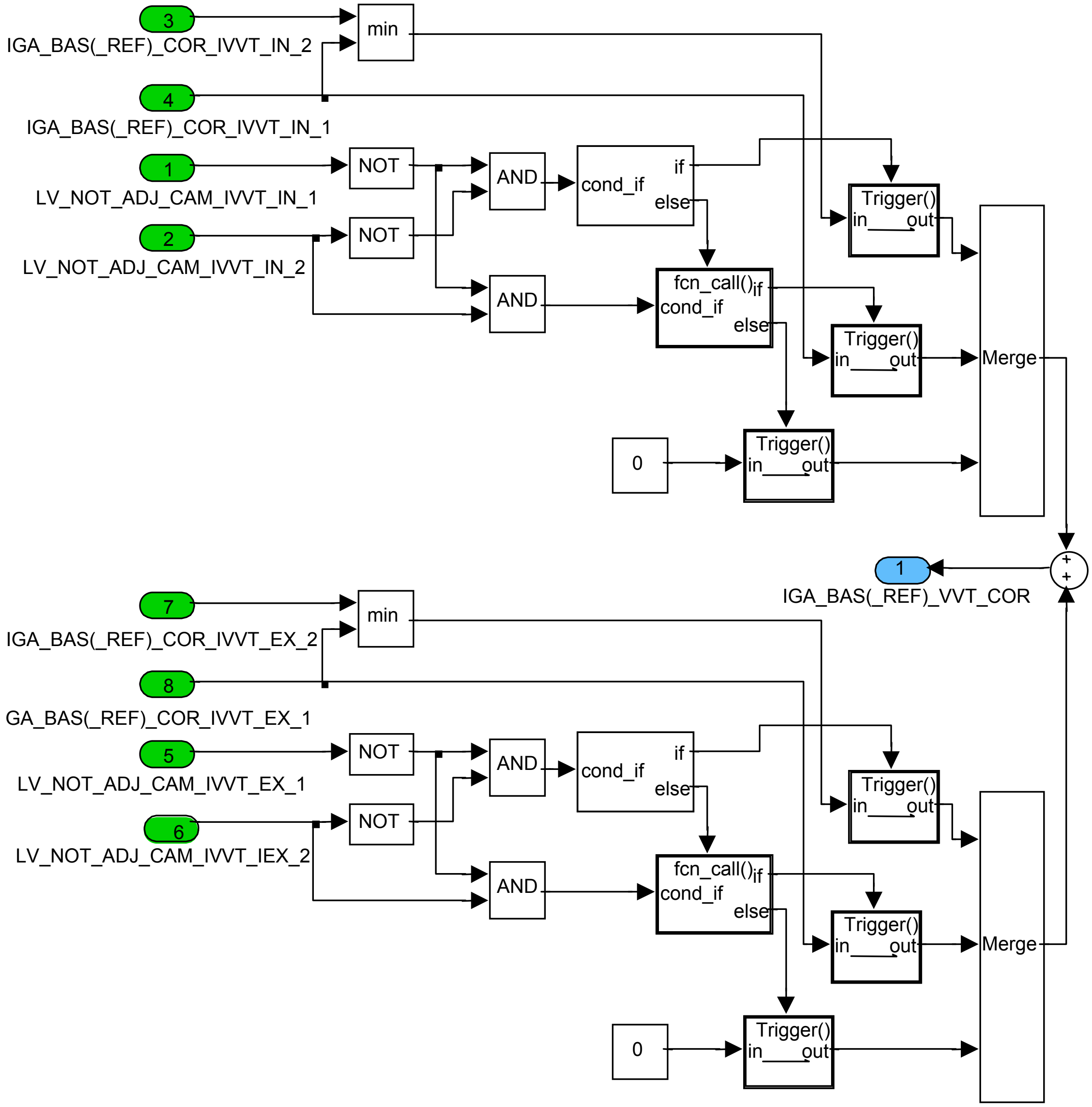


Figure: Resulting ignition angle correction for NC_NR_CBK_IVVT = 2 NLC_IVVT_IN = 1 AND NLC_IVVT_EX = 1.

Formula section:

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_FAC_IGA_BAS_COR_IVVT_IN	6x6	0...FFH	0...1.9921875	7.8125e-3	-
LDPM_VO_IND_1_VVTI	6	0...1C7H	0...170.625	0.375	°CRK
LDPM_CAM_AV_IVVT_IN_1_VVTI	6	0...FFH	60...155.625	0.375	°CRK
Correction factor for basic ignition angle correction; inlet					
IP_FAC_IGA_BAS_COR_IVVT_EX	6x6	0...FFH	0...1.9921875	7.8125e-3	-
LDPM_VO_IND_1_VVTI	6	0...1C7H	0...170.625	0.375	°CRK
LDPM_CAM_AV_IVVT_EX_1_VVTI	6	0...FFH	-40.125...-135.75	0.375	°CRK
Correction factor for basic ignition angle correction; exhaust					
IP_IGA_BAS_COR_IVVT_IN	16x12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Basic ignition angle correction; inlet contribution					
IP_IGA_BAS_COR_IVVT_EX	16x12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Basic ignition angle correction; exhaust contribution					
LC_FAC_IGA_BAS_MAF_N_IVVT_IN	1	0...1H	0...1	1	-
Switch for auxiliary factor for basic ignition angle correction; inlet					
LC_FAC_IGA_BAS_MAF_N_IVVT_EX	1	0...1H	0...1	1	-
Switch for auxiliary factor for basic ignition angle correction; exhaust					
IP_FAC_IGA_BAS_MAF_N_IVVT_IN	16x12	0...FFH	-8...7.9375	0.0625	-
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Auxiliary factor for basic ignition angle correction; inlet contribution					
IP_FAC_IGA_BAS_MAF_N_IVVT_EX	16x12	0...FFH	-8...7.9375	0.0625	-
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Auxiliary factor for basic ignition angle correction; exhaust contribution					
IP_FAC_IGA_REF_COR_IVVT_IN	6x6	0...FFH	0...1.9921875	7.8125e-3	-
LDPM_VO_IND_1_VVTI	6	0...1C7H	0...170.625	0.375	°CRK
LDPM_CAM_AV_IVVT_IN_1_VVTI	6	0...FFH	60...155.625	0.375	°CRK
Correction factor for reference ignition angle correction; inlet					
IP_FAC_IGA_REF_COR_IVVT_EX	6x6	0...FFH	0...1.9921875	7.8125e-3	-
LDPM_VO_IND_1_VVTI	6	0...1C7H	0...170.625	0.375	°CRK
LDPM_CAM_AV_IVVT_EX_1_VVTI	6	0...FFH	-40.125...-135.75	0.375	°CRK
Correction factor for reference ignition angle correction; exhaust					
IP_IGA_REF_COR_IVVT_IN	16x12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Reference ignition angle correction; inlet contribution					
IP_IGA_REF_COR_IVVT_EX	16x12	0...FFH	-48...47.625	0.375	°CRK
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Reference ignition angle correction; exhaust contribution					
LC_FAC_IGA_REF_MAF_N_IVVT_IN	1	0...1H	0...1	1	-
Switch for auxiliary factor for reference ignition angle correction; inlet					
LC_FAC_IGA_REF_MAF_N_IVVT_EX	1	0...1H	0...1	1	-
Switch for auxiliary factor for reference ignition angle correction; exhaust					
IP_FAC_IGA_REF_MAF_N_IVVT_IN	16x12	0...FFH	-8...7.9375	0.0625	-
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Auxiliary factor for reference ignition angle correction; inlet contribution					
IP_FAC_IGA_REF_MAF_N_IVVT_EX	16x12	0...FFH	-8...7.9375	0.0625	-
LDPM_N_5_VVTI	16	0...1FE0	0...8160	1	rpm
LDPM_MAF_1_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
Auxiliary factor for reference ignition angle correction; exhaust contribution					

6.22 Application incidences for knock control, adaptation

6.22.1 Knock Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNK_CTL_DIS	O/V	0...1H	0...1	1	-
knock control disabled due to malfunctions					
IGA_CYL_KNK_x	V	0...80H	-48...0	0.375	°CRK
knock advance correction					
IGA_MV_KNK_ADD	V	0...80H	-48...0	0.375	°CRK
mean value of knock advance correction					

Input data:

LV_ERR_CRK	LV_ERR_CAM	LV_ERR_KNKS_1	IGA_TRA_KNK
IGA_ADJ_KNK_x	IGA_MV_ADJ_KNK	LV_LIH_ERR_CRK	

General information:

To activate the knock control, a macro function (KNK_CTL_DIS) is executed which checks if knock control is disabled due to malfunctions.

If KNK_CTL_DIS = 0 the conditions are fulfilled and the knock control is enabled.

Update Rate: every segment

Formula section:

if LV_ERR_CRK = 1 **or**
LV_LIH_ERR_CRK = 1 **or**
LV_ERR_CAM = 1 **or**
LV_ERR_KNKS_1 = 1
then KNK_CTL_DIS = 1
else KNK_CTL_DIS = 0
endif

IGA_CYL_KNK_x = IGA_ADJ_KNK_x + IGA_TRA_KNK

IGA_MV_KNK_ADD = IGA_MV_ADJ_KNK + IGA_TRA_KNK

The Calculation of IGA_CYL_KNK_x is done at every TDC and for every cylinder.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_IGA_LGRD_3	1	100 ... 7FFFH	0,375 ... 48	0,00146	°crk
gradient limitation when underspending knock load threshold					

6.22.2 Knock control adaptation

6.22.2.1 Adaptation circuit 1

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNK_AD1_DIS	O/V	0...1H	0...1	1	-
adaptation circuit 1 disabled					
LV_KNK_AD_RD_ENA	O/V	0...1H	0...1	1	-
reading from adaptation table enabled					

Input data:

IGA_SP	IGA_AV_MV	TQ_ADD_CH	KNK_CTL_DIS
LV_ERR_TPS	LV_ERR_IGC	LV_ERR_TIA	LV_ERR_TCO
LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_ERR_MAF	LV_ERR_IV[NC_CYL_NR]	LV_IGA_AND_INJ_SWI
LV_TQ_IGA_ACT	LV_ERR_KNKS_1	LV_ERR_CAM_IN_i	LV_ERR_FSD_LAM_LIM_i
LV_ERR_LOAD_PLAUS	LV_ERR_TPS_PLAUS	LV_ERR_MAP	

FUNCTION DESCRIPTION:

General information:

The values are always read. If the conditions for adaptation circuit are not fullfilled, the values are only read but not newly registered.

Formula section:

The conditions for adaptation circuit 1 disabled are fulfilled:

IF

- KNK_CTL_DIS = 1 **or**
- TQ_ADD_CH > 0 **or**
- IGA_SP < IGA_AV_MV **or**
- LV_ERR_TIA = 1 **or**
- LV_ERR_TCO = 1 **or**
- LV_ERR_TPS = 1 **or**
- LV_ERR_MAF = 1 **or**
- LV_ERR_MAP = 1 **or**
- LV_ERR_IGC = 1 **or**
- LV_ERR_IV_x = 1 **or**
- LV_ERR_LSH_UP_i = 1 **or**
- LV_ERR_KNKS_1 = 1 **or**
- LV_ERR_CAM_IN_i = 1 **or**
- LV_ERR_FSD_LAM_LIM_i = 1 **or**
- LV_ERR_LOAD_PLAUS = 1 **or**
- LV_ERR_TPS_PLAUS = 1 **or**
- LV_TQ_IGA_ACT = 1 **or**
- LV_IGA_AND_INJ_SWI = 0

THEN

KNK_AD1_DIS = 1

ELSE
KNK_AD1_DIS = 0

LV_KNK_AD_RD_ENA = 1

6.22.2.2 Adaptation circuit 2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNK_AD2_DIS	O/V	0...1H	0...1	1	-
adaptation circuit 2 disabled					

Input data:

MAF_KNK	LV_ERR_TIA	LV_ERR_IGC	LV_ERR_IV[NC_CYL_NR]
LV_ERR_KNKS_1	LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_ERR_MAF	LV_ERR_CAM_IN_i
LV_ERR_TCO	LV_ERR_TPS	LV_ERR_FSD_LAM_LIM_i	LV_ERR_LOAD_PLAUS
LV_ERR_TPS_PLAUS	LV_ERR_MAP		

Formula section:

The conditions for adaptation circuit 2 disabled are fulfilled:

IF MAF_KNK < C_MAF_MIN_KNK_AD2 **or**
LV_ERR_TIA = 1 **or**
LV_ERR_IGC = 1 **or**
LV_ERR_IV[NC_CYL_NR] = 1 **or**
LV_ERR_KNKS_1 = 1 **or**
LV_ERR_LSH_UP[NC_CBK_EX_NR] = 1 **or**
LV_ERR_MAF = 1 **or**
LV_ERR_MAP = 1 **or**
LV_ERR_CAM_IN_i = 1 **or**
LV_ERR_TCO = 1 **or**
LV_ERR_TPS = 1 **or**
LV_ERR_FSD_LAM_LIM_i = 1 **or**
LV_ERR_LOAD_PLAUS = 1 **or**
LV_ERR_TPS_PLAUS = 1
THEN KNK_AD2_DIS = 1
ELSE KNK_AD2_DIS = 0

Calibration data:

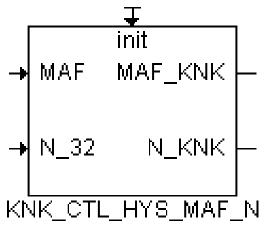
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_MIN_KNK_AD2	1	0...FFH	0...1389	5.447	mg/stk
Minimum air-mass for adaptation circuit 2					

6.23 Knock Control

6.23.1 Overview

6.23.2 Pre-processing

6.23.2.1 Hysteresis Function for MAF and N_32



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_KNK	V/O	0...FFH	0...1389	5.447	mg/Stroke
Mass air flow with hysteresis					
N_KNK	V	0...FFH	0...8160	32	1/min
Engine speed with hysteresis					

Input data:

MAF	N_32		

General information:

Hysteresis functions for engine speed and air-mass changes are defined for the tables relevant for the knock control in order to avoid steps recognisable for the driver due to small changes of the operation point.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: MAF_KNK = N_KNK = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

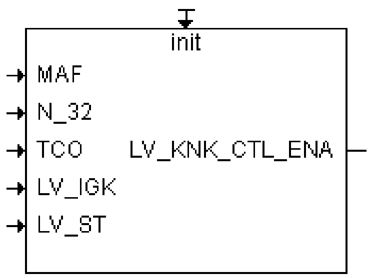
if **|(N_32 - N_KNK)| >= C_N_IGA_HYS_KNK**
N_KNK = N_32
else
N_KNK remains unchanged
end if

if **|(MAF - MAF_KNK)| >= C_MAF_IGA_HYS_KNK**
MAF_KNK = MAF
else
MAF_KNK remains unchanged
end if

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_IGA_HYS_KNK	1	0....FFH	0...8160	32	1/min
Speed Hysteresis for Knock Control					
C_MAF_IGA_HYS_KNK	1	0....FFH	0...1389	5.447	mg/Stroke
Air-Mass Hysteresis for Knock Control					

6.23.2.2 Knock Control Enable



KNK_CTL_ENA

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_KNK_CTL_ENA	V/O	0...1H	0...1	1	-
Boolean for knock control enabled					

Input data:

MAF_KNK	N_32	TCO	LV_IGK
LV_ST			

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: LV_KNK_CTL_ENA = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

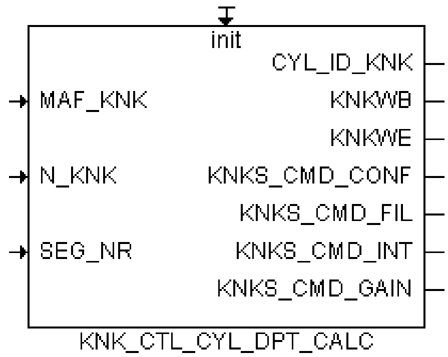
```
if      (LV_ST = 1  or  LV_IGK = 0)
        LV_KNK_CTL_ENA = 0
else
        if      (MAF_KNK  >= IP_MAF_MIN_KNK__N_32__TCO)
                LV_KNK_CTL_ENA = 1
        else
             LV_KNK_CTL_ENA = 0
        end if
end if
```

If the relevant air-mass is below the air-mass threshold (LV_KNK_CTL_ENA = 0), only the noise value calculation is executed.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_MIN_KNK__N_32__TCO	8*4	0....FFH	0...1389	5.447	mg/Stroke
LDPM_N_32_3_9	8	0....FFH	0...8160	32	rpm
LDP_TCO__MAF_MIN_KNK	4	0....FEH	-48...142.5	0.75	°C
Air-mass Threshold Release KNK					

6.23.2.3 Cylinder Dependent Calculations



6.23.2.3.1 Cylinder Number for Knock Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CYL_ID_KNK	V/O	0...FFH	0...255	1	-
cylinder number for knock control					

Input data:

SEG_NR			

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

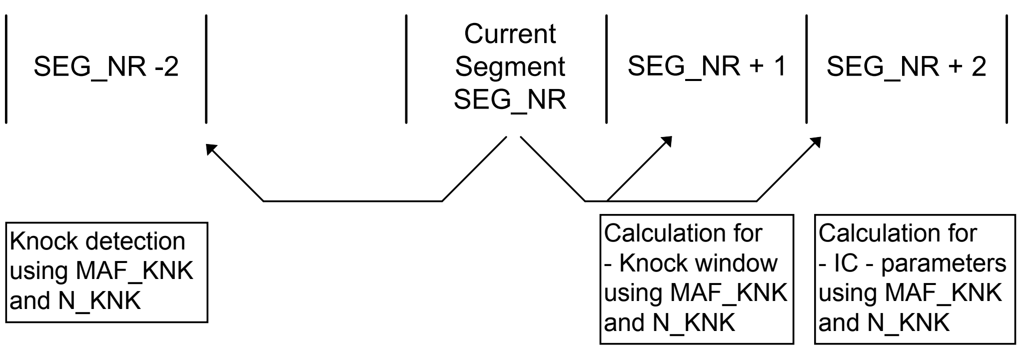
Initialization: CYL_ID_KNK = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

CYL_ID_KNK = SEG_NR - 2

Description:



6.23.2.3.2 Knock Measurement Window

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNKWB_x	V/O	0... RNG_WB_HEX	0… RNG_WB	0.375	° CRK
Beginning of Measurement Window Cyl.x					
KNKWE_x	V/O	1... RNG_WE_HEX	0.375... RNG_WE	0.375	° CRK
End of Measurement Window Cyl.x					

Input data:

MAF_KNK	N_KNK	SEG_NR	

General information:

The knock detection is performed for cylinder-individually defined measurement windows since knock-typical oscillations only occur for certain crankshaft angles. The measurement windows (crank angle of beginning and end) are specified as function of the engine speed and load.
If the estimated 'End of the Knock Measurement Window' is lower or equal compared to the 'Beginning of the Knock measurement Window', both values are initialized.

The crank angle for the beginning of the knock measurement window is taken for each cylinder from the respective table IP_KNKWB_x according to the firing order.

The crank angle for the end of the knock measurement window is taken for each cylinder from the respective table IP_KNKWE_x according to the firing order.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: at reset

KNKWB_x = NC_KNKWB_INI

KNKWE_x = NC_KNKWE_INI

Update Rate: every segment

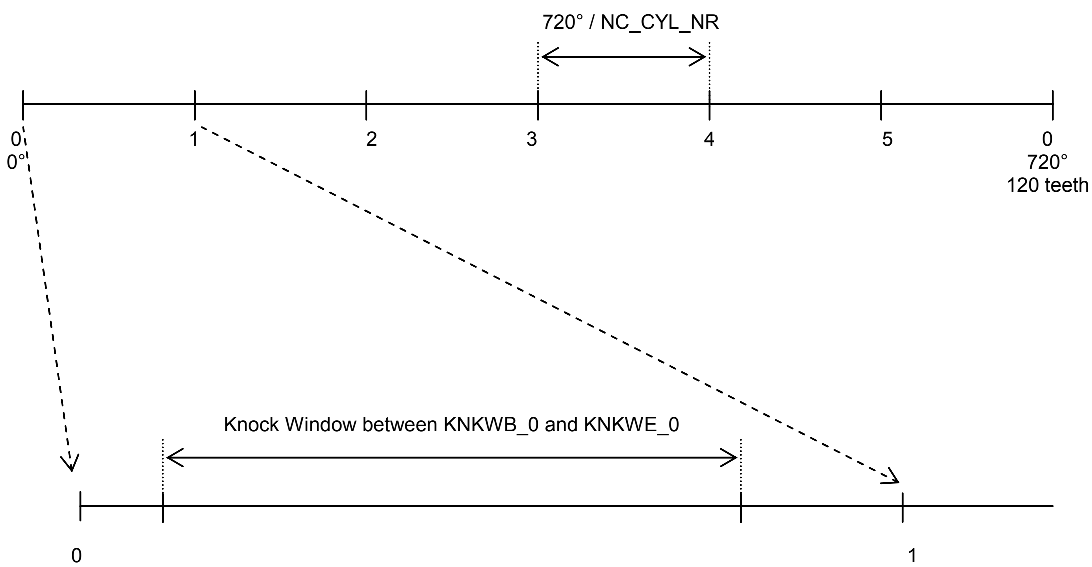
Formula section:

Definition of cylinder dependent knock window ranges:

```
RNG_WB        = (720° / NC_CYL_NR) - 12°
RNG_WB_HEX    = HEX(RNG_WB / 0.375°)
RNG_WE        = (720° / NC_CYL_NR) - 6°
RNG_WE_HEX    = HEX(RNG_WE / 0.375°)

KNKWB_x = IP_KNKWB_x__N__MAF
KNKWE_x = IP_KNKWE_x__N__MAF

if    (KNKWE_x <= KNKWB_x)
      KNKWB_x = NC_KNKWB_INI
      KNKWE_x = NC_KNKWE_INI
end if
```

N.B.: x represents current SEG_NR + 1

Description:
(example for NC_CYL_NR=6; TDC offset = 0°CRK)



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_KNKWB_x__N__MAF	10*4	0... RNG_WB_HEX	0… RNG_WB	0.375	° CRK
LDPM_N_KNKWB	10	0...FFH	0...8160	32	rpm
LDPM_MAF_KNKWB	4	0...FFH	0...1389	5.44	mg/stk
Beginning of Measurement Window Cyl.x					
IP_KNKWE_x__N__MAF	10*4	1... RNG_WE_HEX	0.375... RNG_WE	0.375	° CRK
LDPM_N_KNKWB	10	0...FFH	0...8160	32	rpm
LDPM_MAF_KNKWB	4	0...FFH	0...1389	5.44	mg/stk
End of Measurement Window Cyl.x					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_KNKWB_INI	1	0... RNG_WB_HEX	0… RNG_WB	0.375	° CRK
Beginning of Measurement Window Cyl.x					
NC_KNKWE_INI	1	1... RNG_WE_HEX	0.375... RNG_WE	0.375	° CRK
End of Measurement Window Cyl.x					

6.23.2.3.3 Calibration of ATM40 device

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNKS_CMD_CONF_x	O	0...FFH	0...255	1	-
Configuration value for the analogue channel of knock sensor					
KNKS_CMD_FIL_x	V/O	0...3FH	0...63	1	-
Knock Filter Frequency and Reference Filter Frequency Cyl.x					
KNKS_CMD_INT_x	V/O	0...1FH	0...31	1	-
Integration Time Constant Cyl.x					
KNKS_CMD_GAIN_x	V/O	0...3FH	0...63	1	-
Amplification precontrol Cyl.x					
KNKS_CMD_GAIN_AD_x	V/S	E1...1FH	-31...31	1	-
Gain adaptation for Cyl. x (Signed offset)					
LV_KNK_AD	V	0...1H	0...1	1	-
Flag for adaptative learning of engine noise enabled					

Input data:

N_KNK	MAF_KNK	SEG_NR	NC_KNKS_CONF
LV_KNK_CTL_ENA	LV_KNK_TRA_MAF	LV_KNK_TRA_N	NL_x
N_32	LV_ERR_KNKS_1	C_NL_MIN_DIAG	

FUNCTION DESCRIPTION:

General information:

The knock control in the engine management works with the ATM40 device. The output value of the integrator after the end of the measurement window is fed as Knock Value KNKS through an analogue channel to the control algorithm.
(Further information may be obtained in "Applications Guide to ATM40".)

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization:
KNKS_CMD_INT_x = 0 at reset
KNKS_CMD_FIL_x = 0 at reset
KNKS_CMD_GAIN_x = 0 at reset
KNKS_CMD_CONF_x = value of conf. bit at position x in NC_KNKS_CONF at reset
KNK_CMD_GAIN_AD_x = 0 when ECU is brand new or at EEPROM error

Update Rate: every segment

Activation : **General condition for Cyl. x gain adaptation calculation :**

If LV_KNK_CTL_ENA = 1
And LV_KNK_TRA_MAF = 0
And LV_KNK_TRA_N = 0
And NL_x > C_NL_MIN_DIAG
And C_KNK_AD_N_MIN < N_32 < C_KNK_AD_N_MAX
And LV_ERR_KNKS_1 = 0
Then
LV_KNK_AD = 1
Else
LV_KNK_AD = 0

Formula section:

KNKS_CMD_INT_x = IP_ITC_KNK_x

KNKS_CMD_FIL_x = ID_FIL_FRQ_KNK_x

KNKS_CMD_GAIN_x = IP_GAIN_KNK_x + KNKS_CMD_GAIN_AD_x

KNKS_CMD_GAIN is limited to maximum 63.

Calculation of knock signal amplification controller for Cyl. x (Calculation takes place every C_KNK_AD_CTR TDCs for all cylinders) **:**

If LV_KNK_AD = 1 **during** C_KNK_AD_CTR TDCs

Then *Perform gain adaptation of all cylinders*

If | NL_x – IP_NL_KNK_AD | > C_NL_KNK_AD_THD

Then

If NL_x > IP_NL_KNK_AD

Then KNKS_CMD_GAIN_AD_x $_{i}$ = KNKS_CMD_GAIN_AD_x $_{i-1}$ + 1

Else KNKS_CMD_GAIN_AD_x $_{i}$ = KNKS_CMD_GAIN_AD_x $_{i-1}$ – 1

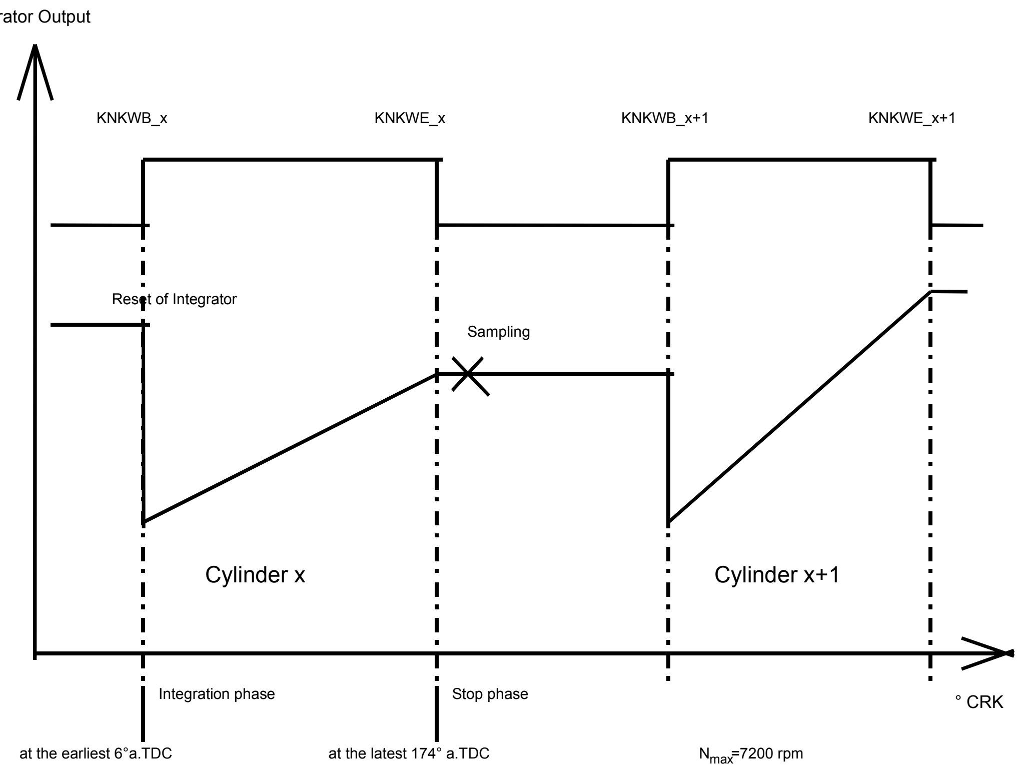
Else

KNKS_CMD_GAIN_AD_x remains unchanged

Limitation : KNKS_CMD_GAIN_AD_x is limited between C_KNK_CMD_GAIN_AD_MIN and C_KNK_CMD_GAIN_AD_MAX.

N.B.: x represents current SEG_NR + 2

Description (example for 4 cylinder engine):



Note: These calibration data have to be adjusted if the processor sampling rate is changed.

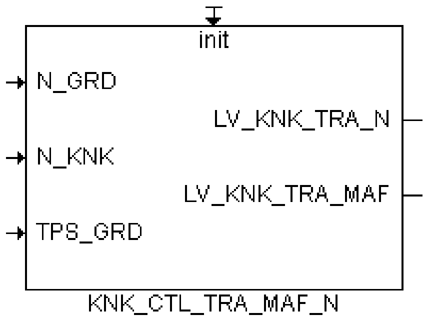
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_FIL_FRQ_KNK_x__N	4	0...3FH	0...63	1	-
LDPM_N_FIL_FRQ_KNK	4	0...FFH	0...8160	32	rpm
Knock Filter Frequency and Reference Filter Frequency Cyl.x					
IP_ITC_KNK_x__N__MAF	16*6	0...1FH	0...31	1	-
LDPM_N_2_9	16	0...FFH	0...8160	32	rpm
LDPM_MAF_1_9	6	0...FFH	0...1389	5.44	mg/stk
Integration Time Constant Cyl.x					
IP_GAIN_KNK_x__N__MAF	16*6	0...3FH	0...63	1	-
LDPM_N_2_9	16	0...FFH	0...8160	32	rpm
LDPM_MAF_1_9	6	0...FFH	0...1389	5.44	mg/stk
Amplification precontrol Cyl.x					
IP_NL_KNK_AD	4	0...FFFFH	0...5	7.63e-5	V
LDP_N_32_NL_KNK_AD	4	0...FFH	0...8160	32	Rpm
Target engine Noise Level					
C_KNK_AD_N_MIN	1	0...FFH	0...8160	32	Rpm
Minimum engine speed to perform knock signal amplification adpatation (typical value 1500 rpm)					
C_KNK_AD_N_MAX	1	0...FFH	0...8160	32	Rpm
Maximum engine speed to perform knock signal amplification adpatation (typical value 4500 rpm)					
C_KNK_AD_CTR	1	0...FFH	0...255	1	TDC
Number of TDC between two knock signal amplification adaptation (typical value 64 TDCs)					
C_NL_KNK_AD_THD	1	0...FFFFH	0...5	7.63e-5	V
Noise level window to perform knock signal amplification adaptation (typical value 0,2 Volt)					
C_KNK_CMD_GAIN_AD_MIN	1	E1...1FH	-31...31	1	-
Minimum value of knock signal amplification adaptation (typical value –15)					
C_KNK_CMD_GAIN_AD_MAX	1	E1...1FH	-31...31	1	-
Minimum value of knock signal amplification adaptation (typical value +15)					

Noncalibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_KNKS_CONF	1	0…FFH	0…255	1	-
Configuration value of analogue channel of knock sensor for all cylinders (eg. 4 cylinders: 00000000b)					

6.23.2.4 Load and Speed Dynamics



6.23.2.4.1 Speed Dynamics Function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_KNK_TRA_N	V	0...1H	0...1	1	-
Logical variable for Engine Speed Dynamic Function active or passive					

Input data:

N_GRD	N_KNK		

FUNCTION DESCRIPTION:

General information:

In order to keep track with the noise values for large engine speed gradients (LV_KNK_TRA_N = 1), the standard filtering constant is exchanged by a faster averaging constant C_NL_CRLC_TRA.
Care needs to be given to the absolute function evaluation, so that there is no difference in the treatment of positive and negative speed gradients.
Furthermore the values of Knock -Adaptation-circuits 1 + 2 are only read at active speed dynamics.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: LV_KNK_TRA_N = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

if **(|N_GRD| >= ID_N_GRD_MAX_KNK__N)**

LV_KNK_TRA_N = 1

else

LV_KNK_TRA_N = 0

end if

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_N_GRD_MAX_KNK__N	8	0...7FH	0...4064	32	1/min*s
LDPM_N_IGA_AD	8	0...FFH	0...8160	32	rpm
Speed Gradient for dynamic identification					

6.23.2.4.2 Load Dynamics Function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_KNK_TRA_MAF	V	0...1H	0...1	1	-
Logical variable for Load Dynamic Function active or passive					

Input data:

MAF_DIF	N_KNK		

FUNCTION DESCRIPTION:

General information:

In order to keep track with the noise values for large (positive) load gradients (LV_KNK_TRA_MAF = 1), the standard filtering constant is exchanged by a faster averaging constant C_NL_CRLC_TRA. Load dynamics is recognised, as soon as the MAF difference exceeds the threshold ID_MAF_DIF_TRA_KNK__N. After the triggering of the load dynamics the special filtering constant C_NL_CRLC_TRA is applied for the number of ID_CYCNR_FAC_TRA_KNK__N cycles. The cycle counter CYCNR_FAC_TRA_KNK is newly re-loaded as long as the MAF difference exceeds the threshold ID_MAF_DIF_TRA_KNK__N. In addition, at active load dynamics function, the knock detection (KNK_THD_x) is made less sensitive with the help of the factor C_FAC_TRA_KNK.
The values from the knock adaptation-circuits 1 + 2 are only read.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: LV_KNK_TRA_MAF = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

if **(MAF_DIF > ID_MAF_DIF_TRA_KNK__N)**

LV_KNK_TRA_MAF = 1 for ID_CYCNR_FAC_TRA_KNK__N cycles

else

if **(ID_CYCNR_FAC_TRA_KNK__N has elapsed)**
LV_KNK_TRA_MAF = 0

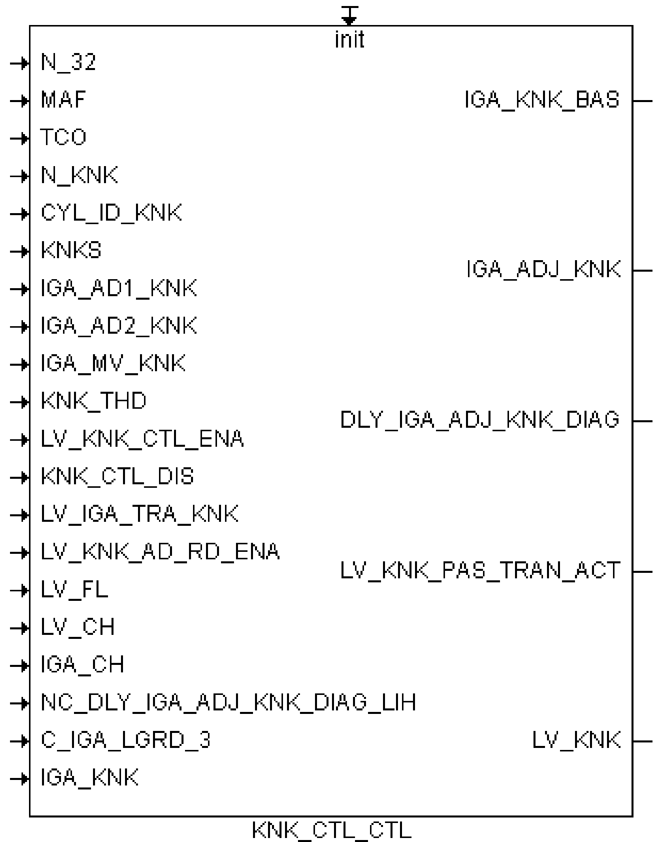
end if

end if

Calibration data:

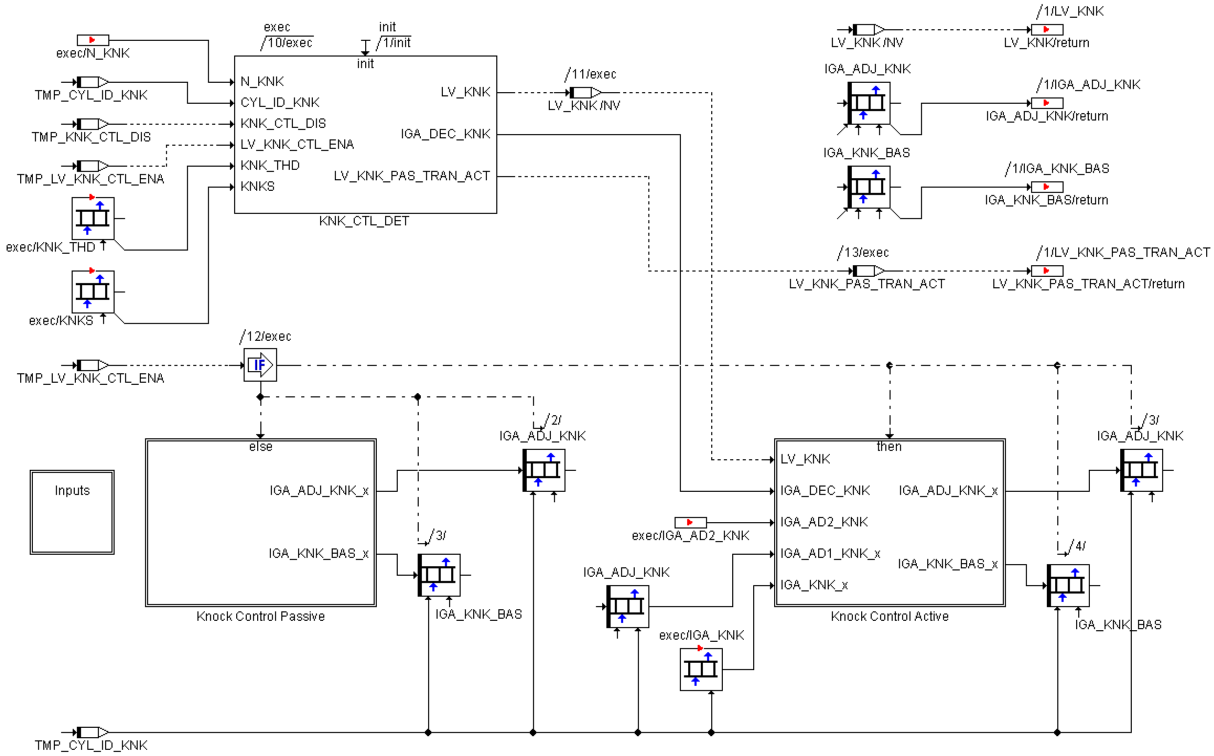
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_MAF_DIF_TRA_KNK__N	4	8000...7FFFH	-694.5...694.5	0.0212	mg/stk
LDPM_N_6_4	4	0...FFH	0...8160	32	rpm
MAF difference for Load Dynamic Recognition					
ID_CYCNR_FAC_TRA_KNK__N	4	0...FFH	0...255	1	*720/Cyl.° CRK
LDPM_N_6_4	4	0...FFH	0...8160	32	rpm
Cycle counter during which the Load Dynamic Function stays active after triggering					

6.23.3 Knock Control



<u>Overview:</u>

```
→ Knock Detection
if    (LV_KNK_CTL_ENA = 1)
      → Knock Control active
            → Calculation of Maximum Retard Adjustment
            → Calculation of Total Sum of Adaptation Circuit 1 and 2
            if    IGA_AD1_AD2_KNK_x  > IGA_ADJ_MAX_KNK – IGA_WOUT_KNK
                  if    (KNK_CTL_DIS = 1)
                        → Limp Home operation
                  else
                        → Normal operation
                        → Control excursion limit
                  end if
                  → Limitation of total ignition retard
            else
                  → Reset of Knock control
            end if
else
      → Knock control passive (Reset after abandoning the Knock-Operating-Limit)
end if
```



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_KNK	V/O	0...1H	0...1	1	-
Logical variable for recognition of knock					
IGA_DEC_KNK	-	1..7FH	0.375...47.625	0.375	°CRK /720°CRK
Spark retard at recognised knocking					
IGA_ADJ_KNK_x	V/O	0...80H	-48...0	0.375	°CRK
Total spark retard through knock control with adaptation (cylinder individual)					
IGA_KNK_BAS_x	-	0...80H	-48...0	0.375	°CRK
Spark retard through knock control prior adaptation (cylinder individual)					
IGA_ADJ_MAX_KNK	V	0 ... FFh	-35.625...60	0.375	°CRK
Total spark retard through knock control including adaptation (cylinder individual)					
FAC_IGA_MAX_KNK	V	0...FFH	0…0.996	0.0039	-
Weighting factor for the max. spark retard against coolant					
IGA_AD1_AD2_KNK_x	V	0...80H	-48...0	0.375	°CRK
Adaptation value for Cyl. x (Adaptation circuit 1 and 2)					
DLY_IGA_ADJ_KNK_DIAG	V	0...FFH	0...25.5	0.1	s
Inhibition time for knock adaptation					
LV_KNK_PAS_TRAN_ACT	-	0...1H	0...1	1	-
Transition for knock control passive to active					
IGA_ADJ_KNK_CTL_DIS	V	0 ... FFh	-35.625...60	0.375	°CRK
Spark retard during limp home operation (KNK_CTL_DIS = 1)					

Input data:

LV_FL	LV_CH	LV_IGA_TRA_KNK	LV_KNK_CTL_ENA
KNK_CTL_DIS	IGA_CH	LV_KNK_AD_RD_ENA	CYL_ID_KNK
N_32	MAF	N_KNK	TCO
KNKSx	KNK_THD_x	IGA_AD1_KNK_x	IGA_AD2_KNK
IGA_MV_KNK	C_IGA_LGRD_3	NC_DLY_IGA_ADJ_KNK_DIAG_LIH	IGA_WOUT_KNK

Application conditions:

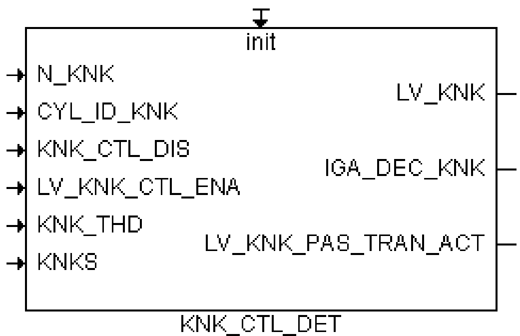
Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialisation: IGA_ADJ_KNK_x = 0 at reset and engine running to engine stop

Update Rate: every segment

6.23.3.1 Knock Detection



FUNCTION DESCRIPTION:

General information:

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialisation: -

Update Rate: every segment

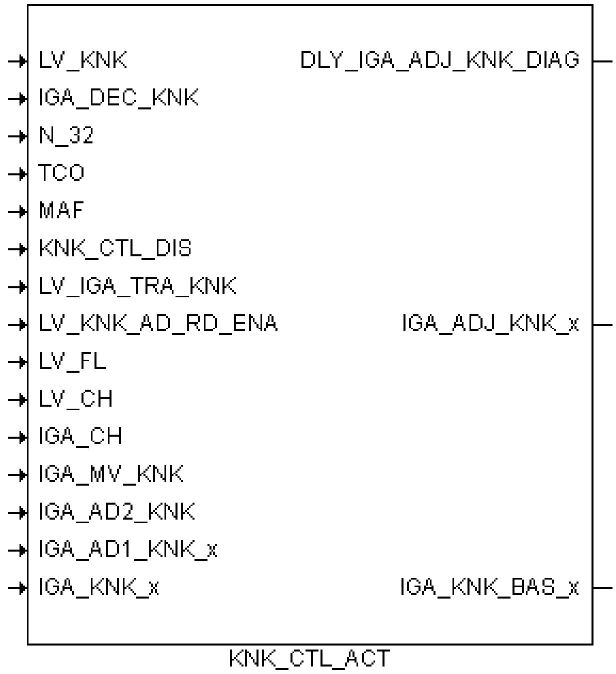
Formula section:

```
if      (LV_KNK_CTL_ENA = 1)
        Knock Control Active:
        if      (KNK_CTL_DIS = 0)
                if      (NC_CYL_NR cycles has elapsed after Passive or Limp-Home)
                        LV_KNK_PAS_TRAN_ACT = 1
                        if      (KNKSx >= KNK_THD_x)
                                LV_KNK = 1
                                if      (KNKSx >= 2 * KNK_THD_x)
                                        Intensity 2:
                                        IGA_DEC_KNK = ID_IGA_DEC_KNK_2
                                else
                                        Intensity 1:
                                        IGA_DEC_KNK = ID_IGA_DEC_KNK_1
                                end if
                        else
                                LV_KNK = 0
                        end if
                else
                        LV_KNK_PAS_TRAN_ACT = 0
                        LV_KNK = 0
                end if
        else
                Limp-Home Intensity 1:
                IGA_DEC_KNK = ID_IGA_DEC_KNK_1
                LV_KNK_PAS_TRAN_ACT = 0
                LV_KNK = 0
        end if
else
        Knock Control Passive:
        LV_KNK_PAS_TRAN_ACT = 0
        LV_KNK = 0
end if
```

N.B.: x represents current CYL_ID_KNK

Different adjustment decrements are also produced due to the different knock intensities. Then it is possible to react to hard knock-strikes with a greater ignition - timing retardation.

6.23.3.2 Knock control active



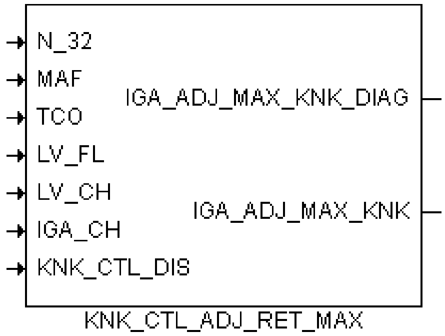
General information:

The spark retard is calculated cylinder-individual at every firing event. The spark retard due to the knock control system is determined by the spark retard due to knocking combustion (IGA_KNK_x) plus the spark retard due to the adaptation circuit 1 (IGA_AD1_KNK_x) and adaptation circuit 2 (IGA_AD2_KNK). The sum of these retard components is limited to a maximum spark retard.

Application conditions:

Activation:	at every engine state except engine stop
Deactivation:	LV_ES = 1
Initialisation:	-
Update Rate:	every segment

6.23.3.2.1 Calculation of Maximum Retard Adjustment



FUNCTION DESCRIPTION:

General information:

Please note that this maximum spark retard is an absolute ignition angle referring to TDC.

Application conditions:

Activation:	at every engine state except engine stop
Deactivation:	-
Initialisation:	-
Update Rate:	every segment

Formula section:

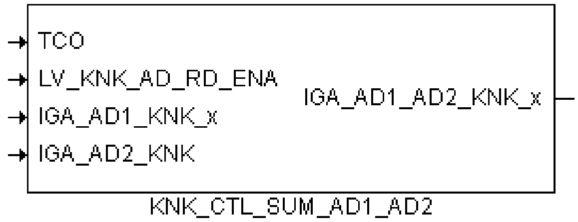
IGA_BAS_MAX_KNK = IP_IGA_BAS_MAX_KNK(N_32, MAF)

IGA_OFS_MAX_KNK = IP_IGA_OFS_MAX_KNK(N_32, MAF)

FAC_IGA_MAX_KNK = IP_FAC_IGA_MAX_KNK(TCO, TIA)

IGA_ADJ_MAX_KNK = IGA_BAS_MAX_KNK + (IGA_OFS_MAX_KNK * FAC_IGA_MAX_KNK)

6.23.3.2.2 Total sum of Adaptation Circuit 1 and 2



FUNCTION DESCRIPTION:

Application conditions:

Activation: at every engine state except engine stop

Deactivation: -

Initialisation: IGA_AD1_AD2_KNK_x = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

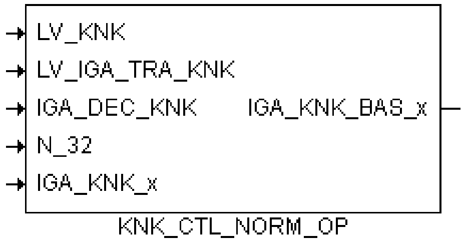
if LV_KNK_AD_RD_ENA = 1

then IGA_AD1_AD2_KNK_x = (IGA_AD1_KNK_x + IGA_AD2_KNK) * IP_FAC_AD_KNK_TCO__TCO

else IGA_AD1_AD2_KNK_x = 0

endif

6.23.3.2.3 Normal Operation



FUNCTION DESCRIPTION:

General information:

The spark retard is calculated cylinder-individual at every firing event. The spark retard due to the knock control system is determined by the spark retard due to knocking combustion (IGA_KNK_x) plus the spark retard due to the adaptation circuit 1 (IGA_AD1_KNK_x) and adaptation circuit 2 (IGA_AD2_KNK). The sum of these retard components is limited to a maximum spark retard.
In case of knock free operation (LV_KNK=0) and LV_KNK_CTL_ENA = 1 the Ignition retard due to knock control is re-set to zero (Controlled reset at Knock-free operation).
The controlled ignition-timing reset remains active until either the map ignition timing (IGA_KNK_BAS_x = 0) is reached or until knocking occurs.

Formula section:

if (LV_KNK = 1)

then IGA_KNK_BAS_x = IGA_KNK_x - IGA_DEC_KNK

else Controlled reset at Knock-free operation

  if (LV_IGA_TRA_KNK = 0)

  then IGA_KNK_BAS_x = IGA_KNK_x + IP_IGA_INC_KNK__N

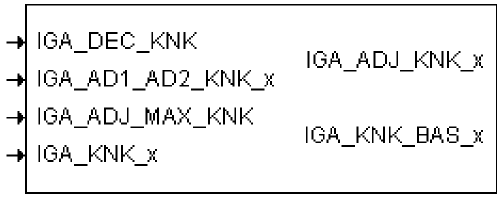
  else IGA_KNK_BAS_x = IGA_KNK_x

  end if

end if

N.B.: x represents current CYL_ID_KNK

6.23.3.2.4 Limp Home Operation



KNK_CTL_LIH

FUNCTION DESCRIPTION:

General information:

After an diagnosis error on the knock control system is detected (KNK_CTL_DIS = 1) the limp home spark retard level is achieved via spark retard Intensity 1 (IGA_DEC_KNK).
This ignition angle is also executed if a knock control related malfunction is detected (KNK_CTL_DIS = 1).

Formula section:

Knock control adaptation is inhibited for NC_DLY_IGA_ADJ_KNK_DIAG_LIH (*100ms)

IGA_ADJ_KNK_CTL_DIS = IP_IGA_KNK_CTL_DIS(N_32, MAF)

check if limphome spark retard level is already reached

if IGA_KNK_x - IGA_DEC_KNK + IGA_AD1_AD2_KNK_x **<**
IGA_ADJ_KNK_CTL_DIS - IGA_WOUT_KNK

limphome spark retard level reached: Limitation to max. absolute (refering to TDC) possible ignition retard

then

IGA_KNK_BAS_x = IGA_ADJ_KNK_CTL_DIS - IGA_AD1_AD2_KNK_x – IGA_WOUT_KNK

IGA_ADJ_KNK_x = IGA_ADJ_KNK_CTL_DIS – IGA_WOUT_KNK

limphome spark retard level not yet reached

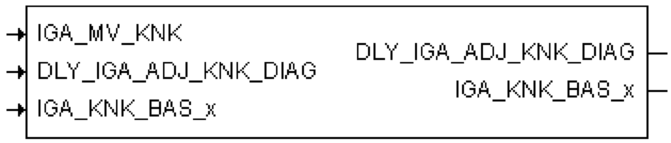
else

IGA_KNK_BAS_x = IGA_KNK_x - IGA_DEC_KNK

IGA_ADJ_KNK_x = IGA_KNK_BAS_x + IGA_AD1_AD2_KNK_x

end if

6.23.3.2.5 Control Exursion limit



KNK_CTL_CTL_EX_LIM

FUNCTION DESCRIPTION:

General information:

The control excursion limit is used to limit the difference of the ignition adjustment angle through knock- control of the single cylinders to a certain value. All 720 °CRK the arithmetic mean value of the cylinder ignition adjustment angles IGA_KNK_BAS_x is calculated:
All 720/Cyl. °CRK a test will be done, to see whether the ignition adjustment angle of the just calculated cylinder is more retarded or advanced than
|IGA_MV_KNK - C_IGA_ DIF_MIN_MAX_KNK|. In this case the knock control adjustment angle is limited to the control excursion limit.

Formula section:

Case Differentiation:

if IGA_KNK_BAS_x > IGA_MV_KNK + C_IGA_DIF_MIN_MAX_KNK

advance limitation:

then IGA_KNK_BAS_x = IGA_MV_KNK + C_IGA_DIF_MIN_MAX_KNK

else

if IGA_KNK_BAS_x < IGA_MV_KNK - C_IGA_DIF_MIN_MAX_KNK

retard limitation:

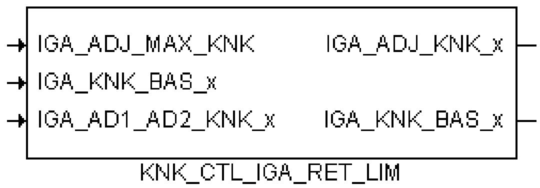
then IGA_KNK_BAS_x = IGA_MV_KNK - C_IGA_DIF_MIN_MAX_KNK

endif

end if

N.B.: x represents current CYL_ID_KNK

6.23.3.2.6 Limitation of total ignition retard



Formula section:

check if limphome spark retard level is already reached

if IGA_KNK_BAS_x + IGA_AD1_AD2_KNK_x **<** IGA_ADJ_MAX_KNK - IGA_WOUT_KNK

limphome spark retard level reached: Limitation to max. absolute (refering to TDC) possible ignition retard

then

IGA_KNK_BAS_x = IGA_ADJ_MAX_KNK – IGA_AD1_AD2_KNK_x – IGA_WOUT_KNK

IGA_ADJ_KNK_x = IGA_ADJ_MAX_KNK – IGA_WOUT_KNK

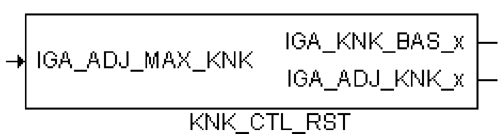
else

IGA_ADJ_KNK_x = IGA_KNK_BAS_x + IGA_AD1_AD2_KNK_x

end if

N.B.: x represents current CYL_ID_KNK

6.23.3.2.7 Reset of Knock control



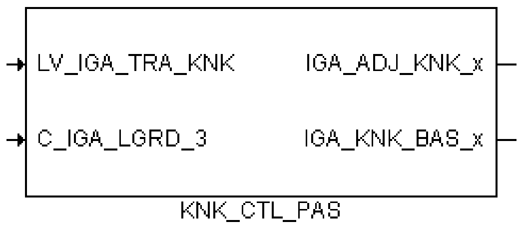
Formula section:

IGA_KNK_BAS_x = 0°CRK

IGA_ADJ_KNK_x = IGA_ADJ_MAX_KNK - IGA_WOUT_KNK

N.B.: x represents current CYL_ID_KNK

6.23.3.3 Knock control passiv (Reset after abandoning the Knock-Operating-Limit)



FUNCTION DESCRIPTION:

General information:

If the operating limit falls below (LV_KNK_CTL_ENA = 0), then the sum of the ignition adjustment angle IGA_KNK_BAS_x + IGA_AD1_KNK_x + IGA_AD2_KNK is re-set to 0, with the change limitation C_IGA_LGRD_3 - however referring to 720° CRK -, in order to reach the map ignition timing.
The adapted ignition adjustment angle itself IGA_AD1_KNK_x and IGA_AD2_KNK remains unchanged, so that this can be used again after crossing the operating limit.

Formula section:

if (LV_IGA_TRA_KNK = 0)

then

IGA_ADJ_KNK_x = IGA_ADJ_KNK_X$_{(N-1)}$ + C_IGA_LGRD_3

IGA_KNK_BAS_x = 0

else

IGA_ADJ_KNK_x = IGA_ADJ_KNK_X$_{(N-1)}$ (remains unchanged)

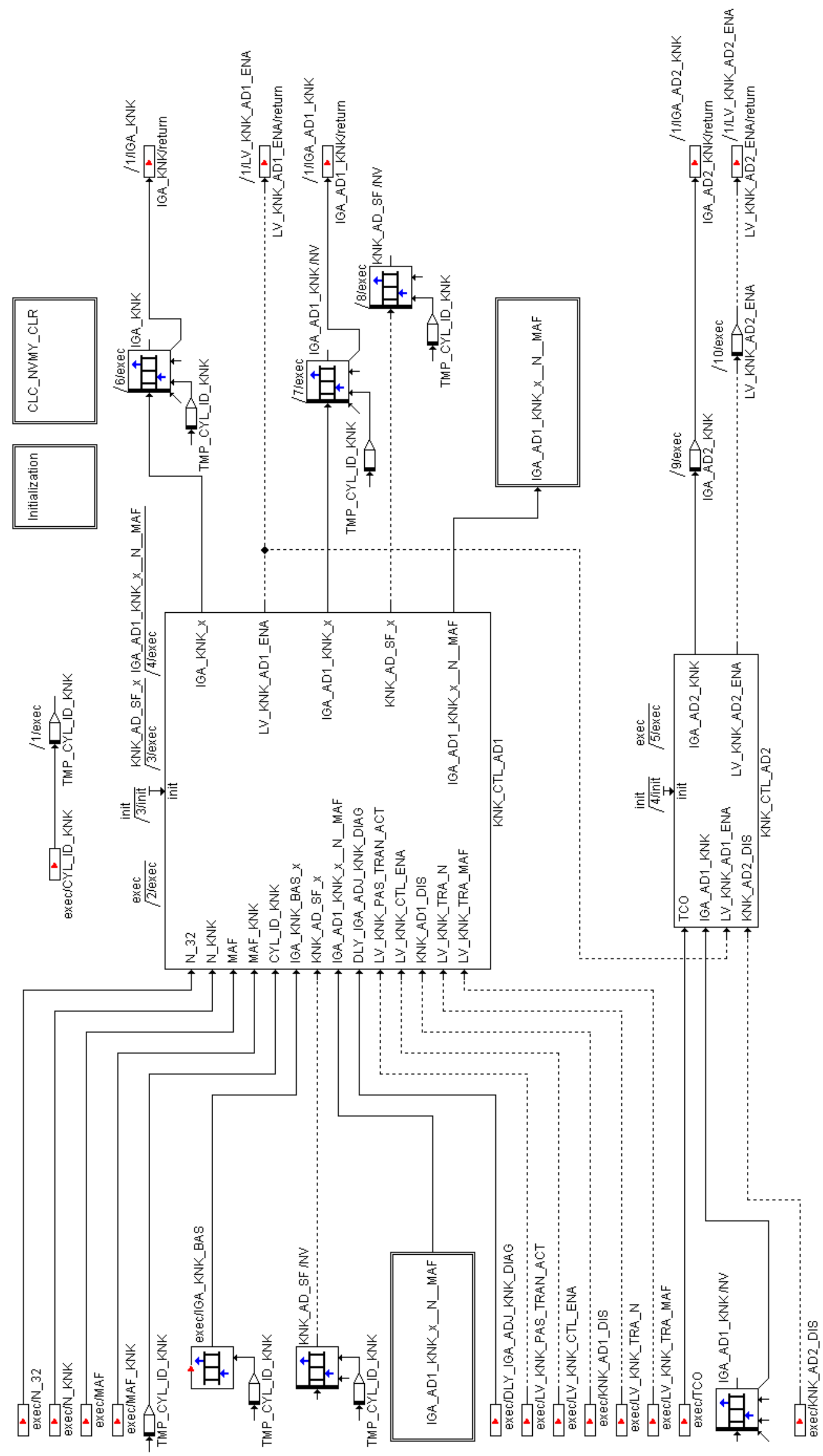
IGA_KNK_BAS_x = 0

end if

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_IGA_DEC_KNK_1__N	3	1...7FH	0.4...47.6	0.38	°CRK /720°CRK
LDPM_N_4_9	3	0...FFH	0...8160	32	rpm
Spark retard at recognised knocking, Intensity 1					
ID_IGA_DEC_KNK_2__N	3	1...7FH	0.4...47.6	0.38	°CRK /720°CRK
LDPM_N_4_9	3	0...FFH	0...8160	32	rpm
Spark retard at recognised knocking, Intensity 2					
IP_IGA_BAS_MAX_KNK__N__MAF	8*8	0...FFH	-35.625 ... 60	0.375	°CRK
LDPM_N_IGA_MAX_KNK	8	0...FFH	0...8160	32	rpm
LDPM_MAF_IGA_MAX_KNK	8	0...FFH	0...1389	5.44	mg/stk
Maximum value for spark retard					
IP_IGA_OFS_MAX_KNK__N__MAF	8*8	0...FFH	-48 ... 47.625	0.375	°CRK
LDPM_N_IGA_MAX_KNK	8	0...FFH	0...8160	32	rpm
LDPM_MAF_IGA_MAX_KNK	8	0...FFH	0...1389	5.44	mg/stk
Maximum value for spark retard					
IP_FAC_IGA_MAX_KNK__TCO__TIA	4*4	0...FFH	0...0.996	0.0039	-
LDP_TCO__FAC_IGA_TCO_KNK	4	0...FEH	-48...142.5	0.75	°C
LDP_TIA__FAC_IGA_TIA_KNK	4	0...FEH	-48...142.5	0.75	°C
Weighting factor for the max. spark retard against coolant					
IP_FAC_AD_KNK_TCO__TCO	2	0...FFH	0...0.996	0.0039	-
LDP_TCO__FAC_AD_KNK_TCO	2	0...FEH	-48...142.5	0.75	°C
Weighting factor for adaption circuit 1 and 2 via coolant temperature					
IP_IGA_INC_KNK__N	4	1...7FFFH	0.001...47.999	0.0015	°CRK /720 CRK
LDPM_N_FIL_FRQ_KNK	4	0...FFH	0...8160	32	rpm
Re-set Increment KNK					
C_IGA_DIF_MIN_MAX_KNK	1	0...80H	0...48	0.375	°CRK
Control excursion limit valid for spark advance and spark retard					
IP_IGA_KNK_CTL_DIS__N__MAF	8*8	0...FFH	-35.625 ... 60	0.375	°CRK
LDPM_N_IGA_MAX_KNK	8	0...FFH	0...8160	32	rpm
LDPM_MAF_IGA_MAX_KNK	8	0...FFH	0...1389	5.44	mg/stk
Spark retard during limp home operation (KNK_CTL_DIS = 1)					

6.23.4 Knock Adaptation

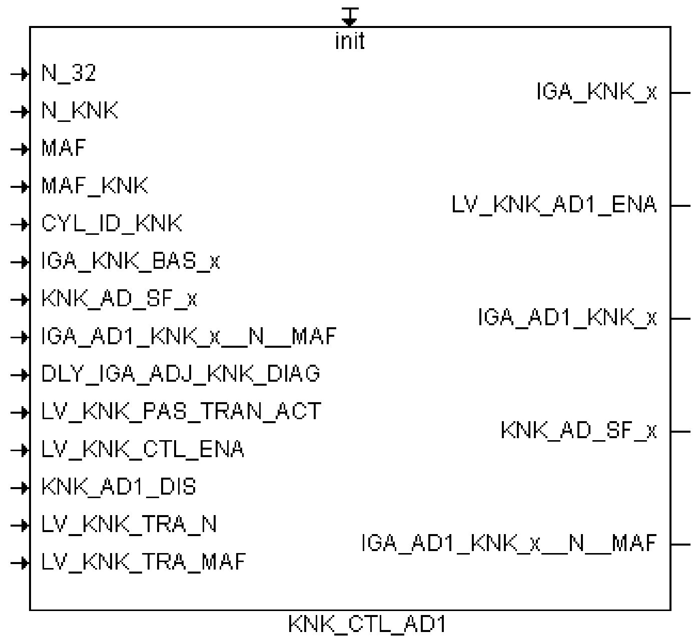


FUNCTION DESCRIPTION:

The knock adaptation consists of two so-called adaptation-circuits. A cylinder individual knock adaptation (adaptation circuit 1), which depends on load and speed, should compensate such as the compression ratio difference between each cylinder. The input for the adaptation circuit 1 is the spark retard due to knocking combustions.

The second part of the knock adaptation (adaptation circuit 2) offsets the ignition of the total speed and load range (in the area where the knock control is active) as well as each cylinder. This adaptation should compensate possible fuel grade differences. The input of the adaptation circuit 2 is the average mean value of all cylinders of the adaptation circuit 1.

6.23.4.1 Adaptation Circuit 1



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_AD1_KNK_x	V/O	0...80H	-48...0	0.375	°CRK
Actual adaptation value for Cyl. x (Adaptation circuit 1)					
Stored adaptation table (circuit 1) value for Cyl. x, engine speed y, and load (MAF) z					
IGA_AD1_KNK_x__N__MAF [8] [6]	V/O/S	0...80H	-48...0	0.375	°CRK
Adaptation table for Cyl. x (Adaptation circuit 1)					
IGA_KNK_x	V/O	0...80H	-48...0	0.375	°CRK
Spark retard due to knocking combustion					
LV_KNK_AD1_ENA	V/O	0...1H	0...1	1	-
Knock adaptation circuit 1 enabled					
KNK_AD_SF_x	-	0...1H	0...1	1	-
Flag for transition from passive to active					

Input data:

N_KNK	MAF_KNK	IGA_KNK_BAS_x	LV_KNK_TRA_MAF
LV_KNK_TRA_N	LV_KNK_CTL_ENA	LV_KNK	ID_IGA_DEC_KNK_1__N
DLY_IGA_ADJ_KNK_DIAG	LV_KNK_PAS_TRAN_ACT	MAF	N_32
KNK_AD1_DIS	IGA_AD_x__N__MAF	CYL_ID_KNK	

FUNCTION DESCRIPTION:

General information:

A table IGA_AD_x__N__MAF exists for each cylinder. These tables contain the cylinderindividual ignition adjustment angles due to knocking combustion, which have been determined by the knock control at the respective operating points. The size of this table is defined by the calibration table ID_IGA_AD1_KNK_SIZE__N__MAF.

Storage of Adaptation-Circuit 1 in the NVMY after the power latch of the ECU

After the power latch mode of the ECU all adapted spark corrections in the cylinder individual adaptation maps IGA_AD_x are stored in the non-volatile memory and read back at the next KEY ON.
The ignition retardation IGA_AD1_KNK_x of the adaptation circuit 1 is increased by C_IGA_AD1_INC_KNK during the post operating phase before storing the values. With this function it is possible to reduce large adapted values without fulfilling the adaptation conditions.
The adaptation tables may be initiated in new condition or through maintenance service with 0 °CRK.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: -

Initialisation: IGA_AD1_KNK_x = 0 at reset and engine running to engine stop

Update Rate: every segment

Formula section:

Activation / Deactivation:

check conditions for adaption 1

if KNK_AD1_DIS = 0
    and C_SWI_AD_KNK = 0
    and LV_KNK_CTL_ENA = 1
    and LV_KNK_TRA_MAF= 0
    and LV_KNK_TRA_N = 0
    and DLY_IGA_ADJ_KNK_DIAG = 0
    and MAF >= ID_MAF_MIN_AD1_KNK
] **and** C_TCO_AD1_MIN_KNK < TCO < C_TCO_AD1_MAX_KNK

Adaptation 1 enabled

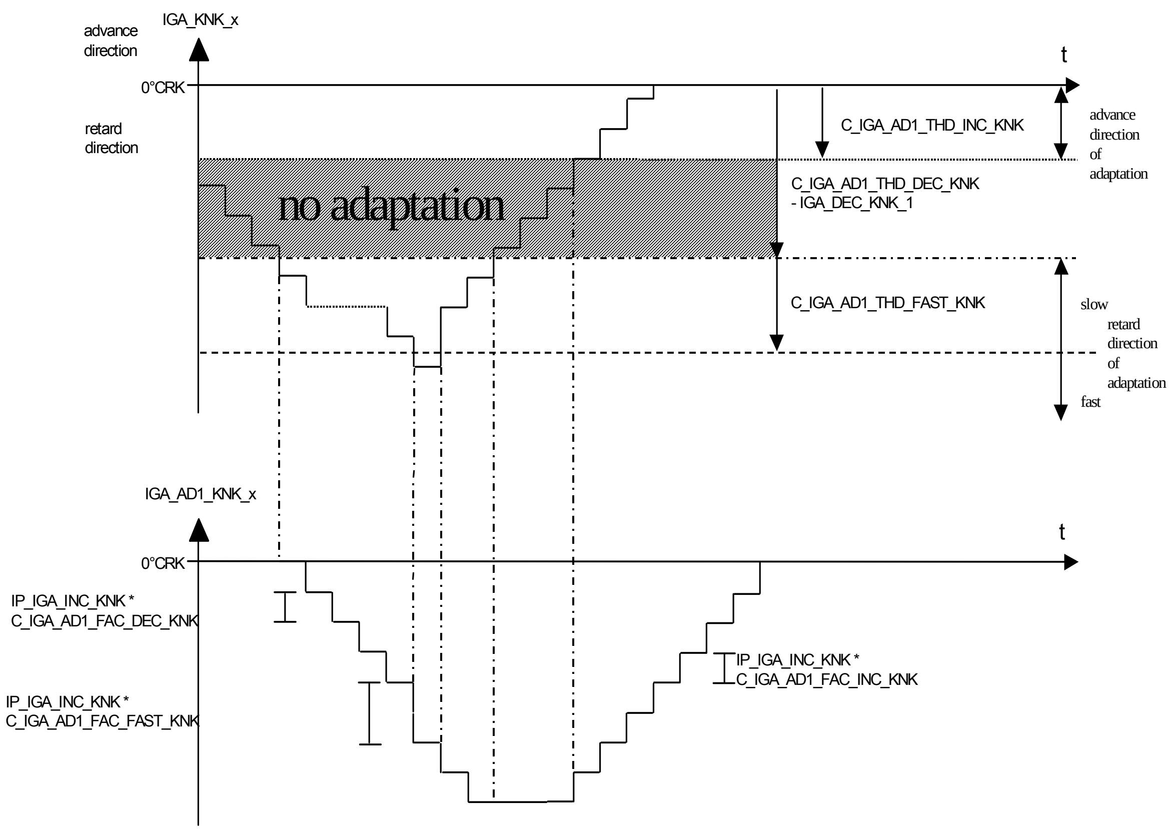
then LV_KNK_AD1_ENA = 1

Adaptation 1 disabled

else LV_KNK_AD1_ENA = 0
Adaptation value IGA_AD1_KNK_x restored from cylinder individual adaptation maps with change limitation:

IGA_AD1_KNK_x →C_IGA_AD1_LGRD_POS/NEG_KNK→ID_IGA_AD1_KNK_x__N__MAF

end if



Adaptation algorithm:

adaptation 1 active
if LV_KNK_AD1_ENA = 1
then
check adaptation direction range
if IGA_KNK_BAS_x > C_IGA_AD1_THD_INC_KNK

adaptation in advance direction
then $IGA\_AD1\_KNK\_x_i$ = $IGA\_AD1\_KNK\_x_{i-1}$ + IP_IGA_INC_KNK * C_IGA_AD1_FAC_INC_KNK

check if knock control adjustment outside dead zone
elseif IGA_KNK_BAS_x < C_IGA_AD1_THD_DEC_KNK - ID_IGA_DEC_KNK_1__N
and
C_IGA_AD1_THD_INC_KNK > C_IGA_AD1_THD_DEC_KNK - ID_IGA_DEC_KNK_1__N
adaptation in retard direction
then
check of adaptation speed range
if IGA_KNK_BAS_x < (C_IGA_AD1_THD_DEC_KNK – ID_IGA_DEC_KNK_1__N - C_IGA_AD1_THD_FAST_KNK)
fast adaptation
then $IGA\_AD1\_KNK\_x_i$ = $IGA\_AD1\_KNK\_x_{i-1}$ - IP_IGA_INC_KNK * C_IGA_AD1_FAC_FAST_KNK

IGA_KNK_x = IGA_KNK_BAS_x + IP_IGA_INC_KNK * C_IGA_AD1_FAC_FAST_KNK
slow adaptation
else $IGA\_AD1\_KNK\_x_i$ = $IGA\_AD1\_KNK\_x_{i-1}$ - IP_IGA_INC_KNK*C_IGA_AD1_FAC_DEC_KNK

IGA_KNK_x = IGA_KNK_BAS_x + IP_IGA_INC_KNK*C_IGA_AD1_FAC_DEC_KNK
end if

knock control adjustment in dead zone
else IGA_AD1_KNK_x remains unchanged

endif
adaptation 1 inactive
else IGA_KNK_x = IGA_KNK_BAS_x

endif

In case of adaptation in retard direction the knock control value IGA_KNK_x is calculated by adding IP_IGA_INC_KNK * (C_IGA_AD1_FAC_DEC_KNK or C_IGA_AD1_FAC_FAST_KNK) to IGA_KNK_BAS_x in advance direction. Due to this the total spark retard IGA_ADJ_KNK_x stays constant and will not be adjusted additional in retard direction by the adaptation circuit 1.

In case of no knock detection (LV_KNK = 0) this behaviour prevents also to adjust the adaptation value in retard direction, while the knock control value runs already in advance direction.

- **Treatment of IGA_AD1_KNK in case of changing operating point :**

if IGA_AD1_KNK_x_{n-1} < IGA_AD1_KNK_x_n

then IGA_AD1_KNK_x_{n-1} →C_IGA_AD1_LGRD_POS_KNK→IGA_AD1_KNK_x_n

else IGA_AD1_KNK_x_{n-1} →C_IGA_AD1_LGRD_NEG_KNK→IGA_AD1_KNK_x_n

endif

- **Treatment during transition from knock control passive to active (LV_KNK_PAS_TRAN_ACT = 1):**

The adaptation value IGA_AD1_KNK_x is restored from the cylinder individual adaptation maps ID_IGA_AD1_KNK_x without change limitation

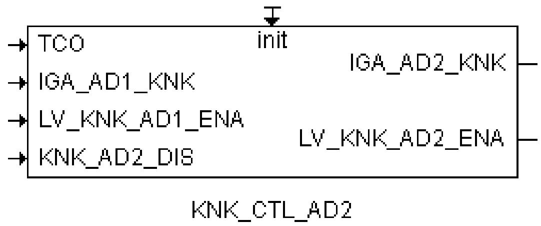
Classification of Tables for Ignition Timing Pre-control for Adaptation Circuit 1:

Pre-control ignition timing for Cylinder x: ID_IGA_AD1_KNK_x__N__MAF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_MAF_MIN_AD_KNK__N	8	0....FFH	0...1389	5.447	mg/Stroke
LDPM_N_IGA_AD	8	0....FFH	0...8160	32	rpm
Air-mass Threshold for Knock Adaptation					
ID_IGA_AD1_KNK_SIZE__N__MAF	8*10	0...80H	-48...0	0.375	°CRK
LDPM_N_IGA_AD	8	0....FFH	0...8160	32	rpm
LDPM_MAF_IGA_AD	10	0....FFH	0...1389	5.447	mg/Stroke
Table for the definition of the size and the LDP division of adaptation table circuit 1					
C_IGA_AD1_LGRD_POS_KNK	1	1...80H	0.0937...12	0.0937	°CRK/720
Change limitation in advance direction for adaptation circuit 1 while crossing from one operating point to another					
C_IGA_AD1_LGRD_NEG_KNK	1	1...80H	0.0937...12	0.0937	°CRK/720
Change limitation in retard direction for adaptation circuit 1 while crossing from one operating point to another					
C_IGA_AD1_FAC_INC_KNK	1	0....FF	0...0.996	0.0039	-
Factor to control the speed of knock adaptation circuit 1 in advance direction					
C_IGA_AD1_FAC_DEC_KNK	1	0....FF	0...0.996	0.0039	-
Factor to control the speed of knock adaptation circuit 1 in retard direction					
C_IGA_AD1_INC_KNK	1	0...80H	0...48	0.375	°CRK
Increment of adapted retardation values during post operating phase					
C_IGA_AD1_FAC_FAST_KNK	1	0...FFH	0...3.984375	0.0156	°CRK
factor for fast knock adaptation					
C_IGA_AD1_THD_DEC_KNK	1	E0...20H	-12...12	0.375	°CRK
C_IGA_AD1_THD_FAST_KNK	1	1...40H	0.375...24	0.375	°CRK
C_IGA_AD1_THD_INC_KNK	1	40...7FH	-24…-0.375	0.375	°CRK
C_TCO_AD1_MIN_KNK	1	0...FEH	-48...142.5	0.75	°C
minimum TCO threshold for knock adaptation circuit 1					
C_TCO_AD1_MAX_KNK	1	0...FEH	-48...142.5	0.75	°C
maximum TCO threshold for knock adaptation circuit 1					
C_SWI_AD_KNK	1	0...1H	0...1	1	-
calculation of adaptation circuit 1 and 2 enabled					

6.23.4.2 Adaptation Circuit 2



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_AD2_KNK	V/O/S	0...80H	-48...0	0.375	°CRK
Adaptation spark retard due to adaptation circuit 2					
LV_KNK_AD2_ENA	V	0...1H	0...1	1	-
Knock adaptation circuit 2 enabled					
IGA_MV_AD1_KNK	-	0...80H	-48...0	0.375	°CRK
meanvalue of actual retard adjustment by adaptation circuit 1					

Input data:

LV_KNK_AD1_ENA	IGA_AD1_KNK_x	TCO	KNK_AD2_DIS

FUNCTION DESCRIPTION:

General information:

At high knock intensities the ignition timing pre-control value IGA_AD2_KNK is moved slowly in retarded direction. A movement in advanced direction follows at longer knock-free operation, whereby the advanced movement is limited to the basic ignition map.

Storage of Adaptation-Circuit 2 in the NVMY after the power latch of the ECU
In new condition or by service maintenance IGA_AD2_KNK is initialised with 0.

For the adaptation Circuit 2 all 720 °CRK an arithmetic mean value is calculated using the cylinder-individual ignition adjustment angle of the adaptation circuit 1.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: -

Initialisation: IGA_MV_AD1_KNK = 0 at reset and engine running to engine stop

Update Rate: every 720°CRK

Formula section:

$$IGA\_MV\_AD1\_KNK = \frac{\sum_{i=1}^{NC\_CYL\_NR} (IGA\_AD1\_KNK\_x)}{NC\_CYL\_NR}$$

if C_TCO_AD2_MIN_KNK < TCO < C_TCO_AD2_MAX_KNK

then LV_TCO_AD2_KNK = 1

else LV_TCO_AD2_KNK = 0

end if

if [(KNK_AD2_DIS=0) AND (LV_KNK_AD1_ENA=1) AND (LV_TCO_AD2_KNK=1)]

Adjustment enabled

then LV_KNK_AD2_ENA = 1

if ((IGA_MV_AD1_KNK ≤ |C_IGA_AD2_THD_INC_KNK|)

and (IGA_MV_AD1_KNK < |C_IGA_AD2_THD_DEC_KNK|)]

No Adaptation

then IGA_AD2_KNK remains unchanged

else Adjustment of adaptation value is possible

see below 'Advance / Retard Adjustment'

end if

else Adjustment disabled

LV_KNK_AD2_ENA = 0

end if

<u>Advance Adjustment</u>:

If the mean value of the ignition adjustment angle IGA_MV_AD1_KNK for C_CYCNR_AD2_INC_KNK combustion cycles (720° CRK) > |C_IGA_AD2_THD_INC_KNK|, the ignition adjustment angle IGA_AD2_KNK is moved towards advance by 0,375 °CRK. If the condition for advance adjustment is fulfilled for C_CYCNR_AD2_INC_KNK combustion cycles IGA_AD2_KNK is further adjusted by an additional 0,375 °CRK step until the basic ignition angle is reached.

if **(IGA_MV_AD1_KNK > |C_IGA_AD2_THD_INC_KNK|) for C_CYCNR_AD2_INC_KNK**
IGA_AD2_KNK = IGA_AD2_KNK + 0,375 °CRK
end if

<u>General limitation of advance adjustment:</u>

if **(IGA_AD2_KNK > 0°CRK)**
IGA_AD2_KNK = 0°CRK
end if

Retard adjustment:

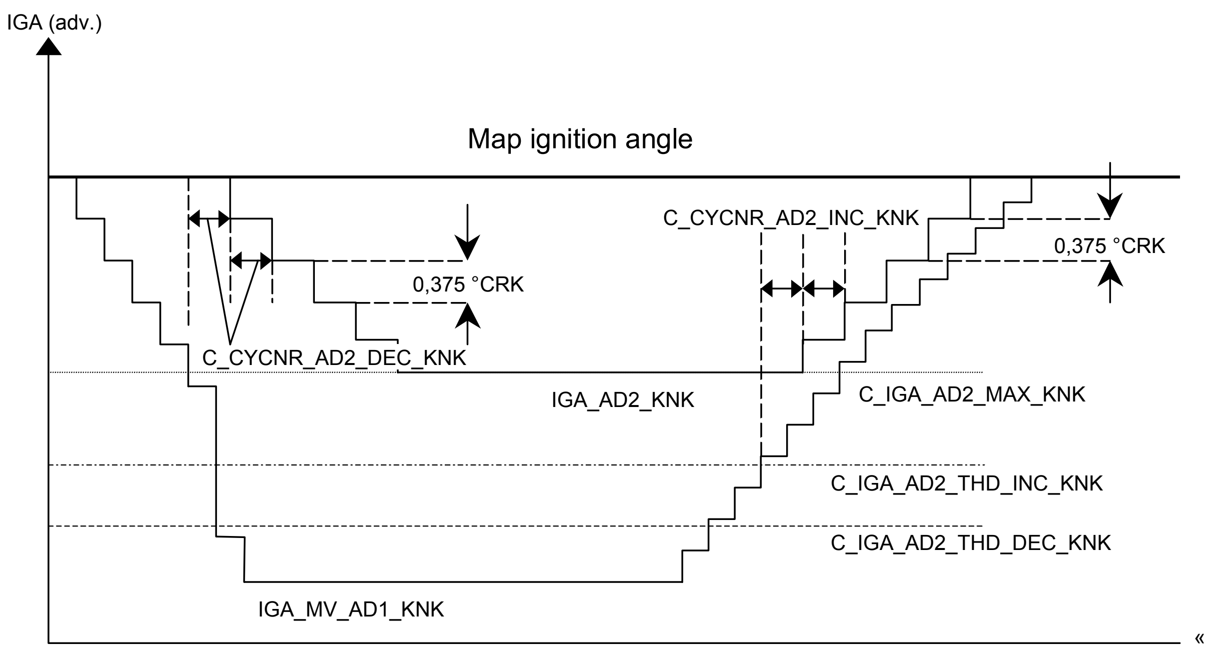
If the mean value of the ignition adjustment angle IGA_AD1_MV_KNK for C_CYCNR_AD2_DEC_KNK comb. cycles (720 °CRK) < │ C_IGA_AD2_THD_DEC_KNK │, then IGA_AD2_KNK is adjusted towards retard by 0,375 °CRK.
If the condition for retard adjustment is fulfilled for C_CYCNR_AD2_DEC_KNK combustion cycles, IGA_AD2_KNK is adjusted by an additional 0,375 °CRK step until C_IGA_AD2_MAX_KNK is reached.

if **(IGA_MV_AD1_KNK < |C_IGA_AD2_THD_DEC_KNK|) for C_CYCNR_AD2_DEC_KNK**
IGA_AD2_KNK = IGA_AD2_KNK - 0,375 °CRK
end if

General limitation of retard adjustment:

if **(IGA_AD2_KNK < C_IGA_AD2_MAX_KNK)**
IGA_AD2_KNK = C_IGA_AD2_MAX_KNK
end if

Description:

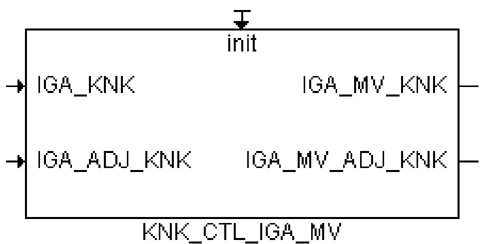


Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_IGA_AD2_THD_INC_KNK	1	0...80H	-48...0	0.375	°CRK
Threshold for adjustment of adaptive ignition angle into advance direction					
C_IGA_AD2_THD_DEC_KNK	1	0...80H	-48...0	0.375	°CRK
Threshold for adjustment of adaptive ignition angle into retard direction					
C_CYCNR_AD2_INC_KNK	1	0...FFFFH	0...65535	1	720°CRK
Cycle counter for adjustment into advance direction					
C_CYCNR_AD2_DEC_KNK	1	0...FFFFH	0...65535	1	720°CRK
Cycle counter for adjustment into retard direction					
C_IGA_AD2_MAX_KNK	1	0...80H	-48...0	0.375	°CRK
maximum threshold of IGA_AD2_KNK in retard direction					
C_TCO_AD2_MIN_KNK	1	0...FEH	-48...142.5	0.75	°C
minimum TCO threshold for knock adaptation circuit 2					
C_TCO_AD2_MAX_KNK	1	0...FEH	-48...142.5	0.75	°C
maximum TCO threshold for knock adaptation circuit 2					

6.23.5 Post-processing

6.23.5.1 Meanvalues of knock-control ignition adjustment



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_MV_KNK	V	0...80H	-48...0	0.375	°CRK
meanvalue of actual retard adjustment without adaptation values					
IGA_MV_ADJ_KNK	V/O	0...80H	-48...0	0.375	°CRK
meanvalue of actual retard adjustment with adaptation values					

Input data:

IGA_KNK_x	IGA_ADJ_KNK_x		

Application conditions:

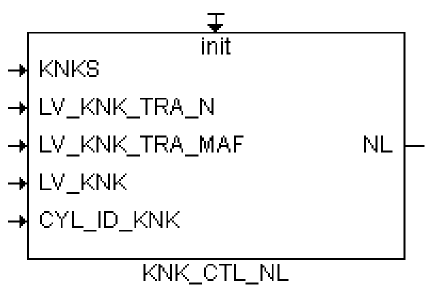
Activation:	at every engine state except engine stop
Deactivation:	-
Initialisation:	IGA_MV_KNK = IGA_MV_ADJ_KNK = 0
	at reset and engine running to engine stop
Update Rate:	every 720°CRK

Formula section:

$$\mathbf{IGA\_MV\_KNK} = \frac{\sum_{i=1}^{NC\_CYL\_NR} (\mathbf{IGA\_KNK\_x})}{\mathbf{NC\_CYL\_NR}}$$

$$\mathbf{IGA\_MV\_ADJ\_KNK} = \frac{\sum_{i=1}^{NC\_CYL\_NR} (\mathbf{IGA\_ADJ\_KNK\_x})}{\mathbf{NC\_CYL\_NR}}$$

6.23.5.2 Noise Value



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
NL_x	V/O	0...FFFFH	0...5	7.6*10-5	V
Floating mean value of the noise signal					

Input data:

KNKSx	LV_ KNK_ TRA_N	LV_KNK_TRA_MAF	LV_KNK
CYL_ID_KNK	C_KNK_THD_MAX		

FUNCTION DESCRIPTION:

The noise value is cylinder-individually calculated as sliding average from the knock values. The averaging constant is either C_NL_CRLC, if no knock control transient function is active (LV_KNK_TRA_N = 0 and LV_KNK_TRA_MAF = 0) or C_NL_CRLC_TRA, when a knock control transient function is active (LV_KNK_TRA_N = 1 or LV_KNK_TRA_MAF = 1).

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: NL_x = C_KNK_THD_MAX at reset or engine running to engine stop

Update Rate: every segment

Formula section:

if **(LV_KNK_TRA_N or LV_KNK_TRA_MAF)**
NL_x = NL_x + (KNKSx – NL_x) * C_NL_CRLC_TRA
else
NL_x = NL_x + (KNKSx – NL_x) * C_NL_CRLC
end if

NL_x shows always the measured noise value from the ATM40 device. The noise value is calculated always (independant from detection of knocking combustion).

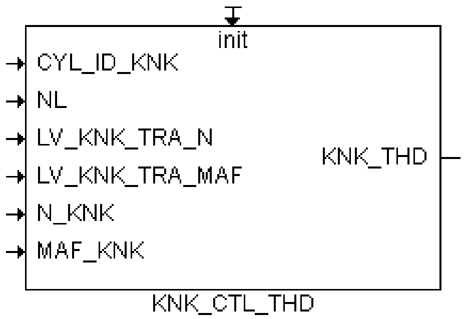
If knocking combustions are detected (LV_KNK = 1) after knock-free operation, the first and the second noise value isn`t used to calculate the noise level (NL_x remains unchanged), the third and the following are taken into consideration for calculation of noise level.

N.B.: x represents current CYL_ID_KNK

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_NL_CRLC	1	0...FFH	0...1	0.004	-
Averaging Constant for Noise Value Calculation					
C_NL_CRLC_TRA	1	0...FFH	0...1	0.004	-
Averaging Constant for Noise Value Calculation for Transient Case					

6.23.5.3 Relative and Absolute Knock Threshold



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
KNK_THD_x	V/O	0...FFH	0...5	0.02	V
Knock Threshold cylinder x					

Input data:

NL_x	CYL_ID_KNK	LV_KNK_TRA_MAF	LV_KNK_TRA_N
N_KNK	MAF_KNK		

FUNCTION DESCRIPTION:

General information:

The knock threshold (KNK_THD_x) is the limit for the knock values (KNKSx), which are used for knock decision. If the knock values are greater or equal to the knock threshold, then knocking is recognised (LV_KNK = 1) and an ignition timing retardation follows. If the knock values are smaller, knocking is not detected (LV_KNK = 0) and the relevant KNK-ignition-angle is reset (→ Knock-free Operation after Knocking).

The knock threshold is calculated for each cylinder separately and depends on the cylinder-individual noise value. Calculation of the knock threshold occurs with help from the cylinder-individual knock factors ID_FAC_THD_KNK_x and knock summand ID_ADD_KNK__N.

If the noise value is above the minimum threshold C_NL_MIN, the knock threshold is calculated on the basis of the cylinder-individually noise value NL_x. If the noise value is below C_NL_MIN this value is used as noise value for calculation of the knock threshold. If the knock threshold is above C_THD_MAX_KNK this value is used as knock threshold (Limitation).

N.B.: The actual knock value $KNKSx_n$ is not used in the calculation of the actual knock threshold $KNK\_THD\_x_n$ but in the calculation of the next threshold $KNK\_THD\_x_{n+1}$. So the decision knocking combustion yes/no is done with using the knock threshold $KNKSx_{n-1}$.

Application conditions:

Activation: at every engine state except engine stop

Deactivation: at engine stop

Initialization: KNK_THD_x = 5V at reset and engine running to engine stop

Update Rate: every segment

Formula section:

The lower absolute threshold is one value for all cylinders and is represented by the constant C_NL_MIN.

```
If    NL_x &lt; C_NL_MIN
      NL_LIM = C_NL_MIN
else
      NL_LIM = NL_x
end if
```

```
if    (LV_KNK_TRA_N = 0 and LV_KNK_TRA_MAF = 0)
      KNK_THD_x = ( NL_LIM + ID_KNK_THD_ADD ) * ID_FAC_THD_KNK_x
else
      KNK_THD_x = ( NL_LIM + ID_KNK_THD_ADD ) * ID_FAC_THD_KNK_x *
                        C_FAC_TRA_KNK
end if
```

The knock threshold is generally limited to C_KNK_THD_MAX:

```
If  (KNK_THD_x > C_KNK_THD_MAX)
    KNK_THD_x = C_KNK_THD_MAX
end if
```

N.B.: x represents current CYL_ID_KNK

(*) The correction factor **C_FAC_TRA_KNK** is part of the KNK-load dynamics function and is effective after recognised load dynamics **ID_CYCNR_FAC_TRA_KNK__N** cycles.

The knock factor is the multiplicative part used for calculation of the knock threshold from the noise value.
The knock summand is an additive correction of the knock noise value. Thereby the knock threshold can be moved, dependent on the speed, in a positive or negative direction.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_THD_KNK_x__N__MAF	16*6	0...FFH	0...7.97	0.031	-
LDPM_N_2_9	16	0...FFH	0...8160	32	rpm
LDPM_MAF_1_9	6	0...FFH	0...1389	5.44	mg/stk
Knock Factor Table Cyl. X					
IP_KNK_THD_ADD__N	16	0...FFH	-2.56...2.54	0.02	V
LDPM_N_2_9	16	0...FFH	0...8160	32	rpm
Knock summand					
C_KNK_THD_MAX	1	0...FFH	0...5	0.02	V
Maximum knock threshold					
C_FAC_TRA_KNK	1	20...FFH	1...7.97	0.031	-
correction factor for load dynamic active					
C_NL_MIN	1	0...FFH	0...5	0.02	V
Minimum noise value					

7 Injection

Table of Contents

Index

G

H

I

M

O

P

V

7.1 Final Injection Timing

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_1_HOM[NC_CYL_NR]	O/V	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, homogeneous mode, first pulse (intermediate value)					
TI_MIN	O/V	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time limitation					
LV_TI_1_HOM_MIN	O/V	0 ... 1H	0 ... 1	1	-
Flag to indicate that minimum injection time is reached					
LV_AUTH_TI_MIN_AFL	O/V	0..1H	0...1	1	-
Logical variable to request homogeneous mode due to injection time limitation					
TI_1_HOM_CLC[NC_CYL_NR]	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, homogeneous mode, first pulse (intermediate value)					
TI_1_HOM_CLC_MIN	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time					
TI_1_HOM_CLC_MAX	O/V	0 ... FFFFH	0 ... 262.14	0.004	ms
Maximum injection time					
TI_1_HOM_CLC_MIN_TMP	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time for temporary calculations					
TI_1_HOM_CLC_MAX_TMP	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Maximum injection time for temporary calculations					
CTR_UPD_NR_CYL	-	0 ... 8H	0 ... 8	1	-
Counts the number of updated cylinder individual injection times					
T_AUTH_TI_MIN_AFL	-	0...FFFFH	1...655.36	0.01	s
Delay time after AFL mode was inhibited due to injection time limitation					

Input data:

TI_TUN_IV[NC_CYL_NR]	MFF_SP_1_HOM[NC_CYL_NR]	TI_FAC[NC_CYL_NR]	NC_IN_REF
TI_ADD[NC_CYL_NR]	TI_ADD_DLY_1_HOM	FAC_TI_1_PRS_HOM	LV_AFL_CLC
TI_TUN_ADD_IV[NC_CYL_NR]	NC_STATE_CLC_RED_ACT	STATE_CLC_RED	VB
CTR_CYL_NR_ST_CLC_INJR	CTR_CYL_NR_STOP_CLC_INJR	STATE_CMB_CTL	

Remark: The input value TI_FAC_x and TI_ADD_x are the output value of the module "Coordination of Injection Time Correction Factors for Cylinder Balancing ". If the Cylinder Balancing Function is not used than TI_FAC_x and TI_ADD_x have to be initalized in an "Initialization Module" as follows:

TI_FAC _x =1

TI_ADD _x = 0

FUNCTION DESCRIPTION:

General information:

This module calculates the cylinder individual injection times, which shall be applied at the injectors.

7.1.1 Calculation of TI_1_HOM_x

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: at reset: TI_1_HOM_CLC_MIN_TMP = 262.14 ms
TI_1_HOM_CLC_MAX_TMP = 0 ms

Recurrence: LV_ST_END = 0: 10 ms
LV_ST_END = 1: segment synchronous

Formula section:

7.1.1.1 Calculation of the minimal applied injection time

In order to avoid linearity problems, the injection time needs minimum limitation.

TI_MIN = IP_TI_MIN (Input: VB)

7.1.1.2 Calculation of the cylinder individual injection times

TI_1_HOM_CLC_x = MFF_SP_1_HOM_x * C_FAC_MFF_TI_STND_k * FAC_TI_1_PRS_HOM * TI_FAC_x * TI_TUN_IV_x + TI_TUN_ADD_IV_x + TI_ADD_DLY_1_HOM + TI_ADD_x

Note:

The pattern NC_IN_REF (project specific) is used for allocation of physical cylinders to intake bank 1 and intake bank 2.

7.1.1.3 Calculation of minimum and maximum injection times

For each calculation run the minimum and maximum injection time will be calculated:

(1) FOR x **=** CTR_CYL_NR_ST_CLC_INJR **TO** CTR_CYL_NR_STOP_CLC_INJR

CTR_UPD_NR_CYL = CTR_UPD_NR_CYL + 1

TI_1_HOM_CLC_MIN_TMP = MIN (TI_1_HOM_CLC_MIN_TMP, TI_1_HOM_CLC[x])

TI_1_HOM_CLC_MAX_TMP = MAX (TI_1_HOM_CLC_MAX_TMP, TI_1_HOM_CLC[x])

(1) ENDFOR

If calculation reduction is not active minimum and maximum injection times will be updated immediately.

If calculation reduction is active, the minimum and maximum injection times will be updated only after all cylinders were calculated – then a new cycle starts. If new extreme values are occurring, minimum and maximum injection times will be updated immediately, independent on the calculation reduction state.

(2) IF STATE_CLC_RED & NC_STATE_CLC_RED_ACT = 0
OR
CTR_UPD_NR_CYL >= NC_CYL_NR

(2) THEN

CTR_UPD_NR_CYL = 0

TI_1_HOM_CLC_MIN = TI_1_HOM_CLC_MIN_TMP

TI_1_HOM_CLC_MAX = TI_1_HOM_CLC_MAX_TMP

Set the temporary minimum and maximum to the opposite limits, therefore it is possible to find new minimum and maximum values on next calculation run.

TI_1_HOM_CLC_MIN_TMP = 262.14 ms

TI_1_HOM_CLC_MAX_TMP = 0 ms

(2) ELSE

TI_1_HOM_CLC_MIN **= MIN(**TI_1_HOM_CLC_MIN, TI_1_HOM_CLC_MIN_TMP**)**

TI_1_HOM_CLC_MAX **= MAX(**TI_1_HOM_CLC_MAX, TI_1_HOM_CLC_MAX_TMP**)**

(2) ENDIF

7.1.1.4 Calculation of the final cylinder individual homogeneous injection times

(3) IF

LC_TI_1_HOM_MAN_ACT = 1

(3) THEN

TI_1_HOM_x = C_TI_1_HOM_MAN

LV_TI_1_HOM_MIN = 0

(3) ELSE

(4) IF

TI_1_HOM_CLC_MIN < TI_MIN

(4) THEN

TI_1_HOM_x = TI_1_HOM_CLC_x + (TI_MIN - TI_1_HOM_CLC_MIN)
Note! Adjust each cylinder to the minimum injection time, therefor the cylinder individual time offsets between the cylinders will be observed.

LV_TI_1_HOM_MIN = 1

(4) ELSE

TI_1_HOM_x = TI_1_HOM_CLC_x

LV_TI_1_HOM_MIN = 0

(4) ENDIF

(3) ENDIF

Note: For calibration purposes the injection time TI_1_HOM_x can be set manually to the value C_TI_1_HOM_MAN.

7.1.2 Calculation of LV_AUTH_TI_MIN_AFL

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: -

Recurrence: 10 ms

If homogeneous lean mode is active (STATE_CMB_CTL = 2) and the electrical injection time is smaller than (TI_MIN + C_TI_MIN_OFS), the combustion mode changes to homogeneous stochiometric mode. Switching back to homogeneous lean mode is allowed at the earliest if the delay time C_ T_AUTH_TI_MIN_AFL has elapsed.

(1) IF

LV_AFL_CLC = 0

(1) THEN

LV_AUTH_TI_MIN_AFL = 0

T_AUTH_TI_MIN_AFL = 0

(1) ELSE

(2) IF

LV_AUTH_TI_MIN_AFL = 0

AND

STATE_CMB_CTL = 2 (homogeneous lean mode)

AND

TI_1_HOM_CLC_MIN < (TI_MIN + C_TI_MIN_OFS)

(2) THEN

LV_AUTH_TI_MIN_AFL = 1

T_AUTH_TI_MIN_AFL = C_ T_AUTH_TI_MIN_AFL

(2) ELSE

(3) IF

T_AUTH_TI_MIN_AFL (n) > 0

(3) THEN

T_AUTH_TI_MIN_AFL (n+1) = T_AUTH_TI_MIN_AFL (n) – 0.01 [s]

(3) ELSE

LV_AUTH_TI_MIN_AFL = 0

(3) ENDIF

(2) ENDIF

(1) ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TI_MIN	8	0...FFFFH	0...262.14	0.004	ms
LDPM_VB_1_INJR	8	0 ... FFH	0 ... 26	0.102	V
Minimum injection time limitation					
C_FAC_MFF_TI_STND_k	1	0...FFFFH	0...0.754891288	1.1518902706E-5	ms/mg
Injector characteristic for mass fuel flow to injection time at standard pressure and temperature					
C_TI_1_HOM_MAN	1	0...FFFFH	0...262.14	0.004	ms
Manual setpoint for injection time in homogeneous mode, 1. pulse					
C_TI_MIN_OFS	1	0...FFH	0...1.02	0.004	ms
Offset to TI_MIN, (injection time limitation at AFL mode and stratified mode)					
C_ T_AUTH_TI_MIN_AFL	1	0...FFFFH	0.01...655.36	0.01	s
Delay time after AFL mode was inhibited due to injection time limitation					
LC_TI_1_HOM_MAN_ACT	1	0...1H	0..1	1	-
Manual switch for manual setpoint for injection time in homogeneous mode					

7.2 Transfer to I/O SW

Overview

This module handles the data transfer between the ASW and the I/O software.

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
NR_TRIG_EOI_LIM_AV	V/O	0 ... 7 H	0 ... 7	1	-
Indication of the actual number of TRIG_EOI_LIM_x					
NR_TRIG_EOI_LIM_PREV	V/O	0 ... 7 H	0 ... 7	1	-
Indication of the previous number of TRIG_EOI_LIM_x					
INJ_MOD_GLOBAL	V/O	0 ... FFH	0 ... 255	1	-
Global injection mode for all cylinders					
STATE_INJ_MOD_SWI_ACT	V/O	0 ... FFH	0 ... 255	1	-
State indicates a currently ongoing injection mode change, if not equal to zero					
TI_ADD_DLY	V/O	8000 ... 7FFFH	-131.07 ... 131.03	0.004	ms
Injector dead time correction, homogeneous mode 1. pulse					
SOI_1_MES_x	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the first injection pulse, estimated					
SOI_2_MES_x	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the additional pulse, estimated					
EOI_1_ MES_x	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the first injection pulse, estimated					
EOI_2_ MES_x	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the additional pulse, estimated					
TI_1_ MES_x	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Actual performed cylinder individual injection time, first pulse					
TI_2_ MES_x	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Actual performed cylinder individual injection time of the additional pulse					
LV_STATE_PREV_IV_x	V/O	0 ... 1H	0 ... 1	1	-
Cylinder individuel flag which indicates whether last injection was activated or deactivated					
PREV_STATE_IV	V/O	0 ... FFH	0 ... 255	1	-
Bit coded byte which indicates whether a cylinder was deactivated or not					
INH_IV_DYN	V/O	0...FFH	0...255	1	-
Shut off pattern for dynamic shut off (fixed cylinder allocation)					
REF_EOI_LIM	O	FC40 ... 0H	-360 ... 0	0.375	°CRK
Offset of the EOI_LIM reference point after TDC					
TI_1_x	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, first pulse					
EOI_1_x	V	0 ... 780H	0 ... 720	0.375	°CRK
Cylinder individual end of injection, first pulse					
INJ_MOD_x	V	0 ... FFH	0 ... 255	1	-
Cylinder individual injection mode					
CRK_PSN_ENG_INJ_UPD	V	0 ... 780H	0 ... 720	0.375	°CRK
Engine position after injection data update					
LV_INJ_UPD_TRM	-	0 ... 1H	0 ... 1	1	-
Flag which indicates that at least one data set was sent to I/O-SW					
LV_INH_INJ_x	V	0 ... 1H	0 ... 1	1	-
Flag that indicates if cylinder shut off is active or not					
LV_INH_INJ_OLD	-	0 ... 1H	0 ... 1	1	-
Flag that indicates if cylinder shut off was active or not – contains the information from old calculation run					
EOI_MIN_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest possible EOI relatet to EOI_LIM					
LV_ADD_PULSE_x	-	0 ... 1H	0 ... 1	1	-
Switch to enable additional injection pulse at homogeneous mode, single injection mode (injection update at transient operation)					

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
CTR_PSN_INH_IV_DYN	-	0 ... 15H	0 ... 15	1	-
Position counter for dynamic cylinder shut off sequence calculation					
NR_CYL_INH_IV_DYN	-	0 ... 7H	0 ... 7	1	-
Destination logical cylinder number for dynamic fuel shut off					
NR_CYL_OFS_INH_IV_DYN	V	0 ... 7H	0 ... 7	1	-
Cylinder offset for destination of current cylinder number for dynamic fuel shut off					
TI_ADD_PULSE_MIN	V	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time for additive injection pulse					
FAC_ADD_PULSE	V	0 ... FFH	0 ... 0.996094	0.0039	-
Weighting factor for injection time update at transient conditions (additive pulse)					

Input data:

TI_1_HOM_x	EOI_1_HOM_x	LV_ES	SUM_INH_INJ
LV_TI_EXT_ADJ_x	TI_EXT_ADJ_x	EOI_MIN_HOM	STATE_INH_IV_DYN
NC_EOI_LIM	LV_ST_END	EOI_MIN_PRE_INJ	INH_INJ
NC_CYL_NR	SOI_MAX	TI_ADD_DLY_1_HOM	LV_ADD_PULSE_ENA
NC_ST_FIRST_INJ_END	NC_ST_ACT_PRE_INJ	EOI_PRE_INJ[NC_CYL_NR]	STATE_ST_PRE
NC_ST_CONF_PRE_END	NC_ST_ACT_SYN_INJ	TI_PRE_INJ[NC_CYL_NR]	VB
NC_ST_ENA_PRE_INJ	PSN_ENG	NC_ECU_SAMPLE_FAST	INH_IV
N_32		NC_INJ_INH_SWI_IV_SHIFT_NR	

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_TI_ADD_PULSE_MIN	8	0 ... FFFFH	0 ... 262.14	0.004	ms
LDPM_VB_1_INJR	8	0 ... FFH	0 ... 26	0.102	V
Minimum injection time for additive injection pulse, homogeneous mode					
IP_FAC_ADD_PULSE	6	0 ... FFH	0 ... 0.996094	0.0039	-
LDP_N_32_IP_FAC_ADD_PULSE	6	0 ... FFH	0 ... 8160	32	rpm
Weighting factor for additive pulses in case of injection time update at transient conditions					
C_EOI_EXT_ADJ	1	0 ... 780H	0 ... 720	0.375	°CRK
End of injection for external adjustment (service tool intervention)					
C_NR_CYL_OFS_INH_IV_DYN	1	0 ... 7H	0 ... 7	1	-
Cylinder offset for destination of current cylinder number for dynamic fuel shut off					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_INJ_MOD_DI	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate that injection is disabled					
NC_INJ_MOD_HOM	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate homogeneous mode					
NC_INJ_MOD_SINGLE	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate single injection mode					
NC_INJ_MOD_PRE_INJ	1	0 ... FFH	0 ..255	1	-
Constant defined to indicate pre injection mode					
NC_INJ_MOD_MASK_1	1	0 ... FFH	0 ..255	1	-
Mask					
NC_INJ_MOD_MASK_2	1	0 ... FFH	0 ..255	1	-
Mask					

Transfer data to I/O SW:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_TI_PLS_1_x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, first pulse					
TRL_EOI_1_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Cylinder individual end of injection, first pulse					
TRL_INJ_MOD_GLOBAL	-	0 ... FFH	0 ... 255	1	-
Global injection mode for all cylinders					
TRL_INJ_MOD_x	-	0 ... FFH	0 ... 255	1	-
Cylinder individual injection mode					
TRL_EOI_MIN_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest possible EOI relatet to EOI_LIM					
TRL_LV_ADD_PULSE_x	-	0 ... 1H	0 ... 1	1	-
Switch to enable additional injection pulse at homogeneous mode, single injection mode (injection update at transient operation)					
TRL_LV_INH_INJ_x	-	0 ... 1H	0 ... 1	1	-
Flag that indicates if cylinder shut off is active or not					
TRL_SOI_MAX	-	0 ... 780H	0 ... 720	0.375	°CRK
Earliest possible SOI					
TRL_TI_ADD_DLY	-	8000 ... 7FFFH	-131.07 ... 131.03	0.004	ms
Injector dead time correction, homogeneous mode 1. pulse					
TRL_TI_ADD_PULSE_MIN	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Minimum injection time for additive injection pulse					
TRL_FAC_ADD_PULSE	-	0 ... FFH	0 ... 0.996094	0.0039	-
Weighting factor for injection time update at transient conditions (additive pulse)					
TRL_TRIG_INJ_UPD	-	-	-	-	-
Trigger for update of calculated injection data					
TRL_NC_EOI_LIM	-	0 ... 780H	0 ... 720	0.375	°CRK
End of phasing range relatet to NC_REF_EOI_LIM					
TRL_EOI_PRE_INJ [NC_CYL_NR]	-	0 ... 780H	0 ... 720	0.375	°CRK
End of injection for preinjection					
TRL_EOI_MIN_PRE_INJ	-	0 ... 780H	0 ... 720	0.375	°CRK
Latest end of injection angle for pre injection					
TRL_TI_PRE_INJ [NC_CYL_NR]	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Pre injection time					

Transfer data from I/O SW:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TRL_INJ_StateOfPrevInj_x	-	-	0...1	1	-
Cylinder individuel flag which indicates whether last injection was activated or deactivated					
TRL_SOI_1_MES_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the first injection pulse, estimated					
TRL_EOI_1_ MES_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the first injection pulse, estimated					
TRL_TI_1_ MES_x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, first pulse					
TRL_SOI_2_MES_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed SOI of the second injection pulse, estimated					
TRL_EOI_2_ MES_x	-	0 ... 780H	0 ... 720	0.375	°CRK
Actual performed EOI of the second injection pulse, estimated					
TRL_TI_2_ MES_x	-	0 ... FFFFH	0 ... 262.14	0.004	ms
Cylinder individual injection time, second pulse					
TRL_TRIG_EOI_LIM(CYL_AV)	-	-	-	-	-
Trigger at every EOI_LIM_x					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CYL_AV	-	0 ... 7 H	0 ... 7	1	-
Function parameter of TRL_TRIG_EOI_LIM, which indicate the actual logical cylinder number for that cylinder which EOI_LIM was reached.					
TRL_NC_REF_EOI_LIM	-	0 ... FC40 H	0 ... -360	0.375	°CRK
Offset of the EOI_LIM reference point after TDC					

Note: The prefix TRL to a data xxx (e.g. TI_1_x) indicates a I/O SW function interface for that data.

FUNCTION DESCRIPTION:

7.2.1 Reset tasks

Application conditions:

Activation: at reset

Deactivation: -

Initialization: -

Recurrence: once at reset

Note! This task has to be called first in calculation order of reset tasks.

This chapter describe the procedures required at reset for an accurate engine start.

7.2.1.1 Formula Section:

TRL_NC_EOI_LIM = NC_EOI_LIM → transfer to I/O-SW

REF_EOI_LIM = TRL_NC_REF_EOI_LIM → receive from I/O-SW

STATE_INJ_MOD_SWI_ACT = 0

7.2.2 First valid tooth tasks

Application conditions:

Activation: at first valid tooth

Deactivation: -

Initialization: -

Recurrence: once at first valid tooth

This chapter describe the procedures required at first valid tooth for an accurate engine start.

7.2.2.1 Formula Section:

INH_IV_DYN = 0

NR_CYL_OFS_INH_IV_DYN = C_NR_CYL_OFS_INH_IV_DYN

INJ_MOD_GLOBAL = NC_INJ_MOD_HOM
OR ...bitwise
NC_INJ_MOD_SINGLE

(1) IF STATE_ST_PRE **=** NC_ST_ENA_PRE_INJ ..bitwise

(1) THEN

(1) FOR x = 0 **TO** (NC_CYL_NR –1) **DO:**

INJ_MOD[x] = INJ_MOD_GLOBAL

TRL_TI_PRE_INJ[x] = TI_PRE_INJ[x] → transfer to I/O-SW

TRL_EOI_ PRE_INJ[x] = EOI_ PRE_INJ[x] → transfer to I/O-SW

TRL_EOI_MIN_PRE_INJ = EOI_MIN_PRE_INJ → transfer to I/O-SW

(1) ENDFOR

(1) ELSE

(2) FOR x = 0 **TO** (NC_CYL_NR –1) **DO:**

INJ_MOD[x] = INJ_MOD_GLOBAL

(2) ENDFOR

(1) ENDIF

LV_INJ_UPD_TRM = 0

Note! Synchronised injection will be enabled after first valid tooth only if at least one injection time calculation was completed (LV_INJ_UPD_TRM = 1) and engine is synchronised.

NR_TRIG_EOI_LIM_AV = 0

NR_TRIG_EOI_LIM_PREV = 0

7.2.2.2 State indication: configuration of preinjection was finished

Indicate in state variable STATE_ST_PRE that the preinjection data are transferred to I/O-SW and the configuration of preinjection was finished.

STATE_ST_PRE **=** STATE_ST_PRE **OR** NC_ST_CONF_PRE_END ...bitwise

7.2.3 Segment synchronous tasks

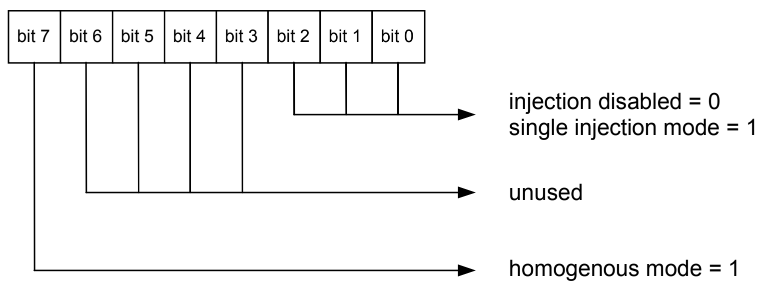
Application conditions:

Activation:	every engine state	
Deactivation:	-	
Initialization:	see item 1.1	
Recurrence:	LV_ST_END = 0:	10 ms
	LV_ST_END = 1:	segment synchronous

7.2.3.1 Generation of INJ_MOD_GLOBAL (global injection mode)

7.2.3.1.1 Definition of INJ_MOD_GLOBAL

INJ_MOD_GLOBAL =



e.g.: homogeneous mode + single injection mode

INJ_MOD_GLOBAL =

1	0	0	0	0	0	0	1

File: INJ_MOD_GLOBAL.vsd

7.2.3.1.2 Definitions of the non calibrateable constants to generate INJ_MOD_GLOBAL

NC_INJ_MOD_DI = 0x00H
NC_INJ_MOD_HOM = 0x80H
NC_INJ_MOD_SINGLE = 0x01H
NC_INJ_MOD_PRE_INJ = 0x04H
NC_INJ_MOD_MASK_1 = 0x80H
NC_INJ_MOD_MASK_2 = 0x07H

7.2.3.1.3 Calculation of INJ_MOD_GLOBAL

Have to be done at first valid tooth only.

7.2.3.2 Transfer data

Cylinder individual transfer data:

(1) FOR x = 0 **TO** (NC_CYL_NR-1) **DO:**
"Transfer Section" for cylinder x (see chapter 7.2.5 Common Transfer Operations).

LV_INH_INJ_x = Bit x of INH_INJ

TRL_LV_INH_INJ_x = LV_INH_INJ_x → transfer to I/O-SW

Note! Pay attention on data consistency of LV_INH_INJ_x.

TRL_LV_ADD_PULSE_x = LV_ADD_PULSE_x → transfer to I/O-SW

TRL_EOI_MIN_x = EOI_MIN_x → transfer to I/O-SW

TRL_TI_PLS_1_x = TI_1_x → transfer to I/O-SW

TRL_EOI_1_x = EOI_1_x → transfer to I/O-SW

(1) ENDFOR

TI_ADD_PULSE_MIN = IP_TI_ADD_PULSE_MIN (Input: VB)

FAC_ADD_PULSE = IP_FAC_ADD_PULSE (Input: N_32)

TI_ADD_DLY = TI_ADD_DLY_1_HOM

Global transfer data:
(mode independent and not cylinder individual data)

TRL_SOI_MAX = SOI_MAX → transfer to I/O-SW

TRL_FAC_ADD_PULSE = FAC_ADD_PULSE → transfer to I/O-SW

TRL_TI_ADD_PULSE_MIN = TI_ADD_PULSE_MIN → transfer to I/O-SW

TRL_TI_ADD_DLY = TI_ADD_DLY → transfer to I/O-SW

After transfer a TRL_TRIG_INJ_UPD call is required.
Identify the engine position after injection data update. It is referenced to EOI_LIM.
Note! If the engine position information is not available CRK_PSN_ENG_INJ_UPD should be assigned to zero.

CRK_PSN_ENG_INJ_UPD = ((720° - PSN_ENG - EOI_LIM_0)+720°) MODULA 720°

Indicate, that at least one data set was sent to I/O-SW.

LV_INJ_UPD_TRM = 1

7.2.4 EOI_LIM triggered tasks

Application conditions:

Activation: every TRIG_EOI_LIM event

Deactivation: -

Initialization: -

Recurrence: at TRL_TRIG_EOI_LIM[x]

Note! x represents the current assigned cylinder number cyl_av, which is used by subsequent data calculations.

7.2.4.1 Transfer data

CTR_PSN_INH_IV_DYN = (STATE_INH_IV_DYN / 2^{16}) **AND** 0xF ... bitwise

(1) FOR x = cyl_av **DO:**

NR_CYL_INH_IV_DYN = (NC_CYL_NR + x - NR_CYL_OFS_INH_IV_DYN)
MODULO NC_CYL_NR

(1) IF (STATE_INH_IV_DYN **AND** $2^{\text{CTR\_PSN\_INH\_IV\_DYN}}$) ≠ 0 ... bitwise

(1) THEN

Cylinder will be switched off:

INH_IV_DYN = INH_IV_DYN **OR** $2^{\text{NR\_CYL\_INH\_IV\_DYN}}$ ... bitwise

(1) ELSE

Cylinder will be switched on:

INH_IV_DYN = INH_IV_DYN **AND NOT**($2^{\text{NR\_CYL\_INH\_IV\_DYN}}$) ... bitwise

(1) ENDIF

(1) ENDFOR

CTR_PSN_INH_IV_DYN = CTR_PSN_INH_IV_DYN + 1

(2) IF CTR_PSN_INH_IV_DYN ≥ NC_INJ_INH_SWI_IV_SHIFT_NR

(2) THEN

CTR_PSN_INH_IV_DYN = 0

(2) ENDIF

STATE_INH_IV_DYN = (STATE_INH_IV_DYN **AND** 0x0000FFFF) ... bitwise
OR ... bitwise
(CTR_PSN_INH_IV_DYN * 2^{16}) ... bitwise

INH_INJ = INH_IV **OR** INH_IV_DYN ... bitwise

LV_INH_INJ_OLD = LV_INH_INJ[NR_CYL_INH_IV_DYN]

LV_INH_INJ[NR_CYL_INH_IV_DYN] = Bit NR_CYL_INH_IV_DYN of INH_INJ

Note! Pay attention on data consistency of INH_INJ, INH_IV_DYN and LV_INH_INJ[x].

SUM_INH_INJ = The sum of the digits of INH_INJ

(3) IF LV_INH_INJ[NR_CYL_INH_IV_DYN] ≠ LV_INH_INJ_OLD

(3) THEN

TRL_LV_INH_INJ[NR_CYL_INH_IV_DYN] = LV_INH_INJ[NR_CYL_INH_IV_DYN]
→ transfer to I/O-SW

(3) ENDIF

Please Note! STATE_INH_IV_DYN is an output data of cylinder shut off specification. The content of the data will be changed in this interrupt procedure. Pay attention on data consistency, even at cylinder shut off module.

7.2.4.2 Transfer of measured injection related data from I/O-SW

After an Injection was performed, the (cylinder individual) SOI, EOI and TI of that injection is indicated at the output interface as:

SOI_1_MES_x ... actual performed SOI of the first injection pulse (estimated)

EOI_1_MES_x ... actual performed EOI of the first injection pulse (estimated)

TI_1_MES_x ... actual performed injection time of the first injection pulse

SOI_2_MES_x ... actual performed SOI of the additional injection pulse (estimated)

EOI_2_MES_x ... actual performed EOI of the additional injection pulse (estimated)

TI_2_MES_x ... actual performed injection time of the additional injection pulse

(1) FOR x = cyl_av **DO:**

SOI_1_MES_x = TRL_SOI_1_MES_x → receive from I/O-SW

EOI_1_MES_x = TRL_EOI_1_MES_x → receive from I/O-SW

TI_1_MES_x = TRL_TI_1_MES_x → receive from I/O-SW

SOI_2_MES_x = TRL_SOI_2_MES_x → receive from I/O-SW

EOI_2_MES_x = TRL_EOI_2_MES_x → receive from I/O-SW

TI_2_MES_x = TRL_TI_2_MES_x → receive from I/O-SW

(1) ENDFOR

7.2.4.3 Calculation of LV_STATE_PREV_IV_x and PREV_STATE_IV

(1) FOR x = cyl_av **DO:**

LV_STATE_PREV_IV_x = TRL_INJ_StateOfPrevInj_x → receive from I/O-SW

(INJ_StateOfPrevInj_x = 1 .... logical cylinder x was injected)

(INJ_StateOfPrevInj_x = 0 .... logical cylinder x was not injected)

(1) IF

LV_STATE_PREV_IV_x = 1

(1) THEN

Set bit_x of PREV_STATE_IV (the corresponding cylinder was injected)

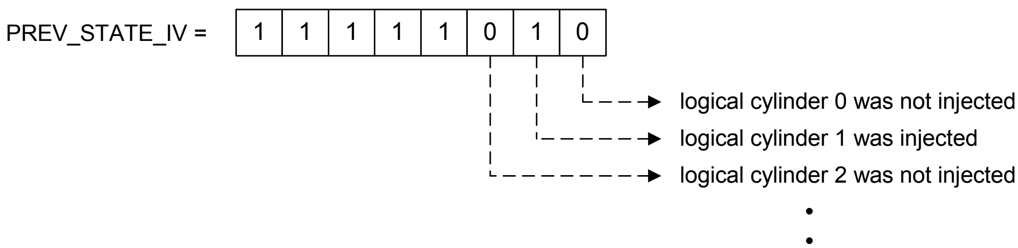
(1) ELSE

Clear bit_x of PREV_STATE_IV (the corresponding cylinder was not injected)

(1) ENDIF

Note! Pay attention on data consistency of PREV_STATE_IV and LV_STATE_PREV_IV_x.

(1) ENDFOR



7.2.4.4 Indication of actual and previous number of TRIG_EOI_LIM

The variable cyl_av is delivered from the I/O SW and indicates the actual TRIG_EOI_LIM

NR_TRIG_EOI_LIM_PREV = NR_TRIG_EOI_LIM_AV

NR_TRIG_EOI_LIM_AV = cyl_av

Note! cyl_av is a parameter of the TRL_TRIG_EOI_LIM function interface.

7.2.5 Common Transfer Operations

Note! x represents an assigned cylinder number, which is defined in the upper data calculation function.

(1) IF

LV_TI_EXT_ADJ_x = 0

(1) THEN

TI_1_x = TI_1_HOM_x

EOI_1_x = EOI_1_HOM_x

(1) ELSE

TI_1_x = TI_EXT_ADJ_x ... service tool intervention

EOI_1_x = C_EOI_EXT_ADJ

(1) ENDIF

EOI_MIN_x = EOI_MIN_HOM

LV_ADD_PULSE_x = LV_ADD_PULSE_ENA

Remark:

Minimum injection time for additional pulse:

In case of single injection and homogeneous mode a second injection pulse can be applied by the basic software at transient engine operation (as a result of an injection update). For this pulse a minimum injection time TI_ADD_PULSE_MIN is defined.

Each of the applied IP_TI_ADD_PULSE_MIN data should be smaller than the smallest calculated TI_MIN.

7.2.6 Start with pre injection

Application conditions:

Deactivation: **(**STATE_ST_PRE **AND** NC_ST_FIRST_INJ_END**)** = NC_ST_FIRST_INJ_END ...bitwise

Activation: **(**STATE_ST_PRE **AND (**NC_ST_ACT_PRE_INJ **OR** NC_ST_CONF_PRE_END**))** = NC_ST_ACT_PRE_INJ **OR** NC_ST_CONF_PRE_END ...bitwise

Initialization: -

Recurrence: NC_ECU_SAMPLE_FAST (the recurrence has to be same recurrence as the output of the ENSD aggregate)

Note! Deactivation condition has to be checked before the activation condition.

Formula Section:

TRL_EOI_ PRE_INJ[0] = EOI_ PRE_INJ[0] → transfer to I/O-SW

TRL_INJ_MOD_GLOBAL = NC_INJ_MOD_PRE_INJ → transfer to I/O-SW

STATE_ST_PRE **=** STATE_ST_PRE **OR** NC_ST_FIRST_INJ_END ...bitwise

Note! Indicate in state variable STATE_ST_PRE that the preinjection mode is transferred to I/O-SW and the preinjection is in progress. One preinjection pulse per cylinder is performed by the I/O SW. After a pulse is done the mode changes from preinjection mode to synchronised mode, managed by the I/O-SW cylinder individual.

7.2.7 Synchronised start

Application conditions:

Deactivation: (STATE_ST_PRE **AND** NC_ST_FIRST_INJ_END) = NC_ST_FIRST_INJ_END ...bitwise

Activation: **(**STATE_ST_PRE **AND (**NC_ST_ACT_SYN_INJ **OR** NC_ST_CONF_PRE_END**))** = NC_ST_ACT_SYN_INJ **OR** NC_ST_CONF_PRE_END ...bitwise

Initialization: -

Recurrence: NC_ECU_SAMPLE_FAST (the recurrence has to be same recurrence as the output of the ENSD aggregate)

Note! Deactivation condition has to be checked before the activation condition.

Formula Section:

(1) IF LV_INJ_UPD_TRM = 1

Check if at least one data set was sent to I/O-SW before starting the injection.

(1) THEN

TRL_INJ_MOD_GLOBAL = INJ_MOD_GLOBAL → transfer to I/O-SW

After transfer a TRL_TRIG_INJ_UPD call is required.

STATE_ST_PRE = STATE_ST_PRE **OR** NC_ST_FIRST_INJ_END ...bitwise
Note! Indicate in state variable STATE_ST_PRE that engine is synchronised and injection has been started.

(1) ENDIF

7.2.8 Engine stop

Application conditions:

Activation: at transition of LV_ES from 0 -> 1

Deactivation: -

Initialization: -

Recurrence: every transition of LV_ES from 0 -> 1

This chapter describe the engine stop procedure as required for this specification.

7.2.8.1 Formula Section:

Disable task interruption up to here:

INJ_MOD_GLOBAL = NC_INJ_MOD_HOM
OR ...bitwise
NC_INJ_MOD_DI

(1) FOR x = 0 **TO** (NC_CYL_NR –1) **DO:**

INJ_MOD[x] = INJ_MOD_GLOBAL

(1) ENDFOR

STATE_ST_PRE = STATE_ST_PRE **OR** NC_ST_FIRST_INJ_END ...bitwise

Now, task interruption can be enabled again.

TRL_INJ_MOD_GLOBAL = NC_INJ_MOD_DI → transfer to I/O-SW

PREV_STATE_IV = 0 (No Injection is performed by I/O-SW from this time)

(2) FOR x = 0 **TO** (NC_CYL_NR –1) **DO:**

LV_STATE_PREV_IV[x] = 0

(2) ENDFOR

7.3 Configuration of Preinjection

7.3.1 Pre injection configuration and phase

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
EOI_PRE_INJ[NC_CYL_NR]	O/V	0 ... 780H	0 ... 720	0.375	°CRK
End of injection for preinjection					
EOI_MIN_PRE_INJ	O/V	0 ... 780H	0 ... 720	0.375	°CRK
Latest end of injection angle for pre injection					
STATE_ST_PRE	O/V	0 ... FFH	0 ... 255	1	-
State for start pre injection					
PSN_DIF_ENG_SYN_THD	V	0 ... 780H	0 ... 720	0.375	°CRK
Difference of the min and max estimated engine position in the non syn. phase to trigger a preinjection					
PSN_DIF_ENG_SYN	V	0 ... 780H	0 ... 720	0.375	°CRK
Difference of the min and max estimated engine position in the non syn. phase					

Input data:

NC_CYL_NR	LV_ERR_CAM	TCO	TI_PRE_INJ[NC_CYL_NR]
PSN_ENG_SYN_MIN	PSN_ENG_SYN_MAX	INH_INJ	N
NC_ECU_SAMPLE_FAST	LV_RUN_ENG		

Description:

The definition of the variables represents the ASW part of the preinjection strategy as well as the interface to the BSW.

FUNCTION DESCRIPTION:

The aim is to **pre-synchronise** and start the engine as fast as possible. The experience has shown, that when the engines was switched off, it will stop in the most of the cases at the same specific positions. This information can be used together with the information of relative engine positions PSN_ENG_SYN_MIN and PSN_ENG_SYN_MAX to pre-synchronise the engine. In the case that the difference of PSN_ENG_SYN_MIN and PSN_ENG_SYN_MAX is lower than the threshold PSN_DIF_ENG_SYN_THD than the probability to put the pre injection on the valid cylinder with a good phasing is 100%. The preinjection shall carried out with the **latest possible** CRK angle for the preinjection, PSN_ENG_SYN_MAX. Due to the fact, that PSN_ENG_SYN_MAX is nearly fix and PSN_ENG_SYN_MIN change from tooth to tooth and in addition to that the EOI phasing calculation is done with PSN_ENG_SYN_MIN, the first pre injection angle has to be corrected by PSN_DIF_ENG_SYN.

The angle ANG_ENG_SYN_DLY describes beside the GAP (see case 1) and the CAM (see case 2) event a third event (see case 3). In the case that the running engine turns already ANG_ENG_SYN_DLY and no CAM or GAP event occur, then the variable PSN_ENG_SYN_MAX will set to the next possible CAM or GAP position. The angle ANG_ENG_SYN_DLY is defined in the aggregate ENSD.

Injection phasing for preinjection:

The cylinder individual start-of-preinjection-angle EOI_PRE_INJ [NC_CYL_NR] is calculated only once after the first valid tooth of the crankshaft signal is detected. It take into consideration the strong engine speed increase at start via the table ID_EOI_PRE_INJ_OFS. The preinjection is finished after all NC_CYL_NR – 1 injections are applied.

Application conditions:

Initialisation: at reset :

STATE_ST_PRE = NC_STATE_ST_PRE_INIT

Activation: At first valid tooth.

Recurrence: only once at activation

Formula section:

PSN_DIF_ENG_SYN_THD = IP_PSN_DIF_ENG_SYN_THD

IF TCO <= C_TCO_MAX_PRE_INJ_SEQ

{first injection will be carried out sequential after synchronisation}

THEN STATE_ST_PRE = NC_ST_ENA_SYN_INJ

{first injection will be carried out as preinjection}

ELSE STATE_ST_PRE = NC_ST_ENA_PRE_INJ

END

EOI_MIN_PRE_INJ = IP_EOI_MIN_PRE_INJ

{latest end of injection angle for pre injection and for identifying of the first injected cylinder}

FOR CYC_PRE = 0 **TO** NC_CYL_NR – 1 **DO**

EOI_PRE_INJ [CYC_PRE] = IP_EOI_PRE_INJ + ID_EOI_PRE_INJ_OFS

IF (TI_PRE_INJ [CYC_PRE] = 0)

{first injection will be carried out sequential after synchronisation}

THEN STATE_ST_PRE = NC_ST_ENA_SYN_INJ

END

END FOR

IF LV_ERR_CAM = 1

THEN STATE_ST_PRE = NC_ST_ENA_SYN_INJ

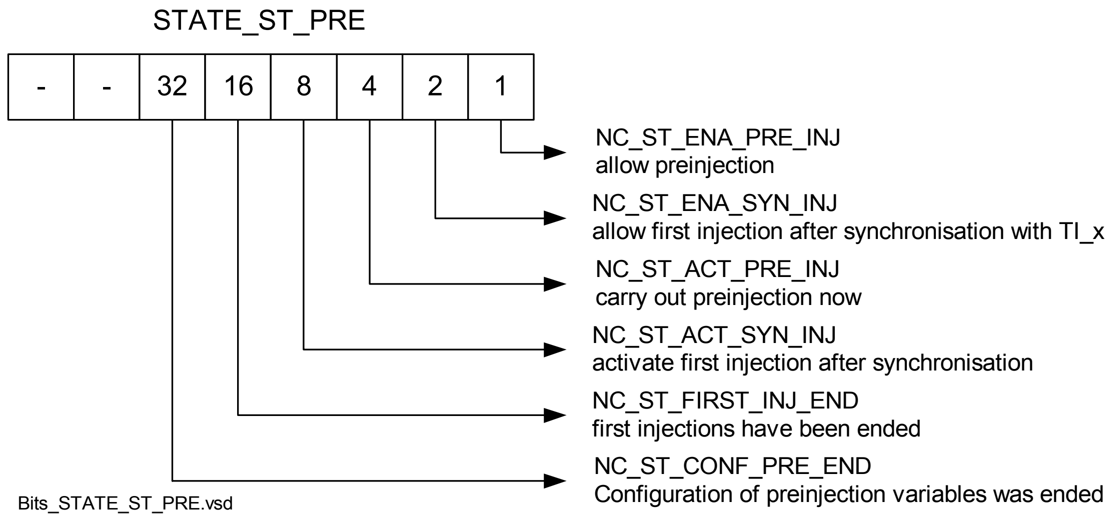
END

In the transfer function to I/O SW the state NC_ST_CONF_PRE_END is set in order to signalise, that the variables were successfully configured and have reached the BSW.

Note:

The counter CYC_PRE is a local variable.

Definition of STATE_ST_PRE:



Bits_STATE_ST_PRE.vsd

The NC variables to indicate the start state are defined as:

NC_STATE_ST_PRE_INIT = 00H

NC_ST_ENA_PRE_INJ = 01H

NC_ST_ENA_SYN_INJ = 02H

NC_ST_ACT_PRE_INJ = 04H

NC_ST_ACT_SYN_INJ = 08H

NC_ST_FIRST_INJ_END = 10H

NC_ST_CONF_PRE_END = 20H

7.3.2 Enable pre injection

FUNCTION DESCRIPTION:

General information:

Several conditions has to be fulfilled in order to permit a preinjection. In the case, that a preinjection shall carried out, the bit of STATE_ST_PRE = NC_ST_ENA_PRE_INJ, otherwise the standard injection with the fuel mass TI_CYL_x is applied synchronously and STATE_ST_PRE = NC_ST_ENA_SYN_INJ.

The preinjection will carried out at the moment when the difference, PSN_DIF_ENG_SYN, of PSN_ENG_SYN_MAX and PSN_ENG_SYN_MIN is lower than the threshold PSN_DIF_ENG_SYN_THD. This event is indicated by setting the bit NC_ST_ACT_PRE_INJ. The function is activated at the moment when the first valid tooth was passed and the pre injection variables were configured.

In the transfer function to I/O SW the state NC_ST_FIRST_INJ_END is set.

Application conditions:

Initialisation: -

Recurrence: NC_ECU_SAMPLE_FAST (the recurrence has to be the same recurrence as the output of the ENSD aggregate)

Deactivation: STATE_ST_PRE **AND** NC_ST_FIRST_INJ_END = NC_ST_FIRST_INJ_END

{Deactivation condition has to be checked before the activation condition}

Activation: STATE_ST_PRE **AND** NC_ST_CONF_PRE_END = NC_ST_CONF_PRE_END

Formula section:

```
PSN_DIF_ENG_SYN = ((PSN_ENG_SYN_MAX - PSN_ENG_SYN_MIN)+720°) MODULA 720°
IF   PSN_DIF_ENG_SYN  <  PSN_DIF_ENG_SYN_THD  AND INH_INJ = 0  THEN
     IF LV_RUN_ENG = 1   THEN                          {engine running detected}

        {Check start with pre injection}
        IF  STATE_ST_PRE AND NC_ST_ENA_PRE_INJ =  NC_ST_ENA_PRE_INJ  THEN
            IF  N < C_N_LOW_ENA_PRE_THD               {low engine speed range detected}
              THEN STATE_ST_PRE = NC_ST_CONF_PRE_END
                                  OR NC_ST_ENA_PRE_INJ
                                  OR  NC_ST_ACT_PRE_INJ       {activate pre injection}
                   EOI_PRE_INJ [0] = EOI_PRE_INJ[0] + PSN_DIF_ENG_SYN  {correct first}
              ELSE STATE_ST_PRE = NC_ST_CONF_PRE_END                   {switch to }
                                  OR NC_ST_ENA_SYN_INJ       { synchronous injection}
            END
        END

        {Check start with synchrony injection}
        IF    STATE_ST_PRE AND NC_ST_ENA_SYN_INJ =  NC_ST_ENA_SYN_INJ  THEN
            IF PSN_DIF_ENG_SYN = 0   THEN                 {synchronisation detected}
               STATE_ST_PRE = NC_ST_CONF_PRE_END
                              OR NC_ST_ENA_SYN_INJ
                              OR  NC_ST_ACT_SYN_INJ           {activate syn. injection}
            END
        END
     END  {of LV_RUN_ENG}
END
```

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_ PSN_DIF_ENG_ SYN_THD	4	0 ... 780H	0 ... 720	0.375	°CRK
LDP_TCO_IP_PSN_DIF_ENG_SYN_THD	4	0 ... FEH	-48 ... 142.5	0.75	°C
Lower threshold of the difference of the estimated engine position in the non syn. phase to allow a preinjection					
IP_EOI_PRE_INJ	4	0 ... 780H	0 ... 720	0.375	°CRK
LDP_TCO_IP_EOI_PRE_INJ	4	0 ... FEH	-48 ... 142.5	0.75	°C
End of injection for pre injection					
ID_EOI_PRE_INJ_OFS	8	F100 ... 0F00H	-1440 ... 1440	0.375	°CRK
LDP_CYC_PRE_ID_EOI_PRE_INJ_OFS	8	0 ... 7H	0 ... 7	1	-
Cylinder individual offset to EOI for pre injection					
IP_EOI_MIN_PRE_INJ	4	0 ... 780H	0 ... 720	0.375	°CRK
LDP_TCO_IP_EOI_MIN_PRE_INJ	4	0 ... FEH	-48 ... 142.5	0.75	°C
Latest end of injection for pre injection					
C_TCO_MAX_PRE_INJ_SEQ	1	0...FEH	-48...142.5	0.75	°C
Maximum coolant temperature for sequential TI_PRE_INJ application					
C_N_LOW_ENA_PRE_THD	1	0...1FE0H	0 ... 8160	1	rpm
The pre injection will be switched off if this engine speed threshold is exceeded.					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ST_ENA_PRE_INJ	1	0 ... FFH	0 ...255	1	-
Flag to signalise that a pre-injection will carried out					
NC_ST_ENA_SYN_INJ	1	0 ... FFH	0 ...255	1	-
Flag to signalise that the first injection is carried out after synchronisation					
NC_ST_FIRST_INJ_END	1	0 ... FFH	0 ...255	1	-
Flag to signalise that the first injection was ended					
NC_ST_ACT_PRE_INJ	1	0 ... FFH	0 ...255	1	-
Flag to signalise that the preinjection has to be activated as first injection now					
NC_ST_ACT_SYN_INJ	1	0 ... FFH	0 ...255	1	-
Activate standard start injection as first injection					
NC_STATE_ST_PRE_INIT	1	0 ... FFH	0 ...255	1	-
Init state of STATE_ST_PRE					
NC_ST_CONF_PRE_END	1	0 ... FFH	0 ...255	1	-
Flag to signalise, that the configuration of the variables was ended at the first valid tooth					

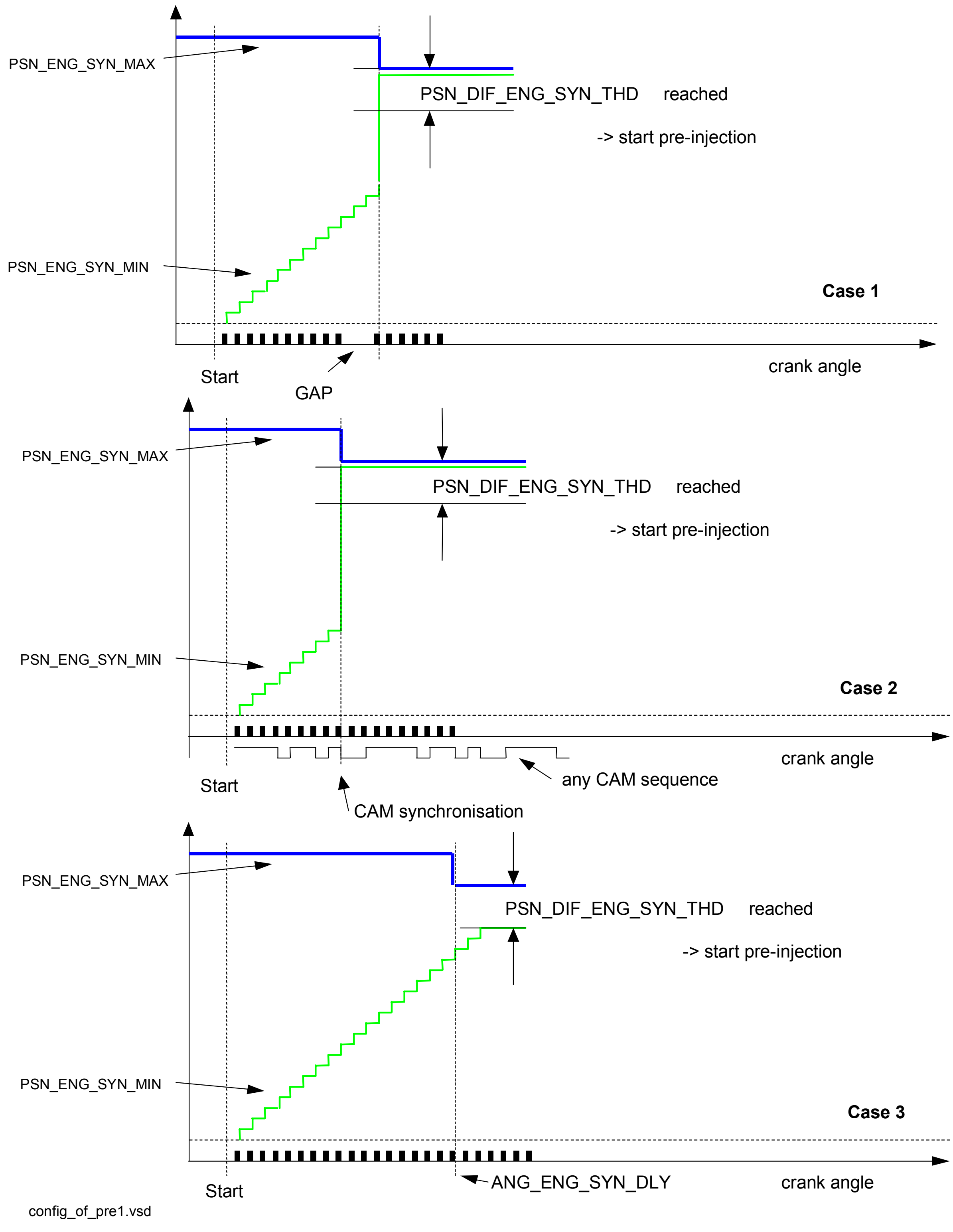


Figure: Schematic begin of pre-injection

7.3.3 Restart function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_ST_REST	V/O	0...FFH	0...1,9922	0,0078	-
Restart-factor for the correction of the mass fuel flow at start					
LV_REST	V/O	0...1H	0...1	1	-
Flag indicating that a restart correction factor not equal to one is applied to the mass fuel flow calculation					

Input data:

TCO	TCO_STOP	T_AST_STOP	T_ES_REST
LV_T_ES_REST_VLD	LV_STALL	LV_ES	LV_ST_ES

FUNCTION DESCRIPTION:

General information:

In order to improve re-starts, a correction factor on the mass fuel flow during pre-injection and start injection is calculated. This factor can in- or decreases the mass fuel flow.

If a valid value for the engine-off-time is available (LV_T_ES_REST_VLD = 1) the factor is calculated dependent on the engine condition at shut down (IP_FAC_REST) and the engine-off-time (IP_REST).
The engine condition at shut down is determined by the coolant temperature at shut down TCO_STOP and the engine running time of the last driving cycle T_AST_STOP which are both stored in non volatile memory. The engine-off-time T_ES_REST and the decision of validity LV_T_ES_REST_VLD comes out of the application incidences for the restart function. In case of engine stalling detection (LV_STALL = 1) an additional correction factor is applied to the calculation of the restart factor.

If no valid value for the engine-off-time is available the standard restart function detects the necessity of a restart correction dependent on T_AST_STOP and the difference between the actual TCO and TCO_STOP (maximal difference: C_TCO_DIF_MAX_REST).
The restart factor is determined by the map IP_FAC_STND_REST dependent on the actual coolant temperature TCO. In case of engine stalling detection (LV_STALL = 1) an additional correction factor is applied to the calculation of the restart factor too.

The flag LV_REST indicates that a restart correction factor not equal to zero is applied to the mass fuel flow calculation.

Application conditions:

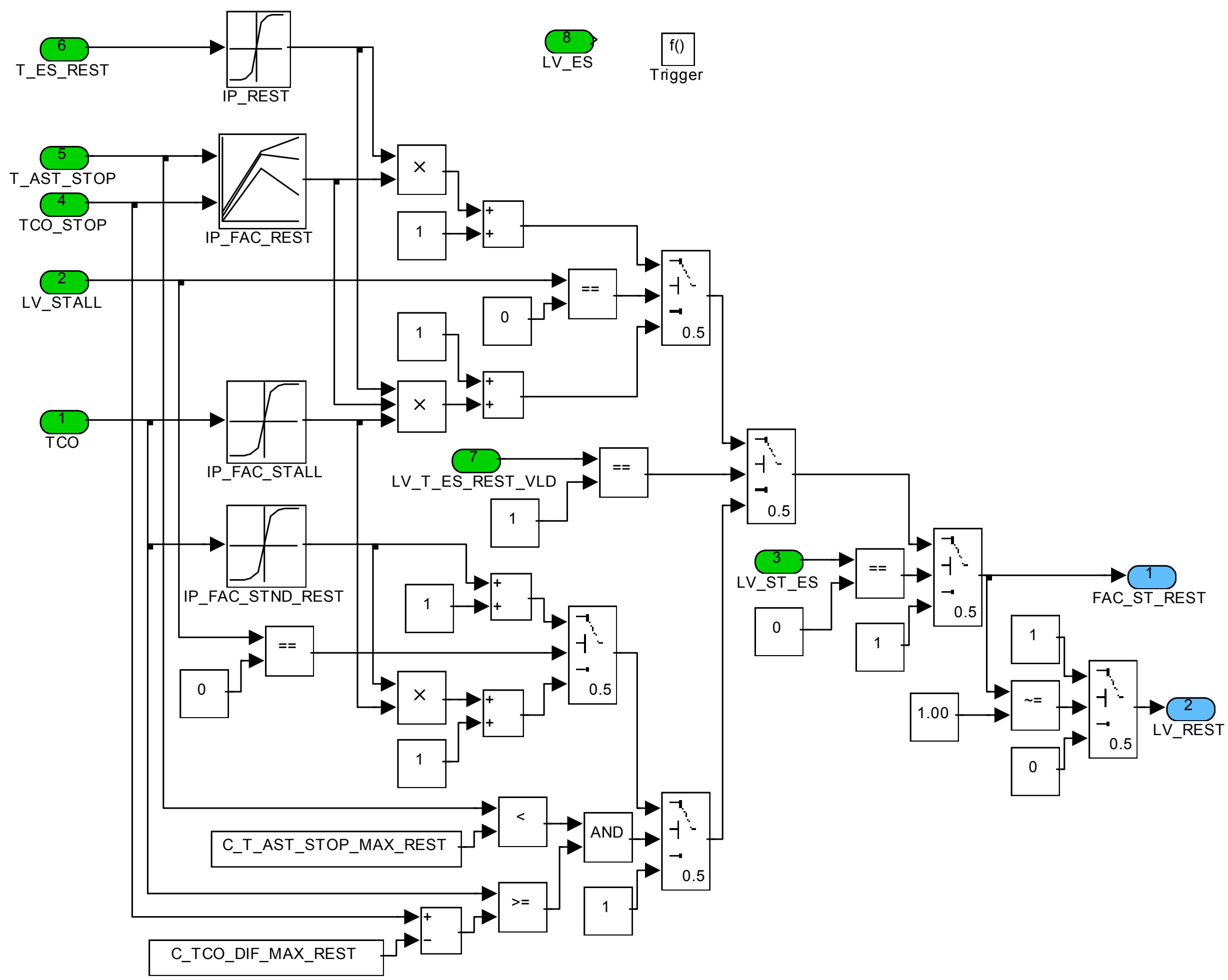
Activation: LV_ES = 1

Deativation: -

Initialisation: at reset: FAC_ST_REST = 1; LV_REST = 0

Recurrence: 10ms

Signal flow diagram:



Formula section:

```
If(1)      LV_ST_ES = 0        then
   If(2)        LV_T_ES_REST_VLD = 1   then
        If(3)   LV_STALL = 0                   then
                     FAC_ST_REST = 1 + IP_FAC_REST * IP_REST
        else(3)          FAC_ST_REST = 1 + IP_FAC_REST * IP_REST
                                   * IP_FAC_STALL
        endif(3)
   else(2)
        If(4)   T_AST_STOP < C_T_AST_STOP_MAX_REST    and
                TCO >= TCO_STOP – C_TCO_DIF_MAX_REST  then
                If(5)       LV_STALL = 0                          then
                             FAC_ST_REST = 1 + IP_FAC_STND_REST
                else(5)          FAC_ST_REST = 1 + IP_FAC_STND_REST
                                        * IP_FAC_STALL
                endif(5)
        else(4)
                FAC_ST_REST = 1
        endif(4)
   endif(2)
else(1)    FAC_ST_REST = 1
endif(1)


If(5)      FAC_ST_REST ≠ 1.00  (80H)        then
           LV_REST = 1
else(5)    LV_REST = 0
endif(5)
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_STND_REST	8	0...FFH	-1.0...0.99215	0.0078123	-
LDPM_TCO_1_FMSP	8	0...FEH	-48...142.5	0.75	°C
Cold start reduction factor in case of re-start with no engine stop time T_ES_REST available					
IP_REST	8x8	0...FFH	-1.0...0.99215	0.0078123	-
LDP_T_ES_REST_IP_REST	8	0...FFFFH	0...65535	1	min
LDPM_TCO_1_FMSP	8	0...FEH	-48...142.5	0.75	°C
Considering the shut-off time for re-start calculation					
IP_FAC_REST	8x8	0...FFH	0...1.99215	0.0078123	-
LDP_T_AST_STOP_ IP_FAC_REST	8	0...FFFFH	0...6553.5	0.1	s
LDP_TCO_STOP_IP_ FAC_REST	8	0...FEH	-48...142.5	0.75	°C
Considering coolant stop temperature and after start time for re-start calculation					
IP_FAC_STALL	8	0...FFH	0...1.99215	0.0078123	-
LDPM_TCO_1_FMSP	8	0...FEH	-48...142.5	0.75	°C
Special cold start reduction factor in case of re-start after engine stall out					
C_T_AST_STOP_MAX_REST	1	0...FFFFH	0...6553.5	0.1	s
Maximum for the time after start at shut down for restart detection					
C_TCO_DIF_MAX_REST	1	0...FEH	-48...142.5	0.75	°C
Maximal difference between at shut down stored and actual coolant temperature for restart detection					

7.3.4 Restart function (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_ES_REST	V/O	0...FFFFH	0...65535	1	[min]
Engine-off-time for the restart calculation					
LV_T_ES_REST_VLD	V/O	0...1H	0...1	1	[-]
Flag indicating that the engine-off-time for the restart calculation is valid					

Input data:

LV_T_ES_NOT_PLAUS	T_ES		

FUNCTION DESCRIPTION:

General information:

This function is used to decide whether the valid engine off time is available or not.

To avoid wrong calculation at unplausible T_ES the bit LV_T_ES_REST_VLD has to be set to zero to tell the restart function to use the standard functionality with the comparison of TCO_STOP (or TCO_REST) and TCO.

Application conditions:

Initialisation: *at reset:*
LV_T_ES_REST_VLD = 0
T_ES_REST = 65535 min

Activation: *LV_ES = 1*

Deactivation: *LV_ES = 0*

Recurrence: 10 ms

Formula section:

If LV_T_ES_NOT_PLAUS = 0

Then LV_T_ES_REST_VLD = 1

T_ES_REST = T_ES

Else LV_T_ES_REST_VLD = 0

T_ES_REST = 65535 min

Endif

7.3.4.1 Ambient pressure correction

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_ST_AMP	V/O	0...FFH	0...1.9922	0.0078	-
Injection correction at start upon AMP					

Input data:

AMP	LV_ST_END		

FUNCTION DESCRIPTION:

General information:

The purpose of this strategy is to correct the injection time depending on the ambient pressure before the MAF (manifold air pressure) variable is available.

Application conditions:

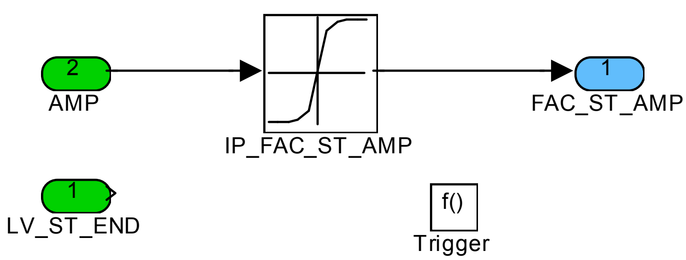
Activation: LV_ST_END =0

Deactivation: -

Initialization: -

Recurrence: 10 ms

Signal flow diagramm:



Formula section:

FAC_ST_AMP = IP_FAC_ST_AMP

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_ST_AMP	6	0...FFH	0...1.9922	0.0078	-
LDP_AMP_IP_FAC_ST_AMP	6	0...FFFFH	0...5434	0.083	hPa
Cranking injection time correction upon ambient pressure					

7.3.5 Basic injection mass for homogeneous mode

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_BAS	V/O	0...FFFFH	0...1389	0.02119	mg/stk
Basic injection fuel mass flow (homogenious charge)					

Input data:

MAF	LV_ST_END		

FUNCTION DESCRIPTION:

General information:

The basic injected fuel mass MFF_BAS in homogeneous mode is calculated out of the stoichiometric factor C_MFF_FAC and the air mass per stroke MAF. It is the required fuel mass to run the engine at lambda 1.

As soon as the engine is running (LV_ES = 0) MFF_BAS is calculated. During start this is not required, but this inhibits a peak in the wallfilm compensation due to its activation.

Application condition:

Activation: every engine state

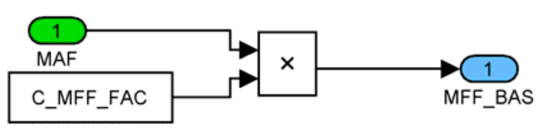
Deactivation: -

Initialization: -

Recurrance: LV_ST_END = 0: 10 ms

LV_ST_END = 1: segment synchronous

Signal flow diagramm:



Formula section:

MFF_BAS = C_MFF_FAC * MAF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MFF_FAC	1	0...FFFFH	0..0.125	1.9074E-6	-
Constant for stoichiometric air/fuel ratio (=1/14.7)					

7.4 Injector dead time corrections

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_ADD_DLY_1_HOM	V/O	8000...7FFFH	-131.07...131.07	0.004	ms
Injector dead time correction, MPI engines					
TI_ADD_DLY_TEMP_COR	V	8000...7FFFH	-131.07...131.07	0.004	ms
Injector dead time correction, MPI engines					

Input data:

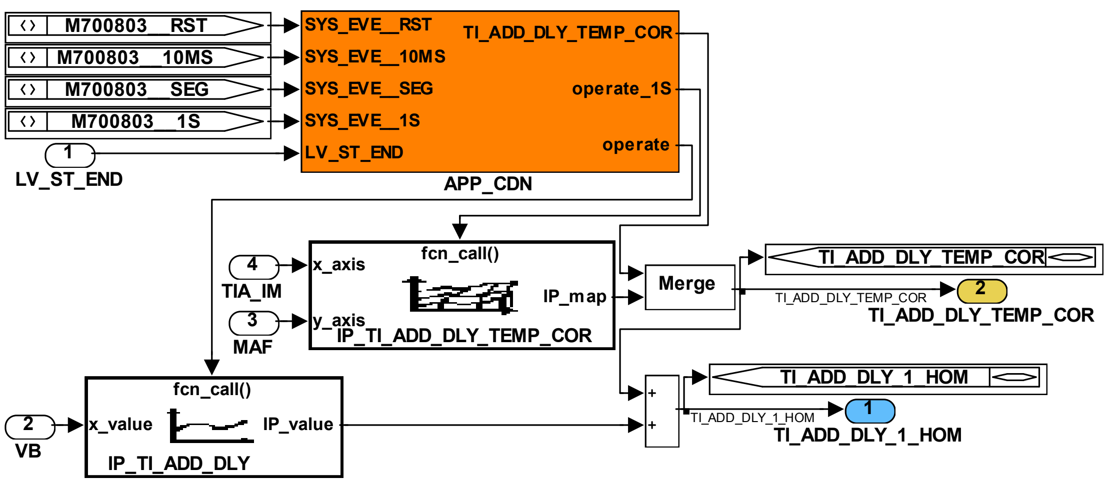
VB	LV_ST_END	MAF	TIA_IM

FUNCTION DESCRIPTION:

General information:

The module calculates the offset of the characteristic line of the injector. The calculation depends on battery voltage VB, intake air temperature TIA and mass air flow MAF.

Signal flow diagram:



7.4.1 Segment synchronous tasks

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: -

Recurrence: LV_ST_END = 0: 10 ms

LV_ST_END = 1: segment synchronous

Formula section:

TI_ADD_DLY_1_HOM = IP_TI_ADD_DLY + TI_ADD_DLY_TEMP_COR

7.4.2 Time synchronous tasks

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: at reset: TI_ADD_DLY_TEMP_COR = 0

Recurrence: 1000 ms

Formula section:

TI_ADD_DLY_TEMP_COR = IP_TI_ADD_DLY_TEMP_COR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TI_ADD_DLY	8	0...FFFFH	-131.07...131.07	0.004	ms
LDPM_VB_1_INJR	8	0...FFH	0...26	0.102	V
Injector dead time correction					
IP_TI_ADD_DLY_TEMP_COR	8*4	0...FFFFH	-131.07...131.07	0.004	ms
LDP_TIA_IM_ IP_TI_ADD_DLY_TEMP	8	0...FEH	-48...142.5	0.75	°C
LDP_MAF_IP_TI_ADD_DLY_TEMP_COR	4	0...FFFFH	0...1389	0.0212	mg/stk
Injector dead time correction vs. TIA and MAF					

7.5 Cold post start correction

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ADD_MFF_CAST	V/O	0...FFFFH	0...3.999939	6.1035e-5	-
Cold post start injection time correction					
ADD_MFF_CAST_INI	-	0...FFH	0...3.984375	0.015625	-
Initialization value of post start enrichment factor					
LV_AST	V/O	0..1H	0...1	1	-
Flag to indicate that after-start function is active					
LV_AST_END	V/O	0..1H	0...1	1	-
Flag to indicate the end of after-start function					
CYC_PUC_WUP	V	0...FFFFH	0...65535	1	-
Cycle counter for PUC operation					
CYC_WUP	V	0...FFFFH	0...65535	1	-
Cycle counter for the deactivation of the cold post start correction					

Input data:

TCO	TIA_IM	CYC_CAST	LV_PUC
LV_AT	LV_ES	LV_ST_END	AMP
N_32	MAP	FAC_ST_REST	LV_ERR_PORT
CONF_PORT	PORT_AV	FAC_FQ_ST_AD	

FUNCTION DESCRIPTION:

General information:

This enrichment takes place just after start in order to ensure the first combustions in cold conditions. CYC_CAST and CYC_WUP counts the number of segments beginning with after-start (LV_ST_END = 1, LV_AST = 1). But CYC_WUP can be stopped or decreased in case of PUC operation dependend on the calibration of C_FAC_DEC_WUP_PUC.

The value ADD_MFF_CAST_INI is determined from IP_ADD_MFF_CAST_INI (or IP_ADD_MFF_CAST_INI_AT in case of an automatic transmission) only once if the function is activated. IP_ADD_MFF_DEAC_CAST is the deactivation factor updated every segment.

In case that the port flap is activated (closed) a different deactivation factor is applied.

Application conditions:

Activation: LV_ST_END = 1 **and** LV_AST_END = 0

Deactivation: -

Initialization: at reset **or** at transition LV_ES 0 ->1:
LV_AST = 0; LV_AST_END = 0; ADD_MFF_CAST = 0; CYC_PUC_WUP = 0

at activation:
LV_AST = 1
If LV_AT = 1
then ADD_MFF_CAST_INI = IP_ADD_MFF_CAST_INI_AT * IP_FAC_REST_CAST * IP_FAC_AMP_CAST
else ADD_MFF_CAST_INI = IP_ADD_MFF_CAST_INI * IP_FAC_REST_CAST * IP_FAC_AMP_CAST

Recurrence: segment synchronous

Formula section:

If LV_PUC = 1

then $CYC\_PUC\_WUP_n = CYC\_PUC\_WUP_{n-1} + 1$

CYC_WUP = CYC_CAST – C_FAC_DEC_WUP_PUC * CYC_PUC_WUP

endif

If CONF_PORT = 1 **and** LV_ERR_PORT = 0
and PORT_AV >= C_PORT_AV_THD_MFF_CAST

then ADD_MFF_CAST = ADD_MFF_CAST_INI * IP_ADD_MFF_DEAC_CAST_PORT * (1 + (FAC_FQ_ST_AD – 1) * C_FAC_FQ_ST_AD_AST)

else

ADD_MFF_CAST = ADD_MFF_CAST_INI * IP_ADD_MFF_DEAC_CAST * (1 + (FAC_FQ_ST_AD – 1) * C_FAC_FQ_ST_AD_AST)**endif**

The value ADD_MFF_CAST_INI is determined from IP_ADD_MFF_CAST_INI (or IP_ADD_MFF_CAST_INI_AT in case of an automatic transmission) only once if the function is activated.

IP_ADD_MFF_DEAC_CAST is the deactivation factor updated every segment.

If ADD_MFF_CAST = 0(0H)

then LV_AST = 0

LV_AST_END_= 1

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_ADD_MFF_CAST_INI	4*12	0...FFH	0...3.984375	0.015625	-
LDP_TIA_IM_IP_ADD_MFF_CAST_INI	4	0...FEH	-48..142.5	0.75	°C
LDPM_TCO_2_FMSP	12	0...FEH	-48..142.5	0.75	°C
Initialization value of post start enrichment factor					
IP_ADD_MFF_CAST_INI_AT	4*12	0...FFH	0...3.984375	0.015625	-
LDP_TIA_IM_IP_ADD_MFF_CAST_AT	4	0...FEH	-48..142.5	0.75	°C
LDPM_TCO_2_FMSP	12	0...FEH	-48..142.5	0.75	°C
Initialization value of post start enrichment factor for automatic transmission					
IP_ADD_MFF_DEAC_CAST	12*12	0...FFH	0...1.9921875	0.0078125	-
LDP_CYC_WUP_IP_ADD_MFF_DEAC	12	0...FFFFH	0..65535	1	-
LDPM_TCO_2_FMSP	12	0...FEH	-48..142.5	0.75	°C
Deactivation factor of post start enrichment					
C_FAC_DEC_WUP_PUC	1	0...FFH	0...1.9921875	0.0078125	-
Factor to reduce the cold post start correction deactivation during PUC					
IP_FAC_REST_CAST	6*8	0...FFH	0...1.9921875	0.0078125	-
LDP_FAC_ST_REST_IP_REST_CAST	6	0...FFH	0...1.992	0.0078	-
LDP_TCO_IP_REST_CAST	8	0...FEH	-48..142.5	0.75	°C
Weighted restart factor for after start correction					
IP_FAC_AMP_CAST	8*6	0...FFH	0...1.9921875	0.0078125	-
LDP_AMP_ IP_FAC_AMP_CAST	8	0...FFFFH	0...5434	0.083	hPa
LDP_TCO_IP_FAC_AMP_CAST	6	0...FEH	-48..142.5	0.75	°C
Weighted AMP factor for after start correction					
C_PORT_AV_THD_MFF_CAST	1	F000H...1000H	-100...100	0.0244	°PORT
PORT_AV treshold to enable the port flap-cold post start correction					
IP_ADD_MFF_DEAC_CAST_PORT	12*12	0...FFH	0...1.9921875	0.0078125	-
LDP_CYC_WUP_IP_ADD_MFF_DEAC_PO RT	12	0...FFFFH	0..65535	1	-
LDPM_TCO_2_FMSP	12	0...FEH	-48..142.5	0.75	°C
Deactivation factor of post start enrichment in case of active port flap					
C_FAC_FQ_ST_AD_AST	1	0...FFH	0...3.984735	0.0157	-
Weighting factor of the fuel quality adaptation for the cold after start correction					

7.6 Warm-up correction

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_MFF_WUP	V/O	0...FFFFH	0...3,9999	6.1036e-5	-
Warm-up correction factor					
FAC_MFF_TCO_WUP	V	0...FFH	0...3.984375	0.015625	-
Deactivation factor for warm up correction					
LV_WUP	V/O	0...1H	0...1	1	-
Flag to indicate that a warm-up correction is active					
LV_WUP_END	V	0...1H	0...1	1	-
Flag to indicate the end of the warm-up correction phase					

Input data:

N_32	MAF	TCO	TCO_ST
LV_ES	LV_ST_END	LV_AUTH_TQ_PAT	AMP
FAC_FQ_ST_AD			

FUNCTION DESCRIPTION:

General information:

The fuel mass setpoint is corrected after start in order to meet the increased fuel requirements for a cold engine.

The influence of the altitude and fuel shut-off due to TCS or GS is taken into consideration.

Application conditions:

Activation: LV_ST_END = 1 **and** LV_WUP_END = 0

Deactivation: -

Recurrence: segment synchronous

Initialization: at reset **or** at transition LV_ES 0 -> 1:

LV_WUP = 0; LV_WUP_END = 0; FAC_MFF_WUP = 1

at activation:

LV_WUP = 1

Formula section:

FAC_MFF_TCO_WUP = IP_FAC_MFF_TCO_WUP

If LV_AUTH_TQ_PAT = 0 (no fuel shut off active due to TCS)

Then FAC_MFF_WUP = 1 + IP_FAC_MFF_WUP
* FAC_MFF_TCO_WUP
* IP_FAC_MFF_AMP_WUP
* (1 + (FAC_FQ_ST_AD – 1) * C_FAC_FQ_ST_AD_WUP)

else (Fuel shut off active due to TCS)

FAC_MFF_WUP = 1 + IP_FAC_MFF_WUP
* FAC_MFF_TCO_WUP
* IP_FAC_MFF_AMP_WUP
* IP_FAC_MFF_ASR_WUP
* (1 + (FAC_FQ_ST_AD – 1) * C_FAC_FQ_ST_AD_WUP)

If FAC_MFF_TCO_WUP = 0 (0H) **then**
LV_WUP = 0
LV_WUP_END = 1
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_MFF_WUP	8 * 8	0H...FFFFH	-0.99999 ...0.99996	3.05175e-5	-
LDPM_N_32_1_FMSP	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_FMSP	8	0...FFH	0...1389	5.4470	mg/stk
Warm-up factor versus working point					
IP_FAC_MFF_TCO_WUP	8 * 8	0...FFH	0...3.984375	0.015625	-
LDP_TCO_IP_FAC_MFF_TCO_WUP	8	0...FEH	-48...142.5	0.75	°C
LDP_TCO_ST_IP_FAC_MFF_TCO_WUP	8	0...FEH	-48...142.5	0.75	°C
Warm-up enrichment deactivation factor					
IP_FAC_MFF_AMP_WUP	8 * 6	0...FFH	0...3.984375	0.015625	-
LDP_AMP_IP_FAC_MFF_AMP_WUP	8	0...FFFFH	0...5434	0.083	hPa
LDP_TCO_ST_IP_FAC_MFF_AMP_WUP	6	0...FEH	-48...142.5	0.75	°C
Weighted AMP factor for warm-up correction					
IP_FAC_MFF_ASR_WUP	6 * 6	0...FFH	0...3.984375	0.015625	-
LDP_TCO_IP_FAC_MFF_ASR_WUP	6	0...FEH	-48...142.5	0.75	°C
LDP_TCO_ST_IP_FAC_MFF_ASR_WUP	6	0...FEH	-48...142.5	0.75	°C
Weighted ASR factor for warm-up correction					
C_FAC_FQ_ST_AD_WUP	1	0...FFH	0...3.984735	0.0157	-
Weighting factor of the fuel quality adaptation for the warm up correction					

7.7 Injection phase for homogeneous mode, 1. pulse

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EOI_1_HOM[NC_CYL_NR]	V/O	0 ... 780H	0 ... 720	0.375	°CRK
End of injection of the first injection for homogeneous mode					
SOI_1_HOM	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Start of injection of the first injection for homogeneous mode					
EOI_MIN_HOM	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Latest possible EOI at homogeneouse mode					
SOI_MAX	V/O	0 ... 780H	0 ... 720	0.375	°CRK
Earliest possible SOI					
LV_CRK_DIF_SOI_EOI_MAX	V/O	0 ... 1H	0 ... 1	1	-
Difference between SOI and EOI is beyond the maximum					
CRK_DIF_SOI_IGN	O	0 ... F00H	0 ... 1440	0.375	°CRK
Output variable for LACO					
EOI_LIM[NC_CYL_NR]	O	F100 ... 0F00H	-1440 ... 1440	0.375	°CRK
Cylinder individual zero position for phasing					
SOI_LIM[NC_CYL_NR]	O	F100 ... 0F00H	-1440 ... 1440	0.375	°CRK
Cylinder individual end position for phasing					
EOI_1_BAS	V	0 ... 780H	0 ... 720	0.375	°CRK
Basic value for end of the first injection					
EOI_1_TCO	V	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
Temperature correction for end of the first injection					
EOI_AST	V	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
EOI offset at after start					
#if (NC_EOI_CST_CYC_ST_ENA = 1)					
EOI_1_TCO_COR	-	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
Coolant temperature correction for EOI					
FAC_EOI_1_TCO_COR	-	0 ... FFH	0 ... 0.996094	3.90625e-3	-
Weighting factor for TCO correction of EOI					
#endif					

Input data:

N_32	FAC_N	TCO	FUP
NC_CYL_NR	TI_1_HOM[NC_CYL_NR]	EOI_1_HOM_COR_EXT	REF_EOI_LIM
CYC_CAST	TI_1_HOM_CLC_MAX	NC_EOI_CST_CYC_ST_ENA	NC_EOI_LIM
LV_CT	LV_ST_END	CTR_CYL_NR_ST_CLC_INJR	N_FAST_INJ
CYC_ST	CAM_SHIFT_IN	CTR_CYL_NR_STOP_CLC_INJR	

FUNCTION DESCRIPTION:

7.7.1 Definition of the injection range

Application conditions:

Activation: at reset

Deactivation: -

Initialization: -

Recurrence: once at reset

The fuel can be injected within a defined window over a range of 720 °CRK. The cylinder individual zero position of this range is defined via the constants NC_REF_EOI_LIM and NC_EOI_LIM with reference to the ignition TDC of cylinder 0, see figure below.

Formula section:

Related to tdc cylinder 0:

FOR x = 0 **TO** (NC_CYL_NR-1) **DO**:

EOI_LIM[x] = REF_EOI_LIM + NC_EOI_LIM - x * (720 / NC_CYL_NR)

SOI_LIM[x] = EOI_LIM[x] + 720 °CRK

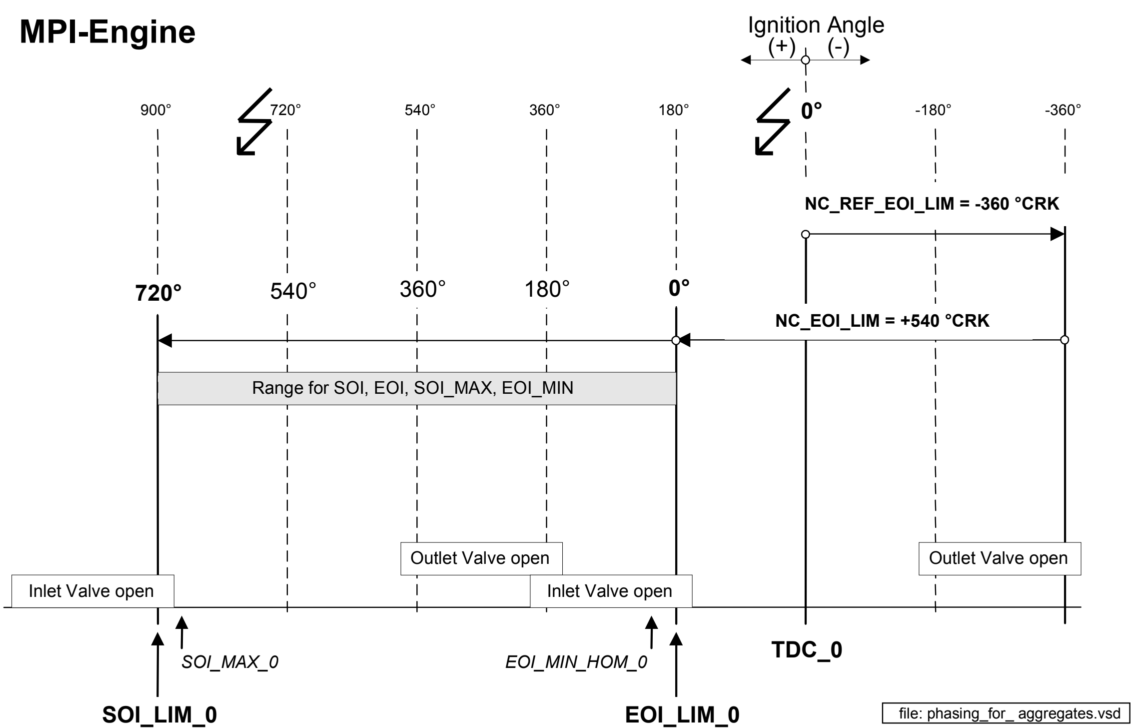
ENDFOR

Definition of the non calibrateable constants:

(NC_REF_EOI_LIM = -360°CRK, defined in I/O SW, assigned to REF_EOI_LIM)

NC_EOI_LIM = 540 °CRK

MPI-Engine



7.7.2 Injection phasing for preinjection

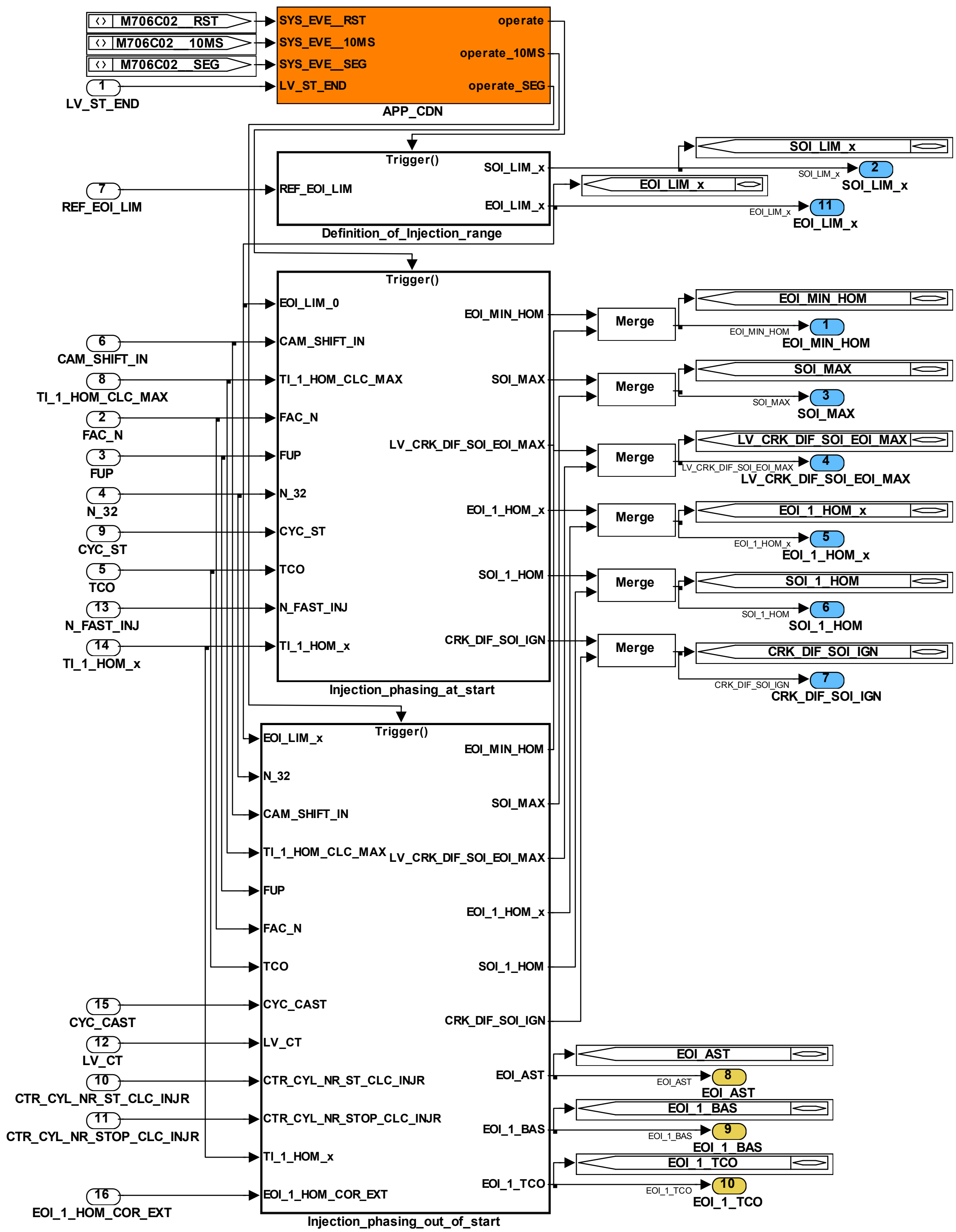
Preinjection is only applied on MPI engines, see Specification “Configuration of preinjection”.

The pre injection function has a separate interface to the basic software. As long as the pre injection function is active the basic software drives the injectors via this interface.

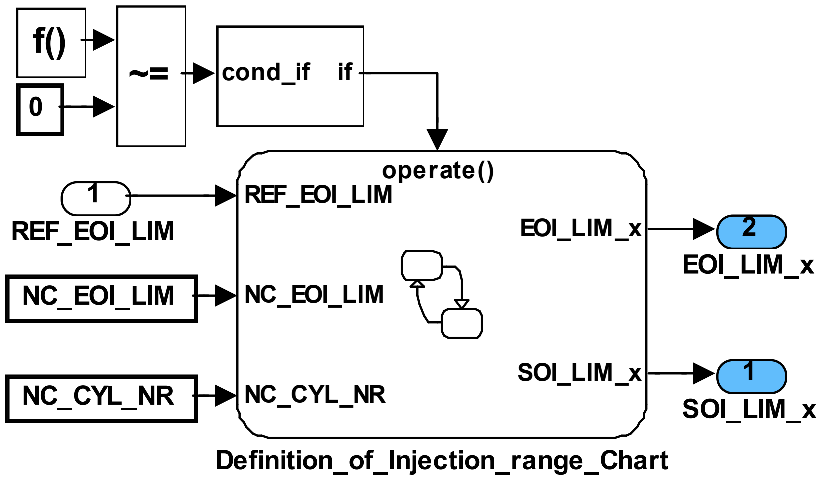
The function is activated after at first valid tooth and is deactivated after a defined number of preinjections are performed.

After the preinjection is finished the injection is performed as described in the following chapters:

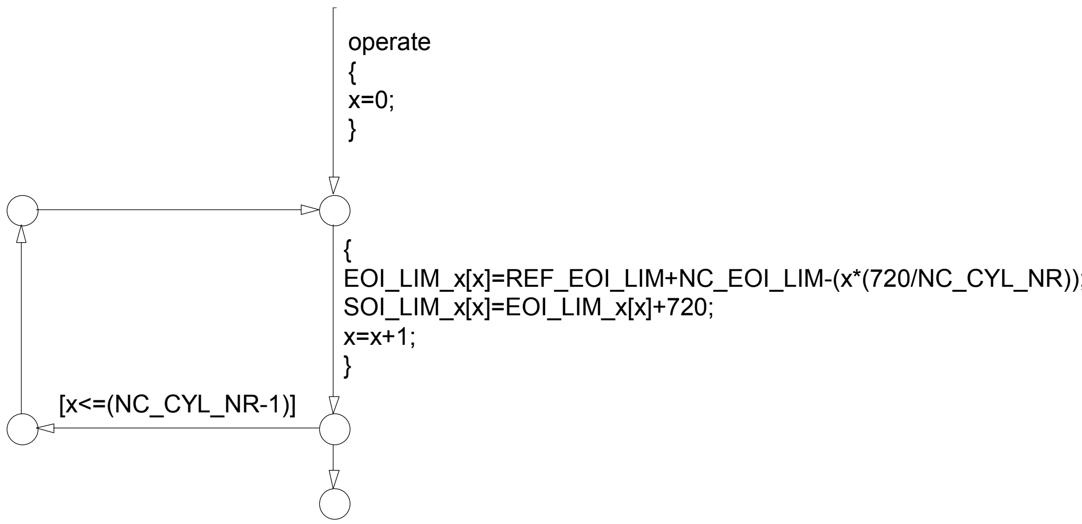
Main system:


M706C02__RST
M706C02__10MS
M706C02__SEG
SYS_EVE__RST
SYS_EVE__10MS
SYS_EVE__SEG
LV_ST_END
operate
operate_10MS
operate_SEG
APP_CDN
1
LV_ST_END
Trigger()
REF_EOI_LIM
SOI_LIM_x
EOI_LIM_x
Definition_of_Injection_range
7
REF_EOI_LIM
SOI_LIM_x
EOI_LIM_x
2
SOI_LIM_x
11
EOI_LIM_x
EOI_LIM_0
CAM_SHIFT_IN
TI_1_HOM_CLC_MAX
FAC_N
FUP
N_32
CYC_ST
TCO
N_FAST_INJ
TI_1_HOM_x
EOI_MIN_HOM
SOI_MAX
LV_CRK_DIF_SOI_EOI_MAX
EOI_1_HOM_x
SOI_1_HOM
CRK_DIF_SOI_IGN
Injection_phasing_at_start
6
CAM_SHIFT_IN
8
TI_1_HOM_CLC_MAX
2
FAC_N
3
FUP
4
N_32
9
CYC_ST
5
TCO
13
N_FAST_INJ
14
TI_1_HOM_x
Merge
EOI_MIN_HOM
1
EOI_MIN_HOM
SOI_MAX
3
SOI_MAX
LV_CRK_DIF_SOI_EOI_MAX
4
LV_CRK_DIF_SOI_EOI_MAX
EOI_1_HOM_x
5
EOI_1_HOM_x
SOI_1_HOM
6
SOI_1_HOM
CRK_DIF_SOI_IGN
7
CRK_DIF_SOI_IGN
EOI_LIM_x
N_32
CAM_SHIFT_IN
TI_1_HOM_CLC_MAX
FUP
FAC_N
TCO
CYC_CAST
LV_CT
CTR_CYL_NR_ST_CLC_INJR
CTR_CYL_NR_STOP_CLC_INJR
TI_1_HOM_x
EOI_1_HOM_COR_EXT
EOI_AST
EOI_1_BAS
EOI_1_TCO
Injection_phasing_out_of_start
15
CYC_CAST
12
LV_CT
10
CTR_CYL_NR_ST_CLC_INJR
11
CTR_CYL_NR_STOP_CLC_INJR
16
EOI_1_HOM_COR_EXT
8
EOI_AST
9
EOI_1_BAS
10
EOI_1_TCO

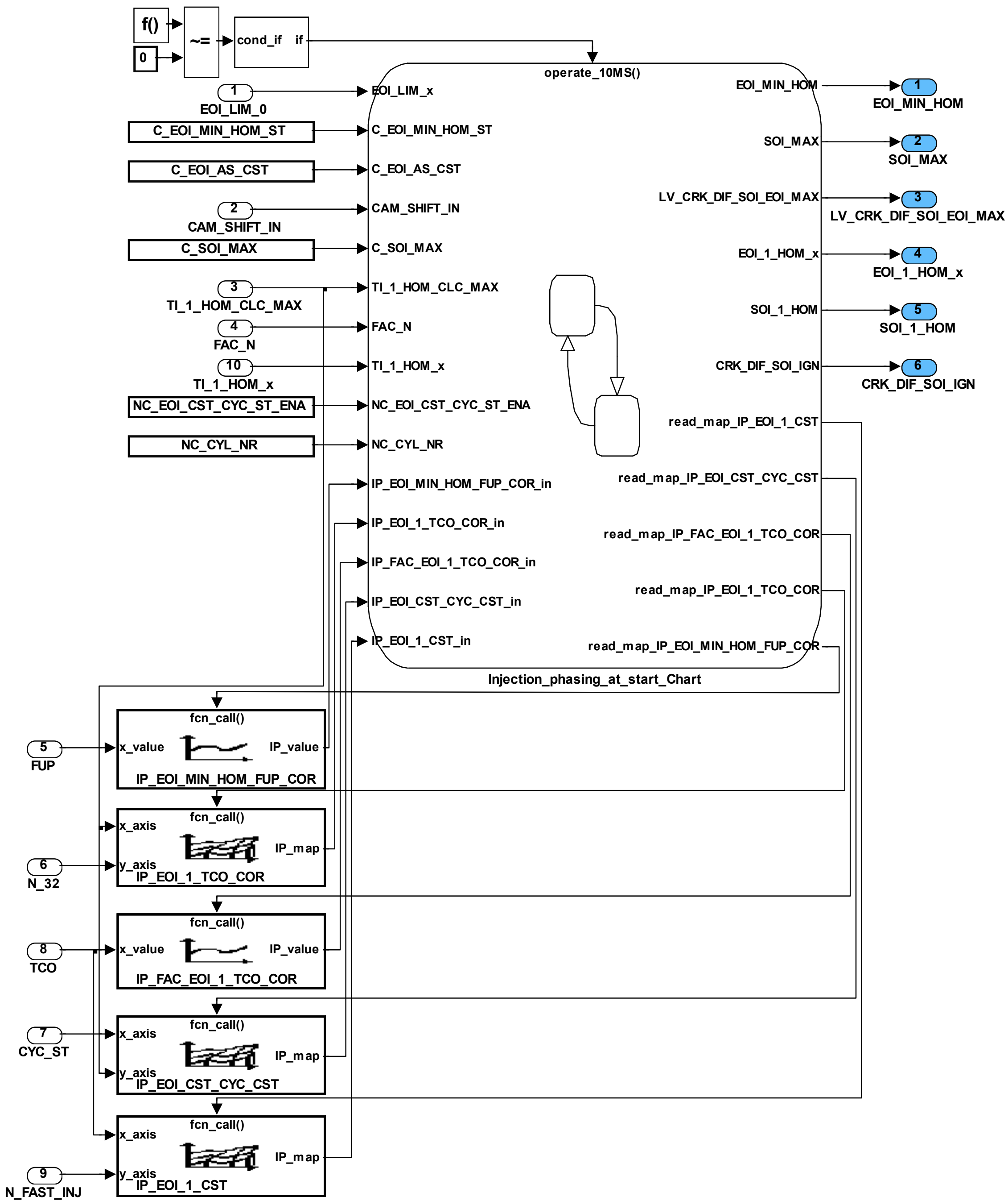
Definition_of_Injection_range:



Definition_of_Injection_range_Chart:

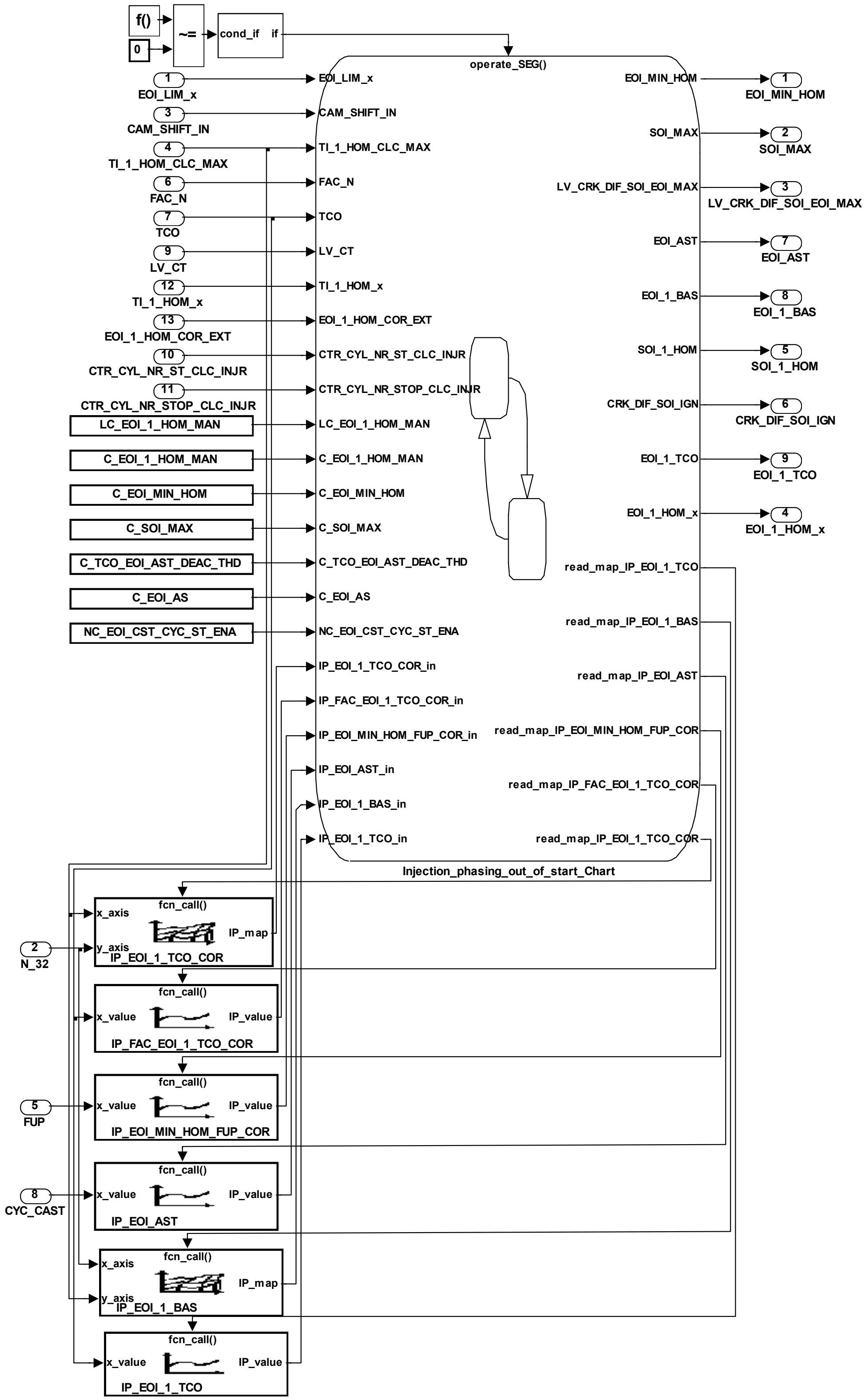


Injection_phasing_at_start:

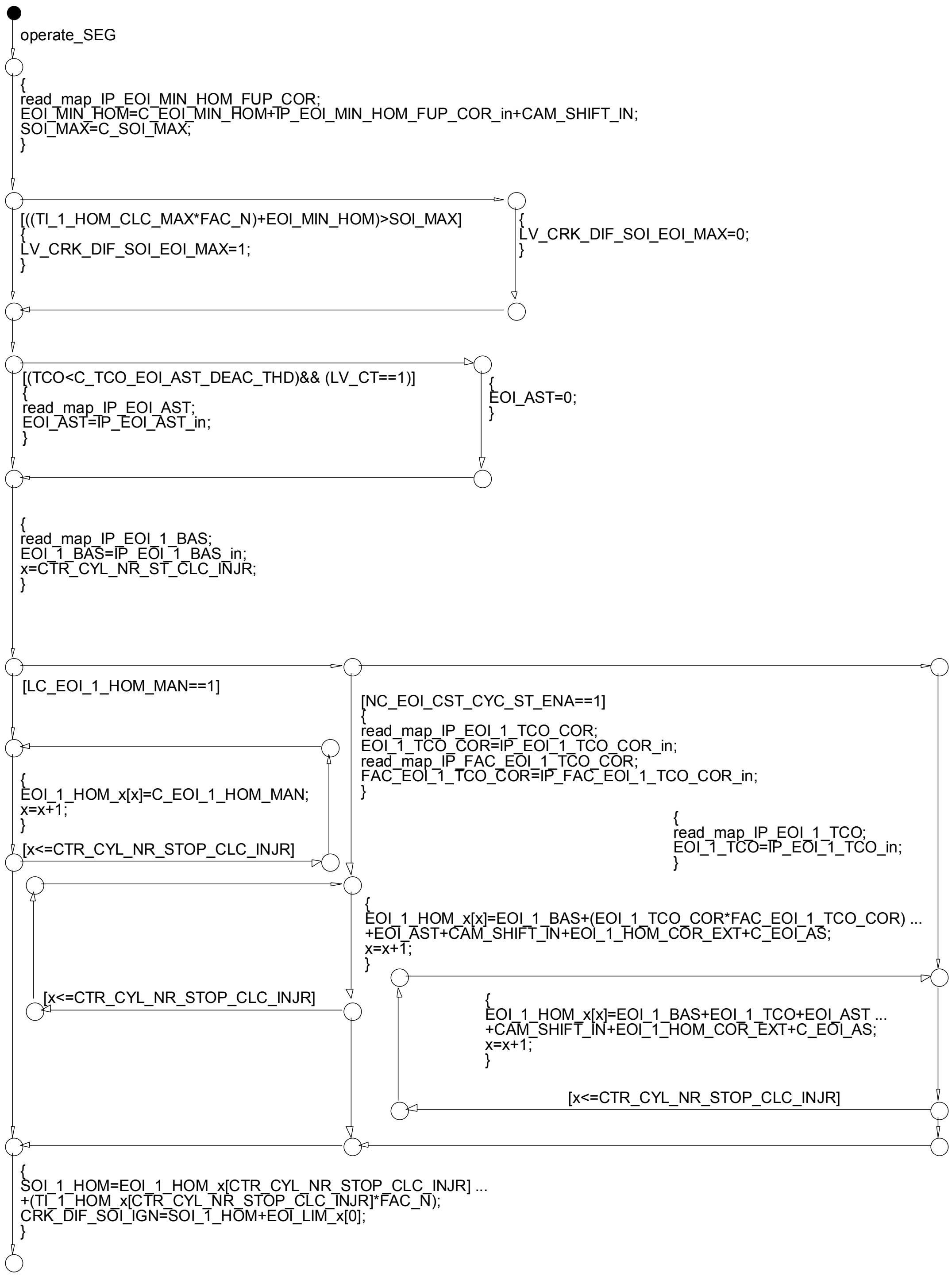


Injection_phasing_at_start_Chart:

Injection_phasing_out_of_start:



Injection_phasing_out_of_start_Chart:



Formula section:

7.7.3 Injection phasing at start

Application conditions:

Activation: LV_ST_END = 0

Deactivation: -

Initialization: -

Recurrence: 10ms

Formula section:

All calculations are related to EOI_LIM[x].

EOI_MIN_HOM = C_EOI_MIN_HOM_ST
+ IP_EOI_MIN_HOM_FUP_COR
+ CAM_SHIFT_IN

SOI_MAX = C_SOI_MAX

Check if the difference between SOI and EOI is beyond the maximum.

(1) IF (TI_1_HOM_CLC_MAX * FAC_N + EOI_MIN_HOM) > SOI_MAX

(1) THEN

LV_CRK_DIF_SOI_EOI_MAX = 1 (TI is beyond the maximum)

(1) ELSE

LV_CRK_DIF_SOI_EOI_MAX = 0 (TI is below the maximum)

(1) ENDIF

At start the injection phase is defined via EOI.

#IF NC_EOI_CST_CYC_ST_ENA = 1 (start method via CYC_ST)

#THEN

EOI_1_TCO_COR = IP_EOI_1_TCO_COR

FAC_EOI_1_TCO_COR = IP_FAC_EOI_1_TCO_COR

FOR x = 0 **TO** (NC_CYL_NR-1) **DO**:

EOI_1_HOM[x] = IP_EOI_CST_CYC_CST
+ EOI_1_TCO_COR * FAC_EOI_1_TCO_COR + C_EOI_AS_CST

ENDFOR

#ELSE

FOR x = 0 **TO** (NC_CYL_NR-1) **DO**:

EOI_1_HOM[x] = IP_EOI_1_CST + C_EOI_AS_CST

ENDFOR

#ENDIF

SOI_1_HOM = EOI_1_HOM[NC_CYL_NR-1] + TI_1_HOM[NC_CYL_NR-1] * FAC_N

Output variable for LACO:

CRK_DIF_SOI_IGN = SOI_1_HOM + EOI_LIM_0

7.7.4 Injection phasing out of start

Out of start the injection phase is defined via EOI.

Application conditions:

Activation:	LV_ST_END = 1
Deactivation:	-
Initialization:	-
Recurrence:	segment synchronous

Formula section:

All calculations are related to EOI_LIM[x].

EOI_MIN_HOM = C_EOI_MIN_HOM
+ IP_EOI_MIN_HOM_FUP_COR
+ CAM_SHIFT_IN

SOI_MAX = C_SOI_MAX

Check it the difference between SOI and EOI is beyond the maximum.

(1) IF (TI_1_HOM_CLC_MAX * FAC_N + EOI_MIN_HOM) > SOI_MAX

(1) THEN

LV_CRK_DIF_SOI_EOI_MAX = 1 (TI is beyond the maximum)

(1) ELSE

LV_CRK_DIF_SOI_EOI_MAX = 0 (TI is below the maximum)

(1) ENDIF

(2) IF

TCO < C_TCO_EOI_AST_DEAC_THD

AND

LV_CT = 1

(2) THEN

EOI_AST = IP_EOI_AST .... EOI offset at after start

(2) ELSE

EOI_AST = 0

(2) ENDIF

EOI_1_BAS = IP_EOI_1_BAS(N32, TI_1_HOM_CLC_MAX) .... basic map

Note: At the interface EOI_1_HOM_COR_EXT either a customer specific functionality is plugged in to correct EOI or it has to be initialized in the initialization module.

Manual setpoint of EOI:

(3) IF LC_EOI_1_HOM_MAN = 1 (Manual setpoint of EOI)

(3) THEN

FOR x = CTR_CYL_NR_ST_CLC_INJR **TO** CTR_CYL_NR_STOP_CLC_INJR **DO**:

EOI_1_HOM[x] = C_EOI_1_HOM_MAN

ENDFOR

(3) ELSE

#IF NC_EOI_CST_CYC_ST_ENA = 1

#THEN

EOI_1_TCO_COR = IP_EOI_1_TCO_COR ... TCO correction

FAC_EOI_1_TCO_COR = IP_FAC_EOI_1_TCO_COR ... Weighting factor for TCO correction

FOR x = CTR_CYL_NR_ST_CLC_INJR **TO** CTR_CYL_NR_STOP_CLC_INJR **DO**:

EOI_1_HOM[x] = EOI_1_BAS + EOI_1_TCO_COR * FAC_EOI_1_TCO_COR

+ EOI_AST

+ CAM_SHIFT_IN .... inlet CAM-position correction

+ EOI_1_HOM_COR_EXT .... interface for external input

+ C_EOI_AS .... correction by application system

ENDFOR

#ELSE

EOI_1_TCO = IP_EOI_1_TCO(TCO) .... TCO correction

FOR x = CTR_CYL_NR_ST_CLC_INJR **TO** CTR_CYL_NR_STOP_CLC_INJR **DO**:

EOI_1_HOM[x] = EOI_1_BAS + EOI_1_TCO + EOI_AST
+ CAM_SHIFT_IN .... inlet CAM-position correction
+ EOI_1_HOM_COR_EXT .... interface for external input
+ C_EOI_AS .... correction by application system

ENDFOR

#ENDIF

(3) ENDIF

SOI_1_HOM = EOI_1_HOM[CTR_CYL_NR_STOP_CLC_INJR]
+ TI_1_HOM[CTR_CYL_NR_STOP_CLC_INJR] * FAC_N

<u>Output variable for LACO:</u>

CRK_DIF_SOI_IGN = SOI_1_HOM + EOI_LIM_0

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_EOI_1_BAS	8*6	0 ... 780H	0 ... 720	0.375	°CRK
LDPM_N_32_1_INJR	8	0 ... FFH	0 ... 8160	32	1/min
LDPM_TI_1_HOM_MAX_1_INJR	6	0 ... FFFFH	0 ... 262.14	0.004	ms
Basic map end of the first injection					
IP_EOI_CST_CYC_CST	8*6	0 ... 780H	0 ... 720	0.375	°CRK
LDP_CYC_ST_IP_EOI_CST_CYC_CST	8	0 ... FFH	0 ... 255	1	-
LDPM_TI_1_HOM_MAX_1_INJR	6	0 ... FFFFH	0 ... 262.14	0.004	ms
Basic map end of the first injection for cold start					
IP_EOI_1_TCO_COR	6*4	0 ... 780H	-360 ... 360	0.375	°CRK
LDPM_TI_1_HOM_MAX_1_INJR	6	0 ... FFFFH	0 ... 262.14	0.004	ms
LDP_N_32_IP_EOI_1_TCO_COR	4	0 ... FFH	0 ... 8160	32	rpm
Coolant temperature correction for EOI					
IP_FAC_EOI_1_TCO_COR	8	0 ... FFH	0 ... 0.996094	3.90625e-3	-
LDPM_TCO_1_INJR	8	0 ... FEH	-48 ... 142.5	0.75	°C
Weighting factor for TCO correction of EOI					
IP_EOI_1_TCO	8	0 ... 780H	-360 ... 360	0.375	°CRK
LDPM_TCO_1_INJR	8	0 ... FEH	-48 ... 142.5	0.75	°C
Coolant temperature correction for EOI					
IP_EOI_1_CST	8*4	0 ... 780H	0 ... 720	0.375	°CRK
LDPM_TCO_1_INJR	8	0 ... FEH	-48 ... 142.5	0.75	°C
LDPM_N_FAST_INJ_1_INJR	4	0 ... 1FE0H	0 ... 8160	1	rpm
Basic map end of injection for cold start					
IP_EOI_MIN_HOM_FUP_COR	4	0 ... 780H	-360 ... 360	0.375	°CRK
LDP_FUP_IP_EOI_MIN_HOM	4	0 ... FFFFH	0 ... 173888	2.65	hPa
FUP correction of EOI_MIN_HOM					
IP_EOI_AST	6	0 ...780 H	-360 ... 360	0.375	°CRK
LDP_CYC_CAST_IP_EOI_AST	6	0 .... FFFFH	0 ... 65635	1	-
EOI offset at after start					
C_TCO_EOI_AST_DEAC_THD	1	0 ... FEH	-48 ... 142.5	0.75	°C
Temperature threshold for deactivation of EOI offset at after start					
C_EOI_MIN_HOM	1	0 ... 780H	0 ... 720	0.375	°CRK
Minimal possible EOI (relevant for additional pulse at transient operation)					
C_EOI_MIN_HOM_ST	1	0 ... 780H	0 ... 720	0.375	°CRK
Minimal possible EOI at start					
C_SOI_MAX	1	0 ... 780H	0 ... 720	0.375	°CRK
Earlies possible SOI					
C_EOI_AS	1	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
EOI-correction of application system					
C_EOI_AS_CST	1	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
EOI-correction of application system at start					
LC_EOI_1_HOM_MAN	1	0 ... 1 H	0 ... 1	1	-
Switch for using the manual setpoit for EOI					
C_EOI_1_HOM_MAN	1	0 ... 780H	0 ... 720	0.375	°CRK
Manual setpoit for EOI					

7.8 Injection phase (application incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAM_SHIFT_EX	O/V	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
Shift of the exhaust camshaft relative to passiv position					
CAM_SHIFT_IN	O/V	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
Shift of the inlet camshaft relative to passiv position					
EOI_1_HOM_COR_EXT	O	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
External correction of end of injection, first pulse					
EOI_2_HOM_COR_EXT	O	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
External correction of end of injection, second pulse					
SOI_1_HOM_COR_EXT	O	FC40 ... 03C0H	-360 ... 360	0.375	°CRK
External correction of start of injection, first pulse					

Input data:

LV_ST_END	C_CAM_INI_IN	C_CAM_OP_IN	CAM_AV_MV_IN
VLFT_AV			

FUNCTION DESCRIPTION:

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: -

Recurrence: LV_ST_END = 0: 10ms

LV_ST_END = 1: segment synchronous

Formula section:

CAM_SHIFT_EX = 0
CAM_SHIFT_IN = (C_CAM_INI_IN – CAM_AV_MV_IN) + (C_CAM_OP_IN – IP_CAM_OP_IN_VVL__VLFT_AV)
EOI_1_HOM_COR_EXT = 0
EOI_2_HOM_COR_EXT = 0
SOI_1_HOM_COR_EXT = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CAM_OP_IN_VVL__VLFT_AV	1*6	0 ... FFFFH	-768 ... +767.9765	0.0234375	°CRK
LDP_VLFT_AV__CAM_OP_IN_VVL	6	0 ... FFFFH	0 ... 65.535	0.001	mm
inlet valve opening angle vs. actual inlet valve lift					

7.9 Fuel mass setpoint for homogeneous mode

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_SP_HOM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1389	0.02119	mg/tdc
Mass fuel flow setpoint for homogeneous mode, bank selective					
MFF_SP_HOM_MV	V/O	0...FFFFH	0...1389	0.02119	mg/tdc
Mass fuel flow setpoint for homogeneous mode, mean value					
MFF_SP_HOM_ENG[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1389	0.02119	mg/tdc
Calculated mass fuel flow output for canister purge					
MFF_SP_HOM_FUP_CTL	O	0...FFFFH	0...1389	0.02119	mg/tdc
Mass fuel flow setpoint for fuel pressure control					
MFF_SP_CLC[NC_CBK_EX_NR]	V	0...FFFFH	0...1389	0.02119	mg/tdc
Calculated mass fuel flow setpoint for homogeneous mode					
MFF_SP_HOM_CLC[NC_CBK_EX_NR]	-	0...FFFFH	0...1389	0.02119	mg/tdc
Calculated mass fuel flow setpoint for homogeneous mode					
MFF_AST	V	0...FFFFH	0...1389	0.02119	mg/tdc
Calculated mass fuel flow for after start					
MFF_LGRD_AST	V	0...FFFFH	0...1389	0.02119	mg/tdc
Limitating gradient for after start					
LV_MFF_AST_ACT[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag to indicate that MFF_AST is applied					

Input data:

TCO	FAC_LAM_COR[NC_CBK_EX_NR]	FAC_FQ_ST_AD	
FAC_LAM_AD_OUT[NC_CBK_EX_NR]		MFF_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	
FAC_MFF_WUP	MFF_ADD_WF	MFF_CST	MFF_ADD_CP
FAC_ST_REST	ADD_MFF_CAST	LV_TI_1_HOM_MIN	FAC_CMBC_WF
LV_ST_END	FAC_ST_AMP	MFF_BAS	
MFF_ADD_EXT	NC_CBK_EX_NR	FAC_MFF_EXT	FAC_MFF_BAS_EXT
FAC_MFF_ST_EXT	MFF_ADD_LAM_AD_OUT_EXT	MFF_ADD_DIAGCP	NC_WF_MFF_BAS

FUNCTION DESCRIPTION:

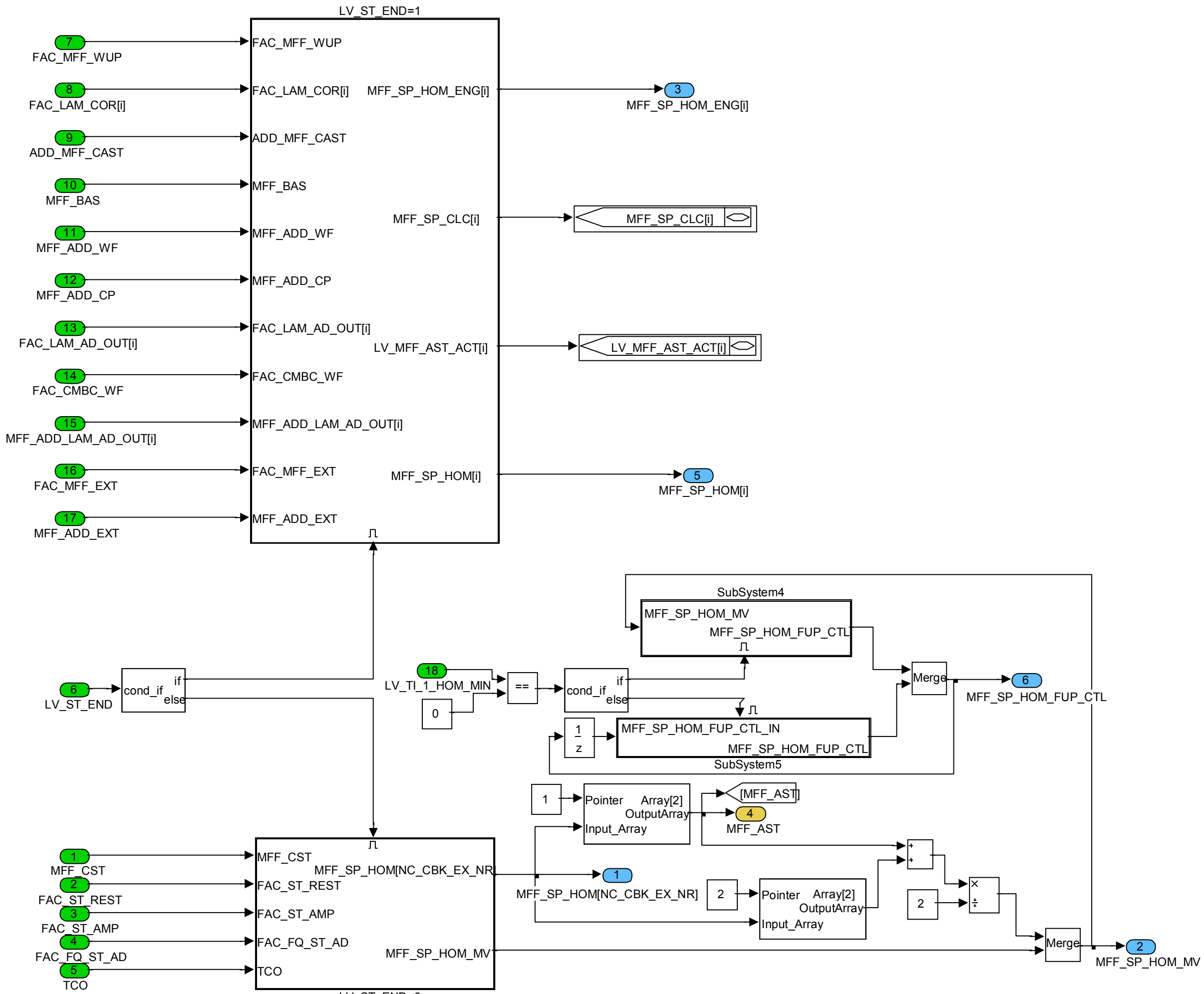
General information:

After key on for the calculation of the injected fuel mass there are different stages distinguished:

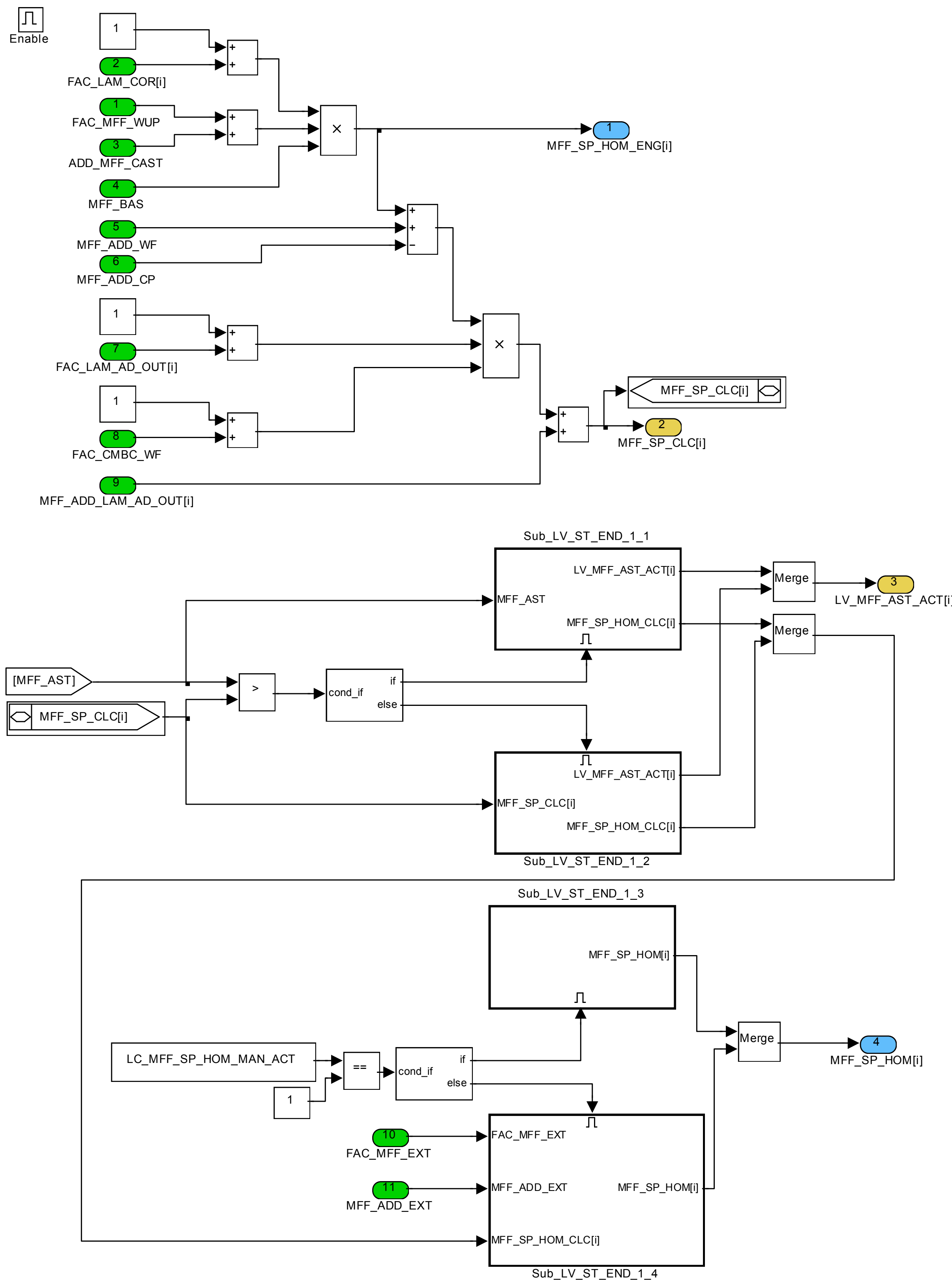
- Start pre injection
- Mass fuel flow at start
- Mass fuel flow out of start

Signal flow diagramm:

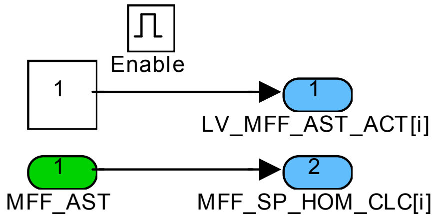
Mainsystem:



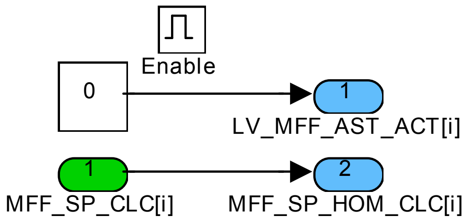
Subsystem LV_ST_END=1:



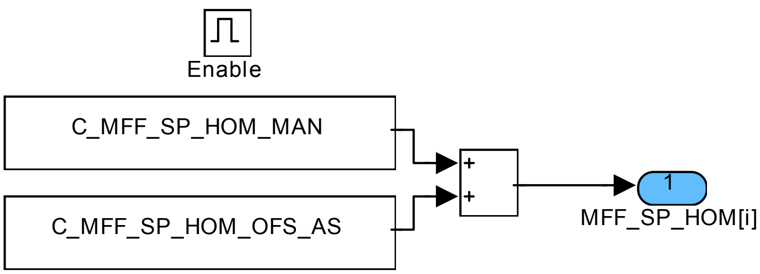
Sub_LV_ST_END_1_1:



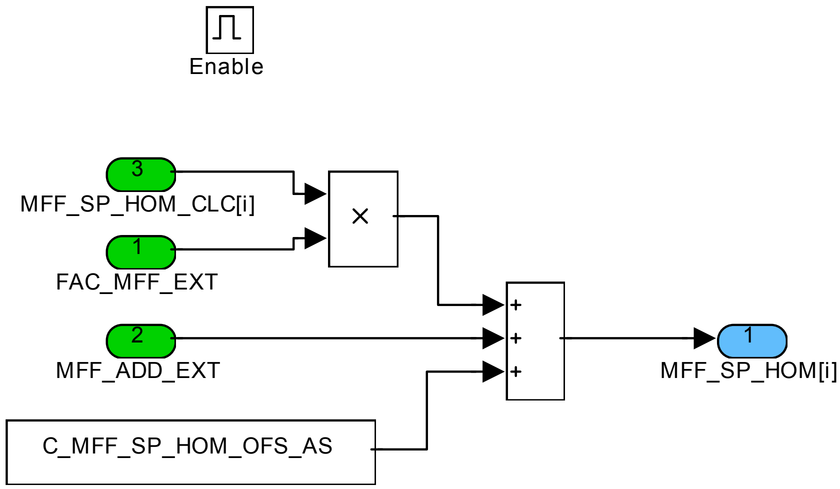
Sub_LV_ST_END_1_2:



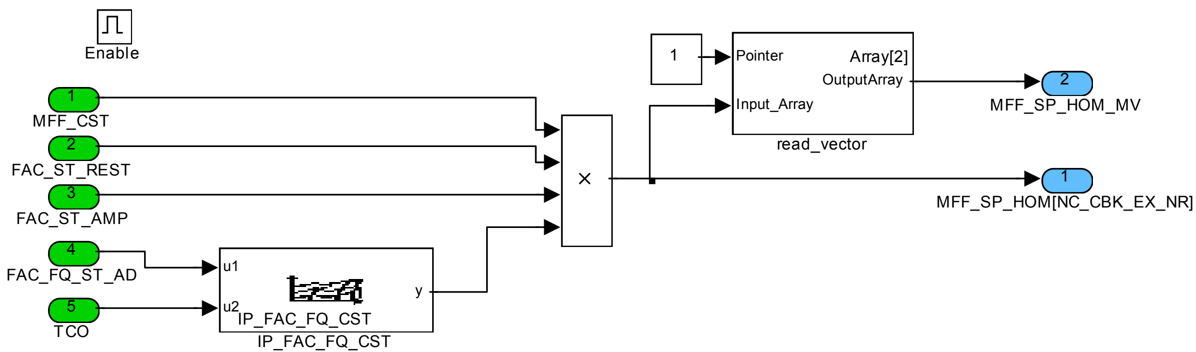
Sub_LV_ST_END_1_3:



Sub_LV_ST_END_1_4:



Subsystem LV_ST_END=0:



Start pre injection

Preinjection is only applied on MPI engines, see Specification “Start pre injection”.

The pre injection function has a separate interface to the basic software. As long as the pre injection function is active the basic software drives the injectors via this interface.

The function is activated at first valid tooth and is deactivated after a defined number of preinjections are performed.

After the preinjection is finished the injection is performed as described in the following chapters:

7.9.1.1 Mass fuel flow at start

Application condition:

Activation:	LV_ST_END = 0
Deactivation:	-
Initialization:	-
Recurrance:	10 ms

MFF_SP_HOM[NC_CBK_EX_NR] =	(MFF_CST	*Start injection fuel mass*
	* FAC_ST_REST	*Re-start correction factor*
	* FAC_ST_AMP	*Altitude correction*
	* IP_FAC_FQ_CST)	*Correction factor from fuel quality adaptation*
	* FAC_MFF_ST_EXT	*external correction factor for start*

MFF_SP_HOM_MV = MFF_SP_HOM[1]

7.9.1.2 Mass fuel flow out of start

Application condition:

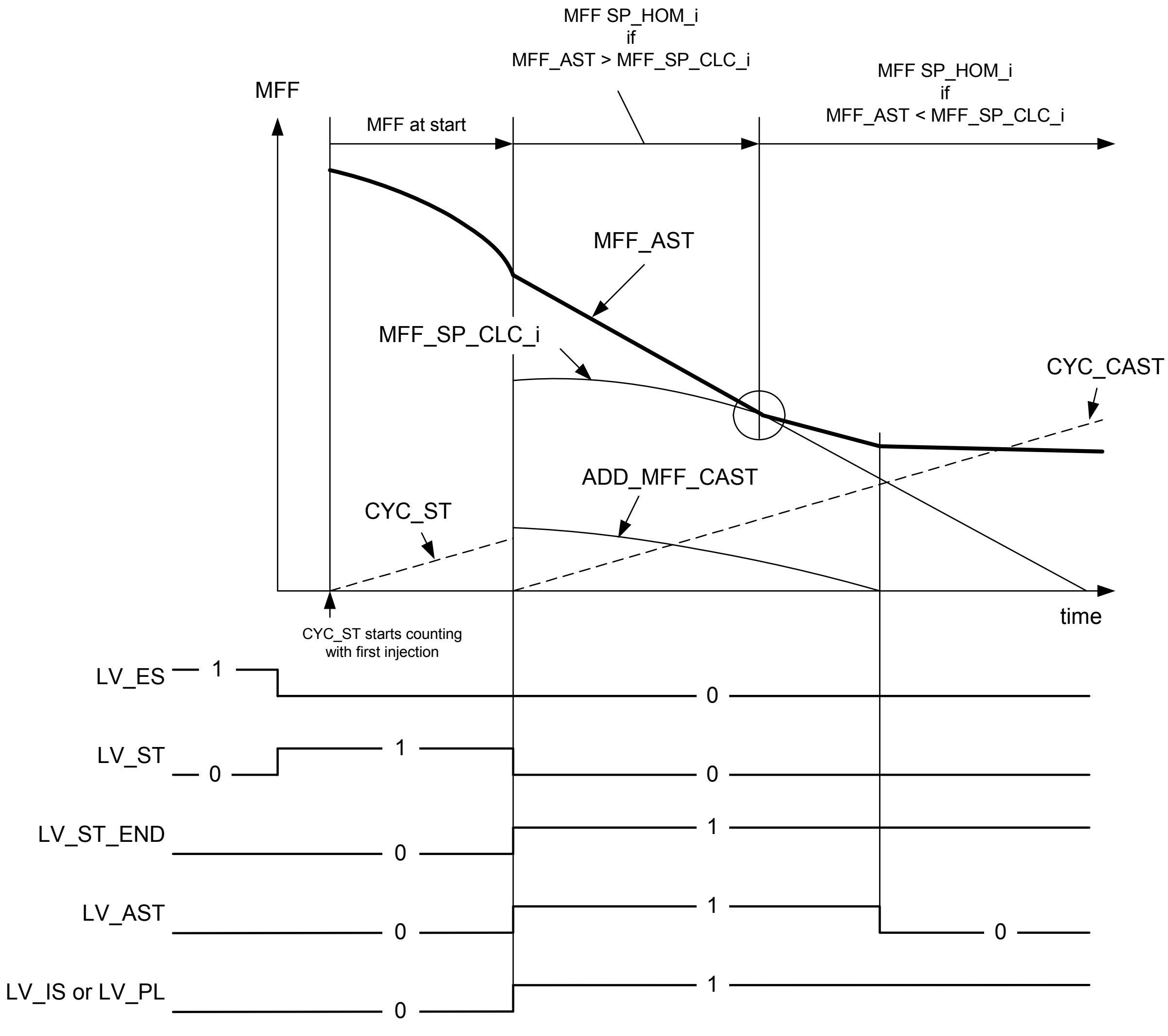
Activation:	LV_ST_END = 1
Deactivation:	-
Initialization:	at LV_ST_END = 0 ->1: MFF_LGRD_AST = IP_MFF_LGRD_AST MFF_AST = MFF_SP_HOM[1] (last value off MFF at start)
Recurrence:	segment synchronous

7.9.1.2.1 Mass fuel flow during after-start

If LV_ST_END is set to one (start is finished) the fuel mass flow for after-start MFF_AST is initialized with the last calculated MFF_SP_HOM[1] (calculated at start). MFF_AST is ramped down with the decrement MFF_LGRD_AST. The decrement is defined once at initialization from IP_ MFF_LGRD_AST.

In parallel MFF_SP_CLC[i] is calculated (see next chapter). Afterwards MFF_SP_CLC[i] and MFF_AST are compaired and the bigger value is selected for futher calculations.

MFF_AST(n) = MFF_AST(n-1) – MFF_LGRD_AST



File: FMSP_HOM.vsd

7.9.1.2.2 Calculation of MFF_SP_CLC[i]

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

MFF_SP_HOM_ENG[i] =	MFF_BAS	*basic fuel mass setpoint hom. mode*
	* (1+ FAC_LAM_COR[i])	*lambda control factor*
	* (FAC_MFF_WUP + ADD_MFF_CAST)	*temperature correction and cold after start corr.*
	* FAC_MFF_BAS_EXT	*external correction factor for basic setpoint*

 # IF
NC_WF_MFF_BAS = 1
 # THEN

MFF_SP_CLC[i]	= **{** MFF_SP_HOM_ENG[i]	
	- MAX(MFF_ADD_CP, MFF_ADD_DIAGCP) **}**	*fuel feed by canister purge*
	* (1+ FAC_LAM_AD_OUT[i])	*multiplicative lambda adaption*
	* (1+ FAC_CMBC_WF)	*combustion chamber wallfilm compensation*
	+ MFF_ADD_LAM_AD_OUT[i]	*additive lambda adaptation*
	+ MFF_ADD_LAM_AD_OUT_EXT	*external correction of the additive lambda adaptation (for vehicles without catalyst)*

 # ELSE

MFF_SP_CLC[i]	= **{** MFF_SP_HOM_ENG[i]	
	+ MFF_ADD_WF	*wall film correction*
	- MAX(MFF_ADD_CP, MFF_ADD_DIAGCP) **}**	*fuel feed by canister purge*
	* (1+ FAC_LAM_AD_OUT[i])	*multiplicative lambda adaption*
	* (1+ FAC_CMBC_WF)	*combustion chamber wallfilm compensation*
	+ MFF_ADD_LAM_AD_OUT[i]	*additive lambda adaptation*
	+ MFF_ADD_LAM_AD_OUT_EXT	*external correction of the additive lambda adaptation (for vehicles without catalyst)*

 # ENDIF

ENDFOR

7.9.1.2.3 Calculation of the output variable MFF_SP_HOM[i]

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

(1) IF

MFF_AST > MFF_SP_CLC[i]

(1) THEN

LV_MFF_AST_ACT[i] = 1

MFF_SP_HOM_CLC[i] = MFF_AST

(1) ELSE

LV_MFF_AST_ACT[i] = 0

MFF_SP_HOM_CLC[i] = MFF_SP_CLC[i]

(1) ENDIF

For calibration purposes the mass fuel setpoint MFF_SP_HOM[i] can be set by a manual value C_MFF_SP_HOM_MAN (LC_MFF_SP_HOM_MAN_ACT = 1):

(2) IF

LC_MFF_SP_HOM_MAN_ACT = 1

(2) THEN

MFF_SP_HOM[i] = C_MFF_SP_HOM_MAN + C_MFF_SP_HOM_OFS_AS

(2) ELSE

MFF_SP_HOM[i] = MFF_SP_HOM_CLC[i]

* FAC_MFF_EXT *.... factor for external correction*

+ MFF_ADD_EXT *.... additive external correction*

+ C_MFF_SP_HOM_OFS_AS *.... correction via application system*

(2) ENDIF

ENDFOR

For two cylinder banks (NC_CBK_EX_NR = 2) the mean value MFF_SP_HOM_MV is calculated:

MFF_SP_HOM_MV = (MFF_SP_HOM[1] + MFF_SP_HOM[2]) / 2

For one cylinder bank solutions (NC_CBK_EX_NR = 1):

MFF_SP_HOM_MV = MFF_SP_HOM[1]

Homogeneous MFF-setpoint for fuel pressure control:

(3) IF LV_TI_1_HOM_MIN = 0

(3) THEN MFF_SP_HOM_FUP_CTL = MFF_SP_HOM_MV

(3) ELSE MFF_SP_HOM_FUP_CTL is frozen

(3) ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MFF_SP_HOM_MAN	1	0 ... FFFFH	0...1389	0.02119	mg/stk
manual setpoint for fuel mass in homogeneous mode					
LC_MFF_SP_HOM_MAN_ACT	1	0...1H	0..1	1	-
switch for manual setpoint for fuel mass in homogeneous mode					
IP_MFF_LGRD_AST	8	0...FFFFH	0...1389	0.02119	mg/stk
LDP_TCO_IP_MFF_LGRD_AST	8	0...FEH	-48...142.5	0.75	°C
Injection time change limitation after Start					
C_MFF_SP_HOM_OFS_AS	1	0 ... FFFFH	0...1389	0.02119	mg/stk
additive mass fuel flow offset value for application					
IP_FAC_FQ_CST	6*6	0...FFH	0 ...1.992	0.0078	-
LDP_FAC_FQ_ST_AD_IP_FAC_FQ_CST	6	0...FFH	0 ...1.992	0.0078	-
LDP_TCO_IP_FAC_FQ_CST	6	0...FEH	-48 ...142.5	0.75	°C
Weighting of the fuel quality adaptation factor for cold start					

7.10 Calculation of MFF setpoint

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_SP[NC_CBK_EX_NR]	V/O	0 ... FFFFH	0 ... 1389	0.02119	mg/stk
Mass fuel flow setpoint after combustion selection, bank selective					
MFF_SP_MV	V/O	0 ... FFFFH	0 ... 1389	0.02119	mg/stk
Mass fuel flow setpoint after combustion selection					
MFF_SP_FUP_CTL	V/O	0 ... FFFFH	0 ... 1389	0.02119	mg/stk
Mass fuel flow setpoint for fuel pressure control					
EFF_INJ_AV	V	0 ... FFFFH	0 ... 1.9999	30.517E-6	-
Actual injection efficiency					

Input data:

LV_ST_END	MFF_SP_HOM_MV	MFF_SP_HOM[NC_CBK_EX_NR]	MFF_SP_HOM_FUP_CTL
MFF_POST_INJ	NC_CBK_EX_NR	NC_CYL_NR	SUM_INH_INJ

FUNCTION DESCRIPTION:

General information:

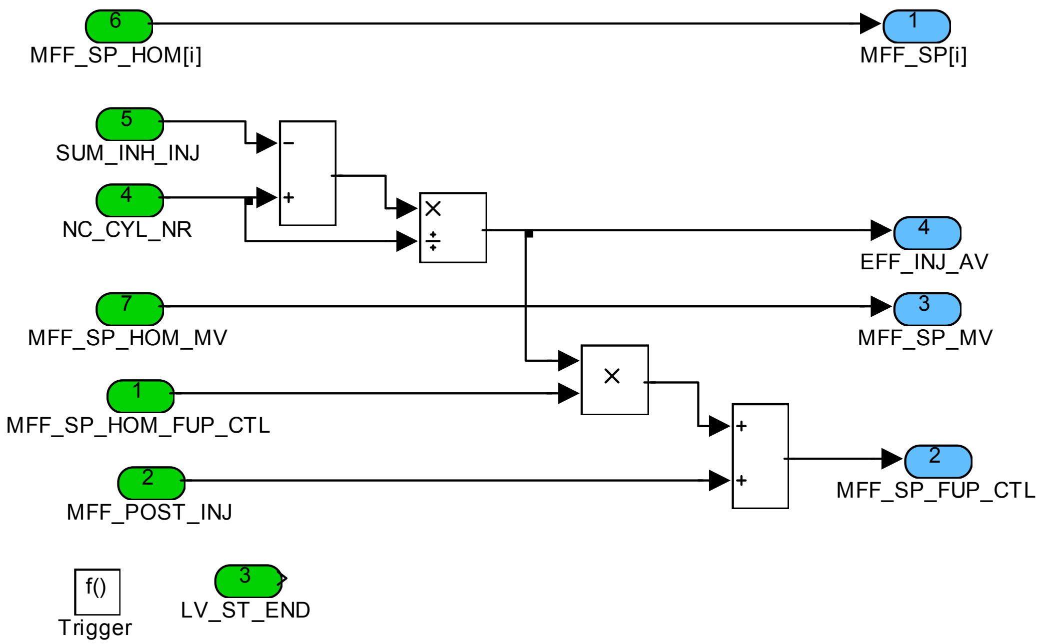
Module is used for MPI engines.

MFF_SP is set to MFF_SP_HOM_MV. MFF_SP_FUP_CTL is input for the fuel pressure (FUP) control functionality. Via the factor EFF_INJ_AV it is considered if a cylinder is shut off.

Application conditions:

Activation:	every engine state	
Deactivation:	-	
Initialization:	-	
Recurrance:	LV_ST_END = 0:	10 ms
	LV_ST_END = 1:	segment synchronous

Signal flow diagramm:



Formula section:

FOR [i] = 1 **TO** [NC_CBK_EX_NR] **DO**:

MFF_SP[i] = MFF_SP_HOM[i]

ENDFOR

EFF_INJ_AV = (NC_CYL_NR – SUM_INH_INJ) / NC_CYL_NR

MFF_SP_MV = MFF_SP_HOM_MV
MFF_SP_FUP_CTL = MFF_SP_HOM_FUP_CTL * EFF_INJ_AV + MFF_POST_INJ

7.11 Calculation of cylinder individual mass fuel flow setpoints

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_SP_1_HOM[NC_CYL_NR]	V/O	0 ... FFFFH	0 ... 1389	0.02119	mg/stk
Cylinder individual mass fuel flow setpoint for homogeneous mode, first pulse					
MFF_COR[NC_CBK_EX_NR]	V	0...FFFFH	0..1.999969	3.051E-5	-
Operating point correction for the injected fuel mass					

Input data:

MFF_SP_HOM[NC_CBK_EX_NR]	MFF_ADD_REAC_CYL[NC_CYL_NR]	FAC_TI_REAC[NC_CYL_NR]
MFF_ADD_WF_CYL[NC_CYL_NR]	MAF	N_32
LV_ST_END	NC_CYL_NR	NC_CBK_EX_NR
NC_WF_MFF_BAS		

FUNCTION DESCRIPTION:

General information:

The module calculates cylinder individual injection setpoints for homogeneous mode at single injection.

Application conditions:

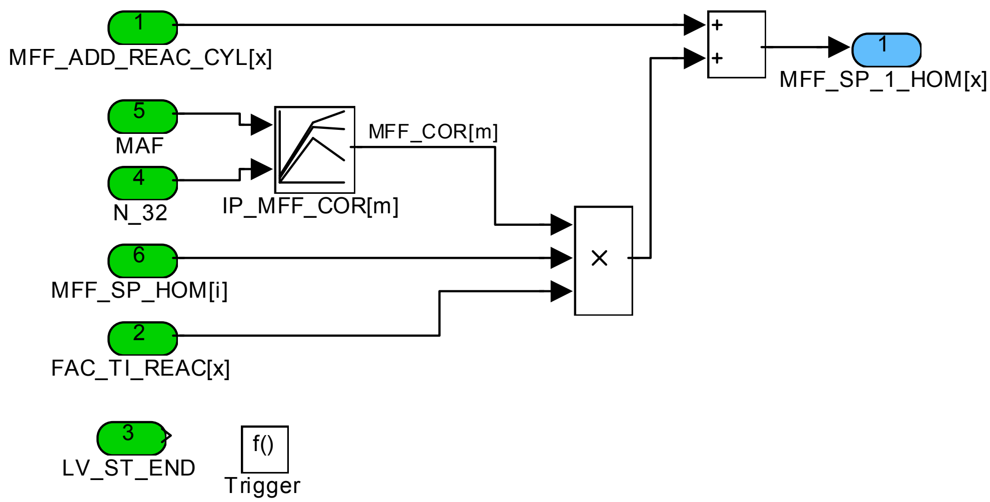
Activation: every engine state

Deactivation: -

Initialization: -

Recurrence: LV_ST_END = 0 : 10 ms
LV_ST_END = 1 : segment synchronous

Signal flow diagram:



Formula section:

FOR [m] = 1 **TO** NC_CBK_EX_NR **DO**:
MFF_COR[m] = IP_MFF_COR[m]
ENDFOR

#IF NC_WF_MFF_BAS = 1
#THEN

FOR [x] = 0 **TO** (NC_CYL_NR-1) **DO**:

MFF_SP_1_HOM[x] = MFF_SP_HOM[i] * FAC_TI_REAC[x] * MFF_COR[m] +
MFF_ADD_REAC_CYL[x] + MFF_ADD_WF_CYL[x]

ENDFOR

#ELSE

FOR [x] = 0 **TO** (NC_CYL_NR-1) **DO**:

MFF_SP_1_HOM[x] = MFF_SP_HOM[i] * FAC_TI_REAC[x] * MFF_COR[m] +
MFF_ADD_REAC_CYL[x]

ENDFOR

#ENDFOR

The index “i” indicates that the variable is allocated to the NC_LAMB_REF.

The index “m” indicates that the variable is allocated to the NC_IN_REF.

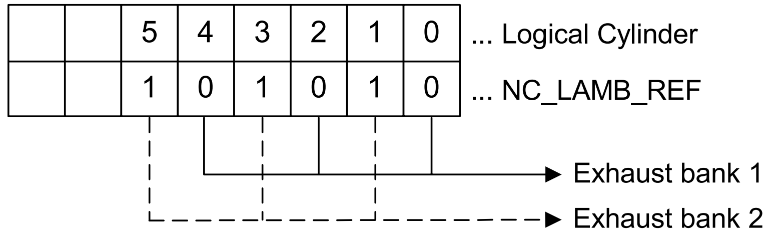
The index “x” indicates the logical cylinder.

The pattern NC_LAMB_REF indicates if a logical cylinder [x] is allocated to exhaust bank 1 (i = 1, bit [x] of NC_LAMB_REF = 0) or to exhaust bank 2 (i = 2, bit [x] of NC_LAMB_REF = 1).

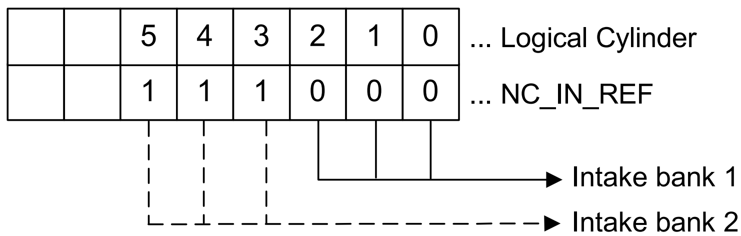
The pattern NC_IN_REF indicates if a logical cylinder [x] is allocated to intake bank 1 (m = 1, bit [x] of NC_IN_REF = 0) or to intake bank 2 (m = 2, bit [x] of NC_IN_REF = 1).

(The values of NC_LAMB_REF and NC_IN_REF are project specific.)

For example a 6 cylinder two exhaust bank engine:



For example a 6 cylinder two intake bank engine:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MFF_COR[NC_CBK_EX_NR]	12*16	0...FFFFH	0..1.999969	3.051E-5	-
LDPM_MAF_2_FMSP	12	0...FFFFH	0..1389	0.02119	mg/stk
LDPM_N_32_2_FMSP	16	0...FFH	0..8160	32	rpm
Operating point correction for the injected fuel mass					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_LAMB_REF	1	0 ... FFH	0 ... 255	1	-
Pattern for allocation of physical cylinders to exhaust bank					
NC_IN_REF	1	0 ... FFH	0 ... 255	1	-
Pattern for allocation of physical cylinders to intake bank					

7.12 Start pre injection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_PRE_INJ [NC_CYL_NR]	V/O	0 ... FFFFH	0...262.14	0.004	ms
Pre-injection time for number NC_CYL_NR of pre-injections					
MFF_PRE_INJ_BAS	V	0 ... FFFFH	0...5555	0.08476	mg/stk
Basic value of pre-injection at (LV_ES)					
MFF_PRE_INJ	V	0 ... FFFFH	0...5555	0.08476	mg/stk
Calculated mass fuel flow setpoint for pre-injection					
FAC_FQ_ST_AD_PRE_INJ	V	0 ... FFH	0...1.992	0.0078	-
Weighted fuel quality factor for pre injection					
N_PRE_INJ	V	0 ... 1FE0H	0...8160	1	Rpm
Engine cranking speed calculated on first valid crank tooth					

Input data:

FAC_ST_REST	TCO	NC_CYL_NR	
C_FAC_MFF_TI_STND_1	FAC_FQ_ST_AD	LV_ST_ES	NC_NR_TOOTH
AMP	TIA		

FUNCTION DESCRIPTION:

General information:

The cylinder individual pre-injection-time TI_PRE_INJ[CYC_PRE] is calculated only once after the first valid tooth of the crankshaft signal is detected.

The calculation of the pre-injection is fuel mass based but the output value TI_PRE_INJ[CYC_PRE] is time based, therefore at the end a conversion from MFF to TI is performed.

The basic pre-injection fuel mass is determined by the map IP_MFF_PRE_INJ_BAS. The behaviour of the intake manifold and the volumetric efficiency are taken into consideration and therefore the pre injection mass will be weighted by a factor from the map ID_FAC_MFF_PRE_DEAC. In case of a previous transition from “start” to “engine stop” the correction factor C_MFF_PRE_FAC_ST_ES for a new pre-injection will be applied. The counter CYC_PRE is only a local variable in the range of 0 to NC_CYL_NR-1.

Engine cranking speed can be considered to determine final pre-injection fuel mass. Cause engine cranking speed (N_PRE_INJ) is updated on the first valid crank tooth using last crank tooth period from BSW, N_PRE_INJ has big deviation from actual working point. So, Decision of ID_FAC_N_MFF_PRE_INJ should be carefully made.

Application conditions:

Activation: after first valid tooth of crankshaft signal is detected

Recurrence: only once at activation.

Formula section:

N_PRE_INJ = 60 / (NC_NR_TOOTH * Tooth Period From BSW)

FOR CYC_PRE = 0 **TO** NC_CYL_NR - 1

If LV_ST_ES = 0

Then

MFF_PRE_INJ_BAS = IP_MFF_PRE_INJ_BAS (TCO)
* ID_FAC_MFF_PRE_DEAC (CYC_PRE , TCO)

Else

// In case of a start break off and the engine operating state "Start" wasn't left,

// LV_ST_ES=1, the correction factor IP_FAC_MFF_PRE_ST_ES is applied:

MFF_PRE_INJ_BAS = IP_MFF_PRE_INJ_BAS (TCO)
* ID_FAC_MFF_PRE_DEAC (CYC_PRE , TCO)
* IP_FAC_MFF_PRE_ST_ES (TCO)

End If

FAC_FQ_ST_AD_PRE_INJ = IP_FAC_FQ_PRE_INJ (FAC_FQ_ST_AD)

MFF_PRE_INJ = MFF_PRE_INJ_BAS Pre-injection fuel mass

* FAC_ST_REST Re-start correction factor

* FAC_FQ_ST_AD_PRE_INJ Fuel quality adaptation

* ID_FAC_N_MFF_PRE_INJ (N_PRE_INJ)Cranking speed correction factor

* IP_FAC_MFF_PRE (TIA, AMP) TIA, altitude correction

End For

<u>Conversion pre-injection fuel mass ⇒ pre-injection time:</u>

The pre-injection time for standard conditions TI_PRE_INJ is calculated with the characteristic of the injection valves:

TI_PRE_INJ [CYC_PRE] = C_FAC_MFF_TI_STND_1 * MFF_PRE_INJ

END FOR

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_MFF_PRE_INJ_BAS	1x8	0 ... FFFFH	0 ... 5555	0.08476	mg/stk
LDP_TCO_IP_MFF_PRE_INJ_BAS	8	0 ... FEH	-48 ... 142.5	0.75	°C
Pre-injection basic value					
ID_FAC_MFF_PRE_DEAC	8x8	0 ... FFH	0 ... 1.9922	2 / 256	-
LDP_CYC_PRE_IP_FAC_MFF_PRE_DEAC	8	0 ... 7H	0 ... 7	1	-
LDPM_TCO_MFF_PRE_INJ_1	8	0…FEH	-48 … 142.5	0.75	°C
Pre injection time deactivation factor					
IP_FAC_MFF_PRE_ST_ES	1*8	0 ... FFH	0 ... 0.9960	0.0039	-
LDPM_TCO_MFF_PRE_INJ_1	8	0…FEH	-48 … 142.5	0.75	°C
Pre-injection basic value correction in case of previous transition ST to ES					
IP_FAC_FQ_PRE_INJ	1x8	0 ... FFH	0 ... 1.992	0.0078	-
LDP_FAC_FQ_ST_AD_IP_FAC_FQ_PRE	8	0 ... FFH	0 ... 1.992	0.0078	-
Weighted fuel quality factor for pre injection					
ID_FAC_N_MFF_PRE_INJ	1*8	0 ... FFH	0 ... 1.9922	2 / 256	-
LDP_N_PRE_INJ_MFF_PRE_INJ_1	8	0 … 1FE0H	0 … 8160	1	[rpm]
Pre-injection basic value correction considering engine cranking speed					
IP_FAC_MFF_PRE	8x6	0 ... FFH	0 ... 1.992	0.0078	-
LDP_TIA_IP_FAC_MFF_PRE	8	0 ... FEH	-48 ... 142.5	0.75	°C
LDP_AMP_IP_FAC_MFF_PRE	6	0 ... FFFFH	0 ... 5434	0.083	hPa
Pre-injection correction value					

7.12.1 Start injection fuel mass (MFF_CST)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_CST	O/V	0...FFFFH	0...1389	0.0211948	[mg/stk]
Basic injection at start (LV_ST)					

Input data:

TCO	N	CYC_ST	LV_ST_END
LV_AT	FAC_MFF_CST_EXT		

FUNCTION DESCRIPTION:

General information:

The basic mass fuel flow at start determined by a map dependent on engine speed N and coolant temperature TCO is multiplied by a deactivation factor dependent on CYC_ST in order to avoid engine flooding. For engines with automatic transmission a different calibration can be used by additional maps.

Application conditions:

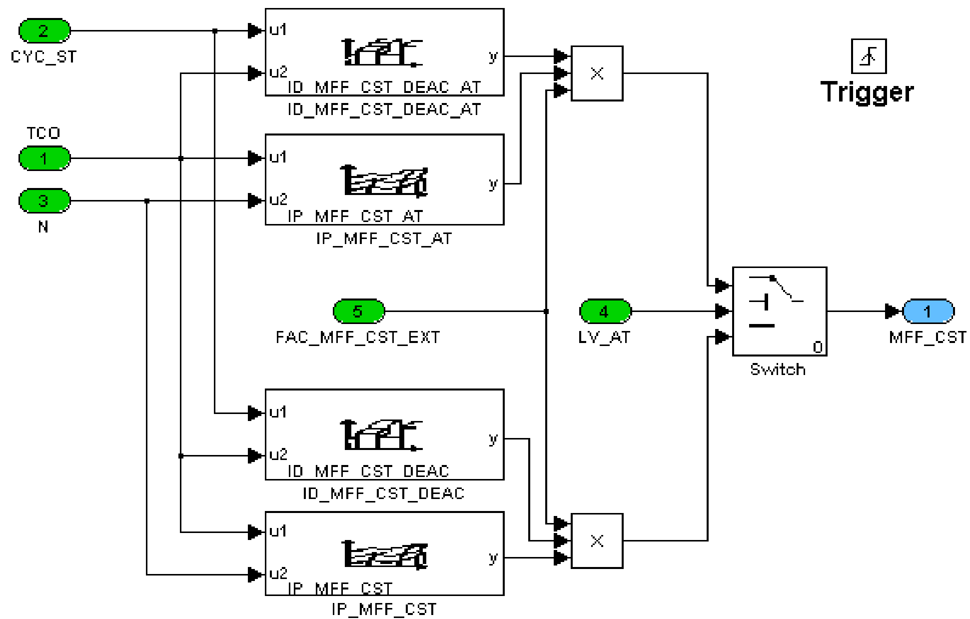
Activation: LV_ST_END = 0

Deactivation: -

Initialization: -

Recurrence: 10 ms

Signal flow diagram:



Formula section:

If LV_AT = 1

then

MFF_CST = IP_MFF_CST_AT * ID_MFF_CST_DEAC_AT
* FAC_MFF_CST_EXT

else

MFF_CST = IP_MFF_CST * ID_MFF_CST_DEAC
* FAC_MFF_CST_EXT

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MFF_CST	4*12	0...FFFFH	0...1389	0.021194	mg/stk
LDPM_N_1_FMSP	4	0...1FE0H	0...8160	1	rpm
LDPM_TCO_2_FMSP	12	0...FEH	-48...142.5	0.75	°C
Basic value of the mass fuel flow at start					
ID_ MFF_CST_DEAC	12*12	0...FFH	0...0.99609	0.0039063	-
LDPM_CYC_ST_1_FMSP	12	0...FFH	0...255	1	-
LDPM_TCO_2_FMSP	12	0...FEH	-48...142.5	0.75	°C
Cranking deactivation factor for the mass fuel flow at start					
IP_MFF_CST_AT	4*12	0...FFFFH	0...1389	0.021194	mg/stk
LDPM_N_1_FMSP	4	0...1FE0H	0...8160	1	rpm
LDPM_TCO_2_FMSP	12	0...FEH	-48...142.5	0.75	°C
Basic value of the mass fuel flow at start for automatic transmission					
ID_ MFF_CST_DEAC_AT	12*12	0...FFH	0...0.99609	0.0039063	-
LDPM_CYC_ST_1_FMSP	12	0...FFH	0...255	1	-
LDPM_TCO_2_FMSP	12	0...FEH	-48...142.5	0.75	°C
Cranking deactivation factor for the mass fuel flow at start for automatic transmission					

7.13 Fuel quality adaptation at engine start

7.13.1 Activation and deactivation of the fuel quality adaptation at engine start

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD	O/V	0...1H	0...1	1	-
Fuel quality adaptation active					
LV_FQ_ST_AD_INH	O/V	0...1H	0...1	1	-
Fuel quality adaptation inhibited					
LV_FQ_ST_AD_END_1	O/V	0...1H	0...1	1	-
End of fuel quality adaptation function					
LV_FQ_ST_AD_FIRST_VLD_TDC	V	0...1H	0...1	1	-
First detected valid top death centre					
LV_FQ_ST_AD_OK_1	V	0...1H	0...1	1	-
Fuel quality adaptation ended correctly, saving adaptation value allowed					
CTR_FQ_ST_AD_END	V	0...FFH	0...255	1	-
Counter to delay the complete end of fuel quality function					
FAC_FQ_ST_AD_SAVE_RNG_1	O/V/S	0...FFH	0 ...1.992	0.0078	-
Saved adaptation factor for the temperature range 1					
FAC_FQ_ST_AD_SAVE_RNG_2	O/V/S	0...FFH	0 ...1.992	0.0078	-
Saved adaptation factor for the temperature range 2					
FAC_FQ_ST_AD_SAVE_RNG_3	O/V/S	0...FFH	0 ...1.992	0.0078	-
Saved adaptation factor for the temperature range 3					

Input data:

CYC_ST_FQ_ST_AD	LV_FQ_ST_AD_END	LV_FQ_ST_AD_NOT_VLD	LV_FQ_ST_AD_UPD
LV_REST	FAC_FQ_ST_AD_RNG_1	FAC_FQ_ST_AD_RNG_2	FAC_FQ_ST_AD_RNG_3
FAC_FQ_ST_AD_FTL	N_32	TCO	

FUNCTION DESCRIPTION:

General information:

The purpose of fuel quality adaptation is to make sure that the engine runs without any dependency on fuel quality. This aim is realised by an increasing or a decreasing of the injection time by a factor. For that the fuel quality is recognised by the engine speed gradient at start.

The bit LV_FQ_ST_AD activates fuel quality adaptation. Some parts of the fuel quality adaptation; the temperature range selection and the adaptation factor output; must be activated earlier because they are necessary for the calculation of the first start injection. The counter CTR_FQ_ST_AD_END retards the end of the fuel quality adaptation function in order to wait for the plausibility of the FTL CAN signal. In the case, that the fuel tank level changed between two engine stops, the saved adaptation factors will be weighted by FAC_FQ_ST_AD_FTL.

The adaptation values are stored if the whole function ended without any error. The stored adaptation factors are initialised to 1 for after the ECU is brand new.

Application conditions:

Activation: LC_FQ_ST_AD_INH_MAN = 0

Deactivation: LC_FQ_ST_AD_INH_MAN = 1 **OR** after transition LV_FQ_ST_AD = 1→0

Initialisation: at *reset* and at *transition from engine run to engine stop*

LV_FQ_ST_AD = 0; LV_FQ_ST_AD_FIRST_VLD_TDC = 0;
CTR_FQ_ST_AD_END = 0

LV_FQ_ST_AD_OK_1 = 0; LV_FQ_ST_AD_END_1 = 0

at checksum error of the NVMY:

FAC_FQ_ST_AD_SAVE_RNG_1 = C_FAC_FQ_ST_AD_SAVE_RNG_1

FAC_FQ_ST_AD_SAVE_RNG_2 = C_FAC_FQ_ST_AD_SAVE_RNG_2

FAC_FQ_ST_AD_SAVE_RNG_3 = C_FAC_FQ_ST_AD_SAVE_RNG_3

Recurrence: every TDC

Formula section:

{ Inhibit function manual}

IF LC_FQ_ST_AD_INH_MAN = 0
THEN LV_FQ_ST_AD_INH = 0
ELSE LV_FQ_ST_AD_INH = 1

{ Because the update of FTL CAN values takes quite long}
{ wait a number of cycles until disable the whole function}

IF LV_FQ_ST_AD_END = 1
THEN inc (CTR_FQ_ST_AD_END)

{ End function completely}

IF LV_FQ_ST_AD_END_1 = 1 **OR**
LV_FQ_ST_AD_NOT_VLD=1 **OR** CTR_FQ_ST_AD_END >= C_DLY_FQ_ST_AD_END
THEN LV_FQ_ST_AD_END_1 = 1
ELSE LV_FQ_ST_AD_END_1 = 0

{adaptation cycle was carried out at least 1 times and engine really starts}

IF LV_FQ_ST_AD_OK_1 = 1
OR (LV_FQ_ST_AD_NOT_VLD = 0 *{no error}*
AND N_32 > C_N_32_MIN_FQ_ST_AD) *{engine starts really}*
THEN LV_FQ_ST_AD_OK_1 = 1
ELSE LV_FQ_ST_AD_OK_1 = 0

{ The analysis of the engine speed increase cycle was carried out at least 1 times}
{ successfully, independent if there was an increment or decrement detected. }

{ The function is now ready to save the adaptation values in the NVMY }

IF LV_FQ_ST_AD_END_1 = 1 **AND** LC_CAL_ST_FQ_ST_AD = 0

AND (LV_FQ_ST_AD_OK_1 = 1 **OR** (FAC_FQ_ST_AD_FTL > C_FAC_FQ_ST_AD_FTL_THD)) **AND** **LV_REST = 0**

THEN

FAC_FQ_ST_AD_SAVE_RNG_1 = FAC_FQ_ST_AD_RNG_1

- (FAC_FQ_ST_AD_RNG_1 - 1) * FAC_FQ_ST_AD_FTL

FAC_FQ_ST_AD_SAVE_RNG_2 = FAC_FQ_ST_AD_RNG_2

- (FAC_FQ_ST_AD_RNG_2 - 1) * FAC_FQ_ST_AD_FTL

FAC_FQ_ST_AD_SAVE_RNG_3 = FAC_FQ_ST_AD_RNG_3

- (FAC_FQ_ST_AD_RNG_3 - 1) * FAC_FQ_ST_AD_FTL

Else If LV_FQ_ST_AD_END_1 = 1 **AND** LC_CAL_ST_FQ_ST_AD = 0

AND (LV_FQ_ST_AD_OK_1 = 1 **OR** (FAC_FQ_ST_AD_FTL > C_FAC_FQ_ST_AD_FTL_THD)) **AND** **LV_REST = 1**

THEN

FAC_FQ_ST_AD_SAVE_RNG_1_{N} = FAC_FQ_ST_AD_SAVE_RNG_1_{N-1} - (FAC_FQ_ST_AD_SAVE_RNG_1_{N-1}- 1) * FAC_FQ_ST_AD_FTL

FAC_FQ_ST_AD_SAVE_RNG_2_{N} = FAC_FQ_ST_AD_SAVE_RNG_2_{N-1}

- (FAC_FQ_ST_AD_SAVE_RNG_2_{N-1}- 1) * FAC_FQ_ST_AD_FTL

FAC_FQ_ST_AD_SAVE_RNG_3_{N} = FAC_FQ_ST_AD_SAVE_RNG_3_{N-1}

- (FAC_FQ_ST_AD_SAVE_RNG_3_{N-1}- 1) * FAC_FQ_ST_AD_FTL

END

IF CYC_ST_FQ_ST_AD >= ID_TDC_DLY_FQ_ST_AD **OR**

LV_FQ_ST_AD_FIRST_VLD_TDC = 1

THEN LV_FQ_ST_AD_FIRST_VLD_TDC = 1

ELSE LV_FQ_ST_AD_FIRST_VLD_TDC = 0

{Enable or disable fuel quality subfunctions}

IF LV_FQ_ST_AD_END_1 = 0 **AND**

LV_FQ_ST_AD_FIRST_VLD_TDC = 1

THEN LV_FQ_ST_AD = 1

ELSE LV_FQ_ST_AD = 0

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
LC_CAL_ST_FQ_ST_AD	1	0...1H	0...1	1	-
Start calibration; no saving of the adaptation value					
ID_TDC_DLY_FQ_ST_AD	1*6	0...FFH	0...255	1	-
LDP_TCO_ID_TDC_DLY_FQ_ST_AD	6	0 ... FEH	-48 ... 142.5	0.75	°C
Number of segments to skip after first injection to start the adaptation					
C_DLY_FQ_ST_AD_END	1	0...FFH	0...255	1	-
Number of segments to skip after the adaptation ended to end the function					
LC_FQ_ST_AD_INH_MAN	1	0...1H	0...1	1	-
Fuel quality adaptation inhibited manual					
C_N_32_MIN_FQ_ST_AD	1	0 ... FFH	0 ... 8160	32	rpm
Engine speed threshold in order to detect a running engine after FQ adaptation has finished					
C_FAC_FQ_ST_AD_SAVE_RNG_1	1	0 ... FFH	0 ... 1.992	0.0078	-
Initialisation value for the NVMY at checksum error or new ECU for temperature range 1					
C_FAC_FQ_ST_AD_SAVE_RNG_2	1	0 ... FFH	0 ... 1.992	0.0078	-
Initialisation value for the NVMY at checksum error or new ECU for temperature range 2					
C_FAC_FQ_ST_AD_SAVE_RNG_3	1	0 ... FFH	0 ... 1.992	0.0078	-
Initialisation value for the NVMY at checksum error or new ECU for temperature range 3					
C_FAC_FQ_ST_AD_FTL_THD	1	0 ... FFH	0 ... 0.996	0.0039	-
Fuel tank level change threshold to ignore condition for new adaptation (default value : FEh)					

7.13.1.1 End of adaptation

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD_END	V	0...1H	0...1	1	-
End of adaptation (but not end the fuel quality function)					
LV_FQ_ST_AD_END_N	V	0...1H	0...1	1	-
End of adaptation (but not end the fuel quality function) due to engine speed overshoot					

Input data:

CYC_FQ_ST_AD	LV_FQ_ST_AD	N_32	N_SP_IS

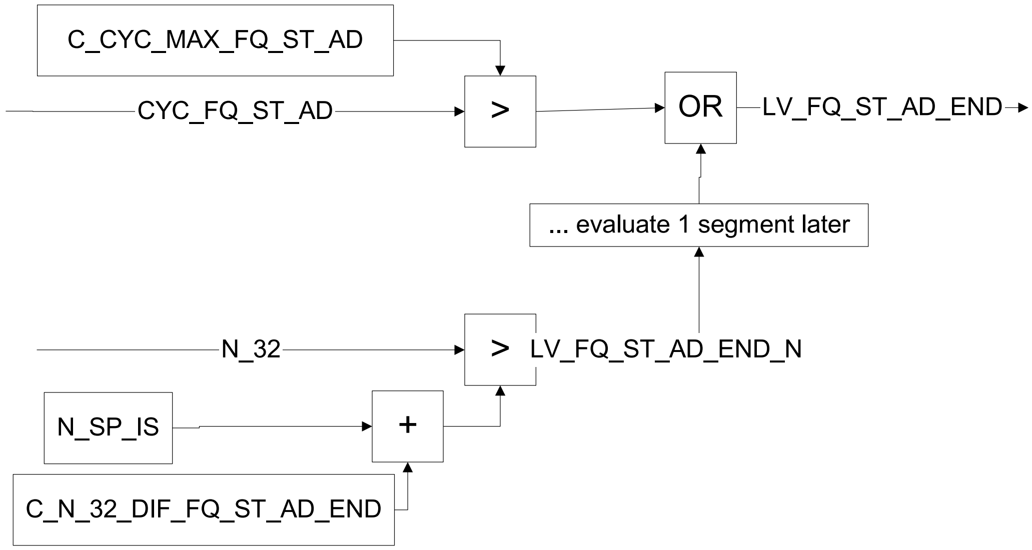
FUNCTION DESCRIPTION:

General information:

The adaptation is activated for a certain number of combustion cycles. If the cycle counter is bigger than the cycle counter threshold for adaptation end, the adaptation is stopped. The cycle counter starts with the value 1. The adaptation is stopped too, in the case when the engine speed overshoot a certain threshold beyond the idle speed setpoint N_SP_IS, due to a very good fuel.

The fuel quality adaptation function ends later (see. LV_FQ_ST_AD_END_1) because the adaptation values still have to be saved in the NVMY.

Signal flow diagram:



Application conditions:

Activation:	LV_FQ_ST_AD = 1
Deactivation:	after transition LV_FQ_ST_AD = 1 → 0
Initialisation:	LV_FQ_ST_AD_END = 0
	LV_FQ_ST_AD_END_N = 0
	at *reset* and at *transition from engine run to engine stop*
Recurrence:	every TDC

Formula section:

IF (CYC_FQ_ST_AD > C_CYC_MAX_FQ_ST_AD) **OR**
(LV_FQ_ST_AD_END_N = 1)
THEN LV_FQ_ST_AD_END = 1
ELSE LV_FQ_ST_AD_END = 0

{The detection of the engine speed overshoot is placed after the evaluation of CYC_FQ_ST_AD, because the FQA will end one segment later and the FQA can still evaluate the last combustion.}
IF (N_32 > N_SP_IS + C_N_32_DIF_FQ_ST_AD_END)
THEN LV_FQ_ST_AD_END_N = 1

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_CYC_MAX_FQ_ST_AD	1	0...FFH	0...255	1	-
Cycle counter threshold for adaptation end					
C_N_32_DIF_FQ_ST_AD_END	1	80...7FH	-4096...4064	32	rpm
Engine speed difference to end the adaptation					

7.13.1.2 Fuel quality adaptation not valid

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD_NOT_VLD	V	0...1H	0...1	1	-
Fuel quality adaptation not valid					

Input data:

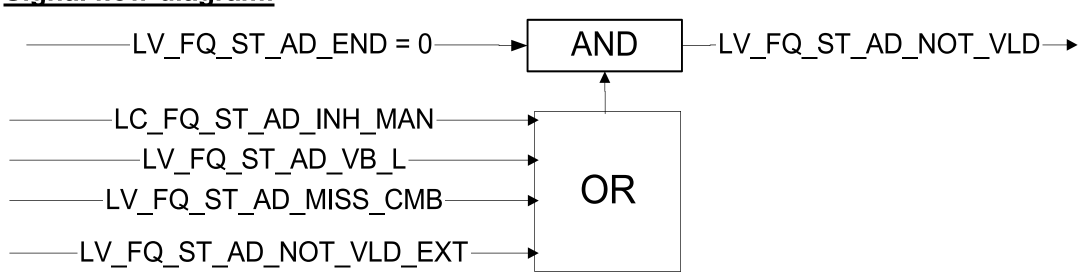
LV_FQ_ST_AD_VB_L	LV_FQ_ST_AD_MISS_CMB	LV_FQ_ST_AD_INH	LV_FQ_ST_AD
LV_FQ_ST_AD_NOT_VLD_EXT	LV_FQ_ST_AD_END		

FUNCTION DESCRIPTION:

General information:

The execution of fuel quality adaptation is prohibited depending on one of the following state bits. The logical constant LC_FQ_ST_AD_INH_MAN is already defined in chapter 1.1. The examination is only carried out as long as the FQA is not finished, LV_FQ_ST_AD_END=0.

Signal flow diagram:



Application conditions:

Activation: LV_FQ_ST_AD_INH = 0

Deactivation: LV_FQ_ST_AD_INH = 1 **OR** after transition LV_FQ_ST_AD = 1→0

Initialisation: LV_FQ_ST_AD_NOT_VLD = 0

at *reset* and at *transition from engine run to engine stop*

Recurrence: every TDC

Formula section:

IF LV_FQ_ST_AD_END = 0 *{FQA is active}*

THEN IF LC_FQ_ST_AD_INH_MAN = 1 **OR** LV_FQ_ST_AD_VB_L =1 **OR**
LV_FQ_ST_AD_MISS_CMB = 1 **OR** LV_FQ_ST_AD_NOT_VLD_EXT = 1
THEN LV_FQ_ST_AD_NOT_VLD = 1

7.13.1.2.1 Adaptation not valid because of low battery voltage

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD_VB_L	V	0...1H	0...1	1	-
Fuel quality adaptation not valid because of low battery voltage					
CTR_VB_MIN_FQ_ST_AD	V	0...FFH	0...255	1	-
Counter for detection of low battery					

Input data:

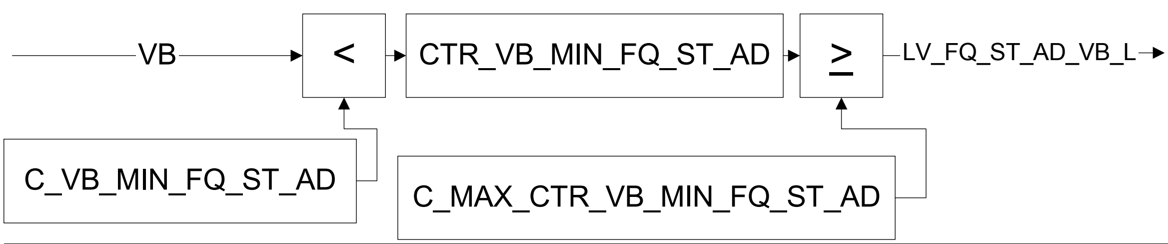
VB	LV_FQ_ST_AD_INH	LV_FQ_ST_AD	LV_ES

FUNCTION DESCRIPTION:

General information:

If battery voltage several times is lower than a threshold the bit LV_FQ_ST_AD_VB_L is set to 1.

Signal flow diagram:



Application conditions:

Activation: LV_FQ_ST_AD_INH = 0 **AND** LV_ES = 0

Deactivation: LV_FQ_ST_AD_INH = 1 **OR** after transition LV_FQ_ST_AD = 1→0 **OR** LV_ES = 1

Initialisation: CTR_VB_MIN_FQ_ST_AD = 0; LV_FQ_ST_AD_VB_L = 0 at *reset* and at *transition from engine run to engine stop*

Recurrence: 20 ms

Formula section:

IF VB < C_VB_MIN_FQ_ST_AD

THEN CTR_VB_MIN_FQ_ST_AD = CTR_VB_MIN_FQ_ST_AD + 1

ELSE CTR_VB_MIN_FQ_ST_AD = CTR_VB_MIN_FQ_ST_AD

IF CTR_VB_MIN_FQ_ST_AD >= C_MAX_CTR_VB_MIN_FQ_ST_AD

THEN LV_FQ_ST_AD_VB_L = 1

ELSE LV_FQ_ST_AD_VB_L = 0

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_VB_MIN_FQ_ST_AD	1	0...FFH	0...26	0.102	V
Minimum battery voltage for fuel quality adaptation					
C_MAX_CTR_VB_MIN_FQ_ST_AD	1	0...FFH	0...255	1	-
Maximum occurence of low battery voltage					

7.13.1.2.2 Adaptation not valid because of less detected combustions

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD_MISS_CMB	V	0...1H	0...1	1	-
Not enough combustions detected					
CTR_CMB_FQ_ST_AD	V	0...FFH	0...255	1	-
Counter for less detected combustions					

Input data:

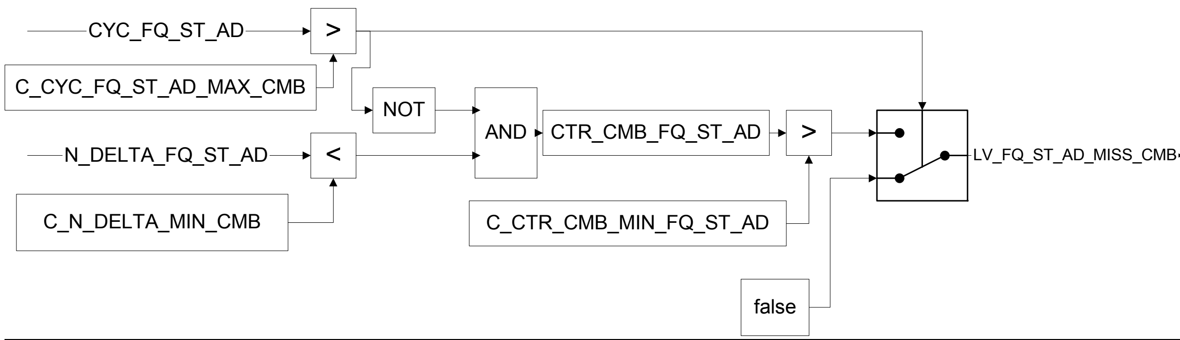
CYC_FQ_ST_AD	N_DELTA_FQ_ST_AD	LV_FQ_ST_AD_INH	LV_FQ_ST_AD
LV_FQ_ST_AD_END			

FUNCTION DESCRIPTION:

General information:

If the cycle counter CYC_FQ_ST_AD is greater than a threshold and the engine speed difference also exceeds its threshold, the value of valid combustions must be greater than a specified threshold. Otherwise the bit LV_FQ_ST_AD_MISS_CMB is set to 1.The examination is deactivated in the case if LV_FQ_ST_AD_END=1.

Signal flow diagram:



Description:

C_CYC_FQ_ST_AD_MAX_CMB must be greater than C_CYC_MAX_FQ_ST_AD to make sure that the inquiry of successful combustions has been finished before the fuel quality adaptation is started.

Application conditions:

Activation: LV_FQ_ST_AD = 1

Deactivation: LV_FQ_ST_AD_INH = 1 **OR** LV_FQ_ST_AD_END = 1 **OR**
after transition LV_FQ_ST_AD = 1→0

Initialisation: CTR_CMB_FQ_ST_AD = 0; LV_FQ_ST_AD_MISS_CMB = 0
at *reset* and at *transition from engine run to engine stop*

Recurrence: every TDC

Formula section:

IF N_DELTA_ FQ_ST_AD < C_N_DELTA_MIN_CMB **AND**
CYC_FQ_ST_AD ≤ C_CYC_FQ_ST_AD_MAX_CMB
THEN CTR_CMB_FQ_ST_AD = CTR_CMB_FQ_ST_AD + 1
ELSE CTR_CMB_FQ_ST_AD = CTR_CMB_FQ_ST_AD

IF CTR_CMB_FQ_ST_AD > C_CTR_CMB_MIN_FQ_ST_AD **AND**
CYC_FQ_ST_AD > C_CYC_FQ_ST_AD_MAX_CMB
THEN LV_FQ_ST_AD_MISS_CMB = 1
ELSE LV_FQ_ST_AD_MISS_CMB = 0

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_CYC_FQ_ST_AD_MAX_CMB	1	0...FFH	0...255	1	-
Threshold for max. value of cycle counter concerning to the validation of combustions					
C_N_DELTA_MIN_CMB	1	E020...1FE0H	-8160...8160	1	rpm
Minimum engine speed difference for validation of combustions					
C_CTR_CMB_MIN_FQ_ST_AD	1	0...FFH	0...255	1	-
Minimum of valid combustions					

7.13.2 1.2. Cycle counter and engine speed difference

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CYC_FQ_ST_AD	V	0...FFH	0...255	1	-
Cycle counter for the fuel quality adaptation					
N_DELTA_FQ_ST_AD	V	E020...1FE0H	-8160...8160	1	rpm
Engine speed difference					
N_ST	V	0...1FE0H	0...8160	1	rpm
Engine speed for fuel quality adaptation					
N_ST_PREV	V	0...1FE0H	0...8160	1	rpm
Engine speed for fuel quality adaptation calculated from previous segment					

Input data:

T_SEG_FQ_ST_AD	LV_FQ_ST_AD	N	NC_CYL_NR

FUNCTION DESCRIPTION:

General information:

For the fuel quality adaptation the engine speed is calculated different from the standard engine speed calculation (different segment). Besides it's necessary to store the engine speed from previous calculation step, because it is used as a map input.

Application conditions:

Activation: LV_FQ_ST_AD = 1

Deactivation: after transition LV_FQ_ST_AD = 1 → 0

Initialisation: CYC_FQ_ST_AD = 0; N_DELTA_FQ_ST_AD = 0; N_ST = 0
at *reset* and at *transition from engine run to engine stop*

Recurrence: every TDC

Formula section:

CYC_FQ_ST_AD = CYC_FQ_ST_AD + 1

N_ST_PREV = N_ST

IF LC_N_STND_FQ_ST_AD = 1

THEN N_ST = N

ELSE

$$N\_ST = \frac{2}{NC\_CYL\_NR \cdot T\_SEG\_FQ\_ST\_AD}$$

END

IF CYC_FQ_ST_AD > 1

THEN N_DELTA_FQ_ST_AD = N_ST – N_ST_PREV

ELSE N_DELTA_FQ_ST_AD = 0

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
LC_N_STND_FQ_ST_AD	1	0...1H	0...1	1	-
switch to the standard engine speed					

7.13.3 Analysis of the engine speed increase

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FQ_ST_AD_INC_FAC	V	0...1H	0...1	1	-
Incrementation of the adaptation factor necessary					
LV_FQ_ST_AD_DEC_FAC	V	0...1H	0...1	1	-
Decrementation of the adaptation factor necessary					
LV_FQ_ST_AD_UPD	V	0...1H	0...1	1	-
Cycle of the analysis ready, update the adaptation factor					
CTR_N_DELTA_MIN_FQ_ST_AD	V	0...FFH	0...255	1	-
Lower limit counter for engine speed difference					
CYC_UPD_FQ_ST_AD	V	0...FFH	0...255	1	-
After this cycle counter expired the decision of increasing or decreasing is made					
CTR_N_DELTA_MAX_FQ_ST_AD	V	0...FFH	0...255	1	-
Upper limit counter for engine speed difference					
N_DELTA_MIN_FQ_ST_AD	V	E020...1FE0H	-8160...8160	1	rpm
Lower limit for engine speed difference					
N_DELTA_MAX_FQ_ST_AD	V	E020...1FE0H	-8160...8160	1	rpm
Upper limit for engine speed difference					
CTR_MAX_N_DELTA_MIN_FQ_ST_AD	V	0...FFH	0...255	1	-
Maximum threshold for counter of minimum engine speed difference					

Input data:

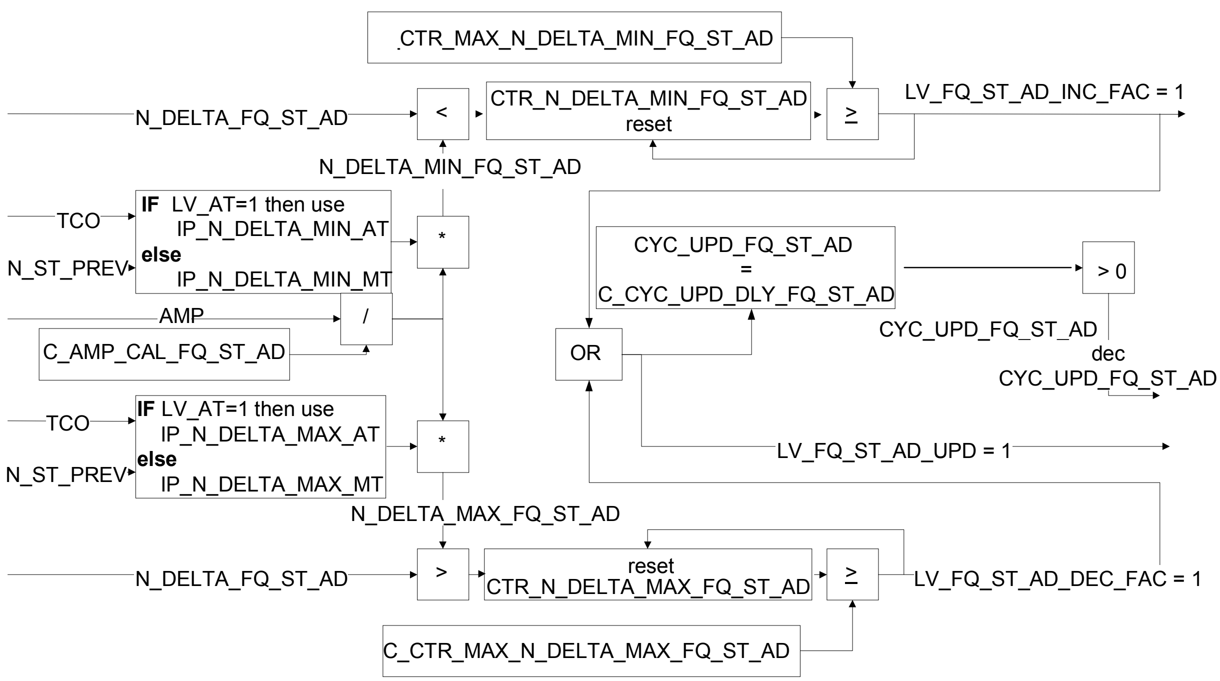
N_DELTA_FQ_ST_AD	TCO	N_ST_PREV	AMP
CYC_FQ_ST_AD	LV_FQ_ST_AD	LV_AT	DIST

FUNCTION DESCRIPTION:

General information:

The whole function is based on the analysis of the increase of the engine speed during the start. If a too low engine speed gradient at start is detected the adaptation factor for the start fuel injection time will be increased. Engine speed gradients being too low are leading to a decrease of the adaptation factor.

Signal flow diagram:



Description:

An upper and a lower limit give the intended engine speed increase. A map corrected by the ambient air pressure determines each limit. These maps are dependent on the previous engine speed and the coolant temperature. Each time the real engine speed increase is outside these limits this event is counted either by the upper limit counter CTR_N_DELTA_MAX_FQ_ST_AD or by the lower limit counter CTR_N_DELTA_MIN_FQ_ST_AD. If an adaptation took place, the function wait a certain number of TDC´s, determined by C_CYC_UPD_DLY_FQ_ST_AD, until a new evaluation can take place. This is due to the effect, that a new adapted fuel amount has to be aspirated, compressed and burned before it can be evaluated. The decision to in- or decrease the adaptation factor is made dependent on the counter maxima. After that the counter is reset.

Application conditions:

Activation: LV_FQ_ST_AD = 1

Deactivation: after transition LV_FQ_ST_AD = 1 → 0

Initialisation: all counters = 0; LV _FQ_ST_AD_INC_FAC = 0;

LV _FQ_ST_AD_DEC_FAC = 0; LV_FQ_ST_AD_UPD = 0

CTR_MAX_N_DELTA_MIN_FQ_ST_AD = ID_CTR_MAX_N_DELTA_MIN_FQ_ST_AD

at *reset* and at *transition from engine run to engine stop*

Recurrence: every TDC

Formula section:

If(1) $ID\_CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N}$ <> $ID\_CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{(N-1)}$

Then(1)

If(2) CTR_N_DELTA_MIN_FQ_ST_AD >= ID_CTR_MAX_N_DELTA_MIN_FQ_ST_AD

Then(2) $CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N}$ =CTR_N_DELTA_MIN_FQ_ST_AD + C_CTR_OFS_N_DELTA_MIN_FQ_ST_AD

Else(2) $CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N}$ = $ID\_CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N}$

Endif(2)

Else(1) $CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N}$ = $CTR\_MAX\_N\_DELTA\_MIN\_FQ\_ST\_AD_{N-1}$

Endif(1)

```
IF(1)CYC_FQ_ST_AD > 1    THEN

    IF(2) LV_AT = 1
      Then
         N_DELTA_MIN_FQ_ST_AD  = IP_N_DELTA_MIN_AT  *   AMP  / C_AMP_CAL_FQ_ST_AD
                                               + IP_N_DELTA_ADD__DIST__N_ST_PREV
         N_DELTA_MAX_FQ_ST_AD = IP_N_DELTA_MAX_AT  *  AMP  / C_AMP_CAL_FQ_ST_AD
      Else
         N_DELTA_MIN_FQ_ST_AD  =  IP_N_DELTA_MIN_MT  *   AMP / C_AMP_CAL_FQ_ST_AD
                                               + IP_N_DELTA_ADD__DIST__N_ST_PREV
         N_DELTA_MAX_FQ_ST_AD = IP_N_DELTA_MAX_MT   * AMP / C_AMP_CAL_FQ_ST_AD
    End(2)

    IF(3)  CYC_UPD_FQ_ST_AD > 0
       THEN dec (CYC_UPD_FQ_ST_AD)
       ELSE
          IF(4)   N_DELTA_FQ_ST_AD < N_DELTA_MIN_FQ_ST_AD
                 THEN   inc (CTR_N_DELTA_MIN_FQ_ST_AD)
                 ELSE  IF N_DELTA_FQ_ST_AD > N_DELTA_MAX_FQ_ST_AD
                        THEN  inc (CTR_N_DELTA_MAX_FQ_ST_AD)
          END(4)
    END(3)
    ELSE(1)   all counters = 0
END(1)

LV_FQ_ST_AD_INC_FAC = 0
LV_FQ_ST_AD_DEC_FAC = 0

IF(5)CTR_N_DELTA_MIN_FQ_ST_AD >= CTR_MAX_N_DELTA_MIN_FQ_ST_AD
    THEN   LV_FQ_ST_AD_INC_FAC = 1
           CTR_N_DELTA_MIN_FQ_ST_AD = 0
           CYC_UPD_FQ_ST_AD = C_CYC_UPD_DLY_FQ_ST_AD
           LV_FQ_ST_AD_UPD = 1
    END
    ELSE
           IF CTR_ N_DELTA_ MAX_FQ_ST_AD >= C_CTR_MAX_N_DELTA_MAX_FQ_ST_AD
              THEN   LV_FQ_ST_AD_DEC_FAC = 1
                     CTR_N_DELTA_MAX_FQ_ST_AD = 0
                     CYC_UPD_FQ_ST_AD = C_CYC_UPD_DLY_FQ_ST_AD
                     LV_FQ_ST_AD_UPD = 1
               END
END(5)
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_N_DELTA_MAX_MT	8*8	0...3FC0H	-8160...8160	1	rpm
LDPM_N_ST_PREV_1_FMSP	8	0...1FE0H	0...8160	1	rpm
LDPM_TCO_5_FMSP	8	0...FEH	-48...142.5	0.75	°C
Interpolated table of maximum engine speed difference for fuel quality adaptation and manual transmission					
IP_N_DELTA_MIN_MT	8*8	0...3FC0H	-8160...8160	1	rpm
LDPM_N_ST_PREV_1_FMSP	8	0...1FE0H	0...8160	1	rpm
LDPM_TCO_5_FMSP	8	0...FEH	-48...142.5	0.75	°C
Interpolated table of minimum engine speed difference for fuel quality adaptation and manual transmission					
IP_N_DELTA_MAX_AT	8*8	0...3FC0H	-8160...8160	1	rpm
LDPM_N_ST_PREV_1_FMSP	8	0...1FE0H	0...8160	1	rpm
LDPM_TCO_5_FMSP	8	0...FEH	-48...142.5	0.75	°C
Interpolated table of maximum engine speed difference for fuel quality adaptation and automatic transmission					
IP_N_DELTA_MIN_AT	8*8	0...3FC0H	-8160...8160	1	rpm
LDPM_N_ST_PREV_1_FMSP	8	0...1FE0H	0...8160	1	rpm
LDPM_TCO_5_FMSP	8	0...FEH	-48...142.5	0.75	°C
Interpolated table of minimum engine speed difference for fuel quality adaptation and automatic transmission					
C_AMP_CAL_FQ_ST_AD	1	0...FFFFH	0...5434	0.083	hPa
Ambient pressure for calibration					
C_CYC_UPD_DLY_FQ_ST_AD	1	0...FFH	0...255	1	-
Delay a new update for star injection time adaptation					
C_CTR_MAX_N_DELTA_MAX_FQ_ST_AD	1	0...FFH	0...255	1	-
Maximum threshold for counter of maximum engine speed difference					
ID_CTR_MAX_N_DELTA_MIN_FQ_ST_AD	3	0...FFH	0...255	1	-
LDP_N_32_ID_CTR_MAX_N_DELTA_MIN	3	0…FFH	0…8160	32	rpm
Maximum threshold for counter of minimum engine speed difference					
IP_N_DELTA_ADD__DIST__N_ST_PREV	8	0...3FC0H	-8160...8160	1	rpm
LDP_DIST_N_DELTA_ADD	8	0x0 .. 0xFFFF	0 .. 655350	10	km
LDPM_N_ST_PREV_1_FMSP	8	0...1FE0H	0...8160	1	[rpm]
additive correction on minimum engine speed threshold for mileage accumulation					
C_CTR_OFS_N_DELTA_MIN_FQ_ST_AD	1	0...7FH	0...127	1	-
Offset on maximum threshold for counter of minimum engine speed difference in case of saturation					

7.13.4 Temperature range selection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FQ_ST_AD_TCO_ST_RNG_0	V	0...1H	0...1	1	-
Coolant temperature at start is higher than temperature range 1					
LV_FQ_ST_AD_TCO_ST_RNG_1	V	0...1H	0...1	1	-
Coolant temperature at start in temperature range 1					
LV_FQ_ST_AD_TCO_ST_RNG_2	V	0...1H	0...1	1	-
Coolant temperature at start in temperature range 2					
LV_FQ_ST_AD_TCO_ST_RNG_3	V	0...1H	0...1	1	-
Coolant temperature at start in temperature range 3					
LV_FQ_ST_AD_TCO_ST_RNG_4	V	0...1H	0...1	1	-
Coolant temperature at start is lower than temperature range 3					
LV_FQ_ST_AD_WF	O/V	0...1H	0...1	1	-
Wall film correction by the fuel quality adaptation allowed					
LV_FQ_ST_AD_AST	O/V	0...1H	0...1	1	-
After start injection correction by the fuel quality adaptation allowed					
FAC_FQ_ST_AD_DEC_TCO	O/V	0...FFH	0...0.996	0.0039	-
Factor to decrement FAC_FQ_ST_AD linear for temperatures above range 1					

Input data:

TCO			

FUNCTION DESCRIPTION:

General information:

The coolant temperature at start is classified into range 0 to 4 but a fuel adaptation will only carried out in the ranges 1 to 3. Temperature range 0 represent the higher temperature area and range 4 is assigned for the lower temperature area. In the range 4 only the adaptation value FAC_FQ_ST_AD from the range 1 and 3 will be applied. In the range 0 the value FAC_FQ_ST_AD will linear decremented with the factor FAC_FQ_ST_AD_DEC_TCO.

Depending on the temperature range the related bit LV_FQ_ST_AD_TCO_ST_RNG_x is set to 1. If one of the 3 bits within temperature range 1 to 3 is set to 1 and the coolant temperature at start is smaller than the max. coolant temperature for wall film correction, the bit LV_FQ_ST_AD_WF is set to 1. If the coolant temperature at start is classified into one of the three temperature ranges and if it is smaller than the max. coolant temperature for after start injection correction, the bit LV_FQ_ST_AD_AST is set to 1.

Signal flow diagram:

Application conditions:

Initialisation: all outputs = 0

at *reset* and at *transition from engine run to engine stop*

Recurrence: first valid tooth before preinjection calculation

Formula section:

```
IF  TCO<C_MAX_TCO_ST_RNG_0  AND  TCO > C_MAX_TCO_ST_RNG_1
    THEN
        LV_FQ_ST_AD_TCO_ST_RNG_0 = 1
        FAC_FQ_ST_AD_DEC_TCO = (TCO  -  C_MAX_TCO_ST_RNG_1) /
                               (C_MAX_TCO_ST_RNG_0 - C_MAX_TCO_ST_RNG_1)
    ELSE
        FAC_FQ_ST_AD_DEC_TCO = 0
        IF  TCO ≤  C_MAX_TCO_ST_RNG_1 AND TCO >  C_MIN_TCO_ST_RNG_1
        THEN
            LV_FQ_ST_AD_TCO_ST_RNG_1 = 1
        ELSE
            IF      TCO ≤   C_MIN _TCO_ST _RNG_1   AND   TCO >   C _MIN_TCO_ST_ RNG_2
            THEN
                    LV_FQ_ST_AD_TCO_ST_RNG_2 = 1
            ELSE
                    IF
                    TCO  ≤    C_MIN_TCO_ST_RNG_2    AND  TCO  >    C_MIN_TCO_ST_RNG_3
                    THEN
                        LV_FQ_ST_AD_TCO_ST_RNG_3 = 1
                    ELSE
                        LV_FQ_ST_AD_TCO_ST_RNG_1 = 0
                        LV_FQ_ST_AD_TCO_ST_RNG_2 = 0
                        LV_FQ_ST_AD_TCO_ST_RNG_3 = 0
                        LV_FQ_ST_AD_TCO_ST_RNG_4 = 1
                        LV_FQ_ST_AD_TCO_ST_RNG_0 = 0
END


IF  (LV_FQ_ST_AD_TCO_ST_RNG_1 = 1)
    OR      (LV_FQ_ST_AD_TCO_ST_RNG_2 = 1)
    OR      (LV_FQ_ST_AD_TCO_ST_RNG_3 = 1)
    THEN
            IF      TCO < C_MAX_TCO_ST_WF_FQ_ST_AD
                    THEN   LV_FQ_ST_AD_WF = 1
                    ELSE   LV_FQ_ST_AD_WF = 0
            END
            IF      TCO < C_MAX_TCO_ST_AST_FQ_ST_AD
                    THEN   LV_FQ_ST_AD_AST = 1
                    ELSE   LV_FQ_ST_AD_AST = 0
            END
    ELSE
            LV_FQ_ST_AD_WF = 0
            LV_FQ_ST_AD_AST = 0
END
```

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_MAX_TCO_ST_RNG_0	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature maximum for temperature range 0					
C_MAX_TCO_ST_RNG_1	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature maximum for temperature range 1					
C_MIN_TCO_ST_RNG_1	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature minimum for temperature range 1					
C_MIN_TCO_ST_RNG_2	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature minimum for temperature range 2					
C_MIN_TCO_ST_RNG_3	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature minimum for temperature range 3					
C_MAX_TCO_ST_WF_FQ_ST_AD	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature maximum for wall film correction					
C_MAX_TCO_ST_AST_FQ_ST_AD	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature maximum for after start injection correction					

7.13.5 In-/decrementation step

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
DELTA_FAC_FQ_ST_AD	V	80...7FH	-1...0.992	0.0078	-
In-/decrementation step for the fuel quality adaptation factor					

Input data:

LV_FQ_ST_AD_TCO_ST_RNG_1	LV_FQ_ST_AD_TCO_ST_RNG_2	LV_FQ_ST_AD_TCO_ST_RNG_3
LV_FQ_ST_AD_INC_FAC	LV_FQ_ST_AD_DEC_FAC	LV_FQ_ST_AD

FUNCTION DESCRIPTION:

General information:

At the beginning of the function either the decrease or increase bit is active. If the incrementation bit is active, one of the incrementation constants, depending on TCO_ST, are taken over as the value of the Inc-/decrementation factor DELTA_FAC_TI ST_AD. If these conditions are fulfilled, the constant related to start temperature is taken over as the value of the Inc-/decrementation factor DELTA_FAC_FQ_ST_AD.

Signal flow diagram:

Application conditions:

Activation: LV_FQ_ST_AD = 1

Deactivation: after transition LV_FQ_ST_AD = 1 → 0

Initialisation: DELTA_FAC_FQ_ST_AD = 0;

at *reset* and at *transition from engine run to engine stop*

Recurrence: every TDC

Formula section:

```
IF(1) LV_FQ_ST_AD_INC_FAC = 1 THEN

    IF      LV_FQ_ST_AD_TCO_ST_RNG_1 = 1 THEN
        DELTA_FAC_FQ_ST_AD = C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_1

    ELSE
        IF      LV_FQ_ST_AD_TCO_ST_RNG_2 = 1 THEN
            DELTA_FAC_FQ_ST_AD = C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_2

        ELSE

            IF      LV_FQ_ST_AD_TCO_ST_RNG_3 = 1 THEN
                    DELTA_FAC_FQ_ST_AD = C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_3
            ELSE
                    DELTA_FAC_FQ_ST_AD = 0


ELSE (1)


   IF(2) LV_FQ_ST_AD_DEC_FAC = 1 THEN

        IF      LV_FQ_ST_AD_TCO_ST_RNG_1 = 1 THEN

                DELTA_FAC_FQ_ST_AD = C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_1

        ELSE

            IF      LV_FQ_ST_AD_TCO_ST_RNG_2 = 1 THEN

                     DELTA_FAC_FQ_ST_AD=C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_2

            ELSE

              IF LV_FQ_ST_AD_TCO_ST_RNG_3 = 1 THEN

                    DELTA_FAC_FQ_ST_AD=C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_3

              ELSE

                    DELTA_FAC_FQ_ST_AD = 0

        END


     ELSE (2)

        DELTA_FAC_FQ_ST_AD = 0

     END(2)

END(1)
```

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_1	1	80...7FH	-1 ... 0.992	0.0078	-
Incrementation step for the temperature range 1					
C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_2	1	80...7FH	-1 ... 0.992	0.0078	-
Incrementation step for the temperature range 2					
C_INC_FAC_FQ_ST_AD_TCO_ST_RNG_3	1	80...7FH	-1 ... 0.992	0.0078	-
Incrementation step for the temperature range 3					
C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_1	1	80...7FH	-1 ... 0.992	0.0078	-
Decrementation step for the temperature range 1					
C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_2	1	80...7FH	-1 ... 0.992	0.0078	-
Decrementation step for the temperature range 2					
C_DEC_FAC_FQ_ST_AD_TCO_ST_RNG_3	1	80...7FH	-1 ... 0.992	0.0078	-
Decrementation step for the temperature range 3					

Remark: The C_INC_... constants have to be set to positive values and
the C_DEC_... constants have to be set to negative values!

7.13.6 Correction factor for changing fuel tank level

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_FQ_ST_AD_FTL	O/V	0...FFH	0 ... 0.996	0.0039	-
Correction for FTL influence of the fuel quality adaptation					
FAC_FTL_CHG	V	0...FFH	0 ... 0.996	0.0039	-
Factor of changed fuel tank level					

Input data:

FTL_AV_FQ_ST_AD	FTL_SAVE_FQ_ST_AD	LV_FQ_ST_AD	

FUNCTION DESCRIPTION:

General information:

In the case that the fuel tank level has changed after engine stop and start again, it can be supposed that the fuel tank has been filled up. For that reason, the fuel quality could have changed and the adaptive factor will be reduced.

Application conditions:

Initialisation: at reset and at transition from engine run to engine stop

FAC_FQ_ST_AD_FTL = 0

Recurrence: every TDC

Activation: LV_FQ_ST_AD = 1

Deactivation: after transition LV_FQ_ST_AD = 1 → 0

FUNCTION DESCRIPTION:

IF FTL_AV_FQ_ST_AD < (FTL_SAVE_FQ_ST_AD – C_FAC_FQ_ST_AD_FTL_HYS)
THEN *{Fuel tank was drained}*
FAC_FQ_ST_AD_FTL = 0.996
ELSE
IF FTL_AV_FQ_ST_AD > FTL_SAVE_FQ_ST_AD
THEN
FAC_ FTL_CHG = (FTL_AV_FQ_ST_AD – FTL_SAVE_FQ_ST_AD) / FTL_AV_FQ_ST_AD
FAC_FQ_ST_AD_FTL = IP_FQ_FTL_CHG
ELSE
FAC_FQ_ST_AD_FTL = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_FQ_ST_AD_FTL_HYS	1	0 ... FFH	0 ... 0.996	0.0039	-
Hysteresis for changing fuel tank level signal					
IP_FQ_FTL_CHG	1*8	0 ... FFH	0 ... 0.996	0.0039	-
LDP_FAC_FTL_CHG_IP_FQ_FTL_CHG	8	0 ... FFH	0 ... 0.996	0.0039	-
Interpolated table to correct the influence of a changed fuel tank level					

7.13.7 Correction factors for not active temperature range and adaptation factors

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
FAC_FQ_ST_AD	O/V	0...FFH	0 ...1.992	0.0078	-
Adaptation factor for fuel quality adaptation					
FAC_FQ_ST_AD_DEC_COR	V	0...FFH	0 ...1.992	0.0078	-
Correction factor for decrementation case					
FAC_FQ_ST_AD_RNG_1	V	0...FFH	0 ...1.992	0.0078	-
Adaptation factor for the temperature range 3					
FAC_FQ_ST_AD_RNG_2	V	0...FFH	0 ...1.992	0.0078	-
Adaptation factor for the temperature range 3					
FAC_FQ_ST_AD_RNG_3	V	0...FFH	0 ...1.992	0.0078	-
Adaptation factor for the temperature range 3					

Input data:

DELTA_FAC_FQ_ST_AD	LV_FQ_ST_AD_TCO_ST_RNG_1	LV_FQ_ST_AD_TCO_ST_RNG_2
LV_FQ_ST_AD_TCO_ST_RNG_3	LV_FQ_ST_AD_END	LV_FQ_ST_AD
LV_FQ_ST_AD_INI_FAC	LV_FQ_ST_AD_TCO_ST_RNG_0	LV_FQ_ST_AD_TCO_ST_RNG_4
FAC_FQ_ST_AD_SAVE_RNG_1	FAC_FQ_ST_AD_SAVE_RNG_2	FAC_FQ_ST_AD_SAVE_RNG_3
LV_FQ_ST_AD_DEC_FAC	FAC_FQ_ST_AD_DEC_TCO	LV_REST

FUNCTION DESCRIPTION:

General information:

If the function has once detected that an in- or decrease of the adaptation value for a specific temperature range is necessary then an in- or decrease of the adaptation values for the other temperature ranges will also be necessary but with a different in-/decrementation step. To handle this the actual in-/decrementation step is added to the adaptation value with different weighting factors.

Description:

As long as LV_FQ_ST_AD = 0 the adaptation factors for the different temperature ranges FAC_FQ_ST_AD_TCO_ST_RNG_x are equal to the saved adaptation factors FAC_FQ_ST_AD_SAVE_TCO_ST_RNG_x. If the adaptation is active (LV_FQ_ST_AD = 1) then the weighted in-/decrementation steps DELTA_FAC_FQ_ST_AD_RNG_x are added to the adaptation factors. These factors are limited by the parameters C_FAC_FQ_ST_AD_MIN_TCO_ST_RNG_x and C_FAC_FQ_ST_AD_MAX_TCO_ST_RNG_x. The variables DELTA_FAC_FQ_ST_AD_RNG_x are only local variables. An initialisation of the adaptation factors can be forced by setting the bit LC_FQ_ST_AD_INI_FAC to 1. In case a decrementation was detected, the factors of the other temperature ranges will be weighted again by the factor C_FAC_FQ_ST_AD_DEC_COR.

Signal flow diagram:

Application conditions:

Activation: first valid tooth before preinjection calculation

Deactivation: LV_FQ_ST_AD = 1 → 0

Initialisation: **IF(1)** LC_FQ_ST_AD_INI_FAC_MAN = 1 or LV_FQ_ST_AD_INI_FAC = 1 **THEN**

FAC_FQ_ST_AD = C_FAC_FQ_ST_AD_INI
FAC_FQ_ST_AD_RNG_1 = C_FAC_FQ_ST_AD_INI
FAC_FQ_ST_AD_RNG_2 = C_FAC_FQ_ST_AD_INI
FAC_FQ_ST_AD_RNG_3 = C_FAC_FQ_ST_AD_INI
ELSE(1)
IF(2) LV_REST = 0 THEN
FAC_FQ_ST_AD = 1
FAC_FQ_ST_AD_RNG_1 = FAC_FQ_ST_AD_SAVE_RNG_1
FAC_FQ_ST_AD_RNG_2 = FAC_FQ_ST_AD_SAVE_RNG_2
FAC_FQ_ST_AD_RNG_3 = FAC_FQ_ST_AD_SAVE_RNG_3
ELSE(2)
FAC_FQ_ST_AD = 1
FAC_FQ_ST_AD_RNG_1 = 1
FAC_FQ_ST_AD_RNG_2 = 1
FAC_FQ_ST_AD_RNG_3 = 1
Endif(2)
ENDIF(1)

at *reset* and *at activation condition transition to true*

Note : Initialisation is done at Reset and at first valid tooth detection (just before formula section (first fac_fq_st_ad computation) and before preinjection calculation.

Recurrence: every TDC

Formula section:

```
IF(1)  LV_FQ_ST_AD = 1AND LV_FQ_ST_AD_END=0 THEN

    IF  LV_FQ_ST_AD_DEC_FAC  = 1
       THEN  FAC_FQ_ST_AD_DEC_COR = C_FAC_FQ_ST_AD_DEC_COR
        ELSE FAC_FQ_ST_AD_DEC_COR = 1
    END

    IF(2)   LV_FQ_ST_AD_TCO_ST_RNG_1 = 1        THEN
            DELTA_FAC_FQ_ST_AD_RNG_1 = DELTA_FAC_FQ_ST_AD
            DELTA_FAC_FQ_ST_AD_RNG_2 = DELTA_FAC_FQ_ST_AD *
                                       C_FAC_COR_1_TCO_ST_RNG_2 *
                                       FAC_FQ_ST_AD_DEC_COR
            DELTA_FAC_FQ_ST_AD_RNG_3 = DELTA_FAC_FQ_ST_AD *
                                      C_FAC_COR_1_TCO_ST_RNG_3 *
                                       FAC_FQ_ST_AD_DEC_COR

    ELSE(2)
            IF(3)   LV_FQ_ST_AD_TCO_ST_RNG_2 = 1        THEN
                    DELTA_FAC_FQ_ST_AD_RNG_2 = DELTA_FAC_FQ_ST_AD
                    DELTA_FAC_FQ_ST_AD_RNG_1 = DELTA_FAC_FQ_ST_AD *
                                               C_FAC_COR_2_TCO_ST_RNG_1 *
                                               FAC_FQ_ST_AD_DEC_COR

                    DELTA_FAC_FQ_ST_AD_RNG_3 = DELTA_FAC_FQ_ST_AD *
                                               C_FAC_COR_2_TCO_ST_RNG_3 *
                                               FAC_FQ_ST_AD_DEC_COR

            ELSE(3)
                    IF(4)   LV_FQ_ST_AD_TCO_ST_RNG_3 = 1        THEN
                            DELTA_FAC_FQ_ST_AD_RNG_3 = DELTA_FAC_FQ_ST_AD
                            DELTA_FAC_FQ_ST_AD_RNG_1 = DELTA_FAC_FQ_ST_AD *
                                                       C_FAC_COR_3_TCO_ST_RNG_1 *
                                                       FAC_FQ_ST_AD_DEC_COR
                            DELTA_FAC_FQ_ST_AD_RNG_2 = DELTA_FAC_FQ_ST_AD *
                                                      C_FAC_COR_3_TCO_ST_RNG_2 *
                                                      FAC_FQ_ST_AD_DEC_COR

                    ELSE(4)         DELTA_FAC_FQ_ST_AD_RNG_1 = 0
                                    DELTA_FAC_FQ_ST_AD_RNG_2 = 0
                                    DELTA_FAC_FQ_ST_AD_RNG_3 = 0
                    END(4)
            END(3)
    END(2)

    FAC_FQ_ST_AD_RNG_1=FAC_FQ_ST_AD_RNG_1 + DELTA_FAC_FQ_ST_AD_RNG_1
    FAC_FQ_ST_AD_RNG_2=FAC_FQ_ST_AD_RNG_2 + DELTA_FAC_FQ_ST_AD_RNG_2
    FAC_FQ_ST_AD_RNG_3=FAC_FQ_ST_AD_RNG_3 + DELTA_FAC_FQ_ST_AD_RNG_3
```

```
    IF(5)   FAC_FQ_ST_AD_RNG_1 > C_FAC_FQ_ST_AD_MAX_RNG_1             THEN
            FAC_FQ_ST_AD_RNG_1 = C_FAC_FQ_ST_AD_MAX_RNG_1
    ELSE
            IF(6)   FAC_FQ_ST_AD_RNG_1 < C_FAC_FQ_ST_AD_MIN_RNG_1     THEN
                    FAC_FQ_ST_AD_RNG_1 = C_FAC_FQ_ST_AD_MIN_RNG_1
            END(6)
    END(5)

    IF(7)   FAC_FQ_ST_AD_RNG_2 > C_FAC_FQ_ST_AD_MAX_RNG_2             THEN
                    FAC_FQ_ST_AD_RNG_2 = C_FAC_FQ_ST_AD_MAX_RNG_2
    ELSE
            IF(8)   FAC_FQ_ST_AD_RNG_2 < C_FAC_FQ_ST_AD_MIN_RNG_2     THEN
                    FAC_FQ_ST_AD_RNG_2 = C_FAC_FQ_ST_AD_MIN_RNG_2
            END(8)
    END(7)


    IF(9)   FAC_FQ_ST_AD_RNG_3 > C_FAC_FQ_ST_AD_MAX_RNG_3             THEN
                    FAC_FQ_ST_AD_RNG_3 = C_FAC_FQ_ST_AD_MAX_RNG_3
    ELSE
            IF(10)  FAC_FQ_ST_AD_RNG_3 < C_FAC_FQ_ST_AD_MIN_RNG_3     THEN
                    FAC_FQ_ST_AD_RNG_3 = C_FAC_FQ_ST_AD_MIN_RNG_3
            END(10)
    END(9)
END(1)


IF(11)  LV_FQ_ST_AD_TCO_ST_RNG_0 = 1    THEN
    FAC_FQ_ST_AD = FAC_FQ_ST_AD_RNG_1  –
                    (FAC_FQ_ST_AD_RNG_1  –  1) * FAC_FQ_ST_AD_DEC_TCO
ELSE
    IF(12)  LV_FQ_ST_AD_TCO_ST_RNG_1 = 1        THEN
            FAC_FQ_ST_AD = FAC_FQ_ST_AD_RNG_1
    ELSE
       IF(13)       LV_FQ_ST_AD_TCO_ST_RNG_2 = 1        THEN
            FAC_FQ_ST_AD = FAC_FQ_ST_AD_RNG_2
       ELSE
            IF(14)  LV_FQ_ST_AD_TCO_ST_RNG_3 = 1 OR
                     LV_FQ_ST_AD_TCO_ST_RNG_4 = 1       THEN
                    FAC_FQ_ST_AD = FAC_FQ_ST_AD_RNG_3
            ELSE(14)
                    FAC_FQ_ST_AD = 1
            END(14)
       END(13)
     END(12)
END(11)
```

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_FAC_COR_1_TCO_ST_RNG_2	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 1 on adaptation value range 2					
C_FAC_COR_1_TCO_ST_RNG_3	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 1 on adaptation value range 3					
C_FAC_COR_2_TCO_ST_RNG_1	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 2 on adaptation value range 1					
C_FAC_COR_2_TCO_ST_RNG_3	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 2 on adaptation value range 3					
C_FAC_COR_3_TCO_ST_RNG_1	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 3 on adaptation value range 1					
C_FAC_COR_3_TCO_ST_RNG_2	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor in-/decrementation for coolant temperature range 3 on adaptation value range 2					
C_FAC_FQ_ST_AD_DEC_COR	1	0...FFH	0 ...1.992	0.0078	-
Weighting factor for decrementation case					
C_FAC_FQ_ST_AD_MAX_RNG_1	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor maximum in temperature range 1					
C_FAC_FQ_ST_AD_MIN_RNG_1	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor minimum in temperature range 1					
C_FAC_FQ_ST_AD_MAX_RNG_2	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor maximum in temperature range 2					
C_FAC_FQ_ST_AD_MIN_RNG_2	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor minimum in temperature range 2					
C_FAC_FQ_ST_AD_MAX_RNG_3	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor maximum in temperature range 3					
C_FAC_FQ_ST_AD_MIN_RNG_3	1	0...FFH	0 ...1.992	0.0078	-
adaptation factor minimum in temperature range 3					
C_FAC_FQ_ST_AD_INI	1	0...FFH	0 ...1.992	0.0078	-
Initialisation value for manual reset of adaptation values					
LC_FQ_ST_AD_INI_FAC_MAN	1	0...01H	0 ...1	1	-
Force initialisation of adaptation values					

7.14 Fuel quality adaptation at engine start (appl. inc)

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_FQ_ST_AD_NOT_VLD_EXT	V/O	0...1H	0...1	-	-
Fuel quality adaptation external error					
CYC_ST_FQ_ST_AD	V/O	0...FFFFH	0...65535	1	-
Cycle counter start after detecting 8 valid tooth					
LV_REST_FQ_ST_AD	V/O	0...1H	0...1	-	-
Restart function active (=1), not active (=0)					
LV_FQ_ST_AD_INI_FAC	V/O	0...1H	0...1	-	-
Force an initialisation of the stored adaptation factors					
T_SEG_FQ_ST_AD	V/O	0 ... FFFFH	0 ... 524.28	0.008	ms
Segment time for fuel quality adaptation					
FTL_AV_FQ_ST_AD	V/O	0 ... FFH	0 ... 0.996	0.0039	-
Fuel tank level actual value for fuel quality adaptation					
FTL_SAVE_FQ_ST_AD	S/V/O	0 ... FFH	0 ... 0.996	0.0039	-
Fuel tank level for fuel quality adaptation stored in the non volatile memory					

Input data:

T_SEG_AV	LV_REST	LV_ERR_CAM	LV_ERR_CRK
LV_ERR_MAF	LV_ERR_MAP	LV_ERR_TCO	FTL_MMV
LV_ERR_VB	LV_ERR_IGC	LV_ERR_IV[NC_CYL_NR]	CYC_ST
PV_AV	LV_FQ_ST_AD_INH		

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_FTL_SAVE_FQ_ST_AD	1	0 ... FFH	0 ... 0.996	0.0039	-
Default value fuel tank level for fuel quality adaptation stored in the non volatile memory					

FUNCTION DESCRIPTION:

General information:

The purpose of the fuel quality adaptation function (FQA) is to make sure that the engine runs without any dependency on fuel quality. This aim is realised by an increasing or a decreasing of a fuel quality factor. For that the fuel quality is recognised by the engine speed gradient at start.

Application conditions:

Initialisation: at reset

LV_FQ_ST_AD_NOT_VLD_EXT = 0
CYC_ST_FQ_ST_AD = 0
LV_REST_FQ_ST_AD = 0
LV_FQ_ST_AD_INI_FAC = 0
T_SEG_FQ_ST_AD = 0
FTL_AV_FQ_ST_AD = 0
/ read* FTL_SAVE_FQ_ST_AD *from non volatile memory */*
***If** non stored NVMY values* **or** *NVMY has a checksum error*

Then FTL_SAVE_FQ_ST_AD = C_ FTL_SAVE_FQ_ST_AD

at transition from engine run to engine stop
FTL_AV_FQ_ST_AD = FTL_MMV / 100 *(actual value)*
FTL_SAVE_FQ_ST_AD = FTL_AV_FQ_ST_AD

Recurrence: every TDC

Activation: LV_FQ_ST_AD_INH = 0

Deactivation: LV_FQ_ST_AD_INH = 1

Formula section:

The execution of fuel quality adaptation function is prohibited depending on one of the following error and state bits or the variable PV_AV is greater than 0. In the case of a port injection engine, the logical variable LV_ST_H_PRS_ACT for a HPDI engine has to be defined and set to 0 in the chapter general.

```
IF  LV_ERR_CAM = 1                      OR
    LV_ERR_CRK = 1                      OR
    LV_ERR_MAF = 1                      OR
    LV_ERR_MAP = 1                      OR
    LV_ERR_TCO = 1                      OR
    LV_ERR_VB = 1                       OR
    LV_ERR_IGC = 1                      OR
    LV_ERR_IV = 1                       OR
    PV_AV > 0                           OR

    THEN
            LV_FQ_ST_AD_NOT_VLD_EXT = 1
    ELSE
            LV_FQ_ST_AD_NOT_VLD_EXT = 0
ENDIF
```

The variable CYC_ST_FQ_ST_AD describes the number of segments from the moment when 8 valid teeth were detected.

CYC_ST_FQ_ST_AD = CYC_ST

If a restart was detected and carried out then this event has to be signalised to the fuel quality adaptation in order to disable the module.

```
        IF          (LV_REST = 1        OR      LV_REST_FQ_ST_AD = 1)
        THEN        LV_REST_FQ_ST_AD = 1
        ELSE        LV_REST_FQ_ST_AD = 0
ENDIF
```

The range of the ECU frequency dependent segment time T_SEG_AV is converted and assigned to the fuel quality adaptation segment time T_SEG_FQ_ST_AD

T_SEG_FQ_ST_AD = T_SEG_AV

The change of the fuel tank level FTL is taken into consideration if a FTL signal is available.

IF LV_ERR_FTL = 0 /* Fuel tank level signal is available */
then FTL_AV_FQ_ST_AD = FTL_MMV / 100
else FTL_AV_FQ_ST_AD = 0.996
endif

Under specific circumstances an initialisation of the stored adaptation factors from the NVMY can be forced.

LV_FQ_ST_AD_INI_FAC = 0

7.15 Catalyst heating correction

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LAMB_AFL_CH	V	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
Lambda setpoint for catalyst heating with lean strategy					
LAMB_CH	O/V	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
Lambda catalyst heating					
LV_LAMB_CH	O/V	0... 1H	0... 1	1	[-]
indicates that lambda catalyst heating is active					

Input Data:

TQI_REQ_SLOW	TCO	N_32	LV_IS
T_AST	TCO_ST	LV_SAWUP	LAMB_SA_CH
STATE_CH	T_AST_COR_CH	FAC_LAMB_CH_EXT	PORT_AV

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_PORT_AV_THD_LAMB_CH	1	F000... 1000H	-100 ...100	0.0244141	[°PORT]
Actual port flap position threshold for switching LAMB_CH in after start.					
IP_LAMB_ADD_CH	8*8	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
LDPM_T_AST_1_EXTC	8	0... FFFFH	0... 6553.5	0.1	[s]
LDPM_TCO_1_EXTC	8	0... FEH	-48... 142.5	0.75	[°C]
Additive correction of lambda setpoint for catalyst heating in homogeneous					
IP_LAMB_CH_IS	6*6	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
LDPM_N_32_5_EXTC	6	0... FFH	0... 8160	32	[rpm]
LDPM_TQI_REQ_SLOW_2_EXTC	6	0... FFFFH	-1024... 1023.96875	0.03125	[Nm]
Basic lambda setpoint for catalyst heating during idle					
IP_LAMB_CH_PORT_AV	6*6	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
LDPM_N_32_5_EXTC	6	0... FFH	0... 8160	32	[rpm]
LDPM_TQI_REQ_SLOW_2_EXTC	6	0... FFFFH	-1024... 1023.96875	0.03125	[Nm]
Basic lambda setpoint for catalyst heating with VCM port flap active.					
IP_LAMB_CH_RGL	8*8	0... 7FFFH	0... 31.9990234375	976.563e-6	[-]
LDPM_N_32_4_EXTC	8	0... FFH	0... 8160	32	[rpm]
LDPM_TQI_REQ_SLOW_1_EXTC	8	0... FFFFH	-1024... 1023.96875	0.03125	[Nm]
Basic lambda setpoint for catalyst heating					
IP_T_AST_THD_LAMB_CH	8	0... FFFFH	0... 6553.5	0.1	[s]
LDPM_TCO_ST_1_EXTC	8	0... FEH	-48... 142.5	0.75	[°C]
Threshold of (corrected) time after start for catalyst heating with lean mixture					

General Information

general specification

For catalyst heating during the engine warm-up after start the engine can be operated with a calibratable air-fuel-ratio.

Application Conditions

Initialization: RST
Recurrence: 20MS
Activation: always
Deactivation: never

Function description

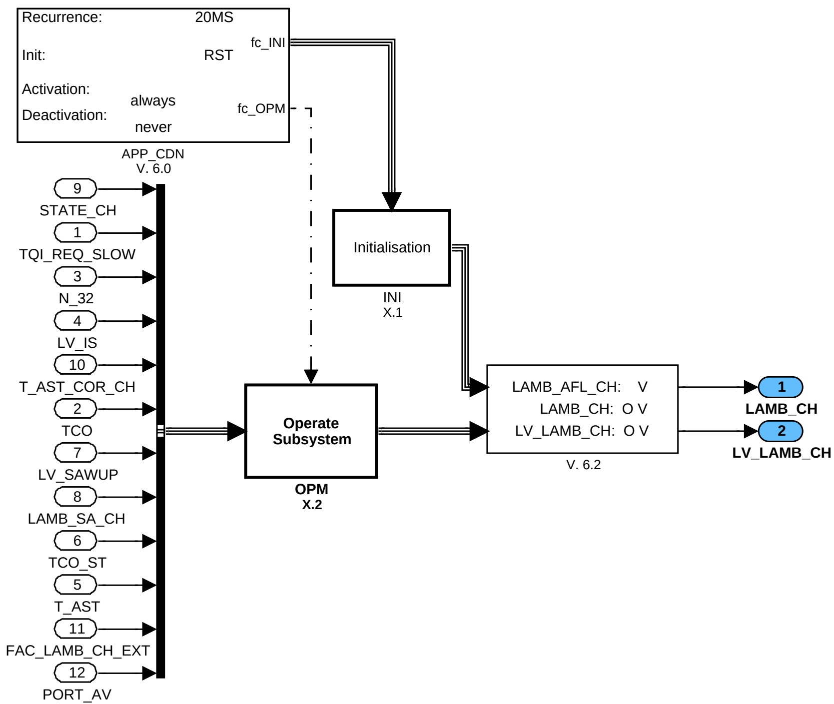


Figure 1:

7.15.1 Initialisation

7.15.1.1 At reset

The signals are initialised at reset.

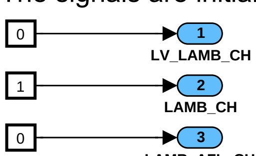


Figure 2:

7.15.2 Recurrence 20 ms

7.15.2.1 Activation condition

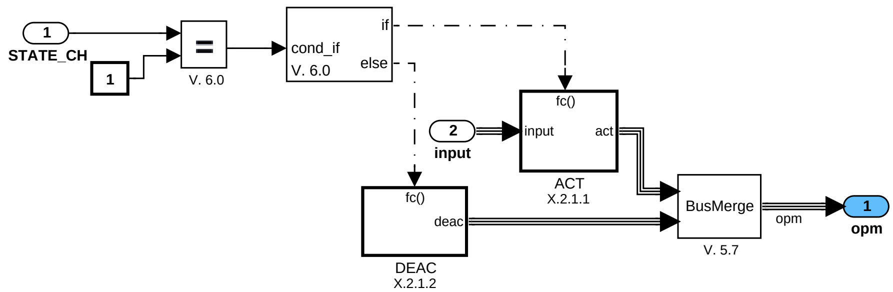


Figure 3:

7.15.2.1.1 Calculation of LAMB_AFL_CH

7.15.2.1.1.1 Part 1

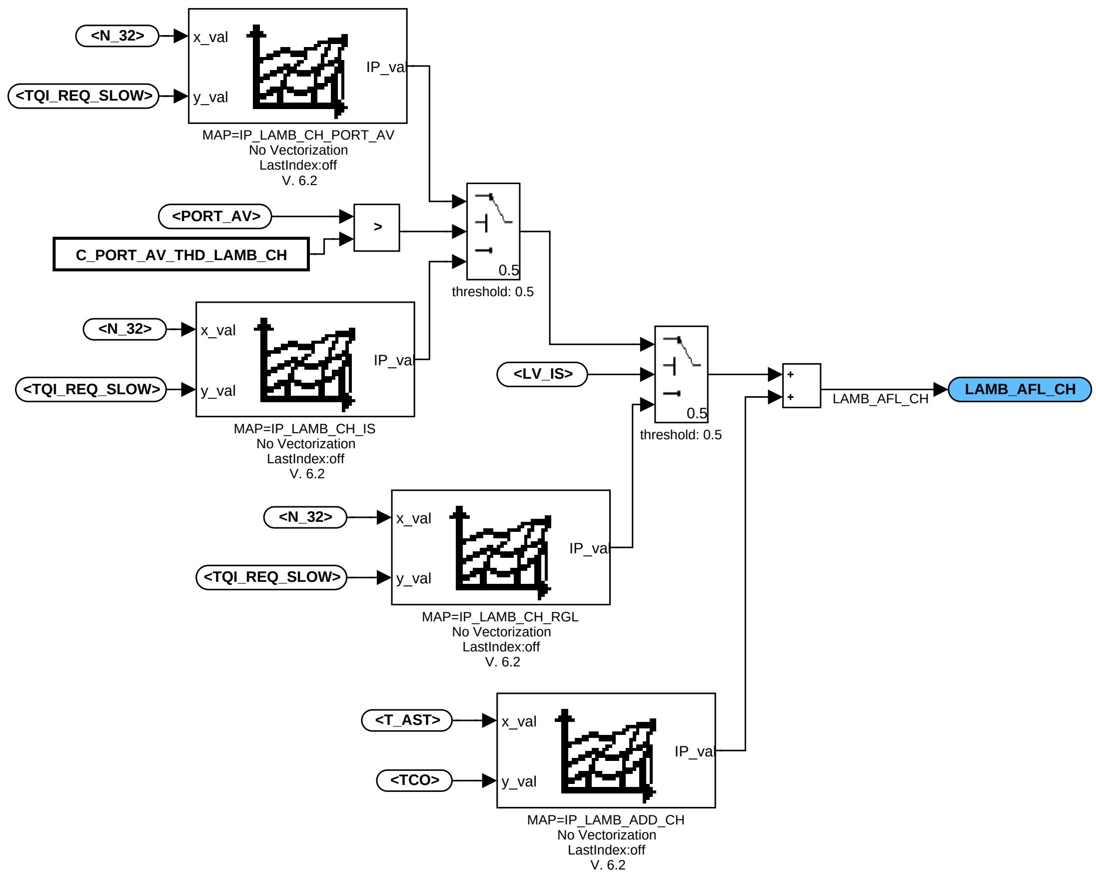


Figure 4:

7.15.2.1.1.2 Part 2

Via FAC_LAMB_CH_EXT it is possible to adapt the difference to lambda = 1. If secondary air is active, LAMB_SA_CH is used. LAMB_CH is used if LV_LAMB_CH is 1 calibratable via IP_T_AST_THD_LAMB_CH.

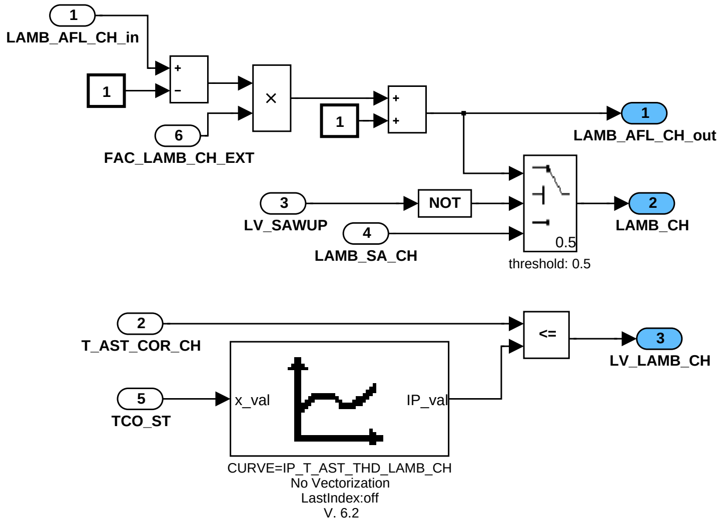


Figure 5:

7.15.2.1.2 At deactivation

7.15.2.1.2.1 Initialisation at deactivation

The signals are initialised at deactivation.

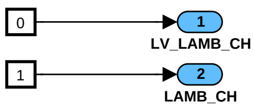


Figure 6:

7.16 Catalyst heating correction (Appl.Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_LAMB_CH_EXT	V/O	0...FFH	0...1.99218	0.0078125	[-]
External correction for lambda setpoint for catalyst heating					

General information:

Output is just initialized. Function for lambda setpoint for cataylst heating can be added.

Application conditions:

Initialisation: *At reset:*

FAC_LAMB_CH_EXT = 1

Comment: 1 is phys. value.

7.17 Fuel mass setpoint of post injection for catalyst heating

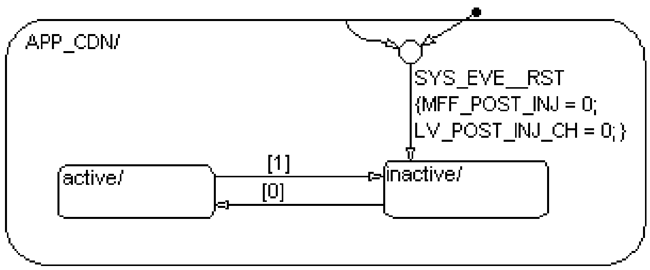
Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_POST_INJ	O/V	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
Mass of fuel flow of post injection in stratified mode					
LV_POST_INJ_CH	O/V	0...1H	0...1	1	-
Catalyst heating by post injection is active					

7.17.1 General information

The output data is just initialized.

Application Condition



Function Description

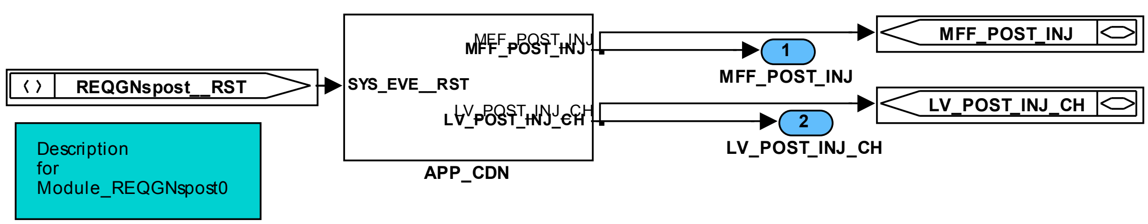


Figure 7 REQGNspost0

7.18 Turbo charger overheating prevention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_TUR_OHP[NC_CBK_EX_NR]	O/V	0...7FFFH	0...31.9990234	9.76563E-4	-
Lambda setpoint for turbo charger overheating prevention					
LV_LAMB_TUR_OHP[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Indicates if turbine overheating prevention is active					
LV_LAMB_TUR_OHP_BOL[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Flag for bottom limit of enrichment (I-part or sum) reached					
TEG_TUR_OHP_DIF[NC_CBK_EX_NR]	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Temperature difference (control deviation) for overheating prevention					
TEG_TUR_OHP_SP	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Turbine temperature setpoint for overheating prevention					
LAMB_TUR_OHP_P[NC_CBK_EX_NR]	V	8000...7FFFH	-32...31.9990234	9.76563E-4	-
P-part of lambda for turbine overheating prevention					
LAMB_TUR_OHP_I[NC_CBK_EX_NR]	V	0...FFFFH	0...1.99996948	3.05176E-5	-
I-part of temperature-controller via lambda for turbine overheating prevention					

Input data:

LV_ST_END	NC_CBK_EX_NR	LAMB_BAS[NC_CBK_EX_NR]	LAMB_SP[NC_CBK_EX_NR]
LV_TQI_TEG_MAX_TUR	TEG_DYN_UP_TUR[NC_CBK_EX_NR]	MAF_STK_SP	N

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_TEG_TUR_OHP_SP	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Filter constant for turbine temperature setpoint for overheating prevention calculation					
C_FAC_LAMB_TUR_OHP_I_NEG	1	0...FFFFH	0...0.00390619	5.96046E-8	1/K
I-part of controler for negative deviation					
C_FAC_LAMB_TUR_OHP_I_POS	1	0...FFFFH	0...0.00390619	5.96046E-8	1/K
I-part of indicated torque controler for positive control deviation					
C_FAC_LAMB_TUR_OHP_P_NEG	1	0...FFFFH	0...0.015625	2.38422E-7	1/K
P-part of controler for negative deviation					
C_FAC_LAMB_TUR_OHP_P_POS	1	0...FFFFH	0...0.015625	2.38422E-7	1/K
P-part of controler for positive deviation					
C_LAMB_TUR_OHP_I_MIN	1	0...FFFFH	0...1.99996948	3.05176E-5	-
Minimum value for I-part of turbo charger overheating protection					
C_LAMB_TUR_OHP_MIN	1	0...7FFFH	0...31.9990234	9.76563E-4	-
Minimum lambda value for turbo charger overheating prevention					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_LAMB_TUR_OHP_RAMP_DOWN	1	0...1H	0...1	1	-
Flag for ramp down LAMB_TUR_OHP at least 1 digit if TEG_TUR_OHP_DIF<0					
LC_LAMB_TUR_OHP_RAMP_UP	1	0...1H	0...1	1	-
Flag for ramp up LAMB_TUR_OHP at least 1 digit if TEG_TUR_OHP_DIF>0					
IP_TEG_TUR_UP_SP	8x8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDP_N_IP_TEG_TUR_UP_SP	8	0...1FE0H	0...8.16E+3	1	rpm
LDP_MAF_STK_SP_IP_TEG_TUR_UP_SP	8	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
Setpoint of turbine temperature for overheating prevention					

7.18.1 General information

If the actual temperature before turbine TEG_DYN_UP_TUR exceeds TEG_TUR_OHP_SP (calibratable via IP_TEG_TUR_UP_SP) the logical value LV_LAMB_TUR_OHP is set to 1 and a P-I-controller is activated. LAMB_TUR_OHP is ramped down to decrease the turbine temperature and afterwards (at uncritical temperatures) ramped up until LAMB_BAS to avoid a step in LAMB_SP.

Application Condition

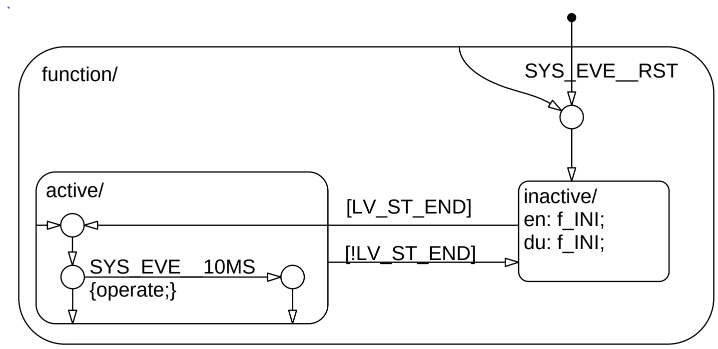


Figure 8 EXTC_ISPCLohc0/ APP_CDN/ Chart

Function Description

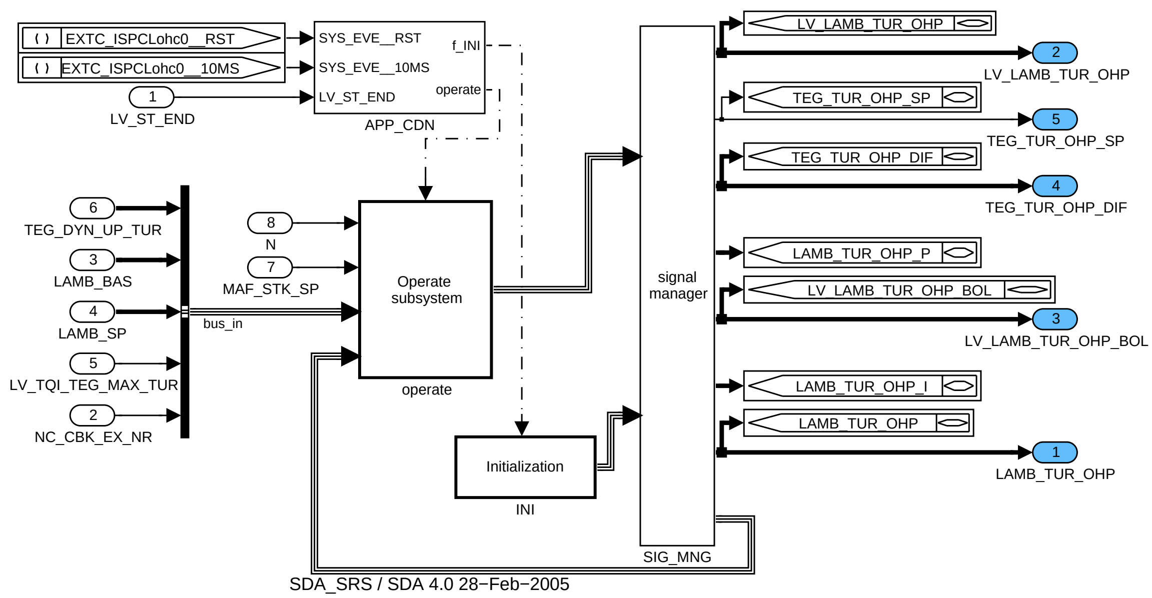


Figure 9 EXTC_ISPCLohc0

7.18.1.1 Initialization

The values are set to passive values at Reset and during engine stop.

7.18.1.2 Operate subsystem

Via IP_TEMP_TUR_OHP_SP the target temperature for the controller is generated. If this value is increasing it is filtered via C_CRLC_TEG_TUR_OHP_SP.

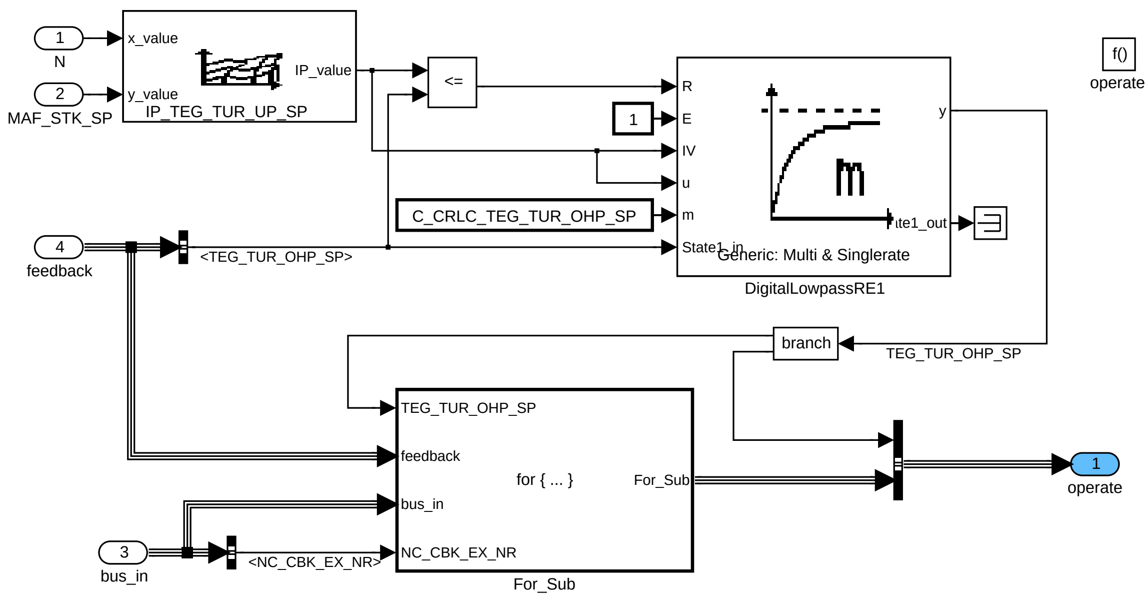


Figure 10 EXTC_ISPCLohc0/ operate

For – loop system

The calculations in this subsystem and lower are done for each exhaust bank separately –the variables have the index n. TEG_TUR_OHP_DIF is the control deviation and input for the controller. If it has negative values the temperature is too high and the controller is activated. There are two parts of the controller: 1. RAMP_DOWN for too high temperature and 2. RAMP_UP for deactivation ramp if TEG_TUR_OHP_DIF is again in positive range. LAMB_TUR_OHP_I <= LAMB_BAS is used for deactivation - to ramp LAMB_TUR_OHP until LAMB_BAS.

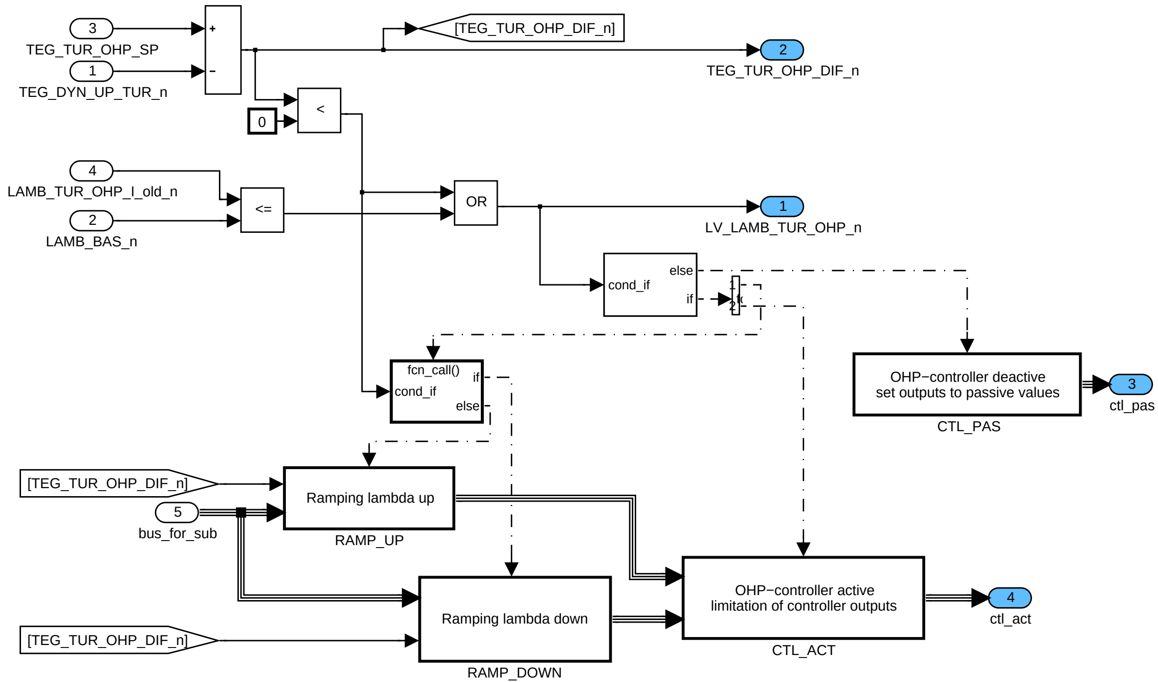


Figure 11 EXTC_ISPCLohc0/ operate/ For_Sub/ SUB

Ramping lambda up

If the temperature is lower than the setpoint the lambda-setpoint is ramped up again via P-Part and I-Part. With LC_LAMB_TUR_OHP_RAMP_UP it is ensured that the I-Part independent from the control deviation increases. The I-part is frozen as long as the torque reduction for overheating prevention is active (LV_TQI_TEG_MAX_TUR=1).

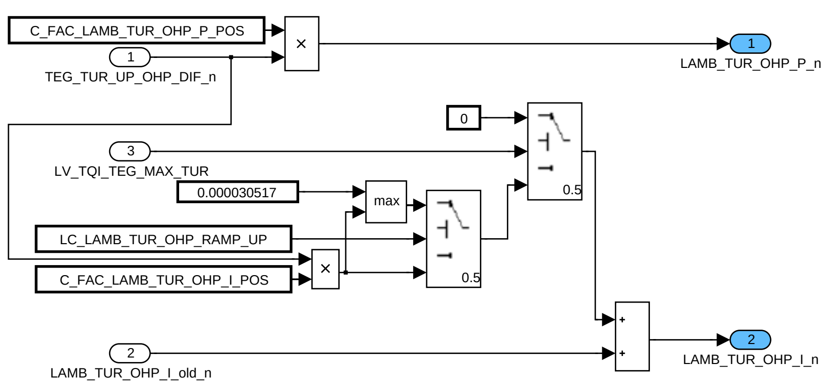


Figure 12 EXTC_ISPCLohc0/ operate/ For_Sub/ SUB/ RAMP_UP/ RAMP_UP_SUB

i

Ramping lambda down

If temperature is in critical range the lambda-setpoint is ramped down via a P-Part and an I-Part to reduce the temperature. Via LC_LAMB_TUR_OHP_RAMP_DOWN it can be ensured that the I-part is ramped down independent from the control deviation (with too small control deviation LAMB_TUR_OHP_I keeps its old value).

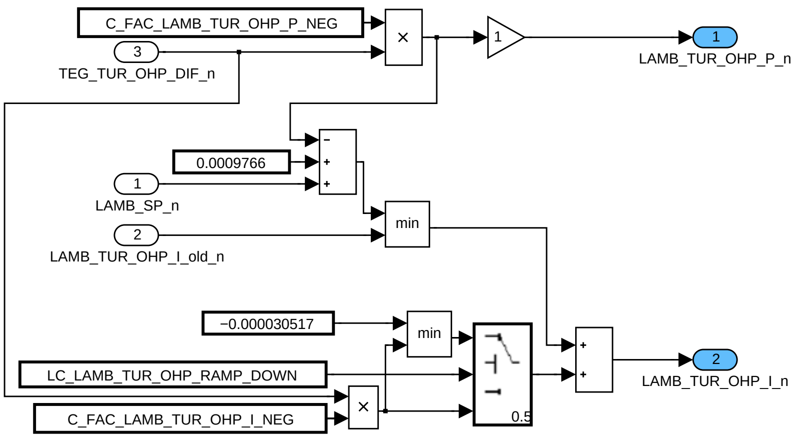


Figure 13 EXTC_ISPCLohc0/ operate/ For_Sub/ SUB/ RAMP_DOWN/ RAMP_DOWN_SUB

OHP-controller active -limitation of controller outputs

During active controller, the I- and P-Part either from RAMP_UP or from RAMP_DOWN subsystem are added. The I-Part and the addition-result are minimum-imitated. If one of this limitations is reached LV_LAMB_TUR_OHP_BOL is set to 1.

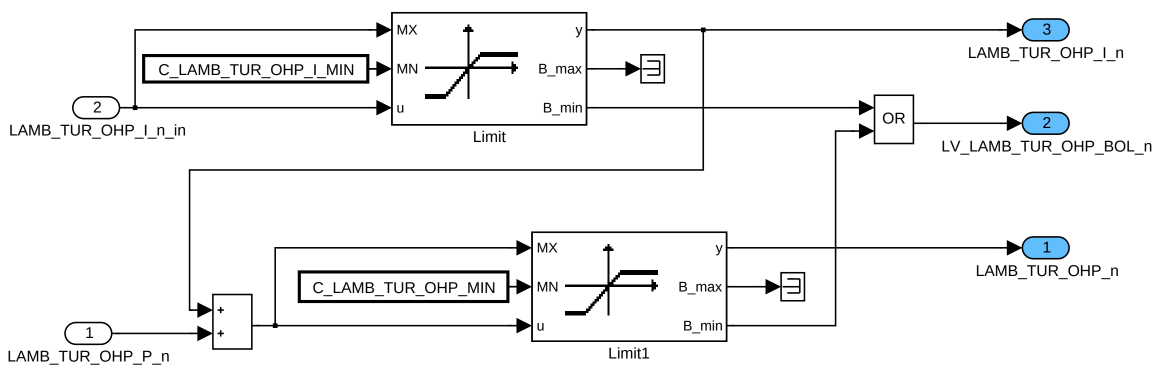


Figure 14 EXTC_ISPCLohc0/ operate/ For_Sub/ SUB/ CTL_ACT/ CTL_ACT_SUB

OHP-controller deactive -set outputs to passive values

This subsystem is calculated, if the controller is deactivated. The outputs are set to passive values.

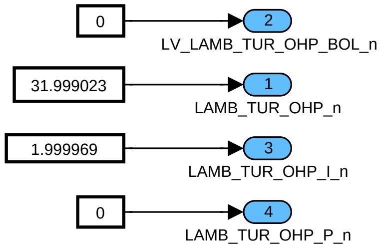


Figure 15 EXTC_ISPCLohc0/ operate/ For_Sub/ SUB/ CTL_PAS/ CTL_PAS_SUB

7.19 Catalyst overheating prevention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LAMB_OHP[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
flag indicating enrichment for overheating prevention					
LAMB_OHP[NC_CBK_EX_NR]	O/V	0...7FFFH	0...31.9990234	9.76563E-4	-
Lambda setpoint for overheating prevention (either for turbocharger or catalyst)					
LAMB_COP[NC_CBK_EX_NR]	O/V	0...7FFFH	0...31.9990234	9.76563E-4	-
Lambda setpoint for catalyst overheating prevention					
LAMB_COP_TMP[NC_CBK_EX_NR][NC_NR_COP_CTL]	V	0...7FFFH	0...31.9990234	9.76563E-4	-
Lambda setpoint for catalyst overheating prevention, matrix: bankselective and for 1 or 2 lambda-controller					
LV_LAMB_COP_TMP[NC_CBK_EX_NR][NC_NR_COP_CTL]	V	0...1H	0...1	1	-
Active lambda catalyst overheating prevention controller, matrix: bankselective and for 1 or 2 lambda-controller					

Input data:

TEMP_COP_CTL[NC_CBK_EX_NR][NC_NR_COP_CTL]	TEMP_COP_ACT[NC_CBK_EX_NR][NC_NR_COP_CTL]	NC_NR_COP_CTL	LV_AST
LV_PUC	LV_SCC[NC_CBK_EX_NR]	LV_ST_END	LAMB_BAS[NC_CBK_EX_NR]
N_32	NC_CBK_EX_NR	MAF_STK_FG_PRED	LV_LAMB_TUR_OHP[NC_CBK_EX_NR]
LAMB_TUR_OHP[NC_CBK_EX_NR]	EFF_IGA_AV	VS	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_LAMB_COP[NC_NR_COP_CTL]	1	0...FFFFH	0...0.003922	5.98459E-8	1/K
Factor for I-controller of catalyst overheating prevention					
C_LAMB_COP_PAS	1	0...7FFFH	0...31.9990234	9.76563E-4	-
passive value for the lambda catalyst overheating prevention					
C_TEMP_COP_ACT_THD[NC_NR_COP_CTL]	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Temperature to activate the catalyst overheating controller					
C_TEMP_COP_CTL_SP[NC_NR_COP_CTL]	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Temperature setpoint of controller					
LC_LAMB_PCTL_INI_COP[NC_NR_COP_CTL]	1	0...1H	0...1	1	-
Use IP_LAMB_COP_PCTL only for initialisation of COP-controller					
IP_FAC_VS_TEMP_COP[NC_NR_COP_CTL]	6	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_VS_IP_FAC_VS_TEMP_COP	6	0...FFH	0...255	1	km/h
VS factor for correction of offset for activate/setpoint temperatures for COP					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_LAMB_COP_MIN	6x6	0...7FFFH	0...31.9990234	9.76563E-4	-
LDP_N_32_IP_LAMB_COP_MIN	6	0...FFH	0...8.16E+3	32	rpm
LDP_EFF_IGA_AV_IP_LAMB_COP_MIN	6	0...FFFFH	0...1.99996948	3.05176E-5	-
Minimum lambda value for catalyst overheating protection					
IP_LAMB_COP_PCTL	6x6	0...7FFFH	0...31.9990234	9.76563E-4	-
LDP_N_32_IP_LAMB_COP_PCTL	6	0...FFH	0...8.16E+3	32	rpm
LDP_MAF_EX_IP_LAMB_COP_PCTL	6	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
Precontrolled lambda for catalyst overheating preventioin					
IP_TEMP_COP_ACT_THD_ADD[NC_NR_COP_CTL]	6x6	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_N_32_IP_TEMP_COP	6	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_EX_IP_TEMP_COP	6	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
Temperature offset for activate the of catalyst overheating controller					
IP_TEMP_COP_CTL_SP_ADD[NC_NR_COP_CTL]	6x6	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_N_32_IP_TEMP_COP	6	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_EX_IP_TEMP_COP	6	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
Temperature setpoint offset for catalyst overheating controller					

7.19.1 General information

If the temperature in the exhaust gas line exceeds a certain limit, the catalyst converter or even the exhaust gas line can be destroyed. To reduce the temperature the lambda is ramped down.

Application Condition

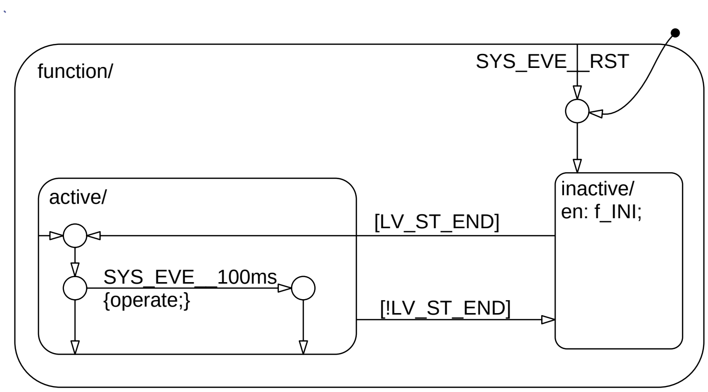


Figure 16 EXTC_REQGNOHL0/ APP_CDN/ Chart

Function Description

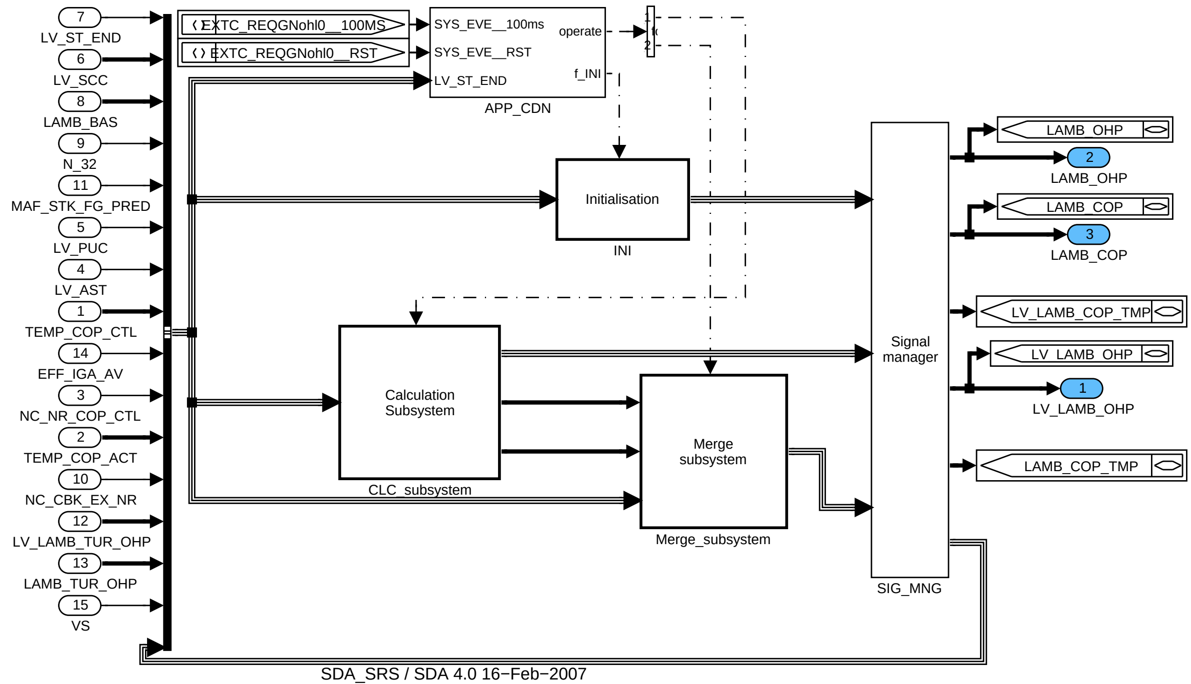


Figure 17 EXTC_REQGNOHL0

7.19.1.1 Initialisation

The values are initialised at reset. Each value of LAMB_COP_TMP is initialised with 32 and each of LV_LAMB_COP with 0.

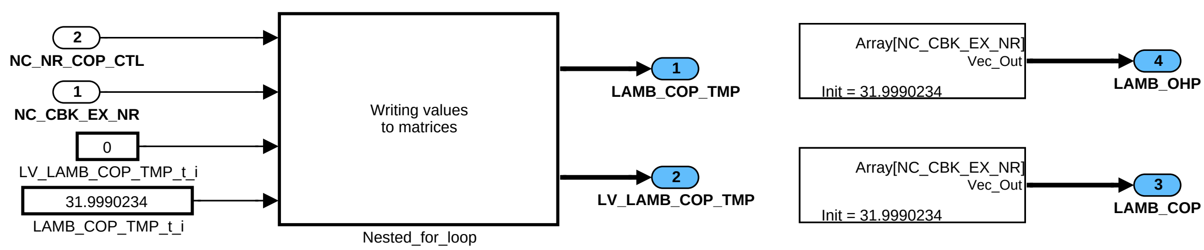


Figure 18 EXTC_REQGNOHL0/ INI/ SUB

7.19.1.2 Calculation subsystem

LAMB_COP_MIN is the minimum lambda-value for overheating prevention. LAMB_COP_PCTL is a pre-controlled value and is used for 1st value at controller activation or as additive part during active controller. This is selectable via LC_LAMB_PCTL_INI_COP.

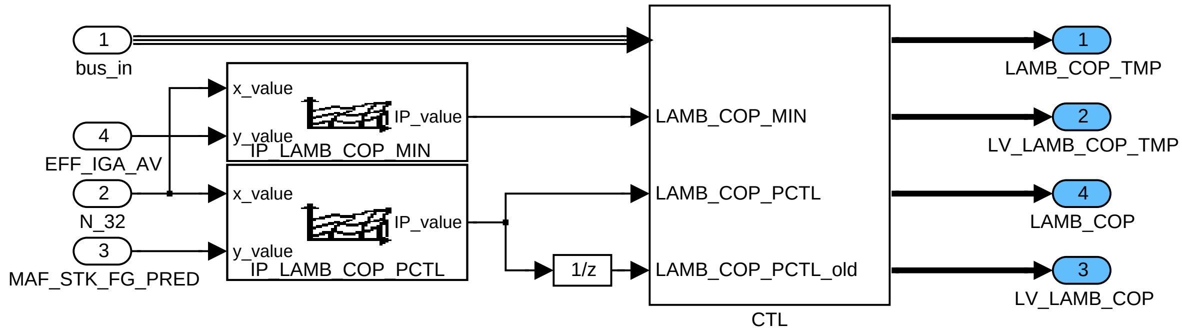


Figure 19 EXTC_REQGNOHL0/ CLC_subsystem/ SUB

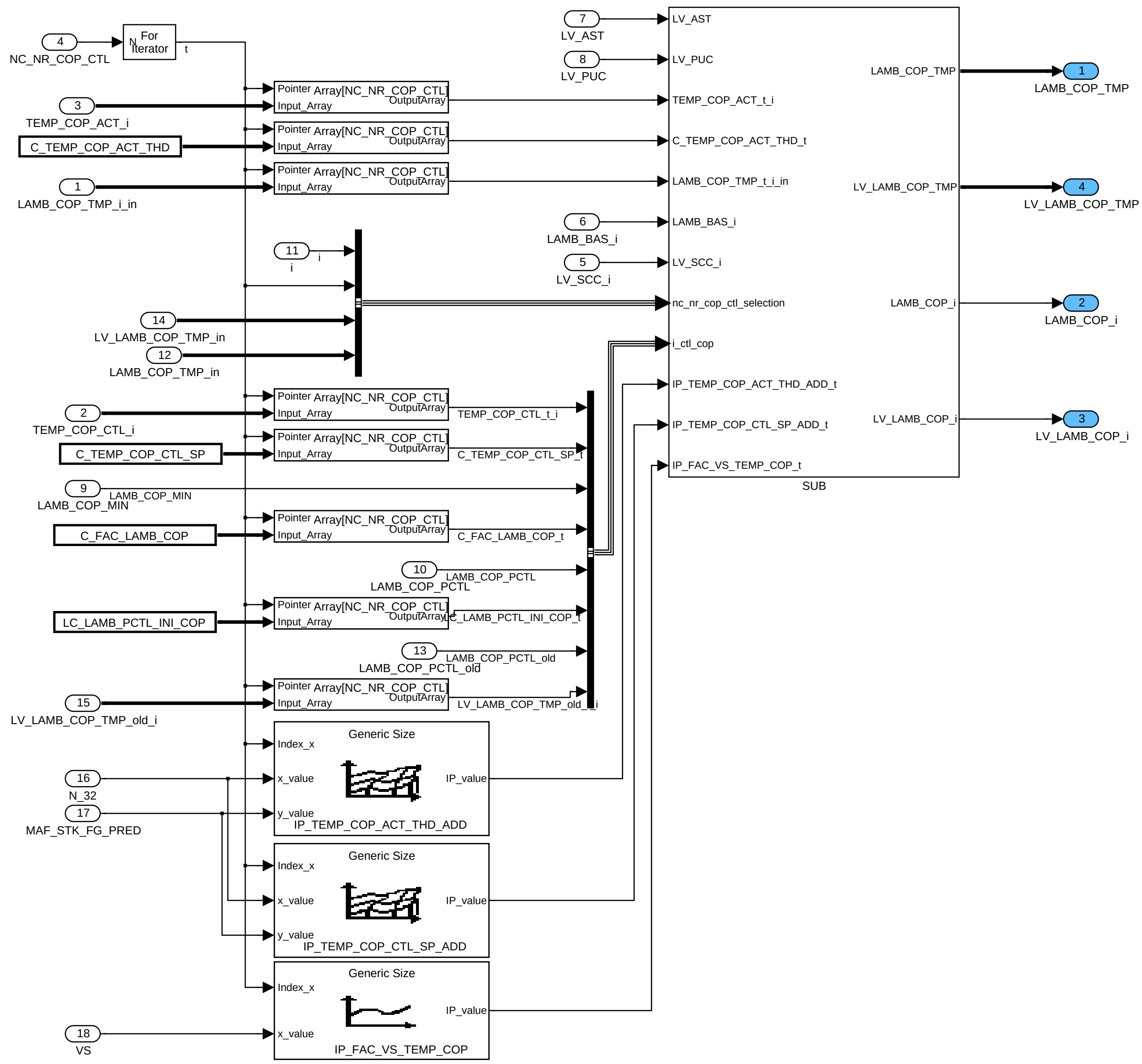


Figure 20 EXTC_REQGNOHL0/ CLC_subsystem/ SUB/ CTL/ bank_selection/ NC_NR_COP_CTL_selection

Activation/Deactivation

This function is inside a nested for-loop-system - one loop from 1 to NC_CBK_EX_NR (index = i) and the other one from 1 to NC_NR_COP_CTL (index = t).

NC_NR_COP_CTL is the number of catalyst overheating prevention controllers. TEMP_COP_CTL and TEMP_COP_ACT can contain 1 or 2 bankselective (NC_CBK_EX_NR) temperatures for e.g. exhaust gas temperature before catalysator and temperature in the catalyst.

The activation of the lambda controllers (for each bank 1 or 2 controllers) is done via TEMP_COP_ACT and C_TEMP_COP_ACT_THD.

If LAMB_COP_TMP_t_i > LAMB_BAS_i the lambda value is ramped up again and the controller can be deactivated.

In the subsystem vector_matrix the scalar LAMB_COP_TMP_t_i is written to a NC_NR_COP_CTL-vector LAMB_COP_INTER (LAMB_COP_i than is the richest lambda value for one exhaust bank) and to a NC_NR_COP_CTL/NC_CBK_EX_NR – dimensional matrix LAMB_COP_TMP. The same way for LV_LAMB_COP_TMP_t_i: LV_LAMB_COP_TMP is the matrix and LV_LAMB_COP is the bank-specific value.

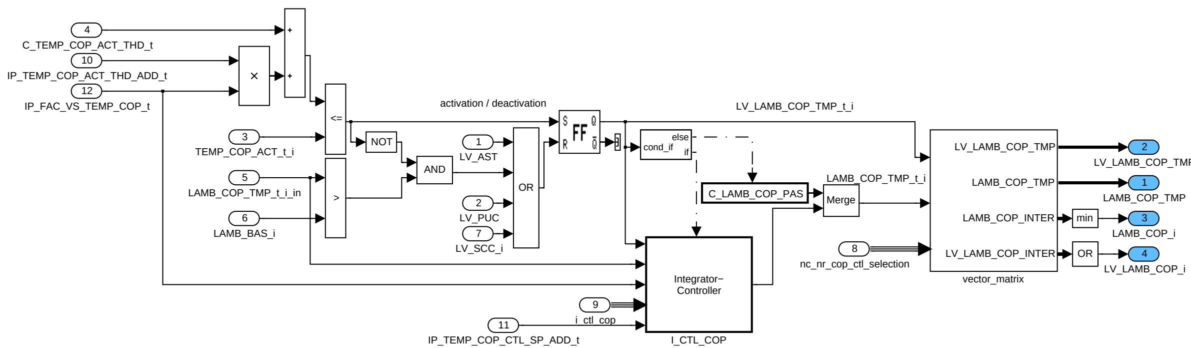


Figure 21 EXTC_REQGNOHL0/ CLC_subsystem/ SUB/ CTL/ bank_selection/ NC_NR_COP_CTL_selection/ SUB

Integrator-Controller

This I – controller is developed to be used with static temperatures, because in this way the control path is eliminated to avoid undesirable oscillation.

C_TEMP_COP_CTL_SP_t minus TEMP_COP_CTL_t_i is the control deviation.

The block "IntegratorKREL" integrates the Product of C_FAC_LAMB_COP_TMP_t and the control deviation. The output is limited to LAMB_COP_MIN and LAMB_COP_PAS.

If LC_ LAMB_PCTL_INI_COP is 1 the controller works as I-controller. If the LC is 0, the precontrolled part via IP_LAMB_COP_PCTL is also used.

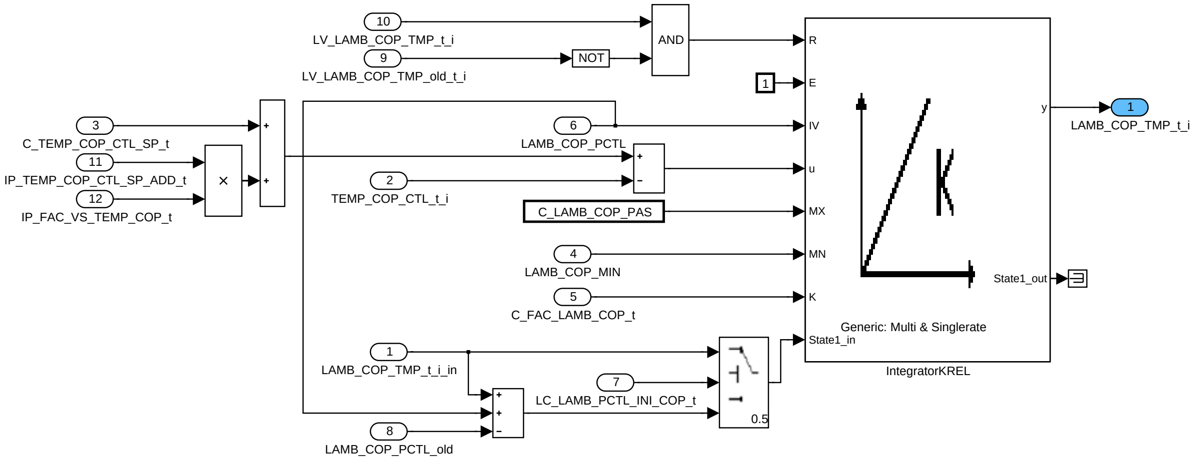


Figure 22 EXTC_REQGNOHL0/ CLC_subsystem/ SUB/ CTL/ bank_selection/ NC_NR_COP_CTL_selection/ SUB/ I_CTL_COP/ SUB

7.19.1.3 Merge subsytem

If LV_LAMB_TUR_OHP (turbine overheating prevention is active) or LV_LAMB_COP (catalyst overheating prevention is active) is 1, the global bit LV_LAMB_OHP is set to 1.

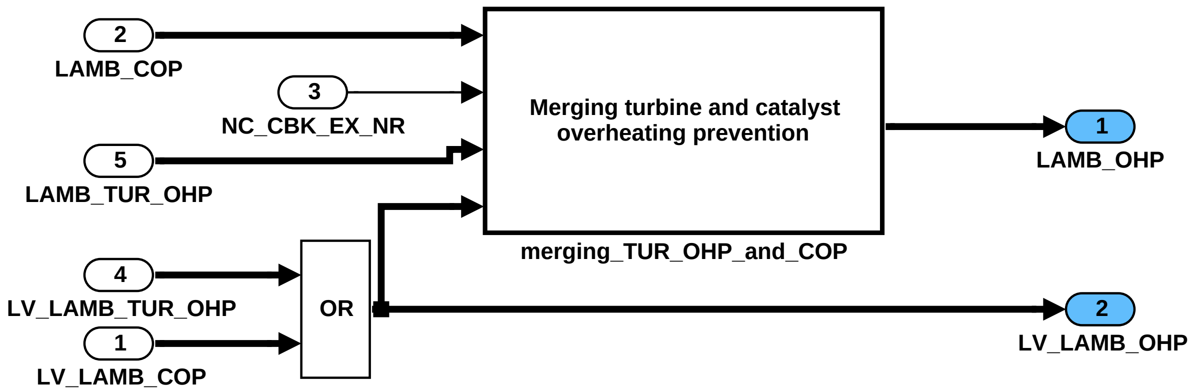


Figure 23 EXTC_REQGNOHL0/ Merge_subsystem/ SUB

<u>Merging turbine and catalyst overheating prevention</u>

This function is inside a FOR-loop system (from 1 to NC_CBK_EX_NR). If LV_LAMB_OHP_i is 1 the richest lambda value is used for the global LAMB_OHP_i.

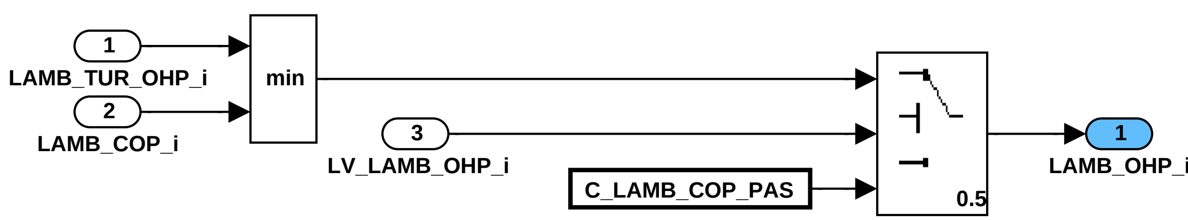


Figure 24 EXTC_REQGNOHL0/ Merge_subsystem/ SUB/ merging_TUR_OHP_and_COP/ SUB

7.20 Overheating prevention (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_COP_CTL[NC_CBK_EX_NR][NC_NR_COP_CTL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Actual temperatures for catalyst overheating prevention - controllers					
TEMP_COP_ACT[NC_CBK_EX_NR][NC_NR_COP_CTL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Actual temperatures for activation of catalyst overheating prevention - controllers					
LV_PUC_INH_TEMP_CAT	O/V	0...1H	0...1	1	-
Flag to inhibit PUC operation due to too high TEMP_CAT					
TEG_TUR_OHP_GRD	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Temperature gradient for activation of overheating prevention					
MAF_KGH_INT_PUC_INH_TEMP_CAT	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
MAF integral as map-input for activation of PUC-inhibition for overheating prevention					

Input data:

NC_CBK_EX_NR	LAMB_SP[NC_CBK_EX_NR]	NC_NR_COP_CTL	MAF_KGH
TEG_DYN_UP_CAT[NC_CBK_EX_NR]	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	TEMP_CAT_STAT_MDL[NC_CBK_EX_NR]	LV_CT
TEG_DYN_UP_TUR[NC_CBK_EX_NR]	TEG_STAT_UP_CAT_MDL[NC_CBK_EX_NR]		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_MAF_KGH_DEAC_PUC_INH	1	0...FFH	0...0.99609375	0.00390625	-
Factor for MAF_KGH to decrease MAF-integral					
C_MAF_KGH_INT_MAX_PUC_INH_TEMP	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Maximum MAF-integral as input for PUC - inhibition map due to overheating prevention					
C_TEMP_CAT_HYS_PUC_INH	1	0...7FFH	0...2.047E+3	1	K
Temperature hysteresis for PUC-inhibition					
IP_TEMP_CAT_THD_PUC_INH	6	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDP_MAF_KGH_INT_IP_TEMP_PUC_INH	6	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Catalyst temperature threshold for PUC inhibition					

7.20.1 General information

TEMP_COP_ACT and TEMP_COP_CTL are used in the module "Catalyst overheating prevention". The exhaust gas temperature before catalysator and the catalysator temperature are controlled for each bank – therefore the outputs are matrices (x-dimension: NC_CBK_EX_NR; y-dimension NC_NR_COP_CTL).

Application Condition

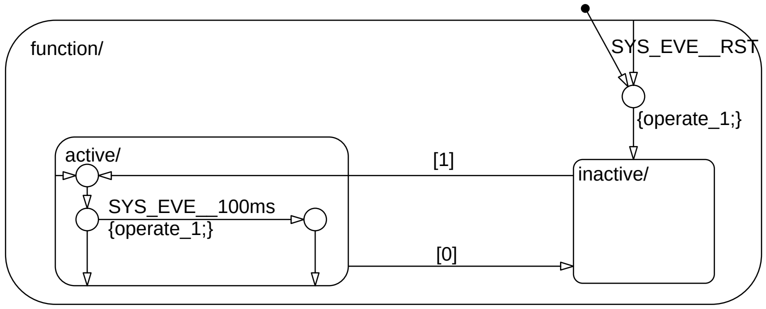


Figure 25 EXTC_ISPCLohpai0/ APP_CDN/ Chart

Function Description

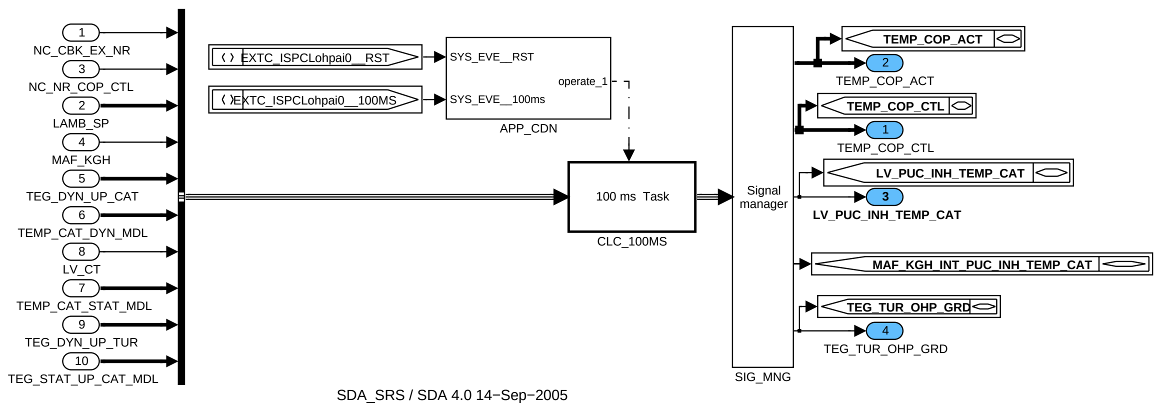


Figure 26 EXTC_ISPCLohpai0

7.20.1.1 100 ms Task

LV_PUC_INH_TEMP_CAT is set depending on catalyst temperature thresholds and reset after passing a hysteresis or if the clutch is opened or the driver request is 0. The activation threshold depends on a special MAF-integral that increases during mixture enrichment (HC-storage) and decreases in stoichiometric or even lean mixture.

TEG_TUR_OHP_GRD is used for turbine protection function activation.

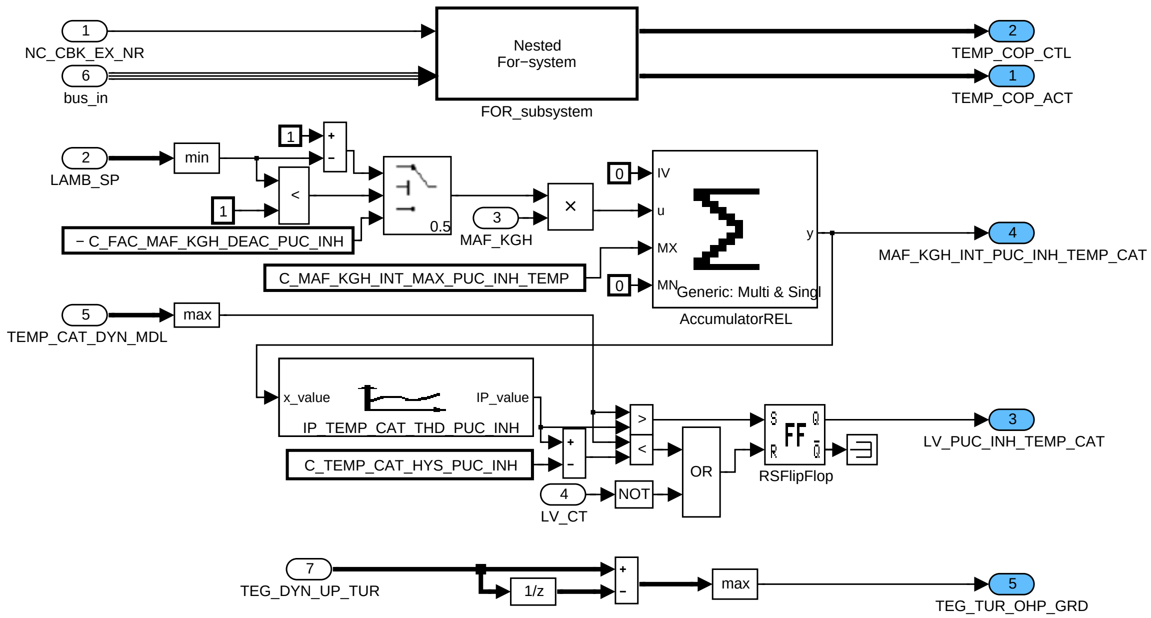


Figure 27 EXTC_ISPCLohpai0/ CLC_100MS/ SUB

Nested For-system

There are 2 temperature-controller for catalyst overheating prevention. The 1st one controls the temperatures before the catalyst and the 2nd one the catalyst temperature itself. For activation dynamic temperatures are used (TEG_DYN_UP_CAT and TEMP_CAT_DYN_MDL) and for controlling static temperatures are used (TEG_STAT_UP_CAT and TEMP_CAT_STAT_MDL).

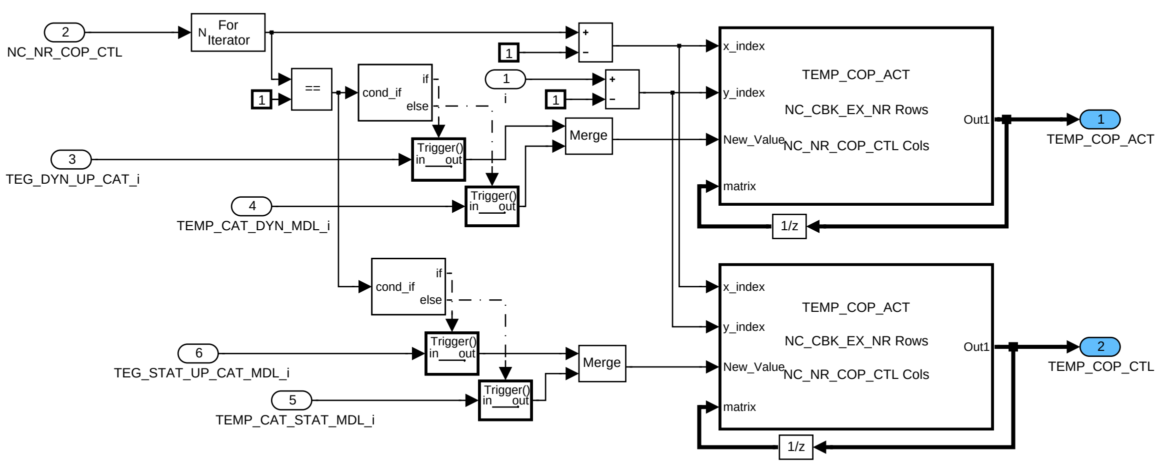


Figure 28 EXTC_ISPCLohpai0/ CLC_100MS/ SUB/ FOR_subsystem/ nested_FOR_system

7.21 Full Load Enrichment

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_FL	O/V	0...7FFFH	0...31.999	0.00097656	-
Lambda Requirement for Full Load Enrichment					
T_FL	V	0...FFFFH	0...1310.7	0.02	s
Time elapsed in FL					

Input data:

LV_FL	LV_ST_END	N_32	LAMB_SP_MV
VS	GEAR	TCO	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LAMB_FL_PAS	1	0...7FFFH	0...31.999	0.00097656	-
Passive Value for Full Load Enrichment					
IP_LAMB_FL	12	0...7FFFH	0...31.999	0.00097656	-
LDP_N_32_IP_LAMB_FL	12	0...FFH	0...8160	32	rpm
LDPM_TCO__IP_LAMB_FL	8	0...FEH	-48...142.5	0.75	C
Lambda Full Load Enrichment					
IP_LAMB_DEC_FL	4	0...7FFFH	0...31.999	0.00097656	-
LDP_N_32_IP_LAMB_DEC_FL	4	0...FFH	0...8160	32	rpm
Lambda decrement for gradient limitation during full load enrichment					
IP_LAMB_FL_IS	12	0...7FFFH	0...31.999	0.00097656	-
LDP_N_32_IP_LAMB_FL_IS	12	0...FFH	0...8160	32	rpm
LDPM_TCO__IP_LAMB_FL	8	0...FEH	-48...142.5	0.75	C
Lambda Full Load Enrichment at vehicle stop racing					
IP_LAMB_DEC_FL_IS	4	0...7FFFH	0...31.999	0.00097656	-
LDP_N_32_IP_LAMB_DEC_FL_IS	4	0...FFH	0...8160	32	rpm
Lambda decrement for gradient limitation during full load enrichment at vehicle stop racing					
C_T_DLY_LAMB_IS	1	0...FFFFH	0...1310.7	0.02	s
Full load enrichment time delay before the vehicle stop racing enrichment					

7.21.1 General Information

In order to maximize the torque output of the engine at higher loads, it is possible to operate the engine with a richer fuel/air mixture. If full load conditions are detected (i.e. LV_FL=1), LAMB_FL will directly impact the lambda setpoint in the module ‘Lambda coordination’.

LAMB_FL is calculated from the map IP_LAMB_FL. In order to reduce torque jumps at the transition to Full load, the algorithm offers the possibility to define a ramp function for LAMB_FL by means of the map IP_LAMB_DEC_FL.

Function Description

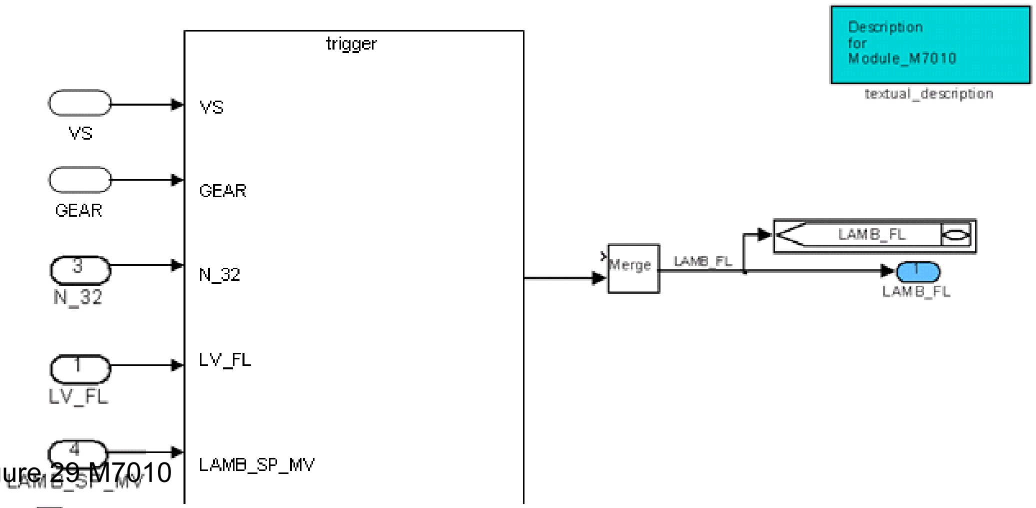


Figure 29 M7010

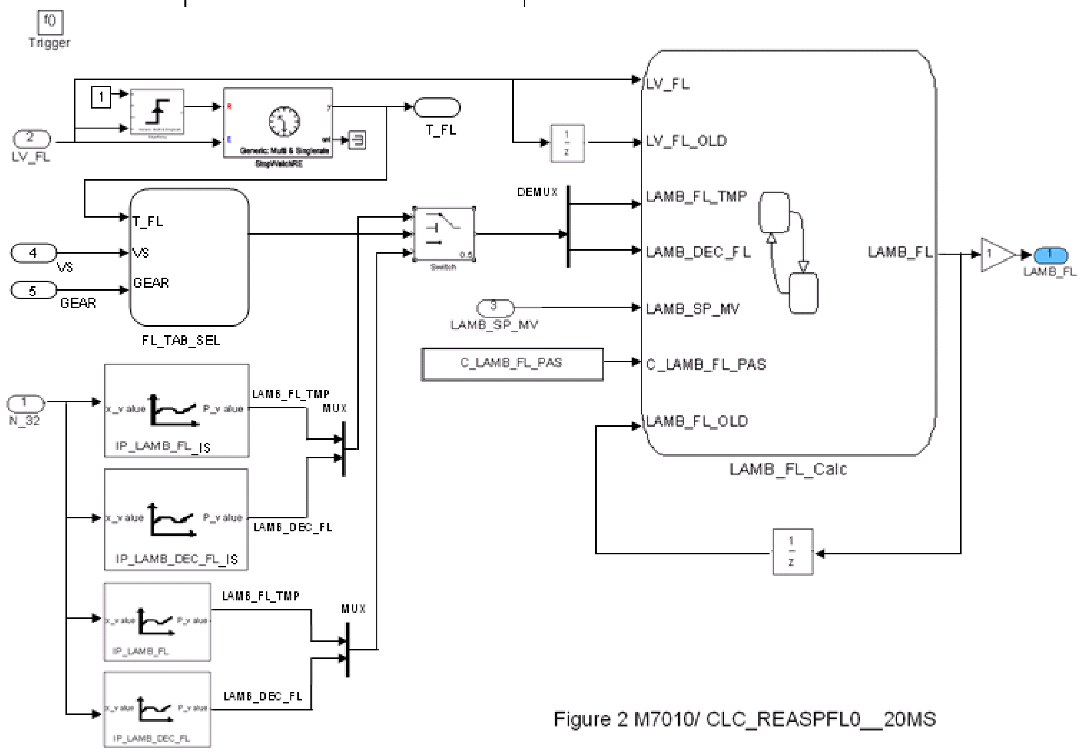


Figure 2 M7010/ CLC_REASPFL0__20MS

Figure 30 M7010/ CLC_REASPFL0__20MS

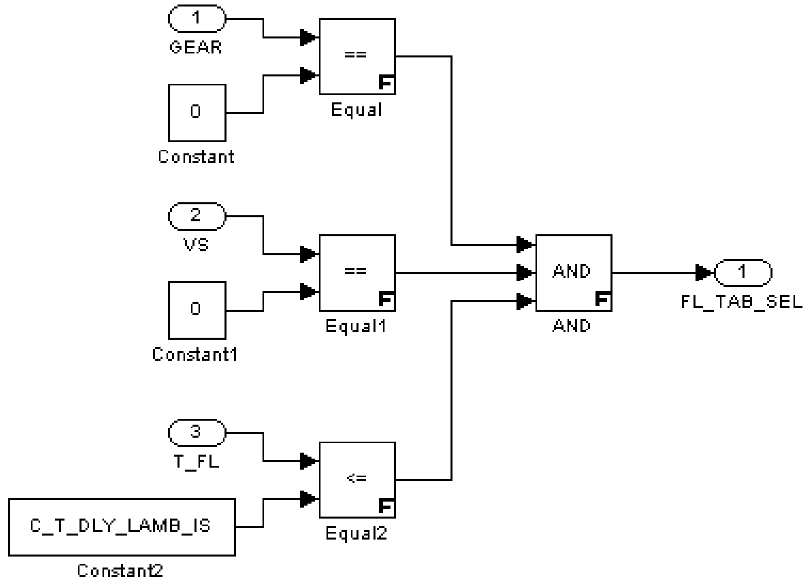
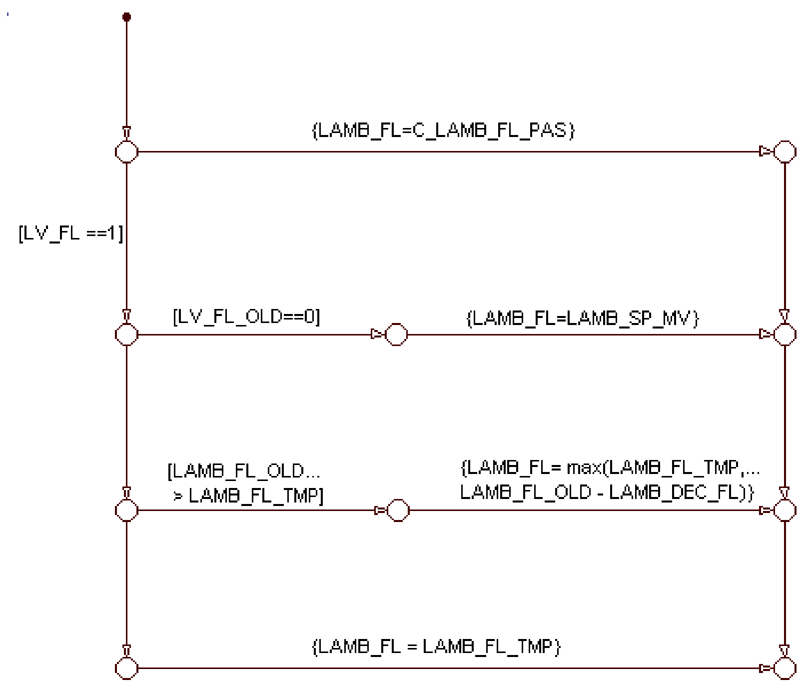


Figure 3 M7010/ FL_TAB_SEL



7.21.2 Idle speed correction (TI_IS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_IS_MMV	V/O	80...7FH	-0.5...0.496	1 / 256	-
Filtered Idle speed correction for nominal injection time.					
TI_IS_MMV_1	-	80...7FH	-0.5...0.496	1 / 256	-
Intermediate Filtered Idle speed correction for nominal injection time.					
TI_IS	V	80...7FH	-0.5...0.496	1 / 256	-
Idle speed correction for nominal injection time.					
LV_TI_IS_ACT	V	0...1H	0...1	1	-
Activation condition for injection time Idle speed correction.					
TI_IS_MMV_MAX	V/O	80...7FH	-0.5...0.496	1 / 256	-
Maximum filtered Idle speed correction for nominal injection time.					
N_DIF_COR_MAX	V/O	8000...7FFFH	-32768...+32767	1	Rpm
Maximum engine speed deviation to idle speed setpoint					
LV_TI_IS_MMV_MAX_DEAC	V/O	0...1H	0...1	1	-
Activation condition for maximum injection time Idle speed learning.					
LV_TI_IS_MMV_MAX_FDOUT	V	0...1H	0...1	1	-
Fade out condition of maximum injection time Idle speed learning due to gear change.					
CTR_TI_IS_MMV_MAX_FDOUT	V	0 ... FFFF	0... 65535	1	
Timer to deactivate the maximum injection time Idle speed learning due to gear change					
N_DIF_COR_2	V	8000...7FFFH	-32768...+32767	1	Rpm
N_DIF_COR corrected with mileage accumulation					
DIST_TI_IS	-	0000...FFFFH	0...+655350	10	km
Distance acculculator for idle speed correction					
LV_TI_IS_MMV_FDOUT	V	0...1H	0...1	1	-
Fade out condition of injection time Idle speed learning due to power steering activation					
CTR_TI_IS_MMV_FDOUT	V	0 ... FFFF	0... 65535	1	
Timer to deactivate the injection time Idle speed learning due to power steering activation					

Input data:

LV_IS	LV_REQ_ISC	N_DIF_COR	TQ_ADD_CH
TCO_ST	LV_ERR_TPS	LV_ERR_TPS_PLAUS	LV_ERR_MAF
LV_ERR_LOAD_PLAUS	LV_ERR_ISC	LV_ERR_ISA_1	LV_ERR_ISA_2
LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_MEC_CPS	LV_ERR_IV[NC_CYL_NR]	LV_ERR_MIS[NC_CYL_NR]
LV_ERR_CPS	PV_AV	GEAR	DIST
VS	LV_N_LIM_ETC_LIH	LV_ERR_MAP	LV_FQ_ST_AD
LV_PSTE	PSP	TCO	LV_ERR_TCO

FUNCTION DESCRIPTION:

General information:

In order to achieve an engine speed stabilization at idle (LV_IS), a mixture correction can be performed as soon as the idle speed control is active.

Injection time correction is calculated versus :

- Deviation to Nominal Idle Speed N_DIF_COR
- Catalyst Heating Torque Reserve TQ_ADD_CH (Slightly Lean Lambda can lead to engine speed undershoot in case of strong CH)

Two TCO_ST zones are defined in order to calibrate different corrections for emission area and for cold start area.

In order to avoid Engine Speed small oscillations when TI_IS is active, final TI-correction is filtered: in decreasing direction if TI_IS > 0, in increasing direction if TI_IS < 0.

Injection time correction is disabled in case of relevant failure.

Application conditions:

Activation : all engine operating states

Initialization:

Initialization at IG key on : TI_IS = 0
TI_IS_MMV_1 = 0
TI_IS_MMV = 0
LV_TI_IS_ACT= 0
TI_IS_MMV_MAX = 0
N_DIF_COR_MAX= 0
LV_TI_IS_MMV_MAX_DEAC = 0
LV_TI_IS_MMV_MAX_FDOUT= 0
CTR_TI_IS_MMV_MAX_FDOUT = 0

Recurrence **:** Segment

Formula section:

Deactivation

If LV_ERR_TPS = 1 **or** LV_ERR_TPS_PLAUS = 1 **or**
LV_ERR_MAF = 1 **or** LV_ERR_LOAD_PLAUS = 1 **or**
LV_ERR_ISC = 1 **or** LV_ERR_ISA_1 = 1 **or** LV_ERR_ISA_2 = 1 **or**
LV_ERR_CPS = 1 **or** LV_ERR_MEC_CPS = 1 **or**
LV_ERR_IV[NC_CYL_NR] = 1 **or** LV_ERR_MIS[NC_CYL_NR] = 1 **or**
LV_ERR_LS_UP_i = 1 **or** LV_ERR_LOAD_TPS_PLAUS = 1 **or**
LV_ERR_MAP = 1 **or** LV_N_LIM_ETC_LIH = 1 or LV_ERR_TCO = 1 or
TCO < C_TCO_MIN_TI_IS_ACT

Then LV_TI_IS_ACT = 0

Else LV_TI_IS_ACT = 1

Endif

** TI_IS calculation :*

If LV_IS = 1 *(engine operating state "idle" active)*

and LV_REQ_ISC = 1 *(idle speed control correction active)*

then

if N_DIF_COR > 0

and LV_TI_IS_MMV_MAX_FDOUT = 0

and TCO_ST **<** C_TCO_ST_TI_IS_MMV_DEAC

and LV_TI_IS_MMV_MAX_DEAC = 0

Then N_DIF_COR_2 = N_DIF_COR + IP_N_DIF_COR_ADD

Else N_DIF_COR_2 = N_DIF_COR

Endif

if TCO_ST < C_TCO_ST_TI_IS

then TI_IS = IP_TI_IS_1__N_DIF_COR_2__TQ_ADD_CH

else TI_IS = IP_TI_IS_2__N_DIF_COR_2__TQ_ADD_CH

Endif

else TI_IS = 0

Endif

<u> TI_IS_MMV_1 calculation :</u>*

If TI_IS_MMV_1 >= 0

Then

If TI_IS_MMV_1 < TI_IS

Then $TI\_IS\_MMV\_1_N = TI\_IS_N$

Else $TI\_IS\_MMV\_1_N = TI\_IS\_MMV_{N-1} + C\_TI\_IS\_CRLC\_POS * (TI\_IS_N – TI\_IS\_MMV_{N-1})$

Endif

Else

If TI_IS_MMV_1 > TI_IS

or TI_IS > C_TI_IS_POS

Then TI_IS_MMV_1 = TI_IS

Else $TI\_IS\_MMV\_1_N= TI\_IS\_MMV_{N-1}+C\_TI\_IS\_CRLC\_NEG*(TI\_IS_N – TI\_IS\_MMV_{N-1})$

Endif

Endif

- inhibition of TI_IS_MMV calculation at power steering activation condition

If (TI_IS_MMV_1 =< 0 and (LV_PSTE = 0 ->1 or PSP > C_PSP_THD_TI_IS_MMV))

Then launch counter CTR_TI_IS_MMV_FDOUT

(counter contineoulsly increase to reset threshold)

LV_TI_IS_MMV_FDOUT = 1

endif

If CTR_TI_IS_MMV_FDOUT >= C_CTR_TI_IS_MMV_FDOUT

Then LV_TI_IS_MMV_FDOUT = 0

CTR_TI_IS_MMV_FDOUT = 0

endif

<u> Final TI correction : TI_IS_MMV calculation</u>*

If LV_TI_IS_ACT = 1 (no relevant failure)

and │TI_IS_MMV_1│ > C_TI_IS_MMV_MIN (dead band)

and VS < C_VS_MIN_TI_IS (vehicle stopped position)

and TCO < C_TCO_MIN_TI_IS

and LV_TI_IS_MMV_FDOUT = 0 **Then** TI_IS_MMV = TI_IS_MMV_1

Else TI_IS_MMV = 0

Endif

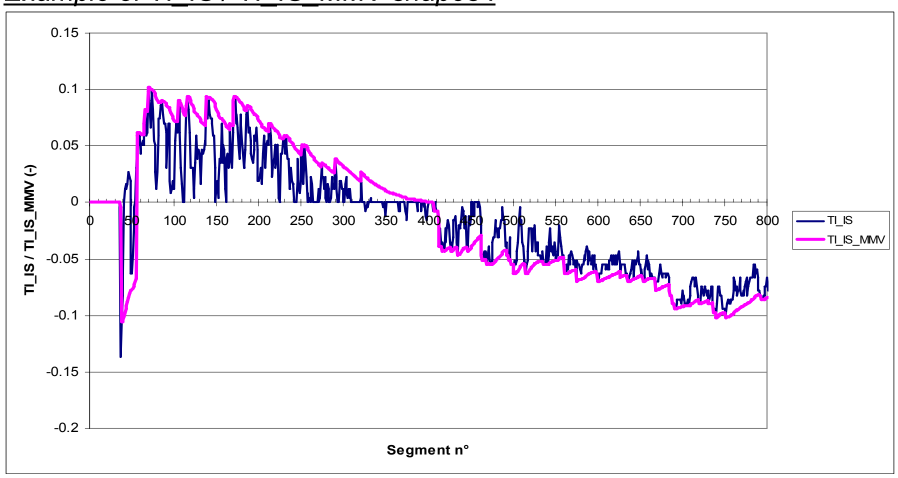
TI_IS_MMV_MAX & N_DIF_COR_MAX calculation

```
If        PV_AV > C_PV_AV_TI_IS_DEAC
Then      LV_TI_IS_MMV_MAX_DEAC = 1
endif

If        GEAR_N <> GEAR_(N-1)
Then      launch timer CTR_TI_IS_MMV_MAX_FDOUT
          LV_TI_IS_MMV_MAX_FDOUT = 1
endif

If        CTR_TI_IS_MMV_MAX_FDOUT > C_CTR_TI_IS_MMV_MAX_FDOUT
Then      LV_TI_IS_MMV_MAX_FDOUT = 0
          CTR_TI_IS_MMV_MAX_FDOUT = 0
endif

If        LV_TI_IS_MMV_MAX_DEAC = 0
    and   LV_FQ_ST_AD = 0
    and   LV_TI_IS_MMV_MAX_FDOUT = 0
    and   TCO_ST < C_TCO_ST_TI_IS_MMV_DEAC
    and   TI_IS_MMV > TI_IS_MMV_MAX
Then      TI_IS_MMV_MAX = TI_IS_MMV
    If          T_AST > C_T_AST_N_DIF_COR_MAX
    Then
          If    N_DIF_COR > N_DIF_COR_MAX
          Then  N_DIF_COR_MAX = N_DIF_COR
          Endif
    Endif
Endif
```

- *DIST_TI_IS calculation for IP_N_DIF_COR_ADD axis input*

DIST_TI_IS = DIST / 655.37

Example of TI_IS / TI_IS_MMV shapes :



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TI_IS_1	8*4	0...FFH	-0.5...0.496	1 / 256	-
LDPM_N_DIF_COR_2_IP_TI_IS	8	0000...FFFFH	-32768...+32767	1	Rpm
LDPM_TQ_ADD_CH_IP_TI_IS	4	0000...FFFFH	-1024...1023.97	0.03125	Nm
Idle speed correction for nominal injection time for low coolant temperature.					
IP_TI_IS_2	8*4	0...FFH	-0.5...0.496	1 / 256	-
LDPM_N_DIF_COR_2_IP_TI_IS	8	0000...FFFFH	-32768...+32767	1	Rpm
LDPM_TQ_ADD_CH_IP_TI_IS	4	0000...FFFFH	-1024...1023.97	0.03125	Nm
Idle speed correction for nominal injection time for high coolant temperature.					
C_TCO_ST_TI_IS	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature at Start threshold for TI_IS correction					
C_TI_IS_POS	1	80...7FH	-0.5...0.496	1 / 256	-
TI_IS correction threshold to set immediately filtered value TI_IS_MMV to current TI_IS					
C_TI_IS_MMV_MIN	1	0...7FH	0...0.496	1 / 256	-
Dead band for TI_IS_MM correction					
C_TI_IS_CRLC_POS	1	0...FFH	0...0.996	0.0039	-
TI_IS filtering constant when TI_IS_MMV > 0					
C_TI_IS_CRLC_NEG	1	0...FFH	0...0.996	0.0039	-
TI_IS filtering constant when TI_IS_MMV < 0					
C_T_AST_N_DIF_COR_MAX	1	0...FFFFH	0...6553.5	0.1	sec
delay before N_DIF_COR_MAX calculation					
C_PV_AV_TI_IS_DEAC	1	0...FFH	0...99.6	0.3906	%
PV_AV value for deactivation of TI_IS_MMV_MAX calculation					
C_CTR_TI_IS_MMV_MAX_FDOUT	1	0 ... FFFF	0... 65535	1	
timer value for deactivation of TI_IS_MMV_MAX calculation due to gear change					
C_TCO_ST_TI_IS_MMV_DEAC	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature at Start threshold for TI_IS_MMV_MAX learning					
C_VS_MIN_TI_IS	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for TI_IS calculation					
IP_N_DIF_COR_ADD	8*4	0000...FFFFH	-32768...+32767	1	Rpm
LDP_DIST_TI_IS_IP_N_DIF_COR_ADD	8	0000...FFFFH	0... 655350	10	km
LDPM_TQ_ADD_CH_IP_TI_IS	4	0000...FFFFH	-1024...1023.97	0.03125	Nm
Correction on N_DIF_COR for TI_IS calculation					
C_TCO_MIN_TI_IS	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant for TI_IS_MMV calculation					
C_CTR_TI_IS_MMV_FDOUT	1	0 ... FFFF	0... 65535	1	
timer value for deactivation of TI_IS_MMV calculation due to power steering activation					
C_PSP_THD_TI_IS_MMV	1	0..FFH	0...25.5	0.1	MPa
Power steering pressure threshold to deactivate TI_IS_MMV calculation					
C_TCO_MIN_TI_IS_ACT	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant for TI_IS activation					

7.22 Wall Film Correction (TI_ADD_WF)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_ADD_WF	O/V	8000...7FFFH	-694.5...694.48	0.02119	mg/stk
Total fuel amount for wall film compensation					
TI_ADD_WF	V	8000...7FFFH	-131,04...131,1	0,004	ms
Total injection time correction for wall film compensation					
LV_TIB_DIF_POS	V	0...1H	0...1	1	-
Bit to detect load transient direction (positive load transient if LV_TIB_DIF_POS = 1)					
SEG_CTR_WF	V	0...FFFFH	0...65535	1	-
Wall film segment counter running down after engine start					
FAC_COR_TCYL_FAST_WF	V	0...FFFFH	0...31,999	0,00049	-
Intake air temperature correction factor for fast wallfilm path					
FAC_COR_TCYL_SLOW_WF	V	0...FFFFH	0...31,999	0,00049	-
Intake air temperature correction factor for slow wallfilm path					
TI_COR_FAST_WF	V	8000...7FFFH	-1048,57...1048,54	0,032	ms
Wallfilm mass for fast path after temperature correction					
TI_COR_SLOW_WF	V	8000...7FFFH	-1048,57...1048,54	0,032	ms
Wallfilm mass for fast slow after temperature correction					
TI_DIF_FAST_PU_WF	V	8000...7FFFH	-1048,57...1048,54	0,032	ms
Raw wallfilm mass for fast path after temperature dependent threshold					
TI_DIF_SLOW_PU_WF	V	8000...7FFFH	-1048,57...1048,54	0,032	ms
Raw wallfilm mass for slow path after temperature dependent threshold					
TI_DIF_FAST_WF_POST_THD	V	8000...7FFFH	-131,04...131,1	0,004	ms
Raw wallfilm mass for fast path after temperature dependent threshold					
TI_DIF_SLOW_WF_POST_THD	V	8000...7FFFH	-131,04...131,1	0,004	ms
Raw wallfilm mass for slow path after temperature dependent threshold					
TI_FAST_WF	-	80000000... 7FFFFFFFH	-131,07...131,06	6,1034E-8	ms
Complete wallfilm correction for fast path before injection time dependent threshold					
TI_FAST_WF_POST_THD	V	8000...7FFFH	-131,04...131,1	0,004	ms
Injection time correction due to fast compensation					
TI_SLOW_WF	-	80000000... 7FFFFFFFH	-131,07...131,06	6,1034E-8	ms
Complete wallfilm correction for slow path before injection time dependent threshold					
TI_SLOW_WF_POST_THD	V	8000...7FFFH	-131,04...131,1	0,004	ms
Injection time correction due to slow compensation					
TIB	V	0...FFFFH	0...262.14	0,004	ms
Basic injection time					
FAC_TI_WF_COR_FQ	V	0...FFFFH	0...7.9999	0,000122	-
Wall film correction factor from fuel quality adaptation					

Input data:

MFF_BAS	C_FAC_MFF_TI_STND_k	FAC_FQ_ST_AD	
EFF_VOL_TEMP_COR_MMV	LV_ES	LV_IS	LV_PU
LV_PUC	LV_ST_END	N_32	TCO
TCO_ST	SEG_NR	TI_ADD_WF_VO	TI_IS_MMV_MAX
LV_FQ_ST_AD_WF			

FUNCTION DESCRIPTION:

General information:

The load detection provides an air-mass flow signal MAF (i.e. the airflow into the cylinder), that is valid for all engine operating states (and also during transient load changes). This value, which is updated segment synchronous, is used to calculate the basic injection time TIB.

To compensate the build-up / removal of the fuel film in the intake ducts, the instationary wall film compensation function provides a (additive) positive / negative injection time TI_ADD_WF.

The transient function is activated for the first time during the transition from the engine operating state ST to PL / IS and applies in all operating states (except ES and ST).

A 2nd order model, with a fast and a (nearly identical) slow part is used to compensate the quantity of the wall-applied fuel film (in the following referred to as "wall-film") stored in the intake ducts.

To obtain the load-dependend wall-film compensation, the wall-film differential quantity between two sampling steps is determined both, for the slow and the fast path on the basis of the injection period difference (i.e. load difference). The fast and slow part of the wall-film differential quantity is determined by two first-order time-delay elements. The total compensation quantity *TI_ADD_WF* is the sum of the fast and the slow quantity.

Calculation recurrence: segment synchronous

Signal flow diagram:

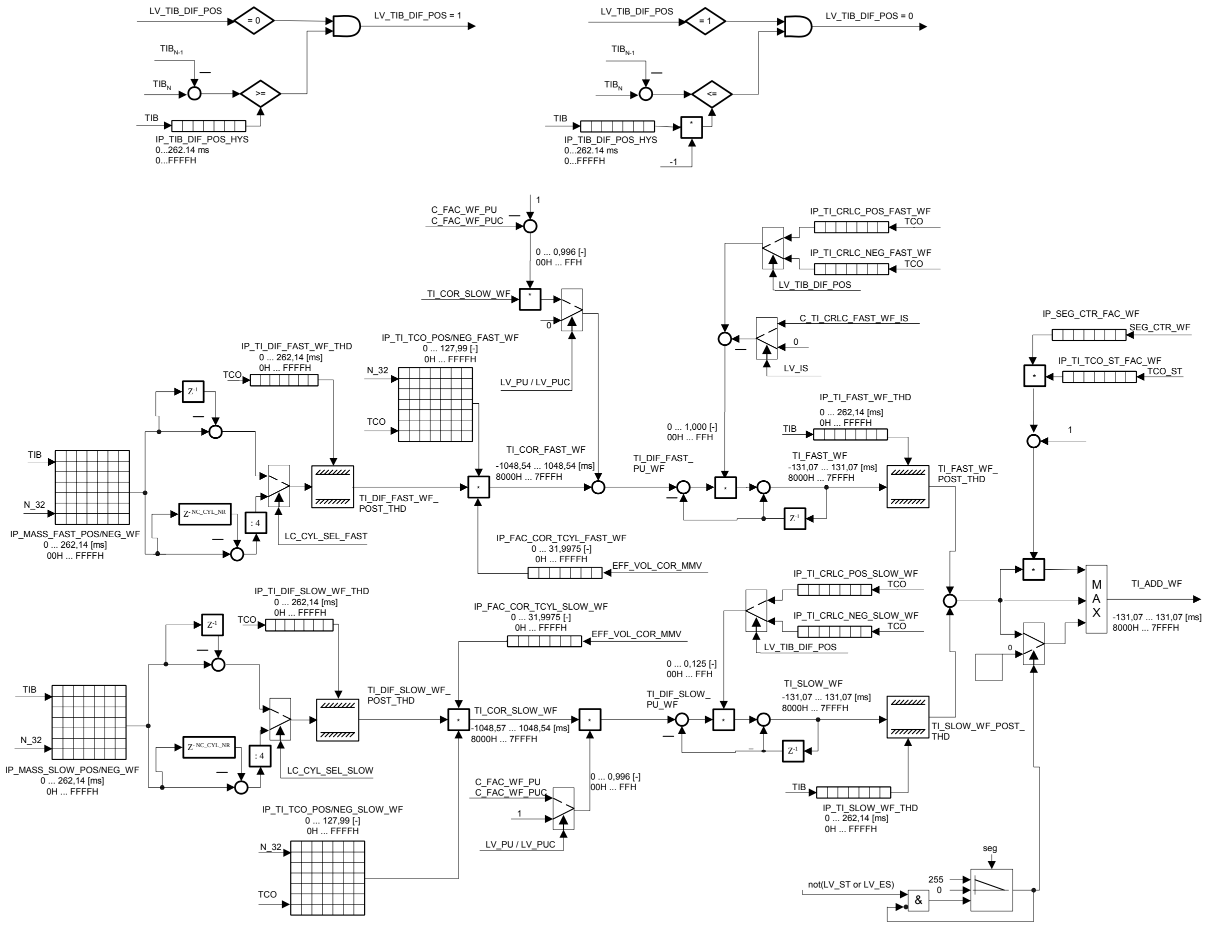


Figure 1: **Signal flow diagram for wall-film compensation function**

Application conditions:

Initialization: ***if*** LV_ST_END = 0 → 1 ***or*** LV_ES = 0 → 1 ***then***

TIB_n = MFF_BAS * C_FAC_MFF_TI_STND_1

TIB_{n-1} *to* $TIB_{n-NC\_CYL\_NR}$ = TIB_n (array)

TI_FAST_WF = 0;

TI_SLOW_WF = 0;

TI_ADD_WF = 0;

CYC_WF = C_SEG_CTR_INH_NEG_WF

TI_COR_SLOW_WF = 0

TI_COR_FAST_WF = 0

TI_DIF_FAST_PU_WF = 0

TI_DIF_SLOW_PU_WF = 0

TI_FAST_WF_POST_THD = 0

TI_SLOW_WF_POST_THD = 0

Endif

At reset: CYC_WF = C_SEG_CTR_INH_NEG_WF

TI_DIF_FAST_WF_THD = IP_TI_DIF_FAST_WF_THD

TI_DIF_SLOW_WF_THD = IP_TI_DIF_SLOW_WF_THD

7.22.1 Initialization of the counter

Initialization: *if* LV_ST_END = 0 → 1 : SEG_CTR_WF = C_SEG_CTR_WF_INI

A segment counter SEG_CTR_WF is initialized with C_SEG_CTR_WF_INI at the transition from start to any other engine operating state and then decremented (every TDC) to zero.

7.22.2 Calculation of the wallfilm compensation (must be done every TDC)

Activation: *if* LV_ST_END = 1

7.22.2.1 Estimation of the direction of the load transient (LV_TIB_DIF_POS)

Based on the basic injection time TIB an estimation of the direction of load transient is done. If the injection time is increasing, the load is increasing (positive load transient, LV_TIB_DIF_POS = 1), otherwise a negative load is added (LV_TIB_DIF_POS = 0). To prevent jittering, a hysteresis IP_TIB_DIF_POS_HYS must be exceeded.

<u>Conversion TI to MFF:</u>

TIB = MFF_BAS * C_FAC_MFF_TI_STND_1

Only for the first calculation run do: TIB_{n-1} to $TIB_{n-NC\_CYL\_NR}$ = TIB_n (array)

if (LV_TIB_DIF_POS = 0) **and**
($TIB_n - TIB_{n-1}$) ≥ IP_TIB_DIF_POS_HYS

then

LV_TIB_DIF_POS = 1

endif

if (LV_TIB_DIF_POS = 1) and

($TIB_n - TIB_{n-1}$) ≤ - IP_TIB_DIF_POS_HYS

then

LV_TIB_DIF_POS = 0

endif

Note: For the first calculation cycle the variable LV_TIB_DIF_POS = 0.

(*n* denotes the actual and *n-1* the previous segment)

7.22.2.2 Determination of the wall-film differential quantity TI_DIF_FAST/SLOW_WF_POST_THD

The amount of the wall film stored in the intake duct *MASS_FAST/SLOW_WF* (which is proportional to the injection timing) depends on the engine speed, the load and the direction of the transient (LV_TIB_DIF_POS).

With the logical variables *LC_CYL_SEL_FAST/SLOW* it is possible to determine the change in wall-film amount cylinderselective (triggered by the segment number SEG_NR). <u>The maximum allowed number of NC_CYL_NR is in the current software limited to NC_CYL_NR = 8. This has to be changed if more required.</u>

The change of the wall-film due to the load transient is

if (LC_CYL_SEL_FAST = 1)
then
if (LV_TIB_DIF_POS = 1)
then
TI_DIF_FAST_WF =

$$\frac{IP\_MASS\_FAST\_POS\_WF(N\_32, TIB_n) - IP\_MASS\_FAST\_POS\_WF(N\_32, TIB_{n - NC\_CYL\_NR})}{NC\_CYL\_NR}$$

else

TI_DIF_FAST_WF =

$$\frac{IP\_MASS\_FAST\_NEG\_WF(N\_32, TIB_n) - IP\_MASS\_FAST\_NEG\_WF(N\_32, TIB_{n - NC\_CYL\_NR})}{NC\_CYL\_NR}$$

endif

else

if (LV_TIB_DIF_POS = 1)
then
TI_DIF_FAST_WF =

IP_MASS_FAST_POS_WF(N_32, TIB_n) – IP_MASS_FAST_POS_WF(N_32, TIB_{n-1})

else

TI_DIF_FAST_WF =

IP_MASS_FAST_NEG_WF(N_32, TIB_n) – IP_MASS_FAST_NEG_WF(N_32, TIB_{n-1})

endif

endif

for the fast path and

if (LC_CYL_SEL_SLOW = 1)
then
if (LV_TIB_DIF_POS = 1)
then
TI_DIF_SLOW_WF =

$$\frac{IP\_MASS\_SLOW\_POS\_WF(N\_32, TIB_n) - IP\_MASS\_SLOW\_POS\_WF(N\_32, TIB_{n - NC\_CYL\_NR})}{NC\_CYL\_NR}$$

else
TI_DIF_SLOW_WF =

$$\frac{IP\_MASS\_SLOW\_NEG\_WF(N\_32, TIB_n) - IP\_MASS\_SLOW\_NEG\_WF(N\_32, TIB_{n - NC\_CYL\_NR})}{NC\_CYL\_NR}$$

endif

else
if (LV_TIB_DIF_POS = 1)
then

TI_DIF_SLOW_WF =
IP_MASS_SLOW_POS_WF(N_32, TIB_n) - IP_MASS_SLOW_POS_WF(N_32, $TIB_{n - 1}$)

else

TI_DIF_SLOW_WF =
IP_MASS_SLOW_NEG_WF(N_32, TIB_n) - IP_MASS_SLOW_NEG_WF(N_32, TIB_{n-1})

endif

endif

for the slow path. This change in the amount of fuel stored in the wall film has to be compensated by the wall-film function.

If the difference in the wall-applied fuel mass is below a temperature dependent threshold, or it is negative and the number of segments after start is still below C_SEG_CTR_INH_NEG_WF, the injection time correction due to the wallfilm is prohibited.

TI_DIF_FAST_WF_THD = IP_TI_DIF_FAST_WF_THD , updated every second

```
if    abs( TI_DIF_FAST_WF ) < TI_DIF_FAST_WF_THD        or
      [        TI_DIF_FAST_WF < 0                   and
               C_SEG_CTR_INH_NEG_WF segments after start are not elapsed
]
then
               TI_DIF_FAST_WF_POST_THD = 0
else
               TI_DIF_FAST_WF_POST_THD = TI_DIF_FAST_WF
Endif
```

TI_DIF_SLOW_WF_THD = IP_TI_DIF_SLOW_WF_THD , updated every second

```
if  abs( TI_DIF_SLOW_WF ) < TI_DIF_SLOW_WF_THD          or
      [        TI_DIF_SLOW_WF < 0                   and
               C_SEG_CTR_INH_NEG_WF segments after start are not elapsed
]
then
               TI_DIF_SLOW_WF_POST_THD = 0
else
               TI_DIF_SLOW_WF_POST_THD = TI_DIF_SLOW_WF
endif
```

7.22.2.3 Temperature and engine speed correction of the wall-film compensation quantity

Depending on engine speed and coolant temperature the fuel quantity for wall-film compensation has to be corrected by a multiplicative factor. This factor is depending on the direction of the load transient:

POS applies if: $TIB_n - TIB_{n-1} \geq 0$
NEG applies if: $TIB_n - TIB_{n-1} < 0$

```
if    LV_TIB_DIF_POS = 1
then
      if    LV_IS = 1     and
            VS = 0        and
            abs (TIB[n] – TIB[n-1]) < IP_TI_DIF_WF_IS_POS
      then
            TI_COR_FAST_WF = TI_DIF_FAST_WF_POST_THD *
                             IP_TI_TCO_POS_FAST_WF_IS
            TI_COR_SLOW_WF = TI_DIF_SLOW_WF_POST_THD *
                             IP_TI_TCO_POS_SLOW_WF_IS
      else
            TI_COR_FAST_WF = TI_DIF_FAST_WF_POST_THD *
                             IP_TI_TCO_POS_FAST_WF
            TI_COR_SLOW_WF = TI_DIF_SLOW_WF_POST_THD *
                             IP_TI_TCO_POS_SLOW_WF
else
      if    LV_IS = 1     and
            VS = 0        and
            abs (TIB[n] – TIB[n-1]) < IP_TI_DIF_WF_IS_NEG
      then
                  TI_COR_FAST_WF = TI_DIF_FAST_WF_POST_THD *
                                   IP_TI_TCO_NEG_FAST_WF_IS
                  TI_COR_SLOW_WF = TI_DIF_SLOW_WF_POST_THD *
                                   IP_TI_TCO_NEG_SLOW_WF_IS
            else
                  TI_COR_FAST_WF = TI_DIF_FAST_WF_POST_THD  *
                                   IP_TI_TCO_NEG_FAST_WF
                  TI_COR_SLOW_WF = TI_DIF_SLOW_WF_POST_THD *
                                   IP_TI_TCO_NEG_SLOW_WF
endif
```

The compensation of effects due to intake air temperature at the inlet valves is done seperately for the fast and slow path

```
FAC_COR_TCYL_FAST_WF    =    IP_FAC_COR_TCYL_FAST_WF
FAC_COR_TCYL_SLOW_WF    =    IP_FAC_COR_TCYL_SLOW_WF
TI_COR_FAST_WF = TI_COR_FAST_WF * FAC_COR_TCYL_FAST_WF
TI_COR_SLOW_WF = TI_COR_SLOW_WF * FAC_COR_TCYL_SLOW_WF
```

After the calculation of TI_COR_FAST/SLOW_WF the software divides the result by 8. If the result is rounded, not the original, but the absolute value of TI_COR_FAST/SLOW_WF shall be rounded. That means, negative values between -1 and -7 would be (divided by 8 and) rounded to 0.

Correction of wall-applied fuel film during PU or PUC
During PU or PUC, a transfer from a slow wall-film quantity into a fast wall-film quantity can be obtained as a function of *C_FAC_WF_PU* or *C_FAC_WF_PUC*.

```
if    LV_PU
```

then

TI_DIF_FAST_PU_WF = TI_COR_FAST_WF +
(1- C_FAC_WF_PU) * TI_COR_SLOW_WF

*TI_DIF_SLOW_PU_WF = TI_COR_SLOW_WF * C_FAC_WF_PU*

elseif LV_PUC
then

TI_DIF_FAST_PU_WF = TI_COR_FAST_WF +
(1- C_FAC_WF_PUC) * TI_COR_SLOW_WF

TI_DIF_SLOW_PU_WF = TI_COR_SLOW_WF * C_FAC_WF_PUC

else

TI_DIF_FAST_PU_WF = TI_COR_FAST_WF

TI_DIF_SLOW_PU_WF = TI_COR_SLOW_WF

endif

7.22.2.4 Calculation of the fast and slow compensation quantity

The fuel quantity *TI_ADD_WF* needed to compensate the wall-applied fuel film consists of the following components: TI_FAST_WF for a quick compensation, *TI_SLOW_WF* for a slow compensation. At idle, the quick reduction constant IP_TI_CRLC_POS/NEG_FAST_WF is reduced by the adjustable value C_TI_CRLC_FAST_WF_IS. The wallfilm compensation quantity is now calculated as follows:

if LV_TIB_DIF_POS = 1 (positive load change)
then

TI_SLOW_WFn = TI_DIF_SLOW_PU_WFn * IP_TI_CRLC_POS_SLOW_WF +

(1 - IP_TI_CRLC_POS_SLOW_WF) * TI_SLOW_WFn-1

if (LV_IS = 0) (not in idle)
then

TI_FAST_WFn = TI_DIF_FAST_PU_WFn * IP_TI_CRLC_POS_FAST_WF +

(1 - IP_TI_CRLC_POS_FAST_WF) * TI_FAST_WFn-1

else (in idle: LV_IS = 1)

TI_FAST_WFn = TI_DIF_FAST_PU_WFn *

(IP_TI_CRLC_POS_FAST_WF - C_TI_CRLC_FAST_WF_IS) +

(1 - (IP_TI_CRLC_POS_FAST_WF - C_TI_CRLC_FAST_WF_IS)) *
TI_FAST_WFn-1

endif

else (negative load *LV_TIB_DIF_POS = 0*)

TI_SLOW_WFn = TI_DIF_SLOW_PU_WFn * IP_TI_CRLC_NEG_SLOW_WF +

(1 - IP_TI_CRLC_NEG_SLOW_WF) * TI_SLOW_WFn-1

if *(LV_IS = 0)* (not in idle)
then

TI_FAST_WFn = TI_DIF_FAST_PU_WFn *
IP_TI_CRLC_NEG_FAST_WF +

(1 - IP_TI_CRLC_NEG_FAST_WF) * TI_FAST_WFn-1

else (in idle: *LV_IS = 1)*

TI_FAST_WFn = TI_DIF_FAST_PU_WFn *

(IP_TI_CRLC_NEG_FAST_WF - C_TI_CRLC_FAST_WF_IS) +

(1 - (IP_TI_CRLC_NEG_FAST_WF - C_TI_CRLC_FAST_WF_IS)) *

TI_FAST_WFn-1

endif

endif

7.22.2.5 Thresholds for TI_FAST_WF and TI_SLOW_WF

To prevent the wall-film function from jittering, under steady state engine operating wall-film corrections are forbidden. If the absolute value of *TI_FAST/SLOW_WF* is less than the calibratable threshold *IP_TI_FAST/SLOW_WF_THD*, no correction of the injection timing will be performed. The thresholds are functions of the basic injection time TIB.

Setting the calibratable constant LC_DEAC_FAST_WF and / or LC_DEAC_SLOW_WF to one will deactivate the fast and / or slow path of the wallfilm compensation.

if abs(TI_FAST_WF) < IP_TI_FAST_WF_THD or

LC_DEAC_FAST_WF = 1

then

TI_FAST_WF_POST_THD = 0

else

TI_FAST_WF_POST_THD = TI_FAST_WF

endif

if abs(TI_SLOW_WF) < IP_TI_SLOW_WF_THD **or**

LC_DEAC_SLOW_WF = 1

then

TI_SLOW_WF_POST_THD = 0

else

TI_SLOW_WF_POST_THD = TI_SLOW_WF

Endif

7.22.2.6 Calculation of the total compensation quantity

The total transition compensation quantity *TI_ADD_WF* is calculated as follows:

TI_ADD_WF = TI_FAST_WF_POST_THD + TI_SLOW_WF_POST_THD

_+ TI_ADD_WF_VO

7.22.2.7 Calculation of wall film correction facor from fuel quality adaptation

FAC_TI_WF_COR_FQ = 1 + ((FAC_FQ_ST_AD * C_FAC_WF_FQ_ST_AD +

TI_IS_MMV_MAX – 1) * IP_FAC_FQ_WF)

7.22.2.8 Correction of TI_ADD_WF for the first tip in

A segment counter SEG_CTR_WF is initialized with C_SEG_CTR_WF_INI at the transition from start to any other engine operating state and then decremented to zero (see chapter 7.22.1).
Dependent on the actual value of SEG_CTR_WF and the coolant temperature at engine start (TCO_ST) the positive wall film compensations of TI_ADD_WF can be increased by a factor.

if TI_ADD_WF > 0 **and**
SEG_CTR_WF > 0

then

TI_ADD_WF = TI_ADD_WF *
(1 + IP_TI_TCO_ST_FAC_WF * IP_SEG_CTR_FAC_WF)

Endif

<u>Conversion of TI to MFF with fuel quality adaptation factor applied:</u>

if LV_FQ_ST_AD_WF = 1

then

if TI_ADD_WF > 0

then MFF_ADD_WF = (TI_ADD_WF / C_FAC_MFF_TI_STND_1)
* FAC_TI_WF_COR_FQ **else**

if LC_WF_COR_NEG_FQ_ENA = 1

then MFF_ADD_WF = TI_ADD_WF / C_FAC_MFF_TI_STND_1
/ FAC_TI_WF_COR_FQ

else MFF_ADD_WF = TI_ADD_WF / C_FAC_MFF_TI_STND_1

endif

endif

elseMFF_ADD_WF = TI_ADD_WF / C_FAC_MFF_TI_STND_1

7.22.2.9 endif**Correction of the wall-film compensation during post-start**

During engine start, the intake manifold pressure is initialized with the adapted ambient pressure. Subsequently the engine evacuates the manifold. Therefore a continously decreasing load signal is obtained, so that negative wall-film compensation quantities are calculated although a wall-applied fuel film is built-up. Therefore negative wall-film quantities are prohibited for C_SEG_CTR_INH_NEG_WF cycles: during the first C_SEG_CTR_INH_NEG_WF cycles (segments) after transition from engine operating state ST to PL or IS, *TI_ADD_WF* is limited to 0.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_WF_PU	1	0...FFH	0...0.996	0.0039	-
Transposition factor for PU					
C_FAC_WF_PUC	1	0...FFH	0...0.996	0.0039	-
Transposition factor for PUC					
C_SEG_CTR_INH_NEG_WF	1	0...FFFFH	0...65535	1	-
Number of segments after start to inhibit negative wallfilm amount (set to 0)					
C_SEG_CTR_WF_INI	1	0...FFFFH	0...65535	1	-
Initial value for number of segments after start with increased pos. wall film compensation					
C_TI_CRLC_FAST_WF_IS	1	0...FFH	0...0.99609	0.0039	-
Fast correlation constant for idle speed					
IP_FAC_COR_TCYL_FAST_WF	8	0...FFFFH	0...31,9975	0,00049	-
LDPM_EFF_VOL_TEMP_COR_MMV_1_WF	8	0...FFFFH	0...1.999969	3.052E-5	-
Intake air temperature correction factor for fast wallfilm path					
IP_FAC_COR_TCYL_SLOW_WF	8	0...FFFFH	0...31,9975	0,00049	-
LDPM_EFF_VOL_TEMP_COR_MMV_1_WF	8	0...FFFFH	0...1.999969	3.052E-5	-
Intake air temperature correction factor for slow wallfilm path					
IP_MASS_FAST_NEG_WF	8*8	0...FFFFH	0...262.14	0.004	ms
LDPM_N_32_WF	8	0...FFH	0...8160	32	rpm
LDPM_TIB_1_WF	8	0...FFFFH	0...262.14	0.004	ms
Wall film mass stored in the intake (fast path, load decrease)					
IP_MASS_FAST_POS_WF	8*8	0...FFFFH	0...262.14	0.004	ms
LDPM_N_32_WF	8	0...FFH	0...8160	32	rpm
LDPM_TIB_1_WF	8	0...FFFFH	0...262.14	0.004	ms
Wall film mass stored in the intake (fast path, load increase)					
IP_MASS_SLOW_NEG_WF	8*8	0...FFFFH	0...262.14	0.004	ms
LDPM_N_32_WF	8	0...FFH	0...8160	32	rpm
LDPM_TIB_1_WF	8	0...FFFFH	0...262.14	0.004	ms
Wall film mass stored in the intake (slow path, load decrease)					
IP_MASS_SLOW_POS_WF	8*8	0...FFFFH	0...262.14	0.004	ms
LDPM_N_32_WF	8	0...FFH	0...8160	32	rpm
LDPM_TIB_1_WF	8	0...FFFFH	0...262.14	0.004	ms
Wall film mass stored in the intake (slow path, load increase)					
IP_SEG_CTR_FAC_WF	6*5	0...FFH	0...0.99609	0.0039	-
LDP_SEG_CTR_WF_IP_SEG_CTR	6	0...FFFFH	0...65535	1	seg.
LDP_TCO_ST_IP_SEG_CTR	5	00...FEH	-48...142.5	0.75	°C
Number of segments dependend factor for first tip in					
IP_TI_CRLC_NEG_FAST_WF	8	0...FFH	0...0.99609	0.0039	-
LDPM_TCO_1_WF	8	0...FEH	-48...142.5	0.75	°C
Fast negative correlation constant					
IP_TI_CRLC_POS_FAST_WF	8	0...FFH	0...0.99609	0.0039	-
LDPM_TCO_1_WF	8	0...FEH	-48...142.5	0.75	°C
Fast positive correlation constant					
IP_TI_CRLC_NEG_SLOW_WF	8	0...FFH	0...0.498	0.0019	-
LDPM_TCO_1_WF	8	0...FEH	-48...142.5	0.75	°C
Slow negative correlation constant					
IP_TI_CRLC_POS_SLOW_WF	8	0...FFH	0...0.498	0.0019	-
LDPM_TCO_1_WF	8	0...FEH	-48...142.5	0.75	°C
Slow positive correlation constant					
IP_TI_DIF_FAST_WF_THD	4	0...FFFFH	0...262.14	0.004	ms
LDPM_TCO_2_WF	4	0...FEH	-48...142.5	0.75	°C
LDPM_TIB_3_WF	2	0...FFFFH	0...262.14	0.004	ms
Threshold for fast wall film correction					
IP_TI_DIF_SLOW_WF_THD	4	0...FFFFH	0...262.14	0.004	ms
LDPM_TCO_2_WF	4	0...FEH	-48...142.5	0.75	°C
LDPM_TIB_3_WF	2	0...FFFFH	0...262.14	0.004	ms
Threshold for slow wall film correction					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TI_FAST_WF_THD	4	0000...FFFFH	0...262.14	0.004	ms
LDPM_TIB_2_WF	4	00...FFFFH	0...262.14	0.004	ms
Threshold for fast wall film correction					
IP_TI_SLOW_WF_THD	4	0000...FFFFH	0...262.14	0.004	ms
LDPM_TIB_2_WF	4	00...FFFFH	0...262.14	0.004	ms
Threshold for slow wall film correction					
IP_TI_TCO_NEG_FAST_WF	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for negative load transients (fast path)					
IP_TI_TCO_NEG_SLOW_WF	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for negative load transients (slow path)					
IP_TI_TCO_POS_FAST_WF	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for positive load transients (fast path)					
IP_TI_TCO_POS_SLOW_WF	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for positive load transients (slow path)					
IP_TI_TCO_ST_FAC_WF	4	00...FFH	0...0.99609	0.0039	-
LDPM_TCO_2_WF	4	00...FEH	-48...142.5	0.75	°C
TCO_ST depending factor for first tip in					
IP_TIB_DIF_POS_HYS	1*8	00...FFFFH	0...262.14	0.004	ms
LDPM_TIB_1_WF	8	00...FFFFH	0...262.14	0.004	ms
Hysteresis in TIB necessary for switching from positive to negative load transient					
LC_CYL_SEL_FAST	1	0...1H	0...1	1	-
Switch for cylinder selective wall film calculation (fast path)					
LC_CYL_SEL_SLOW	1	0...1H	0...1	1	-
Switch for cylinder selective wall film calculation (slow path)					
LC_DEAC_FAST_WF	1	0...1H	0...1	1	-
Switch for deactivation of fast path of wall film calculation					
LC_DEAC_SLOW_WF	1	0...1H	0...1	1	-
Switch for deactivation of slow path of wall film calculation					
IP_FAC_FQ_WF	8*8	0...FFH	0 ...4	0.015625	-
LDP_TCO_ST_IP_FAC_FQ_WF	8	0...FEH	-48 ...142.5	0.75	°C
LDP_TCO_IP_FAC_FQ_WF	8	0...FEH	-48 ...142.5	0.75	°C
Weighting of the fuel quality adaptation factor for wallfim compensation					
IP_TI_TCO_NEG_FAST_WF_IS	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for negative load transients (fast path) – used in idle and VS = 0					
IP_TI_TCO_NEG_SLOW_WF_IS	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for negative load transients (slow path) – used in idle and VS = 0					
IP_TI_TCO_POS_FAST_WF_IS	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for positive load transients (fast path) – used in idle and VS = 0					
IP_TI_TCO_POS_SLOW_WF_IS	8*8	00...FFFFH	0...127.99	0.002	-
LDPM_N_32_WF	8	00...FFH	0...8160	32	rpm
LDPM_TCO_1_WF	8	00...FEH	-48...142.5	0.75	°C
Temperature correction factor for positive load transients (slow path) – used in idle and VS = 0					
IP_TI_DIF_WF_IS_POS	8	0...FFFFH	0...262.14	0.004	ms
LDPM_TCO_3_WF	8	00...FEH	-48...142.5	0.75	°C

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
Treshold to activate positive load transients in idle					
IP_TI_DIF_WF_IS_NEG	8	0...FFFFH	0...262.14	0.004	ms
LDPM_TCO_3_WF	8	00...FEH	-48...142.5	0.75	°C
Treshold to activate negative load transients in idle					
C_FAC_WF_FQ_ST_AD	1	0...FFH	0...1.992	0.0078	-
Scaling factor between FAC_FQ_ST_AD and TI_IS_MMV_MAX					
LC_WF_COR_NEG_FQ_ENA	1	0...1H	0...1	1	-
Switch to activate FQA for negative wall-film intervention					

7.23 Compensation of VO influence on wall film

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_ADD_WF_VO	V/O	8000 ... 7FFFH	-131.04 ... 131.1	0.004	ms
Injection time correction due to compensation of VO influence on wall film, after threshold monitoring					
LV_POS_VO_CHG	V	0 ... 1H	0 ... 1	1	-
Flag that indicates if the VO change is positive or negative					
TI_DIF_WF_VO_POST_THD	V	8000 ... 7FFFH	-131.04 ... 131.1	0.004	ms
Calculated wall film amount to be compensated, after threshold monitoring					
FAC_TCO_ST_COR_WF_POS_VO	V	0 ... FFFFH	0 ... 128	0.00195531	-
Temperature correction factor, positive VO change					
FAC_TCO_ST_COR_WF_NEG_VO	V	0 ... FFFFH	0 ... 128	0.00195531	-
Temperature correction factor, negative VO change					
FAC_WF_VO_FDOUT	V	0 ... 80 H	0 ... 1	0.0078125	-
Fade out factor					
TI_DIF_WF_VO_COR	V	8000 ... 7FFFH	-131.04 ... 131.1	0.004	ms
Calculated wall film amount to be compensated, temperature corrected					
TIB_INT	V	0 ... FFFFH	0 ... 134215.68	2.048	ms
Integarl of TIB					
TIB_INT_TMP	-	0 ... FFFF FFFFH	0 ... 17179869.18	0.004	ms
Integarl of TIB, temporary value					
FAC_TIB_COR	V	0 ... FFFFH	0 ... 2	3.0518 e-5	-
Correction factor for TIB					
TI_DIF_WF_VO	V	8000 ... 7FFFH	-131.04 ... 131.1	0.004	ms
Calculated wall film amount to be compensated					
TI_WF_VO	-	8000 0000 ... 7FFF FFFF	-131.04 ... 131.1	6.103422 e-8	ms
Injection time correction due to compensation of VO influence on wall film					
T_TPS_WF_VO_DEAC	V	0 ... FFFF	0... 655.35	0.01	s
Timer to deactivate the compensation of VO influence on wall film after first tip-in					

Input data:

LV_ST_END	VO	LV_PUC	TIB
TCO_ST	TCO	LV_IS	N_32
NC_CYL_NR	LV_TI_IS_MMV_MAX_DEAC		

FUNCTION DESCRIPTION:

General information:

The backflow of residual gas from the cylinder to the intake manifold depends on the actual valve overlap VO. With increasing VO the backflow increases and evaporates wall film from the intake valve disc and from the wall of the intake manifold until a new wall film equilibrium condition results. Especially at cold engine condition this additional fuel amount leads to a rich mixture within the cylinder. With decrasing VO the backflow decreases and the wall film is build up. This leads to a lean mixture within the cylinder.
In both cases the exhaust emissions downstream catalyst are negatively influenced.

This functionality compensates increase or lack of fuel amount within the cylinder.

Application conditions:

Initialisation:

at reset: FAC_WF_VO_FDOUT = 1

at LV_ST_END = 0 → 1:

FAC_TCO_ST_COR_WF_POS_VO = IP_FAC_TCO_ST_COR_WF_POS_VO

FAC_TCO_ST_COR_WF_NEG_VO = IP_FAC_TCO_ST_COR_WF_NEG_VO

(Temperature correction factors of the calculated wall film amount at start)

at LV_TI_IS_MMV_MAX_DEAC = 0 → 1: launch T_TPS_WF_VO_DEAC timer

Recurrence: segment synchronous

Deactivation: TCO > C_TCO_VO_COR_THD

Activation: LV_ST_END = 1 and (LC_WF_VO_COR_ENA_GLOBAL = 1 or LV_IS = 1)

Formula section:

7.23.1 Determination if the VO change is positive or negative

IF

$VO_n - VO_{(n-1)} >= C\_VO\_CHG\_HYS$

THEN

LV_POS_VO_CHG = 1

ELSEIF

$VO_n - VO_{(n-1)} \leq - C\_VO\_CHG\_HYS$

THEN

LV_POS_VO_CHG = 0

ENDIF

7.23.2 Determination of the wall film amount to be compensated

IF
LV_POS_VO_CHG = 1
THEN
TI_DIF_WF_VO =
(IP_TI_DIF_WF_POS_VO_CHG $_{(n\text{-}NC\_CYL\_NR)}$ - IP_TI_DIF_WF_POS_VO_CHG $_{n}$) / NC_CYL_NR
ELSE
TI_DIF_WF_VO =
(IP_TI_DIF_WF_NEG_VO_CHG $_{(n\text{-}NC\_CYL\_NR)}$ - IP_TI_DIF_WF_NEG_VO_CHG $_{n}$) / NC_CYL_NR
ENDIF

7.23.3 Threshold monitoring of the calculated wall film amount

IF
ABS(TI_DIF_WF_VO) < C_TI_DIF_WF_VO_THD
THEN
TI_DIF_WF_VO_POST_THD = 0
ELSE
TI_DIF_WF_VO_POST_THD = TI_DIF_WF_VO
ENDIF

7.23.4 Fade out factor for the correction

(1) IF
LV_PUC = 1
(1) THEN
TIB_INT_TMP $_{n}$ = TIB_INT_TMP $_{(n - 1)}$
(1) ELSE
TIB_INT_TMP $_{n}$ = TIB_INT_TMP $_{(n - 1)}$ + TIB
(1) ENDIF

TIB_INT = TIB_INT_TMP

(2) IF
LC_TIB_INT_FDOUT_ENA = 1
(2) THEN
If(3) (T_TPS_WF_VO_DEAC > C_T_TPS_WF_VO_DEAC)
Then(3)

FAC_WF_VO_FDOUT = 0

else(3)

FAC_WF_VO_FDOUT = IP_FAC_WF_VO_TIB_INT_FDOUT

endif(3

(2) ELSE

FAC_WF_VO_FDOUT = IP_FAC_WF_VO_TCO_FDOUT... updated every 10ms

(2) ENDIF

7.23.5 Calculation of the compensation quantity

FAC_TIB_COR = IP_ FAC_TIB_COR

IF

LV_POS_VO_CHG = 1

THEN

TI_DIF_WF_VO_COR =

TI_DIF_WF_VO_POST_THD * FAC_TCO_ST_COR_WF_POS_VO * FAC_WF_VO_FDOUT * FAC_TIB_COR

(FAC_TCO_ST_COR_WF_POS_VO ... Temperature correction factor of the calculated wall film amount at start)

$TI\_WF\_VO_{n} = TI\_DIF\_WF\_VO\_COR * C\_FIL\_WF\_POS\_VO + (1 - C\_FIL\_WF\_POS\_VO) * TI\_WF\_VO_{(n-1)}$

ELSE

TI_DIF_WF_VO_COR =

TI_DIF_WF_VO_POST_THD * FAC_TCO_ST_COR_WF_NEG_VO * FAC_WF_VO_FDOUT * FAC_TIB_COR

(FAC_TCO_ST_COR_WF_NEG_VO... Temperature correction factor of the calculated wall film amount at start)

$TI\_WF\_VO_{n} = TI\_DIF\_WF\_VO\_COR * C\_FIL\_WF\_NEG\_VO + (1 - C\_FIL\_WF\_NEG\_VO) * TI\_WF\_VO_{(n-1)}$

ENDIF

7.23.6 Threshold monitoring of the compensation quantity

IF

ABS(TI_WF_VO) < C_TI_WF_VO_THD

THEN

TI_ADD_WF_VO = 0

ELSE

TI_ADD_WF_VO = TI_WF_VO

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VO_CHG_HYS	1	0 ... 1C7H	170.625	0.375	°CRK
Hystereses for determination of the VO change direction					
C_TI_DIF_WF_VO_THD	1	0 ... FFFFH	0 ... 262.14	0.004	ms
Threshold for calculated wall film amount to be compensated					
C_FIL_WF_POS_VO	1	0 ... FFH	0 ... 0.99609375	0.00390625	-
Filter constant for calculation of the compensation quantity, positive VO change					
C_FIL_WF_NEG_VO	1	0 ... FFH	0 ... 0.99609375	0.00390625	-
Filter constant for calculation of the compensation quantity, negative VO change					
C_TI_WF_VO_THD	1	0 ... FFFFH	0 ... 262.14	0.004	ms
Treshold for compensation of VO influence					
LC_WF_VO_COR_ENA_GLOBAL	1	0 ... 1H	0 ... 1	1	-
Logical constant to enable correction at every engine operation state					
LC_TIB_INT_FDOUT_ENA	1	0 ... 1H	0 ... 1	1	-
Logical constant to enable different fade out strategies					
IP_FAC_TCO_ST_COR_WF_POS_VO	8	0 ... FFFFH	0 ... 128	0.00195531	-
LDP_TCO_ST_IP_FAC_TCO_ST_COR	8	0 ... FEH	-48 ... 142.5	0.75	°C
Temperature correction factor, positive VO change					
IP_FAC_TCO_ST_COR_WF_NEG_VO	8	0 ... FFFFH	0 ... 128	0.00195531	-
LDP_TCO_ST_IP_FAC_TCO_ST_COR	8	0 ... FEH	-48 ... 142.5	0.75	°C
Temperature correction factor, positive VO change					
IP_FAC_WF_VO_TIB_INT_FDOUT	3*8	0 ... 80 H	0 ... 1	0.0078125	-
LDP_TIB_INT_IP_FAC_WF_VO_TIB_INT	3	0 ... FFFFH	0 ... 134215.68	2.048	ms
LDPM_TCO_ST_IP_FAC_WF_VO_TIB_INT	8	0 ... FEH	-48 ... 142.5	0.75	°C
Fade out out factor, dependent from integral of TIB and TCO_ST					
IP_FAC_WF_VO_TCO_FDOUT	3*8	0 ... 80 H	0 ... 1	0.0078125	-
LDP_TCO_IP_FAC_WF_VO_TCO_FDOUT	3	0 ... FEH	-48 ... 142.5	0.75	°C
LDPM_TIB_IP_FAC_WF_VO_TIB_INT	8	0 ... FFFFH	0 ... 262.14	0.004	ms
Fade out out factor, TCO dependent					
IP_TI_DIF_WF_POS_VO_CHG	6*6	0 ... FFFFH	-131.04 ...131.1	0.004	ms
LDPM_VO_WF_VO_COR	6	0 ... 1C7H	170.625	0.375	°CRK
LDPM_N_32_ WF_VO_COR	6	0 ... FFH	0 ... 8160	32	rpm
Wall film amount to be compensated, positive VO change					
IP_TI_DIF_WF_NEG_VO_CHG	6*6	0 ... FFFFH	-131.04 ... 131.1	0.004	ms
LDPM_VO_WF_VO_COR	6	0 ... 1C7H	170.625	0.375	°CRK
LDPM_N_32_ WF_VO_COR	6	0 ... FFH	0 ... 8160	32	rpm
Wall film amount to be compensated, negative VO change					
IP_FAC_TIB_COR	8	0 ... FFFFH	0 ... 2	3.0518 e-5	-
LDP_TIB_IP_FAC_TIB_COR	8	0 ... FFFFH	0 ... 262.14	0.004	ms
Correction factor for TIB					
C_TCO_VO_COR_THD	1	0 ... FEH	-48 ... 142.5	0.75	°C
TCO Threshold to deactivate the compensation of VO influence on wall film					
C_T_TPS_WF_VO_DEAC	1	0 ... FFFFH	0 ... 655,35	0.01	s
Timer to deactivate the compensation of VO influence on wall film after first tip-in					

7.23.7 Combustion Chamber Wallfilm Compensation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_CMBC_WF	V/O	0...80H	0...1	0.0078125	-
Enrichment factor for compensation of the cold combustion chamber walls					
FAC_CMBC_WF_FDOUT	V	0...80H	0...1	0.0078125	-
Controlled reset factor of the injection time correction as a function of the energy flow through the engine					
MFF_SUM	V	0...FFFFH	0...44448	0.67823	mg
Total injected mass fuel flow since engine start					

Input data:

TCO	TIA_IM	MAF	LV_ST_END
TCO_ST	LV_PUC	MFF_SP_MV	

FUNCTION DESCRIPTION:

General Information:

To compensate the effects off wall film in the combustion chamber, the combustion chamber wallfilm compensation can be active parallel with post-start, warm-up, and catalyst heating. Thus an increase in the lean mixture at cold start can be avoided. The compensation factor FAC_CMBC_WF depends on integral of the mass fuel flow (MFF_SUM). The difference (TCO – TIA_IM) is used as an indicator for cold cylinder walls.

Application conditions:

Activation: LV_ST_END = 1

Deactivation: -

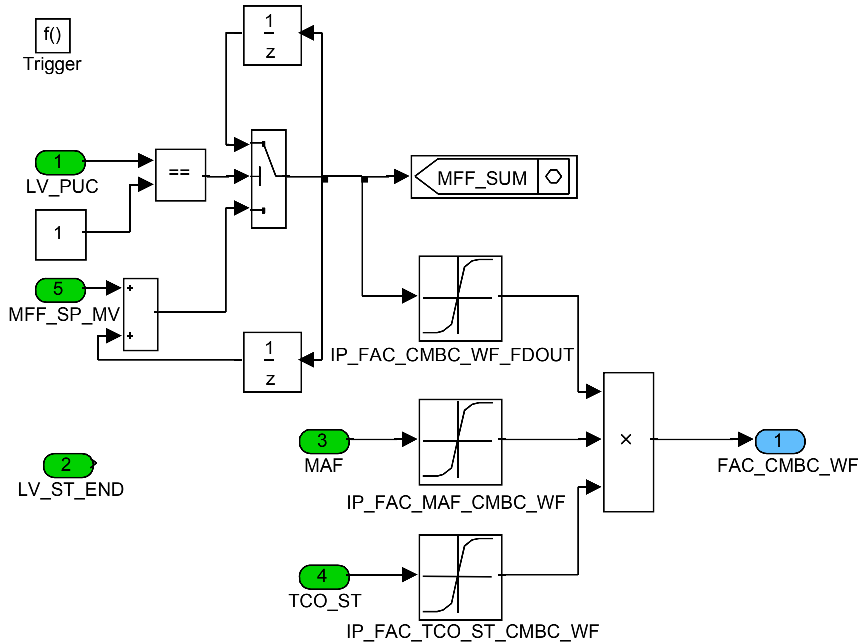
Initialisation: at reset: MFF_SUM = 0
FAC_CMBC_WF_FDOUT = 1

Recurrency: segment synchronous

Duration for the combustion chamber wallfilm compensation.

The value MFF_SP_MV is integrated and the function runs until a certain mass of fuel is reached.

Signal flow diagram:



Formula section:

(1) IF

(TCO – TIA_IM) < ID_TEMP_DIF_TCO_TIA

AND

FAC_CMBC_WF_FDOUT ≠ 0

(1) THEN

(2) IF

LV_PUC = 1

(2) THEN

$MFF\_SUM_{n+1} = MFF\_SUM_n$

(2) ELSE

$MFF\_SUM_{n+1} = MFF\_SUM_n + MFF\_SP\_MV$

(2) ENDIF

FAC_CMBC_WF_FDOUT = IP_ FAC_CMBC_WF_FDOUT

FAC_CMBC_WF = IP_FAC_TCO_ST_CMBC_WF *

* FAC_CMBC_WF_FDOUT * IP_FAC_MAF_CMBC_WF

(1) ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_TEMP_DIF_TCO_TIA	3	0...FEH	-48...142.5	0.75	°C
LDP_TCO_ID_TEMP_DIF_TCO_TIA	3	0...FEH	-48...142.5	0.75	°C
Trigger conditions for combustion camber heating function					
IP_FAC_MAF_CMBC_WF	6	0...80H	0...1	0.0078125	-
LDP_MAF_HB_IP_FAC_CMBC_WF	6	0...FFH	0..1389	5.4272	mg/stk
Load dependency of mass fuel flow correction factor					
IP_FAC_TCO_ST_CMBC_WF	6	0...80H	0...1	0.0078125	-
LDP_TCO_ST_ IP_FAC_TCO_CMBC_WF	6	0...FEH	-48...142.5	0.75	°C
Coolant influence on the mass fuel flow correction factor					
IP_FAC_CMBC_WF_FDOUT	6	0...80H	0...1	0.0078125	-
LDP_MFF_SUM_ IP_FAC_CMBC_WF	6	0...FFFFH	0...44448	0.67823	mg
Controlled fade out of the mass fuel flow correction depending on MFF_SUM					

7.24 Injection Pressure Correction – Homogeneous Mode, 1. Pulse

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TI_1_PRS_HOM	V/O	0 ... FFFFH	0...15.9997	2.4414e-4	-
correction factor to compensate the influence of the varying pressure difference at the injector, 1. pulse					
PRS_DEC_INJ_1_HOM	V/O	0 ... FFFFH	0...173888	2.65	hPa
pressure difference at the injector, 1. pulse					

Input data:

MAP	FUP	FAC_TI_TFU	LV_ST_END

FUNCTION DESCRIPTION:

General information:

The module calculates a correction factor to compensate the influence of the varying pressure difference at the injector (FUP – MAP) and the influence of the fuel temperature on the injected fuel mass.

Application conditions:

Activation: every engine state

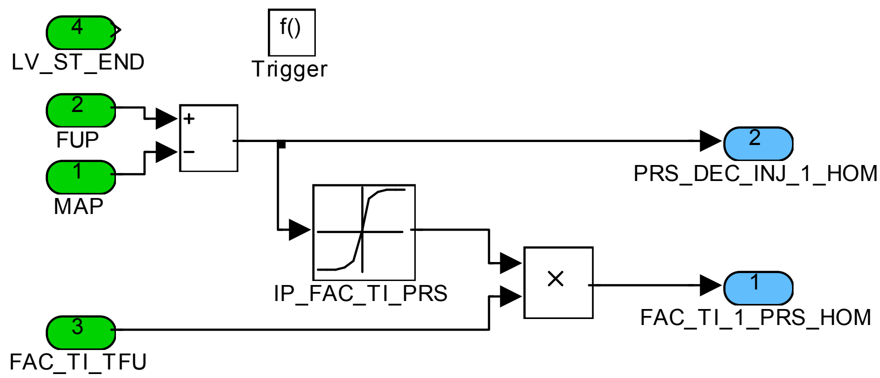
Deactivation: -

Initialisation: at reset: FAC_TI_1_PRS_HOM = 1

Update rate: LV_ST_END = 0: 10 ms

LV_ST_END = 1: segment synchronous

Signal flow diagram:



Formula section:

The pressure difference at the injector, which is input for the map IP_FAC_TI_PRS, is calculated from fuel rail pressure and the manifold pressure.

PRS_DEC_INJ_1_HOM = FUP – MAP

(PRS_DEC_INJ_1_HOM is input data for IP_FAC_TI_PRS)

The total correction factor results from a multiplication of the output of IP_FAC_TI_PRS the fuel temperature dependent input variable FAC_TI_TFU.

FAC_TI_1_PRS_HOM = IP_FAC_TI_PRS * FAC_TI_TFU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TI_PRS	1*12	0...7FFFH	0...7.9997	2.4414e-4	-
LDP_PRS_DEC_INJ_IP_FAC_TI_PRS	12	0...FFFFH	0..173888	2.65	hPa
minimum injection time limitation					

7.25 Predefinition for single injection (FMSP)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ADD_PULSE_ENA	V/O	0 ... 1H	0 ... 1	1	-
Switch to enable additional injection pulse at homogeneous mode, single injection mode (injection update at transient operation)					
INJ_MOD_REQ	O	0 ... FFH	0 ... 255	1	-
Injection mode requested					
MFF_SP_2_HOM[NC_CYL_NR]	V/O	0 ... FFFFH	0 ... 1389	0.02119	mg/tdc
Cylinder individual mass fuel flow setpoint for homogeneous mode, second pulse					
LV_DUI_SWI	O	0 ... 1H	0 ... 1	1	-
Switch for single / double injection					
MFF_SP_1_S[NC_CBK_IN_NR]	O	0 ... FFFFH	0 ... 1389	0.02119	mg/tdc
Mass fuel flow setpoint for stratified mode, first pulse					
MFF_SP_2_S[NC_CBK_IN_NR]	O	0 ... FFFFH	0 ... 1389	0.02119	mg/tdc
Mass fuel flow setpoint for stratified mode, second pulse					
LV_FAC_MFF_SP_HOM_DUI	O	0 ... 1H	0 ... 1	1	-
Boolean variable to indicate whether FAC_MFF_SP_HOM_DUI is set to one or not					
LV_FAC_MFF_SP_S_DUI	O	0 ... 1H	0 ... 1	1	-
Boolean variable to indicate whether FAC_MFF_SP_S_DUI is set to one or not					

Input data:

LV_ST_END	LV_TI_EXT_ADJ[NC_CYL_NR]	NC_INJ_MOD_SINGLE	NC_CBK_IN_NR
NC_CYL_NR			

Application conditions:

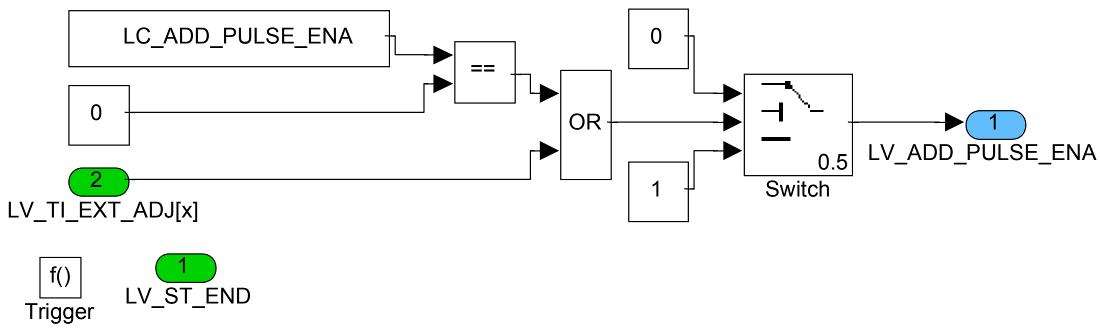
Activation: every engine state

Deactivation: -

Initialization at reset:
LV_DUI_SWI = 0 (single injection)
INJ_MOD_REQ = NC_INJ_MOD_SINGLE
MFF_SP_2_HOM[NC_CYL_NR] = 0
MFF_SP_1_S[NC_CBK_IN_NR] = 0
MFF_SP_2_S[NC_CBK_IN_NR] = 0
LV_FAC_MFF_SP_HOM_DUI = 0
LV_FAC_MFF_SP_S_DUI = 0

Recurrence:
LV_ST_END = 0: 10 ms
LV_ST_END = 1: segment synchronous

Signal flow diagram:



Formula section:

FOR x = 0 **TO** (NC_CYL_NR –1)

IF
LC_ADD_PULSE_ENA = 0
OR
LV_TI_EXT_ADJ[x] = 1
THEN
LV_ADD_PULSE_ENA = 0
ELSE
LV_ADD_PULSE_ENA = 1
ENDIF

END FOR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ADD_PULSE_ENA	1	0 ... 1H	0 ... 1	1	-
Logical constant for enabling additional pulses in case of injection time update at transient conditions, homogeneous mode					

7.26 Fuel temperature correction

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TI_TFU	V/O	0...1FFFH	0...1.9997	2.4414e-4	-
factor for correction of fuel injection time due to a change of fuel temperature variation					

Input data:

TFU			

FUNCTION DESCRIPTION:

General information:

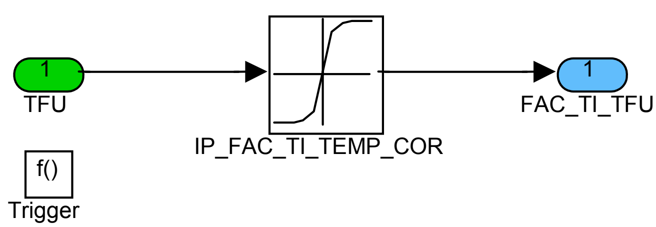
Density fluctuations of fuel due to temperature variation can be compensated with a factor acting on the fuel injection time.

Application condition:

Activation: every engine state
Deactivation: -
Initialisation: at reset: FAC_TI_TFU = 1
Recurrence: every 1s

FUNCTION DESCRIPTION:

Signal flow diagram:



Formula section:

FAC_TI_TFU = IP_FAC_TI_TEMP_COR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TI_TEMP_COR	1*12	0...1FFFH	0...1.9997	2.4414e-4	-
LDP_TFU_IP_FAC_TI_TEMP_COR	12	00...FEH	-48...142.5	0.75	°C
factor for correction of fuel injection due to the fuel temperature					

7.27 Injection time correction by the application system

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TI_TUN_IV[NC_CYL_NR]	V/O	0...FFFFH	0...16	2.44e-4	-
Cylinder selective Injection Time Correction by the Application System - multiplicative value					
TI_TUN_ADD_IV[NC_CYL_NR]	V/O	8000...7FFFH	-131.1 ... 131.1	0.004	ms
Cylinder selective Injection Time Correction by the Application System - additive value					
TI_AS	V/O	0...FFFFH	0...16	2.44e-4	-
Global Injection Time Correction by the Application System - multiplicative					
#if (NC_USE_TI_EXT_ADJ = 1)					
TI_AS_CBK_EX[NC_CBK_EX_NR]	V	0...FFFFH	0...16	2.44e-4	-
Exhaust bank selective injection time correction caused by service tool intervention - multiplicative value					
TI_ADD_AS_CBK_EX[NC_CBK_EX_NR]	V	8000...7FFFH	-131.1 ... 131.1	0.004	ms
Exhaust bank selective injection time correction caused by service tool intervention - additive value					
#endif					

Input data:

LV_FAC_TI_EXT_ADJ	FAC_TI_EXT_ADJ	NC_CYL_NR	NC_CBK_EX_NR
NC_USE_TI_EXT_ADJ			

For development, the application tool can perform a cylinder specific injection time correction. Additionally – for service tool tests – an exhaust bank dependent TI correction can be applied also.

Application condition:

Activation: every engine state

Deactivation: -

Initialization: **at reset:**

(1) FOR x **=** 0 **TO** (NC_CYL_NR -1) **DO:**

TI_TUN_IV[x] = 1

TI_TUN_ADD_IV[x] = 0

(1) ENDFOR

#if (NC_USE_TI_EXT_ADJ = 1)

(2) FOR i **=** 1 **TO** NC_CBK_EX_NR **DO:**

TI_AS_CBK_EX[i] = C_TI_AS_CBK_EX[i]

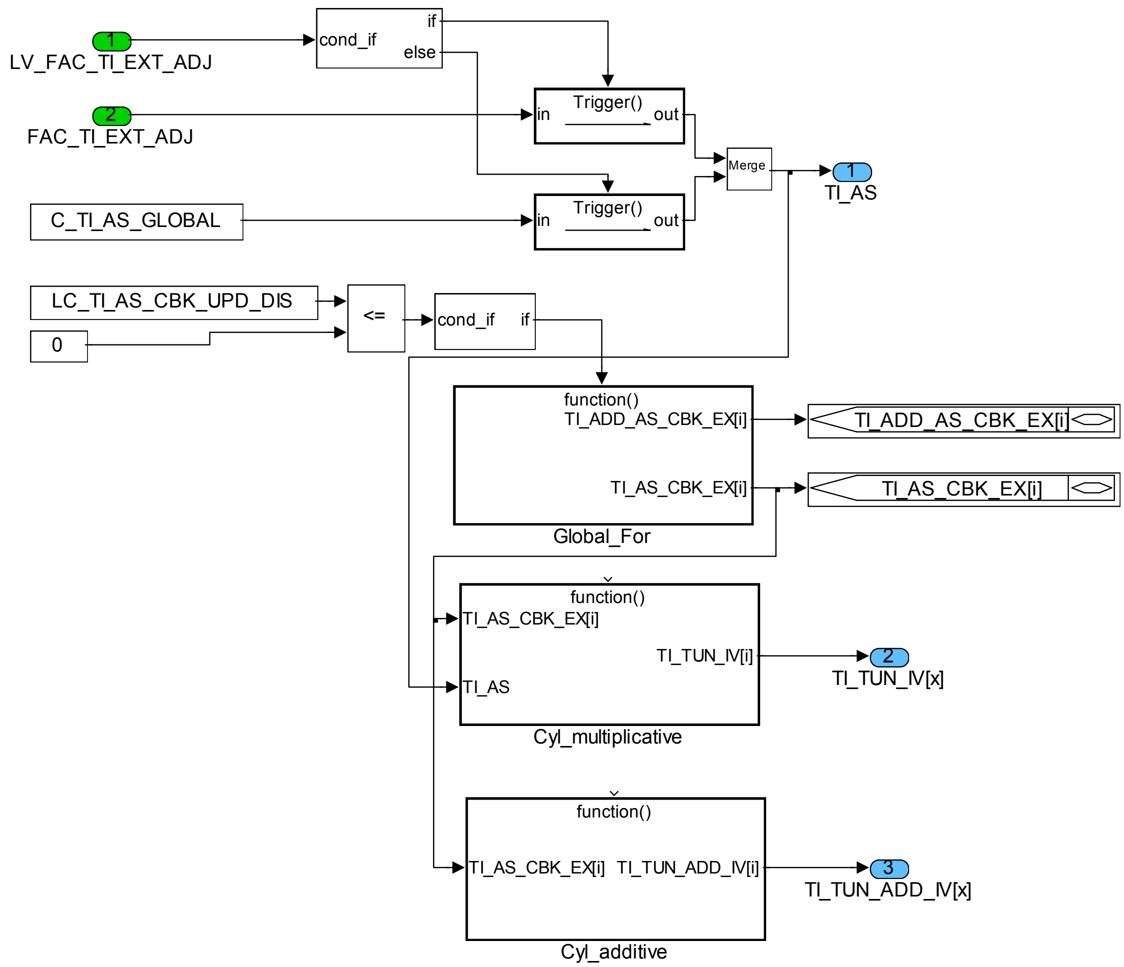
TI_ADD_AS_CBK_EX[i] = C_TI_ADD_AS_CBK_EX[i]

(2) ENDFOR

#endif

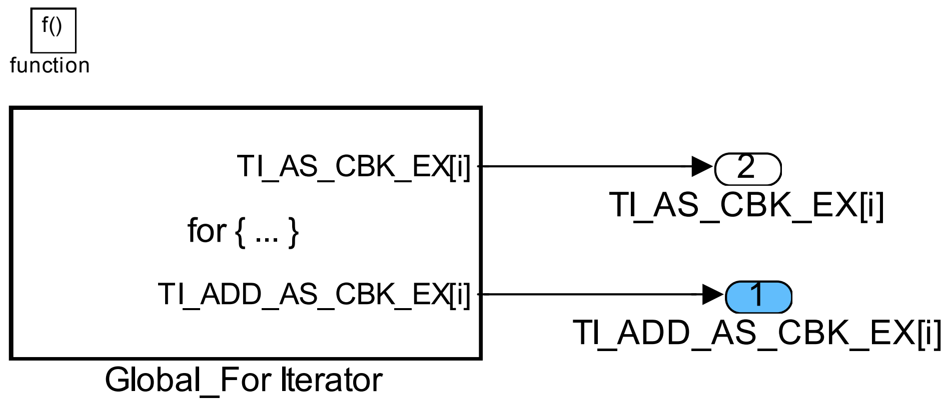
Reccurence: 100ms

Mainsystem:

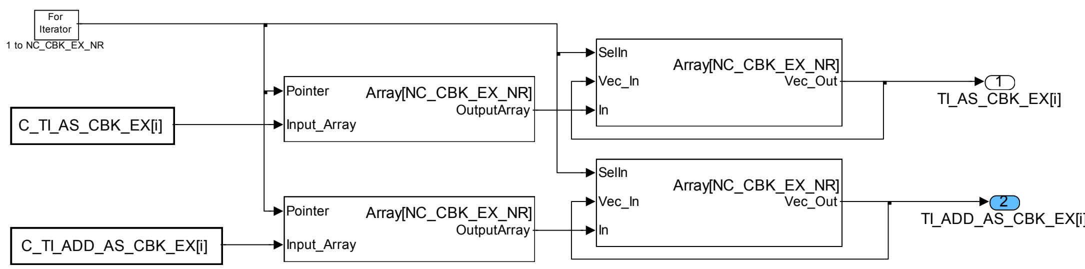


1
LV_FAC_TI_EXT_ADJ
cond_if
if
else
2
FAC_TI_EXT_ADJ
Trigger()
in
out
Merge
1
TI_AS
C_TI_AS_GLOBAL
Trigger()
in
out
LC_TI_AS_CBK_UPD_DIS
0
<=
cond_if
if
function()
TI_ADD_AS_CBK_EX[i]
TI_AS_CBK_EX[i]
Global_For
TI_ADD_AS_CBK_EX[i]
TI_AS_CBK_EX[i]
function()
TI_AS_CBK_EX[i]
TI_TUN_IV[i]
TI_AS
Cyl_multiplicative
2
TI_TUN_IV[x]
function()
TI_AS_CBK_EX[i]
TI_TUN_ADD_IV[i]
Cyl_additive
3
TI_TUN_ADD_IV[x]

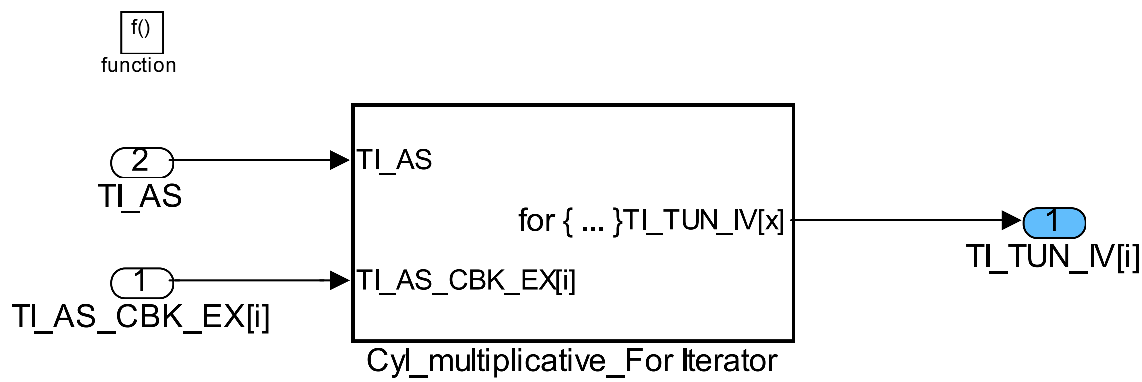
Global_For:



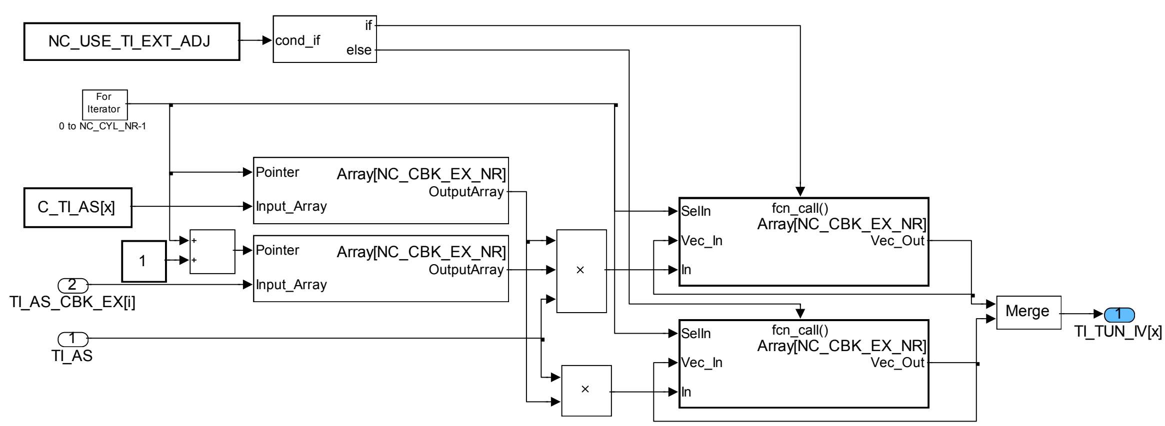
Global_For Iterator:



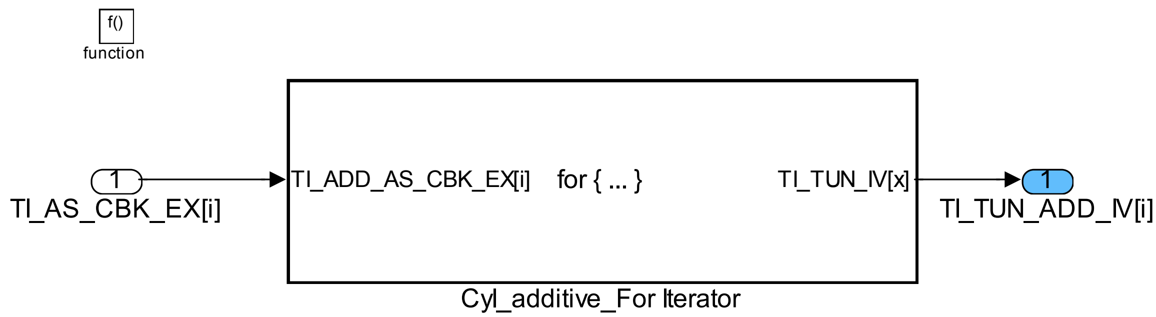
Cyl_multiplicative:



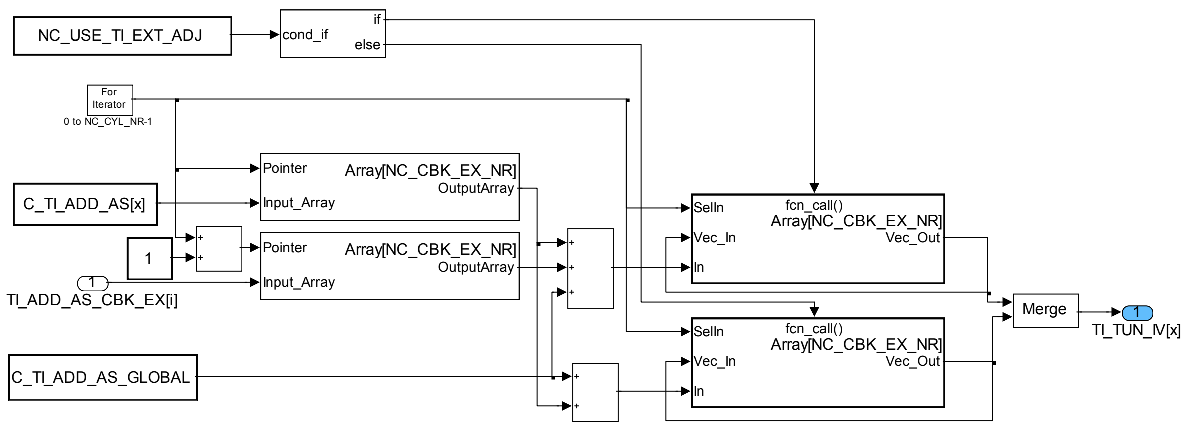
Cyl_mltiplicative_For Iterator:



Cyl_additive:



Cyl_additive_For Iterator:



Formula section:

7.27.1 Global application system/service tool intervention

(1) IF LV_FAC_TI_EXT_ADJ = 1 ... Service tool intervention active

(1) THEN TI_AS = FAC_TI_EXT_ADJ

(1) ELSE TI_AS = C_TI_AS_GLOBAL ... Application system intervention active

(1) ENDIF

#if (NC_USE_TI_EXT_ADJ = 1)

(2) IF LC_TI_AS_CBK_UPD_DIS = 0 ... If switch is not set, update adaptation factors.
If not, a data update from service tool is necessary

(2) THEN

(3) FOR i **=** 1 **TO** NC_CBK_EX_NR **DO:**

TI_AS_CBK_EX[i] = C_TI_AS_CBK_EX[i]

TI_ADD_AS_CBK_EX[i] = C_TI_ADD_AS_CBK_EX[i]

(3) ENDFOR

(2) ENDIF

#endif

7.27.2 Cylinder selective injection time correction - multiplicative

(4) FOR x **=** 0 **TO** (NC_CYL_NR-1) **DO:**

#if (NC_USE_TI_EXT_ADJ = 1)

i = {exhaust bank number which corresponds to cylinder x }

TI_TUN_IV[x] = C_TI_AS[x] ... Cylinder specific application system intervention

* TI_AS ... Global application system/service tool

* TI_AS_CBK_EX[i] ... Used and updated by service tool

#else

TI_TUN_IV[x] = C_TI_AS[x] ... Cylinder specific application system intervention

* TI_AS ... Global application system/service tool

#endif

(4) ENDFOR

7.27.3 Cylinder selective injection time correction – additive

(5) FOR x **=** 0 **TO** (NC_CYL_NR-1) **DO:**

#if (NC_USE_TI_EXT_ADJ = 1)

i = {exhaust bank number which corresponds to cylinder x }

TI_TUN_ADD_IV[x] = C_TI_ADD_AS[x] ... Cylinder specific application system intervention

+ C_TI_ADD_AS_GLOBAL ... Global application system

+ TI_ADD_AS_CBK_EX[i] ... Used and updated by service tool

#else

TI_TUN_ADD_IV[x] = C_TI_ADD_AS[x] ... Cylinder specific application system intervention

+ C_TI_ADD_AS_GLOBAL ... Global application system

#endif

(5) ENDFOR

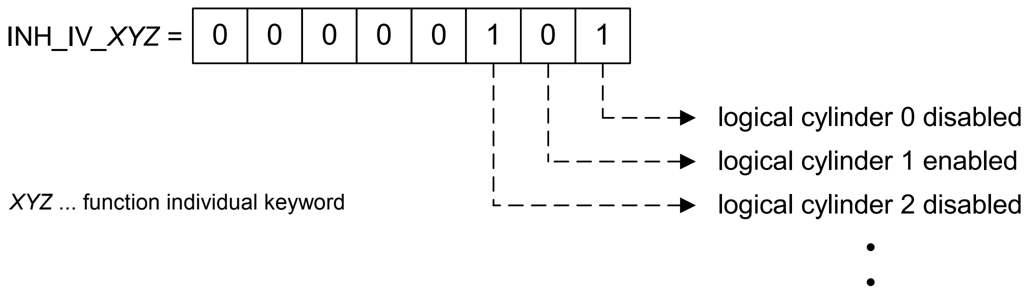
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TI_AS_GLOBAL	1	0...FFFFH	0...16	2.44e-4	-
Global injection time correction					
C_TI_AS[NC_CYL_NR]	1	0...FFFFH	0...16	2.44e-4	-
Injection time factor (cylinder selective) to correct cylinder injection value					
C_TI_ADD_AS_GLOBAL	1	8000...7FFFH	-131.1 ... 131.1	0.004	ms
Global Injection Time Correction - additive value					
C_TI_ADD_AS[NC_CYL_NR]	1	8000...7FFFH	-131.1 ... 131.1	0.004	ms
Cylinder selective Injection Time Correction - additive value					
#if (NC_USE_TI_EXT_ADJ = 1)					
LC_TI_AS_CBK_UPD_DIS	1	0...1H	0...1	1	-
Switch to disable TI_AS_CBK_EX and TI_ADD_AS_CBK_EX data update, for customer delivery it has to be 0					
C_TI_AS_CBK_EX[NC_CBK_EX_NR]	1	0...FFFFH	0...16	2.44e-4	-
Injection time factor (exhaust bank selective) to correct cylinder injection value					
C_TI_ADD_AS_CBK_EX[NC_CBK_EX_NR]	1	8000...7FFFH	-131.1 ... 131.1	0.004	ms
Exhaust bank selective Injection Time Correction - additive value					
#endif					

7.28 Cylinder Shut Off

__Overview__

Static fuel shut off:
In several engine operating- and system states and in case of malfunctions of the injection or ignition system the fuel injection must be disabled at individual cylinders. All the shut off requirements are coordinated, hence these requirements have to have the same structure as described below:



Dynamic fuel shut off:
The information which cylinder has to be shut off static, is coded within a byte (with the length of 8 bit) at the cylinder corresponding position.
In some engine operating states, e. g. fast torque reduction, restart fuel feed or pull fuel cut off, the injection has to be disabled for a certain subsequent numbers of injections. The fuel cut off starts immediately at the next cylinder following an order defined in a pattern. That means, different to the cylinder individual static fuel shut off, there is no predefined and fixed association between cylinders and the shut off sequence.

Please note! Injections, which are already started, are not stopped.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
INH_INJ	O/V	0...FFH	0...255	1	[-]
Final cylinder shut off pattern (fixed cylinder allocation)					
INH_IV	O/V	0...FFH	0...255	1	[-]
Shut off pattern for static cylinder shut off (fixed cylinder allocation)					
INH_IV_IGK	V	0...FFH	0...255	1	[-]
Shut off pattern of IGK					
INH_IV_MON	V	0...FFH	0...255	1	[-]
Shut off pattern of shut off request by monitoring concept					
INH_IV_N_MAX_REQ_FCUT	V	0...FFH	0...255	1	[-]
Shut off pattern of LV_N_MAX_REQ_FCUT (engine speed limitation)					
INH_IV_PUC	V	0...FFH	0...255	1	[-]
Shut off pattern of pull fuel shut off					
INH_SWI_IV	O/V	0...FFFFH	0...65535	1	[-]
Shut off pattern for dynamic cylinder shut off					
LV_FCUT_IND	O/V	0...1H	0...1	1	[-]
At least one cylinder is shut off					
LV_INJ_CUT	O/V	0...1H	0...1	1	[-]
All cylinders shut-off					
LV_SCC[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	[-]
Flag that indicates static single cylinder shut off, exhaust cylinder bank individual					
NR_PAT	O/V	0...FFH	0...255	1	[-]
Index corresponding to the highest shut off level (fuel shut off with pattern)					
NR_PAT_OLD	-	0...FFH	0...255	1	[-]
Index corresponding to the highest shut off level (fuel shut off with pattern) from previous calculation run					
STATE_INH_IV_DYN	O/V	0...FFFFFFFFH	0...4294967295	1	[-]
State of the dynamic cylinder shut off request					
SUM_INH_INJ	O/V	0...8H	0...8	1	[-]
Sum of INH_INJ					
SUM_INH_IV	O/V	0...8H	0...8	1	[-]
Sum of INH_IV					
SUM_INH_IV_CBK[NC_CBK_EX_NR]	O/V	0...8H	0...8	1	[-]
Sum of those bits within INH_IV which are allocated to exhaust cylinder bank i					
SUM_INH_IV_DYN	O/V	0...8H	0...8	1	[-]
Sum of INH_IV_DYN					
SUM_INH_IV_TOT_CBK[NC_CBK_EX_NR]	O/V	0...8H	0...8	1	[-]
Sum of those bits within INH_INJ which are allocated to exhaust cylinder bank i					

Input data:

INH_IV_DYN	INH_IV_EXT	INH_IV_IGC	INH_IV_MIS
LV_IGK	LV_N_MAX_REQ_FCUT	LV_OFF_IV_MON	LV_PUC
LV_SEL_CYL	LV_ST_END	NC_CBK_EX_NR	NC_CYL_NR
NC_INJ_INH_SWI_IV_SHIFT_NR	NC_LAMB_REF	NR_PAT_SCC	NR_PAT_SEL_CYL
NR_TRIG_EOI_LIM_AV			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_INH_SWI_IV	9	0...FFFFH	0...65535	1	[-]
LDP_NR_PAT_ID_INH_SWI_IV	9	0...FFH	0...255	1	[-]
Cylinder shut off pattern for dynamic cylinder shut off					
LC_ENA_FAST_INJ_REAC	1	0...1H	0...1	1	[-]
Switch to enable fast injection reactivation when leaving trailing throttle fuel cutoff					
LC_SWI_MOD_INH_IV_DYN	1	0...1H	0...1	1	[-]

Mode switch between different dynamic fuel shut off algorithm					
LC_SWI_MOD_SCC	1	0...1H	0...1	1	[-]
Mode switch between different LV_SCC[i] calculation (0 : Using static fuel shut-off , 1 : Using static+dynamic fuel shut-off)					

FUNCTION DESCRIPTION:

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: -

Reccurence: if LV_ST_END = 0: 10 ms

if LV_ST_END = 1: segment synchronous

7.28.1 Static fuel shut off

7.28.1.1 Ignition Key

IF LV_IGK = 0

THEN all cylinders are shut off

INH_IV_IGK =

1	1	1	1	1	1	1	1

ELSE no cylinder is shut off

INH_IV_IGK =

0	0	0	0	0	0	0	0

ENDIF

7.28.1.2 Ignition System

INH_IV_IGC (Input pattern)

7.28.1.3 Monitoring Concept

IF LV_OFF_IV_MON = 1

THEN all cylinders are shut off

INH_IV_MON =

1	1	1	1	1	1	1	1

ELSE no cylinder is shut off

INH_IV_MON =

0	0	0	0	0	0	0	0

ENDIF

7.28.1.4 Misfire Detection

INH_IV_MIS (Input pattern)

7.28.1.5 Pull Fuel Cut Off

To guarantee a moderate transition from trailing throttle PU to trailing throttle fuel cut off, the transition is handled by the module “Sequential fuel cut off and restart fuel feed”, see chapter Cylinder Individual Fuel Shut Off with Pattern. After the transition is finished (indicated by the flag LV_SEL_CYL) all cylinders are shut off by INH_IV_PUC.

IF LV_PUC = 1
AND
LV_SEL_CYL = 0

THEN all cylinders are shut off

INH_IV_PUC =	1	1	1	1	1	1	1	1

ELSE no cylinder is shut off

INH_IV_PUC =	0	0	0	0	0	0	0	0

ENDIF

7.28.1.6 Engine speed limitation

IF LV_N_MAX_REQ_FCUT = 1

THEN all cylinders are shut off

INH_IV_N_MAX_REQ_FCUT =	1	1	1	1	1	1	1	1

ELSE no cylinder is shut off

INH_IV_N_MAX_REQ_FCUT =	0	0	0	0	0	0	0	0

ENDIF

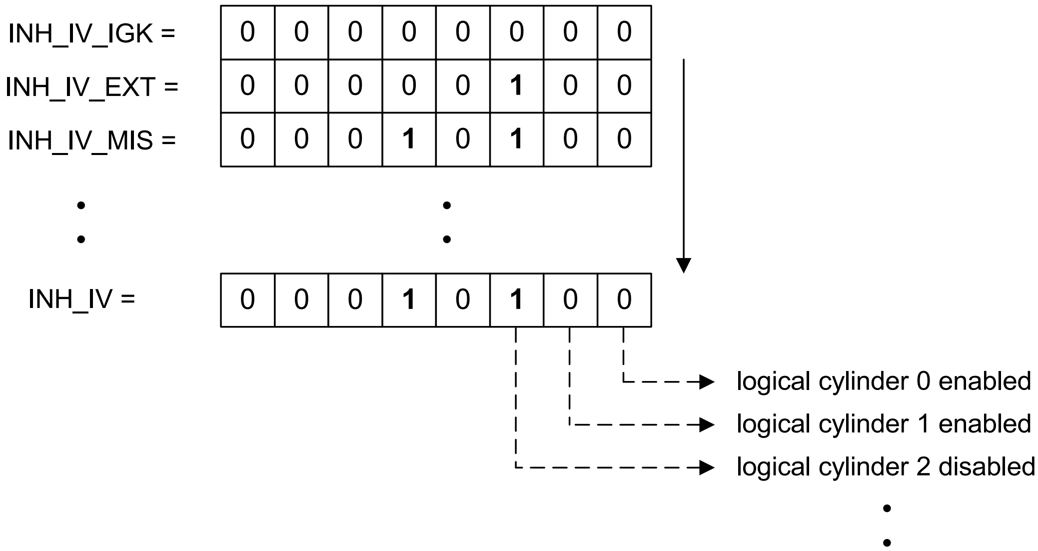
7.28.1.7 Shut Off Pattern from Application Incidences

INH_IV_EXT (Input pattern)

Additional shut off requirements are coordinated within the module “Cylinder Shut Off (Application Incidences)”.

7.28.1.8 Coordination of the Static Fuel Shut Off Requirements

The output pattern INH_IV is the bitwise OR of all pattern, linked with a mask $((2^{NC\_CYL\_NR})-1)$.
As an example see below:



Please note! Injections which are already started are not stopped.

7.28.2 Dynamic fuel shut off

In some engine operating states, e. g. fast torque reduction, restart fuel feed or pull fuel cut off, the injection has to be disabled for a certain subsequent numbers of injections. The fuel cut off starts immediately at the next cylinder following an order defined in a pattern. That means, different to the cylinder individual static fuel shut off, there is no predefined and fixed association between cylinders and the shut off sequence.

Injections, which are already started, are not stopped.

The map ID_INH_SWI_IV contains (NC_CYL_NR * 2 +1) different pattern with calibrate able shut off sequences. Every pattern is allocated to an index NR_PAT (mapping points).

There are two requirements for “Dynamic fuel shut off” to this module:

NR_PAT_SEL_CYL is an index defined by the module “Sequential fuel cut off and restart fuel feed”.

NR_PAT_SCC is an index defined by the module “Torque based pattern calculation”.

NR_PAT is the index corresponding to the highest shut off level (number of cylinders disabled) of these both requirements. A certain pattern is repeated until this index changes.

NR_PAT_OLD = NR_PAT

NR_PAT = MAX (NR_PAT_SEL_CYL, NR_PAT_SCC)

INH_SWI_IV = ID_INH_SWI_IV

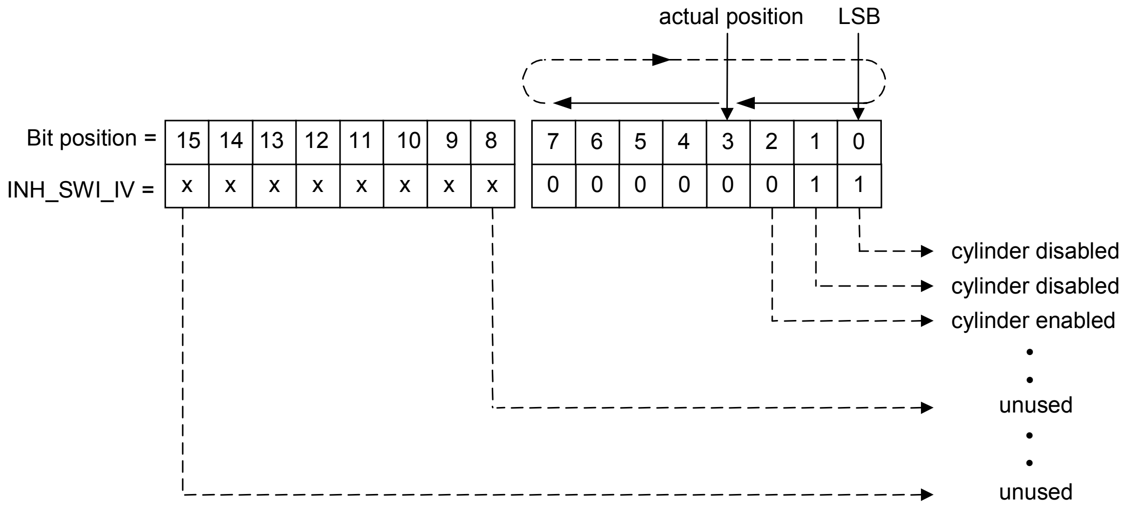
For example a map ID_INH_SWI_IV for an 4-cylinder engine with different shut off sequences :

Shut-off pattern (Index)	Pattern Binary according to firing number for 4cyl engine							
NR_PAT	1	3	4	2	1	3	4	2
0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
2	1	0	0	0	1	0	0	0
3	1	0	0	1	0	0	1	0
4	1	0	1	0	1	0	1	0
5	1	1	0	1	1	1	0	0
6	1	1	1	0	1	1	1	0
7	1	1	1	1	1	1	1	0
8	1	1	1	1	1	1	1	1

Note:

To guarantee that the algorithm works correct, the table ID_INH_SWI_IV has to be applied as follows: The higher the index NR_PAT, the higher the number of cylinders deactivated.

<u>INH_SWI_IV is defined as follows:</u>



(e.g. NC_INJ_INH_SWI_IV_SHIFT_NR = 8, actual bit position = 4):

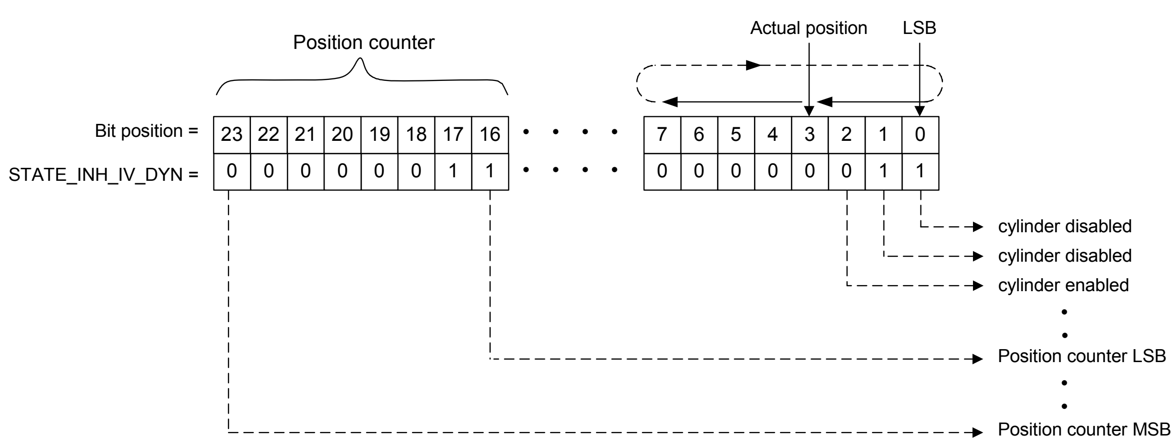
If the bit at a bit position = 1 → cylinder disabled
If the bit at a bit position = 0 → cylinder enabled

The length of the pattern is 16 bit. NC_INJ_INH_SWI_IV_SHIFT_NR defines how many bits are used for the shut off sequence.

NC_CYL_NR < = NC_INJ_INH_SWI_IV_SHIFT_NR < = 16

<u>Calculation of STATE_INH_IV_DYN:</u>

Definition of STATE_INH_IV_DYN:



Note! The position counter data field inside STATE_INH_IV_DYN will be updated at TRIG_EOI_LIM[x].
It contains the 'Actual position' inside the shut off pattern sequence.

Injection reactivation when leaving trailing throttle fuel cutoff:

For a fast injection reactivation when leaving trailing throttle fuel cutoff, the information of the shut off pattern for the dynamic shut off contained in INH_IV_DYN is reset, so INH_INJ is inmediatly reset (see subchapter "Coordination of shut off information for basic software"). The request to activate all cylinders is inmediatly done (INH_INJ = 0), so this allows a faster injection reactivation.It has to be taken into account, that depending on the injection phasing (SOI), the injection will be performed or not.

(1) IF LV_PUC: 1 -> 0 AND LC_ENA_FAST_INJ_REAC = 1

(1) THEN INH_IV_DYN = 0

(1) ENDIF

Select between two different algorithm for STATE_INH_IV_DYN calculation:

(1) IF LC_SWI_MOD_INH_IV_DYN = 0

(1) THEN Classic mode: If NR_PAT increases, then STATE_INH_IV_DYN starts always with a new fuel cut off sequence and 'Actual position' will be set to zero. Otherwise the 'Actual position' stays unchanged and a new shut off pattern – or the old one – is used based on INH_SWI_IV – see picture above.

(2) IF NR_PAT > NR_PAT_OLD

(2) THEN
Start with a new fuel cut off sequence at the first bit position (LSB). Use as sequence the current content of INH_SWI_IV.
STATE_INH_IV_DYN = INH_SWI_IV

(2) ELSE
Continue with the fuel cut off sequence at 'Actual position' and change the shut off pattern depending on INH_SWI_IV:
STATE_INH_IV_DYN = STATE_INH_IV_DYN **AND** 0xFFFF0000 ...bitwise
OR ...bitwise
INH_SWI_IV **AND** 0x0000FFFF ...bitwise

Note! Pay attention of data consistency of STATE_INH_IV_DYN.

(2) ENDIF

(1) ELSE Modern mode: Only if NR_PAT was zero and increases, then STATE_INH_IV_DYN starts with a new fuel cut off sequence and 'Actual position' will be set to zero. Otherwise the 'Actual position' stays unchanged and a new shut off pattern – or the old one – is used based on INH_SWI_IV – see picture above.

(3) IF (NR_PAT_OLD = 0) **AND** (NR_PAT > NR_PAT_OLD)

(3) THEN
Start with a new fuel cut off sequence at the first bit position (LSB). Use as sequence the current content of INH_SWI_IV.
STATE_INH_IV_DYN = INH_SWI_IV

(3) ELSE
Continue with the fuel cut off sequence at 'Actual position' and change the shut off pattern depending on INH_SWI_IV:

STATE_INH_IV_DYN = STATE_INH_IV_DYN **AND** 0xFFFF0000 ...bitwise
OR ...bitwise
INH_SWI_IV **AND** 0x0000FFFF ...bitwise

Note! Pay attention of data consistency of STATE_INH_IV_DYN.

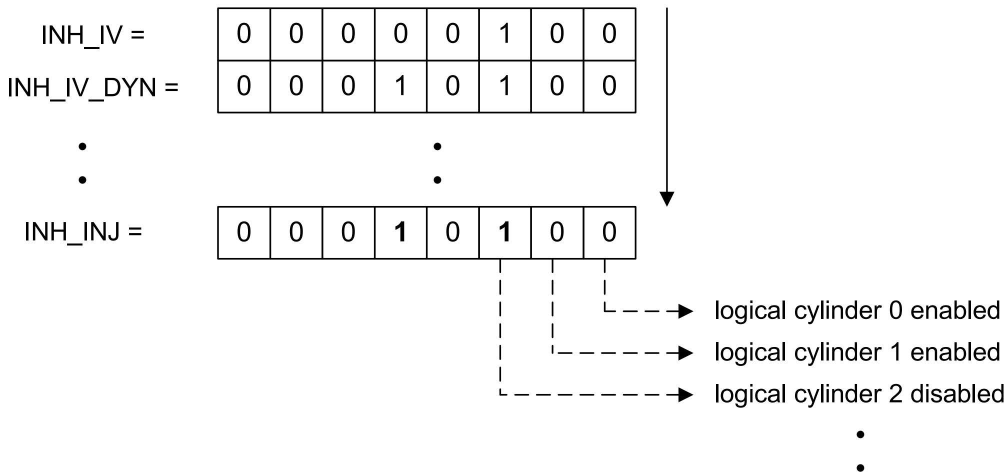
(3) ENDIF

(1) ENDIF

7.28.3 Coordination of shut off information for basic software (INH_INJ)

The output pattern INH_INJ is the bitwise OR of the pattern INH_IV and INH_IV_DYN, linked with a mask $((2^{NC\_CYL\_NR}) - 1)$.

As an example see below:



Note! Pay attention of data consistency of INH_IV.

7.28.4 Shut Off Status Information

SUM_INH_IV .... sum of INH_IV

SUM_INH_IV_CBK_i .... sum of those bits within INH_IV which are allocated to exhaust cylinderbank i, with i = 1 for exhaust cylinderbank 1 and i = 2 for exhaust cylinderbank 2.

SUM_INH_IV_DYN ..... sum of INH_IV_DYN

SUM_INH_INJ ..... sum of INH_INJ

SUM_INH_IV_TOT_CBK_i

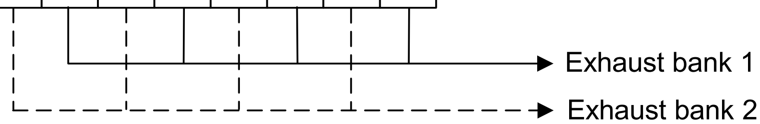
.... sum of those bits within INH_INJ which are allocated to exhaust cylinderbank i, with i = 1 for exhaust cylinder bank 1 and i = 2 for exhaust cylinderbank 2.

Note! Pay attention on data consistency of SUM_INH_INJ and SUM_INH_IV_DYN

Note:

The allocation between physical cylinders and exhaust cylinderbank 1 and 2 is defined by the pattern NC_LAMB_REF. For example an 8 cylinder engine:

7	6	5	4	3	2	1	0	... Logical Cylinder
1	0	1	0	1	0	1	0	... NC_LAMB_REF




```
(1) IF   SUM_INH_INJ <>  0
(1) THEN
        LV_FCUT_IND = 1            (at least one cylinder is shut off)
    (2) IF      SUM_INH_INJ >= NC_CYL_NR
    (2) THEN                       (all cylinders are shut off)
                LV_INJ_CUT = 1
                (3) FOR i = 1 TO NC_CBK_EX_NR DO:
                    LV_SCC[i] = 1        (i = 1 and 2, for exhaust cylinder bank 1 and 2)
                (3) ENDFOR
    (2) ELSE
                LV_INJ_CUT = 0
                (4) FOR i = 1 TO NC_CBK_EX_NR DO:
                    (5) IF      SUM_INH_IV_CBK[i] ≠ 0
                        Or      ( SUM_INH_IV_TOT_CBK[i] ≠ 0
                                And     LC_SWI_MOD_SCC = 1 )
                    (5) THEN
                            LV_SCC[i] = 1
                    (5) ELSE
                            LV_SCC[i] = 0
                    (5) ENDIF
                (4) ENDFOR
    (2) ENDIF
(1) ELSE
        LV_FCUT_IND = 0
        LV_INJ_CUT = 0
        (6) FOR i = 1 TO NC_CBK_EX_NR DO:
            LV_SCC[i] = 0
        (6) ENDFOR
(1)  ENDIF
```

7.29 . Cylinder Shut Off (Application Incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
INH_IV_EXT	V/O	0...FFH	0...255	1	-
Shut off pattern for cylinder shut off (cylinder allocated)					
INH_IV_IMOB	-	0...FFH	0...255	1	-
Cylinder shut off pattern request by immobilizer					
INH_IV_DMF	-	0...FFH	0...255	1	-
Cylinder shut off pattern request by DMF Oscillation Detection					
INH_IV_ENSD	-	0...FFH	0...255	1	-
Cylinder shut off pattern request by Backward Engine Rotation Detection					

Input data:

INH_IV_TPS_AD	INH_IV_MIS_GEN	LV_IMOB_INJ_OFF	INH_IV_EXT_ADJ
LV_ENG_OFF_DMF_INJ	LV_ST_END	LV_IS	LV_IGN_INJ_LOCK_REQ

Application conditions:

Activation: every engine state

Deactivation: -

Initialization: -

Reccurence: if LV_ST_END = 0: 10 ms

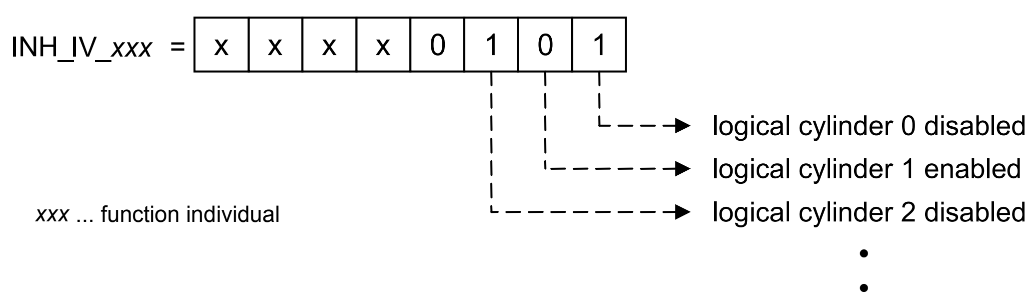
if LV_ST_END = 1: segment synchronous

Formula section:

7.29.1 General

In several engine operating- and system states and in case of malfunctions the fuel injection must be disabled at individual cylinders.

All the shut off requirements are coordinated, hence these requirements have to have the same structure as discribed below:



The information which cylinder has to be shut off static, is coded within a byte (with the length of 4 bit) at the cylinder corresponding position.

7.29.2 Static Fuel Shut Off Requirements

INH_IV_TPS_AD

Shut off request by TPS adaptation, for IGBT protection (all cylinders)

INH_IV_MIS_GEN

Shut off sequence from misfire generator, for tuning purposes (individual cylinders)

INH_IV_IMOB

Shut off request by Immobilizer (all cylinders)

if LV_IMOB_INJ_OFF = 1
then INH_IV_IMOB = xxxx1111
else INH_IV_IMOB = xxxx0000

INH_IV_DMF

Shut off request by Dual Mass Flywheel Oscillation Detection (all cylinders)

if LV_ENG_OFF_DMF_INJ = 1
then INH_IV_DMF = xxxx1111
else INH_IV_DMF = xxxx0000

INH_IV_ENSD

Shut off request by Backward Engine Rotation Detection (all cylinders)

if LV_IGN_INJ_LOCK_REQ = 1
then INH_IV_ENSD = xxxx1111
else INH_IV_ENSD = xxxx0000

INH_IV_EXT_ADJ

Shut off request from external service tester, only in IS available (individual cylinders)

7.29.3 Coordination of the Static Fuel Shut Off Requirements

The output pattern INH_IV_EXT is the bitwise OR of all pattern INH_IV_xxx.
As an example see below:

INH_IV_xxx =	x	x	x	x	0	0	0	0
INH_IV_yyy =	x	x	x	x	0	**1**	0	0
INH_IV_zzz =	x	x	x	x	**1**	**1**	0	0
•				•				
•				•				
INH_IV_EXT =	x	x	x	x	**1**	**1**	0	0

7.30 Injection at restart fuel feed

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
MFF_DIF_WF_x	V	FFFF0001...0H	-1389...0	0.021194	mg/stk
Wall-film degradation					
MFF_ADD_REAC_CYL_x	O/V	0...FFFFH	0...1389	0.021194	mg/stk
Reactivation enrichment					
FAC_TI_REAC_x	O/V	0...FFH	0...1.9921	0.078123	-
Overall factor for reactivation fuel feed					
LV_INF	V	0...1H	0...1	1	-
Basic function Intercept					
LV_TI_REAC	V	0...1H	0...1	1	-
Basic function „Reactivation fuel feed“					
LV_TI_RIS	V	0...1H	0...1	1	-
Basic function „Reactivation fuel feed at idle“					
LV_TI_RPU	V	0...1H	0...1	1	-
Basic function „Reactivation fuel feed at trailing throttle“					
LV_TI_RPL	V	0...1H	0...1	1	-
Basic function „Reactivation fuel feed at part load“					
LV_TI_RAE	V	0...1H	0...1	1	-
Basic function „Reactivation fuel feed at acceleration enrichment“					

Input data:

TCO	N_32	MFF_BAS	LV_STATE_PREV_IV_x
MAP	LV_ST_END	N_GRD	TPS_AV
TPS_GRD	NR_TRIG_EOI_LIM_PREV	NR_TRIG_EOI_LIM_AV	LV_FL
LV_CT	LV_PUC	LV_PU	LV_IS
LV_ES	LV_PL		

General information:

During PUC, the manifold wall-film becomes cut down until it has completely disappeared. In order to guarantee, after a cylinder shut-off, safe firing with fast torque build-up without misfiring of the reactivation cylinder, this must be supplied with an extra amount of fuel for the following reasons:

- increased charge due to lack of residual exhaust gas
- wall-film degradation during cylinder shut-off

Because of the different numbers of cutoff cycles for the cylinders, the extra fuel amount is calculated individually for each cylinder.

The following function delivers, for every cylinder, an additive extra fuel amount MFF_ADD_REAC_CYL_x which considers wall film degradation (partly or completely) as well as an overall factor FAC_TI_REAC for the first 15 engine working cycles after reactivation.

Application conditions:

Activation:	LV_ST_END = 1	
Deactivation:	-	
Initialisation:	at reset and if LV_ES = 0 → 1:	LV_INF = 0 LV_TI_RIS = 0 LV_TI_RPU = 0 LV_TI_RPL = 0 LV_TI_RAE = 0 FAC_TI_REAC = 1
	LV_ES = 0 → 1:	MFF_DIF_WF_x = 0 MFF_ADD_REAC_CYL_x = 0 FAC_TI_REAC_x = 1 CYCNR_REAC = 0
Recurrence:	different	

7.30.1 Control flags for simulation of wall film degradation

7.30.1.1 Intercept function : (LV_INF)

FUNCTION DESCRIPTION:

General information:

In the engine operating states trailing throttle (LV_PU) or trailing throttle fuel cut off (LV_PUC), the **intercept function (LV_INF)** can be enabled below an engine speed threshold C_N_MAX_INF.

For this purpose, the engine speed gradient N_GRD is monitored.

If the negative engine speed gradient exceeds the adjustable constant C_N_GRD_MIN during trailing throttle fuel cut off the injection will be reactivated to counteract the high negative engine speed gradient.

Formula section:

Recurrence: segment synchronous

If	LV_CT	=	1	**and**
	N_32	<	C_N_MAX_INF	**and**
	N_GRD	<	C_N_GRD_MIN	
then	LV_INF	=	1	
else	LV_INF	=	0	

7.30.1.2 Reactivation Fuel Feed functions : (LV_TI_REAC)

In order to rebuild the previously removed wall-applied fuel film during reactivation fuel feed from trailing throttle fuel cut-off (LV_PUC), the injection time is increased by a specific value.

Any transition LV_PUC → LV_IS / LV_PU / LV_PL triggers also the reactivation fuel feed function.

Formula section:

If one of the reactivation fuel feed functions is active **then** LV_TI_REAC is set to 1.

If the Reactivation function is terminated (see Ch. 7: Injection at restart fuel feed, all MFF_ADD_REAC_CYL_x = 0 and all FAC_TI_REAC_x = 1) **or** the engine operating state is neither LV_IS nore LV_PU nore LV_PL **then** LV_TI_REAC is set to 0.

7.30.1.2.1 At Transition from:

(the following calculations are done only at state transition)

PUC to IS

LV_TI_RIS = 1
LV_TI_REAC = 1

LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 0

PUC to PL

IF

(TPS_GRD < ID_TPS_AV_GRD_BOL_AE)

THEN

LV_TI_RPL = 1

LV_TI_RAE = 0

ELSE

LV_TI_RPL = 0

LV_TI_RAE = 1

ENDIF

LV_TI_RIS = 0
LV_TI_REAC = 1

LV_TI_RPU = 0

PUC to PU

LV_TI_RIS = 0
LV_TI_REAC = 1

LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 1

PU to IS

IF
LV_TI_REAC = 1
THEN
LV_TI_RIS = 1
LV_TI_RPL = 0
LV_TI_RAE = 0
LV_TI_RPU = 0
ENDIF

PU to PL

IF

(TPS_GRD < ID_TPS_AV_GRD_BOL_AE)

THEN

LV_TI_RPL = 1

LV_TI_RAE = 0

ELSE

LV_TI_RPL = 0

LV_TI_RAE = 1

ENDIF

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 0

LV_TI_RPU = 0

ENDIF

PU to PUC

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 0
LV_TI_REAC = 0

LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 0

ENDIF

IS to PU

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 0
LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 1

ENDIF

IS to PL

IF

(TPS_GRD < ID_TPS_AV_GRD_BOL_AE)

THEN

LV_TI_RPL = 1

LV_TI_RAE = 0

ELSE

LV_TI_RPL = 0

LV_TI_RAE = 1

ENDIF

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 0
LV_TI_RPU = 0

ENDIF

PL to IS

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 1
LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 0

ENDIF

PL to PU

IF
LV_TI_REAC = 1
THEN

LV_TI_RIS = 0
LV_TI_RPL = 0

LV_TI_RAE = 0

LV_TI_RPU = 1

ENDIF

7.30.2 Simulation of the wall film degradation

Recurrence: segment synchronous

Function description:

Determination of the logical cylinder to be calculated:

Effected logical cylinder x = NR_TRIG_EOI_LIM_AV

In addition:
IF
NR_TRIG_EOI_LIM_PREV not NR_TRIG_EOI_LIM_FMSP_OLD
THEN
an additional logical cylinder x = NR_TRIG_EOI_LIM_PREV is calculated
ENDIF

NR_TRIG_EOI_LIM_FMSP_OLD = NR_TRIG_EOI_LIM_AV

(NR_TRIG_EOI_LIM_FMSP_OLD ..... internal variable)

General information:

The wall-film degradation is represented with the cylinder-individual variables MFF_DIF_WF_x. With beginning cutoff of cylinder x, MFF_DIF_WF_x is allocated the negative current basic fuel mass MFF_BAS, multiplied by ID_FAC_MFF_TCO_NEG_WF. The multiplier takes into account the dependence of the wall film quantity on the coolant temperature. During cutoff of cylinder x, MFF_DIF_WF_x is incremented by an engine speed-dependent value ID_MFF_INC_WF every 720°CRK, until the cylinder cutoff is ended or until MFF_DIF_WF_x = 0 has occurred (MFF_DIF_WF_x > 0).
The start and stop of cylinder cutoff is distinguished by the cylinder selective variable LV_STATE_PREV_IV_x. This LV is updated once per 720°CRK at every cylinders EOI_LIM and shows whether an injection has occurred or not during the previous working cycle.

Formula section:

When LV_STATE_PREV_IV_x changed from 1 to 0:

MFF_DIF_WF_x = -MFF_BAS * ID_FAC_MFF_TCO_NEG_WF

When LV_STATE_PREV_IV_x is 0 and was 0 before:

$$MFF\_DIF\_WF\_x_n = MFF\_DIF\_WF\_x_{n-1} + ID\_MFF\_INC\_WF$$

If
$MFF\_DIF\_WF\_x_n > 0$
then
$MFF\_DIF\_WF\_x_n = 0$

7.30.3 Extra fuel quantity at reactivation

Recurrence: segment synchronous

Function description:

General information:

Additive extra fuel quantity: at reactivation of a cylinder the extra fuel quantity is composed from the current basic fuel mass MFF_BAS multiplied by ID_FAC_MFF_TCO_POS_WF, shortened by the cylinder individual remaining wall-film and multiplied by another factor dependant from manifold pressure MAP. In the subsequent course this extra fuel quantity is reduced by decrementing it till 0.

Multiplicative extra fuel quantity: at reactivation of a cylinder the counter CYCNR_REAC starts with 1 and will be incremented after every (main-) injection once per working cycle. The extra fuel amount is realised by the table value FAC_TI_REAC. Moreover, this factor depends on the type of reactivation.

Formula section:

If LV_STATE_PREV_IV_x changed from 0 to 1:
First enrichment after reactivation (last working cycle with LV_STATE_PREV_IV_x = 0):
MFF_ADD_REAC_CYL_x = (MFF_BAS * ID_FAC_MFF_TCO_POS_WF + MFF_DIF_WF_x) * IP_FAC_MFF_MAP_WF

If LV_STATE_PREV_IV_x is 0 and was 0 before:
MFF_ADD_REAC_CYL_x = (MFF_BAS * ID_FAC_MFF_TCO_POS_WF + MFF_DIF_WF_x) * IP_FAC_MFF_MAP_WF

Following enrichment (when LV_STATE_PREV_IV_x = 1):
$MFF\_ADD\_REAC\_CYL\_x_n = MFF\_ADD\_REAC\_CYL\_x_{n-1} - ID\_MFF\_DEC\_WF$
$MFF\_ADD\_REAC\_CYL\_x_n$ is limited to 0 and the calculation is finished when 0 is reached and 15 cycles are calculated.

When LV_STATE_PREV_IV_x changed from 0 to 1:
CYCNR_REAC_x = 1

Later: $CYCNR\_REAC\_x_n = CYCNR\_REAC\_x_{n-1} + 1$ (till = 16)

When CYCNR_REAC_x < 16,
then FAC_TI_REAC_x = ID_FAC_TI_RIS or ID_FAC_TI_RPL

ID_FAC_TI_RIS is used for “soft REAC” when (LV_TI_RIS = 1 or LV_TI_RPU = 1 LV_TI_RPL = 1) and LV_INF = 0 and LV_FL = 0 and LV_TI_RAE = 0
ID_FAC_TI_RPL is used for “hard REAC” in all other cases.

If
LV_TI_RPL or LV_TI_RAE
then
if
TPS_GRD ≥ ID_TPS_GRD_BOL_AE

then
LV_TI_RAE = 1
LV_TI_RPL = 0
else
LV_TI_RAE = 0
LV_TI_RPL = 1
endif
endif

If
(LV_TI_RIS = 1 or LV_TI_RPU = 1 or LV_TI_RPL = 1)
and LV_INF = 0
and LV_FL = 0
and LV_TI_RAE = 0
then
FAC_TI_REAC_x = ID_FAC_TI_RIS
else
FAC_TI_REAC_x = ID_FAC_TI_RPL
endif

If a change of the status “soft REAC” to the status “hard REAC” once occurs (change from ID_FAC_TI_RIS to ID_FAC_TI_RPL), the status “hard REAC” is kept until the reactivation is finished (all variables FAC_TI_REAC_x are similar to 1).

If CYCNR_REAC_x = 16,
then FAC_TI_REAC_x = 1.0

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
ID_FAC_MFF_TCO_NEG_WF	1x8	0...FFH	0...0.49805	0.001953125	-
LDPM_TCO_1_FMSP	8	0...FEH	-48...142.5	0.75	°C
Wall film factor, begin of fuel cutoff					
ID_MFF_INC_WF	1x8	0...FFFFH	0...1389	0.02119	mg/stk
LDPM_N_32_1_FMSP	8	0...FFH	0...8160	32	rpm
Wall-film degradation					
ID_FAC_MFF_TCO_POS_WF	1x8	0...FFH	0...0.49805	0.001953125	-
LDPM_TCO_1_FMSP	8	0...FEH	-48...142.5	0.75	°C
Wall film factor, end of fuel cutoff					
IP_FAC_MFF_MAP_WF	1x4	0...FFH	0...0.996	0.0039063	-
LDP_MAP_IP_FAC_MFF_MAP_WF	4	0...FFFFH	0...5434	0.083	hPa
MAP dependant factor					
ID_MFF_DEC_WF	1x8	0...FFFFH	0...1389	0.02119	mg/stk
LDPM_N_32_1_FMSP	8	0...FFH	0...8160	32	rpm
Wall film build-up					
ID_FAC_TI_RIS	1x16	0...FFH	0...1.9922	0.0078125	-
LDP_CYCNR_REAC_ID_FAC_TI_RIS	16	1...10H	1...16	1	-
Reactivation correction, soft reactivation					
ID_FAC_TI_RPL	1x16	0...FFH	0...1.9922	0.0078125	-
LDP_CYCNR_REAC_ID_FAC_TI_RPL	16	1...10H	1...16	1	-
Reactivation correction, hard reactivation					
C_N_MAX_INF	1	0...FFH	0...8160	32	rpm
Maximum engine speed for intercept function					
C_N_GRD_MIN	1	80...00H	-4096...0	32	rpm/s
engine speed gradient threshold for activating the intercept function					
ID_TPS_GRD_BOL_AE	6*4	0 ... FFH	0 ... 2987.5	11,71	°TPS/ sec
LDP_N_32_ ID_TPS_GRD_BOL_AE	6	0 ... FFH	0 ... 8160	32	rpm
LDP_TPS_AV_ID_TPS_GRD_BOL_AE	4	0 ... 3FFFH	0 ... 119,53	0,0073	°TPS
Throttle position gradient threshold for ignition angle intervention in LV_RPL					

7.31 Sequential fuel cut off and restart fuel feed

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_SEL_CYL	V/O	0...1H	0...1	1	-
Boolean for Cylinder specific transition PU-PUC, PUC-IS or PUC-PL ongoing					
NR_PAT_SEL_CYL	V/O	0...FFH	0...255	1	-
Selected index of Fuel cut off pattern for transition PU-PUC, PUC-IS, PUC-PL					
T_FCUT_AJ	V	0...FF H	0...2.55	0.01	s
Timer value during jerk minimal sequential fuel cut-off					
T_OSC_DT	V	0...FF H	0...2.55	0.01	s
Period of time of the oscillation of the drive-train					
NR_SEL_CYL	V	1...8 H	1...8	1	-
Index number of the selective cylinder pattern					
CASE_SEL_CYL	V	0...3 H	0...3	1	-
Case selector					
TQI_SP_SLOW_DIF	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Difference in requested torque for slow path from actual to last segment					

Input data:

LV_IGK	N_DIF_COR	N_GRD	TQI_SP_SLOW
LV_PU	LV_PUC	LV_PL	LV_IS
LV_DT	LV_AT	GEAR	LV_ES
LV_ST	T_SEG_AV		

FUNCTION DESCRIPTION:

General information:

At transition from trailing throttle to trailing throttle fuel cut off, torque jumps generally occur at trailing throttle fuel cut off which are noticeable as jerks in the vehicle. To reduce the torque at this transition, the ignition timing is retarded and the combustion air is adjusted to small values.

These procedures however do not always achieve the required torque reduction, because ignition timing and combustion air can only be reduced that far, that jerk free engine operation is still possible.

To support the above mentioned torque reduction measurement, it is possible that the injection is switched on and/or switched off for individual cylinders within a defined number of engine cycles (on and/or deactivation) incase of PUC and reentry into Idle.

The engine speed gradient is monitored in order to cancel the selective reentry idle at rapid engine speed drops. If the engine speed gradient is below the threshold C_N_GRD_MIN_SEL_RIS or N_DIF_COR is greater than or equal to C_N_DIF_MAX_SEL_RIS then all cylinders are immediately activated (Pattern index 0)

7.31.1 Calculation of TQI_SP_SLOW dynamic

Description:

This value is needed for fast restart of fuel feed.

Application conditions:

Initialisation: TQI_SP_SLOW_DIF = 0

Recurrence: 20ms

Activation: at every engine state

Deactivation: -

Formula section:

TQI_SP_SLOW_DIF = TQI_SP_SLOW(n) - TQI_SP_SLOW(n-1)

7.31.2 NR_PAT_SEL_CYL calculation

Recurrence: Segment-synchronous

7.31.2.1 Calculation of T_OSC_DT:

```
if LV_AT = 0
then      T_OSC_DT = ID_T_OSC_DT(GEAR)
else      T_OSC_DT = 0
endif
```

Description (Example for 4-cylinder engines)

At a transition LV_PU → LV_PUC,. LV_PUC → LV_IS and LV_PUC → LV_PL , 32 cycles (8 x 720°CRK) in case of a 4-cylinder can be influenced. To realize this, a table ID_PAT_SEL_CYL contains the sequences of injection patterns (0...7), predefined in the table ID_PAT_IND. “ NR_SEL_CYL ” means the logical number in the shut off sequence (1...8).

Application conditions:

Activation:

At active cylinder specific transitions LV_SEL_CYL =1

7.31.2.2 Transition from PU → PUC (CASE 1)

The entrance in the transition pattern for PUC follows, when all conditions for PUC are fulfilled.

if LC_ENA_FCUT_AJ = 0 or LV_AT = 1
then
NR_SEL_CYL starts from 1 and is incremented every 720 °KW until its maximum value. In case of deactivation NR_SEL_CYL is reset to 1.

```
else
if (LV_DT = 0 and LC_INH_DT_PU = 0)
then    NR_SEL_CYL = 8
        NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)
        T_FCUT_AJ is initialized with T_OSC_DT / 2
else    NR_SEL_CYL = C_NR_SEL_CYL_1
        NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)
        LV_SEL_CYL = 1
        if T_FCUT_AJ - T_SEG_AV = 0
```

then NR_SEL_CYL = 8

NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)

T_FCUT_AJ is initialized with T_OSC_DT / 2

LV_SEL_CYL = 0

else $NR\_SEL\_CYL_n = NR\_SEL\_CYL_{n-1}$

T_FCUT_AJ is decremented

end

end

end

The transition is completed, when the last of the 32 transition cycles has run out. The pattern index applied as last one has to be the highest used for the cut-off strategy , that means all cylinders are cut off. NR_SEL_CYL is reset to 1.

If in the course of the transition pattern or while the time sequence an engine operating state change to PU, PL or IS happens, all cylinders are switched on immediatedly (pattern index : “0“), NR_SEL_CYL is reset to 1.

7.31.2.3 Transition from PUC → IS or PUC → PU (CASE 2)

The entrance in the transition pattern for RIS (reactivation idle speed) follows, when all conditions for IS resp. PU are fulfilled.

if LC_ENA_FCUT_AJ = 0 or LV_AT = 1

then

NR_SEL_CYL starts from 1 and is incremented every 720 °KW until its maximum value.

else

if (LV_DT = 0 and LC_INH_DT_PU = 0)

then NR_SEL_CYL = 8

NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)

T_FCUT_AJ is initialized with T_OSC_DT / 2

else NR_SEL_CYL = C_NR_SEL_CYL_2

NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)

LV_SEL_CYL = 1

if T_FCUT_AJ - T_SEG_AV = 0

then NR_SEL_CYL = 8

NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)

T_FCUT_AJ is initialized with T_OSC_DT / 2

LV_SEL_CYL = 0

else $NR\_SEL\_CYL_n = NR\_SEL\_CYL_{n-1}$

T_FCUT_AJ is decremented

end

end

end

The transition pattern is completed, when the last cycle of the 32 transition cycles has run out. The pattern number applied as last one has to be „0“ , that means all cylinders are switched on. NR_SEL_CYL is reset to 1.

If in the course of the transition pattern or while time sequence a engine operating state change to PUC happens, the transition pattern or time sequence for RIS runs out to its end and after that the transition pattern for PUC starts, if required.

If the function is deactivated (see deactivation conditions), all cylinders are switched on immediatedly. In case of deactivation NR_SEL_CYL is reset to 1.

7.31.2.4 Transition from PUC → PL (CASE 3)

The entrance in the transition pattern for RPL (reactivation part load) follows, when all conditions for PL are fulfilled.

```
if LC_ENA_FCUT_AJ = 0 or LV_AT = 1
then
        NR_SEL_CYL starts from 1 and is incremented every 720 °KW until its
        maximum value. In case of deactivation NR_SEL_CYL is reset to 1.
else
        if (LV_DT = 0 and LC_INH_DT_PU = 0)
        then    NR_SEL_CYL = 8
                NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)
                T_FCUT_AJ is initialized with T_OSC_DT / 2
        else    NR_SEL_CYL = C_NR_SEL_CYL_3
                NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)
                LV_SEL_CYL = 1
                if T_FCUT_AJ - T_SEG_AV = 0
                then    NR_SEL_CYL = 8
                        NR_PAT_SEL_CYL = ID_PAT_SEL_CYL(NR_SEL_CYL)
                        T_FCUT_AJ is initialized with T_OSC_DT / 2
                        LV_SEL_CYL = 0
                else    NR_SEL_CYL_n = NR_SEL_CYL_n-1
                        T_FCUT_AJ is decremented
                end
        end
end
```

The transition pattern is completed, when the last cycle of the 32 transition cycles has run out. The pattern number applied as last one has to be „0“ , that means all cylinders are switched on. NR_SEL_CYL is reset to 1.

If in the course of the transition pattern a engine operating state change to PUC happens, the transition pattern for reentry PL runs out to its end and after that the transition pattern for PUC starts, if required. If the function is deactivated, all cylinders are switched on immediatedly.

Deactivation:

- LV_ES = 1 **or**
- LV_ST = 1 **or**
- LV_IGK = 0 **or**
- Pattern completed **or**
- N_GRD ≤ C_N_GRD_MIN_SEL_RIS **or**
- N_DIF_COR ≥ C_N_DIF_MAX_SEL_RIS **or** (only valid for transition PUC → IS resp PUC → PU ; checked each segment)
- TQI_SP_SLOW_DIF ≥ C_TQI_SP_SLOW_DIF_MAX (only valid for transition PUC →PL; checked each segment)

At passive cylinder specific transitions:

LV_SEL_CYL = 0 and NR_SEL_CYL = 1

T_FCUT_AJ is initialized with T_OSC_DT / 2

Transition tables

The transition pattern indexes are adjustable in the table ID_PAT_SEL_CYL. There are 8 fade out patterns which are applicable per transition table (definition of patterns see chapter 7, "Cylinder shut off with pattern").

The following cases are distinguished in ID_PAT_SEL_CYL:

CASE 1: Trailing throttle fuel cut off (PUC)
CASE 2: Restart IS (RIS)
CASE 3: Restart PL (RPL)

Example for shut-off sequences with 8 levels and a shut off pattern for the 4 next injections (4-cylinder)

CASE_SEL_CYL↓ NR_SEL_CYL →

	1	***2***	***3***	***4***	***5***	***6***	***7***	***8***
1	1	1	2	2	3	3	4	4
2	5	5	5	6	6	7	7	0
3	5	6	6	7	0	0	0	0

According shut off sequences: (0 : Injection, 1: no injection)

Case 1 1 0 0 0 1 0 0 0 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1

Case 2 0 1 1 1 0 1 1 1 0 1 1 1 0 1 0 1 0 1 0 1 0 0 0 1 0 0 0 1 0 0 0 0

Case 3 0 1 1 1 0 1 0 1 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Formula section:

NR_PAT_SEL_CYL = ID_PAT_SEL_CYL

Application assistances

Handling of reactivation functions (out of PUC):

Injection (Reactivation fuel feed) : To ensure, that all cylinders get at least once the same amount of TI_ADD_REAC, TI_ADD_REAC is not decremented as long as sequential transition PUC → IS resp. PUC → PL is ongoing.

Ignition: The according reactivation functions are not concerned by sequential transition functions.

Handling of engine states:

The detection of PU, PUC, PL and IS is not concerned by sequential transition functions.

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
ID_PAT_SEL_CYL	3*8	0...FFH	0...255	1	-
LDP_CASE_SEL_CYL_ID_PAT_SEL_CYL	3	1...3H	1...3	1	-
LDP_NR_SEL_CYL_ ID_PAT_SEL_CYL	8	1...8H	1...8	1	-
Table for cylinder specific restart and trailing throttle fuel cut off					
C_N_GRD_MIN_SEL_RIS	1	80H...7FH	-4096...4064	32	1/min/s
N_GRD threshold for cancelling cylinder individual cut-off for restart IS					
C_N_DIF_MAX_SEL_RIS	1	8000H..7FFFH	-32768..32767	1	rpm
N_DIF_COR threshold for deactivation of cylinder individual cut-off for restart IS					
ID_T_OSC_DT	8	0...FF H	0 ... 2.55	0.01	s
LDP_GEAR_1_DRVB	8	0...FF H	0 ... 255	1	-
Characteristic period of time of the eigenfrequency of the powertrain					
LC_INH_DT_PU	1	0 , 1 H	0 , 1	1	-
Logical variable to inhibit LV_DT impact on jerk minimal sequential fuel cut-off activation					
LC_ENA_FCUT_AJ	1	0 , 1 H	0 , 1	1	-
Logical variable to enable jerk minimal sequential fuel cut-off					
C_NR_SEL_CYL_1	1	1...8 H	1...8	1	-
Start pattern index for transition from PU -> PUC (case 1)					
C_NR_SEL_CYL_2	1	1...8 H	1...8	1	-
Start pattern index for transition from PUC -> IS, PU (case 2)					
C_NR_SEL_CYL_3	1	1...8 H	1...8	1	-
Start pattern index for transition from PUC -> PL (case 3)					
C_TQI_SP_SLOW_DIF_MAX	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
TQI_SP_SLOW_DIF threshold for cancelling cylinder individual cut-off for restart in PL					

7.32 Upstream oxygen sensor operative readiness detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LS_UP_READY[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Logical variable for operative readiness of upstream oxygen sensor					
LV_LS_UP_READY_CDN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Logical variable for forced operative readiness of upstream oxygen sensor					
LV_DLY_CHK_LS_UP_READY[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Logical variable set if delay time of operability detection has elapsed					
LV_T_TOUT_LS_UP_READY[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Logical variable set if delay time T_TOUT_LS_UP_READY has elapsed					
T_TOUT_LS_UP_READY[NC_CBK_EX_NR]	V	0...FFFFH	0...1310.7	0.02	[s]
Timer for forced activation of linear lambda sensor					
T_DLY_CHK_LS_UP_READY[NC_CBK_EX_NR]	V	0...FFFFH	0...1310.7	0.02	[s]
Delay timer for start of readiness detection after engine start					
LV_LS_UP_DEW_DC_VLD[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Dewpoint passed at least once in this driving cycle					
LV_MAF_INT_PUC_ACT_VLD	V	0...1H	0...1	1	[-]
Flag indicating that using MAF_INT_PUC_ACT is safe					

Input data:

VLS_UP[NC_CBK_EX_NR]	R_IT_LS_UP[NC_CBK_EX_NR]		
TEG_DYN_LS_UP[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_ERR_STK_LS_UP[NC_CBK_EX_NR]	MAF_INT_PUC_ACT
LV_PUC	LV_TEMP_DEW_LS_UP[NC_CBK_EX_NR]	TCO_ST	NC_CBK_EX_NR
LV_ST_END			

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

The Lambda probe is initialized as not operable (LV_LS_UP_READY[i] = 0, LV_LS_UP_READY_CDN[i] = 0) after leaving engine start. Also, a delay of length ID_T_DLY_CHK_LS_UP_READY is started, which has to pass before the main part of the readiness check is enabled.

Independently, a warmup delay timer starts with the value of ID_T_TOUT_LS_UP_READY whenever a prolonged fuel cut off phase has ended or if the dewpoint was exceeded for the first time in a driving cycle. This is to diagnose the time until readiness recognition after start and after excessive cooling due to driving conditions. The flag LV_T_TOUT_LS_UP_READY[i] is set if this time has elapsed and readiness check is enabled via LV_T_DLY_CHK_LS_UP_READY[i].

The Lambda probe is recognized as ready when VLS_UP[i] falls below the lean voltage threshold C_VLS_UP_AFL_THD_READY or when VLS_UP[i] rises above the rich voltage threshold C_VLS_UP_AFR_THD_READY or when the internal resistance R_IT_LS_UP[i] falls below the threshold C_R_IT_THD_LS_UP_READY.

The initial delay occurs only once after engine start, while the warmup delay is applied every time the dewpoint is revoked or a long fuel cutoff phase is detected. If extended fuel cutoff is expected to cool down the sensor too much, a limit for the mass air flow integral during PUC may also be applied.

Limited operability

If a heater error (LV_ERR_LSH_UP[i] = 1) or a sensor signal stroke error (LV_ERR_STK_LS_UP[i] = 1) is present, readiness detection will take into account heating by the exhaust gas. The flag LV_LS_UP_READY_CDN[i] is set to indicate this limited operability condition. In this mode, readiness is assumed if the exhaust gas temperature exceeds the threshold C_TEG_DYN_MIN_LS_UP_READY, the dewpoint was passed and the signal has exceeded the limits of C_VLS_UP_AFL_MAX_DIAG_READY or C_VLS_UP_AFR_MIN_DIAG_READY at least once.

Readiness is revoked if the temperature falls below the threshold. The flag LV_T_TOUT_LS_UP_READY[i] is reset to zero and the warmup timer is restarted as soon as the temperature is high enough again.

REMARKs:

1) Delayed readiness detection is monitored by testing for (LV_T_TOUT_LS_UP_READY[i] = 1 and LV_LS_UP_READY[i] = 0). If necessary for other reasons, closed loop lambda control after ID_T_TOUT_LS_UP_READY timeout may be forced by using (LV_T_TOUT_LS_UP_READY[i] = 1) as an alternative condition for closed loop enable.
2) This revision of readiness detection relies on a good dewpoint detection to run soon after engine start.
3) If excessive cooling in PUC is not a concern, C_MAF_INT_PUC_MAX_LS_UP_READY must be set to its upper limit (hex value FFFFh). If readiness must be revoked during long PUC phases, this constant should be set to the maximum length of a PUC phase which is tolerated by the sensor.

<u>**Application conditions:**</u>

Initialisation:

<u>At ECU reset, LV_IGK 0→1 and when clearing error memory:</u>
LV_LS_UP_READY[i] = 0
LV_LS_UP_READY_CDN[i] = 0
LV_T_TOUT_LS_UP_READY[i] = 0
LV_DLY_CHK_LS_UP_READY[i] = 0
LV_LS_UP_DEW_DC_VLD[i] = 0
LV_MAF_INT_PUC_ACT_VLD = 0
T_TOUT_LS_UP_READY[i] = C_T_TOUT_LS_UP_READY
T_DLY_CHK_LS_UP_READY[i] = ID_T_DLY_CHK_LS_UP_READY

Recurrence:

T_SAMPLE = 20ms

Activation:

LV_ST_END = 1

Deactivation:

LV_ST_END = 0

Formula section:

if T_DLY_CHK_LS_UP_READY[i] = 0
then
LV_DLY_CHK_LS_UP_READY[i] = 1
else
dec T_DLY_CHK_LS_UP_READY[i]
end

% create a flag that reacts to the first dewpoint passage only

If LV_TEMP_DEW_LS_UP[i] = 1
then LV_LS_UP_DEW_DC_VLD[i] = 1
else LV_LS_UP_DEW_DC_VLD[i] = 0
end

% create a flag that is set if MAF_INT_PUC_ACT can safely be used for diagnosis
% (catch the case where LV_PUC=1 and MAF_INT_PUC_ACT still has the old value)

If (LV_PUC = 0)
then
LV_MAF_INT_PUC_ACT_VLD = 0
elseif MAF_INT_PUC_ACT ≤ C_MAF_INT_PUC_MAX_LS_UP_READY
% must not touch the flag if MAF_INT_PUC_ACT fulfils the comparision in the condition below
then
LV_MAF_INT_PUC_ACT_VLD = 1
end

If (LV_LS_UP_DEW_DC_VLD[i] = 0 **or** (LV_MAF_INT_PUC_ACT_VLD = 1 **and**
MAF_INT_PUC_ACT > C_MAF_INT_PUC_MAX_LS_UP_READY))
then

% sensor signal declared unavailable due to operating conditions, arrange for new timeout cycle

LV_LS_UP_READY[i] = 0

LV_T_TOUT_LS_UP_READY[i] = 0

T_TOUT_LS_UP_READY[i] = C_T_TOUT_LS_UP_READY

else

% Part 1: timeout calculation

if T_TOUT_LS_UP_READY[i] > 0

then

dec T_TOUT_LS_UP_READY[i]

end

if LV_DLY_CHK_LS_UP_READY[i] = 1

then

if T_TOUT_LS_UP_READY[i] = 0

then

LV_T_TOUT_LS_UP_READY[i] = 1

end

% Part 2: do the main part of readiness checks

execute the section "readiness determination"

end

end

section "readiness determination":

if (LV_ERR_LSH_UP[i] = 1 **or** LV_ERR_STK_LS_UP[i] = 1)

then

% limited operability: set/reset readiness flag depending on exhaust gas temperature

LV_LS_UP_READY_CDN[i] = 1

if (TEG_DYN_LS_UP[i] > C_TEG_DYN_MIN_LS_UP_READY **and**

(VLS_UP[i] ≥ C_VLS_UP_AFR_MIN_DIAG_READY **or** VLS_UP[i] <

C_VLS_UP_AFL_MAX_DIAG_READY))

then

LV_LS_UP_READY[i] = 1

elseif (TEG_DYN_LS_UP[i] ≤ C_TEG_DYN_MIN_LS_UP_READY)

then

% start new timeout cycle before re-enabling the sensor signal

LV_LS_UP_READY[i] = 0

LV_T_TOUT_LS_UP_READY[i] = 0

T_TOUT_LS_UP_READY[i] = C_T_TOUT_LS_UP_READY

```
        end
    else
    % normal operation: monitor signal level and internal resistance for thresholds
        if (VLS_UP[i] < C_VLS_UP_AFL_THD_READY or
        VLS_UP[i] > C_VLS_UP_AFR_THD_READY or
        R_IT_LS_UP[i] < C_R_IT_THD_LS_UP_READY)
        then
            LV_LS_UP_READY[i] = 1
        end
    end
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_T_DLY_CHK_LS_UP_READY	8	0...FFFFH	0...1310.7	0.02	[s]
LDP_TCO_ST_ID_T_DLY_CHK_LS_UP	8	0...FEH	-48...142.5	0.75	[°C]
Delay time for begin of readiness detection after engine start					
C_VLS_UP_AFL_THD_READY	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Lean VLS_UP[i] threshold for operative readiness detection					
C_VLS_UP_AFR_THD_READY	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Rich VLS_UP[i] threshold for operative readiness detection					
C_VLS_UP_AFR_MIN_DIAG_READY	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Rich VLS_UP[i] diagnostic threshold for detection of limited operability					
C_VLS_UP_AFL_MAX_DIAG_READY	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Lean VLS_UP[i] diagnostic threshold for detection of limited operability					
C_R_IT_THD_LS_UP_READY	1	0...FFFFH	0...65535	1	[Ohm]
Internal resistance threshold for detection of limited sensor operability					
C_TEG_DYN_MIN_LS_UP_READY	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Exhaust gas temperature minimum for the detection of the limited sensor operability					
C_MAF_INT_PUC_MAX_LS_UP_READY	1	0...FFFFH	0...2912.66666	0.0444444	[g]
Maximum MAF integral value during PUC used for reset of operability					
C_T_TOUT_LS_UP_READY	1	0...FFFFH	0...1310.7	0.02	[s]
Time-out constant for readiness detection -> forced readiness detection					

7.33 Upstream oxygen sensor heater management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LSHPWM_UP[NC_CBK_EX_NR]	V/O	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle, acquired also by BSW					
V_EFC_LSH_UP[NC_CBK_EX_NR]	V/0	0 ... FFH	0 ... 26	0.102	V
Effective final output value of heater voltage					
V_EFC_CLC_LSH_UP[NC_CBK_EX_NR]	V	0 ... FFFFH	0 ... 26	396.73e-6	V
Effective calculated heater voltage value prior to overvoltage protection					
V_EFC_CTL_ADD_LSH_UP[NC_CBK_EX_NR]	V	80 ... 7FH	-13.05 ... 12.95	0.102	V
Additive effective heater voltage from closed loop controller					
V_EFC_CTL_P_LSH_UP[NC_CBK_EX_NR]	V	8000 ... 7FFFH	-13.00 ... 13.00	396.73e-6	V
Additive effective heater voltage from closed loop controller, P term					
V_EFC_CTL_I_LSH_UP[NC_CBK_EX_NR]	V	8000 ... 7FFFH	-13.00 ... 13.00	396.73e-6	V
Additive effective heater voltage from closed loop controller, I term					
V_EFC_ADD_LSH_UP_REST	V	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage offset in case of active LSH_REST and low R_IT.					
TTIP_MES_DIF_LS_UP[NC_CBK_EX_NR]	V	8000 ... 7FFFH	-2048...2047.94	62.5e-3	°C
Difference between actual and set oxygen sensor tip temperature					
T_POW_RISE_LSH_UP[NC_CBK_EX_NR]	V/O	0 ... FFFFH	0 ... 6553.5	0.1	s
Timer indicating the duration of the pre-heating and post dew point heating phases					
T_V_EFC_TOL_LSH_UP[NC_CBK_EX_NR]	V	0 ... FFH	0 ... 25.5	0.1	s
Timer indicating permitted effective heater voltage overvoltage duration					
T_V_EFC_LIM_LSH_UP[NC_CBK_EX_NR]	V	0 ... FFH	0 ... 25.5	0.1	s
Timer indicating the duration of the effective heater voltage limitation preventing overvoltage					
LV_LSH_CTL_CLL_LSH_UP[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean variable indicating closed loop control mode active					
LV_V_EFC_LIM_BOL_LSH_UP[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to sum of voltage components falling below zero.					
LV_V_EFC_LIM_TOL_LSH_UP[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to exceeding threshold for period of time.					
LV_V_EFC_LIM_MAX_LSH_UP[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to exceeding absolute maximum voltage spec.					
LV_V_EFC_LIM_PROT_VB_LSH_UP[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean variable indicating effective heater voltage limited due to excessive battery voltage level					
LV_LSH_UP_REST[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean variable indicating that the preheating has been interrupted by the restart detection function.					
STATE_LSH_UP[NC_CBK_EX_NR]	V/O	00H 01H 02H 03H 04H 05H 06H	LSH_OFF LSH_POW_RISE LSH_POW_RED LSH_POW_FALL LSH_POW_CTL LSH_VB_PROT LSH_TEMP_PROT	-	-
Present heater state					

Input data:

LV_INH_LSH_UP[NC_CBK_EX_NR]	LV_LS_UP_READY_CDN[NC_CBK_EX_NR]	TEMP_INI_LS_UP[NC_CBK_EX_NR]	LV_INH_LSH_CTL_CLL_LSH_UP[NC_CBK_EX_NR]
LV_ST_END	VS	TTIP_MES_LS_UP[NC_CBK_EX_NR]	LV_TEG_MIN_THD[NC_CBK_EX_NR]
VB	N_32	TEG_CAT_UP_MDL[NC_CBK_EX_NR]	LV_TTIP_MES_VLD_LS_UP[NC_CBK_EX_NR]
NC_CBK_EX_NR	T_AST	LSHPWM_EXT_LS_UP[NC_CBK_EX_NR]	LV_V_REF_VLD_R_IT_LS_UP[NC_CBK_EX_NR]
C_VLS_UP_AFR_THD_READY	C_VLS_UP_AFL_THD_READY	TCO_STOP	TCO_ST
VLS_UP	R_IT_LS_UP	CTR_CYCNR_R_IT_LS_UP_VLD	

FUNCTION DESCRIPTION:

General information:

If two separate cylinder banks are concerned, then
 i = 1, for cylinder bank 1
 i = 2, for cylinder bank 2

The goal of the heater control shall be to control the oxygen sensor heating such that the optimal operating temperature is reached in the shortest time possible. At the same time, the maximum permissible temperature gradient (possible damage to ceramic due to thermal stress), the possible occurrence of water splash under the dew point (possible damage to ceramic due to thermal shock) and the absolute maximum ratings specified for the sensor shall be taken into account.

The function shall permit oxygen sensor heating under open and closed loop control.

Signal flow:

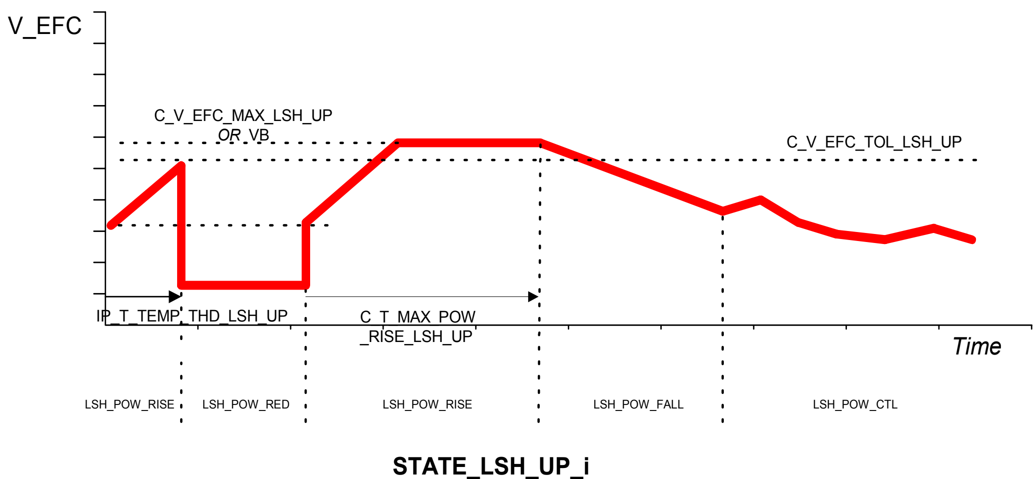


Figure 1: Course of effective heater voltage over time / state

Description:

The heater control shall be realised by use of a state machine. Each upstream oxygen sensor shall have its own state machine and each state machine shall run independently of the other.

The heater control shall be initialised to the LSH_OFF state at the start of a new driving cycle. Hence the oxygen sensor will not initially be electrically heated.

Heating shall commence should the activation conditions listed below be met. i.e. when the engine has transitioned from the start state (LV_ST_END = 1) and the heater management inhibit bit for the corresponding bank is not set (LV_INH_LSH_UP[i] = 0). The appropriate heater state shall change to LSH_POW_RISE and the initial value for the effective heater voltage shall be determined by obtaining the oxygen sensor temperature at start (TEMP_INI_LS_UP[i]) from input data. This shall then be used to obtain the appropriate effective heater starting voltage from map IP_V_EFC_INI_LSH_UP.

Should the state LSH_POW_RISE is switched from LSH_VB_PROT or LSH_TEMP_PROT, the effective heater voltage shall be incremented with the gradient of

C_V_EFC_INC_PORT_LSH_DOWN. In the other cases, the effective heater voltage shall be incremented with the gradient of C_V_EFC_INC_LSH_DOWN. In order to prevent sensor from over heating when the sensor is already warm, C_V_EFC_INC_PORT_LSH_DOWN has to be set samller than C_V_EFC_INC_LSH_DOWN.

The continuation of the heating gradient shall depend on whether the dew point at the sensor location has been determined to have been exceeded, i.e. the temperature of the inner manifold wall at the sensor location has exceeded the temperature threshold where water in the exhaust gas may condense, as determined by LV_TEG_MIN_THD[i].

Should the dew point not have been exceeded (LV_TEG_MIN_THD[i] = 0) but timer T_POW_RISE_LSH_UP[i] has equalled or exceeded IP_T_TEMP_THD_LSH_UP, it shall be assumed that condensed water splash at the oxygen sensor location may occur. In order to protect the oxygen sensor by keeping the temperature of the ceramic lower than the specified critical temperature, the heater power shall be reduced immediately by setting the effective heater voltage to C_V_EFC_RED_LSH_UP. In this case, the heater state shall be set to LSH_POW_RED.

The reduced heating state shall persist until the dew point in the exhaust gas system has been determined to have been exceeded, i.e. until LV_TEG_MIN_THD[i] = 1, at which point the heater state shall return to LSH_POW_RISE. The effective heater voltage shall be set to the same effective heater voltage as determined at start (IP_V_EFC_INI_LSH_UP), which may be modified by the calibration data C_V_EFC_STEP_LSH_UP, may be further incremented with a gradient of C_V_EFC_INC_LSH_UP without danger of damaging the oxygen sensor and the timer T_POW_RISE_LSH_UP[i] shall be reinitialised with the IP_T_TEMP_THD_LSH_UP determined at function start.

Should the initial exhaust gas temperature at start have been determined to exceed the dew point (LV_TEG_MIN_THD[i] = 1), the heater control shall not enter the reduced heater power state but continue to incrementally increase the heater power within the LSH_POW_RISE state.

Dependent on whether closed loop control is permitted, the heater management shall enter either LSH_POW_CTL or LSH_POW_FALL. Should closed loop be permitted (LV_INH_LSH_CTL_CLL_LSH_UP[i] = 0) and the ceramic temperature reference be valid (LV_V_REF_VLD_R_IT_LS_UP[i] = 1) and ceramic temperature exceed the set temperature (TTIP_MES_LS_UP[i] ≥ C_TTIP_SP_LS_UP[i]) then the state machine shall change to LSH_POW_CTL and the timer shall be frozen. This state change shall permit the controller to take over heating when the the operative readiness has been detected in order to prevent under- & overshoot of the ceramic temperature. Should the conditional operative readiness be set (LV_LS_UP_READY_CDN[i] = 1) then the state machine shall also enter state LSH_POW_CTL and stop the timer. This shall permit the heater stability diagnosis to be carried out even when the heater is sufficiently defective to prevent the set temperature from being reached. Should the closed loop controller be inhibited or the ceramic temperature calibration be faulty, and T_POW_RISE_LSH_UP[i] be determined to have equalled or exceeded the value C_T_MAX_POW_RISE_LSH_UP (irrespective of the dew point having been exceeded or not), the heater state shall change to LSH_POW_FALL and the timer shall be frozen.

If there has been a hot sensor restart i.e. TCO_ST >= (TCO_STOP - C_TCO_DIF_MAX_LSH_UP_REST) and VLS_UP moves outside the thresholds for readiness detection i.e. (VLS_UP > C_VLS_UP_AFR_THD_READY or VLS_UP < C_VLS_UP_AFL_THD_READY) the heater management shall leave "LSH_POW_RISE" and enter "LSH_POW_CTL" and the bit LV_LSH_UP_REST shall be set. This condition prevents sensors with powerful heaters (e.g. NTK UFLO 3.7 Ohm) and no closed loop heater capability from overheating during a hot start or repeated engine stop and start following a cold start. The constant C_TCO_ST_MIN_LSH_UP_REST allows this condition to be activated above a certain TCO_ST or to be deactivated completely for WRAF sensors / sensors with greater thermal inertia / weaker heater.

If LV_LSH_UP_REST = 1 the first C_CTR_R_IT_LS_UP_VLD_REST valid internal resistance values are checked and the effective heater voltage is offset by C_V_EFC_ADD_LSH_UP_REST whenever R_IT_LS_UP is less than C_R_IT_LS_UP_MIN_REST.

In the LSH_POW_FALL state, the effective heater voltage shall be reduced with a gradient of C_V_EFC_DEC_LSH_UP until such time that it falls below or equals the value set for the open loop control at the current engine operating point. For cylinder bank 1, this value is determined by map IP_V_EFC_CTL_LSH_UP alone. For cylinder bank 2, this value is determined by IP_V_EFC_CTL_LSH_UP * IP_FAC_V_EFC_TEG_LSH_UP. The latter factor shall be necessary to accommodate differences in heat transfer rates between cylinder banks at the oxygen sensor location, particularly for transversal mounted V6-engines. At such time the effective heater voltage falls below or equals the respective threshold, the heater state shall change to LSH_POW_CTL.

Should, however, the Boolean flag LV_TEG_MIN_THD[i] be reset whilst in the state LSH_POW_FALL, then the closed loop controller terms shall be reset, the Boolean flags to indicate closed loop control and effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RED following the procedures outlined above.

The control of the effective heater voltage may be carried out in open loop or closed loop mode. Unless either the closed loop control inhibit bit (LV_INH_LSH_CTL_CLL_LSH_UP[i]) is set or the reference voltage validity bit (LV_V_REF_VLD_R_IT_LS_UP[i]) is reset, the effective heater voltage shall be controlled in closed loop mode. Closed loop control conditions met shall be indicated by setting LV_LSH_CTL_CLL_LSH_UP[i]. Should at any time one of the above conditions no longer be met, the heater controller shall revert to open loop control, reset the closed loop variables and reset LV_LSH_CTL_CLL_LSH_UP[i].

In the open loop heater control mode, the effective heater voltage shall be obtained from map IP_V_EFC_CTL_LSH_UP and is dependent upon the engine speed and the modelled exhaust gas temperature. For cylinder bank 2, an additional multiplicative factor (IP_FAC_V_EFC_TEG_LSH_UP) may be required for the reasons outlined above.

In the closed loop heater control mode, a PI controller, that affects the open loop effective heater voltage in an additive manner, shall regulate the effective heater voltage such that the temperature specified by C_TTIP_SP_LS_UP[i] is maintained. The closed loop control may be further described as follows:

The current temperature of the oxygen sensor (TTIP_MES_LS_UP[i]) shall be obtained and the difference between this temperature and the set temperature computed to obtain the input variable to the controller (TTIP_MES_DIF_LS_UP[i]). Note that this variable shall be computed in both open and closed loop control modes but shall only be used in the closed loop control mode.

The controller shall then determine the appropriate additive voltage (V_EFC_CTL_ADD_LSH_UP[i]) by computing the P and I terms separately and then adding the two terms.

The P term (V_EFC_CTL_P_LSH_UP[i]) shall calculate a voltage proportional to the temperature difference and thus converting a temperature delta into voltage. The P term shall only be calculated when the absolute temperature difference (TTIP_MES_DIF_LS_UP[i]) equals or exceeds a constant (C_TTIP_DIF_MIN_CTL_P_LS_UP). This shall be achieved via the map IP_FAC_xxx_P_LSH_UP_CTL, where xxx stands for either POS or NEG dependent on the sign of the temperature difference. The map values are dependent on engine speed and exhaust gas temperature. A further dimensionless constant C_FAC_CTL_P_LSH_UP shall permit global parameter changes to be made.

The I term (V_EFC_CTL_I_LSH_UP[i]) shall calculate an integral of a voltage that shall be proportional to the temperature difference and thus converting a temperature delta into voltage. This shall be achieved via the map IP_FAC_xxx_I_LSH_UP_CTL, where xxx stands for either POS or NEG dependent on the sign of the temperature difference. The map values are dependent on engine speed and exhaust gas temperature. A further dimensionless constant C_FAC_CTL_I_LSH_UP shall permit global parameter changes to be made.

Once the additive voltage has been determined, this shall then be added to the open loop control voltage along with a calibration system constant (C_V_EFC_AS_LSH_UP) to permit a global change in the total effective heater voltage to be made. Thus the calculated effective heater voltage (V_EFC_CLC_LSH_UP[i]) shall be computed from the open loop value, the closed loop additive term and the global correction value.

The closed loop control computations shall be suspended should either the Boolean flag LC_LSH_CTL_CLL_INH_LSH_UP[i] be set or the ceramic temperature no longer be valid, for example due to an active reference voltage measurement, as indicated by LV_TTIP_MES_VLD_LS_UP[i] being reset. This shall freeze the P and I terms at their last calculated value. Upon revoking the suspension, the controller shall continue to compute from where the point immediately prior to the suspension, i.e. shall not start anew. Furthermore should the sum of the components used to calculate the effective heater voltage be negative, LV_V_EFC_LIM_BOL_LSH_UP[i] shall be set. This shall prevent the controller from decreasing the effective heater voltage further but not prevent it from increasing the voltage. When the sum of the components is no longer negative, LV_V_EFC_LIM_BOL_LSH_UP[i] shall be reset. Similarly, should either of the overvoltage limitation Boolean flags LV_V_EFC_LIM_TOL_LSH_UP[i] & LV_V_EFC_LIM_MAX_LSH_UP[i] be set, as described later, the controller shall be prevented from increasing the effective heater voltage further but not prevent it from decreasing the voltage. When the overvoltage flags are no longer set and the other applicable conditions are met, the controller shall resume normal operation.

Should, however, the Boolean flag LV_TEG_MIN_THD[i] be reset whilst in the state LSH_POW_CTL, then the closed loop controller terms shall be reset, the Boolean flags to indicate closed loop control and effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RED following the procedures outlined above.

Should at any time the battery voltage VB exceed the threshold C_VB_MAX_PROT_LSH_UP, as determined by the state of logical variable LV_V_EFC_LIM_PROT_VB_LSH_UP[i], the oxygen sensor heater power shall be reduced by reducing the effective heater voltage used to calculate the driver duty cycle to C_V_EFC_PROT_VB_LSH_UP. The heater state shall change to LSH_VB_PROT on the next recurrence of the state machine and the effective heater voltage set to the aforementioned limit. This shall prevent damage to the sensor due to excessive heater voltages e.g. due to jump start from 24 V.

Should the battery voltage recover when in the state LSH_VB_PROT, as indicated by LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0, T_POW_RISE_LSH_UP[i] shall be reset, V_EFC_LSH_UP[i] shall be initialised to IP_V_EFC_INI_LSH_UP calculated after the start, the closed loop controller terms shall be reset, the Boolean flags to indicate closed loop control and effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RISE following the procedures outlined above.

Short term voltage excursions shall cause the heater power to be limited but will only cause the state machine to change state should the voltage excursion persist until the next recurrence of the state machine. The state machine shall not be able to enter the

LSH_VB_PROT state directly from LSH_OFF but shall be required to pass through LSH_POW_RISE.

The sensor shall also be protected from over-temperature. This shall be achieved by evaluating the value of the modelled exhaust gas temperature TEG_CAT_UP_MDL[i] in all states other than LSH_OFF. Should the exhaust gas temperature exceed the threshold C_TEG_CAT_UP_MDL_MAX_LSH_UP, the calculated effective heater voltage shall be reduced to 0, the state shall change to LSH_TEMP_PROT thus suspending closed loop control, if active.

Should the exhaust gas temperature fall below the threshold, the state shall change dependent on the state active immediately prior to the over-temperature protection. If the state was LSH_POW_CTL, the state machine shall return to this state and the effective heater voltage shall take the new value as computed by the open / closed loop control. The closed loop controller shall not be started anew. If the state was any other than LSH_POW_CTL, the state shall change to LSH_POW_RISE. In this case T_POW_RISE_LSH_UP[i] shall be reset, V_EFC_LSH_UP[i] shall be initialised to IP_V_EFC_INI_LSH_UP calculated after the start, the closed loop controller terms shall be reset, the Boolean flags to indicate closed loop control and effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RISE following the procedures outlined above.

Furthermore, in order to protect the sensor heater against long term high voltage stress it shall be necessary to limit the effective heater voltage should the value of V_EFC_CLC_LSH_UP[i] equal or exceed the threshold C_V_EFC_TOL_LSH_UP.

In this instance, the timer T_V_EFC_TOL_LSH_UP[i] shall be started. Should the overvoltage condition persist, such that this timer equals or exceeds the threshold C_T_MAX_V_EFC_TOL_LSH_UP, timer T_V_EFC_LIM_LSH_UP[i] shall be started and the Boolean flag LV_V_EFC_LIM_TOL_LSH_UP[i] shall be set. This flag shall prevent the controller from increasing effective the voltage further. If however the value of V_EFC_CLC_LSH_UP[i] should fall below the threshold C_V_EFC_TOL_LSH_UP at any time, LV_V_EFC_LIM_TOL_LSH_UP[i] shall be reset.

Once timer T_V_EFC_LIM_LSH_UP[i] has been started, Boolean flag LV_V_EFC_LIM_TOL_LSH_UP[i] shall be set or reset depending on the value of V_EFC_CLC_LSH_UP[i]. Once the timer T_V_EFC_LIM_LSH_UP[i] equals or exceeds threshold C_T_MAX_V_EFC_LIM_LSH_UP, both timers shall be reset and timer T_V_EFC_TOL_LSH_UP[i] must exceed the threshold C_T_MAX_V_EFC_TOL_LSH_UP again in order to permit further limiting to take place.

To prevent damage to the sensor, V_EFC_CLC_LSH_UP[i] shall not be permitted to exceed the lower of the values C_V_EFC_MAX_LSH_UP, C_V_EFC_MAX_ST_LSH_UP or VB dependent on the value of T_POW_RISE_LSH_UP[i].

If T_POW_RISE_LSH_UP[i] equals or exceeds the constant C_T_MAX_V_EFC_MAX_ST_LSH_UP STEP or conditions regarding T_AST and STATE_LSH_UP[i] then V_EFC_CLC_LSH_UP[i] shall be checked against VB & C_V_EFC_MAX_LSH_UP. If either of the values is exceeded then Boolean flag LV_V_EFC_LIM_MAX_LSH_UP[i] shall be set and V_EFC_CLC_LSH_UP[i] limited to the lower of VB & C_V_EFC_MAX_LSH_UP otherwise LV_V_EFC_LIM_MAX_LSH_UP[i] shall be reset.

If T_POW_RISE_LSH_UP[i] is lower then constant C_T_MAX_V_EFC_MAX_ST_LSH_UP then V_EFC_CLC_LSH_UP[i] shall be checked against VB & C_V_EFC_MAX_ST_LSH_UP. If either of the values is exceeded then Boolean flag LV_V_EFC_LIM_MAX_LSH_UP[i] shall be set and V_EFC_CLC_LSH_UP[i] limited to the lower of VB & C_V_EFC_MAX_ST_LSH_UP otherwise LV_V_EFC_LIM_MAX_LSH_UP[i] shall be reset.

The flag LV_V_EFC_LIM_MAX_LSH_UP[i] shall prevent the controller from increasing effective the voltage further and permit the final effective heater voltage to be limited, as described later.

Should timer T_V_EFC_LIM_LSH_UP[i] be unequal to zero, i.e. overvoltage no longer tolerated as it has been present for certain time, V_EFC_LSH_UP[i] shall be the lower of the values V_EFC_CLC_LSH_UP[i] & C_V_EFC_LIM_LSH_UP otherwise V_EFC_LSH_UP[i] shall be set to V_EFC_CLC_LSH_UP[i].

Once the effective heater voltage has been determined throughout the various heater states other than LSH_OFF, it shall be used to compute the PWM duty cycle (LSHPWM_UP[i]) that shall control the appropriate heater driver. The heater power shall be corrected to take into account deviations in the measured battery voltage and limited to the range bounded by NC_LSHPWM_TOL_LSH_UP & NC_LSHPWM_BOL_LSH_UP. The evaluation of excess battery voltage shall be determined at the same recurrence as the computation of the duty cycle as described above. The PWM value may also be modified by use of constant C_FAC_V_EFC_AS_LSH_UP.

Should at any time an inhibit flag be set (LV_INH_LSH_UP[i] = 1), the corresponding oxygen sensor heater shall be switched off and the STATE_LSH_UP[i] set to LSH_OFF. This shall allow project specific application conditions to turn off oxygen sensor heating according to project philosophy. Should the heater function be inhibited, then the PWM duty cycle shall be set to input LSHPWM_EXT_LS_UP[i]. On a transition of the inhibit bit LV_INH_LSH_UP[i] from 1 $\rightarrow$ 0, the heater function shall start from anew.

Application conditions:

Activation:

At any engine operating state.

Deactivation:

-

Time recurrence:

The state machine, sensor protection and effective heater voltage definition shall be carried out once every 100 ms.

The determination of the oxygen sensor heater driver duty cycle and associated excessive battery voltage protection shall be carried out once every 10 ms.

Initialisation:

The following variable initialisation shall take place at the beginning of a new driving cycle:

STATE_LSH_UP[i] = LSH_OFF
T_POW_RISE_LSH_UP[i] = 0
T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
LSHPWM_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
V_EFC_LSH_UP[i] = 0
V_EFC_CLC_LSH_UP[i] = 0
V_EFC_ADD_LSH_UP_REST = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0

LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0
LV_LSH_UP_REST = 0
TTIP_MES_DIF_LS_UP[i] = 0

NOTE: Projects not making use of the inhibit flags LV_INH_LSH_UP[i] & LV_INH_LSH_CTL_CLL_LSH_UP[i] shall ensure that they are initialised with and remain 0, thus never disabling the oxygen sensor heater function or thus never disabling the oxygen sensor heater closed loop control function respectively

Formula section:

7.33.1 Heater management state machine

The state machine shall remain in its current state and carry out the actions specified to occur within that state once per recurrence unless otherwise specified. The state machine shall only move to another state when one of the conditions has been determined to be met.

Note: The priorities of the conditions to change between states shall be defined by the order in which these conditions are listed within the appropriate state as described below.

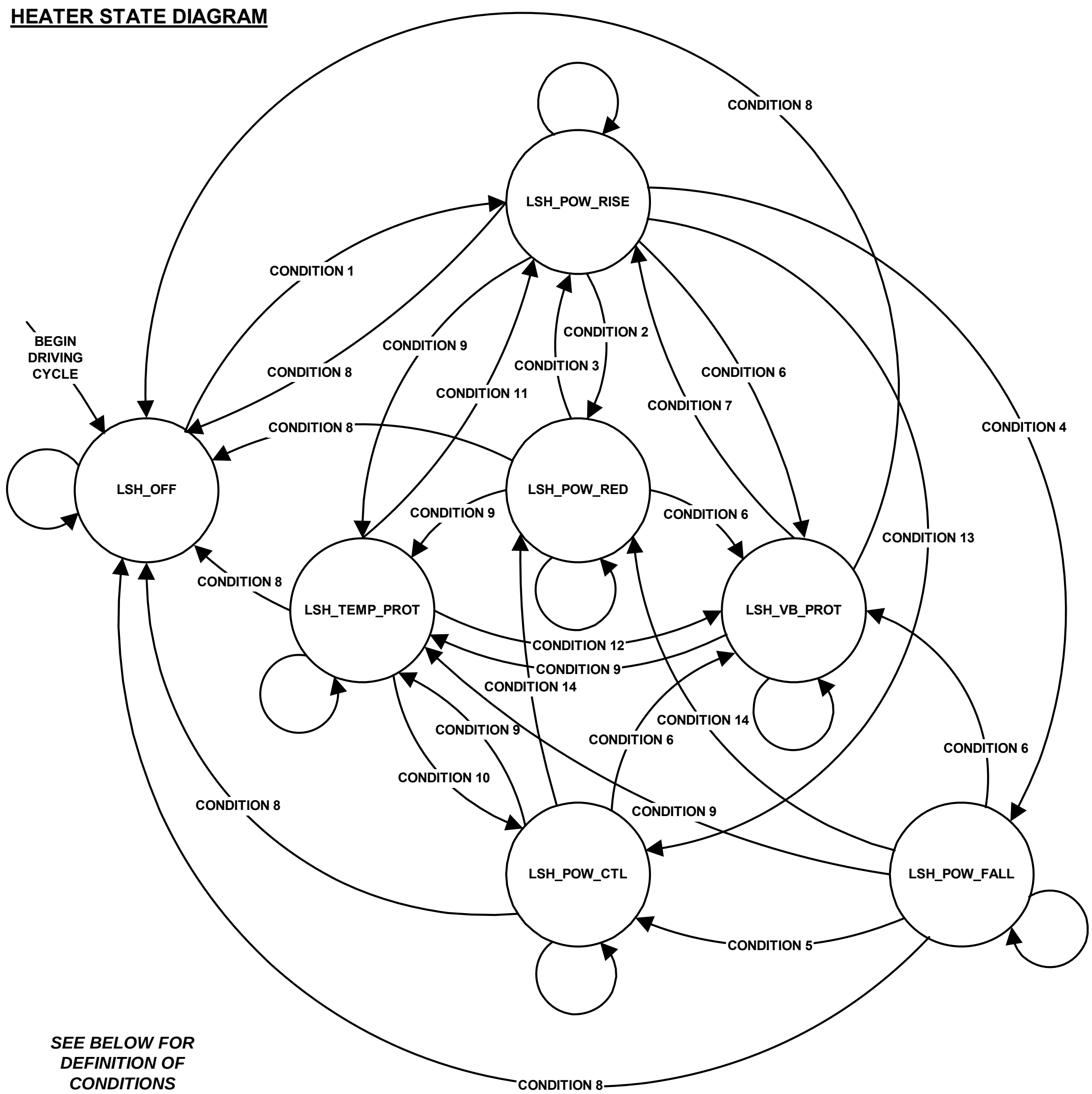


Figure 2: Heater management state diagram

STATE_LSH_UP[i] "LSH_OFF"

Actions:

T_POW_RISE_LSH_UP[i] = 0
T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
V_EFC_LSH_UP[i] = 0
V_EFC_CLC_LSH_UP[i] = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0
TTIP_MES_DIF_LS_UP[i] = 0

Note: The above listed actions must be carried out at least once when the state has been entered for the first time or from any other state!

Condition 1:"LSH_OFF to LSH_POW_RISE"

LV_ST_END = 1 **&**
LV_INH_LSH_UP[i] = 0

Transition actions:

V_EFC_CLC_LSH_UP[i] = IP_V_EFC_INI_LSH_UP (TEMP_INI_LS_UP[i])
Timer T_POW_RISE_LSH_UP[i] shall be started
STATE_LSH_UP[i] = LSH_POW_RISE

STATE_LSH_UP[i] "LSH_POW_RISE"

Actions:

If Previous state of STATE_LSH_UP[i] = LSH_VB_PROT or LSH_TEMP_PROT

Then $V\_EFC\_CLC\_LSH\_UP[i]_N = V\_EFC\_CLC\_LSH\_UP[i]_{N-1}$ $+ C\_V\_EFC\_INC\_PROT\_LSH\_UP * 100$ ms

Else $V\_EFC\_CLC\_LSH\_UP[i]_N = V\_EFC\_CLC\_LSH\_UP[i]_{N-1}$

$+ C\_V\_EFC\_INC\_LSH\_UP * 100$ ms

Endif

Note: See section "General oxygen sensor voltage protection" for notes on limiting the applied heater voltage.

Condition 8:"LSH_POW_RISE to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

Condition 9:"LSH_POW_RISE to LSH_TEMP_PROT"

TEG_CAT_UP_MDL[i] > C_TEG_CAT_UP_MDL_MAX_LSH_UP **&**
LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_TEMP_PROT

Condition 6:"LSH_POW_RISE to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_VB_PROT

Condition 2:"LSH_POW_RISE to LSH_POW_RED"

T_POW_RISE_LSH_UP[i] ≥ IP_T_TEMP_THD_LSH_UP (TEMP_INI_LS_UP[i]) **&**
LV_TEG_MIN_THD[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_POW_RED

Condition 13:"LSH_POW_RISE to LSH_POW_CTL"

(LV_INH_LSH_CTL_CLL_LSH_UP[i] = 0 **&**
LV_V_REF_VLD_R_IT_LS_UP[i] = 1 **&**
TTIP_MES_LS_UP[i] ≥ C_TTIP_SP_LS_UP[i] - ID_TTIP_SP_DIF_LSH_UP **&**
LV_TTIP_MES_VLD_LS_UP[i] = 1) **or**
LV_LS_UP_READY_CDN[i] = 1 **or**
(TCO_ST > C_TCO_ST_MIN_LSH_UP_REST **&**
TCO_ST >= TCO_STOP - C_TCO_DIF_MAX_LSH_UP_REST **&**
(VLS_UP > C_VLS_UP_AFR_THD_READY **or** VLS_UP < C_VLS_UP_AFL_THD_READY))
Transition actions:

Stop timer T_POW_RISE_LSH_UP[i] and freeze value

If (TCO_ST > C_TCO_ST_MIN_LSH_UP_REST **&**
TCO_ST >= TCO_STOP - C_TCO_DIF_MAX_LSH_UP_REST **&**
(VLS_UP > C_VLS_UP_AFR_THD_READY **or** VLS_UP < C_VLS_UP_AFL_THD_READY))
then LV_LSH_UP_REST = 1 *%bit set if transition to "LSH_POW_CTL" caused by LSH_UP_REST conditions.*
STATE_LSH_UP[i] = LSH_POW_CTL

Condition 4:"LSH_POW_RISE to LSH_POW_FALL"

(LV_INH_LSH_CTL_CLL_LSH_UP[i] = 1 **or**
LV_V_REF_VLD_R_IT_LS_UP[i] = 0) **&**
T_POW_RISE_LSH_UP[i] ≥ C_T_MAX_POW_RISE_LSH_UP +
IP_T_TEMP_THD_LSH_UP (TEMP_INI_LS_UP[i])

Transition actions:

Stop timer T_POW_RISE_LSH_UP[i] and freeze value
STATE_LSH_UP[i] = LSH_POW_FALL

STATE_LSH_UP[i] "LSH_POW_RED"

Actions:

V_EFC_CLC_LSH_UP[i] = C_V_EFC_RED_LSH_UP

<u>Condition 8</u>:*"LSH_POW_RED to LSH_OFF"*

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

<u>Condition 9</u>:*"LSH_POW_RED to LSH_TEMP_PROT"*

TEG_CAT_UP_MDL[i] > C_TEG_CAT_UP_MDL_MAX_LSH_UP **&** LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_TEMP_PROT

<u>Condition 6</u>:*"LSH_POW_RED to LSH_VB_PROT"*

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_VB_PROT

<u>Condition 3</u>:*"LSH_POW_RED to LSH_POW_RISE"*

LV_TEG_MIN_THD[i] = 1

Transition actions:

V_EFC_CLC_LSH_UP[i] = IP_V_EFC_INI_LSH_UP (TEMP_INI_LS_UP[i]) *(same value as at start)* + C_V_EFC_STEP_LSH_UP
Re-initialise timer T_POW_RISE_LSH_UP[i] with IP_T_TEMP_THD_LSH_UP (TEMP_INI_LS_UP[i]) *(same value as at start)*
STATE_LSH_UP[i] = LSH_POW_RISE

Note: See section "General oxygen sensor voltage protection" for notes on limiting the applied heater voltage. The function shall prevent the effective heater voltage from becoming negative by limiting to 0 where necessary.

STATE_LSH_UP[i] "LSH_POW_FALL"

Actions:

$V\_EFC\_CLC\_LSH\_UP[i]_{N} = V\_EFC\_CLC\_LSH\_UP[i]_{N-1} - C\_V\_EFC\_DEC\_LSH\_UP * 100$ ms

Note: See section "General oxygen sensor voltage protection" for notes on limiting the applied heater voltage. The function shall prevent the effective heater voltage from becoming negative by limiting to 0 where necessary.

<u>Condition 8</u>:*"LSH_POW_FALL to LSH_OFF"*

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

<u>Condition 9</u>:*"LSH_POW_FALL to LSH_TEMP_PROT"*

TEG_CAT_UP_MDL[i] > C_TEG_CAT_UP_MDL_MAX_LSH_UP **&** LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_TEMP_PROT

<u>Condition 6</u>:*"LSH_POW_FALL to LSH_VB_PROT"*

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_VB_PROT

<u>Condition 14</u>:"LSH_POW_FALL to LSH_POW_RED"

LV_TEG_MIN_THD[i] = 0

Transition actions:

T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
TTIP_MES_DIF_LS_UP[i] = 0
STATE_LSH_UP[i] = LSH_POW_RED

<u>Condition 5</u>:"LSH_POW_FALL to LSH_POW_CTL"

For Bank 1; i = 1

V_EFC_LSH_UP[i] ≤ IP_V_EFC_CTL_LSH_UP (N_32, TEG_CAT_UP_MDL[i])

For Bank 2; i = 2

V_EFC_LSH_UP[i] ≤ IP_V_EFC_CTL_LSH_UP (N_32, TEG_CAT_UP_MDL[i]) *
IP_FAC_V_EFC_TEG_LSH_UP

Transition actions:

STATE_LSH_UP[i] = LSH_POW_CTL

STATE_LSH_UP[i] "LSH_POW_CTL"

Actions:

If
LV_LSH_UP_REST = 1 **and**
CTR_CYCNR_R_IT_LS_UP_VLD < C_CTR_R_IT_LS_UP_VLD_REST **then**
    If
    R_IT_LS_UP < C_R_IT_LS_UP_MIN_REST
        Then
        V_EFC_ADD_LSH_UP_REST = C_V_EFC_ADD_LSH_UP_REST
        Else
        V_EFC_ADD_LSH_UP_REST = 0
    Endif
Else
V_EFC_ADD_LSH_UP_REST = 0
Endif

Determine deviation between desired operating temperature and measured temperature:

TTIP_MES_DIF_LS_UP[i] = C_TTIP_SP_LS_UP[i] - TTIP_MES_LS_UP[i]

Note: TTIP_MES_DIF_LS_UP[i] shall be computed in both open loop and closed loop control modes.

Determine whether closed loop control permitted, if not revert to open loop control:

If (LV_INH_LSH_CTL_CLL_LSH_UP[i] = 0) **&**
(LV_V_REF_VLD_R_IT_LS_UP[i] = 1)

then ***Closed loop control:***

LV_LSH_CTL_CLL_LSH_UP[i] = 1
Case A*: Closed loop control*

else ***Open loop control:***

LV_LSH_CTL_CLL_LSH_UP[i] = 0

PI controller shall be re-initialised, i.e.:
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0

endif.

Case A*: Closed loop control:*

Determine conditions for computation of closed loop control variables:

If (LC_LSH_CTL_CLL_INH_LSH_UP[i] = 1) **or**
(LV_TTIP_MES_VLD_LS_UP[i] = 0)

then ***General suspension of closed loop control computation, i.e.:***

V_EFC_CTL_ADD_LSH_UP$[i]_N$ = V_EFC_CTL_ADD_LSH_UP$[i]_{N-1}$
V_EFC_CTL_P_LSH_UP$[i]_N$ = V_EFC_CTL_P_LSH_UP$[i]_{N-1}$
V_EFC_CTL_I_LSH_UP$[i]_N$ = V_EFC_CTL_I_LSH_UP$[i]_{N-1}$

else ***Normal closed loop control computation:***

Case B*: Closed loop control & Not generally suspended:*

endif.

End Case A*: Closed loop control:*

Case B*: Closed loop control & Not generally suspended:*

If (TTIP_MES_LS_UP[i] > C_TTIP_SP_LS_UP[i]) **&** *i.e. Temperature difference negative*
(LV_V_EFC_LIM_BOL_LSH_UP[i] = 0) *i.e. no lower limiting active*

then Compute I term of controller:
V_EFC_CTL_I_LSH_UP$[i]_N$ = V_EFC_CTL_I_LSH_UP$[i]_{N-1}$ +
TTIP_MES_DIF_LS_UP[i] *
IP_FAC_NEG_CTL_I_LSH_UP (N_32, TEG_CAT_UP_MDL[i])*
C_FAC_CTL_I_LSH_UP

If (**|** TTIP_MES_DIF_LS_UP[i] **|** ≥ C_TTIP_DIF_MIN_CTL_P_LS_UP)

then Compute P term of controller:
V_EFC_CTL_P_LSH_UP[i] = TTIP_MES_DIF_LS_UP[i] *
IP_FAC_NEG_CTL_P_LSH_UP (N_32, TEG_CAT_UP_MDL[i])*
C_FAC_CTL_P_LSH_UP

else

V_EFC_CTL_P_LSH_UP[i] = 0

endif

else If (TTIP_MES_LS_UP[i] ≤ C_TTIP_SP_LS_UP[i]) **&**
i.e. Temperature difference positive or zero;
(LV_V_EFC_LIM_MAX_LSH_UP[i] = 0) **&**
(LV_V_EFC_LIM_TOL_LSH_UP[i] = 0) *i.e. no upper limiting active*

then Compute I term of controller:
$V\_EFC\_CTL\_I\_LSH\_UP[i]_N$ = $V\_EFC\_CTL\_I\_LSH\_UP[i]_{N-1}$ +
TTIP_MES_DIF_LS_UP[i] *
IP_FAC_POS_CTL_I_LSH_UP (N_32, TEG_CAT_UP_MDL[i])*
C_FAC_CTL_I_LSH_UP

If (**|** TTIP_MES_DIF_LS_UP[i] **|** ≥ C_TTIP_DIF_MIN_CTL_P_LS_UP)

then Compute P term of controller:
V_EFC_CTL_P_LSH_UP[i] = TTIP_MES_DIF_LS_UP[i] *
IP_FAC_POS_CTL_P_LSH_UP (N_32, TEG_CAT_UP_MDL[i])*
C_FAC_CTL_P_LSH_UP

else

V_EFC_CTL_P_LSH_UP[i] = 0

endif

endif

endif.

Note: In cases where the effective heater voltage has been limited as shown by the state of the Boolean flag LV_V_EFC_LIM_BOL_LSH_UP[i], the controller shall only be permitted to increase the effective voltage. In the case of limiting via Boolean flags LV_V_EFC_LIM_TOL_LSH_UP[i] & LV_V_EFC_LIM_MAX_LSH_UP[i] the closed loop control shall only be permitted to decrease the effective heater voltage. In all other cases, the controller shall effectively be suspended, i.e. P & I terms shall remain unchanged but applied, until the applicable limitation is revoked. Upon revoking the limitation, the function shall resume to compute the P & I terms according to the conditions specified above.

Combine P and I terms:
V_EFC_CTL_ADD_LSH_UP[i] =
V_EFC_CTL_P_LSH_UP[i] + V_EFC_CTL_I_LSH_UP[i]

Note: Although variables V_EFC_CTL_P_LSH_UP[i] and V_EFC_CTL_I_LSH_UP[i] are defined as words, the summation shall be carried out as a 16 bit addition and the resultant 16 bit value converted to an 8 bit value, by taking the high byte, and placed in V_EFC_CTL_ADD_LSH_UP[i]. This shall prevent controller from stopping short of the target temperature.

End Case B: *Closed loop control; Not suspended:*

Compute effective heater voltage from open loop control and closed loop delta voltages:

For Bank 1; i = 1

V_EFC_LSH_UP_CLC[i] = IP_V_EFC_CTL_LSH_UP (N_32, TEG_CAT_UP_MDL[i]) +
V_EFC_LSH_UP_CTL_ADD[i] + C_V_EFC_AS_LSH_UP + V_EFC_ADD_LSH_UP_REST

Note: If (IP_V_EFC_CTL_LSH_UP + V_EFC_LSH_UP_CTL_ADD[i] + C_V_EFC_AS_LSH_UP) < 0 , i.e. would cause the effective heater voltage to be negative, V_EFC_LSH_UP_CLC[i] shall be limited to 0 and further reduction of the effective heater voltage via by the closed loop controller P & I terms shall be prevented by setting LV_V_EFC_LSH_UP_LIM_BOL[i], until the

sum less than zero condition is no longer met at which point LV_V_EFC_LSH_UP_LIM_BOL[i] shall be reset.

For Bank 2; i = 2

V_EFC_LSH_UP_CLC[i] = IP_V_EFC_CTL_LSH_UP (N_32, TEG_CAT_UP_MDL[i])* IP_FAC_V_EFC_TEG_LSH_UP + V_EFC_LSH_UP_CTL_ADD[i] + C_V_EFC_AS_LSH_UP + V_EFC_ADD_LSH_UP_REST

Note: If (IP_V_EFC_CTL_LSH_UP* IP_FAC_V_EFC_TEG_LSH_UP + V_EFC_LSH_UP_CTL_ADD[i] + C_V_EFC_AS_LSH_UP) < 0 , i.e. would cause the effective heater voltage to be negative, V_EFC_LSH_UP_CLC[i] shall be limited to 0 and further reduction of the effective heater voltage via by the closed loop controller P & I terms shall be prevented by setting LV_V_EFC_LSH_UP_LIM_BOL[i], until the sum less than zero condition is no longer met at which point LV_V_EFC_LSH_UP_LIM_BOL[i] shall be reset.

Condition 8:"LSH_POW_CTL to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

Condition 9:"LSH_POW_CTL to LSH_TEMP_PROT"

TEG_CAT_UP_MDL[i] > C_TEG_CAT_UP_MDL_MAX_LSH_UP **&**
LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_TEMP_PROT

Condition 6:"LSH_POW_CTL to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_VB_PROT

Condition 14:"LSH_POW_CTL to LSH_POW_RED"

LV_TEG_MIN_THD[i] = 0

Transition actions:

T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
TTIP_MES_DIF_LS_UP[i] = 0
STATE_LSH_UP[i] = LSH_POW_RED

STATE_LSH_UP[i] "LSH_VB_PROT"

Actions:

V_EFC_CLC_LSH_UP[i] = C_V_EFC_PROT_VB_LSH_UP

<u>Condition 8</u>:“LSH_VB_PROT to LSH_OFF”

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

<u>Condition 9</u>:“LSH_VB_PROT to LSH_TEMP_PROT”

TEG_CAT_UP_MDL[i] > C_TEG_CAT_UP_MDL_MAX_LSH_UP **&**
LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

STATE_LSH_UP[i] = LSH_TEMP_PROT

<u>Condition 7</u>:“LSH_VB_PROT to LSH_POW_RISE”

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

Transition actions:

T_POW_RISE_LSH_UP[i] = 0
V_EFC_CLC_LSH_UP[i] = IP_V_EFC_INI_LSH_UP (TEMP_INI_LS_UP[i])
+ C_V_EFC_ADD_VB_PROT_LSH_UP
T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
TTIP_MES_DIF_LS_UP[i] = 0
STATE_LSH_UP[i] = LSH_POW_RISE

STATE_LSH_UP[i] “LSH_TEMP_PROT”

Actions:

V_EFC_CLC_LSH_UP[i] = 0

<u>Condition 8</u>:“LSH_TEMP_PROT to LSH_OFF”

LV_ST_END = 0 **or** LV_INH_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_OFF

<u>Condition 12</u>:“LSH_TEMP_PROT to LSH_VB_PROT”

LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

Transition actions:

STATE_LSH_UP[i] = LSH_VB_PROT

Condition 10:"LSH_TEMP_PROT to LSH_POW_CTL"

TEG_CAT_UP_MDL[i] ≤ C_TEG_CAT_UP_MDL_MAX_LSH_UP **&**
Previous state of STATE_LSH_UP[i] = LSH_POW_CTL

Transition actions:

STATE_LSH_UP[i] = LSH_POW_CTL

Condition 11:"LSH_TEMP_PROT to LSH_POW_RISE"

TEG_CAT_UP_MDL[i] ≤ C_TEG_CAT_UP_MDL_MAX_LSH_UP **&**
Previous state of STATE_LSH_UP[i] ≠ LSH_POW_CTL

Transition actions:

T_POW_RISE_LSH_UP[i] = 0
V_EFC_CLC_LSH_UP[i] = IP_V_EFC_INI_LSH_UP (TEMP_INI_LS_UP[i])
+ C_V_EFC_ADD_TEMP_PROT_LSH_UP
T_V_EFC_TOL_LSH_UP[i] = 0
T_V_EFC_LIM_LSH_UP[i] = 0
V_EFC_CTL_P_LSH_UP[i] = 0
V_EFC_CTL_I_LSH_UP[i] = 0
V_EFC_CTL_ADD_LSH_UP[i] = 0
LV_LSH_CTL_CLL_LSH_UP[i] = 0
LV_V_EFC_LIM_BOL_LSH_UP[i] = 0
LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
TTIP_MES_DIF_LS_UP[i] = 0
STATE_LSH_UP[i] = LSH_POW_RISE

Note: Should none of the conditions have been determined to be met, the state machine shall remain in the same state.

7.33.2 Oxygen sensor heater voltage protection

Prolonged exposure to heater voltages that exceed the absolute maximum ratings specified shall be prevented. The preventative measures shall be applicable to all heater states other than LSH_OFF.

If STATE_LSH_UP[i] ≠ LSH_OFF
then **If** (V_EFC_CLC_LSH_UP[i] ≥ C_V_EFC_TOL_LSH_UP)
then **If** (T_V_EFC_TOL_LSH_UP[i] ≥ C_T_MAX_V_EFC_TOL_LSH_UP)
then **If** T_V_EFC_LIM_LSH_UP[i] ≥
C_T_MAX_V_EFC_LIM_LSH_UP
then LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
Reset timer T_V_EFC_LIM_LSH_UP[i]
Reset timer T_V_EFC_TOL_LSH_UP[i]
else LV_V_EFC_LIM_TOL_LSH_UP[i] = 1
Increment timer T_V_EFC_LIM_LSH_UP[i]
endif
else LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
Reset timer T_V_EFC_LIM_LSH_UP[i]


```
                        Increment timer T_V_EFC_TOL_LSH_UP[i]
                endif
        else    LV_V_EFC_LIM_TOL_LSH_UP[i] = 0
                If      T_V_EFC_LIM_LSH_UP[i] = 0
                or      T_V_EFC_LIM_LSH_UP[i] ≥
                                C_T_MAX_V_EFC_LIM_LSH_UP
                then    Reset timer T_V_EFC_LIM_LSH_UP[i]
                        Reset timer T_V_EFC_TOL_LSH_UP[i]
                else    Increment timer T_V_EFC_LIM_LSH_UP[i]
                endif
        endif
        If      T_POW_RISE_LSH_UP[i] ≥ C_T_MAX_V_EFC_MAX_ST_LSH_UP
        or      (T_AST ≥ C_T_AST_MAX_LSH_UP
        and     STATE_LSH_UP[i] = "LSH_POW_CTL")
then    If      V_EFC_CLC_LSH_UP[i] > VB
                or      V_EFC_CLC_LSH_UP[i] > C_V_EFC_MAX_LSH_UP
                then    LV_V_EFC_LIM_MAX_LSH_UP[i] = 1
                        V_EFC_CLC_LSH_UP[i] = MIN (VB, C_V_EFC_MAX_LSH_UP)
                else    LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
                endif
else    If      V_EFC_CLC_LSH_UP[i] > VB
                or      V_EFC_CLC_LSH_UP[i] > C_V_EFC_MAX_ST_LSH_UP
                then    LV_V_EFC_LIM_MAX_LSH_UP[i] = 1
                        V_EFC_CLC_LSH_UP[i] =
                                    MIN (VB, C_V_EFC_MAX_ST_LSH_UP)
                else    LV_V_EFC_LIM_MAX_LSH_UP[i] = 0
                endif
        endif
        If      T_V_EFC_LIM_LSH_UP[i] <> 0
        then    V_EFC_LSH_UP[i] =
                            MIN (V_EFC_CLC_LSH_UP[i], C_V_EFC_LIM_LSH_UP)
        else    V_EFC_LSH_UP[i] = V_EFC_CLC_LSH_UP[i]
        endif
    endif
```

7.33.3 Oxygen sensor heater driver duty cycle computation

The PWM duty cycle for the appropriate heater driver shall continually be calculated in all heater states at the specified recurrence.

If (STATE_LSH_UP[i] = LSH_OFF)

then LSHPWM_UP[i] = LSHPWM_EXT_LS_UP[i]

else **If** (VB ≥ C_VB_MAX_PROT_LSH_UP)

then LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 1

LSHPWM_UP[i] = $C\_V\_EFC\_PROT\_VB\_LSH\_UP^2 * 100 / VB_N^2$

else LV_V_EFC_LIM_PROT_VB_LSH_UP[i] = 0

LSHPWM_UP[i] = $V\_EFC\_LSH\_UP[i]_N^2 *$

$C\_FAC\_V\_EFC\_AS\_LSH\_UP * 100 / VB_N^2$

If LSHPWM_UP[i] ≥ NC_LSHPWM_TOL_LSH_UP

then LSHPWM_UP[i] = NC_LSHPWM_TOL_LSH_UP

else **If** LSHPWM_UP[i] ≤ NC_LSHPWM_BOL_LSH_UP

then LSHPWM_UP[i] = NC_LSHPWM_BOL_LSH_UP

else LSHPWM_UP[i] = LSHPWM_UP[i]

endif

endif

endif

endif.

The value C_FAC_V_EFC_AS_LSH_UP is a correction signal generated in the application system.

NOTE: The recurrence of the heater driving duty cycle shall occur at the same recurrence as the Vbatt acquisition. This shall prevent short duration voltage drop-outs during engine start from causing long duration Vbatt corrections being made where the Vbatt has since recovered. This may otherwise cause excessive current to flow in the heater driver.

The computation of the duty cycle shall be checked to ensure that the result remains positive (i.e. ≥ 0) at all times and that now under- or overflows are caused by the values of the calibration system constants.

The multiplication of the corrected voltage ratio by 100 to obtain a unit of percent (%) may not necessarily be implemented in the SW but in the data bank definition of the variables.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_V_EFC_INI_LSH_UP	1*6	0 ... FFH	0 ... 26	0.102	V
LDPM_TEMP_INI_LS_UP_1_EGCP	6	0000 ... FFFFH	-2048...2047.94	62.5e-3	°C
Initial effective heater voltage					
IP_V_EFC_CTL_LSH_UP	6*6	0...FFH	0...26	0.102	V
LDPM_N_32_1_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_UP_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	0.0625	°C
Open loop control effective heater voltage					
IP_T_TEMP_THD_LSH_UP	1*6	0 ... FFH	0 ... 25.5	0.1	s
LDPM_TEMP_INI_LS_UP_1_EGCP	6	0000 ... FFFFH	-2048...2047.94	62.5e-3	°C
Duration until critical water splash sensor temperature reached					
IP_FAC_V_EFC_TEG_LSH_UP	1*6	0...FFH	0...4	0.015625	-
LDP_VS_IP_FAC_V_EFC_TEG_LSH_UP	6	0 ... FFH	0 ... 255	1	km/h
Factor for the heat transfer rate, bank 2 only					
C_V_EFC_INC_LSH_UP	1	0...FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage increment to raise temperature					
C_V_EFC_INC_PROT_LSH_UP	1	0...FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage increment to raise temperature in case of transition from TEMP_PROT or VB_PROT					
C_V_EFC_DEC_LSH_UP	1	0 ... FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage decrement to lower temperature					
C_V_EFC_RED_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Reduced effective heater voltage during danger of water splash damage					
C_V_EFC_STEP_LSH_UP	1	80 ... 7FH	-13.05 ... 12.95	0.102	V
Additional effective heater voltage step on transition from LSH_POW_RED to LSH_POW_RISE					
C_T_AST_MAX_LSH_UP	1	0 ... FFFFH	0 ... 6553.5	0.1	s
Time after start at which step in max. effective heater voltage applied, if LSH_POW_CTL active (T_POW_RISE_LSH_UP[NC_CBK_EX_NR] stopped)					
C_T_MAX_POW_RISE_LSH_UP	1	0 ... FFFFH	0 ... 6553.5	0.1	s
Duration of open loop pre-heating in LSH_POW_RISE from dew-point to set-point temperature					
C_T_MAX_V_EFC_TOL_LSH_UP	1	0 ... FFH	0 ... 25.5	0.1	s
Maximum permitted duration where excessive effective heater voltage tolerated					
C_T_MAX_V_EFC_LIM_LSH_UP	1	0 ... FFH	0 ... 25.5	0.1	s
Maximum duration of effective heater voltage limiting					
C_T_MAX_V_EFC_MAX_ST_LSH_UP	1	0 ... FFH	0 ... 25.5	0.1	s
Time after start at which step in max. effective heater voltage applied					
C_V_EFC_TOL_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage threshold above which voltage limiting may occur if threshold persistently exceeded					
C_V_EFC_LIM_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage limit under persistent excessive heater voltage condition					
C_VB_MAX_PROT_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Maximum permitted battery voltage threshold, over which overvoltage detected					
C_V_EFC_PROT_VB_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage in the case of battery voltage overvoltage condition					
C_V_EFC_MAX_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Absolute maximum permitted effective heater voltage after initial phase; threshold & limit					
C_V_EFC_MAX_ST_LSH_UP	1	0 ... FFH	0 ... 26	0.102	V
Absolute maximum permitted effective heater voltage during initial phase; threshold & limit					
C_V_EFC_ADD_LSH_UP_REST	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage reduction in case of active LSH_REST and low R_IT.					
C_V_EFC_ADD_TEMP_PROT_LSH_UP	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage reduction in case of transition from TEMP_PROT to POW_RISE					
C_R_IT_LS_UP_MIN_REST	1	0 ... FFFFH	0 ... 65535	1	Ω
Minimum R_IT after LSH_REST below which LSH voltage is reduced.					
C_CTR_R_IT_LS_UP_VLD_REST	1	0 ... FFFFH	0 ... 65535	1	-
Number of valid R_IT values checked after LSH_REST.					
C_TTIP_SP_LS_UP[NC_CBK_EX_NR]	1	8000 ... 7FFFH	-2048...2047.938	62.5e-3	°C

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
Oxygen sensor set temperature for closed loop control					
IP_FAC_NEG_CTL_P_LSH_UP	6*6	0 ... FFH	0 ... 0.255	0.001	V/K
LDPM_N_32_1_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_UP_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	62.5e-3	°C
Closed loop controller P-term factor for temperature too high; decrease heater voltage					
IP_FAC_POS_CTL_P_LSH_UP	6*6	0 ... FFH	0 ... 0.255	0.001	V/K
LDPM_N_32_1_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_UP_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	62.5e-3	°C
Closed loop controller P-term factor for temperature too low; increase heater voltage					
C_FAC_CTL_P_LSH_UP	1	0 ... FFH	0...6.348	24.89e-3	-
Multiplicative scaling factor for closed loop P-term					
C_TTIP_DIF_MIN_CTL_P_LS_UP	1	0 ... 7FFFH	0...2047.94	62.5e-3	°C
Minimum required temperature difference in order to calculate P term of controller					
IP_FAC_NEG_CTL_I_LSH_UP	6*6	0 ... FFH	0 ... 0.255	0.001	V/K
LDPM_N_32_1_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_UP_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	62.5e-3	°C
Closed loop controller I-term factor for temperature too high; decrease heater voltage					
IP_FAC_POS_CTL_I_LSH_UP	6*6	0 ... FFH	0 ... 0.255	0.001	V/K
LDPM_N_32_1_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_UP_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	62.5e-3	°C
Closed loop controller I-term factor for temperature too low; increase heater voltage					
ID_TTIP_SP_DIF_LSH_UP	1*4	8000 ... 7FFFH	-2048...2047.938	62.5e-3	°C
LDP_T_POW_RISE_LSH_UP_ID_TTIP	4	0 ... FFFFH	0 ... 6553.5	0.1	s
Reducing value for TTIP threshold of transition conditions LSH_POW_RISE to LSH_POW_CTL					
C_FAC_CTL_I_LSH_UP	1	0 ... FFH	0...6.348	24.89e-3	-
Multiplicative scaling factor for closed loop I-term					
C_V_EFC_AS_LSH_UP	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage calibration system additive factor					
C_FAC_V_EFC_AS_LSH_UP	1	0 ... FFH	0 ... 1.9922	7.8125e-3	-
Multiplicative correction factor to effective heater voltage					
C_TEG_CAT_UP_MDL_MAX_LSH_UP	1	0...7FF0H	0 ... 2047	62.5e-3	°C
Maximum permitted exhaust gas temperature above which LSH switched off					
C_TCO_DIF_MAX_LSH_UP_REST	1	0...FEH	-48 ... 142.5	0.75	°C
Maximum TCO decrease since last engine stop for hot sensor restart detection					
C_TCO_ST_MIN_LSH_UP_REST	1	0...FEH	-48 ... 142.5	0.75	°C
Minimum TCO_ST to activate hot sensor restart detection					
LC_LSH_CTL_CLL_INH_LSH_UP[NC_CBK_EX_NR]	1	0 ... 01H	0 ... 1	1	-
Boolean constant to permit suspension of closed loop heater control					
C_V_EFC_ADD_VB_PROT_LSH_UP	1	80…7FH	-13.05…12.95	0.102	V
Reduction of initial lambda sensor heater voltage after VB_PROT					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_LSHPWM_TOL_LSH_UP	1	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle upper limit					
NC_LSHPWM_BOL_LSH_UP	1	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle lower limit					

7.34 Application incidences for upstream oxygen sensor heater management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for heating not met					
LV_INH_LSH_CTL_CLL_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions closed loop heater control not met					
TEMP_INI_LS_UP[NC_CBK_EX_NR]	V/O	8000 ... 7FFFH	-2048...2047.94	62.5e-3	°C
Oxygen sensor temperature at engine start					

Input data:

NC_CBK_EX_NR	LV_ST_END	LV_ERR_OBD_VLD_LSH_UP[i]	LV_ERR_TTIP_OBD_LS_UP[i]
		LV_LSH_SCG_ACT_LSH_UP[i]	LV_ERR_LSH_UP[NC_CBK_EX_NR]
TEG_DYN_LS_UP[NC_CBK_EX_NR]		LV_LSHPWM_UP_EXT_ADJ	

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank.

Description:

The Boolean constant LC_LSH_UP[i] shall permit the appropriate oxygen sensor to be heated. When set, the Boolean inhibit flag LV_INH_LSH_UP[i] shall be reset. When reset, LV_INH_LSH_UP[i] shall be set.

The Boolean flag LV_INH_LSH_CTL_CLL_LSH_UP[i] shall indicate whether closed loop heating shall be inhibited or not. Should either the Boolean constant LC_LSH_CTL_CLL_LSH_UP[i] be reset or the Boolean input data LV_ERR_OBD_VLD_LSH_UP[i], LV_ERR_TTIP_OBD_LS_UP[i], LV_LSH_SCG_ACT_LSH_UP[i] or LV_ERR_LSH_UP[i] be set then the Boolean inhibit flag LV_INH_LSH_CTL_CLL_LSH_UP[i] shall be set (open loop heater control), otherwise LV_INH_LSH_CTL_CLL_LSH_UP[i] shall be reset (closed loop heater control permitted).

The oxygen sensor temperature at start TEMP_INI_LS_UP[i] shall be obtained from the value TEG_DYN_LS_UP out of the model in EXTD.

Application conditions:

Activation:

Transition from engine start: LV_ST_END = 1

Deactivation:

Transition to engine stop: LV_ST_END = 0

Time recurrence:

Once every 1 s.

Initialisation:

The following variable initialisation shall take place at the beginning of a new driving cycle.

LV_INH_LSH_UP[i] = 0
LV_INH_LSH_CTL_CLL_LSH_UP[i] = 0

TEMP_INI_LS_UP[i] = TEG_DYN_LS_UP[i]

Formula section:

If LV_LSHPWM_UP_EXT_ADJ = 1

Then LV_INH_LSH_UP[i] = 1

Else LV_INH_LSH_UP[i] = **!** LC_LSH_UP[i]

Endif

If LC_LSH_CTL_CLL_LSH_UP[i] = 0

or LV_ERR_OBD_VLD_LSH_UP[i] = 1

or LV_ERR_TTIP_OBD_LS_UP[i] = 1

or LV_LSH_SCG_ACT_LSH_UP[i] = 1

or LV_ERR_LSH_UP[i] = 1

then LV_INH_LSH_CTL_CLL_LSH_UP[i] = 1

else LV_INH_LSH_CTL_CLL_LSH_UP[i] = 0

endif.

Calculation of Oxygen sensor temperature at engine start

One time determination after finishing engine state start

IF LV_ST_END = 1 for first time after LV_ST_END = 0

Then TEMP_INI_LS_UP[i] = TEG_DYN_LS_UP[i]

Endif

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_LSH_UP[NC_CBK_EX_NR]	1	0...1H	0...1	1	-
Boolean variable to enable oxygen sensor heating					
LC_LSH_CTL_CLL_LSH_UP[NC_CBK_EX_NR]	1	0...1H	0...1	1	-
Boolean variable to enable closed loop oxygen sensor heater control					

7.35 Downstream oxygen sensor signal voltage

7.35.1 Signal evaluation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_DOWN_DRV1_MMV[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.5...2.499924	7.6294E-5	V/10ms
Mean of buffer containing 1st derivative of VLS_DOWN[i] signal					
VLS_DOWN_DRV1_MMV_MIN[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.5...2.499924	7.6294E-5	V/10ms
Minimum value of mean of buffer containing 1st derivative of VLS_DOWN[i] signal in lean phase					
VLS_DOWN_DRV1_ABS_MAX[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V/10ms
Absolute maximum value within buffer containing 1st derivative of VLS_DOWN[i] signal					
VLS_DOWN_MMV_MIN[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V
Mean oxygen sensor lean voltage					
VLS_DOWN_MMV_MAX[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V
Mean oxygen sensor rich voltage					
VLS_DOWN_BOL[NC_CBK_EX_NR]	V/O	0...3FFH	0…4.995117	4.8828E-3	V
Variable switching threshold for calculation of VLS_DOWN_MMV_MIN[i]					
VLS_DOWN_TOL[NC_CBK_EX_NR]	V/O	0...3FFH	0…4.995117	4.8828E-3	V
Variable switching threshold for calculation of VLS_DOWN_MMV_MAX[i]					
VLS_DOWN_MMV_HYS[NC_CBK_EX_NR]	V	0...3333H	0…0.999985	7.6294E-5	V
Variable hysteresis used to determine switching thresholds					
LV_VLS_DOWN_MMV_LIM[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating VLS_DOWN_MMV_xxx[i] signals being limited when set					
LV_VLS_DOWN_MMV_ACT	V/O	0...1H	0...1	1	-
Boolean flag indicating calibrateable delay since PUC deactivation passed					

Input data:

LV_ST_END	LV_ES	LV_PUC	NC_CBK_EX_NR
VLS_DOWN[NC_CBK_EX_NR]			

FUNCTION DESCRIPTION:

General information:

The function shall evaluate the oxygen sensor signal voltage and compute a number of resultant values that may be used by other functions. These values shall include:

1. Moving mean of sensor voltage first derivative, i.e. average signal gradient
2. Minimum value of moving mean of sensor voltage first derivative in lean phase
3. Absolute maximum of sensor voltage first derivative over defined number of previous samples, i.e. measure of sensor voltage stability
4. Moving mean value of lean sensor voltage
5. Moving mean value of rich sensor voltage.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2

otherwise (NC_CBK_EX_NR = 1)

i = 1, for single exhaust cylinder bank

Signal flow diagram:

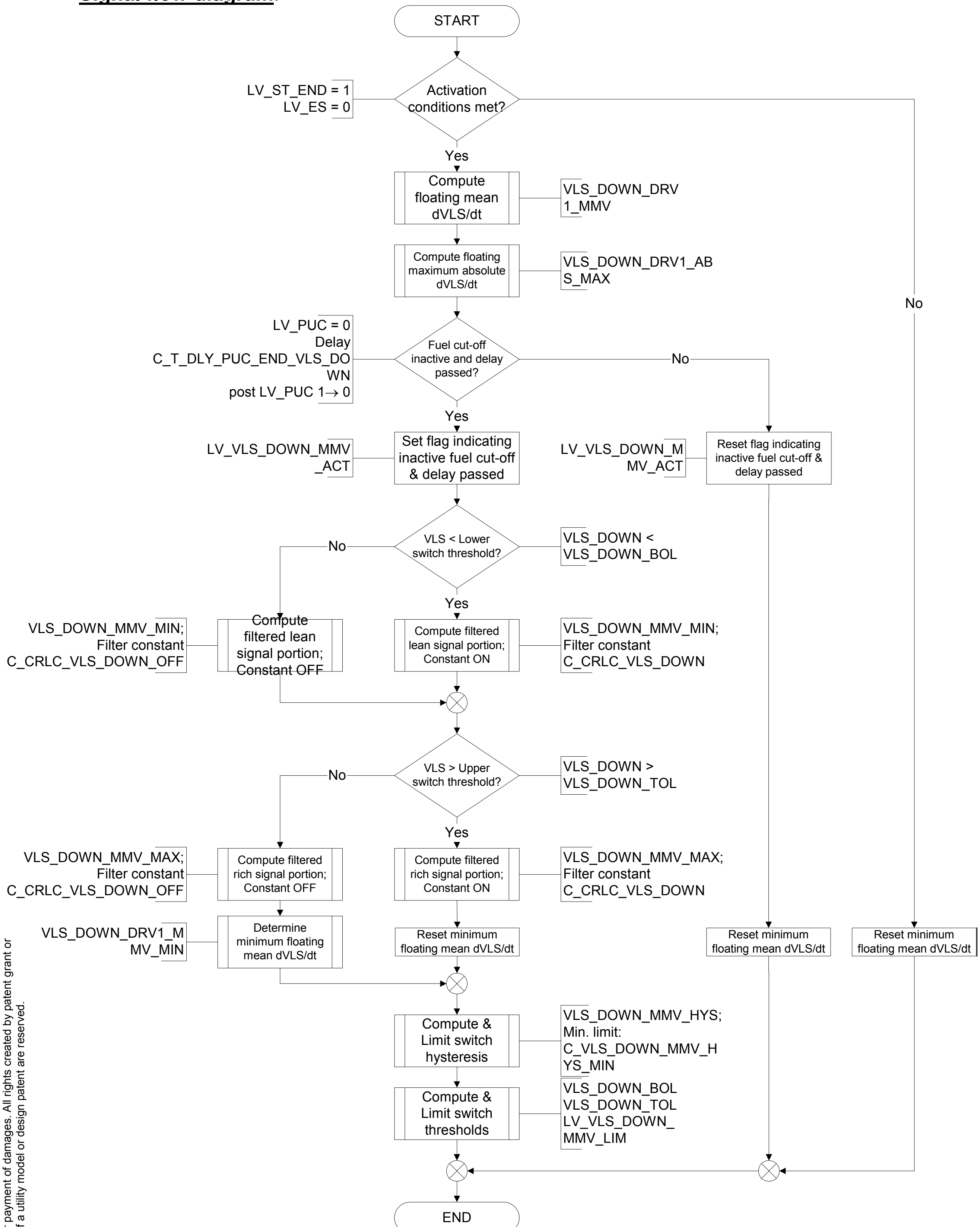


Figure 1 Function flow chart, Signal evaluation

Description:

The function shall evaluate the downstream oxygen sensor signal VLS_DOWN[i] and provide variables quantifying the signal characteristics: Floating mean gradient, Floating maximum signal deviation between samples, Filtered mean rich signal voltage & Filtered mean lean signal voltage (voltage will deviate from stable level signal with ageing of the catalyst). This shall be carried out as follows:

The function shall be activated should the engine state transition out of start (LV_ST_END = 1) and shall be deactivated should the engine stop state (LV_ES = 1) be entered.

Once the activation conditions have been met the function may be described in the following blocks:

1. Computation of moving mean of first derivative VLS_DOWN_DRV1_MMV[i]: The moving mean value of oxygen sensor signal voltage first derivative VLS_DOWN_DRV1_MMV[i] shall be computed by averaging the last differences between consecutive VLS_DOWN[i] samples, i.e. the most current difference ($VLS\_DOWN[i]_N$ - $VLS\_DOWN[i]_{N-1}$) and the previous 5 differences. This shall indicate a measure for the signal gradient.

2. Computation of absolute maximum of first derivative VLS_DOWN_DRV1_ABS_MAX[i]: The absolute maximum of the oxygen sensor voltage first derivative VLS_DOWN_DRV1_ABS_MAX[i] shall be computed by taking the maximum of the absolute values of the last differences between consecutive VLS_DOWN[i] samples, i.e. the most current difference ($VLS\_DOWN[i]_N$ - $VLS\_DOWN[i]_{N-1}$) and the previous 5 differences. This shall indicate a measure for the signal stability.

3. Computation of moving mean value of lean and rich sensor voltages, VLS_DOWN_MMV_xxx[i]: The computation of the moving mean value of lean and rich sensor voltages, associated switch thresholds and switch hysteresis shall only be carried out when the overrun fuel cut-off flag is reset (LV_PUC = 0) and the time C_T_DLY_PUC_END_VLS_DOWN has passed since the deactivation of the last occurring engine state PUC (i.e. LV_PUC $1 \rightarrow 0$). The lean and rich mean signal voltages shall be implemented by use of first order low pass filtering and floating switching thresholds. For example:

Should the VLS_DOWN[i] signal exceeds the threshold VLS_DOWN_TOL[i], the rich mean signal voltage VLS_DOWN_MMV_MAX[i] shall be computed by filtering the VLS_DOWN[i] signal with filter constant C_CRLC_VLS_DOWN. This shall continue until threshold VLS_DOWN_TOL[i] is no longer exceeded, after which VLS_DOWN_MMV_MAX[i] shall be computed by filtering with filter constant C_CRLC_VLS_DOWN_OFF. This constant is much smaller than C_CRLC_VLS_DOWN and the effect is to cause the VLS_DOWN_MMV_MAX[i] to follow the rich peaks. The OFF constant shall permit the VLS_DOWN_MMV_MAX[i] signal to follow the VLS_DOWN[i] signal for sudden decreases in signal amplitude, i.e. maximum VLS_DOWN[i] lower than VLS_DOWN_TOL[i].

The same procedure shall be carried out for the computation of VLS_DOWN_MMV_MIN[i], i.e. where VLS_DOWN[i] falls below VLS_DOWN_BOL[i], VLS_DOWN_MMV_MIN[i] shall be computed with filter constant C_CRLC_VLS_DOWN and where VLS_DOWN[i] no longer falls below VLS_DOWN_BOL[i], the filter constant C_CRLC_VLS_DOWN_OFF shall be used.

4. Determination of the minimum value of the moving mean of first derivative VLS_DOWN_DRV1_MMV_MIN[i]: The same conditions as for the computation of VLS_DOWN_MMV_xxx[i] shall apply. Should VLS_DOWN[i] fall below or equal VLS_DOWN_TOL[i], $VLS\_DOWN\_DRV1\_MMV\_MIN[i]_N$ shall be equal to the lower of $VLS\_DOWN\_DRV1\_MMV\_MIN[i]_{N-1}$ & VLS_DOWN_DRV1_MMV[i]. When VLS_DOWN[i] exceeds VLS_DOWN_TOL[i], the LV_PUC condition in conjunction with delay

C_T_DLY_PUC_END_VLS_DOWN is not met or the function is deactivated, VLS_DOWN_DRV1_MMV_MIN[i] shall be reset to 0.

5. Compute & limit switch hysteresis: The value of the hysteresis, VLS_DOWN_MMV_HYS[i], used to determine the switching thresholds VLS_DOWN_BOL[i] & VLS_DOWN_TOL[i] shall be made proportional using factor C_FAC_VLS_DOWN_MMV_HYS to the oxygen sensor signal amplitude, as determined by taking the absolute difference of the moving mean lean / rich variables. The hysteresis shall also be limited so that it may not fall below the calibrateable threshold C_VLS_DOWN_MMV_HYS_MIN.

6. Compute & limit switch thresholds: The function shall limit the minimum amplitude of the difference between the moving mean lean / rich variables (VLS_DOWN_MMV_MAX[i] - VLS_DOWN_MMV_MIN[i]) to twice that of the switch hysteresis, VLS_DOWN_MMV_HYS[i].

Should the amplitude of the aforementioned difference fall below the said threshold, both the switch thresholds VLS_DOWN_BOL[i] & VLS_DOWN_TOL[i] shall be set to the average of the moving mean lean / rich variables, the moving mean lean / rich variables are recalculated based on the limited switching thresholds and hysteresis and the flag LV_VLS_DOWN_MMV_LIM[i] shall be set to indicate that switch threshold limiting is active.

Should no limiting be required, the switch thresholds VLS_DOWN_BOL[i] & VLS_DOWN_TOL[i] shall be calculated based on the switch hysteresis, moving mean lean variable and moving mean rich variable respectively. The flag LV_VLS_DOWN_MMV_LIM[i] shall be reset to indicate that no switch threshold limiting is being carried out.

The above blocks shall be cycled through once every function recurrence.

Application conditions:

Recurrence:

The function shall be carried out once every oxygen sensor acquisition, i.e. every 10 ms.

Initialisation:

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop (LV_ES).

VLS_DOWN_MMV_MIN[i] = C_VLS_DOWN_INI_MIN
VLS_DOWN_MMV_MAX[i] = C_VLS_DOWN_INI_MAX
VLS_DOWN_BOL[i] = C_VLS_DOWN_INI_MAX
VLS_DOWN_TOL[i] = C_VLS_DOWN_INI_MIN
VLS_DOWN_MMV_HYS[i] = C_VLS_DOWN_MMV_HYS_MIN
LV_VLS_DOWN_MMV_LIM[i] = 0
LV_VLS_DOWN_MMV_ACT = 0
VLS_DOWN_DRV1_MMV[i] = 0
VLS_DOWN_DRV1_MMV_MIN[i] = 0
VLS_DOWN_DRV1_ABS_MAX[i] = 0

Buffer contents from which VLS_DOWN_DRV1_MMV[i] & VLS_DOWN_DRV1_ABS_MAX[i] computed = 0.

Activation:

If LV_ST_END = 1

then ***Function activated***

endif.

Deactivation:

If LV_ST_END = 0

then ***Function deactivated***
VLS_DOWN_DRV1_MMV_MIN[i] = 0

endif.

Formula section:

The following function shall be carried out in the order listed and as indicated by the function flow chart:

Computation of moving mean of first derivative VLS_DOWN_DRV1_MMV[i]

$$\text{VLS\_DOWN\_DRV1\_MMV[i]} = \frac{\sum_{x=0}^{5}\left(\text{VLS\_DOWN}[i]_{N-x} - \text{VLS\_DOWN}[i]_{N-(x+1)}\right)}{6 * 10\text{ms}}$$

The above may be facilitated for example by use of a 6 value ring buffer. The most current difference, (VLS_DOWN$[i]_N$ - VLS_DOWN$[i]_{N-1}$), would overwrite the oldest difference in the buffer, (VLS_DOWN$[i]_{N-5}$ - VLS_DOWN$[i]_{N-6}$), and the average of the buffer contents recalculated.

Computation of absolute maximum of first derivative number of previous samples VLS_DOWN_DRV1_ABS_MAX[i]

$$\text{VLS\_DOWN\_DRV1\_ABS\_MAX[i]} = \text{MAXIMUM}\left|{}_{x=0}^{5}\left(\text{VLS\_DOWN}[i]_{N-x} - \text{VLS\_DOWN}[i]_{N-(x+1)}\right)\right|$$

The above may be facilitated for example by use of the same 6 value ring buffer, as proposed above for the computation of VLS_DOWN_DRV1_MMV[i]. The function would calculate the absolute values of the buffer contents and take the maximum of these values.

Computation of moving mean value of lean and rich sensor voltages, VLS_DOWN_MMV_xxx[i]

If (LV_PUC = 0) **&**
(delay since last LV_PUC 1 → 0 transition ≥ C_T_DLY_PUC_END_VLS_DOWN)

then LV_VLS_DOWN_MMV_ACT = 1
If (VLS_DOWN[i] < VLS_DOWN_BOL[i])

then *Compute moving mean lean signal, filter constant ON*

else *Compute moving mean lean signal, filter constant OFF*

endif

If (VLS_DOWN[i] > VLS_DOWN_TOL[i])

then *Compute moving mean rich signal, filter constant ON*
VLS_DOWN_DRV1_MMV_MIN[i] = 0

else *Compute moving mean rich signal, filter constant OFF*
Determine minimum of moving mean of first derivative

endif

Compute & limit switch hysteresis

Compute & limit switch thresholds

else LV_VLS_DOWN_MMV_ACT = 0
Reset timer used for delay C_T_DLY_PUC_END_VLS_DOWN
VLS_DOWN_DRV1_MMV_MIN[i] = 0

endif.

Compute moving mean lean signal, filter constant ON

The moving mean of the lean signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_DOWN.

$$VLS\_DOWN\_MMV\_MIN[i]_N = VLS\_DOWN\_MMV\_MIN[i]_{N-1} * (1-C\_CRLC\_VLS\_DOWN) + VLS\_DOWN[i]_N * C\_CRLC\_VLS\_DOWN$$

Compute moving mean lean signal, filter constant OFF

The moving mean of the lean signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_DOWN_OFF.

$$VLS\_DOWN\_MMV\_MIN[i]_N = VLS\_DOWN\_MMV\_MIN[i]_{N-1} * (1-C\_CRLC\_VLS\_DOWN\_OFF) + VLS\_DOWN[i]_N * C\_CRLC\_VLS\_DOWN\_OFF$$

Compute moving mean rich signal, filter constant ON

The moving mean of the rich signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_DOWN.

$$VLS\_DOWN\_MMV\_MAX[i]_N = VLS\_DOWN\_MMV\_MAX[i]_{N-1} * (1-C\_CRLC\_VLS\_DOWN) + VLS\_DOWN[i]_N * C\_CRLC\_VLS\_DOWN$$

Compute moving mean rich signal, filter constant OFF

The moving mean of the rich signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_DOWN_OFF.

$$VLS\_DOWN\_MMV\_MAX[i]_N = VLS\_DOWN\_MMV\_MAX[i]_{N-1} * (1-C\_CRLC\_VLS\_DOWN\_OFF) + VLS\_DOWN[i]_N * C\_CRLC\_VLS\_DOWN\_OFF$$

Determine minimum of moving mean of first derivative VLS_DOWN_DRV1_MMV_MIN[i]

The determination of the minimum of VLS_DOWN_DRV1_MMV[i] shall take place when the condition VLS_DOWN[i] ≤ VLS_DOWN_TOL[i] has been met. In this case:

VLS_DOWN_DRV1_MMV_MIN[i] = MIN(VLS_DOWN_DRV1_MMV_MIN[i],
VLS_DOWN_DRV1_MMV[i])

Should the above condition no longer met, the LV_PUC with delay condition not be met or the initial activation conditions not be met then VLS_DOWN_DRV1_MMV_MIN[i] shall be reset to 0.

Compute & limit switch hysteresis

VLS_DOWN_MMV_HYS[i] = | (VLS_DOWN_MMV_MAX[i] - VLS_DOWN_MMV_MIN[i]) | * C_FAC_VLS_DOWN_MMV_HYS

If (VLS_DOWN_MMV_HYS[i] < C_VLS_DOWN_MMV_HYS_MIN)

then VLS_DOWN_MMV_HYS[i] = C_VLS_DOWN_MMV_HYS_MIN

endif.

Compute & limit switch thresholds

If ((VLS_DOWN_MMV_MAX[i] - VLS_DOWN_MMV_MIN[i])
≤ 2 * VLS_DOWN_MMV_HYS[i])

then VLS_DOWN_BOL[i] = (VLS_DOWN_MMV_MAX[i] + VLS_DOWN_MMV_MIN[i]) / 2
VLS_DOWN_TOL[i] = (VLS_DOWN_MMV_MAX[i] + VLS_DOWN_MMV_MIN[i]) / 2

VLS_DOWN_MMV_MIN[i] = VLS_DOWN_BOL[i] - VLS_DOWN_MMV_HYS[i]
VLS_DOWN_MMV_MAX[i] = VLS_DOWN_TOL[i] + VLS_DOWN_MMV_HYS[i]

LV_VLS_DOWN_MMV_LIM[i] = 1

else VLS_DOWN_BOL[i] = VLS_DOWN_MMV_MIN[i] + VLS_DOWN_MMV_HYS[i]
VLS_DOWN_TOL[i] = VLS_DOWN_MMV_MAX[i] - VLS_DOWN_MMV_HYS[i]

LV_VLS_DOWN_MMV_LIM[i] = 0

endif.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VLS_DOWN_MMV_HYS_MIN	1	0...3333H	0…0.999985	7.6294E-5	V
Minimum hysteresis for the computation of the thresholds VLS_DOWN_TOL[i] and VLS_DOWN_BOL[i]					
C_VLS_DOWN_INI_MIN	1	0...3FFH	0…4.995117	4.8828E-3	V
Lower initialisation value for VLS_DOWN_TOL[i]. VLS_DOWN_MMV_MIN[i]					
C_VLS_DOWN_INI_MAX	1	0...3FFH	0…4.995117	4.8828E-3	V
Upper initialisation value for VLS_DOWN_BOL[i]. VLS_DOWN_MMV_MAX[i]					
C_CRLC_VLS_DOWN	1	0...FFFFH	0...0.999985	1.5259E-5	-
Filter constant for determination of VLS_DOWN_MMV_xxx[i] values when respective threshold passed					
C_CRLC_VLS_DOWN_OFF	1	0...FFFFH	0...0.999985	1.5259E-5	-
Filter constant for determination of VLS_DOWN_MMV_xxx[i] values when respective threshold not passed					
C_FAC_VLS_DOWN_MMV_HYS	1	0...FFH	0...0.996094	3.9063E-3	-
Multiplicative factor governing amplitude of hysteresis					
C_T_DLY_PUC_END_VLS_DOWN	1	0...FFFFH	0...655.35	1E-2	s
Delay between completion of engine state PUC and computation of VLS_DOWN_MMV_xxx[i]					

7.36 Downstream oxygen sensor operative readiness detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LS_DOWN_READY[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Logical variable for operative readiness of downstream oxygen sensor(s)					

Input data:

LV_PUC	MAF_INT_PUC	LV_ES	NC_CBK_EX_NR
VLS_DOWN[NC_CBK_EX_NR]		TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]	
LV_TEG_CAT_DOWN_MIN_THD[NC_CBK_EX_NR]		STATE_LSH_DOWN[NC_CBK_EX_NR]	

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

The operative readiness of the downstream oxygen sensor(s) shall be determined by this function to allow other functions to act accordingly dependent upon the readiness.

Description:

The operative readiness shall only be determined for a downstream oxygen sensor that has not yet been determined as being operatively ready (i.e. LV_LS_DOWN_READY[i] = 0).

The operative readiness shall be determined by analysing the downstream oxygen sensor voltage (VLS_DOWN[i]) and determining whether the voltage has left a pre-determined voltage band (voltage level at ADC input driven by potential divider in input interface for cold oxygen sensor, voltage level will leave band dependent upon A/F ratio present at the oxygen sensor location).

A second condition shall permit the forced operative readiness state, should the voltage level condition not be met, by assuming that when the catalyst temperature model has reached a pre-determined threshold, that the downstream oxygen sensor is also at operating temperature and functioning.

The operative readiness of the downstream oxygen sensor(s) shall be recorded in the variable LV_LS_DOWN_READY[i].

Application conditions:

Initialisation:

The oxygen sensor shall be initialised as inoperable (LV_LS_DOWN_READY[i]= 0) after a reset, in the state *Engine Stop*, when during PUC the MAF integral exceeds a calibrated threshold or when the dew point was exceeded once and LV_TEG_CAT_DOWN_MIN_THD[i] switches again to 0

if [*reset occurs* **or** LV_ES = 1 **or** (LV_PUC = 1 **and**
MAF_INT_PUC >= C_MAF_INT_PUC_MAX_DOWN_LS_READY) **or**
LV_TEG_CAT_DOWN_MIN_THD[i] = 1 -> 0]
then
LV_LS_DOWN_READY[i]= 0
endif

Recurrence:

T_SAMPLE = 100 ms

Activation:

LV_LS_DOWN_READY[i] = 0 **and**
(LV_PUC = 0 **or** MAF_INT_PUC < C_MAF_INT_PUC_MAX_DOWN_LS_READY)

Deactivation:

LV_LS_DOWN_READY[i] = 1 **or**
(LV_PUC = 1 **and** MAF_INT_PUC >= C_MAF_INT_PUC_MAX_DOWN_LS_READY) **or**
LV_TEG_CAT_DOWN_MIN_THD[i] = 1 -> 0

Formula section:

if (((C_VLS_DOWN_AFR_THD_READY < VLS_DOWN[i] **or**
VLS_DOWN[i] < C_VLS_DOWN_AFL_THD_READY) **and**
STATE_LSH_DOWN[i] = LSH_POW_CTL) **or**
TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_LS_DOWN_READY)
then
LV_LS_DOWN_READY[i]= 1
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEG_CAT_DOWN_LS_DOWN_READY	1	0...7FF0H	0...2047	0.0625	°C
Temperature threshold above which O2 sensor recognised as operatively ready					
C_VLS_DOWN_AFL_THD_READY	1	0...3FFH	0...4.995117	4.8828E-3	V
Lean VLS_DOWN[i] threshold for operative readiness detection					
C_VLS_DOWN_AFR_THD_READY	1	0...3FFH	0...4.995117	4.8828E-3	V
Rich VLS_DOWN[i] threshold for operative readiness detection					
C_MAF_INT_PUC_MAX_DOWN_LS_READY	1	0...FFFFH	0...1820.43123	2.7778E-2	g
Maximum MAF integral during PUC for operability reset					

7.37 Downstream oxygen sensor heater management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit

LSHPWM_DOWN[NC_CBK_EX_NR]	V/O	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle; acquired also by BSW					
V_EFC_LSH_DOWN[NC_CBK_EX_NR]	V/O	0 ... FFH	0 ... 26	0.102	V
Effective final output value of heater voltage					
V_EFC_CLC_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... FFFFH	0 ... 26	396.73e-6	V
Effective calculated heater voltage value prior to overvoltage protection					
V_EFC_ADD_LSH_DOWN_REST	V	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage offset in case of active LSH_REST and low R_IT.					
T_POW_RISE_LSH_DOWN[NC_CBK_EX_NR]	V/O	0 ... FFFFH	0 ... 6553.5	0.1	s
Timer indicating the duration of the pre-heating and post dew point heating phases					
T_V_EFC_TOL_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... FFH	0 ... 25.5	0.1	s
Timer indicating permitted effective heater voltage overvoltage duration					
T_V_EFC_LIM_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... FFH	0 ... 25.5	0.1	s
Timer indicating the duration of the effective heater voltage limitation preventing overvoltage					
LV_V_EFC_LIM_BOL_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to sum of voltage components falling below zero.					
LV_V_EFC_LIM_TOL_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to exceeding threshold for period of time.					
LV_V_EFC_LIM_MAX_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean flag indicating effective heater voltage limited due to exceeding absolute maximum voltage spec.					
LV_V_EFC_LIM_PROT_VB_LSH_DOWN[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean variable indicating effective heater voltage limited due to excessive battery voltage level					
LV_LSH_DOWN_REST[NC_CBK_EX_NR]	V	0 ... 01H	0 ... 1	1	-
Boolean variable indicating that the preheating has been interrupted by the restart detection function.					
STATE_LSH_DOWN[NC_CBK_EX_NR]	V/O	00H 01H 02H 03H 04H 05H 06H	LSH_OFF LSH_POW_RISE LSH_POW_RED LSH_POW_FALL LSH_POW_CTL LSH_VB_PROT LSH_TEMP_PROT	-	-
Present heater state					

Input data:

LV_INH_LSH_DOWN[NC_CBK_EX_NR]		TEMP_INI_LS_DOWN[NC_CBK_EX_NR]	LSHPWM_EXT_LS_DOWN[NC_CBK_EX_NR]
LV_ST_END	VS	TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]	LV_TEG_CAT_DOWN_MIN_THD[NC_CBK_EX_NR]
VB	N_32	T_AST	NC_CBK_EX_NR
C_VLS_DOWN_AFR_THD_READY	C_VLS_DOWN_AFL_THD_READY	TCO_STOP	TCO_ST
VLS_DOWN	R_IT_LS_DOWN		CTR_CYCNR_R_IT_LS_DOWN_VLD

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then

i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

The goal of the heater control shall be to control the oxygen sensor heating such that the optimal operating temperature is reached in the shortest time possible. At the same time, the maximum permissible temperature gradient (possible damage to ceramic due to thermal stress), the possible occurrence of water splash under the dew point (possible damage to ceramic due to thermal shock) and the absolute maximum ratings specified for the sensor shall be taken into account. The function shall permit oxygen sensor heating under open loop control.

Signal flow:

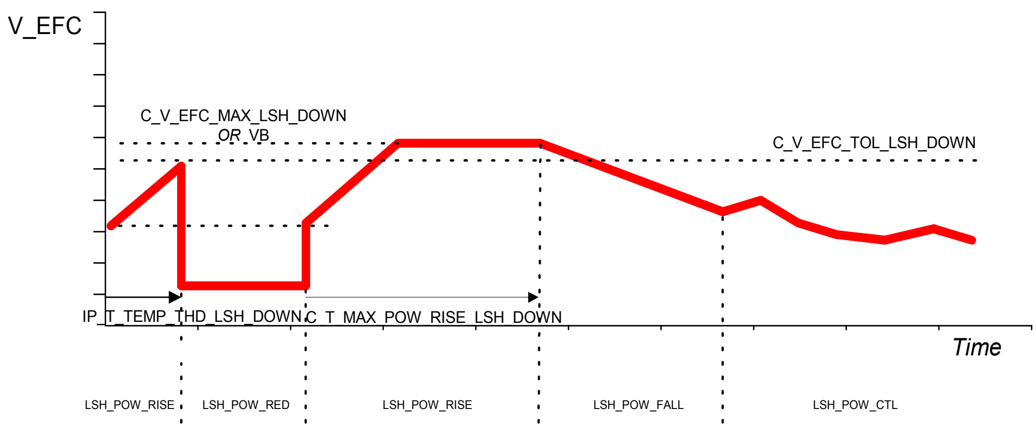


Figure 1: Course of effective heater voltage over time / state

Description:

The heater control shall be realised by use of a state machine. Each downstream oxygen sensor shall have its own state machine and each state machine shall run independently of the other.

The heater control shall be initialised to the LSH_OFF state at the start of a new driving cycle. Hence the oxygen sensor will not initially be electrically heated.

Heating shall commence should the activation conditions listed below be met. i.e. when the engine has transitioned from the start state (LV_ST_END = 1) and the heater management inhibit bit for the corresponding bank is not set (LV_INH_LSH_DOWN[i] = 0). The appropriate heater state shall change to LSH_POW_RISE and the initial value for the effective heater voltage shall be determined by obtaining the oxygen sensor temperature at start (TEMP_INI_LS_DOWN[i]) from input data. This shall then be used to obtain the appropriate effective heater starting voltage from map IP_V_EFC_INI_LSH_DOWN.

Should the state LSH_POW_RISE is switched from LSH_VB_PROT or LSH_TEMP_PROT, the effective heater voltage shall be incremented with the gradient of C_V_EFC_INC_PORT_LSH_DOWN. In the other cases, the effective heater voltage shall be incremented with the gradient of C_V_EFC_INC_LSH_DOWN. In order to prevent sensor from over heating when the sensor is already warm, C_V_EFC_INC_PORT_LSH_DOWN has to be set samller than C_V_EFC_INC_LSH_DOWN.

The continuation of the heating gradient shall depend on whether the dew point at the sensor location has been determined to have been exceeded, i.e. the temperature of the inner manifold wall at the sensor location has exceeded the temperature threshold where water in the exhaust gas may condense, as determined by LV_TEG_CAT_DOWN_MIN_THD[i].

Should the dew point not have been exceeded (LV_TEG_CAT_DOWN_MIN_THD[i] = 0) but timer T_POW_RISE_LSH_DOWN[i] has equalled or exceeded IP_T_TEMP_THD_LSH_DOWN, it shall be assumed that condensed water splash at the oxygen sensor location may occur. In order to protect the oxygen sensor by keeping the temperature of the ceramic lower than the specified critical temperature, the heater power shall be reduced immediately by setting the effective heater voltage to C_V_EFC_RED_LSH_DOWN. In this case, the heater state shall be set to LSH_POW_RED.

The reduced heating state shall persist until the dew point in the exhaust gas system has been determined to have been exceeded, i.e. until LV_TEG_CAT_DOWN_MIN_THD[i] = 1, at which point the heater state shall return to LSH_POW_RISE. The effective heater voltage shall be set to the same effective heater voltage as determined at start (IP_V_EFC_INI_LSH_DOWN), which may be modified by the calibration data C_V_EFC_STEP_LSH_DOWN, may be further incremented with a gradient of C_V_EFC_INC_LSH_DOWN without danger of damaging the oxygen sensor and the timer T_POW_RISE_LSH_DOWN[i] shall be reinitialised with the IP_T_TEMP_THD_LSH_DOWN determined at function start.

Should the initial exhaust gas temperature at start have been determined to exceed the dew point (LV_TEG_CAT_DOWN_MIN_THD[i] = 1), the heater control shall not enter the reduced heater power state but continue to incrementally increase the heater power within the LSH_POW_RISE state.

Should T_POW_RISE_LSH_DOWN[i] be determined to have equalled or exceeded the value C_T_MAX_POW_RISE_LSH_DOWN (irrespective of the dew point having been exceeded or not), the heater state shall change to LSH_POW_FALL and the timer shall be frozen.

In the LSH_POW_FALL state, the effective heater voltage shall be reduced with a gradient of C_V_EFC_DEC_LSH_DOWN until such time that it falls below or equals the value set for the open loop control at the current engine operating point. For cylinder bank 1, this value is determined by map IP_V_EFC_CTL_LSH_DOWN alone. For cylinder bank 2, this value is determined by IP_V_EFC_CTL_LSH_DOWN * IP_FAC_V_EFC_TEG_LSH_DOWN. The latter factor shall be necessary to accommodate differences in heat transfer rates between cylinder banks at the oxygen sensor location, particularly for transversal mounted V6-engines. At such time the effective heater voltage falls below or equals the respective threshold, the heater state shall change to LSH_POW_CTL.

If there has been a hot sensor restart i.e. TCO_ST >= (TCO_STOP - C_TCO_DIF_MAX_LSH_DOWN_REST) and VLS_DOWN moves outside the thresholds for readiness detection i.e. (VLS_DOWN > C_VLS_DOWN_AFR_THD_READY or VLS_DOWN < C_VLS_DOWN_AFL_THD_READY) the heater management shall leave "LSH_POW_RISE" and enter "LSH_POW_CTL" and the bit LV_LSH_DOWN_REST shall be set. This condition prevents sensors with powerful heaters (e.g. NTK UFLO 3.7 Ohm) and no closed loop heater capability from overheating during a hot start or repeated engine stop and start following a cold start. The constant C_TCO_ST_MIN_LSH_DOWN_REST allows this condition to be activated above a certain TCO_ST or to be deactivated completely for WRAF sensors / sensors with greater thermal inertia / weaker heater.

If LV_LSH_DOWN_REST = 1 the first C_CTR_R_IT_LS_DOWN_VLD_REST valid internal resistance values are checked and the effective heater voltage is offset by C_V_EFC_ADD_LSH_DOWN_REST whenever R_IT_LS_DOWN is less than C_R_IT_LS_DOWN_MIN_REST.

Should, however, the Boolean flag LV_TEG_CAT_DOWN_MIN_THD[i] be reset whilst in the state LSH_POW_FALL, the heater control shall resume at state LSH_POW_RED following the procedures outlined above.

The control of the effective heater voltage is carried out in open loop mode.

In the open loop heater control mode, the effective heater voltage shall be obtained from map IP_V_EFC_CTL_LSH_DOWN and is dependent upon the engine speed and the modelled exhaust gas temperature. For cylinder bank 2, an additional multiplicative factor (IP_FAC_V_EFC_TEG_LSH_DOWN) may be required for the reasons outlined above.

To the open loop control voltage a calibration system constant (C_V_EFC_AS_LSH_DOWN) can be added to permit a global change in the total effective heater voltage to be made. Thus the calculated effective heater voltage (V_EFC_CLC_LSH_DOWN[i]) shall be computed from the open loop value and the global correction value.

Furthermore should the sum of the components used to calculate the effective heater voltage be negative, LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be set. This shall prevent the controller from decreasing the effective heater voltage further but not prevent it from increasing the voltage. When the sum of the components is no longer negative, LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be reset. Similarly, should either of the overvoltage limitation Boolean flags LV_V_EFC_LIM_TOL_LSH_DOWN[i] & LV_V_EFC_LIM_MAX_LSH_DOWN[i] be set, as described later, the controller shall be prevented from increasing the effective heater voltage further but not prevent it from decreasing the voltage. When the overvoltage flags are no longer set and the other applicable conditions are met, the controller shall resume normal operation.

Should, however, the Boolean flag LV_TEG_CAT_DOWN_MIN_THD[i] be reset whilst in the state LSH_POW_CTL, then the open loop controller terms shall be reset, the Boolean flags to indicate open loop control and effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RED following the procedures outlined above.

Should at any time the battery voltage VB exceed the threshold C_VB_MAX_PROT_LSH_DOWN, as determined by the state of logical variable LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i], the oxygen sensor heater power shall be reduced by reducing the effective heater voltage used to calculate the driver duty cycle to C_V_EFC_PROT_VB_LSH_DOWN. The heater state shall change to LSH_VB_PROT on the next recurrence of the state machine and the effective heater voltage set to the aforementioned limit. This shall prevent damage to the sensor due to excessive heater voltages e.g. due to jump start from 24 V.

Should the battery voltage recover when in the state LSH_VB_PROT, as indicated by LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0, T_POW_RISE_LSH_DOWN[i] shall be reset, V_EFC_LSH_DOWN[i] shall be initialised to IP_V_EFC_INI_LSH_DOWN calculated after the start, the heater control shall resume at state LSH_POW_RISE following the procedures outlined above.

Short term voltage excursions shall cause the heater power to be limited but will only cause the state machine to change state should the voltage excursion persist until the next recurrence of the state machine. The state machine shall not be able to enter the LSH_VB_PROT state directly from LSH_OFF but shall be required to pass through LSH_POW_RISE.

The sensor shall also be protected from over-temperature. This shall be achieved by evaluating the value of the modelled exhaust gas temperature TEG_CAT_DOWN_MDL[i] in all states other than LSH_OFF. Should the exhaust gas temperature exceed the threshold C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN, the calculated effective heater voltage shall be reduced to 0, the state shall change to LSH_TEMP_PROT thus suspending open loop control, if active.

Should the exhaust gas temperature fall below the threshold, the state shall change dependent on the state active immediately prior to the over-temperature protection. If the state was LSH_POW_CTL, the state machine shall return to this state and the effective heater voltage shall take the new value as computed by the open loop control. If the state was any other than LSH_POW_CTL, the state shall change to LSH_POW_RISE. In this case T_POW_RISE_LSH_DOWN[i] shall be reset, V_EFC_LSH_DOWN[i] shall be initialised to IP_V_EFC_INI_LSH_DOWN calculated after the start, effective heater voltage limitation shall be reset and the heater control shall resume at state LSH_POW_RISE following the procedures outlined above.

Furthermore, in order to protect the sensor heater against long term high voltage stress it shall be necessary to limit the effective heater voltage should the value of V_EFC_CLC_LSH_DOWN[i] equal or exceed the threshold C_V_EFC_TOL_LSH_DOWN.

In this instance, the timer T_V_EFC_TOL_LSH_DOWN[i] shall be started. Should the overvoltage condition persist, such that this timer equals or exceeds the threshold C_T_MAX_V_EFC_TOL_LSH_DOWN, timer T_V_EFC_LIM_LSH_DOWN[i] shall be started and the Boolean flag LV_V_EFC_LIM_TOL_LSH_DOWN[i] shall be set. This flag shall prevent the controller from increasing effective the voltage further. If however the value of V_EFC_CLC_LSH_DOWN[i] should fall below the threshold C_V_EFC_TOL_LSH_DOWN at any time, LV_V_EFC_LIM_TOL_LSH_DOWN[i] shall be reset.

Once timer T_V_EFC_LIM_LSH_DOWN[i] has been started, Boolean flag LV_V_EFC_LIM_TOL_LSH_DOWN[i] shall be set or reset depending on the value of V_EFC_CLC_LSH_DOWN[i]. Once the timer T_V_EFC_LIM_LSH_DOWN[i] equals or exceeds threshold C_T_MAX_V_EFC_LIM_LSH_DOWN, both timers shall be reset and timer T_V_EFC_TOL_LSH_DOWN[i] must exceed the threshold C_T_MAX_V_EFC_TOL_LSH_DOWN again in order to permit further limiting to take place.

To prevent damage to the sensor, V_EFC_CLC_LSH_DOWN[i] shall not be permitted to exceed the lower of the values C_V_EFC_MAX_LSH_DOWN, C_V_EFC_MAX_ST_LSH_DOWN or VB dependent on the value of T_POW_RISE_LSH_DOWN[i].

If T_POW_RISE_LSH_DOWN[i] equals or exceeds the constant C_T_MAX_V_EFC_MAX_ST_LSH_DOWN or conditions regarding T_AST and STATE_LSH_DOWN[i] then V_EFC_CLC_LSH_DOWN[i] shall be checked against VB & C_V_EFC_MAX_LSH_DOWN. If either of the values is exceeded then Boolean flag LV_V_EFC_LIM_MAX_LSH_DOWN[i] shall be set and V_EFC_CLC_LSH_DOWN[i] limited to the lower of VB & C_V_EFC_MAX_LSH_DOWN otherwise LV_V_EFC_LIM_MAX_LSH_DOWN[i] shall be reset.

If T_POW_RISE_LSH_DOWN[i] is lower then constant C_T_MAX_V_EFC_MAX_ST_LSH_DOWN then V_EFC_CLC_LSH_DOWN[i] shall be checked against VB & C_V_EFC_MAX_ST_LSH_DOWN. If either of the values is exceeded then Boolean flag LV_V_EFC_LIM_MAX_LSH_DOWN[i] shall be set and V_EFC_CLC_LSH_DOWN[i] limited to the lower of VB & C_V_EFC_MAX_ST_LSH_DOWN otherwise LV_V_EFC_LIM_MAX_LSH_DOWN[i] shall be reset.

The flag LV_V_EFC_LIM_MAX_LSH_DOWN[i] shall prevent the controller from increasing effective the voltage further and permit the final effective heater voltage to be limited, as described later.

Should timer T_V_EFC_LIM_LSH_DOWN[i] be unequal to zero, i.e. overvoltage no longer tolerated as it has been present for certain time, V_EFC_LSH_DOWN[i] shall be the lower of the values V_EFC_CLC_LSH_DOWN[i] & C_V_EFC_LIM_LSH_DOWN otherwise V_EFC_LSH_DOWN[i] shall be set to V_EFC_CLC_LSH_DOWN[i].

Once the effective heater voltage has been determined throughout the various heater states other than LSH_OFF, it shall be used to compute the PWM duty cycle (LSHPWM_DOWN[i]) that shall control the appropriate heater driver. The heater power shall be corrected to take into account deviations in the measured battery voltage and limited to the range bounded by NC_LSHPWM_TOL_LSH_DOWN & NC_LSHPWM_BOL_LSH_DOWN. The evaluation of excess battery voltage shall be determined at the same recurrence as the computation of the duty cycle as described above. The PWM value may also be modified by use of constant C_FAC_V_EFC_AS_LSH_DOWN.

Should at any time an inhibit flag be set (LV_INH_LSH_DOWN[i] = 1), the corresponding oxygen sensor heater shall be switched off and the STATE_LSH_DOWN[i] set to LSH_OFF. This shall allow project specific application conditions to turn off oxygen sensor heating according to project philosophy. Should the heater function be inhibited, then the PWM duty cycle shall be set to input LSHPWM_EXT_LS_DOWN[i]. On a transition of the inhibit bit LV_INH_LSH_DOWN[i] from 1 $\rightarrow$ 0, the heater function shall start from anew.

<u>Application conditions:</u>

Activation:

At any engine operating state.

Deactivation:

-

Time recurrence:

The state machine, sensor protection and effective heater voltage definition shall be carried out once every 100 ms. T_SAMPLE_1 = 100ms

The determination of the oxygen sensor heater driver duty cycle and associated excessive battery voltage protection shall be carried out once every 10 ms. T_SAMPLE_2 = 10ms

Initialisation:

The following variable initialisation shall take place at the beginning of a new driving cycle:

STATE_LSH_DOWN[i] = LSH_OFF

T_POW_RISE_LSH_DOWN[i] = 0

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

LSHPWM_DOWN[i] = 0

V_EFC_LSH_DOWN[i] = 0

V_EFC_CLC_LSH_DOWN[i] = 0

V_EFC_ADD_LSH_DOWN_REST = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

LV_LSH_DOWN_REST = 0

NOTE: Projects not making use of the inhibit flags LV_INH_LSH_DOWN[i] shall ensure that they are initialised with and remain 0, thus never disabling the oxygen sensor heater function or thus never disabling the oxygen sensor heater open loop control function respectively

<u>Formula section:</u>

7.37.1 Heater management state machine

The state machine shall remain in its current state and carry out the actions specified to occur within that state once per recurrence unless otherwise specified. The state machine shall only move to another state when one of the conditions has been determined to be met.

Note: The priorities of the conditions to change between states shall be defined by the order in which these conditions are listed within the appropriate state as described below.

HEATER STATE DIAGRAM

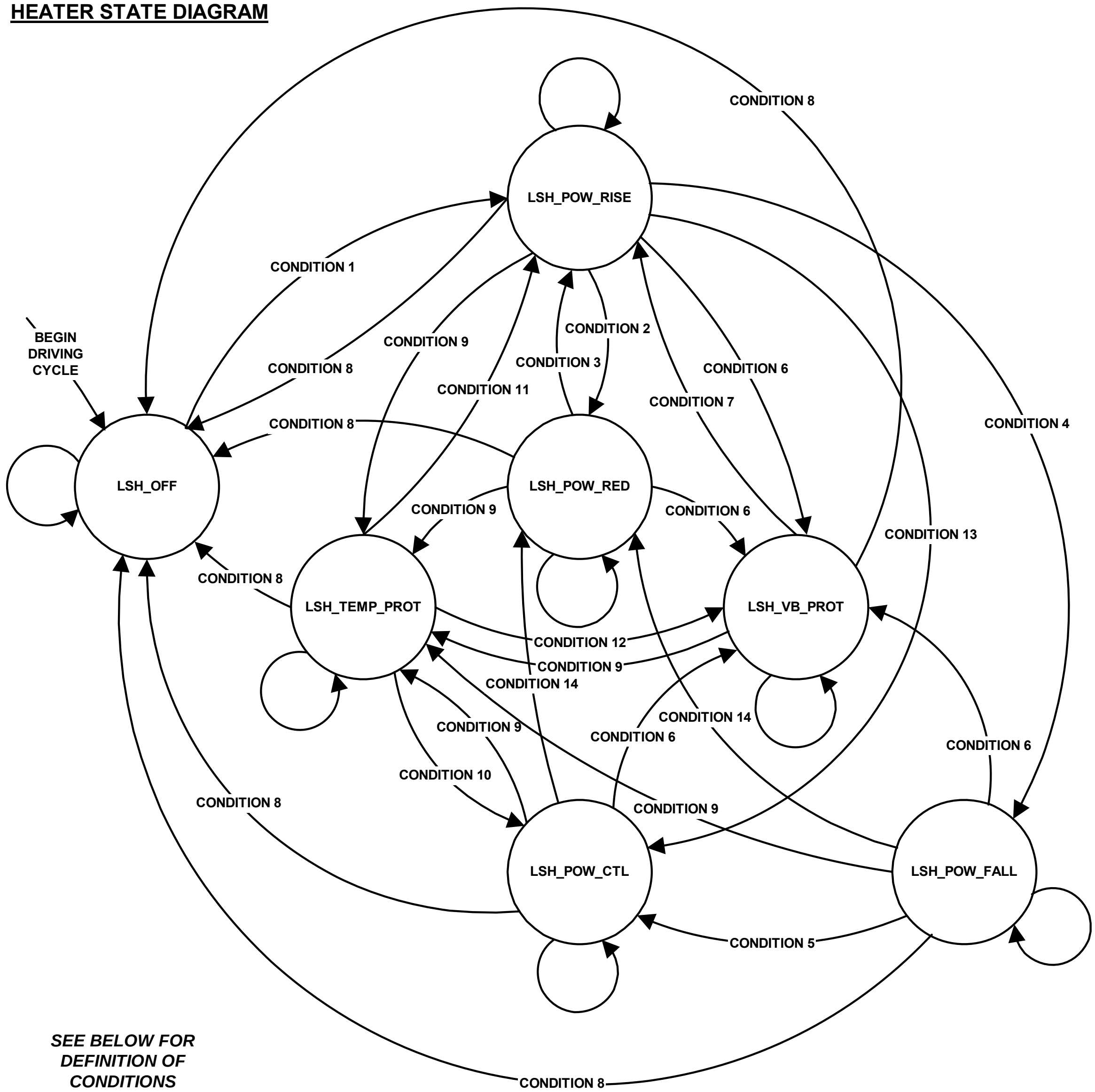


Figure 2: Heater management state diagram

STATE_LSH_DOWN[i] "LSH_OFF"

Actions:

T_POW_RISE_LSH_DOWN[i] = 0

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

V_EFC_LSH_DOWN[i] = 0

V_EFC_CLC_LSH_DOWN[i] = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Note: The above listed actions must be carried out at least once when the state has been entered for the first time or from any other state! The action below must be executed on every recurrence while in this state.

Condition 1:"LSH_OFF to LSH_POW_RISE"

LV_ST_END = 1 **&**

LV_INH_LSH_DOWN[i] = 0

Transition actions:

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_INI_LSH_DOWN (TEMP_INI_LS_DOWN[i])

Timer T_POW_RISE_LSH_DOWN[i] shall be started

STATE_LSH_DOWN[i] = LSH_POW_RISE

STATE_LSH_DOWN[i] "LSH_POW_RISE"

Actions:

If Previous state of STATE_LSH_DOWN[i] = LSH_TEMP_PROT or LSH_VB_PROT

Then $V\_EFC\_CLC\_LSH\_DOWN[i]_{N} = V\_EFC\_CLC\_LSH\_DOWN[i]_{N-1} +$ C_V_EFC_INC_PROT_LSH_DOWN*T_SAMPLE_1

Else $V\_EFC\_CLC\_LSH\_DOWN[i]_{N} = V\_EFC\_CLC\_LSH\_DOWN[i]_{N-1} +$ C_V_EFC_INC_LSH_DOWN* T_SAMPLE_1

Endif

Increment timer T_POW_RISE_LSH_DOWN[i]

Note: See section "Oxygen sensor heater voltage protection" for notes on limiting the applied heater voltage.

Condition 8:"LSH_POW_RISE to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

Condition 9:"LSH_POW_RISE to LSH_TEMP_PROT"

TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_TEMP_PROT

Condition 6:"LSH_POW_RISE to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_VB_PROT

Condition 2:"LSH_POW_RISE to LSH_POW_RED"

T_POW_RISE_LSH_DOWN[i] ≥ IP_T_TEMP_THD_LSH_DOWN (TEMP_INI_LS_DOWN[i]) **&**

LV_TEG_CAT_DOWN_MIN_THD[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_POW_RED

Condition 13:"LSH_POW_RISE to LSH_POW_CTL"

(TCO_ST > C_TCO_ST_MIN_LSH_DOWN_REST **&**
TCO_ST >= TCO_STOP - C_TCO_DIF_MAX_LSH_DOWN_REST **&**
(VLS_DOWN > C_VLS_DOWN_AFR_THD_READY **or** VLS_DOWN < C_VLS_DOWN_AFL_THD_READY))
Transition actions:

LV_LSH_DOWN_REST = 1 *%bit set if transition to "LSH_POW_CTL" caused by the LSH_DOWN_REST conditions.*

Stop timer T_POW_RISE_LSH_DOWN[i] and freeze value

STATE_LSH_DOWN[i] = LSH_POW_CTL

Condition 4:"LSH_POW_RISE to LSH_POW_FALL"

T_POW_RISE_LSH_DOWN[i] ≥ C_T_MAX_POW_RISE_LSH_DOWN + IP_T_TEMP_THD_LSH_DOWN

% IP_ ... (TEMP_INI_LS_DOWN[i])

Transition actions:

Stop timer T_POW_RISE_LSH_DOWN[i] and freeze value

STATE_LSH_DOWN[i] = LSH_POW_FALL

STATE_LSH_DOWN[i] "LSH_POW_RED"

Actions:

V_EFC_CLC_LSH_DOWN[i] = C_V_EFC_RED_LSH_DOWN

Condition 8:"LSH_POW_RED to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

<u>Condition 9</u>:*"LSH_POW_RED to LSH_TEMP_PROT"*

TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_TEMP_PROT

<u>Condition 6</u>:*"LSH_POW_RED to LSH_VB_PROT"*

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_VB_PROT

<u>Condition 3</u>:*"LSH_POW_RED to LSH_POW_RISE"*

LV_TEG_CAT_DOWN_MIN_THD[i] = 1

Transition actions:

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_INI_LSH_DOWN

+ C_V_EFC_STEP_LSH_DOWN

% IP_ ... (TEMP_INI_LS_DOWN[i]) *same value as at start*

T_POW_RISE_LSH_DOWN[i] = IP_T_TEMP_THD_LSH_DOWN

% TEMP_INI_LS_DOWN[i] *same value as at start*

STATE_LSH_DOWN[i] = LSH_POW_RISE

Note: See section "Oxygen sensor heater voltage protection" for notes on limiting the applied heater voltage. The function shall prevent the effective heater voltage from becoming negative by limiting to 0 where necessary.

STATE_LSH_DOWN[i] "LSH_POW_FALL"

Actions:

$V\_EFC\_CLC\_LSH\_DOWN[i]_N$ = $V\_EFC\_CLC\_LSH\_DOWN[i]_{N-1}$ - C_V_EFC_DEC_LSH_DOWN *

T_SAMPLE_1

Note: See section "Oxygen sensor heater voltage protection" for notes on limiting the applied heater voltage. The function shall prevent the effective heater voltage from becoming negative by limiting to 0 where necessary.

<u>Condition 8</u>:*"LSH_POW_FALL to LSH_OFF"*

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

<u>Condition 9</u>:*"LSH_POW_FALL to LSH_TEMP_PROT"*

TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_TEMP_PROT

Condition 6:"LSH_POW_FALL to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_VB_PROT

Condition 14:"LSH_POW_FALL to LSH_POW_RED"

LV_TEG_CAT_DOWN_MIN_THD[i] = 0

Transition actions:

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

STATE_LSH_DOWN[i] = LSH_POW_RED

Condition 5:"LSH_POW_FALL to LSH_POW_CTL"

For bank 1; i = 1

V_EFC_LSH_DOWN[i] ≤ IP_V_EFC_CTL_LSH_DOWN (N_32, TEG_CAT_DOWN_MDL[i])

For bank 2; i = 2

V_EFC_LSH_DOWN[i] ≤ IP_V_EFC_CTL_LSH_DOWN (N_32, TEG_CAT_DOWN_MDL[i])
* IP_FAC_V_EFC_TEG_LSH_DOWN

Transition actions:

STATE_LSH_DOWN[i] = LSH_POW_CTL

STATE_LSH_DOWN[i] "LSH_POW_CTL"

Actions:

Compute effective heater voltage from open loop control and closed loop delta voltages:

```
If
LV_LSH_DOWN_REST = 1 and
CTR_CYCNR_R_IT_LS_DOWN_VLD < C_CTR_R_IT_LS_DOWN_VLD_REST then
    If
    R_IT_LS_DOWN < C_R_IT_LS_DOWN_MIN_REST
        Then
        V_EFC_ADD_LSH_DOWN_REST = C_V_EFC_ADD_LSH_DOWN_REST
        Else
        V_EFC_ADD_LSH_DOWN_REST = 0
    Endif
Else
V_EFC_ADD_LSH_DOWN_REST = 0
Endif
```

For Bank 1; i = 1

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_CTL_LSH_DOWN + C_V_EFC_AS_LSH_DOWN + V_EFC_ADD_LSH_DOWN_REST

% IP_ ... (N_32, TEG_CAT_DOWN_MDL[i])

Note: If (IP_V_EFC_CTL_LSH_DOWN + C_V_EFC_AS_LSH_DOWN) < 0 , i.e. would cause the effective heater voltage to be negative, V_EFC_CLC_LSH_DOWN[i] shall be limited to 0 and LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be set, until the sum less than zero condition is no longer met at which point LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be reset.

For Bank 2; i = 2

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_CTL_LSH_DOWN * IP_FAC_V_EFC_TEG_LSH_DOWN

+ C_V_EFC_AS_LSH_DOWN + V_EFC_ADD_LSH_DOWN_REST

% IP_ ... (N_32, TEG_CAT_DOWN_MDL[i])

Note: If (IP_V_EFC_CTL_LSH_DOWN * IP_FAC_V_EFC_TEG_LSH_DOWN + C_V_EFC_AS_LSH_DOWN) < 0 , i.e. would cause the effective heater voltage to be negative, V_EFC_CLC_LSH_DOWN[i] shall be limited to 0 and LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be set, until the sum less than zero condition is no longer met at which point LV_V_EFC_LIM_BOL_LSH_DOWN[i] shall be reset.

Condition 8:"LSH_POW_CTL to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

Condition 9:"LSH_POW_CTL to LSH_TEMP_PROT"

TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_TEMP_PROT

Condition 6:"LSH_POW_CTL to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_VB_PROT

Condition 14:"LSH_POW_CTL to LSH_POW_RED"

LV_TEG_CAT_DOWN_MIN_THD[i] = 0

Transition actions:

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

STATE_LSH_DOWN[i] = LSH_POW_RED

STATE_LSH_DOWN[i] "LSH_VB_PROT"

Actions:

V_EFC_CLC_LSH_DOWN[i] = C_V_EFC_PROT_VB_LSH_DOWN

Condition 8:"LSH_VB_PROT to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

Condition 9:"LSH_VB_PROT to LSH_TEMP_PROT"

TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

STATE_LSH_DOWN[i] = LSH_TEMP_PROT

Condition 7:"LSH_VB_PROT to LSH_POW_RISE"

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

Transition actions:

T_POW_RISE_LSH_DOWN[i] = 0

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_INI_LSH_DOWN
+ C_V_EFC_ADD_VB_PROT_LSH_DOWN
% IP_ ... (TEMP_INI_LS_DOWN[i]) *same value as at start*

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

STATE_LSH_DOWN[i] = LSH_POW_RISE

STATE_LSH_DOWN[i] "LSH_TEMP_PROT"

Actions:

V_EFC_CLC_LSH_DOWN[i] = 0

Condition 8:"LSH_TEMP_PROT to LSH_OFF"

LV_ST_END = 0 **or** LV_INH_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_OFF

Condition 12:"LSH_TEMP_PROT to LSH_VB_PROT"

LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

Transition actions:

STATE_LSH_DOWN[i] = LSH_VB_PROT

Condition 10:"LSH_TEMP_PROT to LSH_POW_CTL"

TEG_CAT_DOWN_MDL[i] ≤ C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

Previous state of STATE_LSH_DOWN[i] = LSH_POW_CTL

Transition actions:

STATE_LSH_DOWN[i] = LSH_POW_CTL

Condition 11:"LSH_TEMP_PROT to LSH_POW_RISE"

TEG_CAT_DOWN_MDL[i] ≤ C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN **&**

Previous state of STATE_LSH_DOWN[i] ≠ LSH_POW_CTL

Transition actions:

T_POW_RISE_LSH_DOWN[i] = 0

V_EFC_CLC_LSH_DOWN[i] = IP_V_EFC_INI_LSH_DOWN (TEMP_INI_LS_DOWN[i])

+ C_V_EFC_ADD_TEMP_PROT_LSH_DOWN

% IP_ ... (TEMP_INI_LS_DOWN[i]) *same value as at start*

T_V_EFC_TOL_LSH_DOWN[i] = 0

T_V_EFC_LIM_LSH_DOWN[i] = 0

LV_V_EFC_LIM_BOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0

STATE_LSH_DOWN[i] = LSH_POW_RISE

Note: Should none of the conditions have been determined to be met, the state machine shall remain in the same state.

7.37.2 Oxygen sensor heater voltage protection

Prolonged exposure to heater voltages that exceed the absolute maximum ratings specified shall be prevented. The preventative measures shall be applicable to all heater states other than LSH_OFF.

If STATE_LSH_DOWN[i] ≠ LSH_OFF

then **If** V_EFC_CLC_LSH_DOWN[i] ≥ C_V_EFC_TOL_LSH_DOWN

then **If** T_V_EFC_TOL_LSH_DOWN[i] ≥ C_T_MAX_V_EFC_TOL_LSH_DOWN

then **If** T_V_EFC_LIM_LSH_DOWN[i] ≥ C_T_MAX_V_EFC_LIM_LSH_DOWN

then LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0

Reset timer T_V_EFC_LIM_LSH_DOWN[i]

Reset timer T_V_EFC_TOL_LSH_DOWN[i]

```
                    else    LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 1
                            Increment timer T_V_EFC_LIM_LSH_DOWN[i]
                    endif
            else    LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0
                    Reset timer T_V_EFC_LIM_LSH_DOWN[i]
                    Increment timer T_V_EFC_TOL_LSH_DOWN[i]
            endif
    else    LV_V_EFC_LIM_TOL_LSH_DOWN[i] = 0
            If      T_V_EFC_LIM_LSH_DOWN[i] = 0
            or      T_V_EFC_LIM_LSH_DOWN[i] ≥
                            C_T_MAX_V_EFC_LIM_LSH_DOWN
            then    Reset timer T_V_EFC_LIM_LSH_DOWN[i]
                    Reset timer T_V_EFC_TOL_LSH_DOWN[i]
            else    Increment timer T_V_EFC_LIM_LSH_DOWN[i]
            endif
    endif
    If      T_POW_RISE_LSH_DOWN[i] ≥
                    C_T_MAX_V_EFC_MAX_ST_LSH_DOWN
    or      (T_AST ≥ C_T_AST_MAX_LSH_DOWN
    and     STATE_LSH_DOWN[i] = "LSH_POW_CTL")
    then    If      V_EFC_CLC_LSH_DOWN[i] > VB
            or      V_EFC_CLC_LSH_DOWN[i] > C_V_EFC_MAX_LSH_DOWN
            then    LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 1
                    V_EFC_CLC_LSH_DOWN[i] =
                                    MIN (VB; C_V_EFC_MAX_LSH_DOWN)
            else    LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0
            endif
    else    If      V_EFC_CLC_LSH_DOWN[i] > VB
            or      V_EFC_CLC_LSH_DOWN[i] > C_V_EFC_MAX_ST_LSH_DOWN
            then    LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 1
                    V_EFC_CLC_LSH_DOWN[i] =
                                    MIN (VB, C_V_EFC_MAX_ST_LSH_DOWN)
            else    LV_V_EFC_LIM_MAX_LSH_DOWN[i] = 0
            endif
    endif
    If      T_V_EFC_LIM_LSH_DOWN[i] <> 0
```

then V_EFC_LSH_DOWN[i] =

MIN (V_EFC_CLC_LSH_DOWN[i], C_V_EFC_LIM_LSH_DOWN)

else V_EFC_LSH_DOWN[i] = V_EFC_CLC_LSH_DOWN[i]

endif

endif

7.37.3 Oxygen sensor heater driver duty cycle computation

The PWM duty cycle for the appropriate heater driver shall continually be calculated in all heater states at the specified recurrence.

If STATE_LSH_DOWN[i] = LSH_OFF

then LSHPWM_DOWN[i] = LSHPWM_EXT_LS_DOWN[i]

else **If** VB ≥ C_VB_MAX_PROT_LSH_DOWN

then LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 1

LSHPWM_DOWN[i] = $C\_V\_EFC\_PROT\_VB\_LSH\_DOWN^2 * 100 / VB_N^2$

else LV_V_EFC_LIM_PROT_VB_LSH_DOWN[i] = 0

LSHPWM_DOWN[i] = $V\_EFC\_LSH\_DOWN[i]_N^2$ *

C_FAC_V_EFC_AS_LSH_DOWN *100 / VB_N^2

If LSHPWM_DOWN[i] ≥ NC_LSHPWM_TOL_LSH_DOWN

then LSHPWM_DOWN[i] = NC_LSHPWM_TOL_LSH_DOWN

else **If** LSHPWM_DOWN[i] ≤ NC_LSHPWM_BOL_LSH_DOWN

then LSHPWM_DOWN[i] = NC_LSHPWM_BOL_LSH_DOWN

else LSHPWM_DOWN[i] = LSHPWM_DOWN[i]

endif

endif

endif

endif.

The value C_FAC_V_EFC_AS_LSH_DOWN is a correction signal generated in the application system.

NOTE: The recurrence of the heater driving duty cycle shall occur at the same recurrence as the Vbatt acquisition. This shall prevent short duration voltage drop-outs during engine start from causing long duration Vbatt corrections being made where the Vbatt has since recovered. This may otherwise cause excessive current to flow in the heater driver.

The computation of the duty cycle shall be checked to ensure that the result remains positive (i.e. ≥ 0) at all times and that now under- or overflows are caused by the values of the calibration system constants.

The multiplication of the corrected voltage ratio by 100 to obtain a unit of percent (%) may not necessarily be implemented in the SW but in the data bank definition of the variables.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_V_EFC_INI_LSH_DOWN	1*6	0 ... FFH	0 ... 26	0.102	V
LDPM_TEMP_INI_LS_DOWN_1_EGCP	6	0000 ... FFFFH	-2048...2047.94	62.5e-3	°C
Initial effective heater voltage					
IP_V_EFC_CTL_LSH_DOWN	6*6	0...FFH	0...26	0.102	V
LDPM_N_32_3_EGCP	6	0 ... FFH	0 ... 8160	32	rpm
LDPM_TEG_CAT_DOWN_MDL_1_EGCP	6	0...7FF0H	0 ... 2047	0.0625	°C
Open loop control effective heater voltage					
IP_T_TEMP_THD_LSH_DOWN	1*6	0 ... FFFFH	0 ... 6553.5	0.1	s
LDPM_TEMP_INI_LS_DOWN_1_EGCP	6	0000 ... FFFFH	-2048...2047.94	62.5e-3	°C
Duration until critical water splash sensor temperature reached					
IP_FAC_V_EFC_TEG_LSH_DOWN	1*6	0...FFH	0...4	0.015625	-
LDPM_VS_1_EGCP	6	0 ... FFH	0 ... 255	1	km/h
Factor for the heat transfer rate, bank 2 only					
C_V_EFC_INC_LSH_DOWN	1	0...FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage increment to raise temperature					
C_V_EFC_INC_PROT_LSH_DOWN	1	0...FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage increment to raise temperature in case of transition from TEMP_PROT or VB_PROT					
C_V_EFC_DEC_LSH_DOWN	1	0 ... FFH	0 ... 1.625	6.373e-3	V / 100ms
Effective heater voltage decrement to lower temperature					
C_V_EFC_RED_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Reduced effective heater voltage during danger of water splash damage					
C_V_EFC_STEP_LSH_DOWN	1	80 ... 7FH	-13.05 ... 12.95	0.102	V
Additional effective heater voltage step on transition from LSH_POW_RED to LSH_POW_RISE					
C_T_MAX_POW_RISE_LSH_DOWN	1	0 ... FFFFH	0 ... 6553.5	0.1	s
Duration of open loop pre-heating in LSH_POW_RISE from dew-point to set-point temperature					
C_T_MAX_V_EFC_TOL_LSH_DOWN	1	0 ... FFH	0 ... 25.5	0.1	s
Maximum permitted duration where excessive effective heater voltage tolerated					
C_T_MAX_V_EFC_LIM_LSH_DOWN	1	0 ... FFH	0 ... 25.5	0.1	s
Maximum duration of effective heater voltage limiting					
C_T_AST_MAX_LSH_DOWN	1	0 ... FFFFH	0 ... 6553.5	0.1	s
Time after start at which step in max. effective heater voltage applied, if LSH_POW_CTL active (T_POW_RISE_LSH_DOWN[i] stopped)					
C_T_MAX_V_EFC_MAX_ST_LSH_DOWN	1	0 ... FFH	0 ... 25.5	0.1	s
Time after start at which step in max. effective heater voltage applied					
C_V_EFC_TOL_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage threshold above which voltage limiting may occur if threshold persistently exceeded					
C_V_EFC_LIM_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage limit under persistent excessive heater voltage condition					
C_VB_MAX_PROT_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Maximum permitted battery voltage threshold, over which overvoltage detected					
C_V_EFC_PROT_VB_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Effective heater voltage in the case of battery voltage overvoltage condition					
C_V_EFC_MAX_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Absolute maximum permitted effective heater voltage after initial phase; threshold & limit					
C_V_EFC_MAX_ST_LSH_DOWN	1	0 ... FFH	0 ... 26	0.102	V
Absolute maximum permitted effective heater voltage during initial phase; threshold & limit					
C_V_EFC_AS_LSH_DOWN	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage calibration system additive factor					
C_V_EFC_ADD_LSH_DOWN_REST	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage reduction in case of active LSH_REST and low R_IT.					
C_V_EFC_ADD_TEMP_PROT_LSH_DOWN	1	80 ... 7FH	-13.05 ...12.95	0.102	V
Effective heater voltage reduction in case of transition from TEMP_PROT to POW_RISE					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_R_IT_LS_DOWN_MIN_REST	1	0 ... FFFFH	0 ... 65535	1	Ω
Minimum R_IT after LSH_REST below which LSH voltage is reduced.					
C_CTR_R_IT_LS_DOWN_VLD_REST	1	0 ... FFFFH	0 ... 65535	1	-
Number of valid R_IT values checked after LSH_REST.					
C_FAC_V_EFC_AS_LSH_DOWN	1	0 ... FFH	0 ... 1.9922	7.8125e-3	-
Multiplicative correction factor to effective heater voltage					
C_TEG_CAT_DOWN_MDL_MAX_LSH_DOWN	1	0...7FF0H	0 ... 2047	62.5e-3	°C
Maximum permitted exhaust gas temperature above which LSH switched off					
C_TCO_DIF_MAX_LSH_DOWN_REST	1	0...FEH	-48 ... 142.5	0.75	°C
Maximum TCO decrease since last engine stop for hot sensor restart detection					
C_TCO_ST_MIN_LSH_DOWN_REST	1	0...FEH	-48 ... 142.5	0.75	°C
Minimum TCO_ST to activate hot sensor restart detection					
C_V_EFC_ADD_VB_PROT_LSH_DOWN	1	80…7FH	-13.05…12.95	0.102	V
Effective heater voltage reduction in case of transition from VB_PROT to POW_RISE					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_LSHPWM_TOL_LSH_DOWN	1	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle upper limit					
NC_LSHPWM_BOL_LSH_DOWN	1	0 ... FFH	0 ... 99.61	0.391	%
Heater driver PWM duty cycle lower limit					

7.37.4 Application incidences for downstream oxygen sensor heater management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for heating not met					
TEMP_INI_LS_DOWN[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2048...2047.938	62.5e-3	°C
Oxygen sensor temperature at start					

Input data:

		LV_ST_END	LV_LSHPWM_DOWN_EXT_ADJ
NC_CBK_EX_NR	TEG_DYN_LS_DOWN[NC_CBK_EX_NR]		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Description:

The Boolean constant LC_LSH_DOWN[i] shall permit the appropriate oxygen sensor to be heated. When set, the Boolean inhibit flag LV_INH_LSH_DOWN[i] shall be reset. When reset, LV_INH_LSH_DOWN[i] shall be set.

The oxygen sensor temperature at start TEMP_INI_LS_DOWN[i] shall be obtained from the value TEG_DYN_LS_DOWN out of the model in EXTD.

Application conditions:

Activation:

LV_ST_END = 1

Deactivation:

LV_ST_END = 0

Time recurrence:

Once every 1 s.

Initialisation:

The following variable initialisation shall take place at the beginning of a new driving cycle.

LV_INH_LSH_DOWN[i] = 0
TEMP_INI_LS_DOWN[i] = TEG_DYN_LS_DOWN[i]

Formula section:

If LV_LSHPWM_DOWN_EXT_ADJ = 1

Then LV_INH_LSH_DOWN[i] = 1

Else LV_INH_LSH_DOWN[i] = **!** LC_LSH_DOWN[i]

Endif

Once at every LV_ST_END 0-> 1

TEMP_INI_LS_DOWN[i] = TEG_DYN_LS_DOWN[i]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_LSH_DOWN[NC_CBK_EX_NR]	1	0...1H	0...1	1	-
Boolean variable to enable oxygen sensor heating					

7.38 Air-Fuel cycle evaluation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_AFL[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating detection of lean Air-Fuel mixture when set					
T_AFL_CYC[NC_CBK_EX_NR]	V/O	0...FFFFH	0...655.35	0.01	s
Lean mixture cycle time					
T_AFL_CYC_HLD[NC_CBK_EX_NR]	V/O	0...FFFFH	0...655.35	0.01	s
Lean mixture cycle time, held until next new value available					
T_AFR_CYC[NC_CBK_EX_NR]	V/O	0...FFFFH	0...655.35	0.01	s
Rich mixture cycle time					
T_AFL_LOCK[NC_CBK_EX_NR]	V	0...FFH	0...2.55	0.01	s
Timer indicating duration of LV_AFL[i] lock					
CTR_AFL_CYC[NC_CBK_EX_NR]	V/O	0...FFFFH	0...65535	1	-
Counter indicating number of air fuel cycles					

Input data:

LV_ST_END	LV_ES	MAF	N_32
LV_LAM_LSCL[NC_CBK_EX_NR]		LV_LS_UP_READY[NC_CBK_EX_NR]	
VLS_UP[NC_CBK_EX_NR]		NC_CBK_EX_NR	

FUNCTION DESCRIPTION:

General information:

The function shall evaluate the oxygen sensor signal voltage and compute a number of resultant values that may be used by other functions. These values shall include:

1. Flag indicating whether Air-Fuel Ratio is currently lean or rich
2. Cycle times for lean and rich periods
3. Timer indicating duration of LV_AFL[i] lock-out
4. Counter indicating number of complete AF cycles since last lambda controller activation

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
  i = 1, for exhaust cylinder bank 1
  i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
  i = 1, for single exhaust cylinder bank

Signal flow diagram:

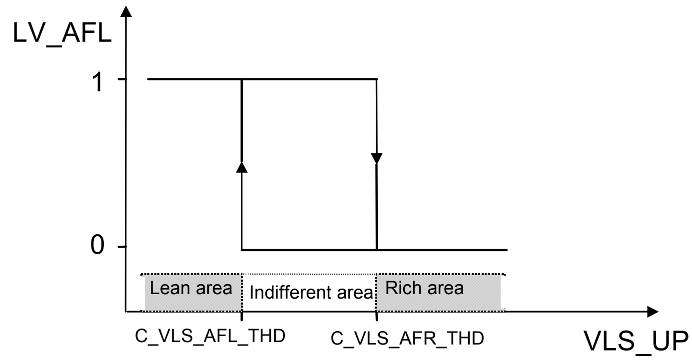


Figure 1 Diagram showing state and hysteresis of LV_AFL[i] as function of VLS_UP[i]

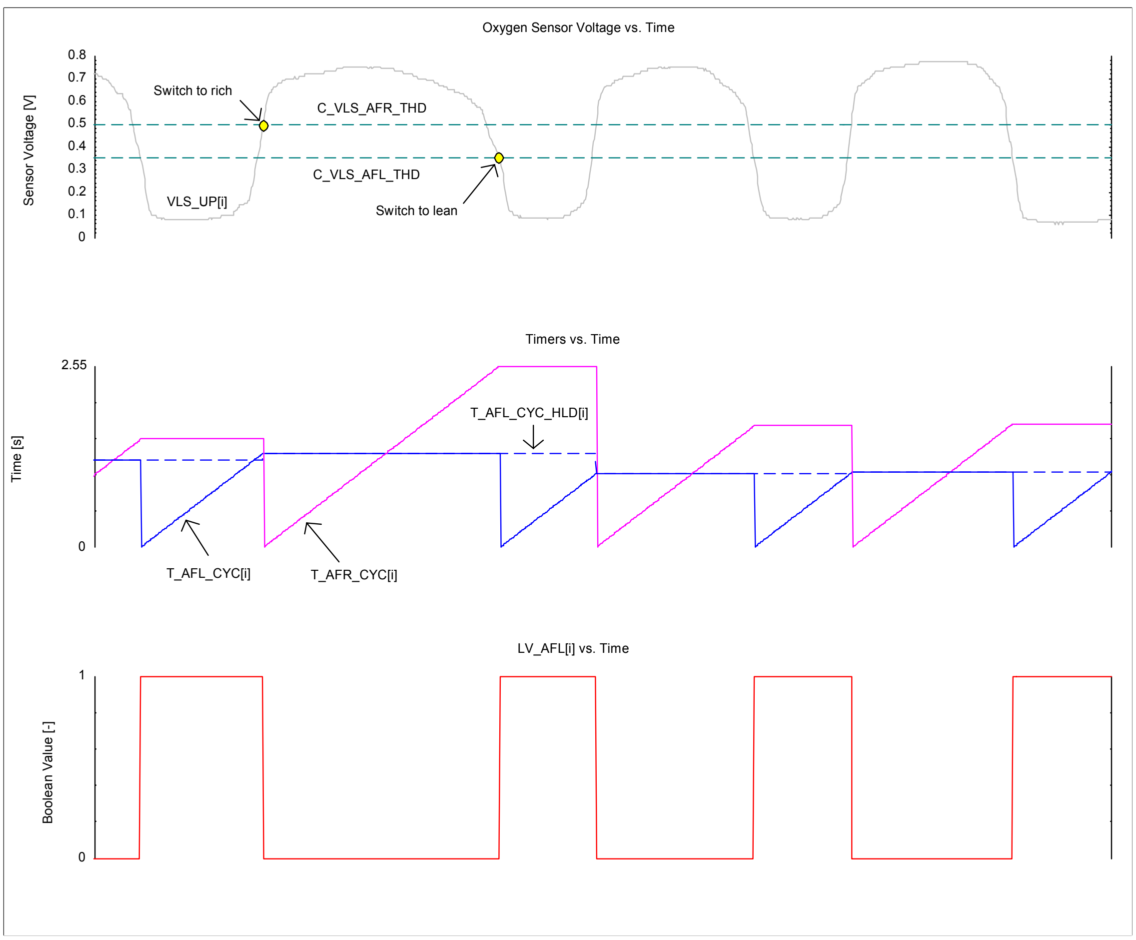


Figure 2 Physical relationship between values, Air-Fuel cycle evaluation

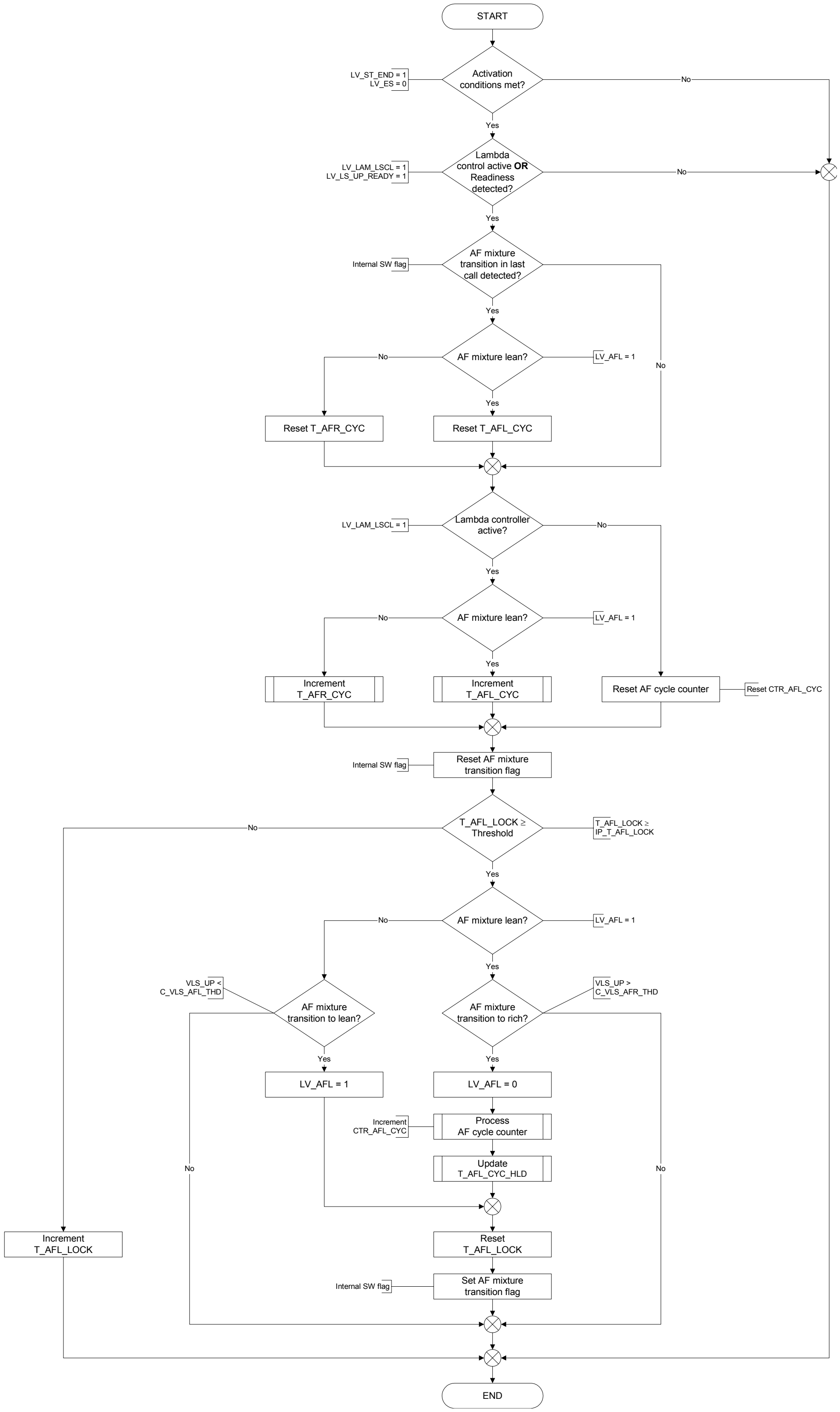


Figure 3 Function flow chart, Air-Fuel cycle evaluation

Description:

The function shall provide an indication of the current Air-Fuel mixture ratio, i.e. mixture lean or rich, and determine the lean and rich cycle time separately for further evaluation by other functions. This shall be carried out as follows:

The function shall be activated should the engine state transition out of start (LV_ST_END = 1) and shall be deactivated should the engine stop state (LV_ES = 1) be entered.

Two further activation conditions apply to the function: The lambda controller shall be active (LV_LAM_LSCL[i] = 1) or the oxygen sensor shall be determined to be in a state of operative readiness (LV_LS_UP_READY[i]).

Once the activation conditions have been met the function may be described in three blocks:

1. Resetting of lean / rich cycle timers: Once a AF mixture transition has been detected elsewhere in the function, the respective timer shall be reset dependent on the current state of the AF mixture, as determined by LV_AFL[i] (rich to lean transition; reset T_AFL_CYC[i], lean to rich transition; reset T_AFR_CYC[i]).

2. Timer / counter incrementation: Should the lambda controller be active, as determined by LV_LAM_LSCL[i] = 1, the function shall increment the respective timer dependent on the current state of the AF mixture, as determined by LV_AFL[i] (lean mixture; increment T_AFL_CYC[i], rich mixture; increment T_AFR_CYC[i]). Should at any time the lambda controller be deactivated, the function shall reset the counter CTR_AFL_CYC[i].

3. AF transition detection: The function shall detect a change in AF mixture from lean to rich or vice versa ignoring possible jitter in the oxygen sensor signal. This shall be facilitated by the following method: If the timer T_AFL_LOCK[i] falls below the threshold IP_T_AFL_LOCK, the timer shall be incremented and no further action shall take place in the current recurrence. Should, during following recurrence cycles, the timer equal or exceed the threshold and if the AF mixture flag is currently indicating lean, the function shall observe the VLS_UP[i] signal for a change to rich as indicated by VLS_UP[i] exceeding the threshold C_VLS_AFR_THD. Once this has occurred, the LV_AFL[i] flag shall be changed to indicate a rich mixture (LV_AFL[i] = 0), a counter CTR_AFL_CYC[i] shall be incremented to indicate the number of complete AF cycles since the last activation of the lambda controller, the output data T_AFL_CYC_HLD[i] shall be updated with T_AFL_CYC[i], the timer T_AFL_LOCK[i] shall be reset and a SW internal flag set to indicate the AF mixture transition. The same procedure, except incrementation of counter CTR_AFL_CYC[i], applies for a currently rich mixture. A transition to a lean mixture shall be detected by VLS_UP[i] falling below threshold C_VLS_AFL_THD.

Remark: The timer T_AFL_LOCK[i] and associated engine operating point dependent threshold IP_T_AFL_LOCK reduce the effect of rapid oxygen sensor signal changes, noise generated by cylinder to cylinder AF mixture deviations, that would cause the lambda controller to execute a P-jump. The mapped lock-out time represents the delay between injection of the AF mixture and the burnt AF mixture reaching and being detected by the oxygen sensor. The lock-out time shall not exceed the exhaust gas delay for any particular engine operating point otherwise the lambda controller frequency will be affected. Oxygen sensor signal changes occurring within this delay time may not be considered to be due to the change in controlled lambda but from noise and shall be ignored.

The above three blocks shall be cycled through once every function recurrence.

Application conditions:

Recurrence:

The function shall be carried out once every oxygen sensor acquisition, i.e. every 10 ms.

T_SAMPLE = 10 ms

Initialisation:

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop (LV_ES).

LV_AFL[i] = 0
T_AFL_CYC[i] = 0
T_AFL_CYC_HLD[i] = 0
T_AFR_CYC[i] = 0
T_AFL_LOCK[i] = 2.55 (i.e. FFH)
CTR_AFL_CYC[i] = 0

Activation:

If (LV_ST_END = 1) **&**
(LV_ES = 0)

then ***Function activated***

endif.

Deactivation:

If (LV_ST_END = 0) **or**
(LV_ES = 1)

then ***Function deactivated***

endif.

Formula section:

If (LV_LAM_LSCL[i] = 1) **or**
(LV_LS_UP_READY[i] = 1)

then **If** (*AF mixture transition flag* = ***Set***)

then **If** (LV_AFL[i] = 1)

then T_AFL_CYC[i] = 0

else T_AFR_CYC[i] = 0

endif

endif

If (LV_LAM_LSCL[i] = 1)

then **If** (LV_AFL[i] = 1)

then *Increment T_AFL_CYC[i]*

else *Increment T_AFR_CYC[i]*

endif

else CTR_AFL_CYC[i] = 0

endif
Reset *AF mixture transition flag*

If (T_AFL_LOCK[i] ≥ IP_T_AFL_LOCK)

then If (LV_AFL[i] = 1)

then If (VLS_UP[i] > C_VLS_AFR_THD)

then LV_AFL[i] = 0
Increment CTR_AFL_CYC[i]
T_AFL_CYC_HLD[i] = T_AFL_CYC[i]
T_AFL_LOCK[i] = 0
Set *AF mixture transition flag*

endif

else If (VLS_UP[i] < C_VLS_AFL_THD)

then LV_AFL[i] = 1
T_AFL_LOCK[i] = 0
Set *AF mixture transition flag*

endif

endif

else *Increment T_AFL_LOCK[i]*

endif

endif.

Increment T_AFL_CYC[i] / Increment T_AFR_CYC[i]

The timers T_AFL_CYC[i] & T_AFR_CYC[i] shall be incremented according to their specified conditions until they reach a maximum value of FFFFH at which point they shall no longer be incremented.

Increment CTR_AFL_CYC[i]

The Air-Fuel cycle counter CTR_AFL_CYC[i] shall be incremented for every lean to rich AF mixture transition, to indicate that a complete AF cycle has passed. The counter shall be incremented in such manner until it reaches a maximum value of FFFFH at which point it shall no longer be incremented.

Note: The internal flag indicating a AF mixture transition shall be reset after the AF counter CTR_AFL_CYC[i] has been processed.

Increment T_AFL_LOCK[i]

The timer T_AFL_LOCK[i] shall be incremented according to the specified conditions until it reaches a maximum value of FFH at which point it shall no longer be incremented.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VLS_AFR_THD	1	0...3FFH	0...4.995117	4.8828E-3	V
Threshold for detection of rich mixture					
C_VLS_AFL_THD	1	0...3FFH	0...4.995117	4.8828E-3	V
Threshold for detection of lean mixture					
IP_T_AFL_LOCK	8*8	0...FFH	0...2.55	0.01	s
LDPM_N_32_2_EGCP	8	0...FFH	0...8160	32	rpm
LDPM_MAF_1_EGCP	8	0...FFFFH	0...1389.342	0.0212	mg/stk
LV_AFL[i] transition lock-out time after transition of AF mixture					

7.39 Upstream oxygen sensor signal voltage

7.39.1 Signal evaluation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_UP_DRV1_MMV[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.5...2.499924	7.6294E-5	V/10ms
Mean of buffer containing 1st derivative of VLS_UP[i] signal					
VLS_UP_DRV1_MMV_MIN[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.5...2.499924	7.6294E-5	V/10ms
Minimum value of mean of buffer containing 1st derivative of VLS_UP[i] signal in lean phase					
VLS_UP_DRV1_ABS_MAX[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V/10ms
Absolute maximum value within buffer containing 1st derivative of VLS_UP[i] signal					
VLS_UP_MMV_MIN[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V
Mean oxygen sensor lean voltage					
VLS_UP_MMV_MAX[NC_CBK_EX_NR]	V/O	0...FFFFH	0...4.999924	7.6294E-5	V
Mean oxygen sensor rich voltage					
VLS_UP_BOL[NC_CBK_EX_NR]	V/O	0...3FFH	0…4.995117	4.8828E-3	V
Variable switching threshold for calculation of VLS_UP_MMV_MIN[i]					
VLS_UP_TOL[NC_CBK_EX_NR]	V/O	0...3FFH	0…4.995117	4.8828E-3	V
Variable switching threshold for calculation of VLS_UP_MMV_MAX[i]					
VLS_UP_MMV_HYS[NC_CBK_EX_NR]	V	0...3333H	0…0.999985	7.6294E-5	V
Variable hysteresis used to determine switching thresholds					
LV_VLS_UP_MMV_LIM[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating VLS_UP_MMV_xxx[i] signals being limited when set					
LV_VLS_UP_MMV_STK_VLD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating VLS_UP_MMV_xxx[i] signals valid when set					

Input data:

LV_ST_END	LV_ES	LV_PUC	NC_CBK_EX_NR
CTR_AFL_CYC[NC_CBK_EX_NR]		LV_LAM_LSCL[NC_CBK_EX_NR]	
VLS_UP[NC_CBK_EX_NR]			

FUNCTION DESCRIPTION:

General information:

The function shall evaluate the oxygen sensor signal voltage and compute a number of resultant values that may be used by other functions. These values shall include:

1. Moving mean of sensor voltage first derivative, i.e. average signal gradient
2. Minimum value of moving mean of sensor voltage first derivative in lean phase
3. Absolute maximum of sensor voltage first derivative over defined number of previous samples, i.e. measure of sensor voltage stability
4. Moving mean value of lean sensor voltage
5. Moving mean value of rich sensor voltage.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank

Signal flow diagram:

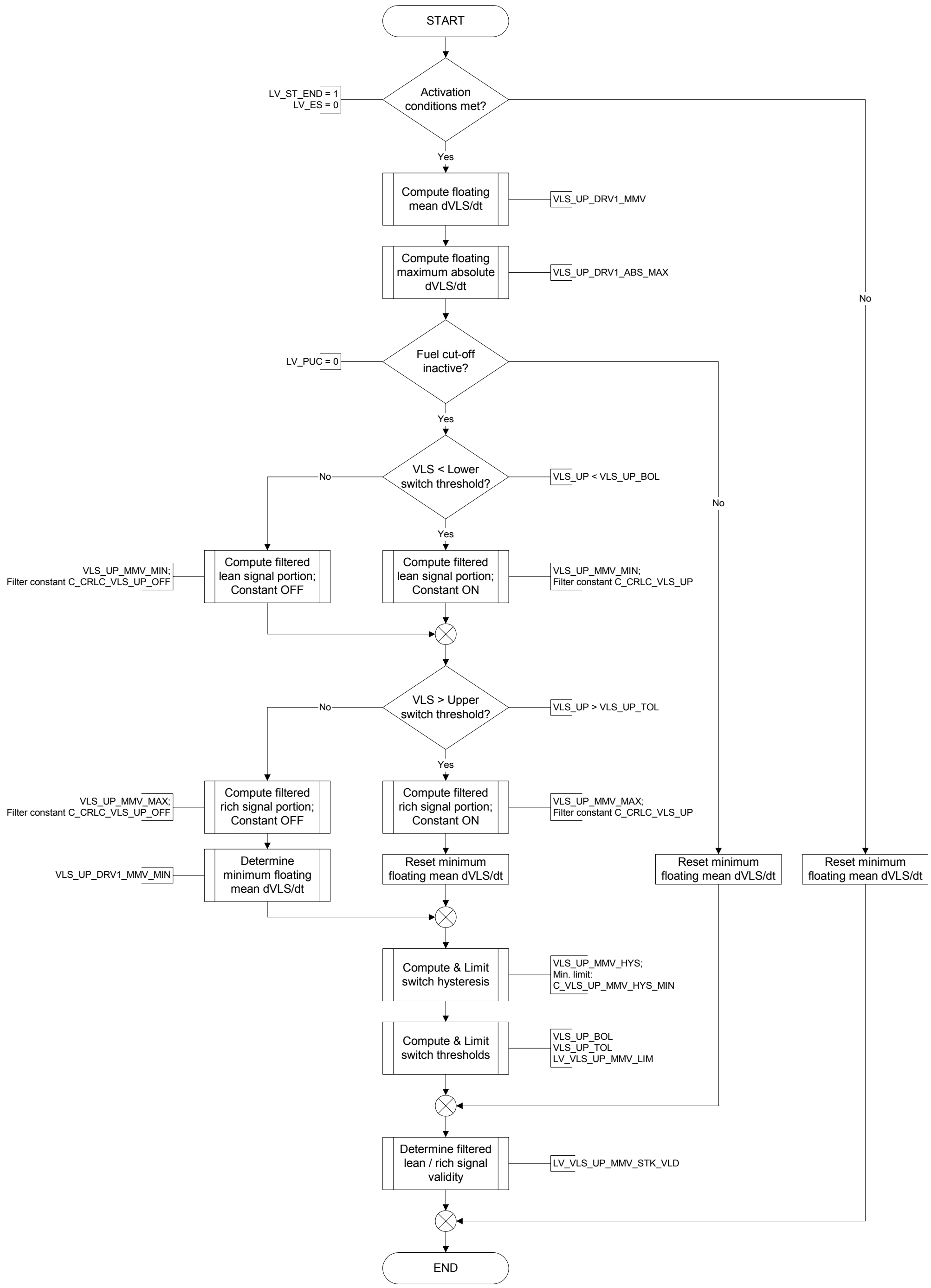


Figure 1 Function flow chart, Signal evaluation

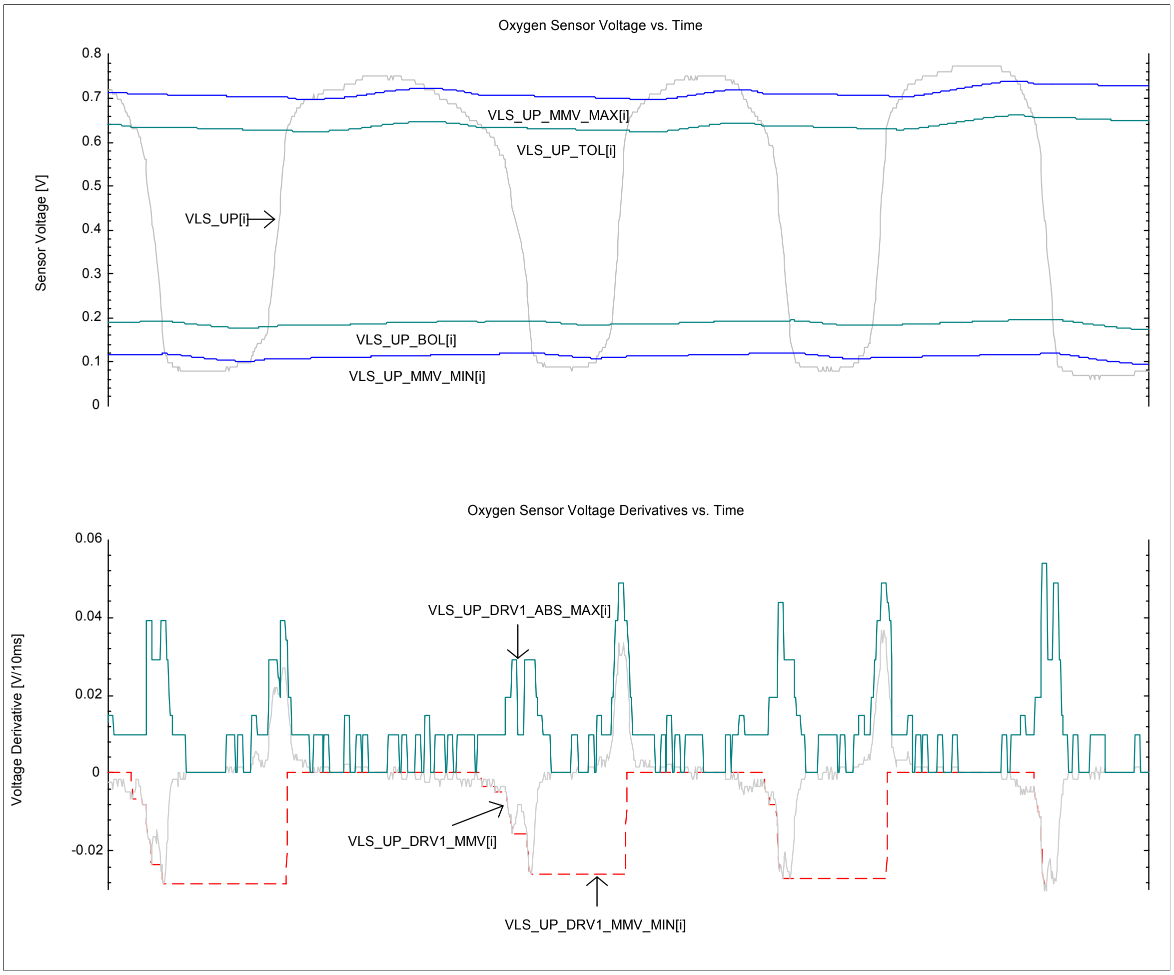


Figure 2 Physical relationship between values, Signal evaluation

Description:

The function shall evaluate the upstream oxygen sensor signal VLS_UP[i] and provide variables quantifying the signal characteristics: Floating mean gradient, Floating maximum signal deviation between samples, Filtered mean rich signal voltage & Filtered mean lean signal voltage. Furthermore, the validity of the latter two characteristics shall be provided. This shall be carried out as follows:

The function shall be activated should the engine state transition out of start (LV_ST_END = 1) and shall be deactivated should the engine stop state (LV_ES = 1) be entered.

Once the activation conditions have been met the function may be described in the following blocks:

1. Computation of moving mean of first derivative VLS_UP_DRV1_MMV[i]: The moving mean value of oxygen sensor signal voltage first derivative VLS_UP_DRV1_MMV[i] shall be computed by averaging the last differences between consecutive VLS_UP[i] samples, i.e. the most current difference ($VLS\_UP[i]_N$ - $VLS\_UP[i]_{N-1}$) and the previous 5 differences. This shall indicate a measure for the signal gradient.

2. Computation of absolute maximum of first derivative VLS_UP_DRV1_ABS_MAX[i]: The absolute maximum of the oxygen sensor voltage first derivative VLS_UP_DRV1_ABS_MAX[i] shall be computed by taking the maximum of the absolute values of the last differences between consecutive VLS_UP[i] samples, i.e. the most current difference ($VLS\_UP[i]_N$ - $VLS\_UP[i]_{N-1}$) and the previous 5 differences. This shall indicate a measure for the signal stability.

3. Computation of moving mean value of lean and rich sensor voltages, VLS_UP_MMV_xxx[i]: The computation of the moving mean value of lean and rich sensor voltages, associated switch thresholds and switch hysteresis shall only be carried out should the logical flag for overrun fuel cut-off be reset (LV_PUC = 0). The lean and rich mean signal voltages shall be implemented by use of first order low pass filtering and floating switching thresholds. For example:

Should the VLS_UP[i] signal exceeds the threshold VLS_UP_TOL[i], the rich mean signal voltage VLS_UP_MMV_MAX[i] shall be computed by filtering the VLS_UP[i] signal with filter constant C_CRLC_VLS_UP. This shall continue until threshold VLS_UP_TOL[i] is no longer exceeded, after which VLS_UP_MMV_MAX[i] shall be computed by filtering with filter constant C_CRLC_VLS_UP_OFF. This constant is much smaller than C_CRLC_VLS_UP and the effect is to cause the VLS_UP_MMV_MAX[i] to follow the rich peaks. The OFF constant shall permit the VLS_UP_MMV_MAX[i] signal to follow the VLS_UP[i] signal for sudden decreases in signal amplitude, i.e. maximum VLS_UP[i] lower than VLS_UP_TOL[i].

The same procedure shall be carried out for the computation of VLS_UP_MMV_MIN[i], i.e. where VLS_UP[i] falls below VLS_UP_BOL[i], VLS_UP_MMV_MIN[i] shall be computed with filter constant C_CRLC_VLS_UP and where VLS_UP[i] no longer falls below VLS_UP_BOL[i], the filter constant C_CRLC_VLS_UP_OFF shall be used.

4. Determination of the minimum value of the moving mean of first derivative VLS_UP_DRV1_MMV_MIN[i]: The same conditions as for the computation of VLS_UP_MMV_xxx[i] shall apply. Should VLS_UP[i] fall below or equal VLS_UP_TOL[i], $VLS\_UP\_DRV1\_MMV\_MIN[i]_N$ shall be equal to the lower of $VLS\_UP\_DRV1\_MMV\_MIN[i]_{N-1}$ & VLS_UP_DRV1_MMV[i]. When VLS_UP[i] exceeds VLS_UP_TOL[i], LV_PUC = 1 or the function is deactivated, VLS_UP_DRV1_MMV_MIN[i] shall be reset to 0.

5. Compute & limit switch hysteresis: The value of the hysteresis, VLS_UP_MMV_HYS[i], used to determine the switching thresholds VLS_UP_BOL[i] & VLS_UP_TOL[i] shall be made proportional using factor C_FAC_VLS_UP_MMV_HYS to the oxygen sensor signal amplitude, as determined by taking the absolute difference of the moving mean lean / rich

variables. The hysteresis shall also be limited so that it may not fall below the calibrateable threshold C_VLS_UP_MMV_HYS_MIN.

6. Compute & limit switch thresholds: The function shall limit the minimum amplitude of the difference between the moving mean lean / rich variables (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) to twice that of the switch hysteresis, VLS_UP_MMV_HYS[i].

Should the amplitude of the aforementioned difference fall below the said threshold, both the switch thresholds VLS_UP_BOL[i] & VLS_UP_TOL[i] shall be set to the average of the moving mean lean / rich variables, the moving mean lean / rich variables are recalculated based on the limited switching thresholds and hysteresis and the flag LV_VLS_UP_MMV_LIM[i] shall be set to indicate that switch threshold limiting is active.

Should no limiting be required, the switch thresholds VLS_UP_BOL[i] & VLS_UP_TOL[i] shall be calculated based on the switch hysteresis, moving mean lean variable and moving mean rich variable respectively. The flag LV_VLS_UP_MMV_LIM[i] shall be reset to indicate that no switch threshold limiting is being carried out.

7. Determine validity of moving mean lean / rich sensor voltages: The function shall determine whether the moving mean lean / rich signals are valid to permit certain use in other functions. The validity of the signals VLS_UP_MMV_MIN[i] & VLS_UP_MMV_MAX[i], as indicated by the logical variable LV_VLS_UP_MMV_STK_VLD[I], is determined to be given once the moving mean lean / rich signals have begun to stabilise after function activation, i.e. when VLS_UP_MMV_MIN[i] is less than VLS_UP_MMV_MAX[i] and once sufficient Air-Fuel cycles, as denoted by CTR_AFL_CYC[i], have passed since lambda controller activation (C_CTR_VLS_UP_AFL_CYC_MIN cycles) and where the lambda controller is still active (LV_LAM_LSCL[i] = 1).

The above blocks shall be cycled through once every function recurrence.

Application conditions:

Recurrence:

The function shall be carried out once every oxygen sensor acquisition, i.e. every 10 ms.

Initialisation:

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop (LV_ES).

VLS_UP_MMV_MIN[i] = C_VLS_UP_INI_MIN
VLS_UP_MMV_MAX[i] = C_VLS_UP_INI_MAX
VLS_UP_BOL[i] = C_VLS_UP_INI_MAX
VLS_UP_TOL[i] = C_VLS_UP_INI_MIN
VLS_UP_MMV_HYS[i] = C_VLS_UP_MMV_HYS_MIN
LV_VLS_UP_MMV_LIM[i] = 0
VLS_UP_DRV1_MMV[i] = 0
VLS_UP_DRV1_MMV_MIN[i] = 0
VLS_UP_DRV1_ABS_MAX[i] = 0
LV_VLS_UP_MMV_STK_VLD[i] = 0

Buffer contents from which VLS_UP_DRV1_MMV[i] & VLS_UP_DRV1_ABS_MAX[i] computed = 0.

Activation:

If (LV_ST_END = 1) **&**
(LV_ES = 0)

then ***Function activated***

endif.

Deactivation:

If (LV_ST_END = 0) **or**
(LV_ES = 1)

then ***Function deactivated***
VLS_UP_DRV1_MMV_MIN[i] = 0

endif.

Formula section:

The following function shall be carried out in the order listed and as indicated by the function flow chart:

Computation of moving mean of first derivative VLS_UP_DRV1_MMV[i]

$$VLS\_UP\_DRV1\_MMV[i] = \frac{\sum_{x=0}^{5}\left(VLS\_UP[i]_{N-x} - VLS\_UP[i]_{N-(x+1)}\right)}{6 * 10ms}$$

The above may be facilitated for example by use of an 6 value ring buffer. The most current difference, ($VLS\_UP[i]_N$ - $VLS\_UP[i]_{N-1}$), would overwrite the oldest difference in the buffer, ($VLS\_UP[i]_{N-5}$ - $VLS\_UP[i]_{N-6}$), and the average of the buffer contents recalculated.

Computation of absolute maximum of first derivative number of previous samples VLS_UP_DRV1_ABS_MAX[i]

$$VLS\_UP\_DRV1\_ABS\_MAX[i] = MAXIMUM\left|{}_{x=0}^{5}\left(VLS\_UP[i]_{N-x} - VLS\_UP[i]_{N-(x+1)}\right)\right|$$

The above may be facilitated for example by use of the same 6 value ring buffer, as proposed above for the computation of VLS_UP_DRV1_MMV[i]. The function would calculate the absolute values of the buffer contents and take the maximum of these values.

Computation of moving mean value of lean and rich sensor voltages, VLS_UP_MMV_xxx[i]

If (LV_PUC = 0)

then **If** (VLS_UP[i] < VLS_UP_BOL[i])

then *Compute moving mean lean signal, filter constant ON*

else *Compute moving mean lean signal, filter constant OFF*

endif

If (VLS_UP[i] > VLS_UP_TOL[i])

then *Compute moving mean rich signal, filter constant ON*
VLS_UP_DRV1_MMV_MIN[i] = 0

else *Compute moving mean rich signal, filter constant OFF*
Determine minimum of moving mean of first derivative

endif

Compute & limit switch hysteresis

Compute & limit switch thresholds

else VLS_UP_DRV1_MMV_MIN[i] = 0

endif

Determine validity of moving mean lean / rich sensor voltages.

Compute moving mean lean signal, filter constant ON

The moving mean of the lean signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_UP.

$VLS\_UP\_MMV\_MIN[i]_N = VLS\_UP\_MMV\_MIN[i]_{N-1} * (1 - C\_CRLC\_VLS\_UP) + VLS\_UP[i]_N * C\_CRLC\_VLS\_UP$

Compute moving mean lean signal, filter constant OFF

The moving mean of the lean signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_UP_OFF.

$VLS\_UP\_MMV\_MIN[i]_N = VLS\_UP\_MMV\_MIN[i]_{N-1} * (1 - C\_CRLC\_VLS\_UP\_OFF) + VLS\_UP[i]_N * C\_CRLC\_VLS\_UP\_OFF$

Compute moving mean rich signal, filter constant ON

The moving mean of the rich signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_UP.

$VLS\_UP\_MMV\_MAX[i]_N = VLS\_UP\_MMV\_MAX[i]_{N-1} * (1 - C\_CRLC\_VLS\_UP) + VLS\_UP[i]_N * C\_CRLC\_VLS\_UP$

Compute moving mean rich signal, filter constant OFF

The moving mean of the rich signal voltage shall be realised by use of a 1st order low pass digital filter with the filter constant C_CRLC_VLS_UP_OFF.

$VLS\_UP\_MMV\_MAX[i]_N = VLS\_UP\_MMV\_MAX[i]_{N-1} * (1 - C\_CRLC\_VLS\_UP\_OFF) + VLS\_UP[i]_N * C\_CRLC\_VLS\_UP\_OFF$

Determine minimum of moving mean of first derivative VLS_UP_DRV1_MMV_MIN[i]

The determination of the minimum of VLS_UP_DRV1_MMV[i] shall take place when the condition VLS_UP[i] ≤ VLS_UP_TOL[i] has been met. In this case:

VLS_UP_DRV1_MMV_MIN[i] = MIN(VLS_UP_DRV1_MMV_MIN[i], VLS_UP_DRV1_MMV[i])

Should the above condition no longer met, LV_PUC be active or the initial activation conditions not be met then VLS_UP_DRV1_MMV_MIN[i] shall be reset to 0.

Compute & limit switch hysteresis

VLS_UP_MMV_HYS[i] = **|** (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) **|** * C_FAC_VLS_UP_MMV_HYS

If (VLS_UP_MMV_HYS[i] < C_VLS_UP_MMV_HYS_MIN)

then VLS_UP_MMV_HYS[i] = C_VLS_UP_MMV_HYS_MIN

endif.

Compute & limit switch thresholds

If ((VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) ≤ 2 * VLS_UP_MMV_HYS[i])

then VLS_UP_BOL[i] = (VLS_UP_MMV_MAX[i] + VLS_UP_MMV_MIN[i]) / 2
VLS_UP_TOL[i] = (VLS_UP_MMV_MAX[i] + VLS_UP_MMV_MIN[i]) / 2

VLS_UP_MMV_MIN[i] = VLS_UP_BOL[i] - VLS_UP_MMV_HYS[i]
VLS_UP_MMV_MAX[i] = VLS_UP_TOL[i] + VLS_UP_MMV_HYS[i]

LV_VLS_UP_MMV_LIM[i] = 1

else VLS_UP_BOL[i] = VLS_UP_MMV_MIN[i] + VLS_UP_MMV_HYS[i]
VLS_UP_TOL[i] = VLS_UP_MMV_MAX[i] - VLS_UP_MMV_HYS[i]

LV_VLS_UP_MMV_LIM[i] = 0

endif.

Determine validity of moving mean lean / rich sensor voltages

If (VLS_UP_MMV_MIN[i] < VLS_UP_MMV_MAX[i]) **&**
(LV_LAM_LSCL[i] = 1) **&**
(CTR_AFL_CYC[i] ≥ C_CTR_VLS_UP_AFL_CYC_MIN)

then LV_VLS_UP_MMV_STK_VLD[i] = 1

else LV_VLS_UP_MMV_STK_VLD[i] = 0

endif.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_VLS_UP_MMV_HYS	1	0...FFH	0...0.999600	3.92E-3	-
Multiplicative factor governing amplitude of hysteresis					
C_VLS_UP_MMV_HYS_MIN	1	0...3333H	0…0.999985	7.6294E-5	V
Minimum hysteresis for the computation of the thresholds VLS_UP_TOL[i] and VLS_UP_BOL[i]					
C_VLS_UP_INI_MIN	1	0...3FFH	0…4.995117	4.8828E-3	V
Lower initialisation value for VLS_UP_TOL[i]. VLS_UP_MMV_MIN[i]					
C_VLS_UP_INI_MAX	1	0...3FFH	0…4.995117	4.8828E-3	V
Upper initialisation value for VLS_UP_BOL[i]. VLS_UP_MMV_MAX[i]					
C_CRLC_VLS_UP	1	0...FFFFH	0...0.999999	1.5259E-5	-
Filter constant for determination of VLS_UP_MMV_xxx[i] values when respective threshold passed					
C_CRLC_VLS_UP_OFF	1	0...FFFFH	0...0.999999	1.5259E-5	-
Filter constant for determination of VLS_UP_MMV_xxx[i] values when respective threshold not passed					
C_CTR_VLS_UP_AFL_CYC_MIN	1	0...FFFFH	0...65535	1	-
Minimum number of AF cycles after lambda controller activation prior to validating VLS_UP_MMV_xxx[i] signals					

7.40 Manager of the exhaust gas temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEG_MDL_CBK	O	0...3H	1...4	1	-
array index for the actual calculated cylinder bank for the component model					
TEG_MDL_INP	O	0...FH	1...16	1	-
array index for the input values for the component model					
CTR_TEG_MDL	O	0...FH	1...16	1	-
index for actual calculated component model in the exhaust gas temperature model					
CTR_TUBE_MDL	O	0...5H	1...6	1	-
index for actual calculated tube model					
CTR_CAT_MDL	O	0...3H	1...4	1	-
index for actual calculated catalyst model					
CTR_TUR_MDL	O	0...1H	1...2	1	-
index for actual calculated turbine model					
T_SAMPLE_TEG	O	0...64H	0...1	0.01	s
sample time for exhaust gas temperature model					
TEG[NC_NR_TEG_MDL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
dynamic exhaust gas temperatures					
TEG_STAT_MDL[NC_NR_TEG_MDL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
static exhaust gas model temperatures					
TEG_DYN_MDL[NC_NR_TEG_MDL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
dynamic exhaust gas model temperatures					
HUM_EX_GAS_MDL[NC_NR_TEG_MDL]	O/V	0...3FFH	0...1.023E+3	1	g/kg
modelled humidity of the exhaust gas at different possitions in the exhaust line					
LV_TEMP_DEW_MDL[NC_NR_TEG_MDL]	O/V	0...1H	0...1	1	-
flags indicating that the dew point is passed at the different model positions					

Input data:

LV_ES	TCO	TEG_MES[NC_NR_TEG_SENS]	LV_ERR_TEG_MES[NC_NR_TEG_SENS]
TEMP_TUBE[NC_NR_TUBE_MDL]	TEMP_CAT[NC_NR_CAT_MDL]	TEMP_TUR[NC_NR_TUR_MDL]	NC_NR_TEG_MDL
NC_NR_ENG_OUT_MDL	NC_NR_TUBE_MDL	NC_NR_CAT_MDL	NC_NR_TUR_MDL
NC_NR_TEG_SENS	NC_TEG_MDL_INP_CONF[NC_NR_TEG_MDL]	NC_TEG_MDL_CBK_CONF[NC_NR_TEG_MDL]	NC_TEG_MDL_TYP_CONF[NC_NR_TEG_MDL]
NC_TEG_MDL_INST_CONF[NC_NR_TEG_MDL]	NC_TEG_MDL_T_SAMPLE_CONF[NC_NR_TEG_MDL]	NC_TEG_MDL_SAMPLE_OFS_CONF[NC_NR_TEG_MDL]	NC_TEG_SENS_CONF[NC_NR_TEG_MDL]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TEG_MDL_SENS_ENA[NC_NR_TEG_SENS]	1	0...1H	0...1	1	-
switch to use the sensor information in the exhaust gas temperature model					

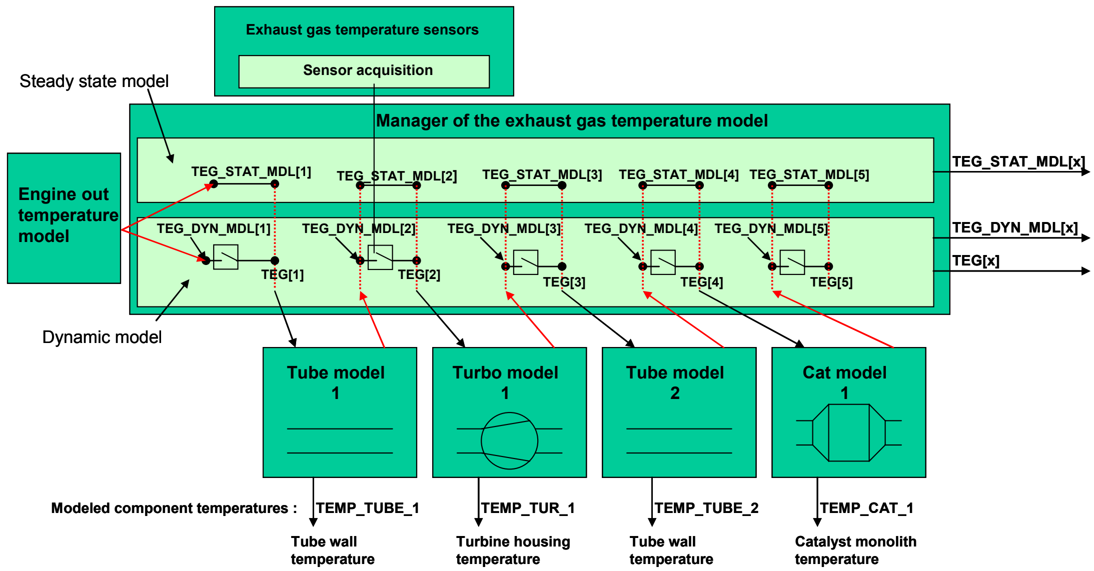
Export actions:

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)
writes back the calculated data back to the arrays of the manager (TEG_DYN_MDL, ...)

7.40.1 General information:

The manager of the exhaust gas temperature model is the module which coordinates the different component models. Dependent on the configuration the manager calls the different

component models (Engine out, Tube, Catalyst, Turbine model) in the configured order and update rate. The results of the different component models are written back by the action "ACTION_EXTD_Data2Manag" into the values:
TEG_DYN_MDL
TEG_STAT_MDL
HUM_EX_GAS_MDL
LV_DEW_TEMP_DEW_MDL
Example:



If a sensor value is available and no sensor error is detected the output TEG, which is input for the next model, is equal to the measured value (TEG_MES) if the corresponding LC_TEG_MDL_SENS_ENA[NC_NR_TEG_SENS] is set to one. On the other hand if an error is detected or no sensor is available or the LC_TEG_MDL_SENS_ENA is equal zero TEG[i] is equal to TEG_DYN_MDL[i].

All dynamic exhaust gas temperatures are initializied with the component temperature upstream its position. The component temperature itselfe is initializied in the component model by using the stored temperature at engine shut down weighted by a cooling down factor dependent on the engine off duration time T_ES. Due to the fact that the engine out model has no component temperature TCO is used as component temperature.

In the given example the exhaust gas temperatures initialized as following:
TEG[1] = TEG_DYN_MDL[1] = TCO
TEG[2] = TEG_DYN_MDL[2] = TEMP_TUBE[1]
TEG[3] = TEG_DYN_MDL[2] = TEMP_TUR[1]
TEG[4] = TEG_DYN_MDL[2] = TEMP_TUBE[2]
TEG[5] = TEG_DYN_MDL[1] = TEMP_CAT[1]
The steady state temperatures TEG_STAT_MDL, the humidities HUM_EX_GAS_MDL and the dewpoint flags of the model LV_TEMP_DEW_MDL are all initialized with zero.

Function Description

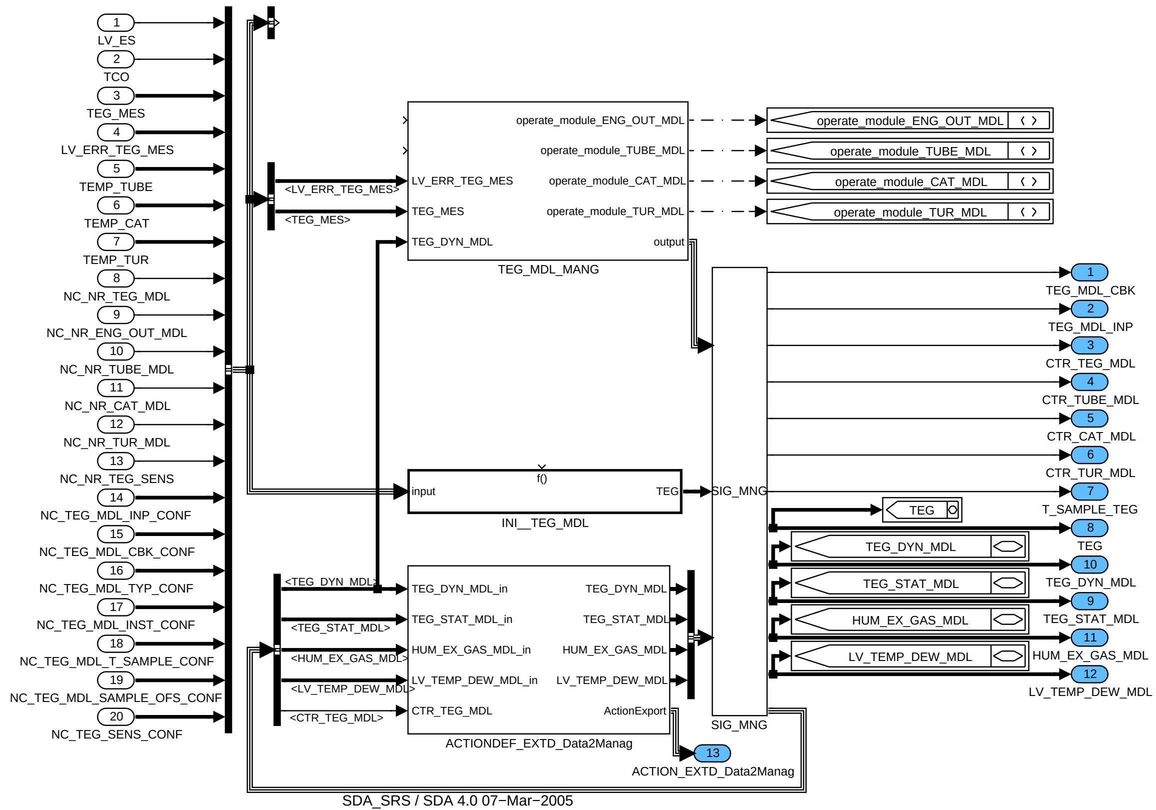


Figure 31 EXTD_MDLAD0

7.40.1.1 SUBFUNCTION: ACTIONDEF_EXTD_Data2Manag

Description for ACTION_EXTD_Data2Manag

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)

Please enter the deteailed Action Description here.

Parameter	Type	Hex. limits	Phys. limits	Resol.	Unit
Hum	IN	0...3FFH	0...1.023E+3	1	g/kg
-					
LvDew	IN	0...1H	0...1	1	-
-					
TegDyn	IN	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
-					
TegStat	IN	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
-					

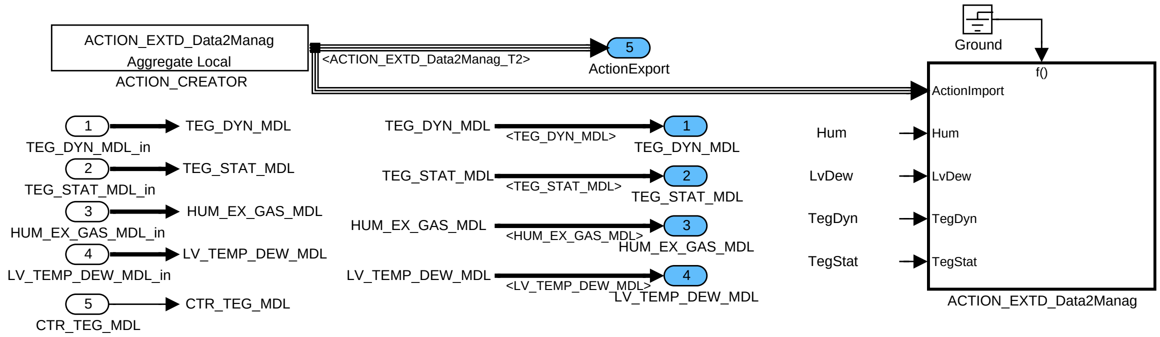


Figure 32 EXTD_MDLAD0/ ACTIONDEF_EXTD_Data2Manag

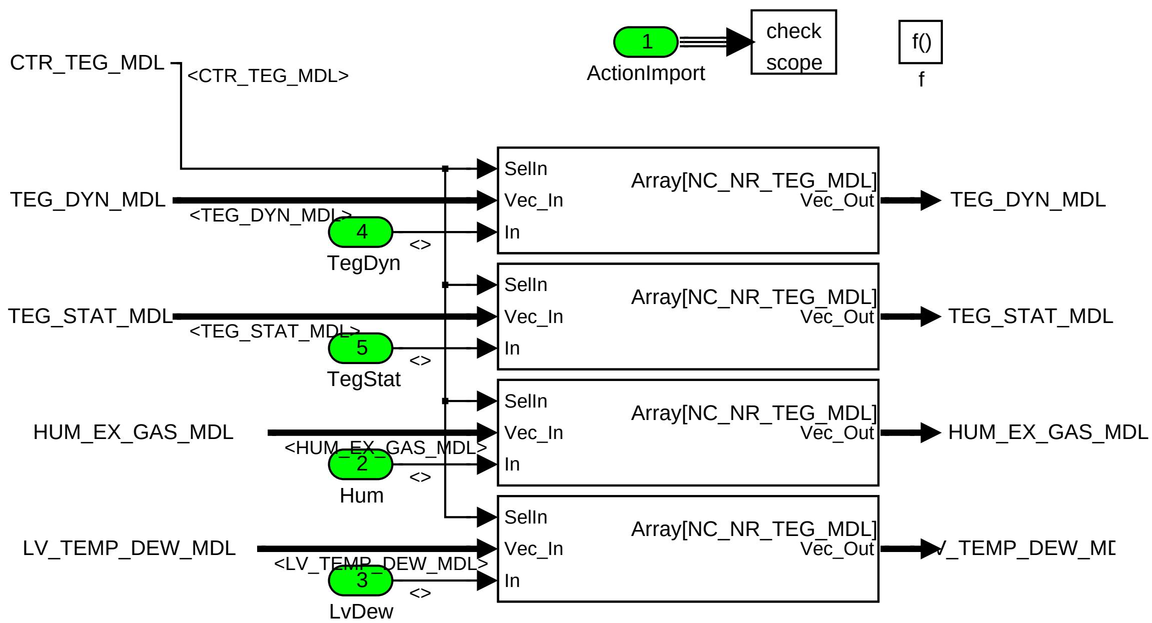


Figure 33 EXTD_MDLAD0/ ACTIONDEF_EXTD_Data2Manag/ ACTION_EXTD_Data2Manag

7.41 Dew point detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TEMP_DEW_LS_UP[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Flag indicating that dew point is passed at lambda sensor up catalyst					
LV_TEMP_DEW_LS_DOWN[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Flag indicating that dew point is passed at lambda sensor down catalyst					
CTR_COLD_ST_LS_UP	O/V/S	0...FFH	0...255	1	-
counter of engine start without leaving dew point phase at all upstream lambda sensor					
CTR_COLD_ST_LS_DOWN	O/V/S	0...FFH	0...255	1	-
counter of engine start without leaving dew point phase at all downstream lambda sensor					
MAF_KGH_INT_DEW	V	0...FFFFH	0...9.10208E+3	0.13888884	g
integral of MAF_KGH for the dew point detection					

Input data:

LV_TEMP_DEW_MDL[NC_NR_TEG_MDL]	TEMP_TUBE[NC_NR_TUBE_MDL]	LV_ES	LV_ST_END
STATE_SA	LV_PUC	MAF_FG_CYL	EFF_IGA_AV
TCO_ST	TAM_ST	NC_CBK_EX_NR	NC_NR_TEG_MDL
NC_NR_TUBE_MDL	TEMP_TUR[NC_NR_TUR_MDL]	NC_NR_TUR_MDL	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_SEL_TEMP_DEW_MDL_LS_DOWN[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector which value of LV_TEMP_DEW_MDL is used for the LV_TEMP_DEW_LS_DOWN calculation					
C_CTR_SEL_TEMP_DEW_MDL_LS_UP[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector which value of LV_TEMP_DEW_MDL is used for the LV_TEMP_DEW_LS_UP calculation					
C_CTR_SEL_TUBE_LS_DOWN[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector which value of TEMP_TUBE is used for the LV_TEMP_DEW_LS_DOWN calculation					
C_CTR_SEL_TUBE_LS_UP[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector which value of TEMP_TUBE is used for the LV_TEMP_DEW_LS_UP calculation					
C_CTR_SEL_TUR_LS_UP[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector which value of TEMP_TUR is used for the LV_TEMP_DEW_LS_UP calculation					
C_FAC_MAF_KGH_INT_DEW_SA	1	0...FFH	0...3.984375	0.015625	-
correction factor for MAF_KGH_INT_DEW calculation for active secundary air					
LC_LS_UP_DOWN_TUR	1	0...1H	0...1	1	-
switch to use turbine wall temperatures instead of exhaust manifold wall temperatures for the dew point for LS up catalyst					
ID_FAC_MAF_KGH_INT_THD_LS_DOWN	6	0...FFH	0...15.9375	0.0625	-
LDPM_CTR_COLD_ST_1_EXTD	6	0...FFH	0...255	1	-
factor to increase the MAF_KGH_INT threshold for the dew point detection of the downstream sensor					
ID_FAC_MAF_KGH_INT_THD_LS_UP	6	0...FFH	0...15.9375	0.0625	-
LDPM_CTR_COLD_ST_1_EXTD	6	0...FFH	0...255	1	-
factor to increase the MAF_KGH_INT threshold for the dew point detection of the upstream sensor					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TEMP_THD_DEW_LS_DOWN	12	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_TCO_ST_1_EXTD	12	0...FEH	-48...142.5	0.75	°C
wall temperature threshold for passed dew point of lambda sensor down catalyst					
IP_TEMP_THD_DEW_LS_UP	12	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_TCO_ST_1_EXTD	12	0...FEH	-48...142.5	0.75	°C
wall temperature threshold for passed dew point of lambda sensor up catalyst					
IP_FAC_MAF_KGH_INT_DEW	6x8	0...FFH	0...15.9375	0.0625	-
LDP_EFF_IGA_IP_FAC_MAF_INT_DEW	6	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_MAF_FG_CYL_IP_FAC_MAF_INT	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
weighting factor for MAF_KGH integration for dewpoint detection					
IP_MAF_KGH_INT_DEW_THD_LS_DOWN	12x8	0...FFFFH	0...9.10208E+3	0.13888884	g
LDPM_TCO_ST_1_EXTD	12	0...FEH	-48...142.5	0.75	°C
LDPM_TAM_ST_1_EXTD	8	0...FEH	-48...142.5	0.75	°C
MAF_KGH_INT_DEW treshold for dew point detection for the lambda sensor down catalyst					
IP_MAF_KGH_INT_DEW_THD_LS_UP	12x8	0...FFFFH	0...9.10208E+3	0.13888884	g
LDPM_TCO_ST_1_EXTD	12	0...FEH	-48...142.5	0.75	°C
LDPM_TAM_ST_1_EXTD	8	0...FEH	-48...142.5	0.75	°C
MAF_KGH_INT_DEW treshold for dew point detection for the lambda sensor up catalyst					

7.41.1 Dew point detection

This module is used to detect when the dew phase in the exhaust line is finished. This is necessary because the lambda sensors can only be heated with full power if there is no risk of thermal shock due to water drops. Therefore for each sensor (up- /down stream of the catalyst; bankselektive) a flag is set when the dew phase is over:

LV_TEMP_DEW_LS_UP[NC_CBK_EX_NR] for the upstream sensor

LV_TEMP_DEW_LS_DOWN[NC_CBK_EX_NR] for the downstream sensor

Three conditions are checked to set these bits:

1. The condition is that the physical model of the water condesation in the exhaust gas temperature model determines that there is no condensate in the exhaust line upstream the sensor. Therefore the exhaust gas temperature model delivers the flags LV_TEMP_DEW_MDL[NC_NR_TEG_MDL] for the different positions in the exhaust line.

2. Additionaly the model based wall temperatures are checked. The second condition is fulfilled if the coreponding temperatures (TEMP_TUBE; TEMP_TUR) are above the calibratable threshholds:
- IP_TEMP_THD_DEW_LS_UP for the upstream sensor
- IP_TEMP_THD_DEW_LS_DOWN for the downstream sensor

3. In the case that the exhaust gas temperature model is not jet well calibrated a third condition can be helpful to avoid sensor problems because it can be calibrated very fastly. Therefore a weithed integral of the mass air flow MAF_KGH_INT_DEW is calculated. The third condition is fulfilled if this integral is above the following threshholds:
- IP_MAF_KGH_INT_DEW_THD_LS_UP for the upstream sensor
- IP_MAF_KGH_INT_DEW_THD _LS_DOWN for the downstream sensor

To set the flags LV_TEMP_DEW_LS_xx either condition 1. and 2. are fullfilled or condition 3. is fullfilled.

The module consists out of three major function parts because of different update rates:

1. Calculation of the mass air flow integral: CLC__MAF_KGH_INT_DEW (update rate 1s)
2. Determination of the dew point (setting of the flags): CLC__LV_TEMP_DEW (update rate 100ms)
3. Incrementation of the cold start counter: CLC__COLD_ST_CTR (once at engine off)

Function Description

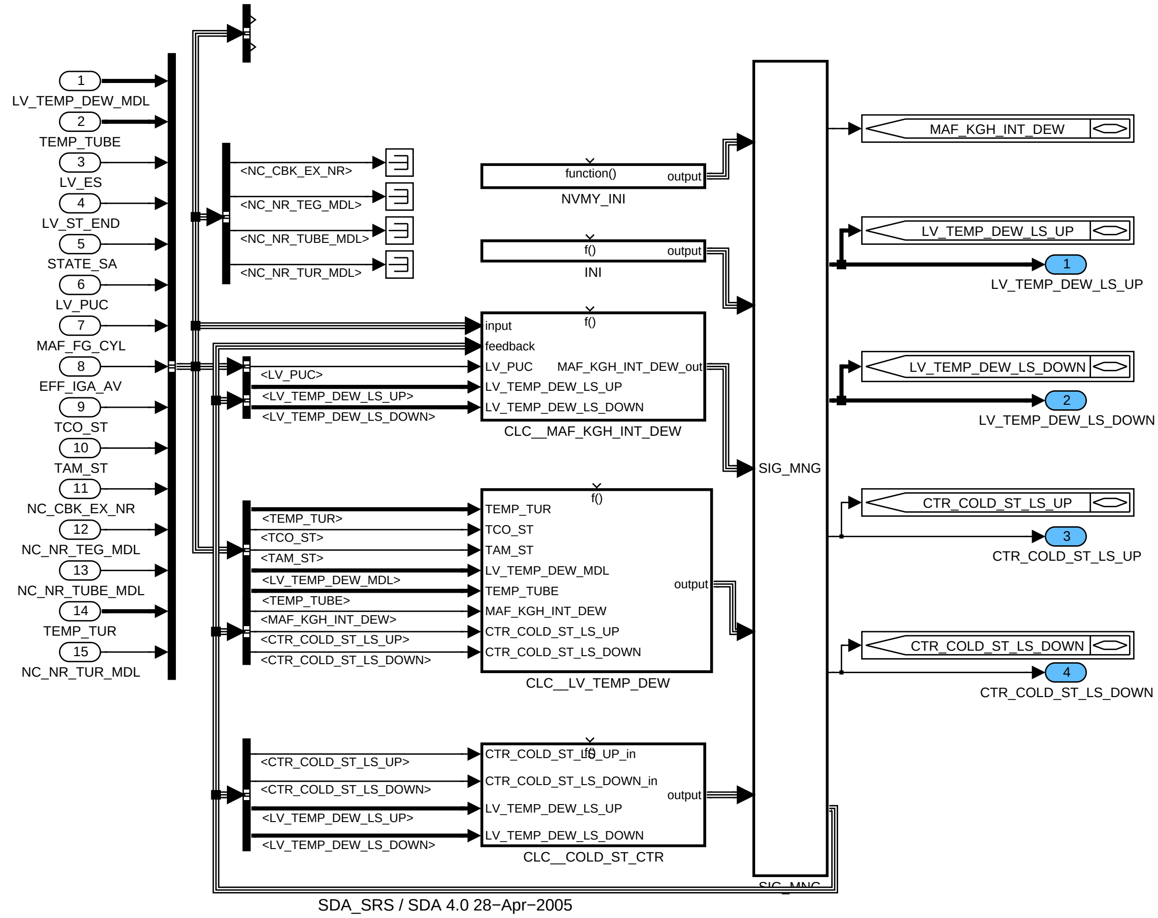


Figure 34
EXTD_REQGNdew0

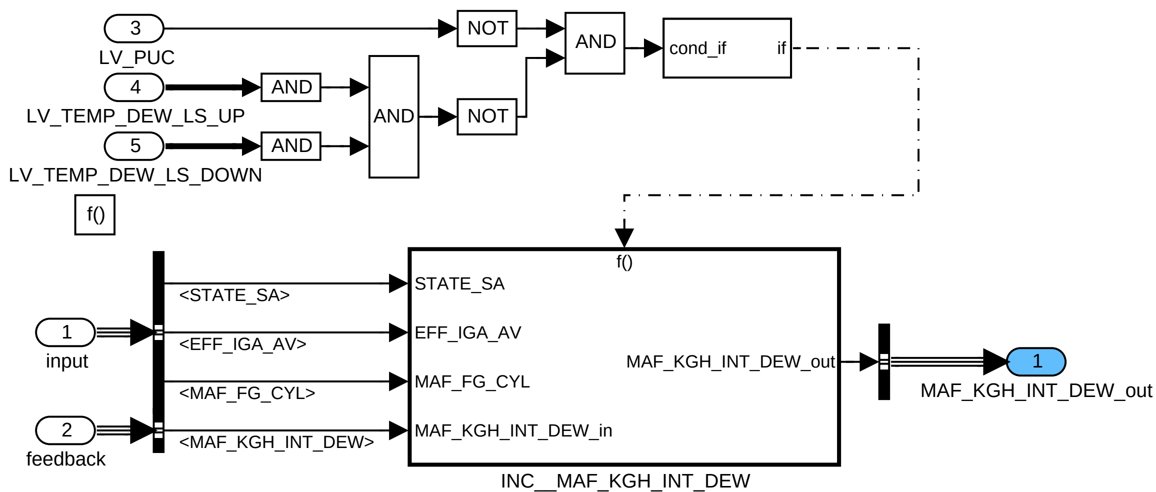


Figure 35 EXTD_REQGNdew0/
CLC__MAF_KGH_INT_DEW

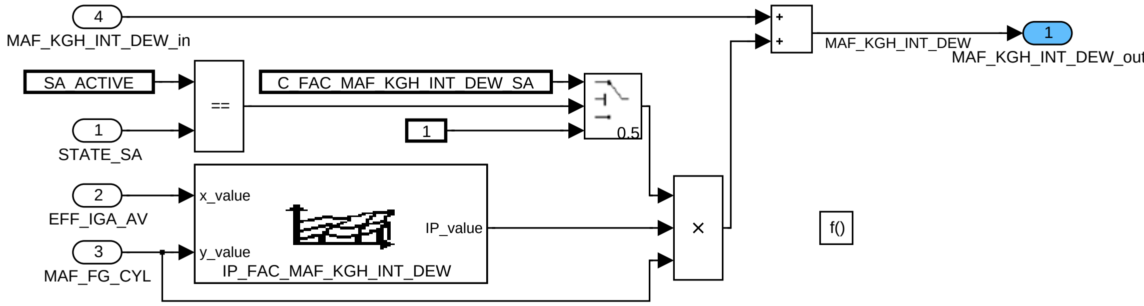


Figure 36 EXTD_REQGNdew0/ CLC__MAF_KGH_INT_DEW/
INC__MAF_KGH_INT_DEW

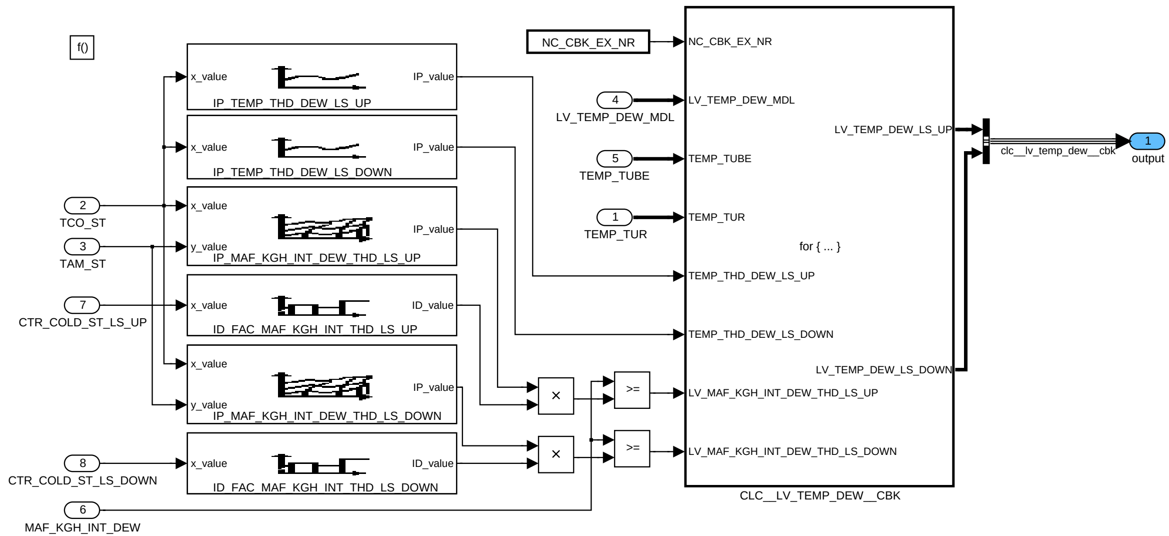
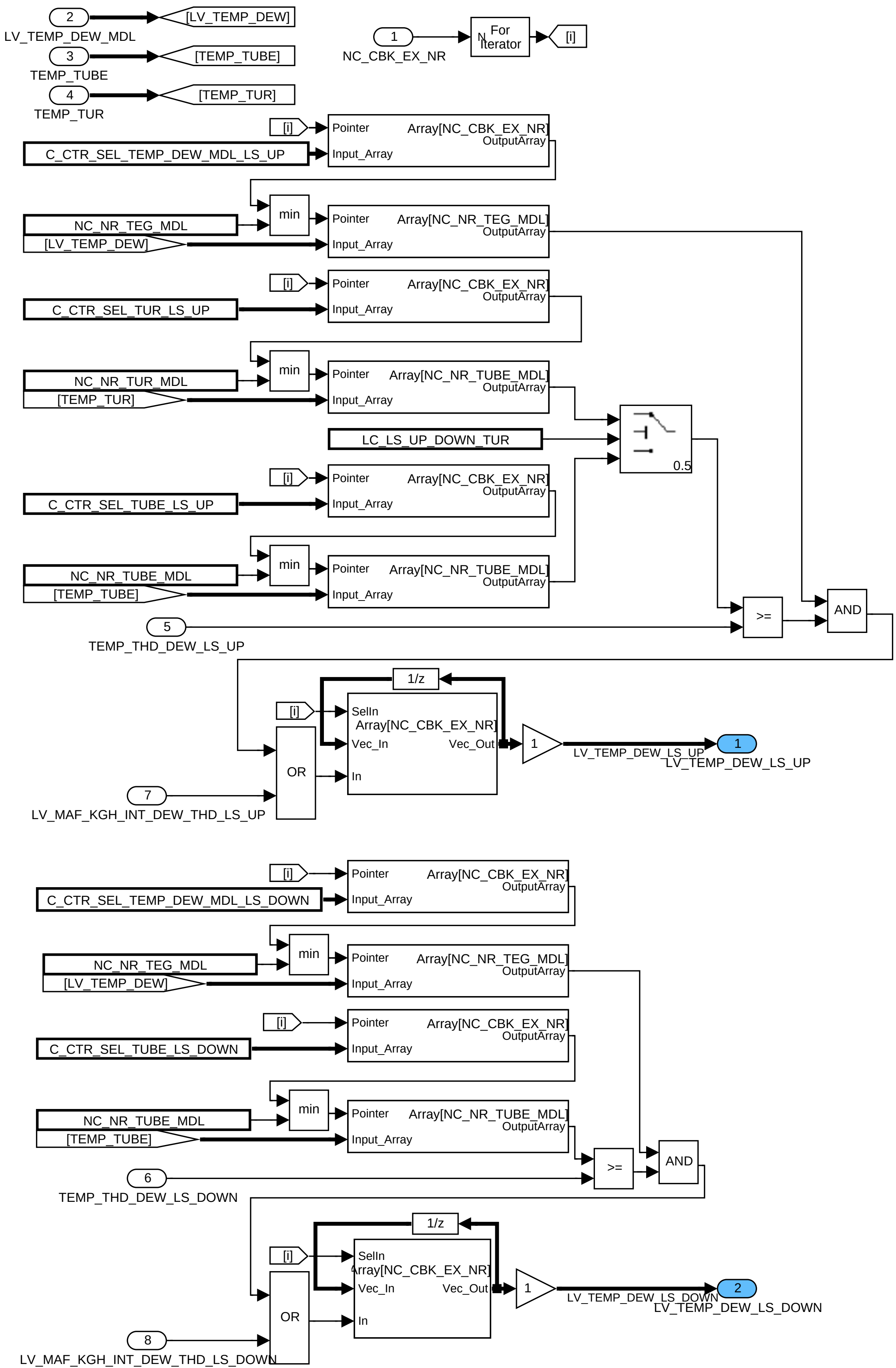


Figure 37 EXTD_REQGNdew0/ CLC__LV_TEMP_DEW

EXTD_REQGNDEW0/CLC__LV_TEMP_DEW/CLC__LV_TEMP_DEW__CBK



2
LV_TEMP_DEW_MDL
[LV_TEMP_DEW]
3
TEMP_TUBE
[TEMP_TUBE]
4
TEMP_TUR
[TEMP_TUR]
1
NC_CBK_EX_NR
N
For
Iterator
[i]
Pointer
Input_Array
Array[NC_CBK_EX_NR]
OutputArray
C_CTR_SEL_TEMP_DEW_MDL_LS_UP
min
NC_NR_TEG_MDL
Array[NC_NR_TEG_MDL]
C_CTR_SEL_TUR_LS_UP
NC_NR_TUR_MDL
Array[NC_NR_TUBE_MDL]
LC_LS_UP_DOWN_TUR
0.5
C_CTR_SEL_TUBE_LS_UP
NC_NR_TUBE_MDL
>=
AND
5
TEMP_THD_DEW_LS_UP
1/z
SelIn
Vec_In
Vec_Out
In
OR
7
LV_MAF_KGH_INT_DEW_THD_LS_UP
LV_TEMP_DEW_LS_UP
C_CTR_SEL_TEMP_DEW_MDL_LS_DOWN
C_CTR_SEL_TUBE_LS_DOWN
6
TEMP_THD_DEW_LS_DOWN
8
LV_MAF_KGH_INT_DEW_THD_LS_DOWN
LV_TEMP_DEW_LS_DOWN

Figure 38 EXTD_REQGNdew0/ CLC__LV_TEMP_DEW/ CLC__LV_TEMP_DEW__CBK

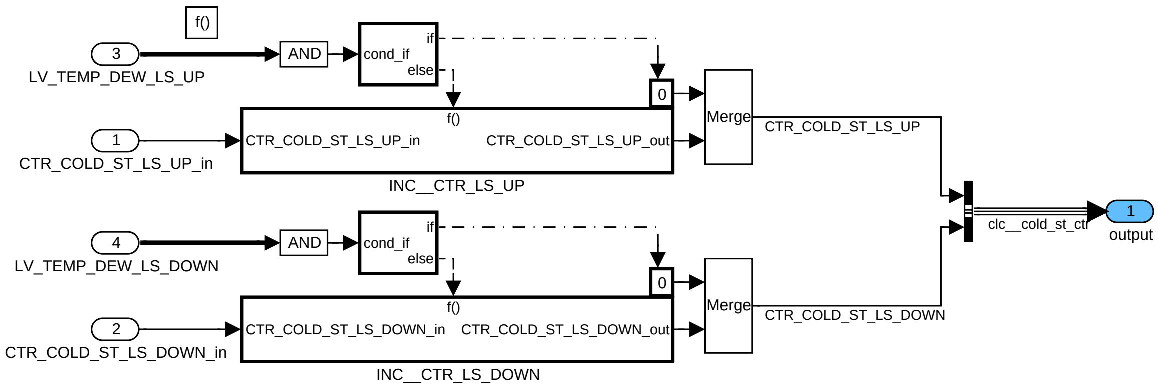


Figure 39 EXTD_REQGNdew0/ CLC__COLD_ST_CTR

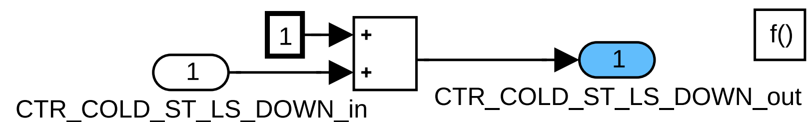


Figure 40 EXTD_REQGNdew0/ CLC__COLD_ST_CTR/ INC__CTR_LS_DOWN

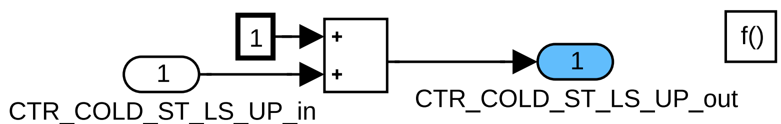


Figure 41 EXTD_REQGNdew0/ CLC__COLD_ST_CTR/ INC__CTR_LS_UP

7.42 Engine out exhaust gas temperature model

Input data:

LV_ES	TCO	TCO_ST	TIA_CYL
LV_PUC	T_PUC	TAM	N_TEG
MAF_TEG	IGA_AV_MV_CBK_TEG[NC_CBK_EX_NR]	LAMB_CBK_TEG_ENG[NC_CBK_EX_NR]	NR_CYL_SCC_CBK[NC_CBK_EX_NR]
SAF_KGH_CBK_TEG[NC_CBK_EX_NR]	MEF_KGH_CBK_TEG[NC_CBK_EX_NR]	MAF_KGH_CBK_TEG[NC_CBK_EX_NR]	TEG_ENG_OUT_ADD_CUS[NC_NR_ENG_OUT_MDL]
FAC_TEG_ENG_OUT_CUS[NC_NR_ENG_OUT_MDL]	IGA_REF_TEG_ADD_CUS[NC_NR_ENG_OUT_MDL]	IGA_BAS_TEG_ADD_CUS[NC_NR_ENG_OUT_MDL]	FAC_TEG_ADD_IGA_CUS[NC_NR_ENG_OUT_MDL]
TEG_MDL_CBK	NC_CYL_NR	NC_CBK_EX_NR	NC_NR_TEG_MDL
NC_NR_ENG_OUT_MDL			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_HUM_ENG_OUT	1	0...3FFH	0...1.023E+3	1	g/kg
humidity of the engine out exhaust gas					
IP_FAC_TEG_SCC_ADD	10	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_T_PUC_IP_TEG_SCC_ADD	10	0...FFFFH	0...655.35	0.01	s
-					
IP_FAC_TEG_TIA	8	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_TIA_CYL_IP_FAC_TEG_TIA	8	0...FEH	-48...142.5	0.75	°C
multiplicative correction of engine out exhaust gas temperature due to intake air temperature					
IP_HUM_EX_GAS_MAX	8	0...3FFH	0...1.023E+3	1	g/kg
LDP_TEG_IP_HUM_EX_GAS_MAX	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
maximal humidity of the exhaust gas tependent on the exhaust gas temperature					
IP_RATIO_TEG_SCC	5	0...FFH	0...0.99609375	0.00390625	-
LDPM_NR_CYL_SCC_CBK_1_EXTD	5	0...50H	0...8	0.1	-
ratio of exhaust gas comming from shut off cylinders					
IP_TEG_ADD_IGA	12	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
LDP_IGA_DIF_IP_TEG_ADD_IGA	12	0...FFH	-20.25...75.375	0.375	°CRK
additive correction of engine out exhaust gas temperaturedue to ignition					
IP_TEG_ADD_LAMB	8	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
LDPM_LAMB_ENG_1_EXTD	8	0...7FFFH	0...31.9990234	9.76563E-4	-
additive correction of engine out exhaust gas temperature due to lambda					
IP_FAC_TEG_ADD_IGA[NC_NR_ENG_OUT_MDL]	10x12	0...FFH	0...1.9921875	0.0078125	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
correction factor on additive correction of the engine out exhaust gas temperature due to ignition					
IP_FAC_TEG_ADD_LAMB[NC_NR_ENG_OUT_MDL]	10x12	0...FFH	0...1.9921875	0.0078125	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
correction factor on additive correction of the engine out exhaust gas temperature due to lambda					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TEG_TCO	8x8	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_TCO_IP_FAC_TEG_TCO	8	0...FEH	-48...142.5	0.75	°C
LDP_TCO_ST_IP_FAC_TEG_TCO	8	0...FEH	-48...142.5	0.75	°C
multiplicative warm up correction of engine out exhaust gas temperature due to coolant temperature					
IP_IGA_BAS_TEG[NC_NR_ENG_OUT_MDL]	10x12	0...FFH	-35.625...60	0.375	°CRK
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
basic ignition angle for the exhaust gas temperature model					
IP_IGA_REF_TEG[NC_NR_ENG_OUT_MDL]	10x12	0...FFH	-35.625...60	0.375	°CRK
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
reference ignition angle for the exhaust gas temperature model					
IP_LAMB_BAS_TEG	10x12	0...7FFFH	0...31.9990234	9.76563E-4	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
basic lambda for the different engine operating point for the exhaust gas temperature model calibration					
IP_TEG_ADD_SA	6x6	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
LDP_SAF_KGH_CBK_TEG_IP_TEG_SA	6	0...FFFFH	0...1.02398E+3	0.015625	kg/h
LDP_MAF_KGH_CBK_TEG_IP_TEG_SA	6	0...FFFFH	0...4.09594E+3	0.0625	kg/h
additive correction of engine out exhaust gas temperature due to secundary air					
IP_TEG_ENG_OUT_BAS[NC_NR_ENG_OUT_MDL]	10x12	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
basic engine out exhaust gas temperature for reference conditions					
IP_TEG_SCC[NC_NR_ENG_OUT_MDL]	6x8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDP_N_TEG_IP_TEG_SCC	6	0...1FE0H	0...8.16E+3	1	rpm
LDP_TCO_IP_TEG_SCC	8	0...FEH	-48...142.5	0.75	°C
engine out exhaust gas temperature in case of full cylinder shut off after very long PUC duration					

Import actions:

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)
writes back the calculated data back to the arrays of the manager (TEG_DYN_MDL, ...)

7.42.1 General information:

The "engine out exhaust gas temperature model" is part of the complete exhaust gas temperature model. This part contains the calculation of the effective engine out exhaust gas temperature dependent on the engine operating parameters and the engine out exhaust gas humidity.

This module handles the following dependencies of the engine out exhaust gas temperature:
- Engine operating point (Engine speed N_TEG and load MAF_TEG)
- Ignition angle dependency
- Lambda dependency
- Correction for secondary air
- Engine out temperature during PUC operation
- Warm up correction versus coolant temperature TCO
- Correction versus intake air temperature TIA_CYL

For all other dependencies (VVT, EGR or MPL_XX) it is possible to add project specific corrections:
- TEG_ENG_OUT_ADD_CUS: additive correction on the engine out temperature
- FAC_TEG_ENG_OUT_CUS: multiplicative correction on the engine out temperature
- IGA_REF_TEG_ADD_CUS: additive correction on the reference ignition angle for the ignition angle dependency of the engine out temperature
- IGA_BAS_TEG_ADD_CUS: additive correction on the basic ignition angle for the ignition angle dependency of the engine out temperature
- FAC_TEG_ADD_IGA_CUS: additive correction of the map IP_FAC_TEG_ADD_IGA

Function Description

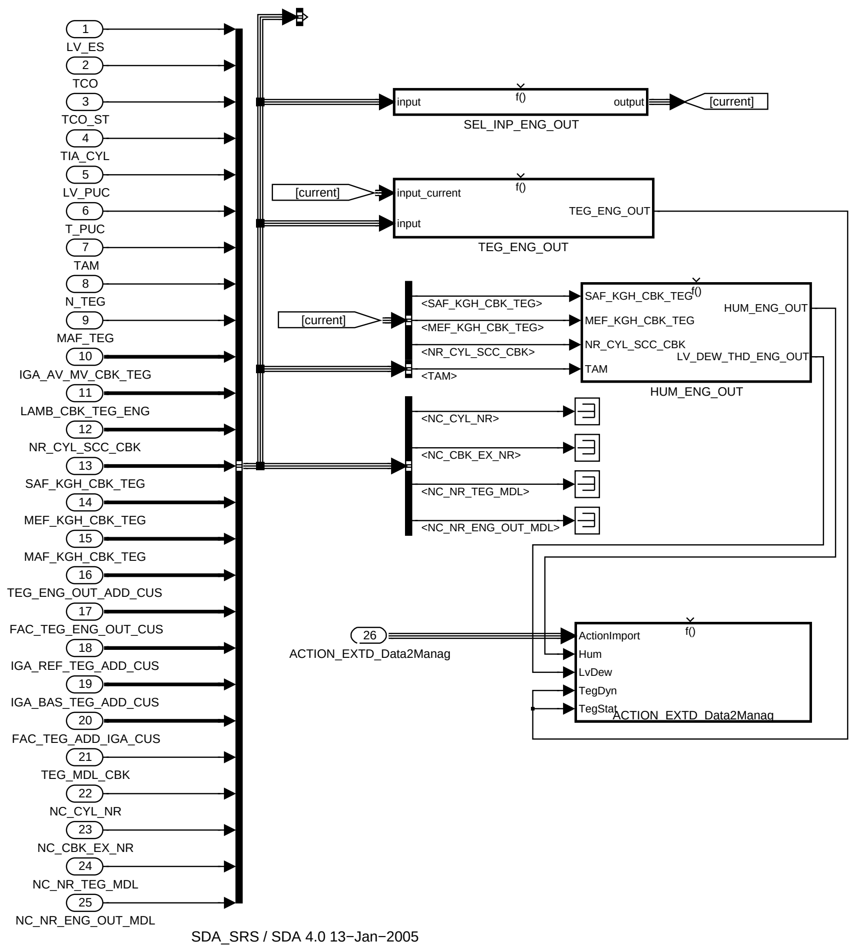


Figure 42 EXTD_MDLADengou0

7.42.1.1 HUM_ENG_OUT

In this function part the engine out exhaust gas humidity is determined. Therefore it is important to know that at lambda = 1 the engine produces a nearly constant amount of water (per kg exhaust gas) due to the combustion. This value can be calibrated in the calibration constant C_HUM_ENG_OUT. In the case of cylinder shut off no combustion takes place and so this amount of water has to be reduced dependent on the number of shut off cylinder on this cylinder bank. The so calculated exhaust gas humidity has to be increased by the intake air humidity. Thereby it is supposed that the relative humidity of the intake air is 100% because this is the worst case for the dew point calculation. So the intake air humidity is determined the map IP_HUM_EX_GAS_MAX with the ambient temperature TAM as input. Therefore TAM has to be converted to TEG_AM to have the same resolution as the exhaust gas temperature.

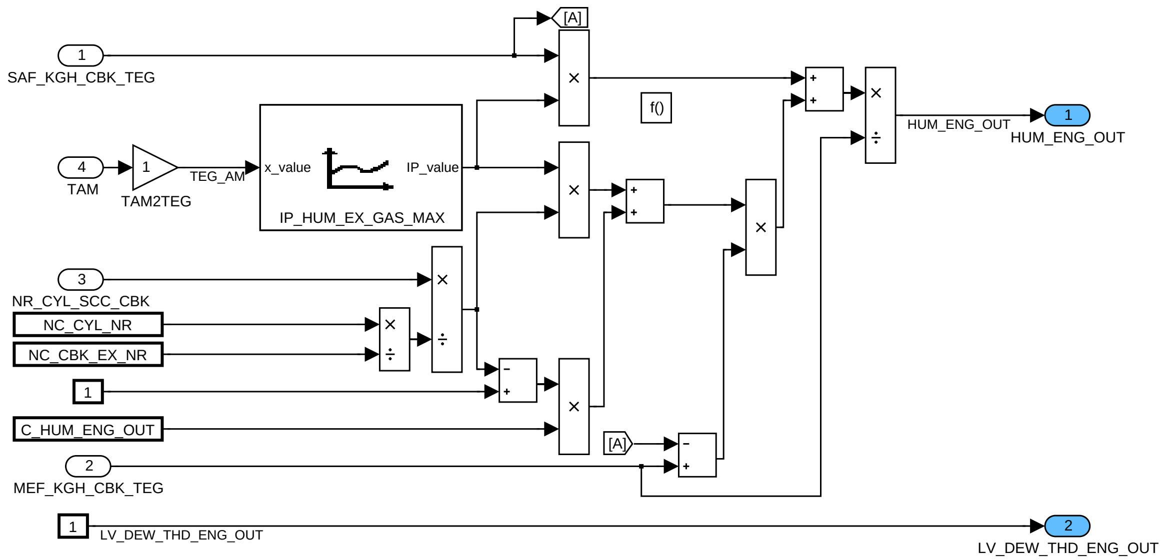


Figure 43 EXTD_MDLADengou0/ HUM_ENG_OUT

7.42.1.2 TEG_ENG_OUT

The calculation of the engine out exhaust gas temperature is split up in two parts. One part for the basic map and all additive correction CLC__TEG_ENG_OUT_BAS and one for all multiplicative corrections and the PUC calculations COR__TEG_ENG_OUT.

<u>CLC__TEG_ENG_OUT_BAS</u>

In this function part the basic effective engine out exhaust gas temperature is calculated. This basic engine out temperature is stored in the map IP_TEG_ENG_OUT_BAS. All corrections on this map will be zero if secondary air is inactive and the other parameters are equal to there calibrated basic values:

IGA_AV_MV_CBK_i = IP_IGA_BAS_TEG + project specific correction

LAMB_CBK_TEG_ENG_i = IP_LAMB_BAS_TEG

The ignition angle dependency is comparable with the ignition angle dependency of the torque model.

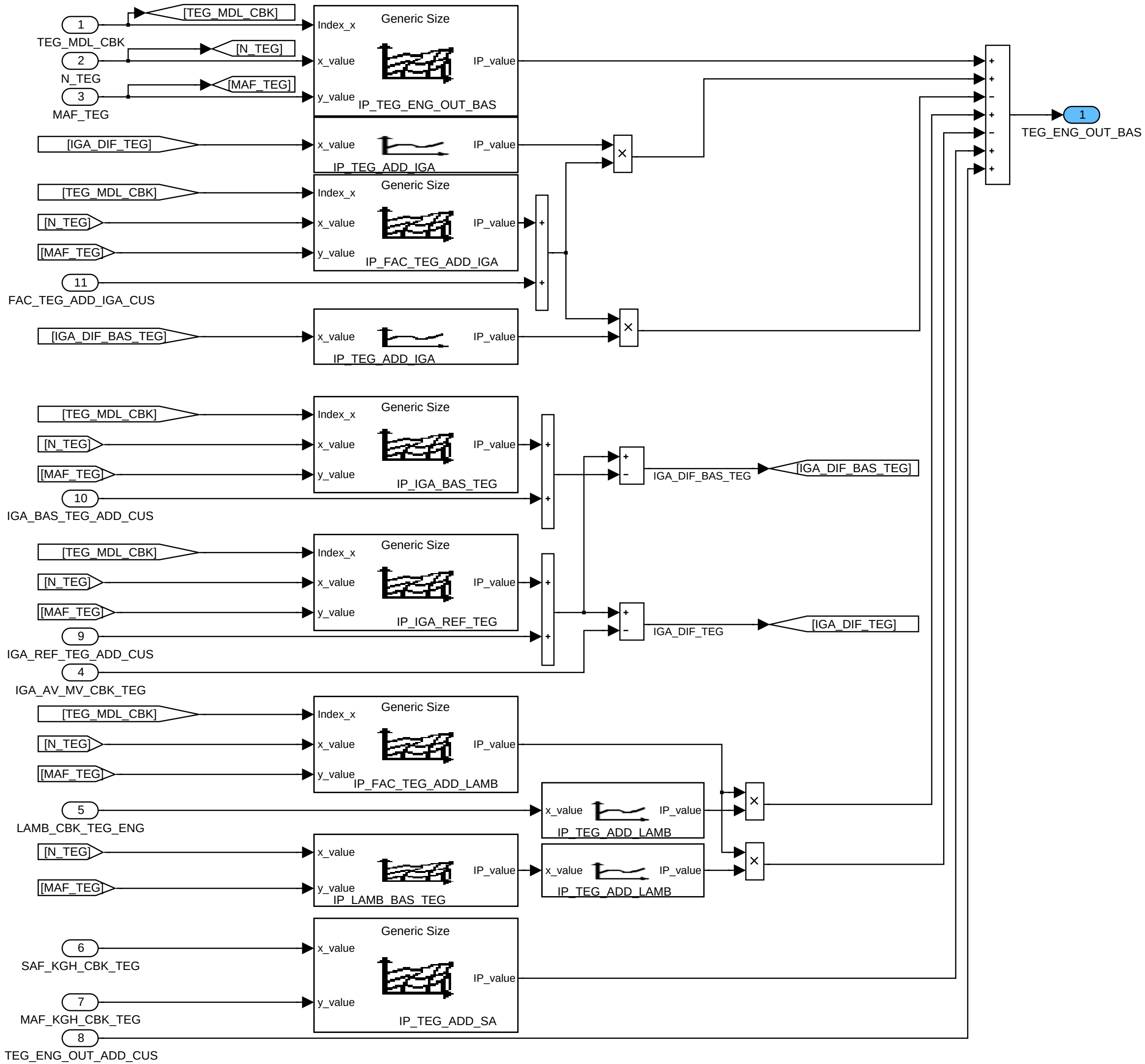


Figure 44 EXTD_MDLADengou0/ TEG_ENG_OUT/ CLC__TEG_ENG_OUT_BAS

COR__TEG_ENG_OUT

This part includes the multiplicative corrections on the previously calculated basic engine out exhaust gas temperature and the calculation of the engine out temperature at cylinder shut off TEG_SCC. It is important to know that all multiplicative corrections are calculated with the absolute value of the engine out exhaust gas temperature.

For the warm up phase it is possible to influence the engine out exhaust gas temperature dependent on TCO and TCO_ST with the map IP_FAC_TEG_TCO.

In the same way the exhaust gas temperature can be weighted dependent on TIA_CYL with the map IP_TEG_TIA.

The map IP_RATIO_TEG_SCC is used to switch over to PUC operation and to have a mixture temperature in case of single cylinder shut off. TEG_SCC is calculated in an own sub-function.

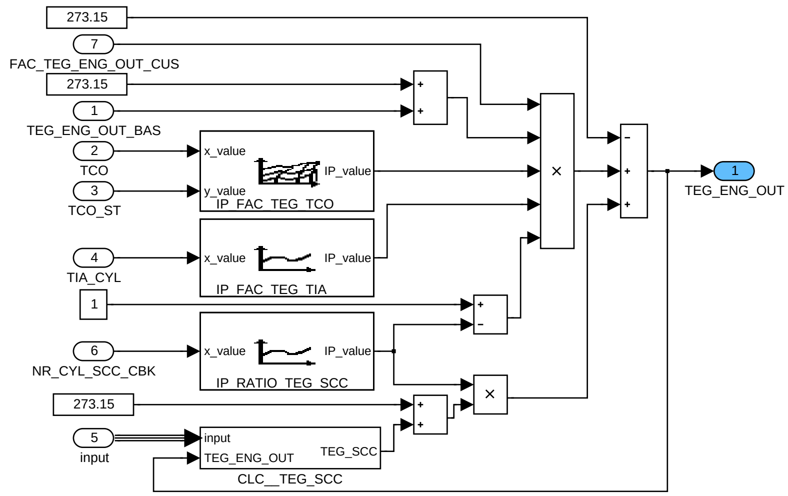


Figure 45 EXTD_MDLADengou0/ TEG_ENG_OUT/ COR__TEG_ENG_OUT

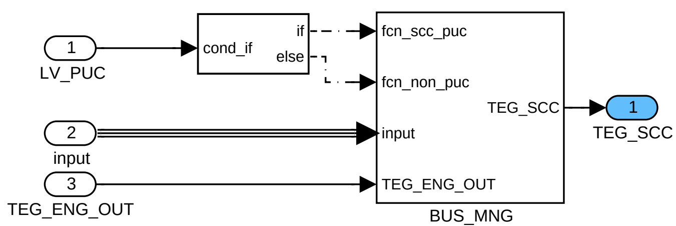
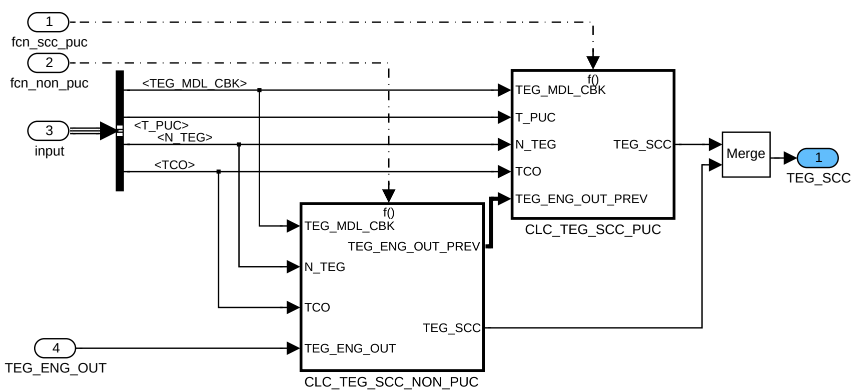
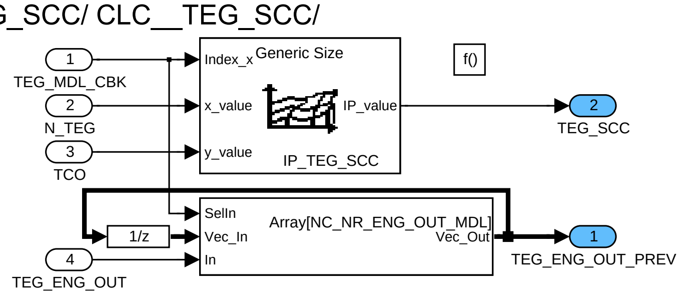


Figure 46 EXTD_MDLADengou0/ TEG_ENG_OUT/ COR__TEG_ENG_OUT/ CLC__TEG_SCC/



CLC__TEG_SCC

Figure 47 EXTD_MDLADengou0/ TEG_ENG_OUT/ COR__TEG_ENG_OUT/ CLC__TEG_SCC/ CLC__TEG_SCC/



BUS_MNG

Figure 48 EXTD_MDLADengou0/ TEG_ENG_OUT/ COR__TEG_ENG_OUT/ CLC__TEG_SCC/ CLC__TEG_SCC/ BUS_MNG/

CLC_TEG_SCC_NON_PUC

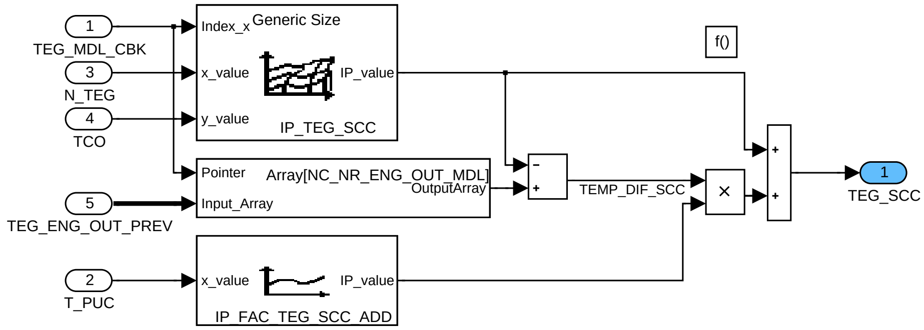


Figure 49 EXTD_MDLADengou0/ TEG_ENG_OUT/ COR__TEG_ENG_OUT/ CLC__TEG_SCC/ CLC__TEG_SCC/ BUS_MNG/ CLC_TEG_SCC_PUC

7.43 Tube exhaust gas temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_TUBE[NC_NR_TUBE_MDL]	O/V/S	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
tube temperatures					
EGY_DEW_INT_TUBE[NC_NR_TUBE_MDL]	V/S	0...FFFFH	0...327.675	0.005	kJ
dew energy integral of the tubes					
POW_HEAT_LOSS_TUBE[NC_NR_TUBE_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat losses of the differnt tubes					
POW_HEAT_ADD_TUBE[NC_NR_TUBE_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat flow from the exhaust gas to the differnt tubes					
EGY_HEAT_ADD_TUBE[NC_NR_TUBE_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
sum of all heat energy added to the differnt tubes					
TEG_DYN_DOWN_TUBE_FIL[NC_NR_TUBE_MDL]	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
fitlered exhaust gas temperature down tube					
EGY_DEW_TUBE[NC_NR_TUBE_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
dew energy added to the different tubes					
TEG_DYN_DOWN_TUBE_TMP[NC_NR_TUBE_MDL]	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
temporary exhaust gas temperature down tube					
MASS_EX_GAS_TUBE	-	0...FFFFFFFFH	0...2.98262E+3	6.94444E-7	kg
interim value definition for the mass of the exhaust gas flowing through the tube within sample time					
RATIO_COOL_DOWN_TUBE	-	0...FFH	0...0.99609375	0.00390625	-
engine cooling down progress dependent on coolant temperature					
TEMP_DIF_TUBE_1	-	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
interim value for temperature difference 1					
TEMP_DIF_TUBE_2	-	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
interim value for temperature difference 2					
TEMP_DIF_TUBE_3	-	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
interim value for temperature difference 3					
TEMP_DIF_TUBE_4	-	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
interim value for temperature difference 4					
TEMP_TUBE_CLC[NC_NR_TUBE_MDL]	-	0...FFFFFFFFH	-273.15 ... 1.77485E+3	4.76837E-7	°C
internal calculated tube temperature (high resolution)					

Input data:

LV_ES	TAM	VS	T_ES
LV_T_ES_NOT_PLAUS	TCO	TCO_STOP	N_TEG
MEF_CBK_TEG[NC_CBK_EX_NR]	MEF_KGH_CBK_TEG[NC_CBK_EX_NR]	T_SAMPLE_TEG	CTR_TUBE_MDL
TEG_MDL_INP	TEG_MDL_CBK	TEG[NC_NR_TEG_MDL]	TEG_STAT_MDL[NC_NR_TEG_MDL]
HUM_EX_GAS_MDL[NC_NR_TEG_MDL]	NC_CBK_EX_NR	NC_NR_TEG_MDL	NC_NR_TUBE_MDL

IP_HUM_EX_GAS_MAX	LV_PUC		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CAPA_HEAT_TEG	1	0...FFH	0...2.55	0.01	kJ/(kg*K)
heat capacity of the exhaust gas					
C_EGY_DEW	1	0...FFH	0...5.1	0.02	kJ/g
dew energy per g water					
C_FAC_POW_HEAT_LOSS_TUBE[NC_NR_TUBE_MDL]	1	0...FFFFH	0...29.8020412	4.5475E-4	nW/K^4
factor for the calculation of heat losses of the tube due to radiation					
LC_NO_TEMP_TUBE_CAL_ACT[NC_NR_TUBE_MDL]	1	0...1H	0...1	1	-
switch to allow initialisation and different correlation constants if no TEMP_TUBE calibration is done					
LC_TEG_DYN_DOWN_TUBE_FIL_SEL[NC_NR_TUBE_MDL]	1	0...1H	0...1	1	-
switch to select the filtered value as output					
IP_CAPA_HEAT_TUBE[NC_NR_TUBE_MDL]	8	0...FFFFH	0...65.535	0.001	kJ/K
LDPM_TEMP_1_EXTD	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
heat capacity of the differnt tubes					
IP_CRLC_TEG_DYN_DOWN_TUBE[NC_NR_TUBE_MDL]	10	0...FFFFH	0...0.99998474	1.52588E-5	-
LDPM_MEF_KGH_CBK_TEG_1_EXTD	10	0...FFFFH	0...4.09594E+3	0.0625	kg/h
filtering constant for the the exhaust gas temperature down tube					
IP_CRLC_TEG_DYN_DOWN_TUBE_DEC[NC_NR_TUBE_MDL]	10	0...FFFFH	0...0.99998474	1.52588E-5	-
LDPM_MEF_KGH_CBK_TEG_1_EXTD	10	0...FFFFH	0...4.09594E+3	0.0625	kg/h
filtering constant for the the exhaust gas temperature down tube for decreasing temperatures					
IP_CRLC_TEG_DYN_DOWN_TUBE_PUC[NC_NR_TUBE_MDL]	4	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_MEF_KGH_CBK_TEG_1_EXTD	4	0...FFFFH	0...4.09594E+3	0.0625	kg/h
Filtering constant for the exhaust gas temperature down tube during PUC					
IP_FAC_INI_TUBE	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_T_ES_1_EXTD	8	0...FFFFH	0...6.5535E+4	1	min
factor for tube cooling down dependent on engine off duration time					
IP_FAC_INI_TUBE_STND	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_RATIO_COOL_DOWN_1_EXTD	8	0...FFH	0...0.99609375	0.00390625	-
factor for tube cooling down in case of non plausible engine off duration time					
IP_FAC_POW_HEAT_LOSS_TUBE[NC_NR_TUBE_MDL]	8	0...FFFFH	0...1.02398437	1.5625E-5	kW/K
LDPM_VS_1_EXTD	8	0...FFH	0...255	1	km/h
calculation factor for the calculation of the heat flow to ambient of the differnt tubes					
IP_FAC_VS_TEG[NC_NR_TUBE_MDL]	8	0...FFH	0...3.984375	0.015625	-
LDPM_VS_1_EXTD	8	0...FFH	0...255	1	km/h
Temperature influence by the vehicle speed					
IP_FAC_POW_HEAT_ADD_TUBE[NC_NR_TUBE_MDL]	10x12	0...FFFFH	0...0.99998474	1.52588E-5	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MEF_CBK_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
calculation factor for the calculation of the heat flow from the exhaust gas to the tube					

Import actions:

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)
Acording Action Definition not found

7.43.1 General information:

The "tube exhaust gas temperature model" is part of the complete exhaust gas temperature model. This part models the behaviour of a simple exhaust tube like the exhaust manifold or other tubes where the exhaust gas is just cooled down. Therefore the outlet temperature TEG_DYN_DOWN_TUBE and the tube wall temperature TEMP_TUBE is calculated dependent on the ambient temperature TAM , on the inlet exhaust gas temperature TEG_DYN_UP_TUBE, on the mass exhaust gas flow MEF_KGH_CBK_TEG and on other input values. TEG_DYN_UP_TUBE and TEG_DYN_DOWN_TUBE are no visible values because they are just interim values for the calculation. The manager of the exhaust gas temperature model stores the result of the calculation in the array TEG with the correct index.

Initialisation of SRAM values at checksum error:

TEMP_TUBE[NC_NR_TUBE_MDL] = 0°C (1112H)

TEG_DYN_DOWN_TUBE_FIL[NC_NR_TUBE_MDL] = 0°C (1112H)

EGY_DEW_INT_TUBE[NC_NR_TUBE_MDL] = 0kJ (0H)

TEG_DYN_DOWN_TUBE_TMP[NC_NR_TUBE_MDL] = 0°C (1112H)

Application Condition

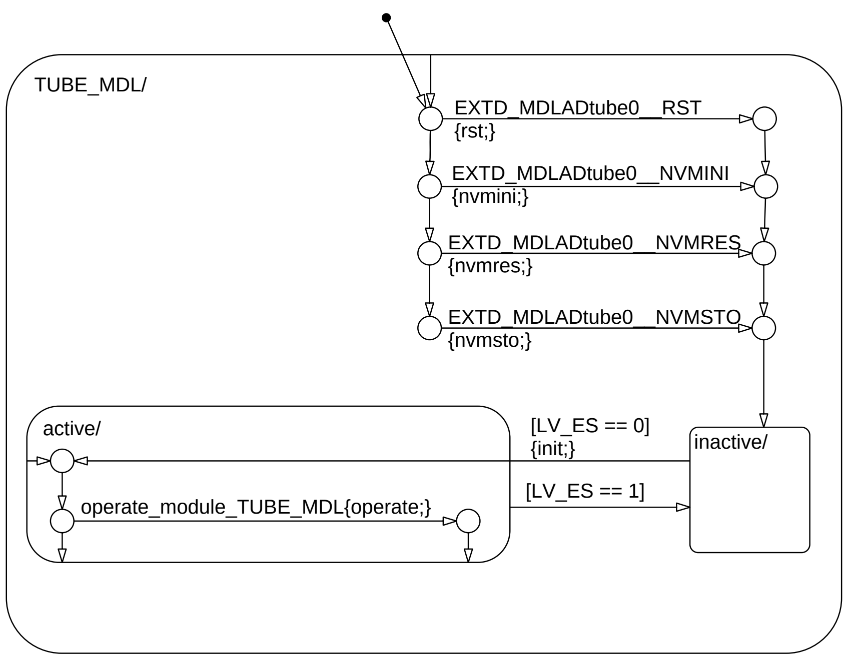


Figure 50 EXTD_MDLADtube0/ APP_CDN/ Chart

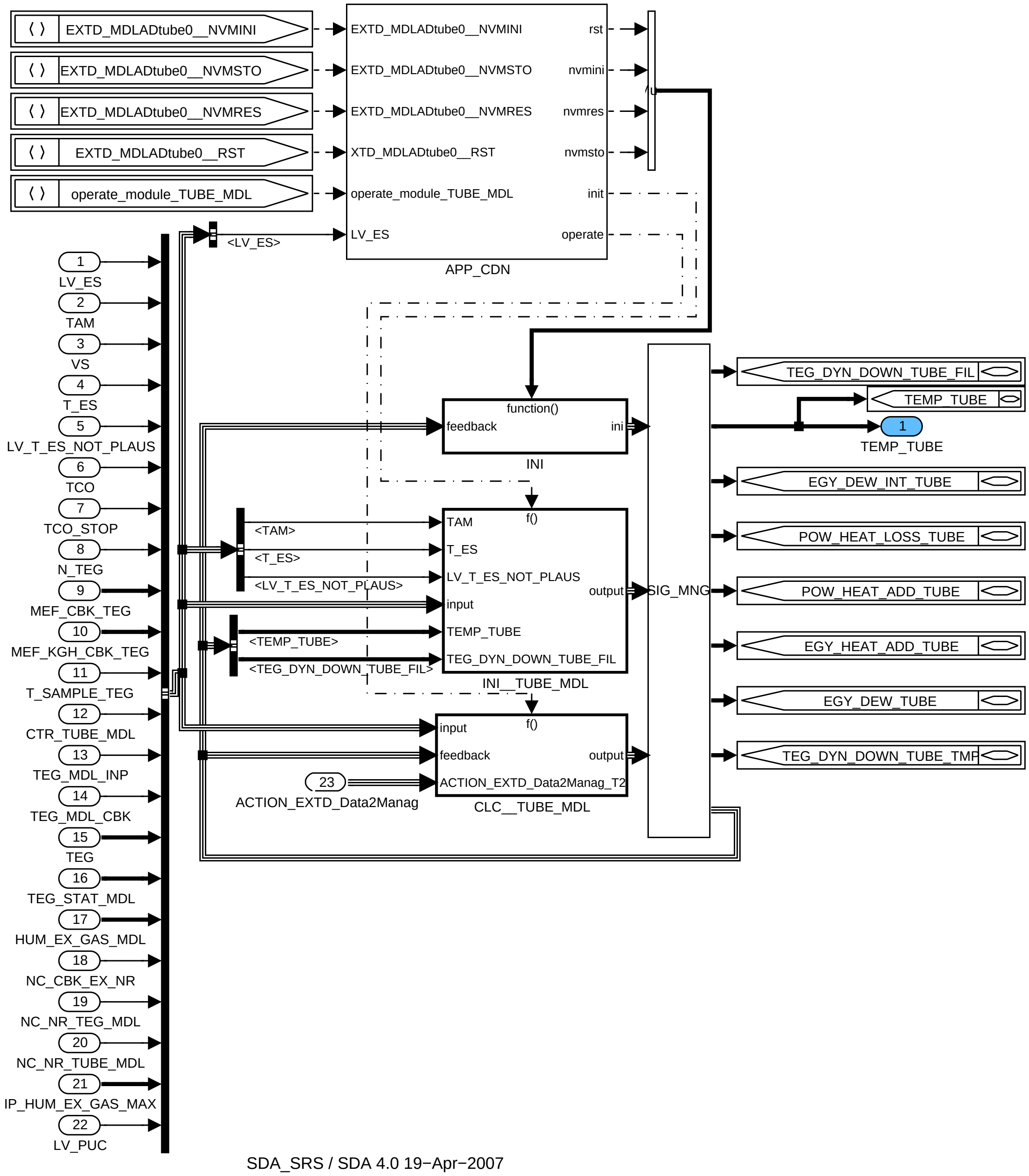


Figure 51 EXTD_MDLADtube0

7.43.1.1 Initialisation

Reset

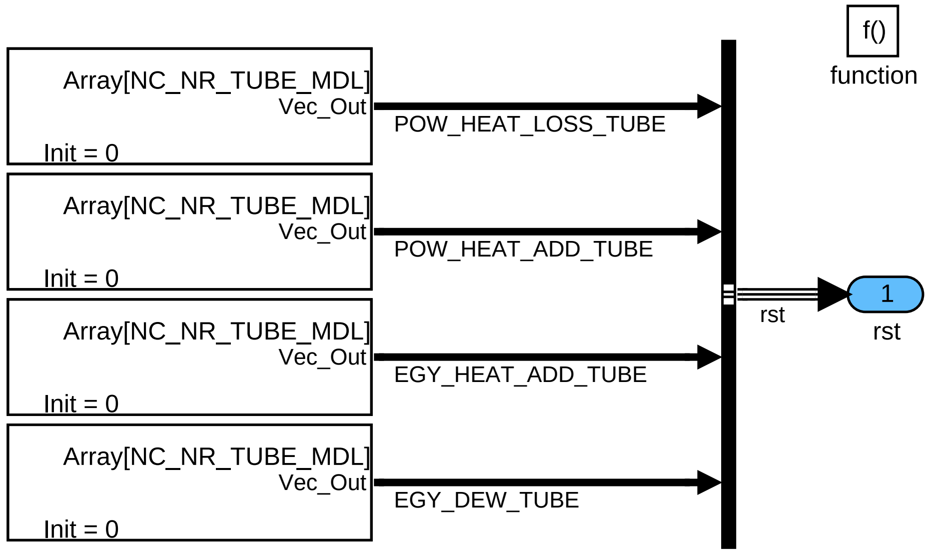


Figure 52 EXTD_MDLADtube0/ INI/ RST

Initialisation in case of brand new ECU or checksum error

TEMP_TUBE[NC_NR_TUBE_MDL] = 0°C (1112H)

TEG_DYN_DOWN_TUBE_FIL[NC_NR_TUBE_MDL] = 0°C (1112H)

EGY_DEW_INT_TUBE[NC_NR_TUBE_MDL] = 0kJ (0H)

TEG_DYN_DOWN_TUBE_TMP[NC_NR_TUBE_MDL] = 0°C (1112H)

Read NVMY

Read TEMP_TUBE[NC_NR_TUBE_MDL] and EGY_DEW_INT_TUBE[NC_NR_TUBE_MDL] from NVMY. Initialisation of TEG_DYN_DOWN_TUBE_FIL[NC_NR_TUBE_MDL] with TEMP_TUBE[NC_NR_TUBE_MDL].

Store NVMY

With LC_NO_TEMP_TUBE_CAL_ACT = 1 it can be chosen to store the value of TEG_DYN_DOWN_TUBE_FIL instead of TEMP_TUBE in EEPROM. This allows initialisation at next driving cycle even if TEMP_TUBE calculation is not calibrated.

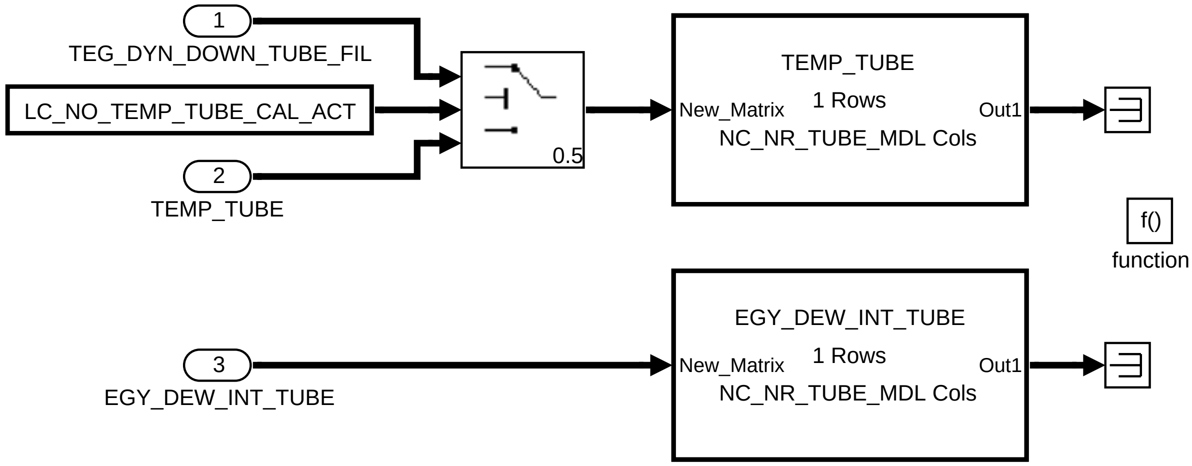


Figure 53 EXTD_MDLADtube0/ INI/ NVMSTO

7.43.1.2 INI__TUBE_MDL

For the "tube exhaust gas temperature model" the only value which has to be initialized is the tube wall temperature TEMP_TUBE because the tube is cooling down if the engine and also the ECU is off. There are two different calculations for the initialisation. One for a valid engine off duration time T_ES and one for the case that there is no engine off duration time is not available or not valid (LV_T_ES_NOT_VLD=1).

In both cases a factor is determined which represents the cooling down progress. If the factor is one the tube has not cooled down and is equal to the last tube wall temperature stored in the non volatile memory. If the factor is zero TEMP_TUBE is initialized with the ambient temperature TAM.

In the case that there is a valid engine off duration time available the factor is determined by a map versus T_ES.

In the other case the factor is determined by a map versus the coolant temperature cooling down progress RATIO_COOL_DOWN which is calculated out off TCO, TCO_STOP and TAM.

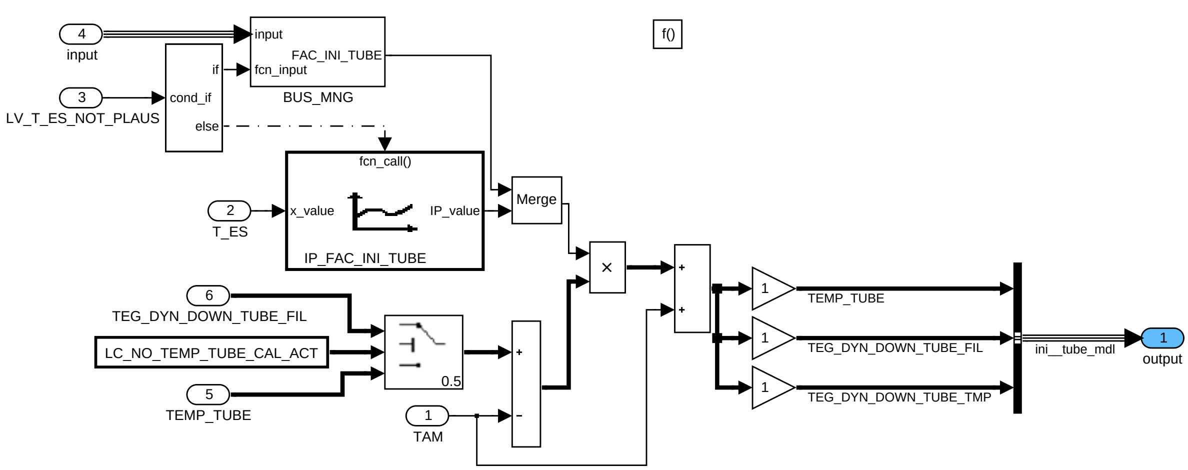


Figure 54 EXTD_MDLADtube0/ INI__TUBE_MDL

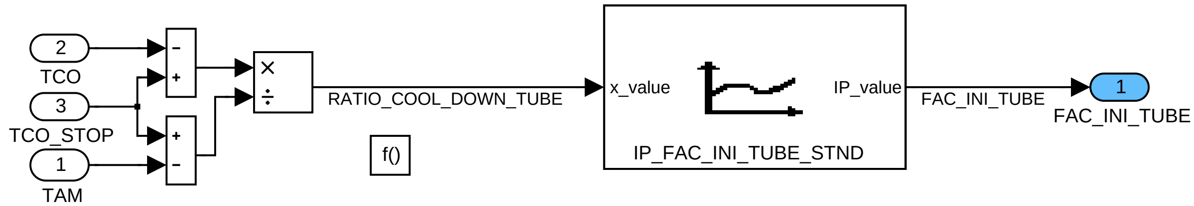


Figure 55 EXTD_MDLADtube0/ INI__TUBE_MDL/ BUS_MNG/ CLC_FAC_INI_TUBE_STND

7.43.1.3 CLC__TUBE_MDL

The calculation of the tube model is split up in four part:

- selection of the input values out of the arrays: SEL_INP_TUBE
- calculation of the steady state model: STAT_MDL_TUBE
- calculation of the dynamic model with dew point simulation: DYN_MDL_TUBE
- the writing back of the result to the manager by the action

SEL_INP_TUBE

The manager of the exhaust gas temperature model tells the tube model which values out of the arrays has to be used for the calculation. Thereby TEG_MDL_CBK is the index for the exhaust bank and TEG_MDL_INP determines the input index.

STAT_MDL_TUBE

For the steady state model the same calibration data are used as in the dynamic model. Out of the calibrated heat losses and heat gains and due to the condition for steady state that losses and gains are equal the exhaust gas temperature down tube for steady state is calculated with this equation.

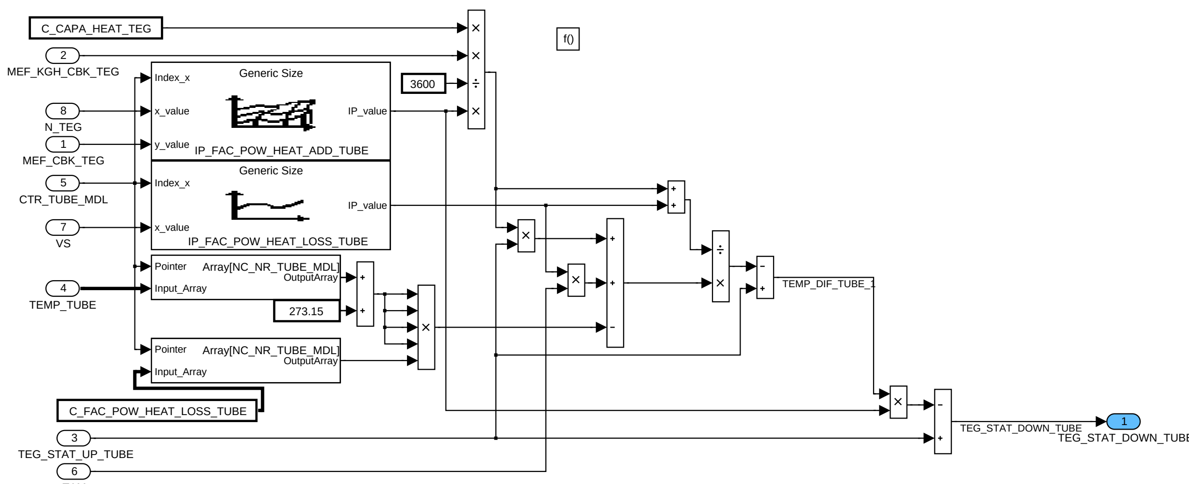


Figure 56 EXTD_MDLADtube0/ CLC__TUBE_MDL/ STAT_MDL_TUBE

DYN_MDL_TUBE

The dynamic part of the tube model is again split up in six sub-functions:

- calculation of the heat flows out of the calibration data: CLC__POW_HEAT_FLOW_TUBE

- calculation of the heat energy added to the tube during sample time of the model: CLC__EGY_HEAT_ADD_TUBE
- calculation of the exhaust gas temperature downstream the tube: CLC__TEG_DYN_DOWN_TUBE
- filtering of the calculated exhaust gas temperature downstream tube: CLC__TEG_DYN_DOWN_TUBE_FIL
- dew point calculations for the tube: CLC__DEW_TUBE
- determination of the temperature in/decrease of the tube wall temperature: CLC__TEMP_TUBE

CLC__POW_HEAT_FLOW_TUBE

In this sub-function the heat flows from the exhaust gas to the tube (heat gain) and the heat flow form the tube to the ambient (heat losses) are calculated based on the formulas:

Heat transfer of the exhaust gas to the tube:

$P_{\rightarrow tube} = \dot{Q} = \dot{m} \times c_p \times \left(t_{in} - t_{out}\right)$ with:

$\dot{m}$ = mass exhaust gas flow = MEF_KGH_CBK_TEG[kg/h]/3600[s/h]

c_p = heat capacity of the exhaust gas = C_CAPA_HEAT_TEG[kJ/(kg*K)]

t_{in} = inlet temperature = TEG_DYN_UP_TUBE

t_{out} = outlet temperature = TEG_DYN_DOWN_TUBE

$(t_{in} - t_{out}) = FAC\_POW\_HEAT\_ADD\_TUBE \times (t_{in} - t_{wall})$

Heat transfer from the tube to the ambient:

$P_{tube\rightarrow} = \dot{Q} = \alpha \times A \times \left(t_{wall} - t_{ambient}\right)$ with:

α = heat transfer coefficient

A = heat transfer surface

$\alpha \times A$ = IP_FAC_POW_HEAT_LOSS_TUBE

t_{wall} = wall temperature TEMP_TUBE

$t_{ambient}$ = ambient temperature TAM

Heat radiation to ambient (Stefan-Boltzmann):

$P_{tube\rightarrow} = \Phi_S = \varepsilon \times A \times T_{wall}^4$ with:

ε = heat radiation constant

A = heat radiation surface

$\varepsilon \times A$ = C_FAC_POW_HEAT_LOSS_TUBE

T_{wall} = wall temperature in Kelvin = TEMP_TUBE + 273.15

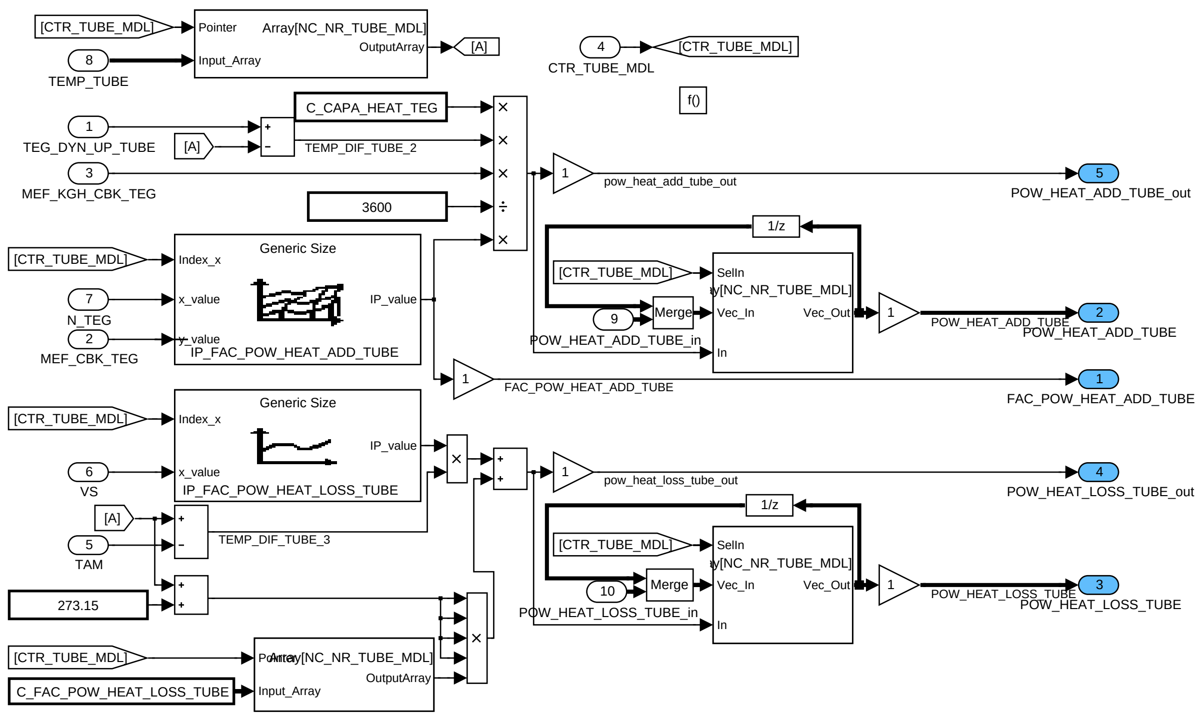


Figure 57 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__POW_HEAT_FLOW_TUBE

CLC__EGY_HEAT_ADD_TUBE

EGY_HEAT_ADD_TUBE is the overall heat energy which is transferred to the tube during sample time:

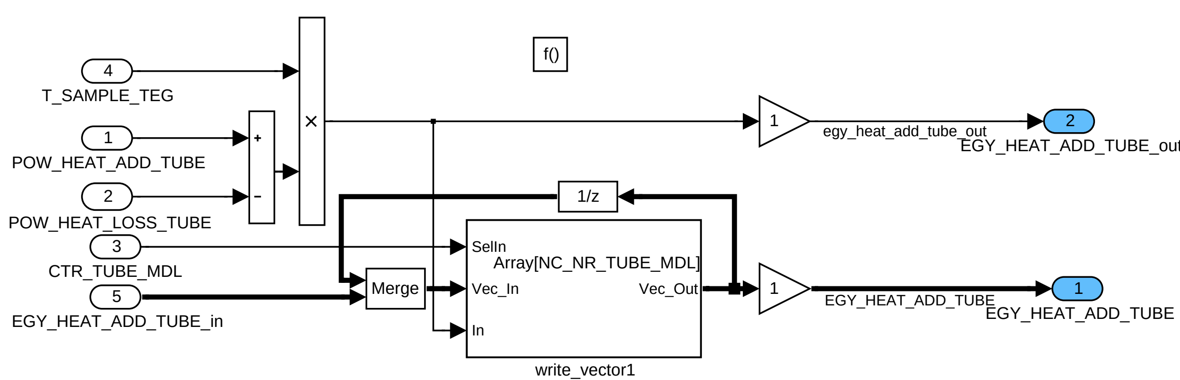


Figure 58 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__EGY_HEAT_ADD_TUBE

CLC__TEG_DYN_DOWN_TUBE

Out of the definition of FAC_POW_HEAT_ADD:

$$FAC\_POW\_HEAT\_ADD = \frac{(t_{in} - t_{out})}{(t_{in} - t_{wall})} =$$

$$\frac{(TEG\_DYN\_UP\_TUBE - TEG\_DYN\_DOWN\_TUBE)}{(TEG\_DYN\_UP\_TUBE - TEMP\_TUBE)} \Rightarrow$$

$$TEG\_DYN\_DOWN\_TUBE = TEG\_DYN\_UP\_TUBE -$$
$$FAC\_POW\_HEAT\_ADD\_TUBE \times (TEG\_DYN\_UP\_TUBE - TEMP\_TUBE)$$

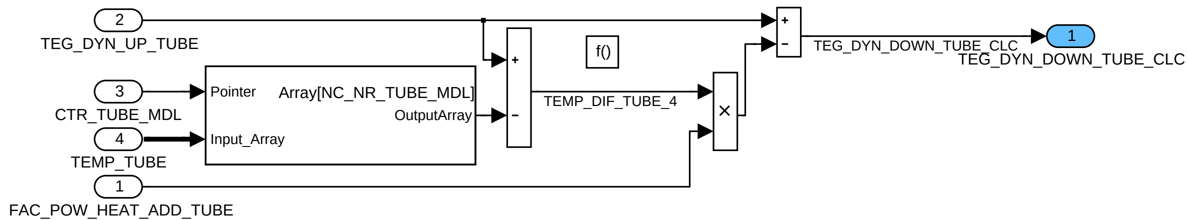


Figure 59 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__TEG_DYN_DOWN_TUBE

CLC__DEW_TUBE

The dew point calculation starts with the determination of the maximum of the humidity of the exhaust gas downstream tube HUM_DOWN_TUBE_MAX. The map for the calculation is dependent on the calculated exhaust gas temperature downstream tube. In the next step the bit LV_DEW_THD_TUBE is calculated in the sub-function CLC__LV_DEW_THD_TUBE. If this flag is set no dew point calculations are carried out (see NO_EGY_DEW_TUBE). Therefore EGY_DEW_TUBE and EGY_DEW_INT_TUBE are set to zero and the downstream exhaust gas humidity is equal to the upstream humidity.

If LV_DEW_THD_TUBE is not set CLC__EGY_DEW_INT_TUBE is calculated.

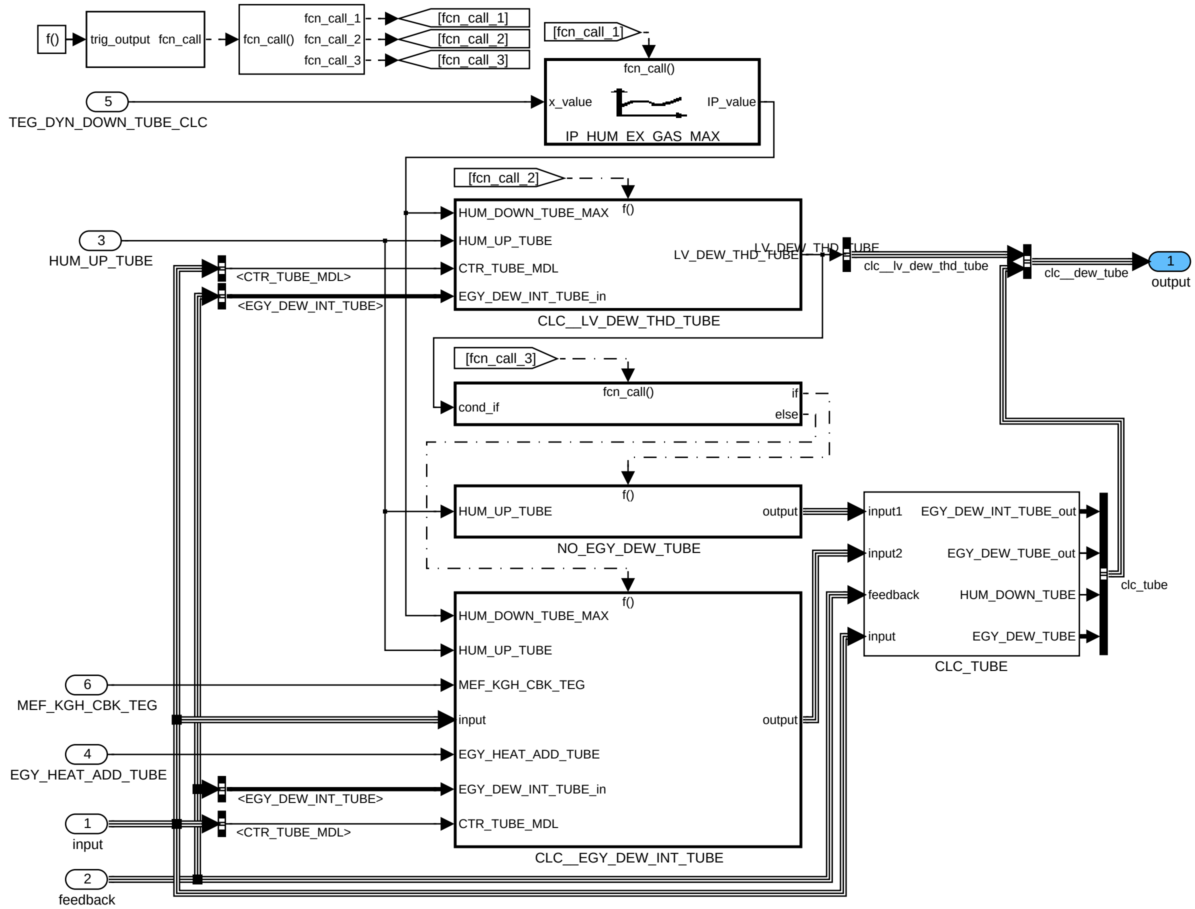


Figure 60 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE

CLC__LV_DEW_THD_TUBE

For the decision whether a dew point calculation is not necessary the following conditions have to be fulfilled:

- There is no condensation water in the tube: EGY_DEW_INT_TUBE = 0 and
- The exhaust gas temperature is high enough to take away the incoming water gaseous: HUM_DOWN_TUBE_MAX > HUM_UP_TUBE

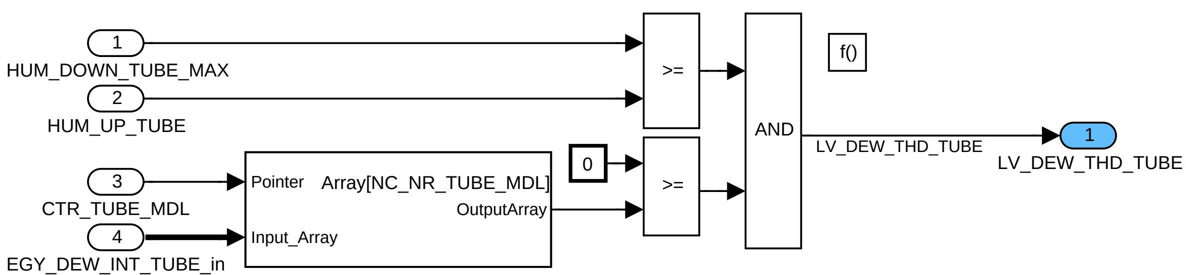


Figure 61 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ CLC__LV_DEW_THD_TUBE

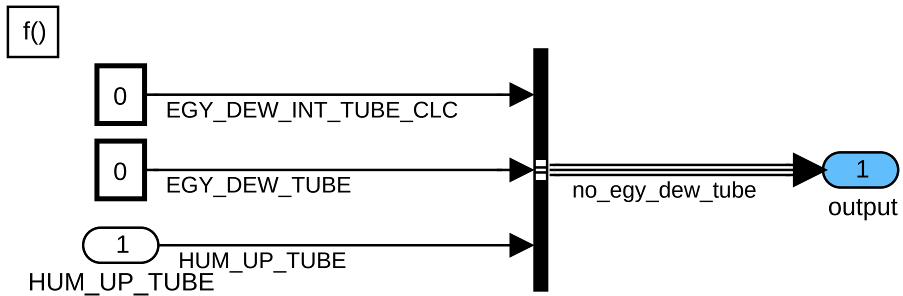


Figure 62 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ NO_EGY_DEW_TUBE

CLC__EGY_DEW_INT_TUBE

If LV_DEW_THD_TUBE is not set the following calculation is done:

At the first step it is necessary to decide whether condensation water film is build up or reduced. If the exhaust gas is to cold to take all water with it (HUM_DOWN_TUBE_MAX < HUM_UP_TUBE) the water film is build up and the dew energy EGY_DEW_TUBE is positive (EGY_DEW_POS is calculated). The other way round (HUM_DOWN_TUBE > HUM_UP_TUBE) the water film is reduced (EGY_DEW_NEG is calculated). The integrated dew energy EGY_DEW_INT_TUBE is in- or decreasing dependent on EGY_DEW_TUBE.

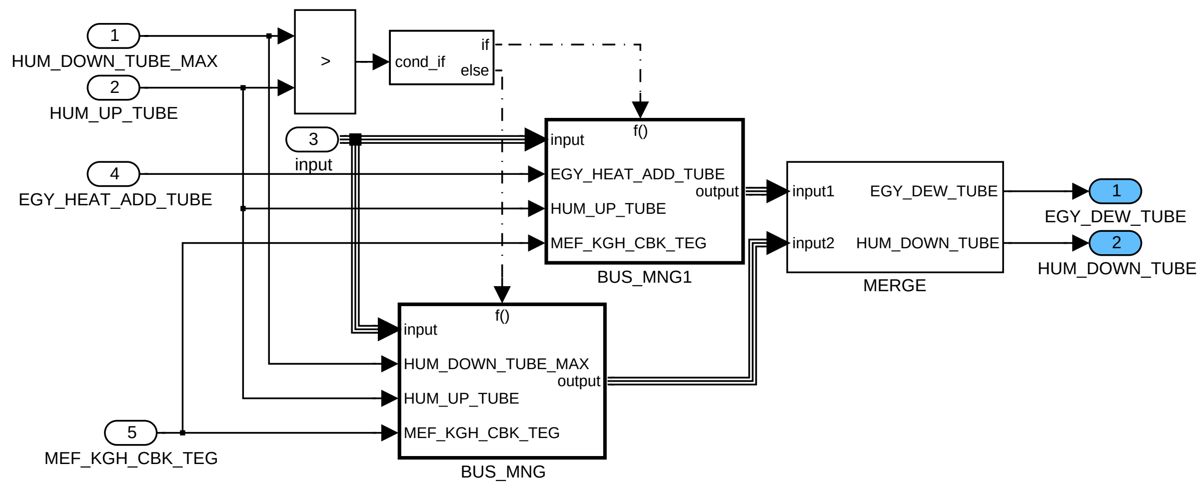


Figure 63 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ CLC__EGY_DEW_INT_TUBE/

CLC_XX_TUBE

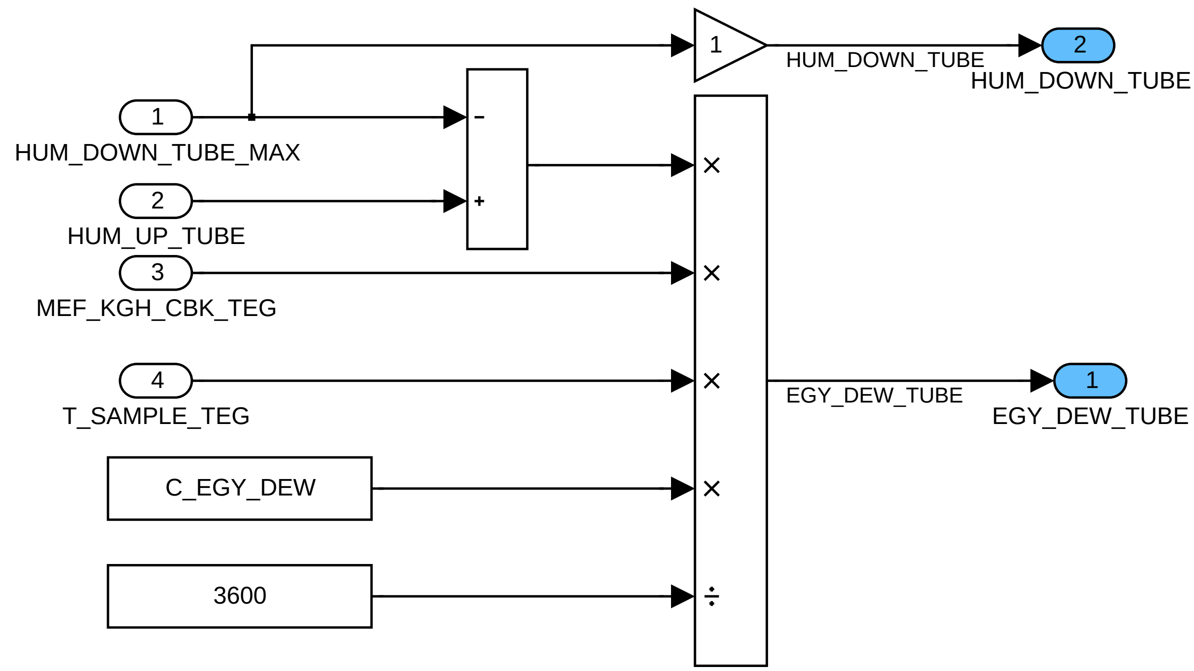


Figure 64 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ CLC__EGY_DEW_INT_TUBE/ CLC_XX_TUBE/ BUS_MNG/ EGY_DEW_POS

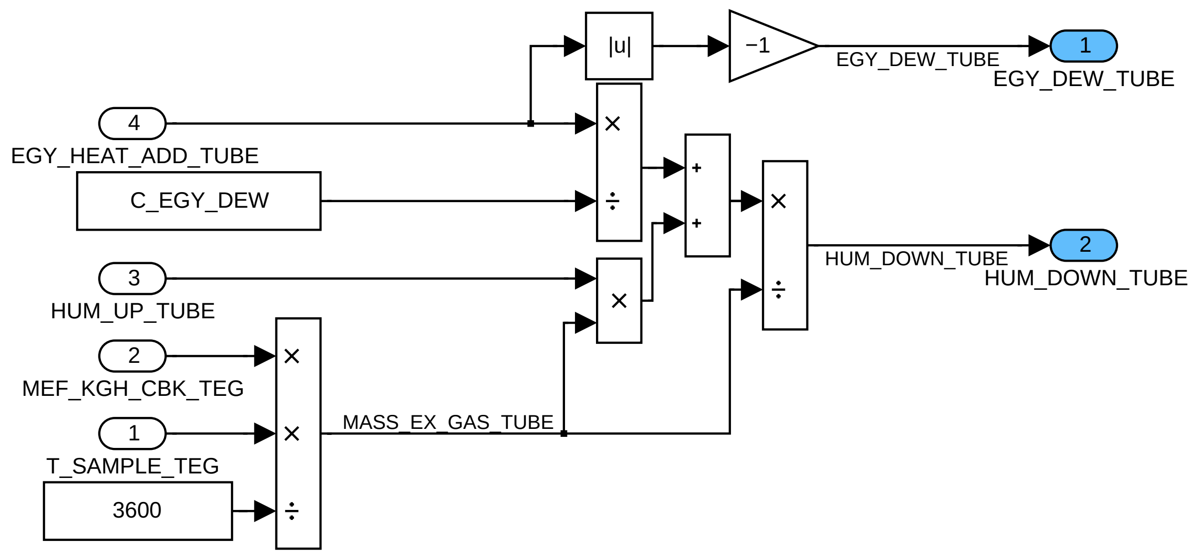


Figure 65 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ CLC__EGY_DEW_INT_TUBE/ CLC_XX_TUBE/ BUS_MNG1/ EGY_DEW_NEG

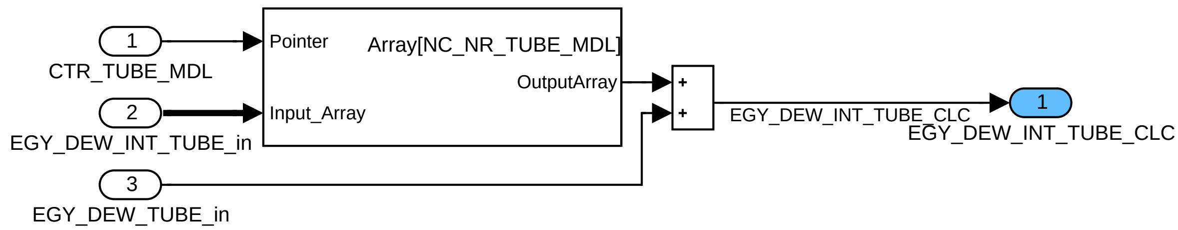


Figure 66 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__DEW_TUBE/ CLC__EGY_DEW_INT_TUBE/ CLC_EGY_DEW_INT_TUBE

CLC__TEMP_TUBE

If the dew point calculation is finished and EGY_DEW_TUBE is calculated the new tube wall temperature is calculated. Therefore the tube wall temperature change is calculated:

$$\Delta t_{wall} = \frac{Q}{m \times c_p} = \frac{EGY\_HEAT\_ADD\_TUBE + EGY\_DEW\_TUBE}{IP\_CAPA\_HEAT\_TUBE}$$

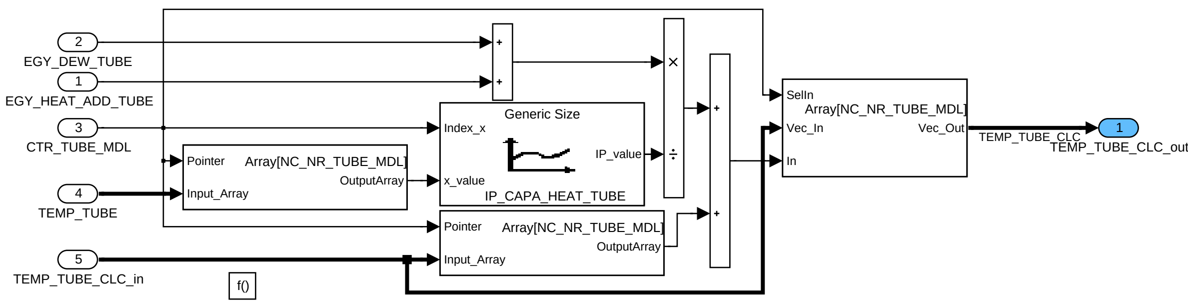


Figure 67 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__TEMP_TUBE

CLC__TEG_DYN_DOWN_TUBE_FIL

To simulate the measured behaviour of the exhaust gas temperature downstream tube it is possible to filter the previously calculated TEG_DYN_DOWN_TUBE. Due to the fact that this would only simulate the sensor filtering it is possible to decide whether the filtered value is used in the next component model. Therefore the switch LC_TEG_DYN_DOWN_TUBE_FIL_SEL has to be set to one. The filtered visible value TEG_DYN_DOWN_TUBE_FIL is always calculated to compare the model with the measured value.

Once no proper TEMP_TUBE calibration is done, LC_NO_TEMP_TUBE_CAL_ACT=1 allows switching between different filtering maps IP_CRLC_TEG_DYN_DOWN_TUBE[NC_NR_TUBE_MDL] and IP_CRLC_TEG_DYN_DOWN_TUBE_DEC[NC_NR_TUBE_MDL] depending on increasing or decreasing exhaust gas temperatures inside the tube. LC_NO_TEMP_TUBE_CAL_ACT=1 allows also activating an additional map for vehicle speed dependancy IP_FAC_VS_TEG[NC_NR_TUBE_MDL].

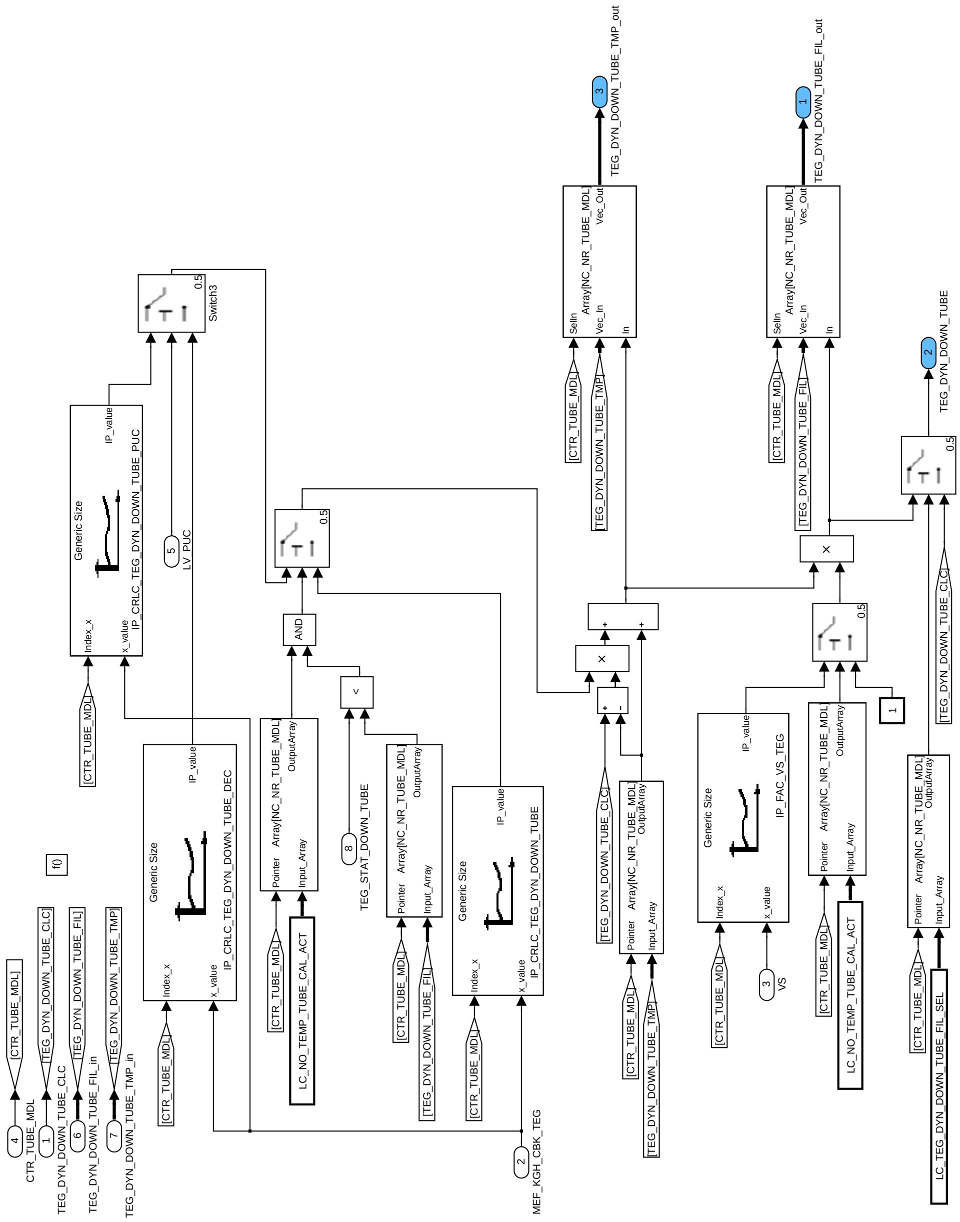


Figure 68 EXTD_MDLADtube0/ CLC__TUBE_MDL/ DYN_MDL_TUBE/ CLC__TEG_DYN_DOWN_TUBE_FIL

7.44 Catalyst exhaust gas temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_CAT[NC_NR_CAT_MDL]	O/V/S	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
catalyst temperatures					
TEMP_CAT_STAT[NC_NR_CAT_MDL]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
steady state catalyst temperatures					
POW_EXO_CAT_STAT[NC_NR_CAT_MDL]	V	0...FFFFH	0...65.535	0.001	kW
steady state exothermic of the different catalysts					
POW_HEAT_ADD_CAT_1[NC_NR_CAT_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat flow from exhaust gas to the first part of the catalyst					
EGY_HEAT_ADD_CAT_2[NC_NR_CAT_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
sum of all heat energy added to the second part of the catalyst					
EGY_DEW_CAT_2[NC_NR_CAT_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
dew energy added to the second part of the catalyst					
EGY_DEW_INT_CAT_2[NC_NR_CAT_MDL]	V/S	0...FFFFH	0...327.675	0.005	kJ
dew energy integral of the tubes					
CTR_TEG_LOAD_H[NC_NR_CAT_MDL]	V	0...FFFFH	0...65.535	0.001	-
Counter of high load determination for EXTD					
CTR_TEG_LOAD_H_PUC[NC_NR_CAT_MDL]	V	0...FFFFH	0...65.535	0.001	-
Counter of high load determination for EXTD - frozen for PUC					
TEMP_CAT_DIF_CLC[NC_NR_CAT_MDL]	V	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
Temperature difference between dynamic and static catalyst temperature					
EGY_HEAT_ADD_CAT_1[NC_NR_CAT_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
sum of all heat energy added to the first part of the catalyst					
EGY_DEW_CAT_1[NC_NR_CAT_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
dew energy added to the first part of the catalyst					
POW_EXO_CAT_DYN[NC_NR_CAT_MDL]	V	0...FFFFH	0...65.535	0.001	kW
exothermic of the different catalysts					
POW_HEAT_LOSS_CAT[NC_NR_CAT_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat losses of the catalyst					
EGY_DEW_INT_CAT_1[NC_NR_CAT_MDL]	V/S	0...FFFFH	0...327.675	0.005	kJ
dew energy integral of the tubes					
POW_HEAT_ADD_CAT_2[NC_NR_CAT_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat flow from exhaust gas to the second part of the catalyst					

Input data:

LV_ES	TAM	VS	LV_PUC
T_PUC	EFF_CAT_DIAG[NC_CBK_EX_NR]	T_ES	LV_T_ES_NOT_PLAUS
TCO	TCO_STOP	N_TEG	MAF_TEG
MEF_KGH_CBK_TEG[NC_CBK_EX_NR]	LAMB_CBK_TEG_ENG[NC_CBK_EX_NR]	LAMB_CBK_TEG_CAT[NC_CBK_EX_NR]	POW_EXO_CAT_ADD_CUS[NC_NR_CAT_MDL]
FAC_POW_EXO_CAT_CUS[NC_NR_CAT_MDL]	T_SAMPLE_TEG	CTR_CAT_MDL	TEG_MDL_INP

TEG_MDL_CBK	TEG[NC_NR_TEG_MDL]	TEG_STAT_MDL[NC_NR_TEG_MDL]	HUM_EX_GAS_MDL[NC_NR_TEG_MDL]
NC_CBK_EX_NR	NC_NR_TEG_MDL	NC_NR_CAT_MDL	C_EGY_DEW
C_CAPA_HEAT_TEG	IP_HUM_EX_GAS_MAX		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_TEG_LOAD_H_MAX[NC_NR_CAT_MDL]	1	0...FFFFH	0...65.535	0.001	-
Max limitation for counter of high load determination for EXTD					
C_FAC_POW_HEAT_LOSS_CAT[NC_NR_CAT_MDL]	1	0...FFFFH	0...29.8020412	4.5475E-4	nW/K^4
factor for the calculation of heat losses of the tube due to radiation					
ID_CTR_DEC_TEG_PUC[NC_NR_CAT_MDL]	6	8000...7FFFH	-32.768...32.767	0.001	-
LDP_T_PUC_ID_CTR_DEC_TEG	6	0...FFFFH	0...655.35	0.01	s
Decrement value for counter high load determination for EXTD at PUC					
IP_FAC_INI_CAT	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_T_ES_1_EXTD	8	0...FFFFH	0...6.5535E+4	1	min
factor for catalyst cooling down dependent on engine off duration time					
IP_FAC_INI_CAT_STND	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_RATIO_COOL_DOWN_1_EXTD	8	0...FFH	0...0.99609375	0.00390625	-
factor for catalyst cooling down in case of non plausible engine off duration time					
ID_CTR_INC_DEC_TEG_LOAD_H[NC_NR_CAT_MDL]	8x4	8000...7FFFH	-32.768...32.767	0.001	-
LDP_MEF_KGH_CBK_TEG_ID_CTR_INC	8	0...FFFFH	0...4.09594E+3	0.0625	kg/h
LDP_TEG_ID_CTR_INC	4	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Increment / decrement value for counter high load determination for EXTD					
IP_CAPA_HEAT_CAT_1[NC_NR_CAT_MDL]	8x8	0...FFFFH	0...65.535	0.001	kJ/K
LDPM_TEMP_1_EXTD	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_TEMP_CAT_DIF_CLC_1_EXTD	8	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
heat capacity of the first part of the different catalysts					
IP_CAPA_HEAT_CAT_2[NC_NR_CAT_MDL]	8x8	0...FFFFH	0...65.535	0.001	kJ/K
LDPM_TEMP_1_EXTD	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDPM_TEMP_CAT_DIF_CLC_1_EXTD	8	0...FFFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
heat capacity of the second part of the different catalysts					
IP_FAC_LAMB_COR_POW_EXO_CAT[NC_NR_CAT_MDL]	10x12	0...FFH	0...2.55	0.01	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
weighting factor versus operating point for the lambda dependency of the catalyst exothermic					
IP_FAC_LAMB_POW_EXO_CAT	8x8	0...FFH	-1...1.55	0.01	-
LDPM_LAMB_ENG_1_EXTD	8	0...7FFFH	0...31.9990234	9.76563E-4	-
LDP_LAMB_CAT_IP_FAC_LAMB_EXO	8	0...7FFFH	0...31.9990234	9.76563E-4	-
correction factor of catalyst exothermic dependent on lambda					
IP_FAC_POW_HEAT_LOSS_CAT[NC_NR_CAT_MDL]	8x10	0...FFFFH	0...1.02398437	1.5625E-5	kW/K
LDPM_VS_1_EXTD	8	0...FFH	0...255	1	km/h
LDPM_MEF_KGH_CBK_TEG_1_EXTD	10	0...FFFFH	0...4.09594E+3	0.0625	kg/h
calculation factor for the calculation of the heat flow to ambient of the differnt catalystst					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TEMP_POW_EXO_CAT	8x4	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_TEMP_CAT_IP_FAC_EXO_TEMP	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
LDP_EFF_CAT_DIAG_IP_TEMP_EXO	4	0...FFH	0...1.9921875	0.0078125	-
light off factor for catalyst exthermic dependent on catalyst temperature and catalyst diagnosis value					
IP_POW_EXO_CAT[NC_NR_CAT_MDL]	10x12	0...FFFFH	0...65.535	0.001	kW
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MAF_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
basic catalyst exothermic					
IP_POW_EXO_CAT_PUC[NC_NR_CAT_MDL]	6x8	0...FFFFH	0...65.535	0.001	kW
LDP_T_PUC_IP_POW_EXO_CAT_PUC	6	0...FFFFH	0...655.35	0.01	s
LDP_CTR_LOAD_IP_POW_EXO_CAT_PUC	8	0...FFFFH	0...65.535	0.001	-
Catalyst exothermic at PUC					

Import actions:

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)
writes back the calculated data back to the arrays of the manager (TEG_DYN_MDL, ...)

7.44.1 General information:

The "catalyst exhaust gas temperature model" is part of the complete exhaust gas temperature model. This part models the behaviour of a catalyst where the exhaust gas is heated up in the first part of the catalyst by the exothermic reaction and cooled down in the second part. Therefore the model is also split up into two parts.

The catalyst monolith temperature TEMP_CAT is calculated in the fist part and the outlet exhaust gas temperature TEG_DYN_DOWN_CAT is calculated in the second part. TEG_DYN_UP_CAT and TEG_DYN_DOWN_CAT are no visible values because they are just interim values for the calculation. The manager of the exhaust gas temperature model stores the result of the calculation in the array TEG with the correct index.

Initialisation of SRAM value at checksum error:
TEMP_CAT[NC_NR_CAT_MDL] = 0°C (1112H)
EGY_DEW_INT_CAT_1[NC_NR_CAT_MDL] = 0 (0H)
EGY_DEW_INT_CAT_2[NC_NR_CAT_MDL] = 0 (0H)

Application Condition

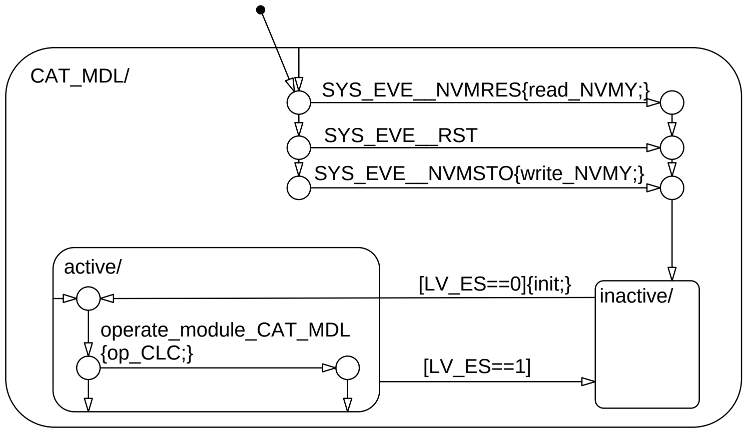


Figure 69 EXTD_MDLADcat0/ APP_CDN/ Chart

Function Description

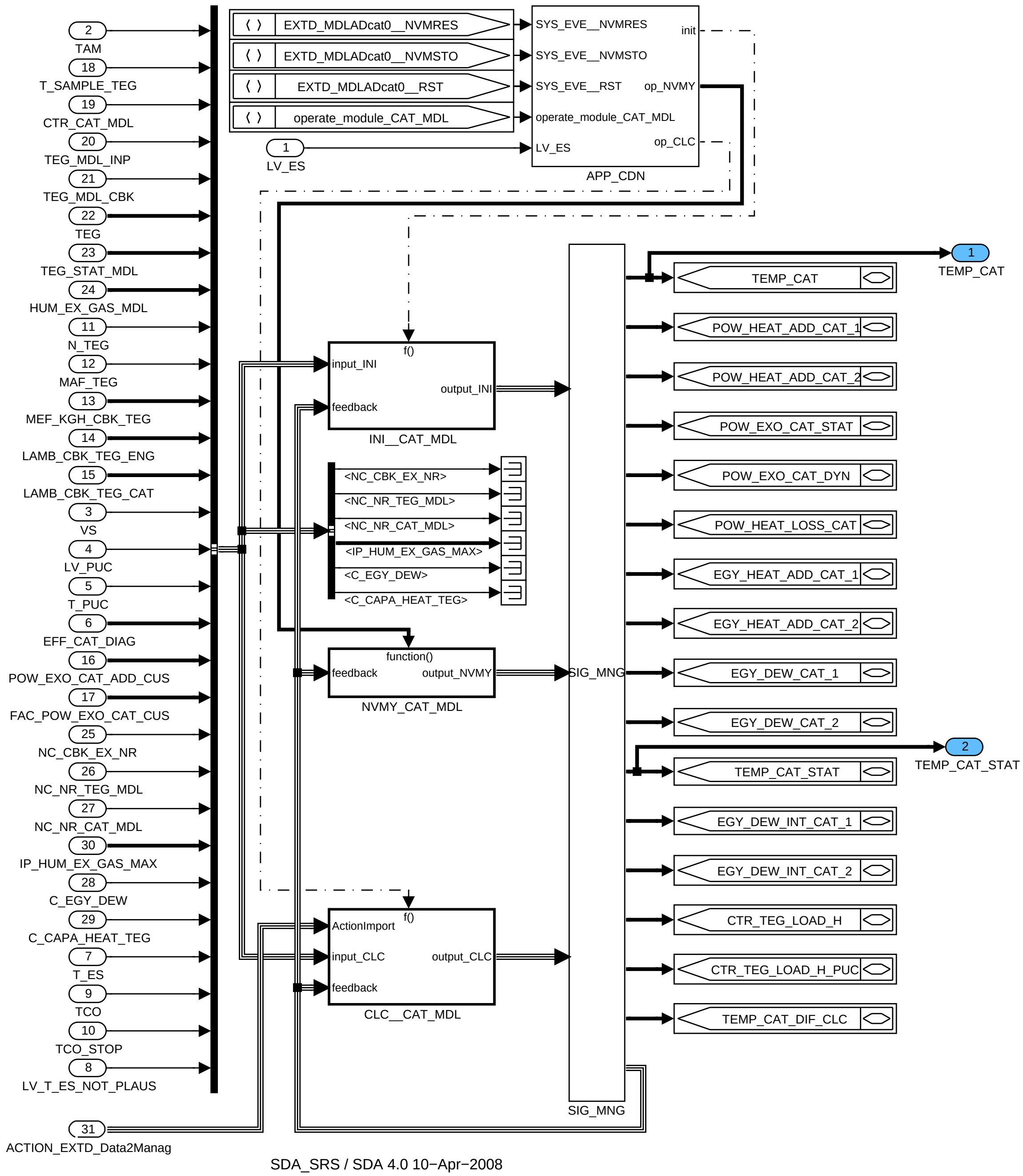


Figure 70 EXTD_MDLADcat0

7.44.1.1 NVMY

TEMP_CAT[NC_NR_CAT_MDL], EGY_DEW_INT_CAT_1[NC_NR_CAT_MDL] and EGY_DEW_INT_CAT_2[NC_NR_CAT_MDL] are read and write from NVMY.

7.44.1.2 INI__CAT_MDL

For the "catalyst exhaust gas temperature model" the only value which has to be initialized is the catalyst monolith temperature TEMP_CAT because the catalyst is cooling down if the engine and also the ECU is off. There are two different calculations for the initialisation. One for a valid engine off duration time T_ES and one for the case that there is no engine off duration time is not available or not valid (LV_T_ES_NOT_VLD=1).

In both cases a factor is determined which represents the cooling down progress. If the factor is one the catalyst has not cooled down and is equal to the last catalyst monolith temperature stored in the non volatile memory. If the factor is zero TEMP_CAT is initialized with the ambient temperature TAM.

In the case that there is a valid engine off duration time available the factor is determined by a map versus T_ES.

In the other case the factor is determined by a map versus the coolant temperature cooling down progress RATIO_COOL_DOWN which is calculated out off TCO, TCO_STOP and TAM.

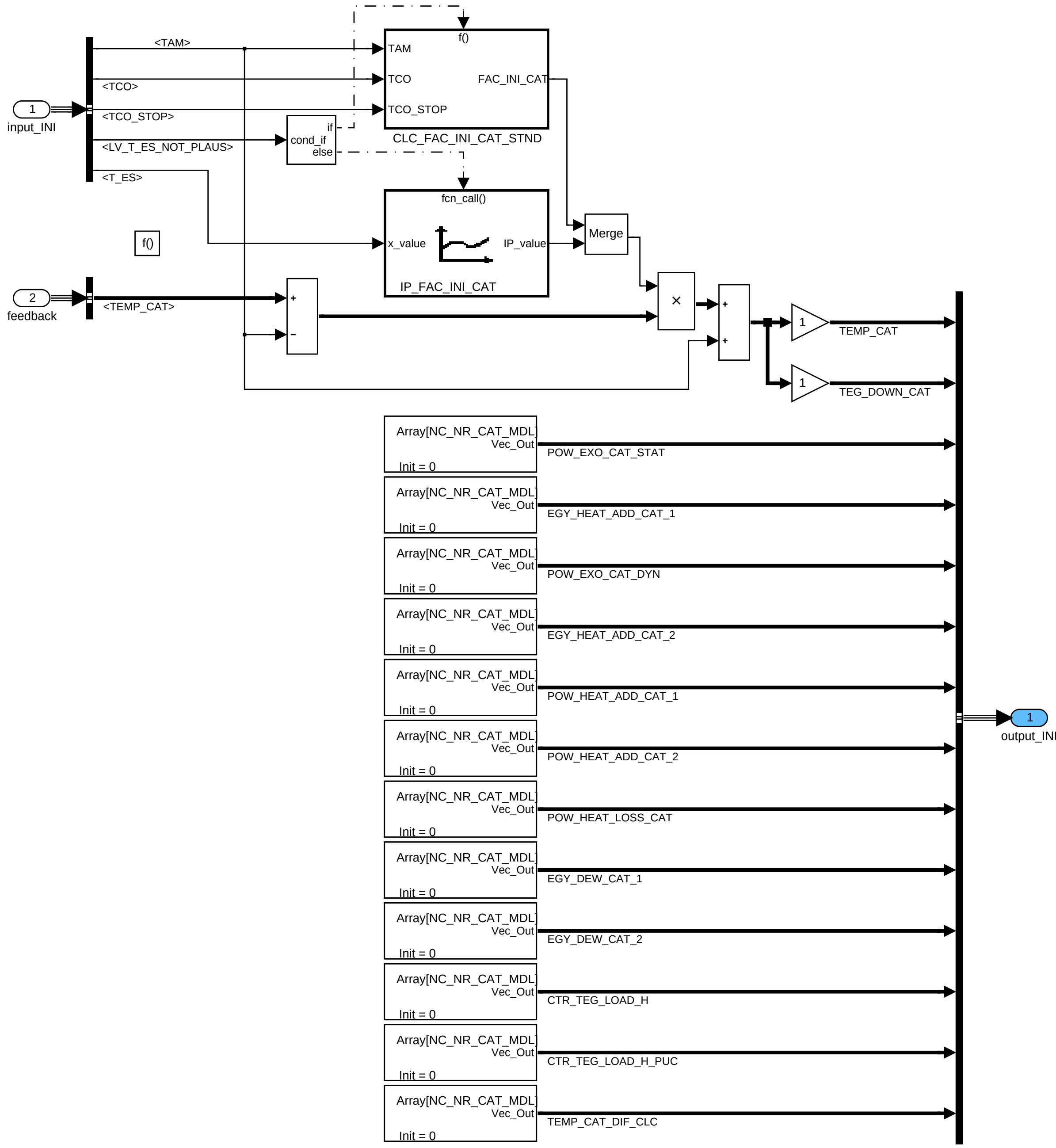


Figure 71 EXTD_MDLADcat0/ INI__CAT_MDL

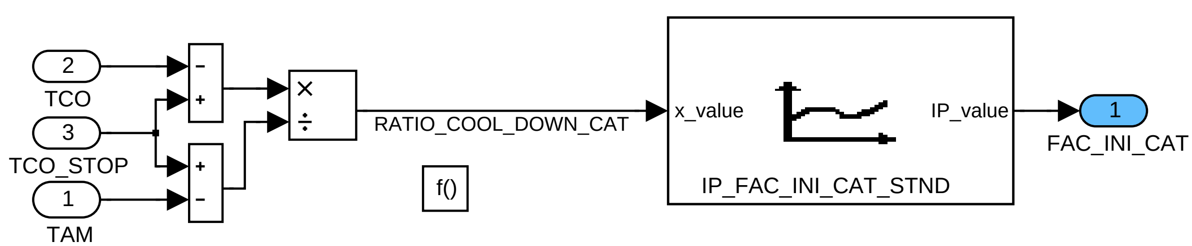


Figure 72 EXTD_MDLADcat0/ INI__CAT_MDL/ CLC_FAC_INI_CAT_STND

7.44.1.3 CLC__CAT_MDL

The calculation of the tube model is split up into five parts:

- selection of the input values out of the arrays: SEL_INP_CAT
- calculation of the exothermic: POW_EXO_CAT
- calculation of the steady state model: STAT_MDL_CAT
- calculation of the dynamic model with dew point simulation: DYN_MDL_CAT
- the writing back of the result to the manager by the action

POW_EXO_CAT

The basic exothermic power for lambda =1 without secondary is stored in the IP_POW_EXO_CAT. This value has to be corrected for different lambda by the map IP_FAC_LAMB_POW_EXO_CAT and this correction can be weighted versus engine operating point.

For PUC operation the exothermic power is normally zero because the exhaust gas is fresh air but at the transition to PUC the exothermic can become very high because of stored HC and CO. Therefore the map IP_POW_EXO_CAT_PUC can be calibrated with a factor greater than one for the first time after PUC operation. For long PUC operation time T_PUC the factor has to be zero.

High load counter CTR_TEG_LOAD_H_PUC used to consider high load operation duration for succeeding PUC condition temperature shape. Maps ID_CTR_INC_DEC_TEG_LOAD_H, ID_CTR_DEC_TEG_PUC and the calibration constant C_CTR_TEG_LOAD_H_MAX added for calculation of CTR_TEG_LOAD_H_PUC. This counter used for correction of the catalyst model via IP_POW_EXO_CAT_PUC during PUC.

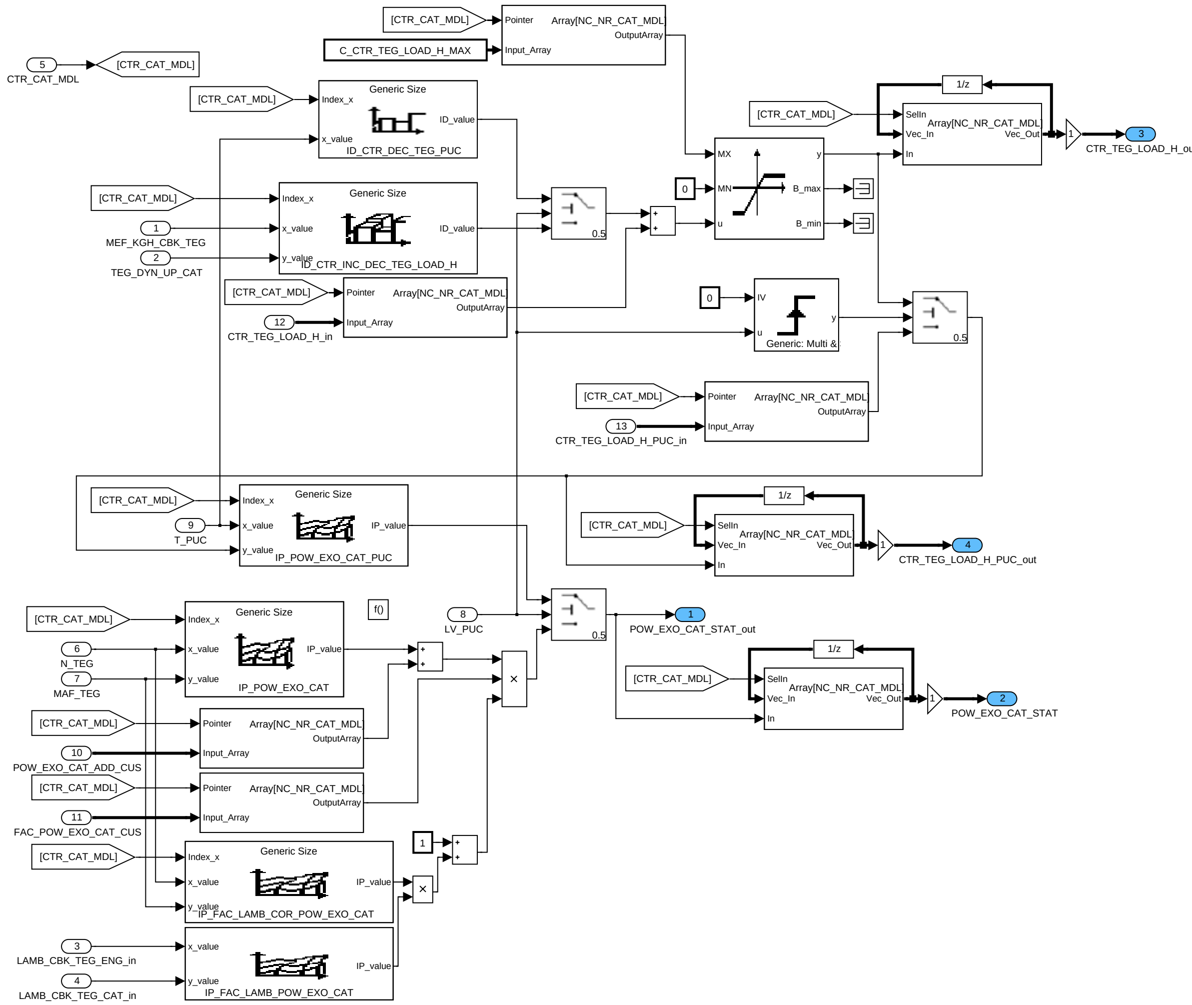


Figure 73 EXTD_MDLADcat0/ CLC__CAT_MDL/ POW_EXO_CAT

STAT_MDL_CAT

For the steady state model the same calibration data are used as in the dynamic model. During steady state conditions the catalyst monolith temperature TEMP_CAT_STAT for the first part is equal to the inlet steady state temperature TEG_STAT_UP_CAT plus the exothermic POW_EXO_CAT_STAT converted into a temperature dependent on the mass exhaust gas flow MEF_KGH_CBK_TEG and there heat capacity. For the second part TEG_STAT_DOWN_CAT is calculated in the same way but with now the heat losses are subtracted from the steady state monolith temperature TEMP_CAT_STAT.

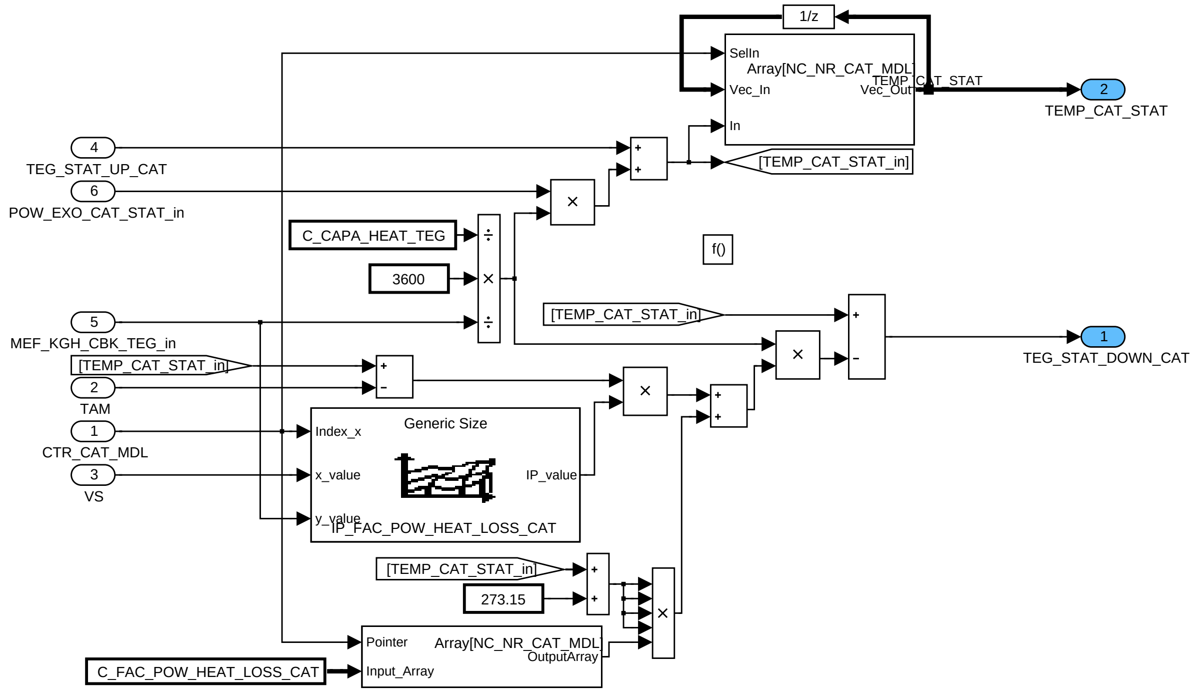


Figure 74 EXTD_MDLADcat0/ CLC__CAT_MDL/ STAT_MDL_CAT

DYN_MDL_CAT

The dynamic part of the catalyst model is again split up into six sub-functions:

- calculation of the heat flows: CLC__POW_HEAT_FLOW_CAT
- calculation of the heat energy: CLC__EGY_HEAT_ADD_CAT
- dew point calculations for the first part of the catalyst: CLC__DEW_CAT_1
- dew point calculations for the second part of the catalyst: CLC__DEW_CAT_2
- determination of the catalyst monolith temperature (first part): CLC__TEMP_CAT
- determination of the outlet exhaust gas temperature: CLC__TEG_DYN_DOWN_CAT

CLC__POW_HEAT_FLOW_CAT

The same formulas like in the "tube exhaust gas temperature model" are used for the following calculations. The only difference is that there is no factor FAC_POW_HEAT_ADD_CAT because it is assumed that the heat transfer coefficient for the heat transfer between exhaust gas and catalyst monolith is so big that the factor is one for all mass exhaust gas flows. A factor of one leads to the equality of exhaust gas temperature and monolith temperature. Additionally it is assumed that in the first part of the catalyst there are only heat gains due to exothermic reaction no heat losses and in the second part it is the other way round.

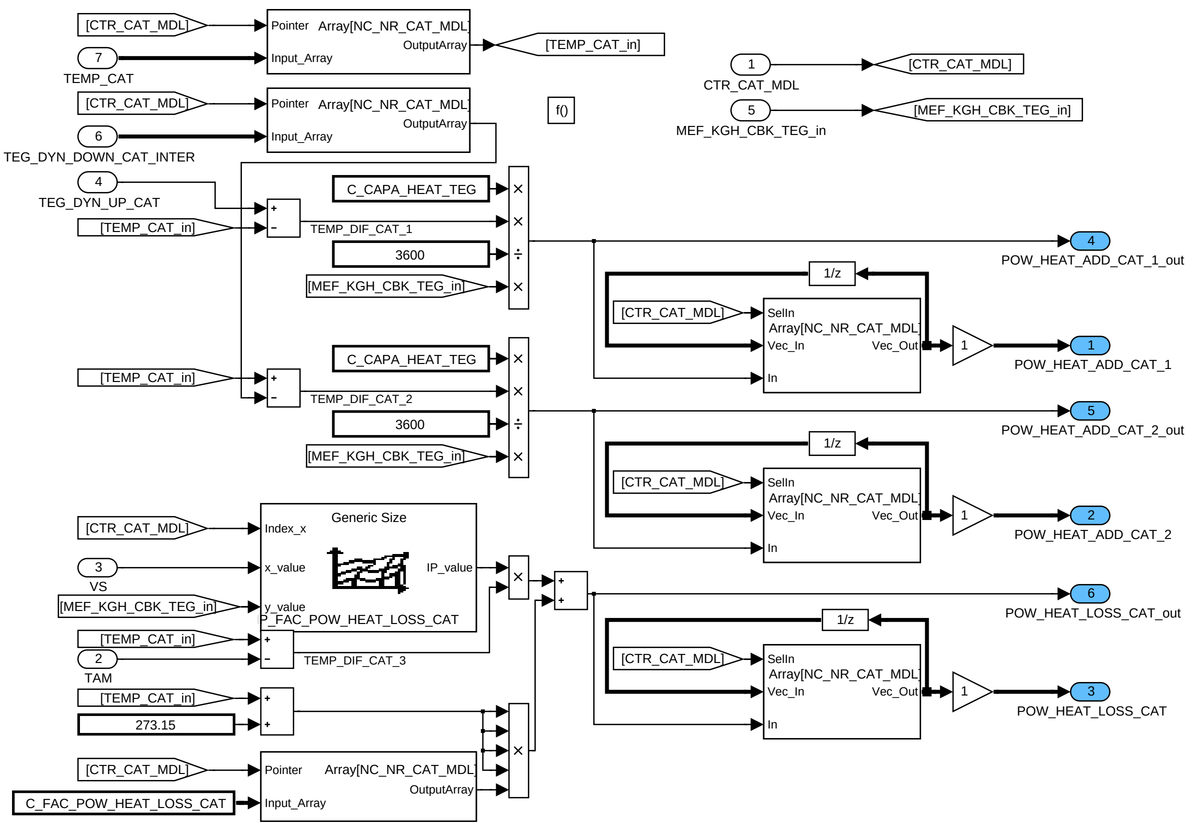


Figure 75 EXTD_MDLADcat0/ CLC__CAT_MDL/ DYN_MDL_CAT/ CLC__POW_HEAT_FLOW_CAT

CLC__EGY_HEAT_ADD_CAT

EGY_HEAT_ADD_CAT is the overall (except dew energy) heat energy added to the different parts of the catalyst. For the first part EGY_HEAT_ADD_CAT_1 is calculated out of the sum of POW_HEAT_ADD_CAT_1 and the POW_EXO_CAT_DYN multiplied with the sample time. In the second part the calculation is similar to a tube model: EGY_HEAT_ADD_CAT_2 is calculated out of the sum of heat gains and heat losses multiplied with the sample time.

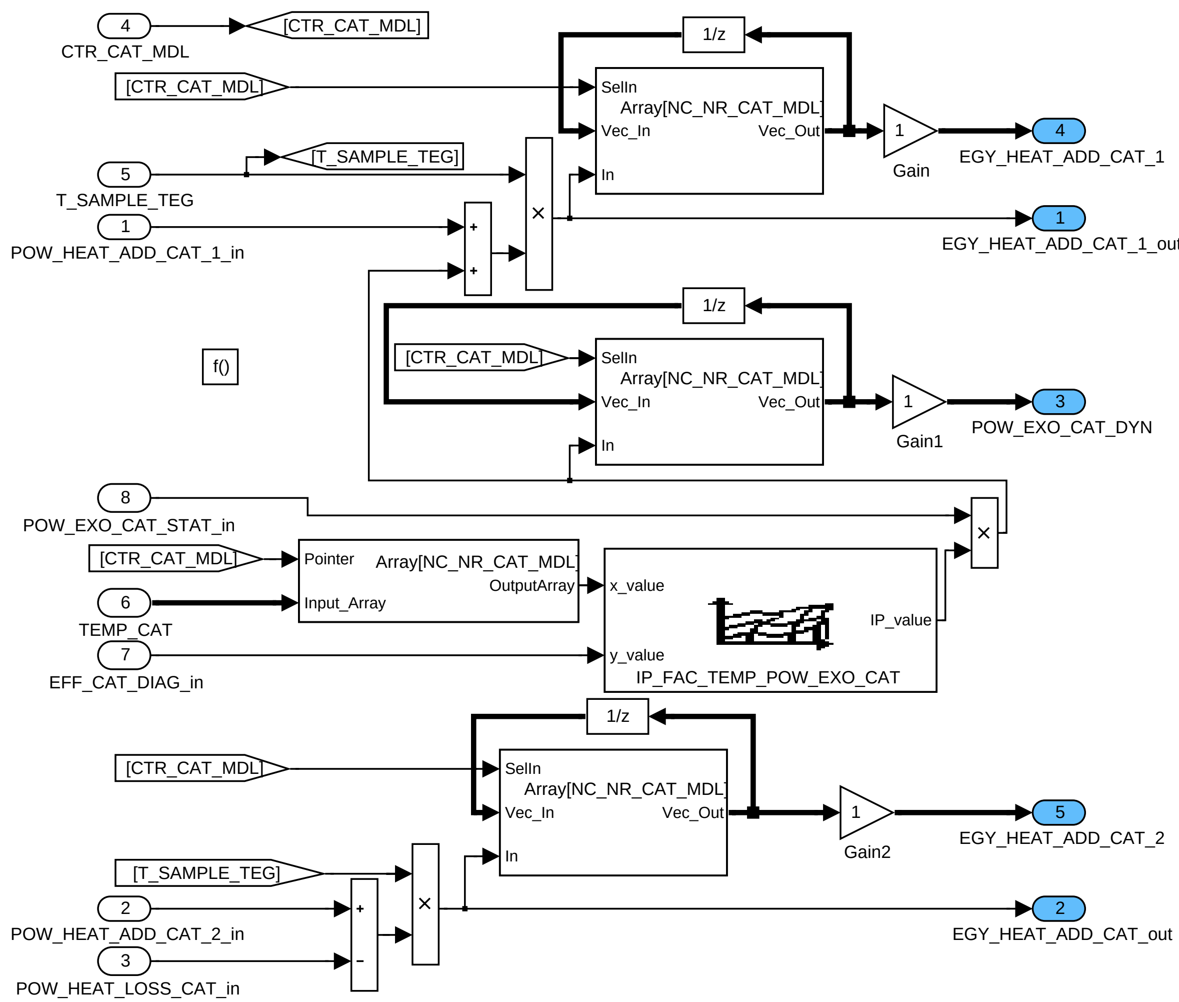


Figure 76 EXTD_MDLADcat0/ CLC__CAT_MDL/ DYN_MDL_CAT/ CLC__EGY_HEAT_ADD_CAT

CLC__DEW_CAT_1

All dew point calculations are similar to the dew point calculation in the "Tube exhaust gas temperature model": see CLC__DEW_TUBE

Replace:

CTR_TUBE_MDL → CTR_CAT_MDL
HUM_UP_TUBE → HUM_UP_CAT
HUM_DOWN_TUBE_MAX → HUM_CAT_MAX
HUM_DOWN_TUBE → HUM_CAT
TEG_DYN_DOWN_TUBE_CLC → TEMP_CAT
EGY_HEAT_ADD_TUBE → EGY_HEAT_ADD_CAT_1
EGY_DEW_TUBE → EGY_DEW_CAT_1
EGY_DEW_INT_TUBE → EGY_DEW_INT_CAT_1

CLC__DEW_CAT_2

All dew point calculations are similar to the dew point calculation in the "Tube exhaust gas temperature model": see CLC__DEW_TUBE

Replace:

CTR_TUBE_MDL → CTR_CAT_MDL
HUM_UP_TUBE → HUM_CAT
HUM_DOWN_TUBE_MAX → HUM_DOWN_CAT_MAX
HUM_DOWN_TUBE → HUM_DOWN_CAT
TEG_DYN_DOWN_TUBE_CLC → TEG_DYN_DOWN_CAT_INTER
EGY_HEAT_ADD_TUBE → EGY_HEAT_ADD_CAT_2
EGY_DEW_TUBE → EGY_DEW_CAT_2
EGY_DEW_INT_TUBE → EGY_DEW_INT_CAT_2

CLC__TEMP_CAT

If the dew point calculation is finished and EGY_DEW_CAT_1 was calculated the new catalyst monolith temperature is calculated. Therefore the catalyst monolith temperature change is calculated similar to the tube wall temperature in the tube model:

$$\Delta t_{monolith} = \frac{Q}{m \times c_p} = \frac{EGY\_HEAT\_ADD\_CAT + EGY\_DEW\_CAT\_1}{IP\_CAPA\_HEAT\_CAT\_1}$$

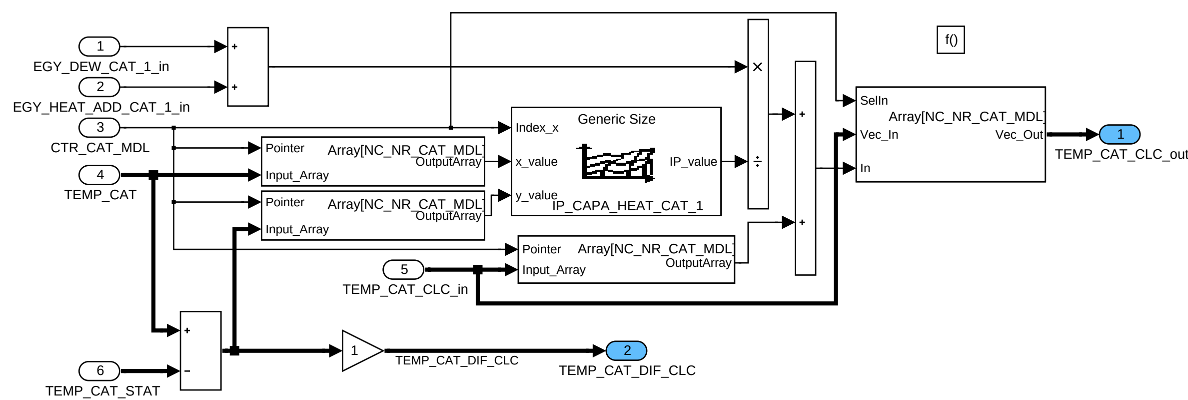


Figure 77 EXTD_MDLADcat0/ CLC__CAT_MDL/ DYN_MDL_CAT/ CLC__TEMP_CAT

CLC__TEG_DYN_DOWN_CAT

Due to the fact that the catalyst outlet exhaust gas temperature is equal to the monolith temperature of the second part the calculation of TEG_DYN_DOWN_CAT is equal to the calculation for TEMP_CAT but with all parameters of the second part.

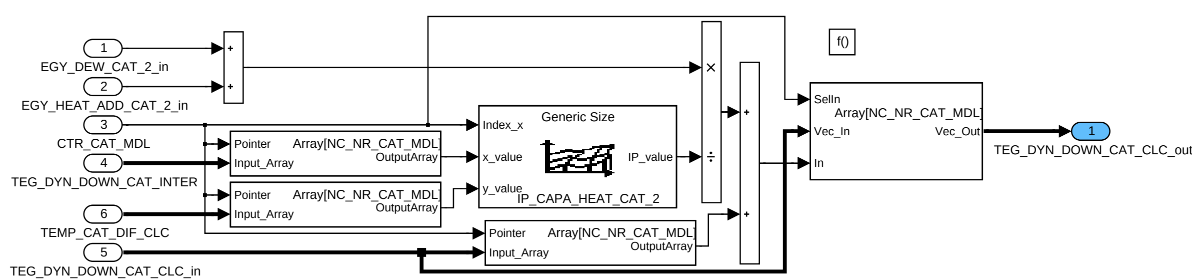


Figure 78 EXTD_MDLADcat0/ CLC__CAT_MDL/ DYN_MDL_CAT/ CLC__TEG_DYN_DOWN_CAT

7.45 Turbine exhaust gas temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_TUR[NC_NR_TUR_MDL]	O/V/S	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
turbine temperatures					
POW_HEAT_ADD_TUR[NC_NR_TUR_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat flow from the exhaust gas to the differnt turbines					
EGY_HEAT_ADD_TUR[NC_NR_TUR_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
sum of all heat energy added to the different turbines					
EGY_DEW_TUR[NC_NR_TUR_MDL]	V	80000000...7FFFFFFFH	-32.768...32.768	1.52588E-8	kJ
dew energy added to the different turbines					
POW_HEAT_LOSS_TUR[NC_NR_TUR_MDL]	V	8000...7FFFH	-32.768...32.767	0.001	kW
heat losses of the differnt turbines					
EGY_DEW_INT_TUR[NC_NR_TUBE_MDL]	V/S	0...FFFFH	0...327.675	0.005	kJ
dew energy integral of the turbines					
TEMP_TUR_STAT[NC_NR_TUR_MDL]	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
static turbine temperatures					
TEG_DYN_DOWN_TUR_FIL[NC_NR_TUR_MDL]	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
fitlered exhaust gas temperature down turbine					

Input data:

LV_ES	TAM	VS	T_ES
LV_T_ES_NOT_PLAUS	TCO	TCO_STOP	POW_CBK_TEG_TUR[NC_CBK_EX_NR]
N_TEG	MEF_CBK_TEG[NC_CBK_EX_NR]	MEF_KGH_CBK_TEG[NC_CBK_EX_NR]	T_SAMPLE_TEG
CTR_TUR_MDL	TEG_MDL_INP	TEG_MDL_CBK	TEG[NC_NR_TEG_MDL]
TEG_STAT_MDL[NC_NR_TEG_MDL]	HUM_EX_GAS_MDL[NC_NR_TEG_MDL]	NC_CBK_EX_NR	NC_NR_TEG_MDL
NC_NR_TUR_MDL	C_EGY_DEW	C_CAPA_HEAT_TEG	IP_HUM_EX_GAS_MAX

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_POW_HEAT_LOSS_TUR[NC_NR_TUR_MDL]	1	0...FFFFH	0...29.8020412	4.5475E-4	nW/K^4
factor for the calculation of heat losses of the turbine due to radiation					
LC_TEG_DYN_DOWN_TUR_FIL_SEL[NC_NR_TUR_MDL]	1	0...1H	0...1	1	-
switch to select the filtered value as output					
LC_TUR_MDL_ENA	1	0...1H	0...1	1	-
switch to activate turbine model					
IP_CAPA_HEAT_TUR[NC_NR_TUR_MDL]	8	0...FFFFH	0...65.535	0.001	kJ/K
LDPM_TEMP_1_EXTD	8	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
heat capacity of the differnt turbines					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CRLC_TEG_DYN_DOWN_TUR[NC_NR_TUR_MDL]	10	0...FFFFH	0...0.99998474	1.52588E-5	-
LDPM_MEF_KGH_CBK_TEG_1_EXTD	10	0...FFFFH	0...4.09594E+3	0.0625	kg/h
filtering constant for the exhaust gas temperature down turbine					
IP_FAC_INI_TUR	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_T_ES_1_EXTD	8	0...FFFFH	0...6.5535E+4	1	min
factor for turbine cooling down dependent on engine off duration time					
IP_FAC_INI_TUR_STND	8	0...FFH	0...0.99609375	0.00390625	-
LDPM_RATIO_COOL_DOWN_1_EXTD	8	0...FFH	0...0.99609375	0.00390625	-
factor for turbine cooling down in case of non plausible engine off duration time					
IP_FAC_POW_HEAT_LOSS_TUR[NC_NR_TUR_MDL]	8	0...FFFFH	0...1.02398437	1.5625E-5	kW/K
LDPM_VS_1_EXTD	8	0...FFH	0...255	1	km/h
calculation factor for the calculation of the heat flow to ambient of the differnt turbines					
IP_FAC_POW_HEAT_ADD_TUR[NC_NR_TUR_MDL]	10x12	0...FFFFH	0...0.99998474	1.52588E-5	-
LDPM_N_TEG_1_EXTD	10	0...1FE0H	0...8.16E+3	1	rpm
LDPM_MEF_CBK_TEG_1_EXTD	12	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
calculation factor for the calculation of the heat flow from the exhaust gas to the turbine					

Import actions:

ACTION_EXTD_Data2Manag(IN <Hum>, IN <LvDew>, IN <TegDyn>, IN <TegStat>)
writes back the calculated data back to the arrays of the manager (TEG_DYN_MDL, ...)

7.45.1 General information:

The "turbine exhaust gas temperature model" is part of the complete exhaust gas temperature model. This part models the behaviour of a turbine in the exhaust line. In principle it is combination of a very easy turbine model and a tube model. The turbine model reduces the inlet exhaust gas temperature dependent on the turbine power POW_CBK_TEG_TUR.

Additionally there is the possibility to skip the turbine model to deal with non turbine engine variants in one software.

Initialisation of SRAM values at checksum error:

TEMP_TUR[NC_NR_TUR_MDL] = 0°C (1112H)

EGY_DEW_INT_TUR[NC_NR_TUR_MDL] = 0kJ (0H)

Function Description

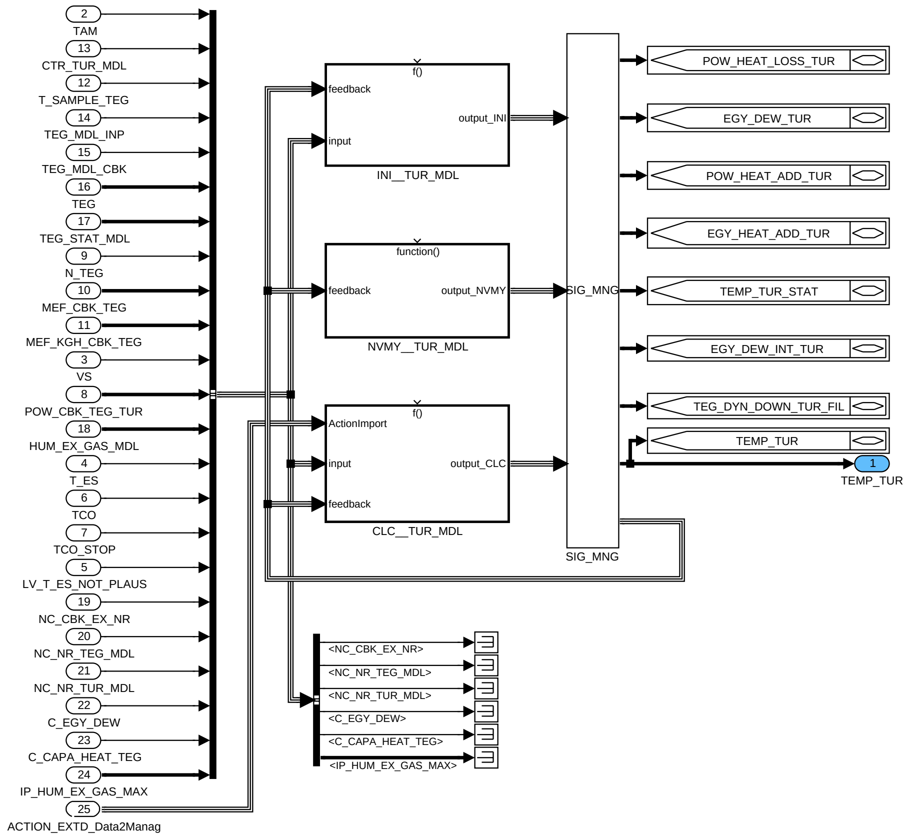


Figure 79 EXTD_MDLADturb0

7.45.1.1 INI__TUR_MDL

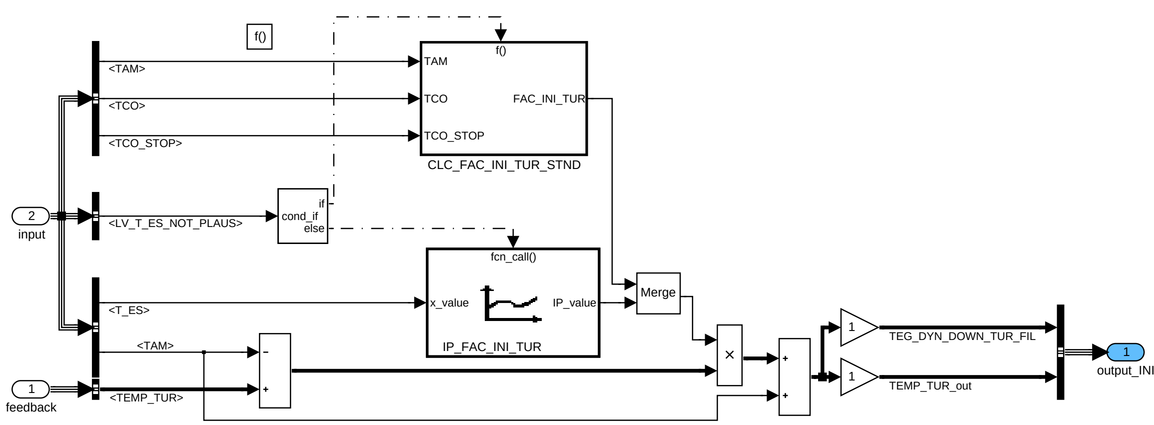


Figure 80 EXTD_MDLADturb0/ INI__TUR_MDL

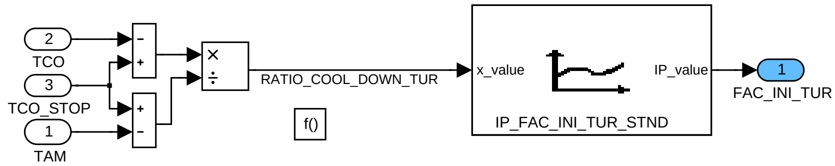


Figure 81 EXTD_MDLADturb0/ INI__TUR_MDL/ CLC_FAC_INI_TUR_STND

7.45.1.2 CLC__TUR_MDL

The calculation of the turbine model is split up in five part:
- selection of the input values out of the arrays: SEL_INP_TUR
- feed through in case of skipped turbine model: NOT_CLC
- calculation of the steady state model: STAT_MDL_TUR
- calculation of the dynamic model with dew point simulation: DYN_MDL_TUR
- the writing back of the result to the manager by the action

The switch LC_TUR_MDL_ENA has to be set to one to calculate the turbine model otherwise the calculation is skipped and the outlet temperature is equal to the inlet temperature. Due to the fact the most of the calculations are similar to the tube model most of the textual descriptions are skipped.

<u>SEL_INP_TUR</u>

The manager of the exhaust gas temperature model tells the turbine model which values out of the arrays has to be used for the calculation. Thereby TEG_MDL_CBK is the index for the exhaust bank and TEG_MDL_INP determines the input index.

<u>NOT_CLC</u>

Like described above the outlet temperature and the outlet humidity is set to there inlet values if the turbine model is skipped. Additionally LV_DEW_THD_TUR is set to one.

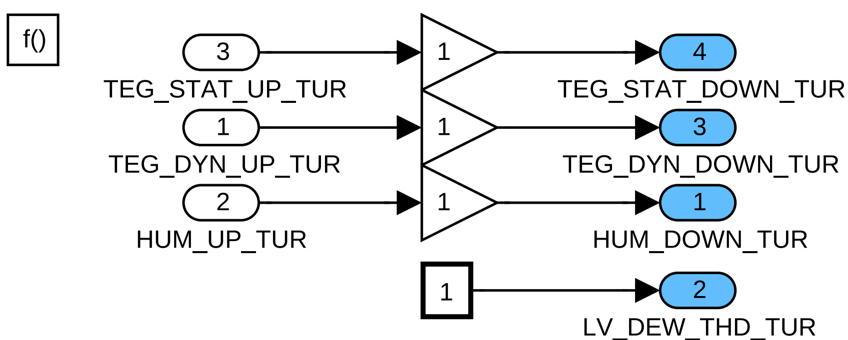


Figure 82 EXTD_MDLADturb0/ CLC__TUR_MDL/ NOT_CLC

STAT_MDL_TUR

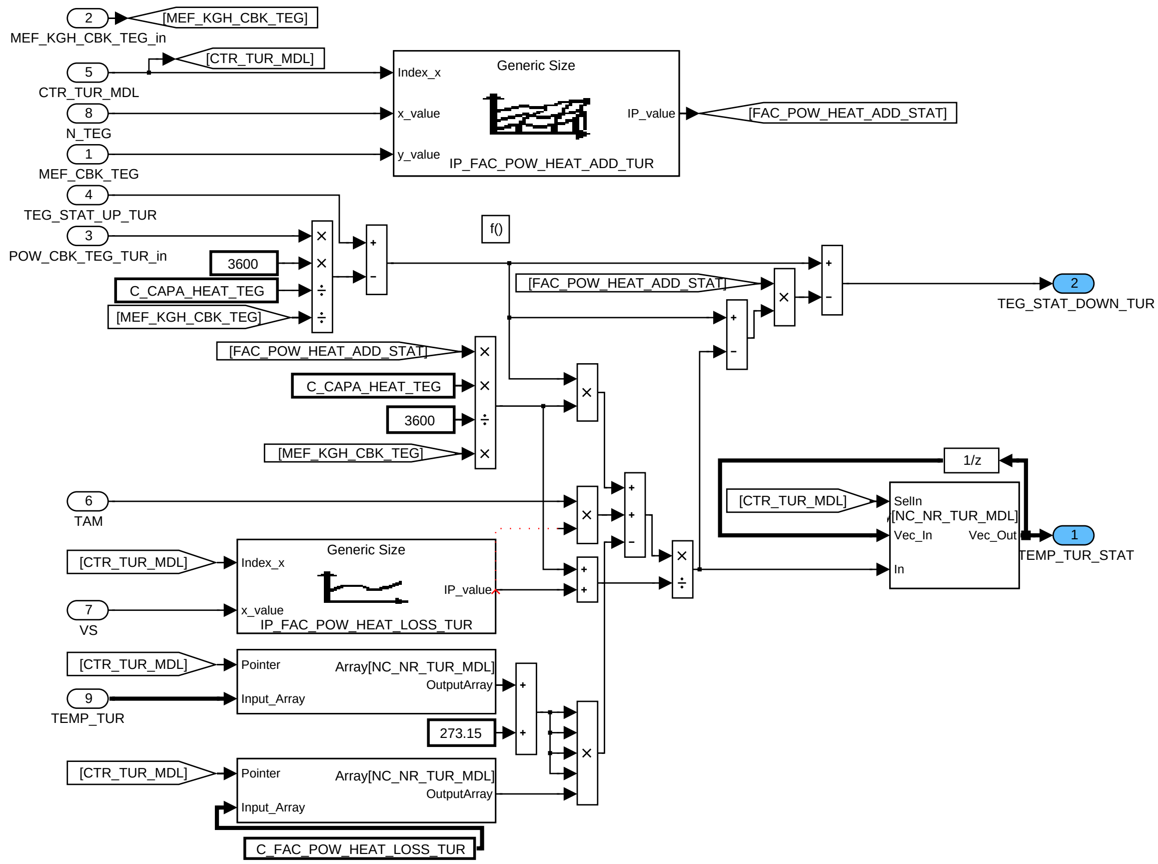


Figure 83 EXTD_MDLADturb0/ CLC__TUR_MDL/ STAT_MDL_TUR

DYN_MDL_TUR

The dynamic part of the turbine model is again split up in seven sub-functions:
- calculation of the temperature decrease due to the turbine: CLC__TEG_DYN_POW_TUR
- calculation of the heat flows out of the calibration data: CLC__POW_HEAT_FLOW_TUBE
- calculation of the heat energy added to the tube during sample time of the model: CLC__EGY_HEAT_ADD_TUBE
- calculation of the exhaust gas temperature downstream the tube: CLC__TEG_DYN_DOWN_TUBE
- filtering of the calculated exhaust gas temperature downstream tube: CLC__TEG_DYN_DOWN_TUBE_FIL
- dew point calculations for the tube: CLC__DEW_TUBE
- determination of the temperature in/decrease of the tube wall temperature: CLC__TEMP_TUBE

CLC__TEG_DYN_POW_TUR

The turbine power POW_CBK_TEG_TUR is the calculated turbine power and has to be determined by the turbo charger functionality. The turbine inlet temperature is decreased dependent on this power and dependent on the mass exhaust gas flow. This decreased temperature TEG_DYN_POW_TUR is used as input for all other calculations which are similar to the calculations in the tube exhaust gas temperature model.

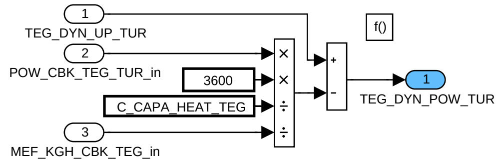


Figure 84 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__TEG_DYN_POW_TUR

CLC__POW_HEAT_FLOW_TUR

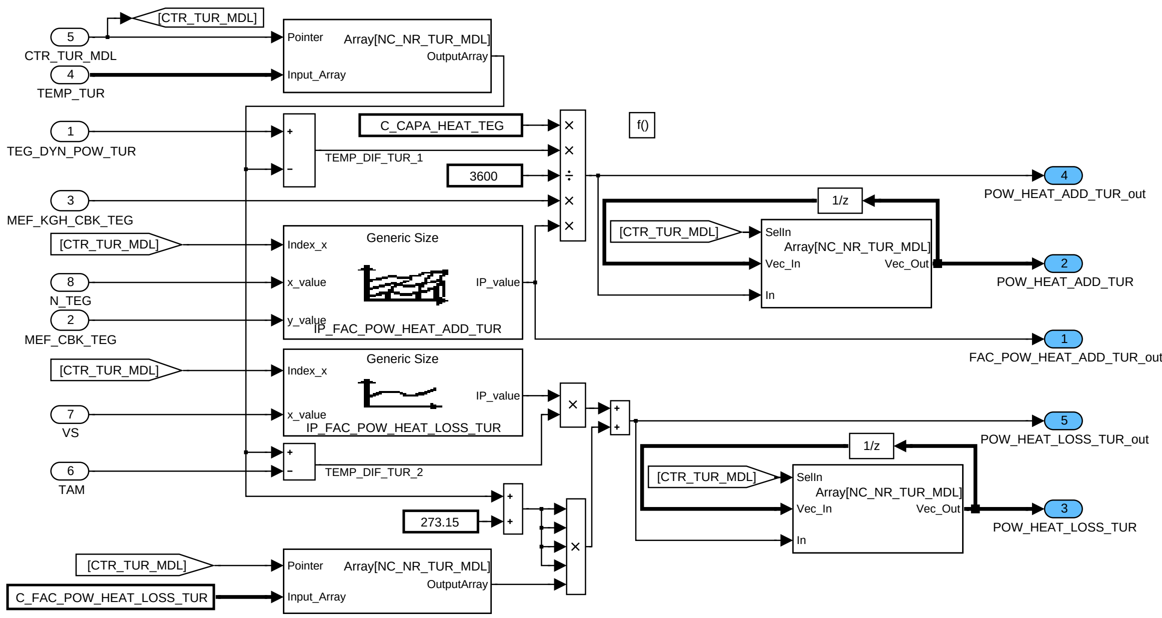


Figure 85 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__POW_HEAT_FLOW_TUR

CLC__EGY_HEAT_ADD_TUR

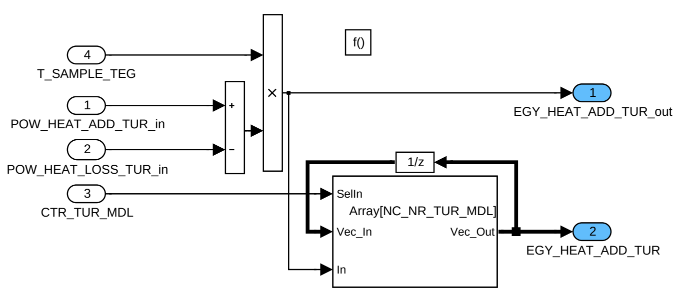


Figure 86 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__EGY_HEAT_ADD_TUR

CLC__TEG_DYN_DOWN_TUR

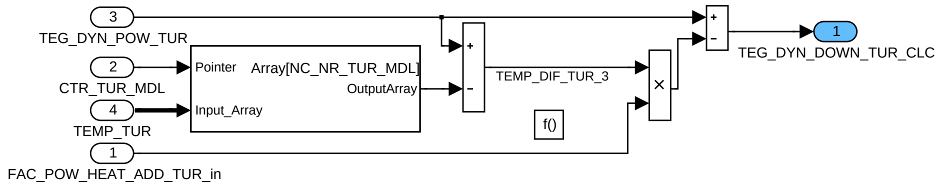


Figure 87 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__TEG_DYN_DOWN_TUR

CLC__DEW_TUR

The dew point calculations are similar to the calculations in the tube model. So no detailed function description is added for the function part.

CLC__TEMP_TUR

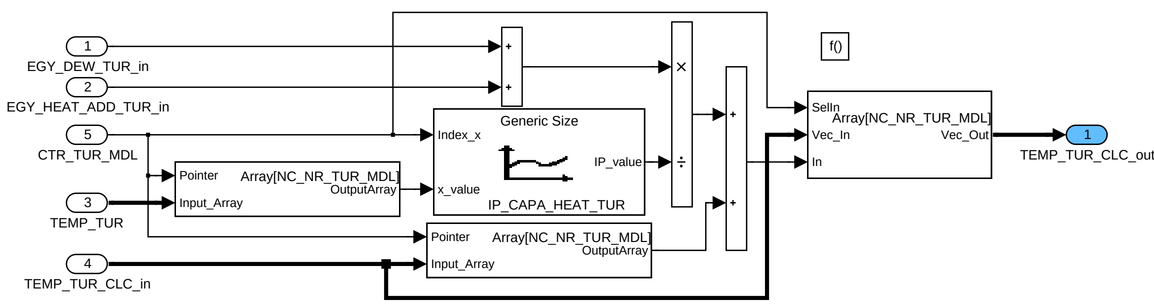


Figure 88 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__TEMP_TUR

CLC__TEG_DYN_DOWN_TUR_FIL

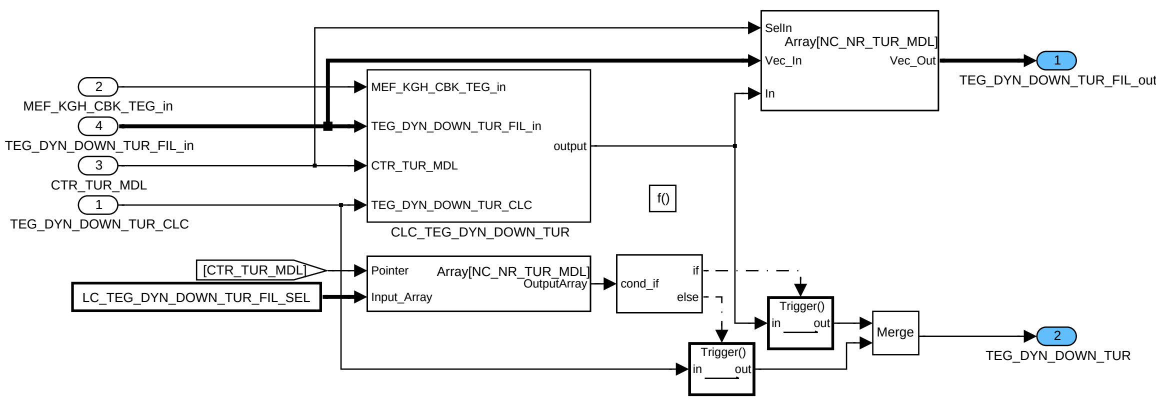


Figure 89 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__TEG_DYN_DOWN_TUR_FIL

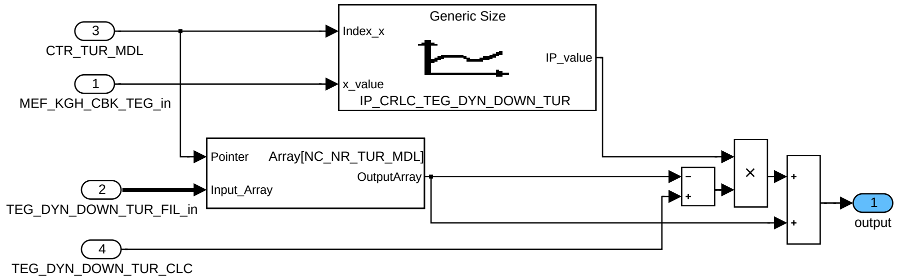


Figure 90 EXTD_MDLADturb0/ CLC__TUR_MDL/ DYN_MDL_TUR/ CLC__TEG_DYN_DOWN_TUR_FIL/ CLC_TEG_DYN_DOWN_TUR

7.46 Exhaust gas temperature model (Appl. Inc)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_TEG	O/V	0...1FE0H	0...8.16E+3	1	rpm
mean value of N for exhaust gas temperature model					
MAF_TEG	O/V	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
mean value of MAF for the exhaust gas temperature model					
MAF_KGH_CBK_TEG[NC_CBK_EX_NR]	O/V	0...FFFFH	0...4.09594E+3	0.0625	kg/h
mass air flow in kg per hour for the calculation of the exhaust gas temperature model					
MEF_CBK_TEG[NC_CBK_EX_NR]	O/V	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
mass exhaust gas flow in mg/stk					
MEF_KGH_CBK_TEG[NC_CBK_EX_NR]	O/V	0...FFFFH	0...4.09594E+3	0.0625	kg/h
mass exhaust gas flow in kg per hour for the calculation of the exhaust gas temperature model					
SAF_KGH_CBK_TEG[NC_CBK_EX_NR]	O/V	0...FFFFH	0...1.02398E+3	0.015625	kg/h
secundary air mass flow in kg per hour for the calculation of the exhaust gas temperature model					
IGA_AV_MV_CBK_TEG[NC_CBK_EX_NR]	O/V	0...FFH	-35.625...60	0.375	°CRK
mean value of IGA_AV_MV_CBK for exhaust gas temperature model					
LAMB_CBK_TEG_ENG[NC_CBK_EX_NR]	O/V	0...7FFFH	0...31.9990234	9.76563E-4	-
combustion lambda of the engine for the calculation of the exhaust gas temperature model					
LAMB_CBK_TEG_CAT[NC_CBK_EX_NR]	O/V	0...7FFFH	0...31.9990234	9.76563E-4	-
lambda of the exhaust gas at the catalyst for the calculation of the exhaust gas temperature model					
NR_CYL_SCC_CBK[NC_CBK_EX_NR]	O/V	0...50H	0...8	0.1	-
number of shutt off cylinders for the calculation of the exhaust gas temperature model					
TEG_ENG_OUT_ADD_CUS[NC_NR_ENG_OUT_MDL]	O/V	8000...7FFFH	-2.048E+3 ... 2.04794E+3	0.0625	K
customer specific additive corrections on the engine out exhaust gas temperature					
FAC_TEG_ENG_OUT_CUS[NC_NR_ENG_OUT_MDL]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
customer specific correction factor on the engine out exhaust gas temperature					
IGA_REF_TEG_ADD_CUS[NC_NR_ENG_OUT_MDL]	O/V	80...7FH	-48...47.625	0.375	°CRK
additive customer specific correction on the reference ignition angle for the exhaust gas temperature model					
IGA_BAS_TEG_ADD_CUS[NC_NR_ENG_OUT_MDL]	O/V	80...7FH	-48...47.625	0.375	°CRK
additive customer specific correction on the basic ignition angle for the exhaust gas temperature model					
FAC_TEG_ADD_IGA_CUS[NC_NR_ENG_OUT_MDL]	O/V	80...7FH	-1...0.9921875	0.0078125	-
additive customer specific correction on IP_FAC_TEG_ADD_IGA					
POW_EXO_CAT_ADD_CUS[NC_NR_CAT_MDL]	O/V	8000...7FFFH	-32.768...32.767	0.001	kW
customer specific additive correction of the catalyst exothermic					
FAC_POW_EXO_CAT_CUS[NC_NR_CAT_MDL]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
customer specific correction factor for the catalyst exothermic					
POW_CBK_TEG_TUR[NC_CBK_EX_NR]	O/V	0...FFFFH	0...65.535	0.001	kW
turbine power for the calculation of the exhaust gas temperature model					

Input data:

N	MAF	MAF_FG_CYL	IGA_AV_MV_CBK[NC_CBK_EX_NR]
LAMB_SP[NC_CBK_EX_NR]	SUM_INH_INJ	SUM_INH_IV_CBK[NC_CBK_EX_NR]	MFF_SP_MV
POW_CHA[NC_NR_TCHA]	NC_CBK_EX_NR	NC_CYL_NR	NC_MAF_FAC_CYL
NC_NR_ENG_OUT_MDL	NC_NR_CAT_MDL	NC_NR_TCHA	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CRLC_LAMB_CBK_TEG	8	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_MAF_FG_CYL_IP_CRLC_LAMB_TEG	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Filtering constant for the filter of LAMB_CBK_TEG_CAT					

7.46.1 General information:

This module is just a template which deals with the most project needs. It is used as interface to the exhaust gas temperature model. It creates the necessary values for the calculation of the exhaust gas temperature model and gives the project the possibility for customer specific corrections of the main input values like:
Filtering of the ignition angle, engine speed, mass air flow, ...
Using measured lambda instead of lambda set point
Different calculation of the mass exhaust gas flow for projects with an unsymmetrical exhaust manifold

Additionally additive and multiplicative corrections on the engine out temperature and the catalyst exothermic can be applied by the values:
TEG_ENG_OUT_ADD_CUS
FAC_TEG_ENG_OUT_CUS
IGA_REF_TEG_ADD_CUS
IGA_BAS_TEG_ADD_CUS
FAC_TEG_ADD_IGA_CUS
POW_EXO_CAT_ADD_CUS
FAC_POW_EXO_CAT_CUS

These values can be used to add corrections for example for variable valve timing, external exhaust gas recirculation, bi-fuel engines or any other corrections which are necessary for the project.

Application Condition

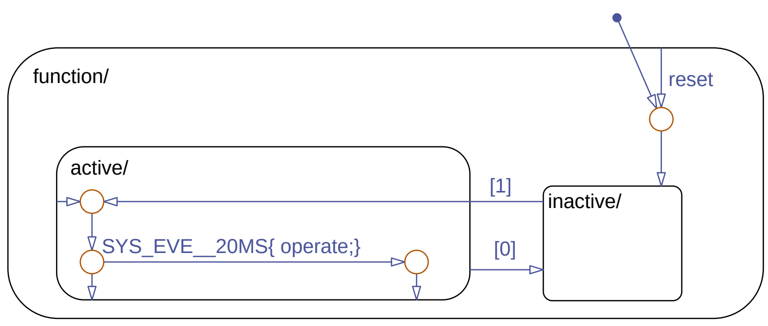


Figure 91 EXTD_MDLADai0/ APP_CDN/ Chart

Function Description

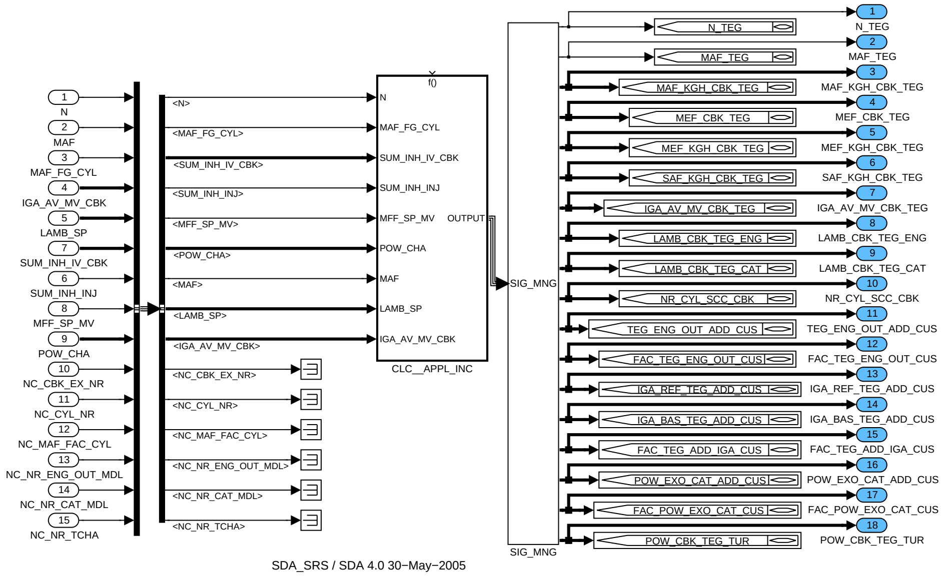


Figure 92 EXTD_MDLADai0

7.46.1.1 Calculation of the application incidences: CLC__APPL_INC

In this function part the initialisation of the non calculated corrections is done. Factors are initialized with one and additive corrections are initialized with zero.

Additionally the mapping of engine speed, mass air flow, ignition angle ... to the corresponding values for calculation of the exhaust gas temperature model is defined.

For the calculation of the mass exhaust gas flow the injected fuel mass, the mass air flow, the secondary air flow and the number of exhaust banks is used.

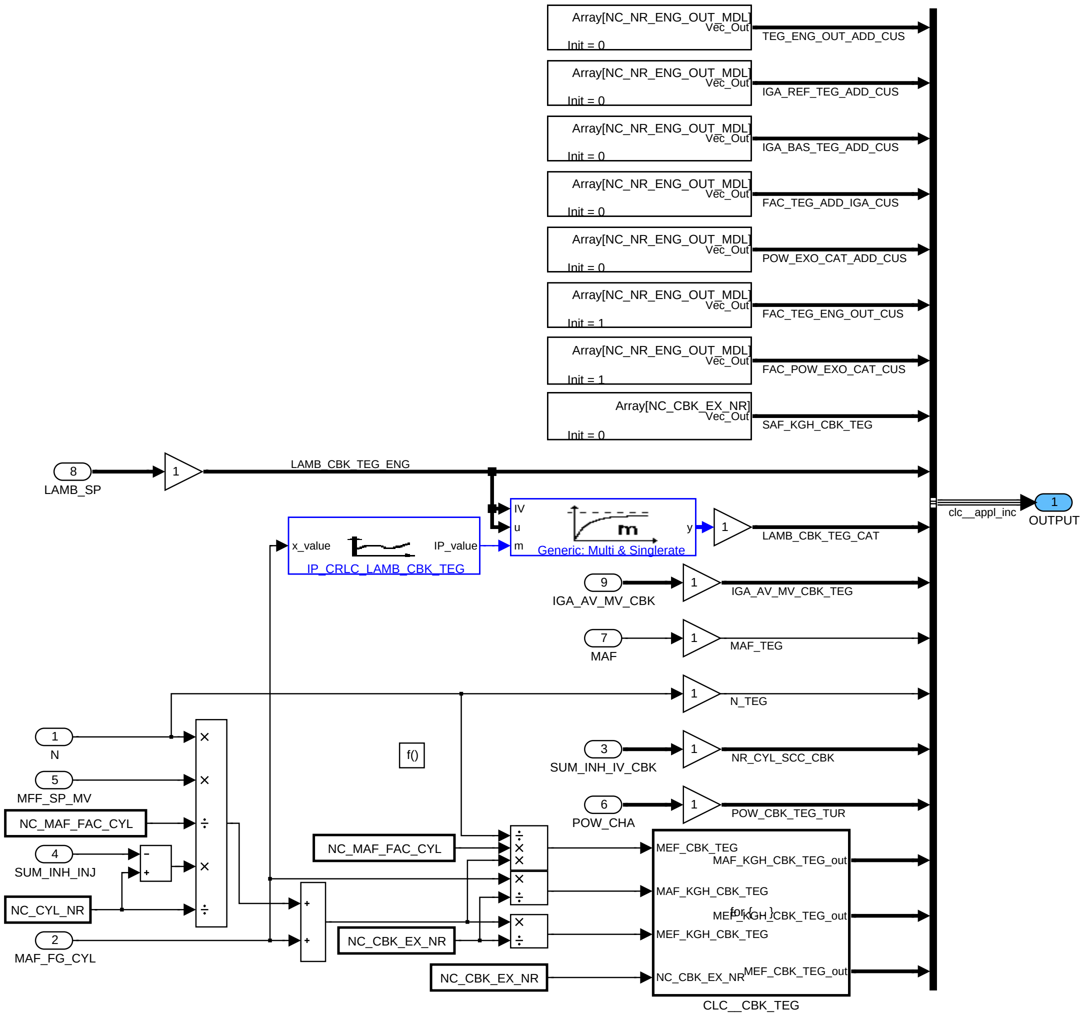


Figure 93 EXTD_MDLADai0/ CLC__APPL_INC

Setting of vectors for exhaust bank selective values: CLC__CBK_TEG

This function part describes that the single scalar values are stored in a vector with the same name.

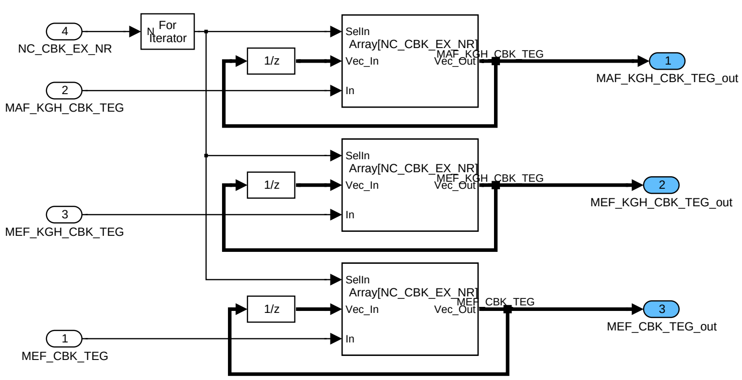


Figure 94 EXTD_MDLADai0/ CLC__APPL_INC/ CLC__CBK_TEG

7.46.2 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEG_DYN_LS_UP_MAX_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Maximum exhaust gas temperature at upsteam O2 sensor during current driving cycle					
TEG_DYN_LS_UP_MIN_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Minimum exhaust gas temperature at upsteam O2 sensor during current driving cycle					
TEG_DYN_UP_CAT_MAX_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Maximum exhaust gas temperature upstream catalyst during current driving cycle					
TEG_DYN_UP_CAT_MIN_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Minimum exhaust gas temperature upstream catalyst during current driving cycle					
TEMP_CAT_DYN_MDL_MAX_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Maximum catalyst temperature during current driving cycle					
TEMP_CAT_DYN_MDL_MIN_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Minimum catalyst temperature during current driving cycle					
TEG_DYN_LS_DOWN_MAX_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Maximum exhaust gas temperature at downstream O2 sensor during current driving cycle					
TEG_DYN_LS_DOWN_MIN_DC	O/V	0...7FFFH	-273.15...1774.7	0.0625	°C
Minimum exhaust gas temperature at downstream O2 sensor during current driving cycle					

Input data:

TEG_DYN_LS_UP[NC_CBK_EX_NR]	TEG_DYN_UP_CAT[NC_CBK_EX_NR]	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	TEG_DYN_LS_ DOWN [NC_CBK_EX_NR]
LV_ES	TCO		

Application conditions:

Initialisation: at LV_IGK 0->1 and reset:

TEG_DYN_LS_UP_MAX_DC = 0H
TEG_DYN_UP_CAT_MAX_DC = 0H
TEMP_CAT_DYN_MDL_MAX_DC = 0H
TEG_DYN_LS_DOWN_MAX_DC = 0H

TEG_DYN_LS_UP_MIN_DC = 7FFFH
TEG_DYN_UP_CAT_MIN_DC = 7FFFH
TEMP_CAT_DYN_MDL_MIN_DC = 7FFFH
TEG_DYN_LS_DOWN_MIN_DC = 7FFFH

Recurrence: 1s

Activation: LV_ES = 0

Deactivation: -

Formula section:

```
If      TEG_DYN_LS_UP > TEG_DYN_LS_UP_MAX_DC
Then    TEG_DYN_LS_UP_MAX_DC = TEG_DYN_LS_UP
Endif

If      TEG_DYN_UP_CAT > TEG_DYN_UP_CAT_MAX_DC
Then    TEG_DYN_UP_CAT_MAX_DC = TEG_DYN_UP_CAT
Endif

If      TEMP_CAT_DYN_MDL > TEMP_CAT_DYN_MDL_MAX_DC
Then    TEMP_CAT_DYN_MDL_MAX_DC = TEMP_CAT_DYN_MDL
Endif

If      TEG_DYN_LS_DOWN > TEG_DYN_LS_DOWN_MAX_DC
Then    TEG_DYN_LS_DOWN_MAX_DC = TEG_DYN_LS_DOWN
Endif

If      TCO > C_TCO_MIN_TEG_MIN_ACQ
Then If        TEG_DYN_LS_UP < TEG_DYN_LS_UP_MIN_DC
     Then      TEG_DYN_LS_UP_MIN_DC = TEG_DYN_LS_UP
     Endif
     If        TEG_DYN_UP_CAT < TEG_DYN_UP_CAT_MIN_DC
     Then      TEG_DYN_UP_CAT_MIN_DC = TEG_DYN_UP_CAT
     Endif
     If        TEMP_CAT_DYN_MDL < TEMP_CAT_DYN_MDL_MIN_DC
     Then      TEMP_CAT_DYN_MDL_MIN_DC = TEMP_CAT_DYN_MDL
     Endif
     If        TEG_DYN_LS_DOWN < TEG_DYN_LS_DOWN_MIN_DC
     Then      TEG_DYN_LS_DOWN_MIN_DC = TEG_DYN_LS_DOWN
     Endif
Endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_TEG_MIN_ACQ	-	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature for minimum TEG aquisition					

7.47 Lambda Control Correction (FAC_LAM_COR)

7.47.1 Calculation of FAC_LAM_COR

7.47.2 Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_LAM_COR[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Limited lambda controller output plus pre-control correction					
FAC_LAM_PCTL[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
Lambda control - pre-control factor					
FAC_LAM_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Lambda controller output					
FAC_LAM_OUT_MV_CAT_DIAG[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
Mean value of lambda controller output, used for calculation of O2-load					
FAC_LAM_OUT_MV_CAT_DIAG_SAVE[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
Stored mean value of lambda controller output at the beginning of disturbed O2 – load calculation					
MAF_SUM_FAC_LAM_OUT[NC_CBK_EX_NR]	V	80000000...7FFFFFFFH	-102400...102399.9999	4.7684E-5	kg/h*%
Accumulated lambda controller output weighted in MAF_CYL					
MAF_CYL_SUM[NC_CBK_EX_NR]	V	0...FFFFFFFFH	0...134217727.96875	0.03125	kg/h
Accumulated air mass flow					
FAC_P_POS_O2L[NC_CBK_EX_NR]	V	0...7FFFH	0...49.998474	1.5259E-3	%
P-jump for additional O2-load on the rich side					
FAC_P_NEG_O2L[NC_CBK_EX_NR]	V	0...7FFFH	0...49.998474	1.5259E-3	%
P-jump for additional O2-load on the lean side					
FAC_LAM_ABSV_AFR[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
Absolute maximum value of lambda controller output - maximum value of the rich side					
FAC_LAM_ABSV_AFL[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
Absolute maximum value of lambda controller output - minimum value of the lean side					
FAC_LAM_ABSV_STAT_MAX[NC_CBK_EX_NR]	V	0...FFFFH	0...49.999237	7.6294E-4	%
Stationary absolute maximum value of lambda controller output - mean value from rich and lean side					
FAC_LAM_LIM[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Limited value of lambda controller output					
FAC_LAM_MAX[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Maximum of lambda control injection time correction factor					
FAC_LAM_MIN[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Minimum of lambda control injection time correction factor					
FAC_LAM[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
FAC_LAM_OUT[i] value after a P jump					
FAC_LAM_MV[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Mean value of lambda controller output					
FAC_LAM_MV_MMV[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
FAC_LAM_MV[i] average value					
FAC_LAM_MV_MMV_CP[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Mean value of lambda controller output, used by canister purge					
FAC_LAM_MV_MMV_LDC[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Mean value of the controller output, used by the function "Mean value calculation for limited dynamics"					
FAC_LAM_MV_MMV_LDC_DIAG[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
Mean value of the controller output, used by limited dynamics for cat efficiency diagnosis					
T_DLY_NEG_LAM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1279.980469	1.9531E-2	ms

Total LAM -P-jump delay time from rich to lean transition					
T_DLY_POS_LAM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1279.980469	1.9531E-2	ms
Total LAM -P-jump delay time from lean to rich transition					
T_DLY_BAS_POS_LAM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1279.980469	1.9531E-2	ms
Positive basic LAM -P-jump delay time					
T_DLY_BAS_NEG_LAM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...1279.980469	1.9531E-2	ms
Negative basic LAM -P-jump delay time					
O2L_AFR_LAM[NC_CBK_EX_NR]	V	8000...7FFFH	-1.308444...1.308405	3.9931E-5	g
Actual value of O2-load on the rich side					
O2L_AFR_LAM_NOM[NC_CBK_EX_NR]	V	8000...7FFFH	-1.308444...1.308405	3.9931E-5	g
Nominal value of O2-load up to realisation of additional O2-load on the rich side					
O2L_AFL_LAM[NC_CBK_EX_NR]	V/O	8000...7FFFH	-1.308444...1.308405	3.9931E-5	g
Actual value of O2-load on the lean side					
O2L_LAM[NC_CBK_EX_NR]	V/O	0...7FFFH	0...1.308405	3.9931E-5	g
Updated O2-load value					
O2L_CAT_DIAG_MISS_AFR[NC_CBK_EX_NR]	V	0...7FFFH	0...1.308405	3.9931E-5	g
Value of the missing O2-load on the rich side					
O2L_CAT_DIAG_MISS_AFL[NC_CBK_EX_NR]	V	0...7FFFH	0...1.308405	3.9931E-5	g
Value of the missing O2-load on the lean side					
O2L_AFR_DLY[NC_CBK_EX_NR]	V	8000...7FFFH	-1.308444...1.308405	3.9931E-5	g
Additional O2-load value on the rich side					
O2L_AFL_DLY[NC_CBK_EX_NR]	V	8000...7FFFH	-1.308444...1.308405	3.9931E-5	g
Additional O2-load value on the lean side					
T_AFR_CYC_LAM[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
Rich half-period control cycle - used for calculation of O2-load					
T_AFL_CYC_LAM[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
Lean half-period control cycle - used for calculation of O2-load					
T_SUM_AFL_AFR_CYC_LAM[NC_CBK_EX_NR]	V/O	0...FFFFH	0...655.35	0.01	s
Control cycle - used for calculation of O2-load					
CTR_T_DLY_LAM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
counter for delay of P-jump (= 0: calculation of I-share, = 1: calculation of P-jump, > 1: delay running)					
CTR_T_DLY_SAVE_LAM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
initial value of counter saved at transition state (lean↔rich)					
CTR_RAF_CHG[NC_CBK_EX_NR]	V/O	0...FFH	0...255	1	-
Counter of air fuel ratio changes					
CTR_RAF_CHG_SAVE[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
The initial value of P-jump-counter, for the calculation of number of P-jump after PL-PU change to IS					
CTR_FAC_P_POS[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
Counter of p- jumps after disturbed O2-load calculation release					
MAF_INT_O2L_TRO[NC_CBK_EX_NR]	V	0...FFFFH	0...1820.416667	0.027778	g
integral of air mass flow during the disturbed O2-load calculation release					
LV_FAC_LAM_ADJ_LAM_AD_END[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Controller shift caused by lambda adaptation was carried out					
LV_FAC_LAM_SHIFT_CP_END[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-

Controller shift caused by canister purge was carried out					
LV_FAC_LAM_DIAGCP_END[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Controller shift caused by EVAP diagnosis was carried out					
LV_LAM_STOP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation signal for the lambda controller stop-mode					
LV_FAC_LAM_LIM_MIN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Lower limit of lambda controller output reached					
LV_FAC_LAM_LIM_MAX[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Upper limit of lambda controller output reached					
LV_FAC_LAM_LIM_INI[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Initialisation flag for limitation constants					
LV_PL_PU_TRAN_IS_LAM[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
transition flag PL-PU to IS					
LV_T_DLY_CHG_LAM[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Indication flag - T_LAM_DLY_POS or T_LAM_DLY_NEG changed during the P-jump delay					
LV_T_DLY_POS_LIM_LAM[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
limitation flag indicating that the T_LAM_DLY_POS has been limited to zero					
LV_O2L_LAM_RLS[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag indicating that the calculation of O2-load is released - used by catalyst diagnosis					
LV_O2L_LAM_RLS_AFR_TRO[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the calculation of O2-load on the rich side is disturbed					
LV_O2L_LAM_RLS_AFL_TRO[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the calculation of O2-load on the lean side is disturbed					
LV_MAF_INT_O2L_TRO_THD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag indicating that the threshold of MAF - integral during the disturbed O2 - load calculation exceeded					
LV_O2L_LAM_RLS_SET_WAIT[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the release of O2-load calculation is in wait status					
LV_O2L_LAM_RLS_RST_WAIT[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the reset of O2-load calculation release is in wait status					
LV_FAC_P_POS_O2L[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the positive P-jump for additional O2-load was carried out					
LV_FAC_P_NEG_O2L[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the negative P-jump for additional O2-load was carried out					
LV_O2L_SP_UPD_WAIT[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the wait status set to update the missing O2-load					
LV_O2L_MISS_AFR_UPD[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the missing O2-load on the rich side was updated					
LV_O2L_MISS_AFL_UPD[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag indicating that the missing O2-load on the lean side was updated					
LV_FAC_P_POS_LAM[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag indicates the beginning of the rich- (=1) or the lean- (=0) half-cycle in lambda controller output					
MFF_ADD_CP	V/O	0...FFFFH	0...1389	0.02119	mg/stk
Canister Purge Fuel Mass correction corresponding to Lambda Factor Shift					
MFF_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	- 694.511... 694.489	0.021194	mg/stk
Lambda Adaptation Fuel Mass set point offset corresponding to Lambda Factor Shift					
FAC_LAM_AD_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.9985	0.0015258	%
Lambda Adaptation Fuel Mass set point factor corresponding to Lambda Factor Shift					

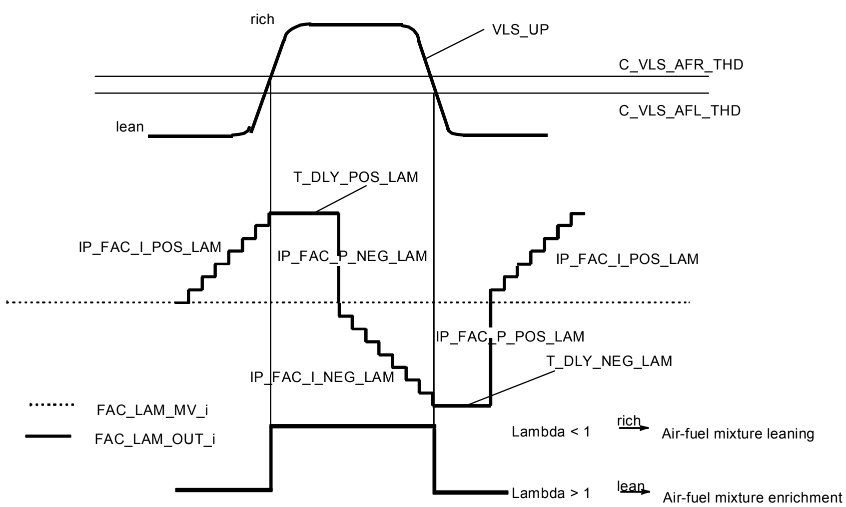
Input data:

LV_ES	LV_ST_END	LV_IS	LV_AT
LV_PL	LV_PU	LV_TI_1_HOM_MIN	LV_AFL[NC_CBK_EX_NR]
LV_FAC_LAM_DIAGCP	FAC_LAM_SHIFT_CP	NC_CBK_EX_NR	N_32
MAF	MAF_CYL	O2L_SP_CAT_DIAG[NC_CBK_EX_NR]	LV_FAC_LAM_SHIFT_CP

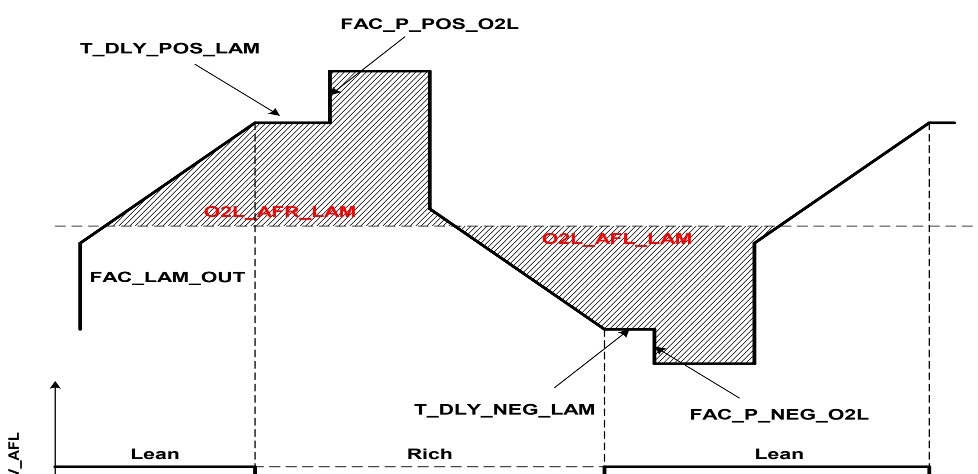
VLS_DIF_LAM_ADJ[NC_CBK_EX_NR]	LV_FAC_LAM_ADJ_LAM_AD[NC_CBK_EX_NR]
LV_LS_UP_READY[NC_CBK_EX_NR]	LV_LAM_LSCL[NC_CBK_EX_NR]
T_DLY_LAM_ADJ[NC_CBK_EX_NR]	CRLC_FAC_LAM_MV_MMV_CP
FAC_LAM_DIAGCP[NC_CBK_EX_NR]	FAC_LAM_ADJ_LAM_AD[NC_CBK_EX_NR]
FAC_LAM_AD[NC_CBK_EX_NR]	LAMB_SP_HOM[NC_CBK_EX_NR]
LV_O2L_LAM_REQ_CAT_DIAG[NC_CBK_EX_NR]	MFF_ADD_LAM_CP
MFF_LAM_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	FAC_LAM_AD_LAM_OUT[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

LV_O2L_LAM_RLS[i] = 0



LV_O2L_LAM_RLS[i] = 1



General Information:

The lambda control correction adjusts the fuel-air mixture to richness 1. The lambda controller uses a P/I strategy. It is adjusted to produce a leaner or a richer mixture according to the oxygen sensor information.

The P/I values are different in Idle (LV_IS) and in Part Load (LV_PL) or Trailing Throttle (LV_PU).

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank

Description:

Application conditions:

Activation: - LV_ST_END = 0

Recurrence: T_SAMPLE= 10 ms

MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]

FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]

MFF_ADD_CP = MFF_ADD_LAM_CP

Application conditions:

Activation: - LV_ST_END = 1

Recurrence: T_SAMPLE= 10 ms

Initialisation:

at transition LV_ES 0 → 1

FAC_LAM_COR[i] = 0

FAC_LAM_PCTL[i] = 0

FAC_LAM_OUT[i] = 0

FAC_LAM_LIM[i] = 0

T_DLY_POS_LAM[i] = 0

T_DLY_NEG_LAM[i] = 0

T_DLY_BAS_POS_LAM[i] = 0

T_DLY_BAS_NEG_LAM[i] = 0

CTR_RAF_CHG[i] = 0

CTR_RAF_CHG_SAVE[i] = 0

LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0

LV_FAC_LAM_LIM_MIN[i] = 0

LV_FAC_LAM_LIM_MAX[i] = 0

FAC_LAM[i] = 0

FAC_LAM_MV[i] = 0

FAC_LAM_MV_MMV[i] = 0

FAC_LAM_MV_MMV_CP[i] = 0

FAC_LAM_MV_MMV_LDC[i] = 0

FAC_LAM_MV_MMV_LDC_DIAG[i] = 0

LV_FAC_LAM_SHIFT_CP_END[i] = 0

LV_FAC_LAM_DIAGCP_END[i] = 0

FAC_LAM_MAX[i] = 0

FAC_LAM_MIN[i] = 0

LV_FAC_LAM_LIM_INI[i] = 0

LV_T_DLY_POS_LIM_LAM[i] = 0

CTR_T_DLY_SAVE_LAM[i] = 0

CTR_T_DLY_LAM[i] = 0

CTR_FAC_P_POS[i] = 0

FAC_LAM_OUT_MV_CAT_DIAG[i] = 0

FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i] = 0

MAF_SUM_FAC_LAM_OUT[i] = 0

MAF_CYL_SUM[i] = 0

MAF_INT_O2L_TRO[i] = 0

FAC_LAM_ABSV_AFR[i] = 0

FAC_LAM_ABSV_AFL[i] = 0

FAC_LAM_ABSV_STAT_MAX[i] = 0

FAC_P_POS_O2L[i] = 0

FAC_P_NEG_O2L[i] = 0

O2L_AFR_LAM[i] = 0

O2L_AFR_LAM_NOM[i] = 0

O2L_AFL_LAM[i] = 0

O2L_CAT_DIAG_MISS_AFR[i] = 0

O2L_CAT_DIAG_MISS_AFL[i] = 0

O2L_AFR_DLY[i] = 0

O2L_AFL_DLY[i] = 0

O2L_LAM[i] = 0

T_AFR_CYC_LAM[i] = 0

T_AFL_CYC_LAM[i] = 0

LV_O2L_LAM_RLS[i] = 0

LV_FAC_P_POS_O2L[i] = 0

LV_FAC_P_NEG_O2L[i] = 0

LV_FAC_P_POS_LAM[i] = 0

LV_O2L_SP_UPD_WAIT[i] = 0

LV_O2L_MISS_AFR_UPD[i] = 0

LV_O2L_MISS_AFL_UPD[i] = 0

LV_O2L_LAM_RLS_SET_WAIT[i] = 0

LV_O2L_LAM_RLS_RST_WAIT[i] = 0

LV_O2L_LAM_RLS_AFR_TRO[i] = 0

LV_O2L_LAM_RLS_AFL_TRO[i] = 0

LV_PL_PU_TRAN_IS_LAM = 0

LV_MAF_INT_O2L_TRO_THD[i] = 0

MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]

FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]

MFF_ADD_CP = MFF_ADD_LAM_CP

at reset:

The value of LV_LAM_STOP shall be set to zero. This variable is used as input in lambda adaptation and is not relevant to the functionality of binary lambda control.

LV_LAM_STOP[i] = 0

Formula section:

7.47.3 Calculation of the pre-control value (FAC_LAM_PCTL):

FAC_LAM_PCTL[i] = (1 / LAMB_SP_HOM[i] - 1) * 100%

7.47.4 Controller settings and shifts

The lambda controller takes a defined value under specific operating conditions.

if LV_AT = 1
then
xT = “AT”
else
xT = “MT”
endif

7.47.5 Controller:

if LV_LAM_LSCL[i] = 0 (A/F control loop opened)

then

FAC_LAM_OUT[i] = 0
FAC_LAM_LIM[i] = 0
T_DLY_POS_LAM[i] = 0
T_DLY_NEG_LAM[i] = 0
T_DLY_BAS_POS_LAM[i] = 0
T_DLY_BAS_NEG_LAM[i] = 0
LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0
LV_FAC_LAM_LIM_MIN[i] = 0
LV_FAC_LAM_LIM_MAX[i] = 0
FAC_LAM[i] = 0
FAC_LAM_MV[i] = 0
FAC_LAM_MV_MMV[i] = 0
FAC_LAM_MV_MMV_CP[i] = 0
FAC_LAM_MV_MMV_LDC[i] = 0
FAC_LAM_MV_MMV_LDC_DIAG[i] = 0
LV_FAC_LAM_SHIFT_CP_END[i] = 0
LV_FAC_LAM_DIAGCP_END[i] = 0
LV_FAC_LAM_LIM_INI[i] = 0
LV_T_DLY_POS_LIM_LAM[i] = 0
CTR_T_DLY_SAVE_LAM[i] = 0
CTR_T_DLY_LAM[i] = 0
CTR_FAC_P_POS[i] = 0
FAC_LAM_OUT_MV_CAT_DIAG[i] = 0
FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i] = 0
MAF_SUM_FAC_LAM_OUT[i] = 0
MAF_CYL_SUM[i] = 0
MAF_INT_O2L_TRO[i] = 0
FAC_LAM_ABSV_AFR[i] = 0
FAC_LAM_ABSV_AFL[i] = 0
FAC_LAM_ABSV_STAT_MAX[i] = 0
FAC_P_POS_O2L[i] = 0
FAC_P_NEG_O2L[i] = 0
O2L_AFR_LAM[i] = 0

O2L_AFR_LAM_NOM[i] = 0

O2L_LAM[i] = 0

O2L_AFL_LAM[i] = 0

O2L_CAT_DIAG_MISS_AFR[i] = 0

O2L_CAT_DIAG_MISS_AFL[i] = 0

O2L_AFR_DLY[i] = 0

O2L_AFL_DLY[i] = 0

T_AFR_CYC_LAM[i] = 0

T_AFL_CYC_LAM[i] = 0

LV_O2L_LAM_RLS[i] = 0

LV_FAC_P_POS_O2L[i] = 0

LV_FAC_P_NEG_O2L[i] = 0

LV_FAC_P_POS_LAM[i] = 0

LV_O2L_SP_UPD_WAIT[i] = 0

LV_O2L_MISS_AFR_UPD[i] = 0

LV_O2L_MISS_AFL_UPD[i] = 0

LV_O2L_LAM_RLS_SET_WAIT[i] = 0

LV_O2L_LAM_RLS_RST_WAIT[i] = 0

LV_O2L_LAM_RLS_AFR_TRO[i] = 0

LV_O2L_LAM_RLS_AFL_TRO[i] = 0

LV_PL_PU_TRAN_IS_LAM = 0

LV_MAF_INT_O2L_TRO_THD[i] = 0

MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]

FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]

MFF_ADD_CP = MFF_ADD_LAM_CP

else (A/F control loop closed)

calculate "lambda controller limits (7.47.5.1)"

if LV_TI_1_HOM_MIN = 0
then

if LV_IS = 1 **and** LV_PL_PU_TRAN_IS_LAM[i] = 0

then

calculate "controller setting at idle (7.47.5.2)"

else

if LV_PL = 1 **or** LV_PU = 1 **or**

(LV_PL_PU_TRAN_IS_LAM[i] =1 **and** LV_IS =1)

then

calculate "controller setting at part load or

trailing throttle (7.47.5.3)"

endif

endif

calculate "FAC_LAM_OUT_MV_CAT_DIAG and O_2 - load (7.47.5.4)"

calculate "controller shift by canister purge function or

EVAP diagnosis (7.47.5.5)"

calculate "controller shift by lambda adaptation (7.47.5.6)"

calculate "controller limitation - calculation of

FAC_LAM_LIM (7.47.5.7)"

calculate "P-jump counter, FAC_LAM and FAC_LAM_MV (7.47.5.8)"

calculate "FAC_LAM_MV_MMV_CP, FAC_LAM_MV_MMV,

FAC_LAM_MV_MMV_LDC and FAC_LAM_MV_MMV_LDC_DIAG (7.47.5.9)"

else

$FAC\_LAM\_OUT[i]_k = FAC\_LAM\_OUT[i]_{k-1}$

the controller shall be stopped if the flag for injection time limitation is set to 1 (LV_TI_1_HOM_MIN = 1). That means the value of FAC_LAM_OUT[i] shall be frozen to its last value, and all calculations shall be stopped. This is valid as long as the condition LV_TI_1_HOM_MIN is set to 1.

endif

calculate "update of CTR_T_DLY_LAM (7.47.5.11)"

endif

7.47.5.1 Lambda controller limits

The limits for the lambda controller are C_FAC_LAM_MAX and C_FAC_LAM_MIN.
In case of closed A/F-loop before operability is detected (LV_LAM_LSCL = 1 and LV_LS_UP_READY = 0) the limits C_FAC_LAM_INI_MIN and C_FAC_LAM_INI_MAX are valid instead of C_FAC_LAM_MIN and C_FAC_LAM_MAX. These limits are valid until a transition from lean to rich is detected.
If the lambda adaptation value FAC_LAM_AD ≠ 0, these limits shall be changed.

```
If      LV_LS_UP_READY[i] = 0
then
        LV_FAC_LAM_LIM_INI[i] = 1
else
        LV_FAC_LAM_LIM_INI[i] = 0
endif
if      (LV_FAC_LAM_LIM_INI[i] = 1 and LV_AFL[i] 1 → 0)
then
        LV_FAC_LAM_LIM_INI[i] = 0
endif
if       FAC_LAM_AD[i] = 0
then
        if       LV_FAC_LAM_LIM_INI[i] = 1
        then
                FAC_LAM_MAX[i] = C_FAC_LAM_INI_MAX
                FAC_LAM_MIN[i] = C_FAC_LAM_INI_MIN
        else
                FAC_LAM_MAX[i] = C_FAC_LAM_MAX
                FAC_LAM_MIN[i] = C_FAC_LAM_MIN
        endif
else
        if       FAC_LAM_AD[i] > 0
        then
                if      LV_FAC_LAM_LIM_INI[i] = 1
                then
                        FAC_LAM_MAX[i] = C_FAC_LAM_INI_MAX - FAC_LAM_AD_OUT[i]
                        FAC_LAM_MIN[i] = C_FAC_LAM_INI_MIN
                else
                        FAC_LAM_MAX[i] = C_FAC_LAM_MAX - FAC_LAM_AD_OUT[i]
                        FAC_LAM_MIN[i] = C_FAC_LAM_MIN
                endif
        else
                if       FAC_LAM_AD[i] < 0
                then
                        if       LV_FAC_LAM_LIM_INI[i] = 1
                        then
                                FAC_LAM_MAX[i] = C_FAC_LAM_INI_MAX
                                FAC_LAM_MIN[i] = C_FAC_LAM_INI_MIN -
FAC_LAM_AD_OUT[i]
                        else
                                FAC_LAM_MAX[i] = C_FAC_LAM_MAX
                                FAC_LAM_MIN[i] = C_FAC_LAM_MIN - FAC_LAM_AD_OUT[i]
                        endif
                endif
        endif
```

endif

7.47.5.2 Controller setting at idle:

7.47.5.2.1 Transition from lambda > 1 to lambda < 1:

```
if    LV_AFL[i] = 0
then
    if      C_T_DLY_POS_IS_LAM_xT + T_DLY_LAM_ADJ[i] < 0
    then
            T_DLY_POS_LAM[i] = 0
            LV_T_DLY_POS_LIM_LAM[i] = 1
    else
            LV_T_DLY_POS_LIM_LAM[i] = 0
            T_DLY_POS_LAM[i] = C_T_DLY_POS_IS_LAM_xT +
                                              T_DLY_LAM_ADJ[i]
    endif

    if ((| T_DLY_POS_LAM[i]k - T_DLY_POS_LAM[i]k-1 | <
              C_T_DLY_CHG_LAM_MIN) orCTR_T_DLY_LAM[i] <= 1)
    then
            LV_T_DLY_CHG_LAM[i] = 0
    else
            LV_T_DLY_CHG_LAM[i] = 1
    endif
endif

if    LV_AFL[i] 1 → 0
then
    CTR_T_DLY_LAM[i]k = T_DLY_POS_LAM[i] / T_SAMPLE + 1
    CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]k
else
     if     LV_T_DLY_CHG_LAM[i] = 1
     then
              CTR_T_DLY_LAM[i]k = MAX((T_DLY_POS_LAM[i] / T_SAMPLE
                     +1 – (CTR_T_DLY_SAVE_LAM[i] – CTR_T_DLY_LAM[i]k-1 )), 1)
              CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]k

     endif
 endif
```

7.47.5.2.2 Transition from lambda < 1 to lambda > 1:

```
if    LV_AFL[i] = 1
then
      if        LV_T_DLY_POS_LIM_LAM[i] = 1
      then
                T_DLY_NEG_LAM[i] = C_T_DLY_NEG_IS_LAM_xT -
                                  (C_T_DLY_POS_IS_LAM_xT + T_DLY_LAM_ADJ[i])
      else
                T_DLY_NEG_LAM[i] = C_T_DLY_NEG_IS_LAM_xT
      endif

      if ((| T_DLY_NEG_LAM[i]k - T_DLY_NEG_LAM[i]k-1 | < C_T_DLY_CHG_LAM_MIN) or
                CTR_T_DLY_LAM[i] <= 1)

      then
                LV_T_DLY_CHG_LAM[i] = 0
      else
                LV_T_DLY_CHG_LAM[i] = 1
      endif
endif

if    LV_AFL[i] 0 → 1
then
      CTR_T_DLY_LAM[i]k = T_DLY_NEG_LAM[i]/ T_SAMPLE + 1
      CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]k
else
       if       LV_T_DLY_CHG_LAM[i] = 1
       then
                  CTR_T_DLY_LAM[i]k = MAX((T_DLY_NEG_LAM[i] / T_SAMPLE
                          + 1 – (CTR_T_DLY_SAVE_LAM[i] – CTR_T_DLY_LAM[i]k-1)), 1)
                  CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]k

       endif
  endif
```

7.47.5.2.3 Calculation of controller output

The counter CTR_T_DLY_LAM is used as a trigger to determine the instant of the control action. For CTR_T_DLY_LAM > 1 the delay time is waited, for CTR_T_DLY_LAM = 1 the P-jump action will be taken and for CTR_T_DLY_LAM = 0 the integral action will be released.

```
If    LV_AFL[i] = 0
then
      if      CTR_T_DLY_LAM[i] = 1
      then
              FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                              IP_FAC_P_NEG_IS_LAM * C_FAC_GAIN_P_LAM
              LV_FAC_P_POS_LAM[i] = 0
      elseif  CTR_T_DLY_LAM[i] = 0
      then
              FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                              IP_FAC_I_NEG_IS_LAM * C_FAC_GAIN_I_LAM
      else
              FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
      endif
else
      if      CTR_T_DLY_LAM[i] = 1
      then
              FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                              IP_FAC_P_POS_IS_LAM * C_FAC_GAIN_P_LAM
              LV_FAC_P_POS_LAM[i] = 1
      elseif  CTR_T_DLY_LAM[i] = 0
      then
              FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                              IP_FAC_I_POS_IS_LAM * C_FAC_GAIN_I_LAM
      else
              FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
      endif
endif
```

7.47.5.3 Controller setting in part load or in trailing throttle:

7.47.5.3.1 Calculation of the transition flag PL-PU to IS

7.47.5.3.2 (LV_PL_PU_TRAN_IS_LAM)

After entry in idle (LV_IS) operation from part load (LV_PL) or trailing throttle (LV_PU), the P / I component of the lambda controller is switched to the P / I value for Idle (LV_IS) after an adjustable C_NR_RAF_CHG_IS_LAM number of P jumps.

```
if    (LV_PL_PU_TRAN_IS_LAM[i] = 1 and LV_IS = 1)
then
      if (CTR_RAF_CHG[i] – CTR_RAF_CHG_SAVE[i]) > C_NR_RAF_CHG_IS_LAM
      then   LV_PL_PU_TRAN_IS_LAM[i] =0
      endif
else
      LV_PL_PU_TRAN_IS_LAM[i] = 1
      CTR_RAF_CHG_SAVE[i] = CTR_RAF_CHG[i]
endif
```

7.47.5.3.3 Transition from lambda > 1 to lambda < 1:

```
if    LV_AFL[i] = 0
then
      if      (IP_T_DLY_POS_LAM_xT + T_DLY_LAM_ADJ[i]) < 0
      then
              T_DLY_POS_LAM[i] = 0
              LV_T_DLY_POS_LIM_LAM[i] = 1
              T_DLY_BAS_POS_LAM[i] = IP_T_DLY_POS_LAM_xT
      else
              LV_T_DLY_POS_LIM_LAM[i] = 0
              T_DLY_POS_LAM[i] = IP_T_DLY_POS_LAM_xT + T_DLY_LAM_ADJ[i]
              T_DLY_BAS_POS_LAM[i] = IP_T_DLY_POS_LAM_xT
      endif

      if ((│T_DLY_POS_LAM[i]k - T_DLY_POS_LAM[i]k-1│ <
                          C_T_DLY_CHG_LAM_MIN) or  CTR_T_DLY_LAM[i] <= 1)
      then
              LV_T_DLY_CHG_LAM[i] = 0
      else
              LV_T_DLY_CHG_LAM[i] = 1
      endif
endif

if    LV_AFL[i] 1 → 0
then
```

CTR_T_DLY_LAM[i]$_k$ = T_DLY_POS_LAM[i] / T_SAMPLE + 1
CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]$_k$
else
if LV_T_DLY_CHG_LAM[i] = 1
then
CTR_T_DLY_LAM[i]$_k$ = MAX((T_DLY_POS_LAM[i] / T_SAMPLE
+ 1 – (CTR_T_DLY_SAVE_LAM[i] – CTR_T_DLY_LAM[i]$_{k-1}$)), 1)
CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]$_k$
endif
endif

7.47.5.3.4 Transition from lambda < 1 to lambda > 1:

if LV_AFL[i] = 1
then
if LV_T_DLY_POS_LIM_LAM[i] = 1
then
T_DLY_NEG_LAM[i] = IP_T_DLY_NEG_LAM_xT -
(IP_T_DLY_POS_LAM_xT + T_DLY_LAM_ADJ[i])
T_DLY_BAS_NEG_LAM[i] = IP_T_DLY_NEG_LAM_xT
else
T_DLY_BAS_NEG_LAM[i] = IP_T_DLY_NEG_LAM_xT
T_DLY_NEG_LAM[i] = IP_T_DLY_NEG_LAM_xT
endif

if (($\left|$ T_DLY_NEG_LAM[i]$_k$ - T_DLY_NEG_LAM[i]$_{k-1}\right|$ <
C_T_DLY_CHG_LAM_MIN) **or** CTR_T_DLY_LAM[i] <= 1)
then
LV_T_DLY_CHG_LAM[i] = 0
else
LV_T_DLY_CHG_LAM[i] = 1
endif
endif

if LV_AFL[i] 0 $\rightarrow$ 1
then
CTR_T_DLY_LAM[i]$_k$ = T_DLY_NEG_LAM[i] / T_SAMPLE + 1
CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]$_k$
else
if LV_T_DLY_CHG_LAM[i] = 1
then
CTR_T_DLY_LAM[i]$_k$ = MAX((T_DLY_NEG_LAM[i] / T_SAMPLE
+ 1 – (CTR_T_DLY_SAVE_LAM[i] – CTR_T_DLY_LAM[i]$_{k-1}$)), 1)
CTR_T_DLY_SAVE_LAM[i] = CTR_T_DLY_LAM[i]$_k$
endif
endif

7.47.5.3.5 Calculation of controller output

The counter CTR_T_DLY_LAM is used as a trigger to determine the instant of the control action. For CTR_T_DLY_LAM > 1 the delay time is waited, for CTR_T_DLY_LAM = 1 the P-jump action will be taken and for CTR_T_DLY_LAM = 0 the integral action will be released.
In case of a request from catalyst diagnosis the O_2 - load shall be adjusted to the requested setpoint. In that case the deviation of lambda controller output from the mean value of controller output shall be monitored. Should the absolute value of deviation exceed the maximum threshold, so the request shall be rejected. The maximum threshold is defined as the product of a constant, C_FAC_MV_O2L_MAX, and the absolute steady state maximum value of lambda controller output FAC_LAM_ABSV_STAT_MAX[i], which covers the ageing influence of the lambda sensor.

<u>**Detection of O2-load request and calculation of missing O2-load**</u>

```
if    LV_O2L_LAM_REQ_CAT_DIAG[i] = 1
then
      if  (LV_O2L_LAM_RLS[i] = 0 and (LV_O2L_LAM_RLS_RST_WAIT[i] = 0 or
                 LV_O2L_LAM_RLS_AFR_TRO[i] = 1 or LV_O2L_LAM_RLS_AFL_TRO[i] = 1))
      then
              LV_O2L_LAM_RLS_SET_WAIT[i] = 1
      endif
endif
if    LV_O2L_LAM_REQ_CAT_DIAG[i] = 0
then
      if      (LV_O2L_LAM_RLS[i] = 1 and LV_O2L_LAM_RLS_SET_WAIT[i] = 0)
      then
              LV_O2L_LAM_RLS_RST_WAIT[i] = 1
      elseif  (LV_O2L_LAM_RLS[i] = 0 and LV_O2L_LAM_RLS_SET_WAIT[i] = 1)
      then
              LV_O2L_LAM_RLS_SET_WAIT[i] = 0
              CTR_FAC_P_POS[i] = 0
              LV_MAF_INT_O2L_TRO_THD[i] = 0
              LV_O2L_LAM_RLS_AFR_TRO[i] = 0
              LV_O2L_LAM_RLS_AFL_TRO[i] = 0
              MAF_INT_O2L_TRO[i] = 0
      endif
endif
if  (LV_O2L_LAM_RLS[i] = 1 and LV_MAF_INT_O2L_TRO_THD[i] = 1 and
                                              LV_FAC_P_POS_LAM[i]k-1 0 → 1)
then
      LV_MAF_INT_O2L_TRO_THD[i] = 0
      MAF_INT_O2L_TRO[i] = 0
endif
```

if (LV_O2L_LAM_RLS_SET_WAIT[i] = 1 **and**
LV_O2L_LAM_RLS_AFR_TRO[i] = 0 **and** LV_O2L_LAM_RLS_AFL_TRO[i] = 0)

then

if LV_FAC_P_POS_LAM[i]$_{k-1}$ = 0 $\rightarrow$1

then

LV_O2L_LAM_RLS[i] = 1

LV_O2L_LAM_RLS_SET_WAIT[i] = 0

endif

elseif (LV_O2L_LAM_RLS_SET_WAIT[i] = 1 **and**
LV_O2L_LAM_RLS_AFR_TRO[i] = 1)

then

if LV_FAC_P_POS_LAM[i]$_{k-1}$ = 1 $\rightarrow$ 0

then

increment CTR_FAC_P_POS[i]

if CTR_FAC_P_POS[i] $\geq$ C_CTR_AFR_TRO_MIN

then

if abs(FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i] -
FAC_LAM_OUT_MV_CAT_DIAG[i]**)** <
C_FAC_LAM_MV_CAT_DIAG_DIF_MAX

then

LV_O2L_LAM_RLS[i] = 1

LV_O2L_LAM_RLS_SET_WAIT[i] = 0

LV_O2L_LAM_RLS_AFR_TRO[i] = 0

CTR_FAC_P_POS[i] = 0

endif

endif

endif

elseif (LV_O2L_LAM_RLS_SET_WAIT[i] = 1 **and** LV_O2L_LAM_RLS_AFL_TRO[i] = 1)

then

if LV_FAC_P_POS_LAM[i]$_{k-1}$ = 0 $\rightarrow$ 1

then

increment CTR_FAC_P_POS[i]

if CTR_FAC_P_POS[i] $\geq$ C_CTR_AFL_TRO_MIN

then

if abs(FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i] -

FAC_LAM_OUT_MV_CAT_DIAG[i]**)** <
C_FAC_LAM_MV_CAT_DIAG_DIF_MAX

then

LV_O2L_LAM_RLS[i] = 1

LV_O2L_LAM_RLS_SET_WAIT[i] = 0

LV_O2L_LAM_RLS_AFL_TRO[i] = 0

CTR_FAC_P_POS[i] = 0

endif

endif

endif

endif

if LV_O2L_LAM_RLS_RST_WAIT[i] = 1

then

if $LV\_FAC\_P\_POS\_LAM[i]_{k-1} = 0 \rightarrow 1$
then
LV_O2L_LAM_RLS[i] = 0
LV_O2L_LAM_RLS_RST_WAIT[i] = 0
endif
endif

if (CTR_T_DLY_LAM[i] = 0 **and** LV_O2L_LAM_RLS[i] = 1)
then
if **abs(**FAC_LAM_OUT[i] - FAC_LAM_OUT_MV_CAT_DIAG[i]**)** >
C_FAC_MV_O2L_MAX * FAC_LAM_ABSV_STAT_MAX[i]
then
if $LV\_FAC\_P\_POS\_LAM[i]_{k-1} = 1$
then
LV_O2L_LAM_RLS_AFL_TRO[i] = 1
else
LV_O2L_LAM_RLS_AFR_TRO[i] = 1
endif
LV_O2L_LAM_RLS[i] = 0
LV_O2L_LAM_RLS_SET_WAIT[i] = 0
endif
endif

if (LV_AFL[i] $1 \rightarrow 0$ **and** $CTR\_T\_DLY\_LAM[i]_{k-1} > 0$ **and** $CTR\_T\_DLY\_LAM[i]_{k} > 0$
and LV_O2L_LAM_RLS[i] = 1)
then
LV_O2L_LAM_RLS_AFL_TRO[i] = 1
LV_O2L_LAM_RLS[i] = 0
LV_O2L_LAM_RLS_SET_WAIT[i] = 0
LV_O2L_LAM_RLS_RST_WAIT[i] = 0
CTR_T_DLY_LAM[i] = 0
If abs(FAC_LAM_OUT[i] - FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]**)** ≤
C_FAC_P_MV_MAX

```
    then
        FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
    else
        if FAC_LAM_OUT[i] > FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
        then
            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] - C_FAC_P_MV_MAX
        elseif FAC_LAM_OUT[i] < FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
        then
            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] + C_FAC_P_MV_MAX
        endif
    endif
endif

if  (LV_AFL[i] 1 → 0 and CTR_T_DLY_LAM[i]k-1 > 0 and CTR_T_DLY_LAM[i]k > 0
                                          and LV_O2L_LAM_RLS[i] = 0)
then
    CTR_FAC_P_POS[i] = 0
endif

if  (LV_AFL[i] 0 → 1 and CTR_T_DLY_LAM[i]k-1 > 0 and CTR_T_DLY_LAM[i]k > 0
                                          and LV_O2L_LAM_RLS[i] = 1)
then
    LV_O2L_LAM_RLS_AFR_TRO[i] = 1
    LV_O2L_LAM_RLS[i] = 0
    LV_O2L_LAM_RLS_SET_WAIT[i] = 0
    LV_O2L_LAM_RLS_RST_WAIT[i] = 0
    CTR_T_DLY_LAM[i] = 0
    If abs(FAC_LAM_OUT[i] - FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]) ≤
                                                    C_FAC_P_MV_MAX
    then
        FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
    else
        if FAC_LAM_OUT[i] > FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
        then
            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] - C_FAC_P_MV_MAX
        elseif FAC_LAM_OUT[i] < FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i]
        then
            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] + C_FAC_P_MV_MAX
        endif
    endif
endif

if  (LV_AFL[i] 0 → 1 and CTR_T_DLY_LAM[i]k-1 > 0 and CTR_T_DLY_LAM[i]k > 0
                                          and LV_O2L_LAM_RLS[i] = 0)
then
    CTR_FAC_P_POS[i] = 0
endif

if  ((LV_FAC_LAM_LIM_MIN[i] = 1 or LV_FAC_LAM_LIM_MAX[i] = 1 or
                  LV_FAC_LAM_DIAGCP = 1) and LV_O2L_LAM_RLS[i] = 1)
then
    if      LV_AFL[i] = 0
    then
```

```
                LV_O2L_LAM_RLS_AFR_TRO[i] = 1

        else
                LV_O2L_LAM_RLS_AFL_TRO[i] = 1
        endif
        LV_O2L_LAM_RLS[i] = 0
        LV_O2L_LAM_RLS_SET_WAIT[i] = 0
        LV_O2L_LAM_RLS_RST_WAIT[i] = 0
endif

if      LV_O2L_LAM_RLS[i] = 0
then
        LV_FAC_P_POS_O2L[i] = 0
        LV_FAC_P_NEG_O2L[i] = 0
endif


if      (LV_O2L_LAM_RLS_AFR_TRO[i] = 1 or LV_O2L_LAM_RLS_AFL_TRO[i] = 1)
then
        MAF_INT_O2L_TRO[i]k [g] = MAF_INT_O2L_TRO[i]k-1 [g]+
                                   MAF_CYL [kg/h]*T_SAMPLE [ms]*1/3600 [(g*h)/(kg*ms)]
endif

if      MAF_INT_O2L_TRO[i] ≥ C_MAF_INT_O2L_TRO_THD
then
        LV_MAF_INT_O2L_TRO_THD[i] = 1
endif


if      (LV_AFL[i] 1 → 0 and CTR_T_DLY_LAM[i]k-1 = 0 and CTR_T_DLY_LAM[i]k > 1)
then
        LV_O2L_SP_UPD_WAIT[i] = 1
endif

if      (LV_AFL[i] 0 → 1 and CTR_T_DLY_LAM[i]k-1 = 0 and CTR_T_DLY_LAM[i]k > 1)
then
        LV_O2L_SP_UPD_WAIT[i] = 1
endif

if      (LV_AFL[i] = 0 and LV_O2L_SP_UPD_WAIT[i] = 1)
then
        if      CTR_T_DLY_LAM[i] = 1
        then
                O2L_CAT_DIAG_MISS_AFR[i] = O2L_SP_CAT_DIAG[i] -
                                                        abs(O2L_AFR_LAM[i])
                O2L_AFR_LAM_NOM[i] = O2L_AFR_LAM[i]
                LV_O2L_SP_UPD_WAIT[i] = 0
                LV_O2L_MISS_AFR_UPD[i] = 1
        endif
endif

if      (LV_AFL[i] 1 → 0 and CTR_T_DLY_LAM[i]k-1 = 0 and CTR_T_DLY_LAM[i]k = 1)
then
```

```
        O2L_CAT_DIAG_MISS_AFR[i] = O2L_SP_CAT_DIAG[i] - abs(O2L_AFR_LAM[i])
        O2L_AFR_LAM_NOM[i] = O2L_AFR_LAM[i]
        LV_O2L_SP_UPD_WAIT[i] = 0
        LV_O2L_MISS_AFR_UPD[i] = 1
    endif

    if    (LV_AFL[i] = 1 and LV_O2L_SP_UPD_WAIT[i] = 1)
    then
        if      CTR_T_DLY_LAM[i] = 1
        then
                LV_O2L_SP_UPD_WAIT[i] = 0
                O2L_CAT_DIAG_MISS_AFL[i] = O2L_SP_CAT_DIAG[i] -
                                                    abs(O2L_AFR_LAM_NOM[i])
                LV_O2L_MISS_AFL_UPD[i] = 1
        endif
    endif

    if    (LV_AFL[i] 0 → 1 and CTR_T_DLY_LAM[i]k-1 = 0 and CTR_T_DLY_LAM[i]k = 1)
    then
        O2L_CAT_DIAG_MISS_AFL[i] = O2L_SP_CAT_DIAG[i] - abs(O2L_AFR_LAM_NOM[i])
        LV_O2L_SP_UPD_WAIT[i] = 0
        LV_O2L_MISS_AFL_UPD[i] = 1
    endif

    if    LV_FAC_P_POS_LAM[i]k-1 0 → 1
    then
        LV_O2L_MISS_AFL_UPD[i] = 0
    endif

    if    LV_FAC_P_POS_LAM[i]k-1 1 → 0
    then
        LV_O2L_MISS_AFR_UPD[i] = 0
    endif
```

7.47.5.3.5.1 Calculation of the controller output - case ' LV_O2L_LAM_RLS[i] = 0':

Calculation of the controller output on the rich side 'LV_AFL[i] = 0'

```
    if    (LV_O2L_LAM_RLS[i] = 0 and LV_AFL[i] = 0)
    then
        if      CTR_T_DLY_LAM[i] = 1
        then
                FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                        IP_FAC_P_NEG_LAM * C_FAC_GAIN_P_LAM
                LV_FAC_P_POS_LAM[i] = 0
        elseif  CTR_T_DLY_LAM[i] = 0
        then
                FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                        IP_FAC_I_NEG_LAM *C_FAC_GAIN_I_LAM
```

elseif CTR_T_DLY_LAM[i] > 1
then
FAC_LAM_OUT[i]$_k$ = FAC_LAM_OUT[i]$_{k-1}$

endif
endif

Calculation of the controller output on the lean side 'LV_AFL[i] = 1'

if (LV_O2L_LAM_RLS[i] = 0 **and** LV_AFL[i] = 1)
then
if CTR_T_DLY_LAM[i] = 1
then
FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
IP_FAC_P_POS_LAM * C_FAC_GAIN_P_LAM
LV_FAC_P_POS_LAM[i] = 1

elseif CTR_T_DLY_LAM[i] = 0
then
FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
IP_FAC_I_POS_LAM *C_FAC_GAIN_I_LAM

elseif CTR_T_DLY_LAM[i] > 1
then
FAC_LAM_OUT[i]$_k$ = FAC_LAM_OUT[i]$_{k-1}$

endif
endif

7.47.5.3.5.2 Calculation of the controller output - case ' LV_O2L_LAM_RLS[i] = 1':

Depending on the state of the logical calibration data LC_FAC_P_O2L_LAM the control algorithm changes. Should the logical calibration data be set to 1, an additional P - jump has to be carried out to reach the missing O2 - load faster. Should the calibration data be set to zero, the missing O2 - load would be reached by solely delayed P - jump in opposite direction.

7.47.5.3.5.2.1 Calculation of the controller output - case 'LC_FAC_P_O2L_LAM = 1':

Calculation of the controller output on the rich side 'LV_AFL[i] = 0'

if (LV_O2L_LAM_RLS[i] = 1 **and** LC_FAC_P_O2L_LAM = 1 **and** LV_AFL[i] = 0)
then
if abs(O2L_AFR_DLY[i]**)** < O2L_CAT_DIAG_MISS_AFR[i]
then
if CTR_T_DLY_LAM[i] = 1
then
if LV_FAC_P_POS_O2L[i] = 0
then
if (IP_FAC_P_POS_O2L + FAC_LAM_OUT[i]
- FAC_LAM_OUT_MV_CAT_DIAG[i]) ≤
IP_FAC_POS_O2L_MAX

```
                        then
                                FAC_P_POS_O2L[i] = IP_FAC_P_POS_O2L
                        else
                                FAC_P_POS_O2L[i] = IP_FAC_POS_O2L_MAX
                                   - FAC_LAM_OUT[i] + FAC_LAM_OUT_MV_CAT_DIAG[i]
                        endif
                        if      (FAC_LAM_OUT[i] + FAC_P_POS_O2L[i]
                                             - FAC_LAM_OUT_MV_CAT_DIAG[i]) ≤
                                                            C_FAC_ DIF _O2L_MIN
                        then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG[i]  +
                                             C_FAC_ DIF _O2L_MIN
                        else
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                                            FAC_P_POS_O2L[i]
                        endif
                        LV_FAC_P_POS_O2L[i] = 1

                elseif  LV_FAC_P_POS_O2L[i] = 1
                then
                        FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
                endif

        elseif  CTR_T_DLY_LAM[i] = 0
        then
                FAC_LAM_OUT[i] = FAC_LAM_OUT[i]
                                      - IP_FAC_I_NEG_LAM *C_FAC_GAIN_I_LAM
        elseif  CTR_T_DLY_LAM[i] > 1
        then
                FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
        endif

elseif abs(O2L_AFR_DLY[i]) ≥ O2L_CAT_DIAG_MISS_AFR[i]
then
        if      CTR_T_DLY_LAM[i] = 1
        then
                if      LV_FAC_P_POS_O2L[i] = 0
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                   IP_FAC_P_NEG_LAM * C_FAC_GAIN_P_LAM
                elseif  LV_FAC_P_POS_O2L[i] = 1
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] - FAC_P_POS_O2L[i] -
                                   IP_FAC_P_NEG_LAM * C_FAC_GAIN_P_LAM
                        LV_FAC_P_POS_O2L[i] = 0
                endif
                        LV_FAC_P_POS_LAM[i] = 0

        elseif  CTR_T_DLY_LAM[i] = 0
        then
                FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                      IP_FAC_I_NEG_LAM *C_FAC_GAIN_I_LAM
        elseif  CTR_T_DLY_LAM[i] > 1
```

```
            then
                    FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
            endif
        endif
    endif
```

Calculation of the controller output on the lean side 'LV_AFL[i] = 1'

```
    if      (LV_O2L_LAM_RLS[i] = 1 and LC_FAC_P_O2L_LAM = 1 and LV_AFL[i] = 1)
    then
        if abs(O2L_AFL_DLY[i]) < O2L_CAT_DIAG_MISS_AFL[i]
        then
            if      CTR_T_DLY_LAM[i] = 1
            then
                    if      LV_FAC_P_NEG_O2L[i] = 0
                    then
                            if IP_FAC_P_NEG_O2L - FAC_LAM_OUT[i]
                                            + FAC_LAM_OUT_MV_CAT_DIAG ≤
                                                    IP_FAC_NEG_O2L_MAX
                            then
                                    FAC_P_NEG_O2L[i] = IP_FAC_P_NEG_O2L
                            else
                                        FAC_P_NEG_O2L[i] = IP_FAC_NEG_O2L_MAX
                                    + FAC_LAM_OUT[i] - FAC_LAM_OUT_MV_CAT_DIAG
                            endif
                            if      (FAC_P_NEG_O2L[i] -  FAC_LAM_OUT[i]
                                            + FAC_LAM_OUT_MV_CAT_DIAG[i]) ≤
                                                    C_FAC_ DIF _O2L_MIN
                            then
                            FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG[i]  -
                                            C_FAC_ DIF _O2L_MIN
                            else
                                    FAC_LAM_OUT[i] = FAC_LAM_OUT[i] –
                                                    FAC_P_NEG_O2L[i]
                            endif

                            LV_FAC_P_NEG_O2L[i] = 1

                    elseif  LV_FAC_P_NEG_O2L[i] = 1
                    then
                            FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
                    endif
            elseif  CTR_T_DLY_LAM[i] = 0
            then
                    FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                    IP_FAC_I_POS_LAM *C_FAC_GAIN_I_LAM
            elseif  CTR_T_DLY_LAM[i] > 1
            then
                    FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
            endif

        elseif  abs(O2L_AFL_DLY[i]) ≥ O2L_CAT_DIAG_MISS_AFL[i]
        then
```

```
            if      CTR_T_DLY_LAM[i] = 1
            then
                    if      LV_FAC_P_NEG_O2L[i] = 0
                    then
                            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                          IP_FAC_P_POS_LAM * C_FAC_GAIN_P_LAM
                    elseif  LV_FAC_P_NEG_O2L[i] = 1
                    then
                            FAC_LAM_OUT[i] = FAC_LAM_OUT[i] + FAC_P_NEG_O2L[i]
                                        + IP_FAC_P_POS_LAM * C_FAC_GAIN_P_LAM
                            LV_FAC_P_NEG_O2L[i] = 0
                    endif
                    LV_FAC_P_POS_LAM[i] = 1

            elseif  CTR_T_DLY_LAM[i] = 0
            then
                    FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                            IP_FAC_I_POS_LAM *C_FAC_GAIN_I_LAM
            elseif  CTR_T_DLY_LAM[i] > 1
            then
                    FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
            endif
        endif
    endif
```

7.47.5.3.5.2.2 Calculation of the controller output - case 'LC_FAC_P_O2L_LAM = 0':

<u>Calculation of the controller output on the rich side 'LV_AFL[i] = 0'</u>

```
    if   (LV_O2L_LAM_RLS[i] = 1 and LC_FAC_P_O2L_LAM = 0 and LV_AFL[i] = 0)
    then
        if      abs(O2L_AFR_DLY[i]) < O2L_CAT_DIAG_MISS_AFR[i]
        then
            if CTR_T_DLY_LAM[i] = 1
            then
                  if        (FAC_LAM_OUT[i] - FAC_LAM_OUT_MV_CAT_DIAG[i]) ≤
                                                              C_FAC_DIF _O2L_MIN
                  then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG[i]  +
                                                  C_FAC_ DIF _O2L_MIN
                  else
                          FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
                  endif

            elseif  CTR_T_DLY_LAM[i] = 0
            then
                    FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                            IP_FAC_I_NEG_LAM *C_FAC_GAIN_I_LAM
            elseif  CTR_T_DLY_LAM[i] > 1
            then
                    FAC_LAM_OUT[i]k = FAC_LAM_OUT[i]k-1
            endif
```

```
        elseif  abs(O2L_AFR_DLY[i]) ≥ O2L_CAT_DIAG_MISS_AFR[i]
        then
                if CTR_T_DLY_LAM[i] = 1
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                     IP_FAC_P_NEG_LAM * C_FAC_GAIN_P_LAM
                        LV_FAC_P_POS_LAM[i] = 0

                elseif  CTR_T_DLY_LAM[i] = 0
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] -
                                      IP_FAC_I_NEG_LAM *C_FAC_GAIN_I_LAM
                elseif  CTR_T_DLY_LAM[i] > 1
                then
                        FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1
                endif
        endif
    endif
```

Calculation of the controller output on the lean side 'LV_AFL[i] = 1'

```
    if    (LV_O2L_LAM_RLS[i] = 1 and LC_FAC_P_O2L_LAM = 0 and LV_AFL[i] = 1)
    then
        if abs(O2L_AFL_DLY[i]) < O2L_CAT_DIAG_MISS_AFL[i]
        then
                if CTR_T_DLY_LAM[i] = 1
                then
                        if (FAC_LAM_OUT_MV_CAT_DIAG[i] -  FAC_LAM_OUT[i] ) ≤
                                                      C_FAC_DIF_O2L_MIN
                        then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT_MV_CAT_DIAG[i]  -
                                              C_FAC_DIF_O2L_MIN
                        else
                        FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1

                        endif

                elseif  CTR_T_DLY_LAM[i] = 0
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                      IP_FAC_I_POS_LAM *C_FAC_GAIN_I_LAM
                elseif  CTR_T_DLY_LAM[i] > 1
                then
                        FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1

                endif

        elseif abs(O2L_AFL_DLY[i]) ≥ O2L_CAT_DIAG_MISS_AFL[i]
        then
                if CTR_T_DLY_LAM[i] = 1
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
```

```
                                        IP_FAC_P_POS_LAM * C_FAC_GAIN_P_LAM
                        LV_FAC_P_POS_LAM[i] = 1

                elseif CTR_T_DLY_LAM[i] = 0
                then
                        FAC_LAM_OUT[i] = FAC_LAM_OUT[i] +
                                                IP_FAC_I_POS_LAM *C_FAC_GAIN_I_LAM
                elseif CTR_T_DLY_LAM[i] > 1
                then
                        FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1

                endif
        endif
endif
```

7.47.5.4 Calculation of lambda mean value FAC_LAM_OUT_MV_CAT_DIAG[i] and O_2 - load

```
if      LV_FAC_P_POS_LAM[i] = 0 → 1
then
        O2L_LAM[i] = abs(O2L_AFL_LAM[i])
        FAC_LAM_OUT_MV_CAT_DIAG_SAVE[i] = FAC_LAM_OUT_MV_CAT_DIAG[i]
        FAC_LAM_OUT_MV_CAT_DIAG[i] = MAF_SUM_FAC_LAM_OUT[i]/MAF_CYL_SUM[i]
        T_AFR_CYC_LAM[i] = 0
        MAF_SUM_FAC_LAM_OUT[i] = 0
        MAF_CYL_SUM[i] = 0
        O2L_AFR_LAM[i] = 0
endif
if      LV_FAC_P_POS_LAM[i] = 1 → 0
then
        T_SUM_AFL_AFR_CYC_LAM[i] = T_AFL_CYC_LAM[i] + T_AFR_CYC_LAM[i]
        O2L_LAM[i] = abs(O2L_AFR_LAM[i])
        T_AFL_CYC_LAM[i] = 0
        O2L_AFL_LAM[i] = 0
endif

if      LV_FAC_P_POS_LAM[i] = 1
then
        T_AFR_CYC_LAM[i] = T_AFR_CYC_LAM[i] + T_SAMPLE
        O2L_AFR_LAM[i] = O2L_AFR_LAM[i] + 0.23*MAF_CYL[kg/h]
                                        *T_SAMPLE[ms]*(1/3600)[(g*h)/(kg*ms)]
                        *(FAC_LAM_OUT_MV_CAT_DIAG[i] - FAC_LAM_OUT[i])/100
endif

if      LV_FAC_P_POS_LAM[i] = 0
then
        T_AFL_CYC_LAM[i] = T_AFL_CYC_LAM[i] + T_SAMPLE
        O2L_AFL_LAM[i] = O2L_AFL_LAM[i] + 0.23*MAF_CYL[kg/h]
                                        *T_SAMPLE[ms]*(1/3600)[(g*h)/(kg*ms)]
                        *(FAC_LAM_OUT_MV_CAT_DIAG[i] - FAC_LAM_OUT[i])/100
endif

if      (CTR_T_DLY_LAM[i] = 0 or CTR_T_DLY_LAM[i] > 1)
then
```

```
    MAF_SUM_FAC_LAM_OUT[i] = MAF_SUM_FAC_LAM_OUT[i] +
                                          MAF_CYL*FAC_LAM_OUT[i]
    MAF_CYL_SUM[i] = MAF_CYL_SUM[i] + MAF_CYL
endif

if  (LV_O2L_MISS_AFR_UPD[i] = 1 and LV_FAC_P_POS_LAM[i] = 1)
then
    O2L_AFR_DLY[i] = O2L_AFR_DLY[i] + 0.23*MAF_CYL[kg/h]
                          *T_SAMPLE[ms]*(1/3600)[(g*h)/(kg*ms)]
                   *(FAC_LAM_OUT_MV_CAT_DIAG[i] - FAC_LAM_OUT[i])/100
endif

if  (LV_O2L_MISS_AFL_UPD[i] = 1 and LV_FAC_P_POS_LAM[i] = 0)
then
    O2L_AFL_DLY[i] = O2L_AFL_DLY[i] + 0.23*MAF_CYL[kg/h]
                          *T_SAMPLE[ms]*(1/3600)[(g*h)/(kg*ms)]
                   *(FAC_LAM_OUT_MV_CAT_DIAG[i] - FAC_LAM_OUT[i])/100
endif

if  (LV_O2L_MISS_AFL_UPD[i] = 0 and LV_O2L_MISS_AFR_UPD[i] = 1)
then
    O2L_AFL_DLY[i] = 0
endif

if  (LV_O2L_MISS_AFL_UPD[i] = 1 and LV_O2L_MISS_AFR_UPD[i] = 0)
then
    O2L_AFR_DLY[i] = 0
endif
```

7.47.5.5 Controller shift by canister purge function or EVAP- System diagnosis

The controller output FAC_LAM_OUT shall be manipulated when the flag LV_FAC_LAM_SHIFT_CP or LV_FAC_LAM_DIAGCP are set to 1.

```
if  LV_O2L_LAM_RLS[i] = 0
then
    if    CTR_T_DLY_LAM[i] <= 1
    then
          if    LV_FAC_LAM_SHIFT_CP = 1
          then
                if    LV_FAC_LAM_SHIFT_CP_END[i] = 0
                then
                      FAC_LAM_OUT[i]_k= FAC_LAM_OUT[i]_k-1 + FAC_LAM_SHIFT_CP
                      FAC_LAM[i]_k= FAC_LAM[i]_k-1 + FAC_LAM_SHIFT_CP
                      FAC_LAM_MV_MMV[i]_k = FAC_LAM_MV_MMV[i]_k-1 +
                                                    FAC_LAM_SHIFT_CP
                      FAC_LAM_MV_MMV_CP[i]_k = FAC_LAM_MV_MMV_CP[i]_k-1 +
                                                    FAC_LAM_SHIFT_CP
                      FAC_LAM_MV_MMV_LDC[i]_k = FAC_LAM_MV_MMV_LDC[i]_k-1 +
                                                    FAC_LAM_SHIFT_CP
                      FAC_LAM_MV_MMV_LDC_DIAG[i]_k =
                          FAC_LAM_MV_MMV_LDC_DIAG[i]_k-1 + FAC_LAM_SHIFT_CP
                      LV_FAC_LAM_SHIFT_CP_END[i] = 1
                      MFF_ADD_CP = MFF_ADD_LAM_CP
```

```
                        else
                                MFF_ADD_CP k = MFF_ADD_CP k-1
                        endif
                else
                        LV_FAC_LAM_SHIFT_CP_END[i] = 0
                        MFF_ADD_CP = MFF_ADD_LAM_CP
                endif
                if      LV_FAC_LAM_DIAGCP = 1
                then
                        if      LV_FAC_LAM_DIAGCP_END[i] = 0
                        then
                                FAC_LAM_OUT[i] = FAC_LAM_DIAGCP[i]
                                FAC_LAM[i]k-1 = FAC_LAM_DIAGCP[i]
                                FAC_LAM_MV_MMV[i] = FAC_LAM_DIAGCP[i]
                                FAC_LAM_MV_MMV_CP[i] = FAC_LAM_DIAGCP[i]
                                FAC_LAM_MV_MMV_LDC[i] = FAC_LAM_DIAGCP[i]
                                FAC_LAM_MV_MMV_LDC_DIAG[i] = FAC_LAM_DIAGCP[i]
                                LV_FAC_LAM_DIAGCP_END[i] = 1
                        endif
                else
                        LV_FAC_LAM_DIAGCP_END[i] = 0
                endif
        endif
elseif  LV_O2L_LAM_RLS[i] = 1
then
        if      LV_FAC_P_POS_LAM[i] = 0 → 1
        then
                if      LV_FAC_LAM_SHIFT_CP = 1
                then
                        if      LV_FAC_LAM_SHIFT_CP_END[i] = 0
                        then
                                FAC_LAM_OUT[i]k= FAC_LAM_OUT[i]k-1 + FAC_LAM_SHIFT_CP
                                FAC_LAM[i]k= FAC_LAM[i]k-1 + FAC_LAM_SHIFT_CP
                                FAC_LAM_MV_MMV[i]k = FAC_LAM_MV_MMV[i]k-1 +
                                                                FAC_LAM_SHIFT_CP
                                FAC_LAM_MV_MMV_CP[i]k = FAC_LAM_MV_MMV_CP[i]k-1 +
                                                                FAC_LAM_SHIFT_CP
                                FAC_LAM_MV_MMV_LDC[i]k = FAC_LAM_MV_MMV_LDC[i]k-1 +
                                                                FAC_LAM_SHIFT_CP
                                FAC_LAM_MV_MMV_LDC_DIAG[i]k =
                                    FAC_LAM_MV_MMV_LDC_DIAG[i]k-1 + FAC_LAM_SHIFT_CP
                                FAC_LAM_OUT_MV_CAT_DIAG[i] =
                                        FAC_LAM_OUT_MV_CAT_DIAG[i] + FAC_LAM_SHIFT_CP
                                LV_FAC_LAM_SHIFT_CP_END[i] = 1
                                MFF_ADD_CP = MFF_ADD_LAM_CP
                        else
                                MFF_ADD_CP k = MFF_ADD_CP k-1
                        endif
                else
                        LV_FAC_LAM_SHIFT_CP_END[i] = 0
                        MFF_ADD_CP = MFF_ADD_LAM_CP
                endif
        endif
```

endif

7.47.5.6 Controller shift by lambda adaptation

The controller output FAC_LAM_OUT shall be manipulated when the flag LV_FAC_LAM_ADJ_LAM_AD is set to 1.
The value of LV_FAC_LAM_ADJ_LAM_AD_END shall be set to 1 if the shift was carried out. Lambda adaptation would then reset the flag LV_FAC_LAM_ADJ_LAM_AD.

```
if    LV_O2L_LAM_RLS[i] = 0
then
    if      CTR_T_DLY_LAM[i] <= 1
    then
            if      LV_FAC_LAM_ADJ_LAM_AD[i] = 1
            then
                    if      LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0
                    then
                            FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM[i]_k = FAC_LAM[i]_k-1 - FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM_MV_MMV[i]_k      = FAC_LAM_MV_MMV[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM_MV_MMV_CP[i]_k = FAC_LAM_MV_MMV_CP[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM_MV_MMV_LDC[i]_k = FAC_LAM_MV_MMV_LDC[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM_MV_MMV_LDC_DIAG[i]_k =
                                            FAC_LAM_MV_MMV_LDC_DIAG[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            LV_FAC_LAM_ADJ_LAM_AD_END[i] = 1
                            MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]
                            FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]
                    else
                            MFF_ADD_LAM_AD_OUT[i] _k = MFF_ADD_LAM_AD_OUT[i] _k-1
                            FAC_LAM_AD_OUT[i] _k = FAC_LAM_AD_OUT[i] _k-1
                    endif
            else
                    LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0
                    MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]
                    FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]
            endif
    endif
 elseif     LV_O2L_LAM_RLS[i] = 1
 then
    if      LV_FAC_P_POS_LAM[i] = 0 → 1
    then
            if      LV_FAC_LAM_ADJ_LAM_AD[i] = 1
            then
                    if      LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0
                    then
                            FAC_LAM_OUT[i]_k = FAC_LAM_OUT[i]_k-1 -
                                                        FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM[i]_k = FAC_LAM[i]_k-1 - FAC_LAM_ADJ_LAM_AD[i]
                            FAC_LAM_MV_MMV[i]_k      = FAC_LAM_MV_MMV[i]_k-1 -
```

```
                                        FAC_LAM_ADJ_LAM_AD[i]
                FAC_LAM_MV_MMV_CP[i]k = FAC_LAM_MV_MMV_CP[i]k-1 -
                                        FAC_LAM_ADJ_LAM_AD[i]
                FAC_LAM_MV_MMV_LDC[i]k = FAC_LAM_MV_MMV_LDC[i]k-1 -
                                        FAC_LAM_ADJ_LAM_AD[i]
                FAC_LAM_MV_MMV_LDC_DIAG[i]k =
                                   FAC_LAM_MV_MMV_LDC_DIAG[i]k-1 -
                                        FAC_LAM_ADJ_LAM_AD[i]
                FAC_LAM_OUT_MV_CAT_DIAG[i] =
                 FAC_LAM_OUT_MV_CAT_DIAG[i] - FAC_LAM_ADJ_LAM_AD[i]

                LV_FAC_LAM_ADJ_LAM_AD_END[i] = 1
                MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]
                FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]
            else
                MFF_ADD_LAM_AD_OUT[i] k = MFF_ADD_LAM_AD_OUT[i] k-1
                FAC_LAM_AD_OUT[i] k = FAC_LAM_AD_OUT[i] k-1
            endif
        else
            LV_FAC_LAM_ADJ_LAM_AD_END[i] = 0
            MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i]
            FAC_LAM_AD_OUT[i] = FAC_LAM_AD_LAM_OUT[i]
        endif
    endif
endif
```

7.47.5.7 Controller limitation - calculation of FAC_LAM_LIM:

The value of the controller output shall be limited to FAC_LAM_MIN[i] and FAC_LAM_MAX[i] respectively.

```
if     FAC_LAM_OUT[i] > FAC_LAM_MAX[i]
then
    LV_FAC_LAM_LIM_MAX[i] = 1
    LV_FAC_LAM_LIM_MIN[i] = 0
    FAC_LAM_LIM[i] = FAC_LAM_MAX[i]
elseif     FAC_LAM_OUT[i] < FAC_LAM_MIN[i]
then
    LV_FAC_LAM_LIM_MIN[i] = 1
    LV_FAC_LAM_LIM_MAX[i] = 0
    FAC_LAM_LIM[i] = FAC_LAM_MIN[i]
else
    LV_FAC_LAM_LIM_MIN[i] = 0
    LV_FAC_LAM_LIM_MAX[i] = 0
    FAC_LAM_LIM[i] = FAC_LAM_OUT[i]
endif

FAC_LAM_OUT[i] = FAC_LAM_LIM[i]
```

7.47.5.8 Calculation of P-jump counter, FAC_LAM and FAC_LAM_MV

A P-jump takes place if a new P-part correction is done on FAC_LAM_OUT. FAC_LAM is the value of the controller output after each P-jump and FAC_LAM_MV is the mean value of two successive output signals after each P-jump.

if (LV_FAC_P_POS_LAM[i] = 0 $\rightarrow$ 1 **or** LV_FAC_P_POS_LAM[i] = 1 $\rightarrow$ 0)

then

FAC_LAM[i] = FAC_LAM_OUT[i]
FAC_LAM_MV[i]$_k$ = (FAC_LAM[i]$_{k-1}$ + FAC_LAM[i]$_k$) / 2

increment CTR_RAF_CHG[i]

The P-jump counter shows the number of executed P-jumps. The counter is increased without regard of overflow, i.e. if “255” is reached, with the next P-jump CTR_RAF_CHG[i] contains “0”.

endif

7.47.5.9 Calculation of FAC_LAM_MV_MMV_CP, FAC_LAM_MV_MMV, FAC_LAM_MV_MMV_LDC and FAC_LAM_MV_MMV_LDC_DIAG

FAC_LAM_MV_MMV_CP[i]$_k$ = FAC_LAM_MV_MMV_CP[i]$_{k-1}$ + (FAC_LAM_OUT[i]$_k$ - FAC_LAM_MV_MMV_CP[i]$_{k-1}$) * CRLC_FAC_LAM_MV_MMV_CP

used by canister purge function

FAC_LAM_MV_MMV[i]$_k$ = FAC_LAM_MV_MMV[i]$_{k-1}$+ (FAC_LAM_MV[i]$_k$ - FAC_LAM_MV_MMV[i]$_{k-1}$) * C_CRLC_FAC_LAM_MV_MMV

FAC_LAM_MV_MMV_LDC[i]$_k$ = FAC_LAM_MV_MMV_LDC[i]$_{k-1}$+ (FAC_LAM_MV[i]$_k$- FAC_LAM_MV_MMV_LDC[i]$_{k-1}$)*C_CRLC_FAC_LAM_MV_MMV_LDC

used by the function "Mean value calculation for limited dynamics"

FAC_LAM_MV_MMV_LDC_DIAG[i]$_k$ = FAC_LAM_MV_MMV_LDC_DIAG[i]$_{k-1}$+ (FAC_LAM_MV[i]$_k$- FAC_LAM_MV_MMV_LDC_DIAG[i]$_{k-1}$) *C_CRLC_FAC_LAM_MV_MMV_LDC_DIAG

used by limited dynamics for cat efficiency diagnosis

7.47.5.10 Calculation of the absolute maximum value of lambda controller output

if LV_FAC_P_POS_LAM[i] = 0 $\rightarrow$ 1
then
FAC_LAM_ABSV_STAT_MAX[i] = **abs(**FAC_LAM_ABSV_AFR[i] - FAC_LAM_ABSV_AFL[i]**)**
endif

if LV_FAC_P_POS_LAM[i] = 1
then
FAC_LAM_ABSV_AFR[i] = FAC_LAM_OUT[i]

elseif LV_FAC_P_POS_LAM[i] = 0
then
FAC_LAM_ABSV_AFL[i] = FAC_LAM_OUT[i]
endif

7.47.5.11 Update of CTR_T_DLY_LAM

```
if    LV_IS = 1
then
      LV_O2L_LAM_RLS[i] = 0
      LV_FAC_P_POS_O2L[i] = 0
      LV_FAC_P_NEG_O2L[i] = 0
endif
if LV_O2L_LAM_RLS[i] = 0
then
      if CTR_T_DLY_LAM[i] > 0
      then
      decrement    CTR_T_DLY_LAM[i]
            if     CTR_T_DLY_LAM[i] > 1
            then
                   if LV_AFL[i] = 0
                   then
                         if     VLS_DIF_LAM_ADJ[i] < C_VLS_DIF_LAM_ADJ_BOL
                         then
                                CTR_T_DLY_LAM[i] = 1
                         endif
                   else
                         if     VLS_DIF_LAM_ADJ[i] > C_VLS_DIF_LAM_ADJ_TOL
                         then
                                CTR_T_DLY_LAM[i] = 1
                         endif
                   endif
            endif
      endif
elseif      LV_O2L_LAM_RLS[i] = 1
then
      if CTR_T_DLY_LAM[i] > 1
      then
              decrement   CTR_T_DLY_LAM[i]
      elseif  CTR_T_DLY_LAM[i] = 1
      then
            if (LV_FAC_P_POS_LAM[i] = 0 → 1 or LV_FAC_P_POS_LAM[i] = 1 → 0)
```

```
            then
                    CTR_T_DLY_LAM[i] = 0
            endif
    endif

endif
```

7.47.6 Calculation of FAC_LAM_COR:

FAC_LAM_COR[i] = FAC_LAM_LIM[i] + FAC_LAM_PCTL[i]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_LAM_MIN	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Minimum FAC_LAM_OUT value					
C_FAC_LAM_MAX	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Maximum FAC_LAM_OUT value					
C_FAC_LAM_INI_MIN	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Minimum FAC_LAM_OUT value					
C_FAC_LAM_INI_MAX	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Maximum FAC_LAM_OUT value					
C_FAC_LAM_MV_CAT_DIAG_DIF_MAX	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Maximum difference between the actual mean value and the stored mean value to set LV_O2L_LAM_RLS					
C_FAC_P_MV_MAX	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
Maximum permitted p – jump heading the stored mean value in case of LV_O2L_LAM_RLS 1 → 0					
C_T_DLY_POS_IS_LAM_AT	1	0...FFFFH	0...1279.980469	1.9531E-2	ms
Basic LAM -P-jump delay time from lean to rich transition in idle (Automatic shifted transmission)					
C_T_DLY_NEG_IS_LAM_AT	1	0...FFFFH	0...1279.980469	1.9531E-2	ms
Basic LAM -P-jump delay time from rich to lean transition in idle (Automatic shifted transmission)					
C_T_DLY_POS_IS_LAM_MT	1	0...FFFFH	0...1279.980469	1.9531E-2	ms
Basic LAM -P-jump delay time from lean to rich transition in idle (Manual shifted transmission)					
C_T_DLY_NEG_IS_LAM_MT	1	0...FFFFH	0...1279.980469	1.9531E-2	ms
Basic LAM -P-jump delay time from rich to lean transition in idle (Manual shifted transmission)					
C_CRLC_FAC_LAM_MV_MMV	1	0...FFH	0...0.996094	3.9063E-3	-
Correlation constant for the calculation of FAC_LAM_MV_MMV					
C_CRLC_FAC_LAM_MV_MMV_LDC	1	0...FFH	0...0.996094	3.9063E-3	-
Correlation constant for the calculation of FAC_LAM_MV_MMV_LDC					
C_CRLC_FAC_LAM_MV_MMV_LDC_DIAG	1	0...FFH	0...0.996094	3.9063E-3	-
Correlation constant for the calculation of FAC_LAM_MV_MMV_LDC_DIAG					
IP_FAC_P_POS_IS_LAM	8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
Basic value of FAC_LAM_OUT proportional component in idle during transition from lambda < 1 to lambda > 1					
IP_FAC_P_NEG_IS_LAM	8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
Basic value of FAC_LAM_OUT proportional component in idle during transition from lambda > 1 to lambda < 1					
IP_FAC_I_POS_IS_LAM	8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
Basic value of FAC_LAM_OUT integral component in idle when lambda > 1					
IP_FAC_I_NEG_IS_LAM	8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
Basic value of FAC_LAM_OUT integral component in idle when lambda < 1					
IP_FAC_P_POS_LAM	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic value of FAC_LAM_OUT proportional component out of idle during transition from lambda<1 to lambda>1					
IP_FAC_P_POS_O2L	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
P component of additional O2-load on the rich side					
IP_FAC_POS_O2L_MAX	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Maximum of lambda controller output in case of additional O2-load on the rich side					
IP_FAC_P_NEG_LAM	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm

Basic value of FAC_LAM_OUT proportional component out of idle during transition from lambda>1 to lambda<1					
IP_FAC_P_NEG_O2L	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
P component of additional O2-load on the lean side					
IP_FAC_NEG_O2L_MAX	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Maximum of lambda controller output in case of additional O2-load on the lean side					
IP_FAC_I_POS_LAM	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic value of FAC_LAM_OUT integral component out of idle when lambda > 1					
IP_FAC_I_NEG_LAM	8 * 8	0...7FH	0...0.496094	3.9063E-3	-
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic value of FAC_LAM_OUT integral component out idle when lambda < 1					
IP_T_DLY_POS_LAM_AT	8 * 8	0...FFFFH	0...1279.980469	1.9531E-2	ms
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic LAM -P-jump delay time from lean to rich transition (Automatic shifted transmission)					
IP_T_DLY_NEG_LAM_AT	8 * 8	0...FFFFH	0...1279.980469	1.9531E-2	ms
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic LAM -P-jump delay time from rich to lean transition (Automatic shifted transmission)					
IP_T_DLY_POS_LAM_MT	8 * 8	0...FFFFH	0...1279.980469	1.9531E-2	ms
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic LAM -P-jump delay time from lean to rich transition (Manual shifted transmission)					
IP_T_DLY_NEG_LAM_MT	8 * 8	0...FFFFH	0...1279.980469	1.9531E-2	ms
LDPM_MAF_1_LACO	8	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_1_LACO	8	0...FFH	0...8160	32	rpm
Basic LAM -P-jump delay time from rich to lean transition (Manual shifted transmission)					
C_NR_RAF_CHG_IS_LAM	1	0...FFH	0...255	1	-
Lambda controller P jump delay to have idle P/I values					
C_CTR_AFR_TRO_MIN	1	0...FFH	0...255	1	-
Minimum number of cycles to release the O2 – load based lambda controller after disturbance on the rich side					
C_CTR_AFL_TRO_MIN	1	0...FFH	0...255	1	-
Minimum number of cycles to release the O2 – load based lambda controller after disturbance on the lean side					
C_FAC_GAIN_P_LAM	1	0...FFH	0...0.996094	3.9063E-3	-
Scale factor for FAC_LAM_OUT proportional component					
C_FAC_GAIN_I_LAM	1	0...FFH	0...0.996094	3.9063E-3	-
Scale factor for FAC_LAM_OUT integral component					
C_FAC_MV_O2L_MAX	1	0...FFH	0...3.984375	0.015625	-
Scaling factor for maximum threshold to release the O2-load adjustment					
C_VLS_DIF_LAM_ADJ_BOL	1	FC00...3FFH	-5...4.995117	4.8828E-3	V
Bottom limit of difference between setpoint and actual downstream LS signal to hold the P-jump delay					
C_VLS_DIF_LAM_ADJ_TOL	1	FC00...3FFH	-5...4.995117	4.8828E-3	V
Top limit of difference between setpoint and actual downstream LS signal to hold the P-jump delay					
C_MAF_INT_O2L_TRO_THD	1	0...FFFFH	0...1820.416667	0.027778	g
Threshold of MAF - integral during the disturbed O2 - load calculation					
C_T_DLY_CHG_LAM_MIN	1	0...FFFFH	0...1279.980469	1.9531E-2	ms
Minimum p jump dly change to set LV_T_DLY_CHG_LAM and recalculate CTR_T_DLY_LAM					
LC_FAC_P_O2L_LAM	1	0...1H	0...1	1	-
Switch to apply the P-jump for calculation of additional O2-load					
C_FAC_DIF_O2L_MIN	1	0...7FH	0...0.496094	3.9063E-3	-

Minimum difference between lambda controller output and mean value in case of additional O2-load

7.48 Application Incidences for a binary lambda controller

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LAM_LSCL[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation flag of lambda control					
T_DLY_LSCL_REAC_PUC	V	0...FFFFH	0...655.35	0.01	s
Delay after PUC before re-enabling Lambda control					
T_LAM_STOP	V	0...FFFFH	0...655.35	0.01	s
Blocking time of lambda controller after MFF_ADD_WF falling under an applicable threshold					
LV_LAM_STOP_INTR	V	0...1H	0...1	1	-
Flag to interrupt the lambda control in case of exceeding a certain MFF_ADD_WF threshold					
STATE_LS[NC_CBK_EX_NR]	V/O	0...FFH	0...255	1	-
Status of the fuel system - cylinder bank i					

Input data:

LV_ES	LV_ST	LV_ERR_SCP_LS_UP	LV_FL
LV_PUC	LV_FCUT_IND	LV_ASR_IND_ACT	
TCO	LV_IS	NC_CBK_EX_NR	MAF
N_32	TCO_ST	MFF_ADD_WF	CONF_LAM
LV_ERR_IV[NC_CYL_NR]	LV_MIS_STATE_A	LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]
LV_ERR_SCG_LS_UP	LV_ERR_PUC_LS_UP	LV_ERR_STK_LS_UP	LV_ERR_CHG_LS_UP
FAC_LAM_OUT[NC_CBK_EX_NR]		LV_CAT_PURGE_ACT[NC_CBK_EX_NR]	
LV_LS_UP_READY[NC_CBK_EX_NR]		LV_ERR_LSH_UP[NC_CBK_EX_NR]	
LV_ERR_READY_LS_UP[NC_CBK_EX_NR]		LV_ERR_OC_LS_UP	
LV_LAMB_OHP[NC_CBK_EX_NR]		C_T_DLY_LSCL_REAC_PUC	
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]		TEG_DYN_LS_UP[NC_CBK_EX_NR]	

FUNCTION DESCRIPTION:

The calculation shall be done for all exhaust cylinder banks.

For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2

otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank

Application conditions:

Initialisation:

Recurrence: every 10 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

if LV_PUC = 1
then T_DLY_LSCL_REAC_PUC = C_T_DLY_LSCL_REAC_PUC

if LV_PUC = 0
then **if** T_DLY_LSCL_REAC_PUC > 0
then ***decrement*** T_DLY_LSCL_REAC_PUC

if FAC_LAM_OUT[i] < 0
then **if** **abs(**MFF_ADD_WF**)** > IP_MFF_ADD_WF_MAX_LAM
then LV_LAM_STOP_INTR = 1
T_LAM_STOP = IP_T_LAM_STOP

if **abs(**MFF_ADD_WF**)** ≤ IP_MFF_ADD_WF_MAX_LAM
then **if** (LV_LAM_STOP_INTR = 1 **and** T_LAM_STOP > 0)
then ***decrement*** (T_LAM_STOP)
else LV_LAM_STOP_INTR = 0

Determination of LV_LAM_LSCL:

if LV_ES = 0 *(engine stopped inactive)*
and LV_ST = 0 *(engine operating state start inactive)*
and CONF_LAM = 1 *(Closed loop lambda control)*
and LV_CAT_PURGE_ACT[i] = 0 *(no Cat.Purge active)*
and N_32 > C_N_MIN_LAM_LSCL *(engine speed exceeds threshold)*
and LV_FL = 0 *(full load inactive)*
and LV_LAMB_OHP[i] = 0 *(catalyst overheating prevention function inactive)*
and (LV_FCUT_IND = 0 **and** LV_ASR_IND_ACT = 0) *(no cylinder shut off)*
and LV_PUC = 0 *(no trailing-throttle fuel cut-off)*
and LV_ERR_SCP_LS_UP =0
and LV_ERR_SCG_LS_UP =0
and LV_ERR_OC_LS_UP =0
and LV_ERR_PUC_LS_UP =0
and LV_ERR_STK_LS_UP =0

```
and     LV_ERR_CHG_LS_UP =0
and     ( LV_ERR_SWT_LS_UP = 0 or LC_LAM_LSCL_ERR_LS_UP_SWT = 1 )
and     ( LV_ERR_FRQ_LS_UP = 0 or LC_LAM_LSCL_ERR_LS_UP_FRQ = 1 )
and     ( LV_ERR_LSH_UP = 0 or LC_LAM_LSCL_ERR_LSH_UP = 1 )
and     LV_ERR_OBD_LSH_UP=0 or LC_LAM_LSCL_ERR_OBD_LSH_UP =1 )

        no diagnosis error occurred (except "Too long time for close loop" failure)
and     LV_ERR_IV[NC_CYL_NR] = 0        (no IV error present)
and     LV_MIS_STATE_A = 0              (no CARB A misfire detected)
and     MAF > ID_MAF_MIN_LAM_LSCL       (mass air flow exceeds the threshold)
and     LV_LAM_STOP_INTR = 0
and     (LV_LS_UP_READY[i] = 1          (operability of the lambda probe recognised)
        or    TEG_DYN_LS_UP > C_TEG_DYN_MIN_LSCL_READY)
                                        (operability set to detect open circuit)
and     T_DLY_LSCL_REAC_PUC = 0         (delay after PUC elapsed)
then
    if      LV_IS = 1
    then    if      TCO > ID_TCO_MIN_LAM_LSCL_IS
            then    LV_LAM_LSCL[i] = 1
            else    LV_LAM_LSCL[i] = 0
            endif
    else    if      TCO > ID_TCO_MIN_LAM_LSCL
            then    LV_LAM_LSCL[i] = 1
            else    LV_LAM_LSCL[i] = 0
            endif
    endif
else        LV_LAM_LSCL[i] = 0
endif
```

STATE_LS_1 calculation:

STATE_LS_i	
BIT	**Function**
0	lambda control open loop – has not yet satisfied conditions to go closed loop
1	lambda control closed loop – using oxygen sensor(s) as feedback for fuel control
2	lambda control open loop – due to driving conditions (e.g. power enrichment, decelaration enleanment)
3	lambda control open loop – due to detected system fault
4	lambda control closed loop – but fault with at least one oxygen sensor (may be using single oxygen sensor for fuel control)
5...7	reserved – always 0

Only one of the bits can be set at the same time. One of the bits must be set at every time. So the following values are possible for STATE_LS_1:

STATE_LS_1	Function
1	Lambda control open loop - has not yet satisfied conditions to go closed loop
2	Lambda control closed loop - using oxygen sensor(s) as feedback for fuel control
4	Lambda control open loop – due to driving conditions
8	Lambda control open loop – due to detected system fault
16	Lambda control closed loop – but fault with at least one oxygen sensor

For i = 1 to NC_CBK_NR:

if LV_LAM_LSCL_i = 1

then *(state of the fuel system at closed loop control)*

if LV_ERR_READY_LS_UP = 0

then STATE_LS_i = 2
(Lambda control closed loop - using oxygen sensor(s) as feedback for fuel control)

else STATE_LS_i = 16
(Lambda control closed loop – but fault with at least one oxygen sensor)

endif

else *(state of the fuel system at open loop control)*

if LV_ERR_SCP_LS_UP = 1

or LV_ERR_SCG_LS_UP =1

or LV_ERR_OC_LS_UP =1

or LV_ERR_PUC_LS_UP = 1

or LV_ERR_STK_LS_UP = 1

or (LV_ERR_SWT_LS_UP = 1 **and** LC_LAM_LSCL_ERR_LS_UP_SWT = 0)

or (LV_ERR_FRQ_LS_UP = 1 **and** LC_LAM_LSCL_ERR_LS_UP_FRQ = 0)

or (LV_ERR_LSH_UP =1 **and** LC_LAM_LSCL_ERR_LSH_UP = 0)

or (LV_ERR_OBD_LSH_UP = 1 **and** LC_LAM_LSCL_ERR_OBD_LSH_UP =0)

or LV_ERR_CHG_LS_UP = 1

or LV_ERR_IV[NC_CYL_NR] = 1

or LV_MIS_STATE_A = 1

then STATE_LS_i = 8
(Lambda control open loop – due to detected system fault

else **if** LV_ES = 0

and LV_ST = 0

and CONF_LAM = 1

and LV_LS_UP_READY = 1

then **if** LV_IS = 1

and TCO > ID_TCO_MIN_LAM_LSCL_IS

or LV_IS = 0

and TCO > ID_TCO_MIN_LAM_LSCL

then STATE_LS_i = 4
(Lambda control open loop – due to driving conditions)

else STATE_LS_i = 1
(Lambda control open loop - has not yet satisfied conditions to go closed loop

else STATE_LS_i = 1
(Lambda control open loop - has not yet satisfied conditions to go closed loop

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_TCO_MIN_LAM_LSCL_IS	1*6	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_LACO	6	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature for the activation in idle					
ID_TCO_MIN_LAM_LSCL	1*6	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_LACO	6	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature for the activation in out of idle					
ID_MAF_MIN_LAM_LSCL	1*8	0...FFFFH	0...1389	0.021195	mg/stk
LDP_N_32_ID_MAF_MIN_LAM_LSCL	8	0...FFH	0...8160	32	rpm
Minimum mass air flow for activation					
IP_T_LAM_STOP	1*6	0...FFFFH	0...655.35	0.01	s
LDPM_TCO_1_LACO	6	0...FEH	-48...142.5	0.75	°C
Blocking time for lambda controller after wallfilm compensation					
IP_MFF_ADD_WF_MAX_LAM	1*6	0...FFFFH	-694.5...694.48	0.02119	mg/stk
LDPM_TCO_1_LACO	6	0...FEH	-48...142.5	0.75	°C
MFF_ADD_WF threshold for lambda controller deactivation					
C_N_MIN_LAM_LSCL	1	0...FFH	0...8160	32	rpm
Minimum engine speed for activation					
C_T_DLY_LSCL_REAC_PUC	1	0...FFFFH	0...655.35	0.01	s
Delay after PUC before re-enabling Lambda control					
LC_LAM_LSCL_ERR_LS_UP_FRQ	1	0...1H	0...1	1	-
Switch mode to activate closed loop lambda controller in case of upstream oxygen sensor frequency error					
C_TEG_DYN_MIN_LSCL_READY	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Exhaust gas temperature minimum for the detection of the limited sensor operability					
LC_LAM_LSCL_ERR_LS_UP_SWT	1	0...1H	0...1	1	-
Switch mode to activate closed loop lambda controller in case of upstream oxygen sensor switch time error					
LC_LAM_LSCL_ERR_LSH_UP	1	0...1H	0...1	1	-
Switch mode to activate closed loop lambda controller in case of upstream oxygen sensor heater OBDI error					
LC_LAM_LSCL_ERR_OBD_LSH_UP	1	0...1H	0...1	1	[-]
Switch mode to activate closed loop lambda controller in case of upstream oxygen sensor heater OBDII error					

7.49 Downstream fuel trim regulation by LAM - P - Jump delay time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_DIF_LAM_ADJ[NC_CBK_EX_NR]	V/O	FC00...3FFH	-5...4.995117	4.8828E-3	V
difference between setpoint and actual downstream LS signal					
VLS_DIF_SAVE_LAM_ADJ[NC_CBK_EX_NR]	V	FC00...3FFH	-5...4.995117	4.8828E-3	V
saved difference between setpoint and actual downstream LS signal					
T_DLY_P_LAM_ADJ[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.56...2.5599	7.8125E-5	s
LAM -P-jump delay time from P-share					
T_DLY_I_LAM_ADJ[NC_CBK_EX_NR]	V/O	8000...7FFFH	-640...639.980469	1.9531E-2	ms
LAM –P-jump delay time from I-share					
T_DLY_I_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	V/O	8000...7FFFH	-640...639.980469	1.9531E-2	ms
LAM –P-jump delay time from I-share - in case of released O_2 - load guided lambda control (catalyst diagnosis)					
T_DLY_I_MV_LAM_ADJ[NC_CBK_EX_NR]	V	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Mean value of the LAM –P-jump delay time from I-share					
T_DLY_I_AD_LAM_ADJ[NC_CBK_EX_NR]	V/S	8000...7FFFH	-640...639.980469	1.9531E-2	ms
initial value for the I-share (adaptation value)					
T_DLY_I_MV_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	V	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Mean value of the LAM –P-jump delay time from I-share (in case of catalyst diagnosis)					
T_DLY_I_AD_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	V/S	8000...7FFFH	-640...639.980469	1.9531E-2	ms
initial value for the I-share (adaptation value - catalyst diagnosis)					
T_DLY_LAM_ADJ[NC_CBK_EX_NR]	V/O	8000...7FFFH	-2.56...2.5599	7.8125E-5	s
LAM -P-jump delay time from downstream trim controller					
MAF_INT_LAM_ADJ_ACT[NC_CBK_EX_NR]	V	0...FFFFH	0...1820.416667	0.027778	g
integral of air mass flow since LAM activation					
CTR_PER_VLD_LAM_ADJ[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
counter of controller difference (sign changed)					
LV_LAM_ADJ_ACT[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation flag of dynamic fuel trim					
LV_MAF_INT_LAM_ADJ_CAT_PURGE[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Auxiliary flag indicating the transition of LV_CAT_PURGE_ACT[i]					
LV_LAM_ADJ_I_ACT[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation flag for calculation of the trim-controller I-share					
LV_LAM_ADJ_P_ACT[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation flag for calculation of the trim-controller P-share					
LV_LAM_ADJ_I_CAT_DIAG_ACT[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Activation flag for calculation of the trim-controller I-share - in case of released O_2 - load guided lambda control (catalyst diagnosis)					
LV_LAM_ADJ_PER_VLD[NC_CBK_EX_NR]	V/O	0...01H	0...1	1	-
internal state variable for the monitoring of the controller difference (sign)					
VLS_SP_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...3FFH	0...4.995117	4.8828E-3	V
Setpoint for downstream fuel trim controller					

Input data:

NC_CBK_EX_NR	MAF_CYL	N_32	MAF
LV_ES	LV_DC	MAF_HB	LV_ST_END
VLS_AV_LAM_ADJ[NC_CBK_EX_NR]		EFF_CAT_DIAG[NC_CBK_EX_NR]	
TEMP_CAT_STAT_MDL[NC_CBK_EX_NR]		FAC_LAM_OUT[NC_CBK_EX_NR]	
FAC_LAM_MIN[NC_CBK_EX_NR]		FAC_LAM_MAX[NC_CBK_EX_NR]	
LV_LDC_LAM_ADJ[NC_CBK_EX_NR]		LV_LS_DOWN_READY[NC_CBK_EX_NR]	
LV_INH_LAM_ADJ[NC_CBK_EX_NR]		LV_CAT_PURGE_ACT[NC_CBK_EX_NR]	
LV_LAM_LSCL[NC_CBK_EX_NR]		LV_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	
LV_INH_LAM_ADJ_I[NC_CBK_EX_NR]			

FUNCTION DESCRIPTION:

General information:

The trim-controller causes a compensation of ageing effects of the upstream O_2-sensor as well as a better observance of the catalyst window during all the life of the vehicle.
According to downstream O_2-sensor voltage, a P-jump time delay is applied on the lambda regulation. The control value is composed of P and I-shares, both are determined depending on the downstream probe signal. The variance of the probe voltage from desired value is the basic characteristic. The desired value can be applied depending on the operating point, so that the dynamic lambda can be adjusted according to the operating range.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank

Application conditions:

Initialisation:

T_DLY_I_AD_LAM_ADJ[i] shall be read out the non-volatile memory.

In case of an E²Prom error, T_DLY_I_AD_LAM_ADJ[i] and T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i] shall be initialised by 0.

at reset:

T_DLY_I_LAM_ADJ[i] = T_DLY_I_AD_LAM_ADJ[i]

T_DLY_I_LAM_ADJ_CAT_DIAG[i] = T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i]

T_DLY_I_MV_LAM_ADJ[i] = T_DLY_I_AD_LAM_ADJ[i]

T_DLY_I_MV_LAM_ADJ_CAT_DIAG[i] = T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i]

at start of new driving cycle - LV_DC 0 → 1:

T_DLY_I_LAM_ADJ[i] = T_DLY_I_AD_LAM_ADJ[i]

T_DLY_I_LAM_ADJ_CAT_DIAG[i] = T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i]

T_DLY_I_MV_LAM_ADJ[i] = T_DLY_I_AD_LAM_ADJ[i]

T_DLY_I_MV_LAM_ADJ_CAT_DIAG[i] = T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i]

VLS_DIF_LAM_ADJ[i] = 0

VLS_DIF_SAVE_LAM_ADJ[i] = 0

T_DLY_P_LAM_ADJ[i] = 0

MAF_INT_LAM_ADJ_ACT[i] = 0

LV_LAM_ADJ_ACT[i] = 0

LV_MAF_INT_LAM_ADJ_CAT_PURGE[i] = 0

LV_LAM_ADJ_P_ACT[i] = 0

LV_LAM_ADJ_I_ACT[i] = 0

LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 0

CTR_PER_VLD_LAM_ADJ[i] = 0

LV_LAM_ADJ_PER_VLD[i] = 0

Recurrence: T_SAMPLE = 100 ms

Activation:

- LV_ST_END = 1

Deactivation:

- LV_ES = 1

Formula section:

7.49.1 Calculation of the deviation

The deviation is the difference between the setpoint value and the measured downstream sensor voltage.

7.49.2 Calculation of the internal state variable LV_LAM_ADJ_PER_VLD[i]

A statement regarding the behaviour of the downstream lambda sensor voltage is required e.g. to distinguish inside of the OBD trim-controller diagnosis whether an error on the upstream or downstream sensor occurs. LV_LAM_ADJ_PER_VLD[i] = 1 announces that the controller difference VLS_DIF_LAM_ADJ[i] was at least 10 times positive and 10 times negative. Otherwise and after every function activation must LV_LAM_ADJ_PER_VLD[i] set to 0.

7.49.3 Calculation of controller components

The dynamic fuel controller is made of a proportional- and an I-action controller. At the beginning of a new driving cycle the initial value of the integrator shall be initialised by the stored adaptation value.

7.49.3.1 Calculation of the air-mass-flow-integral

Regarding the first controller update after activation, the integral of air-mass-flow is calculated. Should the value of the integral be below the threshold, the calculation of the controller must be prevented.

7.49.3.1.1 Conditional release after PUC

After the catalyst enrichment function was completed, the threshold C_MAF_INT_LAM_ADJ_ACT_CAT_PURGE is the trigger for the release.

7.49.3.1.2 Conditional release after new activation

Should the value of the integral be over the threshold C_MAF_INT_LAM_ADJ_ACT, the calculation of the controller is released.

7.49.3.2 Calculation of the proportional-action controller (P component)

The P-share of the controller may only be carried out in applied N/MAF-areas. When these conditions are not fulfilled the corresponding P-share shall be set to 0.

7.49.3.3 Calculation of the integral-action controller (I component)

The I-share of the controller may only be carried out in applied MAF-areas provided that the limited dynamic is fulfilled. When these conditions are not fulfilled the corresponding I-share shall be stopped. At the beginning of a new driving cycle T_DLY_I_LAM_ADJ[i] is initialised by the stored value T_DLY_I_AD_LAM_ADJ[i] (adaptation value).
In case of activated catalyst diagnosis, the I component of dynamic fuel trim is released independent of MAF - area, being calculated with IP_T_DLY_I_LAM_ADJ_CAT_DIAG (LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 1).
In case of Cylinder shut-off, I component is stopped via LV_INH_LAM_ADJ_I[i] until VLS_DOWN indicates again a Rich mixture.

7.49.3.4 Limitation of the integrator

The output of the integrator shall be limited to C_T_DLY_I_MIN_LAM_ADJ and C_T_DLY_I_MAX_LAM_ADJ.

7.49.3.5 Calculation of the I-share mean value

Regarding the adaptation, the calculation of the I-share mean value is required using a low-pass filter. At the beginning of a new driving cycle T_DLY_I_MV_LAM_ADJ[i] is initialised by the stored value T_DLY_I_AD_LAM_ADJ[i] (adaptation value).

7.49.4 Calculation of the controller output

The total delay time of the downstream trim regulation is normalised by IP_FAC_T_DLY_LAM_ADJ on the corresponding Lambda shift.

7.49.5 Calculation of the adaptation value T_DLY_I_AD_LAM_ADJ[i] and T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i]

The value T_DLY_I_AD_LAM_ADJ[i] and T_DLY_I_AD_LAM_ADJ_CAT_DIAG[i] shall be calculated at the end of the driving cycle and stored in the flash memory.

```
if    VLS_DIF_LAM_ADJ[i] ≠ 0
then
      VLS_DIF_SAVE_LAM_ADJ[i] = VLS_DIF_LAM_ADJ[i]
endif

VLS_SP_LAM_ADJ[i] = IP_VLS_SP_LAM_ADJ[i]
VLS_DIF_LAM_ADJ[i]  =  VLS_SP_LAM_ADJ[i] - VLS_AV_LAM_ADJ[i]        (7.49.1)

if    (LV_LAM_ADJ_PER_VLD[i] = 0 and
            VLS_DIF_SAVE_LAM_ADJ[i] * VLS_DIF_LAM_ADJ[i] < 0)                 (7.49.2)
then
      increment CTR_PER_VLD_LAM_ADJ[i]
      if    CTR_PER_VLD_LAM_ADJ[i] > 20
      then
            LV_LAM_ADJ_PER_VLD[i] = 1
            CTR_PER_VLD_LAM_ADJ[i] = 0
      endif
endif


if    (LV_INH_LAM_ADJ[i] = 0 and LV_LS_DOWN_READY[i] = 1 and
            LV_CAT_PURGE_ACT[i] = 0 and LV_LAM_LSCL[i] = 1 and
            TEMP_CAT_STAT_MDL[i] > C_TEMP_CAT_MIN_LAM_ADJ_ACT and
            (FAC_LAM_MIN[i] < FAC_LAM_OUT[i] < FAC_LAM_MAX[i]) and
            EFF_CAT_DIAG[i] < C_EFF_CAT_DIAG_MAX_LAM_ADJ_ACT)
then
      MAF_INT_LAM_ADJ_ACT[i]n [g] = MAF_INT_LAM_ADJ_ACT[i]n-1 [g]+
                  MAF_CYL [kg/h]*T_SAMPLE [ms]*1/3600 [(g*h)/(kg*ms)]  (7.49.3.1)


      if    LV_MAF_INT_LAM_ADJ_CAT_PURGE[i] = 1  (7.49.3.1.1)
      then
            if  MAF_INT_LAM_ADJ_ACT[i] > C_MAF_INT_LAM_ADJ_ACT_CAT_PURGE
            then
                LV_LAM_ADJ_ACT[i] = 1
                LV_MAF_INT_LAM_ADJ_CAT_PURGE[i] = 0
            endif
      else
            if  MAF_INT_LAM_ADJ_ACT[i] > C_MAF_INT_LAM_ADJ_ACT       (7.49.3.1.2)
            then
                LV_LAM_ADJ_ACT[i] = 1
            endif
      endif
else
      LV_LAM_ADJ_ACT[i] = 0
      MAF_INT_LAM_ADJ_ACT[i] = 0
      if    LV_MAF_INT_LAM_ADJ_CAT_PURGE[i] = 0
      then
            LV_MAF_INT_LAM_ADJ_CAT_PURGE[i] = LV_CAT_PURGE_ACT[i]
      endif
endif
```

if (1) LV_LAM_ADJ_ACT[i] = 1
then (1)
  if (2) LV_LAM_ADJ_CAT_DIAG[i] = 0 **or** LC_ENA_NORM_TRM_MAN_ACT = 1
  then (2)
    if (3) ((C_MAF_MIN_LAM_ADJ_P_ACT < MAF <
      C_MAF_MAX_LAM_ADJ_P_ACT) **and**
      (C_N_MIN_LAM_ADJ_P_ACT < N_32 < C_N_MAX_LAM_ADJ_P_ACT))
    then (3)
      LV_LAM_ADJ_P_ACT[i] = 1
      T_DLY_P_LAM_ADJ[i] = IP_T_DLY_P_LAM_ADJ[i] *
        IP_FAC_T_DLY_P_LAM_ADJ *
        IP_FAC_T_DLY_P_CAT_DIAG[i] (7.49.3.2.A)
    Else (3)
      LV_LAM_ADJ_P_ACT[i] = 0
      T_DLY_P_LAM_ADJ[i] = 0
    Endif (3)
  Else (2)
    LV_LAM_ADJ_P_ACT[i] = 1
    T_DLY_P_LAM_ADJ[i] = IP_T_DLY_P_LAM_ADJ_CAT_DIAG[i] *
      IP_FAC_T_DLY_P_LAM_ADJ *
      IP_FAC_T_DLY_P_CAT_DIAG[i] (7.49.3.2.B)
  Endif (2)

  If (4) LV_LAM_ADJ_CAT_DIAG[i] = 0 **or** LC_ENA_NORM_TRM_MAN_ACT = 1
  Then (4)
    LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 0
    If (5) [IP_MAF_MIN_LAM_ADJ_I_ACT < MAF <
      IP_MAF_MAX_LAM_ADJ_I_ACT **and**
      LV_LDC_LAM_ADJ[i] = 1 *(Limited dynamic is fulfilled)* **and**
      LV_INH_LAM_ADJ_I[i] = 0]
    Then (5)
      LV_LAM_ADJ_I_ACT[i] = 1
      T_DLY_I_LAM_ADJ[i]$_k$ = T_DLY_I_LAM_ADJ[i]$_{k-1}$ +
        IP_CRLC_T_DLY_I_LAM_ADJ * IP_T_DLY_I_LAM_ADJ[i]
        (7.49.3.3.A)
      In order to avoid rounding error that leads to zero-readout, following calculation shall be carried out:

      if (6) (IP_CRLC_T_DLY_I_LAM_ADJ*IP_T_DLY_I_LAM_ADJ[i] = 0 **and**
        IP_CRLC_T_DLY_I_LAM_ADJ > 0 **and** IP_T_DLY_I_LAM_ADJ > 0)
      Then (6)
        T_DLY_I_LAM_ADJ[i]$_k$ = T_DLY_I_LAM_ADJ[i]$_{k-1}$ + 1.9531E-2 ms

      Elseif (6) (IP_CRLC_T_DLY_I_LAM_ADJ*IP_T_DLY_I_LAM_ADJ[i] = 0 **and**
        IP_CRLC_T_DLY_I_LAM_ADJ > 0 **and** IP_T_DLY_I_LAM_ADJ < 0)
      Then (6)
        T_DLY_I_LAM_ADJ[i]$_k$ = T_DLY_I_LAM_ADJ[i]$_{k-1}$ - 1.9531E-2 ms
      Endif (6)

      T_DLY_I_LAM_ADJ[i]$_k$ = MINMAX(T_DLY_I_LAM_ADJ[i]$_k$,
        C_T_DLY_I_MIN_LAM_ADJ, C_T_DLY_I_MAX_LAM_ADJ) (7.49.3.4.A)

$T\_DLY\_I\_MV\_LAM\_ADJ[i]_k$ = $T\_DLY\_I\_MV\_LAM\_ADJ[i]_{k-1}$ +
($T\_DLY\_I\_LAM\_ADJ[i]_k$ - $T\_DLY\_I\_MV\_LAM\_ADJ[i]_{k-1}$) *
C_CRLC_T_DLY_I_MMV_LAM_ADJ (7.49.3.5.A)

Else (5)
LV_LAM_ADJ_I_ACT[i] = 0
Endif (5)
Else (4)
LV_LAM_ADJ_I_ACT[i] = 0
If (7) LV_INH_LAM_ADJ_I[i] = 0
Then (7)
LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 1
$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$ = $T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_{k-1}$ +
IP_CRLC_T_DLY_I_LAM_ADJ * IP_T_DLY_I_LAM_ADJ_CAT_DIAG[i] (7.49.3.3.B)

In order to avoid rounding error that leads to zero-readout, following calculation shall be carried out:

if (8) (IP_CRLC_T_DLY_I_LAM_ADJ*IP_T_DLY_I_LAM_ADJ_CAT_DIAG[i] =
0
and IP_CRLC_T_DLY_I_LAM_ADJ > 0 **and**
IP_T_DLY_I_LAM_ADJ_CAT_DIAG[i] > 0)
Then (8)
$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$ =
$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_{k-1}$ + 1.9531E-2 ms

Elseif (8) (IP_CRLC_T_DLY_I_LAM_ADJ
*IP_T_DLY_I_LAM_ADJ_CAT_DIAG[i] = 0 **and**
IP_CRLC_T_DLY_I_LAM_ADJ > 0 **and**
IP_T_DLY_I_LAM_ADJ_CAT_DIAG[i] < 0)
Then (8)
$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$ =
$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_{k-1}$ - 1.9531E-2 ms
Endif (8)

$T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$ =
MINMAX($T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$, C_T_DLY_I_MIN_CAT_DIAG,
C_T_DLY_I_MAX_CAT_DIAG) (7.49.3.4.B)
$T\_DLY\_I\_MV\_LAM\_ADJ\_CAT\_DIAG[i]_k$ =
$T\_DLY\_I\_MV\_LAM\_ADJ\_CAT\_DIAG[i]_{k-1}$ +
($T\_DLY\_I\_LAM\_ADJ\_CAT\_DIAG[i]_k$ -
$T\_DLY\_I\_MV\_LAM\_ADJ\_CAT\_DIAG[i]_{k-1}$) *
C_CRLC_T_DLY_I_MMV_CAT_DIAG (7.49.3.5.B)
Else (7)
LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 0
Endif (4)
Else (1)
T_DLY_P_LAM_ADJ[i] = 0
LV_LAM_ADJ_PER_VLD[i] = 0
CTR_PER_VLD_LAM_ADJ[i] = 0
LV_LAM_ADJ_P_ACT[i] = 0
LV_LAM_ADJ_I_ACT[i] = 0
LV_LAM_ADJ_I_CAT_DIAG_ACT[i] = 0
Endif (1)

if LV_LAM_ADJ_CAT_DIAG[i] = 0 **or** LC_ENA_NORM_TRM_MAN_ACT = 1
then
T_DLY_LAM_ADJ[i] = (T_DLY_P_LAM_ADJ[i] + T_DLY_I_LAM_ADJ[i]) *
IP_FAC_T_DLY_LAM_ADJ (7.49.4.A)
else
T_DLY_LAM_ADJ[i] = (T_DLY_P_LAM_ADJ[i] + T_DLY_I_LAM_ADJ_CAT_DIAG[i]) *
IP_FAC_T_DLY_LAM_ADJ (7.49.4.B)
endif

if LV_DC 1 $\rightarrow$ 0
then
$T\_DLY\_I\_AD\_LAM\_ADJ[i]_{CDC} = T\_DLY\_I\_AD\_LAM\_ADJ[i]_{LDC} + (T\_DLY\_I\_MV\_LAM\_ADJ[i]_{k} - T\_DLY\_I\_AD\_LAM\_ADJ[i]_{LDC}) * C\_CRLC\_T\_DLY\_AD\_LAM\_ADJ$ (7.49.5.1)

$T\_DLY\_I\_AD\_LAM\_ADJ\_CAT\_DIAG[i]_{CDC} = T\_DLY\_I\_AD\_LAM\_ADJ\_CAT\_DIAG[i]_{LDC} + (T\_DLY\_I\_MV\_LAM\_ADJ\_CAT\_DIAG[i]_{k} - T\_DLY\_I\_AD\_LAM\_ADJ\_CAT\_DIAG[i]_{LDC}) * C\_CRLC\_T\_DLY\_AD\_CAT\_DIAG$ (7.49.5.2)

endif

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TEMP_CAT_MIN_LAM_ADJ_ACT	1	0...7FFFH	-273.15...1.77479 E+3	0.0625	°C
min. TEMP_CAT threshold for activation					
C_MAF_INT_LAM_ADJ_ACT_CAT_PURGE	1	0...FFFFH	0...1820.416667	0.027778	G
Threshold for MAF integral while restart of LAM after PUC					
C_MAF_INT_LAM_ADJ_ACT	1	0...FFFFH	0...1820.416667	0.027778	g
Threshold for MAF integral while restart of LAM after inactive					
C_EFF_CAT_DIAG_MAX_LAM_ADJ_ACT	1	0...FFH	0...1.992188	7.8125E-3	-
Threshold of catalyst conversion capability for activation					
C_MAF_MIN_LAM_ADJ_P_ACT	1	0...FFFFH	0...1389	0.021195	mg/stk
min MAF threshold for activation P-share					
C_MAF_MAX_LAM_ADJ_P_ACT	1	0...FFFFH	0...1389	0.021195	mg/stk
max MAF threshold for activation P-share					
C_N_MIN_LAM_ADJ_P_ACT	1	0...FFH	0...8160	32	rpm
min N threshold for activation P-share					
C_N_MAX_LAM_ADJ_P_ACT	1	0...FFH	0...8160	32	rpm
max N threshold for activation P-share					
C_T_DLY_I_MIN_LAM_ADJ	1	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Min limit of I-share					
C_T_DLY_I_MAX_LAM_ADJ	1	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Max limit of I-share					
C_T_DLY_I_MIN_CAT_DIAG	1	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Min limit of I-share in case of catalyst diagnosis					
C_T_DLY_I_MAX_CAT_DIAG	1	8000...7FFFH	-640...639.980469	1.9531E-2	ms
Max limit of I-share in case of catalyst diagnosis					
C_CRLC_T_DLY_I_MMV_LAM_ADJ	1	0...FFFFH	0...0.999985	1.5259E-5	-
correlation constant for I-share mean value calculation					
C_CRLC_T_DLY_I_MMV_CAT_DIAG	1	0...FFFFH	0...0.999985	1.5259E-5	-
correlation constant for I-share mean value calculation (in case of catalyst diagnosis)					
C_CRLC_T_DLY_AD_LAM_ADJ	1	0...FFFFH	0...0.999985	1.5259E-5	-
correlation constant for AD-share calculation					
C_CRLC_T_DLY_AD_CAT_DIAG	1	0...FFFFH	0...0.999985	1.5259E-5	-
correlation constant for AD-share calculation (catalyst diagnosis)					
IP_VLS_SP_LAM_ADJ[NC_CBK_EX_NR]	8*8	0...3FFH	0...4.995117	4.8828E-3	V
LDPM_N_32_3_LACO	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_LACO	8	0...FFH	0...1389	5.447059	mg/stk
setpoint for downstream fuel trim controller					
IP_T_DLY_P_LAM_ADJ[NC_CBK_EX_NR]	12	0...FFFFH	-640...639.980469	1.9531E-2	ms
LDPM_VLS_DIF_LAM_ADJ_1_LACO	12	0...07FFH	-5...4.995117	4.8828E-3	V
LAM - P - jump delay time from P-share					
IP_FAC_T_DLY_P_LAM_ADJ	6	0...FFFFH	0...3.999939	6.1035E-5	-
LDP_MAF_CYL_IP_FAC_P_LAM_ADJ	6	0...FFFFH	0...2047.96875	0.03125	kg/h
Weight factor for P-share regarding engine load and speed conditions					
IP_FAC_T_DLY_P_CAT_DIAG[NC_CBK_EX_NR]	8	0...FFFFH	0...0.999985	1.5259E-5	-
LDPM_EFF_CAT_DIAG_1_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
Weight factor for P-share regarding catalyst conversion capability					
IP_MAF_MIN_LAM_ADJ_I_ACT	6	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_2_LACO	6	0...FFH	0...8160	32	rpm
Min MAF threshold for activation I-share					
IP_MAF_MAX_LAM_ADJ_I_ACT	6	0...FFFFH	0...1389	0.021195	mg/stk
LDPM_N_32_2_LACO	6	0...FFH	0...8160	32	rpm
Max MAF threshold for activation I-share					

IP_T_DLY_I_LAM_ADJ[NC_CBK_EX_NR]	12*8	0...FFFFH	-640...639.980469	1.9531E-2	ms
LDPM_VLS_DIF_LAM_ADJ_1_LACO	12	0...07FFH	-5...4.995117	4.8828E-3	V
LDPM_EFF_CAT_DIAG_1_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
LAM - P - jump delay time from I-share					
IP_T_DLY_I_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	12*8	0...FFFFH	-640...639.980469	1.9531E-2	ms
LDPM_VLS_DIF_LAM_ADJ_2_LACO	12	0...07FFH	-5...4.995117	4.8828E-3	V
LDPM_EFF_CAT_DIAG_2_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
LAM - P - jump delay time from I-share - in case of activated catalyst diagnosis					
IP_T_DLY_P_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	12*8	0...FFFFH	-640...639.980469	1.9531E-2	ms
LDPM_VLS_DIF_LAM_ADJ_2_LACO	12	0...07FFH	-5...4.995117	4.8828E-3	V
LDPM_EFF_CAT_DIAG_2_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
LAM - P - jump delay time from P-share - in case of activated catalyst diagnosis					
IP_CRLC_T_DLY_I_LAM_ADJ	6	0...FFFFH	0...0.999985	1.5259E-5	-
LDP_MAF_CYL_IP_CRLC_I_LAM_ADJ	6	0...FFFFH	0...2047.96875	0.03125	kg/h
correlation constant for I-share calculation					
IP_FAC_T_DLY_LAM_ADJ	8*8	0...FFFFH	0...3.999939	6.1035E-5	-
LDPM_N_32_3_LACO	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_LACO	8	0...FFH	0...1389	5.447059	mg/stk
Normalisation factor for total LAM -P-jump delay time regarding engine load and speed conditions					
LC_ENA_NORM_TRM_MAN_ACT	1	0...1H	0...1	1	-
Switch to enable the trim controller functioning only in the normal mode					

7.50 Application Incidences for Downstream fuel trim

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
inhibition flag for trim control					
LV_INH_LAM_ADJ_I[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
inhibition flag for trim control I-Term					
LV_LDC_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
limited dynamic conditions for trim control					
VLS_AV_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...3FFH	0...4.995117	4.8828E-3	V
sensor voltage actual value for trim control					
VLS_AV_TMP_LAM_ADJ[NC_CBK_EX_NR]	V	0...FFFFH	0...4.999924	7.6294E-5	V
auxiliary sensor voltage actual value for filter calculation					
FAC_LAM_MV_OFS_LDC_LAM_ADJ[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
filtered lambda controller output offset for limited dynamics calculation					
N_OFS_LDC_LAM_ADJ[NC_CBK_EX_NR]	V	E020...1FE0H	-8160...8160	1	rpm
engine speed offset for limited dynamics calculation					
PV_AV_OFS_LDC_LAM_ADJ[NC_CBK_EX_NR]	V	80...7FH	-50...49.609375	0.390625	%
actual pedal value offset for limited dynamics calculation					
MAF_OFS_LDC_LAM_ADJ[NC_CBK_EX_NR]	V	8000...7FFFH	-694.510597... 694.489403	0.021195	mg/stk
air mass flow offset for limited dynamics calculation					
MAF_INT_LDC_LAM_ADJ[NC_CBK_EX_NR]	V	0...FFFFH	0...1820.416667	0.027778	g
MAF integral during limited dynamics not fulfilled					
VLS_DELTA_LAM_ADJ_CAT_DIAG	V/O	0...3FFH	0...4.995117	4.8828E-3	V
sensor voltage set point shift in case of active cat efficiency diagnosis					

Input data:

LV_ERR_CRK	LV_ERR_MAF	LV_ERR_CPS	LV_ERR_TPS
LV_ERR_CAM	LV_ERR_TCO	MAF_CYL	MAF_DELTA_LDC
N_DELTA_LDC	PV_AV_DELTA_LDC	NC_CBK_EX_NR	LV_MIS_STATE_A
LV_ERR_IGC	LV_ERR_IV[NC_CYL_NR]	LV_ERR_MEC_IVVT_IN_i	LV_ERR_SLV_IVVT_IN_i
LV_ERR_CAT_DIAG	LV_ERR_RATIO_CHK	LV_MIS_STATE_B	
LV_ERR_FSD[NC_CBK_EX_NR]		LV_ERR_LSH_UP[NC_CBK_EX_NR]	
LV_ERR_LSH_DOWN[NC_CBK_EX_NR]		LV_ERR_EL_LS_UP[NC_CBK_EX_NR]	
LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]		LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]	
LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]		LV_ERR_LAM_LIM[NC_CBK_EX_NR]	
VLS_DOWN[NC_CBK_EX_NR]		FAC_LAM_MV_DELTA_LDC[NC_CBK_EX_NR]	
LV_ERR_MEC_OPEN_CPS		LV_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	
EFF_CAT_DIAG[NC_CBK_EX_NR]		LV_ERR_LS_DOWN[NC_CBK_EX_NR]	
T_SUM_AFL_AFR_CYC_LAM[NC_CBK_EX_NR]		LV_SCC[NC_CBK_EX_NR]	
LV_ERR_MAP			

FUNCTION DESCRIPTION:

General information:

There are many errors, which make it necessary to stop the trim control when they occur.

In case of cylinder shut-off (for example Traction Control torque reduction request), Trim controller I-term is inhibited via LV_INH_LAM_ADJ_I until VLS_DOWN indicates again a Rich mixture (VLS_DOWN > thd.).

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank

Application conditions:

Initialisation: at ECU reset:

LV_INH_LAM_ADJ[i]	=	1
LV_INH_LAM_ADJ_I[i]	=	0
LV_LDC_LAM_ADJ[i]	=	0
N_OFS_LDC_LAM_ADJ[i]	=	0
MAF_OFS_LDC_LAM_ADJ[i]	=	0
PV_AV_OFS_LDC_LAM_ADJ[i]	=	0
FAC_LAM_MV_OFS_LDC_LAM_ADJ[i]	=	0
MAF_INT_LDC_LAM_ADJ[i]	=	0
VLS_AV_LAM_ADJ[i]	=	0
VLS_AV_TMP_LAM_ADJ	=	0

Activation:

- at every engine operating state

Deactivation:

-

Recurrence: T_SAMPLE = 100 ms

Formula section:

Calculation of the inhibition flag for trim control:

if LV_ERR_CPS = 1
(canister purge system error active)
or LV_ERR_MEC_OPEN_CPS = 1
(canister purge system mechanical error active)
or LV_MIS_STATE_A = 1
(misfire state A active)
or LV_ERR_FSD[i] = 1 and LC_ENA_LAM_ADJ_ERR_FSD = 0)
(fuel system diagnosis error active)
or LV_ERR_MAF = 1
(mass air flow sensor error active)

or LV_ERR_MAP = 1
(manifold absolute pressure sensor error active)
or LV_ERR_TPS = 1
(throttle position sensor error active)
or LV_ERR_CRK = 1
(crankshaft sensor error active)
or LV_ERR_CAM = 1
(camshaft sensor error active)
or LV_ERR_TCO = 1
(coolant temperature sensor error active)
or LV_ERR_LSH_UP[i] = 1
(lambda sensor heater upstream error active)
or LV_ERR_LSH_DOWN[i] = 1
(lambda sensor heater downstream error active)
or LV_ERR_EL_LS_UP[i] = 1
(lambda sensor signal upstream error active)
or LV_ERR_EL_LS_DOWN[i] = 1
(lambda sensor signal downstream error active)
or LV_ERR_LS_DOWN[i] = 1
(lambda sensor signal downstream error - confirmed diagnostic)
or LV_ERR_FRQ_LS_UP[i] = 1 and LC_ENA_LAM_ADJ_ERR_FRQ_LS_UP = 0)
(lambda sensor signal frequency error active)
or LV_ERR_SWT_LS_UP[i] = 1
(lambda sensor signal switching time error active)
or LV_ERR_LAM_LIM[i] = 1
(lambda controller output signal is limited)
or LV_ERR_IGC = 1
(ignition coil error active)
or LV_ERR_IV[NC_CYL_NR] = 1
(injection valve error active)
or LV_ERR_MEC_IVVT_IN_i = 1
(crankshaft to inlet camshaft mechanics violation detected)
or LV_ERR_SLV_IVVT_IN_i = 1
(IVVT inlet solenoid valve error avtive)
or LV_ERR_CAT_DIAG = 1 and LC_ENA_LAM_ADJ_ERR_CAT_DIAG = 0)
(catalyst diagnosis error active)
or LV_ERR_RATIO_CHK = 1
(inconsistencies between actual load and throttle position detected)
or LV_MIS_STATE_B = 1
(CARB B misfire detected)

then
LV_INH_LAM_ADJ[i] = 1
else
LV_INH_LAM_ADJ[i] = 0
endif

Calculation of the inhibition flag for trim control I-term:

```
if    LV_SCC[i] = 1
then          LV_INH_LAM_ADJ_I[i] = 1
else
      if      VLS_DOWN[i] > C_VLS_DOWN_INH_LAM_ADJ_I_SCC
      then    LV_INH_LAM_ADJ_I[i] = 0
      endif
  endif
```

Calculation of the actual value of downstream oxygen sensor voltage:

VLS_DELTA_LAM_ADJ_CAT_DIAG = IP_VLS_DELTA_LAM_ADJ_CAT_DIAG

```
if
(LV_LAM_ADJ_CAT_DIAG[i] = 0 and LC_VLS_DOWN_FIL_ACT = 0)
then
      VLS_AV_TMP_LAM_ADJ[i] = VLS_DOWN[i]
      VLS_AV_LAM_ADJ[i] = VLS_AV_TMP_LAM_ADJ[i]
else
      VLS_AV_TMP_LAM_ADJ[i] = VLS_AV_TMP_LAM_ADJ[i] +
                              (VLS_DOWN[i] - VLS_AV_TMP_LAM_ADJ[i] +
                                     VLS_DELTA_LAM_ADJ_CAT_DIAG)
                                  * IP_CRLC_VLS_LAM_ADJ_CAT_DIAG

      VLS_AV_LAM_ADJ[i] = VLS_AV_TMP_LAM_ADJ[i]

endif
```

SW hint: Definitions of "VLS" variables are not the same!

Detection of limited dynamic conditions for trim control:

```
if    | N_DELTA_LDC | > C_N_DYW_LDC_LAM_ADJ
then
      if    |N_DELTA_LDC - N_OFS_LDC_LAM_ADJ[i]| > C_N_DYW_LDC_LAM_ADJ
      then
            N_OFS_LDC_LAM_ADJ[i] = N_DELTA_LDC
            MAF_INT_LDC_LAM_ADJ[i] = 0
      endif
else
      N_OFS_LDC_LAM_ADJ[i] = 0
endif

if    | MAF_DELTA_LDC | > C_MAF_DYW_LDC_LAM_ADJ
then
      if    |MAF_DELTA_LDC - MAF_OFS_LDC_LAM_ADJ[i]| >
                                                  C_MAF_DYW_LDC_LAM_ADJ
      then
         MAF_OFS_LDC_LAM_ADJ[i] = MAF_DELTA_LDC
         MAF_INT_LDC_LAM_ADJ[i] = 0
      endif
else
      MAF_OFS_LDC_LAM_ADJ[i] = 0
endif

if    | PV_AV_DELTA_LDC | > C_PV_AV_DYW_LDC_LAM_ADJ
then
      if  |PV_AV_DELTA_LDC - PV_AV_OFS_LDC_LAM_ADJ[i]| >
                                                  C_PV_AV_DYW_LDC_LAM_ADJ
      then
            PV_AV_OFS_LDC_LAM_ADJ[i] = PV_AV_DELTA_LDC
            MAF_INT_LDC_LAM_ADJ[i] = 0
      endif
else
      PV_AV_OFS_LDC_LAM_ADJ[i] = 0
endif
```

if | FAC_LAM_MV_DELTA_LDC[i] | > C_FAC_LAM_MV_DYW_LDC_LAM_ADJ

then

if |FAC_LAM_MV_DELTA_LDC[i] - FAC_LAM_MV_OFS_LDC_LAM_ADJ[i]| > C_FAC_LAM_MV_DYW_LDC_LAM_ADJ

then

FAC_LAM_MV_OFS_LDC_LAM_ADJ[i] = FAC_LAM_MV_DELTA_LDC[i]

MAF_INT_LDC_LAM_ADJ[i] = 0

endif

else

FAC_LAM_MV_OFS_LDC_LAM_ADJ[i] = 0

endif

if MAF_INT_LDC_LAM_ADJ[i] < C_MAF_INT_LDC_LAM_ADJ

then

$MAF\_INT\_LDC\_LAM\_ADJ[i]_n$ [g] = $MAF\_INT\_LDC\_LAM\_ADJ[i]_{n-1}$ [g]+ MAF_CYL [kg/h]*T_A [ms]*1/3600 [(g*h)/(kg*ms)]

LV_LDC_LAM_ADJ[i] = 0

else

LV_LDC_LAM_ADJ[i] = 1

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DYW_LDC_LAM_ADJ	1	0...1FE0H	0...8160	1	rpm
engine speed window for limited dynamic conditions trim control					
C_MAF_DYW_LDC_LAM_ADJ	1	0...FFFFH	0...1389	0.021195	mg/stk
air mass flow window for limited dynamic conditions trim control					
C_FAC_LAM_MV_DYW_LDC_LAM_ADJ	1	0...7FFFH	0...49.998474	1.5259E-3	%
lambda control output window for limited dynamic conditions trim control					
C_PV_AV_DYW_LDC_LAM_ADJ	1	0...FFH	0...99.609375	0.390625	%
pedal value window for limited dynamic conditions trim control					
C_MAF_INT_LDC_LAM_ADJ	1	0...FFFFH	0...1820.416667	0.027778	g
air mass flow integral for duration of violation of limited dynamic conditions trim control					
LC_VLS_DOWN_FIL_ACT	1	0...1H	0...1	1	-
Switch to manually activate VLS_DOWN filter					
IP_VLS_DELTA_LAM_ADJ_CAT_DIAG	8	0...3FFH	0...4.995117	4.8828E-3	V
LDPM_EFF_CAT_DIAG_1_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
sensor voltage set point shift in case of active cat efficiency diagnosis					
IP_CRLC_VLS_LAM_ADJ_CAT_DIAG	6*8	0...FFH	0...0.996094	3.9063E-3	-
LDP_T_SUM_AFL_AFR_IP_CRLC_VLS	6	0...FFFFH	0...655.35	0.01	s
LDPM_EFF_CAT_DIAG_1_LACO	8	0...FFH	0...1.992188	7.8125E-3	-
correlation constant for filtering downstream signal for catalyst efficiency diagnosis					
C_VLS_DOWN_INH_LAM_ADJ_I_SCC	1	0...3FFH	0....4.995	0.0049	V
Minimum Downstream O2 sensor voltage to enable again dynamic Trim I-term after Cylinder Shut-off					
LC_ENA_LAM_ADJ_ERR_FSD	1	0...1H	0...1	1	[-]
Switch to manually activate fuel trim with FSD error					
LC_ENA_LAM_ADJ_ERR FRQ_LS_UP	1	0...1H	0...1	1	[-]
Switch to manually activate fuel trim with FRQ_LS_UP error					
LC_ENA_LAM_ADJ_ERR CAT_DIAG	1	0...1H	0...1	1	[-]
Switch to manually activate fuel trim with CAT_DIAG error					

7.51 Lambda adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_LAM_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	-694.510597...694.489403	0.021195	mg/stk
fuel mass set point offset, output from lambda adaptation					
FAC_LAM_AD_LAM_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
fuel mass set point factor, output from lambda adaptation					
FAC_MFF_ADD_LAM_AD_OUT[NC_CBK_EX_NR]	V/O	8000...7FFFH	-100...99.996948	3.0518E-3	%
relative lambda adaptation offset quotient					
MFF_ADD_LAM_AD[NC_CBK_EX_NR]	V/S	8000...7FFFH	-694.510597...694.489403	0.021195	mg/stk
fuel mass set point offset, stored value of lambda adaptation					
MFF_DELTA_ADD_LAM_AD[NC_CBK_EX_NR]	V	8000...7FFFH	-694.510597...694.489403	0.021195	mg/stk
difference of lambda adaptation offset to previous adaptation					
MFF_DELTA_ADD_LAM_AD_H_RES[NC_CBK_EX_NR]	V	8000...7FFFH	-10.851728...10.851397	3.3117E-4	mg/stk
difference integral with high resolution of lambda adaptation offset					
FAC_LAM_AD[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
fuel mass set point factor of lambda adaptation, interpolation of both factor areas (high and low field)					
FAC_L_RNG_LAM_AD[NC_CBK_EX_NR]	O/V/S	8000...7FFFH	-50...49.998474	1.5259E-3	%
fuel mass set point factor of low field, stored value of lambda adaptation					
FAC_H_RNG_LAM_AD[NC_CBK_EX_NR]	O/V/S	8000...7FFFH	-50...49.998474	1.5259E-3	%
fuel mass set point factor of high field, stored value of lambda adaptation					
FAC_DELTA_L_RNG_LAM_AD[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
difference of lambda adaptation factor of low field to previous adaptation					
FAC_DELTA_H_RNG_LAM_AD[NC_CBK_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	%
difference of lambda adaptation factor of high field to previous adaptation					
FAC_LAM_ADJ_LAM_AD[NC_CBK_EX_NR]	V/O	8000...7FFFH	-50...49.998474	1.5259E-3	%
output value from lambda adaptation for the lambda controller shift					
LV_FAC_LAM_ADJ_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag to request lambda controller shift					
LV_LAM_AD_STOP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
logical variable indicating stop of the lambda adaptation cycle					
LV_LAM_AD_STOP_CBK_EX	V	0...1H	0...1	1	-
logical variable indicating stop of at least one bank of the lambda adaptation cycle					
LV_LAM_AD_END	V/O	0...1H	0...1	1	-
logical value indicating temporary end of lambda adaptation					
T_PRI_LAM_AD[NC_CBK_EX_NR]	V	0...FFFFH	0...6553.5	0.1	s
priority time for next requested lambda adaptation					
T_PRI_TOT_LAM_AD	V/O	0...FFFFH	0...6553.5	0.1	s
minimum priority time of all exhaust cylinder banks for next requested lambda adaptation					
LV_LAM_AD_AFS_REQ	V/O	0...1H	0...1	1	-
request flag for combustion manager to force lambda eq. 1 conditions for lambda adaptation					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LAM_AD_CDN	V/O	0...1H	0...1	1	-
flag for time scheduler indicating good conditions for lambda adaptation					
FAC_MFF_ADD_FAC_LAM_AD[NC_CBK_EX_NR]	V/O/S	8000...7FFFH	-100...99.996948	3.0518E-3	%
lambda adaptation correction for scan tool (factor and relative offset)					
CTR_RAF_CHG_LAM_AD[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
counter indicating the number lean rich changes (P jumps)					
LV_FAC_L_RNG_LIM_MAX_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating top limitation of lambda adaptation factor of lower area					
LV_FAC_L_RNG_LIM_MIN_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating bottom limitation of lambda adaptation factor of lower area					
LV_FAC_H_RNG_LIM_MAX_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating top limitation of lambda adaptation factor of upper area					
LV_FAC_H_RNG_LIM_MIN_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating bottom limitation of lambda adaptation factor of upper area					
LV_MFF_ADD_LIM_MAX_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating top limitation of lambda adaptation offset					
LV_MFF_ADD_LIM_MIN_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating bottom limitation of lambda adaptation offset					
LV_FAC_H_RNG_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating that lambda adaptation in factor learning upper field is active					
LV_FAC_L_RNG_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating that lambda adaptation in factor learning lower field is active					
LV_MFF_ADD_RNG_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating that lambda adaptation in offset learning field is active					
STATE_LAM_AD[NC_CBK_EX_NR]	O/V	0H 1H 2H 3H 4H 5H 6H 7H	INIT WAIT CDN_FAC_L CDN_FAC_H CDN_ADD ADAPT_FAC_L ADAPT_FAC_H ADAPT_ADD	1	-
state of lambda adaptation					
MFF_SEL_LAM_AD	V	0…FFFFH	0…1389	0.02119	Mg/stk
Selected fuel mass flow for lambda adaptation activation					

Input data:

TCO	N_32
FAC_LAM_MV_MMV[NC_CBK_EX_NR]	CRLC_LAM_AD[NC_CBK_EX_NR]
MFF_SP[NC_CBK_EX_NR]	LV_LAM_AD_ACT[NC_CBK_EX_NR]
LV_LDC_LAM_AD[NC_CBK_EX_NR]	NR_RAF_CHG_LAM_AD[NC_CBK_EX_NR]
MFF_BAS	LV_LAM_LSCL[NC_CBK_EX_NR]
LV_ST_END	LV_LAM_AD_ENA
CTR_RAF_CHG[NC_CBK_EX_NR]	LV_LAM_STOP[NC_CBK_EX_NR]
LV_FAC_LAM_ADJ_LAM_AD_END[NC_CBK_EX_NR]	LV_LAM_AD_DEAC_ERR[NC_CBK_EX_NR]
C_TCO_MIN_LAM_AD	LV_IGK
NC_CBK_EX_NR	LV_INH_CTR_RAF_CHG[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

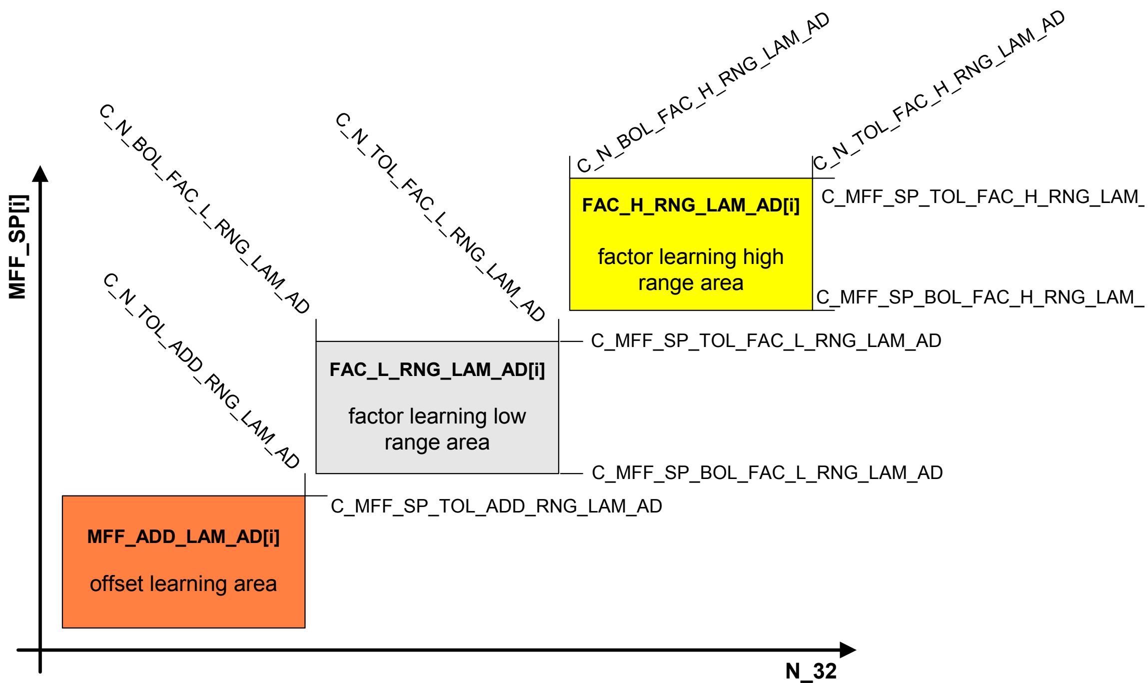
In order to compensate serial production tolerances of components, an adaptive (additive and two multiplicative) correction is calculated versus the averaged output of the lambda controller.
The additive and multiplicative adaptation corrections are used for calculating the injection time for all engine operating states, except at 'engine stop' and 'engine start'

The calculation shall be done for all exhaust cylinder banks.

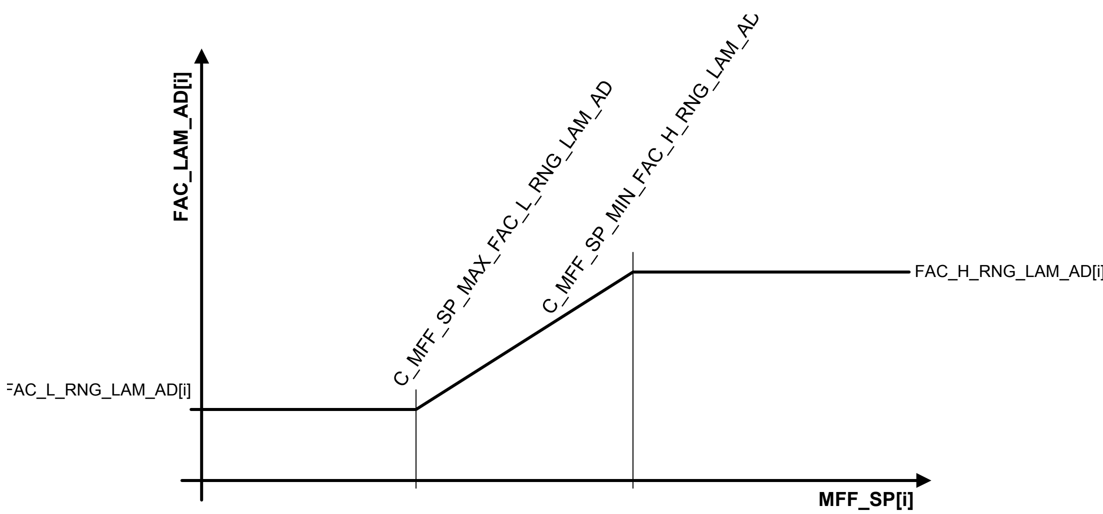
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2,

otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank.

Adaptation field diagram:



Interpolation diagram:



Description:

Lambda adaptation, for precision reason, needs to be performed at lambda equal 1 conditions. The function itself is activated by the corresponding LV_LAM_AD_ACT[i]. Depending on MFF_SEL_LAM_AD and N_32 three different adaptation fields (one offset and two factor areas) are observed. MFF_SEL_LAM_AD is determined as below,

If LC_SEL_MFF_SP_LAM_AD =1

Then MFF_SEL_LAM_AD = MFF_SP[i]

Else MFF_SEL_LAM_AD = MFF_BAS

EndIf

According to project philosophy homogeneous mode and suitable conditions can be forced by setting:

- LC_MFF_ADD_RNG_AFS_REQ_LAM_AD,
- LC_FAC_L_RNG_AFS_REQ_LAM_AD and / or
- LC_FAC_H_RNG_AFS_REQ_LAM_AD.

to 1. Then LV_LAM_AD_AFS_REQ is set to 1 in order to request lambda eq. 1 condition from the combustion manager.

Application conditions:

Initialisation: if ECU is brand new or at EEPROM error:
FAC_L_RNG_LAM_AD[i] = 0
FAC_H_RNG_LAM_AD[i] = 0
MFF_ADD_LAM_AD[i] = 0
FAC_MFF_ADD_FAC_LAM_AD[i] = 0

at ECU reset:
LV_LAM_AD_END = 0
LV_FAC_LAM_ADJ_LAM_AD[i] = 0
FAC_LAM_ADJ_LAM_AD[i] = 0 %
LV_LAM_AD_STOP[i] = 0
LV_LAM_AD_STOP_CBK_EX = 0
STATE_LAM_AD[i] = "INIT"
MFF_DELTA_ADD_LAM_AD[i] = 694.489403 mg/stk
MFF_DELTA_ADD_LAM_AD_H_RES[i] = 0 mg/stk
FAC_DELTA_L_RNG_LAM_AD[i] = 49.998474 %
FAC_DELTA_H_RNG_LAM_AD[i] = 49.998474 %
T_PRI_LAM_AD[i] = 6553.5 s
T_PRI_TOT_LAM_AD = 6553.5 s
LV_MFF_ADD_LIM_MIN_LAM_AD[i] = 0
LV_MFF_ADD_LIM_MAX_LAM_AD[i] = 0
LV_FAC_L_RNG_LIM_MIN_LAM_AD[i] = 0
LV_FAC_L_RNG_LIM_MAX_LAM_AD[i] = 0
LV_FAC_H_RNG_LIM_MIN_LAM_AD[i] = 0
LV_FAC_H_RNG_LIM_MAX_LAM_AD[i] = 0

Recurrence: T_SAMPLE = 20 ms

Activation: in all engine operating states

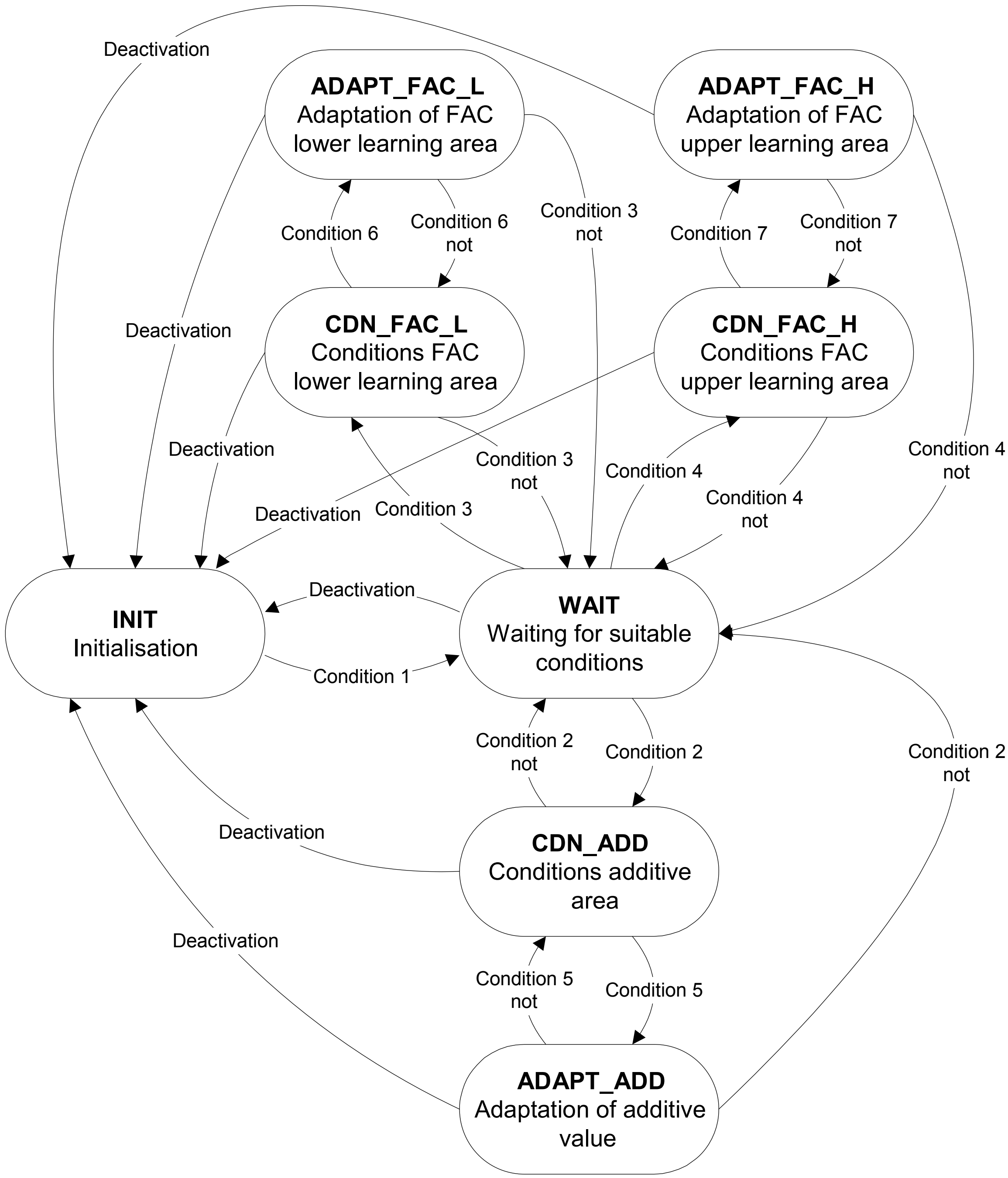
Deactivation: -

7.51.1 Sequence for lambda adaptation determination (state machine)

The state machine shall remain in its current state and carry out the actions specified to occur within that state once per recurrence unless otherwise specified. The state machine shall only move to another state when one of the conditions has been determined to be met.

The priority of the conditions to change between states shall be defined by the order in which these conditions are listed within the appropriate state as described below.

State machine diagram:



7.51.1.1 State "INIT"

Description:

After every engine start at every new activation the sequence starts with the INIT and switches to the state "WAIT" when activation conditions are fulfilled.

Formula section:

Actions:

LV_LAM_AD_END = 0

LV_LAM_AD_CDN = 0

LV_MFF_ADD_RNG_LAM_AD[i] = 0

LV_FAC_L_RNG_LAM_AD[i] = 0

LV_FAC_H_RNG_LAM_AD[i] = 0

LV_LAM_AD_AFS_REQ = 0

Condition 1: Transition to state "WAIT"

IF (LV_LAM_AD_ACT[i] = 1)
THEN
  STATE_LAM_AD[i] = "WAIT"
ELSE
  STATE_LAM_AD[i] = "INIT"
ENDIF

7.51.1.2 State "WAIT"

Description:

The state machine remains in this state as long as the conditions for a lambda adaptation are not fulfilled or not fulfilled anymore. T_PRI_LAM_AD[i] is used by the generic time control manager.

Formula section:

Actions:

T_PRI_LAM_AD[i] = **MAX**(ID_T_PRI_MFF_ADD_RNG_LAM_AD;
ID_T_PRI_FAC_L_RNG_LAM_AD;
ID_T_PRI_FAC_H_RNG_LAM_AD)

LV_LAM_AD_END = 0

LV_LAM_AD_CDN = 0

LV_MFF_ADD_RNG_LAM_AD[i] = 0

LV_FAC_L_RNG_LAM_AD[i] = 0

LV_FAC_H_RNG_LAM_AD[i] = 0

LV_LAM_AD_AFS_REQ = 0

Deactivation: Transition to state "INIT"

IF (LV_LAM_AD_ACT[i] = 0)
THEN
STATE_LAM_AD[i] = "INIT"
ENDIF

Condition 2: Transition to state "CDN_ADD"

IF (N_32 < C_N_TOL_ADD_RNG_LAM_AD) **AND**
(MFF_SEL_LAM_AD **<** C_MFF_TOL_ADD_RNG_LAM_AD)
THEN
STATE_LAM_AD[i] = "CDN_ADD"
ENDIF

Condition 3: Transition to state "CDN_FAC_L"

IF (N_32 > C_N_BOL_FAC_L_RNG_LAM_AD) **AND**
(N_32 < C_N_TOL_FAC_L_RNG_LAM_AD) **AND**
(MFF_SEL_LAM_AD > ID_MFF_BOL_FAC_L_RNG_LAM_AD) **AND**
(MFF_SEL_LAM_AD < ID_MFF_TOL_FAC_L_RNG_LAM_AD)
THEN
STATE_LAM_AD[i] = "CDN_FAC_L"
ENDIF

Condition 4: Transition to state "CDN_FAC_H"

IF (N_32 > C_N_BOL_FAC_H_RNG_LAM_AD) **AND**
(N_32 < C_N_TOL_FAC_H_RNG_LAM_AD) **AND**
(MFF_SEL_LAM_AD > ID_MFF_BOL_FAC_H_RNG_LAM_AD) **AND**
(MFF_SEL_LAM_AD < ID_MFF_TOL_FAC_H_RNG_LAM_AD)
THEN
STATE_LAM_AD[i] = "CDN_FAC_H"
ENDIF

7.51.1.3 State "CDN_ADD"

Description:

The flag LV_LAM_AD_CDN is set to indicate possible adaptation conditions for use at time scheduler (between CP and lambda adaptation). The time scheduler enables the lambda adaptation by setting LV_LAM_AD_ENA.

Formula section:

Actions:

T_PRI_LAM_AD[i] = ID_T_PRI_MFF_ADD_RNG_LAM_AD

LV_LAM_AD_CDN = 1

LV_MFF_ADD_RNG_LAM_AD[i] = 0

LV_FAC_L_RNG_LAM_AD[i] = 0

LV_FAC_H_RNG_LAM_AD[i] = 0

LV_LAM_AD_END = 0

```
IF (LV_LAM_AD_ENA = 1) AND          /* canister purge function interaction */   (LC_MFF_ADD_RNG_AFS
THEN
   LV_LAM_AD_AFS_REQ = 1            /* lambda eq. 1 required */
ELSE
   LV_LAM_AD_AFS_REQ = 0
ENDIF
```

Deactivation: Transition to state "INIT"

```
IF (LV_LAM_AD_ACT[i] = 0)
THEN
   STATE_LAM_AD[i] = "INIT"
ENDIF
```

Condition 2 NOT: Transition to state "WAIT"

```
IF (N_32 > C_N_TOL_ADD_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD > C_MFF_TOL_ADD_RNG_LAM_AD)
THEN
   STATE_LAM_AD[i] = "WAIT"
ENDIF
```

Condition 5: Transition to state "ADAPT_ADD"

```
IF (LV_LAM_AD_ENA = 1) AND          /* canister purge function interaction */
   (LV_LAM_LSCL[i] = 1) AND         /* lambda control active */
   (LV_LAM_STOP[i] = 0) AND         /* lambda control not in stop mode */
   (LV_LDC_LAM_AD[i] = 1)           /* limited dynamics fulfilled */
THEN
   STATE_LAM_AD[i] = "ADAPT_ADD"
ENDIF
```

Actions during transition: CTR_RAF_CHG_LAM_AD[i] = 0

7.51.1.4 State "CDN_FAC_L"

Description:

The flag LV_LAM_AD_CDN is set to indicate possible adaptation conditions for use at time scheduler (between CP and lambda adaptation). The time scheduler enables the lambda adaptation run by setting LV_LAM_AD_ENA.

Formula section:

Actions:

T_PRI_LAM_AD[i] = ID_T_PRI_FAC_L_RNG_LAM_AD

LV_LAM_AD_CDN = 1

LV_MFF_ADD_RNG_LAM_AD[i] = 0

LV_FAC_L_RNG_LAM_AD[i] = 0

LV_FAC_H_RNG_LAM_AD[i] = 0

LV_LAM_AD_END = 0

```
IF (LV_LAM_AD_ENA = 1) AND                  /* canister purge function interaction */
   (LC_FAC_L_RNG_AFS_REQ_LAM_AD = 1)        /* lambda eq. 1 shall be forced */
THEN
   LV_LAM_AD_AFS_REQ = 1                    /* lambda eq. 1 required */
ELSE
   LV_LAM_AD_AFS_REQ = 0
ENDIF
```

Deactivation: Transition to state "INIT"

```
IF (LV_LAM_AD_ACT[i] = 0)
THEN
   STATE_LAM_AD[i] = "INIT"
ENDIF
```

Condition 3 NOT: Transition to state "WAIT"

```
If (N_32 < C_N_BOL_FAC_L_RNG_LAM_AD) OR
   (N_32 > C_N_TOL_FAC_L_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD < ID_MFF_BOL_FAC_L_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD > ID_MFF_TOL_FAC_L_RNG_LAM_AD)
THEN
   STATE_LAM_AD[i] = "WAIT"
ENDIF
```

Condition 6: Transition to state "ADAPT_FAC_L"

```
IF (LV_LAM_AD_ENA = 1) AND                 /* canister purge function interaction */
   (LV_LAM_LSCL[i] = 1) AND                /* lambda control active */
   (LV_LAM_STOP[i] = 0) AND                /* lambda control not in stop mode */
   (LV_LDC_LAM_AD[i] = 1)                  /* limited dynamics fulfilled */
THEN
   STATE_LAM_AD[i] = "ADAPT_FAC_L"
ENDIF
```

Actions during transition: CTR_RAF_CHG_LAM_AD[i] = 0

7.51.1.5 State "CDN_FAC_H"

Description:

The flag LV_LAM_AD_CDN is set to indicate possible adaptation conditions for use at time scheduler (between CP and lambda adaptation). The time scheduler enables the lambda adaptation run by setting LV_LAM_AD_ENA.

Formula section:

Actions:

T_PRI_LAM_AD[i] = ID_T_PRI_FAC_H_RNG_LAM_AD

LV_LAM_AD_CDN = 1

LV_MFF_ADD_RNG_LAM_AD[i] = 0

LV_FAC_L_RNG_LAM_AD[i] = 0

LV_FAC_H_RNG_LAM_AD[i] = 0

LV_LAM_AD_END = 0

```
IF (LV_LAM_AD_ENA = 1) AND                 /* canister purge function interaction */
   (LC_FAC_H_RNG_AFS_REQ_LAM_AD = 1)       /* lambda eq. 1 shall be forced /
THEN
   LV_LAM_AD_AFS_REQ = 1                   /* lambda eq. 1 required */
ELSE
   LV_LAM_AD_AFS_REQ = 0
ENDIF
```

Deactivation: Transition to state "INIT"

```
IF (LV_LAM_AD_ACT[i] = 0)
THEN
   STATE_LAM_AD[i] = "INIT"
ENDIF
```

Condition 4 NOT: Transition to state "WAIT"

IF (N_32 < C_N_BOL_FAC_H_RNG_LAM_AD) **OR**
(N_32 > C_N_TOL_FAC_H_RNG_LAM_AD) **OR**
(MFF_SEL_LAM_AD < ID_MFF_BOL_FAC_H_RNG_LAM_AD) **OR**
(MFF_SEL_LAM_AD > ID_MFF_TOL_FAC_H_RNG_LAM_AD)
THEN
STATE_LAM_AD[i] = "WAIT"
ENDIF

Condition 7: Transition to state "ADAPT_FAC_H"

IF (LV_LAM_AD_ENA = 1) **AND** /* *canister purge function interaction* */
(LV_LAM_LSCL[i] = 1) **AND** /* *lambda control active* */
(LV_LAM_STOP[i] = 0) **AND** /* *lambda control not in stop mode* */
(LV_LDC_LAM_AD[i] = 1) /* *limited dynamics fulfilled* */
THEN
STATE_LAM_AD[i] = "ADAPT_FAC_H"
ENDIF

Actions during transition: CTR_RAF_CHG_LAM_AD[i] = 0

7.51.1.6 State "ADAPT_ADD"

Description:

Once the adaptation conditions are active, an adaptation takes place after NR_RAF_CHG_LAM_AD[i] lean ↔ rich air fuel ratio changes (by consideration of the value of CTR_RAF_CHG[i] signal). Another adaptation occurs after another NR_RAF_CHG_LAM_AD[i] lean ↔ rich air fuel ratio changes. In order to observe the priority of further lambda adaptations, the MMV value of the adapted factors and their respective differences to the further calculation are determined. In order to avoid big injection time deviation and to detect system failure the lambda adaptation values are limited.

Formula section:

Actions:

```
LV_MFF_ADD_RNG_LAM_AD[i] = 1

IF (LV_LAM_AD_STOP[i] = 0) AND
   (LV_INH_CTR_RAF_CHG[i] OR (CTR_RAF_CHG[i]n ≠ CTR_RAF_CHG[i]n-1))
THEN
   CTR_RAF_CHG_LAM_AD[i] = CTR_RAF_CHG_LAM_AD[i] + 1
   IF (CTR_RAF_CHG_LAM_AD[i] >= NR_RAF_CHG_LAM_AD[i])
   THEN
      FAC_LAM_ADJ_LAM_AD[i] = CRLC_LAM_AD[i] * FAC_LAM_MV_MMV[i]
      MFF_DELTA_ADD_LAM_AD_H_RES[i] = MFF_DELTA_ADD_LAM_AD_H_RES[i] +
                                          FAC_LAM_ADJ_LAM_AD[i] * MFF_SP[i]
      MFF_DELTA_ADD_LAM_AD[i] = MFF_DELTA_ADD_LAM_AD_H_RES[i] [1]
      IF (MFF_DELTA_ADD_LAM_AD[i] ≠ 0
      THEN
         T_PRI_LAM_AD[i] = ID_T_PRI_MFF_ADD_RNG_LAM_AD
         MFF_ADD_LAM_AD[i] = MFF_ADD_LAM_AD[i] + MFF_DELTA_ADD_LAM_AD[i]
         IF (MFF_ADD_LAM_AD[i] < C_MFF_MIN_ADD_RNG_LAM_AD)
         THEN
            MFF_ADD_LAM_AD[i] = C_MFF_MIN_ADD_RNG_LAM_AD
            LV_MFF_ADD_LIM_MIN_LAM_AD[i] = 1
         ELSE
            LV_MFF_ADD_LIM_MIN_LAM_AD[i] = 0
            IF (MFF_ADD_LAM_AD[i] > C_MFF_MAX_ADD_RNG_LAM_AD)
            THEN
               MFF_ADD_LAM_AD[i] = C_MFF_MAX_ADD_RNG_LAM_AD
               LV_MFF_ADD_LIM_MAX_LAM_AD[i] = 1
            ELSE
               LV_MFF_ADD_LIM_MAX_LAM_AD[i] = 0
            ENDIF
         ENDIF
         MFF_DELTA_ADD_LAM_AD_H_RES[i] = 0
         LV_LAM_AD_STOP[i] = 1
      ENDIF
   ENDIF
ENDIF
```

Deactivation: Transition to state "INIT"

```
IF (LV_LAM_AD_ACT[i] = 0)
THEN
   STATE_LAM_AD[i] = "INIT"
ENDIF
```

Condition 2 NOT: Transition to state "WAIT"

```
IF (N_32 > C_N_TOL_ADD_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD > C_MFF_TOL_ADD_RNG_LAM_AD)
THEN
   STATE_LAM_AD[i] = "WAIT"
ENDIF
```


[1] *SW hint: do not use bit shift operation; this would lead to wrong result for negative values*

Condition 5 NOT: Transition to state "CDN_ADD"

IF (LV_LAM_AD_ENA = 0) **OR**
(LV_LAM_LSCL[i] = 0) **OR**
(LV_LAM_STOP[i] = 1) **OR**
(LV_LDC_LAM_AD[i] = 0)
THEN
STATE_LAM_AD[i] = "CDN_ADD"
ENDIF

7.51.1.7 State "ADAPT_FAC_L"

Description:

Analogous procedure as described for state "ADAPT_ADD".

Formula section:

Actions:

LV_FAC_L_RNG_LAM_AD[i] = 1

```
IF (LV_LAM_AD_STOP[i] = 0) AND
   (LV_INH_CTR_RAF_CHG[i] OR (CTR_RAF_CHG[i]n ≠ CTR_RAF_CHG[i]n-1))
THEN
   CTR_RAF_CHG_LAM_AD[i] = CTR_RAF_CHG_LAM_AD[i] + 1
   IF (CTR_RAF_CHG_LAM_AD[i] >= NR_RAF_CHG_LAM_AD[i])
   THEN
      FAC_LAM_ADJ_LAM_AD[i] = CRLC_LAM_AD[i] * FAC_LAM_MV_MMV[i]
      FAC_DELTA_L_RNG_LAM_AD[i] = FAC_LAM_ADJ_LAM_AD[i]
      T_PRI_LAM_AD[i] = ID_T_PRI_FAC_L_RNG_LAM_AD
      FAC_L_RNG_LAM_AD[i] = FAC_L_RNG_LAM_AD[i] + FAC_DELTA_L_RNG_LAM_AD[i]
      IF (FAC_L_RNG_LAM_AD[i] < C_FAC_MIN_L_RNG_LAM_AD)
      THEN
         FAC_L_RNG_LAM_AD[i] = C_FAC_MIN_L_RNG_LAM_AD
         LV_FAC_L_RNG_LIM_MIN_LAM_AD[i] = 1
      ELSE
         LV_FAC_L_RNG_LIM_MIN_LAM_AD[i] = 0
         IF (FAC_L_RNG_LAM_AD[i] > C_FAC_MAX_L_RNG_LAM_AD)
         THEN
            FAC_L_RNG_LAM_AD[i] = C_FAC_MAX_L_RNG_LAM_AD
            LV_FAC_L_RNG_LIM_MAX_LAM_AD[i] = 1
         ELSE
            LV_FAC_L_RNG_LIM_MAX_LAM_AD[i] = 0
         ENDIF
      ENDIF
      LV_LAM_AD_STOP[i] = 1
   ENDIF
ENDIF
```

Deactivation: Transition to state "INIT"

IF (LV_LAM_AD_ACT[i] = 0)
THEN
STATE_LAM_AD[i] = "INIT"
ENDIF

Condition 3 NOT: Transition to state "WAIT"

IF (N_32 < C_N_BOL_FAC_L_RNG_LAM_AD) **OR**
(N_32 > C_N_TOL_FAC_L_RNG_LAM_AD) **OR**
(MFF_SEL_LAM_AD < ID_MFF_BOL_FAC_L_RNG_LAM_AD) **OR**
(MFF_SEL_LAM_AD > ID_MFF_TOL_FAC_L_RNG_LAM_AD
THEN
STATE_LAM_AD[i] = "WAIT"
ENDIF

Condition 6 NOT: Transition to state "CDN_FAC_L"

IF (LV_LAM_AD_ENA = 0) **OR**
(LV_LAM_LSCL[i] = 0) **OR**
(LV_LAM_STOP[i] = 1) **OR**
(LV_LDC_LAM_AD[i] = 0)
THEN
STATE_LAM_AD[i] = "CDN_FAC_L"
ENDIF

7.51.1.8 State "ADAPT_FAC_H"

Description:

Analogous procedure as described for state "ADAPT_ADD".

Formula section:

Actions:

```
LV_FAC_H_RNG_LAM_AD[i] = 1

IF (LV_LAM_AD_STOP[i] = 0) AND
   (LV_INH_CTR_RAF_CHG[i] OR (CTR_RAF_CHG[i]n ≠ CTR_RAF_CHG[i]n-1))
THEN
   CTR_RAF_CHG_LAM_AD[i] = CTR_RAF_CHG_LAM_AD[i] + 1
   IF (CTR_RAF_CHG_LAM_AD[i] >= NR_RAF_CHG_LAM_AD[i])
   THEN
      FAC_LAM_ADJ_LAM_AD[i] = CRLC_LAM_AD[i] * FAC_LAM_MV_MMV[i]
      FAC_DELTA_H_RNG_LAM_AD[i] = FAC_LAM_ADJ_LAM_AD[i]
      T_PRI_LAM_AD[i] = ID_T_PRI_FAC_H_RNG_LAM_AD
      FAC_H_RNG_LAM_AD[i] = FAC_H_RNG_LAM_AD[i] + FAC_DELTA_H_RNG_LAM_AD[i]
      IF (FAC_H_RNG_LAM_AD[i] < C_FAC_MIN_H_RNG_LAM_AD)
      THEN
         FAC_H_RNG_LAM_AD[i] = C_FAC_MIN_H_RNG_LAM_AD
         LV_FAC_H_RNG_LIM_MIN_LAM_AD[i] = 1
      ELSE
         LV_FAC_H_RNG_LIM_MIN_LAM_AD[i] = 0
         IF (FAC_H_RNG_LAM_AD[i] > C_FAC_MAX_H_RNG_LAM_AD)
         THEN
            FAC_H_RNG_LAM_AD[i] = C_FAC_MAX_H_RNG_LAM_AD
            LV_FAC_H_RNG_LIM_MAX_LAM_AD[i] = 1
         ELSE
            LV_FAC_H_RNG_LIM_MAX_LAM_AD[i] = 0
         ENDIF
      ENDIF
      LV_LAM_AD_STOP[i] = 1
   ENDIF
ENDIF
```

Deactivation: Transition to state "INIT"

```
IF (LV_LAM_AD_ACT[i] = 0)
THEN
   STATE_LAM_AD[i] = "INIT"
ENDIF
```

Condition 4 NOT: Transition to state "WAIT"

```
IF (N_32 < C_N_BOL_FAC_H_RNG_LAM_AD) OR
   (N_32 > C_N_TOL_FAC_H_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD < ID_MFF_BOL_FAC_H_RNG_LAM_AD) OR
   (MFF_SEL_LAM_AD > ID_MFF_TOL_FAC_H_RNG_LAM_AD)
THEN
   STATE_LAM_AD[i] = "WAIT"
ENDIF
```

Condition 7 NOT: Transition to state "CDN_FAC_H"

IF (LV_LAM_AD_ENA = 0) **OR**
  (LV_LAM_LSCL[i] = 0) **OR**
  (LV_LAM_STOP[i] = 1) **OR**
  (LV_LDC_LAM_AD[i] = 0)
THEN
  STATE_LAM_AD[i] = "CDN_FAC_H"
ENDIF

7.51.2 Rewriting of adaptation correction with non volatile stored values

Description:

In case of an error of the WRAF sensor the adaptation correction values MFF_ADD_LAM_AD[i], FAC_L_RNG_LAM_AD[i] and FAC_H_RNG_LAM_AD[i] must be rewritten with the values stored at the end of the last driving cycle.

Formula section:

IF (LV_LAM_AD_DEAC_ERR[i] = 0 -> 1)
THEN
  rewrite MFF_ADD_LAM_AD[i] *with non volatile stored value*
  rewrite FAC_L_RNG_LAM_AD[i] *with non volatile stored value*
  rewrite FAC_H_RNG_LAM_AD[i] *with non volatile stored value*
ENDIF

7.51.3 Determination of the interpolated lambda adaptation factor

Description:

In order to meet the physical behaviour of engine components (especially the air mass flow sensor), the actual multiplicative factor is calculated out of the two determined factors by interpolation at every state.

Formula section:

```
IF (MFF_SP[i] <= IP_MFF_SP_MAX_FAC_L_RNG_LAM_AD)
THEN
   FAC_LAM_AD[i] = FAC_L_RNG_LAM_AD[i]
ELSE
   IF (MFF_SP[i] >= IP_MFF_SP_MIN_FAC_H_RNG_LAM_AD)
   THEN
      FAC_LAM_AD[i] = FAC_H_RNG_LAM_AD[i]
   ELSE    /* interpolation */
      FAC_LAM_AD[i] = FAC_L_RNG_LAM_AD[i] +
                      (MFF_SP[i] - IP_MFF_SP_MAX_FAC_L_RNG_LAM_AD) *
                      ( (FAC_H_RNG_LAM_AD[i] - FAC_L_RNG_LAM_AD[i]) /
                        (IP_MFF_SP_MIN_FAC_H_RNG_LAM_AD -
                         IP_MFF_SP_MAX_FAC_L_RNG_LAM_AD) )
   ENDIF
ENDIF
```

7.51.4 Calculation of the output signals for fuel mass flow set point correction

Description:

In order to avoid mixture problems when the engine has not reached its service temperature (warm up), lambda adaptation factors and their corresponding relative intermediate results can be decreased versus TCO and their own value (which is applied when the engine has its service temperature).
This is performed by weighting the adaptive factors as long as the lambda adaptation is disabled due to TCO ≤ C_TCO_MIN_LAM_AD (see: "application incidences").
The quotient of the additive relative factor is determined. The values MFF_LAM_ADD_LAM_AD_OUT[i] and MFF_SP[i] should be conditioned so, that FAC_MFF_ADD_LAM_AD_OUT[i] does not exceed the physical range of +/-100 %!
The relative lambda adaptation FAC_MFF_ADD_FAC_LAM_AD[i] as long term adaptation is calculated out of both adaptation factors.

Formula section:

```
IF (TCO <= C_TCO_MIN_LAM_AD)
THEN
   MFF_LAM_ADD_LAM_AD_OUT[i]  = (MFF_ADD_LAM_AD[i] * IP_FAC_N_MFF_ADD_LAM_AD) *
                                IP_FAC_WUP_MFF_ADD_LAM_AD
   FAC_LAM_AD_LAM_OUT[i] = (FAC_LAM_AD[i]  * IP_FAC_N_FAC_LAM_AD) *
                           IP_FAC_WUP_FAC_LAM_AD
ELSE
   MFF_LAM_ADD_LAM_AD_OUT[i] = MFF_ADD_LAM_AD[i] * IP_FAC_N_MFF_ADD_LAM_AD
   FAC_LAM_AD_LAM_OUT[i] = FAC_LAM_AD[i] * IP_FAC_N_FAC_LAM_AD
ENDIF

FAC_MFF_ADD_LAM_AD_OUT[i] = MFF_LAM_ADD_LAM_AD_OUT[i] / MFF_SP[i]

IF (LV_IGK = 1) AND (LV_ST_END =1)
THEN
   FAC_MFF_ADD_FAC_LAM_AD[i] = FAC_LAM_AD_LAM_OUT[i] +
FAC_MFF_ADD_LAM_AD_OUT[i]
ENDIF
```

7.51.5 Interface to lambda controller

Description:

A shift of the lambda controller output after the calculation of a valid adaptation value by FAC_LAM_ADJ_LAM_AD[i] is required in order to avoid rapid changes in the air fuel ratio caused by the lambda adaptation. FAC_LAM_ADJ_LAM_AD[i] is calculated in the states "ADAPT_ADD", "ADAPT_FAC_L" and "ADAPT_FAC_H".
The shift of the lambda controller output is not carried out, if the lambda controller is stopped. To avoid a loss of adaptation values in that case, the lambda controller shall confirm the successful lambda controller shift.
If the lambda controller does not confirm the controller shift by FAC_LAM_ADJ_LAM_AD[i] by setting LV_FAC_LAM_ADJ_LAM_AD_END[i] to 1, the lambda adaptation stops further calculations (LV_LAM_AD_STOP[i] is set to 1) until the controller shift is confirmed. FAC_LAM_ADJ_LAM_AD[i] shall be set to 0 and LV_FAC_LAM_ADJ_LAM_AD[i] and LV_LAM_AD_STOP[i] are reset, if at a stopped lambda adaptation the deactivating conditions are fulfilled.

Formula section:

IF (LV_LAM_AD_STOP[i] = 1) **AND**
   (LV_LAM_STOP[i] = 0)
THEN
   LV_FAC_LAM_ADJ_LAM_AD[i] = 1
ENDIF

If (LV_FAC_LAM_ADJ_LAM_AD_END[i] = 1)
THEN
   LV_FAC_LAM_ADJ_LAM_AD[i] = 0
   LV_LAM_AD_STOP[i] = 0
   CTR_RAF_CHG_LAM_AD[i] = 0
ENDIF

7.51.6 Check of requirements for time scheduler

Description:

In order to observe the priority of further lambda adaptations, the differences between the old and the new adapted factors are observed. According to the observed deviations the new maximum time for next adaptation is calculated regarding the present conditions. The output value for the time scheduler T_PRI_TOT_LAM_AD is not bank selective and represents the minimum time of all cylinder banks. In case of single exhaust cylinder bank T_PRI_TOT_LAM_AD is equal to T_PRI_LAM_AD[i] of this single bank.

Formula section:

set T_PRI_TOT_LAM_AD to 6553.5 s before executing FOR loop

set LV_LAM_AD_STOP_CBK_EX to 0 before executing FOR loop

T_PRI_TOT_LAM_AD = **MIN**(T_PRI_TOT_LAM_AD; T_PRI_LAM_AD[i])

LV_LAM_AD_STOP_CBK_EX = LV_LAM_AD_STOP_CBK_EX **OR** LV_LAM_AD_STOP[i]

the following IF statement must be executed after the FOR loop

```
IF (LV_LAM_AD_STOP_CBK_EX = 1)
THEN
   IF (T_PRI_TOT_LAM_AD ≥ C_T_PRI_MAX_LAM_AD)
   THEN
      LV_LAM_AD_END = 1
   ELSE
      LV_LAM_AD_END = 0
   ENDIF
ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_WUP_MFF_ADD_LAM_AD	5*5	0...FFH	0...0.996094	3.9063E-3	-
LDPM_TCO_2_LACO	5	0...FEH	-48...142.5	0.75	°C
LDP_MFF_ADD_LAM_AD_IP_FAC_WUP	5	0...FFFFH	-694.510597...694.489403	0.021195	mg/stk
factor on lambda adaptation offset during warm up					
IP_FAC_WUP_FAC_LAM_AD	5*5	0...FFH	0...0.996094	3.9063E-3	-
LDPM_TCO_2_LACO	5	0...FEH	-48...142.5	0.75	°C
LDP_FAC_LAM_AD_IP_FAC_WUP_FAC	5	0...FFFFH	-50...49.998474	1.5259E-3	%
factor on lambda adaptation factor during warm up					
IP_FAC_N_MFF_ADD_LAM_AD	6	0...FFH	0...0.996094	3.9063E-3	-
LDPM_N_32_4_LACO	6	0...FFH	0...8160	32	rpm
factor on lambda adaptation offset depending on engine speed					
IP_FAC_N_FAC_LAM_AD	6	0...FFH	0...0.996094	3.9063E-3	-
LDPM_N_32_4_LACO	6	0...FFH	0...8160	32	rpm
factor on lambda adaptation factor depending on engine speed					
ID_T_PRI_MFF_ADD_RNG_LAM_AD	8	0...FFFFH	0...6553.5	0.1	s
LDP_MFF_DELTA_ADD_LAM_AD_T_PRI	8	0...FFFFH	-694.510597...694.489403	0.021195	mg/stk
priority time for next requested lambda adaptation according to offset difference					
ID_T_PRI_FAC_L_RNG_LAM_AD	8	0...FFFFH	0...6553.5	0.1	s
LDP_FAC_DELTA_L_RNG_LAM_AD_PRI	8	0...FFFFH	-50...49.998474	1.5259E-3	%
priority time for next requested lambda adaptation according to factor difference or lower area					
ID_T_PRI_FAC_H_RNG_LAM_AD	8	0...FFFFH	0...6553.5	0.1	s
LDP_FAC_DELTA_H_RNG_LAM_AD_PRI	8	0...FFFFH	-50...49.998474	1.5259E-3	%
priority time for next requested lambda adaptation according to factor difference or upper area					
C_T_PRI_MAX_LAM_AD	1	0...FFFFH	0 ... 6553.5	0.1	s
maximum priority time to set temporary end of adaptation flag					
C_N_TOL_ADD_RNG_LAM_AD	1	0...FFH	0...8160	32	rpm
maximum engine speed for offset lambda adaptation					
C_N_BOL_FAC_L_RNG_LAM_AD	1	0...FFH	0...8160	32	rpm
minimum engine speed for lower area factor lambda adaptation					
C_N_TOL_FAC_L_RNG_LAM_AD	1	0...FFH	0...8160	32	rpm
maximum engine speed for lower area factor lambda adaptation					
C_N_BOL_FAC_H_RNG_LAM_AD	1	0...FFH	0...8160	32	rpm
minimum engine speed for upper area factor lambda adaptation					
C_N_TOL_FAC_H_RNG_LAM_AD	1	0...FFH	0...8160	32	rpm
maximum engine speed for upper area factor lambda adaptation					
C_MFF_TOL_ADD_RNG_LAM_AD	1	0...FFFFH	0...1389	0.021195	mg/stk
maximum fuel mass set point for offset lambda adaptation					
ID_MFF _BOL_FAC_L_RNG_LAM_AD	8	0...FFFFH	0...1389	0.021195	mg/stk
LDP_N_32_ID_MFF _BOL_L_RNG	8	0...FFH	0...8.16E+3	32	rpm
minimum fuel mass set point for lower area factor lambda adaptation					
ID_MFF _BOL_FAC_H_RNG_LAM_AD	8	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
LDP_N_32_ID_MFF _BOL_LAM_AD	8	0...FFH	0...8.16E+3	32	rpm
minimum fuel mass set point for upper area factor lambda adaptation					
ID_MFF _TOL_FAC_L_RNG_LAM_AD	8	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
LDP_N_32_ID_MFF _TOL_LAM_AD	8	0...FFH	0...8.16E+3	32	rpm
maximum fuel mass set point for lower area factor lambda adaptation					
ID_MFF _TOL_FAC_H_RNG_LAM_AD	8	0...FFFFH	0...1389	0.021195	mg/stk
LDP_N_32_ID_MFF _TOL_H_RNG	8	0...FFH	0...8.16E+3	32	rpm
maximum fuel mass set point for upper area factor lambda adaptation					
IP_MFF_SP_MAX_FAC_L_RNG_LAM_AD	8	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
LDPM_N_32_5_LACO	8	0...FFH	0...8.16E+3	32	rpm
fuel mass set point threshold for consideration of only lower field lambda adaptation factor					
IP_MFF_SP_MIN_FAC_H_RNG_LAM_AD	8	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
LDPM_N_32_5_LACO	8	0...FFH	0...8.16E+3	32	rpm
fuel mass set point threshold for consideration of only upper field lambda adaptation factor					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_MIN_L_RNG_LAM_AD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
minimum value of lower area lambda adaptation factor					
C_FAC_MAX_L_RNG_LAM_AD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
maximum value of lower area lambda adaptation factor					
C_FAC_MIN_H_RNG_LAM_AD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
minimum value of upper area lambda adaptation factor					
C_FAC_MAX_H_RNG_LAM_AD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
maximum value of upper area lambda adaptation factor					
C_MFF_MIN_ADD_RNG_LAM_AD	1	8000...7FFFH	-694.510597...694.489403	0.021195	mg/stk
minimum value of lambda adaptation offset					
C_MFF_MAX_ADD_RNG_LAM_AD	1	8000...7FFFH	-694.510597...694.489403	0.021195	mg/stk
maximum value of lambda adaptation offset					
LC_FAC_L_RNG_AFS_REQ_LAM_AD	1	0...1H	0...1	1	-
logical calibration to allow forced lambda eq. 1 conditions for factor adaptation at upper area					
LC_FAC_H_RNG_AFS_REQ_LAM_AD	1	0...1H	0...1	1	-
logical calibration to allow forced lambda eq. 1 conditions for factor adaptation at upper area					
LC_MFF_ADD_RNG_AFS_REQ_LAM_AD	1	0...1H	0...1	1	-
logical calibration to allow forced lambda eq. 1 conditions for offset adaptation					
LC_SEL_MFF_LAM_AD	1	0…1H	0…1	1	-
Selection of fuel mass flow for lambda adaptation activation condition(0:MFF_BAS,1:MFF_SP[i])					

7.52 Mean value calculation for limited dynamics

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_LAM_MV_DELTA_LDC[NC_CBK_EX_NR]	O/V	8000...7FFFH	-50...49.9985	0.00152588	%
difference between FAC_LAM_MV[i] and moving mean value of FAC_LAM_MV[i] for limited dynamics detection					
MAF_DELTA_LDC	O/V	8000...7FFFH	-694.511 ... 694.489	0.0211948	mg/stk
difference between MAF and moving mean value of MAF for limited dynamics detection					
N_DELTA_LDC	O/V	E020...1FE0H	-8.16E+3 ... 8.16E+3	1	rpm
difference between N and moving mean value of N for limited dynamics detection					
PV_AV_DELTA_LDC	O/V	80...7FH	-50...49.6094	0.390625	%
difference between pedal value and moving mean value of pedal value for limited dynamics detection					
MAF_MMV_LDC	V	0...FFFFH	0...1.389E+3	0.0211948	mg/stk
moving mean value of MAF for limited dynamics detection					
N_MMV_LDC	V	0...1FE0H	0...8.16E+3	1	rpm
moving mean value of N for limited dynamics detection					
PV_AV_MMV_LDC	V	0...FFH	0...99.6094	0.390625	%
moving mean value of pedal value for limited dynamics detection					

Input data:

FAC_LAM_MV[NC_CBK_EX_NR]	FAC_LAM_MV_MMV_LDC[NC_CBK_EX_NR]	LV_ST_END	MAF
N	NC_CBK_EX_NR	PV	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_MAF_MMV_LDC	1	0...FFFFH	0...0.999985	1.52588E-5	-
correlation constant for moving mean value calculation of MAF for limited dynamics					
C_CRLC_N_MMV_LDC	1	0...FFFFH	0...0.999985	1.52588E-5	-
correlation constant for moving mean value calculation of N for limited dynamics					
C_CRLC_PV_AV_MMV_LDC	1	0...FFFFH	0...0.999985	1.52588E-5	-
correlation constant for moving mean value calculation of pedal value for limited dynamics					

7.52.1 LACO_ISPCLMVLDC0

This function calculates only differences between base values and their moving means values. These differences or deltas are to be used within the function that needs limited dynamic conditions. The evaluation of the limited dynamic bit is part of the function that need this flag. So the mean value calculation is concentrated in this function.

NC_CBK_EX_NR defines the number of exhaust banks.

For vector elements the variable extension "_i" is used in the model instead of "[i]" as found in the textual description.

Function Description

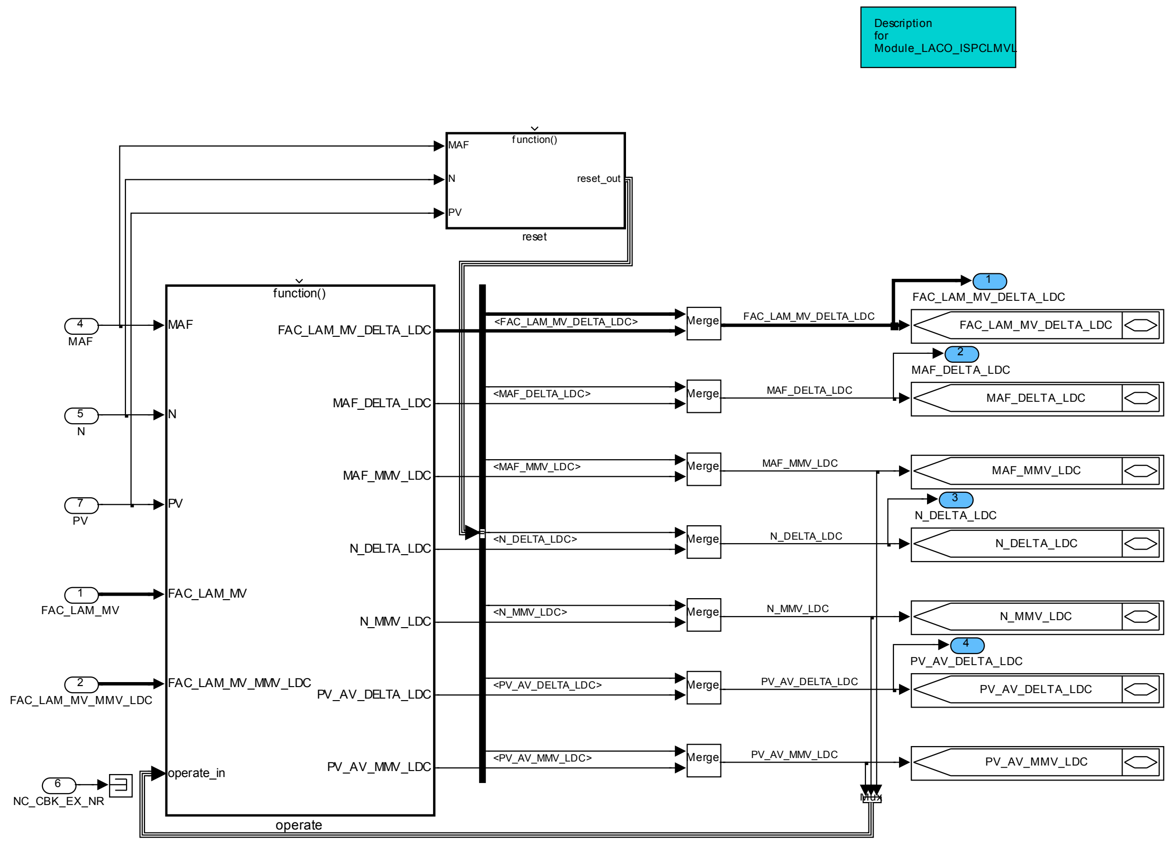


Figure 95 LACO_ISPCLMVLDC0

operate

This function calculates for the detection of the limited dynamics the moving mean values of engine speed (N and N_MMV_LDC), air mass flow (MAF and MAF_MMV_LDC) and degree of activation of the accelerator pedal (PV and PV_AV_MMV_LDC). Furthermore the difference between these input values and the moving mean values is calculated.

For the lambda control output only the difference is evaluated because the moving mean value calculation is located in the lambda control module itself because of easier realisation of shift operations of the moving mean value (FAC_LAM_MV_MMV_LDC[i]).

Remark: The difference calculation of lambda control output is a vector operation.

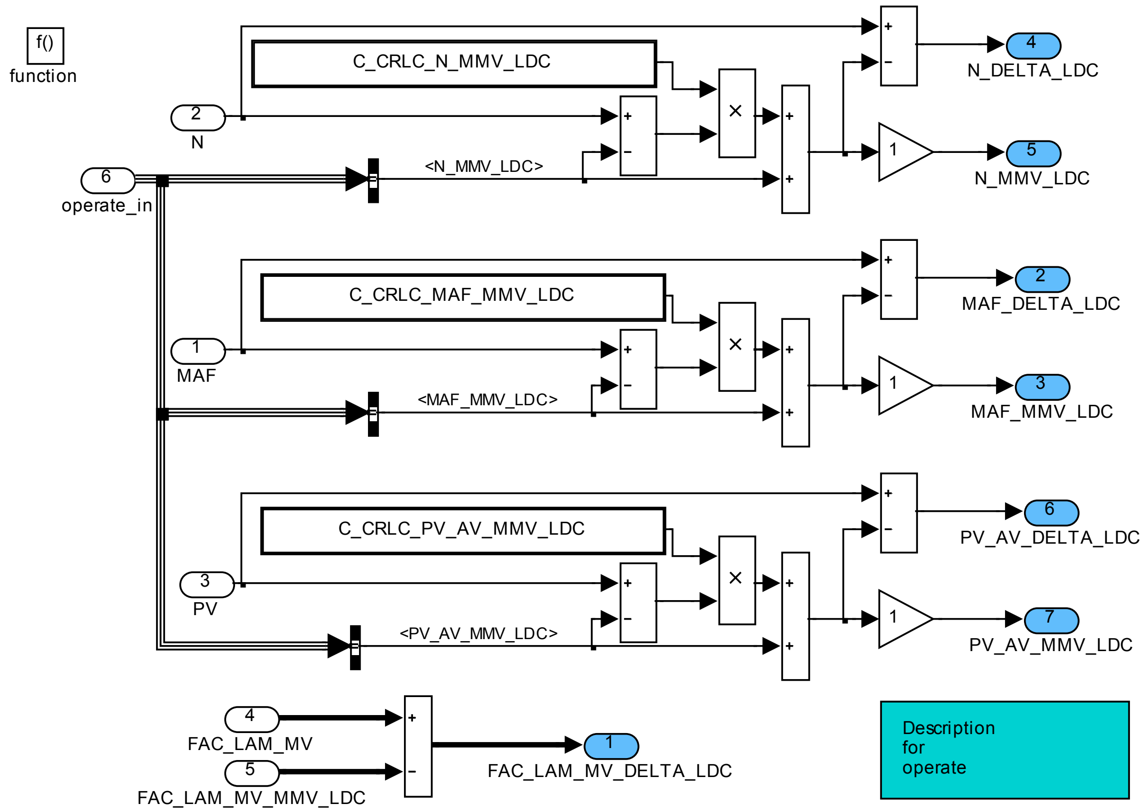


Figure 96 LACO_ISPCLMVLDC0/ operate

7.53 Application incidence for the Lambda Adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LAM_AD_ACT[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
flag indicating that lambda adaptation can be activated					
LV_LAM_AD_DEAC_ERR[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag indicating that lambda adaptation is deactivated due to an error					
CRLC_LAM_AD[NC_CBK_EX_NR]	V/O	0...FFH	0...0.996094	3.9063E-3	-
correlation constant for calculation of adaptive values					
NR_RAF_CHG_LAM_AD[NC_CBK_EX_NR]	V/O	1...FFH	1...255	1	-
number of lean <-> rich air fuel ratio changes between two adaptations					
LV_INH_CTR_RAF_CHG[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
flag to inhibit the lean / rich counting during adaptation					
LV_LDC_LAM_AD[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
limited dynamic conditions for lambda adaptation					
N_OFS_LDC_LAM_AD[NC_CBK_EX_NR]	V	E020...1FE0H	-8160...8160	1	rpm
engine speed offset for limited dynamics calculation					
MAF_OFS_LDC_LAM_AD[NC_CBK_EX_NR]	V	8000...7FFFH	-694.510597... 694.489403	0.021195	mg/stk
air mass flow offset for limited dynamics calculation					
PV_AV_OFS_LDC_LAM_AD[NC_CBK_EX_ NR]	V	80...7FH	-50...49.609375	0,390625	[%]
actual pedal value offset for limited dynamics calculation 1					
FAC_LAM_MV_OFS_LDC_LAM_AD[NC_CB K_EX_NR]	V	8000...7FFFH	-50...49.998474	1.5259E-3	mg/stk
filtered lambda controller output offset for limited dynamics calculation					
MAF_INT_LDC_LAM_AD[NC_CBK_EX_NR]	V	0...FFFFH	0...1820.416667	0.027778	g
air mass flow integral during limited dynamics not fulfilled					

Input data:

	FAC_LAM_MV_DELTA_LDC[NC_CBK_EX_NR]
LV_ERR_LAM_LIM[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]
TCO	NC_CBK_EX_NR
LV_ST_END	TIA_IM
MAF	MAF_CYL
LV_ERR_MEC_OPEN_CPS	LV_ERR_EL_CPS
LV_MAF_PULS	LV_ERR_TCO
LV_ERR_MAF	LV_IS
N_DELTA_LDC	LV_ERR_TIA_IM
PV_AV_DELTA_LDC	MAF_DELTA_LDC
LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_LAM_LIM_LAM_AD[NC_CBK_EX_NR]
LV_ERR_CRK	LV_ERR_IGC
LV_ERR_IV[NC_CYL_NR]	LV_ERR_MEC_IVVT_IN_i
LV_ERR_SLV_IVVT_IN_i	LV_ERR_TPS
LV_ERR_ISA_i	LV_ERR_RATIO_CHK
LV_MIS_STATE_A	LV_MIS_STATE_B
LV_ERR_MAP	

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.

For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2,

otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Description:

The activation conditions of the lambda adaptation depend of:

- coolant temperature thresholds,
- temperature of intake air into manifold threshold,
- air mass flow threshold,
- no air mass flow pulsation detected,
- and several error flags that must be zero

If all activation conditions are fulfilled LV_LAM_AD_ACT[i] is set to 1. That means that the lambda adaptation can get active if no other inhibition defined in the lambda adaptation module is set. In case of upstream sensor failure LV_LAM_AD_DEAC_ERR[i] is set to 1.

The detection of the limited dynamics are calculated based the current engine speed, air mass flow, pedal value and filtered lambda control output. The input to this function is the difference between each of the mentioned base values and its moving mean values. If one of this differences exceeds a calibrated threshold a air mass flow integral is started to be calculated until a calibrated MAF threshold is reached. During the calculation the limited dynamics are not fulfilled (LV_LDC_LAM_AD[i] = 0). In order to reset difference between base value and moving mean value after each threshold exceeding a offset variable is introduced (see formula section).

Furthermore the correlation constant for calculation of adaptive values and the number of lean ↔ rich air fuel ratio changes between two adaptations are determined depending on idle speed conditions or request for forced lambda adaptation. The flag LV_INH_CTR_RAF_CHG_CTR[i] indicating that the counting of lean / rich mixture changes is inhibited is set to 1 when forced lambda adaptation is requested.

Application conditions:

Initialisation: at ECU reset:
LV_LAM_AD_ACT[i] = 0
CRLC_LAM_AD[i] = 0
NR_RAF_CHG_LAM_AD[i] = 1
LV_LDC_LAM_AD[i] = 0
N_OFS_LDC_LAM_AD[i] = 0
MAF_OFS_LDC_LAM_AD[i] = 0
PV_AV_OFS_LDC_LAM_AD[i] = 0
FAC_LAM_MV_OFS_LDC_LAM_AD[i] = 0
MAF_INT_LDC_LAM_AD[i] = 0
LV_LAM_AD_DEAC_ERR[i] = 0

Recurrence: T_SAMPLE = 20 ms

Activation: LV_ST_END = 1

Deactivation: -

Formula section:

Determination of activation conditions for lambda adaptation:

IF (TCO < C_TCO_MAX_LAM_AD) **AND**
(TCO > C_TCO_MIN_LAM_AD) **AND**
(TIA_IM < C_TIA_MAX_LAM_AD) **AND**
(MAF < C_MAF_MAX_LAM_AD) **AND**
(LV_MAF_PULS = 0) **AND**
(LV_ERR_EL_CPS = 0) **AND**
(LV_ERR_MEC_OPEN_CPS = 0) **AND**
(LV_ERR_TCO = 0) **AND**
(LV_ERR_TIA_IM = 0) **AND**
(LV_ERR_MAF = 0) **AND**
(LV_ERR_MAP = 0) **AND** {(LV_ERR_LAM_LIM[i] = 0) **or** (LC_ENA_LAM_AD_LAM_LIM = 1) }
AND
(LV_ERR_LSH_UP[i] = 0) **AND**
(LV_ERR_CRK = 0) **AND**
(LV_ERR_IGC = 0) **AND**
(LV_ERR_IV[NC_CYL_NR] = 0) **AND**
(LV_ERR_MEC_IVVT_IN_i = 0) **AND**
(LV_ERR_SLV_IVVT_IN_i = 0) **AND**
(LV_ERR_TPS = 0) **AND**
(LV_ERR_ISA_i = 0) **AND**
(LV_ERR_RATIO_CHK = 0) **AND**
(LV_MIS_STATE_A = 0) **AND**
(LV_MIS_STATE_B = 0)

THEN
LV_LAM_AD_ACT[i] = 1
ELSE
LV_LAM_AD_ACT[i] = 0
ENDIF

LV_LAM_AD_DEAC_ERR[i] = LV_ERR_LS_UP[i]

Detection of limited dynamic conditions for lambda adaptation:

```
IF (ABS(N_DELTA_LDC) > C_N_DYW_LDC_LAM_AD)
THEN
  IF (ABS(N_DELTA_LDC - N_OFS_LDC_LAM_AD[i]) > C_N_DYW_LDC_LAM_AD)
  THEN
    N_OFS_LDC_LAM_AD[i] = N_DELTA_LDC
    MAF_INT_LDC_LAM_AD[i] = 0
  ENDIF
ELSE
  N_OFS_LDC_LAM_AD[i] = 0
ENDIF

IF (ABS(MAF_DELTA_LDC) > C_MAF_DYW_LDC_LAM_AD)
THEN
  IF (ABS(MAF_DELTA_LDC - MAF_OFS_LDC_LAM_AD[i]) > C_MAF_DYW_LDC_LAM_AD)
  THEN
    MAF_OFS_LDC_LAM_AD[i] = MAF_DELTA_LDC
    MAF_INT_LDC_LAM_AD[i] = 0
  ENDIF
ELSE
  MAF_OFS_LDC_LAM_AD[i] = 0
ENDIF

IF (ABS(PV_AV_DELTA_LDC) > C_PV_AV_DYW_LDC_LAM_AD)
THEN
  IF (ABS(PV_AV_DELTA_LDC - PV_AV_OFS_LDC_LAM_AD[i]) >
     C_PV_AV_DYW_LDC_LAM_AD)
  THEN
    PV_AV_OFS_LDC_LAM_AD[i] = PV_AV_DELTA_LDC
    MAF_INT_LDC_LAM_AD[i] = 0
  ENDIF
ELSE
  PV_AV_OFS_LDC_LAM_AD[i] = 0
ENDIF

IF (ABS(FAC_LAM_MV_DELTA_LDC[i]) > C_FAC_LAM_MV_DYW_LDC_LAM_AD)
THEN
  IF (ABS(FAC_LAM_MV_DELTA_LDC[i] – FAC_LAM_MV_OFS_LDC_LAM_AD[i]) >
     C_FAC_LAM_MV_DYW_LDC_LAM_AD)
  THEN
    FAC_LAM_MV_OFS_LDC_LAM_AD[i] = FAC_LAM_MV_DELTA_LDC[i]
    MAF_INT_LDC_LAM_AD[i] = 0
  ENDIF
ELSE
  FAC_LAM_MV_OFS_LDC_LAM_AD[i] = 0
ENDIF


IF (MAF_INT_LDC_LAM_AD[i] < C_MAF_INT_LDC_LAM_AD)
THEN
  MAF_INT_LDC_LAM_AD[i] [g] = MAF_INT_LDC_LAM_AD[i] [g] +
                        MAF_CYL [kg/h] * T_SAMPLE [ms] * 1/3600 [(g*h) / (kg*ms)]
  LV_LDC_LAM_AD[i] = 0
ELSE
  LV_LDC_LAM_AD[i] = 1
ENDIF
```

Determination of correlation constant for lambda adaptation:

```
IF (LV_LAM_LIM_LAM_AD[i] = 1)
THEN
   LV_INH_CTR_RAF_CHG[i] = 1
   CRLC_LAM_AD[i] = C_CRLC_LAM_AD_LAM_LIM
   NR_RAF_CHG_LAM_AD[i] = C_T_DLY_LAM_AD_LAM_LIM / T_SAMPLE
ELSE
   LV_INH_CTR_RAF_CHG[i] = 0
   IF (LV_IS = 1)
   THEN
      CRLC_LAM_AD[i] = C_CRLC_LAM_AD_IS
      NR_RAF_CHG_LAM_AD[i] = C_NR_RAF_CHG_LAM_AD_IS
   ELSE
      CRLC_LAM_AD[i] = C_CRLC_LAM_AD
      NR_RAF_CHG_LAM_AD[i] = C_NR_RAF_CHG_LAM_AD
   ENDIF
ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_LAM_AD	1	0...FEH	-48...142.5	0.75	°C
minimum coolant temperature for adaptive learning					
C_TCO_MAX_LAM_AD	1	0...FEH	-48...142.5	0.75	°C
maximum coolant temperature for adaptive learning					
C_TIA_MAX_LAM_AD	1	0...FEH	-48...142.5	0.75	°C
maximum air temperature for adaptive learning					
C_MAF_MAX_LAM_AD	1	0...FFFFH	0...1389	0.021195	mg/stk
maximum mass air flow for adaptive learning					
C_CRLC_LAM_AD	1	0...FFH	0...0.996094	3.9063E-3	-
correlation constant for lambda adaptation					
C_CRLC_LAM_AD_IS	1	0...FFH	0...0.996094	3.9063E-3	-
correlation constant for lambda adaptation at idle speed					
C_CRLC_LAM_AD_LAM_LIM	1	0...FFH	0...0.996094	3.9063E-3	-
correlation constant for lambda adaptation in case of lambda controller limitation					
C_NR_RAF_CHG_LAM_AD	1	1...FFH	1...255	1	-
number of lean <-> rich air fuel ratio changes between two adaptations					
C_NR_RAF_CHG_LAM_AD_IS	1	1...FFH	1...255	1	-
number of lean <-> rich air fuel ratio changes between two adaptations at idle speed					
C_T_DLY_LAM_AD_LAM_LIM	1	1...FFH	0.02...5.10	0.02	s
time between two adaptations in case of lambda controller limitation					
C_N_DYW_LDC_LAM_AD	1	0...1FE0H	0...8160	1	rpm
engine speed window for limited dynamic conditions trim control					
C_MAF_DYW_LDC_LAM_AD	1	0...FFFFH	0...1389	0.021195	mg/stk
air mass flow window for limited dynamic conditions lambda adaptation					
C_FAC_LAM_MV_DYW_LDC_LAM_AD	1	0...7FFFH	0...49.998474	1.5259E-3	%
lambda control output window for limited dynamic conditions lambda adaptation					
C_PV_AV_DYW_LDC_LAM_AD	1	0...FFH	0...99.609375	0.390625	%
pedal value window for limited dynamic conditions lambda adaptation					
C_MAF_INT_LDC_LAM_AD	1	0...FFFFH	0...1820.416667	0.027778	g
air mass flow integral for duration of violation of limited dynamic conditions lambda adaptation					
LC_ENA_LAM_AD_LAM_LIM	1	0…1H	0…1	1	-
Switch to enable lambda adaptation in case of lambda controller limit error detection					

7.54 Lambda Setpoint for Single Cylinder Fuel Shut Off

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LAMB_SCC[NC_CBK_EX_NR]	V/O	0 ... 7FFFH	0 ... 31.999	9.7656E-4	-
Lambda setpoint for single cylinder shut off					

Input data:

LV_SCC[NC_CBK_EX_NR]	LV_ST_END	LV_ES	TCO
NC_CBK_EX_NR			

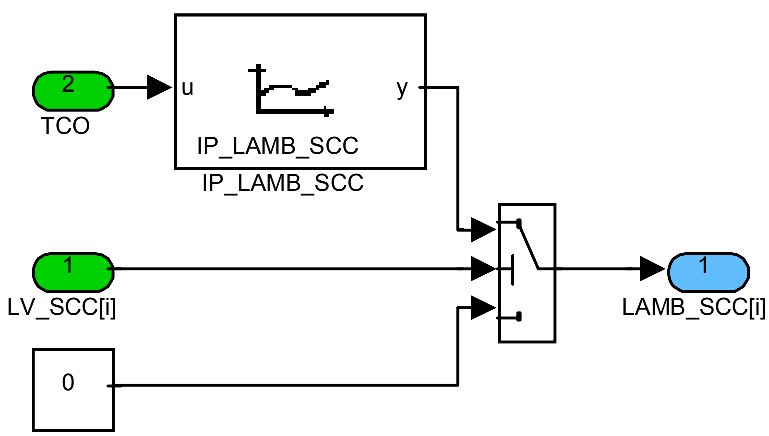
FUNCTION DESCRIPTION:

General information:

In case of single cylinder shut off one of the bank selective flags LV_SCC[i] is set to 1. Then the corresponding lambda value LAMB_SCC[i] is set to a value according to the mapping point of IP_LAMB_SCC (which should be equal or bigger than one). This is neccesary to protect the catalyst from unburnt hydrocarbons.

LAMB_SCC[i] is an input for the module “Lambda Coordination” and it is handled there with the highest priority against all other lambda requests.

Signal flow diagram:



Application conditions:

Activation: LV_ST_END = 1

Deactivation: -

Initialization at Reset or LV_ES = 0 -> 1: LAMB_SCC[i] = 0

Recurrence: every 10 ms

Formula section:

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

IF

LV_SCC[i] = 1

THEN

LAMB_SCC[i] = IP_ LAMB_SCC

ELSE

LAMB_SCC[i] = 0

ENDIF

ENDFOR

Note: All mapping points of IP_LAMB_SCC should be equal or bigger than one.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_LAMB_SCC	8	0...7FFFH	0 ... 31.999	9.7656E-4	-
LDP_TCO_IP_LAMB_SCC	8	0...FEH	-48...142.5	0.75	°C
Lambda setpoint for single cylinder shut off					

7.55 Basic Lambda Setpoint including Homogeneous Lean Mode

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_BAS[NC_CBK_EX_NR]	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
bank selective basic lambda setpoint					
LAMB_BAS_MV	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
basic lambda setpoint, mean value					
LAMB_BAS_H_RES[NC_CBK_EX_NR]	V/O	0...7FFFH	0 ... 1.999939	6.1035E-5	-
bank selective basic lambda setpoint with high resolution					
LAMB_HOM_AFL	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
lambda setpoint for homogeneous lean combustion					
LAMB_AFL_1	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
lambda setpoint for homogeneous lean step value					
LAMB_HOM_AFS[NC_CBK_EX_NR]	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
stoichiometric lambda setpoint					
LAMB_HOM_AFS_MV	V/O	0...7FFFH	0.. 31.999	9.7656E-4	-
stoichiometric lambda setpoint, mean value					
LAMB_HOM_AFS_H_RES[NC_CBK_EX_NR]	V/O	0...7FFFH	0 ... 1.999939	6.1035E-5	-
stoichiometric lambda setpoint with high resolution					
LV_LAMB_BAS_OFS_CAT_DIAG[NC_CBK_EX_NR]	V	0 ... 1H	0 ... 1	1	-
Flag to indicate that lambda offset for catalyst diagnosis is bigger than zero					

Input data:

N_32	TQI_REQ_SLOW	TCO	TCO_ST
LV_ST_END	MAF	LV_HOM_AFL_ACT	LV_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]
NC_CBK_EX_NR			

FUNCTION DESCRIPTION:

General information:

The basic lambda setpoint LAMB_BAS[i] is calculated as a function of load and engine speed. It is possible to correct the load - speed depending lambda setpoint for different coolant temperatures and the coolant temperature at engine start. This guarantees good driveability for lower coolant temperatures.

Two homogeneous engine operating modes are possible, the stoichiometric ($\lambda = 1$) mode and the lean air-fuel-mixture mode ($\lambda > 1$).

Signal flow diagram:

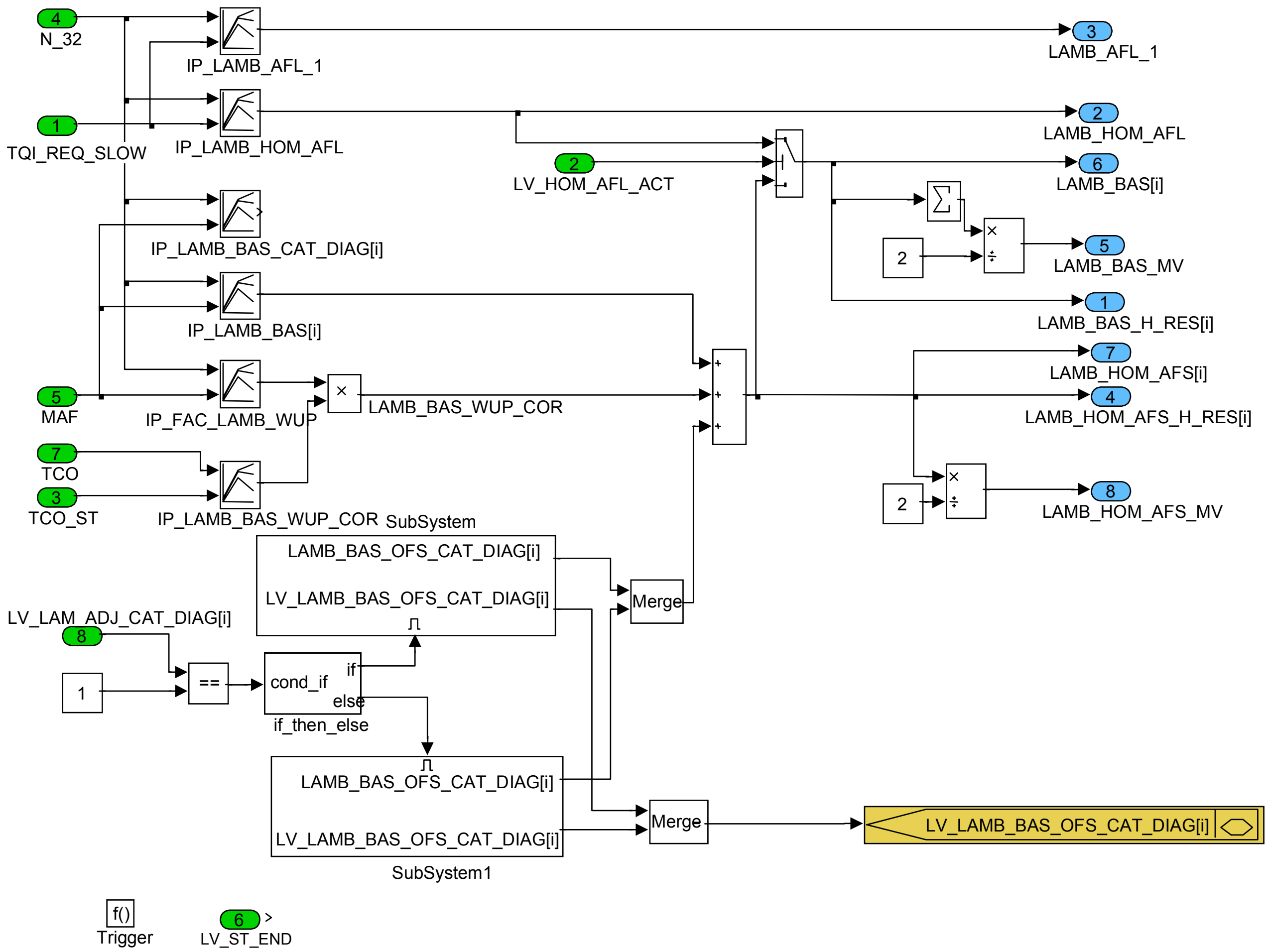


Diagram 1: Main flow

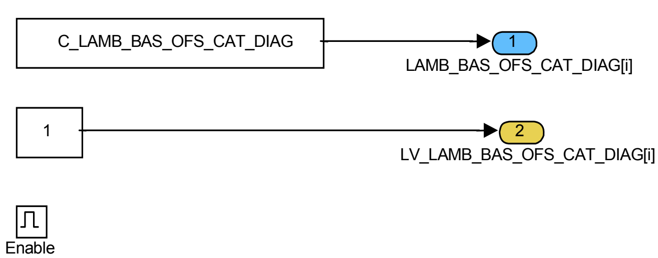


Diagram 2: Sub System

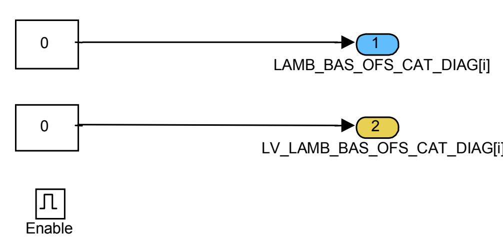


Diagram 3: Sub System1

Application conditions:

Activation: LV_ST_END = 1

Deactivation: -

Initialisation: at Reset or LV_ES = 0 -> 1: LAMB_BAS_H_RES[i] = 1
LAMB_BAS[i] = 1
LAMB_AFL_1 = 1
LAMB_HOM_AFL = 1
LAMB_HOM_AFS[i] = 1
LAMB_HOM_AFS_H_RES[i] = 1

Recurrence: 10 ms

Formula section:

The calculation of the basic lambda setpoint is executed in every engine operating state except engine start and engine stop. The load - speed depending value is corrected versus coolant temperature and coolant temperature at engine start and switches to AFL mode when requested. The step value for the switch is an extra output value.

The basic lambda setpoint is calculated with 2 different resolutions (LAMB_BAS[i] with a resolution of 0.98E-3 and LAMB_BAS_H_RES[i] with a resolution of 6.10E-5). The higher resolution is indicated in the variable by the suffix "_H_RES". If a transformation from high resolution variables to low resolution variables is necessary this calculation is indicated by a bit shift of 4 to the right (>> 4). The low resolution is 16-times lower ($2^4 = 16$). Vice versa from low to high resolution the bit shift is 4 times to the left (<<4).

LAMB_AFL_1 = IP_LAMB_AFL_1

LAMB_BAS_WUP_COR = IP_LAMB_BAS_WUP_COR * IP_FAC_LAMB_WUP

LAMB_HOM_AFL = IP_LAMB_HOM_AFL

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

(1) IF

LV_LAM_ADJ_CAT_DIAG[i] = 1

(In case of active catalyst diagnosis an offset is added to the basic lambda setpoint)

(1) THEN

LAMB_BAS_OFS_CAT_DIAG[i] = C_LAMB_BAS_OFS_CAT_DIAG

LV_LAMB_BAS_OFS_CAT_DIAG[i] = 1

(1) ELSE

LAMB_BAS_OFS_CAT_DIAG[i] = 0

LV_LAMB_BAS_OFS_CAT_DIAG[i] = 0

(1) ENDIF

LAMB_HOM_AFS_H_RES[i] = IP_LAMB_BAS[i]
+ LAMB_BAS_WUP_COR
+ LAMB_BAS_OFS_CAT_DIAG[i]

Remark: LAMB_HOM_AFS_H_RES[i] and LAMB_BAS_H_RES[i] are limited to range 0 ... 0x7FFF

LAMB_HOM_AFS[i] = LAMB_HOM_AFS_H_RES[i] (>> 4)

(2) IF LV_HOM_AFL_ACT = 1

(2) THEN LAMB_BAS[i] = LAMB_HOM_AFL
LAMB_BAS_H_RES[i] = LAMB_BAS[i] (<< 4)

(2) ELSE LAMB_BAS_H_RES[i] = LAMB_HOM_AFS_H_RES[i]
LAMB_BAS[i] = LAMB_HOM_AFS[i]

(2) ENDIF

ENDFOR

For two cylinder banks (NC_CBK_EX_NR = 2) the mean values LAMB_BAS_MV and LAMB_HOM_AFS_MV are calculated:

LAMB_BAS_MV = (LAMB_BAS[1] + LAMB_BAS[2]) / 2

LAMB_HOM_AFS_MV = (LAMB_HOM_AFS [1] + LAMB_HOM_AFS [2]) / 2

For single bank system (NC_CBK_EX_NR = 1):

LAMB_BAS_MV = LAMB_BAS[1]

LAMB_HOM_AFS_MV = LAMB_HOM_AFS [1]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_LAMB_BAS[NC_CBK_EX_NR]	8*8	0...7FFFH	0 ... 1.999939	6.1035E-5	-
LDPM_N_32_1_LASP	8	0...FFH	0...8160	32	rpm
LDPM_MAF_1_LASP	8	0..FFFFH	0 ... 1389	0.02119	mg/stk
basic lambda setpoint					
IP_LAMB_BAS_WUP_COR	8*8	0 ... FFFFH	-2 ... 1.9999	6.1035E-5	-
LDP_TCO_IP_LAMB_BAS_WUP_COR	8	0...FEH	-48...142.5	0.75	°C
LDP_TCO_ST_IP_LAMB_BAS_WUP_COR	8	0...FEH	-48...142.5	0.75	°C
warm-up correction of basic lambda					
IP_LAMB_HOM_AFL	8*8	0...7FFFH	0.. 31.999	9.7656E-4	-
LDP_N_32_IP_LAMB_HOM_AFL	8	0..FFH	0...8160	32	rpm
LDP_TQI_REQ_SLOW_IP_LAMB_HOM	8	0..FFFFH	-1024..1023.97	0.03125	Nm
lambda setpoint for homogeneous lean combustion					
IP_LAMB_AFL_1	8*8	0...7FFFH	0.. 31.999	9.7656E-4	-
LDP_N_32_IP_LAMB_AFL_1	8	0..FFH	0...8160	32	rpm
LDP_TQI_REQ_SLOW_IP_LAMB_AFL_1	8	0..FFFFH	-1024..1023.97	0.03125	Nm
lambda setpoint for homogeneous lean step value					
IP_FAC_LAMB_WUP	6*6	0...FFH	0...1.9921	7.8123 e-3	-
LDP_N_32_IP_FAC_LAMB_WUP	6	0...FFH	0...8160	32	rpm
LDP_MAF_IP_FAC_LAMB_WUP	6	0..FFFFH	0 ... 1389	0.02119	mg/stk
correction factor for basic lambda warm-up					
C_LAMB_BAS_OFS_CAT_DIAG	1	8000 ... 7FFFH	-2 ... 1.9999	6.1035E-5	-
Offset for basic lambda setpoint due to catalyst diagnosis active					

7.56 Lambda coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_SP[NC_CBK_EX_NR]	V/O	0...7FFFH	0....31.999	9.7656E-4	-
lambda setpoint					
LAMB_SP_HOM[NC_CBK_EX_NR]	V/O	0...7FFFH	0....1.999939	6.1035E-5	-
lambda setpoint with high resolution					
LAMB_SP_MV	V/O	0...7FFFH	0....31.999	9.7656E-4	-
lambda setpoint mean value					
LAMB_BAS_COR[NC_CBK_EX_NR]	V/O	0...7FFFH	0....31.999	9.7656E-4	-
lambda output for TQI_SP_MAF calculation					
LAMB_BAS_COR_MV	V/O	0...7FFFH	0....31.999	9.7656E-4	-
lambda output for TQI_SP_MAF calculation – mean value -					
LAMB_BAS_COR_1_H_RES[NC_CBK_EX_NR]	V	0...7FFFH	0....1.999939	6.1035E-5	-
lambda setpoint used for lambda = 1 detection					
LAMB_SP_BAS[NC_CBK_EX_NR]	V/O	0...7FFFH	0....31.999	9.7656E-4	-
lambda feedback for basic torque calculation					
LAMB_SP_BAS_MV	V/O	0...7FFFH	0....31.999	9.7656E-4	-
mean value of LAMB_SP_BAS[i]					
LAMB_BAS_COR_2_H_RES[NC_CBK_EX_NR]	-	0...7FFFH	0....1.999939	6.1035E-5	-
lambda setpoint used for lambda = 1 detection					

Input data:

LAMB_FL	LV_LAMB_SP_REQ_ACT	LAMB_BAS_COR_MAX	LAMB_CH
LAMB_SO2P	LAMB_SP_REQ	LAMB_SP_TQI	LV_ST_END
LV_ES	LV_SAWUP	LV_TQ_LAMB_ACT	LV_TI_CH
LV_LAMB_SP_REQ_DIAG_ACT[NC_CBK_EX_NR]	LAMB_SP_DIAG_LS_UP_DOWN[NC_CBK_EX_NR]	LAMB_COP[NC_CBK_EX_NR]	LAMB_BAS_H_RES[NC_CBK_EX_NR]
LAMB_SCC[NC_CBK_EX_NR]	LAMB_TUR_OHP[NC_CBK_EX_NR]	LAMB_RGN[NC_CBK_EX_NR]	LAMB_SP_EXT_ADJ[NC_CBK_EX_NR]
DELTA_LAMB_SP_PUC[NC_CBK_EX_NR]	LV_LAMB_SP_EXT_ADJ[NC_CBK_EX_NR]	NC_CBK_EX_NR	

FUNCTION DESCRIPTION:

General information:

This module is calculated in homogenous mode only.

For the calculation of the lambda setpoint a coordination between the different lambda requirements with different priorities must be made. If there are lambda requirements of the same priority, the minimum or maximum lambda value is selected, see figure. The calculation is executed every 10 ms in every engine operating state except engine start.
The module manages, for example, to increase the engine torque (LAMB_FL) or to protect the catalyst from overheating (LAMB_COP[i]) or to regenerate the NOx-trap catalyst (LAMB_RGN[i]) or to heat up the catalyst more quickly after start (LAMB_CH) or not or handles the desulfation (LAMB_SO2P). The lambda request for single cylinder shut off is inserted twice to ensure the highest priority (higher than full load) and a feedback to torque calculation.

The input LAMB_BAS_H_RES[i] has a resolution of 6.1E-5. The high resolution is indicated by the suffix "_H_RES". All calculations in this module have to be done with this high resolution. For this all other inputs to the module must be transformed to this high resolution. The output variables are needed by the SW-modules which calculate with the high resolution and/or with a low resolution of 9.8E-4. Therefore a transformation back to lower resolution is necessary. This is indicated by a bit shift of 4 to the right (>> 4).

The low resolution is 16-times lower ($2^4 = 16$). Output variables without the suffix "_H_RES" have low resolution.

Signal flow diagram:

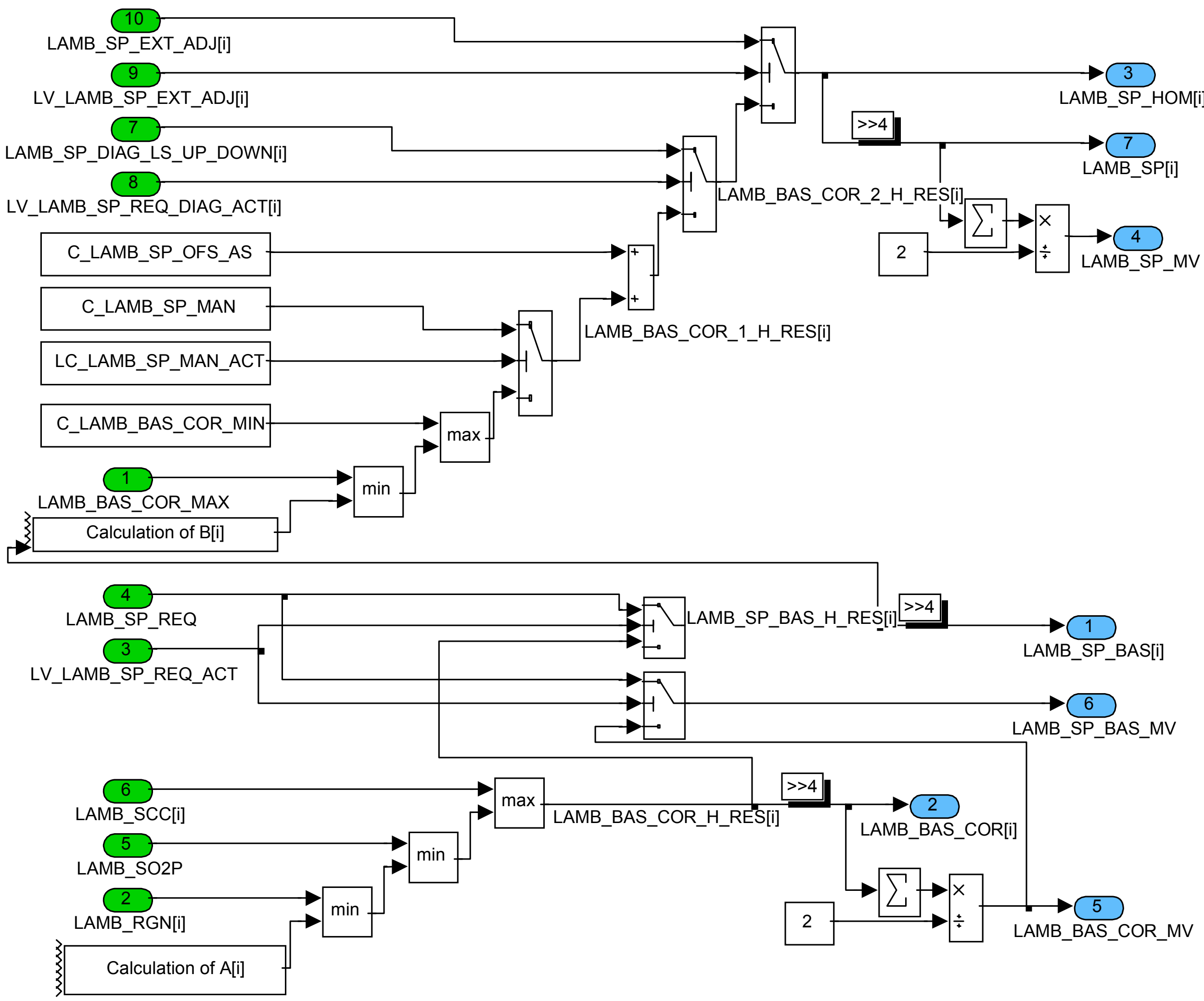


Diagram 4: Main flow

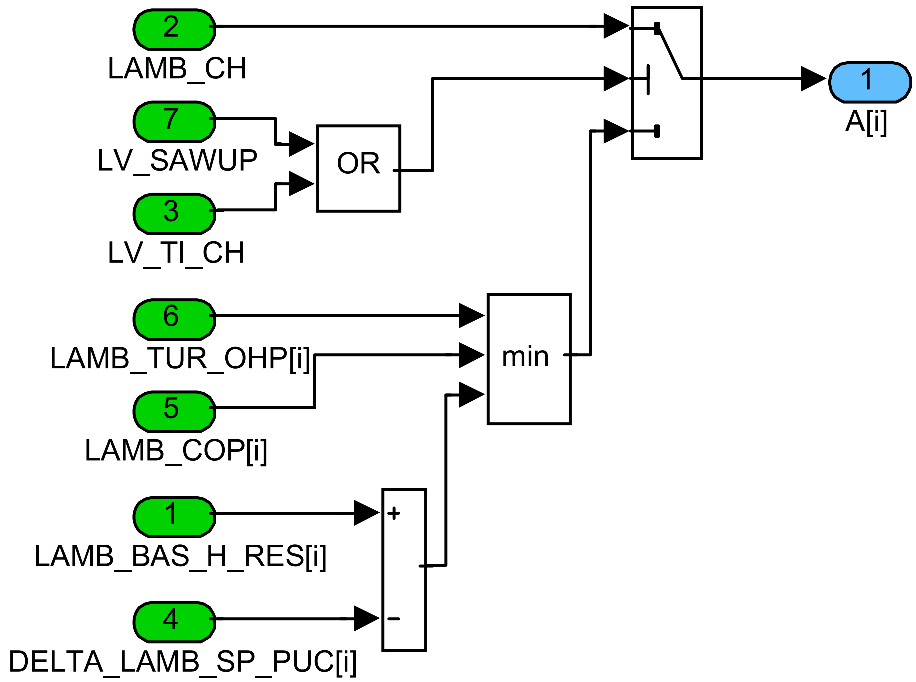


Diagram 5: Calculation of A[i]

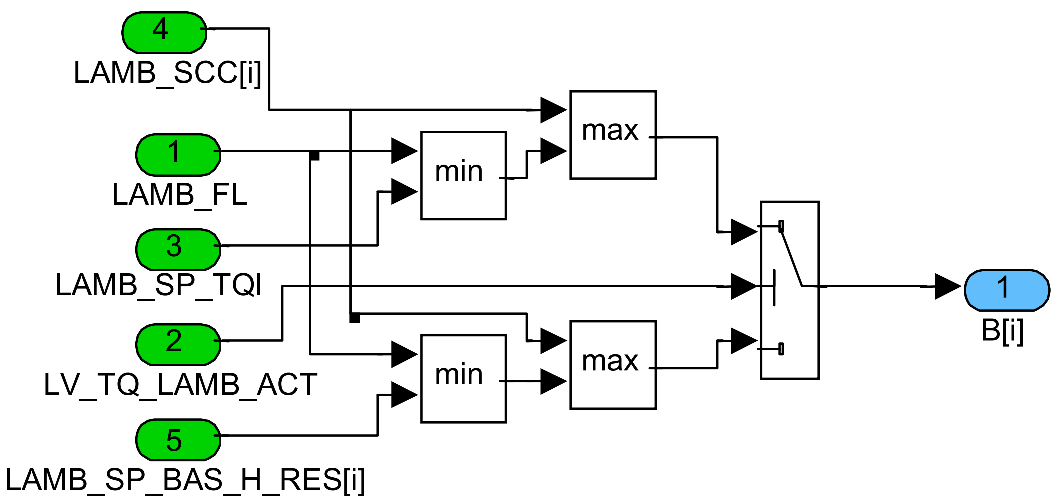


Diagram 6: Calculation of B[i]

Application conditions:

Activation: LV_ST_END = 1

Deactivation: -

Initialisation: at Reset or LV_ES = 0 -> 1: LAMB_SP[i] = 1
LAMB_SP_HOM[i] = 1
LAMB_BAS_COR[i] = 1

Recurrence: 10 ms

Formula section:

To simplify the survey, two intermediate values A[i] and B[i] are calculated. They have the high resolution (6.10E-5), but they are neither output nor visible.

7.56.1 Coordination of intermediate value A[i]:

The lambda request catalyst heating (LAMB_CH) is inserted if one of the two values LV_TI_CH=1 or LV_SAWUP=1.

In the other case, the minimal choice between basic lambda LAMB_BAS_H_RES[i] (including pull fuel cut-off (PUC) correction), LAMB_TUR_OHP[i] and LAMB_COP[i] for turbo charger respectively catalyst overheating prevention is chosen.

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

IF LV_TI_CH = 1
OR
LV_SAWUP = 1

THEN INTERMEDIATE_VALUE_A [i] = LAMB_CH

ELSE INTERMEDIATE_VALUE_A[i]
= MIN (LAMB_TUR_OHP[i], LAMB_COP[i], (LAMB_BAS_H_RES[i] –
DELTA_LAMB_SP_PUC[i]))

ENDIF
ENDFOR

7.56.2 Coordination of LAMB_BAS_COR[i] and it's mean value LAMB_BAS_COR_MV

The coordination handles the different lambda requirements with priority claim via minimum and maximum selections. At first, the smaller value between catalyst regeneration request (LAMB_RGN[i]) and the intermediate value A[i] is selected and entered in a second minimum selection with LAMB_SO2P for catalyst desulfation. Next is a maximum selection between the value selected before and the lambda request for single cylinder cut off LAMB_SCC[i]. The finally chosen value (shifted to low resolution) is a lambda output for TQI_SP_MAF calculation called LAMB_BAS_COR[i] .

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

LAMB_BAS_COR_H_RES[i] = MAX(LAMB_SCC[i], MIN(LAMB_SO2P, MIN(LAMB_RGN[i], INTERMEDIATE_VALUE_A[i])))

LAMB_BAS_COR [i] = LAMB_BAS_COR_H_RES[i] (>>4)

ENDFOR

For two cylinder banks (NC_CBK_EX_NR = 2) the mean value LAMB_BAS_COR is calculated:

LAMB_BAS_COR_MV = (LAMB_BAS_COR[1] + LAMB_BAS_COR[2]) / 2

For one bank solutions (NC_CBK_EX_NR = 1):

LAMB_BAS_COR_MV = LAMB_BAS_COR[1]

7.56.3 Calculation of LAMB_SP_BAS[i]

The input variable LV_LAMB_SP_REQ_ACT indicates whether there is a lambda setpoint request active or not. If the flag is active (LV_LAMB_SP_REQ_ACT = 1), LAMB_SP_BAS_H_RES[i] is set to the input value LAMB_SP_REQ, else LAMB_SP_BAS_H_RES[i] is equal to LAMB_BAS_COR_H_RES[i].

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

IF LV_ LAMB_SP_REQ_ACT = 1

THEN LAMB_SP_BAS_H_RES[i] = LAMB_SP_REQ (<<4)

ELSE LAMB_SP_BAS_H_RES[i] = LAMB_BAS_COR_H_RES[i]

ENDIF

LAMB_SP_BAS[i] = LAMB_SP_BAS_H_RES[i] (>>4)

ENDFOR

IF LV_ LAMB_SP_REQ_ACT = 1

THEN LAMB_SP_BAS_MV = LAMB_SP_REQ

ELSE LAMB_SP_BAS_MV = LAMB_BAS_COR_MV

ENDIF

7.56.4 Calculation of the intermediate value B[i]

In case of transition from hom. mode to hom. lean mode or back, the bit LV_TQ_LAMB_ACT is set to 1, and LAMB_SP_TQI is applied (if Single Cylinder Shut Off or Full Load Enrichment is not active).

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

IF LV_TQ_LAMB_ACT = 1

THEN INTERMEDIATE_VALUE_B[i] = MAX(LAMB_SCC[i], MIN(LAMB_FL, LAMB_SP_TQI))

ELSE INTERMEDIATE_VALUE_B[i] = MAX(LAMB_SCC[i], MIN(LAMB_FL, LAMB_SP_BAS_H_RES[i])

ENDIF

ENDFOR

7.56.5 Calculation of LAMB_BAS_COR_1_H_RES[i]

The intermediate value B[i] is checked, if a minimum or maximum threshold is exceeded. The minimum threshold is a calibratable constant and the maximum threshold is the input from an external module. After that, a manual application value C_LAMB_SP_MAN can be activated by a switch.

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

IF LC_LAMB_SP_MAN_ACT = 1

THEN LAMB_BAS_COR_1_H_RES[i] = C_LAMB_SP_MAN

ELSE LAMB_BAS_COR_1_H_RES[i] = MAX (C_LAMB_BAS_COR_MIN, MIN(LAMB_BAS_COR_MAX, INTERMEDIATE_VALUE_B[i]))

ENDIF

ENDFOR

7.56.6 Lambda coordination outputs LAMB_SP[i], LAMB_SP_H_RES[i] and the mean value LAMB_SP_MV

If the Lambda Sensor Test Function ist active (LV_LAMB_SP_REQ_DIAG_ACT[i] = 1), the output of the Lambda Coordination ist set to LAMB_SP_DIAG_LS_UP_DOWN[i].

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

(1) IF LV_LAMB_SP_REQ_DIAG_ACT[i] = 1

(1) THEN LAMB_BAS_COR_2_H_RES[i] = LAMB_SP_DIAG_LS_UP_DOWN[i]

(1) ELSE LAMB_BAS_COR_2_H_RES[i] = LAMB_BAS_COR_1_H_RES[i] + C_LAMB_SP_OFS_AS

(1) ENDIF

(2) IF LV_LAMB_SP_EXT_ADJ[i] = 1

(2) THEN LAMB_SP_HOM[i] = LAMB_SP_EXT_ADJ[i]

(2) ELSE LAMB_SP_HOM[i] = LAMB_BAS_COR_2_H_RES[i]

(2) ENDIF

LAMB_SP[i] = LAMB_SP_HOM[i] (>>4)

ENDFOR

For two cylinder banks (NC_CBK_EX_NR = 2) the mean value LAMB_SP_MV is calculated:

LAMB_SP_MV = (LAMB_SP[1] + LAMB_SP[2]) / 2

For one bank solutions (NC_CBK_EX_NR = 1):

LAMB_SP_MV = LAMB_SP[1]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LAMB_BAS_COR_MIN	1	0...7FFFH	0....1.999939	6.1035E-5	-
minimal value for lambda setpoint					
LC_LAMB_SP_MAN_ACT	1	0...1H	0...1	1	-
logical variable switch to manual lambda setpoint					
C_LAMB_SP_MAN	1	0...7FFFH	0....1.999939	6.1035E-5	-
manual lambda setpoint					
C_LAMB_SP_OFS_AS	1	C000...3FFFH	-1...0.999969	6.1035E-5	-
calibrateable offset for lambda setpoint					

7.57 Catalyst enrichment function

7.57.1 Calculation of MAF integral during and after PUC phase

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_INT_PUC	V/O	0...FFFFH	0...1820.4167	0.0278	g
integral of mass air flow while active PUC					
MAF_INT_PUE	V/O	0...FFFFH	0...1820.4167	0.0278	g
integral of mass air flow while PUC is not active					

Input data:

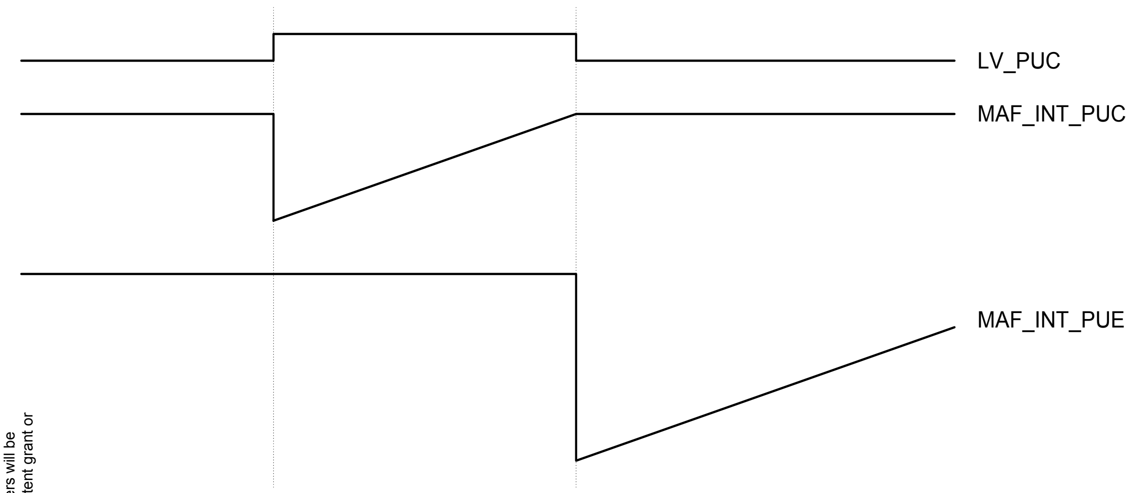
LV_PUC	MAF_CYL	LV_ST_END	LV_IGK

FUNCTION DESCRIPTION:

General information:

During pull fuel cutoff phase the mass air flow integral MAF_INT_PUC is calculated. This integral is used by the catalyst purge function for pre and main catalyst and by diagnosis functions for plausibility checks. After PUC the integral MAF_INT_PUE is calculated.

Signal flow diagram:



Description:

The calculation of MAF_INT_PUC is executed every TA as long as the operation status PUC is fulfilled (LV_PUC = 1). MAF_INT_PUC is initialized by 0 at each rising edge of LV_PUC.

MAF_INT_PUE is calculated when LV_PUC is not set and is initialized by 0 at each falling edge of LV_PUC.

Application conditions:

Initialisation: at LV_IGK = 0 -> 1 and at ECU reset
MAF_INT_PUC = 0
MAF_INT_PUE = 0

Recurrence: TA = 20 ms

Activation: LV_ST_END = 1

Deactivation: -

Formula section:

IF LV_PUC = 0 → 1 **THEN**
  MAF_INT_PUC = 0
ENDIF

IF LV_PUC = 1 → 0 **THEN**
  MAF_INT_PUE = 0
ENDIF

IF LV_PUC = 1 THEN
  MAF_INT_PUC = MAF_INT_PUC + MAF_CYL * NC_FAC_MAF_INT_20
ELSE
  MAF_INT_PUE = MAF_INT_PUE + MAF_CYL * NC_FAC_MAF_INT_20
ENDIF

NC_FAC_MAF_INT_20 = TA / 3.6 = 0.02 s / 3.6

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_FAC_MAF_INT_20	1	0...FFFFH	0...0.892	1.362E-5	s
unit adaptation factor [kg/h] → [g] (set to 0.02 s / 3.6)					

7.57.2 Catalyst purge function for pre and main catalyst

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ACT_INT_PUC_i	V/O	0...1H	0...1	1	-
state of the enrichment function for pre and main cat balancing					
LV_PCAT_PURGE_PUC_i	V/O	0...1H	0...1	1	-
state of the enrichment function for pre cat balancing after PUC phase					
LV_PCAT_PURGE_SA_i	V/O	0...1H	0...1	1	-
state of the enrichment function for pre cat balancing after SA phase					
LV_PCAT_PURGE_HST_i	V/O	0...1H	0...1	1	-
state of the enrichment function for pre cat balancing after HST phase					
LV_PCAT_PURGE_CH_i	V/O	0...1H	0...1	1	-
state of the enrichment function for pre cat balancing after CH phase					
MAF_INT_ACT_PUC_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
value of the mass air flow integral for the function deactivation					
MAF_INT_CAT_LOAD_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
measure of the purge amount of the pre catalyst					
LV_MAIN_CAT_PURGE_i	V/O	0...1H	0...1	1	-
state of the enrichment function for main cat balancing					
MAF_INT_CAT_LOAD_MAIN_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
measure of the purge amount of the main catalyst					
DELTA_LAMB_SP_PUC_i	V/O	0...80H	0...0.125	9.766E-4	-
lambda controller set point shift					
LV_VLS_DOWN_PUC_ACT_MAX_i	V	0...1H	0...1	1	-
logical indicating that downstream sensor signal is/was below calibration limit					
MAF_INT_PURGE_REST_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
estimated rest integral for catalyst purge					
LV_DOWN_RAMP_ACT_i	V	0...1H	0...1	1	-
flag indicating that down ramping from lambda set point shift to zero is active					
MAF_INT_SAMPLE_STEP_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
MAF integral to be added or subtracted each sample step					
FAC_ADD_PCAT_MAIN_CAT_PUC_i	V	0...FFH	0...3.984375	0.0156	-
additional weighting factor for IP_FAC_PCAT_MAIN_CAT_PUC					
LV_PUC_TRIG_i	V	0...1H	0...1	1	-
trigger bit for cat purge after PUC (needed if transition to LV_PUC = 0 but other condition are not yet fulfilled)					
MAF_INT_TOT_PURGE_i	V	0...80 0000H	0...1820.4444	2.1701E-4	g
total MAF integral for pre and main cat purge					
QUO_MAF_INT_PURGE_i	V	0...FFH	0...1	0.004	-
quotient: current MAF rest integral / total MAF integral					

Input data:

LV_PUC	LV_ST_END	LV_SAWUP	LV_TI_CH
CAT_DIAG_i	TEMP_MAIN_CAT_i	TCO_ST	LV_ACT_INT_PUC_AFL_i
MAF_INT_PUC	MAF_CYL	LV_INH_INT_PUC_i	STATE_LSH_DOWN_i
LV_IND_FCUT	NC_FAC_MAF_INT_20	LV_ERR_LS_DOWN_i	TEMP_PCAT_i
LV_IGK			

FUNCTION DESCRIPTION:

General information:

The catalyst purge function realizes a fast adjustment of optimized conditions for a high catalyst efficiency after a transition from pull fuel cutoff (PUC) or other lean air fuel mixture conditions to stoichiometric conditions. That requires a fuel enrichment in order to reduce the oxygen content inside the catalyst. The degree of the fuel enrichment depends on the oxygen and the NO_2 content in the catalyst.

The different purge cases after lean air fuel mixture condition, that are described within this function, are catalyst purge:

- after pull fuel cutoff (PUC),
- after secondary air injection (SA),
- after hot start (HST),
- catalyst heating (CH) with lean air fuel mixture condition,
- that is requested from other functions.

The function takes into account a cat configuration with pre and main catalyst. During pre cat purge a measure for the amount of the main cat purge is calculated. By means of calibration data the main cat purge can also be switched off (all values within IP_FAC_PCAT_MAIN_CAT_PUC must be set to zero).

For the pre cat purge a calculated amount of MAF (in case of PUC) or a calibrated amount of MAF (in case of SA, HST, CH) must pass through the catalyst. During this phase a measure for the amount of the main cat purge is determined. The requested cat purge is an exception because the here described function only reacts on the request and sets the lambda set point shift. The activation and deactivation of the cat purge is localized in the function that sends the request.

All cat purge functions – except the requested one – use a ramp to increase the lambda set point shift and to decrease it in case the main cat purge was activated. This is due to driveability. Both ramps can be switched off by setting the calibration data to its maximum value.

If two separate catalyst systems are concerned then

i = 1, for cylinder bank 1,

i = 2, for cylinder bank 2,

otherwise

i = 1.

Signal flow diagram:

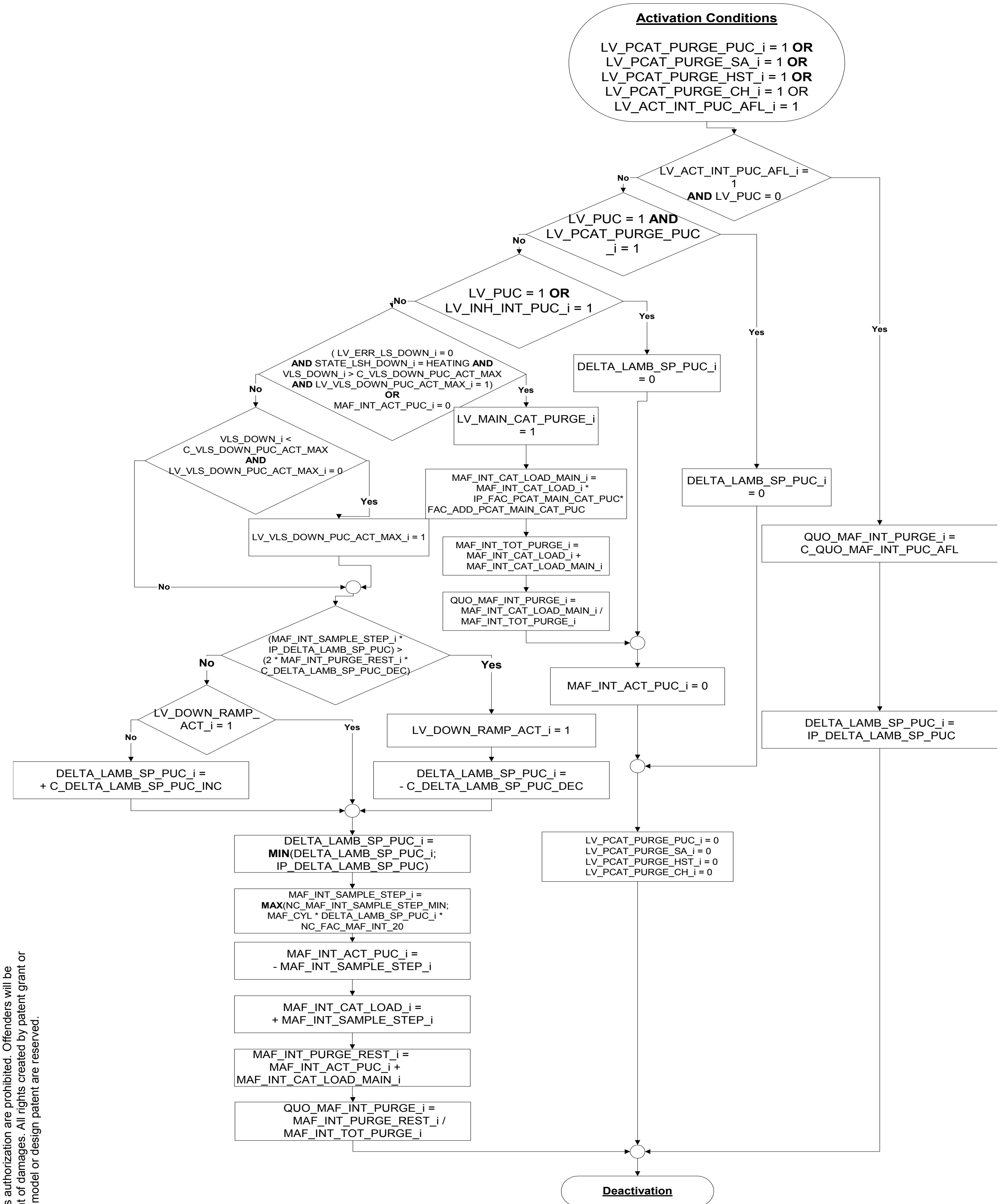


Figure 97: Flow chart for pre cat purge / in case of requested purge pre and main cat purge

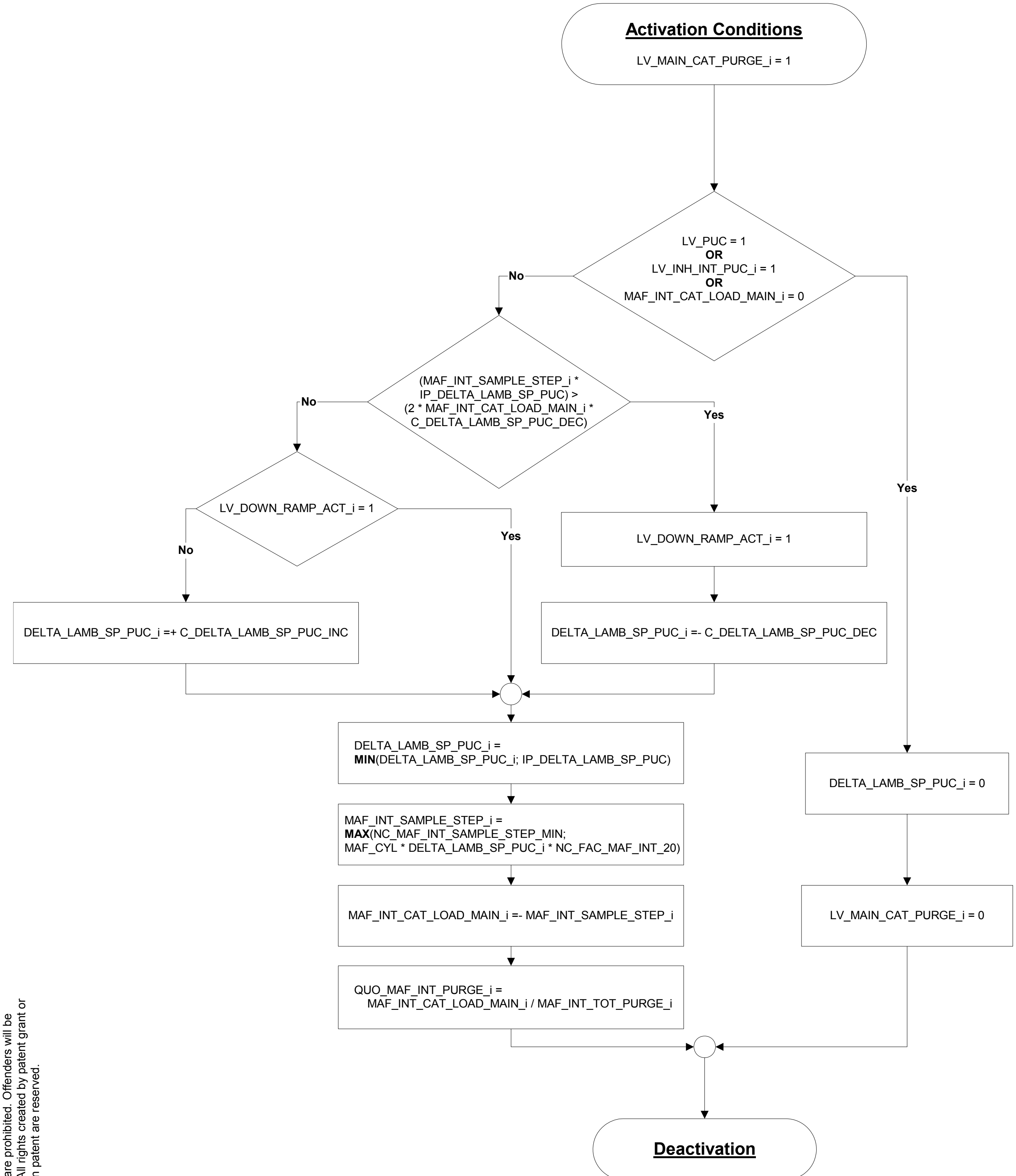


Figure 98: Flow chart for main cat purge

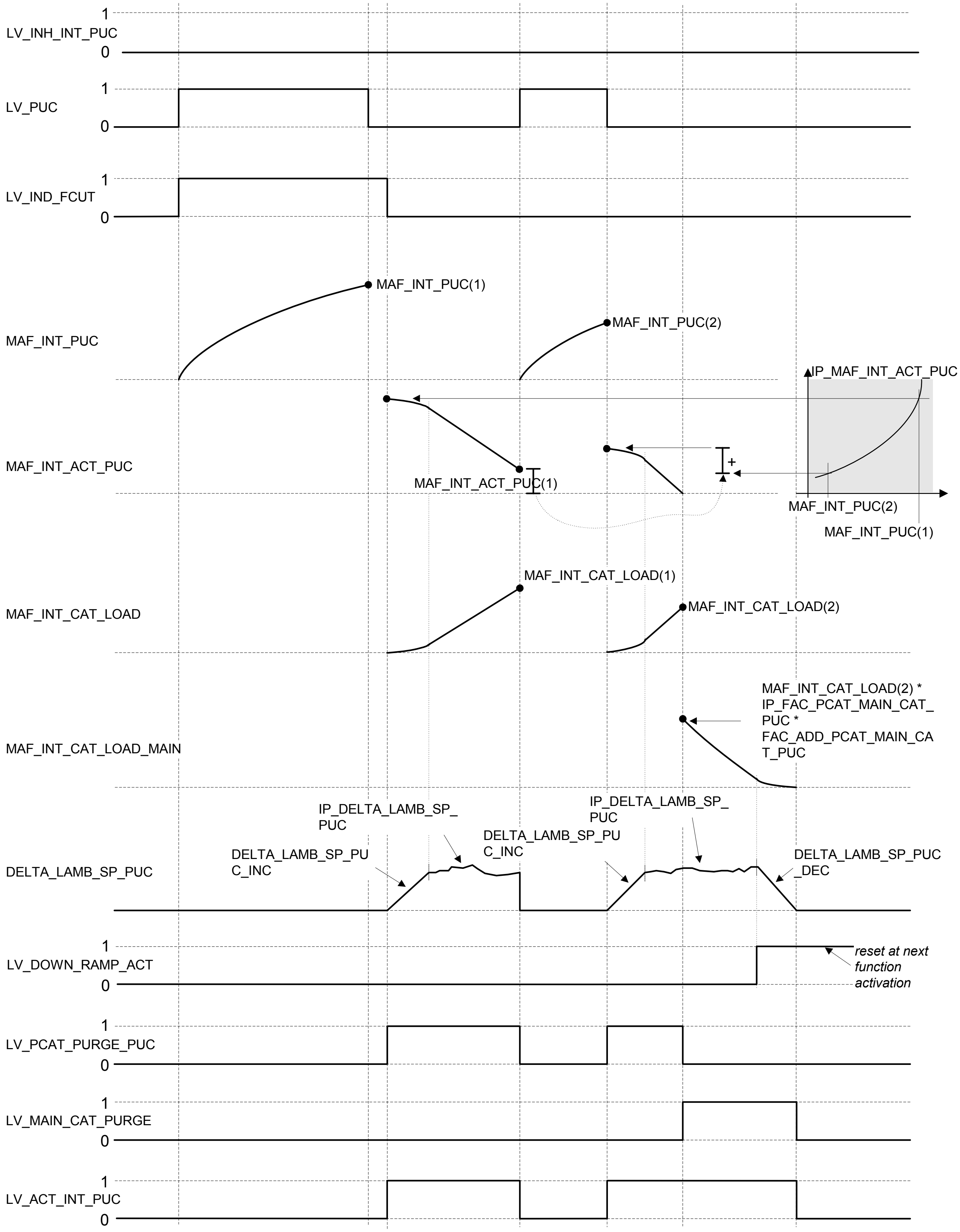

1
LV_INH_INT_PUC
0
1
LV_PUC
0
1
LV_IND_FCUT
0
MAF_INT_PUC(1)
MAF_INT_PUC
MAF_INT_PUC(2)
IP_MAF_INT_ACT_PUC
MAF_INT_ACT_PUC
MAF_INT_ACT_PUC(1)
+
MAF_INT_PUC(2)
MAF_INT_PUC(1)
MAF_INT_CAT_LOAD(1)
MAF_INT_CAT_LOAD
MAF_INT_CAT_LOAD(2)
MAF_INT_CAT_LOAD(2) * IP_FAC_PCAT_MAIN_CAT_PUC * FAC_ADD_PCAT_MAIN_CAT_PUC
MAF_INT_CAT_LOAD_MAIN
IP_DELTA_LAMB_SP_PUC
IP_DELTA_LAMB_SP_PUC
DELTA_LAMB_SP_PUC_INC
DELTA_LAMB_SP_PUC_INC
DELTA_LAMB_SP_PUC
DELTA_LAMB_SP_PUC_DEC
1
LV_DOWN_RAMP_ACT
0
reset at next function activation
1
LV_PCAT_PURGE_PUC
0
1
LV_MAIN_CAT_PURGE
0
1
LV_ACT_INT_PUC
0

Figure 99: Pre and main cat purge exemplarily shown for "after PUC purge"

Description:

Activation:

The general activation conditions of the cat purge function (pre and main cat purge) are summarized by the logical LV_INH_INT_PUC_i. It must be set to 0 as a pre condition to start the enrichment function for all lean air fuel mixture conditions - except for requested cat purge.

LV_PCAT_PURGE_PUC_i activates the function for cat purge after PUC phase (PUC). The logical is set to 1 if no inhibition is detected (LV_INH_INT_PUC_i = 0) and additionally the MAF integral calculated during PUC (MAF_INT_PUC) has exceeded the calibration limit C_MAF_INT_PUC_MIN or MAF_INT_ACT_PUC_i is above 0. This latter condition is an indication that the previous cat purge function after PUC was interrupted before the truncation condition MAF_INT_ACT_PUC_i = 0 was reached. Furthermore a transition from PUC phase (LV_PUC = 1 → 0) must have been recognized and therefore LV_PUC_TRIG must be set to 1 and no cylinder must be shut off (LV_IND_FCUT = 0).

LV_PCAT_PURGE_SA_i activates the function for cat purge after secondary air (SA) injection. The logical is set to 1 if no inhibition is detected (LV_INH_INT_PUC_i = 0) and the transition from "SA active" to "SA not active" (LV_SAWUP = 1 → 0) occurs. As cat purge after PUC has the highest priority LV_PCAT_PURGE_PUC_i and LV_MAIN_CAT_PURGE_i must both not be set. No cylinder must be shut off (LV_IND_FCUT = 0).

LV_PCAT_PURGE_HST_i activates the function for cat purge after hot start (HST). The logical is set to 1 if no inhibition is detected (LV_INH_INT_PUC_i = 0) and the end of a hot start is recognized (LV_ST_END = 0 → 1 and TCO_ST >= C_TCO_THD_HST). As cat purge after hot start has the lowest priority the logical variables LV_PCAT_PURGE_PUC_i, LV_PCAT_PURGE_SA_i, LV_PCAT_PURGE_CH_i and LV_MAIN_CAT_PURGE_i must not be set. No cylinder must be shut off (LV_IND_FCUT = 0).

LV_PCAT_PURGE_CH_i activates the function for cat purge after catalyst heating (CH) under lean conditions. The logical is set to 1 if no inhibition is detected (LV_INH_INT_PUC_i = 0) and the end of the heating phase (LV_TI_CH = 1 → 0) is recognized. As cat purge after PUC has the highest priority LV_PCAT_PURGE_PUC_i and LV_MAIN_CAT_PURGE_i must both not be set. Cat purge after SA and cat purge after CH have the same priority. No cylinder must be shut off (LV_IND_FCUT = 0).

LV_ACT_INT_PUC_AFL_i indicates that cat purge is requested by another function. The responsibility for activating cat purge is at the calling function.

Initialization:

If the flag for ignition key LV_IGK switches from 0 to 1 or at ECU reset the variables MAF_INT_ACT_PUC_i, DELTA_LAMB_SP_PUC_i, LV_ACT_INT_PUC_i, LV_PCAT_PURGE_PUC_i, LV_PCAT_PURGE_SA_i, LV_PCAT_PURGE_HST_i, LV_PCAT_PURGE_CH_i, LV_MAIN_CAT_PURGE_i and some internal variables (see formula section) must be set to 0.

For all different cases of catalyst purge – except for requested purge - the following variables are initialized at function activation as follows:

LV_DOWN_RAMP_ACT_i = 0

LV_VLS_DOWN_PUC_ACT_MAX_i = 0

MAF_INT_CAT_LOAD_i = 0

MAF_INT_SAMPLE_STEP_i = 0

MAF_INT_PURGE_REST_i = NC_MAF_INT_PURGE_MAX

QUO_MAF_INT_PURGE_i = 1 (FF hex)

In case of cat purge after PUC phase MAF_INT_ACT_PUC_i is initialized by IP_MAF_INT_ACT_PUC * IP_FAC_CAT_EFF_PUC + MAF_INT_ACT_PUC_i. This latter value is the remaining MAF integral of a previous purge function activation. MAF_INT_ACT_PUC_i is furthermore limited to the maximum value of C_MAF_INT_ACT_PUC_MAX. The factor FAC_ADD_PCAT_MAIN_CAT_PUC_i, that is an additional weighting factor for IP_FAC_PCAT_MAIN_CAT_PUC, is initialized by 1, because the map is calibrated for cat purge after PUC phase and therefore no weighting is necessary. IP_FAC_PCAT_MAIN_CAT_PUC is the temperature dependent volume ratio between pre and main cat. This factor map is used for main cat purge and must be calibrated with 0 in case that no main cat is implemented or no main cat purge is desired.

In case of cat purge after secondary air injection MAF_INT_ACT_PUC_i is initialized by C_MAF_INT_PCAT_PURGE_SA * IP_FAC_CAT_EFF_PUC. The factor FAC_ADD_PCAT_MAIN_CAT_PUC_i is initialized by C_FAC_ADD_PCAT_MAIN_CAT_PUC_SA. This latter calibration data is a weighting factor in order to realize deviations from the map that is given for cat purge after PUC phase. By means of this data main cat purge can also be switched off only for cat purge after secondary air injection.

In case of cat purge after hot start MAF_INT_ACT_PUC_i is initialized by C_MAF_INT_PCAT_PURGE_HST * IP_FAC_CAT_EFF_PUC. The factor FAC_ADD_PCAT_MAIN_CAT_PUC_i is initialized by C_FAC_ADD_PCAT_MAIN_CAT_PUC_HST. By means of this latter calibration data main cat purge can also switched off only for cat purge after hot start.

In case of cat purge after lean catalyst heating MAF_INT_ACT_PUC_i is initialized by C_MAF_INT_PCAT_PURGE_CH * IP_FAC_CAT_EFF_PUC. The factor FAC_ADD_PCAT_MAIN_CAT_PUC_i is initialized by C_FAC_ADD_PCAT_MAIN_CAT_PUC_CH. By means of this latter calibration data main cat purge can also switched off only for cat purge after catalyst heating.

After MAF_INT_ACT_PUC_i was calculated MAF_INT_CAT_LOAD_MAIN_i will be initialized for all different purge cases by MAF_INT_ACT_PUC_i * IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i. The calculation is executed by using the 16 high-order bits of MAF_INT_ACT_PUC_i (right shift of 7) and rounding the lowest of these bits (((MAF_INT_ACT_PUC_i>>6)+1)>>1). So the multiplication can be executed with a 16 bit operation.

With this calculation a pre estimation of the main cat purge amount is done. At the end of pre cat purge the main cat purge amount is updated with the real MAF integral during pre cat purge (this can be different from MAF_INT_ACT_PUC_i in case of termination via downstream sensor). Finally the total purge integral is calculated (MAF_INT_TOT_PURGE_i).

Function description of pre cat purge:

Figure 97 describes the functionality for pre cat purge after PUC, SA, HST, CH and for the requested cat purge. Figure 98 shows the main cat purge. In Figure 99 the function sequence is given for cat purge after PUC with and without interruption of a new PUC phase. The function sequence for the other cases is similar; only the determination of the starting value for MAF_INT_ACT_PUC_i is different (see initialization).

Firstly the function detects whether a cat purge is requested by another function and PUC is not active. In this case the lambda set point shift is applied as long as LV_ACT_INT_PUC_AFL_i is set to 1.

If the function was activated because of cat purge after PUC (LV_PCAT_PURGE_PUC_i = 1) and a new PUC phase is recognized (LV_PUC = 1), the lambda set point shift is set to 0 and the activation conditions are all set to 0. The remaining MAF integral MAF_INT_ACT_PUC_i is not set to 0 in order to use it for next cat purge after PUC.

If one of the inhibition conditions is fulfilled (that means LV_INH_INT_PUC_i = 1) or a new PUC phase is recognized (in this case LV_PCAT_PURGE_PUC_i = 0), the lambda set point shift is set to 0, the remaining MAF integral MAF_INT_ACT_PUC_i is set to 0 and the activation conditions are all set to 0.

If the MAF integral MAF_INT_ACT_PUC_i has reached 0 or if the downstream sensor detects a rich mixture (VLS_DOWN_i > C_VLS_DOWN_PUC_ACT_MAX), the activation conditions for pre cat purge are set to 0 and the activation condition for main cat purge is set to 1. The purge amount for main cat purge (MAF_INT_CAT_LOAD_MAIN_i) is calculated based on the real pre cat purge MAF integral (MAF_INT_CAT_LOAD_i). Also this multiplication can be executed with a 16 bit operation by using only the 16 high-order bits and rounding of MAF_INT_CAT_LOAD_i as described above. The total purge integral (MAF_INT_TOT_PURGE_i) is updated and the ratio between of current rest integral and total MAF integral is calculated (QUO_MAF_INT_PURGE_i). The MAF integral MAF_INT_ACT_PUC_i is set to 0 (in case of sensor has detected rich mixture it could be still above 0).

The downstream sensor signal is only evaluated if the sensor is in heating mode and no error is detected (STATE_LSH_DOWN_i = HEATING and LV_ERR_LS_DOWN_i = 0). The detection of a rich mixture via downstream sensor is furthermore only activated if the sensor voltage was at least once below C_VLS_DOWN_PUC_ACT_MAX since the function was activated. This is indicated by LV_VLS_DOWN_PUC_ACT_MAX. It is set to 1 if the voltage was below the threshold. Otherwise it is 0.

Based of the current MAF_CYL and of the remaining MAF integral the remaining time can be calculated and compared to the time needed to ramp down the lambda set point shift. If the remaining time gets below half of the down ramp time the down ramping of DELTA_LAMB_SP_PUC_i is initiated. The down ramping is realized by subtracting the decrement C_DELTA_LAMB_SP_PUC_DEC from the current set point shift and taking the minimum of the result and the map value.

The remaining time is calculated by:

$t_{remaining}$ = MAF_INT_PURGE_REST_i / (MAF_CYL * IP_DELTA_LAMB_SP_PUC)

The ramp down time is calculated by:

t_{ramp_down} = DELTA_LAMB_SP_PUC_i * TA / C_DELTA_LAMB_SP_PUC_DEC

The condition $t_{remaining} < 0.5 * t_{ramp\_down}$ is realized by:

((MAF_CYL * IP_DELTA_LAMB_SP_PUC * NC_FAC_MAF_INT_20) * IP_DELTA_LAMB_SP_PUC) > (2 * MAF_INT_PURGE_REST_i * C_DELTA_LAMB_SP_PUC_DEC)

Depending on the condition above the determination of the lambda set point shift (DELTA_LAMB_SP_PUC_i) is executed. The down ramping is realized by a maximum allowed decrease per TA of the lambda shift. The up ramping is realized similar (maximum allowed increase of the shift per TA) if the condition is not fulfilled. If the down ramping was started once the bit LV_DOWN_RAMP_ACT_i is set. So, - by evaluating this flag -, if the down ramping condition is not fulfilled any mode (e.g. change of MAF_CYL) the lambda set point shift is kept constant.

Then the following calculations are carried out:

- calculation of MAF during one sample step (MAF_INT_SAMPLE_STEP_i); if the calculation result is 0 then MAF_INT_SAMPLE_STEP_i is set to NC_MAF_INT_SAMPLE_STEP_MIN;
- decreasing of MAF_INT_ACT_PUC_i by MAF_INT_SAMPLE_STEP_i;
- increasing of MAF_INT_CAT_LOAD_i by MAF_INT_SAMPLE_STEP_i;
- calculation of the remaining MAF integral MAF_INT_PURGE_REST_i;
- calculation of the ratio between current rest integral and total MAF integral QUO_MAF_INT_PURGE_i.

Function description of main cat purge:

If the condition LV_MAIN_CAT_PURGE_i is set to 1 the purge function for the main catalyst is activated. LV_MAIN_CAT_PURGE_i is set to one if the pre cat purge was not interrupted.

For the case that cat purge was requested by another function (via LV_ACT_INT_PUC_AFL_i) the purge of the main cat is included in the functionality for the pre cat. This became necessary because the control of cat purge in this case is outside the here described function.

The main cat purge is executed as long as the MAF integral MAF_INT_CAT_LOAD_MAIN_i has not reached 0. The start value was calculated during pre cat purge. When MAF_INT_CAT_LOAD_MAIN_i has reached 0 the lambda set point shift (DELTA_LAMB_SP_PUC_i) and LV_MAIN_CAT_PURGE_i are both set to 0. DELTA_LAMB_SP_PUC_i should be already 0 but as the determination of the down ramping condition is only a predication the resetting of DELTA_LAMB_SP_PUC_i is done anyway. If the main cat purge is interrupted by a new PUC phase or by the case that the inhibition conditions are getting valid DELTA_LAMB_SP_PUC_i and LV_MAIN_CAT_PURGE_i are also immediately set to 0.

As long as main cat purge is active the following calculations are carried out:

- determination of lambda set point shift as described for pre cat purge; the rest integral is determined directly by using MAF_INT_CAT_LOAD_MAIN_i;
- calculation of MAF during one sample step (MAF_INT_SAMPLE_STEP_i);
  if the calculation result is 0 then MAF_INT_SAMPLE_STEP_i is set to NC_MAF_INT_SAMPLE_STEP_MIN;
- decreasing of MAF_INT_CAT_LOAD_MAIN_i by MAF_INT_SAMPLE_STEP_i;
- calculation of the ratio between current rest integral and total MAF integral QUO_MAF_INT_PURGE_i.

Calculation of LV_ACT_INT_PUC_i:

If one of the activation conditions for pre cat purge or for main cat purge is active the flag LV_ACT_INT_PUC_i must be set to one. The conditions are given with the flags LV_PCAT_PURGE_PUC_i, LV_PCAT_PURGE_SA_i, LV_PCAT_PURGE_HST_i, LV_PCAT_PURGE_CH_i, LV_ACT_INT_PUC_AFL_i and LV_MAIN_CAT_PURGE_i.

Application conditions:

Initialisation:

At LV_IGK = 0 -> 1 and at ECU reset:
MAF_INT_ACT_PUC_i = 0
DELTA_LAMB_SP_PUC_i = 0
LV_ACT_INT_PUC_i = 0
LV_PCAT_PURGE_PUC_i = 0
LV_PCAT_PURGE_SA_i = 0
LV_PCAT_PURGE_HST_i = 0
LV_PCAT_PURGE_CH_i = 0
LV_MAIN_CAT_PURGE_i = 0
LV_PUC_TRIG_i = 0
LV_PUC_OLD_i = LV_PUC
LV_SAWUP_OLD_i = LV_SAWUP
LV_ST_END_OLD_i = LV_ST_END
LV_TI_CH_OLD_i = LV_TI_CH

For CAT purge after PUC phase (LV_PCAT_PURGE_PUC_i = 1):

```
IF LV_PCAT_PURGE_PUC_i = 0 → 1 THEN
   LV_DOWN_RAMP_ACT_i = 0
   LV_VLS_DOWN_PUC_ACT_MAX_i = 0
   MAF_INT_CAT_LOAD_i = 0
   MAF_INT_CAT_LOAD_MAIN_i = 0
   MAF_INT_SAMPLE_STEP_i = 0
   MAF_INT_PURGE_REST_i = NC_MAF_INT_PURGE_MAX
   MAF_INT_ACT_PUC_i = MIN(C_MAF_INT_ACT_PUC_MAX;
                      (IP_MAF_INT_ACT_PUC * IP_FAC_CAT_EFF_PUC + MAF_INT_ACT_PUC_i) )
   FAC_ADD_PCAT_MAIN_CAT_PUC_i = 1
   MAF_INT_CAT_LOAD_MAIN_i =
                    (16 high-order bits with lowest bit rounded of MAF_INT_ACT_PUC_i) *
                    IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i
   MAF_INT_TOT_PURGE_i = MAF_INT_ACT_PUC_i + MAF_INT_CAT_LOAD_MAIN_i
   QUO_MAF_INT_PURGE_i = 1
   LV_PCAT_PURGE_SA_i = 0  /* reset all other purges */
   LV_PCAT_PURGE_HST_i = 0
   LV_PCAT_PURGE_CH_i = 0
   LV_MAIN_CAT_PURGE_i = 0
ENDIF
```

For CAT purge after SA phase (LV_PCAT_PURGE_SA_i = 1):

```
IF LV_PCAT_PURGE_SA_i = 0 → 1 THEN
   LV_DOWN_RAMP_ACT_i = 0
   LV_VLS_DOWN_PUC_ACT_MAX_i = 0
   MAF_INT_CAT_LOAD_i = 0
   MAF_INT_CAT_LOAD_MAIN_i = 0
   MAF_INT_SAMPLE_STEP_i = 0
   MAF_INT_PURGE_REST_i = NC_MAF_INT_PURGE_MAX
   MAF_INT_ACT_PUC_i = C_MAF_INT_PCAT_PURGE_SA * IP_FAC_CAT_EFF_PUC
   FAC_ADD_PCAT_MAIN_CAT_PUC_i = C_FAC_ADD_PCAT_MAIN_CAT_PUC_SA
   MAF_INT_CAT_LOAD_MAIN_i =
                    (16 high-order bits with lowest bit rounded of MAF_INT_ACT_PUC_i) *
                    IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i
   MAF_INT_TOT_PURGE_i = MAF_INT_ACT_PUC_i + MAF_INT_CAT_LOAD_MAIN_i
   QUO_MAF_INT_PURGE_i = 1
   LV_PCAT_PURGE_PUC_i = 0  /* reset all other purges */
   LV_PCAT_PURGE_HST_i = 0
   LV_PCAT_PURGE_CH_i = 0
   LV_MAIN_CAT_PURGE_i = 0
ENDIF
```

For CAT purge after HST phase (LV_PCAT_PURGE_HST_i = 1):

IF LV_PCAT_PURGE_HST_i = 0 → 1 **THEN**
LV_DOWN_RAMP_ACT_i = 0
LV_VLS_DOWN_PUC_ACT_MAX_i = 0
MAF_INT_CAT_LOAD_i = 0
MAF_INT_CAT_LOAD_MAIN_i = 0
MAF_INT_SAMPLE_STEP_i = 0
MAF_INT_PURGE_REST_i = NC_MAF_INT_PURGE_MAX
MAF_INT_ACT_PUC_i = C_MAF_INT_PCAT_PURGE_HST * IP_FAC_CAT_EFF_PUC
FAC_ADD_PCAT_MAIN_CAT_PUC_i = C_FAC_ADD_PCAT_MAIN_CAT_PUC_HST
MAF_INT_CAT_LOAD_MAIN_i =
(*16 high-order bits with lowest bit rounded of* MAF_INT_ACT_PUC_i) *
IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i
MAF_INT_TOT_PURGE_i = MAF_INT_ACT_PUC_i + MAF_INT_CAT_LOAD_MAIN_i
QUO_MAF_INT_PURGE_i = 1
LV_PCAT_PURGE_PUC_i = 0 */* reset all other purges */*
LV_PCAT_PURGE_SA_i = 0
LV_PCAT_PURGE_CH_i = 0
LV_MAIN_CAT_PURGE_i = 0
ENDIF

For CAT purge after CH phase (LV_PCAT_PURGE_CH_i = 1):

IF LV_PCAT_PURGE_CH_i = 0 → 1 **THEN**
LV_DOWN_RAMP_ACT_i = 0
LV_VLS_DOWN_PUC_ACT_MAX_i = 0
MAF_INT_CAT_LOAD_i = 0
MAF_INT_CAT_LOAD_MAIN_i = 0
MAF_INT_SAMPLE_STEP_i = 0
MAF_INT_PURGE_REST_i = NC_MAF_INT_PURGE_MAX
MAF_INT_ACT_PUC_i = C_MAF_INT_PCAT_PURGE_CH * IP_FAC_CAT_EFF_PUC
FAC_ADD_PCAT_MAIN_CAT_PUC_i = C_FAC_ADD_PCAT_MAIN_CAT_PUC_CH
MAF_INT_CAT_LOAD_MAIN_i =
(*16 high-order bits with lowest bit rounded of* MAF_INT_ACT_PUC_i) *
IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i
MAF_INT_TOT_PURGE_i = MAF_INT_ACT_PUC_i + MAF_INT_CAT_LOAD_MAIN_i
QUO_MAF_INT_PURGE_i = 1
LV_PCAT_PURGE_PUC_i = 0 */* reset all other purges */*
LV_PCAT_PURGE_SA_i = 0
LV_PCAT_PURGE_HST_i = 0
LV_MAIN_CAT_PURGE_i = 0
ENDIF

Recurrence: The function is executed every TA = 20 ms

Activation:

For pre cat purge (see Figure 97):
LV_PCAT_PURGE_PUC_i = 1 **OR**
LV_PCAT_PURGE_SA_i = 1 **OR**
LV_PCAT_PURGE_HST_i = 1 **OR**
LV_PCAT_PURGE_CH_i = 1 **OR**
LV_ACT_INT_PUC_AFL_i = 1

For main cat purge (see Figure 98):
LV_MAIN_CAT_PURGE_i = 1

Deactivation:

For pre cat purge:
LV_PCAT_PURGE_PUC_i = 0 **AND**
LV_PCAT_PURGE_SA_i = 0 **AND**
LV_PCAT_PURGE_HST_i = 0 **AND**
LV_PCAT_PURGE_CH_i = 0 **AND**
LV_ACT_INT_PUC_AFL_i = 0

For main cat purge:
LV_MAIN_CAT_PURGE_i = 0

Formula section:

PUC end detection:

```
IF LV_PUC = 0 AND LV_PUC_OLD_i = 1 AND
   ( MAF_INT_PUC > C_MAF_INT_PUC_MIN OR MAF_INT_ACT_PUC_i > 0 ) AND
   LV_INH_INT_PUC_i = 0                        /* no general function inhibition */
THEN
   LV_PUC_TRIG_i = 1
ENDIF
LV_PUC_OLD_i = LV_PUC
```

Set condition for CAT purge after PUC phase (LV_PCAT_PURGE_PUC_i = 1):

```
IF LV_PUC_TRIG_i = 1 AND                       /* end of PUC was detected */
   LV_IND_FCUT = 0
THEN
   LV_PCAT_PURGE_PUC_i = 1
   LV_MAIN_CAT_PURGE_i = 0
   LV_PUC_TRIG_i = 0
   /* here the initialization code for cat purge after PUC phase can be included */
ENDIF
```

Set condition for CAT purge after SA phase (LV_PCAT_PURGE_SA_i = 1):

```
IF LV_SAWUP = 0 AND
   LV_SAWUP_OLD_i = 1 AND                      /* end of SA detected */
   LV_PCAT_PURGE_PUC_i = 0 AND                 /* pre cat purge after PUC not active */
   LV_MAIN_CAT_PURGE_i = 0 AND                 /* no main cat purge active */
   LV_INH_INT_PUC_i = 0 AND                    /* no general function inhibition */
   LV_IND_FCUT = 0                             /* no cylinder shut off */
THEN
   LV_PCAT_PURGE_SA_i = 1
   /* here the initialization code for cat purge after SA phase can be included */
ENDIF
LV_SAWUP_OLD_i = LV_SAWUP
```

Set condition for CAT purge after HST phase (LV_PCAT_PURGE_HST_i = 1):

```
IF LV_ST_END = 1 AND
   LV_ST_END_OLD_i = 0 AND                /* end of start detected */
   TCO_ST >= C_TCO_THD_HST AND            /* HST temperature */
   LV_PCAT_PURGE_PUC_i = 0 AND            /* pre cat purge after PUC not active */
   LV_PCAT_PURGE_SA_i = 0 AND             /* pre cat purge after SA not active */
   LV_PCAT_PURGE_CH_i = 0 AND             /* pre cat purge after CH not active */
   LV_MAIN_CAT_PURGE_i = 0 AND            /* no main cat purge active */
   LV_INH_INT_PUC_i = 0 AND               /* no general function inhibition */
   LV_IND_FCUT = 0                        /* no cylinder shut off */
THEN
   LV_PCAT_PURGE_HST_i = 1
   /* here the initialization code for cat purge after HST phase can be included */
ENDIF
LV_ST_END_OLD_i = LV_ST_END
```

Set condition for CAT purge after CH phase (LV_PCAT_PURGE_CH_i = 1):

```
IF LV_TI_CH = 0 AND LV_TI_CH_OLD_i = 1 AND   /* end of cat heating detected */
   LV_PCAT_PURGE_PUC_i = 0 AND               /* pre cat purge after PUC not active */
   LV_MAIN_CAT_PURGE_i = 0 AND               /* no main cat purge active */
   LV_INH_INT_PUC_i = 0 AND                  /* no general function inhibition */
   LV_IND_FCUT = 0                           /* no cylinder shut off */
THEN
   LV_PCAT_PURGE_CH_i = 1
   /* here the initialization code for cat purge after CH phase can be included */
ENDIF
LV_TI_CH_OLD_i = LV_TI_CH
```

Pre cat purge function:

```
IF LV_ACT_INT_PUC_AFL_i = 1 AND LV_PUC = 0
THEN
   QUO_MAF_INT_PURGE_i = C_QUO_MAF_INT_PUC_AFL
   DELTA_LAMB_SP_PUC_i = IP_DELTA_LAMB_SP_PUC
ELSE
   IF LV_PUC = 1 AND LV_PCAT_PURGE_PUC_i = 1
   THEN
      DELTA_LAMB_SP_PUC_i = 0
      LV_PCAT_PURGE_PUC_i = 0
      LV_PCAT_PURGE_SA_i = 0
      LV_PCAT_PURGE_HST_i = 0
      LV_PCAT_PURGE_CH_i = 0
   ELSE
      IF LV_PUC = 1 OR LV_INH_INT_PUC_i = 1
      THEN
         MAF_INT_ACT_PUC_i = 0
         DELTA_LAMB_SP_PUC_i = 0
         LV_PCAT_PURGE_PUC_i = 0
         LV_PCAT_PURGE_SA_i = 0
         LV_PCAT_PURGE_HST_i = 0
         LV_PCAT_PURGE_CH_i = 0
      ELSE
         IF ( LV_ERR_LS_DOWN_i = 0 AND STATE_LSH_DOWN_i = HEATING AND
               VLS_DOWN_i > C_VLS_DOWN_PUC_ACT_MAX AND
               LV_VLS_DOWN_PUC_ACT_MAX_i = 1 ) OR MAF_INT_ACT_PUC_i = 0
         THEN
            LV_MAIN_CAT_PURGE_i = 1
            MAF_INT_CAT_LOAD_MAIN_i =
                        (16 high-order bits with lowest bit rounded of MAF_INT_CAT_LOAD_i) *
                        IP_FAC_PCAT_MAIN_CAT_PUC_i * FAC_ADD_PCAT_MAIN_CAT_PUC_i
            MAF_INT_TOT_PURGE_i = MAF_INT_CAT_LOAD_i + MAF_INT_CAT_LOAD_MAIN_i
            QUO_MAF_INT_PURGE_i = MAF_INT_CAT_LOAD_MAIN_i / MAF_INT_TOT_PURGE_i
            MAF_INT_ACT_PUC_i = 0
            LV_PCAT_PURGE_PUC_i = 0
            LV_PCAT_PURGE_SA_i = 0
            LV_PCAT_PURGE_HST_i = 0
            LV_PCAT_PURGE_CH_i = 0
         ELSE
            IF VLS_DOWN_i < C_VLS_DOWN_PUC_ACT_MAX AND
               LV_VLS_DOWN_PUC_ACT_MAX_i = 0
            THEN
               LV_VLS_DOWN_PUC_ACT_MAX_i = 1
            ENDIF
```

```
            IF ( MAF_INT_SAMPLE_STEP_i * IP_DELTA_LAMB_SP_PUC ) >
               ( 2 * MAF_INT_PURGE_REST_i * C_DELTA_LAMB_SP_PUC_DEC )
            THEN
               LV_DOWN_RAMP_ACT_i = 1
               DELTA_LAMB_SP_PUC_i =
                                     DELTA_LAMB_SP_PUC_i - C_DELTA_LAMB_SP_PUC_DEC
            ELSE
               IF LV_DOWN_RAMP_ACT_i = 0 THEN
                  DELTA_LAMB_SP_PUC_i =
                                     DELTA_LAMB_SP_PUC_i + C_DELTA_LAMB_SP_PUC_INC
               ENDIF
            ENDIF
            IF DELTA_LAMB_SP_PUC_i > IP_DELTA_LAMB_SP_PUC
            THEN
               DELTA_LAMB_SP_PUC_i = IP_DELTA_LAMB_SP_PUC
            ENDIF
            MAF_INT_SAMPLE_STEP_i =
                                   MAF_CYL * DELTA_LAMB_SP_PUC_i * NC_FAC_MAF_INT_20
            IF MAF_INT_SAMPLE_STEP_i = 0
            THEN
               MAF_INT_SAMPLE_STEP_i = NC_MAF_INT_SAMPLE_STEP_MIN
            ENDIF
            MAF_INT_ACT_PUC_i = MAF_INT_ACT_PUC_i - MAF_INT_SAMPLE_STEP_i
            MAF_INT_CAT_LOAD_i = MAF_INT_CAT_LOAD_i + MAF_INT_SAMPLE_STEP_i
            MAF_INT_PURGE_REST_i = MAF_INT_ACT_PUC_i + MAF_INT_CAT_LOAD_MAIN_i
            QUO_MAF_INT_PURGE_i = MAF_INT_PURGE_REST_i / MAF_INT_TOT_PURGE_i
         ENDIF
      ENDIF
   ENDIF
ENDIF
```

Main cat purge function:

```
IF LV_PUC = 1 OR LV_INH_INT_PUC_i = 1OR MAF_INT_CAT_LOAD_MAIN_i = 0
THEN
  DELTA_LAMB_SP_PUC_i = 0
  LV_MAIN_CAT_PURGE_i = 0
  LV_ACT_INT_PUC_i = 0
ELSE
  IF ( MAF_INT_SAMPLE_STEP_i * IP_DELTA_LAMB_SP_PUC ) >
    ( 2 * MAF_INT_CAT_LOAD_MAIN_i * C_DELTA_LAMB_SP_PUC_DEC )
  THEN
    LV_DOWN_RAMP_ACT_i = 1
    DELTA_LAMB_SP_PUC_i = DELTA_LAMB_SP_PUC_i - C_DELTA_LAMB_SP_PUC_DEC
  ELSE
    IF LV_DOWN_RAMP_ACT_i = 0 THEN
      DELTA_LAMB_SP_PUC_i = DELTA_LAMB_SP_PUC_i + C_DELTA_LAMB_SP_PUC_INC
    ENDIF
  ENDIF
  IF DELTA_LAMB_SP_PUC_i > IP_DELTA_LAMB_SP_PUC
  THEN
    DELTA_LAMB_SP_PUC_i = IP_DELTA_LAMB_SP_PUC
  ENDIF
  MAF_INT_SAMPLE_STEP_i = MAF_CYL * DELTA_LAMB_SP_PUC_i * NC_FAC_MAF_INT_20
  IF MAF_INT_SAMPLE_STEP_i = 0
  THEN
    MAF_INT_SAMPLE_STEP_i = NC_MAF_INT_SAMPLE_STEP_MIN
  ENDIF
  MAF_INT_CAT_LOAD_MAIN_i = MAF_INT_CAT_LOAD_MAIN_i - MAF_INT_SAMPLE_STEP_i
  QUO_MAF_INT_PURGE_i = MAF_INT_CAT_LOAD_MAIN_i / MAF_INT_TOT_PURGE_i
ENDIF
```

Calculation of LV_ACT_INT_PUC_i:

```
LV_ACT_INT_PUC_i =    LV_PCAT_PURGE_PUC_i OR LV_PCAT_PURGE_SA_i OR
                      LV_PCAT_PURGE_HST_i OR LV_PCAT_PURGE_CH_i OR
                      LV_ACT_INT_PUC_AFL_i OR LV_MAIN_CAT_PURGE_i
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_DELTA_LAMB_SP_PUC	6*8	0...80H	0....0.125	9.766E-4	-
LDP_MAF_CYL	6	0...FFFFH	0...2047.969	0.03125	kg/h
LDP_QUO_MAF_INT_PURGE	8	0...FFH	0...1	0.004	-
basic value for lambda controller set point shift					
C_VLS_DOWN_PUC_ACT_MAX	1	0...3FFH	0...4.995	4.89E-3	V
max. VLS_DOWN threshold for deactivation					
IP_MAF_INT_ACT_PUC	1*6	0...FFFFH	0...1820.4167	0.0278	g
LDP_MAF_INT_PUC	6	0...FFFFH	0...1820.4167	0.0278	g
initial value for mass air flow integral for pre cat purge after PUC phase					
IP_FAC_CAT_EFF_PUC	6*6	0...FFH	0...1	3.9E-3	-
LDP_TEMP_PCAT	6	0...FFFFH	-33...990	0.0156	°C
LDPM_CAT_DIAG	6	0...7FH	0...1.984	1.56E-2	-
weight factor for the lambda shift depending on the result of the catalyst diagnosis and of the cat temperature					
C_MAF_INT_PUC_MIN	1	0...FFFFH	0...1820.4167	0.0278	g
minimum MAF_INT threshold for cat purge activation after PUC phase					
C_MAF_INT_ACT_PUC_MAX	1	0...FFFFH	0...1820.4167	0.0278	g
upper limit for MAF_INT_ACT_PUC_i					
C_MAF_INT_PCAT_PURGE_SA	1	0...FFFFH	0...1820.4167	0.0278	g
initial value for mass air flow integral for pre cat purge after SA phase					
C_MAF_INT_PCAT_PURGE_HST	1	0...FFFFH	0...1820.4167	0.0278	g
initial value for mass air flow integral for pre cat purge after HST phase					
C_MAF_INT_PCAT_PURGE_CH	1	0...FFFFH	0...1820.4167	0.0278	g
initial value for mass air flow integral for pre cat purge after CH phase					
IP_FAC_PCAT_MAIN_CAT_PUC	6*6	0...FFH	0...3.984375	0.0156	-
LDP_TEMP_MAIN_CAT	6	0...FFFFH	-33...990	0.0156	°C
LDPM_CAT_DIAG	6	0...7FH	0...1.984	1.56E-2	-
weighting factor containing volume ratio (pre / main cat) and cat temperature during main cat purge					
C_FAC_ADD_PCAT_MAIN_CAT_PUC_SA	1	0...FFH	0...3.984375	0.0156	-
additional weighting factor for IP_FAC_PCAT_MAIN_CAT_PUC for main cat purge after SA					
C_FAC_ADD_PCAT_MAIN_CAT_PUC_HST	1	0...FFH	0...3.984375	0.0156	-
additional weighting factor for IP_FAC_PCAT_MAIN_CAT_PUC for main cat purge after HST					
C_FAC_ADD_PCAT_MAIN_CAT_PUC_CH	1	0...FFH	0...3.984375	0.0156	-
additional weighting factor for IP_FAC_PCAT_MAIN_CAT_PUC for main cat purge after CH					
C_DELTA_LAMB_SP_PUC_INC	1	0...80H	0....0.125	9.766E-4	-
limitation for DELTA_LAMB_SP_PUC increase					
C_DELTA_LAMB_SP_PUC_DEC	1	0...80H	0....0.125	9.766E-4	-
limitation for DELTA_LAMB_SP_PUC decrease					
C_TCO_THD_HST	1	0...FEH	-48...142.5	0.75	°C
temperature threshold for hot start					
C_QUO_MAF_INT_PUC_AFL	1	0...FFH	0...1	0.004	-
calibration MAF integral quotient for the that LV_ACT_INT_PUC_AFL_i is active					

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_MAF_INT_PURGE_MAX	1	0...80 0000H	0...1820.4444	2.1701E-4	g
maximum value of MAF integral (set to 80 0000H)					
NC_MAF_INT_SAMPLE_STEP_MIN	1	0...80 0000H	0...1820.4444	2.1701E-4	g
minimum MAF integral to be added or subtracted each sample step (set to 1h)					

7.58 Application incidences for catalyst enrichment function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_INT_PUC_i	V/O	0...1H	0...1	1	-
catalyst purge function inhibited					
LV_INH_INT_PUC_PUE_i	V/O	0...1H	0...1	1	-
catalyst purge function inhibited (remains equal to 1 until next PUC)					
TEMP_MAIN_CAT_i	V/O	0...FFFFH	-33...990	0.0156	°C
modeled or assumed temperature of main cat					
TEMP_PCAT_i	V/O	0...FFFFH	-33...990	0.0156	°C
Modeled or assumed temperature of pre cat					
LV_ACT_INT_PUC_AFL_i	V/O	0...1H	0...1	1	-
(LV only used with GDi engines)					

Input data:

VLS_UP_i	TCO	MAF_CYL	LV_MFF_AD_ENA
TEMP_CAT	N_32	DELTA_LAMB_SP_PUC	LV_ACT_INT_PUC_i
MAF_INT_PUE	LV_PUC	LV_ASR_ACT	VLS_DOWN_i
MAF_INT_PUC_NOT_ACT			

FUNCTION DESCRIPTION:

General information:

If two separate catalyst systems are concerned then

i = 1, for cylinder bank 1,

i = 2, for cylinder bank 2.

If only one is considered then

i = 1

Description:

The activation conditions for the cat purge function (pre and main cat purge) are summarized by the logical LV_INH_INT_PUC_i. It must be set to 0 (cat purge not inhibited) as a pre condition to start the enrichment function for all lean air fuel mixture conditions - except for requested cat purge.

One of the following conditions must be met in order to set LV_INH_INT_PUC_i = 1:

- engine speed exceed a calibration limit: N_32 > C_N_MAX_CAT_PURGE
- engine cooling temperature is below or equal calibration limit:
  TCO ≤ C_TCO_ACT_PUC_MIN
- ASR intervention : LV_ASR_ACT = 1
- mass air flow exceed a calibration limit:
  MAF_CYL > C_MAF_CYL_ACT_INT_PUC_MAX
- the upstream oxygen sensor voltage crossed the voltage threshold C_THD_VLS_UP_CAT_PURGE from higher to lower
- the cat purge lambda difference will become lower or equal than C_THD_DELTA_LAMB_SP_PUC during the release of a Lambda Adaption phase
- If VLS_UP is below the threshold C_VLS_UP_THD_LEAN_PURGE after combustion MAF integral MAF_INT_PUE exceeds C_MAF_INT_PUE_MIN_LEAN_PURGE the purge mixture is determined to be lean (e.g. intake system leak).
- If VLS_DOWN is over threshold C_VLS_DOWN_MAX_CAT_PURGE (PUC is not long enough to make pre-catalyst lean) when MAF_INT_PUC_NOT_ACT exceeds C_MAF_INT_PUC_NOT_ACT_MAX_PURGE.

If these conditions are not met, then LV_INH_INT_PUC_i = 0.

As LV_INH_INT_PUC_i remains to 1 only 20msec. if one of above condition is met, a second flag LV_INH_INT_PUC_PUE_i is defined to indicate to Downstream O2 sensor diagnosis (PUE-diagnosis) that Catalyst Purge was early stopped : this flag is set to 1 if LV_INH_INT_PUC_i = 1 and reseted only at next PUC phase.

The temperature of the main cat is estimated by assuming a constant temperature difference (C_TEMP_DIF_PCAT_MAIN_CAT) between pre and main cat temperature. TEMP_CAT is the modeled pre cat temperature.

Application conditions:

Initialisation:	LV_INH_INT_PUC_i = LV_INH_INT_PUC_PUE_i = 1 all other outputs = 0
Recurrence:	TA = 20 ms
Activation:	always
Deactivation:	never

Formula section:

Set activation condition for inhibit CAT purge (LV_INH_INT_PUC_i = 1):

IF N_32 > C_N_MAX_CAT_PURGE **OR**
TCO ≤ C_TCO_ACT_PUC_MIN **OR**
MAF_CYL > C_MAF_CYL_ACT_INT_PUC_MAX **OR**
LV_ASR_ACT = 1 **OR**
(VLS_UP_i $_{(n-1)}$ > C_THD_VLS_UP_CAT_PURGE **AND**
VLS_UP_i $_{(n)}$ ≤ C_THD_VLS_UP_CAT_PURGE **AND**
LV_ACT_INT_PUC_i = 1) **OR**
(DELTA_LAMB_SP_PUC$_{(n-1)}$ > C_THD_DELTA_LAMB_SP_PUC **AND**
DELTA_LAMB_SP_PUC$_{(n)}$ ≤ C_THD_DELTA_LAMB_SP_PUC **AND**
LV_ACT_INT_PUC_i = 1 **AND**
LV_MFF_AD_ENA = 1) **OR**
(MAF_INT_PUE > C_MAF_INT_PUE_MIN_LEAN_PURGE **AND**
VLS_UP_i ≤ C_VLS_UP_THD_LEAN_PURGE **AND**
LV_ACT_INT_PUC = 1) **OR**
(VLS_DOWN_i > C_VLS_DOWN_MAX_CAT_PURGE **AND**
MAF_INT_PUC_NOT_ACT > C_MAF_INT_PUC_NOT_ACT_MAX_PURGE)

THEN
LV_INH_INT_PUC_i = 1
ELSE
LV_INH_INT_PUC_i = 0
ENDIF

Set activation condition for inhibit CAT purge for PUE- diagnosis (LV_INH_INT_PUC_PUE_i = 1):

If LV_INH_INT_PUC_i = 1
Then LV_INH_INT_PUC_PUE_i = 1
Else
if LV_PUC = 1
Then LV_INH_INT_PUC_PUE_i = 0
Endif
Endif

Evaluation of main / pre cat temperature:

TEMP_MAIN_CAT_i = TEMP_CAT - C_TEMP_DIF_PCAT_MAIN_CAT

(physical, HEX converted)

TEMP_PCAT_i = TEMP_CAT

(physical, HEX converted)

(Only necessary for GDi engines)

LV_ACT_INT_PUC_AFL_i = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_ACT_PUC_MIN	1	0...FEH	-48...142.5	0.75	°C
minimum TCO threshold for cat purge activation					
C_N_MAX_CAT_PURGE	1	0...FFH	0....8160	32	rpm
maximum engine speed for cat purge					
C_MAF_CYL_ACT_INT_PUC_MAX	1	0...FFFFH	0...2047.96875	0.03125	kg/h
Maximum MAF_CYL threshold for active cat purge function					
C_TEMP_DIF_PCAT_MAIN_CAT	1	0...FFFFH	-33...990	0.0156	°C
temperature difference between pre and main cat					
C_THD_VLS_UP_CAT_PURGE	1	0...3FFH	0...4.995	0.0049	V
VLS_UP – threshold to inhibit cat purge function					
C_THD_DELTA_LAMB_SP_PUC	1	0...80H	0...0.125	9.766e-4	-
DELTA_LAMB_SP_PUC – threshold to inhibit cat purge function					
C_VLS_UP_THD_LEAN_PURGE	1	0...3FFH	0...4.995	0.0049	V
VLS_UP value below which the mixture during purge is determined to be lean					
C_MAF_INT_PUE_MIN_LEAN_PURGE	1	0...FFFFH	0...1820.42	2.78E-02	[g]
MAF_INT_PUE minimum for VLS_UP monitoring to determine lean mixture during catalyst purge					
C_VLS_DOWN_MAX_CAT_PURGE	1	0...3FFH	0...4.995	0.0049	V
Down stream voltage threshold for cat purge activation					
C_MAF_INT_PUC_NOT_ACT_MAX_PURGE	1	0…FFFFH	0…2.91267E+3	0.0444444 4	g
Maximum air mass flow integral to allow inhibition by VLS_DOWN signal					

7.59 LASP Application Incidences

7.59.1 External Adjustment coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LAMB_SP_EXT_ADJ[NC_CBK_EX_NR]	O	0 ... 1H	0 ... 1	1	-
Falg that indicates that external adjustment of the lambda setpoint is active					
LAMB_SP_EXT_ADJ[NC_CBK_EX_NR]	O	0...7FFFH	0....31.999	9.7656E-4	-
external adjustment of the lambda setpoint					

Input data:

NC_CBK_EX_NR			

FUNCTION DESCRIPTION:

Application conditions:

Initialisation: -

Recurrence: 10 ms

Activation: LV_ST_END = 1

Deactivation: -

Formula Section :

FOR [i] = 1 **TO** NC_CBK_EX_NR **DO**:

LV_LAMB_SP_EXT_ADJ[i] = 0
LAMB_SP_EXT_ADJ[i] = 0

ENDFOR

7.59.2 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_SP_MIN_DC	O/V	0...7FFFH	0...31,999	9,7656E-4	-
Minimum lambda setpoint during current driving cycle					
LAMB_LS_UP_MIN_DC	O/V	0...7FFFH	0...31.999	9.7656E-4	-
Minimum lambda signal during current driving cycle					

Input data:

LAMB_SP[NC_CBK_EX_NR]	LV_ST_END	NC_STATE_LSL_UP_IF	LAMB_LS_UP[NC_CBK_EX_NR]

Application conditions:

Initialisation: at LV_IGK 0->1 and reset:

LAMB_SP_MIN_DC = LAMB_LS_UP_MIN_DC = 7FFFH

Recurrence: 10 ms

Activation: LV_ST_END = 1

Deactivation: -

Formula section:

If LAMB_SP < LAMB_SP_MIN_DC

Then LAMB_SP_MIN_DC = LAMB_SP

Endif

```
If      NC_STATE_LSL_UP_IF = 1                         //linear upstream O2-sensor
   Then     If     LAMB_LS_UP < LAMB_LS_UP_MIN_DC
            Then   LAMB_LS_UP_MIN_DC = LAMB_LS_UP
                          Endif
                   Endif
```

7.60 FMSP Application Incidences (MPI version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_ADD_EXT	O	0 ... FFFFH	0 ... 1389	0.021194	mg/stk
Additive fuel amount from external intervention					
FAC_MFF_EXT	V/O	0 ... FFH	0 ... 1.99215	0.0078413	-
Factor for fuel amount from external intervention					
FAC_MFF_BAS_EXT	O	0 ... FFH	0 ... 1.99215	0.0078413	-
External correction factor for basic setpoint					
FAC_MFF_ST_EXT	O	0 ... FFH	0 ... 1.99215	0.0078413	-
External correction factor for start					
MFF_ADD_LAM_AD_OUT_EXT	V/O	8000...7FFFH	-694.510597 ...694.489403	0.021194	mg/stk
External correction of the additive lambda adaptation					
MFF_SP_MAX_DC	O/V	0...FFFFH	0...1389	0,02119	mg/stk
Maximum engine fuel quantitiy during current driving cycle					
STATE_PORT_MFF_COR	V/O	0H 1H 2H	PORT_PASSIVE PORT_OPEN PORT_CLOSE	1	-
State of port flag fuel correction					
CYC_PORT_OPEN	V/O	0 ... FFH	0 ... 255	1	-
Cycle counter of port flag opening correction					
CYC_PORT_CLOSE	V/O	0 ... FFH	0 ... 255	1	-
Cycle counter of port flag closing correction					
LV_PORT_OPEN_DET	V/O	0 ... 1H	0 ... 1	1	-
Flag indicates that port flag open is detected					
LV_PORT_CLOSE_DET	V/O	0 ... 1H	0 ... 1	1	-
Flag indicates that port flag close is detected					
FAC_MFF_CST_EXT	O	0...FFH	0...1.99218	0.0078125	[-]
Factor for start fuel amount from external intervention					

Input data:

LV_ST_END	MFF_CO_IS	N_32	C_N_TOL_ADD_RNG_LAM_AD
MFF_SP_1_HOM[NC_CYL_NR]	FAC_PORT_DEAC	TCO	MAF_STK_FG_PRED
MFF_SP_HOM_CLC[NC_CBK_EX_NR]	PORT_AV_GRD	CONF_PORT	LV_ERR_PORT

FUNCTION DESCRIPTION:

General information:

The external intervention factor FAC_MFF_EXT is used as port flap correction factor.

During port flap closing (=activation) a wall film enleanment compensation can be considered; during port flap opening (= deactivation to passive position) a wall film enrichment compensation can be considered to compensate lambda deviations.

The wall film compensation is only applied in case of port flap variant set (CONF_PORT = 1) or no error on port flap control (LV_ERR_PORT = 0).

Application conditions:

Initialisation: at reset: FAC_MFF_EXT = 1
FAC_MFF_CST_EXT = 1
FAC_MFF_BAS_EXT = 1
FAC_MFF_ST_EXT = 1
MFF_SP_MAX_DC = 0
MFF_ADD_EXT = 0
MFF_ADD_LAM_AD_OUT_EXT = 0
LV_PORT_OPEN_DET = 0
LV_PORT_CLOSE_DET = 0
STATE_PORT_MFF_COR = PORT_PASSIVE
CYC_PORT_OPEN = 0
CYC_PORT_CLOSE = 0

Recurrence: LV_ST_END = 0: 10 ms
LV_ST_END = 1: segment sychronous

Activation: every engine state

Deactivation: -

Formula section:

MFF_ADD_EXT = 0

FAC_MFF_BAS_EXT = 1

FAC_MFF_CST_EXT = 1

FAC_MFF_ST_EXT = 1

MFF_ADD_LAM_AD_OUT_EXT = C_N_TOL_ADD_RNG_LAM_AD * MFF_CO_IS/N_32

If MFF_SP_1_HOM[0] > MFF_SP_MAX_DC

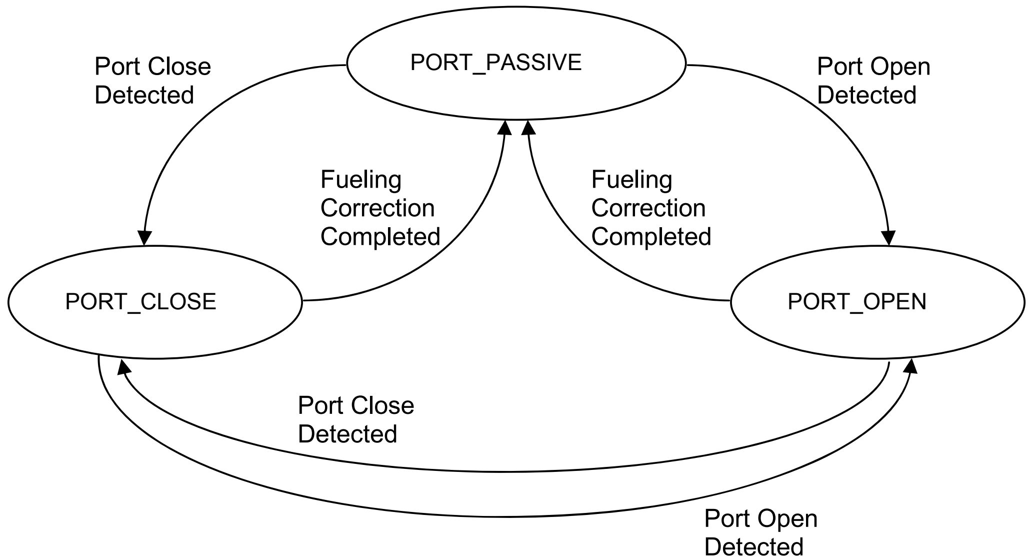
Then MFF_SP_MAX_DC = MFF_SP_1_HOM[0]

Endif

If CONF_PORT = 1 and LV_ERR_PORT = 0, then the following Port flap correction is calculated.

Port flap correction:

Diagram of port flap fuel correction state machine is shown below.



7.60.1 State PORT_PASSIVE

Formula Section :

FAC_MFF_EXT = 1

CYC_PORT_OPEN = 0

CYC_PORT_CLOSE = 0

Port flap open detection and Transition condition to PORT_OPEN

if PORT_AV_GRD <= C_PORT_GRD_OPEN_DET //port flap close-to-open transition

When the port flap opens, the gradient becomes negative, with C_PORT_GRD_OPEN_DET can be determined if the function gets active or not

then LV_PORT_OPEN_DET = 1

endif

Port flap close detection and Transition condition to PORT_CLOSE

if PORT_AV_GRD >= C_PORT_GRD_CLOSE_DET //port flap open-to-close transition

When the port flap closes, the gradient becomes positive, with C_PORT_GRD_CLOSE_DET can be determined if the function gets active or not

then LV_PORT_CLOSE_DET = 1

endif

7.60.1.1 Transition condition to PORT_OPEN

Formula Section :

if LV_PORT_OPEN_DET = 1
and FAC_PORT_DEAC >= C_FAC_PORT_OPEN_THD
then STATE_PORT_MFF_COR = PORT_OPEN
endif

7.60.1.2 Transition condition to PORT_CLOSE

Formula Section :

if LV_PORT_CLOSE_DET = 1
and FAC_PORT_DEAC <= C_FAC_PORT_CLOSE_THD
then STATE_PORT_MFF_COR = PORT_CLOSE
endif

7.60.2 State PORT_OPEN

Formula Section :

As long as the state machine stays in PORT_OPEN, the cycle counter is incremented continuously every recurrence.

Increment CYC_PORT_OPEN //limits to maximum

Fuel correction determination during PORT_OPEN transition

if LC_PORT_MFF_ADD_SEL = 1
then // Chosen to use additive to correct port open fueling
IF MFF_SP_HOM_CLC = 0
Then FAC_MFF_EXT = 1
Else
FAC_MFF_EXT = (1 + IP_MFF_ADD_PORT_OPEN / MFF_SP_HOM_CLC)
* IP_COR_TCO_MFF_PORT_OPEN
else // Chosen to use factor to correct port open fueling
FAC_MFF_EXT = IP_FAC_MFF_PORT_OPEN * IP_COR_TCO_MFF_PORT_OPEN
endif

7.60.2.1 Transition condition to PORT_PASSIVE

After completion of the port transition fuelling correction, state transits to PORT_PASSIVE

if CYC_PORT_OPEN >= C_CYC_PORT_OPEN_MAX //correction complete

then STATE_PORT_MFF_COR = PORT_PASSIVE

LV_PORT_OPEN_DET = 0 //reset port open/close detection

LV_PORT_CLOSE_DET = 0

endif

7.60.2.2 Transition condition to PORT_CLOSE

During the brief port-open fueling period, when port-close is detected, state transits to PORT_CLOSE through PORT_PASSIVE.

if PORT_AV_GRD >= C_PORT_GRD_CLOSE_DET //port flap open-to-close transition

then STATE_PORT_MFF_COR = PORT_PASSIVE

LV_PORT_OPEN_DET = 0

LV_PORT_CLOSE_DET = 1

endif

7.60.3 State PORT_CLOSE

Formula Section :

As long as the state machine stays in PORT_CLOSE, the cycle counter is incremented continuously every recurrence.

Increment CYC_PORT_CLOSE // limits to maximum

Fuel correction determination during PORT_CLOSE transition

if LC_PORT_MFF_ADD_SEL = 1

then // Chosen to use additive to correct port open fueling

IF MFF_SP_HOM_CLC = 0

Then FAC_MFF_EXT = 1

Else

FAC_MFF_EXT =(1+ IP_MFF_ADD_PORT_CLOSE / MFF_SP_HOM_CLC)

* IP_COR_TCO_MFF_PORT_CLOSE

else // Chosen to use factor to correct port open fueling

FAC_MFF_EXT = IP_FAC_MFF_PORT_CLOSE

* IP_COR_TCO_MFF_PORT_CLOSE

endif

7.60.3.1 Transition condition to PORT_PASSIVE

After completion of the port transition fueling correction, state transits to PORT_PASSIVE

if CYC_PORT_CLOSE >= C_CYC_PORT_CLOSE_MAX //correction complete

then STATE_PORT_MFF_COR = PORT_PASSIVE

LV_PORT_CLOSE_DET = 0 //reset port open/close detection

LV_PORT_OPEN_DET = 0

endif

7.60.3.1.1 Transition condition to PORT_OPEN

During the port close fueling period, when port-open is detected, state transits to PORT_OPEN through PORT_PASSIVE.

if PORT_AV_GRD <= C_PORT_GRD_OPEN_DET //port flap open-to-close transition

then STATE_PORT_MFF_COR = PORT_PASSIVE

LV_PORT_OPEN_DET = 1

LV_PORT_CLOSE_DET = 0

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CYC_PORT_OPEN_MAX	1	0...FFH	0...255	1	-
Number of segments to complete port-open fuel correction					
C_CYC_PORT_CLOSE_MAX	1	0...FFH	0...255	1	-
Number of segments to complete port-close fuel correction					
IP_MFF_ADD_PORT_OPEN	16x16	0 ... FFFFH	-694.48... 694.46	0.021194	mg/stk
LDPM_CYC_PORT_OPEN_IP_MFF	16	0H...FFH	0...255	1	-
LDPM_MAF_STK_FG_PRED_IP_MFF	16	0000...FFFFH	0...2778	0.0424	mg/stk
Fuel additive for Port flag open correction					
IP_FAC_MFF_PORT_OPEN	16x16	0 ... FFH	0 ... 1.99215	0.0078413	-
LDPM_CYC_PORT_OPEN_IP_MFF	16	0H...FFH	0...255	1	-
LDPM_MAF_STK_FG_PRED_IP_MFF	16	0000...FFFFH	0...2778	0.0424	mg/stk
Fuel factor for Port flag open correction					
IP_MFF_ADD_PORT_CLOSE	16x16	0 ... FFFFH	-694.48... 694.46	0.021194	mg/stk
LDPM_CYC_PORT_CLOSE_IP_MFF	16	0H...FFH	0...255	1	-
LDPM_MAF_STK_FG_PRED_IP_MFF	16	0000...FFFFH	0...2778	0.0424	mg/stk
Fuel additive for Port flag close correction					
IP_FAC_MFF_PORT_CLOSE	16x16	0 ... FFH	0 ... 1.99215	0.0078413	-
LDPM_CYC_PORT_CLOSE_IP_MFF	16	0H...FFH	0...255	1	-
LDPM_MAF_STK_FG_PRED_IP_MFF	16	0000...FFFFH	0...2778	0.0424	mg/stk
Fuel factor for Port flag open correction					
LC_PORT_MFF_ADD_SEL	1	0...1H	0...1	1	-
Calibration switch to chose between additive or factor correction for port flap (1-> additive, 0 -> factor)					
IP_COR_TCO_MFF_PORT_OPEN	8	0 ... FFH	0 ... 1.99215	0.0078413	-
LDPM_TCO_IP_COR_TCO	8	0H...FEH	-48...142.5	0.75	°C
TCO dependent correction factor for port flag open					
IP_COR_TCO_MFF_PORT_CLOSE	8	0 ... FFH	0 ... 1.99215	0.0078413	-
LDPM_TCO_IP_COR_TCO	8	0H...FEH	-48...142.5	0.75	°C
TCO dependent correction factor for port flag close					
C_FAC_PORT_OPEN_THD	1	0...FFFFH	0...0.9999847	0.0000153	-
Port flag open limit to start fuel correction					
C_FAC_PORT_CLOSE_THD	1	0...FFFFH	0...0.9999847	0.0000153	-
Port flag close limit to start fuel correction					
C_PORT_GRD_OPEN_DET	1	F000H...1000H	-1000...1000	0.2441	°PORT/s
PORT_AV_GRD treshold to determine port flap close to open transition					
C_PORT_GRD_CLOSE_DET	1	F000H...1000H	-1000...1000	0.2441	°PORT/s
PORT_AV_GRD treshold to determine port flap open to close transition					

7.61 INJR Application Incidences

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_N	V/O	0 ... 1FE0H	0 ... 48.96	0.006	°CRK/ms
Engine speed factor [°CRK/ms]					
N_FAST_INJ	O	0...1FE0H	0...8160	1	rpm
Engine Speed- Resolution 1 rpm					
CTR_CYL_NR_ST_CLC_INJR	O	0 ... 7H	0 ... 7	1	-
Number of first cylinder in calculation order					
CTR_CYL_NR_STOP_CLC_INJR	O	0 ... 7H	0 ... 7	1	-
Number of last cylinder in calculation order					
CTR_CBK_IN_NR_ST_CLC_INJR	O	0 ... 1H	1 ... 2	1	-
Start number of intake bank for calculations					
CTR_CBK_IN_NR_STOP_CLC_INJR	O	0 ... 1H	1 ... 2	1	-
Stop number of intake bank for calculations					
CTR_CBK_EX_NR_ST_CLC_INJR	O	0 ... 1H	1 ... 2	1	-
Start number of exhaust bank for calculations					
CTR_CBK_EX_NR_STOP_CLC_INJR	O	0 ... 1H	1 ... 2	1	-
Stop number of exhaust bank for calculations					
LV_FAC_TI_EXT_ADJ	O	0 ... 1 H	0 ... 1	1	-
Flag for external injection time adjustment factor active					
FAC_TI_EXT_ADJ	O	0 ... FFH	0 ... 1.992	0 0078	-
external injection time adjustment factor					
LV_TI_EXT_ADJ[NC_CYL_NR]	O	0 ... 1 H	0 ... 1	1	-
Flag for external injection time adjustment					
TI_EXT_ADJ[NC_CYL_NR]	O	0 ... FFFFH	0 ... 262.14	0.004	ms
External injection time adjustment					

Input data:

N	LV_ST_END	CTR_CYL_NR_ST_CLC	CTR_CYL_NR_STOP_CLC
		CTR_CBK_IN_NR_ST_CLC	CTR_CBK_IN_NR_STOP_CLC
		CTR_CBK_EX_NR_ST_CLC	CTR_CBK_EX_NR_STOP_CLC

FUNCTION DESCRIPTION:

General information:

This specification is a application incidences template for project specific adaptations.

7.61.1 General Tasks

Application conditions :

Activation : every engine state

Deactivation : -

Initialization: at reset : FAC_TI_EXT_ADJ = 1

LV_TI_EXT_ADJ[NC_CYL_NR] = 0

TI_EXT_ADJ[NC_CYL_NR] = 0

Recurrence : LV_ST_END = 0: 10 ms

LV_ST_END = 1: segment sychronous

Formula Section :

N_FAST_INJ = N

FAC_N = N_FAST_INJ * 360 / (1000 * 60)

LV_FAC_TI_EXT_ADJ = 0

FAC_TI_EXT_ADJ = 1

(1) FOR x = 0 **TO** (NC_CYL_NR -1)

LV_TI_EXT_ADJ[x] = 0

TI_EXT_ADJ[x]= 0

(1) END FOR

7.61.2 Runtime reduction

Application conditions :

Activation : every engine state

Deactivation : -

Initialization: -

Recurrence : LV_ST_END = 0: 10 ms

LV_ST_END = 1: segment sychronous

This chapter describes runtime reduction steps. First step is to reduce cylinder dependent calculations from all cylinders to one cylinder, dependent on the engine speed. An additional step is to reduce bank dependent calculations to that intake or exhaust bank, which is allocated to the related cylinder by NC_IN_REF and NC_LAMB_REF. The control variables for cylinder and bank dependent calculations will be imported via CTR_CYL_NR_xxx and CTR_CBK_xxx data.

Formula Section :

7.61.2.1 Determine cylinder and bank numbers for follow up calculations

CTR_CYL_NR_ST_CLC_INJR = CTR_CYL_NR_ST_CLC

CTR_CYL_NR_STOP_CLC_INJR = CTR_CYL_NR_STOP_CLC

CTR_CBK_IN_NR_ST_CLC_INJR = CTR_CBK_IN_NR_ST_CLC

CTR_CBK_IN_NR_STOP_CLC_INJR = CTR_CBK_IN_NR_STOP_CLC
CTR_CBK_EX_NR_ST_CLC_INJR = CTR_CBK_EX_NR_ST_CLC
CTR_CBK_EX_NR_STOP_CLC_INJR = CTR_CBK_EX_NR_STOP_CLC

7.62 EXTD output selection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEG_DYN_UP_CAT[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Dynamic exhaust gas temperature upstream catalysator					
TEG_DYN_DOWN_CAT[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
dynamic exhaust gas temperatures downstream catalyst					
TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Modelled catalyst temperature under dynamic conditions					
TEMP_CAT_STAT_MDL[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Modelled catalyst temperature under steady conditions					
TEG_DYN_LS_UP[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Exhaust gas temperatures at the lambda sensor upstream catalyst					
TEG_DYN_LS_DOWN[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Exhaust gas temperatures at the lambda sensor downstream catalyst					
TEG_DYN_UP_TUR[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
dynamic exhaust gas temperature upstream turbine					
TEG_STAT_UP_TUR_MDL[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
steady state exhaust gas temperature upstream turbine					
TEG_STAT_UP_CAT_MDL[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
steady state exhaust gas temperature upstream catalyst					
TEG_STAT_DOWN_CAT_MDL[NC_CBK_EX_NR]	O/V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
steady state exhaust gas temperature downstream catalyst					

Input data:

TEG[NC_NR_TEG_MDL]	TEG_STAT_MDL[NC_NR_TEG_MDL]	TEMP_CAT[NC_NR_CAT_MDL]	TEMP_CAT_STAT[NC_NR_CAT_MDL]
LV_ES	NC_CBK_EX_NR	NC_NR_TEG_MDL	NC_NR_CAT_MDL

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_SEL_TEG_DOWN_CAT[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for TEG_DYN_DOWN_CAT and TEG_STAT_DOWN_CAT_MDL					
C_CTR_SEL_TEG_LS_DOWN[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for the exhasut gas temperature at the downstream lambda sensor					
C_CTR_SEL_TEG_LS_UP[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for the exhasut gas temperature at the upstream lambda sensor					
C_CTR_SEL_TEG_UP_CAT[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for TEG_DYN_UP_CAT and TEG_STAT_UP_CAT_MDL					

general specification

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_SEL_TEG_UP_TUR[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for TEG_DYN_UP_TUR and TEG_STAT_UP_TUR_MDL					
C_CTR_SEL_TEMP_CAT[NC_CBK_EX_NR]	1	0...FFH	1...256	1	-
selector for the catalyst temperature					

7.62.1 EXTD_SELECT0

Function Description

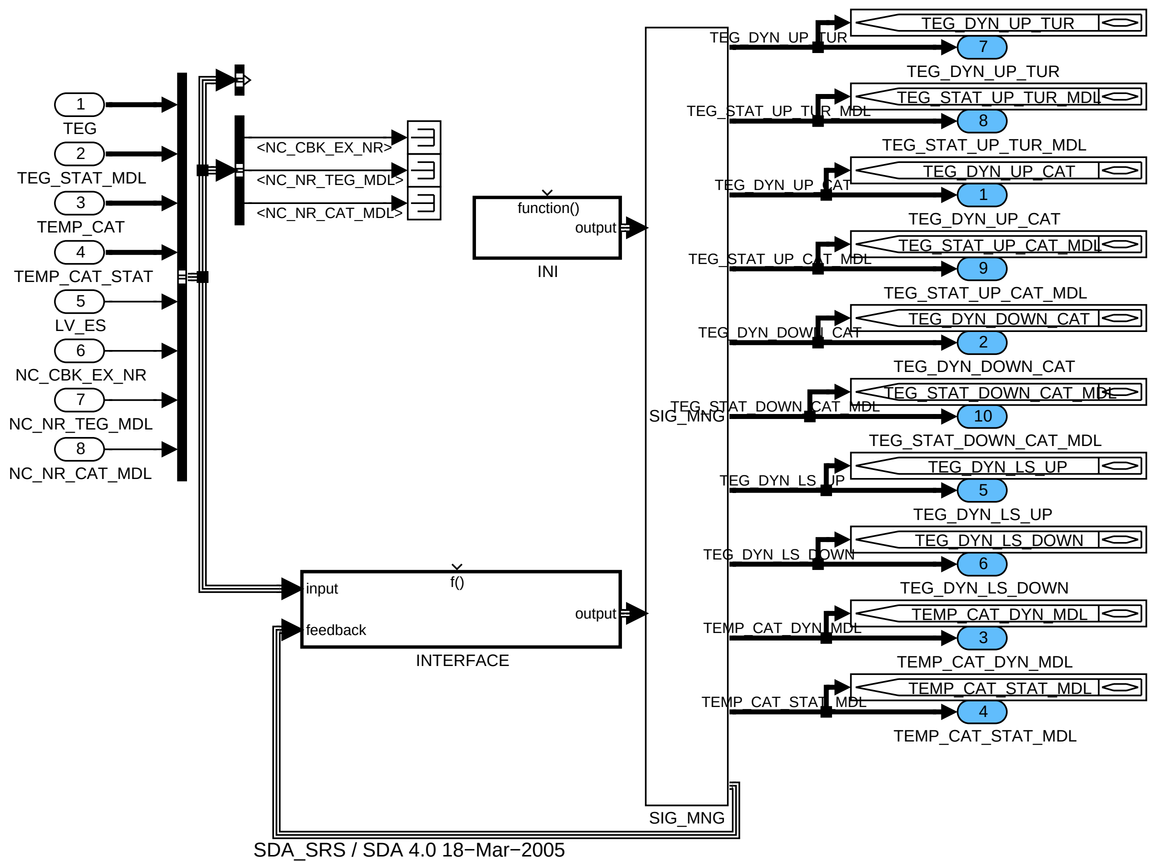


Figure 100 EXTD_SELECT0

7.62.1.1 SUBFUNCTION: INTERFACE

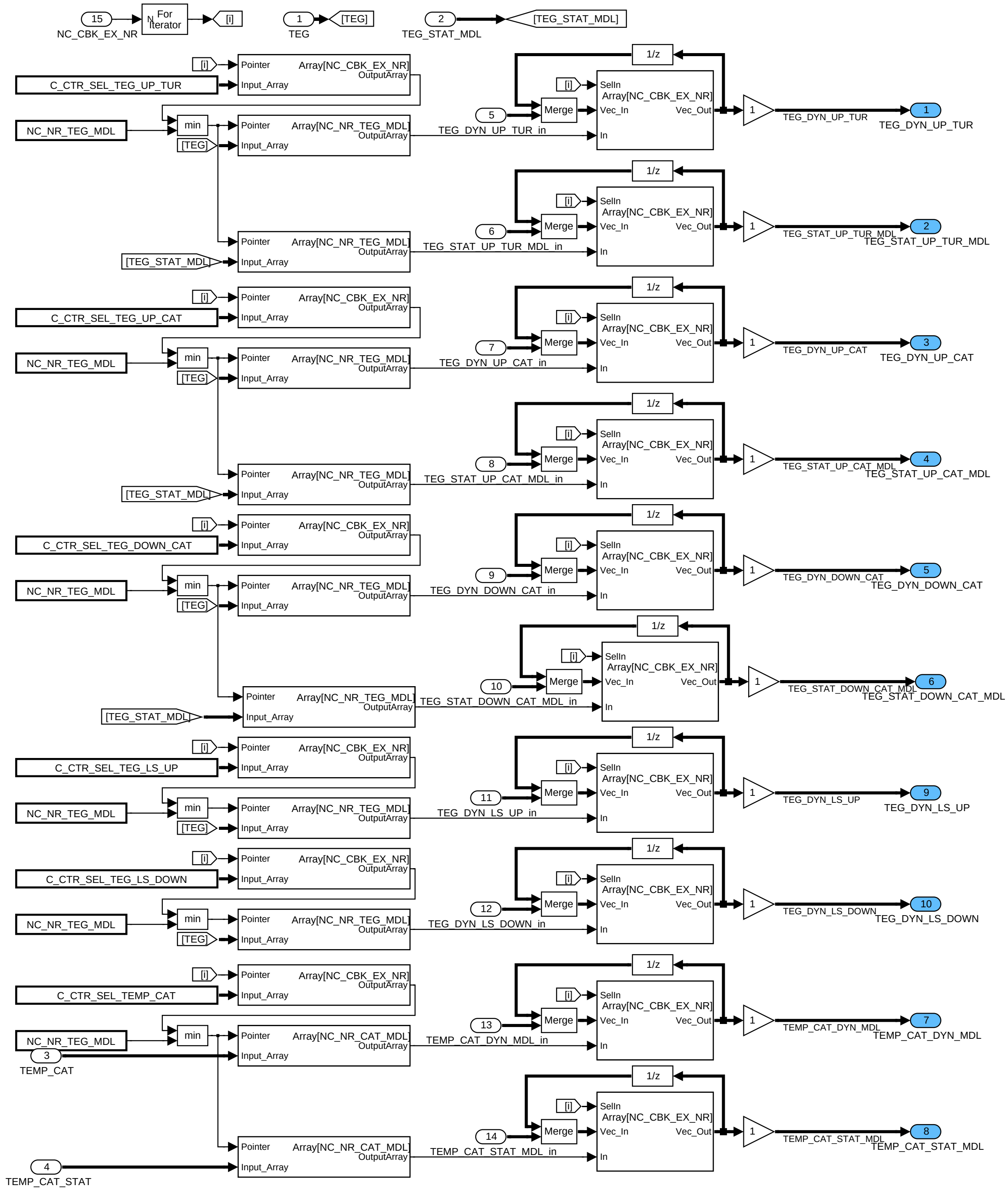


Figure 101 EXTD_SELECT0/ INTERFACE/ SEL_CBK

8 Engine Speed Control

Table of Contents

Index

V

8.1 Idle Speed Control General

The following figure 1 gives an overview of the single modules of the idle speed controller.

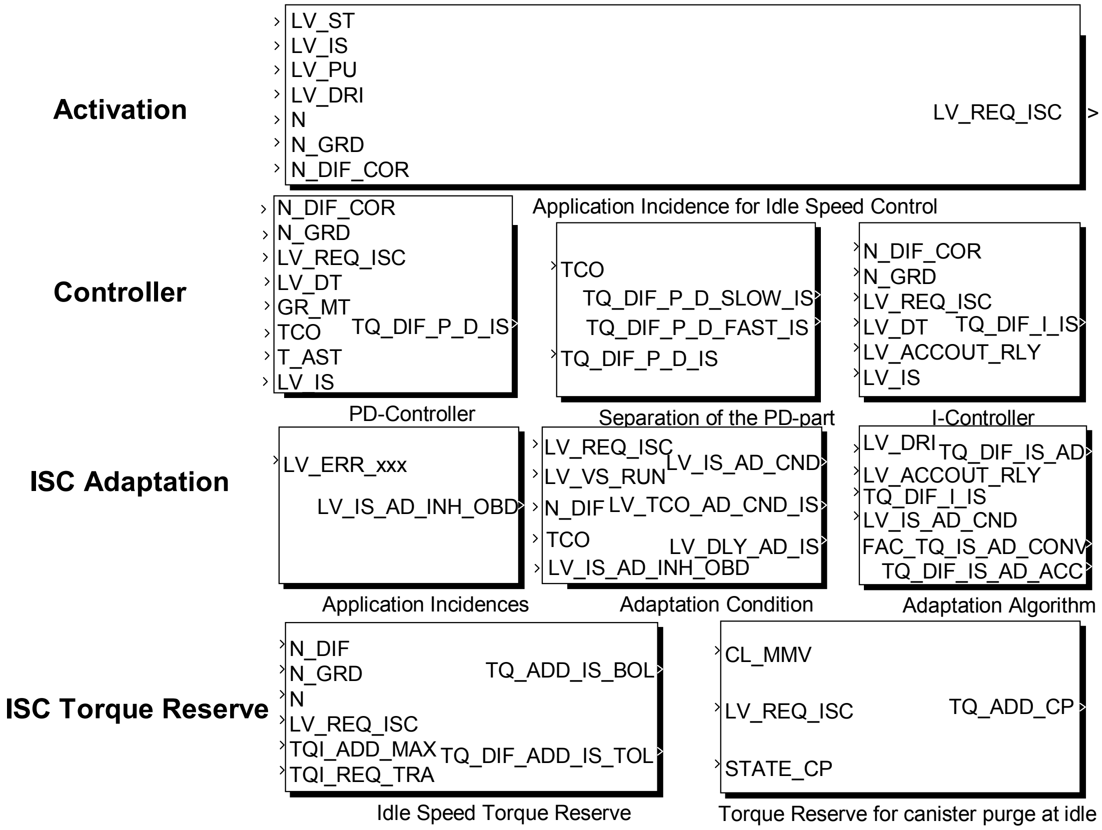


Figure 1: *ISC overview*

8.2 Idle Speed Controller

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_DIF_P_D_SLOW_IS	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
PD-part for slow-path of torque coordination					
TQ_DIF_P_D_FAST_IS	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
PD-part for fast-path of torque coordination					
TQ_DIF_I_IS	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
I-part					
LV_PAS_RAMP_ACT_P_D_IS	V/O	0, 1H	0, 1	1	-
Logical value for PD-part passive ramp active					
LV_PAS_RAMP_ACT_I_IS	V/O	0, 1H	0, 1	1	-
Logical value for I-part passive ramp active					
TQ_DIF_P_D_IS	V	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
PD-part before separation into fast/slow path					
TQ_ADD_I_IS	V	8000 ... 7FFFH	-4 ... 3.999	1.22E-4	Nm
Incrememt of limited integrator					
TQ_ADD_P_D_RAMP	V	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Increment of the PD-part ramps; this value is added per update rate					
TQ_ADD_I_RAMP	V	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Increment of the I-part ramps; this value is added per update rate					
FAC_N_GRD_IS	V	0 ... FF H	0 ... 1.992	0.008	-
Factor in dependance of TCO and TOIL for weighting the speed gradient influence on the PD-part (for LV_DT=0)					
N_GRD_FIL_IS	V	80 ... 7F H	-4096 ... 4064	32	rpm/s
Filtered Engine speed gradient					
N_SP_IS_COR_GRD_FIL_IS	V	80 ... 7F H	-4096 ... 4064	32	rpm/s
Filtered corrected Idle speed set-point gradient					
N_GRD_P_D_IS	V	80 ... 7F H	-4096 ... 4064	32	rpm/s
Filtered gradient; gradient input of the PD-part maps					
N_GRD_I_IS	V	80 ... 7F H	-4096 ... 4064	32	rpm/s
Filtered gradient; gradient input of the I-part maps					
LV_TQ_P_D_ACT	V	0, 1 H	0, 1 H	1	-
Logical variable for PD-part active (STATE_P_D_ISC = P_D_NORMAL)					
STATE_P_D_ISC	V	0 H 1 H 2 H	P_D_PASSIVE P_D_NORMAL RAMP_TO_ ZERO	-	-
State of the PD-part: P_D_PASSIVE, P_D_NORMAL, RAMP_TO_ZERO					
STATE_I_ISC	V	0 H 1 H 2 H 3 H	I_PASSIVE I_NORMAL RAMP_TO_PAS RAMP_TO_LIM	-	-
State of the integral part: I_PASSIVE, I_NORMAL, RAMP_TO_PAS, RAMP_TO_LIM					
T_RAMP_LIM_P_D_IS	-	0 .. FFFF H	0 ... 655.35	0.01	s
Time constant for ramp limit of P-, D-controller output in case of idle speed controller passive					
T_RAMP_LIM_I_IS	-	0 .. FFFF H	0 ... 1310.7	0.02	s
Time constant for ramp limit of I-controller output in case of idle speed controller passive					

Input data:

N_DIF_COR	N_GRD	LV_REQ_ISC	LV_DT
GR_MT	T_AST	TCO	LV_ISC_INH_EXT_ADJ
LV_ACCOUT_RLY	LV_ES	N_SP_IS_COR_GRD	LV_N_INC_TCU_ACT
GR_AT			

8.2.1 General

The controller consits of three parts:

- PD-part
- PD-part separation into slow and fast path
- I-part

The controller can be enabled/disabled with several calibration bits:

- If LC_ISC_INH == 1, then the idle speed controller is completly disabled:

TQ_DIF_P_D_IS = 0
TQ_DIF_P_D_SLOW_IS = 0
TQ_DIF_P_D_FAST_IS = 0
TQ_DIF_I_IS = 0

- If LC_TQ_DIF_P_D_INH_AS == 1, then only the PD-part is disabled:

TQ_DIF_P_D_IS = 0
TQ_DIF_P_D_SLOW_IS = 0
TQ_DIF_P_D_FAST_IS = 0

- If LC_TQ_DIF_I_INH_AS == 1, then only the I-part is disabled:

TQ_DIF_I_IS = 0

- If LC_TQ_ADD_I_INH_AS == 1, then only the I-part's increment (input of the integrator) is set to zero:

TQ_ADD_I_IS = 0 $\Rightarrow$ TQ_DIF_I_IS = const.

- Additionally, the controller can be disabled completely by an external diagnosis tool. Therefore, the variable LV_ISC_INH_EXT_ADJ is used. If LV_ISC_INH_EXT_ADJ == 1, then

TQ_DIF_P_D_IS = 0
TQ_DIF_P_D_SLOW_IS = 0
TQ_DIF_P_D_FAST_IS = 0
TQ_DIF_I_IS = 0.

8.2.2 Filtering of N_GRD

Application conditons:

Initialization: N_GRD_FIL_IS = N_SP_IS_COR_GRD_FIL_IS = 0 rpm/s at reset
Recurrence 10 ms
Activation: at every engine state
Scheduling: This function has to be calculated immediate **before the controller** and **after the application incidences**.

In order to be able to smooth the gradient for the idle speed control, a 1st order low-pass filter can be used to define N_GRD_FIL_IS.

Formula section:

$$N\_GRD\_FIL\_IS_k = N\_GRD\_FIL\_IS_{k-1} + (N\_GRD_k - N\_GRD\_FIL\_IS_{k-1}) \cdot C\_CRLC\_N\_GRD.$$

$$N\_SP\_IS\_COR\_GRD\_FIL\_IS_k = N\_SP\_IS\_COR\_GRD\_FIL\_IS_{k-1} + (N\_SP\_IS\_COR\_GRD_k - N\_SP\_IS\_COR\_GRD\_FIL\_IS_{k-1}) * C\_CRLC\_N\_GRD$$

8.2.3 PD- Controller

General information:

The PD-part is devided into three states:

STATE_P_D_ISC = P_D_PASSIVE:
Initialization (at reset or from LV_IGK = 0 → 1; from all other states in case of LV_ES = 1)

STATE_P_D_ISC = P_D_NORMAL:
In this state the evaluation of the controller maps takes place.

STATE_P_D_ISC = RAMP_TO_ZERO:
Ramp to passive value, which is always zero for the PD-part; in this state the PD-part is ramped down to zero in case of deactivation of the PD-part

Application conditons:

Initialization: TQ_DIF_P_D_IS = 0 Nm at reset
Recurrence 10 ms
Activation: at every engine state
Scheduling: This function has to be calculated immediate **before the module "Torque Coordination"** and **after the application incidences and the filtered gradient determination**.

Formula section:

For special function test, it´s possible to set TQ_DIF_P_D_IS = 0 Nm by application system or diagnostic tool (LV_ISC_INH_EXT_ADJ). The PD-controller can be activated/deactivated with the following calibration bits:

```
IF      { LC_TQ_DIF_P_D_INH_AS == 1 }          OR
        { LC_ISC_INH == 1 }                    OR
        { LV_ISC_INH_EXT_ADJ == 1 }
THEN
        TQ_DIF_P_D_IS = 0 Nm;
        TQ_DIF_P_D_SLOW_IS = 0 Nm;
        TQ_DIF_P_D_FAST_IS = 0 Nm;
ELSE
        Normal calculation
```

```
IF      LV_DT == 1
THEN    T_RAMP_LIM_P_D_IS = C_T_RAMP_LIM_P_D_DT_IS
ELSE    T_RAMP_LIM_P_D_IS = C_T_RAMP_LIM_P_D_IS
```

8.2.3.1 State Machine

FUNCTION DESCRIPTION:

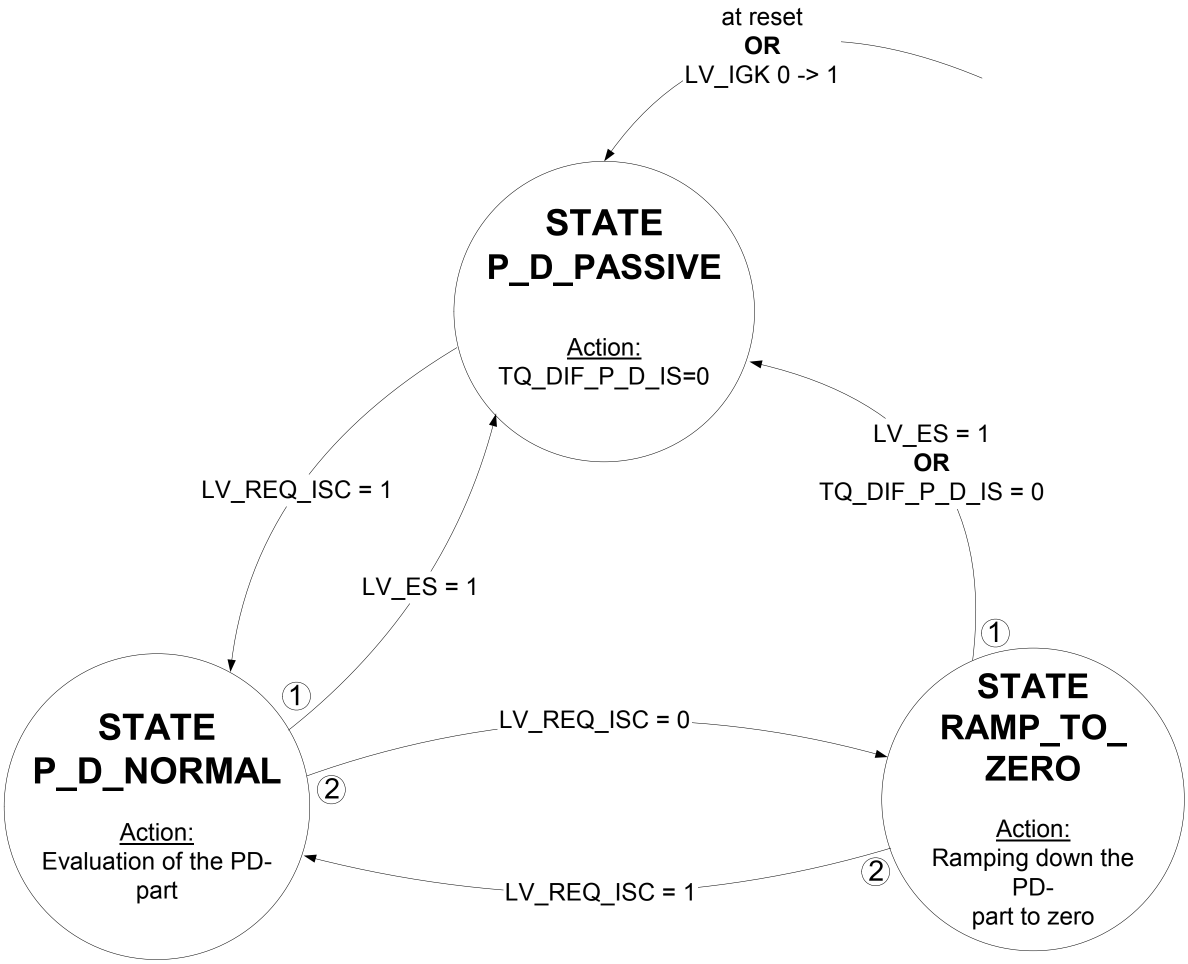


Figure 1: *State of the PD-part*

STATE_P_D_PASSIVE:

Actions:

TQ_DIF_P_D_IS = 0;

Transition conditions:

Transition to P_D_NORMAL: LV_REQ_ISC = 1

STATE_P_D_NORMAL:

Actions:

Signal flow diagram:

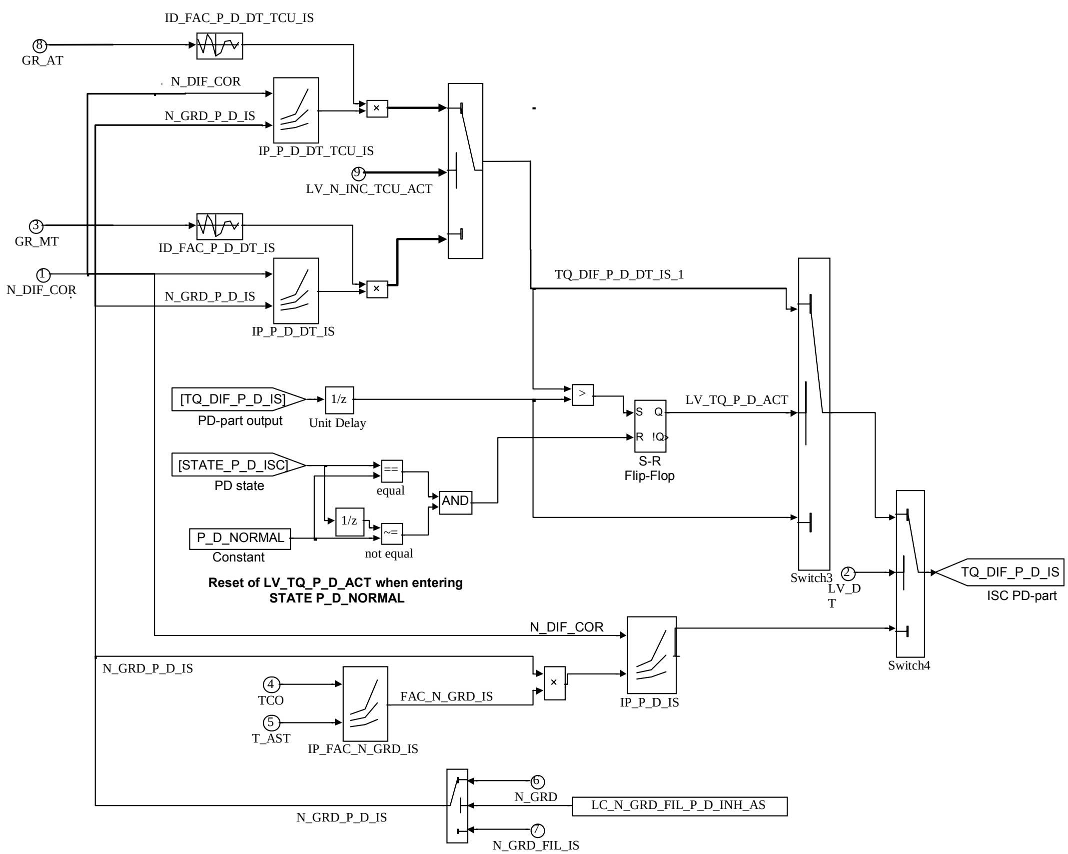


Figure 2: *Actions in state P_D_NORMAL*

Formular section:

Gradient switch:

IF LC_N_GRD_FIL_P_D_INH_AS = 1

THEN N_GRD_P_D_IS = N_GRD – (C_FAC_N_SP_IS_COR_P_D * N_SP_IS_COR_GRD)

ELSE N_GRD_P_D_IS = N_GRD_FIL_IS – (C_FAC_N_SP_IS_COR_P_D * N_SP_IS_COR_GRD_FIL_IS)

Two cases have to be distinguished in this state:

1. If **drivetrain** is **not engaged**, then TQ_DIF_P_D_IS is always the output of the map IP_P_D_IS.
2. If **drivetrain** is **engaged**, then TQ_DIF_P_D_IS is TQ_DIF_P_D_DT_IS_1, **with one exception at controller activation:** If the controller is activated when speed is falling at overspeed, then the controller may produce negative torque requests. These lead to torque jumps, which could be felt as inconvenient by the driver. To avoid these negative torque jumps at PD-ISC request, the output value TQ_DIF_P_D_IS is limited to zero, until the DT-map output value TQ_DIF_P_D_DT_IS_1 has a zero crossing and starts producing positive values, figure 3. This is reached with the logic shown in figure 2. LV_TQ_P_D_ACT is reseted when entering into P_D_NORMAL.

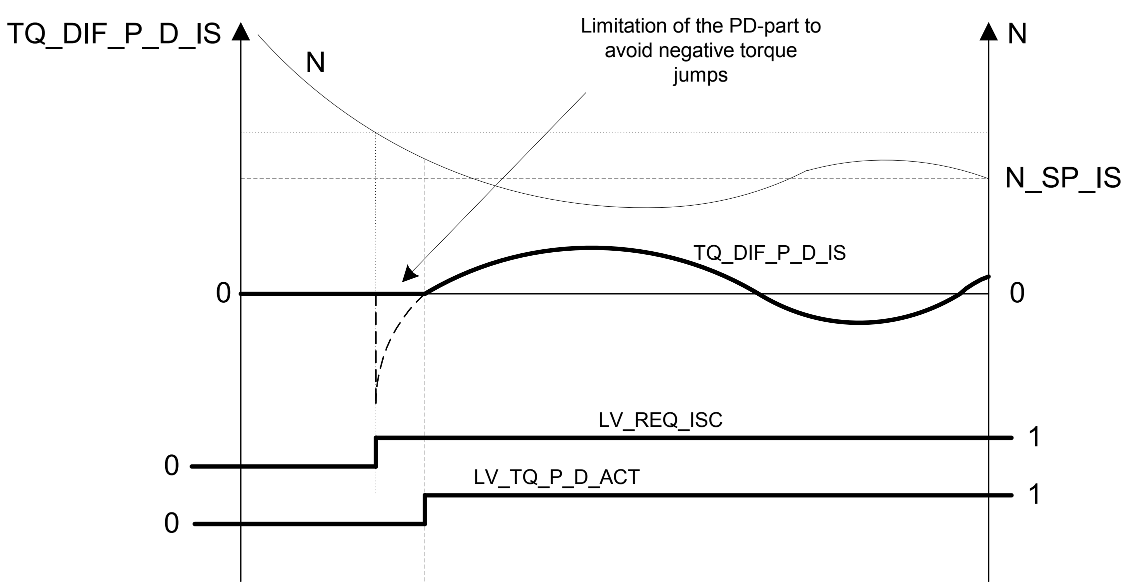


Figure 3: *Activation of the PD-part in case of LV_REQ_ISC = 1* ***AND*** *TQ_DIF_P_D_DT_IS_1 > TQ_DIF_P_D_IS*

IF $TQ\_DIF\_P\_D\_DT\_IS\_1_k$ > $TQ\_DIF\_P\_D\_IS_{k-1}$

OR $TQ\_DIF\_P\_D\_DT\_IS\_1_k$ > C_TQ_MIN_P_D_ACT

THEN LV_TQ_P_D_ACT = 1;

ELSE IF $STATE_k$ == P_D_NORMAL AND $STATE_{k-1}$ ~= P_D_NORMAL

THEN LV_TQ_P_D_ACT = 0;

Either TQ_DIF_P_D_IS is zero or equal to a ramp value (STATE RAMP_TO_ZERO) at activation of the PD-part, TQ_DIF_P_D_IS is always activated if TQ_DIF_P_D_DT_IS_1 is positive or higher than the ramp value.

The calculation of the PD-part is done as described in the following.

Evaluation of the maps:

If LV_N_INC_TCU_ACT = 0
then TQ_DIF_P_D_DT_IS_1 = IP_P_D_DT_IS · ID_FAC_P_D_DT_IS;
else TQ_DIF_P_D_DT_IS_1 = IP_P_D_DT_TCU_IS · ID_FAC_P_D_DT_TCU_IS
end

if LV_DT = 1

then if LV_TQ_P_D_ACT = 1

then TQ_DIF_P_D_IS = TQ_DIF_P_D_DT_IS_1

else TQ_DIF_P_D_IS = $TQ\_DIF\_P\_D\_IS_{k-1}$

else TQ_DIF_P_D_IS = IP_P_D_IS;

Transition conditions:

Transition to P_D_PASSIVE: LV_ES = 1
Transition to RAMP_TO_ZERO: LV_REQ_ISC = 0

Actions during transition to RAMP_TO_ZERO:

TQ_ADD_P_D_RAMP = [0 -$TQ\_DIF\_P\_D\_IS_{k-1}$] / T_RAMP_LIM_P_D_IS * update rate
LV_PAS_RAMP_ACT_P_D_IS = 1

The information about ramp limit operation due to deactivation of ISC is needed in the monitoring level. Therefore the flag LV_PAS_PAMP_ACT_P_D_IS is set.

STATE_RAMP_TO_ZERO:

Actions:

To prevent a torque jump if the PD part is on a certain value and the driver hits the pedal (transition to part load), it´s necessary to reduce the PD-part’s value by a deactivation ramp. The **target value** of the ramp operation is always TQ_DIF_P_D_IS = 0 [Nm].

if |TQ_DIF_P_D_IS| >= | TQ_ADD_P_D_RAMP |

then TQ_DIF_P_D_ISk = TQ_DIF_P_D_ISk-1 + TQ_ADD_P_D_RAMP

else TQ_DIF_P_D_IS = 0

The **minimum value** to be added per update rate while the ramp is active, is the resolution of TQ_ADD_P_D_RAMP. If the theoretically calculated value TQ_ADD_P_D_RAMP is smaller than the resolution, then the software will always limit TQ_ADD_P_D_RAMP to the resolution. This influences the real ramping time, which is in this case smaller than the calibrated one. The ramping slopes become steeper.

If the ramp is active while reentering into idle, STATE_P_D_ISC changes to P_D_NORMAL.

Transition conditions:

Transition to P_D_PASSIVE:	LV_ES = 1 **OR** TQ_DIF_P_D_IS = 0 Nm
Transition to P_D_NORMAL:	LV_REQ_ISC = 1

Actions during transition to P_D_PASSIVE:

LV_PAS_RAMP_ACT_P_D_IS = 0

Actions during transition to P_D_NORMAL:

LV_PAS_RAMP_ACT_P_D_IS = 0

LV_TQ_P_D_ACT = 0

8.2.3.2 Separation of PD-Controller Output (Slow / Fast)

This function is meant to have a possibility to reduce the controller's PD-gain in dependance of TCO to prevent a strong excitement of the engine and so to reduce the danger of an oscillating speed.

Signal flow diagram:

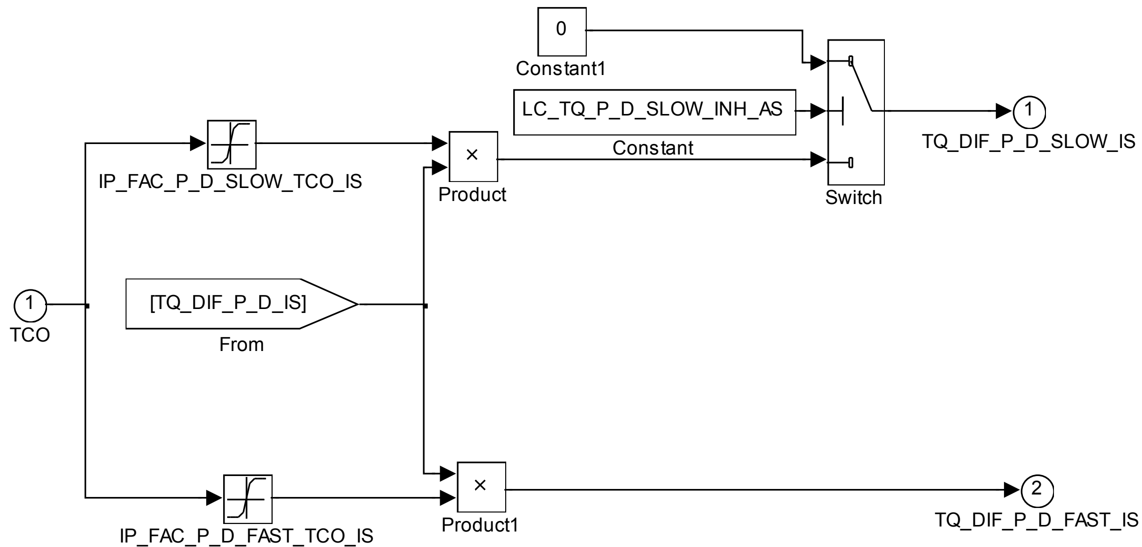


Figure 4: *Seperation into a slow and a fast path for the PD-controller*

For special function test, it´s possible to set TQ_DIF_P_D_SLOW_IS = 0 [Nm] by application system.

Determination of TQ_DIF_P_D_SLOW_IS:

if LC_TQ_P_D_SLOW_INH_AS == 1

then TQ_DIF_P_D_SLOW_IS = 0 [Nm]

else TQ_DIF_P_D_SLOW_IS = TQ_DIF_P_D_IS * IP_FAC_P_D_SLOW_TCO_IS

Determination of TQ_DIF_P_D_FAST_IS:

TQ_DIF_P_D_FAST_IS = TQ_DIF_P_D_IS * IP_FAC_P_D_FAST_TCO_IS

8.2.4 I-Controller

General information:

The integral part is devided into four states:

STATE_I_ISC = I_PASSIVE:
Initialization (at reset or when LV_IGK = 0 → 1; from all other states in case of LV_ES = 1)

STATE_I_ISC = I_NORMAL:
In this state the evaluation of the controller maps and the "normal" integration takes place.

STATE_I_ISC = RAMP_TO_LIM:
Ramp to limits; in this state the integral part is ramped down to the calibratible top/bottom limit values.

STATE_I_ISC = RAMP_TO_PAS:
Ramp to passive value 0 (); in this state the integral part is ramped down to zero in case LV_REQ_ISC = 0.

Application conditions:

Activation:	at every engine state
Point in time of calculation:	This function has to be calculated immediate **before the module "Minimum torque at clutch"** and **after speed and speed gradient determination**.
Initialization:	TQ_DIF_I_IS = 0 (STATE_I_ISC = 0, see the following figure 4).
Update Rate:	20 ms

Formula section:

For special function test, it´s possible to set TQ_ADD_I_IS = 0 Nm or TQ_DIF_I_IS = 0 Nm by application system or diagnostic tool (LV_ISC_INH_EXT_ADJ). The I-controller can be activated/deactivated with the following calibration bits:

IF LC_TQ_DIF_I_INH_AS = 1
or LC_ISC_INH = 1
or LV_ISC_INH_EXT_ADJ = 1
then TQ_DIF_I_IS = 0 Nm;
else **if** C_TQ_ADD_I_INH_AS = 1
then DD_I_IS = 0 Nm

Determination of the top/bottom limit values:

IF LV_DT = 1
THEN TQ_TOL_I_IS = C_TQ_TOL_I_DT_IS
TQ_BOL_I_IS = C_TQ_BOL_I_DT_IS
T_RAMP_LIM_I_IS = C_T_RAMP_LIM_I_DT_IS
ELSE TQ_TOL_I_IS = C_TQ_TOL_I_IS
TQ_BOL_I_IS = C_TQ_BOL_I_IS
T_RAMP_LIM_I_IS = C_T_RAMP_LIM_I_IS

The passive value (**target value**) of the ramp operation is depending on the activation status of the climatic compressor:

IF LV_ACCOUT_RLY = 1
THEN TQ_PAS_I_IS = C_TQ_PAS_I_ACC_IS
ELSE TQ_PAS_I_IS = C_TQ_PAS_I_IS

FUNCTION DESCRIPTION:

Signal flow diagram:

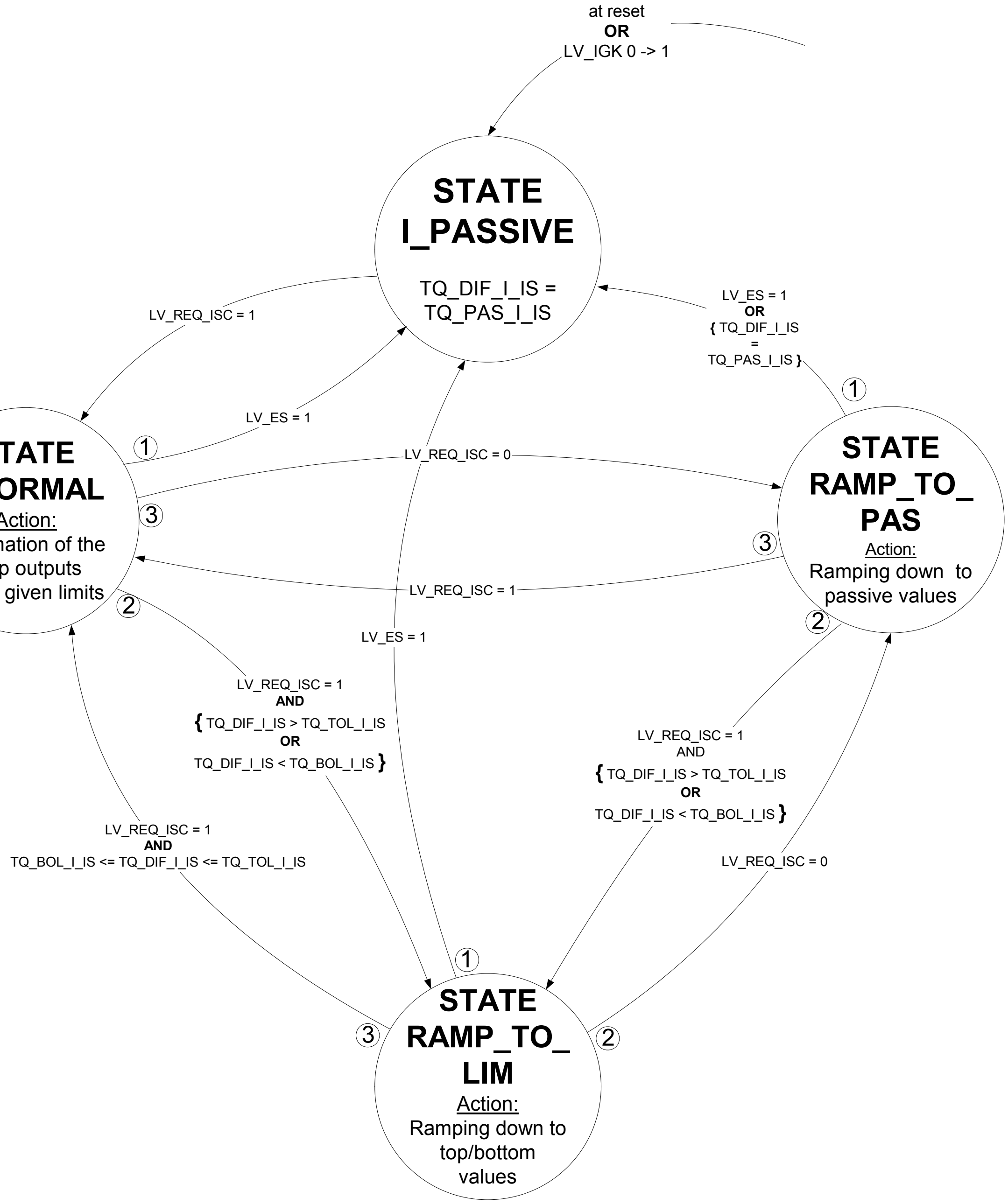


Figure 5: *State diagram of the integral part*

STATE I_PASSIVE

Actions:

TQ_DIF_I_IS = TQ_PAS_I_IS;

Transition conditions:

Transition to I_NORMAL: LV_REQ_ISC = 1

STATE I_NORMAL

Actions:

It can be chosen with LC_N_GRD_FIL_I_INH_AS, if a filtered gradient is used or not:

if LC_N_GRD_FIL_I_INH_AS = 1
then N_GRD_I_IS = N_GRD – (C_FAC_N_SP_IS_COR_I * N_SP_IS_COR_GRD)
else N_GRD_I_IS = N_GRD_FIL – (C_FAC_N_SP_IS_COR_I * N_SP_IS_COR_GRD_FIL_IS)

Calculation:

$TQ\_DIF\_I\_IS_{k} = TQ\_DIF\_I\_IS_{k-1} + TQ\_ADD\_I\_IS_{k}$

Determination of TQ_ADD_I_IS:

In STATE I_NORMAL, see figure 4, the "normal" integration takes place. A torque difference TQ_ADD_I_IS is determined in dependence of the speed deviation N_DIF_COR and the speed gradient N_GRD_I_IS (for LV_DT = 0) and only in dependance of the speed deviation N_DIF_COR for LV_DT = 1, figure 6.

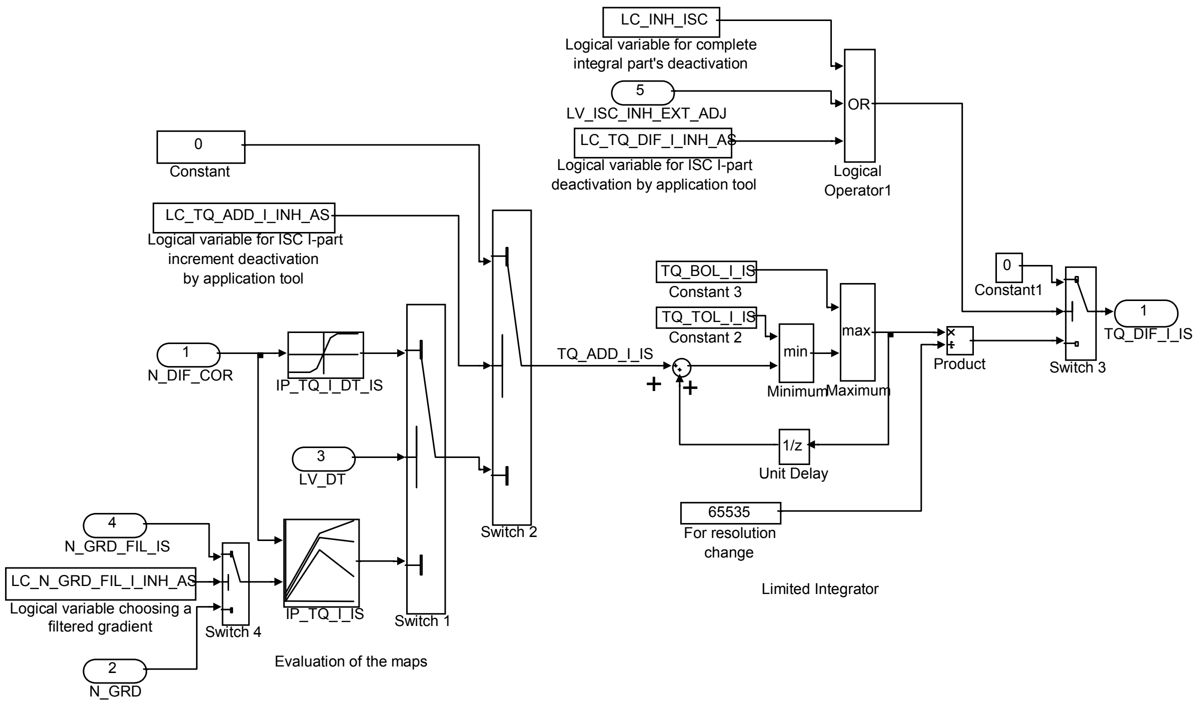


Figure 6: *Determination of TQ_ADD_I_IS and TQ_DIF_I_IS.*

The devision by 65535 is done to change the higher integration resolution to the usual torque resolution.

To reach two important targets, TQ_ADD_I_IS should be calibrated as follows:

- compensation of a speed offset
- automatic precalibration of the integral part: if there is a load active after the driver released the gas pedal (ACC, power steering, etc), then speed falls with a higher gradient into idle range than in comparision to the situation without load. This means, that the integral part should carry higher (positive) values to intercept speed. That means, that with the help of the gradient dependancy in the integral part an automatic precalibration of the TQ_ADD_I_IS (so TQ_DIF_I_IS) can be reached, which is load dependant. It must be said, that this interception function must be synchronized with the PD-part calibration. With the correct calibration, the passive values are not needed any more.

The integration takes place within calibratible boundaries, figure 7.

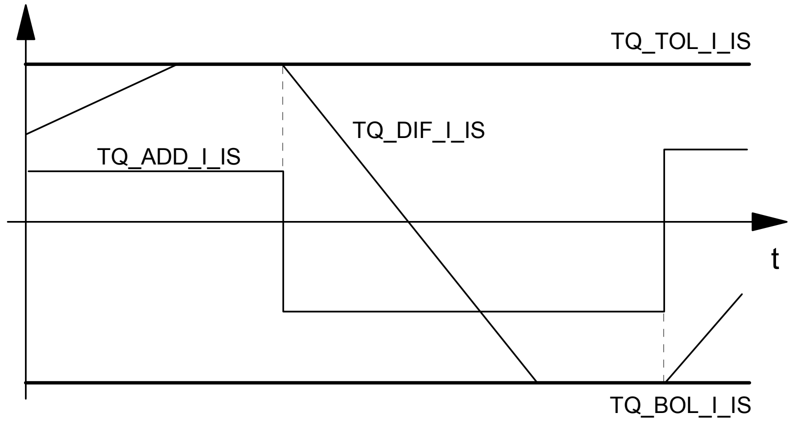


Figure 7: *Limited integration*

Transition conditions:

Transition to I_PASSIVE:	LV_ES = 0	
Transition to RAMP_TO_LIM:	LV_REQ_ISC = 1	**AND**
	[	
	TQ_DIF_I_IS > TQ_TOL_I_IS	**OR**
	TQ_DIF_I_IS < TQ_BOL_I_IS	
	]	
Transition to RAMP_TO_PAS:	LV_REQ_ISC = 0	

Actions during transition to RAMP_TO_LIM:

IF TQ_DIF_I_IS < TQ_BOL_I_IS

THEN TQ_LIM_I_IS = TQ_BOL_I_IS

ELSE TQ_LIM_I_IS = TQ_TOL_I_IS

ENDIF

TQ_ADD_I_RAMP = **[** TQ_LIM_I_IS – TQ_DIF_I_IS **]** / T_RAMP_LIM_I_IS * update rate

Actions during transition to RAMP_TO_PAS:

TQ_ADD_I_RAMP = **[** TQ_PAS_I_IS – TQ_DIF_I_IS **]** / T_RAMP_LIM_I_IS * update rate

LV_PAS_RAMP_ACT_I_IS = 1

STATE_RAMP_TO_LIM

If this state is active and so the integrator output is outside the allowed boundaries, its value is controlled back to the valid boundaries on a ramp.

Actions:

IF TQ_DIF_I_IS < TQ_BOL_I_IS
THEN TQ_LIM_I_IS = TQ_BOL_I_IS
ELSE TQ_LIM_I_IS = TQ_TOL_I_IS
ENDIF

IF ($|$TQ_LIM_I_IS - TQ_DIF_I_IS$_{k-1}|$ > $|$TQ_ADD_I_RAMP$|$)
THEN TQ_DIF_I_IS$_k$ = TQ_DIF_I_IS$_{k-1}$ + TQ_ADD_I_RAMP
ELSE TQ_DIF_I_IS = TQ_LIM_I_IS
ENDIF

The **minimum value** to be added per update rate while the ramp is active, is the resolution of TQ_ADD_I_RAMP. If the theoretically calculated value TQ_ADD_I_RAMP is smaller than the resolution, then the software will always limit TQ_ADD_I_RAMP to the resolution. This influences the real ramping time, which is in this case smaller than the calibrated one. The ramping slopes become steeper.

Transition conditions:

Transition to I_PASSIVE:	LV_ES = 1
Transition to RAMP_TO_PAS:	LV_REQ_ISC = 0
Transition to I_NORMAL:	LV_REQ_ISC = 1 **AND** TQ_BOL_I_IS $\leq$ TQ_DIF_I_IS $\leq$ TQ_TOL_I_IS

Actions during transition to RAMP_TO_PAS:

TQ_ADD_I_RAMP = **[** TQ_PAS_I_IS – TQ_DIF_I_IS **]** / T_RAMP_LIM_I_IS * update rate

LV_PAS_RAMP_ACT_I_IS = 1

STATE RAMP TO PAS

At deactivation of ISC the integrator output is controlled to a passive-value by ramp limit activation. The determination of the ramp takes place in the same way as described before for the top/bottom limit ramps.

Actions:

The ramp limit operation starts at deactivation of the ISC. The necessary increment (**TQ_ADD_I_RAMP**) to reach the required ramp-slope **is only calculated once while transitioning into this state**. This value is valid until deactivation of the ramp limit operation.

While transition into this state, so while the ramp is active, there could be a change in TQ_PAS_I_IS. If in this case TQ_ADD_I_RAMP is negative for example and the passive value is higher than the actual value, then the new requested passive value can never be reached, because TQ_ADD_I_RAMP is only calculated once. Therefore TQ_PAS_I_IS is fixed while transition into this state to prevent a wrong comparison in the following condition:

IF ($|$TQ_PAS_I_IS - TQ_DIF_I_IS$_k|$ > $|$TQ_ADD_I_RAMP$|$)

THEN TQ_DIF_I_IS$_k$ = TQ_DIF_I_IS$_{k-1}$ + TQ_ADD_I_RAMP

ELSE TQ_DIF_I_IS = TQ_PAS_I_IS

ENDIF

The **minimum value** to be added per update rate while the ramp is active, is the resolution of TQ_ADD_I_RAMP. If the theoretically calculated value TQ_ADD_I_RAMP is smaller than the resolution, then the software will always limit TQ_ADD_I_RAMP to the resolution. This influences the real ramping time, which is in this case smaller than the calibrated one. The ramping slopes become steeper.

Transition conditions:

Transition to I_PASSIVE:	LV_ES = 1	**OR**
	TQ_DIF_I_IS = TQ_PAS_I_IS	
Transition to RAMP_TO_LIM:	LV_REQ_ISC = 1	**AND**
	[	
	TQ_DIF_I_IS > TQ_TOL_I_IS	**OR**
	TQ_DIF_I_IS < TQ_BOL_I_IS	
	]	
Transition to I_NORMAL:	LV_REQ_ISC = 1	

Actions during transition to RAMP_TO_LIM:

IF TQ_DIF_I_IS < TQ_BOL_I_IS

THEN TQ_LIM_I_IS = TQ_BOL_I_IS

ELSE TQ_LIM_I_IS = TQ_TOL_I_IS

ENDIF

TQ_ADD_I_RAMP = **[** TQ_LIM_I_IS – TQ_DIF_I_IS **]** / T_RAMP_LIM_I_IS * update rate

LV_PAS_RAMP_ACT_I_IS = 0

Actions during transition to I_PASSIVE:

LV_PAS_RAMP_ACT_I_IS = 0

Actions during transition to I_NORMAL:

LV_PAS_RAMP_ACT_I_IS = 0

If idle speed control is reacitvated during the ramp limit operation then the integrator starts at the current TQ_DIF_I_IS value (Transition to I_NORMAL).

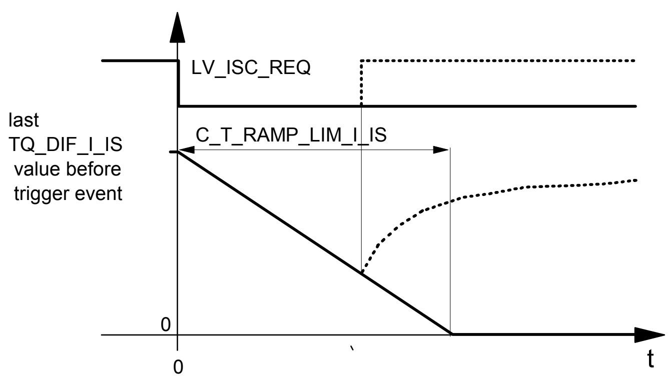


Figure 6: *Reactivation of the controller in case on an active ramp: the integral part starts again at the last ramp value (Transition to I_NORMAL)*

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_P_D_DT_IS	12x8	0 ... FFFF H	-1024 ... 1023.97	0.03125	Nm
LDPM_N_DIF_COR_P_D_DT_IS	12	0 .. FFFF H	-32768 .. 32767	1	1/min
LDPM_N_GRD_P_D_DT_IS	8	0 .. FF H	-4096 ... 4064	32	1/min/s
P- and D-component for idle speed controller at closed drive train					
IP_P_D_IS	12x8	0 ... FFFF H	-1024 ... 1023.97	0.03125	Nm
LDP_N_DIF_COR_P_D_IS	12	0 .. FFFF H	-32768 .. 32767	1	1/min
LDP_N_GRD_P_D_IS	8	0 .. FF H	-4096 ... 4064	32	1/min/s
P- and D-component for idle speed controller at open drive train					
ID_FAC_P_D_DT_IS	8	0 ... FFFF H	0 ... 1.9999	3.05E-5	-
LDPM_GR_MT_2	8	0 ... FF H	0 ... 255	1	-
Weighting factor for IP_P_D_DT_IS					
IP_FAC_N_GRD_IS	8x8	0 ...FF	0 ... 1.992	0.08	-
LDP_TCO__FAC_N_GRD_IS	8	0 ... FE	-48 ... 142.5	0.75	°C
LDP_T_AST__FAC_N_GRD_IS	8	0 ... FFFF	0 ... 6553.5	0.1	s
Weighting factor for N_GRD input for IP_P_D_IS					
LC_TQ_DIF_P_D_INH_AS	1	0, 1	0, 1	1	-
Bit for inhibiting the PD-part - default value = 0					
LC_ISC_INH	1	0 ... 1 H	0 ... 1	1	-
Bit for inhibiting the PD- **and** integral part - default value = 0					
C_FAC_N_SP_IS_COR_P_D	1	0 .. FF H	0 ... 0.996	0.0039	-
Weighting factor to take into account N_SP_IS_COR gradient in N_GRD_P_D_IS					
C_FAC_N_SP_IS_COR_I	1	0 .. FF H	0 ... 0.996	0.0039	-
Weighting factor to take into account N_SP_IS_COR gradient in N_GRD_I_IS					
C_CRLC_N_GRD	1	0 .. FFFF H	0 ... 0.999985	1.52E-5	-
Filter correlation constant					
C_T_RAMP_LIM_P_D_IS	1	0 .. FFFF H	0 ... 655.35	0.01	s
Time constant for ramp limit of P-, D-controller output in case of idle speed controller passive					
C_T_RAMP_LIM_P_D_DT_IS	1	0 .. FFFF H	0 ... 655.35	0.01	s
Time constant for ramp limit of P-, D-controller output in case of idle speed controller passive and LV_DT=1					
LC_N_GRD_FIL_P_D_INH_AS	1	0 ... 1 H	0 ... 1	1	-
Logical variable for switching between N_GRD and N_GRD_FIL_IS					
C_TQ_MIN_P_D_ACT	1	8000 ... 7FFF H	-1024 ... 1023.97	0.03125	Nm
Minimum torque to set LV_TQ_P_D_ACT = 1					
IP_FAC_P_D_SLOW_TCO_IS	8x6	0 ... FFFF H	0 ... 1.9999	30.53E-6	-
LDPM_TCO_IS_1	8	0 ... FE H	-48 ... 142.5	0.75	°C
LDP_TQ_DIF_P_D_IS	6	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
TCO, P-D request depending factor for slow-path of torque coordination					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_P_D_FAST_TCO_IS	8x4	0 ... FFFF H	0 ... 1.9999	30.53E-6	-
LDPM_TCO_IS_1	8	0 ... FE H	-48 ... 142.5	0.75	°C
LDP_TQ_ADD_CH	4	8000 ... 7FFF H	-1024 ... 1023.97	0.03125	Nm
TCO, Catalyst Heating Torque Reserve depending factor for fast-path of torque coordination					
LC_TQ_P_D_SLOW_INH_AS	1	0, 1	0, 1	1	-
Logical variable for deactivation of TQ_DIF_P_D_IS - default value at initialisation = 0					
IP_TQ_I_IS	12x8	0 ... FFFF H	-4 ... 3.999	1.22E-4	Nm
LDP_N_DIF_COR_TQ_I_IS	12	0 .. FFFF H	-32768 .. 32767	1	1/min
LDP_N_GRD_TQ_I_IS	8	0 .. FF H	-4096 ... 4064	32	1/min/s
I-component for idle speed controller at open drive train					
C_TQ_BOL_I_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Bottom limit for I-controller output					
C_TQ_BOL_I_DT_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Bottom limit for I-controller output and drive train closed					
C_TQ_TOL_I_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Top limit for I-controller output					
C_TQ_TOL_I_DT_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Top limit for I-controller output and drive train closed					
IP_TQ_I_DT_IS	12	0 ... FFFF H	-4 ... 3.999	1.22E-4	Nm
LDP_N_DIF_COR__TQ_I_DT_IS	12	0 .. FFFF H	-32768 .. 32767	1	1/min
I-component for idle speed controller at closed drive train					
LC_TQ_ADD_I_INH_AS	1	0, 1	0, 1	1	-
Logical variable for ISC I-part map output deactivation - default value at initialisation = 0					
LC_TQ_DIF_I_INH_AS	1	0, 1	0, 1	1	-
Logical variable for ISC I-part deactivation (completely) - default value at initialisation = 0					
LC_N_GRD_FIL_I_INH_AS	1	0, 1	0, 1	1	-
Logical variable for changing between N_GRD and N_GRD_FIL_IS					
C_TQ_PAS_I_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Final passive value for I-controller output					
C_TQ_PAS_I_ACC_IS	1	8000 .. 7FFF H	-1024 ... 1023.97	0.03125	Nm
Final passive value for I-controller output and air conditioning compressor active					
C_T_RAMP_LIM_I_IS	1	0 .. FFFF H	0 ... 1310.7	0.02	s
Time constant for ramp limit of I-controller output in case of idle speed controller passive Resolution is directly coupled with update rate					
C_T_RAMP_LIM_I_DT_IS	1	0 .. FFFF H	0 ... 1310.7	0.02	s
Time constant for ramp limit of I-controller output in case of idle speed controller passive and drive train engaged Resolution is directly coupled with update rate					
IP_P_D_DT_TCU_IS	12x8	0 ... FFFF H	-1024 ... 1023.97	0.03125	Nm
LDPM_N_DIF_COR_P_D_DT_IS	12	0 .. FFFF H	-32768 .. 32767	1	1/min
LDPM_N_GRD_P_D_DT_IS	8	0 .. FF H	-4096 ... 4064	32	1/min/s
P- and D-component for idle speed controller at closed drive train					
ID_FAC_P_D_DT_TCU_IS	8	0 ... FFFF H	0 ... 1.9999	3.05E-5	-
LDP_GR_AT__P_D_DT_IS	8	0 ... FF H	0 ... 255	1	-
Weighting factor for IP_P_D_DT_TCU_IS					

8.3 Application Incidences for Idle Speed Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_REQ_ISC	V/O	0, 1 H	0, 1	1	-
Logical variable for idle speed controller activatIon request					
STATE_REQ_ISC	V	0 H 1 H 2 H 3 H 4 H	ISC_PASSIVE AFTER_START NORMAL TRAILING_THR PART_LOAD	-	-
Actual state for ISC request					
N_GRD_FIL_ISC_ACT_AST	V/O	80 ... 7FH	-4096 ... 4064	32	rpm/s
Filtered gradient for ISC after start activation					
LV_NEG_N_GRD_FIL_MEM	V	0, 1 H	0, 1	1	-
Memory bit for falling gradient after start					
T_CTR_PU_ISC_ACT	V/O	0...FFFFH	0...655.35	0.01	s
Time counter for ISC activation in PU					

Input data:

LV_ST	LV_IS	N	N_GRD
LV_PU	N_SP_IS	VS	LV_DT
LV_PL			

Application conditions:

Activation: **IF** LC_ISC_ACT_CDN_SWI == 1
THEN
Use these application incidences
ELSE
LV_REQ_ISC = LV_IS;
ENDIF;

At reset: STATE_REQ_ISC = ISC_PASSIVE

Update rate: 10 ms

8.3.1 General

The activation of the idle speed controller depends on different driving situations. Four states are distinguished. The actual status of the idle speed activation is stored in STATE_REQ_ISC.

State 0: Idle speed controller is passive **(STATE_REQ_ISC = ISC_PASSIVE)**.

State 1: Activation of the ISC after start **(STATE_REQ_ISC = AFTER_START)**.

State 2: "Normal activation": Activiation of the ISC in case of LV_IS = 1 **(STATE_REQ_ISC = NORMAL)**

State 3: Activation of the ISC if speed remains stationary at a certain value in trailing throttle state **(STATE_REQ_ISC = TRAILING_THR)**

State 4: Activation of the ISC in part load in case of underspeed, **(STATE_REQ_ISC = PART_LOAD)**.

The following state diagram illustrates the connection between these three cases, figure 1.

State machine:

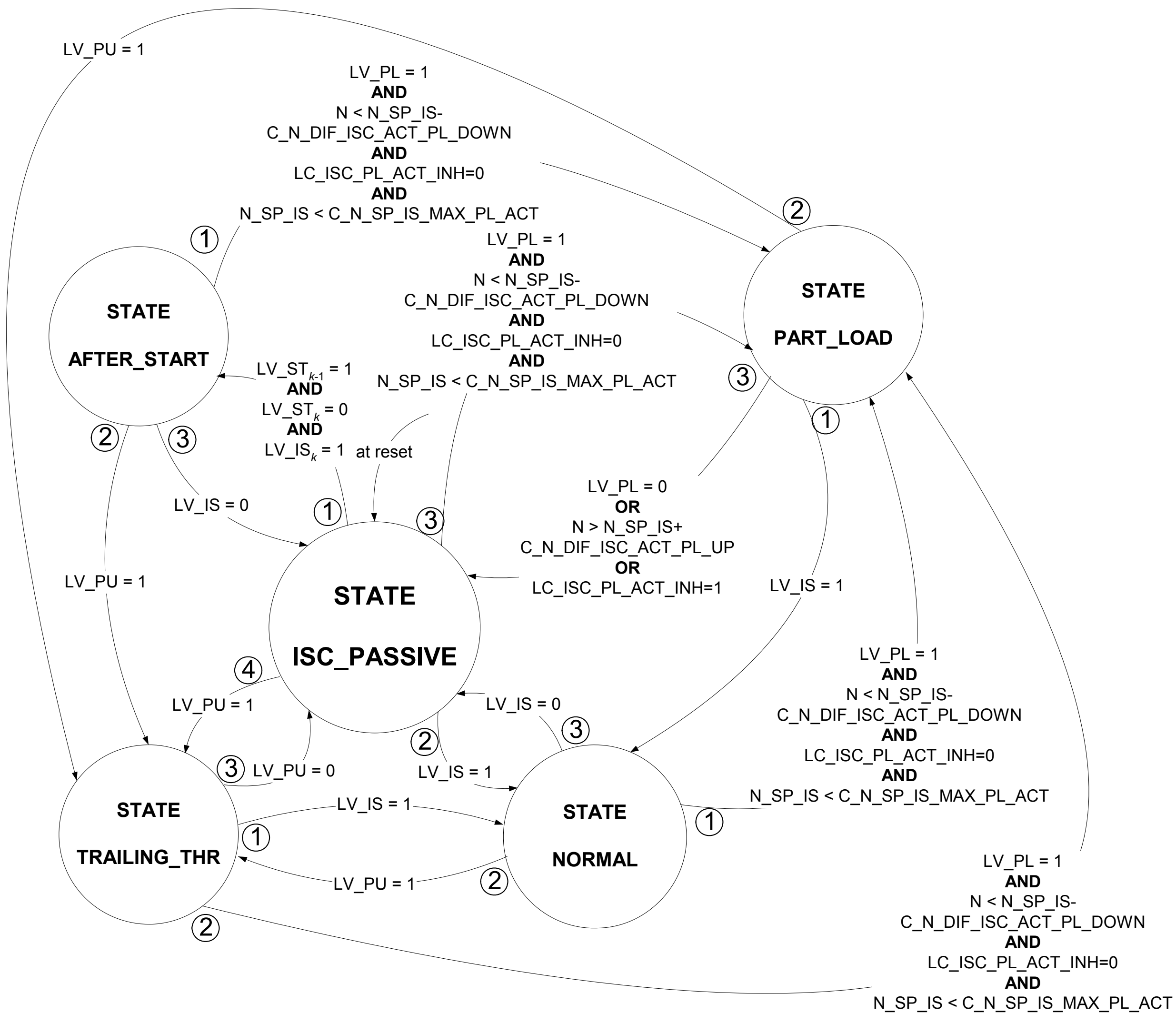


Figure 1: *State diagram of the ISC activation.*
The numbers at different transitions of one state to another represent the priority in which the transitions have to be checked.

8.3.2 ISC Passive (STATE ISC_PASSIVE)

FUNCTION DESCRIPTION:

This state is the idle speed controller passive state. The controller is deactivated if there is a transition to this state.

Formula section:

Action: LV_REQ_ISC = 0

Transition: see figure 1

Action during transition to AFTER_START: LV_NEG_N_GRD_FIL_MEM = 0
N_GRD_FIL_ISC_ACT_AST = 0

8.3.3 Checking ISC activation conditions after start (Entry into IS after ST, STATE AFTER_START)

FUNCTION DESCRIPTION:

After transition from engine state ST to IS, the activation of the ISC is requested if speed reaches a calibratible offset from N_SP_IS with a negative gradient N_GRD, figure 2. If speed does not reach this offset after decreasing and increasing again, the controller is also activated.

Transition: see figure 1

Action:

IF LC_ISC_ACT_AST_CDN_SWI == 1
THEN
LV_REQ_ISC = 1
ELSE
{

% Calculation of a filtered gradient for after start activation

$N\_GRD\_FIL\_ISC\_ACT\_AST_{k} = N\_GRD\_FIL\_ISC\_ACT\_AST_{k-1} +$
$+ C\_CRLC\_N\_GRD\_FIL\_ISC\_ACT\_AST \cdot \{ N\_GRD_{k} - N\_GRD\_FIL\_ISC\_ACT\_AST_{k-1} \}$

% If N reaches the calibratible offset N_SP_IS + C_N_DIF_ISC_ACT_AST
% with a negative gradient, then the controller is activated. If
% speed increases again *before* reaching N_SP_IS + C_N_DIF_ISC_ACT_AST,
% then the controller is activated, too. The falling gradient
% is layed up in LV_NEG_N_GRD_FIL_MEM.

IF N_GRD_FIL_ISC_ACT_AST < 0
THEN
LV_NEG_N_GRD_FIL_MEM = 1

IF **{** N < N_SP_IS + C_N_DIF_ISC_ACT_AST **}**
THEN
LV_REQ_ISC =1
ENDIF

ELSEIF **{** N_GRD_FIL_ISC_ACT_AST >= 0 **}** **AND**
{ LV_NEG_N_GRD_FIL_MEM == 1**}**

THEN
LV_REQ_ISC =1

ENDIF
}

After setting LV_NEG_N_GRD_FIL_MEM, its value will not be changed until the after start is left to STATE 0.

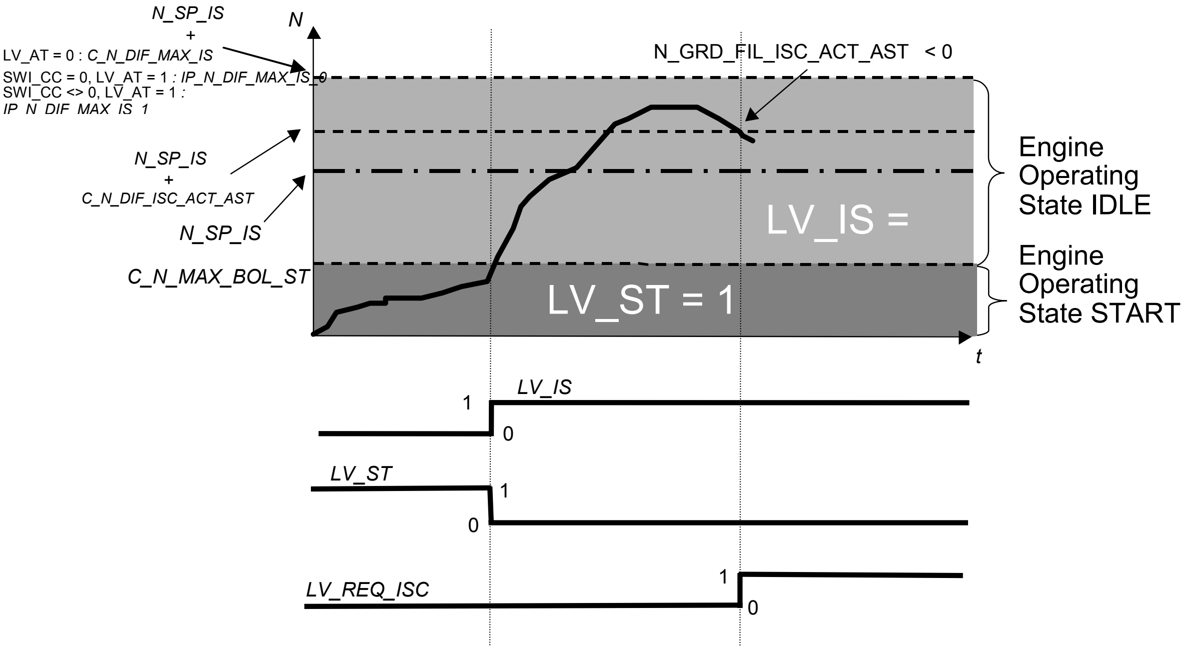


Figure 2: *ISC activation after start*

8.3.4 Checking ISC activation conditions for "normal activation", STATE NORMAL)

The idle speed controller is activated on principle in case of LV_IS = 1 (except STATE 1 is active).

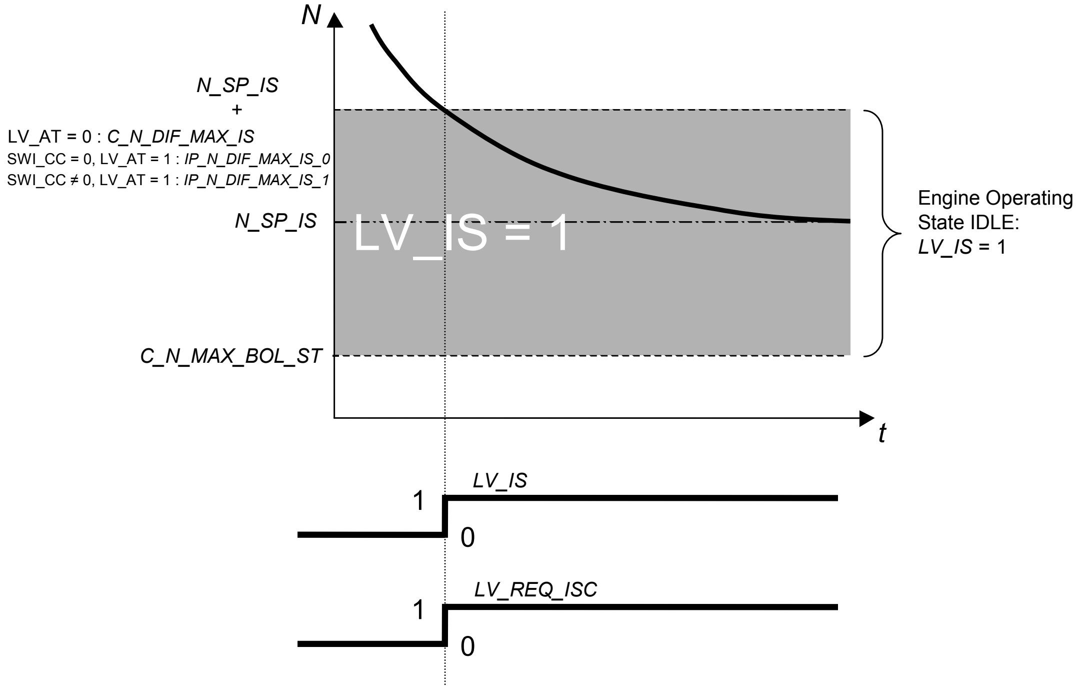


Figure 3: *Activation of the ISC in case of LV_IS = 1*

Formula section:

Transition: see figure 1

Action: LV_REQ_ISC = 1;

8.3.5 ISC activation in case of low engine speed decrease in trailing throttle state (LV_PU = 1, STATE TRAILING_THR)

If speed decrease in trailing throttle state is very low or speed does not decrease, the idle speed controller is activated after a calibratible time *C_T_DLY_ISC_ACT_PU*. The condition for that is a standing vehicle (VS=0). The counter is reseted and started when entering PU ($LV\_PU_{k-1} = 0 \rightarrow LV\_PU_k = 1$), figure 5.

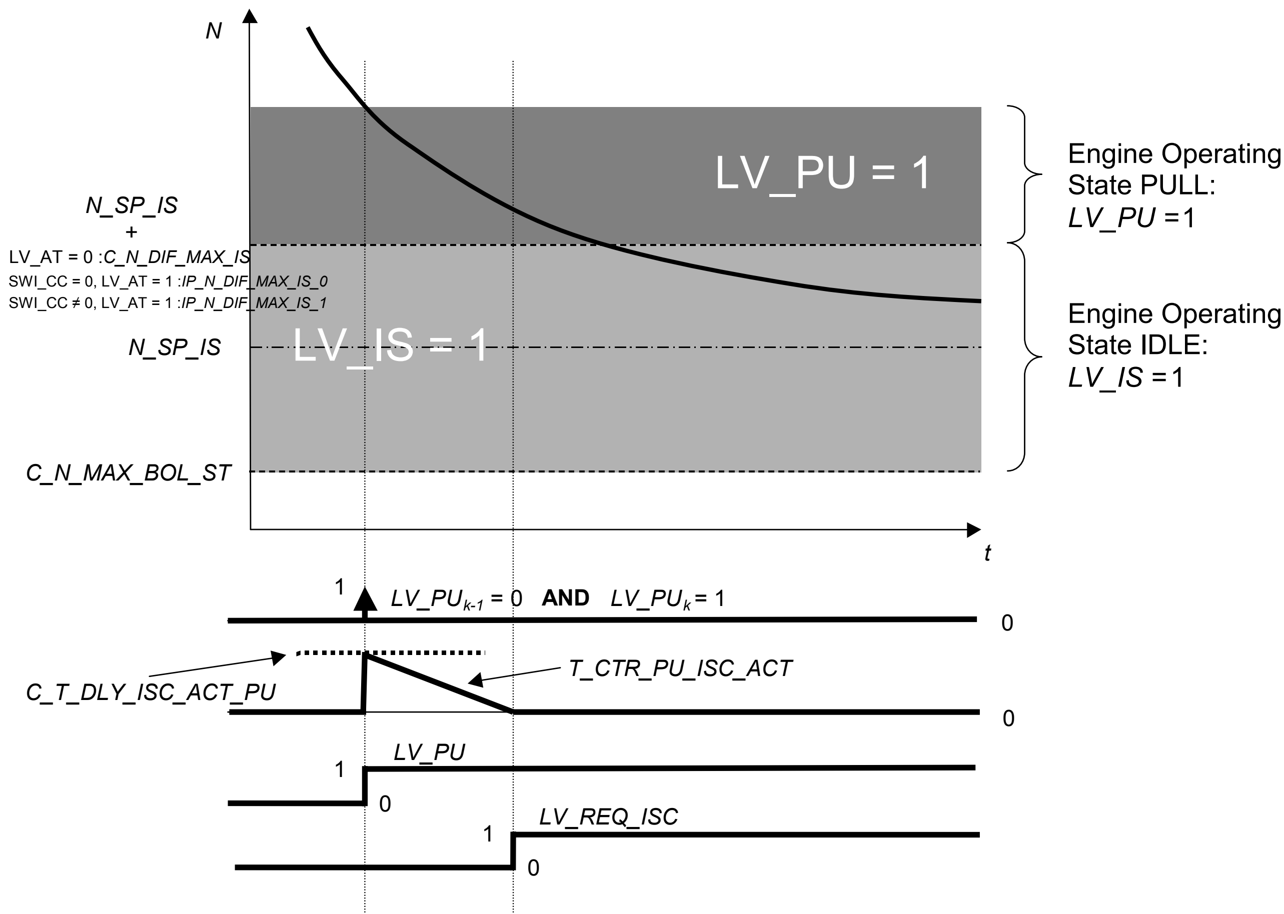


Figure 4: *Activation of the ISC in PU in case the speed decrease is too low and* ***the vehicle does not move (VS = 0)****. The counter starts from a tunable value C_T_DLY_ISC_ACT_PU and stops at zero.*

Formular section:

Transition: see figure 1

Action: *Initialize T_CTR_PU_ISC_ACT with C_T_DLY_ISC_ACT_PU and ramp it down*

IF **{** VS == 0 **OR** LV_DT == 0 **}** **AND** **{** T_CTR_PU_ISC_ACT == 0 **}**
THEN
LV_REQ_ISC = 1
ELSE
LV_REQ_ISC = 0
ENDIF

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_ISC_ACT_CDN_SWI	1	0, 1 H	0, 1	1	-
Calibratible logical bit for switching between different activation conditions					
LC_ISC_ACT_AST_CDN_SWI	1	0,1 H	0, 1	1	-
Switch for using different after start ISC activation conditions					
C_N_DIF_ISC_ACT_AST	1	8000 ... 7FFFH	-32768 ... 32767	1	rpm
Additive term to define the low limit for ISC activation after start					
C_CRLC_N_GRD_FIL_ISC_ACT_AST	1	0 ... FFFF H	0 ... 0.999985	1.52E-5	-
Filter correlation factor for speed gradient filtering in connection with after start activation of ISC					
C_T_DLY_ISC_ACT_PU	1	0...FFFFH	0...655.35	0.01	s
Time limit for ISC activation in PU					

8.3.6 Conditions for ISC part load activation (STATE PART_LOAD)

In part load, the driver or cruise control torque request can be so small, that the idle speed setpoint N_SP_IS cannot be held and speed is decreasing. In this case, the idle speed controller is used to guaranty N >= N_SP_IS. The conditions for activating the ISC in this case depend on the speed and a calibratible hysteresis, figure 6.

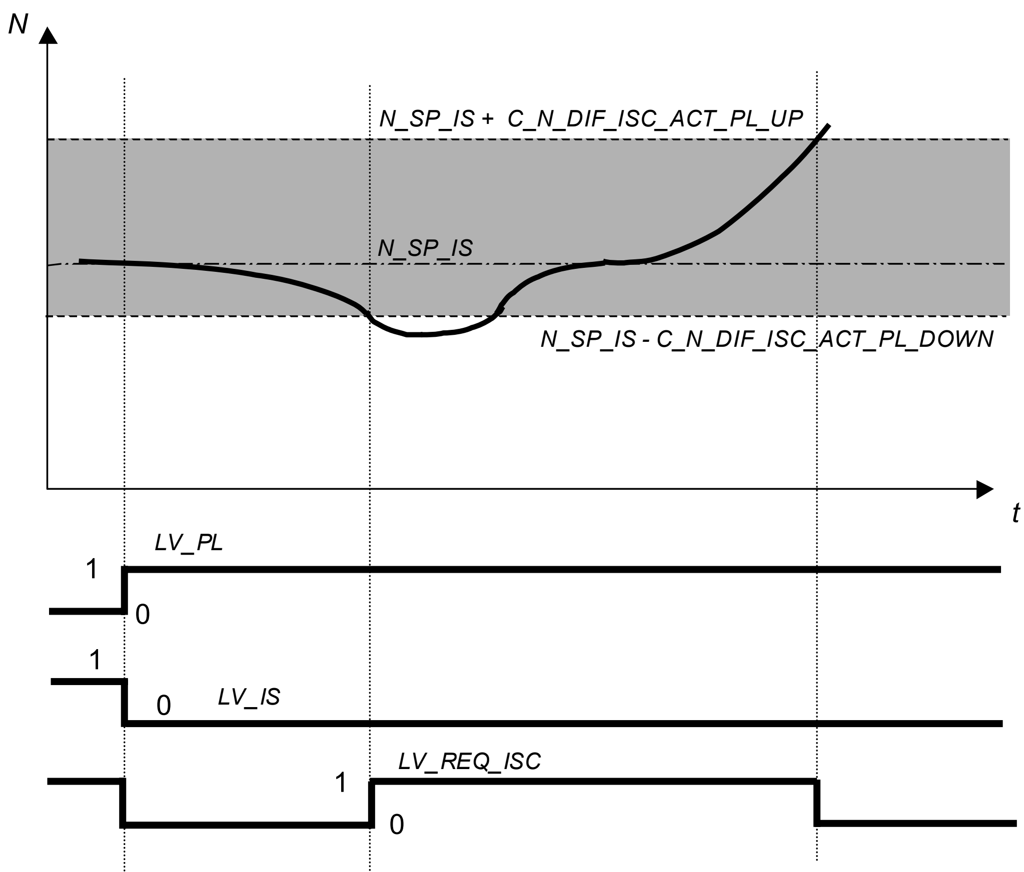


Figure 5: *Activating the ISC in part load*

Formula section:

Transition: see figure 1

Action: LV_REQ_ISC = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DIF_ISC_ACT_PL_DOWN	1	8000 ... 7FFFH	-32768 ... 32767	1	rpm
Additive term to define the low limit for ISC activation in PL					
C_N_DIF_ISC_ACT_PL_UP	1	8000 ... 7FFFH	-32768 ... 32767	1	rpm
Additive term to define the upper limit for ISC deactivation in PL					
C_N_SP_IS_MAX_PL_ACT	1	0...1FE0H	0...8160	1	rpm
Speed setpoint threshold for part load activation					
LC_ISC_PL_ACT_INH	1	0...1 H	0...1	1	-
Calibratible bit to inhibit ISC activation in part load					

8.4 Idle Speed Adaptation

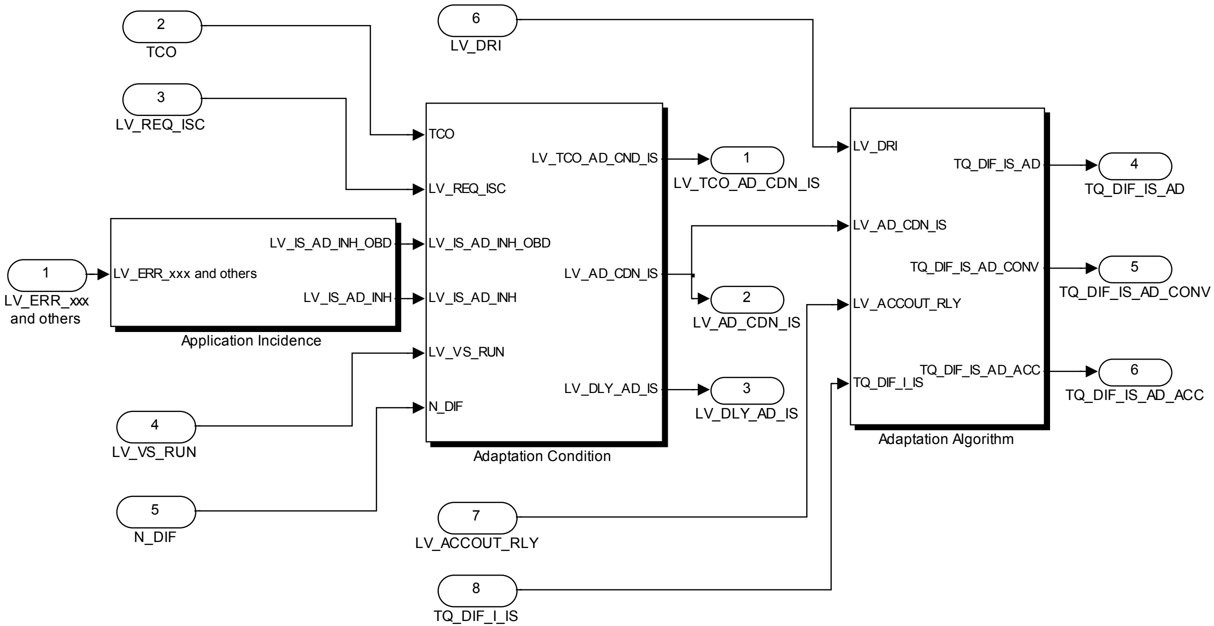


Figure 1: Idle Speed Adaptation, Overview

Functional Description:

The present adaptation is introduced to compensate spread and series variation of engine, air conditioning compressor and hydraulic converter torque losses. It is calculated during idle speed, using the integral part of the idle speed controller for the adaptation of three different parameters:

- TQ_DIF_IS_AD is the adaptation of the basic engine torque losses, without taking the drivetrain into account (disengaged gear),
- TQ_DIF_IS_AD_CONV is an adaptation of the hydraulic converter torque request at idle speed (for automatic transmission only),
- TQ_DIF_IS_AD_ACC is an adaptation value which consideres two adaptation cases depending on the actual situation:

  • air conditioning compressor and converter active at the same time
  • only air conditioning compressor active

The function consists of the following blocks:

- Application Incidence: Checks whether a relevant error has occurred.
- Adaptation Condition: Checks whether the adaptation conditions are fulfilled.
- Adaptation Algorithm: Calculation of the specific adaptation parameters.

These parameters are integrated directly in the specific functions, i.e. the adaptation of the converter torque TQ_DIF_IS_AD_CONV is used in the function "Converter torque" ...

Adaptation parameters for new torque consuming components should be integrated in the same way into the adaptation algorithm module.

8.5 Application Incidence

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_IS_AD_INH_OBD	V/O	0...1H	0...1	1	-
Application incidence value					
LV_IS_AD_INH	V/O	0...1H	0...1	1	-
Application incidence value (additional adaptation inhibitation conditions, project specific)					

Input data:

LV_ERR_IV[NC_CYL_NR]	LV_ERR_IGC	LV_ERR_TCO	LV_ERR_EL_CPS
LV_ERR_TIA_IM	LV_ERR_TIA_CYL	LV_ERR_MAF	LV_ERR_VS
LV_ERR_TIA_THR	LV_MIS_STATE_A	LV_ERR_MEC_CPS	LV_ERR_LAM_LIM_i
LV_ERR_TAM	LV_TQ_ADD_ENG_STALL_ACT	LV_ERR_MAP	NC_ETC_CONF

FUNCTION DESCRIPTION:

Application conditions:

Initialization: at reset

LV_IS_AD_INH_OBD = 0

LV_IS_AD_INH = 0

Recurrence: 100 ms

Activation: at every engine state

Formula section:

IF NC_ETC_CONF = 1
THEN LV_IS_AD_INH = LV_TQ_ADD_ENG_STALL_ACT
ELSE LV_IS_AD_INH = 0
ENDIF

LV_IS_AD_INH is introduced to allow project specific adaptation inhibitation conditions. It is sampled in the adaptation conditions.

Signal flow diagram:

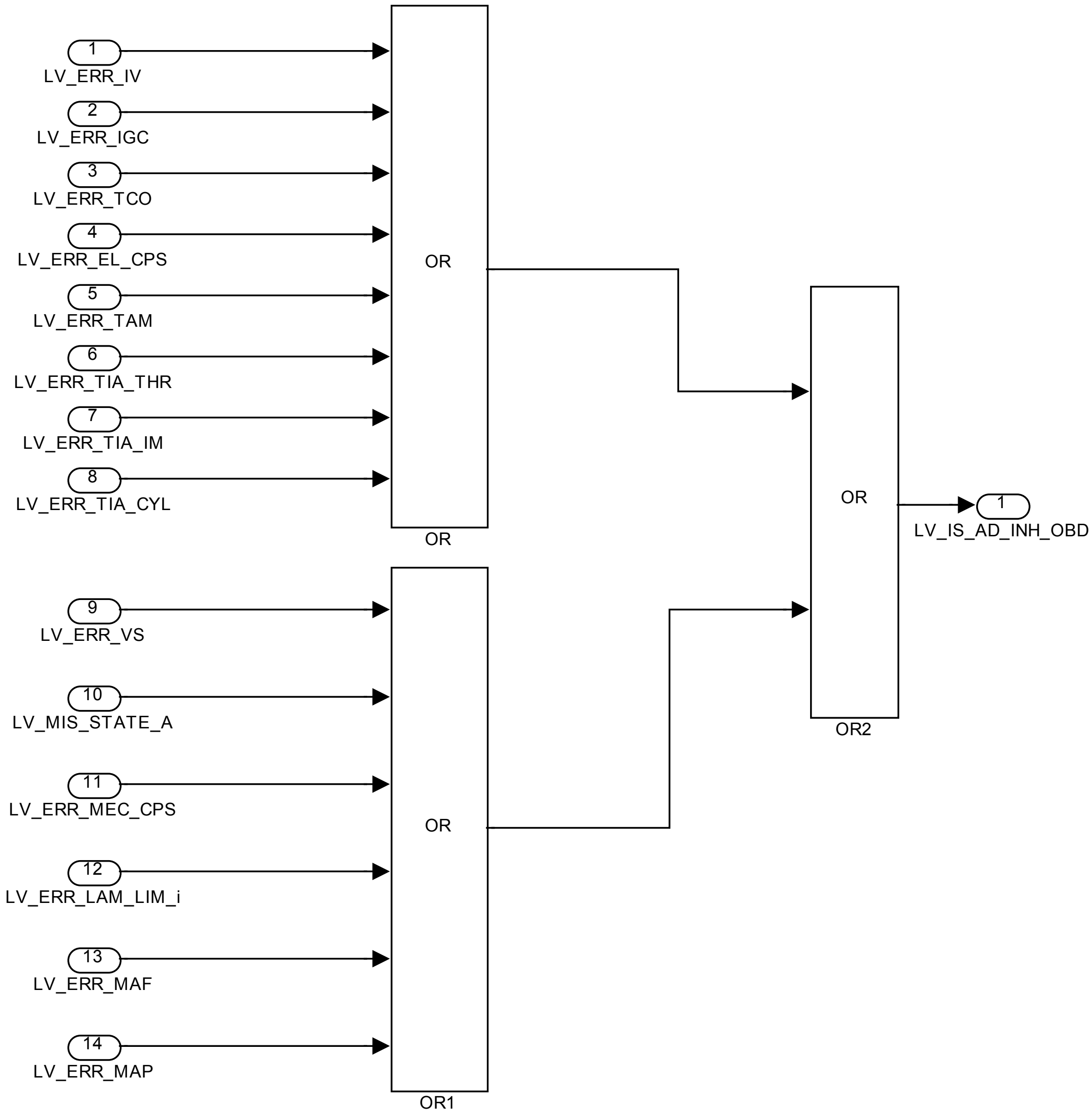


***Figure 1**: Application Incidence; calculation of LV_IS_AD_INH_OBD*

Description:

The block *Application Incidence* considers the following error bits:

LV_ERR_IV, LV_ERR_IGC, LV_ERR_TCO, LV_ERR_EL_CPS, LV_ERR_TAM, LV_ERR_TIA_THR, LV_ERR_TIA_IM, LV_ERR_TIA_CYL, LV_ERR_MAF, LV_ERR_MAP, LV_ERR_VS,, LV_MIS_STATE_A, LV_ERR_MEC_CPS, LV_ERR_LAM_LIM_i

with $i \in N$.

If one of these error bits is set, LV_IS_AD_INH_OBD = 1.
Adaptation is not allowed, if LV_IS_AD_INH_OBD = 1 (see application conditions).

8.6 Adaptation Condition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_AD_CDN_IS	V/O	0...1H	0...1	1	-
Adaptation condition					
LV_TCO_AD_CDN_IS	V/O	0...1H	0...1	1	-
Logical bit “Coolant temperature in operating range”					
LV_DLY_AD_IS	V/O	0...1H	0...1	1	-
Logical bit “Idle speed has settled for a given time delay”					

Input data:

LV_REQ_ISC	LV_VS_RUN	N_DIF	TCO
LV_IS_AD_INH_OBD	LV_IS_AD_INH		

FUNCTION DESCRIPTION:

Application condtions:

Activation: at every engine state

Deactivation: -

Initialization: LV_AD_CDN_IS = 0

LV_TCO_AD_CDN_IS = 0

LV_DLY_AD_IS = 0

Recurrence: 100 ms

Signal flow diagram:

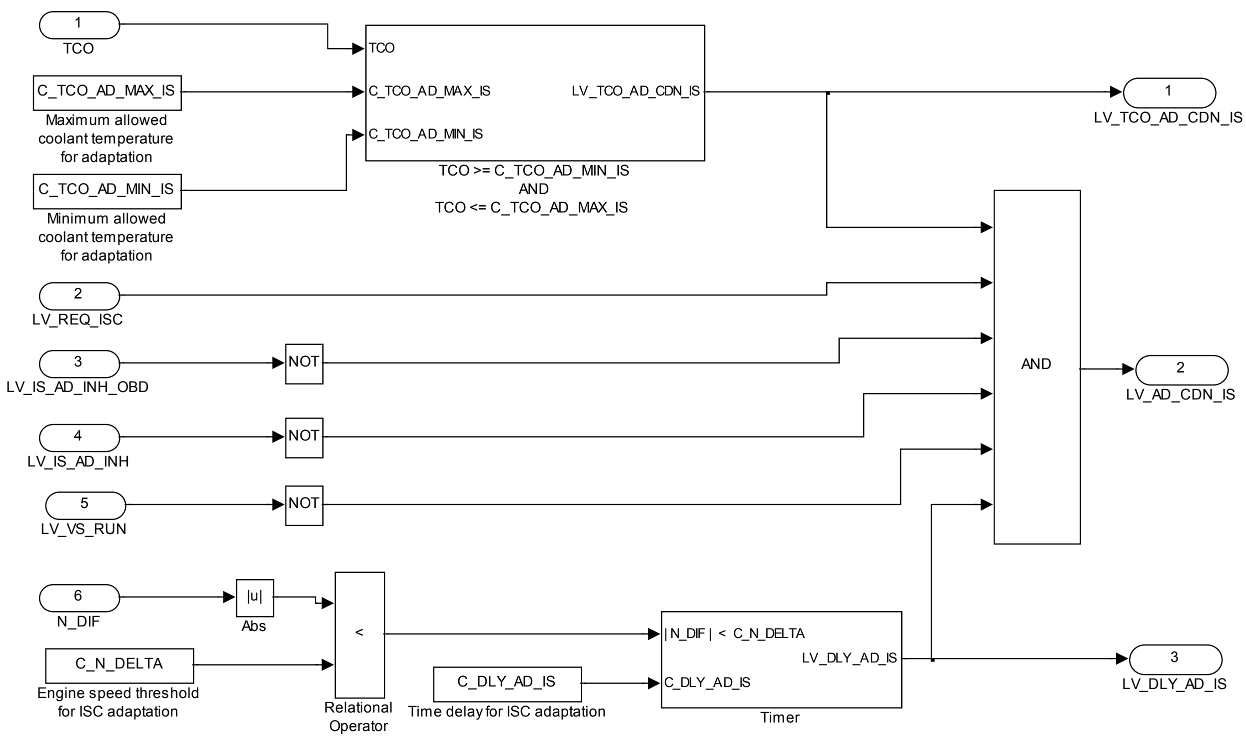


***Figure 1**: Adaptation Conditions*

Formula section:

The adaptation process is released when all of the following conditions are fulfilled (LV_AD_CDN_IS = 1):

- engine running in idle state (LV_REQ_ISC = 1)
- idle speed has settled for a given time delay (T_DLY_IS_AD is the timer value):

IF { ABS(N_DIF) < C_N_DELTA **AND** T_DLY_IS_AD >= C_DLY_AD_IS }

THEN

LV_DLY_AD_IS = 1

END

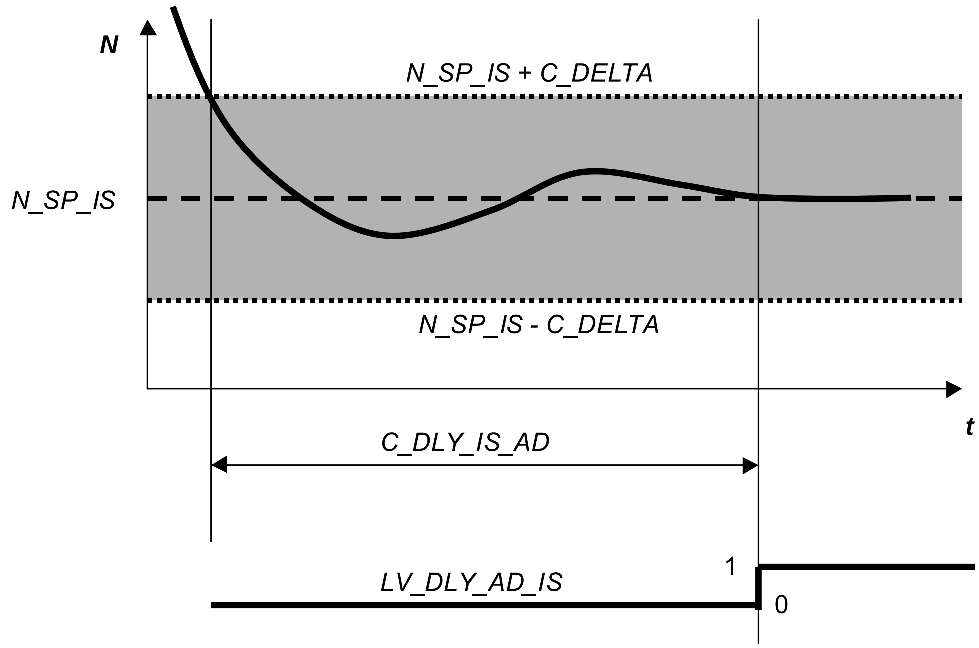


Figure 2: *Speed condition for adaptation activation*

- vehicle stopped (LV_VS_RUN = 0)
- coolant temperature in operating range:

IF { C_TCO_AD_MIN_IS ≤ TCO ≤ C_TCO_AD_MAX_IS }

THEN

LV_TCO_AD_CDN_IS = 1

ELSE

LV_TCO_AD_CDN_IS = 0

END

- LV_IS_AD_INH_OBD = 0 (see module "Application Incidence")
- LV_IS_AD_INH = 0 (see module "Application Incidence")

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_AD_MAX_IS	1	0...FEH	-48...143	0.75	°C
temperature top limit for ISC adaptation					
C_TCO_AD_MIN_IS	1	0...FEH	-48...143	0.75	°C
temperature bottom limit for ISC adaptation					
C_N_DELTA	1	0...1FE0H	0...8160	1	rpm
engine speed threshold for ISC adaptation					
C_DLY_AD_IS	1	0...FFH	0...255	1	100 ms
time delay for ISC adaptation					

8.7 Adaptation Algorithm

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_DIF_IS_AD	V/S/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Basic ISC adaptation					
TQ_DIF_IS_AD_INTER	V/S	80000000 ... 7FFFFFFF H	-1024...1024	4.76844 E-07	Nm
Basic ISC adaptation, intermediate value with high resolution (long signed word)					
TQ_DIF_IS_AD_CONV	V/S/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Hydraulic converter adaptation torque					
TQ_DIF_IS_AD_CONV_INTER	V/S	80000000 ... 7FFFFFFF H	-1024...1024	4.76844 E-07	Nm
Hydraulic converter adaptation torque, intermediate value with high resolution (long signed word)					
TQ_DIF_IS_AD_ACC_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
ACC torque adaptation					
TQ_DIF_IS_AD_ACC_1_INTER	V/S	80000000 ... 7FFFFFFF H	-1024...1024	4.76844 E-07	Nm
ACC torque adaptation, intermediate value with high resolution (long signed word)					
TQ_DIF_IS_AD_ACC_CONV	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
ACC and Hydraulic converter torque adaptation					
TQ_DIF_IS_AD_ACC_CONV_INTER	V/S	80000000 ... 7FFFFFFF H	-1024...1024	4.76844 E-07	Nm
ACC and Hydraulic converter torque adaptation, intermediate value with high resolution (long signed word)					
TQ_DIF_IS_AD_ACC	V/S/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Low-pass filtered ACC torque adaptation or ACC and Hydraulic converter torque adaptation					

Input data:

LV_DRI	LV_ACCOUT_RLY	TQ_DIF_I_IS	LV_AD_CDN_IS

Signal flow diagram:

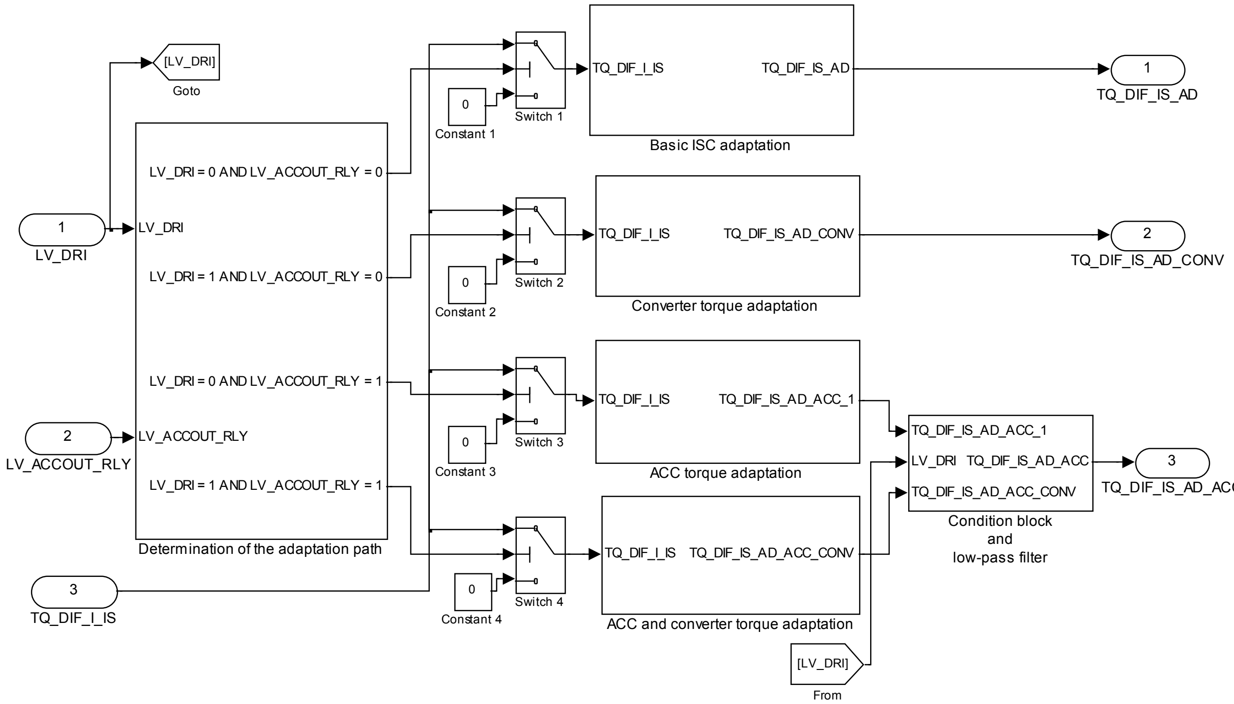


Figure 1: *Adaptation Algorithm*

The four adaptation blocks and the condition block will be described in the following function description.

Initialization:

At first initialization and after a checksum failure, all parameters are set to zero:

TQ_DIF_IS_AD	= 0,
TQ_DIF_IS_AD_INTER	= 0,
TQ _DIF_IS_AD_CONV	= 0,
TQ _DIF_IS_AD_CONV_INTER	= 0
TQ_DIF_IS_AD_ACC	= 0,
TQ_DIF_IS_AD_ACC_1_INTER	= 0,
TQ_DIF_IS_AD_ACC_CONV_INTER	= 0.

Application conditions:

Activation:

LV_AD_CDN_IS = 1

Deactivation:

LV_AD_CDN_IS = 0

Update rate:

100 ms

FUNCTION DESCRIPTION:

As soon as the adaptation conditions are fulfilled (LV_AD_CDN_IS = 1), one of the four adaptation parameters is calculated dependent on the actual situation:

- The adaptation of ACC and CONV (TQ_DIF_IS_AD_ACC_CONV) is calculated if the gear is engaged (LV_DRI = 1) and the air conditioning compressor is switched on (LV_ACCOUT_RLY = 1).
- If LV_ACCOUT_RLY = 1 and LV_DRI = 0, the adaptation of the air conditioning compressor is calculated.
- The Hydraulic converter torque adaptation TQ_DIF_IS_AD_CONV is calculated if LV_DRI = 1 and LV_ACCOUT_RLY =0.
- The basic adaptation of the idle speed controller (TQ_DIF_IS_AD) is calculated if the gear is disengaged (LV_DRI = 0) and the air conditioning compressor is switched off (LV_ACCOUT_RLY = 0).

Therefore, a fraction of the integral part of the idle speed controller (C_FAC_TQ_DIF_IS_AD_xx * TQ_DIF_I_IS) is added at each cycle to the concerned adaptation parameter, whose value is limited through a min/max limiter. The adaptation parameters are stored, even after deactivation of the function. These values are used as start values after re-entering into the same adaptation path.

The outputs of the different adaptation paths are always covered with values of the last adaptation, even if the adaptation conditions are not fulfilled or the engine is not in idle. But these values will only be adapted (changed) if the certain adaptation path is activated and they are certainly only considered in the single modules (“Torque Loss and Lead for ACC”, “Converter Torque”) in conformance with the actual situation (ACC on, drivetrain engaged). Only the basic torque losses (TQ_DIF_IS_AD) are always considered in the module “Torque Losses”.

TQ_DIF_IS_AD_CONV is considered in the converter torque calculation module, TQ_DIF_IS_AD_ACC and TQ_DIF_IS_AD_ACC_CONV in the calculation of the air conditioning compressor losses. TQ_DIF_IS_AD is added in the torque losses.

The three torque adaptation blocks "Basic ISC adaptation", "ACC torque adaptation" and "ACC and converter torque adaptation" have the following structure, figure 2.

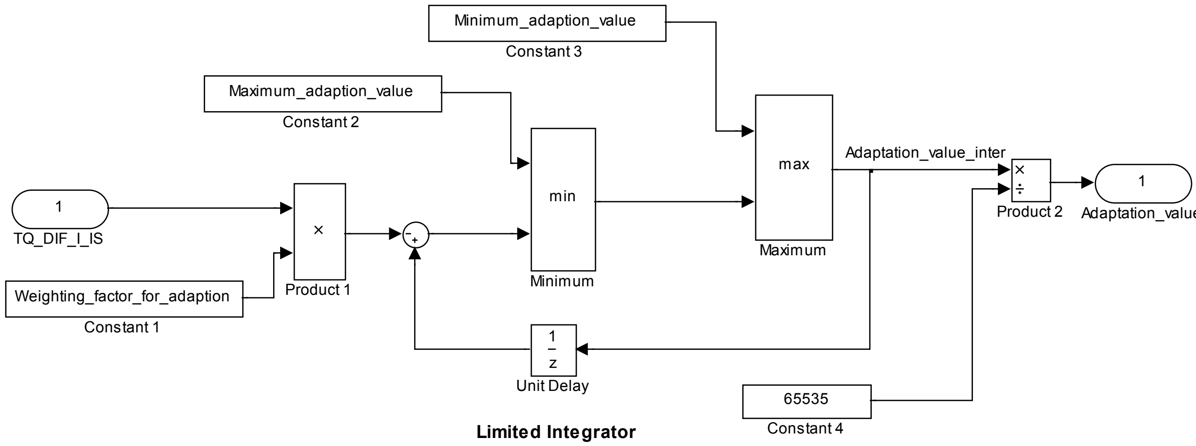


Figure 2: *Adaptation algorithm for the torque adaptation values*

The following constants are valid depending on the situation:

IF { LV_DRI = 0 } **AND** { LV_ACCOUT_RLY = 0 }

```
% calculate TQ_DIF_IS_AD
```

Weighting_factor_for_adaptation = C_FAC_TQ_DIF_IS_AD;
Maximum_adaptation_value = C_TQ_DIF_IS_AD_MAX;
Minimum_adaptation_value = C_TQ_DIF_IS_AD_MIN;
Adaptation_value_inter = TQ_DIF_IS_AD_INTER;
Adaptation_value = TQ_DIF_IS_AD;

ENDIF;

IF { LV_DRI = 0 } **AND** { LV_ACCOUT_RLY = 1 }

```
% calculate TQ_DIF_IS_AD_ACC_1
```

Weighting_factor_for_adaptation = C_FAC_TQ_DIF_IS_AD_ACC;
Maximum_adaptation_value = C_TQ_DIF_IS_AD_ACC_MAX;
Minimum_adaptation_value = C_TQ_DIF_IS_AD_ACC_MIN;
Adaptation_value_inter = TQ_DIF_IS_AD_ACC_1_INTER;
Adaptation_value = TQ_DIF_IS_AD_ACC_1;

ENDIF;

```
IF  { LV_DRI = 1 } AND { LV_ACCOUT_RLY = 1 }
  % calculate TQ_DIF_IS_AD_ACC_CONV
  Weighting_factor_for_adaptation = C_FAC_TQ_DIF_IS_AD_ACC_CONV;
  Maximum_adaptation_value        = C_TQ_DIF_IS_AD_ACC_CONV_MAX;
  Minimum_adaptation_value        = C_TQ_DIF_IS_AD_ACC_CONV_MIN;
  Adaptation_value_inter          = TQ_DIF_IS_AD_ACC_CONV_INTER;
  Adaptation_value                = TQ_DIF_IS_AD_ACC_CONV;
ENDIF;
```

The integration is done with high resolution values (long signed word, xxx_INTER). These long signed word values are stored. The following equations are valid:

TQ_DIF_IS_AD = TQ_DIF_IS_AD_INTER / 65535

TQ_DIF_IS_AD_CONV = TQ_DIF_IS_AD_CONV_INTER / 65535

TQ_DIF_IS_AD_ACC_1 = TQ_DIF_IS_AD_ACC_1_INTER / 65535

TQ_DIF_IS_AD_ACC_CONV = TQ_DIF_IS_AD_ACC_CONV_INTER / 65535

This describes the conversion from high resolution (signed long word) to “normal” resolution (signed word). Only the high word is taken from the signed long word value.

The following figure 3 shows the determination of the adaptation value for the converter torque calculation, which describes the hydraulic converter torque losses.

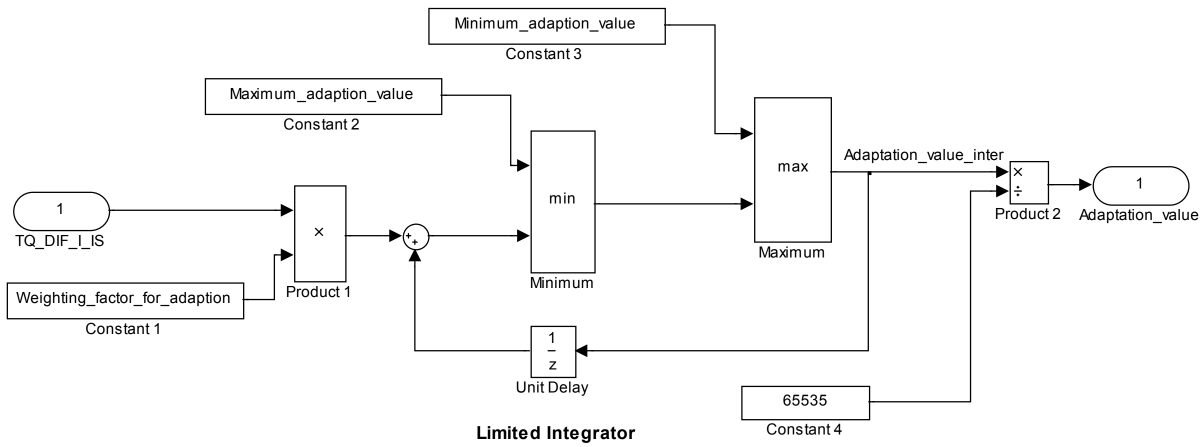


Figure 3: *Adaptation of the converter losses*

Mind that the summing point of the integrator for the converter torque determination consists of two plus signs, while those of the others consist of a minus and a plus (compare figure 2 with figure 3).

The following constants are valid for the converter torque adaptation value:

IF { LV_DRI = 1 } **AND** { LV_ACCOUT_RLY = 0 }

```
% calculate TQ_DIF_IS_AD_CONV
```

Weighting_factor_for_adaptation = C_FAC_TQ_DIF_IS_AD_CONV;
Maximum_adaptation_value = C_TQ_DIF_IS_AD_CONV_MAX;
Minimum_adaptation_value = C_TQ_DIF_IS_AD_CONV_MIN;
Adaptation_value_inter = TQ_DIF_IS_AD_CONV_INTER;
Adaptation_value = TQ_DIF_IS_AD_CONV;

ENDIF;

In order to be able to use one input for the ACC torque losses module, a switch is introduced to change between TQ_DIF_IS_AD_ACC_1 and TQ_DIF_IS_AD_ACC_CONV in dependance of the actual situation. To prevent abrupt changes in TQ_DIF_IS_AD_ACC (in case the switch changes from TQ_DIF_AD_IS_ACC_CONV to TQ_DIF_AD_IS_ACC_1), the output value of the switch is smoothed by a low pass filter, figure 4.

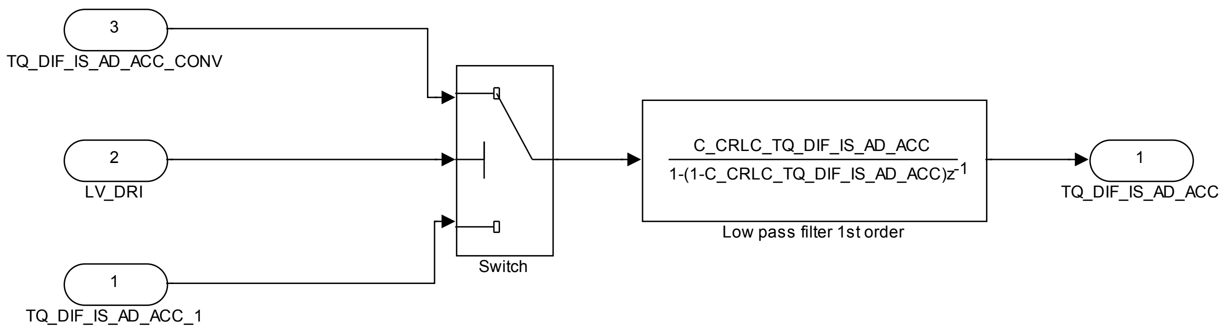


Figure 4: *Switch and low pass filter for smoothed transitions from "ACC torque adaptation" to "ACC and converter torque adaptation" and opposite*

The condition “LV_DRI=1” in the last IF-THEN command consideres the transient effects of the low-pass filter. If drivetrain is engaged, the engine is in idle and ACC is off, it’s advantageous to keep the switch at the upper position (TQ_DIF_IS_AD_ACC = TQ_DIF_IS_AD_ACC_CONV): if in this case the driver switches on ACC, TQ_DIF_IS_AD_ACC is already at the correct adaptation value in conformance with the actual situation. The additional switching condition “LV_ACCOUT_RLY=1” is not necessary, since TQ_DIF_IS_AD_ACC is added to TQ_LOSS_ACC in “Torque Loss and Lead for ACC” only in case of “ACC on”.

```
% Switch no. 37 in figure 1 and output low-pass filter
```

IF LV_DRI = 1

TQ_DIF_IS_AD_ACC = TQ_DIF_IS_AD_ACC_CONV (*low-pass filtered*);

ELSE

TQ_DIF_IS_AD_ACC = TQ_DIF_IS_AD_ACC_1 (*low-pass filtered*);

END;

During the first ACC adaptations, TQ_DIF_IS_AD_ACC may be erroneous and should therefore be limited to small values.

The speed of the adaptation process is defined through the calibration parameter C_FAC_TQ_DIF_IS_AD_xx and the function’s update rate, and should be very slow compared to variations of the integral part of the idle speed controller (C_FAC_TQ_DIF_IS_AD_xx << 1). In this case, the compensation of the added torque request is done automatically by the idle speed controller.

Further on, C_FAC_TQ _DIF_IS_AD_xx should be adjusted when the update rate changes.

Calibration data:

Name	Dim.	Hex. Limit	Phys. Limit	Resol.	Unit.
C_FAC_TQ_DIF_IS_AD	1	0...FFFFH	0...1	1.53e-5	-
Calibration parameter, LV_ACC=0, LV_DRI=0					
C_FAC_TQ_DIF_IS_AD_CONV	1	0...FFFFH	0...1	1.53e-5	-
Calibration parameter, LV_ACC=0, LV_DRI=1					
C_FAC_TQ_DIF_IS_AD_ACC	1	0...FFFFH	0...1	1.53e-5	-
Calibration parameter, LV_ACC=1					
C_FAC_TQ_DIF_IS_AD_ACC_CONV	1	0...FFFFH	0...1	1.53e-5	-
Calibration parameter, LV_ACC=1 AND LV_DRI = 1					
C_TQ_DIF_IS_AD_CONV_MAX	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Max. limitation for converter torque adaptation					
C_TQ_DIF_IS_AD_CONV_MIN	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Min. limitation of converter torque adaptation					
C_TQ_DIF_IS_AD_ACC_MAX	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Max. limitation for ACC torque adaptation					
C_TQ_DIF_IS_AD_ACC_MIN	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Min. limitation for ACC adaptation					
C_TQ_DIF_IS_AD_MAX	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Max. limitation of engine torque losses adaptation					
C_TQ_DIF_IS_AD_MIN	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Min. limitation of engine torque losses adaptation					
C_TQ_DIF_IS_AD_ACC_CONV_MAX	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Max. limitation of air conditioning compressor and converter torque adaptation					
C_ TQ_DIF_IS_AD_ACC_CONV_MIN	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Min. limitation of air conditioning compressor and converter torque adaptation					
C_CRLC_TQ_DIF_IS_AD_ACC	1	0 ... FFH	0 ... 0.9984	0.0039	-
Filter constant for „LOW_PASS_1“ (smoothing of the ACC-adaptation values)					

8.8 Idle Speed Torque Reserve

8.8.1 Idle Speed Torque Reserve for Idle Speed control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_N_DIF_OFS_PRED	V	0 .. FF H	0 ... 7.96875	0.03125	s
Prediction time for engine speed difference					
N_DIF_OFS_PRED	V	8000 ... 7FFFH	-32768 .. 32767	1	1/min
Predicted engine speed difference for torque reserve calculation					
N_DIF_PRED	V	8000 ... 7FFFH	-32768 .. 32767	1	1/min
Predicted idle speed control variable					
TQ_ADD_IS_BOL	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Basic torque reserve at idle					
TQ_DIF_ADD_IS_TOL	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Idle speed torque reserve limitiation for stabilizing the ISC at high torque reserves					

Input data:

N_DIF	N_GRD	N	LV_IS
TQI_ADD_MAX_TOL	TQI_REQ_TRA	TQI_REQ_TRA	N_SP_IS
TQ_ADD_ENG_STALL	TCO	TQ_ADD_N_DIF_INT_IS	

General Information:

The module „Idle Speed Torque Reserve“ consists of two parts:

1. Idle speed torque reserve due to low engine speed
2. Idle speed torque reserve limitiation for stabilizing the ISC at high torque reserves

1. Idle speed torque reserve due to low engine speed

Depending on the predicted engine speed difference N_DIF_PRED a torque reserve by increasing the load is calculated. If actual engine speed falls it is possible to increase load before the actual engine speed is below the engine speed setpoint for idle speed. At the same time a spark retard compensates the supposed torque increase. So a quick torque increase by spark advance can be realized to stabilize engine speed at its setpoint.

For negative gradients of TQ_ADD_IS_BOL_1 a PT1-filter for TQ_ADD_IS_BOL calculation is applied to avoid torque jumps caused by fast changes of ignition angle. For positive gradients of the input TQ_ADD_IS_BOL_1 the output value TQ_ADD_IS_BOL is identical to the input.

2. Idle speed torque reserve limitiation for stabilizing the ISC at high torque reserves

At high torque reserve requests (ex.: catalyst heating) the ignition angle can be retarded down to the minimum ignition angle IGA_MIN. This may cause problems with idle speed stability. In order to prevent speed from oscillating in case the actual ignition angle is in the near of IGA_MIN caused by high torque reserves, the maximum availible torque reserve must be limited. In order to reach this, a tunable calibration constant C_TQ_DIF_ADD_IS_TOL is introduced. C_TQ_DIF_ADD_IS_TOL has the same influence than TQ_ADD_IS_BOL: while the last defines a minimum torque reserve, C_TQ_DIF_ADD_IS_TOL takes influence on the maximum torque reserve in idle.

The availible torque reserve range is shown in figure 1.

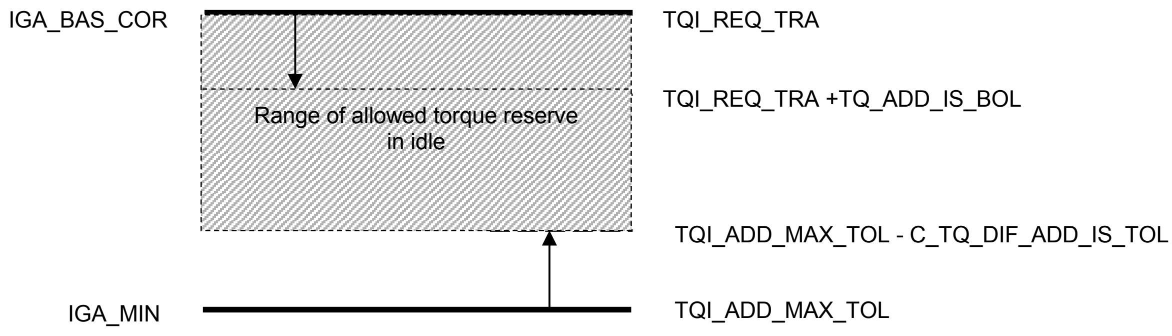


Figure 1: *Availible torque reserve range after limiting:*
TQI_ADD_MAX_TOL - C_TQ_DIF_ADD_IS_TOL defines the maximum torque reserve in idle

Since C_TQ_DIF_ADD_IS_TOL is a calibratable constant, this may cause a conflict situation as shown in figure 2.

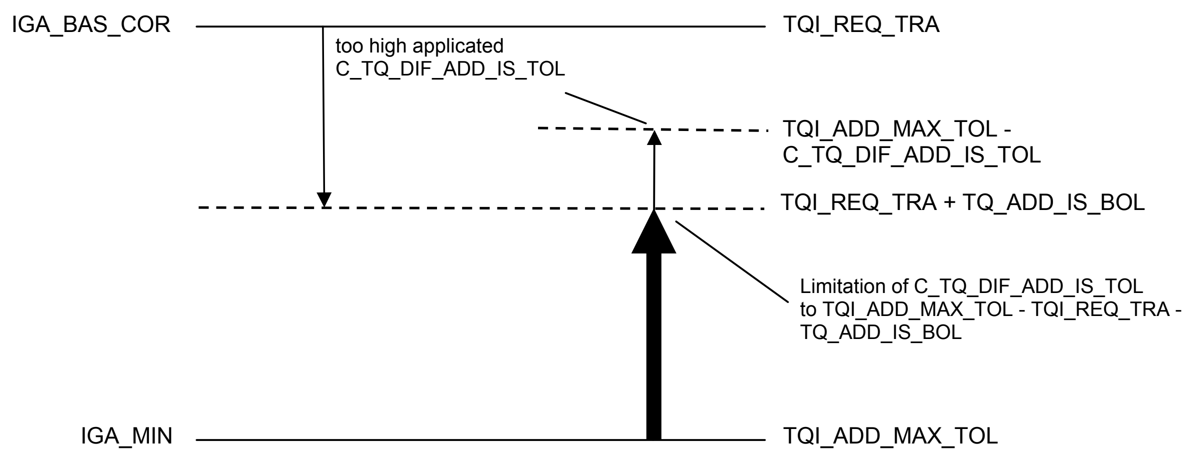


Figure 2: *Limiting C_TQ_DIF_ADD_IS_TOL in case of a conflict*

In this case TQ_ADD_IS_BOL has a higher priority than C_TQ_DIF_ADD_IS_TOL, since the possibility of moving the ignition angle in direction spark advance by building up a torque reserve is more essential than keeping it at a certain point from IGA_MIN, which „only“ causes irregular running. The condition for a non-critical situation, figure 1, is

TQI_REQ_TRA + TQ_ADD_IS_BOL <= TQI_ADD_MAX_TOL - C_TQ_DIF_ADD_IS_TOL (1)

Solving this relation leads to

C_TQ_DIF_ADD_IS_TOL <= TQI_ADD_MAX_TOL - TQI_REQ_TRA - TQ_ADD_IS_BOL (2)

If relation (2) is not fulfilled, C_TQ_DIF_ADD_IS_TOL is limited, figure 3.

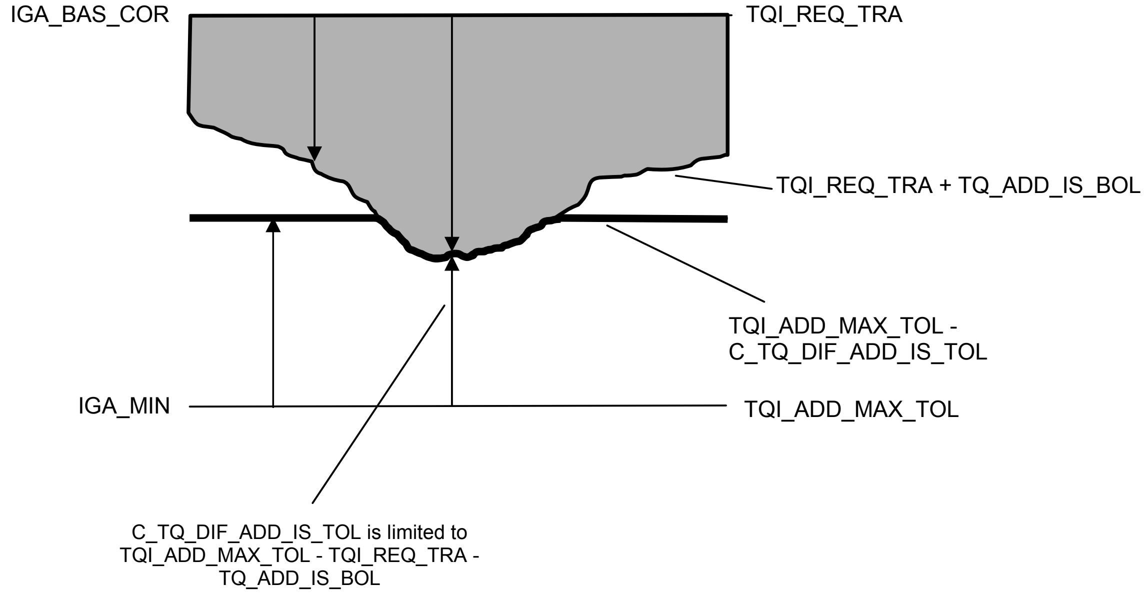


***Figure 3:** Limiting C_TQ_DIF_ADD_IS_TOL in case TQI_REQ_TRA + TQ_ADD_IS_BOL is greater than TQI_ADD_MAX_TOL - C_TQ_DIF_ADD_IS_TOL*

Signal flow diagram:

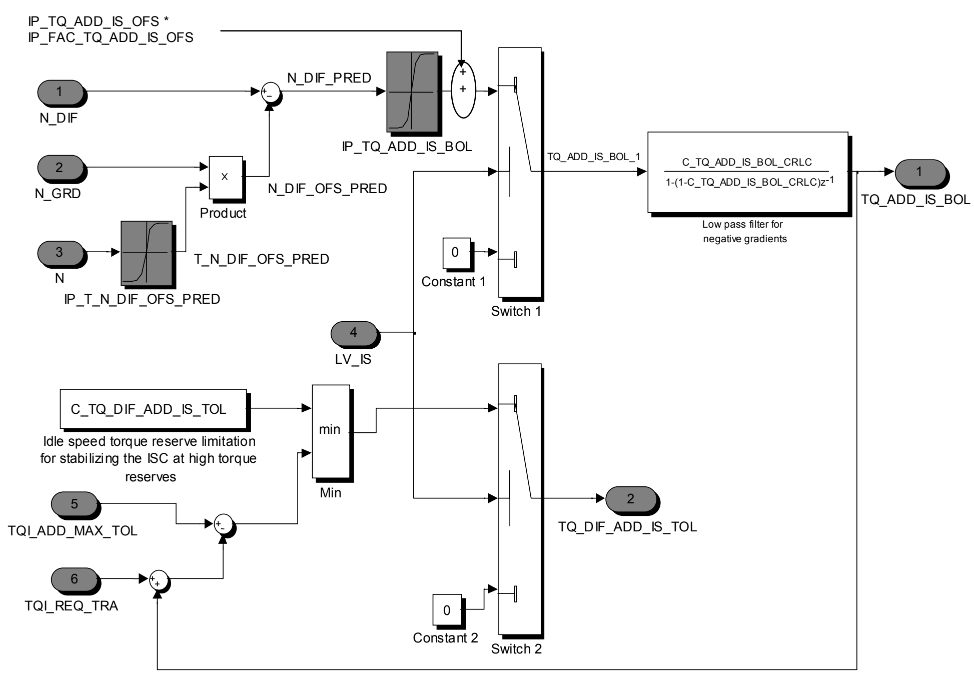


***Figure 4:** Structure torque reserve in idle*

FUNCTION DESCRIPTION:

General information:

Activation: LV_IS = 1

Deactivation: if LV_IS = 0

then T_N_DIF_OFS_PRED = 0 s

N_DIF_OFS_PRED = 0 rpm

N_DIF_PRED = N_DIF

TQ_DIF_ADD_IS_TOL = 0 Nm

Initialization: at reset

TQ_ADD_IS_BOL = 0 Nm

TQ_DIF_ADD_IS_TOL = 0 Nm

Update Rate: 10 ms

PT1-Filter negative gradient:

if $TQ\_ADD\_IS\_BOL\_1_{k} + TQ\_ADD\_ENG\_STALL + TQ\_ADD\_N\_DIF\_INT\_IS >= TQ\_ADD\_IS\_BOL_{k-1}$

then $TQ\_ADD\_IS\_BOL_{k} = TQ\_ADD\_IS\_BOL\_1_{k} + TQ\_ADD\_ENG\_STALL + TQ\_ADD\_N\_DIF\_INT\_IS$

else $TQ\_ADD\_IS\_BOL_{k} = TQ\_ADD\_IS\_BOL_{k-1} + C\_TQ\_ADD\_IS\_BOL\_CRLC * ((TQ\_ADD\_IS\_BOL\_1_{k} + TQ\_ADD\_ENG\_STALL + TQ\_ADD\_N\_DIF\_INT\_IS) - TQ\_ADD\_IS\_BOL_{k-1})$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_T_N_DIF_OFS_PRED__N	8x1	0 .. FF H	0 ... 7.96875	0.03125	s
LDP_N_T_N_DIF_OFS_PRED	8	0...1FE0H	0...8160	1	1/min
Time constant for calculation of predicted engine speed deviation					
IP_TQ_ADD_IS_BOL__N_DIF_PRED	8x1	0...7FFFH	0 .. 1023.97	0.03125	Nm
LDP_N_DIF_PRED_TQ_ADD_IS_BOL	8	0...FFFFH	-32768 .. 32767	1	1/min
Torque reserve at idle due to a deviation of engine speed to set point at idle					
IP_TQ_ADD_IS_OFS	6x6	0...7FFFH	0 .. 1023.97	0.03125	Nm
LDP_N_SP_IS_TQ_ADD_IS_OFS	6	0...1FE0H	0 .. 8160	1	rpm
LDP_TQI_REQ_TRA_TQ_ADD_IS_OFS	6	0...FFFFh	-1024...1023.97	0.03125	Nm
Torque reserve offset at idle depends on N_SP_IS and TQI_REQ_TRA					
IP_FAC_TQ_ADD_IS_OFS	6x1	0...FFH	0 .. 1	0.0039	-
LDP_TCO_ FAC_TQ_ADD_IS_OFS	6	00...FE H	-48...142.5	0.75	°C
Factor for Torque reserve offset at idle depends on TCO					
C_TQ_ADD_IS_BOL_CRLC	1x1	0...FFH	0...0.9984	0.0039	-
Filter constant for torque reserve					
C_TQ_DIF_ADD_IS_TOL	1x1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Idle speed torque reserve limitiation for stabilizing the ISC at high torque reserves					

8.8.2 Idle Speed Torque Reserve to prevent engine stall

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_ENG_STALL	V/O	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Additional Idle torque reserve to avoid engine stall in case of deviation between TPS_AV1 and TPS_AV_2					
LV_TQ_ADD_ENG_STALL_ACT	V/O	0…1H	0…1	1	-
Activation condition for additional Idle torque reserve to avoid engine stall					
TPS_AV_DIF	V	0…3FFFH	0… 119.5	0.0073	°TPS
Absolute difference between TPS_AV_1 and TPS_AV_2					

Input data:

LV_CT	MAF_MES	TPS_AV_1	TPS_AV_2
N	ERR_SYM_MAF_SCG_OC	LV_ST	LV_ERR_TPS
STATE_TPS_DIAG	LV_ES	MAP_MES	ERR_SYM_EL_MAP
ERR_SYM_LOAD_TPS_PLAUS			

General Information:

With ETC system, a deviation between TPS inputs (TPS_AV_1 & TPS_AV_2) can lead to an engine stall. For example, if TPS_AV_1 is abnormally higher than TPS_AV_2, calculated TPS position (average of TPS_AV_1 and TPS_AV_2) will increase. To keep throttle position set-point, system will close the throttle, and this is leading to an engine stall.

To avoid this, an additional torque reserve (to increase throttle position set point) is triggered in case of :

- deviation between TPS_AV_1 and TPS_AV_2 and
- low position of TPS_AV_1 or TPS_AV_2 and
- low measured MAF *(indicates that throttle is abnormally closed)* and
- Engine speed undershot

FUNCTION DESCRIPTION:

Application conditions :

Initialization: at reset

TQ_ADD_ENG_STALL = 0 Nm

LV_TQ_ADD_ENG_STALL_ACT = 0

TPS_AV_DIF = 0

Update Rate: 10 ms

Activation: **If** LV_ES = 0

and LV_ST = 0

and LV_CT = 1

and STATE_TPS_DIAG = 0

and ERR_SYM_MAF_SCG_OC = 0

and ERR_SYM_EL_MAP = 0
and ERR_SYM_LOAD_TPS_PLAUS = 0
and LV_ERR_TPS = 0
and MAF_MES < C_MAF_MES_ENG_STALL
and MAP_MES < C_MAP_MES_ENG_STALL
and N < C_N_ENG_STALL_MIN
and TPS_AV_DIF >= C_TPS_AV_DIF_ENG_STALL
and (TPS_AV_1 < C_TPS_AV_ENG_STALL
or TPS_AV_2 < C_TPS_AV_ENG_STALL)
then LV_TQ_ADD_ENG_STALL_ACT = 1

Deactivation: **if** LV_ES = 1
or LV_ST = 1
or LV_CT = 0
or STATE_TPS_DIAG > 0
or ERR_SYM_MAF_SCG_OC > 0
or ERR_SYM_EL_MAP > 0
or ERR_SYM_LOAD_TPS_PLAUS > 0
or LV_ERR_TPS = 1
or N > C_N_ENG_STALL_MAX
or TPS_AV_DIF < C_TPS_AV_DIF_ENG_STALL
then LV_TQ_ADD_ENG_STALL_ACT = 0

Formula section :

- *TPS_AV_DIF calculation :*

  TPS_AV_DIF = | TPS_AV_2 – TPS_AV_1 |

- *TQ_ADD_ENG_STALL calculation :*

  If LV_TQ_ADD_ENG_STALL_ACT = 1
  Then TQ_ADD_ENG_STALL = IP_TQ_ADD_ENG_STALL__TPS_AV_DIF
  Else TQ_ADD_ENG_STALL = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ADD_ENG_STALL __TPS_AV_DIF	4x1	0...7FFFH	0 .. 1023.97	0.03125	Nm
LDP_TPS_AV_DIF_TQ_ADD_ENG_STALL	4	0…3FFFH	0… 119.5	0.0073	°TPS
Additional torque reserve to avoid engine stall in case of deviation between TPS_AV_1 and TPS_AV_2					
C_MAF_MES_ENG_STALL	1	0…FFFFH	0…1389	0.0212	mg/STK
Measured Mass Air Flow limit to trigger anti stall torque reserve					
C_N_ENG_STALL_MIN	1	0...1FE0H	0...8160	1	rpm
Low Engine Speed limit to trigger anti stall torque reserve					
C_N_ENG_STALL_MAX	1	0...1FE0H	0...8160	1	rpm
High Engine Speed limit to stop anti stall torque reserve					
C_TPS_AV_DIF_ENG_STALL	1	0…3FFFH	0… 119.5	0.0073	°TPS
TPS_AV deviation to trigger anti stall torque reserve					
C_TPS_AV_ENG_STALL	1	0…3FFFH	0… 119.5	0.0073	°TPS
Low TPS_AV limit to trigger anti stall torque reserve					
C_MAP_MES_ENG_STALL	1	0…FFFFH	0...5.434E+3	0.0829175 2	hPa
Measured manifold pressure limit to trigger anti stall torque reserve					

8.8.3 Idle Speed Torque Reserve increase to eliminate engine speed oscillations

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_N_DIF_INT_IS	V/O	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Additional Idle torque reserve to eliminate engine speed oscillations					
LV_N_DIF_INT_IS	V	0...1H	0...1	1	-
Flag to indicate if check for idle speed oscillations is active					
N_DIF_INT_IS	V	0...FFFFH	0...65535	1	rpm
Integrated delta engine speed setpoint to actual engine speed					

Input data:

LV_IS	N_DIF_COR		

General Information:

In certain conditions, it may be possible that during idle an engine speed oscillation occurs that cannot be damped by the normal idle controller. In this module an oscillation is detected and an increased torque reserve is applied. This give the idle controller more authority (on the fast / spark path) which allows better damping of oscillations.

FUNCTION DESCRIPTION:

Application conditions :

Initialization: at reset and deactivation

TQ_ADD_N_DIF_INT_IS = 0 Nm

LV_N_DIF_INT_IS = 0

N_DIF_INT_IS = 0

Update Rate: 10 ms

Activation: LV_IS = 1

Deactivation: (LV_IS = 0 **and** N_DIF_INT_IS = 0) **or** LV_ES = 1

Formula section :

Start / stop integrator

if LV_IS = 0

then LV_N_DIF_INT_IS = 0

endif

if LV_IS = 1 **and** N_DIF_COR > 0 **and** T_AST > C_T_AST_N_DIF_INT

then LV_ N_DIF_INT_IS = 1

endif

Fill / empty integrator

if LV_ N_DIF_INT_IS = 1

then

$N\_DIF\_INT\_IS_n$ = max (0,

$N\_DIF\_INT\_IS_{n-1}$ + | N_DIF_COR | - C_N_DIF_INT_DEC)

NB Absolute value of N_DIF_COR is used

else

$N\_DIF\_INT\_IS_n$ = max (0, $N\_DIF\_INT\_IS_{n-1}$ - C_N_DIF_INT_DEC)

endif

Calculate desired torque reserve

TQ_ADD_N_DIF_INT_IS = IP_TQ_ADD_N_DIF_INT_IS (N_DIF_INT_IS)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ADD_N_DIF_INT_IS	4	0...7FFFH	0 .. 1023.97	0.03125	Nm
LDP_N_DIF_INT_IS_TQ_ADD	4	0…FFFFH	0… 65535	1	rpm
Additional torque reserve to avoid engine speed oscillations					
C_N_DIF_INT_DEC	1	0…FFFFH	0… 65535	1	rpm
Integrator decrement for engine speed oscillation detection					
C_T_AST_N_DIF_INT	1	0...FFFFH	0...6553.5	0.1	s
Time after start to enable engine speed oscillation detection					

8.9 Torque Reserve for canister purge at idle

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_CP	V/O	8000 .. 7FFFF H	-1024 ... 1023.97	0.03125	Nm
Torque reserve for canister purge at idle					

Input data:

CL_MMV	LV_IS	CP_STATE	

General Information:

In the canister purge function, a relative flow is used for controlling the canister purge solenoid (CPS):

REL_FLOW_CPS = MAF_CPS / MAF_CYL. (1)

This relative flow is a function of the canister load and so is constant:

REL_FLOW_CPS = f(CL) = const. (2)

These two equations mean that raising *MAF_CYL* by opening the throttle leads to a raising of *MAF_CPS* in order to keep the condition in eq.(2) fulfilled. This means that the purging rates of the canister purge solenoid raise, too and this leads to a better scouring of the ACF. In other words, if a torque reserve is produced in idle when the canister purge function is active, the purging rates raise.

The function has another very important effect. Canister purge solenoids have lower relative opening errors at high mass flows. As explained, opening the throttle by a torque reserve means opening the CPS and this means reducing relative errors. This leads to a better controlling of the CPS.

The torque reserve for canister purge in idle should be active, when the canister purge function is in a controlled mode. In this case the state of evaporative emission control is MAX_PURGE:

CP_STATE = MAX PURGE (3)

Signal flow diagram:

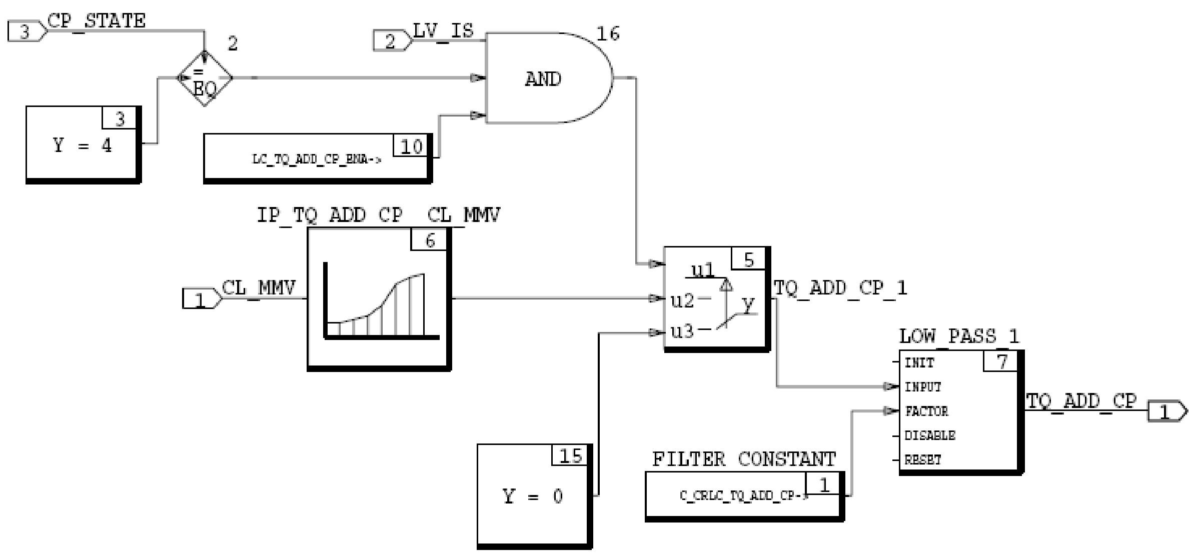


Figure 1: *Torque reserve for canister purge at idle*

FUNCTION DESCRIPTION:

General information:

Activation:		LV_IS = 1
	and	CP_STATE = MAX_PURGE
	and	LC_TQ_ADD_CP_ENA = 1
Deactivation:	**if**	LV_IS = 0
	or	CP_STATE ≠ MAX_PURGE
	or	LC_TQ_ADD_CP_ENA = 0
	When the function is deactivated, the value TQ_ADD_CP_1 should be equal zero: TQ_ADD_CP_1 = 0.	
Initialization:	at reset	TQ_ADD_CP = 0 Nm
Update Rate:	10 ms	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ADD_CP__CL_MMV	8x1	0...FFFFH	-1024...1023.97	0.03125	Nm
LDP_CL_MMV	8	0...FFFFH	0...8	0.000122	-
Map for determination of TQ_ADD_CP in dependence of CL_MMV					
C_CRLC_TQ_ADD_CP	1x1	0..FFH	0..0.996	0.0039	-
Filter constant					
LC_TQ_ADD_CP_ENA	1	0,1	0,1	1	-
Logical variable for activating/disactivating the torque reserve for canister purge in idle					

8.10 Idle speed setpoint for catalyst heating

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_SP_IS_CH	V/O	0...1FE0H	0...8160	1	[rpm]
Idle speed setpoint for catalyst heating					
LV_N_SP_IS_CH	V/O	0...1H	0...1	1	[-]
Flag for active idle speed increase for catalyst heating					

General information:

Outputs are just initialized. Function for idle speed setpoint for cataylst heating can be added.

Application conditions:

Initialisation: *At reset:*

N_SP_IS_CH = 0
LV_N_SP_IS_CH = 0

9 Auxiliary functions

Table of Contents

Index

V

9.1 Interface to Immobilizer Specification

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ANS_COM_IMOB_SMK	V	0H 1H 2H 3H 4H	NA Positive Negative Virgin Neutral	1	[-]
Immobilization status received from Smart-Key ICU					
ANS_ERR_COM_IMOB	V	0...FFH	0...255	1	[-]
SMATRA status {OK [00], Invalid message from ICU [01], No answer from ICO [02], Negative response from ICU [03]}					
CDN_DIAG_IMOB_MIL	V	0...7H	0...7	1	[-]
Diagnosis condition for each symptom					
CONF_IMOB_IGC	V	0...1H	0...1	1	[-]
Configuration Switch to select Ignition Order for Immobilizer Configuration during Development					
CONF_IMOB_SMARTRA	O/V/S	0H 1H 2H	Not Relevant SMARTRA-II SMARTRA-III	1	[-]
Learnt Immobilizer component type for Smartra Type Immobilizer					
CONF_IMOB_VER	O/V/S	0H 1H 2H 3H	Non-Imob SMARTRA Smart-Key Smart-Key 2	1	[-]
Learnt Immobilizer component type					
CTR_KEY_FAST_IMOB	V/S	0...FFH	0...255	1	[-]
Number of fast key ignition recognition					
CTR_LIH_IMOB	V/S	0...FFH	0...255	1	[-]
Number of starts by tester in limp home mode					
CTR_LOCKED_BY_TIMER	V/S	0...100H	0...256	1	[-]
Indicator for number of transitions to STATE_IMOB = LOCKED BY TIMER					
CTR_SDR_1_IMOB	V/S	0...FFFFH	0...65535	1	[-]
Indicator for number of engine lock due to immobilizer function					
CTR_SDR_2_IMOB	V/S	0...FFFFH	0...65535	1	[-]
Indicator for number of failure communication					
CTR_SDR_3_IMOB	V/S	0...FFFFH	0...65535	1	[-]
Indicator for number of Key On / Key Off					
CTR_SDR_4_IMOB	V/S	0...FFFFH	0...65535	1	[-]
Indicator for number of Key read but not learn					
ERR_SYM_IMOB_MIL	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error symptom IMOB_MIL diagnosis					
KEY_IMOB[8]	V/S	0...FFFFFFFFH	$0..2^{32}-1$	1	[-]
Array of key codes learnt (IDE).					
KEY_NR_LEARN_IMOB	V/S	0...FFH	0...255	1	[-]
Number of taught keys					
KEY_STATE_IMOB	V	0...FFH	0...3	0	[-]
Status for the current read key :{not yet checked [00], learn [01], virgin(not yet stored) [02], invalid(wrong or no key read) [03]}					
LV_ANS_ERR_IMOB	V	0...1H	0...1	1	[-]
Erroneous encoded value received					
LV_ANS_NOT_IMOB	V	0...1H	0...1	1	[-]
No correct answer received before end of delay allowed to the communication					
LV_ANS_POS_IMOB	V	0...1H	0...1	1	[-]
Positive encoded value received					

LV_CDN_DIAG_IMOB_MIL	O/V	0...01H	0...1	1	[-]
Diagnosis condition IMOB_MIL diagnosis					
LV_END_DIAG_IMOB_MIL	O/V	0...01H	0...1	1	[-]
End of diagnosis IMOB_MIL diagnosis					
LV_ERR_COM_P1MAX_IMOB	V	0...1H	0...1	1	[-]
This flag is set when the EMS doesn't receive the response entirely					
LV_ERR_COM_P2MAX_IMOB	V	0...1H	0...1	1	[-]
This flag is set when the EMS doesn't receive the response					
LV_ERR_COM_P4MAX_IMOB	V	0...1H	0...1	1	[-]
This flag is set when the EMS can't transmit the request					
LV_ERR_COM_RESP_IMOB	V	0...1H	0...1	1	[-]
This flag is set if the length or the checksum is not correct					
LV_ERR_IMOB_MIL	O/V	0...01H	0...1	1	[-]
Boolean for error currently present on IMOB_MIL command signal.					
LV_ERR_VS_IMOB	V	0...1H	0...1	1	[-]
Vehicle speed acquisition failure saved in none volatile memory					
LV_IMOB_IGC_OFF	O/V	0...1H	0...1	1	[-]
Prohibition of ignition by Immobilizer					
LV_IMOB_INJ_OFF	O/V	0...1H	0...1	1	[-]
Prohibition of injection by Immobilizer					
LV_INH_DIAG_IMOB_MIL	O/V	0...1H	0...1	1	[-]
Diagnosis inhibition IMOB_MIL diagnosis					
LV_KEY_FAST_IMOB	V	0...1H	0...1	1	[-]
Short previous key On					
LV_LIH_IMOB	V	0...1H	0...1	1	[-]
Limp home activation (true : limp home activated)					
LV_LOCK_IMOB	V	0...1H	0...1	1	[-]
Immobilization application status (true : injection and ignition are locked)					
LV_MAX_CTR_LOCKED_BY_TIMER	V	0...1H	0...1	1	[-]
Maximum number of incorrect password/PIN entering reached					
LV_PWL_IMOB	V	0...1H	0...1	1	[-]
Flag to maintain power latch for key learning at EOL (true : power latch is maintained)					
LV_SECU_1_IMOB	V	0...1H	0...1	1	[-]
Passenger safety case 1					
LV_SECU_2_IMOB	V	0...1H	0...1	1	[-]
Passenger safety case 2					
LV_SECU_A_IMOB	V	0...1H	0...1	1	[-]
System security case A					
LV_SECU_B_IMOB	V/S	0...1H	0...1	1	[-]
System security case B					
N_32_IMOB	V	0...FFH	0...8160	32	[rpm]
Engine speed saved in none volatile memory					
SEK_IMOB_SMARTRA3	S	0... FFFFFFFFFFFF	0..2^{32}-1	1	[-]
Secret Encryption Key for Smartra-III					
STATE_IMOB	V/S	0...FFH	0...4	0	[-]
Global EMS status for Immobilizer: {not yet checked [00], learnt [01], virgin [02], neutral [03], locked by timer [04]}					
STATE_IMOB_EMS	V/S	0...10H	0...10	0	[-]
Detailed EMS status: {not yet checked[00]: learnt_smartra[01]: learnt_smk[02]: virgin[03]: virgin_smartra[04]: virgin_smk[05]: neutral[06]: neutral_smartra[07]: neutral_smk[08]: locked by timer[09]					
STATE_IMOB_SMARTRA3	O/V	0H 1H 2H 3H 4H	Not Relevant VIRGIN LEARNT NEUTRAL LOCKED	1	[-]
State of Smartra-III					
VIN_IMOB	S	0...FFFFH	0...65535	1	[-]
Learnt VIN (Vehicle Identification Number) Code from ICU					
VS_IMOB	V	0...FFH	0...255	1	[km/h]

Vehicle speed saved in none volatile memory

Input data:

N_32	VS	LV_IGK	LV_ES
LV_ERR_VS	LV_IMOB_MIL_EXT_ADJ	LV_ACT_IMOB_MIL_EXT_ADJ	BF_LOCKED_BY_TIMER
TCO	LV_CDN_VB_OBD1	CONF_IMOB	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAX_DLY_ACT_IMOB	1	0 ... FFFFH	0 ... 655.35 s	0.01	s
Maximum delay allowed to activate immobilization function (**typical value 2 s**)					
C_N_SECU_1_IMOB	1	0...FFH	0... 8160 rpm	32	rpm
Engine speed threshold for Passenger safety case 1 (typical value : 1024 rpm)					
C_N_SECU_B_IMOB	1	0...FFH	0... 8160 rpm	32	rpm
Engine speed threshold in immobilization auto-protection strategy case B (typical value : 704 rpm)					
C_VS_SECU_1_IMOB	1	0...FFH	0... 255 km/h	1	km/h
Vehicle speed threshold for Passenger safety case 1 (typical value : 10 km/h)					
C_VS_SECU_B_IMOB	1	0...FFH	0... 255 km/h	1	km/h
Vehicle speed threshold in immobilization auto-protection strategy case B (typical value : 10 km/h)					
C_DLY_SECU_2_IMOB	1	0...FFH	0...255 s	1	S
Delay for Safety strategy for normal state (typical value 8 s)					
C_DLY_SECU_2_IMOB_LIH	1	0...FFH	0...255	1	S
Delay for Safety strategy for limp-home state (typical value 8 s)					
C_DLY_SECU_A_IMOB	1	0...FFH	0...255 s	1	S
Delay in immobilization auto-protection strategy (typical value 30 s)					
C_IGN_DLY_IMOB	1	0...FFH	0...25.5 s	0.1	S
Delay between injection stop and ignition stop (typical value 500 ms)					
C_DLY_LOCK_IMOB	1	0...FFFFH	0... 65535 s	1	S
Delay for EMS to stay in Locked By Timer State (typical value 3600 s = 1 hour)					
C_MAX_ACK_IMOB	1	1...FFH	1...255	-	-
Maximum number to send the Acknowledge message w/h error entry for Smartra (typical value is 3)					
C_MAX_ACK_IMOB_SMK	1	1...FFH	1...255	-	-
Maximum number to send the Acknowledge message w/h error entry for Smart-Key (typical value is 255)					
C_T_INI_COM_IMOB_SMARTRA	1	1...FFH	1...255	-	msec
Communication start time after IGK detection for SMARTRA					
C_T_INI_COM_IMOB_SMK_1	1	1...FFH	1...255	-	msec
Communication start time after IGK detection for Smart-Key 1					
C_T_INI_COM_IMOB_SMK_2	1	1...FFH	1...255	-	msec
Communication start time after IGK detection for Smart-Key 2					
C_TCO_MAX_CRK_THD_1	1	0...FEH	-48...142.5	0.75	[°C]
Deep cold condition threshold for maximum cranking time (typical value -15deg)					
C_TCO_MAX_CRK_THD_2	1	0...FEH	-48...142.5	0.75	[°C]
cold condition threshold for maximum cranking time (typical value 0deg)					
C_TCO_MAX_CRK_THD_3	1	0...FEH	-48...142.5	0.75	[°C]
warm condition threshold for maximum cranking time (typical value 30deg)					
C_CONF_IMOB_IGC	1	0 ... 1H	0 ... 1	1	-
Configuration Switch to select Ignition Order for Immobilizer Configuration during Development					

C_ABC_INC_ IMOB_MIL	1	0...FFH	0...255	1	-
Debounce counter increment IMOB_MIL diagnosis					
C_ABC_MAX_ IMOB_MIL	1	01...FFH	1...255	1	-
Debounce counter maximum value – IMOB_MIL diagnosis					

Configuration data:

Name	Type	Dim.	Hex. limits	Phys. limits	Resol.	Unit
NC_KEY_FAST_MIN_1	1U	1	0...FFH	0...25.5 s	100	ms
Minimum threshold (1) to detect twice Key On (fix set value **0.5 s**)						
NC_KEY_FAST_MAX_1	1U	1	0...FFH	0...25.5 s	100	ms
Maximum threshold (1) to detect twice Key On (t fix set value **1.5 s**)						
NC_KEY_FAST_MIN_2	1U	1	0...FFH	0...25.5 s	100	ms
Minimum threshold (2) to detect twice Key On (fix set value 0.2 s)						
NC_KEY_FAST_MAX_2	1U	1	0...FFH	0...25.5 s	100	ms
Maximum threshold (2) to detect twice Key On (fix set value **1.5 s**)						
NC_DLY_LIH_MIN_1_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **5 s**)						
NC_DLY_LIH_MIN_2_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **3 s**)						
NC_DLY_LIH_MIN_3_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **0.2 s**)						
NC_DLY_LIH_MIN_4_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **0.2 s**)						
NC_DLY_LIH_MAX_2_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **10 s**)						
NC_DLY_LIH_MAX_3_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **5 s**)						
NC_DLY_LIH_MAX_4_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Delay for limp home function and immobilizer lamp (fix set value **3 s**)						
NC_DLY_LIH_BLINK_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Immobilizer lamp blinking time (fix set value **5 s**)						
NC_CTR_LIH_IMOB	1U	1	0...FFH	0...255	-	-
Number maximum of starts by tester in limp home mode (fix set value **255**)						
NC_CTR_KEY_FAST_IMOB	1U	1	0...FFH	0...255	-	-
Number maximum of fast key ignition recognition (fix set value **32**)						
NC_CTR_KEY_FAST_IMOB_SMK_2	1U	1	0...FFH	0...255	-	-
Number maximum of fast key ignition recognition for SMK-2 (fix set value **64**)						
NC_T_MAX_TEACH_IMOB	1U	1	0...FFH	0...25.5 s	100	ms
Maximum time for Ign OFF during key teaching process (fix set value **10s**)						

General information:

This module is used to decribe the interface to Immobilizer related functions which are not mentioned in general SW Documentation due to confidentiality.

The applied Immobilizer Functionality to this SW release is based on Specification "Immo_Spec_Theta_Rev641".

9.2 Air condition compressor control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_RLY_ACCOUT	V	0...01H	0...1	1	-
Boolean of air condition compressor relay (OFF / ON).					
LV_RLY_ACCOUT_CTRL	V	0...01H	0...1	1	-
Boolean to activate ISAPWM correction related to air condition compressor relay					
LV_RLY_ACCOUT_ST	V	0...01H	0...1	1	-
Boolean of air condition compressor relay (OFF / ON) at cranking					
LV_UPHILL	V	0...01H	0...1	1	-
Boolean of vehicle uphill detection					
LV_UPHILL_DLY	V	0...01H	0...1	1	-
Boolean of vehicle uphill + additional delay to disable air condition compressor					
T_ACCIN_DLY	V	0...FFFFH	0...6553,5	0,1	sec
Time counter to disable A/C compressor after Start					
T_ACCIN_OFF_DLY	V	0...FFFFH	0...6553,5	0,1	sec
Time counter to deactivate vehicle take-off detection					
STATE_ACC_DLY	V	0...0H 1…1H 2…2H 4…4H 8…8H 9...9H	NO_ACCIN_DLY ACCIN_DLY_1 ACCIN_DLY_4 ACCIN_DLY_5 ACCIN_DLY_6 ACCIN_DLY_7	-	-
Time delay status for A/C compressor					
T_GS_ACT_DLY	V	0...FFFFH	0...655,35	0,01	sec
Time counter to disable A/C control after A/T gear shift					
LV_ACC_CFA	V/O	0...01H	0...1	1	-
selected ACC input for CFA control (0 = LV_ACIN, 1 = LV_RLY_ACCOUT) depending on LC_ACC_CFA_SEL					
T_MIN_ACCIN_OFF_DLY	V	0...FFFFH	0...6553.5	0.1	sec
Time counter to disable A/C control at vehicle fast					
T_VEH_STOP_BRK	V	0...FFFFH	0...2621.4	0.04	sec
Time counter to ensable A/C control at vehicle stop					
LV_AC_OFF_REQ_BRK	V	0...01H	0...1	1	-
A/C off request for break boost pressure build up					

Input data:

LV_ACIN	LV_ACCIN	LV_ST	LV_FL
LV_PL	LV_IS	N_32	TCO
VS	LV_CT	TPS_GRD	LV_ES
AMP_AD	LV_ERR_MAF	CONF_ACC	T_AST
TCO_ST	TIA_ST	LV_TPS_GRD_UP	CONF_ACP
ACP	TIA	VB	PV_AV
LV_ACCOUT_EXT_ADJ	LV_ACT_ACCOUT_EXT_ADJ	LV_GS_ACT	LV_ERR_MAP
TPS	LV_AT	GR_AT	GR_MT
LV_TPS_GRD_UP	LV_BLS	LV_ERR_BLS_BTS	MAP

FUNCTION DESCRIPTION:

General information:

The air condition compressor relay LV_RLY_ACCOUT is controlled by the ECU. It is enabled and disabled versus LV_ACIN and LV_ACCIN information. In order to keep a stable engine speed when A/C compressor is activated or dis-activated, idle speed valve correction are triggered in advance with LV_RLY_ACCOUT_CTRL.

In case of Start in high altitude, A/C compressor is disabled during a specific time delay in order to improve braking performance (lower pressure in intake manifold).

In order to eliminate slogging noise when A/C compressor is clutched after long soaking time, the A/C compressor can be activated during cranking.

In case of A/T vehicle, to avoid a shock with the A/C activation transition during A/T gear shift, A/C control needs to be delayed after the A/T gear shift done.

Remind : LV_ACIN : Air condition selected
LV_ACCIN : Air condition requested

Application conditions:

Initialisation: at transition engine run to engine stop all variables are reset

Recurrence: 40ms; except deactivation condition 7 , 8 (100ms)

Activation: (***LV_RLY_ACCOUT = ON***)

If CONF_ACC = 1 (Corresponding “Variant Coding” is set)
and engine operating states engine stopped (LV_ES) not active
and engine operating state start (LV_ST) not active
and air condition selected (LV_ACIN = ACIN)
and air condition requested (LV_ACCIN = ACCIN)
and LV_RLY_ACCOUT= OFF since more than C_ACCIN_DLY_5 seconds (time counted with T_ACCIN_DLY, STATE_ACC_DLY = ACCIN_DLY_5)
and TCO ≤ C_TCO_5_CFA
and LV_AC_OFF_REQ_BRK = 0
and LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

then LV_RLY_ACCOUT_CTRL is enabled immediately
and LV_RLY_ACCOUT is enabled after IP_ACCIN_DLY_2__N_32 seconds.
and Timer T_GS_ACT_DLY is reset

* *Deactivation* (**LV_RLY_ACCOUT = OFF**) :

If one of the deactivation conditions are true, LV_RLY_ACCOUT and LV_RLY_ACCOUT_CTRL are disabled whatever the activation conditions are.

1 - **If** Transition from LV_ST -> LV_IS
or LV_ST -> LV_PL
then STATE_ACC_DLY = ACCIN_DLY_4
If AMP_AD < C_AMP_AD_ACCIN_DLY_4 **or**
LV_ERR_MAF = 1 **or**
LV_ERR_MAP = 1
then T_ACCIN_DLY = IP_ACCIN_DLY_4__TCO_ST__TIA_ST
else T_ACCIN_DLY = ID_ACCIN_DLY_4__TCO_ST

else
If Transition LV_ACIN = 0 -> 1 **and**
T_AST > C_ACIN_DLY_4
then T_ACCIN_DLY reseted to 0
else Timer normally down to 0

2 - **If** T_ACCIN_DLY = 0

then LV_RLY_ACCOUT_CTRL is enabled immediately
LV_RLY_ACCOUT is enabled **after** IP_ACCIN_DLY_2__N_32

else LV_RLY_ACCOUT and LV_RLY_ACCOUT_CTRL are disabled immediately

3 - **If** acceleration enrichment due to full load (LV_FL) active
and VS **<** C_ACCIN_VS_MAX

then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after
LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

then STATE_ACC_DLY = ACCIN_DLY_1
LV_RLY_ACCOUT_CTRL and LV_RLY_ACCOUT are disabled immediately
during C_ACCIN_DLY_1 seconds (time counted with T_ACCIN_DLY).

Remark:
If LC_ENA_ACCIN_CT = 1
Then If LV_CT = 1 (no acceleration)
Then A/C deactivation timer is initialized

4 - **If** LV_ACCIN **=** 0 (-)
then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after
LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)
then LV_RLY_ACCOUT_CTRL is disabled immediately
LV_RLY_ACCOUT is disabled **after** IP_ACCIN_DLY_3__N_32 seconds.

5 - **If** TCO **>** C_TCO_5_CFA

then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after
LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

then LV_RLY_ACCOUT_CTRL is disabled immediately

LV_RLY_ACCOUT is disabled **after** IP_ACCIN_DLY_3__N_32 seconds

until TCO ≤ C_TCO_5_CFA - C_ACCOUT_TCO_HYS

6 - **If** TPS_GRD **>** C_TPS_GRD_ACCIN_OFF *(vehicle take-off detection)*

and LV_TPS_GRD_UP = 1

and LV_CT **=** 0

and VS **<** C_VS_ACCIN_OFF

and N_32 **<** C_N_ACCIN_OFF

and T_ACCIN_OFF_DLY = 0

then STATE_ACC_DLY = ACCIN_DLY_6

LV_RLY_ACCOUT_CTRL and LV_RLY_ACCOUT are disabled immediately **during** C_ACCIN_DLY_6 seconds (time counted with T_ACCIN_DLY)

and Time counter T_ACCIN_OFF_DLY is set to C_T_ACCIN_OFF_DLY

Remark:
If LC_ENA_ACCIN_CT = 1
Then If LV_CT = 1 (no acceleration)
Then A/C deactivation timer is initialized

7 - **If** CONF_ACP = 1

and ACP **>** C_ACP_MAX_RLY_ACCOUT_OFF

then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after

LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

then LV_RLY_ACCOUT_CTRL and LV_RLY_ACCOUT are disabled immediately
until ACP <= C_ACP_MAX_RLY_ACCOUT_OFF - C_ACP_MAX_HYS

8 - **If** CONF_ACP = 1

and ACP **<** C_ACP_MIN_RLY_ACCOUT_OFF

then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after

LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

then LV_RLY_ACCOUT_CTRL and LV_RLY_ACCOUT are disabled immediately
until ACP >= C_ACP_MIN_RLY_ACCOUT_OFF + C_ACP_MIN_HYS

9 - **If** PV_AV **>** C_PV_AV_ACCIN_OFF_UPHILL *(vehicle uphill detection)*

and VS **<** C_VS_ACCIN_OFF_UPHILL

and N_32 **<** C_N_ACCIN_OFF_UPHILL

then LV_UPHILL = 1

LV_UPHILL_DLY = 1

Endif

If one of above three conditions are not fulfilled

and LV_UPHILL = 1

and LV_UPHILL_DLY = 1

then LV_UPHILL = 0

LV_UPHILL_DLY = 0 after C_T_ACCIN_OFF_DLY_UPHILL.

If LV_UPHILL_DLY = 1

then if LV_GS_ACT = 0 and time delay of C_T_GS_ACT_DLY is elapsed after

LV_GS_ACT = 1 -> 0 (time counted with T_GS_ACT_DLY)

Then LV_RLY_ACCOUT_CTRL and LV_RLY_ACCOUT are disabled

Immediately, until LV_UPHILL_DLY = 0

Endif

10 - If VS > C_VS_MIN_ACCIN_OFF (vehicle acceleration while driving)

and { TPS > C_TPS_MIN_ACCIN_OFF

or (LV_AT = 1 **and** TPS_GRD > IP_TPS_GRD_MIN_ACCIN_OFF_AT

and LV_TPS_GRD_UP =1)

or (LV_AT = 0 **and** TPS_GRD > IP_TPS_GRD_MIN_ACCIN_OFF_MT

and LV_TPS_GRD_UP =1)}

and T_MIN_ACCIN_OFF_DLY = 0

Then STATE_ACC_DLY = ACCIN_DLY_7

LV_RLY_ACCOUT_CTRL **and** LV_RLY_ACCOUT are disabled immdediatly

during C_ACCIN_DLY_7 seconds (time counted with T_ACCIN_DLY)

and T_MIN_ACCIN_OFF_DLY = C_T_MIN_ACCIN_OFF_DLY

11 – To build up break boost pressure at low ambient pressure, brake pedal pressed, low vehicle speed and power steering turned, A/C off can be requested.

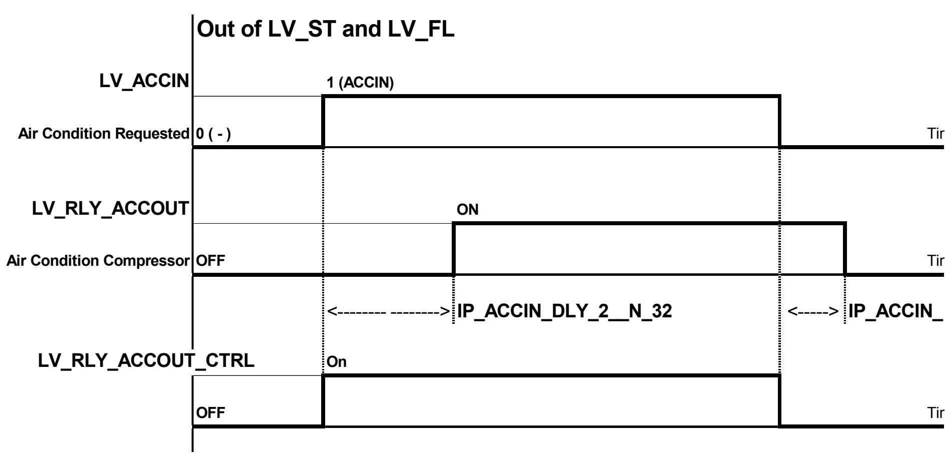
A/C to be reactivated after certain time passed with vehicle completely stopped or vehicle speed is above threshold after vehicle stopped with A/C on .

A/C can be deactivated again when vehicle moves slowly after stopped with A/C on.

If LV_AC_OFF_REQ_BRK = 0 and T_VEH_STOP_BRK = 0

then If		AMP_AD < C_AMP_AD_THD_ACCIN_OFF_BRK
	and	VS < C_VS_THD_ACCIN_OFF_BRK
	and	MAP > C_MAP_THD_ACCIN_OFF_BRK
	and	LV_BLS = 1
	and	LV_ERR_BLS_BTS = 0
then		
		LV_RLY_ACCOUT_CTRL **and** LV_RLY_ACCOUT are disabled immdediatly
		LV_AC_OFF_REQ_BRK = 1
endif		
else		
if		VS < C_VS_THD_ACCIN_ON_BRK
then		T_VEH_STOP_BRK = T_VEH_STOP_BRK + 40ms
else		T_VEH_STOP_BRK = 0
endif		
if		T_VEH_STOP_BRK => C_T_VEH_STOP_ACCIN_ON_BRK
	or	VS => C_VS_THD_ACCIN_ON_BRK
then		
		LV_AC_OFF_REQ_BRK = 0
endif		
endif		

Description:



Activation of air condition compressor relay at engine start

** Initialization :*

LV_RLY_ACCOUT_ST is set to 0 at engine stop (LV_ES)

** Activation* (**LV_RLY_ACCOUT_ST = ON)**

If Transition from ES to ST

and TIA > C_TAM_MIN_ACC_ST

and TCO – TIA < C_HYS_TAM_ACC_ST

and VB > C_VB_MIN_ACC_ST

Then LV_RLY_ACCOUT_ST = 1 during C_T_ACCOUT_ST_ACT

Else LV_RLY_ACCOUT_ST = 0

Activation of air condition compressor relay by external tester

If CONF_ACC = 1 and LV_RLY_ACCOUT_EXT_ADJ = 1, then RLY_ACCOUT is controlled by LV_ACT_ACCOUT_EXT_ADJ

Priority Rules:

LV_ACT_ACCOUT_EXT_ADJ has highest priority on LV_RLY_ACCOUT : If CONF_ACC = 1 and LV_ACCOUT_EXT_ADJ = 1 then LV_RLY_ACCOUT **=** LV_ACT_ACCOUT_EXT_ADJ

LV_RLY_ACCOUT_ST has priority on LV_RLY_ACCOUT : If LV_RLY_ACCOUT_ST = 1, then LV_RLY_ACCOUT **=** LV_RLY_ACCOUT_ST

Selection of air condition input for cooling fan control:

** Initialization :* LV_ACC_CFA = 0

** Activation* : at every engine state

If LC_ACC_CFA_SEL = 1

then **if** LV_RLY_ACCOUT = 1 *(A/C compressor active)*

then LV_ACC_CFA = 1

else LV_ACC_CFA = 0

else **if** LV_ACIN = 1 *(A/C selected in dashboard)*

then LV_ACC_CFA = 1

else LV_ACC_CFA = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ACCIN_VS_MAX	1	0...FFH	0...255	1	km/h
Maximum vehicle speed to deactivate the air condition compressor in full load (LV_FL).					
C_PV_AV_ACCIN_OFF_UPHILL	1	0...FFH	0...99,6	0,3906	%
PV_AV threshold to detect vehicle uphill driving					
C_TPS_GRD_ACCIN_OFF	1	0...FFH	0...2988	11,71	°TPS/sec
TPS_GRD threshold to detect vehicle take off					
C_VS_ACCIN_OFF	1	0...FFH	0...255	1	km/h
Vehicle speed threshold to detect vehicle take off					
C_N_ACCIN_OFF	1	0...FFH	0...8160	32	rpm
Engine speed threshold to detect vehicle take off					
C_VS_ACCIN_OFF_UPHILL	1	0...FFH	0...255	1	km/h
Vehicle speed threshold to detect vehicle uphill					
C_N_ACCIN_OFF_UPHILL	1	0...FFH	0...8160	32	rpm
Engine speed threshold to detect vehicle uphill					
C_T_ACCIN_OFF_DLY	1	0...FFFFH	0...6553,5	0,1	sec.
Minimum time between two vehicle take-off detection to deactivate A/C compressor					
C_T_ACCIN_OFF_DLY_UPHILL	1	0...FFFFH	0...6553,5	0,1	sec.
Delay time to deactivate A/C compressor after LV_UPHILL = 0					
C_ACCIN_DLY_1	1	01...FFH	0,1...25,5	0,1	sec
Time delay to deactivate air condition compressor in full load (LV_FL)					
IP_ACCIN_DLY_2	3	0001...FFFFH	0,01...655,35	0,01	sec
LDPM_N_32_IP_ACCIN_DLY	3	00...FFH	0...8160	32	rpm
Time delay to activate air condition compressor when LV_ACCIN = 1.					
IP_ACCIN_DLY_3	3	0001...FFFFH	0,01...655,35	0,01	sec
LDPM_N_32_IP_ACCIN_DLY	3	00...FFH	0...8160	32	rpm
Time delay to deactivate air condition compressor when LV_ACCIN = 0.					
C_AMP_AD_ACCIN_DLY_4	1	0000...FFFFH	0...5434	0,083	hPa
Minimum ambient pressure to detect high altitude to deactivate air condition compressor after Start.					
ID_ACCIN_DLY_4	6	01...FFFFH	0,1...6553,5	0,1	sec
LDPM_TCO_ST_IP_ACCIN_DLY_4	6	0...FEH	-48...142,5	0,75	°C
Time delay to deactivate air condition compressor after Start in case of low altitude.					
C_ACIN_DLY_4	1	01...FFFFH	0,1...6553,5	0,1	sec
Time delay after Start to check transition LV_ACIN = 0 -> 1.					
IP_ACCIN_DLY_4	1	01...FFFFH	0,1...6553,5	0,1	sec
LDPM_TCO_ST_IP_ACCIN_DLY_4	6	0...FEH	-48...142,5	0,75	°C
LDP_TIA_ST_IP_ACCIN_DLY_4	4	0...FEH	-48...142,5	0,75	°C
Time delay to deactivate air condition compressor after Start in case of high altitude.					
C_ACCIN_DLY_5	1	01...FFH	0,1...25,5	0,1	sec
Time delay between two air condition activations.					
C_ACCIN_DLY_6	1	01...FFH	0,1...25,5	0,1	sec
Time delay between to deactivate air condition compressor in case of vehicle take off.					
C_TCO_5_CFA	1	0...FEH	-48...142.5	0,75	°C
Coolant temperature threshold to switch OFF A/C compressor					
C_ACCOUT_TCO_HYS	1	0...FFH	0...191,25	0,75	°C
Coolant temperature hysteresis to start again A/C compressor after TCO > TCO_5_CFA					
C_ACP_MAX_RLY_ACCOUT_OFF	1	0...FFH	0...510	2	PSI
Maximum ACP for A/C compressor off					
C_ACP_MIN_RLY_ACCOUT_OFF	1	0...FFH	0...510	2	PSI
Minimum ACP for A/C compressor off					
C_ACP_MAX_HYS	1	0...FFH	0...510	2	PSI
Hysteresis for maximum ACP for A/C compressor off					
C_ACP_MIN_HYS	1	0...FFH	0...510	2	PSI
Hysteresis for minimum ACP for A/C compressor off					
C_TAM_MIN_ACC_ST	1	0...FEH	-48...142.5	0,75	°C
Minimum intake air temperature to activate A/C compressor at cranking					
C_HYS_TAM_ACC_ST	1	0...FFH	0...191,25	0,75	°C

Temperature hysteresis between TCO and TIA to activate A/C compressor at cranking					
C_VB_MIN_ACC_ST	1	0...FFH	0...25.8984	26/256	Volt
Minimum VB to activate A/C compressor at cranking					
C_T_ACCOUT_ST_ACT	1	0...FFH	0...2.55	0.01	sec
Time counter for A/C compressor ON at cranking					
C_T_GS_ACT_DLY	1	0...FFFFH	0...655,35	0,01	sec
Time delay to disable A/C control after A/T gear shift transition (LV_GS_ACT = 1->0)					
LC_ENA_ACCIN_CT	1	0...01H	0...1	1	-
Switch to enable ACCIN by LV_CT = 1					
LC_ACC_CFA_SEL	1	0...1H	0...1	1	-
cal. switch to select which A/C input to be used for cooling fan control (1 = LV_RLY_ACCOUT, 0 = LV_ACIN)					
C_VS_MIN_ACCIN_OFF	1	0...FFH	0...255	1	km/h
Minimum vehicle speed to deactivate the air condition compressor for fast acceleration					
C_TPS_MIN_ACCIN_OFF	1	0...FFH	0...119.5		°TPS
IP_TPS_GRD_MIN_ACCIN_OFF_MT	1*8	0...FFH	0...2988	11.71	°TPS/sec
LDPM_GR_MT_2	8	0...FFH	0...255	1	
Minimum TPS gradient threshold to deactivate the air condition compressor for fast acceleration (MT)					
IP_TPS_GRD_MIN_ACCIN_OFF_AT	1*8	0...FFH	0...2988	11.71	°TPS/sec
LDP_GR_AT	8	0...FFH	0...255	1	
Minimum TPS gradient threshold to deactivate the air condition compressor for fast acceleration (AT)					
C_ACCIN_DLY_7	1	01...FFH	0.1...25.5	0.1	sec
Time delay to deactivate air condition compressor in case of fast acceleration					
C_T_MIN_ACCIN_OFF_DLY	1	0...FFFFH	0...6553.5	0.1	sec
Minimum time between fast accelerations to deactivate A/C compressor					
C_VS_THD_ACCIN_OFF_BRK	1	0...FFH	0...255	1	km/h
Vehicle speed threshold to deactivate air condition for break booster pressure build up					
C_VS_THD_ACCIN_ON_BRK	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for reactivating air condition					
C_AMP_AD_THD_ACCIN_OFF_BRK	1	0000...FFFFH	0...5434	0,083	hPa
Ambient pressure threshold to deactivate air condition for break booster pressure build up					
C_T_VEH_STOP_ACCIN_ON_BRK	1	0...FFFFH	0...2621.4	0.04	sec
Time counter for air condition reactivation					
C_MAP_THD_ACCIN_OFF_BRK	1	0...FFFFH	0...5434	0.0829175	hPa
MAP threshold for deactivating air condition					

9.3 EGRC dummy outputs

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
AV_FLOW_EGRV	O	0..FFFFH	0..2048	0.03125	kg/h
Current EGR flow at EGR valve					
IGA_BAS_EGR_COR	O	80..7FH	-48..47.625	0.375	°CRK
Added basic spark advance with EGR					
IGA_BAS_S_EGR_COR	O	80..7FH	-48..47.625	0.375	°CRK
Added basic spark advance with EGR in stratified					
IGA_REF_EGR_COR	O	80..7FH	-48..47.625	0.375	°CRK
Added reference spark advance with EGR					
LV_REQ_DIAG_EGR	O	0..1H	0..1	1	-
EGR system diagnosis request					
LV_REQ_AD_EGR	O	0..1H	0..1	1	-
EGR valve adaptation request					
LV_REQ_ON_EGR	O	0...1H	0...1	-	-
Request for EGR flow activation					
LV_END_DIAG_EGR	O	0..1H	0..1	1	-
EGR system diagnosis ended					
LV_END_AD_EGR	O	0..1H	0..1	1	-
EGR adaptation ended					
LV_CMD_EGRV_EGR_DIAG	O	0..1H	0..1	1	-
EGR valve activated by EGR diagnosis/adaptation					
LV_CDN_ON_EGR	O	0...1H	0...1	-	-
EGR flow activation condition					
LV_ERR_EGR	O	0..1H	0..1	1	-
Present failure of EGR system					
LV_ERR_EGR_2	O	0..1H	0..1	1	-
Stuck valve or potentiometer failure					
LV_ERR_EL_EGR	O	0...1H	0...1	1	-
Boolean for error currently present on electrical part of exhaust gas recirculation (after debounce)					
LV_ERR_POTI_EGR	O	0...1H	0...1	1	-
Boolean for error on potentiometer of EGR valve due to supply voltage failure or potentiometer failure					
TEGR_ESTIM_DOWN	O	0..535H	0..1000	0.75	°C
EGR temperature for EGR gas flowing into the manifold					
SP_FLOW_EGRV	O	0..FFFFH	0..2048	0.03125	kg/h
EGR flow setpoint at the valve					
V_EGR	O	0...3FFH	0...4.995117	5/1023	V
EGR valve opening raw acquisition					

Input data:

General information:

For systems without external Exhaust Gas Recirculation (NC_EGR_CONF = 0) the aggregate outputs of EGRC have to be initialized to neutral values. If no EGR is available, the present module and the EGRC configuration module will be the only modules of the EGRC aggregate present in the ECU software. If EGR is available, the present module will not be present in the ECU software.

Recurrence : once after ECU reset.

Formula section:

AV_FLOW_EGRV = 0
IGA_BAS_EGR_COR = 0
IGA_BAS_S_EGR_COR = 0
LV_REQ_DIAG_EGR = 0
LV_REQ_AD_EGR = 0
LV_REQ_ON_EGR = 0
EGRPWM_CMD = 0
LV_END_DIAG_EGR = 0
LV_END_AD_EGR = 0
LV_CMD_EGRV_EGR_DIAG = 0
IGA_REF_EGR_COR = 0
LV_CDN_ON_EGR = 0
LV_ERR_EGR = 0
LV_ERR_EGR_2 = 0
LV_ERR_EL_EGR = 0
LV_ERR_POTI_EGR = 0
TEGR_ESTIM_DOWN = 0
SP_FLOW_EGRV = 0
V_EGR = 0

9.4 CHRG dummy outputs

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PSN_RCL_MES	O	0h..FFFFh	0..99.998474	1.526e-3	%
Measured recirculation actuator position					
LV_ERR_PSN_RCL	O	0h..1h	0..1	1	-
Recirculation actuator position sensor error					
LV_PSN_RCL_SP_EXT_ADJ	O	0h ... 1h	0 ... 1	1	-
Logical variable for recirculation actuator test, set to 1 if request from service tool is accepted					
LV_PSN_RCL_SP_AD	O	0h..1h	0..1	1	-
setpoint from adaptation or check functions has to be used -> adaptation active					
LV_RCL_AD_BOL_VLD	O	0h..1h	0..1	1	-
stored adaptation data for lower stop are successful adapted values					
LV_ERR_RCL_LIH_CHK	O	0h...1h	0...1	1	-
error in check of limp home spring (disable active RFP in next cycle)					
PSN_RCL_SP_EXT_ADJ	O	0h..FFFFh	0..99.998474	1.526e-3	%
Recirculation actuator setpoint requested by service tool					
PRS_VAC_PUMP	O	0h..FFFFh	0..5434	0.083	hPa
Pressure in the reservoir of the vacuum pump					
LV_ERR_PRS_WG_ACR	O	0h..1h	0..1	1	-
Wastegate actuator pressure sensor error					
LV_ERR_N_TCHA	O	0h..1h	0..1	1	-
Turbo charger speed sensor error					
MAF_THR_TCHA	V/O	0 ... FFFFH	0...2047.96875	0,03125	kg/h
Air mass flow through to the throttle for CHRG					
FLOW_ENG_TCHA	V/O	0 ... FFFFH	0...2047.96875	0,03125	kg/h
Exhaust gas flow through one turbo charger					
MAF_KGH_SP_TCHA	V/O	0 ... FFFFH	0...2047.96875	0,03125	Kg/h
Mass air flow setpoint for CHRG					
MAF_KGH_TCHA	V/O	0 ... FFFFH	0...2047.96875	0.03125	kg/h
Mass air flow for CHRG (inlet branch average)					
PRS_CHRG_DOWN_MAX	O	0h..FFFFh	0..5434	0.083	hPa
Pressure down charger aggregate (aggregate main output) for MAF_MAX_COR calcualtion					
MAF_KGH_AIC_TCHA [NC_NR_TCHA]	V/O	0 ... FFFFH	0...2047.96875	0.03125	kg/h
Air cleaner mass air flow for CHRG (inlet branch specific)					
LV_TCHA_CONF	V/O	0h..1h	0..1	1	-
Indicator whether system is equipped with a turbocharger					
MAF_KGH_AIC_MV_TCHA	V/O	0 ... FFFFH	0...2047.96875	0.03125	kg/h
Air cleaner mass air flow for CHRG (inlet branch average)					
PRS_AIC_DOWN_TCHA [NC_NR_TCHA]	V/O	0h..FFFFh	0..5434	0.083	hPa
Pressure downstream air cleaner for CHRG					
PRS_CHRG_DOWN	O/V	0...FFFFH	0...5.434E+3	0.082918	hPa
Pressure down charger aggregate (aggregate main output) (equal to PUT)					

PUT_SP_MAN	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure upstream throttle setpoint - manual value					
PWM_WG_MAN	O/V	0... FFFFH	0... 99.9984741	1.5259e-3	[%]
PWM at EPC for wastegate actuation - manual value					
ERR_SYM_PUT	O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom PUT sensor					

Input data:

NC_CBK_IN_NR	MAF_THR	PRS_AIC_DOWN	FLOW_ENG
NC_NR_TCHA	AMP	MAF_KGH_SP	LC_TCHA_CONF
MAF_KGH	PV_AV	N_32	PUT
ERR_SYM_EL_PUT			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PWM_WG_ACR_EPC_MAN	12x12	0...FFFFH	0...99.9984741	0.0015259	%
LDP_N_32_IP_PWM_WG_ACR_EPC_MAN	12	0...FFH	0...8.16E+3	32	rpm
LDP_PV_AV_IP_PWM_WG_ACR_EPC_MAN	12	0...FFH	0...99.609375	0.390625	%
Constant wastegate PWM signal provided by the application system					
IP_PUT_SP_AS	12x12	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDP_N_32_IP_PUT_SP_AS	12	0...FFH	0...8.16E+3	32	rpm
LDP_PV_AV_IP_PUT_SP_AS	12	0...FFH	0...99.609375	0.390625	%
Manual pressure up throttle setpoint depending on N_32 and PV_AV					

General information:

During development optional modules can be not implemented yet. Other modules can need the outputs of these not implemented functions as inputs. To satisfy their interfaces the output variables of of these not implemented functions are initialized here.
Specific for VW EMS2 Entwicklungsplattform: suitable only for systems with mono-turbo, one exhaust bank, low MAF range !

PRS_CHRG_DOWN_MAX is a dummy to satisfy INSY inputs.

Application conditions:

Initialisation: PRS_CHRG_DOWN_MAX = 5434 hPa
PRS_AIC_DOWN_TCHA [1] = AMP
LV_TCHA_CONF = LC_TCHA_CONF

standard initialisation of all other outputs with 0

Recurrence: see below

Activation: LC_TCHA_CONF = 1

Dectivation: LC_TCHA_CONF = 0

Formula section:

All output variables not calculated in the formula section keep their initialization values.

Calculations to be performed before the CHRG package

Recurrence: segment synchronous

MAF_KGH_AIC_TCHA [1] = MAF_KGH
MAF_KGH_AIC_MV_TCHA = MAF_KGH_AIC_TCHA [1]

MAF_THR_TCHA = MAF_ THR
MAF_KGH_TCHA = MAF_KGH

PRS_AIC_DOWN_TCHA [1] = PRS_AIC_DOWN

Recurrence: 10 ms

FLOW_ENG_TCHA = FLOW_ ENG
MAF_KGH_SP_TCHA = MAF_KGH_SP

PWM_WG_MAN = IP_PWM_WG_ACR_EPC_MAN (N_32, PV_AV)

PUT_SP_MAN = IP_PUT_SP_AS (N_32, PV_AV)

ERR_SYM_PUT = ERR_SYM_EL_PUT

Recurrence: multi-segment synchronous

PRS_CHRG_DOWN = PUT

9.5 Camshaft setpoints for catalyst heating

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_CAM_SP_IVVT_CH	O	0...FFH	0...0.99609	3.9063e-3	[-]
Interpolation factor for Camshaft setpoints for catalyst heating					
CAM_SP_IVVT_CH_IN	O	0...FFH	0...95.625	0.375	[°CRK]
Inlet camshaft setpoint during catalyst heating					
CAM_SP_IVVT_CH_EX	O	0...FFH	0...95.625	0.375	[°CRK]
Outlet camshaft setpoint during catalyst heating					

General information:

Outputs are just initialized.

Application conditions:

Initialisation: *At reset:*

FAC_CAM_SP_IVVT_CH = 0
CAM_SP_IVVT_CH_IN = 0
CAM_SP_IVVT_CH_EX = 0

9.6 Fuel pressure for catalyst heating

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FUP_RNG_H_SP_CH	O/V	0...FFFFH	0...255.996094	0.00390625	MPa
Fuel pressure setpoint for catalyst heating					

Input data:

N_32	MAF_STK_FG_PRED	STATE_CH	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FUP_RNG_H_SP_CH_PAS	1	0...FFFFH	0...255.996094	0.00390625	MPa
Passive fuel pressure setpoint for catalyst heating					
IP_FUP_RNG_H_SP_CH	6x6	0...FFFFH	0...255.996094	0.00390625	MPa
LDPM_N_32_3_EXTC	6	0...FFH	0...8.16E+3	32	rpm
LDPM_MAF_STK_FG_PRED_1_EXTC	6	0...FFFFH	0...2.778E+3	0.04238956	mg/stk
Fuel pressure setpoint during catalyst heating					

9.6.1 General information

FUP_RNG_H_SP_CH is input for aggregate FUSL (Fuel supply) and is used for STATE_CH = CH_AST=1.

Application Condition

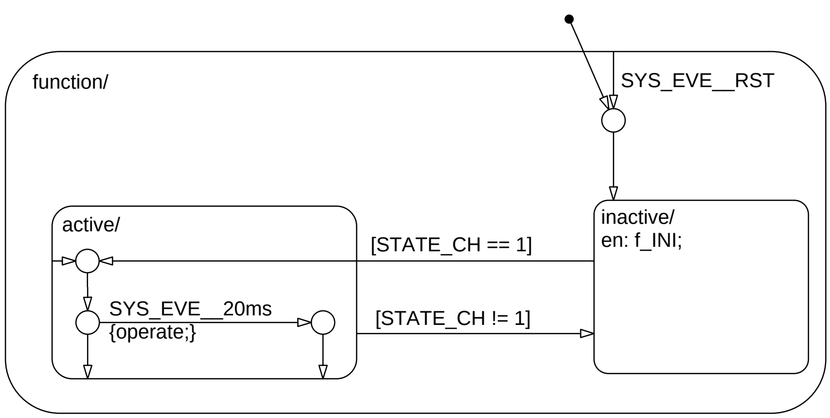


Figure 1 EXTC_REQGNfup0/ APP_CDN/ Chart1

Function Description

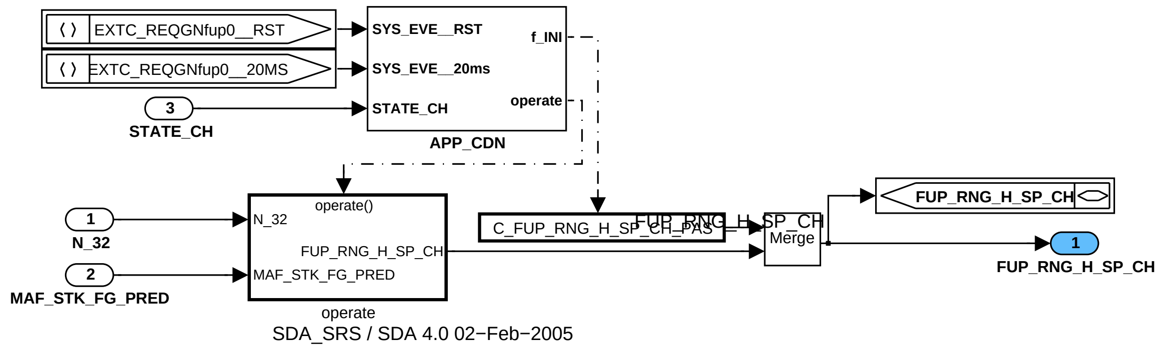


Figure 2 EXTC_REQGNfup0

9.6.1.1 Operate Subsystem

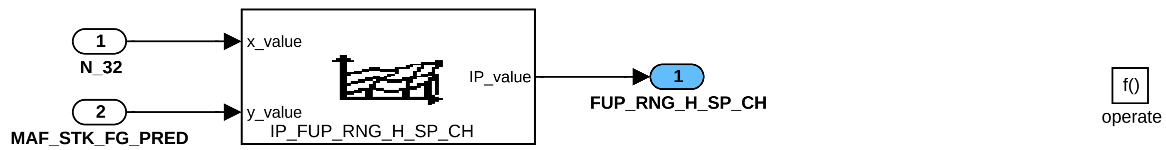


Figure 3 EXTC_REQGNfup0/ operate

9.7 Exhaust flap

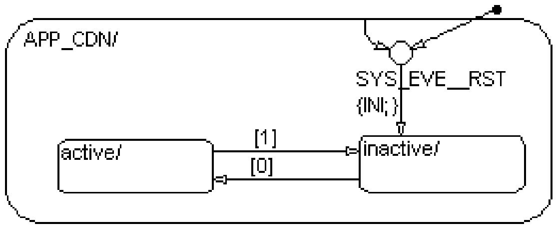
Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_EF	O	0...1H	0...1	1	-
Logical variable for the exhaust flap					

9.7.1 General information

The output data for the exhaust flap activation is just initialized.

Application Condition



Function Description

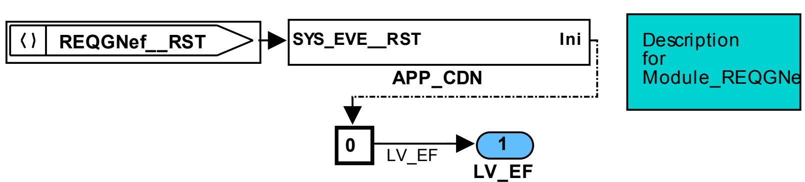


Figure 4 REQGNef

9.8 Adaptation of lower and higher of VIM (variable intake manifold)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_VIM_AD_SHO	V/O/S	0...3FFH	0...4.99511	4.8828e-3	[V]
Saved feedback value of V_VIM adaptation in VIM position "short"					
V_VIM_SHO	V/O	0...3FFH	0...4.99511	4.8828e-3	[V]
Measured feedback value of V_VIM in VIM position "short"					
V_VIM_AD_LONG	V/O/S	0...3FFH	0...4.99511	4.8828e-3	[V]
Saved feedback value of V_VIM adaptation in VIM position "long"					
V_VIM_LONG	V/O	0...3FFH	0...4.99511	4.8828e-3	[V]
Measured feedback value of V_VIM in VIM position "long"					
LV_VIM_AD_VLD	V/O/S	0...1H	0...1	1	[-]
Logical variable for "Adaptation values valid and adaptation successful"					
STATE_VIM_AD	V/O	0H 1H 2H 3H 4H 5H	VIM_AD_PUC_TIMER VIM_AD_SHORT_STOP VIM_AD_LONG_STOP VIM_AD_PLAUSI_CHECK VIM_AD_OK VIM_AD_ERR	1	[-]
State of VIM-Adaptation					
LV_VIM_AD_REQ	V/O	0...1H	0...1	1	[-]
Logical variable for requested VIM- adaptation					
LV_VIM_SP_AD	V/O	0...1H	0...1	1	[-]
VIM setpoint during active adaptation					
LV_VIM_AD_ACT	V/O	0...1H	0...1	1	[-]
Logical variable for active VIM- adaptation (switch to requested positions, plausibility check)					
T_MIN_PUC_VIM_AD	V	0...FFH	0...5.1	0.02	[s]
Time to stabilize the conditions in PUC before start of adaptation					
T_HLD_VIM_AD	V	0...FFH	0...5.1	0.02	[s]
Time to hold the VIM in a defined end position before adaptation is started					
T_VIM_AD	V	0...FFFFH	0...1310.7	0.02	[s]
Time of VIM-Adaptation					
VIM_AD_ERR_CTR	V	0...FFH	0...255	1	[-]
Counter for failed VIM adaptation					

Input data:

LV_IGK	V_VIM	LV_PUC	PQ
VB_MMV	TCO	N_32	LV_ERR_VIM
LV_ERR_VCC_SENS_SUB	LV_ERR_VIM_FB_EL	CONF_VIM	

FUNCTION DESCRIPTION:

General information:

The delivered voltage from the VIM- potentiometer indicates the position of the variable intake manifold. Physically the potentiometer delivers a linear rising voltage-characteristic with 0...5V. The stops in short and long position are adapted once for the whole engine life.

This occurs while engine operating state PUC, to avoid drivability effects and to guarantee sufficient low pressure in the intake manifold for the vacuum box.

The learned values are stored at the end of the driving cycle as „non-volatile“.

If no VIM- adaptation was fulfilled for the whole engine life or the adaptation values were lost (LV_VIM_AD_VLD = 0) the values V_VIM_AD_SHO/ LONG are set to default values. Otherwise these values are stored in the non volatile memory of the ECU and are valid till the next adaptation was successful.

While adaptation is active and STATE_VIM_AD = VIM_AD_SHORT_STOP the setpoint LV_VIM_SP_AD is set to 1 (request to control the control valve). After a calibratable delay time (C_T_HLD_VIM_AD) V_VIM_SHO is calculated as an average value (starting with V_VIM_AD_SHO) for C_T_VIM_AD seconds.

Then the VIM is set to the long position (STATE_VIM_AD = VIM_AD_LONG_STOP) (LV_VIM_SP_AD is set to 0) and after the same hold time V_VIM_LONG is calculated as an average value (starting with V_VIM_AD_LONG) for C_T_VIM_AD seconds.

If these values V_VIM_SHO and V_VIM_LONG are within a valid range, these values are copied into V_VIM_AD_SHO/ LONG and saved (together with LV_VIM_AD_VLD) in the non volatile memory of the ECU.

The adaptation has to be calculated immediately before adaptation diagnosis.

Application conditions:

Initialisation: at reset and LV_IGK = 0 $\rightarrow$ 1

	VIM_AD_ERR_CTR = 0	
If	at least one VIM adaptation took place for whole engine life	
and	no checksum error was detected	
Then	LV_VIM_AD_VLD = 1	*;adaptation values available*
Else	LV_VIM_AD_VLD = 0	*;no adaptation values are available*
Endif		

If	LV_VIM_AD_VLD = 1	
Then	V_VIM_SHO = V_VIM_AD_SHO	
	V_VIM_LONG = V_VIM_AD_LONG	
	STATE_VIM_AD = VIM_AD_OK	
	LV_VIM_AD_REQ = 0	
Else	V_VIM_AD_SHO = C_V_VIM_SHO_INI	*;never a successful adaptation*
	V_VIM_SHO = C_V_VIM_SHO_INI	*fulfilled, no adaptation values*
	V_VIM_AD_LONG = C_V_VIM_LONG_INI	*available,*
	V_VIM_LONG = C_V_VIM_LONG_INI	
	STATE_VIM_AD = VIM_AD_PUC_TIMER	
	LV_VIM_AD_REQ = 1	
	LV_VIM_SP_AD = 0	
	T_MIN_PUC_VIM_AD = C_T_MIN_PUC_VIM_AD	

Endif

Recurrence: 20 ms

Activation: CONF_VIM = 2

Formula section:

Activation of adaptation:

```
If      (LV_VIM_AD_REQ = 1                           and    ;adaptation requested
        [(LC_VIM_AD_PUC_ENA = 1 and   LV_PUC=1)
        or      LC_VIM_AD_PUC_ENA = 0 ]
        TCO > C_TCO_MIN_VIM_AD                       and
        N_32 > C_N_32_MIN_VIM_AD                     and
        N_32 < C_N_32_MAX_VIM_AD                     and
        VB_MMV > C_VB_MMV_MIN_VIM_AD                 and
        PQ < C_PQ_MAX_VIM_AD                         and
        PQ > C_PQ_MIN_VIM_AD                         and
        LV_ERR_VIM = 0                               and
        LV_ERR_VCC_SENS SUB = 0                      and
        LV_ERR_VIM_FB_EL = 0
Then    timer T_MIN_PUC_VIM_AD is started and decremented
Else            LV_VIM_AD_ACT = 0
                LV_VIM_SP_AD = 0
                T_MIN_PUC_VIM_AD = C_T_MIN_PUC_VIM_AD
        If      STATE_VIM_AD ≠ VIM_AD_OK
        and     STATE_VIM_AD ≠ VIM_AD_ERR
        then    STATE_VIM_AD = VIM_AD_PUC_TIMER
        Endif
Endif
If      T_MIN_PUC_VIM_AD = 0
  and   STATE_VIM_AD = VIM_AD_PUC_TIMER
Then    STATE_VIM_AD = VIM_AD_SHORT_STOP
        T_HLD_VIM_AD = C_T_HLD_VIM_AD
        T_VIM_AD = C_T_VIM_AD
Endif
```

Adaptation algorithm (this is a sequence of actions to do in this order):

Adaptation of short position:

If STATE_VIM_AD = VIM_AD_SHORT_STOP

Then LV_VIM_AD_ACT = 1 *;VIM adaptation active*

LV_VIM_SP_AD = 0 *;switching to short position*

timer T_HLD_VIM_AD is decremented

If timer T_HLD_VIM_AD = 0 *;enough time in short position*

Then timer T_VIM_AD is decremented

Endif

If T_VIM_AD > 0 *;adaptation value calculation active*

Then $V\_VIM\_SHO_N = V\_VIM\_SHO_{N-1} * (1 – C\_CRLC\_V\_VIM) + V\_VIM_N * C\_CRLC\_V\_VIM$

Else STATE_VIM_AD = VIM_AD_LONG_STOP

T_HLD_VIM_AD = C_T_HLD_VIM_AD

T_VIM_AD = C_T_VIM_AD

;adaptation of short position finished, switch to next adaptation step

Endif

Endif

Adaptation of long position:

If STATE_VIM_AD = VIM_AD_LONG_STOP

Then LV_VIM_SP_AD = 1 *;switching to long position*

T_HLD_VIM_AD is decremented

If timer T_HLD_VIM_AD = 0 *;enough time in long position*

Then timer T_VIM_AD is decremented

Endif

If T_VIM_AD > 0 *;adaptation value calculation active*

Then $V\_VIM\_LONG_N = V\_VIM\_LONG_{N-1} * (1 – C\_CRLC\_V\_VIM) + V\_VIM_N * C\_CRLC\_V\_VIM$

Else STATE_VIM_AD = VIM_AD_PLAUSI_CHECK

;adaptation of long position finished, switch to next adaptation step

Endif

Endif

Plausibility check:

If STATE_VIM_AD = VIM_AD_PLAUSI_CHECK

Then **If** $|V\_VIM\_SHO - C\_V\_VIM\_SHO\_INI| \leq C\_V\_VIM\_AD\_SHO\_HYS$

And $|V\_VIM\_LONG - C\_V\_VIM\_LONG\_INI| \leq C\_V\_VIM\_AD\_LONG\_HYS$

Then V_VIM_AD_SHO = V_VIM_SHO *;new adaptation value to be*

V_VIM_AD_LONG = V_VIM_LONG *saved in EEPROM*

LV_VIM_AD_VLD = 1 *;adaptation values are valid;*

STATE_VIM_AD = VIM_AD_OK *;adaptation finished and o.k.*

Else **If** VIM_AD_ERR_CTR >= C_VIM_AD_ERR_CTR_MAX

Then STATE_VIM_AD = VIM_AD_ERR

LV_VIM_AD_VLD = 0

Else VIM_AD_ERR_CTR = VIM_AD_ERR_CTR $_{n-1}$ +1

STATE_VIM_AD = VIM_AD_PUC_TIMER

LV_VIM_AD_VLD = 0

LV_VIM_AD_REQ = 1 *;adaptation not finished*

LV_VIM_AD_ACT = 1 *;adaptation active*

Endif

Endif

Endif

Adaptation finished:

IF STATE_VIM_AD = VIM_AD_OK

or STATE_VIM_AD = VIM_AD_ERR

Then LV_VIM_AD_ACT = 0 *;adaptation inactive*

LV_VIM_AD_REQ = 0 *;adaptation finished*

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_VIM_SHO_INI	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Default value for VIM feedback for VIM in short position					
C_V_VIM_LONG_INI	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Default value for VIM feedback for VIM in long position					
C_T_MIN_PUC_VIM_AD	1	0...FFH	0...5.1	0.02	[s]
Minimum time in PUC with fulfilled conditions to start adaptation algorithm					
C_TCO_MIN_VIM_AD	1	0...FEH	-48...142.5	0.75	[°C]
Minimum TCO threshold to start adaptation					
C_N_32_MIN_VIM_AD	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed to start adaptation					
C_N_32_MAX_VIM_AD	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed to start adaptation					
C_VB_MMV_MIN_VIM_AD	1	0...FFH	0...25.89843	0.1015625	[V]
Minimum battery voltage to start adaptation					
C_T_HLD_VIM_AD	1	0...FFH	0...5.1	0.02	[s]
Time to hold VIM position before adaptation starts					
C_T_VIM_AD	1	0...1FEH	0...10.2	0.02	[s]
Time of adaptation mean value calculation					
C_V_VIM_AD_SHO_HYS	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Hysteresis on C_V_VIM_SHO_INI to detect a valid adaptation value					
C_V_VIM_AD_LONG_HYS	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Hysteresis on C_V_VIM_LONG_INI to detect a valid adaptation value					
C_PQ_MAX_VIM_AD	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Maximum PQ threshold to start adaptation					
C_PQ_MIN_VIM_AD	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Minimum PQ threshold to start adaptation					
C_CRLC_V_VIM	1	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation constant for adaptation value evaluation					
C_VIM_AD_ERR_CTR_MAX	1	0...FFH	0...255	1	[-]
Maximum counter of failed VIM adaptation befor entering VIM_AD_ERR -state					
LC_VIM_AD_PUC_ENA	1	0...1H	0...1	1	[-]
Logical variable for VIM adaptation in PUC (1=only in PUC ; 0= at all engine states)					

9.9 Variable intake manifold

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_VIM_SP	O/V	0... 1H	0... 1	1	[-]
Logical variable of VIM control (value after final switch control)					
LV_VIM_SP_ENA	V	0... 1H	0... 1	1	[-]
Logical variable for enabled					
LV_VIM_SP_RAW	V	0... 1H	0... 1	1	[-]
Logical variable of VIM control (raw value)					
MAF_SP_TQI_VIM_SWI	V	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Load setpoint for for switching VIM (with hysteresis)					
N_32_VIM_COR	V	0... FFH	0... 8160	32	[rpm]
Engine speed offset to predict the engine speed at switch point					
N_32_VIM_SWI	V	0... FFH	0... 8160	32	[rpm]
Engine speed for switching VIM (with hysteresis)					

Input Data:

N_32	CONF_VIM	GEAR	LV_ACT_EXT_ADJ_VIM
LV_ERR_MAF	LV_ERR_MAP	LV_EXT_ADJ_VIM	LV_IS
MAF_SP_TQI	N_GRD	TCO	TIA_THR
T_AST	VB	LV_FL	LV_VIM_AD_ACT
LV_VIM_SP_AD			

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_GEAR_MIN_VIM	1	0... 6H	0... 6	1	[-]
Minimum gear to enable the VIM control					
C_MAF_SP_TQI_HYS_VIM	1	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Hysteresis of engine load to switch VIM					
C_N_32_HYS_VIM	1	0... FFH	0... 8160	32	[rpm]
Hysteresis of engine speed to switch VIM					
C_T_AST_MIN_VIM	1	0... FFFFH	0... 6553.5	0.1	[s]
Minimum time after start to enable the VIM control					
C_TCO_MIN_VIM	1	0... FEH	-48... 142.5	0.75	[°C]
Coolant temperature to enable the VIM control					
C_TIA_THR_MIN_VIM	1	0... FEH	-48... 142.5	0.75	[°C]
Air temperature at throttle to enable the VIM control					
C_VB_MAX_VIM	1	0... FFH	0... 25.8984375	0.1015625	[V]
maximum battery voltage to release VIM function					
C_VB_MIN_VIM	1	0... FFH	0... 25.8984375	0.1015625	[V]
minimum battery voltage to release VIM function					
ID_VIM_SP	8*8	0... 1H	0... 1	1	[-]
LDPM_N_32_VIM_COR_VIM_SP	8	0... FFH	0... 8160	32	[rpm]
LDP_MAF_SP_TQI_VIM_SP	8	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Map for VIM- setpoint					
ID_VIM_SP_FL	8	0... 1H	0... 1	1	[-]
LDPM_N_32_VIM_COR_VIM_SP	8	0... FFH	0... 8160	32	[rpm]
Map for VIM - setpoint at Full Load					
IP_N_32_VIM_OFS	4*4	0... FFH	-4096 ...4064	32	[rpm]
LDP_N_32_N_32_VIM_OFS	4	0... FFH	0... 8160	32	[rpm]

LDP_N_GRD_N_32_VIM_OFS	4	0... FFH	-4096 ...4064	32	[rpm/s]
Offset map for corrected engine speed for VIM control					

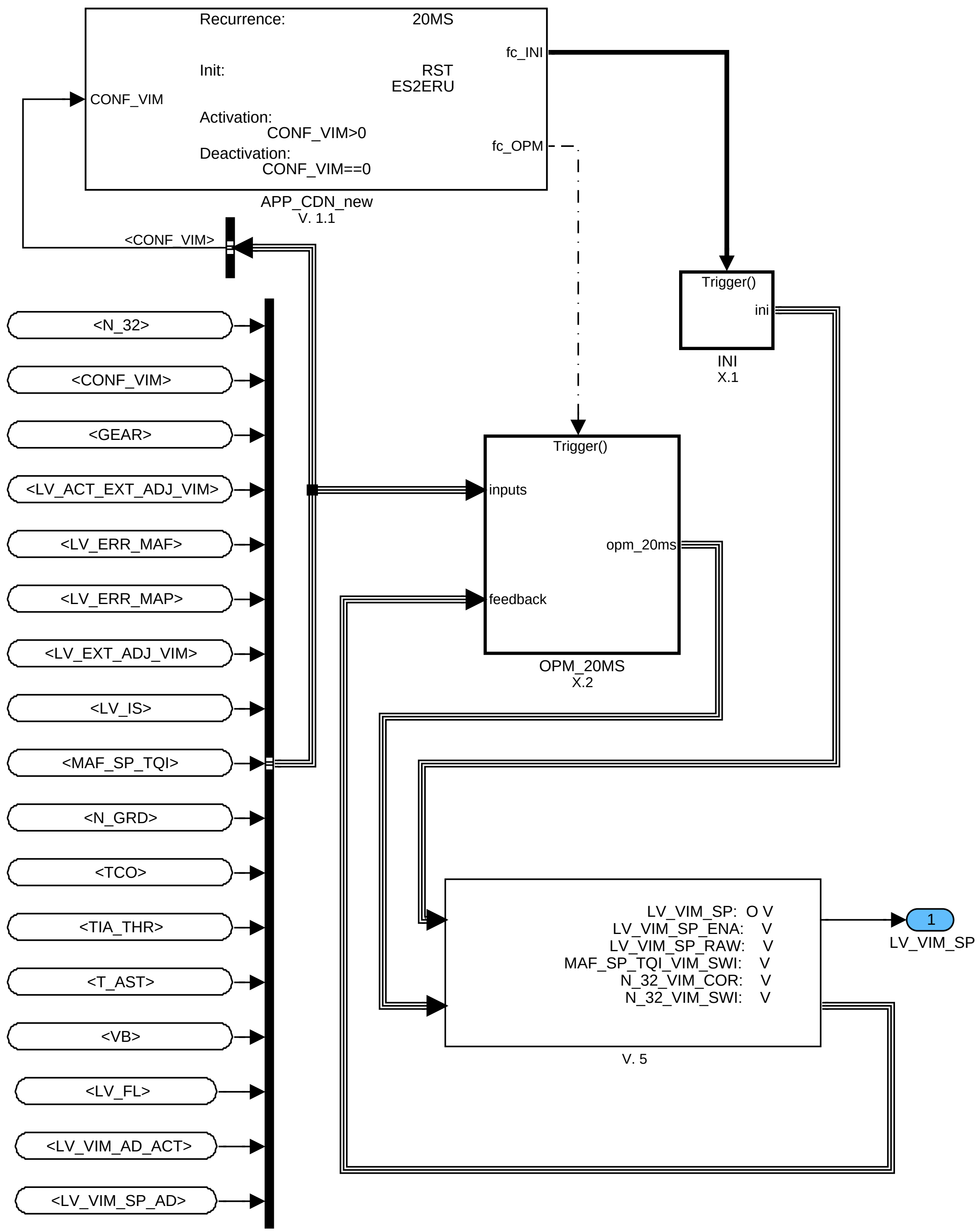
General Information

The variable intake manifold serves to increase the engine torque by adjusting the length of the intake manifold. Lengthen the intake manifold increases the engine torque with low engine speed, shorten the lenght helps to increase engine power with high engine speed.
The setpoint of the variable intake manifold (LV_VIM_SP) is set by the (not interpolated) table ID_VIM_SP depending on N_32 and MAF_SP_TQI.

Application Conditions

Initialization: RST, ES2ERU
Recurrence: 20MS
Activation: CONF_VIM > 0
Deactivation: CONF_VIM == 0

Function description



SDA_SRS / SDA V 5.0.4 17−Jan−2007

Figure 5:
Path: VIMA_M9006

9.9.1 Initialisation

9.9.1.1 Initialisation at Reset

All output variables initialised to 0.

9.9.1.2 Initialisation at Reset or LV_ES = 0 -> 1

Figure 6:
Path: VIMA_M9006/INI/INI_RST_ES2ERU

9.9.2 Recurrence 20 ms

9.9.2.1 Calculation of the final VIM - setpoint

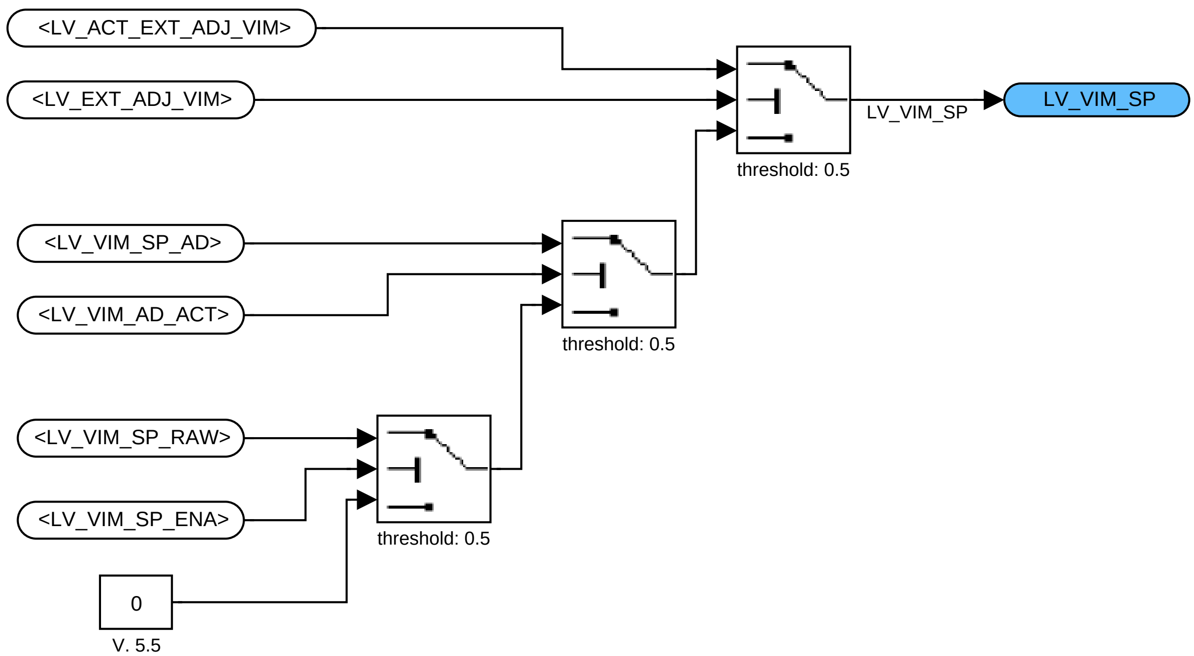


Figure 7:
Path: VIMA_M9006/OPM_20MS/SWI_CTL

9.9.2.2 Enable conditions for VIM - control

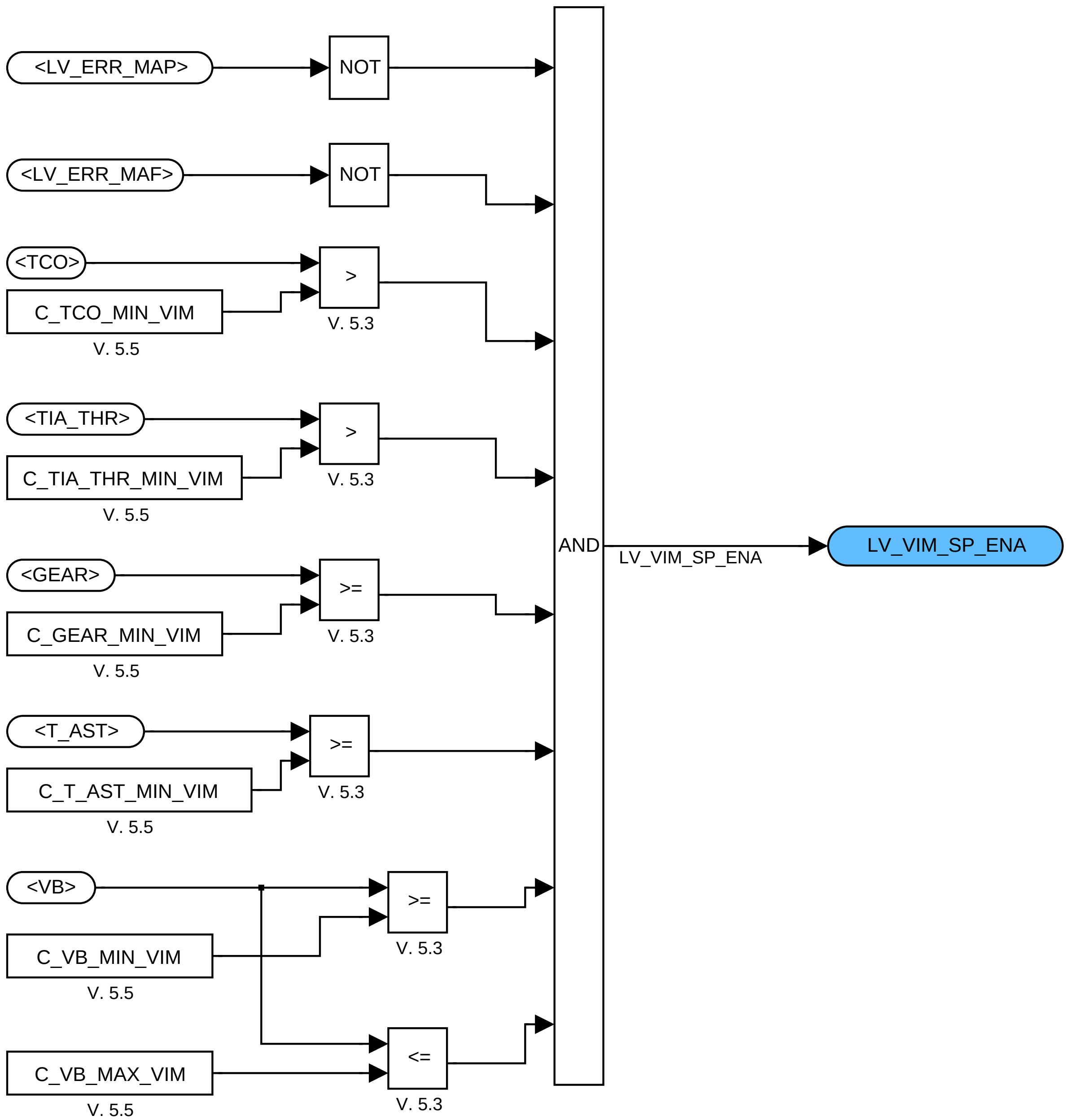


Figure 8:
Path: VIMA_M9006/OPM_20MS/VIM_SP_ENA

9.9.2.3 Calculation of the basic VIM - setpoint LV_VIM_SP_RAW

To avoid drivability problems by switching too often, hysteresises on the switching engine speed and load setpoint are set. If the value of N_32_VIM_COR or MAF_SP_TQI leaves the hysteresis, then the new setpoint is fixed via ID_VIM_SP and N_32_VIM_COR and MAF_SP_TQI are saved as N_32_VIM_SWI and MAF_SP_TQI_VIM_SWI.

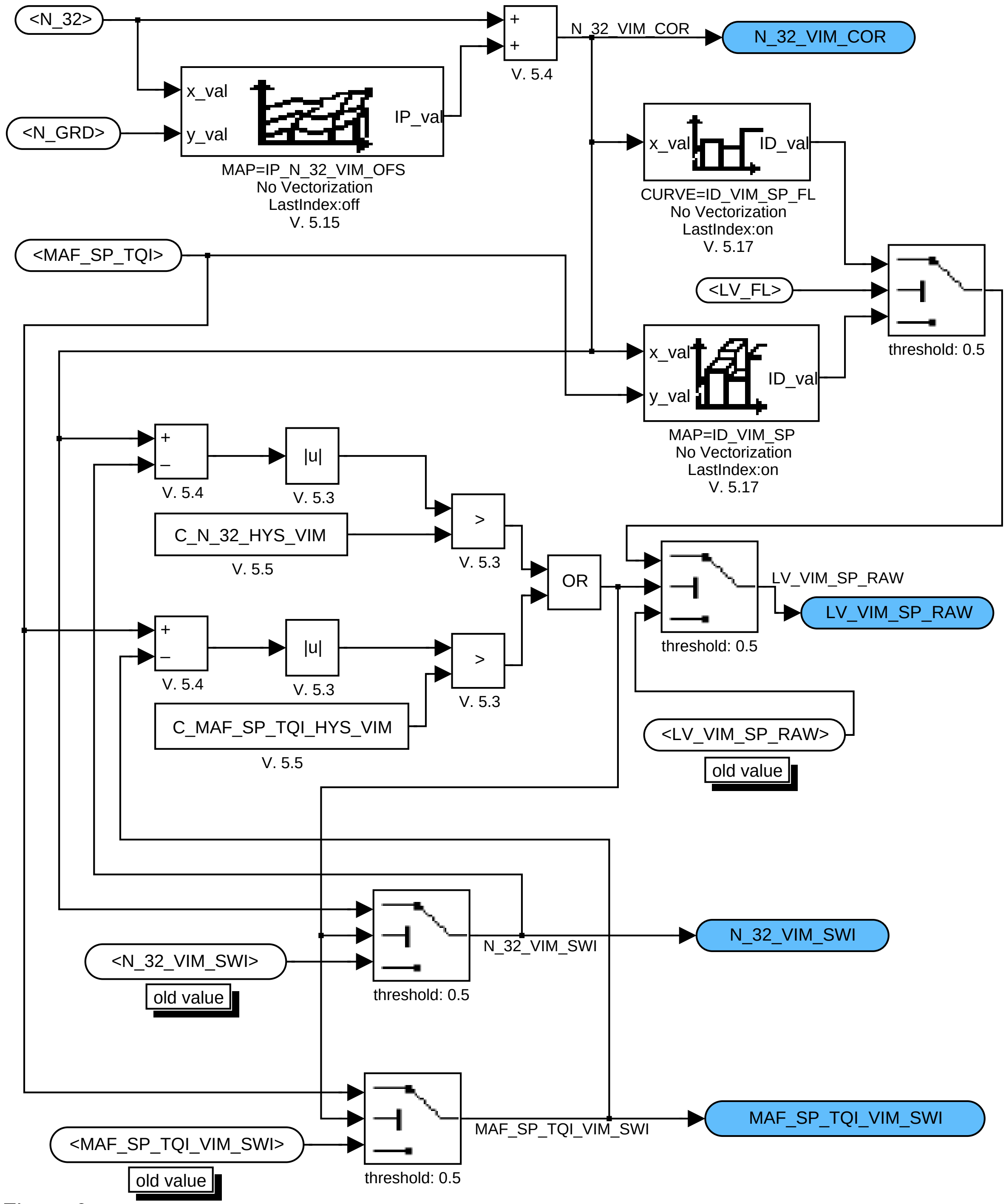


Figure 9:
Path: VIMA_M9006/OPM_20MS/VIM_SP_RAW

9.10 Translation module for extended MAF variables

9.10.1 Exported MAF variables from INSY with extended range that are calculated segment synchronously

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
MAF_KGH_ACQ	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Mass air flow per segment measured					
MAF_KGH_CYL	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Air mass flow out of the intake manifold					
MAF_KGH_ENG	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Air mass flow per segment in kg/h					
MAF_KGH_FG_CYL	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Fresh air mass flow in the cylinder					
MAF_KGH_FG_PRED_CYL	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Predicted fresh gas out of the manifold					
MAF_KGH_MDL_DIF	V/O	8000...7FFFH	-2048... 2047.9375	0.0625	[kg/h]
Deviation of model air mass flow					
MAF_KGH_MDL_MV	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Model air mass flow mean value					
MAF_KGH_THR	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Air mass flow at the throttle					
MAF_STK_DIF	V/O	8000...7FFFH	-1389.02119... 1388.97880	0.0423896	[mg/stk]
Mass air flow difference per segment					
MAF_STK_FG_PRED	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
Predicted air mass for calculation of basic fuel injection					
MAF_STK_FG_PRED_HB	V/O	0...FFH	0...2767.19066	10.851728	[mg/stk]
same as MAF but with a lower resolution (HB means High Byte)					
MAF_STK_MMV	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
MAF moving mean value					
MAF_STK_CYL	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
Air mass flow out of the intake manifold per stroke					

Input data:

MAF	MAF_CYL	MAF_DIF	MAF_FG_CYL
MAF_HB	MAF_KGH	MAF_KGH_FG_PRED	MAF_KGH_MES
MAF_MDL_MV	MAF_MMV	MAF_THR	MAF_MDL_DIF
MAF_CYL_STK			

FUNCTION DESCRIPTION:

General information:

As there are projects that work with a higher mass air flow than other projects before, an extension of the MAF ranges has to be done. This module provides all the MAF variables that are exported out of the aggregate INSY with the new ranges and the new resolution. So the interfaces between INSY and other aggregates are satisfied.

Application conditions:

Activation: all engine operating states

Initialisation:

Recurrence: segment synchronously

Formula section:

MAF_KGH_ACQ = MAF_KGH_MES
MAF_KGH_CYL = MAF_CYL
MAF_KGH_ENG = MAF_KGH
MAF_KGH_FG_CYL = MAF_FG_CYL
MAF_KGH_FG_PRED_CYL = MAF_KGH_FG_PRED
MAF_KGH_MDL_DIF = MAF_MDL_DIF
MAF_KGH_MDL_MV = MAF_MDL_MV
MAF_KGH_THR = MAF_THR
MAF_STK_DIF = MAF_DIF
MAF_STK_FG_PRED = MAF
MAF_STK_FG_PRED_HB = MAF_HB
MAF_STK_MMV = MAF_MMV
MAF_STK_CYL = MAF_CYL_STK

9.10.2 Exported MAF variables from INSY with extended range that are calculated every 10ms

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
MAF_KGH_SP_ENG	V/O	0...FFFFH	0...4095.9375	0.0625	[kg/h]
Setpoint mass air flow into the manifold					
MAF_STK_SP	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
MAF setpoint output for inverse air path					
MAF_STK_SP_S	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
MAF setpoint for stratified mode					

Input data:

MAF_KGH_SP	MAF_SP	MAF_SP_S	NC_INJ_CONF

FUNCTION DESCRIPTION:

General information:

As there are projects that work with a higher mass air flow than other projects before, an extension of the MAF ranges has to be done. This module provides all the MAF variables that are exported out of the aggregate INSY with the new ranges and the new resolution. So the interfaces between INSY and other aggregates are satisfied.

Application conditions:

Activation: all engine operating states

Initialisation:

Recurrence: 10ms

Formula section:

MAF_KGH_SP_ENG = MAF_KGH_SP
MAF_STK_SP = MAF_SP
#If NC_INJ_CONF = 1
#Then
MAF_STK_SP_S = MAF_SP_S
#Else
MAF_STK_SP_S = 0
#Endif

9.10.3 Exported MAF variables from INSY with extended range that are calculated every 20ms

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
MAF_KGH_INT_PUC_ACT	V/O	0...FFFFH	0...5825.33333	0.0888889	[g]
air mass flow integral during pull cut off phase					
MAF_KGH_INT_PUC_NOT_ACT	V/O	0...FFFFH	0...5825.33333	0.0888889	[g]
air mass flow integral out of pull cut off phase					

Input data:

MAF_INT_PUC_ACT	MAF_INT_PUC_NOT_ACT		

FUNCTION DESCRIPTION:

General information:

As there are projects that work with a higher mass air flow than other projects before, an extension of the MAF ranges has to be done. This module provides all the MAF variables that are exported out of the aggregate INSY with the new ranges and the new resolution. So the interfaces between INSY and other aggregates are satisfied.

Application conditions:

Activation: all engine operating states

Initialisation:

Recurrence: 20ms

Formula section:

MAF_KGH_INT_PUC_ACT = MAF_INT_PUC_ACT
MAF_KGH_INT_PUC_NOT_ACT = MAF_INT_PUC_NOT_ACT

9.10.4 Exported MAF variables from INSY with extended range that are calculated every 40ms

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
MAF_STK_MAX_COR	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
Maximum possible fresh air entering in the cylinder (open throttle)					
MAF_STK_MIN_COR	V/O	0...FFFFH	0...2778	0.0423896	[mg/stk]
Minimum available fresh air going into the cylinder					

Input data:

MAF_MAX_COR	MAF_MIN_COR		

FUNCTION DESCRIPTION:

General information:

As there are projects that work with a higher mass air flow than other projects before, an extension of the MAF ranges has to be done. This module provides all the MAF variables that are exported out of the aggregate INSY with the new ranges and the new resolution. So the interfaces between INSY and other aggregates are satisfied.

Application conditions:

Activation: all engine operating states

Initialisation:

Recurrence: 40ms

Formula section:

MAF_STK_MAX_COR = MAF_MAX_COR
MAF_STK_MIN_COR = MAF_MIN_COR

9.11 General Port Flap

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PORTPWM_TMP	V/O	8000H...7FFFH	-100...99.99	0.003	%
PORTPWM before overload protection					
V_PORT	V/O	0...3FFH	0...5	0.00488	V
Port flap potentiometer signal					
FRQ_PORTPWM	V/O	0...FFH	1000...8650	30	Hz
Frequency of PWM for Port Flap					

Input data:

PORTPWM_CTL	PORT_FB_BAS	PORTPWM_DIAG	CONF_PORT
LV_ERR_PORT	LV_PORT_AD_VLD	V_PORT_AD_TOL	V_PORT_AD_BOL
PORT_SP			

FUNCTION DESCRIPTION:

General information:

The port flap inside the intake port influences the gas flow into the combustion chamber, and thus, the gas motion within the chamber. The activation of the port flap increases the gas velocity into the combustion chamber. This improves the gas mixture formation process, ignition properties of the mixture and the combustion process. The port flap is only closed one time after start and opened again dependent on different after Start conditions

The intake port is fully open if the flap is in 0° position and fully closed at 73°. The port flap is driven by an electric DC motor with worm gear. The motor is driven by an H-bridge and energized with a duty cycle. Positive duty cycles activate the worm gear motor to close the Port Flap negative values of the duty cycle are used to open the Port flap. 0% duty cycle keep the port flap in the actual position due to the selflocking of the worm gear. The actual port flap position is measured by means of a potentiometer, PORT_FB_BAS.

To avoid hard strikes against the mechanical stop positions the PORTPWM_TMP is limited to C_PORTPWM_TMP_MAX during acting of PORT Flap close to the mechanical stop positions, which are learned during Adaptation.

Application conditions:

Recurrence: 10 ms

Activation: CONF_PORT = 1

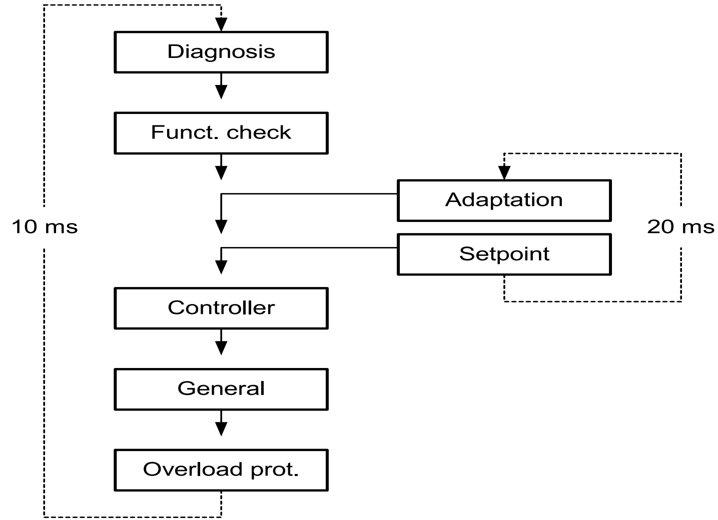


Figure: Sequence of port flap tasks.

Formula section:

Frequency of PWM for Port Flap:

FRQ_PORTPWM = C_FRQ_PORTPWM

Potentiometer signal:

V_PORT = PORT_FB_BAS

Determination of port flap energization PWM:

If$_1$ LV_ERR_PORT = 0

Then$_1$

If$_2$ LV_PORT_AD_VLD= 1

Then$_2$ *mechanical Stop Position learned*

If$_3$ PORT_SP = 1

And V_PORT > V_PORT_AD_TOL – C_V_PORT_DIF_THD

Then$_3$ PORTPWM_TMP = min (PORTPWM_CTL; C_PORTPWM_TMP_MAX)

Limit Port PWM near TOL to avoid hard strikes against mechanical stop

Else$_3$

If$_4$ PORT_SP = 0

And V_PORT < V_PORT_AD_BOL + C_V_PORT_DIF_THD

Then$_4$ PORTPWM_TMP = max (PORTPWM_CTL;

-C_PORTPWM_TMP_MAX)

Limit Port PWM near BOL to avoid hard strikes against *mechanical stop*

Hint: As the PWM values to open PORT have to be negative here the <u>absolute</u> smallest value has to be result of the MIN election (i.e.-10% versus -30% --> PORTPWM_TMP is limited to -10%)

Else$_4$ PORTPWM_TMP = PORTPWM_CTL

Endif$_4$

Endif$_3$

Else$_2$ PORTPWM_TMP = PORTPWM_CTL

Endif$_2$

Else$_1$ PORTPWM_TMP = PORTPWM_DIAG

Endif$_1$

<u>Calibration data:</u>

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FRQ_PORTPWM	1	0...FFH	1000...8650	30	Hz
Frequency of PWM for Port Flap					
C_V_PORT_DIF_THD	1	0...3FFH	0...5	0.00488	V
Threshold to limit PORTPWM_TMP to avoid hard hard stop of Port Flap					
C_PORTPWM_TMP_MAX	1	8000H...7FFFH	-100...99.99	0.003	%
Max PORTPWM for Port Flap acting near mechanical stops to avoid hard strikes against stop position					

9.12 Port flap setpoint

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PORT_SP	V/O	0...1H	0...1	1	-
Port flap setpoint 0 = Port Flap open 1 = Port Flap closed					
LV_PORT_SP_PAS	V	0...1H	0...1	1	-
Bit indication, whether PORT SP remains open					

Input data:

LV_PORT_SP_EXT_ADJ	PORT_SP_EXT_ADJ	TCO_ST	VB
TIA_ST	T_AST	AMP	N_32
LV_PORT_AD_VLD	LV_PL	LV_IGK	

FUNCTION DESCRIPTION:

General information:

The port flap influences the gas flow into the combustion chamber, and thus, the gas motion within the chamber.

The Port Flap will be closed, if all activation conditions for closing are fullfilled (i.e TCO_ST, TIA_ST, AMP, VB). The closing is additionally allowed at warm engine if the adaptation has not been performed yet to be able to do adaptation at end of line test or at service. (Hint: This also has to be considered in the calibration of the IP_PORT_SP)

The Port Flap will be opened depending on T_AST and TCO_ST and will remain opened till the next restart.

The Port Flap is only closed during Idle speed condition and will be opened (and remains open) as soon as Part Load is detected

Additionally the Port Flap can be activated by external request or by Application system.

Application conditions:

Recurrence: 20 ms

Initialization: *At reset:* PORT_SP = 0 (open)

LV_PORT_SP_PAS = 0

At LV_IGK = 0--> 1: LV_PORT_SP_PAS = 0

Activation: In every engine state

Formula section:

Setpoint Calculation:

If$_1$ LC_PORT_SP_MAN = 1

Then$_1$ *Manual setpoint*

PORT_SP = C_PORT_SP_MAN

Else$_1$ *Normal setpoint algorithm*

If$_2$ LV_PORT_SP_EXT_ADJ = 1

Then$_2$ *External setpoint*

PORT_SP = PORT_SP_EXT_ADJ

Else$_2$ *Defined setpoint*

If$_3$ TCO_ST < C_TCO_MIN_PORT

or [TCO_ST >= C_TCO_MAX_PORT **and** LV_PORT_AD_VLD = 1]

to allow PORT_SP = 1 at warm engine only for adaptation

or TIA_ST < C_TIA_MIN_PORT

or AMP < C_AMP_MIN_PORT

or VB < C_VB_MIN_PORT

or LV_PL = 1

Then$_3$ LV_PORT_SP_PAS = 1

Condition for flap operation not fulfilled

PORT_SP = 0

Else$_3$ *Condition for flap operation fulfilled*

If$_4$ LV_IGK = 1

And LV_PORT_SP_PAS = 0

Then$_4$ *Port SP bit not passive*

PORT_SP = IP_PORT_SP

Else$_4$ PORT_SP = 0

Endif$_4$

Endif$_3$

Endif$_2$

Endif$_1$

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
IP_PORT_SP	8*4	0...1H	0...1	1	-
LDP_TCO_ST_IP_PORT_SP	8	0...FEH	-48...142.5	0.75	°C
LDP_T_AST_IP_PORT_SP	4	0...FFFFH	0...6553.5	0.1	s
Port flap setpoint					
C_TCO_MIN_PORT	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature for port flap performance					
C_TCO_MAX_PORT	1	0...FEH	-48...142.5	0.75	°C
Maximum coolant temperature for port flap performance					
C_TIA_MIN_PORT	1	0...FEH	-48...142.5	0.75	°C
Minimum intake air temperature for port flap performance					
C_AMP_MIN_PORT	1	0...FFFFH	0...5434	0.0829175	hPa
Minimum ambient pressures for port flap performance					
C_VB_MIN_PORT	1	0...FFH	0...26	0.1	V
Minimum battery voltage for port flap performance					
C_PORT_SP_MAN	1	0...1H	0...1	1	-
Manual port flap setpoint by application					
LC_PORT_SP_MAN	1	0...1H	0...1	1	-
Switch for manual port flap setpoint by application: 1 = manual, 0 = normal setpoint algorithm					

9.13 Port Flap Position

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PORT_AV	V/O	F000H...1000H	-100...100	0.0244	°PORT
Actual port flap position					
PORT_AV_GRD	V/O	F000H...1000H	-1000...1000	0.2441	°PORT/s
Gradient of PORT_AV					
FAC_PORT_DEAC	V/O	0...FFFFH	0...0.9999847	1.5258e-5	-
Relative port flap position for further calculations					
V_PORT_OPEN_MAX_TOT_DC	V/O/S	0...3FFH	0...5	0.00488	V
Former/current Driving cycle max. Port Poti value at open PORT					
V_PORT_OPEN_MIN_TOT_DC	V/O/S	0...3FFH	0...5	0.00488	V
Former/current Driving cycle min. Port Poti value at open PORT					
V_PORT_CLOSE_MAX_TOT_DC	V/O/S	0...3FFH	0...5	0.00488	V
Former/current Driving cycle max. Port Poti value at closed PORT					
V_PORT_CLOSE_MIN_TOT_DC	V/O/S	0...3FFH	0...5	0.00488	V
Former/current Driving cycle min. Port Poti value at closed PORT					
PORT_AV_GRD_OPEN_MAX_TOT_DC	V/O/S	F000H...1000H	-1000...1000	0.2441	°PORT/s
Former/current Driving cycle min. Port Gradient value Max limit					
PORT_AV_GRD_OPEN_MIN_TOT_DC	V/O/S	F000H...1000H	-1000...1000	0.2441	°PORT/s
Former/current Driving cycle min. Port Gradient value MIN limit					
PORT_AV_GRD_CLOSE_MAX_TOT_DC	V/O/S	F000H...1000H	-1000...1000	0.2441	°PORT/s
Former/current Driving cycle max. Port Gradient value MAX limit					
PORT_AV_GRD_CLOSE_MIN_TOT_DC	V/O/S	F000H...1000H	-1000...1000	0.2441	°PORT/s
Former/current Driving cycle max. Port Gradient value MIN limit					
PORT_AV_GRD_MAX_DC	V/O	F000H...1000H	-1000...1000	0.2441	°PORT/s
Max Port Gradient at current Driving cycle					
PORT_AV_GRD_MIN_DC	V/O	F000H...1000H	-1000...1000	0.2441	°PORT/s
Min Port Gradient at current Driving cycle					

Input data:

V_PORT	V_PORT_SLOPE	V_PORT_AD_BOL	PORT_SP
LV_ERR_PORT_POTI	C_PORT_BOL	C_PORT_TOL	T_PORT_DLY4
T_PORT_DLY5			

FUNCTION DESCRIPTION:

General information:

A linear potentiometer is used for the port flap position feed-back. The potentiometer is supplied by 5 V from the ECU. The physical range of the flap position corresponds to the electrical range of the Potentiometer, see the figure "Port flap position feed-back".

The real port flap stop position, C_PORT_BOL and C_PORT_TOL, and the real potentiometer signal are adapted after movement see "Adaptation of Lower and Higher Port Flap Stop Positions".

The actual port flap position PORT_AV is calculated from the measured signal V_PORT. As mentioned above, a linear potentiometer is used. The quantities V_PORT_BOL and V_PORT_SLOPE are determined in "Adaptation of Lower and Higher Port Flap Stop Positions". If there is an error of the potentiometer the replacement value C_PORT_AV_DFT is taken.

The port flap position is needed in other functionalities, e.g., volumetric efficiency. In order to express the opening relatively the relative port flap position FAC_PORT_DEAC is calculated. The reference angle is C_PORT_TOL, so FAC_PORT_DEAC has its values from 0 to 1.

For Fleet Monitoring the MIN/MAX values of Potentiometer and Gradient are stored.

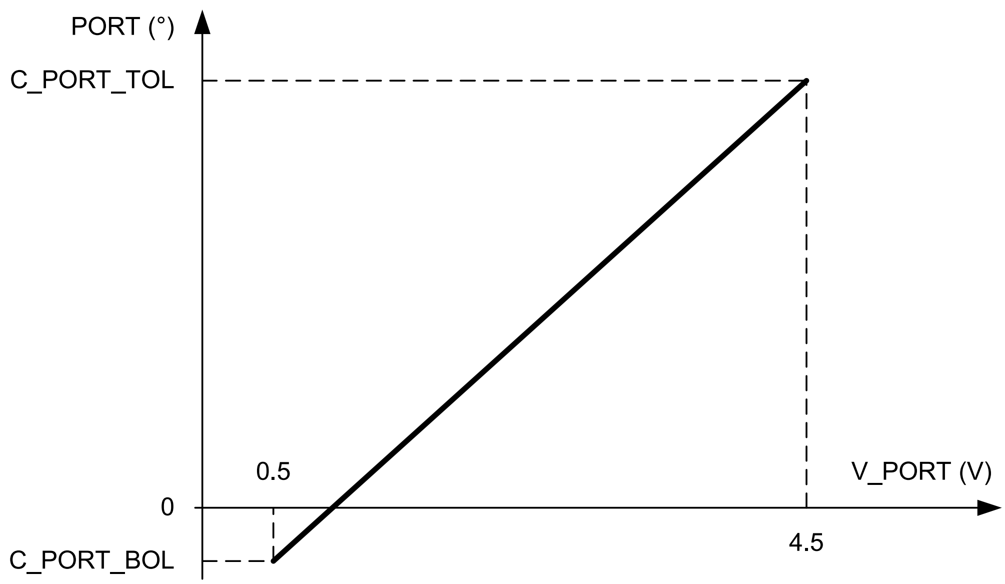


Figure: Port flap position feed-back.

Application conditions:

Initialisation: *at first Power up or loss of NVMY*

V_PORT_OPEN_MAX_TOT_DC = 0H (0V)

V_PORT_OPEN_MIN_TOT_DC = 3FFH (5V)

V_PORT_CLOSE_MAX_TOT_DC = 0H (0V)

V_PORT_CLOSE_MIN_TOT_DC = 3FFH (5V)

PORT_AV_GRD_OPEN_MAX_TOT_DC = F000H (-1000°/sec)

PORT_AV_GRD_OPEN_MIN_TOT_DC = 0H (0°/sec)

PORT_AV_GRD_CLOSE_MAX_TOT_DC = 0H (0°/sec)

PORT_AV_GRD_CLOSE_MIN_TOT_DC = 1000H (1000°/sec)

Otherwise restored from NVMY

at reset:

PORT_AV_GRD_MAX_DC = 0H (0°/sec)

PORT_AV_GRD_MIN_DC = 0H (0°/sec)

Recurrence: 10 ms

Activation: At every engine state

Formula section:

Port flap position feed-back:

if$_1$ LV_ERR_PORT_POTI = 1

then$_1$ PORT_AV = C_PORT_AV_DFT

else$_1$ PORT_AV = (V_PORT - V_PORT_AD_BOL) * V_PORT_SLOPE + C_PORT_BOL

endif$_1$ PORT_AV_GRD = (PORT_AV(n) – PORT_AV(n-4))*25 /sec

Calculation of FAC_PORT_DEAC:

FAC_PORT_DEAC = abs(PORT_AV) / C_PORT_TOL

Calculation of MIN/MAX values for Fleet Monitoring:

```
if          PORT_SP = 0
      and   PORT_AV_GRD <  - C_PORT_GRD_MOVE_DET
            movement of Port Flap is detected (opening)
then  if    PORT_AV_GRD < PORT_AV_GRD_MIN_DC
      then  PORT_AV_GRD_MIN_DC = PORT_AV_GRD
      endif

if          PORT_SP = 1
      and   PORT_AV_GRD > C_PORT_GRD_MOVE_DET
            movement of Port Flap is detected (closing)
then  if    PORT_AV_GRD > PORT_AV_GRD_MAX_DC
      then  PORT_AV_GRD_MAX_DC = PORT_AV_GRD
      endif
endif

if          PORT_SP = 0
      and   T_PORT_DLY4 = 0
            Port Flap open and not moving anymore
then  if          V_PORT < V_PORT_OPEN_MIN_TOT_DC
```

```
        then            V_PORT_OPEN_MIN_TOT_DC = V_PORT
        endif
        if              V_PORT > V_PORT_OPEN_MAX_TOT_DC
        then            V_PORT_OPEN_MAX_TOT_DC = V_PORT
        endif
        if              PORT_AV_GRD_MIN_DC < - C_PORT_GRD_MOVE_DET
                        Port Flap already opened before
        then    if      PORT_AV_GRD_MIN_DC < PORT_AV_GRD_OPEN_MIN_TOT_DC
                then    PORT_AV_GRD_OPEN_MIN_TOT_DC = PORT_AV_GRD_MIN_DC
                endif
                if      PORT_AV_GRD_MIN_DC > PORT_AV_GRD_OPEN_MAX_TOT_DC
                then    PORT_AV_GRD_OPEN_MAX_TOT_DC= PORT_AV_GRD_MIN_DC
                endif
        endif
endif

if              PORT_SP = 1
        and     T_PORT_DLY5 = 0
                Port Flap closed and not moving anymore
then    if              V_PORT < V_PORT_CLOSE_MIN_TOT_DC
        then            V_PORT_CLOSE_MIN_TOT_DC = V_PORT
        endif
        if              V_PORT > V_PORT_CLOSE_MAX_TOT_DC
        then            V_PORT_CLOSE_MAX_TOT_DC = V_PORT
        endif
        if              PORT_AV_GRD_MAX _DC > C_PORT_GRD_MOVE_DET
                        Port Flap already closed before
        then    if      PORT_AV_GRD_MAX_DC < PORT_AV_GRD_CLOSE_MIN_TOT_DC
                then    PORT_AV_GRD_CLOSE_MIN_TOT_DC = PORT_AV_GRD_MAX_DC
                endif
                if      PORT_AV_GRD_MAX_DC > PORT_AV_GRD_CLOSE_MAX_TOT_DC
                then    PORT_AV_GRD_CLOSE_MAX_TOT_DC = PORT_AV_GRD_MAX_DC
                endif
        endif
```

endif **Calibration data:**

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PORT_AV_DFT	1	F000H...1000H	-100...100	0.0244	°PORT
Replacement value for port flap position in case of potentiometer error					
C_PORT_GRD_MOVE_DET	1	F000H...1000H	-1000...1000	0.2441	°PORT/s
Threshold for PORT_AV_GRD movement detection					

9.14 Port Flap Position Controller

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PORTPWM_CTL	V/O	8000H...7FFFH	-100...99.99	0.003	%
Pulse width modulation after port position controller					
T_PORT_CLOSE_DLY	V	0...FFH	0...2.55	0.01	S
Timer to deactivate Port Flap CTL at PORT_AV_GRD smaller C_PORT_AV_GRD_CLOSE					
T_PORT_OPEN_DLY	V	0...FFH	0...2.55	0.01	S
Timer to deactivate Port Flap CTL at PORT_AV_GRD smaller C_PORT_AV_GRD_OPEN					
LV_PORT_TOL_CDN	V/O	0...1H	0...1	1	-
Flag indicating conditions for PORT_TOL check are fullfilled					
LV_PORT_BOL_CDN	V/O	0...1H	0...1	1	-
Flag indicating conditions for PORT_BOL check are fullfilled					
V_PORT_OLD_CLOSE	V	0...3FFH	0...5	0.00488	V
Old Position of Port Flap after closing (to detect drift of Flap)					
V_PORT_OLD_OPEN	V	0...3FFH	0...5	0.00488	V
Old Position of Port Flap after opening (to detect drift of Flap)					

Input data:

PORT_SP	V_PORT	PORT_AV_GRD	LV_ES

FUNCTION DESCRIPTION:

General information:

The Port Flap controller is driven by IP_PORTPWM_CTL_CLOSE or IP_PORTPWM_CTL_OPEN map dependent on the desired Setpoint and Potentiometer Position to open or to close the Flap. As soon as the Port Flap is not moving anymore and the difference between actual position and position after last energizing is smaller than a threshold a counter is decreased. If the counter is at 0 the PORTPWM_CTL is switched to Holding PWM for closed position to counteract Airflow at closed position and switched off for open position and the flags LV_PORT_TOL_CDN (for Closed position) respectively LV_PORT_BOL_CDN (for open Position) are set.

This flags are used to activate the check of mechanical stop positions. (see Adapatation of Port Flap Stop positions)

Application conditions:

Initialization: *at reset:*

PORTPWM_CTL=0%

T_PORT_CLOSE_DLY = C_T_PORT_CLOSE_DLY

T_PORT_OPEN _DLY = C_T_PORT_OPEN_DLY

V_PORT_OLD_CLOSE = 0H

V_PORT_OLD_OPEN = 0H

Recurrence: 10 ms

Activation: every engine state

Deactivation:

Formula section:

If$_1$ PORT_SP = 1

Then$_1$ T_PORT_OPEN_DLY = C_T_PORT_OPEN_DLY

request to close the Port Flap

If$_2$ abs (PORT_AV_GRD) < C_PORT_AV_GRD_CLOSE

and V_PORT_OLD_CLOSE – V_PORT < C_V_PORT_DIF_MAX_CLOSE

Then$_2$ *Port Flap is not moving anymore and actual position is not drifted more than C_V_PORT_DIF_MAX_CLOSE compared to old position*

If$_3$ T_PORT_CLOSE_DLY = 0

Then$_3$ **If$_4$** LV_ES = 0

Then$_4$ PORTPWM_CTL = C_PORTPWM_HLD_CLOSE

Holding PWM to counteract Air flow is applied at running engine

Else$_4$ PORTPWM_CTL = 0%

PWM is switched off at stopped engine

Endif$_4$

LV_PORT_TOL_CDN = 1

Condition for TOL value check are fulfilled

Else$_3$ PORTPWM_CTL = IP_PORTPWM_CTL_CLOSE

LV_PORT_TOL_CDN = 0

LV_PORT_BOL_CDN = 0

V_PORT_OLD_CLOSE = V_PORT

Decrement T_PORT_CLOSE_DLY by 1h each 10ms

Endif$_3$

Else$_2$ T_PORT_CLOSE_DLY = C_T_PORT_CLOSE_DLY

PORTPWM_CTL = IP_PORTPWM_CTL_CLOSE

LV_PORT_TOL_CDN = 0

LV_PORT_BOL_CDN = 0

V_PORT_OLD_CLOSE = V_PORT

Endif$_2$

Else$_1$ T_PORT_CLOSE_DLY = C_T_PORT_CLOSE_DLY

request to open the Port Flap

If$_4$ abs (PORT_AV_GRD) < C_PORT_AV_GRD_OPEN

and V_PORT – V_PORT_OLD_OPEN < C_V_PORT_DIF_MAX_OPEN

Then$_4$ *Port Flap is not moving anymore and actual position is not drifted more* *than C_V_PORT_DIF_MAX_OPEN compared to old position*

If$_5$ T_PORT_OPEN _DLY = 0

Then$_5$ PORTPWM_CTL = 0%

LV_PORT_BOL_CDN = 1

Condition for BOL value check are fulfilled

Else$_5$ PORTPWM_CTL = IP_PORTPWM_CTL_OPEN

LV_PORT_BOL_CDN = 0

LV_PORT_TOL_CDN = 0

V_PORT_OLD_OPEN = V_PORT

Decrement T_PORT_OPEN_DLY by 1h each 10ms

Endif$_5$

Else$_4$ T_PORT_OPEN_DLY = C_T_PORT_OPEN_DLY

PORTPWM_CTL = IP_PORTPWM_CTL_OPEN

LV_PORT_BOL_CDN = 0

LV_PORT_TOL_CDN = 0

V_PORT_OLD_OPEN = V_PORT

Endif$_4$

Endif$_1$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PORTPWM_CTL_CLOSE	12	0...FFFFH	-100...99.99	0.003	%
LDP_V_PORT_PORTPWM_CTL_CLOSE	12	0...3FFH	0...5V	0.0244	°PORT
PORT PWM control for closing Port Flap					
IP_PORTPWM_CTL_OPEN	12	0...FFFFH	-100...99.99	0.003	%
LDP_V_PORT_PORTPWM_CTL_OPEN	12	0...3FFH	0...5V	0.0244	°PORT
PORT PWM control for opening Port Flap					
C_PORT_AV_GRD_CLOSE	1	F000...1000H	-1000...1000	0.2441	°PORT/s
Minimum Threshold before T_PORT_CLOSE_DLY counter is activated					
C_T_PORT_CLOSE_DLY	1	0...FFH	0...2.55	0.01	s
Timer to deactivate PORTPWM_CTL at closing after low gradient is detected					
C_PORT_AV_GRD_OPEN	1	F000...1000H	-1000...1000	0.2441	°PORT/s
Minimum Threshold before T_PORT_OPEN_DLY counter is activated					
C_T_PORT_OPEN _DLY	1	0...FFH	0...2.55	0.01	s
Timer to deactivate PORTPWM_CTL at opening after low gradient is detected					
C_V_PORT_DIF_MAX_CLOSE	1	0...3FFH	0...5	0.00488	V
Max drift between old position and acutal position before VCM motor is reactivated (for closed position)					
C_V_PORT_DIF_MAX_OPEN	1	0...3FFH	0...5	0.00488	V
Max drift between old position and acutal position before VCM motor is reactivated (for open position)					
C_PORTPWM_HLD_CLOSE	1	8000H...7FFFH	-100...99.99	0.003	%
Holding PWM to counteract airflow in closed position					

9.15 Adaptation of Lower and Higher Port Flap Stop Positions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_PORT_AD_TOL	V/O/S	0...3FFH	0...5	4.8875e-3	V
Saved adapted TOL-value					
V_PORT_TOL	V	0...3FFH	0...5	4.8875e-3	V
TOL-value					
V_PORT_AD_BOL	V/O/S	0...3FFH	0...5	4.8875e-3	V
Saved adapted BOL-value					
V_PORT_BOL	V	0...3FFH	0...5	4.8875e-3	V
BOL-value					
V_PORT_SLOPE	V/O	0...800H	0...40	1.9531e-3	°PORT/V
Potentiometer slope					
LV_PORT_AD_VLD	V/O/S	0...1H	0...1	1	-
Adapted BOL-value and TOL-value are valid adaptation results					
LV_PORT_AD_TOL_VLD	V	0...1H	0...1	1	-
Apapted TOL values are valid					
LV_PORT_AD_BOL_VLD	V	0...1H	0...1	1	-
Adapted BOL values are valid					
LV_PORT_AD_ERR	V/O	0...1H	0...1	1	-
Adaptation error					

Input data:

V_PORT	LV_PORT_TOL_CDN	LV_PORT_BOL_CDN	LV_IGK
LV_ERR_VCC_SENS_SUB	LV_ERR_PORT_POTI	LV_ERR_PORT_EL	

FUNCTION DESCRIPTION:

General information:

The port flap position is measured by means of a linear potentiometer. The physical range of the flap position corresponds to the electrical range from In order to have individual value for each port flap an adaptation of lower and higher port flap stop position is done. The adaptation delivers the potentiometer signals of the flap in the stop positions V_PORT_AD_TOL and V_PORT_AD_BOL. They are saved in non-volatile memory, i.e., they are available at the beginning of a new driving cycle, if there has not been a checksum error. The potentiometer slope can be calculated from adapted values.

The adaptation is only performed in case of unvalid adaptation results, i.e adaptation was not sucessfully finished or due to external Adaptation request.

Application conditions:

9.15.1 Initialization:

At first power on or NVMY error :

LV_PORT_AD_VLD = 0

V_PORT_AD_TOL = C_V_PORT_AD_TOL

V_PORT_AD_BOL = C_V_PORT_AD_BOL

At Reset (Normal Initialisation):

LV_PORT_AD_VLD, V_PORT_AD_BOL and V_PORT_AD_TOL are restored from NVMY

V_PORT_TOL = C_V_PORT_AD_TOL
V_PORT_BOL = C_V_PORT_AD_BOL
V_PORT_SLOPE = (C_PORT_TOL - C_PORT_BOL) / (V_PORT_AD_TOL -
- V_PORT_AD_BOL)
LV_PORT_AD_ERR = 0
LV_PORT_AD_BOL_VLD = 0
LV_PORT_AD_TOL_VLD = 0

Recurrence: 20ms
Activation: *LV_PORT_AD_VLD = 0*
Deactivation: *LV_PORT_AD_VLD = 1*

Formula section:

Learning of BOL values (Port Flap open position)

if(1) LV_PORT_BOL_CDN = 1
and LV_IGK = 1
and LV_ERR_PORT_EL = 0
and LV_ERR_PORT_POTI = 0
and LV_ERR_VCC_SENS_SUB = 0
and LV_PORT_AD_TOL_VLD = 1
then(1) V_PORT_BOL = V_PORT
if(2) V_PORT_BOL <= C_V_PORT_BOL_MAX
and V_PORT_BOL >= C_V_PORT_BOL_MIN
(check violation of BOL limits)
then(2) LV_PORT_AD_BOL_VLD = 1
else(2) LV_PORT_AD_BOL_VLD = 0
LV_PORT_AD_ERR = 1
endif(2)
endif(1)

Learning of TOL values (Port Flap close position)

```
if(1)           LV_PORT_TOL_CDN = 1
       and      LV_IGK = 1
       and      LV_ERR_PORT_EL = 0
       and      LV_ERR_PORT_POTI = 0
       and      LV_ERR_VCC_SENS_SUB = 0
then(1)         V_PORT_TOL = V_PORT
       if(2)          V_PORT_TOL <= C_V_PORT_TOL_MAX
                and   V_PORT_TOL >= C_V_PORT_TOL_MIN
                      (check violation of  TOL limits)
       then(2)        LV_PORT_AD_TOL_VLD = 1
       else(2)        LV_PORT_AD_TOL_VLD = 0
                      LV_PORT_AD_ERR = 1
       endif(2)
endif(1)
```

At the end of Powerlatch:

Save the learned values

```
if(1)           LV_PORT_AD_BOL_VLD = 1
       and      LV_PORT_AD_TOL_VLD = 1
then(1)         V_PORT_AD_BOL = V_PORT_BOL
                V_PORT_AD_TOL = V_PORT_TOL
                LV_PORT_AD_VLD = 1
endif(1)
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_PORT_AD_TOL	1	0..3FFH	0..5	4.8875e-3	V
potentiometer voltage value for stop during adaptation of the top, default value: 4.5V					
C_V_PORT_AD_BOL	1	0..3FFH	0..5	4.8875e-3	V
potentiometer voltage value for stop during adaptation of the bottom, defautl value: 0.5V					
C_V_PORT_TOL_MIN	1	0..3FFH	0..5	4.8875e-3	V
minimum value for TOL to check violation of limits, default value: 4V					
C_V_PORT_TOL_MAX	1	0..3FFH	0..5	4.8875e-3	V
maximum value for TOL to check violation of limits, default value: 5V					
C_V_PORT_BOL_MIN	1	0..3FFH	0..5	4.8875e-3	V
minimum value for BOL to check violation of limits, default value: 0.1V					
C_V_PORT_BOL_MAX	1	0..3FFH	0..5	4.8875e-3	V
maximum value for BOL to check violation of limits, default value: 1V					
C_PORT_BOL	1	F000..1000H	-100...100	0.0244	°PORT
BOL-value which belongs to V_PORT_BOL , default value : 0°PORT					
C_PORT_TOL	1	F000H..1000H	-100...100	0.0244	°PORT
TOL-value which belongs to V_PORT_TOL , default value : 73 °PORT					

9.16 Overload protection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PORTPWM	O/V	8000H...7FFFH	-100...99.99	0.003	%
Pulse width modulation for port flap electric motor energization					
PORTPWM_TMP_1	-	8000H...7FFFH	-100...99.99	0.003	%
Temporary port flap pulse width modulation					
T_PORT_DLY1	V	0...FFFFH	0...1310.7	0.02	s
Timer for delay time before PORTPWM-limitation becomes active					
T_PORT_DLY2	O/V	0...FFFFH	0...1310.7	0.02	s
Timer for delay time before PORTPWM-limitation becomes inactive					
STATE_OVL_PORT	O/V	0H 1H 2H	Normal Wait wo. Limit. Limitation	-	-
States of overload protection					
VB_MMV_PORT	V	0...FFH	0...26	0.1	V
Battery voltage moving mean value					
T_PORT_DIAG_EL	-	0...FFH	0...2.55	0.01	sec
Timer for applying PORTPWM for Powerstage Diag					

Input data:

PORTPWM_TMP	VB	TCO	LV_ERR_PORT_EL
LV_ERR_PORT	LV_ES		

General information:

In order to avoid the port flap electric motor overload the timer T_PORT_DLY1 starts if the critical pulse width modulation C_PORTPWM_OVL_LIM is exceed. If the timer T_PORT_DLY1 reaches its limit C_T_PORT_DLY1 the pulse width modulation is reduced to the value C_PORTPWM_DFT and the timer is reset.

The calibratable delay time C_T_PORT_DLY2 defines the duration of the PWM on C_PORTPWM_DFT. If PORTPWM_TMP undershoots C_PORTPWM_OVL_LIM before the delay C_T_PORT_DLY2 expires, then PORTPWM_TMP is the output signal. The count down of the delay C_T_PORT_DLY2 continues.

To perform the Powerstage diagnosis at least a Minimum PWM is necessary. Therefore if there is no request to move the Port Flap (PORTPWM_TMP = 0), then a PWM switch pattern is applied.

Application conditions:

Initialization: At reset: VB_MMV_PORT = 12 V

T_PORT_DIAG_EL = C_T_PORT_DIAG_EL

Reccurence: 10 ms

Activation: At every engine state

Overview:

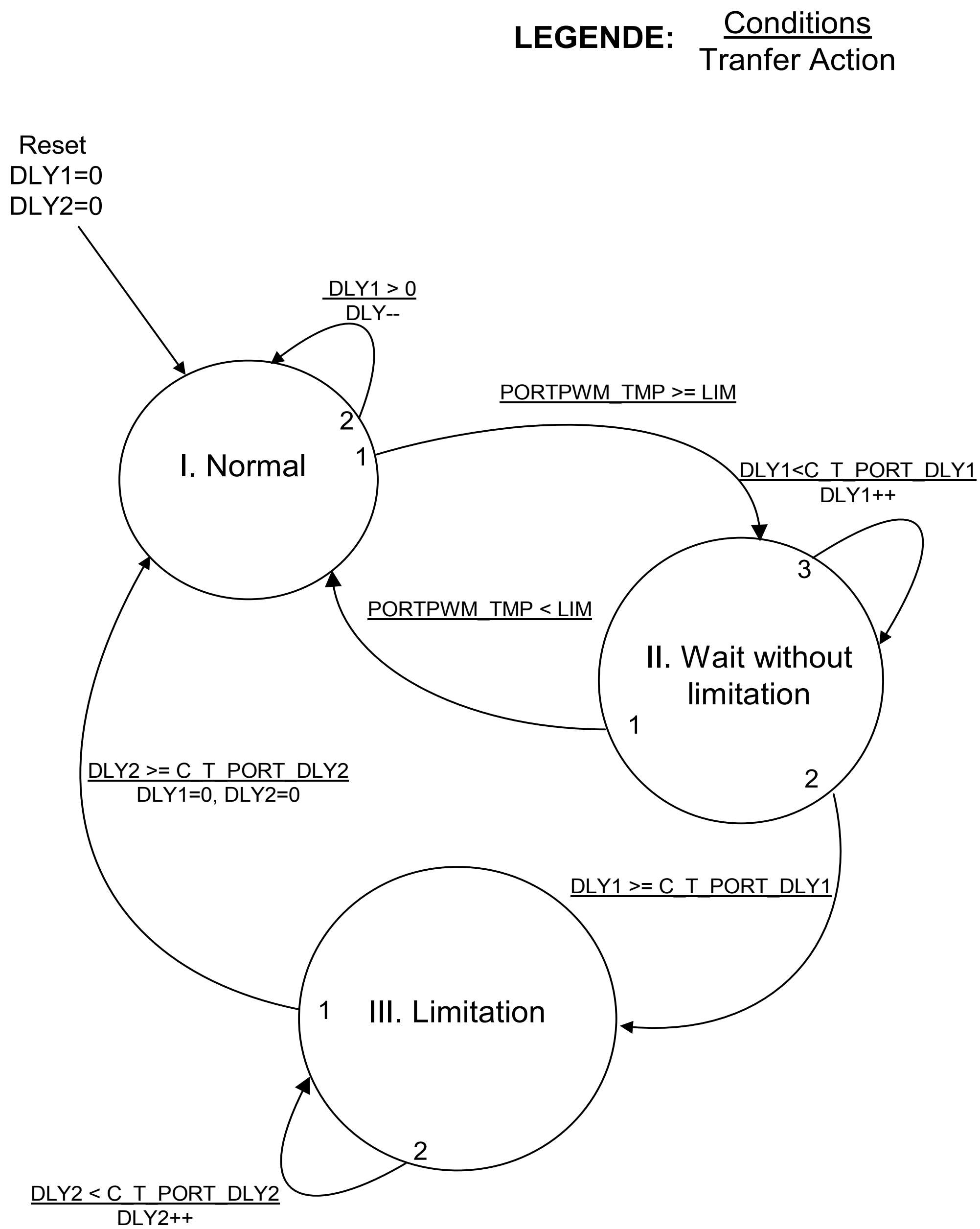


Figure: Overload protection state diagram

Formula section:

STATE I) Normal:

Inside action:

```
if        T_PORT_DLY1 > 0
then      T_PORT_DLY1(n) = T PORT DLY1 (n-1) -1
endif
STATE_OVL_PORT = 0

If        PORTPWM_TMP = 0%
    and   LV_ES = 0
then   if        T_PORT_DIAG_EL > 0
       then      T_PORT_DIAG_EL(n) = T_PORT_DIAG_EL(n-1) -1h
          if          T_PORT_DIAG_EL > 0.5 * C_T_PORT_DIAG_EL
          then        PORTPWM_TMP_1 =  C_PORTPWM_DIAG_EL
          else        PORTPWM_TMP_1 =  C_PORTPWM_DIAG_EL * (-1)
          endif
       else      T_PORT_DIAG_EL = C_T_PORT_DIAG_EL
       endif
else      PORTPWM_TMP_1 = PORTPWM_TMP
endif
```

Transfer to II

```
if        |PORTPWM_TMP| >= C_PORTPWM_OVL_LIM
then      goto STATE II
```

STATE II) Wait without limitation:

Inside action:

```
T_PORT_DLY1(n) = T_PORT_DLY1(n-1) + 1
STATE_OVL_PORT = 1
PORTPWM_TMP_1 = PORTPWM_TMP
```

Transfer to I

```
if        |PORTPWM_TMP| < C_PORTPWM_OVL_LIM
then      goto STATE I
```

Transfer to III

```
if        T_PORT_DLY1 >= C_T_PORT_DLY1
then      goto STATE III
```

III) Limitation:

Inside action:

T_PORT_DLY2(n) = T_PORT_DLY2 (n-1)+1

if |PORTPWM_TMP| >= C_PORTPWM_OVL_LIM
then **if** PORTPWM_TMP >= 0
then PORTPWM_TMP_1 = C_PORTPWM_DFT
else PORTPWM_TMP_1 = C_PORTPWM_DFT*(-1)
endif
else PORTPWM_TMP_1 = PORTPWM_TMP
endif

STATE_OVL_PORT = 2

Transfer to I

if T_PORT_DLY2 >= C_ T_PORT_DLY2
then T_PORT_DLY1 = 0
T_PORT_DLY2 = 0
endif

VB_MMV_PORT = $VB\_MMV\_PORT_{n-1}$
+ C_CRLC_VB_MMV_PORT * (VB - $VB\_MMV\_PORT_{n-1}$)

if LV_ERR_PORT = 0
then PORTPWM = C_PORTPWM_AS
+ (PORTPWM_TMP_1 * IP_FAC_VB_PORTPWM * IP_FAC_TCO_PORTPWM)
else **if** LV_ERR_PORT_EL = 1
then PORTPWM = 0%
in case of electrical failure no Controller reaction is requested
else PORTPWM = PORTPWM_TMP
in case of other Port failure Controller reaction is not limited by overload protection
endif
endif

Limit PORTPWM to -100...99.99%

Calibration data:

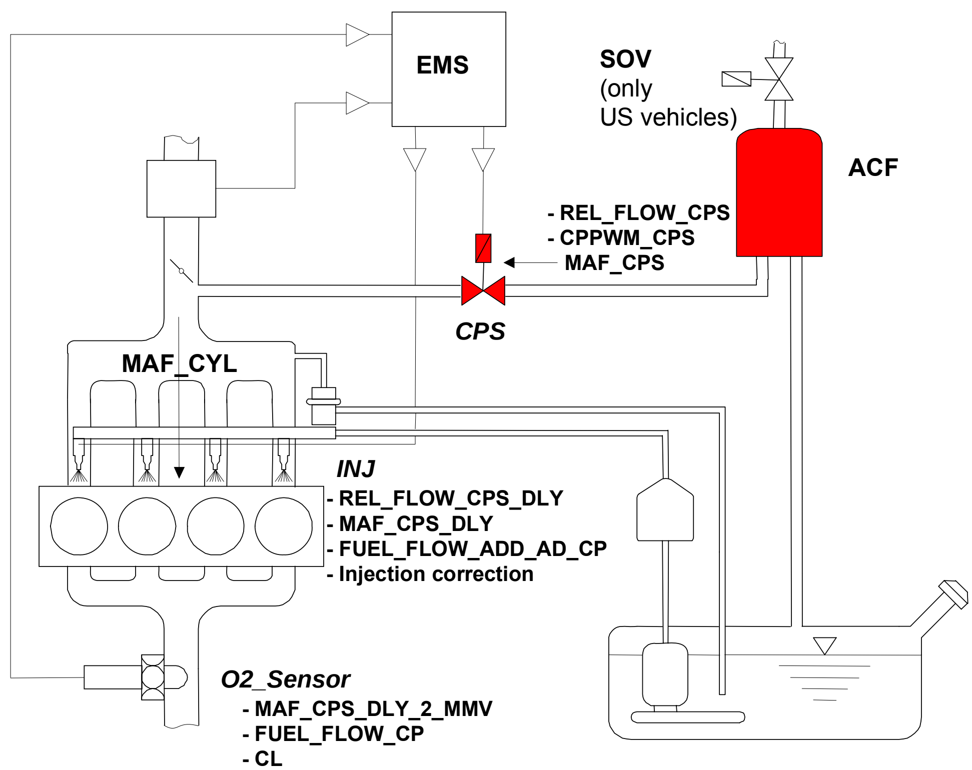
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PORTPWM_OVL_LIM	1	0...7FFFH	0...99.99	0.003	%
Overload protection limitation of pulse width modulation for port flap electric motor					
C_T_PORT_DLY1	1	0...FFFFH	0...1310.7	0.02	s
Delay time before PORTPWM-limitation becomes active					
C_T_PORT_DLY2	1	0...FFFFH	0...1310.7	0.02	s
Delay time before PORTPWM-limitation becomes inactive					
C_PORTPWM_DFT	1	0...7FFFH	0...99.99	0.003	%
Overload protection reduced pulse width modulation for port flap electric motor					
C_PORTPWM_AS	1	8000H...7FFFH	-100...99.99	0.003	%
Additional pulse width modulation for port flap by application					
C_CRLC_VB_MMV_PORT	1	0...FFH	0.996	0.0039	-
Correlation constant for battery voltage filtering					
IP_FAC_VB_PORTPWM	1x8	0...FFH	0...1.999	0.0078	-
LDP_VB_MMV_PORT_IP_FAC_VB	8	0...FFH	0...26	0.1	V
Correction factor for port flap pulse width modulation depending on battery voltage					
IP_FAC_TCO_PORTPWM	1x8	0...FFH	0...1.999	0.0078	-
LDP_TCO_IP_FAC_TCO_PORTPWM	8	0...FEH	-48...142.5	0.75	°C
Correction factor for port flap pulse width modulation depending on coolant temperature					
C_T_PORT_DIAG_EL	1	0...FFH	0...2.55	0.01	sec
Recurrence time for switch pattern PWM for Powerstage diagnosis					
C_PORTPWM_DIAG_EL	1	0...7FFFH	0...99.99	0.003	%
Minimum PWM ratio for Powerstage diagnosis					

9.17 Evaporative Emission Control

General information:

The evaporative emissions of the fuel tank system are stored in a activated charcoal filter ACF to prevent an escape of the gas to the environment. The loading capacity of the filter is limited. So the filter must be purged temporarily. Therefore the canister purge solenoid CPS, that is positioned between the charcoal filter and the intake manifold is opened by the engine management system. To prevent driveability and emission problems, the opening of the CPS and so the purge flow FUEL_FLOW_CP must be controlled.

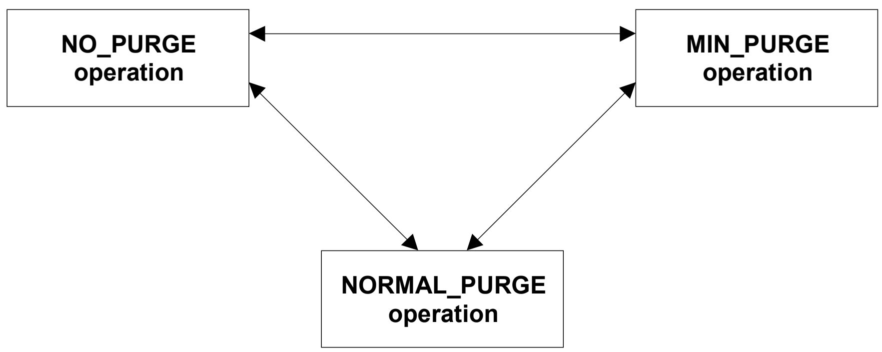
With the Evaporative Emission Control function the opening of the CPS is controlled in dependence on the fuel mass stored in the canister. Therefore the canister load CL is calculated using the measured mass air flow MAF, the deviation of the lambda-control when the CPS is open, the mass flow FLOW_CPS through the CPS and the stoichiometric constant of the fuel C_FUEL_FAC_CP. The opening period CPPWM of the CPS is controlled in dependence on the canister load CL.



general specification

<u>The Evaporative Emission Control consists of the following partial functions:</u>

NO_PURGE	no purging of the activated charcoal filter (ACF)
MIN_PURGE	purging of the activated charcoal filter (ACF) with unknown loading of the ACF, purging with low mass flow through the canister purge solenoid
NORMAL_PURGE	purging in dependence on the canister load CL of the activated charcoal filter (ACF), the MAX_PURGE consists of the phases RAMP_OPEN, MAX_PURGE and RAMP_CLOSE



During enabled Evaporative Emission Control (LV_CP_ENA = 1) one of the three partial function is activated in dependence of the engine operating conditions.

<u>Application conditions:</u>

Activation: LV_CP_ENA = 1

Recurrence: 100ms

9.17.1 Activation of the Evaporative Emission Control Function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CP_ENA	V/O	0...1H	0...1	1	-
Logical value for enabled EVAP-Control					

Input data:

LV_ES	LV_ST	LV_AST	TCO
T_AST_DC	TCO_ST	CONF_CP	EOL_STATE_DIAGCP

FUNCTION DESCRIPTION:

General information:

The Evaporative Emission Control function is enabled (LV_CP_ENA=1), if the coolant temperature is higher than a threshold value C_TCO_MIN_CP and the time after start is greater than a threshold ID_T_AST_CP__TCO_ST. The function is active during all engine operating states except the following states:

engine stop, engine start, post-start function function active.

Formula section:

```
if          TCO ≥ C_TCO_MIN_CP
     and    LV_ES = 0
     and    LV_ST = 0
     and    LV_AST = 0
     and    CONF_CP > 0
then        if      EOL_STATE_DIAGCP <> EOL_PAS
            then    LV_CP_ENA = 1
            else    if      T_AST_DC > ID_T_AST_CP__TCO_ST
                    then    LV_CP_ENA = 1
                    else    LV_CP_ENA = 0
                            CP_STATE = CP_NOT_ACTIVE (0 Hex.)
                            CP_T_STATE = CP_T_NOT_ACTIVE (0 Hex.)
                            LV_CP_ACT = 0
                    endif
            endif
else        LV_CP_ENA = 0
            CP_STATE = CP_NOT_ACTIVE (0 Hex.)
            CP_T_STATE = CP_T_NOT_ACTIVE (0 Hex.)
            LV_CP_ACT = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_CP	1	0...FEH	-48...142.5	0.75	°C
Minimum temperature for EVAP-Control enabled					
ID_T_AST_CP__TCO_ST	1*4	0...FFFFH	0...6553.5	0.1	sec
LDP_TCO_ST_ID_T_AST_CP	4	0...FEH	-48...142.5	0.75	°C
Time threshold for EVAP-Control enabled					

9.17.2 Time control of the Evaporative Emission Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CP_T_STATE	V	0H...0H 1H...1H 2H...2H 3H...3H	CP_T_NOT_ACTIVE T_CP_ST_ACTIVE T_CP_ACTIVE T_TI_AD_ACTIVE	-	-
State of Evaporative Emission Control Function					
T_CP	V	0H...FFFFH	0...6553.5	0.1	sec
Accumulated time of actual CP time range					
T_CP_AD_1	V	0H...FFFFH	0...6553.5	0.1	sec
Accumulated time of total TI-adaptation duration					

Input data:

CL_MMV	C_CL_MIN_CP	LV_CP_ENA	T_MIN_CP
T_CP_TI_AD_SP			

FUNCTION DESCRIPTION:

General information:

The engine operation is divided into 3 time ranges. After transition from the engine operating state start (LV_ST 1->0) the time period CP_T_1 is active (CP_T_STATE=1 (Hex.)) if the Evaporative Emission Control is enabled (LV_CP_ENA=1). During this period the Evaporative Emission Control is possible if the conditions for the function are fulfilled. This period ends if the time T_CP exceeds the maximum time C_T_1_CP or the canister load CL_MMV is lower than a threshold value C_CL_MIN_CP and the time T_CP was greater than T_MIN_CP (see chapter “Partial Functions of the Evaporative Emission Control”). After that period there is a continuous alternation of the time periods TI_T_AD and CP_T_2. First the lambda adaptation is activated if the conditions are fulfilled. This phase is active for a time T_CP_AD_1 (CP_T_STATE=3 (Hex.)). When the time T_CP is greater than T_CP_AD_1 the period CP_T_2 (CP_T_STATE=2 (Hex.)) is active and the Evaporative Emission Control is possible if the conditions are fulfilled. This period ends if the time T_CP exceeds the maximum time C_T_2_CP or the canister load CL_MMV is lower than a threshold value C_CL_MIN_CP and the time T_CP was greater than T_MIN_CP (see chapter “Partial Functions of the Evaporative Emission Control”).

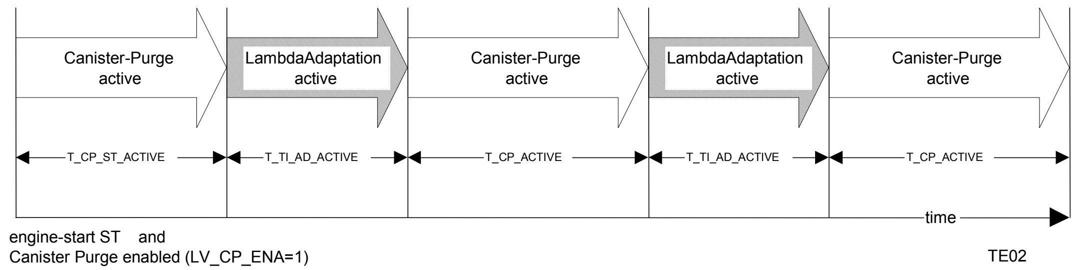
The time period for lambda adaptation is variable due to the stop conditions for the purge period. If the purge period is stopped because the time T_CP has reached the limit C_T_1/2_CP the time period T_CP_AD_1= T_CP_TI_AD_SP. If the purge period is stopped because the canister load CL_MMV is lower than a threshold value C_CL_MIN_CP the time period T_CP_AD_1= T_CP_TI_AD_SP +C_T_1/2_CP-T_CP (actual time at the end of the purge period).

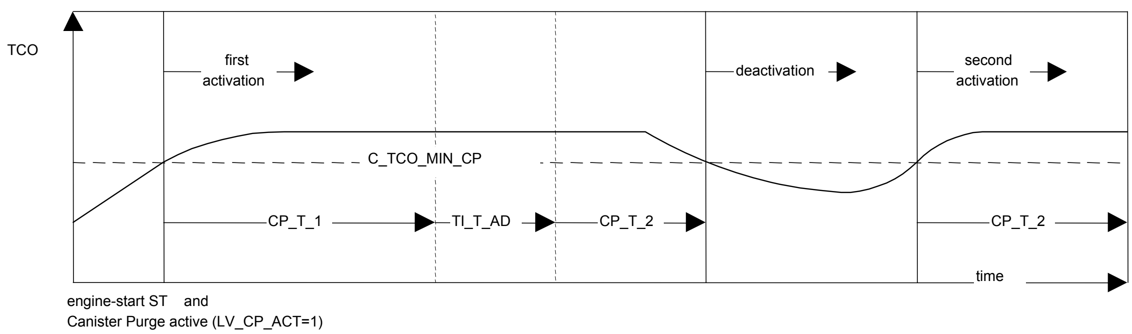
The time T_CP is the actual time state of the Evaporative Emission Control.

If the Evaporative Emission Control is activated after engine start ST, the lambda adaptation is activated first (CP_T_STATE=3 (Hex.)).

When the Evaporative Emission Control is not enabled (LV_CP_ENA=0), the time control is not active (CP_T_STATE=0) and the lambda adaptation is possible, if the conditions are fulfilled (see chapters “Lambda Adaptation and Evaporative Emission Control” and “Injection”).

Signal flow diagram:





Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_1_CP	1	0H...FFFFH	0...6553.5	0.1	sec
Duration of first purge phase after engine start					
C_T_2_CP	1	0H...FFFFH	0...6553.5	0.1	sec
Duration of purge phase					

9.17.3 Partial Functions of the Evaporative Emission Control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CP_STATE	V/O	0H...0H 1H...1H 2H...2H 3H...3H 4H...4H 5H...5H 6H...6H	CP_NOT_ACTIVE NO_PURGE MIN_PURGE RAMP_OPEN MAX_PURGE RAMP_CLOSE WAIT_RAMP_OPEN	-	-
State of Evaporative Emission Control Function					
CP_TRA_STATE	V/O	0H...0H 1H...1H 2H...2H 3H...3H 4H...4H 5H...5H 6H...6H 7H...7H 8H...8H 9H...9H AH...AH BH...BH CH...CH	NO_TRA NO_TO_MIN_TRA MIN_TO_NO_TRA OPEN_TO_NO_TRA OPEN_TO_MIN_TRA MAX_TO_NO_TRA MAX_TO_MIN_TRA CLOSE_TO_NO_TRA CLOSE_TO_MIN_TRA OPEN_TO_CLOSE_TRA MAX_TO_CLOSE_TRA WAIT_TO_OPEN_TRA WAIT_TO_MIN_TRA	-	-
State of transitions between partial functions of the Evaporative Emission Control					
CP_NORM_DEAC_STATE	V	0H...0H 1H...1H 2H...2H	NO_DEAC RAMP_OPEN_DEAC MAX_DEAC	-	-
Deactivation states of NORMAL_PURGE					
LV_CP_ACT	V/O	0...1H	0...1	1	-
Logical value for EVAP-Control in MIN_PURGE or RAMP_OPEN or MAX_PURGE or RAMP_CLOSE state					
LV_MAX_PURGE_END	V	0H...1H	0...1	1	-
Logical value for end of MAX_PURGE operation					
LV_NORM_PURGE_END_1	V/O	0H...1H	0...1	1	1
Logical value for first end of NORMAL_PURGE operation during driving cycle					
LV_RAMP_CLOSE_END	V/O	0H...1H	0...1	1	-
Logical value for end of RAMP_CLOSE operation					
LV_RAMP_OPEN_END	V/O	0H...1H	0...1	1	-
Logical value for end of RAMP_OPEN operation					
LV_TI_LAM_COR_CP	V/O	0H...1H	0...1	1	-
Logical value for TI_LAM correction during RAMP_OPEN operation					
LV_WAIT_RAMP_OPEN_CP	V	0H...1H	0...1	1	-
Logical value for WAIT_RAMP_OPEN_PURGE state active					
CL	V	0H...FFFFH	0...8	0.000122	-
Canister load					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CL_DI	V	0H...FFFFH	0...8	0.000122	-
Canister load CL for disabled NORMAL_PURGE operation					
CL_MMV	V/O	0H...FFFFH	0...8	0.000122	-
Moving mean value of the canister load CL					
CL_MMV_DI	V	0H...FFFFH	0...8	0.000122	-
Moving mean value of the canister load CL for disabled NORMAL_PURGE operation					
CL_MMV_OLD	V/O	0H...FFFFH	0...8	0.000122	-
Canister load for injection time correction calculation					
CL_MMV_OLD_DI	V	0H...FFFFH	0...8	0.000122	-
Canister load for injection time correction calculation for disabled NORMAL_PURGE operation					
CL_MMV_NORM_PURGE_END	V/O	0H...FFFFH	0...8	0.000122	-
Canister load at the end of NORMAL_PURGE_MAX_PURGE operation					
FUEL_FLOW_CP	V	8000H...7FFFH	-8...8	0.000244	kg/h
Fuel flow from the ACF					
LAM_DIF_CP	V	8000...7FFFH	-50...50	0.0015	%
Lambda deviation with active canister purge					
LAM_0_CP	V	8000H...7FFFH	-50...50	0.0015	%
Start value of the lambda control injection time correction TI_LAM before the CPS is opened					
LAM_0_CP_DI	V	8000H...7FFFH	-50...50	0.0015	%
Start value of the lambda control injection time correction TI_LAM for disabled NORMAL_PURGE operation					
LAM_P_SUM_CP	V	0H...FFH	0...255	1	-
Saved actual number of P-jumps of the lambda controller					
MAF_CYL_DLY	V	0H...FFFFH	0...2047.97	0.03125	kg/h
Filtered value of MAF_FG_CYL (Padé Filter)					
MAF_CYL_DLY_OLD	V	0H...FFFFH	0...2047.97	0.03125	kg/h
Old value of MAF_CYL_DLY for Padé Filter calculation for MAF_CYL_DLY					
MAF_CYL_OLD	V	0H...FFFFH	0...2047.97	0.03125	kg/h
$MAF\_FG\_CYL_{(n-1)}$ for Padé Filter calculation for MAF_CYL_DLY					
MAF_CYL_OLD2	V	0H...FFFFH	0...2047.97	0.03125	kg/h
$MAF\_FG\_CYL_{(n-2)}$ for Padé Filter calculation for MAF_CYL_DLY					
MAF_DLY_MMV	V/O	0H...FFFFH	0...2047.97	0.03125	kg/h
Moving mean value of MAF_CYL_DLY					
REL_FLOW_CPS	V	0H...FFFFH	0...1	0.000015	-
Setpoint value for the relative flow (MAF_CPS/MAF_FG_CYL) through the CPS					
REL_FLOW_CPS_DI	V	0H...FFFFH	0...1	0.000015	-
Relative flow (MAF_CPS/MAF_FG_CYL) through the CPS for disabled NORMAL_PURGE operation					
REL_FLOW_CPS_OLD	V	0H...FFFFH	0...1	0.000015	-
Relative flow (MAF_CPS/MAF_FG_CYL) for injection time correction					
REL_FLOW_CPS_OLD_MIN	V/O	0H...FFFFH	0...1	0.000015	-
Relative flow (MAF_CPS/MAF_FG_CYL) for injection time correction					
REL_FLOW_CPS_OLD_DI	V	0H...FFFFH	0...1	0.000015	-
Relative flow (MAF_CPS/MAF_FG_CYL) for injection time correction for disabled NORMAL_PURGE operation					
REL_FLOW_MAX_CPS	O	0H...FFFFH	0...1	0.000015	-
Current value from map IP_REL_FLOW_MAX_CPS__N_32__MAF					
T_DI_MAX_CP	V	0H...FFFFH	0...6553.5	0.1	sec
Time counter for disabled MAX_PURGE					
T_DI_RAMP_OPEN_CP	V	0H...FFFFH	0...6553.5	0.1	sec
Time counter for disabled RAMP_OPEN					
T_WAIT_RAMP_OPEN	V	0H...FFFFH	0...6553.5	0.1	sec
Time counter for WAIT_RAMP_OPEN_PURGE state					
TI_LAM_CP	V/O	8000H...7FFFH	-50...50	0.0015	%
Correction value for TI_LAM correction during RAMP_OPEN operation					
TI_LAM_CP_MMV	V	8000H...7FFFH	-50...50	0.0015	%
Moving mean value of the lambda control injection time correction TI_LAM					
REL_FLOW_CPS_TMP_MAX	V	0H...FFFFH	0...1	0.000015	-
Temporary setpoint value for the relative flow (MAF_CPS/MAF_FG_CYL) through the CPS for purge slow opening after EVAP diagnosis abortion					
REL_FLOW_CPS_TMP_INC	V	0H...FFFFH	0...1	0.000015	-
Temporary setpoint increment value for the relative flow for purge slow opening after EVAP abortion					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
REL_FLOW_CPS_INC	-	0H...FFFFH	0...1	0.000015	-
Setpoint increment value for the relative flow through the CPS for purge slow opening after EVAP abortion					
LV_MAX_CP_TRA_DIAGCP	V	0...1H	0...1	-	-
Logical value to activate purge slow opening after EVAP diagnosis abortion					
LV_CP_MAX_ST_DIAGCP	V	0...1H	0...1	-	-
Logical value to check MAX_PURGE at the start of EVAP leakage monitoring					
T_DLY_4_DIAGCP	1	0H...FFFFH	0...6553.5	0.1	sec
Purge slow opening activation time delay after EVAP diagnosis abortion					
LV_MAX_CP_END_DIAGCP	V	0...1H	0...1	-	-
Logical value to disable purge slow opening after EVAP diagnosis abortion					
LV_T_DLY_4_END_DIAGCP	V	0...1H	0...1	-	-
Logical value to disable T_DLY_4 calculations after EVAP diagnosis abortion					
T_CP_AD_1_OLD	V	0H...FFFFH	0...6553.5	0.1	sec
Saved accumulated time of total TI-adaptation duration					

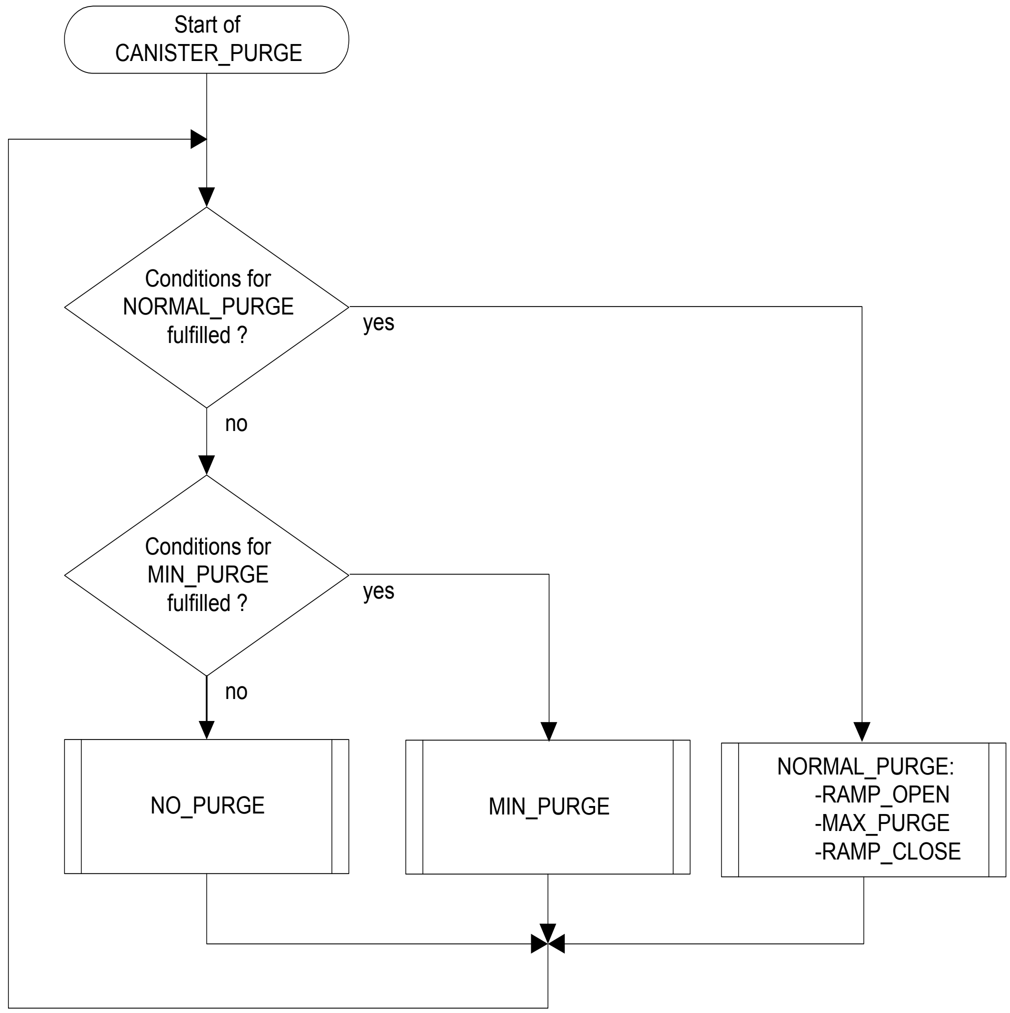
Input data:

N_32	TI_LAM_i	LV_CP_RAMP_OPEN_STOP	LV_LAM_LIM_MAX
LV_CP_ENA	LAM_MV_i	LV_LAM_LIM_MAX_CYCNR	LV_LAM_LIM_MIN
LV_LDC_CP	LV_LSCL_i	LV_OBD_AUTH_CP_CLOSE	LV_OBD_AUTH_CP_MIN
LV_FCUT_CP	LV_IS	LV_RST_NORM_PURGE_END	STATE_SAV_DIAG
MAF_FG_CYL	LV_CP_SET_CLOSE	FUEL_FLOW_ADD_AD_CP	REL_FLOW_CPS_DLY_MMV
MAF_CYL	LAM_P_CTR	FUEL_FLOW_ADD_AD_DLY	LV_ACT_DIAGCP
MAF	TIA	MAF_CPS_DLY_2_MMV	LV_ACT_DIAGCPS
PRS_CPS	T_PL_DIAGCP	LV_REL_FLOW_MIN_CPS_S_ACT	REL_FLOW_MIN_CPS_S
STATE_DIAGCP	DTP	TCO	TQ_DIF_I_IS
LV_ERR_TCO	LV_ERR_TIA	T_IS_AST	ACP
C_ACP_MAX_RLY_ACCOUT_OFF	LV_LAM_STOP[NC_CBK_EX_NR]	LV_LAM_NOT_STAT_CDN	CL_MMV_CLC_END
T_CP_AD_1			

General information:

After activation of the function there is a continuous check, which partial function is active.

Signal flow diagram:



- ***Initializition of the Function***

Initialization at Reset:

CP_STATE =CP_NOT_ACTIVE (0 Hex.)

CP_TRA_STATE =NO_TRA (0 Hex.)

CP_NORM_DEAC_STATE =NO_DEAC (0 Hex.)

CP_T_STATE =CP_T_NOT_ACTIVE (0 Hex.)

Initialization at Engine Stop -> Engine Run:

LV_NORM_PURGE_END_1=0

LV_CP_MAX_ST_DIAGCP = 0

LV_MAX_CP_END_DIAGCP = 0

LV_T_DLY_4_END_DIAGCP = 0

T_DLY_4_DIAGCP = 0

Setting of Initilization values for RAMP_OPEN (see chapter RAMP_OPEN)

LV_CP_ACT =0

<u>For the different functions the following conditions must be fulfilled:</u>

- ***Conditions for NO_PURGE:***

1. Cylinder Shut-Off or Pull Fuel-Cut-Off active (LV_FCUT_CP=1) or
2. Authorization by projectspecific conditions fulfilled (LV_OBD_AUTH_CP_CLOSE=1) or
3. Lambda adaptation time period active (CP_T_STATE=3 (Hex.))
   and **{** MIN_PURGE active (CP_STATE=2 (Hex.) or
   NO_PURGE active (CP_STATE=1 (Hex.) or
   LV_RAMP_CLOSE_END=1 **}**
4. EVAP-system monitoring active (LV_ACT_DIAGCP=1 or LV_ACT_DIAGCPS=1) or
5. Lambda sensor diagnosis active (LV_CP_SET_CLOSE=1) or
6. Difference pressure environment-intake manifold PRS_CPS = AMP-MAP is lower than a threshold value C_PRS_MAX_1_CPS (supercharger engine) or
7. If NO_PURGE is active PRS_CPS = AMP-MAP is lower than a threshold value C_PRS_MAX_2_CPS (supercharger engine) (tuning: C_PRS_MAX_2_CPS>C_PRS_MAX_1_CPS) or
8. Idle speed active (LV_IS=1) and ambient air temperature TIA is lower than a threshold value C_TIA_MIN_CP (acoustic problems with the CPS at low ambient temperatures) or
9. **{**Injection time correction of lambda controller TI_LAM exceeds threshold values
   (TI_LAM > IP_TI_LAM_MAX_CP__TIA or TI_LAM < C_TI_LAM_MIN_CP)
   ->this check must ***only be done*** in case of secondary air system diagnosis
   STATE_SAV_DIAG=not active (= 0hex) and leak detection not active LV_ACT_DIAGCPS = 0**}** or
10. Lambda controller limited (LV_LAM_LIM_MAX_CYCNR=1) or

11. Maximum value of REL_FLOW_CPS is limited to 0
(IP_FUEL_FLOW_MAX_CP=0 or IP_REL_FLOW_MAX_CPS=0)
12. Lambda controller active with limitation
(LV_LAM_LIM_MAX=1 or LV_LAM_LIM_MIN=1)

- ***Conditions for MIN_PURGE:***

1. Conditions for NO_PURGE are not fulfilled and
2. Authorization by projectspecific conditions fulfilled or
(LV_OBD_AUTH_CP_MIN=1)
3. Lambda control in open-loop state (LV_LSCL_i = 0) or
Activation signal for the lambda controller stop mode is fulfilled
(LV_LAM_STOP = 1) or
Flag stop mode activation of lambda controller by wall film is setted
(LV_LAM_NOT_STAT_CDN = 1) or
4. Long time hot idle soaking and unstable engine conditions with
Engine idle state (LV_IS = 1) and
High intake air temperature (TIA > C_TIA_MAX_MIN_PURGE) and
High coolant temperature (TCO > C_TCO_MAX_MIN_PURGE) and
No temperature sensor failure (LV_ERR_TCO = 0 and LV_ERR_TIA = 0) and
Time limit for idle time after starting reached
(T_IS_AST > C_T_IS_AST_MIN_PURGE) and
High purge flow during unstable (REL_FLOW_CPS > C_REL_FLOW_CPS_MIN_PURGE) or
High torque difference during unstable (TQ_DIF_I_IS > C_TQ_DIF_I_IS_MIN_PURGE)
Then MIN_PURGE is activated and kept during C_T_MIN_CP_MIN_PURGE

- ***Conditions for NORMAL_PURGE:***

*Conditions for **RAMP_OPEN**:*

1. Conditions for NO_PURGE are not fulfilled and
2. Conditions for MIN_PURGE are not fulfilled and
3. Limited dynamics for speed and load (LV_LDC_CP=1) and
4. Engine speed within defined limits (C_N_CP_BOL<N_32<C_N_CP_TOL) and
5. Mass air flow within defined limits (IP_MAF_CP_BOL<MAF<IP_MAF_CP_TOL) and
6. Difference pressure environment-intake manifold PRS_CPS=AMP-MAP is and
greater than a threshold value C_PRS_MAX_3_CPS
7. Mean value of lambda control within threshold values and
(This check is only done once at start of RAMP_OPEN)
(C_LAM_0_CP_MIN < LAM_MV_i < IP_LAM_0_CP_MAX__TIA)
8. MAX_PURGE or RAMP_CLOSE not active or
9. RAMP_OPEN operation not finished (LV_RAMP_OPEN_END=0)

*Conditions for **MAX_PURGE**:*

1. Conditions for NO_PURGE are not fulfilled and
2. Conditions for MIN_PURGE are not fulfilled and
3. RAMP_OPEN operation normally finished (LV_RAMP_OPEN_END=1) or
   MAX_PURGE was disabled and disabling time was lower than
   a threshold value C_T_DI_MAX_CP (CP_NORM_DEAC_STATE=2 (Hex.))

*Conditions for **RAMP_CLOSE**:*

1. Time limit for RAMP_OPEN or MAX_PURGE operation reached or
   (T_CP>C_T_1_CP for first purge after start or T_CP>C_T_2_CP)
2. Canister load CL_MMV is lower than a threshold value and time limit
   for MAX_PURGE operation (T_CP> T_MIN_CP) is reached

If the one of the functions MIN_PURGE or RAMP_OPEN or MAX_PURGE or RAMP_CLOSE is active, the logical value LV_CP_ACT is set.

The transitions between the partial functions are defined in a separate chapter.

Formula section:

IF LV_CP_ENA=0
THEN CP_STATE=0 (Hex.) Evaporative emission control is not active

ELSEIF LV_FCUT_CP=1
or LV_OBD_AUTH_CP_CLOSE=1
or { CP_T_STATE=3(Hex.)
and (CP_STATE=2(Hex.)
or CP_STATE=1(Hex.)
or LV_RAMP_CLOSE_END=1) }
or LV_ACT_DIAGCP=1
or LV_ACT_DIAGCPS=1
or LV_CP_SET_CLOSE=1
or PRS_CPS<C_PRS_MAX_1_CPS
or (CP_STATE=1 (Hex.)
and PRS_CPS<C_PRS_MAX_2_CPS)
or (LV_IS=1 and TIA<C_TIA_MIN_CP)
or TI_LAM > IP_TI_LAM_MAX_CP__TIA
or TI_LAM < C_TI_LAM_MIN_CP
or LV_LAM_LIM_MAX_CYCNR=1
or IP_FUEL_FLOW_MAX_CP=0
or IP_REL_FLOW_MAX_CPS=0
or LV_LAM_LIM_MAX=1
or LV_LAM_LIM_MIN=1
THEN CP_STATE=1 (Hex.)
NO_PURGE activated

ELSE IF LV_OBD_AUTH_CP_MIN=1
or LV_LSCL_i = 0
or LV_LAM_STOP = 1
or LV_LAM_NOT_STAT_CDN = 1
THEN CP_STATE=2 (Hex.) **MIN_PURGE activated**
ENDIF
IF LV_IS = 1
and TIA > C_TIA_MAX_MIN_PURGE
and TCO > C_TCO_MAX_MIN_PURGE
and LV_ERR_TIA = 0

and LV_ERR_TCO = 0
and T_IS_AST > C_T_IS_AST_MIN_PURGE
and ACP > C_ACP_MAX_RLY_ACCOUT_OFF – C_ACP_OFS_MIN_PURGE
and (TQ_DIF_I_IS > C_TQ_DIF_I_IS_MIN_PURGE
or CL_MMV > C_CL_MMV_MIN_PURGE)
THEN CP_STATE = 2 (Hex.) **MIN_PURGE activated** *(hot soaking condition)*
more than C_T_MIN_CP_MIN_PURGE to allow stabilization
(the timer is decremented when the conditions are not fulfilled anymore)
ENDIF

ELSE IF LV_RAMP_CLOSE_END=0
and LV_MAX_PURGE_END=1
and LV_RAMP_OPEN_END=1
THEN CP_STATE=5 (Hex.)
RAMP_CLOSE activated
ELSEIF LV_RAMP_CLOSE_END=0
and LV_MAX_PURGE_END=0
and LV_RAMP_OPEN_END=1
THEN CP_STATE=4 (Hex.)
MAX_PURGE activated
ELSEIF LV_LDC_CP=1
and C_N_CP_BOL<N_32<C_N_CP_TOL
and IP_MAF_CP_BOL<MAF<IP_MAF_CP_TOL
and PRS_CPS>C_PRS_MAX_3_CPS
and CP_STATE≠4 (Hex.)
and CP_STATE≠5 (Hex.)
and C_LAM_0_CP_MIN < LAM_MV_i < IP_LAM_0_CP_MAX__TIA
(only checked once before RAMP_OPEN)
THEN CP_STATE=3 (Hex.)
RAMP_OPEN activated
ELSE CP_STATE=2 (Hex.)
MIN_PURGE activated
ENDIF
ENDIF

- ***Setting of LV_CP_ACT***

IF CP_STATE = MIN_PURGE (2Hex.)

or	CP_STATE = RAMP_OPEN	(3Hex.)
or	CP_STATE = MAX_PURGE	(4Hex.)
or	CP_STATE = RAMP_CLOSE	(5Hex.)
THEN	LV_CP_ACT=1	
ELSE	LV_CP_ACT=0	
ENDIF		

9.17.3.1 NO_PURGE Operation

General information:

In the NO_PURGE state of the Evaporative Emission Control no calculation of the relative CPS flow REL_FLOW_CPS, the fuel flow from the activated charcoal filter (ACF), the Canister Load CL and the additive adaptation value for the injection time correction will be done.

The transitions from and to the other partial functions are described in a separate chapter.

Formula section:

REL_FLOW_CPS=0

T_DLY_4_DIAGCP = 0

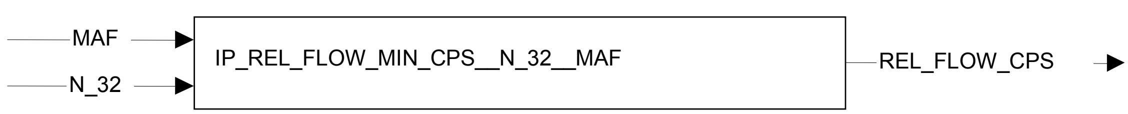
9.17.3.2 MIN_PURGE Operation

General information:

In the MIN_PURGE state the relative CPS flow is calculated from the map IP_REL_FLOW_CPS_MIN.

Signal flow diagram:

Calculation of REL_FLOW_CPS for MIN_PURGE:



(diagram above is valid for LV_REL_FLOW_MIN_CPS_S_ACT = 0)

The transitions from and to the other partial functions are described in a separate chapter.

Formula section:

if	LV_REL_FLOW_MIN_CPS_S_ACT = 0
then	REL_FLOW_CPS = IP_REL_FLOW_MIN_CPS__N_32__MAF

else REL_FLOW_CPS = REL_FLOW_MIN_CPS_S

endif

T_DLY_4_DIAGCP = 0

9.17.3.3 NORMAL_PURGE Operation

FUNCTION DESCRIPTION:

General information:

If the conditions for NORMAL_PURGE are fulfilled the NORMAL_PURGE operation is proceeded in the following sequence:

RAMP_OPEN -> MAX_PURGE -> RAMP_CLOSE

This sequence can change, if one of the phases is disabled for a short time and the interrupt is not longer than an applicable time constant C_T_DI_RAMP_OPEN_CP (CP_NORM_DEAC_STATE=1 (Hex.)) or C_T_DI_MAX_CP (CP_NORM_DEAC_STATE=2 (Hex.)).

Formula section:

REL_FLOW_MAX_CPS = IP_REL_FLOW_MAX_CPS__N_32__MAF

9.17.3.3.1 RAMP_OPEN Operation

FUNCTION DESCRIPTION:

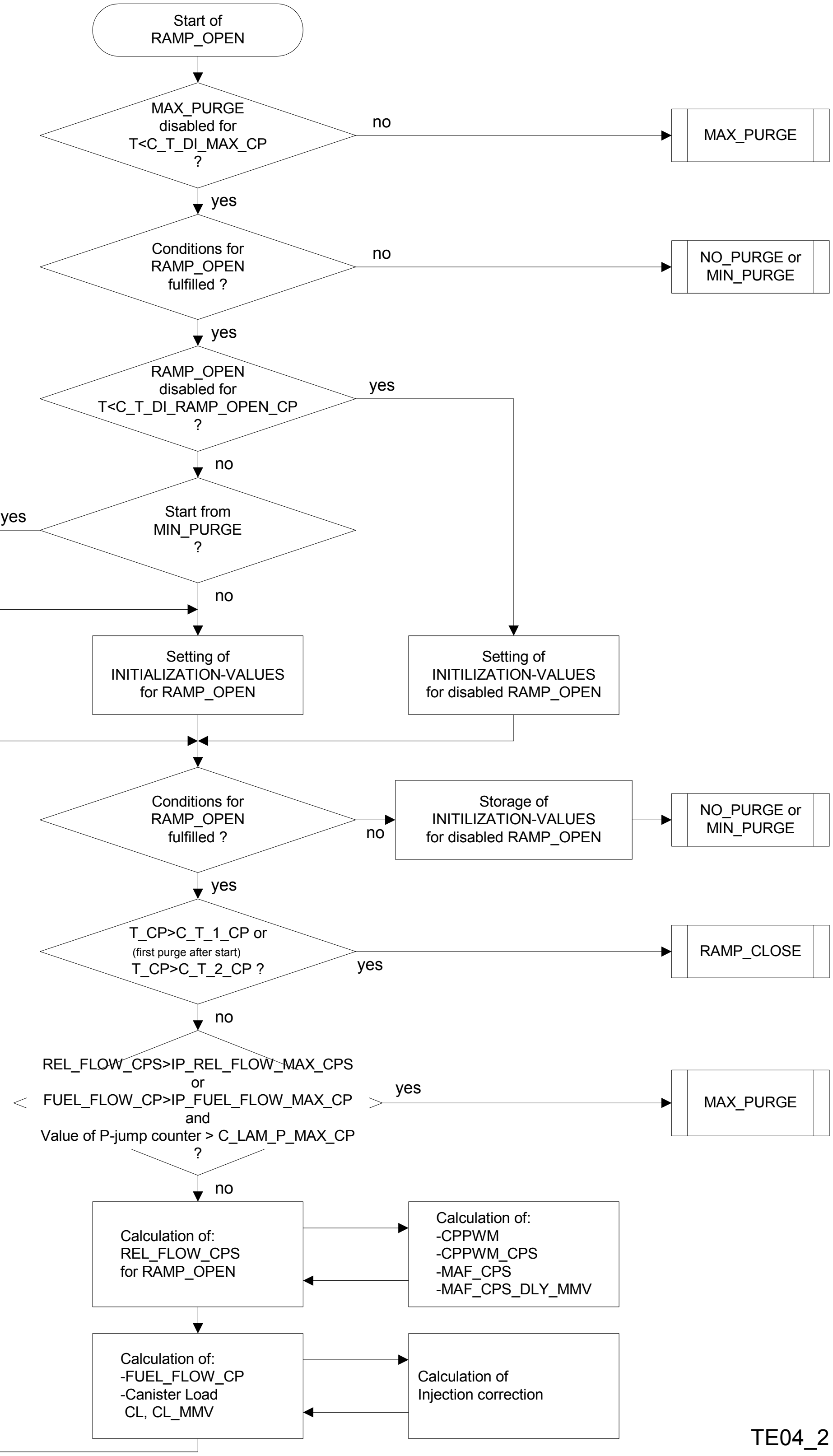
General information:

At the start of the NORMAL_PURGE the load CL of the ACF is not known. Therefore the CPS is opened slowly using a ramp function. So a high deviation of the lambda control shall be prevented. Normally RAMP_OPEN starts from NO_PURGE. If NO_PURGE time (T_CP_AD_1) is long enough, software can start to calculate CL_MMV with initialization data (C_CL_ST_CP). But if NO_PURGE time is shorter than threshold (C_T_CL_ST_CP), software may use the latest CL_MMV (CL_MMV_CLC_END) as its initialization because the stored fuel gas of canister does not change much.

The RAMP_OPEN operation is finished, when the relative flow through the CPS REL_FLOW_CPS is greater than a threshold value IP_REL_FLOW_MAX_CPS or the fuel flow FUEL_FLOW_CP is greater than a maximum map-value IP_FUEL_FLOW_MAX_CP. Then the relative opening of the CPS REL_FLOW_CPS is hold constant at the last value and a counter of P-jumps of the lambda controller starts. When the number of P-jumps is higher than a threshold C_LAM_P_CTR_MAX_CP than the RAMP_OPEN operation ends. Then the logical variable LV_RAMP_OPEN_END is set to 1 and the MAX_PURGE operation starts. If the time T_CP since the start of the RAMP_OPEN operation is greater than the constant C_T_1_CP (for the first purge after start) or C_T_2_CP, the RAMP_OPEN operation ends. Then the logical variables LV_RAMP_OPEN_END and LV_MAX_PURGE_END are set to 1 and the RAMP_CLOSE operation starts immediately.

Signal flow diagram:

RAMP_OPEN



TE04_2

- ***INITIALIZATION-VALUES for RAMP_OPEN***

Before the CPS is opened the following INITIALIZATION-VALUES for RAMP_OPEN are set:

CL_MMV_OLD=0
CL_MMV_DI=0
FUEL_FLOW_CP=0
LAM_0_CP=LAM_MV_i
LV_MAX_PURGE_END=0
LV_RAMP_CLOSE_END=0
LV_RAMP_OPEN_END=0
LV_TI_LAM_COR_CP=0
LV_WAIT_RAMP_OPEN=0
TI_LAM_CP=0
LAM_P_SUM_CP =0
LV_P_JUMP_ACT_CP=0
if T_CP_AD_1_OLD >= C_T_CL_ST_CP
or T_CP_AD_1_OLD = 0 (first RAMP_OPEN)
then CL = C_CL_ST_CP
CL_MMV = C_CL_ST_CP
else CL = CL_MMV_CLC_END
CL_MMV = CL_MMV_CLC_END
Endif

MAF_FG_CYL(n-1)=MAF_FG_CYL
MAF_FG_CYL(n-2)=MAF_FG_CYL
MAF_CYL_DLY=0
MAF_CYL_DLY(n-2)=MAF_FG_CYL
MAF_DLY_MMV=0
REL_FLOW_CPS=0
REL_FLOW_CPS_OLD=0
REL_FLOW_CPS_OLD_MIN=0
T_DI_MAX_CP=0
T_DI_RAMP_OPEN_CP=0
T_WAIT_RAMP_OPEN=0
LV_NORM_PURGE_END_1 = 0
T_DLY_4_DIAGCP = 0

MAF_FG_CYL(n-1) and MAF_FG_CYL(n-2) are internal values for PADÉ-Filter not MAF_FG_CYL from IMM.

- ***INITIALIZATION-VALUES for disabled RAMP_OPEN***

If the RAMP_OPEN operation was disabled (Conditions for RAMP_OPEN are not fulfilled) for a time t<C_T_DI_RAMP_OPEN_CP (CP_NORM_DEAC_STATE=1 (Hex.)), the following INITIALIZATION-VALUES for disabled RAMP_OPEN are set:

CL=CL_DI
CL_MMV=CL_MMV_DI
CL_MMV_OLD=CL_MMV_OLD_DI
REL_FLOW_CPS=REL_FLOW_CPS_DI
REL_FLOW_CPS_OLD=REL_FLOW_CPS_OLD_DI
REL_FLOW_CPS_OLD_MIN=REL_FLOW_CPS_OLD
LAM_0_CP=LAM_0_CP_DI
MAF_CYL_DLY_OLD=MAF_FG_CYL
MAF_DLY_MMV=0
T_DLY_4_DIAGCP = 0

If the RAMP_OPEN operation is disabled, the following actual values for CL_DI, CL_MMV_DI, CL_MMV_OLD_DI, REL_FLOW_CPS_DI, REL_FLOW_CPS_OLD_DI and LAM_0_CP_DI before the operation is disabled are stored:

CL_DI=CL
CL_MMV_DI=CL_MMV
CL_MMV_OLD_DI=CL_MMV_OLD
REL_FLOW_CPS_DI=REL_FLOW_CPS
REL_FLOW_CPS_OLD_DI=REL_FLOW_CPS_OLD

LAM_0_CP_DI=LAM_0_CP

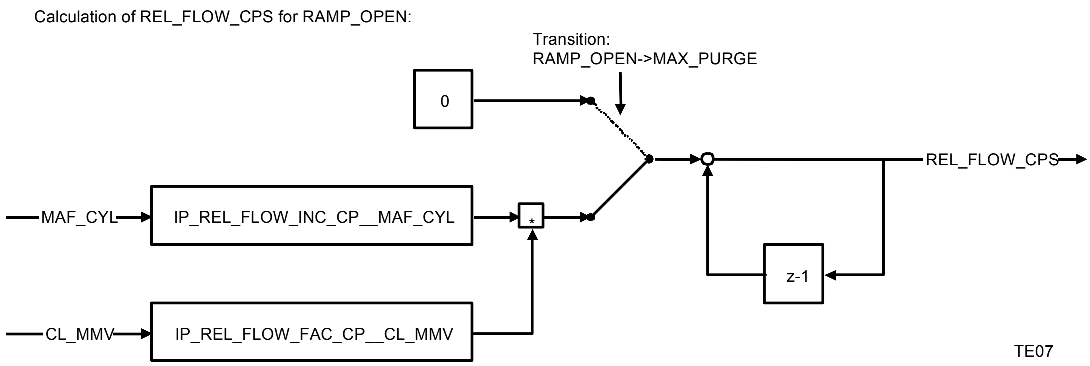
If the disabling time is longer than C_T_DI_RAMP_OPEN_CP or RAMP_OPEN was disabled because of too high deviations of the lambda controller (TI_LAM<C_TI_LAM_MIN_CP or TI_LAM>IP_TI_LAM_MAX_CP) then RAMP_OPEN starts completely new.

- ***Calculation of REL_FLOW_CPS for RAMP_OPEN***

The opening of the CPS is controlled by the value of REL_FLOW_CPS. This value describes the ratio of the flow through the CPS to the mass air flow MAF_FG_CYL. So a too high deviation of the lambda control at transient engine operation can be avoided. When the RAMP_OPEN operation is activated this value is incremented from the INITIALIZATION-VALUES. The increment value depends on the product of the map values of IP_REL_FLOW_INC_CP and IP_REL_FLOW_FAC_CP__CL_MMV. The opening speed of the CPS is so controlled by the mass air flow MAF_CYL and the load CL of the activated charcoal canister ACF.

When the relative flow through the CPS REL_FLOW_CPS is greater than a threshold value IP_REL_FLOW_MAX_CPS or the fuel flow FUEL_FLOW_CP is greater than a maximum map-value IP_FUEL_FLOW_MAX_CP, the relative opening of the CPS REL_FLOW_CPS is hold constant at the last value and a counter of P-jumps of the lambda controller starts. When the number of P-jumps is higher than a threshold C_LAM_P_CTR_MAX_CP than the RAMP_OPEN operation ends.

Signal flow diagram:



- ***Calculation of FUEL_FLOW_CP and Canister Load CL, CL_MMV***

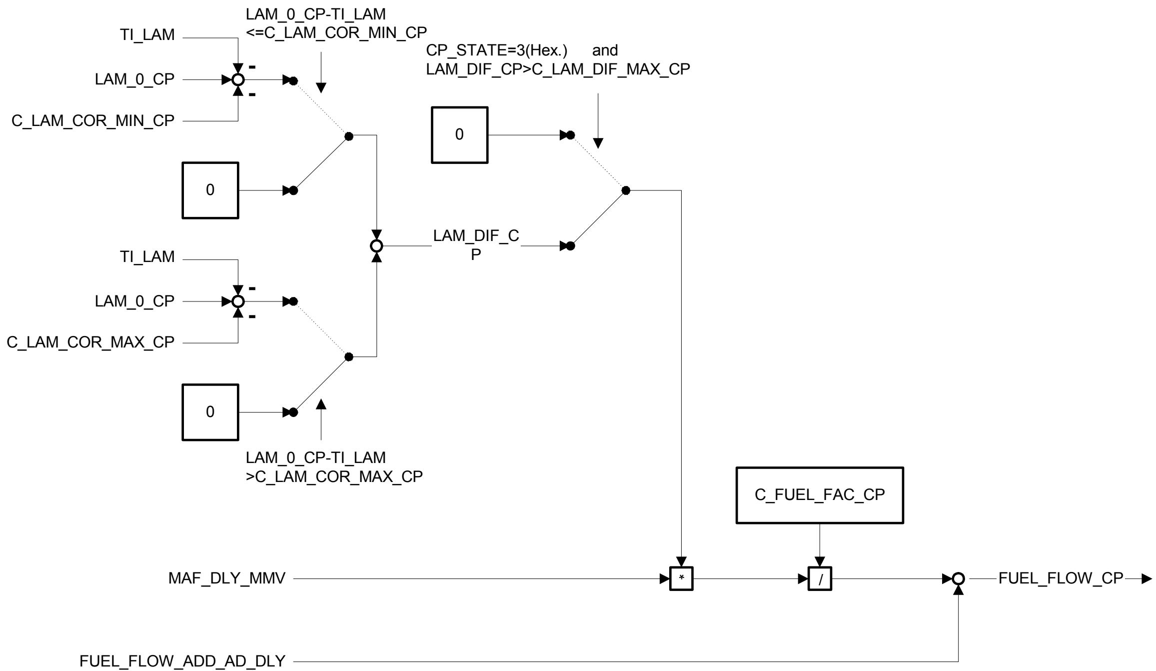
The fuel flow from the ACF when the CPS is open causes a deviation of the lambda control. This fuel flow can be calculated as a function of the difference of the lambda control values with closed/opened CPS LAM_DIF_CP, the mass air flow MAF_DLY_MMV and a constant value C_FUEL_FAC_CP. The constant C_FUEL_FAC_CP considers the influence of the fuel. The mass air flow MAF_DLY_MMV is the filtered value of MAF_FG_CYL using a Padé-filter and a moving mean value calculation to consider the delay time between the mass air flow MAF_FG_CYL and the TI_LAM calculation. The LAM_DIF_CP value can be shifted by a constant value C_LAM_COR_CP_MIN/MAX if they have the same value, to admit deviations of the lambda control. If C_LAM_COR_CP_MIN/MAX is applied as a window, LAM_DIF_CP is 0 in this window to avoid unstable CL behaviour.

The additive adaptive correction of the injected fuel mass flow FUEL_FLOW_ADD_AD_DLY caused by the canister purge is also regarded in the calculation. The calculation of the FUEL_FLOW_ADD_AD_DLY value is described in a separate chapter.

When the deviation of the lambda control LAM_DIF_CP is higher than a threshold value C_LAM_DIF_MAX or lower than -C_LAM_DIF_MAX, the fuel injection time is corrected by an additive correction value. LAM_DIF_CP is then set to 0 because the lambda deviation is corrected.

Signal flow diagram:

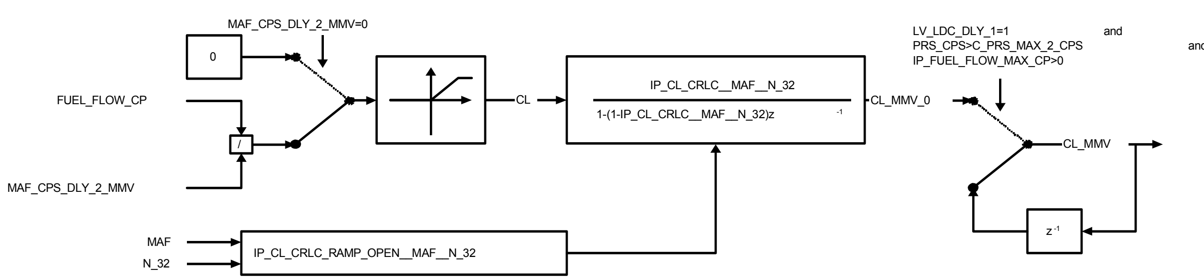
Calculation of FUEL_FLOW_CP for RAMP_OPEN:



TE10

The load CL of the ACF is calculated as the ratio of the fuel flow FUEL_FLOW_CP and the mass flow MAF_CPS_DLY_2_MMV through the CPS. The mass flow MAF_CPS_DLY_2_MMV is the filtered value of MAF_CPS_DLY using a Padé-filter and a moving mean value calculation to consider the delay time between the MAF_CPS_DLY and the FUEL_FLOW_CP calculation. The minimum and maximum values of this ratio are limited by 0 and C_CL_MAX_CP. The value of CL is filtered using a moving mean value calculation. The filter constant CL_CRLC depends on the mass air flow MAF and the engine speed N_32. During RAMP_OPEN operation the filter constant CL_CRLC for the CL_MMV calculation is different to the values during MAX_PURGE/RAMP_CLOSE to realize faster constants. The value of CL_MMV is only calculated when the engine is operated at limited dynamics (LV_LDC_CP=1), the difference pressure environment-intake manifold PRS_CPS is greater than a threshold value C_PRS_MAX_3_CPS and the value of IP_FUEL_FLOW_MAX_CP is greater than 0. Additionally is CL an CL_MMV not calculated after transition from NO_PURGE or MIN_PURGE to RAMP_OPEN until the time delay from the map IP_T_DLY_CP__MAF_CYL plus the time C_T_DLY_CP_CL has elapsed.

Calculation of Canister Load CL, CL_MMV for RAMP_OPEN:



TE12A

Formula section:

Calculation of REL_FLOW_CPS for RAMP_OPEN.

if LV_CP_RAMP_OPEN_STOP = 0
then **REL_FLOW_CPS**(n) = REL_FLOW_CPS(n-1)
+ IP_REL_FLOW_FAC_CP__CL_MMV * IP_REL_FLOW_INC_CP
else **REL_FLOW_CPS**(n) = REL_FLOW_CPS(n-1)
endif

Calculation of LAM_DIF_CP:

IF LAM_0_CP - TI_LAM <= C_LAM_COR_MIN_CP
THEN **LAM_DIF_CP** = LAM_0_CP - TI_LAM - C_LAM_COR_MIN_CP
ELSEIF LAM_0_CP - TI_LAM > C_LAM_COR_MAX_CP
THEN LAM_DIF_CP = LAM_0_CP - TI_LAM - C_LAM_COR_MAX_CP
ENDIF

Calculation of FUEL_FLOW_CP:

FUEL_FLOW_CP = MAF_DLY_MMV * LAM_DIF_CP / C_FUEL_FAC_CP
+FUEL_FLOW_ADD_AD_DLY

Calculation of CL, CL_MMV

IF MAF_CPS_DLY_2_MMV < C_MAF_CPS_DLY_2_MMV_MIN
THEN CL=0
ELSE **CL = FUEL_FLOW_CP / MAF_CPS_DLY_2_MMV**
ENDIF

IF CL<0 THEN CL=0
ELSEIF CL>C_CL_MAX_CP THEN CL=C_CL_MAX_CP
ELSE CL=CL
ENDIF

IF LV_LDC_CP=1
and PRS_CPS > C_PRS_MAX_3_CPS
and IP_FUEL_FLOW_MAX_CP>0
and MAF_CPS_DLY_2_MMV ≥ C_MAF_CPS_DLY_2_MMV_MIN
and T_DLY_4_DIAGCP = 0
and IP_T_DLY_CP__MAF_CYL + C_T_DLY_CP_CL elapsed after Transition
(see text above)
THEN **CL_MMV**(n) = (1-IP_CL_CRLC_RAMP_OPEN) * CL_MMV(n-1)
+ IP_CL_CRLC_RAMP_OPEN * CL(n)
ELSE CL_MMV(n)=CL_MMV(n-1)
ENDIF

9.17.3.3.1.1 Actions for the transition RAMP_OPEN to different states

- ***Decision for transition RAMP_OPEN to MAX_PURGE***

IF **REL_FLOW_CPS <= IP_REL_FLOW_MAX_CPS** AND
FUEL_FLOW_CP <= IP_FUEL_FLOW_MAX_CP
THEN calculations in RAMP_OPEN take place
ELSE actions for the transition to MAX_PURGE take place
ENDIF

Initialization of LAM_P_SUM_CP

IF LAM_P_SUM_CP(n)= 0
THEN LAM_P_SUM_CP(n)= LAM_P_CTR
ELSE LAM_P_SUM_CP(n)= LAM_P_SUM_CP(n-1)
ENDIF
IF LAM_P_CTR - LAM_P_SUM_CP >= C_LAM_P_CTR_MAX_CP
THEN RAMP_OPEN is now finished
CL_MMV_OLD = 0
REL_FLOW_CPS_OLD = 0
REL_FLOW_CPS_OLD_MIN = 0
LAM_P_SUM_CP = 0
Shift of TI_LAM at the end of RAMP_OPEN:
IF LAM_DIF_CP > 0
THEN LV_TI_LAM_COR_CP = 1
TI_LAM_CP = LAM_DIF_CP
ELSE TI_LAM_CP(n)= TI_LAM_CP(n-1)
ENDIF
LAM_DIF_CP = 0
ELSE waiting of C_LAM_P_CTR_MAX_CP P-Jumps
REL_FLOW_CPS(n)=REL_FLOW_CPS(n-1)
all other calculation for RAMP_OPEN are active
ENDIF

- ***P-jump counter phase:***

REL_FLOW_CPS_OLD_MIN= **MIN(**REL_FLOW_CPS_OLD**;** REL_FLOW_CPS_DLY_MMV**)**

- ***Transitions:***

RAMP_OPEN->NO_PURGE or MIN_PURGE or Lambda Adaptation Phase

⇒ LAM_P_SUM_CP = 0

- ***Actions for the transition RAMP_OPEN to RAMP_CLOSE***

Initialization of TI_LAM_CP_MMV

TI_LAM_CP_MMV = TI_LAM

- ***Setting of LV_RAMP_OPEN_END and LV_MAX_PURGE_END***

```
IF          (T_CP>C_T_1_CP    (first purge after start)
            or      T_CP>C_T_2_CP )
THEN        LV_RAMP_OPEN_END=1
            LV_MAX_PURGE_END=1
ELSEIF      (REL_FLOW_CPS>IP_REL_FLOW_MAX_CPS
     or     FUEL_FLOW_CP>IP_FUEL_FLOW_MAX_CP)
     and    LAM_P_CTR - LAM_P_SUM_CP >= C_LAM_P_CTR_MAX_CP
THEN        LV_RAMP_OPEN_END=1
            IF      LAM_DIF_CP > 0      (Reset of TI_LAM)
            THEN  TI_LAM_CP = TI_LAM_CP + LAM_DIF_CP
            ENDIF
ELSE        LV_RAMP_OPEN_END=0
ENDIF
```

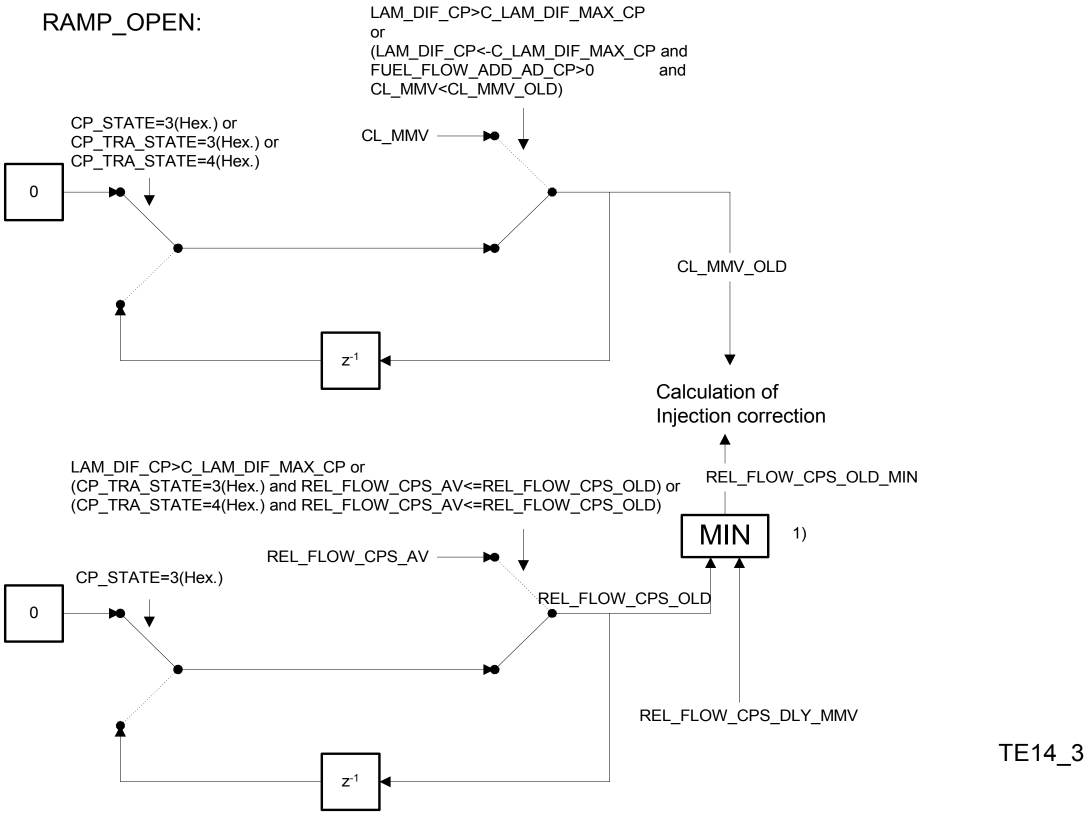
- ***Calculation of CL_MMV_OLD and REL_FLOW_CPS_OLD***

When the RAMP_OPEN operation is disabled the injection time is corrected during the transition to the MIN_PURGE or NO_PURGE operation (CP_TRA_STATE=1 or 2 (Hex.)). For this the last stored value of CL_MMV_OLD and the actual value of REL_FLOW_CPS (if REL_FLOW_CPS_AV is lower than REL_FLOW_CPS_OLD) are used to calculate the value of FUEL_FLOW_ADD_AD_CP. When the transition phase is over, the injection time isn't corrected any longer.

Signal flow diagram:

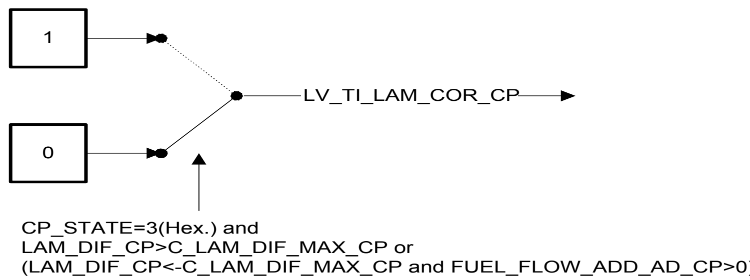
Calculation of REL_FLOW_CPS_OLD and CL_MMV_OLD:

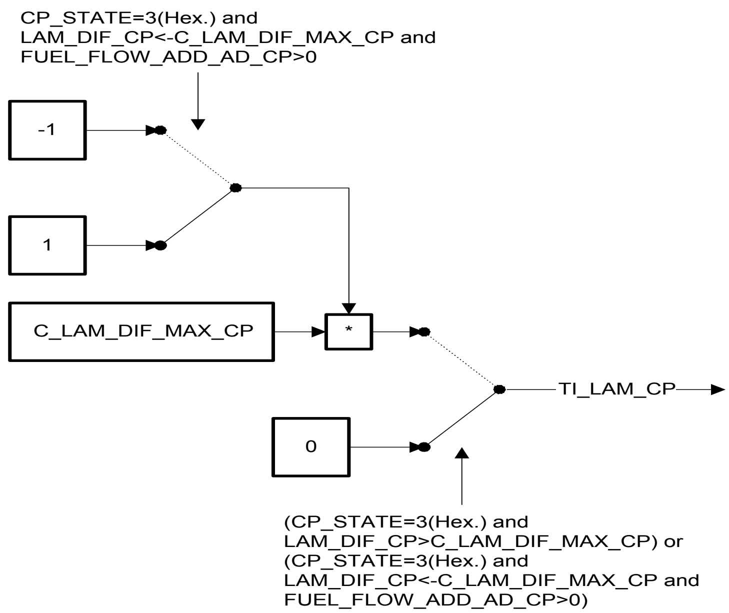
RAMP_OPEN:



1) MIN selection only active at end of RAMP_OPEN when p-jump counter active or at transiton to NO_PURGE or MIN_PURGE

Setting of LV_TI_LAM_COR_CP and TI_LAM_CP:





TE15

Formula section:

- ***Calculation of CL_MMV_OLD and REL_FLOW_CPS_OLD***

```
IF       CP_STATE=3 (Hex.)              #RAMP_OPEN active
THEN     IF       LAM_DIF_CP>C_LAM_DIF_MAX_CP
         THEN     CL_MMV_OLD(n)=CL_MMV(n)
                  REL_FLOW_CPS_OLD(n) =REL_FLOW_CPS_DLY_MMV(n)
                  REL_FLOW_CPS_OLD_MIN = REL_FLOW_CPS_OLD
         ELSE     IF     FUEL_FLOW_ADD_AD_CP>0       and
                         LAM_DIF_CP<-C_LAM_DIF_MAX_CP
                  THEN   CL_MMV_OLD(n)=CL_MMV(n)
                  ELSE   CL_MMV_OLD(n)=CL_MMV_OLD(n-1)
                         REL_FLOW_CPS_OLD=REL_FLOW_CPS_OLD(n-1)
                         REL_FLOW_CPS_OLD_MIN=
                               REL_FLOW_CPS_OLD_MIN(n-1)
         ENDIF

ELSEIF   CP_TRA_STATE(n)=3 (Hex.)        or    #Transition
         CP_TRA_STATE=4 (Hex.))                #RAMP_OPEN
                                               #      ---->
                                               #NO_PURGE or MIN_PURGE
THEN     CL_MMV_OLD(n)=CL_MMV_OLD(n-1)
         REL_FLOW_CPS_OLD_MIN(n)=MIN( REL_FLOW_CPS_DLY_MMV(n)
                                     , REL_FLOW_CPS_OLD)
ELSE     CL_MMV_OLD=0
         REL_FLOW_CPS_OLD=0
         REL_FLOW_CPS_OLD_MIN=0
ENDIF
```

- ***Setting of LV_TI_LAM_COR_CP***

```
IF       CP_STATE=3(Hex.)                and   #RAMP_OPEN
         LAM_DIF_CP > C_LAM_DIF_MAX_CP         #active
THEN     LV_TI_LAM_COR_CP=1
         TI_LAM_CP=C_LAM_DIF_MAX_CP
ELSE     LV_TI_LAM_COR_CP=0
         TI_LAM_CP=0
ENDIF
```

9.17.3.3.2 MAX_PURGE Operation

General information:

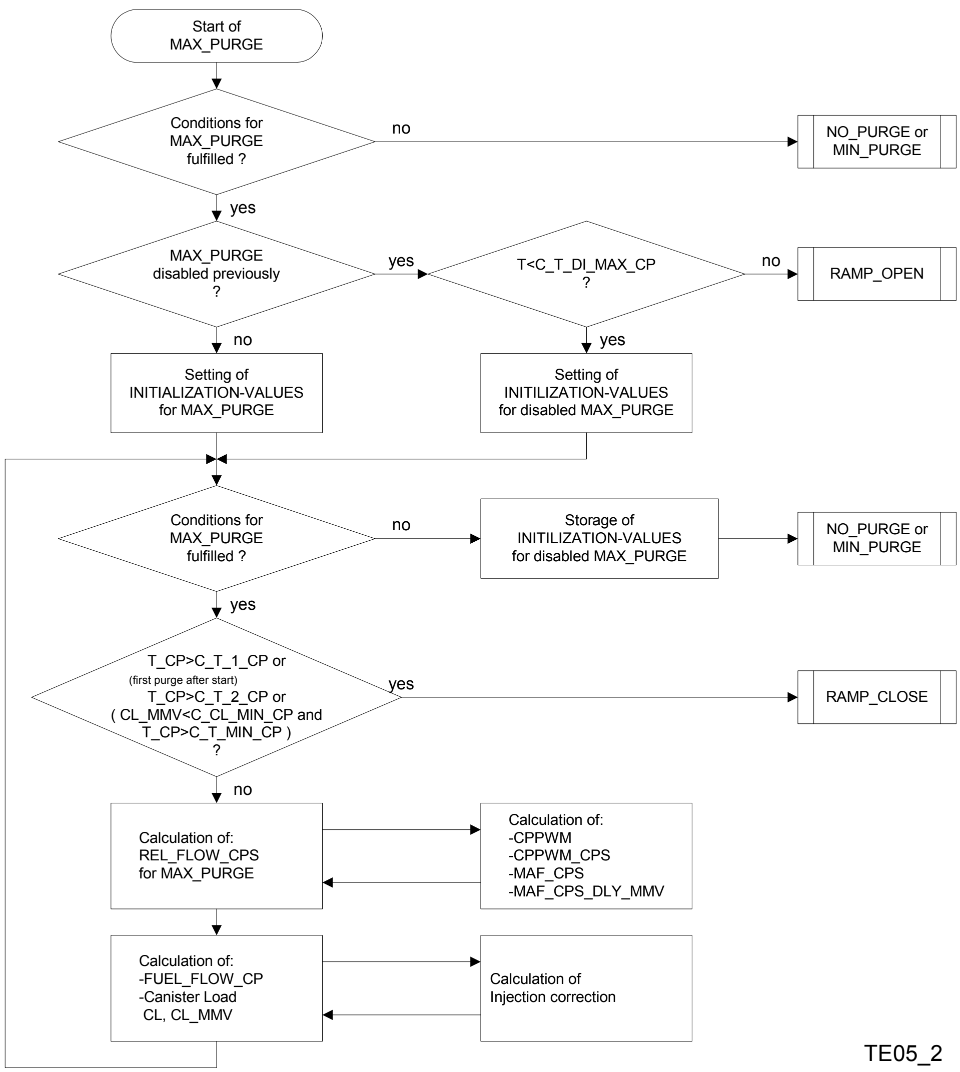
The MAX_PURGE operation is started when the RAMP_OPEN operation is finished (LV_RAMP_OPEN_END=1).

During MAX_PURGE operation the value of REL_FLOW_CPS and so the opening of the CPS is controlled by the load CL_MMV of the ACF.

The MAX_PURGE operation ends, when the purge time T_CP since the start of the NORMAL_PURGE operation is greater than the constant C_T_1_CP (for the first purge after start) or C_T_2_CP or if the canister load CL_MMV is lower than a threshold value C_CL_MIN_CP and the time T_CP is greater than T_MIN_CP. Then the value of LV_MAX_PURGE_END is set to 1 and the RAMP_CLOSE operation starts.

Signal flow diagram:

MAX_PURGE



- ***INITIALIZATION-VALUES for MAX_PURGE***

Before MAX_PURGE operation is started the last calculated values of the RAMP_OPEN operation are set as INITIALIZATION-VALUES for MAX_PURGE. For the initialization of the moving mean value calculation of the TI_LAM value the following value is set:

TI_LAM_CP_MMV(n-1)=TI_LAM

- ***INITIALIZATION-VALUES for disabled MAX_PURGE***

If the MAX_PURGE operation was disabled (Conditions for MAX_PURGE are not fulfilled) for a time t<C_T_DI_MAX_CP, the following INITIALIZATION-VALUES for disabled MAX_PURGE are set:

LAM_0_CP =LAM_0_CP_DI

CL_MMV =CL_MMV_DI

REL_FLOW_CPS =REL_FLOW_CPS_DI

MAF_CYL_DLY_OLD=MAF_FG_CYL

MAF_DLY_MMV =0

If the MAX_PURGE operation is disabled, the following actual values for CL_DI, REL_FLOW_CPS_DI and LAM_0_DI before the operation is disabled are stored:

LAM_0_CP_DI =LAM_0_CP

CL_MMV_DI =CL_MMV

REL_FLOW_CPS_DI=REL_FLOW_CPS

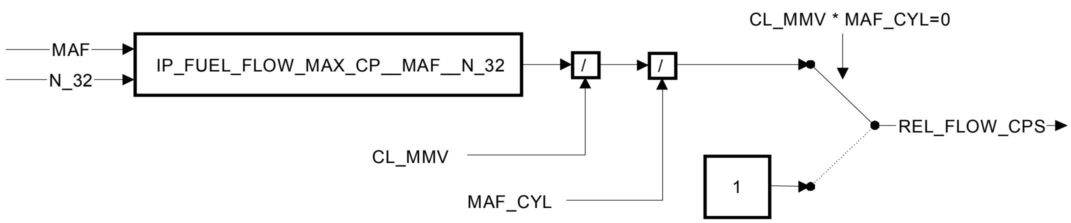
IF the disabling time is longer than C_T_DI_MAX_CP, the value of CL_MMV_DI is set to 0.

- ***Calculation of REL_FLOW_CPS for MAX_PURGE***

The value of REL_FLOW_CPS during MAX_PURGE operation is calculated as a function of the moving mean value of the canister load CL_MMV, the mass air flow into the cylinders of the engine and an applicable value IP_FUEL_FLOW_MAX_CP, that is dependent on the mass air flow MAF and the speed N_32 of the engine. REL_FLOW_CPS is calculated so, that the fuel flow FUEL_FLOW_CP is kept constant to the mapped value IP_FUEL_FLOW_MAX_CP.

<u>Signal flow diagram:</u>

Calculation of REL_FLOW_CPS for MAX_PURGE:

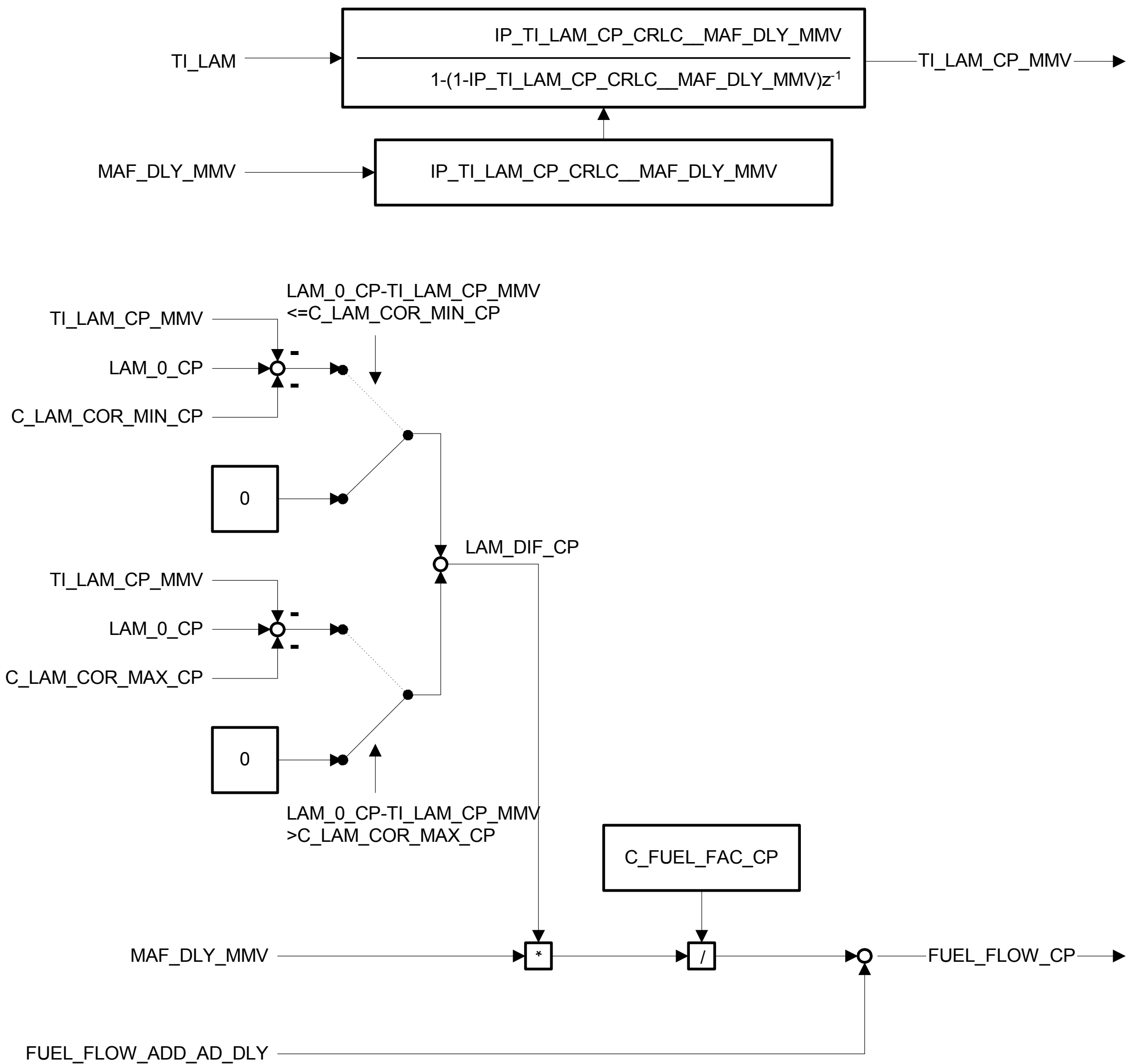


- ***Calculation of FUEL_FLOW_CP and Canister Load CL, CL_MMV***

Unlike the calculation of FUEL_FLOW_CP in RAMP_OPEN operation for this calculation in MAX_PURGE the moving mean value TI_LAM_CP_MMV of TI_LAM is used as input to filter the influence of the closed loop lambda control. The other calculations are the same as in RAMP_OPEN operation.

Signal flow diagram:

Calculation of FUEL_FLOW_CP for MAX_PURGE:

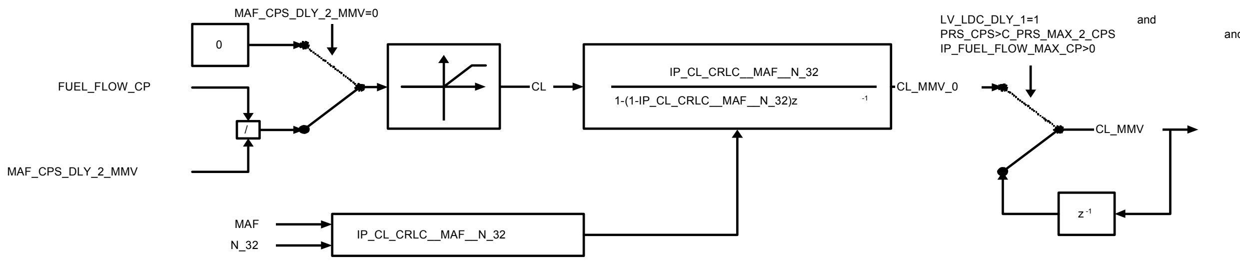


TE11

The calculation of the canister load CL is the same as used in the RAMP_OPEN operation. For the calculation of the moving mean value CL_MMV only the value IP_CL_CRLC is used.

Signal flow diagram:

Calculation of Canister Load CL, CL_MMV for MAX_PURGE:



- ***Calculation of CPPWM, CPPWM_CPS and MAF_CPS***

The calculation of the CPS control signals CPPWM, CPPWM_CPS and the mass flow through the CPS MAF_CPS is described in a separate chapter.

Formula section:

- ***Calculation of REL_FLOW_CPS for MAX_PURGE***

if STATE_DIAGCP = PRS_DYN (01H)
and CP_STATE = MAX_PURGE
then LV_CP_MAX_ST_DIAGCP = 1
LV_MAX_CP_END_DIAGCP = 0
LV_T_DLY_4_END_DIAGCP = 0
else if STATE_DIAGCP = START(00H) or (0C~0EH)
then LV_CP_MAX_ST_DIAGCP = 0
endif

if CL_MMV*MAF_FG_CYL=0
then REL_FLOW_CPS=1
else if STATE_DIAGCP = T_DLY_3
and LC_CONF_CP_TRA_DIAGCP = 1
then REL_FLOW_CPS_TMP_MAX
= IP_FUEL_FLOW_MAX_CP / (CL_MMV * MAF_FG_CYL)
If T_DLY_4_DIAGCP is set to C_T_DLY_4_DIAGCP and becomes 0sec after T_DLY_3 application
(1. T_DLY_4_DIAGCP is set only at $CP\_STATE_{(n-1)}$ = NO_PURGE
2. LV_T_DLY_4_END_DIAGCP = 0 to 1 after T_DLY_4_DIAGCP)
and REL_FLOW_CPS_TMP_INC < REL_FLOW_CPS_TMP_MAX
and LV_CP_MAX_ST_DIAGCP = 1

```
                then    LV_MAX_CP_TRA_DIAGCP = 1
                else    if      REL_FLOW_CPS_TMP_INC ≥ REL_FLOW_CPS_TMP_MAX
                then    LV_MAX_CP_TRA_DIAGCP = 0
                        LV_MAX_CP_END_DIAGCP = 1
                endif
                endif
                if      { T_DLY_4_DIAGCP is not expired (> 0 sec)
                        or      ( DTP ≤ C_DTP_MIN_CP_ACT
                                and     LV_MAX_CP_END_DIAGCP = 0 ) }
                then    REL_FLOW_CPS_INC = 0
                REL_FLOW_CPS_TMP_INC = 0
                REL_FLOW_CPS = 0
                else    if      DTP > (C_DTP_MIN_CP_ACT + C_DTP_MIN_CP_ACT_HYS)
                then    if      LV_MAX_CP_TRA_DIAGCP = 1
                        then    REL_FLOW_CPS_INC = IP_REL_FLOW_INC_CP_TRA
                                REL_FLOW_CPS_TMP_INC(n)
                                = REL_FLOW_CPS_TMP_INC(n-1) + REL_FLOW_CPS_INC
                                REL_FLOW_CPS = REL_FLOW_CPS_TMP_INC
                        else    REL_FLOW_CPS = REL_FLOW_CPS_TMP_MAX
                        endif
                endif
                endif
        else    REL_FLOW_CPS = IP_FUEL_FLOW_MAX_CP / (CL_MMV * MAF_FG_CYL)
                REL_FLOW_CPS_INC = 0
                REL_FLOW_CPS_TMP_INC = 0
                REL_FLOW_CPS_TMP_MAX = 0
                LV_MAX_CP_TRA_DIAGCP = 0
        endif
endif
```

- ***Calculation of FUEL_FLOW_CP***

<u>Calculation of TI_LAM_CP_MMV:</u>

TI_LAM_CP_MMV(n)=(1-IP_TI_LAM_CP_CRLC__MAF_DLY_MMV) * TI_LAM_CP_MMV(n-1)
+IP_TI_LAM_CP_CRLC__MAF_DLY_MMV * TI_LAM(n)

<u>Calculation of LAM_DIF_CP:</u>

IF LAM_0_CP - TI_LAM_CP_MMV <= C_LAM_COR_MIN_CP

THEN LAM_DIF_CP = LAM_0_CP - TI_LAM_CP_MMV - C_LAM_COR_MIN_CP

```
ELSEIF     LAM_0_CP - TI_LAM_CP_MMV > C_LAM_COR_MAX_CP
    THEN   LAM_DIF_CP = LAM_0_CP - TI_LAM_CP_MMV - C_LAM_COR_MAX_CP
    ELSE          LAM_DIF_CP = 0
ENDIF
```

FUEL_FLOW_CP = MAF_DLY_MMV * LAM_DIF_CP/C_FUEL_FAC_CP
+FUEL_FLOW_ADD_AD_DLY

- ***Calculation of CL, CL_MMV***

```
IF          MAF_CPS_DLY_2_MMV < C_MAF_CPS_DLY_2_MMV_MIN
THEN        CL=0
ELSE        CL = FUEL_FLOW_CP / MAF_CPS_DLY_2_MMV
ENDIF

IF          CL<0
THEN        CL=0
ELSEIF      CL>C_CL_MAX_CP
THEN        CL=C_CL_MAX_CP
ELSE        CL=CL
ENDIF

IF          LV_LDC_CP=1
and         IP_FUEL_FLOW_MAX_CP>0
and         MAF_CPS_DLY_2_MMV_MIN ≥ C_MAF_CPS_DLY_2_MMV_MIN
and         T_DLY_4_DIAGCP = 0
THEN
    IF      ( CL(n) > CL_MMV(n-1)  and  PRS_CPS > C_PRS_MAX_3_CPS )
    or      ( CL(n) < CL_MMV(n-1)  and  PRS_CPS > C_PRS_MAX_4_CPS )
    THEN    CL_MMV(n) = (1-IP_CL_CRLC)*CL_MMV(n-1) + IP_CL_CRLC*CL(n)
    ELSE    CL_MMV(n)=CL_MMV(n-1)
    ENDIF
ELSE        CL_MMV(n)=CL_MMV(n-1)
ENDIF
```

- ***Setting of LV_MAX_PURGE_END***

```
IF          {  T_CP>C_T_1_CP            (first purge after start)
            or      T_CP>C_T_2_CP
            or      (CL_MMV<C_CL_MIN_CP and T_CP > T_MIN_CP) }
THEN        LV_MAX_PURGE_END=1
ELSE        LV_MAX_PURGE_END=0
ENDIF
```

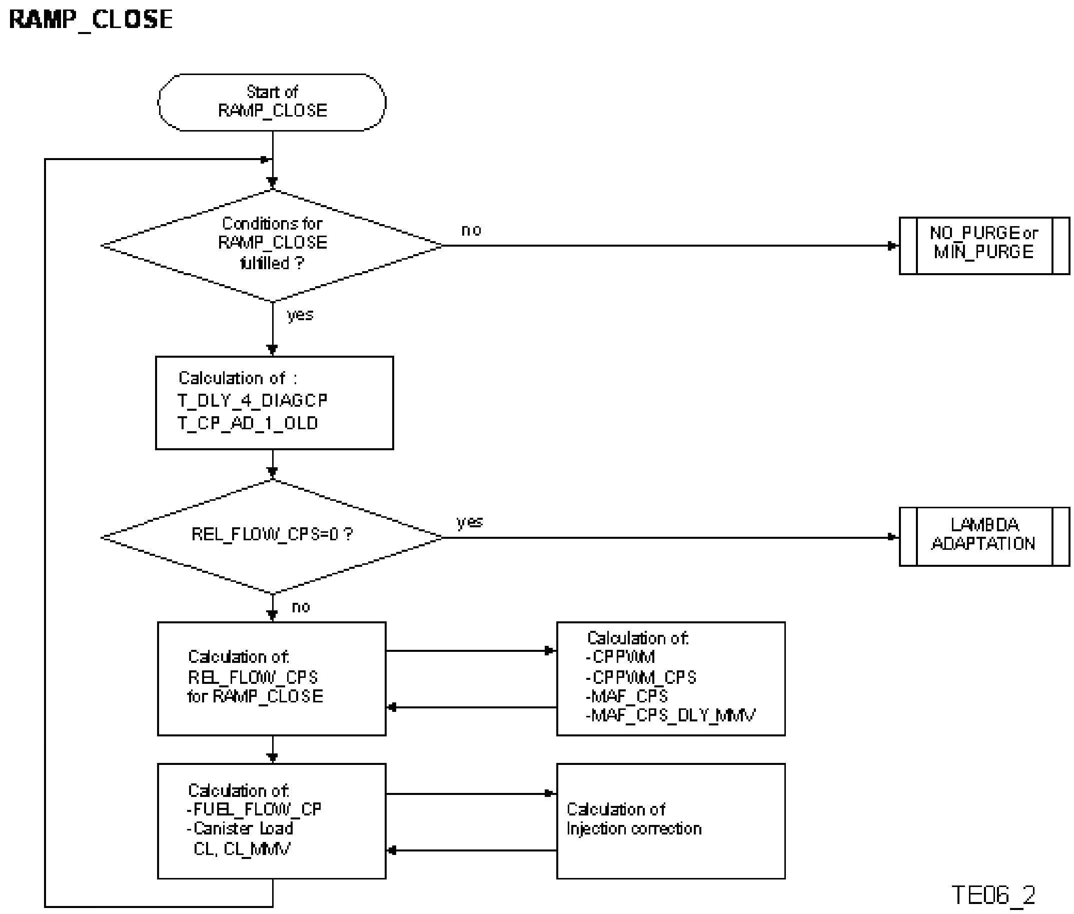
9.17.3.3.3 RAMP_CLOSE Operation

FUNCTION DESCRIPTION:

General information:

The RAMP_CLOSE operation starts, if the RAMP_OPEN or the MAX_PURGE operation is finished.

Signal flow diagram:



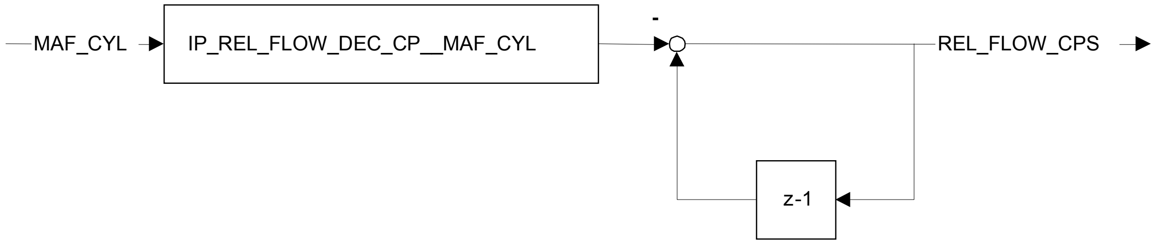
The RAMP_CLOSE operation ends when the value of REL_FLOW_CPS is 0 and T_PL_DIAGCP is 0. Then the variable LV_RAMP_CLOSE_END is set to 1 and the lambda adaptation is started.

- ***Calculation of REL_FLOW_CPS for RAMP_CLOSE***

When the RAMP_CLOSE operation is activated REL_FLOW_CPS is decremented. The decrement value depends on the map values of IP_REL_FLOW_DEC_CP__MAF_CYL. The closing speed of the CPS is so controlled by the mass air flow MAF_CYL. The Initialization value of REL_FLOW_CPS for RAMP_CLOSE is the last calculated value of MIN(REL_FLOW_CPS, IP_REL_FLOW_MAX_CPS) if the preceding operation was MAX_PURGE. If the RAMP_OPEN operation was preceding the Initialization value is the last calculated value of REL_FLOW_CPS of the RAMP_OPEN operation.

Signal flow diagram:

Calculation of REL_FLOW_CPS for RAMP_CLOSE:



- **Calculation of FUEL_FLOW_CP, TI_LAM_CP_MMV and LAM_DIF_CP**

The calculations of this values are the same as in MAX_PURGE operation.

- ***Calculation of CPPWM, CPPWM_CPS and MAF_CPS***

The calculation of the CPS control signals CPPWM, CPPWM_CPS and the mass flow through the CPS MAF_CPS is described in a separate chapter.

Formula section:

- ***Calculation of T_DLY_4_DIAGCP & T_CP_AD_1_OLD***

At the beginning of RAMP_CLOSE, delay timer T_DLY_4_DIAGCP should be set to 0 and NO_PURGE duration time should be kept with T_CP_AD_1_OLD.

T_DLY_4_DIAGCP = 0

T_CP_AD_1_OLD = T_CP_AD_1

- ***Calculation of REL_FLOW_CPS for RAMP_CLOSE***

REL_FLOW_CPS(n)=REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP

- ***Calculation of CL, CL_MMV***

```
If      MAF_CPS_DLY_2_MMV < C_MAF_CPS_DLY_2_MMV_MIN
then    CL=0
else    CL = FUEL_FLOW_CP / MAF_CPS_DLY_2_MMV
endif

if      CL<0
then    CL=0
elseif  CL>C_CL_MAX_CP
then    CL=C_CL_MAX_CP
else    CL=CL
endif

if      LV_LDC_CP=1
and     IP_FUEL_FLOW_MAX_CP>0
then    if      ( CL(n) > CL_MMV(n-1)  and  PRS_CPS > C_PRS_MAX_3_CPS )
        or      ( CL(n) < CL_MMV(n-1)  and  PRS_CPS > C_PRS_MAX_4_CPS )
        then    CL_MMV(n) = (1-IP_CL_CRLC)*CL_MMV(n-1) + IP_CL_CRLC*CL(n)
        else    CL_MMV(n)=CL_MMV(n-1)
        endif
  else  CL_MMV(n)=CL_MMV(n-1)
  endif
```

- ***Setting of LV_RAMP_CLOSE_END and LV_NORM_PURGE_END_1***

```
IF      REL_FLOW_CPS=0
AND     T_PL_DIAGCP = 0
THEN    LV_RAMP_CLOSE_END=1
ELSE    LV_RAMP_CLOSE_END=0
        LV_NORM_PURGE_END_1=0
ENDIF
```

9.17.3.3.4 Calculation of MAF_CYL_DLY and MAF_DLY_MMV:

FAC_1 = IP_T_DLY_CP__MAF_CYL /0.1 ***#(0.1=time recurrence)***

FAC_2 = FAC_1*FAC_1/3

MAF_CYL_DLY$_{(n)}$ = 1 / (1+FAC_1+FAC_2)

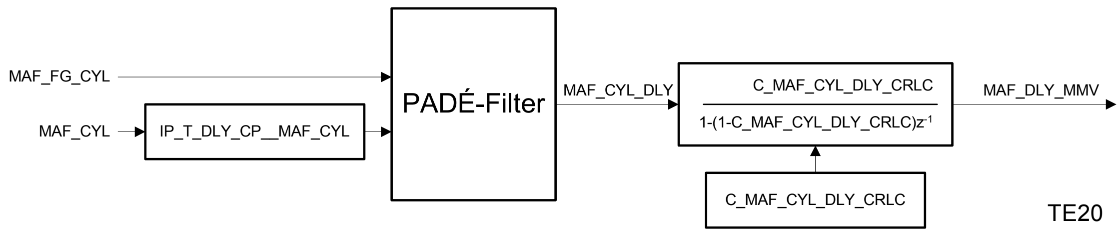
$$* [(1\text{-}FAC\_1+FAC\_2)*(MAF\_FG\_CYL_{(n)} - MAF\_CYL\_DLY_{(n-2)}) + 2*(1\text{-}FAC\_2) * (MAF\_FG\_CYL_{(n-1)} - MAF\_CYL\_DLY_{(n-1)})] + MAF\_FG\_CYL_{(n-2)}$$

$$\mathbf{MAF\_DLY\_MMV_{(n)}} = (1 - C\_MAF\_CYL\_DLY\_CRLC) * MAF\_DLY\_MMV_{(n-1)} + C\_MAF\_CYL\_DLY\_CRLC * MAF\_CYL\_DLY_{(n)}$$

MAF_CYL_OLD2 = MAF_CYL_OLD for next step = $MAF\_FG\_CYL_{(n-2)}$

MAF_CYL_OLD = MAF_FG_CYL for next step = $MAF\_FG\_CYL_{(n-1)}$

Calculation of MAF_CYL_DLY and MAF_CYL_DLY_MMV:



9.17.3.3.5 Storage of values for EVAP System Monitoring

FUNCTION DESCRIPTION:

When the MAX_PURGE operation ends, because the time T_CP is greater than C_T_1_CP (first purge after start) or C_T_2_CP or the canister load CL_MMV is lower than a threshold value C_CL_MIN_CP and the time T_CP is greater than T_MIN_CP, the last calculated value of the canister load CL_MMV is stored as CL_MMV_NORM_PURGE_END before RAMP_CLOSE operation starts. This value is used as Input for the OBDII EVAP-System Monitoring. If MAX_PURGE is disabled because of fuel cut off (LV_FCUT_CP=1) or a too low pressure difference (PRS_CPS<C_PRS_MAX_1_CPS) and during the disabling time the time for the lambda adaptation phase starts, then also the last value of CL_MMV is used as CL_MMV_NORM_PURGE_END.

For the EVAP system monitoring also the constant LV_NORM_PURGE_END_1 is required. This logical value is set to 1 if the RAMP_CLOSE operation ends. If MAX_PURGE is disabled because of fuel cut off (LV_FCUT_CP=1) or a too low pressure difference (PRS_CPS<C_PRS_MAX_1_CPS) and during the disabling time the time for the lambda adaptation phase starts then also LV_NORM_PURGE_END_1 is set to 1.

Formula section:

- ***Calculation of CL_MMV_NORM_PURGE_END***

IF **{** T_CP>C_T_1_CP (first purge after start)
or T_CP>C_T_2_CP
or (CL_MMV<C_CL_MIN_CP and T_CP > T_MIN_CP) **}**
THEN CL_MMV_NORM_PURGE_END=CL_MMV
ELSEIF CP_STATE=MAX_PURGE and
CP_T_STATE=T_TI_AD_ACTIVE and

```
               (LV_FCUT_CP=1 or PRS_CPS<C_PRS_MAX_1_CPS)
THEN           CL_MMV_NORM_PURGE_END=CL_MMV
ELSE           CL_MMV_NORM_PURGE_END=0
ENDIF
```

- ***Calculation of LV_NORM_PURGE_END_1***

```
IF                   LV_RST_NORM_PURGE_END=1
THEN                 LV_NORM_PURGE_END_1=0
ELSEIF               LV_RAMP_CLOSE_END=1
THEN       IF              LV_NORM_PURGE_END_1=0
           THEN            LV_NORM_PURGE_END_1
                           = LV_RAMP_OPEN_END  and  LV_MAX_PURGE_END
           ELSE            LV_NORM_PURGE_END_1 = LV_NORM_PURGE_END_1
           ENDIF
ELSEIF     CP_STATE=MAX_PURGE                    and
           CP_T_STATE=T_TI_AD_ACTIVE             and
           (LV_FCUT_CP=1 or PRS_CPS<C_PRS_MAX_1_CPS)
THEN       LV_NORM_PURGE_END_1 = 1
ELSE       LV_NORM_PURGE_END_1 = LV_NORM_PURGE_END_1
ENDIF
```

9.17.3.4 Transitions between the Partial Functions

General information:

During enabled Evaporative Emission Control (LV_CP_ENA=1), there is a continuous check, which partial function is activated. Depending on the conditions an alternation of the partial functions is possible. So in addition to the normal transitions to the partial functions some extra transitions are possible.

The following transitions are possible:

1. NO_PURGE ----> MIN_PURGE (CP_TRA_STATE=1 (Hex.))
2. NO_PURGE <---- MIN_PURGE (CP_TRA_STATE=2 (Hex.))
3. RAMP_OPEN <---- NO_PURGE (CP_TRA_STATE=0 (Hex.))
4. RAMP_OPEN ----> NO_PURGE (CP_TRA_STATE=3 (Hex.))
5. RAMP_OPEN <---- MIN_PURGE (CP_TRA_STATE=0 (Hex.))
6. RAMP_OPEN ----> MIN_PURGE (CP_TRA_STATE=4 (Hex.))
7. MAX_PURGE <---- NO_PURGE (CP_TRA_STATE=0 (Hex.))
8. MAX_PURGE ----> NO_PURGE (CP_TRA_STATE=5 (Hex.))
9. MAX_PURGE <---- MIN_PURGE (CP_TRA_STATE=0 (Hex.))
10. MAX_PURGE ----> MIN_PURGE (CP_TRA_STATE=6 (Hex.))
11. RAMP_CLOSE ----> NO_PURGE (CP_TRA_STATE=7 (Hex.))
12. RAMP_CLOSE ----> MIN_PURGE (CP_TRA_STATE=8 (Hex.))
13. RAMP_OPEN ----> RAMP_CLOSE (CP_TRA_STATE=9 (Hex.))
14. MAX_PURGE ----> RAMP_CLOSE (CP_TRA_STATE=A (Hex.))
15. WAIT_RAMP_OPEN_PURGE <---- MIN_PURGE (CP_TRA_STATE=B (Hex.))
16. WAIT_RAMP_OPEN_PURGE ----> MIN_PURGE (CP_TRA_STATE=C (Hex.))

Normally the transitions are performed with a decrease/increase of the relative flow through the CPS REL_FLOW_CPS as it is performed in the RAMP_OPEN / RAMP_CLOSE operation with the following exceptions:

- REL_FLOW_CPS is set to 0 without transition in ramp mode:

If injection fuel cut off is active (LV_FCUT_CP=1) or
If PRS_CPS is lower than C_PRS_MAX_1_CPS or
If EVAP / CPS diagnosis is active or
(LV_CP_SET_CLOSE=1, LV_ACT_DIAGCP=1, LV_ACT_DIAGCPS=1)

If injection time correction of lambda controller or
TI_LAM is higher or lower than threshold values
(TI_LAM> IP_TI_LAM_MAX_CP__TIA or TI_LAM<C_TI_LAM_MIN_CP)
If lambda controller is limited (LV_LAM_LIM_MAX_CYCNR=1) or
If maximum value of REL_FLOW_CPS is limited to 0 or
(IP_FUEL_FLOW_MAX_CP=0 or IP_REL_FLOW_MAX_CPS=0)
If Lambda controller active with limitation
(LV_LAM_LIM_MAX=1 or LV_LAM_LIM_MIN=1)

- If the RAMP_OPEN or the MAX_PURGE operation was interrupted (NO_PURGE or MIN_PURGE active) and the interrupt-time is lower than the adjustable times C_T_DI_RAMP_OPEN_CP (RAMP_OPEN operation) or C_T_DI_MAX_CP (MAX_PURGE operation), the INITIALIZATION-VALUES of REL_FLOW_CPS for disabled RAMP_OPEN or disabled MAX_PURGE are set. So the RAMP_OPEN or MAX_PURGE operation starts immediately with the values, which were stored when the operation was interrupted. When the interrupt time is longer than the times C_T_DI_RAMP_OPEN_CP or C_T_DI_MAX_CP the NORMAL_PURGE always starts with the RAMP_OPEN operation. Therefore the variable LV_RAMP_OPEN_END is set to 0 when the MAX_PURGE operation was disabled for a time period longer than C_T_DI_MAX_CP.
- For the transition to the RAMP_OPEN from MIN_PURGE operation the CPS is closed for an adjustable time C_T_WAIT_RAMP_OPEN and then the CPS is opened in order to get useful start values of the lambda controller for the determination of the canister load CL.

Transitions from the partial functions MIN_PURGE or NO_PURGE to RAMP_CLOSE operation are not possible.

The transition ends (CP_TRA_STATE=0 (Hex.)), when the value of REL_FLOW_CPS is equal to the setpoint value of the active partial function.

<u>**Formula section:**</u>

- ***Definition of CP_NORM_DEAC_STATE***

```
IF       CP_STATE(n-1)=3 (Hex.)          and   #Short Deactivation of
         (CP_STATE(n)=1 (Hex.)                 #RAMP_OPEN
         or
         CP_STATE(n)=2 (Hex.))
THEN     FOR        t=0,C_T_DI_RAMP_OPEN_CP
         CP_NORM_DEAC_STATE=1 (Hex.)
ELSEIF   CP_STATE(n-1)=4 (Hex.)          and   #Short Deactivation of
         (CP_STATE(n)=1 (Hex.)                 #MAX_PURGE
         or
         CP_STATE(n)=2 (Hex.))
THEN     FOR        t=0,C_T_DI_MAX_CP
         CP_NORM_DEAC_STATE=2 (Hex.)
ELSE     CP_NORM_DEAC_STATE=0 (Hex.)
ENDIF
```

- ***Transitions:***

NO_PURGE ---> MIN_PURGE

```
IF       IP_REL_FLOW_MIN_CPS(n)- REL_FLOW_CPS(n-1)> IP_REL_FLOW_INC_CP
THEN     REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)+IP_REL_FLOW_INC_CP
         CP_TRA_STATE=1 (Hex.)
ELSE     REL_FLOW_CPS(n)    =IP_REL_FLOW_MIN_CPS(n)
         CP_TRA_STATE=0 (Hex.)
ENDIF
```

MIN_PURGE ---> NO_PURGE

IF	LV_FCUT_CP=1
or	PRS_CPS<C_PRS_MAX_1_CPS
or	LV_RAMP_CLOSE_END=1
or	LV_CP_SET_CLOSE=1
or	LV_ACT_DIAGCP=1
or	LV_ACT_DIAGCPS=1
or	TI_LAM> IP_TI_LAM_MAX_CP__TIA
or	TI_LAM<C_TI_LAM_MIN_CP
or	LV_LAM_LIM_MAX_CYCNR=1
or	IP_REL_FLOW_MAX_CPS=0
or	IP_FUEL_FLOW_MAX_CP=0
or	LV_LAM_LIM_MAX=1
or	LV_LAM_LIM_MIN=1
or	LV_OBD_AUTH_CP_CLOSE = 1
THEN	REL_FLOW_CPS(n)=0
	Settings for CPS-Diagnosis
	CP_TRA_STATE=0 (Hex.)
ELSEIF	REL_FLOW_CPS(n-1) >IP_REL_FLOW_DEC_CP
THEN	REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1) - IP_REL_FLOW_DEC_CP
	CP_TRA_STATE=2 (Hex.)
ELSE	REL_FLOW_CPS(n)=0
	CP_TRA_STATE=0 (Hex.)
ENDIF	

NO_PURGE ---> RAMP_OPEN

IF	CP_NORM_DEAC_STATE=1 (Hex.)
THEN	REL_FLOW_CPS(n)=REL_FLOW_CPS_DI
	CP_TRA_STATE=0 (Hex.)
ELSE	Start of RAMP_OPEN
	LV_NORM_PURGE_END_1=0
	CP_TRA_STATE=0 (Hex.)
ENDIF	

RAMP_OPEN ---> NO_PURGE

IF	LV_FCUT_CP=1
or	PRS_CPS<C_PRS_MAX_1_CPS
or	LV_RAMP_CLOSE_END=1
or	LV_CP_SET_CLOSE=1
or	LV_ACT_DIAGCP=1
or	LV_ACT_DIAGCPS=1
or	TI_LAM> IP_TI_LAM_MAX_CP__TIA
or	TI_LAM<C_TI_LAM_MIN_CP
or	LV_LAM_LIM_MAX_CYCNR=1

```
or          IP_REL_FLOW_MAX_CPS=0
or          IP_FUEL_FLOW_MAX_CP=0
or          LV_LAM_LIM_MAX=1
or          LV_LAM_LIM_MIN=1
or          LV_OBD_AUTH_CP_CLOSE = 1
THEN        REL_FLOW_CPS(n)=0
            CP_TRA_STATE=0 (Hex.)
ELSEIF      REL_FLOW_CPS(n-1) > IP_REL_FLOW_DEC_CP
THEN        REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
            CP_TRA_STATE=3 (Hex.)
ELSE        REL_FLOW_CPS(n)=0
            CP_TRA_STATE=0 (Hex.)
ENDIF
```

MIN_PURGE ---> (WAIT_RAMP_OPEN_PURGE) ---> RAMP_OPEN

```
IF          CP_NORM_DEAC_STATE=1 (Hex.)
THEN        REL_FLOW_CPS(n)=REL_FLOW_CPS_DI
            CP_TRA_STATE=0 (Hex.)
ELSE        FOR    t=0,C_T_WAIT_RAMP_OPEN
                   Activation of WAIT_RAMP_OPEN_PURGE
                   IF      REL_FLOW_CPS(n-1)>IP_REL_FLOW_DEC_CP
                   THEN    REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
                   ELSE    REL_FLOW_CPS(n)=0
                   ENDIF
                   CP_TRA_STATE=B (Hex.)
            Start of RAMP_OPEN
            LV_NORM_PURGE_END_1=0
            CP_TRA_STATE=0 (Hex.)
ENDIF
```

RAMP_OPEN ---> MIN_PURGE

```
IF          REL_FLOW_CPS(n-1)-IP_REL_FLOW_MIN_CPS
            >IP_REL_FLOW_DEC_CP
THEN        REL_FLOW_CPS(n)
            =REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
            CP_TRA_STATE=4 (Hex.)
ELSEIF      REL_FLOW_CPS(n-1)-IP_REL_FLOW_MIN_CPS
            <-IP_REL_FLOW_INC_CP
THEN        REL_FLOW_CPS(n)
            =REL_FLOW_CPS(n-1)+IP_REL_FLOW_INC_CP
            CP_TRA_STATE=4 (Hex.)
ELSE        REL_FLOW_CPS(n)=IP_REL_FLOW_MIN_CPS
            CP_TRA_STATE=0 (Hex.)
```

```
ENDIF



NO_PURGE ---> MAX_PURGE
IF        CP_NORM_DEAC_STATE=2 (Hex.)
THEN      REL_FLOW_CPS(n)=REL_FLOW_CPS_DI
          CP_TRA_STATE=0 (Hex.)
ELSE      Start of RAMP_OPEN operation
          CP_TRA_STATE=0 (Hex.)
ENDIF



MAX_PURGE ---> NO_PURGE
IF        LV_FCUT_CP=1
or        PRS_CPS<C_PRS_MAX_1_CPS
or        LV_RAMP_CLOSE_END=1
or        LV_CP_SET_CLOSE=1
or        LV_ACT_DIAGCP=1
or        LV_ACT_DIAGCPS=1
or        TI_LAM> IP_TI_LAM_MAX_CP__TIA
or        TI_LAM<C_TI_LAM_MIN_CP
or        LV_LAM_LIM_MAX_CYCNR=1
or        IP_REL_FLOW_MAX_CPS=0
or        IP_FUEL_FLOW_MAX_CP=0
or        LV_LAM_LIM_MAX=1
or        LV_LAM_LIM_MIN=1
or        LV_OBD_AUTH_CP_CLOSE = 1
THEN      REL_FLOW_CPS(n)=0
          CP_TRA_STATE=0 (Hex.)
ELSEIF    REL_FLOW_CPS(n-1)>IP_REL_FLOW_DEC_CP
THEN      REL_FLOW_CPS(n)=MIN(REL_FLOW_CPS(n-1) ; IP_REL_FLOW_MAX_CPS)
                                     -IP_REL_FLOW_DEC_CP
          CP_TRA_STATE=5 (Hex.)
ELSE      REL_FLOW_CPS(n)=0
          CP_TRA_STATE=0 (Hex.)
ENDIF

MIN_PURGE ---> MAX_PURGE
IF        CP_NORM_DEAC_STATE=2 (Hex.)
THEN      REL_FLOW_CPS(n)=REL_FLOW_CPS_DI
          CP_TRA_STATE=0 (Hex.)
ELSE      FOR   t=0,C_T_WAIT_RAMP_OPEN
                Activation of WAIT_RAMP_OPEN_PURGE
                IF      REL_FLOW_CPS(n-1) > IP_REL_FLOW_DEC_CP
```

```
                THEN   REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
                ELSE   REL_FLOW_CPS(n)=0
                ENDIF
                CP_TRA_STATE=B (Hex.)
          Start of RAMP_OPEN
          LV_NORM_PURGE_END_1=0
          CP_TRA_STATE=0 (Hex.)
ENDIF
```

MAX_PURGE ---> MIN_PURGE

```
IF        REL_FLOW_CPS(n-1)-IP_REL_FLOW_MIN_CPS > IP_REL_FLOW_DEC_CP
THEN      REL_FLOW_CPS(n) = MIN(REL_FLOW_CPS(n-1), IP_REL_FLOW_MAX_CPS)
            -IP_REL_FLOW_DEC_CP
          CP_TRA_STATE=6 (Hex.)
ELSEIF    REL_FLOW_CPS(n-1) - IP_REL_FLOW_MIN_CPS < -IP_REL_FLOW_INC_CP
THEN      REL_FLOW_CPS(n) =REL_FLOW_CPS(n-1)+IP_REL_FLOW_INC_CP
          CP_TRA_STATE=6 (Hex.)
ELSE      REL_FLOW_CPS(n) = IP_REL_FLOW_MIN_CPS
          CP_TRA_STATE=0 (Hex.)
ENDIF
```

RAMP_CLOSE ---> NO_PURGE

```
IF        LV_FCUT_CP=1
or        PRS_CPS<C_PRS_MAX_1_CPS
or        LV_RAMP_CLOSE_END=1
or        LV_CP_SET_CLOSE=1
or        LV_ACT_DIAGCP=1
or        LV_ACT_DIAGCPS=1
or        TI_LAM> IP_TI_LAM_MAX_CP__TIA
or        TI_LAM<C_TI_LAM_MIN_CP
or        LV_LAM_LIM_MAX_CYCNR=1
or        IP_REL_FLOW_MAX_CPS=0
or        IP_FUEL_FLOW_MAX_CP=0
or        LV_LAM_LIM_MAX=1
or        LV_LAM_LIM_MIN=1
or        LV_OBD_AUTH_CP_CLOSE = 1
THEN      REL_FLOW_CPS(n)=0
          CP_TRA_STATE=0 (Hex.)
ELSEIF    REL_FLOW_CPS(n-1)> IP_REL_FLOW_DEC_CP
THEN      REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
          CP_TRA_STATE=7 (Hex.)
ELSE      REL_FLOW_CPS(n)=0,  Settings for CPS-Diagnosis, CP_TRA_STATE=0 (Hex.)
ENDIF
```

RAMP_CLOSE ---> MIN_PURGE

IF REL_FLOW_CPS(n-1)-IP_REL_FLOW_MIN_CPS > IP_REL_FLOW_DEC_CP
THEN REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)-IP_REL_FLOW_DEC_CP
CP_TRA_STATE=8 (Hex.)
ELSEIF REL_FLOW_CPS(n-1)-IP_REL_FLOW_MIN_CPS < -IP_REL_FLOW_INC_CP
THEN REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)+IP_REL_FLOW_INC_CP
CP_TRA_STATE=8 (Hex.)
ELSE REL_FLOW_CPS(n) = IP_REL_FLOW_MIN_CPS
CP_TRA_STATE=0 (Hex.)
ENDIF

WAIT_RAMP_OPEN_PURGE ---> MIN_PURGE
IF IP_REL_FLOW_MIN_CPS(n)- REL_FLOW_CPS(n-1) > IP_REL_FLOW_INC_CP
THEN REL_FLOW_CPS(n) = REL_FLOW_CPS(n-1)+IP_REL_FLOW_INC_CP
CP_TRA_STATE=C (Hex.)
ELSE REL_FLOW_CPS(n) = IP_REL_FLOW_MIN_CPS(n)
CP_TRA_STATE=0 (Hex.)
ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CL_MAX_CP	1	0H...FFFFH	0...8	0.000122	-
Maximum limit for canister load CL					
C_CL_MIN_CP	1	0H...FFFFH	0...8	0.000122	-
Minimum canister load to stop NORMAL_PURGE_MAX_PURGE					
C_CL_ST_CP	1	0H...FFFFH	0...8	0.000122	-
Initialization value for canister load CL					
C_FUEL_FAC_CP	1	0H...FFFFH	0...256	0.0039	-
Fuel constant					
IP_LAM_0_CP_MAX__TIA	2	0...FFFFh	-50...50	0.0015	%
LDPM_TIA_IP_TI_LAM_MAX_CP	2	0...FEh	-48...142.5	0.75	°C
max. limitation value for LAM_MV_i for start of RAMP_OPEN					
C_LAM_0_CP_MIN	1	8000H...7FFFH	-50...50	0.0015	%
min. limitation value for LAM_MV_i for start of RAMP_OPEN					
C_LAM_COR_MAX_CP	1	8000H...7FFFH	-50...50	0.0015	%
max. correction value for LAM_DIF_CP window					
C_LAM_COR_MIN_CP	1	8000H...7FFFH	-50...50	0.0015	%
min. correction value for LAM_DIF_CP window					
C_LAM_DIF_MAX_CP	1	8000H...7FFFH	-50...50	0.0015	%
Maximum lambda deviation for injection time correction during RAMP_OPEN					
C_LAM_P_CTR_MAX_CP	1	0H...FFH	0...255	1	-
Number of P-jumps of lambda controller for transition RAMP_OPEN->MAX_PURGE					
C_MAF_CYL_DLY_CRLC	1	0H...FFFFH	0...1	0.000015	-
Correlation constant for MAF_CYL_DLY-filter					
C_N_CP_BOL	1	0H...FFH	0...8160	32	rpm
Minimum speed limit for activation of the RAMP_OPEN operation					
C_N_CP_TOL	1	0H...FFH	0...8160	32	rpm
Maximum speed limit for activation of the RAMP_OPEN operation					
C_PRS_MAX_1_CPS	1	8000H..7FFFH	-2717...2717	0.083	hPa
Pressure difference limit (environment-intake manifold) for deactivation of the canister purge					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PRS_MAX_2_CPS	1	8000H..7FFFH	-2717...2717	0.083	hPa
Pressure difference limit (environment-intake manifold) for reactivation of the canister purge					
C_PRS_MAX_3_CPS	1	8000H..7FFFH	-2717...2717	0.083	hPa
Pressure difference limit (environment-intake manifold) for CL_MMV calculation (if CL > CL_MMV)					
C_PRS_MAX_4_CPS	1	8000H..7FFFH	-2717...2717	0.083	hPa
Pressure difference limit (environment-intake manifold) for CL_MMV calculation (if CL < CL_MMV)					
C_T_DI_MAX_CP	1	0H...FFFFH	0...6553.5	0.1	sec
Maximum time for disabling MAX_PURGE operation					
C_T_DI_RAMP_OPEN_CP	1	0H...FFFFH	0...6553.5	0.1	sec
Maximum time for disabling RAMP_OPEN operation					
C_T_WAIT_RAMP_OPEN	1	0H...FFFFH	0...6553.5	0.1	sec
Time for CPS closing for transition MIN_PURGE-->NORMAL_PURGE_RAMP_OPEN					
C_T_DLY_CP_CL	1	0H...FFH	0...25.5	0.1	s
Additional time delay for CL an CL_MMV calculation after recovery into RAMP_OPEN					
C_TIA_MIN_CP	1	0H...FEH	-48...142.5	0.75	°C
Minimum intake air temperature for closing of the CPS					
IP_TI_LAM_MAX_CP__TIA	2	0...FFFFh	-50...50	0.0015	%
LDPM_TIA_IP_TI_LAM_MAX_CP	2	0...FEh	-48...142.5	0.75	°C
Maximum threshold for TI_LAM value to activate NO_PURGE					
C_TI_LAM_MIN_CP	1	8000H...7FFFH	-50...50	0.0015	%
Minimum threshold for TI_LAM value to activate NO_PURGE					
IP_CL_CRLC__N_32__MAF	8*8	0H...FFFFH	0...1	0.000015	-
LDPM_N_32_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...8160	32	rpm
LDPM_MAF_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...1389	5.447	mg/stk
Correlation constant for CL-filter					
IP_CL_CRLC_RAMP_OPEN__N_32__MAF	8*8	0H...FFFFH	0...1	0.000015	-
LDPM_N_32_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...8160	32	rpm
LDPM_MAF_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...1389	5.447	mg/stk
Correlation constant for CL-filter during RAMP_OPEN					
IP_FUEL_FLOW_MAX_CP__N_32__MAF	8*8	0H...FFFFH	-8...8	0.000244	kg/h
LDPM_N_32_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...8160	32	rpm
LDPM_MAF_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...1389	5.447	mg/stk
Maximum fuel flow through the CPS					
IP_MAF_CP_BOL__N_32	1*6	0H...FFH	0...1389	5.447	mg/stk
LDPM_N_32_IP_MAF_CP_BOL	6	0H...FFH	0...8160	32	rpm
Minimum mass air flow limit for activation of the RAMP_OPEN operation and CL_MMV calculation					
IP_MAF_CP_TOL__N_32	1*6	0H...FFH	0...1389	5.447	mg/stk
LDPM_N_32_IP_MAF_CP_BOL	6	0H...FFH	0...8160	32	rpm
Maximum mass air flow limit for activation of the RAMP_OPEN operation and CL_MMV calculation					
IP_REL_FLOW_DEC_CP__MAF_CYL	1*6	0H...FFFFH	0...1	0.000015	-
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Decrement value for ramp decrease of REL_FLOW_CPS					
IP_REL_FLOW_FAC_CP__CL_MMV	1*6	0H...FFFFH	0...2	0.000030	-
LDP_CL_MMV_IP_REL_FLOW_FAC	6	0H...FFFFH	0...8	0.000122	-
Correction factor for ramp increase of REL_FLOW_CPS					
IP_REL_FLOW_INC_CP__MAF_CYL	1*6	0H...FFFFH	0...1	0.000015	-
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Increment value for ramp increase of REL_FLOW_CPS					
IP_REL_FLOW_MAX_CPS__N_32__MAF	8*8	0H...FFFFH	0...1	0.000015	-
LDPM_N_32_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...8160	32	rpm
LDPM_MAF_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...1389	5.447	mg/stk
Maximum limit for REL_FLOW_CPS value					
IP_REL_FLOW_MIN_CPS__N_32__MAF	8*8	0H...FFFFH	0...1	0.000015	-
LDPM_N_32_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...8160	32	rpm
LDPM_MAF_IP_REL_FLOW_MIN_CPS	8	0H...FFH	0...1389	5.447	mg/stk
Map for REL_FLOW_CPS during MIN_PURGE operation					
IP_T_DLY_CP__MAF_CYL	1*6	0H...FFFFH	0...6.4	0.0001	sec
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Time delay for MAF_FG_CYL after recovery into RAMP_OPEN (delay time in seconds)					
IP_TI_LAM_CP_CRLC__MAF_DLY_MMV	1*6	0H...FFFFH	0...1	0.000015	-

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LDP_MAF_DLY_MMV_IP_TI_LAM_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Correlation constant for TI_LAM_CP-filter					
LC_CONF_CP_TRA_DIAGCP	1	0H...1H	0...1	1	-
Configuration for purge slow opening after EVAP diagnosis abortion					
C_T_DLY_4_DIAGCP	1	0H...FFFFH	0...6553.5	0.1	sec
Delay time for purge slow opening after EVAP diagnosis abortion					
C_DTP_MIN_CP_ACT	-	8000…7FFFH	-40.96…40.96	0.00125	hPa
Minimum pressure threshold to activate purge slow opening after EVAP diagnosis abortion					
IP_REL_FLOW_INC_CP_TRA__MAF_CYL	1*6	0H...FFFFH	0...1	0.000015	-
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Increment value for ramp increase of REL_FLOW_CPS after EVAP diagnosis abortion					
C_DTP_MIN_CP_ACT_HYS	-	8000…7FFFH	-40.96…40.96	0.00125	hPa
Minimum pressure threshold hysterisis to activate purge slow opening after EVAP diagnosis abortion					
C_MAF_CPS_DLY_2_MMV_MIN	-	0H…FFFFH	0…8	0.000122	kg/h
Minimum MAF_CPS_DLY_2_MMV threhold to calculate CL and CL_MMV					
C_TIA_MAX_MIN_PURGE	1	0H...FEH	-48...142.5	0.75	°C
Maximun intake air temperature of normal idle soaking for canister purge : high TIA -> hot idle soaking					
C_TCO_MAX_MIN_PURGE	1	0H...FEH	-48...142.5	0.75	°C
Maximun coolant temperature of normal idle soaking for canister purge : high TCO -> hot idle soaking					
C_TQ_DIF_I_IS_MIN_PURGE	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum TQ_DIF_I_IS threshold of MIN_PURGE activation for long hot idle soaking					
C_T_MIN_CP_MIN_PURGE	1	0H...FFFFH	0...6553.5	0.1	sec
Duration time of MIN_PURGE for long hot idle soaking canister purge					
C_T_IS_AST_MIN_PURGE	1	0H...FFFFH	0...65535	1	sec
Idle speed activation time since engine start of normal idle soaking for canister purge					
C_ACP_OFS_MIN_PURGE	1	0H...FFH	0...510	2	PSI
ACP offset condition for long hot idle soaking canister purge					
C_CL_MMV_MIN_PURGE	1	0H...FFFFH	0...8	0.000122	-
CL_MMV condition for long hot idle soaking canister purge					
C_T_CL_ST_CP	1	0H...FFFFH	0...6553.5	0.1	sec
CL Initialization determination time value based on T_CP_AD_1					

9.17.4 Diagrams of Outputs (Examples)

The Figures 1-3 show the signals of

-the relative flow REL_FLOW_CPS through the CPS

-the corrected PWM signal CPPWM_CPS for the CPS opening

-the Lambda-Controller output TI_LAM

-lambda deviation with active canister purge LAM_DIF_CP

-calculated fuel flow through the CPS FUEL_FLOW_CP

-additive adaptive fuel flow FUEL_FLOW_ADD_AD_CP for injection time correction

-additive adaptive injection correction

-calculated canister load CL_MMV

-relative fuel flow REL_FUEL_FLOW_CP through the CPS

as a function of time for different operating conditions.

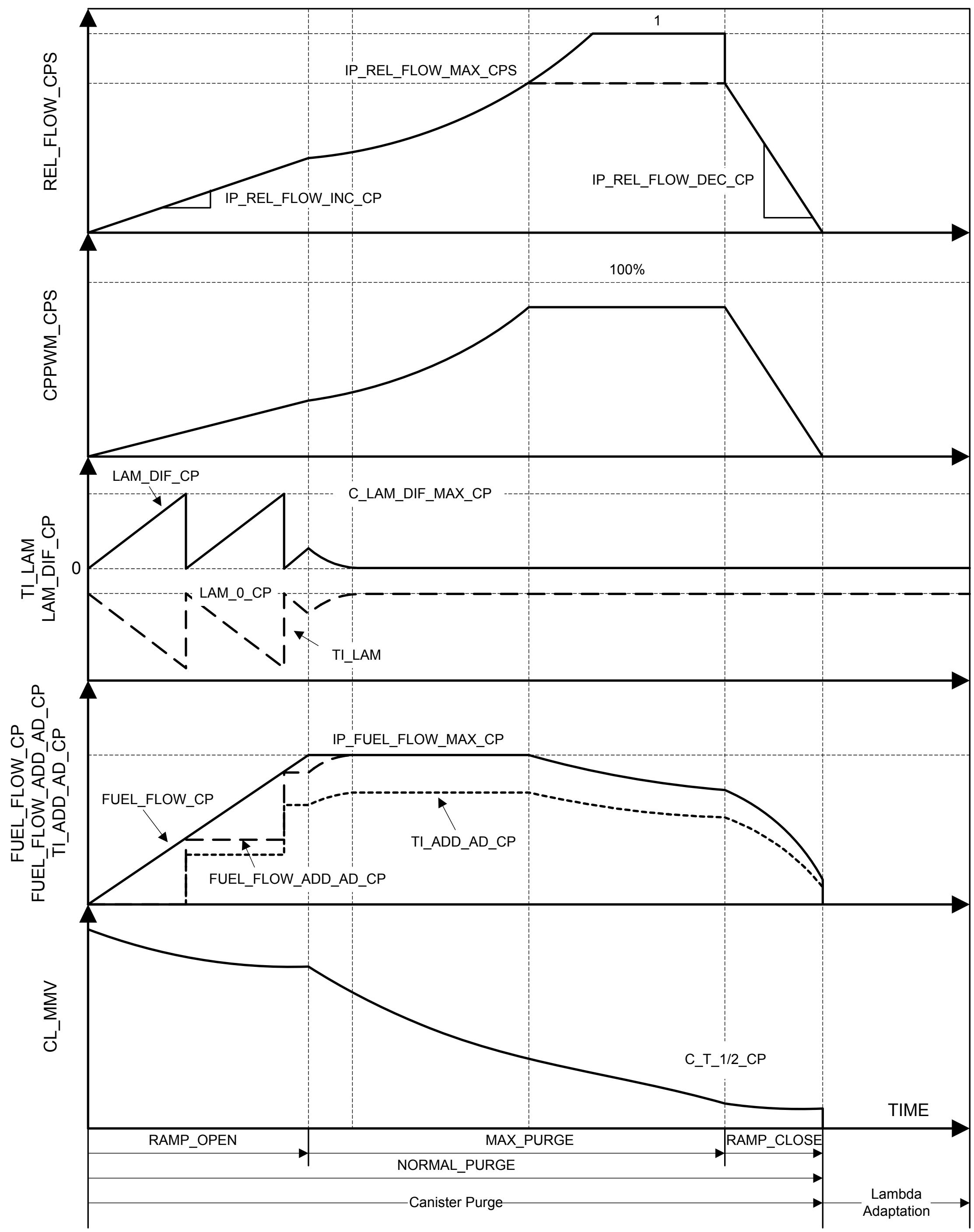


Fig.1 shows an example of operation without interruption.

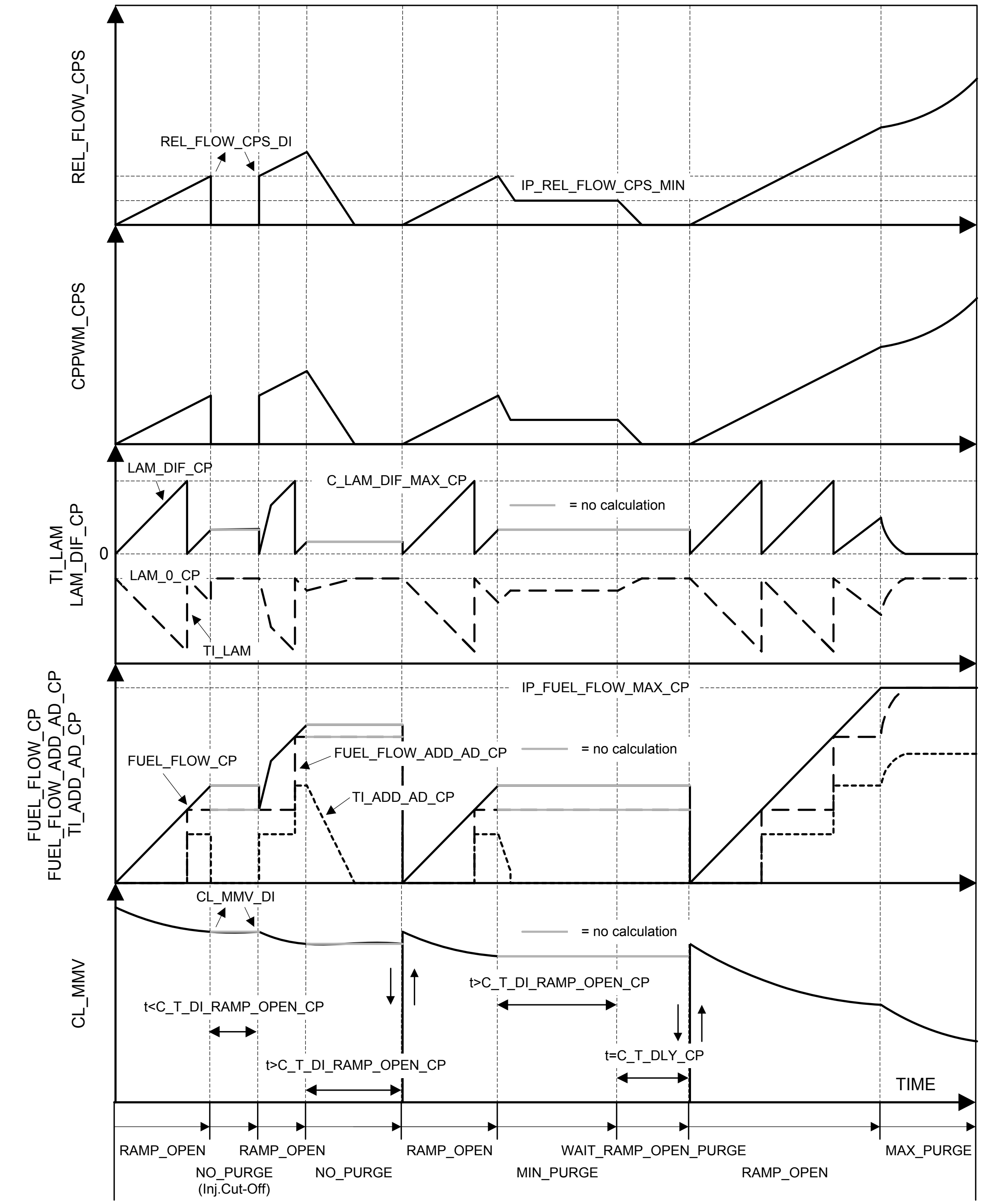


Fig.2 shows an example of RAMP_OPEN operation with some interruptions and transitions to NO_PURGE and MIN_PURGE operation.

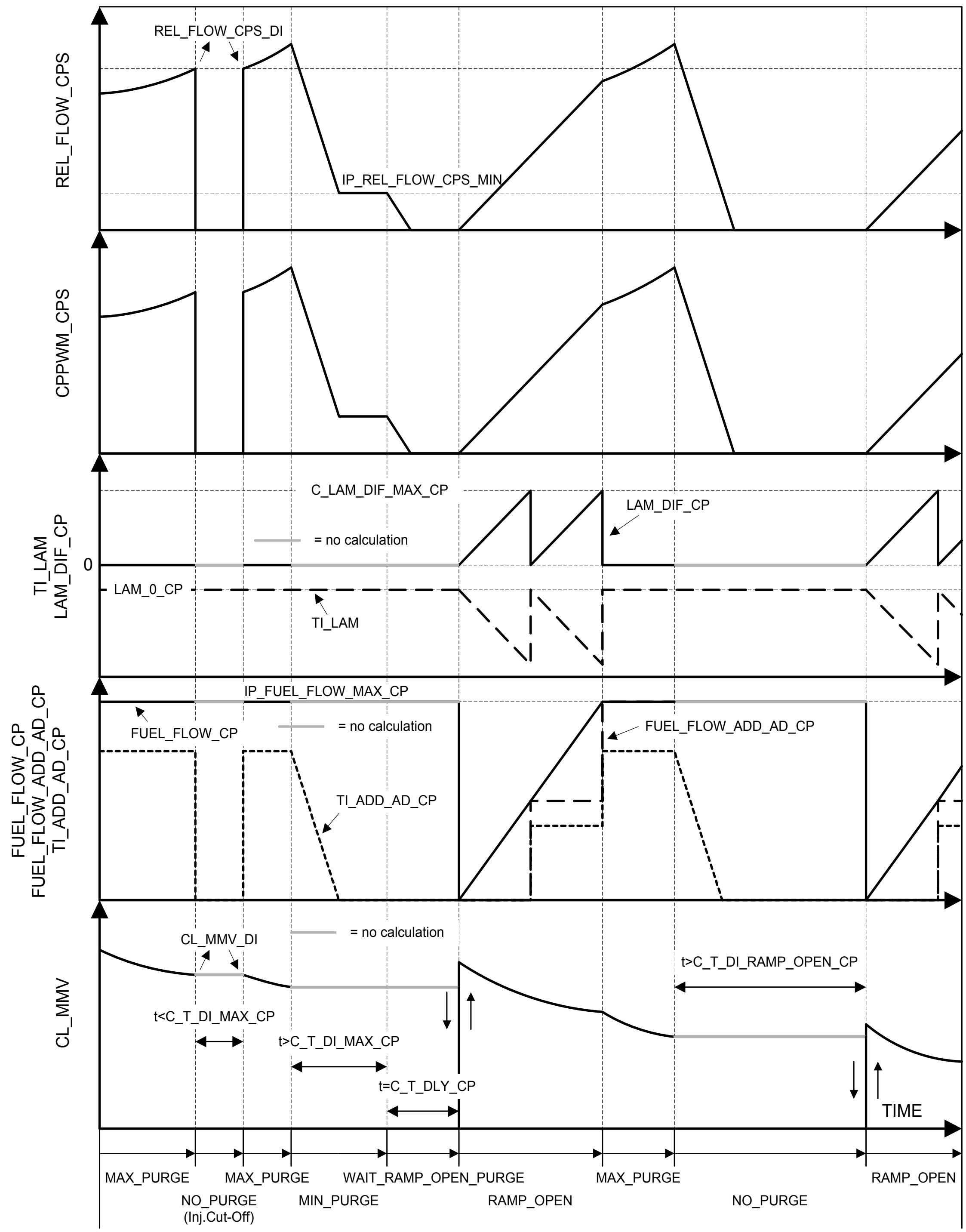


Fig.3 shows an example of MAX_PURGE operation with some interruptions and transitions to NO_PURGE and MIN_PURGE operation.

9.18 Application incidences for the Evaporative Emission Control

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_OBD_AUTH_CP_CLOSE	V/O	0...1H	0...1	1	[-]
Logical value for NO_PURGE authorization by projectspecific conditions					
LV_OBD_AUTH_CP_MIN	V/O	0...1H	0...1	1	[-]
Logical value for MIN_PURGE authorization by projectspecific conditions					
LV_LDC_CP	V/O	0...1H	0...1	1	[-]
Logical value for limited dynamic conditions for CP					
LV_FCUT_CP	O	0...1H	0...1	1	[-]
Logical value for fuel cut off					
LV_ACT_DIAGCP	O	0...1H	0...1	1	-
Only for interface reasons: LAM-P-jump delay time of the catalyst purge function					
LV_ACT_DIAGCPS	O	0...1H	0...1	1	-
Flag indicating CP not active and DIAGCP active (1), or the opposite (0)					
LV_LAM_LIM_MAX	O	0...1H	0...1	1	-
Upper threshold of controller output exceeded					
LV_LAM_LIM_MIN	O	0...1H	0...1	1	-
Bottom threshold of controller output exceeded					
LV_CP_RAMP_OPEN_STOP	V/O	0...1H	0...1	1	[-]
Boolean which indicates that the opening ramp of CPS must be stopped until lean mixture is detected					
LV_CLOSE_ACT_CP	V/O	0...1H	0...1	1	[-]
Logical value for CPS valve closed					
LV_RST_NORM_PURGE_END	O	0...1H	0...1	1	-
Bit signalizing high vapour generation or high lambda deviation at evacuation phase					
T_CP_TI_AD_SP	V/O	0..FFFFH	0..6553.5	0.1	s
Time duration of lambda adaption time phase					
T_MIN_CP	V/O	0..FFFFH	0..6553.5	0.1	s
Minimum time duration of canister purge time phase					
T_AST_DC	O	0...FFFFH	0...6553.5	0.1	[s]
Time after first start of driving cycle (RSG-variant)					
MAF_TOT_CP_DLY_MMV	O	0...FFFF H	0...2047.97	0.03125	kg/h
Moving mean value of MAF_CYL_DLY					
TI_LAM	O	8000...7FFF H	-50...49.9992%	0.0015	%
Lambda controller output					
STATE_CP	O	0H 1H 2H 3H 4H 5H 6H	CP_NOT_ACT NO_PURGE MIN_PURGE RAMP_OPEN MAX_PURGE RAMP_CLOSE WAIT_RAMP_ OPEN	-	-
State of Evaporative Emission Control Function					
LV_LAM_ADJ_CP	V/O	0...1H	0...1	1	[-]
Logical value for LAM_COR correction during RAMP_OPEN operation					
LAM_ADJ_CP	V/O	8000...7FFFH	-50...49.9992	0.0015	%
Lambda shift value for lin. Lambda controller					
LV_MFF_AD_ENA	V/O	0...1H	0...1	1	[-]
Logical value for enabled lambda adaptation (lin.)					
LV_LAM_THD_CP	V/O	0...1H	0...1	1	[-]
Interface to enable special lambda controller limits for canister purge					
LV_LAM_LIM_MAX_CYCNR	O	0...1H	0...1	1	[-]
Interface					
NC_CPPWM_CPS_TOL	O	0...7FFFH	0...99.99694	3.0518e-3	[%]
Top limit of CPPWM_CPS					
NC_CPPWM_CPS_BOL	O	0...7FFFH	0...99.99694	3.0518e-3	[%]
Bottom limit of CPPWM_CPS					
LV_REL_FLOW_MIN_CPS_S_ACT	V/O	0...1H	0...1	1	[-]

Logical value for activation of stratified min purge map					
REL_FLOW_MIN_CPS_S	V/O	0...FFFFH	0...1	0.0153e-3	[-]
Relative mass air flow through canister purge solenoid during startified mode					

Input data:

LV_IND_FCUT	LV_ERR_CPS	LV_ERR_MEC_CPS	TI_LAM
LV_ERR_LAM_LIM_i	LV_ERR_SOV	LV_ERR_MEC_SOV	CP_STATE
LV_DIAGCPS_ACT	LV_ERR_TIA	LV_ERR_TCO	LV_ERR_MAF
LV_ERR_TPS	LV_ERR_MTC_DR	LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_IV[NC_CYL_NR]
LV_ERR_IGC	MAF	FAC_LAM_DIAGCP	TI_LAM_CP
LV_FAC_LAM_DIAGCP	LV_AFL[NC_CBK_EX_NR]	LV_TI_LAM_COR_CP	TCO_ST
T_AST	LV_DC	TIA	LV_LAM_LIM_MAX_i
LV_LAM_LIM_MIN	FAC_LAM_LIM	MAF_DLY_MMV	CL_MMV
LV_TI_AD_CP_INH	LV_STATE_RST_CLC_END	LV_TI_1_HOM_MIN	LV_DIAGCP_ACT
FAC_LAM_MV_MMV_CP[NC_CBK_EX_NR]	LV_ERR_MAP	LV_CPS_INH_REQ_CAM_OFS_AD	

Application conditions:

Initialisation: -

Recurrence: 20ms

Activation: at every engine state

Formula section:

The following conditions must be fulfilled for activation of **NO_PURGE** and setting of the flag LV_OBD_AUTH_CP_CLOSE:

If LV_ERR_CPS = 1 (diagnosis error CPS)
or LV_ERR_MEC_CPS = 1 (diagnosis error CPS)
or LV_ERR_SOV = 1 (diagnosis error SOV)
or LV_ERR_MEC_SOV = 1 (diagnosis error SOV)
or LV_ERR_LAM_LIM_i = 1 (diagnosis error lambda control)
or LV_CPS_INH_REQ_CAM_OFS_AD = 1 (Camshaft offset adaptation active)
then LV_OBD_AUTH_CP_CLOSE = 1
else LV_OBD_AUTH_CP_CLOSE = 0

For activation of **MIN_PURGE** and setting of the flag LV_OBD_AUTH_CP_MIN the following conditions must be fulfilled:

If LV_ERR_TIA = 1 (air temperature sensor)
or LV_ERR_TCO = 1 (coolant temperature sensor)
or LV_ERR_MAF = 1 (mass air flow sensor)
or LV_ERR_MAP = 1 (intake manifold pressure sensor)
or LV_ERR_TPS = 1 (throttle position sensor)
or LV_ERR_MTC_DR = 1 (throttle position controller/actuator)
or LV_ERR_LS_UP = 1 (lambda sensor output voltage)
or LV_ERR_IV[NC_CYL_NR] = 1 (injector output)
or LV_ERR_IGC = 1 (ignition output)
then LV_OBD_AUTH_CP_MIN = 1
else LV_OBD_AUTH_CP_MIN = 0

9.18.1 Definition of REL_FLOW_CPS during stratified mode in MIN_PURGE

LV_REL_FLOW_MIN_CPS_S_ACT = 0
REL_FLOW_MIN_CPS_S = 0

9.18.2 Project specific bit conversion for CP function

- Limited dynamic condition for canister purge:

  if **|** MAF_{n-2} **-** MAF_n **| <** C_MAF_DYW_CP
  then LV_LDC_CP = 1
  else LV_LDC_CP = 0

- Special lambda control limits during canister purge:

  if CP_STATE **=** RAMP_OPEN **or** MAX_PURGE **or** WAIT_RAMP_OPEN
  then LV_LAM_THD_CP **=** 1
  else LV_LAM_THD_CP **=** 0

If there is no purging (LV_LAM_THD_CP = 0) the lambda controller output values (P,I,D-parts) are limited by the constants: C_LAM_P_MIN/MAX, C_LAM_I_I_MIN/MAX, C_LAM_D_MIN/MAX and C_LAM_OUT_MIN/MAX.
In case of purging (LV_LAM_THD_CP = 1) the lambda controller output values are limited by: C_LAM_P_MIN/MAX_CP, C_LAM_I_I_MIN/MAX_CP and C_LAM_OUT_MIN/MAX_CP

- Detection of fuel cut off for canister purge:

  LV_FCUT_CP **=** LV_IND_FCUT

- Lambda controler limitation activ for some cyles

  LV_LAM_LIM_MAX_CYCNR **=** 0

- Boolean for Lambda shift value of Lambda controller:

  Before the lambda control calculation is done the following bit has to be set:

  if LV_TI_LAM_COR_CP 0 → 1

  then LV_LAM_ADJ_CP **=** 1

- Lambda shift value for Lambda controller:

  if LV_FAC_LAM_DIAGCP **=** 0

  then LAM_ADJ_CP **=** TI_LAM_CP

  else LAM_ADJ_CP **=** (FAC_LAM_DIAGCP **-** TI_LAM)

- After the lambda control shift is done, the following values have to be reset:

  LV_LAM_ADJ_CP = 0

  LAM_ADJ_CP = 0

- Logical values for enabled lambda adaptation

  LV_MFF_AD_ENA **=** 1

- Bit signalizing high vapour generation or high lambda deviation at evacuation phase:

  LV_RST_NORM_PURGE_END = LV_STATE_RST_CLC_END

- Moving mean value of MAF_CYL_DLY:

  MAF_TOT_CP_DLY_MMV = MAF_DLY_MMV

- State of evaporative emission control function:

  STATE_CP = CP_STATE (assigned to each other without converting the hexvalues)

  STATE_CP is normally input for aggregates and described with other hexvalues of the symbolic statenames (eg. MAX_PURGE = 0Ah; here we have MAX_PURGE = 04h). Due to software-reasons it is necessary to have only ONE definition of a symbolic name. This is no problem, because every access to STATE_CP is done by the symbolic name, and not by the corresponding hex-number.

- Time after first start of driving cycle:

  T_AST_DC = T_AST

- Lambda controller output:

  If LC_TI_LAM_SWI_MAN_ACT = 0

  then TI_LAM = FAC_LAM_LIM

  else TI_LAM = FAC_LAM_MV_MMV_CP[NC_CBK_EX_NR]

- Upper threshold of controller output exceeded:

  LV_LAM_LIM_MAX = LV_LAM_LIM_MAX_1

- Bottom threshold of controller output exceeded:

LV_LAM_LIM_MIN = LV_LAM_LIM_MIN_1

- Only for interface reasons: LAM-P-jump delay time of the catalyst purge function:

  LV_ACT_DIAGCP = LV_DIAGCP_ACT

- Flag indicating CP not active and DIAGCP active (1), or the opposite (0):

  LV_ACT_DIAGCPS = LV_DIAGCPS_ACT

Hint: The bits or data LV_FCUT_CP, LV_RST_NORM_PURGE_END, MAF_TOT_CP_ DLY_MMV, FLOW_CPS, STATE_CP, T_AST_DC, LV_LAM_LIM_MAX , LV_LAM_LIM_MIN, LV_ACT_DIAGCP and LV_ACT_DIAGCPS are not visible.
They have at all times the value of LV_IND_FCUT, LV_STATE_RST_CLC_END, MAF_DLY_MMV, CP_STATE, T_AST, LV_LAM_LIM_MAX_1, LV_LAM_LIM_MIN_1, LV_DIAGCP_ACT, and LV_DIAGCPS_ACT !

9.18.3 Limits of Canister Purge Valve Opening

The value of CPPWM_CPS is limited to meet the requirements of the CPS-diagnosis to

NC_CPPWM_CPS_BOL **=** C_CPPWM_CPS_BOL

NC_CPPWM_CPS_TOL **=** C_CPPWM_CPS_TOL

9.18.4 Stop of opening ramp of CPS valve due to detection of lean mixture

Conditions for stopping opening ramp of the CPS valve are fullfilled if the controller is stopped by injection time limitation (LV_TI_1_HOM_MIN = 1) or if LAM_COR correction during RAMP_OPEN operation is started (LV_LAM_ADJ_CP 0 → 1):

If (1) LV_TI_1_HOM_MIN = 1

or LV_LAM_ADJ_CP 0 → 1

then (1) LV_CP_RAMP_OPEN_STOP = 1

else (1) **If (2)** LV_AFL[NC_CBK_EX_NR] = 1

or CL_MMV >= C_CL_MMV_MAX_CP

then (2) LV_CP_RAMP_OPEN_STOP = 0

9.18.5 Calculation of LV_CLOSE_ACT_CP

If LV_TI_AD_CP_INH = 0

then LV_CLOSE_ACT_CP = 1

else LV_CLOSE_ACT_CP = 0

9.18.6 Variable minimum time duration for canister purge time phase

To optimize the evaporative fuel emissions during hot conditions the minimum canister purge time can be switched from C_T_MIN_CP to C_T_MIN_CP_MAX.

Application conditions:

Initialisation: *T_CP_TI_AD_SP* **=** *C_TI_T_AD_1 in every case*

at reset: **if** LV_DC = 0 **then** LV_T_MIN_CP = 0

Recurrence: *100ms*

Activation: LV_DC=1 **and** LV_T_MIN_CP=0

Dea*ctivation:* LV_DC=0 **or** LV_T_MIN_CP=1

Formula section:

if TCO_ST **<** C_TCO_ST_T_MIN_CP

or TIA **<** C_TIA_T_MIN_CP

or T_AST **>** C_T_AST_T_MIN_CP

then LV_T_MIN_CP **=** 1 (until end of DC)

selection of different time constants

if LV_T_MIN_CP **=** 1

then T_MIN_CP **=** C_T_MIN_CP (cold conditions)

else T_MIN_CP **=** C_T_MIN_CP_MAX (hot conditions)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TI_T_AD_1	1	0..FFFFH	0..6553.5	0.1	s
Standart duration of lambda adaption time phase					
C_T_MIN_CP	1	0H...FFFFH	0...6553.5	0.1	sec
Minimum duration of purge phase					
C_T_MIN_CP_MAX	1	0H...FFFFH	0...6553.5	0.1	sec
Minimum duration of purge phase for hot conditions					
C_TCO_ST_T_MIN_CP	1	0...FEH	-48...142.5	0.75	°C
Coolant start temperature threshold for short duration of purge phase					
C_TIA_T_MIN_CP	1	0...FEH	-48...142.5	0.75	°C
Intake air temperature threshold for short duration of purge phase					
C_T_AST_T_MIN_CP	1	0...FFFFH	0...6553.5	0.1	sec
Time threshold for after which short duration of purge phase is activated anyway					
C_CL_MMV_MAX_CP	1	0...FFFFH	0...8	0.000122	-
Maximum limit for canister load CL					
C_CPPWM_CPS_TOL	1	0...7FFFH	0...99.99	0.0031	%
Top limit of CPPWM_CPS: init value 98.0 %					
C_CPPWM_CPS_BOL	1	0...7FFFH	0...99.99	0.0031	%
Bottom limit of CPPWM_CPS: init value 3.0 %					
LC_TI_LAM_SWI_MAN_ACT	1	0...1 H	0...1	1	-
logical calibration data for switching of input data for output data TI_LAM					
C_MAF_DYW_CP	1	0...FFFFH	0...1389	0.0212	mg/stk
Threshold to detect MAF-dynamic for canister purge					

9.18.7 Application incidences for CPPWM frequency switch

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_CPS_FRQ_SWI	V/O	0...1H	0...1	1	-
Logical value for CPPWM-frequency depending switch between IP_CPPWM_FRQ_0 .._1					
LV_N_CND	-	0...1H	0...1	1	-
Logical value for frequency switch: engine speed condition					
LV_MAF_CND	-	0...1H	0...1	1	-
Logical value for frequency switch: MAF condition					
CPPWM_FRQ	V/O	4...FFH	4...255	1	Hz
Frequency of the canister purge solenoid pwm-signal					

Input data:

MAF_HB	N_32	LV_ACT_DIAGCPS	

General information:

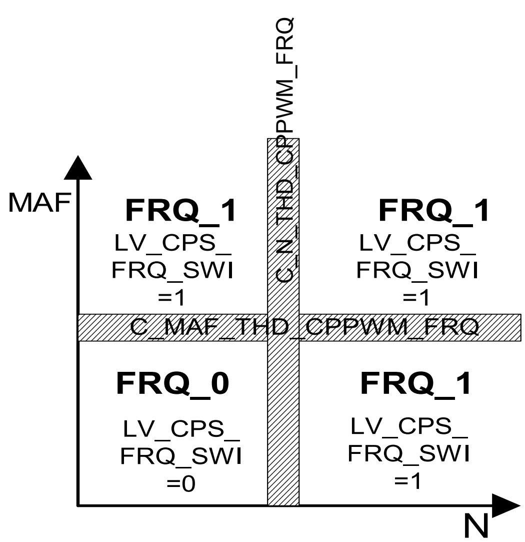
Especially in the non linear flow range area of the canister purge solenoid the minimum flow depends strongly at the PWM-frequency. In general low flow rates can be achieved better with a low PWM-frequency. But at a low PWM-frequency the cylinder to cylinder deviation can become worse.
To find the optimal compromis it is necessary to apply the frequency of the canister purge solenoid PWM-signal.
Two CPPWM-frequencies are applied depending on the actual MAF and engine speed.

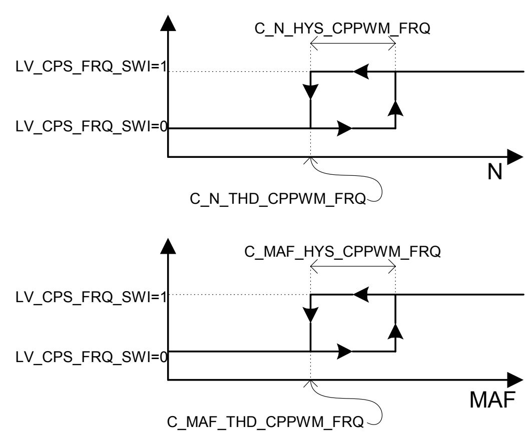
Application conditions:

Recurrence: 100ms

Activation: *at every engine state*

Signal flow diagram:





Formula section:

Calculation of CPPWM_FRQ:

If N_32 **>** **(**C_N_THD_CPPWM_FRQ **+** C_N_HYS_CPPWM_FRQ**)**
then LV_N_CND **=** 1
else **if** N_32 **<** C_N_THD_CPPWM_FRQ
then LV_N_CND **=** 0

If MAF_HB **>** **(**C_MAF_THD_CPPWM_FRQ **+** C_MAF_HYS_CPPWM_FRQ**)**
then LV_MAF_CND **=** 1
else **if** MAF_HB **<** C_MAF_THD_CPPWM_FRQ
then LV_MAF_CND **=** 0

If LV_MAF_CND **=** 1
or LV_N_CND **=** 1
or LV_ACT_DIAGCPS=1
then LV_CPS_FRQ_SWI **=** 1
CPPWM_FRQ **=** C_CPPWM_FRQ_1
else LV_CPS_FRQ_SWI **=** 0
CPPWM_FRQ **=** C_CPPWM_FRQ_0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_THD_CPPWM_FRQ	1	0..FFH	0..1389	5.447	mg/stk
Mass air flow threshold for cppwm-frequency switch					
C_MAF_HYS_CPPWM_FRQ	1	0..FFH	0..1389	5.447	mg/stk
Mass ai flow hysteresis for cppwm-frequency switch					
C_N_THD_CPPWM_FRQ	1	0..FFH	0..8160	32	Rpm
Engine speed threshold for cppwm-frequency switch					
C_N_HYS_CPPWM_FRQ	1	0..FFH	0..8160	32	Rpm
Engine speed hysteresis for cppwm-frequency switch					
C_CPPWM_FRQ_0	1	4...FFH	4...255	1	Hz
Frequency of the canister purge solenoid pwm-signal for low N and low MAF					
C_CPPWM_FRQ_1	1	4...FFH	4...255	1	Hz
Frequency of the canister purge solenoid pwm-signal for high N or high MAF					

9.19 Evaporative emission control - MFF_ADD_LAM_CP calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MFF_ADD_LAM_CP	V/O	0H...FFFFH	0...1389	0.02119	mg/stk
CP fuel mass correction (additive)					
FUEL_FLOW_ADD_AD_CP	V/O	0H...FFFFH	0...8	0.000122	kg/h
Additive adaptive fuel flow fe ACF for the injection time correction					
FUEL_FLOW_ADD_AD_DLY	V/O	0H...FFFFH	0...8	0.000122	kg/h
Filtered additive adaptive fuel flow from the ACF for the injection time correction					

Input data:

LV_CP_ENA	CP_STATE	CP_TRA_STATE	CL_MMV_OLD
REL_FLOW_CPS_OLD_MIN	MAF_FG_CYL	MAF_DLY_MMV	MAF_CPS_DLY
CL_MMV_DI	CL_MMV	MAF_CPS_DLY_2_MMV	N
MAF_CYL	LV_TI_LAM_COR_CP	NC_CYL_NR	

FUNCTION DESCRIPTION:

Application conditions:

Recurrence: 20ms

Activation:

EVAP-Control is enabled, LV_CP_ENA = 1

Deactivation:

EVAP-Control is disabled, LV_CP_ENA = 0

FUEL_FLOW_ADD_AD_DLY = 0
FUEL_FLOW_ADD_AD_CP = 0
MFF_ADD_LAM_CP = 0

General information:

9.19.1.1 Calculation of the Additive Adaptive Correction of the Injection Time MFF_ADD_LAM_CP

When the Evaporative Emission Control is active, the fuel injection time TI is corrected by an additive value MFF_ADD_LAM_CP to avoid emission and driveability problems caused by the fuel flow through the CPS. This correction is only done during NORMAL_PURGE operation.

9.19.1.1.1 Injection Time Correction during RAMP_OPEN

During RAMP_OPEN operation the injection time is corrected as soon as the deviation of the lambda control value LAM_DIF_CP is greater than an adjustable value C_LAM_DIF_MAX_CP. Then the actual calculated relative CPS flow REL_FLOW_CPS_AV and the actual moving mean value of the canister load CL_MMV are used as REL_FLOW_CPS_OLD and CL_MMV_OLD to calculate the value of the FUEL_FLOW_ADD_AD_CP and the lambda controller value TI_LAM is corrected by the value of C_LAM_DIF_MAX_CP. This procedure is repeated everytime, when the value of LAM_DIF_CP is greater than the threshold value C_LAM_DIF_MAX_CP. If the injection time was corrected and after the correction the value of LAM_DIF_CP is lower than the threshold value -C_LAM_DIF_MAX_CP the actual moving mean value of the canister load CL_MMV is used as CL_MMV_OLD to calculate the value of the FUEL_FLOW_ADD_AD_CP and the lambda controller value TI_LAM is not corrected (see also chapter Evaporative Emission Controll).

With the value of FUEL_FLOW_ADD_AD_CP and the actual speed N the additive adaptive correction of the injection mass MFF_ADD_LAM_CP is calculated. This value is added with a minus sign to the formula of the injection time (see chapter INJECTION).

When the RAMP_OPEN operation is disabled the injection time is corrected during the transition to the MIN_PURGE or NO_PURGE operation (CP_TRA_STATE=1 or 2 (Hex.)). For this the last stored value of CL_MMV_OLD and the actual value of REL_FLOW_CPS (if REL_FLOW_CPS_AV is lower than REL_FLOW_CPS_OLD) are used to calculate the value of FUEL_FLOW_ADD_AD_CP. When the transition phase is over, the injection time isn't corrected any longer.

The additive adaptive correction of the injected fuel mass FUEL_FLOW_ADD_AD_CP is also regarded in the calculation of FUEL_FLOW_CP. To consider the time delay between the mass air flow MAF_CYL and TI_LAM calculation a filtered value FUEL_FLOW_ADD_AD_DLY is calculated.

9.19.1.1.2 Injection Time Correction during MAX_PURGE and RAMP_CLOSE

During MAX_PURGE and RAMP_CLOSE operation there is a continuous additive adaptive correction of the injection time. The additive fuel flow FUEL_FLOW_ADD_AD_CP is calculated from the calculated and time delayed mass flow through the CPS MAF_CPS_DLY and the filtered canister load CL_MMV. The delay of MAF_CPS is necessary to take the gas flow time from the CPS to the injection valve into consideration.

The calculation of the value of the additive adaptive injection time correction is the same as it is used during RAMP_OPEN operation.

When the MAX_PURGE or the RAMP_CLOSE operation is disabled the injection time is corrected during the transition to the MIN_PURGE or NO_PURGE operation (CP_TRA_STATE=5,6,7 or 8(Hex.)). Therefore the last calulated value of CL_MMV_DI and the actual value of MAF_CPS_DLY are used to calculate the value of FUEL_FLOW_ADD_AD_CP. After the transition phase there is no correction of the injection time.

The additive adaptive correction of the injected fuel mass FUEL_FLOW_ADD_AD_CP is also regarded in the calculation of FUEL_FLOW_CP. To consider the time delay between the mass air flow MAF_CYL and the lamda controller calculation a filtered value FUEL_FLOW_ADD_AD_DLY is calculated (see also chapter Evaporative Emission Controll).

Formula section:

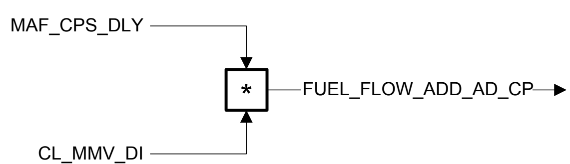
- ***Calculation of FUEL_FLOW_ADD_AD_CP AND FUEL_FLOW_ADD_AD_DLY in NO_PURGE***

```
IF      CP_STATE = NO_PURGE(1h) and  CP_TRA_STATE(n) = OPEN_TO_NO_TRA(3h)
THEN      FUEL_FLOW_ADD_AD_CP(n) = CL_MMV_OLD(n) * MAF_FG_CYL
                                        * REL_FLOW_CPS_OLD_MIN(n)
          FUEL_FLOW_ADD_AD_DLY(n) = CL_MMV_OLD(n) * MAF_DLY_MMV
                                        * REL_FLOW_CPS_OLD_MIN(n)
ELSE      IF      CP_TRA_STATE = 5(Hex.) or CP_TRA_STATE = 7(Hex.)
          THEN  FUEL_FLOW_ADD_AD_CP(n) = MAF_CPS_DLY(n) * CL_MMV_DI(n)
                FUEL_FLOW_ADD_AD_DLY(n) = FUEL_FLOW_ADD_AD_DLY(n-1)
          ELSE  FUEL_FLOW_ADD_AD_CP(n)      =  0
                FUEL_FLOW_ADD_AD_DLY(n)     =  0
          ENDIF
ENDIF
```

- ***Calculation of FUEL_FLOW_ADD_AD_CP and FUEL_FLOW_ADD_AD_DLY in MIN_PURGE***

```
IF        CP_STATE = MIN_PURGE (2h) and  CP_TRA_STATE(n) = 4(Hex.)
THEN      FUEL_FLOW_ADD_AD_CP(n) = CL_MMV_OLD(n) * MAF_FG_CYL
                                    * REL_FLOW_CPS_OLD_MIN(n)
          FUEL_FLOW_ADD_AD_DLY(n) = CL_MMV_OLD(n) * MAF_DLY_MMV
                                    * REL_FLOW_CPS_OLD_MIN(n)
ELSE      IF      CP_TRA_STATE = 6(Hex.) or CP_TRA_STATE = 8(Hex.)
          THEN  FUEL_FLOW_ADD_AD_CP(n) = MAF_CPS_DLY(n) * CL_MMV_DI(n)
                FUEL_FLOW_ADD_AD_DLY(n) = FUEL_FLOW_ADD_AD_DLY(n-1)
          ELSE  FUEL_FLOW_ADD_AD_CP(n)      =  0
                FUEL_FLOW_ADD_AD_DLY(n)     =  0
          ENDIF
ENDIF
```

Disabled MAX_PURGE and RAMP_CLOSE:



TE17

- ***Calculation of FUEL_FLOW_ADD_AD_CP AND FUEL_FLOW_ADD_AD_DLY in RAMP_OPEN***

if CP_STATE = RAMP_OPEN (3h)

then FUEL_FLOW_ADD_AD_CP(n) = CL_MMV_OLD(n) * MAF_FG_CYL * REL_FLOW_CPS_OLD_MIN(n)

FUEL_FLOW_ADD_AD_DLY(n)= CL_MMV_OLD(n) * MAF_DLY_MMV * REL_FLOW_CPS_OLD_MIN(n)

endif

if LV_TI_LAM_COR_CP = 1

then FUEL_FLOW_ADD_AD_CP and FUEL_FLOW_ADD_AD_DLY are remaining like calculated above.

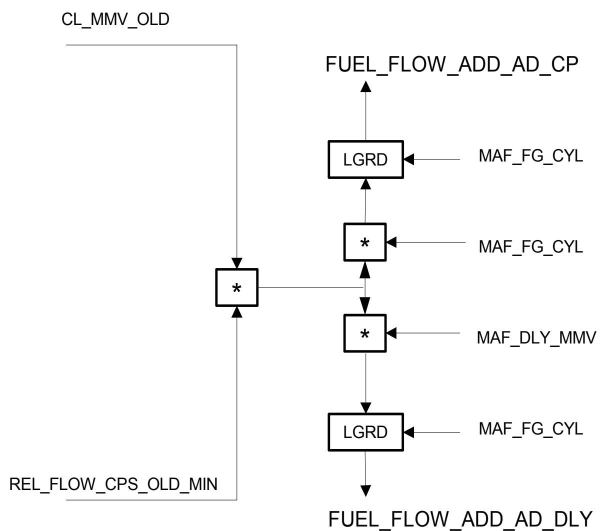
else change limitation active for FUEL_FLOW_ADD_AD_CP and FUEL_FLOW_ADD_AD_DLY (see chapter change limitation)

endif

<u>Signal flow diagram:</u>

Calculation of FUEL_FLOW_ADD_AD_CP:

RAMP_OPEN:



TE14_b3

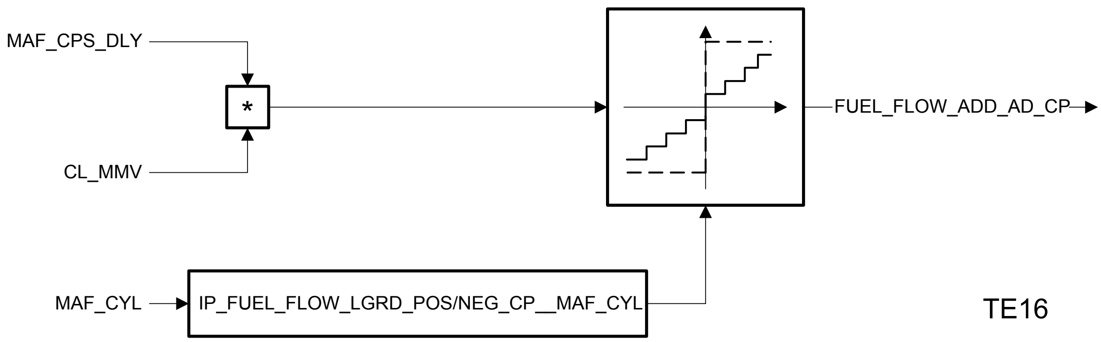
- ***Calculation of FUEL_FLOW_ADD_AD_CP AND FUEL_FLOW_ADD_AD_DLY in MAX_PURGE and RAMP_CLOSE***

if CP_STATE = MAX_PURGE (4h) or CP_STATE = 5(Hex.)
then FUEL_FLOW_ADD_AD_CP(n) = CL_MMV * MAF_CPS_DLY
FUEL_FLOW_ADD_AD_DLY(n) = CL_MMV * MAF_CPS_DLY_2_MMV
change limitation active for FUEL_FLOW_ADD_AD_CP
and FUEL_FLOW_ADD_AD_DLY (see chapter change Limitation)
endif

Signal flow diagram:

Calculation of FUEL_FLOW_ADD_AD_CP:

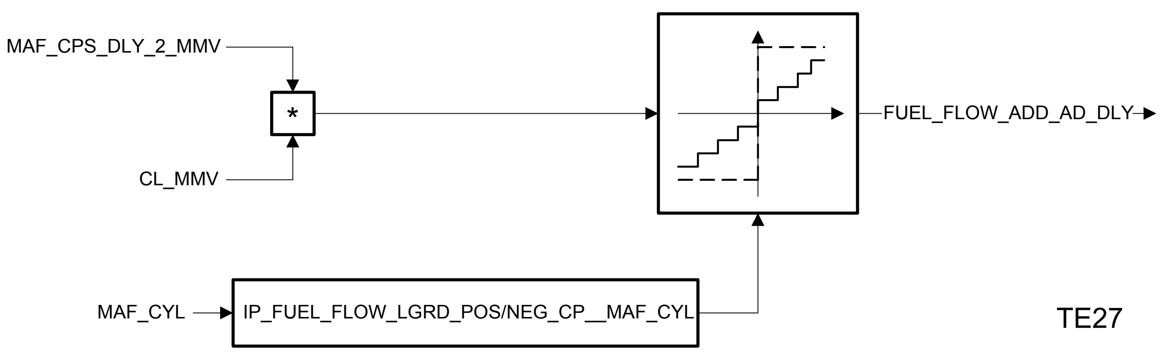
MAX_PURGE and RAMP_CLOSE:



TE16

Calculation of FUEL_FLOW_ADD_AD_DLY:

MAX_PURGE and RAMP_CLOSE:



TE27

- ***Change limitation for FUEL_FLOW_ADD_AD_CP and FUEL_FLOW_ADD_AD_DLY in RAMP_OPEN, MAX_PURGE and RAMP_CLOSE***

```
IF          FUEL_FLOW_ADD_AD_CP(n) - FUEL_FLOW_ADD_AD_CP(n-1)
                  >IP_FUEL_FLOW_LGRD_POS_CP
THEN        FUEL_FLOW_ADD_AD_CP(n) = FUEL_FLOW_ADD_AD_CP(n-1)
                                        + IP_FUEL_FLOW_LGRD_POS_CP
ELSEIF      FUEL_FLOW_ADD_AD_CP(n-1) - FUEL_FLOW_ADD_AD_CP(n)
                  > IP_FUEL_FLOW_LGRD_NEG_CP
THEN        FUEL_FLOW_ADD_AD_CP(n) = FUEL_FLOW_ADD_AD_CP(n-1)
                                        - IP_FUEL_FLOW_LGRD_NEG_CP
ELSE        FUEL_FLOW_ADD_AD_CP(n) = remains unchanged
ENDIF


IF          FUEL_FLOW_ADD_AD_DLY(n) - FUEL_FLOW_ADD_AD_DLY(n-1)
                  >IP_FUEL_FLOW_LGRD_POS_CP
THEN        FUEL_FLOW_ADD_AD_DLY(n) = FUEL_FLOW_ADD_AD_DLY(n-1)
                                        + IP_FUEL_FLOW_LGRD_POS_CP
ELSEIF      FUEL_FLOW_ADD_AD_DLY(n-1) - FUEL_FLOW_ADD_AD_DLY(n)
                  > IP_FUEL_FLOW_LGRD_NEG_CP
THEN        FUEL_FLOW_ADD_AD_CP(n) = FUEL_FLOW_ADD_AD_CP(n-1)
                                        - IP_FUEL_FLOW_LGRD_NEG_CP
ELSE        FUEL_FLOW_ADD_AD_CP(n) = remains unchanged
ENDIF
```

- ***Calculation of FUEL_FLOW_ADD_AD_CP and FUEL_FLOW_ADD_AD_DLY in WAIT_RAMP_OPEN***

```
IF          CP_STATE = WAIT_RAMP_OPEN (6h)
THEN        FUEL_FLOW_ADD_AD_CP = 0
            FUEL_FLOW_ADD_AD_DLY = 0
ENDIF
```

- ***Calculation of MFF_ADD_LAM_CP in NORMAL_PURGE***

MFF_ADD_LAM_CP = (FUEL_FLOW_ADD_AD_CP * 10^5) / (N * NC_CYL_NR * 3)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FUEL_FLOW_LGRD_NEG_CP__MAF	1*6	0...FFFFH	0...8	0.000122	kg/h
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0...FFFFH	0...2047.96875	0.03125	kg/h
Limited gradient for negative changes of FUEL_FLOW_ADD_CP					
IP_FUEL_FLOW_LGRD_POS_CP__MAF	1*6	0...FFFFH	0...8	0.000122	kg/h
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0...FFFFH	0...2047.96875	0.03125	kg/h
Limited gradient for positive changes of FUEL_FLOW_ADD_CP					

9.20 CPS control – CPPWM (two frequencys)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CPPWM	V/O	0H...7FFFH	0...99.99	0.0031	%
PWM signal for CPS opening					
CPPWM_SP	V	0H...7FFFH	0...99.99	0.0031	%
Setpoint PWM signal for CPS opening					
CPPWM_CPS	V/O	0H...7FFFH	0...99.99	0.0031	%
Corrected PWM signal for CPS opening					
FLOW_COR_CPS	V/O	0H...FFFFH	0...8	0.000122	kg/h
Corrected flow setpoint through the CPS					
FLOW_SP_CPS	V/O	0H...FFFFH	0...8	0.000122	kg/h
Flow setpoint through the CPS					
MAF_CPS	V/O	0H...FFFFH	0...8	0.000122	kg/h
Mass flow through the CPS					
MAF_CPS_DLY	V/O	0H...FFFFH	0...8	0.000122	kg/h
Filtered value of MAF_CPS (Padé Filter) - Delay CPS - inlet valve (for TI_AD_ADD_CP calculation)					
MAF_CPS_DLY_2	V	0H...FFFFH	0...8	0.000122	kg/h
Filtered value of MAF_CPS (Padé Filter) - Delay CPS - lambda sensor					
MAF_CPS_DLY_2_MMV	V/O	0H...FFFFH	0...8	0.000122	kg/h
Moving mean value of MAF_CPS_DLY_2 (for CL calculation)					
REL_FLOW_CPS_AV	V	0H...FFFFH	0...1	0.000015	-
Actual value of the relative flow (MAF_CPS/MAF_CYL) through the CPS					
REL_FLOW_CPS_DLY	V	0H...FFFFH	0...1	0.000015	-
Filtered value of REL_FLOW_CPS_AV (Padé Filter) - Delay CPS - inlet valve					
REL_FLOW_CPS_DLY_MMV	V/O	0H...FFFFH	0...1	0.000015	-
Moving mean value of REL_FLOW_CPS_DLY (for MAF_CPS_DLY calculation)					
LV_TI_AD_CP_INH	V/O	0H...1H	0...1	1	-
Logical value for enabled lambda adaptation					
PRS_CPS	V/O	8000H...7FFFH	-2717...2717	0.083	hPa
Pressure difference at CPS					

Input data:

REL_FLOW_CPS	AMP	TIA	NC_CPPWM_CPS_TOL
N_32	VB	LV_ACT_DIAGCP	NC_CPPWM_CPS_BOL
MAF_FG_CYL	CPPWM_ADD_AD	CPPWM_EXT_ADJ	CPPWM_CP_DIAGCP
MAF_CYL	LV_ACT_DIAGCPS	LV_CPPWM_EXT_ADJ	FLOW_SP_CPS_EXT
LV_CP_ENA	LV_ACT_DIAGCP		REL_FLOW_MAX_CPS
MAP	LV_CPS_FRQ_SWI		

General information:

The setpoint of the mass flow through the CPS FLOW_COR_CPS is calculated using the actual calculated values of REL_FLOW_CPS, MAF_CYL and the two correction factors IP_FAC_AMP_CP and IP_FAC_TIA_CP. These factors are used to correct the influence of the ambient pressure and the ambient temperature.

The PWM-signal to control the opening of the CPS CPPWM is calculated from the map IP_CPPWM as a function of the flow setpoint FLOW_COR_CPS and the pressure difference PRS_CPS. The influence of the battery voltage VB on the opening of the CPS is corrected by the additive value CPPWM_VB_ADD.

The opening of the CPS can also be controlled from the application system with the additive constant C_CPPWM_AS.

If the OBD II evaporative system monitoring function is active (LV_ACT_DIAGCP=1 or LV_ACT_DIAGCPS=1), the opening of the CPS is controlled by the value of CPPWM_CP_DIAGCP (see chapter OBDII functions).

If the OBD I CPS monitoring function is active (LV_CPPWM_EXT_ADJ=1), the opening of the CPS is controlled by the value of CPPWM_EXT_ADJ. In addition the flow through the CPS can be controlled with the input FLOW_SP_CPS_EXT if LV_ACT_DIAGCPS=1 (see chapter OBDI functions).

When the value of the PWM signal CPPWM is lower than 100%, the mass flow through the CPS MAF_CPS is equal to the setpoint value FLOW_SP_CPS. When the CPS is full opened, MAF_CPS is calculated from the map IP_MAF_CPS as a function of the pressure difference PRS_CPS.

The value of CPPWM_CPS is limited to fulfill the requirements of the CPS-diagnosis : NC_CPPWM_CPS_BOL < CPPWM_CPS < NC_CPPWM_CPS_TOL.

The actual value of REL_FLOW_CPS can be lower than the calculated value. Therefore REL_FLOW_CPS_AV is calculated using MAF_CPS and MAF_FG_CYL. This value is filtered using a Padé-filter and a moving mean value calculation. The filter is needed for the calculation of the MAF_CPS_DLY value. MAF_CPS_DLY is needed for the injection time correction calculation. Here the delay of MAF_CPS is needed to take the time delay between the CPS and the injection valve and mixing effects in the intake manifold into consideration.

9.20.1 Calculation of the Mass Flows MAF_CPS, MAF_CPS_DLY, MAF_CPS_DLY_2_MMV and CPPWM

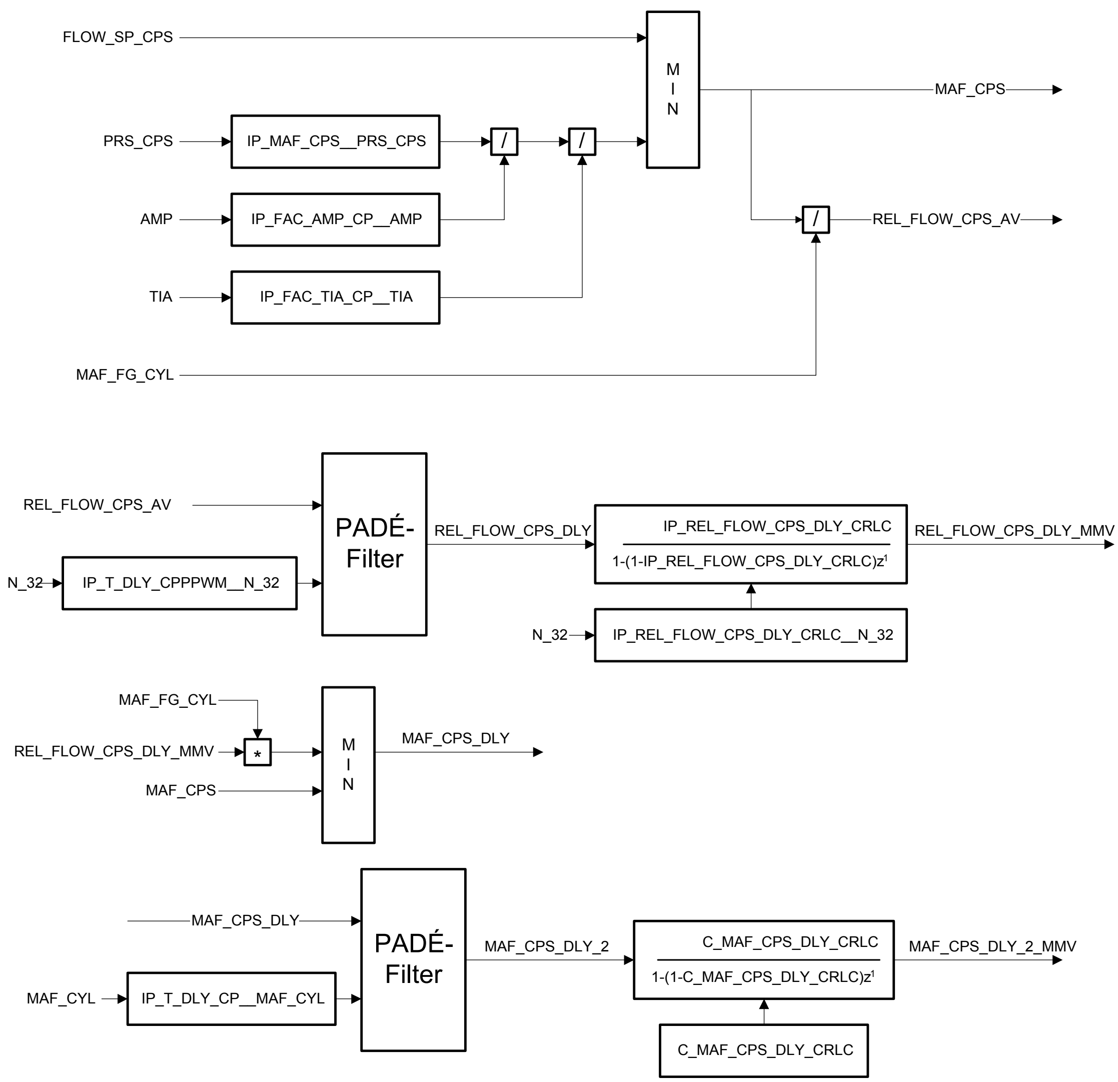
Application conditions:

Recurrence: 20 ms

Activation: LV_CP_ENA = 1

Signal flow diagram:

Calculation of MAF_CPS, REL_FLOW_CPS_DLY_MMV, MAF_CPS_DLY
and MAF_CPS_DLY_2_MMV:



TE19

Formula section:

- ***Calculation of CPPWM, REL_FLOW_CPS_AV and MAF_CPS***

```
IF    LV_CP_ENA = 1
THEN
      IF LV_ACT_DIAGCP = 0 and  ( LV_ACT_DIAGCPS=1 or REL_FLOW_CPS>0 )
      THEN  IF     LV_ACT_DIAGCPS = 1    (EVAP_DIAG function active)
            THEN   FLOW_SP_CPS=FLOW_SP_CPS_EXT
            ELSE   FLOW_SP_CPS = MIN( REL_FLOW_CPS, REL_FLOW_MAX_CPS)
                                         * MAF_FG_CYL
            ENDIF

            IF     LV_CPS_FRQ_SWI = 0
            THEN   CPPWM_SP = MIN ( IP_CPPWM_FRQ_0 , 99.99%) - CPPWM_ADD_AD
            ELSE   CPPWM_SP = MIN ( IP_CPPWM_FRQ_1 , 99.99%)
            ENDIF

            IF     FLOW_SP_CPS = 0
            THEN   CPPWM = CPPMW_SP
            ELSE   CPPWM = CPPWM_SP + IP_CPPWM_VB_ADD__VB
            ENDIF
            MAF_CPS = MIN( IP_MAF_CPS/{IP_FAC_AMP_CP * IP_FAC_TIA_CP};
                                                   FLOW_SP_CPS)
            FLOW_COR_CPS = IP_FAC_AMP_CP*IP_FAC_TIA_CP*FLOW_SP_CPS
            REL_FLOW_CPS_AV = MAF_CPS/MAF_FG_CYL

      ELSE  IF     LV_ACT_DIAGCP = 1   (only used in former EVAP_DIAG functions)
            THEN   CPPWM = CPPWM_CP_DIAGCP
            ELSE   call (Reset function of MAF_CPS_x, REL_FLOW_x and CPPWM values)
            ENDIF
      ENDIF
ELSE        call (Reset function of MAF_CPS_x, REL_FLOW_x and CPPWM values)
ENDIF
```

- **Resetfunction of MAF,REL_FLOW and CPPWM values**

CPPWM = 0
FLOW_SP_CPS = 0
FLOW_COR_CPS = 0
MAF_CPS = 0
MAF_CPS_DLY = 0
$MAF\_CPS\_DLY_{n-1}$ = 0
MAF_CPS_DLY_2 = 0
$MAF\_CPS\_DLY\_2_{n-1}$ = 0
MAF_CPS_DLY_2_MMV = 0
REL_FLOW_CPS_AV = 0
$REL\_FLOW\_CPS\_AV_{n-1}$ = 0
$REL\_FLOW\_CPS\_AV_{n-2}$ = 0
REL_FLOW_CPS_DLY = 0
$REL\_FLOW\_CPS\_DLY_{n-1}$ = 0
REL_FLOW_CPS_DLY_MMV = 0

- ***Calculation of REL_FLOW_CPS_DLY and REL_FLOW_CPS_DLY_MMV***

```
IF      LV_CP_ENA = 1           and
        LV_ACT_DIAGCP = 0       and
        (LV_ACT_DIAGCPS=1 or REL_FLOW_CPS>0)
THEN    FAC_1=IP_T_DLY_CPPWM / 0.02     #(0.02 = time recurrence)
        FAC_2=FAC_1*FAC_1/3
        REL_FLOW_CPS_DLY(n)=1/(1+FAC_1+FAC_2)
                    *[(1-FAC_1+FAC_2)
                    *(REL_FLOW_CPS_AV(n)-REL_FLOW_CPS_DLY(n-2)
                    +2*(1-FAC_2)*(REL_FLOW_CPS_AV(n-1)-REL_FLOW_CPS_DLY(n-1)]
                    +REL_FLOW_CPS_AV(n-2)

        REL_FLOW_CPS_DLY_MMV(n)=(1-IP_REL_FLOW_CPS_DLY_CRLC__N_32)
                    *REL_FLOW_CPS_DLY_MMV(n-1)
                    + IP_REL_FLOW_CPS_DLY_CRLC__N_32
                    *REL_FLOW_CPS_DLY(n)

ELSE    call (Reset function of MAF_CPS_x, REL_FLOW_x and CPPWM values)
ENDIF
```

- ***Calculation of MAF_CPS_DLY***

IF LV_CP_ENA = 1 and
LV_ACT_DIAGCP = 0 and
(LV_ACT_DIAGCPS=1 or REL_FLOW_CPS>0)
THEN MAF_CPS_DLY= **MIN (**REL_FLOW_CPS_DLY_MMV * MAF_FG_CYL **;** MAF_CPS**)**
ELSE MAF_CPS_DLY = 0
ENDIF

- ***Calculation of MAF_CPS_DLY_2 and MAF_CPS_DLY_2_MMV***

IF LV_CP_ENA = 1 and
LV_ACT_DIAGCP=0 and
(LV_ACT_DIAGCPS=1 or REL_FLOW_CPS>0)
THEN FAC_1 = IP_T_DLY_CP / 0.02 ***#(0.02 = time recurrence)***
FAC_2 = FAC_1*FAC_1/3
MAF_CPS_DLY_2(n)=1/(1+FAC_1+FAC_2)
*[(1-FAC_1+FAC_2)
*(MAF_CPS_DLY(n)-MAF_CPS_DLY_2(n-2)
+2*(1-FAC_2)*(MAF_CPS_DLY(n-1)-MAF_CPS_DLY_2(n-1)]
+MAF_CPS_DLY(n-2)
MAF_CPS_DLY_2_MMV(n)=(1-C_MAF_CPS_DLY_CRLC)
*MAF_CPS_DLY_2_MMV(n-1)
+ C_MAF_CPS_DLY_CRLC
*MAF_CPS_DLY_2(n)
ELSE call (Reset function of MAF_CPS_x, REL_FLOW_x and CPPWM values)
ENDIF

9.20.2 Calculation of CPPWM_CPS

Application conditions:

Initialisation: see end of chapter
Recurrence: 20 ms

Activation: always, LV_CP_ENA is not considered

Formula section:

- ***Calculation of CPPWM_CPS***

IF LV_CPPWM_EXT_ADJ=1
THEN CPPWM_CPS=CPPWM_EXT_ADJ
ELSE IF LV_CP_ENA = 1
THEN CPPWM_CPS=CPPWM+C_CPPWM_AS

```
          ELSE  CPPWM_CPS = NC_CPPWM_CPS_BOL
          ENDIF
ENDIF
```

- ***Limitation of CPPWM_CPS***

```
IF        CPPWM_CPS > NC_CPPWM_CPS_TOL
THEN      CPPWM_CPS = NC_CPPWM_CPS_TOL
ELSE      IF    CPPWM_CPS < NC_CPPWM_CPS_BOL
          THEN  CPPWM_CPS = NC_CPPWM_CPS_BOL
          ELSE  CPPWM_CPS = CPPWM_CPS
          ENDIF
ENDIF
```

- ***Calculation of PRS_CPS***

PRS_CPS = AMP - MAP

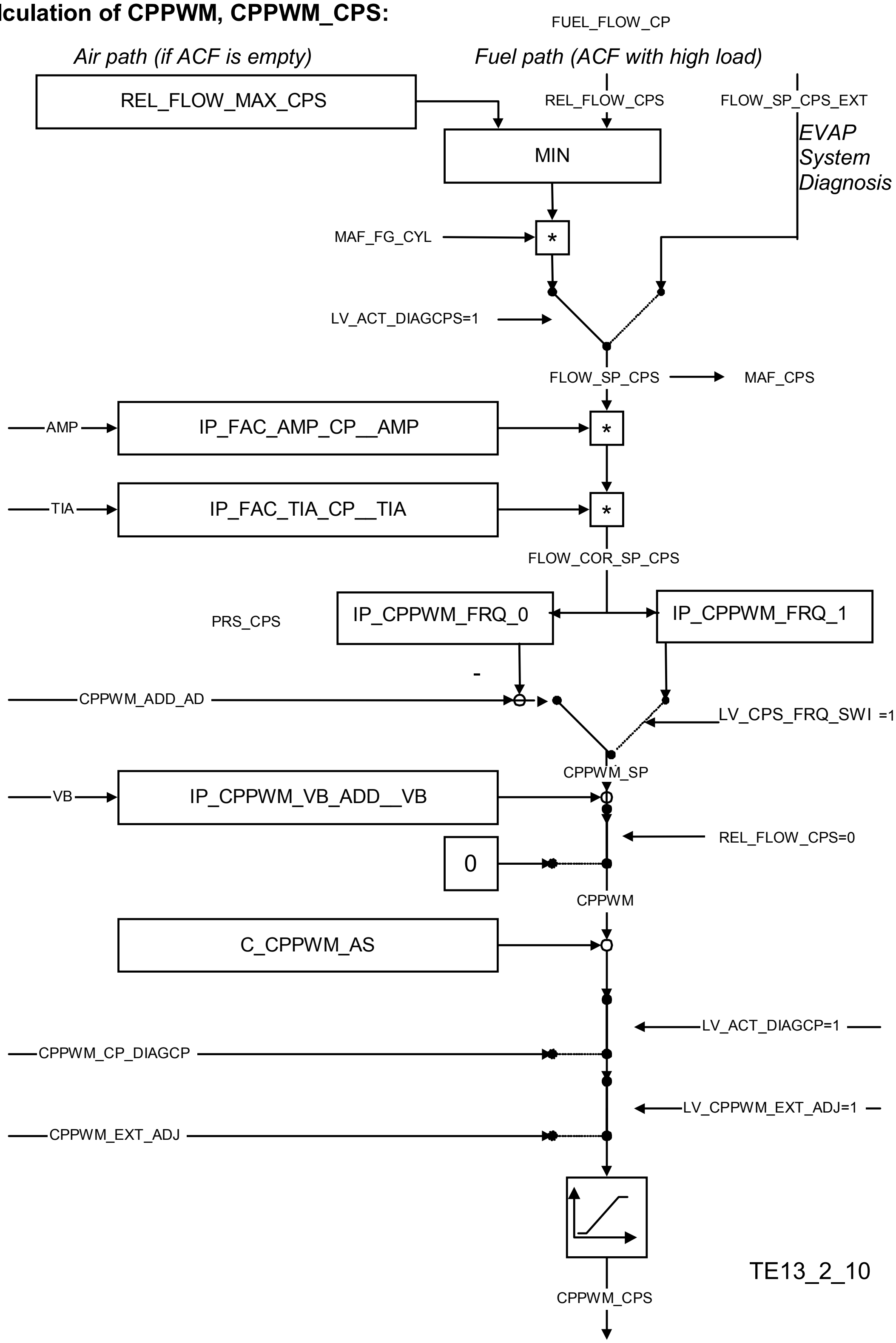
- ***Calculation of LV_TI_AD_CP_INH***

Lambda Adaptation and Evaporative Emission Control. When the Evaporative Emission Control is active, the Lambda Adaptation can be performed always, when the CPS is closed (CPPWM=0).

```
IF        CPPWM =  0
THEN      LV_TI_AD_CP_INH = 0
ELSE      LV_TI_AD_CP_INH = 1
ENDIF
```

Signal flow diagram:

Calculation of CPPWM, CPPWM_CPS:



TE13_2_10

- ***Initializations***

IF LV_CP_ENA = 0 or
LV_ACT_DIAGCP = 1 or
(LV_ACT_DIAGCPS = 0 and REL_FLOW_CPS = 0) or
THEN call (Reset function of MAF_CPS_x, REL_FLOW_x and CPPWM values)
ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CPPWM_AS	1	8000H...7FFFH	-50...49.99	1.526e-3	%
CPPWM adjustment via appliction system					
C_MAF_CPS_DLY_CRLC	1	0H...FFFFH	0...1	0.000015	-
Correlation constant for MAF_CPS_DLY-filter					
IP_CPPWM_FRQ_0	12*16	0H...FFFFH	0...199.99	0.0031	%
LDPM_FLOW_COR_CPS__CPPWM_FRQ	12	0H...FFFFH	0...8	0.00012	kg/h
LDPM_PRS_CPS_IP_CPPWM	16	0H...FFFFH	-2717...2717	0.083	hPa
Characteristic mass flow map for CPS with first CPPWM-frequency					
IP_CPPWM_FRQ_1	12*16	0H...FFFFH	0...199.99	0.0031	%
LDPM_FLOW_COR_CPS__CPPWM_FRQ	12	0H...FFFFH	0...8	0.000122	kg/h
LDPM_PRS_CPS_IP_CPPWM	16	0H...FFFFH	-2717...2717	0.083	hPa
Characteristic mass flow map for CPS with second CPPWM-frequency					
IP_CPPWM_VB_ADD__VB	1*8	0H...7FFFH	-50...49.99	0.0031	%
LDPM_VB_1	8	0H...FFH	0...26	0.102	V
Additive VB correction for CPPWM					
IP_FAC_AMP_CP__AMP	1*6	0H...FFH	0...1.992	0.0078	-
LDP_AMP_IP_FAC_AMP_CP	6	0H...FFFFH	0...5434	0.083	hPa
Correction factor for ambient pressure					
IP_FAC_TIA_CP__TIA	6	0H...FFH	0...1.992	0.0078	-
LDP_TIA_IP_FAC_TIA_CP	6	0H...FEH	-48...142.5	0.75	°C
Correction factor for intake air temperature					
IP_T_DLY_CP__MAF_CYL	1*6	0H...FFFFH	0...6.4	0.0001	sec
LDPM_MAF_CYL_IP_REL_FLOW_INC_CP	6	0H...FFFFH	0...2047.96875	0.03125	kg/h
Time delay for MAF_CYL (delay time in seconds)					
IP_T_DLY_CPPWM__N_32	1*6	0H...FFFFH	0...6.4	0.0001	sec
LDPM_N_32_IP_T_DLY_CPPWM	6	0H...FFH	0...8160	32	rpm
Time delay for REL_FLOW_CPS delay CPS - Injection valce					
IP_REL_FLOW_CPS_DLY_CRLC__N_32	1*6	0H...FFFFH	0...1	0.000015	-
LDPM_N_32_IP_T_DLY_CPPWM	6	0H...FFH	0...8160	32	rpm
Filter constant for REL_FLOW_CPS _DLY_MMV calculation					
IP_MAF_CPS__PRS_CPS	1*16	0H...FFFFH	0...8	0.000122	kg/h
LDPM_PRS_CPS_IP_CPPWM	16	0H...FFFFH	-2717..2717	0.083	hPa
MAF_CPS for fully opened CPS (CPPWM=100%)					

9.21 Canister Purge Solenoid Adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CPPWM_ADD_AD	V/O	0H...7FFFH	0...99.9	0.0031	%
Additive adaptive CPPWM correction					
CPPWM_AD_MEM	V/S	0H...7FFFH	0...99.9	0.0031	%
Memory value of the additive adaptive CPPWM correction					
FLOW_COR_CPS_OLD	V/S	0H...FFFFH	0...8	0.000122	kg/h
Memory value of the corrected flow setpoint through the CPS during adaptation					
CPS_AD_STEP_CTR	V	0H...FFH	0...255	1	-
Adaptation step counter					
T_CPS_AD	V	0H...FFFFH	0...6553.5	0.1	s
Maximum time for validation of CPPWM_AD_MEM value					

Input data:

LV_IS	FLOW_COR_CPS	PRS_CPS	CL_MMV
CP_STATE			

FUNCTION DESCRIPTION:

General information:

tbd

Signal flow diagram:

tbd

Description:

tbd

Application conditions:

Activation:

LC_CPS_AD_ENA=1

Deactivation:

otherwise

Update Rate:

100 ms

Formula section:

The following adaptation conditions are checked the whole time during CPS adaptation.

If LV_IS = 1 and
PRS_CPS > C_PRS_AD_MAX_CPS and
FLOW_COR_CPS < C_FLOW_AD_MAX_CPS and
CP_STATE = 4 (MAX_PURGE active)

Then all adaptation conditions are fulfilled (2)
Else adaptation conditions are not fulfilled (1)

1) Adaptation conditions are not fulfilled

CPS_AD_STEP_CTR = T_CPS_AD = 0

2) All adaptation conditions are fulfilled

Step 0: Start of Adaptation

```
If      CL_MMV >= C_CL_AD_MAX_CPS
Then    FLOW_COR_CPS_OLD = FLOW_COR_CPS
        CPS_AD_STEP_CTR = 1
Else    CPS_AD_STEP_CTR = 0
Endif
```

Step 1: Calculation of the additive adaptive CPPWM correction

```
If      CL_MMV >= C_CL_AD_MAX_HYS_CPS
Then    If      CPPWM_ADD_AD(n) < C_CPPWM_AD_MAX
        Then    CPPWM_ADD_AD(n) = CPPWM_ADD_AD(n-1) + C_CPPWM_AD
        Else    CPPWM_ADD_AD(n) = C_CPPWM_AD_MAX
Else    CPS_AD_STEP_CTR = 2
        CPPWM_ADD_AD(n) = CPPWM_ADD_AD(n-1)
```

Step 2: Calculation of memory value of the additive adaptive CPPWM_AD_MEM

```
If      T_CPS_AD(n) < C_T_AD_MAX_CPS          and
        CL_MMV >= C_CL_AD_MIN_CPS
Then    T_CPS_AD(n) = T_CPS_AD(n-1) + 1
        CPPWM_ADD_AD(n) = CPPWM_ADD_AD(n-1)
Else    T_CPS_AD = 0
        CPS_AD_STEP_CTR = 0
        If      CL_MMV < C_CL_AD_MIN_CPS
        Then    CPPWM_AD_MEM(n) = CPPWM_AD_MEM(n-1)
        Else    CPPWM_AD_MEM(n) = CPPWM_ADD_AD(n)
        Endif
Endif
```

If the following conditions are fulfilled, the calculation of CPPWM_ADD_AD takes place at the whole time during CPS adaptation.

Calculation of CPPWM_ADD_AD

```
If      CPPWM_AD_MEM ≠ 0                and
        CPS_AD_STEP_CTR = 0
Then    If      FLOW_COR_CPS <= FLOW_COR_CPS_OLD
        Then    CPPWM_ADD_AD = CPPWM_AD_MEM
        Else    If      FLOW_COR_CPS < C_FLOW_AD_MAX_CPS
                Then    CPPWM_ADD_AD =
                        (C_FLOW_AD_MAX_CPS - FLOW_COR_CPS) /
                        (C_FLOW_AD_MAX_CPS - FLOW_COR_CPS_OLD)
                        * CPPWM_AD_MEM
                Else    CPPWM_ADD_AD = 0
                Endif
        Endif
Else    CPPWM_ADD_AD = CPPWM_ADD_AD
Endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PRS_AD_MAX_CPS	1	8000H...7FFFH	-2717...2717	0.083	hPa
Threshold of pressure difference AMP-MAP for CPS adaptation					
C_FLOW_AD_MAX_CPS	1	0H...FFFFH	0...8	0.000122	kg/h
Threshold for CPS-flow for CPS adaptation					
C_CL_AD_MAX_CPS	1	0H...FFFFH	0...8	0.000122	-
Maximum threshold value for CPS adaptation					
C_CL_AD_MIN_CPS	1	0H...FFFFH	0...8	0.000122	-
MInimum threshold value for CPS adaptation					
C_CL_AD_MAX_HYS_CPS	1	0H...FFFFH	0...8	0.000122	-
Hysteresis for CPS adaptation					
C_CPPWM_AD	1	0H...7FFFH	0...99.99	0.0031	%
Additive value for CPPWM increase during CPS adaptation					
C_CPPWM_AD_MAX	1	0H...7FFFH	0...99.99	0.0031	%
Maximum adaptation value for CPPWM_AD_MEM during CPS adaptation					
C_T_AD_MAX_CPS	1	0H...FFFFH	0...6535.5	0.1	s
Maximum time for validation of CPPWM_AD_MEM value					
LC_CPS_AD_ENA	1	0H...1H	0...1	1	-
Activation constant for CPS adaptation					

9.22 Cooling and condenser fan control

FUNCTION DESCRIPTION:

General information:

The configuration constant NC_ECF_CONF allows to determine the wanted fan control strategy. In principal it is possible to choose between a RLY- or a PWM-control architecture. A RLY/PWM fan control strategy in parallel is also existing. In this case the aggregate architecture allows to switch either the control of RLY-fan(s) or the control of PWM–fan(s) during ECU runtime depending on the setting of the configuration bit CONF_CFA. The control of both variants (RLY- and PWM-fan(s)) in parallel at the same time is not allowed.

The following fan control architecture is adjusted:

NC_ECF_CONF = 2 ➔ control of cooling fan(s) by Relay(s) **or** PWM(s)
(CONF_CFA = 0 for RLY- control)

(CONF_CFA = 1 for PWM- control)

The configuration bit CONF_CFA is representing the activated control algorithm. Either the functions for RLY-fan(s) or the functions for PWM-fan(s) are activated. Both control functions are never activated in at the same time.

For each RLY- and PWM-controlled fan, control signals have to be generated by the “Cooling and condenser fan control” module to provide valid interfaces to other functions located inside or outside the aggregate ENTE.

As soon as the RLY-fan control function is activated, the PWM-fan control function is disabled and vice versa. It is not possible to vary the number of available cooling fans [NC_ECF_NR] at the vehicle (hardware components) for the RLY- and PWM-control strategy. The setting of the total number of cooling fans at the vehicle [NC_ECF_NR] is equal for both control strategies. For the configuration of NC_ECF_NR the higher amount of the available RLY-fan(s) or PWM-fan(s) should be taken into consideration.

NC_ECF_NR = ***[1]***

Because the number of RLY output stages [NC_ECF_RLY_NR] per cooling fan at the vehicle [NC_ECF_NR] is regardless for the PWM-fan control function, the value have to be adjusted dependent on the hardware configuration of the RLY-fan(s).

NC_ECF_RLY_NR = ***[2]***

9.22.1 Condition for high Coolant Temperature control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_CFA_TCO_H	V/O	0...1H	0...1	1	-
Condition for high TCO cooling fan control					

Input data:

TAM	TIA	VS	N_32
MAF	TPS	AMP_AD	LV_ERR_TPS
LV_ERR_TIA	LV_ERR_ACP		

FUNCTION DESCRIPTION:

General information:

In order to allow a higher TCO value under certain conditions by reduced cooling fan activity (fuel economy improvement), the logical variable LV_CDN_CFA_TCO_H is implemented.

LV_CDN_CFA_TCO_H is calculated independant of CONF_CFA as valid for both cooling fan control methods (Relay or PWM).Application conditions:

Initialisation at RESET: LV_CDN_CFA_TCO_H = 0

Recurrence: **40 msec**

Activation: at every engine operating state

Deactivation: -

Description:

If TAM <= C_TAM_MAX_CFA_TCO_H
and TIA <= C_TIA_MAX_CFA_TCO_H
and VS >= C_VS_MIN_CFA_TCO_H
and N_32 <= C_N_MAX_CFA_TCO_H
and MAF <= C_MAF_MAX_CFA_TCO_H
and TPS <= C_TPS_MAX_CFA_TCO_H
and AMP_AD >= _C_AMP_AD_MIN_CFA_TCO_H
and LV_ERR_TPS = 0
and LV_ERR_TCO = 0
and LV_ERR_TIA = 0
and LV_ERR_ACP = 0
then LV_CDN_CFA_TCO_H =1 *(High TCO cooling fan control can be applied)*

If LV_CDN_CFA_TCO_H =1 *(High TCO cooling fan control active)*

and **(** TAM > C_TAM_MAX_CFA_TCO_H

or TIA > C_TIA_MAX_CFA_TCO_H

or VS < C_VS_MIN_CFA_TCO_H

or N_32 > C_N_MAX_CFA_TCO_H

or MAF > C_MAF_MAX_CFA_TCO_H + C_MAF_HYS_CFA_TCO_H

or TPS > C_TPS_MAX_CFA_TCO_H + C_TPS_HYS_CFA_TCO_H

or AMP_AD < _C_AMP_AD_MIN_CFA_TCO_H

or LV_ERR_TPS = 1

or LV_ERR_TCO = 1

or LV_ERR_TIA = 1

or LV_ERR_ACP = 1 **)**

then LV_CDN_CFA_TCO_H = 0 *(High TCO cooling fan control can be not applied)*

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TAM_MAX_CFA_TCO_H	1	0...FEH	-48...142.5	0.75	°C
maximum TAM to allow high TCO cooling fan control					
C_TIA_MAX_CFA_TCO_H	1	0...FEH	-48...142.5	0.75	°C
maximum TIA to allow high TCO cooling fan control					
C_VS_MIN_CFA_TCO_H	1	0...FFH	0...255	1	km/h
minimum VS to allow high TCO cooling fan control					
C_N_MAX_CFA_TCO_H	1	0...FFH	0...8160	32	rpm
maximum N to allow high TCO cooling fan control					
C_MAF_MAX_CFA_TCO_H	1	0... FFFFH	0... 1389	0.0211948	mg/stk
maximum MAF to allow high TCO cooling fan control					
C_TPS_MAX_CFA_TCO_H	1	0...FFH	0...119.5	0.47	°TPS
maximum TPS to allow high TCO cooling fan control					
C_AMP_AD_MIN_CFA_TCO_H	1	0...FFFFH	0...5434	0.0829175	hPa
minimum AMP_AD to allow high TCO cooling fan control					
C_MAF_HYS_CFA_TCO_H	1	0... FFFFH	0... 1389	0.0211948	mg/stk
MAF hysteresis to reset high TCO cooling fan control					
C_TPS_HYS_CFA_TCO_H	1	0...FFH	0...119.5	0.47	°TPS
TPS hysteresis to reset high TCO cooling fan control					

9.22.2 Cooling fan control (RLY version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR]	V/O	0...01H	0...1	1	-
Boolean for activation of the cooling fan stages [NC_ECF_RLY_NR] for each cooling fan [NC_ECF_NR]					
POW_OUT_ECF_RLY	V/O	0…FFH	0…99.61	0.39	%
Cooling fan power output of all available RLY fan(s) at the vehicle					
STATE_CFA	V/O	0...03H	0...3	1	-
Cooling and condenser fans state (0,1,3).					
STATE_CFA_IS	V/O	0...03H	0...3	1	-
Cooling and condenser fans state for idle - charge actuator opening corrections (0,1,3).					
STATE_CFA_1	V	0...03H	0...3	1	-
Cooling and condenser fans state calculated from TCO, LV_ACC_CFA, VS_STATE (0,1,3)					
STATE_CFA_NEW	V	0...03H	0...3	1	-
Cooling and condenser fans state calculated from TCO, LV_ACC_CFA, VS_STATE without TCO hysteresis (0,1,3)					
TCO_HYS_CFA	V	00...FFH	-48...142.5	0.75	°C
Intermediate variable to take in account TCO hysteresis					
LV_RLY_FAN_L	V	0...01H	0...1	1	-
Logic output from ECU for cooling and condenser fans control (LOW) - (OFF / ON).					
LV_RLY_FAN_H	V	0...01H	0...1	1	-
Logic output from ECU for cooling and condenser fans control (HIGH) - (OFF / ON).					

Input data:

VS_STATE_CFA	TCO	LV_PRS_ACC	LV_RLY_ACCOUT
LV_ES	TIA_THR	LV_ACIN	T_AST
LV_ERR_TCO	LV_ERR_TCO_STUCK	LV_ERR_TCO_GRD	LV_ERR_TCO_PLAUS
CONF_CFA	STATE_ACP	CONF_ACP	LV_ERR_ACP
LV_CFA_EXT_ADJ	STATE_CFA_EXT_ADJ	NC_ECF_RLY_NR	NC_ECF_NR
LV_ACC_CFA	LV_CDN_CFA_TCO_H		

FUNCTION DESCRIPTION:

General information:

This function is only called if CONF_CFA is set to 0 (RLY type cooling fan).

The cooling and condenser fan relays are controlled by the EMS via the logic switches LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_ NR].

Because two relays per cooling fan are available (LOW / HIGH) at the vehicle, the number of RLY output stages per cooling fan is set to NC_ECF_RLY_NR = [2].

The logical variables for the control of the fan stages LV_RLY_FAN_L and LV_RLY_FAN_H are combined within the logical output LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR] as following:

- LV_ACT_ECF[1][1] = LV_RLY_FAN_L (LOW)
- LV_ACT_ECF[1][2] = LV_RLY_FAN_H (HIGH)

LV_RLY_FAN_L and LV_RLY_FAN_H are set **ON** or **OFF** depending on:

- coolant temperature TCO
- air condition status: A/C request LV_ACIN or A/C Compr. Relay LV_RLY_ACCOUT (depending on selection by LC_ACC_CFA_SEL)

- vehicle speed state VS_STATE_CFA
- Pressure in the A/C circuit LV_PRS_ACC
- External adjustment by service tester LV_CFA_EXT_ADJ
- high TCO cooling fan control function LV_CDN_CFA_TCO_H

The computation of the variable STATE_CFA determines the level of the two logic outputs LV_RLY_FAN_L and LV_RLY_FAN_H.

If the pressure in the A/C circuit is high (LV_PRS_ACC = 1) and A/C on (LV_ACC_CFA = 1) then cooling and condenser fans speed is set to high in order to fastly decrease the pressure.

In order to allow a higher TCO value under certain conditions by reduced cooling fan activity (fuel economy improvement), the logical variable LV_CDN_CFA_TCO_H is implemented.

Application conditions:

Initialisation at RESET:	POW_OUT_ECF_RLY = 0% LV_ACT_ECF[NC_ECF_NR][NC_ECF_RLY_NR] = 0
Recurrence:	**40 msec**
Activation:	CONF_CFA = 0 *(RLY type cooling fan)* at every engine operating state
Deactivation:	-

(see: “ENTE scheduler”)

Description:

LV_RLY_FAN_L		LV_RLY_FAN_H = 0: Cooling Fan	LV_RLY_FAN_H = 0: STATE_CFA	LV_RLY_FAN_H = 0: Condenser Fan	LV_RLY_FAN_H = 1: Cooling Fan	LV_RLY_FAN_H = 1: STATE_CFA	LV_RLY_FAN_H = 1: Condenser Fan	LV_ACC_CFA
0		OFF		OFF				0
0			**0**			**2** 1)		
0		OFF		OFF				1
1		LOW		LOW	HIGH		HIGH	0
1			**1**			**3**		
1		LOW		LOW	HIGH		HIGH	1

1) only for actuator test

The states of LV_RLY_FAN_L, LV_RLY_FAN_H, LV_ACC_CFA correspond to:

Logical state **0** ⇔ **OFF:** **+12 Volt** present on the corresponding pin - out of the ECU

Logical state **1** ⇔ **ON:** **Ground** present on the corresponding pin - out of the ECU

Application conditions:

Following intermediate variables are used to manage cooling fan relays and ISA corrections :

STATE_CFA : Final value to command ECU outputs for cooling fan relays

STATE_CFA_IS : used to activate idle speed valve correction. It is set to new STATE_CFA value a little in advance

STATE_CFA_1 : Cooling fan state only determined from TCO, VS_STATE_CFA, LV_ACC_CFA. It does not take in account special cases like Start, High pressure in A/C circuit, cooling fans activation during power latch phase, TCO sensor error

STATE_CFA_NEW : Cooling fan state only determined from indexed table without taking in account TCO hysteresis

STATE_CFA_IS calculation:

STATE_CFA_IS is used to activate the idle speed valve corrections (ISAPWM_CFA).

If Engine operating state LV_ES **= 1**
and TCO **>=** C_TCO_ON_CFA_ES
then A timer is started and STATE_CFA_IS = STATE_CFA_1
when timer has reached ID_ON_CFA_ES__TIA_THR__TCO then STATE_CFA_IS = 0
else **If** T_AST **<=** C_T_OFF_CFA_ST
then STATE_CFA_IS **= 0**
else **If** LV_ERR_TCO **= 1**
(error currently present on coolant temperature sensor)
or LV_ERR_TCO_STUCK = 1
or LV_ERR_TCO_GRD = 1
or LV_ERR_TCO_PLAUS = 1
or LV_ERR_ACP = 1
or (LV_PRS_ACC **= 1** **and** LV_ACC_CFA **= 1**)
then STATE_CFA_IS **= 3** *(high pressure in A/C circuit)*
else STATE_CFA_IS **=** STATE_CFA_1 *(low pressure in A/C circuit)*
endif
endif
endif

STATE_CFA calculation:

STATE_CFA is equal to STATE_CFA_IS except if STATE_CFA_IS value is updated. Then a time delay is applied to increase or decrease STATE_CFA.

If LV_CFA_EXT_ADJ = 1
then STATE_CFA = STATE_CFA_EXT_ADJ
else **If** STATE_CFA_IS (n) **=** STATE_CFA_IS (n-1) *(STATE_CFA_IS stays constant)*
then STATE_CFA **=** STATE_CFA_IS
else
If STATE_CFA_IS(n) **>** STATE_CFA_IS (n-1) *(STATE_CFA_IS increases)*
then STATE_CFA **=** STATE_CFA_IS after the time delay C_T_RTD_INC_2_x_CFA

If [(LV_PRS_ACC **=** 1 or STATE_ACP = ACP_H)
and LV_ACC_CFA **= 1**]
then C_T_RTD_INC_x_CFA **=** C_T_RTD_INC_2_1_CFA
else C_T_RTD_INC_x_CFA **=** C_T_RTD_INC_2_0_CFA
endif
else (*STATE_CFA_IS decreases)*
STATE_CFA **=** STATE_CFA_IS after the time delay C_T_RTD_DEC_2_CFA
endif
endif
endif

STATE_CFA_NEW calculation:

The STATE_CFA_NEW is calculated depending on the configuration of LV_ACC_CFA, CONF_ACP and STATE_ACP.

1a) If LV_ACC_CFA = 0 and CONF_ACP = 0 and LV_CDN_CFA_TCO_H = 0

STATE_CFA_NEW = ID_STATE_CFA__TCO__VS_STATE_CFA

1b) If LV_ACC_CFA = 0 and CONF_ACP = 0 and LV_CDN_CFA_TCO_H = 1

STATE_CFA_NEW = ID_STATE_CFA_TCO_H__TCO__VS_STATE_CFA

2a) If LV_ACC_CFA = 0 and CONF_ACP = 1 and LV_CDN_CFA_TCO_H = 0

STATE_CFA_NEW = ID_STATE_CFA__TCO__VS_STATE_CFA

2b) If LV_ACC_CFA = 0 and CONF_ACP = 1 and LV_CDN_CFA_TCO_H = 1

STATE_CFA_NEW = ID_STATE_CFA_TCO_H__TCO__VS_STATE_CFA

3) If LV_ACC_CFA = 1 and CONF_ACP = 0

STATE_CFA_NEW = ID_ACIN_STATE_CFA__TCO__VS_STATE_CFA

4) If LV_ACC_CFA = 1 and CONF_ACP = 1 and STATE_ACP = ACP_H

STATE_CFA_NEW = ID_STATE_CFA_ACP_H__TCO__VS_STATE_CFA

5) If LV_ACC_CFA = 1 and CONF_ACP = 1 and STATE_ACP = ACP_M_2

STATE_CFA_NEW = ID_STATE_CFA_ACP_M_2__TCO__VS_STATE_CFA

6) If LV_ACC_CFA = 1 and CONF_ACP = 1 and STATE_ACP = ACP_M_1

STATE_CFA_NEW = ID_STATE_CFA_ACP_M_1__TCO__VS_STATE_CFA

7) If LV_ACC_CFA = 1 and CONF_ACP = 1 and STATE_ACP = ACP_L

STATE_CFA_NEW = ID_STATE_CFA_ACP_L__TCO__VS_STATE_CFA

STATE_CFA_1 calculation:

If (LV_ACC_CFA **or** VS_STATE_CFA **or** STATE_ACP changes)
or STATE_CFA_NEW **>** STATE_CFA_1
Then STATE_CFA_1 **=** STATE_CFA_NEW
and TCO_HYS_CFA is set to -48°C

If STATE_CFA_NEW **<** STATE_CFA_1
and TCO_HYS_CFA = -48°C
Then TCO_HYS_CFA = TCO - C_TCO_HYS_CFA

If TCO **<** TCO_HYS_CFA
Then STATE_CFA_1 **=** STATE_CFA_NEW
and TCO_HYS_CFA is set to -48°C

Remark : C_TCO_HYS_CFA must be smaller than TCO intervalls in indexed map ID_(ACIN_)STATE_CFA.

POW_OUT_ECF_RLY = ID_POW_OUT_ECF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_STATE_CFA	6 * 3	0-1-3	0-1-3	-	-
LDPM_TCO__ID_STATE_CFA	6	0...FEH	-48...142.5	0.75	°C
LDPM_VS_STATE_CFA__ID_STATE_CFA	3	0...02H	0...2	1	-
Cooling and condenser fans status vs. TCO and VS_STATE_CFA with LV_ACC_CFA =0 / LV_CDN_CFA_TCO_H =0					
ID_STATE_CFA_TCO_H	6 * 3	0-1-3	0-1-3	-	-
LDP_TCO__ID_STATE_CFA_TCO_H	6	0...FEH	-48...142.5	0.75	°C
LDP_VS_STATE_CFA__ID_STATE_CFA_H	3	0...02H	0...2	1	-
Cooling and condenser fans status vs. TCO and VS_STATE_CFA with LV_ACC_CFA =0 / LV_CDN_CFA_TCO_H =1					
ID_ACIN_STATE_CFA	6 * 3	0-1-3	0-1-3	-	-
LDPM_TCO__ID_STATE_CFA	6	0...FEH	-48...142.5	0.75	°C
LDPM_VS_STATE_CFA__ID_STATE_CFA	3	0...2H	0...2	1	-
Cooling and condenser fans status versus TCO and VS_STATE_CFA with LV_ACC_CFA = 1					
C_TCO_HYS_CFA	1	0...FFH	0...191,25	0,75	°C
Coolant temperature hysteresis for STATE_CFA_1 decrease.					
C_T_RTD_INC_2_0_CFA	1	0...FFH	0...10,24	0,040	sec
Time delay before increasing STATE_CFA if LV_PRS_ACC = 0					
C_T_RTD_INC_2_1_CFA	1	0...FFH	0...10,24	0,040	sec
Time delay before increasing STATE_CFA if LV_PRS_ACC = 1					
C_T_RTD_DEC_2_CFA	1	0...FFH	0...10,24	0,040	sec
Time delay before decreasing STATE_CFA					
C_T_OFF_CFA_ST	1	0...FFFFH	0...6553,5	0.1	sec
Time to deactivate fans when engine has been started.					
C_TCO_ON_CFA_ES	1	0...FEH	-48...142,5	0,75	°C
Minimum coolant temperature to start fans under power latch.					
ID_ON_CFA_ES__TIA_THR_TCO	5 x 5	0...FFFFH	0...65535	1	sec
LDP_TCO__IP_ON_CFA_ES	5	0...FFH	-48...142.5	0.75	°C
LDP_TIA_THR_IP_ON_CFA_ES	5	0...FFH	-48...142.5	0.75	°C
Activation duration of the fans under power latch.					
ID_STATE_CFA_ACP_L	6x3	0-1-3	0-1-3	-	-
LDP_TCO__STATE_CFA _ACP_L	6	0...FEH	-48...142.5	0.75	°C
LDP_VS_STATE_CFA__STATE_CFA_ACP_L	3	0...2H	0...2	1	-
Cooling and condenser fans status versus TCO & VS_STATE_CFA in condition of STATE_ACP = ACP_L					
ID_STATE_CFA_ACP_M_1	6x3	0-1-3	0-1-3	-	-
LDP_TCO__ STATE_CFA _ACP_M_1	6	0...FEH	-48...142.5	0.75	°C
LDP_VS_STATE_CFA__STATE_CFA _ACP_M_1	3	0...2H	0...2	1	-
Basic duty cycle depending on TCO & VS_STATE_CFA in condition of STATE_ACP = ACP_M_1					
ID_STATE_CFA_ACP_M_2	6x3	0-1-3	0-1-3	-	-
LDP_TCO__ STATE_CFA _ACP_M_2	6	0...FEH	-48...142.5	0.75	°C
LDP_VS_STATE_CFA__STATE_CFA _ACP_M_2	3	0...2H	0...2	1	-
Basic duty cycle depending on TCO & VS_STATE_CFA in condition of STATE_ACP = ACP_M_2					
ID_STATE_CFA_ACP_H	6x3	0-1-3	0-1-3	-	-
LDP_TCO__ STATE_CFA _ACP_H	6	0...FEH	-48...142.5	0.75	°C
LDP_VS_STATE_CFA__ STATE_CFA _ACP_H	3	0...2H	0...2	1	-
Basic duty cycle depending on TCO and VS_STATE_CFA in condition of STATE_ACP = ACP_H					
ID_POW_OUT_ECF	3	0...FFH	0...99.61	0.39	%
LDP_STATE_CFA _ID_POW_OUT_ECF	3	0...3H	0...3	1	-
Table for overall cooling fan power output in case of RLY fan control					

9.22.3 Cooling fan control (PWM version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ECFPWM[NC_ECF_NR]	O/V	0...FFH	0...99.61	0.39	%
Pulse width modulated signal for the electronic controlled cooling fan(s)					
POW_OUT_ECF_PWM	O/V	0…FFH	0…99.61	0.39	%
Cooling fan power output of all available RLY fan(s) at the vehicle					
CFAPWM_CFA	V/O	0...FFH	0...99.61	0.39	%
Electric fan control pulse width modulation					
CFAPWM_BAS	V	0...FFH	0...99.61	0.39	%
Basic electric fan control pulse width modulation					
CFAPWM	V	0...FFH	0...99.61	0,39	%
Electric fan control pulse width modulation with no consideration of VB variation					
CFAPWM_MIN_HYS	V	0...FFH	0...99.61	0.39	%
Hysteresis of set point Electric fan speed stage					
CFAPWM_COR_FIL	V	0...FFH	0...99.61	0.39	%
Filtered value of basic electric fan control pulse width modulation					
CFAPWM_LGRD	V	0...FFH	0...99.61	0.39	%
Electric fan control pulse width modulation which filtered by change limitation					
TCO_CFA	V	0...FEH	-48...142.5	0.75	°C
Coolant temperature which considered hysteresis for CFAPWM calculation					

Input data:

TCO	LV_PRS_ACC	LV_IGK	VS_STATE_CFA
VB_MMV	NC_ECF_NR	CONF_ACP	STATE_ACP
LV_CDN_CFA_TCO_H	CONF_CFA	LV_ERR_TCO	LV_ERR_TCO_GRD
LV_ERR_TCO_PLAUS	LV_ERR_TCO_STUCK	T_AST	C_TCO_ON_CFA_ES
ID_ON_CFA_ES	LV_CFA_EXT_ADJ	CFAPWM_EXT_ADJ	VS
LV_ACC_CFA	LV_ES		

FUNCTION DESCRIPTION:

General information:

This function is only called if CONF_CFA is set to 1.

The activation of the PWM cooling fan is controlled by the ECU via ECFPWM[NC_ECF_NR].

Because one RLY or PWM cooling fan is available (CFAPWM_CFA) at the vehicle, the number of fans is set to NC_ECF_NR = 1.

The signal for the control of the pulse width modulated fan CFAPWM_CFA is located within the output variable ECFPWM[NC_ECF_NR] as following:
- ECFPWM[1] = CFAPWM_CFA

The electric fan is controlled with a 300 Hz PWM signal to regulate its rotational speed. The calculated pulse width modulation depends on:

- Coolant temperature	TCO_CFA
- The vehicle speed state for cooling fan	VS_STATE_CFA
- The moving average value of battery voltage	VB_MMV
- Air condition selected / compressor actvie	LV_ACC_CFA
- Status of pressure switch	LV_PRS_ACC
- Air conditioning pressure state	STATE_ACP
- Flag to enable high coolant temperature control	LV_CDN_CFA_TCO_H

Application conditions:

Recurrence: *200ms*

Initialisation :

The following initialisation shall be carried out after a RESET and upon leaving the engine state Engine Stop(LV_ES)
CFAPWM_CFA = C_CFAPWM_MIN

CFAPWM_BAS = 0

CFAPWM = 0

CFAPWM_MIN_HYS = 0

CFAPWM_COR_FIL = 0

CFAPWM_LGRD = 0

Initialisation at RESET:

ECFPWM[NC_ECF_NR] = C_CFAPWM_MIN

POW_OUT_ECF_PWM = 0%

Activation: CONF_CFA = 1 *(PWM type cooling fan)*

at every engine operating state

To start the new calculation of pulse width modulation, the time C_T_CFAPWM_AST has to be elapsed after transition from engine Start ST. The time condition is reset each time the ignition key is switched off.

Deactivation: -

(see: "ENTE scheduler")

9.22.4 Calculation of CAFPWM_CFA at engine stop

If LV_CFA_EXT_ADJ = 1

then CFAPWM_CFA = CFAPWM_EXT_ADJ

else **if** Engine operating state LV_ES **= 1**

then **if** TCO **>=** C_TCO_ON_CFA_ES

then A timer is started and CFAPWM_CFA is calculated.

when the timer has reached ID_ON_CFA_ES__TIA_THR__TCO,

CFAPWM_CFA is set to C_CFAPWM_MIN.

(Due to this function, power latch time could be extended by ID_ON_CFA_ES)

else CFAPWM_CFA = C_CFAPWM_MIN

endif

else **if** T_AST **<=** C_T_CFAPWM_AST

then CFAPWM_CFA = C_CFAPWM_MIN

endif

endif

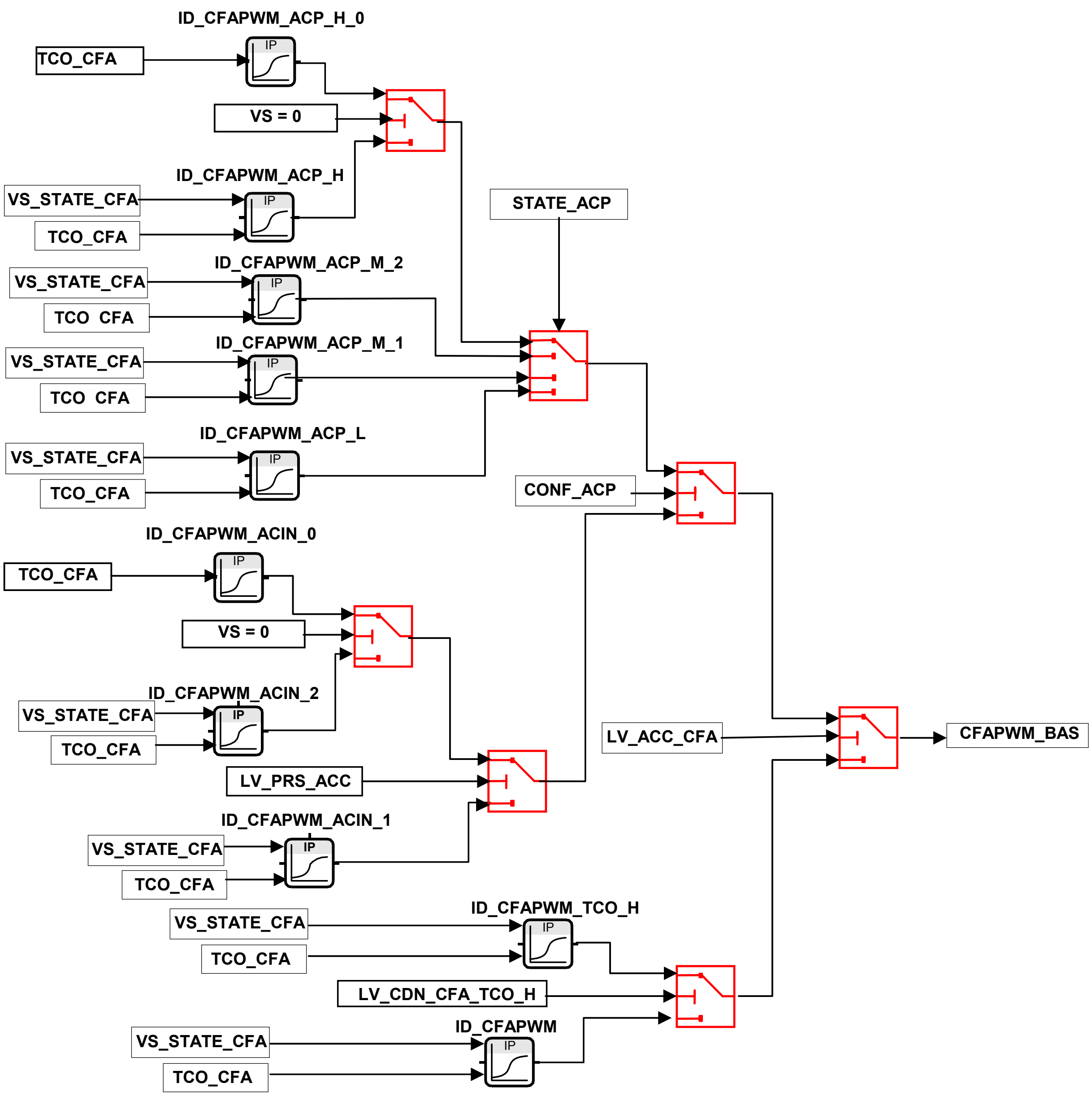
endif

9.22.5 Calculation of CFAPWM_BAS:

Description:

The basic fan control PWM table is selected by condition of LV_ACC_CFA, CONF_ACP and STATE_ACP.

Signal flow diagram:



Formula section:

1) Basic CFAPWM with LV_ACC_CFA = 0

```
if      LV_ACC_CFA = 0
        if      LV_CDN_CFA_TCO_H = 0
        then    CFAPWM_BAS = ID_CFAPWM__VS_STATE_CFA__TCO_CFA
        else    CFAPWM_BAS = ID_CFAPWM_TCO_H__VS_STATE_CFA__TCO_CFA
        endif
endif
```

2) Basic CFAPWM with LV_ACC_CFA = 1

In case of CONF_ACP = 0

```
if      LV_PRS_ACC = 0
then    CFAPWM_BAS = ID_CFAPWM_ACIN_1__VS_STATE_CFA__TCO_CFA
else    if      LV_PRS_ACC = 1
        then    if      VS = 0
                then    CFAPWM_BAS = ID_CFAPWM_ACIN_0__TCO_CFA
                else    CFAPWM_BAS = ID_CFAPWM_ACIN_2__VS_STATE_CFA__TCO_CFA
                endif
        endif
endif
```

In case of CONF_ACP = 1

```
if      STATE_ACP = ACP_L
then    CFAPWM_BAS = ID_CFAPWM_ACP_L__VS_STATE_CFA__TCO_CFA
else    if      STATE_ACP = ACP_M_1
        then    CFAPWM_BAS = ID_CFAPWM_ACP_M_1__VS_STATE_CFA__TCO_CFA
        else    if      STATE_ACP = ACP_M_2
                thenCFAPWM_BAS=ID_CFAPWM_ACP_M_2__VS_STATE_CFA__TCO_CFA
                else    if      STATE_ACP = ACP_H
                        then    if      VS = 0
                                then    CFAPWM_BAS =
                                          ID_CFAPWM_ACP_H_0__VS_STATE_CFA__TCO_CFA
                                else    CFAPWM_BAS =
                                          ID_CFAPWM_ACP_H__VS_STATE_CFA__TCO_CFA
```

endif

endif

endif

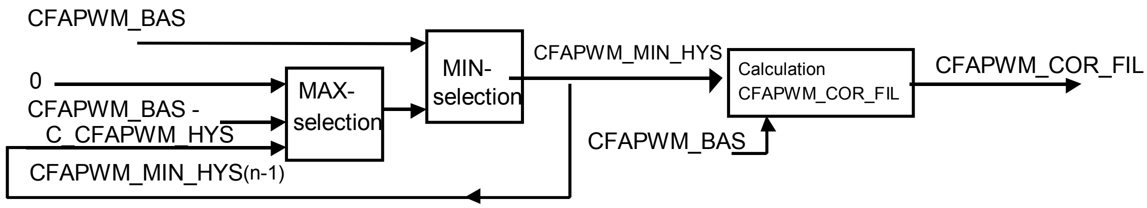
endif

endif

Filtering of CFAPWM_BAS

To avoid frequently changing electric fan speeds (noise problem), CFAPWM_BAS is filtered as follows.

Signal flow diagram:

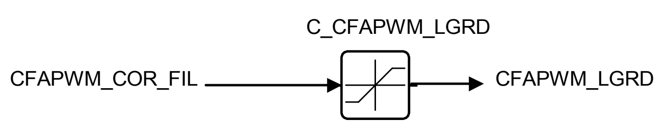


Formula section:
Calculation of CFAPWM_COR_FIL:

if CFAPWM_BAS = 0
then CFAPWM_COR_FIL = 0
else
  If CFAPWM_BAS > 0 and CFAPWM_MIN_HYS = 0
  then CFAPWM_COR_FIL = CFAPWM_BAS
  else CFAPWM_COR_FIL = CFAPWM_MIN_HYS + C_CFAPWM_HYS
  endif
endif

The value C_CFAPWM_HYS determines the filter effect.

Abrupt changes of this speed are slowed down by the gradient limitation:



To have an option for a manual application correction, the application constant is add.

CFAPWM = CFAPWM_LGRD + C_CFAPWM_AS

To take into account the battery voltage variation, the CFAPWM is corrected as following.

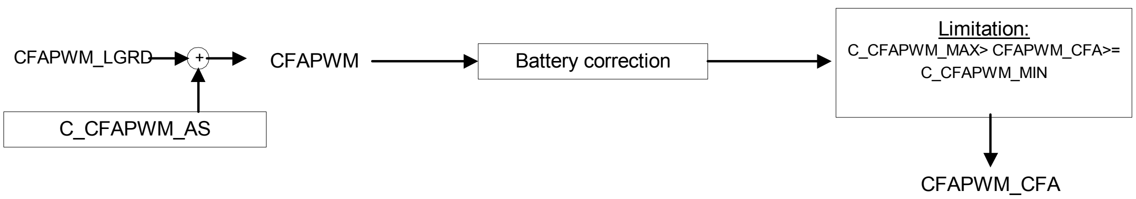
CFAPWM_CFA = IP_CFAPWM_CFA__CFAPWM__VB_MMV

Limitation of CFAPWM_CFA to values between C_CFAPWM_MIN and C_CFAPWM_MAX occurs as the last action before the output.

Limitation:

CFAPWM_CFA >= C_CFAPWM_MIN **and**
CFAPWM_CFA < C_CFAPWM_MAX

Signal flow diagram:



Definition of TCO_CFA

Initialsation : TCO_CFA = TCO at reset or tansition LV_IGK = 0 → 1

If TCO < TCO_CFA - C_TCO_CFA_HYS

Then TCO_CFA = TCO

Else **if** TCO > TCO_CFA

then TCO_CFA = TCO

if TCO < TCO_CFA and the difference between TCO & TCO_CFA is smaller than C_TCO_CFA_HYS, TCO_CFA will not be changed.

Emergency operation by TCO error detection

In case of TCO error detection, CFAPWM_CFA is set to C_CFAPWM_MAX immediately.

If LV_ERR_TCO **= 1**

or LV_ERR_TCO_STUCK = 1

or LV_ERR_TCO_GRD = 1

or LV_ERR_TCO_PLAUS = 1

then CFAPWM_CFA = C_CFAPWM_MAX

POW_OUT_ECF_PWM = ECFPWM[NC_ECF_NR]

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_CFAPWM	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO and VS_STATE in condition of LV_ACC_CFA = 0 and LV_CDN_CFA_TCO_H = 0					
ID_CFAPWM_TCO_H	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_H	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM_H	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO and VS_STATE in condition of LV_ACC_CFA = 0 and LV_CDN_CFA_TCO_H = 1					
ID_CFAPWM_ACP_L	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACP_L	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM_ACP_L	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & STATE_ACP = ACP_L					
ID_CFAPWM_ACP_M_1	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACP_M_1	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM_ACP_M_1	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & STATE_ACP = ACP_M_1					
ID_CFAPWM_ACP_M_2	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACP_M_2	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM_ACP_M_2	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & STATE_ACP = ACP_M_2					
ID_CFAPWM_ACP_H	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACP_H	3	0...2H	0...2	1	-
LDPM_TCO_CFA__CFAPWM_ACP_H	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & STATE_ACP = ACP_H					
ID_CFAPWM_ACP_H_0	3 x 10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACP_H	3	0...2H	0...2	1	-
LDPM_TCO_CFA__CFAPWM_ACP_H	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO in condition of LV_ACC_CFA = 1 & STATE_ACP = ACP_H & VS = 0					
ID_CFAPWM_ACIN_0	10	0...FFH	0...99.61	0.39	%
LDPM_TCO_CFA__CFAPWM_ACIN_1	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO in condition of LV_ACC_CFA = 1 & LV_PRS_ACC = 0 & VS = 0					
ID_CFAPWM_ACIN_1	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACIN_1	3	0...2H	0...2	1	-
LDPM_TCO_CFA__CFAPWM_ACIN_1	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & LV_PRS_ACC = 0					
ID_CFAPWM_ACIN_2	3x10	0...FFH	0...99.61	0.39	%
LDP_VS_STATE_CFA__CFAPWM_ACIN_2	3	0...2H	0...2	1	-
LDP_TCO_CFA__CFAPWM_ACIN_2	10	0...FEH	-48...142.5	0.75	°C
Basic duty cycle depending on TCO & VS_STATE in condition of LV_ACC_CFA = 1 & LV_PRS_ACC = 1					
IP_CFAPWM_CFA	6x6	0...FFH	0...99.61	0.39	%
LDP_CFAPWM__CFAPWM_CFA	6	0...FFH	0...99.61	0.39	%
LDP_VB_MMV__CFAPWM_CFA	6	0...FFH	0...25.8984	26/256	V
Cat substitute temperature initialization map					
C_CFAPWM_HYS	1	0...FFH	0...99.61	0.39	%
Electric fan speed hysteresis					
C_CFAPWM_LGRD	1	0...FFH	0...25	0.098	%/200ms
Limitation gradient of CFAPWM_COR_FIL					
C_CFAPWM_AS	1	0...FFH	-50...49.6094	0.39	%
Application correction					
C_T_CFAPWM_AST	1	0...FFFFH	0...6553,5	0.1	sec
Time condition after start for electric fan active					
C_CFAPWM_MIN	1	0...FFH	0...99.61	0.39	%
Lower CFAPWM_CFA limitation					
C_CFAPWM_MAX	1	0...FFH	0...99.61	0.39	%

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
Upper CFAPWM_CFA limitation					
C_TCO_CFA_HYS	1	0...FFH	0...191.25	0.75	°C
Hysteresis to calculate TCO_CFA					

9.23 ECU power latch phase / ECU power down

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PWL_LOCK_CDN	V/O	0...FFH	0...255	1	-
Power latch extension by: 1: ICU 1 2: ICU 2 4: Engine cooling management 8: Minimum power latch time(T_MIN_PWL) not exceeded 16: IVVT 32: Main relay diagnosis 64 : ISA cleaning function					
LV_PWL	V/O	0...1	0...1	1	-
power latch phase active					
T_MIN_PWL	V	0...FFFFH	0...6553.5	0.1	S
Minimum power latch time					
LV_MU_INH_PWL_TRAN_ES_EL	O	0...1H	0...1	1	-
Flag active in shudown-phase : IGK on is not accepted without reset (Point of no return for ECU)					
CTR_PWL_ACTION	V	0...FFH	0...255	1	-
Counter for proceeding PWL shut down if there is no response from MU					
LV_ECM3_DISABLE_REQ	-	0...1H	0...1	-	-
Flag indicating disable state of MU					
T_PWL	V/O	0...FFFFH	0...6553.5	0.1	[s]
Time elapsed since power-latch phase has started					

Input data:

PWL_LOCK_CDN[bit 0]	LV_IGK		

Import action:

ACTION_ECM3_DisableRequest (OUT<>)
Action to perform a transition to DISABLE

FUNCTION DESCRIPTION:

General information:

The ECU keeps active for a specific time after ignition key off (LV_IGK = 0). This is for example to save adaptation values in EEPROM or for some engine cooling reasons. The minimum ECU power latch time (T_MIN_PWL) can be extended by other functions using the byte PWL_LOCK_CDN.

Application conditions:

Initialisation:

- With every transition to power latch the counter CTR_PWL_ACTION is initialized to NC_CTR_PWL_ACTION and the minimum power latch time T_MIN_PWL is initialized with C_T_MIN_PWL and the related bit of PWL_LOCK_CDN is set
- When leaving the power latch phase the counter CTR_PWL_ACTION is set to NC_CTR:PWL_ACTION and the minimum power latch time is cancelled and the realted bit of PWL_LOCK_CDN is reset
- LV_PWL = 0 with power up and with LV_IGK = 0 --> 1
- T_PWL = 0 at reset and at leaving the power latch phase

- At reset: LV_MU_INH_PWL_TRAN_ES_EL = 0, CTR_PWL_ACTION = 0, LV_ECM3_DISABLE_REQ = 0

Recurrence: 100 ms

Description:

As long as PWL_LOCK_CDN ≠ 0 the power latch time is extended for 100 ms more.

Description of the byte PWL_LOCK_CDN:

PWL_LOCK_CDN = 1	(ICU 1)
PWL_LOCK_CDN = 2	(ICU 2)
PWL_LOCK_CDN = 4	(engine cooling Managment)
PWL_LOCK_CDN = 8	(Minimum power latch time not exceeded)
PWL_LOCK_CDN = 16	(VVTI)
PWL_LOCK_CDN = 32	(Main relay diagnosis)
PWL_LOCK_CDN = 64	(ISA cleaning function)

While ECU power latch time is active (PWL_LOCK_CDN ≠ 0 and T_MIN_PWL ≠ 0):

LV_PWL = 1
T_MIN_PWL = T_MIN_PWL – 100ms
T_PWL = T_PWL +100ms

If (T_MIN_PWL = 0 **and** PWL_LOCK_CDN = 0) **or**
LV_MU_INH_PWL_TRAN_ES_EL = 1 (already in shut down)
Then LV_ECM3_DISABLE_REQ = ACTION_ECM3_DisableRequest()
Remark: for ISA ACTION_ECM3_DisableRequest() is 1 (never changed)
If PWL_LOCK_CDN = 0 **or**
LV_ECM3_DISABLE_REQ =1 (already in shut down)
Then activate shut-down mechanism (Action for shut-down mechanism):
(ECU re-Power-up by ignition switch(IGK) on is forbidden)

LV_MU_INH_PWL_TRAN_ES_EL=0
LV_PWL = 0

- The safety controller (MU) is deactivated, reactivation by IGK is forbidden.
- After deactivation of the MU, the function specific initializations with “end of power latch phase” are done.
- The power-supply-pin of the ECU is switched off.
- ECU is waiting in endless-loop serving the watchdog until power is off or a new ignition key on is detected : LV_KEY_OFF=0
- With new LV_IGK = 1 a complete ECU power up is carried out (ECU reset by SW), not by Watchdog.

Else CTR_PWL_ACTION = CTR_PWL_ACTION -1
LV_MU_INH_PWL_TRAN_ES_EL = 1
Endif
Else CTR_PWL_ACTION = NC CTR_PWL_ACTION
Endif

Remark for ETC version:

The power latch phase prefaces the switch off of the ECU. If the ignition was switched off and as a result the engine stopped, the power latch phase starts. This is shown in ECU_STATE which takes the value ECU_STATE=PWL & LV_PWL=1. The minimum duration of PWL is the calibratable time C_T_MIN_PWL. Till the end of this power latch actions, it is possible to prevent the ECU switch-off by switching ignition on again. If this(switching ignition on again) happens not, next is the switch off of the ECU's monitoring unit (MU) and setting of LV_MU_INH_PWL_TRAN_ES_EL=1.

When this value(LV_MU_INH_PWL_TRAN_ES_EL) is 1, a possible LV_IGK=1 is suppressed in order to allow the ECU-switch-off(the ECU will be switched on immidiately after complete shut down). The switch off of the MU is checked by the main processor. This action extends the power latch phase more. Hereafter the last action before complete shut off takes place : the storage of all nonvolatile data to the memory.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_MIN_PWL	1	0000...FFFFH	0...6553.5	0.1	s
minimum power latch time					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CTR_PWL_ACTION	1	0...FFH	0...255	1	-
Counter for proceeding PWL shut down if there is no response from MU (NC_CTR_PWL_ACTION = 3)					

9.24 Engine Speed Limitation Controller

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_DIF_MAX_LIM	V	E020...1FE0 H	-8160...8160	1	1/min
Engine speed difference as input for the PI-controller					
N_GRD_FIL	V	80...7F H	-4096...4064	32	1/min/s
Engine speed gradient pre-selection for low pass filter					
N_GRD_FIL_FIL	V	80...7F H	-4096...4064	32	1/min/s
Low pass filtered engine speed gradient for predicted engine speed					
N_PRED_MAX_LIM	V	0...1FE0 H	0...8160	1	1/min
Predicted engine speed					
LV_N_MAX	V/O	0...1 H	0...1	1	-
Logical variable engine speed limitation active					
LV_N_MAX_REQ_FCUT	V / O	0...1 H	0...1	1	-
Logical variable fuel cut-off of all cylinder requested for engine speed limitation					
LV_TQ_N_MAX_DIF_I_INI	V/O	0...1 H	0...1	1	-
Logical variable signals initialisation of the integrator					
TQ_N_MAX_DIF	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Torque output of N_MAX-PI-Controller					
TQ_N_MAX_DIF_I	V	80000000...7FFF FFFF H	-1024...1023.97	476.84 E-9	Nm
Torque output of N_MAX-I-Controller					
TQ_N_MAX_DIF_P	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Torque output of N_MAX-P-Controller					
TQ_N_MAX_INP_I	V	8000...7FFF H	-4...3.9997	122.07E-6	Nm
Torque input for N_MAX I-controller					
TQI_N_MAX	V/O	0...7FFF H	0...1023.97	0.03125	Nm
Indicated engine torque setpoint for N_MAX-limitation					
TQI_N_MAX_1	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Intermediate value 1 for indicated engine torque after considering PI-controller output					
TQI_N_MAX_2	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Intermediate value 2 indicated engine torque after considering PI-controller output					
TQI_N_MAX_REF	V	0...7FFF H	0...1023.97	0.03125	Nm
Reference indicated engine torque on which controller output is applied					

Input data:

N_MAX_THD	N	N_GRD	TQI_AV
TQI_REQ_TRA	GEAR	LV_AT	N_32

FUNCTION DESCRIPTION:

General information:

The objective of this function is to protect the engine against overspeed and therefore to control the engine speed. A PI controller is used to hold the engine speed at the engine speed limit setpoint, which derives from the module 'Engine speed limit coordination' from chapter 4. Engine overspeed mainly is controlled by reduction of engine torque. The output value of this module is TQI_N_MAX, which is an input in the torque coordination module. Different tasks are designed in this module:

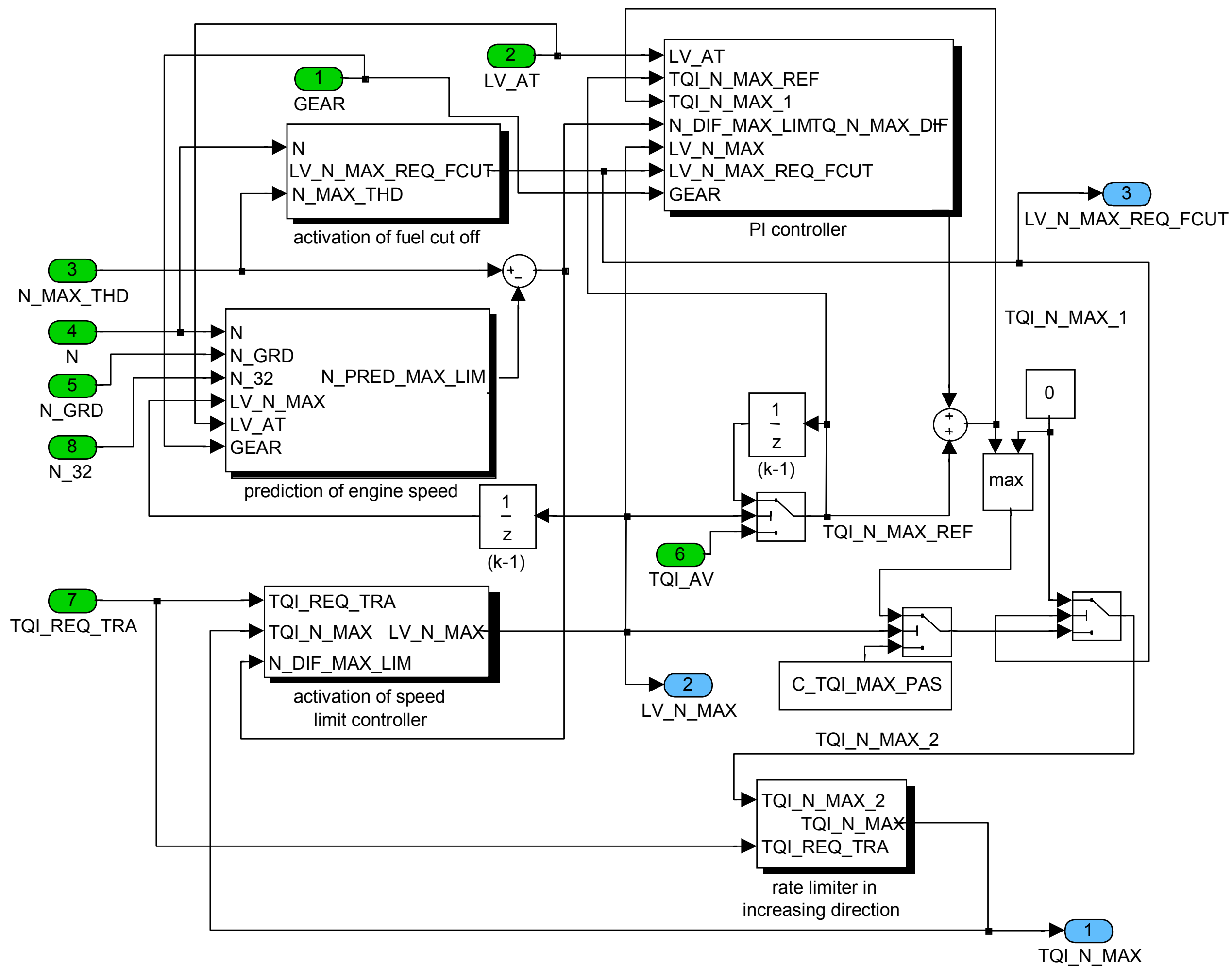
- prediction of engine speed to avoid great engine speed overshoots, especially in the first phase of controlling
- activation of speed limit controller due to speed deviation between predicted engine speed and engine speed limit
- PI controller
- activation of fuel cut off for exceeding the engine speed limit for more than a calibratable difference
- rate limiter for the output in increasing direction to avoid uncomfortable jumps in the torque demand

Application conditions:

Activation:	*at every engine state*		
Deactivation:	-		
Initialization:	TQI_N_MAX	= C_TQI_MAX_PAS	at reset
	N_GRD_FIL	= 0	at reset
	N_GRD_FIL_FIL	= 0	at reset
	LV_N_MAX	= 0	at reset
Update Rate:	10 ms		

9.24.1 Overview

Signal flow diagram:



Formula section:

IF LV_N_MAX == 1

THEN $TQI\_N\_MAX\_REF_k = TQI\_N\_MAX\_REF_{k-1}$

ELSE TQI_N_MAX_REF = TQI_AV

ENDIF

TQI_N_MAX_1 = TQI_N_MAX_REF + TQ_N_MAX_DIF

IF LV_N_MAX_REQ_FCUT == 1

THEN TQI_N_MAX_2 = 0

ELSE **IF** LV_N_MAX == 1

THEN TQI_N_MAX_2 = max(TQI_N_MAX_1, 0 Nm)

ELSE TQI_N_MAX_2 = C_TQI_MAX_PAS

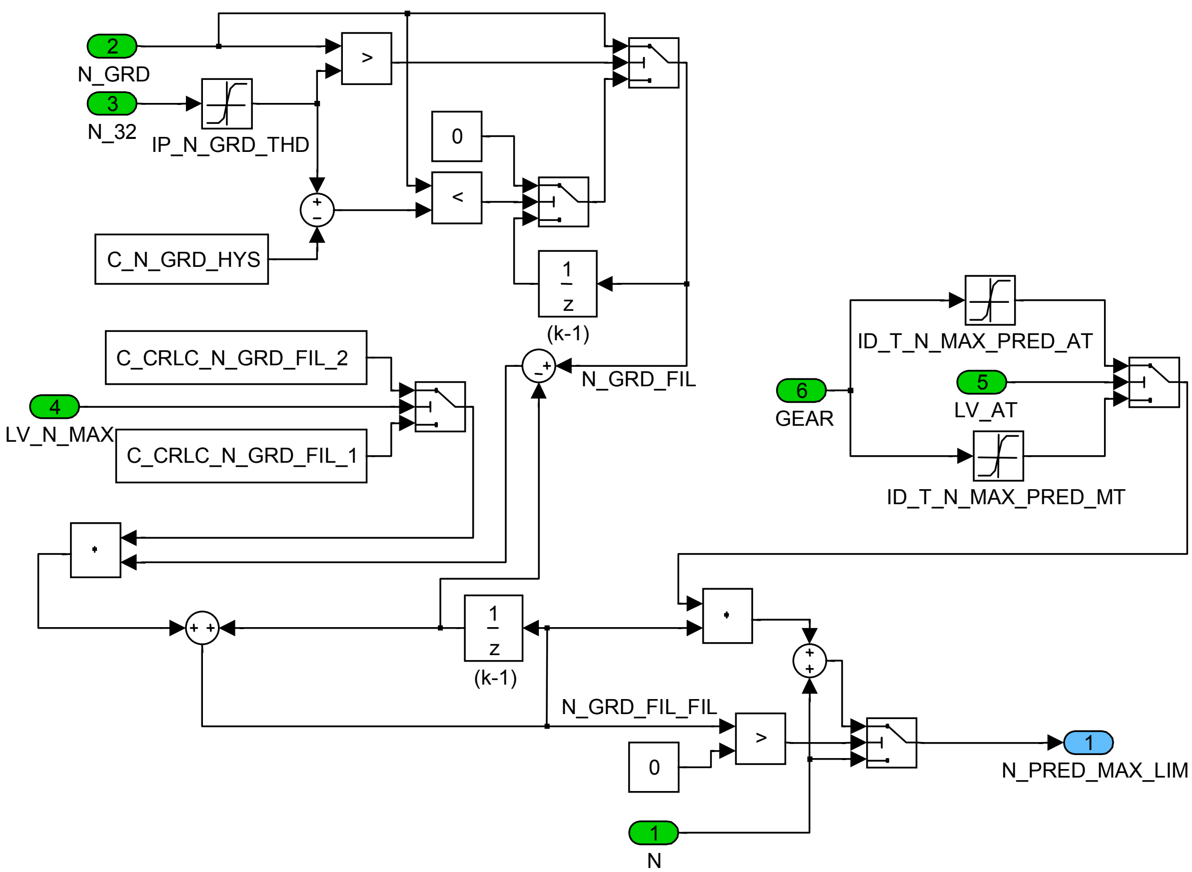
ENDIF

ENDIF

N_DIF_MAX_LIM = N_MAX_THD – N_PRED_MAX_LIM

9.24.2 Prediction of engine speed

Signal flow diagram:



Formula section:

The engine speed prediction shall activate the engine speed limitation controller before the real engine speed reaches the threshold N_MAX_THD. Therefore the complete controller has a PID behaviour at activation, to prevent high engine speed overshoot.

The prediction is deactivated, if the controller has stopped the engine speed increase. During the following period the PI controller can guarantee the engine speed limitation up to N_MAX_THD without predicting the engine speed.

Due to a bad resolution of N_GRD, the prediction is only calculated, if a clear high engine speed gradient is detected. This is possible by comparing the gradient N_GRD with an engine speed depending threshold IP_N_GRD_THD__N_32.

The prediction is stopped latest, if the real engine speed is no longer increasing (N_GRD =< 0).

To prevent sudden changes for the engine speed limit controller input at activation or deactivation, it is necessary to have a PT1 filter for the engine speed controller input. So a negative feed back of the controller output to the real engine speed is prevented.

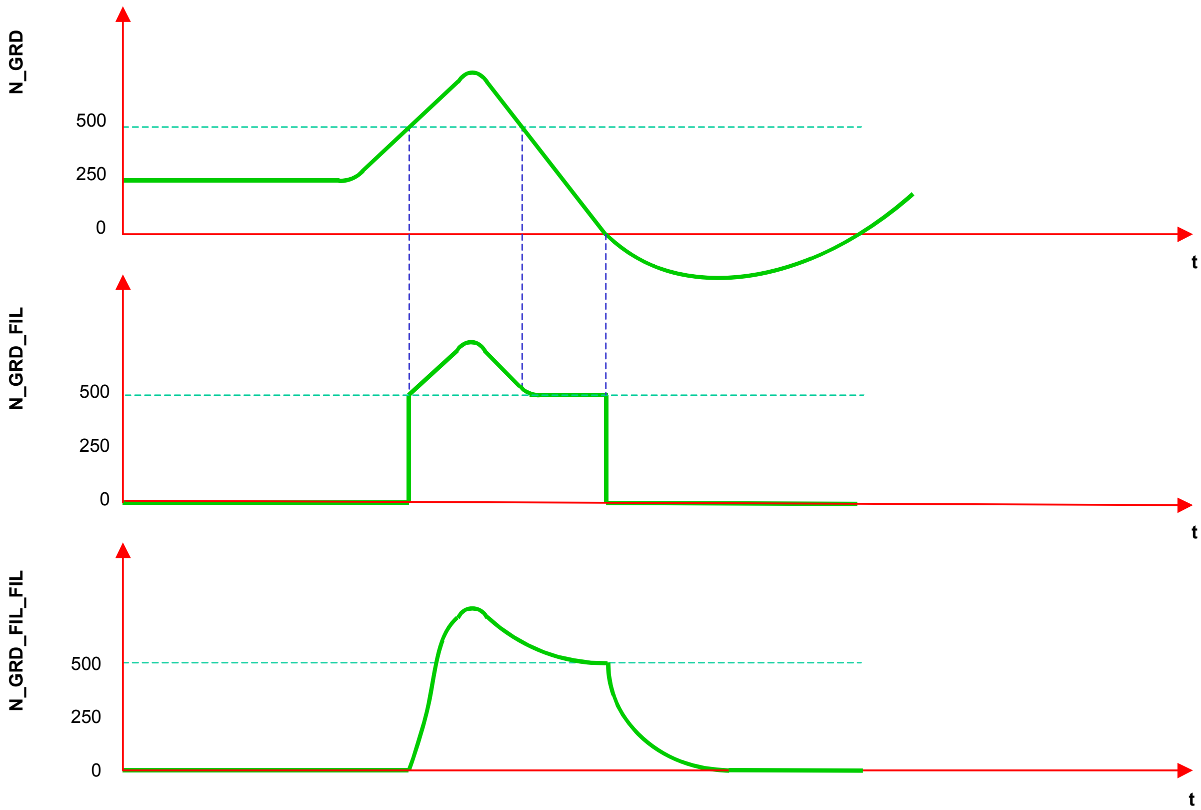


Figure 1: Filtering of engine speed gradient

IF (N_GRD > IP_N_GRD_THD__N_32)

THEN N_GRD_FIL = N_GRD

ELSE **IF** (N_GRD < IP_N_GRD_THD__N_32 - C_N_GRD_HYS)

THEN N_GRD_FIL = 0

ELSE $N\_GRD\_FIL_{k} = N\_GRD\_FIL_{k-1}$

ENDIF

ENDIF

IF LV_N_MAX == 1

THEN N_GRD_FIL_FIL$_k$ = N_GRD_FIL_FIL$_{k-1}$
+ (N_GRD_FIL$_k$ - N_GRD_FIL_FIL$_{k-1}$)
* C_CRLC_N_GRD_FIL_2

ELSE N_GRD_FIL_FIL$_k$ = N_GRD_FIL_FIL$_{k-1}$
+ (N_GRD_FIL$_k$ - N_GRD_FIL_FIL$_{k-1}$)
* C_CRLC_N_GRD_FIL_1

ENDIF

IF (N_GRD_FIL_FIL > 0)

THEN **IF** LV_AT == 1

THEN N_PRED_MAX_LIM = N + N_GRD_FIL_FIL * ID_T_N_MAX_PRED_AT

ELSE N_PRED_MAX_LIM = N + N_GRD_FIL_FIL * ID_T_N_MAX_PRED_MT

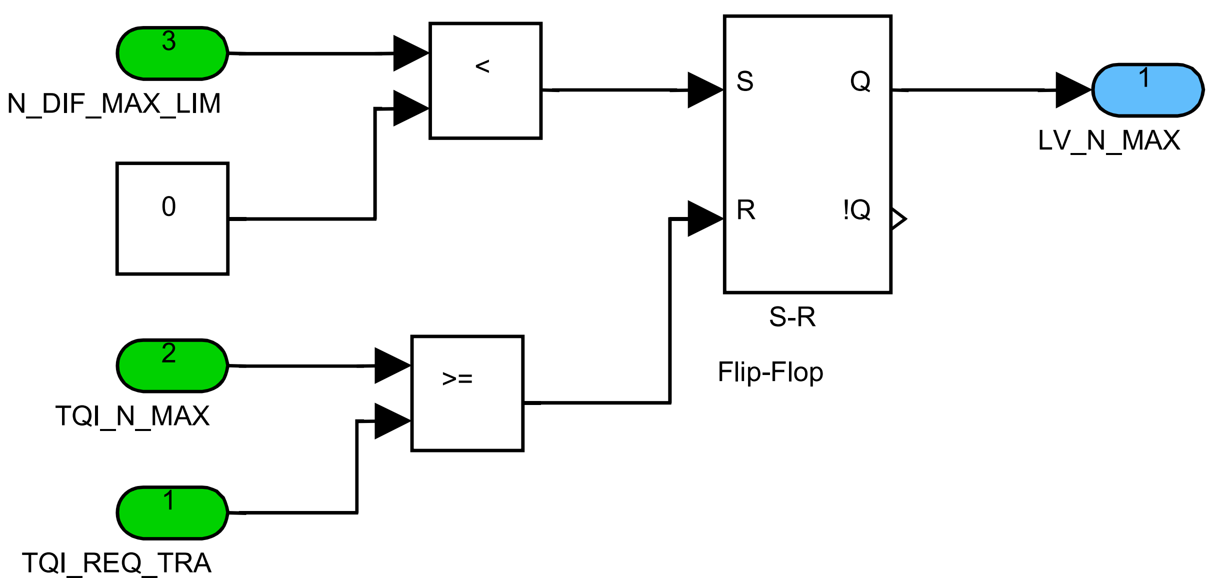
ENDIF

ELSE N_PRED_MAX_LIM = N

ENDIF

9.24.3 Activation of speed limit controller

<u>Signal flow diagram:</u>



<u>Formula section:</u>

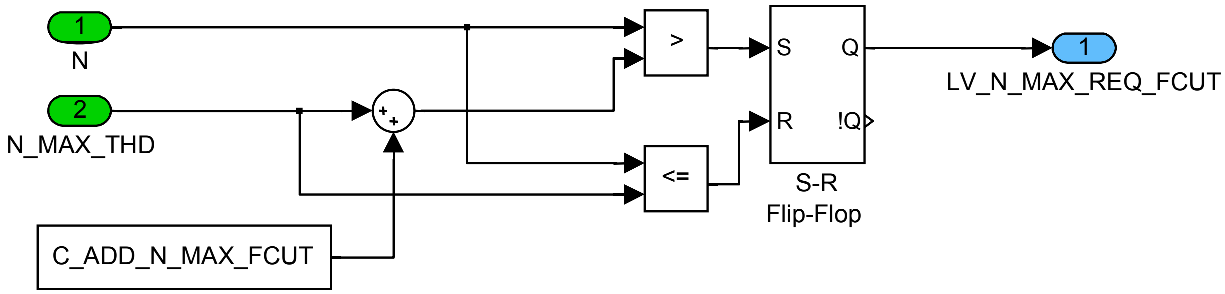
IF (N_DIF_MAX_LIM < 0)

THEN LV_N_MAX = 1

ELSE **IF** (TQI_N_MAX >= TQI_REQ_TRA)
THEN LV_N_MAX = 0
ENDIF
ENDIF

9.24.4 Activation of fuel cut off

Signal flow diagram:

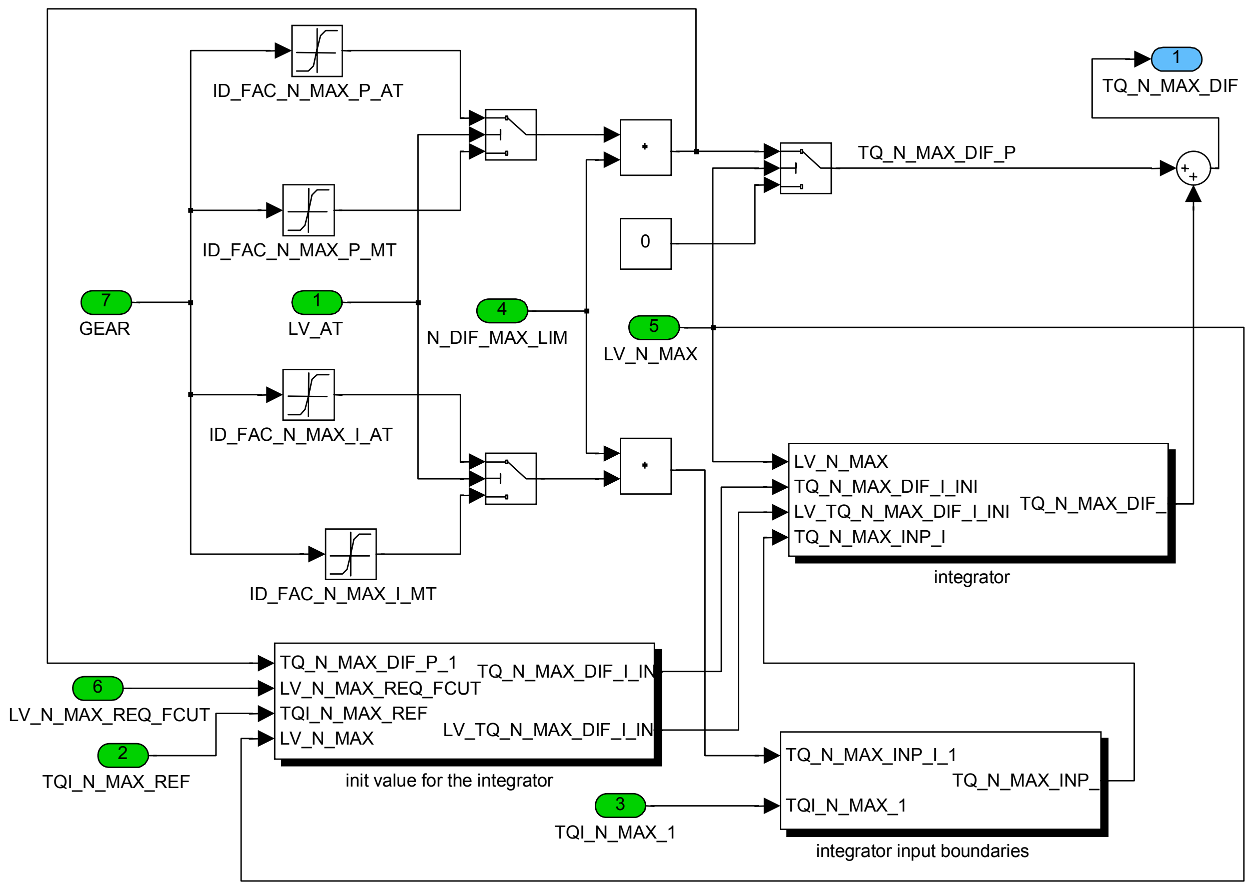


Formula section:

IF (N > (N_MAX_THD + C_ADD_N_MAX_FCUT))
THEN LV_N_MAX_REQ_FCUT = 1
ELSE **IF** N <= N_MAX_THD
THEN LV_N_MAX_REQ_FCUT = 0
ENDIF
ENDIF

9.24.5 PI controller

Signal flow diagram:



Formula section:

TQ_N_MAX_DIF = TQ_N_MAX_DIF_P + TQ_N_MAX_DIF_I

IF LV_AT == 1
THEN TQ_N_MAX_DIF_P_1 = N_DIF_MAX_LIM
* ID_FAC_N_MAX_P_AT__GEAR
ELSE TQ_N_MAX_DIF_P_1 = N_DIF_MAX_LIM
* ID_FAC_N_MAX_P_MT__GEAR
ENDIF
IF LV_N_MAX == 1
THEN TQ_N_MAX_DIF_P = TQ_N_MAX_DIF_P_1
ELSE TQ_N_MAX_DIF_P = 0
ENDIF

IF LV_AT == 1

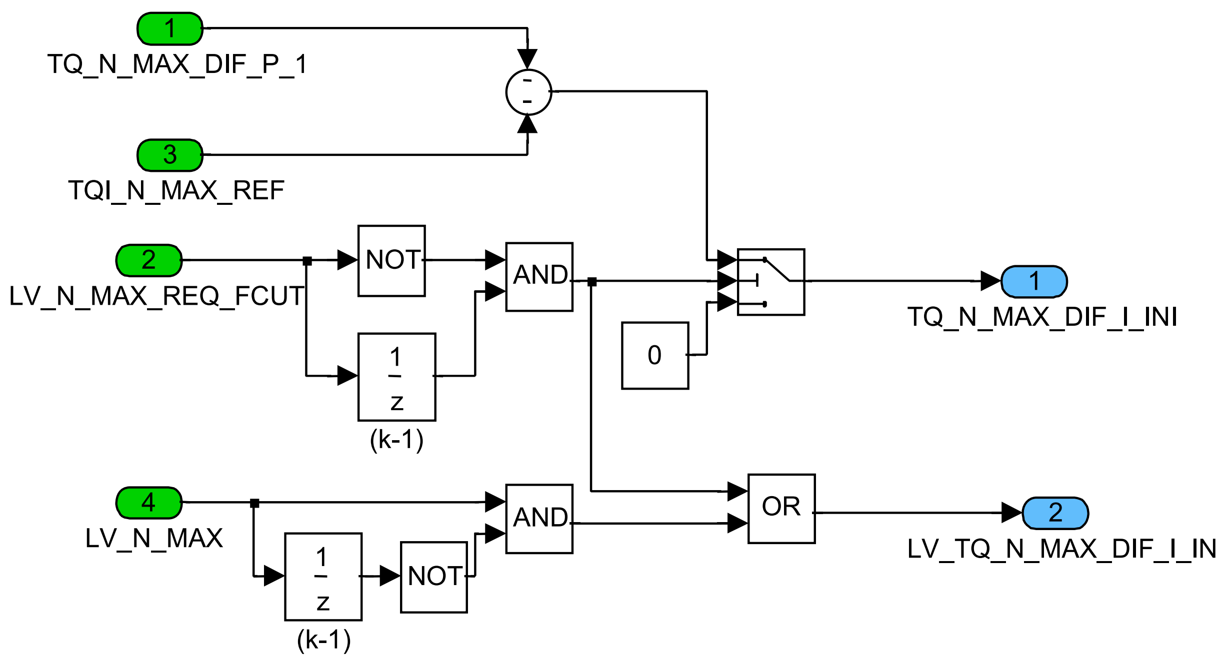
THEN TQ_N_MAX_INP_I_1 = N_DIF_MAX_LIM * ID_FAC_N_MAX_I_AT__GEAR

ELSE TQ_N_MAX_INP_I_1 = N_DIF_MAX_LIM * ID_FAC_N_MAX_I_MT__GEAR

ENDIF

9.24.5.1 Init value for the integrator

Signal flow diagram:



Formula section:

The integrator has to be initialized for the transition of LV_N_MAX from 0 to 1 and for the transition of LV_N_MAX_REQ_FCUT from 1 to 0. In the first case the integrator output is set to zero and in the second case the integrator output has to ensure that the TQI_N_MAX does not show a step (during LV_N_MAX_REQ_FCUT active the TQI_N_MAX is set to zero).

IF (LV_N_MAX_REQ_FCUT$_k$ == 0 **AND** LV_N_MAX_REQ_FCUT$_{k-1}$ == 1)

THEN TQ_N_MAX_DIF_I_INI = - TQ_N_MAX_DIF_P_1 – TQI_N_MAX_REF

ELSE TQ_N_MAX_DIF_I_INI = 0

ENDIF

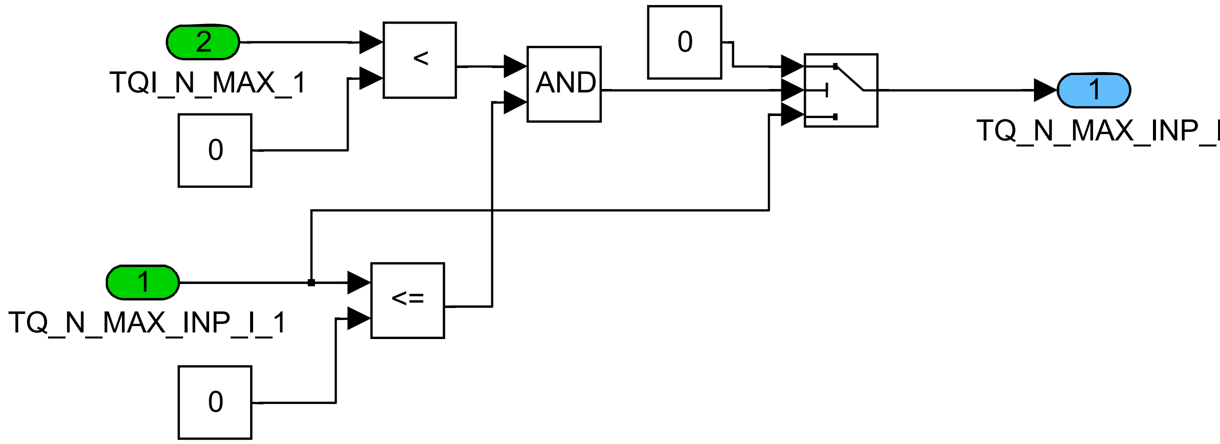
LV_TQ_N_MAX_DIF_I_INI

= (**NOT** LV_N_MAX_REQ_FCUT$_k$ **AND** LV_N_MAX_REQ_FCUT$_{k-1}$)

OR (LV_N_MAX$_k$ **AND NOT** LV_N_MAX$_{k-1}$)

9.24.5.2 Integrator input

Signal flow diagram:



Formula section:

The integrator input has to be set to zero, when TQ_N_MAX_1 is smaller than zero and the calculated input TQ_N_MAX_INP_I_1 is smaller than or equal to zero. This avoids a steadily decrease of the integrator output which could cause problems when reentering the controller.

IF (TQI_N_MAX_1 < 0 **AND** TQ_N_MAX_INP_I_1 <= 0)

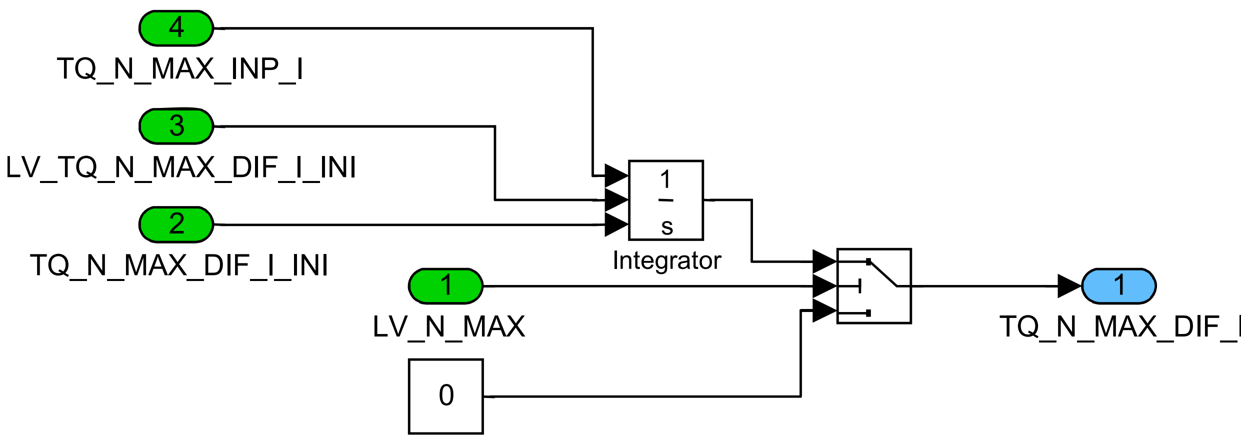
THEN TQ_N_MAX_INP_I = 0

ELSE TQ_N_MAX_INP_I = TQ_N_MAX_INP_I_1

ENDIF

9.24.5.3 Integrator

Signal flow diagram:



Formula section:

IF LV_N_MAX == 1

THEN **IF** (LV_TQ_N_MAX_DIF_I_INI$_k$ == 1 **AND** LV_TQ_N_MAX_DIF_I_INI$_{k-1}$ == 0)

THEN TQ_N_MAX_DIF_I = TQ_N_MAX_DIF_I_INI

ELSE TQ_N_MAX_DIF_I$_k$ = TQ_N_MAX_DIF_I$_{k-1}$ + TQ_N_MAX_INP_I

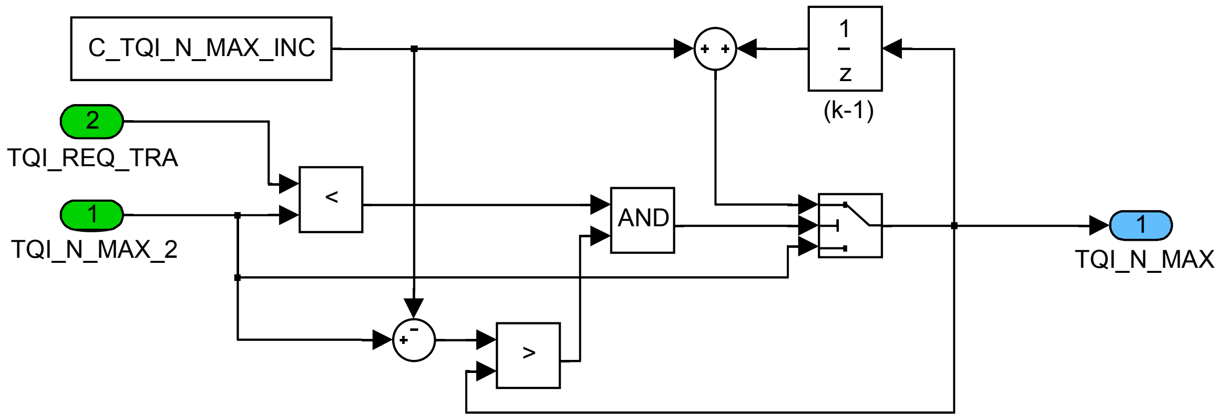
ENDIF

ELSE TQ_N_MAX_DIF_I = 0

ENDIF

9.24.6 Rate limiter in increasing direction

Signal flow diagram:



Formula section:

IF ((TQI_REQ_TRA < TQI_N_MAX_2)

AND ((TQI_N_MAX_2 - C_TQI_N_MAX_INC) > TQI_N_MAX))

THEN $TQI\_N\_MAX_k = TQI\_N\_MAX_{k-1} + C\_TQI\_N\_MAX\_INC$

ELSE TQI_N_MAX = TQI_N_MAX_2

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ADD_N_MAX_FCUT	1	0 .. 1FE0 H	0 .. 8160	1	1/min
Additive threshold to maximum engine speed for activation fuel cut-off all cylinders					
C_CRLC_N_GRD_FIL_1	1	0 .. FFFF H	0 .. 0.99998	15.26E-6	-
N_GRD_FIL low pass filter constant if LV_N_MAX = 0					
C_CRLC_N_GRD_FIL_2	1	0 .. FFFF H	0 .. 0.99998	15.26E-6	-
N_GRD_FIL low pass filter constant if LV_N_MAX = 1					
C_N_GRD_HYS	1	80 .. 7F H	-4096 .. 4064	32	1/min/s
Hysteresis for N_GRD Filtering					
C_TQI_MAX_PAS	1	0 .. 7FFF H	0 .. 1023.97	0.03125	Nm
Passive value for TQI_N_MAX if engine-speed limitation is passive					
C_TQI_N_MAX_INC	1	0 .. 7FFF H	0 .. 1023.97	0.03125	Nm
Maximum increment for the output when limiting increasing rate					
ID_FAC_N_MAX_P_AT__GEAR	9	0 .. FF H	0 .. 7.968	0.03125	Nm/1/min
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
P-control parameter for N_MAX-controller automatic transmission					
ID_FAC_N_MAX_P_MT__GEAR	9	0 .. FF H	0 .. 7.968	0.03125	Nm/1/min
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
P-control parameter for N_MAX-controller manual transmission					
ID_FAC_N_MAX_I_AT__GEAR	9	0 .. FFFF H	0 .. 7.99987793	122.07E-6	Nm/1/min
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
I-control parameter for N_MAX-controller automatic transmission					
ID_FAC_N_MAX_I_MT__GEAR	9	0 .. FFFF H	0 .. 7.99987793	122.07E-6	Nm/1/min
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
I-control parameter for N_MAX-controller manual transmission					
ID_T_N_MAX_PRED_AT__GEAR	9	0 .. FF H	0 .. 7.96875	0.03125	s
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
Prediction time for predicted engine speed for automatic transmission					
ID_T_N_MAX_PRED_MT__GEAR	9	0 .. FF H	0 .. 7.96875	0.03125	s
LDPM_GEAR	9	0 .. FF H	0 ... 255	1	-
Prediction time for predicted engine speed for manual transmission					
IP_N_GRD_THD__N_32	6x1	0 .. FF H	-4096 .. 4064	32	1/min/s
LDP_N_32_N_GRD_THD	6x1	0 .. FF H	0 .. 8160	32	1/min
Engine speed gradient threshold for filtering of basic engine speed gradient					

9.25 Fuel pump relay control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_RLY_EFP	O/V	0...01H	0...1	-	-
Fuel pump relay control					

Input data:

LV_ES	LV_IGK	LV_VS_MAX	LV_INJ_CUT
LV_EFP_EXT_ADJ	LV_ACT_EFP_EXT_ADJ		

FUNCTION DESCRIPTION:

General information:

Recurrence : 40ms

Activation:

The fuel pump is switched **ON** for **C_T_EFP_IGK** during the transition from the control unit initialization to engine operating state engine stopped (LV_ES = 1) with ignition key **ON** (LV_IGK = active).
The fuel pump relay can be also switched by external adjustment (LV_EFP_EXT_ADJ = 1); If external adjustment is active, the relay is controlled by LV_ACT_EFP_EXT_ADJ.

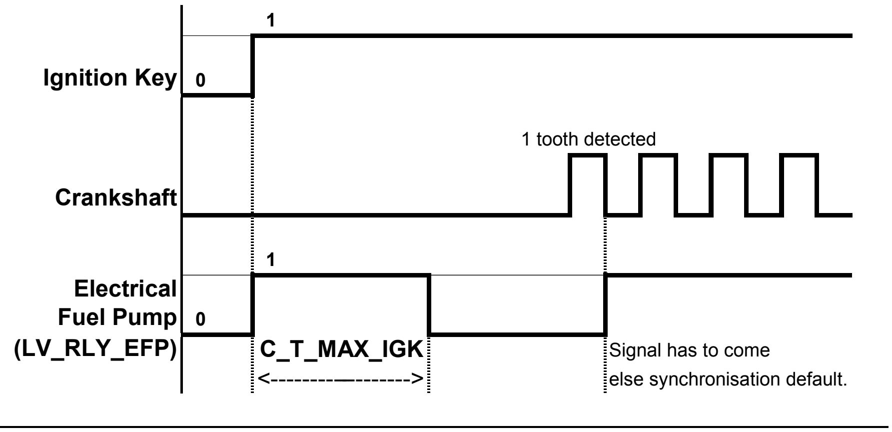
Dectivation:

Following ignition key **OFF** (LV_IGK = passive), the fuel pump is switched **OFF** after a waiting period **C_T_EFP**.

The fuel pump can also be deactivated, after a waiting period **C_T_EFP**, in the following case :

- vehicule speed limitation (LV_VS_MAX = 1) **AND** all cylinder switched off (LV_INJ_CUT = 1)

Description:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_EFP_IGK	1	0... FFFFH	0... 2621.4	0.04	sec.
Time to switch ON the electric fuel pump relay after ignition key ON.					
C_T_EFP	1	0... FFFFH	0... 2621.4	0.04	sec.
Time delay to shut - off the electric fuel pump relay after ignition key OFF.					

9.26 Determination of fuel consumption

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit.
FCO_SUM	V/O	0...FFFFFFFFH	0...549763678,2	0,128	µl
accumulated fuel consumption					
FCO_SUM_VRU	V	0...FFFFFFFFH	0...549763678,2	0,128	µl
accumulated fuel consumption while vehicle running					
FCO_SUM_TEMP	-	0...FFFFH	0...8388,6	0,128	µl
fuel consumption per calculalation cycle					
FCO_KPL	-	0...FFFFH	0...655.35	0,01	Km/L
Instant fuel consumption rate in Km/L					
FCO_KPL_MMV	V	0...FFFFH	0...655.35	0,01	Km/L
Moving mean value of instant fuel consumption rate in Km/L					
FCO_KPL_VRU	V	0...FFFFH	0...655.35	0,01	Km/L
Total fuel consumption rate in Km/L while vehicle running					
FCO_KPL_TOT	V	0...FFFFH	0...655.35	0,01	Km/L
Total fuel consumption rate in Km/L whiel present driving cycle including vehicle stop					
FCO_SUM_DIF	V/O	0...FFFFH	0...8388,6	0,128	µl
FCO_SUM_10 difference every 10 ms					
FCO_SUM_10	V/O	0...FFFFFFFFH	0...549763678,2	0,128	µl
accumulated fuel consumption(10ms task based on FCO_SUM)					

Input data:

LV_IGK	LV_ES	MFF_SP_MV	EFF_SCC_AV
T_SEG_AV	VS	DIST_FCO	

FUNCTION DESCRIPTION:

General information:

The accumulated fuel consumption is calculated based on the setpoint of the injection. This value (MFF_SP in mg/stk.) is converted to the injected fuel quantity in µl while using the conversion factor C_FCO_STND and simultaneous correction with the actual efficiency of fuel cut-off.

FCO_KPL_TOT, overall fuel consumption rate in Km/L , is calculated based on the vehicle moving distance after engine start and accumulated fuel consumption.

FCO_KPL_VRU , overall fuel consumption rate in Km/L while vehicle running, is calculated based on the vehicle moving distance after engine start and accumulated fuel consumption only while vehicle running.

FCO_KPL, instant fuel consumption rate in Km/L, is calculated when vehicle is running (VS > 0) , which can be assumed using instant fuel consumption quantity (FCO_SUM_TEMP) and converted vehicle running distance from vehicle speed information (VS).

Moving mean value of FCO_KPL, FCO_KPL_MMV, is obtained by filtering FCO_KPL.

Application conditions:

Initialisation: At ECU reset :

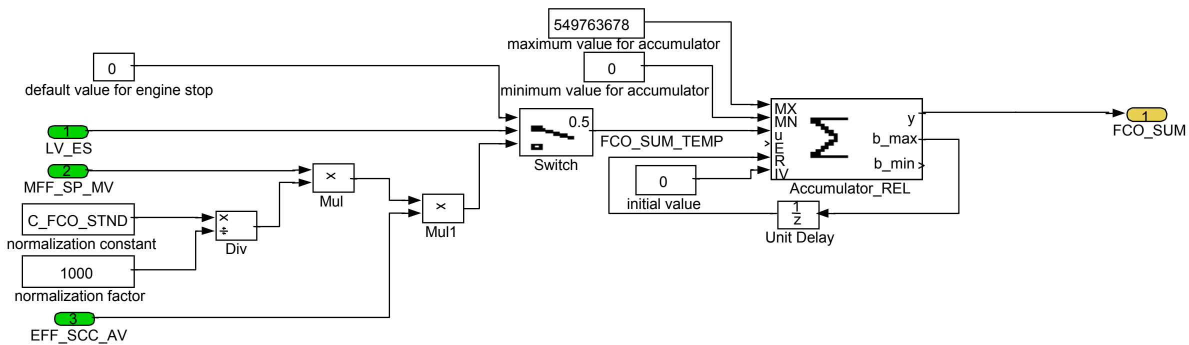
FCO_SUM = FCO_KPL = 0

FCO_SUM_10 = FCO_SUM_DIF = 0

FCO_KPL_VRU = FCO_KPL_TOT = 0

Update rate:	segment synchronous	: Calcuation of fuel consumption
	10msec	*: Calcuation of FCO_SUM_10/DIF*
	1000msec	: Calculation of fuel consumption rate
Activation:	LV_IGK = 1	
Deactivation:	LV_IGK = 0	

Signal flow:



Formula section:

 # Segment synchronous:

IF LV_ES = 0

THEN $FCO\_SUM\_TEMP_N$ = MFF_SP_MV • C_FCO_STND /1000 • EFF_SCC_AV

If VS > 0

Then FCO_SUM_VRU(n) = FCO_SUM_VRU(n-1) + FCO_SUM_TEMP(n)

EndIf

If FCO_SUM_TEMP(n) > 0

And EFF_SCC_AV = 1

Then FCO_KPL(n) = VS * T_SEG_AV / FCO_SUM_TEMP(n) / 3600 *1000000

EndIf

ELSE $FCO\_SUM\_TEMP_N$ = 0 µl

EndIf

$FCO\_SUM_N = (FCO\_SUM_{N-1} + FCO\_SUM\_TEMP_N)$

FCO_KPL_MMV(n) = FCO_KPL_MMV(n-1)

+ C_CRLC_FCO_KPL_MMV * (FCO_KPL(n) – FCO_KPL_MMV(n-1))

 # 10ms task:

FCO_SUM_10 is updated every 10 ms based on FCO_SUM.

FCO_SUM_10 = FCO_SUM

FCO_SUM_DIF = $FCO\_SUM\_10_{N}$ - $FCO\_SUM\_10_{N-1}$

 # 1000msec Task:

If FCO_SUM > 0

And FCO_SUM_VRU > 0

And DIST_FCO > 0

Then // All units are Km/L

FCO_KPL_VRU = DIST_FCO / FCO_SUM_VRU * 1000000

FCO_KPL_TOT = DIST_FCO / FCO_SUM * 1000000

EndIf

Calibration data:

Name	Dim	Hex. Limit	Phys. Limits	Resol.	Unit.
C_FCO_STND	1	0…0BCBH	0…3019	1	µl/g
standardisation constant for fuel consumption					
C_CRLC_FCO_KPL_MMV	1	0…FFH	0…0.996	3.9 E-3	-
Correlation constant for FCO_KPL_MMV calculation					

9.27 Engine and Air Intake System Protection from Overpressure

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_IN_PROT	O/V	0...1H	0...1	1	-
Intake overpressure protection active					
PUT_AMP_DIF	O/V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Pressure upstream throttle difference to ambient pressure					
MAP_AMP_DIF_PRED_GRD	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Predicted value of the rise of the manifold air pressure difference to ambient pressure during the prediction time					
CTR_ABC_IM_PROT	V	0...FFH	0...255	1	-
Anti bounce counter for turbo charger deactivation because of too high charge air pressure					
PUT_AMP_DIF_PRED	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Predicted value of the pressure upstream throttle difference to ambient pressure after the prediction time					
T_MAP_AMP_DIF_PRED	V	0...FFH	0...2.55E+3	10	ms
Prediction time for calculation of predicted relative pressure of the intake manifold to ambient					
LV_IN_PROT_MAP	V	0...1H	0...1	1	-
Intake overpressure protection active caused by manifold air pressure too high					
T_PUT_AMP_DIF_PRED	V	0...FFH	0...2.55E+3	10	ms
Prediction time for calculation of predicted relative pressure of pressure upstream throttle to ambient pressure					
LV_IN_PROT_PUT	V	0...1H	0...1	1	-
Intake overpressure protection active caused by pressure upstream throttle too high					
MAP_AMP_DELTA	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
pressure delta between MAP_AMP_DIF and MAP_AMP_DIF_IN_PROT					
MAP_AMP_DIF	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Manifold air pressure difference to ambient pressure					
MAP_AMP_DIF_PRED	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Predicted value of the manifold air pressure difference to ambient pressure after the prediction time					
PUT_AMP_DIF_PRED_GRD	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Predicted value of the rise of pressure upstream throttle difference to ambient pressure during the prediction time					
PUT_AMP_DELTA	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Pressure delta between PUT_AMP_DIF and PUT_AMP_DIF_IN_PROT					

Input data:

LC_TCHA_CONF	MAF_KGH_TCHA	MAP	AMP
PUT	LV_ST_END		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_ABC_MAX_IM_PROT	1	0...FFH	0...255	1	-
Anti bounce counter threshold for turbo charger deactivation because of too high charge air pressure					
C_PRS_AMP_DIF_HYS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Common pressure hysteresis for LV_IN_PROT_MAP and LV_IN_PROT_PUT determination					
LC_IN_PROT_ACT	1	0...1H	0...1	1	-
logical bit for usage of Intake overpressure protection active controlled via application system					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_IN_PROT_AS	1	0...1H	0...1	1	-
Intake overpressure protection active application system					
IP_MAP_AMP_DIF_IN_PROT	8	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_MAF_KGH_1_CHRG	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Conversion table of manifold air pressure above 1013hPa threshold for intake overpressure protection					
IP_PUT_AMP_DIF_IN_PROT	8	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDPM_MAF_KGH_1_CHRG	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Conversion table of pressure upstream throttle threshold above 1013hPa for intake overpressure protection					
IP_T_MAP_AMP_DIF_PRED	8	0...FFH	0...2.55E+3	10	ms
LDPM_MAF_KGH_1_CHRG	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Conversion table of prediction time of MAP_AMP_DIF gradient for intake overpressure protection					
IP_T_PUT_AMP_DIF_PRED	8	0...FFH	0...2.55E+3	10	ms
LDPM_MAF_KGH_1_CHRG	8	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Conversion table of prediction time of difference pressure upstream throttle minus ambient pressure for protectn					

9.27.1 CHRG_REQGNPROTP0

To prevent the air intake system downstream turbo charger compressor from being annihilated by too high pressure difference inside (manifold air pressure) to outside (ambient air pressure) the following function was developed.

The two different conversion tables for IP_MAP_AMP_DIF_IN_PROT and IP_PUT_AMP_DIF_IN_PROT will contain the pressure values **above ambient pressure at nominal ambient pressure (1013hPa)** which the intake manifold respectively the air intake system between turbo charger compressor and throttle can bear to contain. It is dependent on the air mass flow because at different air mass flows different safety ranges between the adjusted value and a possible damage of components may be desired if different instationary effects occur. This also takes into account that the air intake system upstream and downstream throttle possibly have different maximum pressure values which they can resist.

The pressure prediction depends on the air mass flow because a higher mass air flow will lead to a faster increase of the pressure inside the air intake system at the same failure cause. The prediction time is a measure for the time the functionality looks towards the future with the current pressure gradient. So if a long prediction time is adjusted, the air intake protection function will be very sensible. (Refer to signal overview!)

Each the pressure in the intake manifold and upstream throttle will be predicted in the shown manner. If the predicted pressure, after time T_xxx_AMP_DIF_PRED has expired, exceeds the adjustable threshold xxx_AMP_DIF_IN_PROT the logical variable LV_IN_PROT_xxx is set valid.

Dependent on the time span you want to predict the pressure conditions, the measured pressure value differences (xxx_AMP_DIF_GRD) are weightened differently. In addition to the current measured value a predicted pressure value (xxx_AMP_DIF_PRED) is calculated.

Function Description

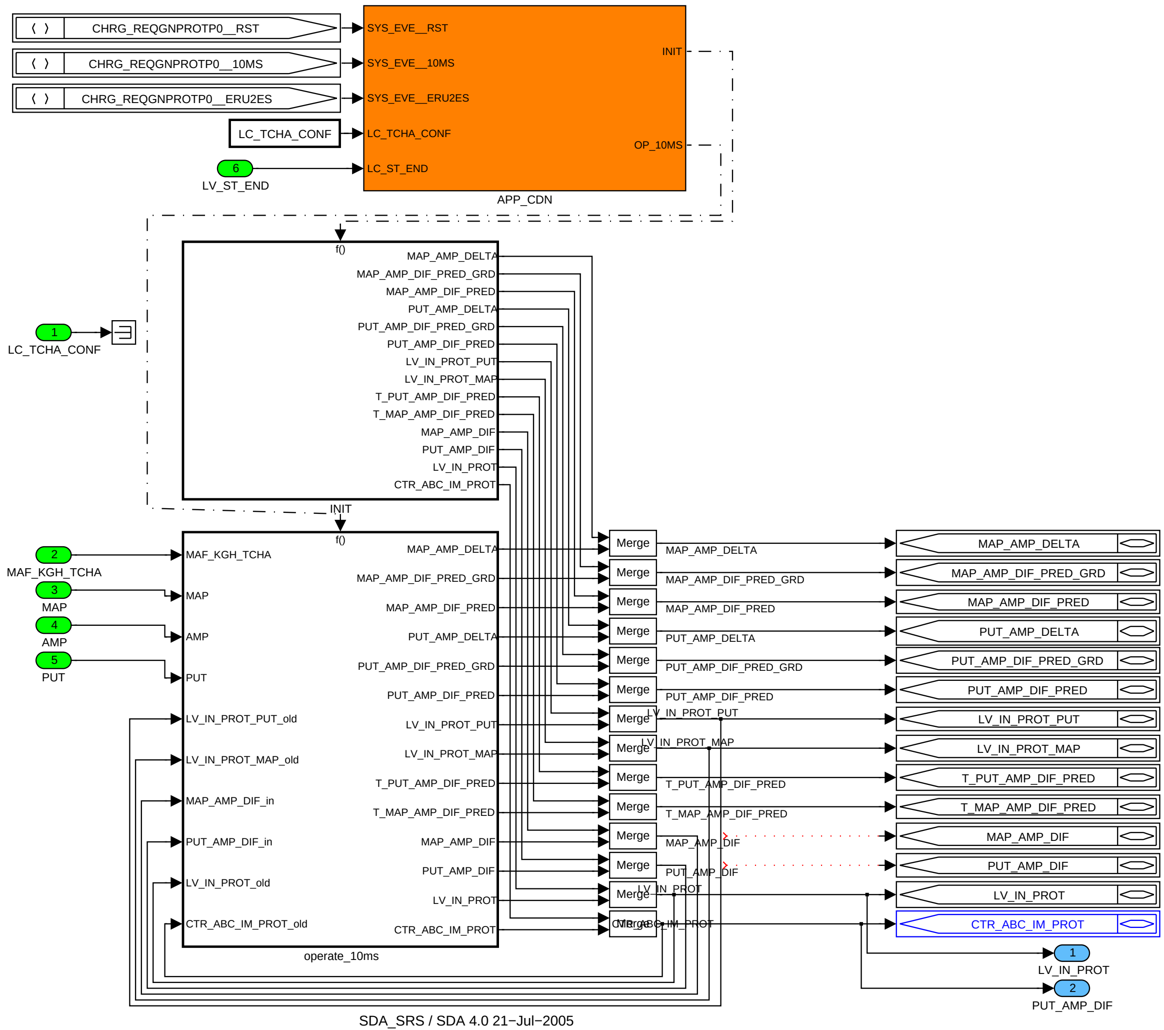


Figure 10 CHRG_REQGNPROTP0

9.27.1.1 SUBFUNCTION: INIT

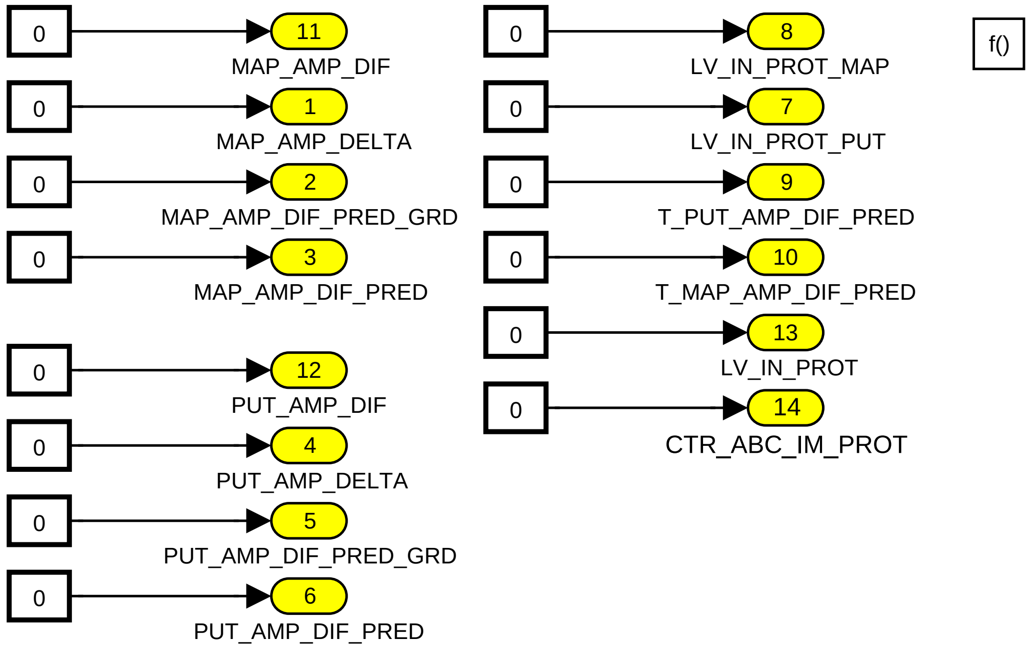


Figure 11 CHRG_REQGNPROTP0/ INIT

9.27.1.2 SUBFUNCTION: operate_10ms

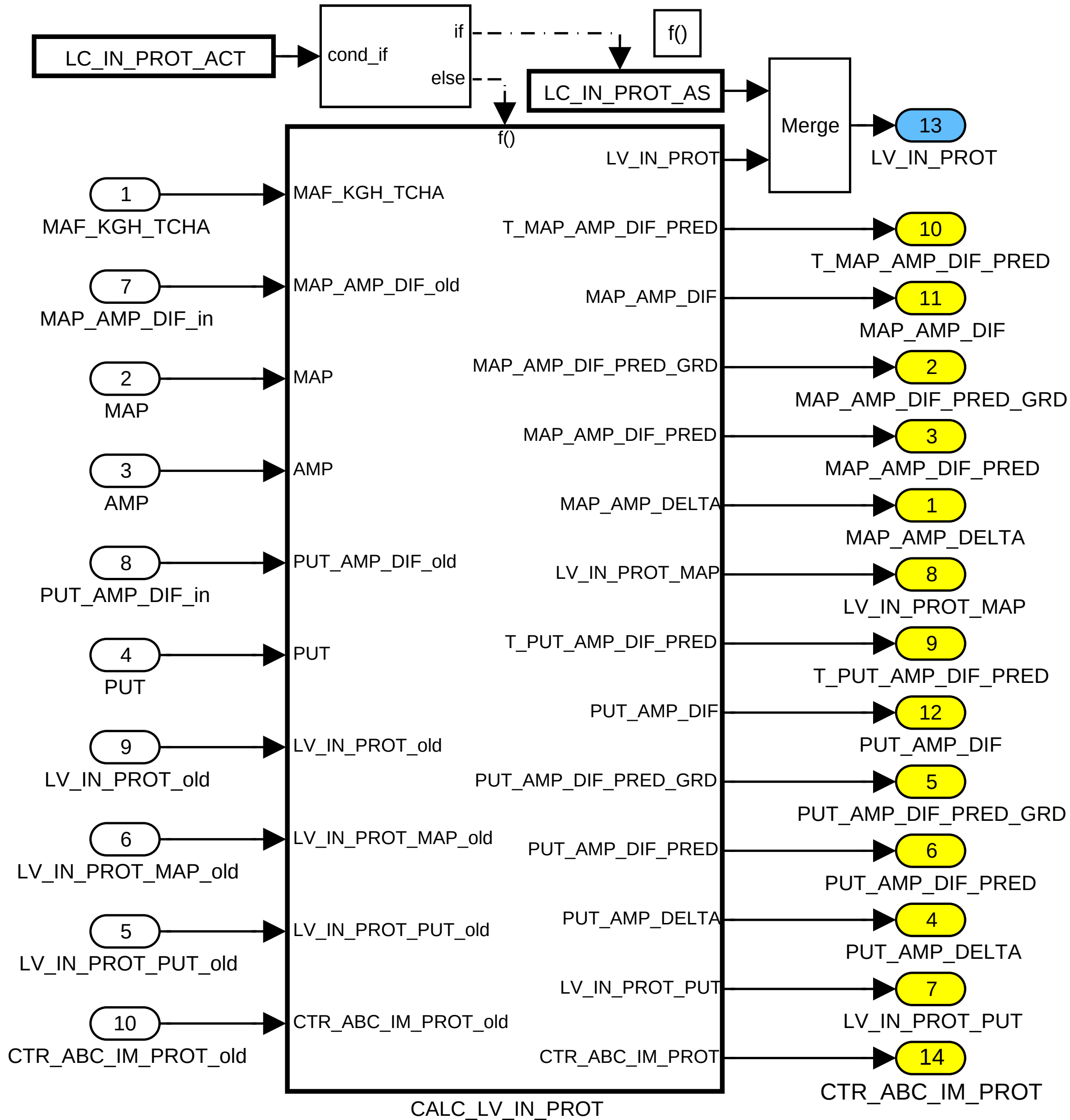


Figure 12 CHRG_REQGNPROTP0/ operate_10ms

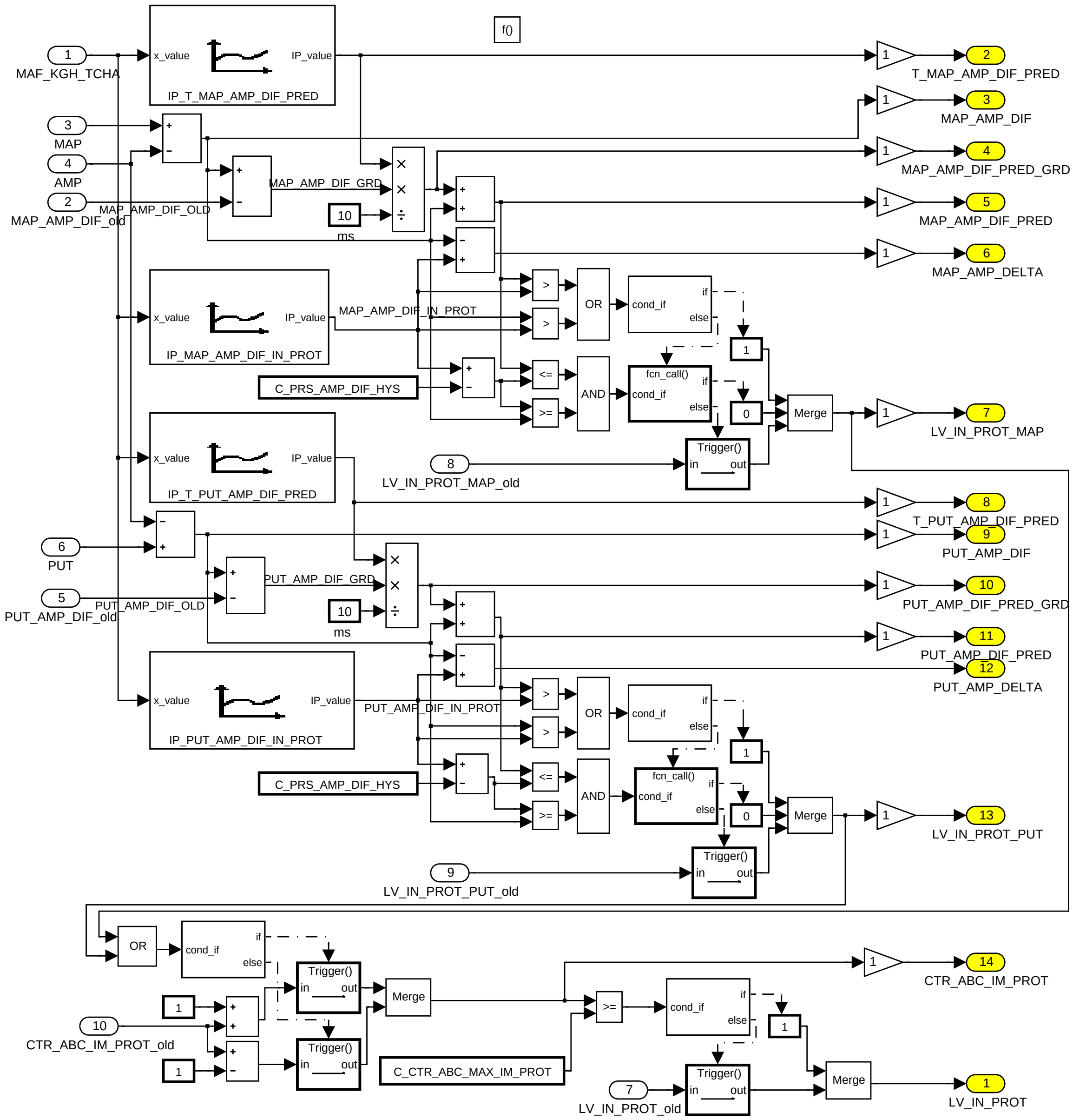


Figure 13 CHRG_REQGNPROTP0/ operate_10ms/ CALC_LV_IN_PROT

9.28 Engine Stop Function for Dual Mass Flywheel Oscillation

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
T_DMF	V	0...FFH	0...2550	10	ms
Timer for engine speed oscillation detection (decrementation type)					
CTR_DMF	V	0...FFH	0...255	1	-
Counter for engine speed oscillation detection					
LV_ENG_OFF_DMF_INJ	V/O	0...1H	0...1	1	-
Injection Shut off request for all cylinders because of dual mass flywheel oscillation					
LV_ENG_OFF_DMF_IGN	V/O	0...1H	0...1	1	-
Ignition Shut off request for all cylinders because of dual mass flywheel oscillation					
LV_ENG_OFF_DMF_TPS_ISA	V/O	0...1H	0...1	1	-
TPS/ISA shut off request because of dual mass flywheel oscillation					

Input data:

N_TOOTH	LV_ES	LV_ST	LV_AT
TCO			

Function description:

General information:

To avoid a strong oscillation of the dual mass flywheel while operating the engine close to the resonance frequency for a longer time it could be necessary to shut-down the engine.

For this the injection (and additionally the ignition) can be deactivated until next ES. The function is only applied to MT vehicles.

Description:

The oscillation detection is done by checking the engine speed (high resolution engine speed N_TOOTH) against exceedance of rpm window. If the engine speed exceeds a minimum threshold C_N_MIN_DMF then a timer is initialized with C_T_DMF and decremented. If now the engine speed exceeds a maximum threshold C_N_MAX_DMF before the timer reached 0 then the DMF oscillation counter CTR_DMF is incremented.

C_N_MIN_DMF and C_N_MAX_DMF should be calibrated around the DMF resonance rpm.

If the counter reaches a maximum threshold the engine is shut down by injection switch-off until the next engine stop detection. Additionally the ignition can be switched-off. TPS/ISA can be closed In order to improve engine shut-off pattern (LC_ENG_OFF_DMF_TPS_ISA = 1).

Once the engine speed leaves the oscillation range above C_N_MAX_RST_DMF the counter is reset to 0.

At low temperatures the dual mass flywheel typically does not have a resonance problem, so the function can be deactivated.

Recurrence: 10 ms

Application conditions:

Activation:

If LV_ES = 0 **and** LV_ST = 0 *(All engine operating state except ES or ST)*

and LV_AT = 0 *(only for M/T vehicle)*

and *TCO > C_TCO_MIN_DMF* (only above certain TCO threshold)

Initialisation (with LV_ES --> LV_ST):

LV_ENG_OFF_DMF_INJ = 0

LV_ENG_OFF_DMF_IGN = 0

LV_ENG_OFF_DMF_TPS_ISA = 0

T_DMF = 0

CTR_DMF = 0

Formula section:

If N_TOOTH > C_N_MAX_RST_DMF *(engine speed above oscillation range)*

or LV_ES = 1 *(engine stopped)*

then CTR_DMF = 0 *(reset oscillation counter)*

and T_DMF = 0 *(reset oscillation timer)*

else *(engine speed within oscillation range, no ES)*

if N_TOOTH < C_N_MIN_DMF *(engine speed below min. threshold)*

then (re-)initialize timer T_DMF with C_T_DMF *(start time ctr. for max. thd oscillation)*
(as long as N_TOOTH is below the threshold the timer has to be restarted again and again in order to count the time between passing the MIN threshold and reaching the MAX threshold)

else decrement T_DMF *(decrement timer until reaching 0)*

if N_TOOTH > C_N_MAX_DMF *(engine speed above max. threshold)*

and T_DMF > 0 *(max threshold exceeded before timer reach 0)*

then increment CTR_DMF *(oscillation event detected)*

and T_DMF = 0 *(reset T_DMF for next oscillation)*

if CTR_DMF > C_CTR_MAX_DMF *(number of oscillations above threshold)*

then LV_ENG_OFF_DMF_INJ = 1 *(deactivate injection irreversible up to next ES)*

if LC_ENG_OFF_DMF_IGN = 1 *(additional ignition shut-off configured)*

then LV_ ENG_OFF_DMF_IGN = 1 *(additionally deactivate ignition)*

if LC_ENG_OFF_DMF_TPS_ISA = 1

then

LV_ENG_OFF_DMF_TPS_ISA = 1

End if

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_N_MAX_RST_DMF	1	0...FFH	0...8160	32	1/min
Maximum engine speed range for dual mass flywheel oscillation					
C_N_MAX_DMF	1	0...FFH	0...8160	32	1/min
maximum engine speed threshold for oscillation detection					
C_N_MIN_DMF	1	0...FFH	0...8160	32	1/min
minimum engine speed threshold for oscillation detection					
C_T_DMF	1	0...FFH	0...2550	10	ms
Initialization value for time counter before decrementation for engine speed oscillation detection					
C_CTR_MAX_DMF	1	0...FFH	0...255	1	-
Maximum number of oscillation events until engine shut down					
C_TCO_MIN_DMF	1	0...FEH	-48...142,5	0,75	°C
Minimum coolant temperature threshold to perform engine shut down for DMF oscillation					
LC_ENG_OFF_DMF_IGN	1	0...1H	0...1	1	-
Allow additional Ignition Shut-Off for engine shut down with DMF oscillation					
LC_ENG_OFF_DMF_TPS_ISA	1	0...1H	0...1	1	-
Logical constant for enabling engine stop function with TPS/ISA control at engine shut-off (0:Disable, 1:Enable)					

9.29 Coolant temperature gradient

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO_GRD	V/O	8000...7FFFH	-96...95.99707	2.93E-03	[°C/s]
Actual valid coolant temperature gradient					
TCO_MMV_TMP	-	0...FE00H	-48...142.5	2.93E-03	[°C]
Temporary coolant temperature moving mean value					
TCO_GRD_MMV	V	8000...7FFFH	-96...95.99707	2.93E-03	[°C/s]
Difference between the last and the actual temporary coolant temperature moving mean value					
TCO_GRD_MMV_TMP	-	8000...7FFFH	-96...95.99707	2.93E-03	[°C/s]
Temporary coolant temperature gradient moving mean value					

Input data:

TCO			

FUNCTION DESCRIPTION:

General information:

The control of the coolant temperature is using the coolant temperature gradient as input to avoid temperature overshooting and oscillations of the coolant temperature.

Because of the rough resolution of the coolant temperature value a filter (PT1 filter_1) is used to smooth the signal. The difference between the actual moving mean value and the last valid moving mean value is calculated at every second. In this case, the gradient values are always related to the physical unit Celsius per second [°C/s]. An additional filter (PT1 filter_2) in combination with an amplifier is generating the coolant temperature gradient for calibration issues.

Application conditions:

Initialisation:

If Reset

then TCO_MMV_TMP = TCO

$TCO\_MMV\_TMP_{(n-1)}$ = TCO

TCO_GRD_MMV = 0

TCO_GRD_MMV_TMP = 0

$TCO\_GRD\_MMV\_TMP_{(n-1)}$ = 0

TCO_GRD = 0

endif

Recurrence: 1000 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

Calculation of the coolant temperature moving mean value:

TCO_MMV_TMP = TCO * C_CRLC_TCO_MMV +

$TCO\_MMV\_TMP_{(n-1)}$ * (1 – C_CRLC_TCO_MMV)

Calculation of the difference between the moving mean values:

TCO_GRD_MMV = ($TCO\_MMV\_TMP_{(n)}$ - $TCO\_MMV\_TMP_{(n-1)}$) / 1 sec.

Calculation of the temperature gradient moving mean value:

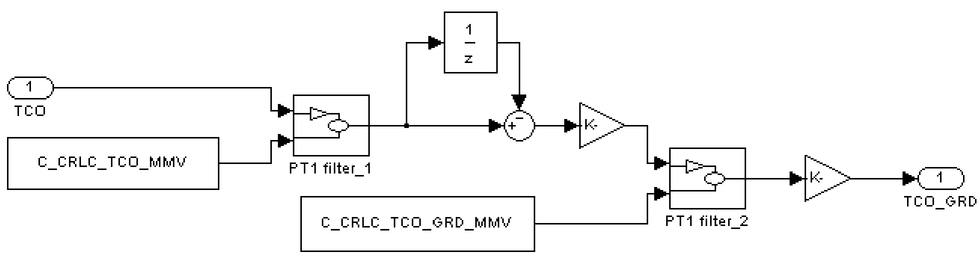
TCO_GRD_MMV_TMP = TCO_GRD_MMV * C_CRLC_TCO_GRD_MMV +

$TCO\_GRD\_MMV\_TMP_{(n-1)}$ * (1 – C_CRLC_TCO_GRD_MMV)

Calculation of the coolant temperature gradient:

TCO_GRD = TCO_GRD_MMV_TMP * C_FAC_TCO_GRD

Signal flow diagram:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_TCO_MMV	1	0...FFH	0...0.99609	3.91E-03	[-]
Correlation constant for the coolant temperature moving mean value calculation					
C_CRLC_TCO_GRD_MMV	1	0...FFH	0...0.99609	3.91E-03	[-]
Correlation constant for the coolant temperature gradient moving mean value calculation					
C_FAC_TCO_GRD	1	0...FFH	0...255	1	[-]
Factor for amplifying of the calculated coolant temperature gradient					

9.30 ENTE Manager 01 – project specific tasks

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
POW_OUT_ECF	O/V	0…FFH	0…99.61	0.39	%
Cooling fan power output of all available fan(s) at the vehicle					
LV_ERR_ECF[NC_ECF_NR]	O/V	0...1H	0...1	1	-
Boolean for electronic controlled cooling fan error					

Input data:

LV_ERR_ECF_EL_PWM	POW_OUT_ECF_PWM	NC_ECF_NR	NC_ECF_CONF
LV_ERR_ECF_EL_RLY	POW_OUT_ECF_RLY	CONF_CFA	

FUNCTION DESCRIPTION:

General information:

This manager specifies the sequencing of project specific ENTE tasks.

Description:

Within the aggregate ENTE, the cooling fan functions (acquisition, control, diagnosis, torque loss) are handled project specific with use of HOOK modules. Because different projects use different update rates for the functions, it is necessary to handle the function calls specific within this module.

Application conditions:

Recurrence: see formula section below

Activation: at every engine operating state

Deactivation: -

Formula section:

#IF NC_ECF_CONF = 0 **(** control of cooling fan(s) only by Relay(s) **)**

Tasks
No action

#ENDIF

#IF NC_ECF_CONF = 1 **(** control of cooling fan(s) only by PWM(s) **)**

Tasks
No action

#ENDIF

#IF NC_ECF_CONF = 2 **(** control of cooling fan(s) by Relay(s) **or** PWM(s) **)**

RESET - Task
POW_OUT_ECF = 0% LV_ERR_ECF[NC_ECF_NR] = 0

IF CONF_CFA = 0

THEN (Relay version)

40ms - Task
ENTE_IFINFECF0 (Acquisition of electronically controlled cooling fan signal – “*Acquisition of cooling fan signal (RLY version)*” chapter 1.1)
ENTE_ACCTLECF0 (Cooling and condenser fan control)
POW_OUT_ECF = POW_OUT_ECF_RLY

100ms - Task
ENTE_REQGNTQECF0 (Torque Loss - Electronic cooling fan)

2000ms - Task
ENTE_OUTDGECF0 (Cooling fan diagnosis)
LV_ERR_ECF[NC_ECF_NR] = LV_ERR_ECF_EL_RLY

ELSE (PWM version)

40ms - Task
ENTE_ACCTLECF0 (Cooling and condenser fan control)

100ms - Task
ENTE_REQGNTQECF0 (Torque Loss - Electronic cooling fan)

200ms - Task
ENTE_IFINFECF0 (Acquisition of electronically controlled cooling fan signal – “*Acquisition of cooling fan signal (PWM version)*” chapter 1.2)
ENTE_ACCTLECF1 (Cooling and condenser fan control)
POW_OUT_ECF = POW_OUT_ECF_PWM

1000ms - Task
ENTE_OUTDGECF1 (Cooling fan diagnosis)
LV_ERR_ECF[NC_ECF_NR] = LV_ERR_ECF_EL_PWM

ENDIF

#ENDIF

9.31 Determination of Battery State

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_QBAT_INI_END	V	0... 1H	0... 1	1	[-]
Flag indicating the calculation of QBAT_INI was performed					
LV_QBAT_REF_END	V	0... 1H	0... 1	1	[-]
Flag indicating a new calculation of QBAT_REF could be finished					
LV_VB_CYC_ENA	O/V/S	0... 1H	0... 1	1	[-]
Battery refreshment cycle activated					
QBAT	V/S	0... 7FFH	0... 120	0.0586224	[Ah]
Charge of Battery					
QBAT_CUM	O/V	80000000... 7FFFFFFFH	-120 ...120	55.8794e-9	[Ah]
Accumulated Charge of Battery					
QBAT_INI	V	0... 7FFH	0... 120	0.0586224	[Ah]
Charge of Battery after Initialisation					
QBAT_REF	V	0... 7FFH	0... 120	0.0586224	[Ah]
Reference determination for Charge of Battery					
QBAT_TOT	O/V/S	0... 7FFFFFFFH	0... 120	55.8794e-9	[Ah]
Total amount of Charge and Discharge since last SOC calculation at initialisation could be performed					
SOC	O/V	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Battery State of Charge					
SOC_INI	O/V	0... FFH	0... 100	0.3921569	[%]
Battery State of Charge at Initialisation					
T_QBAT_REF	V	0... 2000H	0... 81.92	0.01	[s]
Timer to check conditions for Q_BAT_REF detection fullfilled					
T_QBAT_TOT	O/V/S	0... FFFFH	0... 65535	1	[min]
Timer since last SOC calculation at initialisation could be performed					
VB_H_MMV_TMP	V	0... 3FFH	0... 26	0.0254154	[V]
Temporary voltage for SOC determination					
VB_OCV	V	0... 3FFH	0... 26	0.0254154	[V]
Open Circuit voltage (calculated only after Initalisation and for Reference determination)					

Input Data:

CUR_BAT_EFF	CUR_BAT_EFF_MMV	CUR_BAT_INI	CUR_BAT_LIN_FIRST_VLD
CONF_BAT	CONF_BAT_LIN	LV_ES	LV_IGK
LV_SENS_BAT_LIN_VLD	TBAT_CLC	TBAT_INI	TBAT_LIN_FIRST_VLD
T_ES	VB_H_INI	VB_H_MMV	VB_LIN
VB_LIN_FIRST_VLD	SOC_LIN	QBAT_MAX_LIN	LV_ERR_BAT_SENS
LV_VB_SENS_LIN_INI_END	LV_SENS_LIN_READY		

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CUR_BAT_DIF_MAX	1	0... FFFFH	0... 240	3.66217e-3	[A]
Maximum current difference for performing QBAT_REF determination					
C_CUR_BAT_INI_MAX	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Battery current threshold at initialization to allow new SOC INI calculation					
C_QBAT_CUM_SUB	1	80000000... 7FFFFFFFH	-120 ...120	55.8794e-9	[Ah]
Substitute value for QBAT_CUM					

C_QBAT_DFT	1	0... 7FFH	0... 120	0.0586224	[Ah]
Default value for Battery charge (at first power up)					
C_QBAT_MAX	1	0... 7FFH	0... 120	0.0586224	[Ah]
Maximum value for Battery charge fully charged					
C_QBAT_TOT_CYC	1	0... 7FFFFFFFH	0... 120	55.8794e-9	[Ah]
Total amount of charge and discharge to start refresh cycle					
C_QBAT_TOT_SUB	1	0... 7FFFFFFFH	0... 120	55.8794e-9	[Ah]
Substitute value for QBAT_TOT					
C_SOC_AS	1	FF01... FFH	-100 ...100	0.3921569	[%]
Application value to be added to SOC value					
C_SOC_INI_SUB	1	0... FFH	0... 100	0.3921569	[%]
Substitute value for SOC_INI					
C_SOC_MAX_CYC	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Maximum of SOC to stop the refreshment cycle					
C_SOC_SUB	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Substitute value for SOC					
C_T_QBAT_REF	1	0... 2000H	0... 81.92	0.01	[s]
Timer to be decremented before QBAT_REF determination is performed					
C_T_QBAT_TOT_CYC	1	0... FFFFH	0... 65535	1	[min]
Total time to start refresh cycle					
C_T_QBAT_TOT_SUB	1	0... FFFFH	0... 65535	1	[min]
Substitute value for T_QBAT_TOT					
C_VB_H_DIF_MAX	1	0... 3FFH	0... 26	0.0254154	[V]
Maximum voltage difference for performing QBAT_REF determination					
IP_QBAT_FAC_INI	12	0... FFFFH	0... 0.999985	15.3186e-6	[-]
LDPM_V_OCV_IP_QBAT_FAC	12	0... 3FFH	0... 26	0.0254154	[V]
Factor for QBAT_INI weightening					
IP_QBAT_FAC_REF	12	0... FFFFH	0... 0.999985	15.3186e-6	[-]
LDPM_V_OCV_IP_QBAT_FAC	12	0... 3FFH	0... 26	0.0254154	[V]
Factor for QBAT_REF weightening					
IP_T_ES_SOC_INI	8	0... FFFFH	0... 65535	1	[min]
LDP_TBAT_INI_T_ES_SOC_INI	8	0... FEH	-48... 142.5	0.75	[°C]
Engine off time threshold to allow new SOC INI calculation					
IP_T_ES_SOC_INI_LIN	8	0... FFFFH	0... 65535	1	[min]
LDP_TBAT_LIN_FIRST_VALID_T_ES	8	0... FEH	-48... 142.5	0.75	[°C]
Engine off time threshold to allow new SOC INI calculation with LIN configuration					
IP_VB_POL_INI	12*12	0... 3FFH	0... 26	0.0254154	[V]
LDP_CUR_BAT_INI_VB_POL	12	0... FFFFH	-120 ...120	3.66217e-3	[A]
LDP_TBAT_INI_VB_POL	12	0... FEH	-48... 142.5	0.75	[°C]
Initialisation Map for Polarisation Voltage					
IP_VB_POL_LIN_FIRST_VLD	12*12	0... 3FFH	0... 26	0.0254154	[V]
LDP_CUR_BAT_LIN_FIRST_VLD_VB_PO	12	0... FFFFH	-120 ...120	3.66217e-3	[A]
LDP_TBAT_LIN_FIRST_VLD_VB_POL	12	0... FEH	-48... 142.5	0.75	[°C]
Initialisation Map for Polarisation Voltage					
IP_VB_POL_REF	12*12	0... 3FFH	0... 26	0.0254154	[V]
LDP_CUR_BAT_EFF_MMV_VB_POL	12	0... FFFFH	-120 ...120	3.66217e-3	[A]
LDP_TBAT_CLC_VB_POL	12	0... FEH	-48... 142.5	0.75	[°C]
Initialisation Map for Polarisation Voltage					
LC_VB_CYC_ENA_SUB	1	0... 1H	0... 1	1	[-]
Substitute value for battery refreshment cycle activated					
LC_VB_SOC_SUB	1	0... 1H	0... 1	1	[-]
Usage of substitute calibration value (typical value 0 : not actif)					

General Information

general specification

The Battery state of charge is the ratio between the actual battery charge and the maximum charge.
The determination of the actual battery charged is based on the Integration of the current delivered by the battery Sensor over time, Positive current (= battery is charged) and negative current (= battery is discharged). The accumulated Charge value is added to the Initial value of the battery at Reset.

The Initial value is determined two ways:

The battery voltage decreases depending on engine off time and Battery Temperature to a constant value at parked vehicle. If this phase is long enough the battery voltage can be used for first determination of the Initial Load. If the engine off time is to short then the value of the last Driving cycle is used for Initialization.

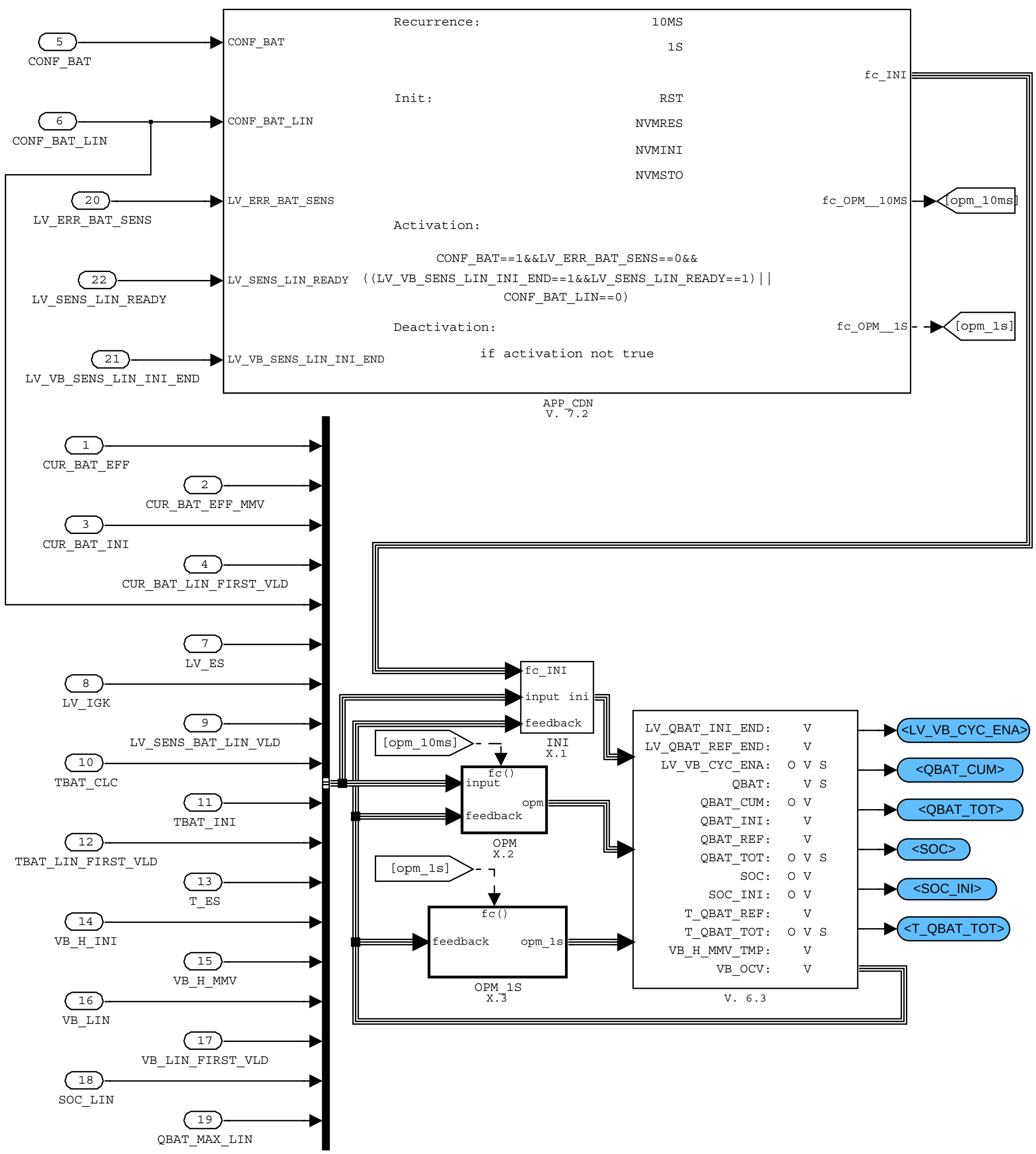
Additionally a new reference calculation is done if the Ignition is on and the alternator is not charging the battery. After the Polarization voltage disappeared a timer is decremented checking voltage and current are constant. After this timer is zero the reference value is calculated by a MAP depending on actual Battery voltage and Battery temperature. The value is corrected depending on the actual current.

The data CONF_BAT_LIN is used for battery sensor choice : 0=analog, 1=semi-smart, 2=smart .

<u>Application Conditions</u>

Initialization: RST, NVMRES, NVMINI, NVMSTO
Recurrence: 10MS, 1S
Activation: CONF_BAT==1 && LV_ERR_BAT_SENS==0 && ((LV_VB_SENS_LIN_INI_END==1 && LV_SENS_LIN_READY==1) || CONF_BAT_LIN==0)
Deactivation: if activation not true

Function description



SDA_SRS / SDA V 5.3.2 / 20−Feb−2008

Figure 14:

9.31.1 Initialization

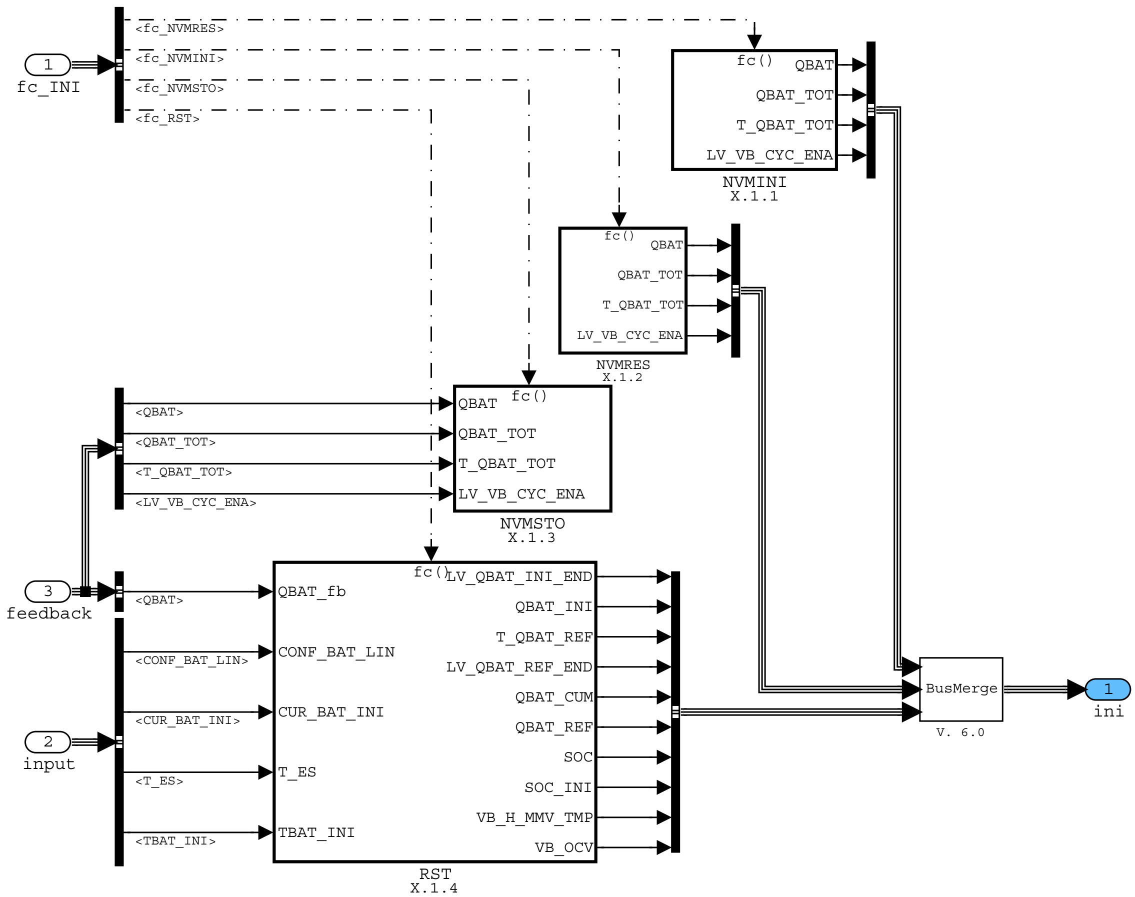


Figure 15:

9.31.1.1 Initialization for Non-volatile memory variable

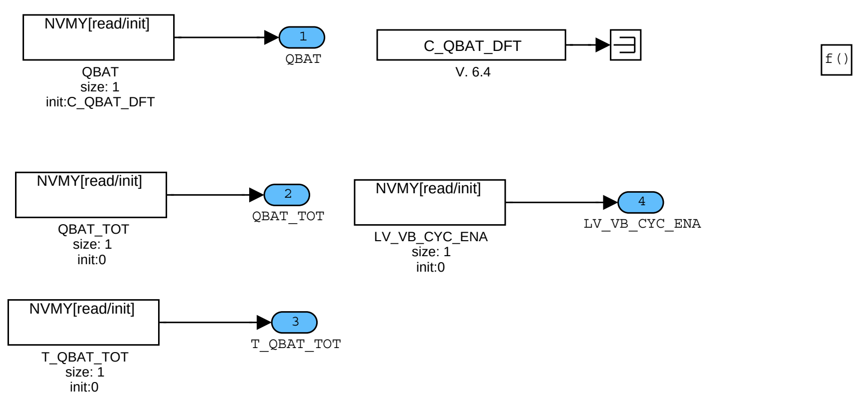


Figure 16:

9.31.1.2 Restore value for Non-volatile memory variable

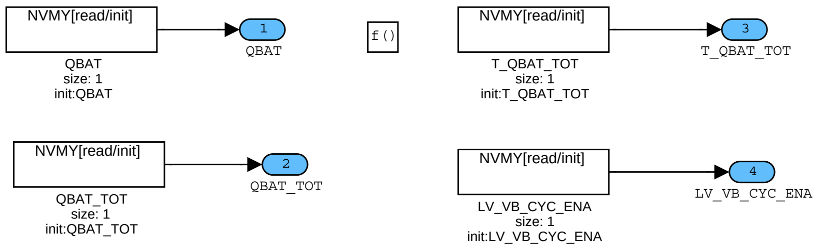


Figure 17:

9.31.1.3 Store value for Non-volatile memory variable

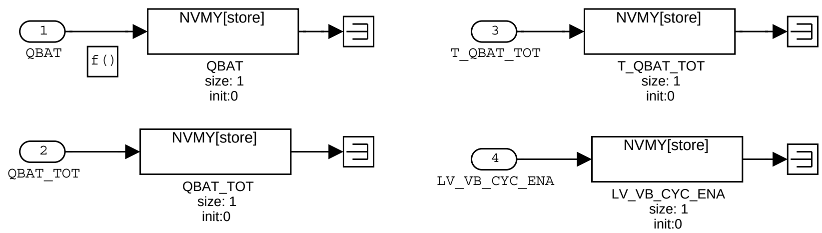


Figure 18:

9.31.1.4 Initialization at Reset

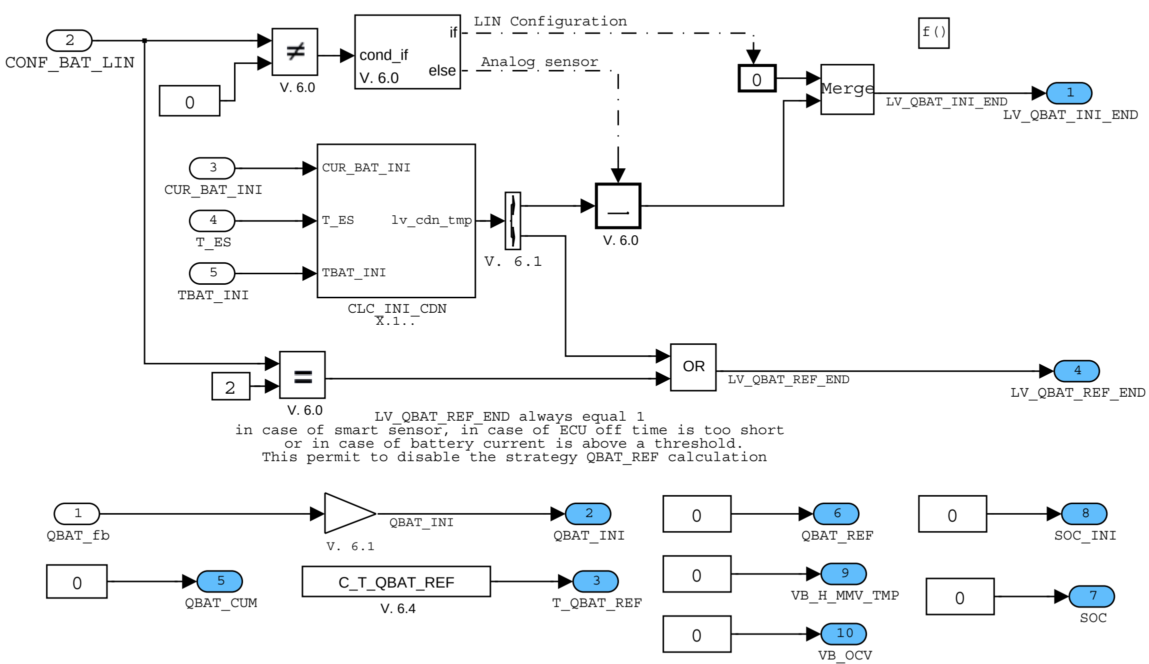


Figure 19:

9.31.1.4.1 Calculation if at initialization, battery condition fill in to calculate new SOC INI

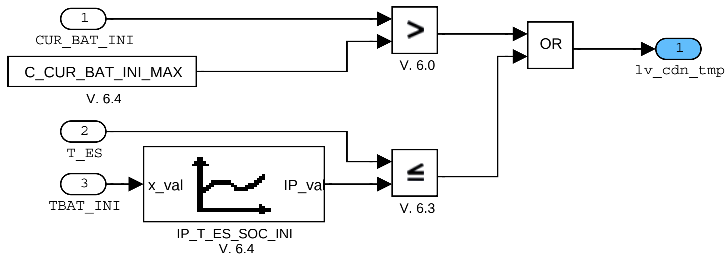


Figure 20:

9.31.2 Formula Section

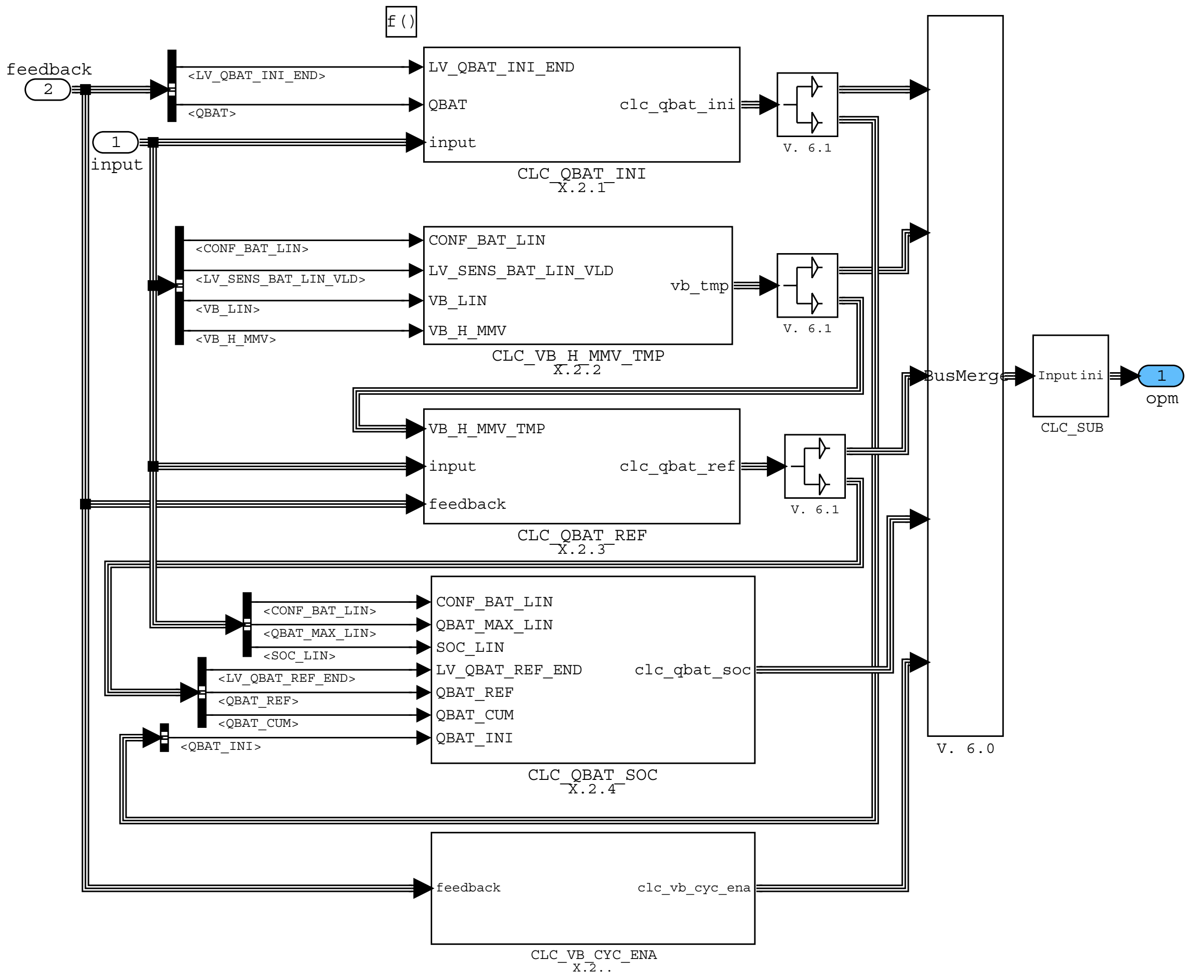


Figure 21:

9.31.2.1 Calculation of QBAT_INI

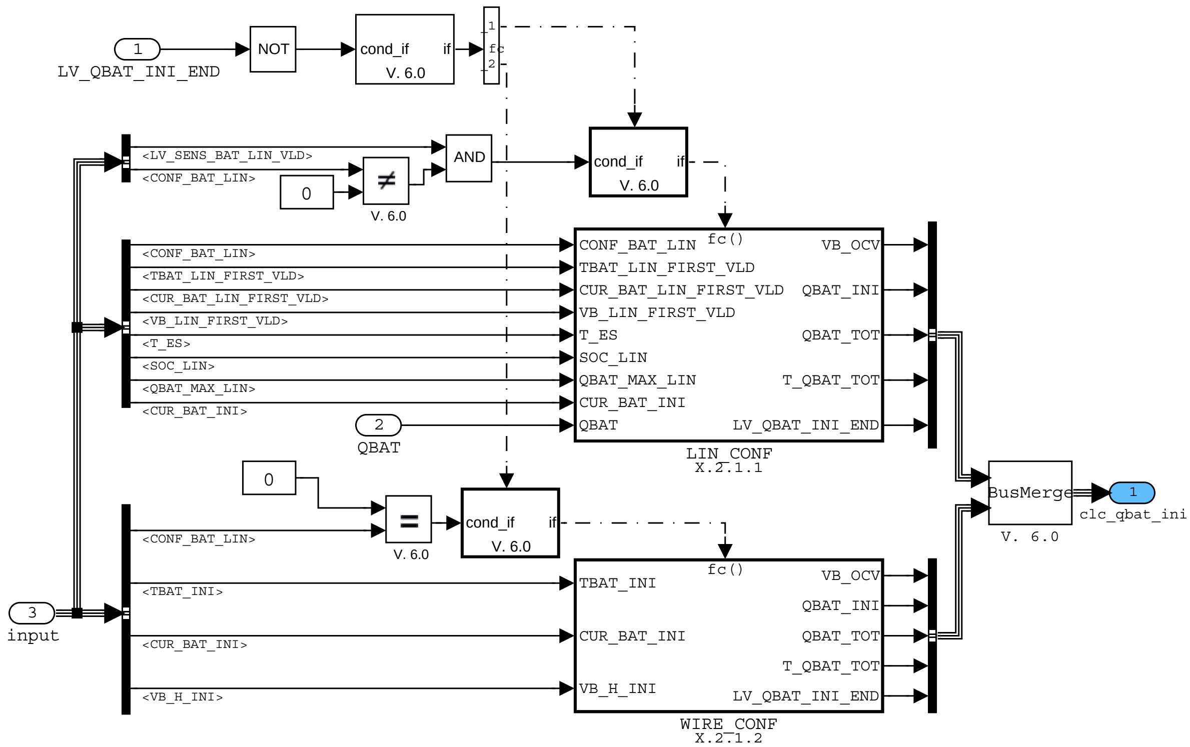


Figure 22:

9.31.2.1.1 Calculation of QBAT_INI in case of Lin Configuration

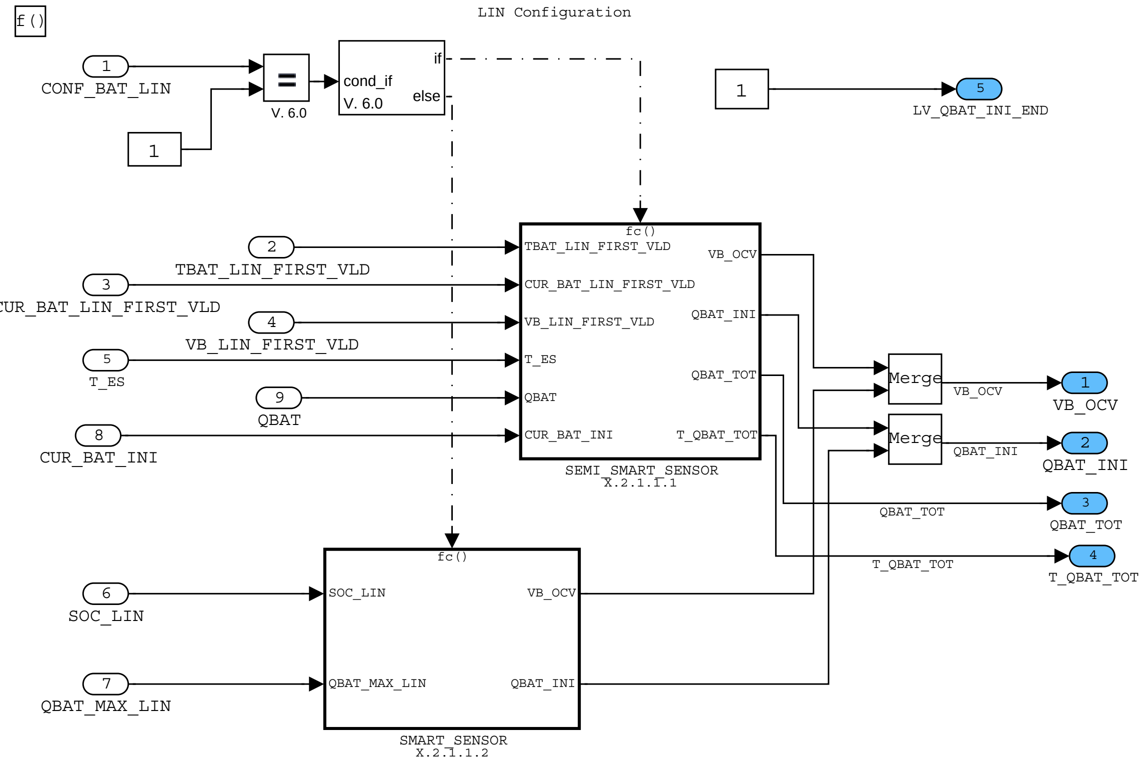


Figure 23:

9.31.2.1.1.1 Calculation of QBAT_INI in case of Lin Configuration with semi-smart sensor.

CONTENT

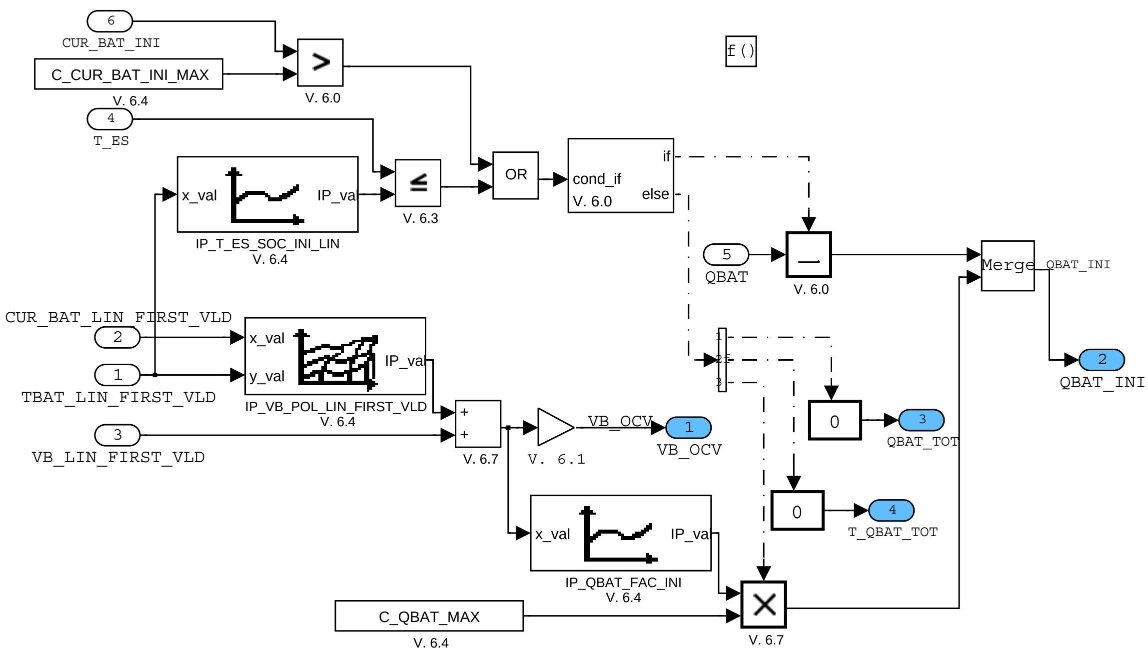


Figure 24:

9.31.2.1.1.2 Calculation of QBAT_INI in case of Lin Configuration with smart sensor.

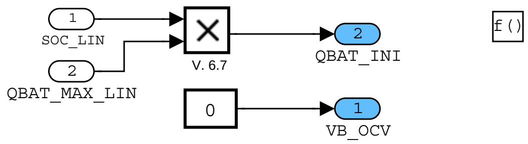


Figure 25:

9.31.2.1.2 Calculation of QBAT_INI in case of Wire Configuration

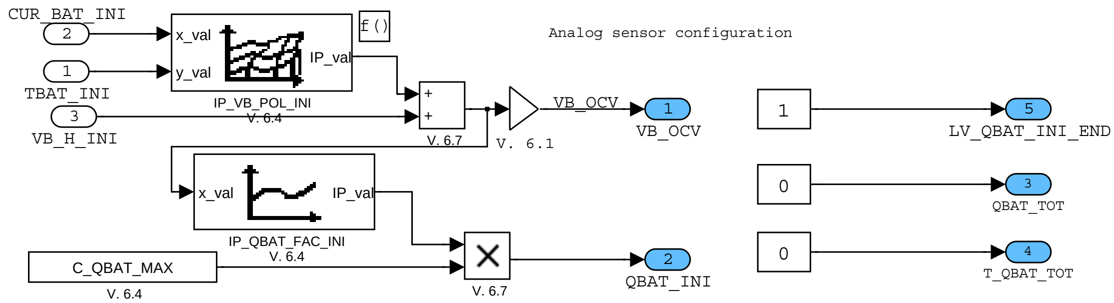


Figure 26:

9.31.2.2 Calculation of VB_H_MMV_TMP

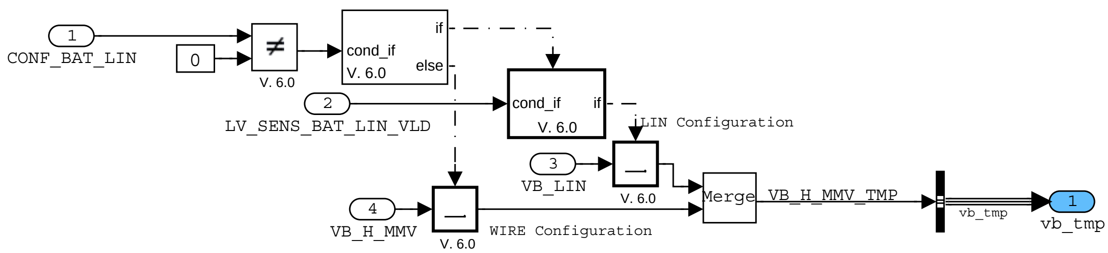


Figure 27:

9.31.2.3 Calculation of condition for QBAT_REF determination:

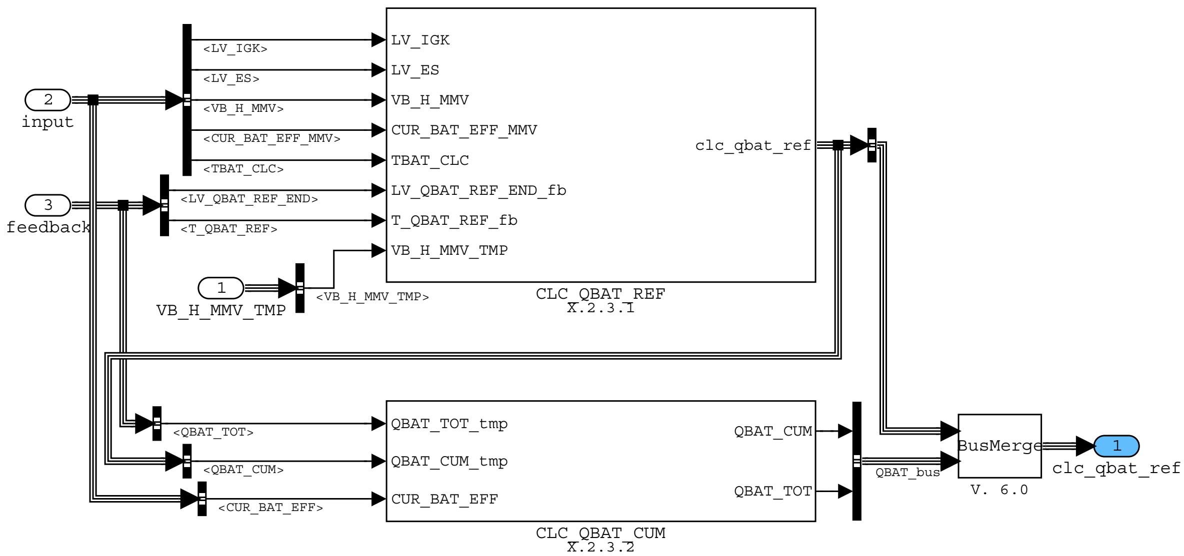


Figure 28:

9.31.2.3.1 Calculation of QBAT_REF

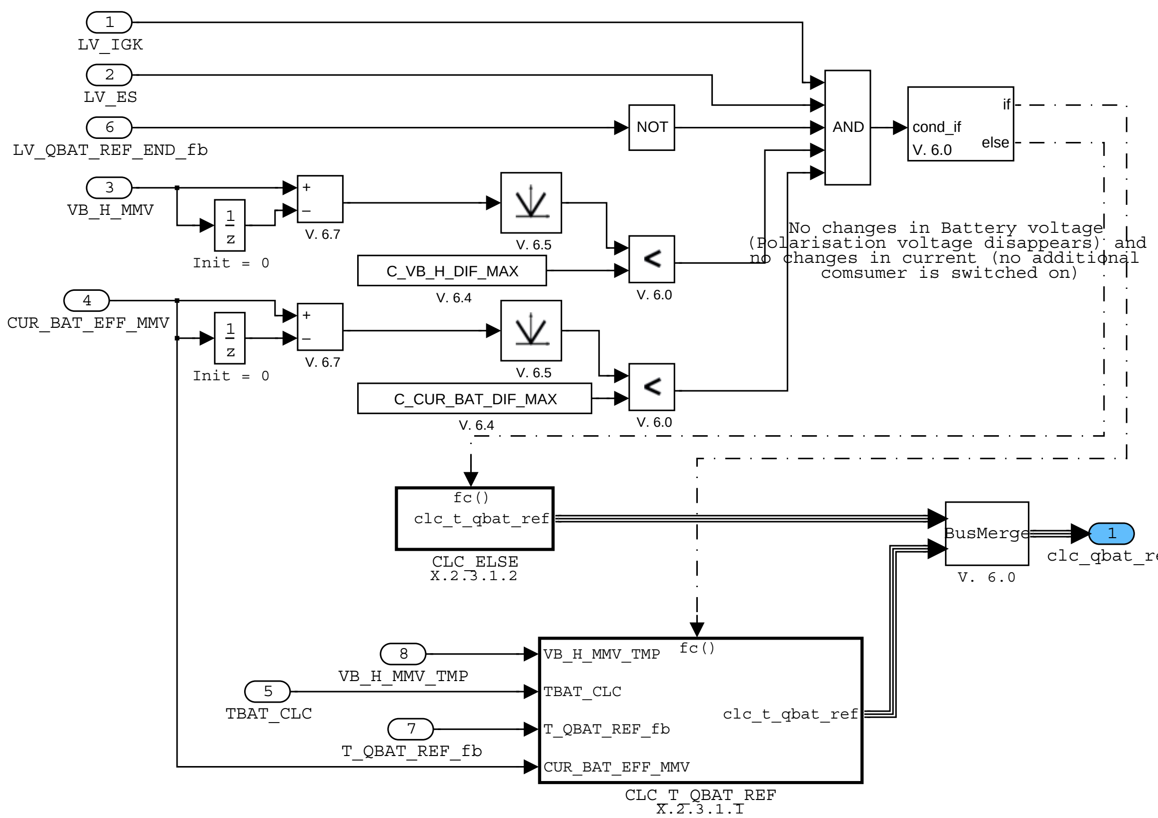


Figure 29:

9.31.2.3.1.1 Timer Calculation

The timer T_QBAT_REF is decremented

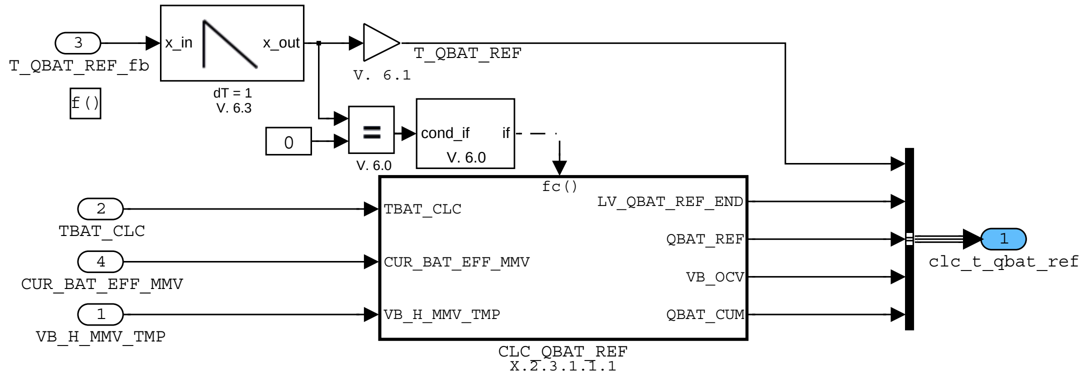


Figure 30:

9.31.2.3.1.1.1 Calculation QBAT_REF

T

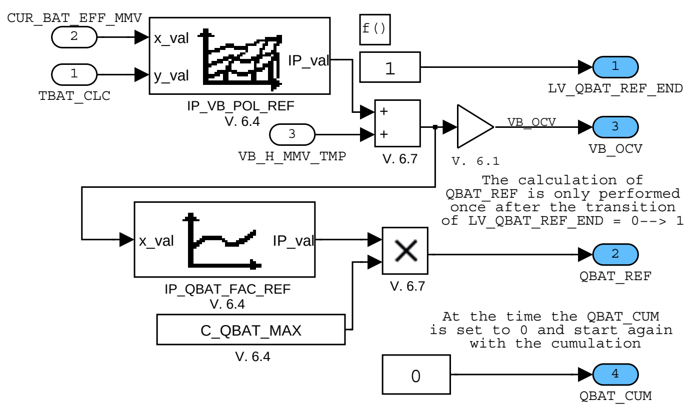


Figure 31:

9.31.2.3.1.2 Else Part

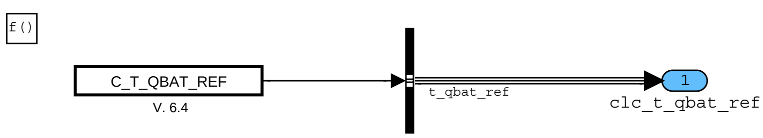


Figure 32:

9.31.2.3.2 Calculation of Accumulated Charge of Battery

Accumulated charge of the battery is calculated.

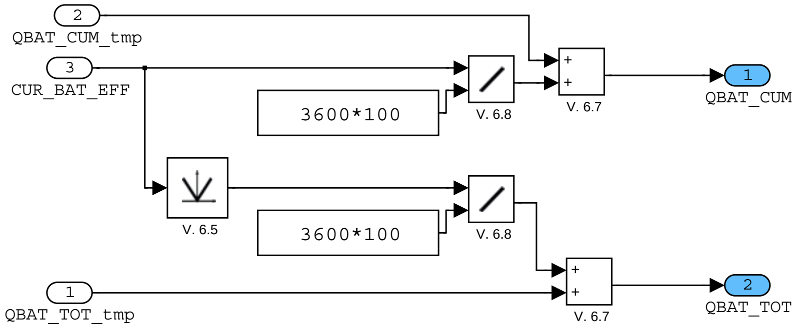


Figure 33:

9.31.2.4 Calculation of Battery Charge and state of Charge:

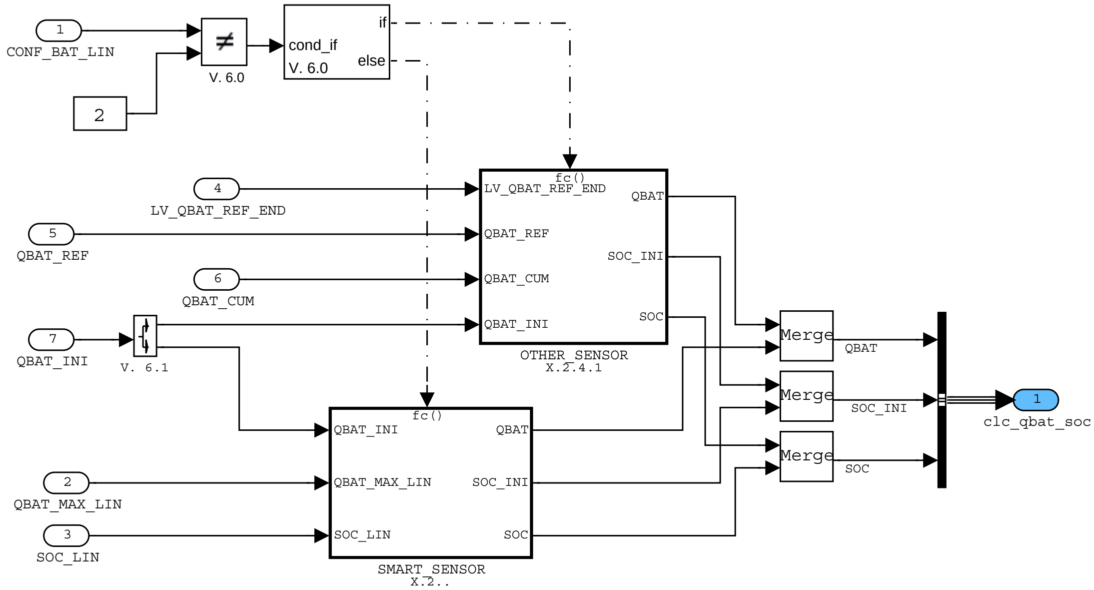


Figure 34:

9.31.2.4.1 Calculation of Battery Charge and State Of Charge for semi-smart sensor

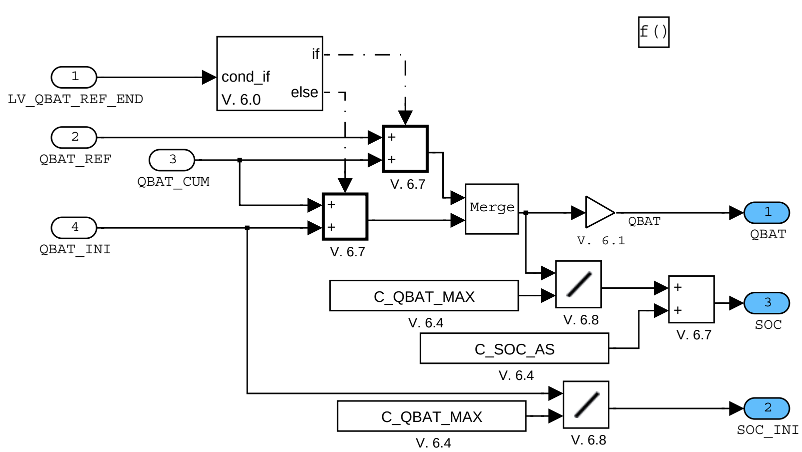


Figure 35:

9.31.2.4.2 Calculation of Battery Charge and State Of Charge for smart sensor

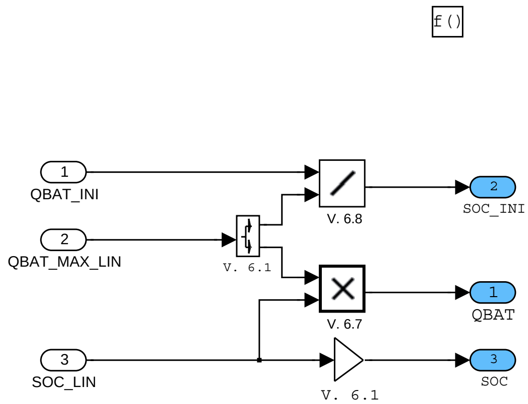


Figure 36:

9.31.2.5 Management of refreshment cycle

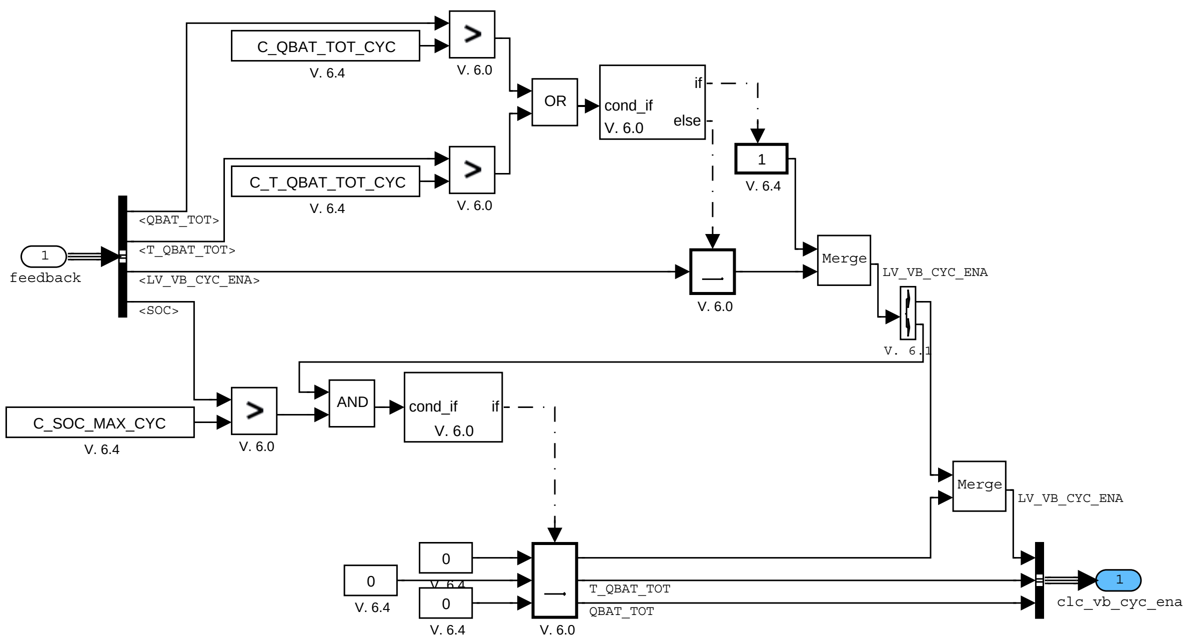


Figure 37:

9.31.3 Timer update each minute

Calculation of timer T_QBAT_TOT

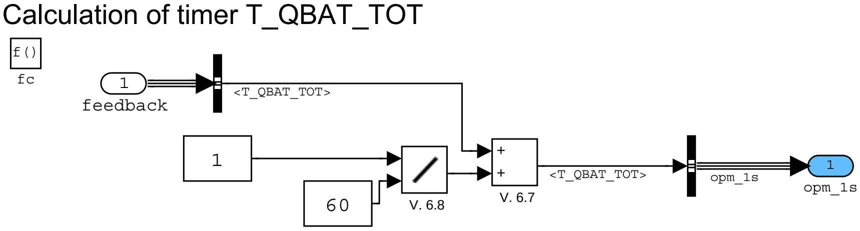


Figure 38:

9.32 Battery charge request

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
CTR_T_RAMP_V_ALTER	V	0... FFH	0... 255	1	[-]
Counter to add a time constant to reduce V_ALTER change					
LV_SOC_HIGH	V	0... 1H	0... 1	1	[-]
Flag inication SOC HIGH value phase is passed					
LV_SOC_LOW	V	0... 1H	0... 1	1	[-]
Flag inication SOC LOW value phase is passed					
QBAT_CUM_HIGH	V	80000000... 7FFFFFFFH	-120 ...120	55.8794e-9	[Ah]
Accumulated Charge during SOC HIGH phase					
QBAT_CUM_LOW	V	80000000... 7FFFFFFFH	-120 ...120	55.8794e-9	[Ah]
Discumulated Charge during SOC LOW phase					
STATE_CHA_TYP_REQ	V	0... FFH	0... 255	1	[-]
State to identify the charge type requested					
T_SOC_LOW	V	0... FFFFH	0... 6553.5	0.1	[s]
Timer to be decreased before SOC_LOW phase is passed					
V_ALT_TMP	V	0... FFH	10.6 ...16	0.0211765	[V]
Alternator voltage (temporary value before Manual modification)					
V_ALT_TMP_1	V	0... FFH	10.6 ...16	0.0211765	[V]
Alternator voltage (termporary value for closed loop calculation)					
V_ALT_TMP_2	V	0... FFH	10.6 ...16	0.0211765	[V]
Alternator voltage (temporary value for SOC HIGH LOW strategy)					
VB_SP_ALTER_REQ	O/V	0... FFH	10.6 ...16	0.0211765	[V]
Alternator setpoint voltage requested (V_ALT)					

Input Data:

TBAT_CLC	QBAT_CUM	SOC	SOC_INI
CONF_BAT	ENG_EFF_STATE	LV_ST_END	STATE_CHA_OPT
T_AST	TCO_ST	LV_VB_CYC_ENA	CUR_BAT_EFF_MMV
LV_ERR_BAT_SENS	LV_VB_SENS_LIN_INI_END	CONF_BAT_LIN	LV_SENS_LIN_READY

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_V_ALT_HIGH [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state HIGH					
C_CRLC_V_ALT_HIGH_1 [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state HIGH_1					
C_CRLC_V_ALT_HIGH_2 [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state HIGH_2					
C_CRLC_V_ALT_LOW_1 [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state LOW_1					
C_CRLC_V_ALT_LOW_2 [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state LOW_2					

C_CRLC_V_ALT_MAX	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state MAX					
C_CRLC_V_ALT_MIN	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state MIN					
C_CRLC_V_ALT_NOM [NC_NB_MAX_CHA_OPT]	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state MIN					
C_CRLC_V_ALT_TMP_1	1	0... FFH	0... 0.99	3.88235e-3	[-]
Correlation Constant for state TMP_1					
C_CUR_BAT_EFF_MMV_MAX	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Maximum value for CUR_BAT_EFF_MMV to start compensation					
C_CUR_BAT_EFF_MMV_MIN	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Minimum value for CUR_BAT_EFF_MMV to start compensation					
C_SOC_HIGH	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Target value for SOC in SOC_HIGH phase					
C_SOC_INI_MIN_HIGH_LOW	1	0... FFH	0... 100	0.3921569	[%]
Minimum battery state of charge requested at initialisation to start charge/discharge stage					
C_SOC_LOW	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Target value for SOC in SOC_LOW phase					
C_SOC_WIN_MAX	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
SOC Maximum value for control window					
C_SOC_WIN_MIN	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
SOC Minimum value for control window					
C_T_RAMP_V_ALTER	1	0... FFH	0... 255	1	[-]
Calibration to add a time constant to reduce V_ALTER change (typical value 2)					
C_V_ALT_DFT	1	0... FFH	10.6 ...16	0.0211765	[V]
Default value to start V_ALT control					
C_V_ALT_DIF	1	0... FFH	0... 5.4000075	0.0211765	[V]
V_ALT Steps to reach SOC window					
C_V_ALT_MAN	1	0... FFH	10.6 ...16	0.0211765	[V]
Manual value for Alternator voltage					
C_V_ALT_T_TMP_1_MAX	1	0... FFH	10.6 ...16	0.0211765	[V]
Maximum value to limit V_ALT_TMP_1 calculation					
C_V_ALT_T_TMP_1_MIN	1	0... FFH	10.6 ...16	0.0211765	[V]
Minimum value to limit V_ALT_TMP_1 calculation					
C_V_ALT_T_TMP_1_MIN_0	1	0... FFH	10.6 ...16	0.0211765	[V]
Minimum value to limit V_ALT_TMP_1 calculation in case no charge option					
C_V_ALT_TMP_INI	1	0... FFH	10.6 ...16	0.0211765	[V]
Default value to start the V_ALT_TMP_1 calculation					
C_V_ALTER_HYS_CHA_TYP	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Hysteresis to avoid oscillation between two states					
C_VB_SP_ALTER_REQ_SUB	1	0... FFH	10.6 ...16	0.0211765	[V]
Substitute value for VB_SP_ALTER_REQ					
IP_QBAT_SOC_HIGH	8	0... FFFFH	-120 ...120	3.66217e-3	[Ah]
LDPM_SOC_INI_IP_QBAT_SOC	8	0... FFH	0... 100	0.3921569	[%]
Accumulated Charge before SOC_HIGH is finished					
IP_QBAT_SOC_LOW	8	0... FFFFH	-120 ...120	3.66217e-3	[Ah]
LDPM_SOC_INI_IP_QBAT_SOC	8	0... FFH	0... 100	0.3921569	[%]
Accumulated Charge before SOC_LOW is finished					
IP_T_AST_SOC_HIGH	8*8	0... FFFFH	0... 6553.5	0.1	[s]
LDPM_SOC_INI_IP_QBAT_SOC	8	0... FFH	0... 100	0.3921569	[%]
LDPM_TCO_ST_IP_TAST_SOC_HIGH	8	0... FEH	-48... 142.5	0.75	[°C]
T_AST before SOC_HIGH is finished					

IP_T_SOC_LOW	8*8	0... FFFFH	0... 6553.5	0.1	[s]
LDPM_SOC_INI_IP_QBAT_SOC	8	0... FFH	0... 100	0.3921569	[%]
LDPM_TCO_ST_IP_TAST_SOC_HIGH	8	0... FEH	-48... 142.5	0.75	[°C]
Time after start of discharge before SOC_LOW is finished					
IP_V_ALT_CYC_ENA	8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
Alternator setpoint voltage when battery refresh cycle activated					
IP_V_ALT_HIGH	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT_2	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage for state HIGH					
IP_V_ALT_HIGH_1	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage for state HIGH_1					
IP_V_ALT_HIGH_2	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDP_SOC_IP_V_ALT_HIGH	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage for state HIGH_2					
IP_V_ALT_LOW_1	8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
Alternator Voltage for state LOW_1					
IP_V_ALT_LOW_1_0	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage for state LOW_1 in case of no charge option					
IP_V_ALT_LOW_2	8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
Alternator Voltage for state LOW_2					
IP_V_ALT_LOW_2_0	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT_2	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage for state LOW_2 in case of no charge option					
IP_V_ALT_MAX	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Maximum Alternator Voltage to prevent Gazing Effect					
IP_V_ALT_MIN	8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
Alternator Voltage for state MIN					
IP_V_ALT_NORM_2_0	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT_2	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Alternator Voltage during discharge mode when engine efficiency state is normal					
IP_V_ALTER_NORM	8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
Alternator voltage for state NOM					
IP_V_ALTER_NORM_0	8*8	0... FFH	10.6 ...16	0.0211765	[V]
LDPM_TBAT_CLC_IP_V_ALT	8	0... FEH	-48... 142.5	0.75	[°C]
LDPM_SOC_IP_V_ALT	8	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]

Alternator voltage for state NOM in case of no charge option					
LC_ENA_SOC_CTL	1	0... 1H	0... 1	1	[-]
Switch activating the SOC Normal strategy					
LC_SOC_HIGH_LOW	1	0... 1H	0... 1	1	[-]
Switch activating the SOC HIGH/LOW strategy					
LC_V_ALT_MAN	1	0... 1H	0... 1	1	[-]
Switch activating manual Alternator voltage					
LC_VB_ALTER_REQ_SUB	1	0... 1H	0... 1	1	[-]
Usage of substitute calibration value (typical value 0 : not active)					

Configuration Data:

Name	Mode	Hex.Limits	Phys.Limits	Resol.	Unit
NC_NB_MAX_CHA_OPT	1	1... FFH	1... 255	1	[-]
Maximum number of charge option calculated (typical value 4)					

General Information

Target of Battery Management is to control the Battery State of Charge in a window, where charging and discharging is possible. This is done by control of the Alternator voltage. Increasing the Alternator voltage increases the SOC (dependent on consumers) and vice versa.
This allows to charge the battery where the engine efficiency is high and to decrease the Alternator Voltage Setpoint in state with low efficiency to decrease torque consumed by Alternator.
The highest priority after cranking is to bring the battery as soon as possible to the desired state of charge to enable restart without problem. Therefore if the SOC is below a threshold the charging has to be done independent of engine efficiency.
Afterwards the SOC is controlled in a window between SOC_WIN_MIN and SOC_WIN_MAX. Target is to keep the Alternator voltage constant till the thresholds are passed. Afterwards the Alternator voltage is de/increased stepwise, till the direction of SOC is changed again. Then the new Alternator voltage is kept constant again.
Additionally it is possible to increase SOC after start depending on different values (T_AST, SOC_INI to high target C_SOC_HIGH and to decrease to a low Target C_SOC_LOW afterwards, before SOC is controlled inside window. This logic can be switched on via LC_SOC_HIGH_LOW.

Application Conditions

Initialization: RST
Recurrence: 100MS
Activation: CONF_BAT==1&&LV_ERR_BAT_SENS==0&&((LV_VB_SENS_LIN_INI_END==1 && LV_SENS_LIN_READY==1) || CONF_BAT_LIN==0)
Deactivation: if activation not true

Function description

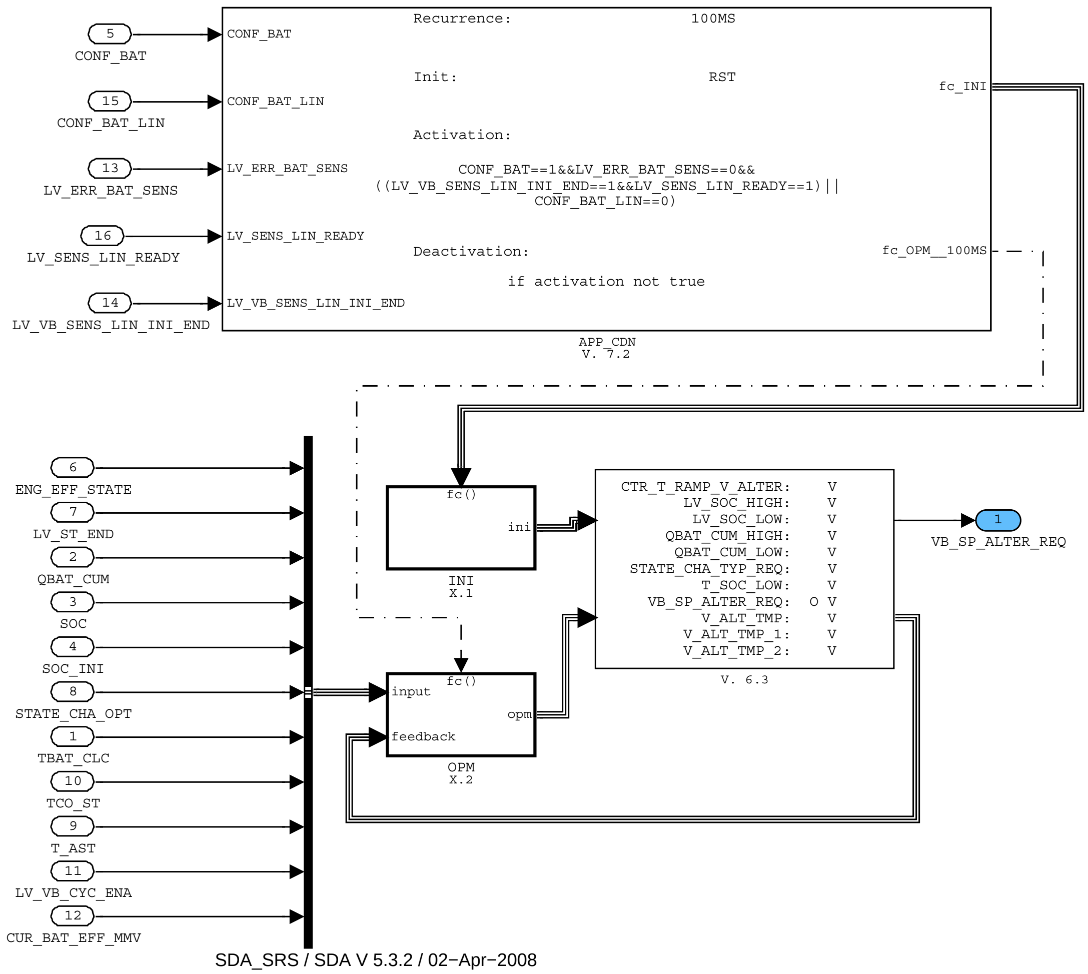


Figure 39:

9.32.1 Initialization

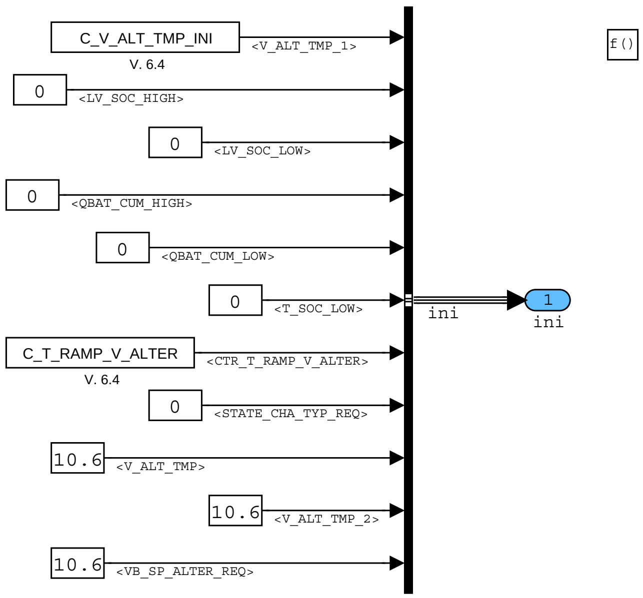


Figure 40:

9.32.2 Formula Section

Split between normal battery charge request and battery refresh cycle.

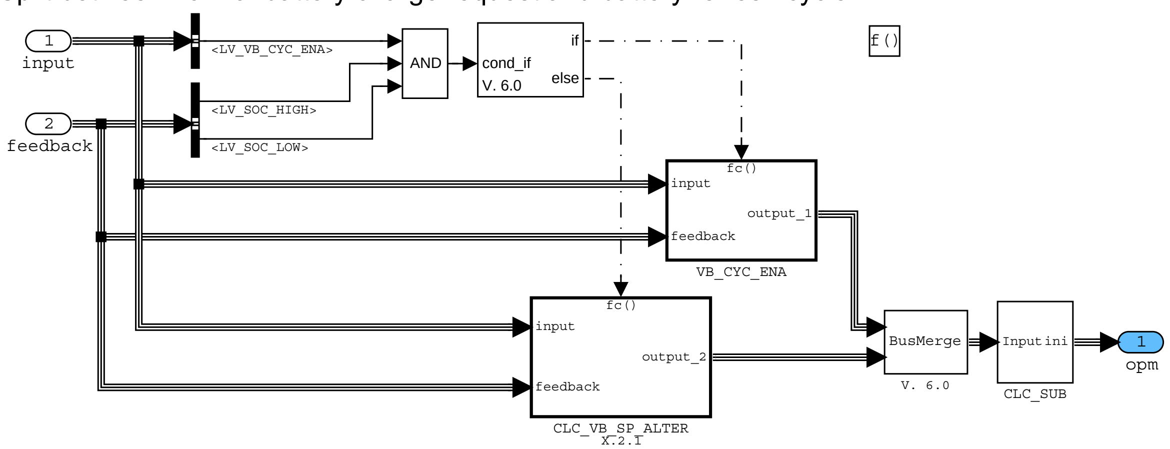


Figure 41:

9.32.2.1 Calculation of battery charge request

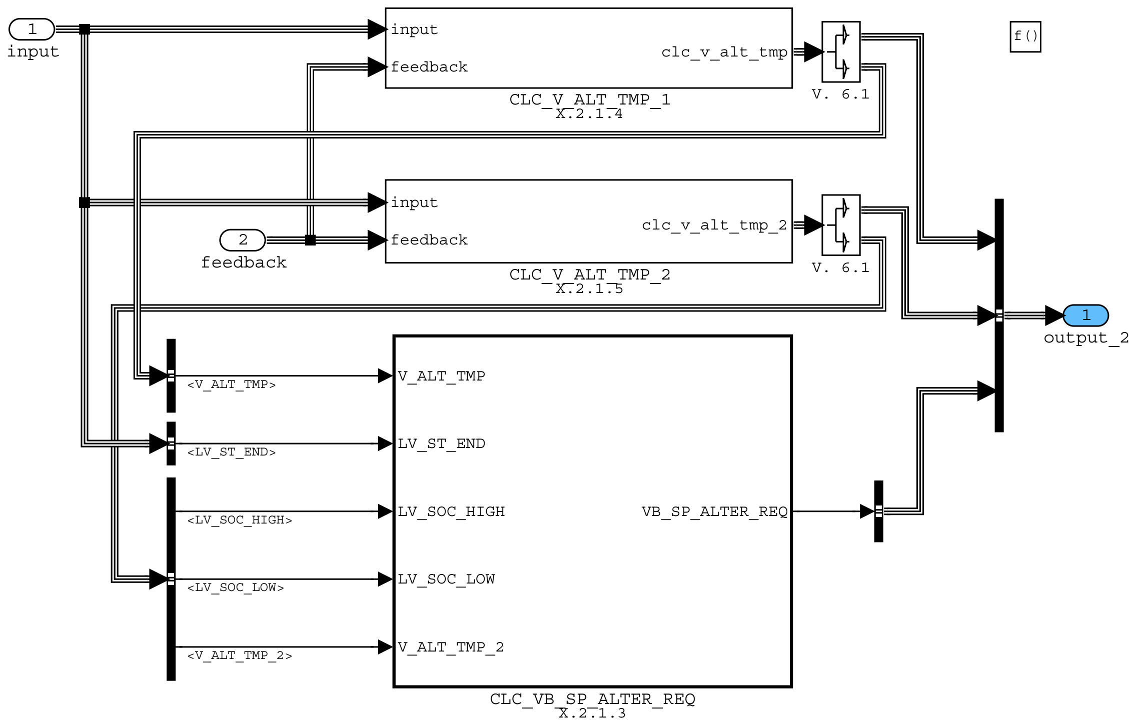


Figure 42:

9.32.2.1.1 Calculation of Alternator voltage VB_SP_ALTER_REQ:

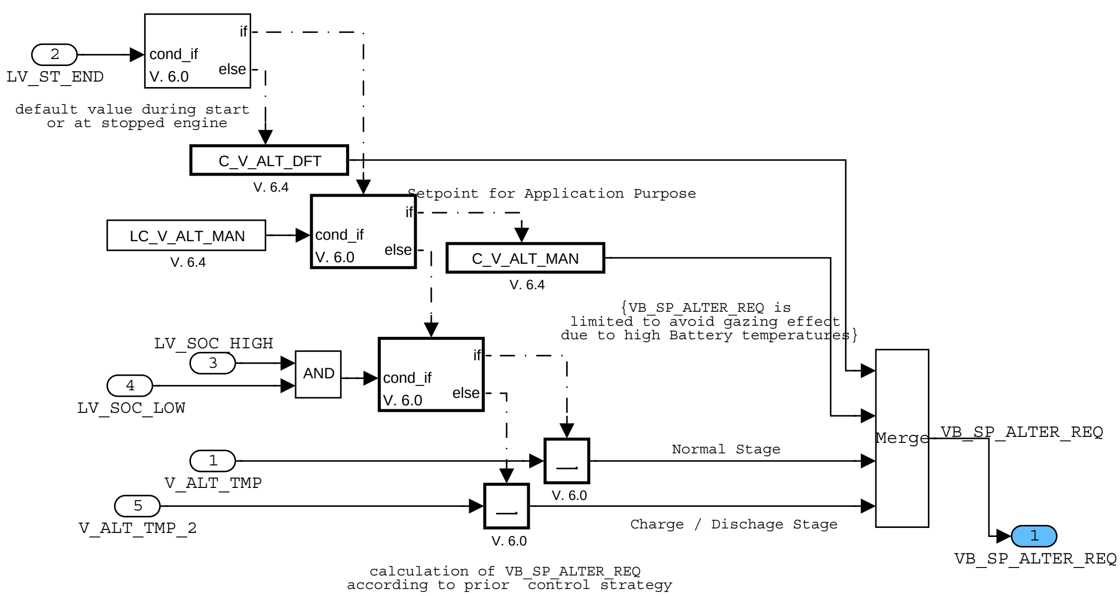


Figure 43:

9.32.2.1.2 Calculation of temporary alternator voltage V_ALT_TMP:

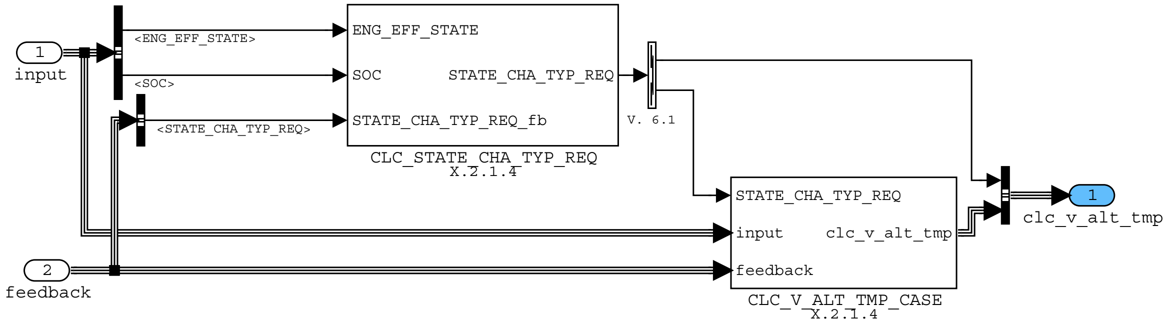


Figure 44:

9.32.2.1.2.1 Calculation of STATE_CHA_TYP_REQ

ENG_EFF_STATE	SOC with Hysteresis C_V_ALTER_HYS_CHA_TYP	Charge type requested
High	SOC >= C_SOC_WIN_MIN	STATE_CHA_TYP_REQ = 1
Low Medium or High	SOC < C_SOC_WIN_MIN	STATE_CHA_TYP_REQ = 2
Low	SOC >= C_SOC_WIN_MIN	STATE_CHA_TYP_REQ = 3
Medium	SOC >= C_SOC_WIN_MIN SOC < C_SOC_WIN_MAX	STATE_CHA_TYP_REQ = 4
Medium	SOC >= C_SOC_WIN_MAX	STATE_CHA_TYP_REQ = 5

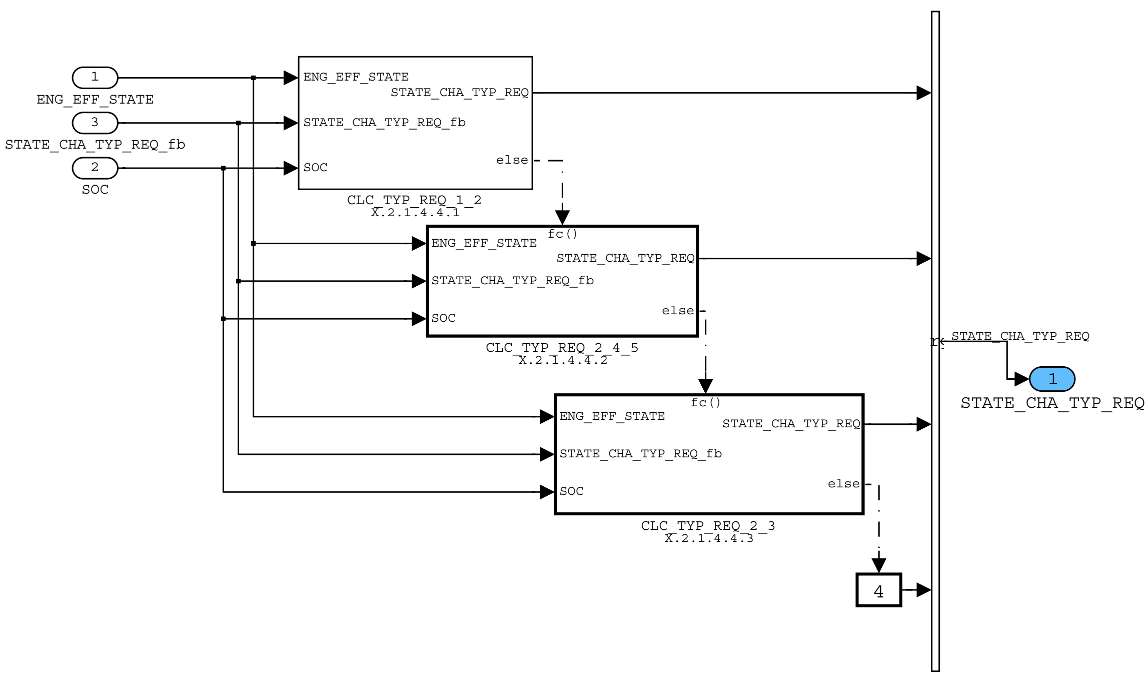


Figure 45:

9.32.2.1.2.1.1 Condition evaluation for STATE_CHA_TYP_REQ = 1

Case of engine efficiency state is high

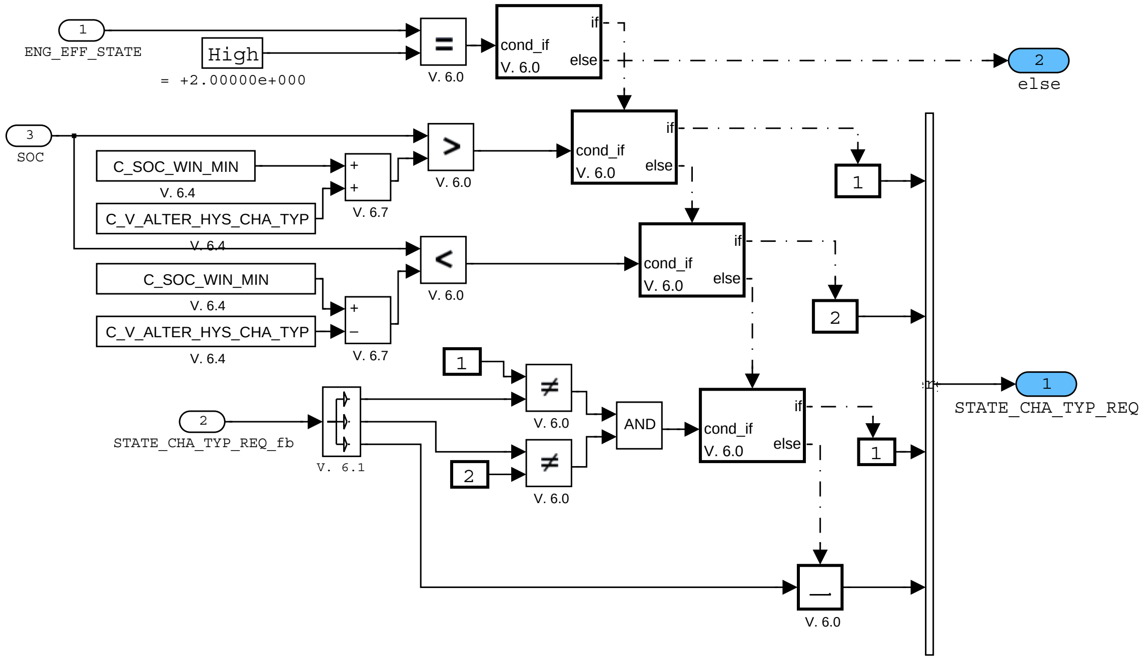


Figure 46:

9.32.2.1.2.1.2 Condition evaluation for STATE_CHA_TYP_REQ = 2, 4 and 5

Case of engine efficiency state is medium

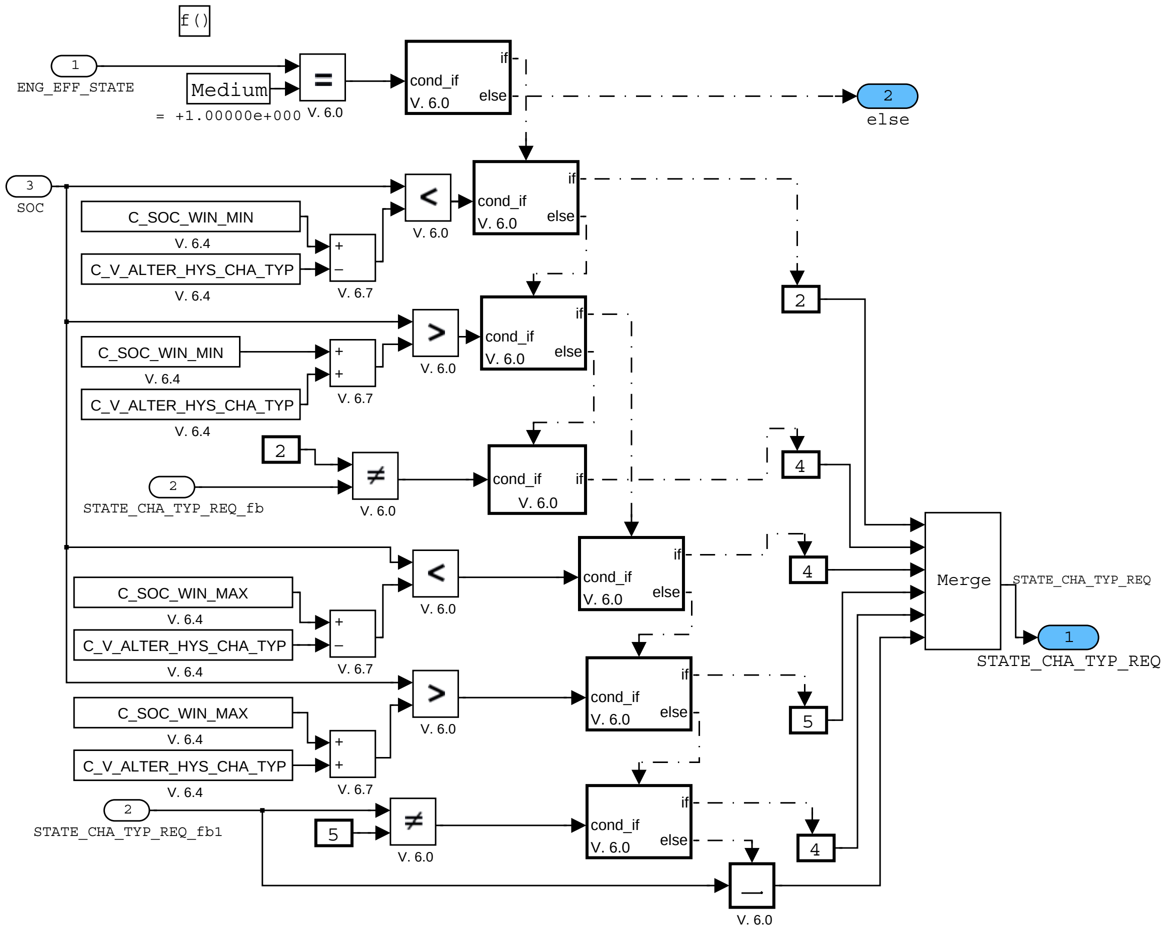


Figure 47:

9.32.2.1.2.1.3 Condition evaluation for STATE_CHA_TYP_REQ = 2 and 3

Case of engine efficiency state is low

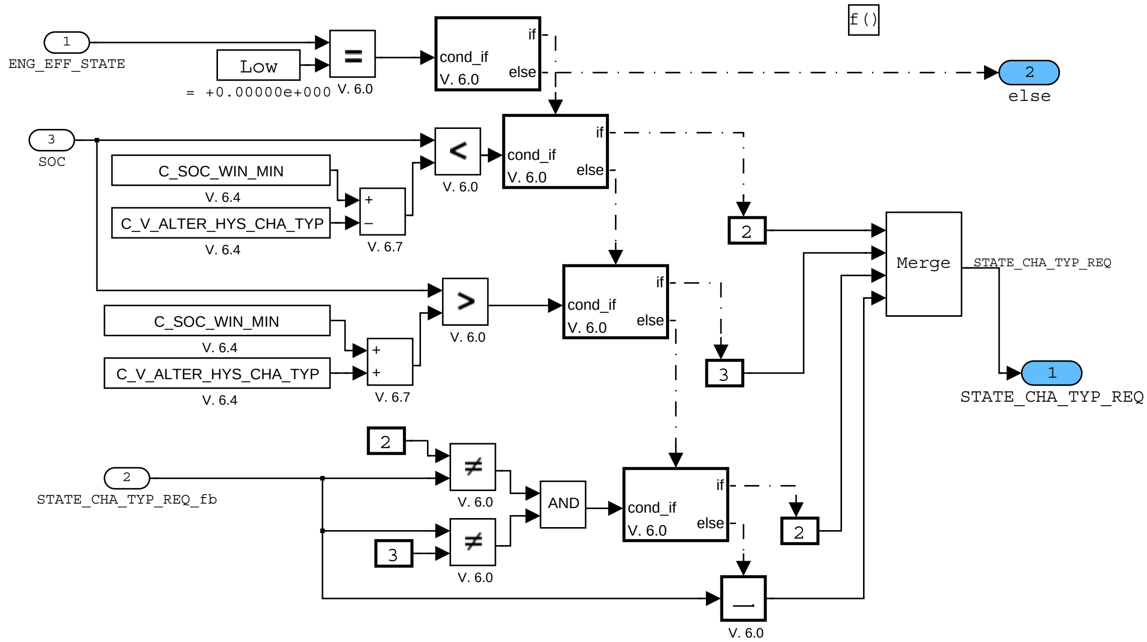


Figure 48:

9.32.2.1.2.2 Calculation of V_ALT_TMP

9.32.2.1.2.2.1 Case STATE_CHA_TYP_REQ = 1

{ V_ALT_TMP is ramped to new target value IP_V_ALT_HIGH_1 with Correlation constant }

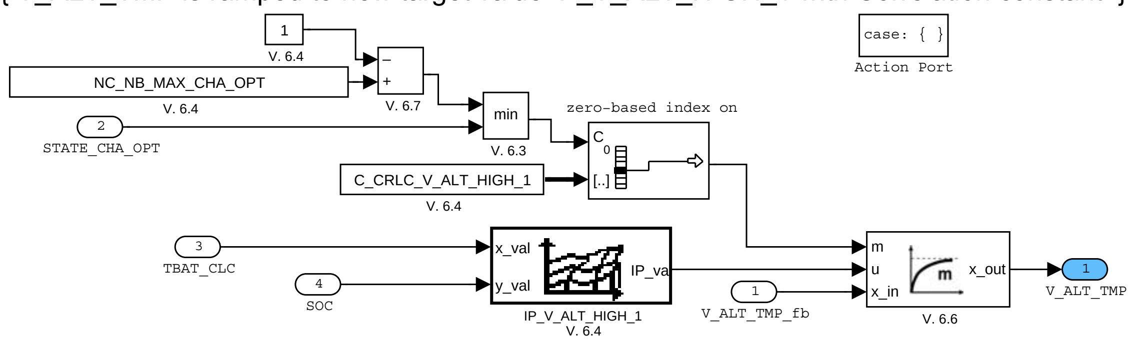


Figure 49:

9.32.2.1.2.2.2 Case STATE_CHA_TYP_REQ = 2

{ V_ALT_TMP is ramped to new target value IP_V_ALT_MIN (with Correlation constant }

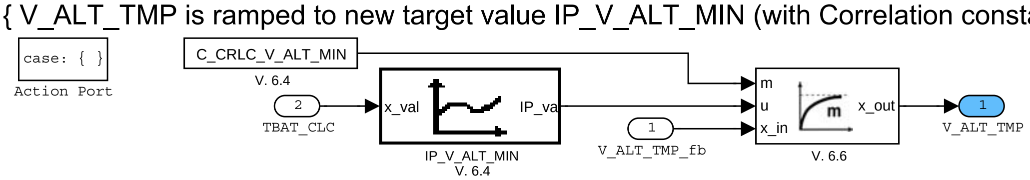


Figure 50:

9.32.2.1.2.2.3 Case STATE_CHA_TYP_REQ = 3

{ V_ALT_TMP is ramped to new target value IP_V_ALT_LOW_1with Correlation constant }

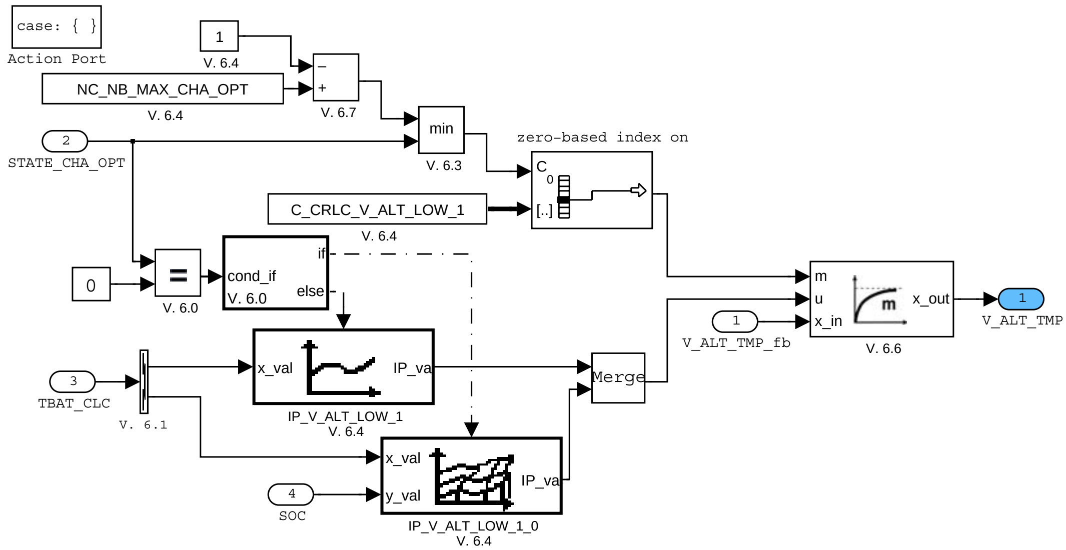


Figure 51:

9.32.2.1.2.2.4 Case STATE_CHA_TYP_REQ = 4

{ Alternator voltage Set point is controlled depending on SOC in Engine efficiency state "Normal"
SOC value is inside control window
Alternator voltage set point jumps last calculated V_ALT_TMP_1 value}

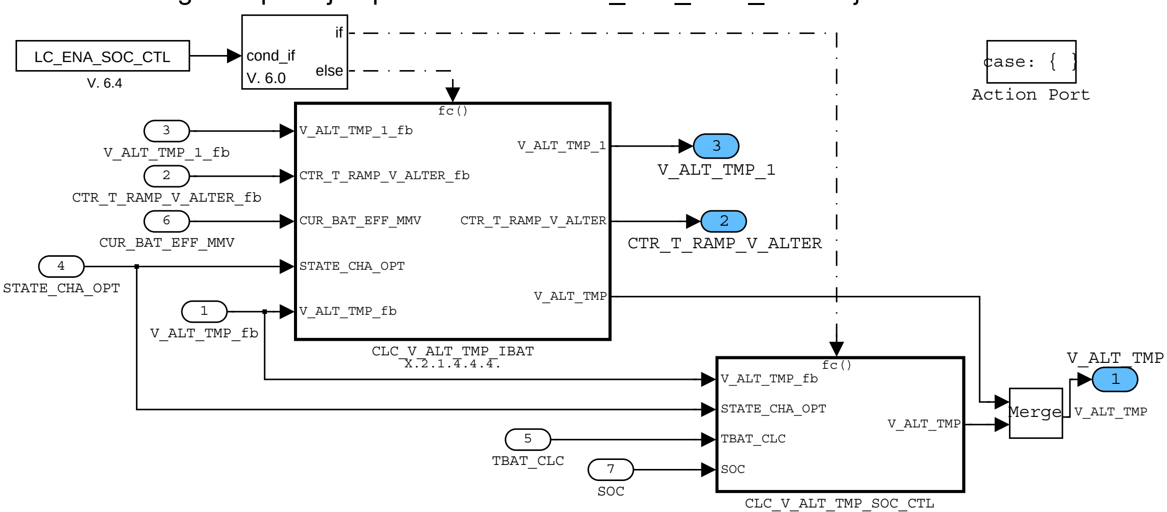


Figure 52:

9.32.2.1.2.2.4.1 Battery charge request adjusted according average current

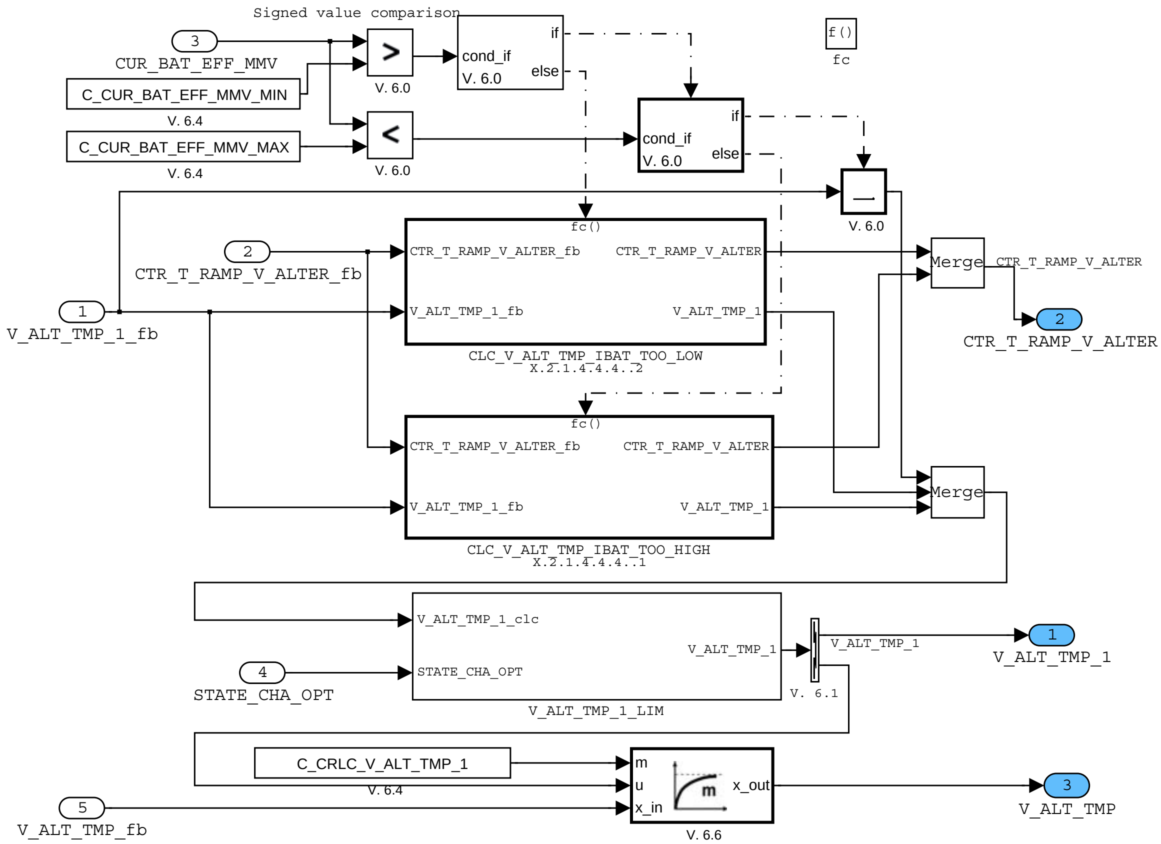


Figure 53:

9.32.2.1.2.2.4.1.1 Case of average current to battery too high

Decrease of alternator voltage step by step

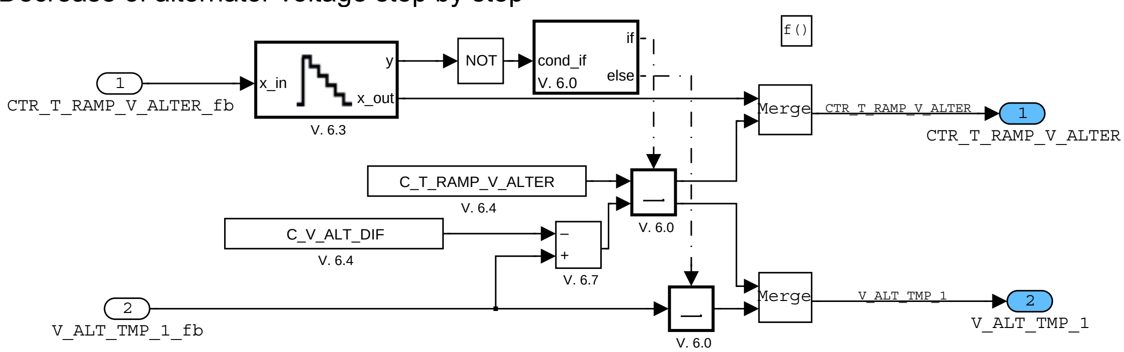


Figure 54:

9.32.2.1.2.2.4.1.2 Case of average current to battery too low

Increase of alternator voltage step by step

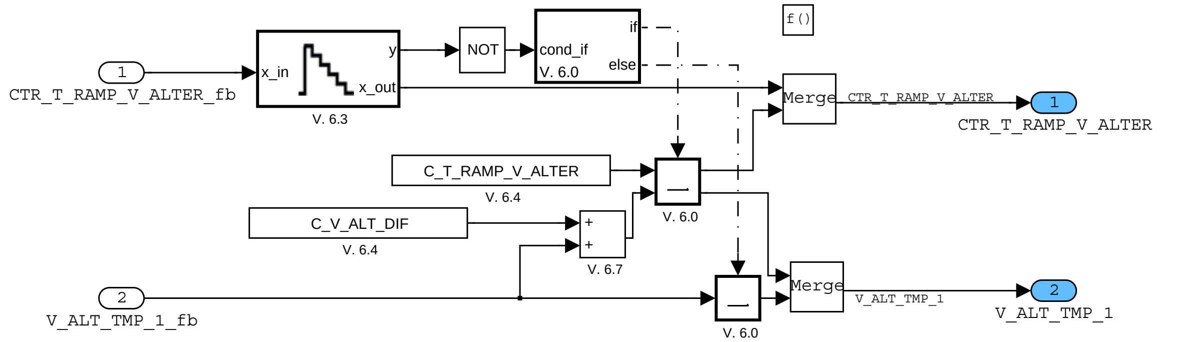


Figure 55:

9.32.2.1.2.2.4.1.3 Limitation of V_ALT_TMP_1

The limitation is based on state charge option

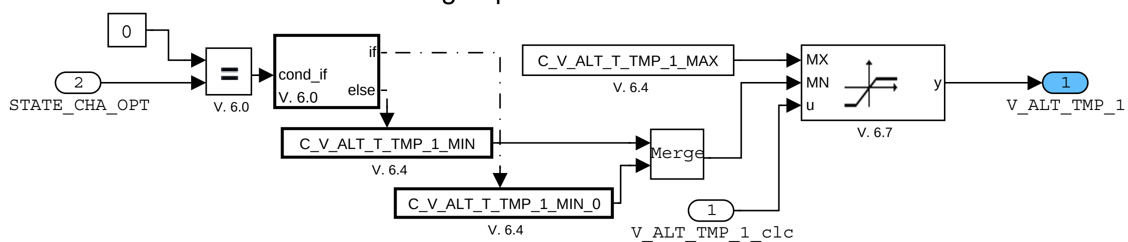


Figure 56:

9.32.2.1.2.2.4.2 Battery charge request is computed according SOC

Two maps are used according state charge option

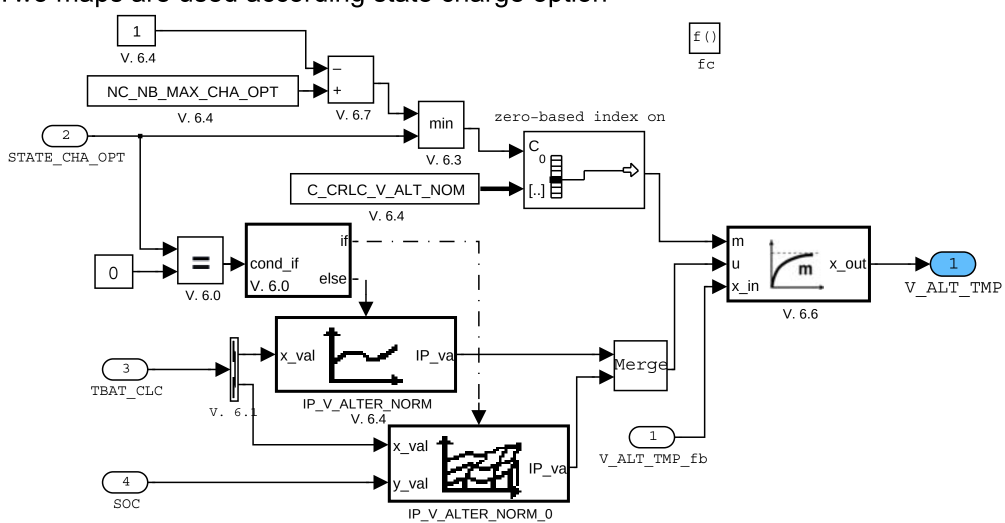


Figure 57:

9.32.2.1.2.2.5 Case STATE_CHA_TYP_REQ = 5

{ Alternator voltage Set point is controlled depending on SOC in Engine efficiency state "Normal" SOC value is above the control window}

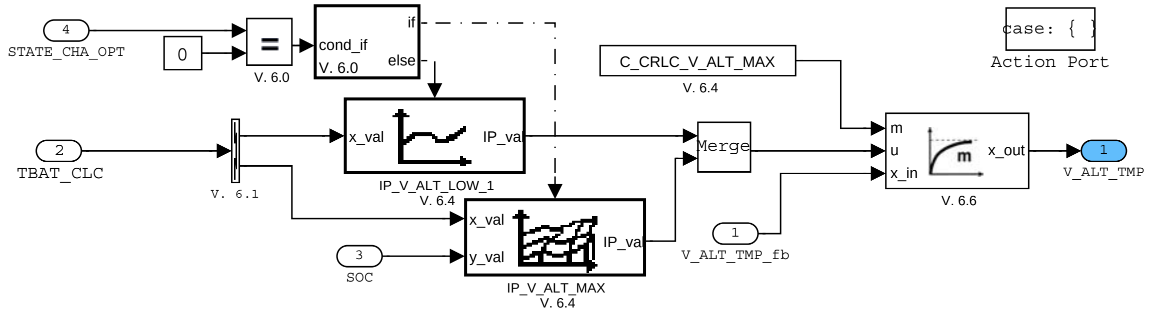


Figure 58:

9.32.2.1.3 Calculation of temporary Alternator Voltage V_ALT_TMP_2 with SOC_HIGH and SOC_LOW strategy before SOC control

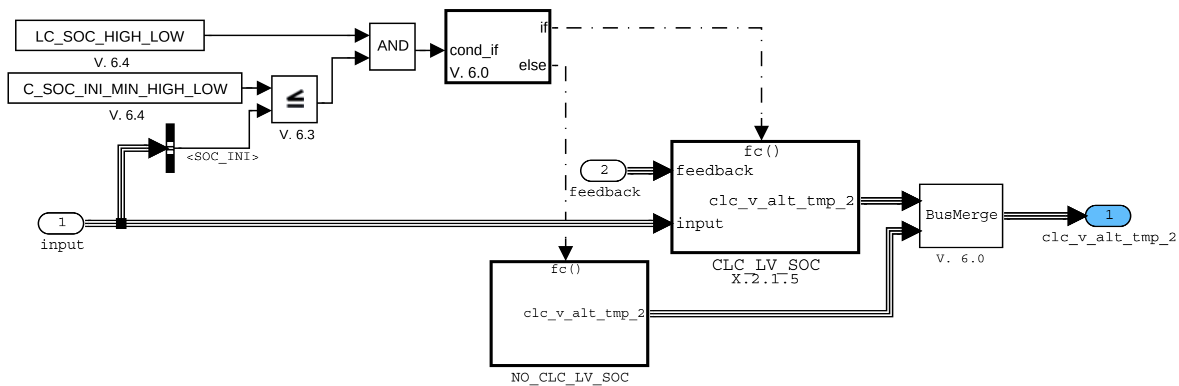


Figure 59:

9.32.2.1.3.1 Management of charge and discharge stage

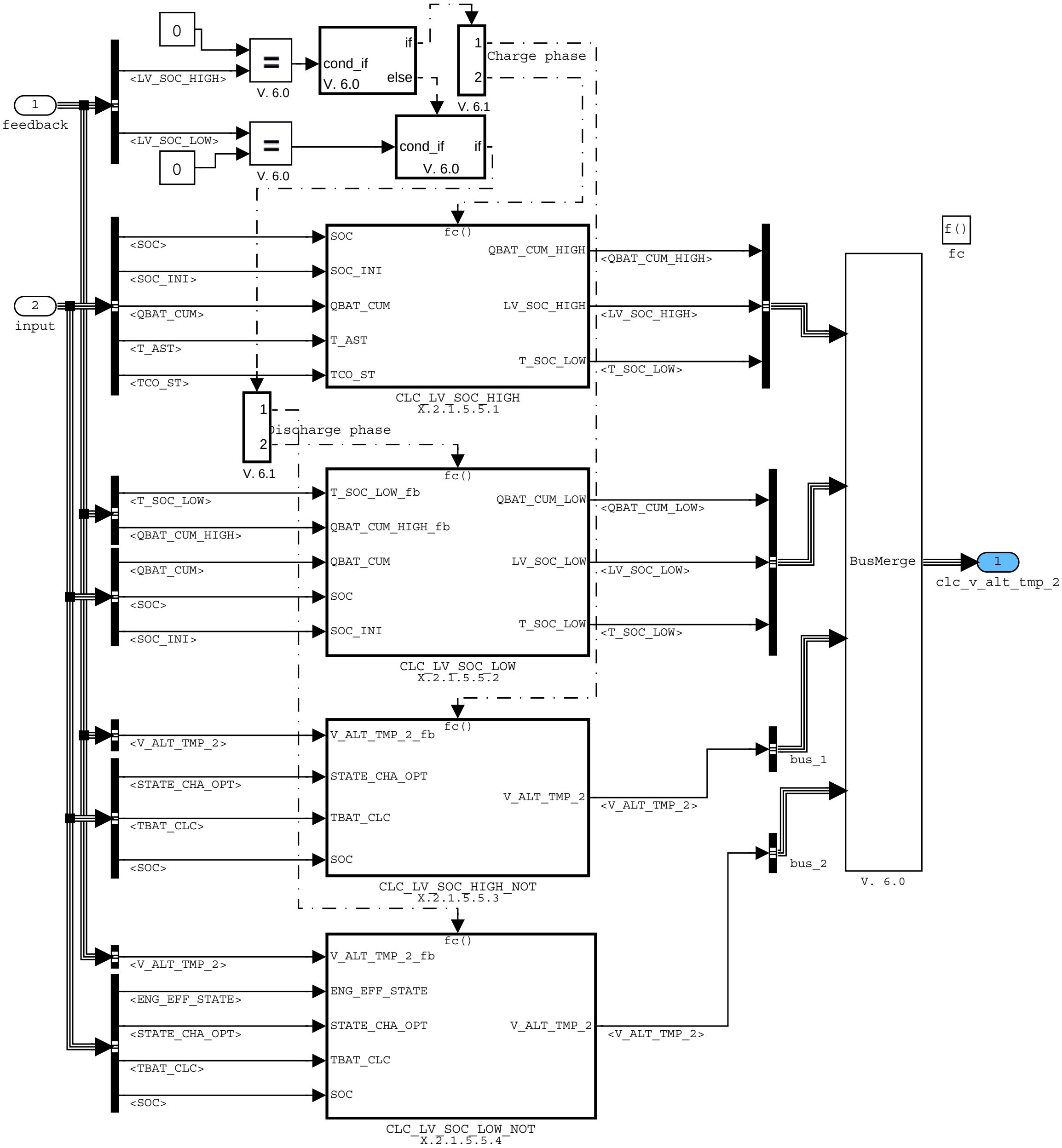


Figure 60:

9.32.2.1.3.1.1 Condition to reach LV_SOC_HIGH

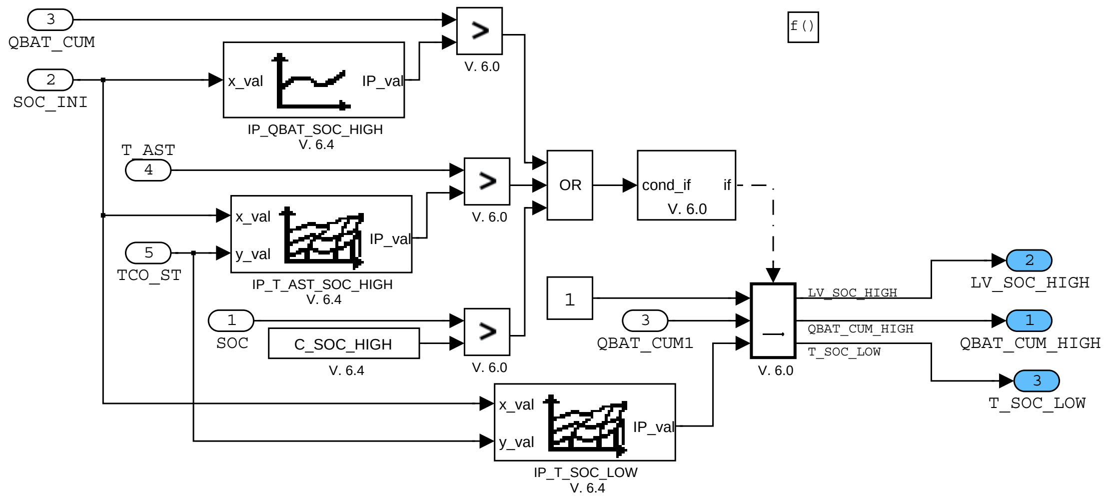


Figure 61:

9.32.2.1.3.1.2 Calculation of LV_SOC_LOW

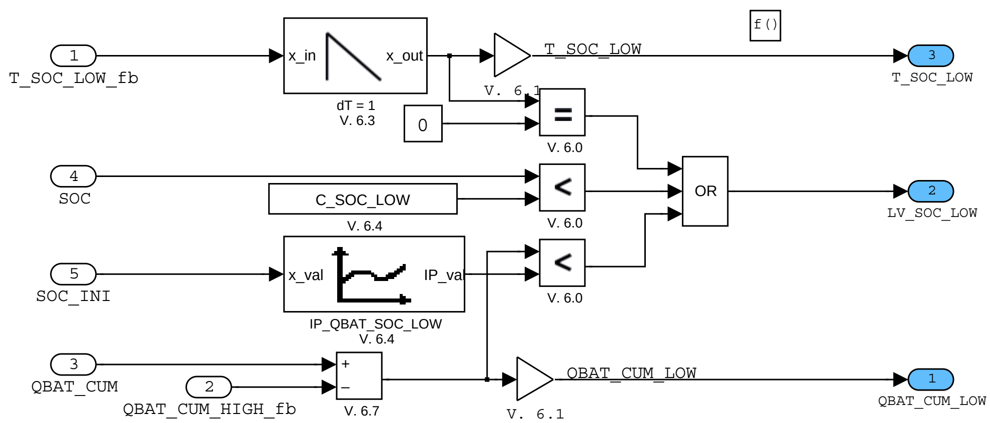


Figure 62:

9.32.2.1.3.1.3 Management of charge mode

LV_SOC_HIGH not yet reached

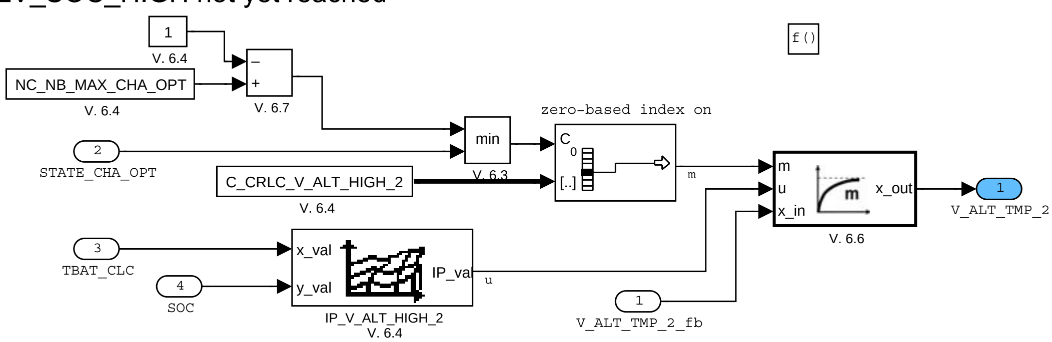


Figure 63:

9.32.2.1.3.1.4 Management of discharge mode

LV_SOC_LOW not yet reached

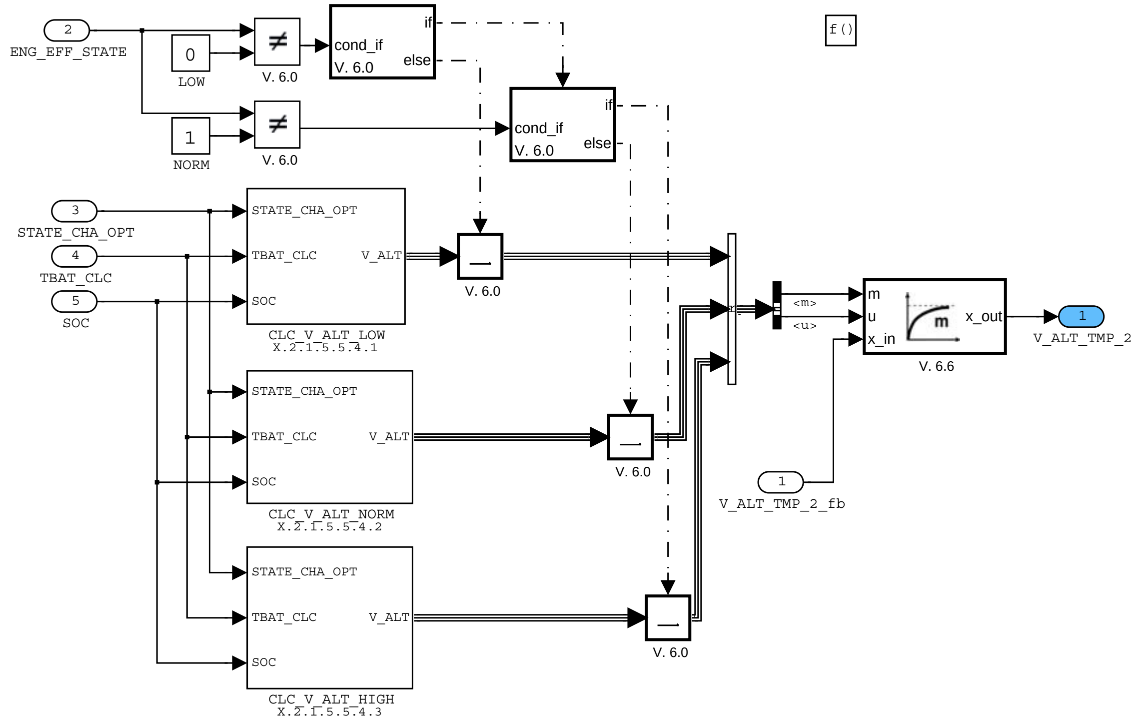


Figure 64:

9.32.2.1.3.1.4.1 Management of discharge mode with engine efficiency low

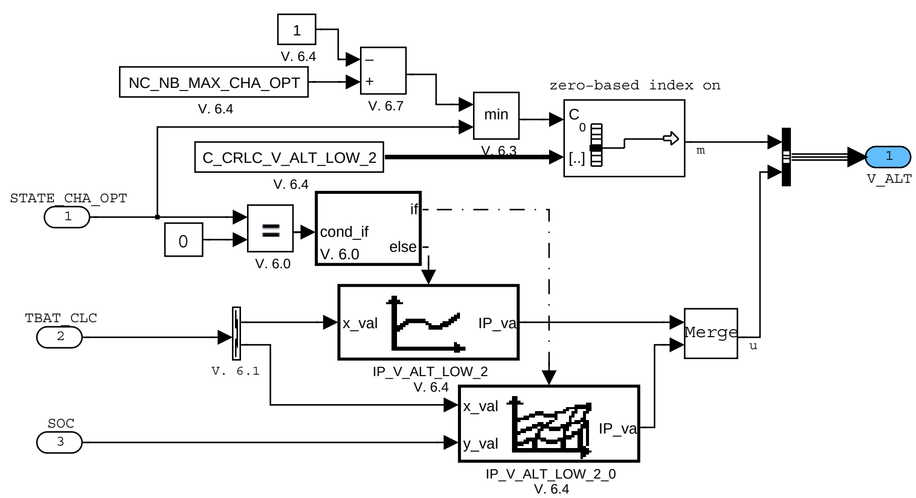


Figure 65:

9.32.2.1.3.1.4.2 Management of discharge mode with engine efficiency medium

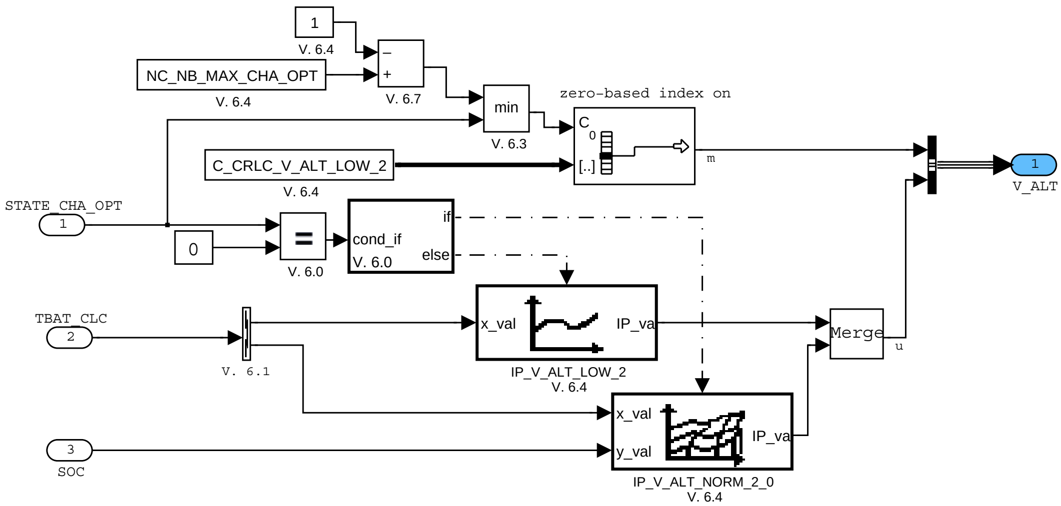


Figure 66:

9.32.2.1.3.1.4.3 Management of discharge mode with engine efficiency high

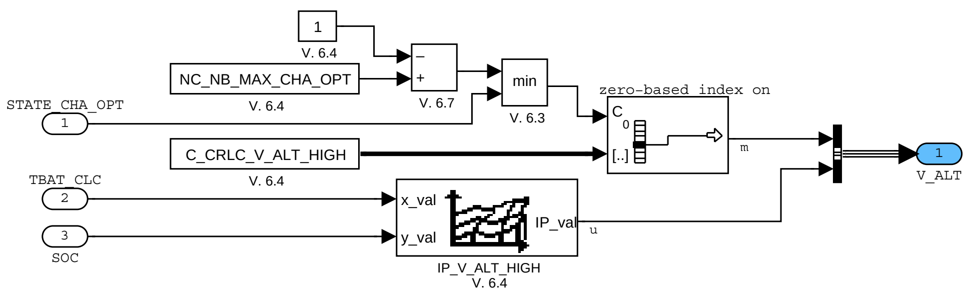


Figure 67:

9.32.2.1.3.2 The strategy charge discharge is bypassed

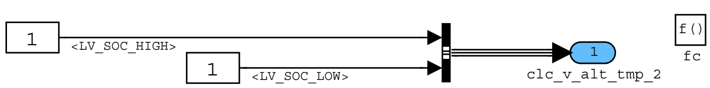


Figure 68:

9.32.2.2 Refreshment cycle started

In case of refreshment cycle is ON the alternator is set to a calibration to force the battery full charging

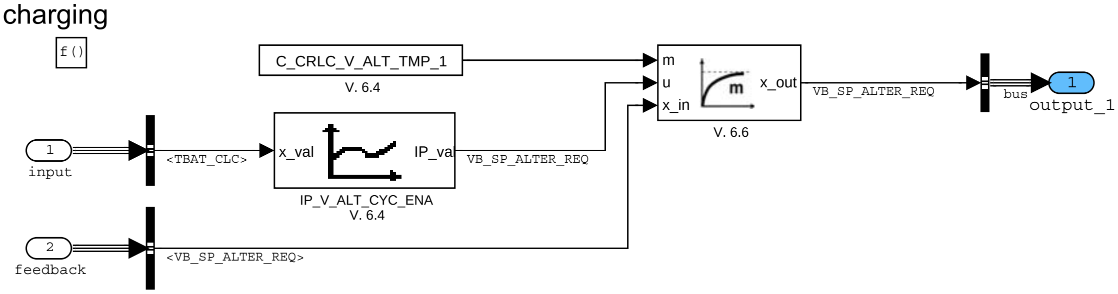


Figure 69:

9.33 Charge option arbitration

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_CHA_OPT	O/V	0...FFFFH	0...65535	1	[-]
Charge option state					
LV_PC_FAN_SWI PREL	V	0...1H	0...1	1	[-]
Preliminary Blower switch detected (0= low; 1= high)					
T_PC_FAN_SWI	V	0...FFH	0..25.5	0.1	sec
Timer to confirm preliminary blower switch					

Input data:

LV_HDLP	V_PC_FAN	LV_FATC_BLWR_MAX_CAN	CONF_PC_FAN
VB			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_STATE_CHA_OPT_INI	1	0...FFFFH	0...65535	1	[-]
Charge option state at initialization					
C_PC_FAN_HIGH	1	0...FFH	0...99.6	0.3905	[%]
Percentage of voltage level to detect blower "HIGH" state					
C_T_PC_FAN_SWI	1	0...FFH	0...25.5	0.1	[sec]
Time before switch of bit 1 of _STATE_CHA_OPT is allowed					

FUNCTION DESCRIPTION:

General information:

The charge option arbitration is calculate according system state :
- Bit 0=1: Head Lamp active
- Bit 1=1: Passenger Compartment Fan High
- Bit 2....10 not defined yet

For the Passenger compartment Fan "HIGH" detection the signal can be transmitted via CAN Information or Analog Input. To avoid switches at Analog input due to quick voltage drops caused by Inductance of blower motor a timer is introduced, which allows a switch only after the voltage level is confirmed during time C_T_PC_FAN_SWI.

Application conditions:

Initialisation: *at Reset:* STATE _CHA_OPT = C_STATE_CHA_OPT_INI
T_PC_FAN_SWI = C_T_PC_FAN_SWI

Recurrence: *100 ms*

Activation: always

Formula section:

if LV_HDLP = 1

```
then     STATE_CHA_OPT = x x x x x x x x x x 1 BIN
else     STATE_CHA_OPT = x x x x x x x x x x 0 BIN
endif

If       CONF_PC_FAN = 1
Then     If    V_PC_FAN < C_PC_FAN_HIGH * VB
         Then  LV_PC_FAN_SWI_PREL = 1
         Else  LV_PC_FAN_SWI_PREL = 0
         Endif
         If    LV_PC_FAN_SWI_PREL(n) <> LV_PC_FAN_SWI_PREL (n-1)
         Then  T_PC_FAN_SWI = C_T_PC_FAN_SWI
         Else  Decrement T_PC_FAN_SWI by 1h each 100msec
         Endif
         If    T_PC_FAN_SWI = 0
         Then  If    LV_PC_FAN_SWI_PREL = 1
               Then  STATE_CHA_OPT = x x x x x x x x x 1 x BIN
               Else  STATE_CHA_OPT = x x x x x x x x x 0 x BIN
               Endif
         Endif
Else     If    CONF_PC_FAN = 2
         Then  If    LV_FATC_BLWR_MAX_INPUT = 1
               Then  STATE_CHA_OPT = x x x x x x x x x 1 x BIN
               Else  STATE_CHA_OPT = x x x x x x x x x 0 x BIN
               Endif
         Else  STATE_CHA_OPT = x x x x x x x x x 0 x BIN
         Endif
Endif
```

9.34 Alternator Power Management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VB_SP_ALTER	V	0...FFH	10.6...16	0.0211	V
Alternator Voltage Setpoint					
VB_ALT_DIF	V	FF00..FFH	-5.4...5.4	0.0211	V
Difference between actual Battery voltage (VB_H_MMV) and Alternator Setpoint voltage					
ALTPWM_PCTL	V	0...7FFH	0...99.89	0.0488	%
Precontrol part of Alternator controller					
ALTPWM_I	V	F800...7FFH	-99.94...99.89	0.0488	%
I-part of Alternator controller					
ALTPWM_P	V	F800...7FFH	-99.94...99.89	0.0488	%
P-part of Alternator controller					
ALTPWM	V/O	0...7FFH	0...99.89	0.0488	%
Pulse width modulation directly after Alternator controller					
T_VB_ALT_DIF_THD_PWM_I_0	V	0...FFH	0...25.5	0.1	s
Timer for active I-part resetting at VB_ALT_DIF over C_VB_ALT_DIF_THD_PWM_I_0					
LV_INH_BEM	V	0...1H	0...1	1	-
Flag indicating BEM is inhibited					

Input data:

CONF_BAT	T_AST	VB_H_MMV	VB_SP_ALTER_REQ
QBAT_TOT	T_QBAT_TOT	TBAT_CLC	LV_ERR_BAT_SENS
LV_ERR_ALTER	LV_ERR_VB_PLAUS	TCO_ST	LV_ERR_ALTER_PLAUS
LV_SENS_LIN_READY	LV_ERR_ALTER_CHA_PLAUS		

FUNCTION DESCRIPTION:

General information:

The required System voltage is transmitted to the Alternator via PWM duty cycle.

The required Voltage setpoint is applied first by a Precontroller ALTPWM_PCTL. To adapt differences between the required Setpoint and the Actual value a closed loop control with PI-controller is realized. The system Voltage deviation is the input signal for the PI-controller.

With the C_CONF_ALT_MAN the controller value can be assigned to manual values C_ALTPWM_MAN. Additionally it is possible to apply an additive part C_ALTPWM_AS to the calculated value

Application conditions:

Inizialization: *at Reset:* ALTPWM_PCTL = 0%

ALTPWM_I = 0%

ALTPWM_P = 0%

ALTPWM = 0%

LV_INH_BEM = 0

Recurrence: 100 ms

Activation: CONF_BAT = 1

Deactivation: CONF_BAT = 0

Formula section:

Controller precalculations:

```
If         QBAT_TOT > C_QBAT_TOT_INH_BEM
     or    T_QBAT_TOT > C_T_QBAT_TOT_INH_BEM
     or    LV_SENS_LIN_READY = 0
then       LV_INH_BEM = 1
else       LV_INH_BEM = 0
endif

if    LV_INH_BEM = 0
then  if    T_AST < IP_T_AST_BEM
      then  VB_SP_ALTER  =  VB_SP_ALTER (n-1) + C_CRLC_ALTER_BEM_AST
                 * (IP_VB_SP_ALTER_BEM_AST  - VB_SP_ALTER(n-1))
                 Enable a slow increase of Alternator load after coldstart
      else  VB_SP_ALTER = VB_SP_ALTER_REQ
      endif
else  VB_SP_ALTER = VB_SP_ALTER (n-1) + C_CRLC_ALTER_INH_BEM *
      (IP_VB_SP_ALTER_INH_BEM   - VB_SP_ALTER(n-1))
endif

VB_ALT_DIF = VB_SP_ALTER – VB_H_MMV
```

Precontroller:

ALTPWM_PCTL = IP_ALTPWM_PCTL

Controller P-part:

ALTPWM_P = IP_ALTPWM_P_BAS * IP_ALTPWM_P_FAC

Controller I-part:

If the difference between the setpoint and actual value is bigger than a threshold the I-part is set to 0. This resetting is active only during the time C_T_VB_ALT_DIF_THD_PWM_I_0 since the time point when VB_ALT_DIF has exceeded the threshold.

```
if    abs(VB_ALT_DIF) > C_VB_ALT_DIF_THD_PWM_I_0
then  if    T_VB_ALT_DIF_THD_PWM_I_0 <
```

C_T_VB_ALT_ DIF_THD_PWM_I_0
then ALTPWM_I(n-1) = 0
T_VB_ALT_DIF_THD_PWM_I_0 =
T_VB_ALT_DIF_THD_PWM_I_0 + 0.1s
endif
else T_VB_ALT_DIF_THD_PWM_I_0 = 0
endif
ALTPWM_I(n) = ALTPWM_I(n-1) +
IP_ALTPWM_I_BAS(n) * IP_ALTPWM_I_FAC(n)
I-part is limited between –C_ALTPWM_I_LIM to C_ALTPWM_I_LIM:
-C_ALTPWM_I_LIM <= ALTPWM_I(n) <= C_ALTPWM_I_LIM

Resultant controller pulse width modulation:

if LC_ALTPWM_MAN = 0
then **if** LV_ERR_ALTER = 0
and LV_ERR_BAT_SENS = 0
and LV_ERR_VB_PLAUS = 0
and LV_ERR_ALTER_CHA_PLAUS = 0
and LV_ERR_ALTER_PLAUS = 0
then ALTPWM = ALTPWM_PCTL+ ALTPWM_P + ALTPWM_I +
C_ALTPWM_AS
Limit ALTPWM to the limits of the Alternator characteristics
C_ALTPWM_LIM_MIN < = ALTPWM < = C_ALTPWM_LIM_MAX
else ALTPWM = 0%
switch off Powerstage for Alternator control
endif
else ALTPWM = C_ALTPWM_MAN
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_ALTPWM_PCTL	8	0...7FFH	0...99.89	0.0488	%
LDPM_VB_SP_ALTER_ IP_ALTPWM	8	0...FFH	10.6...16	0.0211	V
Pre-control value for Alternator control					
IP_ALTPWM_P_BAS	8	0...FFFH	-99.94...99.89	0.0488	%
LDP_VB_ALT_DIF_IP_ALTPWM_P_BAS	8	0...1FFH	-5.4...5.4	0.0211	V
Basic table for P-part of Alternator control					
IP_ALTPWM_P_FAC	8	0...FFH	0...7.97	0.031	-
LDP_VB_H_MMV_IP_ALTPWM_P_FAC	8	0...3FFH	0...26V	0.0254	V
Correction factor for P-part of Alternator controller					
IP_ALTPWM_I_BAS	8	0...FFFH	-99.94...99.89	0.0488	%
LDP_ VB_ALT_DIF_IP_ALTPWM_I_BAS	8	0...1FFH	-5.4...5.4	0.0211	V
Basic table for I-part of Alternator control					
IP_ALTPWM_I_FAC	8	0...FFH	0...7.97	0.031	-
LDP_VB_H_MMV_IP_ALTPWM _I_FAC	8	0...3FFH	0...26V	0.0254	V
Correction factor for I-part of Alternator controller					
C_VB_ALT_DIF_THD_PWM_I_0	1	0..FFH	0...5.4	0.0211	V
VB_ALT_DIF over this threshold resets I-part of position control to 0					
C_T_VB_ALT_DIF_THD_PWM_I_0	1	0...FFH	0...25.5	0.1	s
Duration of active I-part resetting at VB_ALT_DIF over C_VB_ALT_DIF_THD_PWM_I_0					
C_ALTPWM_AS	1	0...7FFH	0...99.89	0.0488	%
Additive value to Alternator PWM (for calibration Purpose)					
C_ALTPWM_I_LIM	1	0...7FFH	0...99.89	0.0488	%
Limitation value for Alternator PWM I_Part					
C_ALTPWM_LIM_MIN	1	0...7FFH	0...99.89	0.0488	%
Minimum Limitation value for Alternator PWM					
C_ALTPWM_LIM_MAX	1	0...7FFH	0...99.89	0.0488	%
Maximum Limitation value for Alternator PWM					
C_ALTPWM_MAN	1	0...7FFH	0...99.89	0.0488	%
Manual value for Alternator PWM					
IP_VB_SP_ALTER_INH_BEM	8	0...FFH	10.6...16	0.0211	V
LDPM_TBAT_CLC_IP_V_ALT	8	0...FEH	-48...142.5	0.75	°C
Alternator Setpoint for Inhibition of BEM					
C_CRLC_ALTER_BEM_AST	1	0...FFH	0...0.9984	0.0039	-
Correlation constant for Activation of BEM after Start					
C_CRLC_ALTER_INH_BEM	1	0...FFH	0...0.9984	0.0039	-
Correlation constant for Inhibition of BEM					
C_QBAT_TOT_INH_BEM	1	0....7FFFFFFFh	0...120	55.87e-9	Ah
Max capacity threshold to inhibit BEM since last OCV calculation					
C_T_QBAT_TOT_INH_BEM	1	0...FFFFh	0....65535	1	Min
Max. Time threshold to inhibit BEM since last OCV calculation					
LC_ALTPWM_MAN	1	0...1H	0...1	1	-
Logical bit activating Manual Alternator PWM					
IP_VB_SP_ALTER_BEM_AST	8x8	0...FFH	10.6...16	0.0211	V
LDP_TCO_ST_IP_VB_SP_ALTER_	8	0...FEH	-48...142.5	0.75	°C
LDP_T_AST_IP_VB_SP_ALTER	8	0...FFFFH	0...6553.5	0.1	sec
Alternator voltage setpoint after start (to drvie with low Alternator performance)					
IP_T_AST_BEM	8	0...FFFFH	0...6553.5	0.1	sec
LDP_TCO_ST_ IP_T_AST_BEM	8	0...FEH	-48...142.5	0.75	°C
Time after start where Alternator voltage is driven with lower performance					

9.35 Cylinder Balancing Manager

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_IGA_ER_BAL	O/V	0H 1H 2H 3H	INITIALIZATION LOCK WAIT CYLINDER_BALANCING	1	-
States of the balancing manager					
LV_IGA_ADD_ER_BAL_ENA	O/V	0...1H	0...1	1	-
Flag for additive ignition angle correction active					
LV_IGA_ADD_AD_ER_BAL_ENA	O/V	0...1H	0...1	1	-
Flag for additive ignition angle adaptation active					
LV_ER_STND_MMV_MAX_IGA_ER_BAL	V	0...1H	0...1	1	-
Flag for maximum Filtered normalized engine roughness value					
LV_ER_STD_MMV_MAX_IGA_ER_BAL	V	0...1H	0...1	1	-
Flag for minimum Filtered normalized engine roughness value					
LV_IGA_ER_BAL_ENA	V	0...1H	0...1	1	-
Flag for cylinder balancing enabled (local)					
CTR_IGA_ER_BAL_ENA	V	0...FFFFH	0...6.5535E+4	1	-
Engine cycle synchr. counter to delay transition from wait state to cylinder balancing state					

Input data:

ER_STND_MMV_BAL[NC_CYL_NR]	ER_STD_MMV_BAL[NC_CYL_NR]	LV_IGA_ER_BAL_ENA_EXT	LC_ER_BAL_STOP_MAN
LV_IGA_AD_ER_BAL_ENA_EXT	LV_DRV1_STND_BAL_FDOUT	TCO	LV_IS
SEG_NR_ER	NC_CYL_NR		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_IGA_ER_BAL_ENA_INI	1	0...FFFFH	0...6.5535E+4	1	-
Initial value for engine cycle synchr. counter to delay transition from wait state to cylinder balancing state					
C_ER_STD_MMV_MAX_IGA_ER_BAL	1	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Maximal allowed standard deviation for engine roughness					
C_ER_STND_MMV_MAX_IGA_ER_BAL	1	8000...7FFFH	-325.78...325.77	0.00994202	1/s²
Maximal allowed value for engine roughness					
C_TCO_MAX_IGA_ADD_ER_BAL	1	0...FEH	-48...142.5	0.75	°C
Upper limit of the coolant temperature-range for additive adaption					
C_TCO_MIN_IGA_ADD_AD_ER_BAL	1	0...FEH	-48...142.5	0.75	°C
Lower limit of the coolant temperature-range for additive adaption					
C_TCO_MIN_IGA_ADD_ER_BAL	1	0...FEH	-48...142.5	0.75	°C
Lower limit of the coolant temperature-range for additive adaption					
LC_IGA_ADD_ER_BAL_ENA	1	0...1H	0...1	1	-
Flag for additive adaption enabled out of idle					

9.35.1 General information:

Cylinder Balancing via Engine Roughness means a correction of the cylinder individual ignition angles to achieve a balancing of the indicated torque (TQI) between the single cylinders based on Engine Roughness calculation. Target is to achieve a smooth engine running.

In order to compensate these torque differences, e.g. caused by manufacturing tolerances of the intake manifold, an additive adaptation correction of the ignition angles is carried out depending on different operation conditions.

The functionality of Balancing Management is realized in form of a state machine. That means that there are states with corresponding settings of outputs, and there are conditions to go from one to another state. The calculation of the adaptation values is controlled by the different states of the state machine.

By means of the logical constant LC_IGA_ADD_ER_BAL_ENA it is possible to enable (LC_IGA_ADD_ER_BAL_ENA = 1) or disable (LC_IGA_ADD_ER_BAL_ENA = 0) the additive adaptation in part load. At idle speed the additive adaptation is always enabled.

Application Condition

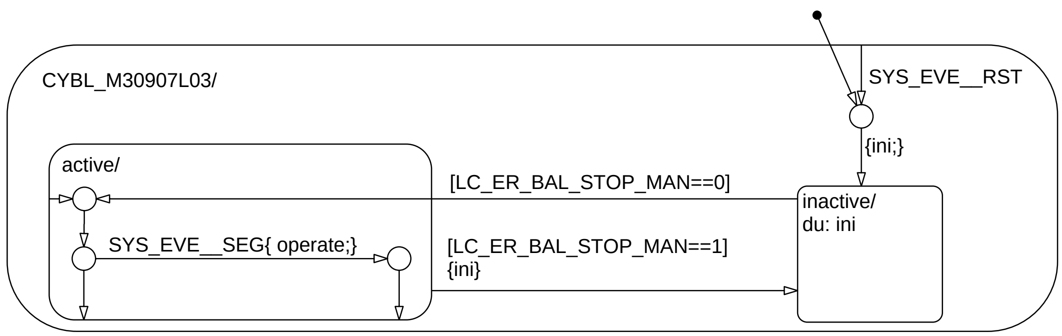


Figure 70 CYBL_M30907L01/ APP_CDN/ Chart

Function Description

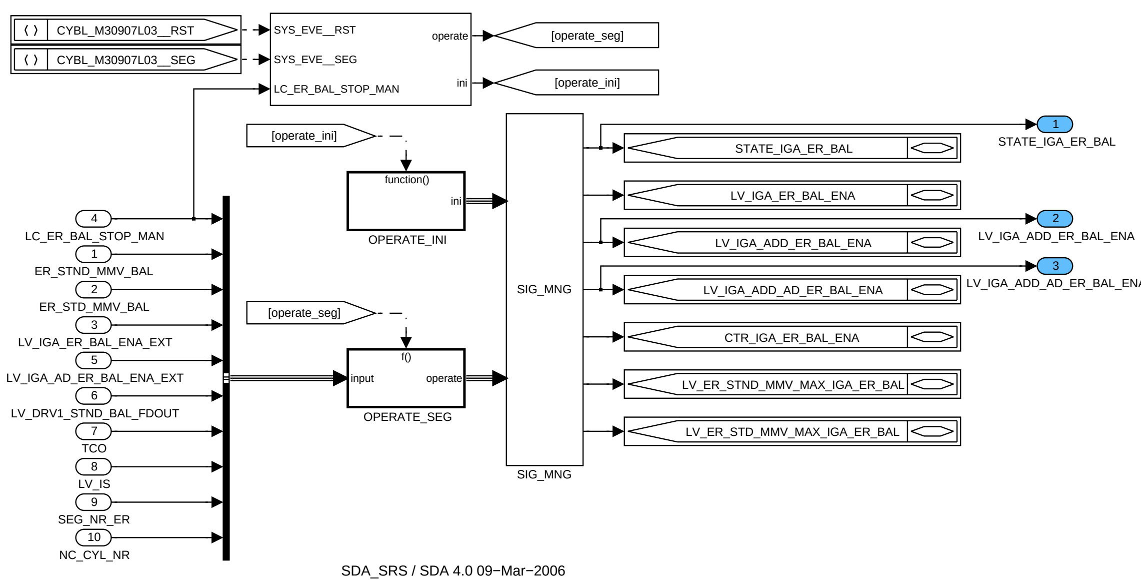


Figure 71 CYBL_M30907L01

9.35.1.1 Calculation of variables at reset task

All output data are set to "0" [hex]

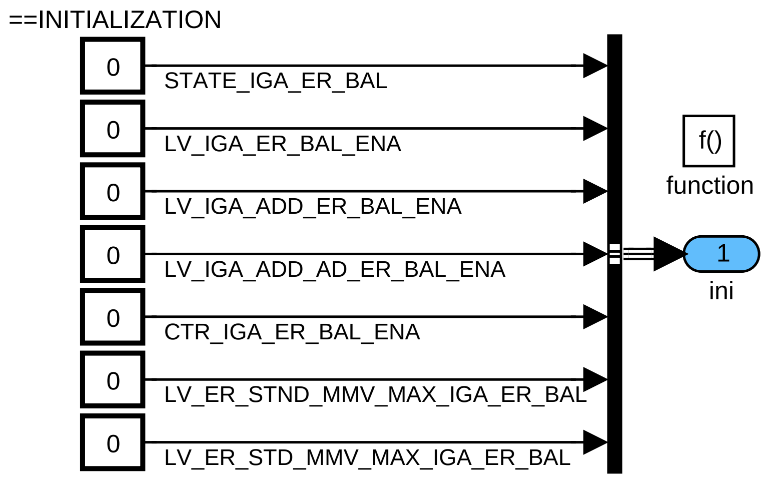


Figure 72 CYBL_M30907L01/ OPERATE_INI

9.35.1.2 Calculation of segment synchronous task

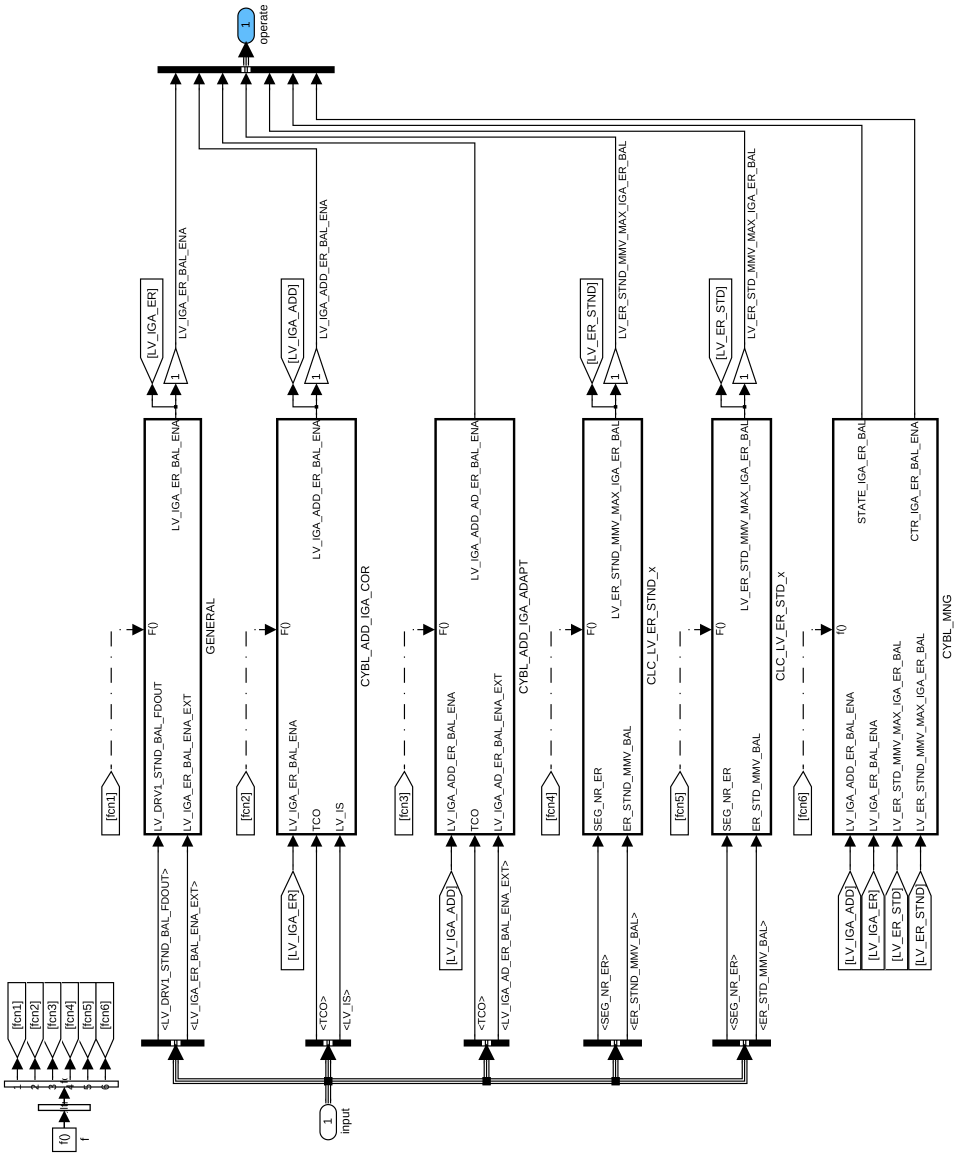


Figure 73 CYBL_M30907L01/ OPERATE_SEG

<u>Calculation of segment synchronous task GENERAL</u>

LV_DRV1_STND_BAL_FDOUT (acceleration fade out)

LV_IGA_ER_BAL_ENA_EXT (from Appl.-Inc.)

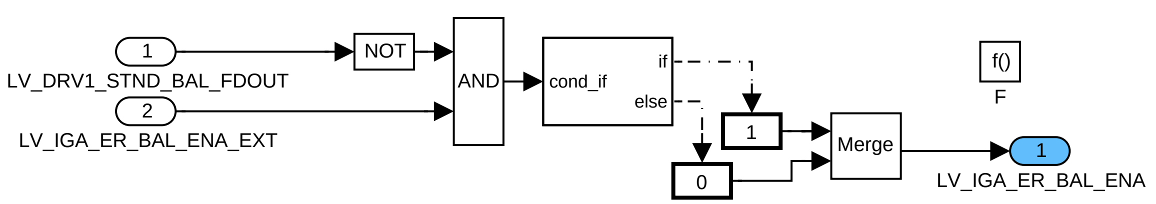


Figure 74 CYBL_M30907L01/ OPERATE_SEG/ GENERAL

Calculation of segment synchronous task CYBL_ADD_IGA_COR

LV_IGA_ADD_ER_BAL_ENA (Cylinder balancing additive ignition angle correction)

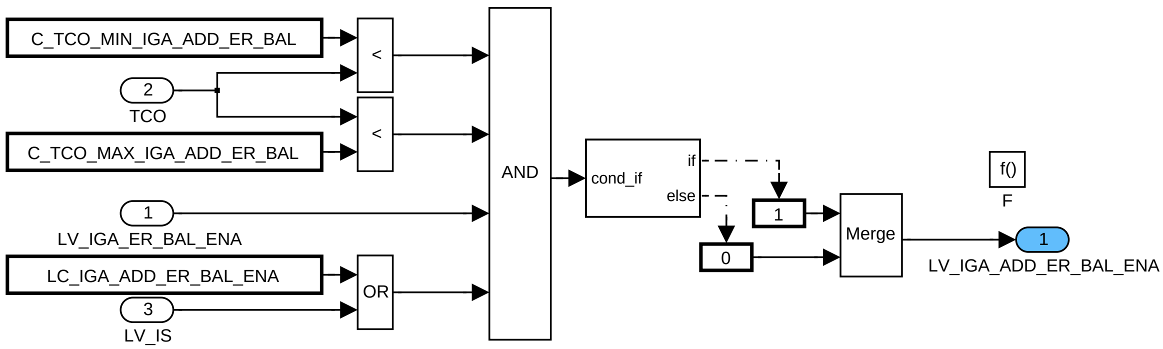


Figure 75 CYBL_M30907L01/ OPERATE_SEG/ CYBL_ADD_IGA_COR

Calculation of segment synchronous task CYBL_ADD_IGA_ADAPT

LV_IGA_ADD_AD_ER_BAL_ENA (Cylinder balancing additive ignition angle adaptation)

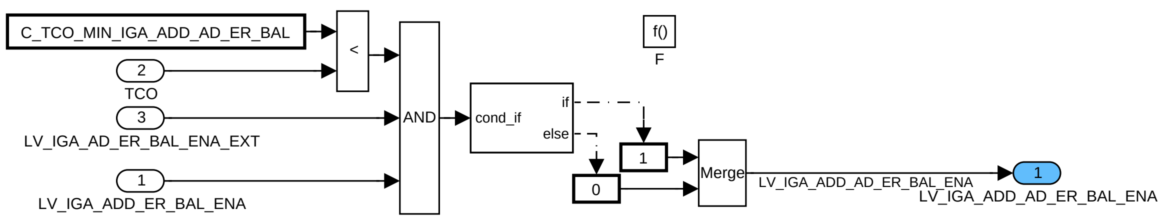


Figure 76 CYBL_M30907L01/ OPERATE_SEG/ CYBL_ADD_IGA_ADAPT

Calculation of segment synchronous task CLC_LV_ER_STND_X

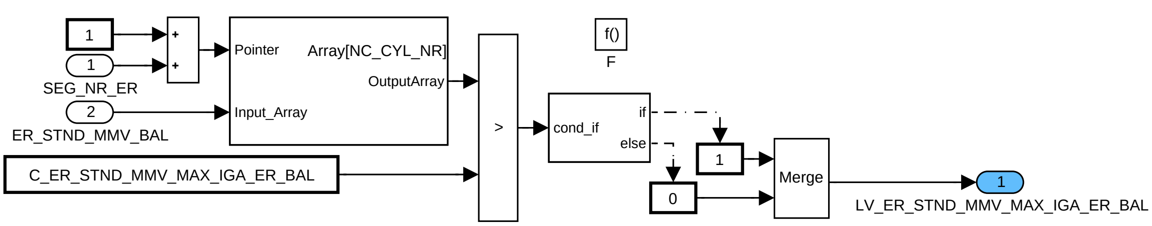


Figure 77 CYBL_M30907L01/ OPERATE_SEG/ CLC_LV_ER_STND_x

Calculation of segment synchronous task CLC_LV_ER_STD_X

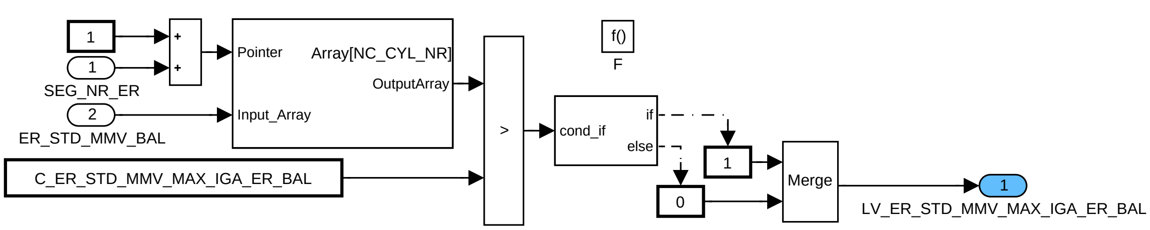
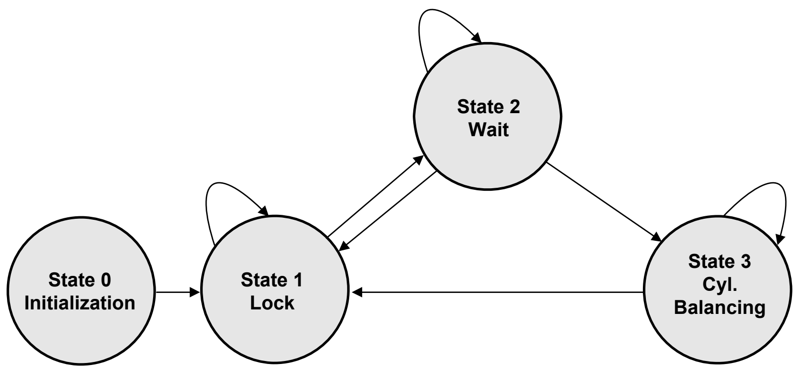


Figure 78 CYBL_M30907L01/ OPERATE_SEG/ CLC_LV_ER_STD_x

Calculation of segment synchronous task CYBL_MNG

Description of the Cylinder Balancing Manager:

State diagram:



Cylinder Balancing Manager: State 0[H] – "Initialization"

After initializing the state changes at once to state "Lock". The state "Initialization" is active for at least one recurrence.

Cylinder Balancing Manager: State 1[H] – "Lock"

The algorithm stays in this state until the flags LV_IGA_ER_BAL_ENA is set to "1". This flag is set when certain conditions for controlling and adaptation are fulfilled.

Cylinder Balancing Manager: State 2[H] – "Wait"

After transition to this state the function – Engine Roughness Signal Preparation for Cylinder Balancing – starts to calculate.

The counter CTR_IGA_ER_BAL_ENA starts to run. It counts down from the calibrate able initial value C_CTR_IGA_ER_BAL_ENA_INI to zero (decrease of "1" every engine cycle).

Furthermore the input values ER_STND_MMV_BAL[SEG_NR_ER] and ER_STD_MMV_BAL[SEG_NR_ER] has to below a certain limit:

Cylinder Balancing Manager: State 3[H] - Cylinder Balancing

After transition to this state the calculation described in the Cylinder Balancing functions (Engine roughness signal preparation for Cylinder balancing, Cylinder balancing via IGA intervention) start to run.

The input values ER_STND_MMV_BAL[SEG_NR_ER] and ER_STD_MMV_BAL[SEG_NR_ER] have to be below a certain limit (analogue to state 2).

In order to compensate torque differences, e.g. caused by manufacturing tolerances of the intake manifold system, an additive adaptation correction of the applied ignition angle is carried out in the state "Cylinder Balancing" depending on several activation conditions.

The additive adaptation correction is applied only in this state (Cylinder Balancing).

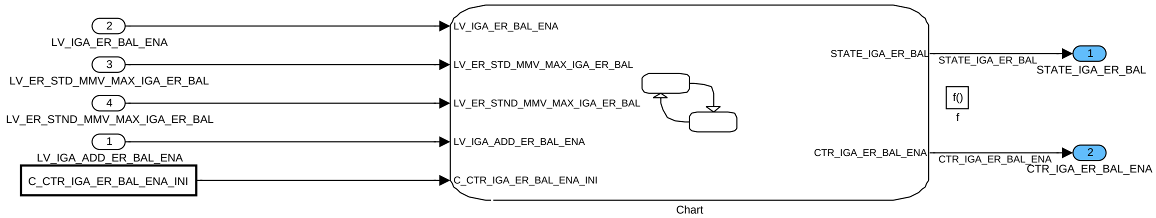


Figure 79 CYBL_M30907L01/ OPERATE_SEG/ CYBL_MNG

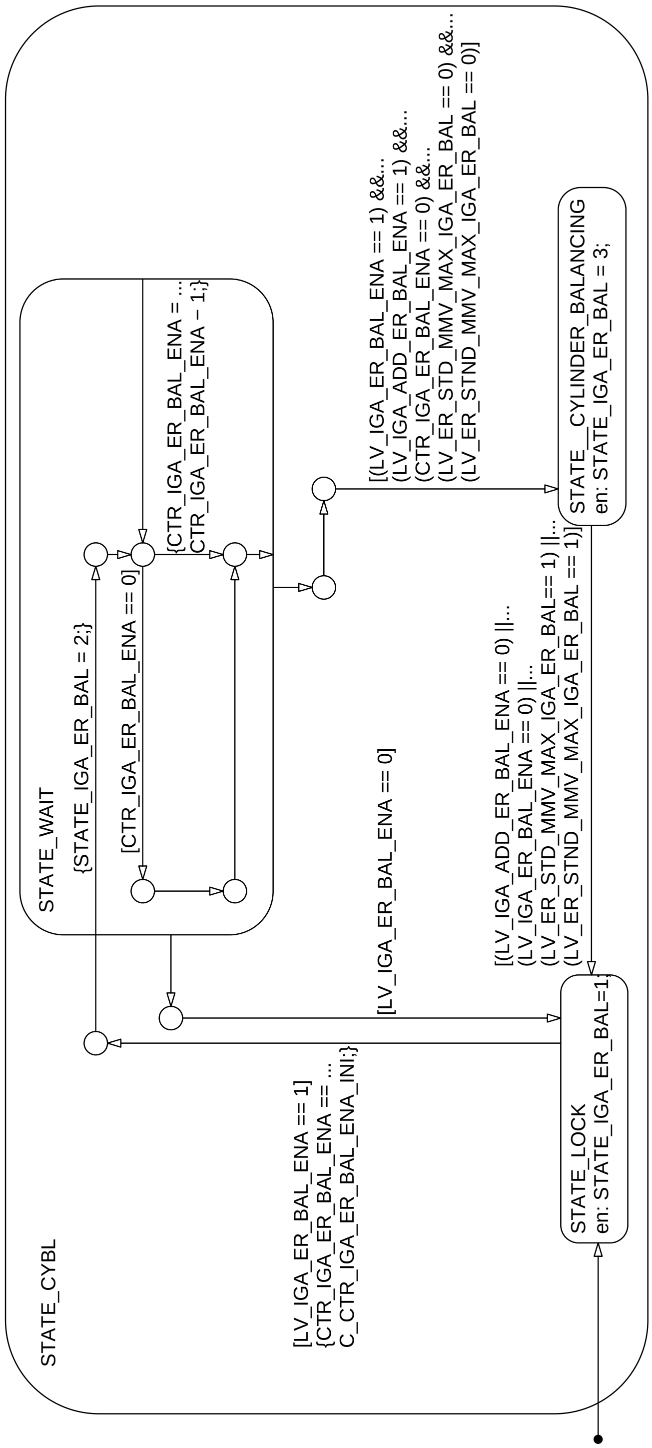


Figure 80 CYBL_M30907L01/ OPERATE_SEG/ CYBL_MNG/ Chart

9.35.1.3 SUBFUNCTION: SIG_MNG

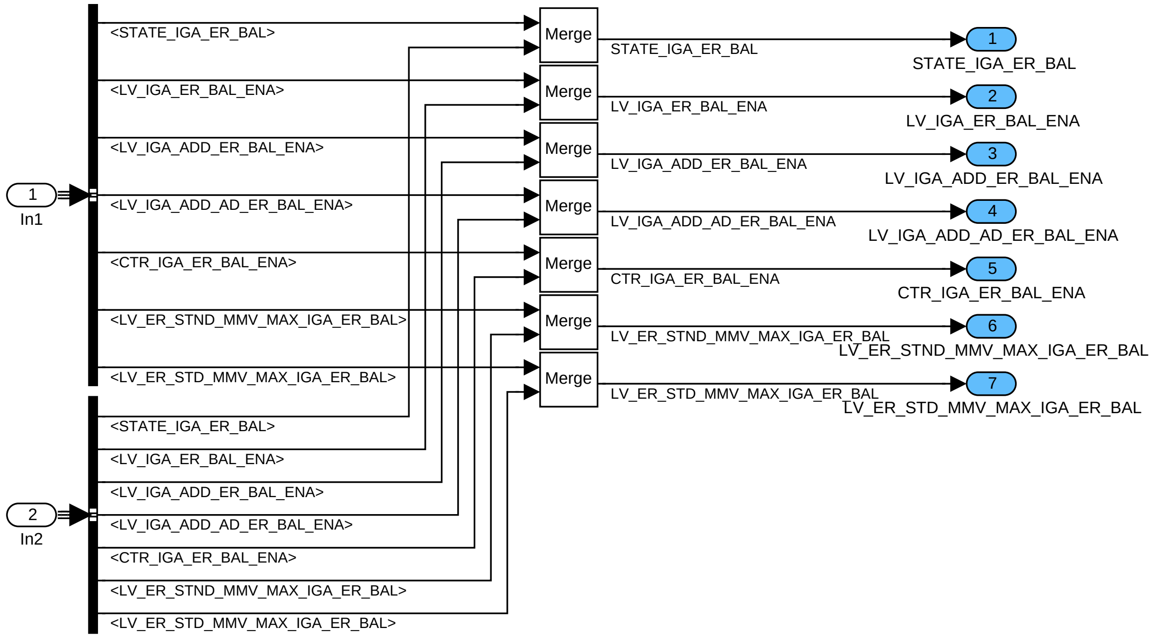


Figure 81 CYBL_M30907L01/ SIG_MNG

9.36 Cylinder balancing manager (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ER_STND_ER_BAL_ACT	O/V	0...1H	0...1	1	-
Activation condition to enable cylinder balancing signal preparation calculation					
LV_DRV1_ER_BAL_ACT	O/V	0...1H	0...1	1	-
Activation condition to enable cylinder balancing fade out calculation					
LV_IGA_ER_BAL_ACT	O/V	0...1H	0...1	1	-
Activation condition to enable cylinder balancing IGA intervention calculation					
LV_IGA_ER_BAL_ENA_EXT	O/V	0...1H	0...1	1	-
External condition to enable cylinder balancing manager calculations					
LV_IGA_AD_ER_BAL_ENA_EXT	O/V	0...1H	0...1	1	-
External condition to enable cylinder balancing adaptation manager calculations					
LV_IGA_ADD_AD_ER_BAL_EXT_ADJ	O/V	0...1H	0...1	1	-
External adjustment of correction values for cylinder balancing via IGA intervention					

Input data:

STATE_IGA_ER_BAL	N_32	N_DIF	VS
TQI_AV	TQI_BAS	EFF_IGA_AV	EFF_IGA_MIN
LV_ENA_ER	LV_SEG_AD_LIM_ER	LV_SEG_AD_AVL_ER	LV_ENA_SEG_T_MES
LV_MIS_STATE_A	LV_MIS_STATE_B4	LV_ERR_CRK	LV_LIH_ERR_CRK
LV_ERR_CAM	LV_REQ_ISC	LV_ERR_ISC	LV_ERR_TCO
LV_ERR_TPS	LV_ACT_DIAGCPS	LV_ERR_EL_CPS	LV_ERR_MEC_OPEN_CPS
LV_PUC	LV_CH	LV_LAM_LSCL[NC_CBK_EX_NR]	T_AST
CL_MMV	GEAR	NC_CBK_EX_NR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CL_MMV_MAX_IGA_AD_ER_BAL	1	0...FFFFH	0...1.99996948	3.05176E-5	-
Maximum canister load for cylinder balancing adaptation via IGA intervention					
C_DRV1_ER_BAL_ACT_MAN	1	0H 1H 2H	NEUTRAL ENABLE DISABLE	1	-
Manual adjustment of activation conditions for cylinder balancing fade out					
C_EFF_IGA_MIN_IGA_AD_ER_BAL	1	0...FFFFH	0...1.99996948	3.05176E-5	-
Minimum ignition angle efficiency for cylinder balancing adaptation via IGA intervention					
C_ER_STND_ER_BAL_ACT_MAN	1	0H 1H 2H	NEUTRAL ENABLE DISABLE	1	-
Manual adjustment of activation conditions for cylinder balancing signal preparation					
C_IGA_ER_BAL_ACT_MAN	1	0H 1H 2H	NEUTRAL ENABLE DISABLE	1	-
Manual adjustment of activation conditions for cylinder balancing IGA intervention					
C_IGA_ER_BAL_COD_MAN	1	0...FFH	0...255	1	-
Code word for manual adjustment of activation conditions for cylinder balancing IGA intervention					
C_N_32_MAX_IGA_ER_BAL	1	0...FFH	0...8.16E+3	32	rpm
Maximum engine speed for cylinder balancing via IGA intervention					
C_N_32_MIN_IGA_ER_BAL	1	0...FFH	0...8.16E+3	32	rpm
Minimum allowed Engine Speed for Cylinder Balancing via IGA intervention					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DIF_MAX_IGA_ER_BAL	1	0...1FE0H	0...8.16E+3	1	rpm
Maximum engine speed deviation for cylinder balancing via IGA intervention					
C_TQI_MAX_IGA_ER_BAL	1	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Maximum engine torque for cylinder balancing via IGA intervention					
C_TQI_MIN_IGA_ER_BAL	1	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Minimum allowed Engine Torque for Cylinder Balancing via IGA intervention					
C_TQ_ADD_MIN_IGA_ER_BAL	1	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Minimum engine torque reserve for cylinder balancing via IGA intervention					
C_T_AST_MIN_IGA_ER_BAL	1	0...FFFFH	0...6.5535E+3	0.1	s
Minimum time after engine start for cylinder balancing via IGA intervention					
C_VS_MAX_IGA_ER_BAL	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for cylinder balancing via IGA intervention					
LC_ER_BAL_STOP_MAN	1	0...1H	0...1	1	-
Flag for additive adaption enabled out of idle					

9.36.1 General information:

The Appl. Inc. module is used to keep the cylinder balancing functions modular. Therefore the activation conditions for the signal preparation functions, the balancing manager and the cylinder balancing adaptation functions are set within this module. The cylinder balancing functions are activated as soon as the corresponding activation condition is enabled (=1).

With the logical variable LV_IGA_ER_BAL_ENA_EXT it is possible to adapt the main enable condition (to start cylinder balancing via IGA intervention) with additional external requirements. The flag is used as input signal for the cylinder balancing manager.

With the logical variable LV_IGA_AD_ER_BAL_ENA_EXT it is possible to adapt the main adaptation condition (to start cylinder balancing adaptation via IGA intervention) with additional external requirements. The flag is used as input signal for the cylinder balancing manager.

A logical calibration constant is available to stop all cylinder balancing interventions at once. In case of LC_ER_BAL_STOP_MAN = 1, several activation conditions for the cylinder balancing functions are disabled (=0). If the logical constant is enabled (=1), a re-initialisation ("0"[hex]) of all output values occurs.

Application Condition

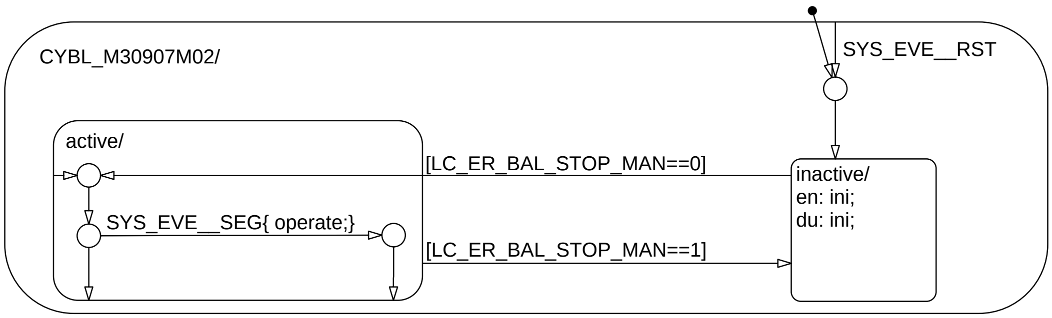


Figure 82 CYBL_M30907M01/ APP_CDN/ Chart

Function Description

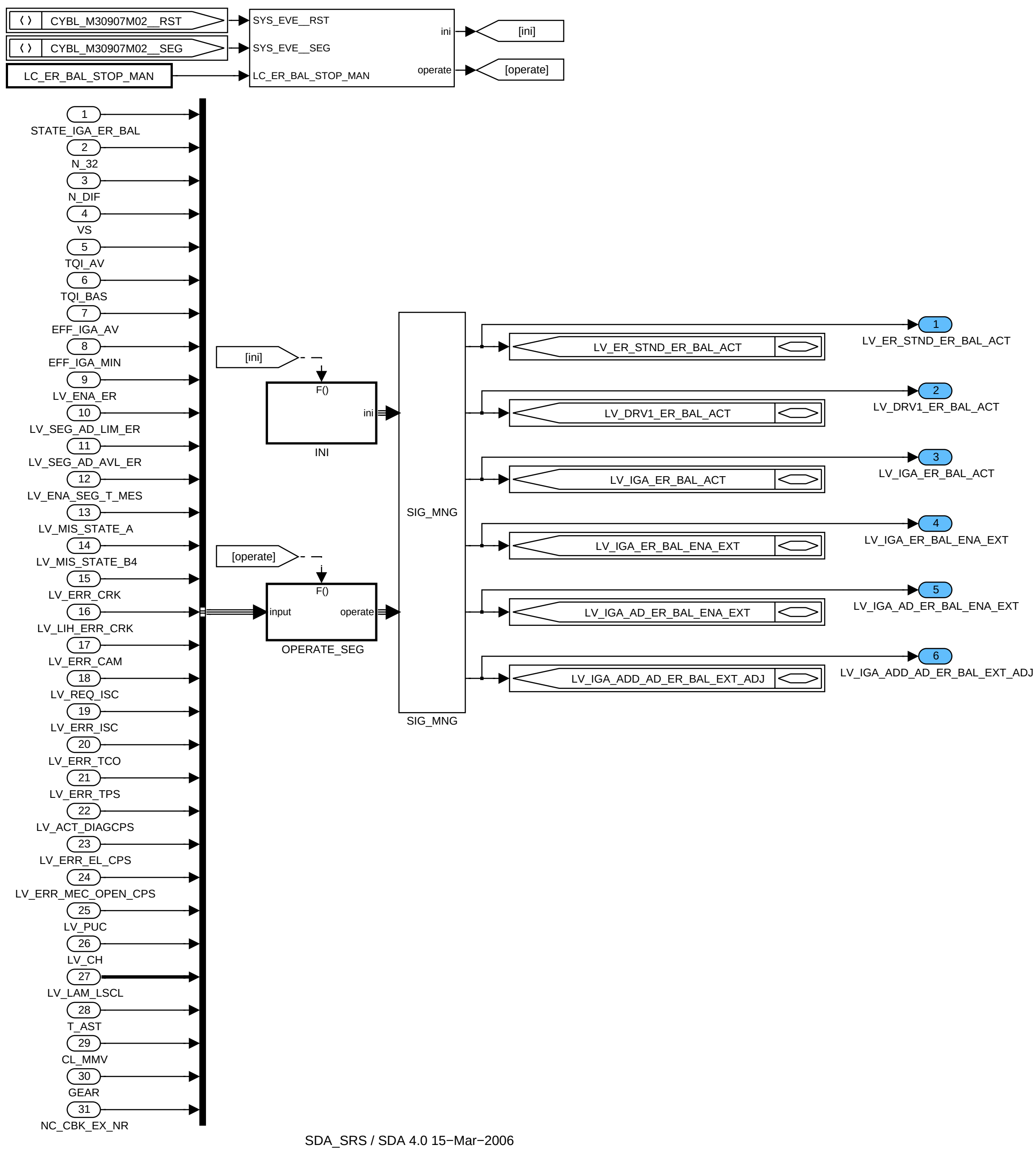


Figure 83 CYBL_M30907M01

9.36.1.1 Calculation of variables at reset task

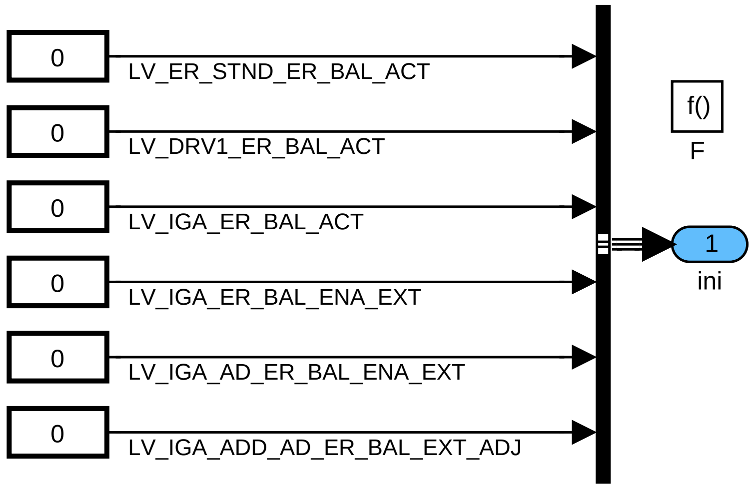


Figure 84 CYBL_M30907M01/ INI

9.36.1.2 Calculation of variables at segment synchronous task

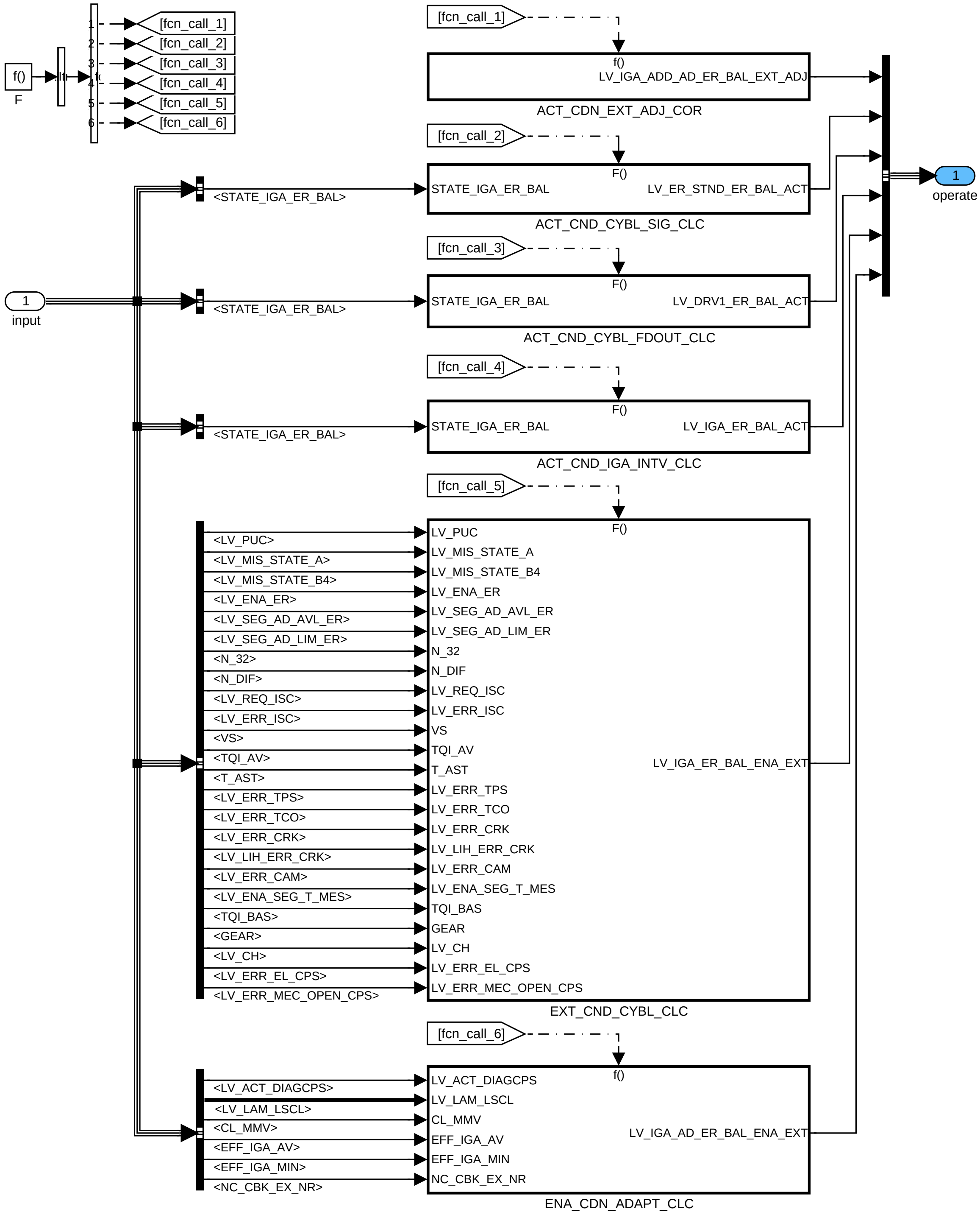


Figure 85 CYBL_M30907M01/ OPERATE_SEG

Activation conditions for external adjustment of cylinder balancing correction values

Figure 86 CYBL_M30907M01/ OPERATE_SEG/ ACT_CDN_EXT_ADJ_COR

Activation conditions for cylinder balancing signal preparation calculation

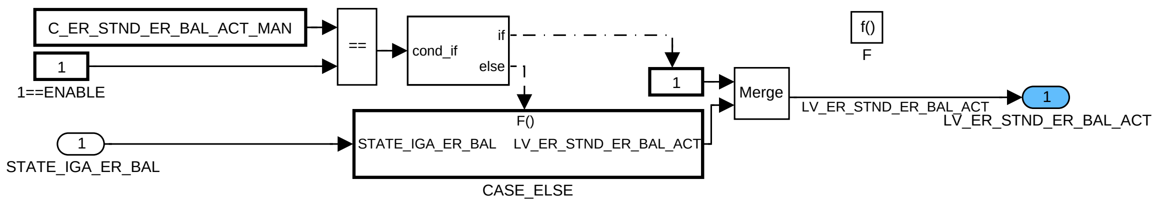


Figure 87 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_SIG_CLC

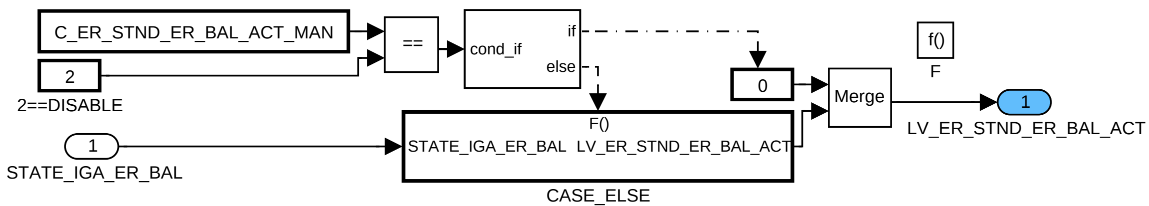


Figure 88 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_SIG_CLC/ CASE_ELSE

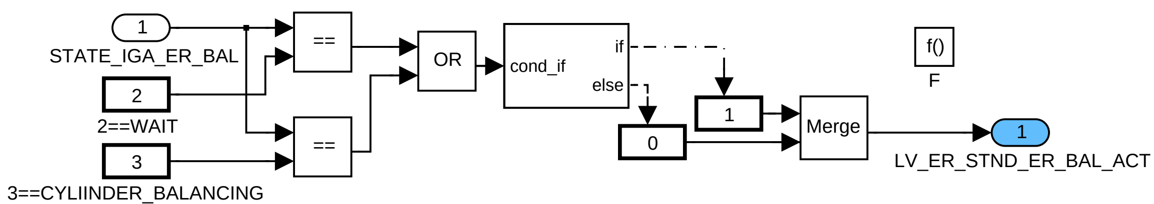


Figure 89 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_SIG_CLC/ CASE_ELSE/ CASE_ELSE

Activation conditions for cylinder balancing fade out calculation

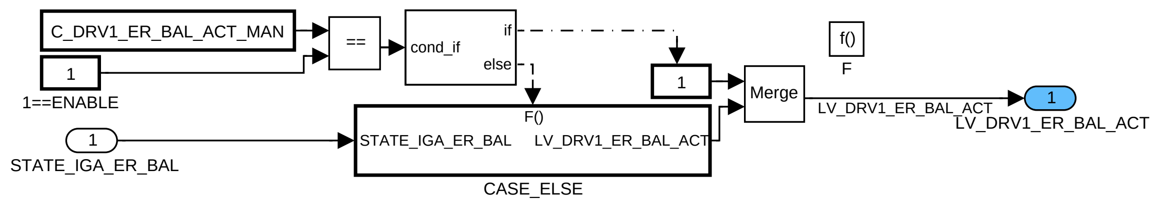


Figure 90 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_FDOUT_CLC

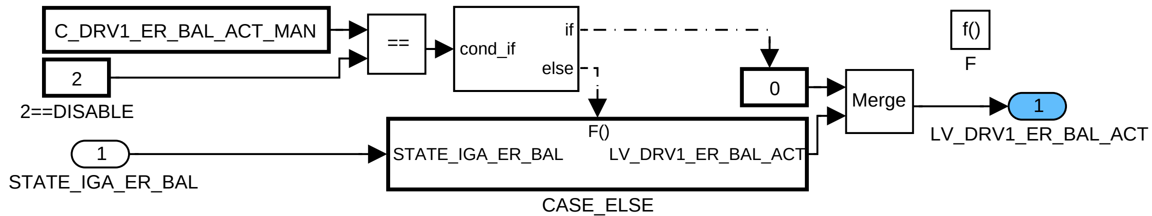


Figure 91 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_FDOUT_CLC/ CASE_ELSE

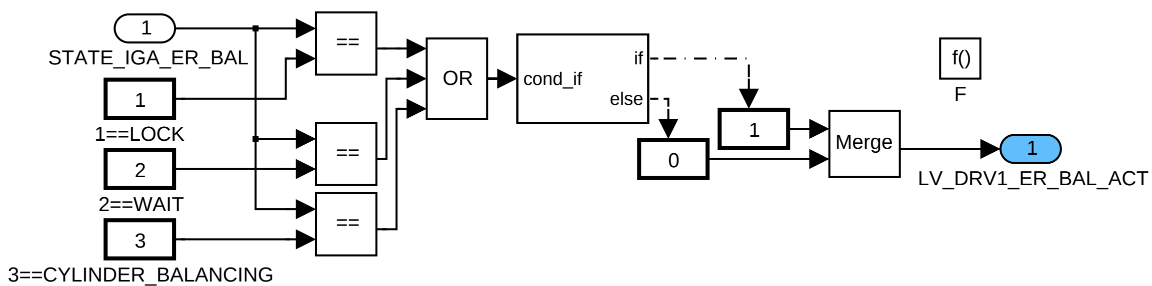


Figure 92 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_CYBL_FDOUT_CLC/ CASE_ELSE/ CASE_ELSE

Activation conditions for cylinder balancing IGA intervention calculation

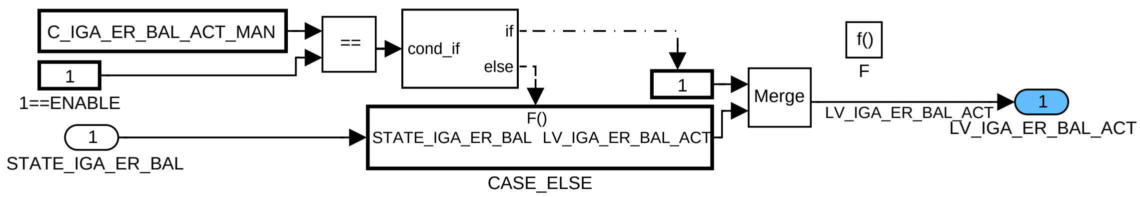


Figure 93 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_IGA_INTV_CLC

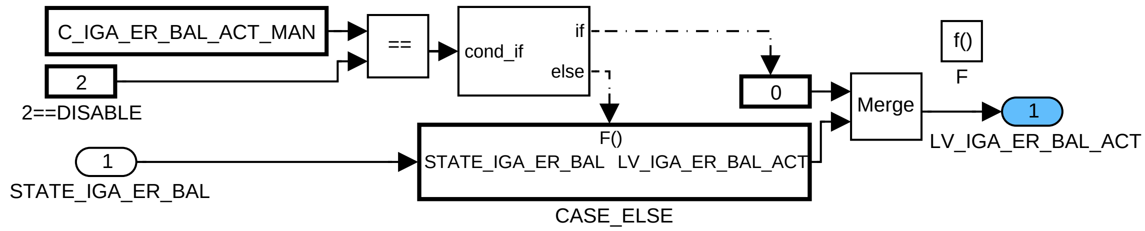


Figure 94 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_IGA_INTV_CLC/ CASE_ELSE

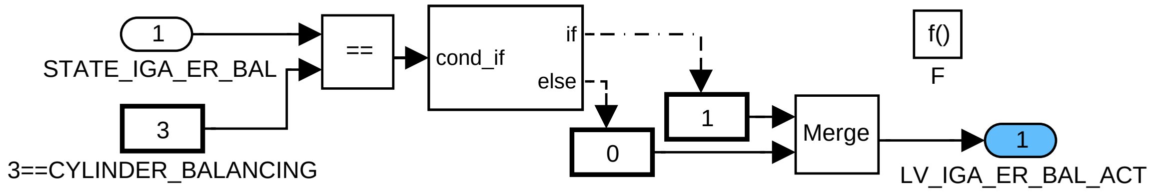


Figure 95 CYBL_M30907M01/ OPERATE_SEG/ ACT_CND_IGA_INTV_CLC/ CASE_ELSE/ CASE_ELSE

Enable condition for cylinder balancing manager calculations

LV_PUC (engine operation state "pull fuel cutoff" not active)

LV_MIS_STATE_A (no CARB A[200 CRK] misfire failure present)

LV_MIS_STATE_B4 (no CARB B4[1000 CRK] misfire failure present)

LV_ENA_ER (calculation of engine roughness (ER) values valid)

LV_SEG_AD_AVL_ER (segment adaptation process achieved at least one time)

LV_SEG_AD_LIM_ER (engine roughness (ER) adaptation values not out of range)

N_32 > C_N_32_MIN_IGA_ER_BAL (minimum engine speed for cylinder balancing)

N_32 < C_N_32_MAX_IGA_ER_BAL (maximum engine speed for cylinder balancing)

N_DIF < C_N_DIF_MAX_IGA_ER_BAL (engine speed deviation for idle speed control within range)

LV_REQ_ISC (idle speed controller activated)

LV_ERR_ISC (no present failure on idle speed control)

VS < C_VS_MAX_IGA_ER_BAL (maximum vehicle speed for cylinder balancing)

TQI > C_TQI_MIN_IGA_ER_BAL (minimum indicated engine torque for cylinder balancing)

TQI_AV < C_TQI_MAX_IGA_ER_BAL (maximum indicated engine torque for cylinder balancing)

T_AST > C_T_AST_MIN_IGA_ER_BAL (minimum time after start for cylinder balancing)

LV_ERR_TPS (no failure present on throttle position sensor)

LV_ERR_TCO (no failure present on coolant temperature sensor)

LV_ERR_CRK (no failure present on crankshaft sensor)

LV_LIH_ERR_CRK (no limp home failure present on crankshaft sensor)

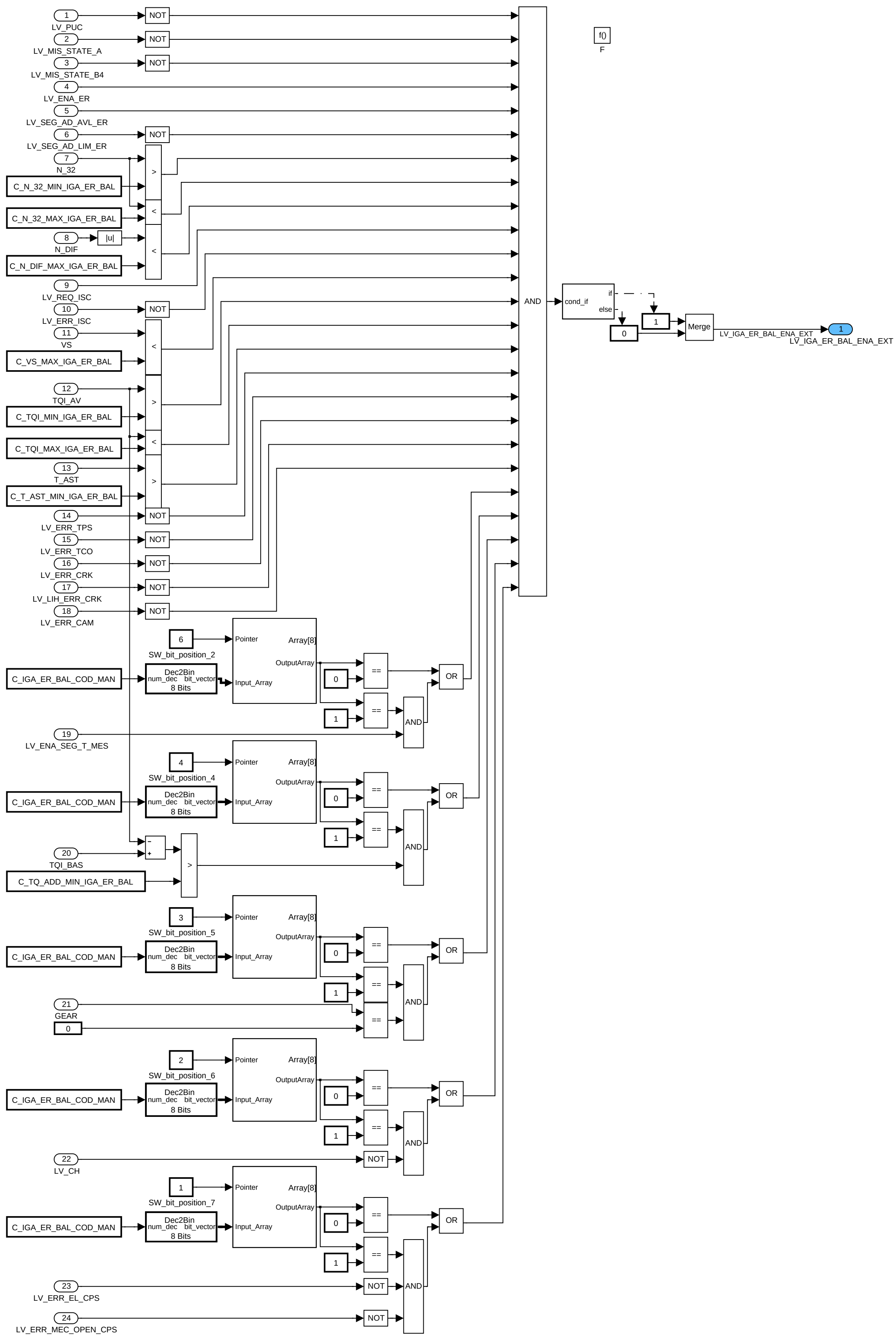
LV_ERR_CAM (no failure present on camshaft sensor)

C_IGA_ER_BAL_COD_MAN[bit2]... (engine roughness (ER) segment time values valid)

C_IGA_ER_BAL_COD_MAN[bit4]... (minimum torque reserve available for cylinder balancing)

C_IGA_ER_BAL_COD_MAN[bit5]... ("P" or "N" gear shift request for automatic transmission (AT) vehicles)

C_IGA_ER_BAL_COD_MAN[bit7]... (no electrical or mechanical failure present on canister purge)



1 LV_PUC
2 LV_MIS_STATE_A
3 LV_MIS_STATE_B4
4 LV_ENA_ER
5 LV_SEG_AD_AVL_ER
6 LV_SEG_AD_LIM_ER
7 N_32
C_N_32_MIN_IGA_ER_BAL
C_N_32_MAX_IGA_ER_BAL
8 N_DIF
|u|
C_N_DIF_MAX_IGA_ER_BAL
9 LV_REQ_ISC
10 LV_ERR_ISC
11 VS
C_VS_MAX_IGA_ER_BAL
12 TQI_AV
C_TQI_MIN_IGA_ER_BAL
C_TQI_MAX_IGA_ER_BAL
13 T_AST
C_T_AST_MIN_IGA_ER_BAL
14 LV_ERR_TPS
15 LV_ERR_TCO
16 LV_ERR_CRK
17 LV_LIH_ERR_CRK
18 LV_ERR_CAM
NOT
AND
f() F
cond_if if else
0
1
Merge
LV_IGA_ER_BAL_ENA_EXT
1 LV_IGA_ER_BAL_ENA_EXT
6 SW_bit_position_2
C_IGA_ER_BAL_COD_MAN
Dec2Bin num_dec bit_vector 8 Bits
Pointer Array[8] OutputArray Input_Array
==
OR
19 LV_ENA_SEG_T_MES
4 SW_bit_position_4
20 TQI_BAS
C_TQ_ADD_MIN_IGA_ER_BAL
3 SW_bit_position_5
21 GEAR
2 SW_bit_position_6
22 LV_CH
1 SW_bit_position_7
23 LV_ERR_EL_CPS
24 LV_ERR_MEC_OPEN_CPS

Figure 96 CYBL_M30907M01/ OPERATE_SEG/ EXT_CND_CYBL_CLC

Enable conditions for cylinder balancing adaptation manager calculations

LV_ACT_DIAGCPS (canister purge functional check (diagnosis) not activated)

C_IGA_ER_BAL_COD_MAN[bit1]... (lambda control of all cylinder banks (exhaust lines) have to be activated, [i = number of cylinder banks])

C_IGA_ER_BAL_COD_MAN[bit3]... (maximum canister load for cylinder balancing adaptation)

C_IGA_ER_BAL_COD_MAN[bit6]... (minimum ignition efficiency for cyllinder balancing adaptation)

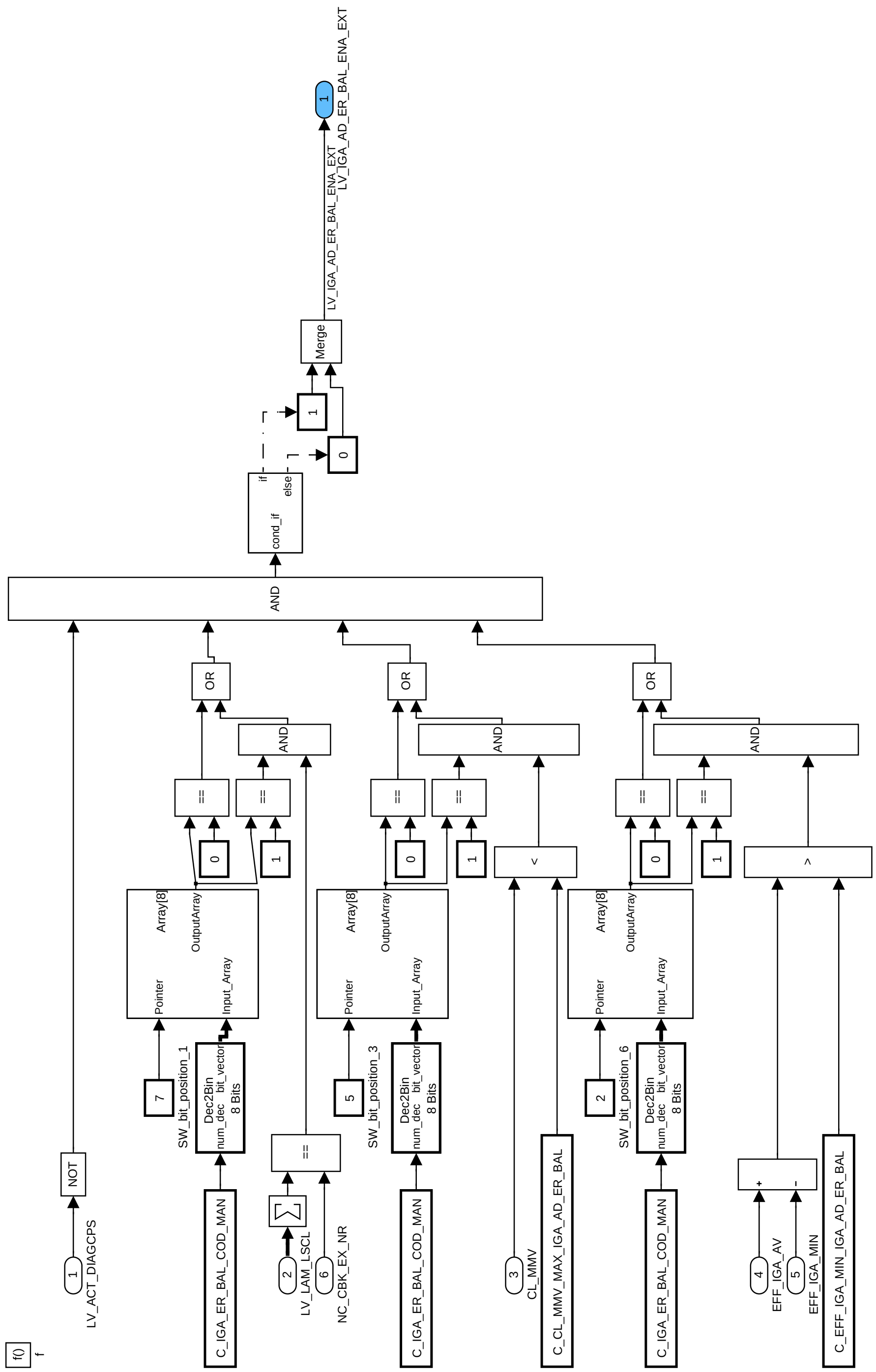


Figure 97 CYBL_M30907M01/ OPERATE_SEG/ ENA_CDN_ADAPT_CLC

9.36.1.3 SUBFUNCTION: SIG_MNG

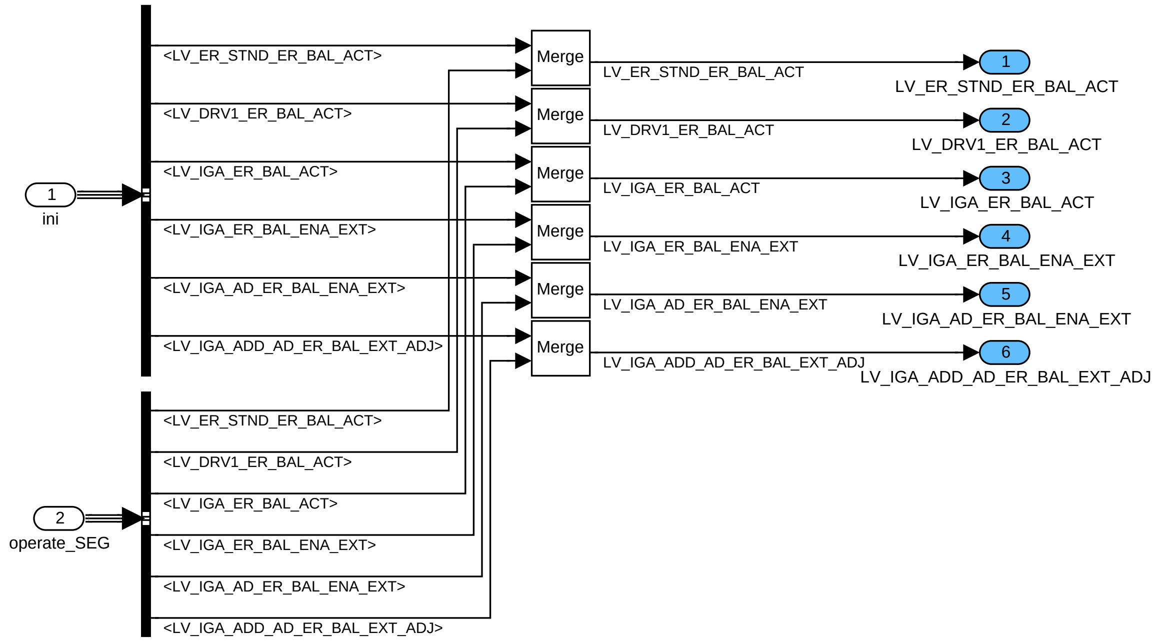


Figure 98 CYBL_M30907M01/ SIG_MNG

9.37 CYBL scheduler

Input data:

ER_STND_MMV_BAL[NC_CYL_NR]	ER_STND_MMV_STD_BAL[NC_CYL_NR]	ER_STD_MMV_BAL[NC_CYL_NR]	LC_ER_BAL_STOP_MAN
IGA_ADD_AD_ER_BAL[NC_CYL_NR]	LC_IGA_ADD_ER_BAL_RST_MAN	LC_IGA_ADD_AD_ER_BAL_RST_MAN	NC_CYL_NR

FUNCTION DESCRIPTION:

General information:

This manager specifies the sequencing of all CYBL tasks.

Description:

Within the cylinder balancing package (CYBL), all functions for Cylinder balancing via engine roughness" (CYBL_ER) are included. This scheduler has to manage the coordination of all involved parts.

Application conditions:

Recurrence: see formula section below

Activation: at every engine operating state

Deactivation: -

Formula section:

9.37.1 Non volatile memory tasks (CYBL_ER):

NVMY_STB	
NVMY_STB	Engine roughness signal preparation for Cylinder balancing (402U) /* ER_STND_MMV_BAL[x] (NVMY) = 0 /* ER_STND_MMV_STD_BAL[x] (NVMY) = 0 /* ER_STD_MMV_BAL[x] (NVMY) = 0
NVMY_STB	Cylinder balancing via IGA intervention (602G) /* IGA_ADD_AD_ER_BAL[x] (NVMY) = 0
NVMY_RST	
NVMY_RST	Engine roughness signal preparation for Cylinder balancing (402U) /* ER_STND_MMV_BAL[x] (NVMY) --> (RAM) /* ER_STND_MMV_STD_BAL[x] (NVMY) --> (RAM) /* ER_STD_MMV_BAL[x] (NVMY) --> (RAM)
NVMY_RST	Cylinder balancing via IGA intervention (602G) /* IGA_ADD_AD_ER_BAL[x] (NVMY) --> (RAM)
NVMY_UPD	
NVMY_UPD	Engine roughness signal preparation for Cylinder balancing (402U) /* ER_STND_MMV_BAL[x] (RAM) --> (NVMY) /* ER_STND_MMV_STD_BAL[x] (RAM) --> (NVMY) /* ER_STD_MMV_BAL[x] (RAM) --> (NVMY)
NVMY_UPD	Cylinder balancing via IGA intervention (602G) /* IGA_ADD_AD_ER_BAL[x] (RAM) --> (NVMY)

```
/* ***************************************************** Description *****************************************************
/* NVMY_STB:  -->  Initialization of NVMY cells at first engine run or NVMY error
/* NVMY_RST:  -->  Initialization of RAM cells with NVMY cells at RESET
/* NVMY_UPD:  -->  Initialization of NVMY cells with RAM cells at ECU power latch
/* [x] is representing [NC_CYL_NR] at the task definition above
/* [m] is representing [NC_CBK_EX_NR] at the task definition above
```

9.37.2 Initialization tasks (CYBL_ER):

RESET (no sequencing behind – only for info)	
RST	/* the RESET of aggregate ENTE has to be performed before CYBL RESET tasks
RST	Cylinder balancing manager (Appl. Inc.) (907M)
RST	Engine roughness signal preparation for Cylinder balancing (402U)
RST	Cylinder balancing manager (907L)
RST	Cylinder balancing via IGA intervention (602G)
STOP_MAN (no system event , section located within SW code- only for info)	
# IF (LC_ER_BAL_STOP_MAN = 1)	
STOP_MAN	Cylinder balancing manager (Appl. Inc.) (907M) /* Initialization at manual stop of function
STOP_MAN	Cylinder balancing manager (907L) /* Initialization at manual stop of function
# ENDIF	
RESET_MAN (no system event, section located within SW code – only for info)	
# IF (LC_IGA_ADD_ER_BAL_RST_MAN = 1)	
RST_MAN	Cylinder balancing via additive MFF intervention (7085) /* Manual reset of correction values
# ENDIF	
# IF (LC_IGA_ADD_AD_ER_BAL_RST_MAN = 1)	
RST_MAN	Cylinder balancing via IGA intervention (602G) /* Manual reset of adaptation correction values
# ENDIF	

```
/* ***************************************************** Description *****************************************************
/* RST:  -->  ECU RESET
/* RST_MAN:  -->  Initialization with use of a logical constant for setting of the wanted values
/* RST_DEAC:  --> Initialization at function deactivation
/* STOP_MAN:  --> Initialization at manual stop of function
/* EXIT_ST:  -->  Initialization at exit start
```

9.37.3 Recurring tasks(CYBL_ER):

SEG Task	
CYBL_ER	Cylinder balancing manager (Appl. Inc.) (907M)
CYBL_ER	Engine roughness signal preparation for Cylinder balancing (402U)
CYBL_ER	Cylinder balancing manager (907L)
CYBL_ER	Cylinder balancing via IGA intervention (602G)
100ms - Task	
CYBL_ER	Cylinder balancing via IGA intervention (602G)

```
/* ***************************************************** Description *****************************************************
```

9.37.4 Main interfering AGGR tasks for CYBL_ER:

AGGR Task before CYBL_ER	
AGGR_xx	
ENSD	Engine Position and Speed Determination
ENRD	Engine Roughness Determination
AGGR_xx	
CYBL_ER Task	
CYBL_ER	Cylinder balancing via engine roughness
AGGR – Task after CYBL_ER	
AGGR_xx	
INJR	Injection Realisation
IGRE	Ignition Realisation
AGGR_xx	
/* *** Description ***	

9.38 IVVT Configuration

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_NOT_ADJ_CAM_IVVT_IN_i	O/V	0...1H	0...1	1	-
Camshaft_IN_i: 0 = with adjustment, 1 = without adjustment					
LV_NOT_ADJ_CAM_IVVT_EX_i	O/V	0...1H	0...1	1	-
Camshaft_EX_i: 0 = with adjustment, 1 = without adjustment					

Input data:

CONF_CAM_VVT_EX			

FUNCTION DESCRIPTION:

General information:

An **i**nfinitely **v**ariable **v**alve **t**iming system (IVVT) is an optional part of the engine. It allows to phase the opening period of the valves. It works with the engine oil system. The complete functionality is included in the aggregate VVTI (**V**ariable **V**alve **Ti**ming). The determination of the phasing is provided by evaluations of the crankshaft and camshaft sensor signals in the aggregate ENSD (**En**gine **S**peed **D**etermination).

Description:

The first step of the IVVT functionality specification is the definition of the engine camshaft phasing configuration, i.e., how many actuators are used. A cylinder bank is a group of cylinders with their own inlet and exhaust camshaft. The number of cylinder banks is given by NC_NR_CBK_IVVT. The inlet and exhaust side can be adjusted on each other independently. The camshaft phasing strategy is given by NLC_IVVT_IN and NLC_IVVT_EX.

There can be 1 or 2 cylinder banks given by NC_NR_CBK_IVVT. The index _i indicates the cylinder bank in data. It is a placeholder for _1 or _1 and _2. The current configuration limits allow four independent actuators, two on the inlet and two on the exhaust side. There are variables describing the inlet and exhaust side and each camshaft individually. They are named …_IVVT_IN, …_IVVT_IN_i, …_IVVT_EX and …_IVVT_EX_i. The abbreviated notation is …_IVVT_IN(_EX) and …_IVVT_IN(_EX)_i. Also, the maximum configuration has variables …_IVVT_IN, …_IVVT_IN_1, …_IVVT_IN_2, …_IVVT_EX, …_IVVT_EX_1 and …_IVVT_EX_2, e.g., a V-engine. On the other side, a minimum configuration, e.g., an engine with one bank and one actuator on the inlet side, has only …_IVVT_IN and …_IVVT_IN_1. The other variables do not exist in this case. It is realized by compilation with NC_NR_CBK_IVVT, NLC_IVVT_IN and NLC_IVVT_EX as compiler switches. This rule is valid for data containing IVVTPWM or IVVTHPWM and ending with …_IN, …_IN_i, …_EX or …_EX_i, too.

There are some reasons for having redundant software, i.e., for more actuators as mounted, and tuneable data for the configuration of the actuators. A compilation creates variables of the type …_IVVT_IN, …_IVVT_IN_i, …_IVVT_EX and …_IVVT_EX_i as mentioned above. Variables for nonexisting actuators (redundant variables) are switched off by LC_NOT_ADJ_CAM_IVVT_IN_i for Inlet and LC_NOT_ADJ_CAM_IVVT_EX_i or CONF_CAM_VVT_EX for Exhaust side.

Application conditions:

Initialization:

At reset:

If LC_NOT_ADJ_CAM_IVVT_IN_i = 0
then LV_NOT_ADJ_CAM_IVVT_IN_i = 0
else LV_NOT_ADJ_CAM_IVVT_IN_i = 1

If LC_NOT_ADJ_CAM_IVVT_EX_i = 0
and CONF_CAM_VVT_EX = 1
then LV_NOT_ADJ_CAM_IVVT_EX_i = 0
else LV_NOT_ADJ_CAM_IVVT_EX_i = 1

This construction ensures that an unintentional change of LC_NOT_ADJ_CAM_IVVT_IN(_EX)_i by the application during IVVT performance will have no effects.

Configuration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
NC_NR_CBK_IVVT	1	1...2H	1...2	1	-
Number of camshaft cylinder banks					
NLC_IVVT_IN	1	0...1H	0...1	1	-
Inlet side configuration: 0 = without actuator, 1 = with actuator					
NLC_IVVT_EX	1	0...1H	0...1	1	-
Exhaust side configuration: 0 = without actuator, 1 = with actuator					

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_NOT_ADJ_CAM_IVVT_IN_i	1	0...1H	0...1	1	-
Camshaft_IN_i: 0 = with adjustment, 1 = without adjustment					
LC_NOT_ADJ_CAM_IVVT_EX_i	1	0...1H	0...1	1	-
Camshaft_EX_i: 0 = with adjustment, 1 = without adjustment					

9.39 IVVT Scheduler

FUNCTION DESCRIPTION:

General information:

The sequencing of individual IVVT tasks is specified in this scheduler. The names of chapters correspond to the individual tasks.

The tasks are divided into core ones and hook ones. The hook tasks are specified in hook modules. Each project can have different algorithms there.

9.39.1 IVVT Sequence of Tasks at Used Camshaft Edges

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_MASK_EDGE_CTL_IVVT_IN_i	-	0...1H	0...1	1	-
1 = current cam. edge corresponds to mask of edges used for controlling, 0 = does not correspond; inlet					
LV_MASK_EDGE_CTL_IVVT_EX_i	-	0...1H	0...1	1	-
1 = current cam. edge corresponds to mask of edges used for controlling, 0 = does not correspond; exhaust					
LV_MASK_EDGE_PSN_IVVT_IN_i	-	0...1H	0...1	1	-
1 = current cam. edge corresponds to mask of edges used for position, 0 = does not correspond; inlet					
LV_MASK_EDGE_PSN_IVVT_EX_i	-	0...1H	0...1	1	-
1 = current cam. edge corresponds to mask of edges used for position, 0 = does not correspond; exhaust					
T_DIF_EDGE_CAM_IVVT_IN_i	O/V	0...FFFFFFFFH	0...17179.87	4e-6	s
Time between previous masked edge and current masked one; inlet					
T_DIF_EDGE_CAM_IVVT_EX_i	O/V	0...FFFFFFFFH	0...17179.87	4e-6	s
Time between previous masked edge and current masked one; exhaust					
IDX_EDGE_CAM_IVVT_IN_i	V	0...1FH	1...32	1	-
Index of camshaft signal edge for IVVT task scheduling; inlet					
IDX_EDGE_CAM_IVVT_EX_i	V	0...1FH	1...32	1	-
Index of camshaft signal edge for IVVT task scheduling; inlet					
LV_CAM_REV_IVVT_IN_i	-	0...1H	0...1	1	-
0 = first camshaft revolution for IVVT task scheduling, 1 = second one; inlet					
LV_CAM_REV_IVVT_EX_i	-	0...1H	0...1	1	-
0 = first camshaft revolution for IVVT task scheduling, 1 = second one; exhaust					

Input data:

IDX_EDGE_CAM_IN_i	IDX_EDGE_CAM_EX_i	NC_NR_EDGE_CAM_IN
T_DIF_EDGE_CAM_IN_i	T_DIF_EDGE_CAM_EX_i	NC_NR_EDGE_CAM_EX
PSN_DIF_EDGE_CAM_IN_i	PSN_DIF_EDGE_CAM_EX_i	NC_NR_CBK_IVVT
STATE_MASK_EDGE_CAM		

FUNCTION DESCRIPTION:

Description:

A camshaft impulse wheel has two edges at least, generally NC_NR_EDGE_CAM_IN(_EX). It is possible to specify which edges will be used by IVVT over two camshaft revolutions. The specification is done by means of the array of bit fields C_IDX_MASK_EDGE_PSN(_CTL)_IVVT_IN(_EX)_i[3]. It is possible to distinguish the camshaft position tasks and the controlling tasks. There are three engine speed ranges that can have different camshaft edge masking. In order to avoid too frequent changes between the ranges at the engine speed threshold the hysteresis C_N_32_MASK_HYS_EDGE_CAM is used. STATE_MASK_EDGE_CAM expresses which engine speed range is currently in use. It is sufficient to update STATE_MASK_EDGE_CAM every 360 °CRK, see Chap. 9.39.3 "IVVT Sequence of 360 °CRK Linked Tasks".

Examples of some masks:

Hexadecimal to binary conversion:

Hexadecimal	Binary			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
A	1	0	1	0
B	1	0	1	1
C	1	1	0	0
D	1	1	0	1
E	1	1	1	0
F	1	1	1	1

The "half-moon-type" camshaft wheel using both edges would have:

C_IDX_MASK_EDGE_... = 0 0 0 0 0 0 0 F H
0000 0000 0000 0000 0000 0000 0000 1111 B

An eight-segment target wheel using every other edge would have:

C_IDX_MASK_EDGE_... = 0 0 0 0 5 5 5 5 H
0000 0000 0000 0000 0101 0101 0101 0101 B

Any camshaft wheel using one edge in two camshaft revolutions would have:

C_IDX_MASK_EDGE_... = 0 0 0 0 0 0 0 1 H
0000 0000 0000 0000 0000 0000 0000 0001 B

This is reasonable for the camshaft position control at very high engine speeds. The time between two used camshaft edges can be still satisfactory for good control quality and it saves runtime.

Further, this part of the scheduler specifies the sequence of calling the tasks at used camshaft edges.

Note to "IVVT Holding PWM": If the engine is not running the recurrence of this task is 10 ms.

The application has a possibility to check the calibration data for IVVT delay. The measurement of the actual delay is enabled if LC_ENA_DLY_MES_IVVT = 1.

Input data not used at the moment:
PSN_DIF_EDGE_CAM_IN(_EX)_i

Application conditions:

Initialization:

At transition "engine run" → "engine stop":

LV_MASK_EDGE_CTL_IVVT_IN(_EX)_i = 0

LV_MASK_EDGE_PSN_IVVT_IN(_EX)_i = 0

Recurrence: *At each camshaft edge*

Activation: *At every engine state*

Formula section:

Masking of camshaft edges and time between masked edges:

```
if      IDX_EDGE_CAM_IN(_EX)_i(n) <= IDX_EDGE_CAM_IN(_EX)_i(n-1)
then    LV_CAM_REV_IVVT_IN(_EX)_i =
        not(LV_CAM_REV_IVVT_IN(_EX)_i)
endif
        IDX_EDGE_CAM_IVVT_IN(_EX)_i =
        IDX_EDGE_CAM_IN(_EX)_i +
        LV_CAM_REV_IVVT_IN(_EX)_i * NC_NR_EDGE_CAM_IN(_EX)
        Edges for position:
        LV_MASK_EDGE_PSN_IVVT_IN(_EX)_i =
        IDX_EDGE_CAM_IVVT_IN(_EX)_i-th bit of
        C_IDX_MASK_EDGE_PSN_IVVT_IN(_EX)_i[STATE_MASK_EDGE_CAM]
        Edges for controlling:
        LV_MASK_EDGE_CTL_IVVT_IN(_EX)_i =
        IDX_EDGE_CAM_IVVT_IN(_EX)_i-th bit of
        C_IDX_MASK_EDGE_CTL_IVVT_IN(_EX)_i[STATE_MASK_EDGE_CAM]
        if      LV_MASK_EDGE_CTL_IVVT_IN(_EX)_i(n-1) = 1
        then    T_DIF_EDGE_CAM_IVVT_IN(_EX)_i(n) = T_DIF_EDGE_CAM_IN(_EX)_i
        else    T_DIF_EDGE_CAM_IVVT_IN(_EX)_i(n) =
                T_DIF_EDGE_CAM_IVVT_IN(_EX)_i(n-1) +
                T_DIF_EDGE_CAM_IN(_EX)_i
        endif
```

Sequence of tasks at used camshaft edges:

```
if      LV_MASK_EDGE_PSN_IVVT_IN(_EX)_i = 1
then    Core: "IVVT Camshaft Position"
        Hook: "IVVT Lock at Transition to Engine Stop"
        Core: "IVVT Bank Mean Camshaft Position and Valve Overlap"
        # NC_NR_CBK_IVVT = 2:
        Core: "IVVT Camshaft Position for Balancing"
        # end
```

endif

if LV_MASK_EDGE_CTL_IVVT_IN(_EX)_i = 1

then *# NC_NR_CBK_IVVT = 2:*

Core: "IVVT Individual Setpoint Selection"

Hook: "IVVT Faulty Actuator Emulation"

Core: "IVVT Setpoint Work-up before Controlling"

 # end

Core: "IVVT Controller"

Core: "IVVT System at Maximum Adjustment Stop Position"

Core: "IVVT Deviation Adaptation of Holding PWM"

Core: "IVVT Drift Adaptation of Holding PWM"

Core: "IVVT Holding PWM Adaptation Manager"

Core: "IVVT Holding PWM"

Hook: "IVVT Camshaft Position Deviation Diagnosis - Application Incidences"

Core: "IVVT Camshaft Position Deviation Diagnosis"

Core: "IVVT Camshaft Position Partial Deviation Diagnosis"

if LC_ENA_DLY_MES_IVVT = 1

then Core: "IVVT Measurement of Delay"

endif

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_IDX_MASK_EDGE_CTL_IVVT_IN_i[3]	1	0...FFFFFFFFH	0...4294967295	1	-
Bit field for masking cam. edges used for controlling (bit=1 → used edge), least signif. bit has index 1; inlet					
C_IDX_MASK_EDGE_CTL_IVVT_EX_i[3]	1	0...FFFFFFFFH	0...4294967295	1	-
Bit field for masking cam. edges used for controlling (bit=1 → used edge), least signif. bit has index 1; exhaust					
C_IDX_MASK_EDGE_PSN_IVVT_IN_i[3]	1	0...FFFFFFFFH	0...4294967295	1	-
Bit field for masking cam. edges used for position (bit=1 → used edge), least signif. bit has index 1; inlet					
C_IDX_MASK_EDGE_PSN_IVVT_EX_i[3]	1	0...FFFFFFFFH	0...4294967295	1	-
Bit field for masking cam. edges used for position (bit=1 → used edge), least signif. bit has index 1; exhaust					
LC_ENA_DLY_MES_IVVT	1	0...1H	0...1	1	-
1 = mesurement of IVVT delay enabled, 0 = disabled					

9.39.2 IVVT Sequence of Camshaft Edge Tasks at Camshaft Sensor Error

Input data:

LV_ERR_CAM_TOT	LV_ERR_LIH_IVVT		

FUNCTION DESCRIPTION:

Description:

In case of a camshaft sensor error no proper performance of the IVVT system is possible. The tasks in Chap. 9.39.1 "IVVT Sequence of Tasks at Used Camshaft Edges" have then no triggering event. Therefore, when an error is recognized some tasks are called up every 360 °CRK in order to ensure proper limp home operation.

Application conditions:

Recurrence: 360 °CRK

Activation: LV_ERR_LIH_IVVT = 1

and LV_ERR_CAM_TOT = 1

Formula section:

Sequence of camshaft edge tasks at camshaft sensor error:

Core: "IVVT Camshaft Position"

Core: "IVVT Controller"

Core: "IVVT Holding PWM"

9.39.3 IVVT Sequence of 360 °CRK Linked Tasks

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
STATE_MASK_EDGE_CAM	V	0...2H	0...2	1	-
Current mask of edges					
CTR_REV_SDL_IVVT	O/V	0...7H	0...7	1	-
Engine revolution counter for triggering IVVT 360 °CRK linked tasks					
LV_TRIG_REV_SDL_IVVT	-	0...1H	0...1	1	-
1 = trigger IVVT 360 °CRK linked tasks, 0 = no trigger					

Input data:

NC_NR_CBK_IVVT	N_32		

FUNCTION DESCRIPTION:

Description:

STATE_MASK_EDGE_CAM for the camshaft edge masking, see Chap. 9.39.1 "IVVT Sequence of Tasks at Used Camshaft Edges", is updated every 360 °CRK.

A 360 °CRK linked task has its reccurence based on 360 °CRK. It means that it can be one engine revolution or two or three, etc. The reccurence of some tasks can be tuned in four engine speed ranges by means of ID_NR_REV_SDL_IVVT. This feature is dedicated for the run time optimization.

Further, this part of the scheduler specifies the sequence of calling the 360 °CRK linked tasks.

If there is a failure in the crankshaft signal acquisition a substitute signal manages the 360 °CRK reccurence.

Application conditions:

Initialization:	At reset:
	STATE_MASK_EDGE_CAM = 0
	At transition "engine run" → "engine stop":
	STATE_MASK_EDGE_CAM = 0
	CTR_REV_SDL_IVVT = 0
	LV_TRIG_REV_SDL_IVVT = 0
Recurrence:	360 °CRK
Activation:	At every engine state

Formula section:

Current mask of edges:

Transition from STATE_MASK_EDGE_CAM = 0:

```
if      N_32 > C_N_32_MASK_1_EDGE_CAM + C_N_32_MASK_HYS_EDGE_CAM
then    STATE_MASK_EDGE_CAM = 1
endif
```

Transition from STATE_MASK_EDGE_CAM = 1:

```
if      N_32 <= C_N_32_MASK_1_EDGE_CAM
then    STATE_MASK_EDGE_CAM = 0
elseif  N_32 > C_N_32_MASK_2_EDGE_CAM + C_N_32_MASK_HYS_EDGE_CAM
then    STATE_MASK_EDGE_CAM = 2
endif
```

Transition from STATE_MASK_EDGE_CAM = 2:

```
if      N_32 <= C_N_32_MASK_2_EDGE_CAM
then    STATE_MASK_EDGE_CAM = 1
```

```
endif
```

Counting revolutions:

```
if      LV_TRIG_REV_SDL_IVVT = 1
then    CTR_REV_SDL_IVVT = 1
else    CTR_REV_SDL_IVVT = CTR_REV_SDL_IVVT + 1
endif
if      CTR_REV_SDL_IVVT >= ID_NR_REV_SDL_IVVT
then    LV_TRIG_REV_SDL_IVVT = 1
else    LV_TRIG_REV_SDL_IVVT = 0
endif
```

Sequence of 360 °CRK Linked Tasks:

```
Core: "IVVT State"
# NC_NR_CBK_IVVT = 2:
        Core: "IVVT Individual Error Check for Bank Balancing"
# end
Core: "IVVT Individual Camshaft Activation"
Core: "IVVT Reference Position Adaptation Manager"
if      LV_TRIG_REV_SDL_IVVT = 1
then    Core: "IVVT Setpoint Premanager"
        Hook: "IVVT Setpoint for Homogeneous Stoichiometric"
        Hook: "IVVT Setpoint for Homogeneous Lean"
        Hook: "IVVT Setpoint for Stratified"
        Core: "IVVT Setpoint Postmanager"
        Core: "IVVT Inlet Adjustment Prioritization to Exhaust One"
        # NC_NR_CBK_IVVT = 1:
        Core: "IVVT Individual Setpoint Selection"
        Hook: "IVVT Faulty Actuator Emulation"
        Core: "IVVT Setpoint Work-up before Controlling"
        # end
        Core: "IVVT Delay"
endif

        # NC_NR_CBK_IVVT = 2:
        Core: "IVVT Adjustment Velocity for Bank Balancing"
        Core: "IVVT Bank Balancing"
        # end
```

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_N_32_MASK_1_EDGE_CAM	1	0...FFH	0...8160	32	rpm
Engine speed treshold 1 for changing mask of used camshaft edges					
C_N_32_MASK_2_EDGE_CAM	1	0...FFH	0...8160	32	rpm
Engine speed treshold 2 for changing mask of used camshaft edges					
C_N_32_MASK_HYS_EDGE_CAM	1	0...FFH	0...8160	32	rpm
Engine speed hysteresis for avoiding too frequent changes between masks of used camshaft edges at engine speed threshold					
ID_NR_REV_SDL_IVVT	4	1...7H	1...7	1	-
LDP_N_32_ID_NR_REV_SDL_IVVT	4	0...FFH	0...8160	32	rpm
Number of engine revolutions between trigerring IVVT 360 °CRK linked tasks					

9.39.4 IVVT Sequence of 1 s Tasks

FUNCTION DESCRIPTION:

Description:

This part of the scheduler specifies the sequence of calling the 1 s tasks.

Application conditions:

Recurrence: 1 s

Activation: At every engine state

Formula section:

Sequence of 1 s tasks:

Hook: "IVVT Factors for AFS Setpoint"

Core: "IVVT Coil Substitute Temperature"

9.39.5 IVVT Sequence of 100 ms Tasks

FUNCTION DESCRIPTION:

Description:

This part of the scheduler specifies the sequence of calling the 100 ms tasks.

Application conditions:

Recurrence: 100 ms

Activation: At every engine state

Formula section:

Sequence of 100 ms tasks:

Hook: "IVVT Release"

Core: "IVVT PWM Frequencies"

Core: "IVVT PWM Adjustment Levels"

Hook: "IVVT Short Trip - Application Incidences"

Hook: "IVVT Substitute PWM - General"

Hook: "IVVT Substitute PWM"

Hook: "IVVT Output Signal Diagnosis - Application Incidences"

Core: "IVVT Output Signal Diagnosis"

Hook: "IVVT Crankshaft to Camshaft Mechanics Violation Diagnosis - Application Incidences"

Core: "IVVT Crankshaft to Camshaft Mechanics Violation Diagnosis"

Core: "IVVT Short Trip"

Hook: "IVVT Camshaft Position Deviation Diagnosis - Rate Based Monitoring"

Hook: "IVVT External Failure"

Core: "IVVT Limp Home Manager"

Hook: "IVVT Disabling Reference Position Adaptation"

Hook: "IVVT Disabling Oil Control Valve Cleaning"

9.39.6 IVVT Sequence of 20 ms Tasks

FUNCTION DESCRIPTION:

Description:

This part of the scheduler specifies the sequence of calling the 20 ms tasks.

Application conditions:

Recurrence: *20 ms*

Activation: At every engine state

Formula section:

Sequence of 20 ms tasks:

Hook: "IVVT Ignition Angle Correction"

9.39.7 IVVT Sequence of 10 ms Tasks

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
T_DLY_MES_IVVT	O/V	0...FFFFH	0...655350	10	ms
10 ms timer for delay measurement					

Input data:

LV_ES			

FUNCTION DESCRIPTION:

Description:

This part of the scheduler specifies the sequence of calling the 10 ms tasks.

The 10 ms timer T_DLY_MES_IVVT is used for measurement of delays in IVVT. It means that differences between values of this timer are calculated. Therefore, the timer must manage the recurring overflow.

Application conditions:

Initialization:

At transition "Engine run" → "Engine stop":

T_DLY_MES_IVVT = 0

Recurrence: 10 ms

Activation: At every engine state

Formula section:

Sequence of 10 ms tasks:

Update (increase) T_DLY_MES_IVVT

Core: "IVVT Oil Control Solenoid Valve Cleaning"

if LV_ES = 1

then Core: "IVVT Holding PWM"

endif

9.40 IVVT State

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ACT_IVVT	O/V	0...1H	0...1	1	-
1 = IVVT active (with control), 0 = inactive (target: passive stop position)					
STATE_IVVT	O/V	0H 1H 2H 3H 4H	PASSIVE READY ACT_ADAPT ENABLE LIMP_HOME	1	-
IVVT state					
LV_AD_REQ_IVVT	O/V	0...1H	0...1	1	-
1 = request for activation adaptation of reference position, 0 = no request					

Input data:

N_32	TOIL	VB	LV_ST_END
LV_ERR_LIH_IVVT	LV_AD_END_IVVT		

FUNCTION DESCRIPTION:

General information:

The IVVT system can be in active or inactive state. The active state means that the camshaft position is controlled to a desired position (setpoint). It is the normal prevailing state. The state is inactive in certain circumstances which do not allow controlling, or it is better not to control. Then, the passive stop position is the target one.

Description:

There are different reasons for getting the IVVT system into the inactive state. The states are depicted by means of STATE_IVVT.

IVVT states:

STATE_IVVT	Abbr.	Description
PASSIVE	PA	inactive: after engine start or out of battery voltage range or oil temperature below minimum limit
READY	RE	inactive: oil temperature above maximum threshold (depending on engine speed)
ACT_ADAPT	AD	inactive: activation adaptation of reference position after "PASSIVE" state, i.e., successful adaptation is required for IVVT activation
ENABLE	EN	active: controlling
LIMP_HOME	LH	inactive or partially active: a failure does not allow proper IVVT performance

The engine oil system needs some time after the engine start to build up oil pressure necessary for IVVT performance. To ensure this, the IVVT system remains in the inactive state, "PASSIVE", for the time ID_DLY_ST_IVVT which depends on the oil temperature.

Further, the reference position adaptation is required, "ACT_ADAPT". The adaptation is enabled if the conditions for IVVT performance are fulfilled, i.e., the battery voltage and the oil temperature (C_VB_MIN_IVVT, C_VB_MAX_IVVT, C_TOIL_MIN_IVVT). The state remains inactive till the adaptation is finished successfully. Then, the IVVT states becomes active, "ENABLE".

The conditions for IVVT performance are checked continiously. If only one of the thresholds C_VB_MIN_IVVT, C_VB_MAX_IVVT or C_TOIL_MIN_IVVT is crossed IVVT is switched into the inactive state, "PASSIVE". If later these conditions are fulfilled again a new activation adaptation of the reference position is required, "ACT_ADAPT".

If the oil temperature rises over IP_TOIL_MAX_IVVT the IVVT system is inactivated, "READY". If the oil temperature returns into the allowed range IVVT becomes active, "ENABLE", if the activation adaptation has been already finished successfully, or remains inactive, "ACT_ADAPT", if the adaptation has been interrupted due to the high temperature.

The inactivation due to C_VB_MIN_IVVT, C_VB_MAX_IVVT or C_TOIL_MIN_IVVT has a higher priority than the one due to IP_TOIL_MAX_IVVT.

There are the hystereses C_VB_HYS_IVVT and C_TOIL_HYS_IVVT for avoiding too frequent transitions between the active and inactive state at the battery voltage and oil temperature thresholds.

The inactivation due to a failure, "LIMP_HOME", is held on till the transition "Engine run" → "Engine stop". Then, the activation starts from "PASSIVE" as usual. The transition from "LIMP_HOME" to "PASSIVE" during the engine operation is possible if LV_ERR_LIH_IVVT becomes 0 and C_DLY_LIH_IVVT > 0. C_DLY_LIH_IVVT delays this transition.

The figure "IVVT state diagram" illustrates transitions between the individuall IVVT states.

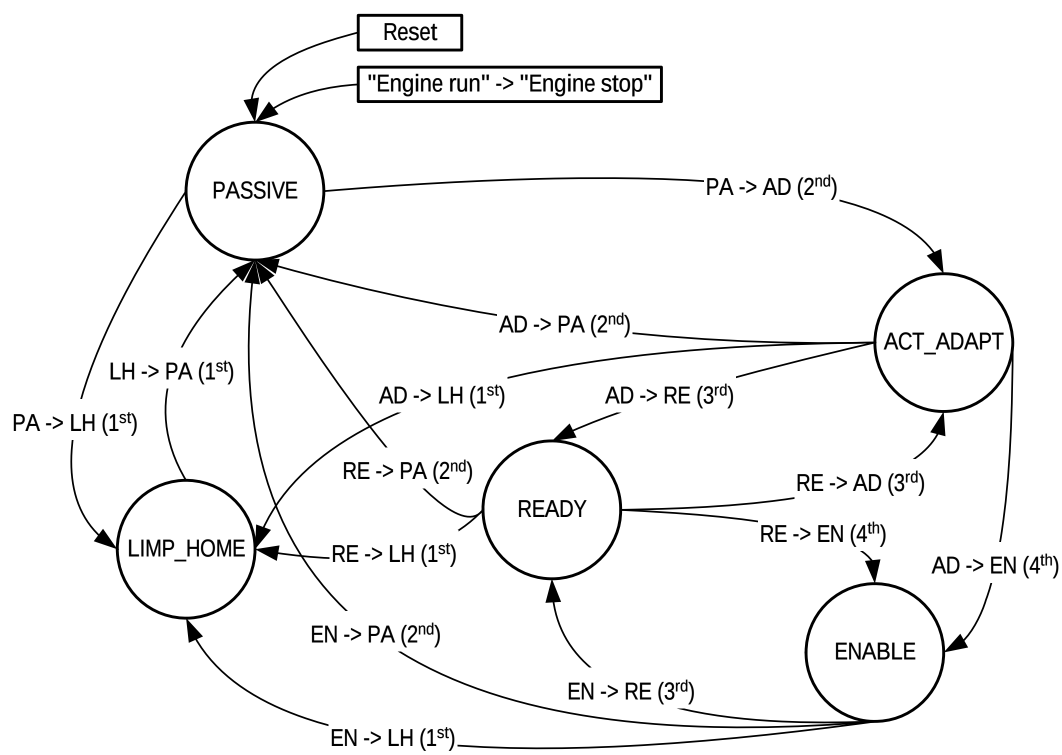


Figure: "IVVT state diagram."

Application conditions:

Initialization:

At reset:

STATE_IVVT = "PASSIVE"

At transition "Engine run" → "Engine stop":

LV_ACT_IVVT = 0

STATE_IVVT = "PASSIVE"

LV_AD_REQ_IVVT = 0

At transition LV_ST_END = 0 → 1:

Start Timer ID_DLY_ST_END_IVVT

Recurrence:

360 °CRK

Activation:

At every engine state

Formula section:

For transition identifiers see figure "IVVT state diagram". The number in brackets at the transition identifier means the order in which the corresponding transition condition is queried in a state.

Transition conditions:

The current state does not change if none of the transition conditions is fulfilled.

"RE → PA", "AD → PA", "EN → PA":

if VB < C_VB_MIN_IVVT
or VB > C_VB_MAX_IVVT
or TOIL < C_TOIL_MIN_IVVT

"PA → AD":

if VB >= C_VB_MIN_IVVT + C_VB_HYS_IVVT
and VB <= C_VB_MAX_IVVT - C_VB_HYS_IVVT
and TOIL >= C_TOIL_MIN_IVVT + C_TOIL_HYS_IVVT
and Timer_ID_DLY_ST_END_IVVT expired

"RE → AD":

if TOIL =< IP_TOIL_MAX_IVVT - C_TOIL_HYS_IVVT
and LV_AD_END_IVVT = 0

"AD → RE", "EN → RE":

if TOIL > IP_TOIL_MAX_IVVT

"AD → EN":

if LV_AD_END_IVVT = 1

"RE → EN":

if TOIL =< IP_TOIL_MAX_IVVT - C_TOIL_HYS_IVVT

and LV_AD_END_IVVT = 1

"PA → LH", "RE → LH", "AD → LH", "EN → LH":

if LV_ERR_LIH_IVVT = 1

"LH → PA":

if C_DLY_LIH_IVVT > 0

and LV_ERR_LIH_IVVT = 0

and Timer C_DLY_LIH_IVVT expired

Transition actions:

"PA → AD":

LV_AD_REQ_IVVT = 1

"AD → EN":

LV_ACT_IVVT = 1

LV_AD_REQ_IVVT = 0

"EN → PA":

LV_ACT_IVVT = 0

"EN → RE":

LV_ACT_IVVT = 0

"RE → EN":

LV_ACT_IVVT = 1

"AD → RE":

LV_AD_REQ_IVVT = 0

"RE → AD":

LV_AD_REQ_IVVT = 1

"PA → LH", "RE → LH", "AD → LH", "EN → LH":

LV_ACT_IVVT = 0

LV_AD_REQ_IVVT = 0

Actions inside states:

"LIMP_HOME":

At transition LV_ERR_LIH_IVVT = 1 → 0:

Start Timer C_DLY_LIH_IVVT

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
ID_DLY_ST_END_IVVT	1x6	1...FFH	0.1...25.5	0.1	s
LDP_TOIL_ID_DLY_ST_END_IVVT	6	0...C8H	-40...160	1	°C
Time to wait after engine start till oil pressure suffices for IVVT operation					
C_TOIL_MIN_IVVT	1	0...C8H	-40...160	1	°C
Minimum oil temperature required for IVVT operation					
IP_TOIL_MAX_IVVT	1x6	0...C8H	-40...160	1	°C
LDP_N_32_IP_TOIL_MAX_IVVT	6	0...FFH	0...8160	32	rpm
Maximum oil temperature allowed for IVVT operation					
C_TOIL_HYS_IVVT	1	0...C8H	0...200	1	°C
Oil temperature hysteresis for avoiding too frequent transitions between IVVT states at limits					
C_VB_MIN_IVVT	1	0...FFH	0...26	0.102	V
Minimum battery voltage required for IVVT operation					
C_VB_MAX_IVVT	1	0...FFH	0...26	0.102	V
Maximum battery voltage allowed for IVVT operation					
C_VB_HYS_IVVT	1	0...FFH	0...26	0.102	V
Battery voltage hysteresis for avoiding too frequent transitions between IVVT states at limits					
C_DLY_LIH_IVVT	1	0...FFH	0...25.5	0.1	s
Wating delay before transition LH->PA after disappearing conditions for limp home					

9.41 IVVT Camshaft Position

FUNCTION DESCRIPTION:

General information:

There are several possibilities for defining a position of the camshaft. The chosen one is described bellow.

Description:

Camshaft phasing design data are illustrated in Figure "Camshaft position". The maximum adjustment range is given by C_CAM_ADJ_RNG_MAX_IVVT_IN(_EX). The next point is the adjustment direction. Actuators working with the engine oil system and controlled by a magnetic valve have a passive stop position, or reference position. It means that the IVVT system without energization moves towards, or remains in the passive stop position. The IVVT system configuration is: the passive stop position of the exhaust camshaft is advanced, of the inlet camshaft retard.

The camshaft position is the crank angle coordinate of the top point of the cam. The crank angle axis has its zero point in the top-center crank position of the gas exchange phase.

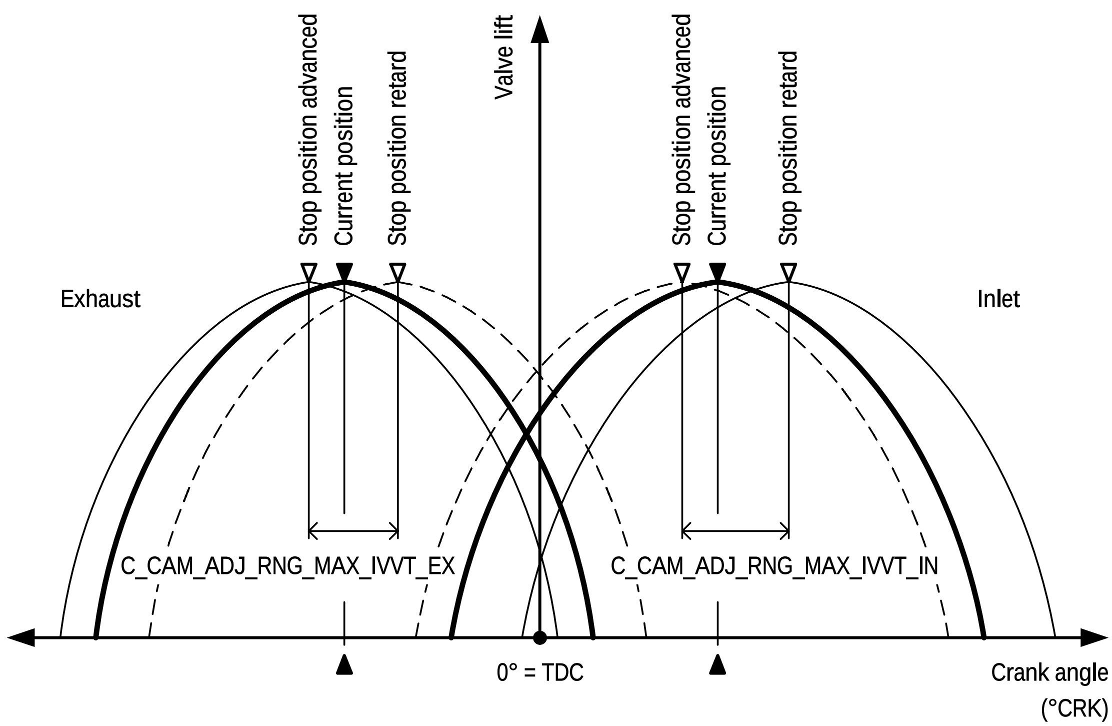


Figure: "Camshaft position."

The position mentioned above represents the design situation. Each component has certain tolerances. Further, the camshaft position is determined from the camshaft and crankshaft sensor signals. There are inaccuracies at the measurement. Therefore, the obtained reference reference position of the camshaft differs from the designed one, i.e., C_CAM_INI_IN(_EX).

This diference is detected by an adaptation of the reference positions. It is obvious that the camshaft must be in its passive stop position. The difference between the design reference position and the current one is PSN_AD_CAM_IN(_EX)[i]. The sign of this quantity is clear from the formula used: difference equals obtained position minus design position.

Information delivered by ENSD during engine operation is the difference between the adapted reference position and current position of a camshaft (difference equals current position minus adapted reference position), see PSN_CAM_IN(_EX)[i] in Figure "Camshaf position in IVVT".

The adapted reference positions define IVVT coordinate systems. It means the adapted reference position of the inlet camshaft is the point C_CAM_INI_IN of the inlet coordinate system. The same applies for the exhaust side with C_CAM_INI_EX. The camshaft positions used in IVVT are CAM_AV_IVVT_IN[i] and CAM_AV_IVVT_EX[i].

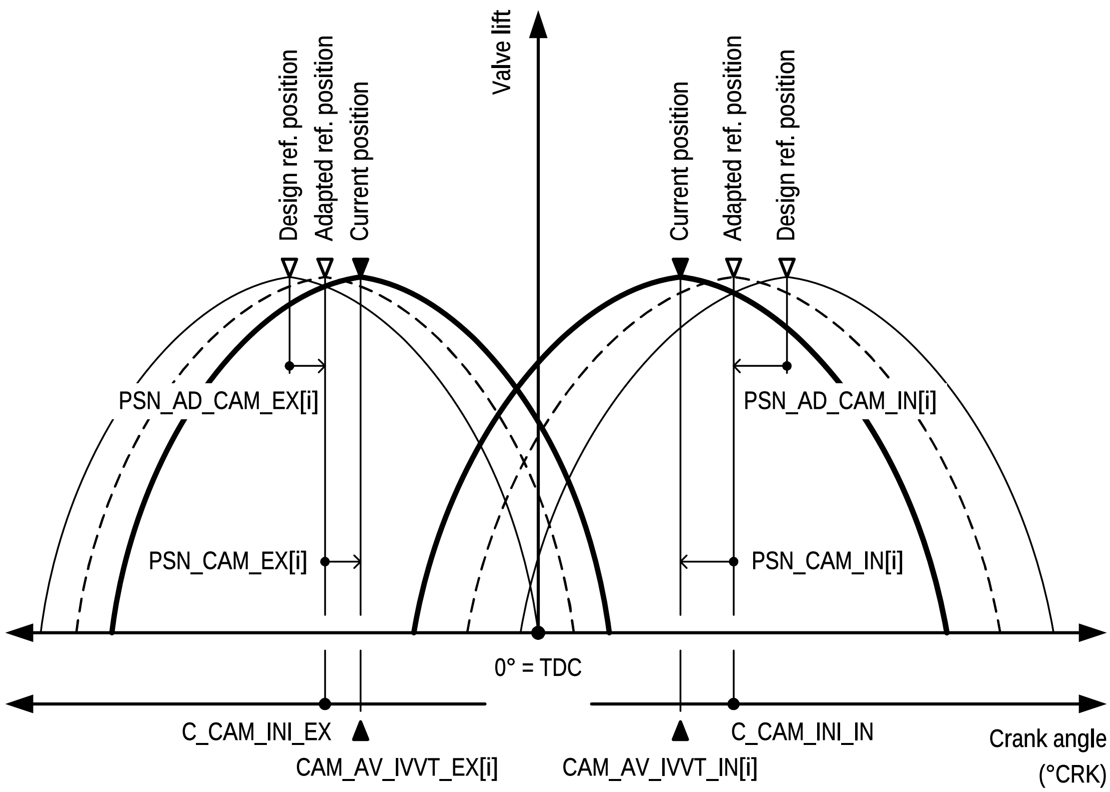


Figure: "Camshaft position in IVVT."

9.41.1 Stop Positions

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_ADJ_MAX_IVVT_IN	V/O	0...FFH	60...155.625	0.375	[°CRK]
Maximum adjustement stop position of inlet camshaft					
CAM_ADJ_MAX_IVVT_EX	V/O	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Maximum adjustement stop position of exhaust camshaft					

Input data:

CONF_CAM_VVT_EX			

FUNCTION DESCRIPTION:

Application conditions:

Initialization:

At reset:

CAM_ADJ_MAX_IVVT_IN = C_CAM_INI_IN - C_CAM_ADJ_RNG_MAX_IVVT_IN

if CONF_CAM_VVT_EX = 1

then CAM_ADJ_MAX_IVVT_EX = C_CAM_INI_EX + C_CAM_ADJ_RNG_MAX_IVVT_EX

else CAM_ADJ_MAX_IVVT_EX = C_CAM_INI_EX

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_INI_IN	1	0...FFH	60...155.625	0.375	[°CRK]
Passive stop position (reference position) of inlet camshaft					
C_CAM_INI_EX	1	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Passive stop position (reference position) of exhaust camshaft					
C_CAM_ADJ_RNG_MAX_IVVT_IN	1	0...A0H	0...60	0.375	[°CRK]
Maximum adjustment range of inlet camshaft actuator					
C_CAM_ADJ_RNG_MAX_IVVT_EX	1	0...A0H	0...60	0.375	[°CRK]
Maximum adjustment range of exhaust camshaft actuator					

9.41.2 Current Position

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_AV_IVVT_IN[NC_NR_CBK_IVVT]	V/O	0...FFH	60...155.625	0.375	[°CRK]
Current position of inlet camshaft for IVVT controlling					
CAM_AV_IVVT_EX[NC_NR_CBK_IVVT]	V/O	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Current position of exhaust camshaft for IVVT controlling					
CAM_IN[NC_NR_CBK_IVVT]	V/O	0...FFH	60...155.625	0.375	[°CRK]
Current position of inlet camshaft					
CAM_EX[NC_NR_CBK_IVVT]	V/O	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Current position of exhaust camshaft					
CAM_FIL_NOT_IVVT_IN[NC_NR_CBK_IVVT]	V	0...FFH	60...155.625	0.375	[°CRK]
Inlet camshaft position not filtered					
CAM_FIL_NOT_IVVT_EX[NC_NR_CBK_IVVT]	V	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Exhaust camshaft position not filtered					
CAM_CHG_IVVT_IN[NC_NR_CBK_IVVT]	V	8000...7FFFH	-96...95.99707	2.9297e-3	[°CRK]
Change of inlet camshaft position					
CAM_CHG_IVVT_EX[NC_NR_CBK_IVVT]	V	8000...7FFFH	-96...95.99707	2.9297e-3	[°CRK]
Change of exhaust camshaft position					
CAM_OSC_AMPL_IVVT_IN[NC_NR_CBK_IVVT]	V	0...7FFFH	0...95.99707	2.9297e-3	[°CRK]
Amplitude of inlet camshaft position change					
CAM_OSC_AMPL_IVVT_EX[NC_NR_CBK_IVVT]	V	0...7FFFH	0...95.99707	2.9297e-3	[°CRK]
Amplitude of exhaust camshaft position change					
CRLC_CAM_FIL_IVVT_IN[NC_NR_CBK_IVVT]	V	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation constant for inlet camshaft position filtering					
CRLC_CAM_FIL_IVVT_EX[NC_NR_CBK_IVVT]	V	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation constant for exhaust camshaft position filtering					
CAM_FIL_IVVT_IN[NC_NR_CBK_IVVT]	V	0...FFFFH	60...155.99853	1.4648e-3	[°CRK]
Filtered inlet camshaft position					
CAM_FIL_IVVT_EX[NC_NR_CBK_IVVT]	V	0...FFFFH	-40.125...-136.12353	-1.4648e-3	[°CRK]
Filtered exhaust camshaft position					

Input data:

LV_ERR_CRK	LV_ERR_CAM_TOT	NLC_IVVT_IN	NLC_IVVT_EX
PSN_AD_CAM_IN[NC_NR_CAM_CBK]	PSN_AD_CAM_EX[NC_NR_CAM_CBK]	PSN_CAM_IN[NC_NR_CAM_CBK]	PSN_CAM_EX[NC_NR_CAM_CBK]
LV_VLD_PSN_CAM_IN[NC_NR_CAM_CBK]	LV_VLD_PSN_CAM_EX[NC_NR_CAM_CBK]		
NC_NR_CBK_IVVT	N_32		

FUNCTION DESCRIPTION:

Description:

The camshaft position determination is disturbed if there are failures at the crankschaft and/or camshaft signal acquisition. If the position cannot be determined C_CAM_INI_IN(_EX) is taken as a substitute value. If there are only sporadic disturbances the previous value is taken.

The camshaft position can oscillate due to changing torques on the camshaft. A mean value is needed for the IVVT position control. The oscillation can disturb the control especially at reaching a steady setpoint.

The mean camshaft position change CAM_CHG_IVVT_IN(_EX)[i] and the amplitude of the changes CAM_OSC_AMPL_IVVT_IN(_EX)[i] is evaluated.

The camshaft position is filtered dependent on CAM_OSC_AMPL_IVVT_IN(_EX)[i] and CAM_CHG_IVVT_IN(_EX)[i], ID_CRLC_CAM_IVVT. Following limit cases can occur:

1. The mean change is zero and the amplitude is non-zero. It means that the camshaft keeps its mean position and oscillates. A strong filtering is to apply.
2. The mean change equals the amplitude. It means that the camshaft changes its mean position without oscillation. A weak filtering is to apply.

An example:

- The mean change equals half of the amplitude. It means that the camshaft changes its mean position as well as oscillate. A moderate filtering is to apply.

If the mean camshaft movement changes its direction the case 1 occurs for short time. It means a strong filering that is not required in such case. Therefore, the descent of the correlation constant is limited by ID_CRLC_CHG_NEG_LIM_CAM_IVVT. Also, the aim to follow the mean change is prioritised to the filtering of the oscillations.

Input data not used at the moment:
PSN_AD_CAM_IN(_EX)[i]

Application conditions:

Initialization:

At reset:

CAM_AV_IVVT_IN(_EX)[i] = C_CAM_INI_IN(_EX)

CAM_FIL_NOT_IVVT_IN(_EX)[i] = C_CAM_INI_IN(_EX)

CAM_FIL_IVVT_IN(_EX)[i] = C_CAM_INI_IN(_EX)

CRLC_CAM_FIL_IVVT_IN(_EX)[i] = 0.99609

 # NC_NR_CBK_IVVT = 1:

CAM_IN[1] = C_CAM_INI_IN

CAM_EX[1] = C_CAM_INI_EX

 # NC_NR_CBK_IVVT = 2:

CAM_IN[1] = CAM_IN[2] = C_CAM_INI_IN

CAM_EX[1] = CAM_EX[2] = C_CAM_INI_EX

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

At every engine state

Formula section:

Camshaft position in IVVT coordination system:

if LV_VLD_PSN_CAM_IN(_EX)[i] = 1
then *Camshaft position can be determined*
CAM_FIL_NOT_IVVT_IN(_EX)[i] = C_CAM_INI_IN(_EX) + PSN_CAM_IN(_EX)[i]
else
if LV_ERR_CRK = 0
and LV_ERR_CAM_TOT = 0
then *No permanent sensor errors, information not valid sporadically*
CAM_FIL_NOT_IVVT_IN(_EX)[i] = CAM_AV_IVVT_IN(_EX)[i](n-1)
else *Camshaft positon cannot be determined, substitute value*
CAM_FIL_NOT_IVVT_IN(_EX)[i] = C_CAM_INI_IN(_EX)
endif

Camshaft position filtering:

CAM_OSC_AMPL_IVVT_IN(_EX)[i](n) =
CAM_OSC_AMPL_IVVT_IN(_EX)[i](n-1) + ID_CRLC_CAM_DIF_IVVT *
(abs(CAM_FIL_NOT_IVVT_IN(_EX)[i](n) - CAM_FIL_NOT_IVVT_IN(_EX)[i](n-1)) -
CAM_OSC_AMPL_IVVT_IN(_EX)[i](n-1))
CAM_CHG_IVVT_IN(_EX)[i](n) =
CAM_CHG_IVVT_IN(_EX)[i](n-1) + ID_CRLC_CAM_DIF_IVVT *
((CAM_FIL_NOT_IVVT_IN(_EX)[i](n) - CAM_FIL_NOT_IVVT_IN(_EX)[i](n-1)) -
CAM_CHG_IVVT_IN(_EX)[i](n-1))
Use abs(CAM_CHG_IVVT_IN(_EX)[i]) for list of data points for ID_CRLC_CAM_IVVT
if CRLC_CAM_FIL_IVVT_IN(_EX)[i](n-1) - ID_CRLC_CAM_IVVT >
ID_CRLC_CHG_NEG_LIM_CAM_IVVT
then *Limit too quick negative change of correlation constant*
CRLC_CAM_FIL_IVVT_IN(_EX)[i] =
CRLC_CAM_FIL_IVVT_IN(_EX)[i](n-1) -
ID_CRLC_CHG_NEG_LIM_CAM_IVVT
else *No limitation of correlation constant necessary*
CRLC_CAM_FIL_IVVT_IN(_EX)[i] = ID_CRLC_CAM_IVVT
endif
CAM_FIL_IVVT_IN(_EX)[i](n) = CAM_FIL_IVVT_IN(_EX)[i](n-1) +
CRLC_CAM_FIL_IVVT_IN(_EX)[i] *
(CAM_FIL_NOT_IVVT_IN(_EX)[i](n) - CAM_FIL_IVVT_IN(_EX)[i](n-1))
CAM_AV_IVVT_IN(_EX)[i] = CAM_FIL_IVVT_IN(_EX)[i](n)

Camshaft position as engine quantity:

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:

CAM_IN[1] = CAM_AV_IVVT_IN[1] + C_CAM_OFS_IVVT_IN[1]

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

CAM_EX[1] = CAM_AV_IVVT_EX[1] + C_CAM_OFS_IVVT_EX[1]

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

CAM_IN(_EX)[1] = CAM_AV_IVVT_IN(_EX)[1] + C_CAM_OFS_IVVT_IN(_EX)[1]

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:

CAM_IN[i] = CAM_AV_IVVT_IN[i] + C_CAM_OFS_IVVT_IN[i]

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

CAM_EX[i] = CAM_AV_IVVT_EX[i] + C_CAM_OFS_IVVT_EX[i]

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

CAM_IN(_EX)[i] = CAM_AV_IVVT_IN(_EX)[i] + C_CAM_OFS_IVVT_IN(_EX)[i]

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_OFS_IVVT_IN[NC_NR_CBK_IVVT]	1	80...7FH	-48...47.625	0.375	[°CRK]
Offset to CAM_IN[i]; inlet					
C_CAM_OFS_IVVT_EX[NC_NR_CBK_IVVT]	1	80...7FH	-48...47.625	0.375	[°CRK]
Offset to CAM_EX[i]; exhaust					
ID_CRLC_CAM_DIF_IVVT	6	0...FFH	0...0.99609	3.9063e-3	[-]
LDPM_N_32_7_VVTI	6	0...FFH	0...8160	32	[rpm]
Correlation constants for calculation of camshaft position changes					
ID_CRLC_CAM_IVVT	9*9	0...FFH	0...0.99609	3.9063e-3	[-]
LDP_CAM_CHG_IVVT_IP_CRLC	9	0...8000H	0...96	2.9297e-3	[°CRK]
LDP_CAM_OSC_AMPL_IVVT_IP_CRLC	9	0...7FFFH	0...95.99707	2.9297e-3	[°CRK]
Correlation constants for calculation of camshaft position changes					
ID_CRLC_CHG_NEG_LIM_CAM_IVVT	6	0...FFH	0...0.99609	3.9063e-3	[-]
LDPM_N_32_7_VVTI	6	0...FFH	0...8160	32	[rpm]
Limitation of negative changes of correlation constant for camshaft position filtering					

9.42 IVVT Bank Mean Camshaft Position and Valve Overlap

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_MV_IN	V/O	0...FFH	60...155.625	0.375	[°CRK]
Inlet camshaft position bank mean value					
CAM_MV_EX	V/O	0...FFH	-40.125...-135.75	-0.375	[°CRK]
Exhaust camshaft position bank mean value					
VO_IND[NC_NR_CBK_IVVT]	V/O	0...1C7H	0...170.625	0.375	[°CRK]
Current individual cylinder bank valve overlap					
VO	V/O	0...1C7H	0...170.625	0.375	[°CRK]
Mean current valve overlap					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN[NC_NR_CBK_IVVT]	LV_NOT_ADJ_CAM_IVVT_EX[NC_NR_CBK_IVVT]	NLC_IVVT_IN	NLC_IVVT_EX
NC_NR_CBK_IVVT			

FUNCTION DESCRIPTION:

Description:

The bank mean camshaft position CAM_MV_IN(_EX) is used for common consideration of both inlet and of both exhaust camshaft position in an V-engine. If there is only one camshaft bank, the mean value is equal the single available value.

The valve overlap is the distance between the inlet valve openig point and the exhaust valve closing point expressed in °CRK, see Figure "Valve overlap". There are two valve overlaps in an V-engine, VO_IND[i]. A mean value can be created, VO. Engines with one camshaft bank have VO equals to VO_IND[1].

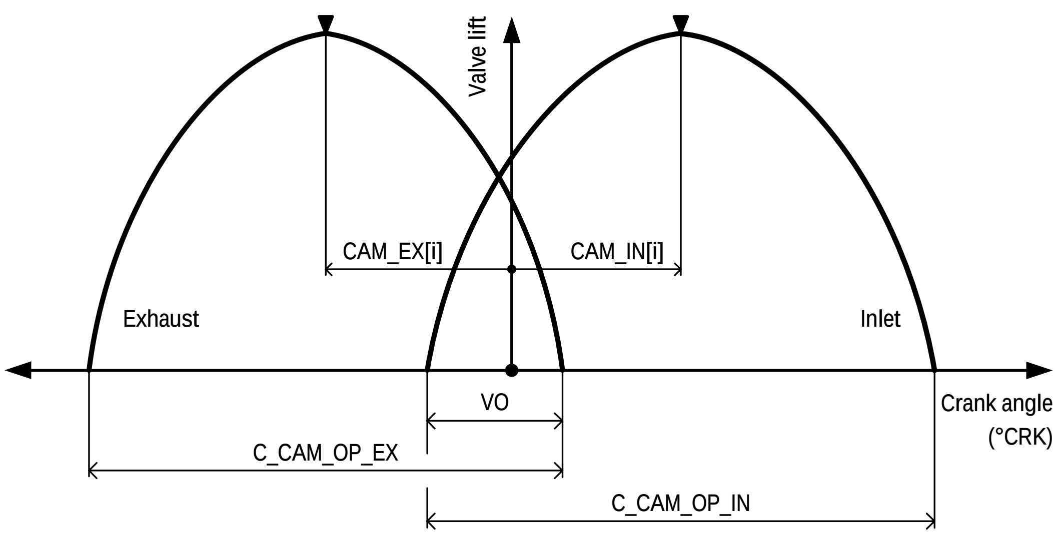


Figure: "Valve overlap."

Application conditions:

Initialization:

At reset:

CAM_MV_IN(_EX) = C_CAM_INI_IN(_EX)

VO = VO_IND[i] =

(C_CAM_OP_IN + C_CAM_OP_EX) / 2 - C_CAM_INI_IN + C_CAM_INI_EX

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

At every engine state

Formula section:

Mean value of camshaft position:

 # NC_NR_CBK_IVVT = 1:

CAM_MV_IN(_EX) = CAM_IN(_EX)[1]

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:

if LV_NOT_ADJ_CAM_IVVT_IN[2] = 0

then

CAM_MV_IN = (CAM_IN[1] + CAM_IN[2]) / 2

else

CAM_MV_IN = CAM_IN[1]

endif

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

if LV_NOT_ADJ_CAM_IVVT_EX[2] = 0

then

CAM_MV_EX = (CAM_EX[1] + CAM_EX[2]) / 2

else

CAM_MV_EX = CAM_EX[1]

endif

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

if LV_NOT_ADJ_CAM_IVVT_IN(_EX)[2] = 0

then

CAM_MV_IN(_EX) = (CAM_IN(_EX)[1] + CAM_IN(_EX)[2]) / 2

else

CAM_MV_IN(_EX) = CAM_IN(_EX)[1]

endif

Valve overlap:

 # NC_NR_CBK_IVVT = 1:

VO_IND[1] = (C_CAM_OP_IN + C_CAM_OP_EX) / 2 - CAM_IN[1] + CAM_EX[1]

VO = VO_IND[1]

 # NC_NR_CBK_IVVT = 2:

VO_IND[1] = (C_CAM_OP_IN + C_CAM_OP_EX) / 2 - CAM_IN[1] + CAM_EX[1]

VO_IND[2] = (C_CAM_OP_IN + C_CAM_OP_EX) / 2 - CAM_IN[2] + CAM_EX[2]

VO = (VO_IND[1] + VO_IND[2]) / 2

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_OP_IN	1	0...FFH	175.125...270.75	0.375	[°CRK]
Opening period of inlet valve					
C_CAM_OP_EX	1	0...FFH	175.125...270.75	0.375	[°CRK]
Opening period of exhaust valve					

9.43 IVVT System at Maximum Adjustment Stop Position

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ADJ_MAX_IVVT_IN[NC_NR_CBK_IVVT]	V/O	0...1H	0...1	1	[-]
1 = IVVT system is close to maximum adjustement stop position, 0 = not close; inlet					
LV_ADJ_MAX_IVVT_EX[NC_NR_CBK_IVVT]	V/O	0...1H	0...1	1	[-]
1 = IVVT system is close to maximum adjustement stop position, 0 = not close; exhaust					

Input data:

CAM_SP_IVVT_IN	CAM_SP_IVVT_EX		
LV_ACT_IND_IVVT_IN[NC_NR_CBK_IVVT]	LV_ACT_IND_IVVT_EX[NC_NR_CBK_IVVT]		

FUNCTION DESCRIPTION:

Description:

The information whether the IVVT system is close to the maximum adjustment stop position is needed in other IVVT functionalities. There can be some effects which do not allow to reach the mean maximum adjustment stop position, e.g., component tolerances, chain/belt oscilations. Also, there would always remain a camshaft position deviation with regard to the setpoint. Diagnoses and adaptations could give wrong results.

If the mean maximum adjustment stop position is exceeded in case of "a long actuator" during a short overshoot, the control algorithm will manage this situation as normal control deviation. Therefore, LV_ADJ_MAX_IVVT_IN(_EX)[i] is 0 in this case.

Application conditions:

Initialization:

At deactivation, at transition "Engine run" → "Engine stop":

LV_ADJ_MAX_IVVT_IN(_EX)[i] = 0

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_ACT_IND_IVVT_IN(_EX)[i] = 1

Formula section:

```
if      Inlet:
                CAM_SP_IVVT_IN <=
                CAM_ADJ_MAX_IVVT_IN + C_CAM_HYS_ADJ_MAX_IVVT
        and     CAM_ADJ_MAX_IVVT_IN <= CAM_AV_IVVT_IN[i]
        and     CAM_AV_IVVT_IN[i] <=
                CAM_ADJ_MAX_IVVT_IN + C_CAM_HYS_ADJ_MAX_IVVT
        Exhaust:
                CAM_SP_IVVT_EX >=
                CAM_ADJ_MAX_IVVT_EX - C_CAM_HYS_ADJ_MAX_IVVT
        and     CAM_ADJ_MAX_IVVT_EX >= CAM_AV_IVVT_EX[i]
        and     CAM_AV_IVVT_EX[i] >=
                CAM_ADJ_MAX_IVVT_EX - C_CAM_HYS_ADJ_MAX_IVVT
then
        LV_ADJ_MAX_IVVT_IN(_EX)[i] = 1
else
        LV_ADJ_MAX_IVVT_IN(_EX)[i] = 0
endif
```

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_ADJ_MAX_IVVT	1	0...A0H	0...60	0.375	[°CRK]
Hysteresis close to maximum adjustment stop position					

9.44 IVVT Reference Position Adaptation Manager

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_DI_AD_REF_CAM_IVVT_IN	O/V	0...1H	0...1	1	-
Adaptation of inlet camshaft reference position: 0 = enabled, 1 = disabled					
LV_DI_AD_REF_CAM_IVVT_EX	O/V	0...1H	0...1	1	-
Adaptation of exhaust camshaft reference position: 0 = enabled, 1 = disabled					
LV_AD_END_IVVT	O/V	0...1H	0...1	1	-
1 = reference position adaptation for active IVVT operation finished, 0 = not finished					
LV_T_AD_REF_IVVT_IN	V	0...1H	0...1	1	-
1 = timer started when setpoint reaches area near reference position, 0 = not started; inlet					
LV_T_AD_REF_IVVT_EX	V	0...1H	0...1	1	-
1 = timer started when setpoint reaches area near reference position, 0 = not started; exhaust					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LV_AD_END_CAM_IN[NC_NR:CAM_CBK]	LV_AD_END_CAM_EX[NC_NR:CAM_CBK]	LV_AD_REQ_IVVT	LV_ACT_IVVT
CAM_SP_IND_IVVT_IN_i	CAM_SP_IND_IVVT_EX_i	NLC_IVVT_IN	NLC_IVVT_EX
LV_CAM_SP_REF_AD_DI_IVVT	NC_NR_CBK_IVVT	CAM_AV_IVVT_IN[NC_NR_CBK_IVVT]	CAM_AV_IVVT_EX[NC_NR_CBK_IVVT]

FUNCTION DESCRIPTION:

General information:

Two cases are distinguished for the reference position adaptation. The first one is the activation adaptation, see "IVVT State". It is necessary for activating the IVVT system, e.g., after the engine start. The second case is an adaptation if IVVT is active. This adaptation is dedicated for the revaluation of the camshaft position determination by ENSD for changed conditions, e.g., the engine temperature (engine lengthening). There is no direct impact on the IVVT functionality.

Description:

The activation adaptation is managed by the IVVT state algorithm, see "IVVT State". It means that if LV_AD_REQ_IVVT = 1 an adaptation is required and the conditions for it are fulfilled. This request resets LV_AD_END_IVVT to 0. If ENSD reports the end of all adaptations LV_AD_END_IVVT is set to 1. LV_DI_AD_REF_CAM_IVVT_IN(_EX) is to 1, too. This is an information for ENSD that the adaptation end has been recognized.

The reference position adaptation is done in the IVVT active state, too. It is assumed that the camshaft position setpoint reaches the reference position in some engine operating points. The adaptation is enabled everytime if the setpoint reaches the area near the reference position, LV_CAM_SP_REF_AD_DI_IVVT equals 0 and the delay C_DLY_AD_REF_IVVT for the camshaft adjusting expires. If ENSD reports the end of all inlet/exhaust adaptations, i.e., the inlet and the exhaust adaptation are independent, LV_DI_AD_REF_CAM_IVVT_IN(_EX) is set to 1, i.e., the adaptation end has been recognized. If the conditions for the adaptation still persist the adaptation is started over and over again.

Application conditions:

Initialization:

At reset, at transition "Engine run" → "Engine stop":

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

LV_AD_END_IVVT = 0

At activation:

if LV_ACT_IVVT = 0

then LV_AD_END_IVVT = 0

  if LV_CAM_SP_REF_AD_DI_IVVT = 0

  then LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 0

  else LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

  endif

endif

LV_T_AD_REF_IVVT_IN(_EX) = 0

At deactivation:

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

Recurrence:

360 °CRK

Activation:

LV_AD_REQ_IVVT = 1

or LV_ACT_IVVT = 1

Formula section:

if LV_AD_END_IVVT = 0

then *IVVT inactive*

  if LV_CAM_SP_REF_AD_DI_IVVT = 0

  then LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 0

  Wait for end of activation adaptation

  else LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

  endif

else *IVVT active*

  if LV_NOT_ADJ_CAM_IVVT_IN(_EX)_1 = 1

  then *Camshaft phasing switched off*

  LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

  else Algorithm for enabling adaptation

  endif

endif

Wait for end of activation adaptation:

if LV_AD_END_CAM_IN(_EX)_1 = 1

then LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

endif

if (LV_AD_END_CAM_IN_1 = 1 **or** LV_NOT_ADJ_CAM_IVVT_IN_1 = 1)

and (LV_AD_END_CAM_EX_1 = 1 **or** LV_NOT_ADJ_CAM_IVVT_EX_1 = 1)

then *ENSD reports end of all adaptations*

LV_AD_END_IVVT = 1

endif

Algorithm for enabling adaptation:

if LV_CAM_SP_REF_AD_DI_IVVT = 0

and CAM_SP_IND_IVVT_IN(_EX)_1 = C_CAM_INI_IN(_EX)

then *Setpoint near reference position and no full load*

if LV_T_AD_REF_IVVT_IN(_EX) = 0

then *Setpoint just reaches area near reference position*

Start Timer_C_DLY_AD_REF_IVVT

LV_T_AD_REF_IVVT_IN(_EX) = 1

endif

if **inlet:**

Timer_C_DLY_AD_REF_IVVT expired

and CAM_AV_IVVT_IN> (C_CAM_INI_IN - C_CAM_DIF_AD_REF_CAM_IVVT_IN)

exhaust:

Timer_C_DLY_AD_REF_IVVT expired

and CAM_AV_IVVT_EX < (C_CAM_INI_EX + C_CAM_DIF_AD_REF_CAM_IVVT_EX)

then *Time for adjusting expired, adaptation possible*

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 0

if LV_AD_END_CAM_IN(_EX)_1 = 1

then *ENSD reports end of all inlet/exhaust adaptations*

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

endif

else *Time for adjusting not expired, adaptation disabled*

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

endif

else *Setpoint not near reference position or full load, adaptation disabled*

LV_DI_AD_REF_CAM_IVVT_IN(_EX) = 1

LV_T_AD_REF_IVVT_IN(_EX) = 0

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_DLY_AD_REF_IVVT	1	1...FFH	0.1...25.5	0.1	s
Time to wait after setpoint reaches area near reference position before enabling adaptation					
C_CAM_DIF_AD_REF_CAM_IVVT_IN	1	0...A0H	0...60	0.375	°CRK
Minimum CAM_AV_IVVT_IN deviation from C_CAM_INI_IN position to enable reference position adaptation					
C_CAM_DIF_AD_REF_CAM_IVVT_EX	1	0...A0H	0...60	0.375	°CRK
Minimum CAM_AV_IVVT_EX deviation from C_CAM_INI_EX position to enable reference position adaptation					

9.45 Camshaft offset adaptation manager

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_LDC_DATA_ACQ_RST	O/V	0...1H	0...1	1	-
Logical bit to reset(=1) the activation conditions and data acquisition functionality					
STATE_CAM_OFS_AD	O/V	0H 1H 2H 3H 4H 5H	PASSIVE DETECTION_OF_AD_OPP REQUEST_AR_RED_AD_CLC ADAPTATION_ACTIVE OPTIMIZATION POST_OPTIMIZATION	1	-
State of camshaft-offset adaptation					
IDX_OPP_CAM_OFS_AD	O/V	1...4H	1...4	1	-
Axis index of ID_N_OPP_CAM_OFS_AD indicating N/MAP operating point for adaptation					
LV_CAM_OFS_AD_OPP_CDN	O/V	0...1H	0...1	1	-
Logical bit indicating whether conditions for CAM_OFS_AD at a certain N/MAP operating point are fullfilled					
LV_CAM_OFS_AD_OPTM_RST	O/V	0...1H	0...1	1	-
Logical bit to reset(=1) the optimisation algorithm					
LV_CPS_INH_REQ_CAM_OFS_AD	O/V	0...1H	0...1	1	-
Logical bit indicating request for Canister purge flow deactivation i.e. request for FLOW_CPS=0					
LV_AR_RED_AD_REQ_CAM_RST	O/V	0...1H	0...1	1	-
Logical bit to reset variables used in AR_RED_ADcalculation request					
CAM_OFS_IVVT_IN[NC_NR_OPP_CAM_OFS_AD]	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Optimisation paramter no. 1 - offset for inlet camshaft position					
CAM_OFS_IVVT_EX[NC_NR_OPP_CAM_OFS_AD]	O/V/S	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Optimisation paramter no. 2 - offset for exhaust camshaft position					
IDX_MAP_OPP_CAM_OFS_AD	O/V	1...4H	1...4	1	-
Axis index of ID_MAP_MES_OPP_CAM_OFS_AD indicating MAP_MES operating point for adaptation					
LV_DET_FIRST_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	O/V/S	0...1H	0...1	1	-
Logical variable indicating whether camshaft-adaptation at the desired oeprating points has already run and delivered valid results.					
LV_PQ_PUT_CTL_SWI_REQ_CAM_AD	O/V	0...1H	0...1	1	-
Logical variable indicating request for switching of PQ threshold from Camshaft-offset adaptation functionality					
PQ_PUT_CTL_CAM_OFS_AD	O/V	0...FFFFH	0...15.9997559	2.44141E-4	-
PQ threshold for INSY_Controller when requested by Camshaft-offset adaptation functionality					
LV_CAM_OFS_AD_SYM_POS_ERR[NC_NR_OPP_CAM_OFS_AD]	O/V/S	0...1H	0...1	1	-
Logical bit indicating that over several driving cycles, the adaptation has been detecting sysptoms for a big camshaft-position error					
LV_DET_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	O/V	0...1H	0...1	1	-
Logical variable indicating whether camshaft-adaptation at the desired operating points has run and delivered valid results					
CTR_CYCNR_CAM_AD_CUR_DC	V	0...FFH	0...255	1	-
Counter for number of adaptation runs done inthe current driving cycle					
MAF_DIF_SQ_TOT_CAM_OFS_AD_RES[NC_NR_OPP_CAM_OFS_AD]	V/S	0...FFFFH	0...255.996094	0.00390625	(kg/h)²
Vector with value of MAF_DIF_CAM_OFS_AD_SQ_TOT when adaptation result is detected as valid					
CTR_DLY_CAM_AD_BEG	V	0...FFFFH	0...6.5535E+4	1	-
Counter for delay time before the next camshaft-offset adaptation run is started					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_CAM_OFS_AD_RES_NOT_PLAUS[NC_NR_OPP_CAM_OFS_AD]	V/S	0...FFFFH	0...6.5535E+4	1	-
Counter to track the number of adaptation runs which detect a very large camshaft-position error					
IDX_MAP_OPP_CAM_AD_TMP	V	0...4H	0...4	1	-
Temporary - takes the value of either IDX_MAP_xx_TMP_1 or IDX_MAP_xx_TMP_2					
IDX_OPP_CAM_OFS_AD_TMP	V	0...4H	0...4	1	-
Temporary - takes the value of either IDX_xx_TMP_1 or IDX_xx_TMP_2					
LV_CAM_OFS_AD_CUR_DC_CMPL	V	0...1H	0...1	1	-
Logical bit indicating that maximum allowed number of adaptation runs in current driving cycle has been reached					
IDX_MAP_OPP_CAM_AD_TMP_1	V	0...4H	0...4	1	-
Axis index of ID_MAP_MES_OPP_CAM_OFS_AD indicating current MAP_MES operating point - value 0 indicates manifold pressure not at one of the values where adaptation can be done					
IDX_MAP_OPP_CAM_AD_TMP_2	V	0...4H	0...4	1	-
Temporary value used only if MAP_MES operating point changes during signal acquisition.					
IDX_OPP_CAM_OFS_AD_TMP_1	V	0...4H	0...4	1	-
Axis index of ID_N_OPP_CAM_OFS_AD indicating current N operating point value 0 indicates engine speed not at one of the values where adaptation can be done					
IDX_OPP_CAM_OFS_AD_TMP_2	V	0...4H	0...4	1	-
Temporary value used only if N operating point changes during signal acquisition.					
LV_DET_CAM_AD_MAP_OPP_CHG	V	0...1H	0...1	1	-
Logical variable indicating that a change in the manifold pressure operating point was detected during the adaptaion.					
LV_DET_CAM_AD_N_OPP_CHG	V	0...1H	0...1	1	-
Logical variable indicating that a change in the engine speed operating point was detected during the adaptaion.					
CTR_CYCNR_CAM_AD_DEAC	V	0...FFH	0...255	1	-
Counter to keep track of the number of times the adaptation was aborted					

Input data:

STATE_AR_RED_AD_ADD	FLOW_CPS	MAP_MES	N
VS	TCO	PQ	LV_IGK
LV_MAP_CTL	LV_CAM_OFS_AD_EXT_REQ	LV_CAM_SP_ADJ_TOT_END	LV_LDC_CAM_OFS_AD_ACT
LV_CAM_SP_ADJ_SIG_SAVE_END	CTR_CYCNR_SIG_REC_CAM_OFS_AD	CTR_DLY_AR_RED_AD_REQ_CAM	CAM_OFS_IVVT_IN_OPTM_OUT
CAM_OFS_IVVT_EX_OPTM_OUT	MAF_DIF_CAM_OFS_AD_SQ_TOT	LV_CAM_OFS_AD_OPTM_END	LV_ERR_CAM_DE_IVVT_IN[NC_NR_CBK_IVVT]
LV_ERR_CAM_DE_IVVT_EX[NC_NR_CBK_IVVT]	NC_NR_OPP_CAM_OFS_AD	NC_NR_MAP_OPP_CAM_OFS_AD	T_CTR_AMP_AD_CMPL
N_GRD	C_CTR_DLY_AR_RED_AD_CAM_MAX_1	LV_AR_RED_AD_ADD_REQ_CAM_CMPL[NC_NR_OPP_CAM_OFS_AD]	AR_RED_AD_ADD_REQ_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]
LV_END_CAM_POS_DIAG_EXT[NC_NR_OPP_CAM_OFS_AD]	LV_CAM_OFS_AD_EXT_INH		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AR_RED_AD_ADD_CAM_OFS_AD_MAX	1	0...7FFFH	0...29.2959805	8.9407E-4	cm²
Maximum allowed value of AR_RED_AD_ADD which is calculated for the CAM_OFS_AD functionality					
C_CAM_OFS_AD_IVVT_MIN	1	0...7FFFH	0...47.9985352	0.00146484	°CRK
Camshaft offset value which is subtracted from every adaptation result to consider only results above lower threshold					
C_CAM_OFS_IVVT_CHG_MIN	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Minimum required change in value between current and old adaptation results in order to declare current results					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
as useable.					
C_CAM_OFS_IVVT_L_THD	1	0...7FFFH	0...47.9985352	0.00146484	°CRK
Minimum value of adaptation results - limitation to consider only effective camshaft-offset values					
C_CAM_OFS_IVVT_MAX	1	0...7FFFH	0...47.9985352	0.00146484	°CRK
Maximum allowed value of adaptation results - limitation to consider only realistic camshaft-offset results.					
C_CTR_CAM_AD_RES_NOT_PLAUS_MAX	1	0...FFFFH	0...6.5535E+4	1	-
Maximum number of adaptation runs resulting in a very large results error after which an error bit is set to indicate a possible large camshaft-position error					
C_CTR_DLY_AD_BEG_MAX	1	0...FFFFH	0...6.5535E+4	1	-
Delay time before then next adaptation run is started					
C_CYCNR_MAX_CAM_OFS_AD_DC	1	0...FFH	0...255	1	-
Maximum number of adaptation runs allowed in a driving cycle					
C_MAF_DIF_CAM_AD_SQ_TOT_MAX	1	0...FFFFH	0...255.996094	0.00390625	(kg/h)²
Thrsehold on total squared error at the end of the optimisation to evaluate quality of results					
C_MAP_MES_HYS_OPP_CAM_OFS_AD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Hysteresis on manifold pressure operating point for adaptation					
C_MAP_MES_MIN_CAM_OFS_AD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum manifold pressure for activation of camshaft-offset-adaptation					
C_MAP_MES_MIN_HYS_CAM_OFS_AD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Hysteresis on minimum manifold pressure for activation of camshaft-offset-adaptation					
C_NR_CAM_AD_DEAC_MAX	1	0...FFH	0...255	1	-
Maximum number of times the adaptation can be aborted before it is inhibited completely for the current driving cycle.					
C_N_GRD_MAX_CAM_OFS_AD	1	0...7FH	0...4.064E+3	32	rpm/s
Maximum value of engine speed gradient above which the camshaft-offset adaptation is deactivated					
C_N_HYS_OPP_CAM_OFS_AD_1	1	0...1FE0H	0...8.16E+3	1	rpm
Hysteresis on desired engine speed operating point for camshaft-offset adaptation					
C_N_HYS_OPP_CAM_OFS_AD_2	1	0...1FE0H	0...8.16E+3	1	rpm
Hysteresis on engine speed, used only if a change of engine speed operating point detected during adaptation					
C_PQ_MAX_HYS_CAM_OFS_AD	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Hysteresis on maximum PQ threshold upto which the Camshaft-offset adaptation functionality can be active					
C_PQ_PUT_CTL_CAM_AD	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Pressure quotient to switch from area to pressure control for Ca,shaft-offset adaptation functionality only					
C_TCO_MIN_CAM_OFS_AD	1	0...FEH	-48...142.5	0.75	°C
TCO threshold for activation of camshaft-offset adaptation					
C_T_AMP_AD_CMPL_MAX_CAM_AD	1	0...FFFFH	0...6.5535E+4	1	s
Maximum allowed time since the last ambinet pressure adaptation to enable Camshaft-offset adaptation					
C_T_DLY_CAM_OFS_AD_OPP_CDN	1	0...FFFFH	0...1.3107E+3	0.02	s
Delay time for starting the camshaft offset adaptation					
C_VS_MIN_CAM_OFS_AD	1	0...FFH	0...255	1	km/h
Minimum engine speed for activation of Camshaft-offset adaptation					
C_VS_MIN_HYS_CAM_OFS_AD	1	0...FFH	0...255	1	km/h
Hysteresis on minimum engine speed for activation of Camshaft-offset adaptation					
LC_CAM_OFS_AD_INH	1	0...1H	0...1	1	-
Logical bit to activate(=0) and deactivate(=1) camshaft-offset adaptation functionality					
LC_CAM_OFS_AD_OPTM_INH	1	0...1H	0...1	1	-
Logical bit to activate(=0) and deactivate(=1) optimisation algorithm					
LC_CAM_OFS_AD_RST	1	0...1H	0...1	1	-
Logical bit to reset(=1) camshaft-offset adaptation functionality					
LC_CAM_SP_ADJ_INH	1	0...1H	0...1	1	-
Logical bit to activate(=0) and deactivate(=1) camshaft adjustment for adaptation purposes					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_INH_AR_RED_AD_ADD_REQ_CAM	1	0...1H	0...1	1	-
Logical bit to inhibit(=1) or allow(=0) AR_RED_AD_ADD calculation request from Camshaft-offset adaptation functionality					
ID_MAP_MES_OPP_CAM_OFS_AD[NC_NR_MAP_OPP_CAM_OFS_AD]	NC_NR_MAP_OPP_CAM_OFS_AD	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDP_MAP_MES_OPP_CAM_OFS_AD	NC_NR_MAP_OPP_CAM_OFS_AD	0...FFFFH	0...5.434E+3	0.08291752	hPa
Desired Manifold pressure operating points for Camshaft-offset adaptation					
ID_N_OPP_CAM_OFS_AD[NC_NR_MAP_OPP_CAM_OFS_AD]	NC_NR_OPP_CAM_OFS_AD	0...1FE0H	0...8.16E+3	1	rpm
LDP_N_OPP_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	0...1FE0H	0...8.16E+3	1	rpm
Desired engine speed operating points for Camshaft-offset adaptation					

9.45.1 INSY_MANAGCAMAD

General Information

In any IC-engine, the positions of the inlet and exhaust camshaft(which have a big influence on the air-mass-flow and emission behaviour) have certain tolerances resulting for e.g. from shift in relative position between camshaft-actuator and actual camshaft position, sensor inaccuracies, manufacturing errors etc. Such tolerances can also vary from one engine to other. Consequently, the camshaft position measured in the reference engine(which was used to calibrate the volumetric efficiency data) could be quite different from a serial production engine making the calculated air mass inaccurate i.e. it would lead to a wrong calculation of the air flowing into the cylinder - the modelled value of MAF_CYL would not be equal to the real air flow into the cylinder, eventually leading to emission problems. In addition to the tolerances in the camshaft position, an error in the throttle area(for e.g. production tolerances or pollution) which can also lead to a wrong air-mass flow, has to be learnt.

The aim of the adaptation is to learn the shift in absolute position of the inlet and exhaust camshafts with respect to the reference engine. The results of the adaptation are offsets to the camshaft position. The adaptation algorithm is based on air flow balance under steady state conditions. It involves the following steps which are coordinated by this module:

waiting for stationary conditions of N, MAP_MES and other variables

adjustment of camshaft to certain inlet and exhaust positions

At each camshaft position, variables like N, MAP_MES, CAM_IN/EX etc. are measured and mean value over certain number of recurrences calculated.

Optimisation algorithm(based on gradient check method) which minimizes the least squared error over all camshaft positions i.e. $\sum(MAF\_THR_i - MAF\_CYL_i)^2$. The parameters that can be varied to find the optimum are CAM_OFS_IVVT_IN, CAM_OFS_IVVT_EX and AR_RED_AD_ADD_REQ_CAM_OFS_AD (reduced area adaptation value to learn non-camshaft related errors)

The variable STATE_CAM_OFS_AD represents the current status of the adaptation. In the state 'Detection_of_AD_OPP', the current engine operating point is compared with the

N/MAP operating points that are needed for camshaft-adaptation. If the current N and MAP_MES match with one of the desired operating points in ID_N_OPP_CAM_OFS_AD and ID_MAP_MES_OPP_CAM_OFS_AD, the bit LV_CAM_OFS_AD _OPP_CDN is set to 1. The index variables IDX_OPP_CAM_ OFS_AD_TMP and IDX_MAP_OPP_CAM_OFS_AD_TMP indicate at which one of the required engine speed/load operating conditions the engine is currently at.

When LV_CAM_OFS_AD_OPP_CDN has been set, the camshaft-adaptation can be started. It is first checked if the AR_RED_AD_ADD value for the detected operating point has already been learnt. If it is has not yet been learnt, then its calculation is triggered in state ‘Request_for_AR_RED_AD’. The reduced-area adaptation values are learnt at a certain inlet/exhaust position where a position error has little influence on the air-mass flow. This is done in conjunction with the Reduced-area adaptation functionality. Once the AR_RED_AD_ADD calculation has been completed, the state machine transits to state ‘Adaptation_Active’. The actual learning of the position error involves the following steps:

- Adjustment of the desired camshaft position (ID_CAM_SP_IVVT_IN/EX_CAM_OFS_AD) – STATE

- Limited dynamic check for relevant variables(LV_LDC_CAM_OFS_ AD_ACT).

- Once the variables have reached a stable value, several signals needed for adaptation are measured over C_CYCNR_SIG_REC_MAX recurrences and their mean value calculated.

- Above mentioned steps done for all the required camshaft positions.

- After signal acquisition mentioned above is carried out successfully(LV_CAM_SP_ADJ_TOT_END=1), the optimisa- tion algorithm is triggered in states ‘OPTIMIZATION' and 'POST_OPTIMIZATION’ .

During the camshaft adjustment process, if the activation conditions are not fulfilled anymore, the adaptation is aborted and all relevant variables initialised in the state 'PASSIVE”. If the adaptation starts again, the adjustment of the camshaft will start again from the first camshaft position combination.

Application Condition

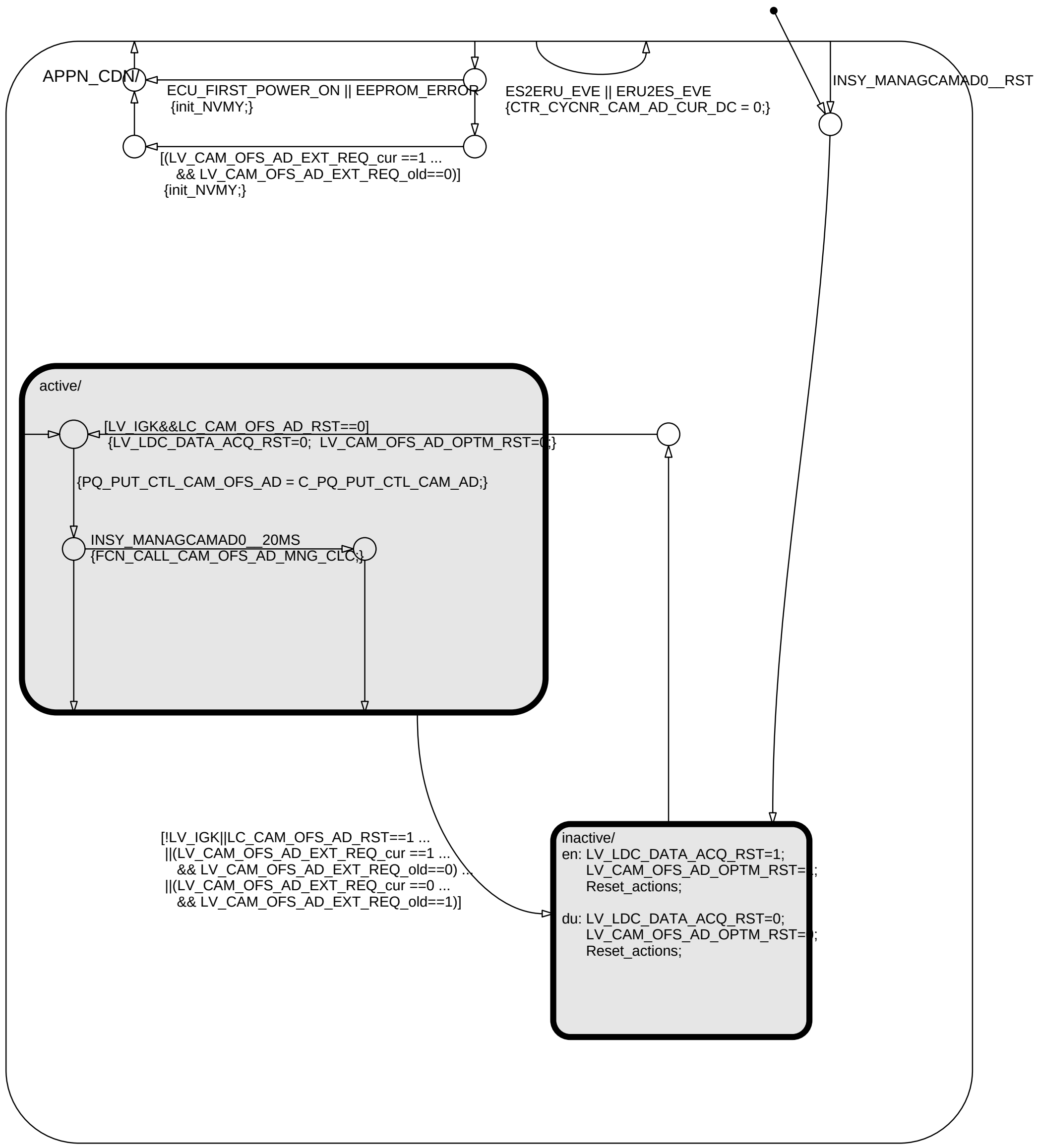


Figure 99 INSY_MANAGCAMAD/ APP_CDN/ Chart

Function Description

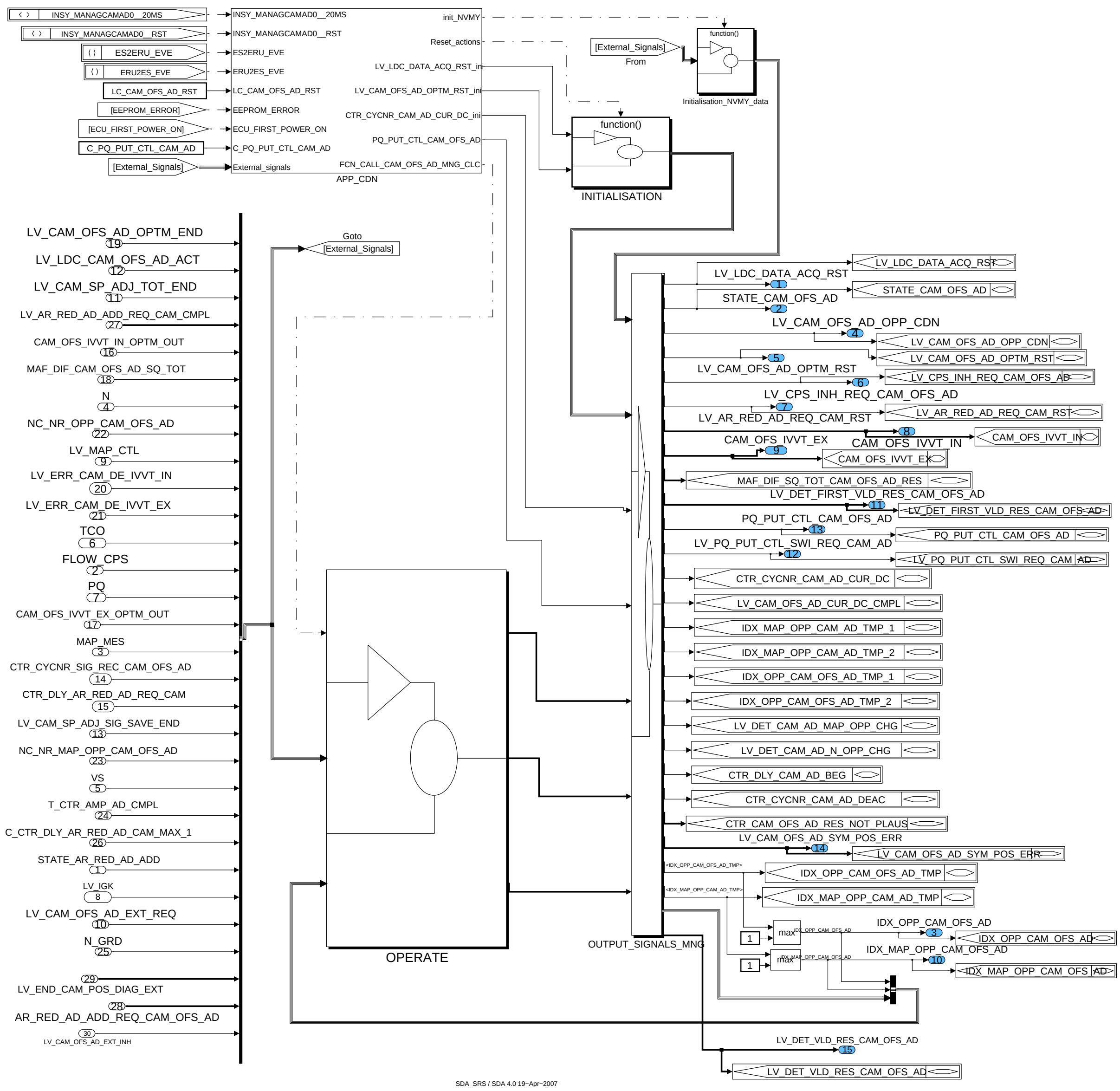


Figure 100 INSY_MANAGCAMAD

9.45.1.1 SUBFUNCTION: INITIALISATION

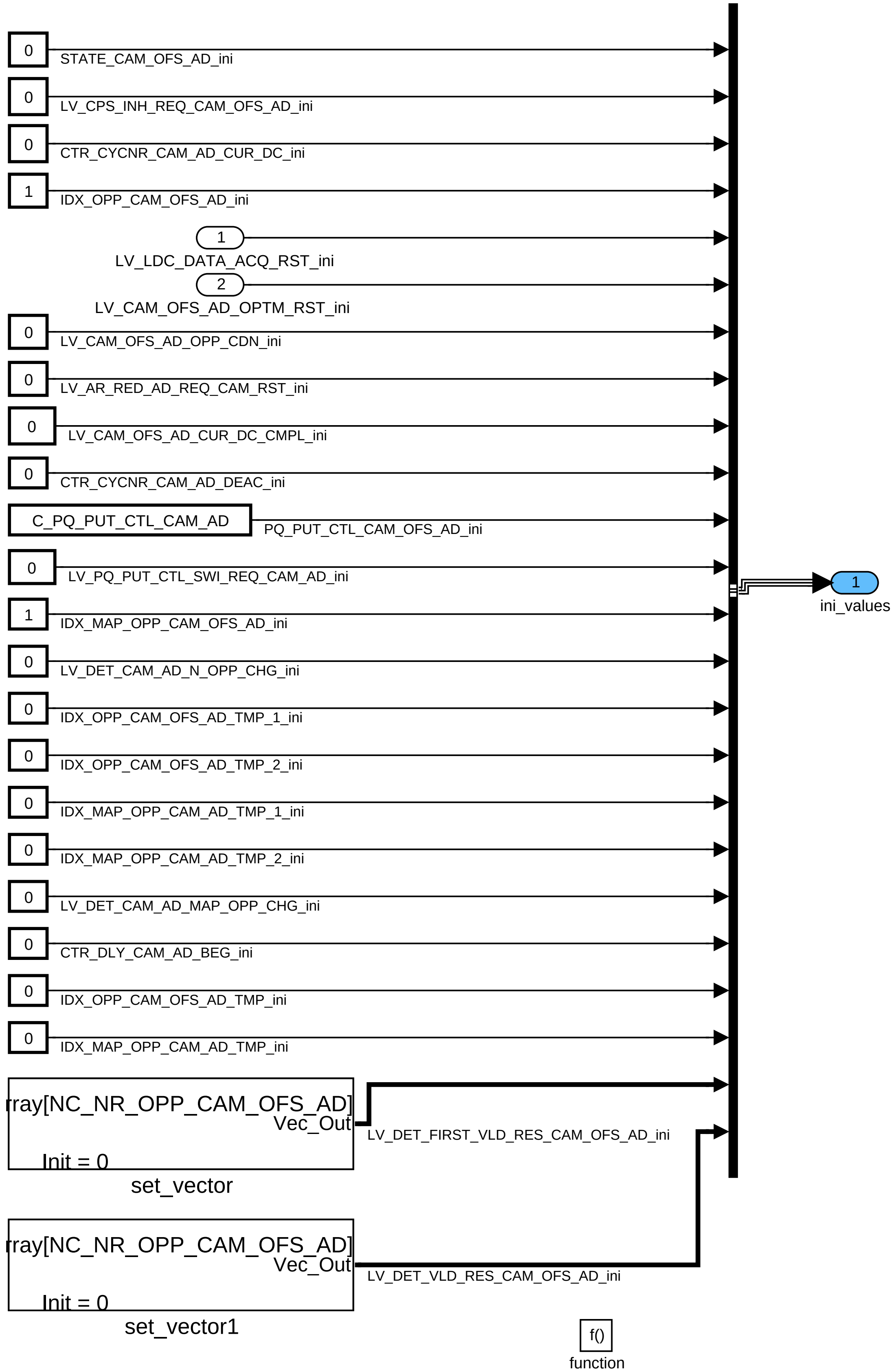

0
STATE_CAM_OFS_AD_ini
0
LV_CPS_INH_REQ_CAM_OFS_AD_ini
0
CTR_CYCNR_CAM_AD_CUR_DC_ini
1
IDX_OPP_CAM_OFS_AD_ini
1
LV_LDC_DATA_ACQ_RST_ini
2
LV_CAM_OFS_AD_OPTM_RST_ini
0
LV_CAM_OFS_AD_OPP_CDN_ini
0
LV_AR_RED_AD_REQ_CAM_RST_ini
0
LV_CAM_OFS_AD_CUR_DC_CMPL_ini
0
CTR_CYCNR_CAM_AD_DEAC_ini
C_PQ_PUT_CTL_CAM_AD
PQ_PUT_CTL_CAM_OFS_AD_ini
0
LV_PQ_PUT_CTL_SWI_REQ_CAM_AD_ini
1
ini_values
1
IDX_MAP_OPP_CAM_OFS_AD_ini
0
LV_DET_CAM_AD_N_OPP_CHG_ini
0
IDX_OPP_CAM_OFS_AD_TMP_1_ini
0
IDX_OPP_CAM_OFS_AD_TMP_2_ini
0
IDX_MAP_OPP_CAM_AD_TMP_1_ini
0
IDX_MAP_OPP_CAM_AD_TMP_2_ini
0
LV_DET_CAM_AD_MAP_OPP_CHG_ini
0
CTR_DLY_CAM_AD_BEG_ini
0
IDX_OPP_CAM_OFS_AD_TMP_ini
0
IDX_MAP_OPP_CAM_AD_TMP_ini
rray[NC_NR_OPP_CAM_OFS_AD]
Vec_Out
Init = 0
set_vector
LV_DET_FIRST_VLD_RES_CAM_OFS_AD_ini
rray[NC_NR_OPP_CAM_OFS_AD]
Vec_Out
Init = 0
set_vector1
LV_DET_VLD_RES_CAM_OFS_AD_ini
f()
function

Figure 101 INSY_MANAGCAMAD/ INITIALISATION

9.45.1.2 NVMY data Handling

The initialisation of the NVMY data should be done during ECU-FIRST-POWER-ON(Brand new ECU) or when an EEPROMN-ERROR occurs or when the adaptation is activated externally (LV_CAM_OFS_AD_EXT_REQ=1). For the external activation, NVMY data are reset only if the adaptation has not been carried out already in the current DC.

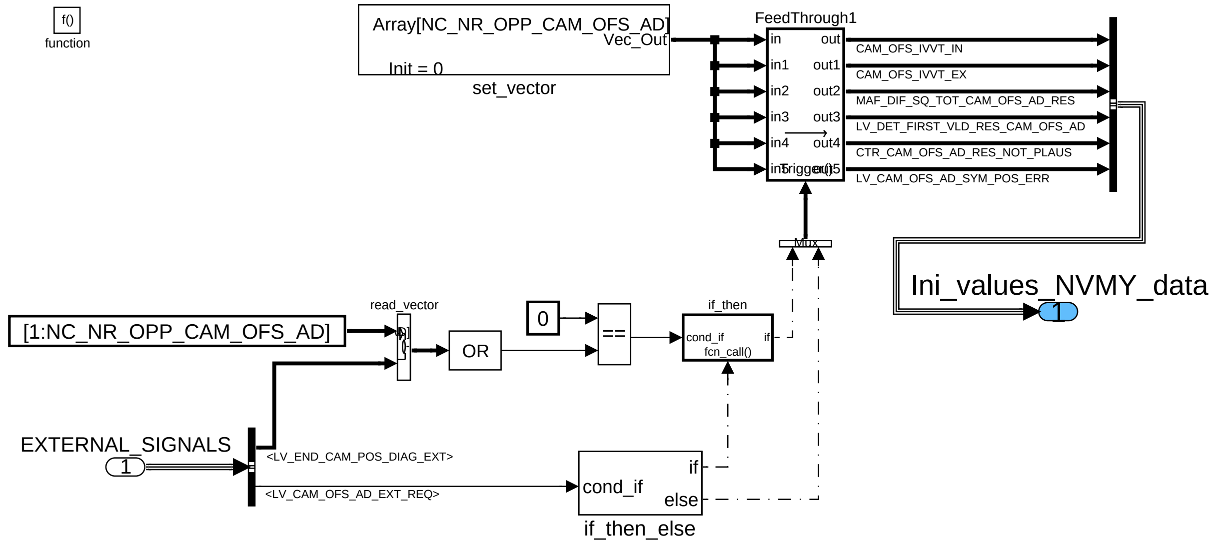


Figure 102 INSY_MANAGCAMAD/ Initialisation_NVMY_data

9.45.1.3 INSY_MANAGCAMAD/OPERATE

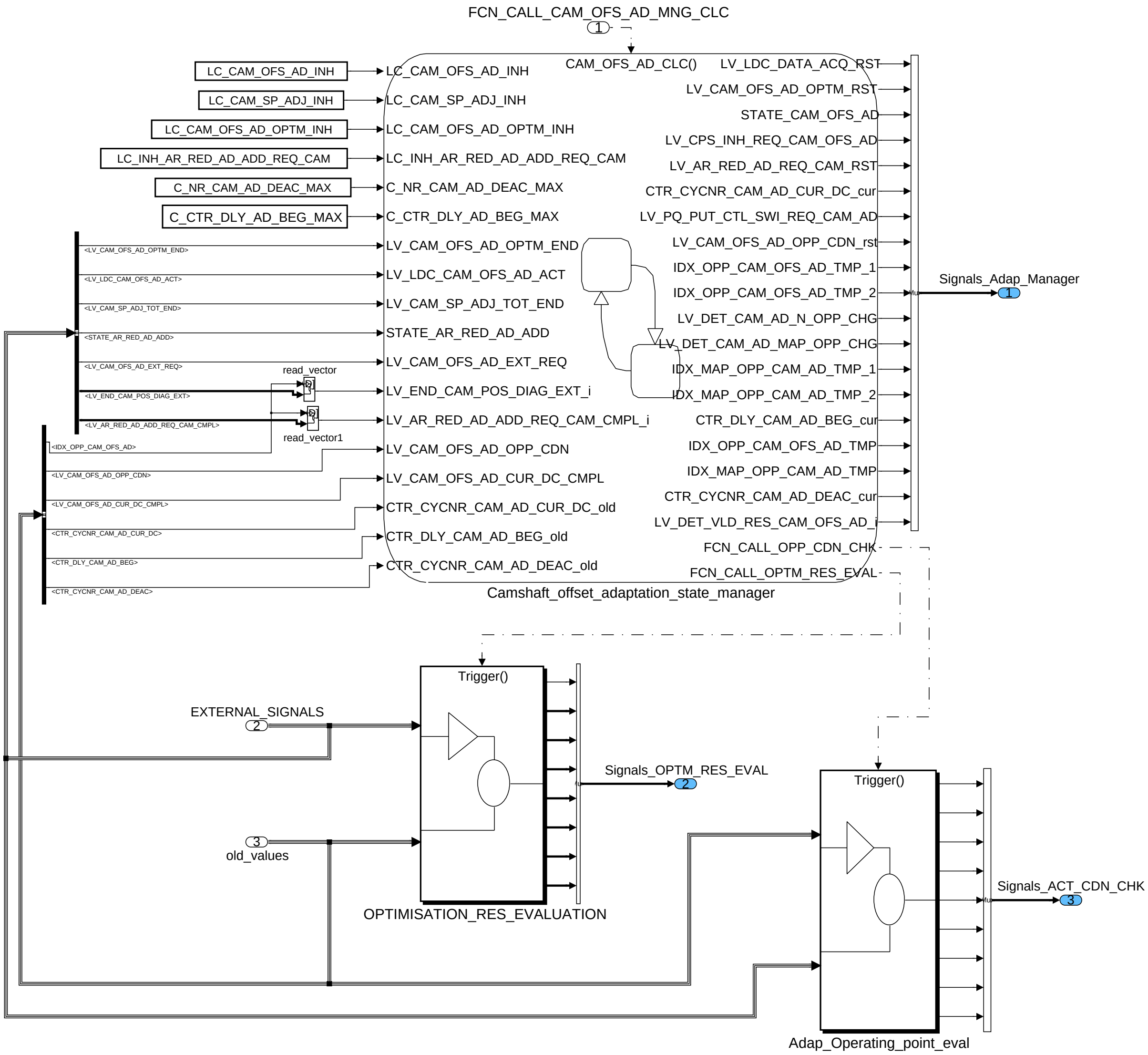


Figure 103 INSY_MANAGCAMAD/ OPERATE

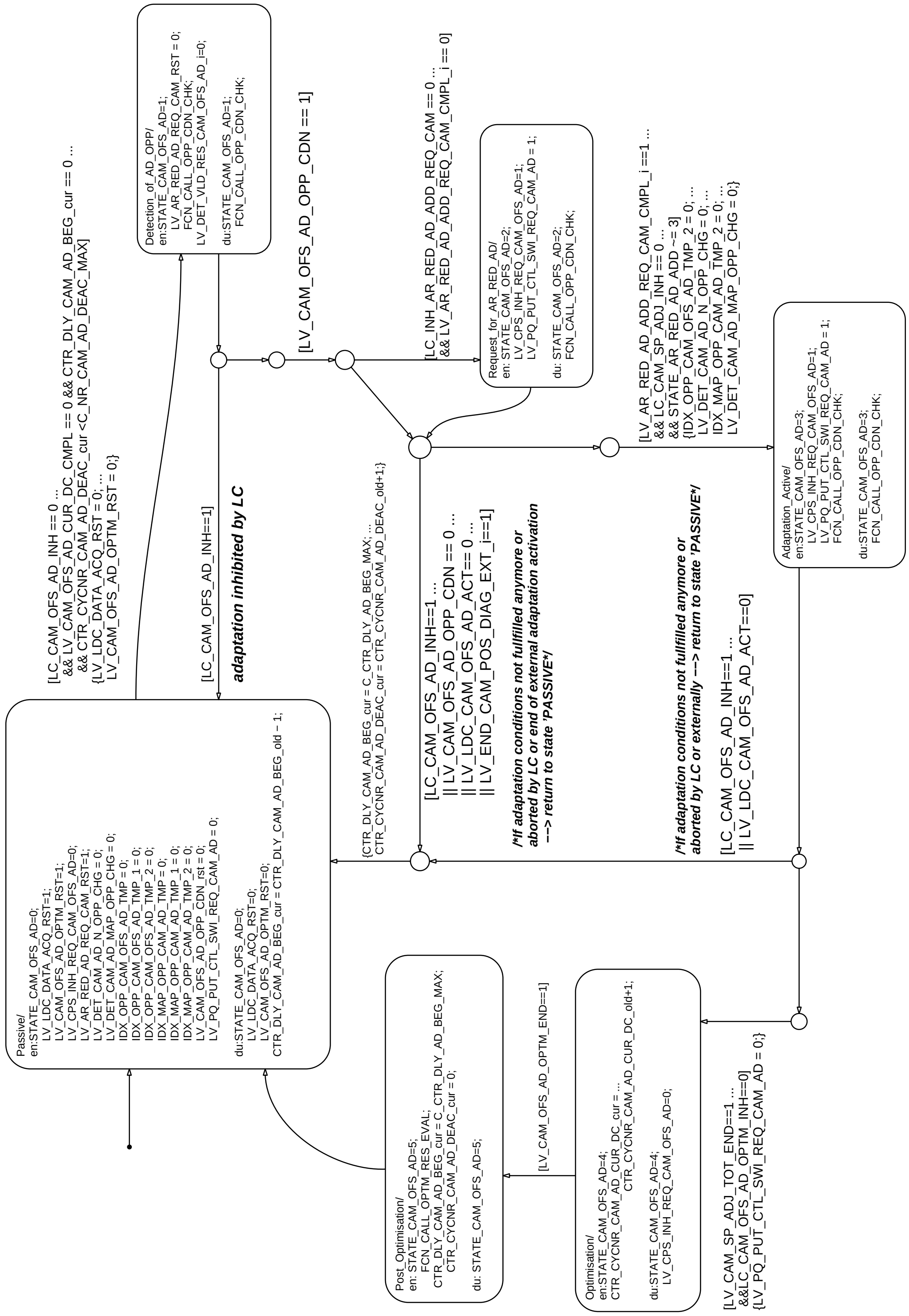


Figure 104 INSY_MANAGCAMAD/ OPERATE/ Camshaft_offset_adaptation_state_manager

Detection and Evaluation of adaptation operating point

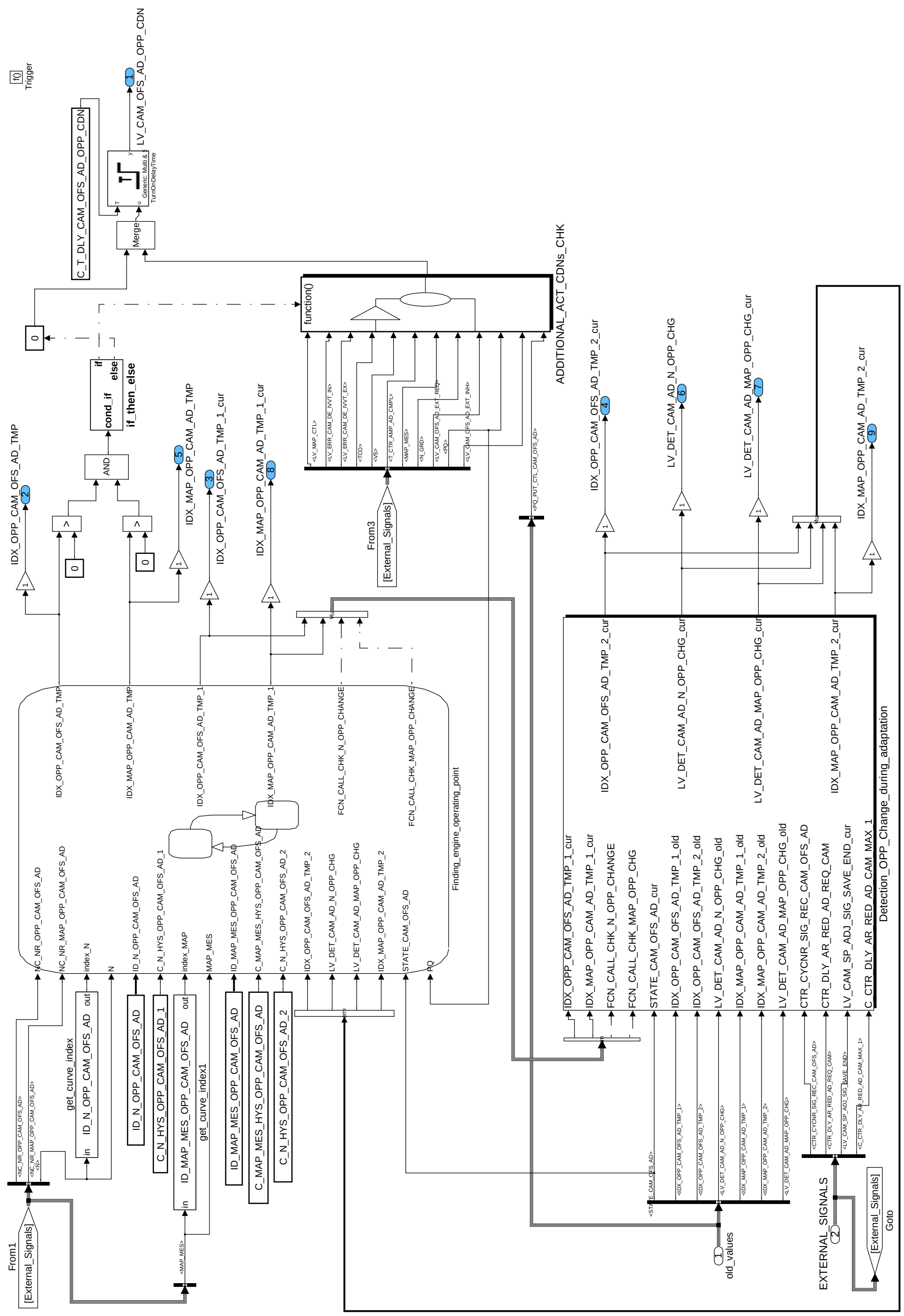


Figure 105 INSY_MANAGCAMAD/ OPERATE/ Adap_Operating_point_eval

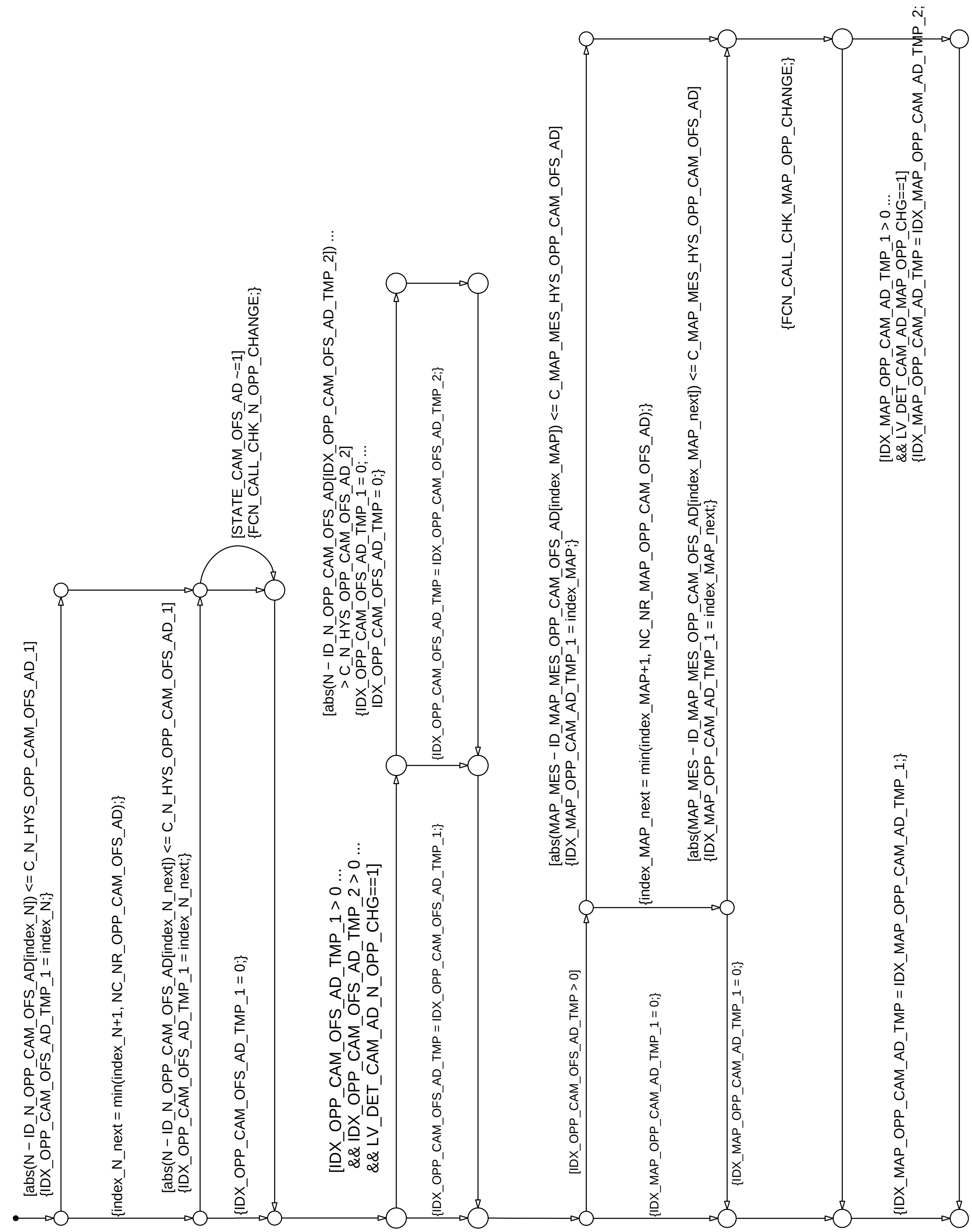


Figure 106 INSY_MANAGCAMAD/ OPERATE/ Adap_Operating_point_eval/ Finding_engine_operating_point

Detection of speed/load operating point change during adaptation

If during the signal acquisition phase of the adaptation or AR_RED_ AD_ADD calculation, the engine speed or manifold pressure were to change leading to a shift in value of IDX_OPP_CAM_OFS_AD or IDX_MAP_OPP_CAM_ OFS_AD, the value of these indices are frozen at their previous value to avoid the adaptation from being interrupted or adjusted the camshaft to a new position For software purposes the following variables were introduced:

IDX_OPP_CAM_OFS_AD = max(1,IDX_OPP_CAM_OFS_AD_TMP)

IDX_MAP_OPP_CAM_OFS_AD = max(1,IDX_MAP_OPP_CAM_AD_TMP)

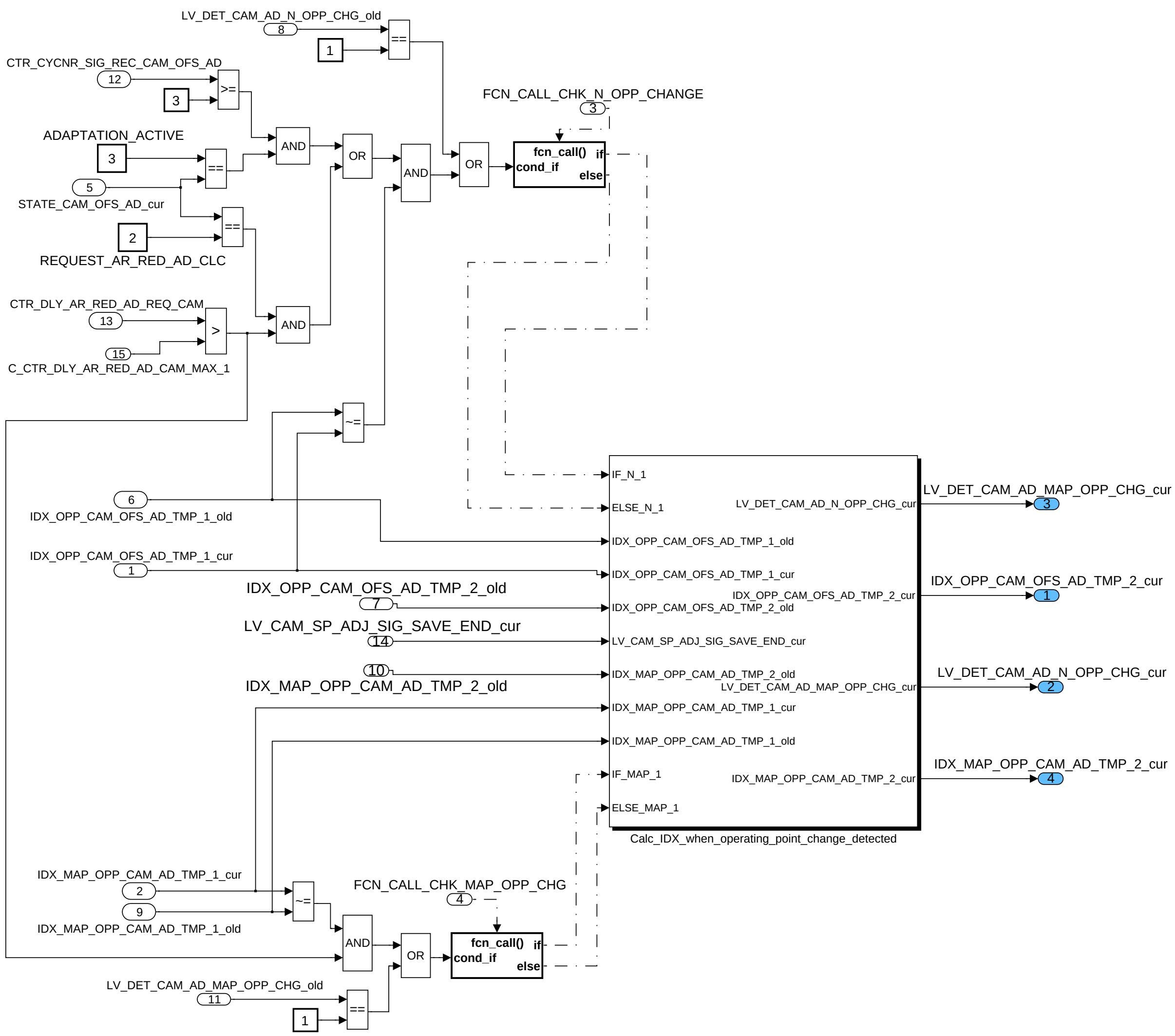


Figure 107 INSY_MANAGCAMAD/ OPERATE/ Adap_Operating_point_eval/ Detection_OPP_Change_during_adaptation

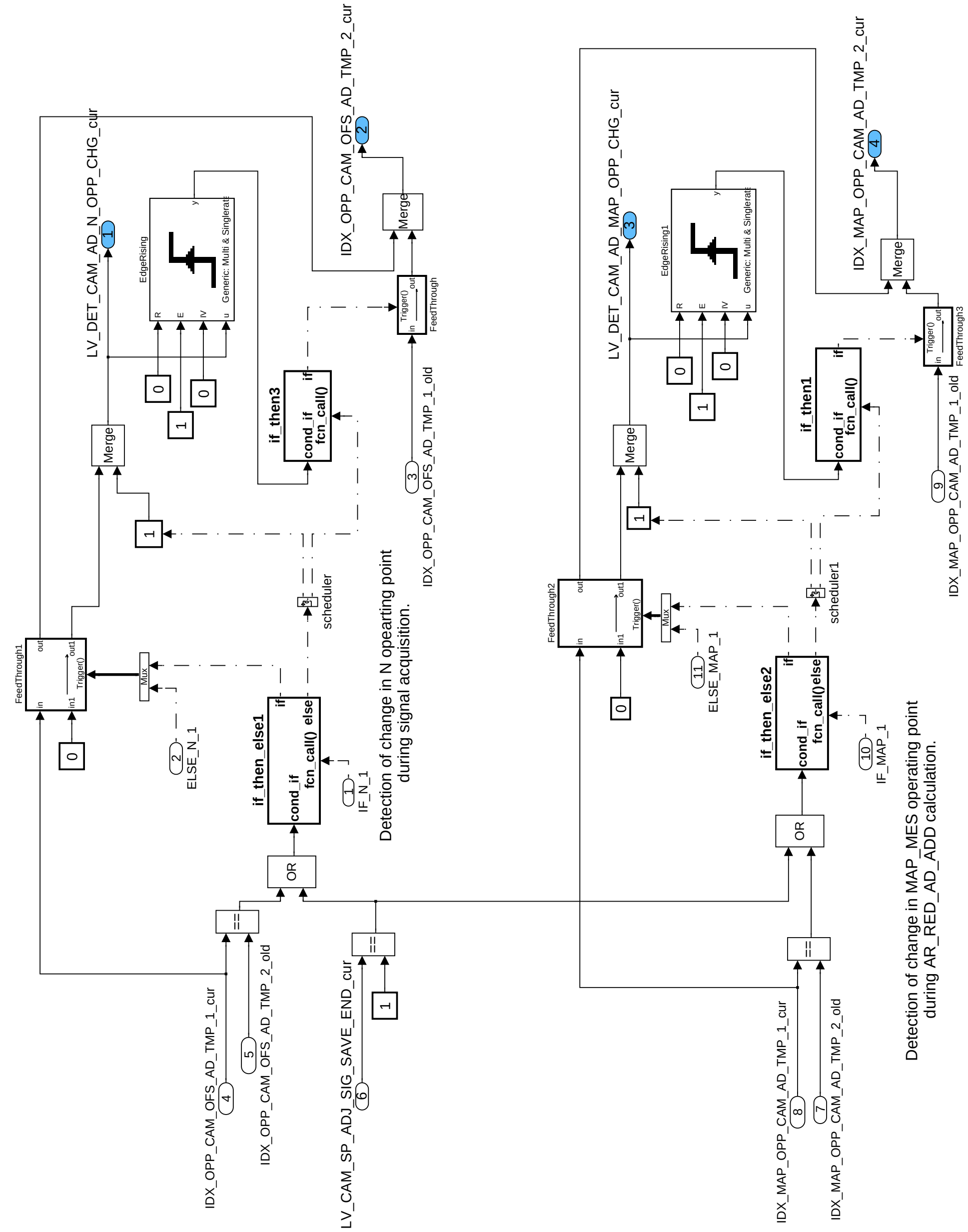


Figure 108 INSY_MANAGCAMAD/ OPERATE/ Adap_Operating_point_eval/ Detection_OPP_Change_during_adaptation/ Calc_IDX_when_operating_point_change_detected

Environmental data check for activation of adaptation

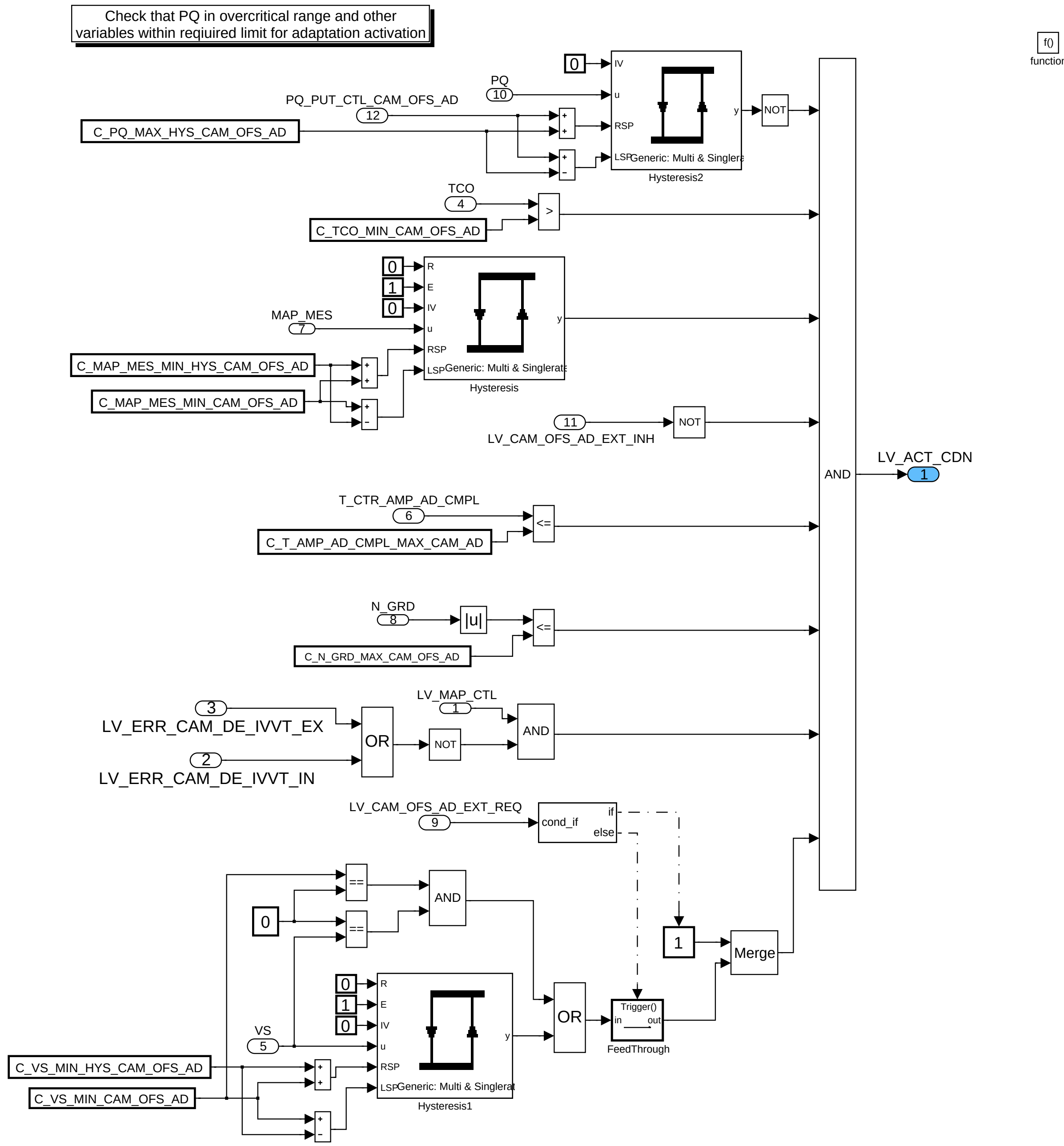


Figure 109 INSY_MANAGCAMAD/ OPERATE/ Adap_Operating_point_eval/ ADDITIONAL_ACT_CDNs_CHK

Evaluation of Adaptation results

After a successful adaptation run, the 'quality' of the results i.e. CAM_OFS_IVVT_IN/EX_OPTM_OUT is checked to make sure that the optimisation algorithm has detected a position error which is within a plausible range. This is done to ensure that other functionalities which might eventually use the adaptation results for a correction of the inlet/exhaust position work with the correct values. This is done by first checking whether CAM_OFS_IVVT_IN/EX are within a certain range – C_CAM_OFS_IVVT_L_THD and

C_CAM_OFS_IVVT_MAX and the squared error at the end of the optimisation is below a certain value. If these conditions are not fulfilled, the results are not utilised.

Under normal driving conditions on the road, the algorithm for the same engine operating point will always vary by several degrees due to reasons like averaging during the signal acquisition and changing environmental conditions. Another reason for the range check is, inlet/exhaust position error below a certain value(depending on the engine, normally around 2-3°CRK) do not have much effective influence on the air-mass flow into the cylinder. The least-squared error on the other hand gives a good indication of how well the optimisation algorithm was able to learn the position error. A high squared error indicates that the algorithm was not able to completely learn the error.

If the adaptation constantly learns a camshaft position error which is bigger than the threshold C_CAM_OFS_IVVT_MAX, then this is used as an indication that there are other errors in the air-path which have not been detected or have changed without being detected by a diagnosis i.e. small hole in intake-manifold, change in Blowby characteristics, a jump in the camshaft-driving-chain etc.

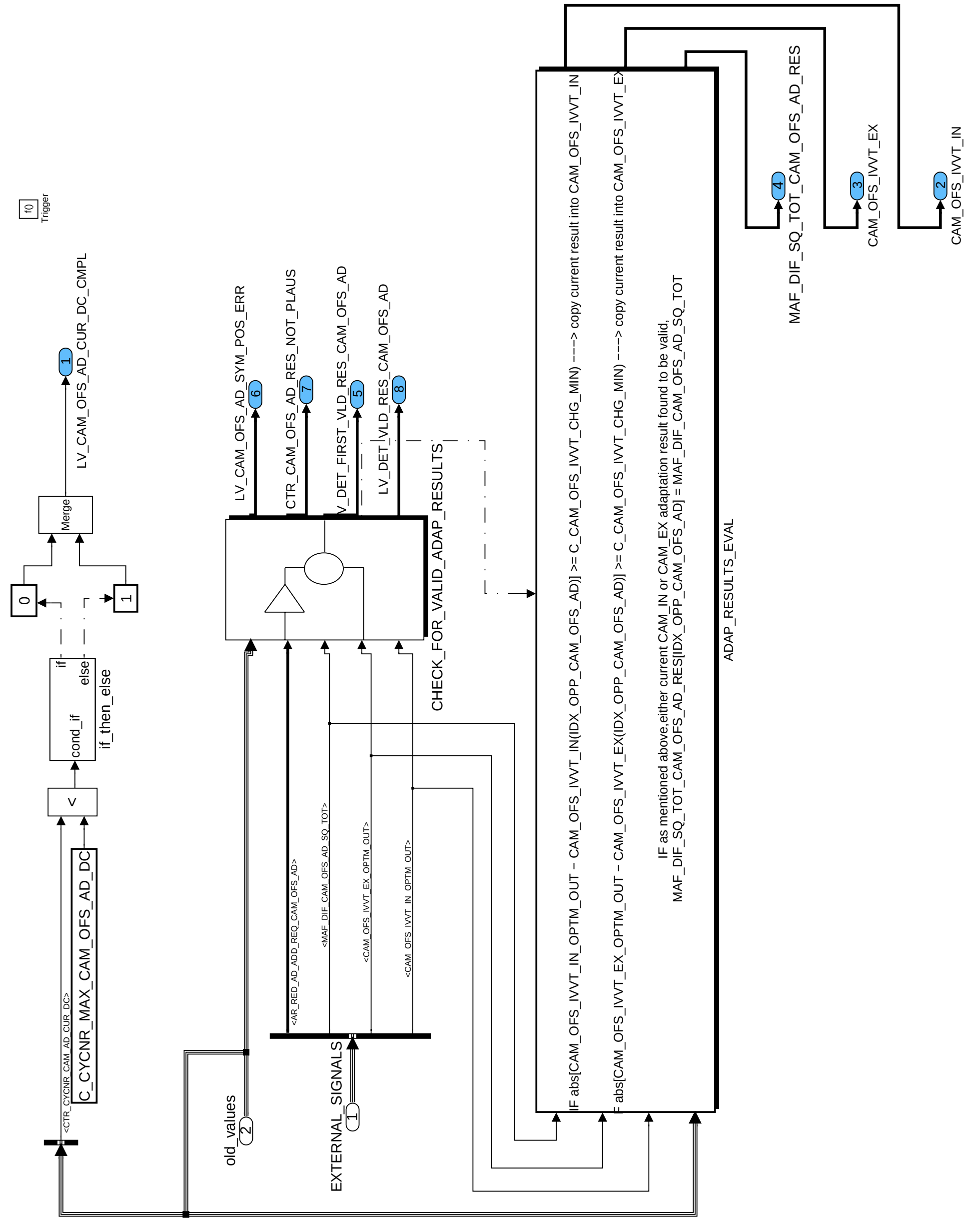


Figure 110 INSY_MANAGCAMAD/ OPERATE/ OPTIMISATION_RES_EVALUATION

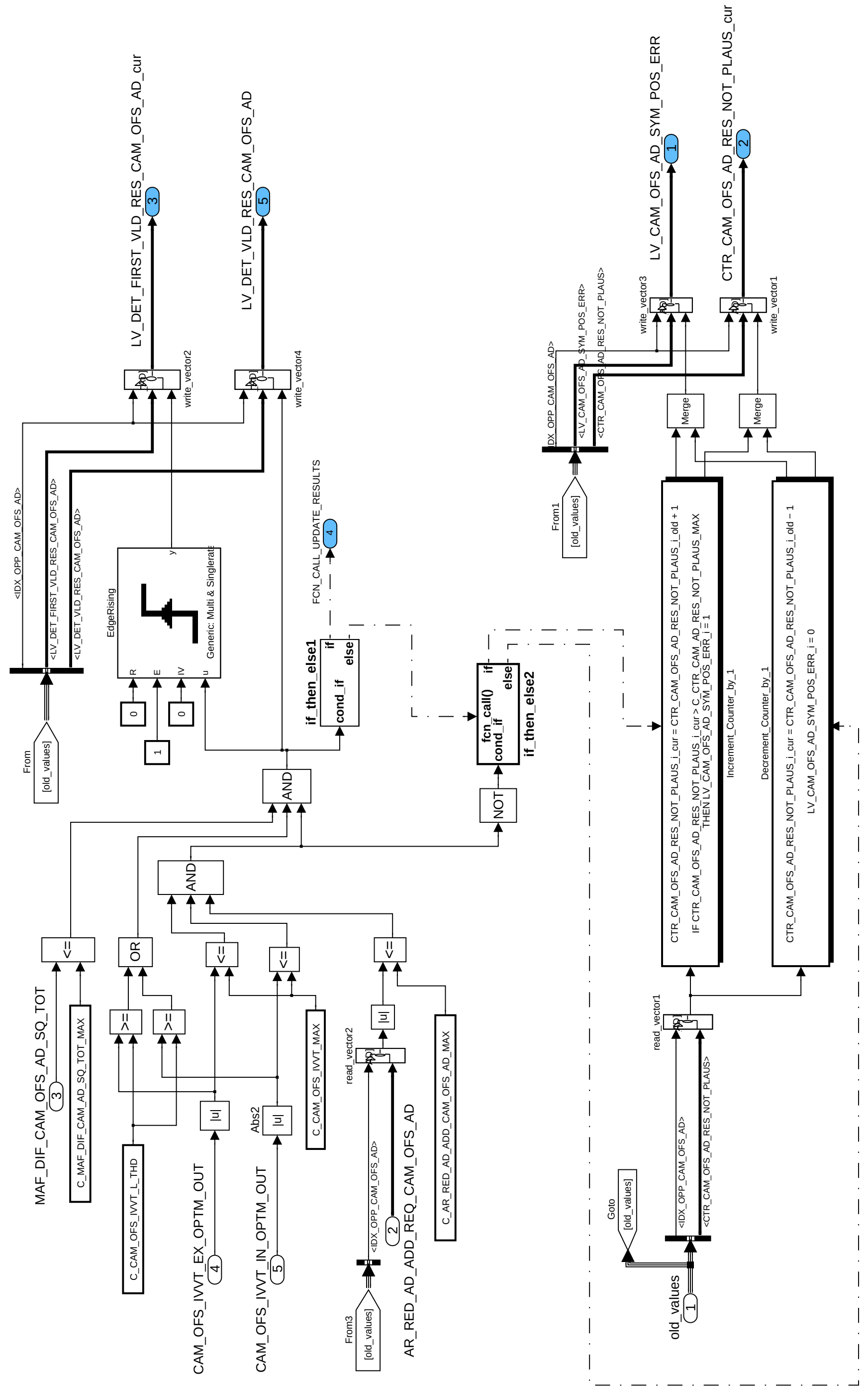


Figure 111 INSY_MANAGCAMAD/ OPERATE/ OPTIMISATION_RES_EVALUATION/ CHECK_FOR_VALID_ADAP_RESULTS

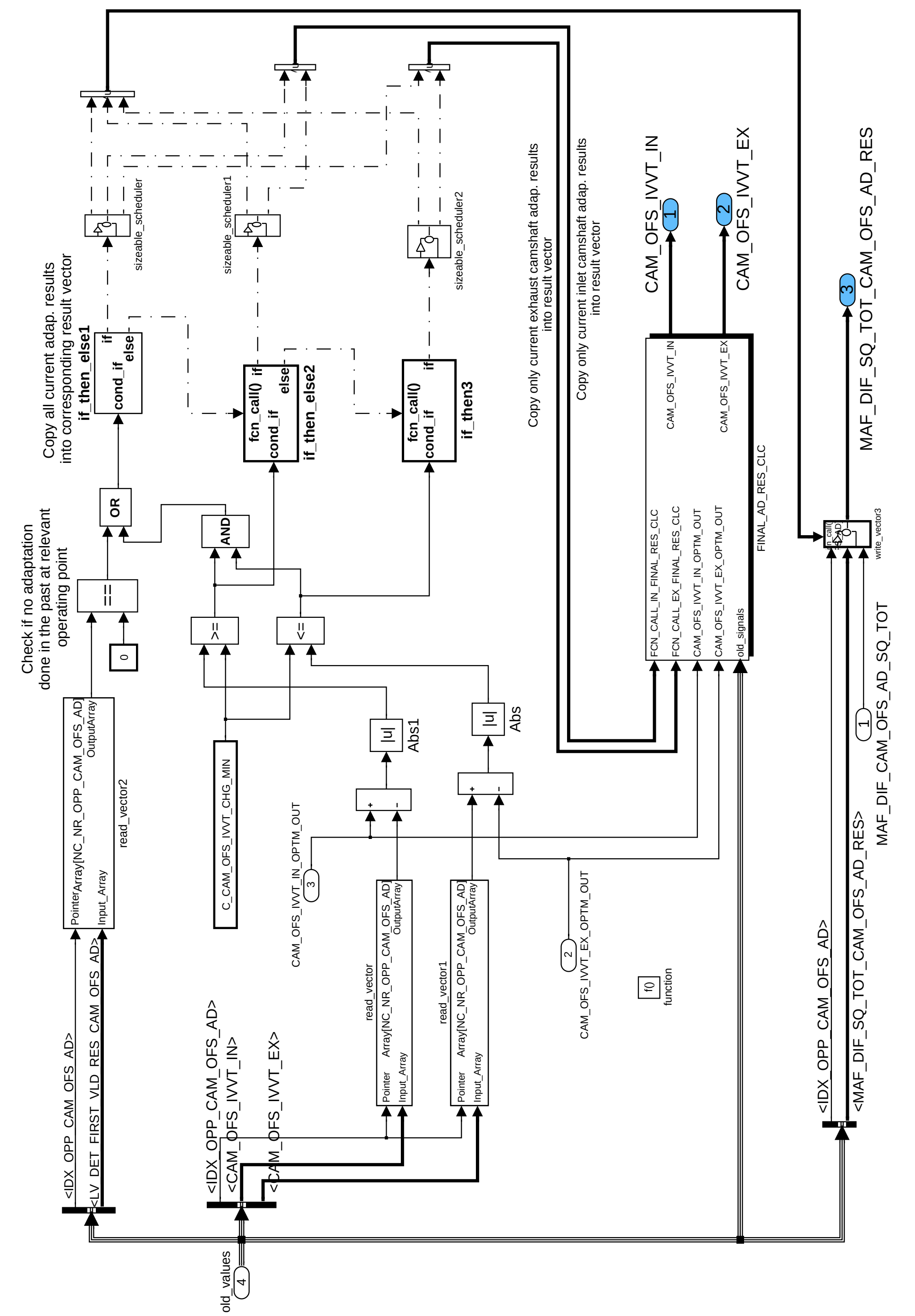


Figure 112 INSY_MANAGCAMAD/ OPERATE/ OPTIMISATION_RES_EVALUATION/ ADAP_RESULTS_EVAL

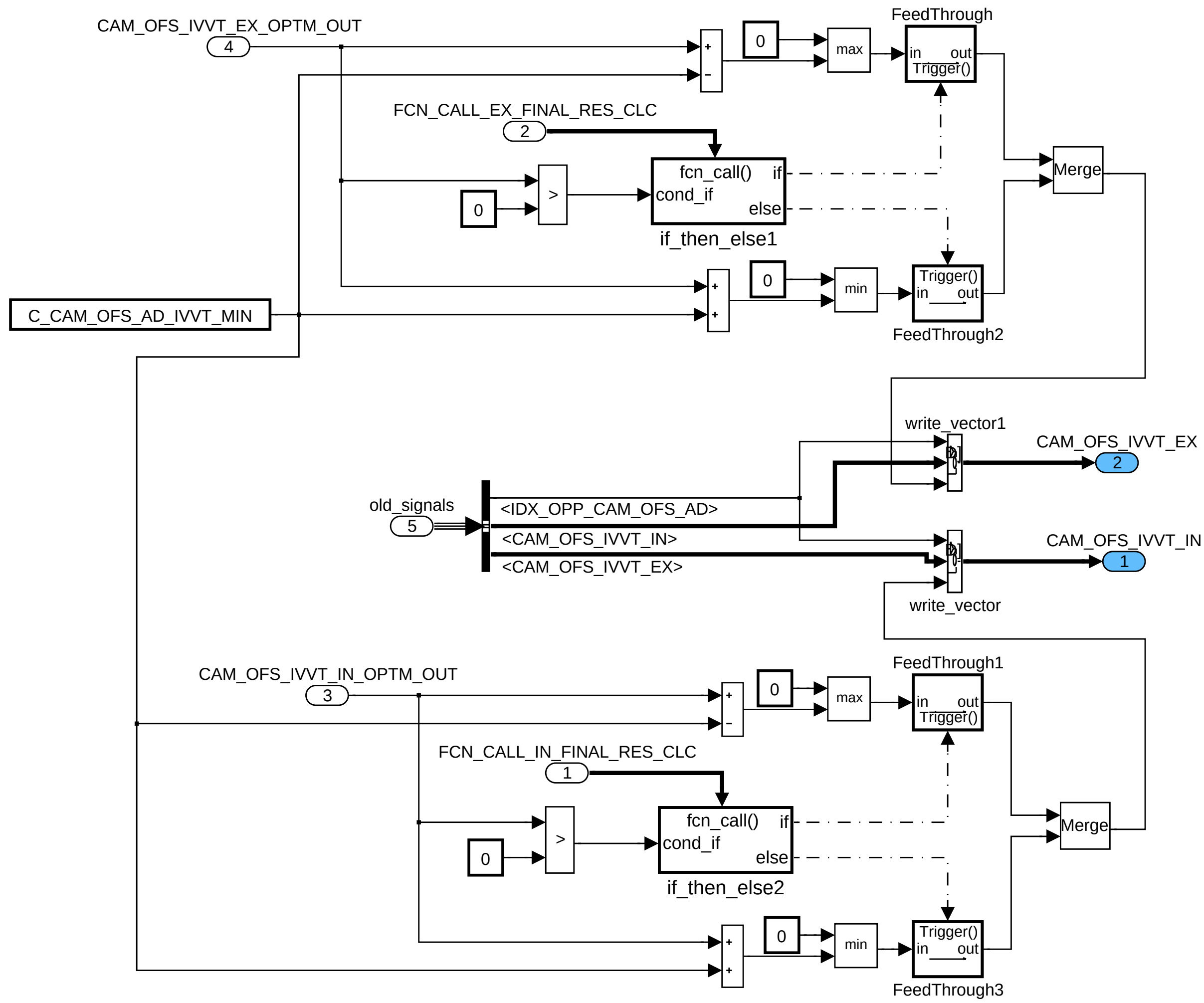


Figure 113 INSY_MANAGCAMAD/ OPERATE/ OPTIMISATION_RES_EVALUATION/ ADAP_RESULTS_EVAL/ FINAL_AD_RES_CLC

9.46 Camshaft offset adaptation manager (Application incidence)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_VOL_SLOP_PORT_CAM_AD	O/V	0...FFFFH	-6.03008833 ... 6.02990431	1.84024E-4	kg/(h*hPa)
Port influence on slope of volumetric efficiency - relevant only for Camshaft offset adaptation functionality					
EFF_VOL_SLOP_VIM_CAM_AD	O/V	0...FFFFH	-6.03008833 ... 6.02990431	1.84024E-4	kg/(h*hPa)
VIM influence on slope of volumetric efficiency - relevant only for Camshaft offset adaptation functionality					
EFF_VOL_OFS_PORT_CAM_AD	O/V	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
Port influence on offset of volumetric efficiency - relevant only for Camshaft offset adaptation functionality					
EFF_VOL_OFS_VIM_CAM_AD	O/V	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
VIM influence on offset of volumetric efficiency - relevant only for Camshaft offset adaptation functionality					
LV_CAM_OFS_AD_EXT_REQ	O/V	0...1H	0...1	1	-
Logical bit indicating external activation of the Camshaft-offset adaptation function					
LV_CAM_POS_ERR_DIAG_ACT	O/V	0...1H	0...1	1	-
Logical bit indicating that Camhaft-offset adaptation is running - used mainly as info for EOL, short trip, workshop test etc.					
LV_ERR_CAM_POS_DIAG[NC_NR_OPP_CAM_OFS_AD]	O/V	0...1H	0...1	1	-
Logical bit indicating detection of Camhaft-position error during workshop test,EOL or short trip					
LV_END_CAM_POS_DIAG_EXT[NC_NR_OPP_CAM_OFS_AD]	O/V	0...1H	0...1	1	-
Logical bit indicating the end of external activation of Camshaft-offset adaptation					
EFF_VOL_OFS_PORT_DRV1_CAM_IN	O/V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_PORT_CAM_AD over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_OFS_VIM_DRV1_CAM_IN	O/V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_VIM_CAM_AD over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_SLOP_PORT_DRV1_CAM_IN	O/V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_PORT_CAM_AD over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_SLOP_VIM_DRV1_CAM_IN	O/V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_VIM_CAM_AD over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_SLOP_PORT_DRV1_CAM_EX	O/V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_PORT_CAM_AD over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_SLOP_VIM_DRV1_CAM_EX	O/V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_VIM_CAM_AD over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_OFS_PORT_DRV1_CAM_EX	O/V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_PORT_CAM_AD over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_OFS_VIM_DRV1_CAM_EX	O/V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_VIM_CAM_AD over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
LV_CAM_OFS_AD_EXT_INH	O/V	0...1H	0...1	1	-
Logical bit indicating project specific inhibition of the Camshaft-offset adaptation functionality					
TPS_AV_CAM_OFS_AD_BEG	V	0...3FFFH	0...119.5	0.00729415	°TPS
TPS value at the begining of the camshaft offset adaptation					

Input data:

LV_AR_RED_AD_ADD_REQ_CAM_CMPL[NC_NR_OPP_CAM_OFS_AD]	N_REQ_CAM_OFS_AD	FAC_PORT_DEAC_MV_REQ_CAM_AD	LV_CAM_OFS_AD_OPTM_END
LV_EFF_VOL_CLC_CAM_OFS_AD	VO_CAM_OFS_AD	LV_LDC_DATA_ACQ_RST	IDX_OPP_CAM_OFS_AD
STATE_CAM_OFS_AD	LV_CAM_OFS_AD_OPTM_RST	LV_CAM_OFS_AD_SYM_POS_ERR[NC_NR_OPP_CAM_OFS_AD]	NC_EFF_VOL_ACR_COR_CONF
NLC_CAM_OFS_AD_EXT_REQ	NC_NR_OPP_CAM_OFS_AD	LV_DC	LV_CAM_POS_DIAG_EXT_REQ
IP_EFF_VOL_SLOP_PORT	IP_EFF_VOL_OFS_PORT	LV_IGK	LV_ERR_MAP
LV_ERR_LOAD_TPS_PLAUS	LV_ERR_VIM_PLAUS	LV_ERR_PORT	LV_ERR_TPS
LV_ERR_MEC_IVVT_IN[NC_NR_CBK_IVVT]	LV_ERR_MEC_IVVT_EX[NC_NR_CBK_IVVT]	LV_ERR_SLV_IVVT_IN[NC_NR_CBK_IVVT]	LV_ERR_SLV_IVVT_EX[NC_NR_CBK_IVVT]
AMP	TIA	LV_CAM_OFS_AD_END	C_VO_OFS_EFF_VOL_GRD_CLC
TPS_AV	N	MAP_DRV1	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_MIN_CAM_OFS_AD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum ambient pressure threshold for activating the camshaft - offset adaptation					
C_CAM_OFS_AD_EXT_REQ_CONF	1	1...3H	1...3	1	-
Calibration constant to configure external adap. request - learning only AR_RED_AD values(=1) or learning AR_RED_AD values and Camshaft-position-error(=2)					
C_MAP_DRV1_MAX_CAM_OFS_AD	1	8000...7FFFH	-82.917525 ... 82.9149945	0.00253044	hPa/ms
Maximum MAP_DRV1 threshold for activating the camshaft offset adaptation					
C_N_MAX_CAM_OFS_AD_BEG	1	0...1FE0H	0...8.16E+3	1	rpm
Maximum engine speed threshold for activating the camshaft offset adaptation					
C_N_MIN_CAM_OFS_AD_BEG	1	0...1FE0H	0...8.16E+3	1	rpm
Minimum engine speed threshold for activating the camshaft offset adaptation					
C_TIA_MAX_CAM_OFS_AD	1	0...FEH	-48...142.5	0.75	°C
Maximum intake air temperature for activating the camshaft - offset adaptation					
C_TIA_MIN_CAM_OFS_AD	1	0...FEH	-48...142.5	0.75	°C
Minimum intake air temperature for activating the camshaft - offset adaptation					
C_TPS_DIF_MAX_CAM_OFS_AD	1	0...3FFFH	0...119.5	0.00729415	°TPS
Maximum TPS difference threshold for activating the camshaft offset adaptation					
LC_CAM_OFS_AD_EXT_REQ_RST	1	0...1H	0...1	1	-
Logical bit to reset results of the adaptation learnt during previous external request run - for testing purposes only					
LC_FIRST_VLD_CAM_OFS_AD	1	0...1H	0...1	1	-
Logical bit to inhibit the camshaft offset adaptation after a valid adaptation result is available					

9.46.1 Function description

Function Description

This specification is used to implement customer specific requirements on the adaptation. It is divided into two parts with the first part used to coordinate the external activation for e.g. during End-of-Line test or workshop tests etc. The request for such activations can be done either through an externally connected tester/scan-tool or by other ECU functionalities. The second part involved the project specific calculation of the data required for the determination of the adaptation specific MAF_THR and MAF_CYL i.e. influence of the VIM and PORT flaps on the volumetric efficiency.

Application Condition

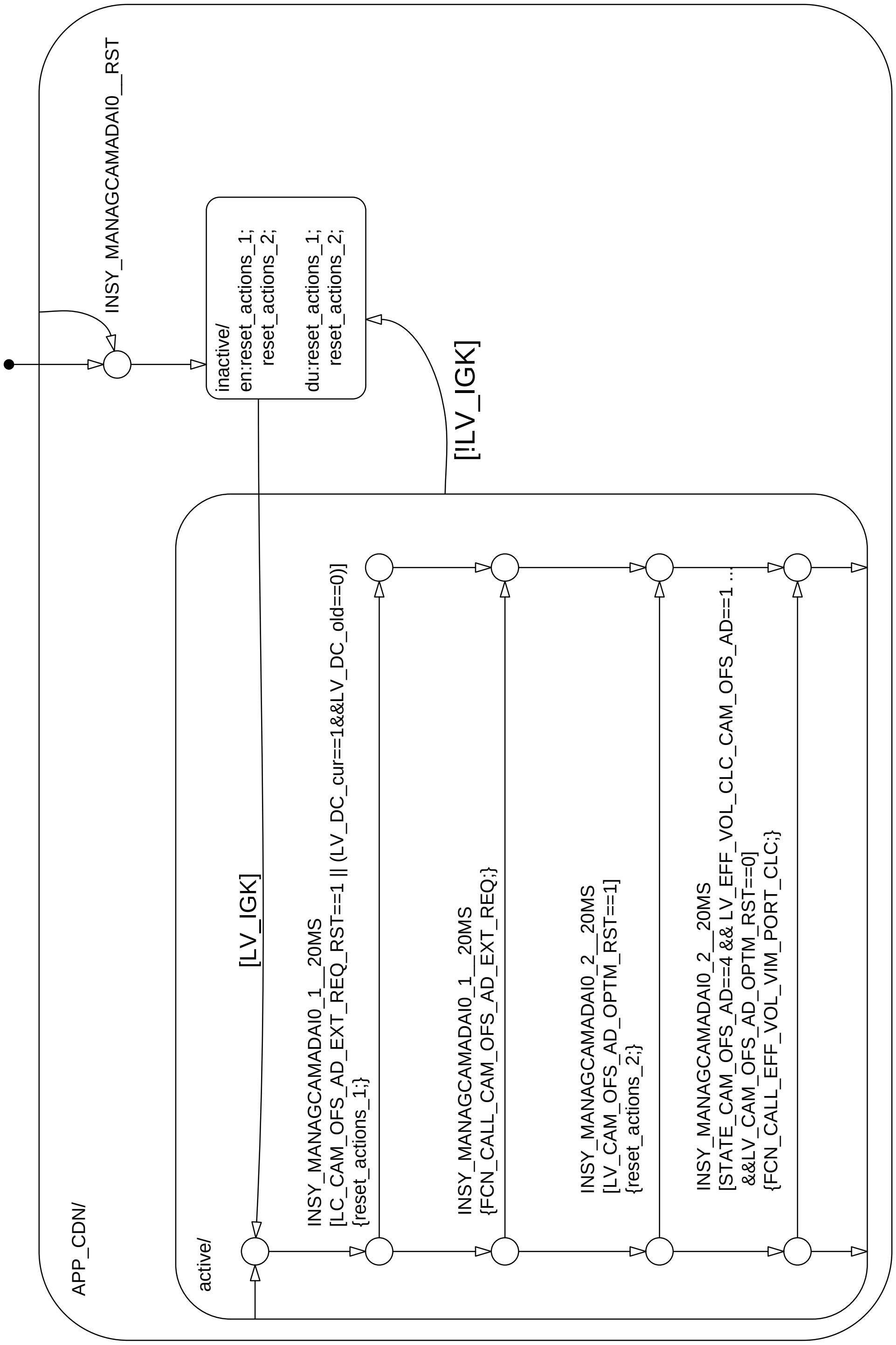


Figure 114 INSY_M909R/ APP_CDN/ Chart1

Function Description

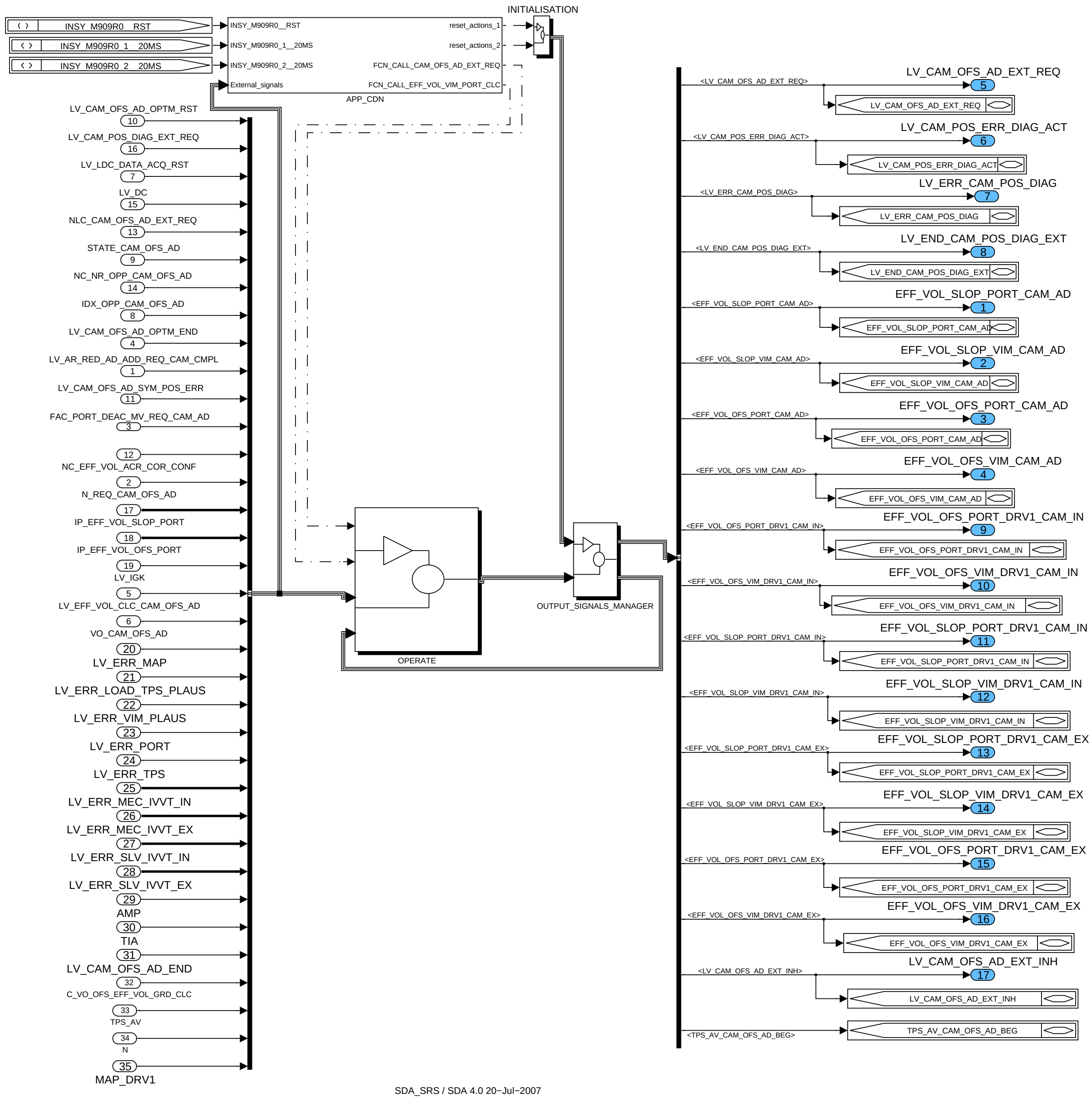


Figure 115 INSY_M909R

9.46.1.1 Initialisation

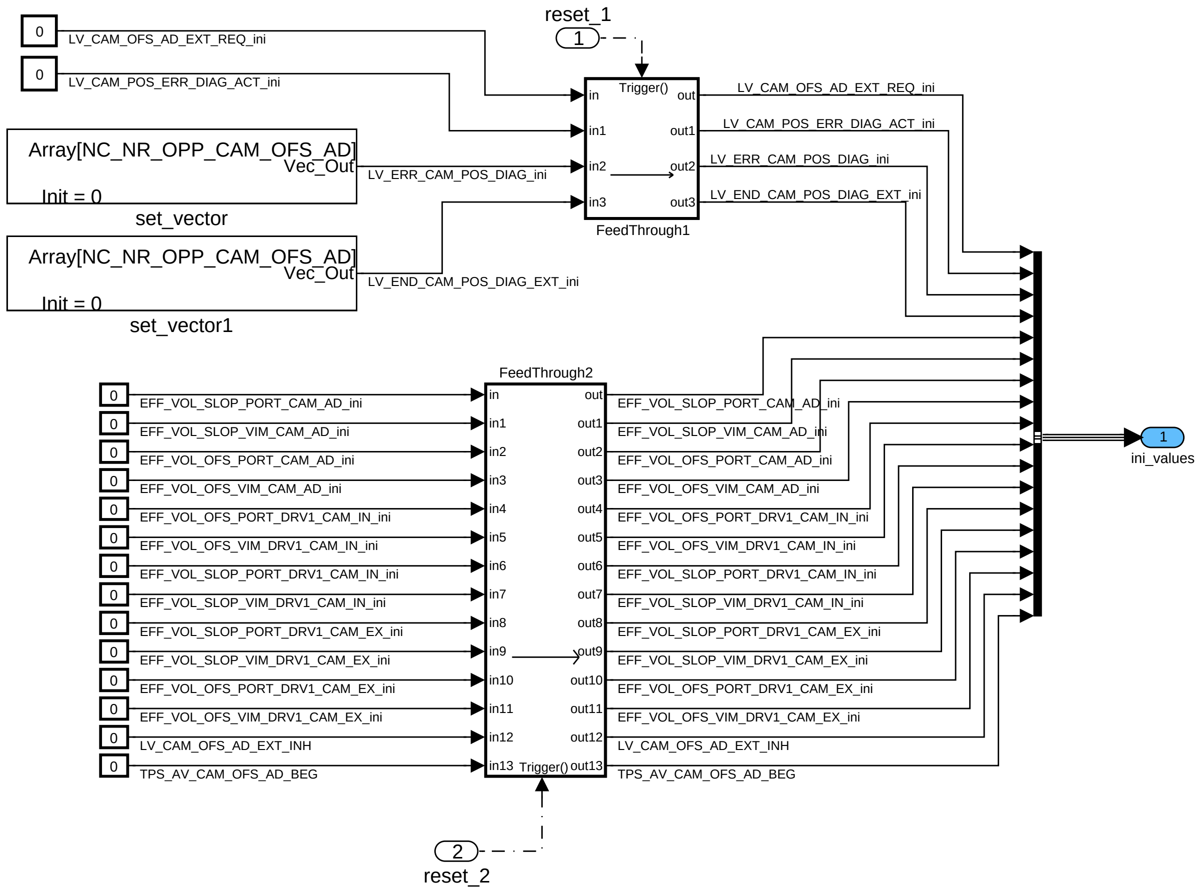


Figure 116 INSY_M909R/ INITIALISATION

9.46.1.2 Formula section

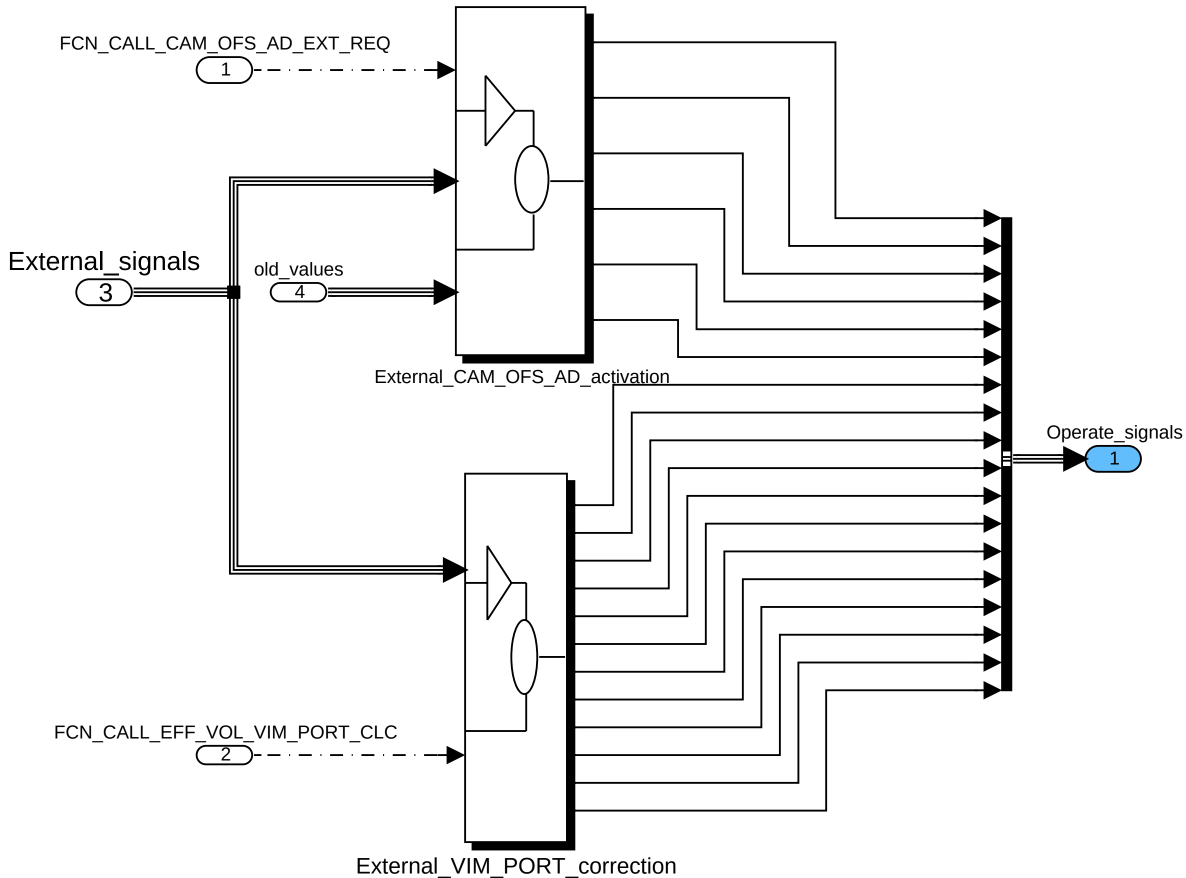


Figure 117 INSY_M909R/ OPERATE

Coordination of external request for Camshaft-offset adaptation

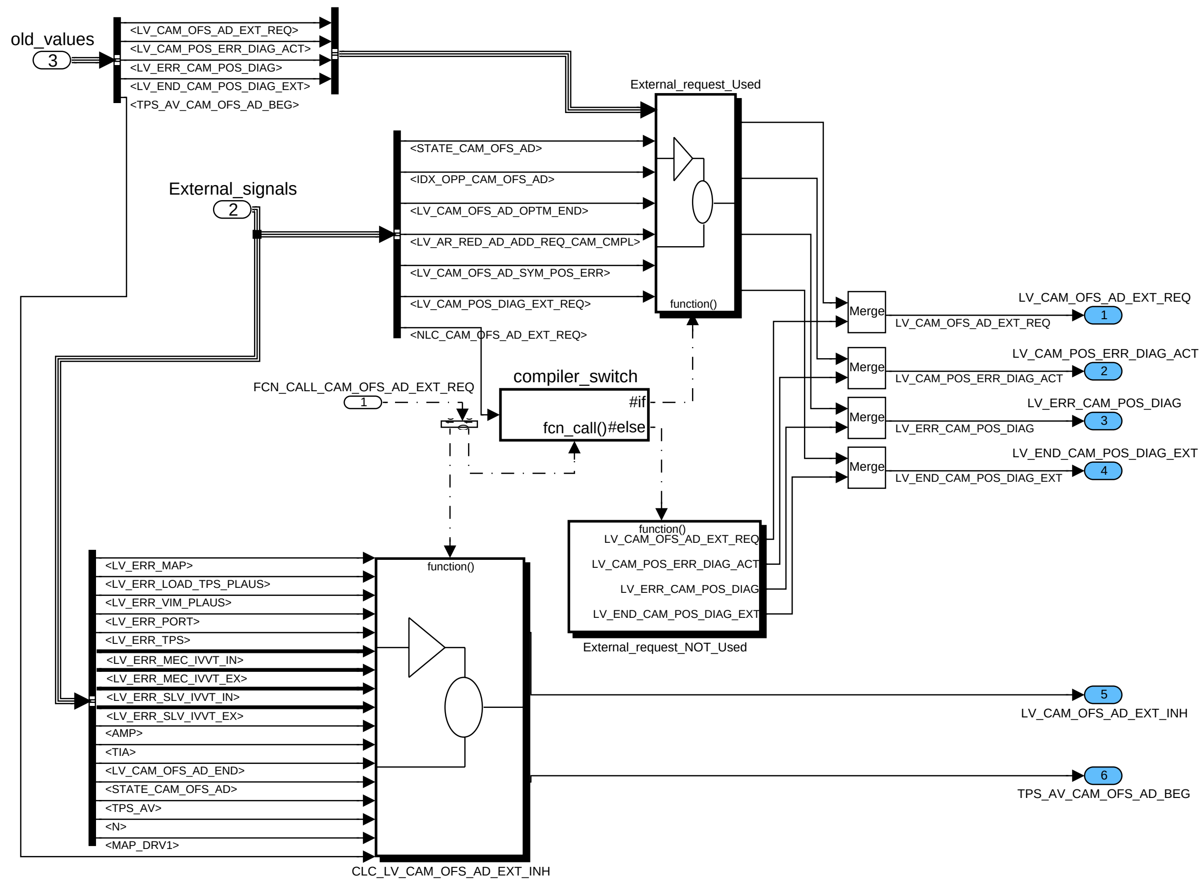


Figure 118 INSY_M909R/ OPERATE/ External_CAM_OFS_AD_activation

Calculation of LV_CAM_OFS_AD_EXT_INH

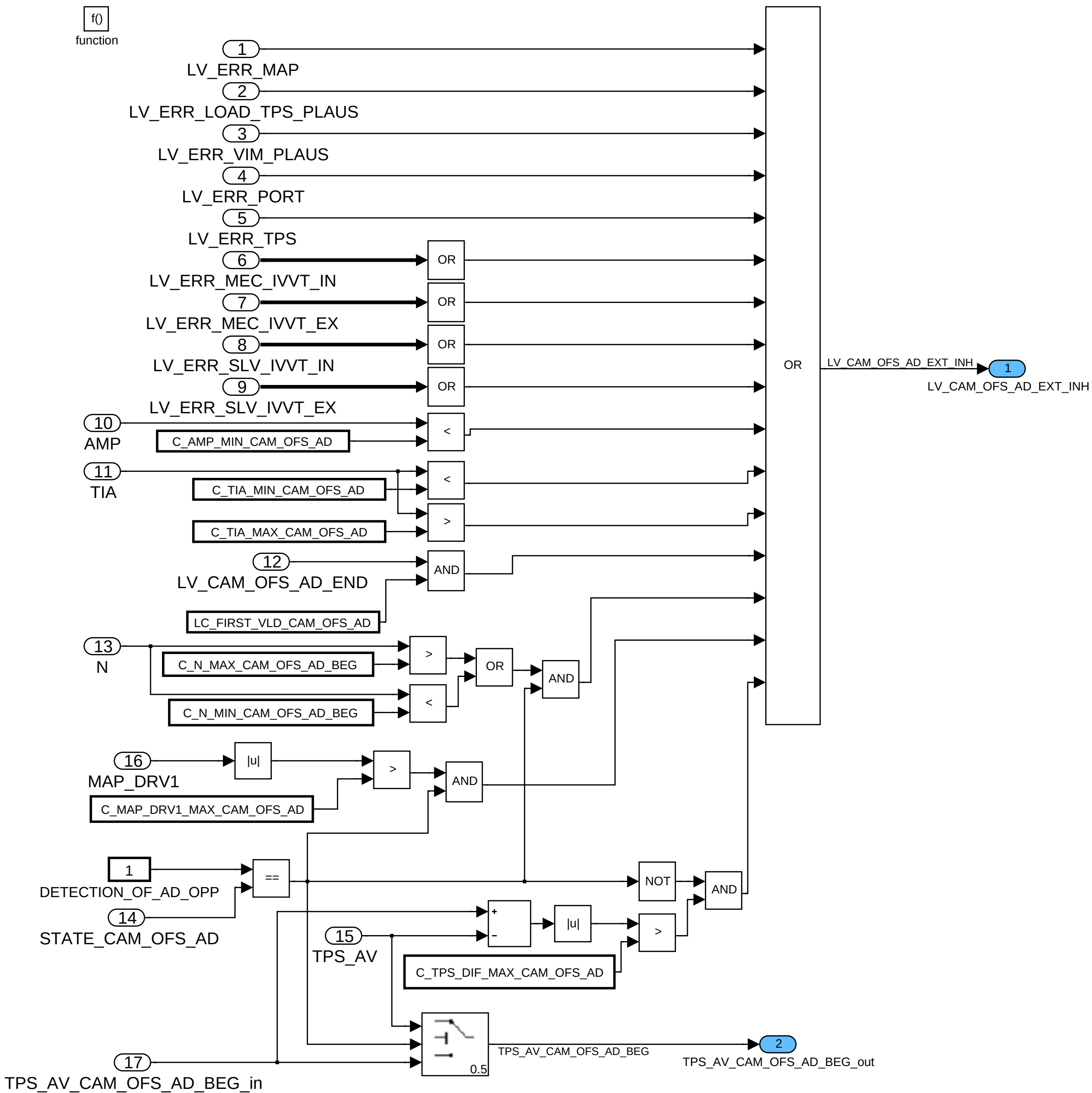


Figure 119 INSY_M909R/ OPERATE/ External_CAM_OFS_AD_activation/ CLC_LV_CAM_OFS_AD_EXT_INH

No external activation of Camshaft-offset adaptation present in system

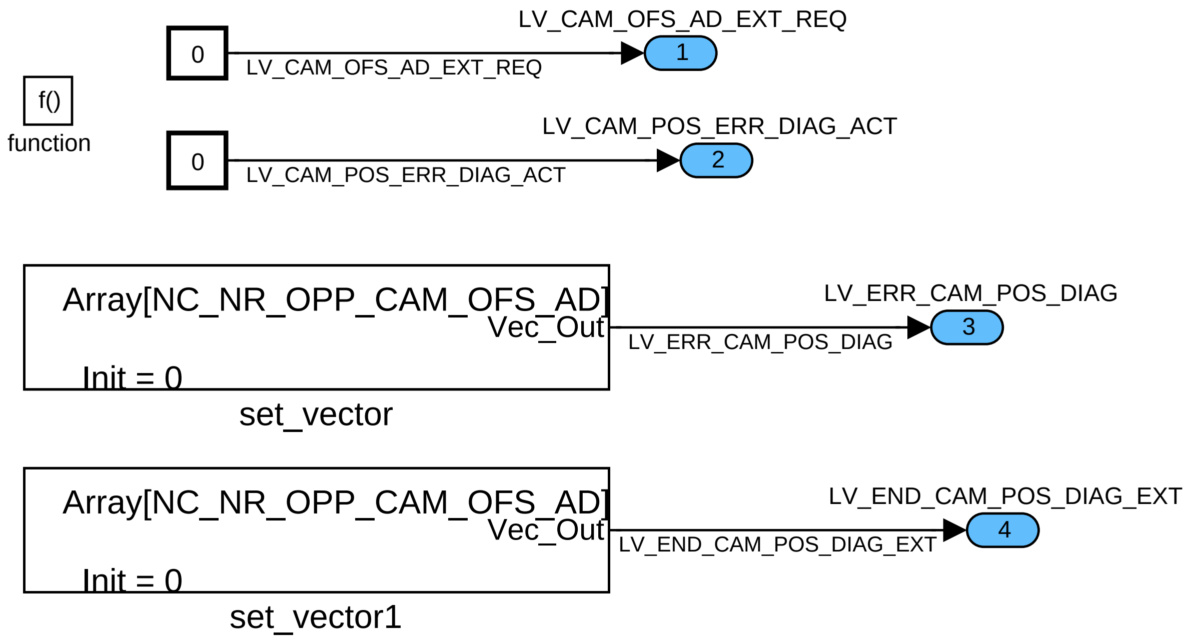


Figure 120 INSY_M909R/ OPERATE/ External_CAM_OFS_AD_activation/ External_request_NOT_Used

Calculation when external request is used

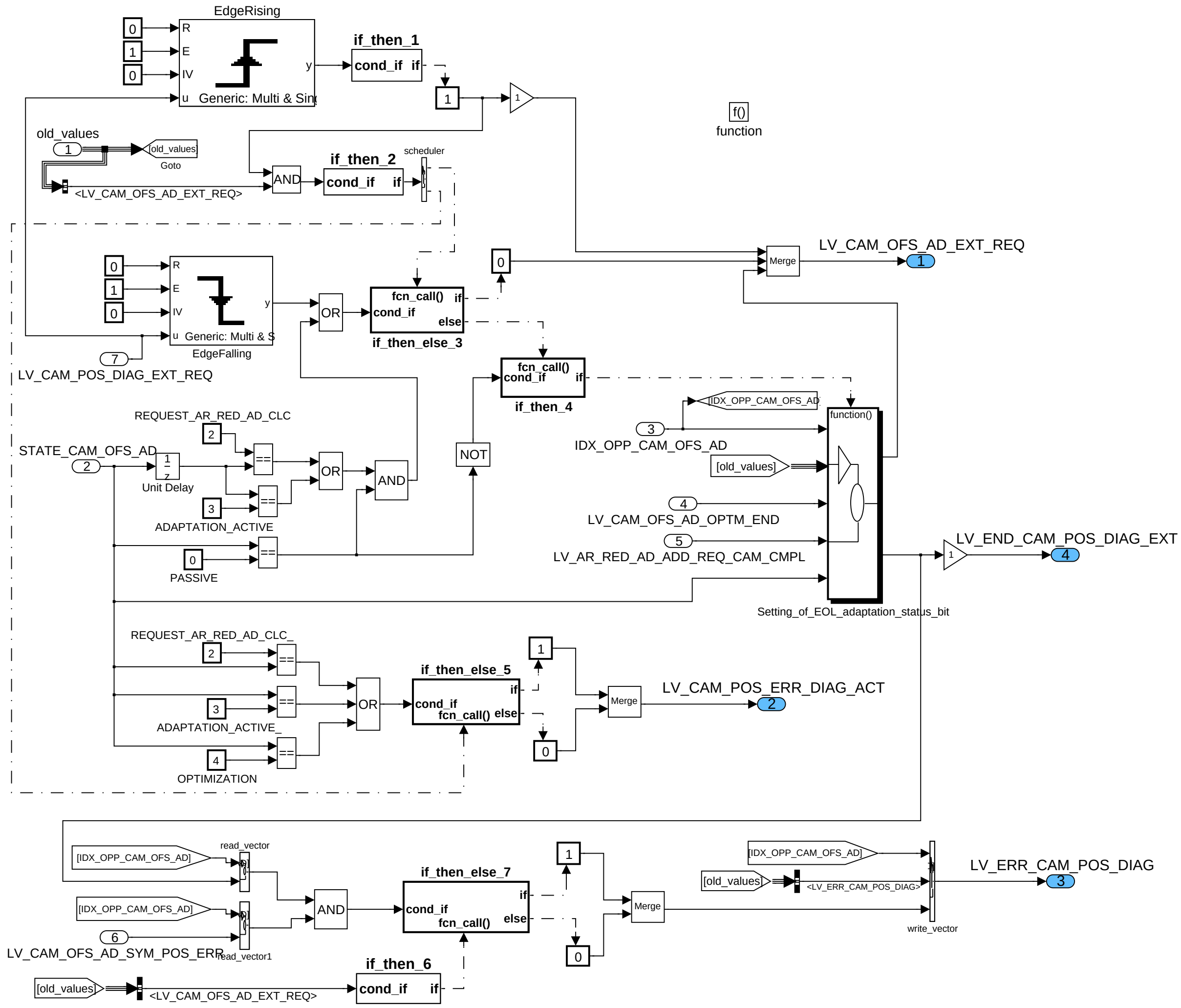


Figure 121 INSY_M909R/ OPERATE/ External_CAM_OFS_AD_activation/ External_request_Used

Calculation of LV_CAM_OFS_AD_EXT_REQ and LV_END_CAM_POS_DIAG_EXT

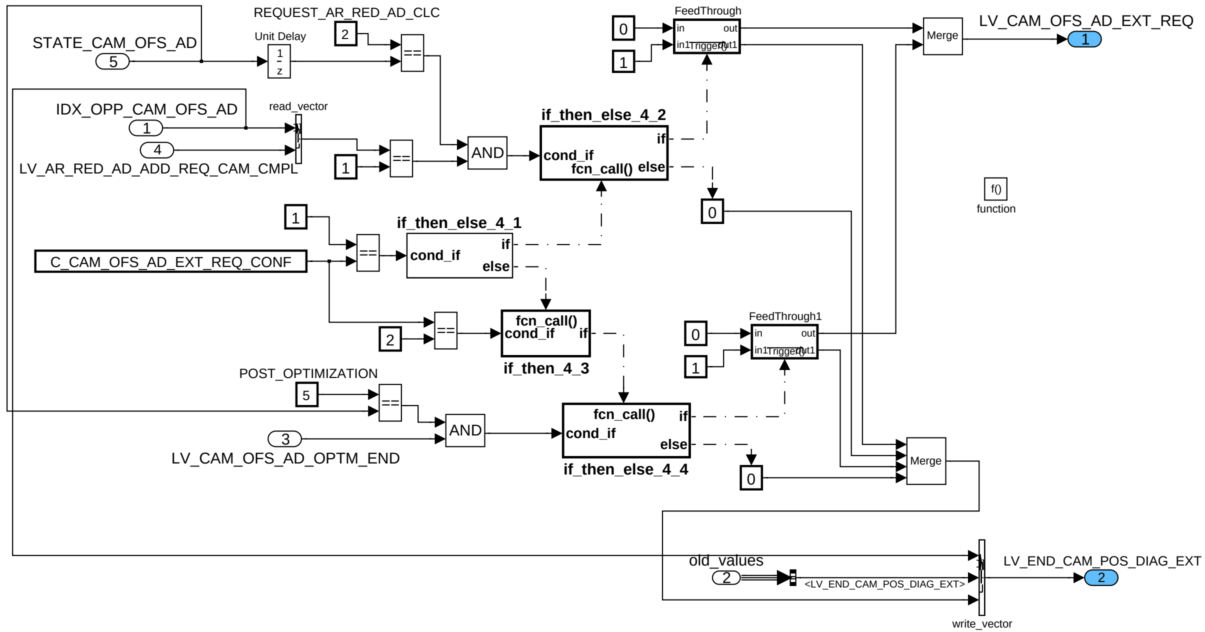


Figure 122 INSY_M909R/ OPERATE/ External_CAM_OFS_AD_activation/ External_request_Used/ Setting_of_EOL_adaptation_status_bit

Calculation of VIM&PORT correction of Volumetric efficiency

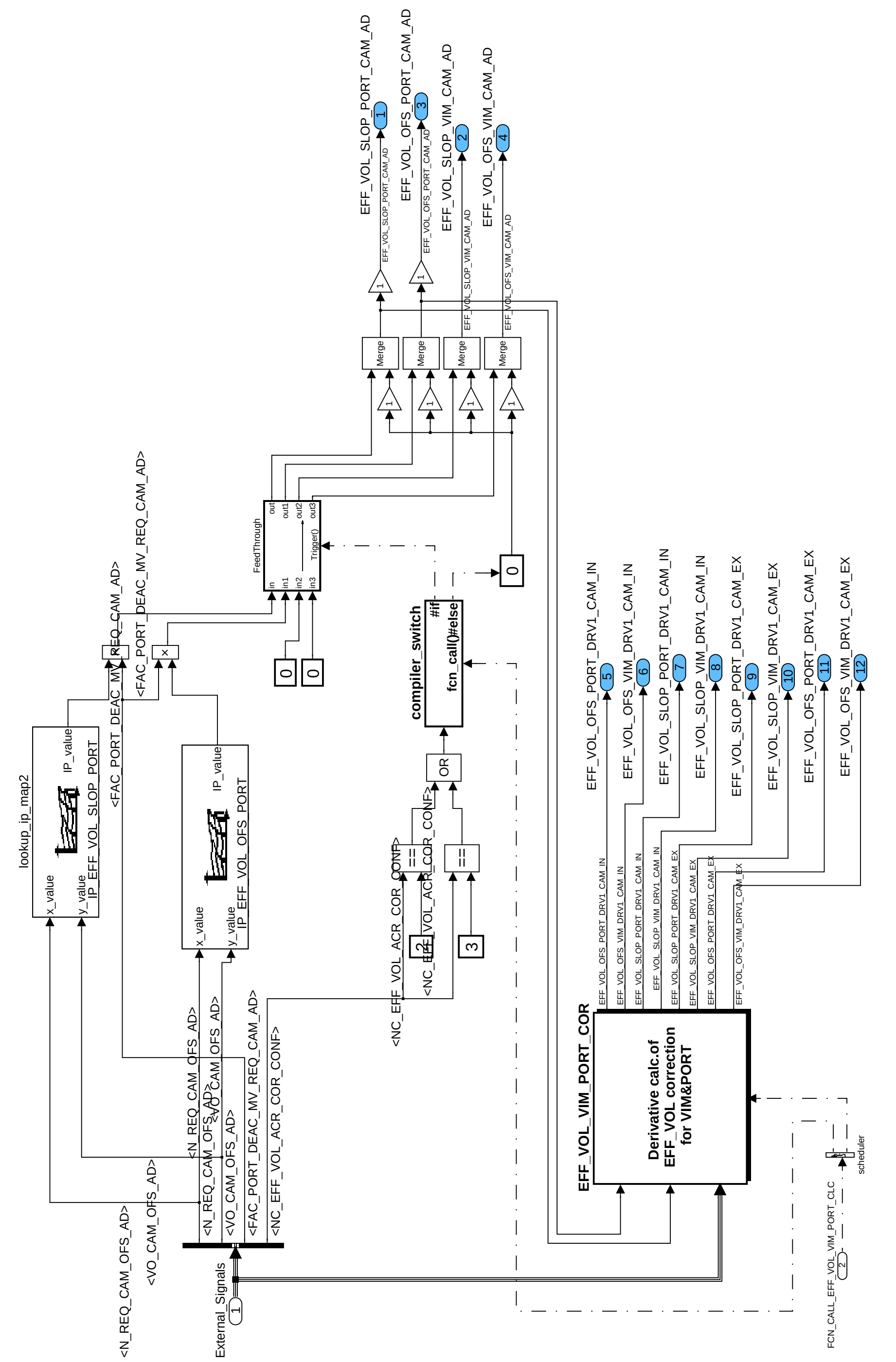


Figure 123 INSY_M909R/ OPERATE/ External_VIM_PORT_correction

Calculation of Effective volume for VIM_PORT correction

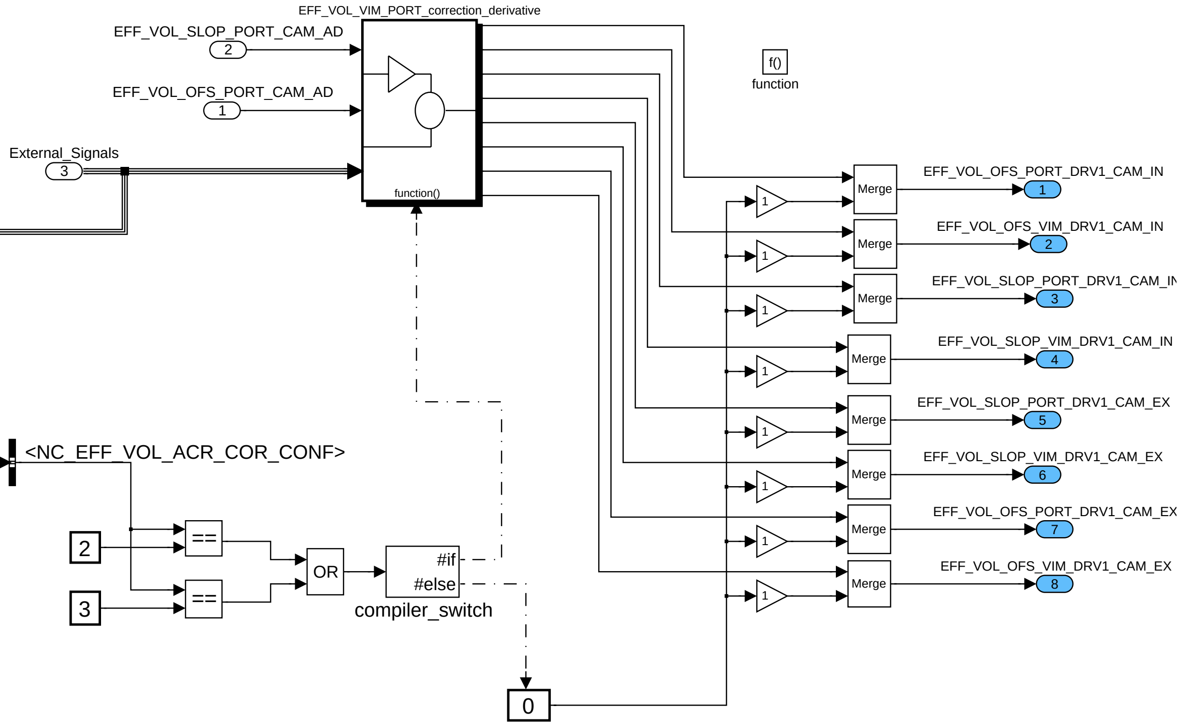


Figure 124 INSY_M909R/ OPERATE/ External_VIM_PORT_correction/ EFF_VOL_VIM_PORT_COR

Calculation of correction derivatives of effective volume for VIM_PORT correction

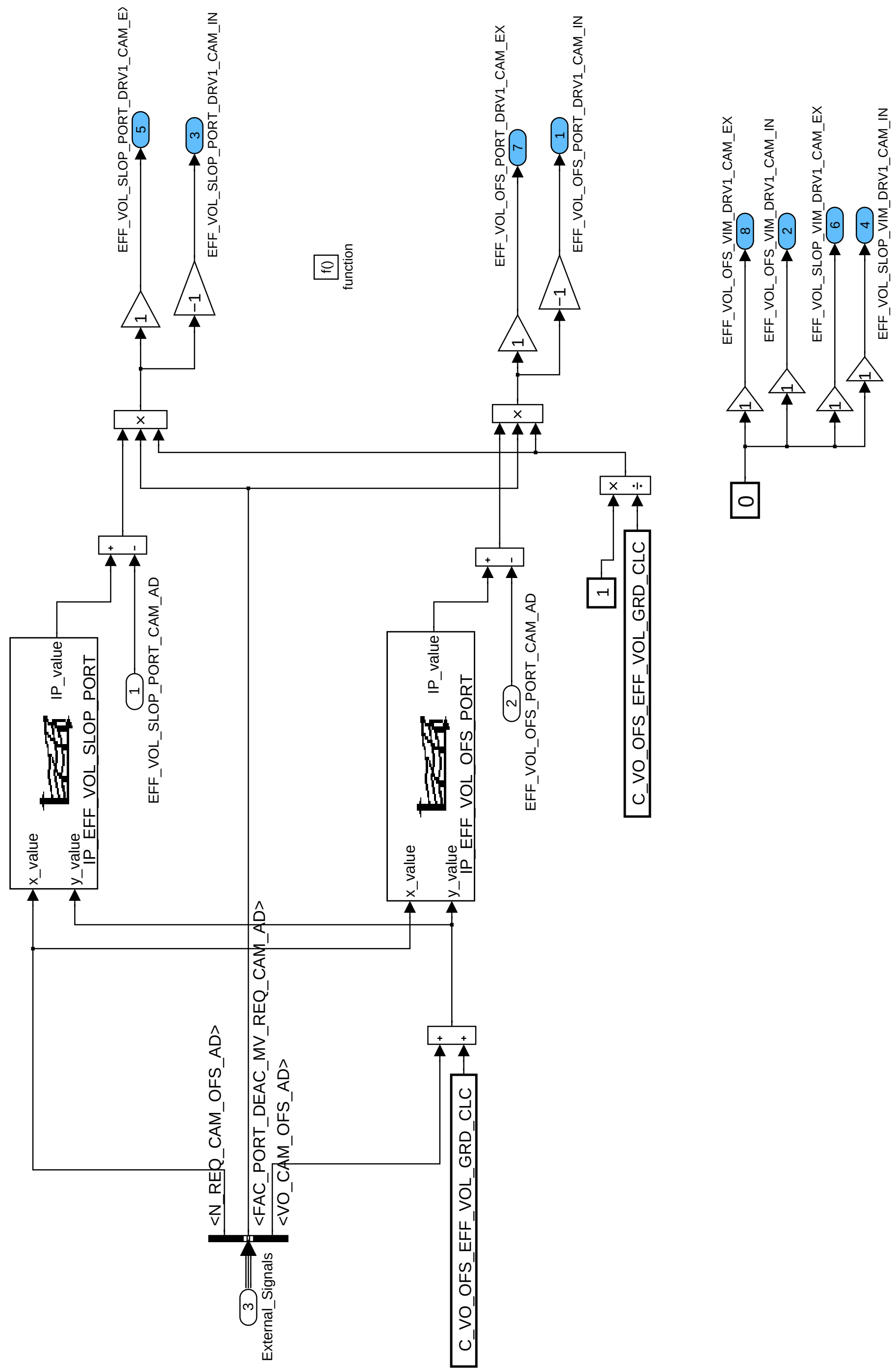


Figure 125 INSY_M909R/ OPERATE/ External_VIM_PORT_correction/ EFF_VOL_VIM_PORT_COR/ EFF_VOL_VIM_PORT_correction_derivative

9.47 Activation conditions and Data Acquisition for Camshaft offset adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CAM_SP_ADJ_TOT_END	O/V	0...1H	0...1	1	-
Logical bit indicating end of camshaft adjustment and signal acquisition					
LV_LDC_CAM_OFS_AD_ACT	O/V	0...1H	0...1	1	-
Logical bit indicating status of limited dynamics check which is done before start of adaptation					
LV_CAM_SP_ADJ_SIG_SAVE_END	O/V	0...1H	0...1	1	-
Logical bit indicating end of signal acquisition					
CTR_CYCNR_SIG_REC_CAM_OFS_AD	O/V	0...FFFFH	0...6.5535E+4	1	-
Counter for number of signal acquisition cycles					
CTR_DLY_AR_RED_AD_REQ_CAM	O/V	0...FFFFH	0...6.5535E+4	1	-
Delay time before activation of AR_RED_AD calculation for CAM_OFS_AD functionality					
LV_CAM_SP_ADJ_CAM_OFS_AD	O/V	0...1H	0...1	1	-
Logical bit indicating to IVVT functionalities that camshaft adjustment for adaptation purposes is ongoing					
CAM_SP_CAM_OFS_AD_IVVT_IN[NC_NR_CBK_IVVT]	O/V	0...FFH	60...155.625	0.375	°CRK
Inlet Camshaft position setpoint for adaptation					
CAM_SP_CAM_OFS_AD_IVVT_EX[NC_NR_CBK_IVVT]	O/V	0...FFH	-40.125 ... -135.75	-0.375	°CRK
Exhaust Camshaft position setpoint for adaptation					
CAM_IN_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD][NC_NR_CBK_IVVT]	O/V/S	0...FFH	60...155.625	0.375	°CRK
Mean value of measured CAM_IN					
CAM_EX_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD][NC_NR_CBK_IVVT]	O/V/S	0...FFH	-40.125 ... -135.75	-0.375	°CRK
Mean value of measured CAM_EX					
N_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...1FE0H	0...8.16E+3	1	rpm
Mean value of measured engine speed at the different camshaft positions					
EFF_VOL_SLOP_AMP_COR_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Mean value of measured EFF_VOL_SLOP_AMP_COR at the different camshaft positions					
EFF_VOL_TEMP_COR_MMV_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Mean value of measured EFF_VOL_TEMP_COR_MMV at the different camshaft positions					
EFF_VOL_OFS_AMP_COR_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...1.99996948	3.05176E-5	-
Mean value of measured EFF_VOL_OFS_AMP_COR at the different camshaft positions					
MAP_MES_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Mean value of measured MAP_MES at the different camshaft positions					
PRS_EX_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Mean value of measured PRS_EX at the different camshaft positions					
MAF_THR_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Mean value of calculated MAF_THR_CAM_OFS_AD at the different camshaft positions					
AR_RED_AD_ADD_REQ_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	O/V	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Result of AR_RED_AD calculation for CAM_OFS_AD functionality purpose					
FAC_VIM_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFH	0...0.99609375	0.00390625	-
Measured FAC_VIM during the signal acquisition phases of camshaft-offset adaptation					
FAC_PORT_DEAC_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	O/V	0...FFFFH	0...0.99998474	1.52588E-5	-
Measured FAC_PORT_DEAC_MV during the signal acquisition phases of camshaft-offset adaptation					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_AR_RED_AD_ADD_REQ_CAM_CMPL[NC_NR_OPP_CAM_OFS_AD]	O/V	0...1H	0...1	1	-
Logical bit indicating AR_RED_AD calculation for CAM_OFS_AD functionality has been completed					
LV_AR_RED_AD_ADD_REQ_CAM_TMP	O/V	0...1H	0...1	1	-
Logical bit indicating status of AR_RED_AD request from CAM_OFS_AD function					
AR_RED_AD_ADD_REQ_CAM_INTER	O/V	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Temporary variables in AR_RED_AD_ADD calculation for CAM_OFS_AD functionality					
AR_RED_AD_FAC_REQ_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	O/V	8000...7FFFH	-0.5 ... 0.49998474	1.52588E-5	-
Result of multiplicative AR_RED_AD calculation for CAM_OFS_AD functionality purpose only					
AR_RED_AD_FAC_REQ_CAM_INTER	O/V	8000...7FFFH	-0.5 ... 0.49998474	1.52588E-5	-
Temporary variable in AR_RED_AD_FAC calculation for CAM_OFS_AD functionality					
CTR_SP_REQ_CAM_SP_ADJ	O/V	0...CH	0...12	1	-
Counter over number of camshaft-positions being used for adaptation					
LV_LDC_CAM_SP_ADJ	V	0...1H	0...1	1	-
Logical bit indicating status of limited dynamics check which is done during camshaft adjustment					
CTR_DLY_TPS_MAP_LDC_CHK	V	0...FFH	0...255	1	-
Counter for delay time before setting limited dynamics status of TPS_AV, MAP_MES and N					
TPS_CAM_OFS_AD_TMP	V	0...3FFFH	0...119.5	0.00729415	°TPS
Temporary variable for TPS_AV limited dynamics check					
MAP_MES_CAM_OFS_AD_TMP	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Temporary variable for MAP_MES limited dynamics check					
LV_CAM_SP_EX_ADJ_ACK[NC_NR_CBK_IVVT]	V	0...1H	0...1	1	-
Logical bit indicating if the desired exhaust camshaft position on each cyl. bank has actually been reached					
LV_CAM_SP_IN_ADJ_ACK[NC_NR_CBK_IVVT]	V	0...1H	0...1	1	-
Logical bit indicating if the desired inlet camshaft position on each cyl. bank has actually been reached					
MAF_THR_SUM_CAM_OFS_AD	V	0...FFFFFFFFH	0...1.34218E+8	0.03125	kg/h
Sum of measured MAF_THR values over C_CYCNR_SIG_REC_MAX recurrences					
N_CAM_OFS_AD_TMP	V	0...1FE0H	0...8.16E+3	1	rpm
Temporary variable for engine speed limited dynamics check					

Input data:

LV_DET_VLD_RES_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	LV_END_CAM_POS_DIAG_EXT[NC_NR_OPP_CAM_OFS_AD]	LV_CAM_OFS_AD_EXT_REQ	N
PRS_EX	PUT[NC_MAP_SENS_NR]	CAM_IN[NC_NR_CBK_IVVT]	CAM_EX[NC_NR_CBK_IVVT]
AR_RED_BAS	MAP_MES	FLOW_CPS	STATE_AR_RED_AD_ADD
MAF_MDL_CON_1	EFF_VOL_OFS_AMP_COR	EFF_VOL_TEMP_COR_MMV	EFF_VOL_SLOP_AMP_COR
TPS_AV	FAC_VIM_TMP	NC_NR_CBK_IVVT	FAC_PORT_DEAC_MV_TMP
NC_NR_OPP_CAM_OFS_AD	NC_NR_SP_REQ_CAM_OFS_AD	AR_RED_DIF_REL	LV_LDC_DATA_ACQ_RST
STATE_CAM_OFS_AD	LV_CAM_OFS_AD_OPP_CDN	IDX_OPP_CAM_OFS_AD	LV_AR_RED_AD_REQ_CAM_RST
IDX_MAP_OPP_CAM_OFS_AD	ID_MAF_FAC_OFS	ID_MAF_FAC_SLOP	AR_RED_AD_FAC_COR
AR_RED_AD_ADD			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AR_RED_AD_ADD_REQ_CAM_MAN_SP	1	8000...7FFFH	-29.2968745 ... 29.2959805	8.9407E-4	cm²
Manual value of AR_RED_AD_ADD for Camshaft-offset adaptation function only					
C_AR_RED_AD_FAC_REQ_CAM_MAN_SP	1	8000...7FFFH	-0.5 ... 0.49998474	1.52588E-5	-
Manual value of AR_RED_AD_FAC for Camshaft offset adaptation function only					
C_AR_RED_AD_REQ_CAM_RST	1	0...4H	0...4	1	-
Calibration switch to set/reset AR_RED_AD_ADD_REQ_CAM_OFS_AD and LV_AR_RED_AD_ADD_REQ_CAM_CMPL					
C_CAM_SP_ADJ_HYS	1	0...FFH	0...95.625	0.375	°CRK
Camshaft-position hysteresis					
C_CAM_SP_IVVT_EX_CHG_STEP	1	0...FFH	0...95.625	0.375	°CRK
Increment steps for exhaust Camshaft adjustment ramp					
C_CAM_SP_IVVT_IN_CHG_STEP	1	0...FFH	0...95.625	0.375	°CRK
Increment steps for inlet Camshaft adjustment ramp					
C_CTR_CAM_POS_ACK_DLY_MAX	1	0...FFFFH	0...6.5535E+4	1	-
Maximum value of Counter to abort adaptation if it is waiting too long for the camshaft to reach the desired position					
C_CTR_DLY_AR_RED_AD_CAM_MAX_1	1	0...FFFFH	0...6.5535E+4	1	-
Maximum value of delay counter before AR_RED_AD calc. is requested by CAM_OFS_AD functionality					
C_CTR_DLY_AR_RED_AD_CAM_MAX_2	1	0...FFFFH	0...6.5535E+4	1	-
Maximum value of delay counter before AR_RED_AD_ADD is reset					
C_CTR_DLY_AR_RED_AD_CAM_MAX_3	1	0...FFFFH	0...6.5535E+4	1	-
Delay counter to wait until AR_RED_DIF_REL has reached a stationary value.					
C_CTR_DLY_TPS_MAP_LDC_MAX	1	0...FFH	0...255	1	-
Delay before starting limited dynamics check for MAP_MES and TPS_AV at each camshaft position used for adaptation					
C_CYCNR_SIG_REC_MAX	1	0...FFFFH	0...6.5535E+4	1	-
Number of recurrencies for signal acquisition during camshaft-offset adaptation					
C_MAP_MES_DELTA_MAX_CAM_AD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Threshold for MAP_MES deviation during determination of limited dynamics					
C_MAP_MES_DELTA_MAX_CAM_AD_2	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Threshold for MAP_MES limited dynamics check during AR_RED_AD calculation					
C_NR_SP_REQ_CAM_OFS_AD	1	1...9H	1...9	1	-
Number of camshaft positions being used for adaptation					
C_N_DELTA_MAX_CAM_AD	1	0...1FE0H	0...8.16E+3	1	rpm
Threshold for N deviation during determination of limited dynamics					
C_N_DELTA_MAX_CAM_AD_2	1	0...1FE0H	0...8.16E+3	1	rpm
Threshold for N limited dynamics check during AR_RED_AD calculation					
C_TPS_DELTA_MAX_CAM_AD	1	0...3FFFH	0...119.5	0.00729415	°TPS
Maximum allowed deviation between TPS_AV and TPS_AV_CAM_OFS_AD_TMP during stationary conditions					
LC_AR_RED_AD_REQ_CAM_OFS_AD	1	0...1H	0...1	1	-
Logical bit to reset the reduced area adaptation during camshaft offset adaptation					
LC_AR_RED_AD_RST_ACT_CAM_AD	1	0...1H	0...1	1	-
Logical bit to activate reset of of AR_RED_AD from CAM_OFS_AD when LV_AR_RED_AD_REQ_CAM_RST = 1					
LC_CAM_SP_ADJ_LDC_TEST	1	0...1H	0...1	1	-
Logical bit for testing purposes to switch value of LV_LDC_CAM_SP_ADJ between calculated value and fixed value of 1					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_CAM_SP_EX_AR_RED_AD_CAM_AD	NC_NR_OPP_CAM_OFS_ADxNC_NR_MAP_OPP_CAM_OFS_AD	0...FFH	-40.125 ... -135.75	-0.375	°CRK
LDPM_NR_OPP_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	1...4H	1...4	1	-
LDPM_NR_MAP_OPP_CAM_AD	NC_NR_MAP_OPP_CAM_OFS_AD	1...5H	1...5	1	-
Exhaust camshaft position required for AR_RED_AD_ADD calculation within Camshaft-offset adaptation functionality					
ID_CAM_SP_IN_AR_RED_AD_CAM_AD	NC_NR_OPP_CAM_OFS_ADxNC_NR_MAP_OPP_CAM_OFS_AD	0...FFH	60...155.625	0.375	°CRK
LDPM_NR_OPP_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	1...4H	1...4	1	-
LDPM_NR_MAP_OPP_CAM_AD	NC_NR_MAP_OPP_CAM_OFS_AD	1...5H	1...5	1	-
Inlet camshaft position required for AR_RED_AD_ADD calculation within Camshaft-offset adaptation functionality					
ID_CAM_SP_IVVT_EX_CAM_OFS_AD[NC_NR_CBK_IVVT]	NC_NR_OPP_CAM_OFS_ADxNC_NR_SP_REQ_CAM_OFS_AD	0...FFH	-40.125 ... -135.75	-0.375	°CRK
LDPM_NR_OPP_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	1...4H	1...4	1	-
LDPM_CTR_SP_REQ_CAM_SP_ADJ	NC_NR_SP_REQ_CAM_OFS_AD	1...CH	1...12	1	-
Exhaust camshaft position required for camshaft-offset adaptation					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_CAM_SP_IVVT_IN_CAM_OFS_AD[NC_NR_CBK_IVVT]	NC_NR_OPP_CAM_OFS_ADxNC_NR_SP_REQ_CAM_OFS_AD	0...FFH	60...155.625	0.375	°CRK
LDPM_NR_OPP_CAM_OFS_AD	NC_NR_OPP_CAM_OFS_AD	1...4H	1...4	1	-
LDPM_CTR_SP_REQ_CAM_SP_ADJ	NC_NR_SP_REQ_CAM_OFS_AD	1...CH	1...12	1	-
Inlet camshaft position required for camshaft-offset adaptation					

9.47.1 INSY_M908F

Application Condition

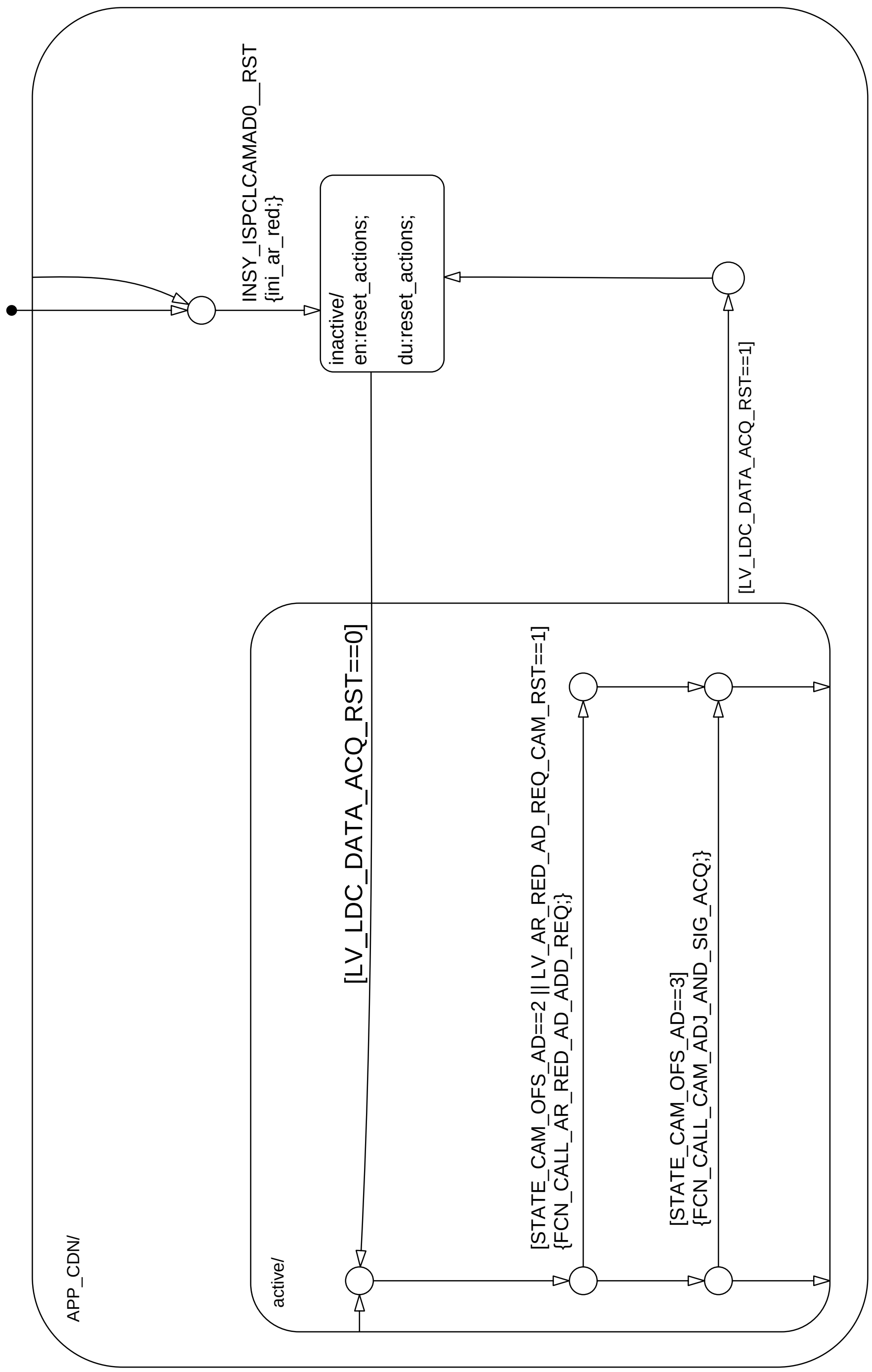


Figure 126 INSY_M908F/ APP_CDN/ Chart

Function Description

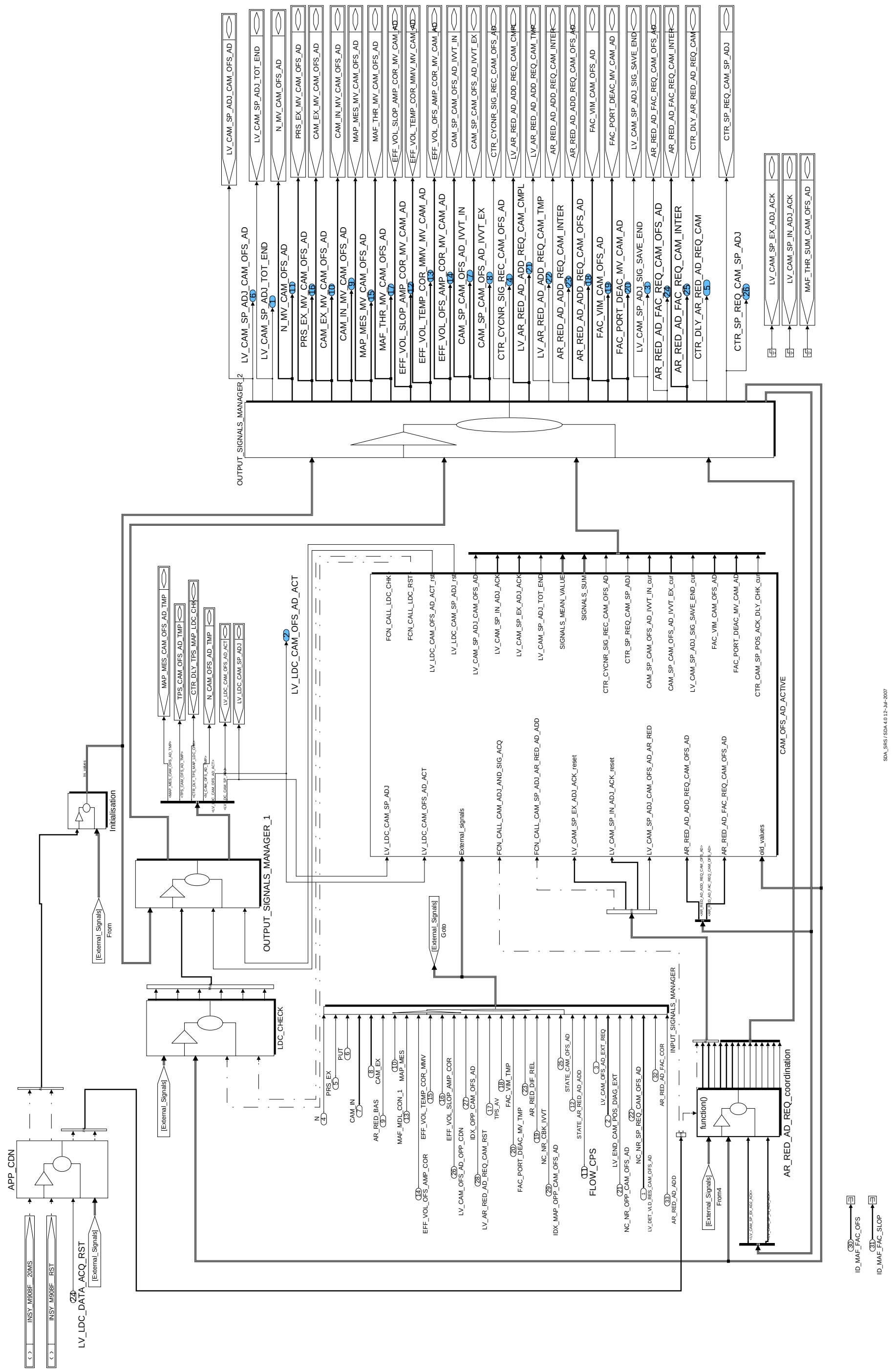


Figure 127 INSY_M908F

9.47.1.1 Initialisation

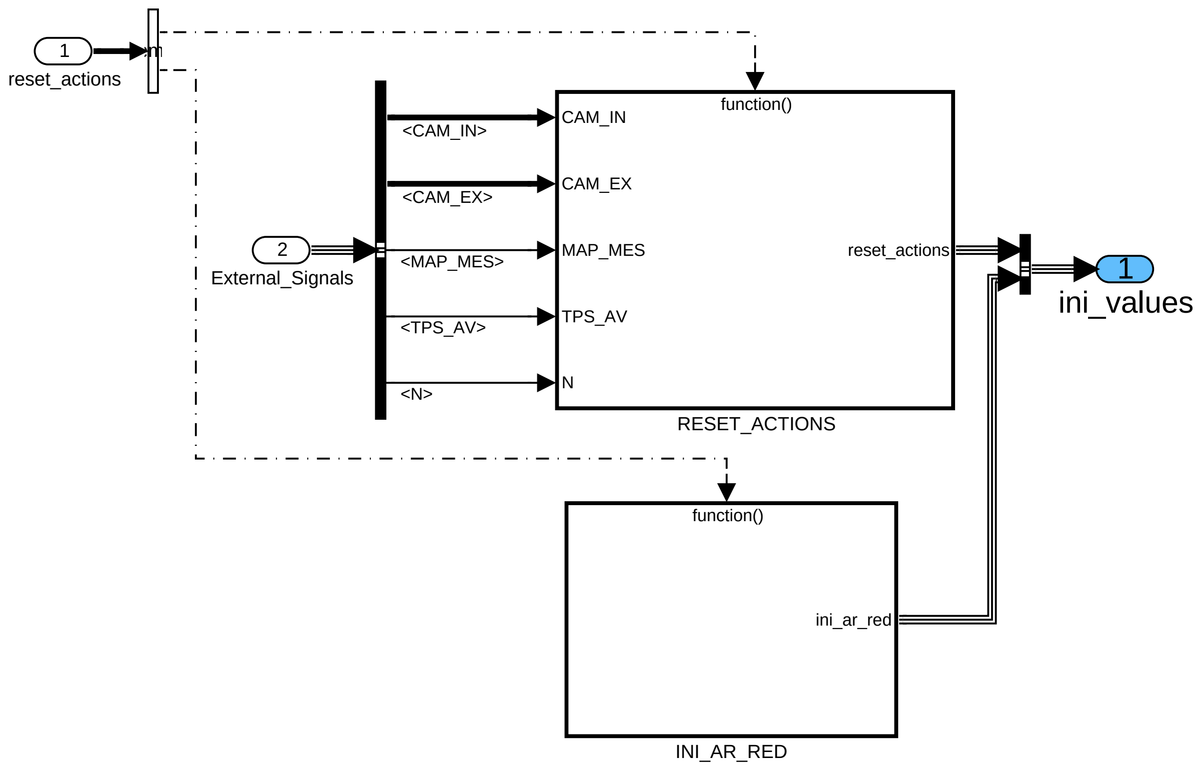


Figure 128 INSY_M908F/ Initialisation

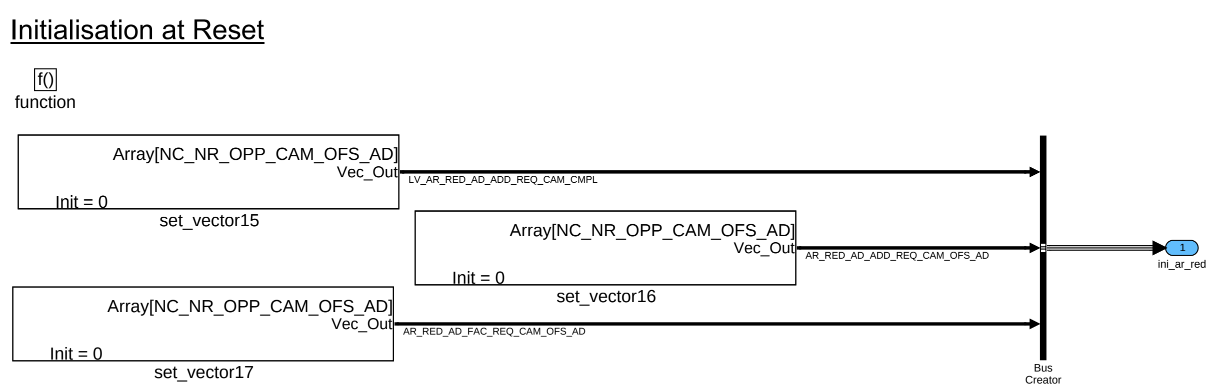


Figure 129 INSY_M908F/ Initialisation/ INI_AR_RED

Initialisation at Reset and LV_LDC_DATA_ACQ_RST = 1

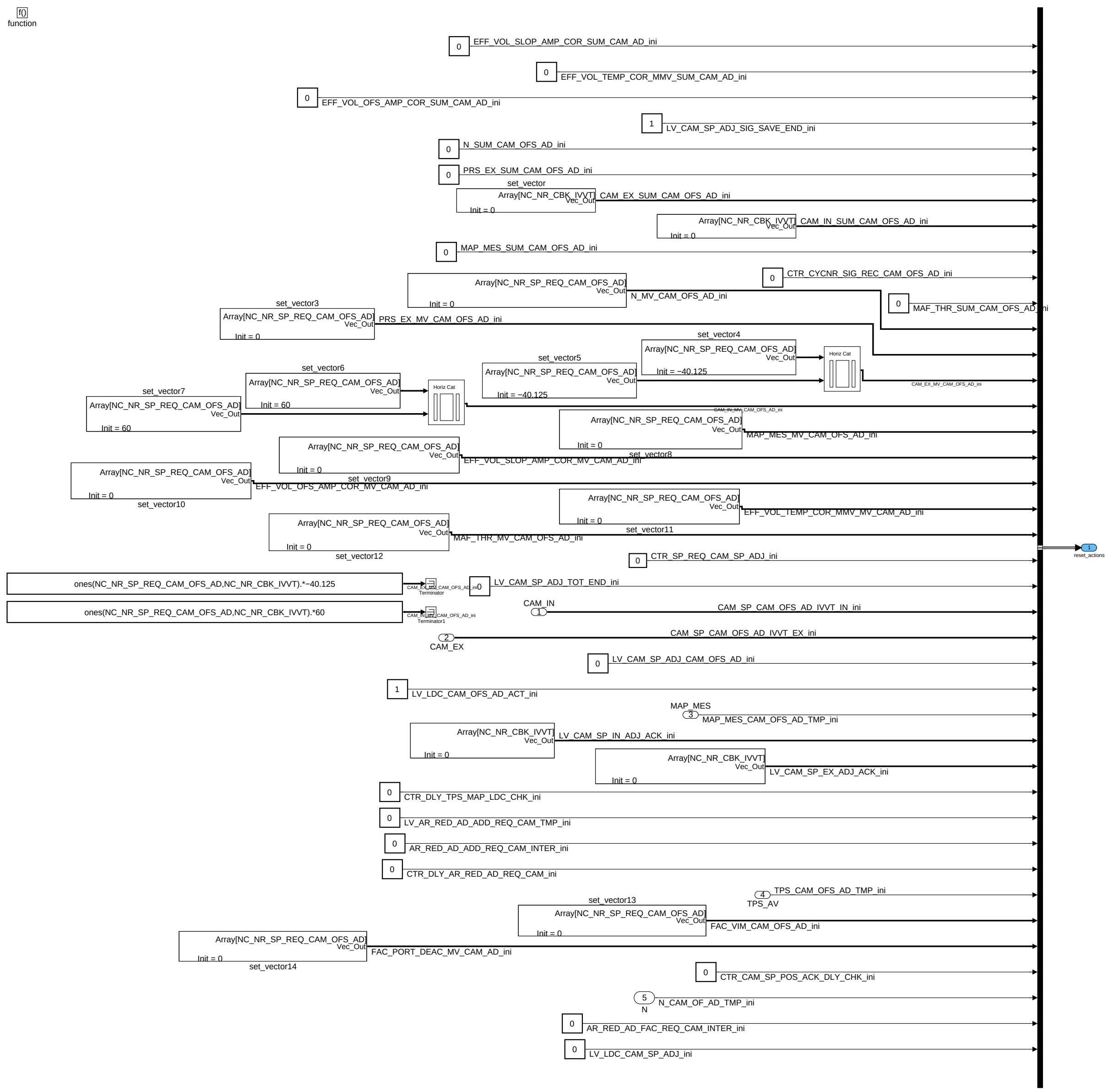


Figure 130 INSY_M908F/ Initialisation/ RESET_ACTIONS

9.47.1.2 AR_RED_AD_ADD calculation for Camshaft offset adaptation

In this subsystem, the request for the calculation of the reduced-area adaptation by the Camshaft-offset adaptation functionality is coordinated. The aim of the CAM_OFS_AD functionality is to learn an error in the inlet/exhaust camshaft position that may lead to an error in the calculation of the air-mass flow into the cylinder(MAF_ CYL). Since the adaptation is based on air-mass-flow balance equations, tolerances in any of the engine components(in addition to the camshaft) that influence air flow into the cylinder will also be interpreted by this functionality as a camshaft position error. In order to avoid this, a camshaft position combination having little influence on the air-mass flow into the cylinder(from ID_CAM_SP_IN/EX_AR_RED_AD_CAM_AD) will be utilised to learn any error/tolerances of the throttle-plate area(reduced-area).

In the intake manifold model, the error in the reduced-area is learnt by the reduced area-adaptation functionality. When the CAM_OFS_AD function has detected that engine is in one of the desired adaptation operating points(LV_CAM_OFS_AD_OPP_ CDN=1), the calculation of AR_RED_AD_ADD is triggered. For this, the existing AR_RED_AD_ADD value learnt by the IMM functionalities is saved temporarily and AR_RED_AD_ADD is reset to zero. Once this is done, the camshaft is adjusted to the desired inlet/exhaust position and the learning of the adaptation value is started. The value learned is written into AR_RED_AD_ADD_ REQ_CAM_OFS_AD.

If the reduced-area adaptation were to be interrupted due to some external influences(for e.g. LV_ERR_MAP/MAF), the adaptation result at that point of time will be saved in AR_RED_AD_ADD_REQ_CAM_OFS_AD. At the start of the next reduced-area adaptation calculation instance for the CAM_OFS_AD , the value saved in AR_RED_AD_ADD_REQ_CAM_OFS_AD will be used as the initial value for

AR_RED_AD_ADD

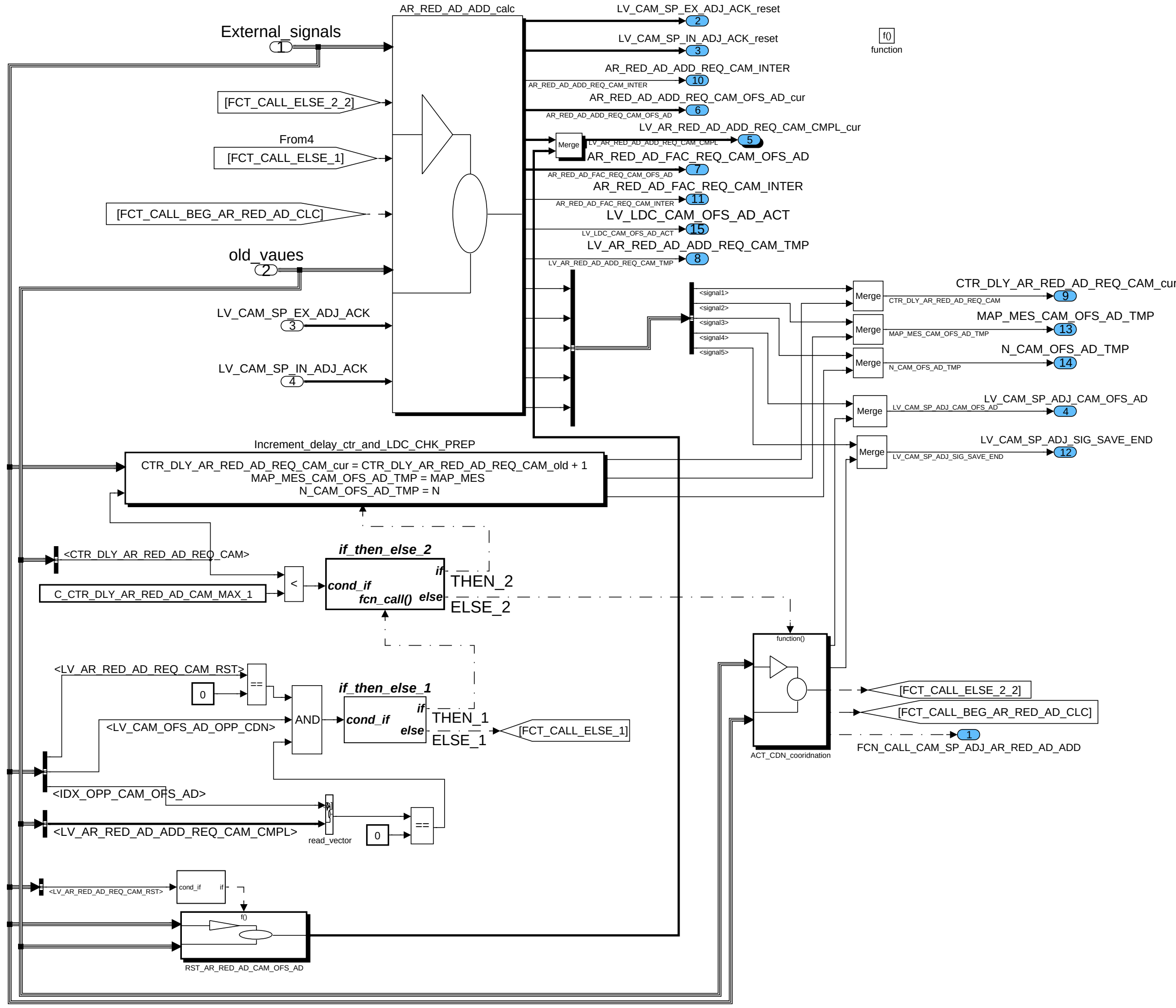


Figure 131 INSY_M908F/ AR_RED_AD_REQ_coordination

Coordination of activation conditions for AR_RED_AD_ADD calculation

When the conditions for the AR_RED_AD_ADD learning are fulfilled, it is checked if FLOW_CPS = 0 before starting the camshaft adjustment in order to avoid learning any errors of the canister-purge-flow model into the AR_RED_AD_ADD value learnt for camshaft-offset adaptation. If however the AR_RED_AD_ADD learning is already running and canister purge flow were to be come active, LV_LDC_CAM_OFS_AD_ACT is set to zero in the subsystem AR_RED_AD_ADD_calc and consequently the entire camshaft-offset adaptation run aborted.

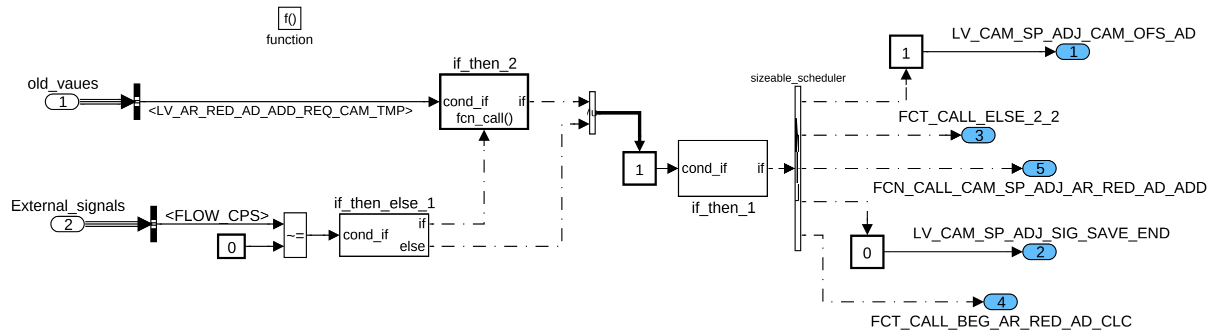


Figure 132 INSY_M908F/ AR_RED_AD_REQ_coordination/ ACT_CDN_cooridnation

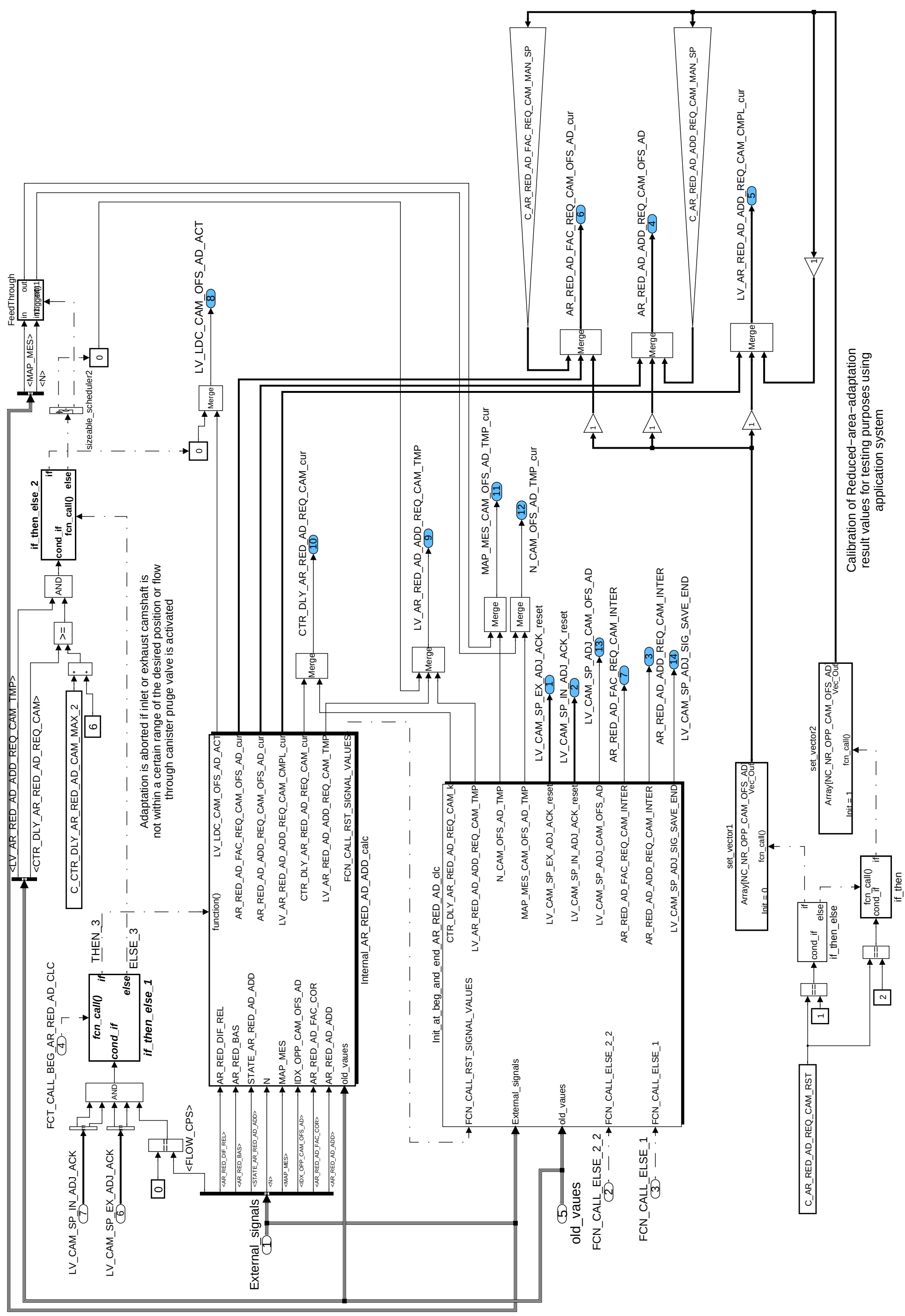


Figure 133 INSY_M908F/ AR_RED_AD_REQ_coordination/ AR_RED_AD_ADD_calc

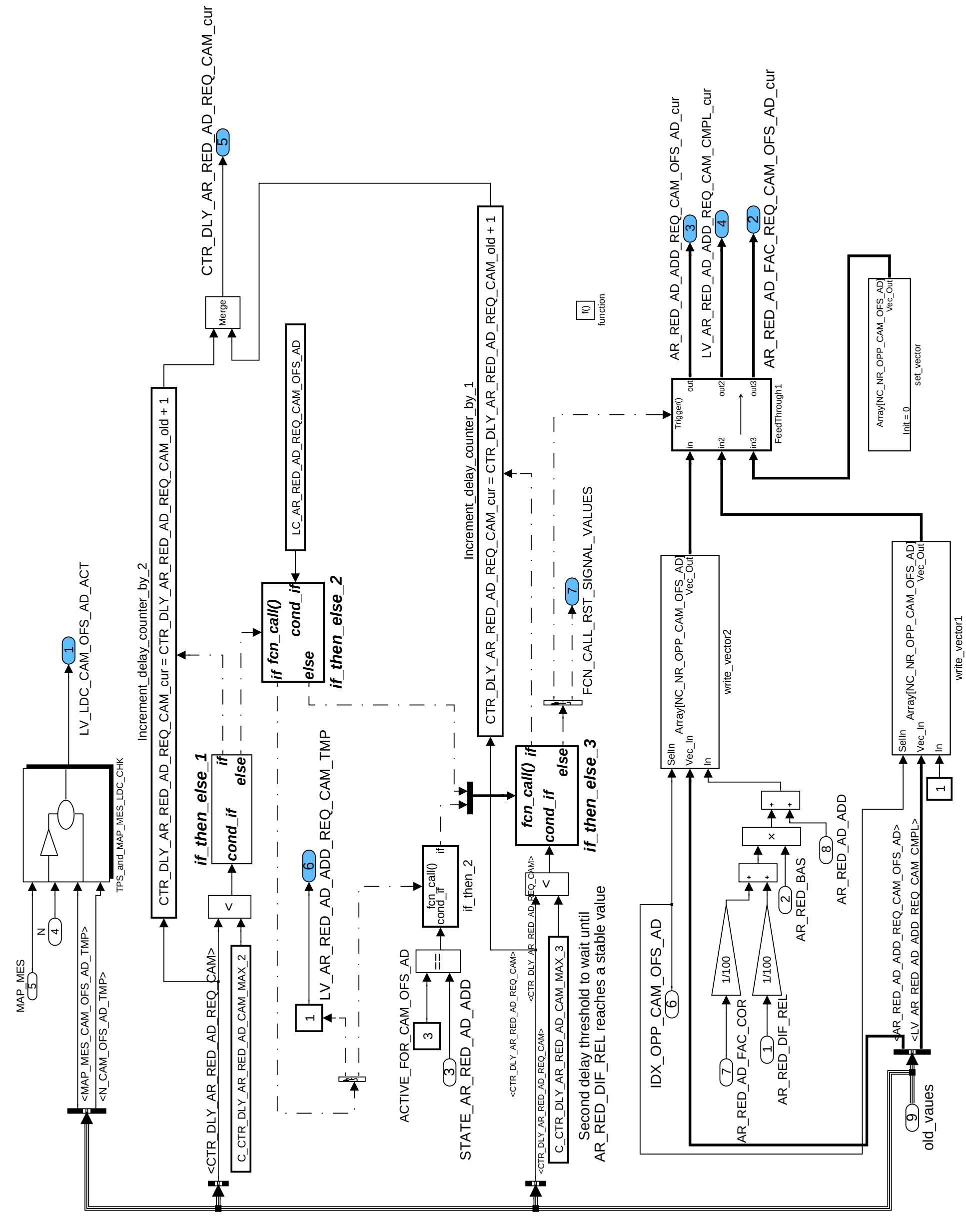


Figure 134 INSY_M908F/ AR_RED_AD_REQ_coordination/ AR_RED_AD_ADD_calc/ Internal_AR_RED_AD_ADD_calc

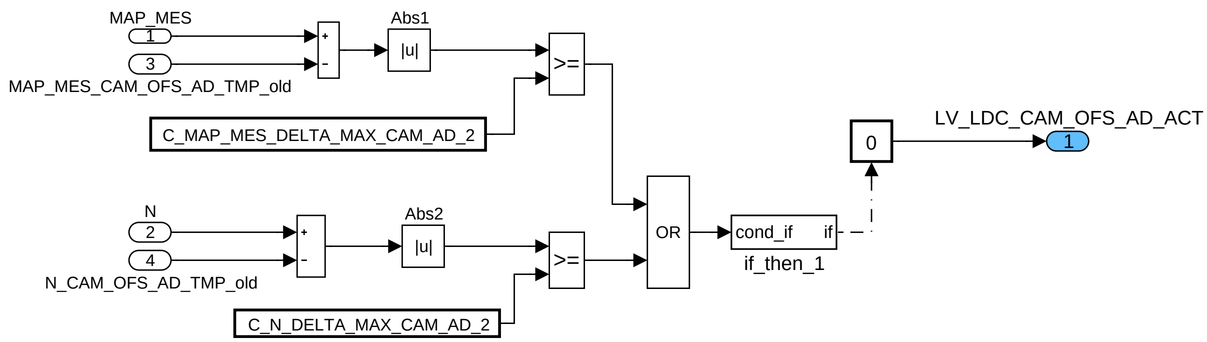


Figure 135 INSY_M908F/ AR_RED_AD_REQ_coordination/ AR_RED_AD_ADD_calc/ Internal_AR_RED_AD_ADD_calc/ TPS_and_MAP_MES_LDC_CHK

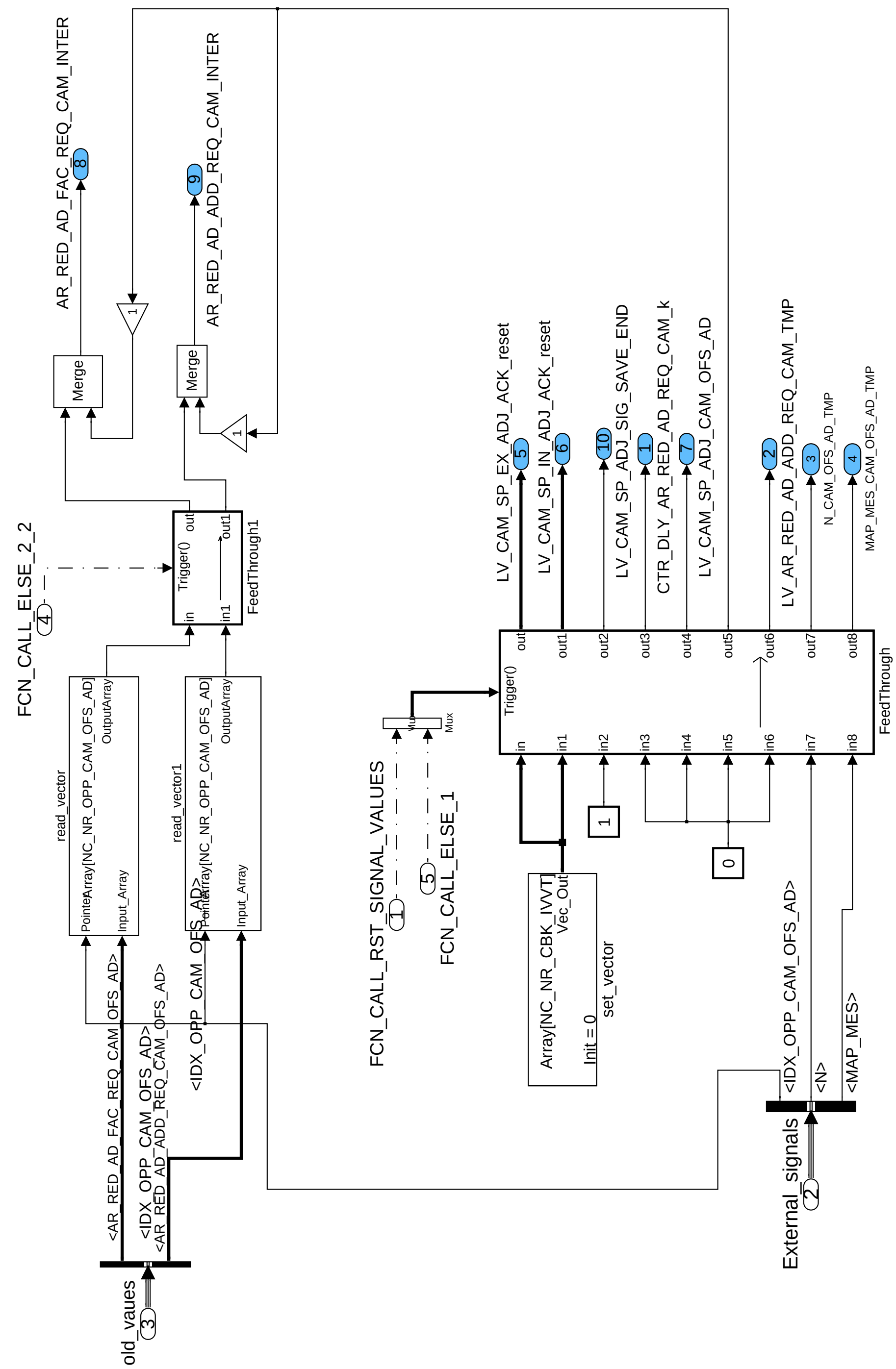


Figure 136 INSY_M908F/ AR_RED_AD_REQ_coordination/ AR_RED_AD_ADD_calc/ Init_at_beg_and_end_AR_RED_AD_clc

Calculation of LV_AR_RED_AD_ADD_REQ_CAM_CMPL

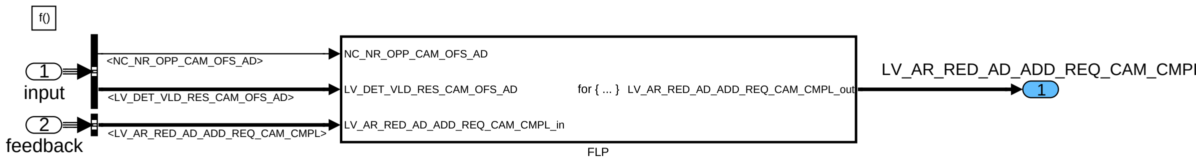


Figure 137 INSY_M908F/ AR_RED_AD_REQ_coordination/ RST_AR_RED_AD_CAM_OFS_AD

Calculation of LV_AR_RED_AD_ADD_REQ_CAM_CMPL in For loop

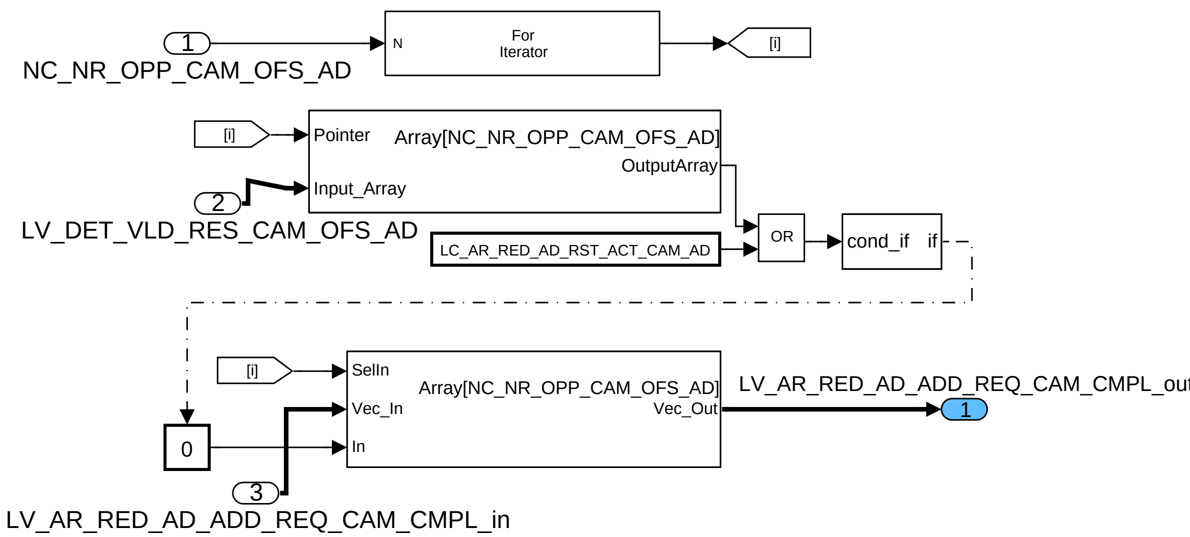


Figure 138 INSY_M908F/ AR_RED_AD_REQ_coordination/ RST_AR_RED_AD_CAM_OFS_AD/ FLP

9.47.1.3 Detection of limited dynamics condition

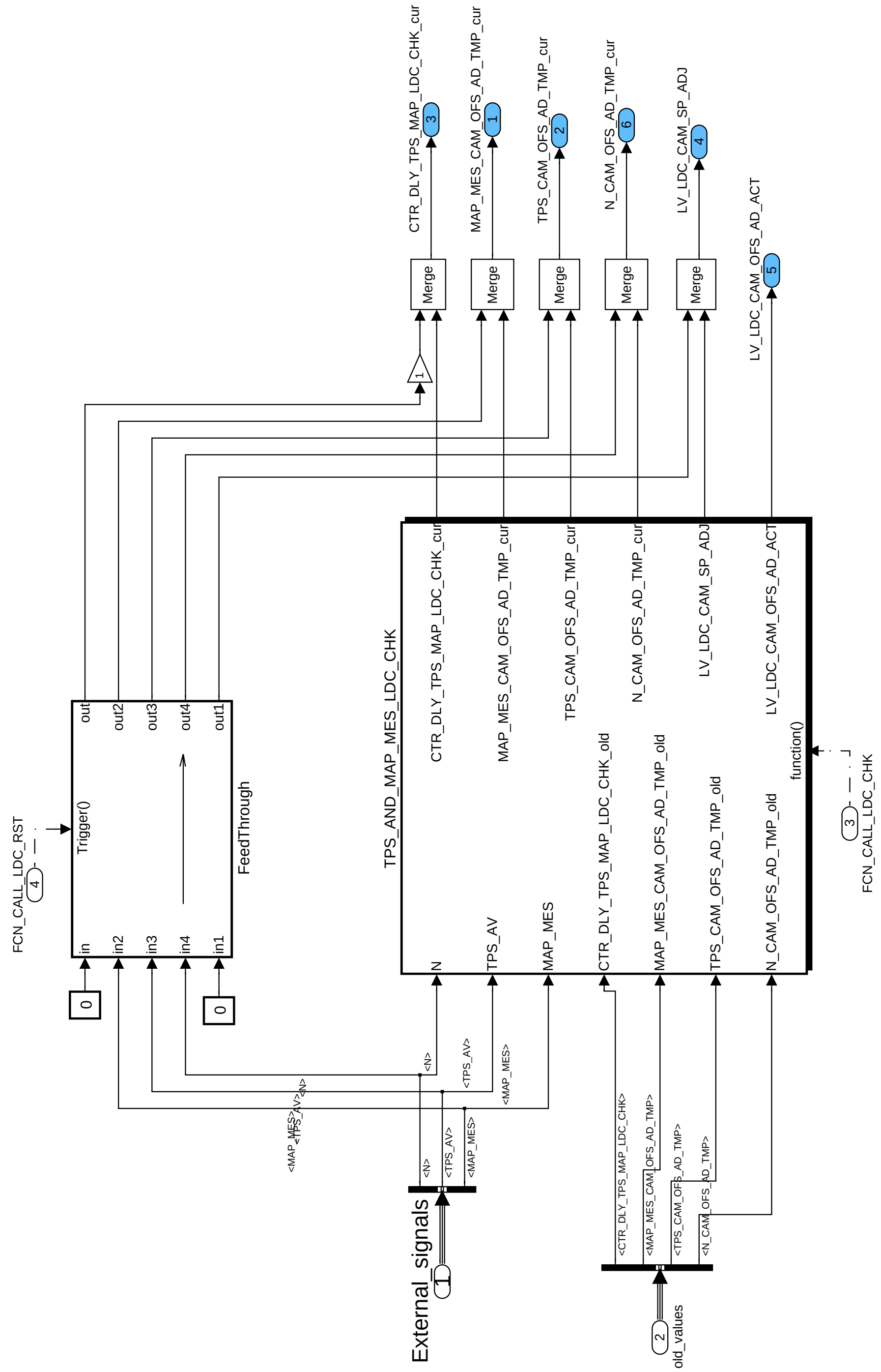


Figure 139 INSY_M908F/ LDC_CHECK

Limited Dynamics check for MAP_MES and other relevant variables

In order to determine if TPS_AV is stationary or not, the difference between TPS_AV and its low-pass filtered value is compare to a threshold. If this difference is greater than the threshold, C_TPS_DELTA_MAX_CAM_AD, then LV_LDC_CAM_ OFS_AD_ACT is set to zero. If however the TPS_AV signal is within the LDC threshold, then the limited dynamics condition check is declared as fulfilled. Consequently the adjustment of the camshaft for adaptation purposes can be started.

The LDC check for MAP_MES is started when the camshaft has reached the desired inlet/exhaust position needed for adaptation. First a delay time is started to take into consideration a change in intake manifold pressure that results from shift in camshaft position. At the end of the delay, it is assumed that MAP_MES has reached a relatively stable value and therefore the pressure value is saved in a temporary variable, MAP_MES_CAM_OFS_AD_TMP. During the signal acquisition phase, the measured manifold pressure is constantly compared with MAP_MES_CAM_OFS_AD_TMP. If there is a significant increase or decrease in the pressure during the signal acquisition phase, the signal acquisition is aborted and the adaptation stopped to avoid learning a wrong position error due to inaccuracies resulting from the averaged MAP_MES value which is used for the MAF_CYL_CAM_OFS_AD calculation.

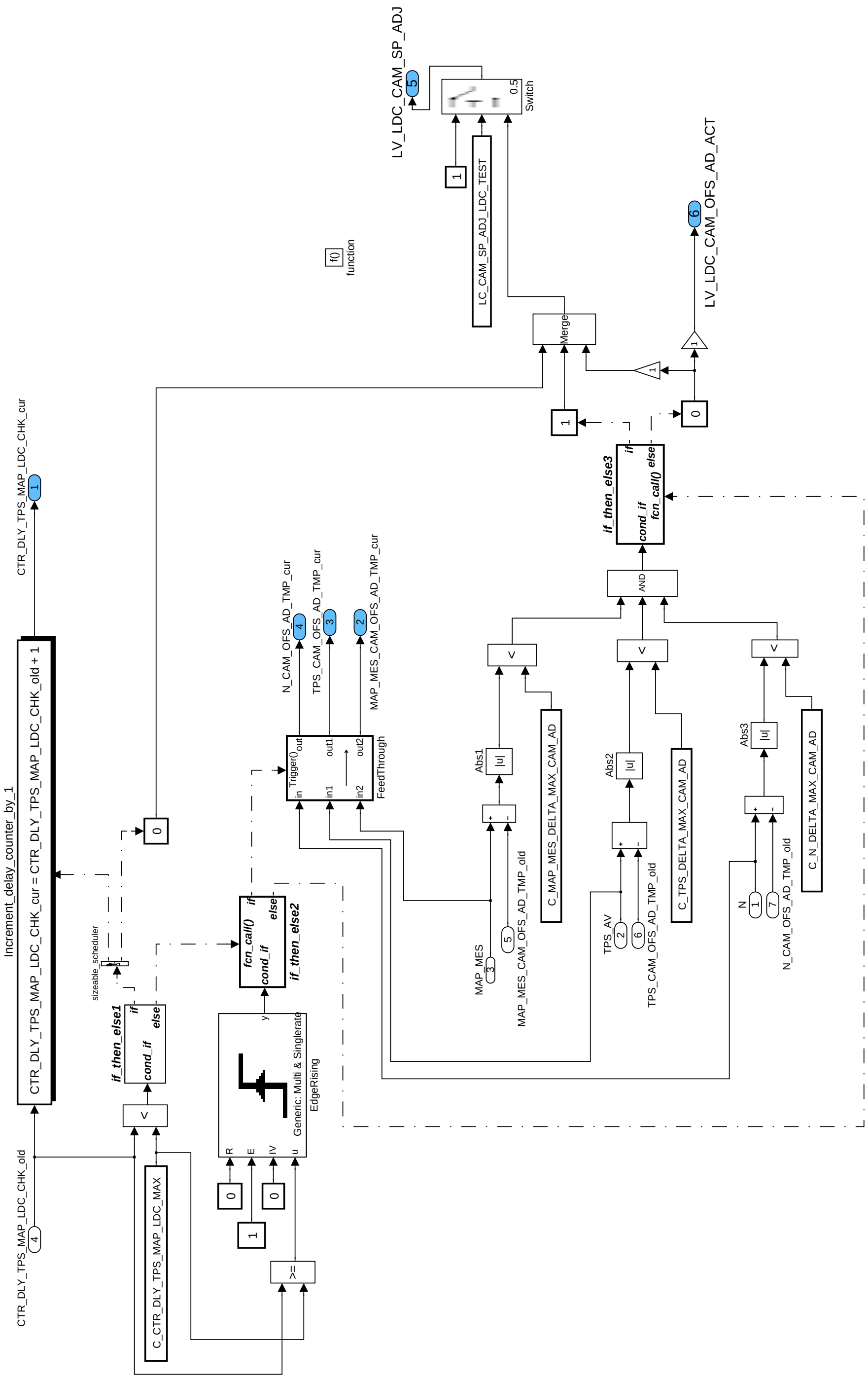

LV_LDC_CAM_SP_ADJ
5
Switch
0.5
1
LC_CAM_SP_ADJ_LDC_TEST
f()
function
LV_LDC_CAM_OFS_AD_ACT
6
Merge
1
1
1
0
CTR_DLY_TPS_MAP_LDC_CHK_cur
1
if_then_else3
cond_if
fcn_call()
if
else
AND
N_CAM_OFS_AD_TMP_cur
4
TPS_CAM_OFS_AD_TMP_cur
3
MAP_MES_CAM_OFS_AD_TMP_cur
2
Trigger()
in
out
in1
out1
in2
out2
FeedThrough
Abs1
Abs2
Abs3
|u|
C_MAP_MES_DELTA_MAX_CAM_AD
C_TPS_DELTA_MAX_CAM_AD
C_N_DELTA_MAX_CAM_AD
Increment_delay_counter_by_1
CTR_DLY_TPS_MAP_LDC_CHK_cur = CTR_DLY_TPS_MAP_LDC_CHK_old + 1
0
sizeable_scheduler
if_then_else1
if_then_else2
MAP_MES
3
MAP_MES_CAM_OFS_AD_TMP_old
5
TPS_AV
2
TPS_CAM_OFS_AD_TMP_old
6
N
1
N_CAM_OFS_AD_TMP_old
7
Generic: Multi & Singlerate
EdgeRising
R
E
IV
u
y
CTR_DLY_TPS_MAP_LDC_CHK_old
4
C_CTR_DLY_TPS_MAP_LDC_MAX
>=
0
0
1

Figure 140 INSY_M908F/ LDC_CHECK/ TPS_AND_MAP_MES_LDC_CHK

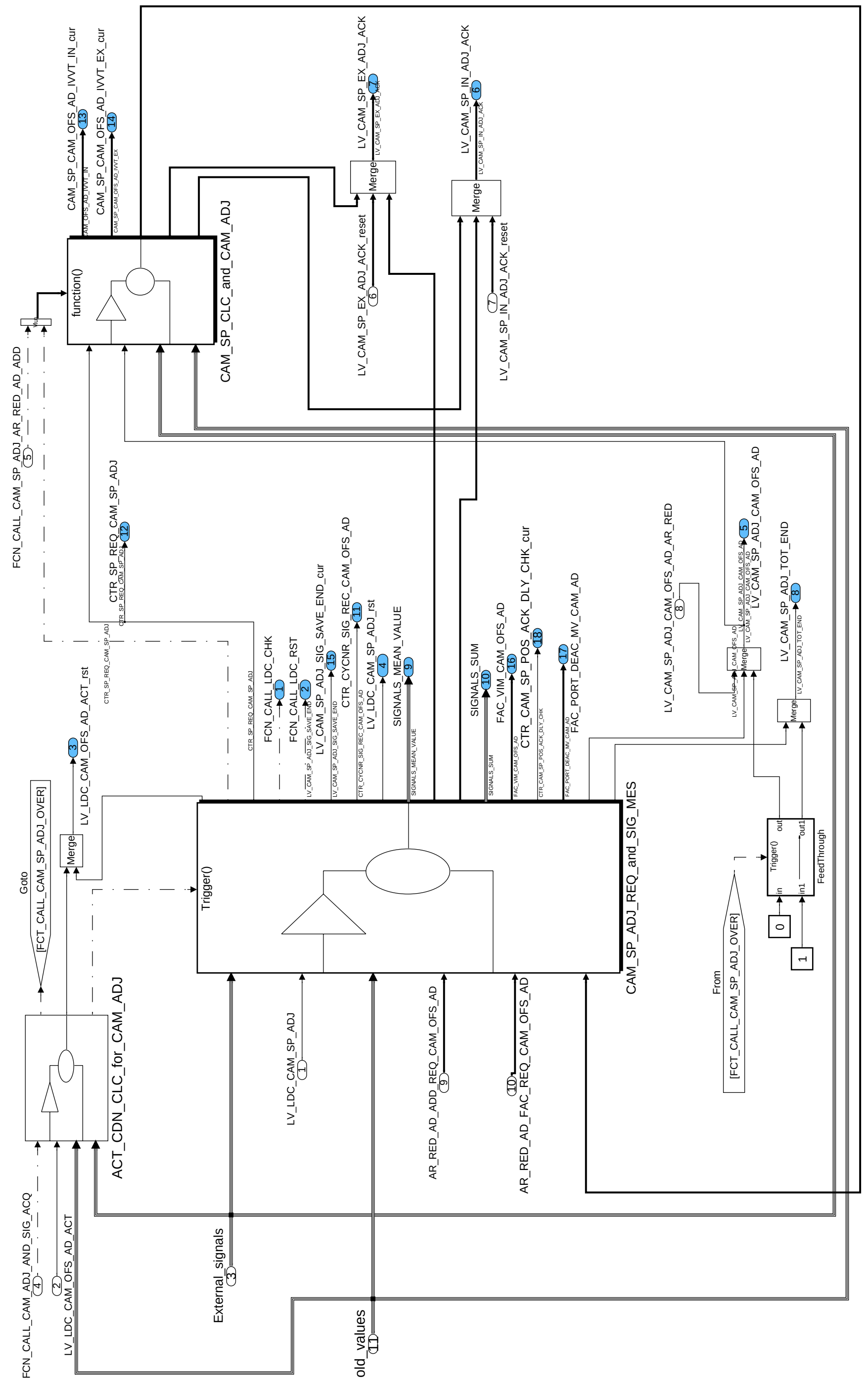


Figure 141 INSY_M908F/ CAM_OFS_AD_ACTIVE

Coordination of activation conditions for camshaft adjustment and signal acquisition

When the conditions for the camshaft adjustment are fulfilled (LV_CAM_OFS_AD_OPP _CDN =1), it is checked if FLOW_CPS = 0 before the inlet and exhaust adjustment is allowed in order to avoid learning any errors in the canister-purge-flow model into the camshaft-position-error. If the canister purge flow were to be come active during the signal acquisition phase, LV_LDC_CAM_OFS_AD_ACT is set to zero in the subsystem

CAM_SP_ADJ_REQ_and_SIG_MES and consequently the entire camshaft-offset adaptation run is aborted.

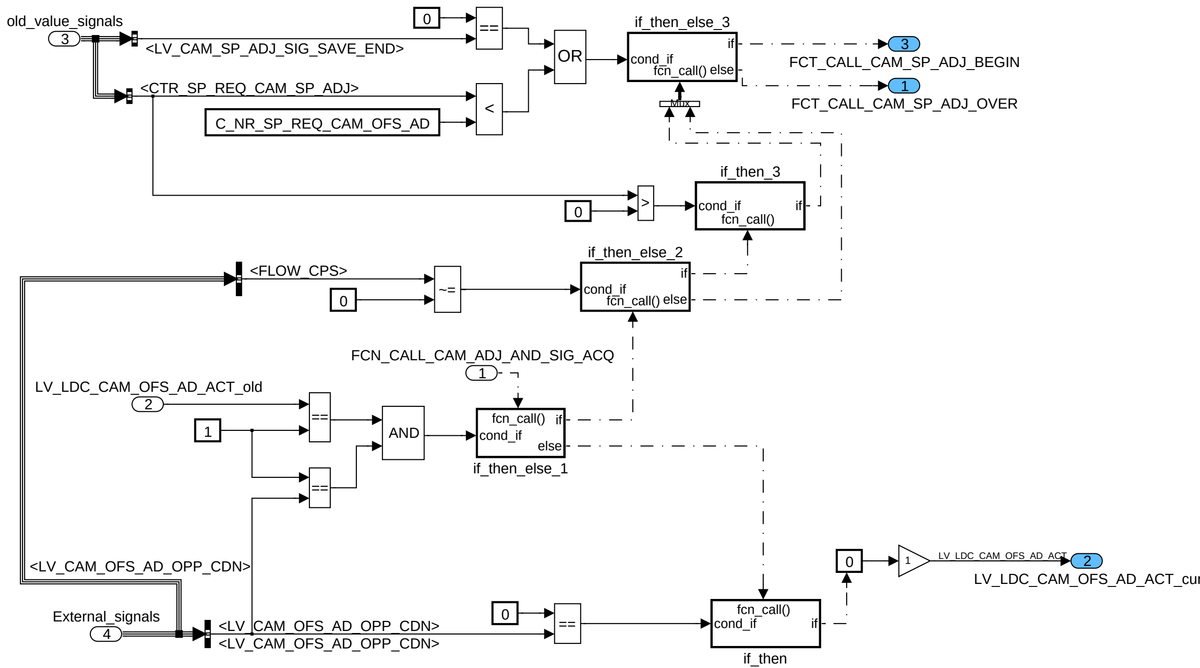


Figure 142 INSY_M908F/ CAM_OFS_AD_ACTIVE/ ACT_CDN_CLC_for_CAM_ADJ

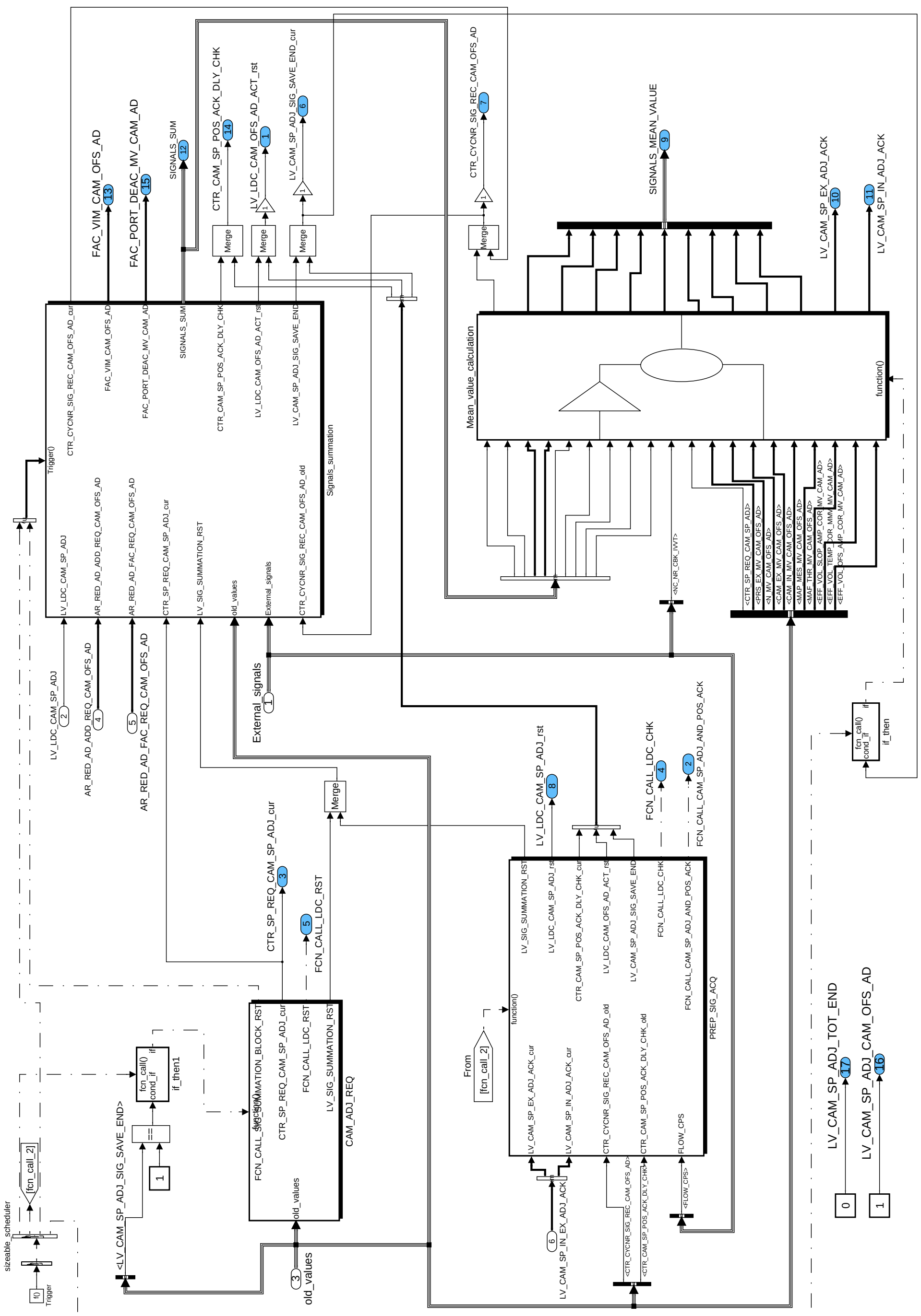


Figure 143 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES

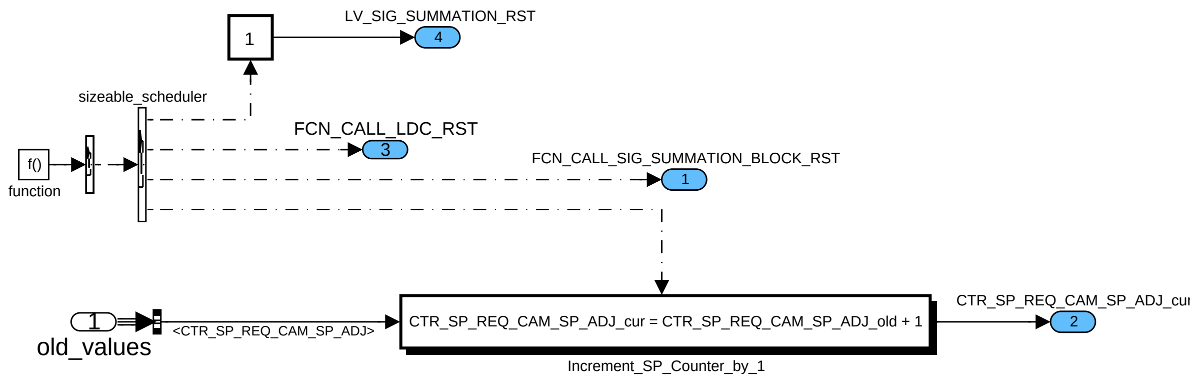


Figure 144 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ CAM_ADJ_REQ

INSY_M908F/CAM_OFS_AD_ACTIVE/CAM_SP_ADJ_REQ_AND_SIG_MES/PREP_SIG_ACQ

The logical bits LV_CAM_SP_IN/EX_ACK_ADJ are checked to see if the inlet and exhaust camshafts on both the cylinder banks have reached the desired position(CAM_SP_CAM_OFS_AD_IVVT_IN/EX). If the desired camshaft position has been set, the limited dynamics check for MAP_MES is started. During the signal acquisition phase, if the camshaft moves away from the desired setpoint(for e.g. due to oscillation), the bit LV_LDC_CAM_SP_ADJ is set to zero and the adaptation-run is aborted by the camshaft-offset adaptation manager functionality.

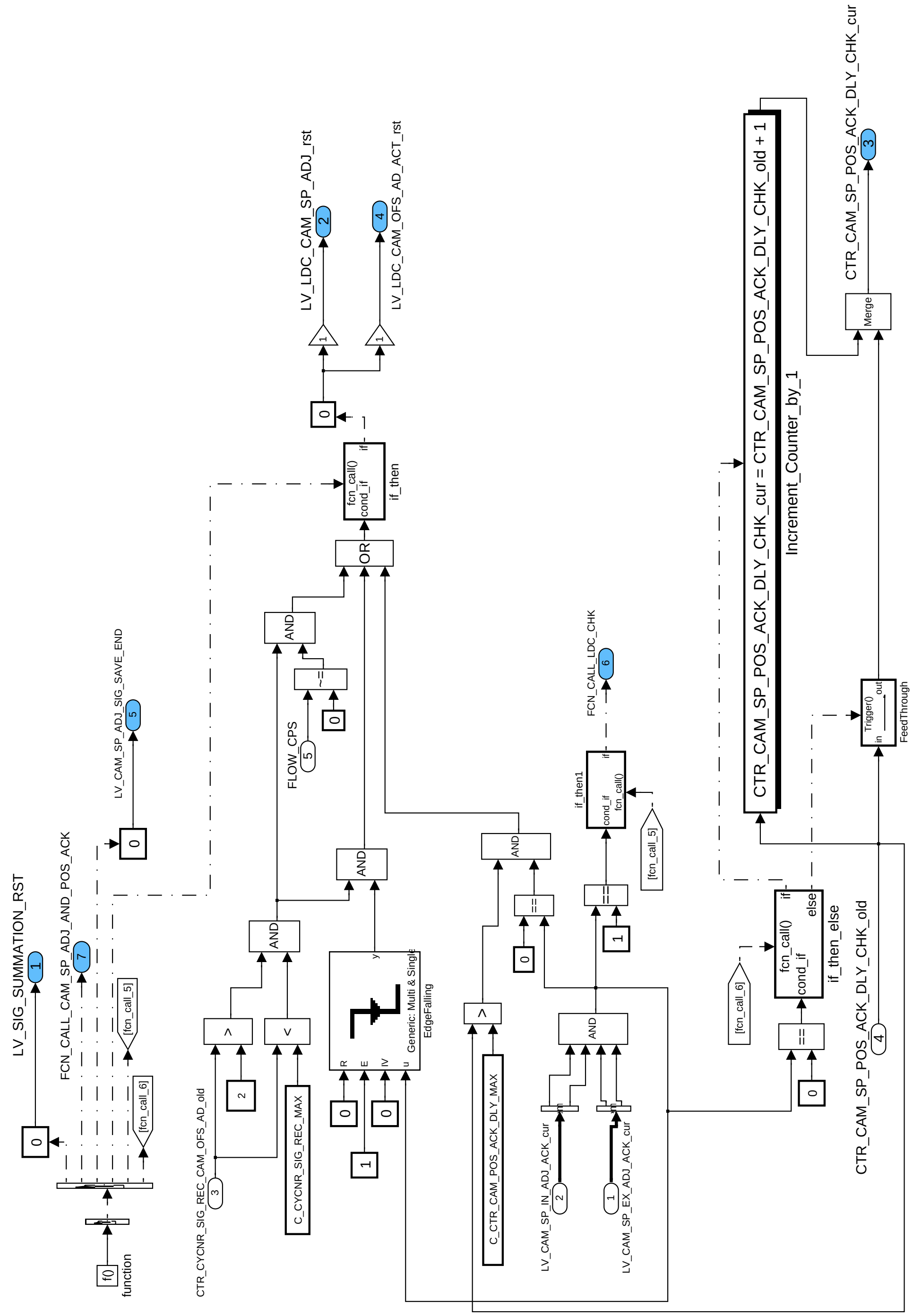


Figure 145 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ PREP_SIG_ACQ

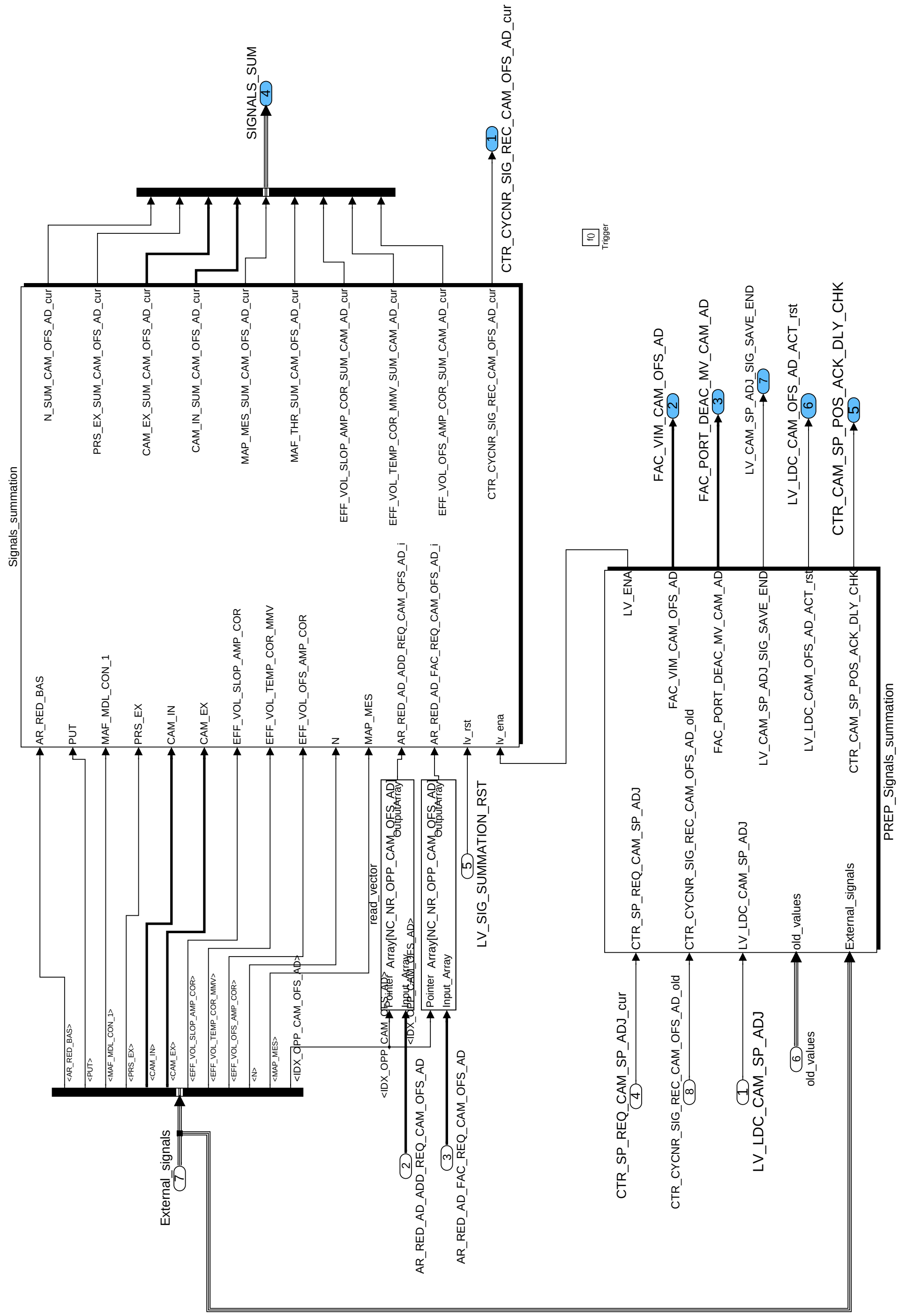


Figure 146 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation

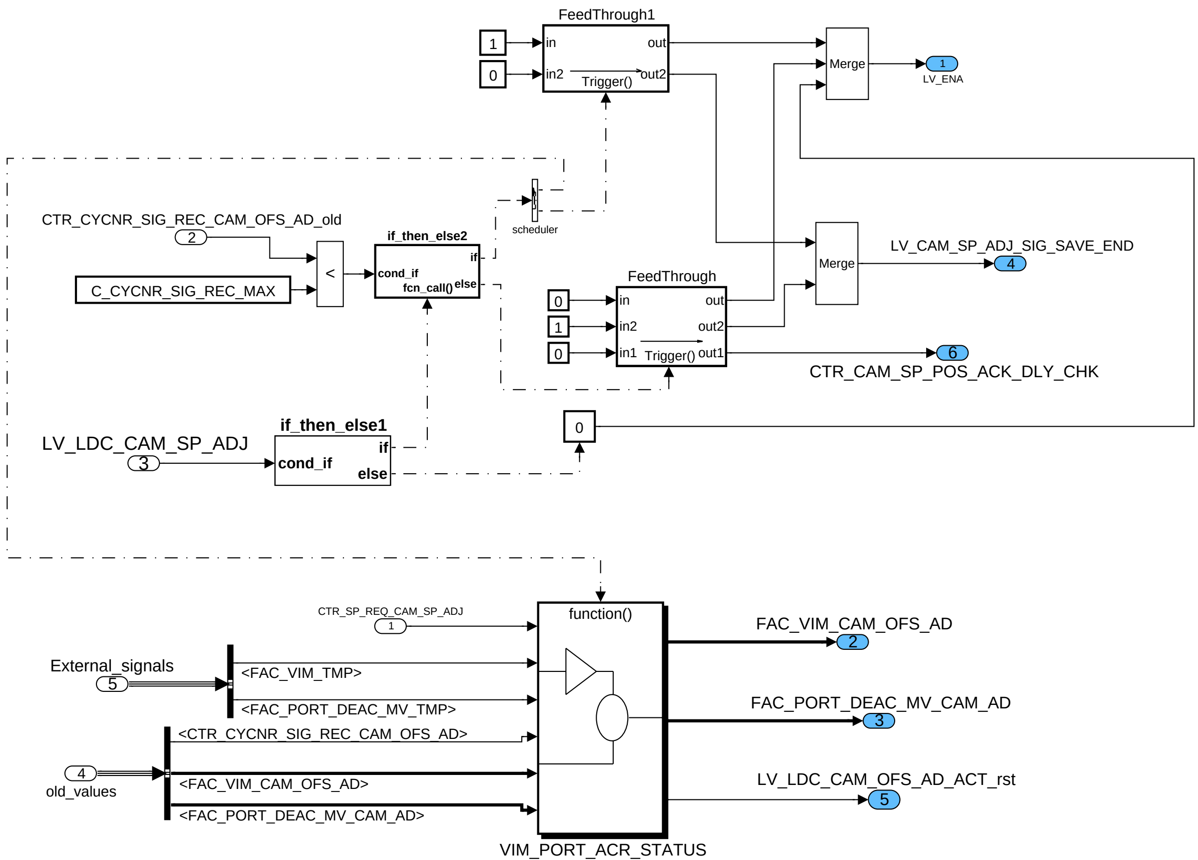


Figure 147 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation/ PREP_Signals_summation

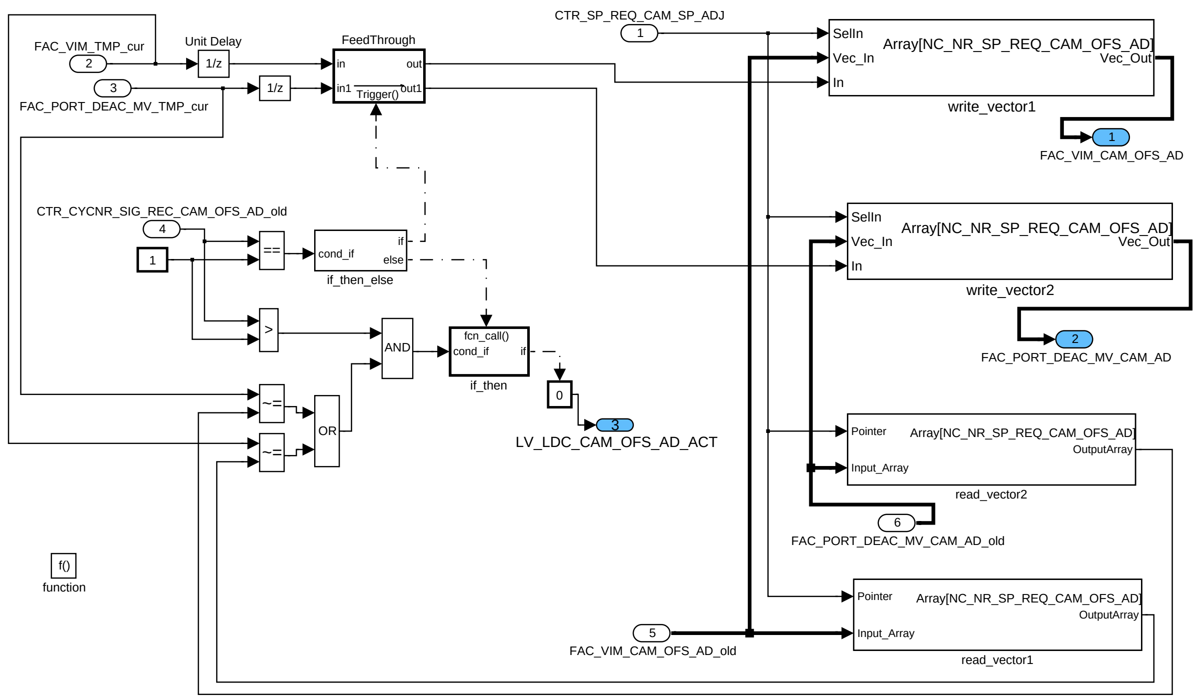


Figure 148 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation/ PREP_Signals_summation/ VIM_PORT_ACR_STATUS

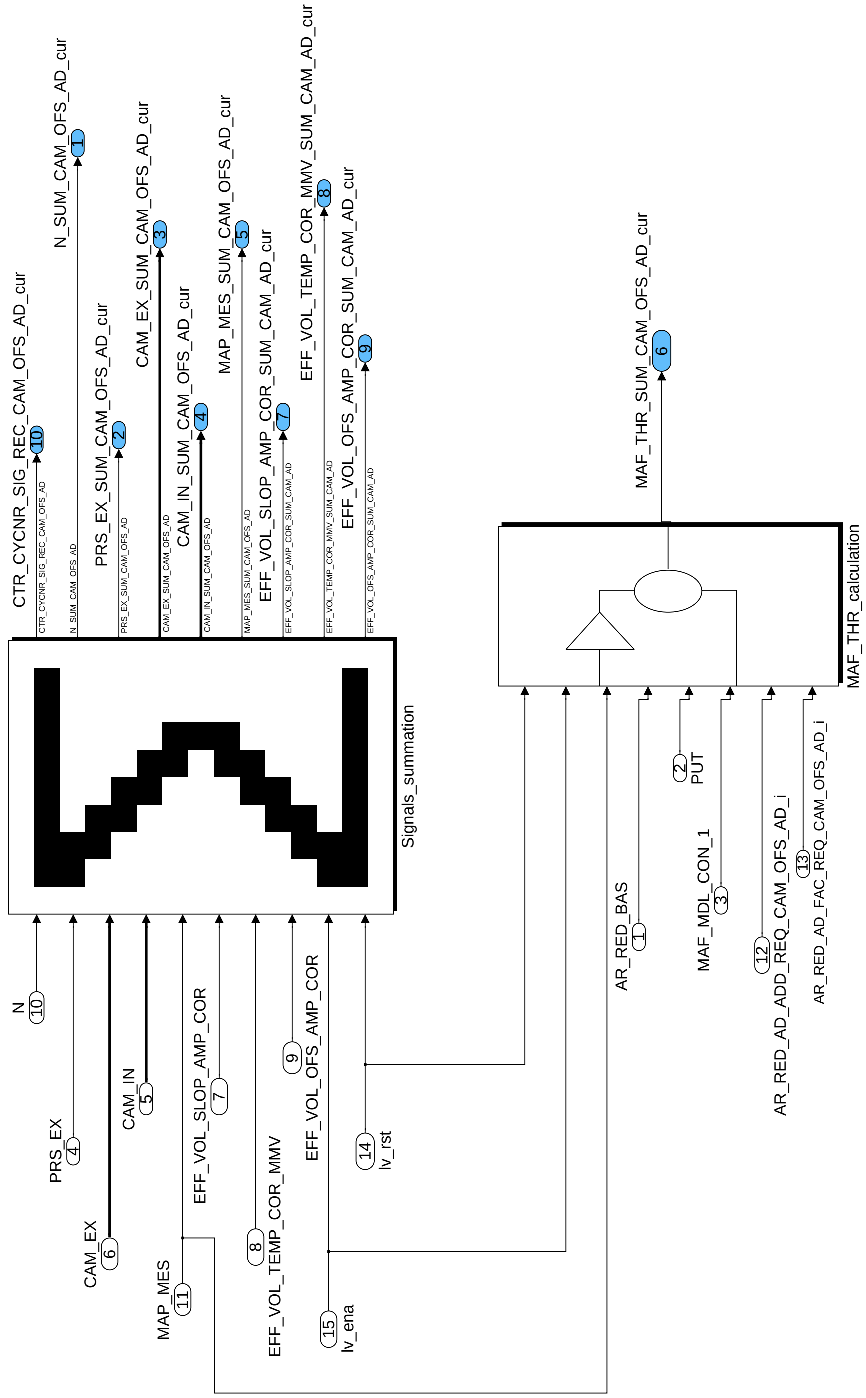


Figure 149 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation/ Signals_summation

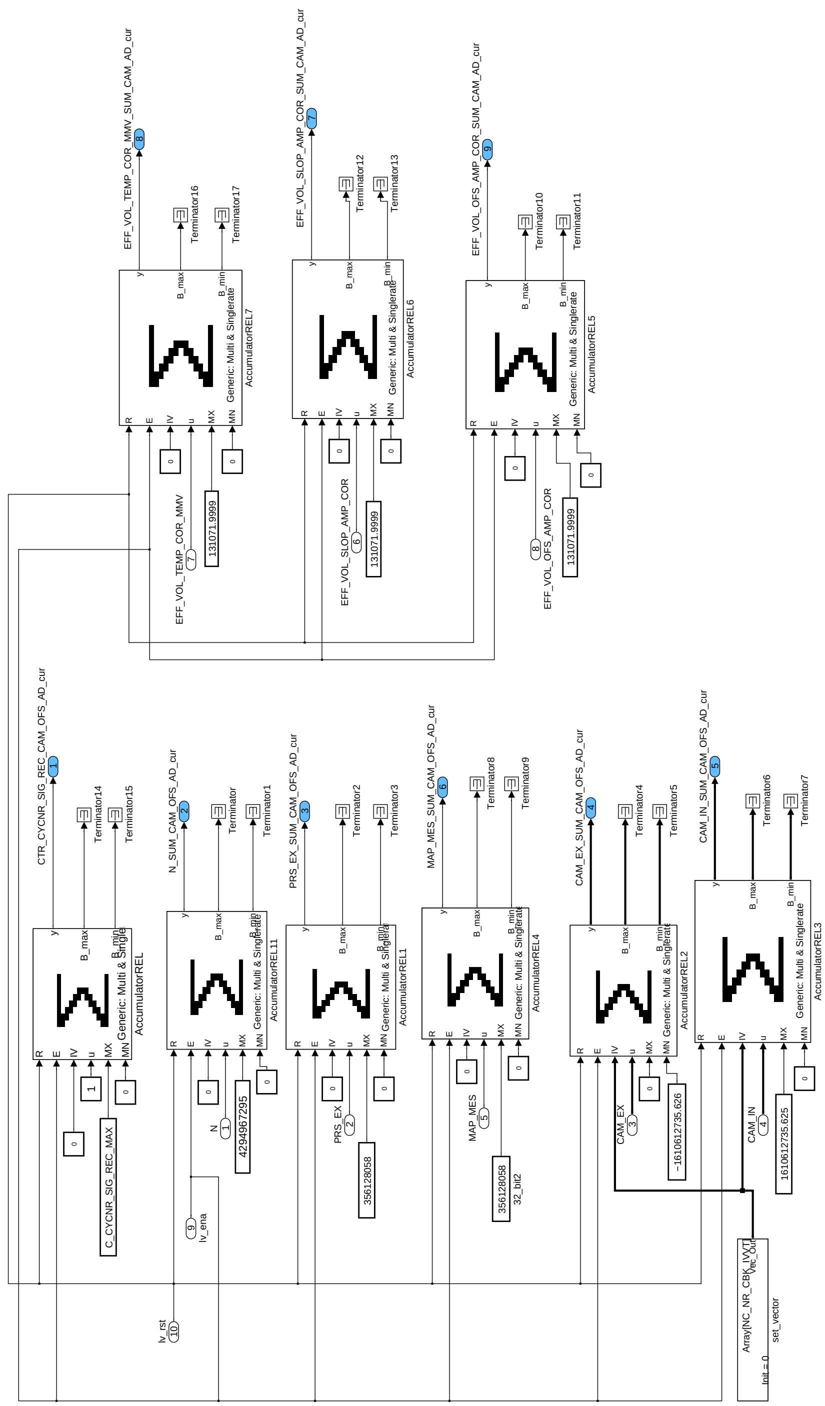


Figure 150 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation/ Signals_summation/ Signals_summation

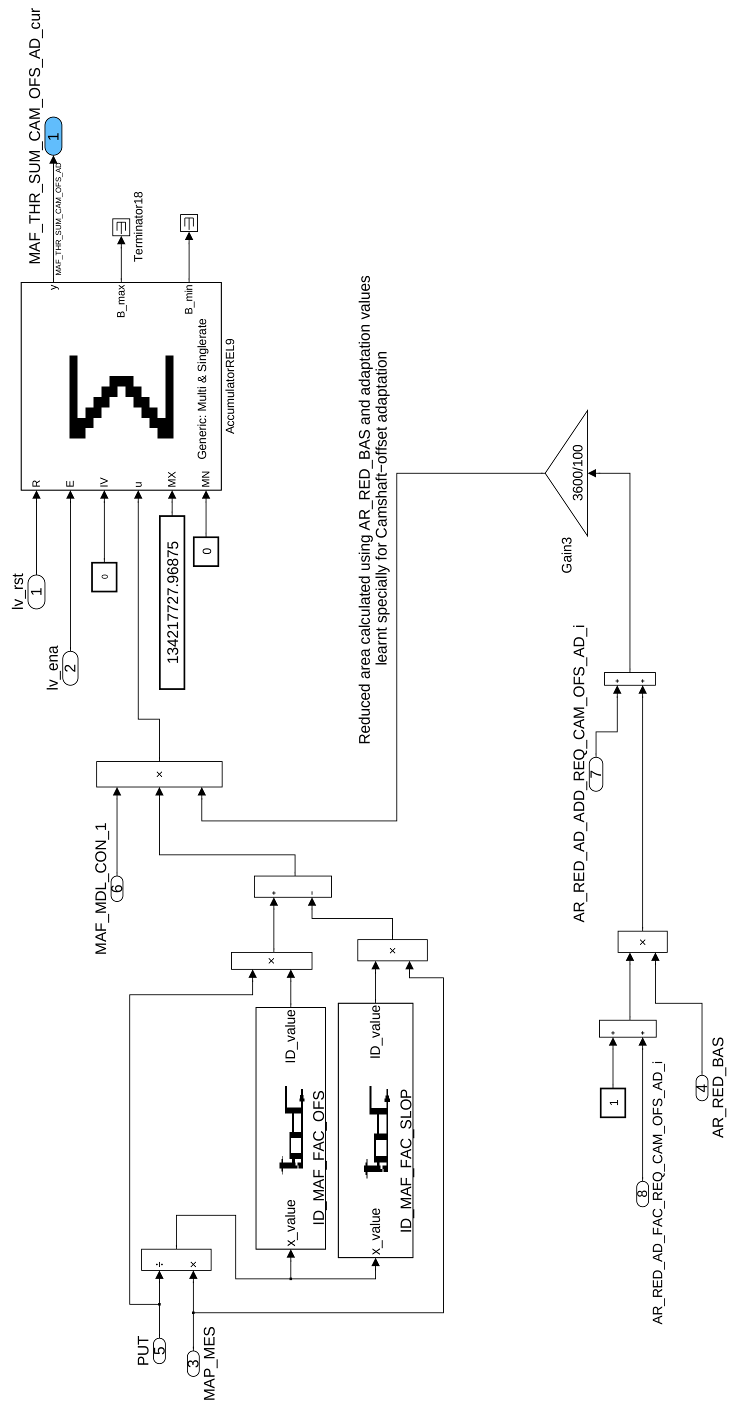


Figure 151 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Signals_summation/ Signals_summation/ MAF_THR_calculation

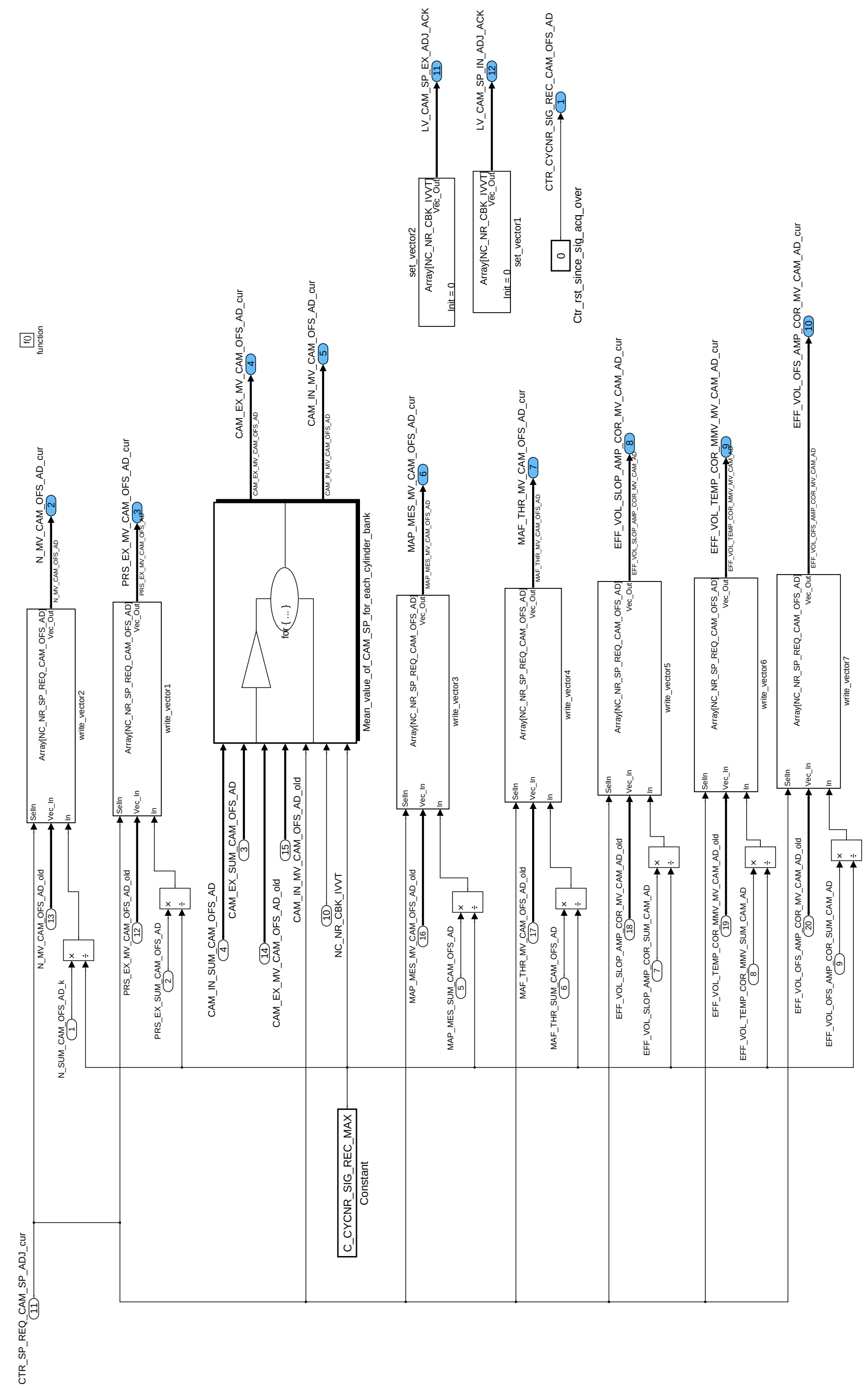


Figure 152 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Mean_value_calculation

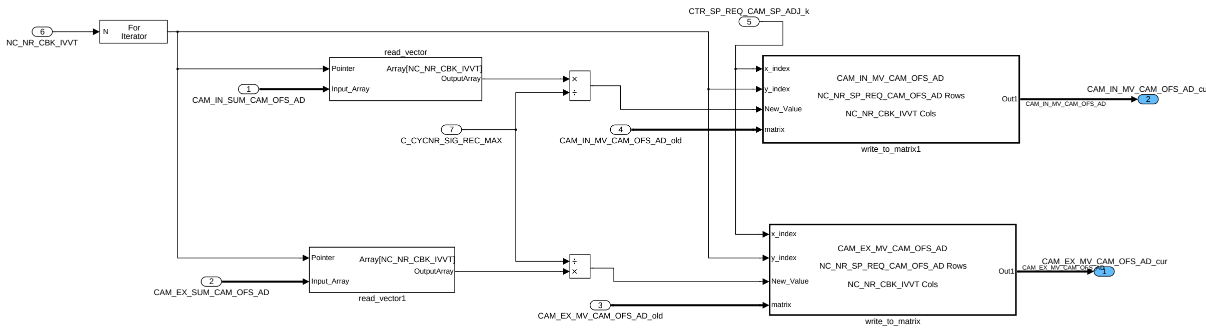


Figure 153 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_ADJ_REQ_and_SIG_MES/ Mean_value_calculation/ Mean_value_of_CAM_SP_for_each_cylinder_bank

Calculation of desired camshaft setpoint and coordination of camshaft adjustment

Based on which N/MAP operating point the engine currently is running on, the inlet and exhaust camshaft positions are chosen from the maps ID_CAM_SP_IVVT_IN/EX_CAM_OFS_AD and then actively adjusted by the VVTI functionality. Once the desired position of the inlet and exhaust camshaft on each cylinder bank has actually been, a logical bit is set(LV_CAM_SP_IN/EX_ADJ_ACK_i) to indicate this status so that the signal acquisition process can be started.

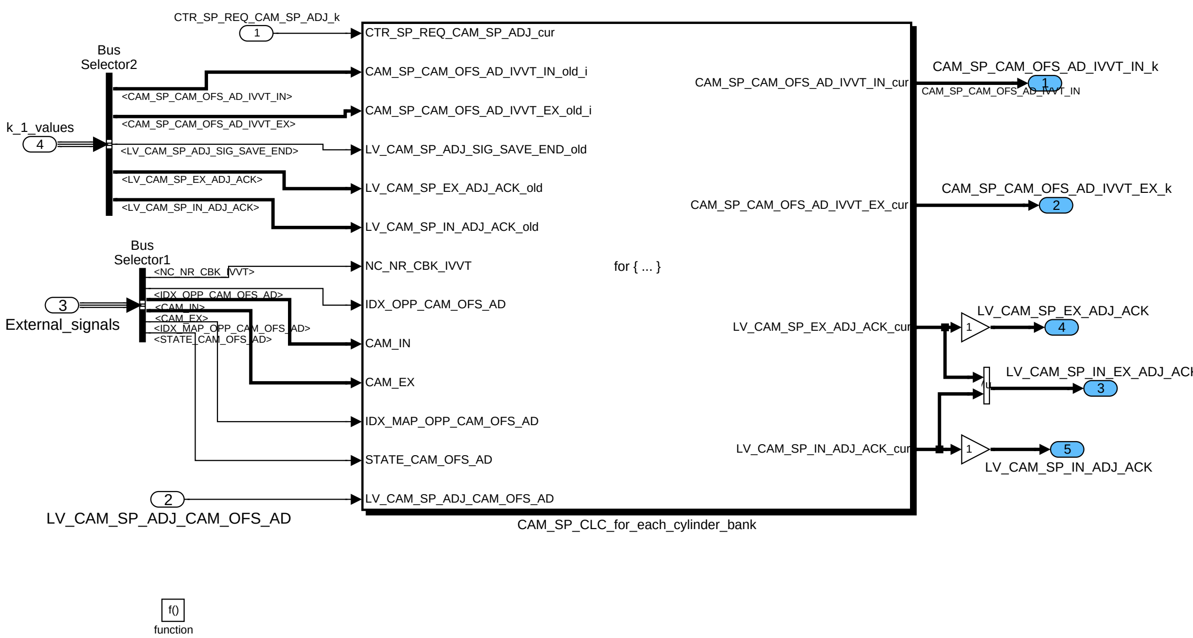


Figure 154 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ

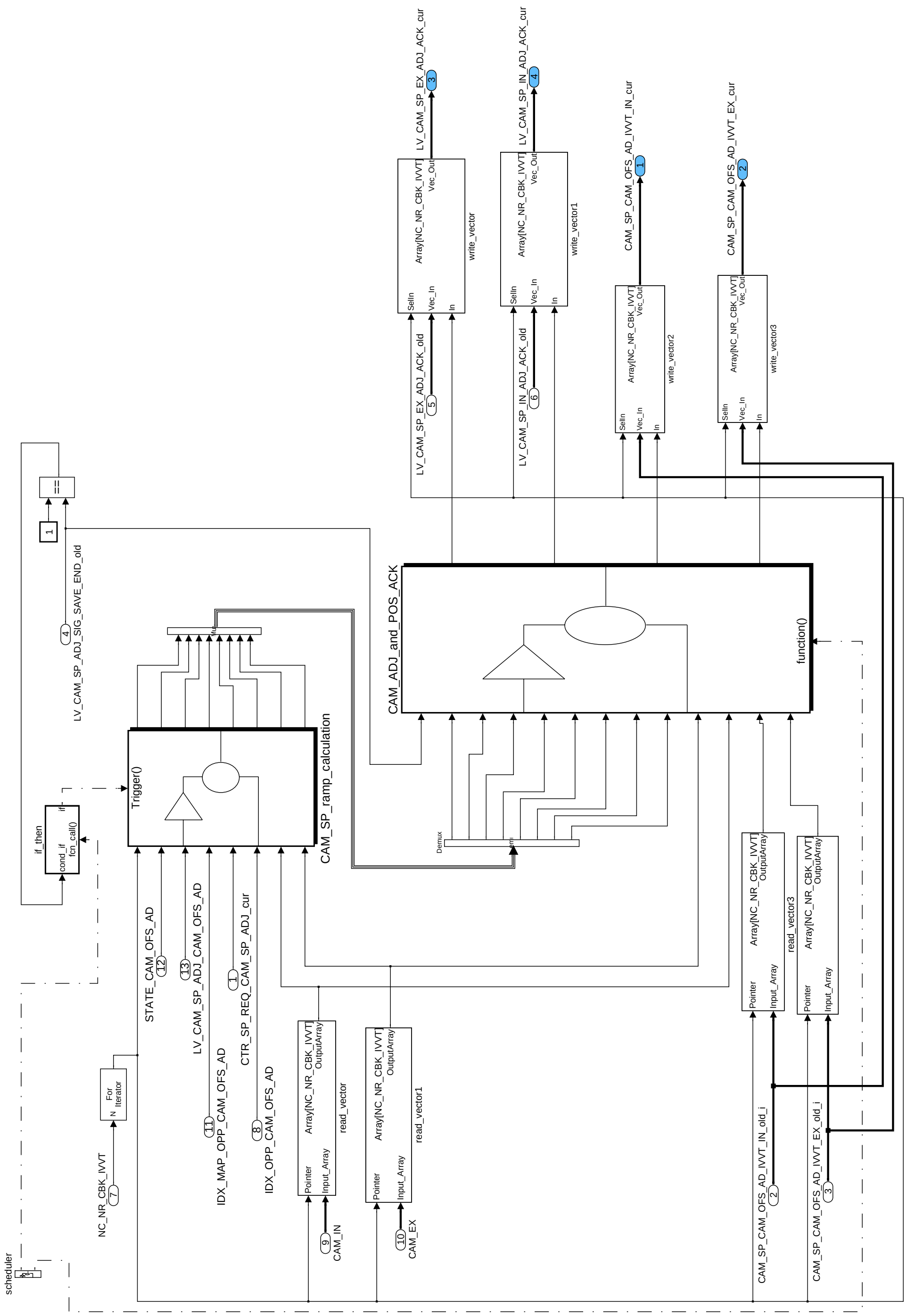


Figure 155 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank

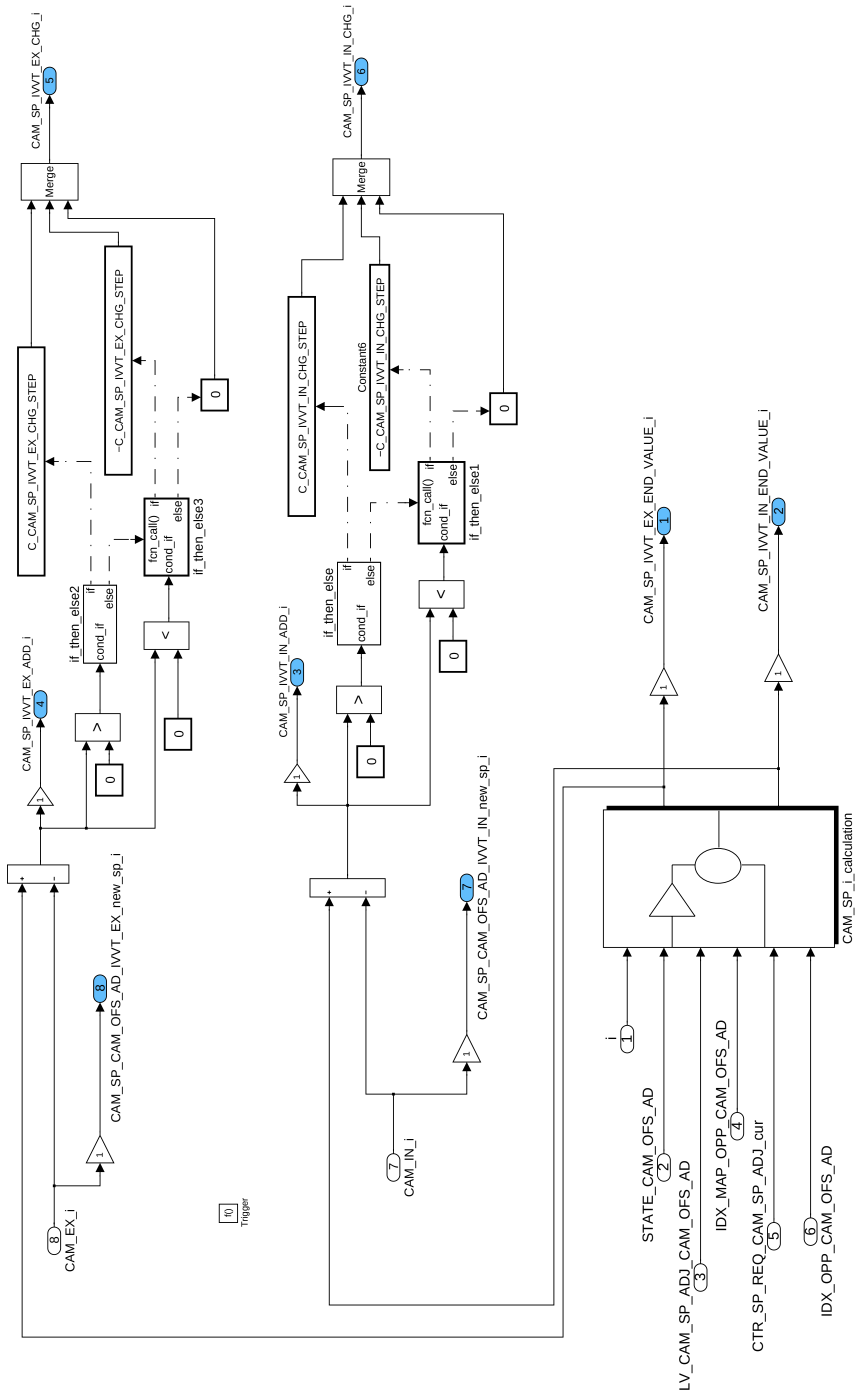


Figure 156 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank/

CAM_SP_ramp_calculation

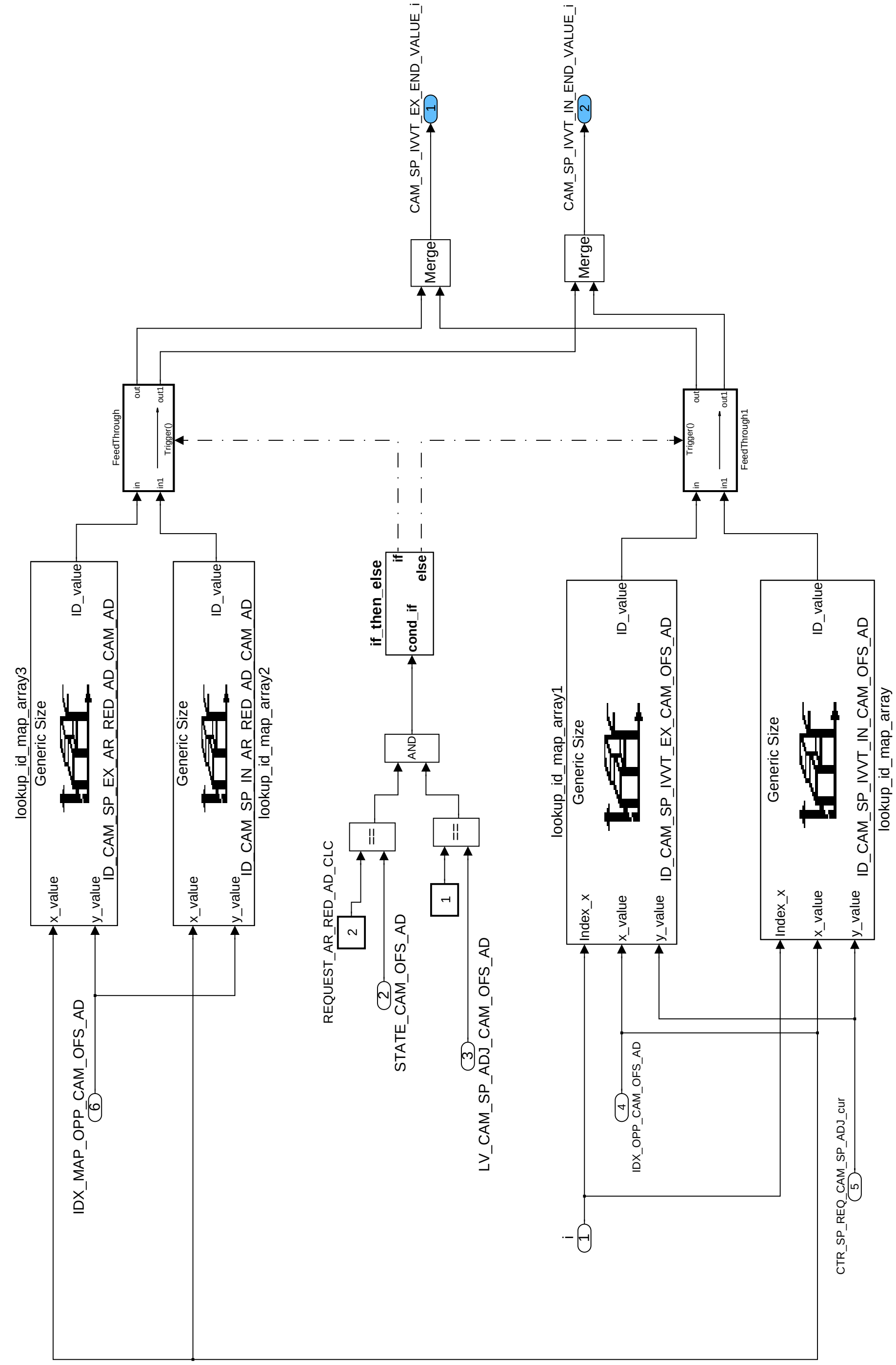


Figure 157 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank/ CAM_SP_ramp_calculation/ CAM_SP_i_calculation

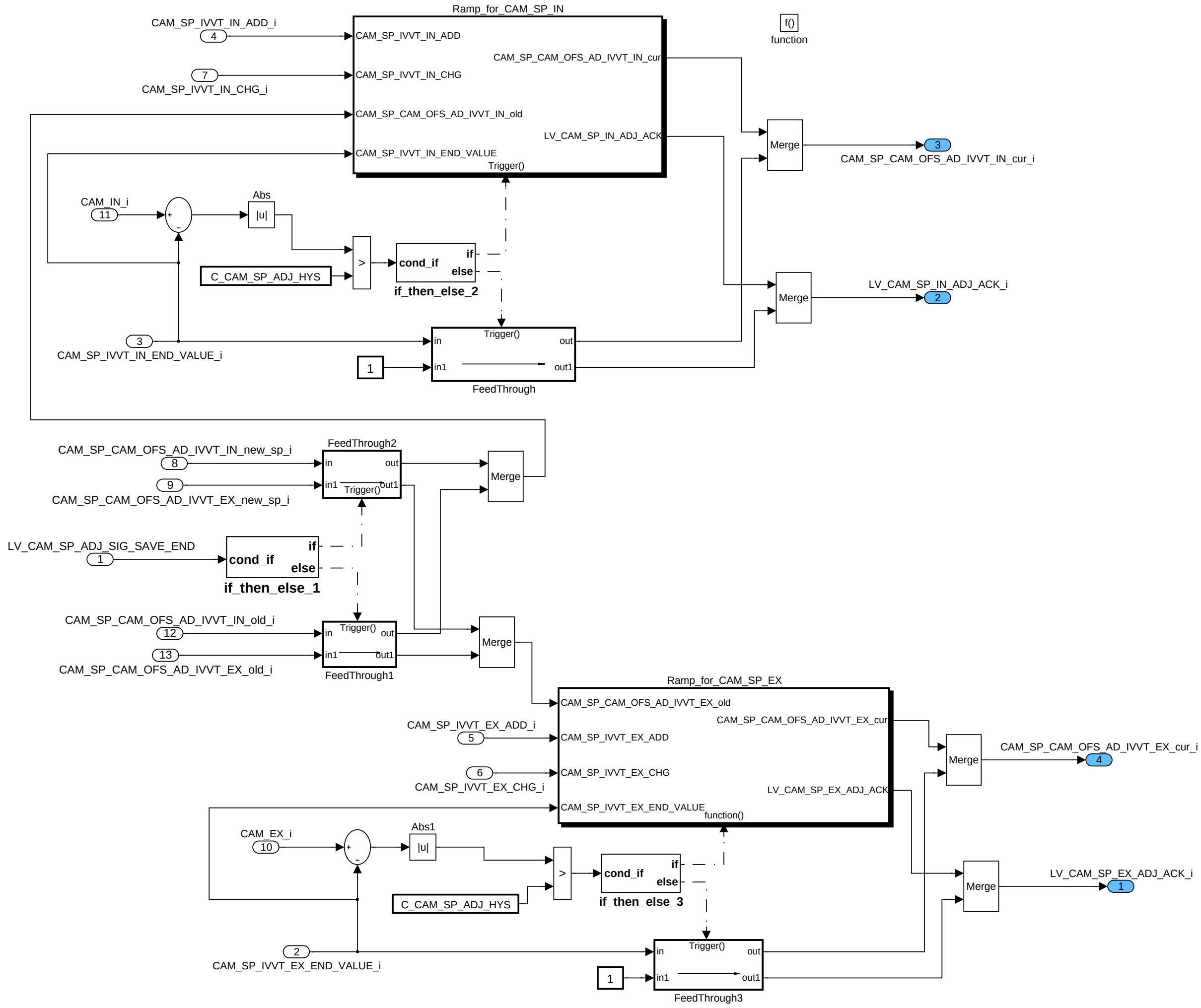


Figure 158 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank/ CAM_ADJ_and_POS_ACK

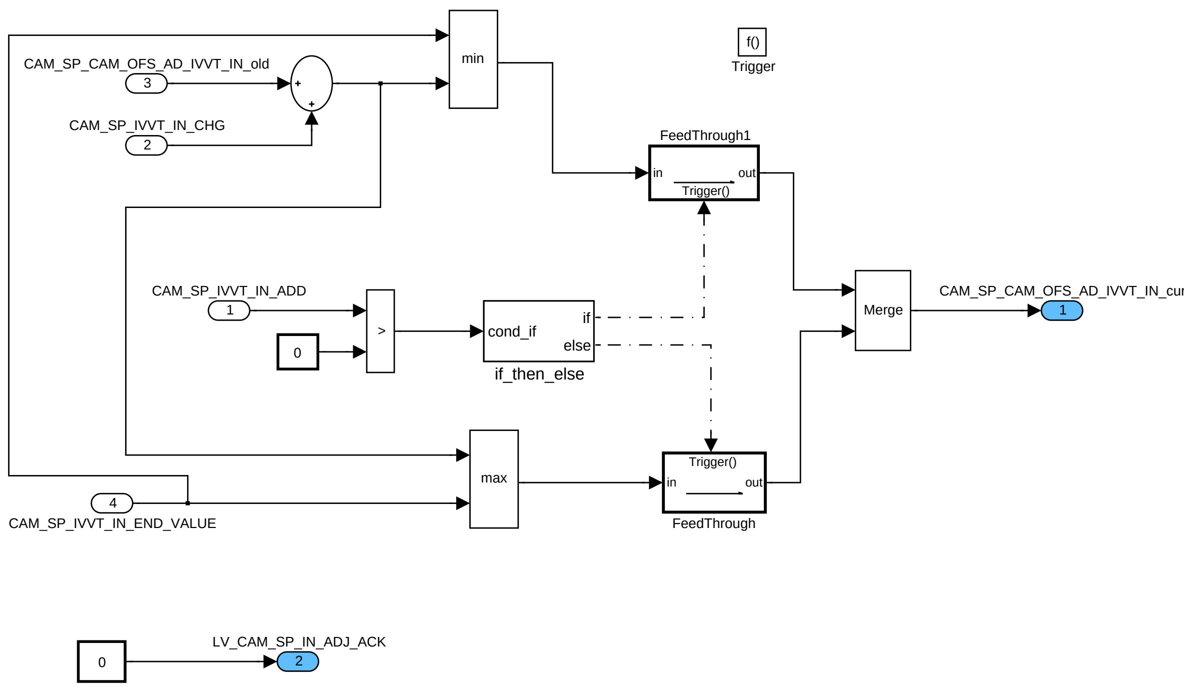


Figure 159 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank/ CAM_ADJ_and_POS_ACK/ Ramp_for_CAM_SP_IN

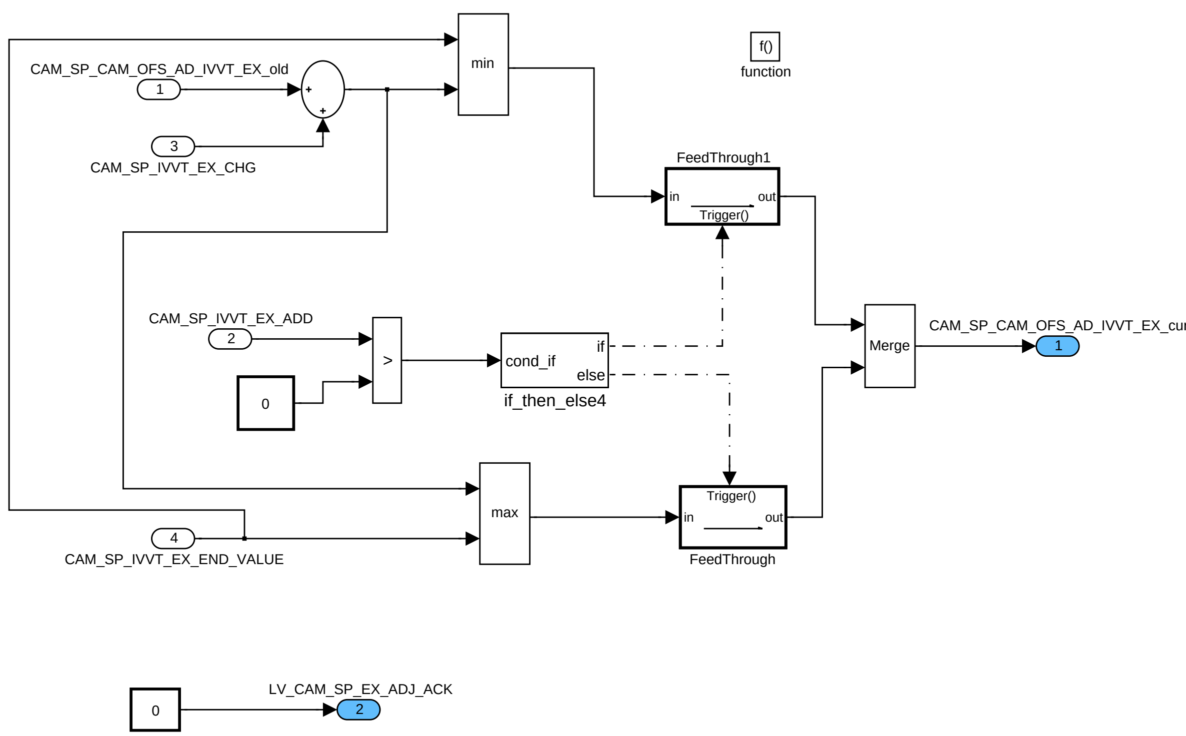


Figure 160 INSY_M908F/ CAM_OFS_AD_ACTIVE/ CAM_SP_CLC_and_CAM_ADJ/ CAM_SP_CLC_for_each_cylinder_bank/ CAM_ADJ_and_POS_ACK/ Ramp_for_CAM_SP_EX

9.48 Optimisation algorithm for Camshaft offset adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAM_MV_IN_REQ_CAM_OFS_AD[3]	O/V	0...FFH	60...155.625	0.375	°CRK
Vector containing CAM_MV_IN related data for EFF_VOL calculation					
CAM_MV_EX_REQ_CAM_OFS_AD[3]	O/V	0...FFH	-40.125...-135.75	-0.375	°CRK
Vector containing CAM_MV_EX related data for EFF_VOL calculation					
N_REQ_CAM_OFS_AD	O/V	0...1FE0H	0...8.16E+3	1	rpm
Engine speed data only for EFF_VOL calculation of Camshaft-offset adaptation functionality					
LV_EFF_VOL_CLC_CAM_OFS_AD	O/V	0...1H	0...1	1	-
Logical bit indicating EFF_VOL calculation for camshaft-offset adaptation functionality ongoing					
LV_CAM_OFS_AD_OPTM_END	O/V	0...1H	0...1	1	-
Logical bit indicating whether optimisation algorithm is over or not					
CAM_OFS_IVVT_EX_OPTM_OUT	O/V	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Result of current optimisation iteration for parameter CAM_OFS_IVVT_EX					
CAM_OFS_IVVT_IN_OPTM_OUT	O/V	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Result of current optimisation iteration for parameter CAM_OFS_IVVT_IN					
FAC_PORT_DEAC_MV_REQ_CAM_AD	O/V	0...FFFFH	0...0.99998474	1.52588E-5	-
Value of FAC_PORT_DEAC_MV used only for EFF_VOL calculation of Camshaft-offset adaptation functionality					
FAC_VIM_REQ_CAM_OFS_AD	O/V	0...FFH	0...0.99609375	0.00390625	-
Value of FAC_VIM used only for EFF_VOL calculation of Camshaft-offset adaptation functionality					
MAF_DIF_CAM_OFS_AD_SQ_TOT	O/V	0...FFFFH	0...255.996094	0.00390625	(kg/h)²
Summation of MAF_DIF_CAM_OFS_AD_SQ over all camshaft positions used for adaptation					
CAM_MV_IN_CAM_OFS_OPTM	O/V	0...FFH	60...155.625	0.375	°CRK
CAM_IN value for optimisation - calculated from vector CAM_IN_MV_CAM_OFS_AD					
CAM_MV_EX_CAM_OFS_OPTM	O/V	0...FFH	-40.125...-135.75	-0.375	°CRK
CAM_EX value for optimisation - calculated from vector CAM_EX_MV_CAM_OFS_AD					
VO_CAM_OFS_AD	O/V	0...1C7H	0...170.625	0.375	°CRK
Valve overlap at each camshaft position which is adjusted during the adaptation					
EFF_VOL_OFS_DRV1_CAM_IN	V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_OFS_DRV1_CAM_EX	V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_SLOP_BAS_DRV1_CAM_IN	V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_BAS over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
CTR_CYC_OPTM_CAM_OFS_AD	V	0...FFFFH	0...6.5535E+4	1	-
Counter for number of optimisation iterations					
EFF_VOL_OFS_BAS_DRV1_CAM_EX	V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_BAS over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_OFS_BAS_DRV1_CAM_IN	V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of EFF_VOL_OFS_BAS over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
EFF_VOL_PRS_EX_COR_CAM_OFS_AD	V	8000...7FFFH	-100 ... 99.9969482	0.00305176	kg/h
EFF_VOL_PRS_EX_COR for camshaft-offset adaptation functionality					
EFF_VOL_FLOW_COR_CAM_OFS_AD	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
EFF_VOL_FLOW_COR for adaptation functionality					
EFF_VOL_SLOP_DRV1_CAM_EX	V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_SP_REQ_CAM_OFS_AD	V	0...CH	0...12	1	-
Counter over number of camshaft-setpoints being used for adaptation					
EFF_VOL_SLOP_DRV1_CAM_IN	V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP over optimisation parameter CAM_OFS_IVVT_IN_OPTM_OUT					
MAF_DIF_SQ_DRV1_TOT_CAM_IN	V	8000...7FFFH	-170.666667 ... 170.661458	0.00520833	(kg/h)²/°CRK
Derivative of total squared error over optimisation paramter CAM_OFS_IVVT_IN_OPTM_OUT					
MAF_DIF_SQ_DRV1_TOT_CAM_EX	V	8000...7FFFH	-170.666667 ... 170.661458	0.00520833	(kg/h)²/°CRK
Derivative of total squared error over optimisation paramter CAM_OFS_IVVT_EX_OPTM_OUT					
MAF_DIF_CAM_OFS_AD	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	kg/h
Difference between MAF_THR_CAM_OFS_AD and MAF_CYL_CAM_OFS_AD each camshaft-position used for adaptation					
PQ_MAP_PRS_EX_CAM_OFS_AD	V	0...FFFFH	0...1.999969	3.05176E-5	-
PQ_MAP_PRS_EX calculated at each camshaft-position used for adaptation					
EFF_VOL_SLOP_AMP_COR_CAM_OFS_AD	V	0...FFFFH	0...1.99996948	3.05176E-5	-
EFF_VOL_SLOP_AMP_COR for adaptation functionality					
EFF_VOL_TEMP_COR_MMV_CAM_OFS_AD	V	0...FFFFH	0...1.99996948	3.05176E-5	-
EFF_VOL_TEMP_COR_MMV for adaptation functionality					
EFF_VOL_OFS_AMP_COR_CAM_OFS_AD	V	0...FFFFH	0...1.99996948	3.05176E-5	-
EFF_VOL_OFS_AMP_COR for adaptation functionality					
EFF_VOL_OFS_CAM_OFS_AD	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Volumetric efficiency offset calculated for adaptation functionality					
EFF_VOL_SLOP_CAM_OFS_AD	V	0...FFFFH	0...12.0599926	1.84024E-4	kg/(h*hPa)
Volumetric efficiency slope calculated for adaptation functionality					
MAF_CYL_CAM_OFS_AD	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
MAF_CYL calculated at each camshaft-position used for adaptation					
MAF_THR_CAM_OFS_AD	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
MAF_THR calculated at each camshaft-position that is used for the adaptation					
MAP_MES_CAM_OFS_AD	V	0...FFFFH	0...5434	0.082918	hPa
MAP_MES at each camshaft-position used for adaptation					
EFF_VOL_FLOW_COR_DRV1_CAM_EX	V	8000...7FFFH	-42.6666667 ... 42.6653645	0.00130208	(kg/h)/°CRK
Derivative of flow correction over optimisation parameter CAM_OFS_IVVT_EX					
EFF_VOL_VO_COR_CAM_OFS_AD	V	8000...7FFFH	-10.24 ... 10.2396875	3.125E-4	-
EFF_VOL_VO_COR calculated at each camshaft-position used for adaptation					
EFF_VOL_SLOP_BAS_DRV1_CAM_EX	V	8000...7FFFH	-2.01002944 ... 2.0099681	6.13412E-5	(kg/h)/(hPa*°CRK)
Derivative of EFF_VOL_SLOP_BAS over optimisation parameter CAM_OFS_IVVT_EX_OPTM_OUT					
EFF_VOL_VO_COR_DIF_CAM_EX	V	8000...7FFFH	-10.24...10.2397	3.125E-4	-
Difference between EFF_VOL_VO_COR values as a function of CAM_MV_EX with fixed CAM_MV_IN value					
CTR_STEP_DIST_OPTM_CAM_IN	V	0...FFFFH	0...71.983644	0.0010984	(h^2*°CRK^2)/(kg^2)
Counter for optimisation step size of parameter CAM_OFS_IVVT_IN					
CTR_STEP_DIST_OPTM_CAM_EX	-	0...FFFFH	0...71.983644	0.0010984	(h^2*°CRK^2)/(kg^2)
Counter for optimisation step size of parameter CAM_OFS_IVVT_EX					
EFF_VOL_FLOW_COR_DRV1_CAM_IN	-	8000...7FFFH	-42.6666... 42.665364	0.001302083	kg/(h*°CRK)
Derivative of flow correction over optimisation parameter CAM_OFS_IVVT_IN					
LV_SWI_OPTM_CAM_OFS_AD	-	0...1H	0...1	1	-
Logical bit indicating whether part 1(=0) or part 2(=1) of the optimisation algorithm is being calculated					
LV_CAM_OFS_AD_OPTM_CYC_END	-	0...1H	0...1	1	-
Logical bit indicating whether current optimisation iteration is over or not					

Input data:

CAM_IN_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD][NC_NR_CBK_IVVT]	CAM_EX_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD][NC_NR_CBK_IVVT]	N_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	EFF_VOL_SLOP_AMP_COR_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]
EFF_VOL_TEMP_COR_MMV_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	EFF_VOL_OFS_AMP_COR_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	MAP_MES_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	PRS_EX_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]
MAF_THR_MV_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]	AR_RED_AD_ADD_REQ_CAM_OFS_AD[NC_NR_OPP_CAM_OFS_AD]	FAC_PORT_DEAC_MV_CAM_AD[NC_NR_SP_REQ_CAM_OFS_AD]	FAC_VIM_CAM_OFS_AD[NC_NR_SP_REQ_CAM_OFS_AD]
STATE_CAM_OFS_AD	LV_CAM_OFS_AD_OPTM_RST	IDX_OPP_CAM_OFS_AD	EFF_VOL_SLOP_BAS_REQ_CAM_OFS_AD[3]
EFF_VOL_OFS_BAS_REQ_CAM_OFS_AD[3]	NC_NR_CBK_IVVT	C_CAM_OP_IN	C_CAM_OP_EX
NC_NR_SP_REQ_CAM_OFS_AD	IP_EFF_VOL_VO_COR_1[NC_NR_STATE_ACR]	IP_EFF_VOL_PRS_EX_COR_1[NC_NR_STATE_ACR]	NC_NR_OPP_CAM_OFS_AD
STATE_ACR_EFF_VOL_CLC	NC_EFF_VOL_ACR_COR_CONF	EFF_VOL_SLOP_PORT_CAM_AD	EFF_VOL_SLOP_VIM_CAM_AD
EFF_VOL_OFS_PORT_CAM_AD	EFF_VOL_OFS_VIM_CAM_AD	EFF_VOL_OFS_PORT_DRV1_CAM_IN	EFF_VOL_OFS_VIM_DRV1_CAM_IN
EFF_VOL_SLOP_PORT_DRV1_CAM_IN	EFF_VOL_SLOP_VIM_DRV1_CAM_IN	EFF_VOL_SLOP_PORT_DRV1_CAM_EX	EFF_VOL_SLOP_VIM_DRV1_CAM_EX
EFF_VOL_OFS_PORT_DRV1_CAM_EX	EFF_VOL_OFS_VIM_DRV1_CAM_EX	NC_NR_STATE_ACR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CAM_EX_STEP_DIST_OPTM_INI	1	0...FFFFH	0...71.9989014	0.00109863	°CRK²/(kg/h)²
Initialisation value for optimisation step distance of parameter: exhaust camshaft offset					
C_CAM_IN_STEP_DIST_OPTM_INI	1	0...FFFFH	0...71.9989014	0.00109863	°CRK²/(kg/h)²
Initialisation value for optimisation step-distance of parameter: inlet camshaft-offset					
C_CAM_MV_EX_OFS_EFF_VOL_GRD_CLC	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Offset to exhaust camshaft-position for calculation of EFF_VOL gradient					
C_CAM_MV_EX_REQ_CAM_OFS_AD_MAX	1	0...FFH	-40.125 ... -135.75	-0.375	°CRK
Minimum exhaust camshaft position that can be used by the adaptation to calculate EFF_VOL data					
C_CAM_MV_EX_REQ_CAM_OFS_AD_MIN	1	0...FFH	-40.125 ... -135.75	-0.375	°CRK
Minimum exhaust camshaft position that can be used by the adaptation to calculate EFF_VOL data					
C_CAM_MV_IN_OFS_EFF_VOL_GRD_CLC	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Offset to inlet camshaft-position for calculation of EFF_VOL gradient					
C_CAM_MV_IN_REQ_CAM_OFS_AD_MAX	1	0...FFH	60...155.625	0.375	°CRK
Maximum inlet camshaft position that can be used by adaptation to caclulate EFF_VOL data					
C_CAM_MV_IN_REQ_CAM_OFS_AD_MIN	1	0...FFH	60...155.625	0.375	°CRK
Minimum inlet camshaft position that can be used by adaptation to calculate EFF_VOL data					
C_CAM_OFS_IVVT_EX_INI_VALUE	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Starting value for optimisation of exhaust camshaft-position offset					
C_CAM_OFS_IVVT_IN_INI_VALUE	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Starting value for optimisation of inlet camshaft-position offset					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CYCNR_OPTM_CAM_OFS_AD	1	0...FFFFH	0...6.5535E+4	1	-
Number of optimisation iterations for camshaft-offset adaptation					
C_DEC_STEP_DIST_OPTM	1	0...FFFFH	0...71.983644	0.0010984	(h^2*°CRK^2)/(kg^2)
Decrement for optimisation step distance of CAM_OFS_IVVT_IN/EX					
C_NR_SP_REQ_CAM_OFS_AD	1	1...9H	1...9	1	-
Number of camshaft positions being used for adaptation					
C_VO_OFS_EFF_VOL_GRD_CLC	1	8000...7FFFH	-48...47.9985352	0.00146484	°CRK
Offset to valve overlap for calculation of EFF_VOL gradient					

9.48.1 Function Description

In the first part(subsystem Optimisation_1), the data required for calculating MAF_CYL, MAF_THR and the derivates of the squared error(MAF_DIF_CAM_OFS_AD_SQ_TOT) at each camshaft position are prepared. If an iteration is over (LV_CAM_OFS_AD_OPTM_CYC_END =1) , the counter CTR_CYC_OPTM_CAM_OFS_AD is incremented and the optimisation results (CAM_OFS_IVVT_IN(_EX)_OPTM_OUT) are used in the calculation of the CAM variables for the next iteration cycle.

Between the first and second optimisation parts, the modules 'Volumetric efficiency' and 'Camshaft-offset adaptation manager (Appl. Inc.)' are called where the data prepared in the first optimisation part is used to calculate EFF_VOL_OFS/SLOP_BAS_REQ_CAM_OFS_AD. This result is then used in part 2 of this module. In the subsystem Optimisation_2, the algorithm to calculate the air mass related variables and camshaft position error involves several steps:

- Calculation of volumetric efficiencies using basic EFF_VOL data and measured data.
- Calculation of MAF_CYL_CAM_OFS_AD, MAF_THR_CAM_OFS_AD and air-mass flow difference.
- Calculation of squared error derivative over CAM_IN and CAM_EX.

In each iteration, the three steps mentioned above are done for each one of the camshaft positions adjusted during the adaptation. At the end of an iteration, the camshaft position error is learnt.

The optimisation calculates a delta to the inlet and exhaust camshaft position respectively. The third optimisation parameter involves learning all non-camshaft related errors which is done in the specification 'Activation conditions and Data Acquisition for Camshaft-offset adaptation'

Application Condition

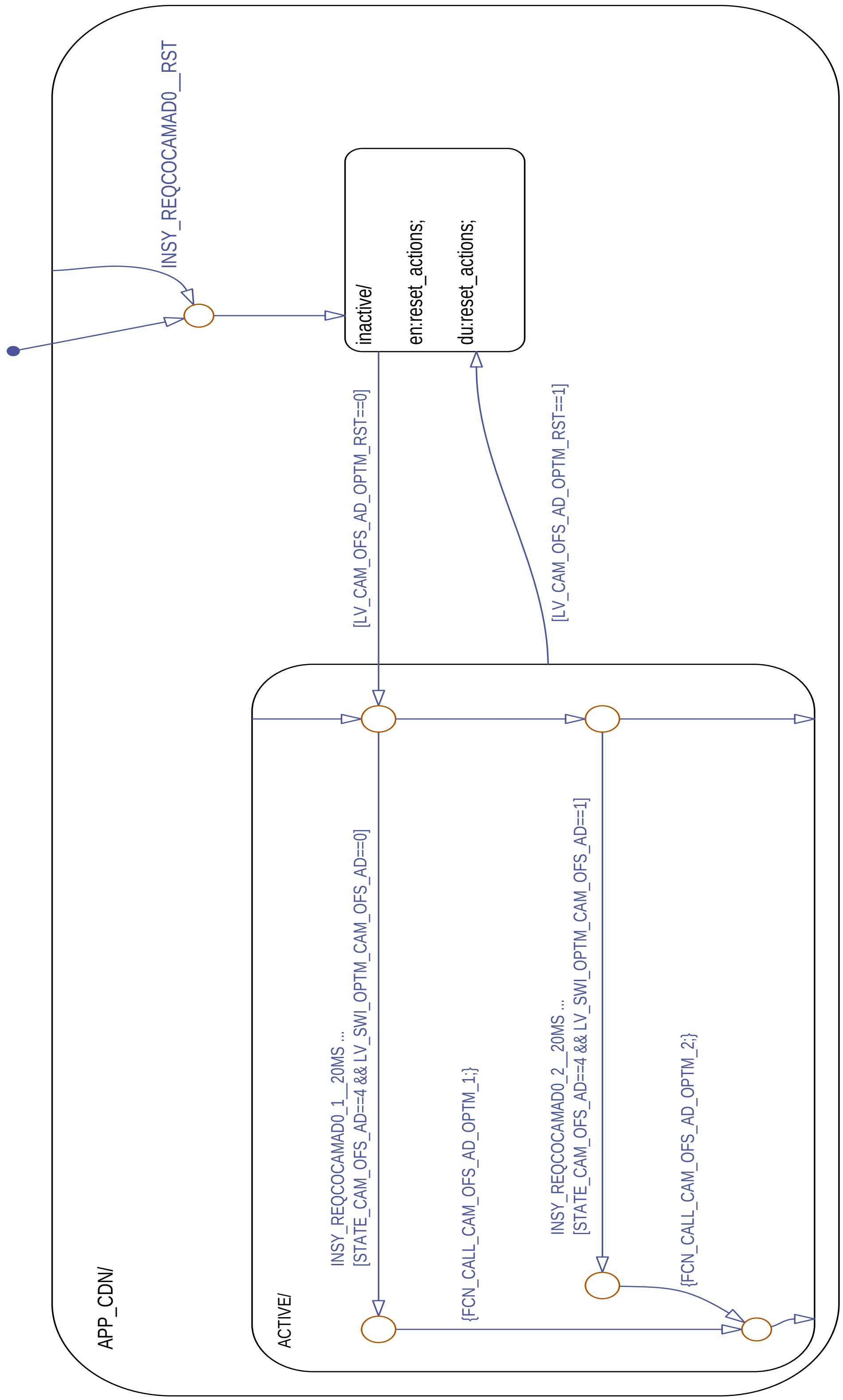
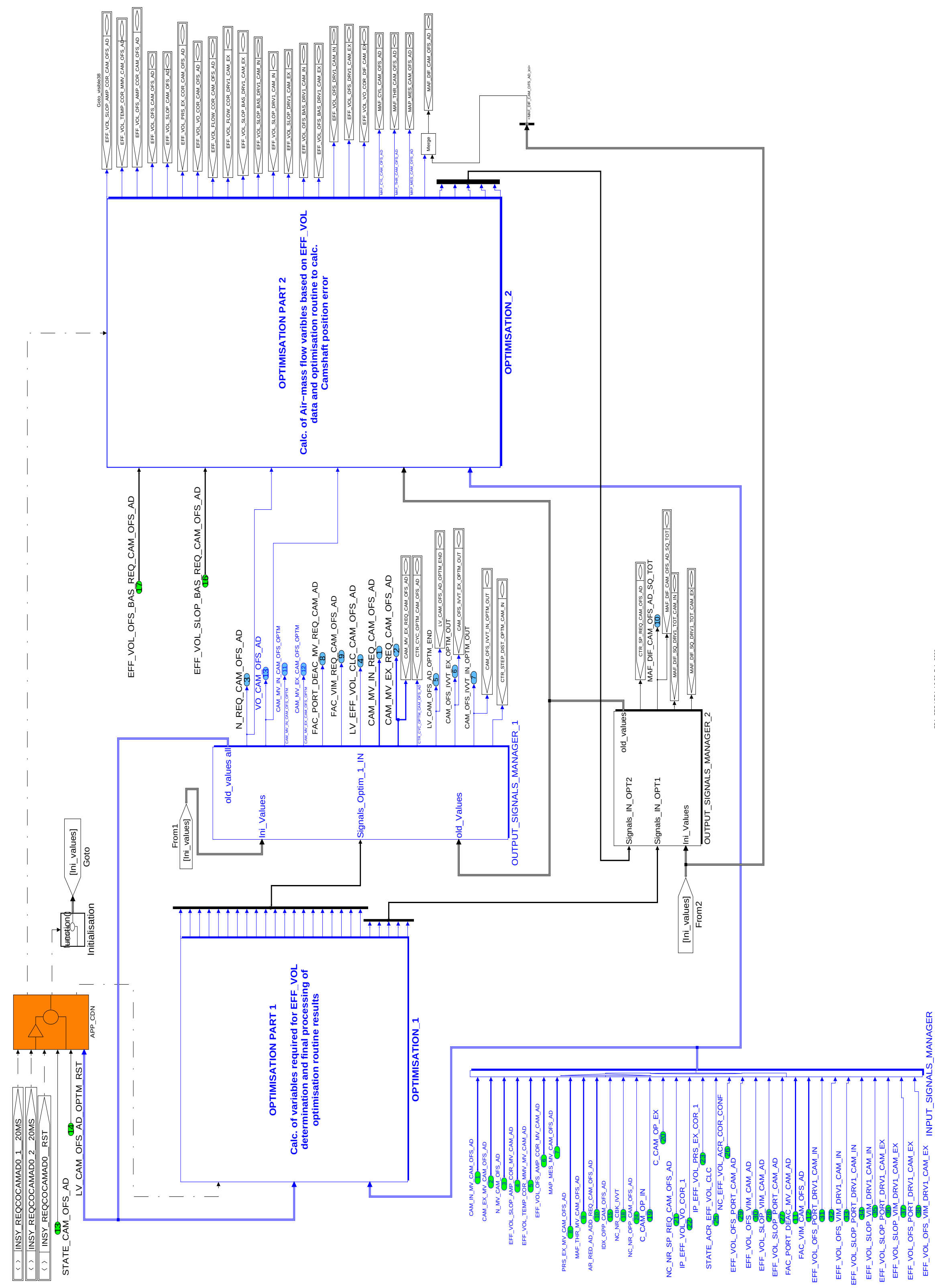

INSY_REQCOCAMAD0__RST
inactive/
en:reset_actions;
du:reset_actions;
[LV_CAM_OFS_AD_OPTM_RST==0]
[LV_CAM_OFS_AD_OPTM_RST==1]
INSY_REQCOCAMAD0_1__20MS ...
[STATE_CAM_OFS_AD==4 && LV_SWI_OPTM_CAM_OFS_AD==0]
{FCN_CALL_CAM_OFS_AD_OPTM_1;}
INSY_REQCOCAMAD0_2__20MS ...
[STATE_CAM_OFS_AD==4 && LV_SWI_OPTM_CAM_OFS_AD==1]
{FCN_CALL_CAM_OFS_AD_OPTM_2;}
APP_CDN/
ACTIVE/

Figure 161 INSY_REQCOCAMAD/ APP_CDN/ Chart

Function Description



OPTIMISATION PART 2
Calc. of Air-mass flow varibles based on EFF_VOL data and optimisation routine to calc. Camshaft position error
OPTIMISATION_2
OPTIMISATION PART 1
Calc. of variables required for EFF_VOL determination and final processing of optimisation routine results
OPTIMISATION_1
OUTPUT_SIGNALS_MANAGER_1
OUTPUT_SIGNALS_MANAGER_2
INPUT_SIGNALS_MANAGER
Initialisation
APP_CDN
EFF_VOL_OFS_BAS_REQ_CAM_OFS_AD
EFF_VOL_SLOP_BAS_REQ_CAM_OFS_AD
STATE_CAM_OFS_AD
LV_CAM_OFS_AD_OPTM_RST

Figure 162 INSY_REQCOCAMAD

9.48.1.1 SUBFUNCTION: Initialisation

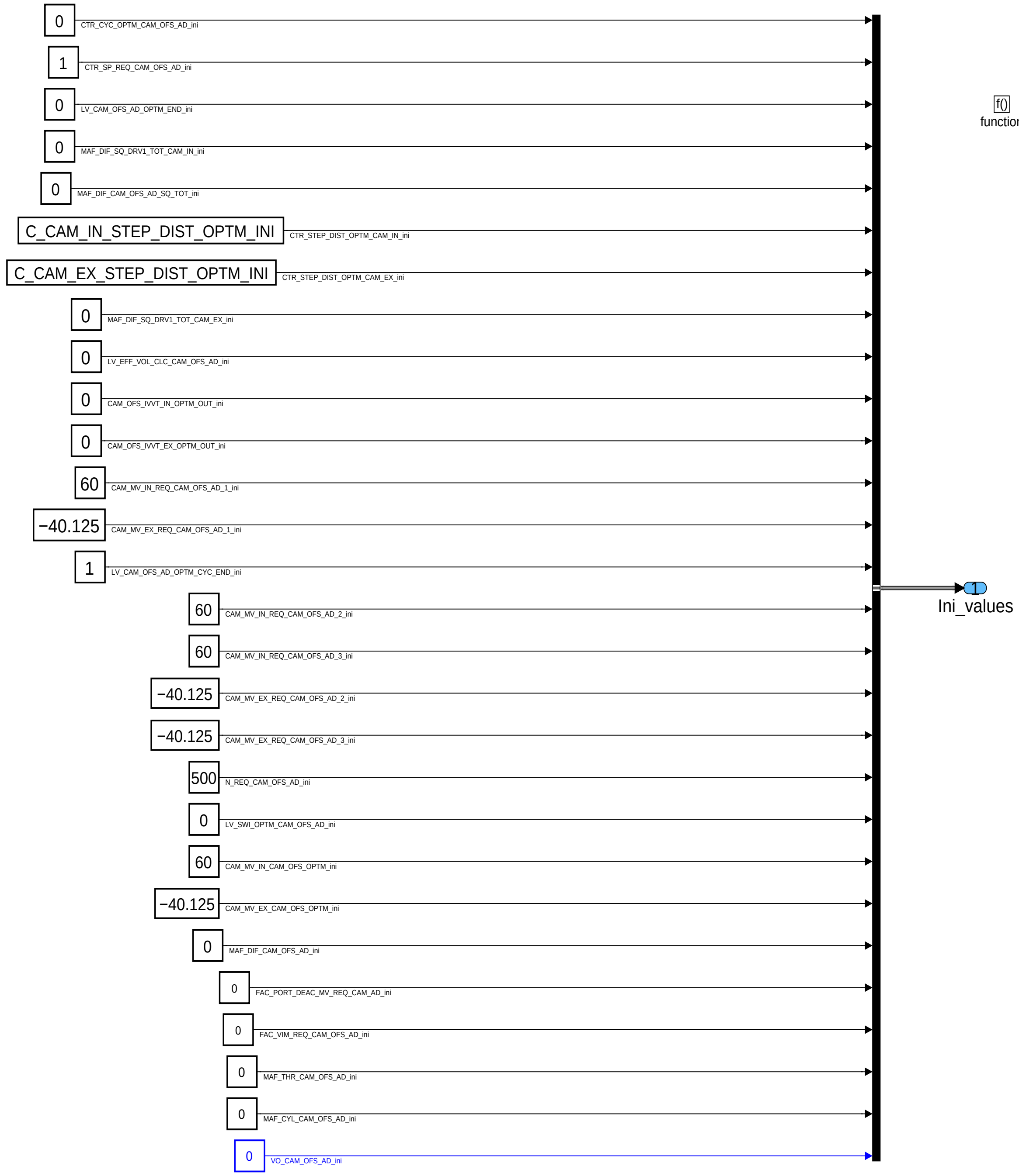


Figure 163 INSY_REQCOCAMAD/ Initialisation

9.48.1.2 SUBFUNCTION: Optimisation_1

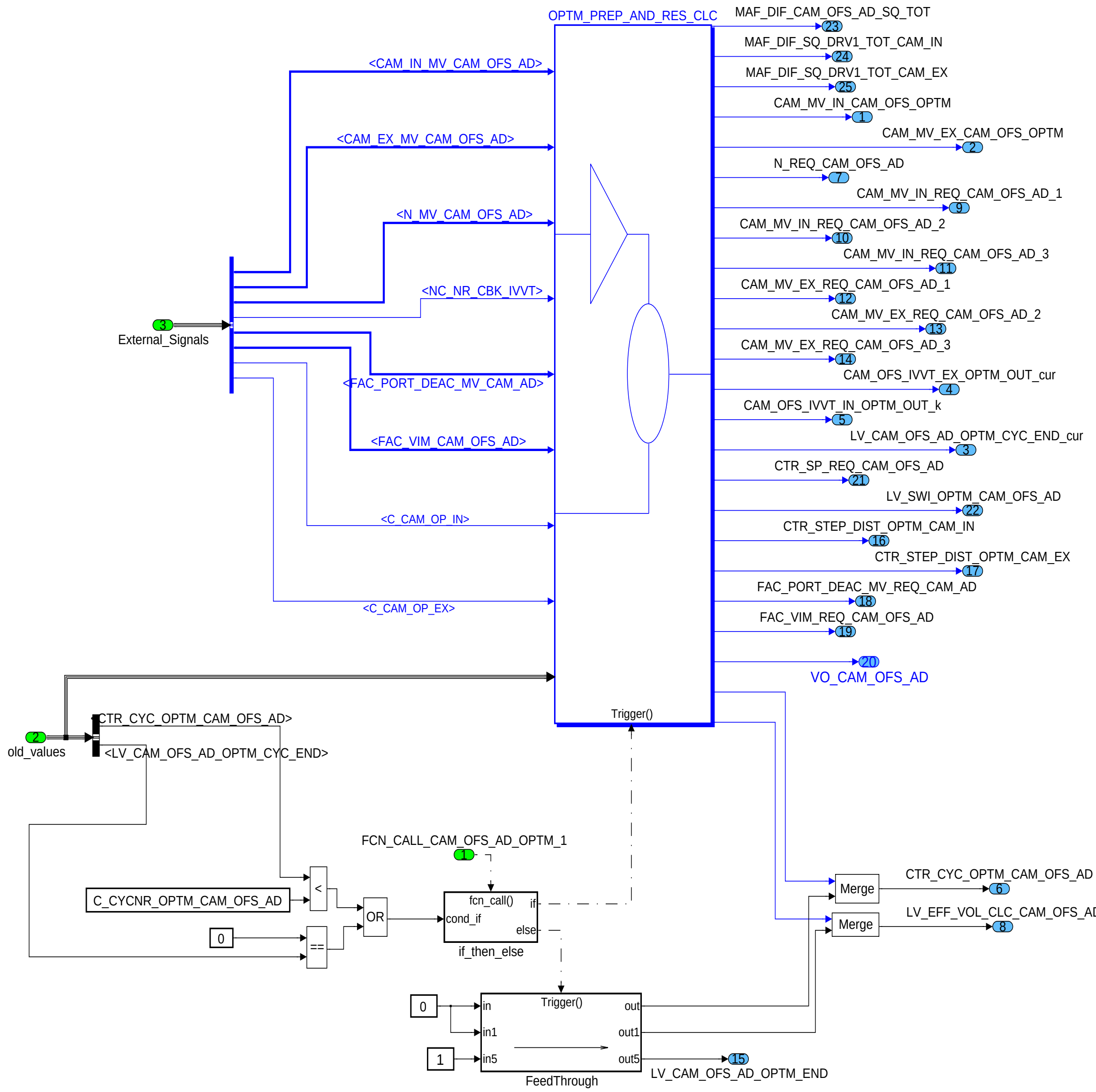


Figure 164 INSY_REQCOCAMAD/ OPTIMISATION_1

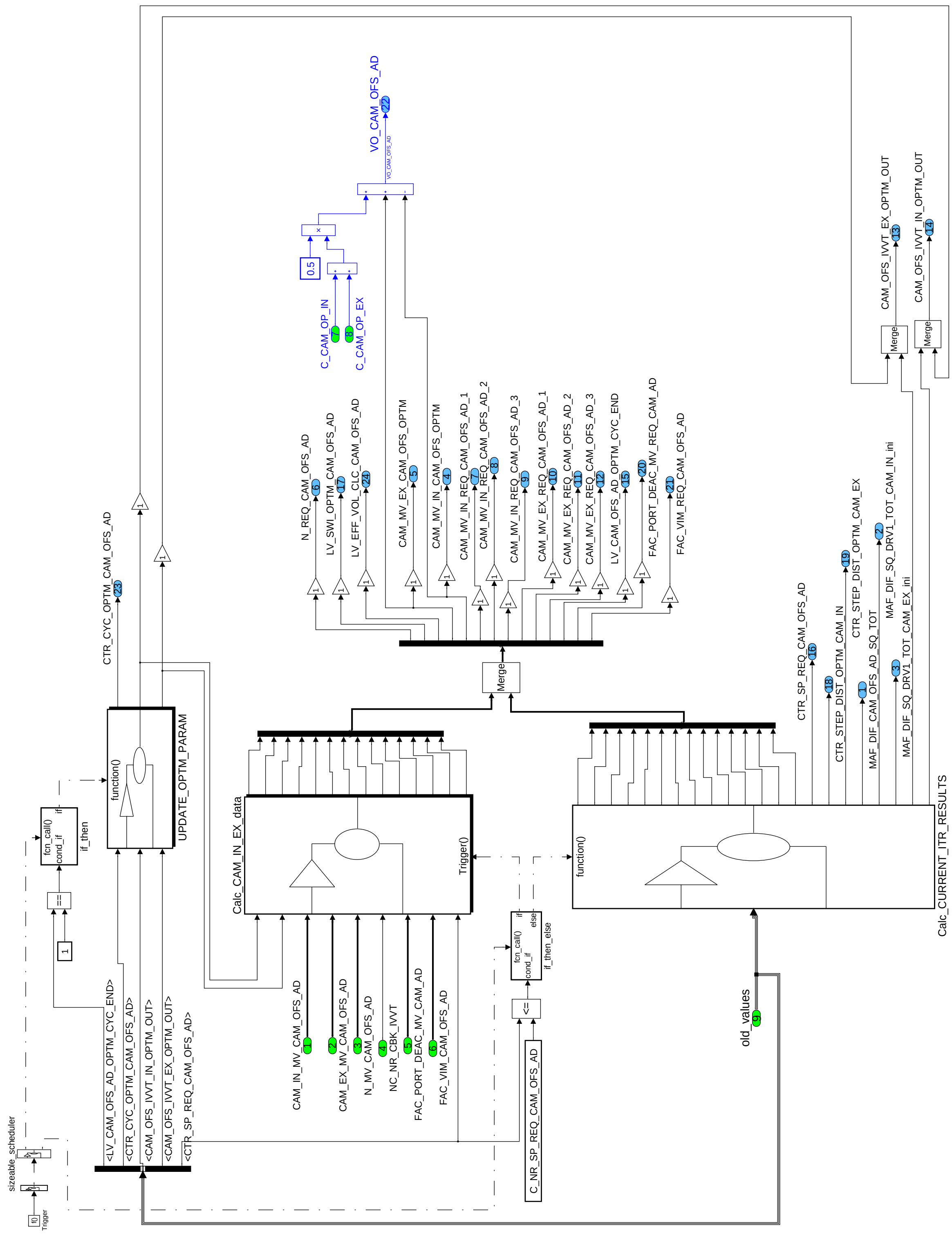


Figure 165 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC

Initialisation of optimisation related variables before start of current iteration

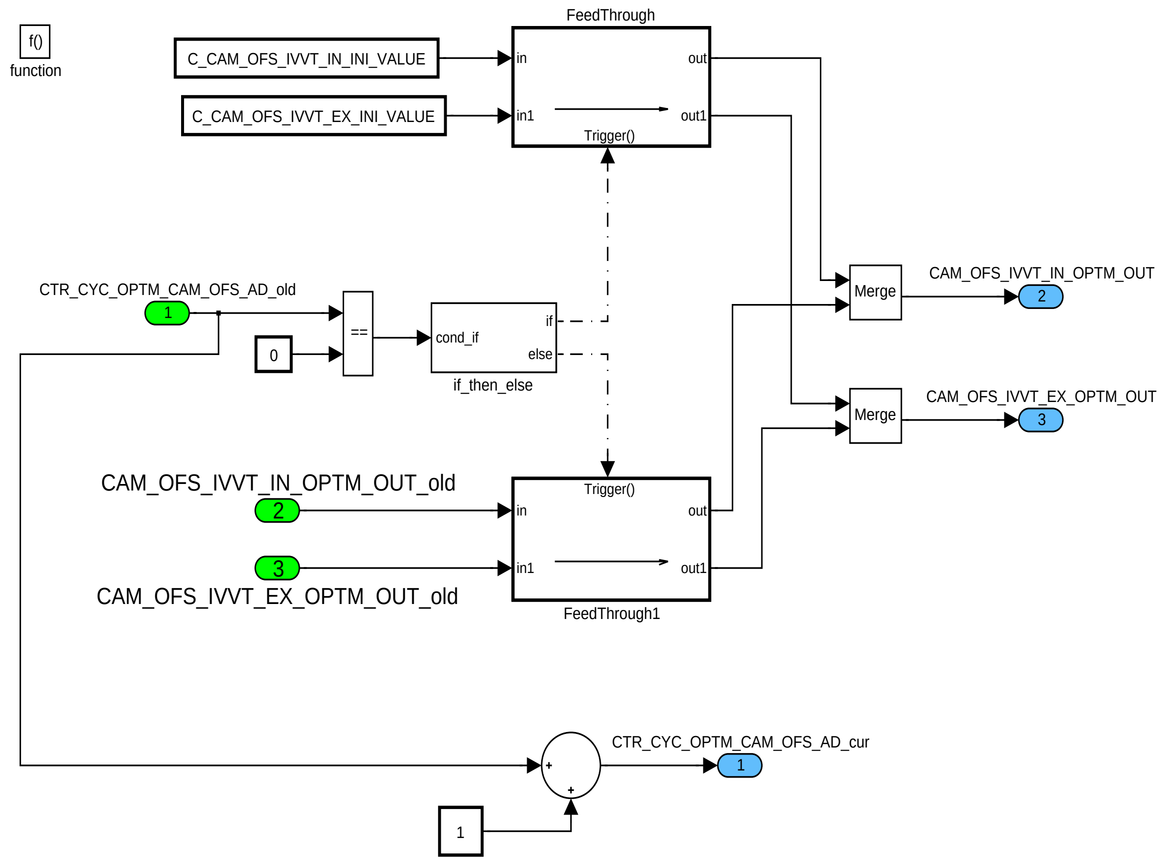


Figure 166 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ UPDATE_OPTM_PARAM

Calculation of CAM_IN/EX data for EFF_VOL calculation

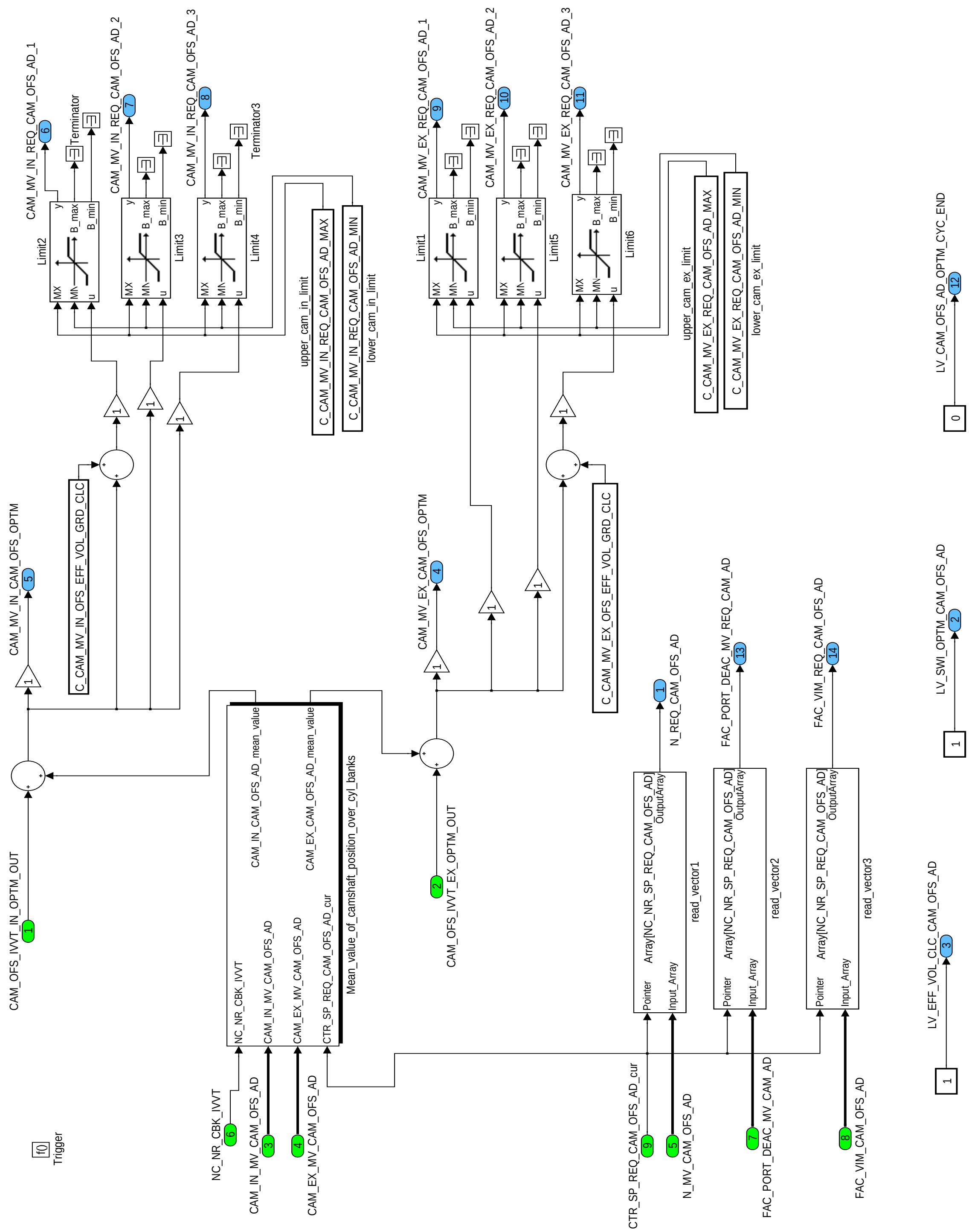


Figure 167 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ Calc_CAM_IN_EX_data

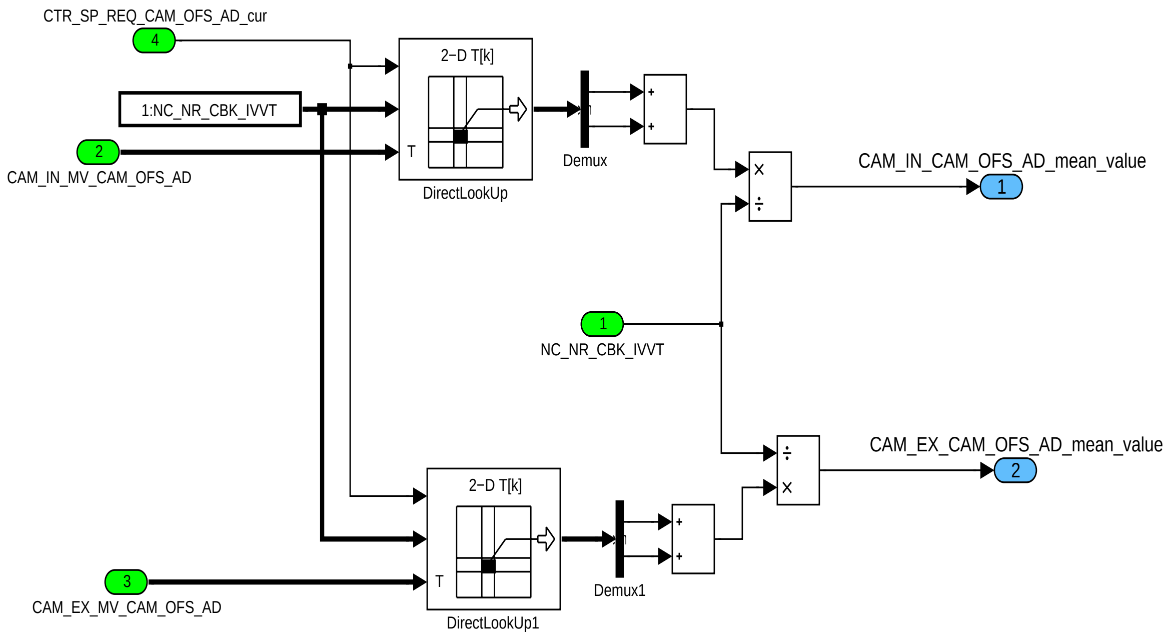


Figure 168 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ Calc_CAM_IN_EX_data/ Mean_value_of_camshaft_position_over_cyl_banks

Calculation of optimisation results and initialisation for next iteration

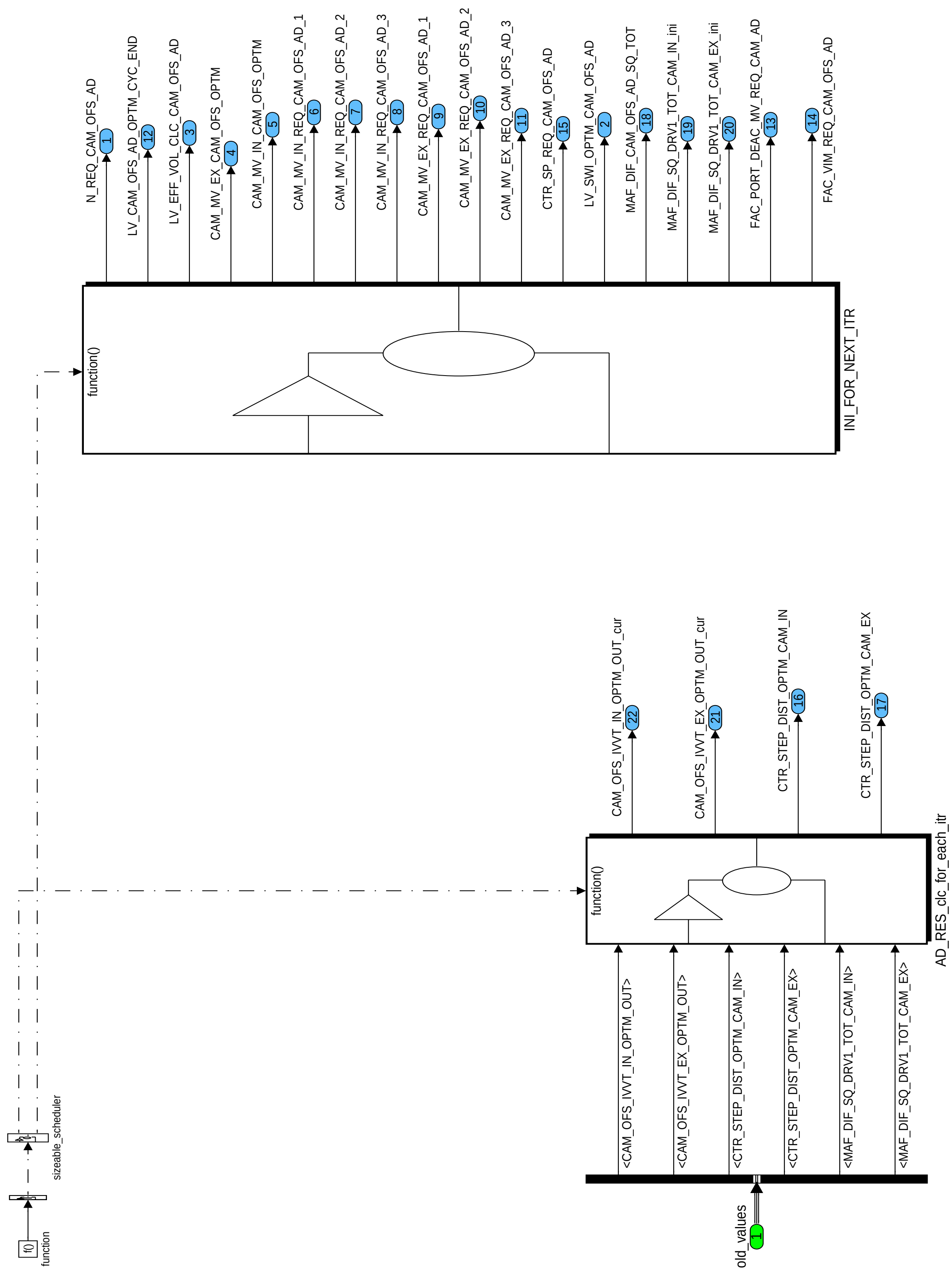


Figure 169 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ Calc_CURRENT_ITR_RESULTS

Calculation of inlet&exhaust camshaft position error at the end of each optimisation iteration

At the end of each optimisation routine, the camshaft-position error is calculated based on the least-sqaured error calculated in the subsystems in the block OPTIMISATION_2. It is based on the following mathematical relationship:

$$Param_i = Param_{i-1} - (2 * optimisation\_step\_size * \sum(e * \frac{de}{dparam}))$$

where i is the iteration number, e=MAF_DIF_CAM_OFS_AD_SQ, optimsation_stepsize = CTR_STEP_DIST_OPTM_IN/EX and the term

$$\sum(e * \frac{de}{dparam}) = MAF\_DIF\_SQ\_DRV1\_TOT\_IN/EX$$

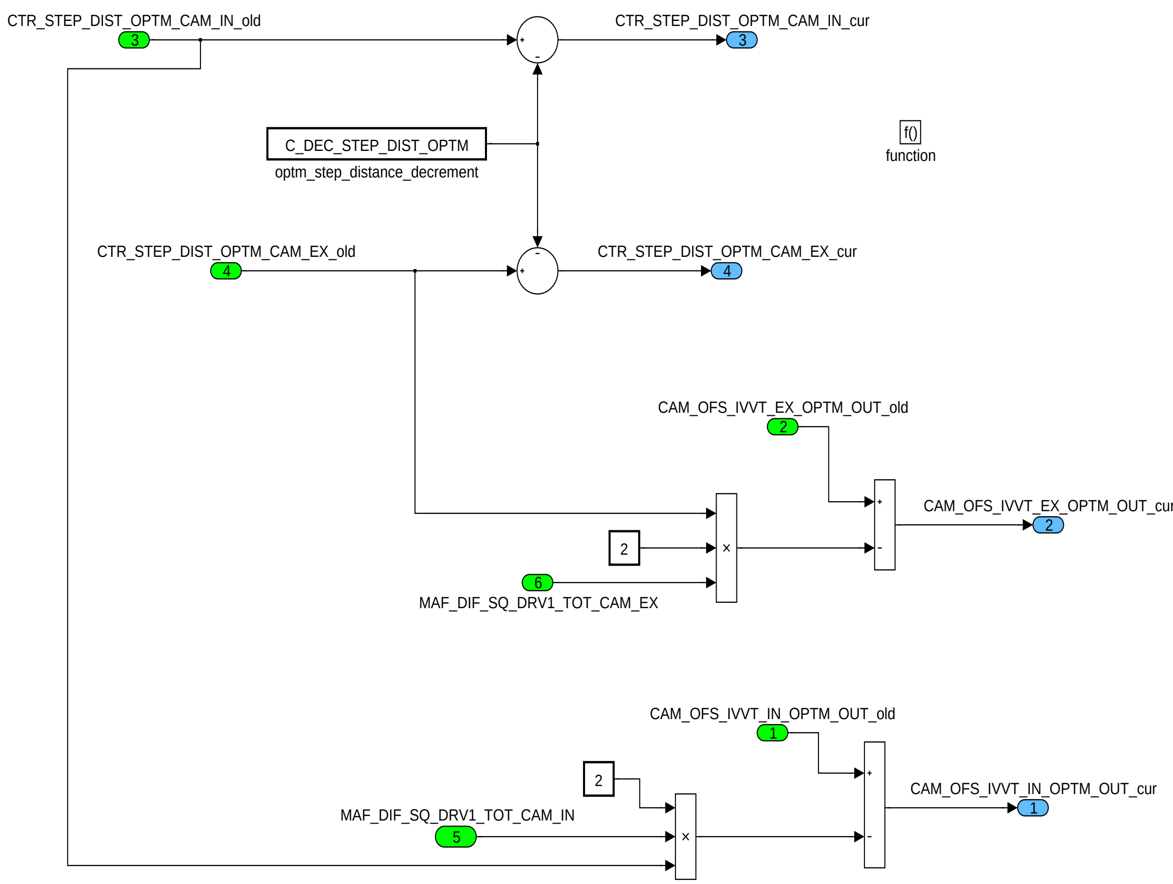


Figure 170 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ Calc_CURRENT_ITR_RESULTS/ AD_RES_clc_for_each_itr

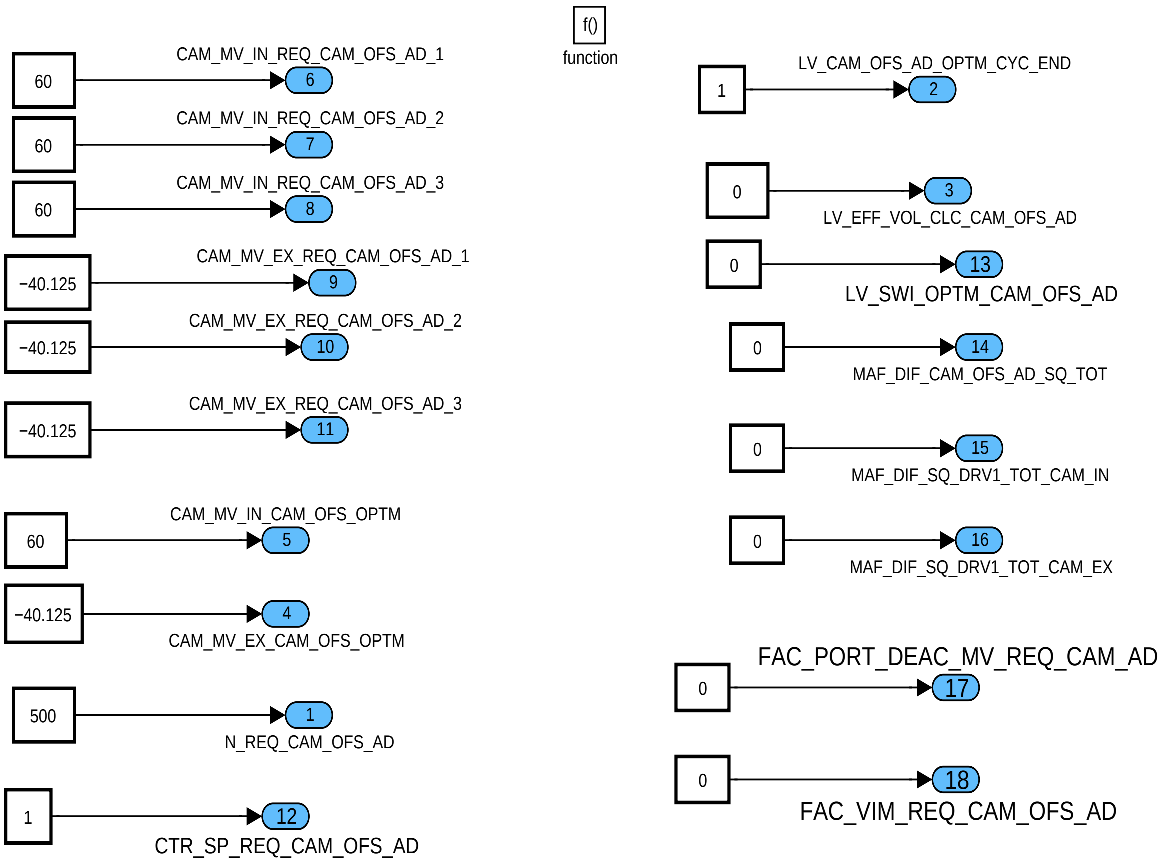


Figure 171 INSY_REQCOCAMAD/ OPTIMISATION_1/ OPTM_PREP_AND_RES_CLC/ Calc_CURRENT_ITR_RESULTS/ INI_FOR_NEXT_ITR

9.48.1.3 SUBFUNCTION: Optimisation_2

Optimisation iterations calculation using results from part 1 (CAM_MV_IN/EX_CAM_OFS_ OPTM etc.) and those delivered by the Volumetric efficiency specification (EFF_VOL_SLOP/OFS_BAS_

REQ_CAM_OFS_AD)

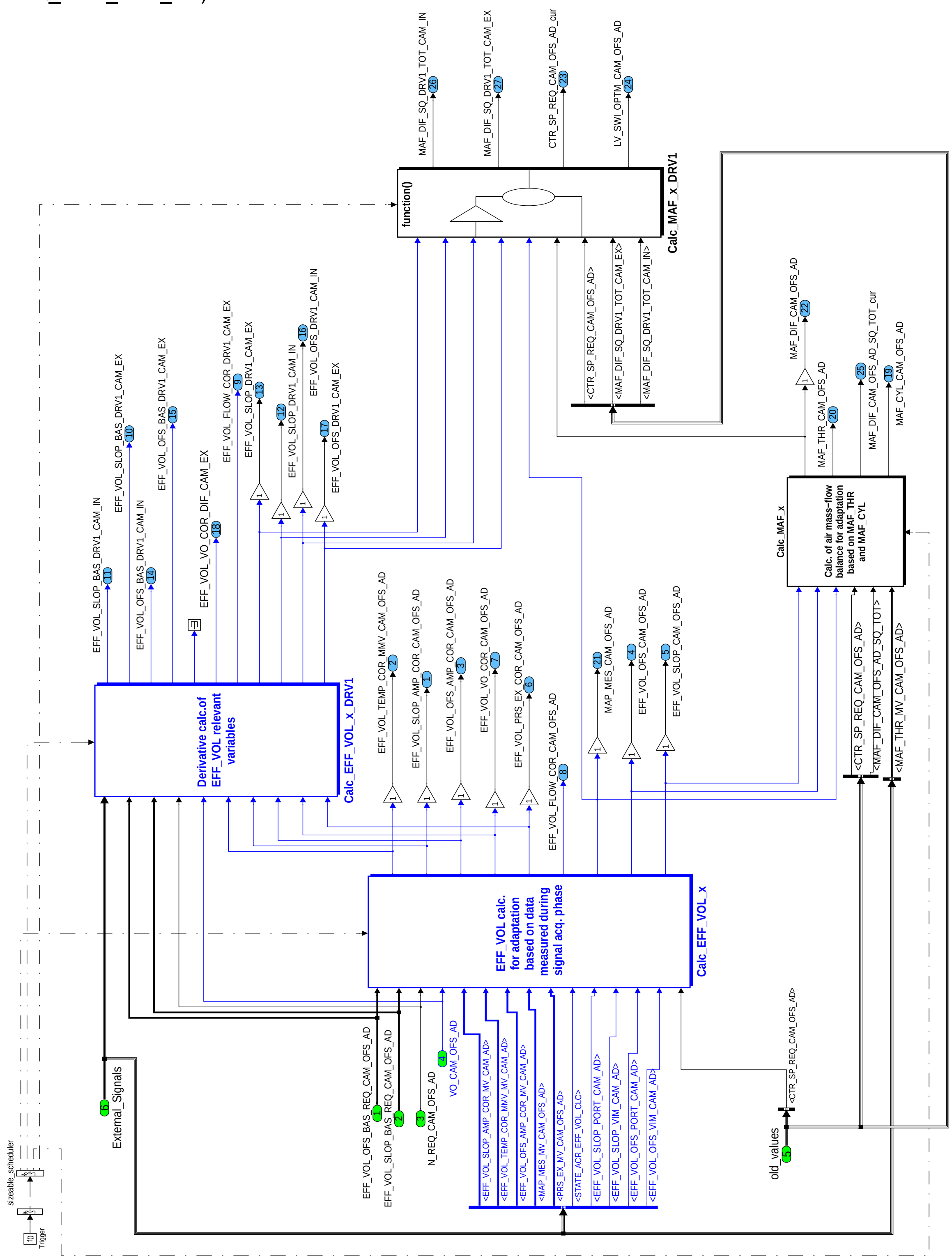


Figure 172 INSY_REQCOCAMAD/ OPTIMISATION_2

Calculation of EFF_VOL related variables required for determination of MAF_CYL_CAM_OFS_AD

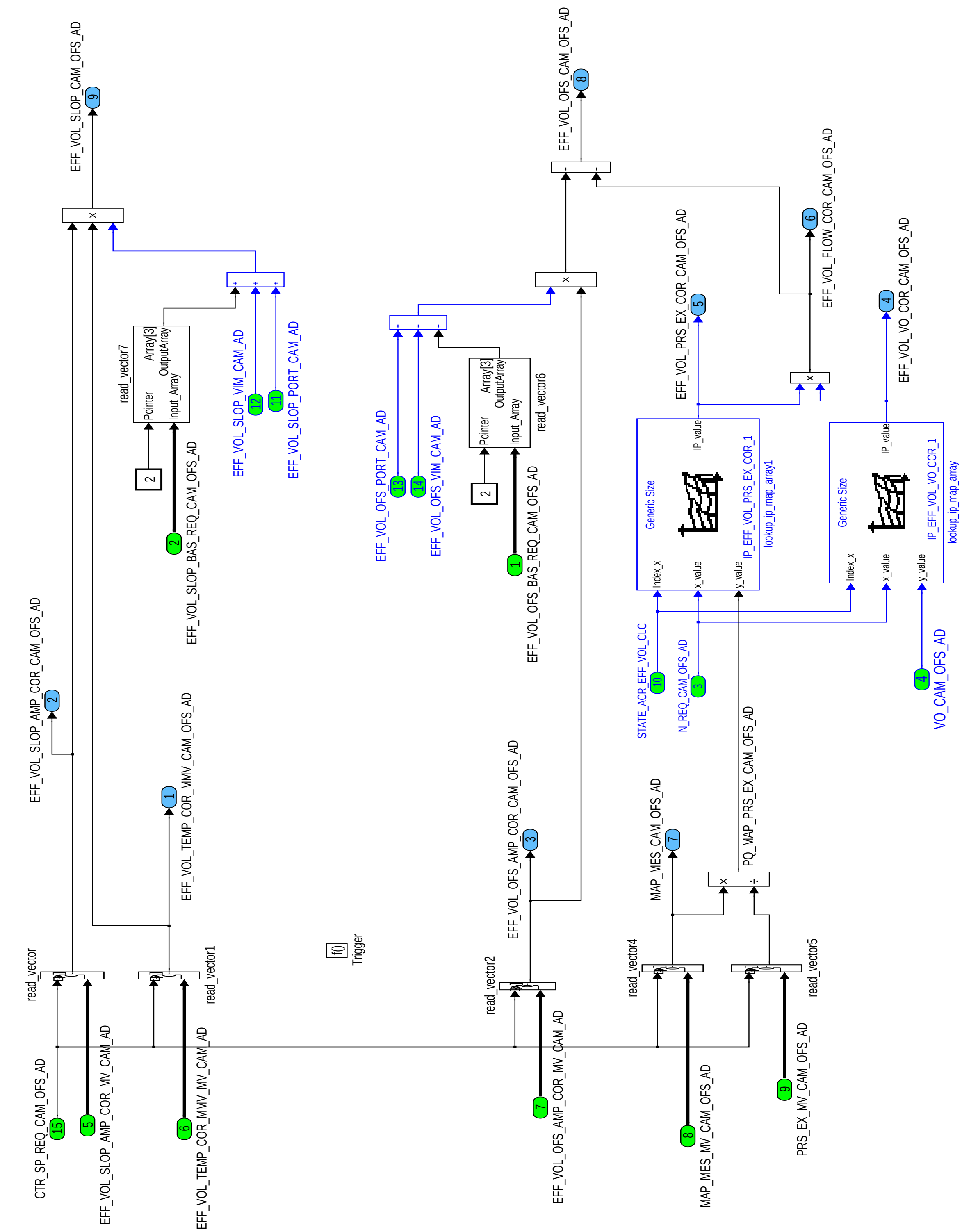


Figure 173 INSY_REQCOCAMAD/ OPTIMISATION_2/ Calc_EFF_VOL_x

Calculation of air-mass flow deviation and related variables

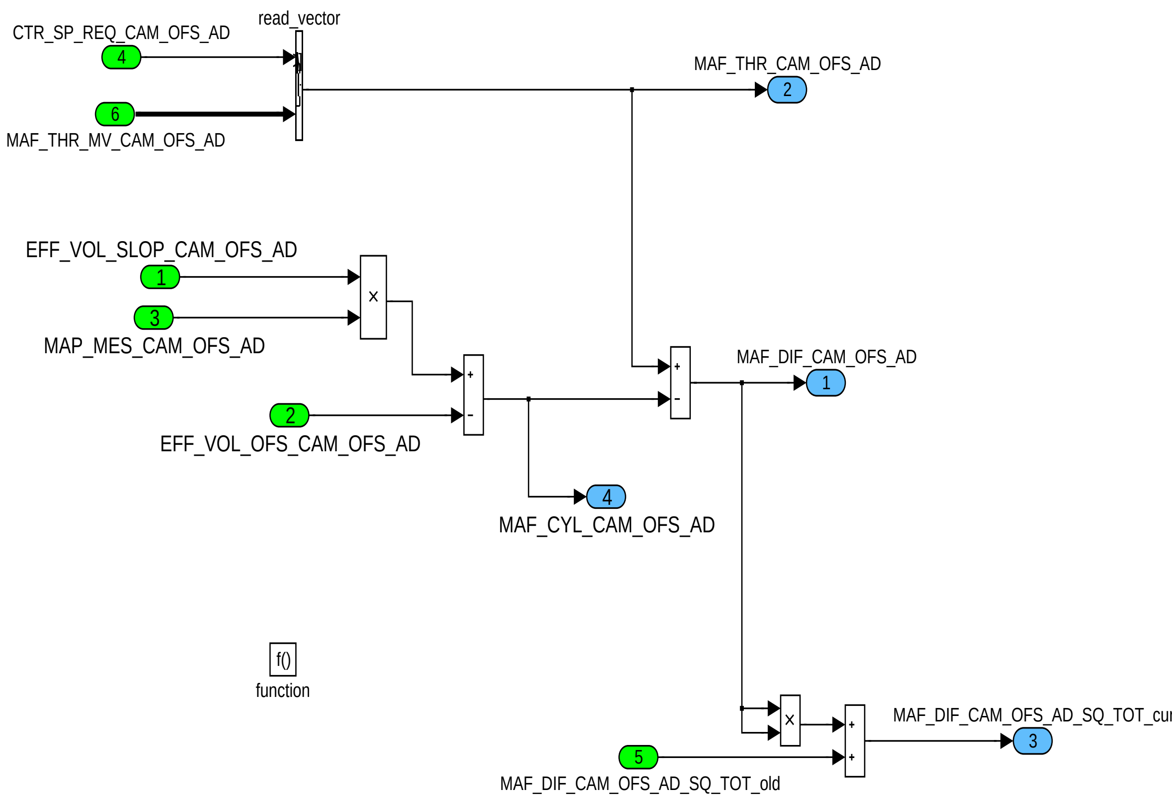


Figure 174 INSY_REQCOCAMAD/ OPTIMISATION_2/ Calc_MAF_x

Calculation of air mass flow derivatives depending on CAM_OFS_IN(_EX)

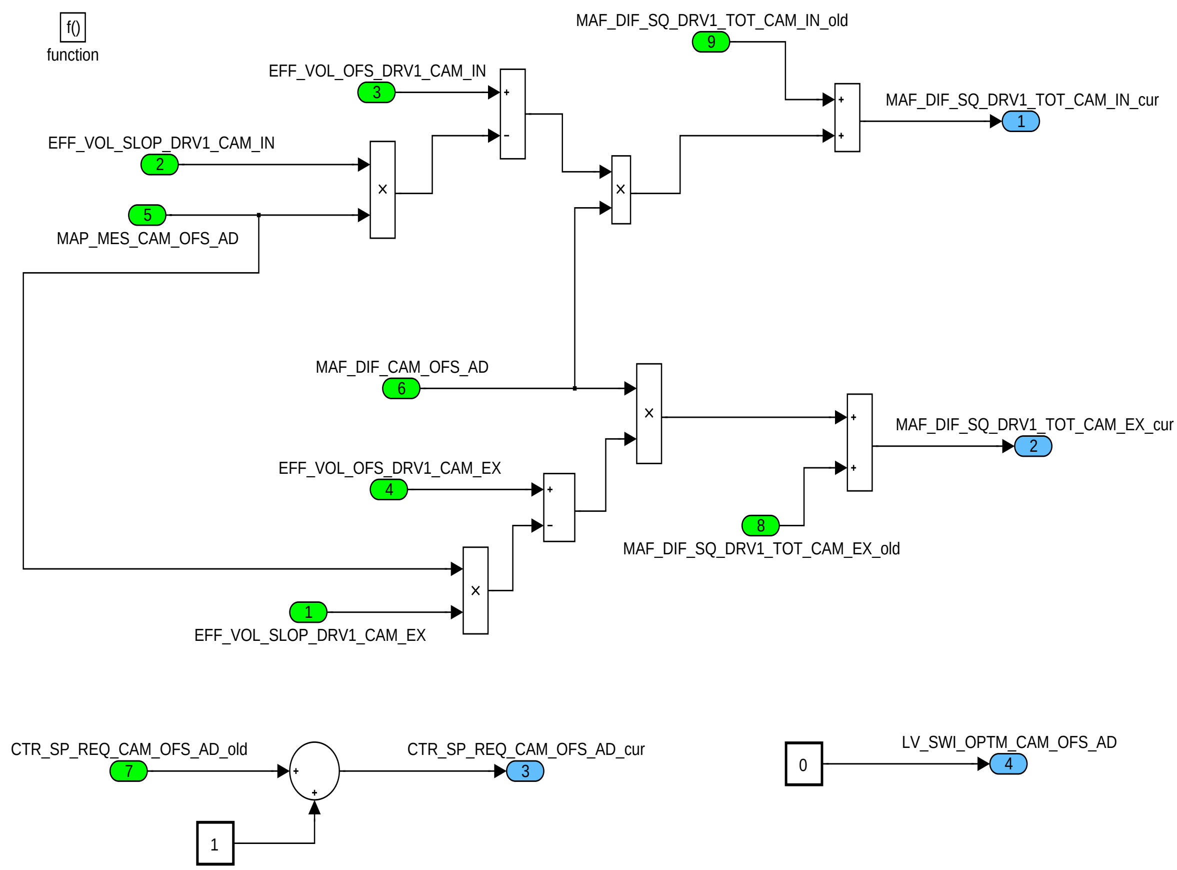


Figure 175 INSY_REQCOCAMAD/ OPTIMISATION_2/ Calc_MAF_x_DRV1

9.49 IVVT Setpoint Premanager

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
STATE_CAM_SP_IVVT_IN	O/V	0H 1H 2H 3H 4H	SP_NOT SP_HOM_AFS SP_HOM_AFL SP_S SP_MAN	1	-
Inlet camshaft position setpoint state					
STATE_CAM_SP_IVVT_EX	O/V	0H 1H 2H 3H 4H	SP_NOT SP_HOM_AFS SP_HOM_AFL SP_S SP_MAN	1	-
Exhaust camshaft position setpoint state					

Input data:

STATE_CMB_CTL	LV_ACT_TOT_IVVT_IN	LV_ACT_TOT_IVVT_EX	

FUNCTION DESCRIPTION:

General information:

The setpoint premanger chooses which type of the setpoint is to take for the current engine state, especially combustion mode or special circumstances. Depending on this choise a setpoint is calculated in a corresponding module if necessary. The calculated or a special setpoint is processed in the setpoint postmanager. Then, it is used in the camshaft position controlling.

Description:

Three combustion modes are considered in an engine. They are: homogeneous stoichiometric "HOM_AFS", homogeneous lean "HOM_AFL" and stratified "S". Each mode has its individual camshaft position setpoints "SP_CMB_HOM_AFS", "SP_CMB_HOM_AFL" and "SP_CMB_S". Each transition among the three combustion modes must adopt one of the three available setpoint sets.

The state STATE_CAM_SP_IVVT_IN(_EX) expresses which type of the setpoint is to take for the current engine state. The setpoint state is "SP_NOT" if the setpoint calculation is not activated. The states "SP_HOM_AFS", "SP_HOM_AFL" and "SP_S" correspond to the three combustion modes.

No calculation of the setpoint for one of the combustion modes is needed if the setpoint is determined by other procedure with a higher priority, i.e. setpoint given by the application manually, "SP_MAN".

Application conditions:

Initialization:

At reset, at deactivation, at transition "Engine run" → "Engine stop":

STATE_CAM_SP_IVVT_IN(_EX) = "SP_NOT"

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

LV_ACT_TOT_IVVT_IN(_EX) = 1

Deactivation:

Not activation

Formula section:

if LC_CAM_SP_MAN_IVVT_IN(_EX) = 1

then *Manual setpoint*

STATE_CAM_SP_IVVT_IN(_EX) = "SP_MAN"

elseif ID_CMB_IVVT = "SP_CMB_HOM_AFS"

then *Setpoint for homogeneous stoichiometric*

STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFS"

elseif ID_CMB_IVVT = "SP_CMB_HOM_AFL"

then *Setpoint for homogeneous lean*

STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFL"

elseif ID_CMB_IVVT = "SP_CMB_S"

then *Setpoint for stratified*

STATE_CAM_SP_IVVT_IN(_EX) = "SP_S"

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_CAM_SP_MAN_IVVT_IN	1	0...1H	0...1	1	-
1 = take manual setpoint, 0 = normal setpoint algorithm; inlet					
LC_CAM_SP_MAN_IVVT_EX	1	0...1H	0...1	1	-
1 = take manual setpoint, 0 = normal setpoint algorithm; exhaust					
ID_CMB_IVVT	9	0...2H	0...2	1	-
LDP_STATE_CMB_CTL_ID_CMB_IVVT	9	0...8H	0...8	1	-
Assigment of camshaft position setpoints (*) to combustion states (**)					

(*) Setpoints for combustion states:

Physical Value	Meaning
0	SP_CMB_HOM_AFS
1	SP_CMB_HOM_AFL
2	SP_CMB_S

(**) Combustion states:

Physical Value	Meaning
0	HOM_AFS
1	AFS_TO_AFL
2	HOM_AFL
3	AFL_TO_AFS
4	HOM_TO_S
5	BACKS
6	S
7	S_TO_HOM
8	BACKHOM

9.50 IVVT Setpoint Postmanager

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_CLC_IVVT_IN	V	0...FFH	60...155.625	0.375	°CRK
Calculated camshaft position setpoint; inlet					
CAM_SP_CLC_IVVT_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
Calculated camshaft position setpoint; exhaust					
CAM_SP_PREL_IVVT_IN	V	0...FFH	60...155.625	0.375	°CRK
Preliminary control camshaft position setpoint; inlet					
CAM_SP_PREL_IVVT_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
Preliminary control camshaft position setpoint; exhaust					
CAM_SP_LGRD_IVVT_IN	V	0...FFH	60...155.625	0.375	°CRK
Help variable for calculation of camshaft position setpoint with limited gradient; inlet					
CAM_SP_LGRD_IVVT_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
Help variable for calculation of camshaft position setpoint with limited gradient; exhaust					
T_CAM_SP_LGRD_IVVT_IN	V	0...FFFFH	0...655350	10	ms
Time at start of limitation of camshaft setpoint gradient; inlet					
T_CAM_SP_LGRD_IVVT_EX	V	0...FFFFH	0...655350	10	ms
Time at start of limitation of camshaft setpoint gradient; exhaust					
CTR_CAM_SP_IVVT_IN	O/V	0...FFH	0...255	1	-
Counter increasing by 1 at every camshaft position setpoint calculation; inlet; with overflow					
CTR_CAM_SP_IVVT_EX	O/V	0...FFH	0...255	1	-
Counter increasing by 1 at every camshaft position setpoint calculation; exhaust; with overflow					

Input data:

STATE_CAM_SP_IVVT_IN	STATE_CAM_SP_IVVT_EX	T_DLY_MES_IVVT	NC_NR_CBK_IVVT
CAM_ADJ_MAX_IVVT_IN	CAM_ADJ_MAX_IVVT_EX	C_CAM_INI_IN	C_CAM_INI_EX
CAM_SP_HOM_AFS_IVVT_IN	CAM_SP_HOM_AFS_IVVT_EX		
CAM_SP_HOM_AFL_IVVT_IN	CAM_SP_HOM_AFL_IVVT_EX		
CAM_SP_S_IVVT_IN	CAM_SP_S_IVVT_EX		
STATE_CTL_IVVT_IN_i	STATE_CTL_IVVT_EX_i		
CAM_SP_IVVT_IN	CAM_SP_IVVT_EX		

FUNCTION DESCRIPTION:

General information:

The matching setpoint is assigned into a common variable for the preparation of the setpoint for controlling. The basic setpoint value is worked up, if necessary, due to IVVT controlling reasons.

Description:

One of the possible setpoints (as described in "IVVT Setpoint Premanager") is chosen in dependence on the STATE_CAM_SP_IVVT_IN(_EX).

The application has the possibility to add the offset CAM_SP_CLC_IVVT_IN(_EX) to the setpoint. This offset is linked with the active adjustment direction, i.e. a positive value moves the setpoint more away from the reference position.

The calculated camshaft position setpoint after the transition from the inactive into the active IVVT state need not to be the reference position. The IVVT system would start the adjustment with its maximal velocity. This could lead to the engine speed disturbances. In order to avoid this the setpoint changes with C_CAM_SP_LGRD_STEP_IVVT steps only. This setpoint with limited gradient is used only during the time C_T_CAM_SP_LGRD_IVVT after the transition into the active state. After that, step changes of the setpoint are not limited any more.

The current calculated setpoint CAM_SP_CLC_IVVT_IN(_EX) is handed over for the camshaft position control only if it differs from the previous setpoint CAM_SP_IVVT_IN(_EX). The difference must be bigger than C_CAM_SP_DIF_MIN_IVVT.

Unsuitable calibration data may cause that the camshaft position setpoint is out of the physically possible range from C_CAM_INI_IN(_EX) to CAM_ADJ_MAX_IVVT_IN(_EX). Therefore, the setpoint is limited to this interval.

If the calculated setpoint is close to the reference position, the reference position is taken for the setpoint.

The information about the point in time of the camshaft position setpoint calculation is needed in other IVVT functionalities. Tasks with a faster recurrency need to know whether the setpoint is old or new from the point of view of the faster task.

Application conditions:

Initialization:

At reset, at deactivation, at transition "Engine run" → "Engine stop":

CAM_SP_PREL_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

At reset, at activation:

CAM_SP_LGRD_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

At activation:

T_CAM_SP_LGRD_IVVT_IN(_EX) = T_DLY_MES_IVVT

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

STATE_CAM_SP_IVVT_IN(_EX) <> "SP_NOT"

Deactivation:

Not activation

Formula section:

Choosing setpoint:

if STATE_CAM_SP_IVVT_IN(_EX) = "SP_MAN"

then *Manual setpoint*

CAM_SP_CLC_IVVT_IN(_EX) = C_CAM_SP_MAN_IVVT_IN(_EX)

elseif STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFS"

then *Setpoint for homogeneous stoichiometric*

CAM_SP_CLC_IVVT_IN(_EX) = CAM_SP_HOM_AFS_IVVT_IN(_EX)

elseif STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFL"

then *Setpoint for homogeneous lean*

CAM_SP_CLC_IVVT_IN(_EX) = CAM_SP_HOM_AFL_IVVT_IN(_EX)

elseif STATE_CAM_SP_IVVT_IN(_EX) = "SP_S"

then *Setpoint for stratified*

CAM_SP_CLC_IVVT_IN(_EX) = CAM_SP_S_IVVT_IN(_EX)

end

Additive contribution by application:

Inlet:

CAM_SP_CLC_IVVT_IN =

CAM_SP_CLC_IVVT_IN - C_CAM_SP_AS_IVVT_IN

Exhaust:

CAM_SP_CLC_IVVT_EX =

CAM_SP_CLC_IVVT_EX + C_CAM_SP_AS_IVVT_EX

Setpoint gradient limitation after transition from inactive into active state:

if T_DLY_MES_IVVT - T_CAM_SP_LGRD_IVVT_IN(_EX) <=

C_T_CAM_SP_LGRD_IVVT

and Inlet:

CAM_SP_LGRD_IVVT_IN > CAM_SP_CLC_IVVT_IN

Exhaust:

CAM_SP_LGRD_IVVT_EX < CAM_SP_CLC_IVVT_EX

then

Inlet:

CAM_SP_LGRD_IVVT_IN(n) = CAM_SP_LGRD_IVVT_IN(n-1) -

C_CAM_SP_LGRD_STEP_IVVT

Exhaust:

CAM_SP_LGRD_IVVT_EX(n) = CAM_SP_LGRD_IVVT_EX(n-1) +

C_CAM_SP_LGRD_STEP_IVVT

Inlet & Exhaust:

CAM_SP_CLC_IVVT_IN(_EX) = CAM_SP_LGRD_IVVT_IN(_EX)

endif

Minimum setpoint change:

if |CAM_SP_IVVT_IN(_EX)(n-1) - CAM_SP_CLC_IVVT_IN(_EX)| >
C_CAM_SP_DIF_MIN_IVVT

 # NC_NR_CBK_IVVT = 1:

or STATE_CTL_IVVT_IN(_EX)_1 = "ADJ"

 # NC_NR_CBK_IVVT = 2:

or STATE_CTL_IVVT_IN(_EX)_1 = "ADJ"

or STATE_CTL_IVVT_IN(_EX)_2 = "ADJ"

then

CAM_SP_PREL_IVVT_IN(_EX)(n) = CAM_SP_CLC_IVVT_IN(_EX)

else

CAM_SP_PREL_IVVT_IN(_EX)(n) = CAM_SP_IVVT_IN(_EX)(n-1)

endif

Setpoint range limitation:

Inlet:

CAM_ADJ_MAX_IVVT_IN <= CAM_SP_PREL_IVVT_IN <= C_CAM_INI_IN

Exhaust:

C_CAM_INI_EX <= CAM_SP_PREL_IVVT_EX <= CAM_ADJ_MAX_IVVT_EX

Setpoint near to the reference position:

if |CAM_SP_PREL_IVVT_IN(_EX) - C_CAM_INI_IN(_EX)| <
C_CAM_SP_HYS_NEUT_IVVT_IN(_EX)

then

CAM_SP_PREL_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

endif

Moment of calculation:

CTR_CAM_SP_IVVT_IN(_EX)(n) = CTR_CAM_SP_IVVT_IN(_EX)(n-1) + 1

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_SP_MAN_IVVT_IN	1	0...FFH	60...155.625	0.375	°CRK
Manual setpoint; inlet					
C_CAM_SP_MAN_IVVT_EX	1	0...FFH	-40.125...-135.75	0.375	°CRK
Manual setpoint; exhaust					
C_CAM_SP_AS_IVVT_IN	1	FF60H...A0H	-60...60	0.375	°CRK
Aditive contribution to setpoint calculated by normal setpoint algorithm; inlet					
C_CAM_SP_AS_IVVT_EX	1	FF60H...A0H	-60...60	0.375	°CRK
Aditive contribution to setpoint calculated by normal setpoint algorithm; exhaust					
C_T_CAM_SP_LGRD_IVVT	1	0...FFH	0...2550	10	ms
Time limit for camshaft setpoint with limited gradient					
C_CAM_SP_DIF_MIN_IVVT	1	0...A0H	0...60	0.375	°CRK
Camshaft position setpoint hysteresis					
C_CAM_SP_HYS_NEUT_IVVT_IN	1	0...A0H	0...60	0.375	°CRK
Neutral area close to reference position, if calculated setpoint lies there, reference position is taken for setpoint; inlet					
C_CAM_SP_HYS_NEUT_IVVT_EX	1	0...A0H	0...60	0.375	°CRK
Neutral area close to reference position, if calculated setpoint lies there, reference position is taken for setpoint; exhaust					
C_CAM_SP_LGRD_STEP_IVVT	1	0...A0H	0...60	0.375	°CRK
Camshaft position setpoint step for gradient limitation after IVVT activation					

9.51 IVVT Inlet Adjustment Prioritization to Exhaust One

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_IVVT_IN	O/V	0...FFH	60...155.625	0.375	°CRK
Control camshaft position setpoint; inlet					
CAM_SP_IVVT_EX	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Control camshaft position setpoint; exhaust					
CAM_MV_MMV_PRI_IVVT	V	0...FFFFH	60...155.9985	1.465e-3	°CRK
Filtered inlet mean camshaft position for inlet to exhaust prioritization					
LV_ACT_PRI_IVVT	V	0...1H	0...1	1	-
1 = inlet prioritization to exhaust active, 0 = not active					

Input data:

STATE_CAM_SP_IVVT_IN	STATE_CAM_SP_IVVT_EX	C_CAM_INI_IN	C_CAM_INI_EX
CAM_SP_PREL_IVVT_IN	CAM_SP_PREL_IVVT_EX	NLC_IVVT_IN	NLC_IVVT_EX
CAM_MV_IN	CAM_MV_EX	N_32	TOIL
CAM_ADJ_MAX_IVVT_EX			

General information:

If the engine speed is low and the oil temperature high there can be insufficient oil supply. If both inlet and exhaust camshaft should move simultaneously there can be slow adjustments of both camshafts. Therefore, the exhaust adjustment is stopped in such case, i.e. the inlet is prioritized to the exhaust.

Description:

If the deviation between the exhaust camshaft position (CAM_MV_EX due to 2-bank systems) and setpoint is bigger than ID_CAM_DIF_MIN_PRI_IVVT an exhaust adjustment is to realize. The inlet side is to check. If no inlet adjustment or only a small one is to realize, i.e. the deviation between the position and setpoint is smaller than ID_CAM_DIF_MIN_PRI_IVVT or the inlet side moves only slowly or does not move, the exhaust is phased in the regular way, otherwise it is held until the above inlet condition is not satisfied.

Application conditions:

Initialization:

At reset, at transition "Engine run" → "Engine stop":

CAM_SP_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

 # NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

CAM_MV_MMV_PRI_IVVT = C_CAM_INI_IN

LV_ACT_PRI_IVVT = 0

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

At every engine state

Deactivation:

Not activation

Formula section:

 # NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:

CAM_SP_IVVT_IN = CAM_SP_PREL_IVVT_IN

 # NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

CAM_SP_IVVT_EX = CAM_SP_PREL_IVVT_EX

 # NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

CAM_SP_IVVT_IN = CAM_SP_PREL_IVVT_IN

if STATE_CAM_SP_IVVT_IN <> "SP_NOT"

and STATE_CAM_SP_IVVT_EX <> "SP_NOT"

then *There is simultaneous inlet and exhaust phasing*

CAM_MV_MMV_PRI_IVVT(n) =

CAM_MV_MMV_PRI_IVVT(n-1) + C_CRLC_CAM_PRI_IVVT *

(CAM_MV_IN - CAM_MV_MMV_PRI_IVVT(n-1))

if abs(CAM_MV_EX - CAM_SP_PREL_IVVT_EX) <

ID_CAM_DIF_MIN_PRI_IVVT

then *Only small exhaust deviation to manage, no prioritization necessary*

CAM_SP_IVVT_EX = CAM_SP_PREL_IVVT_EX

LV_ACT_PRI_IVVT = 0

else

if abs(CAM_MV_IN - CAM_SP_PREL_IVVT_IN) <

ID_CAM_DIF_MIN_PRI_IVVT

or abs(CAM_MV_MMV_PRI_IVVT(n) -

CAM_MV_MMV_PRI_IVVT(n-1)) <

C_CAM_DIF_STAT_IN_PRI_IVVT

then *Only small inlet deviation to manage or inlet position does not change anymore, normal exhaust adjustment*

CAM_SP_IVVT_EX = CAM_SP_PREL_IVVT_EX

LV_ACT_PRI_IVVT = 0

else *Inlet is prioritized to exhaust, i.e. exhaust is held while inlet moving*

if LV_ACT_PRI_IVVT = 0

then

CAM_SP_IVVT_EX = CAM_MV_EX

Limitation: C_CAM_INI_EX <=

CAM_SP_IVVT_EX <=

CAM_ADJ_MAX_IVVT_EX

LV_ACT_PRI_IVVT = 1

endif

endif

endif

else *There is no simultaneous inlet and exhaust phasing*

CAM_SP_IVVT_EX = CAM_SP_PREL_IVVT_EX

CAM_MV_MMV_PRI_IVVT = CAM_MV_IN

LV_ACT_PRI_IVVT = 0

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CRLC_CAM_PRI_IVVT	1	0...FFH	0...0.9984	0.0039	-
Correlation constant for filtered camshaft position for inlet to exhaust prioritization					
ID_CAM_DIF_MIN_PRI_IVVT	6x4	0...FFH	0...95.625	0.375	°CRK
LDPM_N_32_2_VVTI	6	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Minimum difference between camshaft position and setpoint for recognizing upcoming adjustments at inlet to exhaust prioritization					
C_CAM_DIF_STAT_IN_PRI_IVVT	1	0...FFH	0...95.625	0.375	°CRK
Smaller inlet camshaft position change between two recurrences keeps standard exhaust adjustment					

9.52 IVVT Setpoint for Homogeneous Stoichiometric

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_HOM_AFS_IVVT_IN	O/V	0...FFH	60...155.625	0.375	°CRK
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_HOM_AFS_IVVT_EX	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_HOM_AFS_ADD_IVVT_IN	V	FF60...A0H	-60...60	0.375	°CRK
Additive contribution to normal setpoint calculation for homogeneous stoichiometric; inlet					
CAM_SP_HOM_AFS_ADD_IVVT_EX	V	FF60...A0H	-60...60	0.375	°CRK
Additive contribution to normal setpoint calculation for homogeneous stoichiometric; exhaust					
MAF_IVVT	V	0...FFFFH	0...1389	0.02119	mg/stk
Additive contribution to normal setpoint calculation for homogeneous stoichiometric; inlet					
CAM_SP_HOM_AFS_IVVT_TMP_IN	-	0...FFH	60...155.625	0.375	°CRK
Temporary Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_HOM_AFS_IVVT_TMP_EX	-	0...FFH	-40.125...-135.75	0.375	°CRK
Temporary Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_ADD_TMP_IVVT_IN	V	FF60...A0H	-60...60	0.375	°CRK
Additional Value for Inlet due to slow-respons in catalyst heating mode					
CAM_SP_ADD_TMP_IVVT_EX	V	FF60...A0H	-60...60	0.375	°CRK
Additional Value for Exhause due to slow-respons in catalyst heating mode					
MAF_SP_TQI_DYN	V	0...FFFFH	0...1389	0.02119	mg/stk
Scaled air mass flow between MAF and MAF_SP_TQI during turbo lag					
MAF_DIF_IVVT	V	0...FFFFH	0...1389	0.02119	mg/stk
Difference of MAF and MAF_SP_TQI					
FAC_MAF_DYN_IVVT	V	0...FFH	0...0.996	0.0039	-
Weighting factor to scale MAF_IVVT between MAF and MAF_SP_TQI during turbo lag					
CAM_SP_HOM_AFS_IVVT_TMP_2_IN	V	0...FFH	60...155.625	0.375	°CRK
Temporary Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_HOM_AFS_IVVT_TMP_2_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
Temporary Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode					
CAM_SP_DIF_CAM_OFS_AD_IN	V	FF60...A0H	-60...60	0.375	°CRK
Difference between CAM_SP from CAM_OFS_AD and normal CAM_SP after CAM_OFS_AD has finished					
CAM_SP_DIF_CAM_OFS_AD_EX	V	FF60...A0H	-60...60	0.375	°CRK
Difference between CAM_SP from CAM_OFS_AD and normal CAM_SP after CAM_OFS_AD has finished					
CAM_SP_HOM_AFS_IVVT_AMP_H_IN	V	0...FFH	60...155.625	0.375	°CRK
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode for high AMP					
CAM_SP_HOM_AFS_IVVT_AMP_H_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode for high AMP					
CAM_SP_HOM_AFS_IVVT_AMP_L_IN	V	0...FFH	60...155.625	0.375	°CRK
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode for low AMP					
CAM_SP_HOM_AFS_IVVT_AMP_L_EX	V	0...FFH	-40.125...-135.75	0.375	°CRK
exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode for low AMP					
FAC_CAM_SP_PL_AMP_IVVT_IN	V	80...7FH	-4...3.97	0.031	-
Ambient pressure weighting factor for inlet camshaft position setpoint, part load					
FAC_CAM_SP_PL_AMP_IVVT_EX	V	80...7FH	-4...3.97	0.031	-
Ambient pressure weighting factor for exhaust camshaft position setpoint, part load					

Input data:

STATE_CAM_SP_IVVT_IN	STATE_CAM_SP_IVVT_EX	C_CAM_INI_IN	C_CAM_INI_EX
FAC_CAM_SP_IS_IVVT_IN	FAC_CAM_SP_IS_IVVT_EX	LV_FL	T_AST
FAC_CAM_SP_PL_IVVT_IN	FAC_CAM_SP_PL_IVVT_EX	LV_TI_CH	TCO_ST
MAF_SP_TQI	N	LV_IS	MAF_MAX_COR

LC_TCHA_CONF	LV_PU	LV_PUC	LC_MAF_SP_MAN
C_MAF_SP_MAN	CAM_SP_CAM_OFS_AD_IVVT_IN[NC_NR_CBK_IVVT]	CAM_SP_CAM_OFS_AD_IVVT_EX[NC_NR_CBK_IVVT]	LV_CAM_SP_ADJ_CAM_OFS_AD
NC_NR_CBK_IVVT			

FUNCTION DESCRIPTION:

Description:

The measured MAF into the engine is not suitable as load parameter for the determination of the setpoint. The camshaft position has influence on the MAF, also, there can be feedback effects. Therefore, a feedback-free air mass flow is needed instead of the measured one. This is fulfilled by MAF_IVVT, i.e. the current air mass flow setpoint for torque interventions.

The camshaft position setpoints for the homogeneous stoichiometric combustion mode are calculated from maps based on the N and MAF_IVVT. There are maps for two coolant temperature levels. The coolant temperature effects are considered by weighting factors.

In order to speed up the catalyst heating, special camshaft positons can be used. This module provides the setpoint manipulation solely by additive contibutions.

During PU and PUC, both intake and exhaust setpoints are forced to full retard position to ensure minimum values overlap.

Application conditions:

Initialization: At reset: CAM_SP_HOM_AFS_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

CAM_SP_HOM_AFS_ADD_IVVT_IN(_EX) = 0 °CRK

CAM_SP_ADD_TMP_IVVT_IN(_EX) = 0°CRK

Recurrence: 360 °CRK linked task, see "IVVT Scheduler"

Activation: STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFS"

Deactivation: Not activation

Formula section:

MAF_IVVT calculation

MAF_DIF_IVVT = max (0; MAF_SP_TQI - MAF)

FAC_MAF_DYN_IVVT = IP_FAC_MAF_DYN_IVVT

MAF_SP_TQI_DYN =

FAC_MAF_DYN_IVVT * MAF_SP_TQI + (1 - FAC_MAF_DYN_IVVT) * MAF

FAC_CAM_SP_PL_AMP_IVVT_IN/EX = IP_FAC_CAM_SP_PL_AMP_IVVT_IN/EX

The calculation above is done to avoid bigger MAF_IVVT deviations during turbo lag. While there is a big difference between MAF and MAF_SP due to the turbo lag, MAF_IVVT can be set to MAF and back to MAF_SP after the turbo lag is over.

If LC_MAF_SP_SENS = 1

then MAF_IVVT = MAF (from sensor)

```
else    if      LC_MAF_SP_MAN = 1
        then    MAF_IVVT = C_MAF_SP_MAN
        else    if      LC_TCHA_CONF = 1
                then    MAF_IVVT = Min ( MAF_MAX_COR, MAF_SP_TQI_DYN )
                else    MAF_IVVT = Min ( MAF_MAX_COR, MAF_SP_TQI )
                endif
        endif
endif
```

Part load - full load distinguishing:

```
If      LV_PU = 1 or LV_PUC = 1
then    CAM_SP_HOM_AFS_ADD_IVVT_IN = 60 °CRK
        CAM_SP_HOM_AFS_ADD_IVVT_EX = -60 °CRK
    else
        Setpoint due to catalyst heating
endif
```

Setpoint due to catalyst heating:

```
if      LV_TI_CH = 0
    or  T_AST <= C_T_AST_IVVT
then    Catalyst heating inactive or applied delay after start not yet elapsed
        if      LV_IS = 1
        then
                if      CAM_SP_HOM_AFS_ADD_IVVT_IN < 0°CRK
                then    CAM_SP_HOM_AFS_ADD_IVVT_IN (N) =
                        CAM_SP_HOM_AFS_ADD_IVVT_IN (N-1)
                        + C_CAM_SP_HOM_AFS_LGRD_IVVT_IN
                endif
                if      CAM_SP_HOM_AFS_ADD_IVVT_EX > 0°CRK
                then    CAM_SP_HOM_AFS_ADD_IVVT_EX (N) =
                        CAM_SP_HOM_AFS_ADD_IVVT_EX ( N-1)
                        - C_CAM_SP_HOM_AFS_LGRD_IVVT_EX
                endif
                CAM_SP_ADD_TMP_IVVT_IN(_EX) = 0°CRK
        else
                CAM_SP_HOM_AFS_ADD_IVVT_IN(_EX) = 0 °CRK
                CAM_SP_ADD_TMP_IVVT_IN(_EX) = 0°CRK
```

```
        endif
else    if      LV_IS = 1
        then    Idle speed
                if      CAM_SP_ADD_TMP_IVVT_IN >
                        IP_CAM_SP_ADD_CH_IS_IVVT_IN
                then    CAM_SP_ADD_TMP_IVVT_IN (N) =
                        CAM_SP_ADD_TMP_IVVT_IN(n-1)
                        - IP_CAM_SP_HOM_AFS_LGRD_IVVT_IN
                endif
                If      CAM_SP_ADD_TMP_IVVT_EX <
                        IP_CAM_SP_ADD_CH_IS_IVVT_EX
                then    CAM_SP_ADD_TMP_IVVT_EX(n) =
                        CAM_SP_ADD_TMP_IVVT_EX(n-1)
                        + IP_CAM_SP_HOM_AFS_LGRD_IVVT_EX
                endif
                CAM_SP_HOM_AFS_ADD_IVVT_IN =
                MAX(CAM_SP_ADD_TMP_IVVT_IN,
                IP_CAM_SP_ADD_CH_IS_IVVT_IN)
                CAM_SP_HOM_AFS_ADD_IVVT_EX =
                MIN(CAM_SP_ADD_TMP_IVVT_EX,
                IP_CAM_SP_ADD_CH_IS_IVVT_EX)

        else
                if      LV_FL = 0
                then    Part load
                        CAM_SP_HOM_AFS_ADD_IVVT_IN(_EX) =
                        IP_CAM_SP_ADD_CH_PL_IVVT_IN(_EX) *
                        IP_FAC_CAM_SP_ADD_CH_PL_IVVT
                else    Full load
                        CAM_SP_HOM_AFS_ADD_IVVT_IN(_EX) = 0
                endif
        endif
endif
```

Normal setpoint algorithm:

```
if      LV_IS = 1 or LV_MSR_ACT
then    Idle speed or MSR active
```

CAM_SP_HOM_AFS_IVVT_TMP_IN(_EX) =
(IP_CAM_SP_TCO_L_IS_IVVT_IN(_EX)
- IP_CAM_SP_TCO_H_IS_IVVT_IN(_EX))
* FAC_CAM_SP_IS_IVVT_IN(_EX)
+ IP_CAM_SP_TCO_H_IS_IVVT_IN(_EX)

else

if LV_FL = 0

then *Part load*

CAM_SP_HOM_AFS_IVVT_AMP_H_IN(_EX) =
(IP_CAM_SP_TCO_L_AMP_H_IVVT_IN(_EX)
- IP_CAM_SP_TCO_H_AMP_H_IVVT_IN(_EX))
* FAC_CAM_SP_PL_IVVT_IN(_EX)
+ IP_CAM_SP_TCO_H_AMP_H_IVVT_IN(_EX)

CAM_SP_HOM_AFS_IVVT_AMP_L_IN(_EX) =
(IP_CAM_SP_TCO_L_AMP_L_IVVT_IN(_EX)
- IP_CAM_SP_TCO_H_AMP_L_IVVT_IN(_EX))
* FAC_CAM_SP_PL_IVVT_IN(_EX)
+ IP_CAM_SP_TCO_H_AMP_L_IVVT_IN(_EX)

CAM_SP_HOM_AFS_IVVT_TMP_IN(_EX) =
CAM_SP_HOM_AFS_IVVT_AMP_H_IN(_EX)
- (CAM_SP_HOM_AFS_IVVT_AMP_H_IN(_EX)
-CAM_SP_HOM_AFS_IVVT_AMP_L_IN(_EX))
* FAC_CAM_SP_PL_AMP_IVVT_IN(_EX)

else *Full load*

CAM_SP_HOM_AFS_IVVT_TMP_IN(_EX) =
(IP_CAM_SP_TCO_L_FL_IVVT_IN(_EX)
- IP_CAM_SP_TCO_H_FL_IVVT_IN(_EX))
* FAC_CAM_SP_PL_IVVT_IN(_EX)
+ IP_CAM_SP_TCO_H_FL_IVVT_IN(_EX)

endif

endif

CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX) =

CAM_SP_HOM_AFS_IVVT_TMP_IN(_EX) + CAM_SP_HOM_AFS_ADD_IVVT_IN(_EX)

Under certain conditions (LV_CAM_SP_ADJ_OFS_AD = 1), their can be a request for an adjustment of the inlet and exhaust camshaft by the Camshaft Offset Adaptation (CAM_OFS_AD) functionality. In such a case, the setpoints which are normally calculated within the IVVT functionalities are not used but those delivered by CAM_OFS_AD.

If LV_CAM_SP_ADJ_CAM_OFS_AD = 1

Then CAM_SP_HOM_AFS_IVVT_IN(_EX) = CAM_SP_CAM_OFS_AD_IVVT_IN(_EX)

For a two cylinder bank engine (NC_NR_CBK_IVVT=2), CAM_SP_CAM_OFS_AD_IVVT_IN(_EX)[NC_NR_CBK_IVVT] is the mean value over the two IVVT banks.

CAM_SP_DIF_CAM_OFS_AD_IN(_EX) =

CAM_SP_CAM_OFS_AD_IVVT_IN(_EX) -

CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX)

Else

(When the Camshaft-offset adaptation was active, CAM_SP is ramped from the adaptation-CAM_SP to the normal IVVT-CAM_SP. If the normal IVVT-CAM_SP makes a step above C_THD_CAM_SP_CHG no ramping is done.)

If [|(CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX) –

CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX)$_{N-2}$)| < C_THD_CAM_SP_CHG

And |CAM_SP_DIF_CAM_OFS_AD_IN(_EX)| > C_CAM_SP_STEP_MAX]

Then **If** CAM_SP_DIF_CAM_OFS_AD_IN(_EX) > 0

Then CAM_SP_DIF_CAM_OFS_AD_IN(_EX) =

CAM_SP_DIF_CAM_OFS_AD_IN(_EX) - C_CAM_SP_STEP_MAX

Else CAM_SP_DIF_CAM_OFS_AD_IN(_EX) =

CAM_SP_DIF_CAM_OFS_AD_IN(_EX) + C_CAM_SP_STEP_MAX

Endif

CAM_SP_HOM_AFS_IVVT_IN(_EX) =

CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX) +
CAM_SP_DIF_CAM_OFS_AD_IN(_EX)

Else *(normal IVVT-CAM_SP is used, no Camshaft offset adaptation was requested)*

CAM_SP_HOM_AFS_IVVT_IN(_EX) =

CAM_SP_HOM_AFS_IVVT_TMP_2_IN(_EX)

CAM_SP_DIF_CAM_OFS_AD_IN(_EX) = 0

Endif

Endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_CAM_SP_TCO_L_IS_IVVT_IN	6x6	0...FFH	60...155.625	0.375	°CRK
LDPM_N_6_VVTI	6	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_4_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, idle speed					
IP_CAM_SP_TCO_L_IS_IVVT_EX	6x6	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_6_VVTI	6	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_4_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, idle speed					
IP_CAM_SP_TCO_H_IS_IVVT_IN	6x6	0...FFH	60...155.625	0.375	°CRK
LDPM_N_6_VVTI	6	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_4_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, idle speed					
IP_CAM_SP_TCO_H_IS_IVVT_EX	6x6	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_6_VVTI	6	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_4_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, idle speed					
IP_CAM_SP_TCO_L_AMP_H_IVVT_IN	16x12	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, AMP level H, part load					
IP_CAM_SP_TCO_L_AMP_H_IVVT_EX	16x12	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, AMP level H, part load					
IP_CAM_SP_TCO_H_AMP_H_IVVT_IN	16x12	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, AMP level H part load					
IP_CAM_SP_TCO_H_AMP_H_IVVT_EX	16x12	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, AMP level H part load					
IP_CAM_SP_TCO_L_FL_IVVT_IN	16 x6	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_2_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, full load					
IP_CAM_SP_TCO_L_FL_IVVT_EX	16 x6	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_2_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, full load					
IP_CAM_SP_TCO_H_FL_IVVT_IN	16 x6	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_2_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, full load					
IP_CAM_SP_TCO_H_FL_IVVT_EX	16 x6	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LDPM_MAF_IVVT_2_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, full load					
C_T_AST_IVVT	1	1...FFFFH	0.1...6553.5	0.1	s
Time condition for application of the additive contribution to normal setpoint for homogeneous stoichiometric					
IP_CAM_SP_ADD_CH_IS_IVVT_IN	1x8	0...140H	-60...60	0.375	°CRK
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Additive contributions to normal setpoint for homogeneous stoichiometric, idle speed; inlet/					
IP_CAM_SP_ADD_CH_IS_IVVT_EX	1x8	0...140H	-60...60	0.375	°CRK
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Additive contributions to normal setpoint for homogeneous stoichiometric, idle speed; exhaust					
IP_CAM_SP_ADD_CH_PL_IVVT_IN	1x8	0...140H	-60...60	0.375	°CRK
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Additive contributions to normal setpoint for homogeneous stoichiometric, part load; inlet					
IP_CAM_SP_ADD_CH_PL_IVVT_EX	1x8	0...140H	-60...60	0.375	°CRK
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Additive contributions to normal setpoint for homogeneous stoichiometric, part load; exhaust					
IP_FAC_CAM_SP_ADD_CH_PL_IVVT	6x6	0...FFH	0...0.996	0.0039	-
LDPM_N_6_VVTI	6	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_4_VVTI	6	0...FFFFH	0...1389	0.021	mg/stk
Weighting factors for additive contributions to normal setpoint for homogeneous stoichiometric, part load					
LC_MAF_SP_SENS	1	0…1H	0…1	1	-
Manual switch between C_MAF_SP_MAN and MAF					
IP_CAM_SP_HOM_AFS_LGRD_IVVT_IN	1x8	0...140H	-60...60	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
Gradual additive contributions to catalyst heating setpoint for homogeneous stoichiometric; inlet					
IP_CAM_SP_HOM_AFS_LGRD_IVVT_EX	1x8	0...140H	-60...60	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
Gradual additive contributions to catalyst heating setpoint for homogeneous stoichiometric; exhaust					
IP_FAC_MAF_DYN_IVVT	1x8	0...FFH	0...0.996	0.0039	-
LDP_MAF_DIF_IVVT	8	0...FFFFH	0...1389	0.02119	mg/stk
Weighting factor to scale MAF_IVVT between MAF and MAF_SP_TQI during turbo lag					
C_CAM_SP_STEP_MAX	1	0...FFH	0...95.625	0.375	°CRK
Maximum CAM_SP ramping step after Camshaft offset adaptation has finished					
C_THD_CAM_SP_CHG	1	0...FFH	0...95.625	0.375	°CRK
Threshold for ramping CAM_SP after Camshaft offset adaptation has finished					
C_CAM_SP_HOM_AFS_LGRD_IVVT_IN	1	0...A0H	0...60	0.375	°CRK
Gradual additive contributions to normal setpoint for homogeneous stoichiometric, idle speed; inlet					
C_CAM_SP_HOM_AFS_LGRD_IVVT_EX	1	0...A0H	0...60	0.375	°CRK
Gradual additive contributions to normal setpoint for homogeneous stoichiometric, idle speed; exhaust					
IP_CAM_SP_TCO_L_AMP_L_IVVT_IN	16x12	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, AMP level low, part load					
IP_CAM_SP_TCO_L_AMP_L_IVVT_EX	16x12	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level low, AMP level low, part load					
IP_CAM_SP_TCO_H_AMP_L_IVVT_IN	16x12	0...FFH	60...155.625	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk
Inlet camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, AMP level low part load					
IP_CAM_SP_TCO_H_AMP_L_IVVT_EX	16x12	0...FFH	-40.125...-135.75	0.375	°CRK
LDPM_N_7_VVTI	16	0...1FE0H	0...8160	1	rpm
LDPM_MAF_IVVT_3_VVTI	12	0...FFFFH	0...1389	0.021	mg/stk

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
Exhaust camshaft position setpoint for homogeneous stoichiometric combustion mode, coolant temperature level high, AMP level low part load					
IP_FAC_CAM_SP_PL_AMP_IVVT_IN	8x8	0...FFH	-4...3.97	0.031	-
LDPM_AMP_1_VVTI	8	0...FFFFH	0…5434	0.083	hPa
LDPM_MAF_IVVT_5_VVTI	8	0...FFFFH	0...1389	0.021	mg/stk
Ambient pressure weighting factors for inlet camshaft position setpoint, part load					
IP_FAC_CAM_SP_PL_AMP_IVVT_EX	8x8	0...FFH	-4...3.97	0.031	-
LDPM_AMP_1_VVTI	8	0...FFFFH	0…5434	0.083	hPa
LDPM_MAF_IVVT_5_VVTI	8	0...FFFFH	0...1389	0.021	mg/stk
Ambient pressure weighting factors for exhaust camshaft position setpoint, part load					

9.53 IVVT Factors for AFS Setpoint

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
FAC_CAM_SP_IS_IVVT_IN	V	80...7FH	-4...3.97	0.031	-
Coolant temperature weighting factor for inlet camshaft position setpoint, idle speed					
FAC_CAM_SP_IS_IVVT_EX	V	80...7FH	-4...3.97	0.031	-
Coolant temperature weighting factor for exhaust camshaft position setpoint, idle speed					
FAC_CAM_SP_PL_IVVT_IN	V	80...7FH	-4...3.97	0.031	-
Coolant temperature weighting factor for inlet camshaft position setpoint, part load					
FAC_CAM_SP_PL_IVVT_EX	V	80...7FH	-4...3.97	0.031	-
Coolant temperature weighting factor for exhaust camshaft position setpoint, part load					

Input data:

TCO	TCO_ST		

FUNCTION DESCRIPTION:

Description:

The coolant temperature influence on the camshaft position setpoints is taken into account by the weighting factors FAC_CAM_SP_IS_IVVT_IN/EX and FAC_CAM_SP_PL_IVVT_ IN/EX. The characteristics for the part load are used for the full load.

Application conditions:

Initialization:

Recurrence: 1 s

Activation: *At every engine state*

Formula section:

FAC_CAM_SP_IS_IVVT_IN/EX = IP_FAC_CAM_SP_IS_IVVT_IN/EX

FAC_CAM_SP_PL_IVVT_IN/EX = IP_FAC_CAM_SP_PL_IVVT_IN/EXCalibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_FAC_CAM_SP_IS_IVVT_IN	8x8	0...FFH	-4...3.97	0.031	-
LDPM_TCO_2_VVTI	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Coolant temperature weighting factors for inlet camshaft position setpoint, idle speed					
IP_FAC_CAM_SP_IS_IVVT_EX	8x8	0...FFH	-4...3.97	0.031	-
LDPM_TCO_2_VVTI	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Coolant temperature weighting factors for exhaust camshaft position setpoint, idle speed					
IP_FAC_CAM_SP_PL_IVVT_IN	8x8	0...FFH	-4...3.97	0.031	-
LDPM_TCO_2_VVTI	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Coolant temperature weighting factors for inlet camshaft position setpoint, part load					
IP_FAC_CAM_SP_PL_IVVT_EX	8x8	0...FFH	-4...3.97	0.031	-
LDPM_TCO_2_VVTI	8	0...FEH	-48...142.5	0.75	°C
LDPM_TCO_ST_1_VVTI	8	0...FEH	-48...142.5	0.75	°C
Coolant temperature weighting factors for exhaust camshaft position setpoint, part load					

9.54 IVVT Setpoint for Homogeneous Lean

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_HOM_AFL_IVVT_IN	O/V	0...FFH	60...155.625	0.375	°CRK
Inlet camshaft position setpoint for homogeneos lean combustion mode					
CAM_SP_HOM_AFL_IVVT_EX	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Exhaust camshaft position setpoint for homogeneos lean combustion mode					

Input data:

STATE_CAM_SP_IVVT_IN	STATE_CAM_SP_IVVT_EX	C_CAM_INI_IN	C_CAM_INI_EX

FUNCTION DESCRIPTION:

Description:

At the moment the camshaft position setpoint for homogeneos lean combustion mode is always the reference position.

Application conditions:

Initialization:

At reset:

CAM_SP_HOM_AFL_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

STATE_CAM_SP_IVVT_IN(_EX) = "SP_HOM_AFL"

Formula section:

CAM_SP_HOM_AFL_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

9.55 IVVT Setpoint for Stratified

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_S_IVVT_IN	O/V	0...FFH	60...155.625	0.375	°CRK
Inlet camshaft position setpoint for stratified combustion mode					
CAM_SP_S_IVVT_EX	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Exhaust camshaft position setpoint for stratified combustion mode					

Input data:

C_CAM_INI_IN	C_CAM_INI_EX		
STATE_CAM_SP_IVVT_IN	STATE_CAM_SP_IVVT_EX		

FUNCTION DESCRIPTION:

General information:

This is the stub module for the camshaft position setpoint for stratified combustion mode.

Description:

The default camshaft position setpoint is the reference position C_CAM_INI_IN(_EX).

Application conditions:

Initialization:

At reset:

CAM_SP_S_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

STATE_CAM_SP_IVVT_IN(_EX) = "SP_S"

Formula section:

CAM_SP_S_IVVT_IN(_EX) = C_CAM_INI_IN(_EX)

9.56 IVVT Individual Setpoint Selection

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_IND_DIAG_IVVT_IN_i	O/V	0...FFH	60...155.625	0.375	°CRK
Diagnostic camshaft position setpoint for individual camshaft; inlet					
CAM_SP_IND_DIAG_IVVT_EX_i	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Diagnostic camshaft position setpoint for individual camshaft; exhaust					

Input data:

LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
CAM_SP_TRIP_IVVT_IN_i	CAM_SP_TRIP_IVVT_EX_i	CAM_SP_IVVT_IN	CAM_SP_IVVT_EX
CAM_SP_BAL_IVVT_IN_i	CAM_SP_BAL_IVVT_EX_i	NC_NR_CBK_IVVT	
LV_TRIP_ACT_IVVT_IN	LV_TRIP_ACT_IVVT_EX		

FUNCTION DESCRIPTION:

General information:

The individual camshaft setpoint is selected dependent on the IVVT system configuration and on the status of the short trip.

Description:

The setpoint CAM_SP_IND_DIAG_IVVT_IN(_EX)_i is a result of the setpoint strategy and IVVT control algorithm. It is used in the diagnosis and for the camshaft position control. It is possible to emulate a faulty actuator. Then a different setpoint is used for the camshaft position control.

Application conditions:

Initialization:

At reset, at transition "Engine run" → "Engine stop", at deactivation:

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

Recurrence:

 # NC_NR_CBK_IVVT = 1:

360 °CRK linked task, see "IVVT Scheduler"

 # NC_NR_CBK_IVVT = 2:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_ACT_IND_IVVT_IN(_EX)_i = 1

Deactivation:

Not activation

Formula section:

Individual setpoint selection:

if LV_TRIP_ACT_IVVT_IN(_EX) = 0

then

 # NC_NR_CBK_IVVT = 1:

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i = CAM_SP_IVVT_IN(_EX)

 # NC_NR_CBK_IVVT = 2:

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i = CAM_SP_BAL_IVVT_IN(_EX)_i

else

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i = CAM_SP_TRIP_IVVT_IN(_EX)_i

endif

9.57 IVVT Setpoint Work-up before Controlling

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_IND_IVVT_IN_i	O/V	0...FFH	60...155.625	0.375	°CRK
Control camshaft position setpoint for individual camshaft; inlet					
CAM_SP_IND_IVVT_EX_i	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Control camshaft position setpoint for individual camshaft; exhaust					
LV_CAM_SP_DYW_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = limited dynamics of camshaft position setpoint for ind. camshaft fulfilled, 0 = not fulfilled; inlet					
LV_CAM_SP_DYW_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = limited dynamics of camshaft position setpoint for ind. camshaft fulfilled, 0 = not fulfilled; exhaust					
CAM_SP_DYW_IVVT_IN_i	-	0...FFH	60...155.625	0.375	°CRK
Help variable for deterimination of limited dynamics of camshaft position setpoint; inlet					
CAM_SP_DYW_IVVT_EX_i	-	0...FFH	-40.125...-135.75	0.375	°CRK
Help variable for deterimination of limited dynamics of camshaft position setpoint; exhaust					

Input data:

LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
NC_NR_CBK_IVVT			
CAM_ADJ_MAX_IVVT_IN	CAM_ADJ_MAX_IVVT_EX		
C_CAM_SP_HYS_NEUT_IVVT_IN	C_CAM_SP_HYS_NEUT_IVVT_EX		
CAM_SP_IND_DIAG_IVVT_IN_i	CAM_SP_IND_DIAG_IVVT_EX_i		
CAM_SP_IND_DFCT_IVVT_IN_i	CAM_SP_IND_DFCT_IVVT_EX_i		
LV_CAM_SP_DFCT_IVVT_IN_i	LV_CAM_SP_DFCT_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

The individual setpoint value is worked up, if necessary, due to IVVT tuning or controlling reasons.

Description:

It is possible to emulate a faulty actuator. If LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1 then CAM_SP_IND_DFCT_IVVT_IN(_EX)_i is the setpoint used in the camshaft position control algorithm (CAM_SP_IND_IVVT_IN(_EX)_i). It can be for example a blocked actuator. The diagnosis works with the setpoint from the setpoint strategy and IVVT control algorithm CAM_SP_IND_DIAG_IVVT_IN(_EX)_i. If LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 0 then CAM_SP_IND_IVVT_IN(_EX)_i equals CAM_SP_IND_DIAG_IVVT_IN(_EX)_i and the IVVT system works in the normal mode.

The bank balancing algorithm or improper short trip calibration may cause that the setpoint is out of the physically possible range from C_CAM_INI_IN(_EX) to CAM_ADJ_MAX_IVVT_IN(_EX). Therefore, the setpoint is limited to this interval. If the setpoint is close to the reference position, the reference position is taken for the setpoint.

The information about the camshaft position setpoint dynamics is needed in other IVVT functionalities. If there are only small changes it is called limited dynamics.

Application conditions:

Initialization:

At reset, at transition "Engine run" → "Engine stop", at deactivation:

CAM_SP_IND_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

At transition "Engine run" → "Engine stop", at deactivation:

LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 0

At reset, at deactivation:

CAM_SP_DYW_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

Recurrence:

 # NC_NR_CBK_IVVT = 1:

360 °CRK linked task, see "IVVT Scheduler"

 # NC_NR_CBK_IVVT = 2:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_ACT_IND_IVVT_IN(_EX)_i = 1

Deactivation:

Not activation

Formula section:

Actuator failure emulation:

if LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 0

then

CAM_SP_IND_IVVT_IN(_EX)_i = CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

else

CAM_SP_IND_IVVT_IN(_EX)_i = CAM_SP_IND_DFCT_IVVT_IN(_EX)_i

endif

Setpoint range limitation:

Inlet:

CAM_ADJ_MAX_IVVT_IN <= CAM_SP_IND_IVVT_IN_i <= C_CAM_INI_IN

Exhaust:

C_CAM_INI_EX <= CAM_SP_IND_IVVT_EX_i <= CAM_ADJ_MAX_IVVT_EX

Setpoint near to the reference position:

if |CAM_SP_IND_IVVT_IN(_EX)_i - C_CAM_INI_IN(_EX)| < C_CAM_SP_HYS_NEUT_IVVT_IN(_EX)

then

CAM_SP_IND_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

endif

Setpoint limited dynamics:

if abs(CAM_SP_IND_IVVT_IN(_EX)_i - CAM_SP_DYW_IVVT_IN(_EX)_i) <= C_CAM_SP_DIF_DYW_IVVT

then

LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 1

else

LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 0

CAM_SP_DYW_IVVT_IN(_EX)_i = CAM_SP_IND_IVVT_IN(_EX)_i

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_SP_DIF_DYW_IVVT	1	0...FFH	0...95.625	0.375	°CRK
Camshaft position setpoint window for determination of limited dynamics					

9.58 IVVT Individual Camshaft Setpoint

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_MMV_BAL_IVVT_IN_i	V	0...FFFFH	60...155.9985	1.465e-3	°CRK
Filtered position of inlet camshaft for bank balancing					
CAM_MMV_BAL_IVVT_EX_i	V	0...FFFFH	-40.125... -136.1235	1.465e-3	°CRK
Filtered position of exhaust camshaft for bank balancing					
CAM_SP_BAL_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint for individual camshaft for bank balancing; inlet					
CAM_SP_BAL_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint for individual camshaft for bank balancing; exhaust					
LV_ACT_VEL_BAL_IVVT_IN_i	V	0...1H	0...1	1	-
1 = bank balancing due to unequal adjustment velocities active, 0 = inactive; inlet					
LV_ACT_VEL_BAL_IVVT_EX_i	V	0...1H	0...1	1	-
1 = bank balancing due to unequal adjustment velocities active, 0 = inactive; exhaust					
LV_ERR_BAL_IVVT_IN_i	V	0...1H	0...1	1	-
1 = individual camshaft error present at which bank balancing, 0 = not present; inlet					
LV_ERR_BAL_IVVT_EX_i	V	0...1H	0...1	1	-
1 = individual camshaft error present at which bank balancing, 0 = not present; exhaust					
VEL_CAM_BAL_IVVT_IN_i[3]	V	8000...7FFFH	-12288... 12287.625	0.375	°CRK/s
Adjustment velocity for bank balancing; inlet					
VEL_CAM_BAL_IVVT_EX_i[3]	V	8000...7FFFH	-12288... 12287.625	0.375	°CRK/s
Adjustment velocity for bank balancing; exhaust					
VEL_DIR_CAM_BAL_IVVT_IN_i[2][3]	-	FF...1H	-1...1	1	-
Direction of adjustment velocity for its filtering for bank balancing; inlet					
VEL_DIR_CAM_BAL_IVVT_EX_i[2][3]	-	FF...1H	-1...1	1	-
Direction of adjustment velocity for its filtering for bank balancing; exhaust					

Input data:

NC_NR_CBK_IVVT	LV_ERR_LIH_IVVT	TOIL	N_32
LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	CAM_SP_IVVT_IN	CAM_SP_IVVT_EX
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	N	
LV_ERR_CAM_DE_IVVT_IN_i	LV_ERR_CAM_DE_IVVT_EX_i		
LV_ERR_SLV_IVVT_IN_i	LV_ERR_SLV_IVVT_EX_i		
LV_TRIP_ACT_IVVT_IN	LV_TRIP_ACT_IVVT_EX		

FUNCTION DESCRIPTION:

General information:

The position control of each camshaft is realized with the individual setpoint. This enables different setpoints for bank 1 and bank 2, which is used for the bank position balancing.

9.58.1 IVVT Camshaft Position for Balancing

Description:

This subfunction is active only for NC_NR_CBK_IVVT = 2. It is realized in the scheduler.

The evaluation of the differences between the camshafts in individual banks must not be disturbed by the jitter of the position signal or stochastic camshaft movements. Therefore the camshaft position is filtered. The filtering method is the moving mean value creation.

Application conditions:

Initialization:

 # NC_NR_CBK_IVVT = 2:

At reset, at transition "Engine run" → "Engine stop":

CAM_MMV_BAL_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

In every engine state

Formula section:

 # NC_NR_CBK_IVVT = 2:

CAM_MMV_BAL_IVVT_IN(_EX)_i(n) =

CAM_MMV_BAL_IVVT_IN(_EX)_i(n-1) + C_CRLC_CAM_BAL_IVVT *

(CAM_AV_IVVT_IN(_EX)_i - CAM_MMV_BAL_IVVT_IN(_EX)_i(n-1))

9.58.2 IVVT Bank Balancing

Description:

This subfunction is active only for NC_NR_CBK_IVVT = 2. It is realized in the scheduler.

Application conditions:

Initialization:

 # NC_NR_CBK_IVVT = 2:

At reset, at deactivation, at transition "Engine run" → "Engine stop":

CAM_SP_BAL_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

LV_ACT_VEL_BAL_IVVT_IN(_EX)_i = 0

Recurrence:

360 °CRK

Activation:

LV_ACT_IND_IVVT_IN(_EX)_1 = 1

and (LV_ACT_IND_IVVT_IN(_EX)_2 = 1
or LV_NOT_ADJ_CAM_IVVT_IN(_EX)_2 = 1)
and LV_TRIP_ACT_IVVT_IN(_EX) = 0

Deactivation:

Not activation

Formula section:

*# **NC_NR_CBK_IVVT = 2:***

```
if      LV_NOT_ADJ_CAM_IVVT_IN(_EX)_2 = 1
then    2 banks configured, but the second one switched off
        CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_SP_IVVT_IN(_EX)
else    Adjustments in both banks
        if              LV_ERR_LIH_IVVT = 1
                and     LC_ENA_LIH_BAL_IVVT = 1
        then    Failure detected
                Set balancing errors
                if              LV_ERR_BAL_IVVT_IN(_EX)_1 = 1
                        and     LV_ERR_BAL_IVVT_IN(_EX)_2 = 0
                then    Failure in bank 1
                        Position bank 1 is setpoint for bank 2
                elseif          LV_ERR_BAL_IVVT_IN(_EX)_1 = 0
                        and     LV_ERR_BAL_IVVT_IN(_EX)_2 = 1
                then    Failure in bank 2
                        Position bank 2 is setpoint for bank 1
                else    Failure in both banks
                        CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_SP_IVVT_IN(_EX)
                        CAM_SP_BAL_IVVT_IN(_EX)_2 = CAM_SP_IVVT_IN(_EX)
                endif
                LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 0
                LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 0
        else    No failure or failure not detected yet
                if              (       VEL_CAM_BAL_IVVT_IN(_EX)_1[2] >
                                        C_VEL_MIN_CAM_BAL_DYN_IVVT
                                and     VEL_CAM_BAL_IVVT_IN(_EX)_2[2] >
                                        C_VEL_MIN_CAM_BAL_DYN_IVVT)
                        or      (       VEL_CAM_BAL_IVVT_IN(_EX)_1[2] <
                                        -C_VEL_MIN_CAM_BAL_DYN_IVVT
```

```
                        and   VEL_CAM_BAL_IVVT_IN(_EX)_2[2] <
                              -C_VEL_MIN_CAM_BAL_DYN_IVVT)
                  or    LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 1
                  or    LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 1
            then
                  Dynamic balancing possibly needed
            else  Stationary balancing possibly needed
                  if          abs(CAM_MMV_BAL_IVVT_IN(_EX)_1 -
                              CAM_MMV_BAL_IVVT_IN(_EX)_2) <=
                              C_CAM_HYS_BAL_STAT_IVVT_IN(_EX)
                  or    [     abs(CAM_AV_IVVT_IN(_EX)_1 -
                              CAM_SP_IVVT_IN(_EX)) <=
                              C_CAM_DIF_MIN_BAL_IVVT
                        and   abs(CAM_AV_IVVT_IN(_EX)_2 -
                              CAM_SP_IVVT_IN(_EX)) <=
                              C_CAM_DIF_MIN_BAL_IVVT]
                  then  Stationary balancing not needed
                        CAM_SP_BAL_IVVT_IN(_EX)_1 =
                        CAM_SP_IVVT_IN(_EX)
                        CAM_SP_BAL_IVVT_IN(_EX)_2 =
                        CAM_SP_IVVT_IN(_EX)
                  else  Stationary balancing needed
                        if    abs(CAM_MMV_BAL_IVVT_IN(_EX)_1 -
                              CAM_SP_IVVT_IN(_EX)) <
                              abs(CAM_MMV_BAL_IVVT_IN(_EX)_2 -
                              CAM_SP_IVVT_IN(_EX))
                        then  Bank 1 closer to setpoint than bank 2
                              Position bank 2 is setpoint for bank 1
                        else  Bank 2 closer to setpoint than bank 1
                              Position bank 1 is setpoint for bank 2
                        endif
                  endif
            endif
      endif
endif
```

Set balancing errors:

LV_ERR_BAL_IVVT_IN(_EX)_1 =

LV_ERR_CAM_DE_IVVT_IN(_EX)_1 **or** LV_ERR_SLV_IVVT_IN(_EX)_1

LV_ERR_BAL_IVVT_IN(_EX)_2 =

LV_ERR_CAM_DE_IVVT_IN(_EX)_2 **or** LV_ERR_SLV_IVVT_IN(_EX)_2

Position bank 1 is setpoint for bank 2:

CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_SP_IVVT_IN(_EX)

if Inlet:

CAM_AV_IVVT_IN_1 >= C_CAM_INI_IN - C_CAM_THD_BAL_IVVT_IN

Exhaust:

CAM_AV_IVVT_EX_1 <= C_CAM_INI_EX + C_CAM_THD_BAL_IVVT_EX

then

CAM_SP_BAL_IVVT_IN(_EX)_2 = CAM_AV_IVVT_IN(_EX)_1

else

CAM_SP_BAL_IVVT_IN(_EX)_2 =

Inlet:

max(CAM_AV_IVVT_IN_1,CAM_SP_IVVT_IN)

Exhaust:

min(CAM_AV_IVVT_EX_1,CAM_SP_IVVT_EX)

endif

Position bank 2 is setpoint for bank 1:

CAM_SP_BAL_IVVT_IN(_EX)_2 = CAM_SP_IVVT_IN(_EX)

if Inlet:

CAM_AV_IVVT_IN_2 >= C_CAM_INI_IN - C_CAM_THD_BAL_IVVT_IN

Exhaust:

CAM_AV_IVVT_EX_2 <= C_CAM_INI_EX + C_CAM_THD_BAL_IVVT_EX

then

CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_AV_IVVT_IN(_EX)_2

else

CAM_SP_BAL_IVVT_IN(_EX)_1 =

Inlet:

max(CAM_AV_IVVT_IN_2,CAM_SP_IVVT_IN)

Exhaust:

min(CAM_AV_IVVT_EX_2,CAM_SP_IVVT_EX)

endif

Dynamic balancing possibly needed:

```
if      abs(CAM_MMV_BAL_IVVT_IN(_EX)_1 -
        CAM_MMV_BAL_IVVT_IN(_EX)_2) <=
        ID_CAM_HYS_BAL_DYN_IVVT_IN(_EX)
    or  abs(CAM_AV_IVVT_IN(_EX)_1 - CAM_SP_IVVT_IN(_EX)) <=
        C_CAM_DIF_MIN_BAL_IVVT
    or  abs(CAM_AV_IVVT_IN(_EX)_2 - CAM_SP_IVVT_IN(_EX))<=
        C_CAM_DIF_MIN_BAL_IVVT
then    Balancing due to unequal adjustment velocities not needed
    LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 0
    LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 0
    CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_SP_IVVT_IN(_EX)
    CAM_SP_BAL_IVVT_IN(_EX)_2 = CAM_SP_IVVT_IN(_EX)
else    Balancing due to unequal adjustment velocities needed
    if      abs(CAM_MMV_BAL_IVVT_IN(_EX)_1 - CAM_SP_IVVT_IN(_EX)) <
            abs(CAM_MMV_BAL_IVVT_IN(_EX)_2 - CAM_SP_IVVT_IN(_EX))
    then    Bank 1 faster than bank 2
        if      LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 0
        then
            if      VEL_CAM_BAL_IVVT_IN(_EX)_1[2] > 0
            then
                    CAM_SP_BAL_IVVT_IN(_EX)_1 =
                    CAM_AV_IVVT_IN(_EX)_1 +
                    ID_CAM_HYS_BAL_DYN_IVVT_IN(_EX) *
                    C_FAC_CAM_HYS_BAL_DYN_IVVT
            else
                    CAM_SP_BAL_IVVT_IN(_EX)_1 =
                    CAM_AV_IVVT_IN(_EX)_1 -
                    ID_CAM_HYS_BAL_DYN_IVVT_IN(_EX) *
                    C_FAC_CAM_HYS_BAL_DYN_IVVT
            endif
        endif
        CAM_SP_BAL_IVVT_IN(_EX)_2 = CAM_SP_IVVT_IN(_EX)
        LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 1
        LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 0
    else    Bank 2 faster than bank 1
        if      LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 0
```

then

if VEL_CAM_BAL_IVVT_IN(_EX)_2[2] > 0

then

CAM_SP_BAL_IVVT_IN(_EX)_2 =
CAM_AV_IVVT_IN(_EX)_2 +
ID_CAM_HYS_BAL_DYN_IVVT_IN(_EX) *
C_FAC_CAM_HYS_BAL_DYN_IVVT

else

CAM_SP_BAL_IVVT_IN(_EX)_2 =
CAM_AV_IVVT_IN(_EX)_2 -
ID_CAM_HYS_BAL_DYN_IVVT_IN(_EX) *
C_FAC_CAM_HYS_BAL_DYN_IVVT

endif

endif

CAM_SP_BAL_IVVT_IN(_EX)_1 = CAM_SP_IVVT_IN(_EX)

LV_ACT_VEL_BAL_IVVT_IN(_EX)_1 = 0

LV_ACT_VEL_BAL_IVVT_IN(_EX)_2 = 1

endif

endif

9.58.3 IVVT Adjustment Velocity for Bank Balancing

Description:

This subfunction is active only for NC_NR_CBK_IVVT = 2. It is realized in the scheduler.

Application conditions:

Initialization:

 # NC_NR_CBK_IVVT = 2:

At reset, at deactivation, at transition "Engine run" → "Engine stop":

VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[0...1][0...2] = 0

VEL_CAM_BAL_IVVT_IN(_EX)_i[0...2] = 0

Recurrence:

360 °CRK

Activation:

LV_ACT_IND_IVVT_IN(_EX)_i = 1

Deactivation:

Not activation

Formula section:

Calculate preliminary adjustment velocity:

VEL_CAM_BAL_IVVT_IN(_EX)_i[0] {°CRK/s} =
(CAM_MMV_BAL_IVVT_IN(_EX)_i(n) {°CRK} –
CAM_MMV_BAL_IVVT_IN(_EX)_i(n-1) {°CRK}) /
(60 {sec/min} / N {1/min})

Filter adjustment velocity (2-step filtering):

```
for     m = 0 to 1
        VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][0] =
        sign(   VEL_CAM_BAL_IVVT_IN(_EX)_i[m](n) -
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m](n-1)    )
        if              VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][0] =
                        VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][1]
                and     VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][0] =
                        VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][2]
        then
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m+1] =
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m]
        else
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m+1] =
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m+1](n-1) + 0.5 *
                (VEL_CAM_BAL_IVVT_IN(_EX)_i[m] -
                VEL_CAM_BAL_IVVT_IN(_EX)_i[m+1](n-1))
        endif
        for     f = 2 to 1
                VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][f] =
                VEL_DIR_CAM_BAL_IVVT_IN(_EX)_i[m][f-1]
        endfor
endfor
```

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_ENA_LIH_BAL_IVVT	1	0...1H	0...1	1	-
1 = enable limp home balancing algorithm, 0 = use always standard balancing algorithm					
C_VEL_MIN_CAM_BAL_DYN_IVVT	1	0...7FFFH	0...12287.625	0.375	°CRK/s
Minimum adjustment velocity for dynamic bank balancing					
C_CRLC_CAM_BAL_IVVT	1	0...FFH	0...0.99609	0.0039	-
Correlation constant for filtered camshaft position for bank balancing					
C_CAM_DIF_MIN_BAL_IVVT	1	0...FFH	0...95.625	0.375	°CRK
Minimum camshaft deviation from setpoint for activating bank balancing					
C_CAM_THD_BAL_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Camshaft position threshold of faulty bank for considering setpoint at balancing; inlet					
C_CAM_THD_BAL_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Camshaft position threshold of faulty bank for considering setpoint at balancing; exhaust					
C_FAC_CAM_HYS_BAL_DYN_IVVT	1	0...FFH	0...1.992	0.0078	-
Factor for considering actuator inertia at balancing					
ID_CAM_HYS_BAL_DYN_IVVT_IN	4x4	0...FFH	0...95.625	0.375	°CRK
LDPM_N_32_6_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_5_VVTI	4	0...C8H	-40...160	1	°C
Camshaft hysteresis for activating dynamic bank balancing; inlet					
ID_CAM_HYS_BAL_DYN_IVVT_EX	4x4	0...FFH	0...95.625	0.375	°CRK
LDPM_N_32_6_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_5_VVTI	4	0...C8H	-40...160	1	°C
Camshaft hysteresis for activating dynamic bank balancing; exhaust					
C_CAM_HYS_BAL_STAT_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Camshaft hysteresis for activating static bank balancing; inlet					
C_CAM_HYS_BAL_STAT_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Camshaft hysteresis for activating static bank balancing; exhaust					

9.59 IVVT Controller

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ACT_CTL_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = IVVT controller active, 0 = inactive; inlet					
LV_ACT_CTL_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = IVVT controller active, 0 = inactive; exhaust					
T_IVVTPWM_IN_i	O/V	0...FFFFFFFFH	0...17179.87	4e-6	s
Duration of adjustment energization; inlet					
T_IVVTPWM_EX_i	O/V	0...FFFFFFFFH	0...17179.87	4e-6	s
Duration of adjustment energization; exhaust					
IVVTPWM_IN_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Pulse width modulation for adjustment energization; inlet					
IVVTPWM_EX_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Pulse width modulation for adjustment energization; exhaust					
STATE_CTL_IVVT_IN_i	O/V	0H 1H 2H 3H	INIT HLD_DLY HLD_ADAP ADJ	1	-
States of IVVT control; inlet					
STATE_CTL_IVVT_EX_i	O/V	0H 1H 2H 3H	INIT HLD_DLY HLD_ADAP ADJ	1	-
States of IVVT control; exhaust					
STATE_PWM_IVVT_IN_i	O/V	0H 1H 2H	HLD RTD ADC	1	-
States of IVVT pulse width modulation energization; inlet					
STATE_PWM_IVVT_EX_i	O/V	0H 1H 2H	HLD RTD ADC	1	-
States of IVVT pulse width modulation energization; exhaust					
CAM_DIF_IVVT_IN_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between actual camshaft position and camshaft position setpoint; inlet					
CAM_DIF_IVVT_EX_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between actual camshaft position and camshaft position setpoint; exhaust					
CAM_SP_DIF_IVVT_IN_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between camshaft position setpoint and camshaft position modeled with actuator with no delay; inlet					
CAM_SP_DIF_IVVT_EX_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between camshaft position setpoint and camshaft position modeled with actuator with no delay; exhaust					
CAM_MDL_DIF_IVVT_IN_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between actual camshaft position and modeled camshaft position; inlet					
CAM_MDL_DIF_IVVT_EX_i	O/V	FF60H...A0H	-60...60	0.375	°CRK
Difference between actual camshaft position and modeled camshaft position; exhaust					
LV_ADJ_END_PREV_IVVT_IN_i	O/V	0...1H	0...1	1	-
At beginning of controller step: 1 = adjustment energization end in previous controller step recognized, 0 = not recognized; inlet					
LV_ADJ_END_PREV_IVVT_EX_i	O/V	0...1H	0...1	1	-
At beginning of controller step: 1 = adjustment energization end in previous controller step recognized, 0 = not recognized; exhaust					
LV_ADJ_RTD_IVVT_IN_i	O/V	0...1H	0...1	1	-
Adjustment direction: 1 = RTD, 0 = ADC; inlet					
LV_ADJ_RTD_IVVT_EX_i	O/V	0...1H	0...1	1	-
Adjustment direction: 1 = RTD, 0 = ADC; exhaust					

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ADJ_CHG_DIR_IVVT_IN_i	O/V	0...1H	0...1	1	-
Changing adjustment direction; inlet					
LV_ADJ_CHG_DIR_IVVT_EX_i	O/V	0...1H	0...1	1	-
Changing adjustment direction; exhaust					
CTR_CAM_DIF_IVVT_IN_i	O/V	0...FFH	0...255	1	-
Counter increasing by 1 at every camshaft position deviation calculation; inlet; with overflow					
CTR_CAM_DIF_IVVT_EX_i	O/V	0...FFH	0...255	1	-
Counter increasing by 1 at every camshaft position deviation calculation; exhaust; with overflow					
DLY_SLV_THE_IVVT_IN_i	V	0...FFH	0...131.072	0.512	ms
Theoretical solenoid valve delay (component property); inlet					
DLY_SLV_THE_IVVT_EX_i	V	0...FFH	0...131.072	0.512	ms
Theoretical solenoid valve delay (component property); exhaust					
DLY_SLV_OPEN_IVVT_IN_i	V	0...FFH	0...131.072	0.512	ms
Time necessary to open solenoid valve (including previous energization history) at the beginning of controller step; inlet					
DLY_SLV_OPEN_IVVT_EX_i	V	0...FFH	0...131.072	0.512	ms
Time necessary to open solenoid valve (including previous energization history) at the beginning of controller step; exhaust					
DLY_SLV_IVVT_IN_i	V	0...FFH	0...131.072	0.512	ms
Solenoid valve delay considered in controller; inlet					
DLY_SLV_IVVT_EX_i	V	0...FFH	0...131.072	0.512	ms
Solenoid valve delay considered in controller; exhaust					
T_ADJ_EFC_IVVT_IN_i	V	0...FFFFH	0...33553.92	0.512	ms
Effective performed adjustment energization time in previous controller step; inlet					
T_ADJ_EFC_IVVT_EX_i	V	0...FFFFH	0...33553.92	0.512	ms
Effective performed adjustment energization time in previous controller step; exhaust					
CAM_MDL_DLY_NOT_IVVT_IN_i	V	0...FFFFH	60...155.9985	1.465e-3	°CRK
Camshaft position modeled with actuator with no delay; inlet					
CAM_MDL_DLY_NOT_IVVT_EX_i	V	0...FFFFH	-40.125... -136.1235	1.465e-3	°CRK
Camshaft position modeled with actuator with no delay; exhaust					
CAM_SP_PREV_IVVT_IN_i	-	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint in previous controller step; inlet					
CAM_SP_PREV_IVVT_EX_i	-	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint in previous controller step; exhaust					
T_CTL_INT_RESI_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Residual integrated adjustment energization time from P-control; inlet					
T_CTL_INT_RESI_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Residual integrated adjustment energization time from P-control; exhaust					
T_PCTL_LIM_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Limited adjustment energization time from Pre-control; inlet					
T_PCTL_LIM_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Limited adjustment energization time from Pre-control; exhaust					
T_CTL_INT_LIM_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Limited integrated adjustment energization time from P-control; inlet					
T_CTL_INT_LIM_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Limited integrated adjustment energization time from P-control; exhaust					
T_PCTL_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Adjustment energization time from Pre-control; inlet					
T_PCTL_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
Adjustment energization time from Pre-control; exhaust					
VEL_IVS_ADC_IVVT_IN_i	-	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
Inverse actuator adjustment speed in advanced direction; inlet					
VEL_IVS_ADC_IVVT_EX_i	-	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
Inverse actuator adjustment speed in advanced direction; exhaust					
VEL_IVS_RTD_IVVT_IN_i	-	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
Inverse actuator adjustment speed in retard direction; inlet					
VEL_IVS_RTD_IVVT_EX_i	-	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
Inverse actuator adjustment speed in retard direction; exhaust					
CAM_MDL_BUF_IVVT_IN_i [NC_CAM_MDL_BUF_IVVT]	-	0...FFH	60...155.625	0.375	°CRK
Buffer for camshaft position modeling; inlet					
CAM_MDL_BUF_IVVT_EX_i [NC_CAM_MDL_BUF_IVVT]	-	0...FFH	-40.125...-135.75	0.375	°CRK
Buffer for camshaft position modeling; exhaust					
PTR_CAM_MDL_IVVT_IN_i	-	0...31H	0...49	1	-
Pointer pointing to current cell of buffer for camshaft position modeling; inlet					
PTR_CAM_MDL_IVVT_EX_i	-	0...31H	0...49	1	-
Pointer pointing to current cell of buffer for camshaft position modeling; exhaust					
IDX_CAM_MDL_IVVT_IN_i	-	0...31H	0...49	1	-
Index pointing to cell of buffer for camshaft position modeling with delayed value; inlet					
IDX_CAM_MDL_IVVT_EX_i	-	0...31H	0...49	1	-
Index pointing to cell of buffer for camshaft position modeling with delayed value; exhaust					
CAM_MDL_TMP_IVVT_IN_i	-	0...FFH	60...155.625	0.375	°CRK
Modeled camshaft position before limitation to maximum possible change; inlet					
CAM_MDL_TMP_IVVT_EX_i	-	0...FFH	-40.125...-135.75	0.375	°CRK
Modeled camshaft position before limitation to maximum possible change; inlet					
CAM_MDL_DELTA_MAX_IVVT_IN_i	V	0...A0H	0...60	0.375	°CRK
Maximum possible change of camshaft position in controller step; inlet					
CAM_MDL_DELTA_MAX_IVVT_EX_i	V	0...A0H	0...60	0.375	°CRK
Maximum possible change of camshaft position in controller step; exhaust					
CAM_MDL_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Modeled camshaft position; inlet					
CAM_MDL_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Modeled camshaft position; exhaust					
LV_PCTL_RTD_IVVT_IN_i	V	0...1H	0...1	1	-
Pre-control direction: 1 = RTD, 0 = ADC; inlet					
LV_PCTL_RTD_IVVT_EX_i	V	0...1H	0...1	1	-
Pre-control direction: 1 = RTD, 0 = ADC; exhaust					
DLY_IVVT_IN_i	V	0...FFH	0...2550	10	ms
Actuator delay; inlet					
DLY_IVVT_EX_i	V	0...FFH	0...2550	10	ms
Actuator delay; exhaust					
T_CTL_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Adjustment energization time from P-control; inlet					
T_CTL_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Adjustment energization time from P-control; exhaust					
T_CTL_INT_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Integrated adjustment energization time from P-control; inlet					
T_CTL_INT_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
Integrated adjustment energization time from P-control; exhaust					
T_SUM_WOUT_SLV_DLY_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Sum of Pre-control and P-control energization times without considering solenoid valve delay; inlet					
T_SUM_WOUT_SLV_DLY_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Sum of Pre-control and P-control energization times without considering solenoid valve delay; exhaust					
T_SUM_IVVT_IN_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Sum of Pre-control and P-control energization times with considering solenoid valve delay; inlet					
T_SUM_IVVT_EX_i	V	8000H...7FFFH	-16777.216 ...16776.704	0.512	ms
Sum of Pre-control and P-control energization times with considering solenoid valve delay; exhaust					
T_HLD_DLY_ST_IVVT_IN_i	-	0...FFFFH	0...655350	10	ms
Time at transition into IVVT controller state "HLD_DLY"; inlet					
T_HLD_DLY_ST_IVVT_EX_i	-	0...FFFFH	0...655350	10	ms
Time at transition into IVVT controller state "HLD_DLY"; exhaust					

Input data:

LV_VLD_PSN_CAM_IN_i	LV_VLD_PSN_CAM_EX_i	N_32	TOIL
CAM_SP_IND_IVVT_IN_i	CAM_SP_IND_IVVT_EX_i	VB	TMAG_IVVT
C_CAM_SP_HYS_NEUT_IVVT_IN	DLY_ADC_IVVT_IN	DLY_ADC_IVVT_EX	C_CAM_INI_EX
C_CAM_SP_HYS_NEUT_IVVT_EX	DLY_RTD_IVVT_IN	DLY_RTD_IVVT_EX	C_CAM_INI_IN
T_DIF_EDGE_CAM_IVVT_IN_i	LV_ACT_IND_IVVT_IN_i	FRQ_IVVTPWM	T_DLY_MES_IVVT
T_DIF_EDGE_CAM_IVVT_EX_i	LV_ACT_IND_IVVT_EX_i	CAM_AV_IVVT_IN[NC_NR_CBK_IVVT]	CAM_AV_IVVT_EX[NC_NR_CBK_IVVT]
LV_IVVTPWM_SUB_IN_i	LV_IVVTPWM_SUB_EX_i	IVVTPWM_RTD_IN	IVVTPWM_ADC_IN
IVVTPWM_ADC_EX	IVVTPWM_RTD_EX	TMAG_IVVT_EX	

Import actions:

ACTION_INFR_StartPulseIvvt(IN < Ivvt >, IN < Ivvtpwm >, IN < Frq_Ivvtpwm >, IN < T_Ivvtpwm >)
This action starts a time limited pulse width modulation energisation for the camshaft Ivvt.
ACTION_INFR_StopPulseIvvt(IN < Ivvt >)
This action stops the time limited pulse width modulation energisation for the camshaft Ivvt.

FUNCTION DESCRIPTION:

General information:

The IVVT controller is the main tool to adjust the target camshaft position. Small deviations or a drift of the camshaft position from the setpoint are managed by the holding pulse width modulation adaptations.

Description:

There are several kinds of the solenoid valve energization. The low level energization is an energization with PWM close to 0 %. The camshaft moves and remains in the reference position (passive stop position). If the solenoid valve is energized with the high level PWM, i.e., close to 100 %, the camshaft moves and remains in the active stop position. If a camshaft position somewhere in the adjustment range is to hold the holding pulse width modulation energization is used. It is about 40 to 60 % PWM.

Generally, the holding energization is applied. It is interrupted by time limited low or high level energizations (adjustment energizations) if there is a change of the camshaft position setpoint or the actual position does not correspond with the setpoint. If the energization time is very short (less than ~3-7 ms, tuneable) the low level is substituted by 0 % and the high level by 100 %. It does not disturb the output signal diagnosis, see "IVVT Output Signal Diagnosis". The reason for using 0 % level for short energization times is obvious from the figure "Short energization time". There is a similar situation for 100 % when "OFF" start is used.

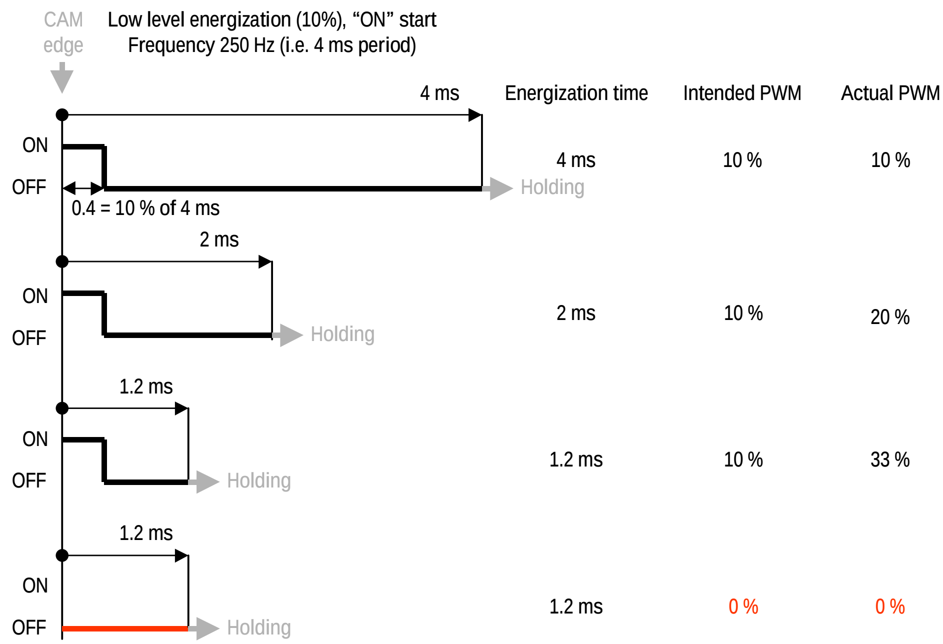


Figure: Short energization time.

The main goals of the IVVT controlling are to exploit the maximum design adjustment speed of the actuator without control overshoots, to keep the deviation of the camshaft position from the setpoint small, to keep the actuator in the holding energization without control interventions as long as possible.

There are three characteristics of the system influencing the design of the IVVT controller. They are the actuator delay, the assumption of linear position independent adjustment speed and the solenoid valve delay.

Due to the actuator delay it would not be suitable to use a control comparing the actual camshaft position and the setpoint. The delay would cause that there would be a big system deviation for a long time. In order to avoid overshoots a slow control would be necessary. The control behaviour would have long undershoots.

Therefore, Pre-control is used in the IVVT controller together with P-control. A special camshaft position is calculated. It is a modelling with an fictitious actuator without the delay, i.e., non-delayed camshaft position. The difference between the camshaft position setpoint and the non-delayed position is used for the Pre-control. The non-delayed position follows immediately a change of the setpoint with the actuator adjustment speed. If the non-delayed position reaches the new setpoint after the end of its change the Pre-control becomes inactive. Also, the Pre-control responds quickly on the setpoint changes and is not too long due to the actuator delay. The Pre-control output is an adjustment energization time.

The Pre-control roughly reaches the setpoint. It is a modelling, also, it must be applied with certain safety in order not to have overshoots. Therefore, the residual system deviation is controlled with P-control. The non-delayed camshaft position is delayed by means of a buffer. This leads to a modelled camshaft position. The difference between the actual camshaft position and the modelled one is used for the P-control. The P-control output is an adjustment energization time, too.

The adjustment energization times from the Pre-control and P-control are summed. The resultant energization time plus the solenoid valve delay is then the controller output. The desired adjustment direction determines whether low or high level energization is taken.

If there are only small system deviations very short energization times are calculated. If they are smaller than a threshold they are integrated. If the threshold is reached the integrated energization time is carried out and the integral is set to 0. The integral is set to 0, too, if an energization into the opposite direction is needed.

The controller is called at each used camshaft edge, see "IVVT Scheduler". The current camshaft position is determined at the camshaft edge. The energization time calculated by the controller can be longer as the time between two used camshaft edges. This effect is considered in the controller. For example, the solenoid valve is opened and its delay need not be added any more.

Hint to the moment of usage and moment of calculation of some variables: Some quantities specified in "Output data" are used in the controller before they are assigned to. It means that the values from the previous controller step are used. Therefore, the initializations are very important, too.

The signal flow within the controller is shown in Figure "Controller signal flow" schematically. The names of quantities are abbreviated, i.e. _IVVT_IN(_EX)_i is avoided for the sake of simplicity and clarity.

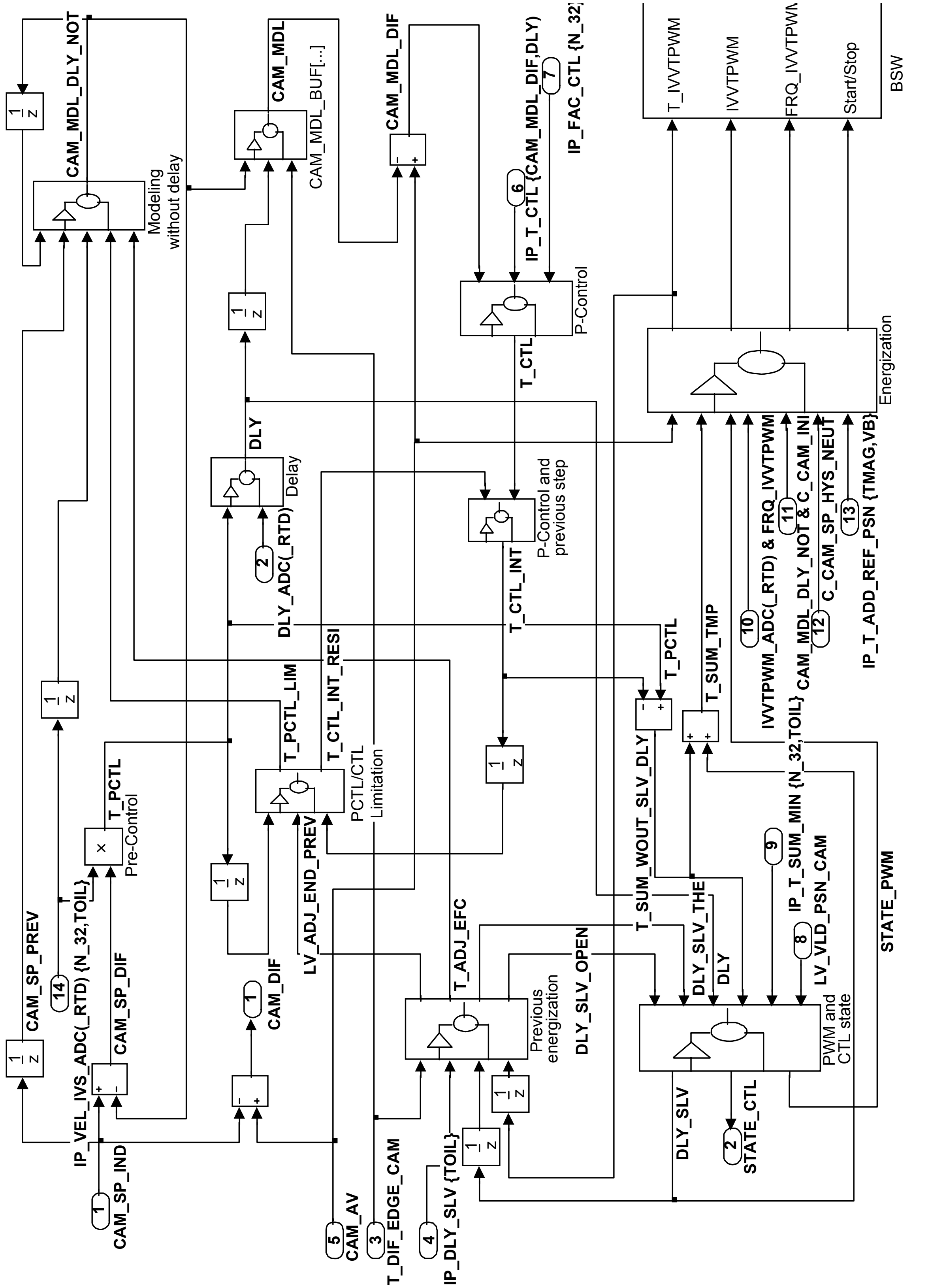


Figure: Controller signal flow.

Application conditions:

Recurrence: At each used camshaft edge, see "IVVT Scheduler"

Activation: LV_ACT_IND_IVVT_IN(_EX)_i = 1
and N_32 <= C_N_32_DI_CTL_IVVT
and LV_IVVTPWM_SUB_IN(_EX)_i = 0

Deactivation: LV_ACT_IND_IVVT_IN(_EX)_i = 0
or N_32 >= C_N_32_DI_CTL_IVVT + C_N_32_HYS_DI_CTL_IVVT
or LV_IVVTPWM_SUB_IN(_EX)_i = 1

Initialization: **At reset:**

STATE_CTL_IVVT_IN(_EX)_i = "INIT"

STATE_PWM_IVVT_IN_i = "RTD", STATE_PWM_IVVT_EX_i = "ADC"

CAM_SP_PREV_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

for m = 0 **to** NC_CAM_MDL_BUF_IVVT - 1
    CAM_MDL_BUF_IVVT_IN(_EX)_i[m] = C_CAM_INI_IN(_EX)
endfor

CAM_MDL_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

At activation:

LV_ACT_CTL_IVVT_IN(_EX)_i = 1

At deactivation and at transition "Engine run" to "Engine stop":

LV_ACT_CTL_IVVT_IN(_EX)_i = 0

yy = IVVT_IN(_EX)_i

ACTION_INFR_StopPulseIvvt(IN yy)

T_IVVTPWM_IN(_EX)_i = 0

STATE_CTL_IVVT_IN(_EX)_i = "INIT"

STATE_PWM_IVVT_IN_i = "RTD", STATE_PWM_IVVT_EX_i = "ADC"

LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 0

T_CTL_INT_RESI_IVVT_IN(_EX)_i = 0

T_PCTL_IVVT_IN(_EX)_i = 0

T_CTL_INT_IVVT_IN(_EX)_i = 0

CAM_SP_PREV_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

for m = 0 **to** NC_CAM_MDL_BUF_IVVT - 1
    CAM_MDL_BUF_IVVT_IN(_EX)_i[m] = C_CAM_INI_IN(_EX)
endfor

CAM_MDL_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

CAM_DIF_IVVT_IN(_EX)_i = 0

Formula section:

xx = IN(_EX)_i

yy = IVVT_IN(_EX)_i

Adjustment energization state at controller call

Evaluation of previous controller step and camshaft position modeling with actuator with no delay

Modeled camshaft position

Pre-control

if$_1$ LV_VLD_PSN_CAM_IN(_EX)_i = 0

then$_1$ Camshaft position information not valid sporadically

else$_1$ P-control

PWM state

Controller state

Adjustment energization

Deviation of camshaft position from setpoint

endif$_1$

Adjustment energization state at controller call:

DLY_SLV_THE_IVVT_IN(_EX)_i = IP_DLY_SLV_IVVT_IN(_EX)

if$_1$ T_IVVTPWM_IN(_EX)_i <= T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

and T_IVVTPWM_IN(_EX)_i > 0

then$_1$ *Adjustment energization end in previous controller step recognized*

LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 1

T_ADJ_EFC_IVVT_IN(_EX)_i =

T_IVVTPWM_IN(_EX)_i - DLY_SLV_IVVT_IN(_EX)_i

if$_2$ T_DIF_EDGE_CAM_IVVT_IN(_EX)_i - T_IVVTPWM_IN(_EX)_i >=

DLY_SLV_THE_IVVT_IN(_EX)_i

then$_2$ *There has been enough time to close, i.e., to reach holding position, solenoid valve*

DLY_SLV_OPEN_IVVT_IN(_EX)_i = DLY_SLV_THE_IVVT_IN(_EX)_i

else$_2$ *Solenoid valve has not fully reached holding position yet*

DLY_SLV_OPEN_IVVT_IN(_EX)_i =

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i - T_IVVTPWM_IN(_EX)_i

endif$_2$

else$_1$ *Adjustment energization end in previous step not recognized, i.e., previous adjustment energization has not been finished yet or no adjustment energization has been requested in previous controller step*

LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 0

if$_3$ T_IVVTPWM_IN(_EX)_i > 0

then$_3$ *Previous adjustment energization has not been finished yet*

T_ADJ_EFC_IVVT_IN(_EX)_i =

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i - DLY_SLV_IVVT_IN(_EX)_i

DLY_SLV_OPEN_IVVT_IN(_EX)_i = 0

else$_3$ *No adjustment energization has been requested in previous controller step*

T_ADJ_EFC_IVVT_IN(_EX)_i = 0

DLY_SLV_OPEN_IVVT_IN(_EX)_i = DLY_SLV_THE_IVVT_IN(_EX)_i

endif$_3$

endif$_1$

<u>Evaluation of previous controller step and camshaft position modeling with actuator with no delay:</u>

if$_1$ LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 1

or STATE_PWM_IVVT_IN(_EX)_i = "HLD"

then$_1$ *Adjustment energization end in previous controller step recognized or no adjustment energization has been requested in previous controller step*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i = CAM_SP_PREV_IVVT_IN(_EX)_i

if$_2$ LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 1

then$_2$ *Adjustment energization end in previous controller step recognized*

T_CTL_INT_RESI_IVVT_IN(_EX)_i = 0

else$_2$ *No adjustment energization has been requested in previous controller step*

T_CTL_INT_RESI_IVVT_IN(_EX)_i = T_CTL_INT_IVVT_IN(_EX)_i

endif$_2$

else$_1$ *Previous adjustment energization has not been finished yet*

if$_3$ T_PCTL_IVVT_IN(_EX)_i > 0

then$_3$ *Pre-control RTD*

if$_4$ T_CTL_INT_IVVT_IN(_EX)_i > 0

then$_4$ *P-control ADC and Pre-control RTD*

if$_5$ T_CTL_INT_IVVT_IN(_EX)_i > T_PCTL_IVVT_IN(_EX)_i

then$_5$ *P-control over Pre-control*

T_PCTL_LIM_IVVT_IN(_EX)_i = 0

T_CTL_INT_LIM_IVVT_IN(_EX)_i =

T_CTL_INT_IVVT_IN(_EX)_i - T_PCTL_IVVT_IN(_EX)_i

else$_5$ *Pre-control over P-control*

T_PCTL_LIM_IVVT_IN(_EX)_i =

T_PCTL_IVVT_IN(_EX)_i - T_CTL_INT_IVVT_IN(_EX)_i

T_CTL_INT_LIM_IVVT_IN(_EX)_i = 0

endif$_5$

else$_4$ *Pre-control ≈ P-control (regular case)*

T_PCTL_LIM_IVVT_IN(_EX)_i = T_PCTL_IVVT_IN(_EX)_i

T_CTL_INT_LIM_IVVT_IN(_EX)_i = T_CTL_INT_IVVT_IN(_EX)_i

endif$_4$

else$_3$ *Pre-control ADC*

if$_6$ T_CTL_INT_IVVT_IN(_EX)_i < 0

then$_6$ *P-control RTD and Pre-control ADC*

if$_7$ T_CTL_INT_IVVT_IN(_EX)_i < T_PCTL_IVVT_IN(_EX)_i

then$_7$ *P-control over Pre-control*

T_PCTL_LIM_IVVT_IN(_EX)_i = 0

T_CTL_INT_LIM_IVVT_IN(_EX)_i =

T_CTL_INT_IVVT_IN(_EX)_i - T_PCTL_IVVT_IN(_EX)_i

else$_7$ *Pre-control over P-control*

T_PCTL_LIM_IVVT_IN(_EX)_i =

T_PCTL_IVVT_IN(_EX)_i - T_CTL_INT_IVVT_IN(_EX)_i

T_CTL_INT_LIM_IVVT_IN(_EX)_i = 0

endif$_7$

else$_6$ *Pre-control ≈ P-control (regular case)*

T_PCTL_LIM_IVVT_IN(_EX)_i = T_PCTL_IVVT_IN(_EX)_i

T_CTL_INT_LIM_IVVT_IN(_EX)_i = T_CTL_INT_IVVT_IN(_EX)_i

endif$_6$

endif$_3$

if$_8$ abs(T_PCTL_LIM_IVVT_IN(_EX)_i) > T_ADJ_EFC_IVVT_IN(_EX)_i

then$_8$ *Pre-control adjustment energization time from previous controller step greater than effective energization time in previous controller step*

if$_9$ T_PCTL_LIM_IVVT_IN(_EX)_i > 0

then$_9$ *Modeling for RTD direction*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i =

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i +

T_ADJ_EFC_IVVT_IN(_EX)_i / VEL_IVS_RTD_IVVT_IN(_EX)_i

if$_{10}$ CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i >

CAM_SP_PREV_IVVT_IN(_EX)_i

then$_{10}$ *Limitation to possible position*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i =

CAM_SP_PREV_IVVT_IN(_EX)_i

endif$_{10}$

else$_{9}$ *Modeling for ADC direction*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i =

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i -

T_ADJ_EFC_IVVT_IN(_EX)_i / VEL_IVS_ADC_IVVT_IN(_EX)_i

if$_{11}$ CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i <

CAM_SP_PREV_IVVT_IN(_EX)_i

then$_{11}$ *Limitation to possible position*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i =

CAM_SP_PREV_IVVT_IN(_EX)_i

endif$_{11}$

endif$_{9}$

T_CTL_INT_RESI_IVVT_IN(_EX)_i =

T_CTL_INT_LIM_IVVT_IN(_EX)_i

else$_{8}$ *Pre-control adjustment energization time from previous controller step smaller than effective energization time in previous controller step*

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i =

CAM_SP_PREV_IVVT_IN(_EX)_i

if$_{12}$ T_CTL_INT_LIM_IVVT_IN(_EX)_i <> 0

then$_{12}$

if$_{13}$ T_CTL_INT_LIM_IVVT_IN(_EX)_i > 0

then$_{13}$ *RTD direction*

T_CTL_INT_RESI_IVVT_IN(_EX)_i =

max(0,

T_CTL_INT_LIM_IVVT_IN(_EX)_i -

(T_ADJ_EFC_IVVT_IN(_EX)_i -

abs(T_PCTL_LIM_IVVT_IN(_EX)_i)))

else$_{13}$ *ADC direction*

T_CTL_INT_RESI_IVVT_IN(_EX)_i =

min(0,

T_CTL_INT_LIM_IVVT_IN(_EX)_i +

(T_ADJ_EFC_IVVT_IN(_EX)_i -

abs(T_PCTL_LIM_IVVT_IN(_EX)_i)))

endif$_{13}$

else$_{12}$ T_CTL_INT_RESI_IVVT_IN(_EX)_i = 0

endif$_{12}$

endif$_{8}$

endif$_1$

Saving camshaft position setpoint for next controller step:

CAM_SP_PREV_IVVT_IN(_EX)_i = CAM_SP_IND_IVVT_IN(_EX)_i

Modeled camshaft position:

Updating buffer for modeled camshaft position:

CAM_MDL_BUF_IVVT_IN(_EX)_i[PTR_CAM_MDL_IVVT_IN(_EX)_i] =

CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i

Consideration of actuator delay, i.e., index of cell with camshaft position corresponding to behaviour of actuator with delay:

IDX_CAM_MDL_IVVT_IN(_EX)_i = PTR_CAM_MDL_IVVT_IN(_EX)_i -

(DLY_IVVT_IN(_EX)_i / T_DIF_EDGE_CAM_IVVT_IN(_EX)_i)

CAM_MDL_TMP_IVVT_IN(_EX)_i =

CAM_MDL_BUF_IVVT_IN(_EX)_i[IDX_CAM_MDL_IVVT_IN(_EX)_i]

PTR_CAM_MDL_IVVT_IN(_EX)_i = PTR_CAM_MDL_IVVT_IN(_EX)_i + 1

Maximum possible camshaft position change in previous controller step:

if$_1$ LV_PCTL_RTD_IVVT_IN(_EX)_i = 1

then$_1$ CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i =

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i / VEL_IVS_RTD_IVVT_IN(_EX)_i

else$_1$ CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i =

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i / VEL_IVS_ADC_IVVT_IN(_EX)_i

endif$_1$

if$_2$ CAM_MDL_IVVT_IN(_EX)_i <= CAM_MDL_TMP_IVVT_IN(_EX)_i

then$_2$ *RTD direction*

if$_3$ CAM_MDL_TMP_IVVT_IN(_EX)_i - CAM_MDL_IVVT_IN(_EX)_i >

CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i

then$_3$ CAM_MDL_IVVT_IN(_EX)_i = CAM_MDL_IVVT_IN(_EX)_i +

CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i

else$_3$ CAM_MDL_IVVT_IN(_EX)_i = CAM_MDL_TMP_IVVT_IN(_EX)_i

endif$_3$

else$_2$ *ADC direction*

if$_4$ CAM_MDL_IVVT_IN(_EX)_i - CAM_MDL_TMP_IVVT_IN(_EX)_i >

CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i

then$_4$ CAM_MDL_IVVT_IN(_EX)_i = CAM_MDL_IVVT_IN(_EX)_i -

CAM_MDL_DELTA_MAX_IVVT_IN(_EX)_i

else$_4$ CAM_MDL_IVVT_IN(_EX)_i = CAM_MDL_TMP_IVVT_IN(_EX)_i

endif$_4$

endif$_2$

Pre-control:

CAM_SP_DIF_IVVT_IN(_EX)_i =

CAM_SP_IND_IVVT_IN(_EX) - CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i

if$_1$ CAM_SP_DIF_IVVT_IN(_EX)_i <> 0

then$_1$ *Pre-control active*

if$_2$ CAM_SP_DIF_IVVT_IN(_EX)_i > 0

then$_2$ *RTD direction*

LV_PCTL_RTD_IVVT_IN(_EX)_i = 1

VEL_IVS_RTD_IVVT_IN(_EX)_i = IP_VEL_IVS_RTD_IVVT_IN(_EX)

T_PCTL_IVVT_IN(_EX)_i =

CAM_SP_DIF_IVVT_IN(_EX)_i * VEL_IVS_RTD_IVVT_IN(_EX)_i

else$_2$ *ADC direction*

LV_PCTL_RTD_IVVT_IN(_EX)_i = 0

VEL_IVS_ADC_IVVT_IN(_EX)_i = IP_VEL_IVS_ADC_IVVT_IN(_EX)

T_PCTL_IVVT_IN(_EX)_i =

CAM_SP_DIF_IVVT_IN(_EX)_i * VEL_IVS_ADC_IVVT_IN(_EX)_i

endif$_2$

else$_1$ *Pre-control inactive*

T_PCTL_IVVT_IN(_EX)_i = 0

endif$_1$

Actuator delay:

if$_3$ LV_PCTL_RTD_IVVT_IN(_EX)_i = 1

then$_3$ DLY_IVVT_IN(_EX)_i = DLY_RTD_IVVT_IN(_EX)

else$_3$ DLY_IVVT_IN(_EX)_i = DLY_ADC_IVVT_IN(_EX)

endif$_3$

Camshaft position information not valid sporadically:

ACTION_INFR_StopPulseIvvt(IN yy)

T_IVVTPWM_IN(_EX)_i = 0

STATE_PWM_IVVT_IN(_EX)_i = "HLD"

if₁ STATE_CTL_IVVT_IN(_EX)_i <> "HLD_ADAP"

then₁ STATE_CTL_IVVT_IN(_EX)_i = "HLD_DLY"

T_HLD_DLY_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT

endif₁

P-control:

CAM_MDL_DIF_IVVT_IN(_EX)_i =

CAM_AV_IVVT_IN(_EX)_i - CAM_MDL_IVVT_IN(_EX)_i

if₁ CAM_MDL_DIF_IVVT_IN(_EX)_i <> 0

and abs(CAM_SP_IND_IVVT_IN(_EX)_i - C_CAM_INI_IN(_EX)) >=

C_CAM_SP_HYS_NEUT_IVVT_IN(_EX)

then₁ *P-control active, i.e., P-control system deviation differs from 0 and camshaft position setpoint not close to reference position*

if₂ CAM_MDL_DIF_IVVT_IN(_EX) < 0

then₂ *Camshaft is too much in ADC position*

T_CTL_IVVT_IN(_EX)_i =

-IP_T_CTL_IVVT_IN(_EX) * IP_FAC_CTL_IVVT_IN(_EX)

Use abs(CAM_MDL_DIF_IVVT_IN(_EX)_i) for IP_T_CTL_IVVT_IN(_EX)

if₃ T_CTL_INT_RESI_IVVT_IN(_EX)_i < 0

then₃ T_CTL_INT_IVVT_IN(_EX)_i =

T_CTL_INT_RESI_IVVT_IN(_EX)_i + T_CTL_IVVT_IN(_EX)_i

else₃ T_CTL_INT_IVVT_IN(_EX)_i = T_CTL_IVVT_IN(_EX)_i

endif₃

else₂ *Camshaft is too much in RDC position*

T_CTL_IVVT_IN(_EX)_i =

IP_T_CTL_IVVT_IN(_EX) * IP_FAC_CTL_IVVT_IN(_EX)

Use abs(CAM_MDL_DIF_IVVT_IN(_EX)_i) for IP_T_CTL_IVVT_IN(_EX)

if₄ T_CTL_INT_RESI_IVVT_IN(_EX)_i > 0

then₄ T_CTL_INT_IVVT_IN(_EX)_i =

T_CTL_INT_RESI_IVVT_IN(_EX)_i + T_CTL_IVVT_IN(_EX)_i

else₄ T_CTL_INT_IVVT_IN(_EX)_i = T_CTL_IVVT_IN(_EX)_i

endif₄

endif₂

else$_1$ *P-control inactive*

T_CTL_IVVT_IN(_EX)_i = 0

T_CTL_INT_IVVT_IN(_EX) = 0

endif$_1$

PWM state:

T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i =

T_PCTL_IVVT_IN(_EX)_i - T_CTL_INT_IVVT_IN(_EX)_i

```
if1        abs(T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i) = 0
     or    (    abs(T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i) <
                IP_T_SUM_MIN_IVVT_IN(_EX)
           and  (    STATE_PWM_IVVT_IN(_EX)_i = "HLD"
                or   LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 1
                or   (    T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i < 0
                     and  LV_ADJ_RTD_IVVT_IN(_EX)_i = 1)
                or   (    T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i >= 0
                     and  LV_ADJ_RTD_IVVT_IN(_EX)_i = 0)
                )
           )
```

then$_1$ *No control action necessary*

STATE_PWM_IVVT_IN(_EX)_i = "HLD"

else$_1$ *There will be adjustment energization*

if$_2$ T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i >= 0

then$_2$ *RTD direction*

LV_ADJ_CHG_DIR_IVVT_IN(_EX)_i =

not(LV_ADJ_RTD_IVVT_IN(_EX)_i)

LV_ADJ_RTD_IVVT_IN(_EX)_i = 1

else$_2$ *ADC direction*

LV_ADJ_CHG_DIR_IVVT_IN(_EX)_i =

LV_ADJ_RTD_IVVT_IN(_EX)_i

LV_ADJ_RTD_IVVT_IN(_EX)_i = 0

endif$_2$

Solenoid valve delay:

if$_3$ LV_ADJ_CHG_DIR_IVVT_IN(_EX)_i = 1

then$_3$ DLY_SLV_IVVT_IN(_EX)_i = DLY_SLV_THE_IVVT_IN(_EX)_i

else$_3$ DLY_SLV_IVVT_IN(_EX)_i = DLY_SLV_OPEN_IVVT_IN(_EX)_i

endif$_3$

Consideration of solenoid valve delay:

if₄ LV_ADJ_RTD_IVVT_IN(_EX)_i = 1

then₄ *RTD direction*

STATE_PWM_IVVT_IN(_EX)_i = "RTD"

T_SUM_IVVT_IN(_EX)_i =

T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i +

DLY_SLV_IVVT_IN(_EX)_i

else₄ *ADC direction*

STATE_PWM_IVVT_IN(_EX)_i = "ADC"

T_SUM_IVVT_IN(_EX)_i =

T_SUM_WOUT_SLV_DLY_IVVT_IN(_EX)_i -

DLY_SLV_IVVT_IN(_EX)_i

endif₄

endif₁

Controller state:

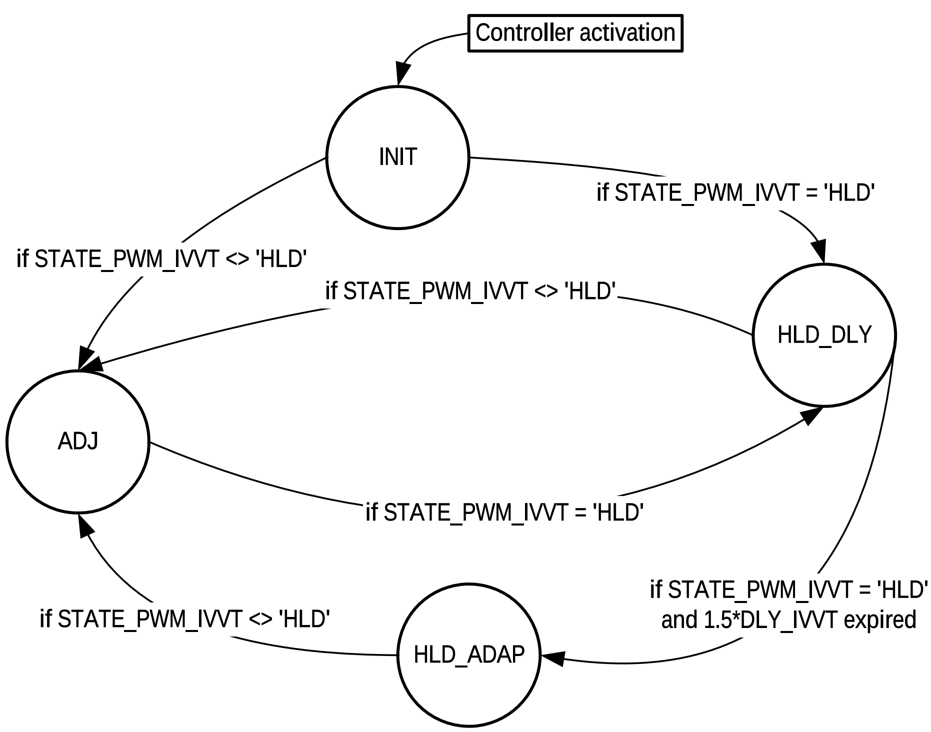


Figure: "IVVT controller state diagram."

STATE_CTL_IVVT_IN(_EX)_i = "INIT":

if₁ STATE_PWM_IVVT_IN(_EX)_i = "HLD"

then₁ STATE_CTL_IVVT_IN(_EX)_i = "HLD_DLY"

```
                T_HLD_DLY_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT
       else1    STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
       endif1
STATE_CTL_IVVT_IN(_EX)_i = "HLD_DLY":
       if2      STATE_PWM_IVVT_IN(_EX)_i = "HLD"
       then2    if3     T_HLD_DLY_ST_IVVT_IN(_EX)_i - T_DLY_MES_IVVT >=
                        1.5 * DLY_IVVT_IN(_EX)
                then3   STATE_CTL_IVVT_IN(_EX)_i = "HLD_ADAP"
                endif3
       else2    STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
       endif2
STATE_CTL_IVVT_IN(_EX)_i = "HLD_ADAP":
       if4      STATE_PWM_IVVT_IN(_EX)_i <> "HLD"
       then4    STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
       endif4
STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
       if5      STATE_PWM_IVVT_IN(_EX)_i = "HLD"
       then5    STATE_CTL_IVVT_IN(_EX)_i = "HLD_DLY"
                T_HLD_DLY_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT
       endif5

Adjustment energization:
       if1      STATE_PWM_IVVT_IN(_EX)_i = "HLD"
       then1    ACTION_INFR_StopPulseIvvt(IN yy)
                T_IVVTPWM_IN(_EX)_i = 0
       else1    if2             abs(CAM_AV_IVVT_IN(_EX)_i - C_CAM_INI_IN(_EX)) <
                                C_CAM_SP_HYS_NEUT_IVVT_IN(_EX)
                        and     CAM_MDL_DLY_NOT_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)
                then2   Camshaft in reference position - additional energization time
                        T_IVVTPWM_IN(_EX)_i = abs(T_SUM_IVVT_IN(_EX)_i) +
                        IP_T_ADD_REF_PSN_IVVT_IN(_EX)
                else2   T_IVVTPWM_IN(_EX)_i = abs(T_SUM_IVVT_IN(_EX)_i)
                endif2
                if3     STATE_PWM_IVVT_IN(_EX)_i = "RTD"
                then3   if4     T_IVVTPWM_IN(_EX)_i > C_T_IVVTPWM_MIN
                        then4   IVVTPWM_IN(_EX)_i = IVVTPWM_RTD_IN(_EX)
                        else4   Inlet:
```

```
                    IVVTPWM_IN_i = 0H
                    Exhaust:
                    IVVTPWM_EX_i = FFFFH
          endif4
    else3 if5       T_IVVTPWM_IN(_EX)_i > C_T_IVVTPWM_MIN
          then5     IVVTPWM_IN(_EX)_i = IVVTPWM_ADC_IN(_EX)
          else5     Inlet:
                    IVVTPWM_IN_i = FFFFH
                    Exhaust:
                    IVVTPWM_EX_i = 0H
          endif5
    endif3
    ACTION_INFR_StartPulseIvvt(
    IN    yy,
    IN    IVVTPWM_xx,
    IN    FRQ_IVVTPWM,
    IN    T_IVVTPWM_xx)
endif1
```

Deviation of camshaft position from setpoint:

CAM_DIF_IVVT_IN(_EX)_i =

CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_IVVT_IN(_EX)_i

CTR_CAM_DIF_IVVT_IN(_EX)_i = CTR_CAM_DIF_IVVT_IN(_EX)_i + 1

Configuration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
NC_CAM_MDL_BUF_IVVT	1	1...32H	1...50	1	-
Buffer size for modelling camshaft position					

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_N_32_DI_CTL_IVVT	1	0...FFH	0...8160	32	rpm
IVVT control is disabled above this engine speed; target position is reference position					
C_N_32_HYS_DI_CTL_IVVT	1	0...FFH	0...8160	32	rpm
Engine speed hysteresis for avoiding too frequent controller activation <-> deactivation transitions at engine speed threshold					
IP_DLY_SLV_IVVT_IN	8	0...FFH	0...131.072	0.512	ms
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Solenoid valve delay; inlet					
IP_DLY_SLV_IVVT_EX	8	0...FFH	0...131.072	0.512	ms
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Solenoid valve delay; exhaust					
IP_VEL_IVS_ADC_IVVT_IN	8x8	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Inverse actuator adjustment speed in advanced direction; inlet					
IP_VEL_IVS_ADC_IVVT_EX	8x8	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Inverse actuator adjustment speed in advanced direction; exhaust					
IP_VEL_IVS_RTD_IVVT_IN	8x8	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Inverse actuator adjustment speed in retard direction; inlet					
IP_VEL_IVS_RTD_IVVT_EX	8x8	1H...FFFFH	0.008...524.28	0.008	ms/(0.375 °CRK)
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Inverse actuator adjustment speed in retard direction; exhaust					
IP_T_CTL_IVVT_IN	8x6	0...FFH	0...131.072	0.512	ms
LDPM_CAM_MDL_DIF_IVVT_1_VVTI	8	0...A0H	0...60	0.375	°CRK
LDPM_DLY_IVVT_1_VVTI	6	0...FFH	0...2550	10	ms
P-control adjustment energization time; inlet					
IP_T_CTL_IVVT_EX	8x6	0...FFH	0...131.072	0.512	ms
LDPM_CAM_MDL_DIF_IVVT_1_VVTI	8	0...A0H	0...60	0.375	°CRK
LDPM_DLY_IVVT_1_VVTI	6	0...FFH	0...2550	10	ms
P-control adjustment energization time; exhaust					
IP_FAC_CTL_IVVT_IN	6	0...FFH	0...3.984375	0.015625	-
LDPM_N_32_2_VVTI	6	0...FFH	0...8160	32	rpm
P-control weighting; inlet					
IP_FAC_CTL_IVVT_EX	6	0...FFH	0...3.984375	0.015625	-
LDPM_N_32_2_VVTI	6	0...FFH	0...8160	32	rpm
P-control weighting; exhaust					
IP_T_SUM_MIN_IVVT_IN	4x8	1H...FFH	0.512...131.072	0.512	ms
LDPM_N_32_3_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Minimum adjustment energization time; inlet					

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_T_SUM_MIN_IVVT_EX	4x8	1H...FFH	0.512...131.072	0.512	ms
LDPM_N_32_3_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Minimum adjustment energization time; exhaust					
IP_T_ADD_REF_PSN_IVVT_IN	4x4	0...FFH	0...131.072	0.512	ms
LDPM_TMAG_IVVT_2_VVTI	4	0...C8H	-40...160	1	°C
LDPM_VB_1_VVTI	4	0...FFH	0..26	0.1	V
Additive adjustment energization time for leaving reference position; inlet					
IP_T_ADD_REF_PSN_IVVT_EX	4x4	0...FFH	0...131.072	0.512	ms
LDPM_TMAG_IVVT_EX_2_VVTI	4	0...C8H	-40...160	1	°C
LDPM_VB_1_VVTI	4	0...FFH	0..26	0.1	V
Additive adjustment energization time for leaving reference position; exhaust					
C_T_IVVTPWM_MIN	1	0...FFH	0...131.072	0.512	ms
Minimum energization time necessary for applying IVVTPWM_ADC(_RTD)_IN(_EX), otherwise 0% and 100%					

9.60 IVVT Delay

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
DLY_ADC_IVVT_IN	O/V	0...FFH	0...2550	10	ms
Calculated adjustment delay; advanced; inlet					
DLY_ADC_IVVT_EX	O/V	0...FFH	0...2550	10	ms
Calculated adjustment delay; advanced; exhaust					
DLY_RTD_IVVT_IN	O/V	0...FFH	0...2550	10	ms
Calculated adjustment delay; retard; inlet					
DLY_RTD_IVVT_EX	O/V	0...FFH	0...2550	10	ms
Calculated adjustment delay; retard; exhaust					

Input data:

LV_ACT_TOT_IVVT_IN	LV_ACT_TOT_IVVT_EX	N_32	TOIL

FUNCTION DESCRIPTION:

General information:

It takes some time till the camshaft phase changes noticeable after the start of the energization. This time is named the adjustment delay. The delay is used in the control algorithm.

Description:

The delay depends especially upon the camshaft (inlet/exhaust), adjustment direction (advanced/retard), engine speed and oil temperature. The delays are in the maps IP_DLY_ADC(_RTD)_IVVT_IN(_EX) depending on the engine speed and oil temperature.

Application conditions:

Initialization:

Recurrence:

360 °CRK linked task, see "IVVT Scheduler"

Activation:

LV_ACT_TOT_IVVT_IN(_EX) = 1

Formula section:

DLY_ADC(_RTD)_IVVT_IN(_EX) = IP_DLY_ADC(_RTD)_IVVT_IN(_EX)

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_DLY_ADC_IVVT_IN	8x8	0...FFH	0...2550	10	ms
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Adjustment delay; advanced; inlet					
IP_DLY_ADC_IVVT_EX	8x8	0...FFH	0...2550	10	ms
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Adjustment delay; advanced; exhaust					
IP_DLY_RTD_IVVT_IN	8x8	0...FFH	0...2550	10	ms
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Adjustment delay; retard; inlet					
IP_DLY_RTD_IVVT_EX	8x8	0...FFH	0...2550	10	ms
LDPM_N_32_1_VVTI	8	0...FFH	0...8160	32	rpm
LDPM_TOIL_1_VVTI	8	0...C8H	-40...160	1	°C
Adjustment delay; retard; exhaust					

9.61 IVVT Measurement of Delay

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
DLY_MES_ADC_IVVT_IN_i	V	0...FFH	0...2550	10	ms
Measured adjustment delay; advanced; inlet					
DLY_MES_ADC_IVVT_EX_i	V	0...FFH	0...2550	10	ms
Measured adjustment delay; advanced; exhaust					
DLY_MES_RTD_IVVT_IN_i	V	0...FFH	0...2550	10	ms
Measured adjustment delay; retard; inlet					
DLY_MES_RTD_IVVT_EX_i	V	0...FFH	0...2550	10	ms
Measured adjustment delay; retard; exhaust					
LV_DLY_MES_ACT_IVVT_IN_i	V	0...1H	0...1	1	-
1 = delay measurement active, 0 = inactive; inlet					
LV_DLY_MES_ACT_IVVT_EX_i	V	0...1H	0...1	1	-
1 = delay measurement active, 0 = inactive; exhaust					
CAM_DLY_TMP_IVVT_IN_i	-	0...FFH	60...155.625	0.375	°CRK
Camshaft position at beginning of delay measurement; inlet					
CAM_DLY_TMP_IVVT_EX_i	-	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position at beginning of delay measurement; exhaust					
T_DLY_TMP_MES_IVVT_IN_i	-	0...FFFFH	0...655350	10	ms
Time at beginning of delay measurement; inlet					
T_DLY_TMP_MES_IVVT_EX_i	-	0...FFFFH	0...655350	10	ms
Time at beginning of delay measurement; exhaust					

Input data:

STATE_CTL_IVVT_IN_i	STATE_CTL_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LV_ADJ_END_PREV_IVVT_IN_i	LV_ADJ_END_PREV_IVVT_EX_i	T_DLY_MES_IVVT	
LV_ADJ_RTD_IVVT_IN_i	LV_ADJ_RTD_IVVT_EX_i		
LV_ADJ_CHG_DIR_IVVT_IN_i	LV_ADJ_CHG_DIR_IVVT_EX_i		
STATE_PWM_IVVT_IN_i	STATE_PWM_IVVT_EX_i		
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i		
CAM_ADJ_MAX_IVVT_IN	CAM_ADJ_MAX_IVVT_EX		
LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

In order to check the calibration data for the calculation of the IVVT delay there is a possibility to measure the current delay (see LC_ENA_DLY_MES_IVVT in "IVVT Scheduler").

Description:

The camshaft must not be close to one of the stop positions. The system must be in the holding position long enough before the measurement can start. Long enough means that STATE_CTL_IVVT_IN(_EX)_i reaches "HLD_ADAP". If the camshaft adjustement is bigger than C_CAM_DIF_MIN_DLY_IVVT after the energization start the delay has passed.

In dependence on the actual reccurence there can be some inaccuracies in the measurement of the delay especially at low engine speed.

Application conditions:

Initialization:

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

At every engine state

Formula section:

if LV_ACT_IND_IVVT_IN(_EX)_i = 0

or

Inlet:

(CAM_AV_IVVT_IN_i >= C_CAM_INI_IN - C_CAM_HYS_DLY_IVVT

and LV_ADJ_RTD_IVVT_IN_i = 1)

or

(CAM_AV_IVVT_IN_i <=

CAM_ADJ_MAX_IVVT_IN + C_CAM_HYS_DLY_IVVT

and LV_ADJ_RTD_IVVT_IN_i = 0)

Exhaust:

(CAM_AV_IVVT_EX_i <= C_CAM_INI_EX + C_CAM_HYS_DLY_IVVT

and LV_ADJ_RTD_IVVT_EX_i = 0)

or

(CAM_AV_IVVT_EX_i >=

CAM_ADJ_MAX_IVVT_EX - C_CAM_HYS_DLY_IVVT

and LV_ADJ_RTD_IVVT_EX_i = 1)

then *Conditions for delay measurement not fulfilled*

LV_DLY_MES_ACT_IVVT_IN(_EX)_i = 0

else

Delay measurement initialization

Delay measurement

endif

Delay measurement initialization:

if STATE_CTL_IVVT_IN(_EX)_i(n-1) = "HLD_ADAP"
and STATE_CTL_IVVT_IN(_EX)_i(n) = "ADJ"
then
LV_DLY_MES_ACT_IVVT_IN(_EX)_i = 1
CAM_DLY_TMP_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i
T_DLY_TMP_MES_IVVT_IN(_EX)_i = T_DLY_MES_IVVT
endif

Delay measurement:

if LV_DLY_MES_ACT_IVVT_IN(_EX)_i = 1
and STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
and LV_ADJ_CHG_DIR_IVVT_IN(_EX)_i = 0
and LV_ADJ_END_PREV_IVVT_IN(_EX)_i = 0
and |CAM_AV_IVVT_IN(_EX)_i - CAM_DLY_TMP_IVVT_IN(_EX)_i| >= C_CAM_DIF_MIN_DLY_IVVT
then
if STATE_PWM_IVVT_IN(_EX)_i = "RTD"
then
DLY_MES_RTD_IVVT_IN(_EX)_i =
T_DLY_MES_IVVT - T_DLY_TMP_MES_IVVT_IN(_EX)_i
else
DLY_MES_ADC_IVVT_IN(_EX)_i =
T_DLY_MES_IVVT - T_DLY_TMP_MES_IVVT_IN(_EX)_i
endif
LV_DLY_MES_ACT_IVVT_IN(_EX)_i = 0
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_DLY_IVVT	1	0...A0H	0...60	0.375	°CRK
Area close to both stop positions where delay measurement is not enabled					
C_CAM_DIF_MIN_DLY_IVVT	1	0...A0H	0...60	0.375	°CRK
Minimal camshaft position adjustment for delay measurement					

9.62 IVVT Coil Substitute Temperature

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
TMAG_IVVT	O/V	0...C8H	-40...160	1	°C
Coil substitute temperature (intake)					
TMAG_IVVT_EX	O/V	0...C8H	-40...160	1	°C
Coil substitute temperature (exhaust)					
TMAG_IVVT_OFS_EX	O/V	0...C8H	-40...160	1	°C
Coil substitute temperature offset between intake and exhaust					
LV_CHG_TMAG_IVVT_IN	O/V	0...1H	0...1	1	-
1 = break point change in TMAG_IVVT list of data points for IP_IVVTHPWM, 0 = no change (intake)					
LV_CHG_TMAG_IVVT_EX	O/V	0...1H	0...1	1	-
1 = break point change in TMAG_IVVT list of data points for IP_IVVTHPWM, 0 = no change (exhaust)					

Input data:

TOIL	TCO	IP_IVVTHPWM	IP_IVVTHPWM_EX
TEG_DYN_UP_CAT	VS	CONF_CAM_VVT_EX	

FUNCTION DESCRIPTION:

General information:

The solenoid valve characteristics depends on the coil temperature. Especially, this temperature affects the holding pulse width modulation. The substitute coil temperature is calculated from maps based on the oil and coolant temperature.

In order to take into account the different mounting positions between the intake OCV and exhaust OCV, for the exhaust coil temperature an offset is added depending on modeled exhaust gas temperature and vehicle speed to describe the heat transfer from the exhaust line to the exhaust coil. For calibration purpose the Coil Temperature Offset of the Exhaust side can be adapted manually.

Description:

The modeled OCV coil temperature TMAG_IVVT(_EX) is used as input for the holding PWM IP_IVVTHPWM(_EX).

The information about a change of the TMAG_IVVT(_EX) breakpoint is indicated by LV_CHG_TMAG_IVVT_IN/EX, and used for the holding PWM adaptation algorithm.

Application conditions:

Initialization: *At reset:* *TMAG_IVVT = IP_TMAG_IVVT*

TMAG_IVVT_OFS_EX = 0

LV_CHG_TMAG_IVVT_IN/EX = 0

If CONF_CAM_VVT_EX = 1

then TMAG_IVVT_EX = IP_TMAG_IVVT

else TMAG_IVVT_EX = 0 H

endif

Recurrence: 1 s

Activation: At every engine state

Formula section:

Coil Temperature Calculation:

OCV Coil Temperature (Intake):

TMAG_IVVT = IP_TMAG_IVVT *(depending on TOIL and TCO)*

OCV Coil Temperature (Exhaust):

If CONF_CAM_VVT_EX = 1

then exhaust camshaft with adjustment

If LC_TMAG_IVVT_OFS_EX_MAN = 1

then *Manual Adjustment of TMAG_IVVT_EX*

TMAG_IVVT_OFS_EX = IP_TMAG_IVVT_OFS_EX_MAN

TMAG_IVVT_EX = TMAG_IVVT + TMAG_IVVT_OFS_EX

else *Offset for Exhaust OCV Coil Temperature depending on TEG_DYN_CAT_UP*

TMAG_IVVT_OFS_EX = IP_TMAG_IVVT_OFS_TEG

TMAG_IVVT_EX is filtered with weighting factor depending on vehicle speed

$$TMAG\_IVVT\_EX_{(n)} = TMAG\_IVVT\_EX_{(n-1)} * (1 - IP\_CRLC\_TMAG\_IVVT\_VS) + (TMAG\_IVVT + TMAG\_IVVT\_OFS\_EX) * IP\_CRLC\_TMAG\_IVVT\_VS$$

endif

endif

Break point change in TMAG_IVVT(_EX) list of data points for IP_IVVTHPWM(_EX):

if Occurence of break point change in TMAG_IVVT(_EX) list of data points for IP_IVVTHPWM(_EX)

then LV_CHG_TMAG_IVVT_IN/EX = 1

else LV_CHG_TMAG_IVVT_IN/EX = 0

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_TMAG_IVVT	10x10	0...C8H	-40...160	1	°C
LDP_TOIL_IP_TMAG_IVVT	10	0...C8H	-40...160	1	°C
LDP_TCO_IP_TMAG_IVVT	10	0...FEH	-48...142.5	0.75	°C
OCV Coil Temperature (Intake)					
IP_TMAG_IVVT_OFS_TEG	6	0...C8H	-40...160	1	°C
LDP_TEG_DYN_UP_CAT_IP_TMAG_OFS	6	0…7FFFH	-273.15…1775	0.0625	°C
Offset on OCV Coil Temperature for Exhaust					
IP_CRLC_TMAG_IVVT_VS	6	0…FFH	0…0.996	0.004	-
LDP_VS_IP_CRLC_TMAG_IVVT_VS	6	0…FFH	0…255	1	km/h
Weighting Factor on Exhaust Offset for OCV Coil Temperature					
LC_TMAG_IVVT_OFS_EX_MAN	1	0...1H	0...1	1	-
Logical Variable for Enabling Manual Offset for Coil Temperature IVVT_EX					
IP_TMAG_IVVT_OFS_EX_MAN	6	0...C8H	-40...160	1	°C
LDP_TOIL_IP_TMAG_IVVT_MAN	6	0...C8H	-40...160	1	°C
Manual Offset for Coil Temperature IVVT_EX					

9.63 IVVT Holding PWM

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
IVVTHPWM_IN_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Holding pulse width modulation; inlet					
IVVTHPWM_EX_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Holding pulse width modulation; exhaust					
IVVTHPWM_AD_DISP_IN_i	V	0…FFFFH	-50...49.99847	0.001526	%
Current value from adaptation table for holding pulse width modulation; inlet					
IVVTHPWM_AD_DISP_EX _i	V	0…FFFFH	-50...49.99847	0.001526	%
Current value from adaptation table for holding pulse width modulation; exhaust					

Input data:

C_CAM_SP_HYS_NEUT_IVVT_IN	IVVTPWM_RTD_IN	IVVTHPWM_DE_AD_IN_i	TMAG_IVVT
C_CAM_SP_HYS_NEUT_IVVT_EX	IVVTPWM_ADC_EX	IVVTHPWM_DE_AD_EX_i	C_CAM_INI_EX
IVVTHPWM_AD_IN_i[12]	IVVTHPWM_AD_EX_i[12]	IVVTPWM_SUB_IN_i	FRQ_IVVTHPWM
CAM_SP_IND_IVVT_IN_i	CAM_SP_IND_IVVT_EX_i	IVVTPWM_SUB_EX_i	VB
LV_ACT_CTL_IVVT_IN_i	LV_ACT_CTL_IVVT_EX_i	IVVTPWM_CLE_SLV_IN	C_CAM_INI_IN
LV_IVVTPWM_SUB_IN_i	LV_IVVTPWM_SUB_EX_i	IVVTPWM_CLE_SLV_EX	TMAG_IVVT_EX
LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	LV_ACT_CLE_SLV_IVVT	LV_IVVTPWM

Import actions:

ACTION_INFR_SetHoldIvvt(IN < Ivvt >, IN < Ivvthpwm >, IN < Frq_Ivvthpwm >)
This action sets a holding pulse width modulation energisation for the camshaft Ivvt.

FUNCTION DESCRIPTION:

General information:

The holding pulse width modulation signal must be distinguished for the inactive and active IVVT state.

Description:

If the state is inactive small pulse width is needed in order to reach and hold the reference position. If the IVVT state is active the controller determines when the current camshaft position is to be held. The corresponding holding PWM IP_IVVTHPWM(_EX) is a characteristic of the solenoid valve. It depends on the coil temperature and battery voltage.

Due to several reasons, e.g., component tolerances, inaccuracy in the coil temperature determination, the holding pulse width modulation is adapted. The adaptation contribution IVVTHPWM_AD_IN(_EX)_i[…] is always added to the basic value IP_IVVTHPWM(_EX).

If the adaptation is in progress IVVTHPWM_DE_AD_IN/EX_i is non-zero, otherwise it is zero.

It is possible to offset the holding pulse width modulation for each camshaft by means of C_IVVTHPWM_OFS_IN(_EX)_i.

It is possible to apply substitute solenoid valve energization for each individual camshaft, e.g., manually or externally.

If the IVVT system is inactive the PWM signal for the passive stop position can be replaced by the oil control solenoid valve cleaning signal IVVTPWM_CLE_SLV_IN(_EX). Active cleaning is indicated by LV_ACT_CLE_SLV_IVVT = 1.

Application conditions:

Initialization: At reset:

IVVTHPWM_IN_i = IVVTPWM_RTD_IN

IVVTHPWM_EX_i = IVVTPWM_ADC_EX

xx = IN(_EX)_i

yy = IVVT_IN(_EX)_i

ACTION_INFR_SetHoldIvvt(
IN yy,
IN IVVTHPWM_xx,
IN FRQ_IVVTHPWM)

Recurrence: 10 ms or at each used camshaft edge, see "IVVT Scheduler"

Activation: At every engine state

Formula section:

if LV_IVVTPWM = 0

then *Power for solenoid valve energization not yet available*

IVVTHPWM_IN(_EX)_i = 0 %

elseif LV_IVVTPWM_SUB_IN(_EX)_i = 1

then *Substitute solenoid valve energization*

IVVTHPWM_IN(_EX)_i = IVVTPWM_SUB_IN(_EX)_i

elseif LV_ACT_IND_IVVT_IN(_EX)_i = 0

then *IVVT inactive*

if LV_ACT_CLE_SLV_IVVT = 0

then *Energization for reference stop position*

IVVTHPWM_IN_i = IVVTPWM_RTD_IN

IVVTHPWM_EX_i = IVVTPWM_ADC_EX

else *Oil control valve cleaning active*

IVVTHPWM_IN(_EX)_i = IVVTPWM_CLE_SLV_IN(_EX)

endif

else *IVVT active*

Inlet:

if CAM_SP_IND_IVVT_IN_i >

C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

or LV_ACT_CTL_IVVT_IN_i = 0

Exhaust:

if CAM_SP_IND_IVVT_EX_i <
C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX
or LV_ACT_CTL_IVVT_EX_i = 0

then *Setpoint close to reference position or controller inactive*

IVVTHPWM_IN_i = IVVTPWM_RTD_IN

IVVTHPWM_EX_i = IVVTPWM_ADC_EX

else

IVVTHPWM_IN(_EX)_i =

IP_IVVTHPWM(_EX)	+ *basic HPWM*
IVVTHPWM_AD_IN(_EX)_i[...]$_{(index)}$	+ *adaptation contribution*
IVVTHPWM_DE_AD_IN(_EX)_i	+ *from progressing deviation adapt.*
C_IVVTHPWM_OFS_IN(_EX)_i	*offset*

endif

(index) : IVVTHPWM_AD_IN(_EX) table break point for current TMAG_IVVT(_EX)

Current adaptation contribution:

IVVTHPWM_AD_DISP_IN(_EX)_i =

current value from adaptation table IVVTHPWM_AD_IN(_EX)_i[…];

endif

xx = IN(_EX)_i

yy = IVVT_IN(_EX)_i

```
ACTION_INFR_SetHoldIvvt(
IN      yy,
IN      IVVTHPWM_xx,
IN      FRQ_IVVTHPWM)
```

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_IVVTHPWM	12x4	0...FFFFH	0...99.99847	0.001526	%
LDP_TMAG_IVVT_1_VVTI	12	0...C8H	-40...160	1	°C
LDPM_VB_1_VVTI	4	0...FFH	0...26	0.1	V
Holding pulse width modulation					
IP_IVVTHPWM_EX	12x4	0...FFFFH	0...99.99847	0.001526	%
LDP_TMAG_IVVT_EX_1_VVTI	12	0...C8H	-40...160	1	°C
LDPM_VB_1_VVTI	4	0...FFH	0...26	0.1	V
Holding pulse width modulation; exhaust					
C_IVVTHPWM_OFS_IN_i	1	80...7FH	-50...49.6	0.39	%
Holding pulse width modulation offset; inlet					
C_IVVTHPWM_OFS_EX_i	1	80...7FH	-50...49.6	0.39	%
Holding pulse width modulation offset; exhaust					

9.64 IVVT PWM Frequencies

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
FRQ_IVVTHPWM	O/V	32H...1F4H	50...500	1	Hz
Frequency of holding pulse width modulation; both inlet and exhaust					
FRQ_IVVTPWM	O/V	32H...1F4H	50...500	1	Hz
Frequency of pulse width modulation for adjustment energization; both inlet and exhaust					

Input data:

VB			

FUNCTION DESCRIPTION:

General information:

The frequencies of IVVT energization signals are calculated here.

Description:

The frequency of the pulse width modulations is a function of the battery voltage.

Application conditions:

Initialization: At reset:

FRQ_IVVTHPWM = IP_FRQ_IVVTHPWM

FRQ_IVVTPWM = IP_FRQ_IVVTPWM

Recurrence: 100 ms

Activation: In every engine state

Formula section:

FRQ_IVVTHPWM = IP_FRQ_IVVTHPWM

FRQ_IVVTPWM = IP_FRQ_IVVTPWM

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_FRQ_IVVTHPWM	4	32H...1F4H	50...500	1	Hz
LDPM_VB_1_VVTI	4	0...FFH	0...26	0.1	V
Frequency of holding pulse width modulation; both inlet and exhaust					
IP_FRQ_IVVTPWM	4	32H...1F4H	50...500	1	Hz
LDPM_VB_1_VVTI	4	0...FFH	0..26	0.1	V
Frequency of pulse width modulation for adjustment energization; both inlet and exhaust					

9.65 IVVT PWM Adjustment Levels

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
FAC_COR_IVVTPWM_IN	O/V	0...FFH	0...1.9922	0.0078	-
Correction factor for adjustment pulse width modulation (intake)					
FAC_COR_IVVTPWM_EX	O/V	0...FFH	0...1.9922	0.0078	-
Correction factor for adjustment pulse width modulation (exhaust)					
IVVTPWM_RTD_IN	O/V	0...FFH	0...99.6	0.39	%
Current pulse width modulation for retard adjustment; inlet					
IVVTPWM_ADC_IN	O/V	0...FFH	0...99.6	0.39	%
Current pulse width modulation for advanced adjustment; inlet					
IVVTPWM_ADC_EX	O/V	0...FFH	0...99.6	0.39	%
Current pulse width modulation for advanced adjustment; exhaust					
IVVTPWM_RTD_EX	O/V	0...FFH	0...99.6	0.39	%
Current pulse width modulation for retard adjustment; exhaust					

Input data:

VB	TMAG_IVVT	TMAG_IVVT_EX	CONF_CAM_VVT_EX

FUNCTION DESCRIPTION:

General information:

The levels of adjustment energizations are calculated here.

Description:

The functional level of the adjustment pulse width modulation is defined by C_IVVTPWM_ADC/RTD_IN/EX. There is the factor IP_FAC_VB_IVVTPWM for the battery voltage influence and IP_FAC_TMAG_IVVTPWM_IN(_EX) for solenoid temperature influence. The resulting correction factor is the product of the individual factors.

Application conditions:

Initialization: At reset:

FAC_COR_IVVTPWM_IN(_EX) = IP_FAC_VB_IVVTPWM *
IP_FAC_TMAG_IVVTPWM_IN(_EX)

IVVTPWM_ADC(_RTD)_IN = C_IVVTPWM_ADC(_RTD)_IN

If CONF_CAM_VVT_EX = 1

then IVVTPWM_ADC(_RTD)_EX = C_IVVTPWM_ADC(_RTD)_EX

else IVVTPWM_ADC(_RTD)_EX = 0 [%]

Recurrence: 100 ms

Activation: In every engine state

Formula section:

FAC_COR_IVVTPWM_IN(_EX) = IP_FAC_VB_IVVTPWM *
IP_FAC_TMAG_IVVTPWM_IN(_EX)

IVVTPWM_ADC(_RTD)_IN = C_IVVTPWM_ADC(_RTD)_IN *
FAC_COR_IVVTPWM_IN

If CONF_CAM_VVT_EX = 1

then IVVTPWM_ADC(_RTD)_EX = C_IVVTPWM_ADC(_RTD)_EX *
FAC_COR_IVVTPWM_EX

else IVVTPWM_ADC(_RTD)_EX = 0 [%]

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_IVVTPWM_ADC_IN	1	0...FFH	0...99.6	0.39	%
Pulse width modulation for adjustment towards advanced position; inlet					
C_IVVTPWM_ADC_EX	1	0...FFH	0...99.6	0.39	%
Pulse width modulation for adjustment towards advanced position; exhaust					
C_IVVTPWM_RTD_IN	1	0...FFH	0...99.6	0.39	%
Pulse width modulation for adjustment towards retard position; inlet					
C_IVVTPWM_RTD_EX	1	0...FFH	0...99.6	0.39	%
Pulse width modulation for adjustment towards retard position; exhaust					
IP_FAC_VB_IVVTPWM	4	0...FFH	0...1.922	0.0078	-
LDPM_VB_1_VVTI	4	0...FFH	0...26	0.1	V
Battery voltage correction of adjustment pulse width modulation					
IP_FAC_TMAG_IVVTPWM_IN	4	0...FFH	0...1.922	0.0078	-
LDPM_TMAG_IVVT_2_VVTI	4	0...C8H	-40...160	1	°C
Solenoid temperature correction of adjustment pulse width modulation (intake)					
IP_FAC_TMAG_IVVTPWM_EX	4	0...FFH	0...1.922	0.0078	-
LDPM_TMAG_IVVT_EX_2_VVTI	4	0...C8H	-40...160	1	°C
Solenoid temperature correction of adjustment pulse width modulation (exhaust)					

9.66 IVVT Deviation Adaptation of Holding PWM

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_DE_AD_REQ_IVVT_IN_i	V	0...1H	0...1	1	-
1 = request for deviation adaptation of HPWM, 0 = no request; inlet					
LV_DE_AD_REQ_IVVT_EX_i	V	0...1H	0...1	1	-
1 = request for deviation adaptation of HPWM, 0 = no request; exhaust					
LV_DE_AD_END_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = end of deviation adaptation of HPWM; 0 = active; inlet					
LV_DE_AD_END_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = end of deviation adaptation of HPWM; 0 = active; exhaust					
CAM_DIF_FIL_DE_AD_IVVT_IN_i	V	B000H...5000H	-59.997...60	0.00293	°CRK
Filtered camshaft deviation for HPWM deviation adaptation; inlet					
CAM_DIF_FIL_DE_AD_IVVT_EX_i	V	B000H...5000H	-59.997...60	0.00293	°CRK
Filtered camshaft deviation for HPWM deviation adaptation; exhaust					
IVVTHPWM_DE_AD_IN_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current value of deviation adaptation to be added to holding pulse width modulation; inlet					
IVVTHPWM_DE_AD_EX_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current value of deviation adaptation to be added to holding pulse width modulation; exhaust					
IVVTHPWM_DE_AD_SAVE_IN_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current result of deviation adaptation of holding pulse width modulation; inlet					
IVVTHPWM_DE_AD_SAVE_EX_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current result of deviation adaptation of holding pulse width modulation; exhaust					
CAM_DIF_INT_DE_AD_IVVT_IN_i	V	8000H...7FFFH	-100.663296 …100.660224	0.003072	°CRK*s
Camshaft position deviation integrator for restarting deviation adaptation of HPWM; inlet					
CAM_DIF_INT_DE_AD_IVVT_EX_i	V	8000H...7FFFH	-100.663296 …100.660224	0.003072	°CRK*s
Camshaft position deviation integrator for restarting deviation adaptation of HPWM; exhaust					
CTR_CYC_DE_AD_IVVT_IN_i	V	0...1770H	0...6000	1	-
Counter of cycles of deviation adaptation of HPWM; inlet					
CTR_CYC_DE_AD_IVVT_EX_i	V	0...1770H	0...6000	1	-
Counter of cycles of deviation adaptation of HPWM; exhaust					
CTR_CYC_RAMP_DE_AD_IVVT_IN_i	V	0...FFH	0...255	1	-
Number of cycles between two HPWM adaptation steps; inlet					
CTR_CYC_RAMP_DE_AD_IVVT_EX_i	V	0...FFH	0...255	1	-
Number of cycles between two HPWM adaptation steps; exhaust					

Input data:

STATE_PWM_IVVT_IN_i	STATE_PWM_IVVT_EX_i	LV_CHG_TMAG_IVVT_IN	TOIL
T_DIF_EDGE_CAM_IVVT_IN_i	CAM_SP_DIF_IVVT_IN_i	LV_VLD_PSN_CAM_IN_i	C_CAM_INI_EX
T_DIF_EDGE_CAM_IVVT_EX_i	CAM_SP_DIF_IVVT_EX_i	LV_VLD_PSN_CAM_EX_i	N_32
LV_CAM_SP_DYW_IVVT_IN_i	CAM_SP_IND_IVVT_IN_i	CAM_DIF_IVVT_EX_i	C_CAM_INI_IN
LV_CAM_SP_DYW_IVVT_EX_i	CAM_SP_IND_IVVT_EX_i	CAM_DIF_IVVT_IN_i	
C_CAM_SP_HYS_NEUT_IVVT_IN	LV_ADJ_MAX_IVVT_IN_i	LV_ACT_IND_IVVT_IN_i	
C_CAM_SP_HYS_NEUT_IVVT_EX	LV_ADJ_MAX_IVVT_EX_i	LV_ACT_IND_IVVT_EX_i	
LV_CHG_TMAG_IVVT_EX			

FUNCTION DESCRIPTION:

General information:

As obvious from its name there should be no actuator movement if it is energized by the holding pulse width modulation. If it is not fulfilled, e.g. due to component tolerances, ageing effects, etc., deviations of the camshaft position from the setpoint can occur. The deviation adaptation determines a correction leading to proper holding pulse width modulation.

Description:

Let us assume the following situation: The setpoint remains almost the same after its change. It is indicated by LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 1. The controller needs some time to get the camshaft position to the setpoint. If there is a deviation from the setpoint after this time, an improper value in the map IP_IVVTHPWM is assumed. Possible reasons are the solenoid valve tolerances, deviations of the actual temperature of the solenoid valve from the modeled one, ageing effects, etc.

The adaptation is requested formally if there is a change of the setpoint that lead to a control adjustment. After the change, if the conditions for the adaptations are fulfilled the adaptation is started.

The first phase of the adaptation is waiting for the controller fading away. It is counted in the adaptation cycles. One adaptation cycle is the period between two used camshaft edges. The controller fading away IP_CYC_MIN_DE_AD_IVVT must be over at first. Short before this event (IP_CYC_MIN_DE_AD_IVVT * C_FAC_RED_CYC_MIN_DE_AD_IVVT) a filtering of the camshaft deviation starts. The camshaft position deviation (absolute value) must not be bigger than C_CAM_HYS_DE_AD_IVVT. Otherweise, a failure is assumed in this case. The adaptation is terminated and a diagnosis (for enduring position deviation) is enabled. Vice versa, the adaptation is disabled while the adaptation is active.

The priciple of the adaptation is as follows. The holding pulse width modulation is being changed by means of IVVTHPWM_DE_AD_IN(_EX)_i. This quantity is increased or decreased so that the deviation is diminishing. If the deviation becomes smaller than C_CAM_HYS_DE_AD_END_IVVT or the position crosses the setpoint the adaptation is finished successfully. The resulting contribution IVVTHPWM_DE_AD_IN(_EX)_i to the holding pulse width modulation is limited to C_IVVTHPWM_HYS_DE_AD and assigned into IVVTHPWM_DE_AD_SAVE_IN(_EX)_i. This quantity is then saved into an adaptation table, see "IVVT Holding PWM Adaptation Manager".

If the engine operation remains steady after the adaptation the camshaft position deviation is integrated, CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i. If this quantity reaches the threshold C_CAM_DIF_INT_DE_AD_REST_IVVT the adaptation is restarted. This restarting must be enabled by LC_ENA_DE_AD_REST_IVVT. Another reason for the adaptation request is a change of the TMAG_IVVT breaking point in the map IP_IVVTHPWM.

If the maximum number of the adaptation cycles is reached the adaptation is terminated without any saving into the adaptation table.

No adaptation is activated if the oil temperature is too high and the engine speed too low. The oil pressure might not be sufficient for the adaptation. The adaptation results might not be reliable.

Application conditions:

Initialization:

At reset: LV_DE_AD_REQ_IVVT_IN(_EX)_i = 1
LV_DE_AD_END_IVVT_IN(_EX)_i = 1

At deactivation: LV_DE_AD_REQ_IVVT_IN(_EX)_i = 1
LV_DE_AD_END_IVVT_IN(_EX)_i = 1
IVVTHPWM_DE_AD_IN(_EX)_i = 0
CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i = 0

Recurrence: At each used camshaft edge, see "IVVT Scheduler"

Activation: LV_ACT_IND_IVVT_IN(_EX)_i = 1
and (TOIL <= C_TOIL_MAX_IVVTHPWM_AD
or N_32 >= C_N_32_MIN_IVVTHPWM_AD)

Formula section:

Deviation adaptation request

Conditions for deviation adaptation

if LV_DE_AD_END_IVVT_IN(_EX)_i = 0
then *Conditions for adaptation fulfilled*
if CTR_CYC_DE_AD_IVVT_IN(_EX)_i <=
IP_CYC_MIN_DE_AD_IVVT * C_FAC_RED_CYC_MIN_DE_AD_IVVT
then CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i = CAM_DIF_IVVT_IN(_EX)_i
else *Filtering of CAM_DIF_IVVT_IN(_EX)_i from now*
Filtering of camshaft deviation
endif
if CTR_CYC_DE_AD_IVVT_IN(_EX)_i < IP_CYC_MIN_DE_AD_IVVT
then *Wait for camshaft position control fade away*
There is no action
elseif CTR_CYC_DE_AD_IVVT_IN(_EX)_i >
IP_CYC_MIN_DE_AD_IVVT + C_CYC_MAX_DE_AD_IVVT
or |CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i| >=
C_CAM_HYS_DE_AD_IVVT
then Deviation too big or maximum number of cycles for adaptation reached
elseif |CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i| <=
C_CAM_HYS_DE_AD_END_IVVT
or CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i(n-1) *
CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i(n) <= 0

then Fulfillment of adaptation target
elseif LV_VLD_PSN_CAM_IN(_EX)_i <> 0
then Adaptation algorithm
endif
CTR_CYC_DE_AD_IVVT_IN(_EX)_i(n) =
CTR_CYC_DE_AD_IVVT_IN(_EX)_i(n-1) + 1
endif

Deviation adaptation request:

if LV_DE_AD_REQ_IVVT_IN(_EX)_i = 0
and [
(|CAM_SP_DIF_IVVT_IN(_EX)_i| > C_CAM_HYS_RST_DE_AD_IVVT
and STATE_PWM_IVVT_IN(_EX)_i <> "HLD")
or (LV_CHG_TMAG_IVVT_IN(_EX)(n-1) = 0
and LV_CHG_TMAG_IVVT_IN(_EX)(n) = 1)
or (abs(CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i) >=
C_CAM_DIF_INT_DE_AD_REST_IVVT
and LC_ENA_DE_AD_REST_IVVT = 1)
]
then LV_DE_AD_REQ_IVVT_IN(_EX)_i = 1
endif

Conditions for deviation adaptation:

if$_1$ LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 1
and LV_ADJ_MAX_IVVT_IN(_EX)_i = 0
and Inlet:
(CAM_SP_IND_IVVT_IN_i <=
C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN)
Exhaust:
(CAM_SP_IND_IVVT_EX_i >=
C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX)
then$_1$ *Setpoint: limited dynamics; deviation not too big; not close to reference position*
if$_2$ LV_DE_AD_REQ_IVVT_IN(_EX)_i = 1
then$_2$ *Adaptation required*
if$_3$ LV_DE_AD_END_IVVT_IN(_EX)_i = 1
then$_3$ *Kick-off of adaptation*
LV_DE_AD_END_IVVT_IN(_EX)_i = 0
CTR_CYC_DE_AD_IVVT_IN(_EX)_i = 0

IVVTHPWM_DE_AD_IN(_EX)_i = 0

IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 0

CTR_CYC_RAMP_DE_AD_IVVT_IN(_EX)_i = 0

CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i = 0

endif$_3$

else$_2$ *Conditions for adaptation fulfilled; adaptation not required, integrate deviation for potential restarting*

LV_DE_AD_END_IVVT_IN(_EX)_i = 1

IVVTHPWM_DE_AD_IN(_EX)_i = 0

if$_4$ LC_ENA_DE_AD_REST_IVVT = 1

then$_4$ CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i(n) =

CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i(n-1) +

CAM_DIF_IVVT_IN(_EX)_i *

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

endif$_4$

endif$_2$

else$_1$ *Conditions for adaptation not fulfilled*

LV_DE_AD_END_IVVT_IN(_EX)_i = 1

IVVTHPWM_DE_AD_IN(_EX)_i = 0

CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i = 0

endif$_1$

<u>Filtering of camshaft deviation:</u>

CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i(n) =

CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i(n-1) +

C_CRLC_CAM_DIF_FIL_DE_AD_IVVT *

(CAM_DIF_IVVT_IN(_EX)_i - CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i(n-1))

<u>Deviation too big or maximum number of cycles for adaptation reached:</u>

LV_DE_AD_REQ_IVVT_IN(_EX)_i = 0

LV_DE_AD_END_IVVT_IN(_EX)_i = 1

IVVTHPWM_DE_AD_IN(_EX)_i = 0

CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i = 0

Fulfillment of adaptation target:

IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = IVVTHPWM_DE_AD_IN(_EX)_i

if |IVVTHPWM_DE_AD_SAVE_IN(_EX)_i| > C_IVVTHPWM_HYS_DE_AD

then IVVTHPWM_DE_AD_SAVE_IN(_EX)_i =
sign(IVVTHPWM_DE_AD_SAVE_IN(_EX)_i) *
C_IVVTHPWM_HYS_DE_AD

endif

LV_DE_AD_REQ_IVVT_IN(_EX)_i = 0

LV_DE_AD_END_IVVT_IN(_EX)_i = 1

IVVTHPWM_DE_AD_IN(_EX)_i = 0

CAM_DIF_INT_DE_AD_IVVT_IN(_EX)_i = 0

Adaptation algorithm

if$_1$ CTR_CYC_RAMP_DE_AD_IVVT_IN(_EX)_i >=
IP_CYC_RAMP_DE_AD_IVVT_IN(_EX)
Use abs(CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i) for
IP_CYC_RAMP_DE_AD_IVVT_IN(_EX)

then$_1$ *Carry out adaptation step*
Use abs(CAM_DIF_FIL_DE_AD_IVVT_IN(_EX)_i) for
IP_IVVTHPWM_CHG_DE_AD

if$_2$ Inlet: CAM_DIF_FIL_DE_AD_IVVT_IN_i > 0

then$_2$ IVVTHPWM_DE_AD_IN_i(n) = IVVTHPWM_DE_AD_IN_i(n-1) +
IP_IVVTHPWM_CHG_DE_AD

else$_2$ IVVTHPWM_DE_AD_IN_i(n) = IVVTHPWM_DE_AD_IN_i(n-1) -
IP_IVVTHPWM_CHG_DE_AD

endif$_2$

if$_3$ Exhaust: CAM_DIF_FIL_DE_AD_IVVT_EX_i < 0

then$_3$ IVVTHPWM_DE_AD_EX_i(n) = IVVTHPWM_DE_AD_EX_i(n-1) +
IP_IVVTHPWM_CHG_DE_AD

else$_3$ IVVTHPWM_DE_AD_EX_i(n) = IVVTHPWM_DE_AD_EX_i(n-1) -
IP_IVVTHPWM_CHG_DE_AD

endif$_3$

CTR_CYC_RAMP_DE_AD_IVVT_IN(_EX)_i = 0

else$_1$ *Count cycles between two HPWM changes*
CTR_CYC_RAMP_DE_AD_IVVT_IN(_EX)_i(n) =
CTR_CYC_RAMP_DE_AD_IVVT_IN(_EX)_i(n-1) +1

endif$_1$

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_N_32_MIN_IVVTHPWM_AD	1	0...FFH	0...8160	32	rpm
Minimum engine speed for HPWM adaptations					
C_TOIL_MAX_IVVTHPWM_AD	1	0...C8H	-40...160	1	°C
Maximum oil temperature for HPWM adaptations					
LC_ENA_DE_AD_REST_IVVT	1	0...1H	0...1	1	-
1 = restarting deviation adaptation of HPWM enabled, 0 = not enabled					
C_CAM_HYS_RST_DE_AD_IVVT	1	0...A0H	0...60	0.375	°CRK
Hysteresis of camshaft position setpoint difference for reseting deviation adaptation of HPWM					
C_CAM_DIF_INT_DE_AD_REST_IVVT	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Treshold of camshaft position deviation integral for restarting deviation adaptation of HPWM					
C_CAM_HYS_DE_AD_IVVT	1	0...A0H	0...60	0.375	°CRK
Hysteresis of camshaft position deviation for starting deviation adaptation of HPWM					
IP_CYC_MIN_DE_AD_IVVT	4x4	0...1770H	0...6000	1	-
LDP_N_32_IP_CYC_MIN_DE_AD_IVVT	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_2_VVTI	4	0...C8H	-40...160	1	°C
Number of cycles (adaptation cycles) for camshaft position control fading away to start adaptation of HPWM					
C_FAC_RED_CYC_MIN_DE_AD_IVVT	1	0...FFH	0...0.9984	0.0039	-
Factor for reduction of control fading away for starting camshaft deviation filtering					
C_CYC_MAX_DE_AD_IVVT	1	0...1770H	0...6000	1	-
Maximum number of cycles for adaptation					
IP_CYC_RAMP_DE_AD_IVVT_IN	5	0...FFH	0...255	1	-
LDPM_CAM_DIF_FIL_IVVT_1_VVTI	5	0...5000H	0...60	0.00293	°CRK
Number of cycles between two HPWM adaptation steps; inlet					
IP_CYC_RAMP_DE_AD_IVVT_EX	5	0...FFH	0...255	1	-
LDPM_CAM_DIF_FIL_IVVT_1_VVTI	5	0...5000H	0...60	0.00293	°CRK
Number of cycles between two HPWM adaptation steps; exhaust					
IP_IVVTHPWM_CHG_DE_AD	5x4	0...7FFFH	0...49.99847	0.001526	%
LDPM_CAM_DIF_FIL_IVVT_1_VVTI	5	0...5000H	0...60	0.00293	°CRK
LDPM_TOIL_2_VVTI	4	0...C8H	-40...160	1	°C
Step of deviation adaptation (HPWM change)					
C_IVVTHPWM_HYS_DE_AD	1	0...7FFFH	0...49.99847	0.001526	%
Hysteresis for maximal contribution of deviation adaptation					
C_CRLC_CAM_DIF_FIL_DE_AD_IVVT	1	0...FFH	0...0.9984	0.0039	-
Correlation constant for camshaft deviation filtering					
C_CAM_HYS_DE_AD_END_IVVT	1	0...5000H	0...60	0.00293	°CRK
Hysteresis of camshaft position deviation for ending deviation adaptation of HPWM					

9.67 IVVT Drift Adaptation of Holding PWM

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_DRIFT_AD_ACT_IVVT_IN_i	V	0...1H	0...1	1	-
1 = drift adaptation of HPWM active, 0 = inactive; inlet					
LV_DRIFT_AD_ACT_IVVT_EX_i	V	0...1H	0...1	1	-
1 = drift adaptation of HPWM active, 0 = inactive; exhaust					
T_DRIFT_AD_IVVT_IN_i	V	0...FFFFFFFFH	0...17179.87	4e-6	s
Timer for drift calculation; inlet					
T_DRIFT_AD_IVVT_EX_i	V	0...FFFFFFFFH	0...17179.87	4e-6	s
Timer for drift calculation; exhaust					
CAM_DRIFT_AD_ST_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Camshaft position at drift adaptation start; inlet					
CAM_DRIFT_AD_ST_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position at drift adaptation start; exhaust					
CTR_CYC_DRIFT_AD_IVVT_IN_i	V	0...FFH	0...255	1	-
Counter of drift adaptation cycles for drift calculation, overflow not allowed; inlet					
CTR_CYC_DRIFT_AD_IVVT_EX_i	V	0...FFH	0...255	1	-
Counter of drift adaptation cycles for drift calculation, overflow not allowed; exhaust					
VEL_DRIFT_IVVT_IN_i	V	FF01H...FFH	-95.6...95.6	0.375	°CRK/s
Drift velocity; inlet					
VEL_DRIFT_IVVT_EX_i	V	FF01H...FFH	-95.6...95.6	0.375	°CRK/s
Drift velocity; exhaust					
IVVTHPWM_DRIFT_AD_SAVE_IN_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current result of drift adaptation of holding pulse width modulation; inlet					
IVVTHPWM_DRIFT_AD_SAVE_EX_i	O/V	8000H...7FFFH	-50...49.99847	0.001526	%
Current result of drift adaptation of holding pulse width modulation; exhaust					

Input data:

T_DIF_EDGE_CAM_IVVT_IN_i	T_DIF_EDGE_CAM_IVVT_EX_i	TOIL	N_32
STATE_CTL_IVVT_IN_i	STATE_CTL_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LV_DE_AD_END_IVVT_IN_i	LV_DE_AD_END_IVVT_EX_i		
CAM_SP_IND_IVVT_IN_i	CAM_SP_IND_IVVT_EX_i		
LV_ADJ_RTD_IVVT_IN_i	LV_ADJ_RTD_IVVT_EX_i		
CAM_ADJ_MAX_IVVT_IN	CAM_ADJ_MAX_IVVT_EX		
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i		
C_TOIL_MAX_IVVTHPWM_AD	C_N_32_MIN_IVVTHPWM_AD		
LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

The drift adaptation is a fine tuning of the holding pulse width modulation.

Description:

The adaptation procedure is activated if STATE_CTL_IVVT_IN(_EX)_i changes from "HLD_DLY" to "HLD_ADAP". The current camshaft position is saved and a time measurement is started.

Formally, the next step is the drift calculation. The state STATE_CTL_IVVT_IN(_EX)_i is still "HLD_ADAP", also, no calculation is carried out. The time measurement continues, i.e., the adaptation is not aborted, while conditions for the drift adaptation are fulfilled.

The drift velocity is calculated when "HLD_ADAP" has been held for IP_CYC_MIN_DRIFT_AD_IVVT drift adaptation cycles at least and has been left due to an adjustment, i.e., STATE_CTL_IVVT_IN(_EX)_i becomes "ADJ". The holding pulse width modulation correction IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i is calculated from ID_IVVTHPWM_COR_DRIFT_AD dependent on the velocity drift. The sign of the velocity drift determines the sign of the correction so that the drift becomes smaller.

No adaptation is activated if the oil temperature is too high and the engine speed too low. The oil pressure might not be sufficient for the adaptation. The adaptation results might not be reliable.

Application conditions:

Initialization:

At reset:

CAM_DRIFT_AD_ST_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

```
        LV_ACT_IND_IVVT_IN(_EX)_i = 1
and     (       TOIL <= C_TOIL_MAX_IVVTHPWM_AD
        or      N_32 >= C_N_32_MIN_IVVTHPWM_AD)
```

Formula section:

Drift adaptation kick-off:

```
if              LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 0
        and     STATE_CTL_IVVT_IN(_EX)_i(n-1) = "HLD_DLY"
        and     STATE_CTL_IVVT_IN(_EX)_i(n) = "HLD_ADAP"
then
        LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 1
        T_DRIFT_AD_IVVT_IN(_EX)_i = 0
        CAM_DRIFT_AD_ST_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i
        CTR_CYC_DRIFT_AD_IVVT_IN(_EX)_i = 0
        IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 0
endif
```

Drift calculation:

```
if      LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 1
then
        if              STATE_CTL_IVVT_IN(_EX)_i = "ADJ"
```

and CTR_CYC_DRIFT_AD_IVVT_IN(_EX)_i >=
IP_CYC_MIN_DRIFT_AD_IVVT

then

VEL_DRIFT_IVVT_IN(_EX)_i =
(CAM_AV_IVVT_IN(_EX)_i - CAM_DRIFT_AD_ST_IVVT_IN(_EX)_i) /
T_DRIFT_AD_IVVT_IN(_EX)_i

IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i =
ID_IVVTHPWM_COR_DRIFT_AD

Use abs(VEL_DRIFT_IVVT_IN(_EX)_i) for list of data points for
ID_IVVTHPWM_COR_DRIFT_AD

Note: "tmp" is only text auxiliary means; begin of validity

if VEL_DRIFT_IVVT_IN_i > 0 **then** tmp = 1 **endif**
if VEL_DRIFT_IVVT_IN_i < 0 **then** tmp = -1 **endif**
if VEL_DRIFT_IVVT_EX_i > 0 **then** tmp = -1 **endif**
if VEL_DRIFT_IVVT_EX_i < 0 **then** tmp = 1 **endif**

IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i =
IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i * tmp

Note: "tmp"; end of validity

LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 0

endif

endif

<u>Conditions for drift adaptation:</u>

if LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 1

then

if STATE_CTL_IVVT_IN(_EX)_i(n) = "HLD_ADAP"

and LV_DE_AD_END_IVVT_IN(_EX)_i = 1

and <u>Inlet:</u>

CAM_SP_IND_IVVT_IN_i and CAM_AV_IVVT_IN_i are not close to stop positions

CAM_SP_IND_IVVT_IN_i <
C_CAM_INI_IN - C_CAM_HYS_DRIFT_AD_IVVT

and

[(CAM_AV_IVVT_IN_i <=
C_CAM_INI_IN - C_CAM_HYS_DRIFT_AD_IVVT

and LV_ADJ_RTD_IVVT_IN_i = 1)

or

(CAM_AV_IVVT_IN_i >=

CAM_ADJ_MAX_IVVT_IN + C_CAM_HYS_DRIFT_AD_IVVT

and LV_ADJ_RTD_IVVT_IN_i = 0)]

Exhaust:

CAM_SP_IND_IVVT_EX_i and CAM_AV_IVVT_EX_i are not close to stop positions

CAM_SP_IND_IVVT_EX_i >

C_CAM_INI_EX + C_CAM_HYS_DRIFT_AD_IVVT

and

[(CAM_AV_IVVT_EX_i >=

C_CAM_INI_EX + C_CAM_HYS_DRIFT_AD_IVVT

and LV_ADJ_RTD_IVVT_EX_i = 0)

or

(CAM_AV_IVVT_EX_i <=

CAM_ADJ_MAX_IVVT_EX - C_CAM_HYS_DRIFT_AD_IVVT

and LV_ADJ_RTD_IVVT_EX_i = 1)]

then

T_DRIFT_AD_IVVT_IN(_EX)_i(n) =

T_DRIFT_AD_IVVT_IN(_EX)_i(n-1) +

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

CTR_CYC_DRIFT_AD_IVVT_IN(_EX)_i(n) =

CTR_CYC_DRIFT_AD_IVVT_IN(_EX)_i(n-1) + 1

else

LV_DRIFT_AD_ACT_IVVT_IN(_EX)_i = 0

endif

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_DRIFT_AD_IVVT	1	0...A0H	0...60	0.375	°CRK
Hysteresis close to stop positions where drift adaptation of HPWM is not enabled					
IP_CYC_MIN_DRIFT_AD_IVVT	4	1...FFH	1...255	1	-
LDP_N_32_IP_CYC_MIN_DRIFT_IVVT	4	0...FFH	0...8160	32	rpm
Minimum number of drift adaptation cycles for drift calculation					
ID_IVVTHPWM_COR_DRIFT_AD	6	0...7FFFH	0...49.99847	0.001526	%
LDP_VEL_DRIFT_ID_IVVTHPWM_COR	6	0...FFH	0...95.625	0.375	°CRK/s
Correction value for HPWM depending on drift velocity					

9.68 IVVT Holding PWM Adaptation Manager

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
IVVTHPWM_AD_IN_i[12]	O/S/V	0...FFFFH	-50...49.99847	0.001526	%
Table for adapted values for IVVT holding pulse width modulation; inlet					
IVVTHPWM_AD_EX_i[12]	O/S/V	0...FFFFH	-50...49.99847	0.001526	%
Table for adapted values for IVVT holding pulse width modulation; exhaust					
IVVTHPWM_AD_AS_IN_i[12]	V	0...FFFFH	-50...49.99847	0.001526	%
Copy (for application purposes) of table for adapted values for IVVT holding pulse width modulation; inlet					
IVVTHPWM_AD_AS_EX_i[12]	V	0...FFFFH	-50...49.99847	0.001526	%
Copy (for application purposes) of table for adapted values for IVVT holding pulse width modulation; exhaust					
LV_IVVTHPWM_DE_AD_SAVE_IN_i	-	0...1H	0...1	1	-
0 = contribution of HPWM deviation adaptation is to save, 1 = is not to save; inlet					
LV_IVVTHPWM_DE_AD_SAVE_EX_i	-	0...1H	0...1	1	-
0 = contribution of HPWM deviation adaptation is to save, 1 = is not to save; exhaust					
LV_IVVTHPWM_DRIFT_AD_SAVE_IN_i	-	0...1H	0...1	1	-
0 = contribution of HPWM drift adaptation is to save, 1 = is not to save; inlet					
LV_IVVTHPWM_DRIFT_AD_SAVE_EX_i	-	0...1H	0...1	1	-
0 = contribution of HPWM drift adaptation is to save, 1 = is not to save; exhaust					

Input data:

IVVTHPWM_DE_AD_SAVE_IN_i	TMAG_IVVT	TMAG_IVVT_EX	
IVVTHPWM_DE_AD_SAVE_EX_i	IP_IVVTHPWM	IP_IVVTHPWM_EX	
IVVTHPWM_DRIFT_AD_SAVE_IN_i	LV_ACT_IND_IVVT_IN_i		
IVVTHPWM_DRIFT_AD_SAVE_EX_i	LV_ACT_IND_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

The contributions of the two holding pulse width modulation adaptations, i.e., deviation and drift adaptation, are saved into the adaptation table IVVTHPWM_AD_IN(_EX)_i[…] in this manager.

Description:

The properties of the adaptation table IVVTHPWM_AD_IN(_EX)_i[…] are specified by means of the dummy ID_IVVTHPWM_AD_IN(_EX)_i, see "Calibration data". The index of the adaptation map is independent from IP_IVVTHPWM(_EX) The adapted values are saved dependent on the solenoid valve temperature TMAG_IVVT(_EX). The "[...]" part of the indentifier expresses saving into the cell belonging to the current TMAG_IVVT(_EX).

After the adaptation end the variables IVVTHPWM_DE_AD_SAVE_IN(_EX)_i and IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i do not change till the next adaptation kick-off. They are set 0 at this moment, see the adaptation procedures.

The values in the table cannot exceed the limits C_IVVTHPWM_AD_MIN and C_IVVTHPWM_AD_MAX.

Application conditions:

Initialization:

In case of checksum error:

IVVTHPWM_AD_IN(_EX)_i[12] = IVVTHPWM_AD_AS_IN(_EX)_i[12] = 0 %

At activation:

LV_IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 1

LV_IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 1

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_ACT_IND_IVVT_IN(_EX)_i = 1

Formula section:

Saving deviation adaptation contribution:

```
if      IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 0
then
        LV_IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 0
else
        if      LV_IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 0
        then
                IVVTHPWM_AD_IN(_EX)_i[…] = IVVTHPWM_AD_IN(_EX)_i[…] +
                IVVTHPWM_DE_AD_SAVE_IN(_EX)_i
                Limit IVVTHPWM_AD_IN(_EX)_i[…]:
                C_IVVTHPWM_AD_MIN <= IVVTHPWM_AD_IN(_EX)_i[…] <=
                C_IVVTHPWM_AD_MAX
                IVVTHPWM_AD_AS_IN(_EX)_i[…] = IVVTHPWM_AD_IN(_EX)_i[…]
                LV_IVVTHPWM_DE_AD_SAVE_IN(_EX)_i = 1
        endif
endif
```

Saving drift adaptation contribution:

if IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 0
then
LV_IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 0
else
if LV_IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 0
then
IVVTHPWM_AD_IN(_EX)_i[…] = IVVTHPWM_AD_IN(_EX)_i[…] + IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i
Limit IVVTHPWM_AD_IN(_EX)_i[…]:
C_IVVTHPWM_AD_MIN <= IVVTHPWM_AD_IN(_EX)_i[…] <= C_IVVTHPWM_AD_MAX
IVVTHPWM_AD_AS_IN(_EX)_i[…] = IVVTHPWM_AD_IN(_EX)_i[…]
LV_IVVTHPWM_DRIFT_AD_SAVE_IN(_EX)_i = 1
endif
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
ID_IVVTHPWM_AD_IN_i	1x12	0...FFFFH	-50...49.99847	0.001526	%
LDP_TMAG_IVVT_3_VVTI	12	0...C8H	-40...160	1	°C
Dummy ID for specification of properties of adaptation table IVVTHPWM_AD_IN_i[12]; inlet					
ID_IVVTHPWM_AD_EX_i	1x12	0...FFFFH	-50...49.99847	0.001526	%
LDP_TMAG_IVVT_EX_3_VVTI	12	0...C8H	-40...160	1	°C
Dummy ID for specification of properties of adaptation table IVVTHPWM_AD_EX_i[12]; exhaust					
C_IVVTHPWM_AD_MIN	1	0...FFFFH	-50...49.99847	0.001526	%
Minimum for limitation of adapted values from HPWM adaptations					
C_IVVTHPWM_AD_MAX	1	0...FFFFH	-50...49.99847	0.001526	%
Maximum for limitation of adapted values from HPWM adaptations					

9.69 IVVT Oil Control Solenoid Valve Cleaning

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ACT_CLE_SLV_IVVT	V/O	0...1H	0...1	1	-
Cleaning of oil control solenoid valve: 0 = inactive, 1 = active					
IVVTPWM_CLE_SLV_IN	V/O	0...FFH	0...99.6	0.39	%
Pulse width modulation for oil control solenoid valve cleaning (intake)					
IVVTPWM_CLE_SLV_EX	V/O	0...FFH	0...99.6	0.39	%
Pulse width modulation for oil control solenoid valve cleaning (exhaust)					
LV_DLY_CLE_SLV_IVVT	V	0...1H	0...1	1	-
Oil control solenoid valve cleaning delay: 0 = not started, 1 = started					
T_ST_CLE_SLV_IVVT	-	0...FFFFH	0...655350	10	ms
Start time for oil control solenoid valve cleaning time measurements					
CTR_CYC_CLE_SLV_IVVT	V	0...FFH	0...255	1	-
Counter of oil control solenoid valve cleaning cycles					
CTR_STEP_CLE_SLV_IVVT	V	0...FFH	0...255	1	-
Counter of oil control solenoid valve cleaning steps in one cycle					
LV_ACT_PWL_CLE_SLV_IVVT	V	0...1H	0...1	1	-
Cleaning of oil control solenoid valve in power-latch: 0 = inactive, 1 = active					

Input data:

LV_IGK	LV_ES	T_DLY_MES_IVVT	LV_DI_CLE_SLV_IVVT
FAC_COR_IVVTPWM_IN	TOIL	VB	LV_IVVTPWM
C_VB_MIN_IVVT	C_VB_MAX_IVVT	LV_PWL	FAC_COR_IVVTPWM_EX
CONF_CAM_VVT_EX			

Imported actions:

ACTION_FCTM_HoldPostOpPhase (IN <FctName>, IN <TimeOut>)
This action is used to request to hold the post-operating phase
ACTION_FCTM_ReleasePostOpPhase (IN <FctName>)
This action is used to give the authorization to release the post-operating phase

FUNCTION DESCRIPTION:

General information:

Particles in the oil control valve of the IVVT system can disturb the system performance. These debris do not affect the oil control valve operation if they are away from the oil flow control area. This can be achieved by movements of the valve control piston between its stop positions. Such procedure is called oil control solenoid valve cleaning.

Description:

The cleaning is done if the engine is stopped. If it is before the engine start (LV_ES = 1 and LV_IGK = 1) there is the delay C_T_DLY_ST_CLE_SLV_IVVT before cleaning. The start (LV_ES = 1 $\rightarrow$ 0) usually follows the transition LV_IGK = 0 $\rightarrow$ 1 very quickly. The cleaning would be interrupted soon after its beginning. If delay C_T_DLY_ST_CLE_SLV_IVVT with LV_IGK = 1 is reached it is to assume that the driver will wait some time before starting.

The engine is not running in the power latch (LV_ES = 1 and LV_IGK = 0). The oil pressure may be high at the power latch beginning, especially at low oil temperatures. As the actuator might move during the cleaning, there is the delay C_T_DLY_PWL_CLE_SLV_IVVT.

The basic unit of the cleaning is a cycle. The cycle consists of 2 * C_CTR_STEP_CLE_SLV_IVVT steps. The first cycle half has high level pulse width modulation, the second half has low level PWM. There can be C_CTR_CYC_CLE_SLV_IVVT cycles in one driving cycle. The cleaning signal is specified by ID_IVVTPWM_CLE_SLV. It is possible to create a smoothed signal in order to limit possible noise at the cleaning. The solenoid valve temperature and the battery voltage are taken into account by the correction factor FAC_COR_IVVTPWM_IN(_EX).

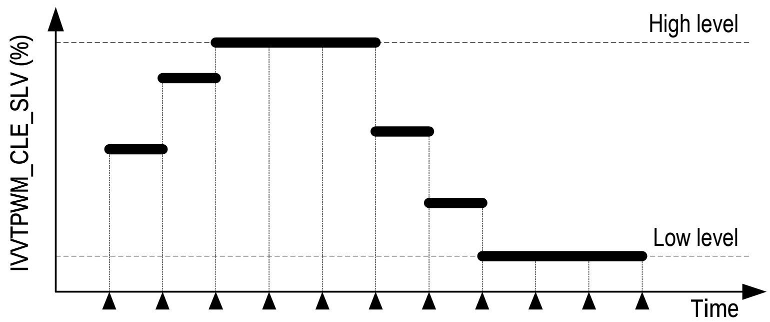


Figure: "Oil control solenoid valve cleaning cycle"

The cleaning can be disabled by LV_DI_CLE_SLV_IVVT (application incidence).

The oil control solenoid valve cleaning occurs in the power-latch too. The power-latch is to hold while the cleaning is active.

Application conditions:

Initialization: *At reset:* CTR_CYC_CLE_SLV_IVVT = 1

At activation: LV_DLY_CLE_SLV_IVVT = 0

CTR_STEP_CLE_SLV_IVVT = 1

At deactivation:

LV_ACT_CLE_SLV_IVVT = 0

if LV_ACT_PWL_CLE_SLV_IVVT = 1

then ACTION_FCTM_ReleasePostOpPhase(CLE_SLV_IVVT, CALL)

LV_ACT_PWL_CLE_SLV_IVVT = 0

endif

Recurrence: 10 ms

Activation: LV_ES = 1

and LV_IVVTPWM = 1

and CTR_CYC_CLE_SLV_IVVT <= C_CTR_CYC_CLE_SLV_IVVT

and TOIL > C_TOIL_MIN_CLE_SLV_IVVT

and VB >= C_VB_MIN_IVVT

and VB <= C_VB_MAX_IVVT

and LV_DI_CLE_SLV_IVVT = 0

Deactivation: Not activation

Formula section:

Delay before cleaning:

```
if      LV_DLY_CLE_SLV_IVVT = 0
then    Delay before cleaning
        LV_DLY_CLE_SLV_IVVT = 1
        T_ST_CLE_SLV_IVVT = T_DLY_MES_IVVT
elseif          LV_ACT_CLE_SLV_IVVT = 0
        and     [       (       T_DLY_MES_IVVT - T_ST_CLE_SLV_IVVT >=
                                C_T_DLY_ST_CLE_SLV_IVVT
                        and     LV_IGK = 1)
                or      (       T_DLY_MES_IVVT - T_ST_CLE_SLV_IVVT >=
                                C_T_DLY_PWL_CLE_SLV_IVVT
                        and     LV_IGK = 0)
                ]
then    Cleaning kick off
        LV_ACT_CLE_SLV_IVVT = 1
endif
```

Cleaning:

```
if              LV_ACT_CLE_SLV_IVVT = 1
then            Cleaning active
                IVVTPWM_CLE_SLV_IN= ID_IVVTPWM_CLE_SLV
                                * FAC_COR_IVVTPWM_IN
                CTR_STEP_CLE_SLV_IVVT = CTR_STEP_CLE_SLV_IVVT + 1
        If              CONF_CAM_VVT_EX = 1
        then            IVVTPWM_CLE_SLV_EX = ID_IVVTPWM_CLE_SLV
                                * FAC_COR_IVVTPWM_EX
        else            IVVTPWM_CLE_SLV_EX = 0 [%]
                if      CTR_STEP_CLE_SLV_IVVT > 2 *
                        C_CTR_STEP_CLE_SLV_IVVT
                then    Cleaning cycle finished
                        CTR_STEP_CLE_SLV_IVVT = 1
                        CTR_CYC_CLE_SLV_IVVT = CTR_CYC_CLE_SLV_IVVT + 1
                endif
        endif
endif
```

Power-latch:

if LV_PWL(n-1) = 0

and LV_PWL(n) = 1

and CTR_CYC_CLE_SLV_IVVT <= C_CTR_CYC_CLE_SLV_IVVT

then ACTION_FCTM_HoldPostOpPhase

(CLE_SLV_IVVT, C_DLY_MAX_PWL_CLE_SLV_IVVT, CALL)

LV_ACT_PWL_CLE_SLV_IVVT = 1

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TOIL_MIN_CLE_SLV_IVVT	1	0...C8H	-40...160	1	°C
Minimum oil temperature for oil control valve cleaning					
C_CTR_STEP_CLE_SLV_IVVT	1	0...FFH	0...255	1	-
Number of oil control solenoid valve cleaning steps in half cycle					
C_CTR_CYC_CLE_SLV_IVVT	1	0...FFH	0...255	1	-
Number of oil control solenoid valve cleaning cycles in one procedure					
C_T_DLY_ST_CLE_SLV_IVVT	1	0...FFH	0...10200	40	ms
Delay before oil control solenoid valve cleaning if engine not yet started					
C_T_DLY_PWL_CLE_SLV_IVVT	1	0...FFH	0...10200	40	ms
Delay before oil control solenoid valve cleaning during power latch					
ID_IVVTPWM_CLE_SLV	6	0...FFH	0...99.6	0.39	%
LDP_CTR_STEP_ID_IVVTPWM_CLE	6	0...FFH	0...255	1	-
Pulse width modulation of oil control solenoid valve cleaning cycle					
C_DLY_MAX_PWL_CLE_SLV_IVVT	1	0...FFH	0...25.5	0.1	s
Maximum duration of power-latch holding request due to IVVT oil control solenoid valve cleaning					

9.70 IVVT Release

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_IVVTPWM	O/V	0...1H	0...1	1	-
1 = solenoid valve energization enabled, 0 = not enabled					

Input data:

FUNCTION DESCRIPTION:

General information:

The start of the energization of the IVVT power stage(s) is specified here.

Description:

The solenoid valve energization is always enabled.

Application conditions:

Initialization:

Recurrence:

100 ms

Activation:

At every engine state

Formula section:

LV_IVVTPWM = 1

9.71 IVVT Substitue PWM - General

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
T_PER_ACR_TEST_IVVT	V	0...FFH	0...25500	100	ms
Timer for IVVT actuator test signal generation					
IVVTPWM_ACR_TEST_IVVT	V	0...FFH	0...99.609	0.39	%
Level of IVVT actuator test signal					
LV_IVVTPWM_ACR_TEST_IN_i	V/O	0...FFH	0...1	1	-
1 = IVVT actuator test active, 0 = inactive; inlet					
LV_IVVTPWM_ACR_TEST_EX_i	V/O	0...FFH	0...1	1	-
1 = IVVT actuator test active, 0 = inactive; exhaust					

Input data:

STATE_ACR_TEST_IVVT	LV_ES	CONF_CAM_VVT_EX	

FUNCTION DESCRIPTION:

General information:

An actuator test is done by means of a special PWM signal.

Description:

An actuator test function can request 1 Hz 50 % PWM signal for each individual camshaft. This is indicated by STATE_ACR_TEST_IVVT. The test function guarantees that some conditions are fulfilled when requesting tests. Those relevant for the IVVT system are that the solenoids can be energised and that the engine speed is 0.

The 1 Hz 50 % PWM test signal is generated by means of 100 % PWM 0.5 s lasting and 0 % PWM lasting 0.5 s with the frequency of the holding energisation.

Application conditions:

Initialization:

At transition "engine stop" → "engine run":

T_PER_ACR_TEST_IVVT = 0

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1:

LV_IVVTPWM_ACR_TEST_IN_1 = 0

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_EX = 1:

LV_IVVTPWM_ACR_TEST_EX_1 = 0

Recurrence: 100 ms

Activation: LV_ES = 1

Deactivation: No activation

Formula section:

Set all actuator test activation flags zero:

 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1:
LV_IVVTPWM_ACR_TEST_IN_1 = 0
 # NC_NR_CBK_IVVT = 1, NLC_IVVT_EX = 1:
LV_IVVTPWM_ACR_TEST_EX_1 = 0

Calculate test signal and activate corresponding camshaft:

if STATE_ACR_TEST_IVVT <> "PASSIVE"
then *Actuator test active*
  if STATE_ACR_TEST_IVVT(n) <> STATE_ACR_TEST_IVVT(n-1)
    or T_PER_ACR_TEST_IVVT >= 1000 ms
  then *Change of tested camshaft or test signal period over*
    T_PER_ACR_TEST_IVVT = 0 ms
  endif
  if T_PER_ACR_TEST_IVVT < 500 ms
  then *High level half period*
    IVVTPWM_ACR_TEST_IVVT = FFH
  else *Low level half period*
    IVVTPWM_ACR_TEST_IVVT = 0H
  endif
  T_PER_ACR_TEST_IVVT = T_PER_ACR_TEST_IVVT + 100 ms
  Activate corresponding individual camshaft:
  # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1
  if STATE_ACR_TEST_IVVT = "IN_1"
  then LV_IVVTPWM_ACR_TEST_IN_1 = 1
  endif
  # NC_NR_CBK_IVVT = 1, NLC_IVVT_EX = 1
  if STATE_ACR_TEST_IVVT = "EX_1"
  and CONF_CAM_VVT_EX = 1
  then LV_IVVTPWM_ACR_TEST_EX_1 = 1
  endif
endif

9.72 IVVT Substitute PWM

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_IVVTPWM_SUB_IN_i	O/V	0...1H	0...1	1	-
1 = substitute solenoid valve energization, 0 = IVVT algorithm energization; inlet					
LV_IVVTPWM_SUB_EX_i	O/V	0...1H	0...1	1	-
1 = substitute solenoid valve energization, 0 = IVVT algorithm energization; exhaust					
IVVTPWM_SUB_IN_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Substitute pulse width modulation; inlet					
IVVTPWM_SUB_EX_i	O/V	0...FFFFH	0...99.99847	0.001526	%
Substitute pulse width modulation; exhaust					

Input data:

LV_IVVTPWM_ACR_TEST_IN_i	LV_IVVTPWM_ACR_TEST_EX_i	IVVTPWM_ACR_TEST_IVVT
LV_ERR_VB_OC		

FUNCTION DESCRIPTION:

General information:

The normal energization of the IVVT solenoid valves can be substituted by a special energization.

Description:

The solenoid valve energization is disabled if error on Voltage supply of ECU is detected and can be set manually or a test signal can be handed over.

Application conditions:

Initialization:

Recurrence:

100 ms

Activation:

At every engine state

Formula section:

if LV_ERR_VB_OC = 1

then disable engergization

LV_IVVTPWM_SUB_IN(_EX)_i = 1

IVVTPWM_SUB_IN(_EX)_i = 0 %

else **if** LC_IVVTPWM_MAN_IN(_EX)_i = 1

then *Manual energization*

LV_IVVTPWM_SUB_IN(_EX)_i = 1

IVVTPWM_SUB_IN(_EX)_i = C_IVVTPWM_MAN_IN(_EX)_i

else **if** LV_IVVTPWM_ACR_TEST_IN(_EX)_i = 1

then *IVVT actuator test*

LV_IVVTPWM_SUB_IN(_EX)_i = 1

IVVTPWM_SUB_IN(_EX)_i = IVVTPWM_ACR_TEST_IVVT

else

LV_IVVTPWM_SUB_IN(_EX)_i = 0

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_IVVTPWM_MAN_IN_i	1	0...1H	0...1	1	-
1 = manual solenoid valve energization, 0 = IVVT algorithm energization; inlet					
LC_IVVTPWM_MAN_EX_i	1	0...1H	0...1	1	-
1 = manual solenoid valve energization, 0 = IVVT algorithm energization; exhaust					
C_IVVTPWM_MAN_IN_i	1	0...FFFFH	0...99.99847	0.001526	%
Manual pulse width modulation; inlet					
C_IVVTPWM_MAN_EX_i	1	0...FFFFH	0...99.99847	0.001526	%
Manual pulse width modulation; exhaust					

9.73 IVVT Disabling Oil Control Valve Cleaning

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_DI_CLE_SLV_IVVT	O/V	0...1H	0...1	1	-
1 = oil control valve cleaning disabled, 0 = not disabled					

Input data:

FUNCTION DESCRIPTION:

General information:

The oil control valve cleaning can be disabled due to specific reasons.

Description:

The oil control valve cleaning is not disabled due to specific reasons.

Application conditions:

Initialization:

Recurrence:

100 ms

Activation:

At every engine state

Formula section:

LV_DI_CLE_SLV_IVVT = 0

9.74 IVVT Disabling Reference Position Adaptation

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_CAM_SP_REF_AD_DI_IVVT	O/V	0...1H	0...1	1	-
1 = reference position adaptation if setpoint in ref. pos. disabled, 0 = not disabled					

Input data:

N_32	MAF_HB	TOIL	LV_IGK
CTR_ABC_SLV_IVVT_IN_i	CTR_ABC_SLV_IVVT_EX_i	CTR_ABC_CAM_DYN_IVVT_IN_i	CTR_ABC_CAM_DYN_IVVT_EX_i
CTR_ABC_MEC_IVVT_IN_i	CTR_ABC_MEC_IVVT_EX_i	CTR_ABC_TRIP_IVVT_IN_i	CTR_ABC_TRIP_IVVT_EX_i
CTR_ABC_CAM_STAT_IVVT_IN_i	CTR_ABC_CAM_STAT_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

It might be advantageous to disable the reference position adaptation under certain conditions. E.g., a new reference position can be adapted at very high engine speed due to huge centrifugal forces and camshaft sensor dirft. Such operating state usually outlasts only for short time, and the newly learnt reference position is not suitable for the most operating states.

Description:

The adaptation is disabled if the engine load is over C_MAF_HB_MAX_AD_REF_DI_IVVT, the engine speed is outside of the range C_N_32_MIN(_MAX)_AD_REF_DI_IVVT and the oil temperature is outside of the range C_TOIL_MIN(_MAX)_AD_REF_DI_IVVT.

The adaptation is also disabled if IVVT is under error conditions, such as electrical, deviation (stuck, plausibility) and short trip.

Application conditions:

Initialization:

At reset, at deactivation:

LV_CAM_SP_REF_AD_DI_IVVT = 1

Recurrence:

100 ms

Activation:

At every engine state

Formula section:

if MAF_HB > C_MAF_HB_MAX_AD_REF_DI_IVVT

or N_32 < C_N_32_MIN_AD_REF_DI_IVVT
or N_32 > C_N_32_MAX_AD_REF_DI_IVVT
or TOIL < C_TOIL_MIN_AD_REF_DI_IVVT
or TOIL > C_TOIL_MAX_AD_REF_DI_IVVT
or CTR_ABC_SLV_IVVT_IN_i <> 0 **or** CTR_ABC_SLV_IVVT_EX_i <> 0
or CTR_ABC_CAM_STAT_IVVT_IN_i <> 0
or CTR_ABC_CAM_STAT_IVVT_EX_i <> 0
or CTR_ABC_CAM_DYN_IVVT_IN_i <> 0
or CTR_ABC_CAM_DYN_IVVT_EX_i <> 0
or CTR_ABC_TRIP_IVVT_IN_i <> 0
or CTR_ABC_TRIP_IVVT_EX_i <> 0
or CTR_ABC_MEC_IVVT_IN_i <> 0
or CTR_ABC_MEC_IVVT_EX_i <> 0
or LV_IGK = 0

then

LV_CAM_SP_REF_AD_DI_IVVT = 1

else

LV_CAM_SP_REF_AD_DI_IVVT = 0

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_MAF_HB_MAX_AD_REF_DI_IVVT	1	0...FFH	0..1389	5.447	mg/stk
Maximum allowed engine load for IVVT reference position adaptation					
C_N_32_MIN_AD_REF_DI_IVVT	1	0...FFH	0...8160	32	rpm
Minimum required engine speed for IVVT reference position adaptation					
C_N_32_MAX_AD_REF_DI_IVVT	1	0...FFH	0...8160	32	rpm
Maximum allowed engine speed for IVVT reference position adaptation					
C_TOIL_MIN_AD_REF_DI_IVVT	1	0...C8H	-40...160	1	°C
Minimum required oil temperature for IVVT reference position adaptation					
C_TOIL_MAX_AD_REF_DI_IVVT	1	0...C8H	-40...160	1	°C
Maximum allowed oil temperature for IVVT reference position adaptation					

9.75 IVVT Lock at Transition to Engine Stop

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_CAM_LOCK_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = actuator locked at transition to engine stop, 0 = not locked; inlet					
LV_CAM_LOCK_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = actuator locked at transition to engine stop, 0 = not locked; exhaust					

Input data:

FUNCTION DESCRIPTION:

General information:

The ENSD aggregate needs the information whether the actuator locking system was locked at the transition to engine stop or not at the next engine start.

Description:

LV_CAM_LOCK_IVVT_IN(_EX)_i is handled as if the actuator were never locked at the transition to engine stop.

Application conditions:

Initialisation: LV_IGK = 0-->1

LV_CAM_LOCK_IVVT_IN(EX)=1

Recurrence: At each used camshaft edge, see "IVVT Scheduler"

Activation: At every engine state

Formula section:

LV_CAM_LOCK_IVVT_IN(EX) = 1 (always remain 1)

9.76 Pressure upstream throttle setpoint

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
PUT_SP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
pressure before throttle set point (prs_up_thr-control)					
STATE_PUT_SP	O/V	0... 2H	0... 2	1	[-]
State whether pressure up throttle setpoint is below AMP (=0), between AMP and PUT_BAS (=1) or above PUT_BAS (=2)					

Input Data:

LC_TCHA_CONF	LV_TQI_TCHA_PROT	MAP_SP	PUT_SP_TOL
PUT_SP_BOL	AMP	PUT_WG_OPEN	PUT_MDL_DIF_I_MMV
PUT_SP_MAN			

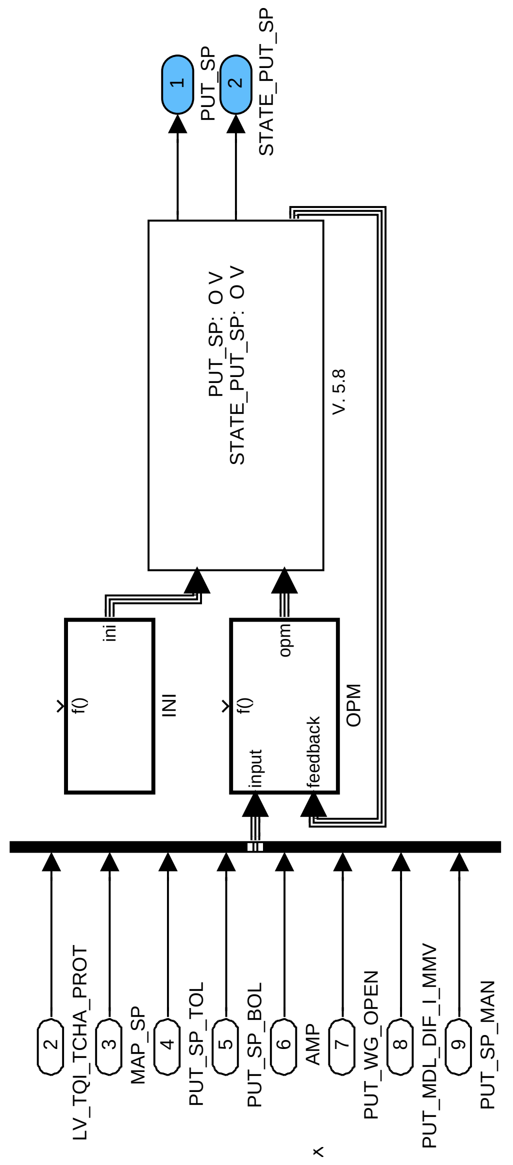
Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_PUT_SP_TQI_TCHA_PROT	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Correlation constant for PUT_SP filtering for turbo charger protection					
IP_PUT_SP	8	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_MAP_SP_IP_PUT_SP	8	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure up throttle setpoint					
LC_PUT_MDL_DIF_PUT_SP	1	0... 1H	0... 1	1	[-]
Switch whether PUT_MDL_DIF_I_MMV influences PUT_SP					
LC_PUT_SP_AS	1	0... 1H	0... 1	1	[-]
Logical constant for manual pressure up throttle setpoint					
LC_PUT_SP_TQI_TCHA_PROT_ACT	1	0... 1H	0... 1	1	[-]
Activation of PUT_SP filtering for turbo charger protection					

General Information

Pressure upstream throttle setpoint PUT_SP is calculated in dependence of the requested intake manifold air pressure. In normal operation it is equal or slightly higher than MAP_SP, bottom and top limit of PUT_SP can be defined in the according application incidences.

Function description



SDA_SRS / SDA V 5.0.4 29−Mar−2006

Figure 176:
Path: CHRG_DEFSPPUT0

9.76.1 CHRG_DEFSPPUT0/INI

CONTENT

9.76.1.1 CHRG_DEFSPPUT0/INI/CLC_INI

CONTENT

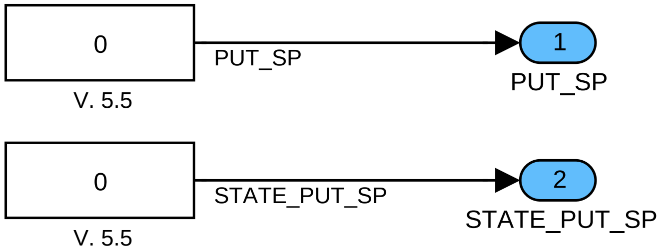


Figure 177:
Path: CHRG_DEFSPPUT0/INI/CLC_INI

9.76.2 CHRG_DEFSPPUT0/OPM

9.76.2.1 CHRG_DEFSPPUT0/OPM/CLC_OPM

The regular pressure upstream throttle setpoint is low passed filtered in case of torque limitation for turbo charger protection. The result may be overwritten by a manual value.

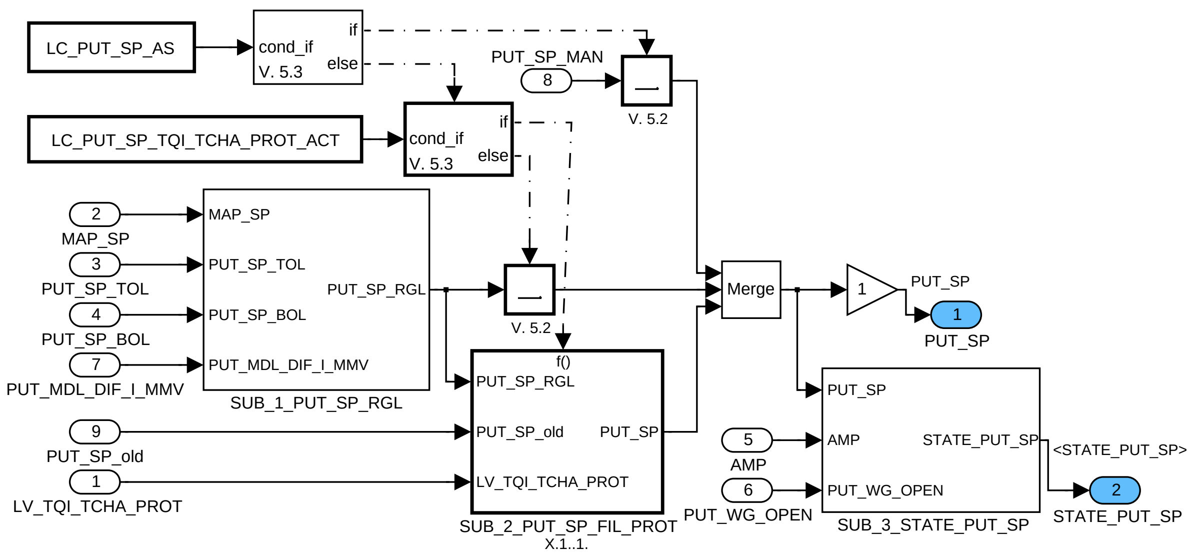


Figure 178:
Path: CHRG_DEFSPPUT0/OPM/CLC_OPM

9.76.2.1.1 No title given for this section

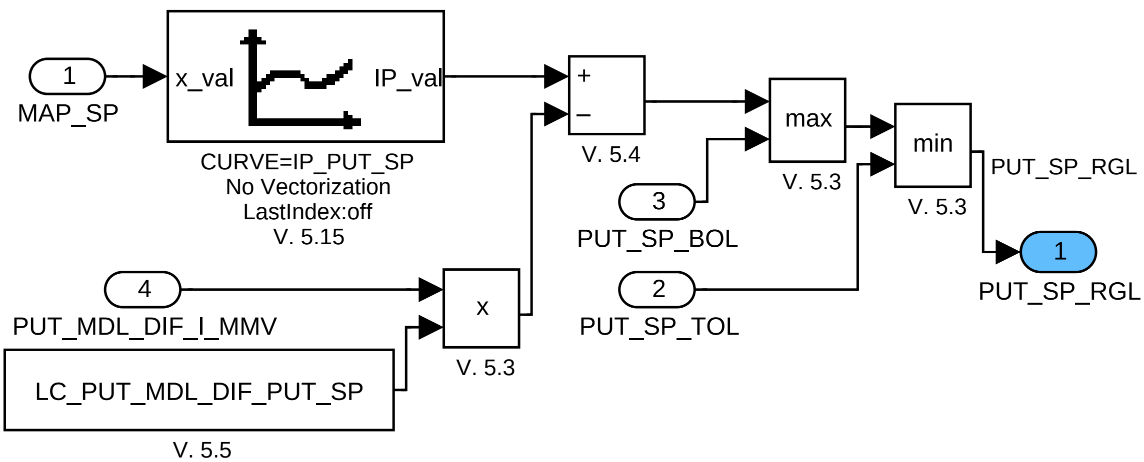


Figure 179:
Path: CHRG_DEFSPPUT0/OPM/CLC_OPM/SUB_1_PUT_SP_RGL

9.76.2.1.2 CHRG_DEFSPPUT0/OPM/CLC_OPM/SUB_2_PUT_SP_FIL_PROT

When engine torque limitation for turbo charger protection is active (LV_TQI_TCHA_PROT = 1) PUT_SP is not allowed to increase as rapidly as MAP_SP in order to stabilize the turbo charger operation setpoint. In this case PUT_SP is low pass filtered.

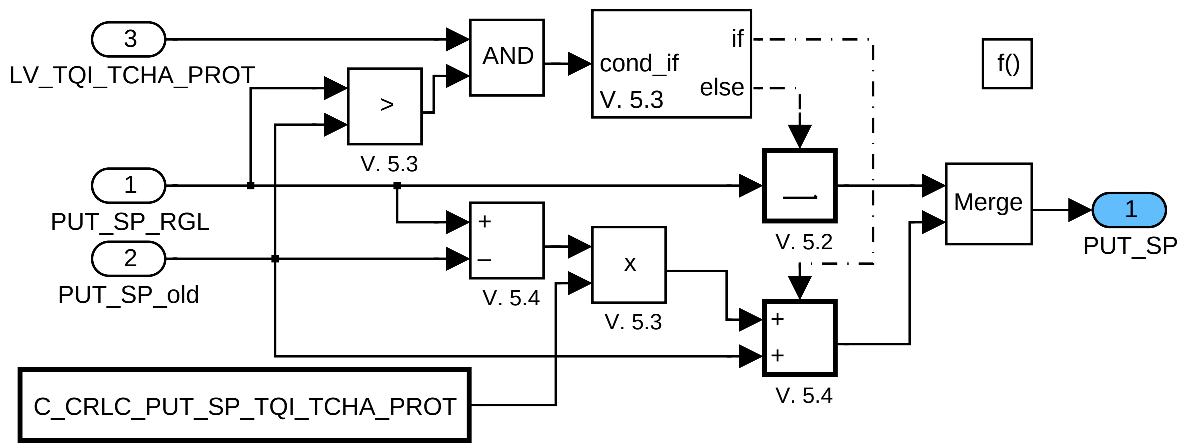


Figure 180:
Path: CHRG_DEFSPPUT0/OPM/CLC_OPM/SUB_2_PUT_SP_FIL_PROT

9.76.2.1.3 CHRG_DEFSPPUT0/OPM/CLC_OPM/SUB_3_STATE_PUT_SP

STATE_PUT_SP shows whether PUT_SP is below ambient pressure (= 0), between ambient pressure and basic charge air pressure (= 1) or above basic charge air pressure (= 2).

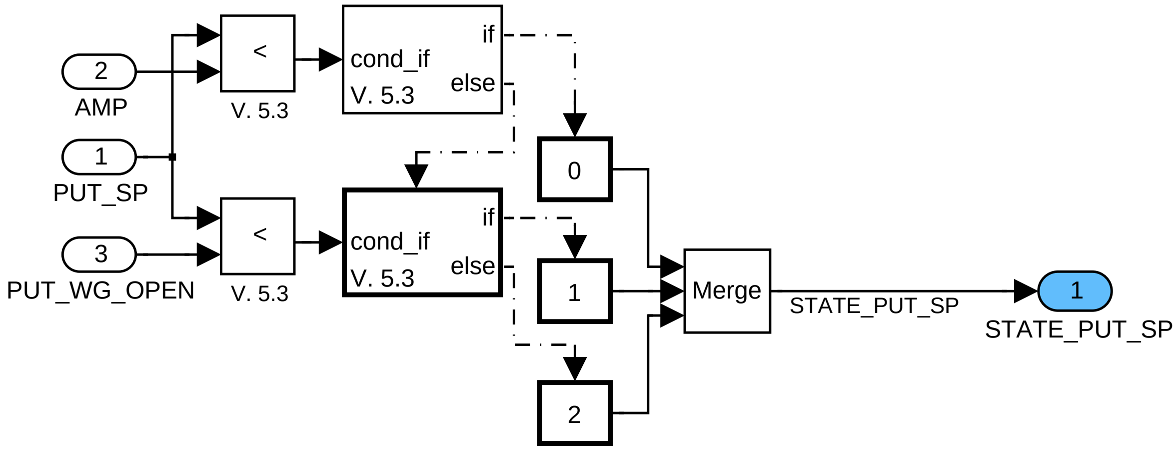


Figure 181:
Path: CHRG_DEFSPPUT0/OPM/CLC_OPM/SUB_3_STATE_PUT_SP

9.77 Pressure upstream throttle setpoint (Application Incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PUT_SP_TOL	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Top limit of pressure upstream throttle setpoint					
PUT_SP_BOL	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Bottom limit of pressure upstream throttle setpoint					
LV_PUT_OPTM_RESP_ACT	O/V	0...1H	0...1	1	-
Indicator that "best response" mode is active					

Input data:

LC_TCHA_CONF	LV_PUT_ADD_TCHA_BOOST_REQ	MAP_SP_PQ_MAX	IP_PUT_SP
LV_PV_SPT_ACT	MAF_KGH_SP_TCHA	TAM	VOL_FLOW_CHA_UP_SP
PRS_CHA_UP_MV			
IP_PQ_CHA_SURGE_DL			
IP_PRS_LOSS_ICO			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_PQ_CHA_SURGE_PUT_SP_TOL	1	0...FFH	0...1.9921875	0.0078125	-
Surge line weighting factor for PUT_SP_TOL calculation					
LC_PUT_ADD_TCHA_BOOST_ACT	1	0...1H	0...1	1	-
Activation switch for "best response" mode at requested boost					
LC_PUT_OPTM_RESP_ENA	1	0...1H	0...1	1	-
Enable "best response" mode					

9.77.1 CHRG_DEFSPPUT0

Bottom and top limit of charge air pressure setpoint PUT_SP can be defined here.

A bottom limit can be activated for Boost request (LV_PUT_ADD_TCHA_BOOST_REQ= 1) or in "sport" mode (= "Best response" mode).

Function Description

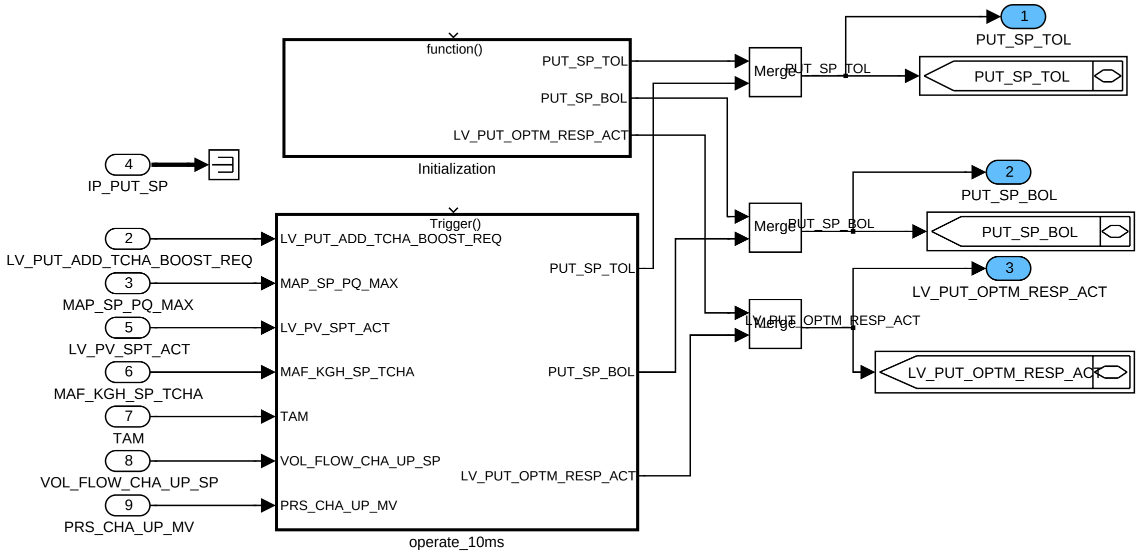


Figure 182 CHRG_DEFSPPUT0

9.77.1.1 SUBFUNCTION: Initialization

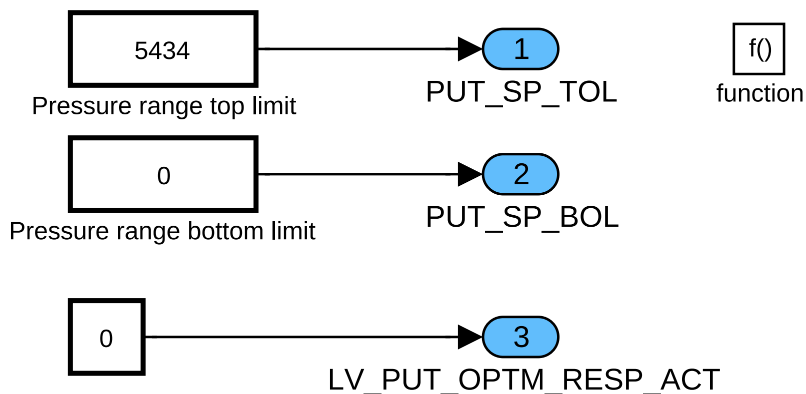


Figure 183 CHRG_DEFSPPUT0/ Initialization

9.77.1.2 SUBFUNCTION: operate_10ms

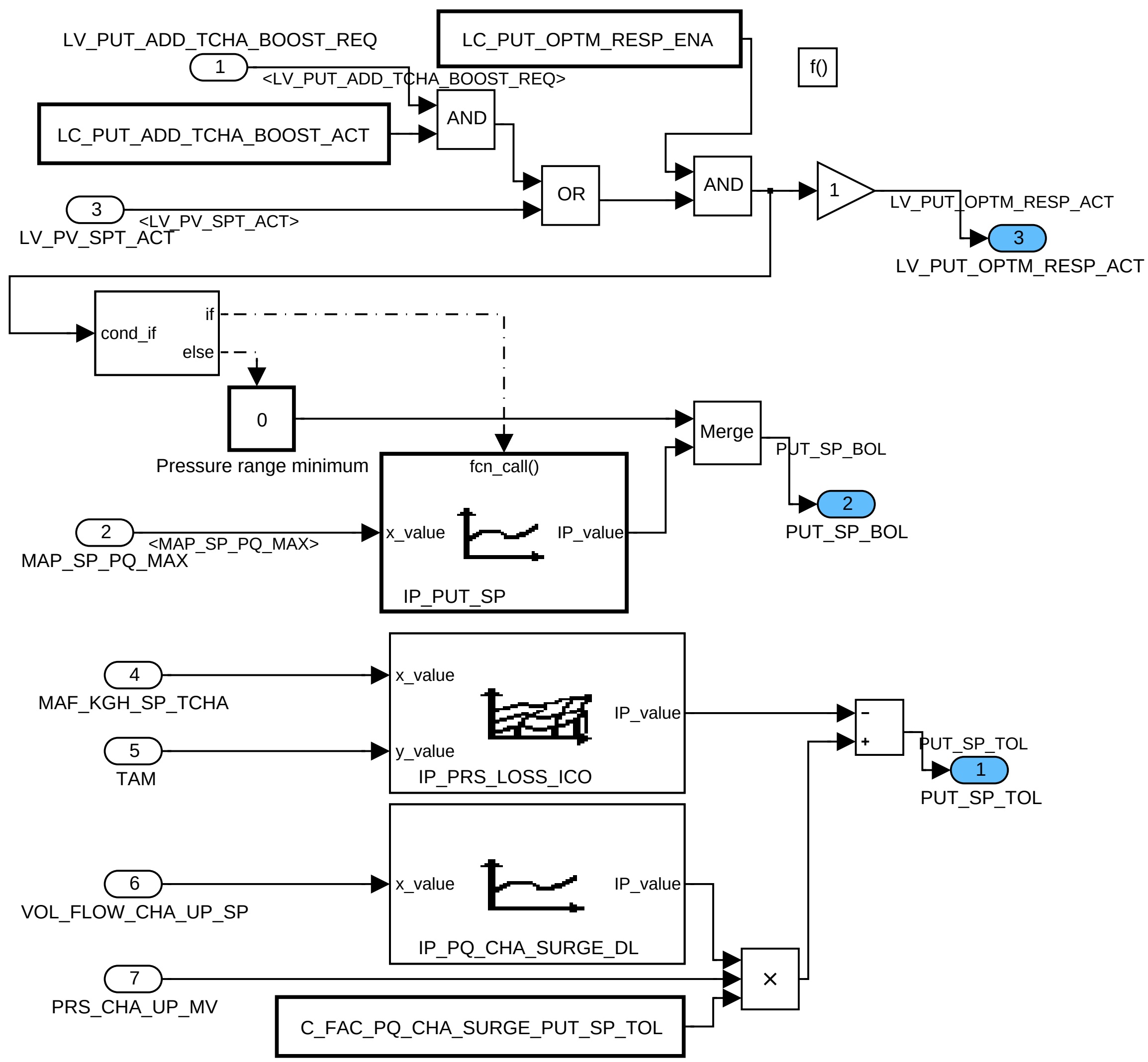


Figure 184 CHRG_DEFSPPUT0/ operate_10ms

9.78 Charger model for turbo charger at setpoint (Appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_CHA_SP	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
mass air flow through charger at setpoint					
VOL_FLOW_CHA_UP_SP	O/V	0...FFFFH	0...1.99996948	3.05176E-5	m3/s
volumetric flow upstream charger at setpoint					
PQ_CHA_SP	O/V	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure ratio before charger set point					
PRS_CHA_DOWN_SP	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
pressure downstream charger at setpoint					
PRS_CHA_UP_SP_CLC	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure upstream charger at setpoint calculated					
RHO_CHA_UP_SP	V	0...FFFFH	0...2.27552	3.47222E-5	kg/m3
Air density upstream charger at setpoint					
PRS_CHA_UP_SP	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
pressure upstream charger at setpoint					
PRS_CHA_DOWN_SP_CLC	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure downstream charger at setpoint calculated selected					
PQ_CHA_CLC_SP	V	0...FFFFH	0...15.9997559	2.44141E-4	-
Pressure quotient at charger calculated, setpoint					

Input data:

AMP	TIA_ABSV_CHA_UP	MAF_KGH_SP_TCHA	PUT_SP
TAM	LC_TCHA_CONF	IP_PRS_LOSS_ICO	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PQ_CHA_SP_MMV	1	0...FFH	0...0.99609375	0.00390625	-
correlation constant to calculate PQ_CHA_SP					
C_PRS_CHA_DOWN_SP_AS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
charge pressure downstream charger, up Intercooler charger application system					
C_PRS_CHA_UP_SP_AS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
charge pressure upstream charger application system					
LC_PRS_CHA_DOWN_SP_AS	1	0...1H	0...1	1	-
Switch for use charge pressure downstream charger, up Intercooler application system					
LC_PRS_CHA_UP_SP_AS	1	0...1H	0...1	1	-
Switch for use charge pressure upstream charger application system					
IP_PRS_DIF_AIC_TCHA	4	0...FFFFH	0...5.434E+3	0.08291752	hPa
LDP_MAF_KGH_IP_PRS_DEC_AIC_TCHA	4	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Pressure decrease through the air cleaner					

9.78.1 CHRG_ISPCLCHAEN0

This module calculates the conditions at the charger, the intercooler and the air cleaner at the desired setpoint.

In this 1st step it is assumed that the recirculation path will be closed when the charge air pressure control is active. Therefore the mass air flow through air cleaner, charger intercooler and engine at the set point are equal.

Function Description

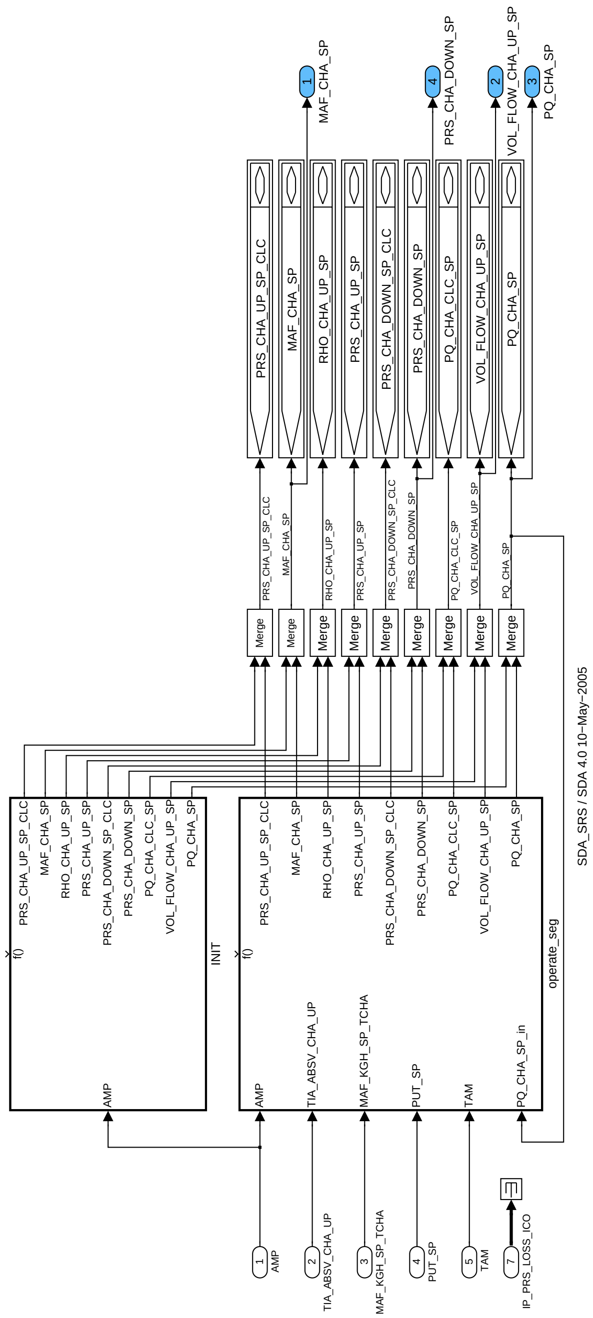


Figure 185 CHRG_ISPCLCHAEN0

9.78.1.1 SUBFUNCTION: INIT

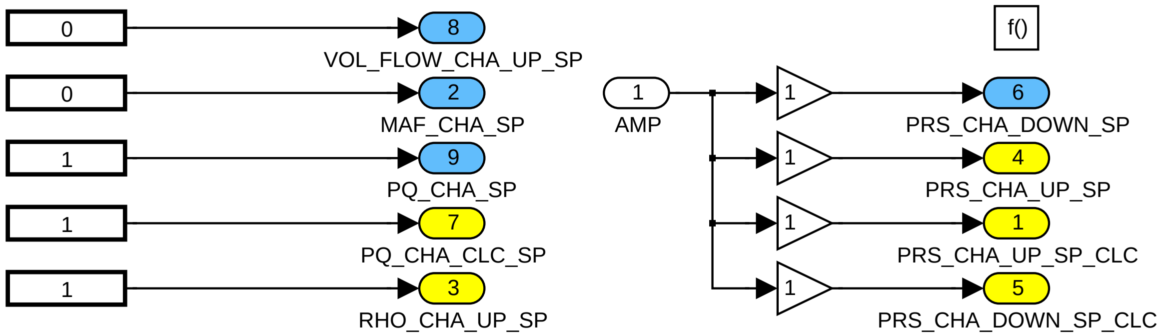


Figure 186 CHRG_ISPCLCHAEN0/ INIT

9.78.1.2 SUBFUNCTION: operate_seg

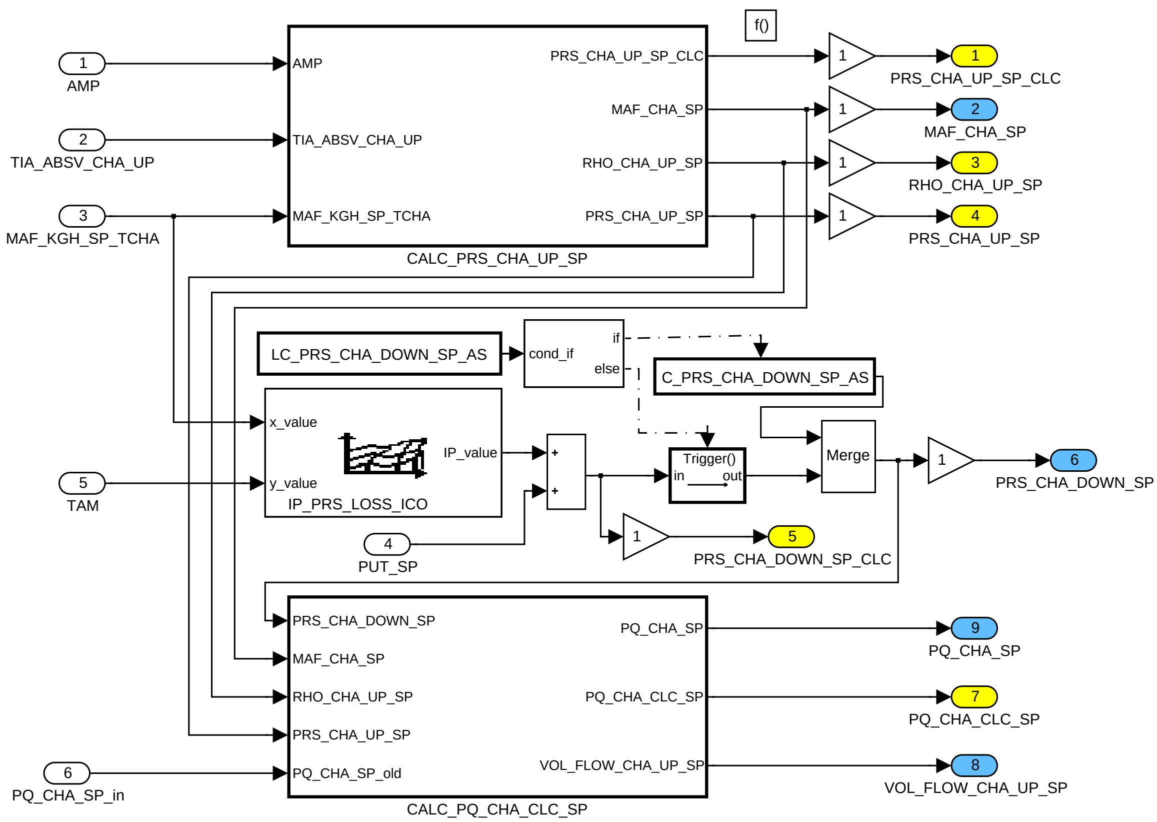


Figure 187 CHRG_ISPCLCHAEN0/ operate_seg

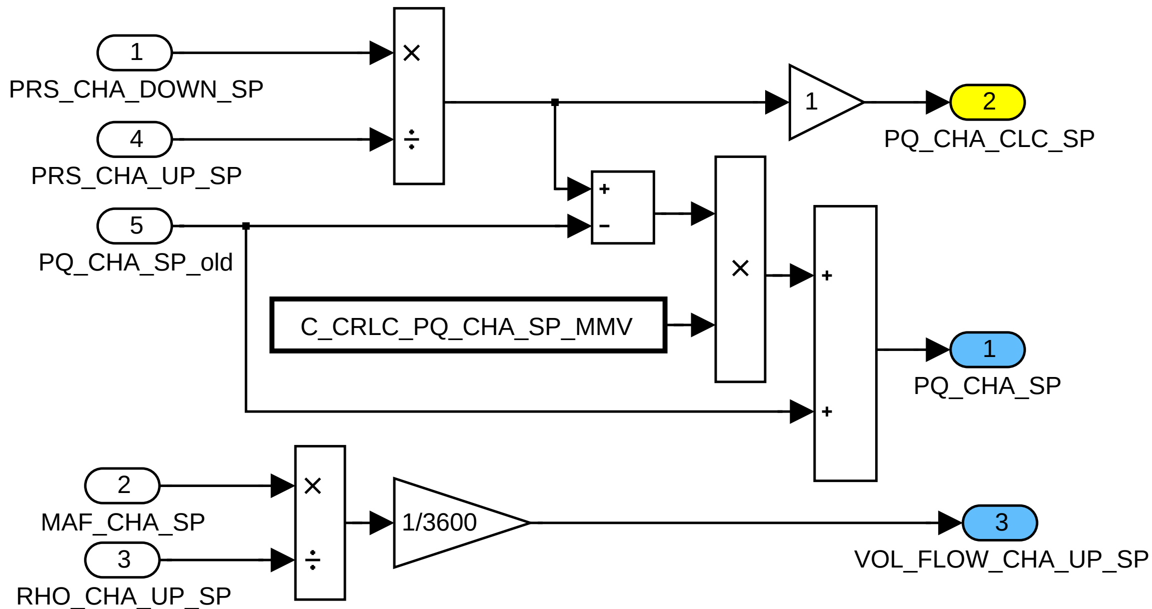


Figure 188 CHRG_ISPCLCHAEN0/ operate_seg/ CALC_PQ_CHA_CLC_SP

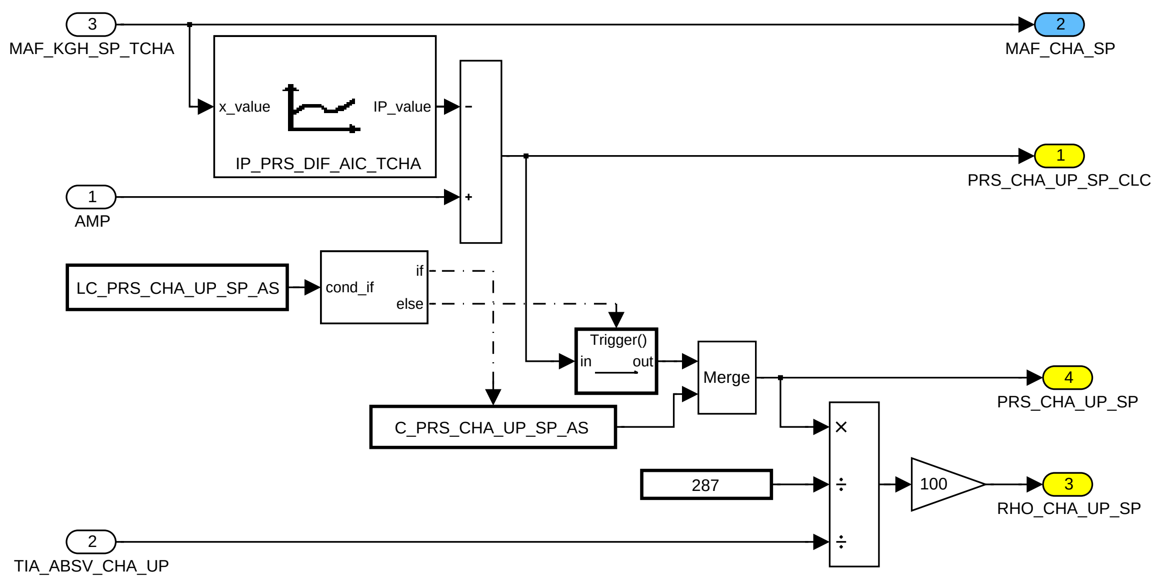


Figure 189 CHRG_ISPCLCHAEN0/ operate_seg/ CALC_PRS_CHA_UP_SP

9.79 Turbo Charger Protection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TQI_TCHA_PROT	O/V	0...1H	0...1	1	-
Logical variable for active turbocharger protection by torque request reduction					
TQI_TCHA_PROT	O/V	0...7FFFH	0...1.02397E+3	0.03125	Nm
Indicated torque for turbo charger protection as an interface to minimum / maximum torque request selection					
LV_TCHA_PROT_ACT	O/V	0...1H	0...1	1	-
Logical variable for active turbocharger protection leading to opening of the wastegate					
LV_N_TCHA_PROT	O/V	0...1H	0...1	1	-
Logical variable for active turbocharger protection leading to opening of the wastegate: charger speed criterium					
LV_TQI_N_TCHA_MAX	O/V	0...1H	0...1	1	-
boolean for active turbine overspeed control via torque reduction					
STATE_TCHA_PROT	V	0...FFFFH	0...6.5535E+4	1	-
Turbo charger protection status					
N_TCHA_LPF_PROT	V	0...FFFFH	0...3.99994E+5	6.10351563	rpm
Filtered charger speed for turbo charger protection					
N_TCHA_MAX_SP_DIF_ACT	V	8000...7FFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
Charger speed setpoint activation threshold for torque limitation					
N_TCHA_MAX_SP_DIF	V	8000...7FFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
Turbo charger speed difference to maximum allowed charger speed setpoint					
TQI_MAX_TCHA_PROT_WG	V	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum indicated torque during torque limitation for turbocharger protection because of wastegate error					
N_TCHA_MAX_SP_TRAN	V	0...FFFFH	0...3.99994E+5	6.10351563	rpm
Transient maximum turbo charger speed setpoint for turbo charger protection by torque setpoint reduction					
TQI_N_TCHA_MAX	V	0...7FFFH	0...1.02397E+3	0.03125	Nm
indicated torque for turbo charger protection from overspeed					
TQI_N_TCHA_MAX_I	V	0...7FFF0000H	0...1.024E+3	4.76852E-7	Nm
I-part of the PI-controller of the turbo overspeed prevention by torque reduction					
CTR_ABC_N_TCHA_PROT	V	0...FFH	0...255	1	-
Anti bounce counter for turbo charger deactivation because of too high charger speed					
N_TCHA_MAX_DIF_TRAN	V	8000...7FFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
Turbo charger speed difference to maximum allowed charger speed					
TQI_N_TCHA_MAX_P	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
P-part of the PI-controller of the turbo overspeed prevention by torque reduction					
LV_TQI_WG_NOT_PLAUS	V	0...1H	0...1	1	-
boolean for active waste gate failure control via torque setpoint reduction					
LV_TQI_RCL_NOT_PLAUS	V	0...1H	0...1	1	-
boolean for active recirculation valve failure control via torque setpoint reduction					

Input data:

LC_TCHA_CONF	N_TCHA	LV_IN_PROT	LV_TEG_MAX_TCHA_DEAC
LV_ERR_WG	LV_ST_END	LV_ERR_RATIO_CHK	PRS_CHRG_DOWN
TQI_BAS_MAX	TQI_ADD_ACT	LV_ERR_CAP_H	LV_ERR_PLAUS_CLOSE_RCL
TQI_AV	N_TCHA_GRD	ERR_SYM_EL_RCL_ACR	LV_ERR_EL_RCL_ACR
AMP	MAP_SP	LV_ERR_AMP	IP_PUT_SP
PUT_SP_TOL	PUT_SP		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_N_TCHA_LPF_PROT	1	0...FFH	0...0.99609375	0.00390625	-
Charger speed filtering constant for turbo charger protection					
C_CRLC_N_TCHA_MAX_SP_TRAN	1	0...FFH	0...0.99609375	0.00390625	-
Maximum charger speed setpoint filtering constant for turbo charger protection					
C_CTR_ABC_MAX_N_TCHA_PROT	1	0...FFH	0...255	1	-
Anti bounce counter for turbo charger deactivation because of too high charger speed					
C_FAC_TQI_N_TCHA_MAX_I_NEG	1	0...FFFFH	0...0.00511992	7.8125E-8	Nm/rpm
i-part of indicated torque controler for negative error variable					
C_FAC_TQI_N_TCHA_MAX_I_POS	1	0...FFFFH	0...0.00511992	7.8125E-8	Nm/rpm
i-part of indicated torque controler for positive error variable					
C_FAC_TQI_N_TCHA_MAX_P_NEG	1	0...FFFFH	0...0.00511992	7.8125E-8	Nm/rpm
p-part of indicated torque controler for negative error variable					
C_FAC_TQI_N_TCHA_MAX_P_POS	1	0...FFFFH	0...0.00511992	7.8125E-8	Nm/rpm
p-part of indicated torque controler for positive error variable					
C_N_TCHA_MAX	1	0...FFFFH	0...3.99994E+5	6.10351563	rpm
Maximum turbo charger speed					
C_N_TCHA_MAX_SP	1	0...FFFFH	0...3.99994E+5	6.10351563	rpm
Maximum turbo charger speed setpoint for turbo charger protection by torque setpoint reduction					
C_PUT_MAX_TQI_TCHA_PROT	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum allowed PUT during torque limitation for turbocharger protection					
C_TQI_MAX_TCHA_PROT	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum allowed indicated torque in case of turbo charger deactivation					
C_TQI_MAX_TCHA_PROT_AMP	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum allowed indicated torque in case of turbo charger deactivation					
C_TQI_MAX_TCHA_PROT_RCL	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum torque in case of recirculation valve failure for turbocharger protection					
C_TQI_MIN_TCHA_PROT	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
minimum torque value for all torque values of all turbo charger protections					
C_TQI_N_TCHA_DIF_THD	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Indicated torque threshold for turbo charger protection deactivation					
C_TQI_N_TCHA_MAX_I_MIN	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
minimum torque value for I-part of turbo charger overheating protection					
C_TQI_N_TCHA_MAX_MIN	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
minimum indicated torque value for turbo charger protection					
C_TQI_TCHA_PROT_AS	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Application system value for TQI_TCHA_PROT					
LC_TQI_SP_TCHA_PROT_REF_INI	1	0...1H	0...1	1	-
Selection of torque setpoint reference at transition LV_TQI_N_TCHA_MAX 0 to 1					
LC_TQI_TCHA_PROT_AS	1	0...1H	0...1	1	-
Selection of TQI_TCHA_PROT value via Application System					
IP_N_TCHA_MAX_SP_DIF_ACT	6	0...FFFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
LDP_N_TCHA_GRD_IP_N_TCHA_SP_ACT	6	0...FFFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
Charger speed setpoint activation threshold for charger speed gradient based torque limitation					
IP_FAC_TQI_N_TCHA_GRD_MAX	6x4	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_N_TCHA_GRD_IP_TQI_N_TCHA	6	0...FFFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
LDP_N_TCHA_DIF_IP_TQI_N_TCHA	4	0...FFFFH	-2E+5 ... 1.99994E+5	6.10351563	rpm
indicated torque setpoint reduction factor for turbocharger protection (turbocharger speed gradient)					

9.79.1 CHRG_REQGNPROT0

FUNCTION DESCRIPTION:

To protect the turbo charger from overspeed (N_TCHA) or in case of:

failure detected on the wastegate

failure detected on the recirculation valve (only if “RCL close” failure state detected)

a torque reduction can be applied.

In case those torque reductions do not show the wanted effect, there are constants for absolute maximum (not applied in case of RCL failure) which cause an immediate complete opening of the wastegate or an immediate torque setpoint reduction.

General information:

The physical limits are the maximum turbine temperature and the turbo charger rotational speed. They depend on the turbine and housing material and usually are among 900°C to 1050°C respectively 160 000rpm to 250 000rpm. The first way to decrease the exhaust gas temperature without loosing engine torque is to enrich the air/fuel ratio. Second step is to decrease the torque request and third step is to open the bypass (waste gate) of the turbine immediately. The first two measures are located in EXTC, the third in the present module. Decreasing the torque setpoint and opening the waste gate are also appropriate solutions if too high rotational speed of the turbo charger is detected.

Decreasing the torque setpoint is also appropriate solution if a failure is detected on the wastegate or on the recirculation valve (remark: for the recirculation valve, it is necessary to decrease torque only if “RCL close” failure state is detected). .

Function description:

The variable N_TCHA_MAX_DIF is the input into an PI-controller. If N_TCHA > C_N_TCHA_MAX_SP the TQI_N_TCHA_MAX_I is ramped down to decrease the turbo charger speed. In case of too high turbo charger speed gradients the temporary torque reduction value is decreased again by multiplying with an turbo charger speed gradient dependent factor. If the turbo charger speed finally drops below the calibratable threshold C_N_TCHA_MAX_SP then TQI_N_TCHA_MAX_I is ramped up again until TQI_ADD_ACT is reached. If TQI_ADD_ACT is reached, TQI_N_TCHA_MAX is set to its passive value of 7FFFh 1023.97Nm again.

The value of the torque request cannot underspend the calibratable constant C_TQI_MIN_TCHA_PROT.

Function Description

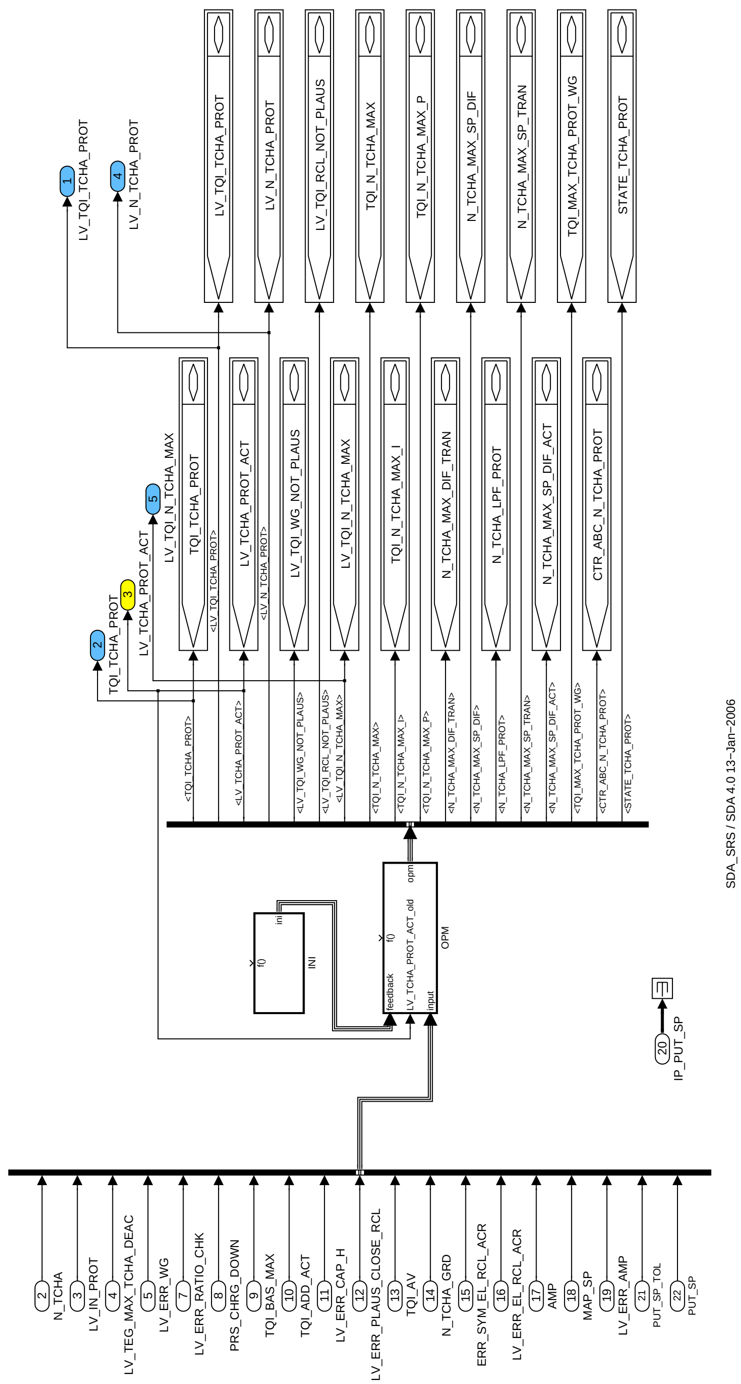


Figure 190 CHRG_REQGNPROT0

9.79.1.1 SUBFUNCTION: INI

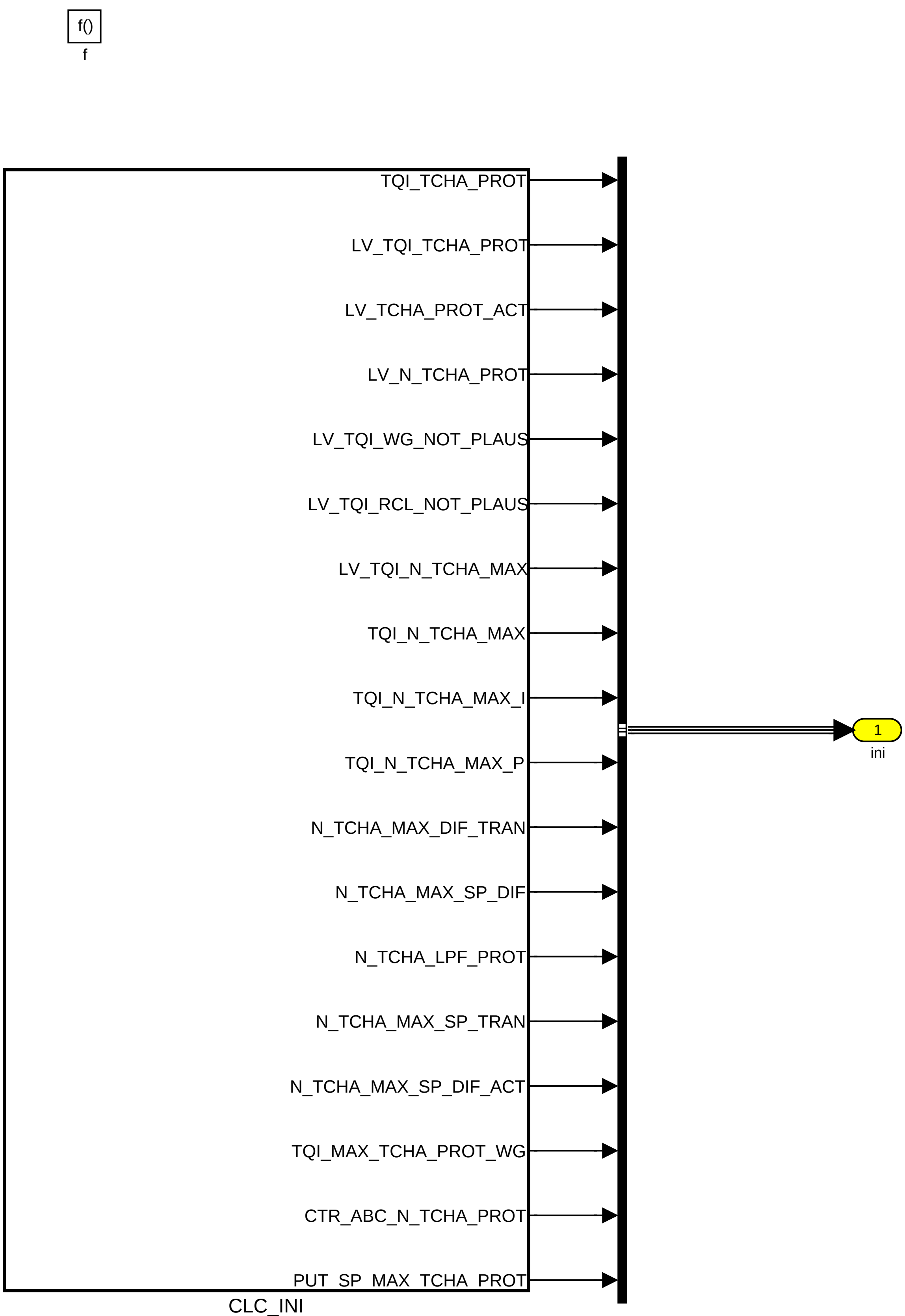

f()
f
TQI_TCHA_PROT
LV_TQI_TCHA_PROT
LV_TCHA_PROT_ACT
LV_N_TCHA_PROT
LV_TQI_WG_NOT_PLAUS
LV_TQI_RCL_NOT_PLAUS
LV_TQI_N_TCHA_MAX
TQI_N_TCHA_MAX
TQI_N_TCHA_MAX_I
TQI_N_TCHA_MAX_P
N_TCHA_MAX_DIF_TRAN
N_TCHA_MAX_SP_DIF
N_TCHA_LPF_PROT
N_TCHA_MAX_SP_TRAN
N_TCHA_MAX_SP_DIF_ACT
TQI_MAX_TCHA_PROT_WG
CTR_ABC_N_TCHA_PROT
PUT_SP_MAX_TCHA_PROT
CLC_INI
1
ini

Figure 191 CHRG_REQGNPROT0/ INI

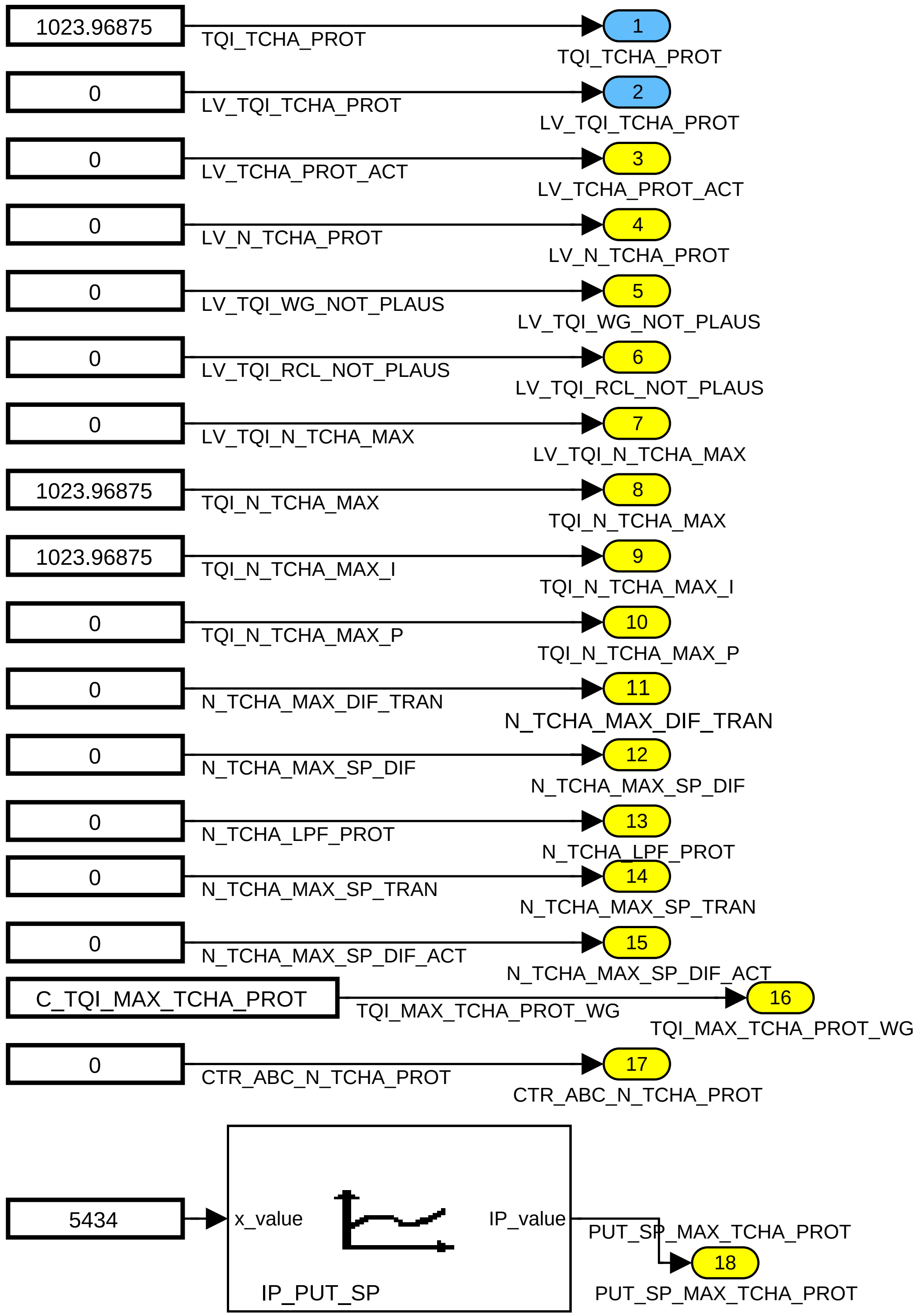


Figure 192 CHRG_REQGNPROT0/ INI/ CLC_INI

9.79.1.2 SUBFUNCTION: OPM

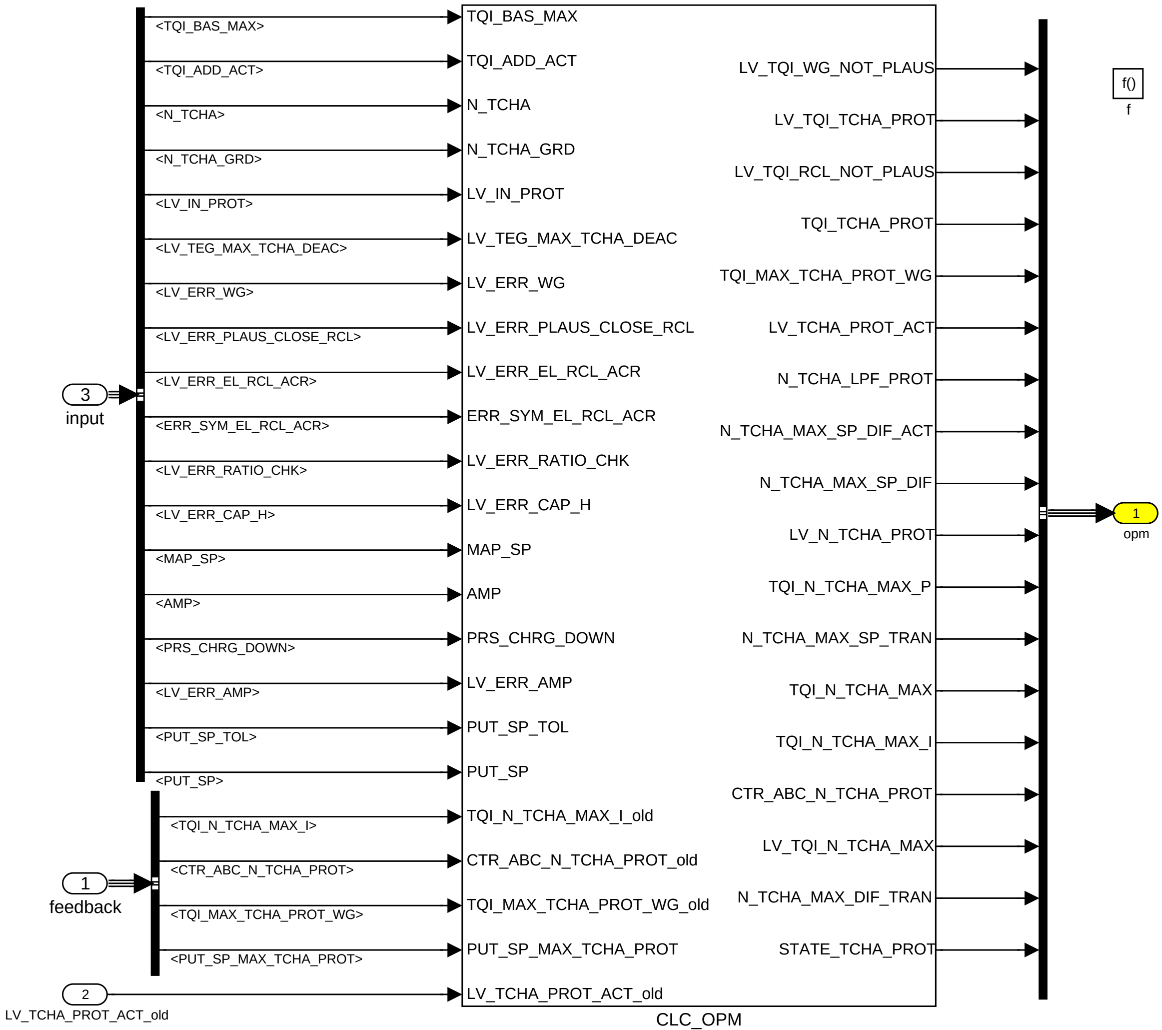


Figure 193 CHRG_REQGNPROT0/ OPM

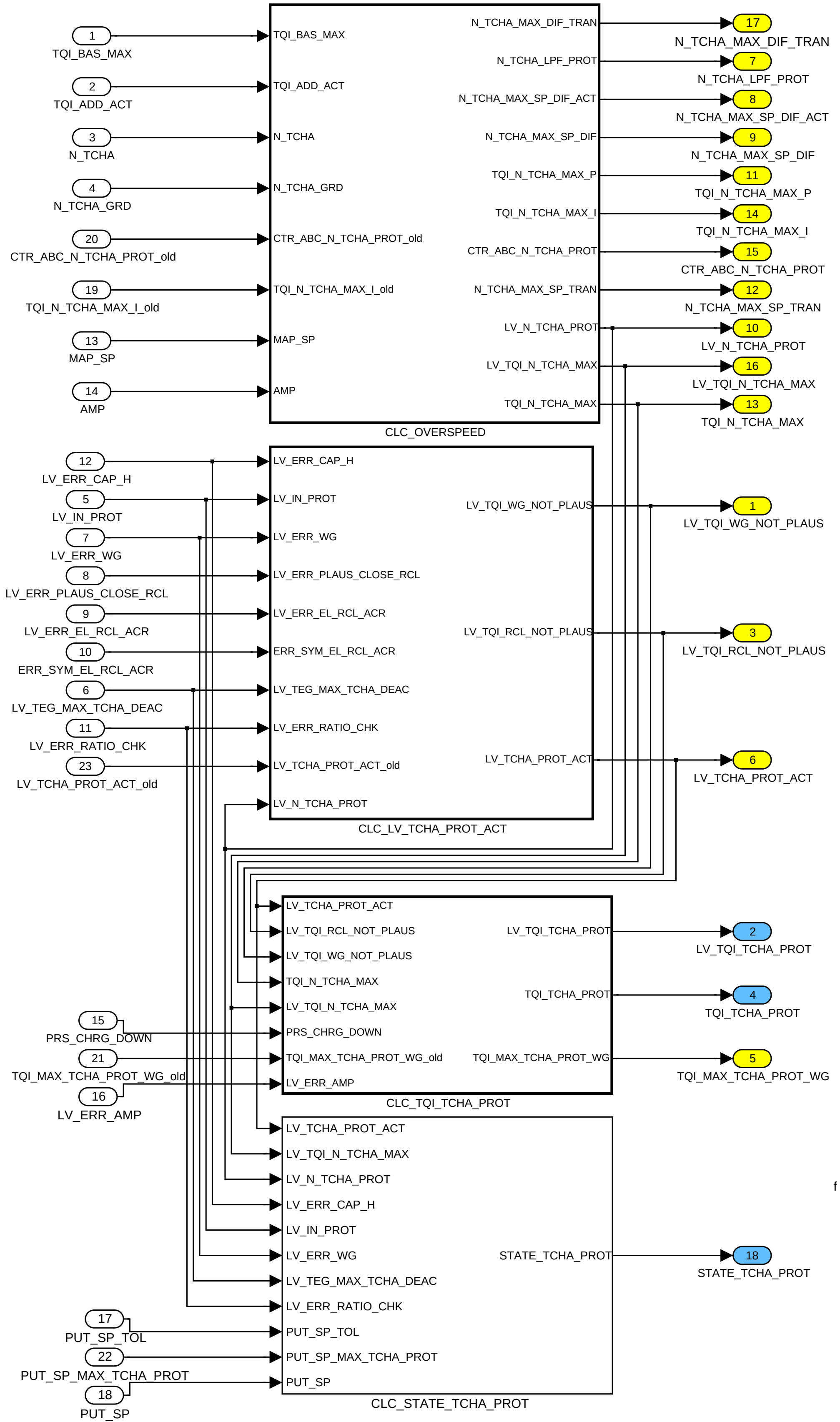

1
TQI_BAS_MAX
2
TQI_ADD_ACT
3
N_TCHA
4
N_TCHA_GRD
20
CTR_ABC_N_TCHA_PROT_old
19
TQI_N_TCHA_MAX_I_old
13
MAP_SP
14
AMP
TQI_BAS_MAX
TQI_ADD_ACT
N_TCHA
N_TCHA_GRD
CTR_ABC_N_TCHA_PROT_old
TQI_N_TCHA_MAX_I_old
MAP_SP
AMP
N_TCHA_MAX_DIF_TRAN
N_TCHA_LPF_PROT
N_TCHA_MAX_SP_DIF_ACT
N_TCHA_MAX_SP_DIF
TQI_N_TCHA_MAX_P
TQI_N_TCHA_MAX_I
CTR_ABC_N_TCHA_PROT
N_TCHA_MAX_SP_TRAN
LV_N_TCHA_PROT
LV_TQI_N_TCHA_MAX
TQI_N_TCHA_MAX
CLC_OVERSPEED
17
N_TCHA_MAX_DIF_TRAN
7
N_TCHA_LPF_PROT
8
N_TCHA_MAX_SP_DIF_ACT
9
N_TCHA_MAX_SP_DIF
11
TQI_N_TCHA_MAX_P
14
TQI_N_TCHA_MAX_I
15
CTR_ABC_N_TCHA_PROT
12
N_TCHA_MAX_SP_TRAN
10
LV_N_TCHA_PROT
16
LV_TQI_N_TCHA_MAX
13
TQI_N_TCHA_MAX
12
LV_ERR_CAP_H
5
LV_IN_PROT
7
LV_ERR_WG
8
LV_ERR_PLAUS_CLOSE_RCL
9
LV_ERR_EL_RCL_ACR
10
ERR_SYM_EL_RCL_ACR
6
LV_TEG_MAX_TCHA_DEAC
11
LV_ERR_RATIO_CHK
23
LV_TCHA_PROT_ACT_old
LV_ERR_CAP_H
LV_IN_PROT
LV_ERR_WG
LV_ERR_PLAUS_CLOSE_RCL
LV_ERR_EL_RCL_ACR
ERR_SYM_EL_RCL_ACR
LV_TEG_MAX_TCHA_DEAC
LV_ERR_RATIO_CHK
LV_TCHA_PROT_ACT_old
LV_N_TCHA_PROT
LV_TQI_WG_NOT_PLAUS
LV_TQI_RCL_NOT_PLAUS
LV_TCHA_PROT_ACT
CLC_LV_TCHA_PROT_ACT
1
LV_TQI_WG_NOT_PLAUS
3
LV_TQI_RCL_NOT_PLAUS
6
LV_TCHA_PROT_ACT
LV_TCHA_PROT_ACT
LV_TQI_RCL_NOT_PLAUS
LV_TQI_WG_NOT_PLAUS
TQI_N_TCHA_MAX
LV_TQI_N_TCHA_MAX
PRS_CHRG_DOWN
TQI_MAX_TCHA_PROT_WG_old
LV_ERR_AMP
LV_TQI_TCHA_PROT
TQI_TCHA_PROT
TQI_MAX_TCHA_PROT_WG
CLC_TQI_TCHA_PROT
15
PRS_CHRG_DOWN
21
TQI_MAX_TCHA_PROT_WG_old
16
LV_ERR_AMP
2
LV_TQI_TCHA_PROT
4
TQI_TCHA_PROT
5
TQI_MAX_TCHA_PROT_WG
LV_TCHA_PROT_ACT
LV_TQI_N_TCHA_MAX
LV_N_TCHA_PROT
LV_ERR_CAP_H
LV_IN_PROT
LV_ERR_WG
LV_TEG_MAX_TCHA_DEAC
LV_ERR_RATIO_CHK
PUT_SP_TOL
PUT_SP_MAX_TCHA_PROT
PUT_SP
STATE_TCHA_PROT
CLC_STATE_TCHA_PROT
17
PUT_SP_TOL
22
PUT_SP_MAX_TCHA_PROT
18
PUT_SP
18
STATE_TCHA_PROT
f

Figure 194 CHRG_REQGNPROT0/ OPM/ CLC_OPM

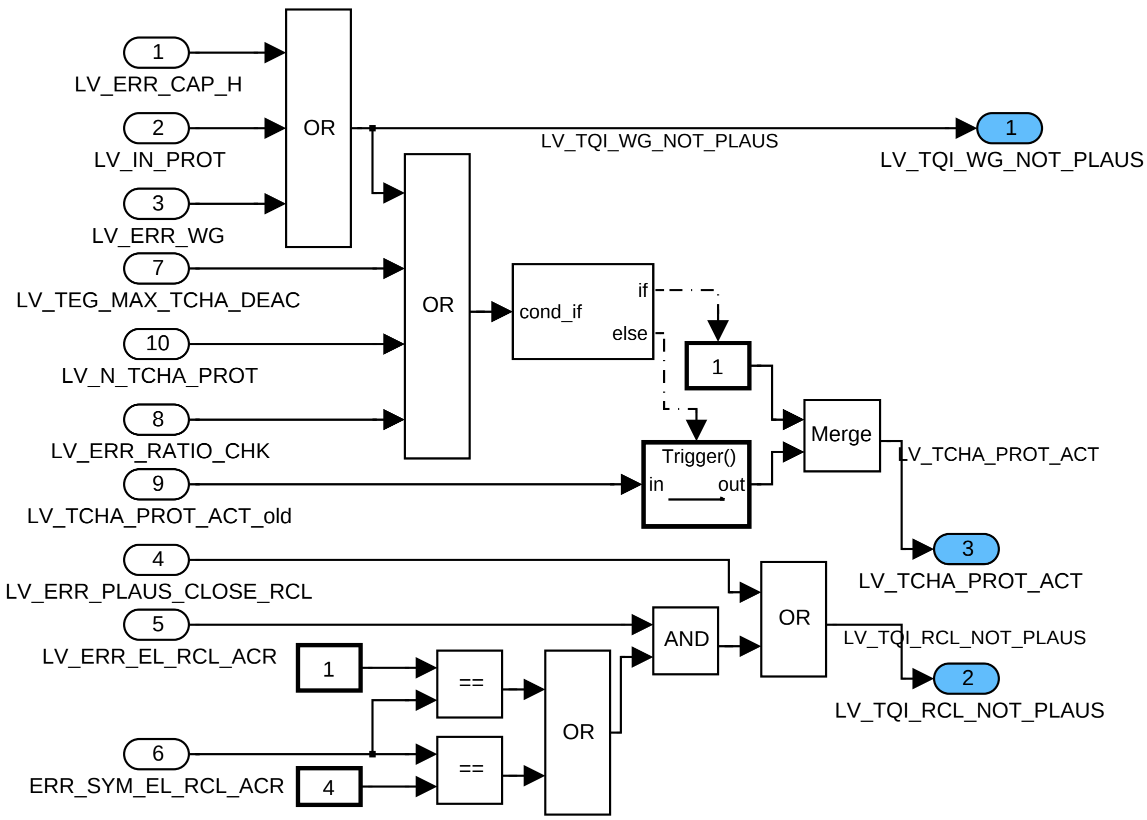


Figure 195 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_LV_TCHA_PROT_ACT

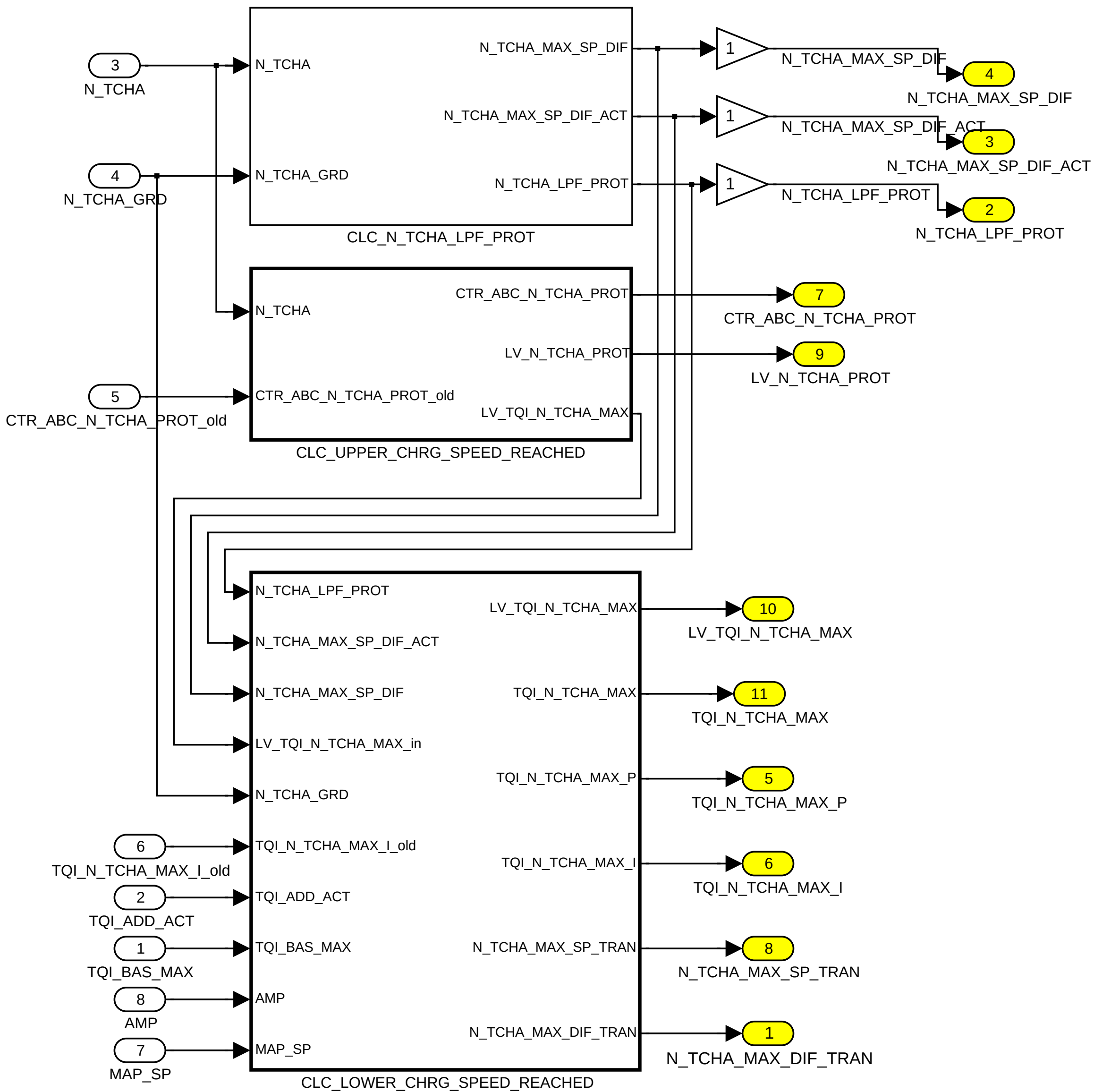


Figure 196 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED

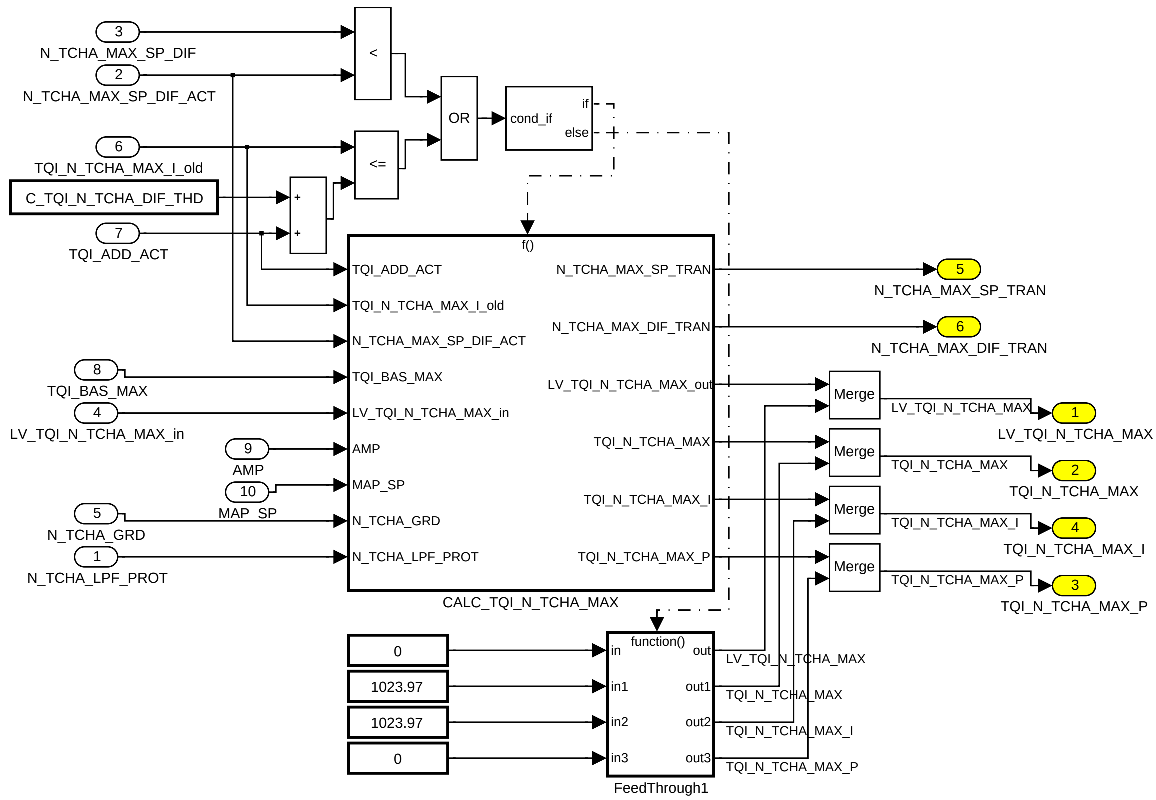


Figure 197 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_LOWER_CHRG_SPEED_REACHED

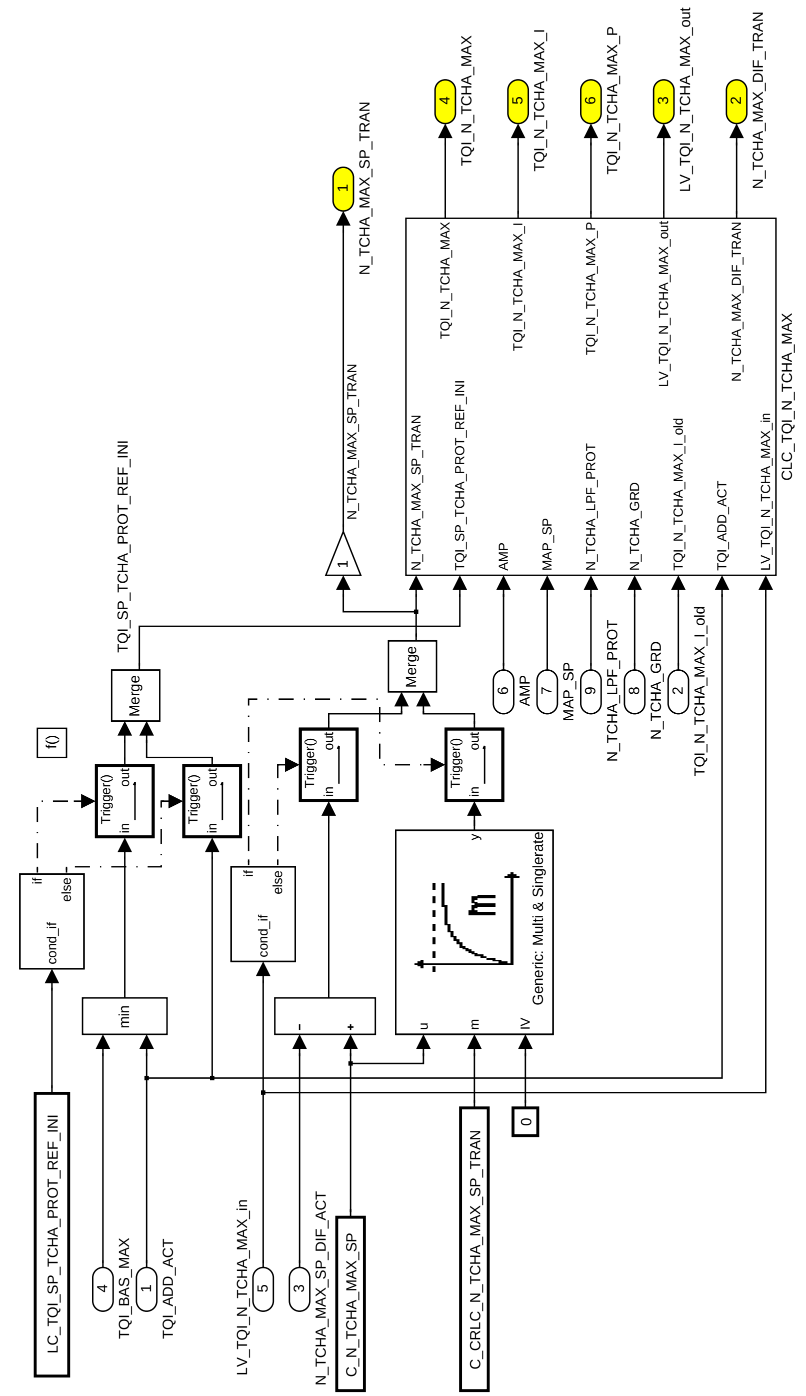


Figure 198 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_LOWER_CHRG_SPEED_REACHED/ CALC_TQI_N_TCHA_MAX

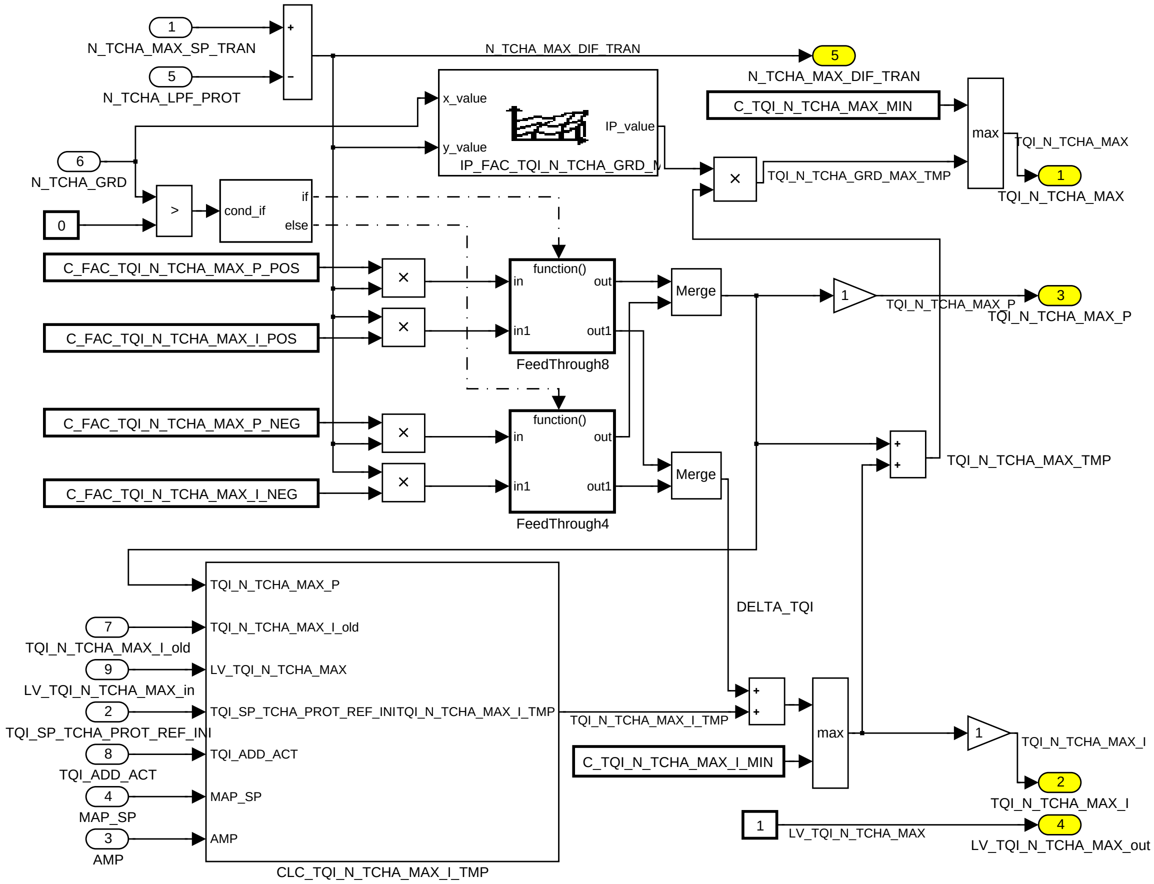


Figure 199 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_LOWER_CHRG_SPEED_REACHED/ CALC_TQI_N_TCHA_MAX/ CLC_TQI_N_TCHA_MAX

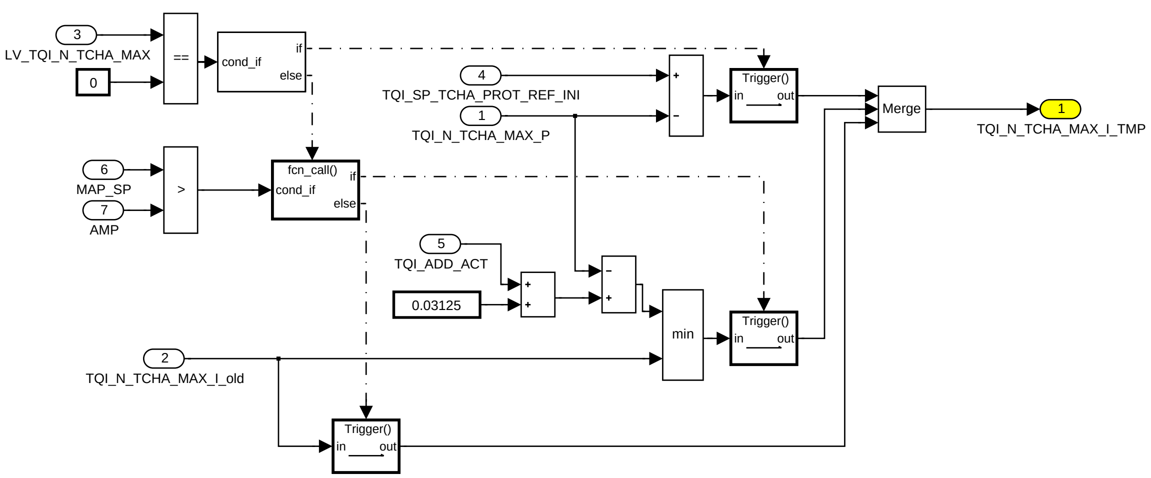


Figure 200 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_LOWER_CHRG_SPEED_REACHED/ CALC_TQI_N_TCHA_MAX/ CLC_TQI_N_TCHA_MAX/ CLC_TQI_N_TCHA_MAX_I_TMP

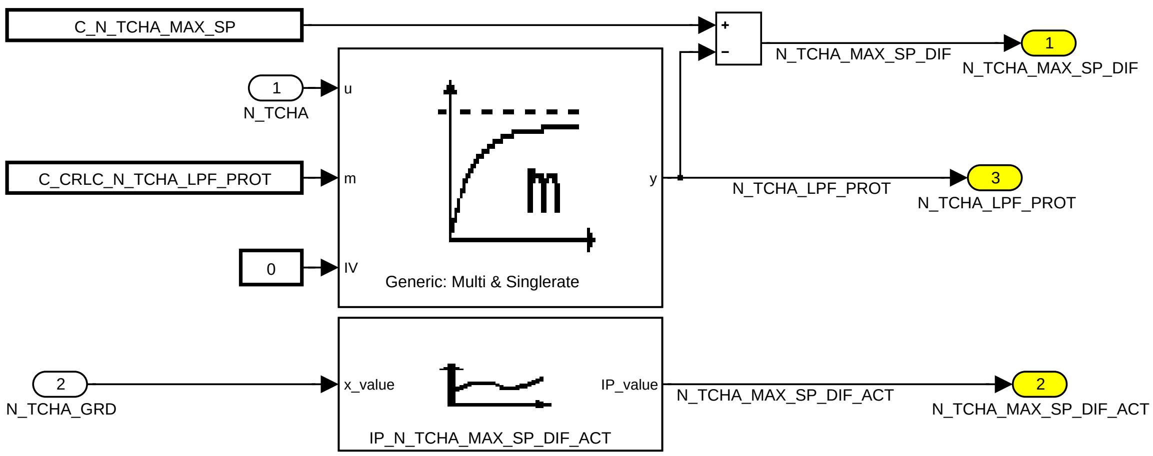


Figure 201 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_N_TCHA_LPF_PROT

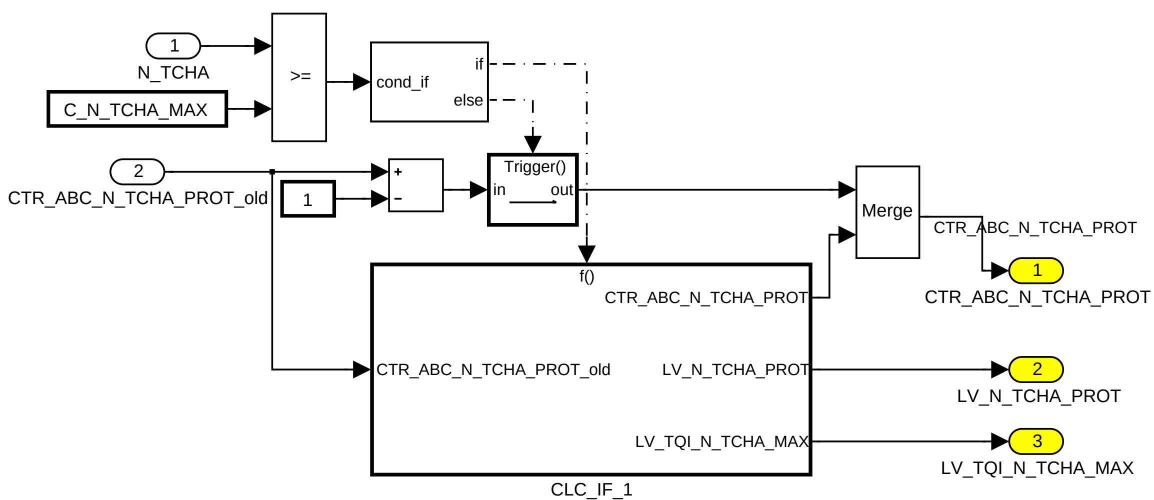


Figure 202 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_UPPER_CHRG_SPEED_REACHED

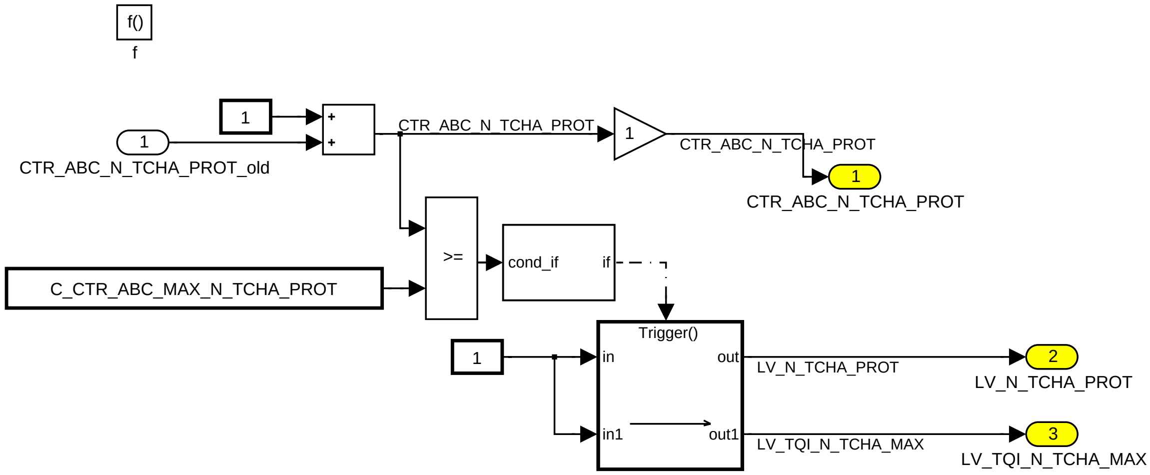


Figure 203 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_OVERSPEED/ CLC_UPPER_CHRG_SPEED_REACHED/ CLC_IF_1

CHRG_REQGNPROT0/OPM/CLC_OPM/CLC_STATE_TCHA_PROT

STATE_TCHA_PROT collects all the information about limited turbo charger performance because of extreme operating conditions. Each limit corresponds to one bit of STATE_TCHA_PROT:

Bit 15 = 32768 = Permanent torque limitation because of load-TPS-error

Bit 14 = 16384 = Permanent torque limitation because of too high exhaust gas temperature

Bit 13 = 8192 = Permanent torque limitation because of wastegate driver error

Bit 12 = 4096 = Permanent torque limitation because of too high charger speed

Bit 11 = 2048 = Permanent torque limitation because of charger control/pressure implausibility

Bit 10 = 1024 = Any permanent torque limitation

Bit 9 = 512 = Temporary torque limitation because of high charger speed

Bit 8 = 256 = Operation at maximum charger pressure ratio

Bit 7 = 128 = Operation at maximum absolute charge air pressure setpoint

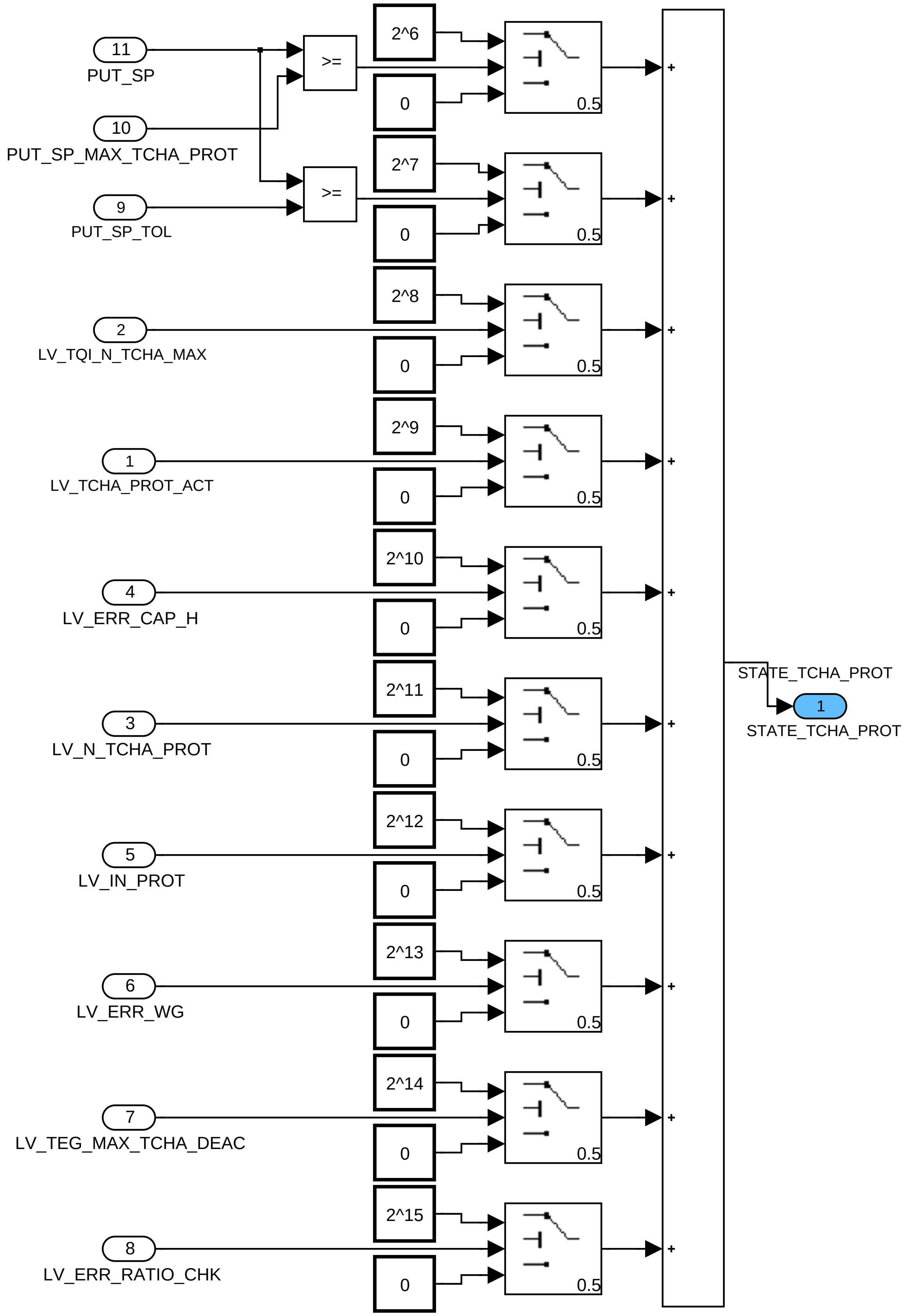

11
PUT_SP
10
PUT_SP_MAX_TCHA_PROT
9
PUT_SP_TOL
>=
>=
2
LV_TQI_N_TCHA_MAX
1
LV_TCHA_PROT_ACT
4
LV_ERR_CAP_H
3
LV_N_TCHA_PROT
5
LV_IN_PROT
6
LV_ERR_WG
7
LV_TEG_MAX_TCHA_DEAC
8
LV_ERR_RATIO_CHK
2^6
0
2^7
0
2^8
0
2^9
0
2^10
0
2^11
0
2^12
0
2^13
0
2^14
0
2^15
0
0.5
STATE_TCHA_PROT
1
STATE_TCHA_PROT

Figure 204 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_STATE_TCHA_PROT

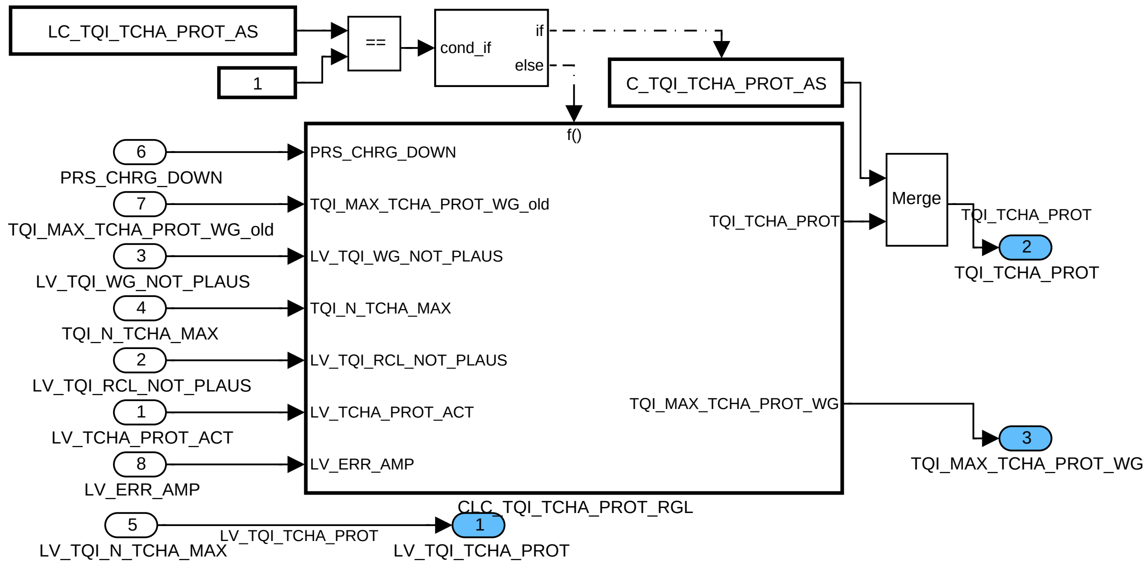


Figure 205 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_TQI_TCHA_PROT

general specification

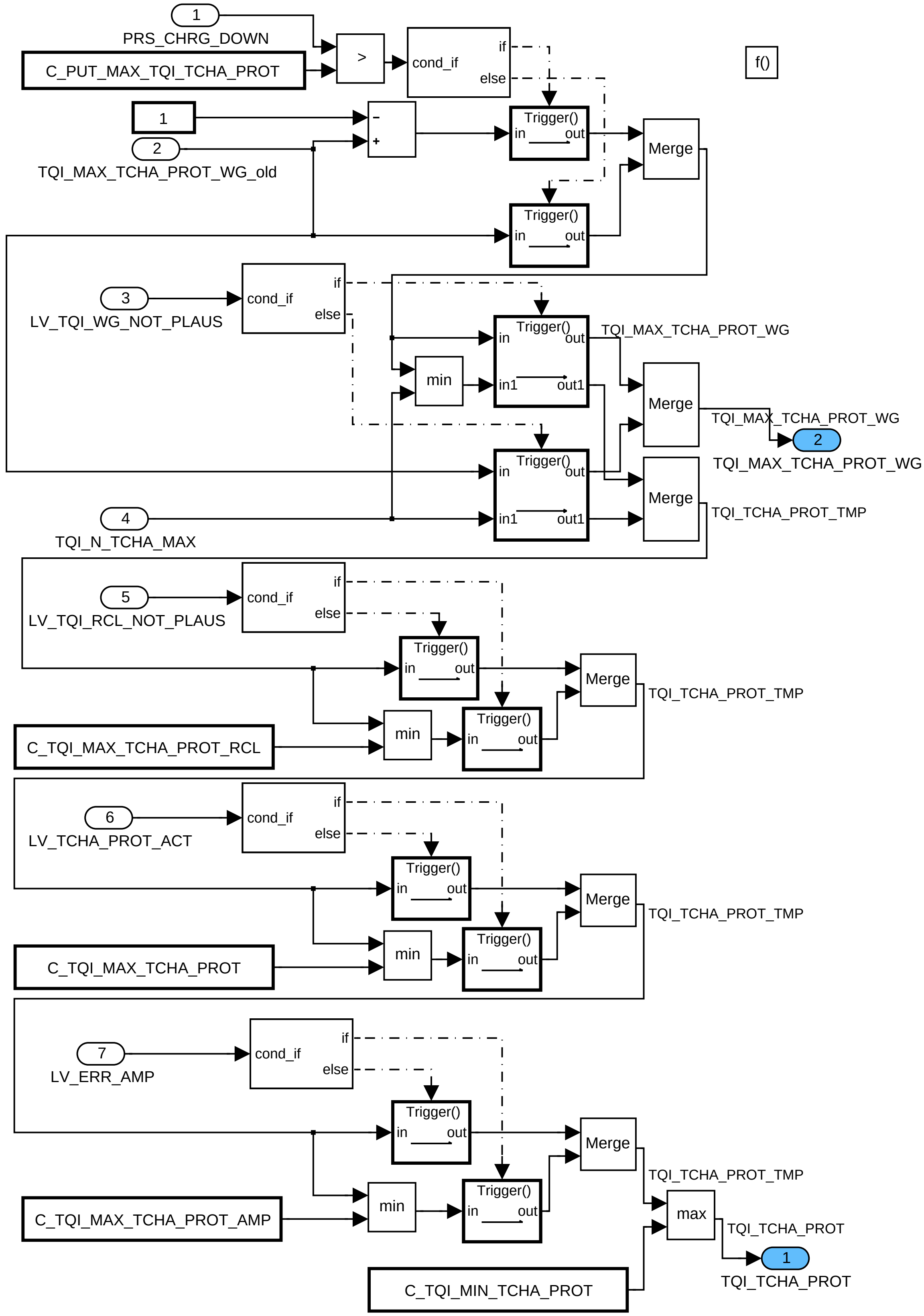


Figure 206 CHRG_REQGNPROT0/ OPM/ CLC_OPM/ CLC_TQI_TCHA_PROT/ CLC_TQI_TCHA_PROT_RGL

9.80 Recirculation actuator Setpoint Determination: PSN_RCL_SP

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FAC_PQ_CHA_SURGE_DL	V	0... FFFFH	0... 3.99993896484	61.0352e-6	[-]
Global corrective factor for surge line definition					
LV_RCL_CLOSE_GRD	O/V	0... 1H	0... 1	1	[-]
Flag for RCL actuator closing request: PV_GRD_RCL or TPS_SP_GRD_RCL dependancy					
LV_RCL_OPEN_REQ	V	0... 1H	0... 1	1	[-]
Recirculation actuator opening request					
LV_RCL_OPEN_REQ_GRD_RCL	V	0... 1H	0... 1	1	[-]
Recirculation actuator opening request: PV_AV_GRD_RCL or TPS_SP_GRD_RCL criteria					
LV_RCL_OPEN_REQ_SURGE	V	0... 1H	0... 1	1	[-]
Recircultation actuator opening request to avoid surge					
LV_RCL_OPEN_REQ_SURGE_ACT	V	0... 1H	0... 1	1	[-]
Recirculation actuator opening request to avoid surge after switch on / off delay					
LV_STATE_IS_RCL	V	0... 1H	0... 1	1	[-]
Idle information for RCL (with delayed reset)					
LV_SURGE	O/V	0... 1H	0... 1	1	[-]
logical bit to display surge line definition					
LV_SURGE_PRED	O/V	0... 1H	0... 1	1	[-]
logical bit to display surge line - predicted definition (based on RCL actuator response time)					
N_TCHA_RCL_CLOSE_INI	V	0... FFFFH	0... 400000	6.1036088	[rpm]
Charger speed at begin of recirculation actuator closing					
PQ_CHA_DIF	V	8000... 7FFFH	-8... 7.99975585938	244.141e-6	[-]
PQ_CHA reference variation					
PQ_CHA_DIF_FIL	V	8000... 7FFFH	-8... 7.99975585938	244.141e-6	[-]
Filtered value of PQ_CHA_DIF					
PQ_CHA_PRED	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Predicted value of PQ_CHA during the prediction time					
PQ_CHA_SURGE_DL	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio at charger surge line limit (component characteristic)					
PQ_CHA_SURGE_DL_HYS	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
hysteresis value for end of surge definition					
PQ_CHA_SURGE_DL_PRED	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio at charger surge limit - predicted definition - (component characteristic)					
PSN_RCL_SP	O/V	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Recirculation actuator position setpoint					
PV_AV_GRD_RCL	O	80... 7FH	-2500... 2480.46875	19.53125	[%/s]
Pedal gradient value used for RCL closing definition request					
PV_GRD_RCL	O/V	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Pedal gradient value used for RCL setpoint determination					
PV_GRD_RCL_OPEN_THD	V	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
PV_GRD_RCL threshold for LV_RCL_OPEN_REQ_GRD_RCL determination					
T_RCL_CLOSE_DLY_SURGE	V	0... FFH	0... 2550	10	[ms]

Delay time for activation of closing of recirculation actuator					
T_RCL_OPEN_DLY_SURGE	V	0... FFH	0... 2550	10	[ms]
Delay time for activation of opening of recirculation actuator					
T_RESP_RCL	V	0... FFH	0... 2550	10	[ms]
RCL actuator response time (opening / closing)					
TPS_SP_GRD_RCL_OPEN_THD	V	8000... 7FFFH	-2987.5... 2987.41	0.0911713	[°TPS/s]
TPS_SP_GRD_RCL threshold for LV_RCL_OPEN_REQ_GRD_RCL determination					
VOL_FLOW_CHA_UP_RED_DIF	V	8000... 7FFFH	-1... 0.99996948242	30.5176e-6	[m3/s]
VOL_FLOW_CHA_UP_RED_MV reference variation					
VOL_FLOW_CHA_UP_RED_DIF_FIL	V	8000... 7FFFH	-1... 0.99996948242	30.5176e-6	[m3/s]
Filtered value of VOL_FLOW_CHA_UP_RED_DIF					
VOL_FLOW_CHA_UP_RED_PRED	V	0... FFFFH	0... 1.99996948242	30.5176e-6	[m3/s]
Predicted value of VOL_FLOW_CHA_UP_RED_MV during the prediction time					

Input Data:

PRS_RCL_UP_MV	PRS_EPC_RCL_DOWN	PRS_CHA_UP_MV	VB
LV_CMD_RCL_OPEN	PQ_CHA_MV	VOL_FLOW_CHA_UP_RED_MV	TIA_CHA_UP
LV_IS	TPS_AV	N_32	LV_IN_PROT
LV_PU	LV_PUC	LV_ACT_RCL_EXT_ADJ	LV_RCL_EXT_ADJ
LC_TCHA_CONF	N_TCHA	LV_TCHA_PROT_ACT	TPS_SP_MDL_MAX
TPS_SP_MDL	PWM_WG	LV_PV_RCL_ACT	PV_RCL
LV_CLU_OPEN	LV_N_MAX	AMP	MAP

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_AMP_MAP_DIF_RCL_OPEN	1	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Pressure threshold to open recirculation valve in non charged engine operation					
C_PQ_CHA_DIF_MIN	1	8000... 7FFFH	-8... 7.99975585938	244.141e-6	[-]
Minimum necessary PQ_CHA increase for PQ_CHA_DIF_FIL det.					
C_PQ_CHA_RCL_OPEN_SET	1	8000... 7FFFH	-8... 7.99975585938	244.141e-6	[-]
Describes the threshold, till which pressure quotient over charger the RCL actuator opening for an improved acceleration behavior is active					
C_PSN_RCL_SP_AS	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Applicaton system value for recirculation actuator position setpoint					
C_PV_GRD_RCL_CLOSE_THD_NEG	1	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Threshold for RCL actuator closing based on PV_GRD_RCL parameter (deceleration)					
C_PV_GRD_RCL_CLOSE_THD_POS	1	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Threshold for RCL actuator closing based on PV_GRD_RCL parameter (acceleration)					
C_PV_GRD_RCL_OPEN_HYS	1	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Hysteresis on PV_GRD_RCL for LV_RCL_OPEN_REQ_GRD_RCL determination					
C_PV_GRD_RCL_OPEN_RST	1	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Threshold to reset RCL actuator opening for an improved acceleration behavior, based on PV_GRD_RCL parameter					

C_PV_GRD_RCL_OPEN_SET	1	8000... 7FFFH	-2500... 2499.9237	0.0762939	[%/s]
Threshold to set RCL actuator opening for an improved acceleration behavior, based on PV_GRD_RCL parameter					
C_PV_MIN_RCL_CLOSE_GRD_PV	1	0... 3FFH	0... 99.90234375	0.0976563	[%]
Minimum pedal value to close recirculation valve via PV gradient					
C_PV_MIN_RCL_CLOSE_GRD_TPS	1	0... 3FFH	0... 99.90234375	0.0976563	[%]
Minimum pedal value to close recirculation valve via TPS gradient					
C_PV_RCL_CLOSE_BOL	1	0... 3FFH	0... 99.90234375	0.0976563	[%]
Pedal value bottom threshold to close recirculation valve					
C_PV_RCL_OPEN_MAX	1	0... 3FFH	0... 99.90234375	0.0976563	[%]
Maximum pedal value to open recirculation valve					
C_T_IS_RCL_MAX	1	0... FFH	0... 2550	10	[ms]
Delay between LV_IS and LV_STATE_IS_RCL					
C_T_RCL_OPEN_PV_DLY_SURGE	1	0... FFH	0... 2550	10	[ms]
Delay time for activation of opening of recirculation actuator					
C_T_RCL_OPEN_TPS_DLY_SURGE	1	0... FFH	0... 2550	10	[ms]
Delay time for activation of opening of recirculation actuator					
C_T_RESP_RCL_OFF_AS	1	0... FFH	0... 2550	10	[ms]
T_RESP_RCL_OFF value via application system					
C_T_RESP_RCL_ON_AS	1	0... FFH	0... 2550	10	[ms]
T_RESP_RCL_ON value via application system					
C_TPS_AV_RCL_DIF_MIN	1	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
Minimum difference between max TPS_SP and current TPS_AV to allow RCL valve opening					
C_TPS_SP_GRD_RCL_CLOSE_THD_NEG	1	8000... 7FFFH	-2987.5... 2987.41	0.0911713	[°TPS/s]
Threshold limit for RCL actuator closing based on TPS_SP_GRD_RCL parameter (deceleration)					
C_TPS_SP_GRD_RCL_CLOSE_THD_POS	1	8000... 7FFFH	-2987.5... 2987.41	0.0911713	[°TPS/s]
Threshold limit for RCL actuator closing based on TPS_SP_GRD_RCL parameter (acceleration)					
C_TPS_SP_GRD_RCL_OPEN_HYS	1	8000... 7FFFH	-2987.5... 2987.41	0.0911713	[°TPS/s]
Hysteresis on TPS_SP_GRD_RCL for LV_RCL_OPEN_REQ_GRD_RCL determination					
C_VOL_FLOW_CHA_UP_RED_DIF_MAX	1	8000... 7FFFH	-1... 0.99996948242	30.5176e-6	[m3/s]
Maximum necessary VOL_FLOW_CHA_UP_RED decrease for VOL_FLOW_CHA_UP_RED_DIF_FIL det.					
IP_FAC_PQ_CHA_DIF	4	0... FFH	0... 0.99609375	3.90625e-3	[-]
LDP_PQ_CHA_MV_IP_FAC_PQ_CHA_DIF	4	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
ponderation factor for PQ_CHA_DIF determination					
IP_FAC_PQ_CHA_SURGE_DL_PRS	6	0... FFFFH	0... 3.99993896484	61.0352e-6	[-]
LDPM_PRS_CHA_UP_MV_1_CHRG	6	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure corrective factor on surge line					
IP_FAC_PQ_CHA_SURGE_DL_TEMP	6	0... FFFFH	0... 3.99993896484	61.0352e-6	[-]
LDP_TIA_CHA_UP_IP_FAC_PQ_CHA	6	0... FEH	-48... 142.5	0.75	[°C]
Pressure corrective factor on surge line					
IP_FAC_VOL_FLOW_CHA_UP_RED_DIF	4	0... FFH	0... 0.99609375	3.90625e-3	[-]
LDP_VOL_FLOW_CHA_UP_RED_IP_FAC	4	0... FFFFH	0... 1.99996948242	30.5176e-6	[m3/s]
ponderation factor for VOL_FLOW_CHA_UP_RED_DIF determination					
IP_PQ_CHA_SURGE_DL	10	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_VOL_FLOW_CHA_UP_RED_3_CHRG	10	0... FFFFH	0... 1.99996948242	30.5176e-6	[m3/s]
Surge line definition (component characteristic) for LV_SURGE detection					
IP_PQ_CHA_SURGE_DL_HYS_IS	6*10	0... FFFFH	0... 15.9997558594	244.141e-6	[-]

LDP_VOL_FLOW_CHA_UP_DIF_1_CHRG	10	0... FFFFH	-1... 0.99996948242	30.5176e-6	[m3/s]
LDPM_PQ_CHA_DIF_1_CHRG	6	0... FFFFH	-8... 7.99975585938	244.141e-6	[-]
Hysteresis value for end of surge definition - in idle					
IP_PQ_CHA_SURGE_DL_HYS_PL	10	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_VOL_FLOW_CHA_UP_RED_3_CHRG	10	0... FFFFH	0... 1.99996948242	30.5176e-6	[m3/s]
Hysteresis value for end of surge definition - in part load					
IP_PV_GRD_RCL_OPEN_THD	6	0... FFFFH	-2500... 2499.9237	0.0762939	[%/s]
LDPM_N_32_2_CHRG	6	0... FFH	0... 8160	32	[rpm]
PV_GRD threshold limit for RCL actuator opening					
IP_T_RCL_CLOSE_DLY_SURGE	6	0... FFH	0... 2550	10	[ms]
LDP_N_TCHA_IP_T_RCL_CLOSE_DLY	6	0... FFFFH	0... 400000	6.1036088	[rpm]
Minimum opening time of recirculation actuator					
IP_T_RESP_RCL_OFF	10*12	0... FFH	0... 255	1	[ms]
LDPM_PRS_RCL_UP_MV_1_CHRG	12	0... 7FFFH	0... 2716.91708374	0.0829163	[hPa]
LDPM_PRS_EPC_RCL_DOWN_1_CHRG	10	0... FFFFH	0... 5434	0.0829175	[hPa]
Basic mapping for T_RESP_RCL_OFF determination					
IP_T_RESP_RCL_OFF_FAC_PRS	6	0... FFH	0... 3.984375	0.015625	[-]
LDPM_PRS_CHA_UP_MV_1_CHRG	6	0... FFFFH	0... 5434	0.0829175	[hPa]
Factor on T_RESP_RCL_OFF versus PRS_CHA_UP_MV					
IP_T_RESP_RCL_ON	10*12	0... FFH	0... 255	1	[ms]
LDPM_PRS_RCL_UP_MV_1_CHRG	12	0... 7FFFH	0... 2716.91708374	0.0829163	[hPa]
LDPM_PRS_EPC_RCL_DOWN_1_CHRG	10	0... FFFFH	0... 5434	0.0829175	[hPa]
Basic mapping for T_RESP_RCL_ON determination					
IP_T_RESP_RCL_ON_FAC_PRS	6	0... FFH	0... 3.984375	0.015625	[-]
LDPM_PRS_CHA_UP_MV_1_CHRG	6	0... FFFFH	0... 5434	0.0829175	[hPa]
Factor on T_RESP_RCL_ON versus PRS_CHA_UP_MV					
IP_T_RESP_RCL_ON_FAC_VB	6	0... FFH	0... 1.9921875	7.8125e-3	[-]
LDP_VB_IP_T_RESP	6	0... FFH	0... 26	0.1019608	[V]
Factor on T_RESP_RCL_ON versus VB					
IP_TPS_SP_GRD_RCL_OPEN_THD	6	0... FFFFH	-2987.5... 2987.408829	0.0911713	[°TPS/s]
LDPM_N_32_2_CHRG	6	0... FFH	0... 8160	32	[rpm]
TPS_SP_GRD_RCL threshold limit for RCL actuator opening					
LC_PSN_RCL_SP_AS	1	0... 1H	0... 1	1	[-]
LC_PSN_RCL_SP_AS = 1: the position of the recirculation actuator can be fixed by application system					
LC_RCL_CLOSE_GRD_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of LV_RCL_CLOSE_GRD effect for RCL setpoint determination					
LC_RCL_GRD_SURGE_PRED_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of throttle or pedal gradient influence on surge prediction					
LC_RCL_OPEN_CS_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of opening of the recirculation actuator caused by LV_CS					
LC_RCL_OPEN_N_MAX_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of opening of the recirculation actuator caused by LV_N_MAX					
LC_RCL_OPEN_PU_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of opening of the recirculation actuator caused by LV_PU or LV_PUC					
LC_SURGE_PRED_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of the possibility of activation on LV_SURGE_PRED					
LC_SURGE_SUPP	1	0... 1H	0... 1	1	[-]
Suppression of the possibility of activation on LV_SURGE					
LC_T_RESP_RCL_AS	1	0... 1H	0... 1	1	[-]
T_RESP_RCL_ON of T_RESP_RCL_OFF selection via application system					

General Information

This module is valid for a binary recirculation actuator. The controller must be "Digital type" (ON/OFF) (i.e.: NC_PSN_RCL_CTL = 1).

Application Conditions

Initialization: RST
Recurrence: 10MS
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

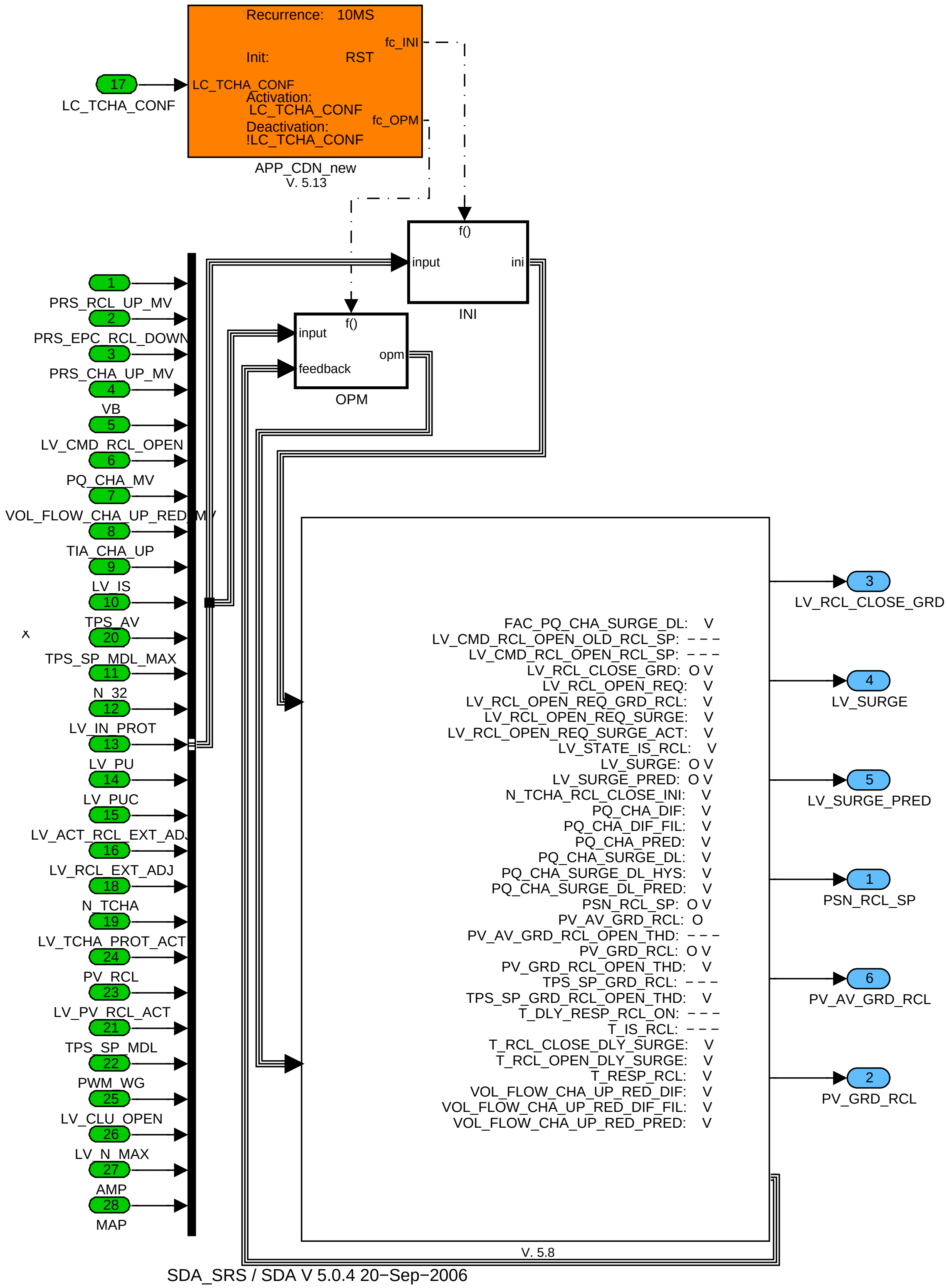


Figure 207:
Path: CHRG_ISPCLPSNRC0

9.80.1 Initialization

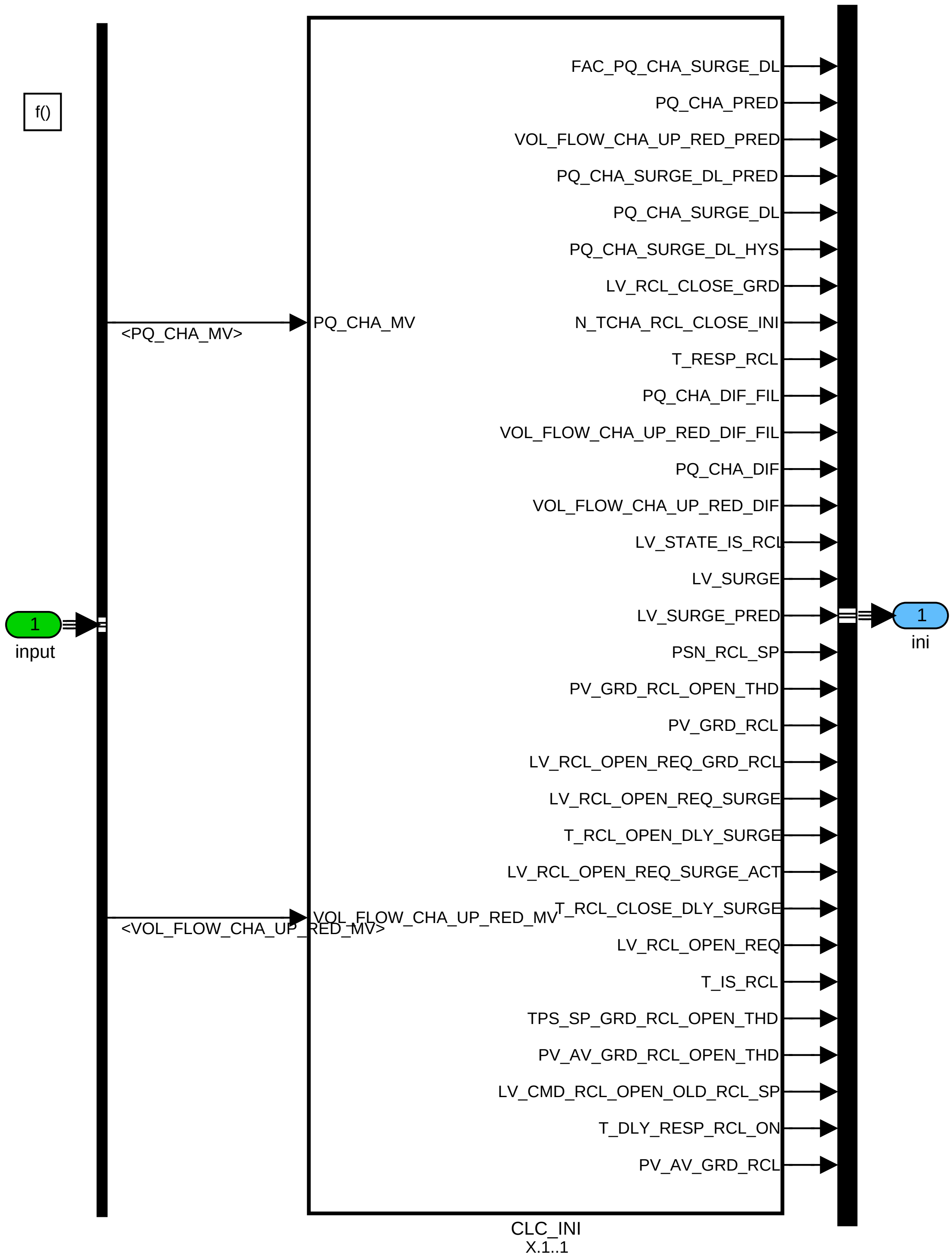


Figure 208:
Path: CHRG_ISPCLPSNRC0/INI

9.80.1.1 Initialization at reset:

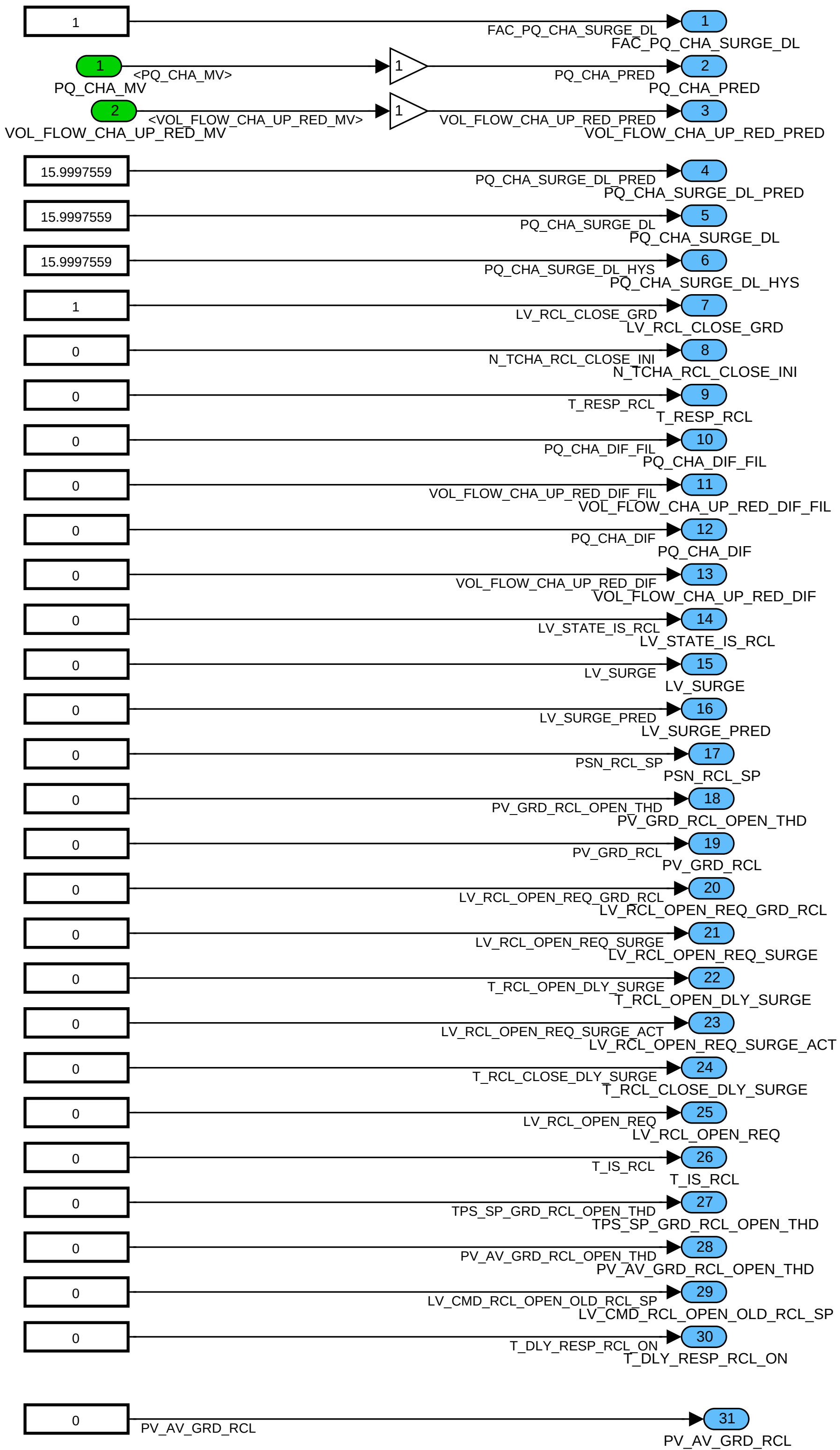


Figure 209:
Path: CHRG_ISPCLPSNRC0/INI/CLC_INI

9.80.2 SUBFUNCTION: operate_10ms

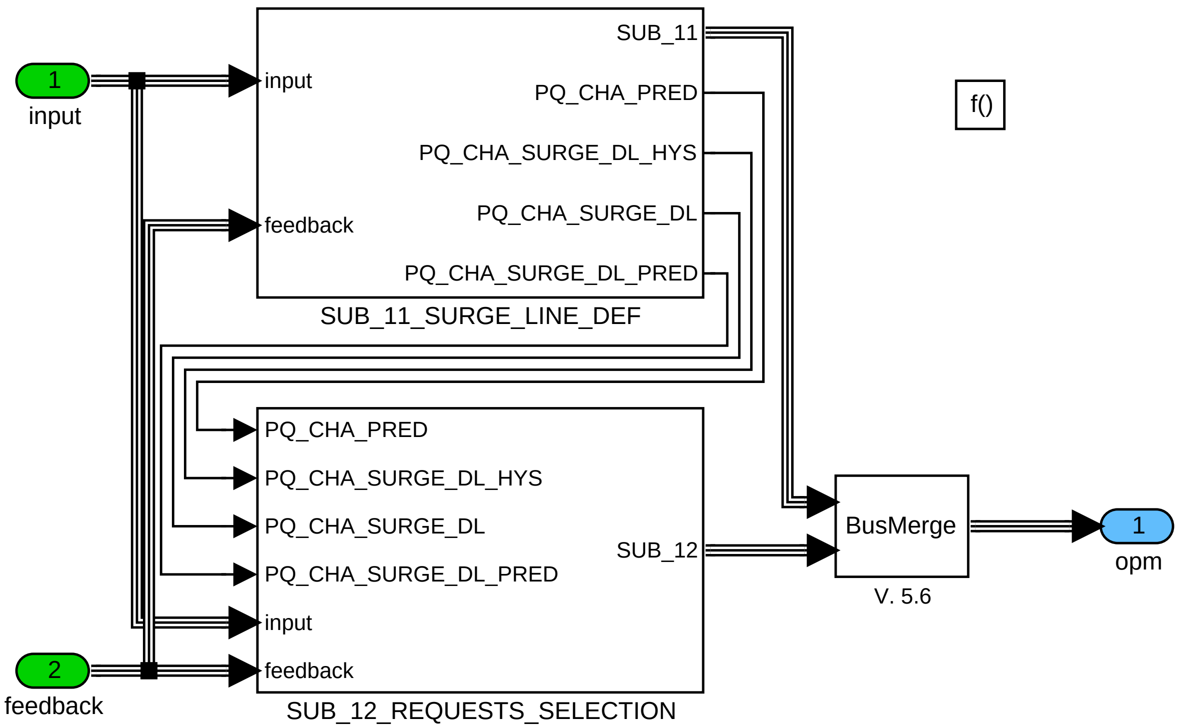


Figure 210:
Path: CHRG_ISPCLPSNRC0/OPM

9.80.2.1 Surge line definition with prediction based on RCL actuator response time:

The RCL activation response time is defined based on pressure conditions. Activation times are determined separately for activation and deactivation.
Surge is detected both based on current volume flow and pressure ratio PQ_CHA_MV and VOL_FLOW_CHA_UP_RED and on predicted pressure ratio PQ_CHA_PRED and VOL_FLOW_CHA_UP_RED_PRED. The prediction horizon is defined as equal to the response time of the RCL actuator. For a bi turbo system surge is determined on average values of the inlet branches.

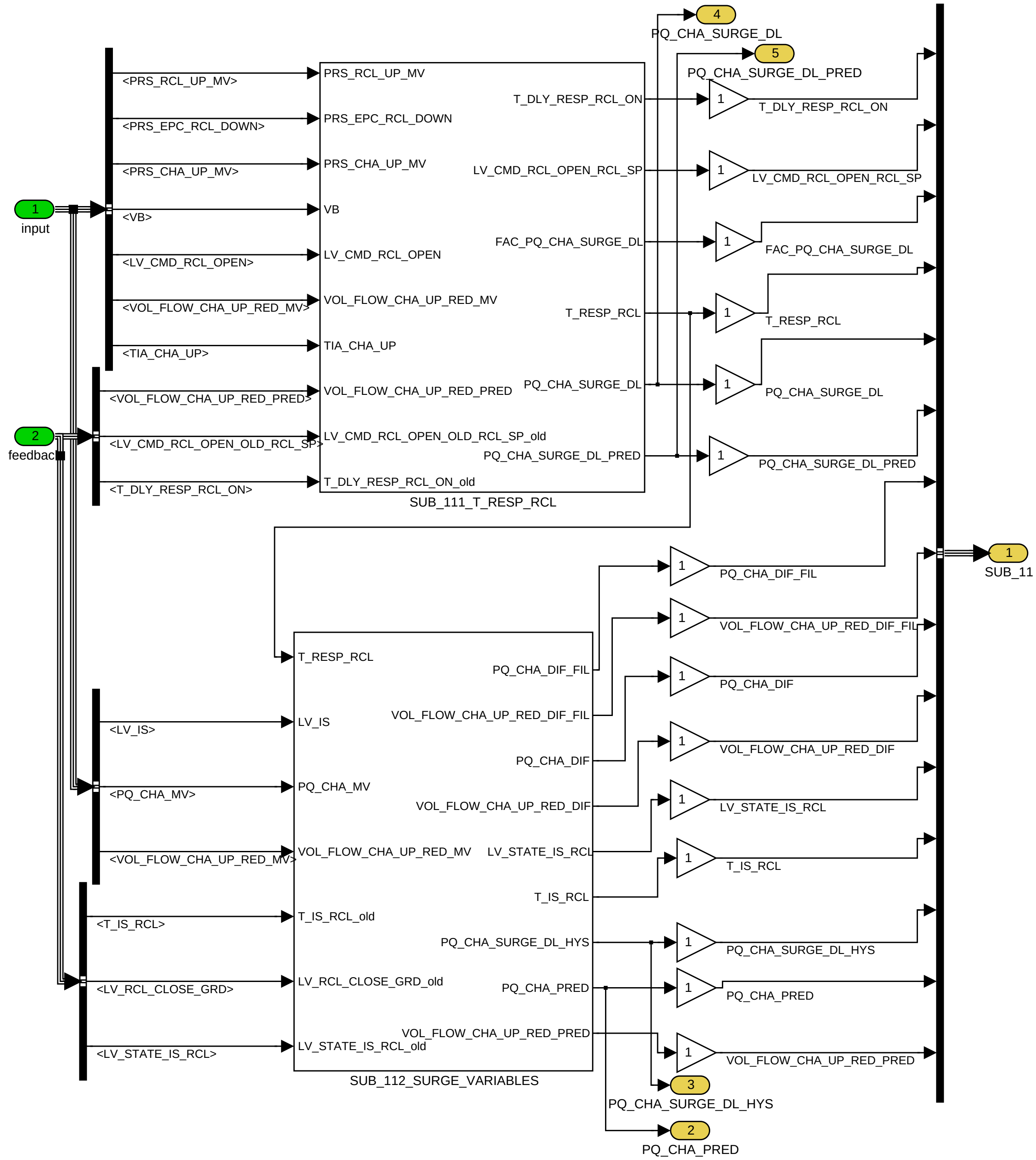


Figure 211:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF

9.80.2.1.1 RCL Response:

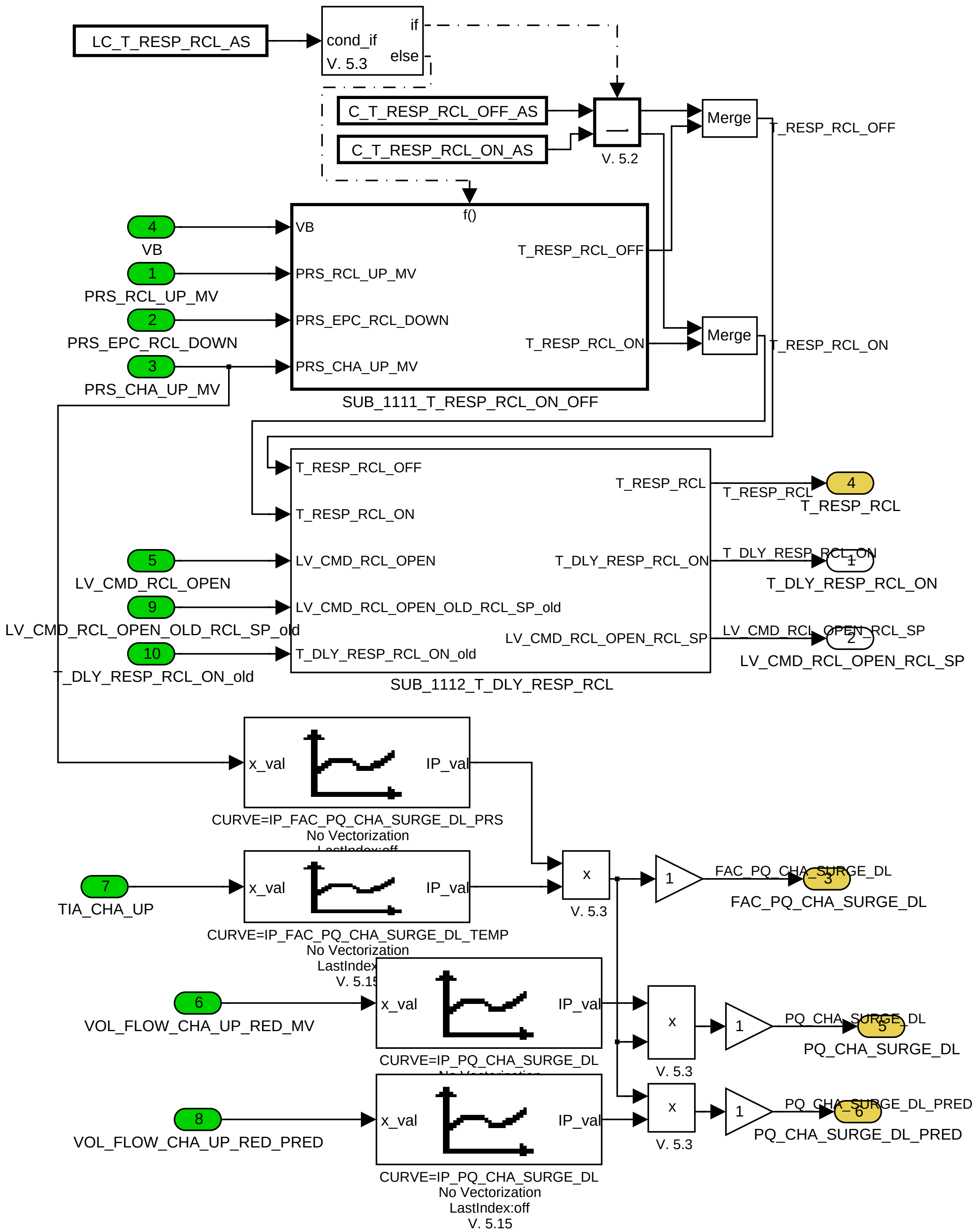


Figure 212:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF/SUB_111_T_RESP_RCL

9.80.2.1.1.1 RCL Actuator Response Time:

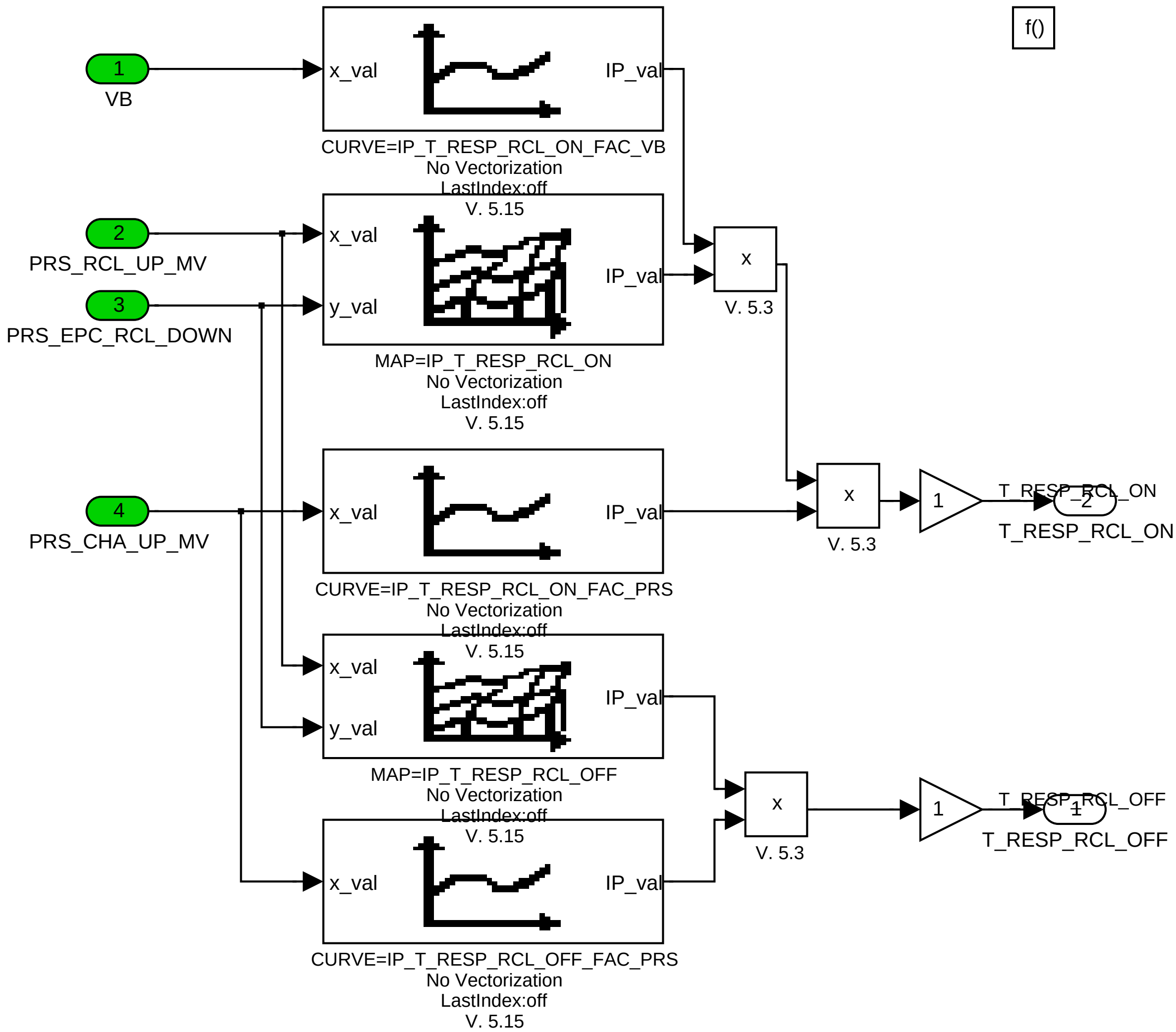


Figure 213:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF/SUB_111_T_RESP_RCL/SUB_1111_T_RESP_RCL_ON_OFF

9.80.2.1.1.2 RCL Actuator Response Delay Time:

When RCL opening command has just been reset (LV_CMD_RCL_OPEN = 0 and LV_CMD_RCL_OPEN_OLD_RCL_SP = 1) T_RESP_RCL_ON is copied to T_DLY_RESP_RCL_ON. Otherwise T_DLY_RESP_RCL_ON is decremented. After a reset of LV_CMD_RCL_OPEN (LV_CMD_RCL_OPEN = 0 and T_DLY_RESP_RCL_ON = 0) T_RESP_RCL_ON stays active.

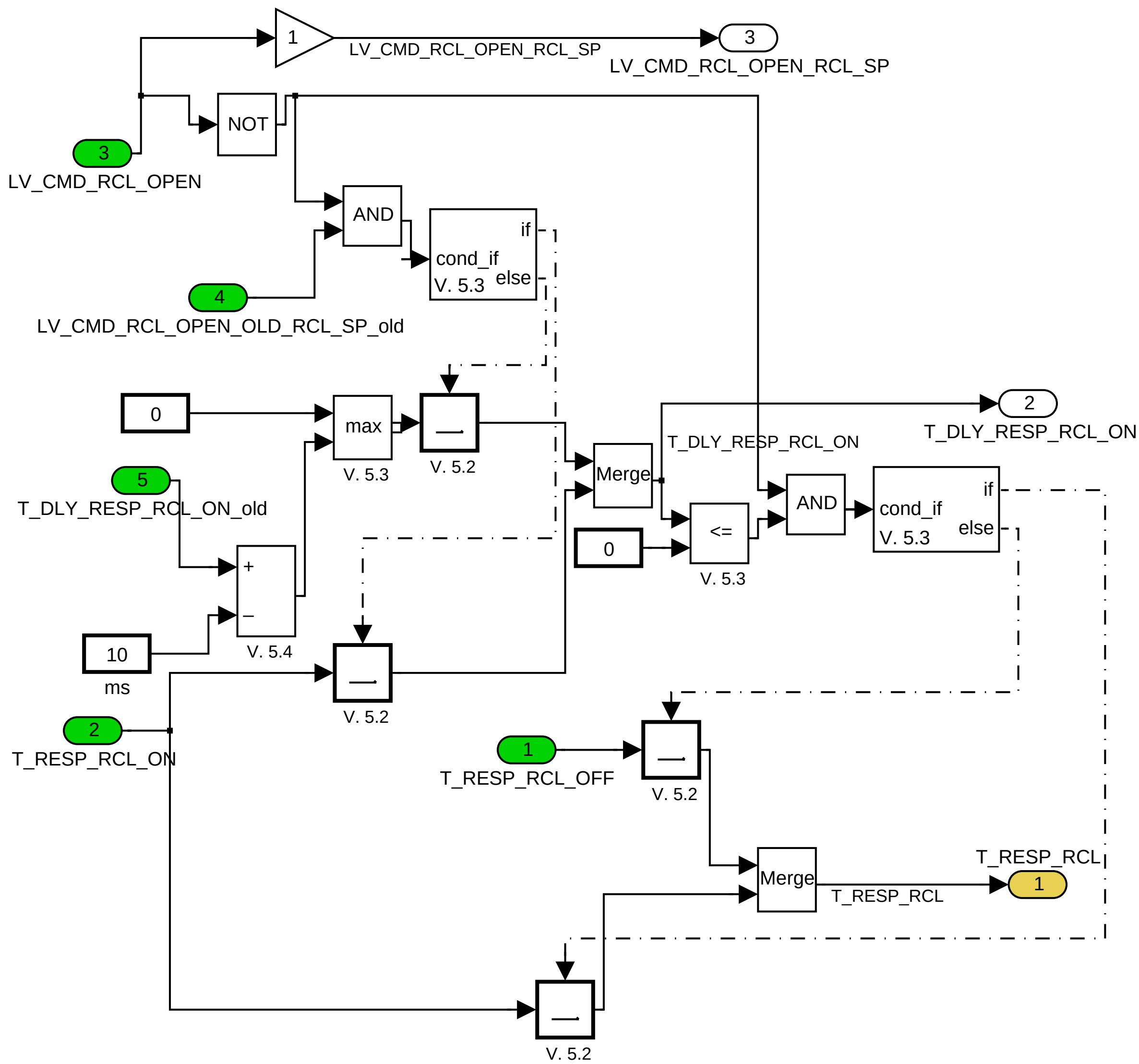


Figure 214:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF/SUB_111_T_RESP_RCL/SUB_1112_T_DLY_RESP_RCL

9.80.2.1.2 Serge Variables:

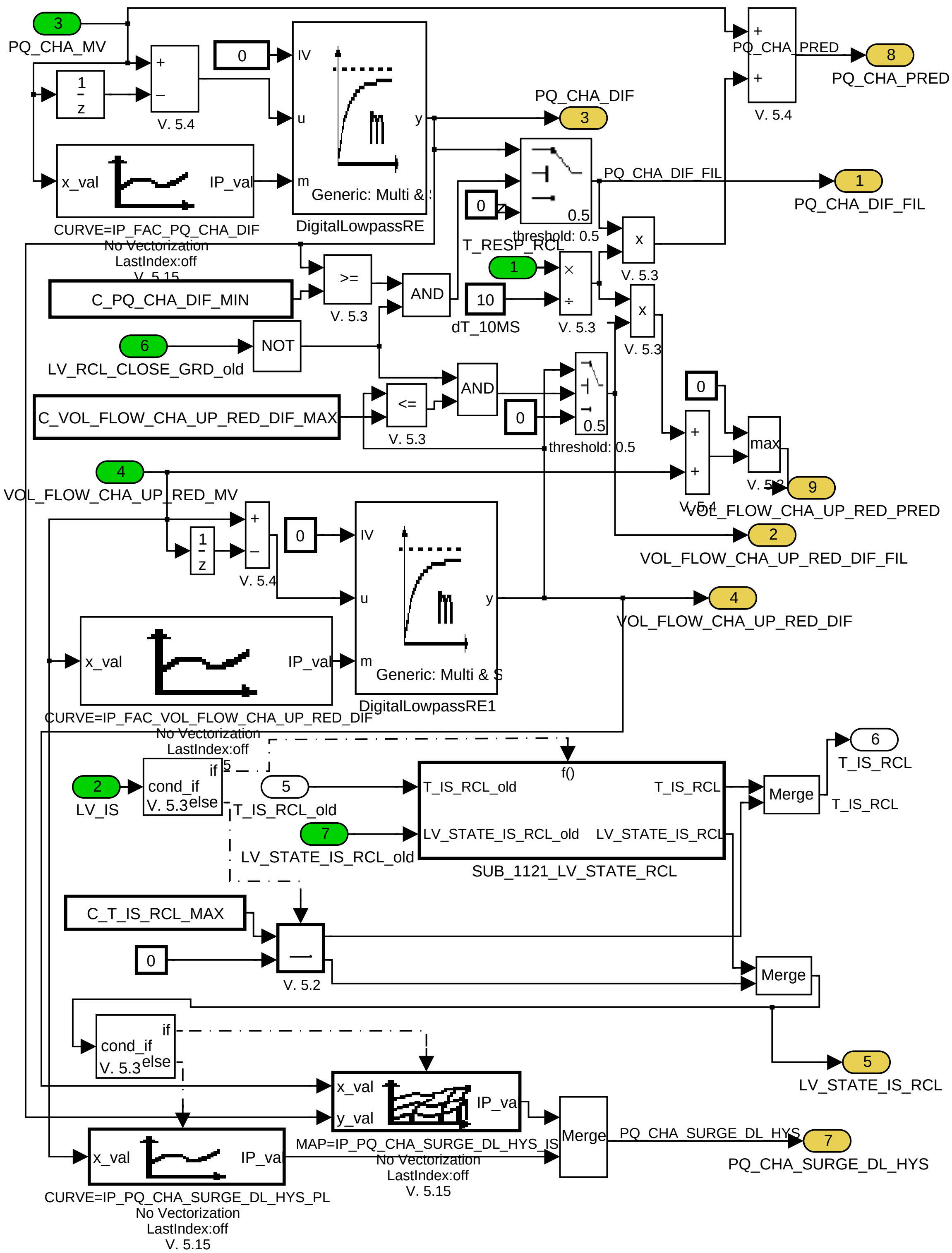


Figure 215:

Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF/SUB_112_SURGE_VARIABLES

9.80.2.1.2.1 Idle Information for RCL:

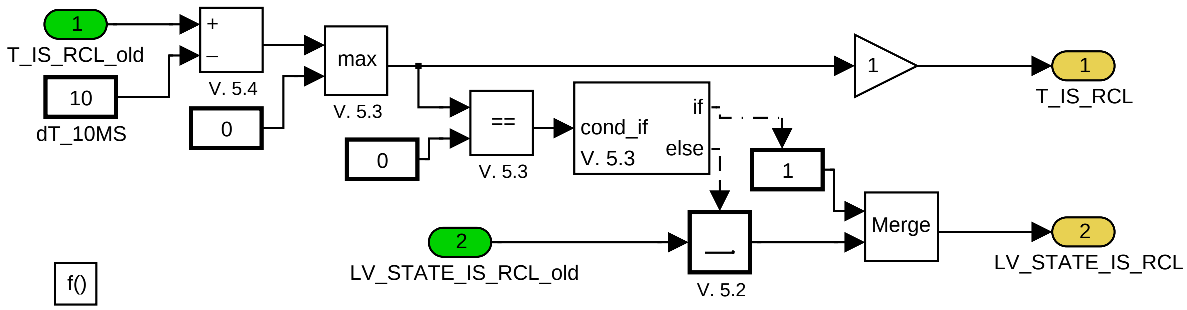
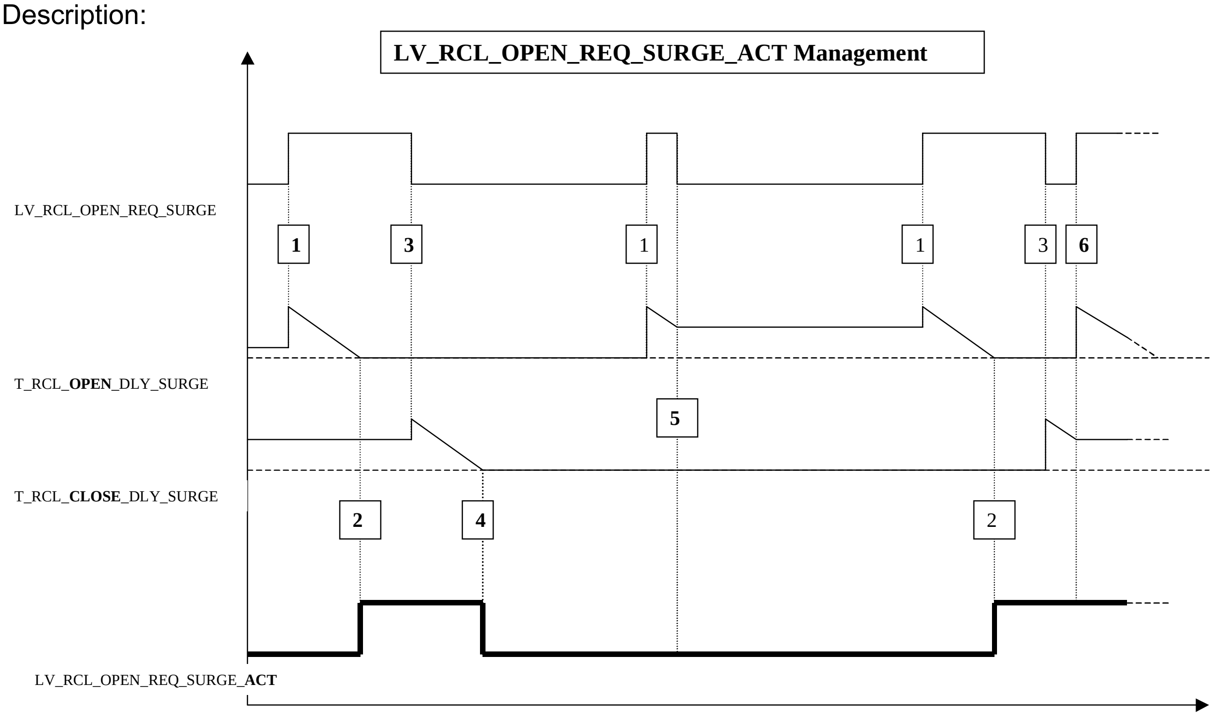


Figure 216:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_11_SURGE_LINE_DEF/SUB_112_SURGE_VARIABLES/SUB_1121_LV_STATE_RCL

9.80.2.2 Different Requests Selection:

Different conditions may occur which lead to the demand of an open recirculation actuator.
The recirculation actuator can be opened if:
. surge protection is active (refer to LV_SURGE, LV_SURGE_PRED definitions) or
. overpressure protection is active (refer to LV_IN_PROT in 30904301) or
. turbo charger protection functionality is active (refer to LV_TCHA_PROT_ACT) or
. the pedal [in case of no torque intervention] is quickly released or
. the throttle [in case of torque intervention] is quickly closed or
. the engine is either in trailing throttle or trailing throttle fuel cut off state (this can be deactivated by application) or
. a request is generated via service tool (end of line test) or application system.
The RCL is forced closed when cruise control is active.
Description:



1: Transition of LV_RCL_OPEN_REQ_SURGE = 0 to 1 (Timer T_RCL_OPEN_DLY_SURGE is initialized with C_T_RCL_OPEN_DLY_SURGE
From 1 to 2: T_RCL_OPEN_DLY_SURGE is decreased as long as LV_RCL_OPEN_REQ_SURGE = 1

2: Timer T_RCL_OPEN_DLY_SURGE has elapsed (LV_RCL_OPEN_REQ_SURGE_ACT is set to 1
3: Transition of LV_RCL_OPEN_REQ_SURGE = 1 to 0 And LV_RCL_OPEN_REQ_SURGE_ACT = 1 (Timer T_RCL_CLOSE_DLY_SURGE is initialized with IP_T_RCL_CLOSE_DLY_SURGE
From 3 to 4: T_RCL_CLOSE_DLY_SURGE is decreased as long as LV_RCL_OPEN_REQ_SURGE = 0
4: Timer T_RCL_CLOSE _DLY_SURGE has elapsed (LV_RCL_OPEN_REQ_SURGE_ACT is set to 0
5: Transition of LV_RCL_OPEN_REQ_SURGE = 1 to 0 (Timer T_RCL_OPEN_DLY_SURGE is frozen and LV_RCL_OPEN_REQ_SURGE_ACT remains unchanged
Timer T_RCL_CLOSE_DLY_SURGE is not initialized as LV_RCL_OPEN_REQ_SURGE_ACT = 0 (ie: no need to plan to close the actuator which is already closed !)
6: Transition of LV_RCL_OPEN_REQ_SURGE = 0 to 1 (Timer T_RCL_CLOSE_DLY_SURGE is frozen and LV_RCL_OPEN_REQ_SURGE_ACT remains unchanged

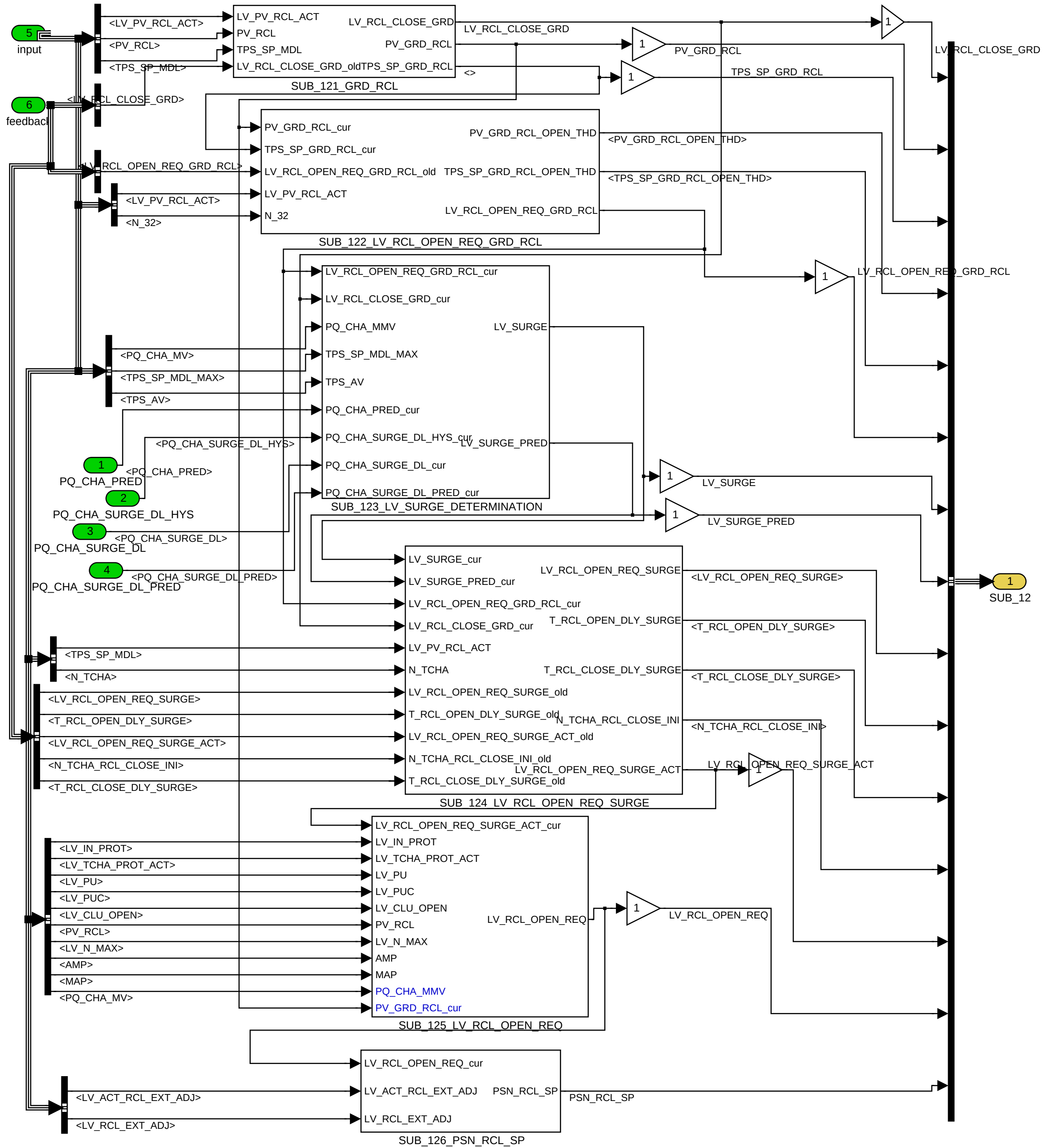


Figure 217:

Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION

9.80.2.2.1 RCL Actuator Closing Request:

LV_PV_RCL_ACT determines whether pedal value PV_RCL (= 1) or throttle opening setpoint TPS_SP_MDL (= 0) controls the recirculation actuator (refer to module Recirculationactuator set point determination Appl. Inc.).

If the torque request by the driver increases pedal value PV_RCL and/or throttle opening setpoint TPS_SP_MDL increase. If both pedal value PV_RCL and/or throttle opening setpoint TPS_SP_MDL and their gradients PV_GRD_RCL and TPS_SP_GRD_RCL exceed activation thresholds then the recirculation actuator is requested to closed (LV_RCL_CLOSE_GRD = 1). If the gradients fall below deactivation thresholds this request is reset (LV_RCL_CLOSE_GRD = 0).

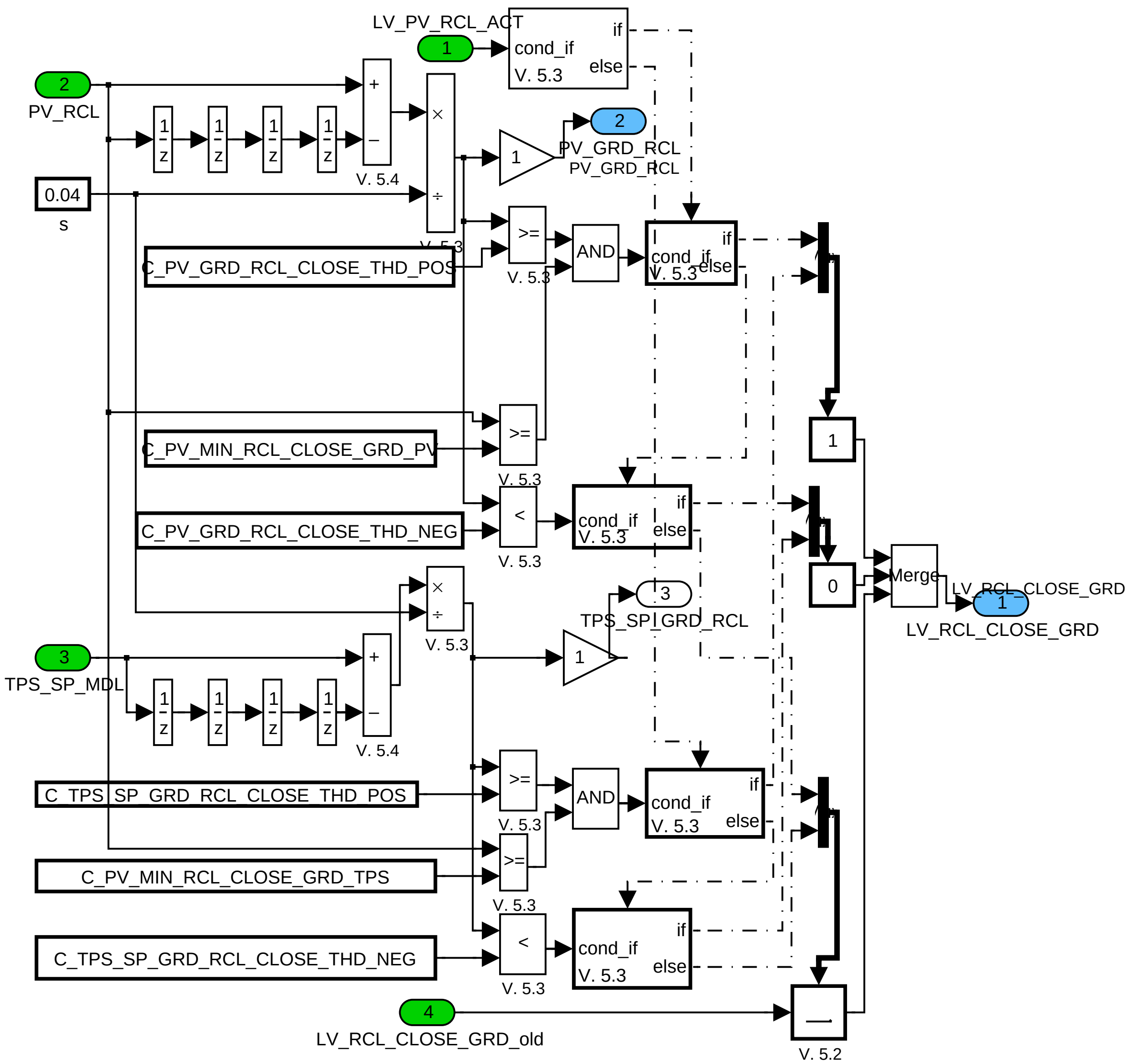


Figure 218:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_121_GRD_RCL

9.80.2.2.2 RCL Actuator Opening Request:

LV_PV_RCL_ACT determines whether pedal value PV_RCL (= 1) or throttle opening setpoint TPS_SP_MDL (= 0) controls the recirculation actuator.
If the torque request by the driver decreases pedal value PV_RCL and/or throttle opening setpoint TPS_SP_MDL decrease. If pedal value gradient PV_GRD_RCL and/or throttle opening setpoint gradient TPS_SP_GRD_RCL fall below activation thresholds then the recirculation actuator is requested to open (LV_RCL_OPEN_REQ_GRD_RCL = 1). If the gradients increase again by a hysteresis C_PV_GRD_RCL_OPEN_HYS then this request is reset (LV_RCL_OPEN_REQ_GRD_RCL = 0).

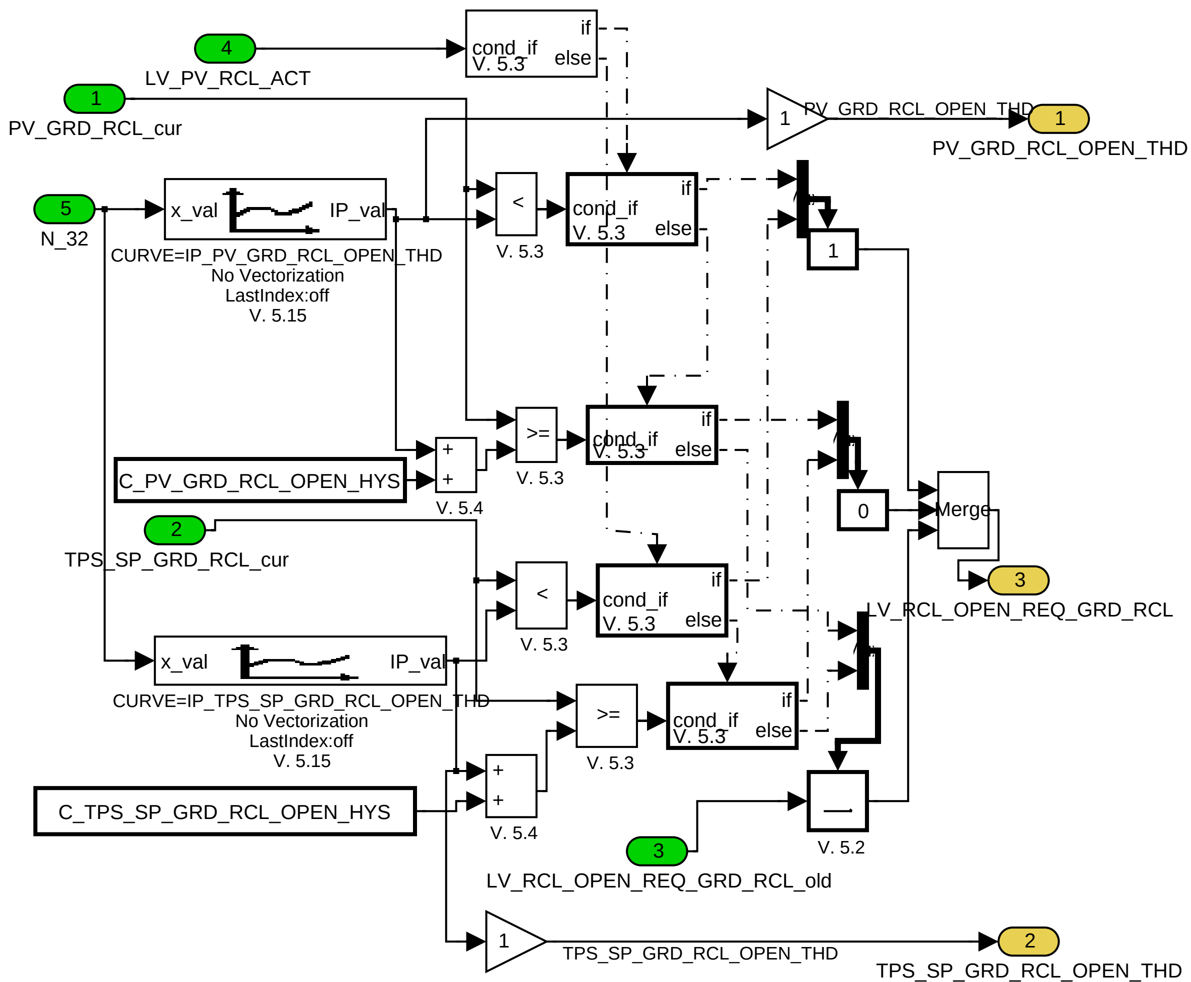


Figure 219:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_122_LV_RCL_OPEN_REQ_GRD_RCL

9.80.2.2.3 Surge Line and Surge line predicted definition:

Surge is detected if the actual throttle opening TPS_AV is close to its maximum possible value TPS_SP_MDL and either the current compressor pressure ratio PQ_CHA_MV is greater than the current surge limit PQ_CHA_SURGE_DL or the predicted compressor pressure ratio PQ_CHA_PRED is greater than the predicted surge limit PQ_CHA_SURGE_DL_PRED or (switchable) recirculation actuator opening is requested via PV/TPS gradient condition.

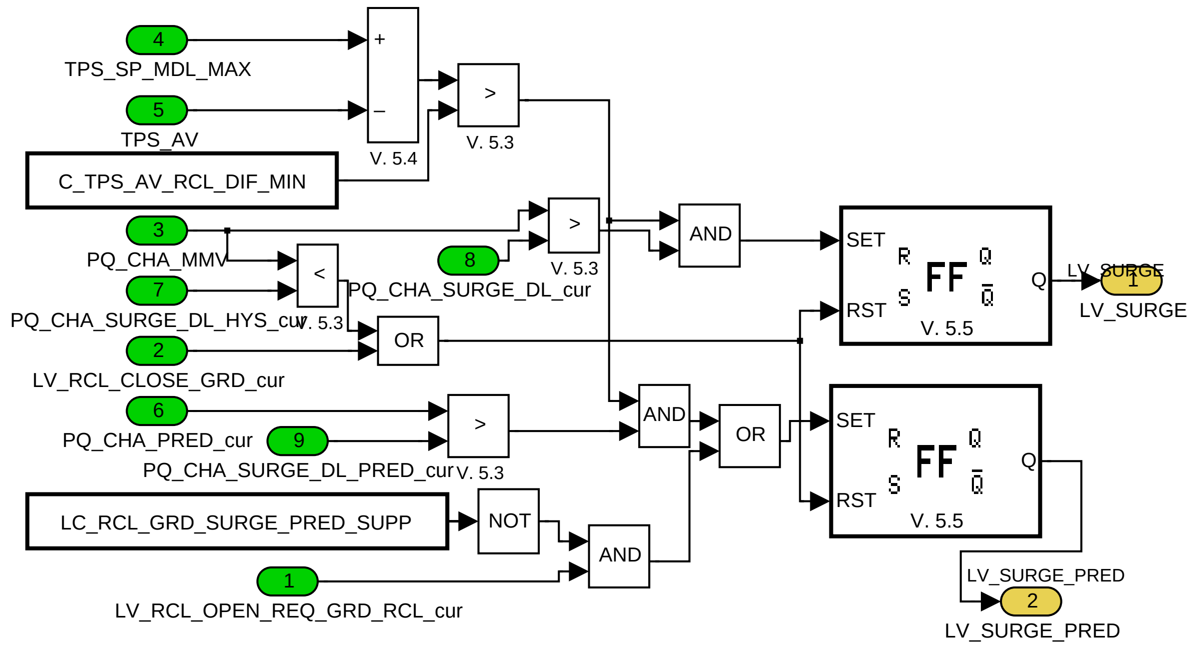


Figure 220:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_123_LV_SURGE_DETERMINATION

9.80.2.2.4 RCL Open request surge:

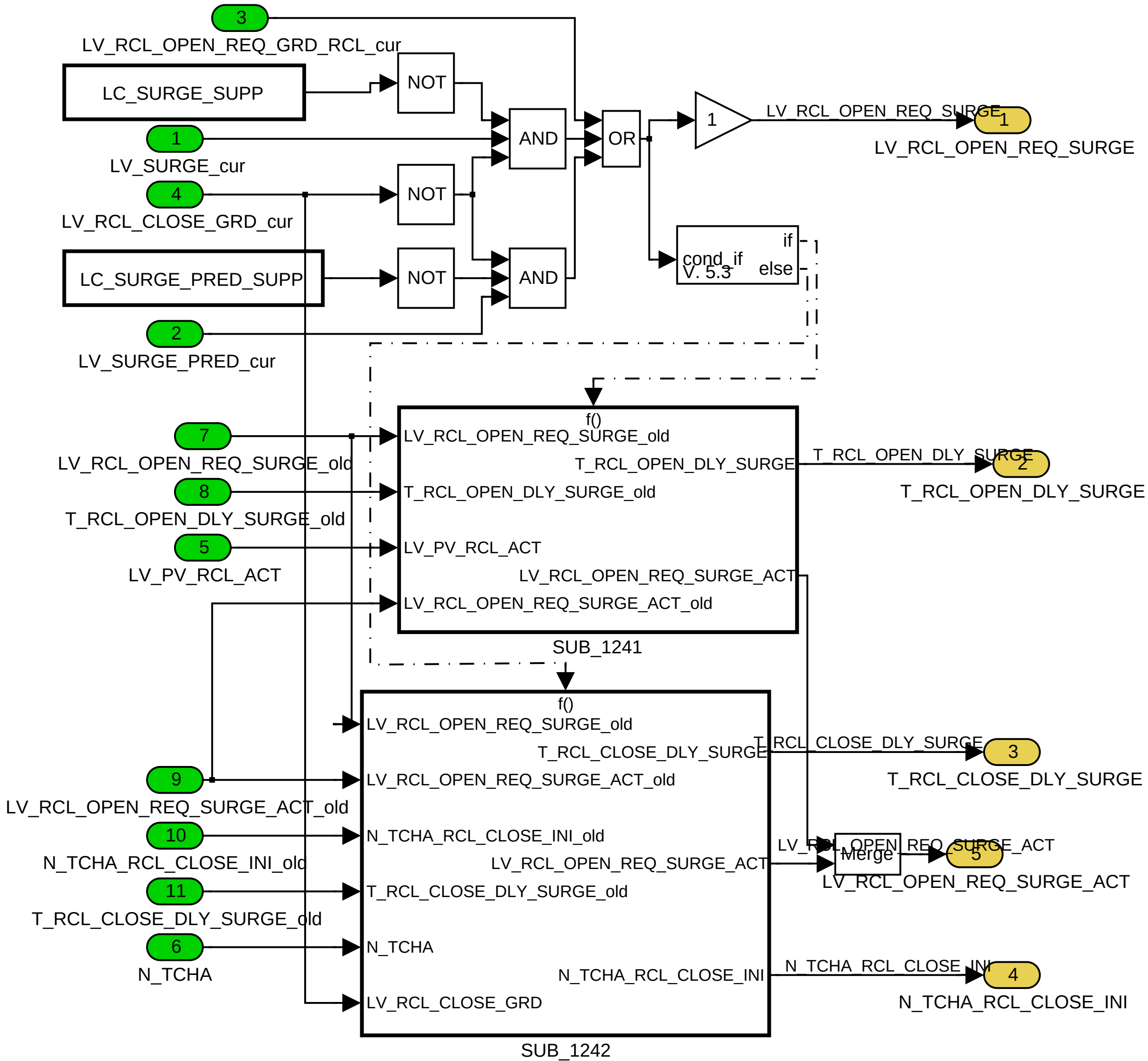


Figure 221:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_124_LV_RCL_OPEN_REQ_SURGE

9.80.2.2.4.1 RCL Opening delay time for activation

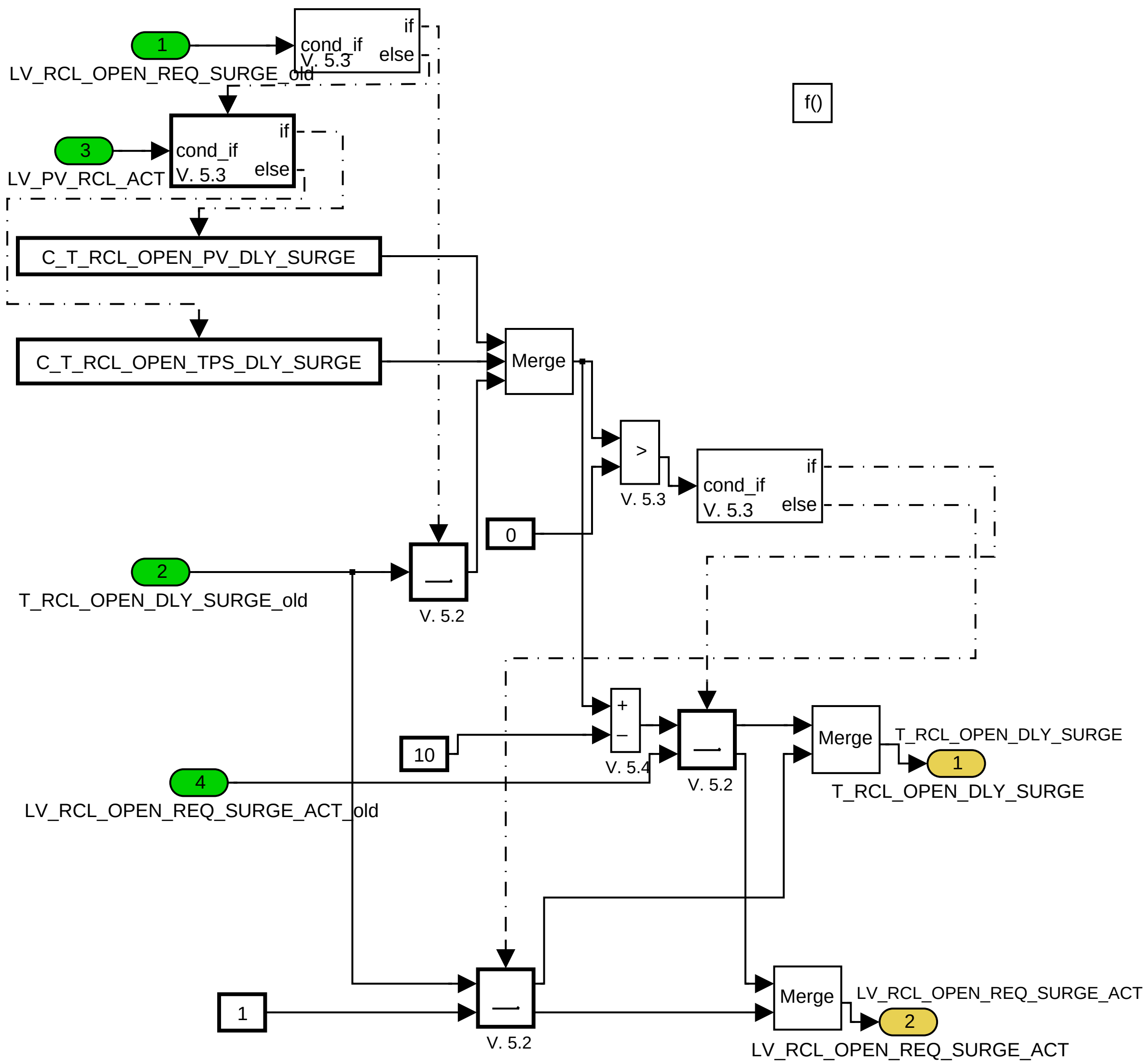


Figure 222:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_124_LV_RCL_OPEN_REQ_SURGE/SUB_1241

9.80.2.2.4.2 RCL closing delay time for activation

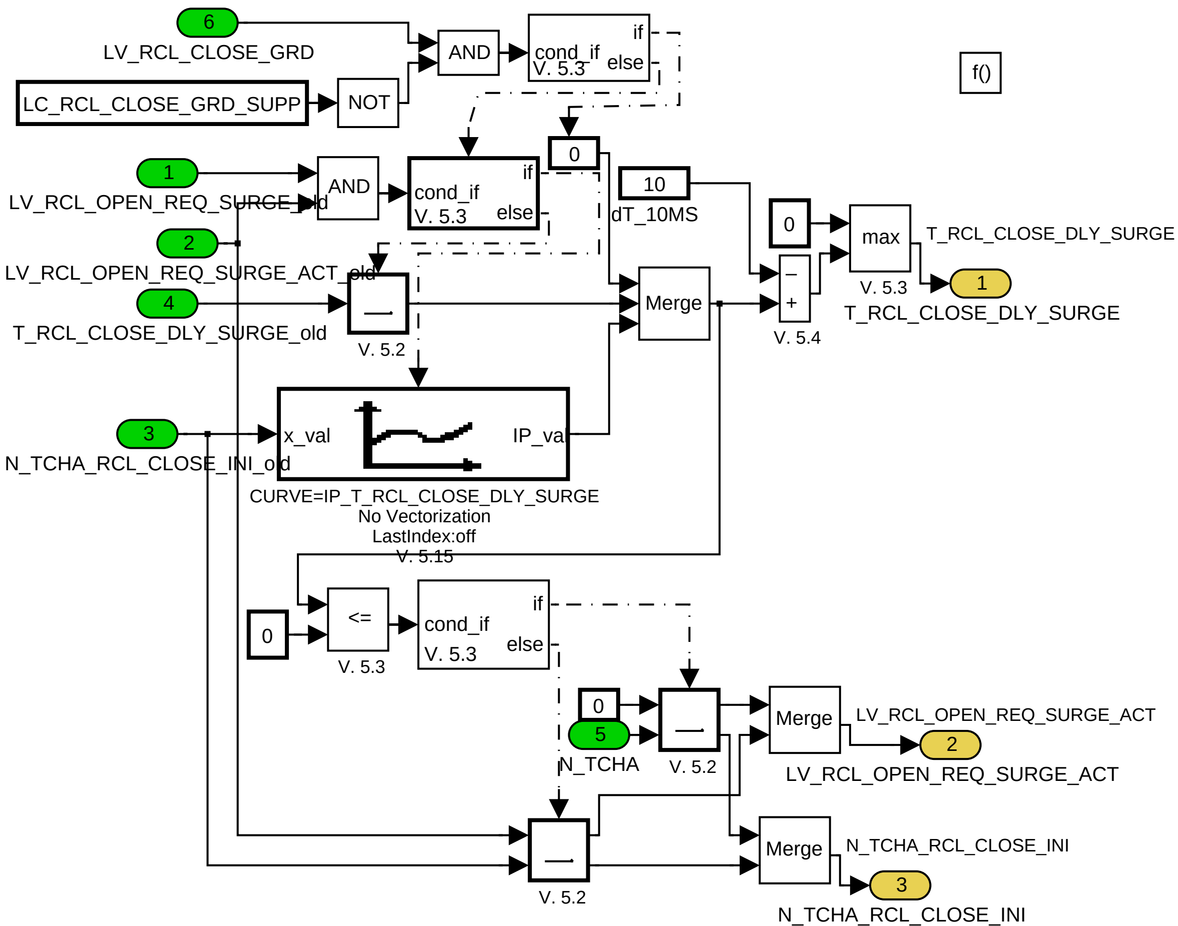


Figure 223:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_124_LV_RCL_OPEN_REQ_SURGE/SUB_1242

9.80.2.2.5 RCL Actuator opening request:

Opening and closing requests for the recirculation actuator are coordinated here. Highest priority for have the opening request because of turbo charger protection LV_TCHA_PROT_ACT, opening request because of intake manifold protection LV_IN_PROT and (switchable) opening request because of non charged engine operation. Opening requests because of PU/PUC (switchable), because of engine speed limitation (switchable), because of surge detection (with pedal value minimum threshold) and because of opened clutch (switchable) have second priority. Second priority requests are blocked if the pedal value is greater than a threshold C_PV_RCL_CLOSE_BOL and the clutch is open (gear shift for fast acceleration). Additionally, to reduce the turbo lag out of low load, it is also possible to open the recirculation valve for a certain time until the turbocharger starts to compress the intake air. The goal of this is to reduce the pressure losses during this time by bypassing the charger.
To activate the RCL opening the gradient for the pedal value has to pass the threshold C_PV_GRD_RCL_OPEN_SET. Moreover the pressure quotient of the charger has to be under the calibrateable value C_PQ_CHA_RCL_OPEN_SET. If the pressure quotient exceeds this threshold the recirculation valve closes again, not to limit the boost pressure in a negative way.

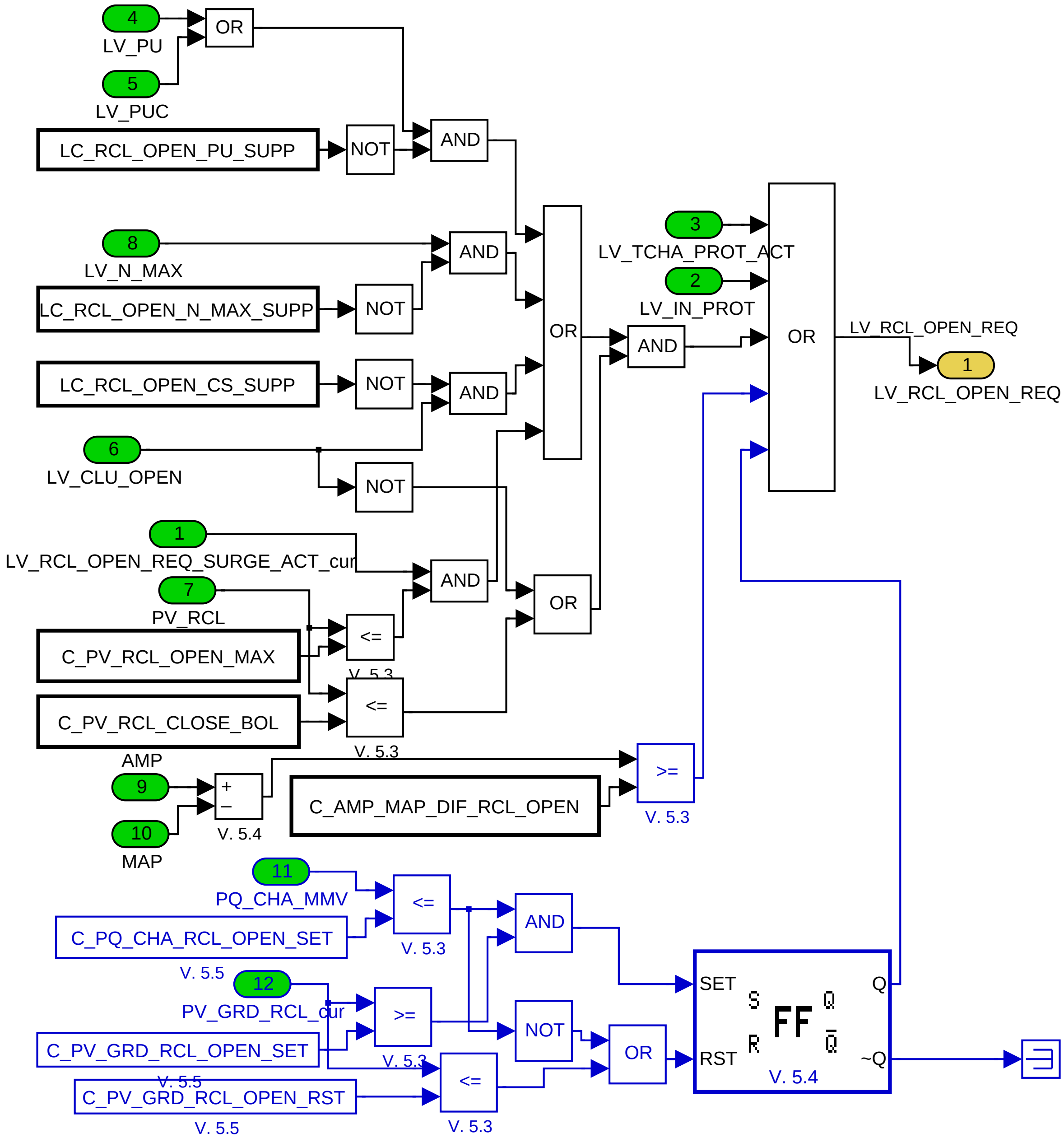


Figure 224:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_125_LV_RCL_OPEN_REQ

9.80.2.2.6 RCL Actuator position set point:

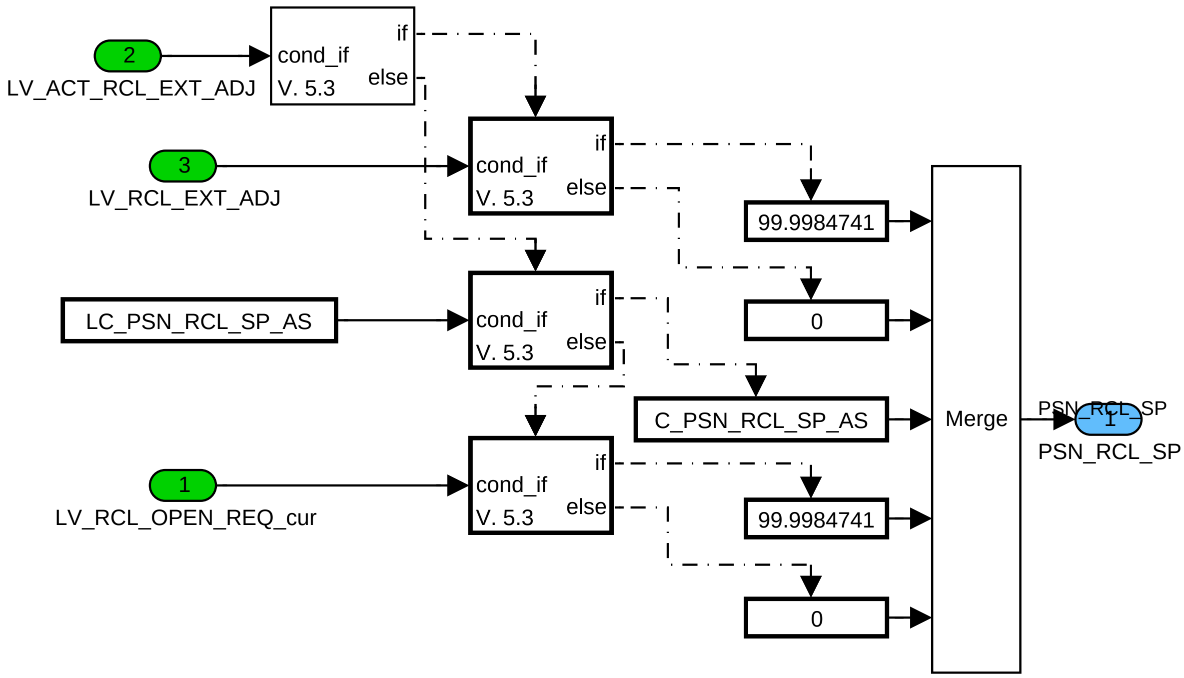


Figure 225:
Path: CHRG_ISPCLPSNRC0/OPM/SUB_12_REQUESTS_SELECTION/SUB_126_PSN_RCL_SP

9.81 Recirculation actuator setpoint determination (Application incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PV_RCL_ACT	O/V	0...1H	0...1	1	-
Selection flag for reference value for RCL setpoint det.: 1 = "PV_AV or PV_AV_VS" value / 0 = "TPS_SP_MDL"					
PV_RCL	O/V	0...3FFH	0...99.9023438	0.09765625	%
Pedal value used for RCL setpoint determination					
LV_TQ_RCL_INTV	O/V	0...1H	0...1	1	-
Torque intervention flag for RCL					

Input data:

LC_TCHA_CONF	LV_RCL_CLOSE_GRD	LV_CS	LV_INTV_CRU
LV_ASR_ACT	LV_POW_AC_TRV	LV_TQ_LIM_INTV	LV_INTV_VSL
PV			

Calibration data:

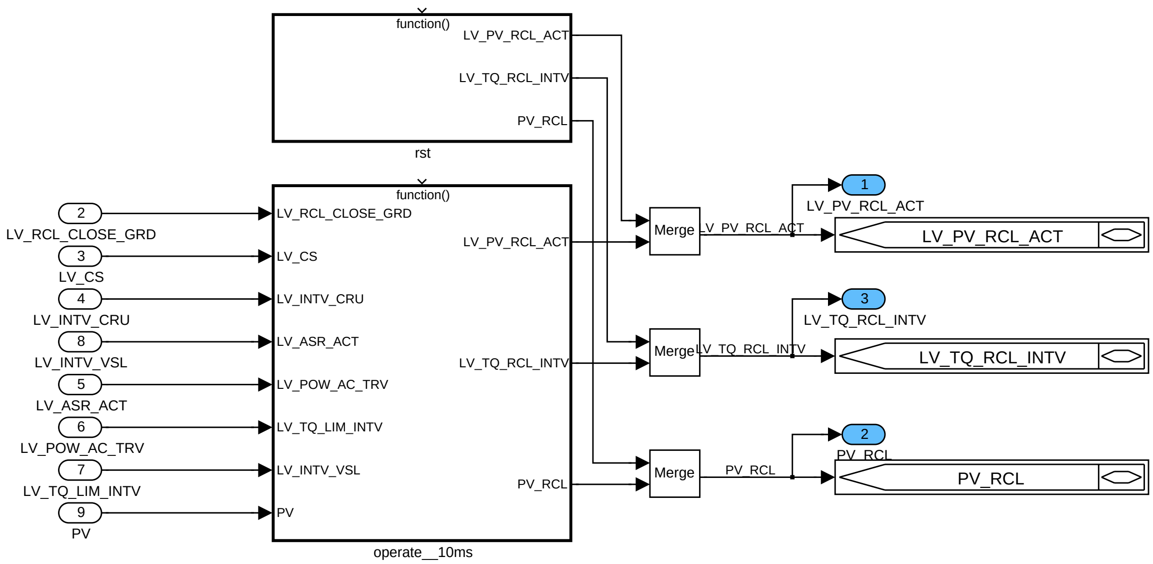
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_PV_AV_RCL	1	0...1H	0...1	1	-
Choice of PV_AV_RCL definition					
LC_RCL_CLOSE_CS_SUPP	1	0...1H	0...1	1	-
Suppression of test on LV_RCL_CLOSE_GRD in case of LV_CS detection					
LC_TQ_RCL_INTV_SUPP	1	0...1H	0...1	1	-
Suppression of LV_TQ_RCL_INTV effect for RCL setpoint determination					

9.81.1 CHRG_ISPCLPRCAI0

The pedal value used for recirculatioin actuator position control and the condition when recirculation actuator control is based on pedal value, when on throttle position can vary for different engines. These calculations are done here project specifically.

The pedal value used for recirculation actuator position control can be the measured pedal value or a reconstructed pedal value including torque several interventions.

Function Description



SDA_SRS / SDA 4.0 01−Apr−2005

Figure 226 CHRG_ISPCLPRCAI0

9.81.1.1 SUBFUNCTION: operate__10ms

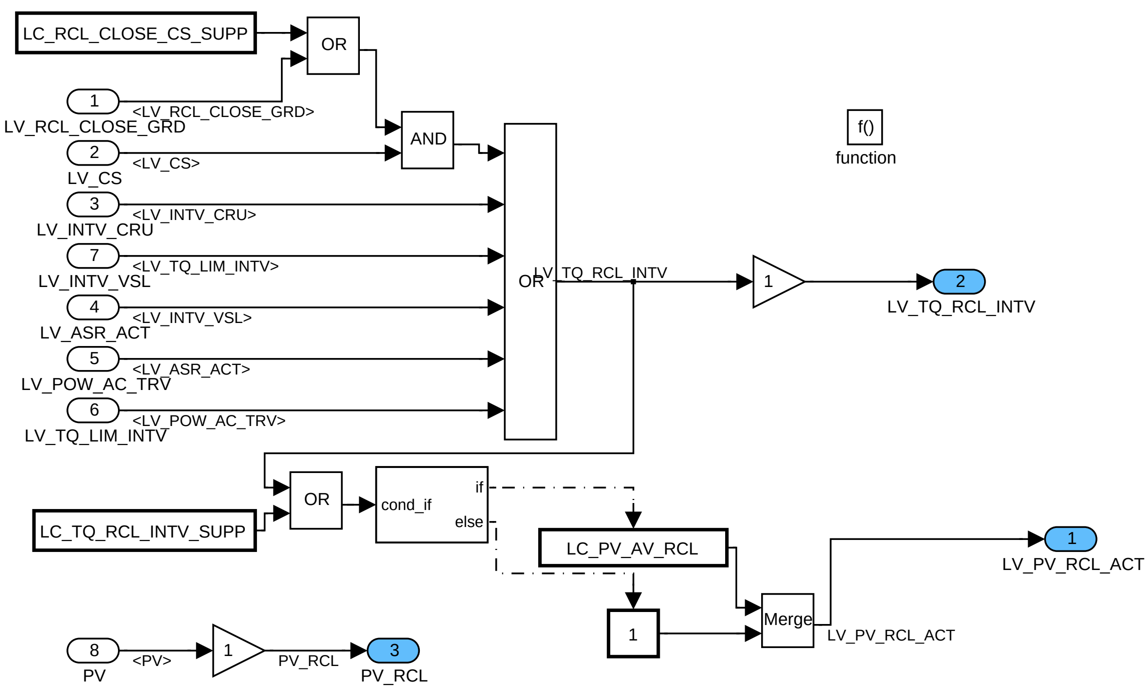


Figure 227 CHRG_ISPCLPRCAI0/ operate__10ms

9.81.1.2 SUBFUNCTION: rst

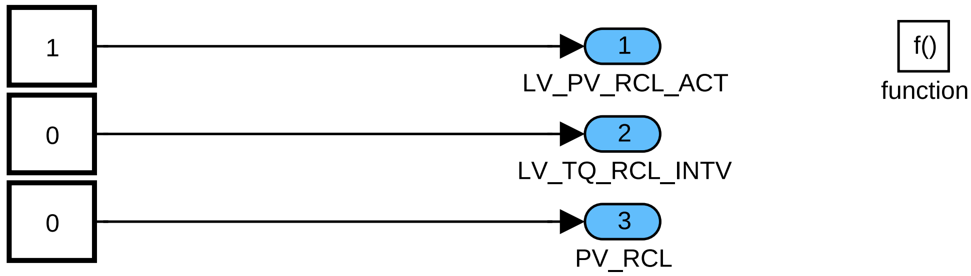


Figure 228 CHRG_ISPCLPRCAI0/ rst

9.82 Charge air pressure control

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
CRLC_PUT_DIF	V	0... FFFFH	0... 0.9999847	15.259e-6	[-]
filter constant, representing time constant for DT1 share					
FAC_POW_PUT_CTL	V	0... FFFFH	0... 7.999878	122.07e-6	[-]
Turbine power factor of PID Controller					
FAC_POW_PUT_CTL_D	V	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Turbine power factor of PID Controller D-PART					
FAC_POW_PUT_CTL_I	O/V	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Turbine power factor of PID Controller I-PART					
FAC_POW_PUT_CTL_P	V	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Turbine power factor of PID Controller P-PART					
FAC_PUT_CTL_BAS	V	0... FFFFH	0... 7.999878	122.07e-6	[-]
Precontrol for FAC_POW_PUT_CTL					
FAC_PUT_CTL_D	V	0... FFFFH	0... 0.376881	5.7508e-6	[1/hPa]
Final D gain of charged air pressure controller					
FAC_PUT_CTL_I	V	0... FFFFH	0... 0.036805	561.61e-9	[1/(hPa*s)]
Final I gain of charged air pressure controller					
FAC_PUT_CTL_P	V	0... FFFFH	0... 0.18844	2.8754e-6	[1/hPa]
Final P gain of charged air pressure controller					
LV_PUT_CTL_ACT	O/V	0... 1H	0... 1	1	[-]
logical value for activating the PUT-controlling					
LV_PUT_CTL_I_ACT	V	0... 1H	0... 1	1	[-]
masked logical value for activating the PUT-controller					
LV_PUT_CTL_I_LIM	V	0... 1H	0... 1	1	[-]
Flag for frozen I-part of charge air pressure control					
LV_PUT_CTL_STAT	O/V	0... 1H	0... 1	1	[-]
PUT controller works near setpoint					
PUT_DIF	O/V	8000... 7FFFH	-5434... 5433.834	0.1658325	[hPa]
PUT deviation for charge air pressure controller					
PUT_DIF_MMV	V	8000... 7FFFH	-5434... 5433.834	0.1658325	[hPa]
filtered control deviation of the prs_up_thr feedback control system					
PUT_DIF_MMV_GRD	V	8000... 7FFFH	-5434... 5433.834	0.1658325	[hPa]
gradient of filtered control deviation of pressure setpoint and actual value					
PUT_DIF_MMV_STAT	V	8000... 7FFFH	-5434... 5433.834	0.1658325	[hPa]
filtered PUT controller deviation for stationary operation recognition					
PUT_MMV_GRD	V	8000... 7FFFH	-5434... 5433.834	0.1658325	[hPa]
Gradient of filtered pressure upstream throttle					
PUT_MMV_STAT	V	0... FFFFH	0... 5434	0.0829175	[hPa]
Filtered PUT for stationary operation recognition					
PWM_WG_CLL	O/V	0... FFFFH	-100... 99.99695	3.0518e-3	[%]
Closed looop control part of PWM for wastegate EPC					

Input Data:

AMP	LC_TCHA_CONF	LV_AT	LV_ENA_PUT
LV_IN_PROT	LV_MTC_CUR_OFF	LV_PWM_WG_EXT_ADJ	LV_ST_END
LV_TCHA_PROT_ACT	LV_TQI_OHP	MAF	MAF_SP
N	N_CHA_SP	N_TCHA	PUT_MMV
PUT_SP	PUT_WG_OPEN	PWM_WG	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_PUT_DIF	1	0... FFFFH	0... 0.9999847	15.259e-6	[-]
Filter constant for charge air pressure control deviation					
C_CRLC_PUT_DIF_MMV_STAT	1	0... FFFFH	0... 0.9999847	15.259e-6	[-]
filter constant for PUT controller deviation for stationary operation recognition					
C_CRLC_PUT_MMV_STAT	1	0... FFFFH	0... 0.9999847	15.259e-6	[-]
Filter constant for PUT for stationary charged air pressure controller operation recognition					
C_FAC_POW_PUT_CTL_BOL	1	0... FFFFH	0... 7.999878	122.07e-6	[-]
Bottom limit of PUT controller output					
C_FAC_POW_PUT_CTL_D_MAX	1	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Upper limit of PUT controller D share					
C_FAC_POW_PUT_CTL_I_BOL	1	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Bottom limit of PUT controller I g					
C_FAC_POW_PUT_CTL_I_INC	1	0... FFFFH	0... 7.999878	122.07e-6	[-]
PUT controller I part increment for anti windup					
C_FAC_POW_PUT_CTL_I_TOL	1	8000... 7FFFH	-4... 3.999878	122.07e-6	[-]
Top limit of PUT controller I gain					
C_FAC_POW_PUT_CTL_RAMP_DOWN	1	0... FFFFH	0... 7.999878	122.07e-6	[-]
PUT controller I part increment for controller deactivation					
C_FAC_POW_PUT_CTL_TOL	1	0... FFFFH	0... 7.999878	122.07e-6	[-]
Top limit of PUT controller output					
C_FAC_PUT_CTL_D_MAN	1	0... FFFFH	0... 0.376881	5.7508e-6	[1/hPa]
Manual PUT controller D gain					
C_FAC_PUT_CTL_I_COR_POS	1	0... FFFFH	0... 7.999878	122.07e-6	[-]
Correction factor of charge air pressure controller I gain for positive PUT_DIF					
C_FAC_PUT_CTL_I_MAN	1	0... FFFFH	0... 0.036805	561.61e-9	[1/(hPa*s)]
Manual PUT controller I gain					
C_FAC_PUT_CTL_P_MAN	1	0... FFFFH	0... 0.18844	2.8754e-6	[1/hPa]
Manual PUT controller P gain					
C_N_PUT_CTL_SRC	1	0... 2H	0... 2	1	[-]
Source of auxiliary rotational speed for charged air pressure control					
C_PUT_DIF_MAX_STAT_ACT	1	8000... 7FFFH	-2717.041... 2716.959	0.0829175	[hPa]
Maximum charge air pressure control deviation for activation of stationary charge air pressure control					
C_PUT_DIF_MAX_STAT_DEAC	1	8000... 7FFFH	-2717.041... 2716.959	0.0829175	[hPa]
Maximum charge air pressure control deviation for deactivation of stationary charge air pressure control					
C_PUT_DIF_MIN_STAT_ACT	1	8000... 7FFFH	-2717.041... 2716.959	0.0829175	[hPa]
Minimum charge air pressure control deviation for activation of stationary charge air pressure control					
C_PUT_DIF_MIN_STAT_DEAC	1	8000... 7FFFH	-2717.041... 2716.959	0.0829175	[hPa]
Minimum charge air pressure control deviation for deactivation of stationary charge air pressure control					
C_PUT_MMV_GRD_MAX_PUT_CTL_I	1	8000... 7FFFH	-2717.041... 2716.959	0.0829175	[hPa]
Maximum pressure upstream throttle gradient to activate I share of charge air pressure controller					
C_PWM_WG_BOL_PUT_CTL_LIM	1	0... FFFFH	0... 99.99847	1.5259e-3	[%]
PWM_WG bottom limit for PUT controller limitation					
C_PWM_WG_TOL_PUT_CTL_LIM	1	0... FFFFH	0... 99.99847	1.5259e-3	[%]
PWM_WG top limit for PUT controller limitation					
C_SWI_PUT_CTL_STAT_PUT_DIF_GRD	1	0... 2H	0... 2	1	[-]
Switch whether PUT_DIF_GRD has influence on recognition of stationary PUT controller operaton					
ID_FAC_PUT_DIF_MMV	4	0... FFFFH	-1... 0.9999695	30.518e-6	[-]
LDP_IDX_ID_FAC_PUT_DIF_MMV [4]	4	0... FFH	0... 255	1	[-]
Weighting factor of delayed PUT_DIF_MMV for gradient calculation					

IP_FAC_PUT_CTL_BAS	6x10	0... FFFFH	0... 7.999878	122.07e-6	[-]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
Precontrol for FAC_POW_PUT_CTL					
IP_FAC_PUT_CTL_D	10x8	0... FFFFH	0... 0.376881	5.7508e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_PUT_DIF_1_CHRG [8]	8	0... FFFFH	-5434... 5433.834	0.1658325	[hPa]
PUT controller D gain for stationary operation					
IP_FAC_PUT_CTL_D_DYN	10x8	0... FFFFH	0... 0.376881	5.7508e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_PUT_DIF_1_CHRG [8]	8	0... FFFFH	-5434... 5433.834	0.1658325	[hPa]
PUT controller D gain for non stationary operation					
IP_FAC_PUT_CTL_D_DYN_AT	10x8	0... FFFFH	0... 0.376881	5.7508e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_PUT_DIF_1_CHRG [8]	8	0... FFFFH	-5434... 5433.834	0.1658325	[hPa]
PUT controller D gain for non stationary operation with automatic transmission					
IP_FAC_PUT_CTL_I	6x10	0... FFFFH	0... 0.036805	561.61e-9	[1/(hPa*s)]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller I gain for stationary operation					
IP_FAC_PUT_CTL_I_DYN	6x10	0... FFFFH	0... 0.036805	561.61e-9	[1/(hPa*s)]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller I gain for non stationary operation					
IP_FAC_PUT_CTL_I_DYN_AT	6x10	0... FFFFH	0... 0.036805	561.61e-9	[1/(hPa*s)]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller I gain for non stationary operation with automatic transmission					
IP_FAC_PUT_CTL_P	6x10	0... FFFFH	0... 0.18844	2.8754e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller P gain for stationary opertion					
IP_FAC_PUT_CTL_P_DYN	6x10	0... FFFFH	0... 0.18844	2.8754e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller P gain for non stationary operation					
IP_FAC_PUT_CTL_P_DYN_AT	6x10	0... FFFFH	0... 0.18844	2.8754e-6	[1/hPa]
LDPM_N_1_CHRG [10]	10	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAF_2_CHRG [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
PUT controller P gain for non stationary operation with automatic transmission					
IP_PWM_WG_CLL	7	0... FFFFH	-100... 99.99695	3.0518e-3	[%]
LDP_FAC_POW_PUT_CTL_IP_PWM_CLL [7]	7	0... FFFFH	0... 7.999878	122.07e-6	[-]
Closed loop wastegate control signal					
IP_T_SYS_D	6x6	0... FFFFH	0... 4.19424	64e-6	[s]
LDPM_N_2_CHRG [6]	6	0... 1FE0H	0... 8160	1	[rpm]
LDP_MAF_SP_IP_T_SYS_D [6]	6	0... FFFFH	0... 1389	0.0211948	[mg/stk]
timeconstant for DT1 share					
LC_FAC_PUT_CTL_D_MAN	1	0... 1H	0... 1	1	[-]
Activation of manual PUT controller D gain					
LC_FAC_PUT_CTL_I_MAN	1	0... 1H	0... 1	1	[-]
Activation of manual PUT controller I gain					
LC_FAC_PUT_CTL_P_MAN	1	0... 1H	0... 1	1	[-]
Activation of manual PUT controller P gain					
LC_INH_PUT_CTL	1	0... 1H	0... 1	1	[-]
Inhibition of PUT control, sets PUT to minimum possible value					

LC_PUT_CTL_ACT_MAN	1	0... 1H	0... 1	1	[-]
Manual activation of PUT controller regardless of protection functions					
LC_PUT_CTL_I_LIM_ACT	1	0... 1H	0... 1	1	[-]
Activation of controller I-share freezing					
LC_PUT_CTL_STAT_AMP	1	0... 1H	0... 1	1	[-]
Switch for enabling the stationary mode in AMP					
LC_PUT_DIF_MMV_PUT_MMV	1	0... 1H	0... 1	1	[-]
Switch whether PUT_MMV is used for PUT_DIF_MMV calculation					
LC_RST_PUT_DIF	1	0... 1H	0... 1	1	[-]
Switch to reset PUT_DIF at inactive PUT controller					

General Information

The purpose of the present PUT controller is to minimize the deviation between the pressure upstream throttle setpoint (PUT_SP) and the actual pressure upstream throttle. When in charged engine operation the throttle is fully opened this PUT controller deviation determines the correspondence of MAF and MAF_SP as well as TQI and TQI_SP accordingly.

Application Conditions

Initialization: RST
Recurrence: 10MS
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

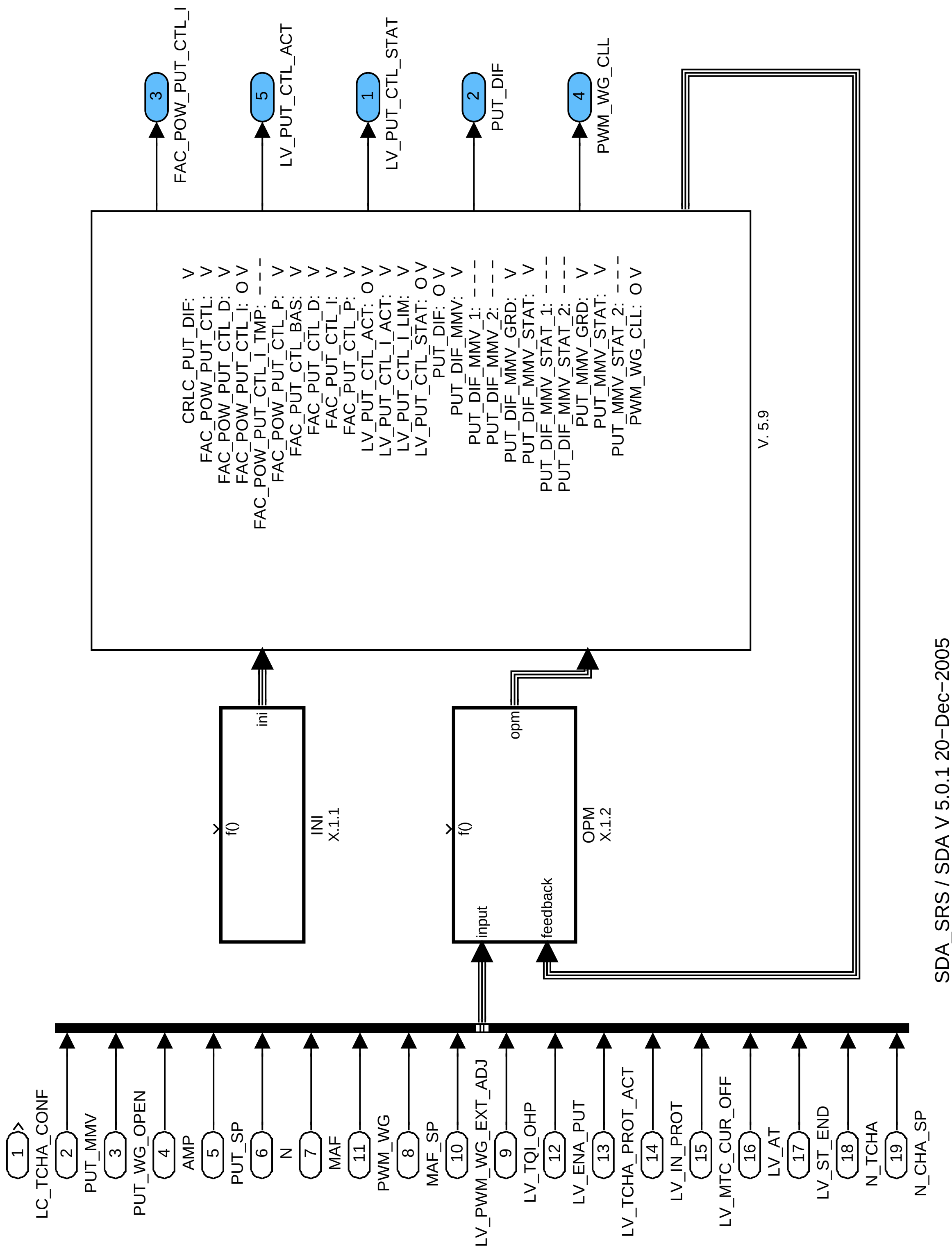


Figure 229:
Path: CHRG_REASPPUT0

9.82.1 CHRG_REASPPUT0/INI

CONTENT

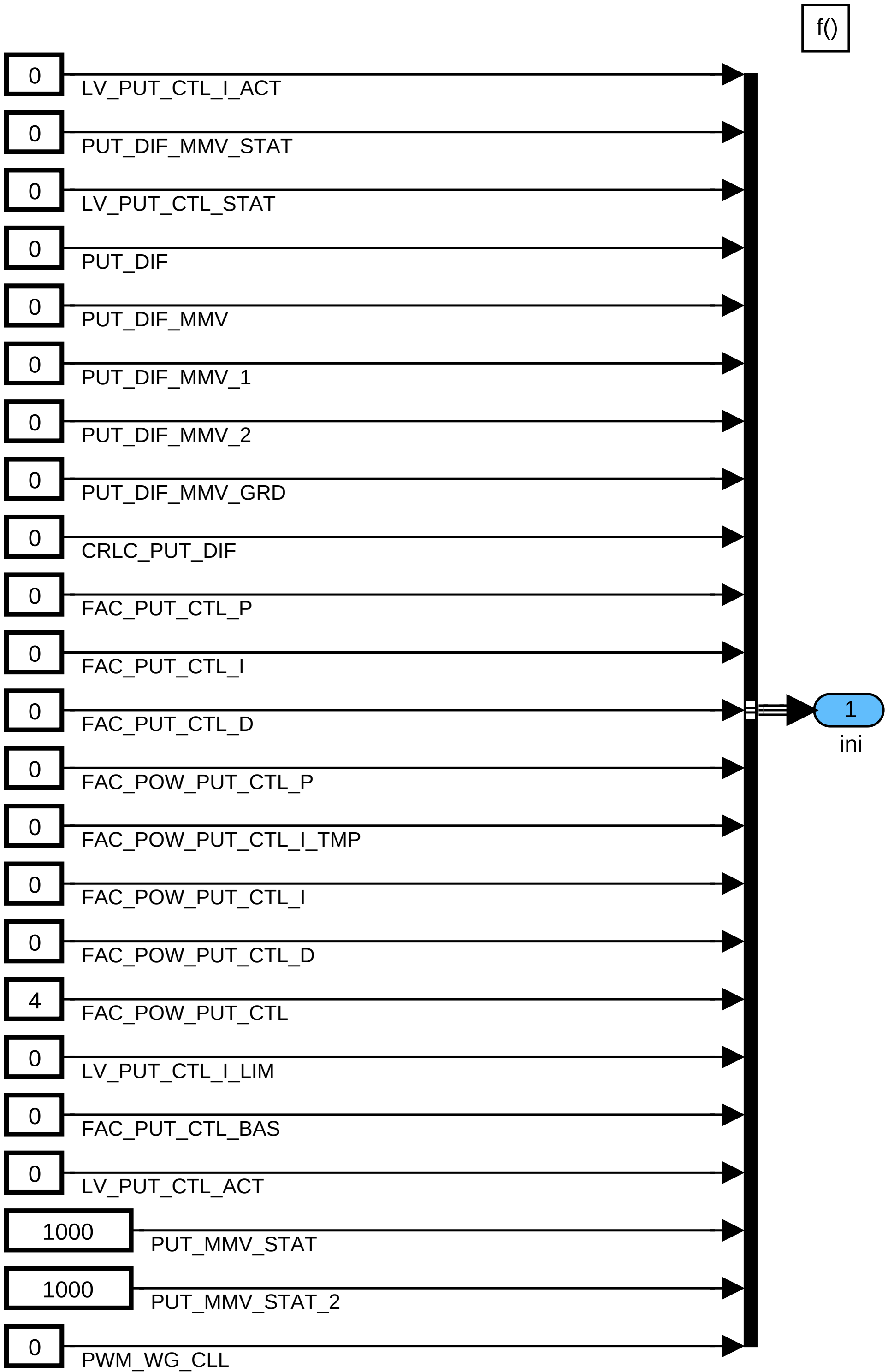


Figure 230:
Path: CHRG_REASPPUT0/INI

9.82.2 CHRG_REASPPUT0/OPM

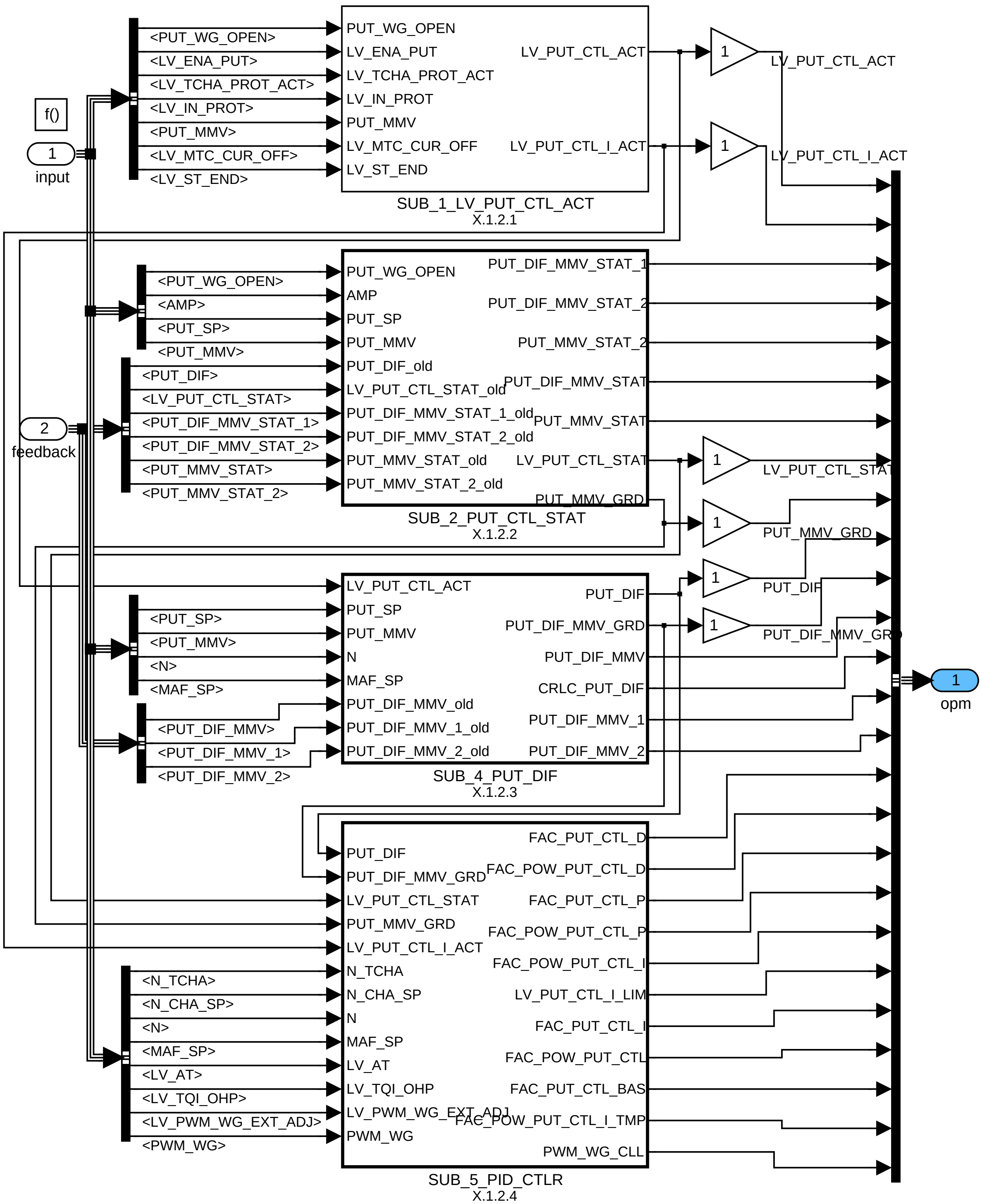


Figure 231:
Path: CHRG_REASPPUT0/OPM

9.82.2.1 CHRG_REASPPUT0/OPM/SUB_1_LV_PUT_CTL_ACT

LV_PUT_CTL_ACT masking

general specification

The I share of the PUT controller is activated only when both the general PUT controller activation conditions are fulfilled and when PUT is greater than the basic charge pressure PUT_WG_OPEN.

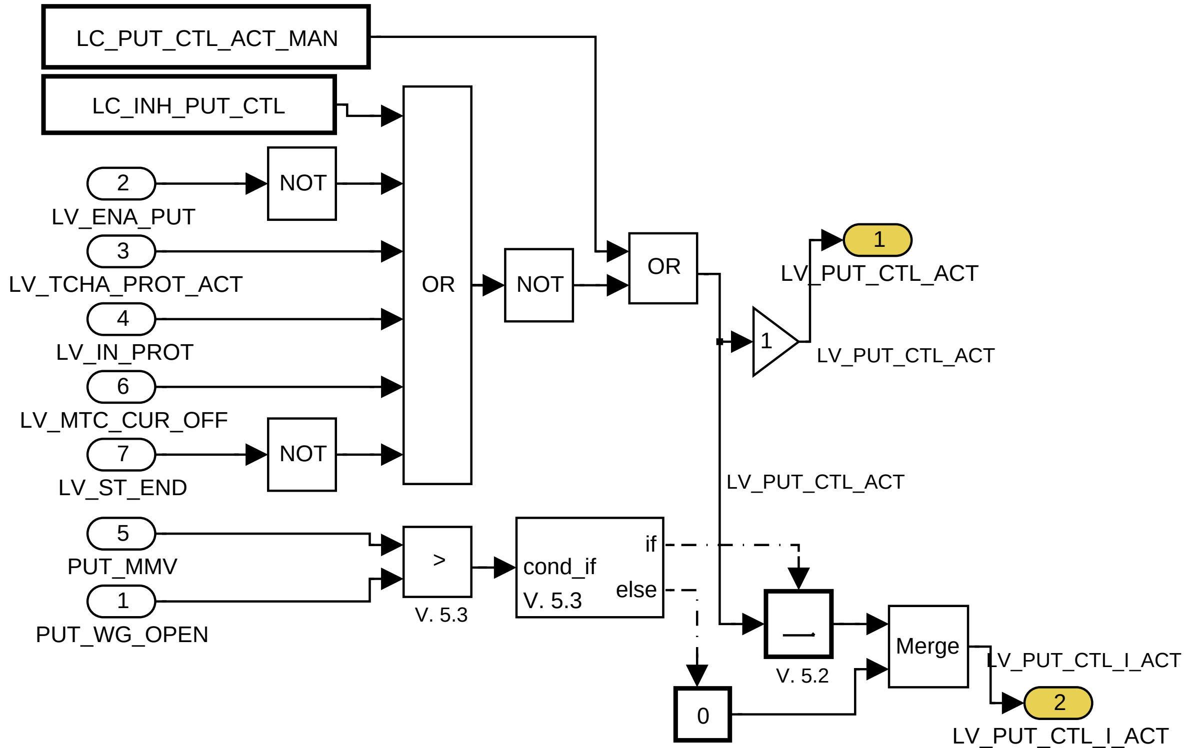


Figure 232:
Path: CHRG_REASPPUT0/OPM/SUB_1_LV_PUT_CTL_ACT

9.82.2.2 CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT

Recognition of stationary PID controller operation
Stationary charge air pressure controller operation is recognized if either
- PUT_DIF enters a window [C_PUT_DIF_Min_STAT_ACT, C_PUT_DIF_MIN_STAT_ACT] or
- PUT_DIF_MMV_STAT or PUT_MMV_STAT gradient changes its sign.

Non stationary charge air pressure controller operation is recognized if either
- PUT_DIF gets higher than C_PUT_DIF_MAX_STAT_DEAC or
- PUT_DIF gets smaller than C_PUT_DIF_MIN_STAT_DEAC.

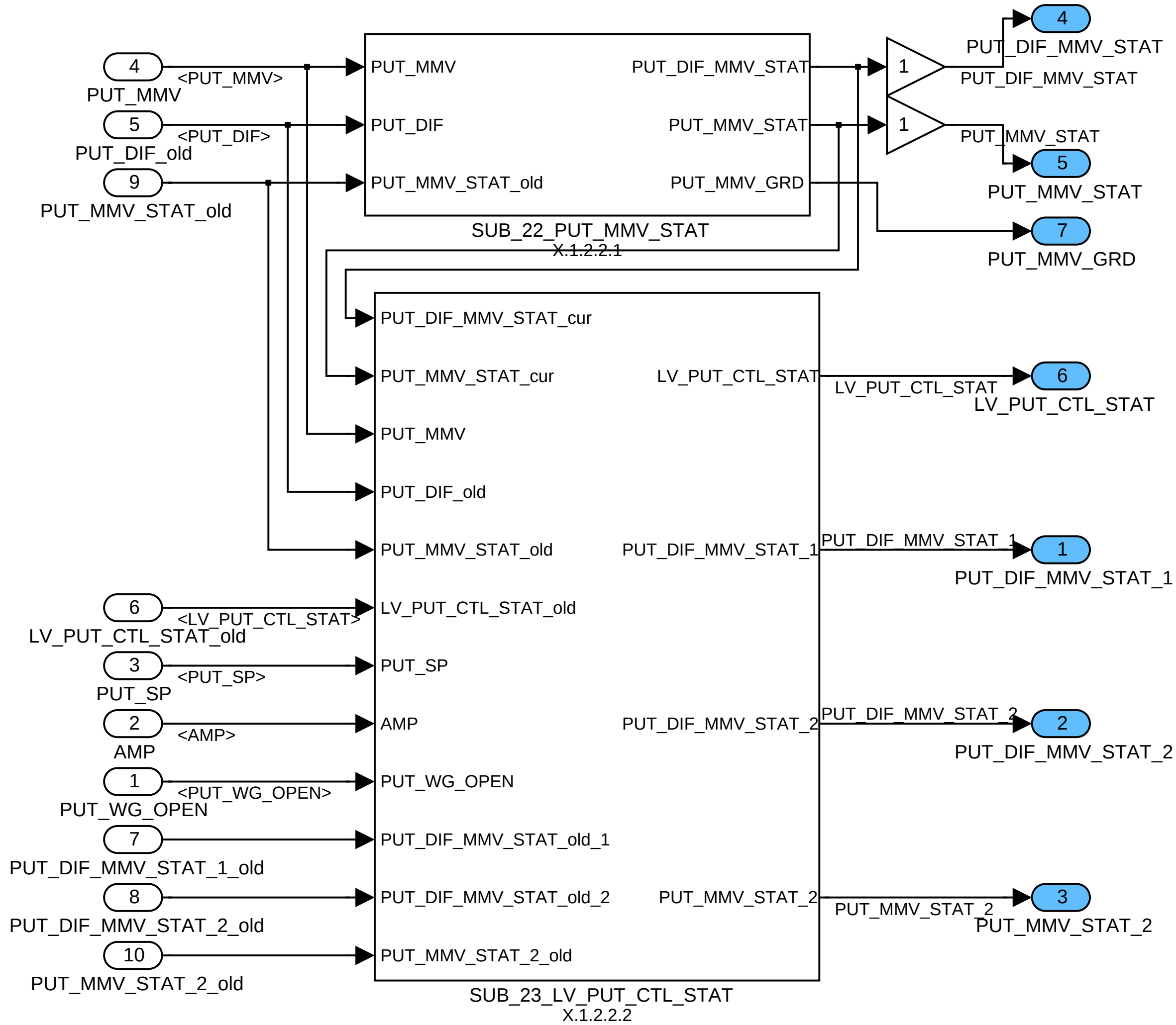


Figure 233:
Path: CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT

9.82.2.2.1 CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT/SUB_22_PUT_MMV_STAT

CONTENT

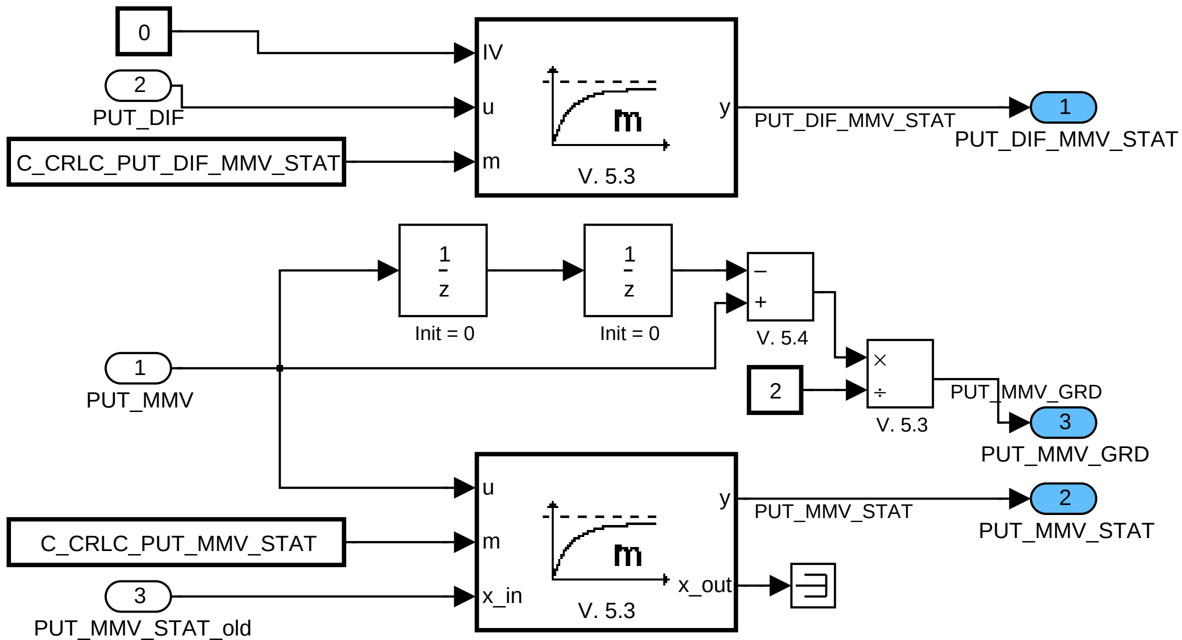


Figure 234:
Path: CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT/SUB_22_PUT_MMV_STAT

9.82.2.2.2 CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT/SUB_23_LV_PUT_CTL_STAT

CONTENT

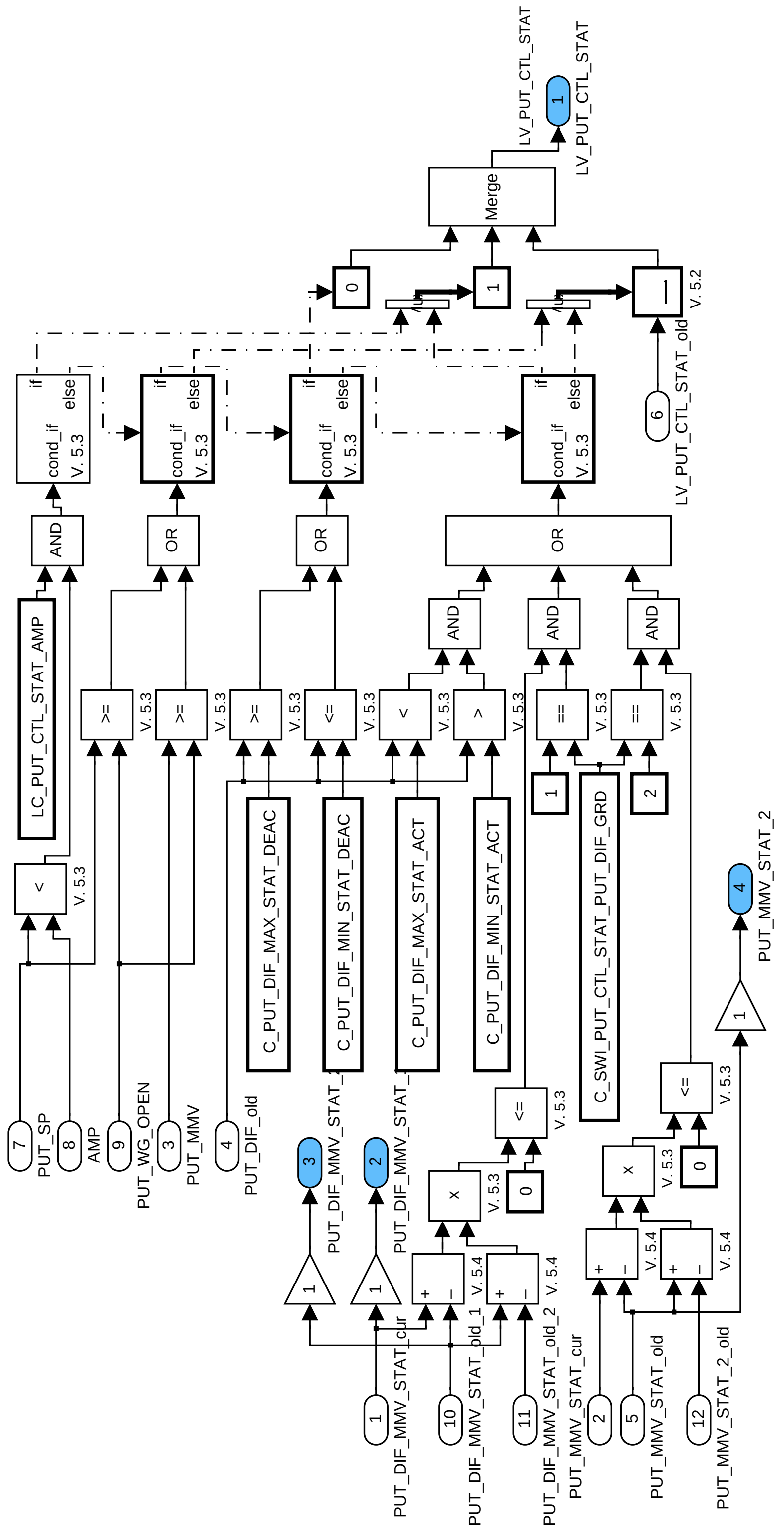


Figure 235:
Path: CHRG_REASPPUT0/OPM/SUB_2_PUT_CTL_STAT/SUB_23_LV_PUT_CTL_STAT

9.82.2.3 CHRG_REASPPUT0/OPM/SUB_4_PUT_DIF

CONTENT

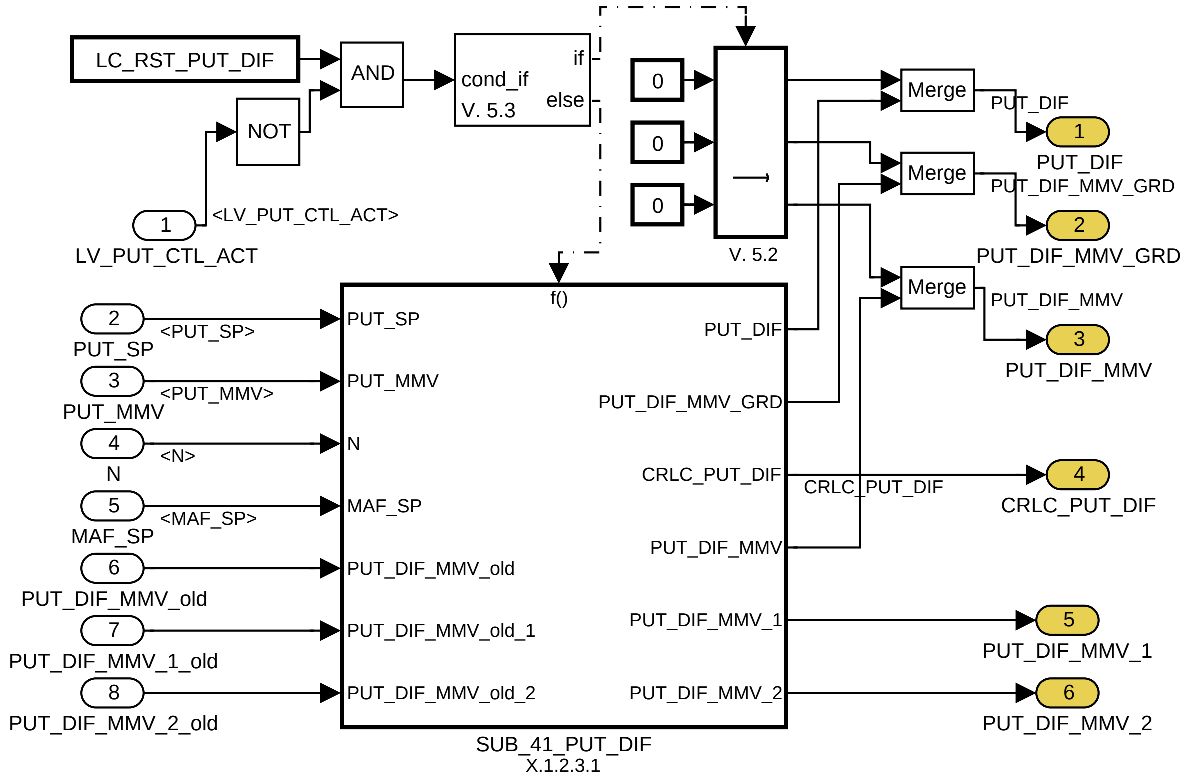


Figure 236:
Path: CHRG_REASPPUT0/OPM/SUB_4_PUT_DIF

9.82.2.3.1 CHRG_REASPPUT0/OPM/SUB_4_PUT_DIF/SUB_41_PUT_DIF

CONTENT

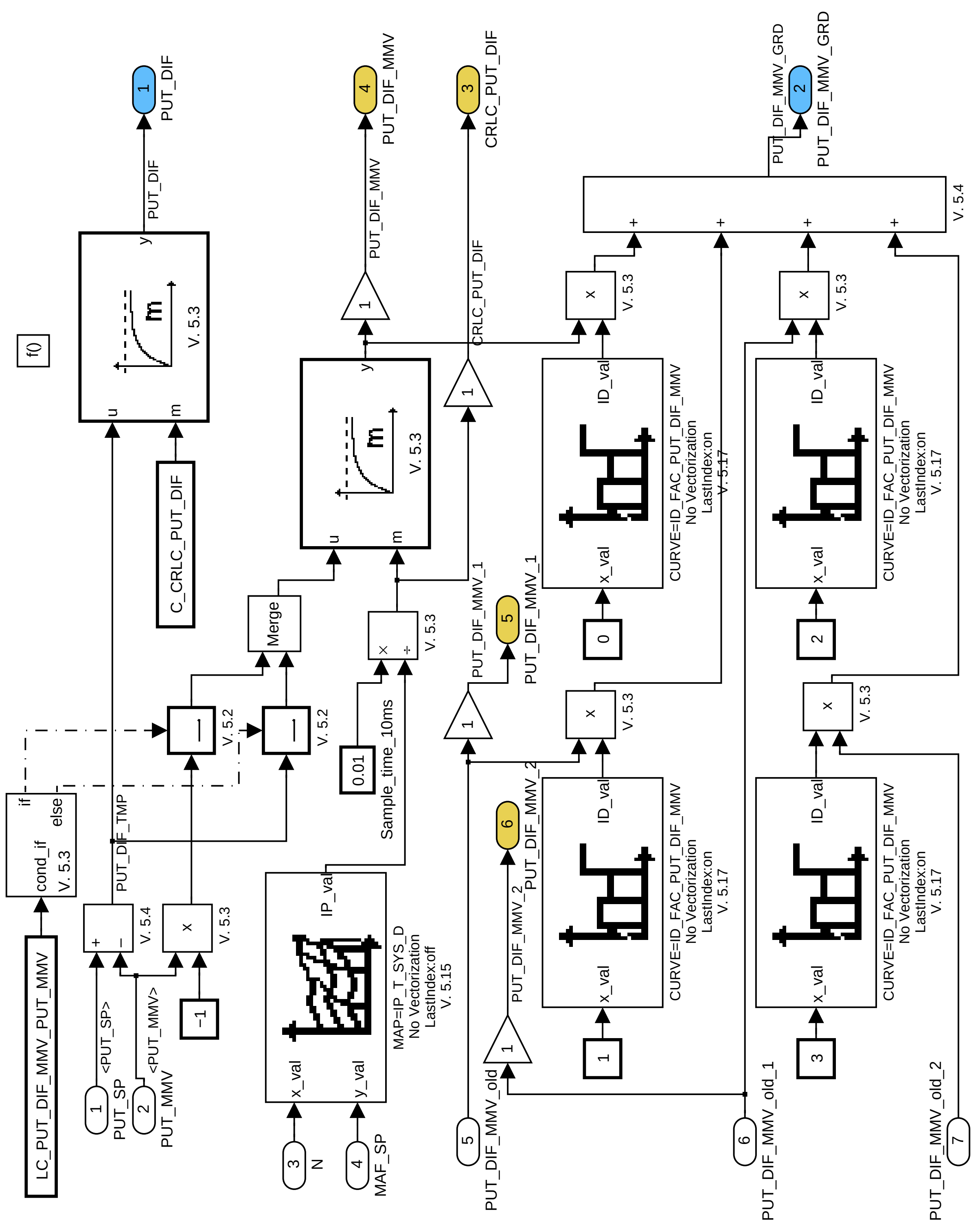


Figure 237:
Path: CHRG_REASPPUT0/OPM/SUB_4_PUT_DIF/SUB_41_PUT_DIF

9.82.2.4 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR

PID-Controller

The pressure up throttle controller is configured as a PID-controller in which the D-Part is inactive at steady state conditions. Each controller factor can be calibrated by a manual value instead of the

interpolation maps. The controller is stopped during engine torque limitation for turbo charger protection.
The controller gain depend on an auxiliary rotational speed N_PUT_CTL. By means of C_N_PUT_CTL_SRC can be chosen whether this rotational speed is engine speed N (= 0), actual charger speed N_TCHA (= 1) or charger speed setpoint N_CHA_SP (= 2).

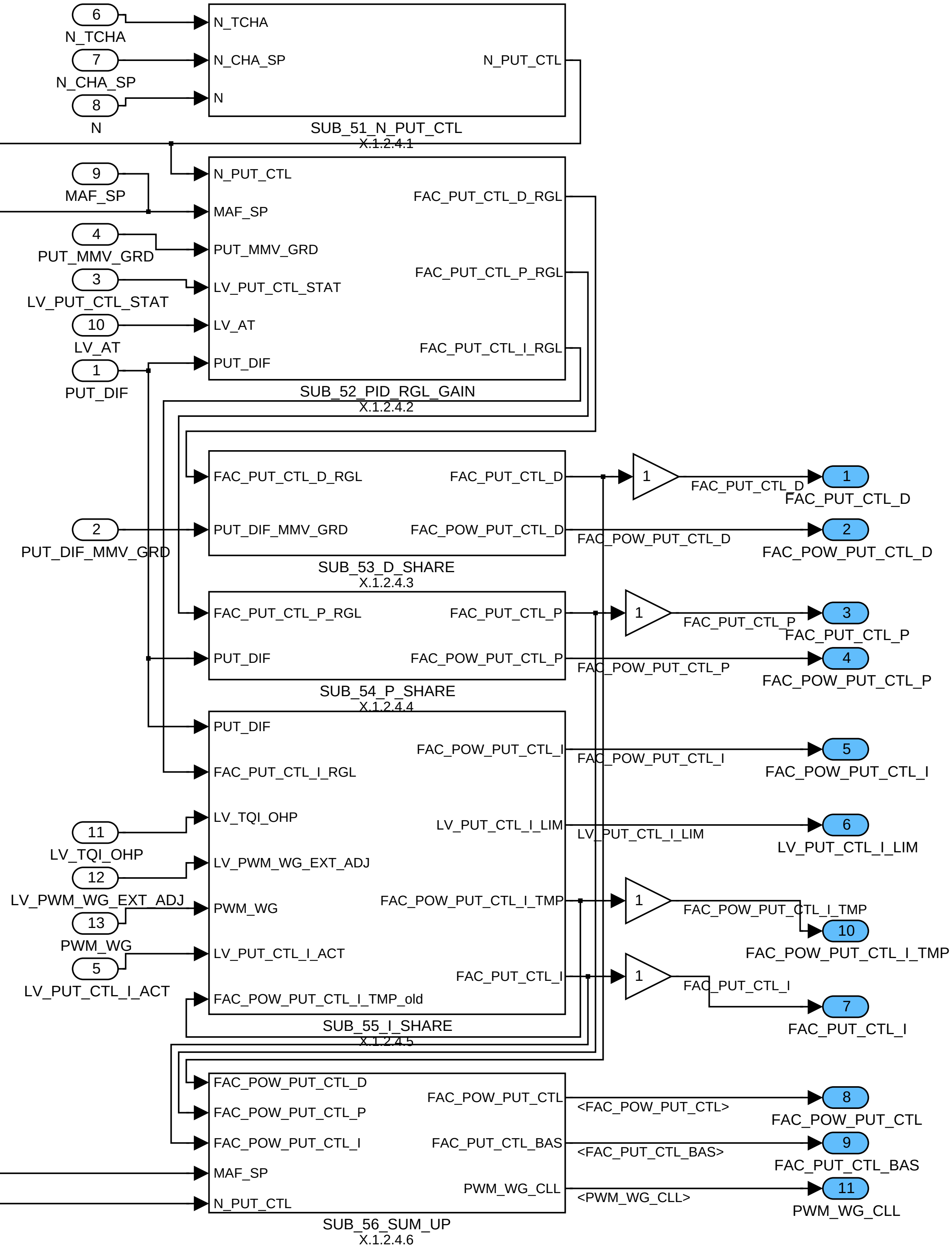


Figure 238:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR

9.82.2.4.1 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_51_N_PUT_CTL

CONTENT

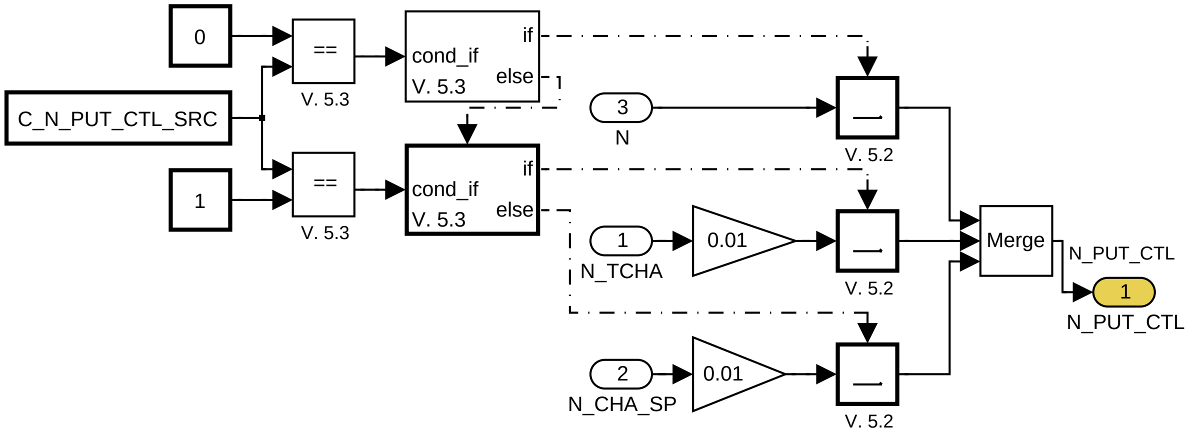


Figure 239:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_51_N_PUT_CTL

9.82.2.4.2 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_52_PID_RGL_GAIN

There are three sets of controller gains
- one for stationary operation
- one for non stationary operation with manual transmission
- one for non stationary operation with automatic transmission.

There is an additional factor for the I gain. In non stationary operation the I share can work symmetrically (C_FAC_PUT_CTL_I_COR_POS = 1) for both negative and positive PUT_DIF. To prevent PUT overshoots it can be limited to work for negative PUT_DIF (i.e. PUT_SP > PUT_MMV) only (C_FAC_PUT_CTL_I_COR_POS = 1). To prevent PUT overshoots for PUT_SP jumps which do not trigger non stationary operation, the I share is deactivated for PUT_MMV gradients bigger than C_PUT_MMV_GRD_MAX_PUT_CTL_I.

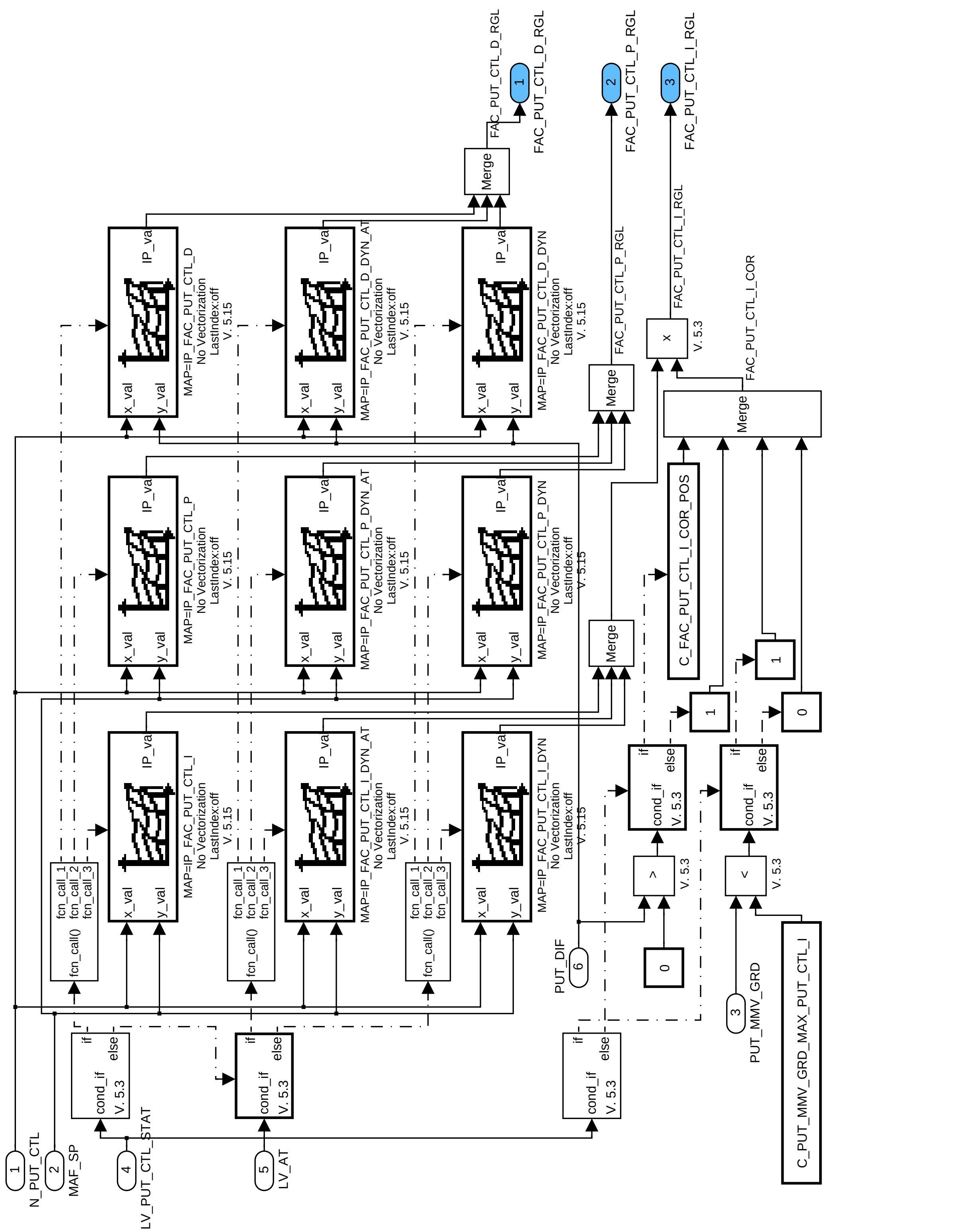


Figure 240:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_52_PID_RGL_GAIN

9.82.2.4.3 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_53_D_SHARE

CONTENT

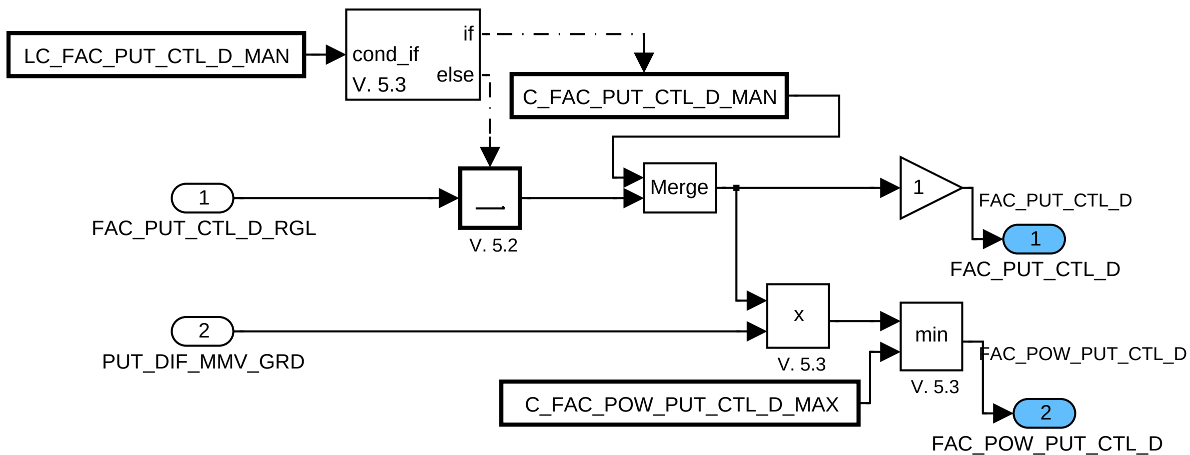


Figure 241:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_53_D_SHARE

9.82.2.4.4 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_54_P_SHARE

CONTENT

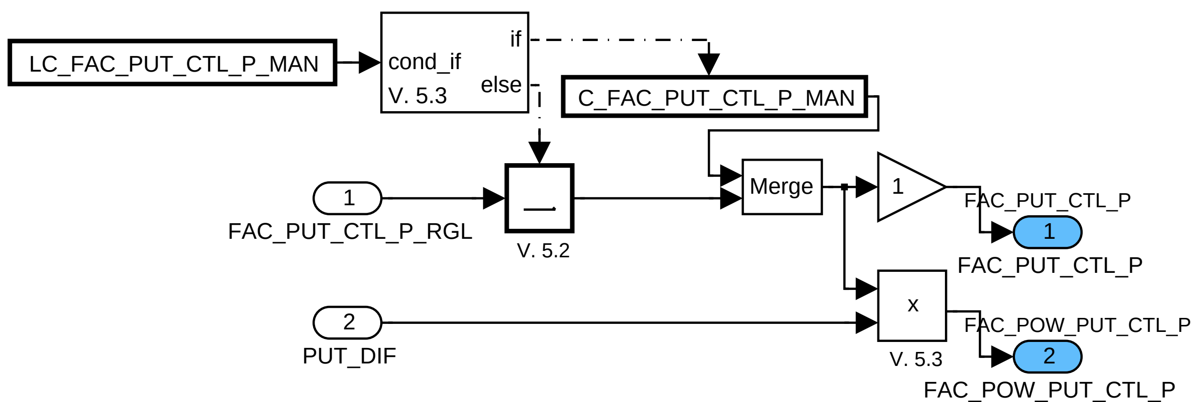


Figure 242:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_54_P_SHARE

9.82.2.4.5 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE

If the I share of the controller is deactivated its value is ramped to 0 with a gradient C_FAC_POW_PUT_CTL_RAMP_DOWN.

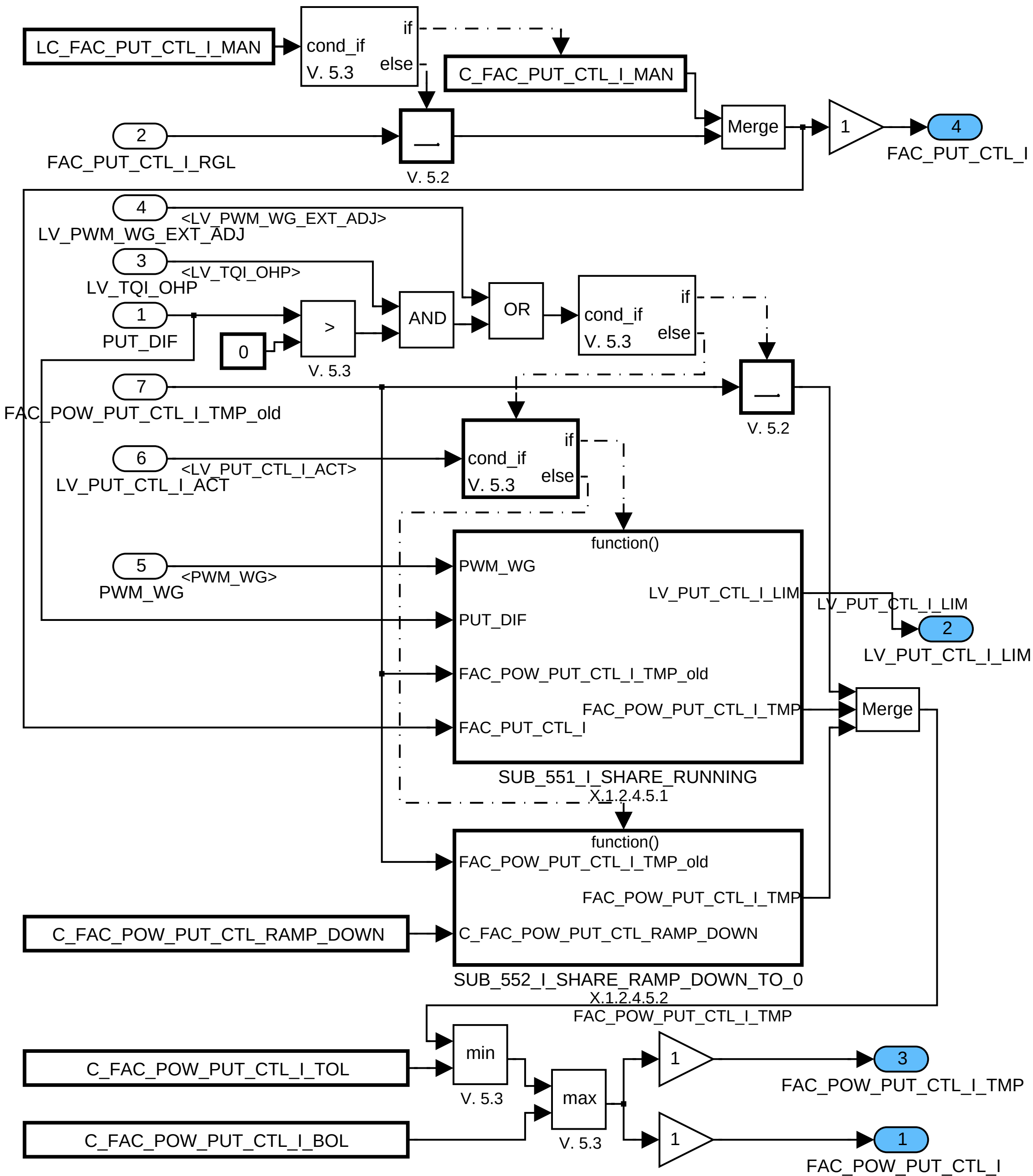


Figure 243:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE

9.82.2.4.5.1 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE/SUB_551_I_SHARE_RUNNING

CONTENT

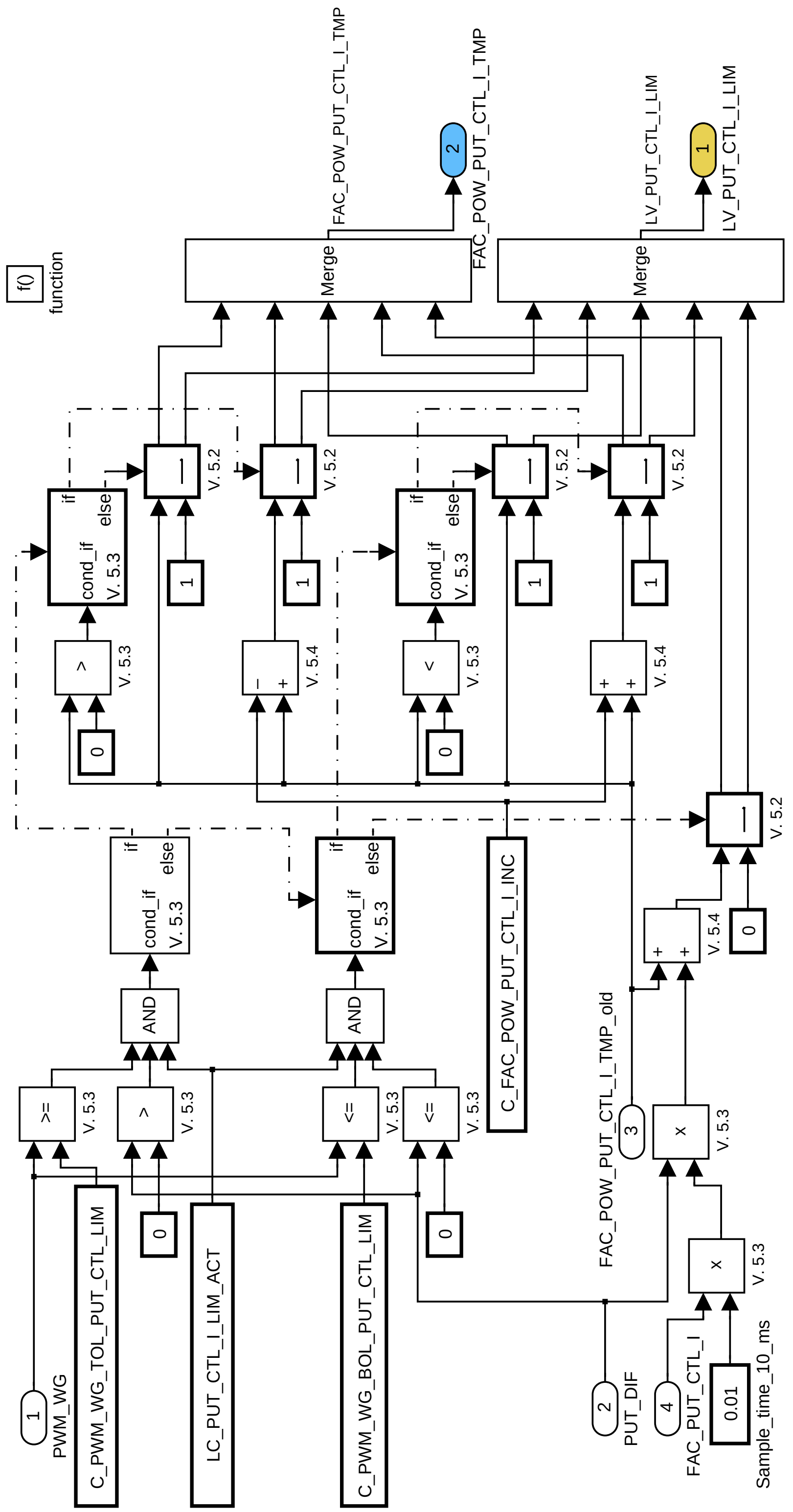


Figure 244:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE/SUB_551_I_SHARE_RUNNING

9.82.2.4.5.2 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE/SUB_552_I_SHARE_RAMP_DOWN_TO_0

CONTENT

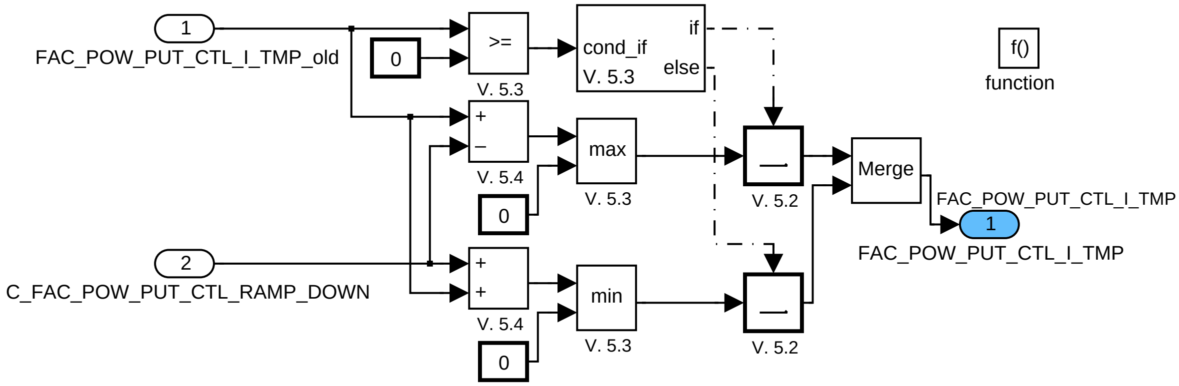


Figure 245:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_55_I_SHARE/SUB_552_I_SHARE_RAMP_DOWN_TO_0

9.82.2.4.6 CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_56_SUM_UP

CONTENT

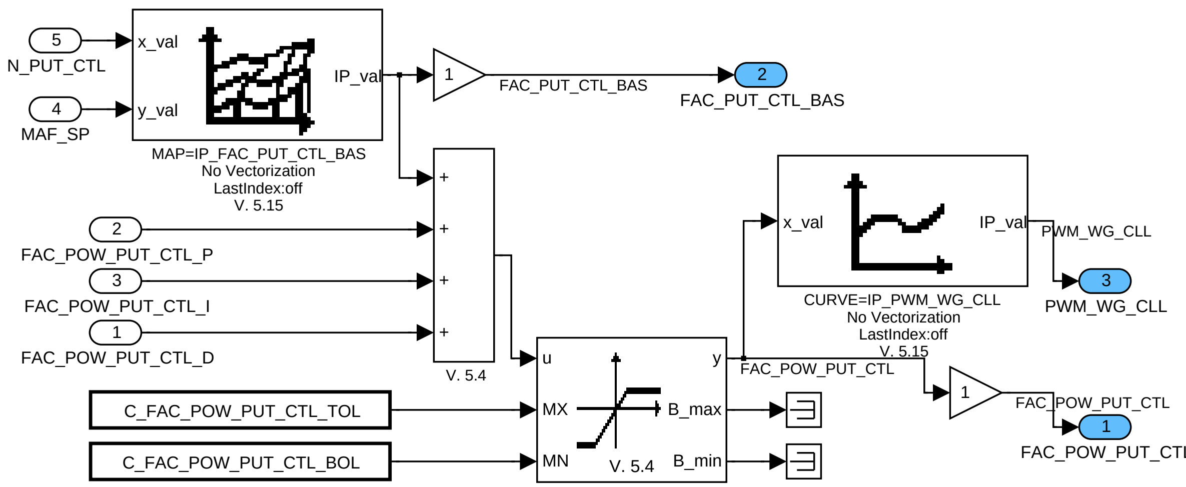


Figure 246:
Path: CHRG_REASPPUT0/OPM/SUB_5_PID_CTLR/SUB_56_SUM_UP

9.83 Charge air pressure control adaptation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_PUT_AD	V	0...FFFFH	0...65535	1	[-]
PUT controller adaptation counter					
T_DLY_PUT_AD	V	0...FFH	0...2.55	0.01	[s]
Delay time for PUT_CTL adaptation					
N_MMV_PUT_AD	V	0...1FE0H	0...8160	1	[rpm]
filtered N for PUT_CTL adaptation					
MAF_MMV_PUT_AD	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
filtered MAF for PUT_CTL adaptation					
MAF_SP_MMV_PUT_AD	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
filtered MAF_SP for PUT_CTL adaptation					
PUT_DIF_MMV_PUT_AD	V	8000...7FFFH	-5434... 5433.83416	0.1658325	[hPa]
filtered PUT control deviation for PUT_CTL adaptation					
MAF_PUT_AD_MIN	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
minimum MAF in the evaluation time window for PUT control adaptation					
MAF_PUT_AD_MAX	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
maximum MAF in the evaluation time window for PUT control adaptation					
MAF_SP_PUT_AD_MIN	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
minimum MAF_SP in the evaluation time window for PUT control adaptation					
MAF_SP_PUT_AD_MAX	V	0...FFFFH	0...1389	0.0211948	[mg/stk]
maximum MAF_SP in the evaluation time window for PUT control adaptation					
N_PUT_AD_MAX	V	0...1FE0H	0...8160	1	[rpm]
maximum engine speed in the evaluation time window for PUT control adaptation					
N_PUT_AD_MIN	V	0...1FE0H	0...8160	1	[rpm]
minimum engine speed in the evaluation time window for PUT control adaptation					
PRS_DIF_WG_ACR_SP_AD_ADD	O/V	8000...7FFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
adapted additive value for PRS_DIF_WG_ACR_SP					
PRS_PUT_AD	V	0...FFFFH	0...5434	0.0829175	[hPa]
filtered PRS_EX_PRS_WG_ACR_SP for PUT_CTL adaptation					
FLOW_PUT_AD	V	0...FFFFH	0...2047.96875	0.03125	[kg/h]
filtered FLOW_WG_SP for PUT_CTL adaptation					
PRS_DIF_WG_ACR_PUT_AD	V	8000...7FFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Auxiliary pressure control deviation for PUT_CTL adaptation					
PRS_DIF_WG_ACR_SP_AD[9][9]	O/V/S	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
map for adapted values for PUT_CTL adaptation					
PRS_DIF_WG_ACR_SP_AD_AS[9][9]	V	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Copy (for application purposes) of map for adapted values for PUT_CTL adaptation					
IDX_PRS_PUT_AD	V	0...FFH	0...255	1	[-]
Pressure index for PUT_CTL adaptation					

IDX_FLOW_PUT_AD	V	0...FFH	0...255	1	[-]
Flow index for PUT_CTL adaptation					

Input data:

FLOW_WG_SP	LV_PUT_CTL_I_ACT	MAF	MAF_SP
N	PRS_EX_PRS_WG_ACR_SP	PRS_WG_ACR	PRS_WG_ACR_SP
PUT_DIF	PWM_WG	TCO	TIA_THR
LV_ST_END	PUT_SP	PUT_MMV	PUT_WG_OPEN

FUNCTION DESCRIPTION:

General information:

An adaptation is carried out to compensate series tolerances. The values of map PRS_DIF_WG_ACR_SP_AD are saved in EEPROM during power latch phase and reread at ECU reset. The indices IDX_PRS_PUT_AD and IDX_FLOW_PUT_AD are used to address PRS_DIF_WG_ACR_SP_AD and the axes LDPM_PRS_1_CHRG and LDPM_FLOW_1_CHRG.

Application conditions:

Initialisation: at ECU reset:

MAF_MMV_PUT_AD = 0
N_MMV_PUT_AD = 0
PUT_DIF_MMV_PUT_AD = 0
MAF_SP_MMV_PUT_AD = 0
PRS_PUT_AD = 1013 hPa
FLOW_PUT_AD = 0
PRS_DIF_WG_ACR_SP_AD_ADD = 0
CTR_PUT_AD = 0
PRS_DIF_WG_ACR_PUT_AD = 0
T_DLY_PUT_AD = C_T_DLY_PUT_AD_INI

if (checksum error) **or** LC_AD_CLR_PUT_CTL **then**
PRS_DIF_WG_ACR_SP_AD [...][...] = 0
endif

Activation: LC_TCHA_CONF = 1 **and** LV_ST_END = 1

Deactivation: LC_TCHA_CONF = 0 **or** LV_ST_END = 0

Formula section:

Moving mean value and min/max calculations to detect steady engine operation

MAF_MMV_PUT_AD $_{n}$ =
MAF_MMV_PUT_AD $_{n-1}$ + C_CRLC_PUT_AD * (MAF - MAF_MMV_PUT_AD $_{n-1}$)
MAF_PUT_AD_MAX $_{n}$ = max (MAF_PUT_AD_MAX $_{n-1}$, MAF_MMV_PUT_AD)
MAF_PUT_AD_MIN $_{n}$ = min (MAF_PUT_AD_MIN $_{n-1}$, MAF_MMV_PUT_AD)

N_MMV_PUT_AD $_{n}$ = N_MMV_PUT_AD $_{n-1}$ + C_CRLC_PUT_AD * (N - N_MMV_PUT_AD $_{n-1}$)
N_PUT_AD_MAX $_{n}$ = max (N_PUT_AD_MAX $_{n-1}$, N_MMV_PUT_AD)
N_PUT_AD_MIN $_{n}$ = min (N_PUT_AD_MIN $_{n-1}$, N_MMV_PUT_AD)

MAF_SP_MMV_PUT_AD $_{n}$ = MAF_SP_MMV_PUT_AD $_{n-1}$ +

C_CRLC_PUT_AD * (MAF_SP - MAF_SP_MMV_PUT_AD $_{n-1}$)
MAF_SP_PUT_AD_MAX $_n$ = max (MAF_SP_PUT_AD_MAX $_{n-1}$, MAF_SP_MMV_PUT_AD)
MAF_SP_PUT_AD_MIN $_n$ = min (MAF_SP_PUT_AD_MIN $_{n-1}$, MAF_SP_MMV_PUT_AD)

PUT_DIF_MMV_PUT_AD $_n$ = PUT_DIF_MMV_PUT_AD $_{n-1}$ +
C_CRLC_PUT_AD * (PUT_DIF - PUT_DIF_MMV_PUT_AD $_{n-1}$)

PRS_DIF_WG_ACR_PUT_AD = PRS_WG_ACR – PRS_WG_ACR_SP

FLOW_PUT_AD $_n$ = FLOW_PUT_AD $_{n-1}$ + C_CRLC_PUT_AD * (FLOW_WG_SP – FLOW_PUT_AD $_{n-1}$)

PRS_PUT_AD $_n$ = PRS_PUT_AD $_{n-1}$ +
C_CRLC_PUT_AD * (PRS_EX_PRS_WG_ACR_SP - PRS_PUT_AD $_{n-1}$)

Read adaptation value

// PRS_DIF_WG_ACR_SP_AD [...] [...] is the value from the adaptation map
// IP_PRS_DIF_WG_ACR_SP_AD (PRS_PUT_AD, FLOW_PUT_AD),
// which is defined as a dummy IP in the calibration data.

PRS_DIF_WG_ACR_SP_AD_ADD =
PRS_DIF_WG_ACR_SP_AD (PRS_EX_PRS_WG_ACR_SP, FLOW_WG_SP)

Adaptation of PRS_DIF_WG_ACR_SP_AD

By means of the tables ID_PRS_PUT_AD_MIN and ID_PRS_PUT_AD_MAX several adaptation-windows (indexed by idx) ID_PRS_PUT_AD_MIN [idx] < LDPM_PRS_1_CHRG [idx] < ID_PRS_PUT_AD_MAX [idx]. If PRS_PUT_AD is in one of these windows, that is ID_PRS_PUT_AD_MIN [idx] <= PRS_PUT_AD <= ID_PRS_PUT_AD_MAX [idx] for a given index idx, an adaptation of the map PRS_DIF_WG_ACR_SP_AD at LDPM_PRS_1_CHRG [idx] is possible. If such a window can be found at the index idx, we set IDX_PRS_PUT_AD = idx, otherwise IDX_PRS_PUT_AD is set to FFh.
The search for the window is performed by means of the data-point-search implemented in the Software-library: For a given PRS_PUT_AD, the data point search for the axis LDPM_PRS_1_CHRG returns an index idx_sw such that LDPM_PRS_1_CHRG [idx_sw] <= PRS_PUT_AD < LDPM_PRS_1_CHRG [idx_sw+1]. If PRS_PUT_AD is smaller or bigger than any value of the axis, idx_sw is set to the first or last index respectively.

By means of the tables ID_FLOW_PUT_AD_MIN and ID_FLOW_PUT_AD_MAX several adaptation-windows (indexed by idx) ID_FLOW_PUT_AD_MIN [idx] < LDPM_FLOW_1_CHRG [idx] < ID_FLOW_PUT_AD_MAX [idx]. If FLOW_PUT_AD is in one of these windows, that is ID_FLOW_PUT_AD_MIN [idx] <= FLOW_PUT_AD <= ID_FLOW_PUT_AD_MAX [idx] for a given index idx, an adaptation of the map PRS_DIF_WG_ACR_SP_AD at LDPM_FLOW_1_CHRG [idx] is possible. If such a window can be found at the index idx, we set IDX_FLOW_PUT_AD = idx, otherwise IDX_FLOW_PUT_AD is set to FFh.
The search for the window is performed by means of the data-point-search implemented in the Software-library:: For a given FLOW_PUT_AD, the data point search for the axis LDPM_FLOW_1_CHRG returns an index idx_sw such that LDPM_FLOW_1_CHRG [idx_sw] <= FLOW_PUT_AD < LDPM_FLOW_1_CHRG [idx_sw+1]. If FLOW_PUT_AD is smaller or bigger than any value of the axis, idx_sw is set to the first or last index respectively.

(1) if T_DLY_PUT_AD = 0 **then** // delay time has elapsed

// search for current pressure index of adaptation map

IDX_PRS_PUT_AD = FFh

// idx_sw is calculated by the software data point search algorithm for the axis
// LDPM_PRS_1_CHRG at PRS_PUT_AD. The software data point search algorithm
// delivers indices starting from 0

(2) for idx_sw = 0 to (length (LDPM_PRS_1_CHRG)-1) **do**

(3) if (LDPM_PRS_1_CHRG [idx_sw] <= PRS_PUT_AD <
LDPM_PRS_1_CHRG [idx_sw+1]) **then**

break

(3) endif

(2) endfor

// test whether current PRS_PUT_AD is within one of the two possible adaptation windows

(4) if (ID_PRS_PUT_AD_MN [idx_sw] <= PRS_PUT_AD <=
ID_PRS_PUT_AD_MAX [idx_sw]) **then**

IDX_PRS_PUT_AD = idx_sw

(4) endif

(41) if (idx_sw < (length (LDPM_PRS_1_CHRG)-1) **then**

(5) if (ID_PRS_PUT_AD_MN [idx_sw+1] <= PRS_PUT_AD <=
ID_PRS_PUT_AD_MAX [idx_sw+1]) **then**

IDX_PRS_PUT_AD = idx_sw+1

(5) endif

(41) endif

// search for current flow index of adaptation map

IDX_FLOW_PUT_AD = FFh

// idx_sw is calculated by the software data point search algorithm for the axis
// LDPM_FLOW_1_CHRG at FLOW_PUT_AD. The software data point search algorithm
// delivers indices starting from 0

(6) for idx_sw = 0 to (length (LDPM_FLOW_1_CHRG)-1) **do**

(7) if (LDPM_FLOW_1_CHRG [idx_sw] <= FLOW_PUT_AD <=
LDPM_FLOW_1_CHRG [idx_sw+1]) **then**

break

(7) endif

(6) endfor

// test whether current FLOW_PUT_AD is within one of the two possible
// adaptation windows

(8) if (ID_FLOW_PUT_AD_MIN [idx_sw] <= FLOW_PUT_AD <=
ID_FLOW_PUT_AD_MAX [idx_sw]) **then**

IDX_FLOW_PUT_AD = idx_sw

(8) endif

(81) if (idx_sw < (length (LDPM_FLOW_1_CHRG)-1) **then**

(9) if (ID_FLOW_PUT_AD_MIN [idx_sw+1] <= FLOW_PUT_AD <=

```
            ID_FLOW_PUT_AD_MAX [idx_sw+1]) then
            IDX_FLOW_PUT_AD = idx_sw+1
        (9) endif
    (81) endif
    (10) if IDX_PRS_PUT_AD < FFh and IDX_FLOW_PUT_AD < FFh and
        abs (PRS_DIF_WG_ACR_PUT_AD) > C_PRS_DIF_WG_ACR_SP_AD_INC then
        // a fitting window for adaptation has been found and adaptation input is greater
        // than one increment of the adaptation map PRS_DIF_WG_ACR_SP_AD
        (11) if LC_PUT_CTL_AD_UPD_MOD then
            // update adaptation point and all elements above in column
            IDX_FLOW_PUT_AD_INI = 0
        (11) else
            // update adaptation point only (default)
            IDX_FLOW_PUT_AD_INI = IDX_FLOW_PUT_AD
        (11) endif
        (12) for IDX_FLOW_PUT_AD_TMP =
            IDX_FLOW_PUT_AD_INI to IDX_FLOW_PUT_AD do
            PRS_DIF_WG_ACR_SP_AD_TMP = PRS_DIF_WG_ACR_SP_AD
                (IDX_PRS_PUT_AD, IDX_FLOW_PUT_AD_TMP)
            (13) if PRS_DIF_WG_ACR_PUT_AD > 0 then
                // find minimum adjacent entry
                PRS_DIF_WG_ACR_SP_AD_MIN = 2716.917 hPa
                (14) if LC_PUT_CTL_AD_DIF_LIM_ACT then
                    (15) for IDX_PRS_PUT_AD_TMP_2 = max (1, IDX_PRS_PUT_AD - 1) to
                        min (length (LDPM_PRS_1_CHRG), IDX_PRS_PUT_AD + 1) do
                        (16) for IDX_FLOW_PUT_AD_TMP_2 =
                            max (1, IDX_FLOW_PUT_AD_TMP - 1) to min (length
                            (LDPM_FLOW_1_CHRG), IDX_FLOW_PUT_AD_TMP + 1) do
                            PRS_DIF_WG_ACR_SP_AD_MIN = min (PRS_DIF_WG_ACR_SP_AD
                                    (IDX_PRS_PUT_AD_TMP_2, IDX_FLOW_PUT_AD_TMP_2),
                                    PRS_DIF_WG_ACR_SP_AD_MIN)
                        (16) endfor
                    (15) endfor
                (14) endif
                // PRS_DIF_WG_ACR_SP_AD_MIN is now equal to the smallest adjacent entry
                (17) if PRS_DIF_WG_ACR_SP_AD_TMP - PRS_DIF_WG_ACR_SP_AD_MIN
                    < C_PRS_DIF_WG_ACR_SP_AD_GRD_MAX and
                    PRS_DIF_WG_ACR_SP_AD_TMP < C_PRS_DIF_WG_ACR_SP_AD_TOL
                    then        // maximum allowed difference between adjacent entries not
                    // reached and top limit of adaptation range not reached
                    CTR_PUT_AD  = CTR_PUT_AD + 1
                    // modify adaptation map
                    PRS_DIF_WG_ACR_SP_AD (IDX_PRS_PUT_AD,
```

IDX_FLOW_PUT_AD_TMP) = PRS_DIF_WG_ACR_SP_AD_TMP + C_PRS_DIF_WG_ACR_SP_AD_INC

(17) endif

(13) else

// find maximum adjacent entry
PRS_DIF_WG_ACR_SP_AD_MAX = -2717.04145 hPa

(18) if LC_PUT_CTL_AD_DIF_LIM_ACT **then**

(19) for IDX_PRS_PUT_AD_TMP_2 = max (1, IDX_PRS_PUT_AD - 1) **to** min (length (LDPM_PRS_1_CHRG), IDX_PRS_PUT_AD + 1) **do**

(20) for IDX_FLOW_PUT_AD_TMP_2 = max (1, IDX_FLOW_PUT_AD_TMP - 1) **to** min (length (LDPM_FLOW_1_CHRG), IDX_FLOW_PUT_AD_TMP + 1) **do**

PRS_DIF_WG_ACR_SP_AD_MAX = max(PRS_DIF_WG_ACR_SP_AD (IDX_PRS_PUT_AD_TMP_2, IDX_FLOW_PUT_AD_TMP_2), PRS_DIF_WG_ACR_SP_AD_MAX)

(20) endfor

(19) endfor

(18) endif

// PRS_DIF_WG_ACR_SP_AD_MAX is now equal to the greatest adjacent entry

(21) if PRS_DIF_WG_ACR_SP_AD_MAX - PRS_DIF_WG_ACR_SP_AD_TMP < C_PRS_DIF_WG_ACR_SP_AD_GRD_MAX **and** PRS_DIF_WG_ACR_SP_AD_TMP > C_PRS_DIF_WG_ACR_SP_AD_BOL **then** // maximum allowed difference between adjacent entries not **//** reached and bottom limit of adaptation range not reached

CTR_PUT_AD = CTR_PUT_AD + 1

// modify adaptation map
PRS_DIF_WG_ACR_SP_AD (IDX_PRS_PUT_AD, IDX_FLOW_PUT_AD_TMP) = PRS_DIF_WG_ACR_SP_AD_TMP – C_PRS_DIF_WG_ACR_SP_AD_INC

(21) endif

(13) endif

(12) endfor

(10) endif

T_DLY_PUT_AD = C_T_DLY_PUT_AD_INI

(1) else

T_DLY_PUT_AD = T_DLY_PUT_AD - 0.01 s

(1)endif

Moving mean value and min/max reset

(22) if T_DLY_PUT_AD >= C_T_DLY_PUT_AD_INI **or** LV_PUT_CTL_I_ACT = 0 **or** LV_PUT_CTL_I_LIM = 1 **or** TIA_THR < C_TIA_MIN_PUT_AD **or** TIA_THR > C_TIA_MAX_PUT_AD **or**

TCO < C_TCO_MIN_PUT_AD **or** TCO > C_TCO_MAX_PUT_AD **or**
abs (PUT_DIF_MMV_PUT_AD) > C_PUT_DIF_MAX_PUT_AD **or**
MAF_PUT_AD_MAX – MAF_PUT_AD_MIN >= C_MAF_WIN_PUT_AD **or**
MAF_SP_PUT_AD_MAX – MAF_SP_PUT_AD_MIN >= C_MAF_SP_WIN_PUT_AD **or**
N_PUT_AD_MAX – N_PUT_AD_MIN >= C_N_WIN_PUT_AD **or**
PUT_SP < PUT_WG_OPEN **or** PUT_MMV < PUT_WG_OPEN **then**

T_DLY_PUT_AD = C_T_DLY_PUT_AD_INI
MAF_PUT_AD_MAX = MAF_MMV_PUT_AD
MAF_PUT_AD_MIN = MAF_MMV_PUT_AD
N_PUT_AD_MAX = N_MMV_PUT_AD
N_PUT_AD_MIN = N_MMV_PUT_AD
MAF_SP_PUT_AD_MAX = MAF_SP_MMV_PUT_AD
MAF_SP_PUT_AD_MIN = MAF_SP_MMV_PUT_AD

(22) endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PUT_AD	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Correlation constant for PUT controller adaptation activation					
C_T_DLY_PUT_AD_INI	1	0...FFH	0...2.55	0.01	[s]
Delay time initialization for PUT controller adaptation					
C_TIA_MIN_PUT_AD	1	0...FEH	-48...142.5	0.75	[°C]
MIN - limit for TIA for PUT controller adaptation					
C_TIA_MAX_PUT_AD	1	0...FEH	-48...142.5	0.75	[°C]
MAX - limit for TIA for PUT controller adaptation					
C_TCO_MIN_PUT_AD	1	0...FEH	-48...142.5	0.75	[°C]
MIN - limit for TCO for PUT controller adaptation					
C_TCO_MAX_PUT_AD	1	0...FEH	-48...142.5	0.75	[°C]
MAX - limit for TCO for PUT controller adaptation					
C_MAF_SP_WIN_PUT_AD	1	0...FFFFH	0...1389	0.0211948	[mg/stk]
threshold for MAF_SP limited condition for PUT controller adaptation					
C_MAF_WIN_PUT_AD	1	0...FFFFH	0...1389	0.0211948	[mg/stk]
threshold for MAF limited condition for PUT controller adaptation					
C_N_WIN_PUT_AD	1	0...1FE0H	0...8160	1	[rpm]
threshold for N limited condition for PUT controller adaptation					
LC_AD_CLR_PUT_CTL	1	0...1H	0...1	1	[-]
Switch to clear PUT control adaptation values after ECU reset					
C_PUT_DIF_MAX_PUT_AD	1	0...FFFFH	0...5434	0.0829175	[hPa]
Maximum PUT_DIF for PUT controller adaptation					
LC_PUT_CTL_AD_UPD_MOD	1	0...1H	0...1	1	[-]
PUT control adaptation update mode (0 - update only 1 element, 1- update column)					
LC_PUT_CTL_AD_DIF_LIM_ACT	1	0...1H	0...1	1	[-]
Activation of gradient limitation of adjacent entries of PUT controller adaptation map					
IP_PRS_DIF_WG_ACR_SP_AD	9*9	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
LDPM_PRS_1_CHRG	9	0...FFFFH	0...5434	0.0829175	[hPa]
LDPM_FLOW_1_CHRG	9	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Dummy IP for specification of properties of adaptation map PRS_DIF_WG_ACR_SP_AD					
ID_PRS_PUT_AD_MIN	9	0...FFFFH	0...5434	0.0829175	[hPa]
LDPM_PRS_1_CHRG	9	0...FFFFH	0...5434	0.0829175	[hPa]
Minimum pressure for PUT controller adaptation					
ID_PRS_PUT_AD_MAX	9	0...FFFFH	0...5434	0.0829175	[hPa]
LDPM_PRS_1_CHRG	9	0...FFFFH	0...5434	0.0829175	[hPa]
Maximum pressure for PUT controller adaptation					
ID_FLOW_PUT_AD_MIN	9	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDPM_FLOW_1_CHRG	9	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Minimum flow for PUT controller adaptation					
ID_FLOW_PUT_AD_MAX	9	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDPM_FLOW_1_CHRG	9	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Maximum flow for PUT controller adaptation					
C_PRS_DIF_WG_ACR_SP_AD_GRD_MAX	1	8000...7FFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Maximum difference of adjacent entries of PUT controller adaptation map					
C_PRS_DIF_WG_ACR_SP_AD_INC	1	0...FFFFH	0...5434	0.0829175	[hPa]
Increment of PUT controller adaptation map entries					
C_PRS_DIF_WG_ACR_SP_AD_BOL	1	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Bottom limit of PUT control adaptation range					
C_PRS_DIF_WG_ACR_SP_AD_TOL	1	0...FFFFH	-2717.04145... 2716.95854	0.0829175	[hPa]
Top limit of PUT control adaptation range					

9.84 Waste gate reduced area setpoint

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AR_RED_WG_SP_MMV	O/V	0...FFFFH	0...58.592855	8.9407E-4	cm²
filtered value of reduced area setpoint for waste gate					
LV_AR_RED_WG_SP_MAX	O/V	0...1H	0...1	1	-
Indication that AR_RED_WG_SP has reached its maximum value					
LV_AR_RED_WG_SP_MIN	O/V	0...1H	0...1	1	-
Indication that AR_RED_WG_SP has reached its minimum value					
PSI_WG_SP	V	0...FFFFH	0...8.88033	1.35505E-4	-
waste gate flow rate setpoint					
AR_RED_WG_SP	V	0...FFFFH	0...58.592855	8.9407E-4	cm²
reduced area setpoint for waste gate					

Input data:

LC_TCHA_CONF	PQ_EX_SP	LV_PUT_CTL_ACT	SIN_WG
FLOW_WG_SP	PRS_EX_SP	FLOW_WG_CON_1	IP_PSI_WG

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AR_RED_WG_OPEN	1	0...FFFFH	0...58.592855	8.9407E-4	cm2
Reduced area of full open wastegate					
C_AR_RED_WG_SP_AMPL	1	0...FFFFH	0...58.592855	8.9407E-4	cm²
AR_RED_WG amplitude of sinusoidal excitation signal					
C_AR_RED_WG_SP_MAN	1	0...FFFFH	0...58.592855	8.9407E-4	cm²
manual reduced area setpoint of WG					
C_CRLC_AR_RED_WG_SP	1	0...FFFFH	0...0.999985	1.52588E-5	-
correlation constant for filtering of AR_RED_WG_SP					
LC_AR_RED_WG_SP_MAN	1	0...1H	0...1	1	-
switch for manual reduced area setpoint of WG					

9.84.1 CHRG_ISPCLARRWG0

This module calculates the setpoint of the reduced area at the waste gate.

Application Condition

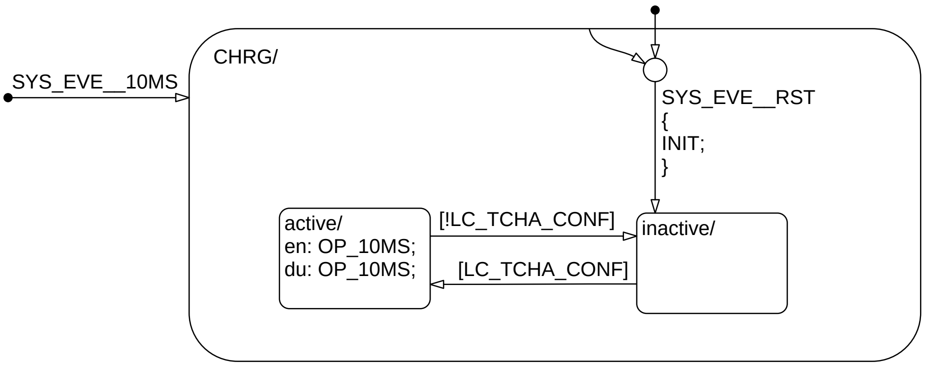


Figure 247 CHRG_ISPCLARRWG0/ APP_CDN/ Chart

Function Description

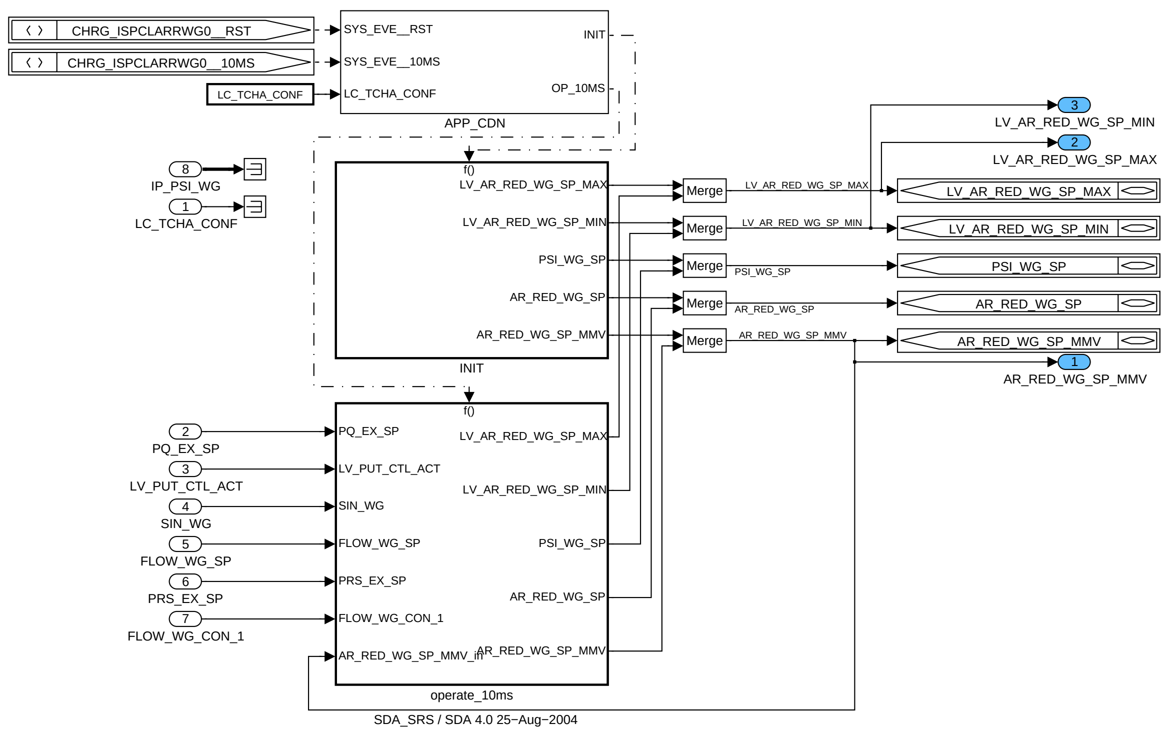


Figure 248 CHRG_ISPCLARRWG0

9.84.1.1 SUBFUNCTION: INIT

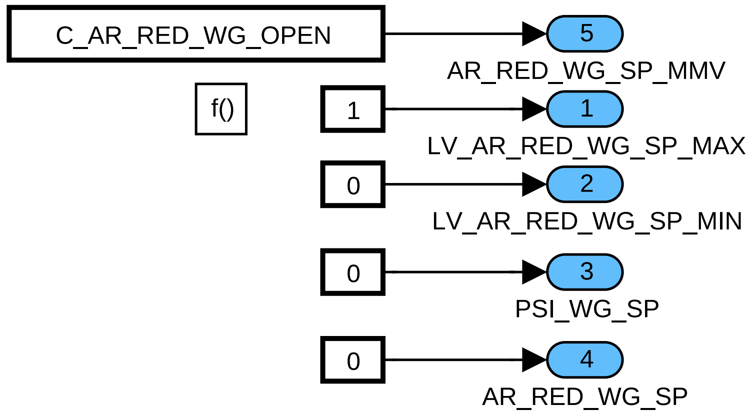


Figure 249 CHRG_ISPCLARRWG0/ INIT

9.84.1.2 SUBFUNCTION: operate_10ms

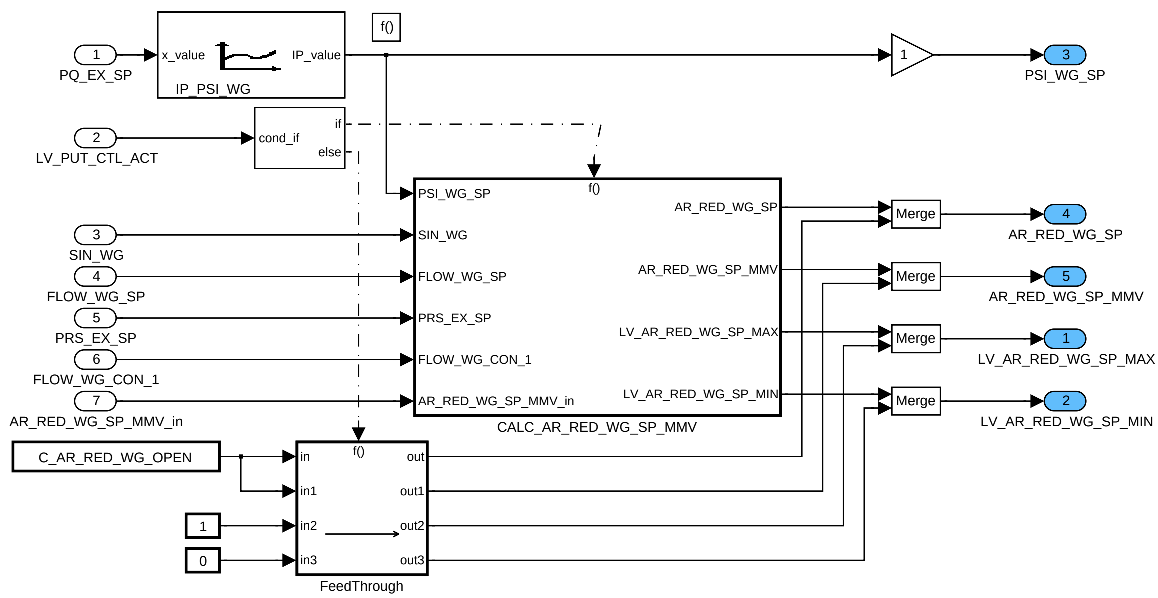


Figure 250 CHRG_ISPCLARRWG0/ operate_10ms

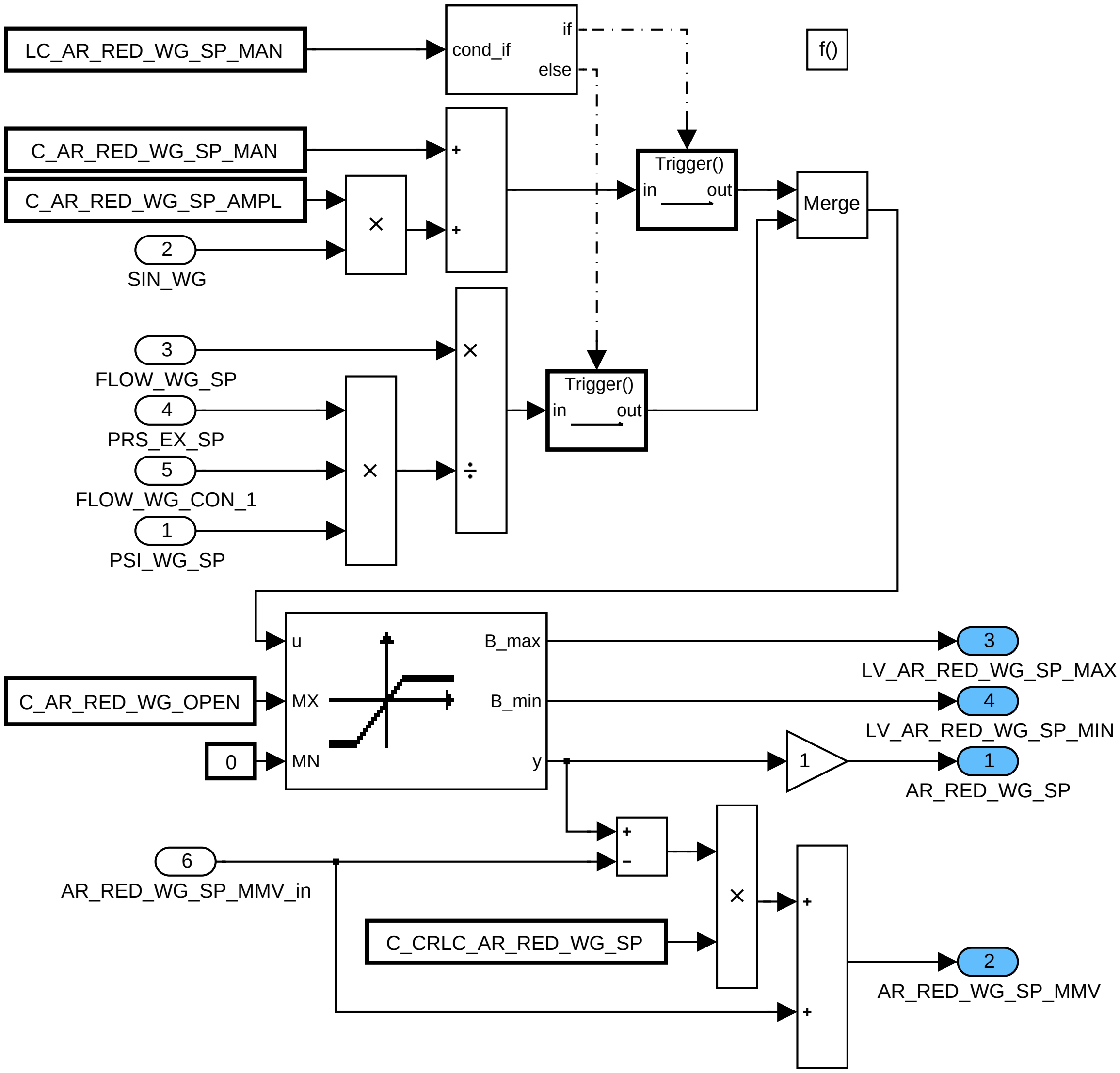


Figure 251 CHRG_ISPCLARRWG0/ operate_10ms/ CALC_AR_RED_WG_SP_MMV

9.85 Wastegate flow setpoint

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FLOW_ENG_SP	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Flow engine setpoint					
FLOW_ENG_TRA	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Transient flow engine for wastegate flow setpoint calculation					
FLOW_WG_SP	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
waste gate flow setpoint					
LV_FLOW_ENG_STAT	V	0...1H	0...1	1	-
Flag for engine flow has reached its setpoint					
FLOW_ENG_MMV	V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Filtered flow engine for wastegate flow setpoint calculation					
LV_FLOW_ENG_MMV_FALL	-	0...1H	0...1	1	-
Flag for filtered engine flow is decreasing					

Input data:

LC_TCHA_CONF	MAF_KGH_SP_TCHA	MFF_SP_MV	N
NC_MAF_FAC_CYL	FLOW_ENG_TCHA	LV_PUT_CTL_STAT	FLOW_TUR_SP
SP_FLOW_EGRV	MAP	AMP	NC_NR_TCHA

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_FLOW_ENG_FLOW_WG_SP	1	0...FFH	0...0.99609	0.00390624	-
Filter coefficient for engine flow filtering for wastegate flow setpoint calculation					
C_FLOW_WG_SP_MAN	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
manual setpoint of WG flow					
LC_FLOW_WG_SP_MAN	1	0...1H	0...1	1	-
switch for manual setpoint of WG flow					
IP_CRLC_FLOW_ENG_TRA	4	0...FFH	0...0.99609	0.00390624	-
LDP_FLOW_ENG_IP_CRLC_FLOW_ENG	4	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Filter coefficient for transient engine flow for wastegate flow setpoint calculation					

9.85.1 CHRG_ISPCLFLOWG0

General information:

In the exhaust manifold the exhaust gas flow through the outlet valves of the engine is divided into turbine flow, wastegate flow and the flow for external exhaust gas recirculation. The wastegate flow setpoint FLOW_WG_SP depends on the other three flows. In stationary operation (i.e. FLOW_ENG_TCHA has reached its setpoint FLOW_ENG_SP after a FLOW_ENG_SP jump) FLOW_WG_SP must be calculated based on FLOW_WG_SP in order not to have a positive feedback which leads to growing engine flow and charged pressure oscillations. Just after a FLOW_ENG_SP jump FLOW_WG_SP must be calculated based on FLOW_ENG_TCHA (< FLOW_ENG_SP at this moment) because otherwise the wastegate opens too early. The transition from FLOW_ENG_TCHA to FLOW_ENG_SP as base for FLOW_WG_SP calculation is done when either FLOW_ENG_MMV reaches its setpoint FLOW_WG_SP or when engine flow swings back before reaching its setpoint.

For a biturbo system all FLOW_xx represent the flows for one turbo charger.

Function Description

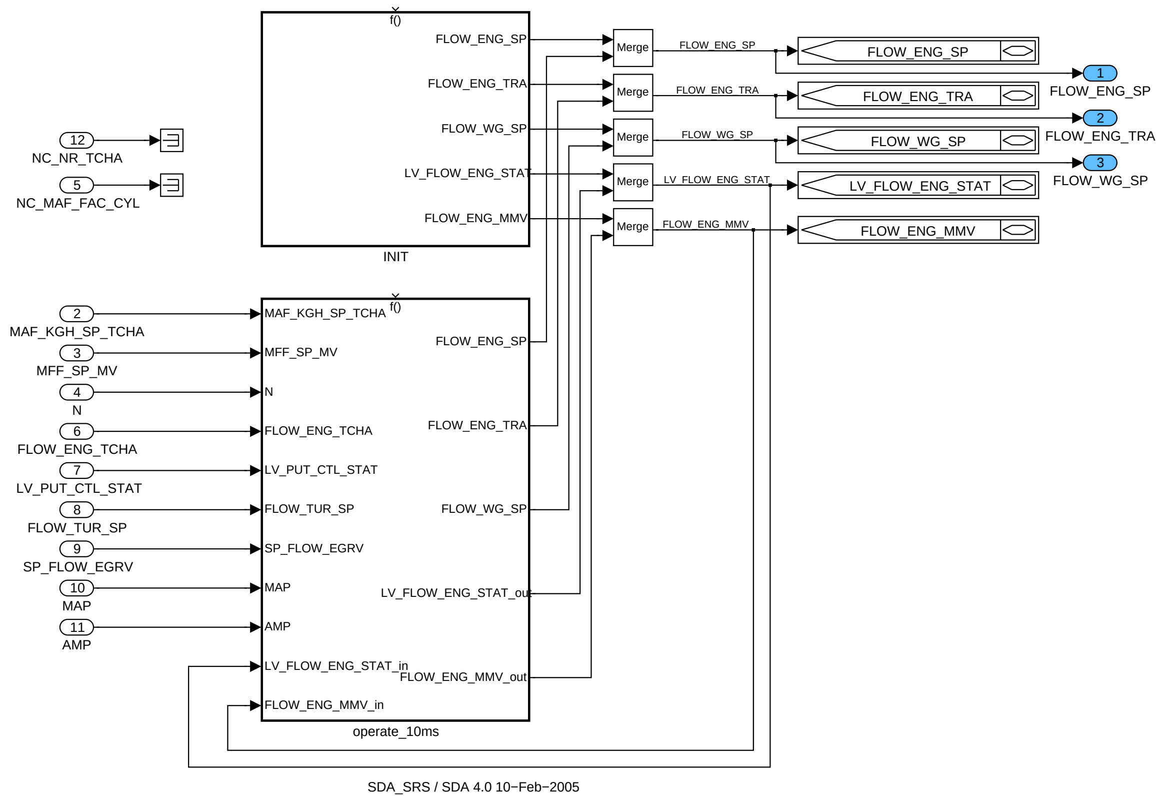


Figure 252 CHRG_ISPCLFLOWG0

9.85.1.1 SUBFUNCTION: INIT

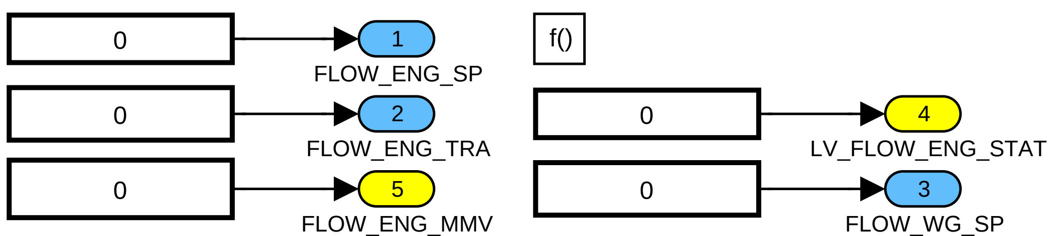


Figure 253 CHRG_ISPCLFLOWG0/ INIT

9.85.1.2 SUBFUNCTION: operate_10ms

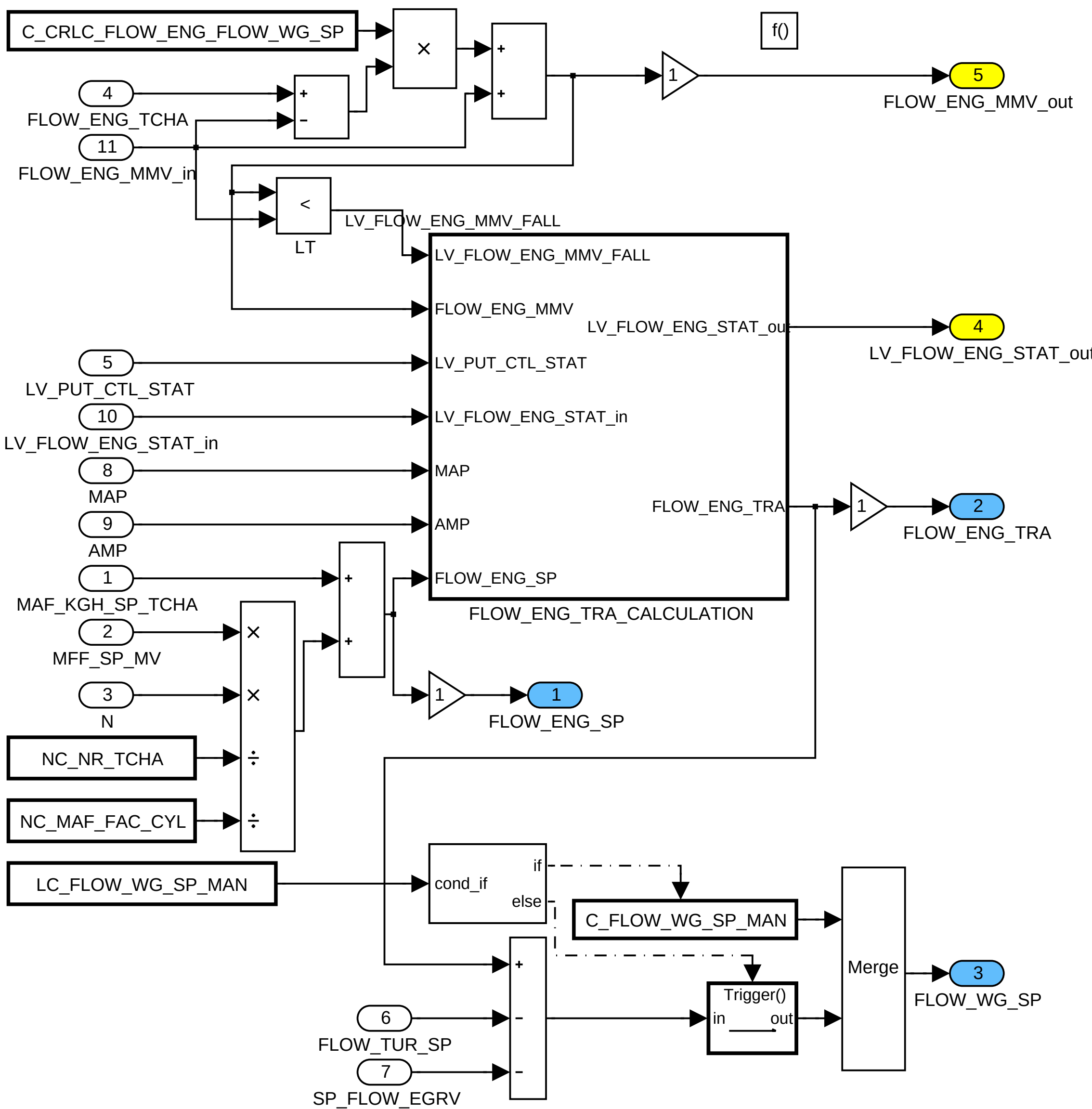


Figure 254 CHRG_ISPCLFLOWG0/ operate_10ms

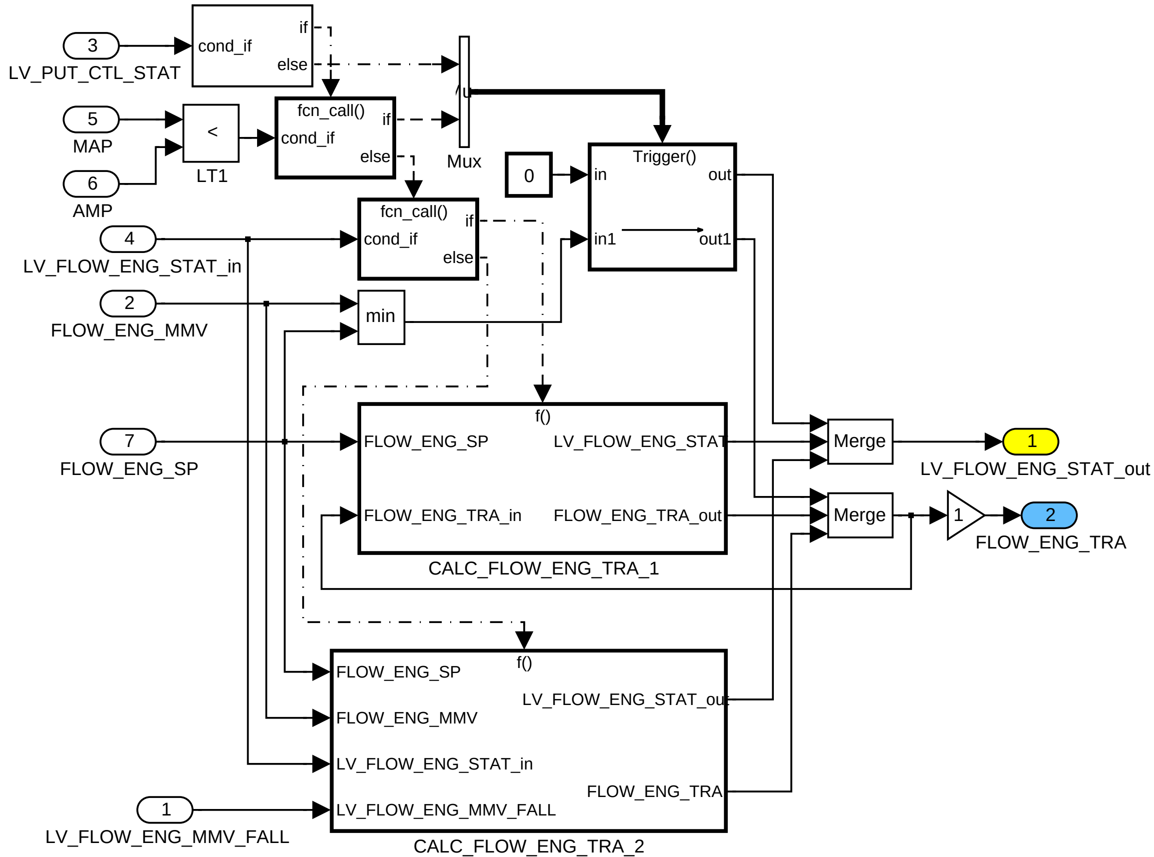


Figure 255 CHRG_ISPCLFLOWG0/ operate_10ms/ FLOW_ENG_TRA_CALCULATION

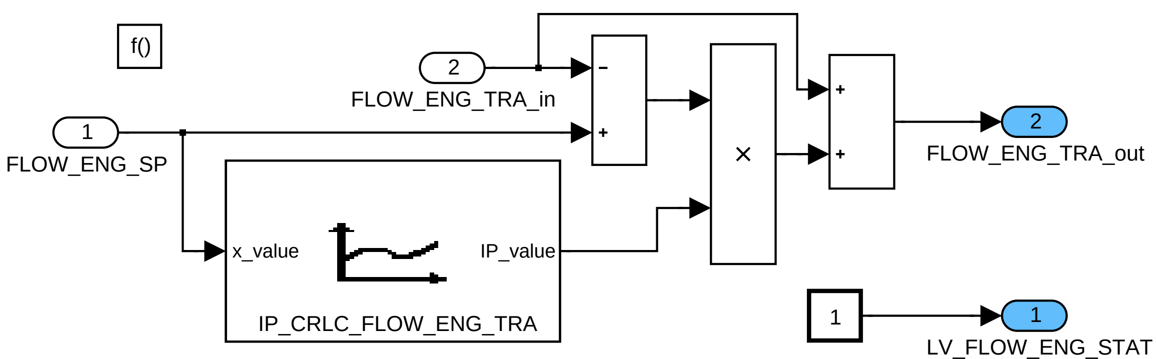


Figure 256 CHRG_ISPCLFLOWG0/ operate_10ms/ FLOW_ENG_TRA_CALCULATION/ CALC_FLOW_ENG_TRA_1

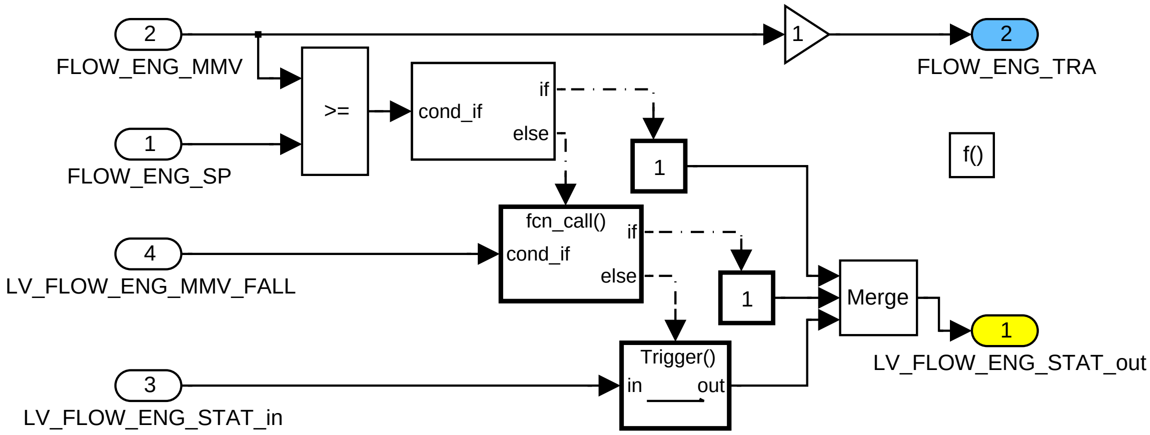


Figure 257 CHRG_ISPCLFLOWG0/ operate_10ms/ FLOW_ENG_TRA_CALCULATION/ CALC_FLOW_ENG_TRA_2

9.86 Wastegate position control (Electropneumatic control version)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FAC_PWM_VB_EPC_WG	O/V	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
Correction factor of PWM_WG due to battery voltage					
PQ_EPC_IN_SP	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio setpoint at EPC input					
PQ_EPC_OUT_SP	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure ratio setpoint at EPC output					
PRS_DIF_WG_ACR_SP	V	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Wastegate actuator difference pressure setpoint					
PRS_EX_PRS_WG_ACR_SP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Corrected exhaust gas pressure for wastegate actuator pressure setpoint calculation					
PRS_WG_ACR_SP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure setpoint at controlled side of WG actuator					
PWM_WG	O/V	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
PWM at EPC for WG actuation					
PWM_WG_BAS	V	8000... 7FFFH	-100... 99.9969482422	3.05176e-3	[%]
Basic precontrol PWM at wastegate EPC					
PWM_WG_COR_OFS	O/V	8000... 7FFFH	-100... 99.9969482422	3.05176e-3	[%]
Air temperature correction offset of wastegate PWM					
PWM_WG_COR_SLOP	O/V	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
Air temperature correction slope of wastegate PWM					
PWM_WG_OPL	V	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Open looop control part of PWM for wastegate EPC					
TEMP_EPC_WG	V	0... FE00H	-48... 142.5	2.92969e-3	[°C]
Wastegate EPC temperature					

Input Data:

LC_TCHA_CONF	FLOW_ENG_SP	FLOW_WG_SP	FLOW_WG_CON_1
LV_PUT_CTL_ACT	FLOW_TUR_SP	PRS_DIF_WG_ACR_SP_AD_ADD	AMP
PRS_WG_ACR_NOT_CTL	C_TYP_EPC_WG_ACR	PRS_WG_ACR_EPC_H_SP	PRS_WG_ACR_EPC_L_SP
VB	LV_PWM_WG_EXT_ADJ	PWM_WG_EXT_ADJ	SIN_WG
PQ_EX_SP	LV_ST_END	TAM	LC_PQ_EPC_OUT
PWM_WG_CLL	TCO	VS	PWM_WG_MAN

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_CRLC_TEMP_EPC_WG	1	0... FFFFH	0... 0.99998474121	15.2588e-6	[-]
Filter constant for wastegate EPC temperature					
C_FLOW_TUR_THD_WG_CLOSE	1	0... FFFFH	0... 2047.96875	0.03125	[kg/h]

Turbine flow request threshold to request closed wastegate					
C_FLOW_WG_CON_STND	1	0... FFFFH	0... 0.0206019	314.365e-9	[s/m]
Temperature coefficient for wastegate flow at standard temperature					
C_PRS_DIF_WG_ACR_SP_WG_CLOSE	1	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Wastegate actuator pressure difference setpoint for closed wastegate					
C_PRS_DIF_WG_ACR_SP_WG_OPEN	1	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Wastegate actuator pressure difference setpoint for open wastegate					
C_PRS_EX_PRS_WG_ACR_SP_STND	1	0... FFFFH	0... 5434	0.0829175	[hPa]
Standard exhaust pressure for wastegate actuator pressure setpoint calculation					
C_PWM_WG_ACR_EPC_AMPL	1	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
Amplitude of sinusoidal wastegate PWM signal provided by the application system					
C_PWM_WG_FRQ	1	A... 400H	10... 1024	1	[Hz]
frequency for PWM_WG, initialization only at reset (typical value 30 Hz or 300 Hz)					
IP_FAC_AMP_PRS_WG_ACR_SP	6*6	0... FFFFH	0... 3.99993896484	61.0352e-6	[-]
LDP_AMP_IP_FAC_AMP_PRS_WG_ACR	6	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_FLOW_ENG_IP_FAC_AMP_PRS_WG	6	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
AMP correction factor for wastegate actuator pressure setpoint					
IP_FAC_PWM_EPC_VB	8	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_VB_IP_FAC_PWM_EPC	8	0... FFH	0... 26	0.1019608	[V]
Correction of PWM_EPC due to battery voltage					
IP_FAC_VS_TEMP_EPC_WG	6	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDP_VS_IP_FAC_VS_TEMP_EPC_WG	6	0... FFH	0... 255	1	[km/h]
Vehicle speed correction factor for wastegate EPC temperature					
IP_FOC_WG_ACR_SPR	4	0... FFE0H	-2048... 2047.9375	0.0625296	[N]
LDPM_PSN_WG_ACR_1_CHRG	4	0... FFFFH	0... 65.535	1e-3	[mm]
Force at WG actuator caused by spring and friction while opening the WG					
IP_PRS_DIF_WG_ACR_SP	12*12	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
LDP_FLOW_WG_SP_IP_PRS_DIF_WG	12	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
LDP_PRS_EX_IP_PRS_DIF_WG_ACR_SP	12	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure difference at wg actuator setpoint depending on flow engine setpoint and wastegate and wastegate position					
IP_PWM_WG_COR_OFS	4	0... FFFFH	-100... 99.9969482422	3.05176e-3	[%]
LDPM_TEMP_EPC_WG_1_CHRG	4	0... FEH	-48... 142.5	0.75	[°C]
Offset for air temperature correction of wastegate PWM					
IP_PWM_WG_COR_SLOP	4	0... FFFFH	0... 1.99996948242	30.5176e-6	[-]
LDPM_TEMP_EPC_WG_1_CHRG	4	0... FEH	-48... 142.5	0.75	[°C]
Slope for air temperature correction of wastegate PWM					
IP_PWM_WG_OPL	8*12	0... FFFFH	-100... 99.9969482422	3.05176e-3	[%]
LDP_PQ_EPC_OUT_SP_IP_PWM_WG	12	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDP_PQ_EPC_IN_SP_IP_PWM_WG	8	0... FFFFH	0... 15.999756	244.141e-6	[-]
Open loop wastegate control signal					
IP_PWM_WG_OPL2	6	0... FFFFH	-100... 99.9969482422	3.05176e-3	[%]
LDP_PRS_DIF_WG_ACR_SP_IP_PWM_WG	6	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Open loop wastegate control signal					
IP_TEMP_EPC_WG_ADD	6	0... FEH	-48... 142.5	0.75	[°C]

LDP_PWM_WG_IP_TEMP_EPC_WG_ADD	6	0... FFFFH	0... 99.9984741211	1.52588e-3	[%]
PWM_WG correction of wastegate EPC temperature					
IP_TEMP_EPC_WG_BAS	6*6	0... FEH	-48... 142.5	0.75	[°C]
LDP_TCO_IP_TEMP_EPC_WG_BAS	6	0... FEH	-48... 142.5	0.75	[°C]
LDP_TAM_IP_TEMP_EPC_WG_BAS	6	0... FEH	-48... 142.5	0.75	[°C]
Basic wastegate EPC temperature					
LC_PWM_WG_ACR_EPC_MAN_ACT	1	0... 1H	0... 1	1	[-]
Switch to determinate whether wastegate PWM signal is provided by the application system					

General Information

Function description

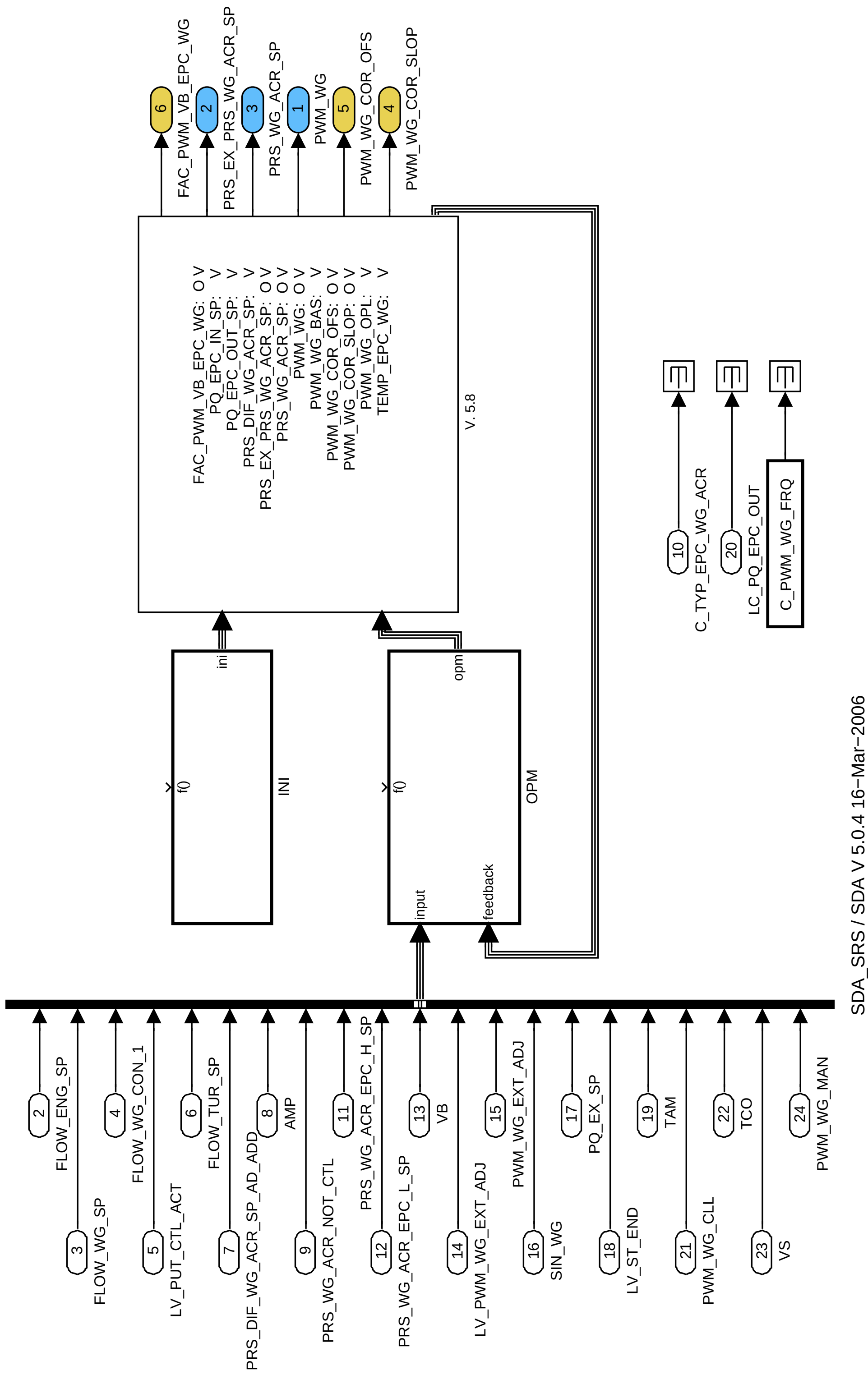


Figure 258:

Path: CHRG_ACCTLWG0

9.86.1 CHRG_ACCTLWG0/INI

CONTENT

9.86.1.1 CHRG_ACCTLWG0/INI/CLC_INI

CONTENT

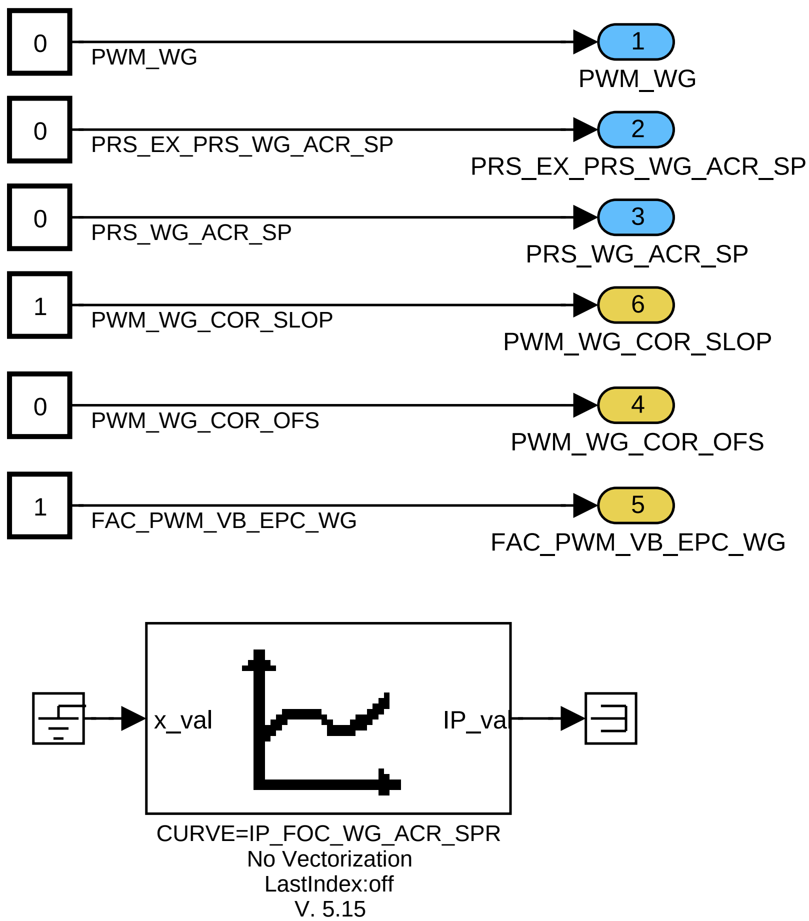


Figure 259:
Path: CHRG_ACCTLWG0/INI/CLC_INI

9.86.2 CHRG_ACCTLWG0/OPM

CONTENT

9.86.2.1 CHRG_ACCTLWG0/OPM/CLC_OPM

CONTENT

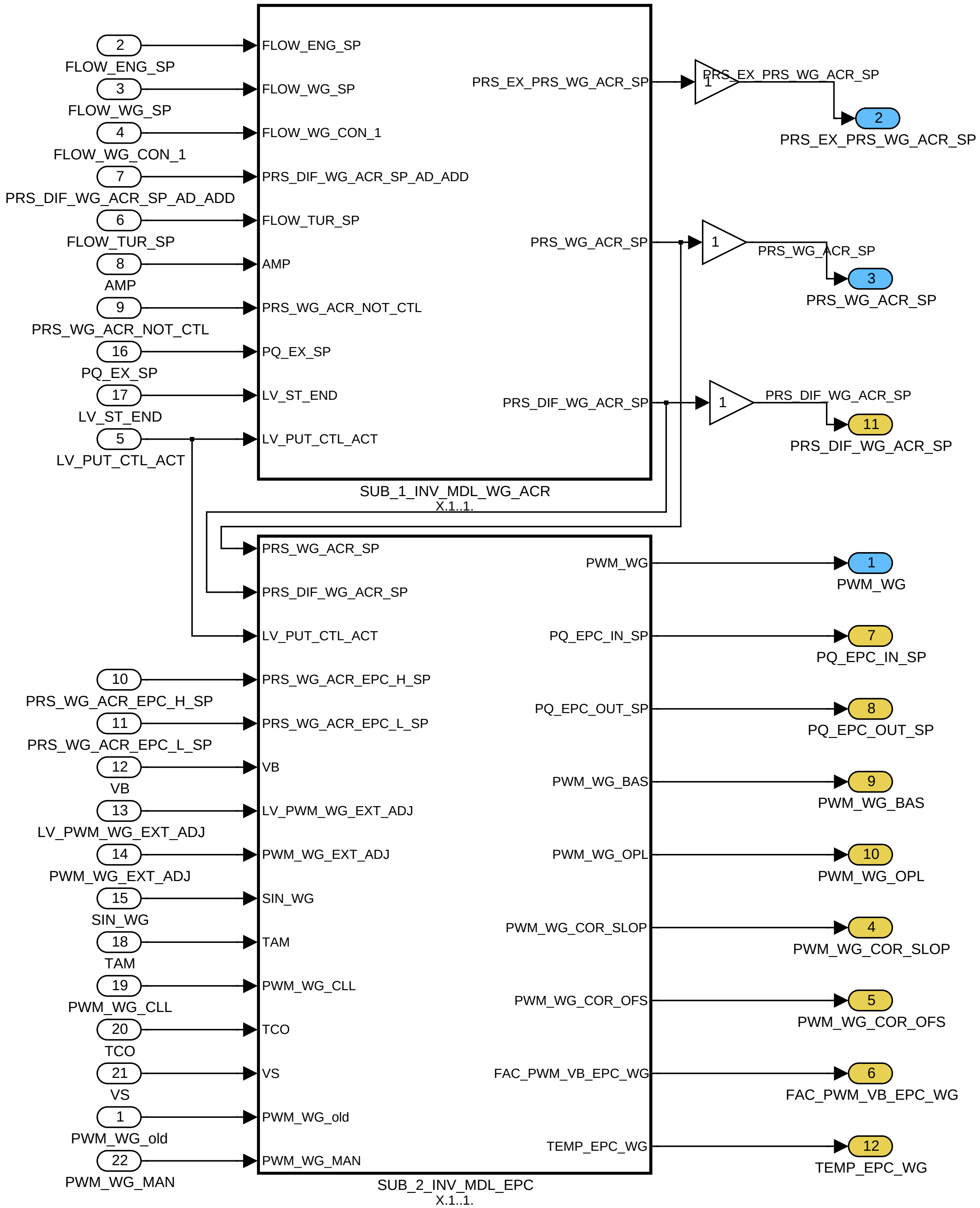


Figure 260:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM

9.86.2.1.1 CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_1_INV_MDL_WG_ACR

CONTENT

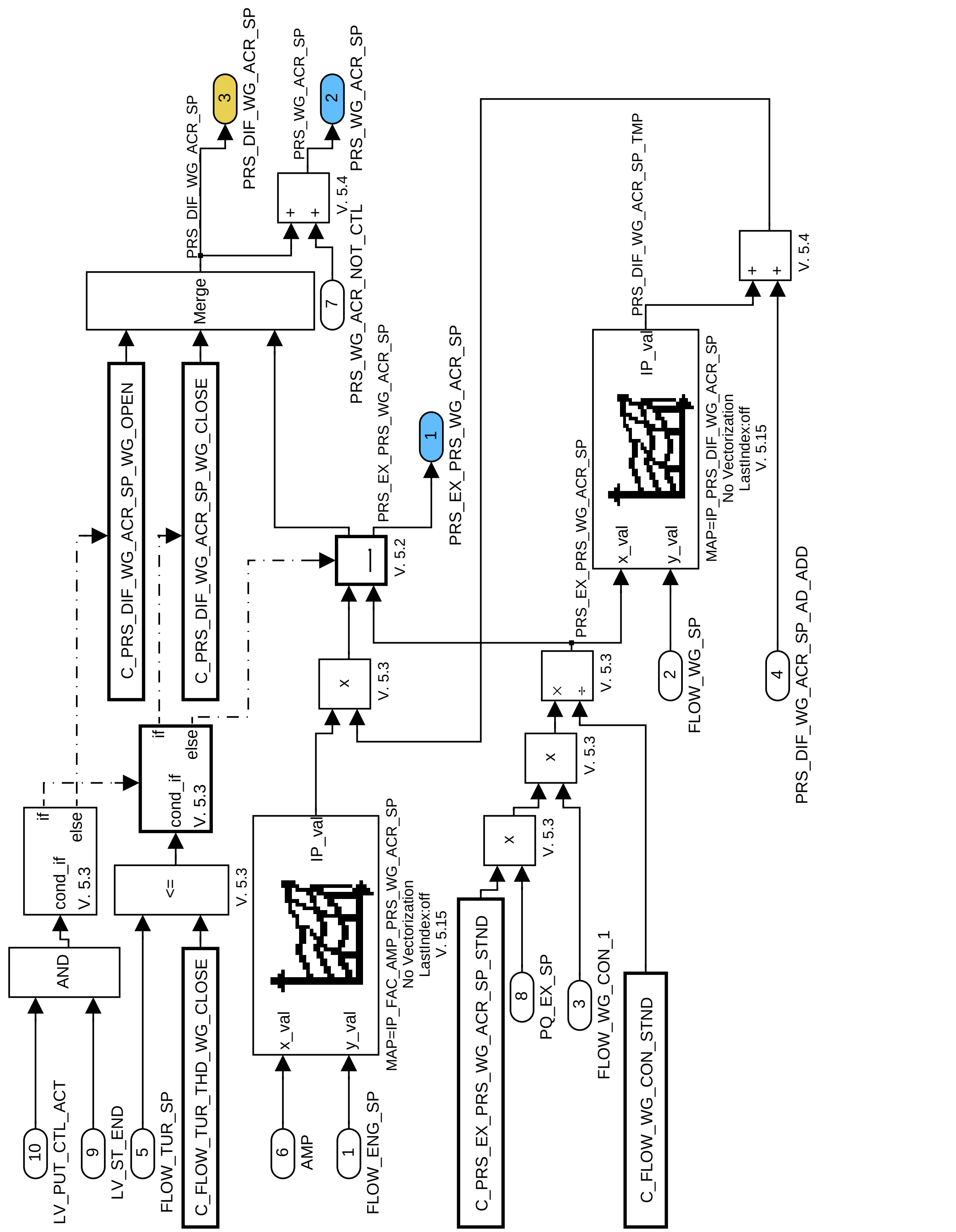


Figure 261:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_1_INV_MDL_WG_ACR

9.86.2.1.2 CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC

CONTENT

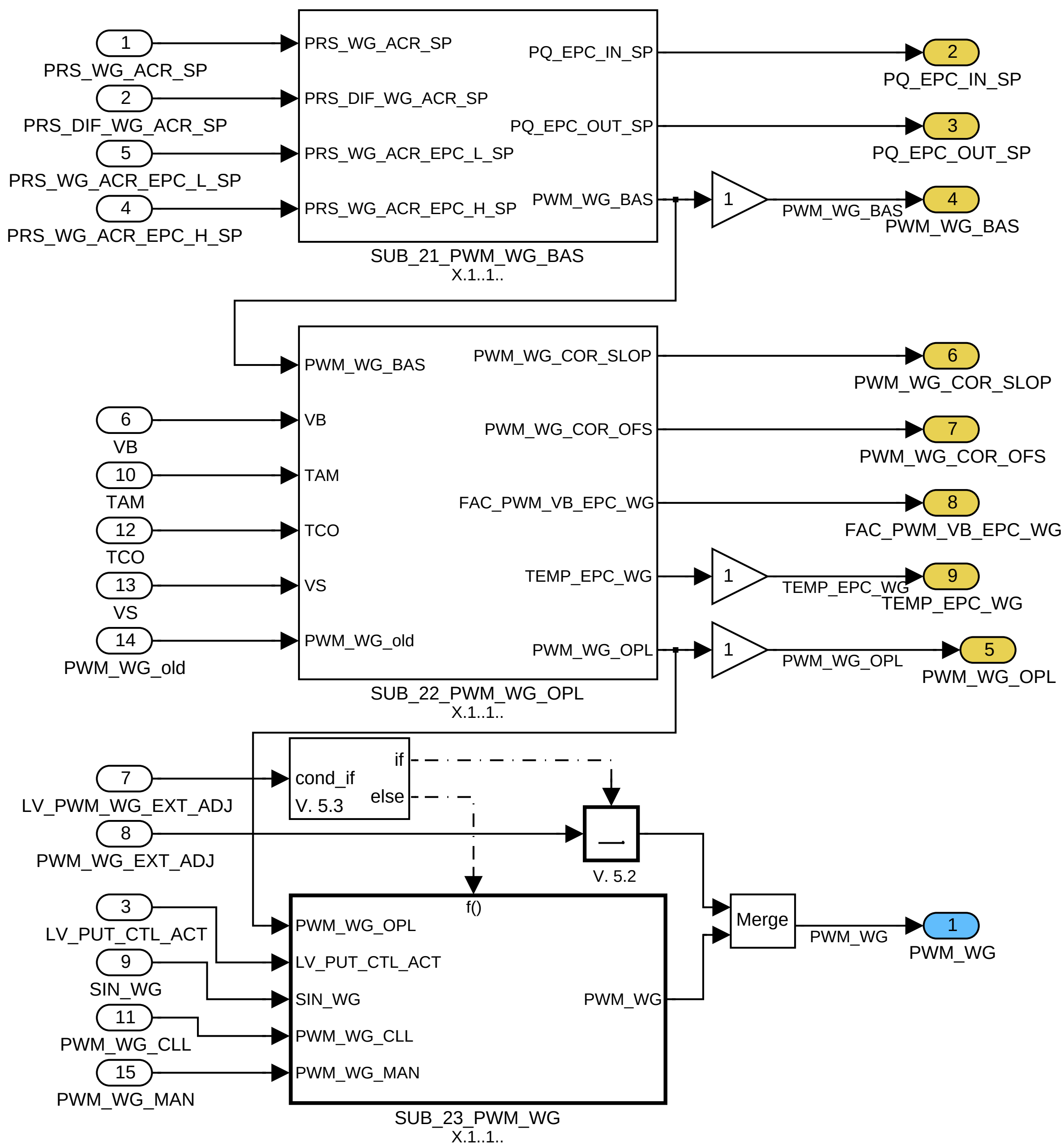


Figure 262:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC

9.86.2.1.2.1 CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_21_PWM_WG_BAS

CONTENT

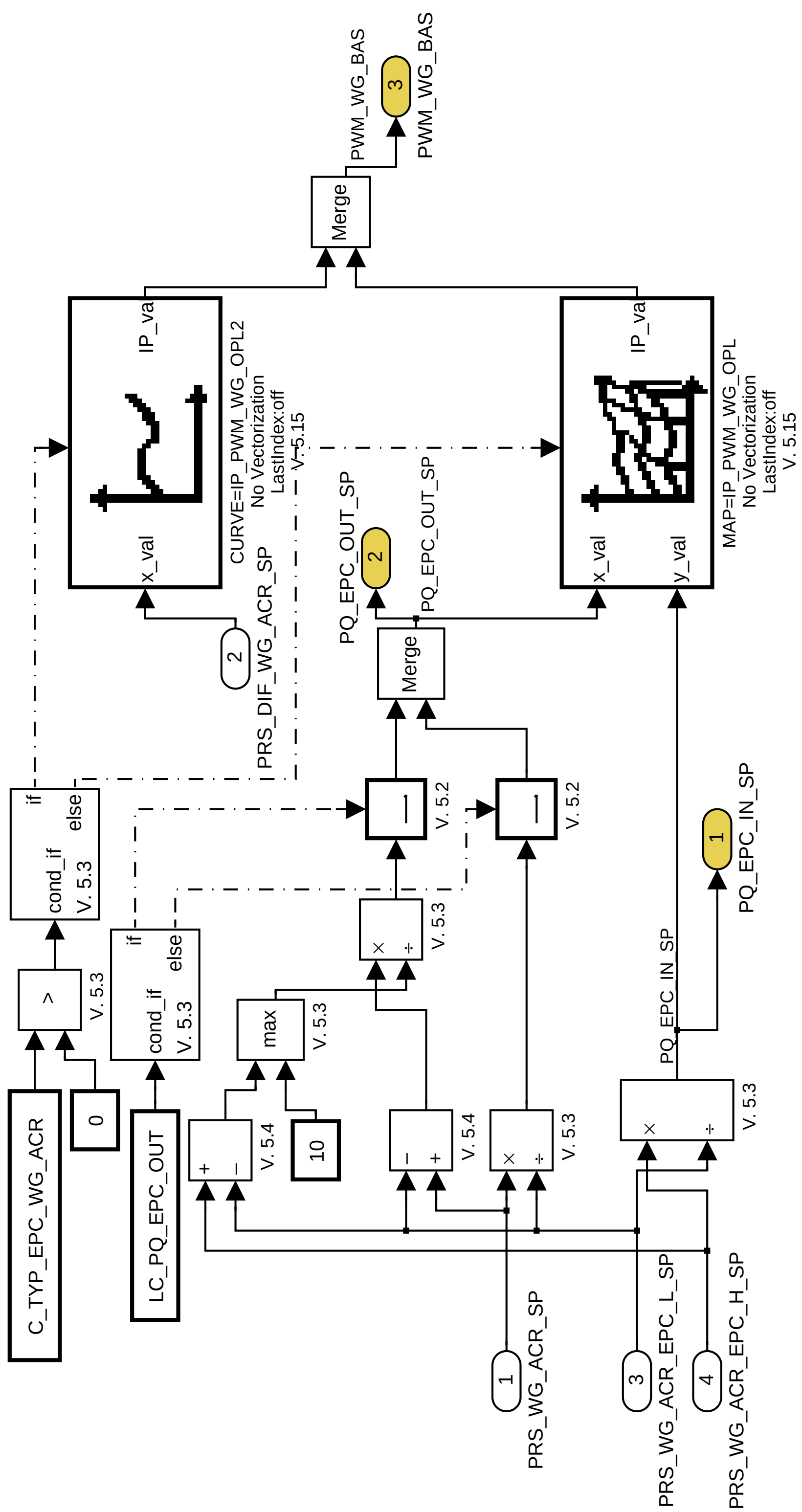


Figure 263:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_21_PWM_WG_BAS

9.86.2.1.2.2 CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_22_PWM_WG_OPL

CONTENT

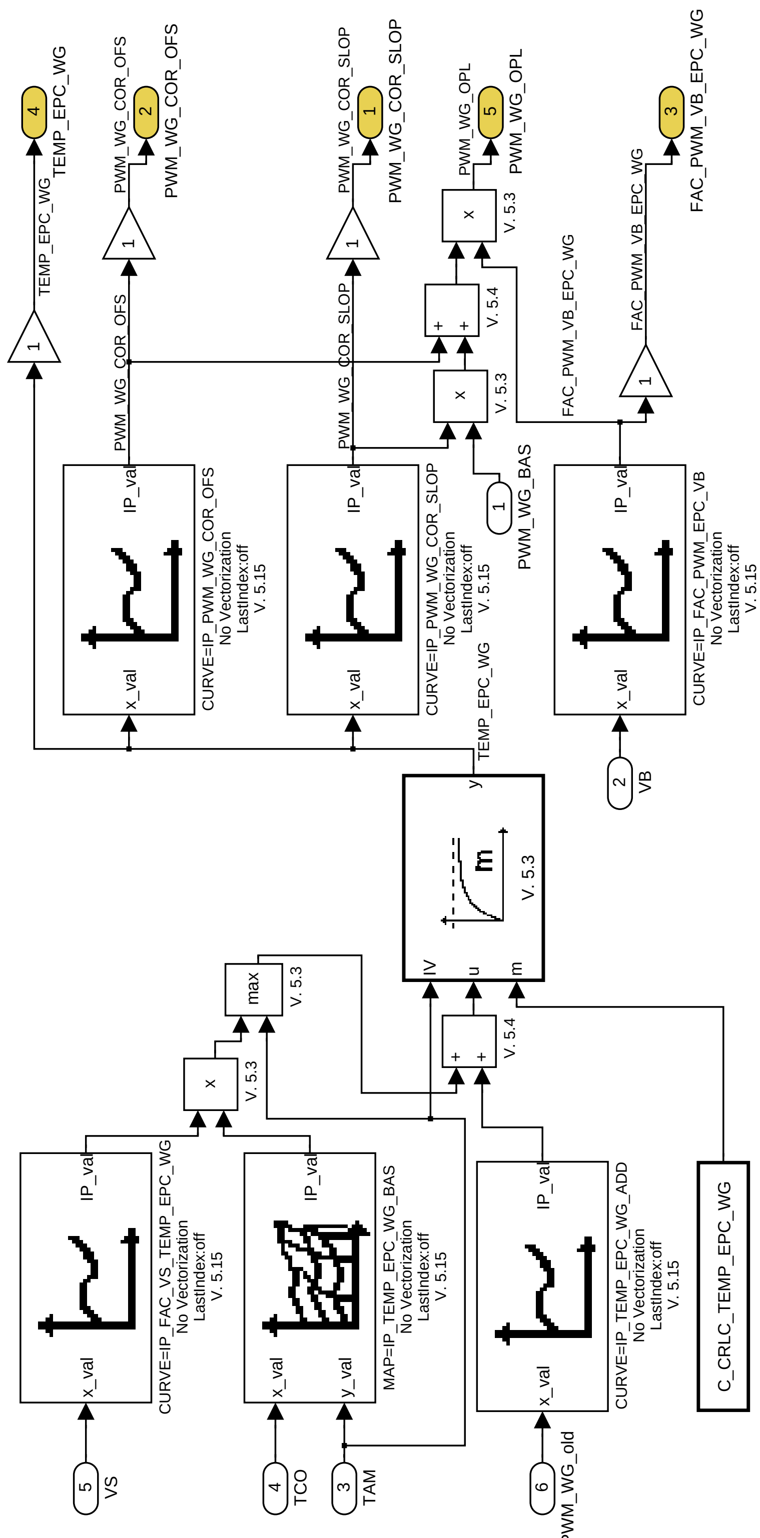


Figure 264:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_22_PWM_WG_OPL

9.86.2.1.2.3 CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_23_PWM_WG

CONTENT

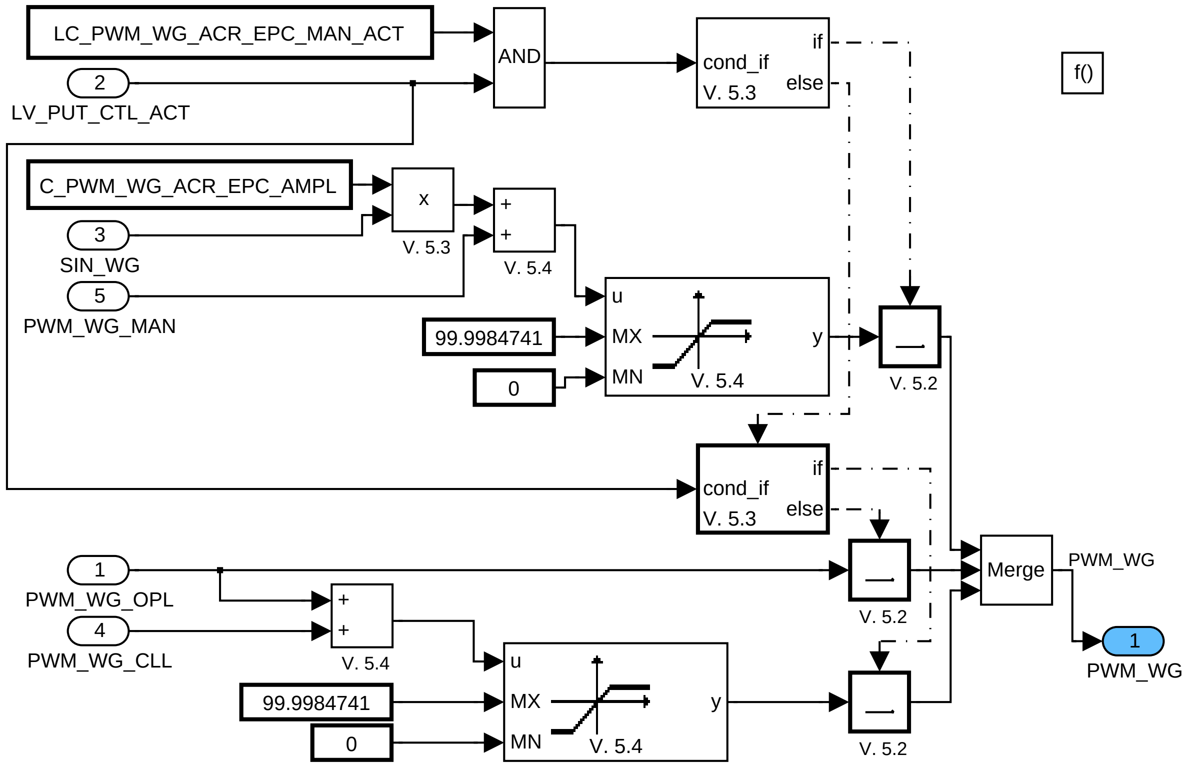


Figure 265:
Path: CHRG_ACCTLWG0/OPM/CLC_OPM/SUB_2_INV_MDL_EPC/SUB_23_PWM_WG

9.87 Wastegate position control application incidences (Electropneumatic control version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PRS_WG_ACR_EPC_H_SP	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure setpoint at high pressure side of wastegate actuator EPC					
PRS_WG_ACR_EPC_L_SP	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure setpoint at low pressure side of wastegate actuator EPC					
PRS_WG_ACR_EPC_H	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure at high pressure side of wastegate actuator EPC					
PRS_WG_ACR_EPC_L	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure at low pressure side of wastegate actuator EPC					
PRS_WG_ACR_NOT_CTL	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Pressure in not controlled chamber of the wastegate actuator					
SIN_WG	O/V	8000...7FFFH	-1...0.99996948	3.05176E-5	-
Sinusoidal wastegate excitation					
LV_SIN_WG_INH	V	0...1H	0...1	1	-
Inhibition flag for sinusoidal wastegate excitation					
T_SIN_WG	V	0...FFFFH	0...655.35	0.01	s
Auxiliary time for sinusoidal excitation					
LV_SIN_WG_ACT	V	0...1H	0...1	1	-
Sinusoidal wastegate excitation active					

Input data:

LC_TCHA_CONF	AMP	PRS_VAC_PUMP	PRS_CHA_UP[NC_NR_TCHA]
PRS_CHA_DOWN_SP	PUT_SP	PRS_CHRG_DOWN	MAP_SP
MAP	PRS_CHA_DOWN[NC_NR_TCHA]	LV_IN_PROT	LV_TCHA_PROT_ACT
NC_NR_TCHA			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_PRS_WG_ACR_EPC_H_SP	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Low pass filter constant for PRS_WG_ACR_EPC_H_SP calculation					
C_PER_SIN_WG	1	0...FFFFH	0...655.35	0.01	s
Period of sinusoidal excitation signal					
C_PRS_WG_ACR_EPC_H_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value of pressure at high pressure side of wastegate actuator EPC					
C_PRS_WG_ACR_EPC_H_SP_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value of pressure setpoint at high pressure side of wastegate actuator EPC					
C_PRS_WG_ACR_EPC_L_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value of pressure at low pressure side of wastgate actuator EPC					
C_PRS_WG_ACR_EPC_L_SP_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value of pressure setpoint at low pressure side of wastegate actuator EPC					
C_PRS_WG_ACR_NOT_CTL_MAN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Manual value of not controlled pressure at wastegate actuator					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_SRC_PRS_WG_ACR_EPC_H	1	1H 1H 3H 4H 5H 6H 7H 8H 9H AH BH CH	USE_AMP USE_PRS_VAC_PUMP USE_PRS_CHA_UP_1 USE_PRS_CHA_UP_2 USE_PRS_CHA_DOWN_SP USE_PRS_CHA_DOWN_1 USE_PRS_CHA_DOWN_2 USE_PUT_SP USE_PUT USE_MAP_SP USE_MAP USE_MAN_VALUE	1	-
Source of pressure at high pressure side of EPC					
C_SRC_PRS_WG_ACR_EPC_H_SP	1	1H 1H 3H 4H 5H 6H 7H 8H 9H AH BH CH	USE_AMP USE_PRS_VAC_PUMP USE_PRS_CHA_UP_1 USE_PRS_CHA_UP_2 USE_PRS_CHA_DOWN_SP USE_PRS_CHA_DOWN_1 USE_PRS_CHA_DOWN_2 USE_PUT_SP USE_PUT USE_MAP_SP USE_MAP USE_MAN_VALUE	1	-
Source of pressure setpoint at high pressure side of EPC					
C_SRC_PRS_WG_ACR_EPC_L	1	1H 1H 3H 4H 5H 6H 7H 8H 9H AH BH CH	USE_AMP USE_PRS_VAC_PUMP USE_PRS_CHA_UP_1 USE_PRS_CHA_UP_2 USE_PRS_CHA_DOWN_SP USE_PRS_CHA_DOWN_1 USE_PRS_CHA_DOWN_2 USE_PUT_SP USE_PUT USE_MAP_SP USE_MAP USE_MAN_VALUE	1	-
Source of pressure at low pressure side of EPC					
C_SRC_PRS_WG_ACR_EPC_L_SP	1	1H 1H 3H 4H 5H 6H 7H 8H 9H AH BH CH	USE_AMP USE_PRS_VAC_PUMP USE_PRS_CHA_UP_1 USE_PRS_CHA_UP_2 USE_PRS_CHA_DOWN_SP USE_PRS_CHA_DOWN_1 USE_PRS_CHA_DOWN_2 USE_PUT_SP USE_PUT USE_MAP_SP USE_MAP USE_MAN_VALUE	1	-
Source of pressure setpoint at low pressure side of EPC					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_SRC_PRS_WG_ACR_NOT_CTL	1	1H 1H 3H 4H 5H 6H 7H 8H 9H AH BH CH	USE_AMP USE_PRS_VAC_PUMP USE_PRS_CHA_UP_1 USE_PRS_CHA_UP_2 USE_PRS_CHA_DOWN_SP USE_PRS_CHA_DOWN_1 USE_PRS_CHA_DOWN_2 USE_PUT_SP USE_PUT USE_MAP_SP USE_MAP USE_MAN_VALUE	1	-
Source of not controlled pressure at wastegate actuator					
LC_SIN_WG_REQ	1	0...1H	0...1	1	-
Request for sinusoidal wastegate excitation					
LC_SIN_WG_RST	1	0...1H	0...1	1	-
Reset of sinusoidal wastegate excitaion					
IP_SIN_WG	7	0...FFFFH	-1...0.99996948	3.05176E-5	-
LDP_ANG_IP_SIN_WG	7	0...FFFFH	0...7.99987793	1.2207E-4	-
Sine signal stored for wastegate excitation					

9.87.1 FUNCTION PART: CHRG_ACCTLWGAI0

Function Description

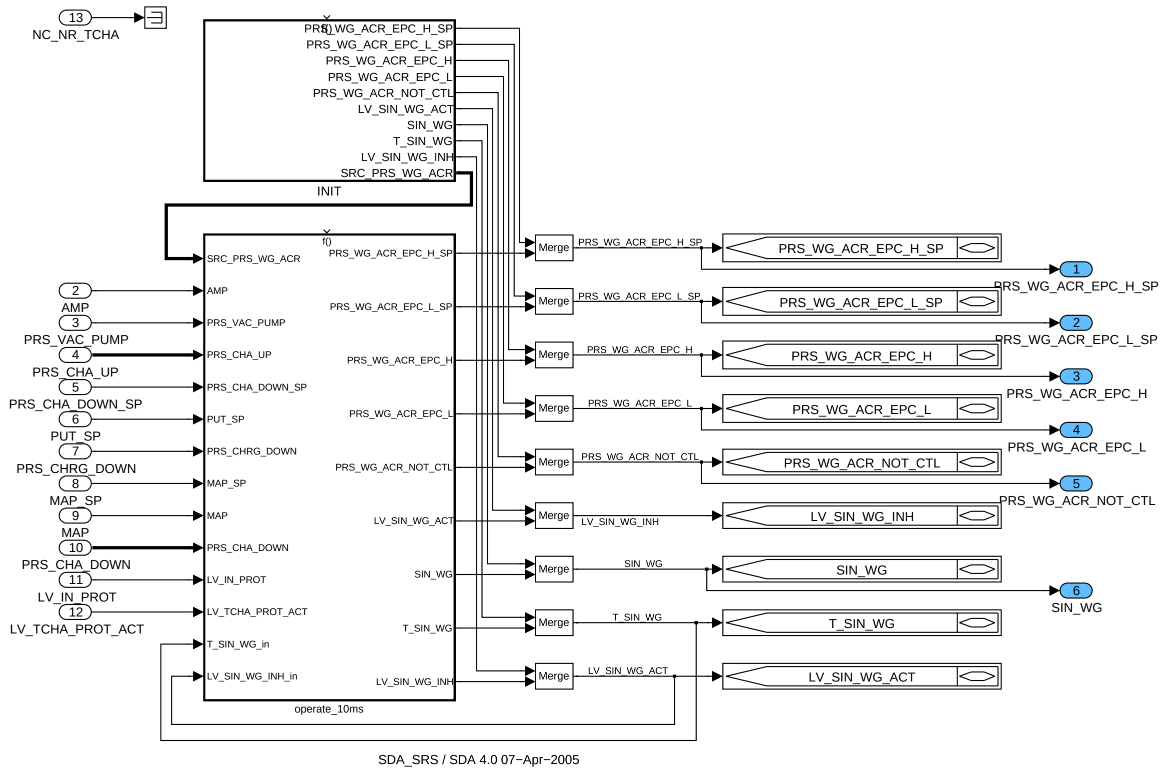


Figure 266 CHRG_ACCTLWGAI0

9.87.1.1 SUBFUNCTION: INIT

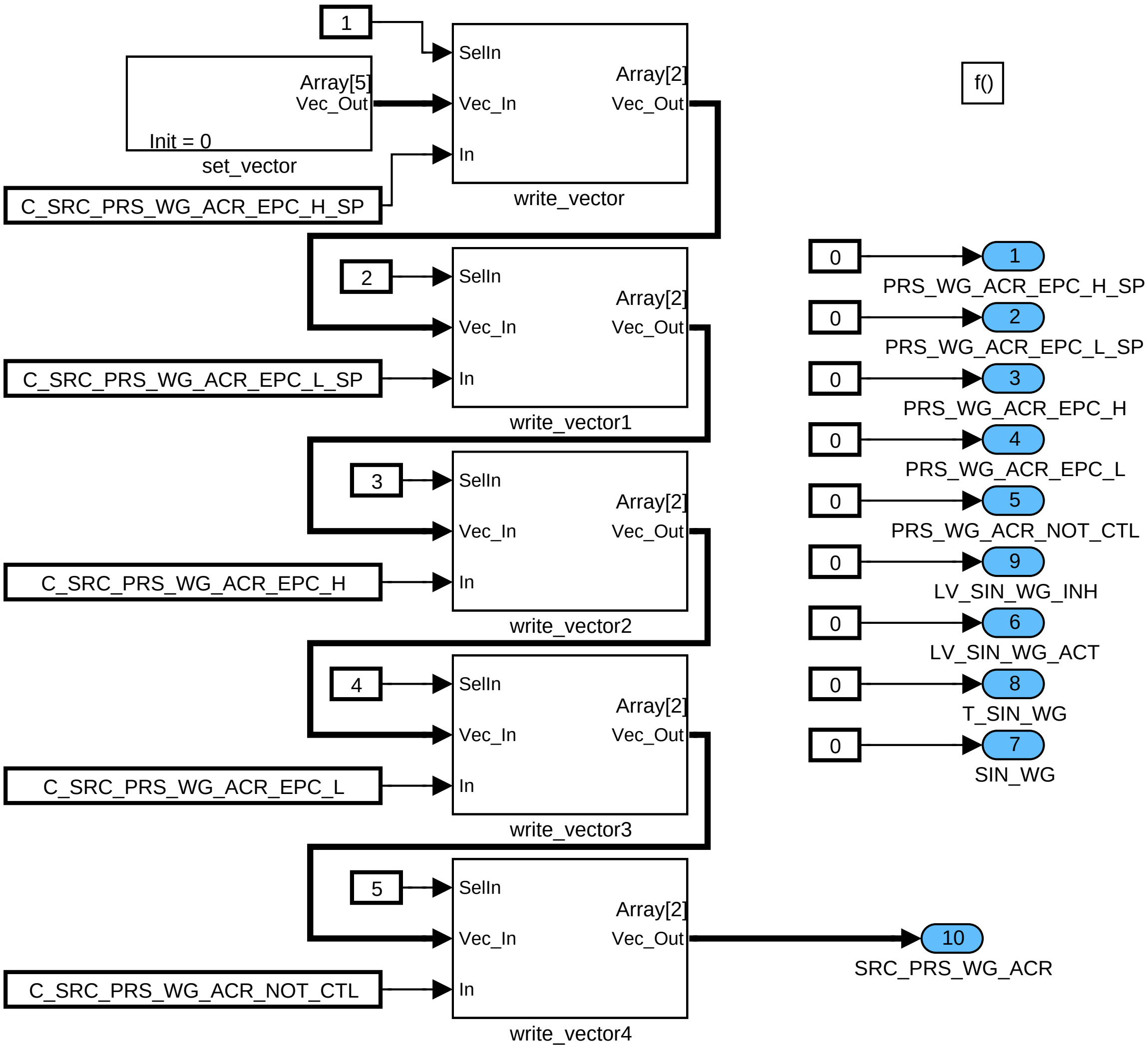


Figure 267 CHRG_ACCTLWGAI0/ INIT

9.87.1.2 SUBFUNCTION: operate_10ms

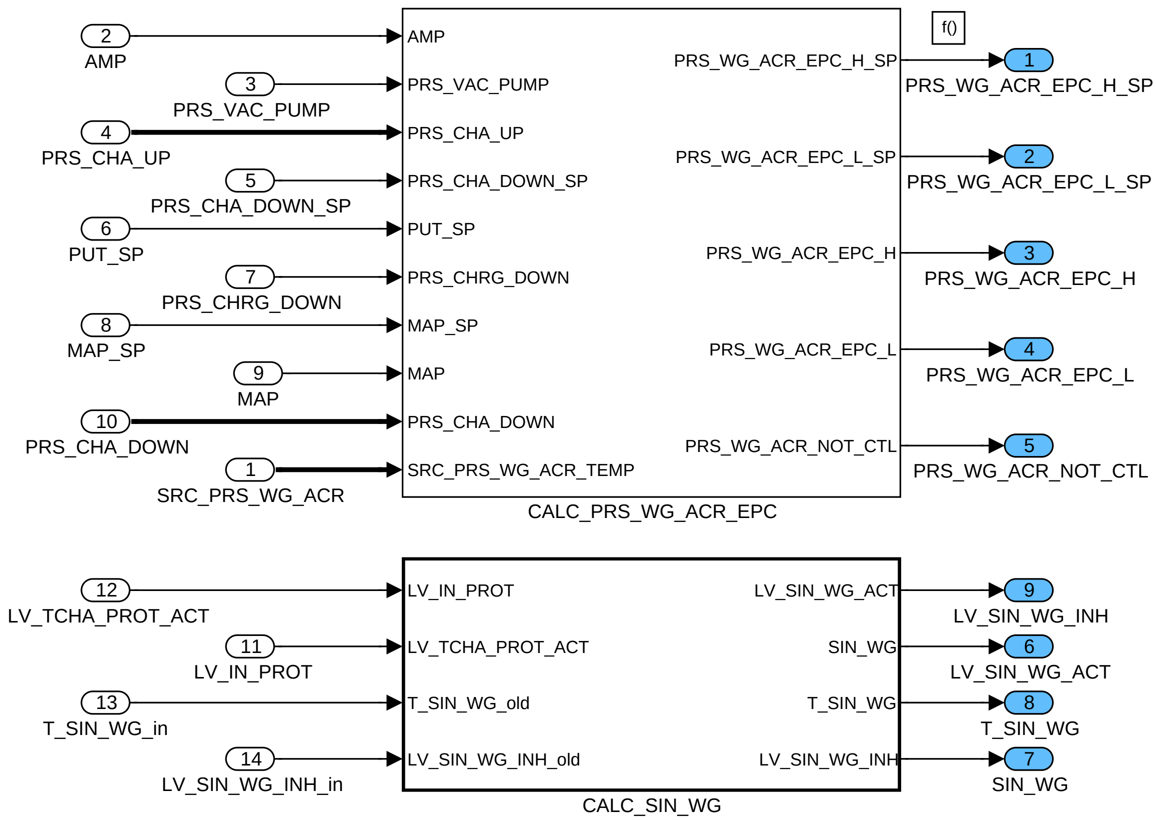


Figure 268 CHRG_ACCTLWGAI0/ operate_10ms

CHRG_ACCTLWGAI0/OPERATE_10MS/CALC_PRS_WG_ACR_EPC

The pressure in either of the two chambers of the pneumatic wastegate actuator can be controlled by means of an electropneumatic converter (EPC). Depending on which pressure is controlled, there can be different pressures at the high pressure and low pressure side of the EPC. These choices are defined here. Any of the input pressures can be copied to any output pressure. Because the configuration settings are checked only after ECU reset, these settings can not be changed during engine run.

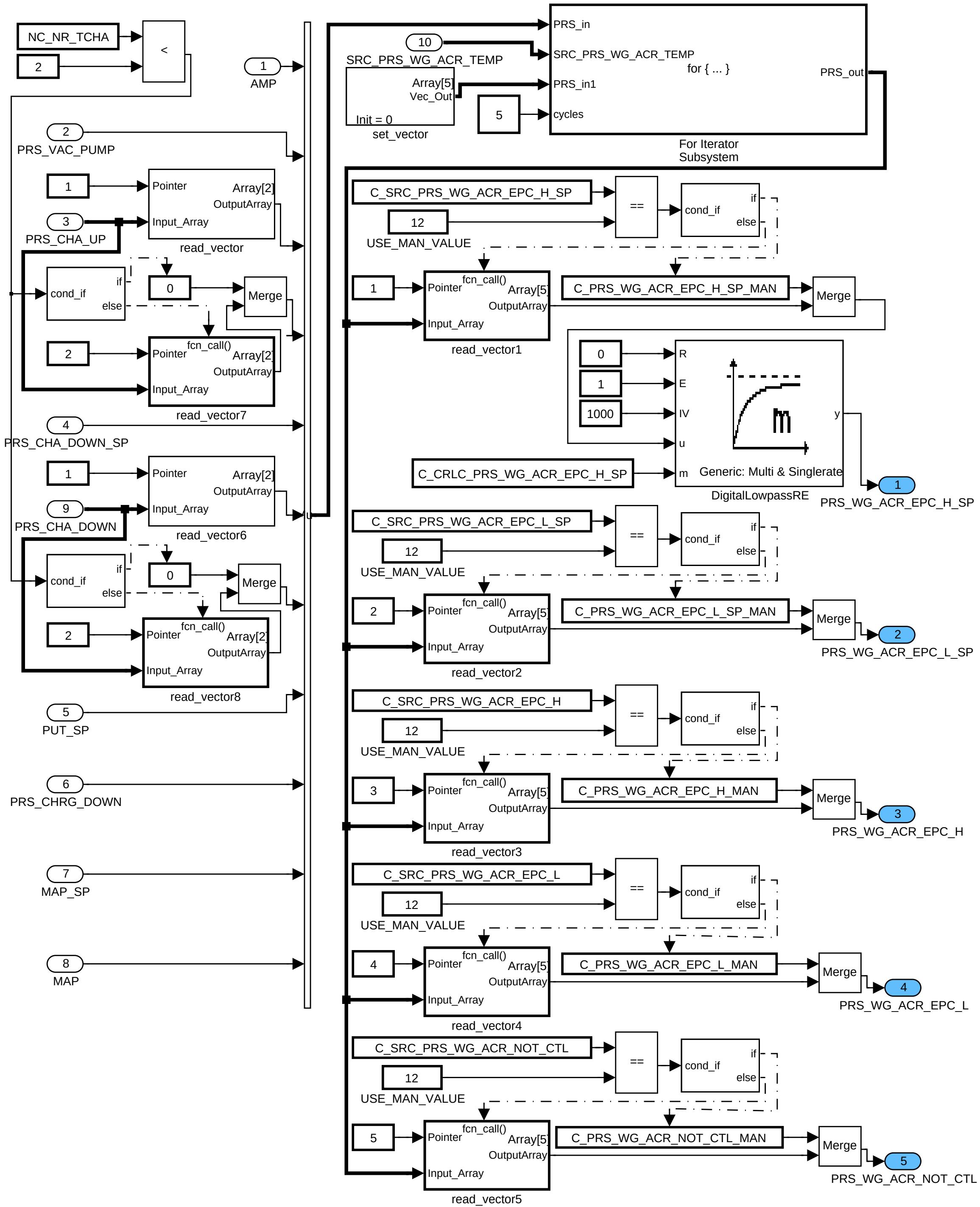


Figure 269 CHRG_ACCTLWGAI0/ operate_10ms/ CALC_PRS_WG_ACR_EPC

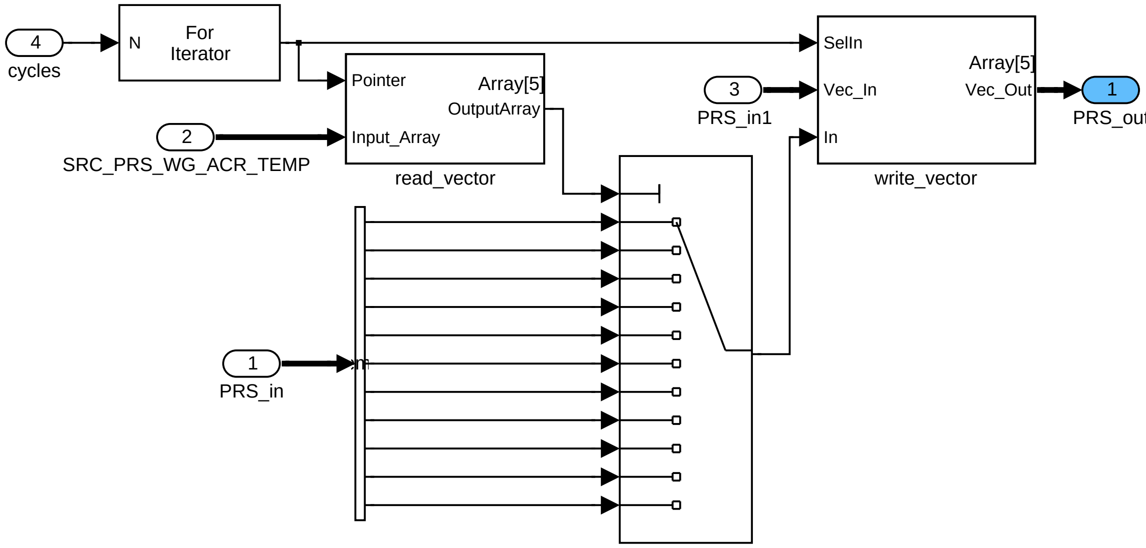


Figure 270CHRG_ACCTLWGAI0/ operate_10ms/ CALC_PRS_WG_ACR_EPC/ For Iterator

Subsystem

CHRG_ACCTLWGAI0/OPERATE_10MS/CALC_SIN_WG

System identification of the turbocharger can be performed by means of a sinusoidal excitation signal applied to the wastegate control signal or the wastegate opening setpoint. This sine signal with a tuneable frequency C_FRQ_SIN_WG is calculated here. It has no dimension, it has to be multiplied with the appropriate amplitude wherever it is used.

In IP_SIN_WG one quarter of the sine wave is stored, any angle is transformed into the first quadrant.

Whenever one of the flags LV_IN_PROT or LV_TCHA_PROT_ACT is set to 1 signaling dangerous conditions for the turbocharger or for the intake manifold the sine signal generation is stopped and can be resumed only after LC_SIN_WG_RST has been set for some time and reset again.

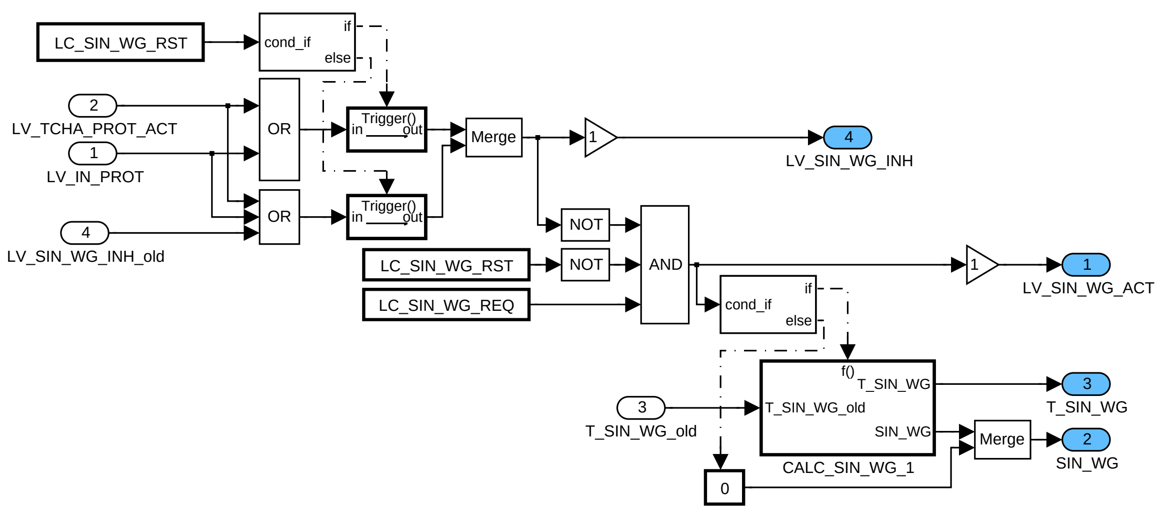


Figure 271 CHRG_ACCTLWGAI0/ operate_10ms/ CALC_SIN_WG

general specification

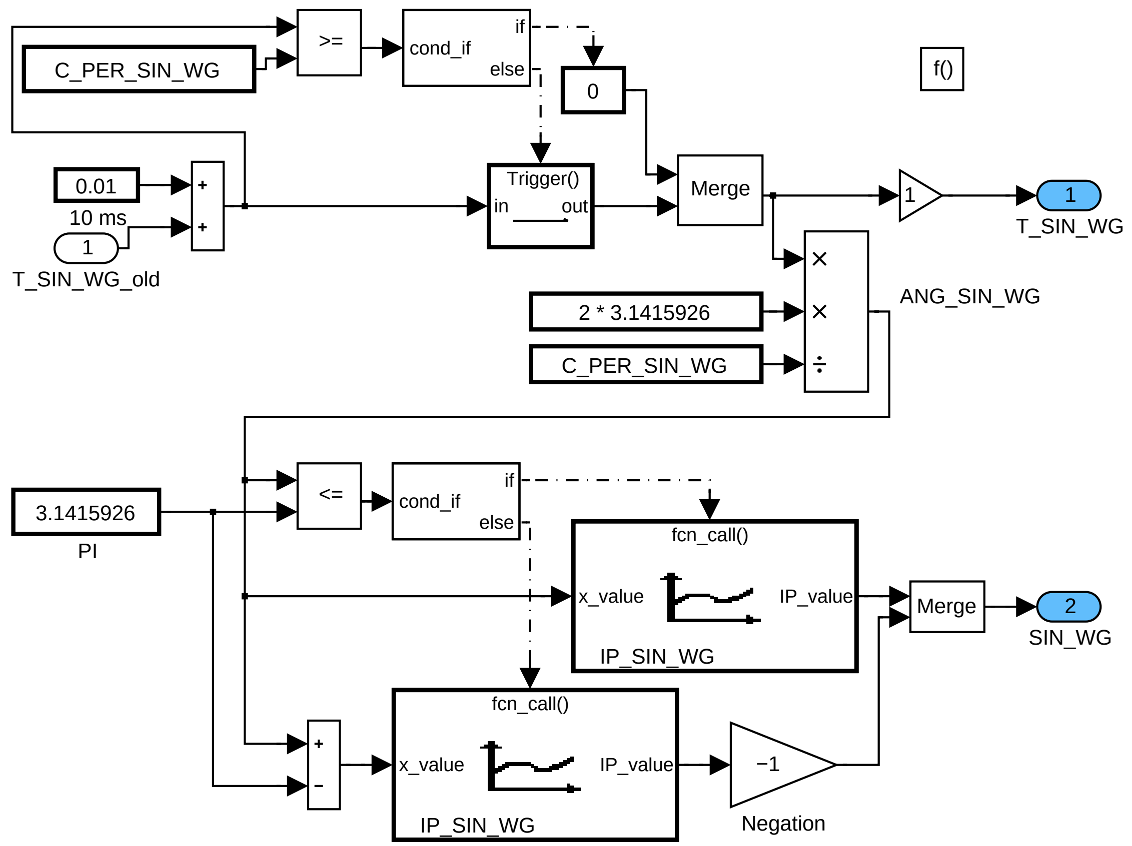


Figure 272 CHRG_ACCTLWGAI0/ operate_10ms/ CALC_SIN_WG/ CALC_SIN_WG_1

9.88 Waste gate position setpoint and exhaust gas force calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_WG_PRS_EX_SP	O/V	0...FFFFH	0...15.999756	2.44141E-4	Nm
torque caused by exhaust gas pressure forces at waste gate					
PSN_WG_SP	O/V	0...3FFFH	0...119.5	0.00729415	°
waste gate position setpoint					
FAC_TQ_WG_PRS_EX_SP	V	0...FFFFH	0...512	0.00781262	cm3
factor for calculation of torque at waste gate caused by exhaust gas pressure					

Input data:

LC_TCHA_CONF	SIN_WG	AR_RED_WG_SP_MMV	PRS_EX_DIF_TUR_SP

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PSN_WG_SP_AMPL	1	0...3FFFH	0...119.5	0.00729415	°
Waste gate position amplitude of sinusoidal excitation signal					
C_PSN_WG_SP_MAN	1	0...3FFFH	0...119.5	0.00729415	°
Manual waste gate position amplitude of sinusoidal excitation signal					
LC_PSN_WG_SP_MAN	1	0...1H	0...1	1	-
Logical calibration bit for switching between calculated and manual waste gate position setpoint					
IP_FAC_TQ_WG_PRS_EX_SP	16	0...FFFFH	0...512	0.00781262	cm3
LDP_PSN_WG_SP_IP_FAC_TQ_WG	16	0...3FFFH	0...119.5	0.00729415	°
factor for calculation of torque setpoint at waste gate caused by exhaust gas pressure					
IP_PSN_WG_SP	16	0...3FFFH	0...119.5	0.00729415	°
LDP_AR_RED_WG_SP_IP_PSN_WG_SP	16	0...FFFFH	0...58.592855	8.9407E-4	cm2
waste gate position setpoint					

9.88.1 CHRG_ISPCLPSNWG0

FUNCTION DESCRIPTION:

General information:

This module calculates the waste gate position setpoint and the torque at the waste gate caused by the exhaust gas pressure difference at the waste gate.

In this 1st step the reduced area setpoint is input, in later versions the flow setpoint through the waste gate, the pressure upstream and downstream the waste gate will be inputs.

Description:

The waste gate position setpoint is calculated dependent on the reduced area setpoint of the waste gate. The map IP_PSN_WG_SP represents this relationship. 0% means WG is closed (reduced area = 0) and 100% means WG is open (maximum reduced area).

The torque caused by gas forces depends on the force on the gate and the effective lever arm. The force depends on the exhaust gas pressure difference and the effective gate area. A positive value means a torque which opens the waste gate.

The influence of the effective area and the lever arm of the gate are summarised in the map IP_FAC_TQ_WG_PRS_EX_SP, because both depend on the waste gate position. The map represents the product of effective lever arm and effective gate area.

Application Condition

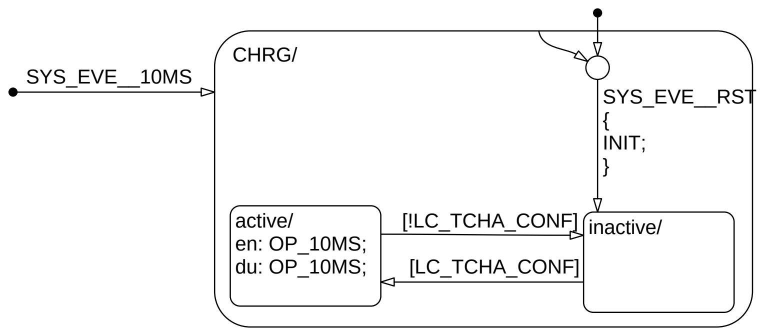


Figure 273 CHRG_ISPCLPSNWG0/ APP_CDN/ Chart

Function Description

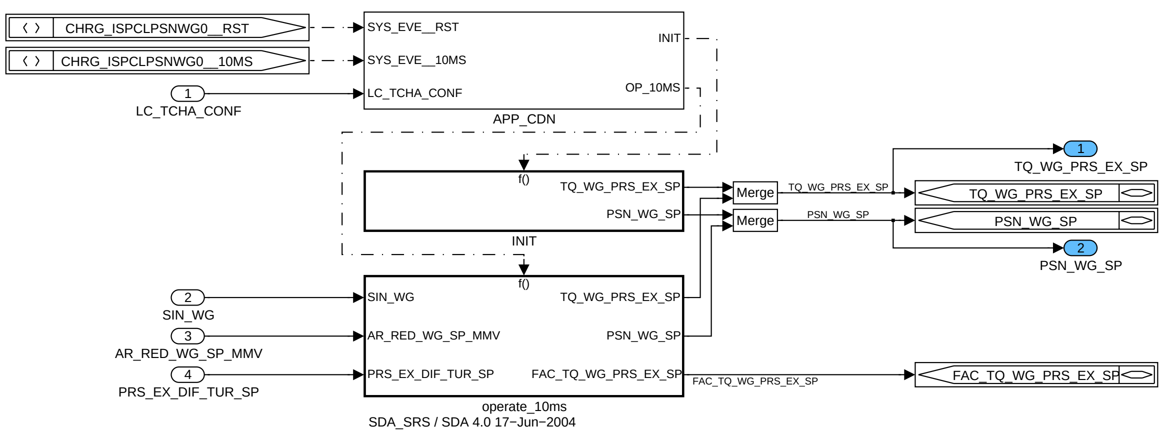


Figure 274 CHRG_ISPCLPSNWG0

9.88.1.1 SUBFUNCTION: INIT

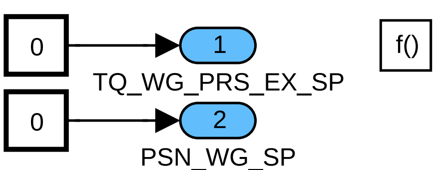


Figure 275 CHRG_ISPCLPSNWG0/ INIT

9.88.1.2 SUBFUNCTION: operate_10ms

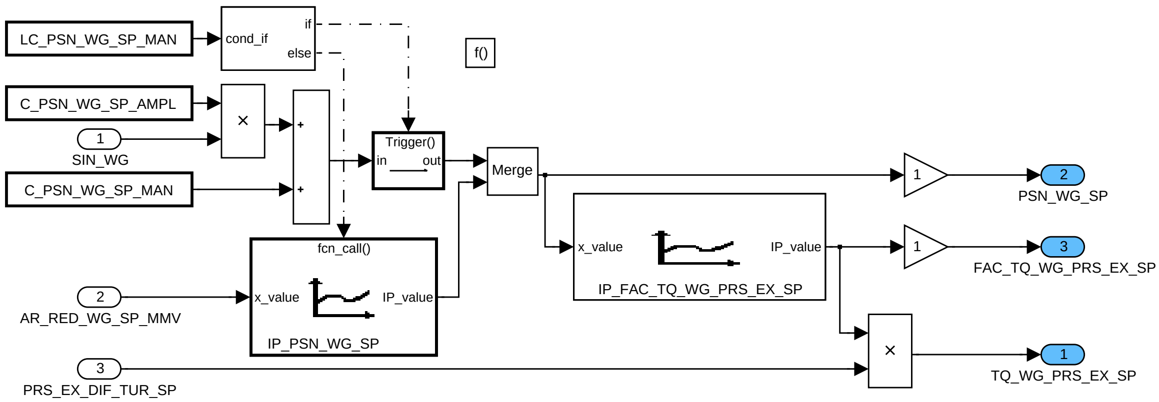


Figure 276 CHRG_ISPCLPSNWG0/ operate_10ms

9.89 Inverse turbine model

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
FAC_POW_TUR_1	V	0... FFFFH	0... 3.99993896484	61.0352e-6	[W/(hPa *sqrt(K)]
Factor for calculation of turbine power based on inverse turbine model					
FAC_POW_TUR_SP	V	0... FFFFH	0... 3.99993896484	61.0352e-6	[W/(hPa *sqrt(K)]
factor for calculation of turbine power (setpoint)					
FLOW_TUR_1	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Turbine flow based on inverse turbine model					
FLOW_TUR_COR_SP	V	0... FFFFH	0... 7.99987792969	122.07e-6	[(kg)/(sqrt(K)*s*bar)]
temperature corrected turbine flow at setpoint					
FLOW_TUR_SP	O/V	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Flow through turbine at setpoint					
N_TUR_COR_SP	V	0... FFFFH	0... 400000	6.1036088	[rpm]
Corrected turbine speed setpoint					
POW_TUR_SP	O/V	0... FFFFH	0... 65.535	1e-3	[kW]
Turbine power setpoint					
PQ_EX_SP	O/V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure quotient over turbine at setpoint					
PQ_TUR_1	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure quotient at turbine (PRS_EX / PRS_EX_PCAT_UP) based on inverse turbine model					
PQ_TUR_SP	V	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Pressure quotient setpoint for turbine					
PRS_EX_DIF_TUR_SP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure difference over turbine at setpoint					
PRS_EX_PCAT_UP_SP	V	0... FFFFH	0... 5434	0.0829175	[hPa]
Exhaust gas pressure pre catalyst setpoint					
PRS_EX_SP	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Exhaust gas pressure setpoint					
PRS_EX_TCHA	O/V	0... FFFFH	0... 5434	0.0829175	[hPa]
Pressure in the exhaust system based on inverse turbine model					

Input Data:

POW_CHA_SP	N_CHA_SP	AMP	FLOW_ENG_SP
TEG_TUR_UP_ABS_SQRT	C_TEG_TUR_UP_ABS_REF_SQRT	LC_TCHA_CONF	IP_PRS_EX_PCAT_UP_INC
IP_FLOW_TUR_COR	POW_CHA_MV	PRS_EX_PCAT_UP	N_TUR_COR
MAP	N	MAP_SP	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_FAC_MAP_PQ_TUR_SP	1	0... FFH	0... 0.99609375	3.90625e-3	[-]
Scaling factor between MAP and MAP_SP for the pulsation correction of PQ_TUR_SP					
C_FLOW_TUR_SP_MAN	1	0... FFFFH	0... 2047.96875	0.03125	[kg/h]

Manual setpoint for turbine flow					
C_POW_TUR_SP_MAN	1	0... FFFFH	0... 65.535	1e-3	[kW]
Manual setpoint for turbine power					
C_PQ_EX_SP_MAN	1	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
Manual setpoint for pressure quotient at turbine					
IP_FAC_AMP_COR_PQ_TUR_SP	6	0... FFH	0... 1.9921875	7.8125e-3	[-]
LDP_AMP_FAC_AMP_COR_PQ_TUR	6	0... FFFFH	0... 5434	0.0829175	[hPa]
Ambient pressure correction factor for MAP depending pulsation influence on PQ_TUR_SP					
IP_FAC_PLS_COR_PQ_TUR_SP	8*10	0... FFH	0... 1.9921875	7.8125e-3	[-]
LDP_N_IP_FAC_PLS_COR_PQ_TUR	10	0... 1FE0H	0... 8160	1	[rpm]
LDP_MAP_IP_FAC_PLS_COR_PQ_TUR	8	0... FFFFH	0... 5434	0.0829175	[hPa]
Correction term for pulsation influence to the turbine efficiency (PQ_TUR_SP correction)					
IP_PQ_TUR_SP	8*10	0... FFFFH	0... 15.9997558594	244.141e-6	[-]
LDPM_N_TUR_COR_1_CHRG	8	0... FFFFH	0... 400000	6.1036088	[rpm]
LDP_FAC_POW_TUR_SP_1_CHRG	10	0... FFFFH	0... 3.99993896484	61.0352e-6	[W/(hPa *sqrt(K)]
Pressure quotient setpoint for turbine					
IP_PRS_EX_PCAT_UP_INC_SP	16	0... FFFFH	0... 5434	0.0829175	[hPa]
LDP_FLOW_ENG_SP_IP_PRS_EX_PCAT	16	0... FFFFH	0... 2047.96875	0.03125	[kg/h]
Pressure increase due to the catalyst for exhaust pressure setpoint					
LC_FLOW_TUR_SP_MAN	1	0... 1H	0... 1	1	[-]
Switch for use of manual setpoint for turbine flow					
LC_POW_TUR_SP_MAN	1	0... 1H	0... 1	1	[-]
Switch for use of manual setpoint for turbine power					
LC_PQ_EX_SP_MAN	1	0... 1H	0... 1	1	[-]
Switch for use of manual setpoint for pressure quotient at turbine					

General Information

Turbine flow setpoint and turbine pressure setpoint for turbo charger precontrol are defined here.
For bi turbo systems all values describe one average exhaust bank.
Using the inverse turbine model the exhaust pressure upstream turbine PRS_EX_TCHA is calculated.

Application Conditions

Initialization: RST, ES2ERU
Recurrence: 10MS
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

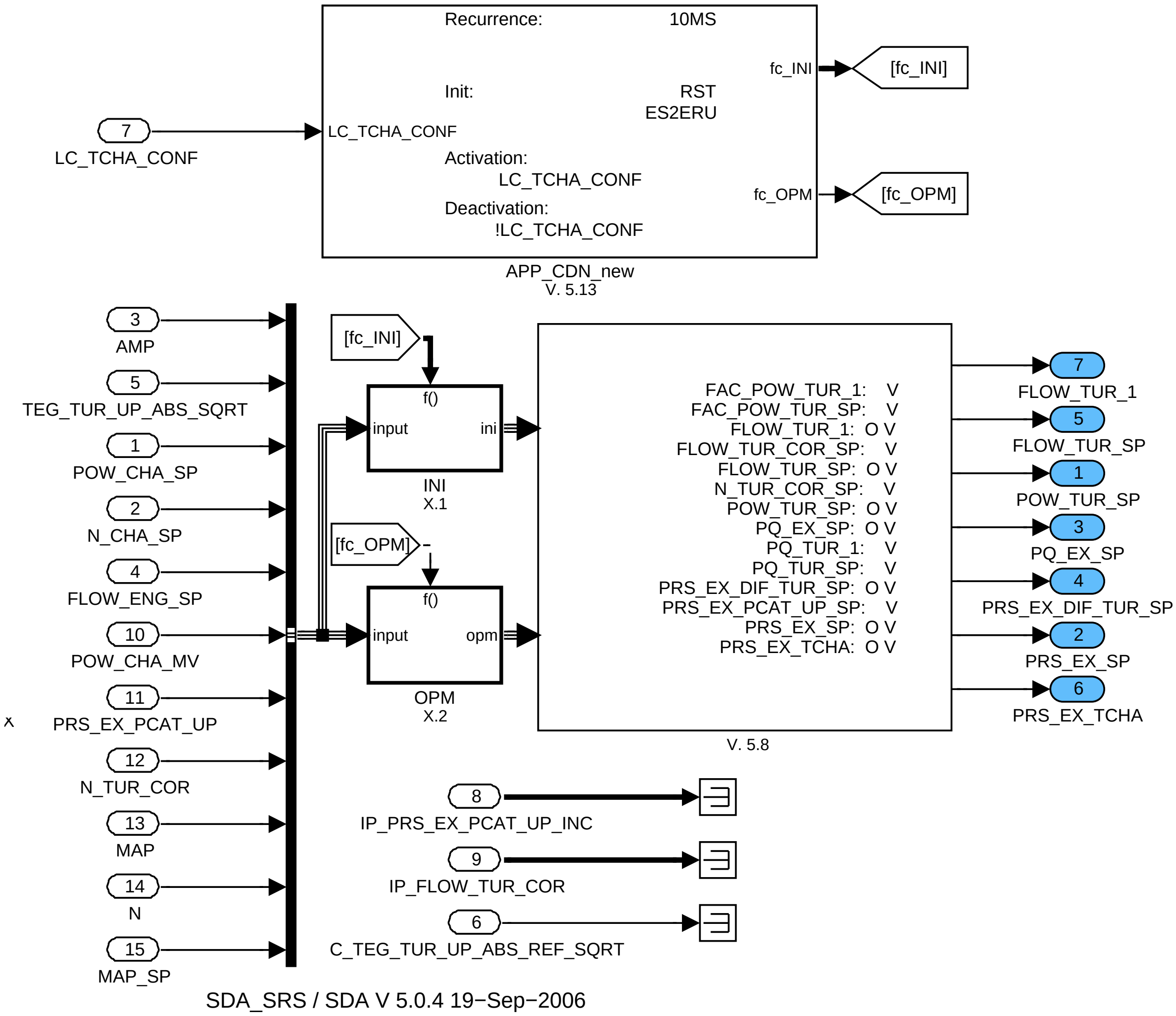


Figure 277:
Path: CHRG_ISPCLTUR0

9.89.1 Calculation at reset and engine stop to engine run

9.89.1.1 Initialisation

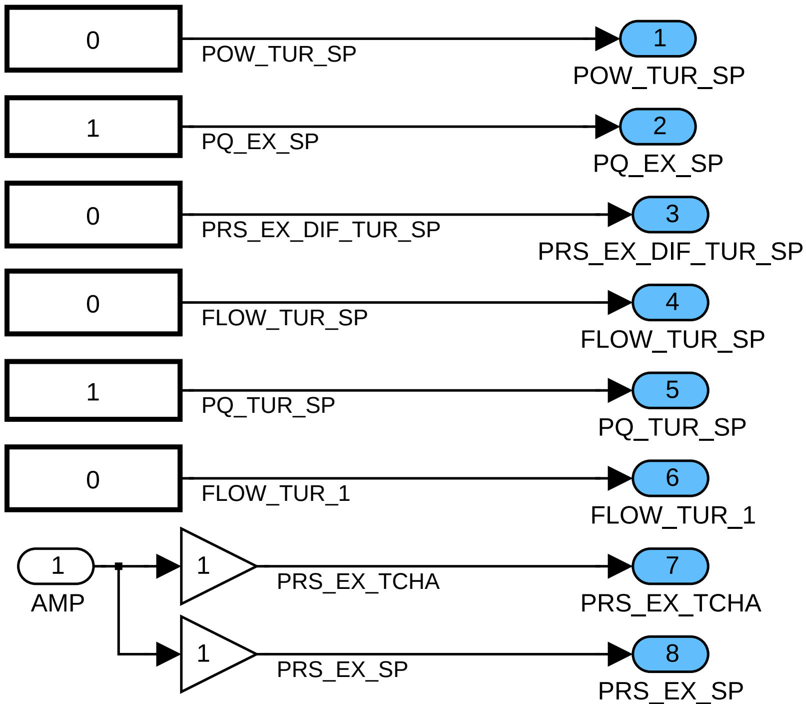


Figure 278:
Path: CHRG_ISPCLTUR0/INI/INI

9.89.2 Recurrence 10 ms

9.89.2.1 Calculation of turbine flow

9.89.2.1.1 Overview of calculation of actual turbine flow

9.89.2.1.1.1 Calculation of actual turbine flow

Exhaust pressure upstream turbine PRS_EX_TCHA is calculated using the inverse turbine model. Current engine operation conditions are corrected to reference conditions. In the turbine maps IP_PQ_TUR_SP and IP_FLOW_TUR_COR the behaviour of the turbine is stored. Additionally the map IP_FAC_PLS_COR_PQ_TUR_SP was introduced to consider a correction term on PQ_TUR due to pulsations which improve the turbine efficiency

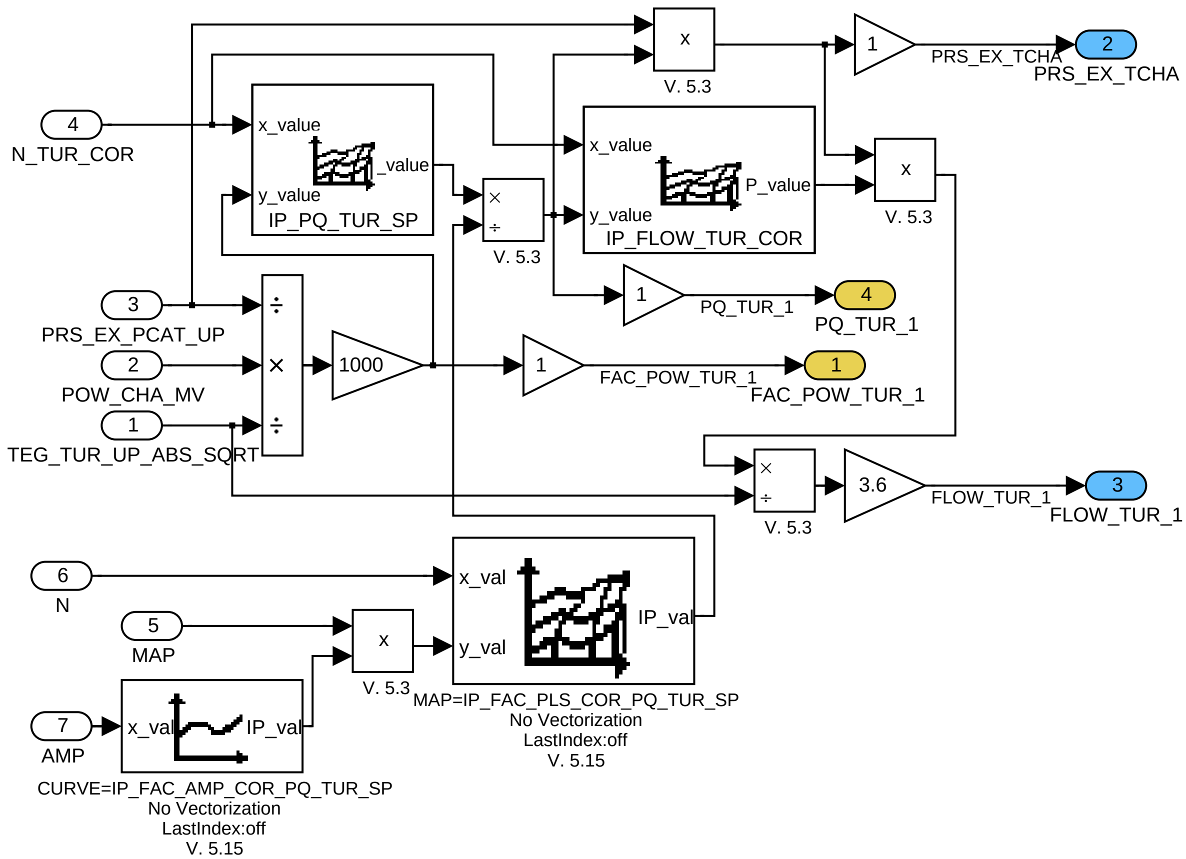


Figure 279:
Path: CHRG_ISPCLTUR0/OPM/CLC_OPM/CALC_FLOW_TUR_1/CLC_FLOW_TUR_1

9.89.2.1.2 Overview of calculation of turbine flow setpoint

9.89.2.1.2.1 Calculation of turbine flow setpoint

Current engine operation conditions are corrected to reference conditions. In the turbine maps IP_PQ_TUR_SP and IP_FLOW_TUR_COR the behaviour of the turbine is stored. Additionally a map for a correction term on PQ_TUR_SP was introduced (IP_FAC_PLS_COR_PQ_TUR_SP) which considers any pulsation effects that improves the turbine efficiency. Turbine pressure ratio and turbine flow setpoints can be replaced by manual settings.

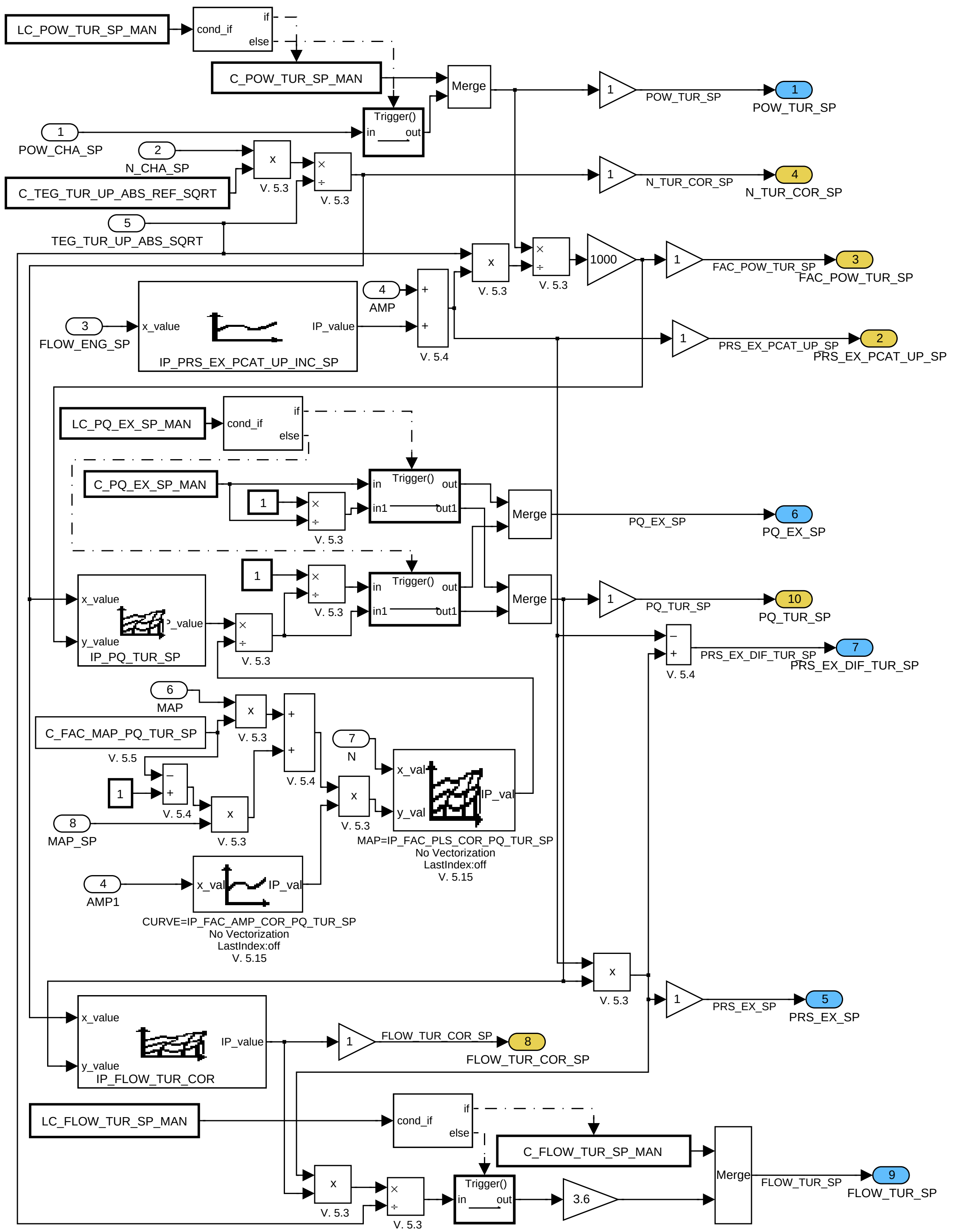


Figure 280:

Path: CHRG_ISPCLTUR0/OPM/CLC_OPM/CALC_FLOW_TUR_SP/CLC_FLOW_TUR_SP

A Diagnosis and Emergency Operation

Table of Contents

Index

O

P

R

U

V

A.1 Table of DTC

A.1.1 Ordered by DTC number

Failure Flag LV_ERR_xx	Error symptom ERR_SYM_xx		DTC-Code	Description
LV_ERR_CAM_STAT_IVVT_IN	8H	static CAM deviation	P0011	"A" Camshaft Position-Timing Over-Advanced or System Performance(Bank 1)
LV_ERR_CAM_DYN_IVVT_IN	8H	dynamic CAM deviation	P0011	"A" Camshaft Position-Timing Over-Advanced or System Performance(Bank 1)
LV_ERR_CAM_PAS_IVVT_IN	8H	Passive CAM deviation	P0011	"A" Camshaft Position-Timing Over-Advanced or System Performance(Bank 1)
LV_ERR_TRIP_IVVT_IN	1H	trip IVVT	P0011	"A" Camshaft Position-Timing Over-Advanced or System Performance(Bank 1)
LV_ERR_CAM_STAT_IVVT_EX	8H	static CAM deviation	P0014	"B" Camshaft Position - Timing Over-Advanced or System Performance
LV_ERR_CAM_DYN_IVVT_EX	8H	dynamic CAM deviation	P0014	"B" Camshaft Position - Timing Over-Advanced or System Performance
LV_ERR_CAM_PAS_IVVT_EX	8H	Passiv CAM deviation (Exhaust	P0014	"B" Camshaft Position - Timing Over-Advanced or System Performance
LV_ERR_TRIP_IVVT_EX	1H	trip IVVT	P0014	"B" Camshaft Position - Timing Over-Advanced or System Performance
LV_ERR_REF_CRK_CAM_IN	1H	camshaft to crankshaft reference violated	P0016	Crankshaft Position – Camshaft Position Correlation(Bank 1 Sensor A)
LV_ERR_TOOTH_OFF_IN	1H	Sudden drift "one tooth off"	P0016	Crankshaft Position – Camshaft Position Correlation(Bank 1 Sensor A)
LV_ERR_REF_CRK_CAM_EX	1H	camshaft to crankshaft reference violated	P0017	Crankshaft Position – Camshaft Position Correlation (Bank 1 / Sensor B)
LV_ERR_TOOTH_OFF_EX	1H	Sudden drift "one tooth off"	P0017	Crankshaft Position – Camshaft Position Correlation (Bank 1 / Sensor B)
LV_ERR_OBD_LSH_UP **(only bin)**	1H	heater circuit malfunction	P0030	HO2S Heater Control Circuit (Bank 1 / Sensor 1)
LV_ERR_OBD_VLD_LSH_UP **(only lin)**	1H	sensor temperature out of range	P0030	HO2S Heater Control Circuit (Bank 1 / Sensor 1)
LV_ERR_LSH_UP	2H	SCG	P0031	HO2S Heater Circuit low (Bank 1 / Sensor 1)
LV_ERR_LSH_UP	1H	SCB	P0032	HO2S Heater Circuit high (Bank 1 / Sensor 1)
LV_ERR_LSH_UP	4H	OL	P0032	HO2S Heater Circuit high (Bank 1 / Sensor 1)
LV_ERR_EL_RCL_ACR **(only TCI)**	2H	SCG	P0034	Turbo Charger Bypass Valve Control Circuit Low
LV_ERR_EL_RCL_ACR **(only TCI)**	1H	SCB	P0035	Turbo Charger Bypass Valve Control Circuit High
LV_ERR_EL_RCL_ACR **(only TCI)**	4H	OL	P0035	Turbo Charger Bypass Valve Control Circuit High
LV_ERR_OBD_LSH_DOWN	1H	heater circuit malfunction	P0036	HO2S Heater Control Circuit (Bank 1 / Sensor 2)
LV_ERR_LSH_DOWN	2H	SCG	P0037	HO2S Heater Circuit low (Bank 1 / Sensor 2)
LV_ERR_LSH_DOWN	1H	SCB	P0038	HO2S Heater Circuit high (Bank 1 / Sensor 2)
LV_ERR_LSH_DOWN	4H	OL	P0038	HO2S Heater Circuit high (Bank 1 / Sensor 2)
LV_ERR_TCHA_PROT **(only TCI)**	2H	Charger speed too high	P0049	Turbo/Super Charger Turbine Overspeed
LV_ERR_TAM_CAN **(only TCI)**	1H	TAM signal from CAN not valid	P0071	Ambient Air Temperature Sensor Range/Performance
LV_ERR_PLAUS_TAM **(only TCI)**	1H	Implausible high	P0071	Ambient Air Temperature Sensor Range/Performance
LV_ERR_PLAUS_TAM **(only TCI)**	2H	Implausible low	P0071	Ambient Air Temperature Sensor Range/Performance
LV_ERR_TAM_CAN **(only TCI)**	2H	Short circuit	P0072	Ambient Air Temperature Sensor Circuit Low Input
LV_ERR_EL_TAM **(only TCI)**	2H	SCG	P0072	Ambient Air Temperature Sensor Circuit Low Input
LV_ERR_EL_TAM **(only TCI)**	1H	SCB + OL	P0073	Ambient Air Temperature Sensor Circuit High Input
LV_ERR_TAM_CAN **(only TCI)**	4H	Open circuit	P0073	Ambient Air Temperature Sensor Circuit High Input

LV_ERR_SLV_IVVT_IN	2H	SCG	P0076	Intake Valve Control Solenoid Circuit Low(Bank 1)
LV_ERR_SLV_IVVT_IN	4H	OL	P0076	Intake Valve Control Solenoid Circuit Low(Bank 1)
LV_ERR_SLV_IVVT_IN	1H	SCB	P0077	Intake Valve Control Solenoid Circuit High(Bank 1)
LV_ERR_SLV_IVVT_EX	2H	SCG	P0079	Exhaust Valve control Solenoid Circuit Low (Bank 1)
LV_ERR_SLV_IVVT_EX	4H	OL	P0079	Exhaust Valve control Solenoid Circuit Low (Bank 1)
LV_ERR_SLV_IVVT_EX	1H	SCB	P0080	Exhaust Valve control Solenoid Circuit High (Bank 1)
LV_ERR_MAF_SCG_OC	2H	SCG + OL	P0102	Mass or Volume Air Flow Circuit Low Input
LV_ERR_MAF_SCP	1H	SCB	P0103	Mass or Volume Air Flow Circuit high Input
LV_ERR_LOAD_TPS_PLAUS	1H	additive adaption out of range	P0106	Manifold Absolute Pressure/Barometric Pressure Circuit Range/Performance
LV_ERR_LOAD_TPS_PLAUS	2H	filtered reduced area controller out of range	P0106	Manifold Absolute Pressure/Barometric Pressure Circuit Range/Performance
LV_ERR_LOAD_TPS_PLAUS	4H	Filtered ambient controller out of range	P0106	Manifold Absolute Pressure/Barometric Pressure Circuit Range/Performance
LV_ERR_LOAD_TPS_PLAUS	8H	Filtered ambient controller out of range	P0106	Manifold Absolute Pressure/Barometric Pressure Circuit Range/Performance
LV_ERR_MAP_PLAUS **(only TCI)**	1H	Plausibility error during engine stop	P0106	Manifold Absolute Pressure/Barometric Pressure Circuit Range/Performance
LV_ERR_EL_MAP	2H	SCG + OL	P0107	Manifold Absolute Pressure/Barometric Pressure Circuit Low Input
LV_ERR_EL_MAP	1H	SCB	P0108	Manifold Absolute Pressure/Barometric Pressure Circuit High Input
LV_ERR_EL_MAP	4H	OL (GND-Line)	P0108	Manifold Absolute Pressure/Barometric Pressure Circuit High Input
LV_ERR_DYN_TIA_IM_CYL	1H	Stuck signal	P0111	Intake Air Temperature Sensor1 Circuit Range/Performance
LV_ERR_INTM_TIA_IM_CYL	1H	Sensor signal intermittent	P0111	Intake Air Temperature Sensor 1 Circuit Range/Performance
LV_ERR_EL_TIA_IM_CYL	2H	SCG	P0112	Intake Air Temperature Sensor1 Circuit Low Input
LV_ERR_EL_TIA_IM_CYL	1H	SCB + OL	P0113	Intake Air Temperature Sensor1 Circuit High Input
LV_ERR_TCO_STUCK	8H	stuck signal	P0116	Engine Coolant Temperature Circuit Range/Perf.
LV_ERR_TCO_STUCK_H	8H	stuck signal	P0116	Engine Coolant Temperature Circuit Range/Perf.
LV_ERR_TCO_STUCK_RNG	8H	stuck signal	P0116	Engine Coolant Temperature Circuit Range/Perf.
LV_ERR_TCO_EL	2H	SCG	P0117	Engine Coolant Temperature Circuit Low Input
LV_ERR_TCO_EL	1H	SCB + OL	P0118	Engine Coolant Temperature Circuit High Input
LV_ERR_TCO_GRD	8H	gradient error	P0119	Engine Coolant Temperature Circuit Intermittent
LV_ERR_TPS_MAF_1 **(only ETC)**	4H	TPS1/MAF is not plausible	P0121	Throttle/Pedal Position Sensor/Switch "A" Circuit Range/Performance
LV_ERR_TPS_EL **(only ISA)**	2H	SCG	P0122	Throttle/Pedal Position Sensor/Switch "A" Circuit Low Input
LV_ERR_TPS_1 **(only ETC)**	2H	SCG	P0122	Throttle/Pedal Position Sensor/Switch "A" Circuit Low Input
LV_ERR_TPS_EL **(only ISA)**	1H	SCB + OL	P0123	Throttle/Pedal Position Sensor/Switch "A" Circuit High Input
LV_ERR_TPS_1 **(only ETC)**	1H	SCB + OL	P0123	Throttle/Pedal Position Sensor/Switch "A" Circuit High Input
LV_ERR_TCO_PLAUS	8H	plausibility fault	P0125	Insufficient Coolant Temperature for Closed Loop Fuel Control
LV_ERR_TH	8H	stuck open	P0128	Coolant Thermostat (Coolant Temp. below Thermostat Regulating Temp.)
LV_ERR_OC_LS_UP **(only bin)**	4H	OL	P0130	O2 Sensor Circuit(Bank 1/ Sensor 1)
LV_ERR_READY_LS_UP **(only bin)**	8H	Delayed readyness	P0130	O2 Sensor Circuit(Bank 1/ Sensor 1)
LV_ERR_SCG_LS_UP **(only bin)**	2H	SCG	P0131	O2 Sensor Circuit Low Voltage(Bank 1 / Sensor 1)

LV_ERR_EL_LSL_UP **(only lin)**	2H	SCG	P0131	O2 Sensor Circuit Low Voltage(Bank 1 / Sensor 1)
LV_ERR_SCP_LS_UP **(only bin)**	1H	SCB	P0132	O2 Sensor Circuit High Voltage(Bank 1 / Sensor 1)
LV_ERR_EL_LSL_UP **(only lin)**	1H	SCB	P0132	O2 Sensor Circuit High Voltage(Bank 1 / Sensor 1)
LV_ERR_FRQ_LS_UP **(only bin)**	8H	slow response	P0133	O2-Sensor Circuit Slow Response (Bank 1 / Sensor 1)
LV_ERR_DYN_VLD_LS_UP **(only lin)**	1H	Too slow dynamic	P0133	O2-Sensor Circuit Slow Response (Bank 1 / Sensor 1)
LV_ERR_PUC_LS_UP **(only bin)**	4H	plausibility fault	P0134	O2 Sensor Circuit No Activity Detected (Bank 1 / Sensor 1)
LV_ERR_STK_LS_UP **(only bin)**	8H	Sensor signal excursion	P0134	O2 Sensor Circuit No Activity Detected (Bank 1 / Sensor 1)
LV_ERR_TTIP_MES_LSH_UP **(only lin)**	1H	Sensor temperature out of range	P0135	O2 Sensor Heater Circuit (Bank 1 / Sensor 1)
LV_ERR_TTIP_MES_LSH_UP **(only lin)**	4H	Invalid sensor temperature	P0135	O2 Sensor Heater Circuit (Bank 1 / Sensor 1)
LV_ERR_OC_LS_DOWN	4H	OL	P0136	O2 Sensor Circuit(Bank 1/ Sensor 2)
LV_ERR_SCG_LS_DOWN	2H	SCG	P0137	O2 Sensor Circuit Low Voltage (Bank 1/ Sensor 2)
LV_ERR_SCP_LS_DOWN	1H	SCB	P0138	O2 Sensor Circuit High Voltage (Bank 1/ Sensor 2)
LV_ERR_SWT_LS_DOWN	8H	rich/lean switch time error	P0139	O2 Sensor Circuit Slow Response
LV_ERR_PUC_LS_DOWN	1H	Plausibility error	P0140	O2 Sensor Circuit No Activity Detected (Bank 1 / Sensor 2)
LV_ERR_PUE_LS_DOWN	8H	invalid signal after puc check	P0140	O2 Sensor Circuit No Activity Detected (Bank 1 / Sensor 2)
LV_ERR_FSD_LAM_LIM	1H	lambda control upper limit reached	P0170	Fuel trim (Bank 1)
LV_ERR_FSD_LAM_LIM	2H	lambda control lower limit reached	P0170	Fule trim (Bank 1)
LV_ERR_TOIL_PLAUS_H	2H	unplausible high signal	P0196	Engine Oil Temp. Sensor Range / Performance
LV_ERR_TOIL_PLAUS_L	1H	unplausible low signal	P0196	Engine Oil Temp. Sensor Range / Performance
LV_ERR_TOIL_STUCK	4H	signal stuck	P0196	Engine Oil Temp. Sensor Range / Performance
LV_ERR_TOIL	1H	SCG	P0197	Engine Oil Temp. Sensor Low Input
LV_ERR_TOIL	2H	SCB + OL	P0198	Engine Oil Temp. Sensor High Input
LV_ERR_TPS_MAF_2 **(only ETC)**	8H	TPS2/MAF is not plausible	P0221	Throttle/Pedal Position Sensor/Switch "B" Circuit Range/Performance
LV_ERR_TPS_2 **(only ETC)**	2H	SCG + OL	P0222	Throttle/Pedal Position Sensor/Switch "B" Circuit Low Input
LV_ERR_TPS_2 **(only ETC)**	1H	SCB	P0223	Throttle/Pedal Position Sensor/Switch "B" Circuit High Input
LV_ERR_RLY_EFP	1H	SCB	P0230	Fuel Pump Primary Circuit
LV_ERR_RLY_EFP	2H	SCG	P0230	Fuel Pump Primary Circuit
LV_ERR_RLY_EFP	4H	OL	P0230	Fuel Pump Primary Circuit
LV_ERR_CAP_H **(only TCI)**	8H	Charge air pressure too high	P0234	Turbo/Super Charger Overboost Condition
LV_ERR_PUT_PLAUS **(only TCI)**	1H	Measured PUT not plausible	P0236	Turbo/Super Charger Boost Sensor "A" Circuit Range/Performance
LV_ERR_PUT_PLAUS **(only TCI)**	2H	Measured PUT or AMP not plausible	P0236	Turbo/Super Charger Boost Sensor "A" Circuit Range/Performance
LV_ERR_PUT_PLAUS **(only TCI)**	4H	Measured PUT or MAP not plausible	P0236	Turbo/Super Charger Boost Sensor "A" Circuit Range/Performance
LV_ERR_EL_PUT **(only TCI)**	2H	SCG + OL	P0237	Turbo/Super Charger Boost Sensor "A" Circuit Low
LV_ERR_EL_PUT **(only TCI)**	1H	SCB	P0238	Turbo/Super Charger Boost Sensor "A" Circuit High
LV_ERR_PUT_WG_OPEN_AD **(only TCI)**	1H	PUT_WG_OPEN adaptation values below lower threshold	P0244	Turbo/Super Charger Wastegate Solenoid "A" Range/Performance
LV_ERR_PUT_WG_OPEN_AD **(only TCI)**	2H	PUT_WG_OPEN adaptation values above upper threshold	P0244	Turbo/Super Charger Wastegate Solenoid "A" Range/Performance

LV_ERR_PRS_DIF_WG_SP_AD(**only TCI)**	1H	PUT_CTL adaptation values below lower threshold	P0244	Turbo/Super Charger Wastegate Solenoid "A" Range/Performance
LV_ERR_PRS_DIF_WG_SP_AD(**only TCI)**	2H	PUT_CTL adaptation values above upper threshold	P0244	Turbo/Super Charger Wastegate Solenoid "A" Range/Performance
LV_ERR_WG **(only TCI)**	2H	SCG	P0245	Turbo/Super Charger Wastegate Solenoid "A" Low
LV_ERR_WG **(only TCI)**	1H	SCB	P0246	Turbo/Super Charger Wastegate Solenoid "A" High
LV_ERR_WG **(only TCI)**	4H	OL	P0246	Turbo/Super Charger Wastegate Solenoid "A" High
LV_ERR_IV_0	2H	SCG	P0261	Cylinder 1 – Injector Circuit Low
LV_ERR_IV_0	4H	OL	P0261	Cylinder 1 – Injector Circuit Low
LV_ERR_IV_0	1H	SCB	P0262	Cylinder 1 – Injector Circuit High
LV_ERR_IV_3	2H	SCG	P0264	Cylinder 2 – Injector Circuit Low
LV_ERR_IV_3	4H	OL	P0264	Cylinder 2 – Injector Circuit Low
LV_ERR_IV_3	1H	SCB	P0265	Cylinder 2 – Injector Circuit High
LV_ERR_IV_1	2H	SCG	P0267	Cylinder 3 – Injector Circuit Low
LV_ERR_IV_1	4H	OL	P0267	Cylinder 3 – Injector Circuit Low
LV_ERR_IV_1	1H	SCB	P0268	Cylinder 3 – Injector Circuit High
LV_ERR_IV_2	2H	SCG	P0270	Cylinder 4 – Injector Circuit Low
LV_ERR_IV_2	4H	OL	P0270	Cylinder 4 – Injector Circuit Low
LV_ERR_IV_2	1H	SCB	P0271	Cylinder 4 – Injector Circuit High
LV_ERR_CAP_L_BAS **(only TCI)**	8H	Basic charge air pressure too low	P0299	Turbo/Super Charger Underboost
LV_ERR_CAP_L **(only TCI)**	8H	Charge air pressure too low	P0299	Turbo/Super Charger Underboost
LV_ERR_MIS_MPL	1H	multiple cylinder misfire	P0300	Random/Multiple Cylinder Misfire Detected
LV_ERR_MIS_MPL	2H	random cylinder misfire	P0300	Random/Multiple Cylinder Misfire Detected
LV_ERR_MIS_0	1H	MIS_A	P0301	Cylinder 1 – Misfire detected
LV_ERR_MIS_0	2H	MIS_B1	P0301	Cylinder 1 – Misfire detected
LV_ERR_MIS_0	4H	MIS_B4	P0301	Cylinder 1 – Misfire detected
LV_ERR_MIS_3	1H	MIS_A	P0302	Cylinder 2 – Misfire detected
LV_ERR_MIS_3	2H	MIS_B1	P0302	Cylinder 2 – Misfire detected
LV_ERR_MIS_3	4H	MIS_B4	P0302	Cylinder 2 – Misfire detected
LV_ERR_MIS_1	1H	MIS_A	P0303	Cylinder 3 – Misfire detected
LV_ERR_MIS_1	2H	MIS_B1	P0303	Cylinder 3 – Misfire detected
LV_ERR_MIS_1	4H	MIS_B4	P0303	Cylinder 3 – Misfire detected
LV_ERR_MIS_2	1H	MIS_A	P0304	Cylinder 4 – Misfire detected
LV_ERR_MIS_2	2H	MIS_B1	P0304	Cylinder 4 – Misfire detected
LV_ERR_MIS_2	4H	MIS_B4	P0304	Cylinder 4 – Misfire detected
LV_ERR_SEG_AD_ER	2H	ER segment adaptation values at the limit	P0315	Crankshaft Position System Variation Not Learned
LV_ERR_KNKS_1	2H	invalid signal	P0326	Knock Sensor 1 Circuit Range/Performance (Bank 1 or Single Sensor)
LV_ERR_KNKS_1	1H	Signal out of range	P0326	Knock Sensor 1 Circuit Range/Performance (Bank 1 or Single Sensor)
LV_ERR_CRK_PLAUS	1H	Signal missing	P0335	Crankshaft Position Sensor A Circuit
LV_ERR_CRK_PLAUS	2H	Signal implausible	P0335	Crankshaft Position Sensor A Circuit
LV_ERR_CRK_SYN	1H	loss of CRK synchronisation	P0336	Crankshaft Position Sensor A Circuit Range/Performance
LV_ERR_CRK_TOOTH	1H	additional or missing teeth	P0336	Crankshaft Position Sensor A Circuit Range/Performance
LV_ERR_PER_CAM_IN	1H	camshaft segment period too short	P0340	Camshaft Position Sensor A Circuit Malfunction(Single Sensor)

LV_ERR_PLAUS_CAM_IN	1H	Signal missing	P0340	Camshaft Position Sensor A Circuit Malfunction(Single Sensor)
LV_ERR_SYN_CAM_IN	1H	camshaft synchronisation failure	P0340	Camshaft Position Sensor A Circuit Malfunction(Single Sensor)
LV_ERR_SYN_CRK_CAM_IN	1H	Signal invalid / CAM not valid for CRK synchronisation	P0340	Camshaft Position Sensor A Circuit Malfunction(Single Sensor)
LV_ERR_SYN_CAM_EX	1H	camshaft	P0365	Camshaft Position Sensor "B" Circuit (Bank 1)
LV_ERR_SYN_CRK_CAM_EX	1H	Signal invalid / CAM not valid for CRK synchronisation	P0365	Camshaft Position Sensor "B" Circuit (Bank 1)
LV_ERR_PLAUS_CAM_EX	1H	Signal missing	P0365	Camshaft Position Sensor "B" Circuit (Bank 1)
LV_ERR_PER_CAM_EX	1H	camshaft segment period too short	P0365	Camshaft Position Sensor "B" Circuit (Bank 1)
LV_ERR_CAT_DIAG	1H	CAT damaged	P0420	Catalyst System Efficiency below Threshold (Bank 1)
LV_ERR_MEC_CPS	1H	stuck in open position	P0441	Evap. Emission System Incorrect Purge Flow
LV_ERR_VAP_LEAK_2	1H	VAP_LEAK_2	P0442	Evap. Emission System – Leak detected (small leak)
LV_ERR_CPS	4H	OL	P0444	Evap. Emission System – Purge Ctrl. Valve Circuit Open
LV_ERR_CPS	1H	SCB	P0445	Evap. Emission System – Purge Ctrl. Valve Circuit Shorted
LV_ERR_CPS	2H	SCG	P0445	Evap. Emission System – Purge Ctrl. Valve Circuit Shorted
LV_ERR_SOV	4H	OL	P0447	Evap. Emission System–Vent Control Circuit Open
LV_ERR_SOV	1H	SCB	P0448	Evap. Emission System – Vent Control Circuit Shorted
LV_ERR_SOV	2H	SCG	P0448	Evap. Emission System – Vent Control Circuit Shorted
LV_ERR_MEC_SOV	1H	stuck in closed position	P0449	Evap. Emission System – Vent valve / Solenoid circuit
LV_ERR_DTP_PLAUS	1H	Signal not plausible	P0451	Evap. Emission System – Pressure Sensor Range / Performance
LV_ERR_DTP_PLAUS_H	2H	Plausibility high	P0451	Evap. Emission System – Pressure Sensor Range / Performance
LV_ERR_DTP_PLAUS_L	4H	Plausibility low	P0451	Evap. Emission System – Pressure Sensor Range / Performance
LV_ERR_DTP_EL	2H	SCG	P0452	Evap. Emission System – Pressure Sensor Low Input
LV_ERR_DTP_EL	1H	SCB + OL	P0453	Evap. Emission System – Pressure Sensor High Input
LV_ERR_DTP_NOISE	1H	Signal noisy	P0454	Evap. Emission System – Pressure Sensor Intermittent
LV_ERR_FUC_MISS	1H	FUC missing (detection during EVAP)	P0455	Evap. Emission System – Leak detected (Large leak)
LV_ERR_VAP_LEAK_10	4H	VAP_LEAK_10	P0455	Evap. Emission System – Leak detected (Large leak)
LV_ERR_VAP_LEAK_1	1H	VAP_LEAK_1	P0456	Evap. Emission System – Leak detected (very small leak)
LV_ERR_VAP_LEAK_1_MIL	1H	VAP_LEAK_1	P0456	Evap. Emission System – Leak detected (very small leak)
LV_ERR_FTL_STUCK	1H	Signal stuck	P0461	Fuel Level Sensor "A" Circuit Range/Performance
LV_ERR_FTL_EL	2H	SCG + OL	P0462	Fuel Level Sensor "A" Circuit Low Input
LV_ERR_FTL_EL	1H	SCB	P0463	Fuel Level Sensor "A" Circuit High Input
LV_ERR_FTL_INTM	1H	Signal intermittent	P0464	Fuel Level Sensor "A" Circuit Intermittent
LV_ERR_VS **(non OBD market)**	2H	plausibility fault	P0501	Vehicle Speed Sensor A Range/Performance

LV_ERR_BLS_BTS **(only ETC)**	4H	BLS-BTS not plausible	P0504	Brake Switch “A”/”B” Correlation
LV_ERR_ISC	2H	RPM lower than expected	P0506	Idle Air Control System – RPM lower than expected
LV_ERR_ISC	1H	RPM higher than expected	P0507	Idle Air Control System – RPM higher than expected
LV_ERR_ACP	2H	SCG + OL	P0532	A/C Refrigerant Pressure Sensor "A" Circuit Low Input
LV_ERR_ACP	1H	SCB	P0533	A/C Refrigerant Pressure Sensor "A" Circuit High Input
LV_ERR_PSP_PLAUS	1H	Signal implausibe	P0551	Power Steering Pressure Sensor/Switch Circuit Range/Performance
LV_ERR_PSP_SWI_PLAUS	1H	Signal implausibe	P0551	Power Steering Pressure Sensor/Switch Circuit Range/Performance
LV_ERR_PSP	2H	SCG + OL	P0552	Power Steering Pressure Sensor/Switch Circuit Low Input
LV_ERR_PSP	1H	SCB	P0553	Power Steering Pressure Sensor/Switch Circuit High Input
LV_ERR_RLY_MAIN	1H	not switched off	P0560	System Voltage
LV_ERR_RLY_MAIN	2H	not switched on	P0560	System Voltage
LV_ERR_RLY_MAIN_DLY	4H	switched on too slow	P0560	System Voltage
LV_ERR_VB_OC	4H	OL	P0560	System Voltage
LV_ERR_VB	2H	System Voltage low	P0562	System Voltage Low
LV_ERR_VB	1H	System Voltage High	P0563	System Voltage High
LV_ERR_CRU_SWI_1 **(only ETC)**	1H	invalid voltage range	P0564	Cruise Control Multi-Function Input "A" Circuit
LV_ERR_CRU_SWI_2 **(only ETC)**	2H	SET/COAST switch stuck	P0564	Cruise Control Multi-Function Input "A" Circuit
LV_ERR_CRU_SWI_3 **(only ETC)**	4H	RES/ACC switch stuck	P0564	Cruise Control Multi-Function Input "A" Circuit
LV_ERR_ECU	1H	ECU selftest error	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_MON **(only ETC)**	8H	General monitoring error	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_N_MAX_MON **(only ETC)**	2H	N_LIM monitoring error	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_TQI_MON **(only ETC)**	4H	torque monitoring	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_MON_3 **(only ETC)**	1H	RAM/ROM error on MC	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_MON_3 **(only ETC)**	4H	Error in selftest/Predrive check error	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_MON_3 **(only ETC)**	8H	Level 2' error	P0605	Internal Control Module Read Only Memory(ROM) Error
LV_ERR_OBD_VLD_LSH_UP **(only lin)**	4H	invalid sensor temperature	P0606	ECM/PCM Processor
LV_ERR_GEN_LOAD	2H	SCB + OL	P0625	Generator Field/ F Terminal Circuit Low
LV_ERR_GEN_LOAD_ES	2H	SCB + OL	P0625	Generator Field/ F Terminal Circuit Low
LV_ERR_GEN_LOAD_ES	1H	SCG	P0626	Generator Field/ F Terminal Circuit High
LV_ERR_VIN_DIAG	1H	VIN not programmed	P0630	VIN Not Programmed or Incompatible – ECM/PCM
LV_ERR_TPS_AD **(only ETC)**	1H	Adaptation conditions exceeded	P0638	Throttle Actuator Control Range/Performance
LV_ERR_TPS_AD **(only ETC)**	2H	Voltage value at LIH-adaptation outside range	P0638	Throttle Actuator Control Range/Performance
LV_ERR_TPS_AD **(only ETC)**	4H	Spring-test and CHK_LIH not passed	P0638	Throttle Actuator Control Range/Performance
LV_ERR_TPS_AD **(only ETC)**	8H	Adaptation values – lower mechanical stop – outside range	P0638	Throttle Actuator Control Range/Performance
LV_ERR_VCC_PVS_2	2H	SCG	P0642	Sensor Reference Voltage “A” Circuit Low
LV_ERR_VCC_PVS_2	4H	Sensor or wiring harness problem	P0642	Sensor Reference Voltage “A” Circuit Low
LV_ERR_VCC_PVS_2	1H	SCB	P0643	Sensor Reference Voltage “A” Circuit High

LV_ERR_RLY_ACCOUT	2H	SCG	P0646	A/C Clutch Relay Control Circuit Low
LV_ERR_RLY_ACCOUT	4H	OL	P0646	A/C Clutch Relay Control Circuit Low
LV_ERR_RLY_ACCOUT	1H	SCB	P0647	A/C Clutch Relay Control Circuit High
LV_ERR_MIL	1H	SCB	P0650	Malfunction Indicator Lamp(MIL) Control Circuit
LV_ERR_MIL	2H	SCG	P0650	Malfunction Indicator Lamp(MIL) Control Circuit
LV_ERR_MIL	4H	OL	P0650	Malfunction Indicator Lamp(MIL) Control Circuit
LV_ERR_TPS_VCC **(only ETC)**	2H	SCG	P0652	Sensor Reference Voltage “B” Circuit Low
LV_ERR_TPS_VCC **(only ETC)**	4H	Sensor or wiring harness problem	P0652	Sensor Reference Voltage “B” Circuit Low
LV_ERR_VCC_TPS **(only ISA)**	2H	SCG	P0652	Sensor Reference Voltage “B” Circuit Low
LV_ERR_VCC_TPS **(only ISA)**	4H	Sensor or wiring harness problem	P0652	Sensor Reference Voltage “B” Circuit Low
LV_ERR_TPS_VCC **(only ETC)**	1H	SCB	P0653	Sensor Reference Voltage “B” Circuit High
LV_ERR_VCC_TPS **(only ISA)**	1H	SCP	P0653	Sensor Reference Voltage “B” Circuit High
LV_ERR_VIM	2H	SCG	P0661	In-Manifold Tuning Valve Control Circuit low
LV_ERR_VIM	4H	OL	P0661	In-Manifold Tuning Valve Control Circuit low
LV_ERR_VIM	1H	SCB	P0662	In-Manifold Tuning Valve Control Circuit high
LV_ERR_VCC_PVS_1 **(only ETC)**	2H	SCG	P0698	Sensor Reference Voltage “C” Circuit Low
LV_ERR_VCC_PVS_1 **(only ETC)**	4H	Sensor or wiring harness problem	P0698	Sensor Reference Voltage “C” Circuit Low
LV_ERR_VCC_PVS_1 **(only ETC)**	1H	SCB	P0699	Sensor Reference Voltage “C” Circuit High
LV_ERR_VCC_SENS_SUB	2H	SCG	P06A4	Sensor Reference Voltage “D” Circuit Low
LV_ERR_VCC_SENS_SUB	4H	Sensor or wiring harness problem	P06A4	Sensor Reference Voltage “D” Circuit Low
LV_ERR_VCC_SENS_SUB	1H	SCB	P06A5	Sensor Reference Voltage “D” Circuit High
LV_ERR_CS **(only ETC)**	1H	SCB / OC	P0704	Clutch s/w Malfunction
LV_ERR_CS **(only ETC)**	2H	SCG	P0704	Clutch s/w Malfunction
LV_ERR_ISA_1 **(only ISA)**	2H	SCG	P1505	Idle Charge Actuator Signal Low of Coil #1
LV_ERR_ISA_1 **(only ISA)**	4H	OL	P1505	Idle Charge Actuator Signal Low of Coil #1
LV_ERR_ISA_1 **(only ISA)**	1H	SCB	P1506	Idle Charge Actuator Signal High of Coil #1
LV_ERR_ISA_2 **(only ISA)**	2H	SCG	P1507	Idle Charge Actuator Signal Low of Coil #2
LV_ERR_ISA_2 **(only ISA)**	4H	OL	P1507	Idle Charge Actuator Signal Low of Coil #2
LV_ERR_ISA_2 **(only ISA)**	1H	SCB	P1508	Idle Charge Actuator Signal High of Coil #2
LV_ERR_IMOB_TP_MOD	1H	passive mode invalid	P1674	Immobilizer TP status error
LV_ERR_IMOB_TP_PROG	1H	transponder prgramming error	P1675	Immobilizer TP programming error
LV_ERR_IMOB_MSG	1H	SMARTRA Invalid message (timing)	P1676	Immobilizer SMARTRA message error
LV_ERR_IMOB_MSG	2H	Smart-Key Invalid message (timing)	P1676	Immobilizer SMARTRA message error
LV_ERR_IMOB_RESP	1H	No answer from SMARTRA	P1690	Immobilizer Smartra No reponse
LV_ERR_IMOB_RESP	2H	No answer from Smart-Key	P1690	Immobilizer Smartra No reponse
LV_ERR_IMOB_SMARTRA_ANT	1H	Antenna coil error	P1691	Immobilizer Antena Coil Error
LV_ERR_IMOB_TP_DAT	1H	invalid transponder data	P1693	Immobilizer TP No/Invalid response
LV_ERR_IMOB_EMS_REQ	1H	invalid request from EMS or corrupted data	P1694	Immobilizer ECU Signal Error
LV_ERR_IMOB_MEM	1H	Checksum wrong in EPROM or impossible immobilizerstatus	P1695	Immobilizer EEPROM Error

LV_ERR_IMOB_AUTH	1H	Authentication failed	P1696	Immobilizer Aunthentication fail
LV_ERR_IMOB_TWICE_IGN_ON	1H	Max. number of tirals exceeded	P1699	Immobilizer Twice IG ON Overtrial
LV_ERR_IMOB_SMARTRA3	1H	Virgin SMARTRA3 at learnt EMS	P169A	Smartra3 Authentication fail
LV_ERR_IMOB_SMARTRA3	2H	Neutral SMARTRA3 at learnt EMS	P169A	Smartra3 Authentication fail
LV_ERR_IMOB_SMARTRA3	4H	Incorrect Authentication of EMS and SMNARTRA3	P169A	Smartra3 Authentication fail
LV_ERR_IMOB_SMARTRA3	8H	Locking of SMARTRA3	P169A	Smartra3 Authentication fail
LV_ERR_PORT_STUCK_OPEN	1H	Port Flap stuck open or open circuit of DC-Motor Supply	P2004	Intake Manifold Runner Control Stuck Open - Bank 1
LV_ERR_PORT_STUCK_CLOSE	1H	Port Flap stuck close	P2006	Intake Manifold Runner Control Stuck Closed - Bank 1
LV_ERR_PORT_EL	8H	Power stage error	P2008	Intake Manifold Runner Control Circuit/Open (Bank 1)
LV_ERR_PORT_AD	8H	Port Flap adaptation failed	P200A	Intake Manifold Runner Performance - Bank 1
LV_ERR_PORT_RNG	1H	Port Flap moves to slow or stucked between open and close	P200A	Intake Manifold Runner Performance - Bank 1
LV_ERR_PORT_RNG	2H	Port Flap shaft not connected properly to DC-Motor unit	P200A	Intake Manifold Runner Performance - Bank 1
LV_ERR_PORT_POTI	2H	SCG	P2016	Intake Manifold Runner Position Sensor/Switch Circuit Low (Bank 1)
LV_ERR_PORT_POTI	1H	SCB + OL	P2017	Intake Manifold Runner Position Sensor/Switch Circuit High (Bank 1)
LV_ERR_FTL_SUB_STUCK	1H	Signal stuck	P2066	Fuel Level Sensor "B" Performance
LV_ERR_FTL_SUB_EL	2H	SCG + OL	P2067	Fuel Level Sensor "B" Circuit Low
LV_ERR_FTL_SUB_EL	1H	SCB	P2068	Fuel Level Sensor "B" Circuit High
LV_ERR_FTL_SUB_INTM	1H	Signal intermittent	P2069	Fuel Level Sensor "B" Circuit Intermittent
LV_ERR_LAM_ADJ **(only bin)**	2H	Fuel trim too lean	P2096	Post Catalyst Fuel Trim System Too Lean (Bank 1)
LV_ERR_DELTA_I_LAM **(only lin)**	2H	AF too lean	P2096	Post Catalyst Fuel Trim System Too Lean (Bank 1)
LV_ERR_VLS_DOWN_DIF **(only lin)**	2H	AF too lean	P2096	Post Catalyst Fuel Trim System Too Lean (Bank 1)
LV_ERR_LAM_ADJ **(only bin)**	1H	Fuel trim too rich	P2097	Post Catalyst Fuel Trim System Too Rich (Bank 1)
LV_ERR_VLS_DOWN_DIF **(only lin)**	1H	AF too rich	P2097	Post Catalyst Fuel Trim System Too Rich (Bank 1)
LV_ERR_DELTA_I_LAM **(only lin)**	1H	AF too rich	P2097	Post Catalyst Fuel Trim System Too Rich (Bank 1)
LV_ERR_MTC_DR **(only ETC)**	8H	power stage error	P2101	Throttle Actuator Control Motor Circuit Range/Performance
LV_ERR_LIH_IS **(only ETC)**	1H	Engine Idle-Speed Limp Home Reaction	P2104	Throttle Actuator Control System - Forced Idle
LV_ERR_LIH_ENG_OFF **(only ETC)**	1H	Engine Shut-Down Limp Home Reaction	P2105	Throttle Actuator Control System - Forced Engine Shut-Down
LV_ERR_LIH_POW_LIM **(only ETC)**	1H	Engine Power Limitation Limp Home Reaction	P2106	Throttle Actuator Control System - Forced Engine Power Limitation
LV_ERR_LIH_N_LIM **(only ETC)**	1H	Engine Speed Limitation Limp Home Reaction	P2110	Throttle Actuator Control System - Forced Engine Speed Limitation
LV_ERR_MTC_CTL **(only ETC)**	1H	PWM-outside range	P2118	Throttle Actuator Control Motor Current Range/Performance
LV_ERR_MTC_CTL **(only ETC)**	4H	small signal detection	P2118	Throttle Actuator Control Motor Current Range/Performance
LV_ERR_MTC_CTL **(only ETC)**	8H	large signal detection	P2118	Throttle Actuator Control Motor Current Range/Performance
LV_ERR_TPS_AD_SPR **(only ETC)**	1H	lower position not reached	P2119	Throttle Actuator Control Throttle Body Range/Performance
LV_ERR_TPS_AD_SPR **(only ETC)**	2H	error lower return spring check	P2119	Throttle Actuator Control Throttle Body Range/Performance
LV_ERR_TPS_AD_SPR **(only ETC)**	4H	upper position not reached	P2119	Throttle Actuator Control Throttle Body Range/Performance

LV_ERR_TPS_AD_SPR **(only ETC)**	8H	error upper return spring check	P2119	Throttle Actuator Control Throttle Body Range/Performance
LV_ERR_TPS_ST_CHK **(only ETC)**	1H	Error in spring check	P2119	Throttle Actuator Control Throttle Body Range/Performance
LV_ERR_TPS_ST_CHK **(only ETC)**	2H	Error in limp-home-check	P2119	Throttle Actuator Control Throttle Body Range/Performance
LV_ERR_PVS_L_1 **(only ETC)**	2H	SCG + OL	P2122	Throttle/Pedal Position Sensor/Switch "D" Circuit Low Input
LV_ERR_PVS_H_1 **(only ETC)**	1H	SCB	P2123	Throttle/Pedal Position Sensor/Switch "D" Circuit High Input
LV_ERR_PVS_L_2 **(only ETC)**	2H	SCG + OL	P2127	Throttle/Pedal Position Sensor/Switch "E" Circuit Low Input
LV_ERR_PVS_H_2 **(only ETC)**	1H	SCB	P2128	Throttle/Pedal Position Sensor/Switch "E" Circuit High Input
LV_ERR_PVS_RATIO **(only ETC)**	8H	Ratio deviation	P2138	Throttle/Pedal Position Sensor/Switch “D” / “E” Voltage Correlation
LV_ERR_MWSS	1H	open wire	P2159	Vehicle Speed Sensor “B” Range/Performance
LV_ERR_VS (**OBD market)**	2H	plausibility fault	P2159	Vehicle Speed Sensor “B” Range/Performance
LV_ERR_FSD_1	1H	max limit (additive)	P2187	System Too Lean at Idle (Bank 1)
LV_ERR_FSD_1	2H	min limit (additive)	P2188	System Too Rich at Idle (Bank 1)
LV_ERR_FSD_1	4H	max limit (multiplicative; lower area)	P2191	System Too Lean at Higher Load (Bank 1)
LV_ERR_FSD_1	8H	min limit (multiplicative; lower area)	P2192	System Too Rich at Higher Load (Bank 1)
LV_ERR_SHIFT_AFL_LSL_UP **(only lin)**	1H	Characteristic line shift too lean	P2195	O2 Sensor Signal Biased/Stuck Lean - Bank 1 Sensor 1
LV_ERR_SHIFT_AFR_LSL_UP **(only lin)**	1H	Characteristic line shift too rich	P2196	O2 Sensor Signal Biased/Stuck Rich - Bank 1 Sensor 1
LV_ERR_AMP_PLAUS **(only TCI)**	1H	Plausibility error during engine stop	P2227	Barometric Pressure Circuit Range/Performance
LV_ERR_AMP **(only TCI)**	2H	SCG + OL	P2228	Barometric Pressure Circuit Low Input
LV_ERR_AMP **(only TCI)**	1H	SCB	P2229	Barometric Pressure Circuit High Input
LV_ERR_OC_LSL_UP **(only lin)**	4H	OC on Pump Cell (Vip)	P2237	O2 Sensor Pumping Current Circuit/Open - Bank 1 Sensor 1
LV_ERR_OC_LSL_UP **(only lin)**	1H	OC of Nernst Cell (Vn)	P2243	O2 Sensor Reference Voltage Circuit/Open - Bank 1 Sensor 1
LV_ERR_OC_LSL_UP **(only lin)**	2H	OC on Virtual Ground (Vg)	P2251	O2 Sensor Reference Ground Circuit/Open - Bank 1 Sensor 1
LV_ERR_PLAUS_CLOSE_RCL **only TCI)**	1H	Stuck close	P2261	Turbo/Super Charger Bypass Valve - Mechanical
LV_ERR_PLAUS_OPEN_RCL **only TCI)**	1H	Stuck open	P2261	Turbo/Super Charger Bypass Valve - Mechanical
LV_ERR_CHK_LS_DOWN **(only lin)**	2H	downstream oxygen sensor too lean	P2270	O2 Sensor Signal Stuck Lean - Bank 1 Sensor 2
LV_ERR_CHK_LS_DOWN **(only lin)**	1H	downstream oxygen sensor too rich	P2271	O2 Sensor Signal Stuck Rich - Bank 1 Sensor 2
LV_ERR_PUC_VLD_LS_UP **(only lin)**	1H	Low voltage during PUC	P2297	O2 Sensor Out of Range During Deceleration
LV_ERR_AIR_LSL_UP **(only lin)**	1H	Sensor on air	P2414	O2 Sensor Exhaust Sample Error
LV_ERR_T_ES_PLAUS	1H	Engine off timer not resetted or activated	P2610	ECM/PCM Internal Engine Off Timer Performance
LV_ERR_T_ES_PLAUS	8H	Engine off timer inaccurate	P2610	ECM/PCM Internal Engine Off Timer Performance
LV_ERR_OC_LSL_UP **(only lin)**	8H	OC on comp. resistor (Vrc)	P2626	O2 Sensor Pumping Current Trim Circuit/Open - Bank 1 Sensor 1
LV_ERR_CAN_BUS_OFF	4H	Communication Problem on the CAN	U0001	High Speed CAN Communication Bus
LV_ERR_TIMEOUT_TCU1	8H	No message received from TCU	U0101	CAN Communication BUS with TCU (Timeout)

A.1.2 Ordered by error name

For generation of the readiness-codes all diagnosis are connected with a CARB status (CARB_CC failures are only considered if they are calibrated as Emission relevant, see "Failure class"):

- CARB_CC Comprehensive component, continuous test
- CARB_MIS Misfire monitoring, continuous test
- CARB_FSD Fuel system monitoring, continuous test
- CARB_CAT Catalyst monitoring, sequential test
- CARB_HC Heated catalyst monitoring, sequential test
- CARB_EVAP Evaporative system monitoring, sequential test
- CARB_SA Secondary air monitoring, sequential test
- CARB_AC A/C system refrigerant monitoring, sequential test
- CARB_LS Oxygen sensor monitoring, sequential test
- CARB_LSH Oxygen sensor heater monitoring, sequential test
- CARB_EGR EGR system monitoring, sequential test

Calibration of C_ERR_CLAS_x (in HEX):

00 h: Diagnosis not active
10 h: Non CARB/EOBD Failure with medium priority, no MIL activation
43 h: emission relevant Failure with low priority, MIL activation
63 h: emission relevant Failure with high priority, MIL activation
93 h: emission relevant failure with high priority, MIL activation (in 1st DC)

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
ACP	1H	SCB	P0533	STD	CC	X	X	10	10	LV_ACCIN LV_RLY_ACCOUT V_ACP_MES_ENVD_H V_ACP_MES_ENVD
	2H	SCG + OL	P0532							
	4H									
	8H									
AIR_LSL_UP_1	1H	Sensor on air	P2414	NO	LS	-	-	-	43	T_AST_ENVD VLS_UP_DIAG_1_ENVD_H VLS_DOWN_1_ENVD_H TTIP_MES_LS_UP_1_ENVD_H
	2H									
	4H									
	8H									
AMP	1H	SCB	P2229	STD_INI	CC	X	X	00	00	MAP_MES_BAS_ENVD V_IGK TPS n_32
	2H	SCG + OL	P2228							
	4H									
	8H									
AMP_PLAUS	1H	Measured AMP not plausible	P2227	NO	CC	X	X	00	00	MAP_MES_BAS_ENVD V_IGK TPS n_32
	2H	Measured AMP or PUT not plausiblel	none							
	4H	Measured AMP or MAP not plausible	none							
	8H									
BLS_BTS	1H			STD_INI	CC	-	X	10	10	PV_AV_SEL_ENVD VS LV_BLS LV_BTS
	2H									
	4H	BLS-BTS not plausible	P0504							
	8H									
CAM_DYN_IVVT_EX_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_EX CAM_EX[1]
	2H									
	4H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	8H	dynamic CAM deviation	P0014							CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD
CAM_DYN_IVVT_IN_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_IN CAM_IN[1] CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD
	2H									
	4H									
	8H	dynamic CAM deviation	P0011							
CAM_PAS_IVVT_EX_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_EX CAM_EX[1] CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD
	2H									
	4H									
	8H	Passiv CAM deviation (Exhaust)	P0014							
CAM_PAS_IVVT_IN_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_IN CAM_IN[1] CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD
	2H									
	4H									
	8H	Passiv CAM deviation (Inlet)	P0011							
CAM_STAT_IVVT_EX_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_EX CAM_EX[1] CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD
	2H									
	4H									
	8H	static CAM deviation	P0014							
CAM_STAT_IVVT_IN_1	1H			STD_INI	EGR	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_IN CAM_IN[1] CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD
	2H									
	4H									
	8H	static CAM deviation	P0011							
CAN_BUS_OFF	1H			STD	CC	X	X	43	43	TCO_ST T_AST_ENVD V_IGK DIAG_INST_ENVD
	2H									
	4H	Communication Problem on the CAN	U0001							
	8H									
CAN_FRF	1H	Freeze frame requested via CAN	P0700	NO	CC	X	X	00	00	TAM TCU_OBD TCO_ST VS
	2H									
	4H									
	8H									
CAP_H	1H			MEM	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD PUT_SP_ENVD
	2H									
	4H									
	8H	Charge air pressure too high	P0234							
CAP_L	1H			MEM	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_WG_OPEN_ENVD PUT_SP_ENVD
	2H									
	4H									
	8H	Charge air pressure too low	P0299							
CAP_L_BAS	1H			MEM	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_WG_OPEN_ENVD PUT_SP_ENVD
	2H									
	4H									
	8H	Basic charge air pressure too low	P0299							
CAT_DIAG_1	1H	CAT damaged	P0420	NO	CAT	X	X	43	43	for bin: RATIO_CYC_AFL RATIO_CYC_AFR EFF_CAT_DIAG_1 T_DLY_I_AD_LAM_ADJ_1_ENVD for lin: VLS_DOWN_1_ENVD_L VLS_UP_1_ENVD_L EFF_CAT_DIAG_1 T_DLY_I_AD_LAM_ADJ_1_ENVD
	2H									
	4H									
	8H									
CHK_LS_DOWN_1	1H	downstream oxygen sensor too rich	P2271	NO	LS	-	-	-	43	VLS_UP_1_ENVD_H R_IT_LS_DOWN_1_ENVD_H LAMB_SP_1_ENVD_H VLS_DOWN_1_ENVD_H
	2H	downstream oxygen sensor too lean	P2270							
	4H									
	8H									
CPS	1H	SCB	P0445	MPL_STD_INI	CC	X	X	43	43	TAM CPPWM_ENVD DTP_ENVD CP_STATE
	2H	SCG	P0445							
	4H	OL	P0444							
	8H									
CRK_PLAUS	1H	Signal missing	P0335	NO	CC	X	X	43	43	ENG_STATE

Failure Flag LV_ERR_xx	Bit	Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	2H	Signal implausible	P0335							SEG_AD_MMV_ER_1_ENVD SEG_AD_MMV_ER_3_ENVD DIAG_INST_ENVD
	4H									
	8H									
CRK_SYN	1H	loss of CRK synchronisation	P0336	MEM	CC	X	X	43	43	ENG_STATE SEG_AD_MMV_ER_1_ENVD SEG_AD_MMV_ER_3_ENVD DIAG_INST_ENVD
	2H									
	4H									
	8H									
CRK_TOOTH	1H	additional or missing teeth	P0336	MEM	CC	X	X	43	43	ENG_STATE SEG_AD_MMV_ER_1_ENVD SEG_AD_MMV_ER_3_ENVD DIAG_INST_ENVD
	2H									
	4H									
	8H									
CRK_TOOTH_PER	1H	tooth period implausible	none	MEM	CC	X	X	00	00	ENG_STATE SEG_AD_MMV_ER_1_ENVD SEG_AD_MMV_ER_3_ENVD DIAG_INST_ENVD
	2H									
	4H									
	8H									
CRU_SWI_1	1H	invalid voltage range	P0564	STD_INI	CC	-	X	10	10	CRU_BAS_ENVD_H CRU_BAS_ENVD_L CRU_SWI_DRIV_STATE STATE_CRU
	2H									
	4H									
	8H									
CRU_SWI_2	1H			STD_INI	CC	-	X	10	10	CRU_BAS_ENVD_H CRU_BAS_ENVD_L CRU_SWI_DRIV_STATE STATE_CRU
	2H	SET/COAST switch stuck	P0564							
	4H									
	8H									
CRU_SWI_3	1H			STD_INI	CC	-	X	10	10	CRU_BAS_ENVD_H CRU_BAS_ENVD_L CRU_SWI_DRIV_STATE STATE_CRU
	2H									
	4H	RES/ACC switch stuck	P0564							
	8H									
CS	1H	SCB + OL	P0704	STD_INI	CC	-	X	10	10	TAM VS LV_BLS LV_BTS
	2H	SCG	P0704							
	4H									
	8H									
CTL_LSL_UP_1	1H	Limitation of pump current		MPL_STD_INI	LS	-	-	-	00	VLS_UP_DIAG_1_ENVD_H LAMB_SP_1_ENVD_H TTIP_MES_LS_UP_1_ENVD_H ERR_DIAG_CTL_LSL_UP[1]
	2H	Limitation of pump voltage								
	4H	VN voltage too high for reference pump current generation								
	8H									
CYL_BAL_LAM_0	1H	minimum diagnosis limit of cylinder individual lambda control reached	none	NO	CC	-	-	-	00	N_32 N_32 N_32 N_32
	2H	maximum diagnosis limit of cylinder individual lambda control reached	none							
	4H									
	8H									
CYL_BAL_LAM_1	1H	minimum diagnosis limit of cylinder individual lambda control reached	none	NO	CC	-	-	-	00	N_32 N_32 N_32 N_32
	2H	maximum diagnosis limit of cylinder individual lambda control reached	none							
	4H									
	8H									
CYL_BAL_LAM_2	1H	minimum diagnosis limit of cylinder individual lambda control reached	none	NO	CC	-	-	-	00	N_32 N_32 N_32 N_32
	2H	maximum diagnosis limit of cylinder individual lambda control reached	none							
	4H									
	8H									
CYL_BAL_LAM_3	1H	minimum diagnosis limit of cylinder individual lambda control reached	none	NO	CC	-	-	-	00	N_32 N_32 N_32 N_32
	2H	maximum diagnosis limit of cylinder individual lambda control reached	none							
	4H									
	8H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
DELTA_I_LAM_1	1H	AF too rich	P2097	NO	FSD	-	-	-	63	VLS_DOWN_1_ENVD_H VLS_UP_DIAG_1_ENVD_H LAMB_DELTA_I_LAM_ADJ_1 _ENVD_H VLS_DIF_LAM_ADJ_1_ENVD_ H
	2H	AF too lean	P2096							
	4H									
	8H									
DTP_EL	1H	SCB + OL	P0453	STD_INI	CC	X	X	43	43	TAM CP_STATE CPPWM_ENVD V_DTP_ENVD
	2H	SCG	P0452							
	4H									
	8H									
DTP_NOISE	1H	Signal noisy	P0454	STC	CC	X	X	43	43	DTP_DIF_ENVD_H DTP_DIF_ENVD_L CPPWM_ENVD V_DTP_ENVD
	2H									
	4H									
	8H									
DTP_PLAUS	1H	Signal not plausible	P0451	STC	CC	X	X	43	43,	TAM DTP_DIF_ENVD_H DTP_DIF_ENVD_L V_DTP_ENVD
	2H									
	4H									
	8H									
DTP_PLAUS_H	1H			STC	CC	X	X	43	43	TAM DTP_DIF_ENVD_H DTP_DIF_ENVD_L V_DTP_ENVD
	2H	Plausibility high	P0451							
	4H									
	8H									
DTP_PLAUS_L	1H			STC	CC	X	X	43	43	TAM DTP_DIF_ENVD_H DTP_DIF_ENVD_L V_DTP_ENVD
	2H									
	4H	Plausibility low	P0451							
	8H									
DYN_TIA_IM_CYL	1H	Stuck signal	P0111	STD_INI	CC	X	X	00	43	TIA_MES TIA_MES_DIF_DIAG_INTM TIA_MES_DIF_DIAG_DYN DIAG_INST_ENVD
	2H									
	4H									
	8H									
DYN_VLD_LS_UP_1	1H	Too slow dynamic	P0133	NO	LS	-	-	-	43	LAMB_SP_1_ENVD_H VLS_UP_DIAG_1_ENVD_H FAC_MV_DIAG_DYN_LSL_UP_ ENVD_H TTIP_MES_LS_UP_1_ENVD_H
	2H									
	4H									
	8H									
ECF_EL_1	1H	SCB	none	MPL_STD _INI	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H	SCG	none							
	4H	OL	none							
	8H									
ECF_EL_2	1H	SCB	none	MPL_STD _INI	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H	SCG	none							
	4H	OL	none							
	8H									
ECF_EL_PWM_1	1H	SCB	none	MPL_STD _INI	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H	SCG	none							
	4H	OL	none							
	8H									
ECU	1H	ECU selftest error	P0605	STD	CC	X	X	93	93	For non-ETC: ENG_STATE VB_MMV ECU_DIAG_SF_HB ECU_DIAG_SF_LB For ETC: ENVD_0_MON ENVD_1_MON ENVD_2_MON ENVD_3_MON
	2H									
	4H									
	8H									
EL_LSL_UP_1	1H	SCB	P0132	MEM	LS	-	-	-	43	STATE_ERR_EL_LSL_UP[1] T_AST_ENVD VLS_UP_1_ENVD_H LSHPWM_UP_1
	2H	SCG	P0131							
	4H									
	8H									
EL_MAP	1H	SCB	P0108	STD_INI	CC	X	X	43	43	VCC_PVS_2_DIAG_ENVD TPS VP_MAP_ENVD TIA_MES
	2H	SCG + OL	P0107							
	4H	OL (GND-Line)	P0108							
	8H									
EL_PUT	1H	SCB	P0238	STD_INI	CC	-	X	00	-	PUT_ENVD VCC_SENS_SUB_DIAG_ENVD PWM_WG_ENVD LV_CMD_RCL_OPEN
	2H	SCG + OL	P0237							
	4H									
	8H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
EL_RCL_ACR	1H	SCB	P0035	MPL_STD_INI	CC	-	X	00	-	LV_CMD_RCL_OPEN VB_MMV PUT_ENVD LV_SURGE
	2H	SCG	P0034							
	4H	OL	P0035							
	8H									
EL_TAM	1H	SCB + OL	P0073	STD_INI	CC	-	X	00	-	TIA_MES T_AST_ENVD TCO_ST V_IGK
	2H	SCG	P0072							
	4H									
	8H									
EL_TIA_IM_CYL	1H	SCB + OL	P0113	STD_INI	CC	X	X	00	43	T_AST_ENVD TCO_ST TIA_MES_ST TIA_MES
	2H	SCG	P0112							
	4H									
	8H									
FL_LS_DOWN_1	1H			MEM	LS	X	X	00	-	N_32 N_32 N_32 N_32
	2H	Signal low during full load	none							
	4H									
	8H									
FRQ_LS_UP_1	1H			STD	LS	X	X	43	-	DIAG_INST_ENVD RATIO_CYC_AFR VLS_UP_MMV_MIN_1_ENVD VLS_UP_MMV_MAX_1_ENVD
	2H									
	4H									
	8H	slow response	P0133							
FSD_1	1H	max limit (additive)	P2187	NO	FSD	X	X	63	63	FAC_LAM_COR_1_ENVD MFF_ADD_LAM_AD_ENVD_L FAC_L_RNG_LAM_AD_ENVD_H FAC_H_RNG_LAM_AD_ENVD_H
	2H	min limit (additive)	P2188							
	4H	max limit (multiplicative; lower area)	P2191							
	8H	min limit (multiplicative; lower area)	P2192							
FSD_FAC_H_1	1H	max limit (multiplicative; upper area)	none	NO	FSD	X	X	00	-	FAC_LAM_COR_1_ENVD MFF_ADD_LAM_AD_ENVD_L FAC_L_RNG_LAM_AD_ENVD_H FAC_H_RNG_LAM_AD_ENVD_H
	2H	min limit (multiplicative; upper area)	none							
	4H									
	8H									
FSD_H_RNG	1H	max limit (multiplicative; upper area)	none	NO	FSD	-	-	-	00	N_32 N_32 N_32 N_32
	2H	min limit (multiplicative; upper area)	none							
	4H									
	8H									
FSD_LAM_LIM_1	1H	lambda control upper limit reached	P0170	NO	FSD	X	X	63	63	FAC_L_RNG_LAM_AD_ENVD_H MFF_ADD_LAM_AD_ENVD_L FAC_LAM_COR_1_ENVD VLS_UP_1_ENVD_H
	2H	lambda control lower limit reached	P0170							
	4H									
	8H									
FTL_EL	1H	SCB	P0463	STD_INI	CC	X	X	43	43	V_FTL FTL_MMV_ENVD CONF_DIAGCP_VOL VS
	2H	SCG + OL	P0462							
	4H									
	8H									
FTL_INTM	1H	Signal intermittent	P0464	MEM	CC	X	X	43	43	WHEEL_GRD_MMV_ENVD_H WHEEL_GRD_MMV_ENVD_L FTL_MES FTL_MMV_ENVD
	2H									
	4H									
	8H									
FTL_STUCK	1H	Signal stuck	P0461	STD	CC	X	X	43	43	FTL_MMV_ENVD FTL_MMV_REF_INI_ENVD FTL_MMV_DIF_POS_DIAG_ENVD FTL_MMV_DIF_NEG_DIAG_ENVD
	2H									
	4H									
	8H									
FTL_STUCK_H	1H			STD	CC	X	X	00	00	FTL_MMV_ENVD FTL_MMV_REF_INI_ENVD FTL_MMV_DIF_POS_DIAG_ENVD FTL_MMV_DIF_NEG_DIAG_ENVD
	2H	Signal stuck	none							
	4H									
	8H									
FTL_SUB_EL	1H	SCB	P2068	STD_INI	CC	X	X	00	00	V_FTL_SUB_ENVD V_FTL_SUB_MMV_ENVD CONF_DIAGCP_VOL VS
	2H	SCG + OL	P2067							
	4H									
	8H									
FTL_SUB_INTM	1H	Signal intermittent	P2069	MEM	CC	X	X	00	00	WHEEL_GRD_MMV_ENVD_H WHEEL_GRD_MMV_ENVD_L
	2H									

Failure Flag LV_ERR_xx	Error symptom ERR_SYM_xx		DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	4H									V_FTL_SUB_ENVD V_FTL_SUB_MMV_ENVD
	8H									
FTL_SUB_STUCK	1H	Signal stuck	P2066	STD	CC	X	X	00	00	V_FTL_SUB_MMV_ENVD V_FTL_SUB_REF_INI_ENVD V_FTL_SUB_DIF_POS_DIAG_ENVD V_FTL_SUB_DIF_NEG_DIAG_ENVD
	2H									
	4H									
	8H									
FUC_MISS	1H	FUC missing (detection during EVAP)	P0455	STC	EVAP	X	X	43	43	TAM DTP_DIF_ACT_ENVD DTP_ENVD FAC_DIAGCP_VOL
	2H									
	4H									
	8H									
FUC_MISS_1	1H			STC	EVAP	X	X	00	00	TAM DTP_DIF_ACT_ENVD DTP_ENVD FAC_DIAGCP_VOL
	2H	FUC missing	none							
	4H									
	8H									
GEN_LOAD	1H			STD	CC	X	X	10	10	V_IGK GEN_LOAD GEN_LOAD_MMV VS
	2H	SCB + OL	P0625							
	4H									
	8H									
GEN_LOAD_ES	1H	SCG	P0626	STD_INI	CC	X	X	10	10	V_IGK GEN_LOAD GEN_LOAD_MMV VS
	2H	SCB + OL	P0625							
	4H									
	8H									
IGC_OL_0	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_OL_1	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_OL_2	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_OL_3	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCG_0	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCG_1	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCG_2	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCG_3	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCP_0	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCP_1	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCP_2	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IGC_SCP_3	Dummy (not supported by HW)		none	STD	CC	X	X	00	00	
IMOB_AUTH	1H	Authentication failed	P1696	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_CONF	1H	Non-immobilizer EMS in SMARTRA type immobilizer vehicle	none	STD	CC	X	X	00	00	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	Non-immobilizer EMS in Smart-Key type immobilizer vehicle	none							
	4H									
	8H									
IMOB_EMS_REQ	1H	invalid request from EMS or corrupted data	P1694	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_MEM	1H	checksum error	P1695	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_MIL	1H	SCB	none	STD	CC	X	X	00	00	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	SCG	none							
	4H	OL	none							
	8H									
IMOB_ MSG	1H	SMARTRA Invalid message (timing)	P1676	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	Smart-Key Invalid message (timing)	P1676							
	4H									
	8H									
IMOB_ RESP	1H	No answer from	P1690	STD	CC	X	X	10	10	V_IGK

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
		SMARTRA								MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	No answer from Smart-Key	P1690							
	4H									
	8H									
IMOB_SMARTRA3	1H	Virgin SMARTRA3 at Learnt EMS	P169A	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	Neutral SMARTRA3 at Learnt EMS	P169A							
	4H	Incorrect Authentication of EMS and SMARTRA3	P169A							
	8H	Locking of SMARTRA3	P169A							
IMOB_SMARTRA_ANT	1H	Antenna coil error	P1691	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_TESTER	1H	Wrong data from tester	none	STD	CC	X	X	00	00	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_TP_DAT	1H	invalid transponder data	P1693	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_TP_MOD	1H	programming mode invalid	P1674	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_TP_PROG	1H	transponder prgramming error	P1675	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H									
	4H									
	8H									
IMOB_TWICE_IGN_ON	1H	Max. number of trials exceeded for SMARTRA	P1699	STD	CC	X	X	10	10	V_IGK MAP_MES_ENVD VS ANS_ERR_COM_IMOB
	2H	Max. number of trials exceeded for Smart-Key								
	4H									
	8H									
INTM_TIA_IM_CYL	1H	Sensor signal intermittent	P0111	MEM	CC	X	X	00	43	TIA_MES TIA_MES_DIF_DIAG_INTM TIA_MES_DIF_DIAG_DYN DIAG_INST_ENVD
	2H									
	4H									
	8H									
ISA_1	1H	SCB	P1506	MPL_STD_INI	CC	X	-	-	-	ENG_STATE MAF_KGH_ENVD TPS ISAPWM_ISA_ENVD
	2H	SCG	P1505							
	4H	OL	P1505							
	8H									
ISA_2	1H	SCB	P1508	MPL_STD_INI	CC	X	-	-	-	ENG_STATE MAF_KGH_ENVD TPS ISAPWM_ISA_ENVD
	2H	SCG	P1507							
	4H	OL	P1507							
	8H									
ISC	1H	RPM higher than expected	P0507	STD_INI	CC	X	X	43	43	TPS_AV_ENVD_H MAP_MES_BAS_ENVD N_SP_IS_ENVD MAF_SP_ENVD
	2H	RPM lower than expected	P0506							
	4H									
	8H									
ISC_CST	1H	RPM higher than expected during CST	none	STD_INI	CC	X	X	0	0	TPS_AV_ENVD_H MAP_MES_BAS_ENVD N_SP_IS_ENVD MAF_SP_ENVD
	2H	RPM lower than expected during CST	none							
	4H									
	8H									
IV_0	1H	SCB	P0262	MPL_STD_INI	CC	X	X	43	43	ENG_STATE TI_1_0_ENVD_H TI_1_0_ENVD_L TI_2_MES_0_ENVD_H
	2H	SCG	P0261							
	4H	OL	P0261							
	8H									
IV_1	1H	SCB	P0268	MPL_STD_INI	CC	X	X	43	43	ENG_STATE TI_1_1_ENVD_H TI_1_1_ENVD_L TI_2_MES_1_ENVD_H
	2H	SCG	P0267							
	4H	OL	P0267							
	8H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
IV_2	1H	SCB	P0271	MPL_STD _INI	CC	X	X	43	43	ENG_STATE TI_1_2_ENVD_H TI_1_2_ENVD_L TI_2_MES_2_ENVD_H
	2H	SCG	P0270							
	4H	OL	P0270							
	8H									
IV_3	1H	SCB	P0265	MPL_STD _INI	CC	X	X	43	43	ENG_STATE TI_1_3_ENVD_H TI_1_3_ENVD_L TI_2_MES_3_ENVD_H
	2H	SCG	P0264							
	4H	OL	P0264							
	8H									
KNKS_1	1H	Signal out of range	P0326	STD	CC	X	X	43	43	KNKS_CMD_GAIN_AD_2 KNKS_CMD_GAIN_AD_3 NL_2_ENVD NL_3_ENVD
	2H	invalid signal	P0326							
	4H									
	8H									
LAM_ADJ_1	1H	Fuel trim too rich	P2097	NO	FSD	X	X	43	-	EFF_CAT_DIAG_1 VLS_DOWN_1_ENVD_H VLS_DOWN_1_ENVD_L ENG_STATE
	2H	Fuel trim too lean	P2096							
	4H									
	8H									
LIH_ENG_OFF	1H	Engine Shut-Down Limp Home	P2105	NO	CC	-	-	93	93	ERR_INTM_DIAG_INST_ACT CPU_LOAD_MAX_ENVD_H CPU_LOAD_MAX_ENVD_L TECU_MAX
	2H									
	4H									
	8H									
LIH_IS	1H	Engine Idle-speed Limp Home	P2104	NO	CC	X	-	93	93	ERR_INTM_DIAG_INST_ACT CPU_LOAD_MAX_ENVD_H CPU_LOAD_MAX_ENVD_L TECU_MAX
	2H									
	4H									
	8H									
LIH_N_LIM	1H	Engine Speed Limitation Limp Home	P2110	NO	CC	X	-	93	93	ERR_INTM_DIAG_INST_ACT CPU_LOAD_MAX_ENVD_H CPU_LOAD_MAX_ENVD_L TECU_MAX
	2H									
	4H									
	8H									
LIH_POW_LIM	1H	Engine Power Limitation Limp Home	P2106	NO	CC	X	-	93	93	ERR_INTM_DIAG_INST_ACT CPU_LOAD_MAX_ENVD_H CPU_LOAD_MAX_ENVD_L TECU_MAX
	2H									
	4H									
	8H									
LOAD_PLAUS	1H			MEM	CC	X	X	10	00	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H	Unplausibility caused by faulty load sensor	none							
LOAD_TPS_PLAUS	1H	additive adaption out of range	P0106	MEM	CC	X	X	00	43	for Non-ETC: TPS TPS_AV_ENVD_H ISAPWM_ISA_ENVD MAP_MES_BAS_ENVD For ETC: TPS_SP_ENVD TPS_AV_1_ENVD TPS_AV_2_ENVD MAP_MES_BAS_ENVD
	2H	multiplicative adaption out of range	P0106							
	4H	filtered reduced area controller out of range	P0106							
	8H	filtered ambient controller out of range	P0106							
LSH_DOWN_1	1H	SCB	P0038	MPL_STD _INI	LSH	X	X	43	43	TCO_ST LSHPWM_DOWN_1 R_IT_LS_DOWN_1_ENVD_H R_IT_LS_DOWN_1_ENVD_L
	2H	SCG	P0037							
	4H	OL	P0038							
	8H									
LSH_UP_1	1H	SCB	P0032	MPL_STD _INI	LSH	X	X	43	43	for Bin: TCO_ST LSHPWM_UP_1 R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L for Lin: TTIP_MES_LS_UP_1_ENVD_H T_AST_ENVD TEG_DYN_LS_UP_1_ENVD_H LSHPWM_UP_1
	2H	SCG	P0031							
	4H	OL	P0032							
	8H									
LSL_UP_IF_1	1H	Wrong initialisation of WRAF sensor controller	none	NO	LS	-	-	-	00	STATE_SYM_OBD_LSL_LSH_UP ERR_SYM_LSL_UP_IF[1] ERR_SYM_OFS_LSL_UP[1] TTIP_MES_LS_UP_1_ENVD_H
	2H	SPI communication error	none							
	4H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	8H									
MAF_DIF_CH_INT	1H			NO	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H	deviation between MAF and MAF setpoin	none							
MAF_SCG_OC	1H			STD_INI	CC	X	X	00	00	for non-ETC: ISAPWM_ISA_ENVD TPS TIA_MES MAF_MES_ENVD for ETC:TPS_SP_ENVD TPS TIA_MES MAF_MES_ENVD
	2H	SCG + OL	P0102							
	4H									
	8H									
MAF_SCP	1H	SCB	P0103	STD_INI	CC	X	X	00	00	for non-ETC: ISAPWM_ISA_ENVD TPS TIA_MES MAF_MES_ENVD for ETC:TPS_SP_ENVD TPS TIA_MES MAF_MES_ENVD
	2H									
	4H									
	8H									
MAP_PLAUS	1H	Measured MAP not plausible	P0106	NO	CC	X	X	00	00	TPS_SP_ENVD TPS_AV_1_ENVD TPS_AV_2_ENVD MAP_MES_BAS_ENVD
	2H	Measured MAP or AMP not plausible	none							
	4H	Measured MAP or PUT not plausible	none							
	8H	Measured MAP or MAF not plausible	none							
MEC_CPS	1H	stuck in open position	P0441	STC	EVAP	X	X	43	43	TIA DTP_DIF_COR_ENVD_H DTP_DIF_COR_ENVD_L DTP_ENVD
	2H									
	4H									
	8H									
MEC_IVVT_EX_1	1H			MEM	CC	X	X	00	00	PSN_AD_CAM_EX TOIL STATE_IVVT CAM_AV_IVVT_EX_1
	2H									
	4H									
	8H	mech IVVT error	none							
MEC_IVVT_IN_1	1H			MEM	CC	X	X	00	00	PSN_AD_CAM_IN_1 TOIL STATE_IVVT CAM_AV_IVVT_IN[1]
	2H									
	4H									
	8H	mech IVVT error	none							
MEC_SOV	1H	stuck in closed position	P0449	STD	CC	X	X	43	43	TIA_ST TAM CPPWM_ENVD V_DTP_ENVD
	2H									
	4H									
	8H									
MIL	1H	SCB	P0650	MPL_STD_INI	CC	X	X	10	10	TAM V_IGK TCO_ST VB_MMV
	2H	SCG	P0650							
	4H	OL	P0650							
	8H									
MIS_0	1H	MIS_A	P0301	NO	MIS	X	X	63	63	LOAD_MIN_SCDN_EQU_MIS_0 LOAD_MAX_SCDN_EQU_MIS_0 N_MIN_SCDN_EQU_MIS_0_ENVD N_MAX_SCDN_EQU_MIS_0_ENVD
	2H	MIS_B1	P0301							
	4H	MIS_B4	P0301							
	8H									
MIS_1	1H	MIS_A	P0303	NO	MIS	X	X	63	63	LOAD_MIN_SCDN_EQU_MIS_1 LOAD_MAX_SCDN_EQU_MIS_1 N_MIN_SCDN_EQU_MIS_1_ENVD N_MAX_SCDN_EQU_MIS_1_ENVD
	2H	MIS_B1	P0303							
	4H	MIS_B4	P0303							
	8H									
MIS_2	1H	MIS_A	P0304	NO	MIS	X	X	63	63	LOAD_MIN_SCDN_EQU_MIS_2 LOAD_MAX_SCDN_EQU_MIS_2 N_MIN_SCDN_EQU_MIS_2_ENVD N_MAX_SCDN_EQU_MIS_2_ENVD
	2H	MIS_B1	P0304							
	4H	MIS_B4	P0304							
	8H									
MIS_3	1H	MIS_A	P0302	NO	MIS	X	X	63	63	LOAD_MIN_SCDN_EQU_MIS_3 LOAD_MAX_SCDN_EQU_MIS_3 N_MIN_SCDN_EQU_MIS_3_ENVD N_MAX_SCDN_EQU_MIS_3_ENVD
	2H	MIS_B1	P0302							
	4H	MIS_B4	P0302							
	8H									
MIS_MPL	1H	multiple cylinder misfire	P0300	NO	MIS	X	X	63	63	TCO_ST FAC_LAM_COR_1_ENVD TIA VS
	2H	random cylinder misfire	P0300							
	4H									
	8H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
MON	1H			NO	CC	-	X	93	93	ENVD_0_MON ENVD_1_MON ENVD_2_MON ENVD_3_MON
	2H									
	4H									
	8H	General monitoring error	P0605							
MON_3	1H	RAM/ROM error on MC	P0605	STD	CC	-	X	93	93	ENVD_0_MON ENVD_1_MON ENVD_2_MON ENVD_3_MON
	2H									
	4H	Error in selftest/Predrive check error	P0605							
	8H	Level 2' error	P0605							
MTC_CTL	1H	PWM-outside range	P2118	STD	CC	-	X	93	93	MTCPWM_ENVD TPS_SP_ENVD TPS_AV_ENVD TIA
	2H									
	4H	small signal detection	P2118							
	8H	large signal detection	P2118							
MTC_DR	1H			STD_INI	CC	-	X	93	93	MTCPWM_MMV_ENVD VB_MMV TPS_AV_ENVD TIA
	2H									
	4H									
	8H	power stage error	P2101							
MWSS	1H	open wire	P2159	STD	CC	X	X	43	43	TCO_ST VS V_MIN_MWSS_DIAG V_MAX_MWSS_DIAG
	2H									
	4H									
	8H									
N_MAX_MON	1H			STD	CC	-	X	93	93	ENVD_0_MON ENVD_1_MON ENVD_2_MON ENVD_3_MON
	2H	N_LIM monitoring error	P0605							
	4H									
	8H									
OBD_LSH_DOWN_1	1H	heater circuit malfunction	P0036	NO	LSH	X	X	43	43	TCO_ST LSHPWM_DOWN_1 R_IT_LS_DOWN_1_ENVD_H R_IT_LS_DOWN_1_ENVD_L X
	2H									
	4H									
	8H									
OBD_VLD_LSH_UP_1	1H	Sensor temperature out of range	P0030	NO	LSH	-	-	-	43	STATE_SYM_OBD_LSL_LSH_UP ERR_SYM_LSL_UP_IF[1] ERR_SYM_OFS_LSL_UP[1] TTIP_MES_LS_UP_1_ENVD_H
	2H	Sensor readyness delayed	none							
	4H	Invalid sensor temperature	P0606							
	8H									
OBD_LSH_UP_1	1H	heater circuit malfunction	P0030	NO	LSH	X	X	43	-	TCO_ST LSHPWM_UP_1 R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L
	2H									
	4H									
	8H									
OC_LS_DOWN_1	1H			MEM	LS	X	X	43	43	R_IT_LS_DOWN_1_ENVD_L VLS_DOWN_1_ENVD_H T_AST_ENVD LSHPWM_DOWN_1
	2H									
	4H	OL	P0136							
	8H									
OC_LSL_UP_1	1H	OC of Nernst Cell (Vn)	P2243	NO	LS	-	-	-	43	ERR_SYM_OC_LSL_UP[1] VLS_UP_DIAG_1_ENVD_H LAMB_SP_1_ENVD_H TTIP_MES_LS_UP_1_ENVD_H
	2H	OC on Virtual Ground (Vg)	P2251							
	4H	OC on Pump Cell (Vip)	P2237							
	8H	OC on comp. resistor (Vrc)	P2626							
OC_LS_UP_1	1H			MEM	LS	X	X	43	-	VLS_UP_1_ENVD_H VLS_UP_1_ENVD_L R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L
	2H									
	4H	OL	P0130							
	8H									
OFS_LSL_UP_1	1H	Limitation of offset correction in low gain	none	STD_INI	LS	-	-	-	00	STATE_SYM_OBD_LSL_LSH_UP ERR_SYM_LSL_UP_IF ERR_SYM_OFS_LSL_UP TTIP_MES_LS_UP_1_ENVD_H
	2H	Limitation of offset correction in high gain	none							
	4H	Exceeded maximum duration of offset correction	none							
	8H									
PER_CAM_EX_1	1H	camshaft segment period too short	P0365	STD	CC	X	X	43	43	PSN_AD_CAM_EX TOIL CAM_SP_IVVT_EX CAM_EX_1
	2H									
	4H									
	8H									
PER_CAM_IN_1	1H	camshaft segment period too short	P0340	STD	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
PLAUS_CAM_EX_1	1H	Signal missing	P0365	NO	CC	X	X	43	43	PSN_AD_CAM_EX TOIL CAM_SP_IVVT_EX CAM_EX_1
	2H									
	4H									
	8H									
PLAUS_CAM_IN_1	1H	Signal missing	P0340	NO	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									
PLAUS_CLOSE_RCL	1H	Stuck close	P2261	MEM	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD LV_SURGE
	2H									
	4H									
	8H									
PLAUS_OPEN_RCL	1H	Stuck open	P2261	MEM	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD LV_SURGE
	2H									
	4H									
	8H									
PLAUS_TAM	1H	Implausible high	P0071	NO	CC	-	X	00	-	TIA_MES T_AST_ENVD TCO_ST V_IGK
	2H	Implausible low	P0071							
	4H									
	8H									
PLAUS_TIA_IM_CYL	1H	TIA increase error	none	STD_INI	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H	TIA decrease error	none							
	4H									
	8H									
PORT_AD	1H			MEM	CC	X	X	43	43	PORTPWM_CTL_ENVD PORT_AV_ENVD V_PORT_ENVD PORTPWM_ENVD
	2H									
	4H									
	8H	Port Flap adaptation failed	P200A							
PORT_EL	1H			MEM	CC	X	X	43	43	PORTPWM_ENVD PORTPWM_CTL_ENVD PORT_AV_ENVD V_PORT_ENVD
	2H									
	4H									
	8H	Power stage error	P2008							
PORT_POTI	1H	SCB + OL	P2017	MEM	CC	X	X	43	43	V_PORT_ENVD PORT_AV_ENVD PORTPWM_ENVD PORTPWM_CTL_ENVD
	2H	SCG	P2016							
	4H									
	8H									
PORT_RNG	1H	Port Flap moves to slow or stucked between open and close	P200A	MEM	CC	X	X	00	43	PORTPWM_CTL_ENVD PORT_AV_ENVD V_PORT_ENVD PORTPWM_ENVD
	2H	Port Flap shaft not connected proberly to DC-Motor unit	P200A							
	4H									
	8H									
PORT_STUCK_CLOSE	1H	Port Flap stuck close	P2006	MEM	CC	X	X	00	43	PORT_AV_ENVD V_PORT_ENVD PORTPWM_ENVD PORTPWM_CTL_ENVD
	2H									
	4H									
	8H									
PORT_STUCK_OPEN	1H	Port Flap stuck open or open circuit of DC-Motor Supply	P2004	MEM	CC	X	X	00	43	PORT_AV_ENVD V_PORT_ENVD PORTPWM_ENVD PORTPWM_CTL_ENVD
	2H									
	4H									
	8H									
PRS_DIF_WG_SP_AD	1H	PUT_CTL adaptation values below lower threshold	P0244	NO	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD PUT_SP_ENVD
	2H	PUT_CTL adaptation values above upper threshold	P0244							
	4H									
	8H									
PSP	1H	SCB	P0553	STD	CC	X	X	10	10	VS V_PSP_MES_ENVD_H V_PSP_MES_ENVD_L TAM
	2H	SCG + OL	P0552							
	4H									
	8H									
PSP_PLAUS	1H	Signal implausible	P0551	STD	CC	X	X	10	10	VS V_PSP_MES_ENVD_H
	2H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	4H									V_PSP_MES_ENVD_L TAM
	8H									
PSP_SWI_PLAUS	1H	Signal implausible	P0551	STD	CC	X	X	10	10	VS V_PSP_MES_ENVD_H V_PSP_MES_ENVD_L TAM
	2H									
	4H									
	8H									
PUC_LS_DOWN_1	1H	plausibility fault	P0140	MEM	LS	X	X	43	43	R_IT_LS_DOWN_1_ENVD_L VLS_UP_1_ENVD_L VLS_DOWN_1_ENVD_H VLS_DOWN_1_ENVD_L
	2H									
	4H									
	8H									
PUC_LS_UP_1	1H			MEM	LS	X	X	43	-	VLS_UP_1_ENVD_L FAC_LAM_COR_1_ENVD VLS_UP_MMV_MIN_1_ENVD VLS_UP_MMV_MAX_1_ENVD
	2H									
	4H	plausibility fault	P0134							
	8H									
PUC_VLD_LS_UP_1	1H	Low voltage during PUC	P2297	NO	LS	-	-	-	43	MAP_ENVD VLS_UP_DIAG_1_ENVD_H STATE_SYM_DIAG_PUC_LSL_UP TTIP_MES_LS_UP_1_ENVD_H
	2H									
	4H									
	8H									
PUE_LS_DOWN_1	1H			NO	LS	X	X	43	43	R_IT_LS_DOWN_1_ENVD_L VLS_UP_1_ENVD_L VLS_DOWN_1_ENVD_H VLS_DOWN_1_ENVD_L
	2H									
	4H									
	8H	invalid signal after puc check	P0140							
PUT_PLAUS	1H	Measured PUT not plausible	P0236	NO	CC	X	X	00	00	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD PUT_SP_ENVD
	2H	Measured PUT or AMP not plausible	none							
	4H	Measured PUT or MAP not plausible	none							
	8H									
PUT_WG_OPEN_AD	1H	PUT_WG_OPEN adaptation values below lower threshold	P0244	NO	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD PRS_AIC_DOWN_ENVD
	2H	PUT_WG_OPEN adaptation values above upper threshold	P0244							
	4H									
	8H									
PVS_BLS_BTS	1H			MEM	CC	-	X	00	00	V_PVS_1_ENVD V_PVS_2_ENVD LV_ERR_VCC_PVS_1 LV_ERR_VCC_PVS_2
	2H									
	4H									
	8H	plausibility fault	none							
PVS_H_1	1H	SCB	P2123	MEM	CC	-	X	93	93	V_PVS_1_ENVD V_PVS_2_ENVD LV_ERR_VCC_PVS_1 LV_ERR_VCC_PVS_2
	2H									
	4H									
	8H									
PVS_H_2	1H	SCB	P2128	MEM	CC	-	X	93	93	V_PVS_1_ENVD V_PVS_2_ENVD LV_ERR_VCC_PVS_1 LV_ERR_VCC_PVS_2
	2H									
	4H									
	8H									
PVS_L_1	1H			MEM	CC	-	X	93	93	V_PVS_1_ENVD LV_ERR_VCC_PVS_1 LV_PVS_H_R_1 LV_ERR_PVS_RATIO
	2H	SCG + OL	P2122							
	4H									
	8H									
PVS_L_2	1H			MEM	CC	-	X	93	93	V_PVS_2_ENVD LV_ERR_VCC_PVS_2 LV_PVS_H_R_2 LV_ERR_PVS_RATIO
	2H	SCG + OL	P2127							
	4H									
	8H									
PVS_MOVE	1H			MEM	CC	-	X	00	00	V_PVS_1_ENVD V_PVS_2_ENVD LV_ERR_VCC_PVS_1 LV_ERR_VCC_PVS_2
	2H									
	4H	MOVE_MAX	none							
	8H									
PVS_RATIO	1H			MEM	CC	-	X	93	93	V_PVS_1_ENVD V_PVS_2_ENVD LV_ERR_VCC_PVS_1 LV_ERR_VCC_PVS_2
	2H									
	4H									
	8H	Ratio deviation	P2138							

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
READY_LS_UP_1	1H			MEM	LS	X	X	43	-	VLS_UP_1_ENVD_H VLS_UP_1_ENVD_L R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L
	2H									
	4H									
	8H	Delayed readyness	P0130							
REF_CRK_CAM_EX_1	1H	camshaft to crankshaft reference violated	P0017	STD_INI	CC	X	X	43	43	PSN_AD_CAM_EX_1 TOIL CAM_SP_IVVT_EX CAM_EX[1]
	2H									
	4H									
	8H									
REF_CRK_CAM_IN_1	1H	camshaft to crankshaft reference violated	P0016	STD_INI	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									
RLY_ACCOUT	1H	SCB	P0647	MPL_STD_INI	CC	X	X	10	10	TIA TCO_ST LV_RLY_ACCOUT_CTRL V_IGK
	2H	SCG	P0646							
	4H	OL	P0646							
	8H									
RLY_EFP	1H	SCB	P0230	MPL_STD_INI	CC	X	X	10	10	ENG_STATE VS V_IGK TIA_ST
	2H	SCG	P0230							
	4H	OL	P0230							
	8H									
RLY_MAIN	1H	not switched off	P0560	STD	CC	X	X	10	10	V_IGK ENG_STATE T_AST_ENVD VS
	2H	not switched on	P0560							
	4H									
	8H									
RLY_MAIN_DLY	1H			NO	CC	-	X	10	10	V_IGK ENG_STATE T_AST_ENVD VS
	2H									
	4H	switched on too slow	P0560							
	8H									
SCG_LS_DOWN_1	1H			MEM	LS	X	X	43	43	LSHPWM_DOWN_1 R_IT_LS_DOWN_1_ENVD_H R_IT_LS_DOWN_1_ENVD_L VLS_DOWN_1_ENVD_L
	2H	SCG	P0137							
	4H									
	8H									
SCG_LS_UP_1	1H			MEM	LS	X	X	43	-	VLS_UP_1_ENVD_H VLS_UP_1_ENVD_L R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L
	2H	SCG	P0131							
	4H									
	8H									
SCP_LS_DOWN_1	1H	SCB	P0138	MEM	LS	X	X	43	43	LSHPWM_DOWN_1 R_IT_LS_DOWN_1_ENVD_H R_IT_LS_DOWN_1_ENVD_L VLS_DOWN_1_ENVD_L
	2H									
	4H									
	8H									
SCP_LS_UP_1	1H	SCB	P0132	MEM	LS	X	X	43	-	VLS_UP_1_ENVD_H VLS_UP_1_ENVD_L R_IT_LS_UP_1_ENVD_H R_IT_LS_UP_1_ENVD_L
	2H									
	4H									
	8H									
SEG_AD_ER	1H			STD	CC	X	X	10	10	TCO_ST SEG_AD_MMV_ER_3_ENVD SEG_AD_MMV_ER_1_ENVD TOIL
	2H	ER segment adaptation values at the limit	P0315							
	4H									
	8H									
SHIFT_AFL_LSL_UP_1	1H	Characteristic line shift too lean	P2195	NO	LS	-	-	-	43	TTIP_MES_LS_UP_1_ENVD_H VLS_DOWN_1_ENVD_H LAMB_LS_UP_1_ENVD_H LAMB_DELTA_LAM_ADJ_1_ ENVD_H
	2H									
	4H									
	8H									
SHIFT_AFR_LSL_UP_1	1H	Characteristic line shift too rich	P2196	NO	LS	-	-	-	43	TTIP_MES_LS_UP_1_ENVD_H VLS_DOWN_1_ENVD_H LAMB_LS_UP_1_ENVD_H LAMB_DELTA_LAM_ADJ_1_ ENVD_H
	2H									
	4H									
	8H									
SLV_IVVT_EX_1	1H	SCB	P0080	MPL_STD_INI	CC	X	X	43	43	CAM_EX[1] TOIL IVVTPWM_EX_1_ENVD V_IGK
	2H	SCG	P0079							
	4H	OL	P0079							
	8H									
SLV_IVVT_IN_1	1H	SCB	P0077	MPL_STD	CC	X	X	43	43	CAM_IN[1]

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	2H	SCG	P0076	_INI						TOIL IVVTPWM_IN_1_ENVD V_IGK
	4H	OL	P0076							
	8H									
SOV	1H	SCB	P0448	MPL_STD _INI	CC	X	X	43	43	TAM CPPWM_ENVD DTP_ENVD CP_STATE
	2H	SCG	P0448							
	4H	OL	P0447							
	8H									
STK_LS_UP_1	1H			MEM	LS	X	X	43	-	VLS_UP_1_ENVD_L FAC_LAM_COR_1_ENVD VLS_UP_MMV_MIN_1_ENVD VLS_UP_MMV_MAX_1_ENVD
	2H									
	4H									
	8H	Sensor signal excursion	P0134							
SWT_LS_DOWN_1	1H			NO	LS	X	X	43	43	ENG_STATE CTR_CYCNR_SWT_LS_DOWN LSHPWM_DOWN_1 VLS_DOWN_MMV_MIN_1_ENVD
	2H									
	4H									
	8H	rich/lean switch time error	P0139							
SWT_LS_UP_1	1H			STD	LS	X	X	00	-	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H	rich/lean switch time error	none							
SYN_CAM_EX_1	1H	camshaft synchronisation failure	P0365	STD	CC	X	X	43	43	PSN_AD_CAM_EX TOIL CAM_SP_IVVT_EX CAM_EX_1
	2H									
	4H									
	8H									
SYN_CAM_IN_1	1H	camshaft synchronisation failure	P0340	STD	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									
SYN_CRK_CAM_EX_1	1H	Signal invalid / CAM not valid for CRK synchronisation	P0365	NO	CC	X	X	43	43	PSN_AD_CAM_EX TOIL CAM_SP_IVVT_EX CAM_EX_1
	2H									
	4H									
	8H									
SYN_CRK_CAM_IN_1	1H	Signal invalid / CAM not valid for CRK synchronisation	P0340	NO	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									
T_ES_PLAUS	1H	Engine off timer not resetted or activated	P2610	STD	CC	X	X	43	43	T_ES_DIAG_ENVD_H T_ES_DIAG_ENVD_L T_AST_ENVD T_ES_DIF
	2H									
	4H									
	8H	Engine off timer value inaccurate	P2610							
T_SEG_ER	1H			STD	CC	X	X	00	00	ENG_STATE SEG_AD_MMV_ER_1_ENVD SEG_AD_MMV_ER_3_ENVD DIAG_INST_ENVD
	2H	Missing/Adding 1 tooth or more on engine roughness segment acquisition	none							
	4H									
	8H									
TAM_CAN	1H	TAM signal from CAN not valid	P0071	STD_INI	CC	-	X	00	-	TIA_MES T_AST_ENVD TCO_ST V_IGK
	2H	Short circuit	P0072							
	4H	Open circuit	P0073							
	8H									
TCHA_PROT	1H			NO	CC	-	X	00	-	PWM_WG_ENVD LV_CMD_RCL_OPEN PUT_ENVD TQI_CHA_PROT_ENVD
	2H	Charger speed too high	P0049							
	4H									
	8H									
TCO_EL	1H	SCB + OL	P0118	STD_INI	CC	X	X	43	43	T_AST_ENVD TIA_MES TCO_MES TCO_ST
	2H	SCG	P0117							
	4H									
	8H									
TCO_GRD	1H			MEM	CC	X	X	43	43	T_AST_ENVD TIA_MES TCO_MES
	2H									
	4H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	8H	gradient error	P0119							TCO_ST
TCO_PLAUS	1H			NO	CC	X	X	43	43	T_AST_ENVD TIA_MES TCO_MES TIA_ST
	2H									
	4H									
	8H	plausibility fault	P0125							
TCO_STUCK	1H			NO	CC	X	X	43	43	TCO_ST DIAG_INST_ENVD TCO_MES_DIF_DIAG_TCO_STUCK TCO_SUB_DIF_DIAG_TCO_STUCK
	2H									
	4H									
	8H	stuck signal	P0116							
TCO_STUCK_H	1H			NO	CC	X	X	43	43	TCO_ST DIAG_INST_ENVD TCO_MES_DIF_DIAG_TCO_STUCK TCO_SUB_DIF_DIAG_TCO_STUCK
	2H									
	4H									
	8H	stuck signal	P0116							
TCO_STUCK_RNG	1H			NO	CC	X	X	43	43	TCO_ST DIAG_INST_ENVD TCO_MES_DIF_DIAG_TCO_STUCK TCO_SUB_DIF_DIAG_TCO_STUCK
	2H									
	4H									
	8H	stuck signal	P0116							
TH	1H			STD	CC	X	X	43	43	T_AST_ENVD TAM TCO_ST TIA_ST
	2H									
	4H									
	8H	stuck open	P0128							
TIMEOUT_TCU1	1H			STD	CC	X	X	43	43	TCO_ST T_AST_ENVD V_IGK DIAG_INST_ENVD
	2H									
	4H									
	8H	No message received from TCU	U0101							
TOIL	1H	SCG	P0197	STD_INI	CC	X	X	00	00	TCO_ST TIA_MES TOIL_MDL V_TOIL
	2H	SCB + OL	P0198							
	4H									
	8H									
TOIL_PLAUS_H	1H			STD	CC	X	X	00	00	TCO_ST TOIL TOIL_MES_DIF_DIAG TOIL_MDL_DIF_DIAG
	2H	unplausible high signal	P0196							
	4H									
	8H									
TOIL_PLAUS_L	1H	unplausible low signal	P0196	STD	CC	X	X	00	00	TCO_ST TOIL TOIL_MES_DIF_DIAG TOIL_MDL_DIF_DIAG
	2H									
	4H									
	8H									
TOIL_STUCK	1H			STD	CC	X	X	00	00	TCO_ST TOIL TOIL_MES_DIF_DIAG TOIL_MDL_DIF_DIAG
	2H									
	4H	signal stuck	P0196							
	8H									
TOOTH_OFF_EX_1	1H	Sudden drift "one tooth off"	P0017	STD_INI	CC	X	X	43	43	PSN_AD_CAM_EX_1 TOIL CAM_SP_IVVT_EX CAM_EX[1]
	2H									
	4H									
	8H									
TOOTH_OFF_IN_1	1H	Sudden drift "one tooth off"	P0016	STD_INI	CC	X	X	43	43	PSN_AD_CAM_IN_1 TOIL CAM_SP_IVVT_IN CAM_IN[1]
	2H									
	4H									
	8H									
TPS_EL	1H	SCB + OL	P0123	STD	CC	X	-	-	-	VCC_TPS_DIAG_ENVD V_TPS_1_ENVD TPS TPS_AV_ENVD_H
	2H	SCG	P0122							
	4H									
	8H									
TPS_1	1H	SCB + OL	P0123	MEM	CC	-	X	93	93	VCC_TPS_DIAG_ENVD V_TPS_1_ENVD V_TPS_2_ENVD TPS_AV_1_ENVD
	2H	SCG	P0122							
	4H									
	8H									
TPS_2	1H	SCB	P0223	MEM	CC	-	X	93	93	VCC_TPS_DIAG_ENVD V_TPS_1_ENVD

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	2H	SCG + OL	P0222							V_TPS_2_ENVD TPS_AV_2_ENVD
	4H									
	8H									
TPS_AD	1H	Adaptation conditions exceeded	P0638	NO	CC	-	X	93	93	TPS_AD_STEP MTCPWM_ENVD TPS_LIH_ENVD TPS_AV_ENVD
	2H	Voltage value at LIH-adaptation outside range	P0638							
	4H	Spring-test and CHK_LIH not passed	P0638							
	8H	Adaptation values – lower mechanical stop – outside range	P0638							
TPS_AD_GAIN		Dummy (not supported by HW)	none	STD	CC	-	X	00	00	
TPS_AD_SPR	1H	lower position not reached	P2119	NO	CC	-	X	93	93	TPS_AD_STEP MTCPWM_ENVD TPS_LIH_ENVD TPS_AV_ENVD
	2H	error lower return spring check	P2119							
	4H	upper position not reached	P2119							
	8H	error upper return spring check	P2119							
TPS_PLAUS	1H			MEM	CC	X	X	10	00	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H	Unplausibility caused by faulty throttle sensor	none							
TPS_MAF_1	1H			NO	CC	-	X	93	93	VCC_TPS_DIAG_ENVD MAP_ENVD TPS_AV_1_ENVD TPS_AV_2_ENVD
	2H									
	4H	TPS1/MAF is not plausible	P0121							
	8H									
TPS_MAF_2	1H			NO	CC	-	X	93	93	VCC_TPS_DIAG_ENVD MAP_ENVD TPS_AV_1_ENVD TPS_AV_2_ENVD
	2H									
	4H									
	8H	TPS2/MAF is not plausible	P0221							
TPS_ST_CHK	1H	Error in spring check	P2119	NO	CC	-	X	93	93	TPS_AD_STEP MTCPWM_ENVD TPS_LIH_ENVD TPS_AV_ENVD
	2H	Error in limp-home-check	P2119							
	4H									
	8H									
TPS_VCC	1H	SCB	P0653	MEM	CC	-	X	93	93	VCC_TPS_DIAG_ENVD VB_MMV V_TPS_2_ENVD V_TPS_1_ENVD
	2H	SCG	P0652							
	4H	Sensor or wiring harness problem	P0652							
	8H									
TQ_LIM_INTV	1H	actual or requested torque higher than allowed limit	none	STD	CC	X	X	00	00	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H									
TQI_MON	1H			STD	CC	-	X	93	93	ENVD_0_MON ENVD_1_MON ENVD_2_MON ENVD_3_MON
	2H									
	4H	torque monitoring	P0605							
	8H									
TRIP_IVVT_EX_1	1H	trip IVVT	P0014	STD	CC	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_EX CAM_EX[1] CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD
	2H									
	4H									
	8H									
TRIP_IVVT_IN_1	1H	trip IVVT	P0011	STD	CC	X	X	43	43	DIAG_INST_ENVD CAM_SP_IVVT_IN CAM_IN[1] CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD
	2H									
	4H									
	8H									
TTIP_MES_LSH_UP_1	1H	sensor temperature out of range	P0135	NO	LSH	-	-	-	43	STATE_SYM_OBD_LSL_LSH_UP ERR_SYM_LSL_UP_IF[1] ERR_SYM_OFS_LSL_UP[1] TTIP_MES_LS_UP_1_ENVD_H
	2H	sensor readiness delayed	none							
	4H	invalid sensor temperature	P0135							
	8H									
VAP_LEAK_1	1H	VAP_LEAK_1	P0456	STC	EVAP	X	X	43	43	TAM FAC_DIAM_DIAGCP
	2H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	4H									FAC_DIAGCP_MDL_ESTIM_ ENVD
	8H									FAC_DIAGCP_VOL
VAP_LEAK_1_MIL	1H	VAP_LEAK_1	P0456	STC	EVAP	X	X	43	43	TAM FAC_DIAM_DIAGCP FAC_DIAGCP_MDL_ESTIM_ ENVD FAC_DIAGCP_VOL
	2H									
	4H									
	8H									
VAP_LEAK_10	1H			STC	EVAP	X	X	43	43	TAM DTP_DIF_ACT_ENVD DTP_ENVD FAC_DIAGCP_VOL
	2H									
	4H	VAP_LEAK_10	P0455							
	8H									
VAP_LEAK_2	1H	VAP_LEAK_2	P0442	STC	EVAP	X	X	43	43	TAM FAC_DIAM_DIAGCP FAC_DIAGCP_MDL_ESTIM_ENV D FAC_DIAGCP_VOL
	2H									
	4H									
	8H									
VB	1H	System Voltage high	P0563	STD	CC	X	X	43	43	V_IGK ENG_STATE T_AST_ENVD VS
	2H	System Voltage low	P0562							
	4H									
	8H									
VB_OC	1H			MEM	CC	X	X	10	10	V_IGK ENG_STATE T_AST_ENVD VS
	2H									
	4H	OL	P0560							
	8H									
VCC_PVS_1	1H	SCB	P0699	MEM	CC	-	X	93	93	LV_ERR_VCC_PVS_2 VB_MMV VCC_PVS_1_DIAG_ENVD V_PVS_1_ENVD
	2H	SCG	P0698							
	4H	Sensor or wiring harness problem	P0698							
	8H									
VCC_PVS_2	1H	SCB	P0643	for ISA: STD for ETC: MEM	CC	X	X	93	93	for ISA: MAP_MES_BAS_ENVD VB_MMV VCC_PVS_2_DIAG_ENVD V_IGK for ETC: MAP_MES_BAS_ENVD VB_MMV VCC_PVS_2_DIAG_ENVD V_PVS_2_ENVD
	2H	SCG	P0642							
	4H	Sensor or wiring harness problem	P0642							
	8H									
VCC_SENS_SUB	1H	SCB	P06A5	STD	CC	X	X	43	43	for 690: VCC_SENS_SUB_DIAG_ENV D VB_MMV V_PSP_MES_ENVD_H V_PSP_MES_ENVD_L for 691 TCI: VCC_SENS_SUB_DIAG_ENV D VB_MMV DTP_ENVD VP_PUT_ENVD for 692: VCC_SENS_SUB_DIAG_ENV D VB_MMV DTP_ENVD V_PORT_ENVD
	2H	SCG	P06A4							
	4H	Sensor or wiring harness problem	P06A4							
	8H									
VCC_TPS	1H	SCP	P0653	STD	CC	X	-	-	-	VCC_TPS_DIAG_ENVD VB_MMV VCC_SENS_SUB_DIAG_ENVD V_TPS_1_EN V_PSP_MES_ENVD_H VD
	2H	SCG	P0652							
	4H	Sensor or wiring harness problem	P0652							
	8H									
VIM	1H	SCB	P0662	MPL_STD_INI	CC	X	X	00	43	TIA TCO_ST LV_VIM_SP V_IGK
	2H	SCG	P0661							
	4H	OL	P0661							
	8H									
VIM_AD	1H			NO	CC	X	X	00	00	TIA TCO_ST MAP_MES_BAS_ENVD V_VIM_ENVD
	2H									
	4H									
	8H	Adaptation failure	none							
VIM_FB_EL	1H	SCB	none	STD_INI	CC	X	X	00	00	TIA TCO_ST MAP_MES_BAS_ENVD
	2H	SCG / OL	none							
	4H									

Failure Flag LV_ERR_xx		Error symptom ERR_SYM_xx	DTC-Code	ABC type	CARB_xx	690	691	691 TCI	692 SULEV	Enviromental data
	8H									V_VIM_ENVD
VIM_MEC_LONG	1H	Mechanical failure (upper stop not reached)	none	NO	CC	X	X	00	00	TIA TCO_ST MAP_MES_BAS_ENVD V_VIM_ENVD
	2H									
	4H									
	8H									
VIM_MEC_SHO	1H	Mechanical failure (lower stop not reached)	none	NO	CC	X	X	00	00	TIA TCO_ST MAP_MES_BAS_ENVD V_VIM_ENVD
	2H									
	4H									
	8H									
VIN_DIAG	1H	VIN not programmed	P0630	STD	CC	X	X	93	93	N_32 N_32 N_32 N_32
	2H									
	4H									
	8H									
VLS_DOWN_DIF_1	1H	AF too rich	P2097	NO	LS	-	-	-	43	VLS_DOWN_1_ENVD_H VLS_UP_DIAG_1_ENVD_H LAMB_DELTA_I_LAM_ADJ_1_ENVD_H VLS_DIF_LAM_ADJ_1_ENVD_H
	2H	AF too lean	P2096							
	4H									
	8H									
VS	1H			STD	CC	-	-	43	43	TCO_ST VS V_MIN_MWSS_DIAG V_MAX_MWSS_DIAG
	2H	Plausibility fault	P2159 P0501							
	4H									
	8H									
WG	1H	SCB	P0246	MPL_STD_INI	CC	-	X	00	-	PWM_WG_ENVD MAP_MES_ENVD PUT_ENVD VB_MMV
	2H	SCG	P0245							
	4H	OL	P0246							
	8H									

A.2 Anti-bounce algorithms

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_PREV_XX	-	0...FH	0...15	1	[-]
Previous value of ERR_SYM_XX when the diagnosis conditions are fulfilled					
CTR_ABC_XX	V/O/S	0...FFH	0...255	1	[-]
anti bounce counter of diagnosis XX					
CTR_ABC_END_DIAG_XX	V	0...FFH	0...255	1	[-]
Counter for end of diagnosis XX generation					
LV_END_DIAG_XX	V	0...1H	0...1	1	[-]
Diagnostic done completely at least one time					
LV_CDN_DIAG_XX	V	0...1H	0...1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
ERR_SYM_XX	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	0	[-]
For each symptom : status of failure (set to 1 when failure symptom detected)					

Remark : The present failure and the anti-bounce counter may be saved or not (see 'Calculation of the Anti-bounce')

Input data:

LV_IGK	LV_DC	LV_ABC_INH	

Import actions:

#IF NC_ERR_DET_UPD = 1
ACTION_ERRM_UpdErrDet (IN<XX>)
This action indicates each occurrence of the detected error
#ENDIF

Export actions:

Name
ACTION_ERRM_FilterSymptom(IN< XX >, IN< lv_cdn_diag_XX >, IN< err_sym_XX >, IN< C_ABC_INC_XX >, IN< C_ABC_DEC_XX >, IN< C_ABC_MAX_XX >, OUT< LV_ERR_XX >)
This action computes the elementary anti-bounce filter for one failure treatment and returns filter result
ACTION_ERRM_NoFilterSymptom(IN< XX >, IN< lv_cdn_diag_XX >, IN< err_sym_XX >, IN< lv_err_set_XX >, IN< lv_err_reset_XX >, IN< lv_end_diag_XX >, OUT< LV_ERR_XX >)
This action computes the elementary treatment case no filtering is used
ACTION_ERRM_AbcFilterSymptomEnd(IN< XX >, IN< C_ABC_MAX_XX>, OUT< LV_ERR_XX >)
This action erases diagnostic failure and sets the end of diagnostic in case of antibounce filter type
ACTION_ERRM_NoFilterSymptomEnd(IN< XX >, OUT< LV_ERR_XX >)
This action erases diagnostic failure and sets the end of diagnostic in case of no filter
ACTION_ERRM_AbcFilterReset(IN< XX >, OUT< LV_ERR_XX >)
This action resets data filter in case of antibounce filter type
ACTION_ERRM_NoFilterReset(IN< XX >, OUT< LV_ERR_XX >)
This action resets data filter in case of no filter
ACTION_ERRM_GetLvErr(IN< XX >, OUT< LV_ERR_XX >)
This action is used to get LV_ERR_XX value
ACTION_ERRM_GetLvEndDiag(IN< XX >, OUT< LV_END_DIAG_XX >)
This action is used to get LV_END_DIAG_XX value
ACTION_ERRM_GetLvCdnDiag(IN< XX >, OUT< LV_CDN_DIAG_XX >)
This action is used to get LV_CDN_DIAG_XX value
ACTION_ERRM_GetErrSym(IN< XX >, OUT< ERR_SYM_XX >)
This action is used to get ERR_SYM_XX value

ERR_SYM_XX is defined like output data for each diagnosis as following :

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
ERR_SYM_XX	O/V	0..0FH	All combination	1	-
		0H	NO_SYM		
		1H	SYM_0		
		2H	SYM_1		
		4H	SYM_2		
		8H	SYM_3		
		FH	BENCH_MODE		
Symptom : failure without filtering of diagnosis XX					

It is possible to combine several symptoms.

FUNCTION DESCRIPTION:

General information:

The filtering algorithm (anti-bounce or statistic) is used usually for the simple diagnosis (e.g. : usually OBD1 diagnosis). If there is another need, diagnosis manages itself the filtering and the end of diagnosis (e.g. : usually OBD2 diagnosis).

The filtering is used to filter the detected failure. The failure becomes present after filtering.

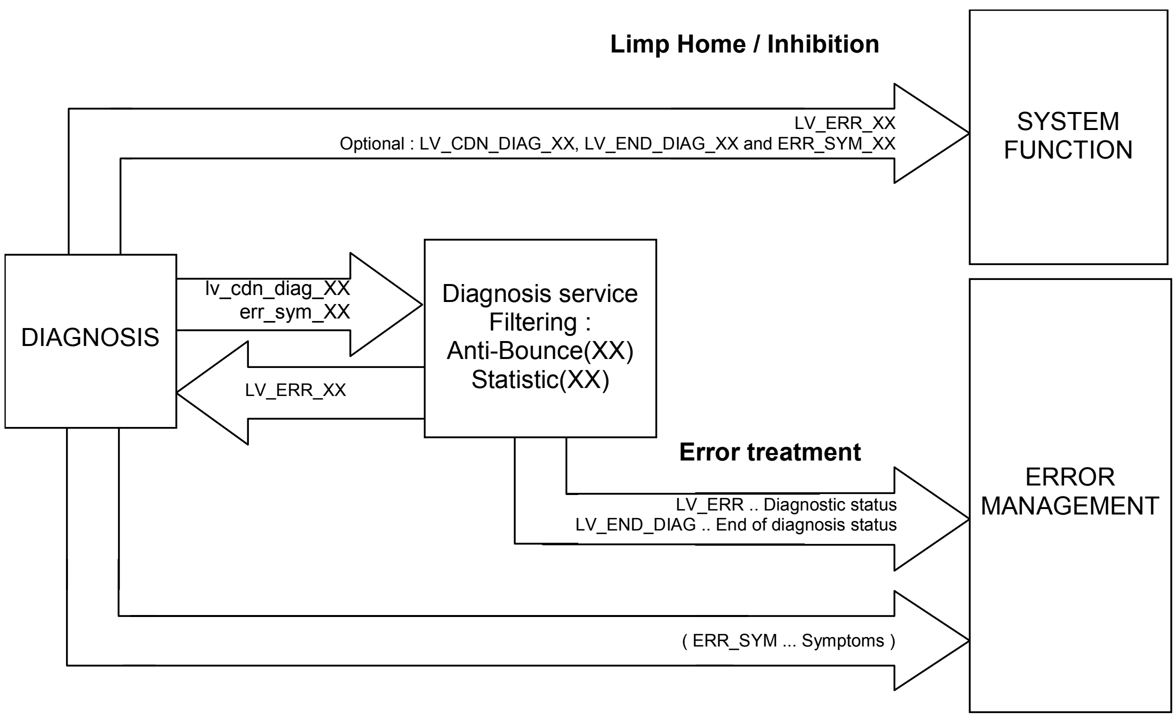
The end of diagnosis definition is realised after a time defined as the minimum time to detect a present failure.

The filtering algorithm (anti-bounce or statistic) uses one counter to check failure detection threshold and one counter to manage the end of diagnosis (Diagnostic result available).

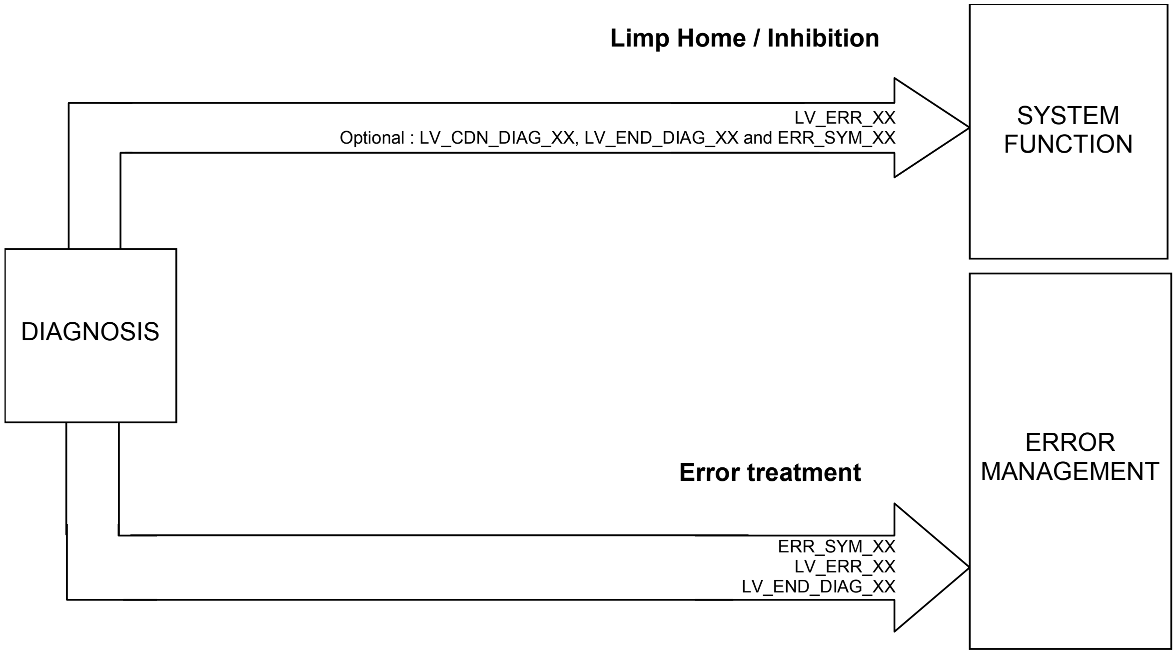
Filter choice selection : according to the diagnostic filtering configuration the filtering algorithm used is the anti-bounce filtering (NLC_ABC_STC_XX = 0) or the standard statistical filtering (NLC_ABC_STC_XX=1)

Signal flow diagram :

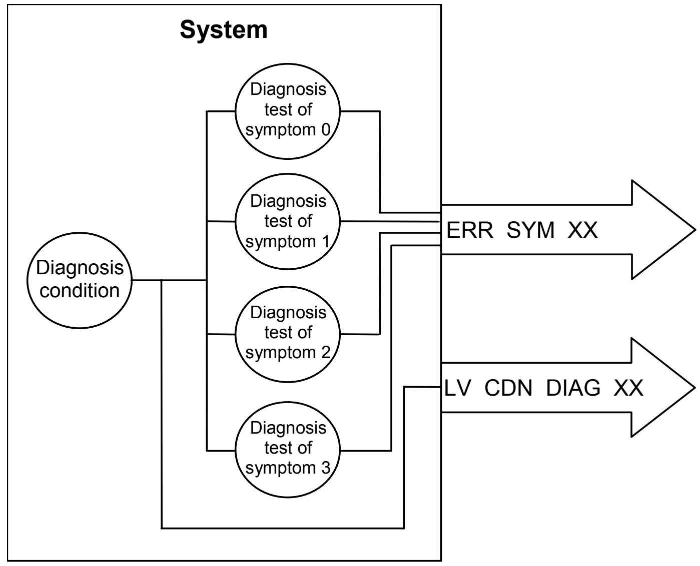
Diagnosis with an anti-bounce algorithm :



Diagnosis without anti-bounce algorithm :



The symptoms are evaluated when diagnosis conditions are fulfilled :



A.2.1 Initialisation of anti-bounce, statistical and multi-conditions filters

This paragraph is dedicated to the initialisation of main data, at the following events:

- ECU reset
- LV_IGK 0 → 1 transition before the end of the power-latch phase
- LV_DC 1 → 0 transition

A.2.1.1 At ECU reset

Management of NVMY

```
If      NVMY not corrupted and not first ECU power-up
Then
        CTR_ABC_ XX is restored from NVMY
        LV_ERR_XX is restored from NVMY
        ERR_SYM_XX is restored from NVMY
Else
        CTR_ABC_ XX = 0
        LV_ERR_XX = 0
        ERR_SYM_XX = 0
Endif

LV_CDN_DIAG_XX = 0
CTR_ABC_END_DIAG_XX = 0
LV_END_DIAG_XX = 0

If      NLC_ABC_NOT_SAVE_XX = 1
Then
        { case of STD_INI, MEM and STC configurations }
        CTR_ABC_XX = 0
        LV_ERR_XX = 0
        ERR_SYM_XX = 0
Endif
```

Configuration evaluation

```
If      NLC_ABC_INI_VALUE_XX = 1
Then
        { case of MEM_INI configuration }
        If      LV_ERR_XX = 1
        Then
                CTR_ABC_XX = C_ABC_MAX_XX - C_ABC_INC_XX
        Else
                CTR_ABC_XX = 0
        Endif
        LV_ERR_XX = 0
        ERR_SYM_XX = 0
Endif
```

A.2.1.2 At LV_IGK 0 → 1 transition, without ECU reset

```
If      NLC_ABC_NOT_SAVE_XX = 1 and NLC_ABC_INI_IGK_XX = 1
Then
        CTR_ABC_ XX = 0
        LV_ERR_XX = 0
        ERR_SYM_XX = 0
Endif

LV_CDN_DIAG_XX = 0

If      NLC_ABC_NOT_SAVE_XX = 0
        and NLC_ABC_INI_VALUE_XX = 1
        and NLC_ABC_INI_IGK_XX = 1
Then
        { case of MEM_INI configuration }
        If      LV_ERR_XX = 1
        Then
                CTR_ABC_XX = C_ABC_MAX_XX - C_ABC_INC_XX
        Else
                CTR_ABC_XX = 0
        Endif
        LV_ERR_XX = 0
        ERR_SYM_XX = 0
Endif

If      NLC_ABC_INI_DC_END_DIAG = 0
Then
        CTR_ABC_END_DIAG_XX = 0
        LV_END_DIAG_XX = 0
Endif
```

A.2.1.3 At LV_DC 1 → 0 transition

```
If      NLC_ABC_INI_DC_END_DIAG = 1
Then
        CTR_ABC_END_DIAG_XX = 0
        LV_END_DIAG_XX = 0
Endif

If      LV_ERR_XX = 0 and NLC_ABC_INI_VALUE_XX = 0
Then
        { all cases excepted MEM_INI configuration }
        CTR_ABC_XX = 0
        ERR_SYM_XX = 0
Endif
```

A.2.2 Recurrence

The recurrence is always managed by the diagnosis recurrence.

A.2.3 Filtering algorithm description

A.2.3.1 Description:

Syntax : ACTION_ERRM_FilterSymptom(IN<XX>, IN<lv_cdn_diag_XX>, IN< err_sym_xx>, IN<C_ABC_INC_XX>, IN<C_ABC_DEC_XX>,IN<C_ABC_MAX_XX>, OUT<LV_ERR_XX>)

Parameter(in) :	XX	Type: index to identify the Diagnostic
	lv_cdn_diag_XX	Type: Diagnostic condition
	err_sym_XX	Status of failure
	C_ABC_INC_XX	Anti-bounce counter increment
	C_ABC_DEC_XX	Anti-bounce counter decrement
	C_ABC_MAX_XX	Maximum value of anti-bounce counter
Parameter(out) :	LV_ERR_XX	Type: Boolean (=1 when failure)

Short Description : This action returns the filter result on symptoms detected at each diagnostic recurrence

A.2.3.2 Antibounce algorithm filtering type

A.2.3.2.1 Calculation of the anti-bounce counter

Description:

Anti-bounce operation:

If the system failure is detected and the diagnosis conditions are fulfilled (LV_CDN_DIAG_XX = 1) then the failure is detected (ERR_SYM_XX ≠ 0) and the anti-bounce counter is incremented. In this case an action is called to manage some functionality (eg : to visualise the sporadic error of all diagnosis using this anti-bounce algorithm).

If the system failure isn't detected and the diagnosis conditions are fulfilled (LV_CDN_DIAG_XX = 1) then the failure isn't detected (ERR_SYM_XX = 0) and the anti-bounce counter is decremented.

If the diagnosis conditions aren't fulfilled (LV_CDN_DIAG_XX = 0) then the detected failure is unchanged ($ERR\_SYM\_XX_{(N)} = ERR\_SYM\_XX_{(N-1)}$) and the anti-bounce counter is unchanged.

The present failure is set when the anti-bounce counter has reached its MAX (unequal to zero).

The configuration allows:

- to save or not the present failure and the anti-bounce counter
- to initialise or not the present failure and the anti-bounce counter at transition LV_IGK 0 → 1
- to manage the decrement of the anti-bounce counter or not (the anti-bounce counter remains at the maximum value)
- to calibrate or not the decrement of the anti-bounce counter.

Anti-bounce bench mode:

To make easier the development phase, it may be interesting to force the LV_ERR_XX flag (present failure) and the associated symptoms in order to free the diagnosis and to test only the error management and/or the application incidence (limp home or inhibition) associated to the present failure. This mode is active only if the LC_ABC_BENCH and NLC_BENCH_MODE flags are set.

To set LV_ERR_XX to one and ERR_SYM_XX = FH (all detected error symptoms) :

Set the C_ABC_INC_XX calibration to the maximum value (FFH).

To set LV_ERR_XX to zero and ERR_SYM_XX = 0 (no detected error symptom) :

Set the C_ABC_INC_XX calibration to zero.

Note:

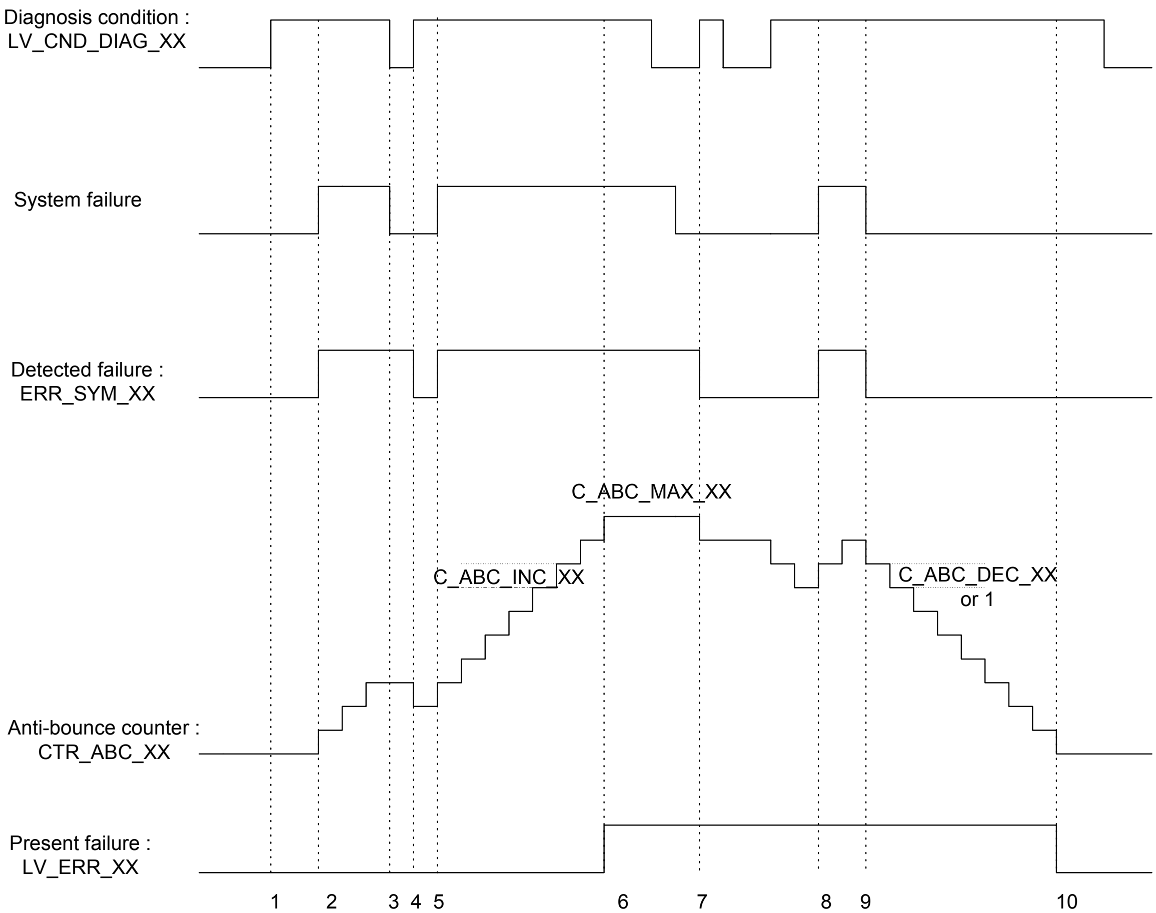
In this mode, the diagnosis condition (LV_CDN_DIAG_XX) and the end of diagnosis (LV_END_DIAG_XX) are set to 1 (LV_CDN_DIAG_XX = 1 means the ERR_SYM_XX is valid and LV_END_DIAG_XX = 1 means the LV_ERR_XX flag is valid).

Anti-bounce inhibition:

For specific needs, the anti-bounce filtering calculation (anti-bounce operation and anti-bounce bench mode) may be inhibited, with LV_ABC_INH inhibition flag.

Signal flow diagram:

1 -The anti-bounce counter may be decremented (NLC_ABC_NOT_DEC_XX = 0):



1: The diagnosis conditions are fulfilled but the failure isn't detected
LV_CDN_DIAG_XX = 1,
ERR_SYM_XX = 0.

➔ CTR_ABC_XX remains at the value 0.
LV_ERR_XX = 0.

2: The diagnosis conditions are fulfilled and the failure is detected.
LV_CDN_DIAG_XX = 1,
ERR_SYM_XX <> 0.

➔ CTR_ABC_XX = 0 + C_ABC_INC_XX
LV_ERR_XX = 0.

3: The diagnosis conditions aren't fulfilled
and the failure remains detected. → CTR_ABC_XX is frozen
LV_CDN_DIAG_XX = 0, LV_ERR_XX = 0.
ERR_SYM_XX <> 0.

.

4: The diagnosis conditions are fulfilled again
and the failure is no more detected. → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX – decrement
ERR_SYM_XX = 0. LV_ERR_XX = 0.

5: The diagnosis conditions are still fulfilled
and the failure is detected again → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX + C_ABC_INC_XX
ERR_SYM_XX <> 0. LV_ERR_XX = 0.

6: The diagnosis conditions are still fulfilled,
the failure is still detected
and the failure becomes present. → CTR_ABC_XX = C_ABC_MAX_XX
LV_CDN_DIAG_XX = 1, LV_ERR_XX = 1.
ERR_SYM_XX <> 0.

7: The diagnosis conditions are fulfilled,
the failure isn't detected
and the failure remains present. → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX – decrement
ERR_SYM_XX = 0. LV_ERR_XX = 1.

8: The diagnosis conditions are still fulfilled,
the failure is detected again
and the failure remains present. → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX + C_ABC_INC_XX
ERR_SYM_XX <> 0. LV_ERR_XX = 1.

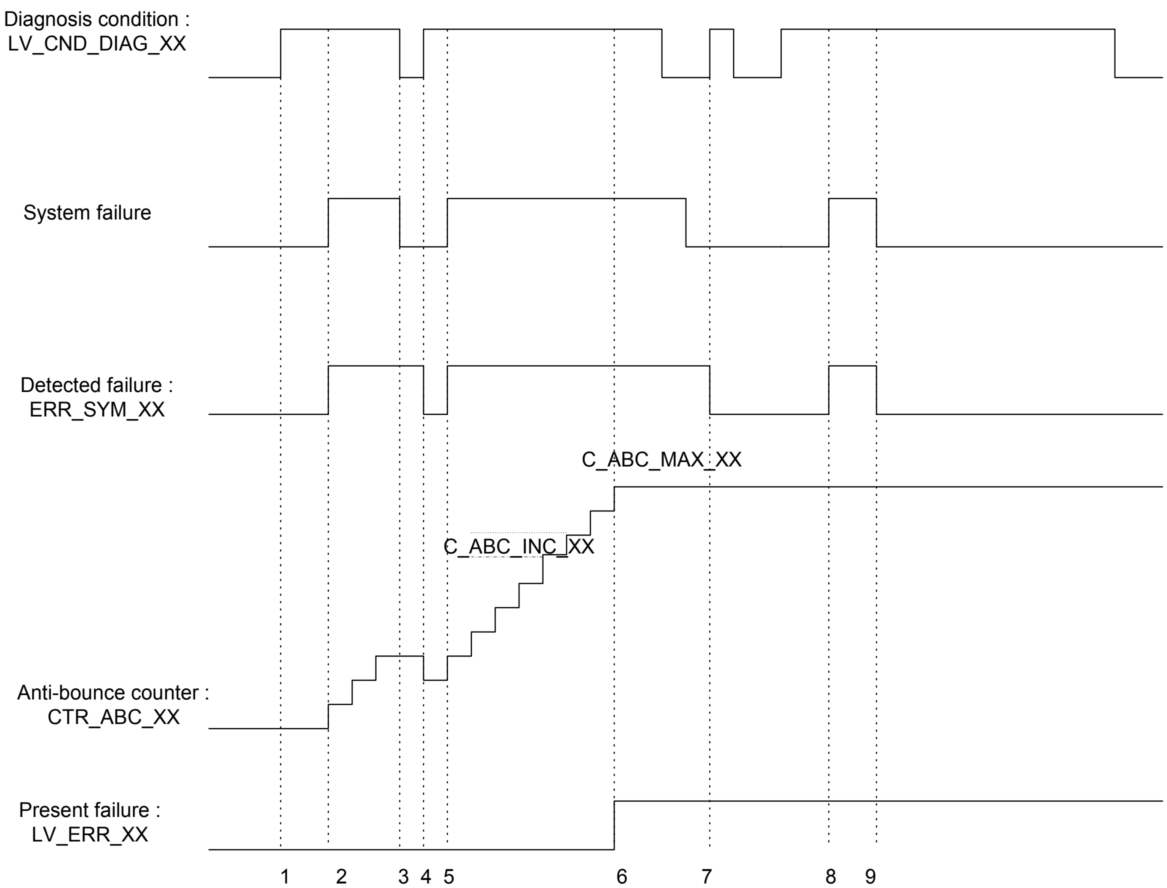
9: The diagnosis conditions are still fulfilled,
the failure isn't detected
and the failure remains present. ➔ CTR_ABC_XX = CTR_ABC_XX – decrement
LV_CDN_DIAG_XX = 1, LV_ERR_XX = 1.
ERR_SYM_XX = 0.

10: The diagnosis conditions are still fulfilled,
the failure isn't detected
and the failure is no more present. ➔ CTR_ABC_XX = 0
LV_CDN_DIAG_XX = 1, LV_ERR_XX = 0.
ERR_SYM_XX = 0.

Remark

- Decrement value = 1 if NLC_ABC_CAL_DEC_XX bit = 0
- Decrement value = C_ABC_DEC_XX if NLC_ABC_CAL_DEC_XX bit = 1.

2-The anti-bounce counter isn't decremented (NLC_ABC_NOT_DEC_XX = 1):



1: The diagnosis conditions are fulfilled but the failure isn't detected
LV_CDN_DIAG_XX = 1,
ERR_SYM_XX = 0.
➔ CTR_ABC_XX remains at the value 0.
LV_ERR_XX = 0.

2: The diagnosis conditions are fulfilled and the failure is detected.
LV_CDN_DIAG_XX = 1,
ERR_SYM_XX <> 0.
➔ CTR_ABC_XX = 0 + C_ABC_INC_XX
LV_ERR_XX = 0.

3: The diagnosis conditions aren't fulfilled
and the failure remains detected. → CTR_ABC_XX is frozen
LV_CDN_DIAG_XX = 0, LV_ERR_XX = 0.
ERR_SYM_XX <> 0.

4: The diagnosis conditions are fulfilled again
and the failure is no more detected. → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX – decrement
ERR_SYM_XX = 0. LV_ERR_XX = 0.

5: The diagnosis conditions are still fulfilled
and the failure is detected again → CTR_ABC_XX =
LV_CDN_DIAG_XX = 1, CTR_ABC_XX + C_ABC_INC_XX
ERR_SYM_XX <> 0. LV_ERR_XX = 0.

6: The diagnosis conditions are still fulfilled,
the failure is still detected
and the failure becomes present. → CTR_ABC_XX = C_ABC_MAX_XX
LV_CDN_DIAG_XX = 1, LV_ERR_XX = 1.
ERR_SYM_XX <> 0.

7: The diagnosis conditions are fulfilled,
the failure isn't detected
and the failure remains present. → CTR_ABC_XX = C_ABC_MAX_XX
LV_CDN_DIAG_XX = 1, (counter not decrement)
ERR_SYM_XX = 0. LV_ERR_XX = 1.

.

8: The diagnosis conditions are still fulfilled,
the failure is detected again
and the failure remains present. → CTR_ABC_XX = C_ABC_MAX_XX
LV_CDN_DIAG_XX = 1, LV_ERR_XX = 1.
ERR_SYM_XX <> 0.

.

.9: The diagnosis conditions are still fulfilled,
the failure isn't detected
and the failure remains present. ➔ CTR_ABC_XX = C_ABC_MAX_XX
LV_CDN_DIAG_XX = 1, (counter not decrement)
ERR_SYM_XX = 0. LV_ERR_XX = 1.

Remark

- Decrement value = 1 if NLC_ABC_CAL_DEC_XX bit = 0
- Decrement value = C_ABC_DEC_XX if NLC_ABC_CAL_DEC_XX bit = 1.

Application conditions:

Initialization: See "Initialisation of anti-bounce, statistical and multi-conditions filters" paragraph

Recurrence: The recurrence is managed by the diagnosis.

Activation: The activation is managed by the diagnosis for the anti-bounce operation or the anti-bounce bench mode.

Deactivation: LV_ABC_INH = 1

Formula section:

Anti-bounce bench mode :

If(1) (LC_ABC_BENCH = 1 **and** NLC_BENCH_MODE=1) **and** ((C_ABC_INC_XX = 0) **or** (C_ABC_INC_XX = FFH))

Then(1)

If(2) C_ABC_INC_XX = 0

Then(2)

ERR_SYM_XX = 0

LV_ERR_XX = 0

CTR_ABC_XX = 0

Else(2) **If(3)** C_ABC_INC_XX = FFH

Then(3)

ERR_SYM_XX = 0FH

LV_ERR_XX = 1

CTR_ABC_XX = C_ABC_MAX_XX

Endif(3)

Endif(2)

Endif(1)

Anti-bounce operation (not in bench mode) :

If(1a) (LV_CDN_DIAG_XX = 1) **and not**((LC_ABC_BENCH = 1 **and** NLC_BENCH_MODE=1) **and** ((C_ABC_INC_XX = 0) **or** (C_ABC_INC_XX = FFH)))

Then(1a)

If(1b) C_ABC_INC_XX = 0 *{ component XX not available }*

Then(1b)

{ the diagnosis result is forced to 'no error' }

ERR_SYM_XX = 0

LV_ERR_XX = 0

CTR_ABC_XX = 0

Else(1b)

If(2) ERR_SYM_XX <> 0

Then(2)

{ at each detected error, a treatment shall be done }

If(3a) NC_ERR_DET_UPD = 1

and LC_ERR_DET_UPD=1

Then(3a)

ACTION_ERRM_UpdErrDet(IN<XX>, SYNCHRONIZATION<CALL>)

```
                    Endif(3a)
                    If(3b)      CTR_ABC_XX <> C_ABC_MAX_XX
                    Then(3b)
                                CTR_ABC_XX = CTR_ABC_XX + C_ABC_INC_XX
                                If(4)       CTR_ABC_XX >= C_ABC_MAX_XX
                                Then(4)
                                            LV_ERR_XX = 1
                                            CTR_ABC_XX = C_ABC_MAX_XX
                                Endif(4)
                    Endif(3b)
            Else(2)
                    If(3)       (NLC_ABC_NOT_DEC_XX = 0 OR LV_ERR_XX = 0)
                    Then(3)
                                If(4)       CTR_ABC_XX <> 0
                                Then(4)     If(5)       NLC_ABC_CAL_DEC_XX = 0
                                            Then(5)
                                                        (without calibratable decrement)
                                                        CTR_ABC_XX = CTR_ABC_XX – 1
                                                        If(6)       CTR_ABC_XX = 0
                                                        Then(6)
                                                                    LV_ERR_XX = 0
                                                        Endif(6)
                                            Else(5)
                                                        (with calibratable decrement)
                                                        CTR_ABC_XX = ( CTR_ABC_XX –
                                                        C_ABC_DEC_XX )
                                                        If(6)       CTR_ABC_XX <= 0
                                                        Then(6)
                                                                    LV_ERR_XX = 0
                                                                    CTR_ABC_XX = 0
                                                        Endif(6)
                                            Endif(5)
                                Endif(4)
                    Endif(3)
            Endif(2)
        Endif(1b)
    Endif(1a)
```

A.2.3.2.2 Calculation of the end of diagnosis

Description:

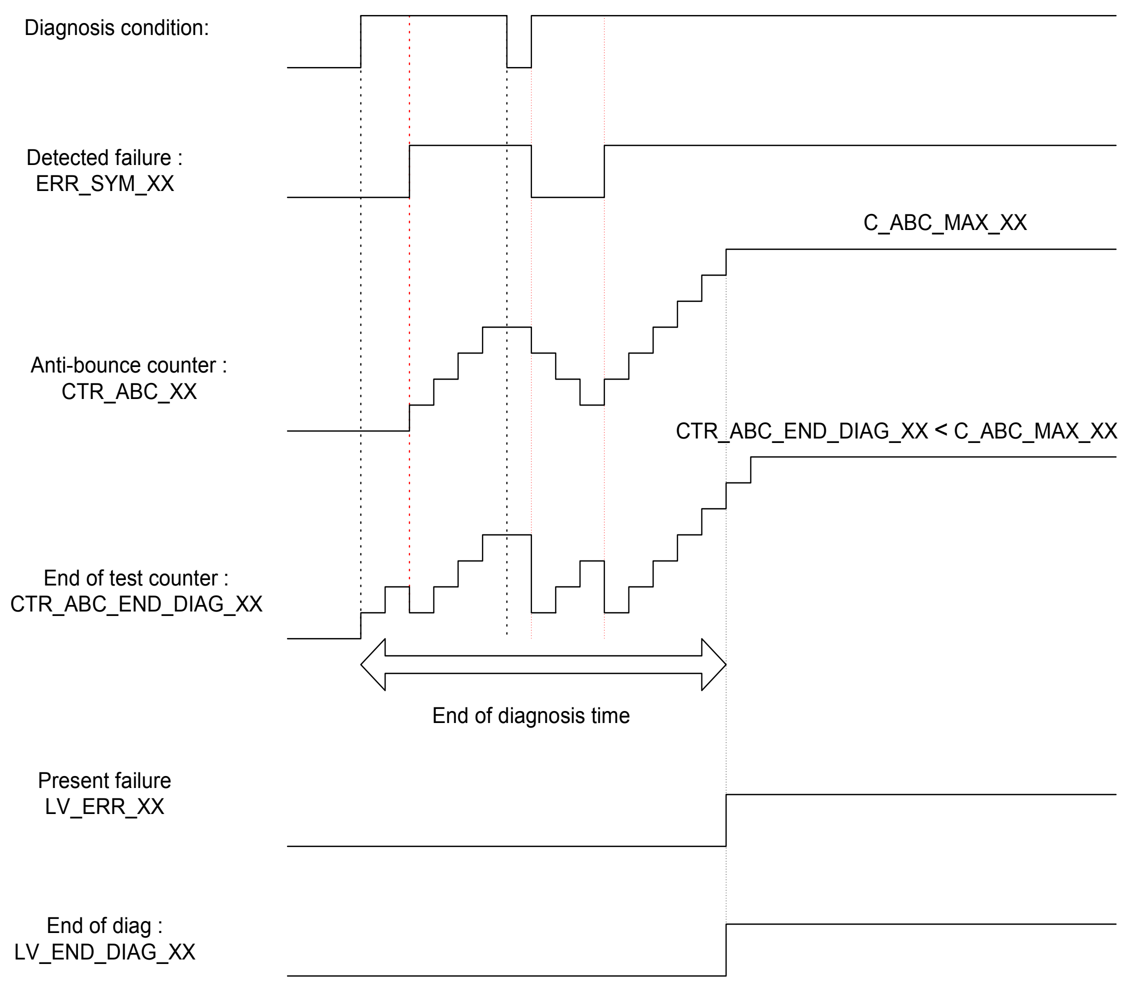
The end of diagnosis definition is set after a time defined like being the time necessary to evaluate the failure (present or not).

Signal flow diagram:

End of diagnosis algorithm illustration in standard case :

The end of diagnosis counter is reset at each transition of the detected failure.

When the failure becomes present : the end of diagnosis is set.

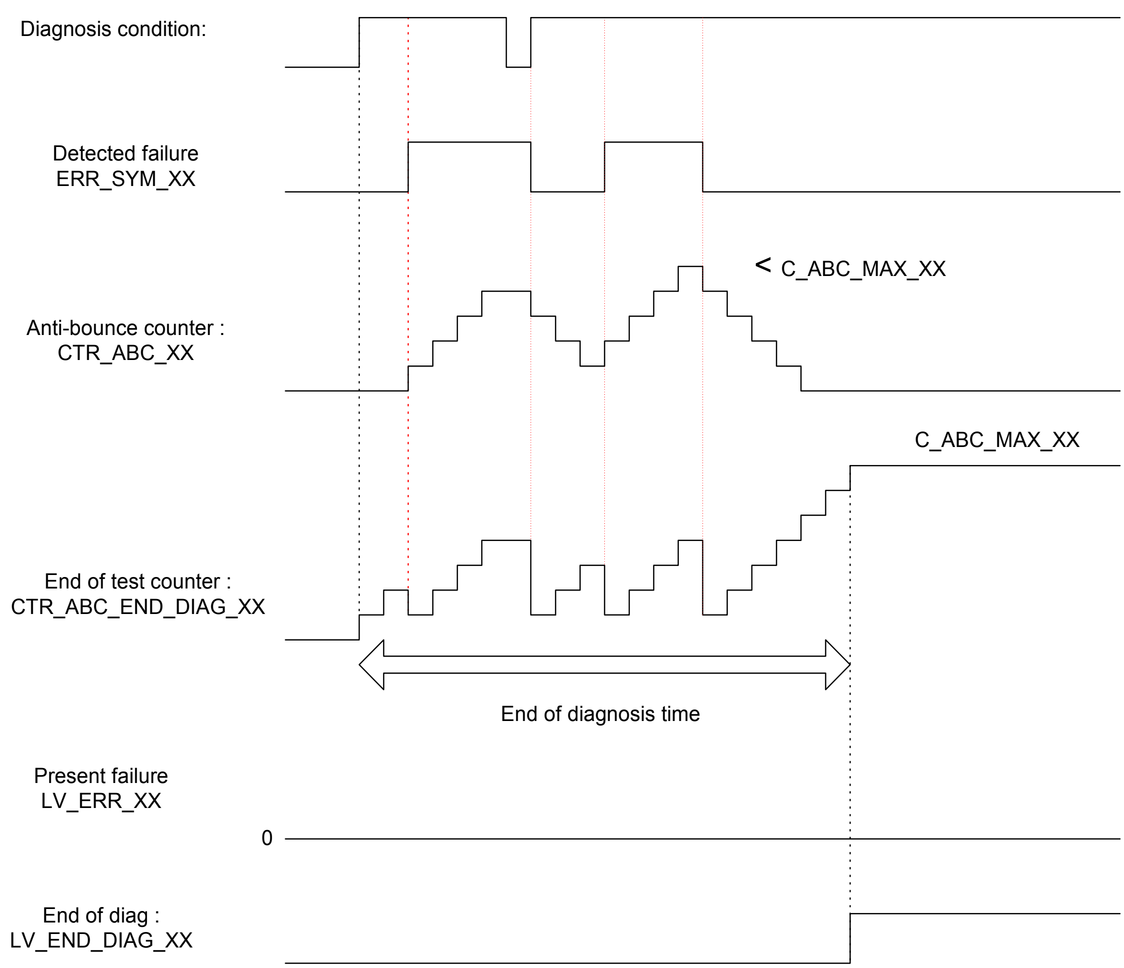


The detected failure will disappear and the anti-bounce counter won't reach the maximum value then the failure won't become present.

The end of diagnosis counter is reset at each transition of the detected failure.

When the detected failure is stable : the end of diagnosis counter is incremented until the maximum value while the diagnosis conditions are fulfilled.

When the end of diagnosis counter has reached the maximum value : the end of diagnosis is set.

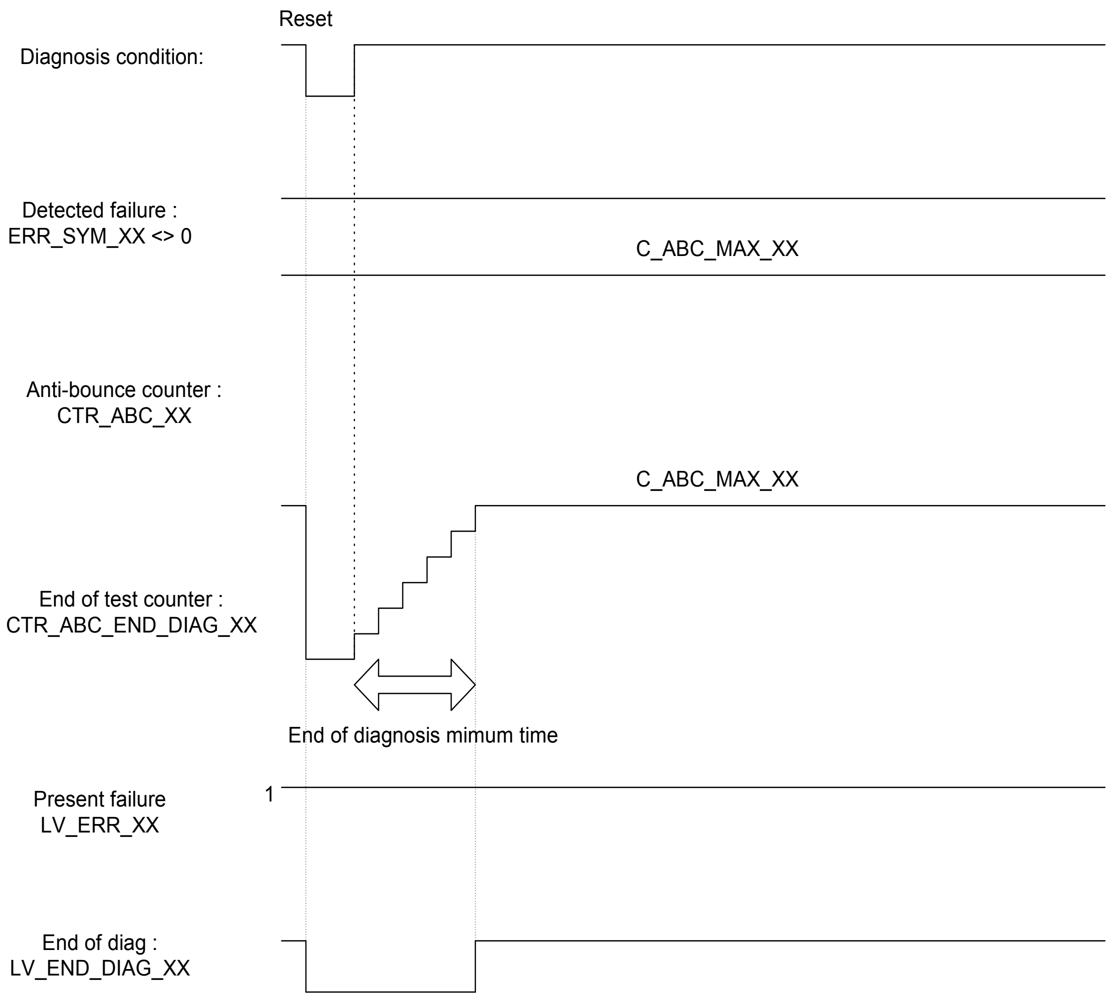


End of diagnosis algorithm illustration in case of reset event :

After the reset event, the detected failure remains present (stable) : the end of diagnosis counter is incremented until the maximum value while the diagnosis conditions are fulfilled.

When the end of diagnosis counter has reached the maximum value : the end of diagnosis is set.

The failure has been present and remains present after reset.



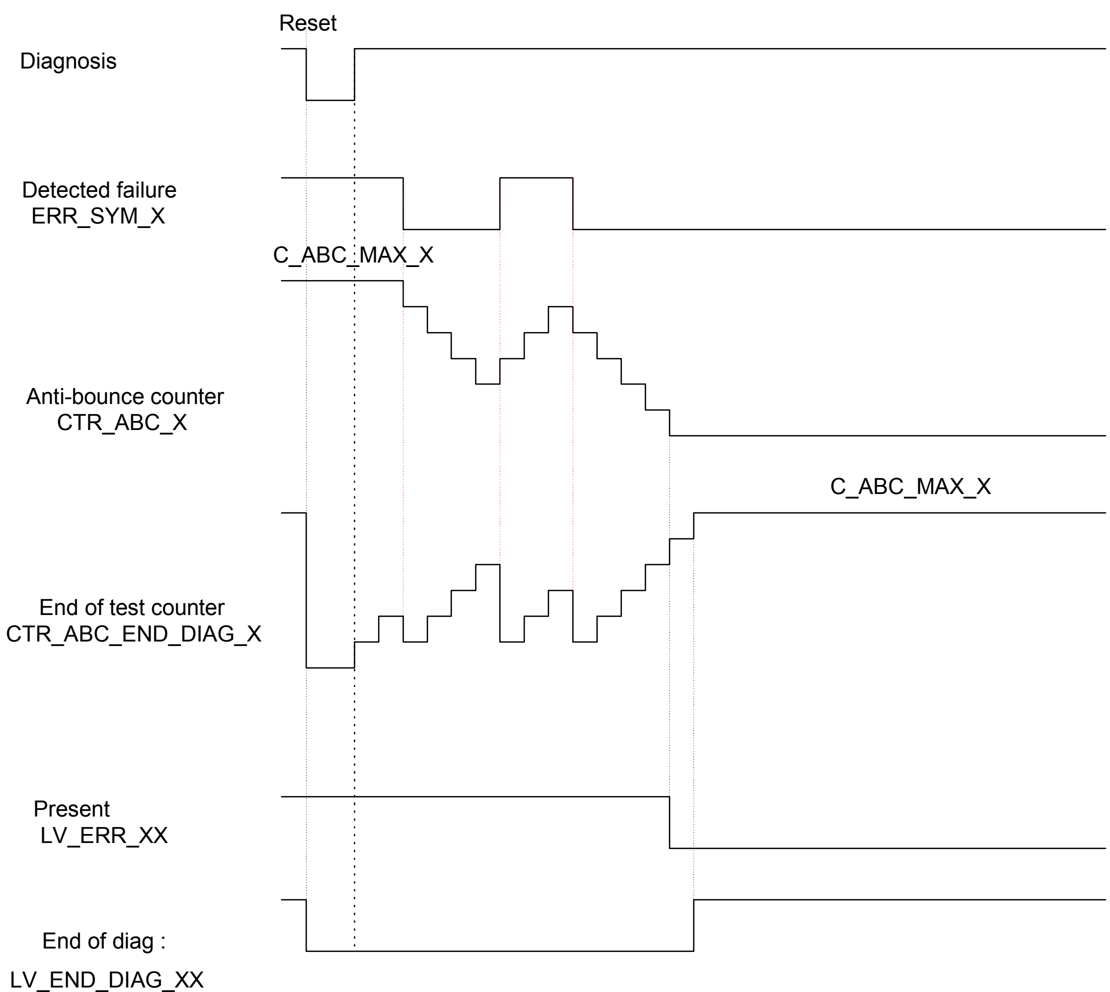
Before the reset, the failure was detected until the failure became present.

After the reset, the detected failure appears during a glitch so the end of diagnosis counter is reset at each transition of the detected failure,

When the detected failure disappears after all (stable) then

- the present failure disappears
- the end of diagnosis counter is incremented until the maximum value while the diagnosis conditions are fulfilled.

  When the end of diagnosis counter has reached the maximum value : the end of diagnosis is set.

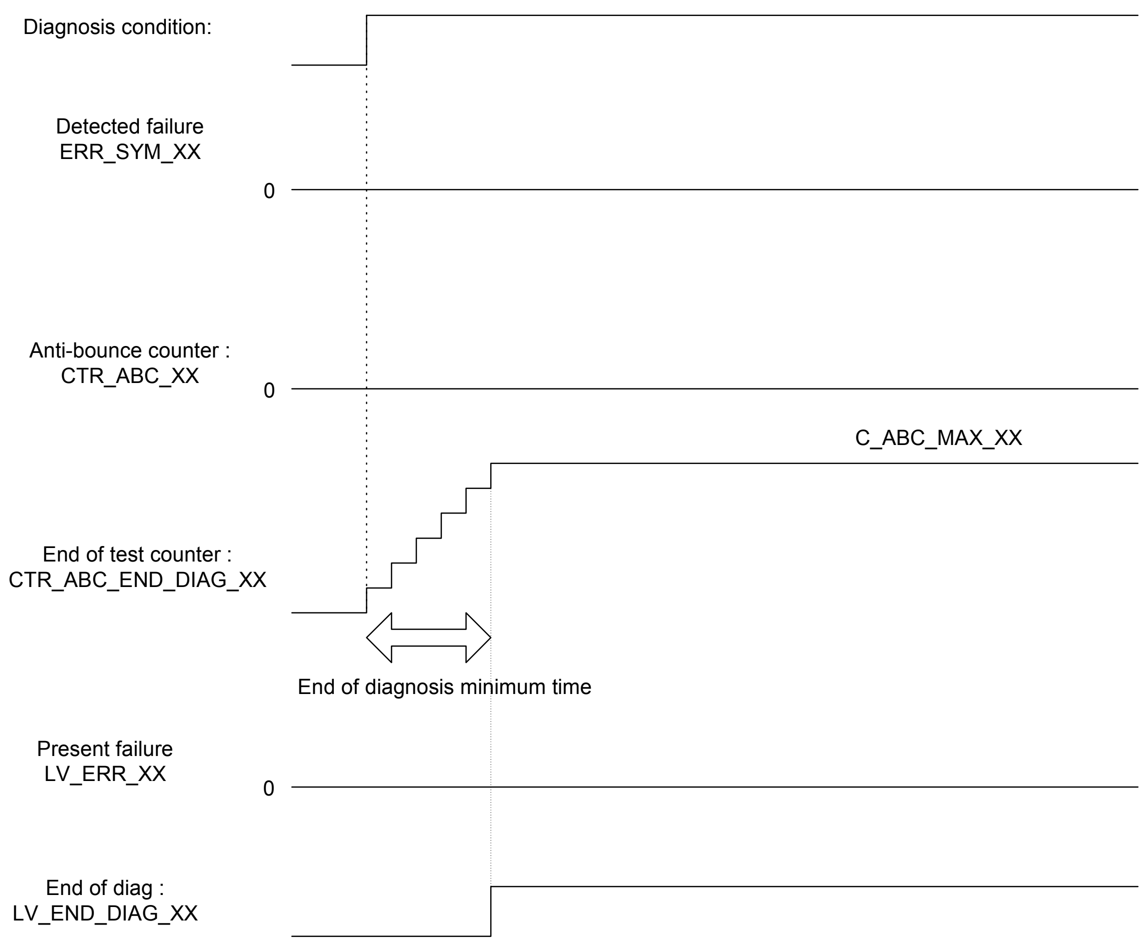


End of diagnosis algorithm illustration when the time to reach the end of diagnosis is minimum

The failure isn't detected (stable) : the end of diagnosis counter is incremented until the maximum value while the diagnosis conditions are fulfilled.

When the end of diagnosis counter has reached the maximum value : the end of diagnosis is set.

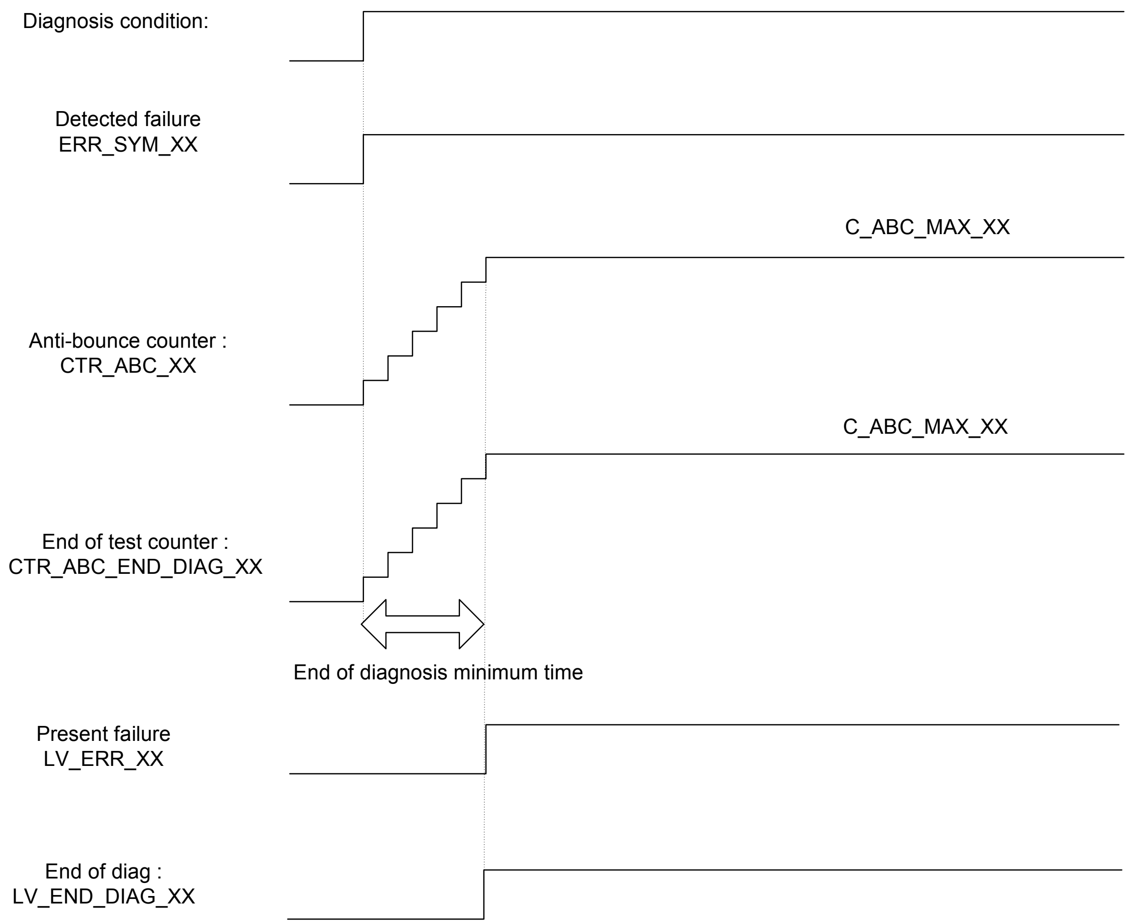
The end of diagnosis appears after the minimum time (the detected failure doesn't appear).



The failure is detected unchangingly (stable) : the end of diagnosis counter is incremented until the maximum value while the diagnosis conditions are fulfilled.

When the failure becomes present : the end of diagnosis is set.

The end of diagnosis and the failure appears after the minimum time (the detected failure is permanent).



Application conditions:

Initialization: See "Initialisation of anti-bounce, statistical and multi-conditions filters" paragraph

Recurrence: The recurrence is managed by the diagnosis.

Activation: The activation is managed by the diagnosis for the anti-bounce operation or the anti-bounce bench mode.

Deactivation: -

Formula section:

If(0) LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*

Then(0)

End of diagnosis in the bench mode:

If(1) (LC_ABC_BENCH = 1 **and** NLC_BENCH_MODE=1) **and** ((C_ABC_INC_XX = 0) **or** (C_ABC_INC_XX = FFH))

Then(1)

LV_CDN_DIAG_XX = 1

LV_END_DIAG_XX = 1

CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX

End of diagnosis in the normal mode:

Else(1)

If(2) C_ABC_INC_XX = 0 *{ component XX not available }*

Then(2)

LV_END_DIAG_XX = 0

CTR_ABC_END_DIAG_XX = 0

Else(2)

If(3a) (LV_CDN_DIAG_XX = 1) **and** (LV_END_DIAG_XX = 0)

Then(3a) **If(4a)** ((ERR_SYM_PREV_XX ≠ 0) **and** (ERR_SYM_XX = 0))

or ((ERR_SYM__PREV_XX = 0) **and** (ERR_SYM_XX ≠ 0))

Then(4a)

CTR_ABC_END_DIAG_XX = 0

Endif(4a)

If(4b) (ERR_SYM_XX ≠ 0)

Then(4b)

CTR_ABC_END_DIAG_XX = (CTR_ABC_END_DIAG_XX+ C_ABC_INC_XX)

Else(4b)

If(5) (NLC_ABC_CAL_DEC_XX = 0)

Then(5)

CTR_ABC_END_DIAG_XX = (CTR_ABC_END_DIAG_XX + 1)

Else(5)

CTR_ABC_END_DIAG_XX =
(CTR_ABC_END_DIAG_XX + C_ABC_DEC_XX)

Endif(5)

Endif(4b)

If(4c) CTR_ABC_END_DIAG_XX >= C_ABC_MAX_XX

Then(4c) LV_END_DIAG_XX = 1

CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX

Endif(4c)

ERR_SYM_PREV_XX = ERR_SYM_XX

Endif(3a)

If(3b) LV_ERR_XX 0 → 1

Then(3b) LV_END_DIAG_XX = 1

Endif(3b)

Endif(2)

Endif(1)

Else(0)

{Inhibition of anti-bounce filtering}

If(1) (LV_CDN_DIAG_XX = 1
or
((LC_ABC_BENCH = 1 **and** NLC_BENCH_MODE=1)
and (C_ABC_INC_XX = 0) **or** (C_ABC_INC_XX = FFH))

Then(1)

{ERR_SYM_PREV_XX is updated at each diagnosis recurrence even during the inhibition, so that when the inhibition is removed ERR_SYM_PREV_XX is updated with the value calculated at the last diagnosis recurrence of the inhibition phase.}

ERR_SYM_PREV_XX = ERR_SYM_XX

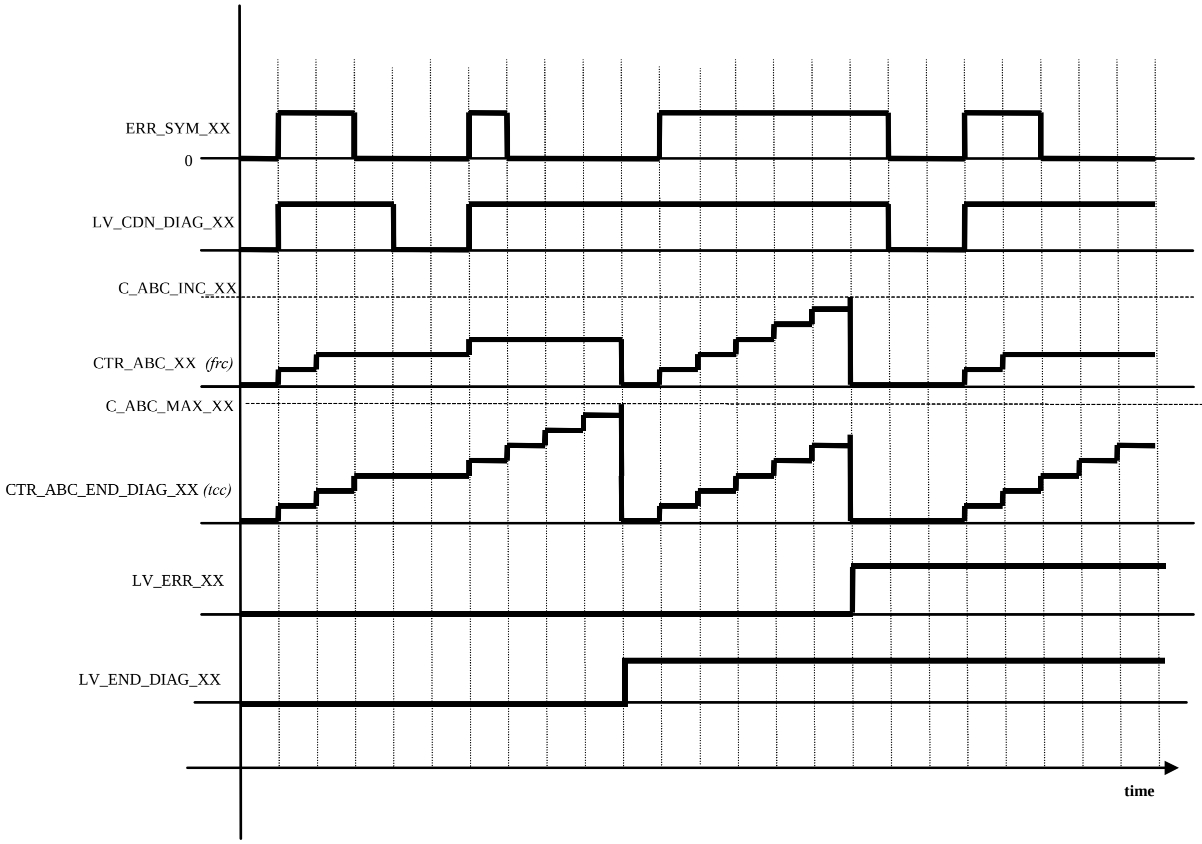
Endif(1)

Endif(0)

A.2.3.3 Standard statistical filtering type

General information:

The purpose is to make an error detection by statistical evaluation.
An error is only detected, if a symptom is active for C_ABC_INC_XX times within the test period C_ABC_MAX XX, with test symptom condition fulfilled (LV_CDN_DIAG_XX = 1).



In this chapter,

- the data CTR_ABC_XX represents the frequency counter
- the calibration C_ABC_INC_XX represents the maximum for frequency counter of the processed symptom
- the data CTR_ABC_END_DIAG_XX represents the period counter increment
- the calibration C_ABC_MAX_XX represents the maximum for test counter of the processed symptom

The calibration C_ABC_MAX_XX must be greater than C_ABC_INC_XX.

Formula section:

Anti-bounce in the bench mode :

```
If      (LC_ABC_BENCH = 1 and NLC_BENCH_MODE=1)
        and
        ((C_ABC_INC_XX = 0) or (C_ABC_INC_XX = FFH))
Then
        If      C_ABC_INC_XX = 0
        Then
                ERR_SYM_XX = 0
                LV_ERR_XX = 0
                CTR_ABC_XX = 0
        Else
                If      C_ABC_INC_XX = FFH
                Then
                        ERR_SYM_XX = 0FH
                        LV_ERR_XX = 1
                        CTR_ABC_XX = C_ABC_INC_XX
                Endif
        Endif
Endif
```

Frequency counter increment:

If LV_CDN_DIAG_XX = 1
and ERR_SYM_XX ≠ 0
and C_ABC_INC_XX ≠ 0
Then
  If NC_ERR_DET_UPD = 1
  and LC_ERR_DET_UPD = 1
  Then
   ACTION_ERRM_UpdErrDet(IN<XX>, SYNCHRONIZATION<CALL>)
  Endif
  CTR_ABC_XX = CTR_ABC_XX + 1
Endif

Period counter increment:

If LV_CDN_DIAG_XX = 1
Then
  CTR_ABC_END_DIAG_XX = CTR_ABC_END_DIAG_XX + 1
Endif

Detection of the error state for the current symptom:

If CTR_ABC_XX >= C_ABC_INC_XX
Then
  LV_ERR_XX = 1
  { The current symptom is entered to failure memory }
Endif

Detection of the error free symptom:

If CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
and CTR_ABC_XX < C_ABC_INC_XX
Then
  LV_ERR_XX = 0
Endif

End of the diagnostic period for statistic symptom:

If CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
or CTR_ABC_XX = C_ABC_INC_XX
Then
  CTR_ABC_XX = 0
  CTR_ABC_END_DIAG_XX = 0
  { End of diagnostic period for statistic symptom is reached }
  LV_END_DIAG_XX=1
Endif

A.2.4 Description for ACTION_ERRM_NoFilterSymptom :

Syntax :	ACTION_ERRM_NoFilterSymptom(IN<XX>, IN<lv_cdn_diag_XX>, IN<err_sym_XX>, IN<lv_err_set_XX>, IN<lv_err_reset_XX>, IN<lv_end_diag_XX>, OUT<LV_ERR_XX>)	
Parameter(in) :	XX	Type: index to identify the Diagnostic
	lv_cdn_diag_XX	Type: Diagnostic condition
	err_sym_XX	Status of failure
	lv_err_set_XX	Type: Boolean to set the failure
	lv_err_reset_XX	Type: Boolean to reset the failure
	lv_end_diag_XX	Type: Boolean (=1 when result available)
Parameter(out) :	LV_ERR_XX	Type: Boolean (=1 when failure)
Short Description :	This action returns the result on symptoms detected at each diagnostic recurrence, when no filter is used.	

A.2.5 System Description:

This action is used in case of diagnostics which don’t need any filtering will use the “NO type” filter.

LV_ERR_XX is managed as a **STD** anti-bounce filtering type (saved in NVMY at end of power latch, ...).

LV_END_DIAG_XX is managed by error management (reset at DC transition, ...).
ERR_SYM_XX is updated at each diagnostic recurrence, to fulfil the diagnostic result.

Formula section :

If LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*
Then

    If lv_end_diag_XX = 1
    Then
        LV_END_DIAG_XX = 1
    Endif

    If lv_err_set_XX = 1
    Then
        LV_ERR_XX = 1
    Endif

    If lv_err_reset_XX = 1
    Then
        LV_ERR_XX = 0
    Endif

    If lv_cdn_diag_XX =1

Then

LV_CDN_DIAG_XX = 1
ERR_SYM_XX = err_sym_xx

Else

LV_CDN_DIAG_XX = 0
{ ERR_SYM_XX is unchanged }

Endif

Endif

A.2.6 Description for ACTION_ERRM_AbcFilterSymptomEnd:

Syntax:	ACTION_ERRM_AbcFilterSymptomEnd(	IN<XX>, IN< C_ABC_MAX_XX>, OUT< LV_ERR_XX >)
Parameter(in):	XX	Type: index to identify the Diagnostic
	C_ABC_MAX_XX	Maximum value of anti-bounce counter
Parameter(out):	LV_ERR_XX	Type: Boolean

Short Description: This action erases diagnostic failure and sets immediately the end of diagnostic information in case of anti-bounce filter (STD, STD_INI, STC, MEM and MEM_INI)

Formula section :

If LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*
Then

LV_CDN_DIAG_XX = 1
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0
CTR_ABC_XX = 0
LV_END_DIAG_XX = 1
CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX

Endif

A.2.7 Description for ACTION_ERRM_NoFilterSymptomEnd:

Syntax:	ACTION_ERRM_NoFilterSymptomEnd(IN<XX>, OUT<LV_ERR_XX>)	
Parameter(in):	XX	Type: index to identify the Diagnostic
Parameter(out):	LV_ERR_XX	Type: Boolean

Short Description: This action erases diagnostic failure and sets immediately the end of diagnostic information in case of no filter (NO)

Formula section :

If LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*
Then

LV_CDN_DIAG_XX = 1
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0

LV_END_DIAG_XX = 1
Endif

A.2.8 Description for ACTION_ERRM_AbcFilterReset:

Syntax: ACTION_ERRM_AbcFilterReset(IN<XX>, OUT<LV_ERR_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_ERR_XX Type: Boolean

Short Description: This action resets data filter in case of ABC filter type

Formula section :

If LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*

Then

LV_CDN_DIAG_XX = 0
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0
CTR_ABC_XX = 0
LV_END_DIAG_XX is unchanged
CTR_ABC_END_DIAG_XX is unchanged

Endif

A.2.9 Description for ACTION_ERRM_NoFilterReset:

Syntax: ACTION_ERRM_NoFilterReset(IN<XX>, OUT<LV_ERR_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_ERR_XX Type: Boolean

Short Description: This action resets data filter in case of no filter usage

Formula section :

If LV_ABC_INH = 0 *{No inhibition of anti-bounce filtering}*

Then

LV_CDN_DIAG_XX = 0
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0
LV_END_DIAG_XX is unchanged

Endif

A.2.10 Description for ACTION_ERRM_GetLvErr:

Syntax: ACTION_ERRM_GetLvErr(IN<XX>, OUT<LV_ERR_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_ERR_XX Type: Boolean

Short Description: This action returns the failure status for the diagnostic XX.

Formula section :

LV_ERR_XX = failure status for the diagnostic XX

A.2.11 Description for ACTION_ERRM_GetLvEndDiag:

Syntax: ACTION_ERRM_GetLvEndDiag(IN<XX>, OUT<LV_END_DIAG_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_END_DIAG_XX Type: Boolean

Short Description: This action returns the end of diagnostic flag for the diagnostic XX.

Formula section :

LV_END_DIAG_XX = end of diagnostic flag for the diagnostic XX

A.2.12 Description for ACTION_ERRM_GetLvCdnDiag:

Syntax: ACTION_ERRM_GetLvCdnDiag(IN<XX>, OUT<LV_CDN_DIAG_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_CDN_DIAG_XX Type: Boolean

Short Description: This action returns the condition status for the diagnostic XX.

Formula section :

LV_CDN_DIAG_XX = condition status for the diagnostic XX

A.2.13 Description for ACTION_ERRM_GetErrSym:

Syntax: ACTION_ERRM_GetErrSym(IN<XX>, OUT<ERR_SYM_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): ERR_SYM_XX Status of failure

Short Description: This action returns the symptoms for the diagnostic XX.

Formula section :

ERR_SYM_XX = symptoms for the diagnostic XX

A.2.14 CPU load optimisation mechanisms

General information:

This section describes the mechanism of optimisation. The principle is to suspend the filter algorithm execution when it is not necessary.

Formula section:

If NLC_BENCH_MODE = 0
Then
{ the bench mode isn't available whatever LC_ABC_BENCH value }
the CPU load is reduced
Else
{ the bench mode is available if LC_ABC_BENCH = 1 }
less CPU load reduction than with NLC_BENCH_MODE = 0
Endif

Case 1 : the diagnostic uses an anti-bounce filtering algorithm

If LV_ERR_XX = 0
and LV_END_DIAG_XX = 1
and ERR_SYM_XX = 0
and CTR_ABC_XX = 0
Then
{ it is not necessary to call the Error Management (including the filtering) }
LV_END_DIAG_XX = 1 *{ unchanged }*
LV_ERR_XX = 0 *{ unchanged }*
Endif

Case 2 : the diagnostic doesn't use an anti-bounce filtering algorithm

If (LV_ERR_XX = 0 **and** LV_END_DIAG_XX = 1 **and** ERR_SYM_XX = 0)
Then
{ it is not necessary to call the Error Management }
LV_END_DIAG_XX = 1 *{ unchanged }*
LV_ERR_XX = 0 *{ unchanged }*
Endif

A.2.15 Synchronisation between Anti-bounce algorithm and Error Management

General information:

This section describes the mechanism of synchronisation of the failure filtering with the Error Management core. The principle is to call the Error Management upon events.

LV_END_DIAG_XX 0 → 1 transition

LV_ERR_XX 0 → 1 transition

LV_ERR_XX 1 → 0 transition

This mechanism is followed whatever the filter (type STD, STD_INI, DEC_CAL, MEM, MEM_INI, STC) but also for diagnoses which don't use any filter (type NO).

A.2.16 Configuration and calibration data

NC_ABC_CONF_FCT_DIAG_XX :

```
┌─┬─┬─┬─┬─┬─┬─┬─┐
└─┴─┴─┴─┴─┴─┴─┴─┘
 | | | | | | | └─>  bit 0 : NLC_ABC_NOT_SAVE_XX
 | | | | | | └───>  bit 1 : NLC_ABC_INI_IGK_XX
 | | | | | └─────>  bit 2 : NLC_ABC_INI_VALUE_XX
 | | | | └───────>  bit 3 : NLC_ABC_NOT_DEC_XX
 | | | └─────────>  bit 4 : NLC_ABC_CAL_DEC_XX
 | | └───────────>  bit 5 : NLC_ABC_STC_XX
 | └─────────────>  bit 6 : Not used
 └───────────────>  bit 7 : Not used
```

Bits definition

- NLC_ABC_NOT_SAVE_XX : if this bit is set to 0 the anti-bounce context (CTR_ABC_XX and LV_ERR_XX) is saved at the end of power latch, only when the failure is in dynamic memory, and restored at the event Reset.
Limitation : the saved configuration is only available if the NVMY manager take into account the filter environment data (LV_ERR_XX, CTR_ABC_XX).

- NLC_ABC_INI_IGK_XX : LV_ERR_XX and CTR_ABC_XX are reset at the transition LV_IGK 0 -> 1 and at the event reset.

- NLC_ABC_INI_VALUE_XX : if this bit is set to 1 then the anti-bounce counter is initialised with C_ABC_MAX_XX – C_ABC_INC_XX if there was a failure between the last and this initialisation (NLC_ABC_NOT_SAVE_XX = 0 to test the failure before this initialisation) else the anti-bounce counter is initialised with 0 value.

- NLC_ABC_NOT_DEC_XX : if this bit is set to 0 then the anti-bounce counter can be decremented if the condition for decrement are fulfilled. If this bit is set to 1 then the anti-bounce counter can be decremented if the condition for decrement are fulfilled until the anti-bounce counter hasn't reached the maximum value; when the anti-bounce counter has reached the maximum value then the anti-bounce counter and the failure remains (the anti-bounce counter can be decremented anymore) until a transition LV_IGK 0 → 1 or reset following the NLC_ABC_INI_IGK_XX bit configuration (If the NLC_ABC_NOT_DEC_XX bit is set to 1 then NLC_ABC_NOT_SAVE_XX must be set to 1).

- NLC_ABC_CAL_DEC_XX : if this bit is set then the anti-bounce decrement is calibratable, the calibration is named C_ABC_DEC_XX else the anti-bounce decrement is equal to 1 and there isn't calibration.

- NLC_ABC_STC_XX : if this bit is set then the filter used is the statistical filter else the filter used is the anti-bounce filter (Standard configuration).

Predefined filter type :

	NLC_ABC_STC_XX	NLC_ABC_CAL_DEC_XX	NLC_ABC_NOT_DEC_XX	NLC_ABC_INI_VALUE_XX	NLC_ABC_INI_IGK_XX	NLC_ABC_NOT_SAVE_XX
STD_INI	0	0	0	0	1	1
STD	0	0	0	0	0	0
MEM	0	0	1	0	1	1
MEM_INI	0	0	1	1	1	0
DEC_CAL	0	1	0	0	0	0
STC	1	0	0	0	1	1
NO	0	0	0	0	0	0

The **standard configuration with initialisation (STD_INI)** is
NC_ABC_CONF_FCT_DIAG_XX = 3H (default configuration)

The data CTR_ABC_XX and LV_ERR_XX are reset at the transition LV_IGK 0 → 1 and reset (NLC_ABC_NOT_SAVE_XX = 1, NLC_ABC_INI_IGK_XX = 1) and the anti-bounce counter is initialised to 0 value (NLC_ABC_INI_VALUE_XX = 0).

The anti-bounce counter may be decremented (NLC_ABC_NOT_DEC_XX = 0).
The decrement value is equal to 1 (NLC_ABC_CAL_DEC_XX = 0).
The filter used is the Anti-bounce (NLC_ABC_STC_XX = 0).

The **statistical configuration (STC)** is NC_ABC_CONF_FCT_DIAG_XX = 23H

The data CTR_ABC_XX and LV_ERR_XX are reset at the transition LV_IGK 0 → 1 and reset (NLC_ABC_NOT_SAVE_XX = 1, NLC_ABC_INI_IGK_XX = 1) and the frequency counter (CTR_ABC_XX) is initialised to 0 value (NLC_ABC_INI_VALUE_XX = 0).

The filter used is the statistical (NLC_ABC_STC_XX = 1).

The **standard configuration (STD)** is NC_ABC_CONF_FCT_DIAG_XX = 0H

The data CTR_ABC_XX and LV_ERR_XX are saved (NLC_ABC_NOT_SAVE_XX = 0, NLC_ABC_INI_IGK_XX = 0, NLC_ABC_INI_VALUE_XX = 0).

The anti-bounce counter may be decrement (NLC_ABC_NOT_DEC_XX = 0).

The decrement value is equal to 1 (NLC_ABC_CAL_DEC_XX = 0).

The **memory configuration (MEM)** is NC_ABC_CONF_FCT_DIAG_XX = BH

The data CTR_ABC_XX and LV_ERR_XX are reset at the transition LV_IGK 0 → 1 and reset (NLC_ABC_NOT_SAVE_XX = 1, NLC_ABC_INI_IGK_XX = 1) and the anti-bounce counter is initialised to 0 value (NLC_ABC_INI_VALUE_XX = 0).

The anti-bounce counter can be decremented if the condition for decrement are fulfilled until the anti-bounce counter hasn't reached the maximum value; when the anti-bounce counter has reached the maximum value then the anti-bounce counter and the failure remains until a transition LV_IGK 0 → 1 or reset (NLC_ABC_NOT_DEC_XX = 1).

The decrement value is equal to 1 (NLC_ABC_CAL_DEC_XX = 0).

The **memory configuration (MEM_INI)** is NC_ABC_CONF_FCT_DIAG_XX = EH

The data CTR_ABC_XX and LV_ERR_XX are initialised at the transition LV_IGK 0 → 1 and reset (NLC_ABC_INI_IGK_XX = 1). The anti-bounce counter is initialised to the C_ABC_MAX_XX – C_ABC_INC_XX value (NLC_ABC_INI_VALUE_XX = 1) only if there was a failure between the last and this initialisation (NLC_ABC_NOT_SAVE_XX = 0 to test the failure before this initialisation) else the anti-bounce counter is initialised with 0 value. The data LV_ERR_XX is initialised with 0 value.

The anti-bounce counter can be decremented if the condition for decrement are fulfilled until the anti-bounce counter hasn't reached the maximum value; when the anti-bounce counter has reached the maximum value then the anti-bounce counter and the failure remains until a transition LV_IGK 0 → 1 or reset (NLC_ABC_NOT_DEC_XX = 1).

The decrement value is equal to 1 (NLC_ABC_CAL_DEC_XX = 0).

The **decrement calibratable configuration (DEC_CAL)** is set with NC_ABC_CONF_FCT_DIAG_XX = 10H

The data CTR_ABC_XX and LV_ERR_XX are saved (NLC_ABC_NOT_SAVE_XX = 0, NLC_ABC_INI_IGK_XX = 0, NLC_ABC_INI_VALUE_XX = 0).

The anti-bounce counter may be decrement (NLC_ABC_NOT_DEC_XX = 0).

The decrement value is calibratable (NLC_ABC_CAL_DEC_XX = 1).

The standard configuration is NLC_ABC_INI_DC_END_DIAG = 0 and NC_ERR_DET_UPD = 0 (default configuration).

Some functionalities (e.g. : to visualise the sporadic error of all diagnosis using this anti-bounce algorithm) aren't used when the failure is detected.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ABC_BENCH	1	0...1H	0...1	1	[-]
Logical constant to manage the anti-bounce bench mode					
LC_ERR_DET_UPD	1	0...1H	0...1	1	[-]
Logical constant to manage some functionality as the visualisation of the sporadic error					
C_ABC_DEC_XX	1	0...FFH	0...255	1	[-]
Anti-bounce counter decrement for XX diagnosis					
C_ABC_INC_XX	1	0...FFH	0...255	1	[-]
XX diagnosis anti-bounce counter increment (for anti-bounce filtering) or frequency counter maximum value (for statistical filtering)					
C_ABC_MAX_XX	1	1...FFH	1...255	1	[-]
XX diagnosis anti-bounce counter maximum value (for anti-bounce filtering) or threshold of period counter for failure detection (for statistical filtering)					

Remark :

The standard calibration is LC_ABC_BENCH = 0 **and** NLC_BENCH_MODE=0 (default calibration) then the bench mode isn't managed.

If the diagnosis doesn't manage calibratable decrement then the C_ABC_DEC_XX calibration isn't defined in calibration data (NLC_ABC_CAL_DEC_XX = 0).

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ABC_CONF_FCT_DIAG_XX	1	0...FFH	0...255	1	[-]
Anti-bounce algorithm configuration of each XX diagnosis					
NLC_ABC_INI_DC_END_DIAG	1	0...1H	0...1	1	[-]
Initialisation of LV_END_DIAG_XX at LV_DC transition or not					
NLC_BENCH_MODE	1	0...1H	0...1	1	[-]
Bench mode configuration (0: the bench mode is removed, max CPU optimisation is reached)					

Configuration detailed description:

NLC_ABC_INI_DC_END_DIAG

1 : Initialisation of LV_END_DIAG_XX at LV_DC 1→0 transition

0 : Initialisation of LV_END_DIAG_XX at LV_IGK 0→1 transition or ECU reset

A.3 Debounce algorithm (Appl. Inc.)

A.3.1 Visualization of intermittent failures

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_INTM_DIAG_INST[NC_NR_ERR_INTM]	V/S	0...FFFFH	0...65535	1	[-]
array of diagnosis instances with intermittent failure					
ERR_INTM_SYM[NC_NR_ERR_INTM]	V/S	0...FFH	0...255	1	[-]
array of error symptoms of intermittent failures					
CTR_ERR_INTM_NR	V/S	0...FFH	0...255	1	[-]
Number of failures stored in intermittent failure array					
ERR_INTM_DIAG_INST_ACT	V	0...FFH	0...255	1	[-]
diagnosis instance to display latest entry of ERR_INTM_DIAG_INST					
ERR_INTM_ENVD_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN][NC_NR_FRF_SET][NC_NR_ERR_INTM]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_SET_CMN : environmental data common to all failure (not fixed by law) stored in set					
ERR_INTM_ENVD_CUS_SET_SPC[NC_NR_ENVD_CUS_SET_SPC][NC_NR_FRF_SET][NC_NR_ERR_INTM]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_SET_SPC : environmental data specific to each failure (not fixed by law) stored in set					

Input data:

ERR_SYM_XX	LV_IGK	DIAG_INST[NC_NR_ERR_DYN]	ENVD_CONF_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN]
ID_ERR_ENVD_XX	LV_ERR_XX	LV_END_DIAG_XX	NC_ERR_DET_UPD
LC_ERR_DET_UPD	NC_NR_ERR_INTM		

Export actions:

ACTION_ERRM_UpdErrDet (IN <XX>)
This action is used to visualize intermittent failures
ACTION_ERRM_IntmClr(IN < PRM_IDX_DYN >)
This action is used to clear one single Intm failure

FUNCTION DESCRIPTION:

General information:

The purpose of this function is the visualization of those diagnosis instances that show intermittent failures (ERR_SYM_XX ≠ 0) but do not enter into the dynamic error management structure, because the antibounce counter does not reach its maximum value.
If a failure of a diagnosis whitout using the anti-bounce algorithm is detected or if CTR_ABC_xx of a diagnosis using the anti-bounce algorithm is incremented for the 2nd time, it is stored in the array ERR_INTM_DIAG_INST[NC_NR_ERR_INTM] and its symptom is stored in the array ERR_INTM_SYM[NC_NR_ERR_INTM]. This is done for every new failure as long as the maximum number of entries NC_NR_ERR_INTM of this structure has not been reached.
If a new failure is stored, it is displayed also in ERR_INTM_DIAG_INST_ACT. If the antibounce counter of a failure reaches its maximum value it will be additionally stored in dynamic error memory.

The ERR_INTM_ENVD_... values refer to the same logic as ENVD_... of the normal FMY. The Information is stored if CTR_ABC_xx of a diagnosis using the anti-bounce algorithm is incremented for the 2nd time or if the failure flag is set for any kind of diagnosis.

A.3.1.1 Storage of intermittent failures and their frequency of appearance

Description:

Syntax : ACTION_ERRM_UpdErrDet (IN <XX>)

Parameter (in) : *Function_Name* Type: index to identify the function

Parameter (out) : *No parameter*

Short description : This action is used to store intermittent failures

Application conditions:

Initialization: -

Recurrence: At action call

Activation: NC_ERR_DET_UPD = 1 **and** LC_ERR_DET_UPD = 1

Formula section:

```
If    ERR_SYM_XX ≠ 0  {at the first time for failures not using anti-bounce algorithm and
                       at the 2nd increment of CTR_ABC_XX for diagnosis using
                       anti-bounce algorithm }
Then  If    diagnosis instance XX is not yet stored
            in ERR_INTM_DIAG_INST[ 0 ... NC_NR_ERR_INTM –1 ]
      Then  If    CTR_ERR_INTM_NR < NC_NR_ERR_INTM
            Then  ERR_INTM_DIAG_INST[ CTR_ERR_INTM_NR ] = XX
                  ERR_INTM_SYM[ CTR_ERR_INTM_NR ] = ERR_SYM_XX
                  ERR_INTM_DIAG_INST_ACT = XX
                  CTR_ERR_INTM_NR = CTR_ERR_INTM_NR + 1
                  ERR_INTM_ENVD_CUS_SET_CMN = ENVD_CONF_CUS_SET_CMN
                  ERR_INTM_ENVD_CUS_SET_SPC = ID_ERR_ENVD_XX
            Endif
      Else  If    transition LV_ERR_XX 0 ->1                  or
                  ( transition LV_END_DIAG_XX 0 ->1 and LV_ERR_XX =1 )
            Then  ERR_INTM_ENVD_CUS_SET_CMN = ENVD_ CONF_CUS_SET_CMN
                  ERR_INTM_ENVD_CUS_SET_SPC = ID_ERR_ENVD_XX
            Endif
      Endif
Endif
```

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_ERR_INTM	1	1...FFH	1...255	1	[-]
Maximum number of diagnosis instances to be listed in ERR_INTM_DIAG_INST NC_NR_ERR_INTM = 10					

A.3.1.2 Clear a single intermittent failure corresponding to dynamic entry (PRM_IDX_DYN)

Description:

Syntax : ACTION_ERRM_IntmClr (IN < PRM_IDX_DYN >)

Parameter (in) : *Function_Name* Type: index to identify the corresponding PRM_IDX_DYN

Parameter (out) : *No parameter*

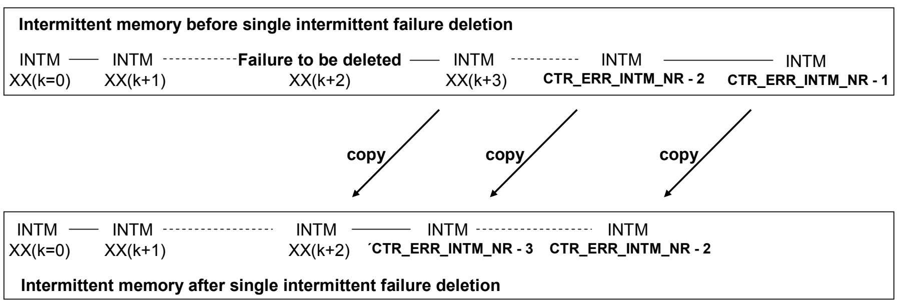
Short description : This action is used to erase a single intermittent failure corresponding to failure entry in Dynamic layer (PRM_IDX_DYN)

Application conditions:

Initialization: -

Recurrence: At action call

Activation:



Formula section:

IF (*NLC_ERR_DET_UPD = 1* **and** *LC_ERR_DET_UPD = 1)*

For i = 0 to CTR_ERR_INTM_NR -1

IF ERR_INTM_DIAG_INST[i] = = DIAG_INST[PRM_IDX_DYN]

Remove corresponding intermittent failure, corresponding intermittent symptom and corresponding Intermittent FrF (CUS_SET_CMN and CUS_SET_SPC)

Sort intermittent array:
- ERR_INTM_DIAG_INST
- ERR_INTM_SYM
- ERR_INTM_ENVD_CUS_SET_CMN
- ERR_INTM_ENVD_CUS_SET_SPC

End For

End If

A.3.1.3 Clear array of intermittent failures

Description:

By setting the calibration bit LC_ERR_INTM_CLR from 0 to 1 all failure entries are reset.

The array of intermittent failures is also initialised on other system events. This depends on the configuration C_ERR_INTM_CLR_INI

Application conditions:

Initialisation: -

Recurrence: -

Activation: at ECU reset, if bit 0 of C_ERR_INTM_CLR_INI = 1

at LV_IGK 0->1, if bit 1 of C_ERR_INTM_CLR_INI = 1

at Error Management cleared, if bit 2 of C_ERR_INTM_CLR_INI = 1

at transition LC_ERR_INTM_CLR 0→1 (recurrence is 1 second)

on failure erase service received

Formula section:

Reset array of intermittent failures:

ERR_INTM_DIAG_INST[0 ... NC_NR_ERR_INTM –1] = 0

ERR_INTM_SYM[0 ... NC_NR_ERR_INTM –1] = 0

CTR_ERR_INTM_NR = 0

ERR_INTM_DIAG_INST_ACT = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ERR_INTM_CLR	1	0...1H	0...1	1	[-]
reset array of intermittent failures					
C_ERR_INTM_CLR_INI	1	0...7H	0...7	1	[-]
control of different init possibilities of array of intermittent failures bit 0: clear of array at ECU reset bit 1: clear of array at LV_IGK 0->1 bit 2: clear of array, if Error management is cleared (initialisation active, if bit = 1)					

A.3.2 Anti-bounce filtering inhibition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ABC_INH	V/O	0...1H	0...1	1	[-]
Anti-bounce filterings inhibition					

Export actions:

ACTION_ERRM_InhibitFiltering (IN<InhibitFilter>)
This action shall be used to inhibit anti-bounce filtering calculations

Description:

Important note: Usage of the API is under project responsibility. By default, the inhibition is not set.

In addition, LV_ABC_INH flag must be securised to avoid any undesired inhibition.

To avoid failure recording during specific ECU phases (e.g. reprogramming,...), diagnoses are inhibited. To realize this inhibition, anti-bounce filtering calculations are frozen.

This inhibition is generally used during reprogramming phase.

Syntax: ACTION_ERRM_InhibitFiltering (IN <InhibitFilter>)

Parameter(in): InhibitFilter filter inhibition

Parameter(out: No parameter

Short Description: This API permits to activate or deactivate the anti-bounce calculation.

Application conditions:

Initialisation: *at ECU reset*

LV_ABC_INH = 0

(default value : Anti-bounce calculation is not inhibited)

Recurrence: -

Activation: at action request

Formula section:

If InhibitFilter = ON
Then
LV_ABC_INH = 1
Else
LV_ABC_INH = 0
Endif

A.4 Multi-condition debounce algorithm

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
CTR_ABC_XX	O/V/S	0...FFH	0...255	1	[-]
anti bounce counter of diagnosis XX					
CTR_ABC_END_DIAG_XX	O/V	0...FFH	0...255	1	[-]
Counter for end of diagnosis XX generation					
CDN_DIAG_PREV_XX	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Diagnostic condition of last failure symptom					

Input data:

LC_ABC_BENCH	NC_ABC_CONF_FCT_DIAG_XX	NLC_ABC_INI_DC_END_DIAG	NC_ERR_DET_UPD
LC_ERR_DET_UPD	LV_IGK	LV_ABC_INH	NLC_BENCH_MODE

Import actions:

#IF NC_ERR_DET_UPD = 1 and NLC_ENA_MULTI_CDN = 1
ACTION_ERRM_UpdErrDet (IN<XX>)
This action indicates each occurrence of the detected error
#ENDIF

Export actions:

ACTION_ERRM_FilterMulticondition(IN< *XX* >, IN< CDN_DIAG_XX >, IN< ERR_DIAG_XX >, IN<C_ABC_INC_XX>, IN<C_ABC_MAX_XX>,OUT<LV_ERR_XX>)
This action compute the elementary antibounce filter for one failure treatment and return filter result
ACTION_ERRM_MultiFilterSymEnd(IN< XX >, IN<C_ABC_MAX_XX>, OUT<LV_ERR_XX>)
This action erases diagnostic failure and sets the end of diagnostic in case of multi-conditions filter type
ACTION_ERRM_MultiFilterReset(IN< XX >,OUT<LV_ERR_XX>)
This action resets data filter in case of multi-conditions filter type

General information:

Description:

According the diagnosis condition information CDN_DIAG_XX and the diagnosis symptom (raw value) information the symptom calculation ERR_SYM_XX is performed. Then the failure symptom is debounced to set the present failure LV_ERR_XX.

A special readiness calculation is done to set the end of diagnosis information. LV_END_DIAG_XX is set only if there was enough time to make a diagnosis of all symptoms or a failure gets present.

Note:

To set the LV_END_DIAG_XX is necessary to have the conditions for all symptoms at the same time for a given time.

Restrictions:

In case of multiple symptoms at the same time, only one symptom will be taken into account according the following priority rule: SYM_0 ==> SYM_3 ; Priority max ==> Priority Min

Only the last symptom present will be taken into account in this algorithm, so in case of symptom change, only the last symptom will be used to increment and decrement the counters.

Application conditions:

Initialization: *At ECU reset and LV_IGK 0 → 1 transition*

CDN_DIAG_PREV_XX = 0

Other data initialization

ERR_SYM_XX = *Refer to “Antibounce algorithms” chapter for initialization*
CTR_ABC_XX = *Refer to “Antibounce algorithms” chapter for initialization*
LV_ERR_XX = *Refer to “Antibounce algorithms” chapter for initialization*
CTR_ABC_END_DIAG_XX = *Refer to “Antibounce algorithms” chapter for initialization*

A.4.1 Multi-condition anti-bounce filter algorithm description

Description for ACTION_ERRM_FilterMulticondition:

Syntax : ACTION_ERRM_FilterMulticondition (IN<XX>, IN<CDN_DIAG_XX>, IN<ERR_DIAG_XX>, IN<C_ABC_INC_XX>, IN<C_ABC_MAX_XX>, OUT<LV_ERR_XX>)

Parameter(in):	XX	Type: index to identify the Diagnostic
	CDN_DIAG_XX	Type: Byte carrier (Each symptom condition)
	ERR_DIAG_XX	Type: Byte carrier (Raw value symptom)
	C_ABC_INC_XX	Type: Calibration byte for the increment value
	C_ABC_MAX_XX	Type: Calibration byte for the maximum value
Parameter(out):	LV_ERR_XX	Type: Boolean (=1 when failure)

Short Description: This action returns information necessary for the diagnostic and the error management.

Formula section:

Anti-bounce bench mode :

```
If(0) LV_ABC_INH = 0
Then
If(1) (LC_ABC_BENCH = 1 and NLC_BENCH_MODE=1) and ((C_ABC_INC_XX = 0) or
(C_ABC_INC_XX = FFH))
Then(1)
        If(2)   C_ABC_INC_XX = 0
        Then(2)
                {Case force no failure}
                LV_ERR_XX = 0
                ERR_SYM_XX =  0
                CTR_ABC_XX = 0
                LV_END_DIAG_XX = 1
                CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
        Else(2)
                If(3)   C_ABC_INC_XX = FFH
                Then(3)
                        {Forced failure}
                        LV_ERR_XX = 1
                        ERR_SYM_XX = 0Fh
                        CTR_ABC_XX = C_ABC_MAX_XX
                        LV_END_DIAG_XX = 1
                        CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
                Endif(3)
        Endif(2)
Endif(1)
```

Symptom calculation:

{Erase not available bit}

CDN_DIAG_XX = CDN_DIAG_XX **and** NC_NR_SYM_XX
ERR_DIAG_XX = ERR_DIAG_XX **and** NC_NR_SYM_XX

If(1) CDN_DIAG_XX ≠ 0 *{Diagnosis of at least one symptom is valid}*
Symptom SYM_0 calculation:

Then(1)
If(2) bit 0 of CDN_DIAG_XX = 1
and
bit 0 of ERR_DIAG_XX = 1
Then(2)
ERR_SYM_XX = 1H
{Memorization of last failure symptom number}
CDN_DIAG_PREV_XX = 1H

Symptom SYM_1 calculation:

Else(2)
If(3) bit 1 of CDN_DIAG_XX = 1
and
bit 1 of ERR_DIAG_XX = 1
Then(3)
ERR_SYM_XX = 2H
{Memorization of last failure symptom number}
CDN_DIAG_PREV_XX = 2H

Symptom SYM_2 calculation:

Else(3)
If(4) bit 2 of CDN_DIAG_XX = 1
and
bit 2 of ERR_DIAG_XX = 1
Then(4)
ERR_SYM_XX = 4H
{Memorization of last failure symptom number}
CDN_DIAG_PREV_XX = 4H

Symptom SYM_3 calculation:

Else(4)
If(5) bit 3 of CDN_DIAG_XX = 1
and
bit 3 of ERR_DIAG_XX = 1
Then(5)
ERR_SYM_XX = 8H
{Memorization of last failure symptom number}
CDN_DIAG_PREV_XX = 8H

Symptom calculation for 'No symptom detected':

```
                              Else(5)
                                 If      bit 0 of CDN_DIAG_XX = 1
                                 Then
                                         bit 0 of ERR_SYM_XX = 0
                                 Endif
                                 If      bit 1 of CDN_DIAG_XX = 1
                                 Then
                                         bit 1 of ERR_SYM_XX = 0
                                 Endif
                                 If      bit 2 of CDN_DIAG_XX = 1
                                 Then
                                         bit 2 of ERR_SYM_XX = 0
                                 Endif
                                 If      bit 3 of CDN_DIAG_XX = 1
                                 Then
                                         bit 3 of ERR_SYM_XX = 0
                                 Endif
                              Endif(5)
                         Endif(4)
                     Endif(3)
                 Endif(2)
Endif(1)
If(1)            (CDN_DIAG_PREV_XX = 0 and CDN_DIAG_XX ≠ 0)
                 or      (bit 0 of CDN_DIAG_PREV_XX = 1 and bit 0 of CDN_DIAG_XX = 1)
                 or      (bit 1 of CDN_DIAG_PREV_XX = 1 and bit 1 of CDN_DIAG_XX = 1)
                 or      (bit 2 of CDN_DIAG_PREV_XX = 1 and bit 2 of CDN_DIAG_XX = 1)
                 or      (bit 3 of CDN_DIAG_PREV_XX = 1 and bit 3 of CDN_DIAG_XX = 1)
Then(1)
                 LV_CDN_DIAG_XX = 1
Else(1)
                 {Anti-bounce value is frozen}
                 LV_CDN_DIAG_XX = 0
Endif(1)
```

Anti-bounce of diagnosis instance XX:

```
If(1)        LV_CDN_DIAG_XX = 1
```

Incrementation of antibounce:

```
Then(1)
            If(2a)   ERR_SYM_XX ≠ 0
            Then(2a) If    C_ABC_INC_XX ≠ 0
                     Then  { at each detected error, a treatment may be done }
                           If     NC_ERR_DET_UPD = 1 and LC_ERR_DET_UPD = 1
                           Then
                                  ACTION_ERRM_UpdErrDet(IN<XX>,
                                  SYNCHRONIZATION<CALL>)
                           Endif

                           If     CTR_ABC_XX ≠ C_ABC_MAX_XX
                           Then
                                  CTR_ABC_XX = CTR_ABC_XX + C_ABC_INC_XX
                                  If     CTR_ABC_XX ≥ C_ABC_MAX_XX
                                  Then
                                         LV_ERR_XX = 1
                                         CTR_ABC_XX = C_ABC_MAX_XX
                                  Endif
                           Endif
                     Endif
            Else (2a)
                     If    CTR_ABC_XX ≠ 0
                     Then
                           {Decrementation of antibounce}
                           CTR_ABC_XX = CTR_ABC_XX – 1
                           If     CTR_ABC_XX = 0              { minimum value }
                           Then
                              LV_ERR_XX = 0
                              {Reset of last failure symptom memorized}
                              CDN_DIAG_PREV_XX = 0H
                           Endif
                     Endif
            Endif(2a)
```

Readiness calculation for diagnosis instance XX:

```
            If(2b)         CTR_ABC_END_DIAG_XX < C_ABC_MAX_XX
```

End of diagnosis is set, if failure gets present:

```
        Then(2b) If   LV_ERR_XX = 1              { failure XX is present }
             Then     LV_END_DIAG_XX = 1
                      CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
```

End of diagnosis is set, if all symptoms can be detected after a specified time:

```
             Else
                      {Check if all symptoms can be detected if all significant bit 0, bit 1,
                      bit 2 and bit 3 of CDN_DIAG_XX = 1 }
                      If      CDN_DIAG_XX = NC_NR_SYM_XX
                              and
                              ERR_SYM_XX = 0
                      Then
                              CTR_ABC_END_DIAG_XX = CTR_ABC_END_DIAG_XX
                                                    + C_ABC_INC_XX
                              If      CTR_ABC_END_DIAG_XX ≥ C_ABC_MAX_XX
                              Then
                                      CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX
                                      LV_END_DIAG_XX = 1
                              Endif
                      Else
                              {Check if a failure is detected}
                              If      ERR_SYM_XX ≠ 0
                              Then
                                      CTR_ABC_END_DIAG_XX = 0
                              Endif
                      Endif
             Endif
        Endif(2b)
Endif(1)

Endif(0)
```

A.4.2 Description for ACTION_ERRM_MultiFilterSymEnd :

Syntax: ACTION_ERRM_MultiFilterSymEnd(IN<XX>, IN<C_ABC_MAX_XX>, OUT<LV_ERR_XX>)

Parameter(in): XX Type: index to identify the Diagnostic
C_ABC_MAX_XX Type: Calibration byte for the maximum value

Parameter(out): LV_ERR_XX Type: Boolean

Short Description: This action erases diagnostic failure and sets immediately the end of diagnostic information in case of multi-conditions filter (MPL_STD_INI)

Formula section :

LV_CDN_DIAG_XX = 1
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0
CTR_ABC_XX = 0
LV_END_DIAG_XX = 1
CTR_ABC_END_DIAG_XX = C_ABC_MAX_XX

A.4.3 Description for ACTION_ERRM_MultiFilterReset:

Syntax: ACTION_ERRM_MultiFilterReset(IN<XX>, OUT<LV_ERR_XX>)

Parameter(in): XX Type: index to identify the Diagnostic

Parameter(out): LV_ERR_XX Type: Boolean

Short Description: This action resets data filter in case of Multiconditions filter type

Formula section :

LV_CDN_DIAG_XX = 0
ERR_SYM_XX = NO_SYM
LV_ERR_XX = 0
CTR_ABC_XX = 0
LV_END_DIAG_XX is unchanged
CTR_ABC_END_DIAG_XX is unchanged

A.4.4 Synchronization between multi-condition algorithm & Error Management

General information:

This section describes the mechanism of synchronisation of the failure filtering with the Error Management core. The principle is to call the Error Management upon events.

LV_END_DIAG_XX 0 → 1 transition
LV_ERR_XX 0 → 1 transition
LV_ERR_XX 1 → 0 transition

A.4.5 Mechanism for CPU load optimisation

General information:

This section describes the mechanism of optimisation. The principle is to suspend the filter algorithm execution when is not necessary. When the bench mode is used for a diagnostic XX the mechanism of CPU load optimisation is disabled for this instance.

Formula section:

If LV_ERR_XX = 0
and LV_END_DIAG_XX = 1
and ERR_SYM_XX = 0
and CTR_ABC_XX = 0
Then
It is not necessary to call error management (including filtering)
LV_END_DIAG_XX = 1 {unchanged}
LV_ERR_XX = 0 {unchanged}
Endif

A.4.6 Configuration data

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_SYM_XX	1	1...FH	1...15	1	[-]
Available symptom(s) for XX diagnostic					

Configuration data detailed description:

Symptoms 0 and 1 available: 03H
Symptoms 0, 1 and 2 available: 07H (typical value)
Symptoms 0, 1, 2 and 3 available: 0FH

Predefined filter type:

	NLC_ABC_STC_XX	NLC_ABC_CAL_DEC_XX	NLC_ABC_NOT_DEC_XX	NLC_ABC_INI_VALUE_XX	NLC_ABC_INI_IGK_XX	NLC_ABC_NOT_SAVE_XX
MPL_STD_INI	0	0	0	0	1	1

The **standard configuration with initialisation (MPL_STD_INI)** is NC_ABC_CONF_FCT_DIAG_XX = 3H (default configuration)

The data CTR_ABC_XX and LV_ERR_XX are Reset at the transition LV_IGK 0 → 1 and Reset (NLC_ABC_NOT_SAVE_XX = 1, NLC_ABC_INI_IGK_XX = 1) and the anti-bounce counter is initialised to 0 value (NLC_ABC_INI_VALUE_XX = 0).

The anti-bounce counter may be decremented (NLC_ABC_NOT_DEC_XX = 0).
The decrement value is equal to 1 (NLC_ABC_CAL_DEC_XX = 0).
The filter used is the Anti-bounce (NLC_ABC_STC_XX = 0).

A.5 Upstream oxygen sensor heater OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_LSH_UP[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Error flag for upstream heater diagnosis OBD I after debouncing					
LV_CDN_DIAG_LSH_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Diagnosis condition for upstream heater diagnosis OBD 1I					
LV_END_DIAG_LSH_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
End of diagnosis for upstream heater diagnosis OBD I					
LV_LSH_SCG_ACT_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that debounce for upstream heater OBDI SCG diagnosis active					
ERR_DIAG_LSH_UP[NC_CBK_EX_NR]	V	0...7H	0...7	1	-
Raw value of error symptom for upstream heater diagnosis OBD I					
ERR_SYM_LSH_UP[NC_CBK_EX_NR]	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom for upstream heater diagnosis OBD I filterd with CDN_DIAG_LSH_UP					
CDN_DIAG_LSH_UP[NC_CBK_EX_NR]	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of upstream heater diagnosis OBD I					

Input data:

LV_ST_END	STATE_LSH_UP[NC_CBK_EX_NR]	**LV_VB_CDN_OBD_1**	TEG_DYN
LSHPWM_UP[NC_CBK_EX_NR]	LV_LSH_OC_LSH_UP[NC_CBK_EX_NR]	LV_LSH_SCP_LSH_UP[NC_CBK_EX_NR]	LV_LSH_SCG_LSH_UP[NC_CBK_EX_NR]
LV_INH_DIAG_LSH_UP[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_IGK	LV_DC

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank.

All symptoms of the current error code are handeled by anti-bouncing.
We assume that only one symptom of an error code can be active at the same time.

Description:

The error detection is effected via the ECU hardware.

The purpose of the diagnosis shall be to detect electrical faults within the oxygen sensor heater circuit as defined by OBD I requirements.

The function is intended for use in a hardware configuration where the heater driver shall carry out the electrical fault diagnosis and provide the electrical status via status flags.

- *Short circuit to Vbatt by Over-temperature* ***or*** *Overcurrent* (**SCP**)
- *Short circuit to GND* (SCG)
- *Open load* (**OC**)

The electrical status flags for each heater driver shall be read out via SPI and decoded by the appropriate function typically every 10 ms and made available to the Application SW in the form of driver and fault specific flags. The flags shall only be set by the appropriate function and may only be reset by the diagnosis function. These flags are:

- **SCP** LV_LSH_SCP_LSH_UP[i]
- SCG LV_LSH_SCG_LSH_UP[i]
- **OC** LV_LSH_OC_LSH_UP[i]

The heater drivers named are capable of detecting an **OC** and **SCP** fault when in the ON state and only capable of detecting a SCG fault in the OFF state. As the heater power is controlled by a PWM signal, the driver will be placed alternately in an ON and then OFF state.

The diagnosis shall be activated should the following conditions be met:

- Engine leaving the engine start state as denoted by LV_ST_END = 1
- Battery voltage falls between pre-set thresholds to protect against false errors at battery voltage extremes as indicated by **LV_VB_CDN_OBD_1**.
- LSHPWM signal greater or equal to or lower or equal to specific calibrateable value. Should the PWM be set to 100 %, no SCG fault can be recognised (e.g. for ATIC39 & ATM3x min. of 250 µs required). Similarly for the **OC** / **SCP** fault, a fault can only be detected when the ON time is longer than a minimum threshold (e.g. for ATM3x for **OC** min. 250 µs). The exact HW thresholds are dependent upon the characteristics of the external circuitry.
- Inhibit flag not set (LV_INH_DIAG_LSH_UP[i])

In exceptionally rare extreme environmental conditions for certain system configurations, it is possible that the heater driver may be driven into over-temperature during initial oxygen sensor heating. In this case, the driver automatically turns off the output stage in self-protection and sets the **SCP** electrical fault bit. In order to mask an incorrect oxygen sensor heater **SCP** fault, the function shall only allow the **SCP** to set a fault during the initial heater phase, if the exhaust gas temperature has exceeded a pre-determined calibrateable threshold.

The oxygen sensor heater fault detection shall be debounced by use of an anti-bounce counter. This shall be handled by the Dynamic Error Management.

Each of the above listed faults shall be described individually below:

Otherwise the diagnosis has to be suppressed (CDN_DIAG_LSH_UP[i] = 0). This is managed within the “Diagnosis condition calculation.

Application conditions:

Activation : every engine state

Deactivation: -

Initialization: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at Reset

NOTE: The input flags LV_LSH_SCP_LSH_UP[i], LV_LSH_SCG_LSH_UP[i] & LV_LSH_OC_LSH_UP[i] shall be initialised to 0 at the beginning of a new driving cycle by the appropriate BSW function.

Recurrence: 1 s

NOTE: The heater PWM frequency shall be at least 2 times faster than the diagnosis frequency to ensure that an oversampling effect does not occur!

Formula section:

CDN_DIAG_LSH_UP[i] = 0

If LV_ST_END = 1
and **LV_VB_CDN_OBD_1** = 1
and LV_INH_DIAG_LSH_UP[i] = 0
then **If** LSHPWM_UP[i] ≥ C_LSHPWM_MIN_DIAG_LSH_UP
and [(STATE_LSH_UP[i] <> LSH_POW_RISE
and STATE_LSH_UP[i] <> LSH_POW_RED)
or TEG_DYN ≥ C_TEG_DYN_MIN_DIAG_LSH_UP]
then bit 0 of CDN_DIAG_LSH_UP[i] = 1 *(**SCP** can be detected))*
endif
If LSHPWM_UP[i] ≤ C_LSHPWM_MAX_DIAG_LSH_UP
then bit 1 of CDN_DIAG_LSH_UP[i] = 1 *(**SCG** can be detected)*
endif
If C_LSHPWM_MIN_DIAG_LSH_UP ≤ LSHPWM_UP[i]
≤ C_LSHPWM_MAX_DIAG_LSH_UP
then bit 2 of CDN_DIAG_LSH_UP[i] = 1 *(**OC** can be detected)*
endif
endif

Error Symptom calculation (raw value from I/O SW)

ERR_DIAG_LSH_UP[i] = 0

If	LV_LSH_SCP_LSH_UP[i] = 1
then	bit 0 of ERR_DIAG_LSH_UP[i] = 1
endif	
If	LV_LSH_SCG_LSH_UP[i] = 1
then	bit 1 of ERR_DIAG_LSH_UP[i] = 1
	LV_LSH_SCG_ACT_LSH_UP[i] = 1
endif	
If	*abc-counter* = 0
then	LV_LSH_SCG_ACT_LSH_UP[i] = 0
endif	
If	LV_LSH_OC_LSH_UP[i] = 1
then	bit 2 of ERR_DIAG_LSH_UP[i] = 1
endif	

For failure and error management treatment the anti-bounce mechanism is called with the parameters CDN_DIAG_LSH_UP[i] and ERR_DIAG_LSH_UP[i].

This algorithm determines:

ERR_SYM_LSH_UP[i]	(= raw value ERR_DIAG_LSH_UP[i] filtered with CDN_DIAG_LSH_UP[i], detected error symptom for lambda sensor heater upstream [i] diagnosis)
LV_ERR_ LSH_UP[i]	(Error flag for debounced error of for lambda sensor heater upstream [i])
LV_CDN_DIAG_ LSH_UP[i]	(Diagnosis condition information)
LV_END_DIAG_ LSH_UP[i]	(End of diagnosis information)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LSHPWM_MIN_DIAG_LSH_UP	1	0...FFH	0...99.609	0.3891	%
Minimum O2-Sensor heater duty cycle to perform the electrical diagnosis					
C_LSHPWM_MAX_DIAG_LSH_UP	1	0...FFH	0...99.609	0.3891	%
Maximum O2-Sensor heater duty cycle to perform the electrical diagnosis					
C_TEG_DYN_MIN_DIAG_LSH_UP	1	0...7FF0H	0...2047	0.0625	°C
Minimum exhaust gas temperature threshold to permit **SCP** to set fault during LSH power rise state					

Configuration data:

Diagnosis	Symptom	Nr	ABC type
Upstream oxygen sensor heater OBDI diagnosis	*Short circuit to battery*	*0*	MPL_STD_INI
	Short circuit to GND	*1*	
	Open circuit	*2*	
LSH_UP[i]			

A.5.1 Application incidences for upstream oxygen sensor heater OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Inhibit flag for upstream oxygen sensor heater OBDI diagnosis					

Input data:

LV_ST_END	NC_CBK_EX_NR	NLC_LSH_RLY_EFP	LV_EFP
LV_ERR_RLY_EFP	CONF_LAM		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks. In case of two separate catalyst systems (NC_CBK_EX_NR = 2) i = 1 for exhaust bank 1, i = 2 for exhaust bank 2.

Description:

In some systems the supply voltage of the O2-Sensor heater may be connected to the electrical fuel pump relay (NLC_LSH_RLY_EFP = 1). In this case, if the electrical fuel pump is switched off or if the fuel pump relay is defective, no heating would occur and the heater driver will recognize an incorrect heater fault. Hence the OBD I heater diagnosis shall be inhibited if LV_EFP = 0 or if LV_ERR_RLY_EFP = 1.

Application conditions:

Initialisation: LV_INH_DIAG_LSH_UP[i] shall be reset at transition LV_ST_END 0 → 1

Recurrence: The function shall be carried out once every 1 s.

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Formula section:

If NLC_LSH_RLY_EFP = 1
then *(O2-Sensor Heater supplied via EFP relay)*
  If LV_EFP = 0 *(EFP relay not active)*
  or LV_ERR_RLY_EFP = 1 *(error at EFP relay)*
  or CONF_LAM = 0 *(open loop lambda control)*
  then LV_INH_DIAG_LSH_UP[i] = 1
  else LV_INH_DIAG_LSH_UP[i] = 0
else *(O2-Sensor Heater supplied via EFP relay)*
  if CONF_LAM = 0 *(open loop lambda control)*
  then LV_INH_DIAG_LSH_UP[i] = 1
  else LV_INH_DIAG_LSH_UP[i] = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_DIAG_LSH_UP	1	0...FFH	0...255	1	-
Anti-bounce counter increment for upstream O2-sensor heater diagnosis.					
C_ABC_MAX_DIAG_LSH_UP	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for upstream O2-sensor heater diagnosis.					

Configuration for diagostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Upstream oxygen sensor heater OBDI diagnosis Bank 1, Sensor 1	*Short circuit to battery*	SYM_0	*STD*
	Short circuit to GND	SYM_1	
	Open circuit	SYM_2	
LSH_UP_1		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
LSHPWM_UP_1
R_IT_LS_UP_1_ENVD_H
R_IT_LS_UP_1_ENVD_L

Application assistances:

Recommendation: The Application assistances shall be maintained for the remaining duration of the driving cycle.

- Evaporative emission system
  Control - Activation of Minimum Operation
  System diagnosis - Inhibit
- Lambda control (Only for exhaust gas temperatures lower than O2 sensor lowest operating temperature)
  Control - Return to open-loop mode (Bank selectively)
  NOTE: Closed loop control may be permitted if the signal voltage excursion is large enough
  Adaptation - Inhibit (Bank selectively)
  Downstream trim regulation - Inhibit (Bank selectively)
- Oxygen sensor / Lambda controller diagnosis:
  O2 sensor upstream voltage excursion diagnosis - Inhibit (Bank selectively)
  O2 sensor upstream response time diagnosis - Inhibit (Bank selectively)
  O2 sensor control limit diagnosis - Inhibit (Bank selectively)
  O2 sensor frequency diagnosis - Inhibit (Bank selectively)
  O2 sensor upstream signal voltage diagnosis (Bank selectively)
  O2 sensor upstream signal plausibility diagnosis (Bank selectively)
  O2 sensor downstream response time diagnosis (Bank selectively)
  O2 sensor downstream signal voltage diagnosis (Bank selectively)
  O2 sensor downstream PUC end diagnosis (Bank selectively)
- Idle speed adaptation - Inhibit
- Knock adaptation - Inhibit
- MAF / TPS-Plausibility diagnosis - Inhibit
- Fuel system diagnosis - Inhibit (indirectly via LV_LSCL)
- Catalyst efficiency diagnosis - Inhibit (Bank selectively) (indirectly via LV_LSCL)
- O2 sensor upstream heater OBD II diagnosis - Inhibit (Bank selectively)

A.6 Downstream oxygen sensor signal monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]	V/O	0...01H	0...1	1	-
Boolean error flag, electrical fault currently present on downstream oxygen sensor signal					

Input data:

LV_ERR_SCG_LS_DOWN[NC_CBK_EX_NR]	LV_IGK	LV_ERR_SCP_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_OC_LS_DOWN[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

According the exhaust gas system shall the variables be defined for [i]=1 to NC_CBK_EX_NR.

The VLS_DOWN[i] value used for diagnosis purposes shall not be affected should the sensor be “pumped”. The value prior to pumping shall be used until the sensor returns to normal operation (no pumping).
Once an electrical fault has been detected and debounced, the plausibility diagnosis shall not be carried out until the electrical fault has been diagnosed as being no longer present. The electrical diagnosis shall remain unaffected by the state of the plausibility diagnosis.
Once an electrical fault has been diagnosed and debounced, the application assistances (Functions that are defined to be affected by the state of the oxygen sensor signal) shall remain active for the remainder of the driving cycle. The restarting of the affected functions during the next driving cycle shall be carried out according to the project philosophy.
The measurement of the oxygen sensor voltage shall not be affected by the detection of a signal fault unless explicitly deactivated.

Description:

The oxygen sensor signal electrical diagnosis shall detect the following faults:

A. Short circuit of the oxygen sensor signal line to GND
B. Short circuit of the oxygen sensor signal line to Vbatt
C. Open circuit (line break) in the connection to oxygen sensor element

Each fault may set the boolean flag (LV_ERR_SCG_LS_DOWN[i]; LV_ERR_SCP_LS_DOWN[i]; LV_ERR_OC_LS_DOWN[i]) to signal that the particular fault is currently active. Should one of the electrical fault flags (A-C) be set, then a global error flag (LV_ERR_EL_LS_DOWN[i]) shall be set to indicate that an electrical fault currently exists on the particular oxygen sensor.
Each of the above listed erro shall be described individually in further detail below.

Application conditions:

Initialisation: all variables shall be initialised to 0 with LV_IGK = 0 -> 1 or reset or at clearing the error memory

Recurrence: 100ms

Activation:

If LV_IGK = 1

Deactivation:

If LV_IGK = 0

Formula section:

If LV_ERR_SCG_LS_DOWN[i] = 1
or LV_ERR_SCP_LS_DOWN[i] = 1
or LV_ERR_OC_LS_DOWN[i] = 1
then LV_ERR_EL_LS_DOWN[i] = 1
else LV_ERR_EL_LS_DOWN[i] = 0
endif

A.6.1 Short circuit of oxygen sensor signal to GND.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SCG_LS_DOWN[NC_CBK_EX_NR]	V/O	0...01H	0...1	1	-
Boolean error flag, short to ground fault currently present on downstream oxygen sensor signal					
ERR_SYM_SCG_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 2H	NO_SYM SYM_1	1	-
Boolean symptom flag, short to ground fault currently present on downstream oxygen sensor signal					
LV_CDN_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor short to ground downstream diagnosis					
LV_END_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of short to ground downstream oxygen sensor signal diagnosis					
T_ACT_PLS_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...6553.5	0.1	s
Delay time for final diagnosis activation					
CTR_SYM_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
Detection time for SCG					
LV_DIAG_ACT_REQ_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Request for starting the downstream sensor active plausibility check					
CTR_R_UPD_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Value of handshake counter for last valid resistance value used for short circuit to ground diagnosis					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_DC	LV_IGK	LV_INH_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]
R_IT_LS_DOWN[NC_CBK_EX_NR]	CTR_CYCNR_R_IT_LS_DOWN_VLD[NC_CBK_EX_NR]		MAF_KGH

Description:

According the exhaust gas system shall the variables be defined for [i]=1 to NC_CBK_EX_NR.

In order to determine the electrical fault, short circuit to GND, the oxygen sensor signal voltage is compared to a calibrateable threshold (C_VLS_THD_DIAG_SCG_LS_DOWN) in conjunction with certain activation conditions.
Should a hard short to GND (i.e. approx. 0 Ω) exist on the oxygen sensor signal line, the signal voltage will tend to 0 V. This condition may also occur should the air-fuel mixture (AFR) become lean due to a fault in the injection system, due to an air leak, intentional or not, or due to a normal operating conditions. Hence the following additional fault conditions shall be required:
The ignition key shall be determined to be on (LV_IGK = 1) and the appropriate inhibit flag shall not be set (LV_INH_DIAG_SCG_LS_DOWN[i]).

Each time that timer T_ACT_PLS_DIAG_SCG_LS_DOWN[i] exceeded the minimum duration (C_T_DLY_DIAG_SCG_LS_DOWN) for the first time, e.g. first time after engine start or after timer T_ACT_PLS_DIAG_SCG_LS_DOWN[i] has been reset, the

CTR_R_UPD_DIAG_SCG_LS_DOWN[I] shall be updated once only with the value CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. This shall ensure that only new internal resistance values be used for diagnosis purposes that have been determined since all the activation conditions have been met.
It shall be determined whether a new internal resistance has been determined by comparing the contents of counter CTR_R_UPD_DIAG_SCG_LS_DOWN[I] with that of CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. Should the counters be unequal, then a new value is available and CTR_R_UPD_DIAG_SCG_LS_DOWN[I] shall be updated with CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. If this conditions and all the above conditions be met, the diagnosis shall be considered to be active.
Should the diagnosis be determined to be active, the oxygen sensor signal voltage shall be compared to the threshold. Should the voltage fall below the threshold and the resistance condition R_IT_LS_DOWN[i] less than threshold be met, a pre - detection counter is incremented. The counter is reseted to 0 if the above mentioned conditions are no more fulfilled. If the pre – detection counter counter reached the maximum value then the active check is triggered or if finished the error “Short to GND” is detected.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 with LV_IGK = 0 -> 1 or reset or at clearing the error memory.

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

LV_DIAG_ACT_REQ_LS_DOWN[i] was deleted from the formula section but it should be initialised by 0 and held as output.

Recurrence: T_SAMPLE = 100ms

Activation / Deactivation:

If LV_IGK = 1

and LV_INH_DIAG_SCG_LS_DOWN[i] = 0

and MAF_KGH > C_MAF_KGH_MIN_DIAG_SCG_LS_DOWN

then If T_ACT_PLS_DIAG_SCG_LS_DOWN[i] ≥ C_T_DLY_DIAG_SCG_LS_DOWN

then *One-off update* CTR_R_UPD_DIAG_SCG_LS_DOWN[I]

If CTR_CYCNR_R_IT_LS_DOWN_VLD[i] ≠ CTR_R_UPD_DIAG_SCG_LS_DOWN[I]

then *“Diagnosis available”*

LV_CDN_DIAG_SCG_LS_DOWN[i] = 1

CTR_R_UPD_DIAG_SCG_LS_DOWN[I] = CTR_CYCNR_R_IT_LS_DOWN_VLD[i]

else *“Diagnosis NOT available”*

LV_CDN_DIAG_SCG_LS_DOWN[i] = 0

endif

```
        else  LV_CDN_DIAG_SCG_LS_DOWN[i] = 0
              T_ACT_PLS_DIAG_SCG_LS_DOWN[i] =
                                  T_ACT_PLS_DIAG_SCG_LS_DOWN[i] + T_SAMPLE
        endif
   else "Diagnosis NOT available"
              T_ACT_PLS_DIAG_SCG_LS_DOWN[i] = 0
              CTR_SYM_DIAG_SCG_LS_DOWN[I] = 0
              LV_CDN_DIAG_SCG_LS_DOWN[i] = 0
   endif
```

Formula section:

```
If            LV_CDN_DIAG_SCG_LS_DOWN[i] = 1
Then If       VLS_DOWN[i] < C_VLS_THD_DIAG_SCG_LS_DOWN
         and  R_IT_LS_DOWN[i] < C_R_IT_MAX_DIAG_SCG_LS_DOWN
         then IfCTR_SYM_DIAG_SCG_LS_DOWN[I]≥ C_CTR_MIN_DIAG_SCG_LS_DOWN
              then
                          ERR_SYM_SCG_LS_DOWN[i] = SYM_1
                          % Error management automatically increment ABC counter

              else    CTR_SYM_DIAG_SCG_LS_DOWN[I] =
                       CTR_SYM_DIAG_SCG_LS_DOWN[I] +1
                      ERR_SYM_SCG_LS_DOWN[i] = NO_SYM
                      % Error management automatically decrement ABC counter
              endif
          else LV_DIAG_ACT_REQ_LS_DOWN[I] = 0
              % Error management automatically decrement ABC counter
              ERR_SYM_SCG_LS_DOWN[i] = NO_SYM
              CTR_SYM_DIAG_SCG_LS_DOWN[I] = 0
          endif
endif
```

Calibration data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
C_VLS_THD_DIAG_SCG_LS_DOWN	1	0...3FFH	0...4.995309	4.883 e-3	V
Oxygen sensor voltage threshold under which short to GND may be present					
C_R_IT_MAX_DIAG_SCG_LS_DOWN	1	0...FFFFH	0...65535	1	Ohm
Maximum resistance threshold oxygen sensor internal resistance during short to ground sensor					
C_CTR_MIN_DIAG_SCG_LS_DOWN	1	0...FFH	0...255	1	-
Minimum duration required for continious low VLS_DOWN[i] detection					
C_T_DLY_DIAG_SCG_LS_DOWN	1	0...FFFFH	0...6553.5	0.1	s
Minimum duration required for low VLS_DOWN[i] detection					
C_MAF_KGH_MIN_DIAG_SCG_LS_DOWN	1	0...FFFFH	0...2047.9687	31.25e-3	kg/h
Minimum MAF_KGH flow required for continious low VLS_DOWN[i] detection					

Diagnosis	Symptom	Nr	ABC type
Downstream Oxygen Sensor Signal	Sym_1	*0*	MEM
SCG_LS_DOWN[i]			

A.6.2 Short circuit of oxygen sensor signal to VBATT

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SCP_LS_DOWN[NC_CBK_EX_NR]	V/O	0...01H	0...1	1	-
Boolean error flag, short to plus fault currently present on downstream oxygen sensor signal					
ERR_SYM_SCP_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H	NO_SYM SYM_0	1	-
Boolean symptom flag, short to plus fault currently present on downstream oxygen sensor signal					
LV_CDN_DIAG_SCP_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor short to plus downstream diagnosis					
LV_END_DIAG_SCP_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of short to plus downstream oxygen sensor signal diagnosis					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_INH_DIAG_SCP_LS_DOWN[NC_CBK_EX_NR]	LV_IGK	LV_DC

Description:

According the exhaust gas system shall the variables be defined for [i]=1 to NC_CBK_EX_NR.

Should a hard short to Vbatt (i.e. approx. 0 Ω) exist on the oxygen sensor signal line, the signal voltage will tend to Vbatt and the analogue digital converter value will tend to 5 V. This effect does not occur under normal conditions as the sensor delivers typically 1 V for rich AFR at the sensor location and may be used for short to Vbatt detection.

In order to determine the electrical fault, short circuit to Vbatt, the oxygen sensor signal voltage (VLS_DOWN[i]) is compared to a calibrateable threshold (C_VLS_THD_DIAG_SCP_LS_DOWN) in conjunction with the herein mentioned activation conditions.
The ignition key shall be determined to be on (LV_IGK = 1) and the appropriate inhibit flag shall not be set (LV_INH_DIAG_SCP_LS_DOWN[i]).
Should the oxygen signal voltage exceed the said threshold, a signal short to Vbatt shall be determined to be present and the debounce counter shall be incremented otherwise the debounce counter shall be decremented.
The error flag LV_ERR_SCP_LS_DOWN[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_SCP_LS_DOWN) shall be initialised to 0 with LV_IGK = 0 -> 1 or reset or at clearing the error memory

LV_END_DIAG_SCP_LS_DOWN is initialised with 0 at every (LV_IGK = 0 ->1 and LV_DC = 0) or reset or at clearing the error memory

Recurrence: 100ms

Activation / Deactivation:

If LV_IGK = 1
and LV_INH_DIAG_SCP_LS_DOWN[i] = 0
then ***“Diagnosis available”***
LV_CDN_DIAG_SCP_LS_DOWN[i] = 1
else ***“Diagnosis NOT available”***
LV_CDN_DIAG_SCP_LS_DOWN[i] = 0
endif

Formula section:

If LV_CDN_DIAG_SCP_LS_DOWN[i] = 1
then **If** VLS_DOWN[i] > C_VLS_THD_DIAG_SCP_LS_DOWN
then ***“Short to Vbatt fault present”***
% Error management automatically increment ABC counter
ERR_SYM_SCP_LS_DOWN[i] = SYM_0 *%SHORT TO PLUS*

else ERR_SYM_SCP_LS_DOWN[i] = "NO_SYM"
% Error management automatically decrement ABC counter
endif
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_VLS_THD_DIAG_SCP_LS_DOWN	1	0...3FFH	0...4.995309	4.883 e-3	V
Oxygen sensor voltage threshold over which short to Vbatt present					

Diagnosis	Symptom	Nr	ABC type
Downstream Oxygen Sensor Signal	Sym_0	*0*	MEM
SCP_LS_DOWN[i]			

A.6.3 Open circuit (line break) in connection to oxygen sensor element.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_OC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...01H	0...1	1	-
Boolean error flag, open circuit fault currently present on downstream oxygen sensor signal					
ERR_SYM_OC_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 4H	NO_SYM SYM_2	1	-
Variable indicating an open circuit fault is currently present on downstream oxygen sensor signal					
LV_T_CDN_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Status conditions of VLS range for the downstream oxygen sensor open circuit diagnosis					
LV_R_IT_VLD_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Valid conditions of resistance of downstream oxygen sensor open circuit diagnosis					
LV_CDN_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of downstream oxygen sensor open circuit diagnosis					
LV_END_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of downstream oxygen sensor open circuit diagnosis					
T_ACT_PLS_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0..6553.5	0.1	s
Delay time for final diagnosis activation					
T_CDN_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0..6553.5	0.1	s
Detection time for status OC before diagnosis					
CTR_R_UPD_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Value of handshake counter for last valid resistance value used for diagnosis					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]	LV_IGK	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]
STATE_LSH_DOWN[NC_CBK_EX_NR]	LV_INH_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	CTR_CYCNR_R_IT_LS_DOWN_VLD[NC_CBK_EX_NR]	R_IT_LS_DOWN[NC_CBK_EX_NR]
LV_LS_DOWN_READY[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_DC	

Description:

According the exhaust gas system shall the variables be defined for [i]=1 to NC_CBK_EX_NR.

Should a complete open circuit occur in the oxygen sensor signal or return line, the measured voltage at the input to the microcontroller will tend to the voltage determined by the operative readiness potential divider (approx. 450 mV) and the measured internal resistance will be abnormally high.
In order to determine an open circuit electrical fault, the oxygen sensor internal resistance (R_IT_LS_DOWN[i]) is compared to a calibrateable threshold (C_R_IT_MIN_DIAG_OC_LS_DOWN) in conjunction with certain activation conditions shown below:
The ignition key shall be determined to be on (LV_IGK = 1) and the appropriate inhibit flag shall not be set (LV_INH_DIAG_OC_LS_DOWN[i]).

The sensor shall be determined to be in a state of operative readiness (LV_LS_DOWN_READY[i] = 1, possibly thorough forced activation) and the temperature operating conditions of the sensor shall be determined to be sufficient to permit normal operation, i.e. No fault shall be determined to be present in the appropriate downstream oxygen sensor heater (LV_ERR_LSH_DOWN[i] = 0) or if so that the operating temperature be determined to be sufficient for sensor operability (TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_MIN_DIAG_OC_LS_DOWN), and the appropriate heater shall be determined to being controlled normally (STATE_LSH_DOWN[i] = LSH_POW_CTL).
Should the above initial conditions be met, a timer (T_ACT_PLS_DIAG_OC_LS_DOWN[i]) shall be incremented. Should the conditions no longer be met, both timers shall be reset and flag LV_T_CDN_DIAG_OC_LS_DOWN[i] reset.
The oxygen sensor signal shall be determined to lie in a calibrateable voltage band (C_VLS_AFL_THD_DIAG_OC_LS_DOWN < VLS_DOWN[i] < C_VLS_AFR_THD_DIAG_OC_LS_DOWN).
Should the above listed pre-activation conditions be determined to have been met, a timer T_CDN_DIAG_OC_LS_DOWN[i] shall be started. The timer shall be required to equal or exceed a minimum calibrateable duration (C_T_CDN_DIAG_OC_LS_DOWN). When no longer met, T_CDN_DIAG_OC_LS_DOWN[i] and LV_T_CDN_DIAG_OC_LS_DOWN[i] shall be reset.
Each time that the T_CDN_DIAG_OC_LS_DOWN[i] has exceeded the minimum duration for the first time, e.g. first time after engine start or after the T_CDN_DIAG_OC_LS_DOWN[i] has been reset, the CTR_R_UPD_DIAG_OC_LS_DOWN[i] shall be updated once only with the value CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. This shall ensure that only new internal resistance values be used for diagnosis purposes that have been determined since all the activation conditions have been met.
It shall be determined whether a new internal resistance has been determined by comparing the contents of counter CTR_R_UPD_DIAG_OC_LS_DOWN[i] with that of CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. Should the counters be unequal, then a new value is available and CTR_R_UPD_DIAG_OC_LS_DOWN[i] shall be updated with CTR_CYCNR_R_IT_LS_DOWN_VLD[i] and a flag LV_R_IT_VLD_DIAG_OC_LS_DOWN[I] be set, thus indicating that the voltage lies within the set band and that a new resistance value is available.
If either LV_T_CDN_DIAG_OC_LS_DOWN[i] is set or T_ACT_PLS_DIAG_OC_LS_DOWN[i] equals or exceeds threshold C_T_DLY_DIAG_OC_LS_DOWN, the diagnosis shall be considered to be available.
Should the diagnosis be determined to be available and the condition LV_R_IT_VLD_DIAG_OC_LS_DOWN[I] has been met, the oxygen sensor internal resistance shall be compared to the said threshold. Should the resistance exceed the threshold, a signal open circuit shall be determined to be present and the debounce counter shall be incremented otherwise the debounce counter shall be decremented.
The error flag LV_ERR_OC_LS_DOWN[i] shall be set once the anti-bounce counter has reached its maximum and reset should the anti-bounce counter reach 0.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 with LV_IGK = 0 -> 1 or reset or at clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence: T_SAMPLE = *100ms*

Activation / Deactivation:

```
If      LV_IGK = 1
and     LV_INH_DIAG_OC_LS_DOWN[i] = 0
and     LV_LS_DOWN_READY[i] = 1
and     (LV_ERR_LSH_DOWN[i] = 0
or      TEG_CAT_DOWN_MDL[i] > C_TEG_CAT_MIN_DIAG_OC_LS_DOWN)
and     STATE_LSH_DOWN[i] = LSH_POW_CTL
then
        T_ACT_PLS_DIAG_OC_LS_DOWN[i] = T_ACT_PLS_DIAG_OC_LS_DOWN[i] +
                                                            T_SAMPLE

    One-off update CTR_R_UPD_DIAG_OC_LS_DOWN[i]
    If CTR_CYCNR_R_IT_LS_DOWN_VLD[i] ≠ CTR_R_UPD_DIAG_OC_LS_DOWN[i]
    then CTR_R_UPD_DIAG_OC_LS_DOWN[i]
                           = CTR_CYCNR_R_IT_LS_DOWN_VLD[i]
         LV_R_IT_VLD_DIAG_OC_LS_DOWN[I] = 1
    else LV_R_IT_VLD_DIAG_OC_LS_DOWN[I] = 0
    endif
    If   C_VLS_AFL_THD_DIAG_OC_LS_DOWN < VLS_DOWN[i]
              < C_VLS_AFR_THD_DIAG_OC_LS_DOWN
    then
         If   T_CDN_DIAG_OC_LS_DOWN[i] ≥ C_T_CDN_DIAG_OC_LS_DOWN
         then LV_T_CDN_DIAG_OC_LS_DOWN[i] = 1
         else LV_T_CDN_DIAG_OC_LS_DOWN[i] = 0
              T_CDN_DIAG_OC_LS_DOWN[i] =
                        T_CDN_DIAG_OC_LS_DOWN[i] + T_SAMPLE
         endif
    else T_CDN_DIAG_OC_LS_DOWN[i] = 0
         LV_T_CDN_DIAG_OC_LS_DOWN[i] = 0
    endif
else    T_CDN_DIAG_OC_LS_DOWN[i] = 0
        T_ACT_PLS_DIAG_OC_LS_DOWN[i] = 0
        LV_R_IT_VLD_DIAG_OC_LS_DOWN[i] = 0
        LV_T_CDN_DIAG_OC_LS_DOWN[i] = 0
endif
If      (T_ACT_PLS_DIAG_OC_LS_DOWN[i] ≥ C_T_DLY_DIAG_OC_LS_DOWN
```

or LV_T_CDN_DIAG_OC_LS_DOWN[i] = 1)

and LV_R_IT_VLD_DIAG_OC_LS_DOWN[i] = 1

then ***"Diagnosis available"***

LV_CDN_DIAG_OC_LS_DOWN[i] = 1

else ***"Diagnosis NOT available"***

LV_CDN_DIAG_OC_LS_DOWN[i] = 0

endif

NOTE: The variable CTR_CYCNR_R_VLS_DOWN_DIAG_i shall be updated with the current value of the input variable CTR_CYCNR_R_IT_LS_DOWN_VLD[i] on each occasion that the TIMER_3 exceeds the minimum for the first time. For further function calls where the TIMER_3 condition is met, the one-off update shall not be carried out. This shall continue until the TIMER_3 condition is no longer met.

Formula section:

If LV_CDN_DIAG_OC_LS_DOWN[i] = 1

then **If** R_IT_LS_DOWN[i] > C_R_IT_MIN_DIAG_OC_LS_DOWN

and LV_T_CDN_DIAG_OC_LS_DOWN[i] = 1

then ***"Open circuit fault present"***

% Error management automatically iccrement ABC counter

ERR_SYM_OC_LS_DOWN[i] = sym_2 %"OPEN LINE"

else ERR_SYM_OC_LS_DOWN[i] = "NO_SYM"

% Error management automatically decrement ABC counter

endif

endif

Calibration data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
C_R_IT_MIN_DIAG_OC_LS_DOWN	1	0...FFFFH	0...65535	1	Ohm
Minimum resistance threshold oxygen sensor internal resistance during open circuit sensor					
C_T_CDN_DIAG_OC_LS_DOWN	1	0...FFFFH	0...6553.5	0.1	s
Minimum required duration where voltage lies outside of the limits for operable sensor to detect fault					
C_T_DLY_DIAG_OC_LS_DOWN	1	0...FFFFH	0...6553.5	0.1	s
Minimum required duration where voltage lies outside of the limits for operable sensor to detect fault					
C_TEG_CAT_MIN_DIAG_OC_LS_DOWN	1	0...7FF0H	0...2047	0.0625	°C
Minimum requested exhaust gas temperature for diagnosis					
C_VLS_AFL_THD_DIAG_OC_LS_DOWN	1	0...3FFH	0...4.995309	4.883 e-3	V
Oxygen sensor voltage-lean mixture threshold for diagnosis					
C_VLS_AFR_THD_DIAG_OC_LS_DOWN	1	0...3FFH	0...4.995309	4.883 e-3	V
Oxygen sensor voltage-rich mixture threshold for diagnosis					

Diagnosis	Symptom	Nr	ABC type

Downstream Oxygen Sensor Signal	SYM_2	*0*	MEM
OC_LS_DOWN[I]			

A.7 Application Incidences for downstream O2-Sensor Diagnosis

A.7.1 Calculation of inhibition for diagnosis and for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_OC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for downstream open line diagnosis not met					
LV_INH_DIAG_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for downstream PUC diagnosis not met					
LV_INH_DIAG_FL_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for downstream FL diagnosis not met					
LV_INH_DIAG_SCG_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for downstream SCG diagnosis not met					
LV_INH_DIAG_SCP_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for downstream short to plus diagnosis not met					
LV_INH_DIAG_RBM_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating application conditions for RBM of downstream PUC diagnosis not met					

Input data:

LV_IGK	LV_ST_END	NC_CBK_EX_NR	LV_MIS_STATE_B1
LV_MIS_STATE_B4	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]	LV_ERR_EL_CPS	LV_ERR_CRK
LV_ERR_MEC_OPEN_CPS	LV_ERR_MAF	STATE_ERR_IV	LV_ERR_RATIO_CHK
LV_MIS_STATE_A	LV_ERR_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_MAP	LV_CDN_VB_OBD1
LV_ERR_FSD_i	LV_LAM_LSCL[NC_CBK_EX_NR]	LV_LAM_LIM_MAX	LV_ACT_INT_PUC_i
LV_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]	CONF_LAM	CONF_CAT_EFF	

FUNCTION DESCRIPTION:

General information:

According the exhaust gas system shall the variable be defined from [i]=1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables are initialises to **1** at every LV_IGK = 0 -> 1

Recurrence: 100ms

Activation: *If LV_IGK = 1*

Deactivation: If the conditions above were not fullfiled.

Before deactivation all variables shall be reset to 1.

Formula section:

Inhibition Conditions for OC/SCP_LS_DOWN Diagnoses:

If LV_IGK = 1
and LV_CDN_VB_OBD1 = 1
and LV_ERR_LSH_DOWN[i] = 0
and LV_ERR_OBD_LSH_DOWN[i] = 0
and CONF_LAM = 1
and CONF_CAT_EFF > 0
Then LV_INH_DIAG_OC_LS_DOWN[i] = 0
LV_INH_DIAG_SCP_LS_DOWN[i] = 0
Else LV_INH_DIAG_OC_LS_DOWN[i] = 1
LV_INH_DIAG_SCP_LS_DOWN[i]=1

Inhibition Conditions for SCG_LS_DOWN Diagnosis:

If LV_IGK = 1
and LV_CDN_VB_OBD1 = 1
and LV_ERR_EL_CPS = 0
and LV_ERR_MEC_OPEN_CPS = 0
and LV_MIS_STATE_A = 0
and LV_MIS_STATE_B1 = 0
and LV_MIS_STATE_B4 = 0
and LV_ERR_MAF = 0
and LV_ERR_RATIO_CHK = 0
and LV_ERR_CRK = 0
and LV_ERR_FSD_i = 0
and LV_LAM_LSCL[i] = 1
and LV_LAM_LIM_MAX = 0
and LV_ACT_INT_PUC_i = 0
and CONF_LAM = 1
and CONF_CAT_EFF > 0
Then LV_INH_DIAG_SCG_LS_DOWN[i] = 0
Else LV_INH_DIAG_SCG_LS_DOWN[i] = 1

Inhibition Conditions for FL_LS_DOWN Diagnosis

:If LV_IGK = 1
and LV_CDN_VB_OBD1 = 1
and LV_ERR_LS_DOWN[i] = 0
and LV_ERR_LSH_DOWN[i] = 0
and LV_ERR_OBD_LSH_DOWN[i] = 0
and STATE_ERR_IV = 0
and LV_ERR_EL_CPS = 0
and LV_ERR_MEC_OPEN_CPS = 0
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
and LV_ERR_RATIO_CHK = 0
and LV_MIS_STATE_A = 0
and LV_MIS_STATE_B1 = 0
and LV_MIS_STATE_B4 = 0
and CONF_LAM = 1
and CONF_CAT_EFF > 0
then LV_INH_DIAG_FL_LS_DOWN[i] = 0
else LV_INH_DIAG_FL_LS_DOWN[i] = 1

Calcualtion of RBM inhibition for plausibility diagnosis during fuel cut-off

:If LV_ERR_LS_DOWN[i] = 0
and LV_ERR_LSH_DOWN[i] = 0
and LV_ERR_OBD_LSH_DOWN[i] = 0
and STATE_ERR_IV = 0
and LV_ERR_EL_CPS = 0
and LV_ERR_MEC_OPEN_CPS = 0
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
and LV_ERR_RATIO_CHK = 0
and LV_MIS_STATE_A = 0
and LV_MIS_STATE_B1 = 0
and LV_MIS_STATE_B4 = 0
then LV_INH_DIAG_RBM_PUC_LS_DOWN[i] = 0
else LV_INH_DIAG_RBM_PUC_LS_DOWN[i] = 1
endif

Inhibition condition for plausibility diagnosis during fuel cut-off

:If	LV_IGK = 1
and	LV_CDN_VB_OBD1 = 1
and	LV_INH_DIAG_RBM_PUC_LS_DOWN[i] = 0
and	CONF_LAM = 1
and	CONF_CAT_EFF > 0
then	LV_INH_DIAG_PUC_LS_DOWN[i] = 0
else	LV_INH_DIAG_PUC_LS_DOWN[i] = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_MAX_SCG_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter maximum for short to ground electrical check					
C_ABC_INC_SCG_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter increment for short to ground electrical check					
C_ABC_MAX_SCP_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter maximum for short to plus electrical check					
C_ABC_INC_SCP_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter increment for short to plus electrical check					
C_ABC_MAX_OC_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter maximum for open line electrical check					
C_ABC_INC_OC_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter increment for open line electrical check					
C_ABC_MAX_PUC_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter maximum for PUC plausibility					
C_ABC_INC_PUC_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter increment for PUC plausibility					
C_ABC_MAX_FL_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter maximum for FL plausibility					
C_ABC_INC_FL_LS_DOWN	1	0...FFH	0...255	1	-
Debounce counter increment for FL plausibility					

Configuration for diagnostic symptoms :

Diagnosis	Symptom	Nr	P-Code/ Failure	P-Code/ Symptom	Hardware config	Recurence	Failure class A/B
Downstream Oxygen Sensor Signal	*Short to ground (bank 1/2)*	*0*	0137/0157			100ms	
SCG_LS_DOWN[i]							
Diagnosis	**Symptom**	**Nr**	**P-Code/ Failure**	**P-Code/ Symptom**	**Hardware config**	**Recurence**	**Failure class A/B**
Downstream Oxygen Sensor Signal	*Short to plus (bank 1/2)*	*0*	0138/0158			100ms	
SCP_LS_DOWN[i]							
Diagnosis	**Symptom**	**Nr**	**P-Code/ Failure**	**P-Code/ Symptom**	**Hardware config**	**Recurence**	**Failure class A/B**

Downstream Oxygen Sensor Signal	*Open line (bank 1/2)*	*0*	0140/0160			100ms	
OC_LS_DOWN[i]							
Diagnosis	**Symptom**	**Nr**	**P-Code/ Failure**	**P-Code/ Symptom**	**Hardware config**	**Recurence**	**Failure class A/B**
Downstream Oxygen Sensor Signal Plausibility	*Signal plausibility fault (bank 1/2)*	*0*	0136/0156			100ms	
PUC_LS_DOWN[i]							
Diagnosis	**Symptom**	**Nr**	**P-Code/ Failure**	**P-Code/ Symptom**	**Hardware config**	**Recurence**	**Failure class A/B**
Downstream Oxygen Sensor Signal Plausibility	*Signal plausibility fault (bank 1/2)*	*0*	0136/0156			100ms	
FL_LS_DOWN[i]							

A.7.2 Interface for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_PUC_LS_DOWN[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of PUC_LS_DOWN monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_PUC_LS_DOWN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Downstream O2 sensor PUC plausibility diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	NC_CBK_EX_NR	LV_INH_DIAG_RBM_PUC_LS_DOWN[i]
LV_ERR_LOAD_TPS_PLAUS	CTR_CDN_RBM_PUC_LS_DOWN_1	CTR_COMP_RBM_PUC_LS_DOWN_1	LV_END_DIAG_PUC_LS_DOWN[NC_CBK_EX_NR]
LV_ERR_MAF_SCG_OC	LV_ERR_VCC_PVS_2	LV_ERR_EL_CPS	LV_ERR_MEC_OPEN_CPS
LV_ERR_OBD_LSH_DOWN[i]	LV_ERR_MAF_SCP	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS
LV_ERR_SCG_LS_DOWN[i]	LV_ERR_SWT_LS_DOWN[i]	LV_ERR_PUE_LS_DOWN[i]	LV_ERR_LSH_DOWN[i]
LV_ERR_FSD[i]	LV_ERR_SCP_LS_DOWN[i]	LV_ERR_OC_LS_DOWN[i]	LV_ERR_FSD_LAM_LIM[i]

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the PUC_LS_DOWN[i] and the Rate-Based Monitoring statistics is defined with STATE_RBM_PUC_LS_DOWN[i] data.

Within STATE_RBM_PUC_LS_DOWN[i], three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_PUC_LS_DOWN[i] = 0 at LV_DC 0 → 1 transition :

bit 0 and bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 0

on failure memory reset :

bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_PUC_LS_DOWN[i] = 1

The pending status of the following failures has to be checked only once:

For ETC and non_ETC variants:

Dependence	Error			
	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_VCC_PVS_2	LV_ERR_EL_CPS	LV_ERR_MEC_OPEN_CPS
	LV_ERR_MAF_SCG_OC	LV_ERR_MAF_SCP	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS
NC_CBK_EX_NR	LV_ERR_OBD_LSH_DOWN[i]	LV_ERR_SWT_LS_DOWN[i]	LV_ERR_PUE_LS_DOWN[i]	LV_ERR_LSH_DOWN[i]
	LV_ERR_SCG_LS_DOWN[i]	LV_ERR_SCP_LS_DOWN[i]	LV_ERR_OC_LS_DOWN[i]	
	LV_ERR_FSD[i]	LV_ERR_FSD_LAM_LIM[i]		

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_PUC_LS_DOWN[i] = 0
and LV_END_DIAG_PUC_LS_DOWN[i] = 1
Then bit 0 of STATE_RBM_PUC_LS_DOWN[i] = 1
Endif

If bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 0
and LV_INH_DIAG_RBM_PUC_LS_DOWN[i] = 1
Then bit 1 of STATE_RBM_PUC_LS_DOWN[i] = 1
Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_PUC_LS_DOWN_1 = 0
then RATE_RBM_PUC_LS_DOWN_1 = FFFFH
else RATE_RBM_PUC_LS_DOWN_1
= (CTR_COMP_RBM_PUC_LS_DOWN_1 / CTR_CDN_RBM_PUC_LS_DOWN_1)
endif

A.7.3 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_DOWN_MIN_TOT_DC	V/O/S	0...3FFH	0...4.995	0.0049	V
Former / current driving cycle minimum downstream oxygen sensor voltage					
VLS_DOWN_MAX_TOT_DC	V/O/S	0...3FFH	0...4.995	0.0049	V
Former / current driving cycle maximum downstream oxygen sensor voltage					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_IGK		

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

VLS_DOWN_MIN_TOT_DC = 3FFH

VLS_DOWN_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: 10ms

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If VLS_DOWN[1] < VLS_DOWN_MIN_TOT_DC

Then VLS_DOWN_MIN_TOT_DC = VLS_DOWN[1]

Endif

If VLS_DOWN[1] > VLS_DOWN_MAX_TOT_DC

Then VLS_DOWN_MAX_TOT_DC = VLS_DOWN[1]

Endif

A.8 Signal plausibility monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAG_CDN_PLAUS_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Boolean flag, plausibility check on downstream oxygen sensor signal possible					
LV_ERR_PLAUS_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean flag, error plausibility check on downstream oxygen sensor signal					

Input data:

LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]	LV_LS_DOWN_READY[NC_CBK_EX_NR]	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]	LV_ERR_PUC_LS_DOWN[NC_CBK_EX_NR]
NC_CBK_EX_NR	STATE_LSH_DOWN[NC_CBK_EX_NR]	LV_IGK	LV_ERR_FL_LS_DOWN[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

Due to possible oxygen sensor defects (e.g. reference air poisoning) or faults in the injection system (e.g. leaking fuel injector), the oxygen sensor may not provide the expected lean or rich AFR signal level during overrun fuel cut-off (PUC) or full load (FL) condition. Hence, the oxygen sensor signal shall be checked for plausibility during this engine operating states.

Description:

If the following conditions are met, then the general conditions for plausibility check are determined to be present:

A. Ignition key shall be determined to be on (LV_IGK = 1)
B. No electrical oxygen sensor fault shall be determined to be present on the sensor being diagnosed (LV_ERR_EL_LS_DOWN[i] = 0).
C. Sensor shall be in a state of operative readiness (LV_LS_DOWN_READY[i] =1).
D. The sensor shall be determined to be at operating temperature (STATE_LSH_DOWN[i] = LSH_POW_CTL & LV_ERR_LSH_DOWN[i] = 0).

The corresponding flag LV_DIAG_CDN_PLAUS_LS_DOWN[i] shall then be set.

For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Application conditions:

Initialisation:

At the transition LV_IGK = 0 -> 1, reset or clearing the error memory, reset of all variables and bits to 0

Recurrence: **T_sample 10 ms**

Activation / Deactivation:

If	LV_IGK = 1
and	LV_ERR_EL_LS_DOWN[i] = 0
and	LV_LS_DOWN_READY[i] = 1
and	LV_ERR_LSH_DOWN[i] = 0
and	STATE_LSH_DOWN[i] = LSH_POW_CTL
then	**“Diagnosis available”**
	LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1
else	**“Diagnosis NOT available”**
	LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 0
endif	

Formula section:

If	LV_ERR_PUC_LS_DOWN[i] = 1
or	LV_ERR_FL_LS_DOWN[i] = 1
then	LV_ERR_PLAUS_LS_DOWN[i] = 1
else	LV_ERR_PLAUS_LS_DOWN[i] = 0
endif	

A.8.1 Overrun fuel cut-off (PUC) oxygen sensor signal plausibility

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Boolean error flag, PUC plausibility fault currently present on downstream oxygen sensor signal					
ERR_SYM_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Boolean error flag, fault currently present on downstream oxygen sensor signal					
LV_CDN_DIAG_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Status conditions of oxygen sensor downstream diagnosis					
LV_END_DIAG_PUC_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Bit to indicate end of complete downstream oxygen sensor signal diagnosis					
MAF_INT_PUC_NOT_LS_DOWN_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...2912.66666	0.0444444	[g]
MAF integral out of PUC in case of certain load threshold exceeded					
MAF_INT_PUC_ACT_OLD	-	0...FFFFH	0...2912.66666	0.0444444	[g]
MAF integral during PUC from previous recurrence					
LV_DIAG_PUC_CHG_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Reminder flag for fuel cut off was active					
VLS_DOWN_PUC_PLAUS_VLD[NC_CBK_EX_NR]	V	0...3FFH	0...4.99511	4.8828e-3	[V]
VLS_DOWN[i] value before entering fuel-cut phase					
LV_VLS_DOWN_PUC_AFR_VLD[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Logical variable indicating that rich voltage threshold has been exceeded before entering PUC phase					
VLS_DOWN_PUC_OK[NC_CBK_EX_NR]	V	0...3FFH	0...4.99511	4.8828e-3	[V]
Last VLS_DOWN value satisfying PUC plausibility					
VLS_DOWN_PUC_ERR[NC_CBK_EX_NR]	V	0...3FFH	0...4.99511	4.8828e-3	[V]
Last VLS_DOWN value not plausible for PUC conditions					
VLS_DOWN_PUC_SAVE[NC_CBK_EX_NR]	V/O/S	0...3FFH	0...4.99511	4.8828e-3	[V]
Mode 6 value describing last VLS_DOWN PUC plausibility result					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_PUC	MAF_INT_PUC_ACT	LV_DIAG_CDN_PLAUS_LS_DOWN[NC_CBK_EX_NR]
LV_INH_DIAG_PUC_LS_DOWN[NC_CBK_EX_NR]	LV_IGK	NC_CBK_EX_NR	MAF_KGH
	LV_DC		

FUNCTION DESCRIPTION:

General information:

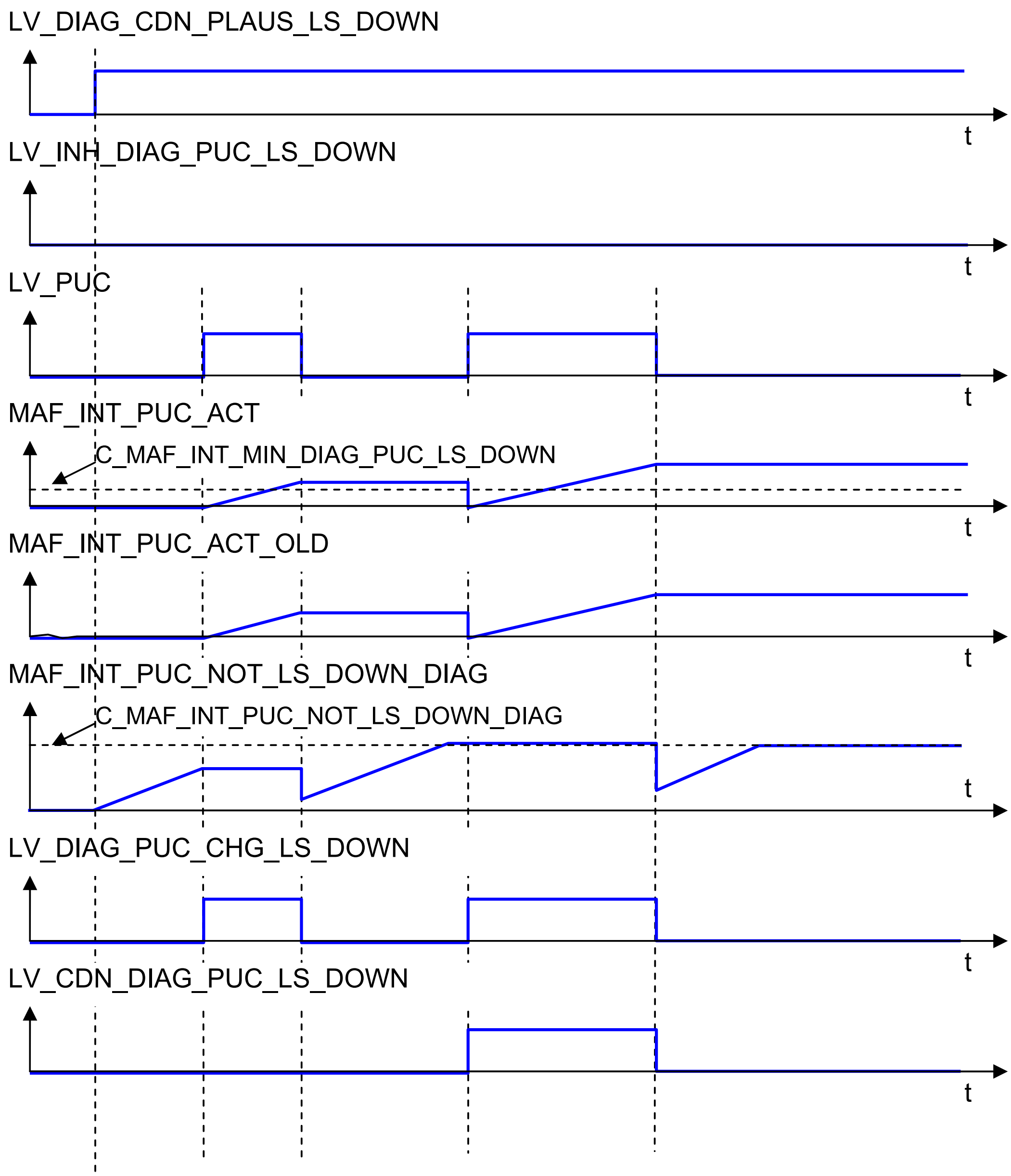
Description:

Should the oxygen sensor signal voltage (VLS_DOWN[i]) exceed the calibrateable thresholds (C_VLS_MIN_DIAG_PUC_LS_DOWN) during engine operating state PUC and the following additional conditions are met, then a fault shall be determined to be present:

A. The appropriate inhibit flag shall not be set (LV_INH_DIAG_PUC_LS_DOWN[i]).
B. Plausibility check set to be possible (LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1)
C. The PUC state (LV_PUC = 1) shall be determined to be stable, as determined by comparing the mass air flow integral calculated after activation (MAF_INT_PUC_ACT) with the threshold range (C_MAF_INT_MIN_VLS_DOWN_DIAG; C_MAF_INT_MAX_VLS_DOWN_DIAG).
D. The energy transfer from the exhaust gas to the oxygen sensor shall exceed the calibratable threshold (C_MAF_INT_PUC_NOT_LS_DOWN_DIAG).
E. The reset of the MAF integral from the last fuel cut off phase must have taken place before activating the diagnosis. Otherwise there is a risk to set the conditions to 1 right at the beginning of the fuel cut off phase. At this time, the lambda sensor voltage downstream is still high and the error symptom would be set for one recurrence.
F. Prior to the PUC phase the VLS_DOWN[i] must have exceeded the voltage thresholds C_VLS_DOWN_PUC_PLAUS_VLD and C_VLS_DOWN_PUC_AFR_VLD.

The error flag LV_ERR_PUC_LS_DOWN[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.



Application conditions:

Initialisation:

At the transition LV_IGK = 0 -> 1, reset or clearing the error memory, reset of all variables and bits (except LV_END_DIAG_xx) to 0, except VLS_DOWN_PUC_SAVE. VLS_DOWN_PUC_SAVE[i] shall be initialised to 0 at new ECU and in case of NVMY checksum errors. In normal case, VLS_DOWN_PUC_SAVE shall be initialised to its saved value. Reset only at clearing error memory.

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence: **T_sample 100ms**

Activation / Deactivation: -

Formula section:

Conditions to start the downstream lambda sensor signal observation

If (1) LV_PUC = 0
And(1) VLS_DOWN[i] > C_VLS_DOWN_PUC_AFR_VLD
Then(1) LV_VLS_DOWN_PUC_AFR_VLD[i] = 1
Elseif(1) LV_PUC = 1 --> 0
Then(1) LV_VLS_DOWN_PUC_AFR_VLD[i] = 0
Endif(1)
if LV_PUC = 0 --> 1
then VLS_DOWN_PUC_PLAUS_VLD[i] = VLS_DOWN[i]
endif

If(1) LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1
And(1) LV_INH_DIAG_PUC_LS_DOWN[i] = 0
And(1) LV_PUC = 1
Then(1) LV_DIAG_PUC_CHG_LS_DOWN[i] = 1
If(2) C_MAF_INT_MAX_DIAG_PUC_LS_DOWN > MAF_INT_PUC_ACT > C_MAF_INT_MIN_DIAG_PUC_LS_DOWN
And(2) MAF_INT_PUC_NOT_LS_DOWN_DIAG[i] >= C_MAF_INT_PUC_NOT_LS_DOWN_DIAG
% diagnosis active after a certain energy transfer out of fuel cut off
And(2) MAF_INT_PUC_ACT > MAF_INT_PUC_ACT_OLD
% MAF integral during PUC from last DC must be reset before starting the diagnosis
And(2) VLS_DOWN_PUC_PLAUS_VLD[i] > C_VLS_DOWN_PUC_PLAUS_VLD
And(2) LV_VLS_DOWN_PUC_AFR_VLD[i] = 1
Then(2) **"Diagnosis available"**
LV_CDN_DIAG_PUC_LS_DOWN[i] = 1
Else(2) **"Diagnosis NOT available"**
LV_CDN_DIAG_PUC_LS_DOWN[i] = 0
Endif(2)
Else(1) LV_CDN_DIAG_PUC_LS_DOWN[i] = 0
If(2a) LV_PUC = 0
And(2a) LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1

And(2a) LV_INH_DIAG_PUC_LS_DOWN[i] = 0

And(2a) MAF_KGH > C_MAF_KGH_MIN_LS_DOWN_DIAG

Then(2a)

If(3a) LV_DIAG_PUC_CHG_LS_DOWN[i] = 1 (fuel cut off was active)

Then(3a) $MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N} = MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N-1} - C\_FAC\_MAF\_INT\_LS\_DOWN\_DIAG * MAF\_INT\_PUC\_ACT$

LV_DIAG_PUC_CHG_LS_DOWN[i] = 0

Endif(3a)

If(3b) MAF_INT_PUC_NOT_LS_DOWN_DIAG[i] <= C_MAF_INT_PUC_NOT_LS_DOWN_DIAG

Then(3b) $MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N} = MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N-1} + MAF\_KGH * T\_SAMPLE[ms] * 1/3600[(g*h) / (kg*ms)]$

Else(3b) $MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N} = MAF\_INT\_PUC\_NOT\_LS\_DOWN\_DIAG[i]_{N-1}$

Endif(3b)

Endif(2a)

Endif(1)

MAF_INT_PUC_ACT_OLD = MAF_INT_PUC_ACT

Conditions to start debouncing

If LV_CDN_DIAG_PUC_LS_DOWN[i] = 1

then **If** VLS_DOWN[i] > C_VLS_MIN_DIAG_PUC_LS_DOWN

then ***"PUC signal plausibility fault present"***

ERR_SYM_PUC_LS_DOWN[i] = sym_0 /"*PUC PLAUSIBILITY ERROR*"

/Error Manager shall debounce this symptom

VLS_DOWN_PUC_ERR[i] = VLS_DOWN[i]

else

ERR_SYM_PUC_LS_DOWN[i] = "*NO_SYM*"

/Error Manager shall decrement the debounce counter

VLS_DOWN_PUC_OK[i] = VLS_DOWN[i]

endif

endif

% At this point, update of the ERRM shall be completed.

For failure and error management treatment the anti-bounce mechanism is called :

ACTION_ERRM_FilterSymptom(IN<NC_IDX_DIAG_ PUC_LS_DOWN[i]>,
IN<LV_CDN_DIAG_ PUC_LS_DOWN[i]>,
IN<ERR_SYM_ PUC_LS_DOWN[i]>, IN<C_ABC_INC_ PUC_LS_DOWN>, IN<1>,
IN<C_ABC_MAX_ PUC_LS_DOWN >, OUT<LV_ERR_ PUC_LS_DOWN[i]>)

% In the following part, ***current*** *values of LV_END_DIAG_PUC_LS_DOWN[i] and LV_ERR_PUC_LS_DOWN[i]*
% (after updating ERRM with the diagnostic result of this recurrence) shall be used:

if LV_END_DIAG_PUC_LS_DOWN[i] 0 to 1

% Mode 06 test value update shall be done only once

then

if LV_ERR_PUC_LS_DOWN[i] = 1

then

VLS_DOWN_PUC_SAVE[i] = VLS_DOWN_PUC_ERR[i]

else

VLS_DOWN_PUC_SAVE[i] = VLS_DOWN_PUC_OK[i]

endif

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_INT_MAX_DIAG_PUC_LS_DOWN	1	0...FFFFH	0...2912.66666	4.44E-02	[g]
Maximum threshold for MAF integral to end downstream PUC signal plausibility					
C_MAF_INT_MIN_DIAG_PUC_LS_DOWN	1	0...FFFFH	0...2912.66666	4.44E-02	[g]
Minimum threshold for MAF integral required to start downstream PUC signal plausibility					
C_VLS_MIN_DIAG_PUC_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
VLS voltage threshold for PUC fault detection					
C_MAF_INT_PUC_NOT_LS_DOWN_DIAG	1	0...FFFFH	0...2912.66666	4.44E-02	[g]
Minimum threshold for MAF integral out of PUC required to start downstream PUC signal plausibility					
C_FAC_MAF_INT_LS_DOWN_DIAG	1	0...FFH	0...31.875	0.125	[-]
Factor for impact of fuel cut off phase to the sensor temperature					
C_MAF_KGH_MIN_LS_DOWN_DIAG	1	0...FFFFH	0...2047.96875	3.125E-2	kg/h
Minimum MAF_KGH to start MAF integral					
C_VLS_DOWN_PUC_PLAUS_VLD	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Rich voltage threshold for activation of PUC plausibility diagnosis					
C_VLS_DOWN_PUC_AFR_VLD	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Rich voltage monitoring threshold between fuel-cut phases					

Configuration for diagnostic symptoms:

Diagnostic XX	Symptom description	Symptom	Filter type
PUC_LS_DOWN[NC_CBK_EX_NR]	Signal high during fuel cut off	SYM_0	MEM

A.8.2 Full load (FL) oxygen sensor signal plausibility

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FL_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean error flag, FL plausibility fault currently present on downstream oxygen sensor signal					
ERR_SYM_FL_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	0	[-]
Boolean error flag, fault currently present on downstream oxygen sensor signal					
LV_CDN_DIAG_FL_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Status conditions of oxygen sensor downstream diagnosis					
LV_END_DIAG_FL_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Bit to indicate end of complete downstream oxygen sensor signal diagnosis					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_FL	MAF_INT_FL	LV_DIAG_CDN_PLAUS_LS_DOWN[NC_CBK_EX_NR]
LV_INH_DIAG_FL_LS_DOWN[NC_CBK_EX_NR]	LV_IGK	NC_CBK_EX_NR	LV_DC

FUNCTION DESCRIPTION:

General information:

Description:

Should the oxygen sensor signal voltage (VLS_DOWN[i]) exceed the calibrateable thresholds (C_VLS_MAX_DIAG_FL_LS_DOWN) during engine operating state FL and the following additional conditions are met, then a fault shall be determined to be present:

A. The appropriate inhibit flag shall not be set (LV_INH_DIAG_FL_LS_DOWN[i]).
B. Plausibility check set to be possible (LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1)
C. The FL state (LV_FL = 1) shall be determined to be stable, as determined by comparing the mass air flow integral calculated after activation (MAF_INT_FL) with the threshold range (C_MAF_INT_MIN_DIAG_FL_LS_DOWN; C_MAF_INT_MAX_DIAG_FL_LS_DOWN).

The error flag LV_ERR_FL_LS_DOWN[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Application conditions:

Initialisation:

At the transition LV_IGK = 0 -> 1, reset or clearing the error memory, reset of all variables and bits (except LV_END_DIAG_xx) to 0

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence: **T_sample 100ms**

Activation / Deactivation:

If LV_DIAG_CDN_PLAUS_LS_DOWN[i] = 1
and LV_INH_DIAG_FL_LS_DOWN[i] = 0
and LV_FL = 1
and C_MAF_INT_MAX_DIAG_FL_LS_DOWN > MAF_INT_FL
> C_MAF_INT_MIN_DIAG_FL_LS_DOWN
then **"Diagnosis available"**
LV_CDN_DIAG_FL_LS_DOWN[i] = 1
else **"Diagnosis NOT available"**
LV_CDN_DIAG_FL_LS_DOWN[i] = 0
endif

Formula section:

If LV_CDN_DIAG_FL_LS_DOWN[i] = 1
then **If** VLS_DOWN[i] < C_VLS_MAX_DIAG_FL_LS_DOWN
then ***"FL signal plausibility fault present"***
ERR_SYM_FL_LS_DOWN[i] = sym_1 / "*FL PLAUSIBILITY ERROR*"
/Error Manager shall debounce this symptom
else
ERR_SYM_FL_LS_DOWN[i] = "*NO_SYM*"
/Error Manager shall decrement the debounce counter
endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_INT_MAX_DIAG_FL_LS_DOWN	1	0...FFFFH	0...1820.42	2.78E-02	[g]
Maximum threshold for MAF integral to end downstream FL signal plausibility					
C_MAF_INT_MIN_DIAG_FL_LS_DOWN	1	0...FFFFH	0...1820.42	2.78E-02	[g]
Minimum threshold for MAF integral required to start downstream FL signal plausibility					
C_VLS_MAX_DIAG_FL_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
VLS voltage threshold for FL fault detection					

Configuration for diagnostic symptoms:

Diagnostic XX	Symptom description	Symptom	Filter type
FL_LS_DOWN[NC_CBK_EX_NR]	Signal low during full load	SYM_1	MEM

A.8.3 Calculation of MAF_INT_FL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_INT_FL	V/O	0...FFFFH	0...1820.42	2.78E-02	[g]
integral of air mass flow while FL active					

Input data:

LV_FL	MAF_CYL	LV_ST_END	LV_IGK

FUNCTION DESCRIPTION:

General information:

Description:

The calculation of MAF_INT_FL expires in 20ms raster as long as the operation status FL is fulfilled even when the application conditions for plausibility check function are not fulfilled. MAF_INT_FL_i is limited to the maximum value if necessary in case of a longer FL phase.
The value of MAF_INT_FL should reset to 0 if the transition LV_FL = 1 -> 0 takes place.

Application conditions:

Initialisation:

At the transition LV_IGK = 0 -> 1 or reset reset of all variables and bits to 0

Recurrence: **Tsample = 20ms**

Activation / Deactivation:

If LV_ST_END = 1

then *calculation enabled*

else *calculation disabled*

endif

Formula section:

If LV_FL = 1

then

$MAF\_INT\_FL_n$ [g] = $MAF\_INT\_FL_n$ [g] + MAF_CYL * T_SAMPLE [ms] * 1/3600 [(g*h) / (kg*ms)]

else

MAF_INT_FL = 0

endif

A.9 Downstream oxygen sensor heater OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_LSH_DOWN[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Error flag for downstream heater diagnosis OBD I after debouncing					
LV_CDN_DIAG_LSH_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Diagnosis condition for downstream heater diagnosis OBD I					
LV_END_DIAG_LSH_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
End of diagnosis for downstream heater diagnosis OBD I					
ERR_DIAG_LSH_DOWN[NC_CBK_EX_NR]	V	0...7H	0...7	1	-
Raw value of error symptom for downstream heater diagnosis OBD I					
ERR_SYM_LSH_DOWN[NC_CBK_EX_NR]	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom for downstream heater diagnosis OBD I filterd with CDN_DIAG_LSH_DOWN					
CDN_DIAG_LSH_DOWN[NC_CBK_EX_NR]	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of downstream heater diagnosis OBD I					

Input data:

LV_ST_END	STATE_LSH_DOWN[NC_CBK_EX_NR]	**LV_VB_CDN_OBD_1**	TEG_CAT_DOWN_MDL[NC_CBK_EX_NR]
LSHPWM_DOWN[NC_CBK_EX_NR]	LV_LSH_OC_LSH_DOWN[NC_CBK_EX_NR]	LV_LSH_SCP_LSH_DOWN[NC_CBK_EX_NR]	LV_LSH_SCG_LSH_DOWN[NC_CBK_EX_NR]
LV_INH_DIAG_LSH_DOWN[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_IGK	LV_DC

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

All symptoms of the current error code are handeled by anti-bouncing.
We assume that only one symptom of an error code can be active at the same time.

Description:

The error detection is effected via the ECU hardware.

The purpose of the diagnosis shall be to detect electrical faults within the oxygen sensor heater circuit as defined by OBD I requirements.

The function is intended for use in a hardware configuration where the heater driver shall carry out the electrical fault diagnosis and provide the electrical status via status flags.

- *Short circuit to Vbatt by Over-temperature* ***or*** *Overcurrent* (**SCP**)
- *Short circuit to GND* (SCG)
- *Open load* (**OC**)

The electrical status flags for each heater driver shall be read out via SPI and decoded by the appropriate function typically every 10 ms and made available to the Application SW in the form of driver and fault specific flags. The flags shall only be set by the appropriate function and may only be reset by the diagnosis function. These flags are:

- **SCP** LV_LSH_SCP_LSH_DOWN[i]
- SCG LV_LSH_SCG_LSH_DOWN[i]
- **OC** LV_LSH_OC_LSH_DOWN[i]

The heater drivers named are capable of detecting an **OC** and **SCP** fault when in the ON state and only capable of detecting a SCG fault in the OFF state. As the heater power is controlled by a PWM signal, the driver will be placed alternately in an ON and then OFF state.

The diagnosis shall be activated should the following conditions be met:

- Engine leaving the engine start state as denoted by LV_ST_END = 1
- Battery voltage falls between pre-set thresholds to protect against false errors at battery voltage extremes as indicated by **LV_VB_CDN_OBD_1**.
- LSHPWM signal greater or equal to or lower or equal to specific calibrateable value. Should the PWM be set to 100 %, no SCG fault can be recognised (e.g. for ATIC39 & ATM3x min. of 250 µs required). Similarly for the **OC** / **SCP** fault, a fault can only be detected when the ON time is longer than a minimum threshold (e.g. for ATM3x for **OC** min. 250 µs). The exact HW thresholds are dependent upon the characteristics of the external circuitry.
- Inhibit flag not set (LV_INH_DIAG_LSH_DOWN[i])

In exceptionally rare extreme environmental conditions for certain system configurations, it is possible that the heater driver may be driven into over-temperature during initial oxygen sensor heating. In this case, the driver automatically turns off the output stage in self-protection and sets the **SCP** electrical fault bit. In order to mask an incorrect oxygen sensor heater **SCP** fault, the function shall only allow the **SCP** to set a fault during the initial heater phase, if the exhaust gas temperature has exceeded a pre-determined calibrateable threshold.

The oxygen sensor heater fault detection shall be debounced by use of an anti-bounce counter. This shall be handled by the Dynamic Error Management.

Each of the above listed faults shall be described individually below:

Otherwise the diagnosis has to be suppressed (CDN_DIAG_LSH_DOWN[i] = 0). This is managed within the “Diagnosis condition calculation.

Application conditions:

Activation : every engine state

Deactivation: -

Initialization: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at reset

NOTE: The input flags LV_LSH_SCP_LSH_DOWN[i], LV_LSH_SCG_LSH_DOWN[i] & LV_LSH_OC_LSH_DOWN[i] shall be initialised to 0 at the beginning of a new driving cycle by the appropriate BSW function.

Recurrence: 1 s

NOTE: The heater PWM frequency shall be at least 2 times faster than the diagnosis frequency to ensure that an oversampling effect does not occur!

Formula section:

CDN_DIAG_LSH_DOWN[i] = 0

If LV_ST_END = 1
and **LV_VB_CDN_OBD_1** = 1
and LV_INH_DIAG_LSH_DOWN[i] = 0
then **If** LSHPWM_DOWN[i] ≥ C_LSHPWM_MIN_DIAG_LSH_DOWN
and [(STATE_LSH_DOWN[i] <> LSH_POW_RISE
and STATE_LSH_DOWN[i] <> LSH_POW_RED)
or TEG_CAT_DOWN_MDL[i] ≥ C_TEG_CAT_MDL_DOWN_MIN_LSH_DOWN]
then bit 0 of CDN_DIAG_LSH_DOWN[i] = 1 *(**SCP** can be detected))*
endif
If LSHPWM_DOWN[i] ≤ C_LSHPWM_MAX_DIAG_LSH_DOWN
then bit 1 of CDN_DIAG_LSH_DOWN[i] = 1 *(**SCG** can be detected)*
endif
If C_LSHPWM_MIN_DIAG_LSH_DOWN ≤ LSHPWM_DOWN[i] ≤ C_LSHPWM_MAX_DIAG_LSH_DOWN
then bit 2 of CDN_DIAG_LSH_DOWN[i] = 1 *(**OC** can be detected)*
endif
endif

Error Symptom calculation (raw value from I/O SW)

ERR_DIAG_LSH_DOWN[i] = 0

If LV_LSH_SCP_LSH_DOWN[i] = 1
then bit 0 of ERR_DIAG_LSH_DOWN[i] = 1
endif
If LV_LSH_SCG_LSH_DOWN[i] = 1
then bit 1 of ERR_DIAG_LSH_DOWN[i] = 1
endif
If LV_LSH_OC_LSH_DOWN[i] = 1
then bit 2 of ERR_DIAG_LSH_DOWN[i] = 1
endif

For failure and error management treatment the anti-bounce mechanism is called with the parameters CDN_DIAG_LSH_DOWN[i] and ERR_DIAG_LSH_DOWN[i].

This algorithm determines:

ERR_SYM_LSH_DOWN[i] (= raw value ERR_DIAG_LSH_DOWN[i] filtered with CDN_DIAG_LSH_DOWN[i], detected error symptom for lambda sensor heater upstream [i] diagnosis)

LV_ERR_ LSH_DOWN[i] (Error flag for debounced error of for lambda sensor heater upstream [i])

LV_CDN_DIAG_ LSH_DOWN[i] (Diagnosis condition information)

LV_END_DIAG_ LSH_DOWN[i] (End of diagnosis information)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LSHPWM_MIN_DIAG_LSH_DOWN	1	0...FFH	0...99.609	0.3891	%
Minimum O2-Sensor heater duty cycle to perform the electrical diagnosis					
C_LSHPWM_MAX_DIAG_LSH_DOWN	1	0...FFH	0...99.609	0.3891	%
Maximum O2-Sensor heater duty cycle to perform the electrical diagnosis					
C_TEG_CAT_MDL_DOWN_MIN_LSH_DOWN	1	0...7FF0H	0...2047	0.0625	°C
Minimum exhaust gas temperature threshold to permit **SCP** to set fault during LSH power rise state					

Configuration data:

Diagnosis	Symptom	Nr	ABC type
Downstream oxygen sensor heater OBDI diagnosis	*Short circuit to battery*	*0*	MPL_STD_INI
	Short circuit to GND	*1*	
	Open circuit	*2*	
SH_DOWN[i]			

A.9.1 Application incidences for downstream oxygen sensor heater OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Inhibit flag for downstream oxygen sensor heater OBDI diagnosis					

Input data:

LV_ST_END	NC_CBK_EX_NR	NLC_LSH_RLY_EFP	LV_EFP
LV_ERR_RLY_EFP	CONF_LAM	CONF_CAT_EFF	

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks. In case of two separate catalyst systems (NC_CBK_EX_NR = 2) i = 1 for exhaust bank 1, i = 2 for exhaust bank 2.

Description:

In some systems the supply voltage of the O2-Sensor heater may be connected to the electrical fuel pump relay (NLC_LSH_RLY_EFP = 1). In this case, if the electrical fuel pump is switched off or if the fuel pump relay is defective, no heating would occur and the heater driver will recognize an incorrect heater fault. Hence the OBD I heater diagnosis shall be inhibited if LV_EFP = 0 or if LV_ERR_RLY_EFP = 1.

Application conditions:

Initialisation: LV_INH_DIAG_LSH_DOWN[i] shall be reset at transition LV_ST_END 0 → 1

Recurrence: The function shall be carried out once every 1 s.

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Formula section:

If NLC_LSH_RLY_EFP = 1
then *(O2-Sensor Heater supplied via EFP relay)*
  If LV_EFP = 0 *(EFP relay not active)*
  or LV_ERR_RLY_EFP = 1 *(error at EFP relay)*
  or CONF_LAM = 0 *(open loop lambda control)*
  or CONF_CAT_EFF = 0
  then LV_INH_DIAG_LSH_DOWN[i] = 1
  else LV_INH_DIAG_LSH_DOWN[i] = 0
else *(O2-Sensor Heater supplied via EFP relay)*
  if CONF_LAM = 0 *(open loop lambda control)*
  or CONF_CAT_EFF = 0

then LV_INH_DIAG_LSH_DOWN[i] = 1

else LV_INH_DIAG_LSH_DOWN[i] = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_DIAG_LSH_DOWN	1	0...FFH	0...255	1	-
Anti-bounce counter increment for downstream O2-sensor heater signal diagnosis.					
C_ABC_MAX_DIAG_LSH_DOWN	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for downstream O2-sensor heater command signal diagnosis.					

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Downstream oxygen sensor heater OBDI diagnosis Bank 1, Sensor 2	*Short circuit to battery*	SYM_0	*STD*
	Short circuit to GND	SYM_1	
	Open circuit	SYM_2	
LSH_DOWN_1		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
LSHPWM_DOWN_1
R_IT_LS_DOWN_1_ENVD_H
R_IT_LS_DOWN_1_ENVD_L

Application assistances:

Recommendation: The Application Assistances shall be maintained for the remaining duration of the driving cycle.

- Lambda control (Only for exhaust gas temperatures lower than O2 sensor lowest operating temperature):
  Downstream trim regulation - Inhibit (Bank selectively)
- Oxygen sensor / Lambda controller diagnosis:
  O2 sensor downstream signal diagnosis – Inhibit (Bank selectively, OC & SCB)
  O2 sensor downstream response time diagnosis – Inhibit (Bank selectively)
  O2 sensor downstream PUC end diagnosis – Inhibit (Bank selectively)
- Catalyst efficiency diagnosis - Inhibit (Bank selectively) (indirectly via LV_LSCL)

A.10 ETC position sensor diagnosis

FUNCTION DESCRIPTION:

General information:

The following TPS diagnosis can be distinguished:

- Diagnosis of the admissible voltage ranges V_TPS_1 and V_TPS_2
  To avoid engine stall during chattering TPS_1or 2 signal the diagnosis threshold at idle can be calibrated lower with two constants C_V_TPS_X_MIN/MAX_IS. So the time to reach the error threshold can be reduced. (HMC bad signal contact test procedure)
- In case of high deviations between TPS_AV_1 and TPS_AV_2, a plausibility check with the main load signal (MAF_TPS_MES) is activated and the faulty TPS channel is searched.
- In the case of one defective TPS channel, the plausibility check of the error-free sensor with the main load signal (MAF_TPS_MES) is activated (second TPS error). The monitoring of the error-free channel runs continuously.

A.10.1 Diagnosis of the admissible voltage ranges V_TPS_1 and V_TPS_2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TPS_1	O/V	0...1h	0...1	1	-
Logic variable (set if TPS_1 is identified as errored)					
LV_CDN_DIAG_TPS_1	O/V	0...1h	0...1	1	-
Diagnosis condition TPS 1 diagnosis					
ERR_SYM_TPS_1	O/V	0h 1h 2h	NO_SYM SYM_0 SYM_1	1	-
Error symptom TPS 1 diagnosis					
LV_END_DIAG_TPS_1	O/V	0...1h	0...1	1	-
End of diagnosis TPS 1 diagnosis					
LV_ERR_TPS_2	O/V	0...1h	0...1	1	-
Logic variable (set if TPS_2 is identified as errored)					
LV_CDN_DIAG_TPS_2	O/V	0...1h	0...1	1	-
Diagnosis condition TPS 2 diagnosis					
ERR_SYM_TPS_2	O/V	0h 1h 2h	NO_SYM SYM_0 SYM_1	1	-
Error symptom TPS 2 diagnosis					
LV_END_DIAG_TPS_2	O/V	0...1h	0...1	1	-
End of diagnosis TPS 2 diagnosis					
TPS_AV_ MAX_DIAG _TPS_IS	V	0...FFh	0...119.5	0.0073	°TPS
TPS_SP threshold for diagnosis of the admissible voltage ranges at IS					
TMP_TPS_AV_1_DIAG_TPS_IS	V	0...FFh	0...119.5	0.0073	°TPS
TPS_AV_1 threshold for diagnosis of the admissible voltage ranges at IS					
TMP_TPS_AV_2_DIAG_TPS_IS	V	0...FFh	0...119.5	0.0073	°TPS
TPS_AV_2 threshold for diagnosis of the admissible voltage ranges at IS					

Input data:

V_TPS_1	V_TPS_2	LV_IGK	LV_INH_DIAG_TPS_1
LV_INH_DIAG_TPS_2	V_TPS_AD_EL_BOL_1	V_TPS_AD_EL_BOL_2	TPS_SP
LV_LIH_MAF	LV_LIH_MAF_TMP	TPS_SLOP	LV_IS

FUNCTION DESCRIPTION:

General information:

In the case of an error-free system, the potentiometer voltages must be within admissible limits.

Description:

The following errors symptoms can be distinguished:

ERR_SYM_TPS_1:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │ │ │ └─> Short circuit to Vbatt or open load    (= SYM_0)
 │ │ └───> Short circuit to ground                (= SYM_1)
 │ └─────> -                                      (= SYM_2)
 └───────> -                                      (= SYM_3)
```

ERR_SYM_TPS_2:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │ │ │ └─> Short circuit to Vbatt                 (= SYM_0)
 │ │ └───> Short circuit to ground or open load   (= SYM_1)
 │ └─────> -                                      (= SYM_2)
 └───────> -                                      (= SYM_3)
```

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset :

set all error variables ..._TPS_x to zero (done by the error-management due to filter-typ MEM)

Recurrence: 10 ms

Activation:

if **(** LV_IGK **and** *{calculation for channel 1 and 2 separate}*
LV_INH_DIAG_TPS_X == 0 **and** V_TPS_x were calculated.**)**
then LV_CDN_DIAG_TPS_X = 1
else LV_CDN_DIAG_TPS_X = 0
endif

Deactivation: When the activation condition is not fulfilled.

Formula section:

TMP_TPS_AV_1_DIAG_TPS_IS =

(C_V_TPS_MAX_DIAG_1_IS – V_TPS_AD_EL_BOL_1) * TPS_SLOP

TMP_TPS_AV_2_DIAG_TPS_IS =

(V_TPS_AD_EL_BOL_2 - C_V_TPS_MIN_DIAG_2_IS) * TPS_SLOP

TPS_AV_MAX_DIAG_TPS_IS = MAX (MIN (TMP_TPS_AV_X_DIAG_TPS_IS), 0)

Error detection TPS 1 short circuit to ground:

```
if      V_TPS_1  <  C_V_TPS_MIN_DIAG_1
then    ERR_SYM_TPS_1 = SYM_1
        LV_ERR_TPS_1 = 1                        {after debounce, set for this DC}
else    Error detection TPS_1 short circuit to Vbatt or open load:
        if      V_TPS_1  >  C_V_TPS_MAX_DIAG_1
        or          [( LV_LIH_MAF = 1 or LV_LIH_MAF_TMP = 1
                      or ERR_SYM_TPS_2 <> "NO_SYM" )
                    and TPS_SP < TPS_AV_ MAX_DIAG _TPS_IS
                    and LV_IS = 1
                    and V_TPS_1 > C_V_TPS_MAX_DIAG_1_IS]
        then    ERR_SYM_TPS_1 = SYM_0
                LV_ERR_TPS_1 = 1                {after debounce, set for this DC}
        else    ERR_SYM_TPS_1 = NO_SYM
        endif
endif
```

Error detection TPS_2 short circuit to ground or open load:

```
if      V_TPS_2  <  C_V_TPS_MIN_DIAG_2
or              [( LV_LIH_MAF = 1 or LV_LIH_MAF_TMP = 1
                  or ERR_SYM_TPS_1 <> "NO_SYM" )
                and TPS_SP < TPS_AV_ MAX_DIAG _TPS_IS
                and LV_IS = 1
                and V_TPS_2 < C_V_TPS_MIN_DIAG_2_IS)]
then    ERR_SYM_TPS_2 = SYM_1                   LV_ERR_TPS_2 = 1  {after debounce,
set for this DC}
else    Error detection TPS_2 short circuit to VBatt:
        if      V_TPS_2  >  C_V_TPS_MAX_DIAG_2
        then    ERR_SYM_TPS_2 = SYM_0
                LV_ERR_TPS_2 = 1                {after debounce, set for this DC}
```

else ERR_SYM_TPS_2 = NO_SYM

endif

endif

Calculation end of diagnosis:

LV_END_DIAG_TPS_1/2 are calculated by error management.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_TPS_MIN_DIAG_1	1	0...3FFH	0...4.9951	0.0049	V
lower diagnostic voltage, pot 1					
C_V_TPS_MIN_DIAG_2	1	0...3FFH	0... 4.9951	0.0049	V
lower diagnostic voltage, pot 2					
C_V_TPS_MAX_DIAG_1	1	0...3FFH	0... 4.9951	0.0049	V
upper diagnostic voltage, pot 1					
C_V_TPS_MAX_DIAG_2	1	0...3FFH	0... 4.9951	0.0049	V
upper diagnostic voltage, pot 2					
C_V_TPS_MAX_DIAG_1_IS	1	0...3FFH	0... 4.9951	0.0049	V
upper diagnostic voltage, pot 1 at IS					
C_V_TPS_MIN_DIAG_2_IS	1	0...3FFH	0... 4.9951	0.0049	V
lower diagnostic voltage, pot 2 at IS					

A.10.2 Throttle position sensor plausibility check

A.10.2.1 Comparison of TPS values TPS_AV_1 and TPS_AV_2

A.10.2.2 Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_TPS_DIAG_STEP	V	0h 1h 2h 4h 8h	INIT RATIO_CHK TPS_1_CHK TPS_2_CHK DBL_ERROR	1	-
State variable indicates the state of the TPS diagnosis / plausibility check					
LV_ERR_TPS_RATIO	O/V	0...1h	0...1	1	-
Logic variable is set, if a TPS ratio error is present					
LV_ERR_TPS_MAF_1	O/V	0...1h	0...1	1	-
Logic variable is set, if a plausibility error between TPS 1 and MAF is detected					
LV_CDN_DIAG_TPS_MAF_1	O/V	0...1h	0...1	1	-
Diagnosis condition TPS_MAF 1					
ERR_SYM_TPS_MAF_1	O/V	0h 4h	NO_SYM SYM_2	1	-
Error symptom TPS_MAF 1					
LV_END_DIAG_TPS_MAF_1	O/V	0...1h	0...1	1	-
End of diagnosis TPS_MAF 1					
LV_ERR_TPS_MAF_2	O/V	0...1h	0...1	1	-
Logic variable is set, if a plausibility error between TPS 2 and MAF is detected					
LV_CDN_DIAG_TPS_MAF_2	O/V	0...1h	0...1	1	-
Diagnosis condition TPS_MAF 2					
ERR_SYM_TPS_MAF_2	O/V	0h 8h	NO_SYM SYM_3	1	-
Error symptom TPS_MAF 2					
LV_END_DIAG_TPS_MAF_2	O/V	0...1h	0...1	1	-
End of diagnosis TPS_MAF 2					
FAC_MAF_TPS	V	0...FFFFh	0...1.999969	3.051e-5	-
Correction factor depend on environment conditions					
MAF_TPS _SUB_1	V	0...FFFFh	0...2047.96875	0.03125	kg/h
Substitute mass air flow calculated with TPS_AV_1					
MAF_TPS _SUB_2	V	0...FFFFh	0...2047.96875	0.03125	kg/h
Substitute mass air flow calculated with TPS_AV_2					
TPS_MAF_ERR_1	V	0...FFFFh	0... 255.9961	0.0039	-
Normalized error between MAF_TPS_MES and MAF_TPS_SUB_1 (TPS ratio error)					
TPS_MAF_ERR_2	V	0...FFFFh	0... 255.9961	0.0039	-
Normalized error between MAF_TPS_MES and MAF_TPS_SUB_2 (TPS ratio error)					
TPS_MAF_ERR_INT_1	V	0...FFFFh	0... 255.9961	0.0039	-
Integrated error between MAF_TPS_MES and MAF_TPS_SUB_1 (TPS ratio error)					
TPS_MAF_ERR_INT_2	V	0...FFFFh	0... 255.9961	0.0039	-
Integrated error between MAF_TPS_MES and MAF_TPS_SUB_2 (TPS ratio error)					
TPS_MAF_ERR_MMV_1	V	0...FFFFh	0... 255.9961	0.0039	-
Moving mean value of the error between MAF_TPS_MES and MAF_TPS_SUB_1 (second TPS error)					
TPS_MAF_ERR_MMV_2	V	0...FFFFh	0... 255.9961	0.0039	-
Moving mean value of the error between MAF_TPS_MES and MAF_TPS_SUB_2 (second TPS error)					

Input data:

LV_IGK	TPS_AV	TPS_AV_1	TPS_AV_2
MAF_TPS_MES	MAF_TPS_EXT	LV_INH_DIAG_TPS_MAF	LV_ERR_MAF
LV_ERR_MAP	N_32	PUT	TIA_THR
TCO	LV_DC		

FUNCTION DESCRIPTION:

General information:

In the case of an error-free system, the throttle position TPS_AV_1 and TPS_AV_2 must not differ by more than C_TPS_RATIO_HYS. If the threshold C_TPS_RATIO_HYS is exceeded, the TPS error detection is done through plausibility with MAF_TPS_MES.

Description:

Detection of the implausible channel in the case of a comparison error within the admissible voltage ranges. To perform this plausibility check, the variables MAF_TPS_SUB_x are calculated on the basis of the throttle position values TPS_AV_x .

The absolute amount of the differences between MAF_TPS_SUB_x and the mass air flow measured are determined, normalized with IP_MAF_TPS_DIF_DIAG and integrated over the period C_T_TPS_MAF_ERR_INT.

After the integration period the TPS channel which has the highest error sum is declared as faulty. The other channel is also declared as faulty, if it’s error sum exceeds the threshold C_TPS_MAF_ERR_INT. If both error sums are identical (threshold not exceeded), TPS channel 1 is declared as faulty.

Above the limitation IP_TPS_MAX_DIAG the value MAF_TPS_SUB_X has no longer a practical relationship to TPS_AV. In this case, no error detection is performed (only first error detection, TPS ratio check).

The variable LV_ERR_TPS_RATIO isn’t subject to the error management and is set, if a TPS ratio error occurs.

Error symptoms are defined for this diagnosis function as:

ERR_SYM_TPS_MAF_1:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> -                                (= SYM_0)
 │ │ └───> -                                (= SYM_1)
 │ └─────> TPS 1 and MAF are not plausible  (= SYM_2)
 └───────>-                                 (= SYM_3)
```

ERR_SYM_TPS_MAF_2:

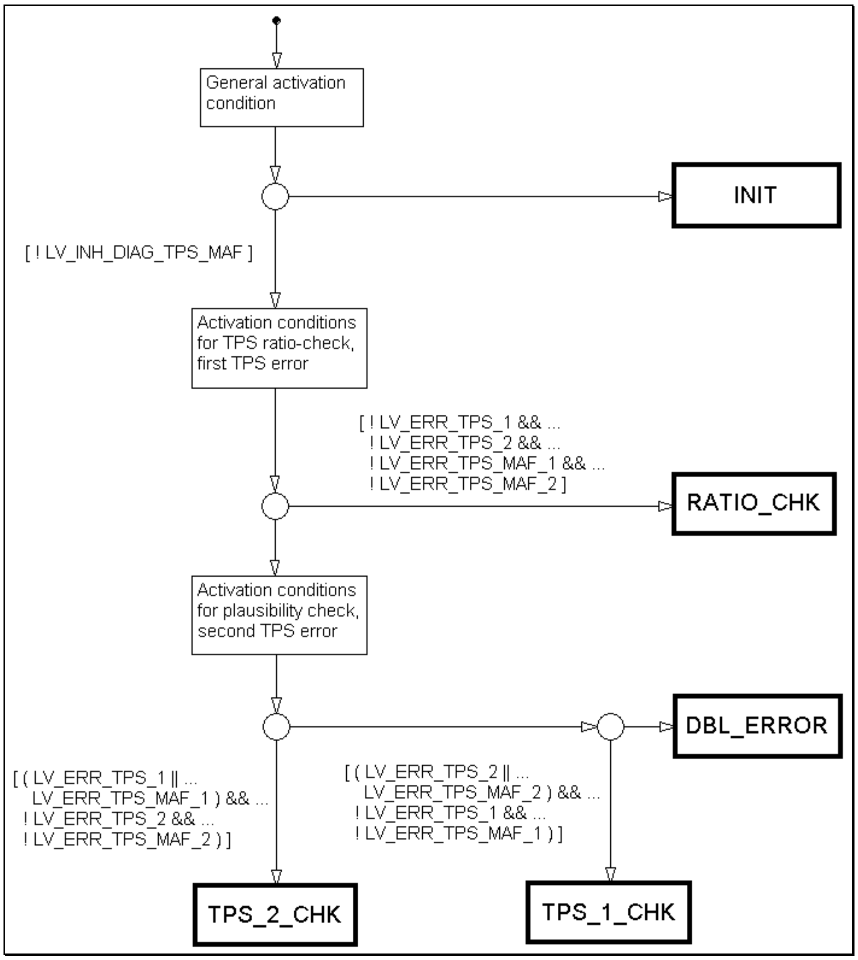
```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> -                                (= SYM_0)
 │ │ └───> -                                (= SYM_1)
 │ └─────> -                                (= SYM_2)
 └───────> TPS 2 and MAF are not plausible  (= SYM_3)
```

State variable for TPS diagnosis

The state variable STAT_TPS_DIAG_STEP indicates the state of the TPS plausibility check.

STATE_TPS_DIAG_STEP	Comment
INIT	general activation conditions not fulfilled
RATIO_CHK	TPS ratio – check, both TPS channels error-free
TPS_1_CHK	TPS 2 error, plausibility check TPS 1 and MAF_TPS_MES
TPS_2_CHK	TPS 1 error, plausibility check TPS 2 and MAF_TPS_MES
DBL_ERROR	TPS 1 and 2 error (double TPS error)

Signal flow diagram:



TPS ratio and plausibility check

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset :

Set all error variables ..._TPS_MAF_x and LV_ERR_TPS_RATIO to zero (except LV_END_DIAG_TPS_MAF_x), STATE_TPS_DIAG_STEP = INIT

LV_END_DIAG_TPS_MAF_x are initialised with 0 at every LV_DC = 0 → 1 **or** reset

Recurrence: 10 ms
Activation: LV_IGK == 1
Deactivation: When the activation condition is not fulfilled.

Formula section:

Detection of the diagnosis state STATE_TPS_DIAG_STEP

```
if      LV_INH_DIAG_TPS_MAF == 0
then    if      LV_ERR_TPS_X == 0             and     {both channels must be errorfree}
                LV_ERR_TPS_MAF_X == 0
        {First TPS error, TPS ratio-check}
        then    STATE_TPS_DIAG_STEP = "RATIO_CHK"
                Symptom calculation TPS Ratio-check:
                if      | TPS_AV_1 - TPS_AV_2 |  >  C_TPS_RATIO_HYS
                then    LV_ERR_TPS_RATIO = 1
                        Diagnosis condition for plausibility-check with MAF_TPS_MES
                        if      TPS_AV < IP_TPS_MAX_DIAG
                        then    LV_CDN_DIAG_TPS_MAF_X = 1
                        else    LV_CDN_DIAG_TPS_MAF_X = 0
                        endif
                else    LV_ERR_TPS_RATIO = 0
                        LV_CDN_DIAG_TPS_MAF_X = 0
                endif
        {Second TPS error, diagnosis condition for plausibility-check with MAF_TPS_MES}
                TPS 1 defective, plausibility check TPS 2
        else    if      [ LV_ERR_TPS_1 or LV_ERR_TPS_MAF_1 ]  and
                        LV_ERR_TPS_2 == 0                     and
                        LV_ERR_TPS_MAF_2 == 0
                then    STATE_TPS_DIAG_STEP = "TPS_2_CHK"
                        LV_CDN_DIAG_TPS_MAF_1 = 0
                        LV_CDN_DIAG_TPS_MAF_2 = 1
                        TPS 2 defective, plausibility check TPS 1
                else    if      [ LV_ERR_TPS_2 or LV_ERR_TPS_MAF_2 ]  and
                                LV_ERR_TPS_1 == 0                     and
                                LV_ERR_TPS_MAF_1 == 0
                        then    STATE_TPS_DIAG_STEP = "TPS_1_CHK"
                                LV_CDN_DIAG_TPS_MAF_1 = 1
                                LV_CDN_DIAG_TPS_MAF_2 = 0
                        else    STATE_TPS_DIAG_STEP = "DBL_ERROR"
                                LV_CDN_DIAG_TPS_MAF_X = 0
                        endif
                endif
        endif
```

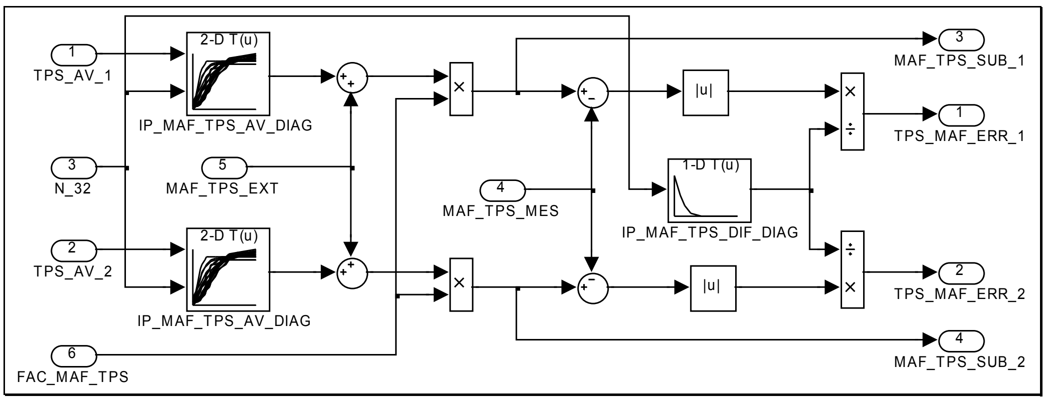
else STATE_TPS_DIAG_STEP = "INIT"
LV_ERR_TPS_RATIO = 0
LV_CDN_DIAG_TPS_MAF_X = 0
endif

A.10.2.3 Calculation of TPS – MAF deviation

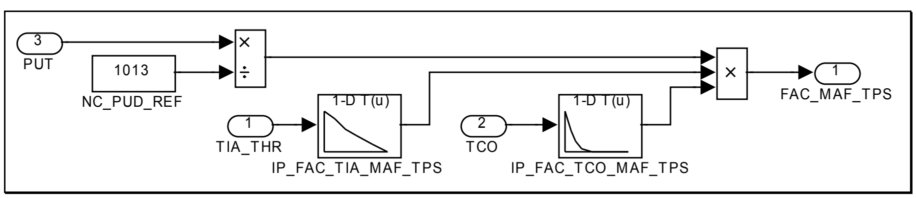
General information:

This function calculates the substitute air mass flow MAF_TPS_SUB_X for both TPS channels based on TPS_AV_X and engine speed. The module supports air mass flows as a result of charging and a correction depend on environment conditions (FAC_MAF_TPS). The variable MAF_TPS_EXT is defined in the corresponding diagnostic applications incidence.
The result of the calculation are the errors between MAF sensor and substitute air mass flow MAF_TPS_SUB_X.

Signal flow diagram:



Calculation of MAF_TPS_SUB and TPS – MAF deviation



Consideration of the environment conditions

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset :
MAF_TPS_SUB_x = TPS_MAF_ERR_x = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

Calculation of MAF_SUB_X: *{calculation for channel 1 and 2 separate}*

if LV_CDN_DIAG_TPS_MAF_X
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
then Consideration of the environment conditions

FAC_MAF_TPS = IP_FAC_TIA_MAF_TPS * IP_FAC_TCO_MAF_TPS ...
* (PUT / NC_PUD_REF)

Calculation of the substitute air mass flow depend on TPS

{TPS - Breakpoints for IP_MAF_TPS_AV_DIAG: TPS_AV_x}
MAF_TPS_SUB_x = [IP_MAF_TPS_AV_DIAG(N_32, TPS_AV_x) ...
+ MAF_TPS_EXT] * FAC_MAF_TPS

Calculation of the normalized TPS – MAF error:

$$TPS\_MAF\_ERR\_x = \frac{|MAF\_TPS\_SUB\_x - MAF\_TPS\_MES|}{IP\_MAF\_TPS\_DIF\_DIAG}$$

else

MAF_TPS_SUB_x = 0
TPS_MAF_ERR_x = 0

endif

A.10.2.4 Diagnosis of a first TPS error, TPS ratio error

FUNCTION DESCRIPTION:

General information:

Detection of the implausible channel in the case of a poti comparison error !

Description:

The substitute values MAF_TPS_SUB_X are used for the plausibility check while the engine is running. The differences between the measured mass air flow and the substitute values are normalized and integrated over a period C_T_TPS_MAF_ERR_INT.
The plausibility check cannot be performed when a mass air flow signal is missing, in this case, the channels are declared as faulty.

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset :
TPS_MAF_ERR_INT = 0;
integration timer t_tps_maf_err_int and t_tps_maf_err_int_2 = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

```
if      STATE_TPS_DIAG_STEP == "RATIO_CHK"    and
        LV_CDN_DIAG_TPS_MAF_1                 and
        LV_CDN_DIAG_TPS_MAF_2

then    Start integration timer C_T_TPS_MAF_ERR_INT
        t_tps_maf_err_int + = 10 ms

        Calculation of TPS_MAF_ERR_INT_X:     {calculation for channel 1 and 2 separate}
        TPS_MAF_ERR_INT_X(k)  = TPS_MAF_ERR_INT_X(k-1) + TPS_MAF_ERR_X

Error detection, integration timer is expired
        if      t_tps_maf_err_int >= C_T_TPS_MAF_ERR_INT
        then    if      TPS_MAF_ERR_INT_1  >= TPS_MAF_ERR_INT_2
                then    ERR_SYM_TPS_MAF_1 = SYM_2
                        LV_ERR_TPS_MAF_1 = 1                   {set for this DC}
                        if      TPS_MAF_ERR_INT_2 > C_TPS_MAF_ERR_INT
                        or      LV_ERR_MAF = 1
                        or      LV_ERR_MAP = 1
                        then    ERR_SYM_TPS_MAF_2 = SYM_3
                                LV_ERR_TPS_MAF_2 = 1           {set for this DC}
                        endif
                else    ERR_SYM_TPS_MAF_2 = SYM_3
                        LV_ERR_TPS_MAF_2 = 1                   {set for this DC}
                        if      TPS_MAF_ERR_INT_1 > C_TPS_MAF_ERR_INT
                        or      LV_ERR_MAF = 1
                        or      LV_ERR_MAP = 1
                        then    ERR_SYM_TPS_MAF_1 = SYM_2
                                LV_ERR_TPS_MAF_1 = 1           {set for this DC}
                        endif
                endif
        else    { Integration is still active }
        endif

else    { Integration timer and error are reset to zero }
        t_tps_maf_err_int       = 0
        TPS_MAF_ERR_INT_X = 0

endif
```


Calculation end of diagnosis:

if STATE_TPS_DIAG_STEP == "RATIO_CHK" **and**
TPS_AV < IP_TPS_MAX_DIAG

then Start end of diagnosis timer C_T_TPS_MAF_ERR_INT
t_tps_maf_err_int_2 + = 10 ms

End of diagnosis timer is expired
if t_tps_maf_err_int_2 **>=** C_T_TPS_MAF_ERR_INT
then *{Set end of diagnosis information}*
LV_END_DIAG_TPS_MAF_1 = 1
LV_END_DIAG_TPS_MAF_2 = 1
endif

else *{End of diagnosis timer is reset to zero}*
t_tps_maf_err_int_2 = 0

endif

A.10.2.5 Diagnosis of a second TPS error

FUNCTION DESCRIPTION:

General information:

In the case of one channel is diagnosed as faulty, a plausibility check for the remaining channel to the other one is no longer possible.

Description:

The substitute value MAF_TPS_SUB_X is used for the plausibility check while the engine is running. The difference between the measured mass air flow and the substitute value is normalized and the mean value is established with a correlation constant.
The plausibility check cannot be performed when a mass air flow signal is missing, in this case, the channel is declared as faulty.

Application conditions:

Initialisation: at LV_IGK = 0 $\rightarrow$ 1 **or** reset :
TPS_MAF_ERR_MMV_x = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

if STATE_TPS_DIAG_STEP == "TPS_1_CHK" *{calculation for channel 1}*

then Calculation of TPS_MAF_ERR_MMV_1:

```
        TPS_MAF_ERR_MMV_1(k)  = TPS_MAF_ERR_MMV_1(k-1)
                              + [ TPS_MAF_ERR_1 – TPS_MAF_ERR_MMV_1(k-1) ]
                              * C_TPS_MAF_ERR_CRLC
```

<u>Error detection:</u>

```
        if      TPS_MAF_ERR_MMV_1 > C_TPS_MAF_ERR_MMV
        or      LV_ERR_MAF = 1
        or      LV_ERR_MAP = 1
        then    ERR_SYM_TPS_MAF_1 = SYM_2
                LV_ERR_TPS_MAF_1 = 1
        endif
else    TPS_MAF_ERR_MMV_1 = 0
endif

if      STATE_TPS_DIAG_STEP == "TPS_2_CHK"    {calculation for channel 2}
then    Calculation of TPS_MAF_ERR_MMV_2:

        TPS_MAF_ERR_MMV_2(k)  = TPS_MAF_ERR_MMV_2(k-1)
                              + [ TPS_MAF_ERR_2 – TPS_MAF_ERR_MMV_2(k-1) ]
                              * C_TPS_MAF_ERR_CRLC

        Error detection:

        if      TPS_MAF_ERR_MMV_2 > C_TPS_MAF_ERR_MMV
        or      LV_ERR_MAF = 1
        or      LV_ERR_MAP = 1
        then    ERR_SYM_TPS_MAF_2 = SYM_3
                LV_ERR_TPS_MAF_2 = 1
        endif
else    TPS_MAF_ERR_MMV_2 = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TPS_RATIO_HYS	1	0H...3FFFh	0...119.5	0.0073	°TPS
Admissible deviation of throttle valve angle between TPS 1 and 2					
IP_TPS_MAX_DIAG	1x8	0...3FFFh	0...119,5	0.0073	°TPS
LDP_N_32_TPS_MAX_DIAG	8	0...FFh	0...8160	32	rpm
Maximum throttle angle for TPS ratio diagnosis					
IP_FAC_TIA_MAF_TPS	1x8	0...FFh	0...1.992	0.0078	-
LDP_TIA_THR_FAC_TIA_MAF_TPS	8	0...Feh	-48...142.5	0.75	°C
Correction of air density depend on TIA_THR					
IP_FAC_TCO_MAF_TPS	1x8	0...FFh	0...1.992	0.0078	-
LDP_TCO_FAC_TIA_MAF_TPS	8	0...Feh	-48...142.5	0.75	°C
Coolant temperature correction					
IP_MAF_TPS_DIF_DIAG	1x8	0...FFFFh	0...2047.96875	0.03125	kg/h
LDP_N32_MAF_TPS_DIF_DIAG	8	0...FFh	0...8160	32	rpm
Maximum MAF difference between MAF_TPS_MES and MAF_TPS_SUB					
IP_MAF_TPS_AV_DIAG	16x16	0...FFFFh	0...2047.96875	0.03125	kg/h
LDP_N_32_MAF_TPS_AV_DIAG	16	0...FFh	0...8160	32	rpm
LDP_TPS_AV_MAF_TPS_AV_DIAG	16	0...3FFFh	0...119.5	0.0073	°TPS
Basic air mass flow through the throttle body depend on TPS and engine speed					
C_T_TPS_MAF_ERR_INT	1	0...FFh	0...2.55	0.01	s
Time for TPS – MAF error integration, TPS ratio error					
C_TPS_MAF_ERR_INT	1	0...NFFFh	0...255.9961	0.0039	-
Limit for the integrated TPS – MAF error, TPS ratio error					
C_TPS_MAF_ERR_CRLC	1	0...FFFFh	0...1	0.000015	-
Correlation constant for filtering of TPS – MAF error, second TPS error					
C_TPS_MAF_ERR_MMV	1	0...FFFFh	0... 255.9961	0.0039	-
Limit for calculation of the floating mean value, second TPS error					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PUT_REF	1	0H...FFFFH	0...5434	0.083	hPa
Reference pressure upstream throttle, value is set to 1013 hPa !					

A.10.3 ETC position sensor diagnosis (appl. inc.)

A.10.3.1 Diagnosis of the admissible voltage ranges V_TPS_1 and V_TPS_2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TPS_1	V/O	0h...1h	0...1	1	-
Inhibition condition for the TPS 1 diagnosis					
LV_INH_DIAG_TPS_2	V/O	0h...1h	0...1	1	-
Inhibition condition for the TPS 2 diagnosis					

Input data:

LV_IGK	LV_ERR_TPS_VCC	LV_ERR_TPS_1	LV_ERR_TPS_2
LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2		

FUNCTION DESCRIPTION:

General information:

The flags LV_INH_DIAG_TPS_1/2 permit to deactivate the corresponding diagnostics.

Application conditions:

Initialisation: at LV_IGK = 0 → 1 or reset:
LV_INH_DIAG_TPS_x = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

{ calculate inhibition condition for the electrical TPS 1 diagnosis }

```
LV_INH_DIAG_TPS_1 = ( LV_ERR_TPS_VCC     or
                      LV_ERR_TPS_1       or
                      LV_ERR_TPS_MAF_1   )
```

{ calculate inhibition condition for the electrical TPS 2 diagnosis }

```
LV_INH_DIAG_TPS_2 = ( LV_ERR_TPS_VCC     or
                      LV_ERR_TPS_2       or
                      LV_ERR_TPS_MAF_2   )
```

Configuration for diagnostic symptoms:

Diagnostic	Symptom description	Symptom	Filter type
TPS 1 electrical diagnosis	Short circuit to Vbatt or open load	SYM_0	*MEM*
	Short circuit to ground	SYM_1	
		SYM_2	
TPS_1		SYM_3	

List of Environmental data to store in failure memory: VCC_TPS_DIAG_ENVD
V_TPS_1_ENVD
V_TPS_2_ENVD
TPS_AV_1_ENVD

Diagnostic	Symptom description	Symptom	Filter type
TPS 2 electrical diagnosis	Short circuit to Vbatt	SYM_0	*MEM*
	Short circuit to ground or open load	SYM_1	
		SYM_2	
TPS_2		SYM_3	

List of Environmental data to store in failure memory: VCC_TPS_DIAG_ENVD
V_TPS_1_ENVD
V_TPS_2_ENVD
TPS_AV_2_ENVD

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TPS_1	1	0...FFH	0...255	1	-
Anti-bounce increment value TPS 1 diagnosis					
C_ABC_MAX_TPS_1	1	1...FFH	1...255	1	-
Anti-bounce maximum value TPS 1 diagnosis					
C_ABC_INC_TPS_2	1	0...FFH	0...255	1	-
Anti-bounce increment value TPS 2 diagnosis					
C_ABC_MAX_TPS_2	1	1...FFH	1...255	1	-
Anti-bounce maximum value TPS 2 diagnosis					

A.10.3.2 Throttle position sensor plausibility check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TPS_MAF	V/O	0...1h	0...1	1	-
Inhibition condition for the TPS – MAF pausibility-check					
MAF_TPS_MES	V/O	0...FFFFh	0...2047.96875	0.03125	kg/h
Mass air flow measured by the main load sensor					
MAF_TPS_EXT	V/O	0...FFFFh	0...2047.96875	0.03125	kg/h
Substitute mass air flow (charging or recirculation flap)					

Input data:

LV_IGK	LV_ST	LV_ES	C_CONF_MAF
MAF_KGH_MES	MAP_MES	EFF_VOL_SLOP	EFF_VOL_OFS_SUM

FUNCTION DESCRIPTION:

General information:

This module provides a external substitute air mass flow based on engine charging or recirulation flap (and so on) for the corresponding TPS plausibility check with main load sensor (MAF_TPS_MES).

The flags LV_INH_DIAG_TPS_MAF permit to deactivate the corresponding diagnostic.

Application conditions:

Initialisation: at LV_IGK = 0 → 1 or reset,
MAF_TPS_EXT = 0 *{ no charging or recirulation flap }*
MAF_TPS_MES = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

{ Inhibition conditions for the TPS – MAF plausibility – check }

```
LV_INH_DIAG_TPS_MAF = ( LV_ES          or
                        LV_ST          or
                        LV_TPS_AD_REQ  or
                        LV_ERR_TPS_VCC )
```

{ Selection of the main load sensor signal }

```
if      C_CONF_MAF = 1
then    MAF_TPS_MES = MAF_KGH_MES
else    MAF_TPS_MES = EFF_VOL_SLOP * MAP_MES – EFF_VOL_OFS_SUM
endif
```

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
TPS 1 plausibility check		SYM_0	*NO*
		SYM_1	
		SYM_2	
TPS_MAF_1	*TPS 1 and MAF are not plausible*	SYM_3	

List of Environmental data to store in failure memory: VCC_TPS_DIAG_ENVD
MAP_ENVD
TPS_AV_1_ENVD
TPS_AV_2_ENVD

Diagnostic	Symptom description	Symptom	Filter type
TPS 2 plausibility check		SYM_0	*NO*
		SYM_1	
		SYM_2	
TPS_MAF_2	*TPS 2 and MAF are not plausible*	SYM_3	

List of Environmental data to store in failure memory: VCC_TPS_DIAG_ENVD
MAP_ENVD
TPS_AV_1_ENVD
TPS_AV_2_ENVD

A.11 ETC power stage diagnosis

Output data:

Name	Mod.	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MTC_DR	O/V	0h...1h	0...1	1	-
MTC_DR error is present					
LV_CDN_DIAG_MTC_DR	O/V	0h...1h	0...1	1	-
Diagnosis condition MTC_DR					
ERR_SYM_MTC_DR	O/V	0h 8h	NO_SYM SYM_3	1	-
Detected symptom MTC_DR					
LV_END_DIAG_MTC_DR	O/V	0...1h	0...1h	1	-
End of diagnosis MTC_DR					

Input data:

LV_IGK	LV_INH_DIAG_MTC_DR	LV_DC	

FUNCTION DESCRIPTION:

General information:

The error detection is effected via the ECU hardware (MTC H-bridge IC).

Description:

The following errors symptoms leads to an error. They can not be distinguished by the software:

- Undervoltage
- Overcurrent
- Overtemperature
- H-bridge hardware disable

Error-symptoms are defined to this diagnosis function as following :

```
[ | | | ]
 |  |  |  └─> -                        (= SYM_0)
 |  |  └───> -                         (= SYM_1)
 |  └──────> -                         (= SYM_2)
 └─────────> ETC power stage error     (= SYM_3)
```

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at reset

Recurrence: 10 ms

Activation:

if LV_IGK **and**
LV_INH_DIAG_MTC_DR == 0
then LV_CDN_DIAG_MTC_DR = 1
else LV_CDN_DIAG_MTC_DR = 0
endif

Deactivation: When the activation condition is not fulfilled.

Formula section:

Call BSW routine to read out the error state lv_mtc_dr_sf of the power stage !

if lv_mtc_dr_sf *{ MTC H-bridge error detected }*
then ERR_SYM_MTC_DR = SYM_3
LV_ERR_MTC_DR = 1 *{ after debounce, set for this DC }*
else ERR_SYM_MTC_DR = NO_SYM
endif

Calculation end of diagnosis:

LV_END_DIAG_MTC_DR is calculated by error management.

A.12 ETC power stage diagnosis (appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_MTC_DR	V/O	0h...1h	0...1	1	-
Inhibition condition for the power stage diagnosis					

Input data:

LV_IGK	LV_ES	LV_ST	VB
LV_MTC_CUR_OFF			

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_MTC_DR permit to deactivate the corresponding diagnostic.

Description:

Application conditions:

Initialisation: at LV_IGK = 0 → 1 or reset,
LV_INH_DIAG_MTC_DR = 1

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled

Formula section:

LV_INH_DIAG_MTC_DR = **(** LV_ES **or**
LV_ST **or**
VB < C_VB_MIN_DIAG_MTC_DR **or**
LV_MTC_CUR_OFF **)**

Configuration for diagnostic symptoms:

Diagnostic ETC	Symptoms	Nb	Filter type
ETC power stage diagnosis		*0*	*MEM*
		1	
		2	
	ETC power stage error	3	

Diagnostic ETC	Symptoms	Nb	P-Code/ Symptom	P-Code/ Global 1	P-Code/ Global 2	Failure class A/B	Readiness/ Code
ETC power stage diagnosis		*0*		P21XX		CARB_CC	
		1					
		2					
	ETC power stage error	3	P2101				

{ P2101 – Throttle actuator control motor circuit range/performance }

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_MTC_DR	1	0..FFH	0...255	1	-
Debounce counter increment, MTC driver diagnosis					
C_ABC_MAX_MTC_DR	1	1..FFH	1...255	1	-
Debounce counter maximum value, MTC driver diagnosis					
C_VB_MIN_DIAG_MTC_DR	1	0 ... FFH	0 ... 26	0.102	V
Minimum voltage threshold for MTC_DR diagnosis					

A.13 ETC actuator diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MTC_CTL	V/O	0h...1h	0...1	1	-
Logical variable is set, if a system deviation error is present					
LV_CDN_DIAG_MTC_CTL	V/O	0h...1h	0...1	1	-
Diagnosis condition of system deviation check					
LV_END_DIAG_MTC_CTL	V/O	0h...1h	0...1	1	-
End of diagnosis, system deviation check					
ERR_SYM_MTC_CTL	V/O	0h 1h 2h 4h 8h	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom of system deviation check					
T_TPS_DIF_DIAG	V	0h...FFh	0...2.55	0.01	s
Debounce time counter for system deviation check					
T_TPS_DIF_SP	V	0h...FFh	0...2.55	0.01	s
Time counter for constant TPS setpoint					
T_END_DIAG_MTC_CTL	V	0h...FFh	0...2.55	0.01	s
Time counter for end of diagnosis					
MTCPWM_MMV	V	0h ... 7FFFh	0...99.997	0.00305	%
Moving mean value of the position controller output					
TPS_SP_ST_DIAG	-	0h...3FFFh	0...119.5	0.0073	°TPS
TPS setpoint at start of constant setpoint detection					

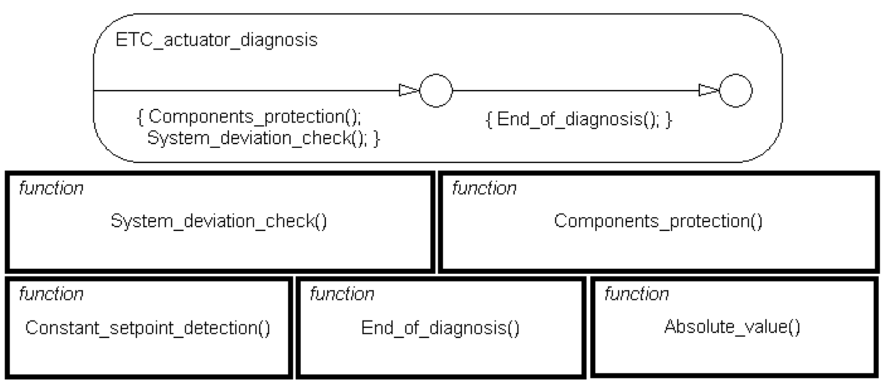
Input data:

LV_IGK	LV_INH_DIAG_MTC_CTL	TPS_SP	TPS_DIF
VB_MMV	MTCPWM	LV_DC	

General information:

The task of this diagnosis is to detect a throttle valve error or a jammed ETC actuator. The diagnosis observes the ETC position controller system deviation TPS_DIF in small and large signal range. The adjustment speed of the ETC actuator depends strongly on the operating voltage, therefore the debounce diagnosis time counter T_TPS_DIF_DIAG is a function of it.

Additionally the PWM output of the digital position controller is monitored by a diagnosis function. A high current of the rotor circuit over a long time can destroy the ETC components.



Statechart: ETC actuator diagnosis

Description:

The following errors symptoms can be distinguished :

- PWM error of the ETC system
- Error in small signal detection
- Error in large signal detection

Error-symptoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> PWM outside range          (= SYM_0)
│ │ └───> -                          (= SYM_1)
│ └─────> small signal detection     (= SYM_2)
└───────> large signal detection     (= SYM_3)
```

Indicated error-symptoms are saved at the end of the software module. Only the last error entry is saved in the generic error management. The error symptom SYM_0 has lower priority than the other symptoms. Only one symptom can be active at a time.

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset:

All error variables ...MTC_CTL, T_TPS_DIF_DIAG, TPS_SP_ST_DIAG, MTCPWM_MMV and lv_tps_dif_dir to zero
(except LV_END_DIAG_MTC_CTL); T_TPS_DIF_SP = C_T_TPS_DIF_SP

LV_END_DIAG_MTC_CTL is initialised with 0
at every LV_DC = 0 → 1 **or** reset

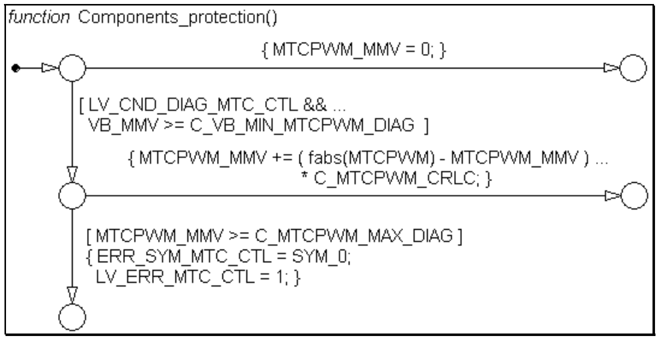
Recurrence: 10 ms

Activation:

if	LV_IGK	**and**
	LV_ERR_MTC_CTL == 0	**and**
	LV_INH_DIAG_MTC_CTL == 0	
then	LV_CDN_DIAG_MTC_CTL = 1	
else	LV_CDN_DIAG_MTC_CTL = 0	
endif		

Components protection

The following function monitors the absolute value of the digital position controller output. If the moving mean value exceeds the diagnosis threshold than an error will be indicated. The calculation of the moving mean value is done by low pass first order. The corresponding correlation factor specifies the speed of whose calculation.



Function: Protection of the ETC components

Formula section:

if LV_CDN_DIAG_MTC_CTL **and** (VB_MMV >= C_VB_MIN_MTCPWM_DIAG)

then *{ Calculation of the MTCPWM moving mean value by a low pass filter first order ! }*

MTCPWM_MMV += [| MTCPWM | - MTCPWM_MMV] * C_MTCPWM_CRLC

{ MTCPWM diagnostic condition depend on moving mean value }

if MTCPWM_MMV >= C_MTCPWM_MAX_DIAG

then ERR_SYM_MTC_CTL = SYM_0 *{ PWM outside range }*

LV_ERR_MTC_CTL = 1 *{ set for this DC }*

endif

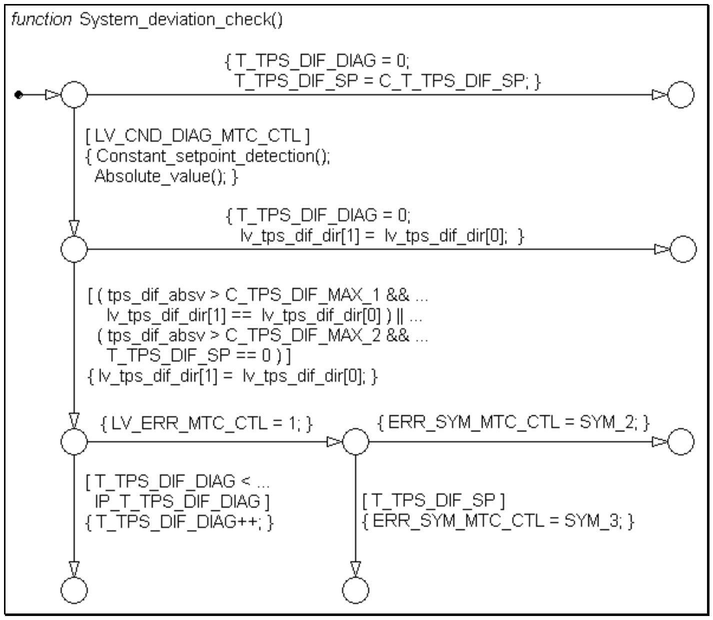
else *{ reset of the moving mean value }*

MTCPWM_MMV = 0

endif

System deviation check

This functionality monitors the absolute value of the system deviation. The algorithm is split in a small and a large signal range detection. The small signal range detection of the system deviation is only active after constant throttle position setpoint detection. The large signal range detection is done when the system deviation exceeds a upper threshold and its sign doesn't change.



Function: System deviation check

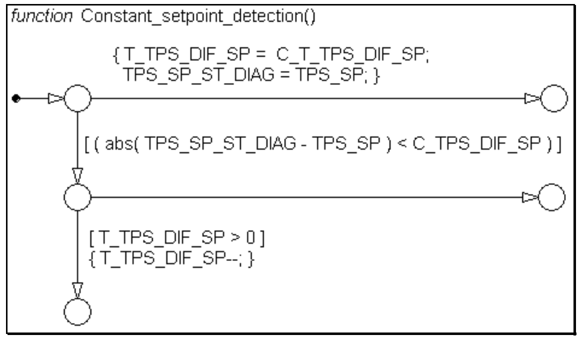
Formula section:

if LV_CDN_DIAG_MTC_CTL
then call Constant_setpoint_detection() *{ function for constant setpoint detection }*
call Absolute_value() *{ function for absolute value determination }*
if [tps_dif_absv > C_TPS_DIF_MAX_1 **and**
lv_tps_dif_dir(k-1) == lv_tps_dif_dir(k)] **or**
[tps_dif_absv > C_TPS_DIF_MAX_2 **and**
T_TPS_DIF_SP == 0]
then *{ save direction information for the next calculation step }*
lv_tps_dif_dir(k-1) = lv_tps_dif_dir(k)
if T_TPS_DIF_DIAG < IP_T_TPS_DIF_DIAG(VB_MMV)
then T_TPS_DIF_DIAG += 1 H
else LV_ERR_MTC_CTL = 1 *{ set for this DC }*
if T_TPS_DIF_SP
then ERR_SYM_MTC_CTL = SYM_3 *{ large signal detection }*
else ERR_SYM_MTC_CTL = SYM_2 *{ small signal detection }*
endif
endif
else *{ timer is reset }*
T_TPS_DIF_DIAG = 0
{ save direction information for the next calculation step }
lv_tps_dif_dir(k-1) = lv_tps_dif_dir(k)
endif
else *{ timers are reset }*
T_TPS_DIF_DIAG = 0
T_TPS_DIF_SP = C_T_TPS_DIF_SP

endif

Function for constant TPS setpoint detection

This function detects a constant throttle position setpoint. The small signal range detection will be started with constant setpoint recognition, otherwise the functionality is set to passive.



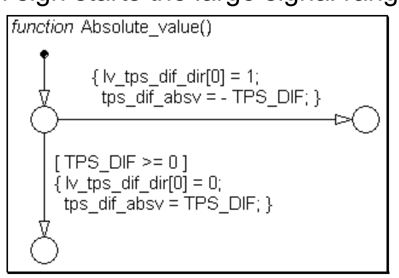
Function: Constant setpoint determination

Formula section:

if | TPS_SP_ST_DIAG – TPS_SP | < C_TPS_DIF_SP
then **if** T_TPS_DIF_SP > 0
then T_TPS_DIF_SP – = 1 H
endif
else *{ timer is reset }*
T_TPS_DIF_SP = C_T_TPS_DIF_SP
TPS_SP_ST_DIAG = TPS_SP
endif

Function for the absolute value determination

This function calculates the absolute value and indicates the sign of the system deviation. A change of the system deviation sign starts the large signal range detection newly again.



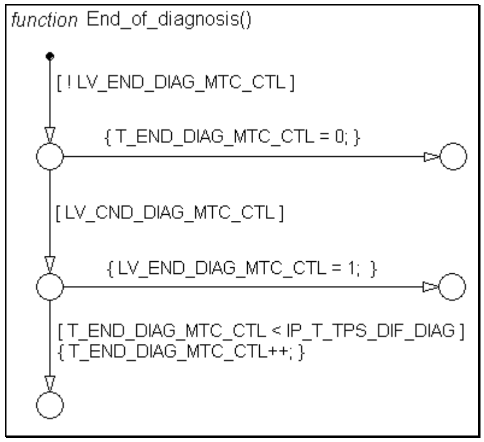
Function: Determination of absolute value

Formula section:

if TPS_DIF >= 0
then *{ set direction information }*
lv_tps_dif_dir(k) = 0
tps_dif_absv = TPS_DIF
else *{ set direction information and absolute value }*
lv_tps_dif_dir(k) = 1

```
        tps_dif_absv        = - TPS_DIF
endif
```

End of diagnosis, System deviation check



Function: End of diagnosis

Formula section:

if LV_END_DIAG_MTC_CTL == 0
then **if** LV_CDN_DIAG_MTC_CTL
    then **if** T_END_DIAG_MTC_CTL < IP_T_TPS_DIF_DIAG(VB_MMV)
        then T_END_DIAG_MTC_CTL += 1 H
        else LV_END_DIAG_MTC_CTL = 1 *{ set for this DC }*
        endif
    else T_END_DIAG_MTC_CTL = 0
    endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_TPS_DIF_MAX_1	1	0h...3FFFh	0...119.5	0.0073	°TPS
Maximum system deviation for large signal check					
C_TPS_DIF_MAX_2	1	0h...3FFFh	0...119.5	0.0073	°TPS
Maximum system deviation for small signal check					
C_TPS_DIF_SP	1	0h...3FFFh	0...119.5	0.0073	°TPS
Threshold for constant setpoint detection					
C_T_TPS_DIF_SP	1	0h...FFh	0 ... 2.55	0.01	s
Time counter for constant TPS setpoint detection					
IP_T_TPS_DIF_DIAG	2	0h...FFh	0 ... 2.55	0.01	s
LDP_VB_MMV_ T_TPS_DIF_DIAG	2	0h...FFh	0 ... 26	0.102	V
Debounce time depend on battery voltage					
C_MTCPWM_MAX_DIAG	1	0h ... 7FFFh	0...99.997	0.00305	%
Maximum permitted PWM output of the ETC position controller					
C_MTCPWM_CRLC	1	0h...FFFFh	0 ... 1	1.53e-5	-
Correlation constant for filtering of MTCPWM					
C_VB_MIN_MTCPWM_DIAG	1	0h...FFh	0 ... 26	0.102	V
Threshold depend on battery voltage for the activation of the MTCPWM monitoring					

Application hint

C_TPS_DIF_MAX_1 has to be set greater than C_TPS_DIF_MAX_2 !

A.14 ETC actuator diagnosis (appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_MTC_CTL	V/O	0h...1h	0...1	1	-
Inhibition condition for the ETC actuator diagnosis					

Input data:

LV_IGK	LV_ES	LV_ST	LV_TPS_AD_ACT
LV_MTC_CUR_OFF			

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_MTC_CTL permit to deactivate the corresponding diagnostic.

Description:

Application conditions:

Initialisation: at LV_IGK = 0 → 1 or reset,
LV_INH_DIAG_MTC_CTL = 1

Recurrence: 10 ms

Activation: LV_IGK == 1

Formula section:

LV_INH_DIAG_MTC_CTL = **(** LV_ES **or**
LV_ST **or**
LV_TPS_AD_ACT **or**
LV_MTC_CUR_OFF **)**

Configuration for diagnostic symptoms:

Diagnostic ETC	Symptoms	Nb	Filter type
ETC actuator diagnosis	*PWM error of the ETC system*	*0*	*NO*
		1	
	Error in small signal detection	2	
	Error in large signal detection	3	

Diagnostic ETC	Symptoms	Nb	P-Code/ Symptom	P-Code/ Global 1	P-Code/ Global 2	Failure class A/B	Readiness/ Code
ETC actuator diagnosis	*PWM error of the ETC system*	*0*	P2118	P21XX		CARB_CC	
		1					
	Error in small signal detection	2	P2118				
	Error in large signal detection	3	P2118				

{ P2118 – Throttle actuator control motor current range/performance }

A.15 ETC actuator adaptation, diagnosis

Output data:

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_TPS_AD	V/O	0...1H	0...1	1	-
Logic variable is set, if diagnosis conditions for TPS adaptation are fulfilled					
LV_ERR_TPS_AD	V/O	0...1H	0...1	1	-
Logic variable is set, if an adaptation error occurs					
ERR_SYM_TPS_AD	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for the TPS adaptation					
LV_END_DIAG_TPS_AD	V/O	0...1H	0...1	1	-
Logic variable is set, if end of diagnosis for the TPS adaptation is reached					
LV_CDN_DIAG_TPS_AD_SPR	V/O	0...1H	0...1	1	-
Logic variable is set, if diagnosis conditions for TPS adaptation are fulfilled					
LV_ERR_TPS_AD_SPR	V/O	0...1H	0...1	1	-
Logic variable is set, if a spring check error occurs during the adaptation					
ERR_SYM_TPS_AD_SPR	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for the spring check during the adaptation					
LV_END_DIAG_TPS_AD_SPR	V/O	0...1H	0...1	1	-
Logic variable is set, if end of diagnosis for the TPS adaptation is reached					
LV_CDN_DIAG_TPS_AD_GAIN	V/O	0...1H	0...1	1	-
Logic variable is set, if diagnosis conditions for TPS adaptation are fulfilled					
LV_ERR_TPS_AD_GAIN	V/O	0...1H	0...1	1	-
Logic variable is set, if the amplification of the measuring amplifier is outside the defined range					
ERR_SYM_TPS_AD_GAIN	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Error symptom for adaptation of the measuring amplifier					
LV_END_DIAG_TPS_AD_GAIN	V/O	0...1H	0...1	1	-
Logic variable is set, if end of diagnosis for the TPS adaptation is reached					
LV_CDN_DIAG_TPS_ST_CHK	V/O	0...1H	0...1	1	-
Logic variable is set, if the diagnosis conditions for the TPS start check are fulfilled					
LV_ERR_TPS_ST_CHK	V/O	0...1H	0...1	1	-
Logic variable is set, if an error occurs in the TPS start check					
ERR_SYM_TPS_ST_CHK	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Error symptom for the TPS start check					
LV_END_DIAG_TPS_ST_CHK	V/O	0...1H	0...1	1	-
Logic variable is set, if end of diagnosis for the TPS start check is reached					
LV_TPS_AD_CUR_OFF	V/O	0...1H	0...1	1	-
Logical variable to switch of the ETC controller during TPS adaptation					
LV_TPS_AD_REQ _ST_CHK	V/O	0...1H	0...1	1	-
TPS start check adaptation request					
LV_INH_TPS_GAIN_1	V/O	0...1H	0...1	1	-
Inhibition condition for the amplified TPS 1 channel					

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
V_TPS_BOL_OFS	V	0...FFFH	0...0.3125	7.631e-5	V
Temporary offset of the continuously BOL adaptation					
V_TPS_AD_LIH_1	V/O/S	0...3FFH	0...4.9951	0.00488	V
Adaptation value for the limp home position, TPS-channel 1					
V_TPS_AD_LIH_2	V/O/S	0...3FFH	0... 4.9951	0.00488	V
Adaptation value for the limp home position, TPS-channel 2					
V_TPS_AD_BOL_1	V/S	0...3FFH	0... 4.9951	0.00488	V
Adaptation value for the lower stop, TPS-channel 1					
V_TPS_AD_BOL_2	V/S	0...3FFH	0... 4.9951	0.00488	V
Adaptation value for the lower stop, TPS-channel 2					
V_TPS_GAIN_AD_BOL	V/S	0...3FFH	0... 4.9951	0.00488	V
Adaptation value for the lower mechanical stop of the amplification TPS 1					
V_TPS_AD_EL_BOL_1	V/O	0...3FFH	0... 4.9951	0.00488	V
Temporary adaptation value for the lower stop, TPS-channel 1					
V_TPS_AD_EL_BOL_2	V/O	0...3FFH	0... 4.9951	0.00488	V
Temporary adaptation value for the lower stop, TPS-channel 2					
V_TPS_GAIN _AD_EL_BOL	V/O	0...3FFH	0... 4.9951	0.00488	V
Temporary adaptation value for the lower mechanical stop of the amplified TPS 1					
V_TPS_AD_TOL_1	V	0...3FFH	0... 4.9951	0.00488	V
Upper position value TPS 1 for the adaptation of the TPS 1 amplifier					
V_TPS_AD_TOL_GAIN_1	V	0...3FFH	0... 4.9951	0.00488	V
Upper amplified position value TPS 1 for the adaptation of the TPS 1 amplifier					
TPS_AD_SLOP_GAIN_1	V/O/S	0... FFFFH	0...7.9999	0.000122	-
Adaptation value of the amplification factor of amplified TPS 1					
TPS_SLOP	V/O	0...FFFFH	0...47.9993	7.3e-4	°TPS/V
Signal slope of the throttle position sensor 1 and 2					
TPS_GAIN_SLOP	V/O	0...FFFFH	0...47.9993	7.3e-4	°TPS/V
Signal slope of the amplified throttle position sensor 1					
TPS_LIH_INI	V/O	0...3FFFH	0...119.5	0.0073	°TPS
Limp home position of the throttle valve – initialization value					
TPS_LIH_1	V/O/S	0...3FFFH	0...119.5	0.0073	°TPS
Limp home position of the throttle valve of TPS 1					
TPS_LIH_2	V/O/S	0...3FFFH	0...119.5	0.0073	°TPS
Limp home position of the throttle valve of TPS 2					
TPS_SP_AD	V/O	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint during adaptation					
TPS_AD_STEP	V/O	00H 01H 02H 03H 04H 05H 06H 07H 08H 09H 0AH 0BH 0CH 0DH 0EH	ST_CHK_LIH ST_GO_TOL ST_SPR_CHK ST_CHK_END AD_INIT AD_LIH_POS AD_GO_BOL AD_BOL_POS AD_GO_LIH AD_SPR_CHK_1 AD_GO_TOL_1 AD_TOL_POS AD_GO_TOL_2 AD_SPR_CHK_2 AD_END	1	-
Steps of the TPS adaptation or start check					
STATE_TPS_BOL_AD	V	0H 1H 2H 3H	BOL_RESET BOL_WAIT BOL_INC BOL_DEC	1	-
State variable indicates the state of the continuously BOL adaptation					
T_TPS_BOL_OFS_INC	V	0...FFFFH	0 ... 2621.4	0.04	s
Waiting period before offset incrementation, continuously BOL adaptation					

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
T_TPS_BOL_OFS_DEC	V	0...FFFFH	0 ... 2621.4	0.04	s
Waiting period before offset decrementation, continuously BOL adaptation					
INH_IV_TPS_AD	V/O	0...FFH	0...255	1	-
Variable to inhibit injection and ignition during TPS adaptation					
CTR_ST_CHK_LIH_OK	V/O/S	0...FFFFH	0...65535	1	-
Start check, limp home check 1 passed					
CTR_ST_CHK_LIH _ERR	V/O/S	0...FFFFH	0...65535	1	-
Start check, limp home check 1 not passed					
T_DLY_TPS_AD	V	0...FFH	0...1.275	0.005	[s]
Waiting period after ignition switch-on, to get plausible battery voltage					

Input data:

LV_IGK	LV_ST	LV_TPS_AD_ACT	LV_TPS_AD_REQ
LV_INH_TPS_AD	V_TPS_1	V_TPS_2	V_TPS_GAIN_1
TPS_ETC	TPS_SP	MTCPWM	TCO
LV_ERR_TPS	LV_MTC_CUR_OFF	NC_CYL_NR	LV_TPS_AD_REQ_VB_LOW
STATE_DR_MU_MC			

FUNCTION DESCRIPTION:

General information:

The throttle position is determined by a two-channeled sensor. Both channels deliver inverse dispersing voltage signals. In order to reduce the measurement inaccuracy, the two signal voltages will be referenced to their supply voltage. After the initial engine start and component change the sensor characteristic is learnt within an adaptation routine. The learned values for the limp-home position, the lower mechanical stop and the TPS 1 amplification are stored at the end of the driving cycle as „non-volatile“.

A.15.1 Initialisation of TPS adaptation and start check

Description:

At start of the TPS adaptation or start check the following initialisation is necessary. *{ This whole chapter is initialisation, it has to be done before initialisation functions of other ETC modules. }*

Application conditions:

Initialisation: at reset **or** LV_IGK = 0 ⇒ 1 **or** LV_TPS_AD_ACT = 0 ⇒ 1 **or**
LV_TPS_AD_REQ _ST_CHK == 1 *{ re-initialisation after start check request}* **or**
LV_TPS_AD_REQ_VB_LOW == 1 *{re-initialisation after VB low during adaptation}*

```
{ Nominal slope of TPS 1 and 2 }
TPS_SLOP                    = C_TPS_SLOP
{ Is an adaptation demanded ? }
if    LV_TPS_AD_REQ == 0
then  { initialisation of time counter }
      t_max_tps_ad          = C_T_MAX_TPS_LIH_CHK
      { Initialisation of the inhibition condition for the amplified TPS 1 channel }
      if    LC_TPS_AD_GAIN_ENA == 1
      then  LV_INH_TPS_GAIN_1 = 0
      else  LV_INH_TPS_GAIN_1 = 1
      endif
```

{ next adaptation step }
TPS_AD_STEP = ST_CHK_LIH
{ Initialization value of the limp-home position }
TPS_LIH_INI = $0.5 \cdot [\mathrm{TPS\_LIH\_1} + \mathrm{TPS\_LIH\_2}]$

else *{ Set inhibition condition for the selection of the amplified TPS 1 channel }*
LV_INH_TPS_GAIN_1 = 1
{ initialisation of the adaptation values }
V_TPS_AD_LIH_X = C_V_TPS_SP_LIH_X
V_TPS_AD_BOL_X = C_V_TPS_AD_BOL_INI_X
V_TPS_GAIN_AD_BOL = C_V_TPS_GAIN_AD_BOL_INI
{ Initialization value of the limp-home position }
TPS_LIH_INI = ...

$$0.5 \cdot \mathrm{TPS\_SLOP} \cdot [\mathrm{C\_V\_TPS\_SP\_LIH\_1} - \mathrm{C\_V\_TPS\_SP\_LIH\_2} \ldots - \mathrm{C\_V\_TPS\_AD\_BOL\_INI\_1} + \mathrm{C\_V\_TPS\_AD\_BOL\_INI\_2}]$$

TPS_LIH_X = TPS_LIH_INI
{ amplification value of the measuring amplifier }
TPS_AD_SLOP_GAIN_1 = C_TPS_AD_SLOP_GAIN_INI
{ initialisation of time counter }
t_max_tps_ad = C_T_MAX_TPS_AD
{ first adaptation step }
TPS_AD_STEP = AD_INIT

endif
{ Set temporary electrical adaptation values ! }
V_TPS_AD_EL_BOL_X = V_TPS_AD_BOL_X
V_TPS_GAIN_AD_EL_BOL = V_TPS_GAIN_AD_BOL
if LC_TPS_AD_GAIN_ENA == 1
then *{ Calculate the slope of the amplified TPS 1 channel ! }*

$$\mathrm{TPS\_GAIN\_SLOP} = \frac{\mathrm{TPS\_SLOP}}{\mathrm{TPS\_AD\_SLOP\_GAIN\_1}}$$

else *{ The amplified TPS 1 channel isn't supported by ECU ! }*

$$\mathrm{TPS\_GAIN\_SLOP} = 0$$

endif
{ TPS start check and adaptation throttle position setpoint request }
TPS_SP_AD = TPS_LIH_1
{ Initialization of the auxiliary variables }
t_step_tps_ad = 0
t_tps_ad_lih = 0
v_tps_x_buf = V_TPS_X
{ to get a plausible battery voltage, waiting period after ECU switch on }
T_DLY_TPS_AD = 0
{ TPS start check adaptation request }
LV_TPS_AD_REQ _ST_CHK = 0
{switch off request ETC power stage }
LV_TPS_AD_CUR_OFF = 1
{inhibit injection for all cylinder }
INH_IV_TPS_AD = (2^NC_CYL_NR) – 1

A.15.2 Diagnosis conditions and error symptom definition

Description:

The diagnosis condition for the TPS adaptation is true as long as the adaptation is active and the diagnosis condition for the TPS start check is also true as long as the start check is active.

The end of diagnosis in error free case is set on the end of the TPS start check / adaptation or latest, if a error is set.

The error symptoms are defined for this diagnosis as:

ERR_SYM_TPS_AD (TPS adaptation)

```
□□□□
│ │ │ └─> adaptation conditions exceeded              (= SYM_0)
│ │ └───> limp-home adaptation outside range          (= SYM_1)
│ └─────> adaptation of the amplifier not possible    (= SYM_2)
└───────> lower mec. stop adaptation outside range    (= SYM_3)
```

ERR_SYM_TPS_AD_SPR (TPS adaptation)

```
□□□□
│ │ │ └─> lower position not reached                  (= SYM_0)
│ │ └───> error lower return spring check             (= SYM_1)
│ └─────> upper position not reached                  (= SYM_2)
└───────> error upper return spring check             (= SYM_3)
```

ERR_SYM_TPS_AD_GAIN (TPS adaptation)

```
□□□□
│ │ │ └─> adapted amplification too low               (= SYM_0)
│ │ └───> adapted amplification too high              (= SYM_1)
│ └─────> -                                           (= SYM_2)
└───────> -                                           (= SYM_3)
```

ERR_SYM_TPS_ST_CHK (TPS start check)

```
□□□□
│ │ │ └─> error in spring check                       (= SYM_0)
│ │ └───> error in limp-home check                    (= SYM_1)
│ └─────> -                                           (= SYM_2)
└───────> -                                           (= SYM_3)
```

Application conditions:

Initialisation: at reset **or** LV_IGK = 0 ⇒ 1 **or** LV_TPS_AD_ACT = 0 ⇒ 1 :

Set all error variables ..._TPS_AD... and ..._TPS_ST_CHK to zero !

Recurrence: 5ms

Activation: LV_IGK == 1

Deactivation: When the activation condition isn't fulfilled.

Formula section:

{ The diagnosis conditions are calculated in front of the adaptation or start check algorithm ! }

```
if      LV_TPS_AD_ACT == 1
then    { Wait C_T_DLY_TPS_AD ms after ignition on to get a plausible battery voltage. The global
        time delay can be finished when ECM3 pre drive check is done and after defined time delay ! }
        if      T_DLY_TPS_AD >=  C_T_DLY_TPS_AD          or
                ( STATE_DR_MU_MC ==  ENABLE      and
                  T_DLY_TPS_AD >=  C_T_DLY_MIN_TPS_AD )
        then    { Set delay time to zero additionally ! }
                T_DLY_TPS_AD = C_T_DLY_TPS_AD
                if    LV_TPS_AD_REQ
                then    { Diagnosis TPS adaptation active }
                        LV_CDN_DIAG_TPS_AD          = 1
                        LV_CDN_DIAG_TPS_AD_SPR      = 1
                        if      LC_TPS_AD_GAIN_ENA == 1    and
                                LV_ERR_TPS_AD_GAIN == 0
                        then    LV_CDN_DIAG_TPS_AD_GAIN = 1
                        else    LV_CDN_DIAG_TPS_AD_GAIN = 0
                        endif
                        LV_CDN_DIAG_TPS_ST_CHK      = 0
                else    { Diagnosis TPS start check active }
                        LV_CDN_DIAG_TPS_AD          = 0
                        LV_CDN_DIAG_TPS_AD_SPR      = 0
                        LV_CDN_DIAG_TPS_AD_GAIN     = 0
                        LV_CDN_DIAG_TPS_ST_CHK      = 1
                endif
        else    { Increase delay counter ! }
                T_DLY_TPS_AD + = 5 ms
        endif
else    { TPS adaptation and start check finished or stopped ! }
        { The conditions are set only to zero if : LV_TPS_AD_ACT = 1⇒ 0 }
        LV_CDN_DIAG_TPS_AD          = 0
        LV_CDN_DIAG_TPS_AD_SPR      = 0
        LV_CDN_DIAG_TPS_AD_GAIN     = 0
        LV_CDN_DIAG_TPS_ST_CHK      = 0
endif
{ The end of diagnoses are calculated after the adaptation or start check algorithm ! }
if      LV_TPS_AD_ACT == 1
then    { Calculation: end of diagnosis TPS adaptation }
        if      LV_TPS_AD_REQ == 0          or      { Adaptation successful, no request }
                LV_ERR_TPS_AD == 1          or      { TPS adaptation error }
                LV_ERR_TPS_AD_SPR == 1      or      { TPS adaptation error }
```

```
            TPS_AD_STEP == AD_END              { TPS adaptation finished successfully }
      then  { Set end of diagnosis information }
            LV_END_DIAG_TPS_AD = 1
            LV_END_DIAG_TPS_AD_SPR   = 1
            if      LC_TPS_AD_GAIN_ENA == 1
            then    LV_END_DIAG_TPS_AD_GAIN = 1
            endif
      endif
      { Calculation: end of diagnosis TPS start check }
      if    LV_ERR_TPS_ST_CHK            or    { TPS start check error }
            TPS_AD_STEP == ST_CHK_END          { TPS start check finished successfully }
      then  { Set end of diagnosis information }
            LV_END_DIAG_TPS_ST_CHK = 1
      endif
endif
```

A.15.3 Co-ordination of the adaptation functions

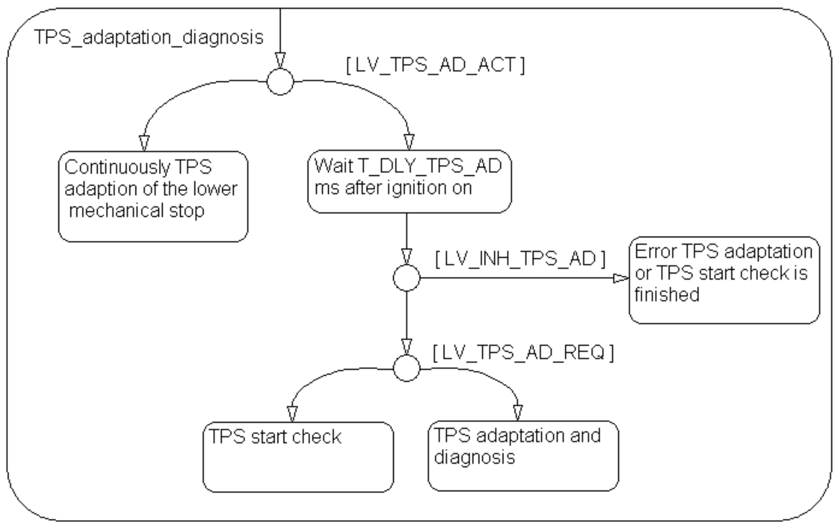
Description:

The adaptation of the throttle position includes two sections: the start routine and adaptation routine. The start routine is a part of the adaptation routine with owns thresholds and is carry out at every switch-on of the engine control unit. The adaptation routine starts only if the adaptation is requested.

The actual adaptation condition LV_TPS_AD_ACT and the adaptation request LV_TPS_AD_REQ are defined in the corresponding application incidence.

To get a plausible battery voltage the check of adaptation or start check is delayed T_DLY_TPS_AD after control unit is switched on.

Signal flow diagram:



Application conditions:

Initialisation: see above

Recurrence: 5 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition isn't fulfilled.

Formula section:

```
if      LV_TPS_AD_ACT
then    { Wait C_T_DLY_TPS_AD ms after ignition on to get a plausible battery voltage. The global
        time delay can be finished when ECM3 pre drive check is done and after defined time delay ! }
        if      T_DLY_TPS_AD >=  C_T_DLY_TPS_AD           or
                ( STATE_DR_MU_MC ==  ENABLE       and
                  T_DLY_TPS_AD >=  C_T_DLY_MIN_TPS_AD )
        then    { Inhibition bit for TPS adaptation, is set in the corresponding application incidence }
                if      LV_INH_TPS_AD
                then    { Is an adaptation requested ? }
                        if    LV_TPS_AD_REQ
                        then    { Error TPS adaptation, inhibition bit is set }
                                ERR_SYM_TPS_AD = SYM_0
                                LV_ERR_TPS_AD = 1     {set for DC without debounce}
                        else    { Start check is finished ! }
                                TPS_AD_STEP            = ST_CHK_END
                                { switch on request ETC power stage }
                                LV_TPS_AD_CUR_OFF  = 0
                                { enable injection for all cylinder }
                                INH_IV_TPS_AD          = 0
                        endif
                else    { Is an adaptation requested ? }
                        if    LV_TPS_AD_REQ
                        then    { see chapter „TPS adaptation and diagnosis“ }
                                call function tps_adaptation()
                        else    { see chapter „TPS start check and diagnosis“ }
                                call function start_check()
                        endif
                endif
                if      LV_ERR_TPS_AD          == 1    or
                        LV_ERR_TPS_AD_SPR  == 1
                then    { Reset adaptation values in case of adaptation errors ! }
                        V_TPS_AD_LIH_X              = C_V_TPS_SP_LIH_X
                        V_TPS_AD_BOL_X              = C_V_TPS_AD_BOL_INI_X
                        V_TPS_AD_EL_BOL_X           = C_V_TPS_AD_BOL_INI_X
                        V_TPS_GAIN_AD_BOL           = C_V_TPS_GAIN_AD_BOL_INI
                        V_TPS_GAIN_AD_EL_BOL        = C_V_TPS_GAIN_AD_BOL_INI
                        { Initialization value of the limp-home position }
                        TPS_LIH_INI                 = C_TPS_LIH_INI
                        TPS_LIH_X                   = C_TPS_LIH_INI
                        { Set inhibition condition for the selection of the amplified TPS 1 channel }
                        LV_INH_TPS_GAIN_1           = 1
                endif
```

endif

else

{ switch on request ETC power stage }

LV_TPS_AD_CUR_OFF = 0

{ enable injection for all cylinder }

INH_IV_TPS_AD = 0

Comment: The continuously lower mechanical stop adaptation is now executed !

endif

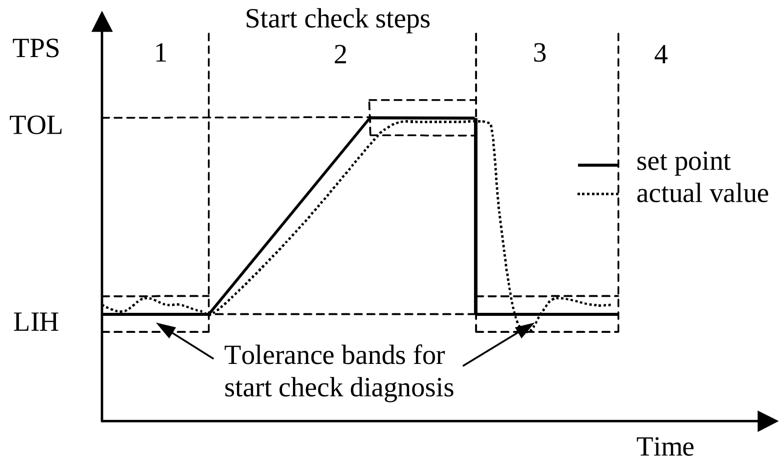
A.15.4 TPS start check and diagnosis

Description:

The Start routine is carried out with every switch-on of the engine control unit (LV_IGK = 1). The TPS-adaptation bit is active initialised (LV_TPS_AD_ACT =1) for the start-check or adaptation-routine. The routine includes the following:

- Limp-home position check
- Adaptation of the limp-home position
- Upper return spring check

The start check is only carried out when the adaptation conditions are maintained otherwise it will be ignored without error entry.



The state variable TPS_AD_STEP indicates the following start check steps:

Adap. area	TPS_AD_STEP	Note
1	ST_CHK_LIH	Check and adaptation of the limp-home position
2	ST_GO_TOL	throttle flap goes into a upper position
3	ST_SPR_CHK	Upper return spring and limp-home check
4	ST_CHK_END	End of start check

Formula section:

```
start_check( )
    switch( TPS_AD_STEP )
        case ST_CHK_LIH:        call function st_chk_lih( )
                                break
        case ST_GO_TOL:         call function st_go_tol( )
                                break
        case ST_SPR_CHK:        call function st_spr_chk( )
                                break
        default:                { TPS start check successful finished }
                                TPS_AD_STEP             = ST_CHK_END
                                { set new ETC setpoint }
                                TPS_SP_AD               = TPS_LIH_1
                                { switch on request ETC power stage }
                                LV_TPS_AD_CUR_OFF       = 0
                                { enable injection for all cylinder }
                                INH_IV_TPS_AD           = 0
    end switch
end function
```

A.15.4.1 function st_chk_lih, check of the limp-home position

Description:

When checking the limp-home position the throttle valve is staying without current hold by spring power in the limp-home position. The actual sensor-signals are compared with their adaptations in the non-volatile memory.

If the actual voltage values during the time interval are outside the hysteresis then no agreement is recognised and a new adaptation is necessary or an engine start must be cancelled.

Formula section:

```
st_chk_lih()
    if      t_max_tps_ad
    then    t_max_tps_ad - = 5 ms
            if      | V_TPS_x – V_TPS_AD_LIH_x | ≤ C_V_TPS_LIH_CHK_HYS
            then    check of the LIH position
                    t_step_tps_ad + = 5 ms
                    if      t_step_tps_ad > C_T_DYW_TPS_LIH_CHK
                    then    Limp-home check was successful !
                            CTR_ST_CHK_LIH_OK + = 1 h
                            { initialisation for the next adaptation step }
                            t_step_tps_ad          = 0
                            t_max_tps_ad           = C_T_MAX_TPS_SP_RST_CHK
                            { enable injection for all cylinder }
                            INH_IV_TPS_AD          = 0
                            { switch on request ETC power stage }
                            LV_TPS_AD_CUR_OFF      = 0
                            { next adaptation step }
                            TPS_AD_STEP            = ST_GO_TOL
                    endif
                    { continuously LIH adaptation enable or disable  }
                    if      LC_TPS_AD_LIH_ENA
                    then    continuously adaptation of the LIH position
                            { If the difference of one of the two sensors is greater than
                            C_V_TPS_AD_HYS ! }
                            if      | v_tps_x_buf - V_TPS_x | > C_V_TPS_AD_HYS
                            then    v_tps_x_buf = V_TPS_x
                                    t_tps_ad_lih= 0
                            endif
                            t_tps_ad_lih + = 5 ms
                            if      t_tps_ad_lih > C_T_TPS_AD_HYS
                            then    if      | V_TPS_x – C_V_TPS_SP_LIH_x | ≤ ...
                                            C_V_TPS_AD_WIN_LIH
                                    then    { calculate adaptation values for LIH - position}
```

$$V\_TPS\_AD\_LIH\_x = \frac{v\_tps\_x\_buf + V\_TPS\_x}{2}$$

{ initialisation of the limp-home position for TPS 1 and 2 }

$$TPS\_LIH\_1 = [V\_TPS\_AD\_LIH\_1 - ... V\_TPS\_AD\_BOL\_1] \cdot TPS\_SLOP$$

$$TPS\_LIH\_2 = [V\_TPS\_AD\_BOL\_2 - ... V\_TPS\_AD\_LIH\_2] \cdot TPS\_SLOP$$

$$TPS\_LIH\_INI = 0.5 \cdot [TPS\_LIH\_1 + TPS\_LIH\_2]$$

else *{ LIH position is outside of the adaptation window, a new adaptation is requested }*

LV_TPS_AD_REQ_ST_CHK = 1

{ Re-initialisation of the adaptation function ! }

TPS_AD_STEP = AD_INIT

endif

{ clear adaptation timer }

t_tps_ad_lih = 0

endif

endif

else *{ reset time counter }*

t_step_tps_ad = 0

t_tps_ad_lih = 0

v_tps_x_buf = V_TPS_x

endif

else Limp-home check was not successful !

CTR_ST_CHK_LIH_ERR + = 1 h

{ initialisation for the next adaptation step }

t_step_tps_ad = 0

t_max_tps_ad = C_T_MAX_TPS_SP_RST_CHK

{ injection is still inhibited }

INH_IV_TPS_AD = (2^NC_CYL_NR) - 1

{ switch on request ETC power stage }

LV_TPS_AD_CUR_OFF = 0

{ next adaptation step }

TPS_AD_STEP = ST_GO_TOL

endif

end function

A.15.4.2 function st_go_tol, throttle flap goes into upper position

Description:

This function is used to positioning the throttle flap into a position above limp-home. The position controller is switched on and the setpoint is increased by change limitation until the setpoint for testing the retracting spring is reached.

Now a timer is started. The actual value of the throttle position has to reach the setpoint within the hysteresis within a limit maximum of time.

Formula section:

st_go_tol()

```
    { engine start without start delay in case of injection is enable }
    if     ( LV_ST == 0 ) or INH_IV_TPS_AD
    then   if     TPS_SP_AD < C_TPS_SP_RST_CHK_BAS
           then   TPS_SP_AD += C_TPS_SP_RST_CHK_LGRD
                  if     TPS_SP_AD > C_TPS_SP_RST_CHK_BAS
                  then   TPS_SP_AD = C_TPS_SP_RST_CHK_BAS
                  endif
           else   if     t_max_tps_ad
                  then   t_max_tps_ad - = 5 ms
                         if     | TPS_ETC – C_TPS_SP_RST_CHK_BAS | ≤ ...
                                 C_TPS_SP_RST_CHK_HYS
                         then   { initialisation for the next adaptation step }
                                t_step_tps_ad          = 0
                                t_max_tps_ad           = C_T_MAX_TPS_LIH_CHK
                                { switch off request ETC power stage }
                                LV_TPS_AD_CUR_OFF      = 1
                                { next adaptation step }
                                TPS_AD_STEP            = ST_SPR_CHK
                         endif
                  else   { error in spring check }
                         ERR_SYM_TPS_ST_CHK   = SYM_0
                         LV_ERR_TPS_ST_CHK    = 1     {set for DC without debounce}
                  endif
           endif
    else   { Start check stopped }
           TPS_AD_STEP            = ST_CHK_END
           { set new ETC setpoint }
           TPS_SP_AD              = TPS_LIH_1
           { switch on request ETC power stage }
           LV_TPS_AD_CUR_OFF      = 0
           { enable injection for all cylinder }
           INH_IV_TPS_AD          = 0
    endif
end function
```

A.15.4.3 Function st_spr_chk(), check of the upper return spring

Description:

This function checks the function of the upper return spring. The ETC power stage is switched off and than the throttle has to return by spring power in the limp-home position, otherwise a return spring error is present. Depending on the result of the test the start routine is finished, starts the adaptation or indicates an start check error (engine speed limitation and ETC power stage deactivation have to be requested)

Formula section:

st_spr_chk()

```
    { engine start without start delay in case of injection is enable }
    if      ( LV_ST == 0 ) or INH_IV_TPS_AD
    then    if      t_max_tps_ad
            then    t_max_tps_ad - = 5 ms
                    if      | V_TPS_x – V_TPS_AD_LIH_x | ≤ C_V_TPS_LIH_CHK_HYS
                    then    t_step_tps_ad + = 5 ms
                            if      t_step_tps_ad > C_T_DYW_TPS_LIH_CHK
                            then    { next adaptation step }
                                    TPS_AD_STEP          = ST_CHK_END
                                    { set new ETC setpoint }
                                    TPS_SP_AD            = TPS_LIH_1
                                    { switch on request ETC power stage }
                                    LV_TPS_AD_CUR_OFF    = 0
                                    { enable injection for all cylinder }
                                    INH_IV_TPS_AD        = 0
                            endif
                    else    { reset time counter }
                            t_step_tps_ad = 0
                    endif
            else    if      INH_IV_TPS_AD == 0
                    then    { error in limp-home check }
                            ERR_SYM_TPS_ST_CHK      = SYM_1
                            LV_ERR_TPS_ST_CHK       = 1    {set for DC without debounce}
                    else    { limp-home not passed, request an new TPS adaptation }
                            LV_TPS_AD_REQ_ST_CHK    = 1
                            { Re-initialisation of the adaptation function ! }
                            TPS_AD_STEP             = AD_INIT
                    endif
            endif
    else    { Start check stopped }
            TPS_AD_STEP             = ST_CHK_END
            { set new ETC setpoint }
            TPS_SP_AD               = TPS_LIH_1
            { switch on request ETC power stage }
            LV_TPS_AD_CUR_OFF       = 0
            { enable injection for all cylinder }
            INH_IV_TPS_AD           = 0
    endif
end function
```

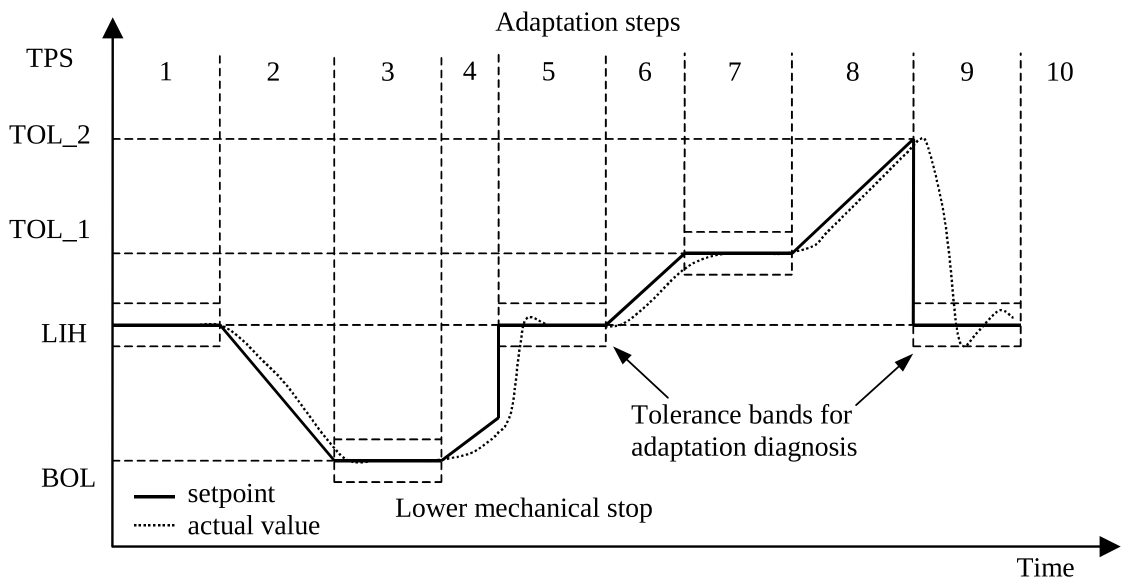
A.15.5 TPS adaptation and diagnosis

Description:

Due to the electrical and the mechanical tolerances of the throttle position system, the sensor characteristic has to be learnt. The adaptation and diagnosis of the lower mechanical stop and limp-home position occurs within the adaptation routines. The adapted lower mechanical stop and the upper check position are used for the calculation of the amplification of the amplified TPS 1 channel. The upper mechanical stop is not learned. During the TPS adaptation no limitation of the throttle position setpoint is active. The learned value for the limp-home position, the lower mechanical stop and the TPS 1 amplification are stored at the end of the driving cycle as „non-volatile".

The TPS adaptation includes the following functions:

- Adaptation and check of the limp-home position
- Adaptation and check of the lower mechanical stop
- Lower return spring check
- Adaptation of the amplifier amplification (TPS 1 channel)
- Upper return spring check



The state variable TPS_AD_STEP indicates the following adaptation steps:

Steps	TPS_AD_STEP	Note
1	AD_LIH_POS	Adaptation and check of the limp-home position
2	AD_GO_BOL	Throttle flap drives in the lower mechanical stop
3	AD_BOL_POS	Adaptation and check of the lower mechanical stop
4	AD_GO_LIH	Throttle flap drives in a position below throttle flap LIH
5	AD_SPR_CHK_1	Check of the lower return spring
6	AD_GO_TOL_1	Throttle drives in a position above throttle flap LIH
7	AD_TOL_POS	Adaptation position of the measuring amplifier
8	AD_GO_TOL_2	Throttle drives in a position above throttle flap LIH
9	AD_SPR_CHK_2	Check of the upper return spring
10	AD_END	Successful adaptation end

Formula section:

tps_adaptation( )

switch(TPS_AD_STEP **)**

{ At the beginning of the adaptation the adaptation values are described with its default-values, the ETC power stage is switched off and injection is disable. }

case AD_INIT:

{ This adaptation step is used only for the adaptation diagnosis to indicate the error location in the state variable TPS_AD_STEP. The initialisation of the adaptation function is done in the chapter "Initialisation of TPS adaptation ...", see above ! }

{ Go to the next adaptation step ! }
TPS_AD_STEP = AD_LIH_POS
break

case AD_LIH_POS: **call function** ad_lih_pos( )
break

case AD_GO_BOL: **call function** ad_go_bol( )
break

case AD_BOL_POS: **call function** ad_bol_pos( )
break

case AD_GO_LIH: **call function** ad_go_lih( )
break

case AD_SPR_CHK_1: **call function** ad_spr_chk_1( )
break

case AD_GO_TOL_1: **call function** ad_go_tol_1( )
break

case AD_TOL_POS: **call function** ad_tol_pos( )
break

case AD_GO_TOL_2: **call function** ad_go_tol_2( )
break

case AD_SPR_CHK_2: **call function** ad_spr_chk_2( )
break

default:
{ adaptation successful finished }
TPS_AD_STEP = AD_END
{ set new ETC setpoint }
TPS_SP_AD = TPS_LIH_1
{ switch on request ETC power stage }
LV_TPS_AD_CUR_OFF = 0
{ enable injection for all cylinder }
INH_IV_TPS_AD = 0

end switch

end function

A.15.5.1 Function ad_lih_pos(), adaptation of the limp-home position

Description:

The first step is the adaptation of the limp-home position. The actuator is without current and is forced by the spring power in the limp home position.

During the adaptation all the voltage values must be in the adaptation windows. A hysteresis band is set up around each of the first recorded values (TPS 1 and 2). If all the values lie in this hysteresis band during an adjustable time then the adaptation values are determined from each first and last value. If the TPS - value lies outside the hysteresis band then the process is newly started for both TPS. The learning algorithm is limited by the maximum time. This procedure is the same for the Limp home position and the lower mechanical stop. Just the calibration values are different. An error is detected if the adaptation could not be carried out during the maximum time C_T_MAX_TPS_AD.

Formula section:

```
ad_lih_pos()
    if      t_max_tps_ad
    then    t_max_tps_ad - = 5 ms
            if      | V_TPS_X – C_V_TPS_SP_LIH_X | ≤ C_V_TPS_AD_WIN_LIH
            { If the difference of one of the two sensors is greater than C_V_TPS_AD_HYS ! }
            then    if      | v_tps_x_buf - V_TPS_X | > C_V_TPS_AD_HYS
                    then    v_tps_x_buf = V_TPS_X
                            t_step_tps_ad= 0
                    endif
                    t_step_tps_ad + = 5 ms
                    if      t_step_tps_ad > C_T_TPS_AD_HYS
                    then    { calculate adaptation values for LIH - position}
                            V_TPS_AD_LIH_x = (v_tps_x_buf + V_TPS_x) / 2
                            { initialisation for the next adaptation step }
                            -
                            { switch on request ETC power stage }
                            LV_TPS_AD_CUR_OFF  = 0
                            { next adaptation step }
                            TPS_AD_STEP        = AD_GO_BOL
                    endif
            else    { reset time counter }
                    t_step_tps_ad = 0
                    v_tps_x_buf   = V_TPS_X
            endif
    else    { limp-home adaptation outside range }
            ERR_SYM_TPS_AD = SYM_1
            LV_ERR_TPS_AD  = 1                      {set for DC without debounce}
    endif
end function
```

$$V\_TPS\_AD\_LIH\_x = \frac{v\_tps\_x\_buf + V\_TPS\_x}{2}$$

A.15.5.2 Function ad_go_bol(), throttle flap goes into lower mechanical stop

Description:

This function is used to drive the throttle flap into the lower mechanical stop. The position controller is switched-on and the setpoint is decreased by change limitation.

Formula section:

```
ad_go_bol ()
    if      TPS_SP_AD > 0
    then    TPS_SP_AD - = C_TPS_SP_BOL_LGRD
    else    { initialisation for the next adaptation step }
            t_max_tps_ad       = C_T_MAX_TPS_AD
            t_step_tps_ad      = 0
            v_tps_x_buf        = V_TPS_X
            v_tps_gain_1_buf   = V_TPS_GAIN_1
            { next adaptation step }
            TPS_AD_STEP        = AD_BOL_POS
    endif
end function
```

A.15.5.3 Function ad_bol_pos(), adaptation of the lower mechanical stop

Description:

As soon as the mechanical stop is reached, the adaptation of the lower mechanical stop for all TPS input channels is started.

During the adaptation all the voltage values must be in the adaptation windows. A hysteresis band is set up around each of the first recorded values (TPS 1 and 2). If all the values lie in this hysteresis band during an adjustable time then the adaptation values are determined from each first and last value. If the TPS - value lies outside the hysteresis band then the process is newly started for both TPS. The learning algorithm is limited by the maximum time. If the maximum adaptation time is exceeded, a adaptation error is indicated.

Formula section:

```
ad_bol_pos()
    if      t_max_tps_ad
    then    t_max_tps_ad - = 5 ms
            if      | V_TPS_X – C_V_TPS_SP_BOL_ | ≤ C_V_TPS_AD_WIN_BOL
            { If the difference of one of the two sensors is greater than C_V_TPS_AD_HYS ! }
            then    if      | v_tps_x_buf – V_TPS_x | > C_V_TPS_AD_HYS
                    then    { reinitialisation of v_tps_..._buf }
                            v_tps_x_buf        = V_TPS_X
                            v_tps_gain_1_buf   = V_TPS_GAIN_1
                            { reset time counter }
                            t_step_tps_ad      = 0
                    endif
                    t_step_tps_ad + = 5 ms
                    if      t_step_tps_ad > C_T_TPS_AD_HYS
                    then    { Calculation of lower mechanical stop and TPS amplification }
```

```
                    call function calculate_AD_values( )
                    { initialisation for the next adaptation step }
                    t_max_tps_ad      = C_T_MAX_TPS_SP_RST_CHK
                    { next adaptation step }
                    TPS_AD_STEP       = AD_GO_LIH
               endif
          else { reset time counter }
               t_step_tps_ad          = 0
               v_tps_x_buf            = V_TPS_X
               v_tps_gain_1_buf       = V_TPS_GAIN_1
          endif
     else { lower mechanical stop adaptation outside range }
          ERR_SYM_TPS_AD = SYM_3
          LV_ERR_TPS_AD  = 1                    {set for DC without debounce}
     endif
end function
```

A.15.5.3.1 Function calculate_AD_values( )

Description:

This function calculates the adaptation values of the lower mechanical stop and limp-home position for both throttle position sensors. The throttle position sensor slope TPS_SLOP is set to the nominal slope C_TPS_SLOP.

Application conditions:

Activation: *Is called from function ad_bol_pos( )*

Formula section:

calculate_AD_values( )

{ Nominal slope of TPS 1 and 2 }

$$TPS\_SLOP = C\_TPS\_SLOP$$

{ Lower mechanical stop of TPS 1 and 2 }

$$V\_TPS\_AD\_BOL\_x = \frac{v\_tps\_x\_buf + V\_TPS\_x}{2}$$

{ Lower mechanical stop of amplified TPS 1 channel }

$$V\_TPS\_GAIN\_AD\_BOL = \frac{v\_tps\_gain\_1\_buf + V\_TPS\_GAIN\_1}{2}$$

{ Calculation of the limp-home position for TPS 1, 2 and the default value }

$$TPS\_LIH\_1 = [V\_TPS\_AD\_LIH\_1 - V\_TPS\_AD\_BOL\_1]... \cdot TPS\_SLOP$$

$$TPS\_LIH\_2 = [V\_TPS\_AD\_BOL\_2 - V\_TPS\_AD\_LIH\_2]... \cdot TPS\_SLOP$$

$$TPS\_LIH\_INI = 0.5 \cdot [TPS\_LIH\_1 + TPS\_LIH\_2]$$

{ Set temporary electrical adaptation values }

```
V_TPS_AD_EL_BOL_1     = V_TPS_AD_BOL_1
V_TPS_AD_EL_BOL_2     = V_TPS_AD_BOL_2
V_TPS_GAIN_AD_EL_BOL  = V_TPS_GAIN_AD_BOL
```

end function

A.15.5.4 Function ad_go_lih(), throttle flap goes below LIH into position

Description:

This function is used to positioning the throttle flap into a position between lower mechanical stop and limp-home. The position controller is switched-on and the setpoint is increased by change limitation until the setpoint for testing the lower return spring is reached.

The throttle flap has to reach the requested position in a defined time limit and hysteresis window, otherwise an error will be indicated.

Formula section:

```
ad_go_lih ()
    if      TPS_SP_AD < C_TPS_SP_SPR_CHK_UP
    then    TPS_SP_AD + = C_TPS_SP_RST_CHK_LGRD
            if      TPS_SP_AD > C_TPS_SP_SPR_CHK_UP
            then    TPS_SP_AD = C_TPS_SP_SPR_CHK_UP
            endif
    else    if      t_max_tps_ad
            then    t_max_tps_ad - = 5 ms
                    if      | TPS_ETC – C_TPS_SP_SPR_CHK_UP | ≤ ...
                             C_TPS_SP_RST_CHK_HYS
                    then    { initialisation for the next adaptation step }
                            t_max_tps_ad            = C_T_MAX_TPS_LIH_CHK
                            t_step_tps_ad           = 0
                            { switch off request ETC power stage }
                            LV_TPS_AD_CUR_OFF       = 1
                            { next adaptation step }
                            TPS_AD_STEP             = AD_SPR_CHK_1
                    endif
            else    { Throttle position for the lower return spring check is not reached ! }
                    ERR_SYM_TPS_AD_SPR    = SYM_0
                    LV_ERR_TPS_AD_SPR     = 1              {set for DC without debounce}
            endif
    endif
end function
```

A.15.5.5 Function ad_spr_chk_1(), check of the upper return spring

Description:

This function checks the function of the upper return spring. The ETC power stage is switched-off and the throttle has to return by spring power in the limp-home position, otherwise a return spring error is present.

Formula section:

```
ad_spr_chk_1()
    if      t_max_tps_ad
    then    t_max_tps_ad - = 5 ms
            if      | V_TPS_X – V_TPS_AD_LIH_X | ≤ C_V_TPS_LIH_CHK_HYS
```

```
            then   t_step_tps_ad + = 5 ms
                   if     t_step_tps_ad > C_T_DYW_TPS_LIH_CHK
                   then   { initialisation for the next adaptation step }
                          t_max_tps_ad              = C_T_MAX_TPS_SP_RST_CHK
                          { Chose next adaptation step dependent on the TPS 1 amplifier }
                          if     LC_TPS_AD_GAIN_ENA == 1
                          then  TPS_AD_STEP      = AD_GO_TOL_1   { with adaptation }
                          else  TPS_AD_STEP      = AD_GO_TOL_2   { without adaptation }
                          endif
                          { set new ETC setpoint }
                          TPS_SP_AD                 = TPS_LIH_1
                          { switch on request ETC power stage }
                          LV_TPS_AD_CUR_OFF  = 0
                   endif
            else   { reset time counter }
                   t_step_tps_ad = 0
            endif
     else   { LIH position not reached, error in lower return spring check ! }
            ERR_SYM_TPS_AD_SPR = SYM_1
            LV_ERR_TPS_AD_SPR    = 1                        {set for DC without debounce}
     endif
end function
```

A.15.5.6 Function ad_go_tol_1(), throttle flap goes into upper position

Description:

After the adaptation of the lower mechanical stop and the limp-home position the adaptation routine goes on with the upper spring test and the adaptation of the amplification from the TPS measuring amplifier. This function is used to positioning the throttle flap into a position above limp-home. The position controller is switched-on and the setpoint is increased by change limitation until the setpoint for the adaptation of the measuring amplifier is reached.

The throttle flap has to reach the requested position in a defined time limit and hysteresis window, otherwise an error will be indicated.

Formula section:

```
ad_go_tol_1()
     if     TPS_SP_AD < C_TPS_SP_RST_CHK
     then   TPS_SP_AD += C_TPS_SP_RST_CHK_LGRD
            if     TPS_SP_AD > C_TPS_SP_RST_CHK
            then   TPS_SP_AD = C_TPS_SP_RST_CHK
            endif
     else   TPS_SP_AD = C_TPS_SP_RST_CHK
            if     t_max_tps_ad
            then   t_max_tps_ad - = 5 ms
                   if     | TPS_ETC – C_TPS_SP_RST_CHK | ≤ ...
                           C_TPS_SP_RST_CHK_HYS
                   then   { initialisation for the next adaptation step }
                          t_max_tps_ad         = C_T_MAX_TPS_AD
```

```
                    t_step_tps_ad        = 0
                    v_tps_1_buf          = V_TPS_1
                    v_tps_gain_1_buf     = V_TPS_GAIN_1
                    { next adaptation step }
                    TPS_AD_STEP          = AD_TOL_POS
               endif
          else { Throttle position for the upper return spring check is not reached ! }
               ERR_SYM_TPS_AD_SPR = SYM_2
               LV_ERR_TPS_AD_SPR    = 1           {set for DC without debounce}
          endif
     endif
end function
```

A.15.5.7 Function ad_tol_pos(), adaptation of the measuring amplifier

Description:

The following function adapts the real amplification of the measuring amplifier used for throttle position sensor 1. After the adaptation the throttle flap drives further in next position for the following upper return spring check. The ETC power stage is still enable.

Formula section:

```
ad_tol_pos()
     if   t_max_tps_ad
     then t_max_tps_ad - = 5 ms
          if   | TPS_ETC - C_TPS_SP_RST_CHK | ≤ ...
               C_TPS_SP_RST_CHK_HYS
          then if   | v_tps_1_buf – V_TPS_1 | > C_V_TPS_AD_HYS
               then v_tps_1_buf       = V_TPS_1
                    v_tps_gain_1_buf  = V_TPS_GAIN_1
                    t_step_tps_ad     = 0
               endif
               t_step_tps_ad + = 5 ms
               if   t_step_tps_ad > C_T_TPS_AD_HYS
               then { Calculate upper reference points ! }
```

$$V\_TPS\_AD\_TOL\_1 = \frac{v\_tps\_1\_buf + V\_TPS\_1}{2}$$

$$V\_TPS\_AD\_TOL\_GAIN\_1 = ... \frac{v\_tps\_gain\_1\_buf + V\_TPS\_GAIN\_1}{2}$$

{ Calculate amplification of the measuring amplifier for TPS 1 ! }

$$TPS\_AD\_SLOP\_GAIN\_1 = ... \frac{V\_TPS\_AD\_TOL\_GAIN\_1 - V\_TPS\_GAIN\_AD\_BOL}{V\_TPS\_AD\_TOL\_1 - V\_TPS\_AD\_BOL\_1}$$

```
                    if   TPS_AD_SLOP_GAIN_1 < C_TPS_AD_SLOP_GAIN_MIN
                    then { Error during adaptation, amplification too low ! }
                         ERR_SYM_TPS_AD_GAIN  = SYM_0
```

```
                    LV_ERR_TPS_AD_GAIN     = 1
                    { The inhibition condition for TPS_GAIN_1 isn't removed ! }
               else if      TPS_AD_SLOP_GAIN_1 > ...
                            C_TPS_AD_SLOP_GAIN_MAX
                    then { Error during adaptation, amplification too high ! }
                         ERR_SYM_TPS_AD_GAIN    = SYM_1
                         LV_ERR_TPS_AD_GAIN     = 1
                         { The inhibition condition for TPS_GAIN_1 isn't removed ! }
                    else { Slope of the amplified TPS 1 }
```

$$TPS\_GAIN\_SLOP = \frac{TPS\_SLOP}{TPS\_AD\_SLOP\_GAIN\_1}$$

```
                         { Remove inhibition condition for TPS_GAIN_1 ! }
                         LV_INH_TPS_GAIN_1 = 0
                    endif
               endif
               { initialisation for the next adaptation step }
               t_max_tps_ad     = C_T_MAX_TPS_SP_RST_CHK
               { next adaptation step }
               TPS_AD_STEP  = AD_GO_TOL_2
          endif
     else { reset time counter }
          t_step_tps_ad      = 0
          v_tps_1_buf        = V_TPS_1
          v_tps_gain_1_buf   = V_TPS_GAIN_1
     endif
else { Adaptation of the measuring amplifier not possible ! }
     ERR_SYM_TPS_AD  = SYM_2
     LV_ERR_TPS_AD   = 1                          {set for DC without debounce}
endif
end function
```

A.15.5.8 Function ad_go_tol_2(), throttle flap goes into upper position

Description:

The function is used to drive the throttle flap in a position above throttle LIH before the upper return spring can be checked. The position controller is still enable and the setpoint is increased using change limitation. The throttle flap has to reach the requested position in a defined time limit and hysteresis window, otherwise an error will be indicated.

Formula section:

ad_go_tol_2()

```
if      TPS_SP_AD < C_TPS_SP_RST_CHK_BAS
then    TPS_SP_AD += C_TPS_SP_RST_CHK_LGRD
        if      TPS_SP_AD > C_TPS_SP_RST_CHK_BAS
        then    TPS_SP_AD = C_TPS_SP_RST_CHK_BAS
        endif
else    TPS_SP_AD = C_TPS_SP_RST_CHK_BAS
        if      t_max_tps_ad
```

```
        then  t_max_tps_ad - = 5 ms
              if    | TPS_ETC – C_TPS_SP_RST_CHK_BAS | ≤ ...
                     C_TPS_SP_RST_CHK_HYS
              then  { initialisation for the next adaptation step }
                    t_max_tps_ad          = C_T_MAX_TPS_LIH_CHK
                    t_step_tps_ad         = 0
                    { switch-off request ETC power stage }
                    LV_TPS_AD_CUR_OFF     = 1
                    { next adaptation step }
                    TPS_AD_STEP           = AD_SPR_CHK_2
              endif
        else  { Throttle position for the upper return spring check not reached ! }
              ERR_SYM_TPS_AD_SPR = SYM_2
              LV_ERR_TPS_AD_SPR   = 1              {set for DC without debounce}
        endif
    endif
end function
```

A.15.5.9 Function ad_spr_chk_2(), check of the lower return spring

Description:

This function checks the function of the lower return spring. The ETC power stage is switched off and than the throttle has to return by spring power in the limp-home position, otherwise a return spring error is present.

Formula section:

```
ad_spr_chk_2()
    if    t_max_tps_ad
    then  t_max_tps_ad - = 5 ms
          if    | V_TPS_X – V_TPS_AD_LIH_X | ≤ C_V_TPS_LIH_CHK_HYS
          then  t_step_tps_ad + = 5 ms
                if    t_step_tps_ad > C_T_DYW_TPS_LIH_CHK
                then  { initialisation for the next adaptation step }
                      TPS_SP_AD  = TPS_LIH_1
                      { next adaptation step }
                      TPS_AD_STEP           = AD_END
                      { switch on request ETC power stage }
                      LV_TPS_AD_CUR_OFF     = 0
                      { enable injection for all cylinder }
                      INH_IV_TPS_AD         = 0
                endif
          else  { reset time counter }
                t_step_tps_ad = 0
          endif
    else  { LIH position not reached, error in upper return spring check ! }
          ERR_SYM_TPS_AD_SPR = SYM_3
          LV_ERR_TPS_AD_SPR   = 1              {set for DC without debounce}
    endif
```

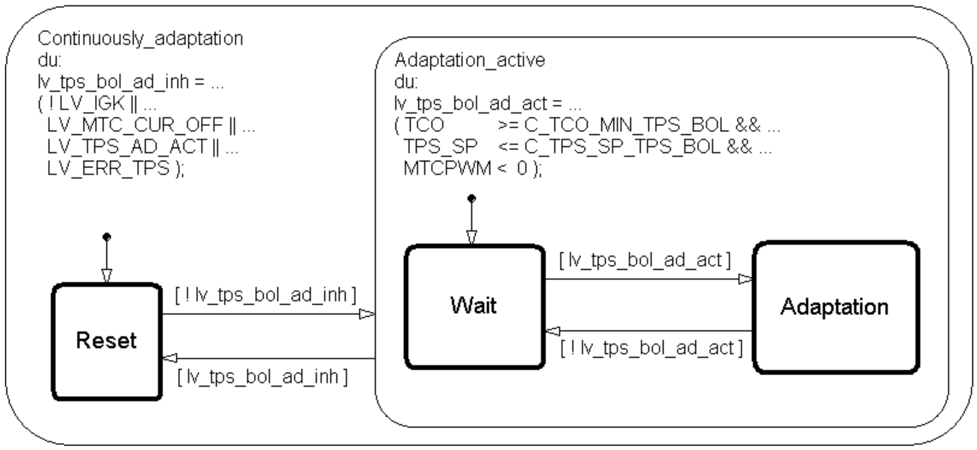
end function

A.15.6 Continuously BOL adaptation

Description:

The electrical values of the lower mechanical stop depend on throttle flap temperature. In some cases the throttle flap drives against the lower mechanical stop if a setpoint < 1 ° TPS is desired. To prevent this behaviour the electrical values of the lower mechanical stop are increased respectively decreased.

Signal flow diagram:



Continuously BOL adaptation

Application conditions:

Initialisation:	at reset **or** LV_IGK = 0 ⇒ 1 :
	STATE_TPS_BOL_AD = BOL_RESET
	V_TPS_AD_EL_BOL_x = V_TPS_AD_BOL_x
	V_TPS_GAIN_AD_EL_BOL = V_TPS_GAIN_AD_BOL
	V_TPS_BOL_OFS = T_TPS_BOL_OFS_INC = T_TPS_BOL_OFS_DEC = 0
Recurrence:	40 ms
Activation:	LV_IGK == 1
Deactivation:	When the activation condition isn’t fulfilled.

Formula section:

{ global inhibition condition for the continuously BOL adaptation }

lv_tps_bol_ad_inh = **(** LV_MTC_CUR_OFF **or**
LV_TPS_AD_ACT **or**
LV_ERR_TPS **)**

{ continuously BOL adaptation }

if lv_tps_bol_ad_inh

then *{ reset of the current adaptation values }*

STATE_TPS_BOL_AD = BOL_RESET
V_TPS_AD_EL_BOL_1 = V_TPS_AD_BOL_1
V_TPS_AD_EL_BOL_2 = V_TPS_AD_BOL_2
V_TPS_GAIN_AD_EL_BOL = V_TPS_GAIN_AD_BOL

V_TPS_BOL_OFS = 0
T_TPS_BOL_OFS_INC = 0
T_TPS_BOL_OFS_DEC = 0

else *{ activation condition for the continuously BOL adaptation }*

lv_tps_bol_ad_act = **(** TCO >= C_TCO_MIN_TPS_BOL **and**
TPS_SP <= C_TPS_SP_TPS_BOL **and**
MTCPWM < 0 **)**

if lv_tps_bol_ad_act

then **call function** bol_adaptation()

{ new calculation of the adaptation values }

V_TPS_AD_EL_BOL_1 = V_TPS_AD_BOL_1 + V_TPS_BOL_OFS
V_TPS_AD_EL_BOL_2 = V_TPS_AD_BOL_2 – V_TPS_BOL_OFS
V_TPS_GAIN_AD_EL_BOL = V_TPS_GAIN_AD_BOL ...
\+ V_TPS_BOL_OFS * TPS_AD_SLOP_GAIN_1

else *{ wait of activation }*

STATE_TPS_BOL_AD = BOL_WAIT
T_TPS_BOL_OFS_INC = 0
T_TPS_BOL_OFS_DEC = 0

endif

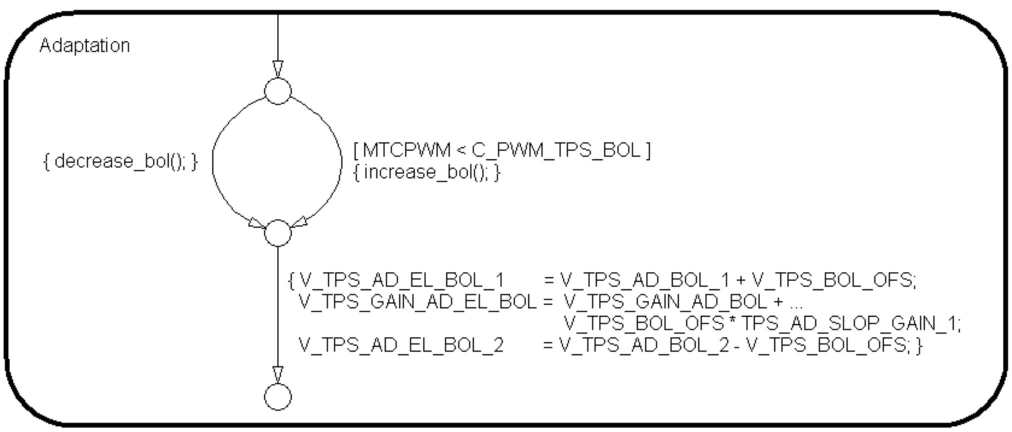
endif

A.15.6.1 Function bol_adaptation(), the continuously adaptation is started

Description:

This function calculates the temporary offset V_TPS_BOL_OFS for a increasing or decreasing of the electrical adaptation values (see also in the signal flow diagram). The new temporary adaptation values are not saved in the non-volatile memory.

Signal flow diagram:



Increase and decrease of the temporary offset

Formula section:

bol_adaptation()

```
        if      MTCPWM < C_PWM_TPS_BOL
        then    { increase temporary adaptation offset }
                STATE_TPS_BOL_AD = BOL_INC
                if      T_TPS_ BOL_OFS_INC >= C_T_TPS_ BOL_OFS_INC
                then    V_TPS_BOL_OFS += C_V_TPS_ BOL_OFS_COR
                        if      V_TPS_BOL_OFS >= C_V_TPS_BOL_OFS_MAX
                        then    V_TPS_BOL_OFS = C_V_TPS_BOL_OFS_MAX
                        endif
                else    T_TPS_BOL_OFS_INC += 1 h
                endif
                T_TPS_BOL_OFS_DEC = 0
        else    { decrease temporary adaptation offset }
                STATE_TPS_BOL_AD = BOL_DEC
                if      T_TPS_BOL_OFS_DEC >= C_T_TPS_BOL_OFS_DEC
                then    V_TPS_BOL_OFS – = C_V_TPS_BOL_OFS_COR
                        if      V_TPS_BOL_OFS <= 0
                        then    V_TPS_BOL_OFS = 0
                        endif
                else    T_TPS_BOL_OFS_DEC += 1 h
                endif
                if      T_TPS_BOL_OFS_INC != 0
                then    T_TPS_BOL_OFS_INC – = 1 h
                endif
        endif
end function
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PWM_TPS_BOL	1	8000... 7FFFH	-100...99.997	0.00305	%
PWM threshold for the selection between offset increasing and decreasing					
C_TCO_MIN_TPS_BOL	1	0...FEH	-48...142.5	0.75	°C
Minimum temperature threshold of the continuously BOL adaptation					
C_V_TPS_AD_BOL_INI_1	1	0...3FFH	0...4.9951	0.0049	V
Start adaptation value for the lower mechanical stop (TPS 1)					
C_V_TPS_AD_BOL_INI_2	1	0...3FFH	0... 4.9951	0.0049	V
Start adaptation value for the lower mechanical stop (TPS 2)					
C_V_TPS_GAIN_AD_BOL_INI	1	0...3FFH	0...4.9951	0.0049	V
Start adaptation value for the lower mechanical stop (amplified TPS 1)					
C_V_TPS_SP_BOL_1	1	0...3FFH	0...4.9951	0.0049	V
Lower limit for plausibility check of the adaptation value lower stop TPS 1					
C_V_TPS_SP_BOL_2	1	0...3FFH	0...4.9951	0.0049	V
Upper limit for plausibility check of the adaptation value lower stop TPS 2					
C_V_TPS_SP_LIH_1	1	0...3FFH	0...4.9951	0.0049	V
Set point of the limp home position, voltage value TPS 1					
C_V_TPS_SP_LIH_2	1	0...3FFH	0...4.9951	0.0049	V
Set point of the limp home position, voltage value TPS 2					
C_T_DLY_MIN_TPS_AD	1	0...FFH	0...1.275	0.005	s
Minimum delay time after control unit is switched on					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_TPS_AD	1	0...FFH	0...1.275	0.005	s
Delay time after control unit is switched on					
C_T_DYW_TPS_LIH_CHK	1	0...FFH	0...1.275	0.005	s
Time period for Dynamic window around Limp Home					
C_T_MAX_TPS_LIH_CHK	1	0...FFH	0...1.275	0.005	s
Maximum time for Limp Home recognition					
C_T_TPS_AD_HYS	1	0...FFH	0...1.275	0.005	s
Time period for hysteresis band to adaptation					
C_T_MAX_TPS_AD	1	0...FFH	0...1.275	0.005	s
Maximum time for adaptation of the LIH and BOL position					
C_T_MAX_TPS_SP_RST_CHK	1	0...FFH	0...1.275	0.005	s
Maximum time to reach the position C_TPS_SP_RST_CHK					
C_T_TPS_BOL_OFS_INC	1	0...FFFFH	0 ... 2621.4	0.04	s
Waiting period before offset incrementation, continuously BOL adaptation					
C_T_TPS_BOL_OFS_DEC	1	0...FFFFH	0 ... 2621.4	0.04	s
Waiting period before offset decrementation, continuously BOL adaptation					
C_V_TPS_AD_WIN_LIH	1	0...3FFH	0...4.9951	0.0049	V
Adaptation window size (+/-) for adaptation of the Limp Home position					
C_V_TPS_AD_HYS	1	0...3FFH	0...4.9951	0.0049	V
Hysteresis band for adaptation value determination					
C_V_TPS_AD_WIN_BOL	1	0...3FFH	0...4.9951	0.0049	V
Adaptation window size (+/-) for adaptation of lower stop					
C_V_TPS_LIH_CHK_HYS	1	0...3FFH	0...4.9951	0.0049	V
Admissible tolerance on checking the limp home position					
C_V_TPS_BOL_OFS_COR	1	0...FFFFH	0...0.3125	7.631e-5	V
Offset increment of one correction step in the continuously BOL adaptation					
C_V_TPS_BOL_OFS_MAX	1	0...FFFFH	0...0.3125	7.631e-5	V
Maximum adaptation offset of the continuously BOL adaptation					
C_TPS_AD_SLOP_GAIN_INI	1	0 ...FFFFH	0...7.9999	0.000122	-
Nominal value of the amplification slope of amplified TPS 1					
C_TPS_AD_SLOP_GAIN_MIN	1	0 ...FFFFH	0...7.9999	0.000122	-
Smallest possible amplification slope of amplified TPS 1					
C_TPS_AD_SLOP_GAIN_MAX	1	0 ...FFFFH	0...7.9999	0.000122	-
Greatest possible amplification slope of amplified TPS 1					
C_TPS_SP_RST_CHK	1	0...3FFFH	0...119.5	0.0073	°TPS
TPS_SP for testing the retracting spring					
C_TPS_SP_RST_CHK_BAS	1	0...3FFFH	0...119.5	0.0073	°TPS
TPS_SP for testing the retracting spring					
C_TPS_SP_SPR_CHK_UP	1	0...3FFFH	0...119.5	0.0073	°TPS
TPS_SP for GO_LIH position					
C_TPS_SP_RST_CHK_HYS	1	1...FFH	0.0073...1.86	0.0073	°TPS
Hysteresis of C_TPS_SP_RST_CHK					
C_TPS_SP_RST_CHK_LGRD	1	1...FFH	0.0073...1.86	0.0073	°TPS/5ms
Change limitation for stopping the position C_TPS_SP_RST_CHK					
C_TPS_SP_BOL_LGRD	1	1...FFH	0.0073...1.86	0.0073	°TPS/5ms
Change limitation for stopping the lower position					
C_TPS_SP_TPS_BOL	1	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint threshold of the continuously BOL adaptation					
C_TPS_LIH_INI	1	0...3FFFH	0...119.5	0.0073	°TPS
Initialisation value of the limp-home position in case of adaptation errors					
C_TPS_SLOP	1	0...FFFFH	0...47.9993	7.3e-4	°TPS/V
Signal slope of the throttle position sensor 1 and 2					
LC_TPS_AD_LIH_ENA	1	0...1H	0...1	1	-
Switch on of the continuously LIH adaptation					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TPS_AD_GAIN_ENA	1	0...1H	0...1	1	-
The logical constant allows to adapt the amplification of the measuring amplifier					

Application hint

C_TPS_SP_RST_CHK_BAS has to be set greater than C_TPS_SP_RST_CHK !

A.16 ETC actuator adaptation, diagnosis (appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TPS_AD_ACT	V/O	0...1H	0...1	1	-
Logical variable to indicate an active or passive TPS adaptation					
LV_TPS_AD_REQ	V/O/S	0...1H	0...1	1	-
Logical variable to request an TPS adaptation					
LV_INH_TPS_AD	V/O	0...1H	0...1	1	-
Logical variable to inhibit the TPS adaptation					
LV_TPS_AD_VB_LOW	V/O	0…1H	0…1	1	-
VB low during TPS adaptation					
LV_TPS_AD_REQ_VB_LOW	V/O	0…1H	0…1	1	-
Logical variable to request TPS adaptation at VB low					
T_DLY_TPS_AD_VB_LOW	V/O	0…FFFFH	0…327.68	0.005	s
Delay time for TPS adaptation error with low VB during adaptation					
LV_MTC_CUR_ERR	V	0...1H	0...1	1	-
Logical variable to indicate the error between actual and requested MTC current					

Input data:

LV_IGK	TIA	TCO	LV_TPS_AD_ACT_EXT_ADJ
N_32	VB	VS	LV_TPS_AD_REQ_ST_CHK
TPS_AD_STEP	LV_ERR_TPS_AD	LV_ERR_TPS_AD_SPR	LV_ERR_TPS_AD_GAIN
LV_ERR_TPS	LV_ERR_TPS_ST_CHK	LV_MTC_CUR_OFF	LC_TPS_AD_GAIN_ENA

FUNCTION DESCRIPTION:

A.16.1 Set adaptation state and request

Description:

The adaptation demand is checked every time the control unit is switched on. The TPS adaptation is necessary in the following cases:

- In initial start (first engine run or control unit exchange), that means no adaptation values are stored in the non volatile memory
- After a component exchange , if the TPS values are lie outside the tolerance window (LV_TPS_AD_REQ_ST_CHK)
- Check sum error (NVMY – error) is current
- Initiated by the tester (LV_TPS_AD_ACT_EXT_ADJ)
- Adaptation request bit LV_TPS_AD_REQ is set

Each of these points ends with a new adaptation request.

Application conditions:

Initialisation: at reset **or** LV_IGK = 0 ⇒ 1

LV_TPS_AD_ACT = 1 *{ initiate TPS adaptation or start check }*
LV_TPS_AD_REQ is initialised with the value from the non-volatile memory !
T_DLY_TPS_AD_VB_LOW = 0
LV_TPS_AD_VB_LOW = 0
LV_TPS_AD_REQ_VB_LOW = 0

```
if    ( Initial start      or        { first engine run or control unit exchange }
        Check sum error  )
then  { set a new adaptation request }
      LV_TPS_AD_REQ = 1
endif
```

Recurrence: 5 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition isn't fulfilled.

Formula section:

```
{ The TPS adaptation or start check are finished or stopped ! }
if    LV_TPS_AD_ACT
then  { TPS adaptation or start check successful finished }
      if    ( TPS_AD_STEP == ST_CHK_END     or
              TPS_AD_STEP == AD_END           )
      then  { Reset of the current adaptation or start check }
            LV_TPS_AD_ACT   = 0
            LV_TPS_AD_REQ_VB_LOW = 0
            if     LV_ERR_TPS_AD_GAIN == 0
            then   { Reset of the adaptation request }
                   LV_TPS_AD_REQ  = 0
            endif
      endif
      { TPS adaptation or start check is stopped in case of errors ! }
      if    ( LV_ERR_TPS_AD == 1            or
              LV_ERR_TPS_AD_SPR == 1         )
      then  { Stop of the current adaptation or start check }
            LV_TPS_AD_ACT   = 0
            LV_TPS_AD_REQ_VB_LOW = 0
            { Set a new adaptation request, the request is
              saved in the non-volatile memory ! }
            LV_TPS_AD_REQ  = 1
      endif
```

```
        {set a new adaptation request if VB low, e.g. during cranking}
        if      LV_TPS_AD_REQ == 1      and
                C_VB_MIN_TPS_AD < VB < C_VB_TPS_AD_VB_LOW
        then    if      T_DLY_TPS_AD_VB_LOW < C_T_TPS_AD_VB_LOW
                then    LV_TPS_AD_REQ_VB_LOW  = 1
                        start timer T_DLY_TPS_AD_VB_LOW
                        {with this adaptation request, TPS adaptation should start at once, no
                        engine start permitted in between}
                else    LV_TPS_AD_VB_LOW = 1
                        LV_TPS_AD_REQ_VB_LOW  = 0
                endif
        else    LV_TPS_AD_REQ_VB_LOW = 0
        endif

        { TPS start check is stopped in case of errors ! }
        if      LV_ERR_TPS_ST_CHK == 1
        then    { Stop of the current start check }
                LV_TPS_AD_ACT   = 0
                { TPS adaptation is not requested in case of start check error ! }
                LV_TPS_AD_REQ  = 0
        endif
endif
{ Set a new adaptation in the actual DC via garage tester }
if      LV_TPS_AD_ACT_EXT_ADJ = 0 ⇒ 1   { request via garage tester }
then    { set a new TPS adaptation at once }
        LV_TPS_AD_ACT   = 1
        LV_TPS_AD_REQ  = 1
endif
{ Set a new adaptation request }
if      ( LC_TPS_AD_REQ = 0 ⇒ 1             or      { request via calibration system }
          LC_TPS_AD_GAIN_ENA = 0 ⇒ 1        or      { request via system configuration }
          LV_TPS_AD_REQ_ST_CHK               )      { request from TPS start check }
then    { Set a new adaptation request, at the end of the DC the adaptation request is saved in the
          non-volatile memory ! }
        LV_TPS_AD_REQ = 1
endif
{ If the TPS start check requests a new TPS adaptation, the TPS adaptation is started at once ( the
activation information LV_TPS_AD_ACT == 1 is still available ) }
```

A.16.2 Inhibition of the TPS adaptation

Description:

After the activation of the adaptation routine the following conditions are requested until LV_TPS_AD_ACT == 0. The inhibition information are used for the TPS adaptation and the TPS start check.

Application conditions:

Initialisation: at reset **or** LV_IGK = 0 ⇒ 1 **or** LV_TPS_AD_ACT = 0 ⇒ 1:
LV_INH_TPS_AD = 0
LV_MTC_CUR_ERR = 0
ctr_mtc_cur_err = 0

Recurrence: 5 ms

Activation: LV_TPS_AD_ACT == 1

Deactivation: When the activation condition isn't fulfilled.

Formula section:

{ Comparison between requested and actual ETC power stage state (switch-on or -off) ! In the constant C_MTC_CUR_TPS_AD the power stage state for every adaptation step is coded. }

```
if      ( 0x01 h & [ C_MTC_CUR_TPS_AD >> TPS_AD_STEP ] ) == LV_MTC_CUR_OFF
then    LV_MTC_CUR_ERR = 0
        ctr_mtc_cur_err = 0
else    { The error between requested and actual ETC power stage state is only debounced ! }
        if      ctr_mtc_cur_err < C_CTR_MTC_CUR_ERR
        then    ctr_mtc_cur_err + = 1 h
        else    LV_MTC_CUR_ERR = 1
        endif
endif
{ calculate inhibit information for the TPS adaptation }
LV_INH_TPS_AD   = (   LV_IGK == 0                               or
                      VB    < C_VB_MIN_TPS_AD                   or
                      LV_TPS_AD_VB_LOW = 1                      or
                      N_32 > C_N_MAX_TPS_AD                     or
                      VS    > NC_VS_MAX_TPS_AD ( 2 km/h )       or
                      TIA   <= C_TIA_MIN_TPS_AD                 or
                      TCO   <= C_TCO_MIN_TPS_AD                 or
                      TCO   >= C_TCO_MAX_TPS_AD                 or
                      LV_ERR_TPS                                or
                      LV_MTC_CUR_ERR                             )
```

A.16.3 Error symptom definition

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
TPS adaption TPS_AD	Adaption conditions exceeded	SYM_0	*No*
	Voltage value at LIH- adaption outside range	SYM_1	
	Spring test and CHK_LIH not passed	SYM_2	
	Adaption values – lower mechanical stop - outside	SYM_3	

List of Environmental Data to store in Failure Memory: TPS_AD_STEP
MTCPWM_ENVD
TPS_LIH_ENVD
TPS_AV_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
TPS adaption, spring check TPS_AD_SPR	Lower position not reached	SYM_0	*No*
	Error lower return spring check	SYM_1	
	Upper position not reached	SYM_2	
	Error upper return spring check	SYM_3	

List of Environmental Data to store in Failure Memory: TPS_AD_STEP
MTCPWM_ENVD
TPS_LIH_ENVD
TPS_AV_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
TPS adaption, amplifier check TPS_AD_GAIN	adapted amplification too low	SYM_0	*No*
	adapted amplification too high	SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
TPS start check TPS_ST_CHK	Error in spring check	SYM_0	*No*
	Error in limp-home check	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TPS_AD_STEP
MTCPWM_ENVD
TPS_LIH_ENVD
TPS_AV_ENVD

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VB_MIN_TPS_AD	1	0...FFH	0...26	0.102	V
VB threshold for ETC adaptation					
C_VB_TPS_AD_VB_LOW	1	0…FFH	0…26	0.102	V
VB threshold for VB low adaptation					
C_N_MAX_TPS_AD	1	0...FFH	0...8160	32	rpm
Maximum engine speed for adaptation					
C_TIA_MIN_TPS_AD	1	0...FEH	-48...142.5	0.75	°C
TIA threshold for TPS adaptation					
C_TCO_MIN_TPS_AD	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO threshold for TPS adaptation					
C_TCO_MAX_TPS_AD	1	0...FEH	-48...142.5	0.75	°C
Maximum TCO threshold for TPS adaptation					
C_T_TPS_AD_VB_LOW	1	0…FFFFH	0…327.68	0.005	s
Possible time to perform TPS adaptation at VB low peak					
C_MTC_CUR_TPS_AD	1	0...FFFFH	0...65535	1	-
Coded information about the ETC power stage state during the TPS adaptation steps					
C_CTR_MTC_CUR_ERR	1	0...FFH	0...255	1	-
Debounce counter for the error between actual and requested MTC current					
LC_TPS_AD_REQ	1	0...1H	0...1	1	-
Logical variable to set a new adaptation request					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_VS_MAX_TPS_AD	1	0...FFH	0...255	1	km/h
Maximum vehicle speed treshold for adaptation, NC_VS_MAX_TPS_AD is set to 2 km/h					

A.17 ETC limp-home management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TPS	O/V	0...1h	0...1	1	-
Global error bit, is set, if a ETC fault occurs					
LV_N_LIM_ETC_LIH	O/V	0...1h	0...1	1	-
Engine speed limitation request					
LV_MTC_CUR_OFF_REQ	O/V	0...1h	0...1	1	-
ETC power stage deactivation request					
LV_TPS_SP_LIH_REQ	O/V	0...1h	0...1	1	-
ETC limp-home position setpoint request					
LV_TPS_TQ_RED_REQ	O/V	0...1h	0...1	1	-
TPS engine torque reduction request					
STATE_N_LIM_ETC_REQ	O/V	0h 1h 2h 4h 8h	N_NOT_LIM N_LIM_CON N_LIM_PVS not used not used	1	-
State variable for engine speed limitation request					
STATE_TPS_DIAG	O/V	0h 1h 2h 4h 8h	TPS_NO_ERROR TPS_1_ERROR TPS_2_ERROR TPS_MAX_ACT TPS_DBL_ERROR	1	-
State variable of the TPS diagnosis					
T_TPS_LIH_REQ	V	0...FFh	0...2,55	0,01	s
Time counter for throttle flap limp-home request					
TQI_MAX_ETC	O/V	8000...7FFFh	-1024...1023.97	0,03125	Nm
Engine torque limitation requests by the TPS state manager					
LV_ERR_LIH_POW_LIM	O/V/S	0...1h	0...1	1	-
Error Flag for Power Limitation Limp Home Reaction					
ERR_SYM_LIH_POW_LIM	O/V	0h 1h	NO_SYM SYM_0	1	-
Error Symptom for Power Limitation Limp Home Reaction					
LV_END_DIAG_LIH_POW_LIM	O/V	0...1h	0...1	1	-
End of Diagnosis for Power Limitation Limp Home Reaction					
LV_CDN_DIAG_LIH_POW_LIM	O/V	0...1h	0...1	1	-
Diagnosis Condition for Power Limitation Limp Home Reaction					
LV_ERR_LIH_N_LIM	O/V/S	0...1h	0...1	1	-
Error Flag for Engine Speed Limitation Limp Home Reaction					
ERR_SYM_LIH_N_LIM	O/V	0h 1h	NO_SYM SYM_0	1	-
Error Symptom for Engine Speed Limitation Limp Home Reaction					
LV_END_DIAG_LIH_N_LIM	O/V	0...1h	0...1	1	-
End of Diagnosis for Engine Speed Limitation Limp Home Reaction					
LV_CDN_DIAG_LIH_N_LIM	O/V	0...1h	0...1	1	-
Diagnosis Condition for Engine Speed Limitation Limp Home Reaction					
LV_ERR_LIH_ENG_OFF	O/V/S	0...1h	0...1	1	-
Error Flag for Engine Shut-Down Limp Home Reaction					
ERR_SYM_LIH_ENG_OFF	O/V	0h 1h	NO_SYM SYM_0	1	-
Error Symptom for Engine Shut-Down Limp Home Reaction					
LV_END_DIAG_LIH_ENG_OFF	O/V	0...1h	0...1	1	-
End of Diagnosis for Engine Shut-Down Limp Home Reaction					
LV_CDN_DIAG_LIH_ENG_OFF	O/V	0...1h	0...1	1	-
Diagnosis Condition for Engine Shut-Down Limp Home Reaction					
LV_ERR_LIH_IS	O/V/S	0...1h	0...1	1	-

Error Flag for Forced Idle Limp Home Reaction					
ERR_SYM_LIH_IS	O/V	0h 1h	NO_SYM SYM_0	1	-
Error Symptom for Forced Idle Limp Home Reaction					
LV_END_DIAG_LIH_IS	O/V	0...1h	0...1	1	-
End of Diagnosis for Forced Idle Limp Home Reaction					
LV_CDN_DIAG_LIH_IS	O/V	0...1h	0...1	1	-
Diagnosis Condition for Forced Idle Limp Home Reaction					

Input data:

LV_IGK	LV_ERR_TPS_VCC	LV_ERR_TPS_1	LV_ERR_TPS_2
LV_ERR_TPS_RATIO	LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2	LV_ERR_MTC_DR
LV_ERR_MTC_CTL	LV_ERR_TPS_AD	LV_ERR_TPS_AD_SPR	LV_ERR_TPS_ST_CHK
LV_ERR_PVS	TPS_LIH	TPS_AV	N_32
TQ_MAX_CLU	TQ_LOSS_REQ_CLU	LV_N_LIM_REQ_RST_CHK	LV_ERR_PVS_L_1
LV_ERR_PVS_H_1	LV_ERR_PVS_L_2	LV_ERR_PVS_H_2	LV_ERR_PVS_RATIO
LV_ERR_TQI_MON	LV_ERR_VCC_PVS_1	LV_ERR_VCC_PVS_2	LV_ERR_MON
LV_ERR_MON_3	LV_ERR_TPS_PLAUS	LV_ERR_N_MAX_MON	TECU_MAX
CPU_LOAD_MAX_ENVD_L	CPU_LOAD_MAX_ENVD_H	LV_ERR_PVS_SNG	LV_ERR_PVS_MPL
ERR_INTM_DIAG_INST_ACT			

FUNCTION DESCRIPTION:

General information:

The ETC management observes and coordinates all functions of the electrical throttle flap, these are the following functions: throttle position determination, throttle position control, actuator adaptation and the corresponding diagnosis.

Furthermore it checks the presence of ETC or PVS related errors which cause a limp home reaction opf the system such as Power Limitation, Engine Speed Limitation, Forced Idle or Engine Shut-Down.

If such a limp home is activated then an additional limp home error is entered to inform the service technician about the type of limp home reaction.

The module is directly calculated after the following diagnosis:

- Throttle position sensor diagnosis
- ETC power stage diagnosis
- ETC position controller diagnosis

A.17.1 TPS state manager

General information:

The TPS state manager coordinates all possible TPS errors (see below), indicates the state of the TPS diagnosis for the determination of the throttle position and requests a engine torque reduction in case of TPS errors. The module is calculated after the TPS diagnosis and indicates in STATE_TPS_DIAG the following states:

STATE_TPS_DIAG	Note	Fault reaction
TPS_NO_ERROR	no faults in sensor system	sensor 1 is active, no fault reaction
TPS_1_ERROR	sensor 1 is faulty	sensor 2 is active, engine torque reduction
TPS_2_ERROR	sensor 2 is faulty	sensor 1 is active, engine torque reduction
TPS_MAX_ACT	compare error	max. selection between sensor 1 & 2
TPS_DBL_ERROR	both sensors are faulty	limp-home request, ETC currentless

Possible TPS errors:

TPS errors	Note
LV_ERR_TPS_VCC	sensor supply voltage
LV_ERR_TPS_1	electrical diagnosis, range check, sensor 1
LV_ERR_TPS_2	electrical diagnosis, range check, sensor 2
LV_ERR_TPS_RATIO	compare error between TPS 1 and TPS 2
LV_ERR_TPS_MAF_1	plausibility error between TPS 1 and MAF
LV_ERR_TPS_MAF_2	plausibility error between TPS 2 and MAF

Signal flow diagram:

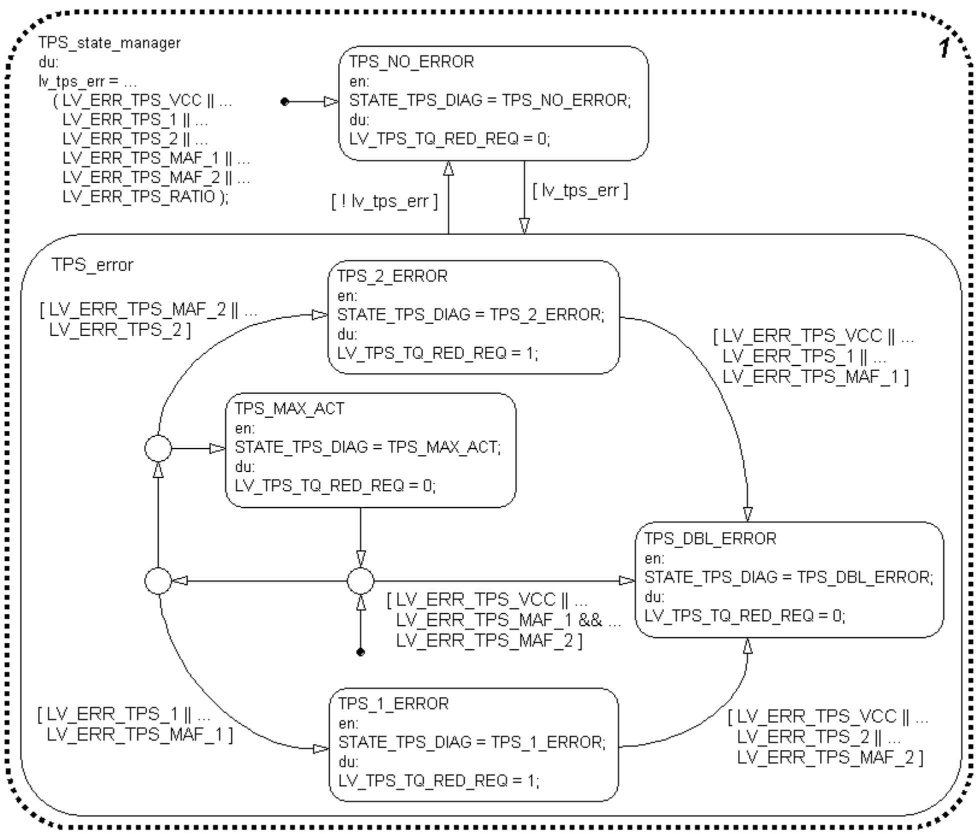


Figure 1: TPS state manager

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset
STATE_TPS_DIAG = TPS_NO_ERROR
LV_TPS_TQ_RED_REQ = 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

```
{ global transition between no TPS error and TPS error }
if   ( LV_ERR_TPS_VCC     or  LV_ERR_TPS_1       or
       LV_ERR_TPS_2       or  LV_ERR_TPS_MAF_1   or
       LV_ERR_TPS_MAF_2   or  LV_ERR_TPS_RATIO )
then lv_tps_err = 1
else lv_tps_err = 0
     STATE_TPS_DIAG = TPS_NO_ERROR
endif

{ TPS state manager }
switch( STATE_TPS_DIAG )
   case    TPS_NO_ERROR:             { no faults in sensor system, new faults are classified }
           LV_TPS_TQ_RED_REQ = 0     { engine torque reduction isn't requested }
           if      lv_tps_err
           then    { default transition, see also in signal flow diagramm }
                   if      LV_ERR_TPS_VCC       or
                           ( LV_ERR_TPS_MAF_1 and LV_ERR_TPS_MAF_2 )
                   then    STATE_TPS_DIAG = TPS_DBL_ERROR
                   else    if      LV_ERR_TPS_1       or
                                   LV_ERR_TPS_MAF_1
                                   then    STATE_TPS_DIAG = TPS_1_ERROR
                                   else    if      LV_ERR_TPS_2       or
                                           LV_ERR_TPS_MAF_2
                                   then     STATE_TPS_DIAG = TPS_2_ERROR
                                   else     STATE_TPS_DIAG = TPS_MAX_ACT
                                   endif
                                   endif
                   endif
           endif
           break
   case    TPS_1_ERROR:              { throttle position valve sensor 1 is faulty }
           LV_TPS_TQ_RED_REQ = 1     { engine torque reduction is requested }
           { transition to state TPS_DBL_ERROR, see also in signal flow diagramm }
           if      LV_ERR_TPS_VCC       or
```

```
                LV_ERR_TPS_2           or
                LV_ERR_TPS_MAF_2
          then  STATE_TPS_DIAG = TPS_DBL_ERROR
          endif
          break
    case  TPS_2_ERROR:                 { throttle position valve sensor 2 is faulty }
          LV_TPS_TQ_RED_REQ = 1        { engine torque reduction is requested }
          { transition to state TPS_DBL_ERROR, see also in signal flow diagramm }
          if    LV_ERR_TPS_VCC         or
                LV_ERR_TPS_1           or
                LV_ERR_TPS_MAF_1
          then  STATE_TPS_DIAG = TPS_DBL_ERROR
          endif
          break
    case  TPS_MAX_ACT:                 { compare error between TPS 1 and 2 }
          LV_TPS_TQ_RED_REQ = 0        { engine torque reduction isn't requested }
          { transition to the error states, see also in signal flow diagramm }
          if    LV_ERR_TPS_VCC         or
                ( LV_ERR_TPS_MAF_1 and LV_ERR_TPS_MAF_2 )
          then  STATE_TPS_DIAG = TPS_DBL_ERROR
          else  if    LV_ERR_TPS_1       or
                      LV_ERR_TPS_MAF_1
                then  STATE_TPS_DIAG = TPS_1_ERROR
                else  if    LV_ERR_TPS_2       or
                            LV_ERR_TPS_MAF_2
                      then  STATE_TPS_DIAG = TPS_2_ERROR
                      else  STATE_TPS_DIAG = TPS_MAX_ACT
                      endif
                endif
          endif
          break
    case  TPS_DBL_ERROR:               { both throttle valve position sensors are faulty }
          LV_TPS_TQ_RED_REQ = 0        { The standard engine torque reduction is requested,
          in the case of double TPS error the engine has to run in sefaty fuel shut-off, the power
          stage is disabled and engine speed limitation is statrted (see below, ETC LIH manager).}
          break
    default
  endswitch
  { The TPS state manager requests a engine torque limitation ! }
  if    LV_TPS_TQ_RED_REQ  and  LC_TQI_MAX_ETC_ENA
  then  TQI_MAX_ETC =  TQ_MAX_CLU * IP_FAC_TPS_TQ_RED( N_32 ) ...
                       – TQ_LOSS_REQ_CLU
  else  TQI_MAX_ETC = 7FFFh  { maximum value 1024 Nm }
  endif
```

A.17.2 General information:

The ETC limp-home manager observes all TPS and ETC diagnosis functions. In case of sefaty critical errors, the ETC limp-home manager switchs off the ETC power stage via disable line or requests a ETC limp-home setpoint for control limp-home position (in case of broken upper return spring, the power stage is enable) and demands a engine speed limitation. The manager distinguishs two modes of engine speed limitation :

Mode	STATE_N_LIM_ETC_REQ	Note
0	N_NOT_LIM	engnie speed limitation isn't demanded
1	N_LIM_CON	a constant engine speed limitation is demanded
2	N_LIM_PVS	a variable engine speed limitation is demanded, the limitation speed is a function of PVS

{ In case of engine speed limitation, the logical variable LV_N_LIM_ETC_LIH is always set ! }

Signal flow diagram:

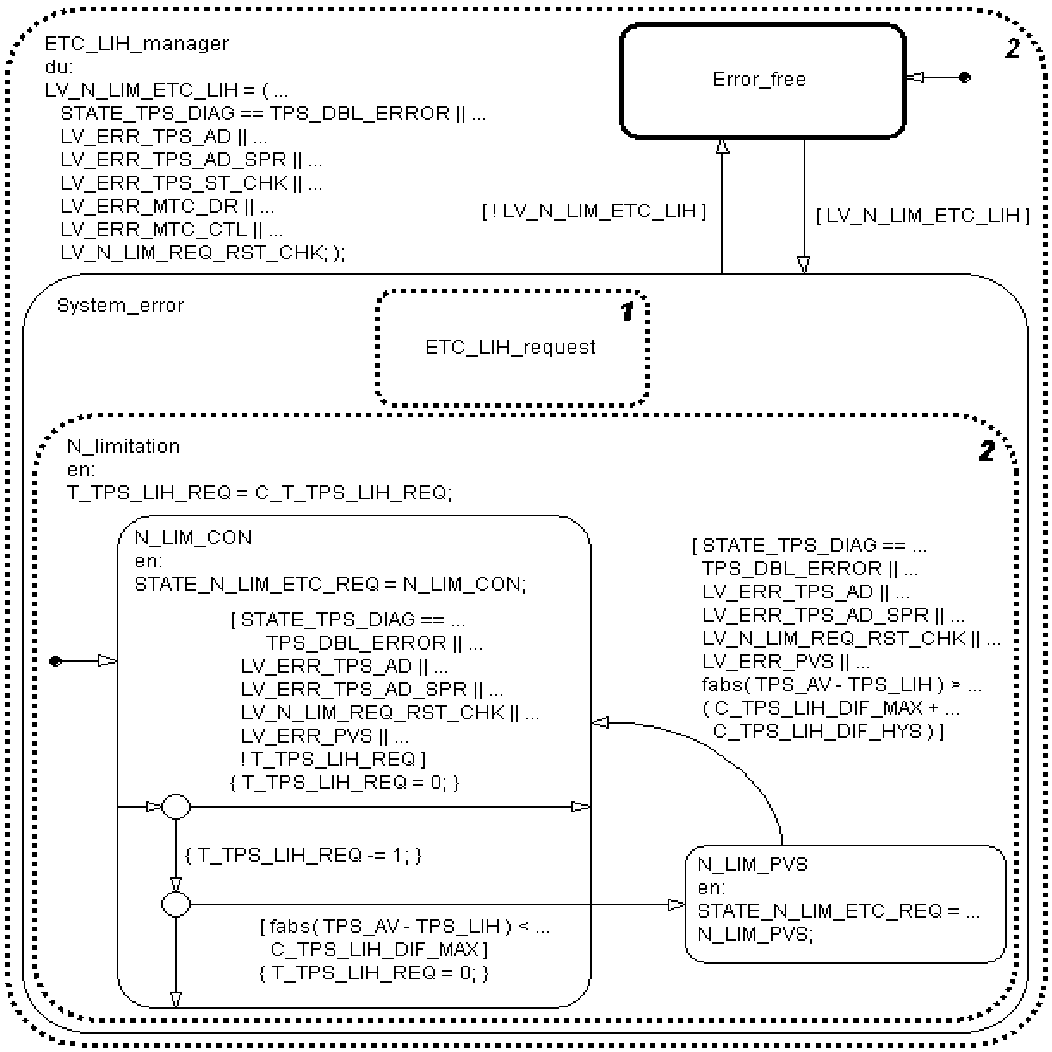


Figure 2: ETC limp-home manager

Application conditions:

Initialisation: at LV_IGK = 0 → 1 **or** reset :

```
LV_N_LIM_ETC_LIH         = 0
LV_MTC_CUR_OFF_REQ       = 0
LV_TPS_SP_LIH_REQ        = 0
STATE_N_LIM_ETC_REQ      = N_NOT_LIM
```

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

```
{ global transition between error free and system error }
if      ( STATE_TPS_DIAG == TPS_DBL_ERROR    or    { TPS double error }
          LV_ERR_MTC_DR                      or    { power stage error }
          LV_ERR_MTC_CTL                     or    { position controller error }
          LV_ERR_TPS_AD                      or    { TPS adaptation error }
          LV_ERR_TPS_AD_SPR                  or    { TPS adaptation error }
          LV_ERR_TPS_ST_CHK                  or    { TPS start check error }
          LV_N_LIM_REQ_RST_CHK )                   {Request after ECU request}
then      { engine speed limitation request }
          LV_N_LIM_ETC_LIH         = 1
else      { reset of the engine speed limitation request }
          LV_N_LIM_ETC_LIH         = 0
          STATE_N_LIM_ETC_REQ      = N_NOT_LIM
endif
{ calculation of power stage disable or ETC LIH setpoint request }
if      LV_N_LIM_ETC_LIH
then    if      STATE_TPS_DIAG == TPS_DBL_ERROR          or
                LV_ERR_TPS_AD                            or
                LV_ERR_TPS_AD_SPR                        or
                LV_ERR_MTC_DR                            or
                LV_ERR_MTC_CTL                           or
                LV_N_LIM_REQ_RST_CHK
        then    { power stage disable and no LIH setpoint are requested }
                LV_MTC_CUR_OFF_REQ   = 1
                LV_TPS_SP_LIH_REQ    = 0
        else    { power stage enable and LIH setpoint are requested }
                LV_MTC_CUR_OFF_REQ   = 0
                LV_TPS_SP_LIH_REQ    = 1
        endif
else    { ETC system is error free, power stage enable, standard ETC setpoint }
        LV_MTC_CUR_OFF_REQ   = 0
        LV_TPS_SP_LIH_REQ    = 0
```

```
endif

{ Engine speed limitaion request }
switch( STATE_N_LIM_ETC_REQ )
   case   N_NOT_LIM: { engine speed limitation isn't demanded }
          { default transition, see also in signal flow diagramm }
          if     LV_N_LIM_ETC_LIH
          then   { set new engine speed limitation state }
                 STATE_N_LIM_ETC_REQ = N_LIM_CON
                 T_TPS_LIH_REQ       = C_T_TPS_LIH_REQ
          endif
          break
   case   N_LIM_CON: { constant engine speed limitation }
          if     STATE_TPS_DIAG == TPS_DBL_ERROR    or
                 LV_ERR_TPS_AD                      or
                 LV_ERR_TPS_AD_SPR                  or
                 LV_ERR_PVS                         or
                 T_TPS_LIH_REQ == 0                 or
                 LV_N_LIM_REQ_RST_CHK
          then   { a variable engine speed limitation is not available too }
                 T_TPS_LIH_REQ = 0
          else   T_TPS_LIH_REQ – = 1 H
                 if     | TPS_AV – TPS_LIH | < C_TPS_LIH_DIF_MAX
                 then   { a variable engine speed limitation is requested }
                        STATE_N_LIM_ETC_REQ = N_LIM_PVS
                        T_TPS_LIH_REQ       = 0
                 endif
          endif
          break
   case   N_LIM_PVS: { variable engine speed limitation, the limitation speed is a function of PVS }
          if     STATE_TPS_DIAG == TPS_DBL_ERROR    or
                 LV_ERR_TPS_AD                      or
                 LV_ERR_TPS_AD_SPR                  or
                 LV_N_LIM_REQ_RST_CHK               or
                 LV_ERR_PVS                         or
                 | TPS_AV – TPS_LIH | > ...
                 ( C_TPS_LIH_DIF_MAX + C_TPS_LIH_DIF_HYS )
          then   { a constant engine speed limitation is requested }
                 STATE_N_LIM_ETC_REQ = N_LIM_CON
          endif
          break
   default
endswitch
```

A.17.3 Global error bit of the ETC system

General information:

The local variable LV_ERR_TPS is defined as global error bit and is provided as input variable for other functions. She isn't subject to the error management and is set, if a ETC fault occurs.

Application conditions:

Initialisation: LV_ERR_TPS = 0 at LV_IGK = 0 → 1 **or** reset

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

```
if    LV_ERR_TPS_VCC          or   { error TPS supply voltage }
      LV_ERR_TPS_1            or   { error TPS 1, electrical diagnosis }
      LV_ERR_TPS_2            or   { error TPS 2, electrical diagnosis }
      LV_ERR_TPS_MAF_1        or   { TPS 1 isn't plausible to MAF }
      LV_ERR_TPS_MAF_2        or   { TPS 2 isn't plausible to MAF }
      LV_ERR_TPS_AD           or   { error TPS adaptation, LIH + BOL }
      LV_ERR_TPS_AD_SPR       or   { error TPS adaptation, spring check }
      LV_ERR_TPS_ST_CHK       or   { error TPS start check }
      LV_ERR_MTC_DR           or   { error in power stage }
      LV_ERR_MTC_CTL               or   { error position control }
      LV_N_LIM_REQ_RST_CHK              {Request after ECU reset}
then  LV_ERR_TPS = 1
else  LV_ERR_TPS = 0
endif
```

A.17.4 Additional Information about Limp Home Reaction

Application conditions:

Initialisation: *at LV_IGK = 0 → 1* ***or*** *reset*
set all error variables to zero
(done by the error-management due to filter-type MEM)

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled.

Formula section:

Engine Power Limitation:

If *(a TPS related error is present which causes Power Limitation)*
LV_ERR_TPS_1 = 1
or LV_ERR_TPS_2 = 1
or LV_ERR_TPS_PLAUS = 1
or LV_ERR_TPS_MAF_1 = 1
or LV_ERR_TPS_MAF_2 = 1
then *(additional error detection of Power Limitation for limp home display)*
If STATE_TPS_DIAG == TPS_DBL_ERROR
then ERR_SYM_LIH_POW_LIM = NO_SYM
and LV_ERR_LIH_POW_LIM = 0
else ERR_SYM_LIH_POW_LIM = SYM_0
and LV_ERR_LIH_POW_LIM = 1
else *(healing of additional limp home error)*
ERR_SYM_LIH_POW_LIM = NO_SYM
and LV_ERR_LIH_POW_LIM = 0

Engine Speed Limitation:

If *(a PVS or TPS related error is present which causes Engine Speed Limitation)*
STATE_TPS_DIAG == TPS_DBL_ERROR
or LV_ERR_PVS_SNG = 1
or LV_ERR_MON = 1
or LV_ERR_TPS_VCC = 1
or LV_ERR_TPS_AD = 1
or LV_ERR_TPS_AD_SPR = 1
or LV_ERR_TPS_ST_CHK = 1

or LV_ERR_MTC_DR = 1

or LV_ERR_MTC_CTL = 1

or LV_ERR_N_MAX_MON = 1

or LV_ERR_TQI_MON = 1

then *(additional error detection of Engine Speed Limitation for limp home display)*

ERR_SYM_LIH_N_LIM = SYM_0

and LV_ERR_LIH_N_LIM = 1

else *(healing of additional limp home error)*

ERR_SYM_LIH_N_LIM = NO_SYM

and LV_ERR_LIH_N_LIM = 0

Engine Shut-Down:

If *(a related error is present which causes Engine Shut-Down)*

LV_ERR_MON_3 = 1

then *(additional error detection of Engine Shut-Down for limp home display)*

ERR_SYM_LIH_ENG_OFF = SYM_0

and LV_ERR_LIH_ENG_OFF = 1

else *(healing of additional limp home error)*

ERR_SYM_LIH_ENG_OFF = NO_SYM

and LV_ERR_LIH_ENG_OFF = 0

Forced Idle:

If *(a related error is present which causes forced Idle)*

LV_ERR_PVS_MPL = 1

then *(additional error detection of forced Idle for limp home display)*

ERR_SYM_LIH_IS = SYM_0

and LV_ERR_LIH_IS = 1

else *(healing of additional limp home error)*

ERR_SYM_LIH_IS = NO_SYM

and LV_ERR_LIH_IS = 0

A.17.5 Configuration for diagnostic symptoms

Diagnostic	Symptom Description	Symptom	Filter type
LIH_POW_LIM	Power Limitation Limp Home active	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LIH_N_LIM	Engine Speed Limitation Limp Home active	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LIH_ENG_OFF	Engine Shut-Down Limp Home active	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LIH_IS	Forced Idle Limp Home active	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: ERR_INTM_DIAG_INST_ACT
CPU_LOAD_MAX_ENVD_H
CPU_LOAD_MAX_ENVD_L
TECU_MAX

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_TPS_LIH_REQ	1	0...FFh	0...2,55	0,01	s
Maximum time for throttle flap limp-home request					
C_TPS_LIH_DIF_MAX	1	0...3FFFh	0...119,5	0,0073	°TPS
Maximum difference between TPS_AV and TPS_LIH					
C_TPS_LIH_DIF_HYS	1	0...FFh	0...1,86	0,0073	°TPS
Hysteresis of the maximum difference between TPS_AV and TPS_LIH					
LC_TQI_MAX_ETC_ENA	1	0...1h	0...1	1	-
Activation of the engine torque limitation in the case of TPS error					
IP_FAC_TPS_TQ_RED	8	0h...FFh	0 ... 1.275	0.005	-
LDP_N_32_ FAC_TPS_TQ_RED	8	0h...FFh	0 ... 8160	32	rpm
Factor engine torque limitiation					

A.18 Engine Position and Speed Diagnosis Manager

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CAM_TOT	V/O	0...1H	0...1	1	[-]
Failure of at least one camshaft sensor on a camshaft with IVVT actuator					
LV_ERR_CRK	V/O	0...1H	0...1	1	[-]
Crankshaft sensor failure					
LV_ERR_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Failure of intake camshaft sensor i					
LV_ERR_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Failure of exhaust camshaft sensor i					
LV_INH_DIAG_SYN_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft i synchronization diagnosis					
LV_INH_DIAG_SYN_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i synchronization diagnosis					
LV_INH_DIAG_PER_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft i period diagnosis					
LV_INH_DIAG_PER_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i period diagnosis					
LV_INH_DIAG_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft i plausibility diagnosis					
LV_INH_DIAG_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i plausibility diagnosis					
LV_INH_DIAG_TOOTH_OFF_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i one-tooth-off diagnosis					
LV_INH_DIAG_TOOTH_OFF_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i one-tooth-off diagnosis					
LV_INH_DIAG_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft i position diagnosis					
LV_INH_DIAG_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i position diagnosis					
LV_INH_DIAG_CRK_SYN	V/O	0...1H	0...1	1	[-]
Inhibition of Crankshaft sensor synchronization diagnosis					
LV_INH_DIAG_CRK_TOOTH	V/O	0...1H	0...1	1	[-]
Inhibition of Crankshaft sensor tooth number diagnosis					
LV_INH_DIAG_CRK_TOOTH_PER	V/O	0...1H	0...1	1	[-]
Inhibition of Crankshaft sensor tooth period diagnosis					
LV_INH_DIAG_CRK_PLAUS	V/O	0...1H	0...1	1	[-]
Inhibition of Crankshaft sensor plausibility diagnosis					
LV_INH_DIAG_CRK_OC	V/O	0...1H	0...1	1	[-]
Inhibition of Crankshaft sensor circuit diagnosis					
LV_INH_DIAG_ENSD	V/O	0...1H	0...1	1	[-]
Inhibition of cam/crk diagnosis during fail safe delay					
LV_IGN_INJ_LOCK_REQ	V/O	0...1H	0...1	1	[-]
Request to lock ignition and/or injection due to backwards rotation detection					
LV_INH_DIAG_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft sensor diagnosis for crankshaft synchronization					

LV_INH_DIAG_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft sensor diagnosis for crankshaft synchronization					
LV_INH_CDN_PER_CAM_IN[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of intake camshaft i period diagnosis					
LV_INH_CDN_PER_CAM_EX[NC_NR_CAM_CBK]	V/O	0...1H	0...1	1	[-]
Inhibition of exhaust camshaft i period diagnosis					

Input data:

LV_ERR_CRK_SYN	LV_ERR_CRK_TOOTH	LV_ERR_CRK_PLAUS	LV_ERR_CRK_OC
NLC_IVVT_IN	NLC_IVVT_EX	LV_ERR_SYN_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_SYN_CAM_EX[NC_NR_CAM_CBK]
LV_ERR_PER_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_PER_CAM_EX[NC_NR_CAM_CBK]	LC_NOT_ADJ_CAM_IVVT_IN[NC_NR_CBK_IVVT]	LC_NOT_ADJ_CAM_IVVT_EX[NC_NR_CBK_IVVT]
LV_ERR_PLAUS_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_PLAUS_CAM_EX[NC_NR_CAM_CBK]	LV_ERR_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_REF_CRK_CAM_EX[NC_NR_CAM_CBK]
LV_ERR_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	LV_ERR_CRK_TOOTH_PER	LV_IGK
LV_STOP_ENG	N_TOOTH	LV_ORNG_CAM_SYN_CRK	LV_ENG_BACK_DET
LV_ENG_BACK_CFM	LV_SYN_VLD	LV_ORNG_TOOTH_PER_CRK	NC_NR_CAM_CBK
LV_LIH_ERR_CAM	LV_ERR_TOOTH_OFF_IN[NC_NR_CAM_CBK]	LV_ERR_TOOTH_OFF_EX[NC_NR_CAM_CBK]	CONF_CAM_VVT_EX
N_32	MAF_HB		

A.18.1 Calculation of Fail Safe Delay

General information:

Calculation of the time delay to inhibit cam/crk diagnosis when there is a risk of reverse engine rotation.

Application conditions:

Initialisation: no initialization value.

Recurrence: 10ms **and** 1 to 0 transition of LV_IGK

Activation: no condition

Deactivation: no condition

Formula section:

If LV_IGK = 0 **And** LV_INH_DIAG_ENSD = 0

Then *check N_TOOTH every 10ms*

If 0 < N_TOOTH < C_N_NOT_REST

Then *Inhibition of all diagnosis functions except sensor circuit diagnosis*

LV_INH_DIAG_ENSD = 1

Wait for a time C_T_DLY_REST_IGK

LV_INH_DIAG_ENSD = 0

Endif

Endif

If LV_STOP_ENG = 1 **and** LV_ENG_BACK_DET = 0

/* to be checked before LV_ENG_BACK_DET reset on LV_STOP_ENG transition */

and /* don't activate fail safe delay if cam/crk synchronization is out of range */

LV_ORNG_CAM_SYN_CRK = 0

and /* don't activate fail safe delay if tooth period is out of range */

LV_ORNG_TOOTH_PER_CRK = 0

and LV_INH_DIAG_ENSD = 0

then **if** 0 < N_TOOTH < C_N_NOT_REST

(N_TOOTH: most recently calculated engine speed)

then *Inhibition of all diagnosis functions except sensor circuit diagnosis*

LV_INH_DIAG_ENSD = 1

Wait for a time C_T_DLY_REST_STALL

Stop timer C_T_DLY_REST_STALL at 0 to 1 transition of LV_IGK

LV_INH_DIAG_ENSD = 0

endif

endif

A.18.2 Diagnosis general inhibition in case of engine backwards rotation detection

General information:

Inhibits cam/crk diagnosis when a reverse engine rotation has been detected until confirmation/nullification.

Application conditions:

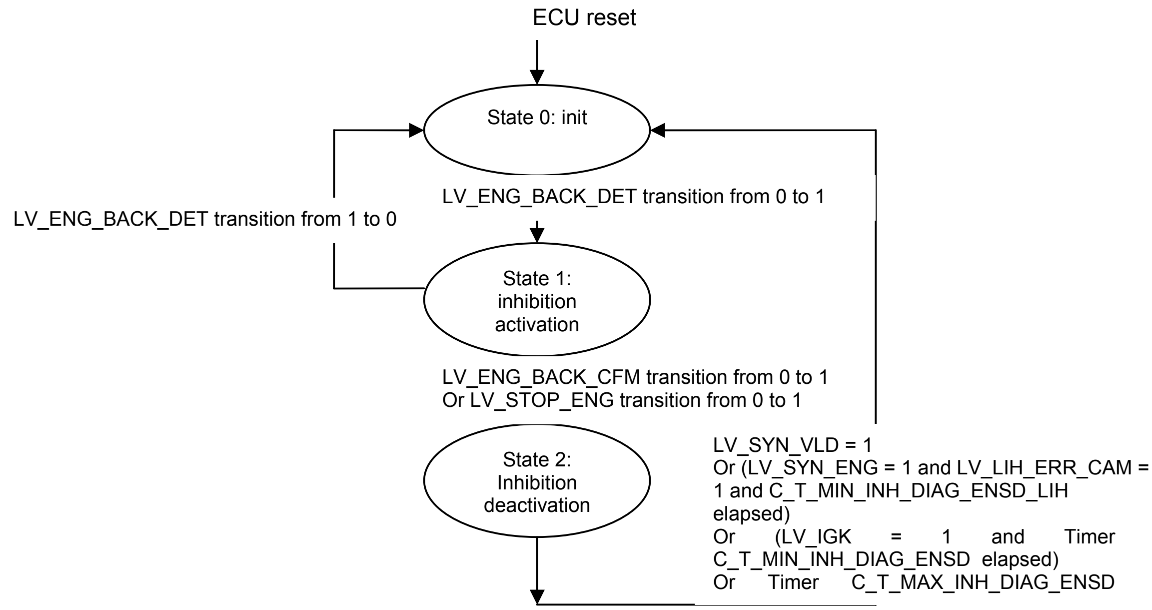
Initialisation: no initialization value.

Recurrence: 10ms **and** transition from 1 to 0 of LV_IGK **and** every update of LV_ENG_BACK_DET or LV_ENG_BACK_CFM **and** transition from 0 to 1 of LV_SYN_VLD **and** transition from 0 to 1 of LV_STOP_ENG **and** transition from 0 to 1 of LV_SYN_ENG in case of camshaft limp_home mode (LV_LIH_ERR_CAM = 1)

Activation: LC_ENG_BACK_INH = 0

Deactivation: when application conditions are not fulfilled

Formula section:



State 0: Initialisation

Input conditions:

External event:	ECU reset
From state 1:	LV_ENG_BACK_DET transition from 1 to 0
From state 2:	LV_SYN_VLD = 1
	Or (LV_IGK = 1 and timer C_T_MIN_INH_DIAG_ENSD elapsed)
	Or timer C_T_MAX_INH_DIAG_ENSD elapsed
	Or (LV_SYN_ENG = 1 and LV_LIH_ERR_CAM = 1 and timer C_T_MIN_INH_DIAG_ENSD_LIH elapsed)

Output conditions:

To state 1:	LV_ENG_BACK_DET transition from 0 to 1
	And Timer C_T_DLY_REST_IGK stopped

Action in the state:

Set timer to C_T_MIN_INH_DIAG_ENSD

Set timer to C_T_MAX_INH_DIAG_ENSD

Set timer to C_T_MIN_INH_DIAG_ENSD_LIH

Transition action:

None

State 1: Inhibition activation

Input conditions:

From state 0: LV_ENG_BACK_DET transition from 0 to 1

Output conditions:

To state 0: ECU reset

Or LV_ENG_BACK_DET transition from 1 to 0

To state 2: LV_ENG_BACK_CFM transition from 0 to 1

Or LV_STOP_ENG transition from 0 to 1

Action in the state: LV_INH_DIAG_ENSD = 1

LV_IGN_INJ_LOCK_REQ = 1

Stop timer C_T_MIN_INH_DIAG_ENSD if on-going

Stop timer C_T_MAX_INH_DIAG_ENSD if on-going

Stop timer C_T_MIN_INH_DIAG_ENSD_LIH if on-going

Transition action:

To state 0: LV_INH_DIAG_ENSD = 0

LV_IGN_INJ_LOCK_REQ = 0

To State 2: Start timer C_T_MAX_INH_DIAG_ENSD

Start timer C_T_MIN_INH_DIAG_ENSD if LV_IGK = 0

Start timer C_T_MIN_INH_DIAG_ENSD_LIH

State 2: Inhibition deactivation

Input conditions:

From state 1: LV_ENG_BACK_CFM transition from 0 to 1

Or LV_STOP_ENG transition from 0 to 1

Output conditions:

To state 0: ECU reset

Or LV_SYN_VLD = 1

Or (LV_IGK = 1 and timer C_T_MIN_INH_DIAG_ENSD elapsed)

Or timer C_T_MAX_INH_DIAG_ENSD elapsed

Or (LV_SYN_ENG = 1 and LV_LIH_ERR_CAM = 1

and timer C_T_MIN_INH_DIAG_ENSD_LIH elapsed)

Action in the state: Start timer C_T_MIN_INH_DIAG_ENSD
on transition of LV_IGK from 1 to 0

Transition action:

To state 0: LV_INH_DIAG_ENSD = 0

LV_IGN_INJ_LOCK_REQ = 0

Remark: if C_T_MIN_INH_DIAG_ENSD = 0, then timer is started at transition of LV_IGK from 1 to 0 and considered as elapsed immediately.

A.18.3 Diagnosis Manager

General information:

One purpose of the diagnosis manager is to provide a summary of diagnostic information, in order to facilitate taking the necessary actions in case of a failure, e.g. switch to limp-home mode. The corresponding outputs are crankshaft limp-home request a general camshaft error flag for IVVT, and a general camshaft error for synchronization.

The other purpose is to avoid getting multiple error codes and freeze frames for the same failure. The corresponding outputs are the inhibit flags for the various diagnosis functions.

i = 1...NC_NR_CAM_CBK

Application conditions:

Initialisation: no initialization value.

Recurrence: every change in the error flags of one of the diagnostic functions

or when error memory is cleared

Activation:

Deactivation:

Formula section:

Setting of LV_ERR_CRK

if LV_ERR_CRK_SYN = 1
or LV_ERR_CRK_TOOTH_PER = 1
or LV_ERR_CRK_TOOTH = 1
or LV_ERR_CRK_PLAUS = 1
or LV_ERR_CRK_OC = 1
then LV_ERR_CRK = 1

if LV_ERR_CRK_SYN = 0
and LV_ERR_CRK_TOOTH_PER = 0
and LV_ERR_CRK_TOOTH = 0
and LV_ERR_CRK_PLAUS = 0
and LV_ERR_CRK_OC = 0
then LV_ERR_CRK = 0

If LV_ERR_CRK = 1 then all other diagnoses on the crankshaft signal will be inhibited via the interfaces LV_INH_DIAG_CRK_SYN, LV_INH_DIAG_CRK_TOOTH, LV_INH_DIAG_CRK_TOOTH_PER, LV_INH_DIAG_CRK_PLAUS and LV_INH_DIAG_CRK_OC, except the one that detected the failure.

Setting of LV_ERR_CAM_IN(EX)[i]

```
if    LV_ERR_SYN_CAM_IN[i] = 1
            or      LV_ERR_PER_CAM_IN [i] = 1
            or      LV_ERR_PLAUS_CAM_IN [i] = 1
            or      LV_ERR_REF_CRK_CAM_IN [i] = 1
            or      LV_ERR_SYN_CRK_CAM_IN [i] = 1
            or      LV_ERR_TOOTH_OFF_IN [i] = 1
            then    LV_ERR_CAM_IN [i] = 1
endif

if    LV_ERR_SYN_CAM_IN [i] = 0
            and     LV_ERR_PER_CAM_IN [i] = 0
            and     LV_ERR_PLAUS_CAM_IN [i] = 0
            and     LV_ERR_REF_CRK_CAM_IN [i] = 0
            and     LV_ERR_SYN_CRK_CAM_IN [i] = 0
            and     LV_ERR_TOOTH_OFF_IN [i] = 0
then        LV_ERR_CAM_IN [i] = 0
endif


If(1)       CONF_CAM_VVT_EX = 0    (no exhaust Camshaft sensor / VVT present)
then(1)     LV_ERR_CAM_EX[i] = LV_ERR_CAM_IN [i]
            LV_INH_DIAG_SYN_CAM_EX[i] = 1
            LV_INH_DIAG_PER_CAM_EX[i] = 1
            LV_INH_DIAG_PLAUS_CAM_EX[i] = 1
            LV_INH_DIAG_REF_CRK_CAM_EX[i] = 1
            LV_INH_DIAG_TOOTH_OFF_EX[i] = 1
else(1)     (exhaust Camshaft sensor / VVT present)
    if      LV_ERR_SYN_CAM_EX [i] = 1
    or      LV_ERR_PER_CAM_EX [i] = 1
    or      LV_ERR_PLAUS_CAM_EX [i] = 1
    or      LV_ERR_REF_CRK_CAM_EX [i] = 1
    or      LV_ERR_SYN_CRK_CAM_EX [i] = 1
    or      LV_ERR_TOOTH_OFF_EX [i] = 1
    then    LV_ERR_CAM_EX [i] = 1
```

```
        endif
        if      LV_ERR_SYN_CAM_EX [i] = 0
        and     LV_ERR_PER_CAM_EX [i] = 0
        and     LV_ERR_PLAUS_CAM_EX [i] = 0
        and     LV_ERR_REF_CRK_CAM_EX [i] = 0
        and     LV_ERR_SYN_CRK_CAM_EX [i] = 0
        and     LV_ERR_TOOTH_OFF_EX [i] = 0
        then    LV_ERR_CAM_EX [i] = 0
        endif
    endif(1)
```

If LV_ERR_CAM_IN(EX)[i] = 1 then other diagnoses on the same camshaft signal will be inhibited via the interfaces LV_INH_DIAG_SYN_CAM_IN(EX)[i], LV_INH_DIAG_PER_CAM_IN(EX)[i], LV_INH_DIAG_PLAUS_CAM_IN(EX)[i], LV_INH_DIAG_REF_CRK_CAM_IN(EX)[i],and LV_INH_DIAG_TOOTH_OFF_IN(EX)[i], except the one that detected the failure.

Setting of LV_ERR_CAM_TOT

VVT actuators existing in the system are given by a logical combination of the configuration data NLC_IVVT_IN(EX) and the calibration data LC_NOT_ADJ_CAM_IVVT_IN(EX)[i].

LV_ERR_CAM_TOT = 1 if LV_ERR_CAM_IN(EX)[i] = 1 for one of the sensors on camshafts with IVVT actuator.

LV_ERR_CAM_TOT = 0 if LV_ERR_CAM_IN(EX)[i] = 0 for all sensors on camshafts with IVVT actuator.

A.18.4 Inhibition of camshaft sensor diagnosis for crankshaft synchronization

General information:

Calculation of the inhibition flag of camshaft sensor diagnosis for crankshaft synchronization in case of possible reverse rotation.

I = 1...NC_NR_CAM_CBK

Application conditions:

Initialisation: LV_INH_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0 at ECU reset **and** 0 to 1 transition of LV_IGK

Recurrence: 10ms **and** transition from 0 to 1 of LV_STOP_ENG **and** transition from 0 to 1 of LV_SYN_VLD

Activation: -

Deactivation: -

Formula section:

If LV_SYN_ENG = 0
and LC_ENG_BACK_INH = 0
and LV_INH_DIAG_ENSD = 0
and 0 < N_TOOTH < C_N_NOT_REST
and LV_ORNG_TOOTH_PER_CRK = 1
then LV_INH_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1
endif
If LV_SYN_VLD = 1
then LV_INH_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0
endif

A.18.5 Inhibition of camshaft sensor diagnosis

General information:

Calculation of the inhibition flag of camshaft sensor diagnosis in case of possible wrong failure conditions (high N or MAF).

I = 1...NC_NR_CAM_CBK

Application conditions:

Initialisation: LV_INH_CDN_PER_CAM_IN(EX)[i] = 0
at ECU reset **or** 0 to 1 transition of LV_IGK **or** FMY reset

Recurrence: Every camshaft signal edge

Activation: -

Deactivation: -

Formula section:

If CONF_CAM_VVT_EX = 0 *(no exhaust Camshaft sensor / VVT present)*
then **if** N_32 > C_N_32_MAX_PER_CAM_IN
or MAF_HB > C_MAF_HB_MAX_PER_CAM_IN
then LV_INH_CDN_PER_CAM_IN[i] = 1
else LV_INH_CDN_PER_CAM_IN[i] = 0
endif
else *(exhaust Camshaft sensor / VVT present)*
if N_32 > C_N_32_MAX_PER_CAM_IN(_EX)
or MAF_HB > C_MAF_HB_MAX_PER_CAM_IN(_EX)
then LV_INH_CDN_PER_CAM_IN(_EX)[i] = 1
else LV_INH_CDN_PER_CAM_IN(_EX)[i] = 0
endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_NOT_REST	1	0...FFH	0...8160	32	[rpm]
Engine speed threshold for cam/crk diagnosis deactivation					
C_T_DLY_REST_IGK	1	0...FFH	0...2550	10	[ms]
Delay time for diagnosis reactivation after key-off at low engine speed					
C_T_DLY_REST_STALL	1	0...FFFFH	0...655350	10	[ms]
Delay time for diagnosis reactivation after engine stalling					
C_T_MIN_INH_DIAG_ENSD	1	0...FFFFH	0...655350	10	[ms]
Minimum delay before reactivating diagnosis in case of key off to on after engine backwards rotation confirmation					
C_T_MAX_INH_DIAG_ENSD	1	0...FFFFH	0...655350	10	[ms]
Maximum delay before reactivating diagnosis after engine backwards rotation confirmation					
LC_ENG_BACK_INH	1	0...1H	0...1	1	[-]
Inhibition of engine backwards rotation detection					
C_T_MIN_INH_DIAG_ENSD_LIH	1	0...FFFFH	0...655350	10	[ms]
Minimum delay before reactivating diagnosis after engine backwards rotation confirmation in camshaft limp-home mode					
C_N_32_MAX_PER_CAM_IN	1	0...FFH	0...8160	32	rpm
Engine speed threshold for diag inhibition of period CAM_IN					
C_N_32_MAX_PER_CAM_EX	1	0...FFH	0...8160	32	rpm
Engine speed threshold for diag inhibition of period CAM_EX					
C_MAF_HB_MAX_PER_CAM_IN	1	0...FFH	0...1389	5.447	mg/stk
Mass aif flow threshold for diag inhibition of period CAM_IN					
C_MAF_HB_MAX_PER_CAM_EX	1	0...FFH	0...1389	5.447	mg/stk
Mass aif flow threshold for diag inhibition of period CAM_EX					

A.19 Camshaft Sensor Diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_PER_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
LV_CDN_DIAG_PER_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
ERR_SYM_PER_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the failure camshaft period too short					
ERR_SYM_PER_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the failure camshaft period too short					
LV_ERR_PER_CAM_IN[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present failure camshaft period too short					
LV_ERR_PER_CAM_EX[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present failure camshaft period too short					
LV_END_DIAG_PER_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of the failure camshaft period too short					
LV_END_DIAG_PER_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of the failure camshaft period too short					
LV_CDN_DIAG_SYN_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for intake camshaft i					
LV_CDN_DIAG_SYN_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for exhaust camshaft i					
ERR_SYM_SYN_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the intake camshaft i sensor diagnosis					
ERR_SYM_SYN_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the exhaust camshaft i sensor diagnosis					
LV_ERR_SYN_CAM_IN[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present intake camshaft i sensor failure					
LV_ERR_SYN_CAM_EX[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present exhaust camshaft i failure sensor out of range					
LV_END_DIAG_SYN_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]

Result available of intake camshaft i sensor diagnosis					
LV_END_DIAG_SYN_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of exhaust camshaft i sensor diagnosis					

Input data:

LV_ORNG_RATIO_CAM_IN[NC_NR_CAM_CBK]	LV_ORNG_PER_CAM_IN[NC_NR_CAM_CBK]	LV_ORNG_RATIO_CAM_EX[NC_NR_CAM_CBK]	LV_ORNG_PER_CAM_EX[NC_NR_CAM_CBK]
LV_ACT_CAM_IN[NC_NR_CAM_CBK]	LV_ACT_CAM_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_SYN_CAM_IN[NC_NR_CAM_CBK]	LV_INH_DIAG_PER_CAM_IN[NC_NR_CAM_CBK]
LV_INH_DIAG_SYN_CAM_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_PER_CAM_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_ENSD	CONF_CAM_VVT_EX
LV_INH_CDN_PER_CAM_IN[NC_NR_CAM_CBK]	LV_INH_CDN_PER_CAM_EX[NC_NR_CAM_CBK]		

A.19.1.1 Camshaft Segment Period Diagnosis

FUNCTION DESCRIPTION:

General information:

Camshaft signal acquisition validates the segment period by minimum limits derived from the maximum engine speed. The diagnostic state indicates when a segment period was too short, e.g. due to supplementary signal transitions. The camshaft signal acquisition will filter out such signal transitions as far as possible. If there are too many additional pulses however, the camshaft signal acquisition may not be able to synchronize any more, or the accuracy of camshaft position measurement may be degraded.

i = 1...NC_NR_CAM_CBK

Description:

Error-symptoms are defined for this diagnostic function as following:

```
□□□□
│ │ │ └─> Camshaft segment period too short "CAM_PER"   (= SYM_0)
│ │ └───> (not used)                                     (= SYM_1)
│ └─────> (not used)                                     (= SYM_2)
└───────> (not used)                                     (= SYM_3)
```

Application conditions:

Initialisation:
LV_CDN_DIAG_PER_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_PER_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_ERR_PER_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PER_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*

Recurrence: every camshaft signal edge

Activation: *for inlet camshaft*

LV_ACT_CAM_IN[i] = 1 AND LV_INH_DIAG_PER_CAM_IN[i] = 0 AND LV_INH_DIAG_ENSD = 0 AND LV_INH_CDN_PER_CAM_IN[i] = 0

for exhaust camshaft

LV_ACT_CAM_EX[i] = 1 AND [LV_INH_DIAG_PER_CAM_EX[i] = 0 and CONF_CAM_VVT_EX = 1] AND LV_INH_DIAG_ENSD = 0 AND LV_INH_CDN_PER_CAM_EX[i] = 0

When the activation conditions are fulfilled then LV_CDN_DIAG_PER_CAM_IN(EX)[i] = 1

Deactivation:

When the activation cond. are not fulfilled then LV_CDN_DIAG_PER_CAM_IN(EX)[i] = 0

Formula section:

Symptoms calculation:

If LV_ORNG_PER_CAM_IN(EX)[i] = 1

Then ERR_SYM_PER_CAM_IN(EX)[i] = “SYM_0” {detection of symptom CAM_PER]

Else ERR_SYM_PER_CAM_IN(EX)[i] = NO_SYM

Endif

LV_ORNG_PER_CAM_IN(EX)[i] is reset after reading.

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering:

Apply filter on current symptoms
If filtering result available (after debounce)
Then LV_ERR_PER_CAM_IN(EX)[i] = filtering result
LV_END_DIAG_PER_CAM_IN(EX)[i] = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms:

Diagnostic Camshaft sensor	Symptom description	Symptom	Filter type
Camshaft segment period diagnosis for intake	*CAM_PER*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	
Camshaft segment period diagnosis for exhaust	*CAM_PER*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	

A.19.1.2 Camshaft Synchronization Diagnosis

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect camshaft sensor failure. Each time the camshaft signal synchronization algorithm fails, it will set the diagnostic flag LV_ORNG_RATIO_CAM_IN(EX)[i] for the appropriate camshaft.

The camshaft signal acquisition will filter out supplementary signal transitions as far as possible. If there are too many additional pulses however, the camshaft signal acquisition will not be able to synchronize any more. Synchronization may also be impossible due to missing or inaccurate signal transitions of the sensor.

i = 1...NC_NR_CAM_CBK

Description:

Error-symptoms are defined for this diagnostic function as following:

```
□□□□
│ │ │ └─> Camshaft synchronization failure "SYNC_LOSS"     (= SYM_0)
│ │ └───> (not used)                                        (= SYM_1)
│ └─────> (not used)                                        (= SYM_2)
└───────> (not used)                                        (= SYM_3)
```

Application conditions:

Initialisation:

LV_CDN_DIAG_SYN_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_SYN_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_ERR_SYN_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
ERR_SYM_SYN_CAM_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*

Recurrence: *every camshaft signal edge*

Activation: *for inlet camshaft*

LV_ACT_CAM_IN[i] = 1 AND LV_INH_DIAG_ENSD = 0

And LV_INH_DIAG_SYN_CAM_IN[i] = 0

for exhaust camshaft

LV_ACT_CAM_EX[i] = 1 AND LV_INH_DIAG_ENSD = 0

And [LV_INH_DIAG_SYN_CAM_EX[i] = 0 and CONF_CAM_VVT_EX = 1]

When activation conditions are fulfilled then LV_CDN_DIAG_SYN_CAM_IN(EX)[i] = 1.

Deactivation:

When activation conditions are not fulfilled then LV_CDN_DIAG_SYN_CAM_IN(EX)[i] = 0.

Formula section:

Symptoms calculation:

If LV_ORNG_RATIO_CAM_IN(EX)[i] = 1

Then ERR_SYM_SYN_CAM_IN(EX)[i] = "SYM_0" {Detection of symptom SYNC_LOSS}

Else ERR_SYM_SYN_CAM_IN(EX)[i] = NO_SYM

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering:

Apply filter on current symptoms
If filtering result available (after debounce)
Then LV_ERR_SYN_CAM_IN(EX)[i] = filtering result
LV_END_DIAG_SYN_CAM_IN(EX)[i] = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms:

Diagnostic Camshaft synchronization	Symptom description	Symptom	Filter type
Camshaft synchronization diagnosis for intake	*SYNC_LOSS*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	
Camshaft synchronization diagnosis for exhaust	*SYNC_LOSS*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_DIAG_PER_CAM_IN	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for intake camshaft segment period diagnosis					
C_ABC_INC_DIAG_PER_CAM_EX	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for exhaust camshaft segment period diagnosis					
C_ABC_MAX_DIAG_PER_CAM_IN	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for intake camshaft segment period diagnosis					
C_ABC_MAX_DIAG_PER_CAM_EX	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for exhaust camshaft segment period diagnosis					
C_ABC_INC_DIAG_SYN_CAM_IN	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for intake camshaft synchronization diagnosis					
C_ABC_INC_DIAG_SYN_CAM_EX	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for exhaust camshaft synchronization diagnosis					
C_ABC_MAX_DIAG_SYN_CAM_IN	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for intake camshaft synchronization diagnosis					
C_ABC_MAX_DIAG_SYN_CAM_EX	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for exhaust camshaft synchronization diagnosis					

A.20 Camshaft Position Diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for intake camshaft i					
LV_CDN_DIAG_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for exhaust camshaft i					
ERR_SYM_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the intake camshaft i position diagnosis					
ERR_SYM_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the exhaust camshaft i position diagnosis					
LV_ERR_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present intake camshaft i failure reference violation					
LV_ERR_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present exhaust camshaft i failure reference violation					
LV_END_DIAG_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of intake camshaft i position diagnosis					
LV_END_DIAG_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of exhaust camshaft i position diagnosis					
LV_CDN_DIAG_TOOTH_OFF_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
One-tooth-off diagnosis condition for intake camshaft i					
LV_CDN_DIAG_TOOTH_OFF_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
One-tooth-off diagnosis condition for exhaust camshaft i					
ERR_SYM_TOOTH_OFF_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the intake i one-tooth-off diagnosis					
ERR_SYM_TOOTH_OFF_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the exhaust i one-tooth-off diagnosis					
LV_ERR_TOOTH_OFF_IN[NC_NR_CAM_CBK]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present intake camshaft i one-tooth-off failure					
LV_ERR_TOOTH_OFF_EX[NC_NR_CAM_CBK]	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present exhaust camshaft i one-tooth-off failure					
LV_END_DIAG_TOOTH_OFF_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]

Result available of intake camshaft i one-tooth-off diagnosis					
LV_END_DIAG_TOOTH_OFF_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of exhaust camshaft i one-tooth-off diagnosis					

Input data:

PSN_EDGE_AD_CAM_IN[NC_NR_EDGE_CAM_IN][NC_NR_CAM_CBK]	PSN_EDGE_AD_CAM_EX[NC_NR_EDGE_CAM_EX][NC_NR_CAM_CBK]	LV_VLD_PSN_CAM_IN[NC_NR_CAM_CBK]	LV_VLD_PSN_CAM_EX[NC_NR_CAM_CBK]
LV_INH_DIAG_REF_CRK_CAM_IN[NC_NR_CAM_CBK]	LV_INH_DIAG_REF_CRK_CAM_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_ENSD	CAM_PSN_LST_REF_AD_IN[NC_NR_CAM_CBK]
CAM_PSN_LST_REF_AD_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_TOOTH_OFF_IN[NC_NR_CAM_CBK]	LV_INH_DIAG_TOOTH_OFF_EX[NC_NR_CAM_CBK]	LV_TOOTH_OFF_DET_ENA_IN[NC_NR_CAM_CBK]
LV_TOOTH_OFF_DET_ENA_EX[NC_NR_CAM_CBK]	CONF_CAM_VVT_EX		

A.20.1 Camshaft position diagnosis

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect when the camshaft reference position is outside of the designed range relative to the engine position from crankshaft. This allows to detect engine misbuilds (wrong assembly of the valve train), camshaft drive defects, or sensor failures that are not detected by the normal signal diagnosis.

i = 1...NC_NR_CAM_CBK z = 0...NC_NR_EDGE_CAM_IN/EX

Description:

Error-symptoms are defined for this diagnostic function as following:

```
□□□□
│ │ │ └─> Camshaft to crankshaft reference violated "REF_CRK_CAM"   (= SYM_0)
│ │ └───> (not used)                                               (= SYM_1)
│ └─────> (not used)                                               (= SYM_2)
└───────> (not used)                                               (= SYM_3)
```

Application conditions

Application conditions:

Initialisation:

LV_CDN_DIAG_REF_CRK_CAM_IN(EX)[i] = *Refer to filtering configuration for the init. value*
LV_END_DIAG_REF_CRK_CAM_IN(EX)[i] = *Refer to filtering configuration for the init. value*
LV_ERR_REF_CRK_CAM_IN(EX)[i] = *Refer to filtering configuration for the init. value*
ERR_SYM_REF_CRK_CAM_IN(EX)[i] = *Refer to filtering configuration for the init. value*

Recurrence: *every update of the camshaft edge position adaptation values PSN_EDGE_AD_CAM_IN(EX)[z][i]*

Activation: LV_VLD_PSN_CAM_IN(EX)[i] = 1
And
for inlet camshaft
LV_INH_DIAG_REF_CRK_CAM_IN[i] = 0
for exhaust camshaft
[LV_INH_DIAG_REF_CRK_CAM_EX[i] = 0
and CONF_CAM_VVT_EX = 1]

And LV_INH_DIAG_ENSD = 0

When the activation cond. are fulfilled then LV_CDN_DIAG_REF_CRK_CAM_IN(EX)[i] = 1.

Deactivation:

When the activation conditions are not fulfilled then
LV_CDN_DIAG_REF_CRK_CAM_IN(EX)[i] = 0

Formula Section

Symptoms calculation:

If PSN_EDGE_AD_CAM_IN(EX)[z][i] < C_TOL_REF_CRK_CAM_MIN_IN(EX)

or PSN_EDGE_AD_CAM_IN(EX)[z][i] > C_TOL_REF_CRK_CAM_MAX_IN(EX)

Then ERR_SYM_REF_CRK_CAM_IN(EX)[i] = "SYM_0" {Detection of symptom REF_CRK_CAM}

Else ERR_SYM_REF_CRK_CAM_IN(EX)[i] = NO_SYM

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering :

apply filter on current symptoms
If filtering result available (after debounce)
then LV_ERR_REF_CRK_CAM_IN(EX)[i] = filtering result
LV_END_DIAG_REF_CRK_CAM_IN(EX)[i] = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms :

Diagnostic Camshaft position diagnosis	Symptom description	Symptom	Filter type
Camshaft position diagnosis for intake	*REF_CRK_CAM*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	
Camshaft position diagnosis for exhaust	*REF_CRK_CAM*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	

A.20.2 One-tooth-off diagnosis

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect a sudden drift is the camshaft position.

i = 1...NC_NR_CAM_CBK   z = 0...NC_NR_EDGE_CAM_IN/EX

Description:

Error-symptoms are defined for this diagnostic function as following:

```
┌─┬─┬─┬─┐
│ │ │ └─> Sudden drift "ONE_TOOTH_OFF"          (= SYM_0)
│ │ └───> (not used)                            (= SYM_1)
│ └─────> (not used)                            (= SYM_2)
└───────> (not used)                            (= SYM_3)
```

Application conditions:

Initialisation:

LV_CDN_DIAG_TOOTH_OFF_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_TOOTH_OFF_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
LV_ERR_TOOTH_OFF_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*
ERR_SYM_TOOTH_OFF_IN(EX)[i] = *Refer to filtering configuration for the initialisation value*

Recurrence: at LV_TOOTH_OFF_DET_ENA_IN(EX)[i] 0 to 1 transition

Activation: *for inlet camshaft*

LV_INH_DIAG_TOOTH_OFF_IN [i] = 0

for exhaust camshaft

[LV_INH_DIAG_TOOTH_OFF_EX[i] = 0 and CONF_CAM_VVT_EX = 1]

And LV_INH_DIAG_ENSD = 0

When the activation conditions are fulfilled then LV_CDN_DIAG_TOOTH_OFF_IN(EX)[i] = 1.

Deactivation:

When the activation cond. are not fulfilled then LV_CDN_DIAG_TOOTH_OFF_IN(EX)[i] = 0

Formula Section

Symptoms calculation:

If CAM_PSN_LST_REF_AD_IN(EX)[i] = PSN_EDGE_AD_CAM_IN(EX)[1][i]

Then ERR_SYM_REF_CRK_CAM_IN(EX)[i] = NO_SYM

Else ERR_SYM_REF_CRK_CAM_IN(EX)[i] = “SYM_0” {Detection of symptom ONE_TOOTH_OFF}

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering :

apply filter on current symptoms
If filtering result available (after debounce)
then LV_ERR_TOOTH_OFF_IN(EX)[i] = filtering result
LV_END_DIAG_TOOTH_OFF_IN(EX)[i] = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms :

Diagnostic One-tooth-off diagnosis	Symptom description	Symptom	Filter type
One-tooth-off diagnosis for intake	*ONE_TOOTH_OFF*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	
One-tooth-off diagnosis for exhaust	*ONE_TOOTH_OFF*	*SYM_0*	*STD_INI*
		SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_MAX_DIAG_TOOTH_OFF_EX	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for exhaust one-tooth-off diagnosis					
C_ABC_INC_DIAG_REF_CRK_CAM_EX	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for exhaust camshaft position diagnosis					
C_ABC_MAX_DIAG_REF_CRK_CAM_IN	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for intake camshaft position diagnosis					
C_ABC_MAX_DIAG_REF_CRK_CAM_EX	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for exhaust camshaft position diagnosis					
Reference position tolerance range for intake camshaft					
Reference position tolerance range for exhaust camshaft					
C_ABC_INC_DIAG_TOOTH_OFF_IN	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for intake one-tooth-off diagnosis					
C_ABC_INC_DIAG_TOOTH_OFF_EX	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for exhaust one-tooth-off diagnosis					
C_ABC_MAX_DIAG_TOOTH_OFF_IN	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for intake one-tooth-off diagnosis					
C_ABC_MAX_DIAG_TOOTH_OFF_EX	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for exhaust one-tooth-off diagnosis					
C_TOL_REF_CRK_CAM_MIN_IN	1	0...200H	-32...0	0.0625	[°CRK]
Minimum reference position tolerance range for intake camshaft					
C_TOL_REF_CRK_CAM_MAX_IN	1	0...1FFH	0....31.9375	0.0625	[°CRK]
Maximum reference position tolerance range for intake camshaft					
C_TOL_REF_CRK_CAM_MIN_EX	1	0...200H	-32...0	0.0625	[°CRK]
Minimum reference position tolerance range for exhaust camshaft					
C_TOL_REF_CRK_CAM_MAX_EX	1	0...1FFH	0....31.9375	0.0625	[°CRK]
Minimum reference position tolerance range for exhaust camshaft					

A.20.3 Camshaft Sensor Diagnosis (Appl. Inc.)

FUNCTION DESCRIPTION:

This is the definition of DTC (P-codes) for the different diagnostic functions of the camshaft signal.

Input data:

LV_ERR_REF_CRK_CAM_IN_i	LV_ERR_PER_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_SYN_CAM_IN_i	
LV_ERR_SYN_CRK_CAM_IN_i	LV_ERR_PLAUS_CAM_IN_i		

Application condition :

Recurrence: At the update of each one of the camshaft sensor diagnostic flags

*** Application Assistance:**

Limp home functions:
- KNK limp home value calculation
- No preinjection performed

The following control functions are inhibited:
- Knock control
- Anti-jerk function
- Engine overheating protection at VS = 0
- Gear ratio calculation
- Downstream fuel trim regulation

The following adaptation functions are inhibited:
- Knock adaptation
- Catalyst heating adaptation (indirect)
- Segment time adaptation

The following diagnosis functions are inhibited:
- Camshaft sensor diagnosis
- Knock sensor diagnosis
- O2-sensor downstream signal voltage diagnosis (only SCG)
- Misfire detection (ER-calculation)
- O2-sensor upstream switching time diagnosis
- O2-sensor upstream frequency diagnosis
- Catalyst efficiency diagnosis
- O2-sensor downstream switching time diagnosis
- O2-sensor downstream OBDII puc-end diagnosis
- Crankshaft oscillation detection

Configuration for diagnositc symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Intake camshaft sensor bank 1 synchronisation diagnosis	Camshaft synchronisation failure	SYM_0	*STD*
		SYM_1	
		SYM_2	
SYN_CAM_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
CAM_SP_IVVT_IN_1
CAM_IN_1

Intake camshaft sensor bank 1 Segment period diagnosis	Camshaft segment period too short	SYM_0	*STD*
		SYM_1	
		SYM_2	
PER_CAM_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
CAM_SP_IVVT_IN_1
CAM_IN_1

Intake camshaft sensor bank 1 Plausibility diagnosis	Signal missing	SYM_0	*STD*
		SYM_1	
		SYM_2	
PLAUS_CAM_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
CAM_SP_IVVT_IN_1
CAM_IN_1

Intake camshaft Sensor Diagnosis for Crankshaft Synchronisation Bank1	Signal invalid/ CAM not valid for CRK synchronisation	SYM_0	*STD*
		SYM_1	
		SYM_2	
SYN_CRK_CAM_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
CAM_SP_IVVT_IN_1
CAM_IN_1

Intake camshaft position diagnosis bank 1	Camshaft to crankshaft reference violated	SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
REF_CRK_CAM_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
CAM_SP_IVVT_IN_1
CAM_IN_1

A.21 Crankshaft Sensor Diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_CRK_TOOTH	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
ERR_SYM_CRK_TOOTH	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the crankshaft failure tooth number error					
LV_INH_MIS_CRK	V/O	0...1H	0...1	1	[-]
Inhibition of misfire detection with crankshaft tooth number error					
LV_ERR_CRK_TOOTH	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present crankshaft failure tooth number error					
LV_END_DIAG_CRK_TOOTH	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of crankshaft tooth number error					
LV_CDN_DIAG_CRK_TOOTH_PER	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for crankshaft diagnosis of implausible tooth period					
ERR_SYM_CRK_TOOTH_PER	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the crankshaft failure implausible tooth period					
LV_ERR_CRK_TOOTH_PER	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present crankshaft failure tooth number error					
LV_END_DIAG_CRK_TOOTH_PER	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of crankshaft diagnosis implausible tooth period					
LV_CDN_DIAG_CRK_SYN	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition for crankshaft diagnosis with loss of synchronization					
ERR_SYM_CRK_SYN	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the crankshaft failure with loss of synchronization					
LV_ERR_CRK_SYN	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present crankshaft failure with loss of synchronization					
LV_END_DIAG_CRK_SYN	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of crankshaft diagnosis with loss of synchronization					

Input data:

LV_CRK_SYN	LV_RUN_ENG	LV_ORNG_TOOTH_PER_CRK	LV_ORNG_NR_TOOTH_CRK
LV_LOST_SYN_CRK	LV_INH_DIAG_CRK_SYN	LV_INH_DIAG_CRK_TOOTH	LV_INH_DIAG_CRK_TOOTH_PER
LV_LIH_ERR_CRK	LV_INH_DIAG_ENSD	LV_STOP_ENG	N_32

General information:

The purpose of the function is to detect failures on the crankshaft signal while the system is synchronized.

The diagnosis is divided in three different parts: tooth number error, implausible tooth period, and loss of synchronization on crankshaft signal.

A.21.1.1 Crankshaft Sensor Diagnosis with Tooth Number Error

FUNCTION DESCRIPTION:

General information:

The number of teeth per crankshaft revolution is monitored by the crankshaft signal acquisition. The algorithm may tolerate a number of missing/additional teeth per revolution without loosing synchronization, depending on the used target wheel and on configuration data. The bit LV_ORNG_NR_TOOTH_CRK indicates wrong tooth number during the most recent crankshaft revolution.

If a tooth was missing or added during one revolution, then all variables based on teeth counting will be produced with an error. This concerns for example spark advance, segment time, misfire segments, camshaft position, etc. The purpose of the function is to provide information when the crankshaft signal is inaccurate, in order to take the necessary actions.

Description:

Error-symptoms are defined for this diagnostic function as following:

```
□□□□
│ │ │ └─> Incorrect crankshaft tooth number "TOOTH_NUMBER"     (= SYM_0)
│ │ └───> (not used)                                           (= SYM_1)
│ └─────> (not used)                                           (= SYM_2)
└───────> (not used)                                           (= SYM_3)
```

Application conditions

Initialisation:

LV_CDN_DIAG_CRK_TOOTH = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_CRK_TOOTH = *Refer to filtering configuration for the initialisation value*
LV_ERR_CRK_TOOTH = *Refer to filtering configuration for the initialisation value*
ERR_SYM_CRK_TOOTH = *Refer to filtering configuration for the initialisation value*

Recurrence: *once per crankshaft revolution, after detection of the reference gap, except in the first revolution after synchronization*

Activation: LV_CRK_SYN = 1
And LV_INH_DIAG_CRK_TOOTH = 0
And LV_INH_DIAG_ENSD = 0
And N_32 > C_N_32_MAX_DIAG_CRK_TOOTH

When the activation conditions are fulfilled then LV_CDN_DIAG_CRK_TOOTH = 1.

Deactivation:

When the activation conditions are not fulfilled then LV_CDN_DIAG_CRK_TOOTH = 0.

Formula Section

Symptoms calculation:

If LV_ORNG_NR_TOOTH_CRK = 0

Then ERR_SYM_CRK_TOOTH = NO_SYM

Else ERR_SYM_CRK_TOOTH = "SYM_0" {Detection of symptom TOOTH_NUMBER}

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering:

Apply filter on current symptoms
If filtering result available (after debounce)
Then
LV_ERR_CRK_TOOTH = filtering result
LV_END_DIAG_CRK_TOOTH = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms :

Diagnostic Crankshaft sensor	Symptom description	Symptom	Filter type
Crankshaft sensor diagnosis with tooth number error	*TOOTH_NUMBER*	*SYM_0*	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

Calculation of missing/added tooth information for misfire

Initialisation:

LV_INH_MIS_CRK = 0 at reset and at engine stalling (LV_STOP_ENG transition from 0 to 1).

Recurrence: *once per crankshaft revolution, after detection of the reference gap, except in the first revolution after synchronization*

Activation: LV_CRK_SYN = 1

Deactivation: LV_CRK_SYN = 0

Formula Section

Symptoms calculation:

If LV_ORNG_NR_TOOTH_CRK = 0

Then LV_INH_MIS_CRK = 0

Else LV_INH_MIS_CRK = 1

Endif

LV_ORNG_NR_TOOTH_CRK is reset after reading at the reference gap

A.21.1.2 Crankshaft Sensor Diagnosis with implausible Tooth Period

FUNCTION DESCRIPTION:

General information:

Crankshaft signal acquisition validates the tooth period by limits derived from the current engine speed.

Before synchronization, LV_ORNG_TOOTH_PER_CRK indicates implausible gradient of the tooth period. This flag is also set if a reference gap occurs during valid teeth phase or when the first reference gap cannot be detected.

After synchronization, LV_ORNG_TOOTH_PER_CRK is set if spikes are detected outside the crankshaft window acceptance window.

Description:

Error-symptoms are defined for this diagnostic function as following:

```
[ | | | ]
 | | | └─> Implausible crankshaft tooth period "TOOTH_PERIOD"   (= SYM_0)
 | | └───> (not used)                                          (= SYM_1)
 | └─────> (not used)                                          (= SYM_2)
 └───────> (not used)                                          (= SYM_3)
```

Application conditions:

Initialisation:

LV_CDN_DIAG_CRK_TOOTH_PER = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_CRK_TOOTH_PER = *Refer to filtering configuration for the initialisation value*
LV_ERR_CRK_TOOTH_PER = *Refer to filtering configuration for the initialisation value*
ERR_SYM_CRK_TOOTH_PER = *Refer to filtering configuration for the initialisation value*

Recurrence: if LV_CRK_SYN = 1: every reference gap
If LV_CRK_SYN = 0: when LV_ORNG_TOOTH_PER_CRK is set

Activation: LV_RUN_ENG= 1
And LV_LIH_ERR_CRK = 0
And LV_INH_DIAG_CRK_TOOTH_PER = 0
And LV_INH_DIAG_ENSD = 0

When the activation conditions are fulfilled then LV_CDN_DIAG_CRK_TOOTH_PER = 1.

Deactivation:

When the activation conditions are not fulfilled then LV_CDN_DIAG_CRK_TOOTH_PER = 0.

Formula Section

Symptoms calculation:

Before crankshaft synchronization (LV_CRK_SYN = 0)

If LV_ORNG_TOOTH_PER_CRK = 0

Then ERR_SYM_CRK_TOOTH_PER = NO_SYM

Else ERR_SYM_CRK_TOOTH_PER = “SYM_0" {Detection of symptom TOOTH_PERIOD}

Endif

After crankshaft synchronization (LV_CRK_SYN = 1)

If LV_ORNG_TOOTH_PER_CRK = 1 set at a tooth event or end of tolerance window

Then ERR_SYM_CRK_TOOTH_PER = “SYM_0" {Detection of symptom TOOTH_PERIOD}

Endif

LV_ORNG_TOOTH_PER_CRK is reset after reading

If LV_ORNG_TOOTH_PER_CRK = 0 at the reference gap

Then ERR_SYM_CRK_TOOTH_PER = NO_SYM

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering:

Apply filter on current symptoms
If filtering result available (after debounce)
Then
LV_ERR_CRK_TOOTH_PER = filtering result
LV_END_DIAG_CRK_TOOTH_PER = 1
Deliver the result to Error Management
Endif

Configuration for diagnostic symptoms :

Diagnostic Crankshaft sensor	Symptom description	Symptom	Filter type
Crankshaft sensor diagnosis with implausible tooth period error	*TOOTH_PERIOD*	*SYM_0*	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

A.21.1.3 Crankshaft Sensor Diagnosis with Loss of Synchronization

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect crankshaft failure when the system looses synchronization on the crankshaft signal.

Synchronization will be lost if the reference gap is not detected at the correct position plus/minus a tolerance of missing/additional teeth, depending on the used target wheel and on configuration data.

If the signal fails completely (no more crankshaft signal edges are detected), then no error will be detected by this diagnosis, since the condition is similar to stopped engine. A failure then may be detected by the crankshaft sensor plausibility diagnosis (e.g. test against the camshaft signal).

Description:

Error-symptoms are defined for this diagnostic function as following:

```
□□□□
│ │ │ └─> Loss of synchronization "SYNC_LOSS"     (= SYM_0)
│ │ └───> (not used)                              (= SYM_1)
│ └─────> (not used)                              (= SYM_2)
└───────> (not used)                              (= SYM_3)
```

Application conditions:

Initialisation:

LV_CDN_DIAG_CRK_SYN = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_CRK_SYN = *Refer to filtering configuration for the initialisation value*
LV_ERR_CRK_SYN = *Refer to filtering configuration for the initialisation value*
ERR_SYM_CRK_SYN = *Refer to filtering configuration for the initialisation value*

Recurrence: *once per crankshaft revolution, after the reference gap, except in the first revolution after synchronization, or when the bit LV_LOST_SYN_CRK is set.*

Activation: LV_CRK_SYN = 1
And LV_INH_DIAG_CRK_SYN = 0
And LV_INH_DIAG_ENSD = 0

When activation conditions are fulfilled then LV_CDN_DIAG_CRK_SYN = 1.

Deactivation:

When activation conditions not are fulfilled then LV_CDN_DIAG_CRK_SYN = 0.

Formula Section

Symptoms calculation:

If LV_LOST_SYN_CRK = 0

Then ERR_SYM_CRK_SYN = NO_SYM

Else ERR_SYM_CRK_SYN = “SYM_0" {Detection of symptom SYNC_LOSS}

Endif

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Filtering:

Apply filter on current symptoms
If filtering result available (after debounce)
Then
LV_ERR_CRK_SYN = filtering result
LV_END_DIAG_CRK_SYN = 1
Deliver the result to Error Management

Endif

Configuration for diagnostic symptoms :

Diagnostic Crankshaft sensor	Symptom description	Symptom	Filter type
Crankshaft sensor diagnosis with loss of synchronization	*SYNC_LOSS*	*SYM_0*	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_DIAG_CRK_TOOTH	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for crankshaft tooth number diagnosis					
C_ABC_INC_DIAG_CRK_TOOTH_PER	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for crankshaft tooth period diagnosis					
C_ABC_INC_DIAG_CRK_SYN	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for crankshaft diagnosis with loss of synchronization					
C_ABC_MAX_DIAG_CRK_TOOTH	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for crankshaft tooth number diagnosis					
C_ABC_MAX_DIAG_CRK_TOOTH_PER	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for crankshaft tooth period diagnosis					
C_ABC_MAX_DIAG_CRK_SYN	1	1...FFH	1...255	1	[-]
Anti-bounce counter maximum for crankshaft diagnosis with loss of synchronization					
C_N_32_MAX_DIAG_CRK_TOOTH	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed to enable tooth number diagnosis					

A.22 Crankshaft Sensor Plausibility Diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_CRK_PLAUS	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
ERR_SYM_CRK_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the crankshaft failure synchronization impossible					
LV_ERR_CRK_PLAUS	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Present crankshaft failure synchronization impossible					
LV_END_DIAG_CRK_PLAUS	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of crankshaft signal diagnosis synchronization impossible					
LV_CDN_DIAG_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
LV_CDN_DIAG_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
ERR_SYM_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the camshaft failure synchronization impossible					
ERR_SYM_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of the camshaft failure synchronization impossible					
LV_ERR_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present camshaft failure synchronization impossible					
LV_ERR_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present camshaft failure synchronization impossible					
LV_END_DIAG_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of camshaft signal diagnosis synchronization impossible					
LV_END_DIAG_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of camshaft signal diagnosis synchronization impossible					
CTR_CYC_ENG_PLAUS_CAM_IN[NC_NR_CAM_CBK]	V	0...FFH	0...255	1	[-]
Engine cycle counter for camshaft signal plausibility diagnosis					
CTR_CYC_ENG_PLAUS_CAM_EX[NC_NR_CAM_CBK]	V	0...FFH	0...255	1	[-]
Engine cycle counter for camshaft signal plausibility diagnosis					
LV_CDN_DIAG_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
LV_CDN_DIAG_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]

Diagnosis condition					
ERR_SYM_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symtoms of the intake camshaft diagnosis for crankshaft synchronization					
ERR_SYM_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symtoms of the exhaust camshaft diagnosis for crankshaft synchronization					
LV_ERR_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present intake camshaft failure for crankshaft synchronization					
LV_ERR_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V/O/S	0H 1H	PASSIVE ACTIVE	1	[-]
Present exhaust camshaft failure for crankshaft synchronization					
LV_END_DIAG_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of intake camshaft diagnosis for crankshaft synchronization					
LV_END_DIAG_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	V	0H 1H	PASSIVE ACTIVE	1	[-]
Result available of exhaust camshaft diagnosis for crankshaft synchronization					
CTR_ORNG_CAM_SYN_CRK	V	0...FFH	0...255	1	[-]
Failure counter to detect camshaft error for crankshaft synchronization					

Input data:

CTR_EDGE_CAM_IN[NC_NR_CAM_CBK]	CTR_EDGE_CAM_EX[NC_NR_CAM_CBK]	LV_SYN_VLD	LV_ACT_CRK
LV_CRK_SYN	LV_SYN_CAM_IN[NC_NR_CAM_CBK]	LV_SYN_CAM_EX[NC_NR_CAM_CBK]	LV_IGK
LV_FIRST_VLD_TOOTH	LV_ORNG_CAM_SYN_CRK	LV_ACT_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]	LV_ACT_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]
LV_INH_DIAG_CRK_PLAUS	LV_INH_DIAG_PLAUS_CAM_IN[NC_NR_CAM_CBK]	LV_INH_DIAG_PLAUS_CAM_EX[NC_NR_CAM_CBK]	LV_INH_DIAG_ENSD
LV_AD_END_CAM_IN[NC_NR_CAM_CBK]	LV_AD_END_CAM_EX[NC_NR_CAM_CBK]	LV_CRK_MISS_RUN_ENG	LV_INH_DIAG_SYN_CRK_CAM_IN[NC_NR_CAM_CBK]
LV_INH_DIAG_SYN_CRK_CAM_EX[NC_NR_CAM_CBK]	CONF_CAM_VVT_EX		

A.22.1.1 Diagnosis of crankshaft sensor

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect a failure if synchronization on crankshaft signal cannot be achieved.

One of the available camshaft signals is selected for crankshaft synchronization by setting one of the flags LV_ACT_SYN_CRK_CAM_IN[i] in the engine position control function. The signal edge counter of that camshaft is observed while the crankshaft is not synchronized. If the count reaches a calibrated number, and the system is still not synchronized with the crankshaft signal, then crankshaft sensor plausibility error is set without further debouncing. If valid crankshaft teeth have already been detected, the symptom will be “implausible signal”, otherwise it will be “missing signal”.

i = 1...NC_NR_CAM_CBK

In case of CONF_CAM_VVT_EX = 0 the exhaust camshaft is not taken into account for the diagnosis.

Description:

Error-symptoms are defined to this diagnostic function as following:

```
┌─┬─┬─┬─┐
│ │ │ └─> Signal missing        (= SYM_0)
│ │ └───> Signal implausible    (= SYM_1)
│ └─────> (not used)            (= SYM_2)
└───────> (not used)            (= SYM_3)
```

Application conditions:

Initialisation:	LV_CDN_DIAG_CRK_PLAUS = 0 at reset
	LV_END_DIAG_CRK_PLAUS = 0 at reset
	ERR_SYM_CRK_PLAUS = NO_SYM at reset
	LV_ERR_CRK_PLAUS = 0 at reset
Recurrence:	every update of CTR_EDGE_CAM_IN(EX)[i]
	And 0 to 1 transition of LV_STOP_ENG
Activation:	LV_ACT_CRK = 1
	And LV_INH_DIAG_CRK_PLAUS = 0
	And LV_INH_DIAG_ENSD = 0

When the activation condition is fulfilled then LV_CDN_DIAG_CRK_PLAUS = 1.

Deactivation:

When the activation condition is not fulfilled then LV_CDN_DIAG_CRK_PLAUS = 0.

Formula Section

Symptoms calculation:

(1)If LV_STOP_ENG = 1 **and** LV_CRK_MISS_RUN_ENG = 1

then ERR_SYM_CRK_PLAUS = "SYM_0"
(Crankshaft signal missing but engine is still turning)

else

(2) IF CONF_CAM_VVT_EX = 1

then

(3) if LV_CRK_SYN = 0 **and** CTR_EDGE_CAM_IN(EX)[i] > C_NR_EDGE_CAM_IN(EX)_SYN_CRK_MAX

then (4) If LV_FIRST_VLD_TOOTH = 1

Then ERR_SYM_CRK_PLAUS = “SYM_1"
(Detection of symptom SIGNAL IMPLAUSIBLE)

Else ERR_SYM_CRK_PLAUS = “SYM_0"
(Detection of symptom SIGNAL MISSING)

(4) Endif

LV_ERR_CRK_PLAUS = 1

LV_END_DIAG_CRK_PLAUS = 1

Deliver the result to Error Management

(3) Endif

(2) Else

(5) if LV_CRK_SYN = 0 **AND** CTR_EDGE_CAM_IN[i] > C_NR_EDGE_CAM_IN_SYN_CRK_MAX**]**

then If LV_FIRST_VLD_TOOTH = 1

Then ERR_SYM_CRK_PLAUS = “SYM_1"
(Detection of symptom SIGNAL IMPLAUSIBLE)

Else ERR_SYM_CRK_PLAUS = “SYM_0"
(Detection of symptom SIGNAL MISSING)

Endif

LV_ERR_CRK_PLAUS = 1

LV_END_DIAG_CRK_PLAUS = 1

Deliver the result to Error Management

(5) Endif

(2)Endif

(1)Endif

LV_ERR_CRK_PLAUS and ERR_SYM_CRK_PLAUS are reset to initialization value at 0 to 1 transition of LV_IGK.

If LV_CRK_SYN = 1

Then LV_END_DIAG_CRK_PLAUS = 1

LV_ERR_CRK_PLAUS = 0

ERR_SYM_CRK_PLAUS = NO_SYM

Deliver the result to Error Management

endif

Configuration for diagnostic symptoms :

Diagnostic **Crankshaft sensor**	**Symptom description**	**Symptom**	**Filter type**
Crankshaft sensor	*SIGNAL MISSING*	*SYM_0*	*NO*
	SIGNAL IMPLAUSIBLE	*SYM_1*	
		SYM_2	
		SYM_3	

A.22.1.2 Camshaft sensor plausibility diagnosis

FUNCTION DESCRIPTION:

General information:

The functional camshaft diagnosis cannot detect a failure when a camshaft sensor delivers no signal at all. Missing signal will be detected if the camshaft edge counter is not incremented at least once per engine cycle.

i = 1...NC_NR_CAM_CBK

Description:

Error-symptoms are defined to this diagnostic function as following:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> Signal missing      (= SYM_0)
 │  │  └────> (not used)          (= SYM_1)
 │  └───────> (not used)          (= SYM_2)
 └──────────> (not used)          (= SYM_3)
```

Application conditions:

Initialisation: LV_CDN_DIAG_PLAUS_CAM_IN(EX)[i] = 0 at reset

LV_END_DIAG_PLAUS_CAM_IN(EX)[i] = 0 at reset

ERR_SYM_PLAUS_CAM_IN(EX)[i] = NO_SYM at reset

LV_ERR_PLAUS_CAM_IN(EX)[i] = 0 at reset

CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i] =

C_NR_CYC_ENG_PLAUS_CAM_MAX at reset

Recurrence: once per engine cycle (two crankshaft revolutions),
at any phasing i.e. on crankshaft revolution which was on-going
at engine synchronization

Activation: LV_CRK_SYN = 1

And [LV_INH_DIAG_PLAUS_CAM_IN(EX)[i] = 0
or LV_ERR_PLAUS_CAM_IN(EX)[i] = 1]

And LV_INH_DIAG_ENSD = 0

And C_NR_CYC_ENG_PLAUS_CAM_MAX <> 0

When the activation condition is fulfilled then LV_CDN_DIAG_PLAUS_CAM_IN(EX)[i] =1.

Deactivation:

When the activation cond. is not fulfilled then LV_CDN_DIAG_PLAUS_CAM_IN(EX)[i] = 0.

Formula Section

Symptoms calculation:

if CTR_EDGE_CAM_IN(EX)[i]$_n$ = CTR_EDGE_CAM_IN(EX)[i]$_{n-1}$

then **if** CONF_CAM_VVT_EX = 1 *(exhaust Cam sensor available)*

then ERR_SYM_PLAUS_CAM_IN(EX)[i] = “SYM_0" *(SIGNAL MISSING)*

increment CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i]

else ERR_SYM_PLAUS_CAM_IN[i] = “SYM_0" *(SIGNAL MISSING)*
increment CTR_CYC_ENG_PLAUS_CAM_IN[i]

endif

else ERR_SYM_PLAUS_CAM_IN(EX)[i] = NO_SYM

decrement CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i]

endif

CTR_EDGE_CAM_IN(EX)[i]$_{n-1}$ will be initialized to zero at engine stop detection.

if CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i] >=
2 * C_NR_CYC_ENG_PLAUS_CAM_MAX

then **if** CONF_CAM_VVT_EX = 1 *(exhaust cam sensor available)*

then LV_ERR_PLAUS_CAM_IN(EX)[i] = 1

LV_END_DIAG_PLAUS_CAM_IN(EX)[i] = 1

else LV_ERR_PLAUS_CAM_IN[i] = 1

LV_END_DIAG_PLAUS_CAM_IN[i] = 1
LV_END_DIAG_PLAUS_CAM_EX[i] = 1

endif

reset CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i]

(to C_NR_CYC_ENG_PLAUS_CAM_MAX)

Deliver the result to Error Management

else **if** CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i] =0

then LV_ERR_PLAUS_CAM_IN(EX)[i] = 0

LV_END_DIAG_PLAUS_CAM_IN(EX)[i] = 1

reset CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i]

(to C_NR_CYC_ENG_PLAUS_CAM_MAX)

Deliver the result to Error Management

endif

endif

LV_ERR_PLAUS_CAM_IN(EX)[i] , ERR_SYM_PLAUS_CAM_IN(EX)[i] and CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i] are reset to initialization value at 0 to 1 transition of LV_IGK.

if LV_SYN_CAM_IN(EX)[i] = 1

then ERR_SYM_PLAUS_CAM_IN(EX)[i] = NO_SYM

LV_ERR_PLAUS_CAM_IN(EX)[i] = 0

LV_END_DIAG_PLAUS_CAM_IN(EX)[i] = 1

reset CTR_CYC_ENG_PLAUS_CAM_IN(EX)[i]

(to C_NR_CYC_ENG_PLAUS_CAM_MAX)

Deliver the result to Error Management

endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Camshaft sensor plausibility diagnosis for intake	*SIGNAL MISSING*	*SYM_0*	*NO*
		SYM_1	
		SYM_2	
		SYM_3	
Camshaft sensor plausibility diagnosis for exhaust	*SIGNAL MISSING*	*SYM_0*	*NO*
		SYM_1	
		SYM_2	
		SYM_3	

A.22.1.3 Camshaft Sensor Diagnosis for Crankshaft Synchronization

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect a failure of the camshaft information for crankshaft synchronization.

Up to four camshaft signals may be available in the system (intake and/or exhaust on one or two cylinder banks). The Engine Position Control selects one camshaft for synchronization by activating the corresponding operating mode. The camshaft signal acquisition will indicate "camshaft signal out of range for crankshaft synchronization" by setting LV_ORNG_CAM_SYN_CRK, which has to be reset by the diagnosis after treatment. A counter is incremented at every recurrence if LV_ORNG_CAM_SYN_CRK is set. The error is detected when the counter reaches a calibrated maximum.

i = 1...NC_NR_CAM_CBK

Description:

Error-symptoms are defined to this diagnostic function as following:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │ │ │ └─> Camshaft signal  not valid for crankshaft synchronization     (= SYM_0)
 │ │ └───> (not used)                                                    (= SYM_1)
 │ └─────> (not used)                                                    (= SYM_2)
 └───────> (not used)                                                    (= SYM_3)
```

Application conditions:

Initialisation:	LV_CDN_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0 at reset
	LV_END_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0 at reset
	ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = NO_SYM at reset
	LV_ERR_SYN_CRK_CAM_IN(EX)[i] = 0 at reset
Recurrence:	every crankshaft reference gap and every camshaft edge
Activation:	LV_ACT_CRK = 1
	And LV_INH_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0
	And LV_INH_DIAG_ENSD = 0

When the activation condition is fulfilled then LV_CDN_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1

Deactivation:

When the activation condition is not fulfilled then LV_CDN_DIAG_SYN_CRK_CAM_IN(EX)[i] = 0

Formula Section

Symptoms calculation:

For LV_ACT_SYN_CRK_CAM_IN(EX)[i] = 1 (camshaft sensor currently selected for synchronization):

if LV_ORNG_CAM_SYN_CRK = 0
then ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = NO_SYM
else **if** CONF_CAM_VVT_EX = 1 *(exhaust cam sensor available)*
then ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = “SYM_0" *(SIGNAL INVALID)*
else ERR_SYM_SYN_CRK_CAM_IN[i] = “SYM_0" *(SIGNAL INVALID)*
endif
increment CTR_ORNG_CAM_SYN_CRK
endif

if CTR_ORNG_CAM_SYN_CRK > C_CTR_ORNG_CAM_SYN_CRK_MAX
then **if** CONF_CAM_ VVT_EX = 1 *(exhaust cam sensor available)*
then LV_ERR_SYN_CRK_CAM_IN(EX)[i] = 1

else LV_ERR_SYN_CRK_CAM_IN [i] = 1
LV_ERR_SYN_CRK_CAM_EX[i] = 0
endif
LV_END_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1
Deliver the result to Error Management
reset CTR_ORNG_CAM_SYN_CRK
endif

if LV_SYN_VLD = 1
then LV_END_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1
ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = NO_SYM
LV_ERR_SYN_CRK_CAM_IN(EX)[i] = 0
Deliver the result to Error Management
reset CTR_ORNG_CAM_SYN_CRK
endif

(1) If CONF_CAM_VVT_EX = 0

then

For LV_ACT_SYN_CRK_CAM_IN[i] = 0

else

For LV_ACT_SYN_CRK_CAM_IN(EX)[i] = 0

(camshaft sensor currently **not** selected for cam/crk synchronization):

if LV_SYN_CAM_IN(EX)[i] = 1

and LV_AD_END_CAM_IN(EX)[i] = 1

then ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = NO_SYM

if LC_CAM_ERR_ENA_VVT = 1

then LV_ERR_SYN_CRK_CAM_IN(EX)[i] = 0

LV_END_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1

Deliver the result to Error Management

if LV_ERR_PLAUS_CAM_IN(EX)[i] = 1

then ERR_SYM_SYN_CRK_CAM_IN(EX)[i] = "SYM_0" *(SIGNAL INVALID)*

LV_ERR_SYN_CRK_CAM_IN(EX)[i] = 1

LV_END_DIAG_SYN_CRK_CAM_IN(EX)[i] = 1

Deliver the result to Error Management

reset CTR_ORNG_CAM_SYN_CRK

LV_ERR_SYN_CRK_CAM_IN(EX)[i], ERR_SYM_SYN_CRK_CAM_IN(EX)[i] and CTR_ORNG_CAM_SYN_CRK are reset to initialization value at 0 to 1 transition of LV_IGK.

(1)**endif**

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Camshaft sensor diagnosis for crankshaft synchronization on intake	*SIGNAL INVALID*	*SYM_0*	*NO*
		SYM_1	
		SYM_2	
		SYM_3	
Camshaft sensor diagnosis for crankshaft synchronization on exhaust	*SIGNAL INVALID*	*SYM_0*	*NO*
		SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_NR_EDGE_CAM_IN_SYN_CRK_MAX	1	1...FFH	1...255	1	[-]
Number of intake camshaft signal edges to detect crankshaft plausibility error before synchronization					
C_NR_EDGE_CAM_EX_SYN_CRK_MAX	1	1...FFH	1...255	1	[-]
Number of exhaust camshaft signal edges to detect crankshaft plausibility error before synchronization					
C_NR_CYC_ENG_PLAUS_CAM_MAX	1	0...7FH	0...127	1	[-]
Number of engine cycles to detect camshaft plausibility error					
C_CTR_ORNG_CAM_SYN_CRK_MAX	1	0...FFH	0...255	1	[-]
Failure count to detect camshaft error for crankshaft synchronization					
LC_CAM_ERR_ENA_VVT	1	0...1H	0...1	1	[-]
Switch to enable camshaft error debouncing for VVT reactivation					

A.22.2 Crankshaft Sensor Circuit Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_CRK_OC	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis condition					
ERR_SYM_CRK_OC	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom of the crankshaft sensor failure open or short circuit					
LV_ERR_CRK_OC	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Crankshaft sensor failure open or short circuit present					
LV_END_DIAG_CRK_OC	V	0H 1H	PASSIVE ACTIVE	1	[-]
Diagnosis result available					

Input data:

LV_IGK	LV_INH_DIAG_CRK_OC		

FUNCTION DESCRIPTION:

General information:

This dummy file replaces the MCPS circuit diagnosis for systems with ACPS. A circuit diagnosis for ACPS is not considered neceessary, since with any of the three lines open or short-circuited, the signal will be constantly high or low. The failure will then be detected by the functional diagnosis.

Application conditions:

Activation: LV_IGK = 1
And LV_INH_DIAG_CRK_OC = 0

When the activation condition is fulfilled then LV_CDN_DIAG_CRK_OC = 1.

Desactivation:

When the activation condition is not fulfilled then LV_CDN_DIAG_CRK_OC = 0.

Formula section:

ERR_SYM_CRK_OC = 0

LV_ERR_CRK_OC = 0

LV_END_DIAG_CRK_OC = 1

A.22.3 Crankshaft Sensor Diagnosis (Appl. Inc.)

FUNCTION DESCRIPTION:

This is the definition of DTC (P-codes) for the different diagnostic functions of the crankshaft signal.

Input data:

LV_ERR_CRK_SYN	LV_ERR_CRK_TOOTH	LV_ERR_CRK_PLAUS	LV_ERR_CRK_TOOTH_PER

Application condition :

Recurrence: At the update of each one of the crankshaft sensor diagnostic flags

*** Application Assistance:**

Limp home functions:
- KNK limp home value calculation
- No preinjection performed
- Engine speed limitation to C_N_MAX_CRK_LIH (only if CRK limp-home is active)-

The following control functions are inhibited:
- Knock control
- Anti-jerk function
- Engine overheating protection at VS = 0
- Gear ratio calculation
- Downstream fuel trim regulation

The following adaptation functions are inhibited:
- Knock adaptation
- Catalyst heating adaptation (indirect)
- Segment time adaptation

The following diagnosis functions are inhibited:
- Camshaft sensor diagnosis
- Knock sensor diagnosis
- O2-sensor downstream signal voltage diagnosis (only SCG)
- Misfire detection (ER-calculation)
- O2-sensor upstream switching time diagnosis
- O2-sensor upstream frequency diagnosis
- Catalyst efficiency diagnosis
- O2-sensor downstream switching time diagnosis
- O2-sensor downstream OBDII puc-end diagnosis
- Crankshaft oscillation detection

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Crankshaft sensor synchronisation diagnosis	Loss of CRK synchronisation	SYM_0	*MEM*
		SYM_1	
		SYM_2	
CRK_SYN		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
SEG_AD_MMV_ER_1_ENVD
SEG_AD_MMV_ER_3_ENVD
DIAG_INST_ENVD

Crankshaft sensor tooth number diagnosis	Additional or missing teeth	SYM_0	*MEM*
		SYM_1	
		SYM_2	
CRK_TOOTH		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
SEG_AD_MMV_ER_1_ENVD
SEG_AD_MMV_ER_3_ENVD
DIAG_INST_ENVD

Crankshaft sensor tooth period diagnosis	Tooth period implausible	SYM_0	*MEM*
		SYM_1	
		SYM_2	
CRK_TOOTH_PER		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
SEG_AD_MMV_ER_1_ENVD
SEG_AD_MMV_ER_3_ENVD
DIAG_INST_ENVD

Crankshaft sensor plausibility diagnosis	Signal missing	SYM_0	*STD*
	Signal implausible	SYM_1	
		SYM_2	
CRK_PLAUS		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
SEG_AD_MMV_ER_1_ENVD
SEG_AD_MMV_ER_3_ENVD
DIAG_INST_ENVD

A.23 Extended Diagnosis for MWSS

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MWSS	O/V	0...1H	0..1	1	-
logical variable for wheelspeed sensor failure after debouncing					
LV_ERR_PRES_MWSS	O/V	0...1H	0..1	1	-
logical variable for wheelspeed sensor failure currently present before debouncing					
V_DIF_MWSS_DIAG	O/V	0...FFH	0..5	19.6e-3	V
difference between max and min voltage of a sample periode					
V_DIF_MAX_MWSS_DIAG	O/V/S	0...FFH	0..5	19.6e-3	V
maximum difference between max and min voltage of a sample periode					
V_MIN_MWSS_DIAG	V	0...FFH	0..5	19.6e-3	V
minimum voltage of a sample periode					
V_MAX_MWSS_DIAG	V	0...FFH	0..5	19.6e-3	V
maximum voltage of a sample periode					
NR_MWSS_DIAG_SAMPLE	V	0...FFH	0...255	1	-
Current value of sample counter					

Input data:

MWSS_DIAG_BAS	LV_CDN_VB_OBD1	VS	CONF_VS

FUNCTION DESCRIPTION:

General information:

With short circuit or open load of one of the sensor lines of a magnetic wheel speed sensor, the input circuitry might still produce an output signal which may trigger the rough road detection algorithm and thus inhibit misfire detection. The purpose of this function is to detect such failures by monitoring the bias node voltage of the input circuitry.

Under normal conditions, there is a DC voltage at the bias node voltage with relatively large tolerances which has a relatively small AC voltage superimposed due to unsymmetric parasitic capacities of the wiring harness.

If the input circuit toggles with a faulty sensor, the superimposed AC voltage will be larger than the noise at o.k. system. This is used as a signal for sensor failure. The hardware should be designed in a way to have a sufficient margin for the detection threshold.

At zero speed it is checked if the limits of the DC voltage for normal operation are exceeded. This allows to detect a short circuit to ground or to battery before the vehicle begins to move.

Description:

The voltage MWSS_DIAG_BAS is sampled by an analog input of the CPU or via analog-multiplexer at a rate which may be inferior to the signal frequency, e.g. every 1 ms to 10 ms. The AC amplitude is then calculated as the difference between maximum and minimum value of a number of samples.

As soon as C_NR_MWSS_DIAG_SAMPLE samples are completed, the maximum and the minimum voltage of this sampling period is available in V_MAX_MWSS_DIAG and V_MIN_MWSS_DIAG. The difference of both is V_DIF_MWSS_DIAG. It equals approximately the peak - to peak - voltage of the sampling periode.

Depending on the vehicle speed at the end of the sampling periode, the values of V_MAX_MWSS_DIAG, V_MIN_MWSS_DIAG or V_DIF_MWSS_DIAG are compared to the respective thresholds to set an error bit and start the debouncing algorithm.

Application conditions:

Activation: The function is active only if CONF_VS = "WHEEL" and **LV_CDN_VB_OBD1 = 1.**

Conditions to detect a failure:

If VS > 0
    and V_DIF_MWSS_DIAG > C_V_THD_MWSS
then LV_ERR_PRES_MWSS = 1
the debounce counter is incremented with C_ABC_INC_MWSS
else if VS = 0
    and (V_MIN_MWSS_DIAG < C_V_THD_MIN_MWSS
or V_MAX_MWSS_DIAG > C_V_THD_MAX_MWSS)
then LV_ERR_PRES_MWSS = 1
the debounce counter is incremented with C_ABC_INC_MWSS
else LV_ERR_PRES_MWSS = 0
    if LV_ERR_MWSS = 0
    then *the debounce counter is decremented*
    endif
endif

If the debounce counter has reached C_ABC_MAX_MWSS
then LV_ERR_MWSS = 1
endif

Error Healing:

If LV_ERR_MWSS = 1
and VS > C_VS_MIN_MWSS_CNL
and LV_ERR_PRES_MWSS = 0
then a delay time C_T_MWSS_MIN_CNL is running

If the delay time C_T_MWSS_MIN_CNL has run out
then LV_ERR_MWSS = 0 *(the debounce counter is set to zero)*

Remark: The time delay C_T_MWSS_MIN_CNL is frozen when one of the conditions mentioned above is currently not present (exceptLV_ERR_PRES_MWSS=1). When all conditions are present again, the counter continues. The time delay starts a new when LV_ERR_PRES_MWSS = 1 is detected.

Emergency operation :

* Limp Home : **none**

* Application Incidence : **none**

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_THD_MWSS	1	0...FFH	0..5	19.6e-3	V
threshold for peak-to-peak voltage to detect a failure at moving vehicle					
C_V_THD_MIN_MWSS	1	0...FFH	0..5	19.6e-3	V
threshold minimum bias node voltage to detect a failure at stopped vehicle					
C_V_THD_MAX_MWSS	1	0...FFH	0..5	19.6e-3	V
threshold maximum bias node voltage to detect a failure at stopped vehicle					
C_NR_MWSS_SAMPLE	1	0...FFH	0...255	1	-
number of samples per sample period					
C_ABC_INC_MWSS	1	0...FFH	0...255	1	-
Increment of anti-bounce-counter					
C_ABC_MAX_MWSS	1	1...FFH	1...255	1	-
Maximum of anti-bounce-counter					
C_VS_MIN_MWSS_CNL	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for healing of MWSS error					
C_T_MWSS_MIN_CNL	1	1...FFFFH	0.1...655.35	0.01	sec
Time delay to heal MWSS error					

Note: For calibration and validation of the thresholds, the maximum value for the peak-to-peak-voltage V_DIF_MWSS_DIAG is stored in the EEPROM with the name V_DIF_MAX_MWSS_DIAG. It will be reset only when the error memory is erased.

Application hint:

C_NR_MWSS_SAMPLE should be set to 20 minimum for a stable V_DIF_MWSS_DIAG.
The delay C_T_MWSS_MIN_CNL must be longer than the maximum time necessary to complete a sample periode.
Set C_VS_MIN_MWSS_CNL to the highest speed that is reached regularly for at least the duration of C_T_MWSS_MIN_CNL.

A.23.1 Application assistances for the extended MWSS diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Magnetic Wheel Speed Sensor MWSS	Open wire	SYM_0	*STD*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
VS
V_MIN_MWSS_DIAG
V_MAX_MWSS_DIAG

Formula section:

Emergency operation :

* Application Assistances:

- The following Control Functions are inhibited:
  - Anti-Jerk Function
  - Rough Road Detection
  - Crankshaft Oscillation Detection
  - Gear Ratio Calculation
  - Cooling Fan Management depending on VS
  - Engine Speed Increasing at Idle depending on VS > 0
  - Engine Overheating Protection at VS = 0
- The following Adaptation Functions are inhibited:
  - Idle Speed Adaptation (only if additionally an CAN_TCU-Error is detected)
- The following Diagnosis Functions are inhibited:
  - Differential Tank Pressure Sensor Plausibility Diagnosis
  - ISA Plausibility Diagnosis
  - Evaporative System Diagnosis
  - O2-Sensor upstream Switching Time Diagnosis
  - O2-Sensor upstream Frequency Diagnosis
  - O2-Sensor downstream Switching Time Diagnosis
  - O2-Sensor downstream OBDII puc-end Diagnosis
  - Catalyst Efficiency Diagnosis
- Engine Speed Limitation is set to C_N_MAX_VS_DIAG
- Idle Speed Regulation Limits Independent on VS
- TPS Adaptation Limitation Activated
- Idle speed independent from VS

A.24 Engine roughness diagnosis

A.24.1 Engine roughness segment time acquisition error

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_T_SEG_ER	V	0...1H	0...1	1	[-]
Diagnosis condition of segment time adaptive values					
ERR_SYM_T_SEG_ER	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptoms of segment time adaptive values					
LV_ERR_T_SEG_ER	V/O	0...1H	0...1	1	[-]
Present failure for segment time adaptive values					
LV_END_DIAG_T_SEG_ER	V	0...1H	0...1	1	[-]
Diagnosis end of segment time adaptive values					

Input data:

LV_DC	LV_INH_MIS_CRK	LV_INH_DIAG_T_SEG_ER	

FUNCTION DESCRIPTION:

The purpose of this error is to diagnose synchronisation errors on engine roughness segments measurement.
An error symptom "*Missing/Adding 1 tooth or more on engine roughness segment acquisition*" is detected when a crankshaft synchronisation error caused by 1 or 2 missing/additional tooth occurs on one crankshaft revolution.

Application conditions:

Initialisation: *at ECU reset or KEY OFF/KEY ON event*

LV_CDN_DIAG_T_SEG_ER = 0

ERR_SYM_T_SEG_ER = *Refer to filtering configuration for the initialisation value*

LV_ERR_T_SEG_ER = *Refer to filtering configuration for the initialisation value*

LV_END_DIAG_T_SEG_ER = *Refer to filtering configuration for the initialisation value*

Recurrence: each 360°Crk at segment task

Activation/Deactivation:

If LV_DC = 1

And LV_INH_DIAG_T_SEG_ER = 0

Then LV_CDN_DIAG_T_SEG_ER = 1 *diagnosis is active*

Else LV_CDN_DIAG_T_SEG_ER = 0 *diagnosis is passive*

EndIf

Formula section:

Error detection

If LV_INH_MIS_CRK = 1

Then ERR_SYM_T_SEG_ER = SYM_1

(symptom "Missing/Adding 1 or 2 teeth on engine roughness segment acquisition" is active, ABC counter starts to increment)

LV_ERR_T_SEG_ER = 1 *(after anti-bounce)*

Else ERR_SYM_T_SEG_ER = NO_SYM

(no symptom is declared, ABC counter starts to decrement)

LV_ERR_T_SEG_ER = 0 *(after anti-bounce)*

EndIf

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_T_SEG_ER	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_T_SEG_ER	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					

Configuration for diagnostic symptoms

Diagnosis T_SEG_ER	Symptom	Nr	ABC type
Engine roughness segment time acquisition	-	*0*	STD_INI
	Missing/Adding 1 tooth or more on engine roughness segment acquisition	*1*	
		2	
T_SEG_ER		*3*	

A.24.2 Engine roughness segment adaptive values out of range error

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CDN_DIAG_SEG_AD_ER	V	0...1H	0...1	1	[-]
Diagnosis condition of ER segment time adaptive values					
ERR_SYM_SEG_AD_ER	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	0	[-]
Symptoms of ER segment time adaptive values					
LV_ERR_SEG_AD_ER	V/O	0...1H	0...1	1	[-]
Present failure for ER segment time adaptive values					
LV_END_DIAG_SEG_AD_ER	V	0...1H	0...1	1	[-]
Diagnosis end of ER segment time adaptive values					

Input data:

LV_SEG_AD_ER	LV_SEG_AD_LIM_ER	LV_INH_DIAG_SEG_AD_ER	LV_SEG_AD_AVL_ER

Description:

The purpose is to diagnose the ER segment adaptive values.

If during or at the end of the segment adaptive values learning process *(LV_SEG_AD_ER = 1 or LV_SEG_AD_ER = 1 –> 0)*, at least one of the engine roughness segment adaptation values is at the limit (LV_SEG_AD_LIM_ER = 1), the error symptom SEG_AD_ER "engine roughness segment adaptation values at the limit" is detected.

Application conditions:

Initialisation: *at ECU reset or KEY OFF/KEY ON event*

LV_CDN_DIAG_SEG_AD_ER = 0

ERR_SYM_SEG_AD_ER = *Refer to filtering configuration for the initialisation value*

LV_ERR_SEG_AD_ER = *Refer to filtering configuration for the initialisation value*

LV_END_DIAG_SEG_AD_ER = *Refer to filtering configuration for the initialisation value*

Recurrence: segment task

Activation/Deactivaton:

If **[(** LV_SEG_AD_ER = 1 -> 0 **And** LC_CDN_DIAG_SEG_AD_ER = 0 **)**
Or **(** LV_SEG_AD_ER = 1 **And** LC_CDN_DIAG_SEG_AD_ER = 1 **)]**
And LV_SEG_AD_AVL_ER = 1 *// adaptive values available*
And LV_INH_DIAG_SEG_AD_ER = 0 *// no application fade-out*
Then LV_CDN_DIAG_SEG_AD_ER = 1 *// diagnosis is active*
Else LV_CDN_DIAG_SEG_AD_ER = 0 *// diagnosis is passive*
EndIf

Formula section:

Error detection

If LV_SEG_AD_LIM_ER = 1 *// adaptive values at the limit*
Then ERR_SYM_SEG_AD_ER = SYM_1
(symptom "ER segment time adaptive values at the limit" is active, ABC counter starts to increment)
LV_ERR_SEG_AD_ER = 1 *(after anti-bounce)*
Else ERR_SYM_SEG_AD_ER = NO_SYM
(no symptom is declared, ABC counter starts to decrement)
LV_ERR_SEG_AD_ER = 0 *(after anti-bounce)*
EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_SEG_AD_ER	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_SEG_AD_ER	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					
LC_CDN_DIAG_SEG_AD_ER	1	0...1H	0...1	1	[-]
Diagnosis condition mode of ER segment time adaptive values					

Configuration for diagnostic symptoms:

Diagnosis SEG_AD_ER	Symptom	Nr	ABC type
ER segment adaptation values at the limit	-	*0*	STD_INI
	ER segment adaptation values at the limit	*1*	
		2	
SEG_AD_ER		*3*	

A.24.3 Engine roughness diagnosis - Application Incidences

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_INH_DIAG_T_SEG_ER	V/O	0...1H	0...1	1	[-]
Specific conditions to inhibit Engine roughness segment time acquisition diagnosis					
LV_INH_DIAG_SEG_AD_ER	V/O	0...1H	0...1	1	[-]
Specific conditions to inhabit Engine roughness segment adaptive values diagnosis					

Input data:

LV_DC	LV_LIH_ERR_CRK		

FUNCTION DESCRIPTION:

General information:

There are no specific conditions to inhabit:

- Engine roughness segment time acquisition diagnosis
- Engine roughness segment adaptive values diagnosis

Application conditions:

Initialisation: *at ECU reset or KEY OFF/KEY ON event*

LV_INH_DIAG_T_SEG_ER = 0

LV_INH_DIAG_SEG_AD_ER = 0

Recurrence: *Every 360°Crk*

Activation: *LV_DC = 1*

Formula section:

If LV_LIH_ERR_CRK = 1

Then LV_INH_DIAG_T_SEG_ER = 1

LV_INH_DIAG_SEG_AD_ER = 1

Else LV_INH_DIAG_T_SEG_ER = 0

LV_INH_DIAG_SEG_AD_ER = 0

Endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
ER segment time acquisition		SYM_0	*STD*
	Missig/ Adding 1 tooth or nore on engine roughness segment acquisition	SYM_1	
		SYM_2	
T_SEG_ER		SYM_3	

List of Environmental Data to store in failure memory: ENG_STATE
SEG_AD_MMV_ER_1_ENVD
SEG_AD_MMV_ER_3_ENVD
DIAG_INST_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
ER segment adaptation values at the limit		SYM_0	*STD*
	ER segment adaption values at the limit	SYM_1	
		SYM_2	
SEG_AD_ER		SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
ENG_STATE
SEG_AD_MMV_ER_1_ENVD
TOIL

A.25 Vehicle Speed Signal Diagnosis (VS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VS	V/S/O	0...1H	0...1	1	-
Boolean for error currently present on vehicle speed signal acquisition.					
T_VS_MIN_DIAG	V/O	0…FFFFH	0 …655.35	0.01	S
Timer for error symptom VS plausibility present					

Input data:

MAF	LV_CDN_VB_OBD1	N	N_MAX_THD
VS		LV_IGK	N_32
TCO	LV_IND_FCUT	LV_AT	LV_ERR_MAF
			LV_ERR_MAP

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the frequent input signal (VS) on the microcontroller.
The application recurrence is **10 msec**.
The diagnosis takes place only if **LV_CDN_VB_OBD1 = 1.**

Application conditions:

Detection:

* Plausibility Check

If LV_IGK **=** 1 (Active)
and VS **=** 0
and N >= C_N_VS_MIN_DIAG(_AT)
and N < N_MAX_THD – C_N_VS_MAX_HYS_DIAG
and MAF >= C_MAF_VS_MIN_DIAG(_AT)
and LV_IND_FCUT **=** 0
and TCO > C_TCO_VS_MIN_DIAG
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
then the time counter T_VS_MIN_DIAG is incremented

If the time counter C_T_VS_MIN_DIAG(_AT) seconds has run out
then ERR_SYM_VS = 1H (SYM_1) 'Invalid Signal' is detected
and LV_ERR_VS **=** 1

Remark: The time delay C_T_VS_MIN_DIAG(_AT) is frozen when one of the above-mentioned diagnosis conditions is currently not present (except VS = 0).
When all conditions are present again, the counter continues.
As soon as VS > 0 is detected the time counter is reset immediately.

Error healing:

If LV_IGK = 1
and VS > 0 **for more than** C_T_VS_MIN_DIAG(_AT) seconds

then LV_ERR_VS = 0

Emergency operation:

* Limp Home: no limp home

Formula section:

* Error Signalling :

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TCO_VS_MIN_DIAG	-	0...FEH	-48...142.5	0.744	°C
Minimum coolant temperature to perform vehicle speed diagnosis.					
C_N_VS_MIN_DIAG(_AT)	1	0...FFH	0...8160	32	rpm
Engine speed threshold for implausible signal condition detected on VS signal. Depending on LV_AT					
C_N_VS_MAX_HYS_DIAG	1	0...FFFFH	0...8160	1	Rpm
Hysteresis of maximum engine speed for VS diagnosis.					
C_MAF_VS_MIN_DIAG(_AT)	1	0...FFH	0...1389	5.45	mg/TDC
Mass air flow threshold for implausible signal condition detected on VS signal. Depending on LV_AT					
C_T_VS_MIN_DIAG(_AT)	1	0...FFFFH	0...655.35	0.01	sec.
Time delay to detect implausible signal on VS signal. Depending on LV_AT					

A.25.1 Application assistances

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Vehicle Speed Sensor Range / Performance		SYM_0	*STD*
	Plausibility fault	SYM_1	
		SYM_2	
VS		SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
VS
V_MIN_MWSS_DIAG
V_MAX_MWSS_DIAG

- **Application Assistances:**
- The following Control Functions are inhibited:
  - Anti-Jerk Function (indirectly)
  - Rough Road Detection (indirectly)
  - Crankshaft Oscillation Detection (indirectly)
  - Gear Ratio Calculation
  - Cooling Fan Management depending on VS
  - Engine Speed Increasing at Idle depending on VS > 0 (indirectly)
  - Engine Overheating Protection at VS = 0
- The following Adaptation Functions are inhibited:
  - Idle Speed Adaptation (only if additionally an CAN_TCU-Error is detected)
- The following Diagnosis Functions are inhibited:
  - Differential Tank Pressure Sensor Plausibility Diagnosis (Noisy Signal)
  - Differential Tank Pressure Sensor Plausibility Diagnosis (Stuck Signal)
  - ISA Plausibility Diagnosis
  - Evaporative System Diagnosis
  - O2-Sensor upstream Switching Time Diagnosis
  - O2-Sensor upstream Frequency Diagnosis
  - O2-Sensor downstream Switching Time Diagnosis
  - O2-Sensor downstream OBDII puc-end Diagnosis
  - Catalyst Efficiency Diagnosis
- Engine Speed Limitation is set to C_N_MAX_VS_ERR
- Idle Speed Regulation Limits Independent on VS
- TPS Adaptation Limitation Activated
- idle speed independent from VS (indirect)

A.26 TIA Sensor(s) Diagnosis: Application Incidences

A.26.1 Application Incidences:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TIA_SUB[NC_SENS_NR_TIA]	O/V	0H...FEH	-48...142.5	0.75	°C
Substitute value for TIA: one value for each TIA sensor of AIRT					
LV_INH_DIAG_INTM_TIA[NC_SENS_NR_TIA]	O/V	0H...1H	0...1	1	-
"1" indicates inhibition of the TIA intermittent Diagnosis : one value for each TIA sensor of AIRT					
LV_INH_DIAG_DYN_TIA[NC_SENS_NR_TIA]	O/V	0H...1H	0...1	1	-
"1" indicates inhibition of the TIA Stuck Signal Diagnosis : one value for each TIA sensor of AIRT					
LV_INH_DIAG_EL_TIA[NC_SENS_NR_TIA]	O/V	0H...1H	0...1	1	-
"1" indicates inhibition of the TIA Electrical Diagnosis: one value for each TIA sensor of AIRT					

Input data:

LV_ERR_TCO	NC_SENS_NR_TIA	C_TIA_SUB_TOL_DIAG	LV_IGK
T_AST	LV_ERR_VS	LV_ERR_MWSS	TIA_ST
LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAF	LV_ERR_MAP	

FUNCTION DESCRIPTION:

Each type of TIA diagnosis (ie: Electrical, Intermittent, Stuck Signal) can be inhibited by setting the relevant calibration byte to 1. The project can add additionnal conditions if desired.

Application Conditions:

Initialization: ***at*** *Reset* **Or** Transition LV_IGK = 0 to 1

For m = 1 **to** NC_SENS_NR_TIA

TIA_SUB[m] = C_TIA_SUB_TOL_DIAG

LV_INH_DIAG_EL_TIA[m] = 0

LV_INH_DIAG_INTM_TIA[m] = 0

LV_INH_DIAG_DYM_TIA[m] = 0

End for *m*

Reccurence: **100 ms**

Activation: at every Engine State

Deactivation: -

Formula section:

For m = 1 **to** NC_SENS_NR_TIA

Inhibition flags definition:

For electrical (EL)

```
If      LC_INH_DIAG_EL_TIA = 1
then    LV_INH_DIAG_EL_TIA[m] = 1
else    if      T_AST  <  C_T_DLY_DIAG_EL_TIA
        then    if      TIA_ST > C_TIA_ST_MIN_DIAG_EL_TIA
                then    LV INH_DIAG_EL_TIA[m] = 0
                else    LV INH_DIAG_EL_TIA[m] = 1
                endif
        else    LV_INH_DIAG_EL_TIA[m] = 0
        endif
endif
```

For intermittent (INTM)

```
If      LC_INH_DIAG_INTM_TIA = 1          or
        LV INH_DIAG_EL_TIA[m] = 1
then    LV_INH_DIAG_INTM_TIA[m] = 1
else    LV INH_DIAG_INTM_TIA[m] = 0
endif
```

For stuck (DYN)

```
If      LC_INH_DIAG_DYN_TIA = 1
   or   LV_ERR_TCO = 1
   or   LV_ERR_VS = 1        (because of DIST calculation)
   or   LV_ERR_MWSS = 1
   or   LV_ERR_LOAD_TPS_PLAUS = 1
   or   LV_ERR_MAF = 1
   or   LV_ERR_MAP = 1
then    LV_INH_DIAG_DYN_TIA[m] = 1
else    LV_INH_DIAG_DYN_TIA[m] = 0
endif
```

TIA_SUB[NC_SENS_NR_TIA] definition:

```
If    LV_ERR_TCO = 0
And         TCO < C_TCO_DIAG
Then        TIA_SUB[m] = C_TIA_SUB_BOL_DIAG
Else        TIA_SUB[m] = C_TIA_SUB_TOL_DIAG
```

End for m

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_EL_TIA	1	0H...FFH	0...255	1	-
Anti – bounce counter increment for TIA electrical failure diagnosis: one value for each TIA sensor of AIRT					
C_ABC_MAX_EL_TIA	1	1H...FFH	1...255	1	-
Maximum value of the anti - bounce for TIA electrical failure diagnosis: one value for each TIA sensor of AIRT					
C_ABC_INC_INTM_TIA	1	0H...FFH	0...255	1	-
Anti-bounce counter increment for TIA intermittent failure diagnosis					
C_ABC_MAX_INTM_TIA	1	1H...FFH	1...255	1	-
Maximum value for anti-bounce counter for TIA intermittent failure diagnosis					
C_ABC_INC_DYN_TIA	1	0H...FFH	0...255	1	-
Anti-bounce counter increment for TIA stuck failure diagnosis					
C_ABC_MAX_DYN_TIA	1	1H...FFH	1...255	1	-
Maximum value for anti-bounce counter for TIA stuck failure diagnosis					
LC_INH_DIAG_EL_TIA	1	0H...1H	0...1	1	-
If set to 1 then the TIA Electrical Diagnosis is inhibited: one value for each TIA sensor of AIRT					
LC_INH_DIAG_INTM_TIA	1	0H...1H	0...1	1	-
If set to 1 then the TIA intermittent Diagnosis is inhibited: one value for each TIA sensor of AIRT					
LC_INH_DIAG_DYN_TIA	1	0H...1H	0...1	1	-
If set to 1 then the TIA stuck signal Diagnosis is inhibited: one value for each TIA sensor of AIRT					
C_TCO_DIAG	1	0H...FEH	-48...142.5	0.75	°C
TCO threshold of TIA_SUB calculation					
C_TIA_SUB_BOL_DIAG	1	0H...FEH	-48...142.5	0.75	°C
Low Limp Home value for TIA failure detected: one value for each TIA sensor of AIRT					
C_TIA_SUB_TOL_DIAG	1	0H...FEH	-48...142.5	0.75	°C
High Limp Home value for TIA failure detected: one value for each TIA sensor of AIRT					
C_T_DLY_DIAG_EL_TIA	1	0...FFFFH	0...6553.5	0.1	s
Delay time for diagnosis activation					
C_TIA_ST_MIN_DIAG_EL_TIA	1	0H...FEH	-48...142.5	0.75	°C
Minimum TIA_ST to activate electrical TIA diagnosis					

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Intake Air temperature Sensor electrical diagnosis	Short circuit to battery or Open circuit	SYM_0	*STD_INI*
	Short circuit to GND	SYM_1	
		SYM_2	
EL_TIA		SYM_3	

List of Environmental Data to store in failure memory: T_AST_ENVD

TCO_ST

TIA_MES_ST

TIA_MES

Diagnostic	Symptom Description	Symptom	Filter type
Intake Air temperature Sensor intermittent diagnosis	signal intermittent	SYM_0	*MEM*
		SYM_1	
		SYM_2	
INTM_TIA		SYM_3	

List of Environmental Data to store in failure memory: TIA_MES

TIA_MES_DIF_DIAG_INTM

TIA_MES_DIF_DIAG_DYN

DIAG_INST_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Intake Air temperature Sensor	signal stuck	SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
DYN_TIA		SYM_3	

List of Environmental Data to store in failure memory: TIA_MES

TIA_MES_DIF_DIAG_INTM

TIA_MES_DIF_DIAG_DYN

DIAG_INST_ENVD

A.26.2 Application assistance:

- TCO Limp Home Calculation modified
- Coolant Temperature initialization modified (only if additionally TCO-Error detected)
- Evaporative Emission Control - Activation of MIN_PURGE instead of MAX_PURGE
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation
  - Knock Adaptation
  - Idle Speed Adaptation
- The following Diagnosis Functions are inhibited:
  - Differential tank pressure sensor plausibility diagnosis (noisy signal)
  - Differential tank pressure sensor plausibility diagnosis (signal constant)
  - O2-Sensor downstream signal voltage diagnosis (only SCG)
  - MAP/TPS Plausibility Diagnosis
  - Evaporative System Diagnosis
  - Fuel System Diagnosis
  - Thermostat Diagnosis

A.26.3 RBM interface for the "TIA_DYN" diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
STATE_RBM_TIA_DYN	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor of TIA Signal Stuck diagnosis for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TIA_DYN	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of TIA Signal Stuck diagnosis, for application purposes					

Input data:

LV_END_DIAG_DYN_TIA[NC_SENS_NR_TIA]	CTR_COMP_RBM_TIA_DYN	CTR_CDN_RBM_TIA_DYN	LV_DC
LV_ERR_TCO_EL	LV_ERR_TCO_PLAUS	LV_ERR_TCO_GRD	
CTR_ERR_DYN_NR	LV_ERR_TCO_STUCK_H	LV_ERR_TCO_STUCK	
LV_ERR_VS	LV_ERR_MWSS		

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TIA_DYN monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TIA_DYN data.

Within STATE_RBM_TIA_DYN, three different informations are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)

  (no intrusive operation, no short trip)

  (general information depending on monitor could be added)
- Monitor disabled because of system malfunction (bit 1)

  (depending on failure status: pending)

  (general information depending on monitor could be added)
- Monitor individual RBM conditions encountered within this DC (bit 2)

  (general information depending on monitor could be added)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_TIA_DYN = 0

at LV_DC 0 → 1 transition :

bit 0, bit 1 and bit 2 of STATE_RBM_TIA_DYN = 0

on failure memory reset :

bit 1 of STATE_RBM_TIA_DYN = 0

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_TCO_EL, LV_ERR_TCO_PLAUS, LV_ERR_TCO_GRD, LV_ERR_TCO_STUCK, LV_ERR_TCO_STUCK_H, LV_ERR_VS, LV_ERR_MWSS

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TIA_DYN = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TIA_DYN = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TIA_DYN = 0

Then

If LV_END_DIAG_TIA_DYN = 1

Then

bit 0 of STATE_RBM_TIA_DYN = 1

Endif

Endif

If bit 1 of STATE_RBM_TIA_DYN = 0

Then

If LV_ERR_TCO_EL = 1 **or**

LV_ERR_TCO_GRD = 1 **or**

LV_ERR_TCO_PLAUS = 1 **or**

LV_ERR_TCO_STUCK = 1 **or**

LV_ERR_TCO_STUCK_H = 1 **or**

LV_ERR_VS = 1 **or**

LV_ERR_MWSS = 1 **or**

Then

bit 1 of STATE_RBM_TIA_DYN = 1

Endif

Endif

No specific condition for TOIL_STUCK :

bit 2 of STATE_RBM_TIA_DYN always equal to 1

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TIA_DYN = 0
then RATE_RBM_TIA_DYN = FFFFH
else RATE_RBM_TIA_DYN
= (CTR_COMP_RBM_TIA_DYN / CTR_CDN_RBM_TIA_DYN)
endif

A.27 Electrical Failure Monitoring:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_EL_TIA[NC_SENS_NR_TIA]	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of TIA electrical detected failure: one value for each TIA sensor of AIRT					
LV_ERR_EL_TIA[NC_SENS_NR_TIA]	O/V	0...1H	0...1	1	-
Present TIA electrical failure: one value for each TIA sensor of AIRT					
LV_END_DIAG_EL_TIA[NC_SENS_NR_TIA]	O/V	0...1H	0...1	1	-
Result of TIA electrical diagnosis: one value for each TIA sensor of AIRT					
LV_CDN_DIAG_EL_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
Conditions for TIA electrical diagnosis: 0 = not fulfilled, 1 = fulfilled: one value for each TIA sensor of AIRT					

Input data:

VP_TIA[NC_SENS_NR_TIA]	LV_INH_DIAG_EL_TIA[NC_SENS_NR_TIA]	LV_CDN_VB_MIN_DIAG	LV_IGK
NC_SENS_NR_TIA	T_AST	MAF_THR	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_THR_MAX_DIAG_EL	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Minimum mass air flow value for Short Cut to Plus or Open Circuit detection condition					
C_T_AST_MIN_DIAG_EL	1	0...FFFFH	0...6.5535E+3	0.1	s
Minimum time after start value for Short Cut to Plus or Open Circuit detection condition					
C_VP_TIA_MAX_DIAG_EL[NC_SENS_NR_TIA]	1	0...7FFFH	0...4.999847	1.52588E-4	V
VP_TIA voltage threshold value for Short Cut to Plus or Open Circuit: one value for each TIA sensor of AIRT					
C_VP_TIA_MIN_DIAG_EL[NC_SENS_NR_TIA]	1	0...7FFFH	0...4.999847	1.52588E-4	V
VP_TIA voltage threshold value for Short Cut to Ground: one value for each TIA sensor of AIRT					

A.27.1 General:

NC_AIRT_TIA_SENS has been defined depending on the TIA hardware definition (NC_TIA_CONF value).

The **Electrical Diagnosis** has to be performed for the NC_SENS_NR_TIA sensors.

(See “Loop For m = 1 to NC_SENS_NR_TIA” below).

Initialization of ERR_SYM_EL_TIA, LV_CDN_DIAG_EL_TIA, LV_END_DIAG_EL_TIA, LV_ERR_EL_TIA depends on the type of anti bounce chosen by the project.

General information:

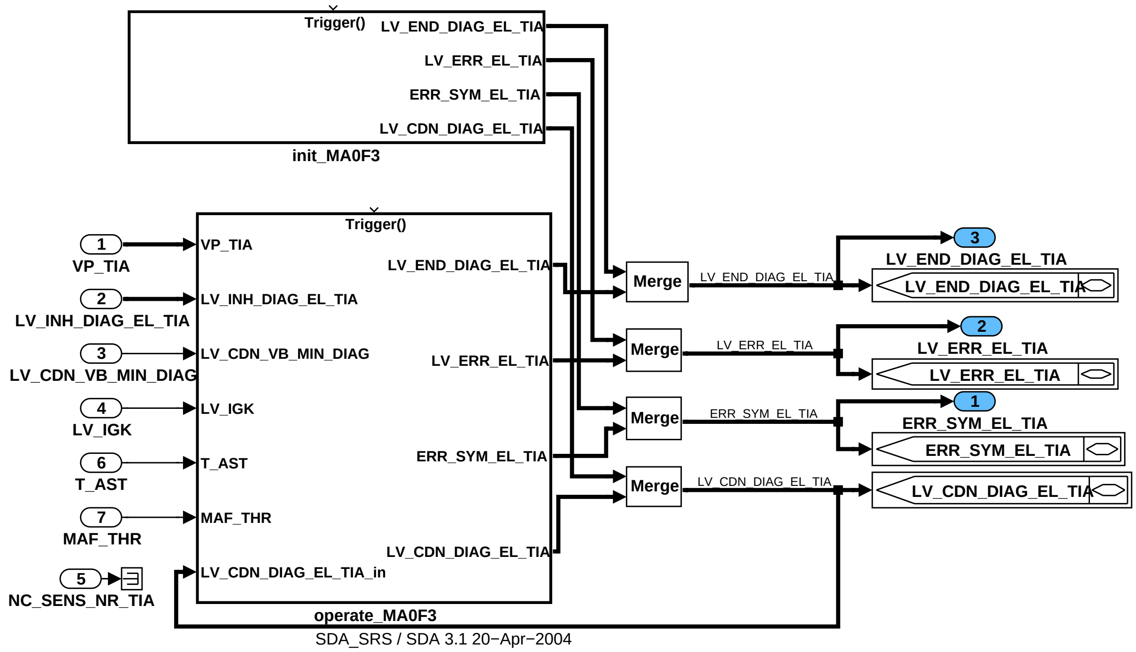
The purpose is to diagnose the voltage value V_TIA from the air temperature sensor(s) and to detect an electrical failure. 2 symptoms are distinguished:

SCP (Short Cut to Plus),

SCG (Short Cut to Ground) or OC (Open Circuit).

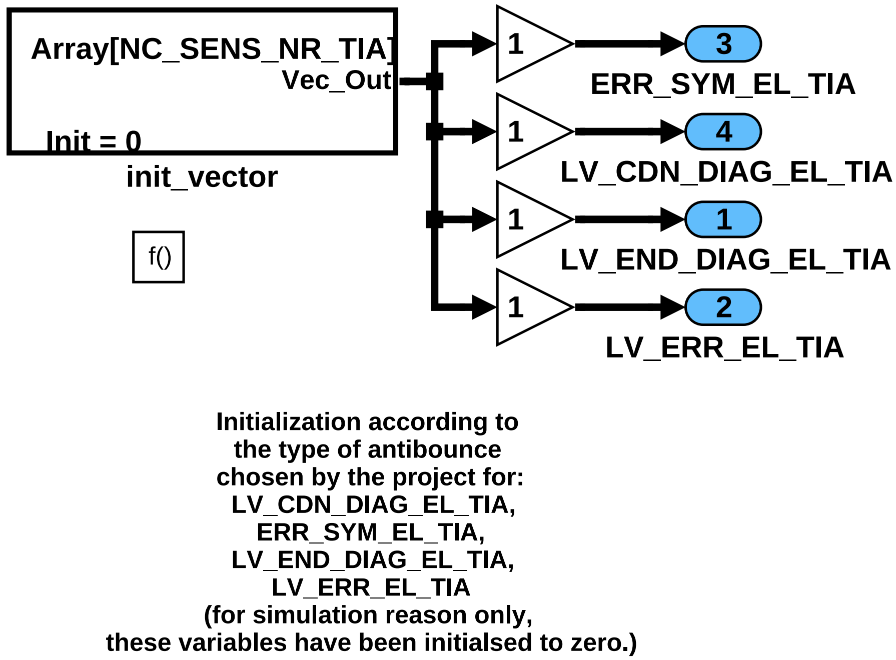
Only one symptom can be active at the same time.

Function Description



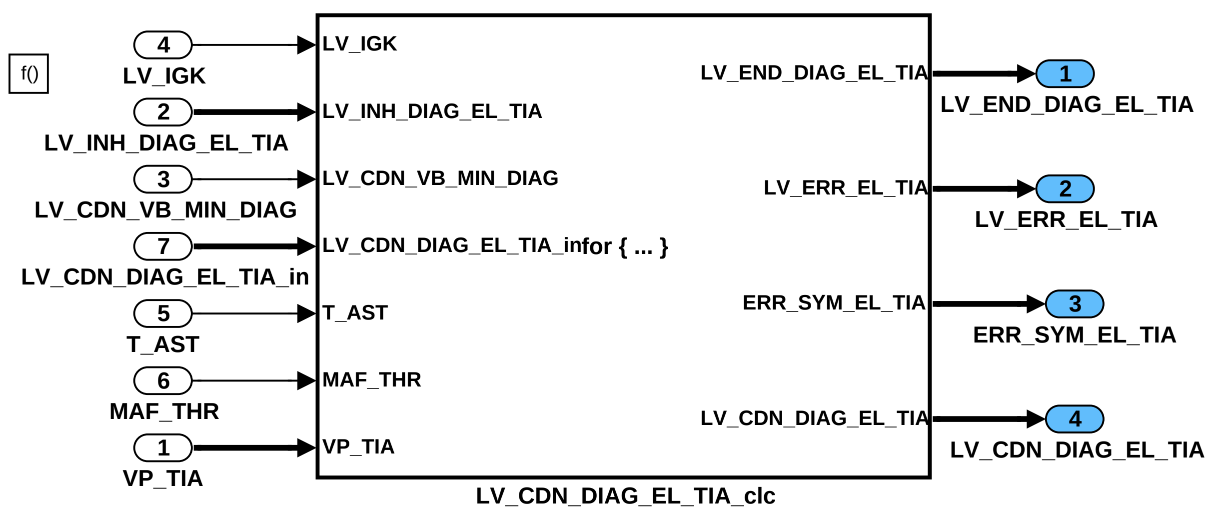
AIRT_SIGDG0

A.27.1.1 SUBFUNCTION: INIT_MA0F3

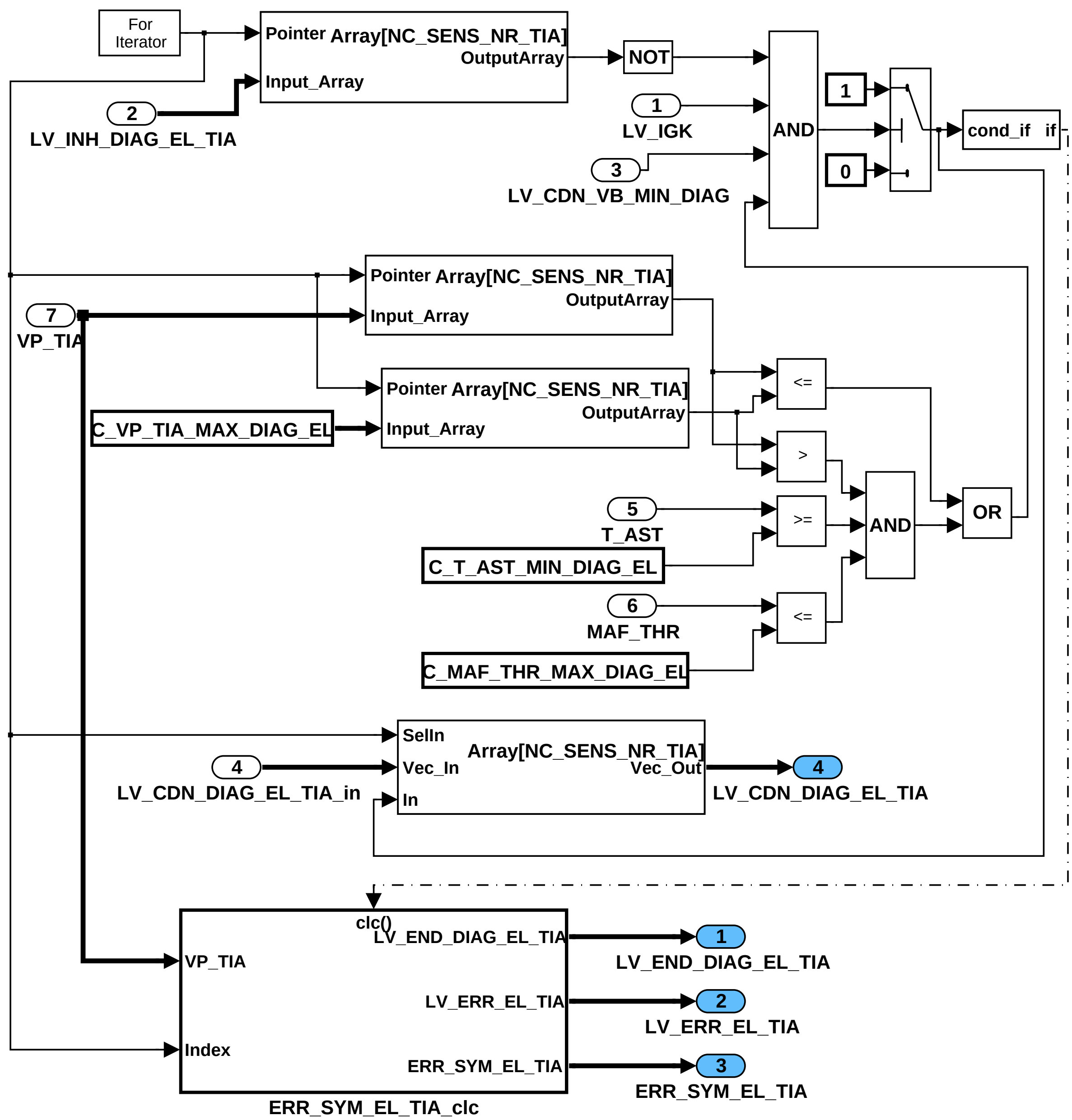


AIRT_SIGDG0/init_MA0F3

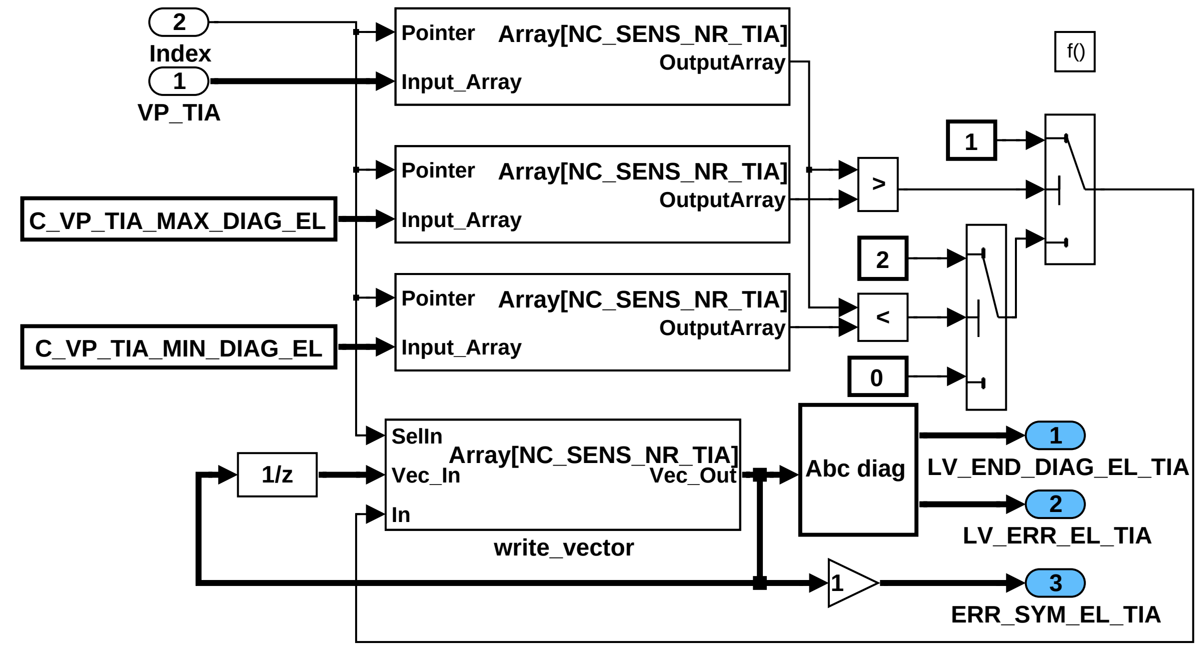
A.27.1.2 SUBFUNCTION: OPERATE_MA0F3



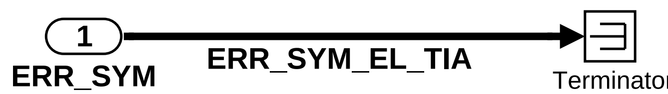
AIRT_SIGDG0/operate_MA0F3

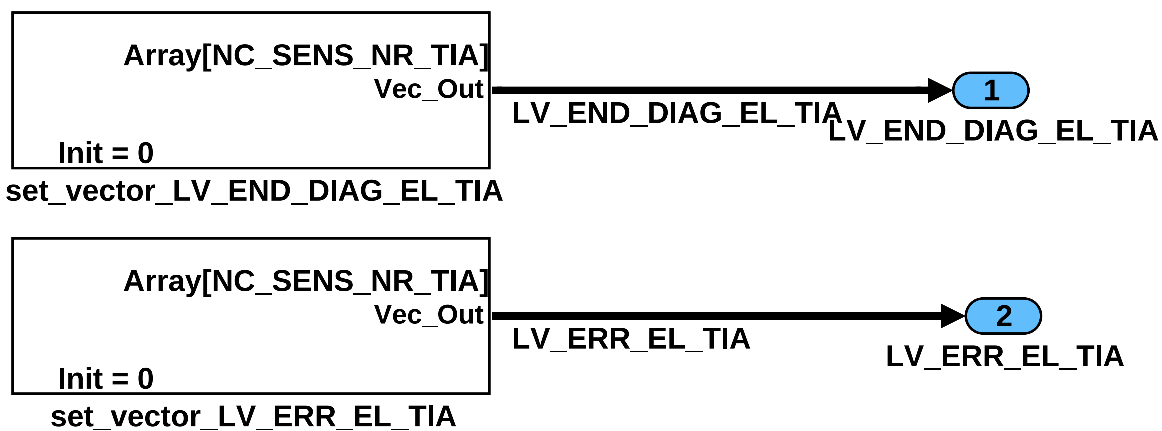


AIRT_SIGDG0/operate_MA0F3/LV_CDN_DIAG_EL_TIA_clc



AIRT_SIGDG0/operate_MA0F3/LV_CDN_DIAG_EL_TIA_clc/ERR_SYM_EL_TIA_clc





AIRT_SIGDG0/operate_MA0F3/LV_CDN_DIAG_EL_TIA_clc/ERR_SYM_EL_TIA_clc/Abc_diag

A.28 Intake Air tempreature sensor intermittent and stuck dignosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_INTM_TIA[NC_SENS_NR_TIA]	O/V	0H 1H	NO_SYM SYM_0	1	-
Symptom of TIA intermittent detected failure: one value for each TIA sensor of AIRT					
LV_ERR_INTM_TIA[NC_SENS_NR_TIA]	O/V	0...1H	0...1	1	-
Present TIA intermittent failure: one value for each TIA sensor of AIRT					
LV_CDN_DIAG_INTM_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
Conditions for TIA intermittent diagnosis: 0 = not fulfilled, 1 = fulfilled: one value for each TIA sensor of AIRT					
TIA_MES_DIF_DIAG_INTM[NC_SENS_NR_TIA]	V	0...FEH	0...190.5	0.75	°C
Intake air temperature difference used to detect INTM failure: one value for each TIA sensor of AIRT					
LV_END_DIAG_INTM_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
Result of intermittent diagnosis: one value for each TIA sensor of AIRT					
ERR_SYM_DYN_TIA[NC_SENS_NR_TIA]	O/V	0H 1H	NO_SYM SYM_0	1	-
Symptom of TIA stuck signal detected failure: one value for each TIA sensor of AIRT					
LV_ERR_DYN_TIA[NC_SENS_NR_TIA]	O/V	0...1H	0...1	1	-
Present TIA stuck signal failure: one value for each TIA sensor of AIRT					
LV_CDN_DIAG_DYN_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
Conditions for TIA stuck signal diagnosis: 0 = not fulfilled, 1 = fulfilled: one value for each TIA sensor of AIRT					
LV_END_DIAG_DYN_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
Result of TIA stuck signal diagnosis: one value for each TIA sensor of AIRT					
TIA_MES_DIF_DIAG_DYN[NC_SENS_NR_TIA]	V	0...FEH	0...190.5	0.75	°C
Intake air temperature difference used to detect DYN failure: one value for each TIA sensor of AIRT					
TIA_MES_MAX_DIAG_DYN[NC_SENS_NR_TIA]	V	0...FEH	-48...142.5	0.75	°C
Maximum TIA_MES used to detect DYN failure: one value for each TIA sensor of AIRT					
TIA_MES_MIN_DIAG_DYN[NC_SENS_NR_TIA]	V	0...FEH	-48...142.5	0.75	°C
Minimum TIA_MES used to detect DYN failure: one value for each TIA sensor of AIRT					
LV_CDN_DYN_TIA_LOAD	V	0...1H	0...1	1	-
MAF and VS condition for TIA : 0 = not fulfilled, 1 = fulfilled					
T_DYN_TIA_INC	V	0...FFFFH	0...6553.5	0.1	S
Timer for TIA signal increase check					
T_DYN_TIA_DEC	V	0...FFFFH	0...6553.5	0.1	S
Timer for TIA signal decrease check					

Input data:

TIA[NC_SENS_NR_TIA]	TIA_MES[NC_SENS_NR_TIA]	TIA_MES_MMV[NC_SENS_NR_TIA]	TIA_MES_ST[NC_SENS_NR_TIA]
LV_INH_DIAG_DYN_TIA[NC_SENS_NR_TIA]	LV_INH_DIAG_INTM_TIA[NC_SENS_NR_TIA]	ERR_SYM_EL_TIA[NC_SENS_NR_TIA]	LV_ERR_EL_TIA[NC_SENS_NR_TIA]
	TCO	LV_IGK	LV_ST_END
TCO_ST	TCO_STOP	CTR_ABC_EL_TIA	LV_CDN_VB_MIN_DIAG
NC_SENS_NR_TIA	VS	MAF_THR	

A.28.1 Intermittent Failure Monitoring (“INTM”):

The purpose is to detect intermittent failure in order to avoid sudden calculation transitions for any function based on a sensor temperature value. Failure is detected when the difference between the raw value and the mean value of TIA_MES overpasses a calibrateable threshold.

NOTE:

- after debouncing, LV_ERR_INTM_TIA[m] and LV_END_DIAG_INTM_TIA[m] are automatically managed.

- in order to avoid wrong detections, very high filtering on TIA_MES_MMV[m] is required.

- as this diagnosis is based on the measured value TIA_MES[m], an uncorrect calibration of the conversion table could lead wrong error detections.

Application conditions :

For m = 1 **to** NC_SENS_NR_TIA

Initialization:

According to the type of the anti-bounce :
LV_CDN_DIAG_INTM_TIA[m] = 0,
LV_ERR_INTM_TIA[m] = 0,
LV_END_DIAG_INTM_TIA[m] = 0,
ERR_SYM_INTM_TIA[m] = NO_SYM

At Reset
TIA_MES_DIF_DIAG_INTM[m] = 0°C

Recurrence: **100ms**

Activation: **If** LV_IGK = 1
And LV_CDN_VB_MIN_DIAG = 1
And LV_INH_DIAG_INTM_TIA[m] = 0
And CTR_ABC_EL_TIA[m] = 0
Then LV_CDN_DIAG_INTM_TIA[m] = 1
Endif

Deactivation: **If** one of the Activation condition is not fulfilled
Then LV_CDN_DIAG_INTM_TIA[m] = 0
Endif

End for m

Formula section:

For m = 1 **to** NC_SENS_NR_TIA

TIA_MES_DIF_DIAG_INTM[m] = |TIA_MES[m] - TIA_MES_MMV[m]|

If LV_CDN_DIAG_INTM_TIA[m] = 1
Then **If** TIA_MES_DIF_DIAG_INTM[m] > C_TIA_MES_DIF_DIAG_INTM
Then ERR_SYM_INTM_TIA[m] = “SYM_0”
And anti-bounce counter is incremented
Else ERR_SYM_INTM_TIA[m] = “NO_SYM”
And anti-bounce counter is decremented
Endif
Else *ERR_SYM_INTM_TIA[m] and the anti-bounce counter remain unchanged*
Endif

End for m

A.28.2 Stuck Signal Failure Monitoring (“DYN”):

Application conditions:

For m = 1 **to** NC_SENS_NR_TIA

Initialization:

According to the type of the anti-bounce :
LV_CDN_DIAG_DYN_TIA[m] = 0,
LV_ERR_DYN_TIA[m] = 0,
LV_END_DIAG_DYN_TIA[m] = 0,
ERR_SYM_DYN_TIA[m] = NO_SYM

At Reset
TIA_MES_DIF_DIAG_DYN[m] = 0°C
TIA_MES_MAX_DIAG_DYN = TIA_MES_ST[m]
TIA_MES_MIN_DIAG_DYN = TIA_MES_ST[m]
LV_CDN_DYN_TIA_LOAD = 0
T_DYN_TIA_DEC = 0
T_DYN_TIA_INC = 0

Recurrence: **100ms**

Activation:

If LV_IGK = 1
And LV_ST_END = 1
And LV_CDN_VB_MIN_DIAG = 1
And LV_INH_DIAG_DYN_TIA[m] = 0
And ERR_SYM_EL_TIA[m] = 0
And LV_ERR_EL_TIA[m] = 0
And ERR_SYM_INTM_TIA[m] = 0
And LV_ERR_INTM_TIA[m] = 0
And C_TIA_MIN_DIAG_DYN < TIA[m] < C_TIA_MAX_DIAG_DYN
And TCO >= C_TCO_MIN_DIAG_DYN
And |TCO_ST - TCO_STOP| >= C_TCO_DIF_TIA_DIAG_DYN
(rem: TCO_STOP is the value of TCO for the latest transition LV_ES = 0 to 1)
And |TCO_ST – TIA_MES_ST[m]| <= C_TEMP_ST_DIF_MAX_DIAG_DYN
And LV_CDN_DYN_TIA_LOAD = 1 (MAF_THR, VS condition for diag)
And LV_END_DIAG_DYN_TIA[m] = 0
(rem: in this way, the diagnosis is performed only once per Driving Cycle)
Then LV_CDN_DIAG_DYN_TIA[m] = 1
Endif

Deactivation:

If one of the Activation condition is not fulfilled
Then LV_CDN_DIAG_DYN_TIA[m] = 0
Endif

End for m

Formula section:

For m = 1 **to** NC_SENS_NR_TIA

(MAF_THR and VS condition for TIA stuck diagnosis)

If LV_ST_END = 1 **and** LV_CDN_DYN_TIA_LOAD = 0**then**

If MAF_THR >= C_MAF_THR_MIN_DYN_TIA_DEC
and VS >= C_VS_MIN_DYN_TIA_DEC
then T_DYN_TIA_DEC = T_DYN_TIA_DEC(n-1) + 0.1 *(100ms)*
endif

If MAF_THR <= C_MAF_THR_MAX_DYN_TIA_INC
and VS <= C_VS_MAX_DYN_TIA_INC
then T_DYN_TIA_INC = T_DYN_TIA_INC(n-1) + 0.1 *(100ms)*
endif

If T_DYN_TIA_DEC >= C_T_DYN_TIA_DEC_MAX
and T_DYN_TIA_INC >= C_T_DYN_TIA_INC_MAX
then LV_CDN_DYN_TIA_LOAD = 1
endif

endif

(Max/Min determination of TIA_MES)

If TIA_MES[m] > TIA_MES_MAX_DIAG_DYN[m]
then TIA_MES_MAX_DIAG_DYN[m] = TIA_MES[m]
If TIA_MES[m] < TIA_MES_MIN_DIAG_DYN[m]
then TIA_MES_MIN_DIAG_DYN[m] = TIA_MES[m]
endif

If LV_END_DIAG_DYN_TIA[m] = 0

Then TIA_MES_DIF_DIAG_DYN[m]

= |TIA_MES_MAX_DIAG_DYN[m] - TIA_MES_MIN_DIAG_DYN[m]|

Endif

If LV_CDN_DIAG_DYN_TIA[m] = 1

Then

If TIA_MES_DIF_DIAG_DYN[m] < IP_TIA_MES_DIF_DYN

Then ERR_SYM_DYN_TIA[m] = “SYM_0”

And anti-bounce counter is incremented

Else ERR_SYM_DYN_TIA[m] = “NO_SYM”

And anti-bounce counter is decremented

Endif

Else *ERR_SYM_DYN_TIA[m] and the anti-bounce counter remain unchanged*

Endif

End for m

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_DIF_TIA_DIAG_DYN	1	0...FEH	0...190.5	0.75	°C
Minimum TCO change to enable TIA stuck signal diagnostic					
C_TCO_MIN_DIAG_DYN	1	0...FEH	-48...142.5	0.75	°C
minimum TCO value to enable TIA stuck signal diagnosis					
C_TEMP_ST_DIF_MAX_DIAG_DYN	1	0...FEH	0...190.5	0.75	°C
maximum temperature difference at start to enable TIA stuck diagnosis					
C_TIA_MAX_DIAG_DYN	1	0...FEH	-48...142.5	0.75	°C
Maximum TIA value to enable TIA stuck signal diagnosis					
C_TIA_MES_DIF_DIAG_INTM	1	0...FEH	0...190.5	0.75	°C
Measured TIA difference threshold for TIA intermittent diagnosis					
IP_TIA_MES_DIF_DYN	5	0...FEH	0...190.5	0.75	°C
LDP_TIA_ST_IP_TIA_MES_DIF_DYN	5	0H...FEH	-48...142.5	0.75	°C
Minimum necessary TIA difference to detect TIA stuck signal error					
C_TIA_MIN_DIAG_DYN	1	0...FEH	-48...142.5	0.75	°C
Minimum TIA value to enable stuck signal diagnosis					
C_MAF_THR_MIN_DYN_TIA_DEC	1	0...FFFFH	0...2047.96875	0.03125	kg/h
Minimum mass air flow value for TIA decrease					
C_MAF_THR_MAX_DYN_TIA_INC	1	0...FFFFH	0...2047.96875	0.03125	kg/h
Maximum mass air flow value for TIA increase					
C_VS_MIN_DYN_TIA_DEC	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for TIA decrease					
C_VS_MAX_DYN_TIA_INC	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for TIA increase					
C_T_DYN_TIA_DEC_MAX	1	0...FFFFH	0...6553.5	0.1	S
Maximum timer value in case TIA decrease check					
C_T_DYN_TIA_INC_MAX	1	0...FFFFH	0...6553.5	0.1	S
Maximum timer value in case TIA increase check					

A.29 Electrical ambient air temperature sensor diagnosis (TAM from analog sensor)

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
ERR_SYM_EL_TAM	O/V	0 1 2 4 8	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	-	[-]
Symptom for Electrical failure of Ambient-air-temperature sensor					
LV_CDN_DIAG_EL_TAM	V	0... 1H	0... 1	1	[-]
Conditions for EL_TAM diagnosis: 0 = not fulfilled, 1 = fulfilled					
LV_END_DIAG_EL_TAM	V	0... 1H	0... 1	1	[-]
Logical variable indicating end of Ambient-air-temperature sensor electrical diagnosis					
LV_ERR_EL_TAM	O/V	0... 1H	0... 1	1	[-]
Logical variable to indicate an electrical failure of the ambient-air-temperature sensor					
T_VS_EL_TAM	V	0...FFFFH	0...6553.5	0.1	[s]
Timer for electrical TAM sensor diagnosis condition					

Input data:

CONF_TAM	LV_CDN_VB_OBD1	VS	LV_IGK
VP_TAM	T_AST		

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the voltage value VP_TAM from the ambient air temperature sensor and to detect an electrical failure.

Error-symptoms and conditions: are defined to this diagnosis function as following

```
□□□□
│ │ │ └─> short circuit to battery / open circuit      (= SYM_0)
│ │ └───> short circuit to ground                      (= SYM_1)
│ └─────> -                                            (= SYM_2)
└───────> -                                            (= SYM_3)
```

Application conditions:

Initialisation: *according filter type: "STD-INI"*

T_VS_EL_TAM = C_T_VS_EL_TAM at reset

Recurrence: *100 ms*

Activation: *CONF_TAM = 2*

Formula section:

```
If      VS >= C_VS_MIN_EL_TAM
Then    T_VS_EL_TAM = T_VS_EL_TAM – 0.1
Endif

If      T_VS_EL_TAM = 0
And     LV_IGK = 1
And     LV_CDN_VB_OBD1 = 1
And     VP_TAM <= C_VP_TAM_MAX_DIAG_EL
        Or      (VP_TAM > C_VP_TAM_MAX_DIAG_EL
                And     T_AST >= C_T_AST_MIN_DIAG_EL_TAM)
Then    LV_CDN_DIAG_EL_TAM = 1
Else    LV_CDN_DIAG_EL_TAM = 0
End

If  LV_CDN_DIAG_EL_TAM = 1
Then    If      VP_TAM > C_VP_TAM_MAX_DIAG_EL
        Then    ERR_SYM_EL_TAM = “SYM_0”
        And anti-bounce counter is incremented
        Else    If      VP_TAM < C_VP_TAM_MIN_DIAG_EL
                Then    ERR_SYM_EL_TAM = “SYM_1”
                And anti-bounce counter is incremented
                Else    ERR_SYM_EL_TAM = “NO_SYM”
                        And anti-bounce counter is decremented
                Endif
        Endif
Else    ERR_SYM_EL_TAM and the anti-bounce counter remain unchanged
Endif
```

Calculation of LV_END_DIAG_EL_TAM and LV_ERR_EL_TAM

LV_ERR_EL_TAM	Calculated by error management : CTR_ABC_EL_TAM reach maximum
LV_END_DIAG_EL_TAM	Calculated by error management

Configuration for diagnostic symptoms :

Diagnostic TAM	*Symptom description*	Symptom	Filter type
Range ambient temperature from analog sensor diagnosis	*short circuit to battery / open circuit*	*SYM_0*	*STD_INI*
	short circuit to ground	*SYM_1*	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_EL_TAM	1	0...FFH	0...255	1	-
Increment debounce counter for failure TAM from analog sensor					
C_ABC_MAX_EL_TAM	1	1...FFH	1...255	1	-
Maximum value debounce counter failure TAM from analog sensor					
C_VP_TAM_MAX_DIAG_EL	1	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Voltage threshold value for short circuit to Battery or open circuit check of TAM sensor(VP_TAM)					
C_VP_TAM_MIN_DIAG_EL	1	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Voltage threshold value for short circuit to Ground check of TAM sensor(VP_TAM)					
C_T_VS_EL_TAM	1	0...FFFFH	0...6553.5	0.1	[s]
Timer for electrical TAM sensor diagnosis condition					
C_VS_MIN_EL_TAM	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for electrical TAM sensor diagnosis					
C_T_AST_MIN_DIAG_EL_TAM	1	0...FFFFH	0...6553.5	0.1	[s]
Minimum time after start value for electrical TAM sensor diagnosis condition					

A.30 Ambient air temperature sensor diagnosis from CAN

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
ERR_SYM_TAM_CAN	V/O	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	[-]
Symptom of TAM failure from CAN					
LV_CDN_DIAG_TAM_CAN	V	0...1H	0...1	1	[-]
Conditions for TAM diagnosis from CAN: 0 = not fulfilled, 1 = fulfilled					
LV_END_DIAG_TAM_CAN	V	0...1H	0...1	1	[-]
Result of TAM diagnosis from CAN					
LV_ERR_TAM_CAN	V/O	0H 1H	PASSIVE ACTIVE	1	[-]
Boolean for CAN_ERR on ambient temperature TAM_CAN					

Input data:

STATE_ERR_TAM_CAN	LV_TAM_CAN_FIRST_VLD	LV_IGK	LV_ERR_CAN_BUS_OFF
CONF_TAM			

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis is to read the CAN error flags of the TAM message from CAN and to transform them in signals necessary for the ECU and ERRM and the calculation of TAM.

Error-symptoms and conditions: are defined to this diagnosis function as following

```
┌─┬─┬─┐
│ │ │ │
└─┴─┴─┘
 │  │  └─> not valid          (= SYM_0)
 │  └────> short circuit      (= SYM_1)
 └───────> open circuit       (= SYM_2)
```

Application conditions:

Initialisation: *LV_IGK = 0 -> 1 and reset initialise all output variables with 0H.*

Recurrence: *100 ms*

Activation: *CONF_TAM = 1*

Formula section:

Conditions calculation:

If LV_TAM_CAN_FIRST_VLD = 1
And LV_ERR_CAN_BUS_OFF = 0
And LV_IGK = 1
Then LV_CDN_DIAG_TAM_CAN = 1
Else LV_CDN_DIAG_TAM_CAN = 0
Endif

Symptoms calculation:

If LV_CDN_DIAG_TAM_CAN = 1
Then **If** STATE_ERR_TAM_CAN = 0H *(no SYM)*
Then ERR_SYM_TAM_CAN = 0H
Elseif STATE_ERR_TAM_CAN = 1H *(TAM signal from CAN not valid)*
Then ERR_SYM_TAM_CAN = 1H
Elseif STATE_ERR_TAM_CAN = 2H *(Short circuit)*
Then ERR_SYM_TAM_CAN = 2H
Else ERR_SYM_TAM_CAN = 4H *(Open circuit)*
Endif
Else ERR_SYM_TAM_CAN = 0H *(no SYM)*
Endif

Calculation of LV_END_DIAG_TAM_CAM and LV_ERR_TAM_CAN

LV_ERR_TAM_CAN — Calculated by error management : CTR_ABC_TAM_CAN reach maximum

LV_END_DIAG_TAM_CAN — Calculated by error management

Configuration for diagnostic symptoms :

Diagnostic TAM	*Symptom description*	Symptom	Filter type
Range ambient temperature from CAN diagnosis	*not valid*	*SYM_0*	*STD_INI*
	Short circuit	*SYM_1*	
	Open circuit	*SYM_2*	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TAM_CAN	1	0...FFH	0...255	1	-
Increment debounce counter for failure TAM from CAN					
C_ABC_MAX_TAM_CAN	1	1...FFH	1...255	1	-
Maximum value debounce counter failure TAM from CAN					

A.31 TAM range check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ORNG_TAM	V/O	0...1H	0...1	1	[-]
Present failure TAM range					
ERR_SYM_ORNG_TAM	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom of TAM range failure					

Input data:

General information:

Application conditions:

Initialization: **At Reset:** LV_ERR_ORNG_TAM = 0
ERR_SYM_ORNG_TAM = “NO_SYM”

Recurrence: no (the outputs are just calculated once)

Activation: always active

Deactivation: -

Formula section:

Calibration data:

A.32 Error flags for TAM, TIA_THR, TIA_IM and TIA_CYL

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TAM	V/O	0...1H	0...1	1	[-]
Boolean for error currently present which might affect temperature TAM (after debounce)					
LV_ERR_TIA_CYL	V/O	0...1H	0...1	1	[-]
Boolean for error currently present which might affect temperature TIA_CYL (after debounce)					
LV_ERR_TIA_IM	V/O	0...1H	0...1	1	[-]
Boolean for error currently present which might affect temperature TIA_IM (after debounce)					
LV_ERR_TIA_THR	V/O	0...1H	0...1	1	[-]
Boolean for error currently present which might affect temperature TIA_THR (after debounce)					
LV_ERR_EL_TIA_THR	V/O	0...1H	0...1	1	[-]
Result of electrical diagnosis on one TIA sensor(s) which might affect temperature TIA_THR					
LV_END_DIAG_EL_TIA_THR	V/O	0...1H	0...1	1	[-]
Result of electrical diagnosis on one TIA sensor(s) which might affect temperature TIA_THR					

Input data:

LV_ERR_EL_TIA[NC_SENS_NR_TIA]	LV_ERR_DYN_TIA[NC_SENS_NR_TIA]	LV_ERR_INTM_TIA[NC_SENS_NR_TIA]	LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA]
LV_ERR_TIA_CHRG_DOWN	LV_ERR_TIA_CHRG_UP	NC_TIA_CONF	NC_IDX_TIA_AM_CHRG
NC_IDX_TIA_CHRG_THR	NC_IDX_TIA_IM_CYL	LV_ERR_EL_TAM	NC_SENS_NR_TIA
LV_END_DIAG_EL_TIA[NC_SENS_NR_TIA]	LV_ERR_PLAUS_TAM	LV_ERR_TAM_CAN	LV_ERR_ORNG_TAM
CONF_TAM			

A.32.1 General information [Version for : NC_CHRG_CONF<>0 And NC_TIA_CONF=10,11,12,13,21,22,23,24,30]

- For each TIA sensor, 3 Error Flags are defined (in Chapter Diagnosis):

LV_ERR_**EL**_TIA,
LV_ERR_**INTM**_TIA,
LV_ERR_**DYN**_TIA,
LV_ERR_**PLAUS**_TIA.

- For each TAM sensor (if available), 2 Error Flags are defined (in Chapter Diagnosis):

LV_ERR_**ORNG**_TAM,
LV_ERR_**PLAUS**_TAM.

Based on these values we are now defining 1 Error Flag associated to each temperature TAM, TIA_THR, TIA_IM and TIA_CYL. They are named:

LV_ERR_TAM,
LV_ERR_TIA_THR,
LV_ERR_TIA_IM,
LV_ERR_TIA_CYL,
LV_ERR_TAM , LV_ERR_TIA_THR, LV_ERR_TIA_IM and LV_ERR_TIA_CYL are defined as Output of AIRT Aggregate.

- Additionnaly LV_ERR_EL_TIA_THR and LV_END_DIAG_EL_TIA_THR are defined for other Aggregate diagnosis based on TIA_THR

Application conditions:

Initialisation : **At** Reset

LV_ERR_TIA_THR, LV_ERR_TIA_IM, LV_ERR_TIA_CYL must be initialized with formula section.

Recurrence : **see below**

Activation : At all Engine States

Deactivation: -

Formula section:

A.32.2 LV_ERR_TAM definition:

Recurrence : **1 s**

A value for NC_TIA_CONF is defined for each hardware definition (see specification Air Temperature Sensor(s) Configuration)

Depending on NC_TIA_CONF, the values of LV_ERR_TAM is defined.

```
If    CONF_TAM <> 0
Then        If   LV_ERR_TAM_CAN=1
            Or LV_ERR_ORNG_TAM=1
            Or LV_ERR_PLAUS_TAM=1
            Or LV_ERR_EL_TAM = 1
            Then LV_ERR_TAM = 1
            Else  LV_ERR_TAM = 0
            Endif
Else        #If NC_TIA_CONF = 10
            #Then  LV_ERR_TAM = LV_ERR_z_TIA[NC_IDX_TIA_IM_CYL]
            #Endif
            #If NC_TIA_CONF = 13 Or 21
            #Then  LV_ERR_TAM = LV_ERR_z_TIA[NC_IDX_TIA_CHRG_THR]
            #Endif
            #If NC_TIA_CONF = 12 Or 24
            #Then  LV_ERR_TAM = LV_ERR_z_TIA[NC_IDX_TIA_CHRG_UP]
            #Endif
            #If NC_TIA_CONF = 11 Or 22 Or 23 Or 30
            #Then  LV_ERR_TAM = LV_ERR_z_TIA[NC_IDX_TIA_AM_CHRG]
            #Endif
Endif
```

A.32.3 LV_ERR_TIA_THR, LV_ERR_TIA_IM and LV_ERR_TIA_CYL definition:

Recurrence : **100 ms.**

In the same way, depending on NC_TIA_CONF value, the values of LV_ERR_TIA_THR, LV_ERR_TIA_IM and LV_ERR_TIA_CYL are defined.

NC_TIA_CONF	10	11	13, 22
LV_ERR_TIA_THR LV_ERR_TIA_IM LV_ERR_TIA_CYL	LV_ERR_**z**_ TIA[NC_IDX_TIA_IM_CYL]	LV_ERR_**z**_ TIA[NC_IDX_TIA_AM_CHRG]	LV_ERR_**z**_ TIA[NC_IDX_TIA_CHRG_THR]

NC_TIA_CONF	12
LV_ERR_TIA_THR LV_ERR_TIA_IM LV_ERR_TIA_CYL	LV_ERR_TIA_CHRG_DOWN

NC_TIA_CONF	23, 24	21, 30
LV_ERR_TIA_THR	LV_ERR_TIA_CHRG_DOWN	LV_ERR_**z**_ TIA[NC_IDX_TIA_CHRG_THR]
LV_ERR_TIA_IM LV_ERR_TIA_CYL	LV_ERR_**z**_ TIA[NC_IDX_TIA_IM_CYL]	

Example: for NC_TIA_CONF = 21:

From column "20" of the first table we define:

- For LV_ERR_TIA_THR:

LV_ERR_TIA_THR is defined based on LV_ERR_**z**_TIA[NC_IDX_TIA_CHRG_THR] as follow:

If LV_ERR_**EL**_TIA[NC_IDX_TIA_CHRG_THR] = 0
And LV_ERR_**INTM**_TIA[NC_IDX_TIA_CHRG_THR] = 0
And LV_ERR_**DYN**_TIA[NC_IDX_TIA_CHRG_THR] = 0
And LV_ERR_**PLAUS**_TIA[NC_IDX_TIA_CHRG_THR] = 0
Then LV_ERR_TIA_THR = 0
Else LV_ERR_TIA_THR = 1
Endif

- For LV_ERR_TIA_IM and LV_ERR_TIA_CYL:

LV_ERR_TIA_IM and LV_ERR_TIA_CYL are defined based on LV_ERR_**z**_TIA[NC_IDX_TIA_IM_CYL] as follow:

If LV_ERR_**EL**_TIA[NC_IDX_TIA_IM_CYL] = 0
And LV_ERR_**INTM**_TIA[NC_IDX_TIA_IM_CYL] = 0
And LV_ERR_**DYN**_TIA[NC_IDX_TIA_IM_CYL] = 0
And LV_ERR_**PLAUS**_TIA[NC_IDX_TIA_IM_CYL] = 0
Then LV_ERR_TIA_IM = LV_ERR_TIA_CYL = 0
Else LV_ERR_TIA_IM = LV_ERR_TIA_CYL = 1
Endif

1st Exception: for the case: ***NC_TIA_CONF = 12***

For these cases, only the Error flag LV_ERR_TIA_CHRG_DOWN is delivered by the Charger Aggregate. Also, LV_ERR_TIA_THR, LV_ERR_TIA_IM, LV_ERR_TIA_CYL are defined as follow:

LV_ERR_TIA_THR = LV_ERR_TIA_IM = LV_ERR_TIA_CYL = LV_ERR_TIA_CHRG_DOWN

2nd Exception: for the 2 cases: ***NC_TIA_CONF = 23 Or 24***

For the same raison, LV_ERR_TIA_THR, LV_ERR_TIA_IM, LV_ERR_TIA_CYL are defined as follow:

- For LV_ERR_TIA_THR:

LV_ERR_TIA_THR = LV_ERR_TIA_CHRG_DOWN

- For LV_ERR_TIA_IM and LV_ERR_TIA_CYL

If LV_ERR_**EL**_TIA[NC_IDX_TIA_IM_CYL] = 0
And LV_ERR_**INTM**_TIA[NC_IDX_TIA_IM_CYL] = 0

And LV_ERR_**DYN**_TIA[NC_IDX_TIA_IM_CYL] = 0
And LV_ERR_**PLAUS**_TIA[NC_IDX_TIA_IM_CYL] = 0
Then LV_ERR_TIA_IM = LV_ERR_TIA_CYL = 0
Else LV_ERR_TIA_IM = LV_ERR_TIA_CYL = 1
Endif

A.32.4 LV_ERR_EL_TIA_THR and LV_END_DIAG_EL_TIA_THR definition:

Recurrence : **100 ms.**

#If NC_TIA_CONF = 10
#Then LV_ERR_EL_TIA_THR = LV_ERR_EL_TIA[NC_IDX_TIA_IM_CYL]
LV_END_DIAG_EL_TIA_THR = LV_END_DIAG_EL_TIA[NC_IDX_TIA_IM_CYL]
#Endif
#If NC_TIA_CONF = 11
#Then LV_ERR_EL_TIA_THR = LV_ERR_EL_TIA[NC_IDX_TIA_AM_CHRG]
LV_END_DIAG_EL_TIA_THR = LV_END_DIAG_EL_TIA[NC_IDX_TIA_AM_CHRG]
#Endif
#If NC_TIA_CONF = 13 **Or** 21 **Or** 22 **Or** 30
#Then LV_ERR_EL_TIA_THR = LV_ERR_EL_TIA[NC_IDX_TIA_CHRG_THR]
LV_END_DIAG_EL_TIA_THR = LV_END_DIAG_EL_TIA[NC_IDX_TIA_CHRG_THR]
#Endif
#If NC_TIA_CONF = 12 **Or** 23 **Or** 24
#Then LV_ERR_EL_TIA_THR = 0 (just to fullfill interfaces)
LV_END_DIAG_EL_TIA_THR = 1 (just to fullfill interfaces)
#Endif

A.33 Error Flags for Charge Air Temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TIA_CHRG_UP	O	0...1H	0...1	1	-
Present failure on one TIA sensor(s) which might affect temperature TIA_CHRG_UP					
LV_ERR_TIA_CHRG_DOWN	O	0...1H	0...1	1	-
Present failure on one TIA sensor(s) which might affect temperature TIA_CHRG_DOWN					

Input data:

NC_TAM_CAN_USE	NC_TIA_CONF	LV_ERR_TIA_THR	LC_TCHA_CONF

A.33.1 CHRG_FCTDGTAIR0

Description:

The aim is to define the errors flags LV_ERR_TIA_CHRG_UP [respectively _DOWN] associated to the temperatures TIA_CHRG_UP [respectively _DOWN]. These two flags are made available to the other Aggregates.

Application Condition

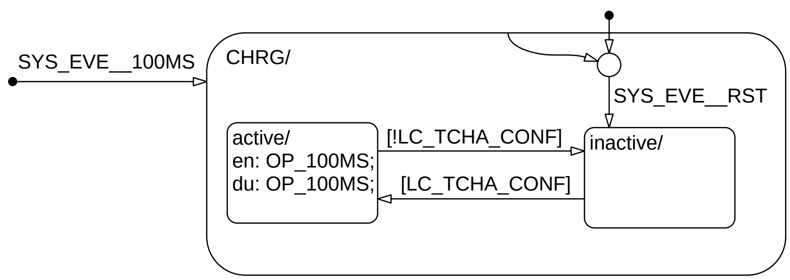


Figure 1 CHRG_FCTDGTAIR0/ APP_CDN/ Chart

Function Description

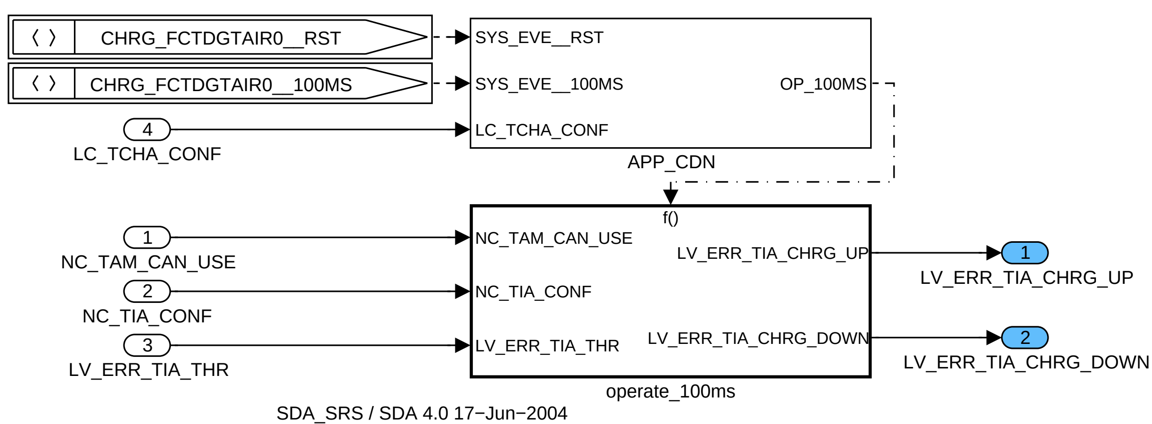


Figure 2 CHRG_FCTDGTAIR0

A.33.1.1 SUBFUNCTION: operate_100ms

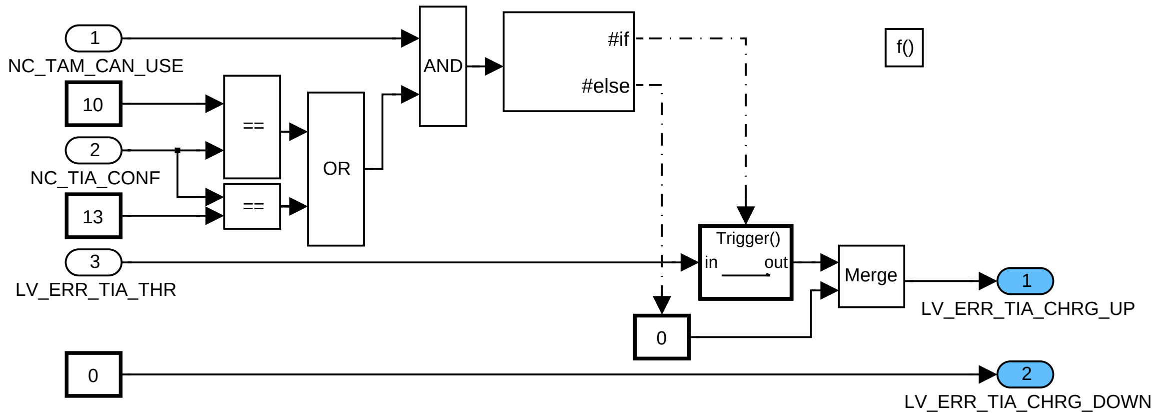


Figure 3 CHRG_FCTDGTAIR0/ operate_100ms

A.34 Coolant temperature diagnosis interface

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DIAG_AST	V/O	0...FFFFH	0...65535	1	[s]
Time after engine start for diagnosis functions					
T_IS_AST	V	0...FFFFH	0...65535	1	[s]
“Idle speed” activation time since engine start					
RATIO_T_IS_AST	V/O	0...FFH	0...99.60937	0.390625	[%]
“Idle speed” activation time in percent since engine start					
T_PUC_AST	V	0...FFFFH	0...65535	1	[s]
"Trailing throttle fuel cut-off" activation time since engine start					
RATIO_T_PUC_AST	V/O	0...FFH	0...99.60937	0.390625	[%]
"Trailing throttle fuel cut-off" activation time in percent since engine start					
T_LOAD_MIN_AST	V	0...FFFFH	0...65535	1	[s]
"Minimum load" time since engine start					
RATIO_T_LOAD_MIN_AST	V/O	0...FFH	0...99.60937	0.390625	[%]
"Minimum load" time in percent since engine start					
T_VS_MIN_AST	V	0...FFFFH	0...65535	1	[s]
"Minimum vehicle speed " time since engine start					
RATIO_T_VS_MIN_AST	V/O	0...FFH	0...99.60937	0.390625	[%]
"Minimum vehicle speed " time in percent since engine start					
T_VS_MAX_AST	V	0...FFFFH	0...65535	1	[s]
"Maximum vehicle speed " time since engine start					
RATIO_T_VS_MAX_AST	V/O	0...FFH	0...99.60937	0.390625	[%]
"Maximum vehicle speed " time in percent since engine start					
LV_CDN_ENA_RATIO_T_AST	V	0...1H	0...1	1	[-]
Boolean for ratio time after engine start enable condition					

Input data:

LV_IS	LV_PUC	MAF_KGH	VS
TAM			

FUNCTION DESCRIPTION:

General information:

The coolant temperature increase after engine start is most dependent on the ambient- and the engine operating conditions. To avoid monitoring failures within the diagnosis functions related to the coolant temperature behaviour, the influence of critical engine driving conditions must be taken into consideration. To allow a comparison to the time elapsed since engine start, the activation time of the critical driving conditions are calculated in percent of the elapsed time. The percentage values are used in several diagnosis functions to cancel monitoring in order to avoid failures.

Application conditions:

Initialisation at Exit start (EXIT_ST):

LV_CDN_ENA_RATIO_T_AST = 1

all other output data are set to “0” [phys]

Initialisation at Engine run to Engine stop (ERU_to_ES):

LV_CDN_ENA_RATIO_T_AST = 0

T_DIAG_AST = 0 s

Recurrence: 1000 ms

Activation: LV_CDN_ENA_RATIO_T_AST = 1

Deactivation: LV_CDN_ENA_RATIO_T_AST = 0

Formula section:

Calculation of the after start time for diagnosis functions:

If T_DIAG_AST < 65535 s

then $T\_DIAG\_AST_{(n)} = T\_DIAG\_AST_{(n-1)} + 1$ s

else $T\_DIAG\_AST_{(n)} = T\_DIAG\_AST_{(n-1)}$

endif

Calculation of the "idle speed" period after start:

If LV_IS = 1

then $T\_IS\_AST_{(n)} = T\_IS\_AST_{(n-1)} + 1$ s

else $T\_IS\_AST_{(n)} = T\_IS\_AST_{(n-1)}$

endif

RATIO_T_IS_AST = (T_IS_AST / T_DIAG_AST) * 100 %

Calculation of the " trailing throttle fuel cut-off " period after start:

If LV_PUC = 1

then $T\_PUC\_AST_{(n)} = T\_PUC\_AST_{(n-1)} + 1$ s

else $T\_PUC\_AST_{(n)} = T\_PUC\_AST_{(n-1)}$

endif

RATIO_T_PUC_AST = (T_PUC_AST / T_DIAG_AST) * 100 %

Calculation of the "low load" period after start:

If MAF_KGH <= IP_MAF_KGH_LOAD_MIN_AST

then $T\_LOAD\_MIN\_AST_{(n)} = T\_LOAD\_MIN\_AST_{(n-1)} + 1$ s

else $T\_LOAD\_MIN\_AST_{(n)} = T\_LOAD\_MIN\_AST_{(n-1)}$

endif

RATIO_T_LOAD_MIN_AST = (T_LOAD_MIN_AST / T_DIAG_AST) * 100 %

Calculation of the "low vehicle speed" period after start:

If VS <= C_VS_THD_VS_MIN_AST

then $T\_VS\_MIN\_AST_{(n)} = T\_VS\_MIN\_AST_{(n-1)} + 1$ s

else $T\_VS\_MIN\_AST_{(n)} = T\_VS\_MIN\_AST_{(n-1)}$

endif

RATIO_T_VS_MIN_AST = (T_VS_MIN_AST / T_DIAG_AST) * 100 %

Calculation of the "high vehicle speed" period after start:

If VS > C_VS_THD_VS_MAX_AST

then $T\_VS\_MAX\_AST_{(n)} = T\_VS\_MAX\_AST_{(n-1)} + 1$ s

else $T\_VS\_MAX\_AST_{(n)} = T\_VS\_MAX\_AST_{(n-1)}$

endif

RATIO_T_VS_MAX_AST = (T_VS_MAX_AST / T_DIAG_AST) * 100 %

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_THD_VS_MIN_AST	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed threshold for “low vehicle speed” period calculation after start					
C_VS_THD_VS_MAX_AST	1	0...FFH	0...255	1	[km/h]
Maximum vehicle speed threshold for “high vehicle speed” period calculation after start					
IP_MAF_KGH_LOAD_MIN_AST	4	0...FFFFH	0...2047.96875	0.03125	[kg/h]
LDP_TAM_IP_MAF_KGH_LOAD_MIN_AST	4	0...FEH	-48...142.5	0.75	[°C]
Ambient temperature dependent minimum engine load threshold for “low load” period calculation after start					

A.35 Coolant temperature sensor diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TCO	V/O	0...1H	0...1	1	[-]
Boolean for error currently present on coolant temperature acquisition					
LV_ERR_TCO_PREL	V/O	0...1H	0...1	1	[-]
Boolean for preliminary error currently present on coolant temperature acquisition					
LV_ERR_TCO_PREL_DET	V/S	0...1H	0...1	1	[-]
Boolean for preliminary error present on coolant temperature acquisition during driving cycle					
LV_CDN_ENA_TCO_PREL	V	0...1H	0...1	1	[-]
Boolean for preliminary coolant temperature error enable conditions					
LV_ERR_TCO_EL	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature signal range error					
ERR_SYM_TCO_EL	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom for coolant temperature signal range error					
LV_CDN_DIAG_TCO_EL	V	0...1H	0...1	1	[-]
Boolean for coolant temperature signal range diagnosis conditions					
LV_END_DIAG_TCO_EL	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature signal range diagnosis					
LV_ERR_TCO_GRD	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature signal gradient error					
ERR_SYM_TCO_GRD	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom for coolant temperature signal gradient error					
LV_CDN_DIAG_TCO_GRD	V	0...1H	0...1	1	[-]
Boolean for coolant temperature signal gradient diagnosis conditions					
LV_END_DIAG_TCO_GRD	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature signal gradient diagnosis					
TCO_MES_LIM_GRD	V	0...FEH	-48...142.5	0.75	[°C]
Limited measured coolant temperature for signal gradient diagnosis					

Input data:

TCO_MES	LV_ERR_TCO_STUCK	LV_ERR_TCO_STUCK_H	LV_ERR_TCO_PLAUS
LV_ERR_TCO_STUCK_RNG	TIA_THR	T_DIAG_AST	LV_IGK
LV_INH_DIAG_TCO_EL	LV_CDN_VB_MIN_DIAG	CTR_ABC_TCO_EL	CTR_ABC_TCO_GRD
LV_INH_DIAG_TCO_GRD	VP_TCO [NC_NR_TCO_SENS]	NC_NR_TCO_SENS	

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis function is to detect electrical failures as defined by OBD I and OBD II requirements. The coolant temperature raw sensor voltage (VP_TCO[0]) is measured with use of an A/D converter by the ECU hardware. The modified sensor signal is converted then into a measured temperature value (TCO_MES). Each by the ECU hardware measured raw sensor value is checked as well as the converted temperature signal.

Application conditions:

See separate chapters:

A.35.1 Global coolant temperature failure

FUNCTION DESCRIPTION:

General information:

A global coolant temperature error is set without debounce as soon as an electrical-, or signal gradient-, or stuck signal-, or plausibility- error is detected. All separate diagnosis functions are performed all the time if the diagnosis conditions are fulfilled. The deactivation of diagnosis functions with use of configuration switches is not possible.

In addition a preliminary coolant temperature error is set as soon as debouncing of an electrical-, or signal gradient- symptom occurs or in case of a plausibility-, or stuck signal-error. The preliminary coolant temperature error will be stored in the non-volatile memory during the ECU's self-holding phase to be available at the beginning of the next driving cycle.

Application conditions:

Initialisation at Reset:

LV_ERR_TCO = 0
LV_CDN_ENA_TCO_PREL = 1
LV_ERR_TCO_PREL = LV_ERR_TCO_PREL_DET *(NVMY)*
LV_ERR_TCO_PREL_DET = LV_ERR_TCO_PREL_DET *(NVMY)*
(NVMY: stored in the non-volatile memory)

Initialisation at Engine stop to Engine run (ES_to_ERU):

LV_ERR_TCO_PREL_DET = 0
LV_CDN_ENA_TCO_PREL = 0

Initialisation at FMY clear:

LV_ERR_TCO = 0
LV_ERR_TCO_PREL = 0
LV_ERR_TCO_PREL_DET = 0

Recurrence: 100 ms
Activation: at every engine operating state

Formula section:

If	LV_ERR_TCO_EL = 1	**or**
	LV_ERR_TCO_GRD = 1	**or**
	LV_ERR_TCO_STUCK = 1	**or**
	LV_ERR_TCO_STUCK_H = 1	**or**
	LV_ERR_TCO_STUCK_RNG = 1	**or**
	LV_ERR_TCO_PLAUS = 1	**or**
then	LV_ERR_TCO = 1	(without debounce)
else	LV_ERR_TCO = 0	
endif		

A.35.1.1 Preliminary coolant temperature failure (at ES after RESET)

Application conditions:

Recurrence:	100 ms
Activation:	LV_CDN_ENA_TCO_PREL = 1
Deactivation:	LV_CDN_ENA_TCO_PREL = 0

Formula section:

If	CTR_ABC_TCO_EL > 0	**or**
	CTR_ABC_TCO_GRD > 0	**or**
	LV_ERR_TCO_PREL_DET = 1	
then	LV_ERR_TCO_PREL = 1	(without debounce)
else	LV_ERR_TCO_PREL = 0	
endif		

A.35.1.2 Preliminary coolant temperature failure (at ERU / at ES after ERU_to_ES)

Application conditions:

Recurrence:	100 ms
Activation:	LV_CDN_ENA_TCO_PREL = 0
Deactivation:	LV_CDN_ENA_TCO_PREL = 1

Formula section:

If	CTR_ABC_TCO_EL > C_CTR_ABC_THD_TCO_PREL	**or**
	CTR_ABC_TCO_GRD > C_CTR_ABC_THD_TCO_PREL	**or**
	LV_ERR_TCO_STUCK = 1	**or**
	LV_ERR_TCO_STUCK_H = 1	**or**
	LV_ERR_TCO_STUCK_RNG = 1	**or**
	LV_ERR_TCO_PLAUS = 1	
then	LV_ERR_TCO_PREL_DET = 1	(irrevocable for current driving cycle)
endif		

If	CTR_ABC_TCO_EL > 0	**or**
	CTR_ABC_TCO_GRD > 0	**or**
	LV_ERR_TCO_STUCK = 1	**or**
	LV_ERR_TCO_STUCK_H = 1	**or**
	LV_ERR_TCO_STUCK_RNG = 1	**or**
	LV_ERR_TCO_PLAUS = 1	
then	LV_ERR_TCO_PREL = 1	(without debounce)
else	LV_ERR_TCO_PREL = 0	
endif		

A.35.2 Coolant temperature signal range diagnosis (TCO_EL)

FUNCTION DESCRIPTION:

General information:

The purpose of the signal range- or electrical diagnosis is to detect electrical errors in the input circuit of the coolant temperature sensor. The analog sensor input signal is converted into a digital value with use of an A/D converter. All symptoms of the error code are handled by anti-bouncing. Only one symptom can be active at a time.

Description:

Error-symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> Short circuit to  Vbatt or Open Circuit "SCP_OC"      (= SYM_0)
 │  │  └────> Short circuit to Ground "SCG"                        (= SYM_1)
 │  └───────> (not used)                                           (= SYM_2)
 └──────────> (not used)                                           (= SYM_3)
```

Application conditions:

Initialisation:

LV_ERR_TCO_EL	=	*Refer to filtering configuration for the initialisation*
LV_END_DIAG_TCO_EL	=	*Refer to filtering configuration for the initialisation*
LV_CDN_DIAG_TCO_EL	=	*Refer to filtering configuration for the initialisation*
ERR_SYM_TCO_EL	=	*Refer to filtering configuration for the initialisation*

Recurrence:	100 ms	
Activation:	LV_IGK = 1	**and**
	LV_CDN_VB_MIN_DIAG = 1	**and**
	LV_INH_DIAG_TCO_EL = 0	
Deactivation:	Activation condition not true	
	At Deactivation: LV_CDN_DIAG_TCO_EL = 0	

Formula section:

Error detection:

if (1)	VP_TCO[0] < C_VP_TCO_MIN_DIAG_TCO_EL	
	// coolant temperature sensor voltage signal (VP_TCO[0])	
then (1)	*(error detection "Short circuit to GND")*	
	LV_CDN_DIAG_TCO_EL = 1	
	ERR_SYM_TCO_EL = "SYM_1"	
	LV_ERR_TCO_EL = 1	*(after debounce)*
else (1)		
if (2)	VP_TCO[0] > C_VP_TCO_MAX_DIAG_TCO_EL	
	// coolant temperature sensor voltage signal (VP_TCO[0])	

then (2) *(error detection "Short circuit to Vbatt or open load" possible)*

if (3) TIA_THR >= C_TIA_THR_MIN_DIAG_TCO_EL
or
TIA_THR < C_TIA_THR_MIN_DIAG_TCO_EL **and**
T_DIAG_AST > C_T_AST_MIN_DIAG_TCO_EL

then (3) LV_CDN_DIAG_TCO_EL = 1
ERR_SYM_TCO_EL = "SYM_0"
LV_ERR_TCO_EL = 1 *(after debounce)*

else (3) LV_CDN_DIAG_TCO_EL = 0

endif (3)

else (2) LV_CDN_DIAG_TCO_EL = 1
ERR_SYM_TCO_EL = "NO_SYM"
LV_ERR_TCO_EL = 0 *(after rebounce)*

endif (2)

endif (1)

End of diagnosis:

For the calculation of LV_END_DIAG_TCO_EL see "Anti-bounce Algorithm, calculation of the end of diagnosis".

A.35.3 Coolant temperature signal gradient diagnosis (TCO_GRD)

FUNCTION DESCRIPTION:

General information:

The purpose of the signal gradient diagnosis is to detect a not plausible gradient on the measured coolant temperature sensor signal. The monitoring is based on a comparison between the actual measured and the last measure coolant temperature value. The symptom of the error code is handled by anti-bouncing.

Description:

Error-symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> -                                            (= SYM_0)
 │  │  └────> -                                            (= SYM_1)
 │  └───────> -                                            (= SYM_2)
 └──────────> Coolant temperature signal gradient error    (= SYM_3)
```

Application conditions:

Initialisation:

LV_ERR_TCO_GRD = *Refer to filtering configuration for the initialisation*
LV_END_DIAG_TCO_GRD = *Refer to filtering configuration for the initialisation*
LV_CDN_DIAG_TCO_GRD = *Refer to filtering configuration for the initialisation*
ERR_SYM_TCO_GRD = *Refer to filtering configuration for the initialisation*

Initialisation at Reset:

$TCO\_MES\_LIM\_GRD_{(n)} = TCO\_MES\_LIM\_GRD_{(n-1)} = TCO\_MES$

Recurrence: 100 ms

Activation:

If LV_IGK = 1 **and**
LV_CDN_VB_MIN_DIAG = 1 **and**
LV_INH_DIAG_TCO_GRD = 0
then LV_CDN_DIAG_TCO_GRD = 1
else LV_CDN_DIAG_TCO_GRD = 0
$TCO\_MES\_LIM\_GRD_{(n)} = TCO\_MES\_LIM\_GRD_{(n-1)} = TCO\_MES$
endif

Formula section:

Error detection:

If $| TCO\_MES\_LIM\_GRD_{(n-1)} - TCO\_MES | >$
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD
then ERR_SYM_TCO_GRD = “SYM_3”
LV_ERR_TCO_GRD = 1 *(after debounce)*
else ERR_SYM_TCO_GRD = “NO_SYM”
LV_ERR_TCO_GRD = 0 *(after rebounce)*
endif

If $TCO\_MES > TCO\_MES\_LIM\_GRD_{(n-1)} +$
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD
then $TCO\_MES\_LIM\_GRD_{(n)} = TCO\_MES\_LIM\_GRD_{(n-1)} +$
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD
else
if $TCO\_MES < TCO\_MES\_LIM\_GRD_{(n-1)} -$
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD
then $TCO\_MES\_LIM\_GRD_{(n)} = TCO\_MES\_LIM\_GRD_{(n-1)} -$
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD
else $TCO\_MES\_LIM\_GRD_{(n)} = TCO\_MES$
endif
endif

End of diagnosis:

For the calculation of LV_END_DIAG_TCO_GRD see “Anti-bounce Algorithm, calculation of the end of diagnosis”.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VP_TCO_MIN_DIAG_TCO_EL	1	0...7FFFH	0...4.999847412	1.53E-04	[V]
Minimum raw sensor signal voltage for the coolant temperature electrical diagnosis					
C_VP_TCO_MAX_DIAG_TCO_EL	1	0...7FFFH	0...4.999847412	1.53E-04	[V]
Maximum raw sensor signal voltage for the coolant temperature electrical diagnosis					
C_TIA_THR_MIN_DIAG_TCO_EL	1	0...FEH	-48...142.5	0.75	[°C]
Minimum intake air temperature value for the coolant temperature electrical diagnosis					
C_T_AST_MIN_DIAG_TCO_EL	1	0...FFFFH	0...65535	1	[s]
Minimum time delay after start for the coolant temperature electrical diagnosis					
C_TCO_MES_GRD_MAX_DIAG_TCO_GRD	1	0...FEH	0...190.5	0.75	[°C]
Maximum coolant temperature gradient for the coolant temperature gradient diagnosis					
C_CTR_ABC_THD_TCO_PREL	1	0...FFH	0...255	1	[-]
Threshold for anti-bounce counter value for preliminary coolant temperature error detection					

A.36 Coolant temperature sensor diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TCO_EL	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature signal range diagnosis inhibit					
LV_INH_DIAG_TCO_GRD	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature signal gradient diagnosis inhibit					

Input data:

LV_IGK	CTR_ABC_TCO_EL		

FUNCTION DESCRIPTION:

General information:

Task of the coolant temperature sensor diagnosis (Appl. Inc.) is to allow the projects to adapt their specific requirements to generic coolant temperature sensor diagnosis. The setting of the interface output variables have to be provided by the projects.

Application conditions:

Initialisation: see separate chapter

Recurrence: see separate chapter

Activation: at every engine operating state

Deactivation: -

A.36.1 Coolant temperature signal range diagnosis - interface parameter

FUNCTION DESCRIPTION:

General information:

Depending on project specific requirements, the electrical diagnosis of the main coolant temperature sensor can be inhibited by setting of LV_INH_DIAG_TCO_EL. As a default value LV_INH_DIAG_TCO_EL is set to “0”.

Application conditions:

Initialisation:

If Reset **or**

Transition: LV_IGK = 0 -> LV_IGK = 1

then LV_INH_DIAG_TCO_EL = 0

endif

Recurrence: 100 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

Calculation of diagnosis interface parameter:

LV_INH_DIAG_TCO_EL = 0

Error treatment:

Error debounce:

Debounce counter increment: C_ABC_INC_TCO_EL
Debounce counter maximum value: C_ABC_MAX_TCO_EL

Configuration for diagnosic symptoms :

Diagnosis	Symptom Description	Symptom	Filter type
coolant temperature signal range diagnosis	Signal Line short to Battery or Signal Line break	SYM_0	STD
	Signal Line short to GND	SYM_1	
		SYM_2	
TCO_EL		SYM_3	

List of Environmental Data to store in Failure Memory: T_AST_ENVD
TIA_MES
TCO_MES
TCO_ST

*** Application Assistances :**

- TIA Limp Home Calculation is modified
- Fans Management set to MAX (STATE_CFA = 3)
- Evaporative Emission Control - Activation of MIN_PURGE instead of MAX_PURGE
- The following Control Functions are inhibited:
  - Downstream Fuel Trim Regulation
  - Progressive Fuel Cut Off and Rewetting at Part Load
  - Engine Overheating Protection at VS = 0
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation
  - Knock Adaptation
  - Idle Speed Adaptation
  – Catalyst Heating Adaptation
- The following Diagnosis Functions are inhibited:
  - TCO Stuck Signal Diagnosis

- TCO Plausibility Diagnosis
- Differential Tank Pressure sensor Plausibility Diagnosis (Noisy signal)
- Differential Tank Pressure sensor Plausibility Diagnosis (Signal constant)
- MAP/TPS Plausibility Diagnosis
- Evaporative System Diagnosis
- Fuel System Diagnosis
- Catalyst Efficiency Diagnosis
- O2-Sensor Upstream Frequency Diagnosis
- O2-Sensor Upstream Switching Time Diagnosis
- O2-Sensor Downstream Signal Voltage Diagnosis (only SCG)
- O2-Sensor Downstream Switching Time Diagnosis
- O2-Sensor Downstream OBDII puc-end Diagnosis
- Thermostat Diagnosis

A.36.2 Coolant temperature signal gradient diagnosis - interface parameter

FUNCTION DESCRIPTION:

General information:

Depending on project specific requirements, the signal gradient diagnosis of the main coolant temperature sensor can be inhibited by setting of LV_INH_DIAG_TCO_GRD. As a default value LV_INH_DIAG_TCO_GRD is set to “0”. When TCO signal line is short to battery or ground, coolant temperature signal gradient diagnosis is inhibited.

Application conditions:

Initialisation:

If Reset **or**

Transition: LV_IGK = 0 -> LV_IGK = 1

then LV_INH_DIAG_TCO_GRD = 0

endif

Recurrence: 100 ms

Activation: at every engine operating state

Deactivation: -

Formula section:

Calculation of diagnosis interface parameter:

IF CTR_ABC_TCO_EL=0

Then

LV_INH_DIAG_TCO_GRD = 0

Else

LV_INH_DIAG_TCO_GRD=1

End IF

Error treatment:

Error debounce:

Debounce counter increment: C_ABC_INC_TCO_GRD
Debounce counter maximum value: C_ABC_MAX_TCO_GRD

Configuration for diagnosic symptoms :

Diagnosis	Symptom Description	Symptom	Filter type
Coolant temperature signal gradient diagnosis		SYM_0	MEM
		SYM_1	
		SYM_2	
TCO_GRD	Coolant temperature signal gradient error	SYM_3	

List of Environmental Data to store in Failure Memory: T_AST_ENVD
TIA_MES
TCO_MES
TCO_ST

*** Application Assistances :**

- TIA Limp Home Calculation is modified
- Fans Management set to MAX (STATE_CFA = 3)
- Evaporative Emission Control - Activation of MIN_PURGE instead of MAX_PURGE.
- The following Control Functions are inhibited:
  - Downstream Fuel Trim Regulation
  - Engine Overheating Protection at VS = 0
  - Progressive Fuel Cut-Off and Rewetting in PL
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation
  - Knock Adaptation
  - Idle Speed Adaptation
- The following Diagnosis Functions are inhibited:
  - MAP/TPS Plausibility Diagnosis
  - TCO Plausibility Diagnosis
  - TCO Stuck Diagnosis
  - Differential Tank Pressure sensor Plausibility Diagnosis (Noisy signal)
  - Differential Tank Pressure sensor Plausibility Diagnosis (Signal constant)
  - Thermostat Diagnosis
  - Evaporative System Diagnosis
  - Fuel System Diagnosis
  - Catalyst Efficiency Diagnosis
  - O2-Sensor Upstream Frequency Diagnosis
  - O2-Sensor Upstream Switching Time Diagnosis

- O2-Sensor Downstream Signal Voltage Diagnosis (only SCG)
- O2-sensor Downstream Switching Time Diagnosis
- O2-sensor Downstream OBDII PUC-end Diagnosis

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TCO_EL	1	0...FFH	0...255	1	[-]
Debounce counter increment value for the coolant temperature sensor electrical diagnosis					
C_ABC_MAX_TCO_EL	1	1...FFH	1...255	1	[-]
Debounce counter maximum value for the coolant temperature sensor electrical diagnosis					
C_ABC_INC_TCO_GRD	1	0...FFH	0...255	1	[-]
Debounce counter increment value for the coolant temperature sensor gradient diagnosis					
C_ABC_MAX_TCO_GRD	1	1...FFH	1...255	1	[-]
Debounce counter maximum value for the coolant temperature sensor gradient diagnosis					

A.37 Canister purge valve diagnosis (LV_ERR_CPS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CPS	O/V	0...1H	0...1	1	-
Logical value for CPS error					
LV_CDN_DIAG_CPS	O/V	0...1H	0...1	1	-
Diagnosis condition CPS diag.					
ERR_SYM_CPS	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Detected error symptom CPS diag.					
LV_END_DIAG_CPS	O/V	0...1H	0...1	1	-
End of diagnosis CPS diag.					
LV_INH_DIAG_CPS	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition CPS diagnosis					
CDN_DIAG_CPS	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					

Input data:

LV_CDN_VB_OBD1	LV_IGK	CONF_CP	

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis shall be to detect electrical faults as defined by OBD I requirements.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.
The diagnosis condition calculation is done according **“Limited”** duty cycle, thus no definition of the PWM diagnosis windows is neccessary.

Error-symptoms are defined to this diagnosis function as following :

```
| | | └─> Short circuit to Vbatt (SCB)      (= SYM_0)
| | └───> Short circuit to GND (SCG)        (= SYM_1)
| └─────> Open load (OC)                    (=SYM_2)
└───────> -                                 (=SYM_3)
```

Application conditions:

Initialisation: all 0 at LV_IGK 0->1 or Reset

Recurrence: 300ms

Activation: at every engine operating state

Formula section:

IF LV_IGK = 1 **AND**

	LV_CDN_VB_OBD1 = 1	**AND**
	CONF_CP > 0	
THEN	LV_INH_DIAG_CPS = 0	
ELSE	LV_INH_DIAG_CPS = 1	
ENDIF		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_CPS	1	0...FFH	0...255	1	-
Debounce counter increment - CPS diagnosis					
C_ABC_MAX_CPS	1	01...FFH	1...255	1	-
Debounce counter maximum value - CPS diagnosis					

A.37.1 Canister purge valve diagnosis (Appl. Inc.)

Input data:

LV_ERR_CPS			

Configuration for diagnositc symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Canister purge valve	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
CPS		SYM_3	

List of Environmental Data to store in failure memory: TAM
CPPWM_ENVD
DTP_ENVD
CP_STATE

Application Assistances :

- Evaporative Emission Control - Activation of NO_PURGE
- Set State Torque Intervention to max. value (Passive Status)
- The following Control Functions are inhibited:
  - Downstream Fuel Trim Regulation
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation
  - Idle Speed Adaptation
  - Segment Time Adaptation
- The following Diagnosis Functions are inhibited:
  - ISA Plausibility Diagnosis
  - Differential Tank Pressure Sensor Noisy Signal Diagnosis
  - Differential Tank Pressure Sensor Constant Signal Diagnosis
  - O2-Sensor Downstream Voltage Excursion Diagnosis
  - O2-Sensor Upstream Signal Voltage Diagnosis (only SCG)
  - O2-Sensor Downstream Signal Voltage Diagnosis
  - Evaporative System Diagnosis
  - O2-Sensor Upstream Switching Time Diagnosis
  - O2-Sensor Upstream Plausibility Diagnosis
  - O2-Sensor Upstream Frequency Diagnosis
  - Fuel System Diagnosis
  - Lambda Limit Diagnosis
  - Catalyst Efficiency Diagnosis
  - O2 Sensor Downstream Switching Time Diagnosis
  - O2 Sensor Downstream OBD2 PUC-end Diagnosis
  - MAP/TPS rationality check

A.38 Injection valves diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_DIAG_IV	O/V	0...FFH	0...255	1	-
Error pattern without debouncing for all injectors					
LV_ERR_IV[NC_CYL_NR]	-	0...1H	0...1	1	-
Error flag for injector [x] after debouncing					
STATE_ERR_IV	O/V	0...FFH	0...255	1	-
Error pattern after debouncing for all injectors					
LV_CND_DIAG_IV [NC_CYL_NR]	V	0...1H	0...1	1	-
Diagnosis condition for injector [x]					
LV_END_DIAG_IV [NC_CYL_NR]	V	0...1H	0...1	1	-
End of diagnosis for injector [x]					
ERR_DIAG_IV [NC_CYL_NR]	-	0H 1H 2H 4H	NO_SYM SYM_SCB SYM_SCG SYM_OC	1	-
Raw value of error symptom for injector [x]					
ERR_SYM_IV [NC_CYL_NR]	O/V	0H 1H 2H 4H	NO_SYM SYM_SCB SYM_SCG SYM_OC	1	-
Error symptom for injector [x] filterd with CDN_DIAG_IV [NC_CYL_NR]					
CDN_DIAG_IV [NC_CYL_NR]	V	0H 1H 2H 3H 4H 5H 6H 7H	NO_SYM SYM_SCB SYM_SCG SYM_SCB_SCG SYM_OC SYM_SCB_OC SYM_SCG_OC SYM_ALL	1	-
Diagnosis condition for each symptom of injector [x]					

Input data:

LV_IGK	LV_INH_DIAG_IV	PREV_STATE_IV	NC_CYL_NR
LV_CDN_VB_OBD1	LV_ST_END	LV_FL	LV_ES
N_32	LV_DC		

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.
STATE_ERR_IV is the general error bit after debouncing for limp-home reactions without entry in Error-memory.
It has to be assured that all cylinders had been injected in the cycle before.

All symptoms of the current error code are handeled by anti-bouncing.
We assume that only one symptom of an error code can be active at the same time.

Description:

The error detection is effected via the ECU hardware.

A valid diagnosis of all failures can only be detected when the injector was operated at least one time since the last changes and the filter times are observed. A short circuit to battery (SCB) and an open circuit (OC) can be detected only when the injector is “ON”. A short circuit to ground (SCG) can be detected only when the injector is “OFF”. Otherwise the diagnosis has to be suppressed (CDN_DIAG_IV [x] = 0). This is managed within the “Diagnosis condition calculation.

Application conditions:

Activation : every engine state

Deactivation: -

Initialization: at reset and at transition LV_IGK 0 -> 1:
all outputs (except LV_END_DIAG_IV) = 0

at reset and at transition LV_DC 0 -> 1:

LV_END_DIAG_IV = 0

Recurrence: 300 ms

Formula section:

(1) FOR x = 0 **TO** (NC_CYL_NR-1) **DO**: (Diagnosis condition calculation)

CDN_DIAG_IV [x] = 0

(2) IF LV_INH_DIAG_IV = 0
and
LV_IGK = 1
and
LV_CDN_VB_OBD1 = 1
and
LV_ES = 0

(2) THEN

(3) IF Bit x of PREV_STATE_IV = 1
and
if hardware valid flag is set

(3)THEN bit 0 of CDN_DIAG_IV [x] = 1 *(SCB can be detected))*

(3) ENDIF

(4) IF LV_FL = 0
or
N_32 < C_N_CDN_DIAG_IV_FL_THD
and
if hardware valid flag is set

(4) THEN bit 1 of CDN_DIAG_IV [x] = 1 *(SCG can be detected)*

(4) ENDIF

(5) IF (LV_FL = 0 or N_32 < C_N_CDN_DIAG_IV_FL_THD)
and
Bit x of PREV_STATE_IV = 1
and
if hardware valid flag is set

(5) THEN bit 2 of CDN_DIAG_IV [x] = 1 *(OC can be detected)*

(5) ENDIF

(2) ENDIF

Error Symptom calculation (raw value from I/O SW)

ERR_DIAG_IV [x] = 0

(6) IF "Error symptom SCB detected by HW"

(6) THEN bit 0 of ERR_DIAG_IV [x] = 1

(6) ENDIF

(7) IF "Error symptom SCG detected by HW"

(7) THEN bit 1 of ERR_DIAG_IV [x] = 1

(7) ENDIF

(8) IF "Error symptom OC detected by HW"

(8)THEN bit 2 of ERR_DIAG_IV [x] = 1

(8) ENDIF

ACTION_ERRM_FilterMulticondition (IV[x], CDN_DIAG_IV[x], ERR_DIAG_IV[x], LV_END_DIAG_IV[x], LV_ERR_IV[x], LV_CDN_DIAG_IV[x], ERR_SYM_IV[x])

Remark: For failure debouncing and error management treatment the algorithm Multi-condition debounce algorithm (part of Error Management AGGR) is used with the parameters CDN_DIAG_IV[x] and ERR_DIAG_IV[x].

This algorithm determines:

ERR_SYM_IV [x] (= raw value ERR_DIAG_IV [x] filtered with CDN_DIAG_IV [x])

and

LV_ERR_IV [x] (Error flag for debounced error of Injector [x])

and

LV_CDN_DIAG_IV [x]

and

LV_END_DIAG_IV [x]

(1) ENDFOR

(2) FOR x = 0 TO (NC_CYL_NR-1) DO:

Calculation of STATE_DIAG_IV:

(9) IF ERR_SYM_IV [x] <> 0
(9) THEN Bit x of STATE_DIAG_IV = 1
(9) ELSE Bit x of STATE_DIAG_IV = 0
(9) ENDIF

Calculation of STATE_ERR_IV:

Bit x of STATE_ERR_IV = LV_ERR_IV [x]

(2) ENDFOR

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_IV	1	0...FFH	0...255	1	-
Anti-bounce increment value					
C_ABC_MAX_IV	1	1...FFH	1...255	1	-
Anti-bounce maximum value					
C_N_CDN_DIAG_IV_FL_THD	1	0 ... FF	0 ... 8160	32	rpm
Threshold disable diagnosis at full load					

A.39 Injection valve diagnosis (Application Incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_IV	V/O	0..1H	0..1	1	-
Inhibition condition for the IV diagnostic					

Input data:

LV_IGK	INH_SWI_IV	N	INH_IV_IGC
LV_OFF_IV_MON	INH_IV_MIS	LV_PUC	INH_IV_TPS_AD
INH_IV_MIS_GEN			

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_IV deactivates (freezes) the power stage diagnostic of all injection valves.

Application conditions:

Activation:

Deactivation:

Initialisation: At reset LV_INH_DIAG_IV = 0

Recurrence: 300 ms

Formula section:

```
IF          INH_SWI_IV = 0
      and   LV_IGK = 1
      and   INH_IV_IGC = 0
      and   LV_OFF_IV_MON = 0
      and   INH_IV_MIS = 0
      and   LV_PUC = 0
      and   INH_IV_TPS_AD = 0
      and   INH_IV_MIS_GEN = 0
      and   N > C_N_DIAG_IV_MIN]

THEN

      LV_INH_DIAG_IV = 0

ELSE

      LV_INH_DIAG_IV = 1
ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DIAG_IV_MIN	1	0 .. 1FE0 H	0 .. 8160	1	rpm
Minimum N threshold to enable Injectors power stage diagnosis. Advised value 320 rpm					

A.39.1 Application assistances

Configuration or calibration data

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_IV_0	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
TI_1_0_ENVD_H
TI_1_0_ENVD_L
TI_2_MES_0_ENVD_H

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_IV_1	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
TI_1_1_ENVD_H
TI_1_1_ENVD_L
TI_2_MES_1_ENVD_H

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_IV_2	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
TI_1_2_ENVD_H
TI_1_2_ENVD_L
TI_2_MES_2_ENVD_H

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_IV_3	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
TI_1_3_ENVD_H
TI_1_3_ENVD_L
TI_2_MES_3_ENVD_H

- **Application Assistances :**
- Evaporative Emission Control - Activation of MIN_PURGE instead of MAX_PURGE
- Set State Torque Intervention to Max. State (Passive State)
- The following Control Functions are inhibited:
  - Lambda Control
  - Downstream Fuel Trim Regulation (indirectly)
  - Anti-Jerk Function
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation (indirectly)
  - Knock Adaptation
  - Idle Speed Adaptation
  - Segment Time Adaptation
- The following Diagnosis Functions are inhibited:
  - Differential Tank Pressure sensor Plausibility Diagnosis (Noisy signal)
  - Differential Tank Pressure sensor Plausibility Diagnosis (Signal constant)
  - MAF/TPS-Plausibility Diagnosis (indirectly)
  - O2-Sensor upstream Voltage Excursion Diagnosis (indirectly)
  - O2-Sensor upstream Signal Voltage Diagnosis (indirectly)
  - O2-Sensor Upstream Plausibility Diagnosis
  - O2-Sensor downstream Signal Voltage Diagnosis (indirectly. only SCG)
  - Lambda Bank Control Limit Diagnosis (indirectly)
  - O2-Sensor downstream Voltage Excursion Diagnosis
  - Evaporative System Diagnosis
  - Fuel System Diagnosis (indirectly)
  - Catalyst Efficiency Diagnosis (indirectly)
  - O2-Sensor upstream Frequency Diagnosis (indirectly)
  - O2-Sensor upstream Switching Time Diagnosis (indirectly)
  - O2-Sensor Downstream Switching Time Diagnosis
  - O2-Sensor Downstream OBDII puc-end Diagnosis
  - Thermostat Diagnosis

A.40 Knock Sensor diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_KNKS_1	V/S/O	0...1H	0...1	1	-
Boolean for error currently present on knock signal acquisition					
LV_ERR_KNKS_1_RNG	V/S	0...1H	0...1	1	-
Boolean for error currently present on knock signal acquisition : noise level too low or too high					
LV_ERR_KNKS_1_DYW	V/S	0...1H	0...1	1	-
Boolean for error currently present on knock signal bandwith (Master algorithm)					
LV_ERR_KNKS_1_SLAVE	V/S	0...1H	0...1	1	-
Boolean for error currently present on knock signal bandwith (Slave algorithm)					
LV_KNKS_DIAG_ENA	V	0...FFH	0...1	1	-
Boolean for knock sensor diagnosis enabled					
CYCNR_DYW_KNKS_DIAG	V	0...FFH	0...255	1	-
Cycle counter to detect knock signal bandwith error (Master algorithm)					
V_INT_KNKS_DIAG	V	1...FFFFH	0.0049...319.95	0.00489	V
Integration of DYW_KNKS_DIAG_2 + DYW_KNKS_DIAG_3					
CYCNR_INT_KNKS_DIAG	V	0...FFH	0...255	1	-
Cycle counter to detect knock signal bandwith error (Slave algorithm)					
ERR_SYM_KNKS_1	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom knock sensor diagnosis					
LV_CDN_DIAG_KNKS_1	V/O	0...1H	0...1	1	-
Diagnosis condition of the diagnosis instance KNKS_1					

Input data:

LV_ES	LV_ST	LV_PU	LV_PUC
MAF	TCO	N_32	LV_ERR_CAM_IN[NC_NR_CAM_CBK]
NL_x	LV_ERR_CRK	ECU_DIAG_SF	IP_MAF_MIN_KNK__N_32__TCO
KNKS_x			

FUNCTION DESCRIPTION:

General information:

The purpose is to observe the analog input signal (KNKS) from the ATM40 device to the microcontroller.

The signal is checked continuously by a range check of the signal and with two algorithms, which observe the signal bandwidth. A Slave (BBINT) and a Master (BBSW) algorithm is used. Both algorithms have to show the same state to increment the ABC counter.

Range check:

The absolute noise value of the ATM40 device is checked if it is inside the normal operating range, C_NL_MIN_DIAG < NL_x < C_NL_MAX_DIAG. An out of range error can be detected in this way.

The Master algorithm (BBSW):

The bandwidth (│NL - KNKS│) of the signal is evaluated for cylinder 2 and 3. In case of short to ground, short to battery or open load the bandwidth is smaller than during normal operation. To monitor the bandwidth of the signal the magnitude is compared with a calibratable threshold C_DYW_MIN_KNKS_DIAG. If the bandwidth magnitude of cylinder 2 and 3 does not exceed C_DYW_MIN_KNKS_DIAG the cycle counter CYCNR_DYW_KNKS_DIAG is incremented by 1 every 360°CRK. This cycle counter is reset as soon as the KNKS value of cylinder 2 or 3 exceeds the threshold C_DYW_MIN_KNKS_DIAG. If the cycle counter reaches C_CYCNR_KNKS_DIAG, a knock sensor failure can be detected if the slave algorithm confirms the failure also. After the failure has been detected the counter CYCNR_DYW_KNKS_DIAG is decremented by 5 if the bandwidth of cyl. 2 or 3 exceeds the threshold C_DYW_MIN_KNKS_DIAG and the state of the ‘Slave algorithm’ is void.

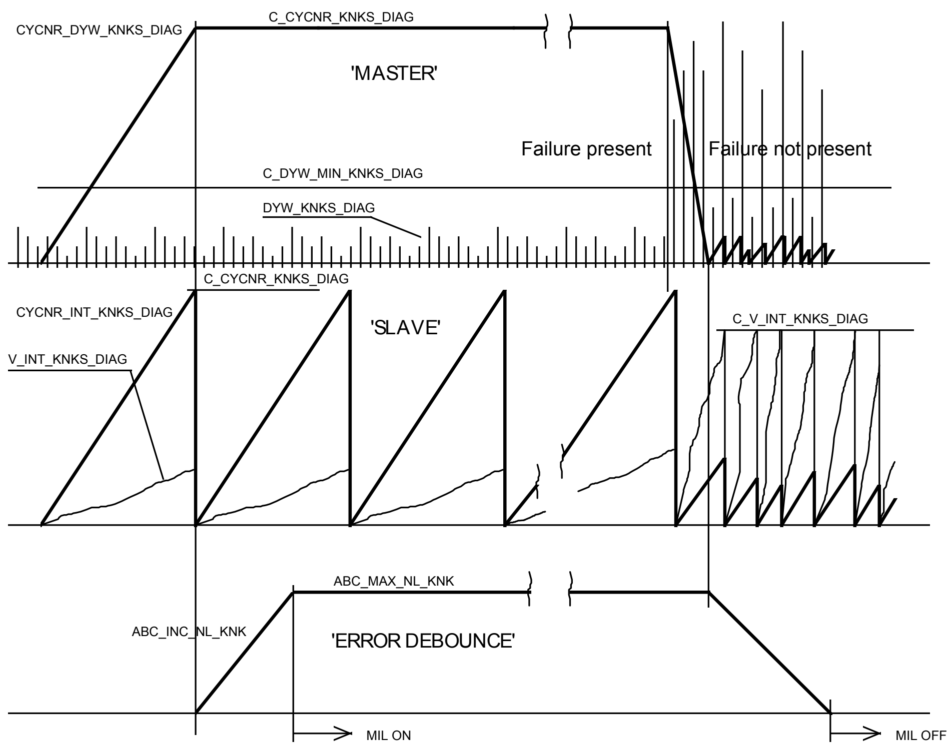
The Slave algorithm (BBINT):

This algorithm uses the bandwidth of the signal as well. The bandwidths of cylinder 2 and 3 are accumulated via an integration method. The accumulated voltage value is compared every revolution with the calibratable threshold C_V_INT_KNKS_DIAG. As soon as the threshold C_V_INT_KNKS_DIAG is reached the cycle counter CYCNR_INT_KNKS_DIAG (incremented every 360°CRK) is reset. If the cycle counter CYCNR_INT_KNKS_DIAG reaches the threshold C_CYCNR_KNKS_DIAG before the voltage threshold is reached the failure is confirmed by the slave algorithm.

If both algorithms detect a failure, the debounce starts (ABC counter is incremented). In this way a unplausible signal can be detected.

After the detection of the failure (ABC = ABC_MAX_NL_KNK) the master algorithm only is used to observe the bandwidth of the signal. If at least one value of the bandwidths exceeds the threshold C_DYW_MIN_KNKS_DIAG the counter CYCNR_DYW_KNKS_DIAG is decremented by 5 every 360°CRK. As soon as the counter CYCNR_DYW_KNKS_DIAG reaches 0 the ABC counter is decremented by 1 every 720°CRK.

Description:



Application conditions:

Recurrence: Crankshaft Synchronous (see details in formula section)

Activation:

- LV_ES = 0 **and** LV_ST = 0 **and** LV_PU = 0 **and** LV_PUC = 0 **and**
- [MAF > IP_MAF_MIN_KNK__N_32__TCO **and**
  MAF > C_MAF_MIN_KNKS_DIAG] **and**
- N_32 > C_N_MIN_KNKS_DIAG **and**
- LV_ERR_CRK = 0 **and**
- LV_ERR_CAM_IN_i = 0 **and**
- No communication failure (SPI bus) is present
  ('ATIC62_SPI_ERR' bit of ECU_DIAG_SF = 0)
  => LV_KNKS_DIAG_ENA **=** 1
  => LV_CDN_DIAG_KNKS_1 **=** 1

Deactivation:

- one of the conditions above not fulfilled
  => LV_KNKS_DIAG_ENA **=** 0
  => LV_CDN_DIAG_KNKS_1 **=** 0

Formula section:

- Every 180° crankshaft :

If NL_x < C_NL_MIN_DIAG
or NL_x > C_NL_MAX_DIAG
then LV_ERR_KNKS_1_RNG = 1
ERR_SYM_KNKS_1 = SYM_0 # Signal out of range
else LV_ERR_KNKS_1_RNG = 0

- Every 360° crankshaft :

Slave algorithm (BBINT) :

$V\_INT\_KNKS\_DIAG_i = V\_INT\_KNKS\_DIAG_{i-1} + DYW\_KNKS\_DIAG\_2 + DYW\_KNKS\_DIAG\_3$

With : $DYW\_KNKS\_DIAG\_2 = |NL\_2 - KNKS\_2|$

$DYW\_KNKS\_DIAG\_3 = |NL\_2 - KNKS\_3|$

$CYCNR\_INT\_KNKS\_DIAG_i = CYCNR\_INT\_KNKS\_DIAG_{i-1} + 1$

If V_INT_KNKS_DIAG >= C_V_INT_KNKS_DIAG
then **If** CYCNR_INT_KNKS_DIAG < C_CYCNR_KNKS_DIAG
then LV_ERR_KNKS_1_SLAVE = 0
and reset (V_INT_KNKS_DIAG, CYCNR_INT_KNKS_DIAG)
else LV_ERR_KNKS_1_SLAVE = 1 -> SLAVE Error is detected
and reset (V_INT_KNKS_DIAG, CYCNR_INT_KNKS_DIAG)

Master algorithm (BBSW) :

If [|KNKS_2 - NL_2| > C_DYW_MIN_KNKS_DIAG **or**
|KNKS_3 - NL_3| > C_DYW_MIN_KNKS_DIAG]

then CYCNR_DYW_KNKS_DIAG = 0

else CYCNR_DYW_KNKS_DIAG = CYCNR_DYW_KNKS_DIAG + 1

If CYCNR_DYW_KNKS_DIAG >= C_CYCNR_KNKS_DIAG
and LV_ERR_KNKS_1_SLAVE **=** 1 (SLAVE Error is detected)

then LV_ERR_KNKS_1_DYW **=** 1 -> MASTER Error is detected
if ERR_SYM_KNKS_1 = NO_SYM
then ERR_SYM_KNKS_1 = SYM_1 # Unplausible signal

Error debounce :

If LV_ERR_KNKS_1_RNG **=** 1

or LV_ERR_KNKS_1_DYW **=** 1 (MASTER_ERROR is detected)

then $ABC_i = ABC_{i-1}$ + ABC_INC_NL_KNK with: i =720°CRK

If ABC >= ABC_MAX_NL_KNK

then LV_ERR_KNKS_1 = 1 (Knock sensor failure dectected)

If after the failure has been debounced and |KNKS_2 - NL_2| > C_DYW_MIN_KNKS_DIAG or |KNKS_3 - NL_3| > C_DYW_MIN_KNKS_DIAG, then the cycle counter CYCNR_DYW_KNKS_DIAG is decremented by 5. As soon as the cycle counter equals 0 and no LV_ERR_KNKS_1_RNG currently present the ABC - Counter is decremented by 1.

If LV_ERR_KNKS_1_DYW = 0 **and** LV_ERR_KNKS_1_RNG = 0

then ERR_SYM_KNKS_1 = NO_SYM

Formula section:

** Debounce :*

Increment value for debounce counter : C_ABC_INC_NL_KNK
Maximum value for debounce counter : C_ABC_MAX_NL_KNK

Emergency operation :

** Limp Home : In the case of a knock sensor error LV_ERR_KNKS_1 = 1 and consequently KNK_CTL_DIS = 1 indicating a knock control relavant error; this in turn triggers the application of the limp home ignition angle retard IGA_KNK_CTL_DIS in the knock control module.*

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_NL_MIN_DIAG	1	0...FFH	0...5	0.02	V
Noise level threshold for knock acquisition failure detected.					
C_NL_MAX_DIAG	1	0...FFH	0...5	0.02	V
Noise level threshold for knock acquisition failure detected.					
C_MAF_MIN_KNKS_DIAG	1	0...FFFFH	0...1389	0.0212	mg/stk
MAF threshold to enable KNKS diagnosis					
C_N_MIN_KNKS_DIAG	1	20...FFH	1024...8160	32	rpm
N_32 threshold to enable KNKS diagnosis					
C_DYW_MIN_KNKS_DIAG	1	1...17FAH	0.0048...29.996	0.00489	V
Bandwidth threshold for failure detection knock sensor ‘Master Algorithm’ (BBSW)					
C_V_INT_KNKS_DIAG	1	1...FFFFH	0.0048...319.95	0.00489	V
Voltage threshold for failure detection knock sensor ‘Slave Algorithm’ (BBINT)					
C_CYCNR_KNKS_DIAG	1	1...FFH	1...255	1	seg
Number of diagnosis cycles to detect a sensor failure					
C_ABC_INC_NL_KNK	1	0...FFH	0...255	1	-
Anti - bounce counter increment for knock acquisition diagnosis.					
C_ABC_MAX_NL_KNK	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter for knock acquisition diagnosis.					

A.40.1 Application Incidences

Configuration or calibration data

Diagnostic	Symptom Description	Symptom	Filter type
Knock Sensor 1 Circuit Malfunction KNKS_1	Signal out of range	SYM_0	*STD*
	Invalid Signal	SYM_1	
		SYM_2	
		SYM_3	

Enviromental data is defined in the table of failures

- **Application Incidence :**
- The following Control Functions are inhibited:
  - Knock Control
- The following Adaptation Functions are inhibited:
  - Knock Adaptation
- Activation of Knock Limp Home Value Calculation

A.41 Air condition compressor relay diagnosis (LV_ERR_RLY_ACCOUT)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_RLY_ACCOUT	O/V	0...01H	0...1	1	-
Boolean for error currently present on RLY_ACCOUT command signal.					
LV_CDN_DIAG_RLY_ACCOUT	O/V	0...01H	0...1	1	-
Diagnosis condition RLY_ACCOUT diagnosis					
ERR_SYM_RLY_ACCOUT	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom RLY_ACCOUT diagnosis					
LV_END_DIAG_RLY_ACCOUT	O/V	0...01H	0...1	1	-
End of diagnosis RLY_ACCOUT diagnosis					
LV_INH_DIAG_RLY_ACCOUT	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition RLY_ACCOUT diagnosis					
CDN_DIAG_RLY_ACCOUT	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					

Input data:

LV_IGK	LV_CDN_VB_OBD1	CONF_ACC	LV_RLY_MAIN_EXT_ADJ
C_T_DLY_RLY_MAIN_DIAG	LV_DC		

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the signal from the driver which controls the air condition relay signal.

The diagnosis is only active if “Vehicle with ACC” is learnt.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.

Error-symptoms are defined to this diagnosis function as following :

```
|_|_|_|_|
 |  |  |  └─> Short circuit to Vbatt (SCB)      (= SYM_0)
 |  |  └───> Short circuit to GND (SCG)         (= SYM_1)
 |  └─────> Open load (OL)                      (= SYM_2)
 └───────> -                                    (= SYM_3)
```

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at reset

Recurrence: 2s

Activation: CONF_ACC = 1

Formula section:

IF LV_IGK = 1
AND LV_CDN_VB_OBD1 = 1
AND C_T_DLY_RLY_MAIN_DIAG has run out after LV_RLY_MAIN_EXT_ADJ = 1-> 0
AND LV_RLY_MAIN_EXT_ADJ = 0
THEN LV_INH_DIAG_RLY_ACCOUT = 0
ELSE LV_INH_DIAG_RLY_ACCOUT = 1
ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_RLY_ACCOUT	1	0...FFH	0...255	1	-
Debounce counter increment RLY_ACCOUT diagnosis					
C_ABC_MAX_RLY_ACCOUT	1	01...FFH	1...255	1	-
Debounce counter maximum value – RLY_ACCOUT diagnosis					

A.41.1 Application incidences for AC compressor relay diagosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
AC relay diagnosis	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
RLY_ACCOUT		SYM_3	

List of Environmental Data to store in Failure Memory: TIA

TCO_ST

LV_RLY_ACCOUT_CTRL

V_IGK

A.42 ECU-selftest

A.42.1 ECU selftest

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ECU	V/S	0...01H	0...1	1	-
Boolean for error currently present at ECU					
ERR_SYM_ECU	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom ECU-selftest Diagnosis					
LV_CDN_DIAG_ECU	V/O	0...01H	0...1	1	-
Diagnosis condition for ECU-selftest diagnosis					
LV_END_DIAG_ECU	V/O	0...01H	0...1	1	-
End of diagnosis ECU-selftest diagnosis					
ECU_DIAG_SF_HB	V/S	0...FFH	0...255	1	-
ECU-Error Byte (Low Byte)					
ECU_DIAG_SF_LB	V/S	0...FFH	0...255	1	-
ECU-Error Byte (High Byte)					
ECU_DIAG_SF	V/O	0...FFFFH	0...65535	1	-
ECU-Error Byte (Word) 0h NO_ERR (no failure) 1h WDT_ERR (Watchdog failure) 2h RAM_ERR (RAM failure) 4h REF_ERR (Reference between ECU SW and Cal.) 8h CKS_SB_ERR (Checksum failure of Siemens boot) 10h CKS_CB_ERR (Checksum failure of customer boot) 20h CKS_ECU_ERR (Checksum failure of ECU software) 40h CKS_CAL_ERR (Checksum failure of Cal. data) 100h ATIC39_SPI_ERR (ATIC39 SPI line failure) 200h ATIC62_SPI_ERR (ATIC62 SPI line failure)					
ECU_DIAG_SF_SAVE	V/S	0...FFFFH	0...65535	1	-
ECU-Error Byte (Word) for saving at NVMY					
ABC_SPI_ATIC39	-	0...FFH	0...255	1	-
Anti bounce counter for ATIC39 SPI line diagnosis					
ABC_SPI_ATIC62	-	0...FFH	0...255	1	-
Anti bounce counter for ATIC62 SPI line diagnosis					

Input data:

LV_PRDR_ACT	STATE_DR_MU_MC	ERR_COD_MC_SAVE	NC_ETC_CONF

FUNCTION DESCRIPTION:

General information:

The purpose of the test is the check of ram areas and communication connections. In the stored ECU-failure bytes the error location is specified more detailed. Most of the checks are performed during system start-up.

ECU selftest error (= SYM_0)

Application conditions:

Initialization :

At first ECU power up, Loss of NVMY or NVMY reset

ECU_DIAG_SF_SAVE = 0

At reset,

ECU_DIAG_SF = 0

ABC_SPI_ATIC39 = 0

ABC_SPI_ATIC62 = 0

ECU_DIAG_SF_SAVE = restored from NVMY

At FMY clear,

ECU_DIAG_SF_SAVE = 0

Before saving ECU_DIAG_SF_SAVE in NVMY,

ECU_DIAG_SF_SAVE = ECU_DIAG_SF_SAVE **or** ECU_DIAG_SF ...bitwise

Fomular section :

a) Watch-dog failure detection at reset

If 'Software watchdog reset status' of ECU_RST_STATUS_HW is set

then Set 'WDT_ERR' bit of ECU_DIAG_SF

endif

b) Coherence failure detection between ECU software and calibration at reset

If reference number of ECU software <> reference number of calibration

then Set 'REF_ERR' bit of ECU_DIAG_SF

endif

c) RAM failure detection at reset (for non-ETC system)

If RAM failure is detected

then Set 'RAM_ERR' bit of ECU_DIAG_SF

endif

In non-ETC system, RAM failure detection at reset should be executed before application system detection (CCP) to avoid CCP RAM failure detection at ECU without CCP RAM.

d) SPI communication diagnosis every 10 ms

```
% ATIC39 SPI line diagnosis %
If      (NC_ETC_CONF = 1 and LV_PRDR_ACT = 0 and STATE_DR_MU_MC = ENABLE)
    or  NC_ETC_CONF = 0
Then    If      ATIC39 SPI communication line is failed
        then    ABC_SPI_ATIC39 = ABC_SPI_ATIC39 + C_ABC_INC_SPI
                If      ABC_SPI_ATIC39 >= C_ABC_MAX_SPI
                then    ABC_SPI_ATIC39 = C_ABC_MAX_SPI
                        Set 'ATIC39_SPI_ERR' bit of ECU_DIAG_SF
                endif
        else    ABC_SPI_ATIC39 = ABC_SPI_ATIC39 - 1
                If      ABC_SPI_ATIC39 = 0
                then    Clear 'ATIC39_SPI_ERR' bit of ECU_DIAG_SF
                endif
        endif
endif
```

e) SPI communication diagnosis every 10 ms

```
% ATIC62 SPI line diagnosis(Knock sensor) %
If      ATIC39 SPI communication line is failed
then    ABC_SPI_ATIC62 = ABC_SPI_ATIC62 + C_ABC_INC_SPI
        If      ABC_SPI_ATIC62 >= C_ABC_MAX_SPI
        then    ABC_SPI_ATIC62 = C_ABC_MAX_SPI
                Set 'ATIC62_SPI_ERR' bit of ECU_DIAG_SF
        endif
else    ABC_SPI_ATIC62 = ABC_SPI_ATIC62 - 1
        If      ABC_SPI_ATIC62 = 0
        then    Clear 'ATIC62_SPI_ERR' bit of ECU_DIAG_SF
        endif
endif
```

f) RAM failure detection at 10 ms (for ETC system)

```
If      ERR_COD_MC_SAVE = 0x0C
then    Set 'RAM_ERR' bit of ECU_DIAG_SF
```

endif

g) Checksum diagnosis every 100 ms

If Each checksum of customer boot, ECU software and calibration data area is not correct

then Set 'CKS_xxx_ERR' bit of ECU_DIAG_SF % xxx ; customer boot, ECU software and calibration data according to area %

endif

RAM failure diagnosis is done only once and diagnosis result is not reversible.

Detection :

Main ECU diagnosis accoding to ECU_DIAG_SF

Recurrence : 10 ms

If ECU_DIAG_SF <> 0

then LV_CDN_DIAG_ECU = 1

ERR_SYM_ECU = SYM_0

LV_END_DIAG_ECU = 1

LV_ERR_ECU = 1

else LV_CDN_DIAG_ECU = 1

ERR_SYM_ECU = NO_SYM

LV_END_DIAG_ECU = 1

LV_ERR_ECU = 0

Emergency operation :

* Limp Home : **none**

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_SPI	1	0...FFH	0...255	1	-
Anti-bounce counter increment for SPI communication diagnosis.					
C_ABC_MAX_SPI	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for SPI communicaiagnosis.					

A.42.2 Check of Vehicle Identification Number (VIN)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VIN_DIAG	V/S	0...01H	0...1	1	-
Boolean for VIN currently not present at ECU					
ERR_SYM_VIN_DIAG	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom VIN-Diagnosis					

Input data:

LV_IGK			

Import actions:

ACTION_ERRM_ClrInfoByDtc (IN <DtcIdentifier>,IN <LevelOfDtc>,OUT <ResultClrInfo>)
ACTION_ERRM_IntmClr(IN < PRM_IDX_DYN >)

FUNCTION DESCRIPTION:

General information:

The purpose of this diagnosis is to check whether VIN is programmed in ECU boot area or not.

The error treatment for this diagnosis allows that the error is removed from dynamic error memory immediately after the VIN was programmed.

Description:

Error-symptoms are defined to this diagnosis function as following:

VIN not programmed (= SYM_0)

Application conditions:

Recurrence: 1s

Activation: LV_IGK = 1 **and**

LC_ENA_VIN_DIAG = 1

Formula section:

If VIN is not programmed

Then ERR_SYM_VIN_DIAG = SYM_0
LV_ERR_VIN_DIAG = 1

Else **If** LV_ERR_VIN_DIAG = 1

Then ACTION_ERRM_IntmClr (
IN < VIN_DIAG >) **and**
ACTION_ERRM_ClrInfoByDtc (

IN < 0630>,

IN < LAW>,

OUT < ResultClrInfo>) **and**

LV_ERR_VIN_DIAG = 0 **and**

ERR_SYM_VIN_DIAG = NO_SYM

Else LV_ERR_VIN_DIAG = 0 **and**

ERR_SYM_VIN_DIAG = NO_SYM

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ENA_VIN_DIAG	1	0...1H	0...1	1	-
Logical constant to enable VIN Diagnosis					

A.42.3 ECU-selftest (Appl. inc.)

A.42.3.1 ECU-selftest

Configuration for diagnostic symptoms:

Diagnosis	Symptom Description	Symptom	Filter type
Internal Control Module Read Only Memory (ROM) Error	ECU selftest error	SYM_0	STD
		SYM_1	
		SYM_2	
ECU		SYM_3	

Environmental Data are defined in the table of failures specification.

*** Application Incidence** : At communication error with ATIC39 the concerned power stage diagnosis is inhibited. Undefined system reactions could occur.

* **Error Type of ECU-error low (ECU_DIAG_SF_LB)**

```
-> WDT_ERR        (Watchdog failure)
-> RAM_ERR        (RAM failure)
-> REF_ERR        (Reference between ECU SW and Cal.)
-> CKS_SB_ERR     (Checksum failure of Siemens boot)
-> CKS_CB_ERR     (Checksum failure of Customer boot)
-> CKS_ECU_ERR    (Checksum failure of ECU SW)
-> CKS_CAL_ERR    (Checksum failure of Cal. data)
-> Unused
```

* **Error Type of ECU-error high (ECU_DIAG_SF_HB)**:

```
-> ATIC39_SPI_ERR   (ATIC39 SPI line failure)
-> ATIC62_SPI_ERR   (ATIC62 SPI line failure)
-> Unused
-> Unused
-> Unused
```

A.42.3.2 Check of Vehicle Identification number (VIN)

Configuration for diagnostic symptoms:

Diagnosis	Symptom Description	Symptom	Filter type
Vehicle Identification Number check	VIN not programmed	SYM_0	STD
		SYM_1	
		SYM_2	
VIN_DIAG		SYM_3	

Environmental Data are defined in the table of failures specification.

A.43 Variable intake manifold diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VIM	V/O	0...01H	0...1	1	-
Boolean for error currently present on VIM signal					
LV_CDN_DIAG_VIM	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis condition VIM diagnosis					
ERR_SYM_VIM	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Detected error symptom VIM diagnosis					
LV_END_DIAG_VIM	O/V	0 ... 1H	0 ... 1	1	-
End of diagnosis VIM diagnosis					
LV_INH_DIAG_VIM	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition VIM diagnosis					
CDN_DIAG_VIM	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					
ERR_SYM_VIM_AD	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error Symptom for VIM flap adaptation failure					
ERR_SYM_VIM_MEC_LONG	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error Symptom for VIM mechanical failure long position					
ERR_SYM_VIM_MEC_SHO	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error Symptom for VIM mechanical failure short position					
LV_CDN_DIAG_VIM_AD	V/O	0...1H	0...1	1	[-]
Diagnosis active flag for VIM adaptation failure					
LV_CDN_DIAG_VIM_MEC_LONG	V/O	0...1H	0...1	1	[-]
Diagnosis active flag for VIM mechanical failure long position (stuck check)					
LV_CDN_DIAG_VIM_MEC_SHO	V/O	0...1H	0...1	1	[-]
Diagnosis active flag for VIM mechanical failure short position (stuck check)					
LV_END_DIAG_VIM_AD	V/O	0...1H	0...1	1	[-]
End of diagnosis flag for VIM adaptation failure					
LV_END_DIAG_VIM_MEC_LONG	V/O	0...1H	0...1	1	[-]
End of diagnosis flag for VIM mechanical failure long position (stuck check)					
LV_END_DIAG_VIM_MEC_SHO	V/O	0...1H	0...1	1	[-]
End of diagnosis flag for VIM mechanical failure short position (stuck check)					
LV_ERR_VIM_AD	V/O	0...1H	0...1	1	[-]
Flag to indicate VIM adaptation failure					
LV_ERR_VIM_MEC_LONG	V/O	0...1H	0...1	1	[-]
Flag to indicate VIM mechanical failure (long position not reached)					
LV_ERR_VIM_MEC_SHO	V/O	0...1H	0...1	1	[-]
Flag to indicate VIM mechanical failure (short position not reached)					
LV_ERR_VIM_PLAUS	V/O	0...1H	0...1	1	[-]
Flag to indicate VIM failure global error					
T_VIM_DIAG_MEC_LONG	V/O	0...FFFFH	0...1310.7	0.02	[s]
Timer for VIM has not reached its long position					

T_VIM_DIAG_MEC_SHO	V/O	0...FFFFH	0...1310.7	0.02	[s]
Timer for VIM has not reached its short position					

Input data:

LV_CDN_VB_OBD1	LV_IGK	LV_ES	CONF_VIM
LV_ST	V_VIM_AD_LONG	LV_ERR_ECU	STATE_VIM_AD
LV_ERR_VCC_SENS_SUB	LV_CDN_VB_OBD2	T_AST	LV_ERR_VIM_FB_EL
PQ	LV_VIM_SP	V_VIM	C_V_VIM_AD_SHO_HYS
C_V_VIM_AD_LONG_HYS	V_VIM_AD_SHO		

A.43.1 VIM power stage diagnosis (ATIC 39)

General information:

The VIM is driven by the ECU via ATIC 39. The failure detection is done by ECU Hardware. The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39” (static).

Error-symptoms: are defined to this diagnosis function as following

```
┌─┬─┬─┬─┐
│ │ │ └─> Short circuit to Vbatt        (= SYM_0)
│ │ └───> Short circuit to GND          (= SYM_1)
│ └─────> Open load                     (= SYM_2)
└───────> -                             (= SYM_3)
```

Application conditions:

Initialisation: at LV_IGK 0 → 1 **or** reset **or** clear failure memory

LV_INH_DIAG_VIM = 0

CDN_DIAG_VIM = 0

Recurrence: 100ms

Activation: CONF_VIM > 0

Formula section:

if	LV_IGK = 1	**and**
	LV_ES = 0	**and**
	LV_ST = 0	**and**
	LV_CDN_VB_OBD1 = 1	**and**
	LV_ERR_ECU = 0	
then	LV_INH_DIAG_VIM = 0	
else	LV_INH_DIAG_VIM = 1	
endif		

ABC-Type: use multi -condition debounce algorithm (MPL_STD_INI).

A.43.2 Diagnosis of VIM adaptation

Description:

For diagnosis the adaptation, the switching time and the electrical value of the feed back potentiometer can be used.

The diagnosis of adaptation (see chapter “adaptation of lower and higher of VIM ”).
Failure symptom:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> -                        (= SYM_0)
 │ │ └───> -                        (= SYM_1)
 │ └─────> -                        (= SYM_2)
 └───────> adaptation failure       (= SYM_3)
```

Application conditions:

Initialisation:	with LV_IGK = 0 --> 1 **or** reset **or** clear failure memory
	LV_CDN_DIAG_VIM_AD = 0
	LV_END_DIAG_VIM_AD = 0
	LV_ERR_VIM_AD = 0
	ERR_SYM_VIM_AD = NO_SYM
Recurrence:	20 ms
Activation:	CONF_VIM = 2 **and** LC_VIM_AD_DIAG_ENA = 1

Formula section:

```
If      STATE_VIM_AD > 0h (VIM_AD_PUC_TIMER)     and
        LV_END_DIAG_VIM_AD = 0
then    LV_CDN_DIAG_VIM_AD    = 1
else    LV_CDN_DIAG_VIM_AD    = 0
endif
 If     LV_CDN_DIAG_VIM_AD    = 1     and
        STATE_VIM_AD = VIM_AD_ERR
then    LV_ERR_VIM_AD = 1        (failure entry without debouncing)
        ERR_SYM_VIM_AD = SYM_3
endif
If      (STATE_VIM_AD = VIM_AD_ERR  or  STATE_VIM_AD = VIM_AD_OK)
then    LV_END_DIAG_VIM_AD = 1
endif
```

ABC Type: No

A.43.3 VIM stuck check

Description:

Usually the VIM switches from open to closed and reverse in a very short time. If the pneumatic system is not o.k. or the VIM has too much friction or even stucks, the switching time becomes too long or the VIM does not switch at all. This leads to a failure entry with symptom “mechanical failure”, “upper stop not reached” or “lower stop not reached”.

Failure symptom up:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> mechanical failure (upper stop not reached)     (= SYM_0)
 │ │ └───>                                                 (= SYM_1)
 │ └─────>                                                 (= SYM_2)
 └───────>                                                 (= SYM_3)
```

Failure symptom down:

```
[ | | | ]
 |  |  |  └─>                                                     (= SYM_0)
 |  |  └───> mechanical failure (lower stop not reached)          (= SYM_1)
 |  └──────>                                                      (= SYM_2)
 └─────────>                                                      (= SYM_3)
```

Application conditions:

Initialisation: with LV_IGK = 0 --> 1 or reset clear of failure memory
T_VIM_DIAG_MEC_LONG = C_T_VIM_DIAG_MEC_LONG
T_VIM_DIAG_MEC_SHO = C_T_VIM_DIAG_MEC_SHO
Following variables are initialized according filter type:
LV_CDN_DIAG_VIM_MEC_LONG
LV_CDN_DIAG_VIM_MEC_SHO
LV_ERR_VIM_MEC_LONG
LV_ERR_VIM_MEC_SHO
LV_END_DIAG_VIM_MEC_LONG
LV_END_DIAG_VIM_MEC_SHO
ERR_SYM_VIM_MEC_LONG
ERR_SYM_VIM_MEC_SHO

Recurrence: 20 ms

Activation: LV_ES ≠ 1 **and** LV_ST≠ 1 **and** CONF_VIM = 2

Formula section:

If LV_CDN_VB_OBD2 = 1
and LV_ERR_VCC_SENS_SUB = 0
and LV_ERR_VIM = 0
and T_AST > C_T_AST_MIN_VIM_DIAG
and LV_ERR_VIM_FB_EL = 0
and C_PQ_MIN_VIM_DIAG < PQ < C_PQ_MAX_VIM_DIAG
and STATE_VIM_AD = VIM_AD_OK
then LV_CDN_DIAG_VIM_MEC_LONG = 1
LV_CDN_DIAG_VIM_MEC_SHO = 1
endif

If LV_CDN_DIAG_VIM_MEC_SHO = 1
and LV_VIM_SP = 0

```
then    if      T_VIM_DIAG_MEC_LONG  ≠  0
        then    T_VIM_DIAG_MEC_LONG = C_T_VIM_DIAG_MEC_LONG
        endif
        if      V_VIM > V_VIM_AD_SHO + C_V_VIM_AD_SHO_HYS (lower stop not reached)
        then    if      T_VIM_DIAG_MEC_SHO  ≠  0
                then    T_VIM_DIAG_MEC_SHO starts counting downwards
                else    LV_ERR_VIM_MEC_ SHO           = 1
                        LV_END_DIAG_VIM_MEC_ SHO      = 1
                        ERR_SYM = SYM_1               (failure entry without debouncing)
                endif
        else    LV_END_DIAG_VIM_MEC_SHO               = 1
                LV_ERR_VIM_MEC_SHO                    = 0
                ERR_SYM = NO_SYM
                T_VIM_DIAG_MEC_SHO = C_T_VIM_DIAG_MEC_SHO
                                          (timer stopped and initialized)
        endif
endif
If      LV_CDN_DIAG_VIM_MEC_LONG  = 1
then    and     LV_VIM_SP = 1
        if      T_VIM_DIAG_MEC_SHO  ≠  0
        then    T_VIM_DIAG_MEC_SHO = C_T_VIM_DIAG_MEC_SHO
        endif
        if      V_VIM<V_VIM_AD_LONG-C_V_VIM_AD_LONG_HYS (upper stop not reached)
                if      T_VIM_DIAG_MEC_LONG  ≠  0
                then    T_VIM_DIAG_MEC_LONG starts counting downwards
                else    LV_ERR_VIM_MEC_ LONG          = 1
                        LV_END_DIAG_VIM_MEC_ LONG     = 1
                        ERR_SYM = SYM_0               (failure entry without debouncing)
                endif
        else    LV_END_DIAG_VIM_MEC_LONG              = 1
                LV_ERR_VIM_MEC_LONG                   = 0
                ERR_SYM = NO_SYM
                T_VIM_DIAG_MEC_LONG = C_T_VIM_DIAG_MEC_LONG
                                          (timer stopped and initialized)
        endif
endif
```

ABC Type: No

A.43.4 Global failure for VIM flap diagnosis

Description:

For interface to other functions a global failure of VIM flap diagnosis is defined.

Formula section:

```
If      LV_ERR_VIM_AD        = 1
   or   LV_ERR_VIM_MEC_LONG  = 1
   or   LV_ERR_VIM_MEC_SHO   = 1
   or   LV_ERR_VIM_FB_EL     = 1
   or   LV_ERR_VCC_SENS_SUB  = 1
then    LV_ERR_VIM_PLAUS     = 1
else    LV_ERR_VIM_PLAUS     = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_VIM_AD_DIAG_ENA	1	0...1H	0...1	1	[-]
Logical variable for enable of LV_ERR_VIM_AD					
C_T_AST_MIN_VIM_DIAG	1	0...FFFFH	0...6553.5	0.1	[s]
minimum time after start to activate VIM mechanical diagnosis					
C_T_VIM_DIAG_MEC_LONG	1	0...FFFFH	0...1310.7	0.02	[s]
timer for mechanical VIM diagnosis long position					
C_T_VIM_DIAG_MEC_SHO	1	0...FFFFH	0...1310.7	0.02	[s]
Timer for mechanical VIM diagnosis short position					
C_PQ_MAX_VIM_DIAG	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
maximum PQ threshold for stuck diagnosis					
C_PQ_MIN_VIM_DIAG	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
minimum PQ threshold for stuck diagnosis					
C_ABC_INC_VIM	1	0...FFH	0...255	1	-
Debounce counter increment - VIM diagnosis					
C_ABC_MAX_VIM	1	01...FFH	1...255	1	-
Debounce counter maximum value - VIM diagnosis					

A.43.5 Application Incidences for Variable Intake Manifold Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_VIM_MEC_SHO	V/O	0...7H	0...7	1	[-]
Interface of monitor VIM_MEC_SHO (VIM feedback diagnosis) for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2= 2)					
LV_INH_DIAG_RBM_VIM_MEC	V/O	0...1H	0...1	1	[-]
VIM_MEC plausibility diagnosis inhibition condition for RBM interface definition					
STATE_RBM_VIM_MEC_LONG	V/O	0...7H	0...7	1	[-]
Interface of monitor VIM_MEC_LONG (VIM feedback diagnosis) for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2= 2)					
T_VIM_SWI_MEC_SHO_ MAX_TOT_DC	V/O/S	0...FFFFH	0...1310.7	0.02	[s]
Former / current driving cycle maximum of time for VIM needed to reach its short position					
T_VIM_SWI_MEC_LONG_ MAX_TOT_DC	V/O/S	0...FFFFH	0...1310.7	0.02	[s]
Former / current driving cycle maximum of time for VIM needed to reach its long position					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_ERR_VCC_SENS_SUB	LV_ERR_VIM
LV_ERR_VIM_FB_EL	LV_END_DIAG_VIM_MEC_SHO	LV_END_DIAG_VIM_MEC_LONG	CONF_VIM
T_VIM_DIAG_MEC_LONG	T_VIM_DIAG_MEC_SHO	LV_CDN_DIAG_VIM_MEC_LONG	C_T_VIM_DIAG_MEC_SHO
C_T_VIM_DIAG_MEC_LONG			

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <VIM_MEC_SHO>, OUT <PendingStatus>)
ACTION_ERRM_CheckPendingStatus (IN <VIM_MEC_LONG>, OUT <PendingStatus>)

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Variable Intake Manifold Diagnosis	Short Circuit to Battery	SYM_0	*MPL_STD_INI*
	Short Circuit to Ground	SYM_1	
	Open Circuit	SYM_2	
VIM	-	SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Diagnosis of VIM adaptation	-	SYM_0	*no*
	-	SYM_1	
	-	SYM_2	
VIM_AD	Adaptation failure	SYM_3	
VIM stuck check lower stop	-	SYM_0	*no*
	Mechanical failure (lower stop not reached)	SYM_1	
	-	SYM_2	

VIM_MEC_SHO	-	SYM_3	
VIM stuck check upper stop	Mechanical failure (upper stop not reached)	SYM_0	*no*
	-	SYM_1	
	-	SYM_2	
VIM_MEC_LONG	-	SYM_3	

A.43.6 Variables for fleet monitoring

A.43.7 - Monitoring of T_VIM_DIAG_MEC_LONG(_SHO):

Initialisation: - at first ECU power up and non-volantile memory lost:
T_VIM_SWI_MEC_LONG(_SHO)_MAX_TOT_DC = 0H
- otherwise: restored from non-volantile memory

Recurrence: *20 ms*

Activation: *LV_CDN_DIAG_VIM_MEC_LONG = 1*

Deactivation: -

Formula section:

If C_T_VIM_DIAG_MEC_LONG(_SHO) - T_VIM_DIAG_MEC_LONG(_SHO) >
T_VIM_SWI_MEC_LONG(_SHO)_MAX_TOT_DC

Then T_VIM_SWI_MEC_LONG(_SHO)_MAX_TOT_DC =
C_T_VIM_DIAG_MEC_LONG(_SHO) - T_VIM_DIAG_MEC_LONG(_SHO)

Endif

A.43.8 RBM Interface for VIM mechanical failure short position

FUNCTION DESCRIPTION:

General information:

With this module the interface between the monitor VIM_MEC_SHO and the Rate-Based Monitoring statistics is defined with STATE_RBM_VIM_MEC_SHO data.

Within STATE_RBM_VIM_MEC_SHO three different information are defined:

Information about bit 0 of STATE_RBM_VIM_MEC_SHO:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_VIM_MEC_SHO:

Monitor disabled because of system malfunction (depending on failure status: pending)

Information about bit 2 of STATE_RBM_VIM_MEC_SHO:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation:

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_VIM_MEC_SHO = 0
LV_INH_DIAG_RBM_VIM_MEC = 0

at LV_DC 0 -> 1 transition :
bit 0, bit 1 of STATE_RBM_VIM_MEC_SHO = 0
bit 2 of STATE_RBM_VIM_MEC_SHO = 1

On failure memory reset :
bit 1 of STATE_RBM_VIM_MEC_SHO = 0

Recurrence: 1s

Activation: CONF_VIM = 2 **and** LV_DC 0 -> 1 transition **and** LV_DC = 1

Formula section:

{ definition of LV_INH_DIAG_RBM_VIM_MEC }

If LV_ERR_VCC_SENS_SUB = 1 **or**
LV_ERR_VIM = 1 **or**
LV_ERR_VIM_FB_EL = 1 **or**
then LV_INH_DIAG_RBM_VIM_MEC = 1

endif

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once:

LV_ERR_VCC_SENS_SUB

LV_ERR_VIM

LV_ERR_VIM_FB_EL

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_VIM_MEC_SHO = 0 **do**

with each VIM failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_VIM_MEC_SHO = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_VIM_MEC_SHO = 0

Then

If LV_END_DIAG_VIM_MEC_SHO = 1

Then bit 0 of STATE_RBM_VIM_MEC_SHO = 1

Endif

Endif

If bit 1 of STATE_RBM_VIM_MEC_SHO = 0

Then

If LV_INH_DIAG_RBM_VIM_MEC = 1

Then bit 1 of STATE_RBM_VIM_MEC_SHO = 1

Endif

Endif

A.43.9 RBM Interface for VIM mechanical failure long position

FUNCTION DESCRIPTION:

General information:

With this module the interface between the monitor VIM_MEC_LONG and the Rate-Based Monitoring statistics is defined with STATE_RBM_VIM_MEC_LONG data.

Within STATE_RBM_VIM_MEC_LONG three different information are defined:

Information about bit 0 of STATE_RBM_VIM_MEC_LONG:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_VIM_MEC_LONG:

Monitor disabled because of system malfunction (depending on failure status: pending)

Information about bit 2 of STATE_RBM_VIM_MEC_LONG:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation:

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_VIM_MEC_LONG = 0

at LV_DC 0 -> 1 transition :
bit 0, bit 1 of STATE_RBM_VIM_MEC_LONG = 0
bit 2 of STATE_RBM_VIM_MEC_LONG = 1

On failure memory reset :
bit 1 of STATE_RBM_VIM_MEC_LONG = 0

Recurrence: 1s

Activation: CONF_VIM = 2 **and** LV_DC 0 -> 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once:

LV_ERR_VCC_SENS_SUB

LV_ERR_VIM

LV_ERR_VIM_FB_EL

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_VIM_MEC_LONG = 0 **do**

with each VIM failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_VIM_MEC_LONG = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_VIM_MEC_LONG = 0

Then

If LV_END_DIAG_VIM_MEC_LONG = 1

Then bit 0 of STATE_RBM_VIM_MEC_LONG = 1

Endif

Endif

If bit 1 of STATE_RBM_VIM_MEC_LONG = 0

Then

If LV_INH_DIAG_RBM_VIM_MEC = 1

Then bit 1 of STATE_RBM_VIM_MEC_LONG = 1

Endif

Endif

A.44 Electrical diagnosis of VIM feed-back

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_VIM_FB_EL	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error Symptom for VIM potentiometer failure					
LV_CDN_DIAG_VIM_FB_EL	V/O	0...1H	0...1	1	[-]
Diagnosis active flag for VIM potentiometer failure					
LV_END_DIAG_VIM_FB_EL	V/O	0...1H	0...1	1	[-]
End of diagnosis flag for VIM potentiometer failure					
LV_ERR_VIM_FB_EL	V/O	0...1H	0...1	1	[-]
Flag to indicate VIM feedback potientiometer failure (electrical failure); for VIM poti					

Input data:

V_VIM	LV_CDN_VB_OBD1	LV_IGK	CONF_VIM
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

A.44.1 Electrical diagnosis of VIM feed-back

Description:

The feed-back value of the potentiometer can be obeserved if it is within sensefull electrical range.

Failure symptom:

```
□□□□
│ │ │ └─> short cut to battery / open loop      (= SYM_0)
│ │ └───> short cut to ground                   (= SYM_1)
│ └─────> -                                     (= SYM_2)
└───────> -                                     (= SYM_3)
```

Application conditions:

Initialisation: with LV_IGK = 0 --> 1 or reset clear of failure memory

Recurrence: 20 ms

Activation: LV_ST = 0 and CONF_VIM = 2

Formula section:

Short cut to battery/ Short cut to ground or open loop

```
If      LV_IGK              = 1   And
        LV_CDN_VB_OBD1      = 1   And
        LV_ERR_VCC_SENS_SUB = 0
Then    LV_CDN_DIAG_VIM_FB_EL = 1
        If    V_VIM > C_V_VIM_MAX_DIAG
        Then  ERR_SYM_VIM_FB_EL = SYM_0            ; Short cut to battery
        Else  If    V_VIM < C_V_VIM_MIN_DIAG
              Then  ERR_SYM_VIM_FB_EL = SYM_1      ;Short cut to ground or open loop
              Else  ERR_SYM_VIM_FB_EL = NO_SYM
              Endif
        Endif
Else    LV_CDN_DIAG_VIM_FB_EL = 0
Endif
```

The LV_END_DIAG-Bit and LV_ERR-Bit is managed by the error memory.

ABC Type: STD_INI

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_VIM_MAX_DIAG	1	0...3FFH	0...4.99511	4.8828e-3	[V]
maximum VIM poti value for electrical diagnosis					
C_V_VIM_MIN_DIAG	1	0...3FFH	0...4.99511	4.8828e-3	[V]
minimum VIM poti value for electrical diagnosis					
C_ABC_INC_VIM_FB_EL	1	0...FFH	0...255	1	[-]
Debounce counter for VIM poti electrical diagnosis					
C_ABC_MAX_VIM_FB_EL	1	1...FFH	1...255	1	[-]
Debounce maximum for VIM poti electrical diagnosis					

A.44.2 VIM feedback sensor electrical diag (Appl. Inc.)

Diagnostic	Symptom Description	Symptom	Filter type
Electrical diagnosis of Variable Intake Manifold feedback sensor	Short Circuit to Battery / Open Circuit	SYM_0	*STD_INI*
	Short Circuit to Ground	SYM_1	
	-	SYM_2	
VIM_FB_EL	-	SYM_3	

List of Environmental Data to store in the Failure Memory:

TIA

V_VIM_ENVD

LV_VIM_SP

VCC_SENS_SUB_DIAG_ENVD

A.45 Diagnosis of Pedal Value Sensor (PVS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PVS	O/V	0...1H	0...1	1	-
Any PVS-error debounced					
LV_ACT_PVS_LIH	O/V	0...1H	0...1	1	-
Flag for activation of additional limitations					
LV_ERR_PVS_PRED	O/V	0...1H	0...1	1	-
PVS-error predicted, but not debounced yet					
LV_ERR_PVS_RATIO_PRED	O/V	0...1H	0...1	1	-
PVS-ratio-error predicted, but not debounced yet					
LV_ERR_PVS_1	O/V	0...1H	0...1	1	-
PVS_1 error debounced					
LV_ERR_PVS_H_1	O/V	0...1H	0...1	1	-
PVS_1 out of range high error debounced					
LV_CDN_DIAG_PVS_H_1	O/V	0...1H	0...1	1	-
Diagnosis condition out of range check high PVS_1					
ERR_SYM_PVS_H_1	O/V	0H 1H	NO_SYM SYM_0	1	-
Symptom channel 1 is out of range high					
LV_END_DIAG_PVS_H_1	O/V	0...1H	0...1	1	-
Result of ORNG_H_1 diagnosis					
LV_ERR_PVS_L_1	O/V	0...1H	0...1	1	-
Out of range low error debounced PVS_1					
LV_CDN_DIAG_PVS_L_1	O/V	0...1H	0...1	1	-
Diagnosis condition out of range check low PVS_1					
ERR_SYM_PVS_L_1	O/V	0H 2H	NO_SYM SYM_1	1	-
Symptom channel 1 is out of range low					
LV_END_DIAG_PVS_L_1	O/V	0...1H	0...1	1	-
Result of ORNG_L_1 diagnosis					
LV_ERR_PVS_2	O/V	0...1H	0...1	1	-
PVS_2 error debounced					
LV_ERR_PVS_H_2	O/V	0...1H	0...1	1	-
PVS_2 out of range high error debounced					
LV_CDN_DIAG_PVS_H_2	O/V	0...1H	0...1	1	-
Diagnosis condition out of range check high PVS_2					
ERR_SYM_PVS_H_2	O/V	0H 1H	NO_SYM SYM_0	1	-
Symptom channel 2 is out of range high					
LV_END_DIAG_PVS_H_2	O/V	0...1H	0...1	1	-
Result of ORNG_H_2 diagnosis					
LV_ERR_PVS_L_2	O/V	0...1H	0...1	1	-
Out of range low error debounced PVS_2					
LV_CDN_DIAG_PVS_L_2	O/V	0...1H	0...1	1	-
Diagnosis condition out of range check low PVS_2					
ERR_SYM_PVS_L_2	O/V	0H 2H	NO_SYM SYM_1	1	-
Symptom channel 2 is out of range low					
LV_END_DIAG_PVS_L_2	O/V	0...1H	0...1	1	-
Result of ORNG_L_2 diagnosis					
LV_ERR_PVS_RATIO	O/V	0...1H	0...1	1	-
PVS-Ratio error debounced					
V_PVS_MAX	V	0...FFFFH	0...319.995117	0.004883	V
maximum voltage for ratio check					
V_PVS_RATIO_HYS	O/V	0...3FFH	0...4.995117	0.004883	V
Max actual PVS Ratio Hysteresis					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_PVS_RATIO	O/V	0...1H	0...1	1	-
Diagnosis condition Ratio Check					
ERR_SYM_PVS_RATIO	O/V	0H 8H	NO_SYM SYM_3	1	-
Symptom ratio deviation of PVS1 and PVS2					
LV_END_DIAG_PVS_RATIO	O/V	0...1H	0...1	1	-
Result of Ratio Check					
LV_PVS_H_R	O/V	0...1H	0...1	1	-
High resistance recognized on both channel					
LV_PVS_H_R_1	O/V	0...1H	0...1	1	-
PVS_H_R_1 error recognized					
LV_PVS_H_R_2	O/V	0...1H	0...1	1	-
PVS_H_R_2 error recognized					
LV_PVS_H_R_1_RLS	O/V	0...1H	0...1	1	-
Flag for memory of PVS-1 high resistance recognition in next DC					
LV_PVS_H_R_2_RLS	O/V	0...1H	0...1	1	-
Flag for memory of PVS-2 high resistance recognition in next DC					
LV_ERR_PVS_MOVE	O/V	0...1H	0...1	1	-
Move Error debounced					
LV_LIH_PVS_MOVE	O/V	0...1H	0...1	1	-
PVS-move-error recognized					
CTR_MOVE	V	0...FFH	0...255	1	-
value of CTR_MOVE					
LV_CDN_DIAG_PVS_MOVE	O/V	0...1H	0...1	1	-
Diagnosis condition Move Check					
ERR_SYM_PVS_MOVE	O/V	0H 4H	NO_SYM SYM_2	1	-
Symptom move error detected					
LV_END_DIAG_PVS_MOVE	O/V	0...1H	0...1	1	-
Result of Move Check					
LV_ERR_PVS_BLS_BTS	O/V	0...1H	0...1	1	-
Plausibility error PVS / Brake debounced					
LV_PVS_BLS_BTS_NOT_PLAUS	O/V	0...1H	0...1	1	-
Plausibility error PVS / Brake recognized					
LV_CDN_DIAG_PVS_BLS_BTS	O/V	0...1H	0...1	1	-
Diagnosis condition Plausibility Check PVS / Brake					
ERR_SYM_PVS_BLS_BTS	O/V	0H 8H	NO_SYM SYM_3	1	-
Symptom plausibility PVS / BLS BTS error detected					
LV_END_DIAG_PVS_BLS_BTS	O/V	0...1H	0...1	1	-
Result of Plausibility Check PVS / Brake					
LV_PVS_BLS_BTS_RES	V	0...1H	0...1	1	-
Reset-Bit PVS / brake not plausible					
V_PVS_MMV_1	V	0...3FFH	0...4.995117	0.004883	V
Filtered signal voltage channel 1 for move check					
V_PVS_MMV_2	V	0...3FFH	0...4.995117	0.004883	V
Filtered signal voltage channel 2 for move check					
V_PVS_MMV_SAVE_2	V	0...3FFH	0...4.995117	0.004883	V
Saved filtered signal voltage channel 2 for move check					
LV_ERR_PVS_ORNG_PRED	O/V	0...1H	0...1	1	-
Predicted error : Pedal value sensor out of range					

Input data:

LV_IGK	V_PVS_1	V_PVS_2	PV_AV_1_H
PV_AV_2_H	N_32	VS	PV_AV_SEL
LV_BLS	LV_BTS	LV_ERR_VCC_PVS_1	LV_ERR_VCC_PVS_2
LV_PVS_H_R_1_DET	LV_PVS_H_R_2_DET	LV_PVS_BLS_BTS_PLAUS_ENA	LV_ERR_VS
C_V_PVS_THD_IS_L_1			

FUNCTION DESCRIPTION:

General information:

The purpose of the diagnosis shall be the detection of electrical faults and not plausible conditions. Therefore the following error-checks are performed:

- Signal range check high
- Signal range check low
- Ratio check
- High resistance check (only for potentiometer systems)
- Move check
- Plausibility brake / PVS

For freeze frame, DTC, error code number, symptom number, emission error type, successive error management, data for MIL-management and anti-bounce configuration see chapters for general diagnosis information and application incidences.

PVS1 out of range check high error symptoms :

```
[ | | | ]
 | | | └─> Short cicuit to Vbatt "SCP"                    (= SYM_0)
 | | └───> (not used)                                     (= SYM_1)
 | └─────> (not used)                                     (= SYM_2)
 └───────> (not used)                                     (= SYM_3)
```

PVS1 out of range check low error symptoms :

```
[ | | | ]
 | | | └─> (not used)                                     (= SYM_0)
 | | └───> Short cicuit to Ground or Open Load "SCG_OL"   (= SYM_1)
 | └─────> (not used)                                     (= SYM_2)
 └───────> (not used)                                     (= SYM_3)
```

PVS2 out of range check high error symptoms :

```
[ | | | ]
 | | | └─> Short cicuit to Vbatt "SCP"                    (= SYM_0)
 | | └───> (not used)                                     (= SYM_1)
 | └─────> (not used)                                     (= SYM_2)
 └───────> (not used)                                     (= SYM_3)
```

PVS2 out of range check low error symptoms :

```
□□□□
| | | └─> (not used)                                    (= SYM_0)
| | └───> Short cicuit to Ground or Open Load "SCG_OL"    (= SYM_1)
| └─────> (not used)                                    (= SYM_2)
└───────> (not used)                                    (= SYM_3)
```

Ratio check error symptoms :

```
□□□□
| | | └─> (not used)                                    (= SYM_0)
| | └───> (not used)                                    (= SYM_1)
| └─────> (not used)                                    (= SYM_2)
└───────> Ratio deviation                               (= SYM_3)
```

Move check error symptoms :

```
□□□□
| | | └─> (not used)                                    (= SYM_0)
| | └───> (not used)                                    (= SYM_1)
| └─────> MOVE_MAX                                      (= SYM_2)
└───────> (not used) MOVE_MAX                                   (= SYM_3)
```

Plausibility check PVS / BLS BTS error symptoms :

```
□□□□
| | | └─> (not used)                                    (= SYM_0)
| | └───> (not used)                                    (= SYM_1)
| └─────> (not used)                                    (= SYM_2)
└───────> PLAUS_MAX                                     (= SYM_3)
```

General Application conditions:

Activation: LV_IGK = 1

Deactivation: otherwise

Initialization: for additional conditions see ‘Application Incidences of PVS

CTR_ADD_ACT_ENA = C_CTR_MAX_PVS_IS_RNG
CTR_H_R_1 = C_CTR_MAX_PVS_H_R_1
CTR_H_R_2 = C_CTR_MAX_PVS_H_R_2
CTR_MOVE = 0
CTR_MOVE_ACT = C_CTR_MAX_PVS_MOVE
CTR_NOT_PLAUS = C_CTR_MAX_PVS_BLS_BTS_PLAUS
CTR_PLAUS = C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR
CTR_PVS_STAT = C_CTR_MAX_PV_AV_CON
ERR_SYM_PVS_BLS_BTS = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_MOVE= *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_H_1 = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_H_2 = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_L_1 = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_L_2 = *Refer to filtering configuration for the initialisation value*
ERR_SYM_PVS_RATIO= *Refer to filtering configuration for the initialisation value*
LV_ACT_PVS_LIH = 0

LV_ADD_ACT_ENA = 0
LV_CDN_DIAG_PVS_BLS_BTS = *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_MOVE= *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_H_1 = *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_H_2 = *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_L_1 = *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_L_2 = *Refer to filtering configuration for the initialisation value*
LV_CDN_DIAG_PVS_RATIO = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_PVS_BLS_BTS = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_PVS_MOVE = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS = 0
LV_ERR_PVS_1 = 0
LV_ERR_PVS_2 = 0
LV_ERR_PVS_H_1 = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_H_2 = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_L_1 = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_L_2 = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_BLS_BTS = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_MOVE= *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_PRED = 0
LV_ERR_PVS_RATIO = *Refer to filtering configuration for the initialisation value*
LV_ERR_PVS_RATIO_PRED = 0
LV_LIH_PVS_MOVE = 0
LV_PVS_1_MOVE_DET(K) = 1
LV_PVS_BLS_BTS_NOT_PLAUS= 0
LV_PVS_H_R = 0
LV_PVS_H_R_1 = 0
LV_PVS_H_R_2 = 0
LV_PVS_MOVE_ACT_SUB = 0
LV_PVS_MOVE_ACT_SUB_DOWN = 0
V_PVS_MMV_1(K)=0
V_PVS_MMV_2(K)=0
V_PVS_MMV_SAVE_2(K) = 0

Recurrence: every 10 ms

Signal flow diagram:

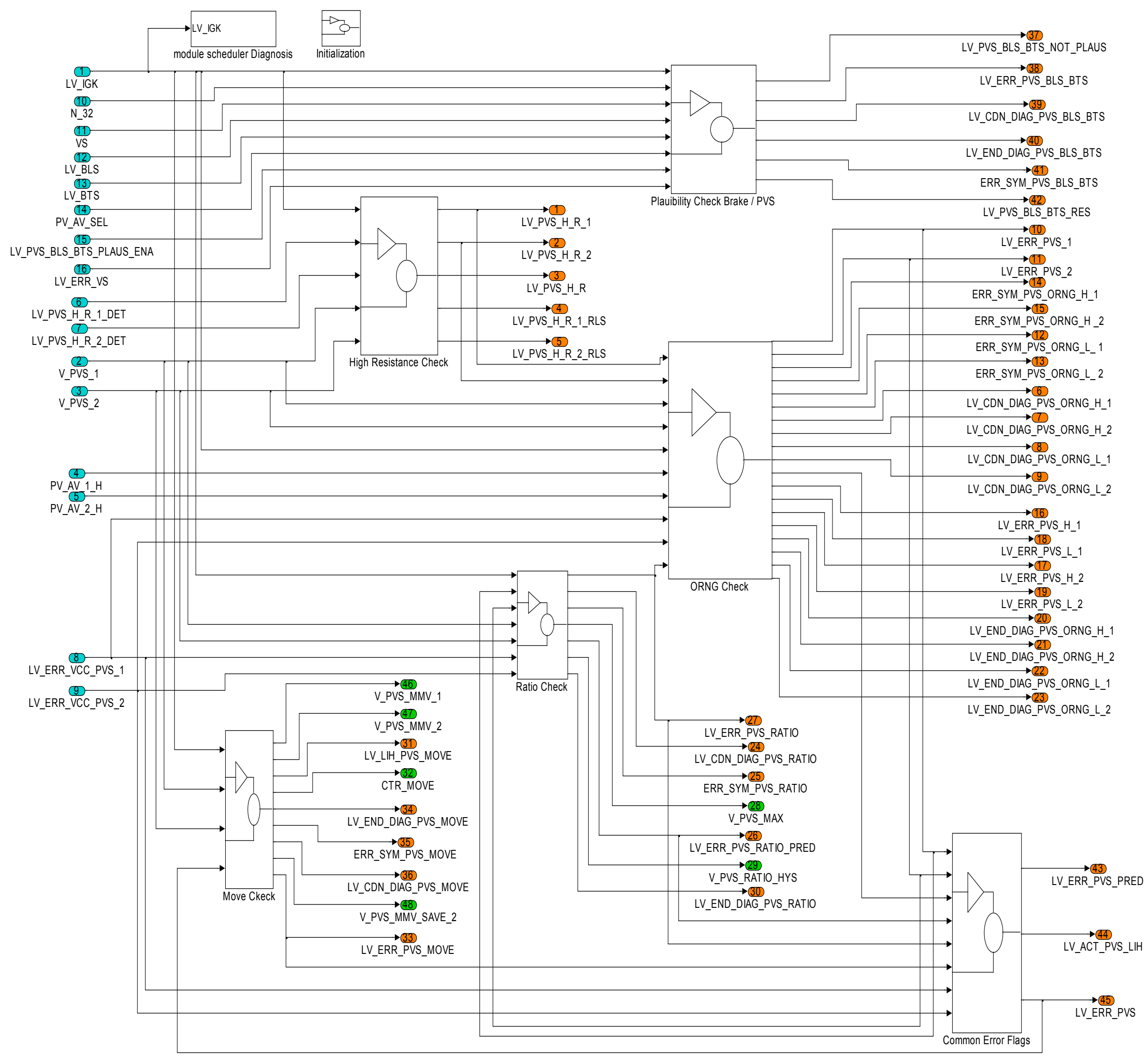


Figure 1: Diagnosis of Pedal Value Sensor (PVS)

Description:

Module scheduler:

The module „Diagnosis of Pedal Value Sensor (PVS)“ is executed at function call CLC_PVS_DIAG__10MS and initialized at CLC_PVS_DIAG__RST.
Each function part is calculated according to the corresponding function call which order is organized in the module scheduler shown below.

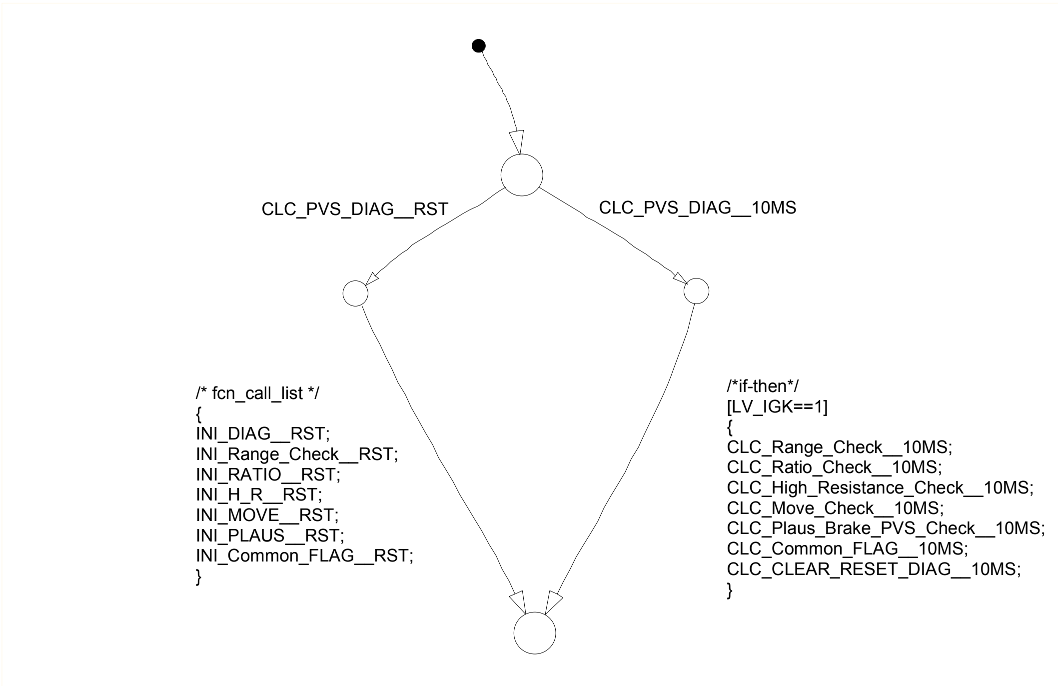


Figure 2: module scheduler Diagnosis

Module calculation order **at** Initialization

1. INITIALIZATION	on fcn-call INI_DIAG__RST
2. ORNG Check	on fcn-call INI_Range_Check_RST
3. Ratio Check	on fcn-call INI_RATIO__RST
4. High Resistance Check	on fcn-call INI_H_R__RST
5. Move Check	on fcn-call INI_MOVE_RST
6. Plausibility Check Brake / PVS	on fcn-call INI_PLAUS__RS
7. Common Error Flags	on fcn-call INI_Common_FLAG__RST

Module calculation order **after** Initialization

1. ORNG check	on fcn-call CLC_Range_Check__10MS
2. Ratio Check	on fcn-call CLC_Ratio_Check__10MS
3. High Resistance Check	on fcn-call CLC_High_Resistance_Check__10MS
4. Move Check	on fcn-call CLC_Move_Check__10MS
5. Plausibility Check Brake / PVS	on fcn-call CLC_Plaus_Brake_PVS_Check__10MS
6. Common Error Flags	on fcn-call CLC_Common_FLAG__10MS
7. INITIALIZATION	on fcn-call CLC_CLEAR_RESET_DIAG__10MS

Initialization:

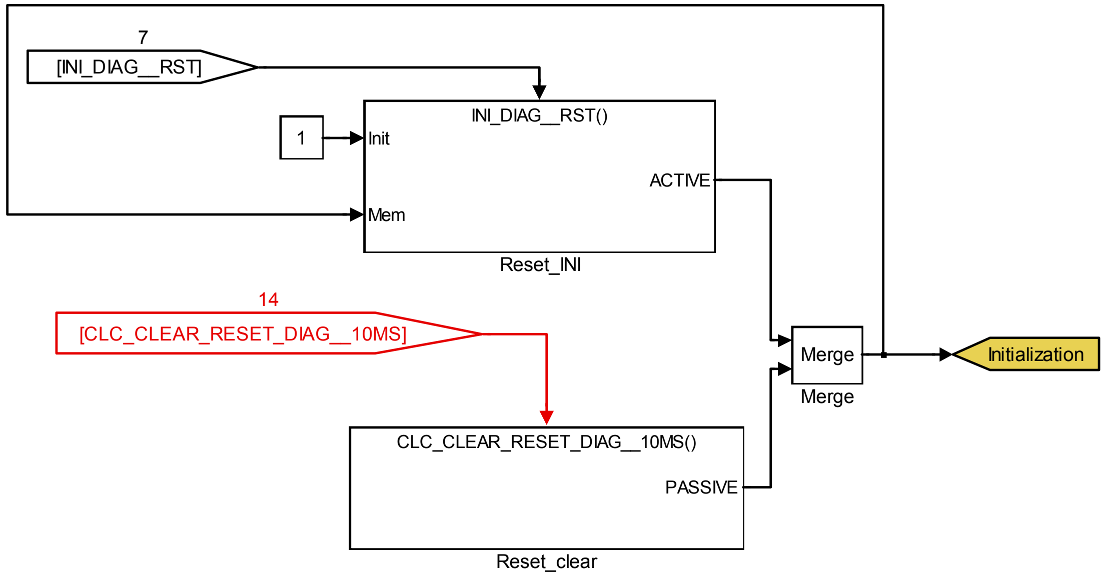


Figure 3: Initialization

At function call **INI_DIAG__RST** INITIALIZATION is set to **active**.
It is set to **passive** again at the end of the next regular calculation cycle at function call **CLC_CLEAR_RESET_DIAG__10MS.**

A.45.1 Out of range check (ORNG)

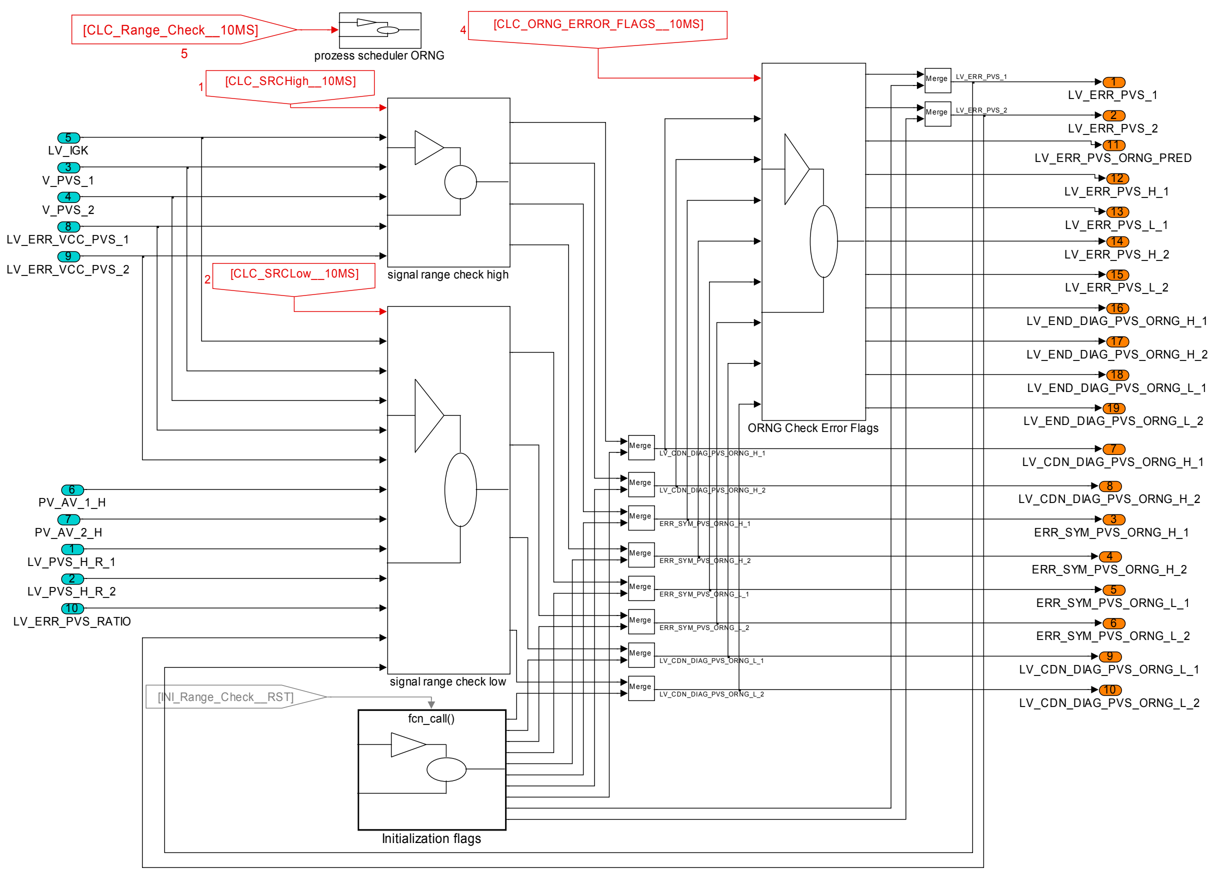


Figure 4: ORNG check

Description:

During normal operation, the output voltages of the PVS-sensors must lie within a permitted range.

If a signal voltage is detected as out of range (above or below), the bit LV_ERR_PVS_ORNG_PRED will be set during debouncing. The bit will be reset, if

– the setting conditions are not fulfilled anymore (diagnosis condition flag + symptom) or

– after the error is debounced, LV_ERR_PVS_x = 1

In order to re-examine, whether a range violation has caused a Ratio-error, the debouncing times of both signal range checks (high / low) have to be selected smaller than the debouncing time of the Ratio-check.

Calculation order:

1. Signal Range Check High — on fcn-call CLC_SRCHigh__10MS
2. Signal Range Check Low — on fcn-call CLC_SRCLow__10MS
3. ORNG Check Error Flags — on fcn-call CLC_ORNG_ERROR_FLAGS__10MS

A.45.1.1 Signal range check high

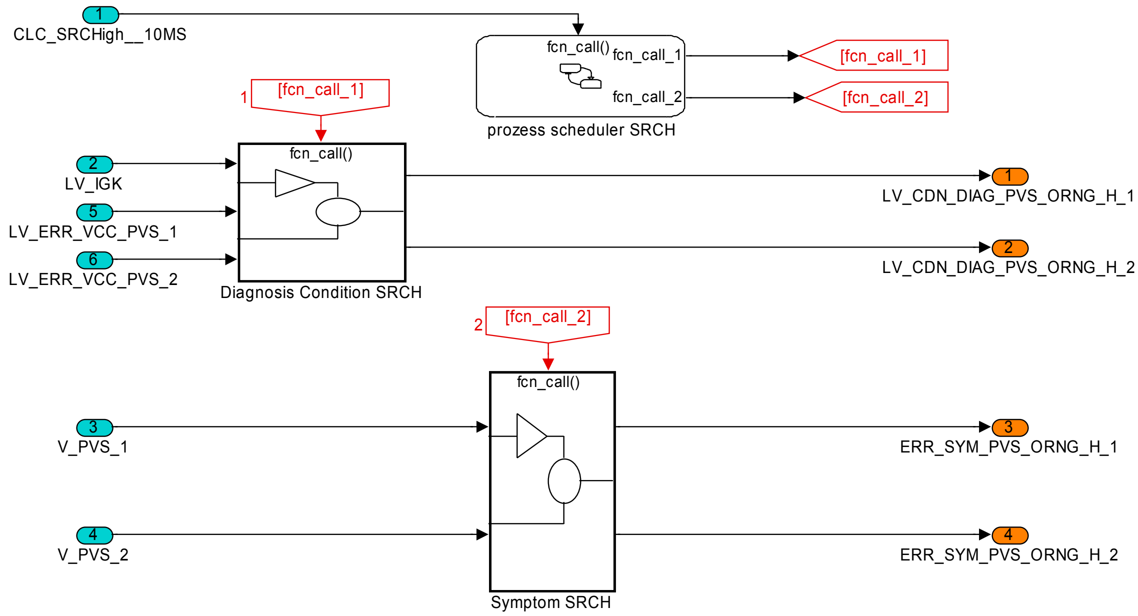


Figure 5: Signal range check high

Description:

In an error-free system, the sensor voltages must lie below the limits C_V_PVS_MAX_DIAG_x.

Calculation order:

1. Diagnosis Condition SRCH on fcn_call_1
2. Symptom SRCH on fcn_call_2

Formula section:

Diagnosis Condition SRCH:

IF LV_IGK == 1 **AND** LV_ERR_VCC_PVS_1 = 0
THEN LV_CDN_DIAG_PVS_H_1 = 1 (Diagnosis channel 1 active)
ELSE LV_CDN_DIAG_PVS_H_1 = 0 (Diagnosis channel 1 passive)
ENDIF

IF LV_IGK == 1 **AND** LV_ERR_VCC_PVS_2 = 0
THEN LV_CDN_DIAG_PVS_H_2 = 1 (Diagnosis channel 2 active)
ELSE LV_CDN_DIAG_PVS_H_2 = 0 (Diagnosis channel 2 passive)
ENDIF

Symptom SRCH_1:

IF V_PVS_1 > C_V_PVS_MAX_DIAG_1
THEN ERR_SYM_PVS_H_1= 1H (short to plus)
//anti-bounce counter (PVS_H_1) + C_ABC_INC_PVS_ORNG_1, only one increment per activated time condition, LV_ERR_PVS_H_1 = 1 after debounced
ELSE ERR_SYM_PVS_H_1= 0H (no symptom)
ENDIF

Filtering:

APPLY filter on current symptom

IF filtering result available (after debounce)

THEN LV_ERR_PVS_H_1 = filtering result

LV_END_DIAG_PVS_H_1 = 1

Deliver the result to Error Management

END

Symptom SRCH_2:

IF V_PVS_2 > C_V_PVS_MAX_DIAG_2
THEN ERR_SYM_PVS_H_2 = 1H (short to plus)
//anti-bounce counter (PVS_H_2) + C_ABC_INC_PVS_ORNG_2, only one increment per activated time condition, LV_ERR_PVS_H_2 = 1 after debounced
ELSE ERR_SYM_PVS_H_2 = 0H (no symptom)
ENDIF

Filtering:

APPLY filter on current symptom

IF filtering result available (after debounce)

THEN LV_ERR_PVS_H_2 = filtering result

LV_END_DIAG_PVS_H_2 = 1

Deliver the result to Error Management

END

A.45.1.2 Signal Range Check Low (SRCL)

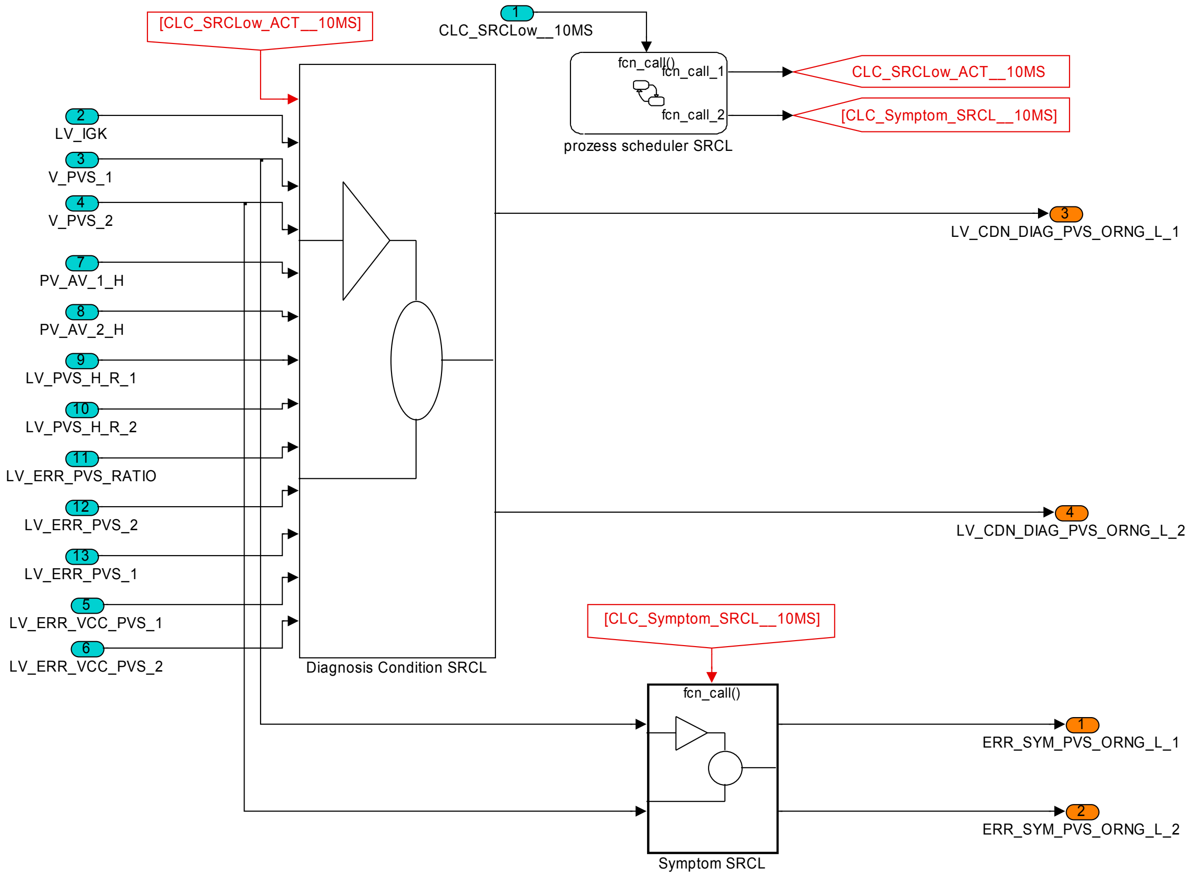


Figure 6: Signal Range Check Low

Description:

Calculation order:

1. Diagnosis Condition SRCL — on fcn-call CLC_SRCLow_ACT__10MS
2. Symptom SRCL — on fcn-call CLC_Symptom_SRCL__10MS

A.45.1.2.1 Diagnosis Condition SRCL:

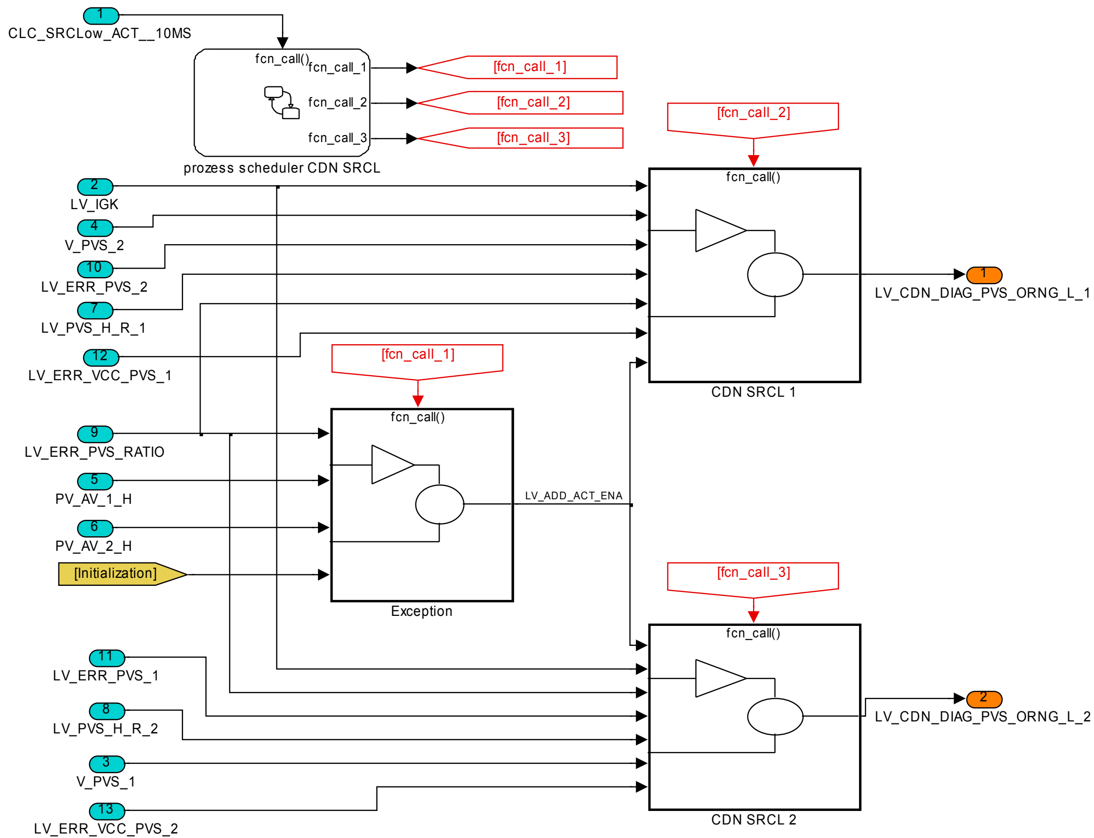


Figure 7: Diagnosis Condition Signal Range Check Low

Description:

The diagnosis conditions of the signal range check low on channel 1 and 2 depend on different conditions for each channel respectively and a common exception condition for both. If the diagnosis conditions are fulfilled, the flag LV_CDN_DIAG_PVS_ORNG_L_x =1.

Calculation order:

1. Exception on fcn_call_1
2. CDN SRCL 1 on fcn_call_2
3. CDN SRCL 2 on fcn_call_3

Exception

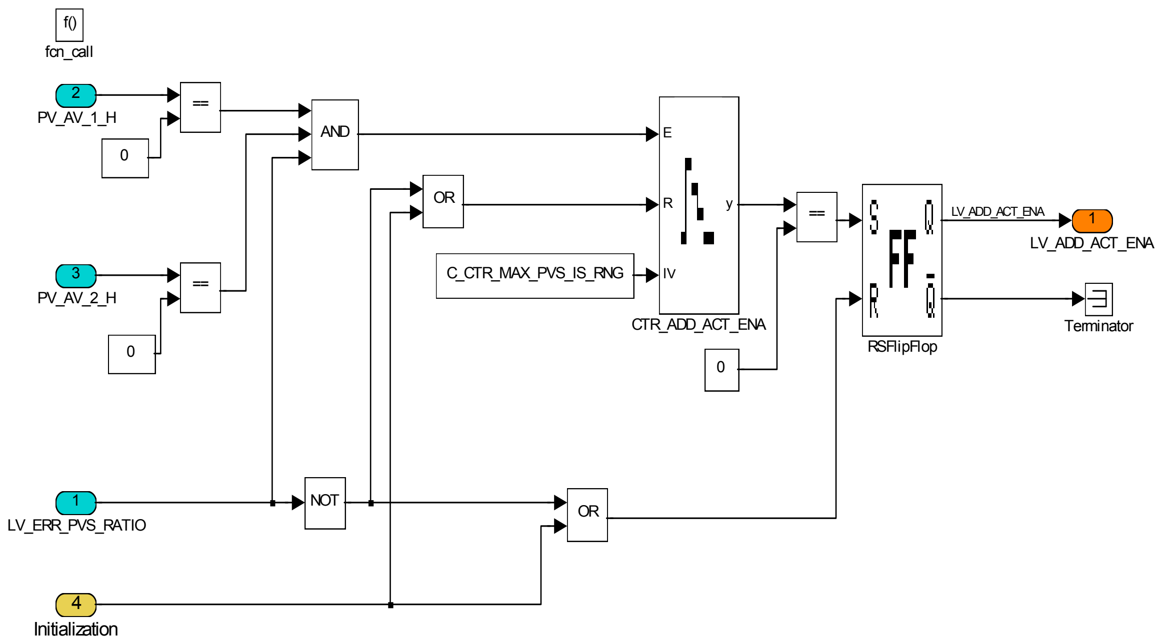


Figure 8: Exception

Description:

If a Ratio-Error is already debounced, the SRCL diagnosis conditions are not fulfilled until both pedal values PV_AV_1_H and PV_AV_2_H have been 0 % for the adjustable time C_CTR_MAX_PVS_IS_RNG. If the exception condition is fulfilled, LV_ADD_ACT_ENA will be set for the whole driving cycle and the SRCL can also be performed even if there is a high resistance and ratio-error.

Example:

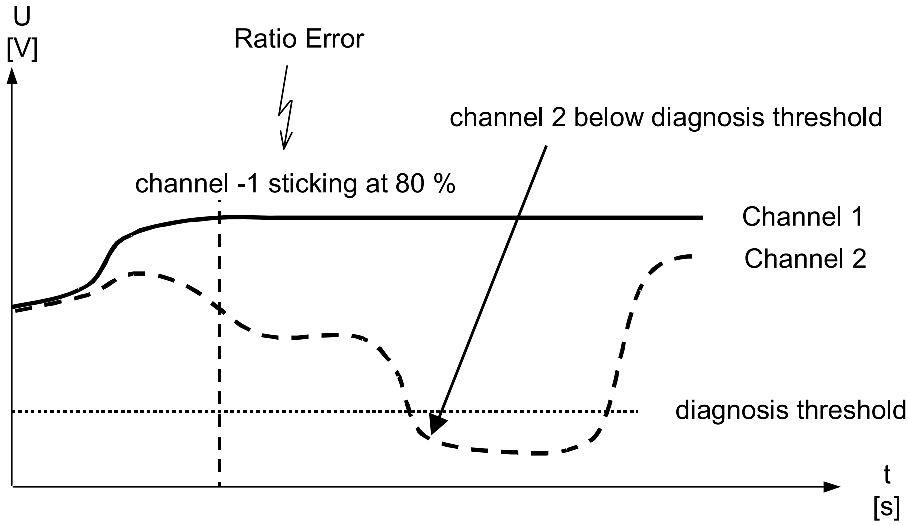


Figure 9: Example

If there is a Ratio-Error debounced **only**, (e.g. due to a sticking channel 1 to PV_AV_1_H = 80 %), there is no information about which channel is faulty.

To avoid a activation and debouncing of Signal Range Check Low if channel 2 underspends the diagnosis threshold as a result of high resistance (after a Ratio Error is already debounced), both channels have to be detected at idle PV_AV_1_H = 0% and PV_AV_2_H = 0% for an adjustable durance to enable.

Formula section:

```
IF          LV_ERR_PVS_RATIO        == 1                                   AND
            INITIALIZATION          == passive
THEN
    IF      PV_AV_1_H == 0                                                 AND
            PV_AV_2_H == 0                                                 AND
            CTR_ADD_ACT_ENA(K) > 0
    THEN    CTR_ADD_ACT_ENA(K) = CTR_ADD_ACT_ENA(K-1) – [update-rate]
    ENDIF

    IF      CTR_ADD_ACT_ENA(K) == 0
    THEN    LV_ADD_ACT_ENA = 1
    ENDIF

ELSE        CTR_ADD_ACT_ENA(K) = C_CTR_MAX_PVS_IS_RNG
            LV_ADD_ACT_ENA = 0
ENDIF
```

CDN SRCL 1

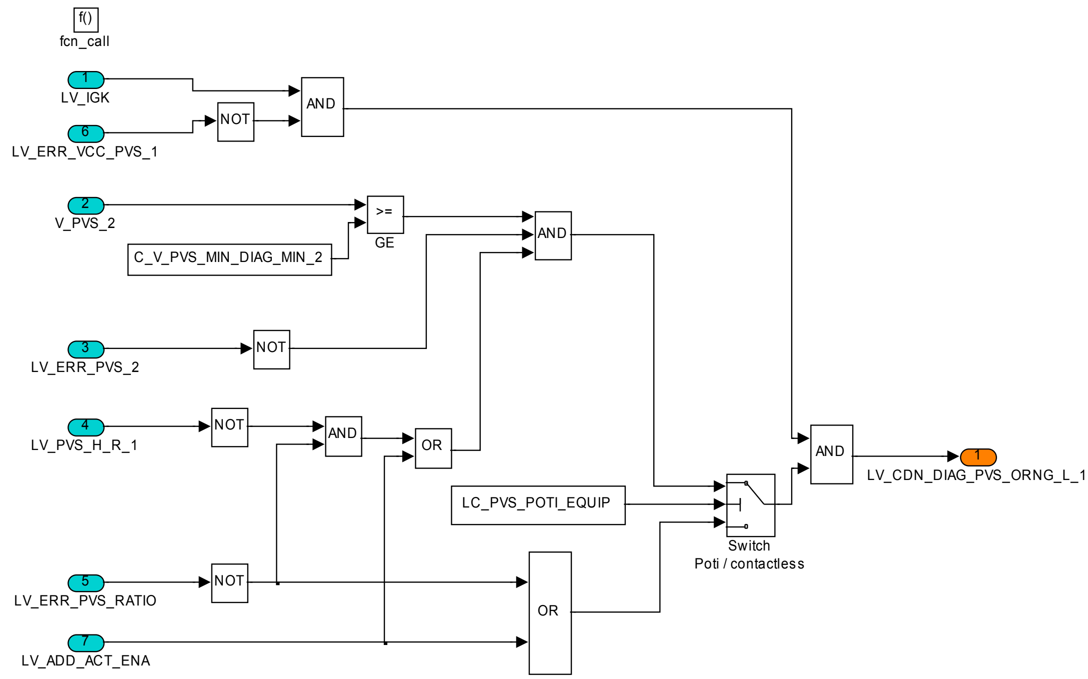


Figure 10: Diagnosis condition channel 1

<u>Description:</u>

Depending on sensor component (potentiometer or contactless) there are different diagnosis conditions. A potentiometer sensor may get disposits at the reversing points which lead to high resistances. Consequently the sensor voltage could fall below the diagnosis thresholds C_V_PVS_MIN_DIAG_1 temporary.

<u>Formula section:</u>

IF LV_IGK ==1 **AND** LV_ERR_VCC_PVS_1 = 0
THEN
IF LC_PVS_POTI_EQUIP == 1 (potentiometer systems)
THEN
IF V_PVS_2 >= C_V_PVS_MIN_DIAG_MIN_2 **AND**
LV_ERR_PVS_2 == 0 **AND**
{ (LV_PVS_H_R_1 == 0 **and** LV_ERR_PVS_RATIO == 0) **or**
LV_ADD_ACT_ENA == 1 **}**
THEN LV_CDN_DIAG_PVS_L_1 = 1
ELSE LV_CDN_DIAG_PVS_L_1 = 0
ENDIF
ELSE (contactless)
IF LV_ERR_PVS_RATIO == 0 **OR**
LV_ADD_ACT_ENA == 1
THEN LV_CDN_DIAG_PVS_L_1 = 1
ELSE LV_CDN_DIAG_PVS_L_1 = 0
ENDIF
ENDIF
ENDIF

CDN SRCL 2

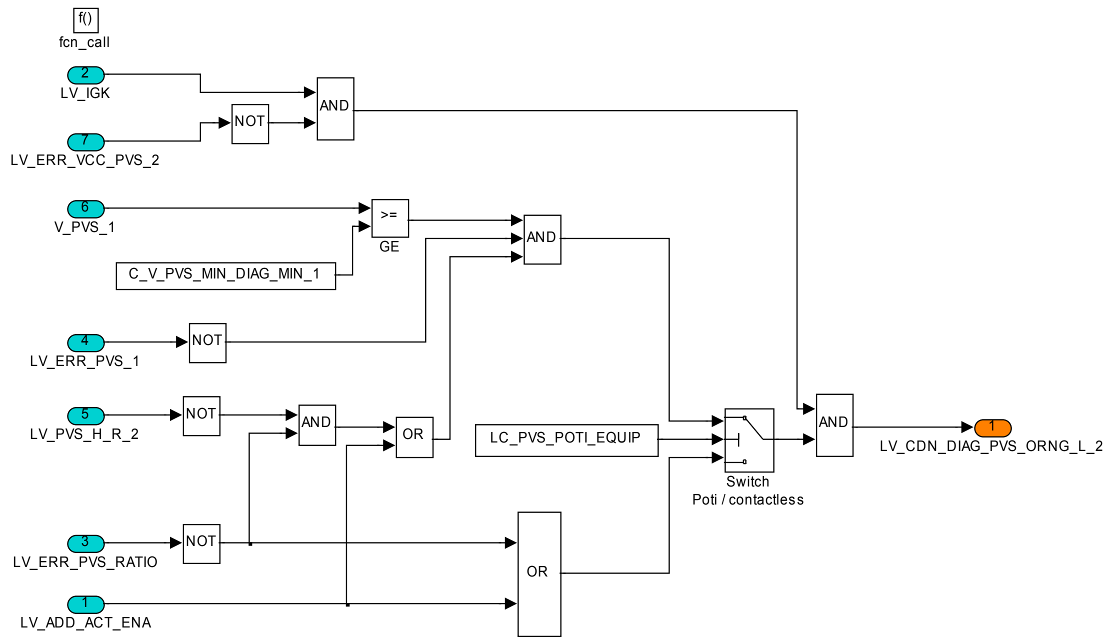


Figure 11: Diagnosis condition channel 2

Description:

Depending on sensor component (potentiometer or contactless) there are different diagnosis conditions. A potentiometer sensor may get disposits at the reversing points which lead to high resistances. Consequently the sensor voltage could fall below the diagnosis thresholds C_V_PVS_MIN_DIAG_2 temporary.

Formula section:

```
IF        LV_IGK ==1 AND LV_ERR_VCC_PVS_2 = 0
THEN
    IF    LC_PVS_POTI_EQUIP == 1          (potentiometer systems)
    THEN
          IF    V_PVS_1  >= C_V_PVS_MIN_DIAG_MIN_1                     AND
                LV_ERR_PVS_1 == 0                                      AND
                { (LV_PVS_H_R_2 == 0  and   LV_ERR_PVS_RATIO == 0)  or
                LV_ADD_ACT_ENA == 1 }
          THEN  LV_CDN_DIAG_PVS_L_2 = 1
          ELSE  LV_CDN_DIAG_PVS_L_2= 0
          ENDIF
    ELSE                                  ( contactless)
          IF    LV_ERR_PVS_RATIO == 0                                  OR
                LV_ADD_ACT_ENA == 1
          THEN  LV_CDN_DIAG_PVS_L_2 = 1
          ELSE  LV_CDN_DIAG_PVS_L_2 = 0
          ENDIF

    ENDIF
ENDIF
```

A.45.1.2.2 Symptom SRCL:

Description:
In an error-free system, the sensor voltages must lie above the limits C_V_PVS_MIN_DIAG_x.

Formula section:

Error Symptom Channel 1:

```
IF      V_PVS_1 < C_V_PVS_MIN_DIAG_1
THEN    ERR_SYM_PVS_L_1 = 2H   (short to ground or open line)
        //anti-bounce counter (PVS_L_1) + C_ABC_INC_PVS_ORNG_1, only one increment
        per activated time condition, LV_ERR_PVS_L_1 = 1 after debounced
ELSE    ERR_SYM_PVS_L_1 = 0H   (NO_SYM)
ENDIF
```

Filtering:

APPLY filter on current symptom

IF filtering result available (after debounce)

THEN LV_ERR_PVS_L_1 = filtering result

LV_END_DIAG_PVS_L_1 = 1

Deliver the result to Error Management

END

Error Symptom Channel 2:

```
IF      V_PVS_2 < C_V_PVS_MIN_DIAG_2
THEN    ERR_SYM_PVS_L_2 = 2H   (short to ground or open line)
        //anti-bounce counter (PVS_L_2) + C_ABC_INC_PVS_ORNG_2, only one increment
        per activated time condition, LV_ERR_PVS_L_2 = 1 after debounced
ELSE    ERR_SYM_PVS_L_2 = 0H   (NO_SYM)
ENDIF
```

Filtering:

APPLY filter on current symptom

IF filtering result available (after debounce)

THEN LV_ERR_PVS_L_2 = filtering result

LV_END_DIAG_PVS_L_2 = 1

Deliver the result to Error Management

END

A.45.1.3 ORNG Check Error Flags

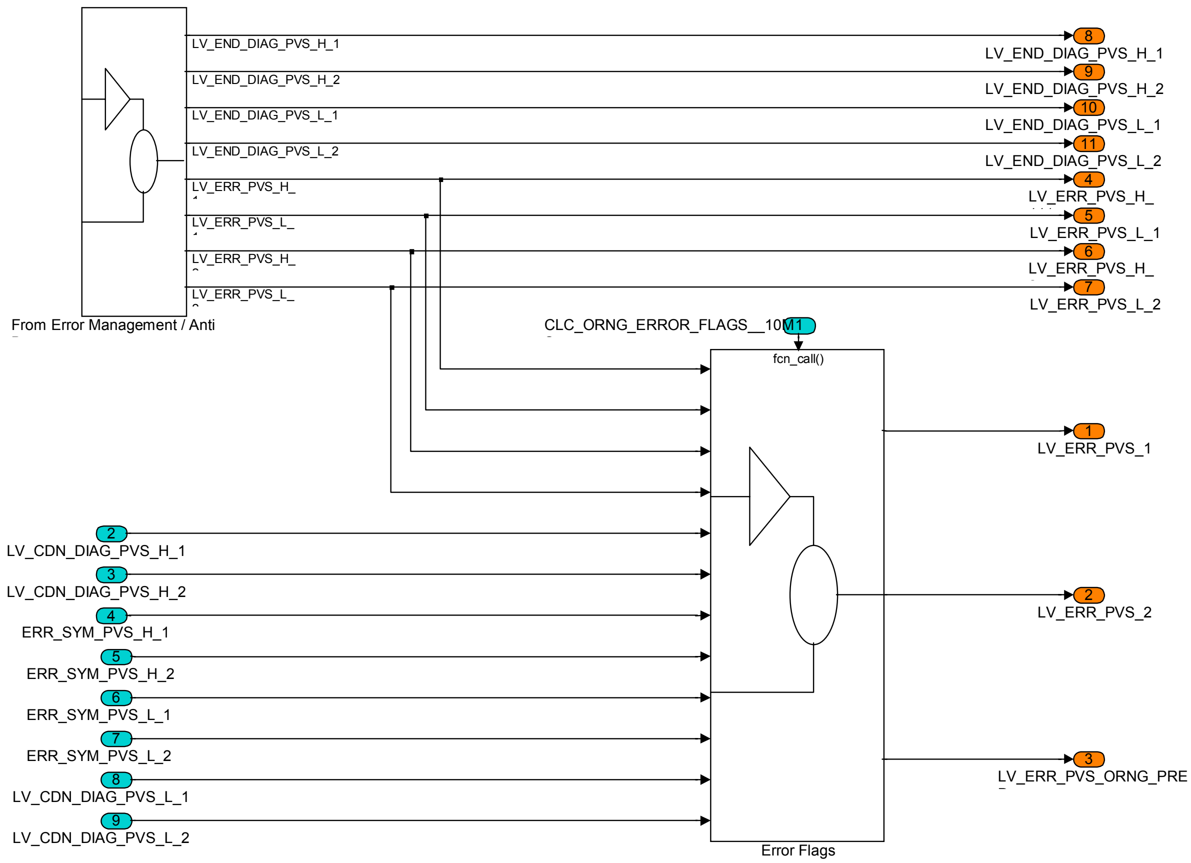


Figure 12: ORNG Check Error Flags

Description:

After debouncing the following error flags are set for the whole driving cycle:

LV_ERR_PVS_H_1

LV_ERR_PVS_H_2

LV_ERR_PVS_L_1

LV_ERR_PVS_L_2

The end of diagnosis flags are set when, the diagnosis result is available or the present error is debounced. They are reset at initialization.

LV_END_DIAG_PVS_H_1

LV_END_DIAG_PVS_H_2

LV_END_DIAG_PVS_L_1

LV_END_DIAG_PVS_L_2

Error Flags

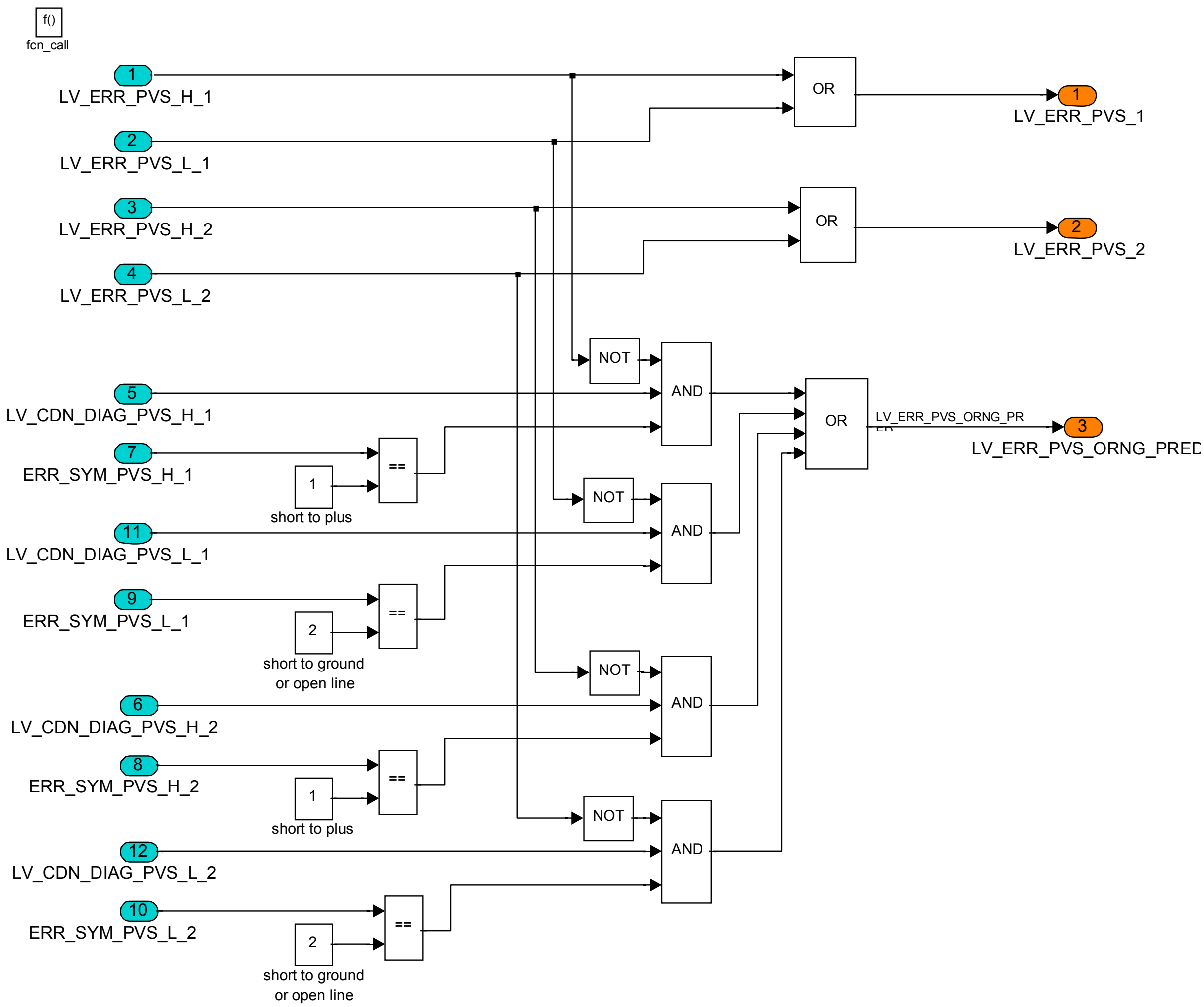


Figure 13: Errors ORNG

The error flags of SRCH and SRCL are combined to build the ORNG error status of each channel.

If a signal voltage is detected as out of range (symptom short to plus or to ground) and the diagnosis is active (LV_CDN_DIAG_PVS_ORNG_x_x ==1), the bit LV_ERR_PVS_ORNG_PRED will be set during debouncing. The bit will be reset, if

– the setting conditions are not fulfilled anymore (diagnosis condition + symptom) or
– after the error is debounced, LV_ERR_PVS_x = 1

Formula section:

ORNG Error debounced on PVS 1:

```
IF      LV_ERR_PVS_H_1 == 1     OR
        LV_ERR_PVS_L_1 == 1
THEN    LV_ERR_PVS_1 = 1
ENDIF
```

ORNG Error debounced on PVS 2:

```
IF      LV_ERR_PVS_H_2 == 1     OR
        LV_ERR_PVS_L_2 == 1
THEN    LV_ERR_PVS_2 =1
ENDIF
```

ORNG Error Predicted

```
IF      [       LV_CDN_DIAG_PVS_H_1 ==1         and
                ERR_SYM_PVS_H_1 == 1H and
          NOT(LV_ERR_PVS_H_1 ==1)                       ]       OR
        [       LV_CDN_DIAG_PVS_L_1 ==1         and
                ERR_SYM_PVS_L_1 == 2H and
          NOT(LV_ERR_PVS_L_1 ==1)                       ]       OR
        [       LV_CDN_DIAG_PVS_H_2 ==1         and
                ERR_SYM_PVS_H_2 == 1H and
          NOT(LV_ERR_PVS_H_2 ==1)                       ]       OR
        [       LV_CDN_DIAG_PVS_L_2 ==1         and
                ERR_SYM_PVS_L_2 == 2H and
          NOT(LV_ERR_PVS_L_2 ==1)                       ]
THEN    LV_ERR_PVS_ORNG_PRED = 1
ELSE    LV_ERR_PVS_ORNG_PRED = 0
ENDIF
```

A.45.2 Ratio Check

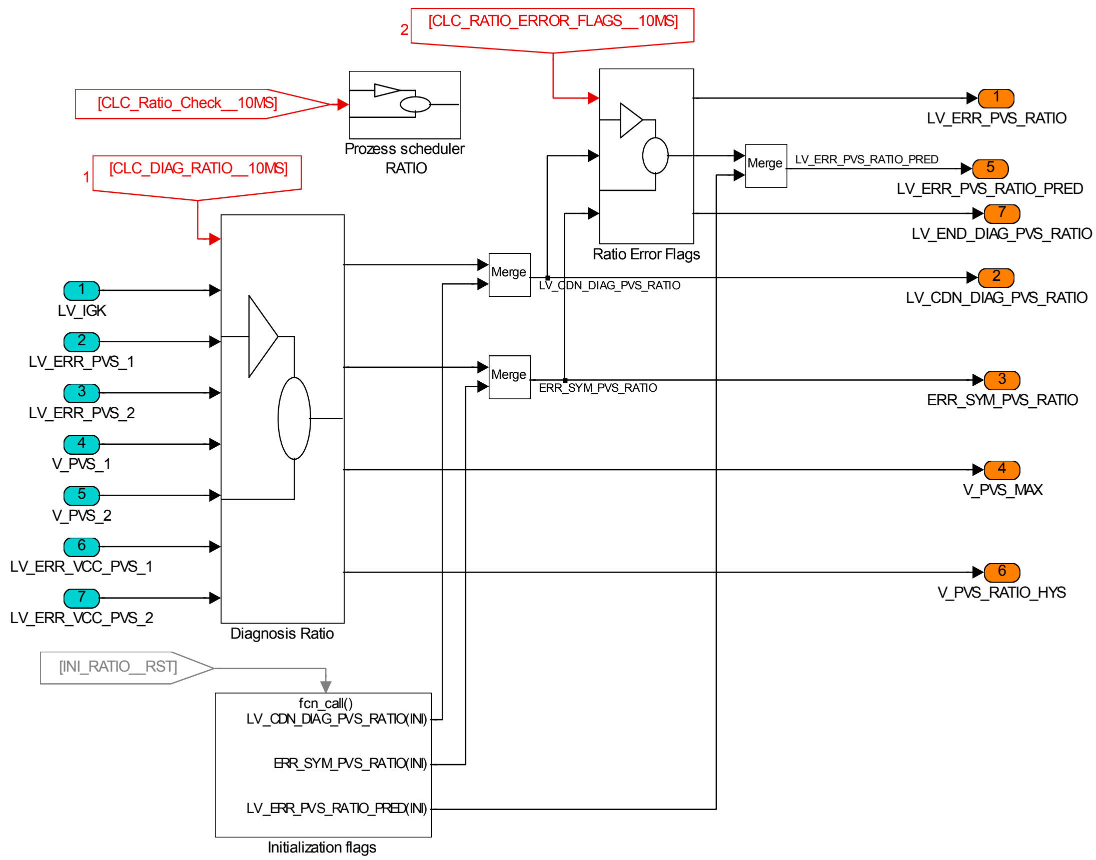


Figure 14: Ratio Check

Description:

The objective of the Ratio Check is to detect a relative deviation of the two PVS voltages.

If the voltages differ more than a permitted hysteresis, the bit LV_ERR_PVS_RATIO_PRED will be set during debouncing. The bit will be reset if:

- the setting conditions are not fulfilled anymore or
- the error is debounced, LV_ERR_PVS_RATIO = 1

In order to re-examine whether a range violation has caused a ratio error, the debounce time of the Ratio Check has to be higher than the debouncing time of the signal range checks (high / low).

Calculation order:

1. Diagnosis Ratio — on fcn-call CLC_DIAG_RATIO__10MS
2. Ratio Error Flags — on fcn-call CLC_RATIO_ERROR_FLAGS__10MS

A.45.2.1 Diagnosis Ratio

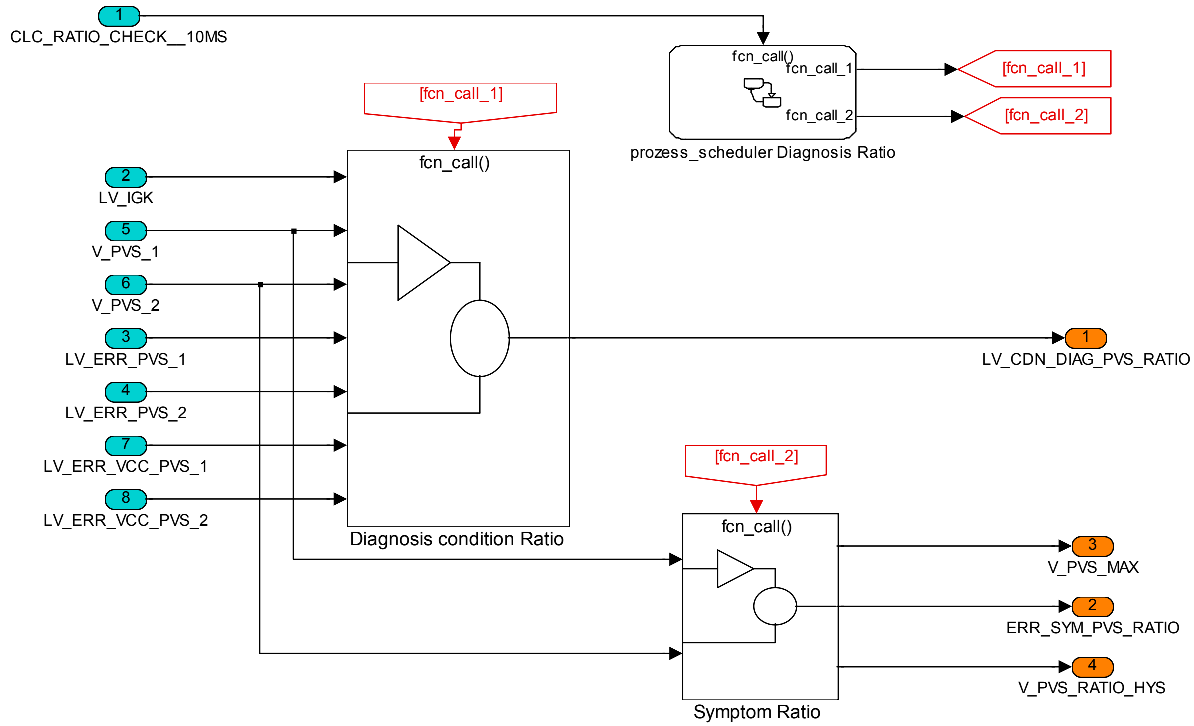


Figure 15: Diagnosis Ratio

Description:

Calculation order:

1. Diagnosis condition Ratio on fcn_call_1
2. Symptom Ratio on fcn_call_2

A.45.2.1.1 Diagnosis condition Ratio

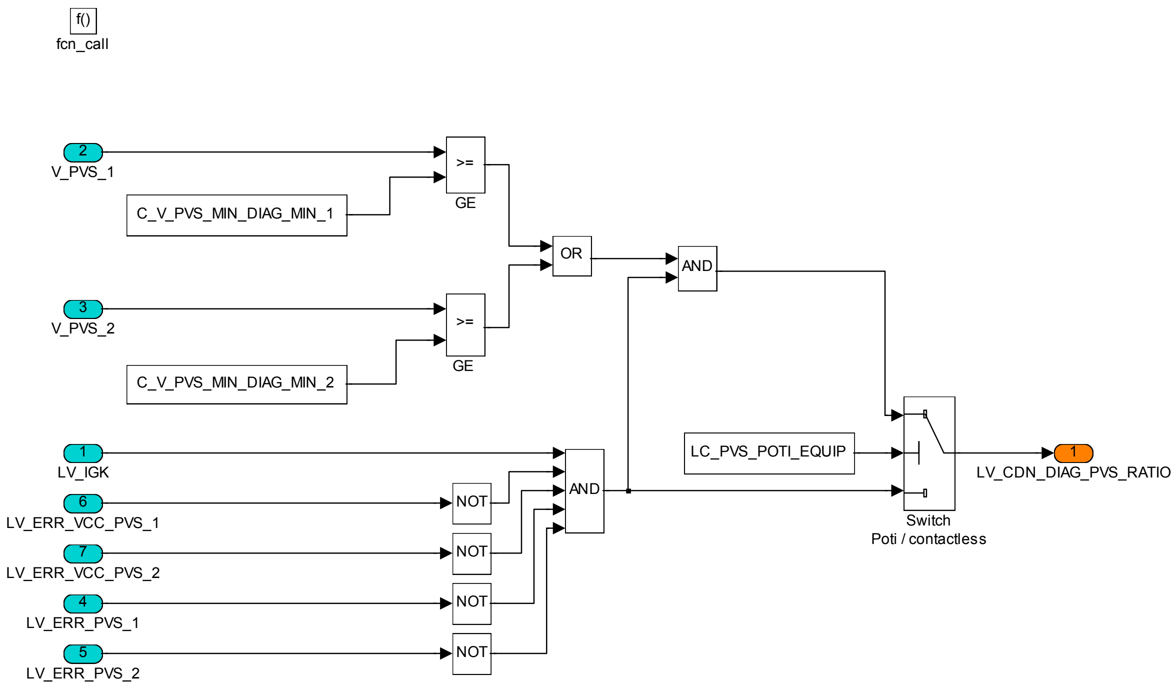


Figure 16: Diagnosis condition Ratio

Description:

As a consequence of different sensor system (potentiometer or contactless), different diagnosis conditions have to be taken into account.

For potentiometer systems (LC_PVS_POTI_EQUIP == 1)

At the lower reversing point a potentiometer may get disposits which lead to high contact resistances (high resistance). This may implicate that a sensor voltage drops.

Therefore the condition for Ratio Check is only fulfilled, if both channels are faultlessly and at least one sensor voltage V_PVS_x is greater or equal C_V_PVS_MIN_DIAG_MIN_x.

For contactless systems (LC_PVS_POTI_EQUIP == 0)

The condition is fulfilled if both channels are faultlessly.

Formula section:

```
IF    LV_IGK == 1                                         AND
      LV_ERR_PVS_1 == 0                                   AND
      LV_ERR_PVS_2 == 0                                   AND
      LV_ERR_VCC_PVS_1 == 0                               AND
      LV_ERR_VCC_PVS_2 == 0                               AND
THEN
      IF    LC_PVS_POTI_EQUIP == 0
      THEN  LV_CDN_DIAG_PVS_RATIO = 1
      ELSE
            IF    [   V_PVS_1 >= C_V_PVS_MIN_DIAG_MIN_1 or
                      V_PVS_2 >= C_V_PVS_MIN_DIAG_MIN_2   ]
            THEN  LV_CDN_DIAG_PVS_RATIO = 1
            ELSE  LV_CDN_DIAG_PVS_RATIO = 0
            ENDIF
      ENDIF
ELSE        LV_CDN_DIAG_PVS_RATIO = 0
ENDIF
```

A.45.2.1.2 Symptom Ratio

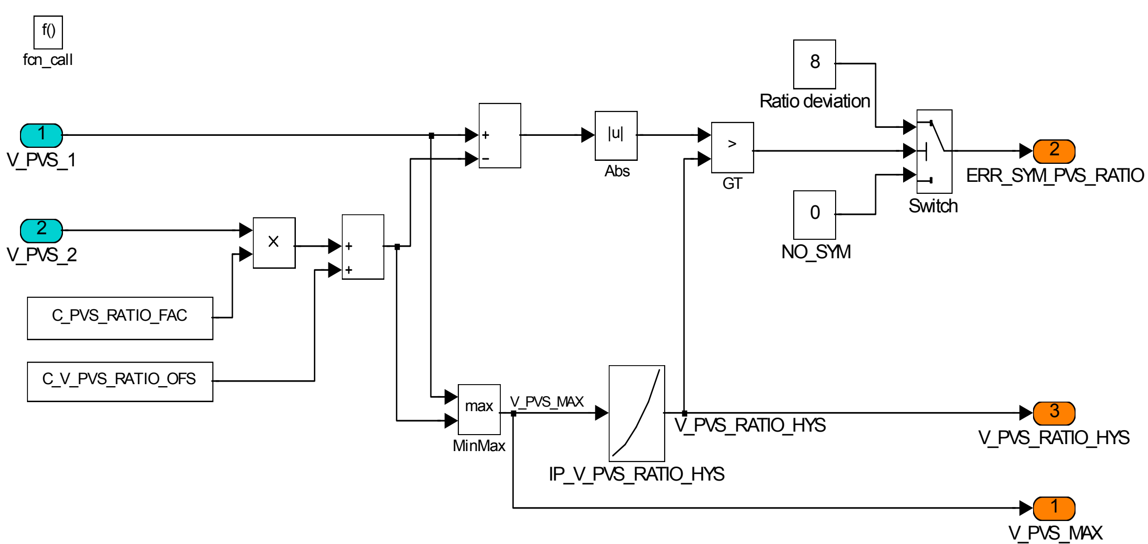


Figure 17: Symptom Ratio

Description:

At an error free system, the sensor voltages may not differ more than an allowed hysteresis depending on the actual gas pedal position. If the deviation of (V_PVS_1 and V_PVS_2 nominal) is higher than the allowed value the symptom “Ratio deviation” is detected.

To avoid that a Ratio Error will be debounced due to a high resistance at upper reversing point, the 1-dimensional map has to be calibrated in conformity with this effect.

Formula section:

V_PVS_MAX = **MAX (**V_PVS_1, (V_PVS_2 * C_PVS_RATIO_FAC +
C_V_PVS_RATIO_OFS)**)** {intermediate results are *0xFFFF* hex size}
{no overflow}

V_PVS_RATIO_HYS = IP_PVS_RATIO_HYS (V_PVS_MAX)

IF ABS **[** V_PVS_1- (V_PVS_2 * C_PVS_RATIO_FAC + C_V_PVS_RATIO_OFS)**]**
> V_PVS_RATIO_HYS {intermediate results are *0xFFFF* hex size}
{no overflow}

THEN ERR_SYM_PVS_RATIO = 8H (Ratio deviation)
//anti-bounce counter (PVS_RATIO) + C_ABC_INC_PVS_RATIO, only one increment per activated time condition, LV_ERR_RATIO = 1 after debounced

ELSE ERR_SYM_PVS_RATIO = 0H (NO_SYM)

Filtering:

APPLY filter on current symptom

IF filtering result available (after debounce)

THEN LV_ERR_PVS_RATIO = filtering result
LV_END_DIAG_PVS_RATIO = 1
Deliver the result to Error Management

A.45.2.2 Ratio Error Flags

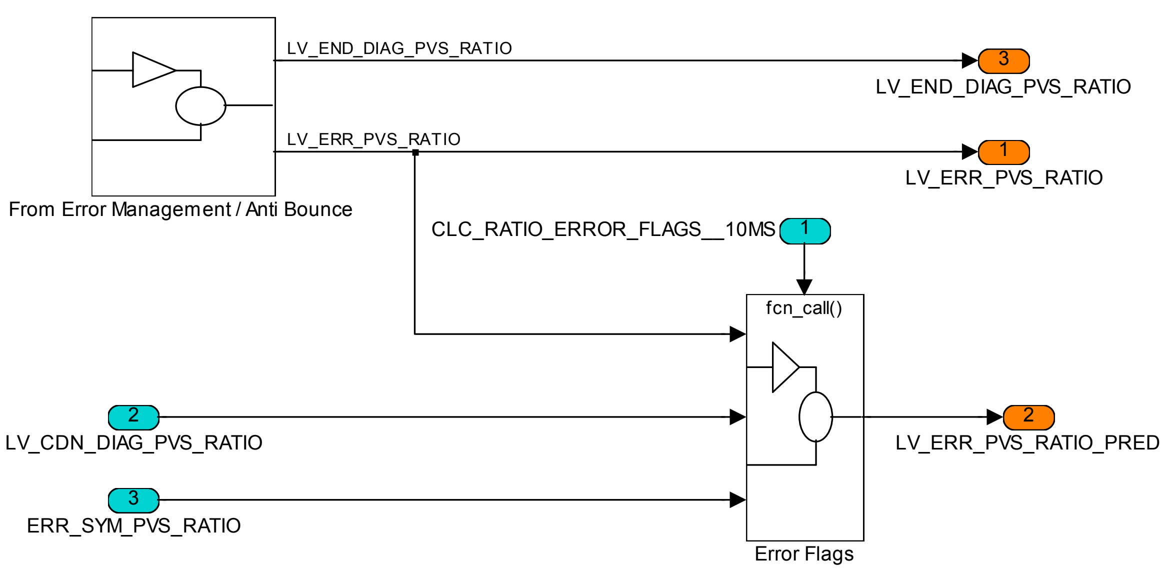


Figure 18: Ratio Error Flags

Description:

After debouncing the error flag LV_ERR_PVS_RATIO is set for the whole driving cycle

The end of diagnosis flag LV_END_DIAG_PVS_RATIO is set when,

- the diagnosis result is available
- the error is debounced

They are reset at initialization.

Error Flags

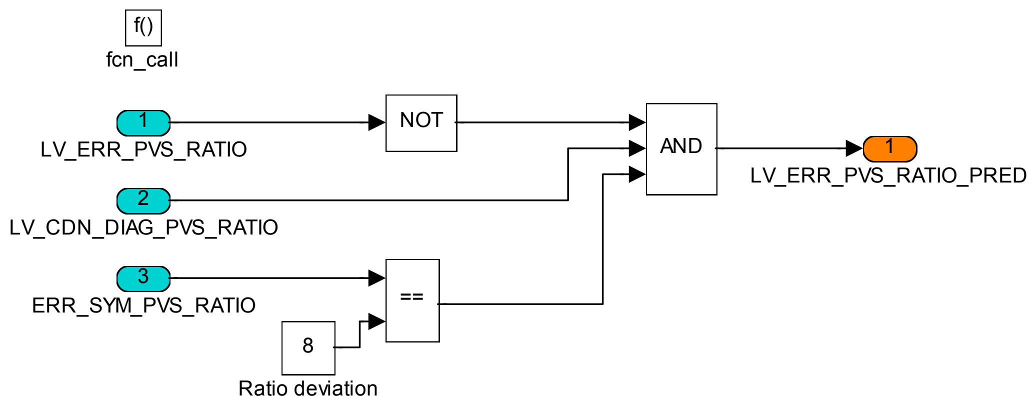


Figure 19: Error Flags

If the symptom is detected and the diagnosis is active (LV_CDN_DIAG_PVS_RATIO == 1), the bit LV_ERR_PVS_RATIO_PRED will be set during debouncing. The bit will be reset, if

- the setting conditions are not fulfilled anymore or
- after the error is debounced, LV_ERR_PVS_RATIO = 1

Formula section:

IF	LV_CDN_DIAG_PVS_RATIO == 1	**AND**
	ERR_SYM_PVS_RATIO == 8H	**AND**
	NOT (LV_ERR_PVS_RATIO==1)	
THEN	LV_ERR_PVS_RATIO_PRED = 1	
ELSE	LV_ERR_PVS_RATIO_PRED = 0	
ENDIF		

A.45.3 High Resistance Check

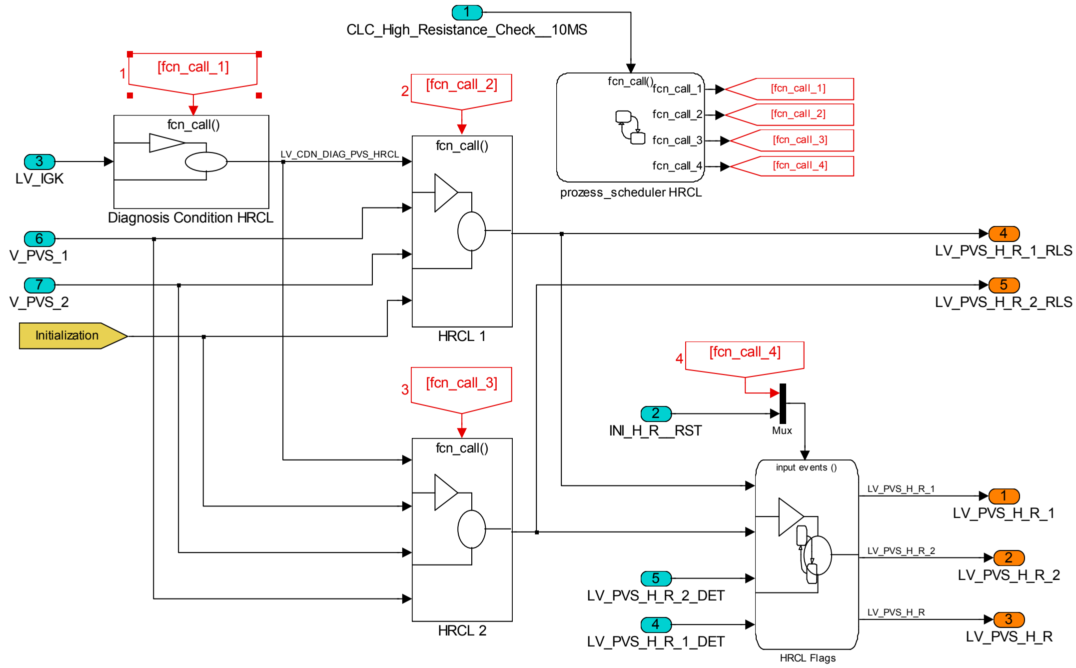


Figure 20: High Resistance Check

Description:

A potentiometer system may get high contact resistances at upper and lower reversing points as a result of settled abrasions.

The intend of this diagnosis function is to detect high resistance at the lower reversing points to differentiate a voltage drop due to

- high resistance or
- other faults (out of range or ratio)

to prevent an unintentional error recognition.

If a contactless PVS sensor is applied, this function can be disabled setting LC_PVS_POTI_EQUIP == 0

If a high resistance is recognized on channel x, the bit LV_PVS_H_R_x is set. The bit LV_PVS_H_R_x_RLS is set additionally to have an information in the next driving cycle whether a high resistance was detected in the previous (calculation of LV_PVS_H_R_x_DET at initialization on the basis of LV_PVS_H_R_x_RLS in last driving cycle).

If a high resistance is recognized on both channels, LV_PVS_H_R is set.

Calculation order:

1. Diagnosis Condition HRCL (High Resistance Check Low) on fcn_call_1
2. HRCL 1 (High Resistance Check Channel 1) on fcn_call_2
3. HRCL 2 (High Resistance Check Channel 2) on fcn_call_3
4. HRCL Flags on fcn_call_4

A.45.3.1 Diagnosis Condition HRCL (High Resistance Check Low)

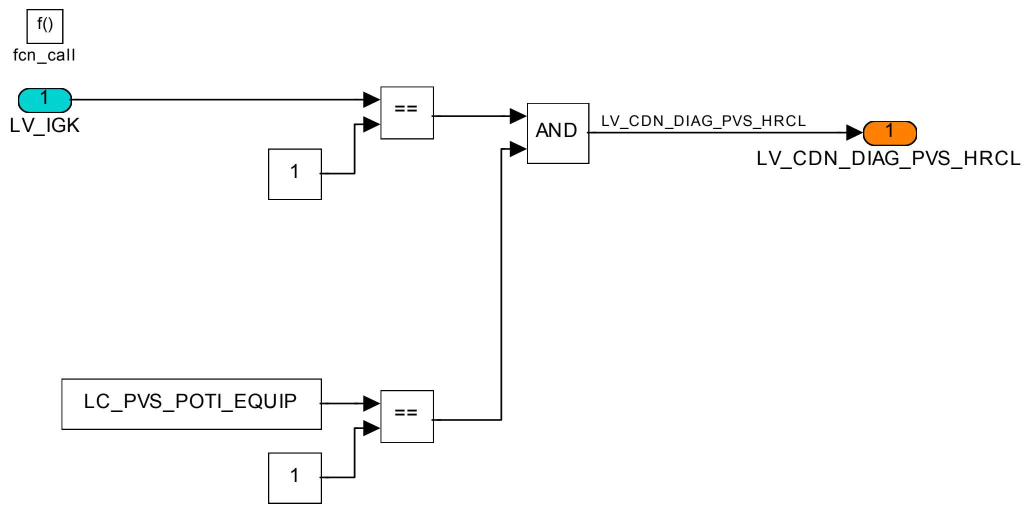


Figure 21: Diagnosis Condition HRCL

Description

The diagnosis condition is fulfilled if LV_IGK == 1 and the logical constant LC_PVS_POTI_EQUIP == 1. The High Resistance Check is performed **only** if the diagnosis condition is true (LV_CDN_DIAG_PVS_HRCL == 1).

Formula section:

IF LV_IGK == 1 **AND**
LC_PVS_POTI_EQUIP == 1
THEN LV_CDN_DIAG_PVS_HRCL = 1
ELSE LV_CDN_DIAG_PVS_HRCL = 0
ENDIF

A.45.3.2 HRCL 1 (High Resistance Check Channel 1)

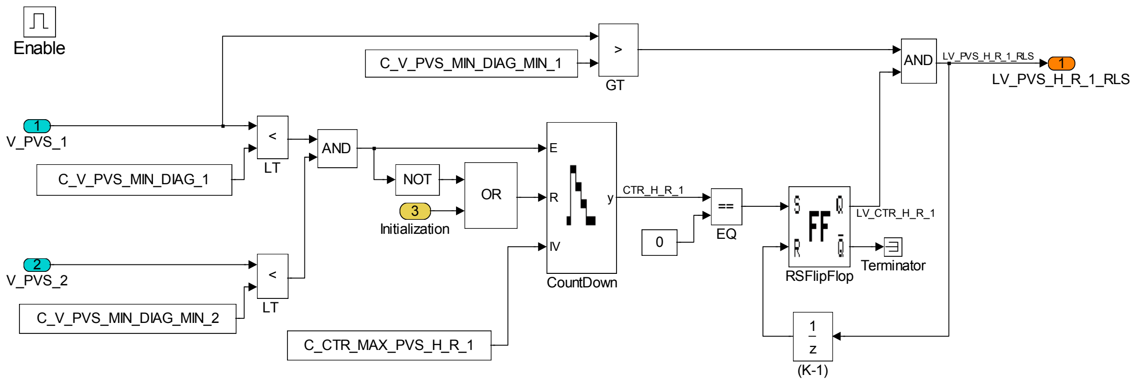


Figure 22: HRCL 1

Description:

If the signal voltage of channel 1 underspends the threshold C_V_PVS_MIN_DIAG_1 while channel 2 is underneath C_V_PVS_MIN_DIAG_MIN_2 the counter is decremented. If channel 1 exceeds the threshold C_V_PVS_MIN_DIAG_MIN_1 after counter has reached 0, a high contact resistance has been recognized (high resistance).

Formula section:

IF **NOT (**V_PVS_1 < C_V_PVS_MIN_DIAG_1 **and**
V_PVS_2 < C_V_PVS_MIN_DIAG_MIN_2**)** **OR**
Initialization
THEN CTR_H_R_1$_{(K)}$ = C_CTR_MAX_PVS_H_R_1
ELSE

IF CTR_H_R_1 $_{(K-1)}$ > 0
THEN CTR_H_R_1$_{(K)}$ = CTR_H_R_1 $_{(K-1)}$ – [update rate]
ENDIF

ENDIF

IF LV_PVS_H_R_1_RLS$_{(K-1)}$ == 0
THEN

IF CTR_H_R_1 $_{(K)}$ == 0
THEN LV_CTR_H_R_1 = 1
ENDIF

ELSE LV_CTR_H_R_1 = 0
ENDIF

IF LV_CTR_H_R_1 = =1 **AND**
V_PVS_1 > C_V_PVS_MIN_DIAG_MIN_1
THEN LV_PVS_H_R_1_RLS$_{(K)}$ = 1
ELSE LV_PVS_H_R_1_RLS$_{(K)}$ = 0
ENDIF

A.45.3.3 HRCL 2 (High Resistance Check Channel 2)

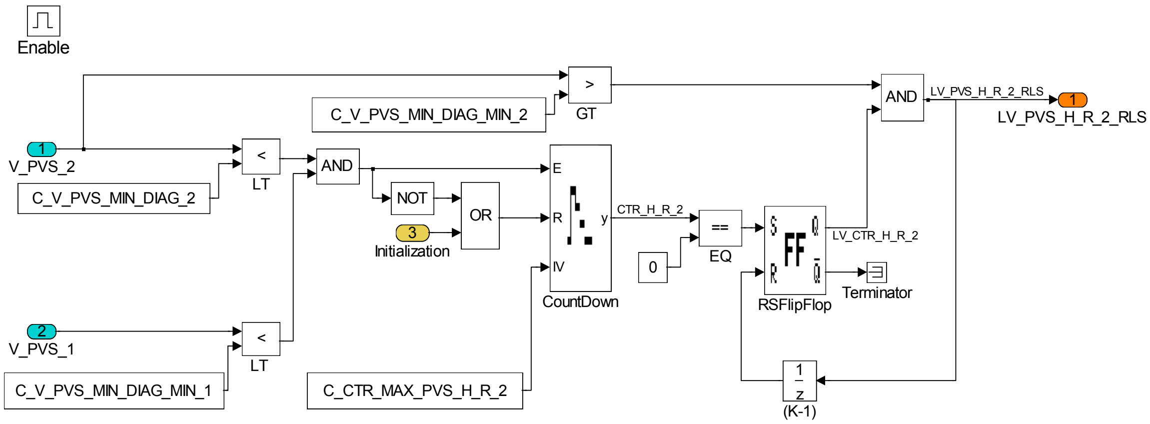


Figure 23: HRCL 2

Description:

If the signal voltage of channel 2 underspends the threshold C_V_PVS_MIN_DIAG_2 while channel 1 is underneath C_V_PVS_MIN_DIAG_MIN_1 the counter is decremented. If channel 2 exceeds the threshold C_V_PVS_MIN_DIAG_MIN_2 after counter has reached 0, a high contact resistance has been recognized (high resistance).

Formula section:

IF **NOT (**V_PVS_2 < C_V_PVS_MIN_DIAG_2 **and**
V_PVS_1 < C_V_PVS_MIN_DIAG_MIN_1**)** **OR**
Initialization
THEN CTR_H_R_2$_{(K)}$ = C_CTR_MAX_PVS_H_R_2
ELSE

IF CTR_H_R_2 $_{(K-1)}$ > 0
THEN CTR_H_R_2$_{(K)}$ = CTR_H_R_2 $_{(K-1)}$ – [update rate]
ENDIF

ENDIF

IF LV_PVS_H_R_2_RLS$_{(K-1)}$ == 0
THEN

IF CTR_H_R_2 $_{(K)}$ == 0
THEN LV_CTR_H_R_2 = 1
ENDIF

ELSE LV_CTR_H_R_2 = 0
ENDIF

IF LV_CTR_H_R_2 = =1 **AND**
V_PVS_2 > C_V_PVS_MIN_DIAG_MIN_2
THEN LV_PVS_H_R_2_RLS$_{(K)}$ = 1
ELSE LV_PVS_H_R_2_RLS$_{(K)}$ = 0
ENDIF

A.45.3.4 HRCL Flags

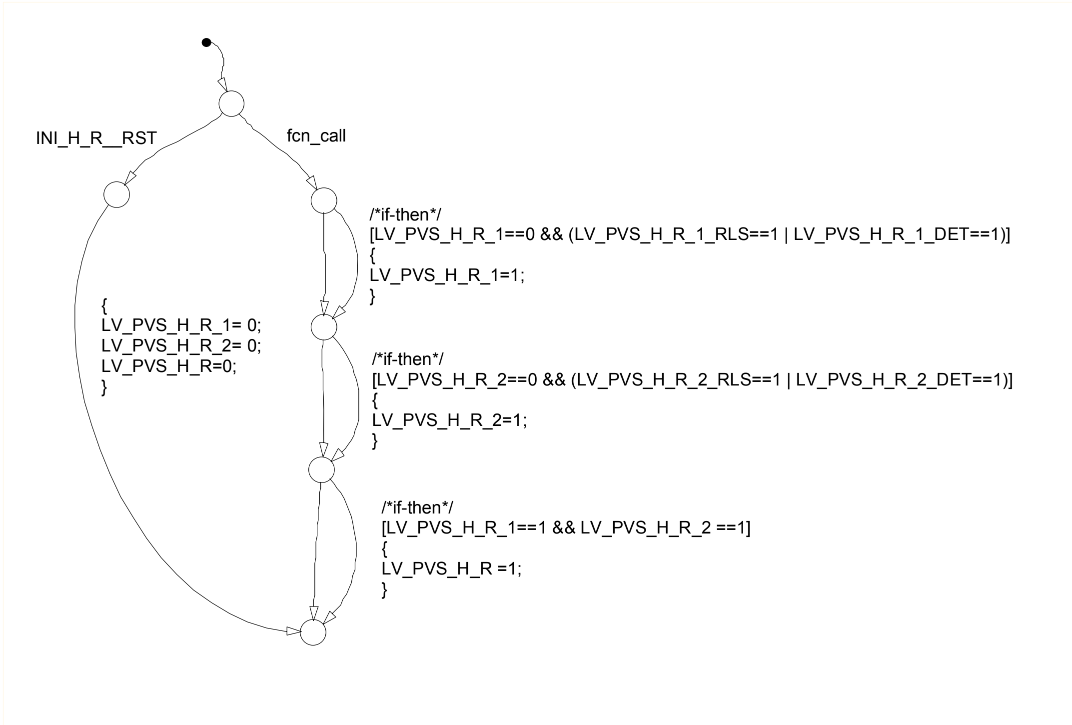


Figure 24: HRCL Flags

Description:

At Initialization the bits LV_PVS_H_R_1, LV_PVS_H_R_2, LV_PVS_H_R are set = 0.

Regular calculation is done as shown above. If a high contact resistance is recognized actually (LV_PVS_H_R_x_RLS == 1) or has been detected in the last driving cycle (LV_PVS_H_R_x_DET ==1), the bit LV_PVS_H_R_x is set respectively. If high resistance is recognized on **both** channels, LV_PVS_H_R is set additionally.

A.45.4 Move Check

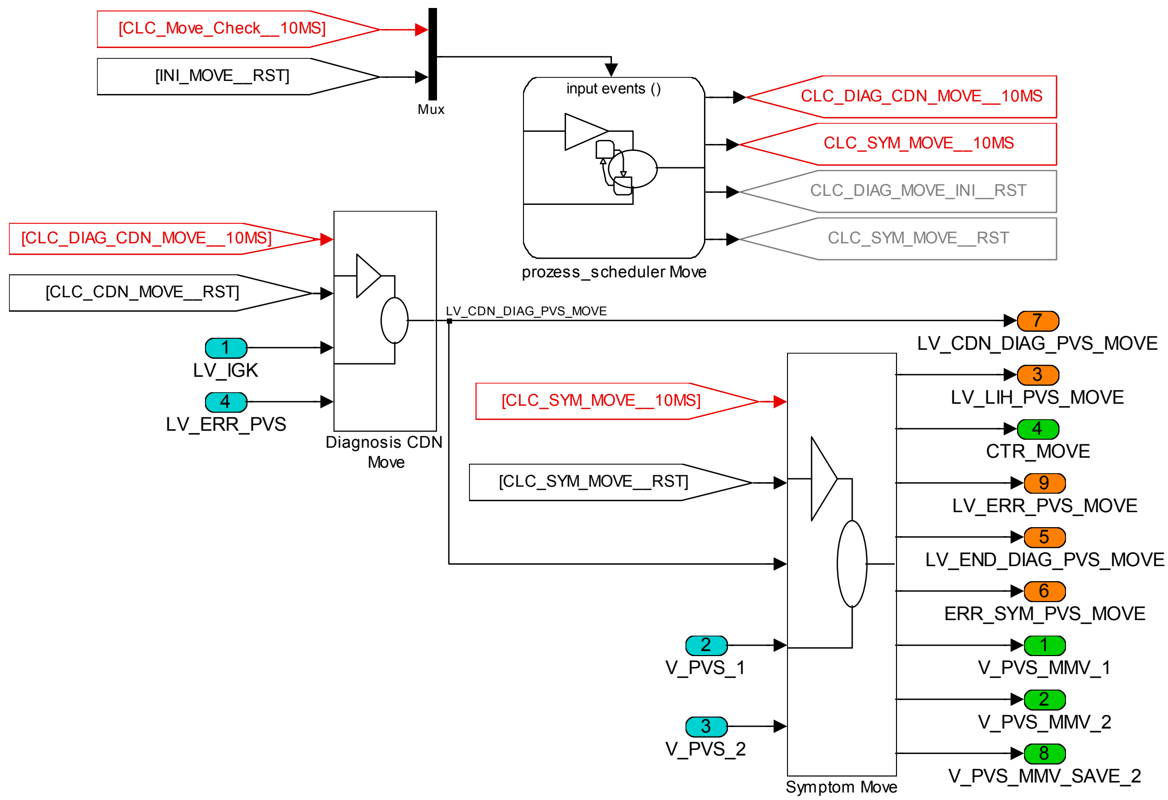


Figure 25: Move Check

Description:

The Move Check is based on the fact that both pedal value sensors are located on one shaft. A change of one channel is tied to the other one. If a motion of only one channel is detected LV_LIH_PVS_MOVE is set temporary. A counter is incremented at every recognition. If the counter reaches the maximum value LV_ERR_PVS_MOVE and LV_LIH_PVS_MOVE are set to 1 permanently.

Calculation order:

1. Diagnosis CDN Move — on fcn-call CLC_DIAG_CDN_MOVE__10MS
2. Symptom Move — on fcn-call CLC_SYM_MOVE__10MS

Initialization order:

1. Diagnosis CDN Move — on fcn-call CLC_CDN_MOVE__RST
2. Symptom Move — on fcn-call CLC_SYM_MOVE__RST

A.45.4.1 Diagnosis CDN Move

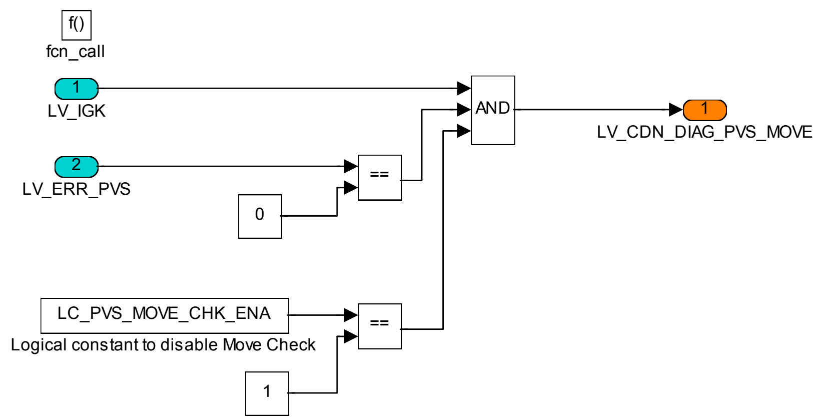


Figure 26: Diagnosis Condition Move

Description:

Regular calculation:

The Move Check is **only** executed (LV_CDN_DIAG_PVS_MOVE = 1) if, LV_IGK = 1, no PVS error is already debounced LV_ERR_PVS = 0 and LC_PVS_MOVE_CHK_ENA = 1.

At Initialization LV_CDN_DIAG_PVS_MOVE is set = 0.

A.45.4.2 Symptom Move

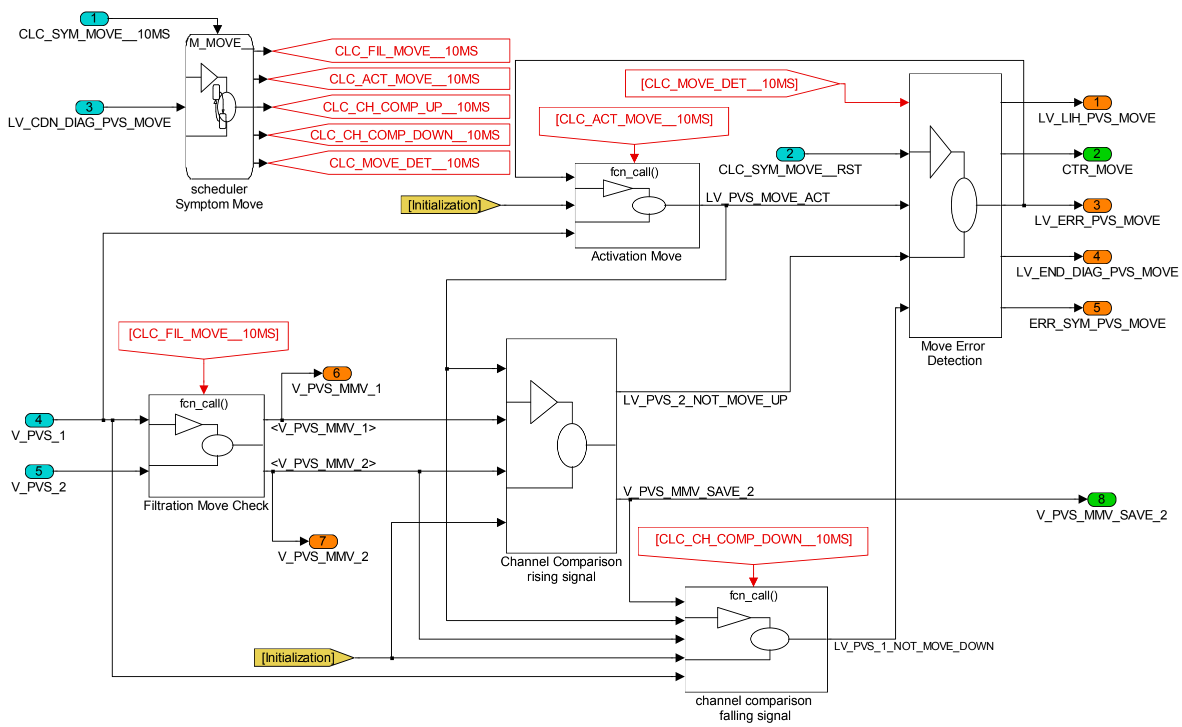


Figure 27: Symptom Move

Description:

Calculation order:

1. Filtration Move Check — on fcn-call CLC_FIL_MOVE__10MS
2. Activation Move — on fcn-call CLC_ACT_MOVE__10MS
3. Channel comparison rising signal — on fcn-call CLC_CH_COMP_UP__10MS
4. Channel comparison falling signal — on fcn-call CLC_CH_COMP_DOWN__10MS
5. Move Error Detection — on fcn_call CLC_MOVE_DET__10MS

A.45.4.2.1 Filtration Move Check

Description:

In order to consider of dynamic behavior, the sensor voltages have to be filtered.

$$V\_PVS\_MMV\_1_{(K)} = V\_PVS\_MMV\_1_{(K-1)} * (1 - C\_CRLC\_PVS\_1\_MOVE) + V\_PVS\_1 * C\_CRLC\_PVS\_1\_MOVE$$

$$V\_PVS\_MMV\_2_{(K)} = V\_PVS\_MMV\_2_{(K-1)} * (1 - C\_CRLC\_PVS\_2\_MOVE) + V\_PVS\_2 * C\_CRLC\_PVS\_2\_MOVE$$

A.45.4.2.2 Activation Move

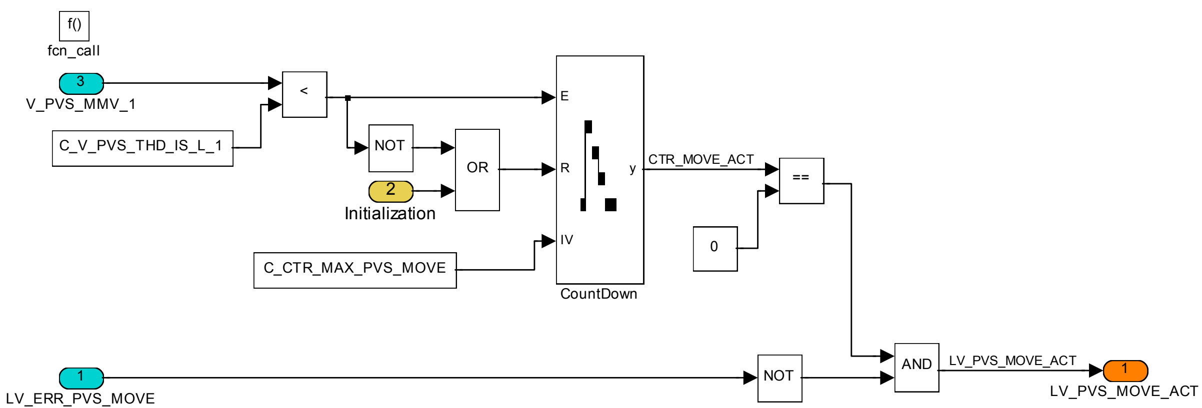


Figure 28: Activation Move

Description:

For activation or re-activation of the diagnosis, the filtered voltage of channel 1 has to be recognized underneath the idle speed threshold C_V_PVS_THD_IS_L_1 for C_CTR_MAX_PVS_MOVE and <u>no</u> Move Error is already debounced.

Formula section:

IF **NOT** (V_PVS_MMV_1 < C_V_PVS_THD_IS_L_1) **OR**
Initialization
THEN CTR_MOVE_ACT$_{(K)}$ = C_CTR_MAX_PVS_MOVE
ELSE

  IF CTR_MOVE_ACT$_{(K-1)}$ > 0
  THEN CTR_MOVE_ACT$_{(K)}$ = CTR_MOVE_ACT$_{(K-1)}$ – [update rate]
  ENDIF

ENDIF

IF CTR_MOVE_ACT$_{(K)}$ == 0 **AND**
NOT (LV_ERR_PVS_MOVE==1)
THEN LV_PVS_MOVE_ACT = 1
ELSE LV_PVS_MOVE_ACT = 0
ENDIF

A.45.4.2.3 Channel comparison rising signal

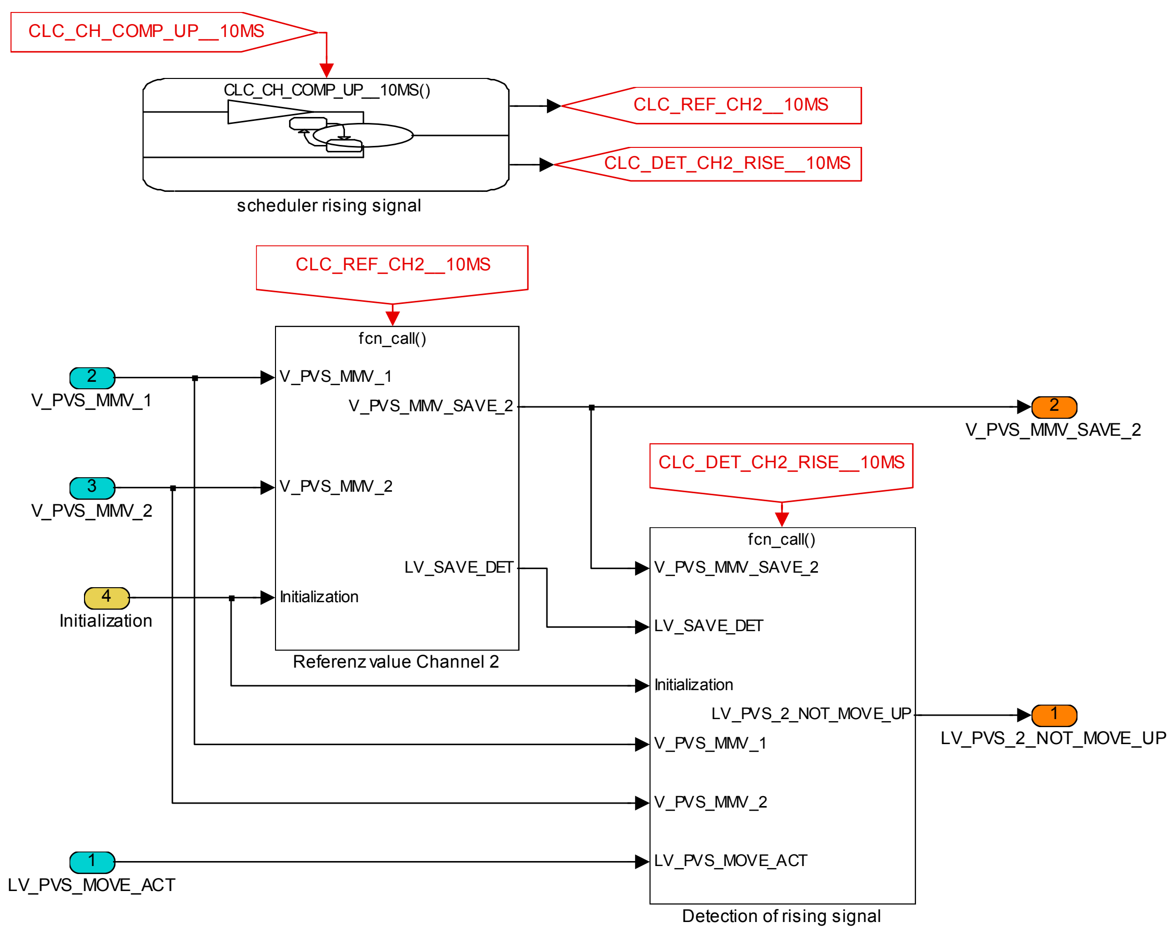


Figure 29: Channel comparison rising signal

Description:

Calculation order:

1. Referenz value Channel 2 on fcn-call CLC_REF_CH2__10MS
2. Detection of rising signal on fcn-call CLC_DET_CH2_RISE__10MS

Referenz value Channel 2:

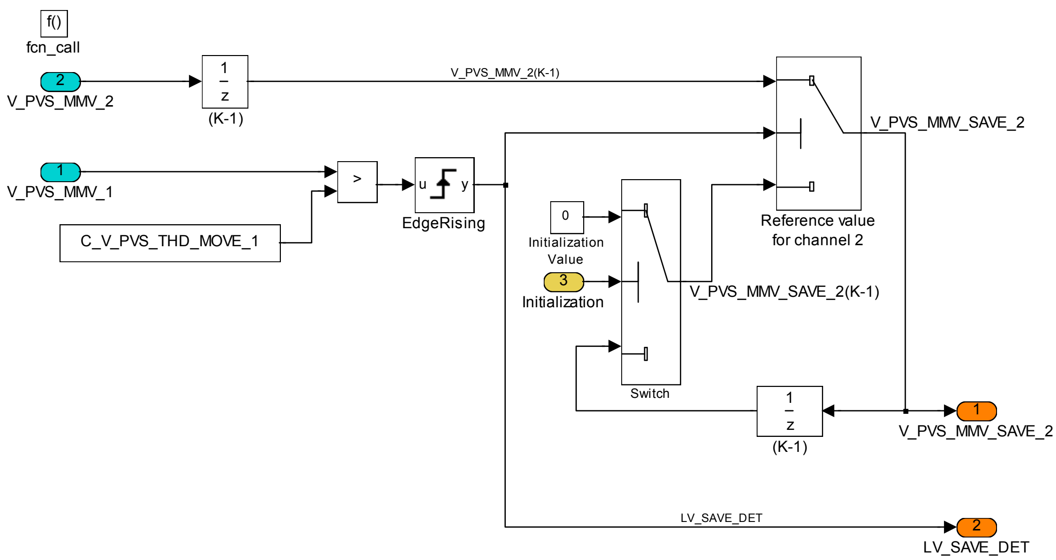


Figure 30: Referenz value channel 2

If PVS-Channel 1 has moved from idle speed position and the filtered voltage V_PVS_MMV_1 exceeds C_V_PVS_THD_MOVE_1, the previous filtered voltage $V\_PVS\_MMV\_2_{(k-1)}$ of channel 2 is saved to V_PVS_MMV_SAVE_2.

Formula section:

At Initialization $V\_PVS\_MMV\_SAVE\_2_{(K-1)} = 0$; V_PVS_MMV_2 = 0

IF $(V\_PVS\_MMV\_1_{(K-1)} <= C\_V\_PVS\_THD\_MOVE\_1)$ **AND**
$(V\_PVS\_MMV\_1_{(K)} > C\_V\_PVS\_THD\_MOVE\_1)$

THEN $V\_PVS\_MMV\_SAVE\_2_{(K)} = V\_PVS\_MMV\_2_{(k-1)}$
LV_SAVE_DET = 1

ELSE $V\_PVS\_MMV\_SAVE\_2_{(K)} = V\_PVS\_MMV\_SAVE\_2_{(K-1)}$
LV_SAVE_DET = 0

ENDIF

Detection of rising signal:

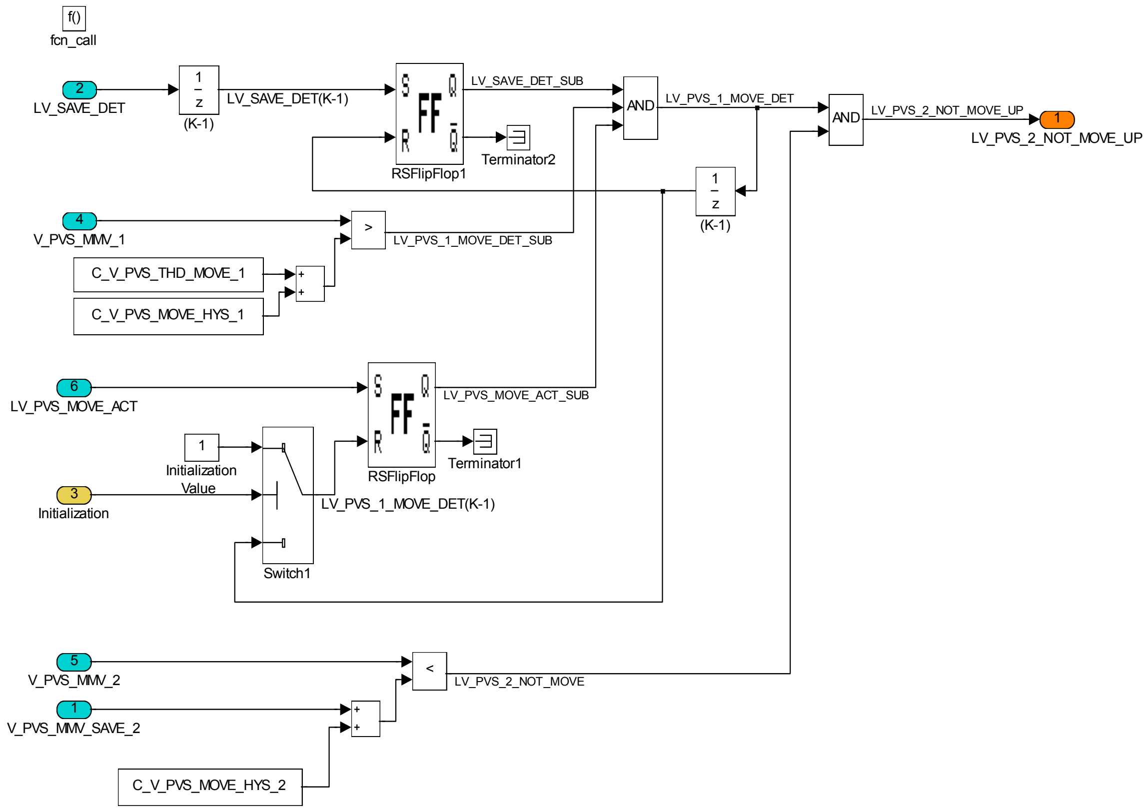


Figure 31: Detection of rising signal

If PVS-Channel 1 exceeds C_V_PVS_THD_MOVE_1 + C_V_PVS_MOVE_HYS_1, Channel 2 must have exceed V_PVS_MMV_SAVE_2 + C_V_PVS_MOVE_HYS_2, too (LV_PVS_2_NOT_MOVE_UP=0). Otherwize a fault would be detected (LV_PVS_2_NOT_MOVE_UP = 1).

The comparison is only executed or executed again if a referenz value was saved before (LV_SAVE_DET_INTER = 1) and the activation condition has been fullfiled (LV_PVS_MOVE_ACT_SUB = 1).

Formula section:

At Initialization LV_PVS_1_MOVE_DET$_{(K-1)}$ = 0

IF LV_PVS_1_MOVE_DET$_{(K-1)}$ == 0
THEN
  IF LV_PVS_MOVE_ACT == 1
  THEN LV_PVS_MOVE_ACT_SUB = 1
  ENDIF
  IF LV_SAVE_DET$_{(K-1)}$ == 1
  THEN LV_SAVE_DET_SUB = 1
  ENDIF
ELSE LV_PVS_MOVE_ACT_SUB = 0
  LV_SAVE_DET_SUB = 0
ENDIF

IF	LV_SAVE_DET_SUB == 1	**AND**
	LV_PVS_MOVE_ACT_SUB == 1	**AND**
	(V_PVS_MMV_1 > C_V_PVS_THD_MOVE_1 + C_V_PVS_MOVE_HYS_1)	
THEN	LV_PVS_1_MOVE_DET = 1	
ELSE	LV_PVS_1_MOVE_DET = 0	
IF	LV_PVS_1_MOVE_DET = 1	**AND**
	V_PVS_MMV_2 < V_PVS_MMV_SAVE_2 + C_V_PVS_MOVE_HYS_2	
THEN	LV_PVS_2_NOT_MOVE_UP = 1	
ELSE	LV_PVS_2_NOT_MOVE_UP = 0	
ENDIF		

A.45.4.2.4 Channel comparison falling signal

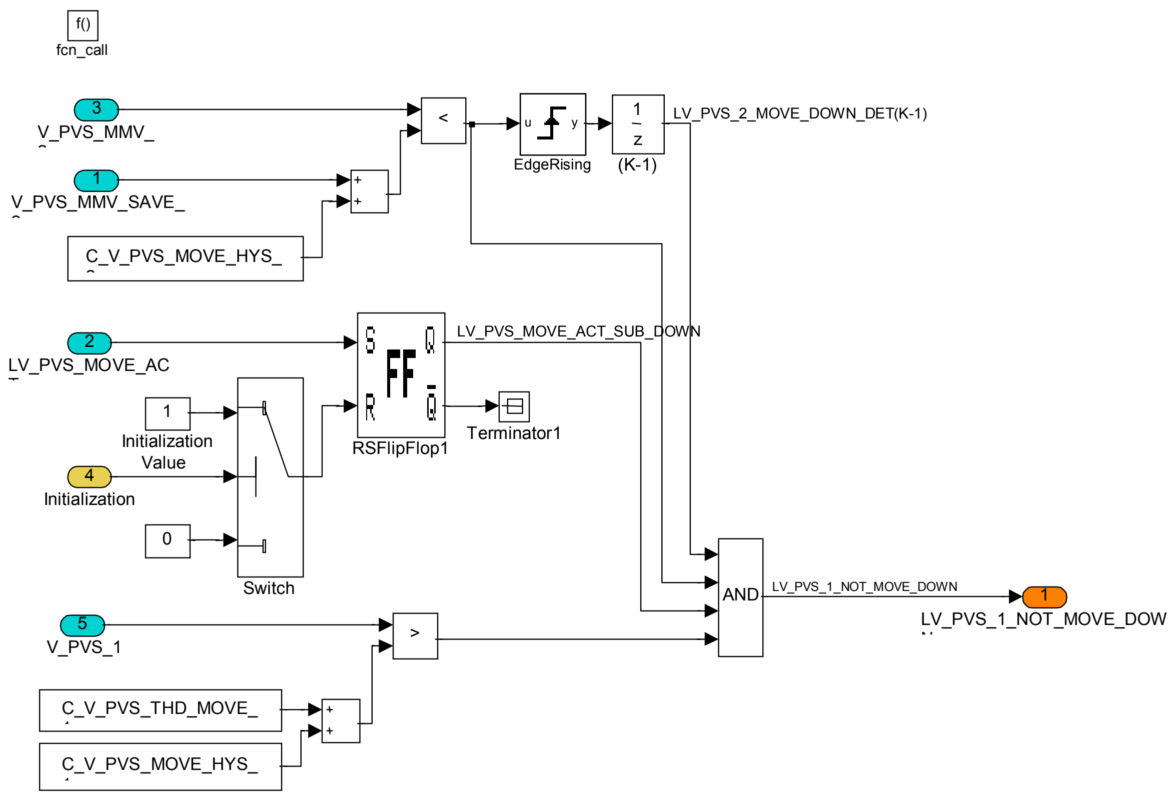


Figure 32: Channel comparison falling signal

Description:

If the filtered channel 2 signal falls below the previously saved value V_PVS_MMV_SAVE_2 (Channel comparison rising signal) + C_V_PVS_MOVE_HYS_2 again, channel 1 must be underneeth C_V_PVS_THD_MOVE_1 + C_V_PVS_MOVE_HYS_1, too. Otherwize a fault would be detected (LV_PVS_1_NOT_MOVE_DOWN$_{(K)}$ = 1).

The comparison is only executed if the activation condition has been fullfiled once (LV_PVS_MOVE_ACT).

Formula section:

At Initialization LV_PVS_MOVE_ACT_SUB_DOWN = 0

```
IF          Initialization
THEN        LV_PVS_MOVE_ACT_SUB_DOWN = 0
ELSE
    IF      LV_PVS_MOVE_ACT == 1
    THEN    LV_PVS_MOVE_ACT_SUB_DOWN = 1
    ENDIF
ENDIF
```

IF V_PVS_MMV_2$_{(K-1)}$ > V_PVS_MMV_SAVE_2 + C_V_PVS_MOVE_HYS_2 **AND**
V_PVS_MMV_2$_{(K)}$ < V_PVS_MMV_SAVE_2 + C_V_PVS_MOVE_HYS_2
THEN LV_PVS_2_MOVE_DOWN_DET = 1
ELSE LV_PVS_2_MOVE_DOWN_DET = 0
ENDIF

IF LV_PVS_2_MOVE_DOWN_DET$_{(K-1)}$ == 1 **AND**
V_PVS_MMV_2$_{(K)}$ < V_PVS_MMV_SAVE_2 + C_V_PVS_MOVE_HYS_2 **AND**
LV_PVS_MOVE_ACT_SUB_DOWN == 1 **AND**
V_PVS_1 > C_V_PVS_THD_MOVE_1 + C_V_PVS_MOVE_HYS_1
THEN LV_PVS_1_NOT_MOVE_DOWN = 1
ELSE LV_PVS_1_NOT_MOVE_DOWN = 0
ENDIF

A.45.4.2.5 Move Error Detection

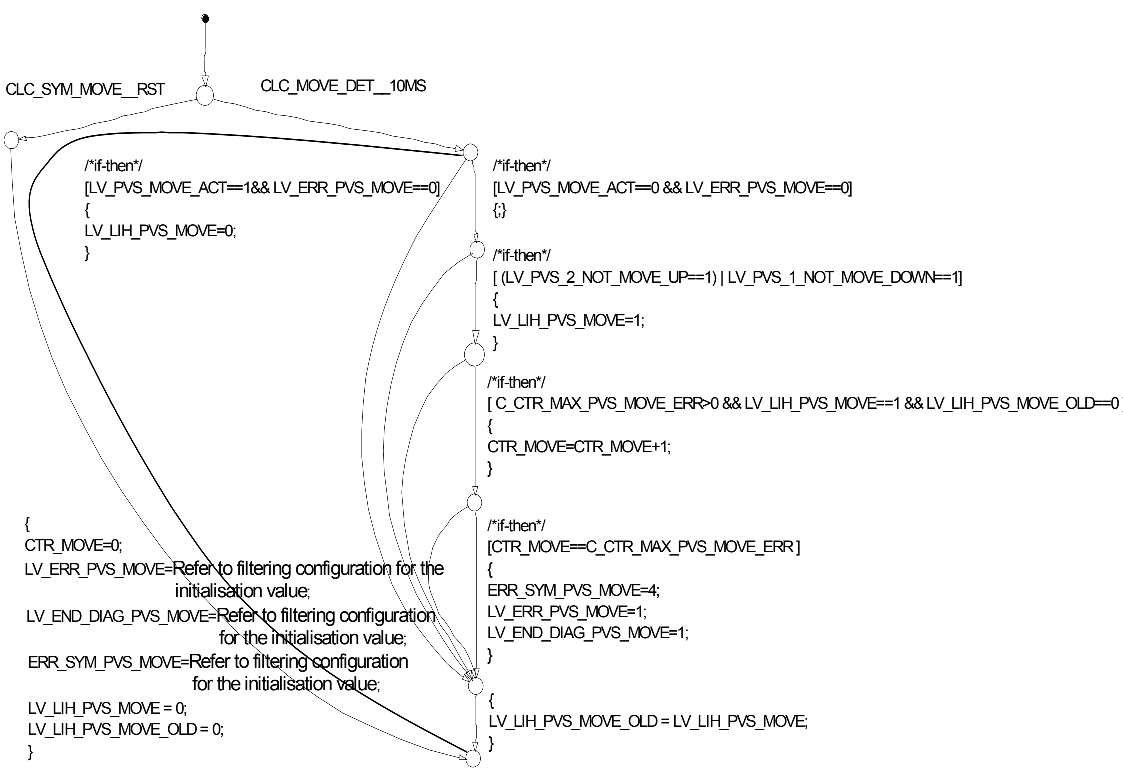


Figure 33: Move Error Detection

Description:

At Initialization following settings are done (on fcn-call CLC_SYM_MOVE__RST):
- CTR_MOVE = 0
- LV_ERR_PVS_MOVE = 0
- LV_END_DIAG_PVS_MOVE = 0
- ERR_SYM_PVS_MOVE = 0H (NO_SYM)
- LV_LIH_PVS_MOVE = 0
- LV_LIH_PVS_MOVE$_{(K-1)}$ = 0

During regular operation (on fcn-call CLC_MOVE_DET__10MS), calculation is done as shown below. If the symptom is detected the first time, LV_LIH_PVS_MOVE will be set and a counter is incremented by 1.
The bit is reset if LV_PVS_MOVE_ACT = 1 and LV_ERR_PVS_MOVE is not already debounced. After setting LV_LIH_PVS_MOVE several times,
- LV_ERR_PVS_MOVE =1
- ERR_SYM_PVS_MOVE = 4H (MOVE_MAX)
- LV_END_DIAG_PVS_MOVE = 1 are set,
- The flag LV_LIH_PVS_MOVE will be kept set.

The debouncing of LV_ERR_PVS_MOVE can be disabled setting C_CTR_MAX_PVS_MOVE_ERR = 0.

A.45.5 Plausibility Check Brake /PVS

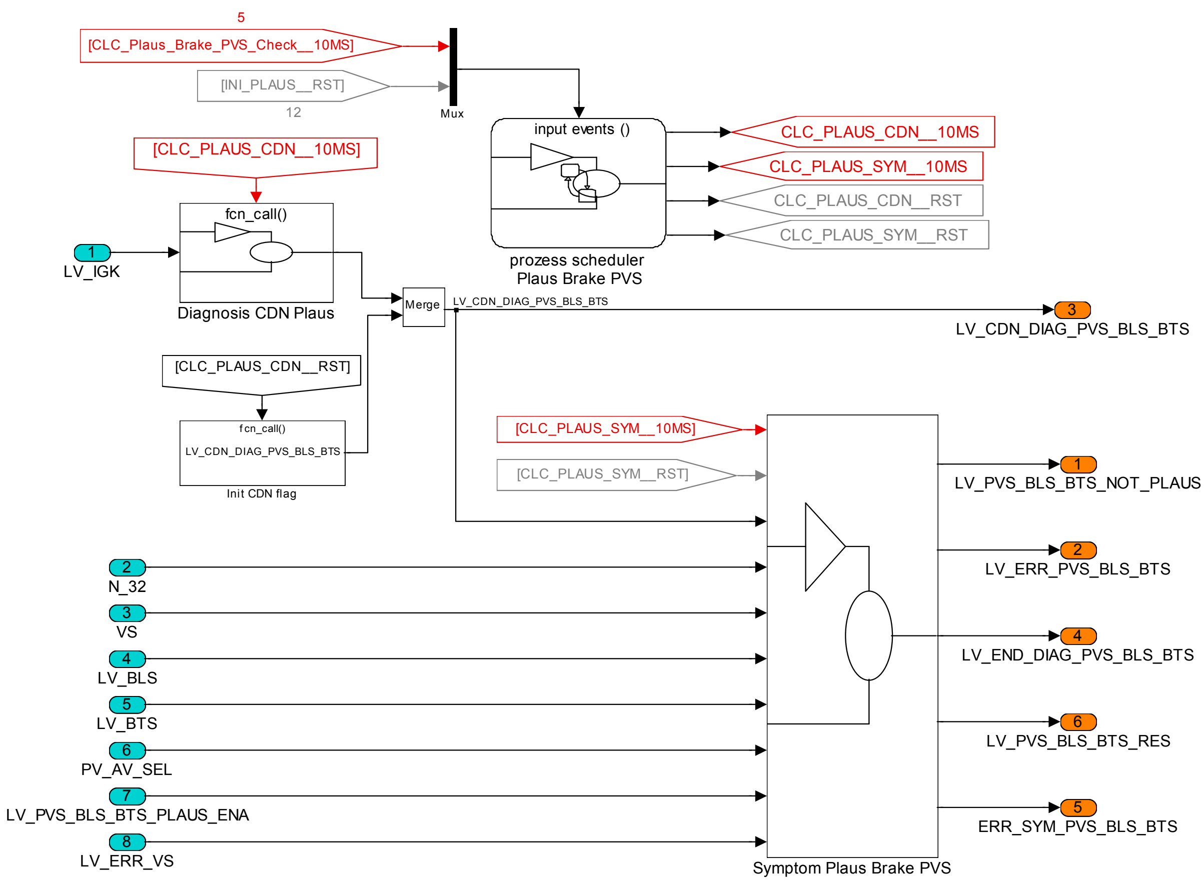


Figure 34: Plausibility Check Brake / PVS

Description:

The intend of this diagnosis function is to detect an undesired gas pedal activation against the drivers demand e.g. due to a mechanically clamping pedal.

If this state is detected, the bit LV_PVS_BLS_BTS_NOT_PLAUS will be set to decrease engine torque. After setting the bit C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR times, the bit LV_ERR_PVS_BLS_BTS will be set.

The debouncing of the bit LV_ERR_PVS_BLS_BTS can be disabled setting C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR = 0.

This function can be disabled setting LC_PVS_BLS_BTS_PLAUS_ENA = 0.

Calculation order:

1. Diagnosis CDN Plaus — on fcn-call CLC_PLAUS_CDN__10MS
2. Symptom Plaus Brake / PVS — on fcn-call CLC_PLAUS_SYM__10MS

Initialization order:

1. Diagnosis CDN Plaus — on fcn-call CLC_PLAUS_CDN__RST
2. Symptom Plaus Brake / PVS — on fcn-call CLC_PLAUS_SYM__RST

A.45.5.1 Diagnosis CDN Plaus

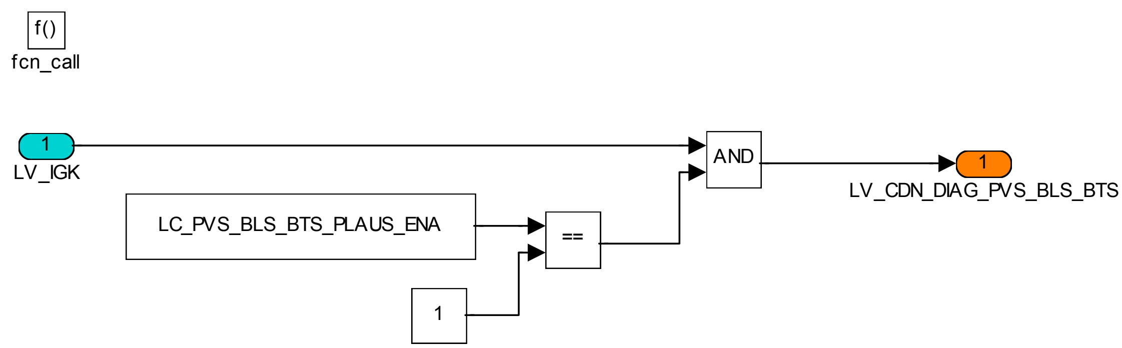


Figure 35: Diagnosis CDN Plaus

Description:

The Plausibility Check is **only** executed if LV_IGK == 1 and the enable bit LC_PVS_BLS_BTS_PLAUS_ENA ==1.

A.45.5.2 Symptom Plaus Brake / PVS

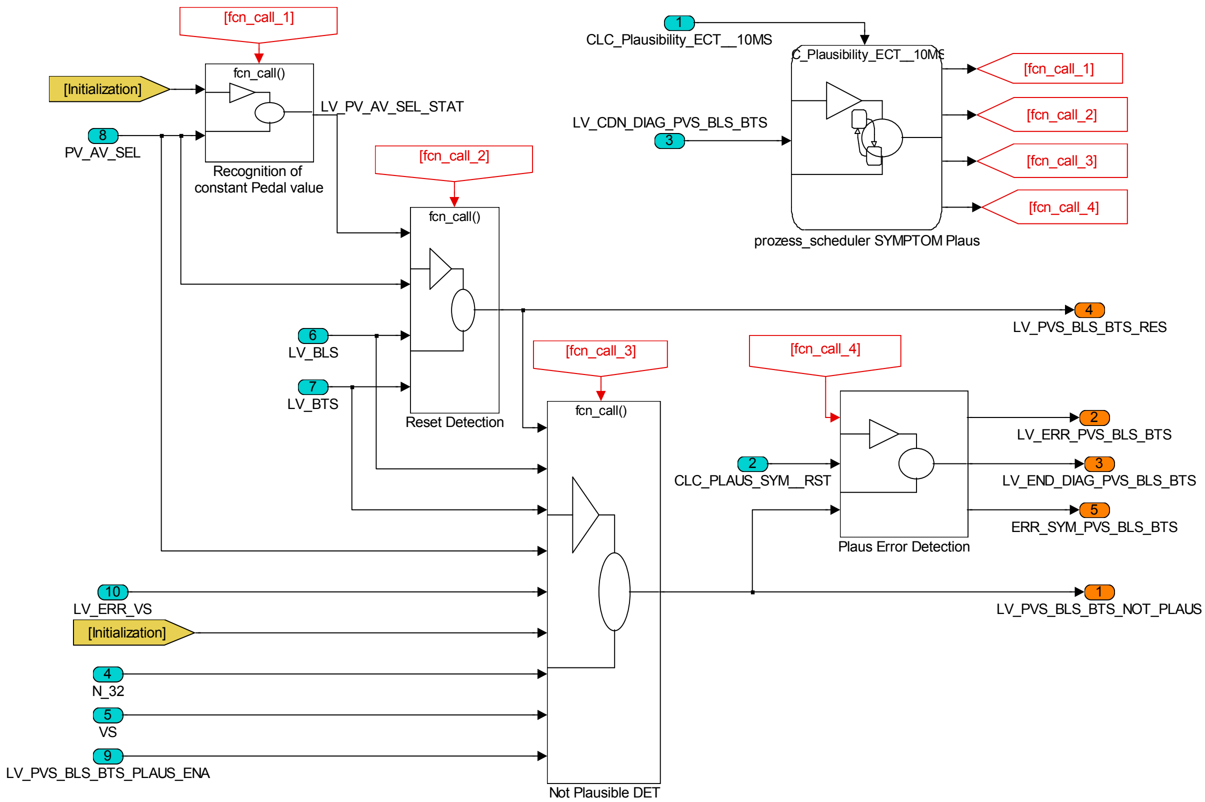


Figure 36: Symptom Plaus Brake / PVS

Description:

Calculation order:

1. Recognition of constant pedal value on fcn_call_1
2. Reset detection on fcn_call_2
3. Not Plausible DET on fcn_call_3
4. Plaus Error Detection on fcn_call_4

A.45.5.2.1 Recognition of constant pedal value

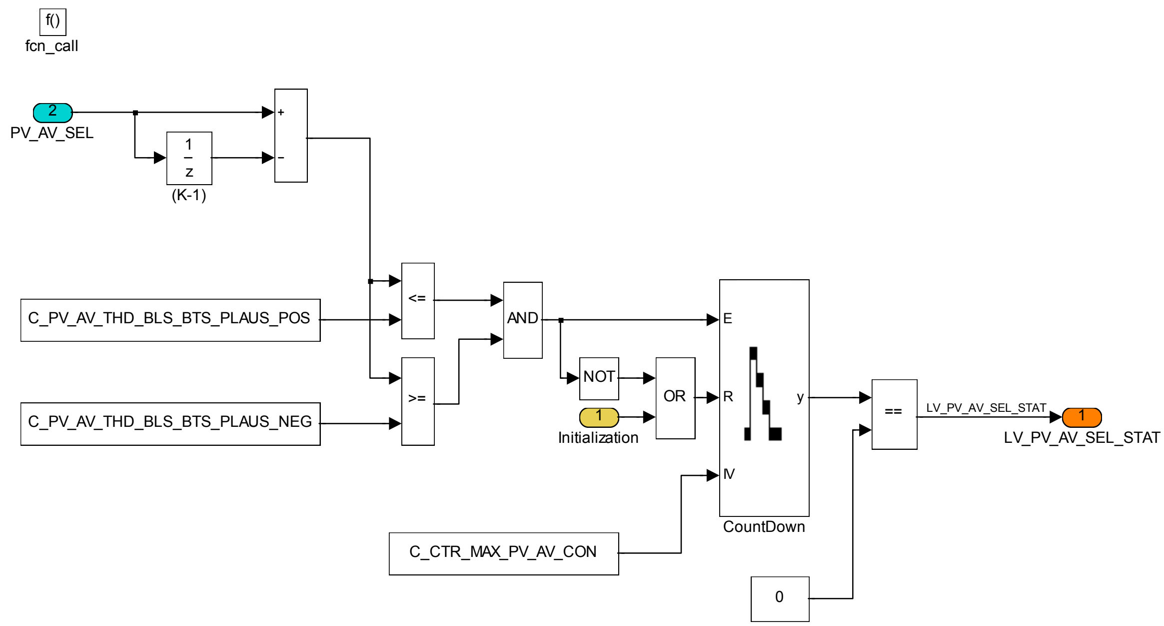


Figure 37: Recognition of constant pedal value

Description:

The gas pedal is recognized as constant LV_PV_AV_SEL_STAT = 1, if the pedal value does not differ more than a permitted value up- and downwards for the time C_CTR_MAX_PV_AV_CON.

Formula section:

IF NOT{(PV_AV_SEL$_{(K)}$ – PV_AV_SEL$_{(K-1)}$ <= C_PV_AV_THD_BLS_BTS_PLAUS_POS) **and**
(PV_AV_SEL$_{(K)}$ – PV_AV_SEL$_{(K-1)}$ >= C_PV_AV_THD_BLS_BTS_PLAUS_NEG)**}** **OR**
Initialization
THEN CTR_PVS_STAT$_{(K)}$ = C_CTR_MAX_PV_AV_CON
ELSE
IF CTR_PVS_STAT$_{(K-1)}$ > 0
THEN CTR_PVS_STAT$_{(K)}$ = CTR_PVS_STAT$_{(K-1)}$ – [update rate]
ENDIF
ENDIF
IF CTR_PVS_STAT$_{(K)}$ == 0
THEN LV_PV_AV_SEL_STAT = 1
ELSE LV_PV_AV_SEL_STAT = 0
ENDIF

A.45.5.2.2 Reset detection

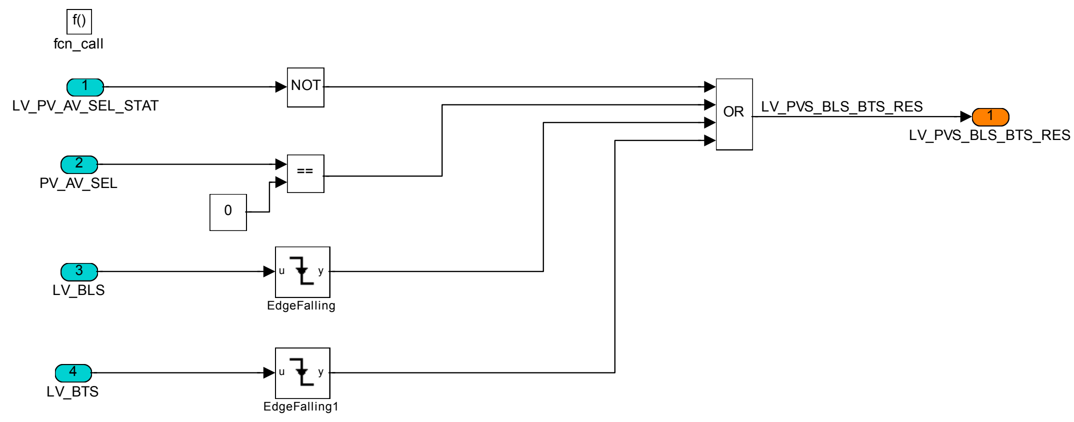


Figure 38: Reset detection

Description:

A recognized implausibility of brake and gas pedal is being reset (LV_PVS_BLS_BTS_RES = 1), if

- the gas pedal is recognized as not constant (LV_PV_AV_SEL_STAT = 0), or
- the actual pedal value PV_AV_SEL = 0 (driver intend is idle), or
- the brake recognition changes to inactive LV_BLS = 1 → 0 or
- the brake recognition changes to inactive LV_BTS = 1 → 0

Formula section:

IF	LV_PV_AV_SEL_STAT == 0	**OR**
	PV_AV_SEL == 0	**OR**
	{LV_BLS$_{(K-1)}$ == 1 **and** LV_BLS$_{(K)}$ == 0**}**	**OR**
	{LV_BTS$_{(K-1)}$ == 1 **and** LV_BTS$_{(K)}$ == 0**}**	
THEN	LV_PVS_BLS_BTS_RES = 1	
ELSE	LV_PVS_BLS_BTS_RES = 0	
ENDIF		

A.45.5.2.3 Not Plausible DET

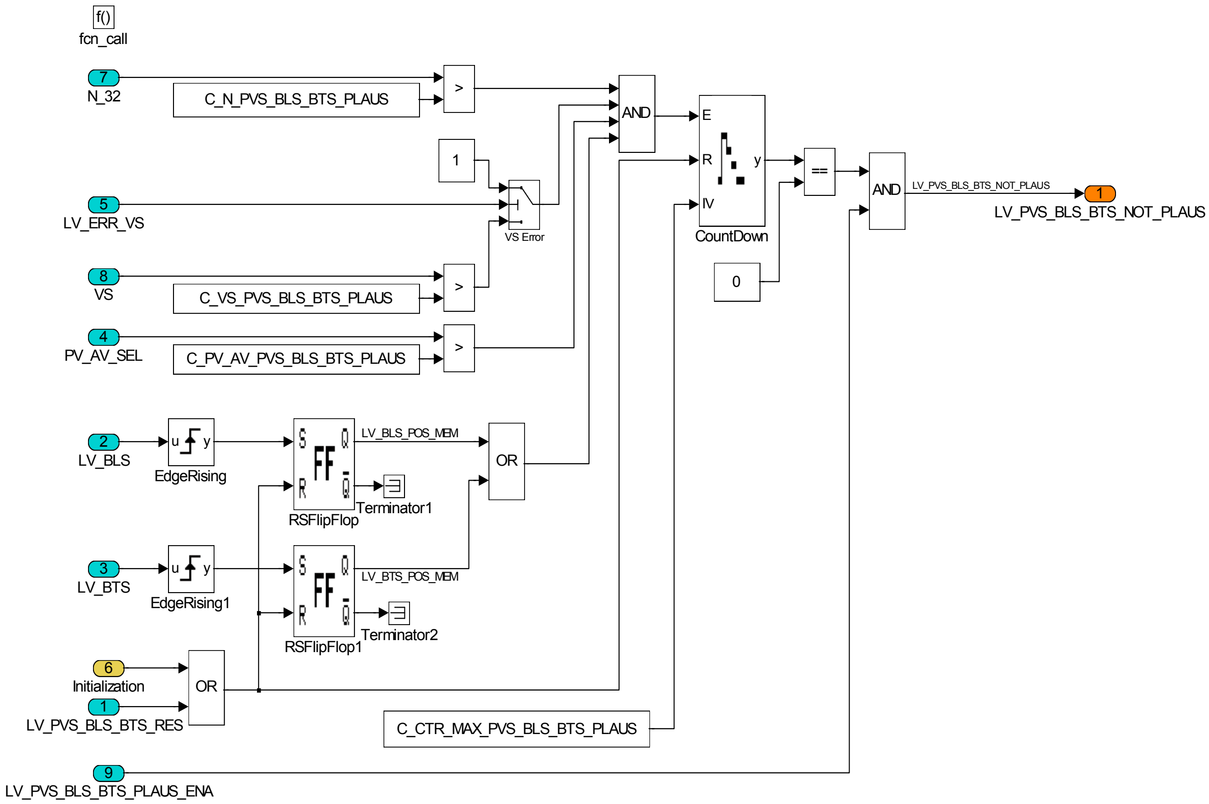


Figure 39: Not Plausible Detection

Description:

An undesired gas pedal activation against the driver demand (e.g. clamping pedal) is detected by using the break information LV_BLS / LV_BTS.

If an activation of the brake is detected **after** the gas pedal has already been activated and

N_32 is above C_N_PVS_BLS_BTS_PLAUS and

VS is above C_VS_PVS_BLS_BTS_PLAUS (not if VS signal is faulty LV_ERR_VS==1)and

PV_AV_SEL is above C_PV_AV_PVS_BLS_BTS_PLAUS

a counter will be decremented. If the counter reaches 0 after the time C_CTR_MAX_PVS_BLS_BTS_PLAUS and LV_PVS_BLS_BTS_PLAUS_ENA = 1, an implausible state of brake and gas pedal will be detected (LV_PVS_BLS_BTS_NOT_PLAUS = 1).

Formula section:

IF	LV_PVS_BLS_BTS_RES ==1 **OR** Initialization	
THEN	CTR_NOT_PLAUS$_{(K)}$ = C_CTR_MAX_PVS_BLS_BTS_PLAUS LV_BLS_POS_MEM = 0 LV_BTS_POS_MEM = 0	
ENDIF		
IF	**(**LV_BLS$_{(K-1)}$ == 0 **and** LV_BLS$_{(K)}$ == 1**)**	
THEN	LV_BLS_POS_MEM = 1	
IF	**(**LV_BTS$_{(K-1)}$ == 0 **and** LV_BTS$_{(K)}$ == 1**)**	
THEN	LV_BTS_POS_MEM = 1	
IF	**(**LV_BLS_POS_MEM ==1 **or** LV_BTS_POS_MEM == 1**)**	**AND**
	N_32 > C_N_PVS_BLS_BTS_PLAUS	**AND**
	{ [VS > C_VS_PVS_BLS_BTS_PLAUS and LV_ERR_VS==0**]** or LV_ERR_VS==1**] }**	**AND**
	PV_AV_SEL > C_PV_AV_PVS_BLS_BTS_PLAUS	**AND**
	CTR_NOT_PLAUS$_{(K)}$ > 0	
THEN	CTR_NOT_PLAUS$_{(K)}$ = CTR_NOT_PLAUS$_{(K-1)}$ – [update rate]	
ENDIF		
IF	CTR_NOT_PLAUS$_{(K)}$ == 0	**AND**
	LV_PVS_BLS_BTS_PLAUS_ENA ==1	
THEN	LV_PVS_BLS_BTS_NOT_PLAUS = 1	
ELSE	LV_PVS_BLS_BTS_NOT_PLAUS = 0	
ENDIF		

A.45.5.2.4 Plaus Error Detection

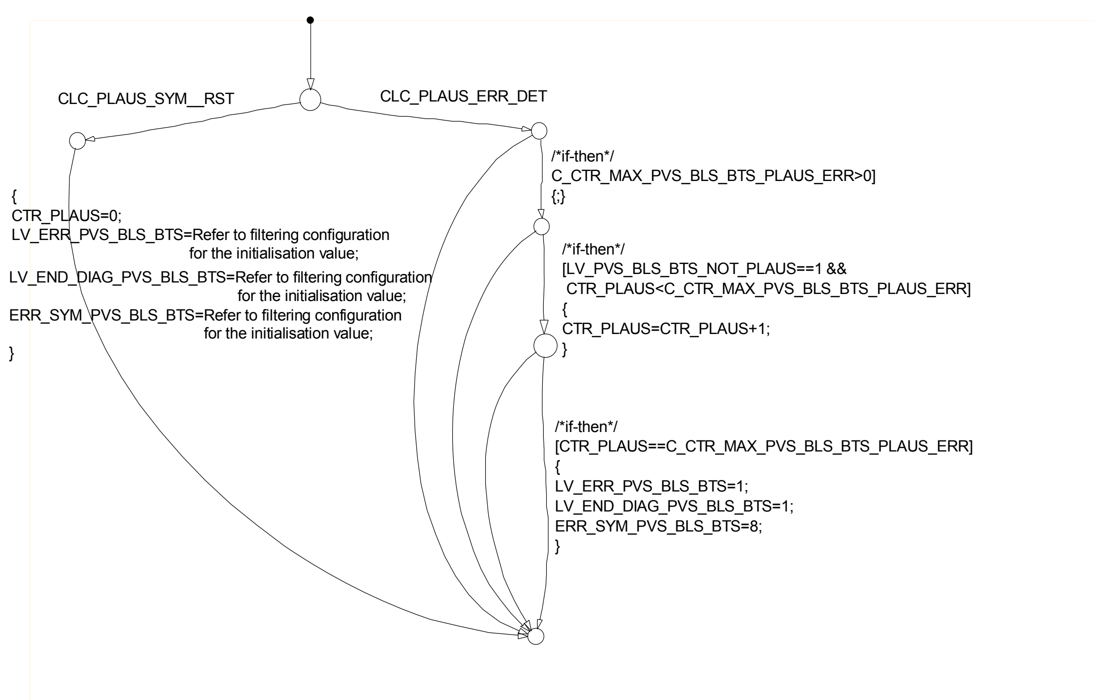


Figure 40: Plaus Error Detection

Description:

At Initialization following settings are done:

- CTR_PLAUS = C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR
- LV_ERR_PVS_BLS_BTS = 0
- LV_END_DIAG_PVS_BLS_BTS = 0
- ERR_SYM_PVS_BLS_BTS = 0H (NO_SYM).

During normal operation calculation is done as shown below. Each detection the counter is increment by one. After C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR times,

- LV_ERR_PVS_BLS_BTS = 1
- ERR_SYM_PVS_BLS_BTS = 8H (PLAUS_MAX)
- LV_END_DIAG_PVS_BLS_BTS = 1.

The debouncing of LV_ERR_PVS_BLS_BTS can be disabled setting C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR = 0.

A.45.6 Common Error Flags

Description:

At Initialization following settings are done:

- LV_ACT_PVS_LIH = 0
- LV_ERR_PVS = 0
- LV_ERR_PVS_PRED = 0

During normal operation calculation is done as shown below.

While a ratio error or out of range error is predicted (LV_ERR_PVS_ORNG_PRED or LV_ERR_PVS_RATIO_PRED), the common flag LV_ERR_PVS_PRED will be set.

If an error is debounced by ratio check or out of range check or supply voltage diagnosis (LV_ERR_PVS_RATIO or LV_ERR_PVS_x or LV_ERR_VCC_PVS_x), the flag LV_ACT_PVS_LIH for PVS limp home coordination will be set.

If LV_ACT_PVS_LIH is set or an error is debounced by the Move Check (LV_ERR_PVS_MOVE), the common error flag LV_ERR_PVS will be set.

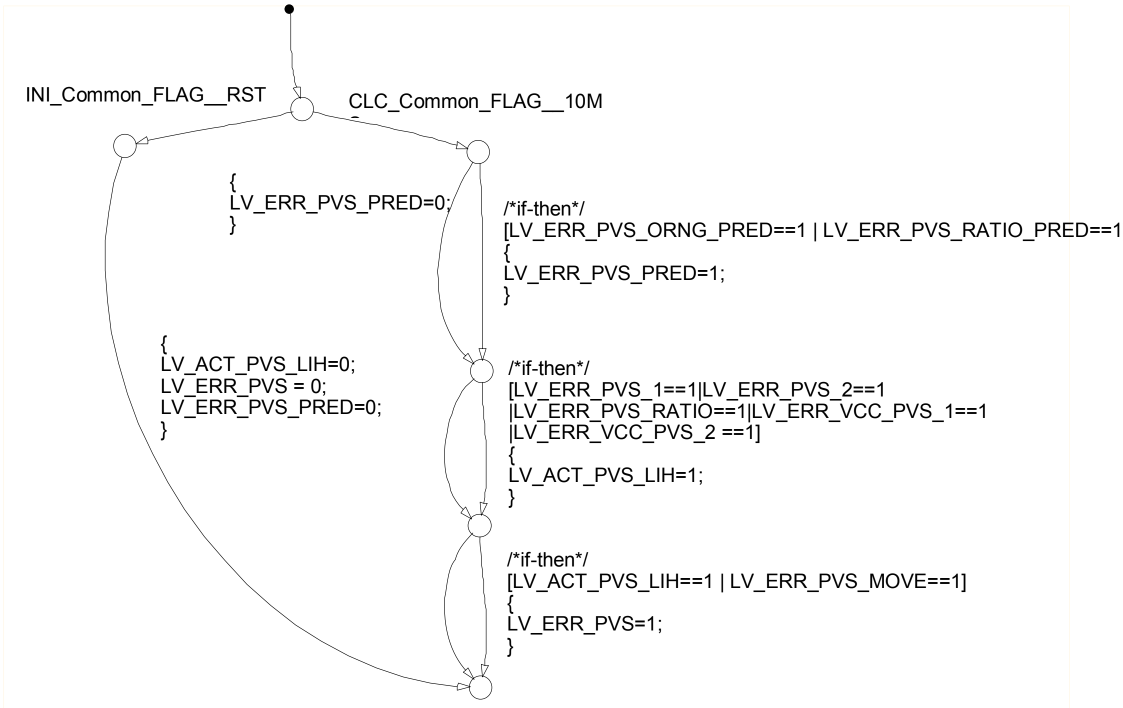


Figure 41: Common Error Flags

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_PVS_POTI_EQUIP	1	0...1H	0...1	1	-
Flag to indicate if it is a potentiometer system					
C_CTR_MAX_PVS_IS_RNG	1	0...FFH	0...2.55	0.01	s
PV_AV_X counter maximum value					
C_V_PVS_MAX_DIAG_1	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis threshold SRCH channel 1					
C_V_PVS_MAX_DIAG_2	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis threshold SRCH channel 2					
C_V_PVS_MIN_DIAG_1	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis threshold SRCL channel 1					
C_V_PVS_MIN_DIAG_2	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis threshold SRCL channel 2					
C_V_PVS_MIN_DIAG_MIN_2	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis activation threshold SRCL , Ratio and HRCL channel 2					
C_V_PVS_MIN_DIAG_MIN_1	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis activation threshold SRCL, Ratio and HRCL channel 1					
C_PVS_RATIO_FAC	1	0...FFFFH	0...7.999878	1.221E-4	-
Ratio of the two PVS-channels					
C_V_PVS_RATIO_OFS	1	FE00...1FFH	-2.5...2.495117	0.004883	V
Offset of the two PVS-channels					
IP_V_PVS_RATIO_HYS	6	0...3FFH	0...4.995117	0.004883	V
LDP_V_PVS_MAX_IP_V_PVS_RATIO	6	0...FFFFH	0...319.995117	0.004883	V
Maximum PVS ratio hystheresis					
C_CTR_MAX_PVS_H_R_1	1	0...FFH	0...2.55	0.01	s
PVS_H_R_1 counter maximum value					
C_CTR_MAX_PVS_H_R_2	1	0...FFH	0...2.55	0.01	s
PVS_H_R_2 counter maximum value					
LC_PVS_MOVE_CHK_ENA	1	0...1H	0...1	1	-
Flag to disable the Move Check					
C_CTR_MAX_PVS_MOVE	1	0...FFH	0...2.55	0.01	s
PVS_MOVE counter maximum value					
C_CRLC_PVS_2_MOVE	1	0...FFH	0...0.9960938	0.003906	-
Correlation constant for the filtration of V_PVS_2					
C_CRLC_PVS_1_MOVE	1	0...FFH	0...0.9960938	0.003906	-
Correlation constant for the filtration of V_PVS_1					
C_V_PVS_THD_MOVE_1	1	0...3FFH	0...4.995117	0.004883	V
Diagnosis threshold MOVE-CHECK channel 1					
C_V_PVS_MOVE_HYS_1	1	0...3FFH	0...4.995117	0.004883	V
Band width for move recognition of channel 1					
C_V_PVS_MOVE_HYS_2	1	0...3FFH	0...4.995117	0.004883	V
Band width for move recognition of channel 2					
C_CTR_MAX_PVS_MOVE_ERR	1	0...FFH	0...255	1	-
maximum number of errors for debouncing					
C_CTR_MAX_PVS_BLS_BTS_PLAUS_ERR	1	0...FFH	0...255	1	-
PVS_BLS_BTS_PLAUS_ERR counter maximum value					
C_N_PVS_BLS_BTS_PLAUS	1	0...FFH	0...8.16E3	32	1/min
Activation threshold for N					
C_VS_PVS_BLS_BTS_PLAUS	1	0...FFH	0...255	1	km/h
Activation threshold for VS					
C_PV_AV_PVS_BLS_BTS_PLAUS	1	0...3FFH	0...99.902344	0.09766	%
Activation threshold for PV_AV					
LC_PVS_BLS_BTS_PLAUS_ENA	1	0...1H	0...1	1	-
Enable bit for plausibility check					
C_CTR_MAX_PVS_BLS_BTS_PLAUS	1	0...FFH	0...2.55	0.01	s
PVS_BLS_BTS_PLAUS counter maximum value					

general specification

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_MAX_PV_AV_CON	1	0...FFH	0...2.55	0.01	s
PV_AV_CON counter maximum value					
C_PV_AV_THD_BLS_BTS_PLAUS_POS	1	FC00...3FFH	-100...99.902344	0.09766	%
Maximum positive pedal value for constant pedal value recognition					
C_PV_AV_THD_BLS_BTS_PLAUS_NEG	1	FC00...3FFH	-100...99.902344	0.09766	%
Maximum negative pedal value for constant pedal value recognition					

A.46 Diagnosis of Pedal Value Sensor (Application Incidences)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_PVS_H_R_1_DET	V/O	0...1H	0...1	1	-
PVS 1 high resistance detected in last driving cycle					
LV_PVS_H_R_2_DET	V/O	0...1H	0...1	1	-
PVS 2 high resistance detected in last driving cycle					
LV_PVS_H_R_1_SAVE	V/O/S	0...1H	0...1	1	-
PVS 1 high resistance information stored in EEPROM					
LV_PVS_H_R_2_SAVE	V/O/S	0...1H	0...1	1	-
PVS 2 high resistance information stored in EEPROM					
LV_PVS_BLS_BTS_PLAUS_ENA	O/V	0...1H	0...1	1	-
Plausibility check PVS/BLS/BTS enable					

Input data:

LV_PVS_H_R_1_RLS	LV_PVS_H_R_2_RLS	LV_IGK	

General information:

The application incidences describe the anti-bounce mechanism and error treatment of the module ‘Diagnosis of pedal value sensor’.

Since it is a two-channel system, the letter ‘x’ is for channel 1 or channel 2, in order to simplify the description.

All conditions flags and error symptoms mentioned in “Error treatment” are only used to describe the configuration of the anti bounce mechanism. They have no functional use here and are therefore not enumerate as inputs.

Application conditions:

Activation: LV_IGK = 1

Deactivation: otherwise

Initialisation: **IF** LV_PVS_H_R_x_SAVE == 1

{High resistance information was stored in non volatile memory}

THEN LV_PVS_H_R_x_DET = 1

{High resistance of channel X was detected in last driving cycle}

LV_PVS_H_R_x_SAVE = 0

{Error information deleted in non volatile memory}

ELSE LV_PVS_H_R_x_DET = 0

ENDIF

LV_PVS_BLS_BTS_PLAUS_ENA = 0

Recurrence: *every 10 ms*

General Description:

The Application Incidences are executed at function call CLC_PVS_DIAG_APPL_INC__10MS and initialized at CLC_PVS_DIAG_APPL_INC__RST.

A.46.1 Initialization conditions

Description:

The application condition for the initialization of e.g. the error bits, the anti-bounce counters and other variables of the modules:

- Diagnosis of Pedal Value Sensor
- Diagnosis of Pedal Value Sensor (Application Incidences) are commonly defined here.

The error bits, anti-bounce-counters and other variables or bits are initialized at reset, at transition from LV_IGK = 0 to 1 and at clearing of the failure memory.

A.46.2 Error treatment

High resistance check:

IF LV_PVS_H_R_x_RLS == 1

{High resistance of channel x detected in current driving cycle}

THEN LV_PVS_H_R_x_SAVE = 1

{High resistance information is stored in the non volatile memory at the end of the driving cycle to be used as information in next driving cycle }

ENDIF

A.46.3 General application incidences

Enable of PVS/BLS/BTS plausibility check:

An ESP-intervention can activate the Brakelight- or Braketest-switch. In this case the check must be disabled by setting the LV_PVS_BLS_BTS_PLAUS_ENA = 0 during ESP or ASR intervention.

IF LC_PVS_BLS_ESP_CHK_ENA = 1 (ESP is able to disable the BLS)

THEN LV_PVS_BLS_BTS_PLAUS_ENA = 0

ELSE LV_PVS_BLS_BTS_PLAUS_ENA = 1

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
PVS_BLS_BTS		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Plausibility fault	SYM_3	

List of Environmental Data to store in Failure Memory: PV_AV_SEL_ENVD
VS
LV_BLS
LV_BTS

Diagnostic	Symptom Description	Symptom	Filter type
ORNG check high PVS_H_1	SCB	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_PVS_1_ENVD
LV_ERR_VCC_PVS_1
LV_PVS_H_R_1
LV_ERR_PVS_RATIO

Diagnostic	Symptom Description	Symptom	Filter type
ORNG check high PVS_H_2	SCB	SYM_0	*MEM*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_PVS_1_ENVD
V_PVS_2_ENVD
LV_ERR_VCC_PVS_1
LV_ERR_VCC_PVS_2

Diagnostic	Symptom Description	Symptom	Filter type
ORNG check low PVS_L_1		SYM_0	*MEM*
	SCG + OL	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: LV_PVS_1_ENVD
LV_ERR_VCC_PVS_1
LV_PVS_H_R_1
LV_ERR_PVS_RATIO

Diagnostic	Symptom Description	Symptom	Filter type
ORNG check low PVS_L_2		SYM_0	*MEM*
	SCG + OL	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_PVS_2_ENVD
LV_ERR_VCC_PVS_2
LV_PVS_H_R_2
LV_ERR_PVS_RATIO

Diagnostic	Symptom Description	Symptom	Filter type
Move check PVS_MOVE		SYM_0	*MEM*
		SYM_1	
	MOVE_MAX	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_PVS_1_ENVD
V_PVS_2_ENVD
LV_ERR_VCC_PVS_1
LV_ERR_VCC_PVS_2

Diagnostic	Symptom Description	Symptom	Filter type
Ratio check PVS_RATIO		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Ratio deviation	SYM_3	

List of Environmental Data to store in Failure Memory: V_PVS_1_ENVD
V_PVS_2_ENVD
LV_ERR_VCC_PVS_1
LV_ERR_VCC_PVS_2

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_PVS_RATIO	1	0...FFH	0...255	1	-
Anti-bounce counter increment in case of a ratio error					
C_ABC_MAX_PVS_RATIO	1	1...FFH	1...255	1	-
Anti-bounce counter maximum					
C_ABC_INC_PVS_ORNG_1	1	0...FFH	0...255	1	-
Anti-bounce counter increment in case of a out of range error channel 1					
C_ABC_MAX_PVS_ORNG_1	1	1...FFH	1...255	1	-
Anti-bounce counter maximum					
C_ABC_INC_PVS_ORNG_2	1	0...FFH	0...255	1	-
Anti-bounce counter increment in case of a out of range error channel 2					
C_ABC_MAX_PVS_ORNG_2	1	1...FFH	1...255	1	-
Anti-bounce counter maximum					
LC_PVS_BLS_ESP_CHK_ENA	1	0...1H	0...1	1	-
ESP is able to disable the BLS if set to 1					

A.47 Ignition diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_IGC_SCP[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Ignition diagnosis condition for short circuit plus					
LV_CDN_DIAG_IGC_SCG[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Ignition diagnosis condition for short circuit to ground					
LV_CDN_DIAG_IGC_OL[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Ignition diagnosis condition for open load					
ERR_SYM_IGC_SCP[NC_CYL_NR]	V	0H 1H	NO_SYM SCP	1	-
Symptom of the ignition coil failure primary short circuit plus					
ERR_SYM_IGC_SCG[NC_CYL_NR]	V	0H 2H	NO_SYM SCG	1	-
Symptom of the ignition coil failure primary short circuit ground					
ERR_SYM_IGC_OL[NC_CYL_NR]	V	0H 4H	NO_SYM OL	1	-
Symptom of the ignition coil failure primary open load					
LV_ERR_IGC_SCP[NC_CYL_NR]	V/S	0H 1H	Passive Active	1	-
Ignition coil failure short circuit plus					
LV_ERR_IGC_SCG[NC_CYL_NR]	V/S	0H 1H	Passive Active	1	-
ignition coil failure primary short circuit ground					
LV_ERR_IGC_OL[NC_CYL_NR]	V/S	0H 1H	Passive Active	1	-
Ignition coil failure primary open load					
LV_END_DIAG_IGC_SCP[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Result of ignition diagnosis for SCP					
LV_END_DIAG_IGC_SCG[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Result of ignition diagnosis for SCG					
LV_END_DIAG_IGC_OL[NC_CYL_NR]	V	0H 1H	Passive Active	1	-
Result of ignition diagnosis OL					
IGC_DIAG_MIS	O/V	0...FFH	0...255	1	-
Non filtered error information for misfire detection					
LV_INH_IGC[NC_CYL_NR]	O/V	0...1H	0...1	1	-
Inhibition for ignition system of cylinder x					
LV_ERR_IGC	O/V	0...1H	0...1	1	-
OR of every failure every 100 ms					

Input data:

LV_OL_IGC[NC_CYL_NR]	LV_ST	IGN_MPL_NR	LV_CDN_VB_OBD1
LV_SCG_IGC[NC_CYL_NR]	LV_ES	LV_ERR_CAM	TCO
LV_SCP_IGC[NC_CYL_NR]	N_32	LV_ES	LV_IGK
LV_IGC_x_EXT_ADJ	IGC_x_EXT_ADJ		

FUNCTION DESCRIPTION:

General information:

The purpose of this diagnosis function is to detect all major failures, which can happen between ECU output and ignition coils.

This function is adaptable on every engine but the type of spark plugs and ignition coils must be taken into account.

The diagnosis is performed separate for each ignition coil. The feedback signal from the specific ignition coil (following the firing order) is evaluated by the microcontroller.

A.47.1 Short circuit to plus: SCP

Description:

In some Hard ware driver the ignition powerstage is not protected against short circuit to ground and they are not able to withstand an overload current for a long time so it is necessary to switch off the powerstage by software.

For ignition coil actuator tests purpose, the SCP diagnostic function has to be enabled, activated and monitored upon the tests requests

The same case and comments (see before) apply here also.

The Ignition output diagnosis function detects short circuit to battery and sets LV_IGC_SCP[x]. It cuts off the ignition driver after a delay time which is defined in the IGBT Protection function. In the ignition diagnosis function the error is debounced.

Application conditions:

1. **Normal Engine Running mode**

Initialisation: all output variables of the function have to be set to 0 at reset ECU *(except saved error flag)*

Recurrence: if NC_IGN_DIAG_TYP =ATM46 every segment

if NC_IGN_DIAG_TYP =ATIC29 every 100 ms

Activation: **If** LV_INH_IGC[x] = 0

And LV_ERR_CAM = 0

And N_32 > C_N_32_MIN_SCP_IGC_DIAG

And IGN_MPL_NR = 0

And if NC_IGN_DIAG_TYP =ATIC29
Failure information for current cylinder
from ATIC29 is valid

endif

Then LV_CDN_DIAG_IGC_SCP[x] = 1

Else LV_CDN_DIAG_IGC_SCP[x] = 0

Endif

2. **Ignition Actuator Tests mode**

Initialisation: all output variables of the function have to be set to 0 at reset ECU *(except saved error flag)*

Recurrence: **if** NC_IGN_DIAG_TYP =ATM46
On transition of IGC_x_EXT_ADJ from 1 to 0
for reading the SCP failure status

Activation: For x = 0 to NC_CYL_NR – 1

If LV_INH_IGC[x] = 0
And LV_ERR_CAM = 0

[And LV_ES = 1 // *Actuator tests activation*

And LV_IGK = 1

And LV_IGC_x_EXT_ADJ = 1]

Then LV_CDN_DIAG_IGC_SCP[x] = 1
Else LV_CDN_DIAG_IGC_SCP[x] = 0
Endif
EndFor

Formula section:

If LV_CDN_DIAG_IGC_SCP =1

Then **If** LV_SCP_IGC[x] = 1

Then ERR_SYM_IGC_SCP[x] = 1H

else ERR_SYM_IGC_SCP[x] = 0H

Else No action on the antibounce counter

Endif

Debounce:

Anti - bounce counter increment: C_ABC_INC_IGC_SCP

Maximum value of anti - bounce counter: C_ABC_MAX_IGC_SCP

If LV_CDN_DIAG_IGC_SCP[x] =1

Then **If** ERR_SYM_IGC_SCP[x] = 1H

Then the antibouce counter is started (see anti bounce counter)

Else The antibounce counter is decremented

Endif

Endif

Ignition cut off conditions:

If C_ABC_MAX_IGC_SCP delay is reached

then LV_INH_IGC[x] = 1

Endif

Ignition restart after cut off:

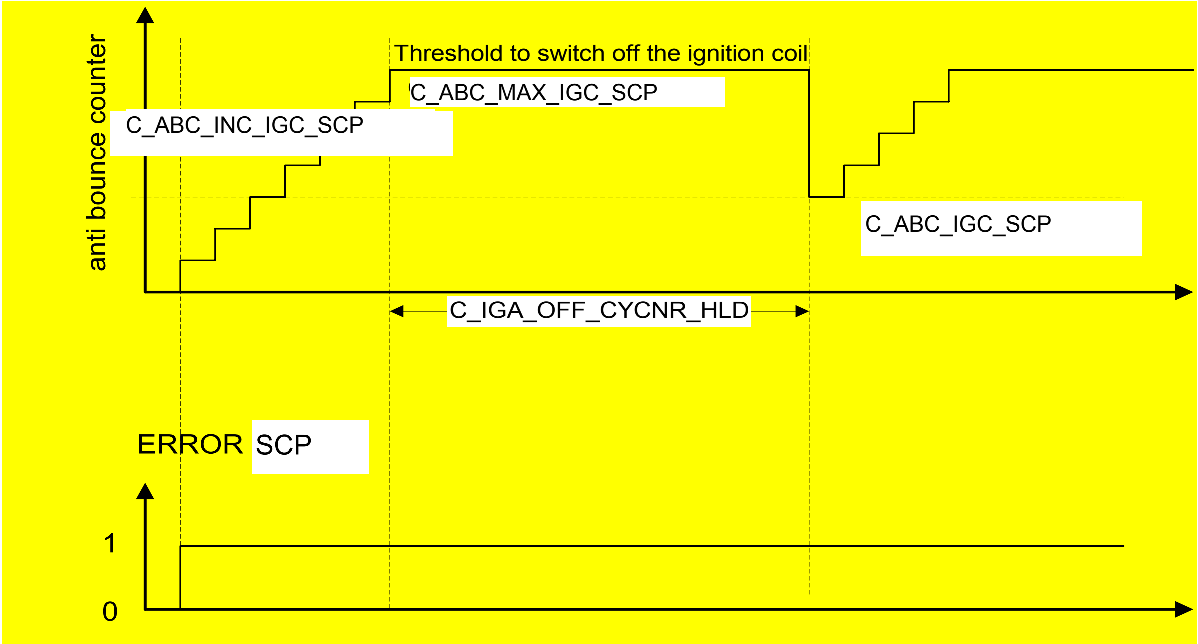
To regenerate the power stage the diagnosis will be switched off (LV_INH_IGC[x] = 1) for C_IGA_OFF_CYCNR_HLD recurrence number (see graphic, failure reaction in case of error).

After the refreshing cycle the debounce counter is set to C_ABC_IGC_SCP (see graphic below) and the ignition will be switched on to test the operating stage again. In case of error it will be debounced again.

This procedure is executed C_SUM_IGC_SCP times before the coil is absolutely switched off, until reset for ATM46 not for ATIC29

In next engine cycle the error is kept present. To take sure that the vehicle is able to restart the ignition will be switched on, till the diagnosis conditions are fulfilled and the diagnosis restarts.

Failure reaction in case of error



Anti bounce configuration data:

Diagnosis	Symptom	SYM	ABC type
Ignition diagnosis for SCP	SCP	*0*	STD
LV_ERR_IGC_SCP			

Application hint:

Some driver as ATM46 does not cut itself in case of SCP failure. For better Driver protection The counter threshold C_ABC_IGC_SCP mustn't exceed 4 cycles For the same reason the counter C_ABC_MAX_IGC_SCP must be set below 12 cycles.
The function can be disabled by the 'enable byte' C_ESB_IGN_CUT_DIAG.

A.47.2 OL / SCG diagnosis

Description

The diagnosis is based on the OL and SCG information from the ignition output diagnosis function.

Application conditions:

Initialisation: all outputs variables have to be set to "0" at reset

Recurrence: if NC_IGN_DIAG_TYP =ATM46 every segment

if NC_IGN_DIAG_TYP =ATIC29 every 100 ms

Activation: **If** LV_ES = 0

And LV_ST = 0

And N_32 > C_N_32_MIN_OL_SCG_IGC_DIAG

And TCO > C_TCO_MIN_IGC_DIAG

And LV_INH_IGC[x] = 0

And LV_ERR_CAM = 0

And No low voltage battery (LV_CDN_VB_OBD1 = 1)

And Counter Number of recurrence > NC_IGC_DLY

And IGN_MPL_NR = 0

And if NC_IGN_DIAG_TYP =ATIC29
Failure information for current cylinder
from ATIC29 is valid
endif

Then LV_CDN_DIAG_IGC_SCG[x] =1

LV_CDN_DIAG_IGC_OL[x] =1

Else LV_CDN_DIAG_IGC_SCG[x] = 0

LV_CDN_DIAG_IGC_OL[x] = 0

Endif

Formula section:

If LV_CDN_DIAG_IGC_SCG[x] =1

And LV_CDN_DIAG_IGC_OL[x] =1

Then **If** LV_OL_IGC[x] = 1

Then ERR_SYM_IGC_OL[x] = 4H

Else ERR_SYM_IGC_OL[x] = 0H

Endif

Else No action on the antibounce counter

Endif

If LV_CDN_DIAG_IGC_SCG[x] =1

And LV_CDN_DIAG_IGC_OL[x] =1

Then **If** LV_SCG_IGC[x] = 1

Then ERR_SYM_IGC_SCG[x] = 2H

Else ERR_SYM_IGC_SCG[x] = 0H

Endif

Else No action on the antibounce counter

Endif

Debounce:

Anti - bounce counter increment: C_ABC_INC_IGC_OL

C_ABC_INC_IGC_SCG

Maximum value of anti - bounce counter:C_ABC_MAX_IGC_OL
C_ABC_MAX_IGC_SCG

If LV_CDN_DIAG_IGC_SCG[x] =1
And LV_CDN_DIAG_IGC_OL[x] =1
Then **If** ERR_SYM_IGC_OL[x] = 4H
Then the antibouce OL counter is started (see anti bounce counter)
Else The antibounce OL counter is decremented
Endif
Endif

If LV_CDN_DIAG_IGC_SCG[x] =1
And LV_CDN_DIAG_IGC_OL[x] =1
Then **If** ERR_SYM_IGC_SCG[x] = 2H
Then the antibouce SCG counter is started (see anti bounce counter)
Else The antibounce SCG counter is decremented
Endif
Endif

Anti bounce configuration data:

Diagnosis	Symptom	SYM	ABC type
Ignition diagnosis for OL and SCG	OL	*2*	STD
	SCG	*1*	
LV_ERR_IGC_OL LV_ERR_IGC_SCG			

A.47.3 Or of ignition failure

Initialisation: all outputs variables have to be set to "0" at reset

Recurrence: Every 100 ms

Formula section:

LV_ERR_IGC = LV_ERR_IGC_SCP[x] **OR** LV_ERR_IGC_SCG[x] **OR** LV_ERR_IGC_OL[x]

A.47.4 Unfiltered ignition error information for misfire detection

Application conditions:

Initialisation: IGC_DIAG_MIS = 0 at reset ECU

Recurrence: if NC_IGN_DIAG_TYP =ATM46 every segment
if NC_IGN_DIAG_TYP =ATIC29 every 100 ms

Formula section:

If LV_SCP_IGC[x] = 1
or LV_OL_IGC[x] = 1
or LV_SCG_IGC[x] = 1
Then set the bit for the actual cylinder in IGC_DIAG_MIS = 1
Else reset the bit for the actual cylinder in IGC_DIAG_MIS = 0
Endif

Description:

The error information of the ignition system should not be filtered. The different error detection are linked by a wired "OR". As soon as an ignition error is detected the information should be available as output cylinder individually in the byte IGC_DIAG_MIS.

IGC_DIAG_MIS

Bit Low to high	4 cylinder	5 cylinder	6 cylinder	8 cylinder
0	Cyl 0	Cyl 0	Cyl 0	Cyl 0
1	Cyl 1	Cyl 1	Cyl 1	Cyl 1
2	Cyl 2	Cyl 2	Cyl 2	Cyl 2
3	Cyl 3	Cyl 3	Cyl 3	Cyl 3
4		Cyl 4	Cyl 4	Cyl 4
5			Cyl 5	Cyl 5
6				Cyl 6
7				Cyl 7

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_ABC_IGC_SCP	1	0...FFH	0...255	1	-
Value of anti-bounce counter after regeneration cycle Typical value for debug = 128					
C_IGA_OFF_CYCNR_HLD	1	0...FFH	0...255	1	-
Duration for regeneration cycle Recurrence number; coil has been switched off Typical value for debug = 16					
C_ABC_INC_IGC_SCP	1	0...FFH	0...255	1	-
Anti – bounce counter increment for the ignition system diagnosis SCP Typical value for debug = 5					
C_ABC_INC_IGC_OL	1	0...FFH	0...255	1	-
Anti – bounce counter increment for the ignition system diagnosis OL Typical value for debug = 5					
C_ABC_INC_IGC_SCG	1	0...FFH	0...255	1	-
Anti – bounce counter increment for the ignition system diagnosis SCG Typical value for debug = 5					
C_ABC_MAX_IGC_SCP	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter for the ignition system diagnosis to switch off the coil Typical value for debug = 255					
C_ABC_MAX_IGC_OL	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter for the ignition system diagnosis OL Typical value for debug = 255					
C_ABC_MAX_IGC_SCG	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter for the ignition system diagnosis SCG Typical value for debug = 255					
C_SUM_IGC_SCP	1	0...FFH	0...255	1	-
Number of cycles before switching off the coil until reset Typical value for debug = 255					
C_ESB_IGN_CUT_DIAG	1	0...01H	0...1	1	-
C_ESB_IGN_CUT_DIAG = 1 will inhibit the function Typical value for debug = 0					
C_TCO_MIN_IGC_DIAG	1	0..FEH	-48..142.5	0.75	°C
Min Temperature for condition diagnosis OL & SCG Typical value for debug = 20 ° (emission cycle starting)					
C_N_32_MIN_SCP_IGC_DIAG	1	0..FFH	0..8160	32	Rpm
Min engine speed for condition diagnosis SCP Typical value for debug = 500 rpm					
C_N_32_MIN_OL_SCG_IGC_DIAG	1	0..FFH	0..8160	32	Rpm
Min engine speed for condition diagnosis OL/SCG Typical value for debug = 500 rpm					

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_IGC_DLY	1	0...FFH	0...255	1	-
Number of recurrence after engine start, which had to pass by to activate the diagnosis value =16					
NC_IGN_DIAG_TYP	1	0..FF	0..255	1	-
ATM46 ATIC29 SHUNT (not supported today) Typical value for debug = ATIC29					

Note: typical value for debug is proposed for default value of the Calibration

A.48 Application incidences for ignition diagnosis

A.48.1 IGBT protection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
INH_IV_IGC	O/V	0..FFH	0..255	1	-
Injection cut off					

Input data:

LV_INH_IGC_x			

Description:

To protect the catalyst in case of ignition driver cut off it is necessary to shut off the injection of the concerning cylinder as well.

Formula section:

Shut off injection sequence of IGBT protection:

if LV_INH_IGC_x = 1
then set bit x in INH_IV_IGC
else reset bit x in INH_IV_IGC
endif

A.48.2 Application incidence P_Code

Recurrence

if NC_IGN_ DIAG_TYP =ATM46 every segment

DTC configuration

Diagnosis / Failure Flag	Symptom		DTC
OL circuit ignition driver cylinder 0	Bit 0	Symptom Failure	P0351
	Bit 1		
	Bit 2		
ERR_SYM_IGC_OL_0	Bit 3		
OL circuit ignition driver cylinder 0	Bit 0	Failure	P0351
	Bit 1		
	Bit 2		
LV_ERR_IGC_OL_0	Bit 3		
SCG circuit ignition driver cylinder 0	Bit 0	Symptom Failure	P0351
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCG_0	Bit 3		
SCG circuit ignition driver cylinder 0	Bit 0	Failure	P0351
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCG_0	Bit 3		
SCP circuit ignition driver cylinder 0	Bit 0	Symptom Failure	P0351
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCP_0	Bit 3		
SCP circuit ignition driver cylinder 0	Bit 0	Failure	P0351
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCP_0	Bit 3		

OL circuit ignition driver cylinder 1	Bit 0	Symptom Failure	P0352
	Bit 1		
	Bit 2		
ERR_SYM_IGC_OL_1	Bit 3		
OL circuit ignition driver cylinder 1	Bit 0	Failure	P0352
	Bit 1		
	Bit 2		
LV_ERR_IGC_OL_1	Bit 3		
SCG circuit ignition driver cylinder 1	Bit 0	Symptom Failure	P0352
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCG_1	Bit 3		
SCG circuit ignition driver cylinder 1	Bit 0	Failure	P0352
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCG_1	Bit 3		
SCP circuit ignition driver cylinder 1	Bit 0	Symptom Failure	P0352
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCP_1	Bit 3		
SCP circuit ignition driver cylinder 1	Bit 0	Failure	P0352
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCP_1	Bit 3		

OL circuit ignition driver cylinder 2	Bit 0	Symptom Failure	P0353
	Bit 1		
	Bit 2		
ERR_SYM_IGC_OL_2	Bit 3		
OL circuit ignition driver cylinder 2	Bit 0	Failure	P0353
	Bit 1		
	Bit 2		
LV_ERR_IGC_OL_2	Bit 3		
SCG circuit ignition driver cylinder 2	Bit 0	Symptom Failure	P0353
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCG_2	Bit 3		
SCG circuit ignition driver cylinder 2	Bit 0	Failure	P0353
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCG_2	Bit 3		
SCP circuit ignition driver cylinder 2	Bit 0	Symptom Failure	P0353
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCP_2	Bit 3		
SCP circuit ignition driver cylinder 2	Bit 0	Failure	P0353
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCP_2	Bit 3		

OL circuit ignition driver cylinder 3	Bit 0	Symptom Failure	P0354
	Bit 1		
	Bit 2		
ERR_SYM_IGC_OL_3	Bit 3		
OL circuit ignition driver cylinder 3	Bit 0	Failure	P0354
	Bit 1		
	Bit 2		
LV_ERR_IGC_OL_3	Bit 3		
SCG circuit ignition driver cylinder 3	Bit 0	Symptom Failure	P0354
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCG_3	Bit 3		
SCG circuit ignition driver cylinder 3	Bit 0	Failure	P0354
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCG_3	Bit 3		
SCP circuit ignition driver cylinder 3	Bit 0	Symptom Failure	P0354
	Bit 1		
	Bit 2		
ERR_SYM_IGC_SCP_3	Bit 3		
SCP circuit ignition driver cylinder 3	Bit 0	Failure	P0354
	Bit 1		
	Bit 2		
LV_ERR_IGC_SCP_3	Bit 3		

A.49 IVVT Crankshaft to Camshaft Mechanics Violation Diagnosis

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_MEC_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis crankshaft to inlet camshaft mechanics violation					
LV_ERR_MEC_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis crankshaft to exhaust camshaft mechanics violation					
LV_END_DIAG_MEC_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet					
LV_END_DIAG_MEC_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust					
LV_CDN_DIAG_MEC_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); inlet					
LV_CDN_DIAG_MEC_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); exhaust					
ERR_SYM_MEC_IVVT_IN_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); inlet					
ERR_SYM_MEC_IVVT_EX_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); exhaust					

Input data:

CAM_ADJ_MAX_IVVT_IN	CAM_ADJ_MAX_IVVT_EX	LV_ES	LV_AD_END_IVVT
LV_INH_MEC_IVVT_IN_i	LV_INH_MEC_IVVT_EX_i	CAM_AV_IVVT_IN[NC_NR_CBK_IVVT]	CAM_AV_IVVT_EX[NC_NR_CBK_IVVT]
C_CAM_INI_IN	C_CAM_INI_EX	LV_IGK	LV_ST
N_32	CONF_CAM_VVT_EX		

FUNCTION DESCRIPTION:

General information:

The crankshaft to camshaft reference violation means that the defined phasing has been changed, e.g., due to incorrect engine repair. If this mistake does not lead to damage of the engine it is detected by a diagnosis by ENSD during the activation adaptation of the reference position.

There can be a jump of the chain/belt during the engine. It can be recognized if the camshaft position leaves the possible adjustment range operation by the crankshaft to camshaft mechanics violation diagnosis.

Description:

The crankshaft to camshaft mechanics violation diagnosis is enabled if the crankshaft to camshaft reference violation diagnosis (ENSD) has not detected errors, i.e., the activation adaptation has finished successfully, and the engine is running.

Error-symptoms are defined to this diagnostic function as follows:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> (not used)                          (= SYM_0)
 │  │  └────> (not used)                          (= SYM_1)
 │  └───────> (not used)                          (= SYM_2)
 └──────────> MEC_IVVT_IN(_EX)_i                  (= SYM_3)
```

Application conditions:

Initialization:

LV_CDN_DIAG_MEC_IVVT_IN(_EX)_i = 0

LV_END_DIAG_MEC_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

LV_ERR_MEC_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

ERR_SYM_MEC_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

At activation:

LV_CDN_DIAG_MEC_IVVT_IN(_EX)_i = 1

At deactivation:

LV_CDN_DIAG_MEC_IVVT_IN(_EX)_i = 0

Recurrence:

100 ms

Activation:

LV_INH_MEC_IVVT_IN(_EX)_i = 0

and LV_ES = 0

and LV_ST = 0

and LV_IGK = 1

and LV_AD_END_IVVT = 1

and N_32 > C_N_MIN_MEC_IVVT_IN(_EX)

and *additionally only for Exhaust side (_EX)*

CONF_CAM_VVT_EX = 1

Formula section:

Symptoms calculation:

SYM_3:

if Inlet:

CAM_AV_IVVT_IN_i >
C_CAM_INI_IN + C_CAM_HYS_MEC_IVVT

or CAM_AV_IVVT_IN_i <
CAM_ADJ_MAX_IVVT_IN - C_CAM_HYS_MEC_IVVT

Exhaust:

CAM_AV_IVVT_EX_i <
C_CAM_INI_EX - C_CAM_HYS_MEC_IVVT

or CAM_AV_IVVT_EX_i >
CAM_ADJ_MAX_IVVT_EX + C_CAM_HYS_MEC_IVVT

then

ERR_SYM_MEC_IVVT_IN(_EX)_i = "SYM_3" active

else

ERR_SYM_MEC_IVVT_IN(_EX)_i = "SYM_3" inactive

endif

Filtering:

Apply filter on current symptoms

if filtering result available (after debounce)

then

LV_ERR_MEC_IVVT_IN(_EX)_i = filtering result

LV_END_DIAG_MEC_IVVT_IN(_EX)_i = 1

Deliver the result to Error Management

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_MEC_IVVT	1	0...7FH	0...47.625	0.375	°CRK
Hysteresis for crankshaft to camshaft mechanics violation diagnosis					
C_N_MIN_MEC_IVVT_IN	1	0...FFH	0...8160	32	rpm
Minimum engine speed for mechanics diagnosis activation : inlet					
C_N_MIN_MEC_IVVT_EX	1	0...FFH	0...8160	32	rpm
Minimum engine speed for mechanics diagnosis activation : exhaust					

A.50 IVVT Camshaft Position Deviation Diagnosis

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_CAM_DE_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft deviation					
LV_ERR_CAM_DE_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft deviation					
LV_ERR_CAM_STAT_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft deviation at steady setpoint					
LV_ERR_CAM_STAT_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft deviation at steady setpoint					
LV_ERR_CAM_STAT_0_IVVT_IN_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft deviation at steady setpoint – previous calculation step					
LV_ERR_CAM_STAT_0_IVVT_EX_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft deviation at steady setpoint – previous calculation step					
LV_END_DIAG_CAM_STAT_IVVT_IN_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet					
LV_END_DIAG_CAM_STAT_IVVT_EX_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust					
LV_END_CAM_STAT_0_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet – previous calculation step					
LV_END_CAM_STAT_0_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust – previous calculation step					
LV_CDN_DIAG_CAM_STAT_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); inlet					
LV_CDN_DIAG_CAM_STAT_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); exhaust					
ERR_SYM_CAM_STAT_IVVT_IN_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); inlet					
ERR_SYM_CAM_STAT_IVVT_EX_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); exhaust					
CAM_DIF_INT_DIAG_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position deviation integrator for diagnosis; inlet					
CAM_DIF_INT_DIAG_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position deviation integrator for diagnosis; exhaust					
CAM_DIF_INT_DIAG_RST_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position deviation integrator for reseting diagnosis; inlet					
CAM_DIF_INT_DIAG_RST_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position deviation integrator for reseting diagnosis; exhaust					
CAM_DIF_DIAG_STAT_SAE_IVVT_IN_i	O/S/V	0...FFH	0...95.625	0.375	°CRK
Camshaft position deviation value for mode 6 output; inlet					
CAM_DIF_DIAG_STAT_SAE_IVVT_EX_i	O/S/V	0...FFH	0...95.625	0.375	°CRK
Camshaft position deviation value for mode 6 output; exhaust					
CTR_CAM_SP_NEUT_IVVT_IN_i	V	0...1770H	0...6000	1	-
Min. number of edges to start diagnosis after setpoint reached reference position; inlet					
CTR_CAM_SP_NEUT_IVVT_EX_i	V	0...1770H	0...6000	1	-
Min. number of edges to start diagnosis after setpoint reached reference position; exhaust					
LV_ERR_CAM_DYN_IVVT_IN_i	O/V	0...1H	0...1	1	-

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
Present failure: failure after filtering of diagnosis enduring inlet camshaft deviation at changing setpoint					
LV_ERR_CAM_DYN_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft deviation at changing setpoint					
LV_ERR_CAM_DYN_0_IVVT_IN_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft deviation at changing setpoint – previous calculation step					
LV_ERR_CAM_DYN_0_IVVT_EX_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft deviation at changing setpoint – previous calculation step					
LV_END_DIAG_CAM_DYN_IVVT_IN_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet					
LV_END_DIAG_CAM_DYN_IVVT_EX_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust					
LV_END_CAM_DYN_0_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet – previous calculation step					
LV_END_CAM_DYN_0_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust – previous calculation step					
LV_CDN_DIAG_CAM_DYN_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); inlet					
LV_CDN_DIAG_CAM_DYN_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); exhaust					
ERR_SYM_CAM_DYN_IVVT_IN_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); inlet					
ERR_SYM_CAM_DYN_IVVT_EX_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); exhaust					
CAM_SP_PREV_DIAG_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Previous camshaft position setpoint for diagnosis; inlet					
CAM_SP_PREV_DIAG_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Previous camshaft position setpoint for diagnosis; exhaust					
CAM_SP_DYW_DIAG_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Help variable for deterimination of limited dynamics of camshaft position setpoint for diagnosis; inlet					
CAM_SP_DYW_DIAG_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Help variable for deterimination of limited dynamics of camshaft position setpoint for diagnosis; exhaust					
T_CAM_SP_DYW_DIAG_IVVT_IN_i	V	0...FFFFH	0...4.194304	6.4e-5	s
Time of keeping one dynamic window of camshaft position setpoint for diagnosis; inlet					
T_CAM_SP_DYW_DIAG_IVVT_EX_i	V	0...FFFFH	0...4.194304	6.4e-5	s
Time of keeping one dynamic window of camshaft position setpoint for diagnosis; exhaust					
DLY_DIAG_DYN_IVVT_IN_i	V	0...FFFFH	0...4.194304	6.4e-5	s
Time of keeping camshaft position setpoint within dynamic window before starting dynamic diagnosis, overflow not allowed; inlet					
DLY_DIAG_DYN_IVVT_EX_i	V	0...FFFFH	0...4.194304	6.4e-5	s
Time of keeping camshaft position setpoint within dynamic window before starting dynamic diagnosis, overflow not allowed; exhaust					
LV_CAM_SP_DYW_DIAG_IVVT_IN_i	V	0...1H	0...1	1	-
1 = limited dynamics of camshaft position setpoint for diagnosis fulfilled, 0 = not fulfilled; inlet					
LV_CAM_SP_DYW_DIAG_IVVT_EX_i	V	0...1H	0...1	1	-
1 = limited dynamics of camshaft position setpoint for diagnosis fulfilled, 0 = not fulfilled; exhaust					
LV_CAM_SP_DYW_0_DIAG_IVVT_IN_i	V	0...1H	0...1	1	-
Value of LV_CAM_SP_DYW_DIAG_IVVT_IN_i in previous calculation step					
LV_CAM_SP_DYW_0_DIAG_IVVT_EX_i	V	0...1H	0...1	1	-
Value of LV_CAM_SP_DYW_DIAG_IVVT_EX_i in previous calculation step					
LV_DIAG_DYN_ACT_IVVT_IN_i	V	0...1H	0...1	1	-

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
1 = dynamic IVVT diagnosis active, 0 = not active; inlet					
LV_DIAG_DYN_ACT_IVVT_EX_i	V	0...1H	0...1	1	-
1 = dynamic IVVT diagnosis active, 0 = not active; exhaust					
CAM_SP_DIAG_DYN_ST_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint at dynamic diagnosis start; inlet					
CAM_SP_DIAG_DYN_ST_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint at dynamic diagnosis start; exhaust					
CAM_DIAG_DYN_ST_IVVT_IN_i	V	0...FFH	60...155.625	0.375	°CRK
Camshaft position at dynamic diagnosis start; inlet					
CAM_DIAG_DYN_ST_IVVT_EX_i	V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position at dynamic diagnosis start; exhaust					
T_DIAG_DYN_IVVT_IN_i	V	0...FFFFH	0...4.194304	6.4e-5	s
IVVT dynamic diagnosis timer; inlet					
T_DIAG_DYN_IVVT_EX_i	V	0...FFFFH	0...4.194304	6.4e-5	s
IVVT dynamic diagnosis timer; exhaust					
CAM_SP_DIR_IVVT_IN_i	V	FF...1H	-1...1	1	-
Direction of setpoint change; inlet					
CAM_SP_DIR_IVVT_EX_i	V	FF...1H	-1...1	1	-
Direction of setpoint change; exhaust					
T_MIN_DIAG_DYN_IVVT_IN_i	V	0...FFH	4.194304	0.016384	s
Time of IVVT dynamic diagnosis evaluation after its start; inlet					
T_MIN_DIAG_DYN_IVVT_EX_i	V	0...FFH	4.194304	0.016384	s
Time of IVVT dynamic diagnosis evaluation after its start; exhaust					
CAM_DIF_DIAG_DYN_SAVE_IN_i	V	0...FFH	0...95.625	0.375	°CRK
Camshaft position change during dynamic diagnosis at symptom setting; inlet					
CAM_DIF_DIAG_DYN_SAVE_EX_i	V	0...FFH	0...95.625	0.375	°CRK
Camshaft position change during dynamic diagnosis at symptom setting; exhaust					
CAM_DIF_DIAG_DYN_SAE_IVVT_IN_i	O/S/V	0...FFH	0...95.625	0.375	°CRK
Camshaft position change during dynamic diagnosis at failure detection; for mode 6 output; inlet					
CAM_DIF_DIAG_DYN_SAE_IVVT_EX_i	O/S/V	0...FFH	0...95.625	0.375	°CRK
Camshaft position change during dynamic diagnosis at failure detection; for mode 6 output; exhaust					
LV_ACT_IND_SAVE_IVVT_IN_i	-	0…1H	0…1	1	[-]
LV_ACT_IND_IVVT_IN in previous calculation step					
LV_ACT_IND_SAVE_IVVT_EX_i	-	0…1H	0…1	1	[-]
LV_ACT_IND_IVVT_EX in previous calculation step					

Input data:

LV_CAM_SP_DYW_IVVT_IN_i	LV_CAM_SP_DYW_IVVT_EX_i	C_CAM_INI_IN	N_32
LV_INH_CAM_STAT_IVVT_IN_i	LV_INH_CAM_STAT_IVVT_EX_i	C_CAM_INI_EX	TOIL
LV_INH_CAM_DYN_IVVT_IN_i	LV_INH_CAM_DYN_IVVT_EX_i	CAM_AV_IVVT_IN_i	VB
LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	CAM_AV_IVVT_EX_i	
LV_DE_AD_END_IVVT_IN_i	LV_DE_AD_END_IVVT_EX_i	CONF_CAM_VVT_EX	
T_DIF_EDGE_CAM_IVVT_IN_i	T_DIF_EDGE_CAM_IVVT_EX_i	LV_ERR_CAM_PAS_IVVT_IN_i	
CAM_SP_IND_DIAG_IVVT_IN_i	CAM_SP_IND_DIAG_IVVT_EX_i	LV_ERR_CAM_PAS_IVVT_EX_i	
C_CAM_SP_HYS_NEUT_IVVT_IN	LV_ERR_SLV_IVVT_IN_i		
C_CAM_SP_HYS_NEUT_IVVT_EX	LV_ERR_SLV_IVVT_EX_i		
IP_CYC_MIN_DE_AD_IVVT			

FUNCTION DESCRIPTION:

General information:

A jammed or a slow-response actuator is detected by means of a comparison of the camshaft position and the setpoint and quantities derived from them. Two cases are distinguished: detection at steady setpoint and at changing setpoint.

Description:

In order to avoid wrong error detections by the camshaft position deviation diagnosis, the diagnosis is enabled only in certain IVVT operation range.

Camshaft position deviation diagnosis at steady setpoint:

There is a strong coherence with the deviation adaptation of the holding pulse width modulation. The deviation adaptation is started after a setpoint change at first. The diagnosis is started after the adaptation. If the camshaft position deviation after a setpoint change is bigger than a threshold the adaptation is omitted and the diagnosis is started immediately.

If the setpoint reaches the reference position the deviation adaptation is inactive. The diagnosis has to be disabled for some time till the camshaft reaches the reference position. It is realized by means of CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i.

The principle of the diagnosis is based on two integrators. The first one, CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i, integrates the absolute value of the deviation in time if the deviation is bigger than the threshold C_CAM_HYS_DE_DIAG_IVVT_IN(_EX). If the deviation is smaller than this threshold the first integrator is set to 0 and the second one, CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i, integrates the threshold C_CAM_HYS_DE_DIAG_IVVT_IN(_EX) in time. If one of the integrators reaches the diagnosis threshold C_CAM_DIF_INT_MAX_DIAG_IVVT_IN(_EX) the symptom ERR_SYM_CAM_STAT_IVVT_IN(_EX)_i can be set. If it is the integrator CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i the symptom is set to failure. If it is the integrator CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i the symptom is set inactive.

This diagnosis principle is not sensitive for short stochastical temporary camshaft position deviations.

CAM_DIF_INT_DIAG_IVVT_IN(_EX) shows the last diagnostic result. It is to compare with C_CAM_DIF_INT_MAX_DIAG_IVVT_IN(_EX). If the integrated value is greater than or equal to the calibration datum it indicates a failure. The opposite case indicates a faultless performance. The saved value is actually zero.

Camshaft position deviation diagnosis at changing setpoint:

The setpoint has to be stable for certain time before starting the dynamic diagnosis. One possibility is that the IVVT system was inactive, i.e. the reference position was the target. Or the setpoint was within the time moving dynamic window IP_CAM_SP_DYW_DIAG_IVVT_IN(_EX) at least C_DLY_MIN_DIAG_DYN_IVVT. The time moving dynamic window means that after the time C_T_CAM_SP_DYW_DIAG_IVVT the current setpoint is taken for the next window. Thus, the diagnosis can work with some setpoint gradient and jitter.

If the setpoint leaves the dynamic window the previous setpoint, the setpoint change direction and the current camshaft position are assigned into dynamic diagnosis start variables. A time counter is started and the diagnosis becomes active. If the setpoint change keeps its mean direction, i.e. there is no strong opposite change and there is a setpoint gradient greater than a minmum required one, the time counting continues. If the timer T_MIN_DIAG_DYN_IVVT_IN(_EX)_i = IP_T_MIN_DIAG_DYN_ADC(_RTD)_IVVT_IN(_EX) elapses the setpoint change over the diagnosis has to be bigger than C_CAM_SP_DIF_DIAG_DYN_IVVT_IN(_EX). These calibration data depend among each other. Then it is possible to evaluate the camshaft position change over the diagnosis. If it is greater than C_CAM_DIF_DIAG_DYN_IVVT_IN(_EX) there is no failure, otherwise a jammed or slow-responsing actuator is assumed.

CAM_DIF_DIAG_DYN_SAVE_IN(_EX)_i shows the last diagnostic result. It is to compare with C_CAM_DIF_DIAG_DYN_IVVT_IN(_EX). If the saved value is greater than or equal to the calibration datum it indicates a faultless performance. The opposite case indicates a failure.

Error-symptoms definition:

Error-symptoms are defined to this diagnostic function as follows:

LV_ERR_CAM_STAT_IVVT_IN(_EX)_i:

```
□□□□
│ │ │ └─> (not used)                    (= SYM_0)
│ │ └───> (not used)                    (= SYM_1)
│ └─────> (not used)                    (= SYM_2)
└───────> CAM_STAT_IVVT_IN(_EX)_i       (= SYM_3)
```

LV_ERR_CAM_DYN_IVVT_IN(_EX)_i:

```
□□□□
│ │ │ └─> (not used)                    (= SYM_0)
│ │ └───> (not used)                    (= SYM_1)
│ └─────> (not used)                    (= SYM_2)
└───────> CAM_DYN_IVVT_IN(_EX)_i        (= SYM_3)
```

Application conditions:

Initialization:

LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 0

LV_END_DIAG_CAM_STAT_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

LV_ERR_CAM_STAT_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

ERR_SYM_CAM_STAT_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

LV_CDN_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 0

LV_END_DIAG_CAM_DYN_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

LV_ERR_CAM_DYN_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

ERR_SYM_CAM_DYN_IVVT_IN(_EX)_i =
Refer to filtering configuration for the initialization value

LV_ERR_CAM_DE_IVVT_IN(_EX)_i =

LV_ERR_CAM_STAT_IVVT_IN(_EX)_i

or LV_ERR_CAM_DYN_IVVT_IN(_EX)_i

or LV_ERR_CAM_PAS_IVVT_IN(_EX)_i

At activation:

if LV_ACT_IND_SAVE_IVVT_IN(_EX)_i = 0

and LV_ACT_IND_IVVT_IN(_EX)_i = 1

then *Start from reference position*

CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

LV_CAM_SP_DYW_0_DIAG_IVVT_IN(_EX)_i = 1

DLY_DIAG_DYN_IVVT_IN(_EX)_i = FFFFH

else

CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i =

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

LV_CAM_SP_DYW_0_DIAG_IVVT_IN(_EX)_i = 0

DLY_DIAG_DYN_IVVT_IN(_EX)_i = 0H

endif

CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i = 0

LV_END_CAM_STAT_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_STAT_0_IVVT_IN(_EX)_i = 0

LV_END_CAM_DYN_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_DYN_0_IVVT_IN(_EX)_i = 0

At deactivation:

LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 0

LV_CDN_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 0

LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 0

At failure memory reset:

LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 0

LV_CDN_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 0

LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 0

CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i = CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

LV_CAM_SP_DYW_0_DIAG_IVVT_IN(_EX)_i = 0

DLY_DIAG_DYN_IVVT_IN(_EX)_i = 0

CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i = 0

CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i = 0

CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_DE_IVVT_IN(_EX)_i = 0

LV_END_CAM_STAT_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_STAT_0_IVVT_IN(_EX)_i = 0

LV_END_CAM_DYN_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_DYN_0_IVVT_IN(_EX)_i = 0

At failure memory reset, in case of checksum error:

CAM_DIF_DIAG_STAT_SAE_IVVT_IN(_EX)_i = 0

CAM_DIF_DIAG_DYN_SAE_IVVT_IN(_EX)_i = 0

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_INH_CAM_STAT_IVVT_IN(_EX)_i = 0

and LV_INH_CAM_DYN_IVVT_IN(_EX)_i = 0

and LV_ACT_IND_IVVT_IN(_EX)_i = 1

and VB >= C_VB_DIAG_MEC_IVVT

and IP_N_32_MIN_DIAG_MEC_IVVT <= N_32 <= IP_N_32_MAX_DIAG_MEC_IVVT

and C_TOIL_MIN_DIAG_MEC_IVVT <= TOIL <= C_TOIL_MAX_DIAG_MEC_IVVT

and LV_ERR_SLV_IVVT_IN(_EX)_i = 0

and LV_ERR_CAM_PAS_IVVT_IN(_EX)_i = 0

and *additionally only for Exhaust side (_EX)*

CONF_CAM_VVT_EX = 1

Deactivation:

Not activation

Formula section:

LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 0

LV_CDN_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 0

Setpoint in dynamic window

Check for setpoint change

if LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 1

then

Symptom detection at changing setpoint

endif

if LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 1

and LV_DE_AD_END_IVVT_IN(_EX)_i = 1

then

Deviation integration at steady setpoint

Symptoms calculation at steady setpoint

else

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i = 0

endif

Filtering

Global error

Mode 06 output

if LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

then

DLY_DIAG_DYN_IVVT_IN(_EX)_i = 0

endif

CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i = CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

LV_CAM_SP_DYW_0_DIAG_IVVT_IN(_EX)_i =

LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i

Setpoint in dynamic window:

if abs(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -

CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i) <=

IP_CAM_SP_DYW_DIAG_IVVT_IN(_EX)

then *Setpoint within dynamic window*

LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 1

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i =

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i +

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

DLY_DIAG_DYN_IVVT_IN(_EX)_i =

DLY_DIAG_DYN_IVVT_IN(_EX)_i +

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

if T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i >=

C_T_CAM_SP_DYW_DIAG_IVVT

then *New dynamic window after defined period*

CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i =

CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

endif

else *Setpoint changes significantly*

LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

T_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

endif

Check for setpoint change:

if LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 0

and LV_CAM_SP_DYW_0_DIAG_IVVT_IN(_EX)_i = 1

and LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 0

and LV_DE_AD_END_IVVT_IN(_EX)_i = 1

and DLY_DIAG_DYN_IVVT_IN(_EX)_i > C_DLY_MIN_DIAG_DYN_IVVT

then *Significant setpoint change recognized*

LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 1

CAM_SP_DIAG_DYN_ST_IVVT_IN(_EX)_i = CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i

CAM_DIAG_DYN_ST_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i

T_DIAG_DYN_IVVT_IN(_EX)_i = T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

Direction of the setpoint change:

CAM_SP_DIR_IVVT_IN(_EX)_i = sign(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i - CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i)

endif

Symptom detection at changing setpoint:

SYM_3:

if CAM_SP_DIR_IVVT_IN(_EX)_i = -1

then

T_MIN_DIAG_DYN_IVVT_IN(_EX)_i = IP_T_MIN_DIAG_DYN_ADC_IVVT_IN(_EX)

else

T_MIN_DIAG_DYN_IVVT_IN(_EX)_i = IP_T_MIN_DIAG_DYN_RTD_IVVT_IN(_EX)

endif

if (T_DIAG_DYN_IVVT_IN(_EX)_i < T_MIN_DIAG_DYN_IVVT_IN(_EX)_i * C_FAC_T_MIN_DIAG_DYN_IVVT_IN(_EX)

and (CAM_SP_DIR_IVVT_IN(_EX)_i = sign(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -

CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i)
or LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 1))
or
(T_DIAG_DYN_IVVT_IN(_EX)_i >=
T_MIN_DIAG_DYN_IVVT_IN(_EX)_i *
C_FAC_T_MIN_DIAG_DYN_IVVT_IN(_EX)
and (CAM_SP_DIR_IVVT_IN(_EX)_i =
sign(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -
CAM_SP_PREV_DIAG_IVVT_IN(_EX)_i)
or LV_CAM_SP_DYW_DIAG_IVVT_IN(_EX)_i = 1)
and C_CAM_SP_DIF_DIAG_DYN_IVVT_IN(_EX) /
T_MIN_DIAG_DYN_IVVT_IN(_EX)_i <=
abs((CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -
CAM_SP_DIAG_DYN_ST_IVVT_IN(_EX)_i) /
T_DIAG_DYN_IVVT_IN(_EX)_i))

then *Setpoint change continues or no strong change in opposite direction*
if T_DIAG_DYN_IVVT_IN(_EX)_i <
T_MIN_DIAG_DYN_IVVT_IN(_EX)_i
then *Time for the setpoint change evaluation not elapsed yet*
T_DIAG_DYN_IVVT_IN(_EX)_i =
T_DIAG_DYN_IVVT_IN(_EX)_i +
T_DIF_EDGE_CAM_IVVT_IN(_EX)_i
else *Evaluation of the setpoint change*
if abs(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -
CAM_SP_DIAG_DYN_ST_IVVT_IN(_EX)_i) >=
C_CAM_SP_DIF_DIAG_DYN_IVVT_IN(_EX)
then *Setpoint change big enough*
LV_CDN_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 1
CAM_DIF_DIAG_DYN_SAVE_IN(_EX)_i =
abs(CAM_AV_IVVT_IN(_EX)_i -
CAM_DIAG_DYN_ST_IVVT_IN(_EX)_i)
if CAM_DIF_DIAG_DYN_SAVE_IN(_EX)_i >=
C_CAM_DIF_DIAG_DYN_IVVT_IN(_EX)
then *Camshaft position changed satisfactory*
ERR_SYM_CAM_DYN_IVVT_IN(_EX)_i =
"SYM_3" inactive
else *Actuator jammed or slow-responsing*

ERR_SYM_CAM_DYN_IVVT_IN(_EX)_i =
"SYM_3" active

endif

endif

LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 0

endif

else *Strong change of setpoint in opposite direction*

LV_DIAG_DYN_ACT_IVVT_IN(_EX)_i = 0

endif

Deviation integration at steady setpoint:

if Inlet:

CAM_SP_IND_DIAG_IVVT_IN_i >

C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

Exhaust:

CAM_SP_IND_DIAG_IVVT_EX_i <

C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX

then

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i =

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i + 1

else

CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i = 0

endif

if Inlet:

CAM_SP_IND_DIAG_IVVT_IN_i <=

C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

or (CAM_SP_IND_DIAG_IVVT_IN_i >

C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

and CTR_CAM_SP_NEUT_IVVT_IN_i >=

IP_CYC_MIN_DE_AD_IVVT

)

Exhaust:

CAM_SP_IND_DIAG_IVVT_EX_i >=

C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX

or (CAM_SP_IND_DIAG_IVVT_EX_i <

C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX

and CTR_CAM_SP_NEUT_IVVT_EX_i >=

IP_CYC_MIN_DE_AD_IVVT

```
                    )
    then
        if      |CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_DIAG_IVVT_IN(_EX)_i| >
                C_CAM_HYS_DE_DIAG_IVVT_IN(_EX)
        then
                CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i(n) =
                CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i(n-1) +
                |CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_DIAG_IVVT_IN(_EX)_i| *
                T_DIF_EDGE_CAM_IVVT_IN(_EX)_i
        else
                CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i = 0
                If      Inlet:
                        CAM_SP_IND_DIAG_IVVT_IN_i <=
                        C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN
                        Exhaust:
                        CAM_SP_IND_DIAG_IVVT_EX_i >=
                        C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX
                then
                        CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i(n) =
                        CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i(n-1) +
                        C_CAM_HYS_DE_DIAG_IVVT_IN(_EX) *
                        T_DIF_EDGE_CAM_IVVT_IN(_EX)_i
                endif
        endif
    endif
```

Symptoms calculation at steady setpoint:

SYM_3:

if CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i >= C_CAM_DIF_INT_MAX_DIAG_IVVT_IN(_EX)

then LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 1
ERR_SYM_CAM_STAT_IVVT_IN(_EX)_i = "SYM_3" active
CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i = 0
CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i = 0

else **if** CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i >= C_CAM_DIF_INT_MAX_DIAG_IVVT_IN(_EX)

then LV_CDN_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 1
ERR_SYM_CAM_STAT_IVVT_IN(_EX)_i = "SYM_3" inactive
CAM_DIF_INT_DIAG_IVVT_IN(_EX)_i = 0
CAM_DIF_INT_DIAG_RST_IVVT_IN(_EX)_i = 0

endif

endif

Filtering:

Apply filter on current symptoms

if filtering result available (after debounce)

then LV_ERR_CAM_STAT_IVVT_IN(_EX)_i = filtering result
LV_END_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 1
Deliver the result to Error Management

endif

Apply filter on current symptoms

if filtering result available (after debounce)

then LV_ERR_CAM_DYN_IVVT_IN(_EX)_i = filtering result
LV_END_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 1
Deliver the result to Error Management

endif

Global error:

LV_ERR_CAM_DE_IVVT_IN(_EX)_i =
LV_ERR_CAM_STAT_IVVT_IN(_EX)_i
or LV_ERR_CAM_DYN_IVVT_IN(_EX)_i
or LV_ERR_CAM_PAS_IVVT_IN(_EX)_i

Mode 06 output:

If (LV_END_CAM_DYN_0_IVVT_IN(_EX)_i = 0
and LV_END_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 1)
or (LV_ERR_CAM_DYN_0_IVVT_IN(_EX)_i = 0
and LV_ERR_CAM_DYN_IVVT_IN(_EX)_i = 1)
then Mode 06 output for dynamic diagnosis
CAM_DIF_DIAG_DYN_SAE_IVVT_IN(_EX)_i =
CAM_DIF_DIAG_DYN_SAVE_IN(_EX)_i
LV_END_CAM_DYN_0_IVVT_IN(_EX)_i =
LV_END_DIAG_CAM_DYN_IVVT_IN(_EX)_i
LV_ERR_CAM_DYN_0_IVVT_IN(_EX)_i =
LV_ERR_CAM_DYN_IVVT_IN(_EX)_i
endif

If (LV_END_CAM_STAT_0_IVVT_IN(_EX)_i = 0
and LV_END_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 1)
or (LV_ERR_CAM_STAT_0_IVVT_IN(_EX)_i = 0
and LV_ERR_CAM_STAT_IVVT_IN(_EX)_i = 1)
then Mode 06 output for static diagnosis
CAM_DIF_DIAG_STAT_SAE_IVVT_IN(_EX)_i =
|CAM_AV_IVVT_IN(_EX)_i – CAM_SP_IND_DIAG_IVVT_IN(_EX)_i|
LV_END_CAM_STAT_0_IVVT_IN(_EX)_i =
LV_END_DIAG_CAM_STAT_IVVT_IN(_EX)_i
LV_ERR_CAM_STAT_0_IVVT_IN(_EX)_i =
LV_ERR_CAM_STAT_IVVT_IN(_EX)_i
endif

Calculate at every recurrence independent on activation conditions:

LV_ACT_IND_SAVE_IVVT_IN(_EX)_i = LV_ACT_IND_IVVT_IN(_EX)_i

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_VB_DIAG_MEC_IVVT	1	0...FFH	0...26	0.1	V
Minimum battery voltage for IVVT mechanical diagnoses					
C_TOIL_MIN_DIAG_MEC_IVVT	1	0...C8H	-40...160	1	°C
Minimum oil temperature for IVVT mechanical diagnoses					
C_TOIL_MAX_DIAG_MEC_IVVT	1	0...C8H	-40...160	1	°C
Maximum oil temperature for IVVT mechanical diagnoses					
IP_N_32_MIN_DIAG_MEC_IVVT	1x4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Minimum engine speed for IVVT mechanical diagnoses					
IP_N_32_MAX_DIAG_MEC_IVVT	1x4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Maximum engine speed for IVVT mechanical diagnoses					
C_CAM_HYS_DE_DIAG_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Hysteresis for camshaft position deviation diagnosis; inlet					
C_CAM_HYS_DE_DIAG_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Hysteresis for camshaft position deviation diagnosis; exhaust					
C_CAM_DIF_INT_MAX_DIAG_IVVT_IN	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position deviation diagnosis; inlet					
C_CAM_DIF_INT_MAX_DIAG_IVVT_EX	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position deviation diagnosis; exhaust					
IP_CAM_SP_DYW_DIAG_IVVT_IN	1x4	0...FFH	0...95.625	0.375	°CRK
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
Camshaft position setpoint window for determination of limited dynamics for diagnosis; inlet					
IP_CAM_SP_DYW_DIAG_IVVT_EX	1x4	0...FFH	0...95.625	0.375	°CRK
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
Camshaft position setpoint window for determination of limited dynamics for diagnosis; exhaust					
C_T_CAM_SP_DYW_DIAG_IVVT	1	0...FFH	4.194304	0.016384	s
Time to keep one dynamic window of camshaft position setpoint for diagnosis					
C_DLY_MIN_DIAG_DYN_IVVT	1	0...FFH	4.194304	0.016384	s
Time to keep camshaft position setpoint within dynamic window before starting dynamic diagnosis					
IP_T_MIN_DIAG_DYN_ADC_IVVT_IN	4x4	0...FFH	4.194304	0.016384	s
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Time of IVVT dynamic diagnosis evaluation after its start; advanced; inlet					
IP_T_MIN_DIAG_DYN_ADC_IVVT_EX	4x4	0...FFH	4.194304	0.016384	s
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Time of IVVT dynamic diagnosis evaluation after its start; advanced; exhaust					
IP_T_MIN_DIAG_DYN_RTD_IVVT_IN	4x4	0...FFH	4.194304	0.016384	s
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Time of IVVT dynamic diagnosis evaluation after its start; retard; inlet					
IP_T_MIN_DIAG_DYN_RTD_IVVT_EX	4x4	0...FFH	4.194304	0.016384	s
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_3_VVTI	4	0...C8H	-40...160	1	°C
Time of IVVT dynamic diagnosis evaluation after its start; retard; exhaust					
C_CAM_SP_DIF_DIAG_DYN_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Minimum necessary camshaft position setpoint change from start to end of dynamic diagnosis; inlet					
C_CAM_SP_DIF_DIAG_DYN_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Minimum necessary camshaft position setpoint change from start to end of dynamic diagnosis; exhaust					
C_CAM_DIF_DIAG_DYN_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Minimum camshaft position change during dynamic diagnosis for no failure detection; inlet					
C_CAM_DIF_DIAG_DYN_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Minimum camshaft position change during dynamic diagnosis for no failure detection; exhaust					
C_FAC_T_MIN_DIAG_DYN_IVVT_IN	1	0...FFH	0...0.99609	0.0039	-
Portion of dynamic diagnosis duration at its begin w/o monitoring minimum required setpoint gradient; inlet					
C_FAC_T_MIN_DIAG_DYN_IVVT_EX	1	0...FFH	0...0.99609	0.0039	-
Portion of dynamic diagnosis duration at its begin w/o monitoring minimum required setpoint gradient; exhaust					

A.51 IVVT Camshaft Position Partial Deviation Diagnosis

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_CAM_PAS_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft partial deviation at steady setpoint					
LV_ERR_CAM_PAS_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft partial deviation at steady setpoint					
LV_ERR_CAM_PAS_0_IVVT_IN_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring inlet camshaft partial deviation at steady setpoint – previous calculation step					
LV_ERR_CAM_PAS_0_IVVT_EX_i	V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis enduring exhaust camshaft partial deviation at steady setpoint – previous calculation step					
LV_END_DIAG_CAM_PAS_IVVT_IN_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet					
LV_END_DIAG_CAM_PAS_IVVT_EX_i	O/V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust					
LV_END_CAM_PAS_0_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet – previous calculation step					
LV_END_CAM_PAS_0_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust – previous calculation step					
LV_CDN_DIAG_CAM_PAS_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); inlet					
LV_CDN_DIAG_CAM_PAS_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); exhaust					
ERR_SYM_CAM_PAS_IVVT_IN_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); inlet					
ERR_SYM_CAM_PAS_IVVT_EX_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); exhaust					
CAM_INT_PAS_DIAG_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator for diagnosis; inlet					
CAM_INT_PAS_DIAG_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator for diagnosis; exhaust					
CAM_INT_PAS_DIAG_RST_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator for reseting diagnosis; inlet					
CAM_INT_PAS_DIAG_RST_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator for reseting diagnosis; exhaust					
CAM_INT_PAS_SAVE_IVVT_IN_i	O/S/V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator value at symptom setting; inlet					
CAM_INT_PAS_SAVE_IVVT_EX_i	O/S/V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position partial deviation integrator value at symptom setting; exhaust					
CAM_INT_PAS_RTD_SAE_IVVT_IN_i	O/S/V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position retard deviation integrator value for mode 6 output; inlet					
CAM_INT_PAS_ADC_SAE_IVVT_EX_i	O/S/V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position advanced deviation integrator value for mode 6 output; exhaust					
CAM_INT_PAS_ADC_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position advance deviation integrator for diagnosis; inlet					
CAM_INT_PAS_ADC_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position advance deviation integrator for diagnosis; exhaust					
CAM_INT_PAS_RTD_IVVT_IN_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
Camshaft position retard deviation integrator for diagnosis; inlet					
CAM_INT_PAS_RTD_IVVT_EX_i	V	0...FFFFH	0...201.3235	0.003072	°CRK*s
Camshaft position retard deviation integrator for diagnosis; exhaust					
LV_ACT_IND_SAVE_PAS_IVVT_IN_i	-	0…1H	0…1	1	[-]
LV_ACT_IND_IVVT_IN in previous calculation step					
LV_ACT_IND_SAVE_PAS_IVVT_EX_i	-	0…1H	0…1	1	[-]
LV_ACT_IND_IVVT_EX in previous calculation step					

Input data:

CTR_CAM_SP_NEUT_IVVT_IN_i	CTR_CAM_SP_NEUT_IVVT_EX_i	C_CAM_INI_IN	VB
C_CAM_SP_HYS_NEUT_IVVT_IN	C_CAM_SP_HYS_NEUT_IVVT_EX	C_CAM_INI_EX	N_32
LV_CAM_SP_DYW_IVVT_IN_i	LV_CAM_SP_DYW_IVVT_EX_i	LV_ACT_IND_IVVT_IN_i	TOIL
LV_INH_CAM_PAS_IVVT_IN_i	LV_INH_CAM_PAS_IVVT_EX_i	LV_ACT_IND_IVVT_EX_i	
IP_N_32_MAX_DIAG_MEC_IVVT	IP_N_32_MIN_DIAG_MEC_IVVT	C_VB_DIAG_MEC_IVVT	
C_TOIL_MAX_DIAG_MEC_IVVT	C_TOIL_MIN_DIAG_MEC_IVVT	LV_ERR_SLV_IVVT_IN_i	
LV_DE_AD_END_IVVT_IN_i	LV_DE_AD_END_IVVT_EX_i	LV_ERR_SLV_IVVT_EX_i	
CAM_SP_IND_DIAG_IVVT_IN_i	CAM_SP_IND_DIAG_IVVT_EX_i	CAM_AV_IVVT_IN_i	
T_DIF_EDGE_CAM_IVVT_IN_i	T_DIF_EDGE_CAM_IVVT_EX_i	CAM_AV_IVVT_EX_i	
LV_END_DIAG_CAM_STAT_IVVT_IN_i	LV_END_DIAG_CAM_STAT_IVVT_EX_i	CONF_CAM_VVT_EX	
LV_END_DIAG_CAM_DYN_IVVT_IN_i	LV_END_DIAG_CAM_DYN_IVVT_EX_i		
LV_ERR_CAM_STAT_IVVT_IN_i	LV_ERR_CAM_STAT_IVVT_EX_i		
LV_ERR_CAM_DYN_IVVT_IN_i	LV_ERR_CAM_DYN_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

A jammed or a one-side slow-response actuator is detected by means of a comparison of the camshaft position and the setpoint and quantities derived from them.

Description:

In order to avoid wrong error detections by the camshaft position partial deviation diagnosis, the diagnosis is enabled only in certain IVVT operation range.

Camshaft position partial deviation diagnosis at steady setpoint:

There is a strong coherence with the deviation adaptation of the holding pulse width modulation. The deviation adaptation is started after a setpoint change at first. The diagnosis is started after the adaptation. If the camshaft position partial deviation after a setpoint change is bigger than a threshold the adaptation is omitted and the diagnosis is started immediately.

If the setpoint reaches the reference position the deviation adaptation is inactive. The diagnosis has to be disabled for some time till the camshaft reaches the reference position. It is realized by means of CTR_CAM_SP_NEUT_IVVT_IN(_EX)_i.

The principle of the diagnosis is based on two integrators. The first one, CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i, integrates the absolute value of the deviation in time if the deviation is bigger than the threshold C_CAM_HYS_PAS_DIAG_IVVT_IN(_EX). If the deviation is smaller than this threshold the first integrator is set to 0 and the second one, CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i, integrates the threshold C_CAM_HYS_PAS_DIAG_IVVT_IN(_EX) in time. If one of the integrators reaches the diagnosis threshold C_CAM_INT_MAX_PAS_IVVT_IN(_EX) while the other integrator does not reach the diagnosis threshold C_CAM_INT_MIN_PAS_IVVT_IN(_EX), the one-side slow response symptom ERR_SYM_CAM_PAS_IVVT_IN(_EX)_i can be set. If it is the integrator CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i the symptom is set to failure. If it is the integrator CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i the symptom is set inactive.

This diagnosis principle is not sensitive for short stochastical temporary camshaft position partial deviations.

CAM_INT_PAS_SAVE_IVVT_IN(_EX)_i shows the last diagnostic result. It is to compare with C_CAM_INT_MAX_PAS_IVVT_IN(_EX). If the saved value is greater than or equal to the calibration datum it indicates a failure. The opposite case indicates a faultless performance. The saved value is actually zero.

Error-symptoms definition:

Error-symptoms are defined to this diagnostic function as follows:

LV_ERR_CAM_PAS_IVVT_IN(_EX)_i:

```
| | | | |
 │  │  │  └─> (not used)                         (= SYM_0)
 │  │  └────> (not used)                         (= SYM_1)
 │  └───────> (not used)                         (= SYM_2)
 └──────────> CAM_PAS_IVVT_IN(_EX)_i             (= SYM_3)
```

Application conditions:

Initialization:

LV_CDN_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 0

LV_END_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 0
Refer to filtering configuration for the initialization value

LV_ERR_CAM_PAS_IVVT_IN(_EX)_i = 0
Refer to filtering configuration for the initialization value

ERR_SYM_CAM_PAS_IVVT_IN(_EX)_i = NO_SYM
Refer to filtering configuration for the initialization value

At activation:

If LV_ACT_IND_SAVE_PAS_IVVT_IN(_EX)_i = 0

and LV_ACT_IND_IVVT_IN(_EX)_i = 1

then CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i = 0

and CAM_INT_PAS_SAVE_IVVT_IN (_EX)_i = 0

and CAM_INT_PAS_ADC_IVVT_IN(_EX)_i = 0

and CAM_INT_PAS_RTD_IVVT_IN(_EX)_i = 0

and CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i = 0

and CAM_INT_PAS_RTD_SAE_IVVT_IN_i = 0

and CAM_INT_PAS_ADC_SAE_IVVT_EX_i = 0

endif

LV_END_CAM_PAS_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_PAS_0_IVVT_IN(_EX)_i = 0

At deactivation:

LV_CDN_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 0

At failure memory reset:

LV_CDN_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_PAS_IVVT_IN(_EX)_i = 0

CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i = 0

CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i = 0

LV_END_CAM_PAS_0_IVVT_IN(_EX)_i = 0

LV_ERR_CAM_PAS_0_IVVT_IN(_EX)_i = 0

At failure memory reset, in case of checksum error:

CAM_INT_PAS_RTD_SAE_IVVT_IN_i = 0

CAM_INT_PAS_ADC_SAE_IVVT_EX_i = 0

Recurrence:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_INH_CAM_PAS_IVVT_IN(_EX)_i = 0

and LV_ACT_IND_IVVT_IN(_EX)_i = 1

and VB >= C_VB_DIAG_MEC_IVVT

and IP_N_32_MIN_DIAG_MEC_IVVT <= N_32 <= IP_N_32_MAX_DIAG_MEC_IVVT

and C_TOIL_MIN_DIAG_MEC_IVVT <= TOIL <= C_TOIL_MAX_DIAG_MEC_IVVT

and LV_ERR_SLV_IVVT_IN(_EX)_i = 0

and LV_ERR_CAM_STAT_IVVT_IN(_EX)_i = 0

and LV_ERR_CAM_DYN_IVVT_IN(_EX)_i = 0

and *additionally only for Exhaust side (_EX)*

CONF_CAM_VVT_EX = 1

Deactivation:

Not activation

Formula section:

LV_CDN_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 0

if LV_CAM_SP_DYW_IVVT_IN(_EX)_i = 1

and LV_DE_AD_END_IVVT_IN(_EX)_i = 1

then

1) Partial deviation integration at steady setpoint

2) Symptoms calculation at steady setpoint

endif

3) Filtering

4) Mode 6 output

1) Partial deviation integration at steady setpoint:

if Inlet:

CAM_SP_IND_DIAG_IVVT_IN_i <=
C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

or (CAM_SP_IND_DIAG_IVVT_IN_i >
C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN
and CTR_CAM_SP_NEUT_IVVT_IN_i >=
IP_CYC_MIN_PAS_NEUT_IVVT
)

Exhaust:

CAM_SP_IND_DIAG_IVVT_EX_i >=
C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX

or (CAM_SP_IND_DIAG_IVVT_EX_i <
C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX
and CTR_CAM_SP_NEUT_IVVT_EX_i >=
IP_CYC_MIN_PAS_NEUT_IVVT
)

then

if CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_DIAG_IVVT_IN(_EX)_i >
C_CAM_HYS_PAS_DIAG_IVVT_IN(_EX)

then

CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n) =
CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n-1) +
|CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_DIAG_IVVT_IN(_EX)_i| *
T_DIF_EDGE_CAM_IVVT_IN(_EX)_i
CAM_INT_PAS_ADC_IVVT_IN(_EX)_i(n) =
CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n)

elseif CAM_SP_IND_DIAG_IVVT_IN(_EX)_i - CAM_AV_IVVT_IN(_EX)_i >
C_CAM_HYS_PAS_DIAG_IVVT_IN(_EX)

then

CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n) =
CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n-1) +
|CAM_AV_IVVT_IN(_EX)_i - CAM_SP_IND_DIAG_IVVT_IN(_EX)_i| *
T_DIF_EDGE_CAM_IVVT_IN(_EX)_i
CAM_INT_PAS_RTD_IVVT_IN(_EX)_i(n) =

CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i(n)

else

CAM_INT_PAS_DIAG_IVVT_IN(_EX)_i = 0

CAM_INT_PAS_ADC_IVVT_IN(_EX)_i = 0

CAM_INT_PAS_RTD_IVVT_IN(_EX)_i = 0

If Inlet :

CAM_SP_IND_DIAG_IVVT_IN_i <=

C_CAM_INI_IN - C_CAM_SP_HYS_NEUT_IVVT_IN

Exhaust :

CAM_SP_IND_DIAG_IVVT_EX_i >=

C_CAM_INI_EX + C_CAM_SP_HYS_NEUT_IVVT_EX

then

CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i(n) =

CAM_INT_PAS_DIAG_RST_IVVT_IN(_EX)_i(n-1) +

C_CAM_HYS_PAS_DIAG_IVVT_IN(_EX) *

T_DIF_EDGE_CAM_IVVT_IN(_EX)_i

endif

endif

endif

2) Symptoms calculation at steady setpoint:

SYM_3:

Inlet side partial deviation (one-side slow response) detection :

if CAM_INT_PAS_ADC_IVVT_IN_i <= C_CAM_INT_MIN_PAS_IVVT_IN

and CAM_INT_PAS_RTD_IVVT_IN_i >= C_CAM_INT_MAX_PAS_IVVT_IN

then

LV_CDN_DIAG_CAM_PAS_IVVT_IN_i = 1

ERR_SYM_CAM_PAS_IVVT_IN_i = "SYM_3" active

CAM_INT_PAS_SAVE_IVVT_IN_i = CAM_INT_PAS_RTD_IVVT_IN_i

CAM_INT_PAS_DIAG_IVVT_IN_i = 0

CAM_INT_PAS_ADC_IVVT_IN_i = 0

CAM_INT_PAS_RTD_IVVT_IN_i = 0

CAM_INT_PAS_DIAG_RST_IVVT_IN_i = 0

else if No failure detection :

CAM_INT_PAS_DIAG_RST_IVVT_IN_i >= C_CAM_INT_MAX_PAS_IVVT_IN

then

LV_CDN_DIAG_CAM_PAS_IVVT_IN_i = 1

ERR_SYM_CAM_PAS_IVVT_IN_i = "NO_SYM" active

CAM_INT_PAS_DIAG_IVVT_IN_i = 0

CAM_INT_PAS_DIAG_RST_IVVT_IN_i = 0

endif

endif

Exhaust side partial deviation (one-side slow response) detection :

if CAM_INT_PAS_RTD_IVVT_EX_i <= C_CAM_INT_MIN_PAS_IVVT_EX

and CAM_INT_PAS_ADC_IVVT_EX_i >= C_CAM_INT_MAX_PAS_IVVT_EX

then

LV_CDN_DIAG_CAM_PAS_IVVT_EX_i = 1

ERR_SYM_CAM_PAS_IVVT_EX_i = "SYM_3" active

CAM_INT_PAS_SAVE_IVVT_EX_i = CAM_INT_PAS_ADC_IVVT_EX_i

CAM_INT_PAS_DIAG_IVVT_EX_i = 0

CAM_INT_PAS_ADC_IVVT_EX_i = 0

CAM_INT_PAS_RTD_IVVT_EX_i = 0

CAM_INT_PAS_DIAG_RST_IVVT_EX_i = 0

else **if** No failure detection :

CAM_INT_PAS_DIAG_RST_IVVT_EX_i >= C_CAM_INT_MAX_PAS_IVVT_EX

then

LV_CDN_DIAG_CAM_PAS_IVVT_EX_i = 1

ERR_SYM_CAM_PAS_IVVT_EX_i = "NO_SYM" active

CAM_INT_PAS_DIAG_IVVT_EX_i = 0

CAM_INT_PAS_DIAG_RST_IVVT_EX_i = 0

endif

endif

3) Filtering:

Apply filter on current symptoms

if filtering result available (after debounce)

then

LV_ERR_CAM_PAS_IVVT_IN(_EX)_i = filtering result

LV_END_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 1

Deliver the result to Error Management

endif

Calculate at every recurrence independent on activation conditions:

LV_ACT_IND_SAVE_PAS_IVVT_IN(_EX)_i = LV_ACT_IND_IVVT_IN(_EX)_i

4) Mode 6 output

If (LV_END_CAM_PAS_0_IVVT_IN_i = 0

and LV_END_DIAG_CAM_PAS_IVVT_IN_i = 1)

or (LV_ERR_CAM_PAS_0_IVVT_IN_i = 0

and LV_ERR_CAM_PAS_IVVT_IN_i = 1)
then CAM_INT_PAS_RTD_SAE_IVVT_IN_i = CAM_INT_PAS_SAVE_IVVT_IN_i
LV_END_CAM_PAS_0_IVVT_IN_i =
LV_END_DIAG_CAM_PAS_IVVT_IN_i
LV_ERR_CAM_PAS_0_IVVT_IN_i =
LV_ERR_CAM_PAS_IVVT_IN_i
endif

If (LV_END_CAM_PAS_0_IVVT_EX_i = 0
and LV_END_DIAG_CAM_PAS_IVVT_EX_i = 1)
or (LV_ERR_CAM_PAS_0_IVVT_EX_i = 0
and LV_ERR_CAM_PAS_IVVT_EX_i = 1)
then CAM_INT_PAS_ADC_SAE_IVVT_EX_i = CAM_INT_PAS_SAVE_IVVT_EX_i
LV_END_CAM_PAS_0_IVVT_EX_i =
LV_END_DIAG_CAM_PAS_IVVT_EX_i
LV_ERR_CAM_PAS_0_IVVT_EX_i=
LV_ERR_CAM_PAS_IVVT_EX_i
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_PAS_DIAG_IVVT_IN	1	0...FFH	0...95.625	0.375	°CRK
Hysteresis for camshaft position partial deviation diagnosis; inlet					
C_CAM_HYS_PAS_DIAG_IVVT_EX	1	0...FFH	0...95.625	0.375	°CRK
Hysteresis for camshaft position partial deviation diagnosis; exhaust					
C_CAM_INT_MIN_PAS_IVVT_IN	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position partial deviation diagnosis; inlet					
C_CAM_INT_MIN_PAS_IVVT_EX	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position partial deviation diagnosis; exhaust					
C_CAM_INT_MAX_PAS_IVVT_IN	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position partial deviation diagnosis; inlet					
C_CAM_INT_MAX_PAS_IVVT_EX	1	0...FFFFH	0...201.3235	0.003072	°CRK*s
Integrator limit for camshaft position partial deviation diagnosis; exhaust					
IP_CYC_MIN_PAS_NEUT_IVVT	4x4	0...1770H	0...6000	1	-
LDP_N_32_IP_CYC_MIN_DE_AD_IVVT	4	0...FFH	0...8160	32	Rpm
LDPM_TOIL_2_VVTI	4	0...C8H	-40...160	1	°C
Min. number of edges to start diagnosis after setpoint reached reference position					

A.52 IVVT Crankshaft to Camshaft Mechanics Violation Diagnosis - Application Incidences

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_INH_MEC_IVVT_IN_i	O/V	0...1H	0...1	1	-
Specific conditions for diagnosis crankshaft to camshaft mechanics violation: 0 = not fulfilled, 1 = fulfilled; inlet					
LV_INH_MEC_IVVT_EX_i	O/V	0...1H	0...1	1	-
Specific conditions for diagnosis crankshaft to camshaft mechanics violation: 0 = not fulfilled, 1 = fulfilled; exhaust					

Input data:

LV_ERR_CRK	LV_ERR_CAM_IN	LV_CDN_VB_OBD1	LV_IGK
LV_ERR_SLV_IVVT_IN_i	LV_ERR_SLV_IVVT_EX_i	LV_ERR_CAM_EX	

FUNCTION DESCRIPTION:

Description:

There are no specific conditions regarding the crankshaft to camshaft mechanics violation diagnosis.

Application conditions:

Recurrence: *100 ms*

Activation: *At every engine state*

Formula section:

If LV_ERR_CRK = 1 **or**
LV_ERR_CAM_IN(_EX) = 1 **or**
LV_IGK=0 **or**
LV_CDN_VB_OBD1=0 **or**
LV_ERR_SLV_IVVT_IN(_EX)_i = 1
Then LV_INH_MEC_IVVT_IN(_EX)_i = 1
Else LV_INH_MEC_IVVT_IN(_EX)_i = 0
Endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
MEC_IVVT_IN_1		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	mech IVVT error (Inlet)	SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_IN
TOIL
STATE_IVVT
CAM_AV_IVVT_IN_1

Diagnostic	Symptom Description	Symptom	Filter type
MEC_IVVT_EX_1		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	mech IVVT error (Exhaust)	SYM_3	

List of Environmental Data to store in Failure Memory: PSN_AD_CAM_EX
TOIL
STATE_IVVT
CAM_AV_IVVT_EX_1

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_ABC_INC_MEC_IVVT_IN	1	0...FFH	0...255	1	-
Anti-bouce counter increment for diagnosis crankshaft to camshaft mechanics violation (Inlet)					
C_ABC_MAX_MEC_IVVT_IN	1	1...FFH	1...255	1	-
Anti-bouce counter maximum value for diagnosis crankshaft to camshaft mechanics violation (Inlet)					
C_ABC_INC_MEC_IVVT_EX	1	0...FFH	0...255	1	-
Anti-bouce counter increment for diagnosis crankshaft to camshaft mechanics violation (Exhaust)					
C_ABC_MAX_MEC_IVVT_EX	1	1...FFH	1...255	1	-
Anti-bouce counter maximum value for diagnosis crankshaft to camshaft mechanics violation (Exhaust)					

A.53 IVVT Camshaft Position Deviation Diagnosis - Application Incidences

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_INH_CAM_STAT_IVVT_IN_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring inlet camshaft deviation diagnosis at steady setpoint					
LV_INH_CAM_STAT_IVVT_EX_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring exhaust camshaft deviation diagnosis at steady setpoint					
LV_INH_CAM_DYN_IVVT_IN_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring inlet camshaft deviation diagnosis at changing setpoint					
LV_INH_CAM_DYN_IVVT_EX_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring exhaust camshaft deviation diagnosis at changing setpoint					
LV_CAM_SP_CHG_IN	V	0...1H	0..1	1	-
Specific conditions on CAM_SP changes to allow CAM_DE diagnosis (Inlet)					
LV_CAM_SP_CHG_EX	V	0...1H	0..1	1	-
Specific conditions on CAM_SP changes to allow CAM_DE diagnosis (Exhaust)					
LV_CAM_SP_CHG_1_IN	-	0...1H	0..1	1	-
CAM_SP out of end position (Inlet)					
LV_CAM_SP_CHG_1_EX	-	0...1H	0..1	1	-
CAM_SP out of end position (Exhaust)					
CTR_CAM_SP_CHG_IN	V	0...FFH	0..255	1	-
Counter of CAM_SP 'changes' (Inlet)					
CTR_CAM_SP_CHG_EX	V	0...FFH	0..255	1	-
Counter of CAM_SP 'changes' (Exhaust)					
LV_INH_CAM_PAS_IVVT_IN_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring inlet camshaft passive diagnosis at changing setpoint					
LV_INH_CAM_PAS_IVVT_EX_i	O/V	0...1H	0...1	1	-
Inhibition condition for enduring exhaust camshaft passive diagnosis at changing setpoint					

Input data:

LV_ERR_TOIL	LV_CDN_VB_OBD1	LV_END_PSN_ACT_DIAG_IVVT_IN_1
LV_ERR_LIH_IVVT	LV_TRIP_ACT_IVVT_IN	LV_END_PSN_ACT_DIAG_IVVT_EX_1
LV_ADJ_MAX_IVVT_IN_i	CAM_SP_IVVT_IN	C_CAM_INI_IN
LV_ACT_IVVT	LV_TRIP_ACT_IVVT_EX	LV_ADJ_MAX_IVVT_EX_i
LV_ERR_TOIL_PLAUS_H	LV_ERR_TOIL_STUCK	LV_ERR_TOIL_PLAUS_L

FUNCTION DESCRIPTION:

Description:

The camshaft position deviation diagnosis is inhibited if the camshaft is close to the maximum adjustment position, or the short trip is active or the oil temperature cannot be determined reliable.

Application conditions:

Initialisation: (at reset)

LV_INH_CAM_STAT_IVVT_IN(_EX)_1 = 1

LV_INH_CAM_DYN_IVVT_IN(_EX)_1 = 1

LV_INH_CAM_PAS_IVVT_IN(_EX)_i = 1

LV_CAM_SP_CHG_IN(_EX) = 0

LV_CAM_SP_CHG_1_IN(_EX) = 0
CTR_CAM_SP_CHG_IN(_EX) = 0

Recurrence: At each used camshaft edge, see "IVVT Scheduler"

Activation: *LV_ACT_IVVT = 1*

Formula section:

Definition of LV_INH_CAM_STAT_IVVT_IN(_EX)_1 &
LV_INH_CAM_DYN_IVVT_IN(_EX)_1 &
LV_INH_CAM_PAS_IVVT_IN(_EX)_1:

if LV_ADJ_MAX_IVVT_IN(_EX)_i = 0
and LV_TRIP_ACT_IVVT_IN(_EX) = 0
and LV_ERR_TOIL = 0
and LV_ERR_LIH_IVVT = 0
and LV_ERR_TOIL_STUCK = 0
and LV_ERR_TOIL_PLAUS_L = 0
and LV_ERR_TOIL_PLAUS_H = 0
and LV_CAM_SP_CHG_IN(_EX) = 1
and LV_END_PSN_ACT_DIAG_IVVT_IN(_EX)_1 = 0
and LV_CDN_VB_OBD1 = 1
then LV_INH_CAM_STAT_IVVT_IN(_EX)_i = 0
LV_INH_CAM_DYN_IVVT_IN(_EX)_i = 0
LV_INH_CAM_PAS_IVVT_IN(_EX)_i = 0
else LV_INH_CAM_STAT_IVVT_IN(_EX)_i = 1
LV_INH_CAM_DYN_IVVT_IN(_EX)_i = 1
LV_INH_CAM_PAS_IVVT_IN(_EX)_i = 1
endif

Definition of LV_CAM_SP_CHG_IN(_EX):

if CTR_CAM_SP_CHG_IN (_EX) < C_CTR_MAX_CAM_SP_CHG
then *waiting for enough CAM_SP 'changes'...*
if *For Intake:*
CAM_SP_IVVT_IN < C_CAM_INI_IN - C_CAM_HYS_AD_REF_IVVT
For Exhaust:
CAM_SP_IVVT_EX < C_CAM_INI_EX + C_CAM_HYS_AD_REF_IVVT
then *CAM_SP was set out of end position*
LV_CAM_SP_CHG_1_IN(_EX) = 1
else if LV_CAM_SP_CHG_1_IN(_EX) = 1
then *CAM_SP returns to end position*

CTR_CAM_SP_CHG_IN (_EX) = CTR_CAM_SP_CHG_IN (_EX) + 1

LV_CAM_SP_CHG_1_IN (_EX) = 0

endif

endif

else CAM deviation diagnosis can be enabled
LV_CAM_SP_CHG_IN(_EX) = 1

endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Enduring camshaft position dynamic deviation inlet 1 CAM_DYN_IVVT_IN_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Dynamic CAM deviation (Inlet)	SYM_3	
Enduring camshaft position dynamic deviation inlet 1 CAM_STAT_IVVT_IN_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Static CAM deviation (Inlet)	SYM_3	
Enduring camshaft position dynamic deviation inlet 1 CAM_PAS_IVVT_IN_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Passive CAM deviation (Inlet)	SYM_3	

List of Environmental Data to store in Failure Memory: DIAG_INST_ENVD
CAM_SP_IVVT_IN_1
CAM_IN_1
CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Enduring camshaft position dynamic deviation exhaust 1 CAM_DYN_IVVT_EX_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Dynamic CAM deviation (Exhaust)	SYM_3	
Enduring camshaft position dynamic deviation exhaust 1 CAM_STAT_IVVT_EX_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Static CAM deviation (Exhaust)	SYM_3	
Enduring camshaft position dynamic deviation exhaust 1 CAM_PAS_IVVT_EX_1		SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
	Passive CAM deviation (Exhaust)	SYM_3	

List of Environmental Data to store in Failure Memory: DIAG_INST_ENVD
CAM_SP_IVVT_EX_1
CAM_EX_1
CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_ABC_INC_CAM_STAT_IVVT_IN	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_STAT_IVVT_IN	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_ABC_INC_CAM_DYN_IVVT_IN	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_DYN_IVVT_IN	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_ABC_INC_CAM_STAT_IVVT_EX	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_STAT_IVVT_EX	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_ABC_INC_CAM_DYN_IVVT_EX	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_DYN_IVVT_EX	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_CAM_HYS_AD_REF_IVVT	1	0...A0H	0...60	0.375	°CRK
Hysteresis for setpoint around reference position for enabling reference position adaptation					
C_CTR_MAX_CAM_SP_CHG	1	0...FFH	0...255	1	-
Number of CAM_SP 'changes' before CAM deviations diagnosis is allowed					
C_ABC_INC_CAM_PAS_IVVT_IN	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_PAS_IVVT_IN	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_ABC_INC_CAM_PAS_IVVT_EX	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_CAM_PAS_IVVT_EX	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					

A.54 IVVT Camshaft Position Deviation Diagnosis - Rate Based Monitoring

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
STATE_RBM_CAM_STAT_IVVT_IN_i	O/V/S	0...FFH	0...255	1	-
Interface of monitor CAM_STAT_IVVT_IN_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
STATE_RBM_CAM_STAT_IVVT_EX_i	O/V/S	0...FFH	0...255	1	-
Interface of monitor CAM_STAT_IVVT_EX_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
STATE_RBM_CAM_DYN_IVVT_IN_i	O/V/S	0...FFH	0...255	1	-
Interface of monitor CAM_DYN_IVVT_IN_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
STATE_RBM_CAM_DYN_IVVT_EX_i	O/V/S	0...FFH	0...255	1	-
Interface of monitor CAM_DYN_IVVT_EX_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
CTR_RBM_2_CAM_DE_IVVT_IN_i	V	0...FFH	0...255	1	-
Counter for RBM 2 times activation condition, overflow not allowed; inlet					
CTR_RBM_2_CAM_DE_IVVT_EX_i	V	0...FFH	0...255	1	-
Counter for RBM 2 times activation condition, overflow not allowed; exhaust					
LV_T_RBM_10_CAM_DE_IVVT_IN_i	V	0...1H	0...1	1	-
Flag for RBM 10 s activation condition; inlet					
LV_T_RBM_10_CAM_DE_IVVT_EX_i	V	0...1H	0...1	1	-
Flag for RBM 10 s activation condition; exhaust					
T_RBM_10_CAM_DE_IVVT_IN_i	V	0...FFH	0...25.5	0.1	s
Timer for RBM 10 s activation condition, overflow not allowed; inlet					
T_RBM_10_CAM_DE_IVVT_EX_i	V	0...FFH	0...25.5	0.1	s
Timer for RBM 10 s activation condition, overflow not allowed; exhaust					
RATE_RBM_CAM_STAT_IVVT_IN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for CAM_STAT_IVVT_IN monitoring , for application purpose					
RATE_RBM_CAM_DYN_IVVT_IN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for CAM_DYN_IVVT_IN , for application purpose					
STATE_RBM_CAM_PAS_IVVT_IN_i	O/V/S	0...FFH	0...255	1	-
Interface of monitor CAM_PAS_IVVT_IN_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
RATE_RBM_CAM_PAS_IVVT_IN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for CAM_PAS_IVVT_IN monitoring , for application purpose					
STATE_RBM_CAM_PAS_IVVT_EX_i	O/V/S	0...FFH	0...255	1	-

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
Interface of monitor CAM_PAS_IVVT_EX_i for the rate based monitoring statistics Bit 0: conditions for montoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1) Bit 7 (optional): the check of the pending status of the failure(s) (at the next LV_DC 0→1 transition), for bit 1 calculation, is required (bit 7 = 1)					
RATE_RBM_CAM_PAS_IVVT_EX_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for CAM_PAS_IVVT_EX monitoring , for application purpose					

Input data:

LV_END_DIAG_CAM_STAT_IVVT_IN_i	LV_END_DIAG_CAM_DYN_IVVT_IN_i	LV_ERR_MAP
LV_END_DIAG_CAM_STAT_IVVT_EX_i	LV_END_DIAG_CAM_DYN_IVVT_EX_i	LV_ERR_TPS
LV_END_DIAG_CAM_PAS_IVVT_IN_i	LV_END_DIAG_CAM_PAS_IVVT_EX_i	
LV_ERR_SLV_IVVT_IN_i	CAM_SP_IND_DIAG_IVVT_IN_i	LV_ERR_MAF
LV_ERR_SLV_IVVT_EX_i	CAM_SP_IND_DIAG_IVVT_EX_i	CONF_CAM_VVT_EX
LV_TRIP_ACT_IVVT_IN	LV_TRIP_ACT_IVVT_EX	LV_DC
CTR_CDN_RBM_CAM_STAT_IVVT_IN_1	CTR_COMP_RBM_CAM_STAT_IVVT_IN_1	LV_ERR_TCO
CTR_CDN_RBM_CAM_STAT_IVVT_EX_1	CTR_COMP_RBM_CAM_STAT_IVVT_EX_1	
CTR_CDN_RBM_CAM_DYN_IVVT_IN_1	CTR_COMP_RBM_CAM_DYN_IVVT_IN_1	CONF_TOIL_MDL
CTR_CDN_RBM_CAM_DYN_IVVT_EX_1	CTR_COMP_RBM_CAM_DYN_IVVT_EX_1	
CTR_CDN_RBM_CAM_PAS_IVVT_IN_1	CTR_COMP_RBM_CAM_PAS_IVVT_IN_1	
CTR_CDN_RBM_CAM_PAS_IVVT_EX_1	CTR_COMP_RBM_CAM_PAS_IVVT_EX_1	

FUNCTION DESCRIPTION:

General information:

The interface between the monitor (non-continuous camshaft deviation diagnosis CAM_DE_IVVT_IN(_EX)_i) and the Rate-Based Monitoring statistics is defined with STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i and STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i

Description:

The Rate-Based Monitoring statistics for the IVVT function specifies additional conditions which have to be fulfilled for having the specified driving cycle. The function has to be activited twice at least or it is activated for 10 seconds at least. Further, no RBM statistics is updated if the diagnosis is done during the IVVT short trip LV_TRIP_ACT_IVVT_IN(_EX) = 1 (external triggering possible, e.g. as a garage test).

These two conditions are realized as follows. The setpoint has to leave the reference position and cross the threshold C_CAM_SP_THD_RBM_2_IVVT_IN(_EX) twice at least or it has to be over C_CAM_SP_THD_RBM_10_IVVT_IN(_EX) for 10 seconds at least.

Application conditions:

Initialization:

At reset:

STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0

STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0

STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0

At failure memory reset:

Bit 1 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0

Bit 1 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0

Bit 1 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0

Recurrence: 100 ms

Activation: LV_DC = 1

Formula section:

At LV_DC = 0 → 1 transition: (for stationary and dynamic)

if LV_ERR_SLV_IVVT_IN(_EX)_i = 1
or LV_ERR_MAP = 1
or LV_ERR_MAF = 1
or LV_ERR_TPS = 1
or (CONF_TOIL_MDL =! 0 **and** LV_ERR_TCO = 1)
or *only for Exhaust side (_EX)*
CONF_CAM_VVT_EX = 0

then Bit 1 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 1
Bit 1 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 1
Bit 1 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 1

Endif

Bit 0 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0

Bit 0 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0

Bit 0 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0

Bit 2 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0

Bit 2 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0

Bit 2 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0

CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i = 0

LV_T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 0

T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 0

End of initialization at LV_DC = 0 → 1 transition

Bit 0: (for stationary)

if Bit 0 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0
then
  if LV_END_DIAG_CAM_STAT_IVVT_IN(_EX)_i = 1
    and LV_TRIP_ACT_IVVT_IN(_EX) = 0
  then Bit 0 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 1
  endif
endif

Bit 0: (for dynamic)

if Bit 0 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0
then
  if LV_END_DIAG_CAM_DYN_IVVT_IN(_EX)_i = 1
    and LV_TRIP_ACT_IVVT_IN(_EX) = 0
  then Bit 0 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 1
  endif
endif

Bit 0: (for partial)

if Bit 0 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0
then
  if LV_END_DIAG_CAM_PAS_IVVT_IN(_EX)_i = 1
    and LV_TRIP_ACT_IVVT_IN(_EX) = 0
  then Bit 0 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 1
  endif
endif

Bit 1: (for stationary)

if Bit 1 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0
then
  if LV_ERR_SLV_IVVT_IN(_EX)_i = 1
    or LV_ERR_MAP = 1
    or LV_ERR_MAF = 1
    or LV_ERR_TPS = 1
    or (CONF_TOIL_MDL =! 0 **and** LV_ERR_TCO = 1)
  then Bit 1 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 1
  endif

```
        endif

    Bit 1: (for dynamic)
        if      Bit 1 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0
        then if             LV_ERR_SLV_IVVT_IN(_EX)_i = 1
                    or      LV_ERR_MAP = 1
                    or      LV_ERR_MAF = 1
                    or      LV_ERR_TPS = 1
                    or      (CONF_TOIL_MDL =! 0 and LV_ERR_TCO = 1)
                then        Bit 1 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 1
                endif
        endif

    Bit 1: (for partial)
        if      Bit 1 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0
        then    if          LV_ERR_SLV_IVVT_IN(_EX)_i = 1
                    or      LV_ERR_MAP = 1
                    or      LV_ERR_MAF = 1
                    or      LV_ERR_TPS = 1
                    or      (CONF_TOIL_MDL =! 0 and LV_ERR_TCO = 1)
                then        Bit 1 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 1
                endif
    endif

    Bit 2:
        if          [       CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i = 1
                    and     CAM_SP_IND_DIAG_IVVT_IN(_EX)_i(n) = C_CAM_INI_IN(_EX)]
                or  [       (       CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i = 0
                            or      CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i = 2)
                    Inlet:
                    and     CAM_SP_IND_DIAG_IVVT_IN_i(n-1) >
                            C_CAM_SP_THD_RBM_2_IVVT_IN
                    and     CAM_SP_IND_DIAG_IVVT_IN_i(n) <=
                            C_CAM_SP_THD_RBM_2_IVVT_IN


                    Exhaust:
                    and     CAM_SP_IND_DIAG_IVVT_EX_i(n-1) <
```

C_CAM_SP_THD_RBM_2_IVVT_EX
and CAM_SP_IND_DIAG_IVVT_EX_i(n) >=
C_CAM_SP_THD_RBM_2_IVVT_EX

then CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i =
CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i + 1

endif

if Inlet:
CAM_SP_IND_DIAG_IVVT_IN_i(n) <=
C_CAM_SP_THD_RBM_10_IVVT_IN
Exhaust:
CAM_SP_IND_DIAG_IVVT_EX_i(n) >=
C_CAM_SP_THD_RBM_10_IVVT_EX

then

if LV_T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 0

then LV_T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 1
T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 0

else T_RBM_10_CAM_DE_IVVT_IN(_EX)_i =
T_RBM_10_CAM_DE_IVVT_IN(_EX)_i + 0.1 s

endif

else LV_T_RBM_10_CAM_DE_IVVT_IN(_EX)_i = 0

endif

(for stationary)

if Bit 2 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 0

then

if CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i >= 3
or T_RBM_10_CAM_DE_IVVT_IN(_EX)_i >= 10

then Bit 2 of STATE_RBM_CAM_STAT_IVVT_IN(_EX)_i = 1

endif

endif

(for dynamic)

if Bit 2 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 0

then

if CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i >= 3
or T_RBM_10_CAM_DE_IVVT_IN(_EX)_i >= 10

then Bit 2 of STATE_RBM_CAM_DYN_IVVT_IN(_EX)_i = 1
endif
endif

<u>(for partial)</u>
if Bit 2 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 0
then **if** CTR_RBM_2_CAM_DE_IVVT_IN(_EX)_i >= 3
or T_RBM_10_CAM_DE_IVVT_IN(_EX)_i >= 10
then Bit 2 of STATE_RBM_CAM_PAS_IVVT_IN(_EX)_i = 1
endif
endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_CAM_STAT_IVVT_IN_1 = 0
then RATE_RBM_CAM_STAT_IVVT_IN_1 = FFFFH
else RATE_RBM_CAM_STAT_IVVT_IN_1
= (CTR_COMP_RBM_CAM_STAT_IVVT_IN_1 / CTR_CDN_RBM_CAM_STAT_IVVT_IN_1)
endif

If CTR_CDN_RBM_CAM_DYN_IVVT_IN_1 = 0
then RATE_RBM_CAM_DYN_IVVT_IN_1 = FFFFH
else RATE_RBM_CAM_DYN_IVVT_IN_1
= (CTR_COMP_RBM_CAM_DYN_IVVT_IN_1 / CTR_CDN_RBM_CAM_DYN_IVVT_IN_1)
endif

If CTR_CDN_RBM_CAM_PAS_IVVT_IN_1 = 0
then RATE_RBM_CAM_PAS_IVVT_IN_1 = FFFFH
else RATE_RBM_CAM_PAS_IVVT_IN_1
= (CTR_COMP_RBM_CAM_PAS_IVVT_IN_1 / CTR_CDN_RBM_CAM_PAS_IVVT_IN_1)
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_SP_THD_RBM_2_IVVT_IN	1	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint threshold for RBM 2 times activation condition; inlet					
C_CAM_SP_THD_RBM_2_IVVT_EX	1	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint threshold for RBM 2 times activation condition; exhaust					
C_CAM_SP_THD_RBM_10_IVVT_IN	1	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint threshold for RBM 10 s activation condition; inlet					
C_CAM_SP_THD_RBM_10_IVVT_EX	1	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint threshold for RBM 10 s activation condition; exhaust					

A.55 IVVT Output Signal Diagnosis

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_SLV_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis IVVT inlet solenoid valve					
LV_ERR_SLV_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure: failure after filtering of diagnosis IVVT exhaust solenoid valve					
CDN_DIAG_SLV_IVVT_IN_i	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of IVVT solenoid valve; inlet bit 0: diagnosis condition for symptom SYM_0 bit 1: diagnosis condition for symptom SYM_1 bit 2: diagnosis condition for symptom SYM_2					
CDN_DIAG_SLV_IVVT_EX_i	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of IVVT solenoid valve; exhaust bit 0: diagnosis condition for symptom SYM_0 bit 1: diagnosis condition for symptom SYM_1 bit 2: diagnosis condition for symptom SYM_2					
ERR_DIAG_SLV_IVVT_IN_i	-	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Raw value of error symptom for IVVT solenoid valve (only parameter); inlet					
ERR_DIAG_SLV_IVVT_EX_i	-	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Raw value of error symptom for IVVT solenoid valve (only parameter); exhaust					

Input data:

LV_INH_DIAG_SLV_IVVT_IN_i	LV_INH_DIAG_SLV_IVVT_EX_i	LV_CDN_VB_OBD1	LV_IVVTPWM
CONF_CAM_VVT_EX			

Import actions:

ACTION_ERRM_FilterMulticondition(IN < xx >, IN < Cdn_Diag_xx >, IN < Err_Diag_xx >, IN < C_Abc_Inc_xx >, IN < C_Abc_Max_xx >, OUT < Lv_Err_xx >)
This action computes the elementary antibounce filter for one failure treatment and returns filter result
ACTION_INFR_GetElDiagSlvVvt(IN < Vvt > OUT < Cdn_Diag >, OUT < Err_Diag >)
This action reads the failure and condition information for a symptom of the SLV VVT power stage of camshaft Vvt

FUNCTION DESCRIPTION:

General information:

The IVVT solenoid valve is driven by the ECU via an output driver. The failure detection is done by ECU Hardware.

The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

Description:

The diagnosis of all failures (SCP, SCG and OC) of the IVVT solenoid power stage is possible if the energisation is not too close to 0 % and to 100 %. The minimum and the maximum duty cycle necessary for the complete diagnosis results from the hardware properties (solenoid resistance and inductivity) and frequency used. The duty cycles used by the IVVT control algorithm (calibration data) have to be within this range. It is possible if the adjustment velocity at PWM little bit greater than the minimum diagnosis PWM equals approximately the velocity at 0 %. Similarly, if the adjustment velocity at PWM little bit smaller than the maximum diagnosis PWM equals approximately the velocity at 100 %.

As described in the control algorithm the holding duty cycle (~40-60 %) can be replaced by a time limited ~10 % or ~90 % energisation. If this energisation is short (never longer than 5 ms - recommended maximum calibration value) then 0 % is used instead of ~10 % and 100 % instead of ~90 %. It means that the deviation of the camshaft position from the setpoint is small. This can occur on every used camshaft edge. The recommended calibration is that the time between two used edges is 10 ms at least. Also, there is 5 ms of the holding duty cycle between the 0 % and 100 % energisations. There is at least one "on" and one "off" period at the frequency 200 Hz and higher (300-400 Hz is usually used for holding). This fits to the recurrence of reading of the diagnostic information every 10 ms even if the 10 ms diagnostic recurrence and the camshaft edges are asynchronous. The detection of a failure can be delayed by ~10 ms, but this is negligible.

An external solenoid valve energisation is possible, i.e., independent on the IVVT control algorithm. Then, the external source must fully manage its impacts on the output signal diagnosis, e.g. inhibition by LV_INH_DIAG_SLV_IVVT_IN(_EX)_i.

After conditions for the diagnosis activation are met the diagnosis is delayed for NC_DLY_PSD_SLV_IVVT recurences in order to avoid usage of wrong infrastructure information.

This diagnosis is called for each configured camshaft, see "IVVT Scheduler".

Application conditions:

Initialisation:

LV_ERR_SLV_IVVT_IN(_EX)_i are initialised according to Filter-type

At reset:

CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0

Recurrence: 100 ms

Activation: Inlet: *At every engine state*

Exhaust: CONF_CAM_VVT_EX = 1

Formula section:

xx = SLV_IVVT_IN(_EX)_i

yy = VVT_IN(_EX)_i

Usage of the diagnosis information (failure symptoms (raw value) ERR_DIAG_SLV_IVVT_IN(_EX)_i and basic diagnosis conditions CDN_DIAG_SLV_IVVT_IN(_EX)_i) received from the infrastructure:

```
if      LV_IVVTPWM = 1
    and LV_INH_DIAG_SLV_IVVT_IN(_EX)_i = 0
    and LV_CDN_VB_OBD1 = 1
then    Diagnosis active
        ACTION_INFR_GetElDiagSlvVvt(
        IN               yy
        OUT              CDN_DIAG_xx,
        OUT              ERR_DIAG_xx,
        SYNCRONIZATION   CALL)
        Basic diagnosis conditions are set according to infrastructure information:
        CDN_DIAG_SLV_IVVT_IN(_EX)_i. Failure symptoms (raw value) are set
        according to infrastructure information: ERR_DIAG_SLV_IVVT_IN(_EX)_i
        if    "This query was reached for NC_DLY_PSD_SLV_IVVT diagnosis
              recurrences at least"
        then  Additional diagnosis conditions
              if    0
              then  Conditions that disable SYM_0 detection
                    bit 0 of CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0
                    { Diagnosis of SYM_0 is not possible }
              endif
              if    0
              then  Conditions that disable SYM_1 detection
                    bit 1 of CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0
                    { Diagnosis of SYM_1 is not possible }
              endif
              if    0
              then  Conditions that disable SYM_2 detection
                    bit 2 of CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0
                    { Diagnosis of SYM_2 is not possible }
              endif
        else  CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0
        endif
else    CDN_DIAG_SLV_IVVT_IN(_EX)_i = 0
endif
```

Failure filtering and error management treatment:

For failure and error management treatment the anti-bounce mechanism is called with the parameters CDN_DIAG_SLV_IVVT_IN(_EX)_i and ERR_DIAG_SLV_IVVT_IN(_EX)_i.

```
ACTION_ERRM_FilterMulticondition(
IN               xx,
IN               CDN_DIAG_xx,
IN               ERR_DIAG_xx,
IN               C_ABC_INC_xx,
IN               C_ABC_MAX_xx,
OUT              LV_ERR_xx,
SYNCRONIZATION   CALL)
```

This algorithm determines:

ERR_SYM_SLV_IVVT_IN(_EX)_i	(detected error symptom for SLV_IVVT_IN(_EX)_i diagnosis)
LV_ERR_SLV_IVVT_IN(_EX)_i	(Error flag for debounced error of SLV_IVVT_IN(_EX)_i)
LV_CDN_DIAG_SLV_IVVT_IN(_EX)_i	(Diagnosis condition information)
LV_END_DIAG_SLV_IVVT_IN(_EX)_i	(End of diagnosis information)

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_ABC_INC_SLV_IVVT_IN	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_SLV_IVVT_IN	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_ABC_INC_SLV_IVVT_EX	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_SLV_IVVT_EX	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					

Configuration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
NC_DLY_PSD_SLV_IVVT	1	1...FH	1...15	1	-
Number of recurrences with diagnosis deactivation after diagnosis activation conditions are met (typical 2)					

A.56 IVVT Output Signal Diagnosis - Application Incidences

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_INH_DIAG_SLV_IVVT_IN_i	O/V	0...1H	0...1	1	-
Specific conditions for solenoid valve diagnosis: 0 = not fulfilled, 1 = fulfilled; inlet					
LV_INH_DIAG_SLV_IVVT_EX_i	O/V	0...1H	0...1	1	-
Specific conditions for solenoid valve diagnosis: 0 = not fulfilled, 1 = fulfilled; exhaust					

FUNCTION DESCRIPTION:

Description:

There are no specific conditions regarding the output signal diagnosis.

Application conditions:

Recurrence: *100 ms*

Activation: *At every engine state*

Formula section:

LV_INH_DIAG_SLV_IVVT_IN(_EX)_i = 0

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
IVVT solenoid valve; inlet 1	SCB	SYM_0	*MPL_STD_INI*
	SCG	SYM_1	
	OL	SYM_2	
SLV_IVVT_IN_1		SYM_3	

List of Environmental Data to store in Failure Memory: CAM_IN_1
TOIL
IVVTPWM_IN_1_ENVD
V_IGK

Diagnostic	Symptom Description	Symptom	Filter type
IVVT solenoid valve; Exhaust 1	SCB	SYM_0	*MPL_STD_INI*
	SCG	SYM_1	
	OL	SYM_2	
SLV_IVVT_EX_1		SYM_3	

List of Environmental Data to store in Failure Memory: CAM_EX_1
TOIL
IVVTPWM_EX_1_ENVD
V_IGK

A.57 IVVT Limp Home Manager

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_LIH_IVVT	O/V	0...1H	0...1	1	-
1 = conditions for IVVT limp home fulfilled, 0 = not fulfilled					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	LV_ERR_CRK	NC_NR_CBK_IVVT
LV_ERR_REF_CRK_CAM_IN_i	LV_ERR_REF_CRK_CAM_EX_i	NLC_IVVT_IN	NLC_IVVT_EX
LV_ERR_MEC_IVVT_IN_i	LV_ERR_MEC_IVVT_EX_i		
LV_ERR_SLV_IVVT_IN_i	LV_ERR_SLV_IVVT_EX_i		
LV_ERR_CAM_DE_IVVT_IN_i	LV_ERR_CAM_DE_IVVT_EX_i		
LV_ERR_LIH_EXT_IVVT	LV_ERR_CAM_TOT		

FUNCTION DESCRIPTION:

Global information:

Proper operation of the IVVT system depends on faultless performance of involved components. If only one of them has a failure the IVVT system switches to the limp home.

Description:

A project specific failure can be considered by means of LV_ERR_LIH_EXT_IVVT.

Application conditions:

Initialization:

Recurrence:

100 ms

Activation:

At every engine state

Formula section:

if LV_ERR_CRK = 1
or LV_ERR_CAM_TOT = 1
or LV_ERR_LIH_EXT_IVVT = 1
 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:
or LV_ERR_REF_CRK_CAM_IN_1 = 1
or LV_ERR_MEC_IVVT_IN_1 = 1
or LV_ERR_CAM_DE_IVVT_IN_1 = 1
or LV_ERR_SLV_IVVT_IN_1 = 1
 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:
or LV_ERR_REF_CRK_CAM_EX_1 = 1
or LV_ERR_MEC_IVVT_EX_1 = 1
or LV_ERR_CAM_DE_IVVT_EX_1 = 1
or LV_ERR_SLV_IVVT_EX_1 = 1
 # NC_NR_CBK_IVVT = 1, NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:
or [(LV_ERR_REF_CRK_CAM_IN_1 = 1
or LV_ERR_MEC_IVVT_IN_1 = 1
or LV_ERR_CAM_DE_IVVT_IN_1 = 1
or LV_ERR_SLV_IVVT_IN_1 = 1)
and LV_NOT_ADJ_CAM_IVVT_IN_1 = 0]
or [(LV_ERR_REF_CRK_CAM_EX_1 = 1
or LV_ERR_MEC_IVVT_EX_1 = 1
or LV_ERR_CAM_DE_IVVT_EX_1 = 1
or LV_ERR_SLV_IVVT_EX_1 = 1)
and LV_NOT_ADJ_CAM_IVVT_EX_1 = 0]
 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:
or LV_ERR_REF_CRK_CAM_IN_1 = 1
or LV_ERR_MEC_IVVT_IN_1 = 1
or LV_ERR_CAM_DE_IVVT_IN_1 = 1
or LV_ERR_SLV_IVVT_IN_1 = 1
or [(LV_ERR_REF_CRK_CAM_IN_2 = 1
or LV_ERR_MEC_IVVT_IN_2 = 1
or LV_ERR_CAM_DE_IVVT_IN_2 = 1
or LV_ERR_SLV_IVVT_IN_2 = 1)
and LV_NOT_ADJ_CAM_IVVT_IN_2 = 0]
 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

or LV_ERR_REF_CRK_CAM_EX_1 = 1
or LV_ERR_MEC_IVVT_EX_1 = 1
or LV_ERR_CAM_DE_IVVT_EX_1 = 1
or LV_ERR_SLV_IVVT_EX_1 = 1
or [(LV_ERR_REF_CRK_CAM_EX_2 = 1
or LV_ERR_MEC_IVVT_EX_2 = 1
or LV_ERR_CAM_DE_IVVT_EX_2 = 1
or LV_ERR_SLV_IVVT_EX_2 = 1)
and LV_NOT_ADJ_CAM_IVVT_EX_2 = 0]

 # NC_NR_CBK_IVVT = 2, NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

or [(LV_ERR_REF_CRK_CAM_IN_1 = 1
or LV_ERR_MEC_IVVT_IN_1 = 1
or LV_ERR_CAM_DE_IVVT_IN_1 = 1
or LV_ERR_SLV_IVVT_IN_1 = 1)
and LV_NOT_ADJ_CAM_IVVT_IN_1 = 0]
or [(LV_ERR_REF_CRK_CAM_IN_2 = 1
or LV_ERR_MEC_IVVT_IN_2 = 1
or LV_ERR_CAM_DE_IVVT_IN_2 = 1
or LV_ERR_SLV_IVVT_IN_2 = 1)
and LV_NOT_ADJ_CAM_IVVT_IN_2 = 0]
or [(LV_ERR_REF_CRK_CAM_EX_1 = 1
or LV_ERR_MEC_IVVT_EX_1 = 1
or LV_ERR_CAM_DE_IVVT_EX_1 = 1
or LV_ERR_SLV_IVVT_EX_1 = 1)
and LV_NOT_ADJ_CAM_IVVT_EX_1 = 0]
or [(LV_ERR_REF_CRK_CAM_EX_2 = 1
or LV_ERR_MEC_IVVT_EX_2 = 1
or LV_ERR_CAM_DE_IVVT_EX_2 = 1
or LV_ERR_SLV_IVVT_EX_2 = 1)
and LV_NOT_ADJ_CAM_IVVT_EX_2 = 0]

then

LV_ERR_LIH_IVVT = 1

else

LV_ERR_LIH_IVVT = 0

endif

A.58 IVVT Individual Activations

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ACT_IND_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = active position control of individual camshaft, 0 = inactive (target: passive stop position); inlet					
LV_ACT_IND_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = active position control of individual camshaft, 0 = inactive (target: passive stop position); exhaust					
LV_ACT_TOT_IVVT_IN	O/V	0...1H	0...1	1	-
1 = inlet side calculations active (at least one inlet camshaft active), 0 = inactive					
LV_ACT_TOT_IVVT_EX	O/V	0...1H	0...1	1	-
1 = exhaust side calculations active (at least one exhaust camshaft active), 0 = inactive					
LV_ERR_BAL_NOT_IVVT_IN_i	V	0...1H	0...1	1	-
1 = individual camshaft error present at which no bank balancing, 0 = not present; inlet					
LV_ERR_BAL_NOT_IVVT_EX_i	V	0...1H	0...1	1	-
1 = individual camshaft error present at which no bank balancing, 0 = not present; exhaust					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	LV_ERR_CRK	LV_ERR_LIH_IVVT
LV_ERR_REF_CRK_CAM_IN_i	LV_ERR_REF_CRK_CAM_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LV_ERR_MEC_IVVT_IN_i	LV_ERR_MEC_IVVT_EX_i	LV_ACT_IVVT	NC_NR_CBK_IVVT
LV_ERR_SLV_IVVT_IN_i	LV_ERR_SLV_IVVT_EX_i	NLC_IVVT_IN	NLC_IVVT_EX
LV_ERR_LIH_EXT_IVVT	LV_ERR_CAM_TOT		
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i		
LV_CAM_SP_DFCT_IVVT_IN_i	LV_CAM_SP_DFCT_IVVT_EX_i		

FUNCTION DESCRIPTION:

General information:

Two-bank engines having IN_1 & IN_2 and/or EX_1 & EX_2 phasing can have failures (enduring camshaft position deviation or electrical failure leading to maximum adjustment position) that lead to different camshaft positions in the individual banks. It is possible to manage a balancing of the banks by means of setting the individual setpoint of the running camshaft dependent on the actual position of the faulty, e.g. jammed, actuator. The IVVT state is to set "Limp home" but it is necessary to keep concerned camshafts activated.

If an actuator failure is emulated the concerned camshaft must remain active in order to have the position control active.

A.58.1 IVVT Individual Error Check for Bank Balancing

Description:

This subfunction is active only for 2-bank systems. It is realized in the scheduler.

Individual errors are checked for keeping concerned camshafts active.

The evaluation of the electrical failure is done by means of C_CAM_HYS_REF_SLV_ERR_IVVT. If an electrical error is detected, what lasts some time, and the camshaft is near the reference position it is a "non-balancing" electrical error.

Application conditions:

Initialization:

Recurrence:

360 °CRK

Activation:

LV_ERR_LIH_IVVT = 1

Deactivation:

Not activation

Formula section:

if LV_ERR_REF_CRK_CAM_IN(_EX)_i = 1
 or LV_ERR_MEC_IVVT_IN(_EX)_i = 1
 or (LV_ERR_SLV_IVVT_IN(_EX)_i = 1
  and Inlet:
  CAM_AV_IVVT_IN_i >
  C_CAM_INI_IN - C_CAM_HYS_REF_SLV_ERR_IVVT
  Exhaust:
  CAM_AV_IVVT_EX_i <
  C_CAM_INI_EX + C_CAM_HYS_REF_SLV_ERR_IVVT)

then

LV_ERR_BAL_NOT_IVVT_IN(_EX)_i = 1

else

LV_ERR_BAL_NOT_IVVT_IN(_EX)_i = 0

endif

A.58.2 IVVT Individual Camshaft Activation

Description:

The best reaction on the following "non-balancing" errors is to inactivate the whole IVVT system:
LV_ERR_CRK
LV_ERR_CAM_TOT
LV_ERR_LIH_EXT_IVVT
LV_ERR_REF_CRK_CAM_IN(_EX)_i
LV_ERR_MEC_IVVT_IN(_EX)_i
LV_ERR_SLV_IVVT_IN(_EX)_i leading to reaching of the reference position

The effects of the error LV_ERR_CAM_DE_IVVT_IN(_EX)_i and of LV_ERR_SLV_IVVT_IN(_EX)_i, which leads to reaching of the maximum adjustment position, can be reduced by the bank balancing.

If there is a "balancing" error and then a "non-balancing" error is detected the whole IVVT system is inactivated. It is assumed that the probability of such case is very small.

The emulation of an actuator failure is indicated by LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1. When the failure leads to an error detection the IVVT system becomes inactive. But the concerned camshaft remains active in order to have the position control active for the failure emulation.

A special case can occur. E.g. there is a camshaft sensor error and the actuator failure emulation is active (LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1). Then, the individual camshaft remains active. This function is a tuning one, also it is managed by application. The application monitors the engine operation. In such case the real failure is to fix and then the failure emulation can continue.

This special case does not occur at effective 2-bank adjustment configuration (camshafts are inactivated even if failure emulation active) because the indvidual activation algorithm allows this easily. To do it for all configurations would require a complex algorithm.

Application conditions:

Initialization:

At transition "engine run" → "engine stop":

LV_ACT_IND_IVVT_IN(_EX)_i = 0

LV_ACT_TOT_IVVT_IN(_EX) = 0

Recurrence:

360 °CRK

Activation:

At every engine state

Formula section:

 # NC_NR_CBK_IVVT = 1:

 # NLC_IVVT_IN = 1, NLC_IVVT_EX = 0:

LV_ACT_IND_IVVT_IN_1 = LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_IN_1

LV_ACT_TOT_IVVT_IN = LV_ACT_IVVT

 # NLC_IVVT_IN = 0, NLC_IVVT_EX = 1:

LV_ACT_IND_IVVT_EX_1 = LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_EX_1

LV_ACT_TOT_IVVT_EX = LV_ACT_IVVT

 # NLC_IVVT_IN = 1, NLC_IVVT_EX = 1:

if LV_NOT_ADJ_CAM_IVVT_IN_1 = 0

then

LV_ACT_IND_IVVT_IN_1 =

LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_IN_1

LV_ACT_TOT_IVVT_IN = LV_ACT_IVVT

endif

if LV_NOT_ADJ_CAM_IVVT_EX_1 = 0

then

LV_ACT_IND_IVVT_EX_1 =

LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_EX_1

LV_ACT_TOT_IVVT_EX = LV_ACT_IVVT

endif

 # NC_NR_CBK_IVVT = 2:

if LV_NOT_ADJ_CAM_IVVT_IN(_EX)_2 = 1

then *2 banks configured, but the second one switched off. No individual activation due to balancing reasons*

LV_ACT_IND_IVVT_IN(_EX)_1 =

LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_IN(_EX)_1

LV_ACT_TOT_IVVT_IN(_EX) = LV_ACT_IVVT

else *Adjustment in both banks, unequal bank camshaft positions possible*

if LV_ERR_LIH_IVVT = 0

then *No error in system, individual activation corresponds to LV_ACT_IVVT*

LV_ACT_IND_IVVT_IN(_EX)_1 =

LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_IN(_EX)_1

LV_ACT_IND_IVVT_IN(_EX)_2 =

LV_ACT_IVVT **or** LV_CAM_SP_DFCT_IVVT_IN(_EX)_2

LV_ACT_TOT_IVVT_IN(_EX) = LV_ACT_IVVT

else *Error in system*

if LV_ERR_CRK = 1

or LV_ERR_CAM_TOT = 1

or LV_ERR_LIH_EXT_IVVT = 1

or LV_ERR_BAL_NOT_IVVT_IN(_EX)_1 = 1

or LV_ERR_BAL_NOT_IVVT_IN(_EX)_2 = 1

then *Errors at which no attempt for any bank balancing, each camshaft inactivated. No actuator failure emulation possible.*

LV_ACT_IND_IVVT_IN(_EX)_1 = 0

LV_ACT_IND_IVVT_IN(_EX)_2 = 0

LV_ACT_TOT_IVVT_IN(_EX) = 0

else *Both banks are kept activate due to balancing, even if both have error*

LV_ACT_IND_IVVT_IN(_EX)_1 = 1

LV_ACT_IND_IVVT_IN(_EX)_2 = 1

LV_ACT_TOT_IVVT_IN(_EX) = 1

endif

endif

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CAM_HYS_REF_SLV_ERR_IVVT	1	0...FFH	0...95.625	0.375	°CRK
Hysteresis at reference position for detection of electrical errors leading to reaching of reference position					

A.59 IVVT External Failure

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_ERR_LIH_EXT_IVVT	O/V	0...1H	0...1	1	-
1 = external failure disturbing IVVT present, 0 = not present					

Input data:

LV_ERR_MAP	LV_ERR_MAF	LV_ERR_TPS	LV_N_LIM_ETC_LIH
LV_MTC_CUR_OFF	LV_ES	LV_ST	CONF_TOIL_MDL
LV_ERR_TCO			

FUNCTION DESCRIPTION:

Global information:

An external failure can disturb the IVVT system performance. Then a flag is set that leads to reaching "Limp home" state. The term external means that the corresponding component is not directly necessary for the camshaft phasing control.

Description:

If only one of the external failures occurs LV_ERR_LIH_EXT_IVVT is set.

Application conditions:

Initialization:

Recurrence: 100 ms

Activation: At every engine state

Formula section:

If LV_ERR_MAP = 1
or LV_ERR_MAF = 1
or LV_ERR_TPS = 1
or LV_N_LIM_ETC_LIH = 1
or (LV_MTC_CUR_OFF = 1 **and** LV_ES = 0 **and** LV_ST = 0)
or (CONF_TOIL_MDL =! 0 **and** LV_ERR_TCO = 1)
then LV_ERR_LIH_EXT_IVVT = 1
else LV_ERR_LIH_EXT_IVVT = 0
endif

A.60 Cooling fan diagnosis

FUNCTION DESCRIPTION:

General information:

Because of a RLY/PWM-fan configuration [NC_ECF_CONF = 2] it is possible to choose the wanted fan diagnosis strategy during ECU runtime with use of a logical constant (CONF_CFA) in combination with project specific requirements. Either the diagnosis function for RLY-fan(s) or the diagnosis function for PWM-fan(s) is activated. It is strongly prohibited to enable both functions in parallel.

For each RLY fan stage as well as each PWM value, a signal at the ECU output has to be generated by the infrastructure to control the available cooling fan(s) (hardware component) at the vehicle. A diagnosis function is necessary in both cases. As soon as the RLY-fan diagnosis function is activated, all PWM-fan diagnosis output data are set to neutral values and vice versa.

A global electronically controlled cooling fan error is set without debounce as soon as the corresponding diagnosis function is detecting an electrical cooling fan failure by one of the available cooling fan stage(s) (RLY or PWM controlled).

A.60.1 Cooling fan diagnosis (RLY version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ECF_EL_RLY	O/V	0...1H	0...1	1	-
Boolean for electronic RLY controlled cooling fan error					
LV_ERR_ECF_EL[1]	O/V	0...1H	0...1	1	-
Boolean for error currently present on RLY_FAN_L command signal					
LV_CDN_DIAG_ECF_EL[1]	O/V	0...1H	0...1	1	-
Diagnosis condition RLY_FAN_L diagnosis					
ERR_SYM_ECF_EL[1]	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom RLY_FAN_L diagnosis					
LV_END_DIAG_ECF_EL[1]	O/V	0...1H	0...1	1	-
End of diagnosis RLY_FAN_L diagnosis					
CDN_DIAG_ECF_EL[1]	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					
LV_ERR_ECF_EL[2]	O/V	0...1H	0...1	1	-
Boolean for error currently present on RLY_FAN_H command signal.					
LV_CDN_DIAG_ECF_EL[2]	O/V	0...1H	0...1	1	-
Diagnosis condition RLY_FAN_H diagnosis					
ERR_SYM_ECF_EL[2]	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom RLY_FAN_H diagnosis					
LV_END_DIAG_ECF_EL[2]	O/V	0...1H	0...1	1	-
End of diagnosis RLY_FAN_H diagnosis					
CDN_DIAG_ECF_EL[2]	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					
LV_INH_DIAG_ECF_EL[1]	V	0...1H	0...1	1	-
Boolean for electronic controlled thermostat signal range diagnosis RLY_FAN_L inhibit					
LV_INH_DIAG_ECF_EL[2]	V	0...1H	0...1	1	-
Boolean for electronic controlled thermostat signal range diagnosis RLY_FAN_H inhibit					

Input data:

LV_IGK	LV_CDN_VB_OBD1	LV_DC	

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the signal from the driver which controls the cooling fan relay signal (low and high).

A global electronically controlled cooling fan error is set without debounce as soon as the diagnosis function is detecting an electrical cooling fan failure by one of the available cooling fan(s).

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.

Error-symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> Short circuit to Vbatt (SCB)        (= SYM_0)
 │  │  └────> Short circuit to GND (SCG)          (= SYM_1)
 │  └───────> Open load (OC)                      (= SYM_2)
 └──────────> -                                   (= SYM_3)
```

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at reset

Recurrence: 2s

Activation: at every engine operating state

(see: "ENTE scheduler")

Formula section:

If LV_IGK = 1 **and**
LV_CDN_VB_OBD1 = 1
then LV_INH_DIAG_ECF_EL[1] = 0
LV_INH_DIAG_ECF_EL[2] = 0
else LV_INH_DIAG_ECF_EL[1] = 1
LV_INH_DIAG_ECF_EL[2] = 1
endif

If LV_ERR_ECF_EL[1] = 1 **or**
LV_ERR_ECF_EL[2] = 1
then LV_ERR_ECF_EL_RLY = 1 (without debounce)
else LV_ERR_ECF_EL_RLY = 0
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_ECF_EL[1]	1	0...FFH	0...255	1	-
Debounce counter increment RLY_FAN_L diagnosis					
C_ABC_INC_ECF_EL[2]	1	0...FFH	0...255	1	-
Debounce counter increment RLY_FAN_H diagnosis					
C_ABC_MAX_ECF_EL[1]	1	01...FFH	1...255	1	-
Debounce counter maximum value – RLY_FAN_L diagnosis					
C_ABC_MAX_ECF_EL[2]	1	01...FFH	1...255	1	-
Debounce counter maximum value – RLY_FAN_H diagnosis					

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Cooling fan relay low diagnosis	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
ECF_EL_1		SYM_3	

Cooling fan relay high diagnosis	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
ECF_EL_2		SYM_3	

1.2 Cooling fan diagnosis (PWM version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ECF_EL_PWM	O/V	0...1H	0...1	1	-
Boolean for electronic controlled PWM cooling fan error					
LV_ERR_ECF_EL_PWM[1]	O/V	0...1H	0...1	1	-
Boolean for error currently present on CFA_PWM command signal					
LV_CDN_DIAG_ECF_EL_PWM[1]	O/V	0...1H	0...1	1	-
Diagnosis condition CFA_PWM diagnosis					
ERR_SYM_ECF_EL_PWM[1]	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom CFA_PWM diagnosis					
LV_END_DIAG_ECF_EL_PWM[1]	O/V	0...1H	0...1	1	-
End of diagnosis CFA_PWM diagnosis					
CDN_DIAG_ECF_EL_PWM[1]	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					
LV_INH_DIAG_ECF_EL_PWM[1]	O/V	0...1H	0...1	1	-
Boolean for electronic controlled thermostat signal range diagnosis CFA_PWM inhibit					

Input data:

LV_IGK	LV_CDN_VB_OBD1	LV_DC	

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the signal from the driver which controls the electronical cooling fan.

A global electronically controlled cooling fan error is set without debounce as soon as the diagnosis function is detecting an electrical cooling fan failure by one of the available cooling fan(s).

The input signal is a modulated control pulse (PWM).

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.
The diagnosis condition calculation is done according **“Limited”** duty cycle, thus no definition of the PWM diagnosis windows is necessary.

Error-symptoms are defined to this diagnosis function as following:

```
□□□□
│ │ │ └─> Short circuit to Vbatt (SCB)          (= SYM_0)
│ │ └───> Short circuit to GND (SCG)           (= SYM_1)
│ └─────> Open load (OC)                       (= SYM_2)
└───────> -                                    (= SYM_3)
```

Application conditions:

Initialization: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 and at reset

Recurrence: 1s

Activation: at every engine operating state

(see: “ENTE scheduler”)

Formula section:

If LV_IGK = 1 **and**
LV_CDN_VB_OBD1 = 1
then LV_INH_DIAG_ECF_EL_PWM[1] = 0
else LV_INH_DIAG_ECF_EL_PWM[1] = 1
endif

If LV_ERR_ECF_EL_PWM[1] = 1
then LV_ERR_ECF_EL_PWM = 1 (without debounce)
else LV_ERR_ECF_EL_PWM = 0
endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
PWM Cooling fan diagnosis	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
ECF_EL_PWM_1		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_ECF_EL_PWM[1]	1	0...FFH	0...255	1	-
Debounce counter increment CFA_PWM diagnosis					
C_ABC_MAX_ECF_EL_PWM[1]	1	01...FFH	1...255	1	-
Debounce counter maximum value – CFA_PWM diagnosis					

A.61 Malfunction Indicator Light diagnosis (LV_ERR_MIL)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MIL	O/V	0...01H	0...1	1	-
Boolean for error currently present on MIL command signal.					
LV_CDN_DIAG_MIL	O/V	0...01H	0...1	1	-
Diagnosis condition MIL diagnosis					
ERR_SYM_MIL	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom MIL diagnosis					
LV_END_DIAG_MIL	O/V	0...01H	0...1	1	-
End of diagnosis MIL diagnosis					
LV_INH_DIAG_MIL	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition MIL diagnosis					
CDN_DIAG_MIL	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					

Input data:

LV_IGK	LV_CDN_VB_OBD1	V_IGK	LV_DC

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the signal from the driver which controls the the malfunction indicator light. As the MIL component is supplied from V_IGK, diagnosis can only be performed if voltage after IGK is available.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.

Error-symptoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> Short circuit to Vbatt (SCB)     (= SYM_0)
 │  │  └────> Short circuit to GND (SCG)       (= SYM_1)
 │  └───────> Open load (OL)                   (= SYM_2)
 └──────────> -                                (= SYM_3)
```

Application conditions:

Initialisation:

-all variables (except LV_END_DIAG_MIL) are initialised with 0 at every LV_IGK = 0 ->1 and reset
-LV_END_DIAG_MIL is initialised with 0 at every (LV_IGK = 0 ->1 and LV_DC = 0) and at reset

Recurrence: 20 ms

Activation: *at every engine operating state*

Formula section:

IF LV_IGK = 1 **AND**
LV_CDN_VB_OBD1 = 1 **AND**
V_IGK > C_V_IGK_MIN_DIAG_MIL
THEN LV_INH_DIAG_MIL = 0
ELSE LV_INH_DIAG_MIL = 1
ENDIF

Error detection

If Hardware detects error symptom 0/1/2 (info from BSW)
Then ERR_SYM_MIL = SYM_0/1/2
LV_ERR_MIL = 1 (after debounce)
Else ERR_SYM_MIL = NO_SYM
LV_ERR_MIL = 0 (after rebounce)
Endif

End of diagnosis

For calculation of LV_END_DIAG_MIL see “Anti-bounce Algorithm, calculation of the end of diagnosis”.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_MIL	1	0...FFH	0...255	1	-
Debounce counter increment MIL diagnosis					
C_ABC_MAX_MIL	1	01...FFH	1...255	1	-
Debounce counter maximum value – MIL diagnosis					
C_V_IGK_MIN_DIAG_MIL		0...FFH	0...26	0,102	V
Minimum Key – battery voltage to perform MIL diagnosis					

A.61.1 MIL diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Malfunction indicator lamp	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
MIL		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
V_IGK
TCO_ST
VB_MMV

A.62 Fuel pump relay diagnosis (LV_ERR_RLY_EFP)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_RLY_EFP	O/V	0...01H	0...1	1	-
Boolean for error currently present on RLY_EFP command signal.					
LV_CDN_DIAG_RLY_EFP	O/V	0...01H	0...1	1	-
Diagnosis condition RLY_EFP diagnosis					
ERR_SYM_RLY_EFP	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom RLY_EFP diagnosis					
LV_END_DIAG_RLY_EFP	O/V	0...01H	0...1	1	-
End of diagnosis RLY_EFP diagnosis					
LV_INH_DIAG_RLY_EFP	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition RLY_EFP diagnosis					
CDN_DIAG_RLY_EFP	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					

Input data:

LV_IGK	LV_CDN_VB_OBD1	LV_DC	

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the signal from the driver which controls the electric fuel pump relay signal.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.

Error-symptoms are defined to this diagnosis function as following :

```
□□□□
| | | └─> Short circuit to Vbatt (SCB)      (= SYM_0)
| | └───> Short circuit to GND (SCG)        (= SYM_1)
| └─────> Open load (OL)                    (= SYM_2)
└───────> -                                 (= SYM_3)
```

Application conditions:

Initialisation: all variables (except LV_END_DIAG_RLY_EFP) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every (LV_IGK = 0 ->1 and LV_DC = 0) and at reset

Recurrence: 300 ms

Activation: *at every engine operating state*

Formula section:

IF LV_IGK = 1 **AND**
LV_CDN_VB_OBD1 = 1
THEN LV_INH_DIAG_RLY_EFP = 0
ELSE LV_INH_DIAG_RLY_EFP = 1
ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_RLY_EFP	1	0...FFH	0...255	1	-
Debounce counter increment RLY_EFP diagnosis					
C_ABC_MAX_RLY_EFP	1	01...FFH	1...255	1	-
Debounce counter maximum value – RLY_EFP diagnosis					

A.62.1 Application incidences for electric fuel pump diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel pump relay diagnosis	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
RLY_EFP		SYM_3	

List of Environmental Data to store in Failure Memory: ENG_STATE
VS
V_IGK
TIA_ST

A.63 Air Conditioning pressure signal diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ACP	V/S/O	0H...1H	0...1	1	-
Error currently present on AC pressure signal acquisition (after debounce)					
ERR_SYM_ACP	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for ACP diagnosis					
LV_CDN_DIAG_ACP	O/V	0H...01H	0...1	1	-
Diagnosis condition for ACP diagnosis					

Input data:

V_ACP_MES	STATE_ACP	CONF_ACP	CONF_ACC
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the analog input signal (V_ACP_MES) from Air conditioning transducer pressure sensor to the micro-controller via the multiplexed.

Application conditions:

The error detection is only performed if the vehicle is equippted with air condition control (CONF_ACC = 1), the air conditioner option has been recognized (CONF_ACP has been set to 1) and if no failure is present on senor supply voltage (LV_ERR_VCC_SENS_SUB = 0).

Recurrence: **100 ms**

Formula section:

If V_ACP_MES < C_ACP_MIN_DIAG

then *`Short to Ground´ or `Line Break´ is detected.*

and the anti-bounce counter is incremented with C_ABC_INC_ACP_DIAG

If V_ACP_MES > C_ACP_MAX_DIAG

then *`Short to Battery Voltage´ is detected.*

and the anti-bounce counter is incremented with C_ABC_INC_ACP_DIAG

If the anti-bounce counter reaches C_ABC_MAX_ACP_DIAG

Then LV_ERR_ACP = 1

If the anti-bounce counter reaches 0

Then LV_ERR_ACP = 0

* Debounce :

Anti - bounce counter increment : C_ABC_INC_ACP_DIAG
Maximum value of anti - bounce counter : C_ABC_MAX_ACP_DIAG

Emergency operation :

AC-Pressure state set to high in case of ACP-Error (Spec xx400Dyy.zzz)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ACP_MIN_DIAG	1	0...3FFH	0...5	0.0195	V
Air conditionning transducer pressure raw acquisition threshold for short detection to ground					
C_ACP_MAX_DIAG	1	0...3FFH	0...5	0.0195	V
Air conditionning transducer pressure raw acquisitioshold for short detection to battery voltage or line break					
C_ABC_INC_ACP_DIAG	1	0...FFH	0...255	1	-
Anti – bounce counter increment for Air conditionning transducer pressure sensor diagnosis					
C_ABC_MAX_ACP_DIAG	1	0...FFH	0...255	1	-
Maximum value of the anti - bounce for Air conditionning transducer pressure sensor diagnosis					

A.63.1 Air Conditioning Pressure sensor diagnosis (Appl. Inc.)

Input data:

LV_ERR_ACP	LV_ACCIN	LV_RLY_ACCOUT	V_ACP_MES_ENVD_H
V_ACP_MES_ENVD_L			

Error treatment:

Error-symtoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> short circuit to battery (SCB)                    (= SYM_0)
 │ │ └───> short circuit to ground (SCG) or open circuit (OC) (= SYM_1)
 │ └─────> -                                                 (= SYM_2)
 └───────> -                                                 (= SYM_3)
```

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
A/C pressure sensor diagnosis	Short circuit to battery (SCB)	SYM_0	*STD*
	Short circuit to ground (SCG) or open circuit (OC)	SYM_1	
		SYM_2	
ACP		SYM_3	

List of Environmental Data to store in Failure Memory: LV_ACCIN
LV_RLY_ACCOUT
V_ACP_MES_ENVD_H
V_ACP_MES_ENVD_L

A.64 Ambient pressure sensor diagnosis (AMP)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_AMP	O/V	0...1H	0...1	1	-
AMP sensor error detected					
LV_CDN_DIAG_AMP	O/V	0...1H	0...1	1	-
Status of diagnosis for AMP sensor					
ERR_SYM_AMP	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom AMP sensor					
LV_END_DIAG_AMP	O/V	0...1H	0...1	1	-
End of diagnosis for Ambient pressure sensor diagnosis					

Input data:

VP_AMP	LV_INH_DIAG_AMP	NC_IDX_DIAG_AMP	VP_AMP_MAX_DIAG
VP_AMP_MIN_DIAG			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_AMP	1	0...FFH	0...255	1	-
Increment debounce counter					
C_ABC_MAX_AMP	1	1...FFH	1...255	1	-
Maximum value debounce counter					

Import actions:

ACTION_ERRM_FilterSymptom(IN <IDX_DIAG>, IN <LV_CDN_DIAG>, IN <ERR_SYM>, IN <ABC_INC>, IN <ABC_DEC>, IN <ABC_MAX>, OUT <LV_ERR>)
This action computes the elementary anti-bounce filter
ACTION_ERRM_GetErrSym(IN <IDX_DIAG>, OUT <ERR_SYM>)
Action that returns the symptom of the failure
ACTION_ERRM_GetLvCdnDiag(IN <IDX_DIAG>, OUT <LV_CDN_DIAG>)
Action that returns the diagnostic condition
ACTION_ERRM_GetLvEndDiag(IN <IDX_DIAG>, OUT <LV_END_DIAG>)
Action that returns the status of the failure availability

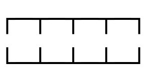
A.64.1 General information:

The purpose of the diagnosis shall be to detect electrical faults as defined by OBD I requirements.

The signal of the altitude pressure sensor on the A/D-input of the microcontroller is checked.

Description:

Error-symptoms are defined to this diagnosis function as following:



| | | └─> Short circuit to battery "SCP" (= SYM_0)
| | └──> Short circuit to Ground "SCG" (= SYM_1)
| └────> - (= SYM_2)
└─────> - (= SYM_3)

Remark: Depending on the Hardware, Open Circuit "OC" is assigned to SYM_0 or to SYM_1.

Error debounce:

- Debounce counter increment: C_ABC_INC_AMP
- Debounce counter decrement: 1
- Debounce counter maximum value: C_ABC_MAX_AMP

Function Description

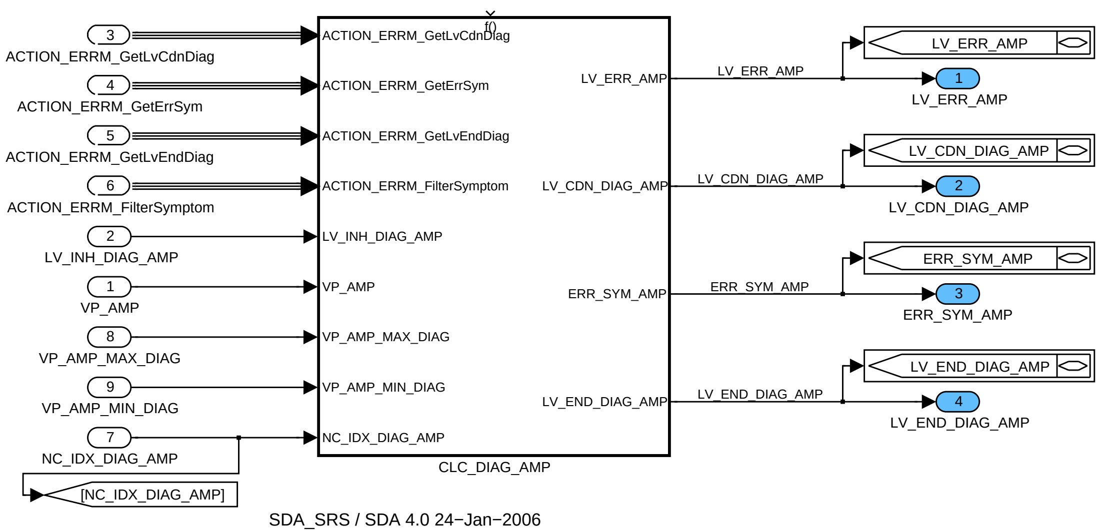


Figure 4 INSY_SIGDGAMP0

A.64.1.1 SUBFUNCTION: CLC_DIAG_AMP

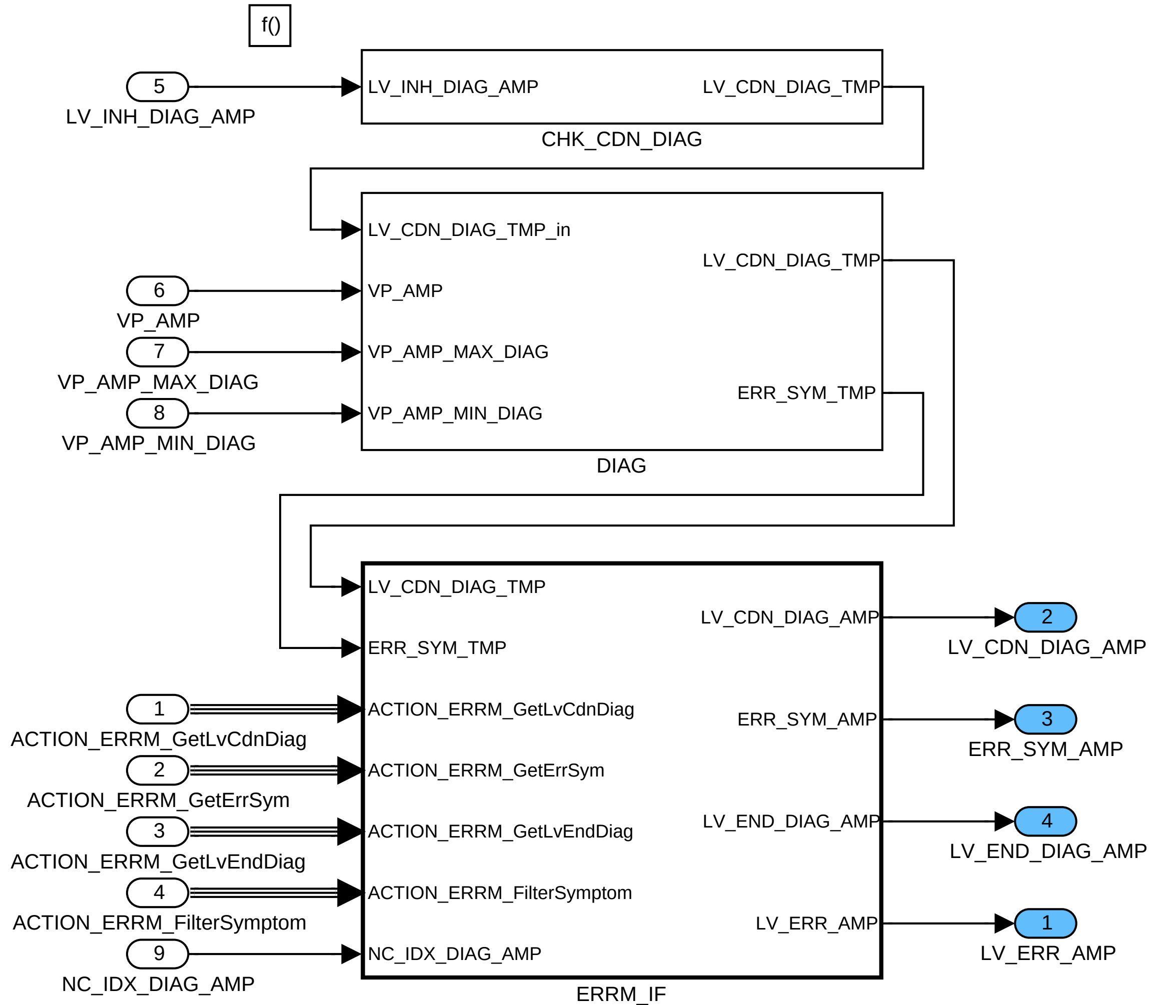


Figure 5 INSY_SIGDGAMP0/ CLC_DIAG_AMP

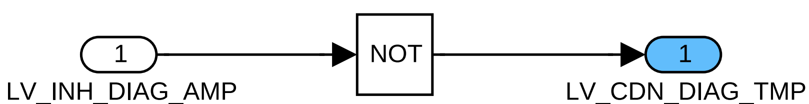


Figure 6 INSY_SIGDGAMP0/ CLC_DIAG_AMP/ CHK_CDN_DIAG

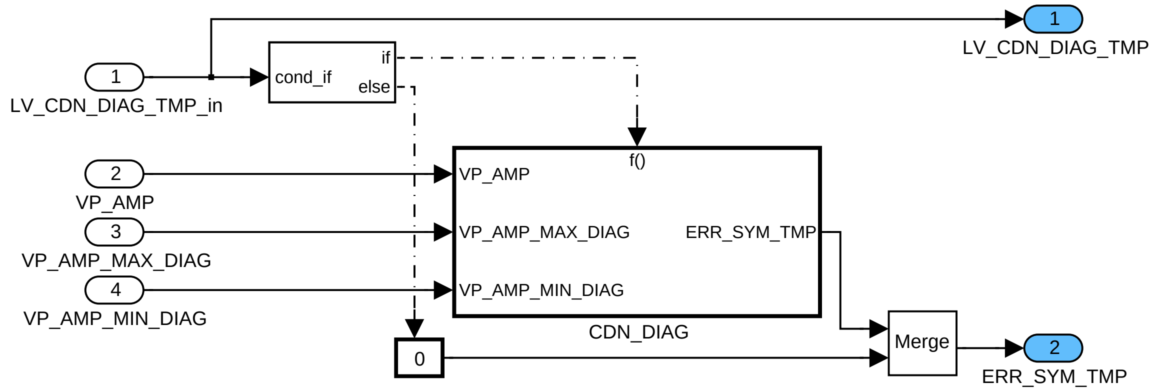


Figure 7 INSY_SIGDGAMP0/ CLC_DIAG_AMP/ DIAG

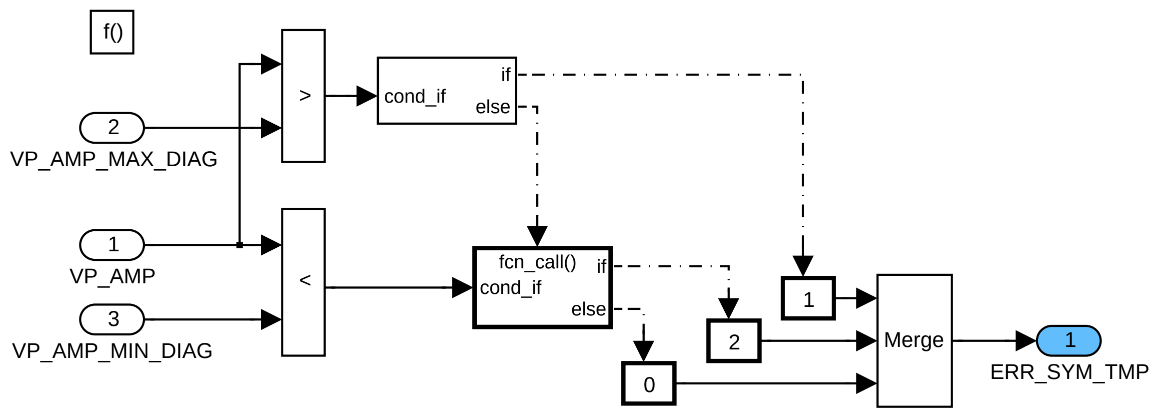


Figure 8 INSY_SIGDGAMP0/ CLC_DIAG_AMP/ DIAG/ CDN_DIAG

<u>Interface to ERRM:</u>

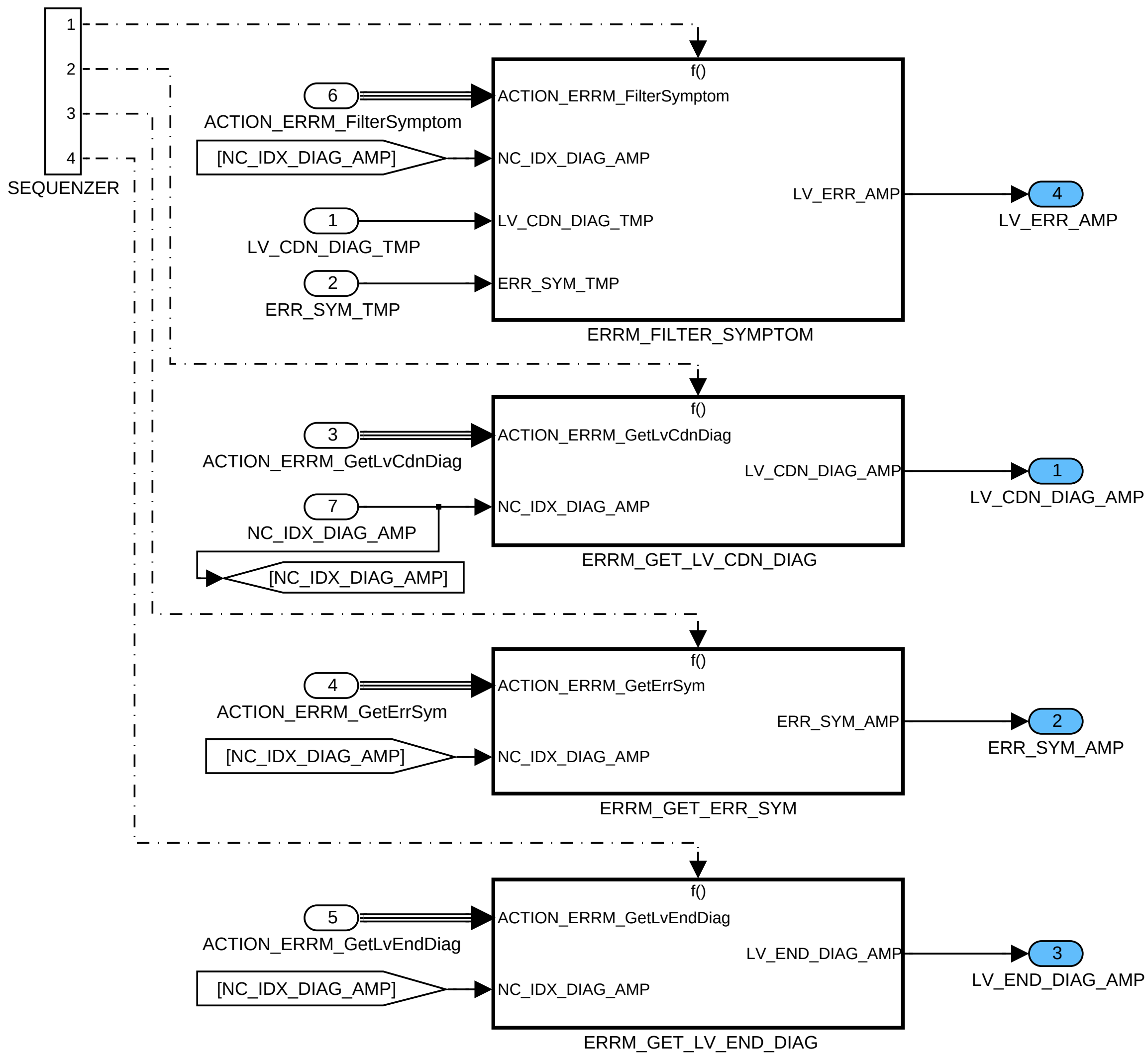


Figure 9 INSY_SIGDGAMP0/ CLC_DIAG_AMP/ ERRM_IF

A.65 Ambient pressure sensor diagnosis (Appl. Inc.)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_INH_DIAG_AMP	O/V	0... 1H	0... 1	1	[-]
Inhibition of AMP diagnostics					
VP_AMP_MAX_DIAG	O/V	0... 7FFFH	0... 4.99984741211	152.59e-6	[V]
Maximum threshold for VP_AMP for error detection					
VP_AMP_MIN_DIAG	O/V	0... 7FFFH	0... 4.99984741211	152.59e-6	[V]
Minimum threshold for VP_AMP for error detection					

Input Data:

LV_AMP_SWI	LV_IGK	NC_AMP_CONF

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_VP_AMP_MAX_DIAG	1	0... 7FFFH	0... 4.99984741211	152.59e-6	[V]
AMP maximum threshold for error detection					
C_VP_AMP_MIN_DIAG	1	0... 7FFFH	0... 4.99984741211	152.59e-6	[V]
AMP minimum threshold for error detection					

General Information

Configuration data:

Diagnosis	Symptom	Nr	ABC type
Ambient pressure sensor diagnosis with short circuit to battery	Short circuit to battery "SCP"	0	STD_INI
Ambient pressure sensor diagnosis with short circuit to ground	Short circuit to Ground "SCG", or Open Circuit "OC"	1	STD_INI

Application Conditions

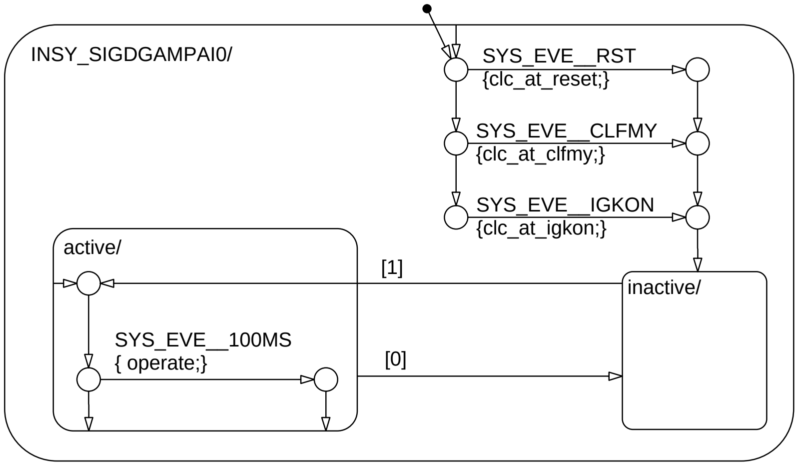
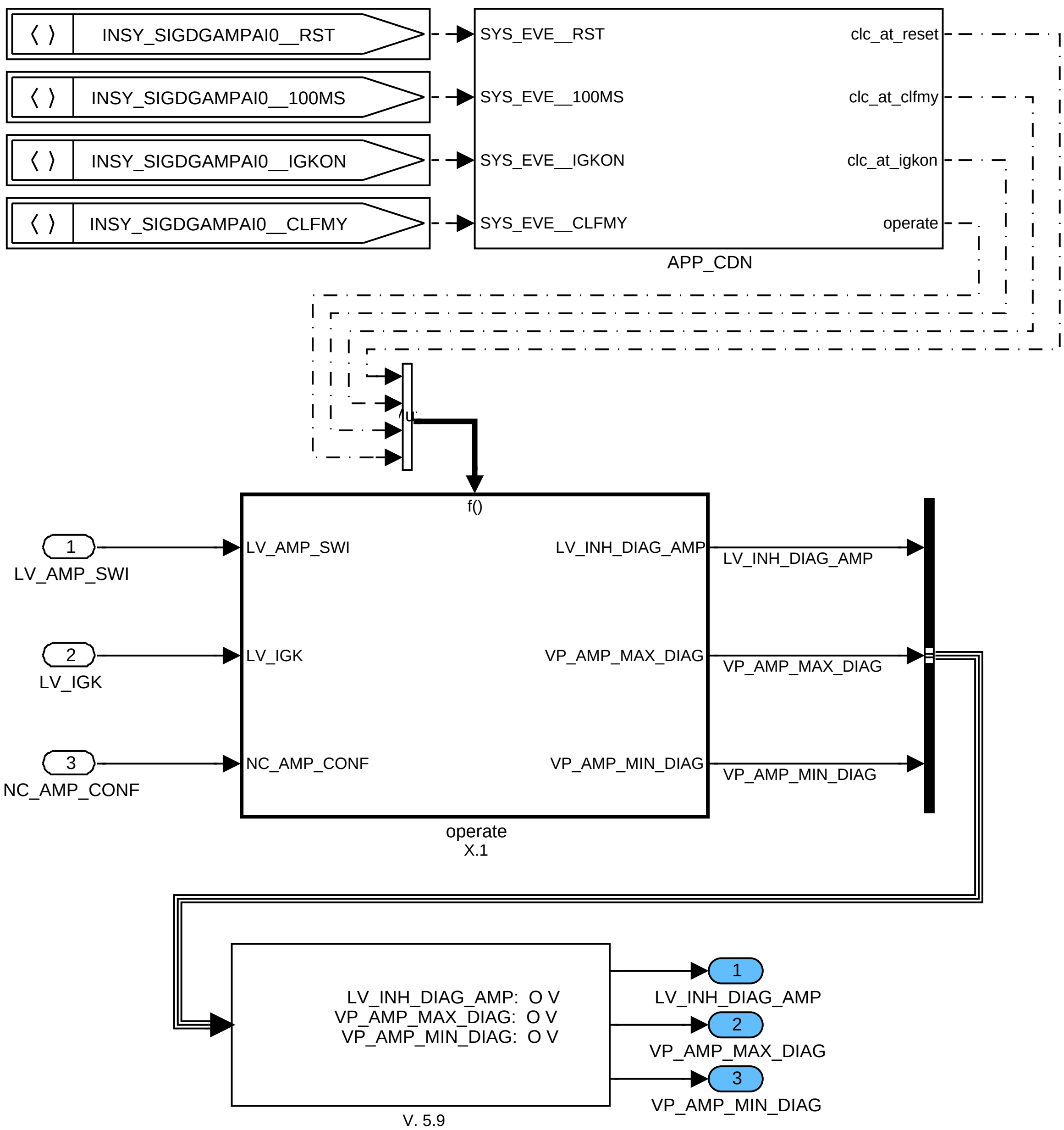


Figure 10:

Function description



SDA_SRS / SDA V 5.0.2.2 09−Feb−2006

Figure 11:

A.65.1 Recurrecne_100ms:

Remark:
For configurations without AMP sensor (NC_AMP_CONF = 0) all outputs are calculated only once at reset and set to 0 and then never changed.

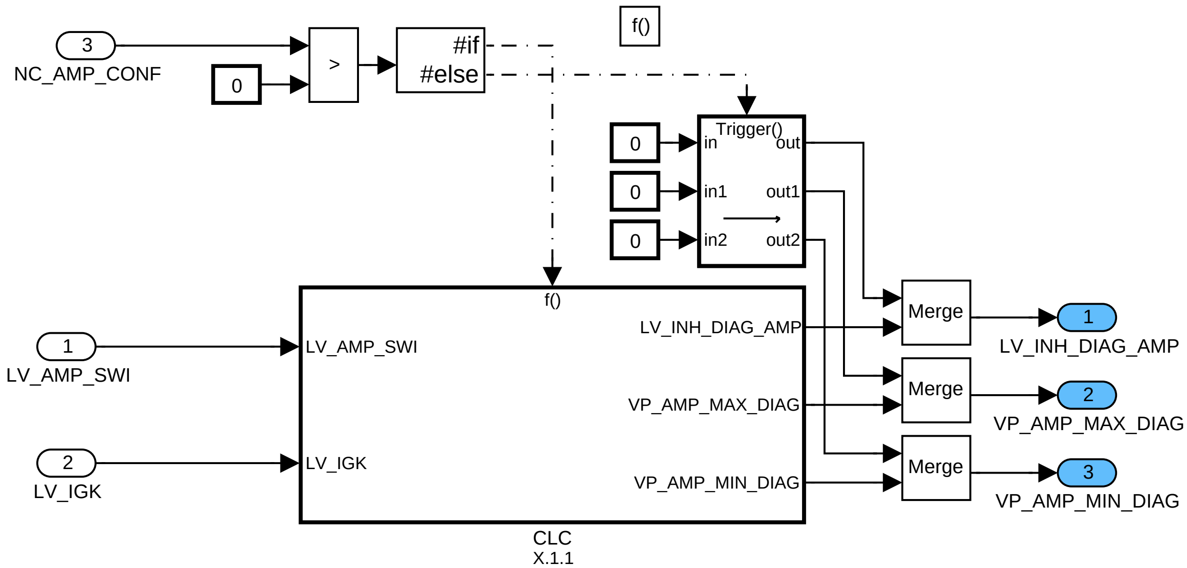


Figure 12:

A.65.1.1 Calculation:

A.65.1.1.1 Calculation of diagnosis thresholds:

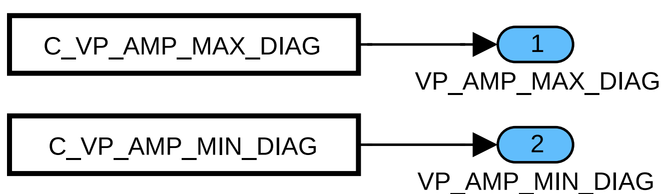


Figure 13:

A.65.1.1.2 Inhibition:

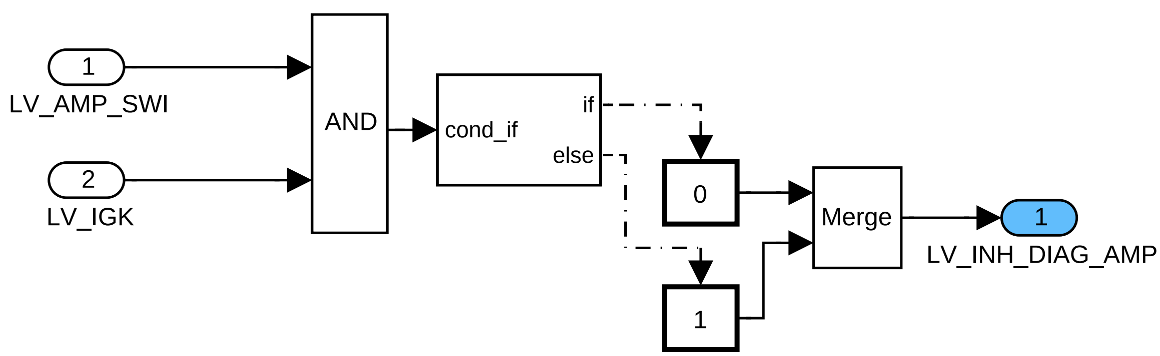


Figure 14:

A.66 Manifold Air Pressure Sensor Diagnosis (MAP)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
ERR_SYM_EL_MAP	O/V	0 1 2 4 8	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	-	[-]
Detected symptom MAP sensor					
LV_CDN_DIAG_EL_MAP	O/V	0... 1H	0... 1	1	[-]
Status of diagnosis for MAP sensor					
LV_END_DIAG_EL_MAP	O/V	0... 1H	0... 1	1	[-]
End of diagnosis for Manifold Air Pressure sensor diagnosis					
LV_ERR_EL_MAP	O/V	0... 1H	0... 1	1	[-]
Electrical MAP sensor error					
LV_T_ST_MAP_DIAG	V	0... 1H	0... 1	1	[-]
Logical bit indicating that the time period C_T_ST_MAP_DIAG has elapsed during start					

Input Data:

LV_INH_DIAG_EL_MAP	VP_MAP	VP_MAP_MAX_DIAG	VP_MAP_MIN_DIAG
NC_IDX_DIAG_MAP	LV_ES	LV_ST	LV_ERR_TPS
LV_ERR_TPS_PLAUS	TPS	N_32	

Calibration Data:

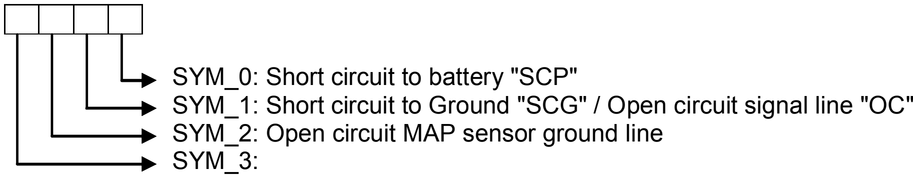
Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_ABC_INC_EL_MAP	1	0... FFH	0... 255	1	[-]
Increment debounce counter					
C_ABC_MAX_EL_MAP	1	1... FFH	1... 255	1	[-]
Maximum value debounce counter					
C_T_ST_MAP_DIAG	1	0... FFFFH	0... 655.35	0.01	[s]
Time after LV_ST for determination of "OC" MAP sensor ground line					
C_VP_MAP_MAX_OC_DIAG	1	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
VP_MAP-Threshold for Ground Line Open Circuit error detection on MAP					
IP_TPS_MAX_MAP_DIAG	6	0... FFH	0... 119.5	0.4686275	[°TPS]
LDP_N_32_IP_TPS_MAX_MAP_DIAG	6	0... FFH	0... 8160	32	[rpm]
Maximum TPS position to perform MAP Ground line Open Circuit detection					

Import Actions:

ACTION_ERRM_FilterSymptom(IN <IDX_DIAG>, **IN** <LV_CDN_DIAG>, **IN** <ERR_SYM>, **IN** <ABC_INC>, **IN** <ABC_DEC>, **IN** <ABC_MAX>, **OUT** <LV_ERR>)
ACTION_ERRM_GetErrSym(IN <IDX_DIAG>, **OUT** <ERR_SYM>)
ACTION_ERRM_GetLvCdnDiag(IN <IDX_DIAG>, **OUT** <LV_CDN_DIAG>)
ACTION_ERRM_GetLvEndDiag(IN <IDX_DIAG>, **OUT** <LV_END_DIAG>)

General Information

Diagnosis of VP_MAP, which is an analogue input signal in the A/Dconverter of the Microprocessor. Errorsymptoms are defined to this diagnosis function as following :



The errormanagement variables are initialized according to filtertype.
Error debounce:
Debounce counter increment: C_ABC_INC_EL_MAP
Debounce counter decrement: 1
Debounce counter maximum value: C_ABC_MAX_EL_MAP

Application Conditions

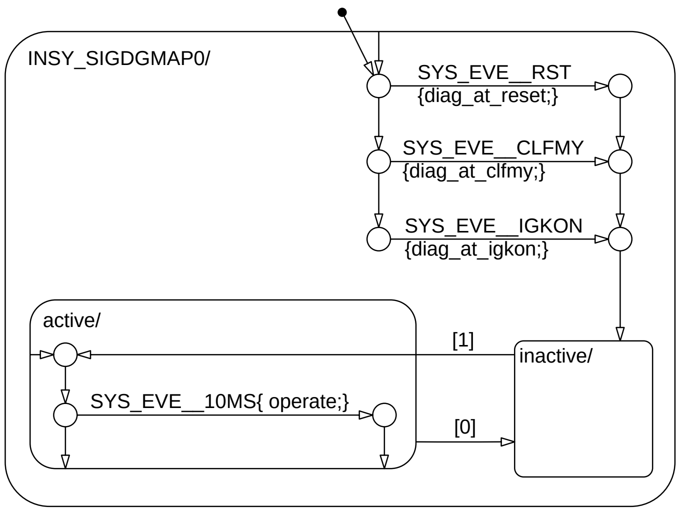
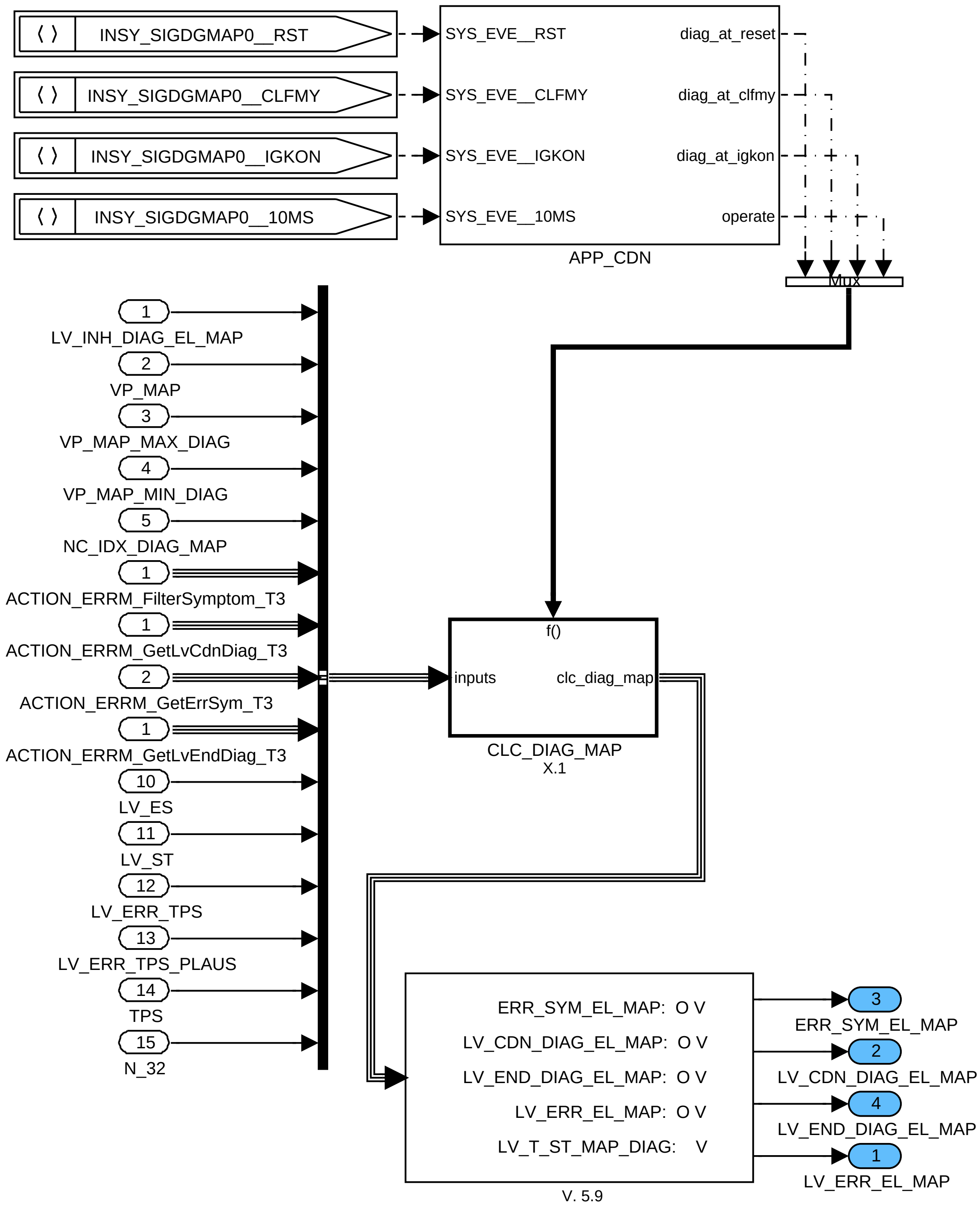


Figure 15:

Function description



SDA_SRS / SDA V 5.0.6 19−Apr−2007

Figure 16:

A.66.1 Calculation.

A.66.1.1 Calculation of LV_CDN_DIAG.

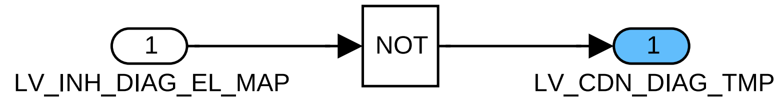


Figure 17:

A.66.1.2 Diagnosis.

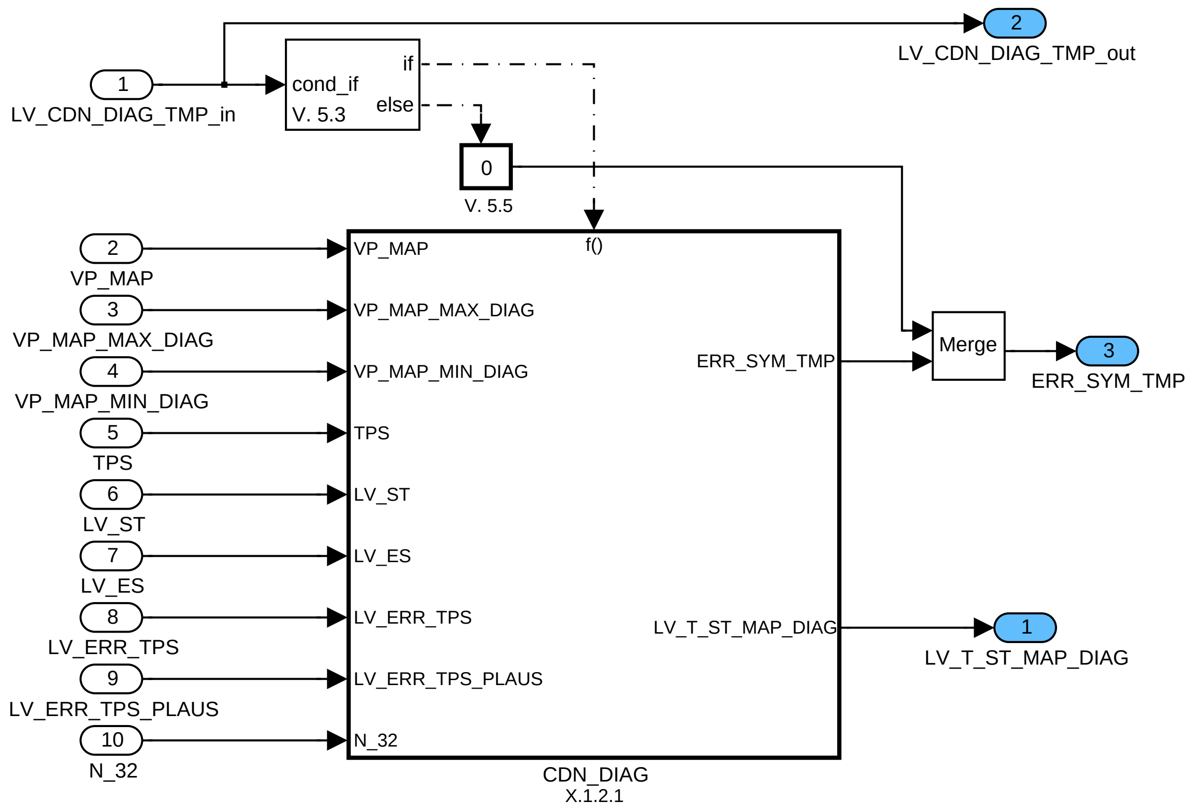


Figure 18:

A.66.1.2.1 Calculation of ERR_SYM.

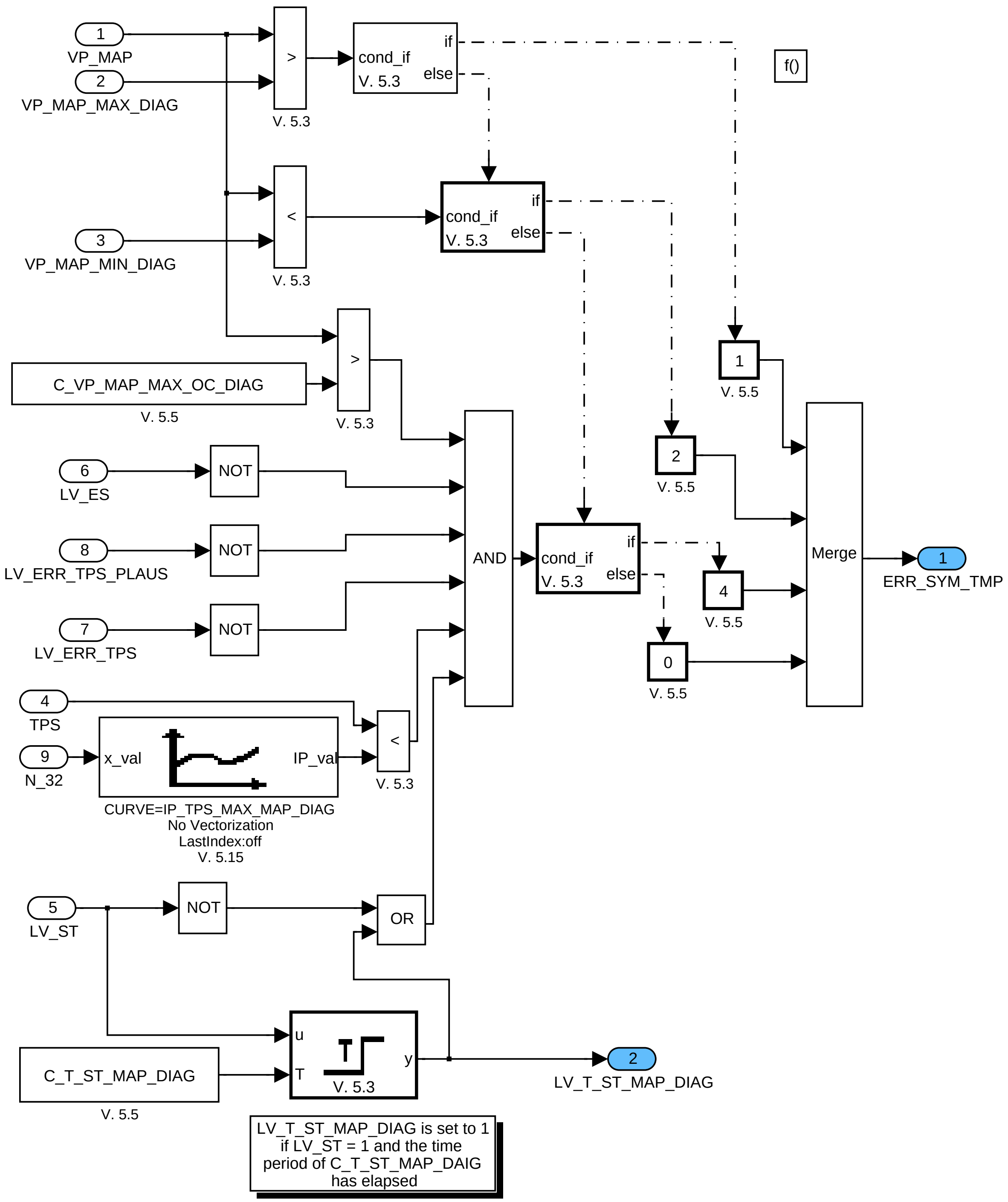


Figure 19:

A.66.1.3 Interface to ERRM:

A.66.1.3.1 ERRM FILTER

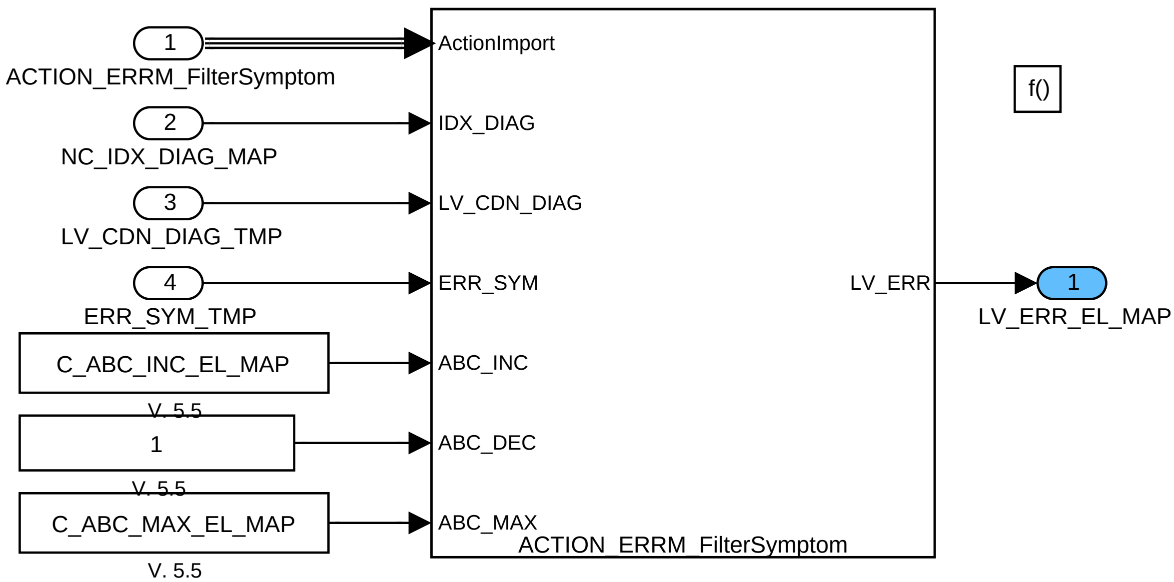


Figure 20:

A.67 Pressure upstream throttle sensor diagnosis (PUT)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_EL_PUT	O/V	0...1H	0...1	1	-
Electrical PUT sensor error					
LV_CDN_DIAG_EL_PUT	O/V	0...1H	0...1	1	-
Status of diagnosis for PUT sensor					
ERR_SYM_EL_PUT	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom PUT sensor					
LV_END_DIAG_EL_PUT	O/V	0...1H	0...1	1	-
End of pressure upstream throttle sensor diagnosis					

Input data:

LV_INH_DIAG_EL_PUT	VP_PUT_MV	VP_PUT_MV_MAX_DIAG	VP_PUT_MV_MIN_DIAG
NC_IDX_DIAG_PUT			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_EL_PUT	1	0...FFH	0...255	1	-
Increment debounce counter					
C_ABC_MAX_EL_PUT	1	1...FFH	1...255	1	-
Maximum value debounce counter					

Import actions:

ACTION_ERRM_FilterSymptom(IN <IDX_DIAG>, IN <LV_CDN_DIAG>, IN <ERR_SYM>, IN <ABC_INC>, IN <ABC_DEC>, IN <ABC_MAX>, OUT <LV_ERR>)
This action computes the elementary anti-bounce filter
ACTION_ERRM_GetErrSym(IN <IDX_DIAG>, OUT <ERR_SYM>)
Action that returns the symptom of the failure
ACTION_ERRM_GetLvCdnDiag(IN <IDX_DIAG>, OUT <LV_CDN_DIAG>)
Action that returns the diagnostic condition
ACTION_ERRM_GetLvEndDiag(IN <IDX_DIAG>, OUT <LV_END_DIAG>)
Action that returns the status of the failure availability

A.67.1 General information:

Diagnosis of VP_PUT, which is an analogue input signal in the A/D-converter of the Microprocessor. Error-symptoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> SYM_0: Short circuit to battery "SCP"
 │  │  └───> SYM_1: Short circuit to Ground "SCG"
 │  └─────> SYM_2
 └───────> SYM_3
```

Remark: Depending on the Hardware, Open Circuit “OC” is assigned to SYM_0 or to SYM_1.

The error-management variables are initialized according to filter-type.

<u>Error debounce:</u>

- Debounce counter increment: C_ABC_INC_EL_PUT
- Debounce counter decrement: 1
- Debounce counter maximum value: C_ABC_MAX_EL_PUT

Function Description

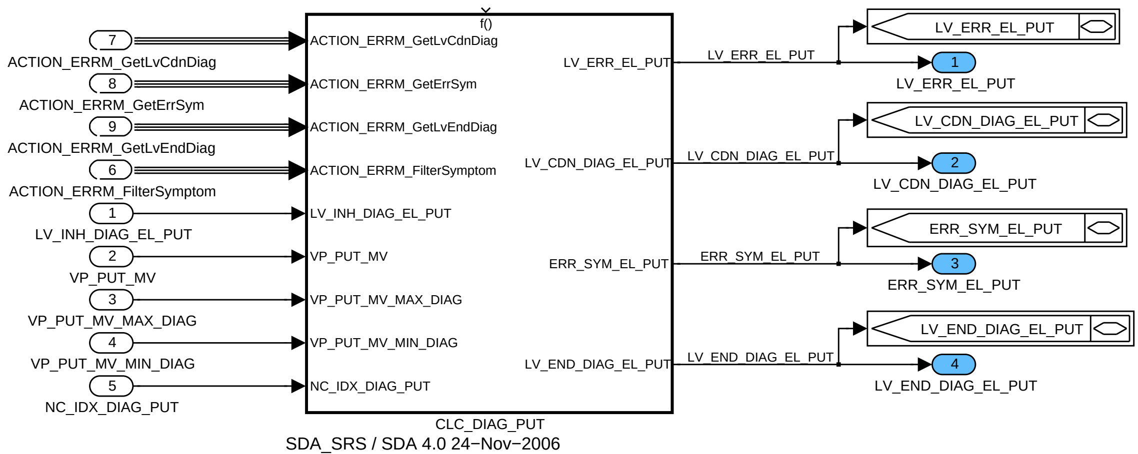


Figure 21 INSY_SIGDGPUT0

A.67.1.1 SUBFUNCTION: CLC_DIAG_PUT

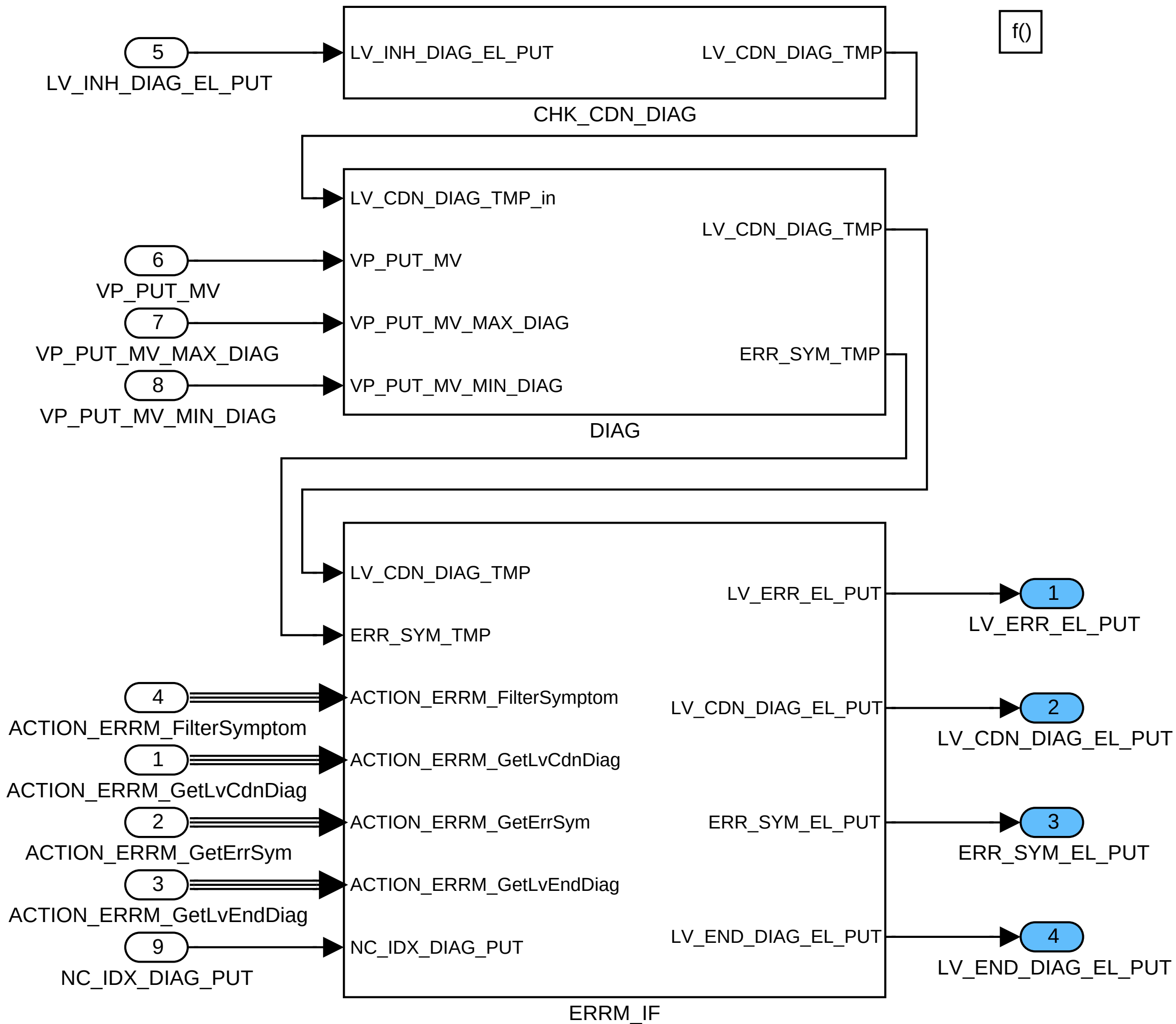


Figure 22 INSY_SIGDGPUT0/ CLC_DIAG_PUT

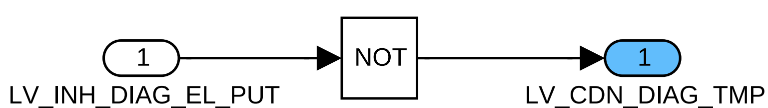


Figure 23 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ CHK_CDN_DIAG

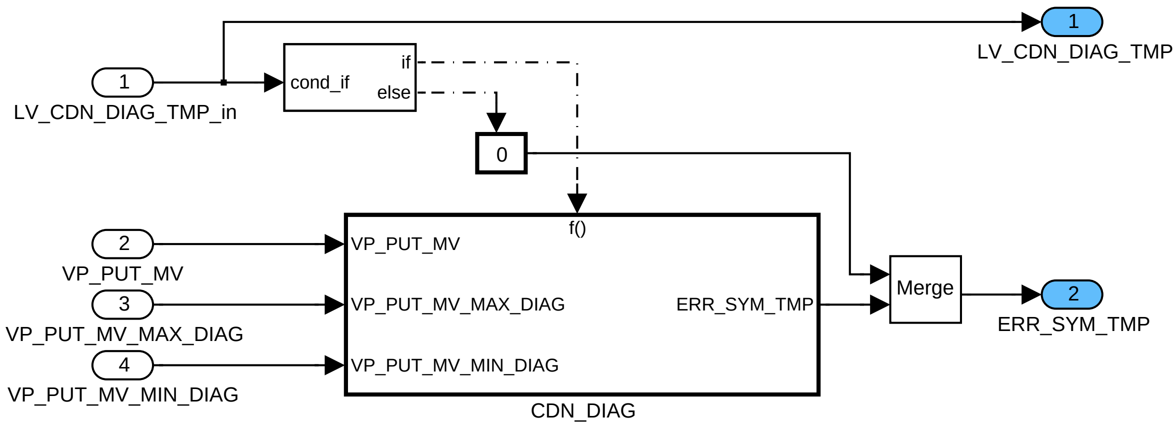


Figure 24 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ DIAG

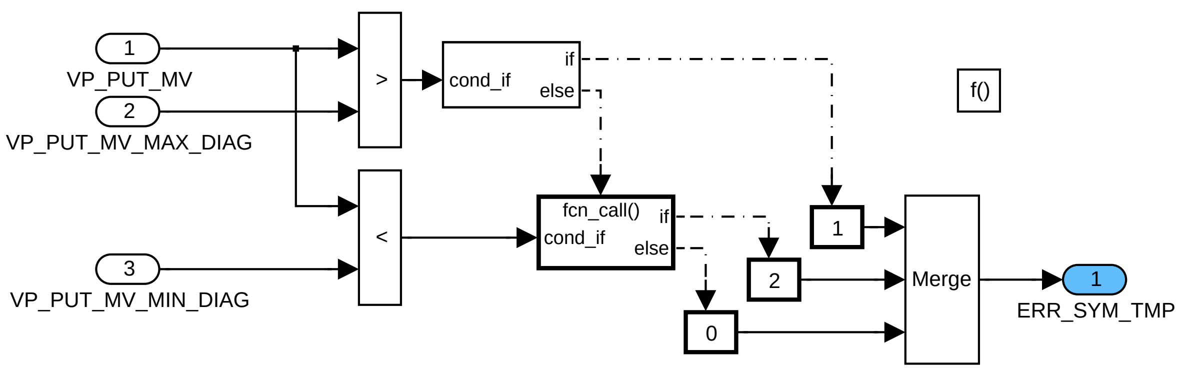


Figure 25 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ DIAG/ CDN_DIAG

Interface to ERRM:

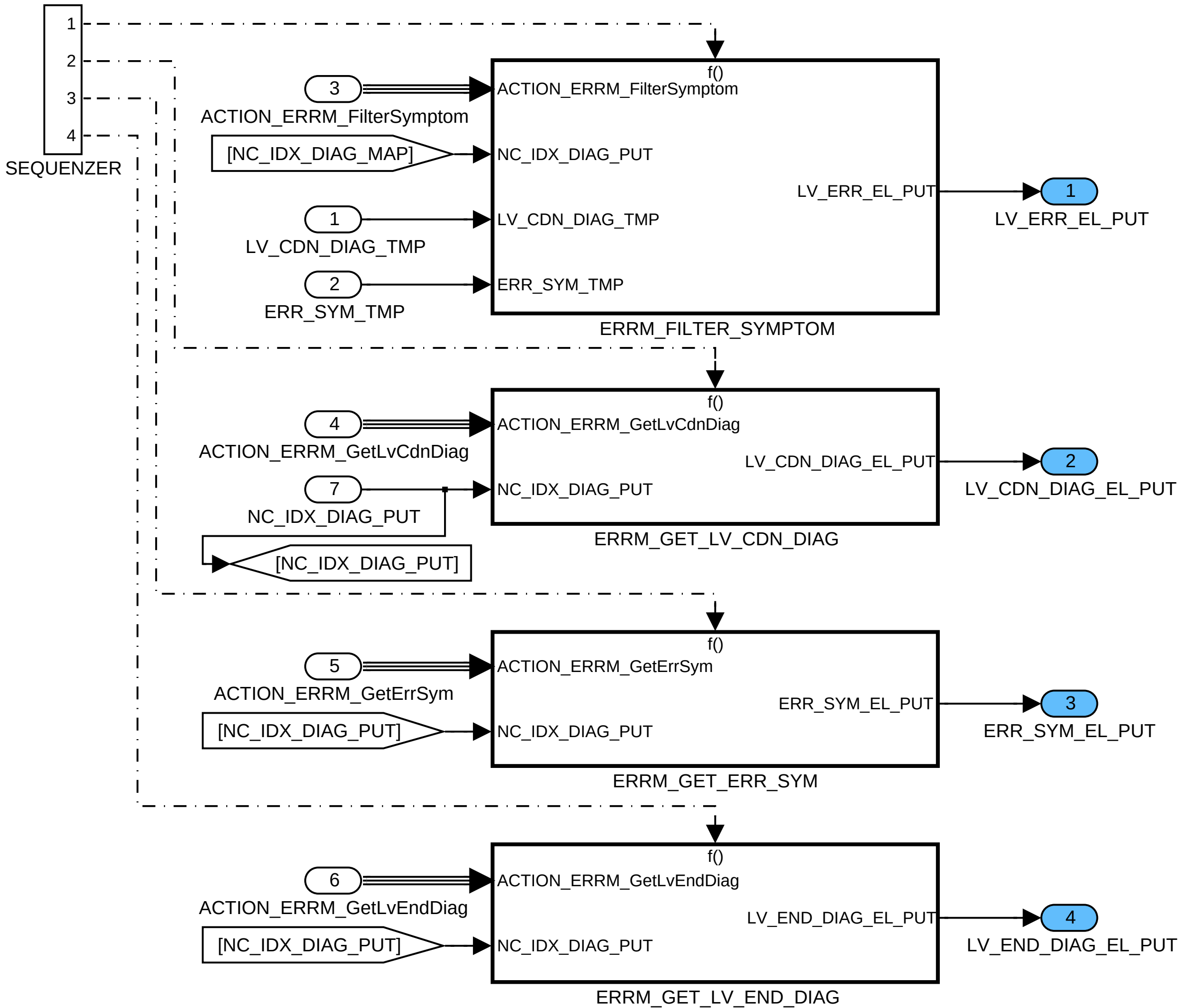


Figure 26 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ ERRM_IF

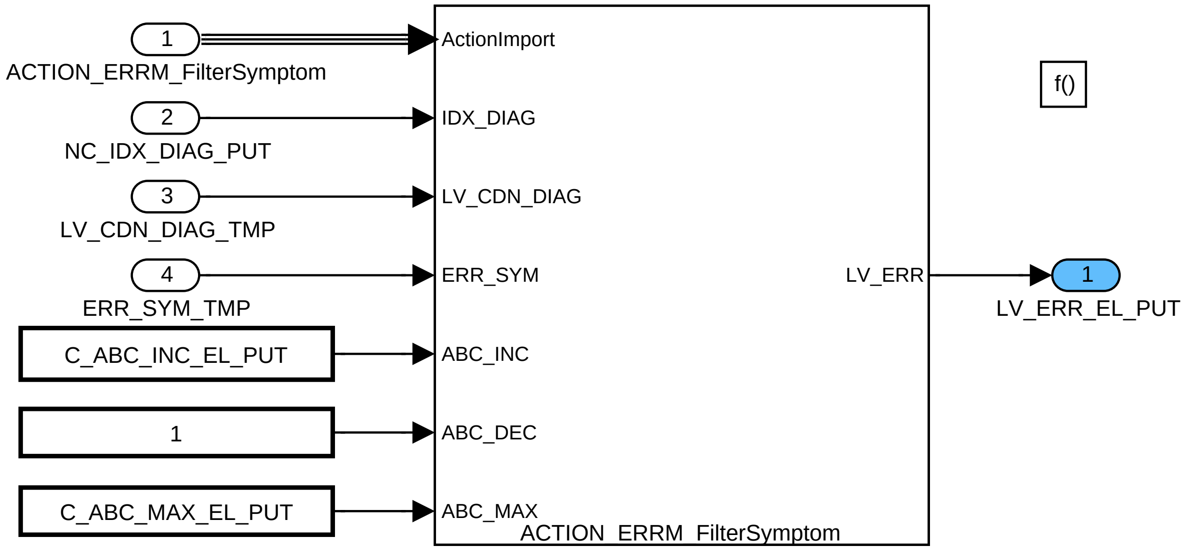


Figure 27 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ ERRM_IF/ ERRM_FILTER_SYMPTOM

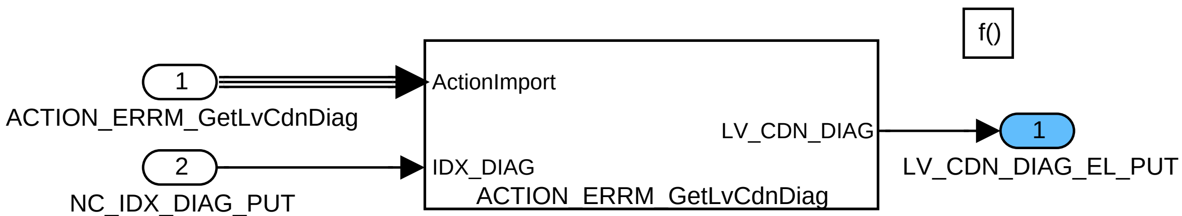


Figure 28 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ ERRM_IF/ ERRM_GET_LV_CDN_DIAG

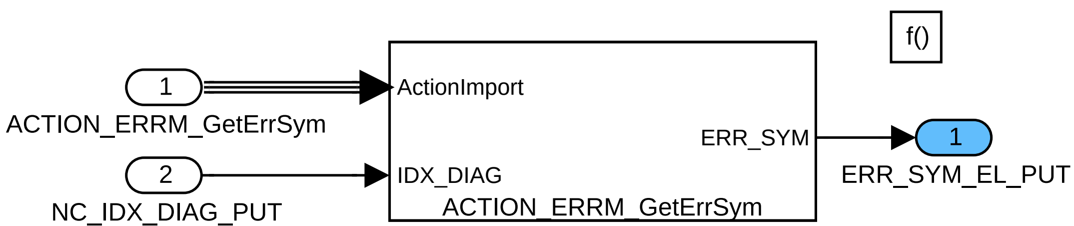


Figure 29 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ ERRM_IF/ ERRM_GET_ERR_SYM

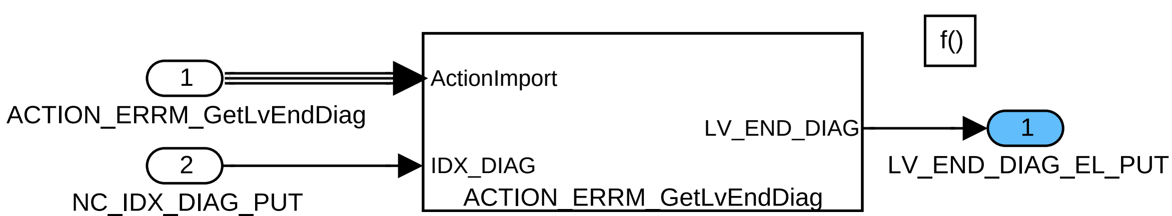


Figure 30 INSY_SIGDGPUT0/ CLC_DIAG_PUT/ ERRM_IF/ ERRM_GET_LV_END_DIAG

A.68 PUT sensor diagnosis (Appl. Inc.)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_ERR_PUT	O/V	0... 1H	0... 1	1	[-]
Summary flag of PUT sensor related errors					
LV_INH_DIAG_EL_PUT	O/V	0... 1H	0... 1	1	[-]
Inhibition of PUT diagnostics					
VP_PUT_MV_MAX_DIAG	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Threshold value of VP_PUT_MV to detect Short circuit to battery					
VP_PUT_MV_MAX_DIAG_ES	O	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Maximum Voltage of the pressure upstream throttle in Engines Stop					
VP_PUT_MV_MIN_DIAG	O/V	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Threshold value of VP_PUT_MV to detect Short circuit to ground					
VP_PUT_MV_MIN_DIAG_ES	O	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Minimum Voltage of the pressure upstream throttle in Engines Stop					

Input Data:

LV_PUT_SWI	LV_IGK	NC_PUT_CONF	LV_ERR_VCC_SENS_SUB
LV_ERR_EL_PUT	LV_ERR_PUT_PLAUS	LV_CDN_VB_OBD1	

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_VP_PUT_MV_MAX_DIAG	1	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Threshold value VP_PUT_MV to detect Short circuit to battery					
C_VP_PUT_MV_MIN_DIAG	1	0... 7FFFH	0... 4.99984741211	152.588e-6	[V]
Threshold value VP_PUT_MV to detect Short circuit to ground					

General Information

Configuration data:

Diagnosis	Symptom	Nr	ABC type
Pressure upstream throttle sensor diagnosis with short circuit to battery	Short circuit to battery“SCP”	0	STD_INI
Pressure upstream throttle sensor diagnosis with short circuit to ground	Short circuit to Ground "SCG" or open circuit	1	STD_INI

Application Conditions

Initialization: RST, CLRFMY, IGKON

general specification

Recurrence: 10MS
Activation: always
Deactivation: never

Function description

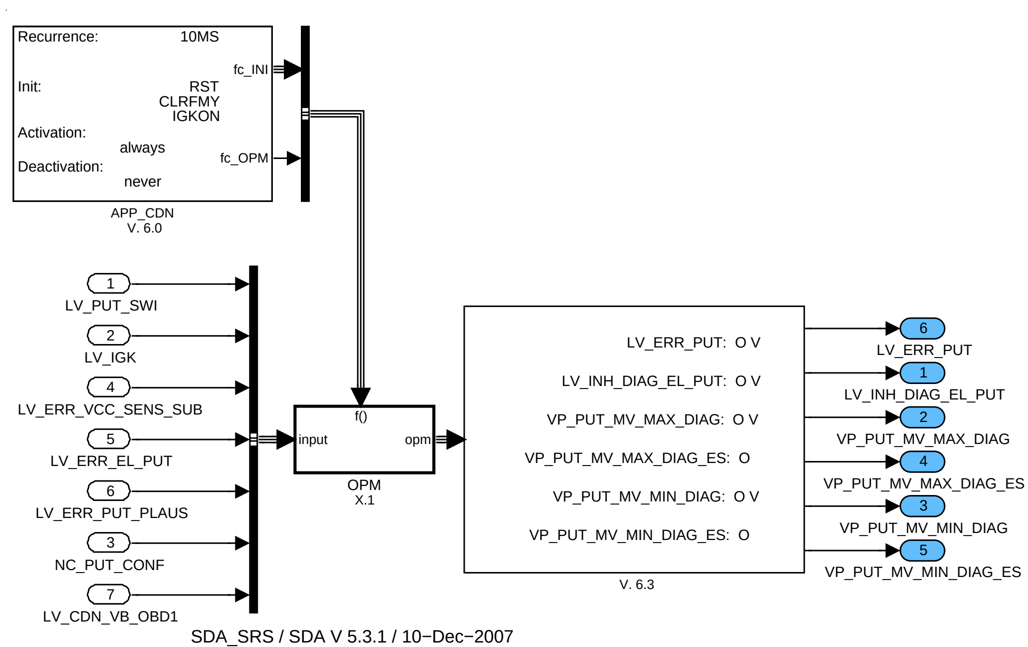


Figure 31:

A.68.1 operate_10ms:

Remark:

For configurations without PUT sensor (NC_PUT_CONF = 0) all outputs are calculated only once at reset and set to 0 and then never changed.

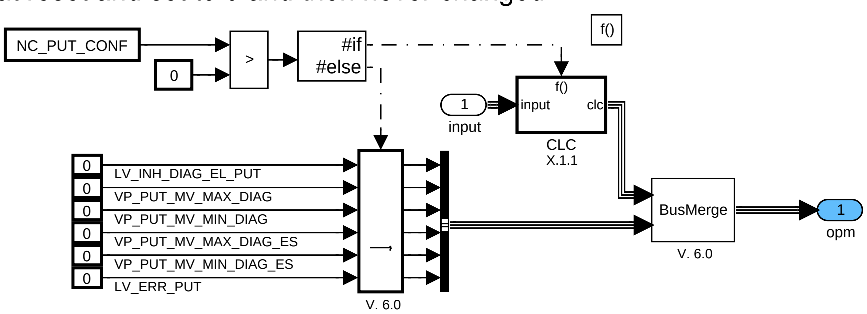


Figure 32:

A.68.1.1 Calculation:

A.68.1.1.1 Calculation of diagnosis thresholds:

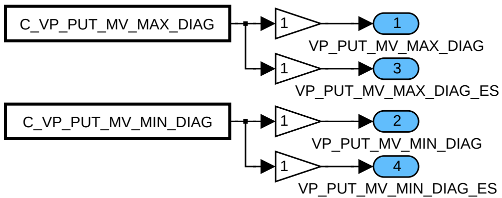


Figure 33:

A.68.1.1.2 Inhibition:

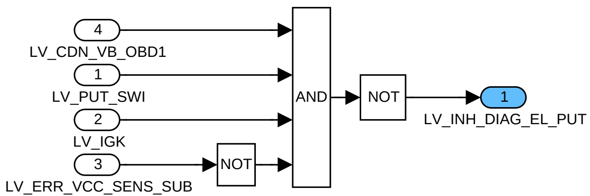


Figure 34:

A.68.1.1.3 Summary flag of PUT related errors:

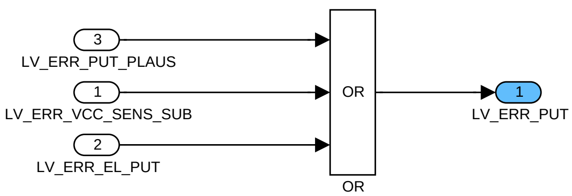


Figure 35:

A.69 MAP sensor diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VP_MAP_MAX_DIAG	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value of VP_MAP_MV to detect Short circuit in signal wire to VB					
VP_MAP_MIN_DIAG	O/V	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value of VP_MAP_MV to detect Short circuit in signal wire to earth or wire break					
LV_INH_DIAG_EL_MAP	O/V	0...1H	0...1	1	-
Inhibition condition for the MAP diagnosis					
LV_ERR_MAP	O/V	0...1H	0...1	1	-
Summary flag of MAP sensor related errors					

Input data:

LV_MAP_SWI	LV_ERR_VCC_MAP	LV_IGK	LV_ERR_EL_MAP
LV_ERR_MAP_PLAUS	LV_CDN_VB_MIN_DIAG		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VP_MAP_MAX_DIAG	1	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value VP_MAP_MV to detect Short circuit in signal wire to VB					
C_VP_MAP_MIN_DIAG	1	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value VP_MAP_MV to detect Short circuit in signal wire to earth or wire break					

A.69.1 General Information:

Definition of the thresholds to detect either short circuit in signal wire to battery or short circuit in signal wire to ground or wire break:

Short circuit in signal wire to VB

VP_MAP_MAX_DIAG = C_VP_MAP_MAX_DIAG

Short circuit in signal wire to ground or wire break

VP_MAP_MIN_DIAG = C_VP_MAP_MIN_DIAG

Inhibition:

Depending on projects specific requirements, the manifold air pressure sensor diagnosis needs can be inhibited by setting LV_INH_DIAG_EL_MAP = 1.

LV_INH_DIAG_EL_MAP is a bit which contains the project- specific activation/deactivation condition.

Application Condition

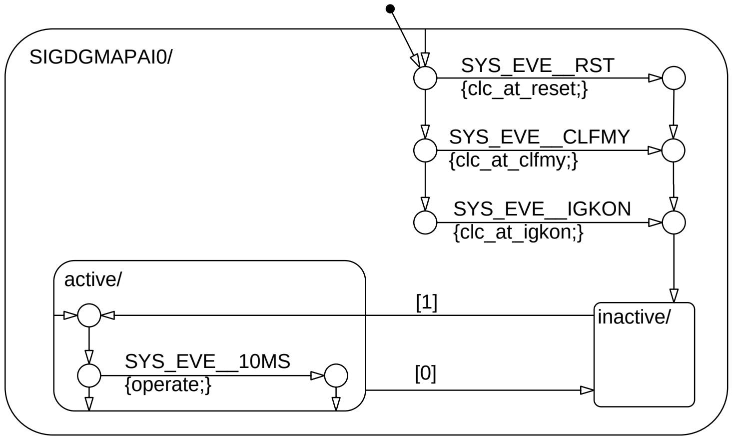


Figure 36 INSY_SIGDGMAPAI0/ APP_CDN/ Chart

Function Description

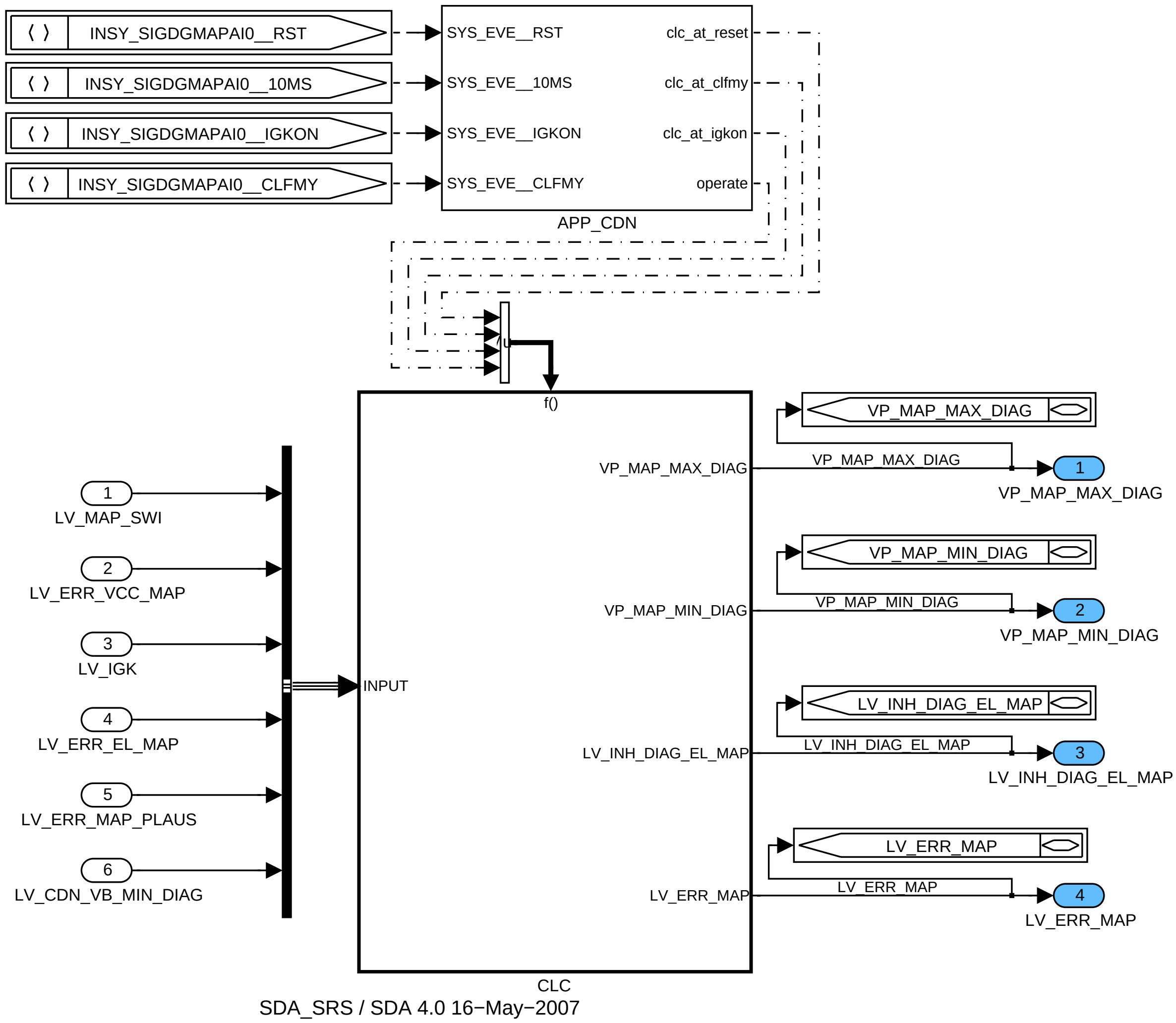
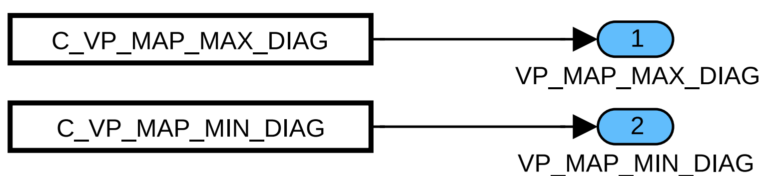


Figure 37 INSY_SIGDGMAPAI0



SUBFUNCTION: CLC

Figure 38 INSY_SIGDGMAPAI0/ CLC/ CLC_THD

Inhibition:

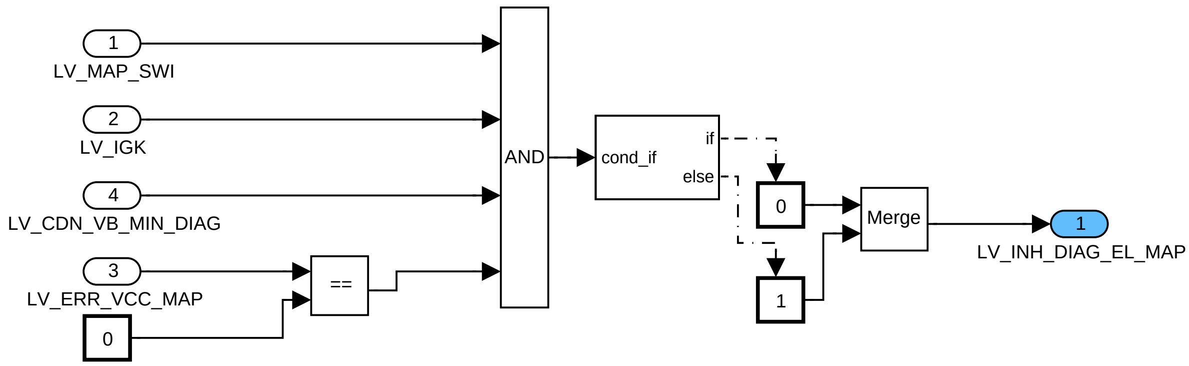


Figure 39 INSY_SIGDGMAPAI0/ CLC/ CLC_LV_INH_DIAG_EL_MAP

Summary flag of MAP sensor related errors:

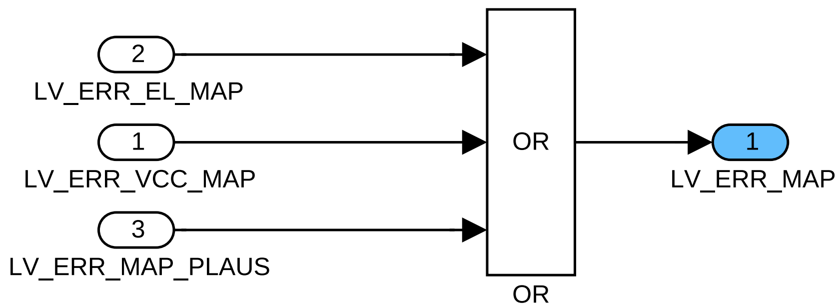


Figure 40 INSY_SIGDGMAPAI0/ CLC/ CLC_LV_ERR_MAP

A.70 Mass Air Flow Sensor Diagnosis (MAF)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MAF_SCP	O/V	0...1H	0...1	1	-
Short circuit in signal wire to VB detected					
ERR_SYM_MAF_SCP	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom short circuit in signal wire to VB					
LV_ERR_MAF_SCG_OC	O/V	0...1H	0...1	1	-
Short circuit in signal wire to earth or wire break (SCG or OC) detected					
ERR_SYM_MAF_SCG_OC	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom Short circuit in signal wire to earth or wire break (SCG or LB)					
LV_ERR_MAF	O/V	0...1H	0...1	1	-
MAF sensor error detected					
LV_CDN_DIAG_MAF_SCG_OC	V	0...1H	0...1	1	-
Status of diagnosis flag for Short circuit in signal wire to earth or wire break (SCG or LB) detected					
LV_CDN_DIAG_MAF_SCP	V	0...1H	0...1	1	-
Status of diagnosis flag for Short circuit in signal wire to VB detected					
LV_END_DIAG_MAF_SCP	V	0...1H	0...1	1	-
End of diagnosis flag for short circuit in signal wire to VB for mass air flow sensor diagnosis					
LV_END_DIAG_MAF_SCG_OC	V	0...1H	0...1	1	-
End of diagnosis flag for short circuit in signal wire to earth or wire break for mass air flow sensor diagnosis					

Input data:

VP_MAF	LV_INH_MAF_DIAG	NC_IDX_DIAG_MAF_SCP	N_32
TPS_AV	LV_INH_MAF_SCG_DIAG	LV_PUC	NC_IDX_DIAG_MAF_SCG_OC

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_MAF_SCG_OC	1	0...FFH	0...255	1	-
Anti-bounce increment value for SCG_OC					
C_ABC_INC_MAF_SCP	1	0...FFH	0...255	1	-
Anti-bounce increment value for SCP					
C_ABC_MAX_MAF_SCG_OC	1	1...FFH	1...255	1	-
Anti-bounce maximum value for SCG_OC					
C_ABC_MAX_MAF_SCP	1	1...FFH	1...255	1	-
Anti-bounce maximum value for SCP					
C_N_MIN_MAF_DIAG	1	0...FFH	0...8.16E+3	32	rpm
Engine speed condition					
C_TPS_MIN_MAF_DIAG_SCG_OC	1	0...3FFFH	0...119.5	0.00729415	°TPS
Throttle position threshold for detection of mass air flow signal line shorted to ground or broken					
C_VP_MAF_MAX_DIAG	1	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value VP_MAF to detect Short circuit in signal wire to VB					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VP_MAF_MIN_DIAG	1	0...7FFFH	0...4.99984741	1.52588E-4	V
Threshold value VP_MAF to detect Short circuit in signal wire to earth or wire break					

Import actions:

ACTION_ERRM_FilterSymptom(IN <IDX_DIAG>, IN <LV_CDN_DIAG>, IN <ERR_SYM>, IN <ABC_INC>, IN <ABC_DEC>, IN <ABC_MAX>, OUT <LV_ERR>)
This action computes the elementary anti-bounce filter
ACTION_ERRM_GetErrSym(IN <IDX_DIAG>, OUT <ERR_SYM>)
Action that returns the symptom of the failure
ACTION_ERRM_GetLvErr(IN <IDX_DIAG>, OUT <LV_ERR>)
Action that returns the status of the debounced failure

A.70.1 General Information :

Analog input signal in the A/D-Input from the Microprocessor.

Description

Error - symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> Signal Line Short to Battery Voltage        (= SYM_0)
 │  │  └───> Signal Line Short to Ground or Open load     (= SYM_1)
 │  └─────> -                                             (= SYM_2)
 └───────> -                                              (= SYM_3)
```

Error debounce:

- Debounce counter increment: C_ABC_INC_MAF_SCP
- Debounce counter decrement: 1
- Debounce counter maximum value: C_ABC_MAX_MAF_SCP

- Debounce counter increment: C_ABC_INC_MAF_SCG_OC
- Debounce counter decrement: 1
- Debounce counter maximum value: C_ABC_MAX_MAF_SCG_OC

Function Description

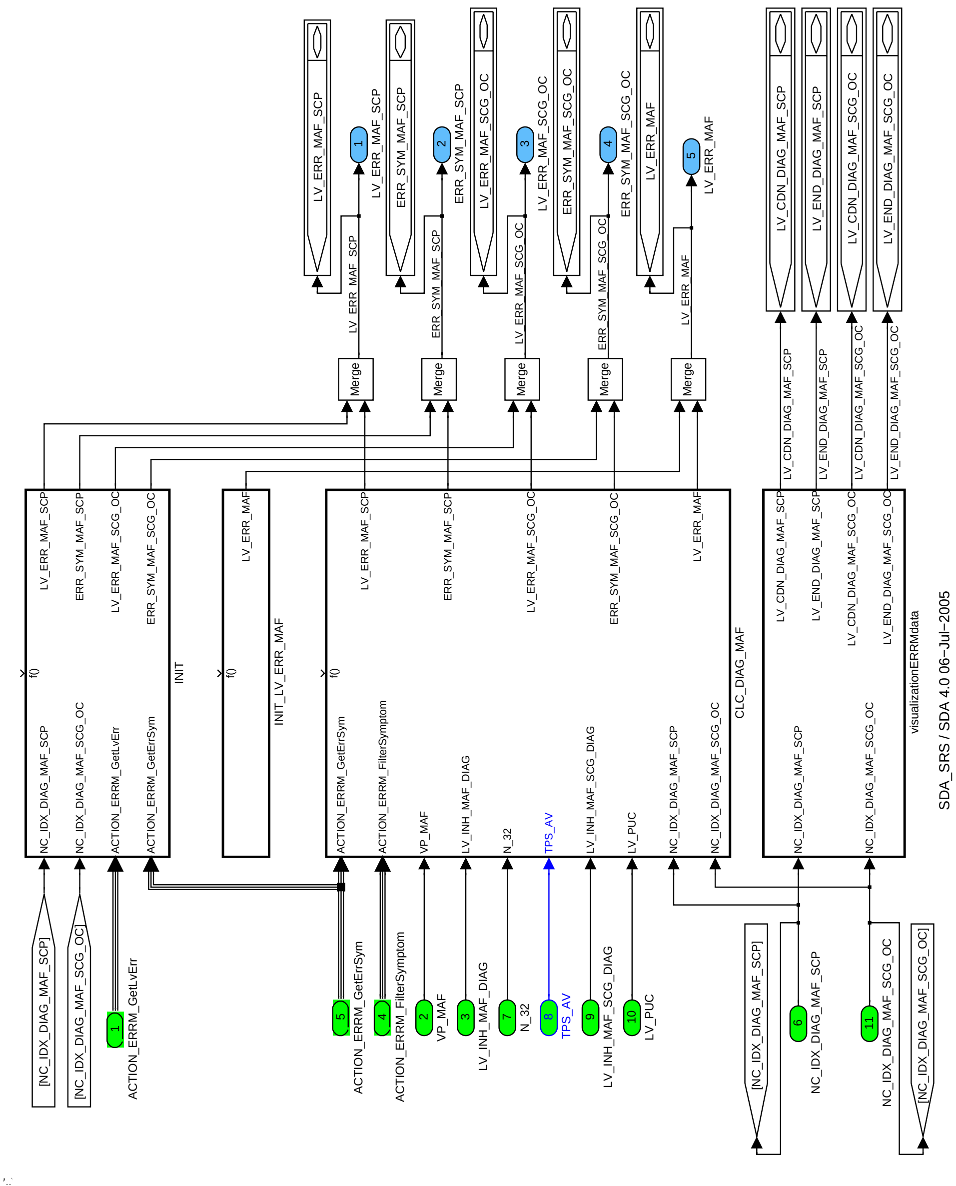


Figure 41 INSY_SIG_DMAF0

A.70.1.1 Initialization:

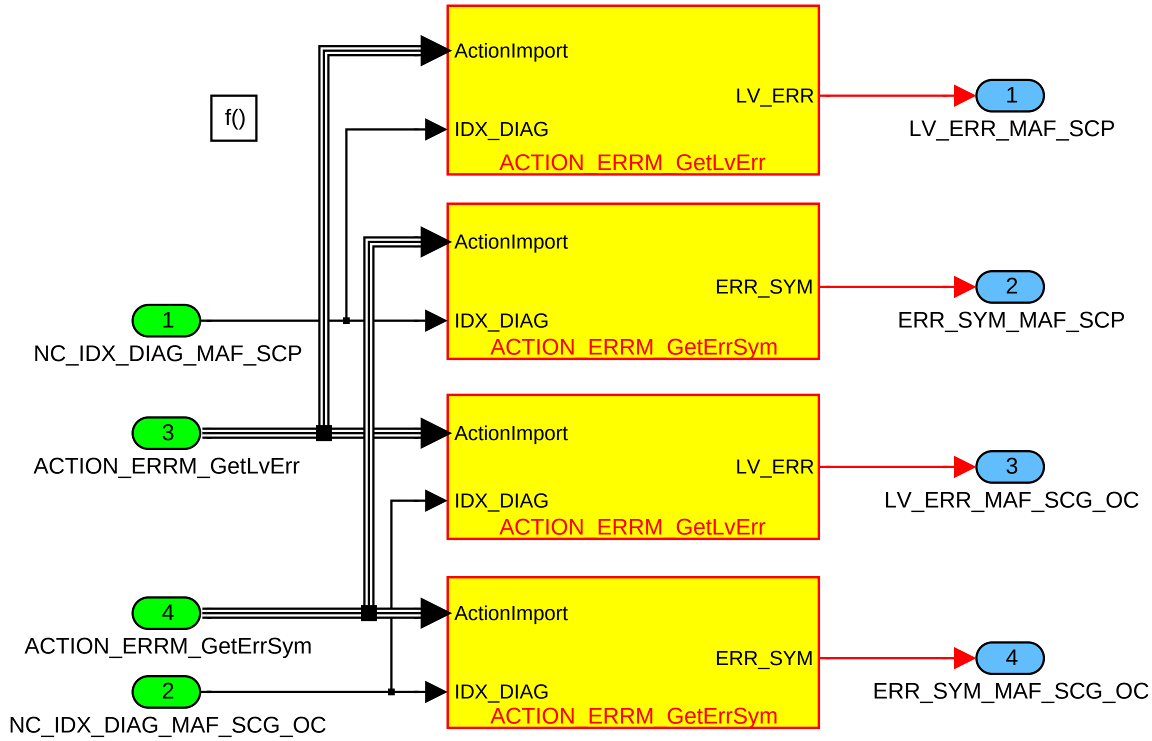


Figure 42 INSY_SIG_DMAF0/ INIT

A.70.1.2 SUBFUNCTION: INIT_LV_ERR_MAF

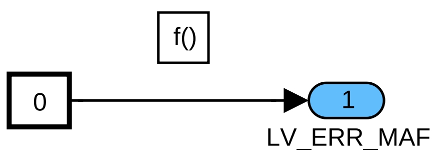


Figure 43 INSY_SIG_DMAF0/ INIT_LV_ERR_MAF

A.70.1.3 SUBFUNCTION: CLC_DIAG_MAF

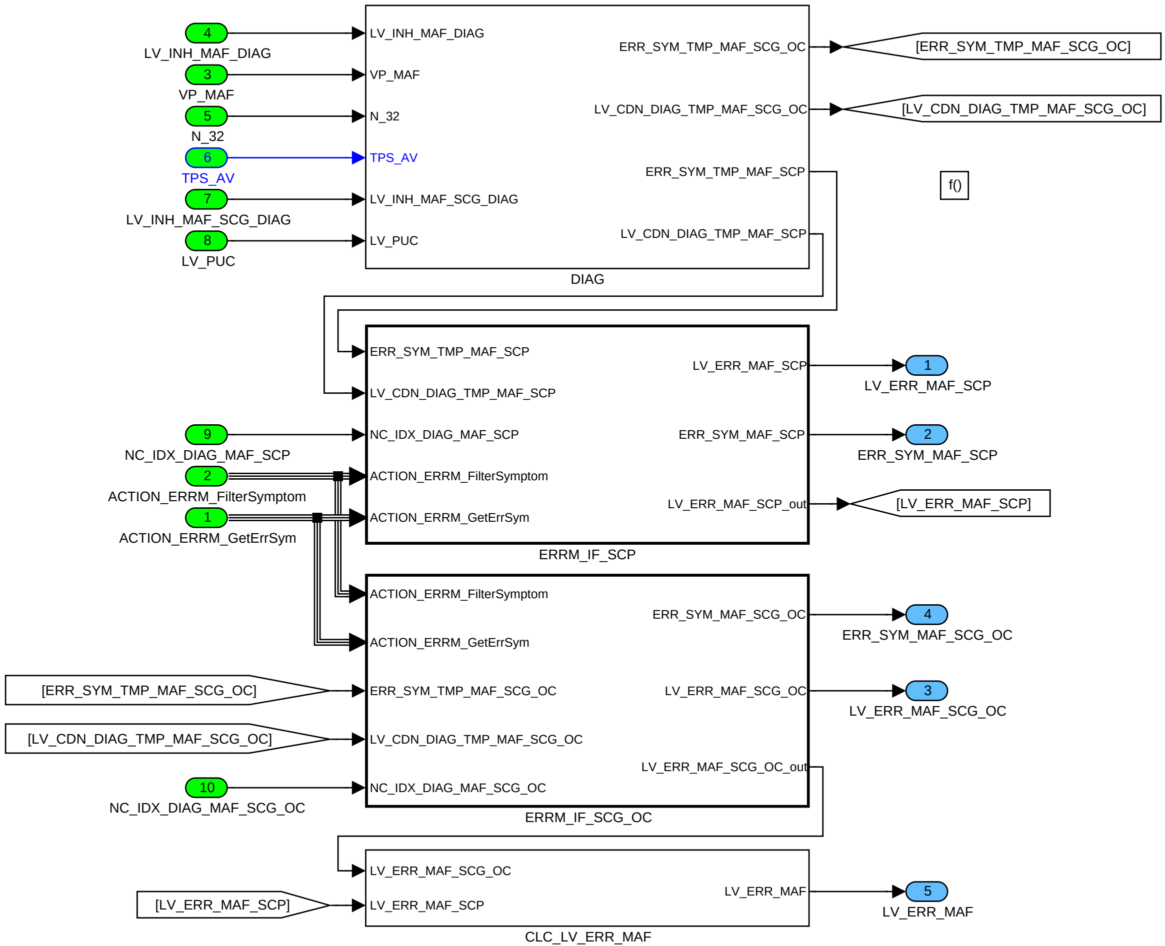


Figure 44 INSY_SIG_DMAF0/ CLC_DIAG_MAF

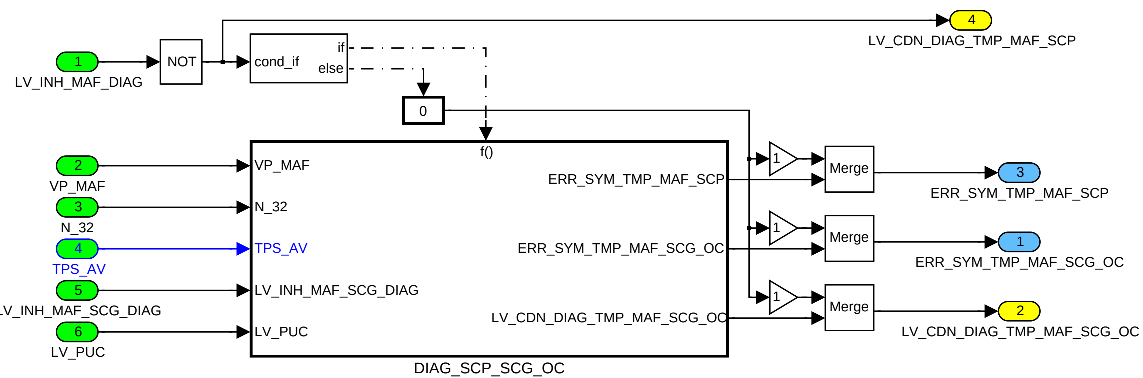


Figure 45 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ DIAG

Short circuit in signal wire to VB (KSU = SYM_0)

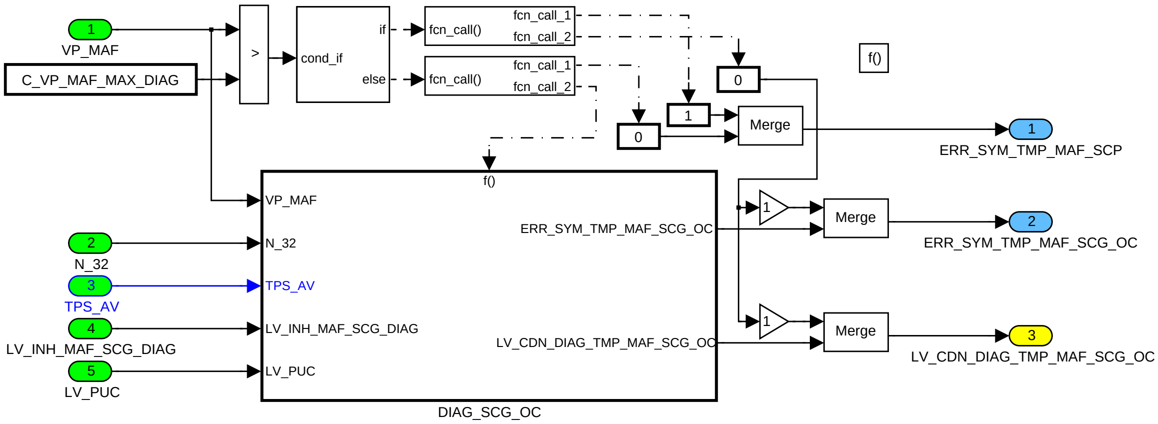


Figure 46 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ DIAG/ DIAG_SCP_SCG_OC

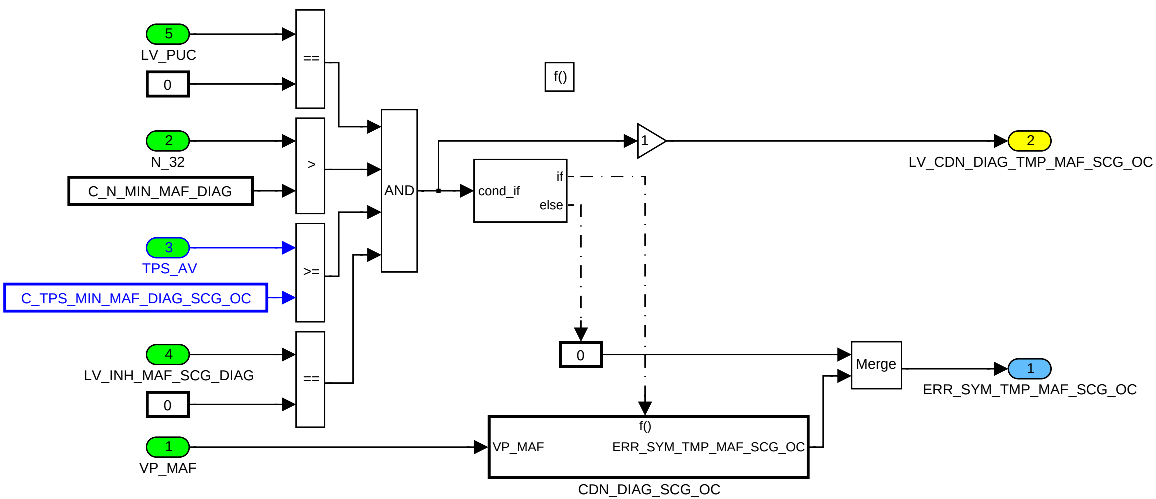


Figure 47 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ DIAG/ DIAG_SCP_SCG_OC/ DIAG_SCG_OC

Short circuit in signal wire to earth or wire break (KSM / UNT = SYM_1)

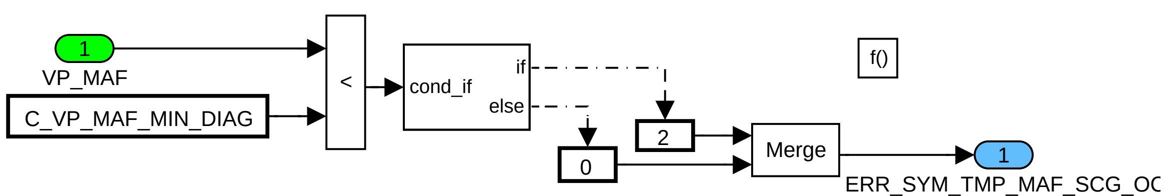


Figure 48 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ DIAG/ DIAG_SCP_SCG_OC/ DIAG_SCG_OC/ CDN_DIAG_SCG_OC

<u>Interface to ERRM (SCP):</u>

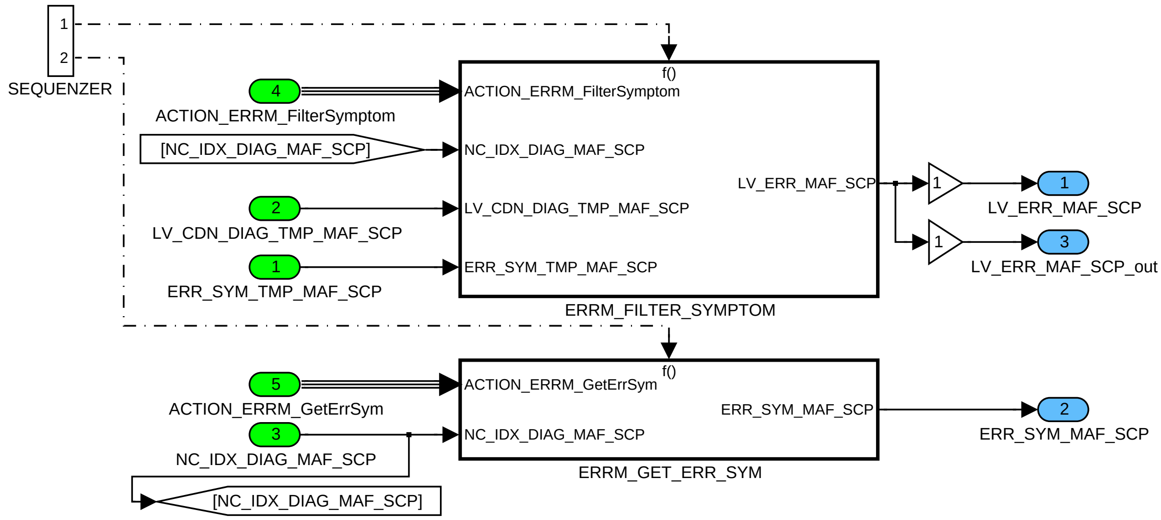


Figure 49 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCP

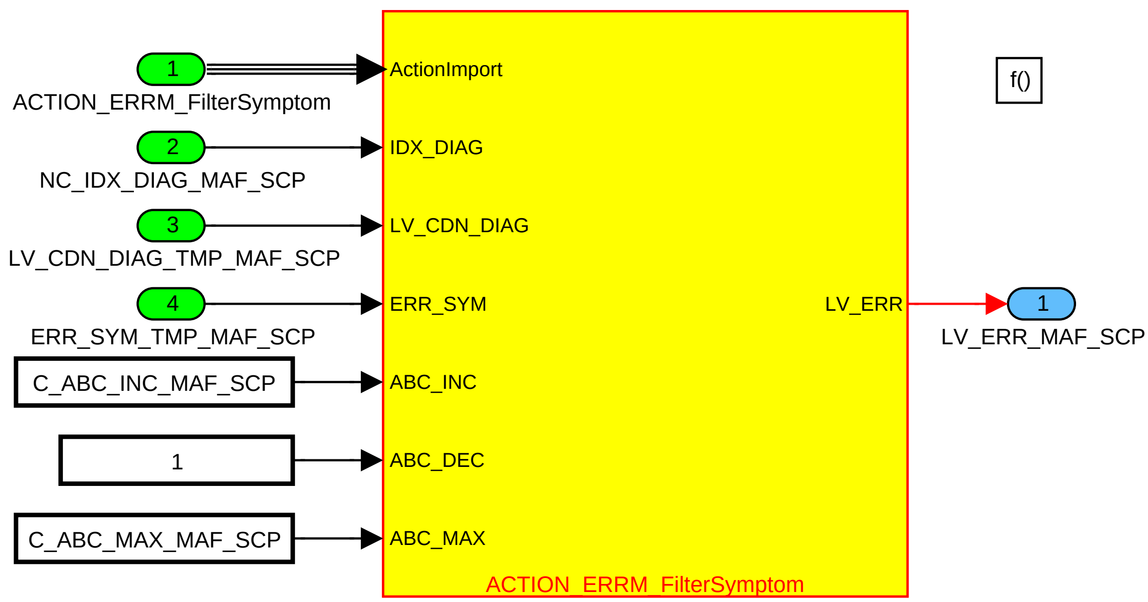


Figure 50 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCP/ ERRM_FILTER_SYMPTOM

Figure 51 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCP/ ERRM_GET_ERR_SYM

Interface to ERRM (SCG_OC):

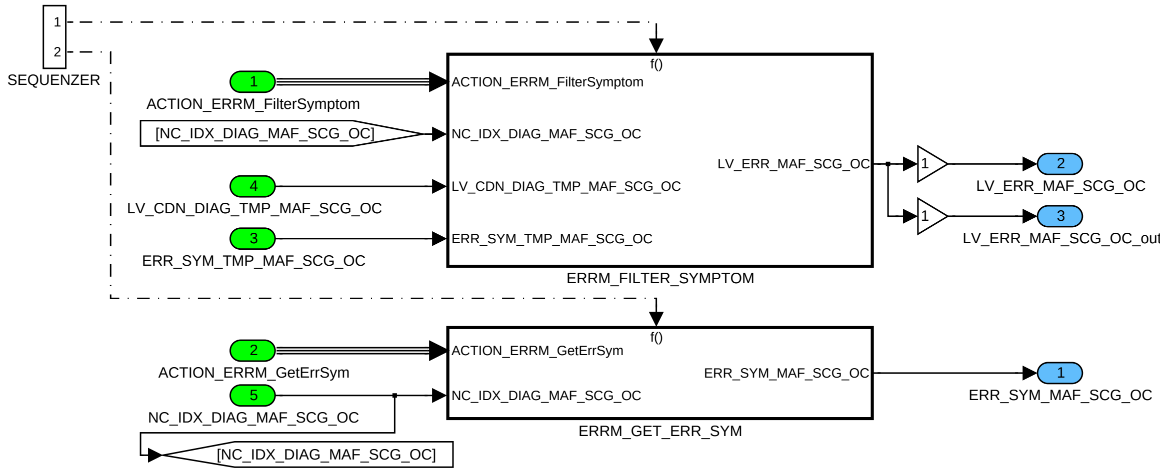


Figure 52 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCG_OC

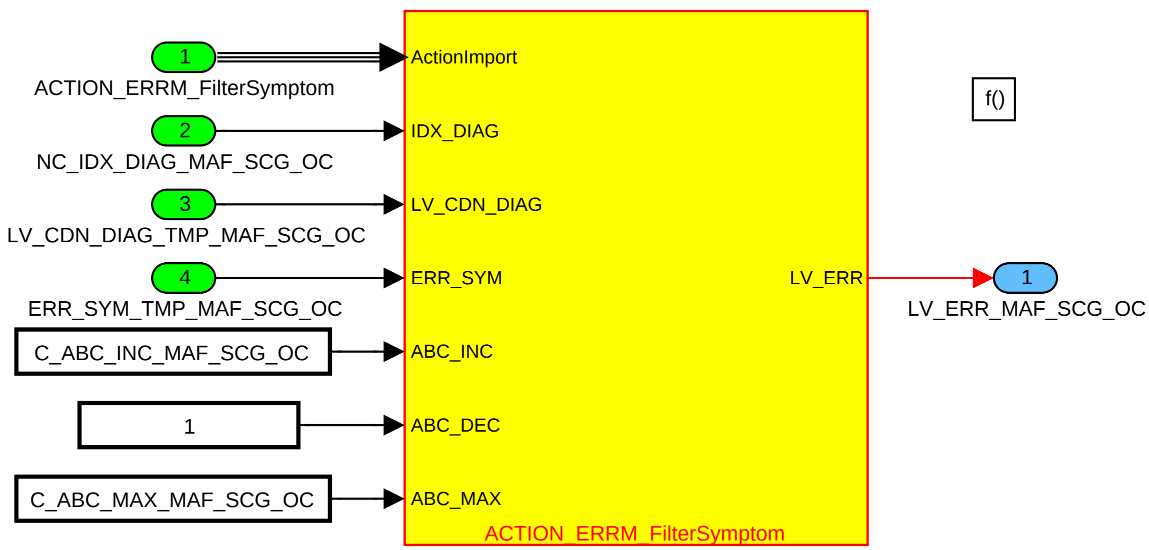


Figure 53 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCG_OC/ ERRM_FILTER_SYMPTOM

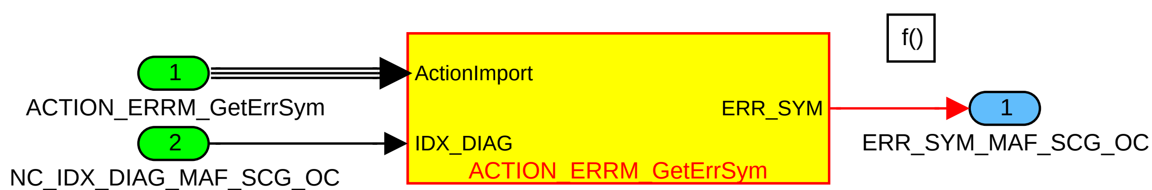


Figure 54 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ ERRM_IF_SCG_OC/ ERRM_GET_ERR_SYM

Remark:

The diagnosis result LV_ERR_MAF=0 is only valid if both diagnosis are finished. (see Antibounce specification).

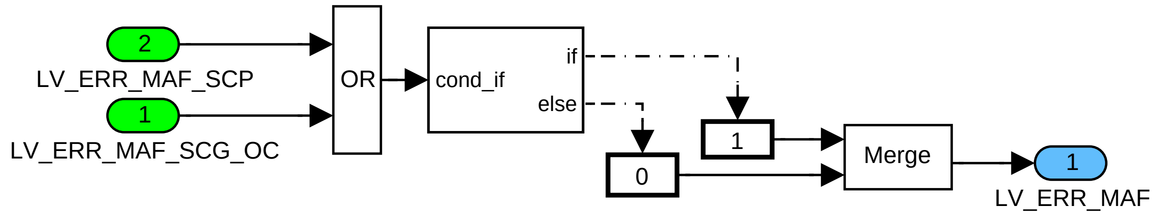


Figure 55 INSY_SIG_DMAF0/ CLC_DIAG_MAF/ CLC_LV_ERR_MAF

A.70.1.4 SUBFUNCTION: visualizationERRMdata

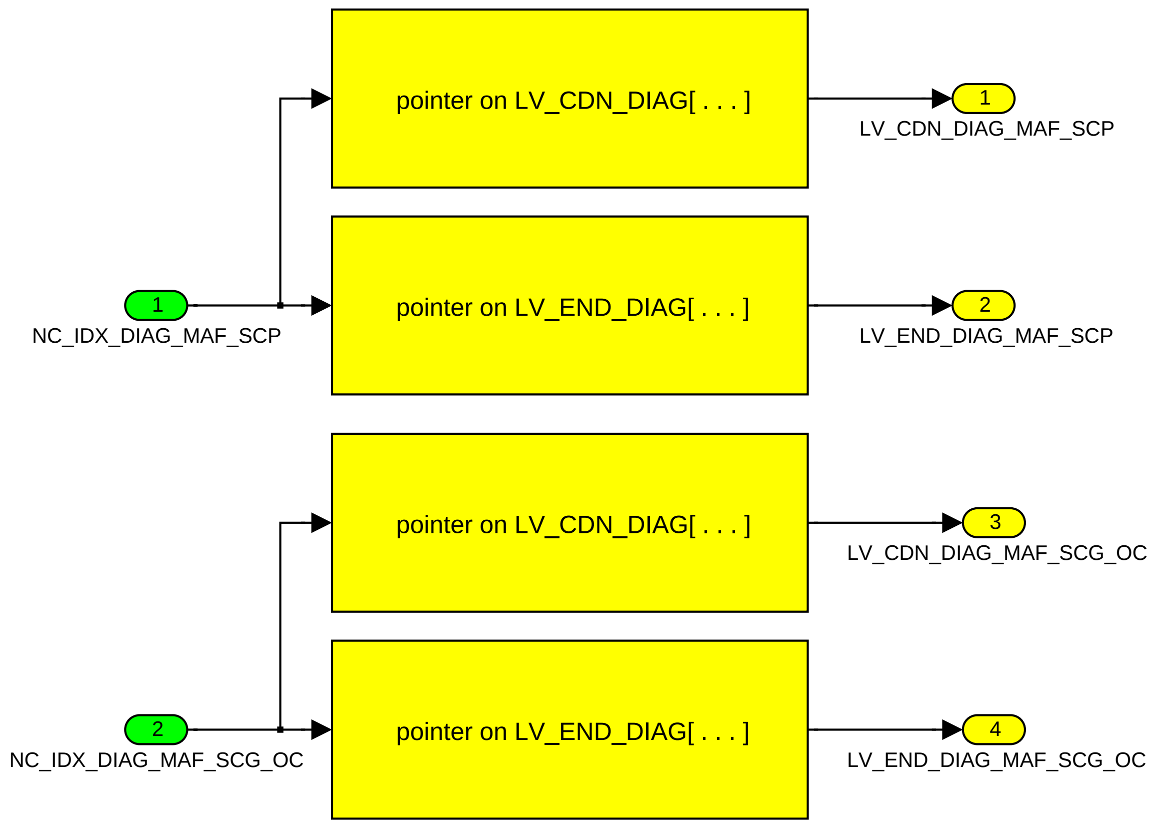


Figure 56 INSY_SIG_DMAF0/ visualizationERRMdata

A.71 Mass air flow sensor diagnosis (App. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_MAF_DIAG	O/V	0...1H	0...1	1	-
Flag to stop the Mass air flow sensor diagnosis					
LV_INH_MAF_SCG_DIAG	O/V	0...1H	0...1	1	-
Flag to inhibit the Mass air flow sensor diagnosis in case of SCG					
T_DLY_IGK_MAF_DIAG	-	0...FFH	0...2.55	0.01	s
Delay time after LV_IGK = 0 -> 1 for MAF diagnosis inhibition					

Input data:

LV_DC	LV_PWL	LV_ERR_RLY_EFP	LV_EFP
LV_MAF_SWI	T_AST	LV_CDN_VB_OBD1	LV_IGK
ERR_SYM_ISA_2	ERR_SYM_ISA_1		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_IGK_MAF_DIAG	1	0...FFH	0...2.55	0.01	s
Delay time after LV_IGK = 0 -> 1 for MAF diagnosis inhibition					
C_T_DLY_MAF_DIAG	1	0...FFH	0...25.5	0.1	s
Delay time after start for MAF diagnosis inhibition					

A.71.1 General information:

Depending on projects specific requirements, the mass air flow sensor diagnosis needs can be inhibited by setting LV_INH_MAF_DIAG = 1.

LV_INH_MAF_DIAG is a bit which contains the project-specific activation/deactivation condition.

The flag LV_INH_MAF_SCG_DIAG deactivates the diagnosis in case of short cut to ground (SCG) symptom appears.

Function Description

A.71.1.1 Initialization:

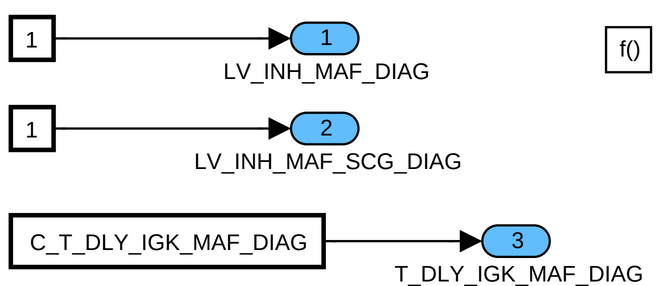


Figure 57 INSY_SIGDGMAFAI0/ INIT

A.71.1.2 Inhibition of the mass air flow sensor diagnosis:

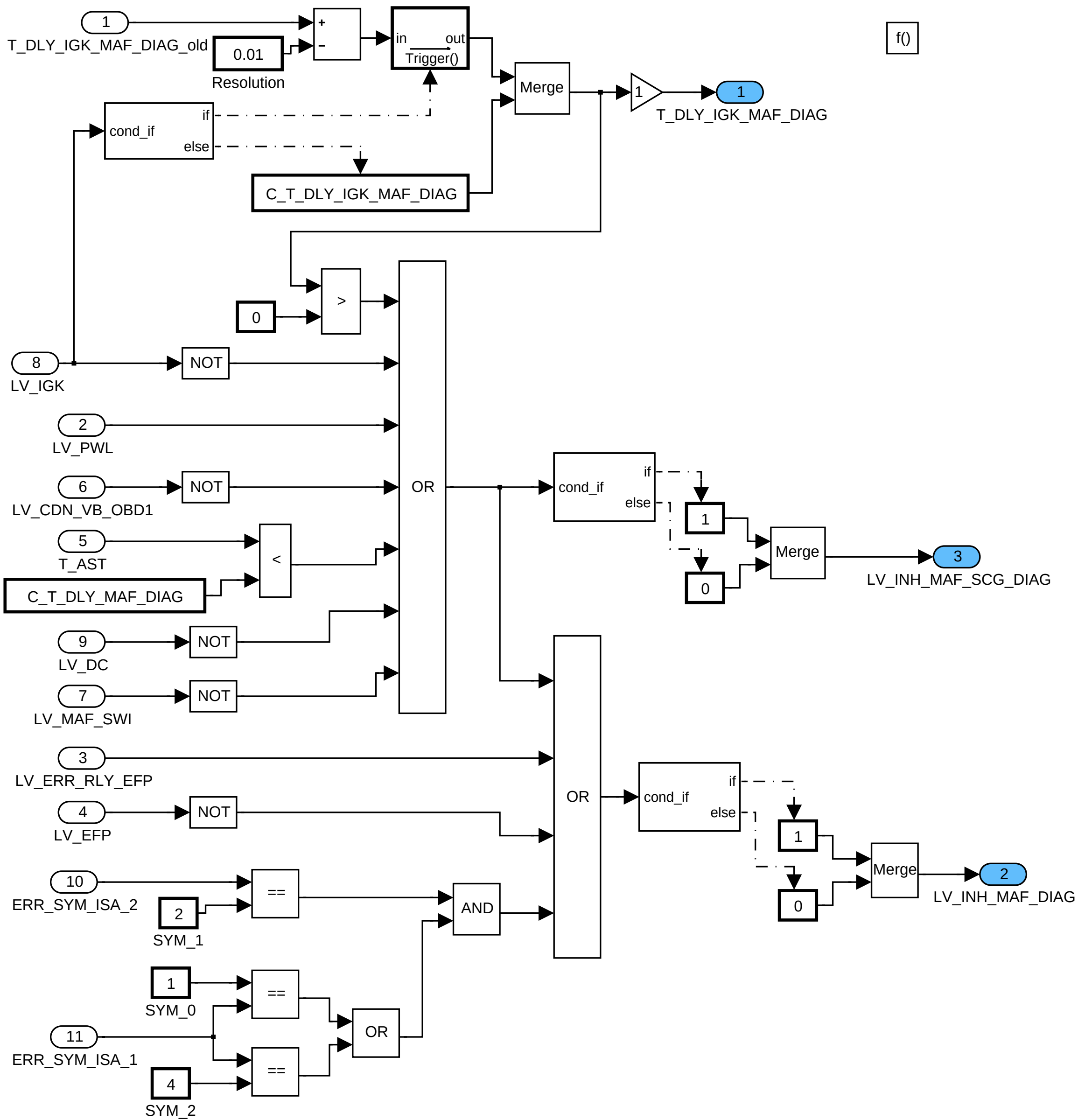


Figure 58 INSY_SIGDGMAFAI0/ CLC_10ms

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Mass air flow sensor diagnosis short circuit to GND or Line break		SYM_0	*STD_INI*
	SCG + OL	SYM_1	
		SYM_2	
MAF_SCG_OC		SYM_3	

List of Environmental Data to store in Failure Memory: for ISA: ISAPWM_ISA_ENVD
TPS
TIA_MES
MAF_MES_ENVD
for ETC: TPS_SP_ENVD
TPS
TIA_MES
MAF_MES_ENVD

Mass air flow sensor diagnosis short circuit to battery	SCB	SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
MAF_SCP		SYM_3	

List of Environmental Data to store in Failure Memory: for ISA: ISAPWM_ISA_ENVD
TPS
TIA_MES
MAF_MES_ENVD
for ETC: TPS_SP_ENVD
TPS
TIA_MES
MAF_MES_ENVD

A.71.2 Application assistances

- **Emergency operation:**

- with an ABC-value unequal to 0 the intake manifold model will not be corrected anymore via the MAF sensor signal to avoid engine stall during the debouncing phase (before final error detection)

- **Application Assistance:**

Limp home functions:
- TPS limp home (calculation modified)
- Engine speed limitation
- Set state torque intervention to max. state (=3)
- EVAP-control activation of MIN-operation

The following control functions are inhibited:
- Downstream Fuel Trim Regulation
- Catalyst Heating
- Torque intervention for gear shift

The following adaptation functions are inhibited:
- Lambda adaptation
- Knock adaptation
- Idle speed adaptation
- Reduced area adaptation learning (indirect)
- Catalyst heating adaptation (indirect)
- Segment time adaptation
- Ambient pressure adaptation

The following diagnosis functions are inhibited:
- Ignition diagnosis
- ISA-plausibility diagnosis
- O2-sensor downstream signal voltage diagnosis (only SCG)
- Differential tank pressure sensor plausibility diagnosis (Noisy signal)
- Differential tank pressure sensor plausibility diagnosis (Signal constant)
- load/TPS-plausibility diagnosis
- Misfire detection (ER-calculation)
- EVAP-diagnosis
- Fuel system diagnosis
- O2-sensor upstream switching time diagnosis
- O2-sensor upstream frequency diagnosis
- Catalyst efficiency diagnosis
- O2-sensor downstream switching time diagnosis
- O2-sensor downstream OBDII puc-end diagnosis
- Thermosthat diagnosis

A.72 Reference voltage diagnosis (VCC_PVS_1_DIAG , VCC_PVS_2_DIAG , TPS_VCC_DIAG and VCC_SENS_SUB_DIAG)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VCC_PVS_1_DIAG	V/O	0...03FFH	0...9.9902	0.0098	V
Supply Voltage for Diagnosis					
LV_ERR_VCC_PVS_1	V/O	0...1H	0...1	1	-
Failure state currently present on the reference voltage VCC_PVS_1					
ERR_SYM_VCC_PVS_1	V	0H 1H 2H 3H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
symptom: failure without filtering of diagnosis for VCC_PVS_1					
LV_CDN_DIAG_VCC_PVS_1	V	0...1H	0...1	1	-
Conditions of diagnosis fulfilled					
LV_END_DIAG_VCC_PVS_1	V	0...1H	0...1	1	-
End of diagnosis					
LV_INH_DIAG_VCC_PVS_1	V/O	0...1H	0...1	1	-
Inhibition of diagnosis					
VCC_PVS_2_DIAG	V/O	0...03FFH	0...9.9902	0.0098	V
Supply Voltage for Diagnosis					
LV_ERR_VCC_PVS_2	V/O	0...1H	0...1	1	-
Failure state currently present on the reference voltage VCC_PVS_2					
ERR_SYM_VCC_PVS_2	V	0H 1H 2H 3H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
symptom: failure without filtering of diagnosis for VCC_PVS_2					
LV_CDN_DIAG_VCC_PVS_2	V	0...1H	0...1	1	-
Conditions of diagnosis fulfilled					
LV_END_DIAG_VCC_PVS_2	V	0...1H	0...1	1	-
End of diagnosis					
LV_INH_DIAG_VCC_PVS_2	V/O	0...1H	0...1	1	-
Inhibition of diagnosis					
VCC_TPS_DIAG	V/O	0...03FFH	0...9.9902	0.0098	V
Supply Voltage for Diagnosis					
LV_ERR_TPS_VCC	V/O	0...1H	0...1	1	-
Failure state currently present on the reference voltage VCC_TPS					
ERR_SYM_TPS_VCC	V	0H 1H 2H 3H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
symptom: failure without filtering of diagnosis for VCC_TPS					
LV_CDN_DIAG_TPS_VCC	V	0...1H	0...1	1	-
Conditions of diagnosis fulfilled					
LV_END_DIAG_TPS_VCC	V	0...1H	0...1	1	-
End of diagnosis					
LV_INH_DIAG_TPS_VCC	V/O	0...1H	0...1	1	-
Inhibition of diagnosis					
VCC_SENS_SUB_DIAG	V/O	0...03FFH	0...9.9902	0.0098	V
Supply Voltage for Diagnosis					
LV_ERR_VCC_SENS_SUB	V/O	0...1H	0...1	1	-
Failure state currently present on the reference voltage VCC_SENS_SUB					
ERR_SYM_VCC_SENS_SUB	V	0H 1H 2H 3H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
symptom: failure without filtering of diagnosis for VCC_SENS_SUB					
LV_CDN_DIAG_VCC_SENS_SUB	V	0...1H	0...1	1	-

Conditions of diagnosis fulfilled					
LV_END_DIAG_VCC_SENS_SUB	V	0...1H	0...1	1	-
End of diagnosis					
LV_INH_DIAG_VCC_SENS_SUB	V/O	0...1H	0...1	1	-
Inhibition of diagnosis					
LV_ERR_VCC	V/O	0...1H	0...1	1	-
Failure state currently present on 1 of the 4 reference voltages VCC_PVS_1 or VCC_PVS_2 or VCC_TPS or VCC_SENS_SUB					

Input data:

VCC_PVS1_DIAG_BAS	VCC_PVS2_DIAG_BAS	VCC_TPS_DIAG_BAS	VCC_SENS_SUB_DIAG_BAS
LV_RLY_MAIN_EXT_ADJ	C_T_DLY_RLY_MAIN_DIAG	LV_IGK	

FUNCTION DESCRIPTION:

General information:

If the supply voltage is invalid, the sensors depending on it would deliver wrong values and could be detected as faulty. Therefore the reference voltages are acquired from the Hardware I/O System and monitored. The diagnosis voltage delivered from HW is half of the real supply voltage.

Acquisition of Diagnosis Voltage :

VCC_PVS_1_DIAG = VCC_PVS1_DIAG_BAS * 2

VCC_PVS_2_DIAG = VCC_PVS2_DIAG_BAS * 2

VCC_TPS_DIAG = VCC_TPS_DIAG_BAS * 2

VCC_SENS_SUB_DIAG = VCC_SENS_SUB_DIAG_BAS * 2

Application conditions:

Initialisation: *at the transition LV_IGK = 0 to 1, at reset event and clear error memory*

LV_INH_DIAG_VCC_PVS_1 = 0

LV_INH_DIAG_VCC_PVS_2 = 0

LV_INH_DIAG_TPS_VCC = 0

LV_INH_DIAG_VCC_SENS_SUB = 0

LV_ERR_VCC_PVS_1 = 0

LV_ERR_VCC_PVS_2 = 0

LV_ERR_TPS_VCC = 0

LV_ERR_VCC_SENS_SUB = 0

Activation: **If** LV_IGK = 1

And LV_INH_DIAG_VCC_PVS_1 = 0

Then LV_CDN_DIAG_VCC_PVS_1 = 1

Else LV_CDN_DIAG_VCC_PVS_1 = 0 (*Deactivation)*

If LV_IGK = 1
And LV_INH_DIAG_VCC_PVS_2 = 0
Then LV_CDN_DIAG_VCC_PVS_2 = 1
Else LV_CDN_DIAG_VCC_PVS_2 = 0 (*Deactivation)*

If LV_IGK = 1
And LV_INH_DIAG_TPS_VCC = 0
Then LV_CDN_DIAG_TPS_VCC = 1
Else LV_CDN_DIAG_TPS_VCC = 0 (*Deactivation)*

If LV_IGK = 1
And LV_INH_DIAG_VCC_SENS_SUB = 0
Then LV_CDN_DIAG_VCC_SENS_SUB = 1
Else LV_CDN_DIAG_ VCC_SENS_SUB = 0 (*Deactivation)*

For VCC_PVS_2 and TPS_VCC the inhibition bit is always set to 0 (LV_INH_DIAG_VCC_PVS_2 = 0 / LV_INH_DIAG_TPS_VCC = 0)

Recurrence: 10 ms

Formula section:

If C_T_DLY_RLY_MAIN_DIAG has run out after LV_RLY_MAIN_EXT_ADJ = 1-> 0
and LV_RLY_MAIN_EXT_ADJ = 0
Then LV_INH_DIAG_VCC_PVS_1 = 0
Else LV_INH_DIAG_VCC_PVS_1 = 1
Endif

(1) IF VCC_PVS_1_DIAG > C_VCC_PVS_1_MAX_DIAG
THEN *Voltage is too high ("SCP" active)*
ERR_SYM_VCC_PVS_1 = SYM_0
ELSE
(2) IF VCC_PVS_1_DIAG < C_VCC_PVS_1_MIN_DIAG
THEN *Voltage is too low ("SCG" active)*
ERR_SYM_VCC_PVS_1 = SYM_1
ELSE
(3) IF VCC_PVS_1_DIAG < C_VCC_PVS_1_MIN_DIAG_EXT
THEN *Voltage is too low (Sensor or wiring harness problem)*

ERR_SYM_VCC_PVS_1 = SYM_2

ELSE ERR_SYM_VCC_PVS_1 = NO_SYM

(3) ENDIF

(2) ENDIF

(1) ENDIF

The error bit LV_ERR_VCC_PVS_1 and the end of diagnosis bit LV_END_DIAG_VCC_PVS_1 are set by the failure memory.

(1) IF VCC_PVS_2_DIAG > C_VCC_PVS_2_MAX_DIAG

THEN *Voltage is too high ("SCP" active)*

ERR_SYM_VCC_PVS_2 = SYM_0

ELSE

(2) IF VCC_PVS_2_DIAG < C_VCC_PVS_2_MIN_DIAG

THEN *Voltage is too low ("SCG" active)*

ERR_SYM_VCC_PVS_2 = SYM_1

ELSE

(3) IF VCC_PVS_2_DIAG < C_VCC_PVS_2_MIN_DIAG_EXT

THEN *Voltage is too low (Sensor or wiring harness problem)*

ERR_SYM_VCC_PVS_2 = SYM_2

ELSE ERR_SYM_VCC_PVS_2 = NO_SYM

(3) ENDIF

(2) ENDIF

(1) ENDIF

The error bit LV_ERR_VCC_PVS_2 and the end of diagnosis bit LV_END_DIAG_VCC_PVS_2 are set by the failure memory

(1) IF VCC_TPS_DIAG > C_VCC_TPS_MAX_DIAG

THEN *Voltage is too high ("SCP" active)*

ERR_SYM_TPS_VCC = SYM_0

ELSE

(2) IF VCC_TPS_DIAG < C_VCC_TPS_MIN_DIAG

THEN *Voltage is too low ("SCG" active)*

ERR_SYM_TPS_VCC = SYM_1

ELSE

(3) IF VCC_TPS_DIAG < C_VCC_TPS_MIN_DIAG_EXT

THEN *Voltage is too low (Sensor or wiring harness problem)*

ERR_SYM_TPS_VCC = SYM_2

ELSE ERR_SYM_TPS_VCC = NO_SYM

(3) ENDIF

(2) ENDIF

(1) ENDIF

The error bit LV_ERR_TPS_VCC and the end of diagnosis bit LV_END_DIAG_TPS_VCC are set by the failure memory

(1) IF VCC_SENS_SUB_DIAG > C_VCC_SENS_SUB_MAX_DIAG

THEN *Voltage is too high ("SCP" active)*

ERR_SYM_VCC_SENS_SUB = SYM_0

ELSE

(2) IF VCC_SENS_SUB_DIAG < C_VCC_ SENS_SUB_MIN_DIAG

THEN *Voltage is too low ("SCG" active)*

ERR_SYM_VCC_SENS_SUB = SYM_1

ELSE

(3) IF VCC_ SENS_SUB_DIAG < C_VCC_SENS_SUB_MAX_DIAG_EXT

THEN *Voltage is too low (Sensor or wiring harness problem)*

ERR_SYM_VCC_SENS_SUB = SYM_2

ELSE ERR_SYM_VCC_SENS_SUB = NO_SYM

(3) ENDIF

(2) ENDIF

(1) ENDIF

The error bit LV_ERR_VCC_SENS_SUB and the end of diagnosis bit LV_END_DIAG_VCC_SENS_SUB are set by the failure memory.

Definition of global errorbit:

If LV_ERR_VCC_PVS_1 = 1

Or LV_ERR_VCC_PVS_2 = 1

Or LV_ERR_TPS_VCC = 1

Or LV_ERR_VCC_SENS_SUB = 1

Then LV_ERR_VCC = 1

Else LV_ERR_VCC = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VCC_PVS_1_MAX_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Max threshold on VCC_PVS_1 to detect a failure					
C_VCC_PVS_1_MIN_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_PVS_1 to detect a failure					
C_VCC_PVS_1_MIN_DIAG_EXT	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_PVS_1 to detect a failure (external ECU)					
C_ABC_INC_VCC_PVS_1	1	0...FFH	0...255	1	-
increment of debounce counter					
C_ABC_MAX_VCC_PVS_1	1	1...FFH	1...255	1	-
Max. increment of debounce counter					
C_VCC_PVS_2_MAX_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Max threshold on VCC_PVS_2 to detect a failure					
C_VCC_PVS_2_MIN_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_PVS_2 to detect a failure					
C_VCC_PVS_2_MIN_DIAG_EXT	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_PVS_2 to detect a failure (external ECU)					
C_ABC_INC_VCC_PVS_2	1	0...FFH	0...255	1	-
increment of debounce counter					
C_ABC_MAX_VCC_PVS_2	1	1...FFH	1...255	1	-
Max. increment of debounce counter					
C_VCC_TPS_MAX_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Max threshold on VCC_TPS to detect a failure					
C_VCC_TPS_MIN_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_TPS to detect a failure					
C_VCC_TPS_MIN_DIAG_EXT	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_TPS to detect a failure (external ECU)					
C_ABC_INC_VCC_TPS	1	0...FFH	0...255	1	-
increment of debounce counter					
C_ABC_MAX_VCC_TPS	1	1...FFH	1...255	1	-
Max. increment of debounce counter					
C_VCC_SENS_SUB_MAX_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Max threshold on VCC_SENS_SUB to detect a failure					
C_VCC_ SENS_SUB_MIN_DIAG	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_SENS_SUB to detect a failure					
C_VCC_ SENS_SUB _MIN_DIAG_EXT	1	0...3FFH	0...10	9.76 E-3	V
Min threshold on VCC_SENS_SUB to detect a failure (external ECU)					
C_ABC_INC_VCC_SENS_SUB	1	0...FFH	0...255	1	-
increment of debounce counter for VCC_SENS_SUB diagnosis					
C_ABC_MAX_VCC_ SENS_SUB	1	1...FFH	1...255	1	-
Max. increment of debounce counter for VCC_SENS_SUB diagnosis					

A.72.1 Reference voltage Diagnosis (appl. inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Sensor Supply Voltage Diagnosis (PVS_1)	SCP	SYM_0	*MEM*
	SCG	SYM_1	
	Sensor or wiring harness problem	SYM_2	
VCC_PVS_1		SYM_3	

Diagnostic	Symptom description	Symptom	Filter type
Sensor Supply Voltage Diagnosis (PVS_2 + MAP)	SCP	SYM_0	*MEM*
	SCG	SYM_1	
	Sensor or wiring harness problem	SYM_2	
VCC_PVS_2		SYM_3	

Diagnostic	Symptom description	Symptom	Filter type
Sensor Supply Voltage Diagnosis (TPS)	SCP	SYM_0	*MEM*
	SCG	SYM_1	
	Sensor or wiring harness problem	SYM_2	
TPS_VCC		SYM_3	

Diagnostic	Symptom description	Symptom	Filter type
Sensor Supply Voltage Diagnosis (PSP; ACP; PORT; VIS;DTP; MAP2)	SCP	SYM_0	*STD*
	SCG	SYM_1	
	Sensor or wiring harness problem	SYM_2	
VCC_SENS_SUB		SYM_3	

A.73 Battery voltage diagnosis

A.73.1 Battery voltage range diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_ERR_VB	V/O	0...1H	0...1	1	-
Boolean for error currently present on system voltage					

Input data:

VB	LV_IGK	LV_ES	LV_ST
	VS	LV_RLY_MAIN_EXT_ADJ	C_T_DLY_RLY_MAIN_DIAG

FUNCTION DESCRIPTION :

General information:

The purpose is to diagnose the system voltage which has to guarantee to perform diagnosis functions.

The diagnosis takes place with a recurrence of 1 sec.

Application conditions :

** Conditions to perform the diagnosis :*

if LV_IGK = 1
and LV_ES = 0
and LV_ST = 0
and *C_T_DLY_RLY_MAIN_DIAG has run out after LV_RLY_MAIN_EXT_ADJ = 1->0* *(Delay after Actuator Test)*
and *LV_RLY_MAIN_EXT_ADJ = 0* *(No Main Relay Actuator Test)*
and VS > C_VS_MIN_VB_DIAG

** Failure detection :*

if VB **<** C_VB_L_DIAG
then the Anti-Bounce counter is incremented with C_ABC_INC_VB
else the Anti-Bounce counter is decremented with 1

If the Anti-Bounce counter reaches C_ABC_MAX_VB, LV_ERR_VB is set to 1.

if VB **>** C_VB_H_DIAG
then the Anti-Bounce counter is incremented with C_ABC_INC_VB
else the Anti-Bounce counter is decremented with 1

If the Anti-Bounce counter reaches C_ABC_MAX_VB, LV_ERR_VB is set to 1.

Failure treatment :

* Debouce :

Anti-bounce counter increment: C_ABC_INC_VB

Maximum value of anti-bounce counter: C_ABC_MAX_VB

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_VB	1	0...FFH	0..255	1	-
. Anti-bounce counter increment for VB					
C_ABC_MAX_VB	1	1...FFH	1...255	1	-
Maximum value of antibounce counter for VB					
C_VS_MIN_VB_DIAG	1	0...FFH	0...255	1	Km/h
Minimum vehicle speed to perform Battery Voltage monitoring					
C_VB_L_DIAG	1	0...6CH	0...10.97	0.1	V
Battery voltage threshold for low battery voltage					
C_VB_H_DIAG	1	0...FFH	0...25.8984	26/256	V
Battery voltage threshold for high battery voltage					

A.73.2 Battery Voltage open circuit diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VB_OC	V/O	0...1H	0...1	1	-
Boolean for open circuit on Battery voltage VB					
ERR_SYM_VB_OC	V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom for VB_OC					
LV_CDN_DIAG_VB_OC	V	0...1H	0...1	1	-
Conditions for VB_OC diagnosis fulfilled					
LV_END_DIAG_VB_OC	V	0...1H	0...1	1	-
End of VB_OC diagnosis					

Input data:

VB_MES	V_IGK_MES	ENG_STATE	V_IGK
T_AST_ENVD	VS	LV_IGK	C_T_DLY_RLY_MAIN_DIAG
LV_RLY_MAIN_EXT_ADJ	V_IGK_BAS	NC_KEY_OFF_THR	

FUNCTION DESCRIPTION:

General information:

Purpose of this diagnosis is to check if the ECU is correctly supplied from battery via VB-Pin. In case of incorrect supply the ECU is still running with risk of hardware damage.

Application conditions:

Initialisation: at transition LV_IGK = 0 to 1 and at ECU reset

Recurrence: 10 ms

Activation: **If** LV_IGK = 1

and V_IGK_BAS > NC_KEY_OFF_THR

and C_T_DLY_RLY_MAIN_DIAG has run out after LV_IGK = 0 --> 1 *(to ensure main relay had enough time to switch on after key on)*

and VB_MES > 1,74 V

and V_IGK_MES > 1,046 V

and C_T_DLY_RLY_MAIN_DIAG has run out after LV_RLY_MAIN_EXT_ADJ = 1->0 *(Delay After Actuator Test)*

and LV_RLY_MAIN_EXT_ADJ=0 (no Actuator Test for Main Relay active)

then LV_CDN_DIAG_VB_OC = 1

else LV_CDN_DIAG_VB_OC = 0

Formula section:

If |VB_MES – V_IGK_MES| > ID_VB_OC_DIAG

Then ERR_SYM_VB_OC = SYM_2

LV_ERR_VB_OC = 1 (after debounce)

Else ERR_SYM_VB_OC = NO_SYM

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Battery Voltage Supply		SYM_0	*MEM*
		SYM_1	
	Open Circuit	SYM_2	
VB_OC		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_VB_OC	1	0...FFH	0...255	1	-
Anti-bounce counter increment for VB_OC Diagnosis					
C_ABC_MAX_VB_OC	1	0...FFH	0...255	1	-
Maximum of anti bounce counter for VB_OC diagnosis					
ID_VB_OC_DIAG	1*6	0...3FFH	0...4,9951	0,0049	V
LDP_V_IGK__VB_OC_DIAG	6	0...FFH	0...26	0.102	V
Maximum allowed difference between VB_MES and V_IGK_MES					

A.73.3 Battery voltage diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Battery Voltage Diagnosis	System voltage high	SYM_0	*STD*
	System voltage low	SYM_1	
		SYM_2	
VB		SYM_3	

List of Environmental Data to store in Failure Memory: V_IGK
ENG_STATE
T_AST_ENVD
VS

Emergency operation ;

* Limp Home : None

A.74 Immobilizer diagnosis

Output data:

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_IMOB_SMARTRA_ANT	V/S	0 ... 01H	0 ... 1	1	-
Antenna coil error					
LV_CDN_DIAG_IMOB_SMARTRA_ANT	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis condition Antenna coil diagnosis					
ERR_SYM_IMOB_SMARTRA_ANT	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom antenna coil diagnosis					
LV_END_DIAG_IMOB_SMARTRA_ANT	O/V	0 ... 1H	0 ... 1	1	-
End of diagnosis antenna coil diagnosis					
LV_ERR_IMOB_MSG	V/S	0...1H	0...1	1	-
Immobilizer invalid message error					
LV_CDN_DIAG_IMOB_MSG	O/V	0...1H	0...1	1	-
Diagnosis condition Immobilizer invalid message diagnosis					
ERR_SYM_IMOB_MSG	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom Immobilizer invalid message diagnosis					
LV_END_DIAG_IMOB_MSG	O/V	0...1H	0...1	1	-
End of diagnosis Immobilizer invalid message diagnosis					
LV_ERR_IMOB_RESP	V/S	0...1H	0...1	1	-
Immobilizer no response error					
LV_CDN_DIAG_IMOB_RESP	O/V	0...1H	0...1	1	-
Diagnosis condition Immobilizer no response diagnosis					
ERR_SYM_IMOB_RESP	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom Immobilizer no response diagnosis					
LV_END_DIAG_IMOB_RESP	O/V	0...1H	0...1	1	-
End of diagnosis Immobilizer no response diagnosis					
LV_ERR_IMOB_TP_PROG	V/S	0 ... 01H	0 ... 1	1	-
Transponder programming Fault					
LV_CDN_DIAG_IMOB_TP_PROG	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis condition IMOB_TP_PROG diagnosis					
ERR_SYM_IMOB_TP_PROG	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Detected error symptom IMOB_TP_PROG diagnosis					
LV_END_DIAG_IMOB_TP_PROG	O/V	0 ... 1H	0 ... 1	1	-
End of diagnosis IMOB_TP_PROG diagnosis					
LV_ERR_IMOB_TP_DAT	V/S	0...1H	0...1	1	-
Invalid transponder data fault					
LV_CDN_DIAG_IMOB_TP_DAT	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_TP_DAT diagnosis					

ERR_SYM_IMOB_TP_DAT	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_TP_DAT diagnosis					
LV_END_DIAG_IMOB_TP_DAT	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_TP_DAT diagnosis					
LV_ERR_IMOB_TP_MOD	V/S	0...1H	0...1	1	-
Invalid transponder programming mode fault					
LV_CDN_DIAG_IMOB_TP_MOD	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_TP_MOD diagnosis					
ERR_SYM_IMOB_TP_MOD	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_TP_MOD					
LV_END_DIAG_IMOB_TP_MOD	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_TP_MOD diagnosis					
LV_ERR_IMOB_AUTH	V/S	0 ... 01H	0 ... 1	1	-
TP authentication failed					
LV_CDN_DIAG_IMOB_AUTH	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis condition IMOB_AUTH diagnosis					
ERR_SYM_IMOB_AUTH	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Detected error symptom IMOB_AUTH diagnosis					
LV_END_DIAG_IMOB_AUTH	O/V	0 ... 1H	0 ... 1	1	-
End of diagnosis IMOB_AUTH diagnosis					
LV_ERR_IMOB_TWICE_IGN_ON	V/S	0...1H	0...1	1	-
Maximum number of Twice IGN ON exceeded					
LV_CDN_DIAG_IMOB_TWICE_IGN_ON	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_TWICE_IGN_ON diagnosis					
ERR_SYM_IMOB_TWICE_IGN_ON	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_TWICE_IGN_ON diagnosis					
LV_END_DIAG_IMOB_IGN_ON	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_TWICE_IGN_ON diagnosis					
LV_ERR_IMOB_EMS_REQ	V/S	0...1H	0...1	1	-
Invalid request from EMS or corrupted data					
LV_CDN_DIAG_IMOB_EMS_REQ	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_EMS_REQ diagnosis					
ERR_SYM_IMOB_EMS_REQ	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_EMS_REQ diagnosis					
LV_END_DIAG_IMOB_EMS_REQ	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_EMS_REQ diagnosis					
LV_ERR_IMOB_MEM	V/S	0...1H	0...1	1	-
EMS internal permanent memory fault					
LV_CDN_DIAG_IMOB_MEM	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_MEM diagnosis					

ERR_SYM_IMOB_MEM	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_MEM diagnosis					
LV_END_DIAG_IMOB_MEM	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_MEM diagnosis					
LV_ERR_IMOB_CONF	V/S	0...1H	0...1	1	-
Configuration fault					
LV_CDN_DIAG_IMOB_CONF	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_CONF diagnosis					
ERR_SYM_IMOB_CONF	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_CONF diagnosis					
LV_END_DIAG_IMOB_CONF	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_CONF diagnosis					
LV_ERR_IMOB_TESTER	V/S	0...1H	0...1	1	-
Tester fault					
LV_CDN_DIAG_IMOB_TESTER	O/V	0...1H	0...1	1	-
Diagnosis condition IMOB_TESTER diagnosis					
ERR_SYM_IMOB_TESTER	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom IMOB_TESTER diagnosis					
LV_END_DIAG_IMOB_TESTER	O/V	0...1H	0...1	1	-
End of diagnosis IMOB_TESTER diagnosis					
LV_ERR_IMOB_SMARTRA3	V/S	0...1H	0...1	1	-
Smartra3 unit fault					
LV_CDN_DIAG_IMOB_SMARTRA3	O/V	0...1H	0...1	1	-
Diagnosis condition for Smartra3 unit diagnosis					
ERR_SYM_IMOB_SMARTRA3	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error symptom SMARTRA3 diagnosis					
LV_END_DIAG_IMOB_SMARTRA3	O/V	0...1H	0...1	1	-
End of diagnosis SMARTRA3 diagnosis					

Input data:

LV_ANS_NOT_IMOB	LV_ERR_COM_P2MAX_IMOB	LV_ERR_COM_P1MAX_IMOB	LV_ERR_COM_P4MAX_IMOB
CONF_IMOB	LV_ERR_COM_RESP_IMOB	CONF_IMOB_VER	

A.74.1 SMARTRA fault

Description:

The purpose is to diagnose the SMARTRA and its communication.
Following failures during transmission may occur :

- SMARTRA receives error code 01h: Antenna error
- SMARTRA receives no response message within the max. communication time
- Communication timing not correct (P1MAX, P2MAX, or P4MAX)
- Received byte has wrong address or the size of the checksum is not correct

Application conditions :

Activation: *CONF_IMOB_VER =1*

During authentication and key teaching process

Deactivation : After good authentication
LV_ERR_IMOB_SMARTRA_ANT = **false**
LV_ERR_IMOB_MSG = **false**
LV_ERR_IMOB_RESP = **false**

Formula section:

If SMARTRA responds with error code 01h (Antenna coil error)
then an internal counter is increased, and try again untill internal counter is 3 in same DC
After **three** times negative response received from SMARTRA, the flag
LV_ERR_IMOB_SMARTRA_ANT = **true** with **SYM_0**

If { Inter-byte time for immobilizer request exceeds P1MAX }
LV_ERR_COM_P1MAX_IMOB = **true**
or
{ Inter-byte time for EMS request exceeds P4MAX }
LV_ERR_COM_P4MAX_IMOB = **true**
or
{ Address or the size or the checksum is not correct }
LV_ERR_COM_RESP_IMOB = **true**
Then an internal counter is increased
After **three** times bad communication during key authentication, the flag
LV_ERR_IMOB_ MSG = **true** with **SYM_0** **// SMARTRA**

If {the time-out delay for communication has elapsed}
LV_ANS_NOT_IMOB = **true**
or
{ No received byte within P2MAX }
LV_ERR_COM_P2MAX_IMOB = **true**
then
If auto-variant learning(auto-detection) has not been done yet
Then wait 1.5 second for response from immobilizer unit
If there is no response from SMARTRA within 1.5 sec
Else LV_ERR_IMOB_RESP = **true** with **SYM_0** **// SMARTRA**
Else an internal counter is increased
After **three** times bad communication during key authentication, the flag
LV_ERR_IMOB_RESP = **true** with **SYM_0** **// SMARTRA**

If CONF_IMOB_SMARTRA = 2 **// SMARTRA3 Only**
Then

If {EMS send 4Bh}
{SMARTRA response with the state byte 02h}
Then an internal counter is increased
After counter reaches three, the flag
LV_ERR_IMOB_SMARTRA3 = 1
ERR_SYM_IMOB_SMARTRA3 = "SYM_0"

If {EMS send 4Bh}
{SMARTRA response with the state byte 03h}
Then an internal counter is increased
After counter reaches three, the flag
LV_ERR_IMOB_SMARTRA3 = 1
ERR_SYM_IMOB_SMARTRA3 = "SYM_1"

Rev.C :
Adding SMT3 Error symptom

If { EMS send 4Bh}
{SMARTRA response with the state byte 01h with an invalid ERN}
Or { EMS send 54h}
{SMARTRA response with the 03h messege (Incorect Password for learnt smartra during key teaching procedure(69 02 03 CS) }

Then an internal counter is increased
After counter reaches three, the flag
LV_ERR_IMOB_SMARTRA3 = 1
ERR_SYM_IMOB_SMARTRA3 = "SYM_2"

If { EMS send 4Bh}
{ Locked SMARTRA response negative response 15h with error code 05h}
Then an internal counter is increased
After counter reaches three, the flag
LV_ERR_IMOB_SMARTRA3 = 1
ERR_SYM_IMOB_SMARTRA3 = "SYM_3"

<u>Emergency operation :</u>

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.2 Smart-Key Communication Fault

Description:

The purpose is to diagnose communication with Smart-Key.
Following failures during transmission may occur :

- EMS receives no response message within the max. communication time
- Communication timing not correct (P1MAX, P2MAX, or P4MAX)
- Received byte has wrong address or the size of the checksum is not correct

Application conditions :

Activation: *CONF_IMOB_VER = 2* ***or*** *CONF_IMOB_VER = 3*
During communication

Deactivation : After good authentication
LV_ERR_IMOB_MSG = **false**
LV_ERR_IMOB_RESP = **false**

Formula section:

If { Inter-byte time for immobilizer request exceeds P1MAX }
LV_ERR_COM_P1MAX_IMOB = **true**
or
{ Inter-byte time for EMS request exceeds P4MAX }
LV_ERR_COM_P4MAX_IMOB = **true**
or
{ Address or the size or the checksum is not correct }
LV_ERR_COM_RESP_IMOB = **true**
Then an internal counter is increased
If C_MAX_DLY_ACT_IMOB elapsed after LV_IGK = 0 -> 1
Or An internal counter > C_MAX_ACK_IMOB_SMK
Then LV_ERR_IMOB_ MSG = **true** with **SYM_1** **// Smart-Key**
EndIf
EndIf

LV_ERR_IMOB_MSG error is detected, when authentification error [Invalid SMART-KEY (with wrong VIN) send "Positive response" with wrong VIN code] in SMART-KEY learnt state.

If {the time-out delay for communication has elapsed}
LV_ANS_NOT_IMOB = **true**
or
{ No received byte within P2MAX }
LV_ERR_COM_P2MAX_IMOB = **true**
Then an internal counter is increased
If C_MAX_DLY_ACT_IMOB elapsed after LV_IGK = 0 -> 1
Or An Internal counter > C_MAX_ACK_IMOB_SMK
Then LV_ERR_IMOB_RESP = **true** with **SYM_1** **// Smart-Key**
EndIf
EndIf

Emergency operation :

* Limp Home : No Limp-Home

A.74.3 Transponder fault

Description:

The purpose is to diagnose the signal from Transponder.

Following failure during transmission may occur :

- SMARTRA code 02h received : invalid Transponder data
  Corrupted Tp data, or more than one Tp in the field or no Tp in the field
- SMARTRA code 04h received : passive mode invalid
  Tp not in password mode, or Tp transport data has been changed
- SMARTRA code 11..17h received : Transponder programming error
  Transponder programming error

Application conditions :

Activation: *CONF_IMOB_VER = 1*
During authentication and key teaching process

Deactivation : After good authentication
LV_ERR_IMOB_TP_PROG = **false**
LV_ERR_IMOB_TP_DAT = **false**
LV_ERR_IMOB_TP_MOD = **false**

Formula section:

If SMARTRA responds with error code 02h

Then an internal counter is increased

After **three** times negative response received from SMARTRA,
The flag LV_ERR_IMOB_TP_DAT = **true** with **SYM_0**

If SMARTRA responds with error code 04h

Then an internal counter is increased

After **three** times negative response received from SMARTRA,

The flag LV_ERR_IMOB_TP_MOD = **true** with **SYM_0**

If SMARTRA responds with error code 11h, 12h, 13h, 14h, 15h, 16h, 17h

Then an internal counter is increased

After **three** times negative response received from SMARTRA,

The flag LV_ERR_IMOB_TP_PROG = **true** with **SYM_0**

Emergency operation :

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.4 Invalid Key fault

General information :

The purpose is to indicate that invalid data are sent by SMARTRA.
Following failure during transmission may occur:
- 3 times wrong code is received from key during authentication
- 3 times detection of a "VIRGIN" transponder at a "LERNT" EMS

Application conditions :

Activation: *CONF_IMOB_VER = 1*
During authentication and key teaching process

Deactivation : After good authentication LV_ERR_IMOB_AUTH = **false**

Formula section:

If { wrong code is received from key during authentication }
LV_ANS_ERR_IMOB = **true**

then an internal counter is increased
After three times wrong code received,
LV_ERR_IMOB_AUTH = **true** with **SYM_0**

Emergency operation :

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.5 EMS fault

Description:

The purpose is to diagnose an invalid request from EMS to the SMARTRA device.
Following failure may occur :
- SMARTRA code 03h received : invalid request from EMS or corrupted data
- The maximum number of (32) of "twice IGN ON" in virgin EMS is exceeded

Application conditions :

Activation: During authentication and key teaching process

Deactivation : After good authentication
LV_ERR_IMOB_TWICE_IGN_ON= **false**
LV_ERR_IMOB_EMS_REQ = **false**

Formula section:

```
If      CONF_IMOB_VER = 1
And     SMARTRA responds with error code 03h
Then    an internal counter is increased
        After three times negative response received from SMARTRA,
        The flag LV_ERR_IMOB_EMS_REQ = true with SYM_0
EndIf

If   LC_ENA_DIAG_IMOB_TWICE_IGN_ON =1
Then If      CONF_IMOB_VER = 1                               // Smartra
         and CTR_KEY_FAST_IMOB = NC_CTR_KEY_FAST_IMOB
     Then    LV_ERR_IMOB_TWICE_IGN_ON = true
             ERR_SYM_IMOB_TWICE_IGN_ON  = 1H      (SYM_0)
     Else    If      CONF_IMOB_VER = 2                       // Smart-Key 1
                 and CTR_KEY_FAST_IMOB = NC_CTR_KEY_FAST_IMOB
             Then    LV_ERR_IMOB_TWICE_IGN_ON = true
                     ERR_SYM_IMOB_TWICE_IGN_ON  = 2H   (SYM_1)
             Else    If     CONF_IMOB_VER = 3                // Smart-Key 2
                        and CTR_KEY_FAST_IMOB = NC_CTR_KEY_FAST_IMOB_SMK_2
                     Then   LV_ERR_IMOB_TWICE_IGN_ON = true
                            ERR_SYM_IMOB_TWICE_IGN_ON  = 2H    (SYM_1)
                     Else   LV_ERR_IMOB_TWICE_IGN_ON = false
                            ERR_SYM_IMOB_TWICE_IGN_ON  = 0H    (NO_SYM)
EndIf
```

Emergency operation :

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.6 EMS internal permanent memory fault

Description:

The purpose is to diagnose if incorrect immobilizer data are stored in flash:

The error flag (LV_ERR_IMOB_MEM) is set if one of the following error happens :
- at least one of the three data sets stored in the flash EPROM has wrong checksum.
- an impossible Immobilizer status is set

Application conditions :

Activation: After reset

Deactivation : -

Formula section:

If Checksum and data read out from flash EPROM does not fit together

or an impossible Immobilizer status is set

then LV_ERR_IMOB_MEM = **true** with **SYM_0**

Emergency operation :

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.7 Tester fault

Description:

The purpose of this diagnosis is to detect if incorrect or invalid data are sent by service-tester.

The error flag /LV_ERR_IMOB_TESTER) is set if one of the following error happens:

- Invalid message from tester (e.g. wrong length of data)

Application conditions :

Activation: CONF_IMOB_VER = 1
And LC_ENA_DIAG_IMOB_TESTER = 1
during tester communication

Deactivation : After error memory reset (by tester): LV_ERR_IMOB_TESTER = **false**

Formula section:

If message with wrong length from tester

then LV_ERR_IMOB_TESTER = **true** with **SYM_0**

EndIf

Emergency operation :

* Limp Home : start possible if certain conditions fulfilled
(see Immobilizer specification)

A.74.8 Immobilizer Configuration Fault

Description:

The purpose of this diagnosis is to detect if a Non-immobilizer EMS is build in a vehicle that is equipped with a SMARTRA device or Smart-Key.

The error flag (LV_ERR_IMOB_CONF) is set if the following error happens:

- A Non-immo EMS detects a SMARTRA connected on the serial communication line.

Application conditions :

Activation: CONF_IMOB = 0
And LC_ENA_DIAG_IMOB_CONF = 1
a test communication on the immobilizer line is started after ignition ON.

Deactivation: **If** configuration is changed to immobilizer (CONF_IMOB = 1) then LV_ERR_IMOB_CONF = **false**

Formula section:

If CONF_IMOB = 0

and ignition is switched ON

then a test communication on immobilizer line is started.

If a SMARTRA device is detected *{positive response from SMARTRA}*

then LV_ERR_IMOB_CONF = **true** with **SYM_0** // SMARTRA

If a Smart-Key device is detected {Positive response from Smart-Key}

Then LV_ERR_IMOB_CONF = **true** with **SYM_1** // Smart-Key

Calibration data:

Name	Type	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ENA_DIAG_IMOB_TESTER	1U	1	0 ... 1H	0 ... 1	1	-
Selection switch to enable IMOB_TESTER Diagnosis						
LC_ENA_DIAG_IMOB_CONF	1U	1	0...1H	0... 1	1	-
Selection Switch to enable IMOB_CONF Diagnosis						
LC_ENA_DIAG_IMOB_TWICE_IGN_ON	1U	1	0...1H	0... 1	1	-
Selection Switch to enable IMOB_TWICE_IGN_ON Diagnosis						

A.75 Immobilizer Diagnosis (appl. inc.)

A.75.1 Application assistances

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
SMARTRA Antenna fault	Antenna coil error	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_SMARTRA_ANT		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
IMOB Message fault	SMARTRA Invalid message(timing)	SYM_0	*STD*
	Smart-Key Invalid message(timing)	SYM_1	
		SYM_2	
IMOB_ MSG		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
IMOB No Response fault	No Answer from SMARTRA	SYM_0	*STD*
	No Answer from Smart-Key	SYM_1	
		SYM_2	
IMOB_ RESP		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Transponder Programming fault	Transponder programming error	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_TP_PROG		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Transponder Data fault	*Invalid transponder data*	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_TP_DAT		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Transponder Programming Mode fault	Programming mode invalid	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_TP_MOD		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Invalid Key fault	*Authentication failed*	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_AUTH		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
EMS Trial Number fault	Max. number of trials exceeded for Smartra	SYM_0	*STD*
	Max. number of trials exceeded for Smart-Key	SYM_1	
		SYM_2	
IMOB_TWICE_IGN_ON		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
EMS Request fault	*Invalid request from EMS or corrupted data*	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_EMS_REQ		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
EMS internal permanent memory fault	Checksum error	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_MEM		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Tester fault	Wrong data from tester	SYM_0	*STD*
		SYM_1	
		SYM_2	
IMOB_TESTER		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Configuration fault	Non- immobilizer EMS in SMARTRA type immobilizer vehicle	SYM_0	*STD*
	Non-immobilizer EMS in Smart-Key type immobilizer vehicle	SYM_1	
		SYM_2	
IMOB_CONF		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
SMARTRA3 fault	Virgin SMARTRA3 at Learnt EMS	SYM_0	*STD*
	Neutral SMARTRA3 at Learnt EMS	SYM_1	
	Incorrect Authentication of EMS and SMARTRA3	SYM_2	
IMOB_SMARTRA3	Locking of SMARTRA3	SYM_3	

A.76 Upstream oxygen sensor electrical failure monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_EL_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, fault currently present on upstream oxygen sensor signal					

Input data:

LV_ERR_SCG_LS_UP[NC_CBK_EX_NR]	LV_IGK	LV_ERR_SCP_LS_UP[NC_CBK_EX_NR]	LV_ERR_OC_LS_UP[NC_CBK_EX_NR]
NC_CBK_EX_NR	LV_ERR_READY_LS_UP[NC_CBK_EX_NR]		

FUNCTION DESCRIPTION:

General information:

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Description:

The oxygen sensor signal electrical diagnosis shall detect the following faults:

A. Short circuit of the oxygen sensor signal line to GND
B. Short circuit of the oxygen sensor signal line to Vbatt
C. Open circuit (line break) in the connection to oxygen sensor element
D. Delayed readiness of the sensor

Should one of the electrical fault flags be set, then a global error flag (LV_ERR_EL_LS_UP[i]) shall be set to indicate that an electrical fault currently exists on the particular oxygen sensor.

Each of the above listed faults shall be described individually in further detail below.

Application conditions:

Initialisation: all variables shall be initialised to 0 at LV_IGK = 0 -> 1 or reset or by clearing the error memory.

Recurrence:

This diagnosis shall be carried out every *100 ms* provided no debounced fault has been detected and unless specifically noted otherwise in the relevant section.

Activation / Deactivation:

If LV_IGK = 1

Formula section:

If LV_ERR_SCP_LS_UP[i] = 1

or LV_ERR_SCG_LS_UP[i] = 1

or LV_ERR_OC_LS_UP[i] = 1

or LV_ERR_READY_LS_UP[i] = 1

then LV_ERR_EL_LS_UP[i] = 1

else LV_ERR_EL_LS_UP[i] = 0

endif

A.76.1 Short circuit of oxygen sensor signal to GND

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, short to GND fault currently present on upstream oxygen sensor signal					
ERR_SYM_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0H 2H	NO_SYM SYM_1	1	-
Variable indicating short to ground fault currently present on upstream oxygen sensor signal					
LV_R_MES_VLD_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Valid conditions of resitance value of oxygen sensor upstream diagnosis short to ground					
LV_CDN_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor upstream diagnosis short to ground					
LV_DIAG_SCG_REQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Request bit to indicate need of CP close for upstream oxygen sensor SCG diagnosis					
LV_END_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of upstream oxygen sensor SCG diagnosis					
LV_LAM_MAX_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that lambda controller limited and CPS closed or inactive					
CTR_R_UPD_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Value of handshake counter for last valid resistance value used for short circuit to ground diagnosis					
T_LAM_LIM_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...6553.5	0.1	s
Time the lambda controller remains limited					
T_ACT_PLS_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...6553.5	0.1	s
Time the lambda controller is active					
MAF_INT_CPS_CLOSE_VLS_UP_MIN[NC_CBK_EX_NR]	V	0...FFFFH	0…1820.4167	0.0277	g
Mass air flow for diagnosis (after CPS closed)					

Input data:

VLS_UP[NC_CBK_EX_NR]	LV_LAM_LSCL[NC_CBK_EX_NR]	FAC_LAM_LIM[NC_CBK_EX_NR]	FAC_LAM_MAX[NC_CBK_EX_NR]
LV_ERR_EL_CPS	LV_IGK	LV_ERR_MEC_OPEN_CPS	LV_CP_CLOSE_ACT
R_IT_LS_UP[NC_CBK_EX_NR]	LV_INH_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	MAF_KGH	LV_ERR_LS_DOWN[NC_CBK_EX_NR]

VLS_DOWN[NC_CBK_EX_NR]	CTR_CYCNR_R_IT_LS_UP_VLD[NC_CBK_EX_NR]	LV_DC	

Description:

In order to determine the electrical fault, short circuit to GND, the oxygen sensor signal voltage shall be compared to a calibratable threshold (C_VLS_MIN_DIAG_SCG_LS_UP) in conjunction with certain activation and additional conditions.

Should a hard short to GND (i.e. approx. 0 Ω) exist on the oxygen sensor signal line, the signal voltage will tend to 0 V. This condition may also occur should the air-fuel mixture (AFR) become lean due to operating conditions or due to an air leak, intentional or not. Hence the following additional fault conditions shall be required:

The initial conditions shall reasonably ensure that no false air is being admitted into the system. One possible source of false air may be due to an open or a faulty Canister Purge Solenoid (CPS). Should a fault be determined to exist in the Evaporative Emissions System (LV_ERR_EL_CPS, LV_ERR_DIAG_CPS), then no further SCG diagnosis shall take place. An additional criterion shall only permit diagnosis to take place above a limiting thresholds of MAF_KGH value (C_MAF_KGH_MIN_DIAG_SCG_LS_UP) as the effect of an air leak is far greater at low engine loads.

In order to discriminate against a lean AFR due to the current engine / driving conditions and a possible short to GND, the lambda controller shall be taken into account: A fault may prevail if the lambda controller is active (LV_LAM_LSCL[i] = 1) and has reached the enriched limit (FAC_LAM_MAX[i]). Should the above activation conditions be met and the lambda controller be active, a timer (T_ACT_PLS_DIAG_SCG_LS_UP[i]) shall be started. If any of the initial conditions are no longer met, all timers and flag LV_LAM_MAX_DIAG_SCG_LS_UP[i] shall be reset and the request for CPS closure revoked (LV_DIAG_SCG_REQ_LS_UP[i]).

On reaching the lambda controller limit, the diagnosis function shall start a timer (T_LAM_LIM_DIAG_SCG_LS_UP[i]). Should the lambda controller remain limited uninterrupted for a pre-determined amount of time (C_T_LAM_LIM_DIAG_SCG_LS_UP), then the closure of the CPS shall be requested (LV_DIAG_SCG_REQ_LS_UP[i]) and state of the Evaporative Emissions System shall be checked. Should the lambda controller fall below the max. limit or the above initial conditions not be met, the timers shall be reset, any requested closure of the CPS revoked and the flag indicating controller limitation & CPS closed or inactive (LV_LAM_MAX_DIAG_SCG_LS_UP[i]) shall be reset. This shall prevent the diagnosis from blocking normal or minimum operation of the Evaporative Emissions System.

In order to discriminate against a lean AFR due to air via the Activated Charcoal Filter (ACF) canister, the state of the Evaporative Emissions System shall be checked as stated above. The steady closed state of the CPS shall be determined via EVAP Control state variables (LV_CP_CLOSE_ACT). Once the CPS has been determined to be closed, a mass air flow integral shall be started (MAF_INT_CPS_CLOSE_VLS_UP_MIN[i]). Should all above conditions continue to be met for a pre-determined threshold (C_MAF_INT_CPS_DIAG_SCG_LS_UP).

Each time that the MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] exceeded the minimum duration (C_MAF_INT_CPS_DIAG_SCG_LS_UP) for the first time, e.g. first time after engine start or after the T_LAM_LIM_DIAG_SCG_LS_UP[i] has been reset, the CTR_R_UPD_DIAG_SCG_LS_UP[i] shall be updated once only with the value CTR_CYCNR_R_IT_LS_UP_VLD[i]. This shall ensure that only new internal resistance values be used for diagnosis purposes that have been determined since all the activation conditions have been met.

It shall be determined whether a new internal resistance has been determined by comparing the contents of counter CTR_R_UPD_DIAG_SCG_LS_UP[i] with that of CTR_CYCNR_R_IT_LS_UP_VLD[i]. Should the counters be unequal, then a new value is available and CTR_R_UPD_DIAG_SCG_LS_UP[i] shall be updated with CTR_CYCNR_R_IT_LS_UP_VLD[i]. If this conditions and all the above conditions be met, the diagnosis shall be considered to be active. LV_LAM_MAX_DIAG_SCG_LS_UP[i] shall then be set.

Should at any time the state of the CPS change from being closed, the diagnosis shall reset LV_LAM_MAX_DIAG_SCG_LS_UP[i] and the mass MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] only.

The diagnosis condition shall be considered to be available if all the following conditions are met:

1) Activation time ellapsed (T_ACT_PLS_DIAG_SCG_LS_UP[i] > C_T_ACT_PLS_DIAG_SCG_LS_UP)
2) A new internal resistance exists (LV_R_MES_VLD_DIAG_SCG_LS_UP[i] = 1)
3) Either the lambda controller is not in limitation or the lambda controller is in limitation and the flag LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 1 is set (*CPS closed and MAF delay ellapsed).*

If the diagnosis is available, the oxygen signal voltage fall below the threshold (VLS_UP[i] < C_VLS_MIN_DIAG_SCG_LS_UP) and the condition R_IT_LS_UP[i] under threshold be met and downstream voltage is over a threshold and lambda controller is still in the limit, a signal short to GND shall be determined to be present and the debounce counter incremented.

If the voltage not fall below the said threshold or the condition R_IT_LS_UP[i] under threshold is no longer met, then either the limiting of the lambda controller may be assumed to be being caused by something other than a short to ground on the oxygen sensor signal or a glitch has caused the voltage to temporarily exceed the SCG voltage threshold. In this case, should the debounce counter be determined to be zero, the timers shall be reset and any request of closure of the CPS shall be revoked otherwise the debounce counter shall be decremented.
It should be noted that the debounce counter shall only be decremented when the debounce counter is not zero, should the voltage or R_IT_LS_UP[i] under threshold criteria not be met.
The error flag LV_ERR_SCG_LS_UP[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

<u>Application conditions:</u>

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 at LV_IGK = 0 -> 1 or reset or by clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence:

The oxygen sensor analog input signal shall be measured every 10 ms. This electrical diagnosis shall be carried out every ***<u>T_SAMPLE = 100 ms</u>*** provided no debounced fault has been detected and unless specifically noted otherwise in the relevant section.

Activation / Deactivation:

If LV_IGK = 1

and LV_INH_DIAG_SCG_LS_UP[i] = 0

and LV_LAM_LSCL[i] = 1

and LV_ERR_EL_CPS = 0

and LV_ERR_MEC_OPEN_CPS = 0

and LV_ERR_LS_DOWN[i] = 0

and MAF_KGH ≥ C_MAF_KGH_MIN_DIAG_SCG_LS_UP

then T_ACT_PLS_DIAG_SCG_LS_UP[i] [ms] =T_ACT_PLS_DIAG_SCG_LS_UP[i] [ms] + T_SAMPLE [ms]

% *One-off update* CTR_R_UPD_DIAG_SCG_LS_UP[i]

If CTR_CYCNR_R_IT_LS_UP_VLD[i] ≠ CTR_R_UPD_DIAG_SCG_LS_UP[i]

then LV_R_MES_VLD_DIAG_SCG_LS_UP[i] = 1

CTR_R_UPD_DIAG_SCG_LS_UP[i] = CTR_CYCNR_R_IT_LS_UP_VLD[i]

else LV_R_MES_VLD_DIAG_SCG_LS_UP[i] = 0

endif

If FAC_LAM_LIM[i] ≥ FAC_LAM_MAX[i]

then T_LAM_LIM_DIAG_SCG_LS_UP[i] [ms]=

T_LAM_LIM_DIAG_SCG_LS_UP[i] [ms]+ T_SAMPLE [ms]

If T_LAM_LIM_DIAG_SCG_LS_UP[i] ≥

C_T_LAM_LIM_DIAG_SCG_LS_UP

then LV_DIAG_SCG_REQ_LS_UP[i] = 1

If LV_CP_CLOSE_ACT = 1

then MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] [g] =

MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] [g] + MAF_KGH * T_SAMPLE [ms] * 1/3600 [(g*h) / (kg*ms)]

If MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] ≥

C_MAF_INT_CPS_DIAG_SCG_LS_UP

then LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 1

else LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 0

endif

else MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] = 0

LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 0

endif

else LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 0

MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] = 0

endif

else MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] = 0

T_LAM_LIM_DIAG_SCG_LS_UP[i] = 0

LV_DIAG_SCG_REQ_LS_UP[i] = 0
LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 0
endif
else MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] = 0
T_LAM_LIM_DIAG_SCG_LS_UP[i] = 0
T_ACT_PLS_DIAG_SCG_LS_UP[i] = 0
LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 0
LV_DIAG_SCG_REQ_LS_UP[i] = 0
Endif
If T_ACT_PLS_DIAG_SCG_LS_UP[i] > C_T_ACT_PLS_DIAG_SCG_LS_UP
and LV_R_MES_VLD_DIAG_SCG_LS_UP[i] = 1
and [(T_LAM_LIM_DIAG_SCG_LS_UP[i] > 0
and LV_LAM_MAX_DIAG_SCG_LS_UP[i] = 1)
or T_LAM_LIM_DIAG_SCG_LS_UP[i] = 0]
then ***"Diagnosis available"***
LV_CDN_DIAG_SCG_LS_UP[i] = 1
else ***"Diagnosis NOT available"***
LV_CDN_DIAG_SCG_LS_UP[i] = 0
endif

Formula section:

If LV_CDN_DIAG_SCG_LS_UP[i] = 1
then **If** VLS_UP[i] < C_VLS_MIN_DIAG_SCG_LS_UP
and R_IT_LS_UP[i] < C_R_MAX_DIAG_SCG_LS_UP
and FAC_LAM_LIM[i] >= FAC_LAM_MAX[i]
then *"Short to GND fault present"*
ERR_SYM_SCG_LS_UP[i] = SYM_1 /"SHORT TO GROUND"
/the ERRM shall automatically debounces this symptom
else ERR_SYM_SCG_LS_UP[i] = "NO_SYM"
endif
else *freeze* ERR_SYM_SCG_LS_UP[i]
endif

The following statements shall be executed according the condition of the ABC Counter:

If ABC counter1 -> 0
then MAF_INT_CPS_CLOSE_VLS_UP_MIN[i] = 0
T_LAM_LIM_DIAG_SCG_LS_UP[i] = 0

LV_DIAG_SCG_REQ_LS_UP[i] = 0

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_VLS_MIN_DIAG_SCG_LS_UP	1	0...3FFH	0…4.995117	4.883 e-3	V
Oxygen sensor voltage threshold under which short to GND may be present					
C_T_ACT_PLS_DIAG_SCG_LS_UP	1	0...FFFFH	0...6553.5	0.1	s
Minimum required delay for forced activation of short to ground diagnosis					
C_T_LAM_LIM_DIAG_SCG_LS_UP	1	0...FFFFH	0...6553.5	0.1	s
Minimum required delay for which lambda controller limited at max. value (continuous lean AFR present)					
C_MAF_INT_CPS_DIAG_SCG_LS_UP	1	0...FFFFH	0…1820.4167	0.0277	g
Minimum required mass air flow for diagnosis (after CPS closed)					
C_MAF_KGH_MIN_DIAG_SCG_LS_UP	1	0...FFFFH	0...2047.9687	31.25e-3	kg/h
Minimum MAF_KGH threshold required to permit diagnosis to take place					
C_R_MAX_DIAG_SCG_LS_UP	1	0...FFFFH	0...65535	1	Ohm
Maximum resitance threshold oxygen sensor internal resistance during short to ground sensor					

Diagnosis	Symptom	Nr	ABC type
Upstream Oxygen Sensor Diagnosis for Short Circuit to Grd	Sym_1	*0*	MEM
SCG_LS_UP[i]			

A.76.2 Short circuit of oxygen sensor signal to VBATT

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SCP_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, short to plus fault currently present on upstream oxygen sensor signal					
ERR_SYM_SCP_LS_UP[NC_CBK_EX_NR]	V/O	0H 1H	NO_SYM SYM_0	1	-
Variable indicating short to battery fault currently present on upstream oxygen sensor signal					
LV_CDN_DIAG_SCP_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor upstream diagnosis short to plus					
LV_END_DIAG_SCP_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of upstream oxygen sensor short to plus diagnosis					

Input data:

VLS_UP[NC_CBK_EX_NR]	LV_IGK	LV_INH_DIAG_SCP_LS_UP[NC_CBK_EX_NR]	LV_DC

Description:

Should a hard short to Vbatt (i.e. approx. 0 Ω) exist on the oxygen sensor signal line, the signal voltage will tend to Vbatt and the analog digital converter value will tend to 5 V. This

effect does not occur under normal conditions as the sensor delivers typically 1 V for rich AFR at the sensor location and may be used for short to Vbatt detection.
In order to determine the electrical fault, short circuit to Vbatt, the oxygen sensor signal voltage (VLS_UP[i]) is compared to a calibratable threshold (C_VLS_MAX_DIAG_ SCP_LS_UP) in conjunction with the aforementioned activation conditions.
If the oxygen signal voltage exceeds the said threshold, a signal short to Vbatt shall be determined to be present and the debounce counter incremented otherwise the debounce counter shall be decremented.
The error flag LV_ERR_SCP_LS_UP[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_SCP_LS_UP) shall be initialised to 0 at LV_IGK = 0 -> 1 ot reset or by clearing the error memory

LV_END_DIAG_SCP_LS_UP is initialised with 0 at every (LV_IGK = 0 ->1 and LV_DC = 0) or reset or at clearing the error memory

Recurrence:

The oxygen sensor analog input signal shall be measured every 10 ms. This electrical diagnosis shall be carried out every *T_SAMPLE = 100 ms* provided no debounced fault has been detected and unless specifically noted otherwise in the relevant section.

Activation / Deactivation:

If LV_IGK = 1
and LV_INH_DIAG_SCP_LS_UP[i] = 0
then ***"Diagnosis available"***
LV_CDN_DIAG_SCP_LS_UP[i]= 1
else ***"Diagnosis NOT available"***
LV_CDN_DIAG_SCP_LS_UP[i]= 0
endif

Formula section:

If LV_CDN_DIAG_SCP_LS_UP[i] = 1
then **If** VLS_UP[i] > C_VLS_MAX_DIAG_SCP_LS_UP
then ***"Short to Vbatt fault present"***
Increment ABC counter
ERR_SYM_SCP_LS_UP[i] = sym_0 /"SHORT TO PLUS"
else ERR_SYM_SCP_LS_UP[i] = "NO_SYM"
endif
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_VLS_MAX_DIAG_SCP_LS_UP	1	0...3FFH	0…4.995117	4.883 e-3	V
Oxygen sensor voltage threshold over which short to Vbatt present					

Diagnosis	Symptom	Nr	ABC type
Upstream Oxygen Sensor Signal	Sym_0	0	MEM
SCP_LS_UP[i]			

A.76.3 Open circuit (line break) in connection to oxygen sensor element.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_OC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, open line fault present on upstream oxygen sensor signal					
ERR_SYM_OC_LS_UP[NC_CBK_EX_NR]	V/O	0H 4H	NO_SYM Sym_2	1	-
Variable indicating open circuit fault currently present on upstream oxygen sensor signal					
LV_CDN_DIAG_OC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor upstream diagnosis open line					
CTR_R_UPD_DIAG_OC_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Value of handshake counter for last vlaid resistance value used for open circuit diagnosis					
LV_END_DIAG_OC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of upstream oxygen sensor open line diagnosis					
CTR_DIAG_OC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Minimum detection counter for OC error symptom detection					

Input data:

VLS_UP[NC_CBK_EX_NR]	LV_LAM_LSCL[NC_CBK_EX_NR]	TEG_CAT_UP_MDL[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]
LV_INH_DIAG_OC_LS_UP[NC_CBK_EX_NR]	LV_IGK	R_IT_LS_UP[NC_CBK_EX_NR]	STATE_LSH_UP[NC_CBK_EX_NR]
CTR_CYCNR_R_IT_LS_UP_VLD[NC_CBK_EX_NR]	LV_DC		

Description:

Should a complete open circuit in the oxygen sensor signal or return line occur, the measured voltage at the input to the microcontroller will tend to the voltage determined by the operative readiness potential divider (approx. 450 mV).
In order to determine the electrical fault, open circuit, the oxygen sensor signal voltage is compared to two calibratable thresholds in conjunction with certain activation and additional conditions. The following additional conditions shall be required:

- No fault in the appropriate upstream oxygen sensor heater circuit has been detected (LV_ERR_LSH_UP[i] = 0) or should a fault be present, that the operating temperature has been determined to be sufficient for sensor operability (TEG_CAT_UP_MDL[I] > C_TEG_CAT_MDL_DIAG_OC_LS_UP).

- The appropriate heater shall be determined to being controlled normally (STATE_LSH_UP[i] = LSH_POW_CTL)
- The lambda controller shall be active (LV_LAM_LSCL[i] = 1, possibly through forced activation).

Should the above conditions and the activation conditions have be met, then the oxygen sensor signal voltage shall be analysed to determine whether it lies within a calibratable voltage band (C_VLS_AFL_DIAG_OC_LS_UP < VLS_UP[i] < C_VLS_AFR_DIAG_OC_LS_UP). A counter (CTR_DIAG_OC_LS_UP[i]) shall keep track of the time that the signal voltage lies within this band since the above additional conditions have been determined to be true. Should the timer equal or exceed a minimum calibratable threshold (C_CTR_DIAG_OC_LS_UP), then a signal open circuit shall be determined to be present and the debounce counter shall be incremented. Should the initial conditions be met but the voltage condition not met, then the debounce counter shall be decremented.
The counter shall be reset if either the signal voltage leaves the threshold band or if the initial conditions listed above are no longer met.

Depending on the calibration of LC_R_IT_CHK_DIAG_OC_LS_UP aside monitoring lambda sensor voltage also the internal resistance value of the lambda probe shall be considered or not. If set, this provides an additional safety margin for the diagnosis for the case that lambda probe is cooling down significantly during long fuel cut-off phases or if operating temperature is not yet reached after engine start.
Note: In case of LC_R_IT_CHK_DIAG_OC_LS_UP = 1 the incrementation of CTR_DIAG_OC_LS_UP and CTR_ABC_OC_LS_UP depends on the updarate of CTR_R_UPD_DIAG_OC_LS_UP (~1 s not T_SAMPLE = 100 ms)!

It should be noted that the debounce counter shall only be incremented or decremented should the initial conditions be met. In all other cases, the debounce counter shall remain frozen.
The error flag LV_ERR_OC_LS_UP[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 at LV_IGK = 0 -> 1 ot reset or by clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence:

This electrical diagnosis shall be carried out every ***T_SAMPLE = 100 ms*** provided no debounced fault has been detected and unless specifically noted otherwise in the relevant section.

Activation / Deactivation:

If LV_IGK = 1

and LV_INH_DIAG_OC_LS_UP[i] = 0
and STATE_LSH_UP[i] = LSH_POW_CTL
and (LV_ERR_LSH_UP[i] = 0
or TEG_CAT_UP_MDL[i] > C_TEG_CAT_MDL_DIAG_OC_LS_UP)
and LV_LAM_LSCL[i] = 1
then **if** LC_R_IT_CHK_DIAG_OC_LS_UP = 1
then ***% One-off update*** CTR_R_UPD_DIAG_OC_LS_UP[i]
If CTR_CYCNR_R_IT_LS_UP_VLD[i]≠ CTR_R_UPD_DIAG_OC_LS_UP[i]
then ***"Diagnosis available"***
CTR_R_UPD_DIAG_OC_LS_UP[i]= CTR_CYCNR_R_IT_LS_UP_VLD[i]
LV_CDN_DIAG_OC_LS_UP[i] = 1
else ***"Diagnosis NOT available"***
LV_CDN_DIAG_OC_LS_UP[i] = 0
endif
else ***"Diagnosis available"***
LV_CDN_DIAG_OC_LS_UP = 1

endif
else ***"Diagnosis NOT available"***
CTR_DIAG_OC_LS_UP[i] = 0
LV_CDN_DIAG_OC_LS_UP[i] = 0
endif

Formula section:

If LV_CDN_DIAG_OC_LS_UP[i] = 1
then **If** C_VLS_AFL_DIAG_OC_LS_UP < VLS_UP[i] < C_VLS_AFR_DIAG_OC_LS_UP
then **if** LC_R_IT_CHK_DIAG_OC_LS_UP = 0
or (LC_R_IT_CHK_DIAG_OC_LS_UP = 1
and R_IT_LS_UP[i] > C_R_MIN_DIAG_OC_LS_UP)
then **If** (CTR_DIAG_OC_LS_UP[i] ≥ C_CTR_DIAG_OC_LS_UP)
then "Open circuit fault present"
ERR_SYM_OC_LS_UP[i] = "SYM_2"
/the ERRM shall automatically debounces this symptom
else ERR_SYM_OC_LS_UP[i] = "NO_SYM"
endif
CTR_DIAG_OC_LS_UP[i] = CTR_DIAG_OC_LS_UP[i] +1

else ERR_SYM_OC_LS_UP[i] = "NO_SYM"

CTR_DIAG_OC_LS_UP[i] = 0

endif

else ERR_SYM_VLS_UP[i] = "NO_SYM"

CTR_DIAG_OC_LS_UP[i] = 0

endif

else *freeze* ERR_SYM_VLS_UP[i]

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_R_IT_CHK_DIAG_OC_LS_UP	1	0...1H	0...1	1	-
Configuration switch to enable internal resistance consideration for OC diagnosis					
C_R_MIN_DIAG_OC_LS_UP	1	0...FFFFH	0...65535	1	Ohm
Minimum resitance threshold oxygen sensor internal resistance during open circuit sensor					
C_CTR_DIAG_OC_LS_UP	1	0...FFH	0...255	1	-
Minimum required duration where voltage lies within voltage band to detect fault					
C_VLS_AFL_DIAG_OC_LS_UP	1	0...3FFH	0…4.995117	4.883 e-3	V
Oxygen sensor voltage-lean mixture threshold for diagnosis					
C_VLS_AFR_DIAG_OC_LS_UP	1	0...3FFH	0…4.995117	4.883 e-3	V
Oxygen sensor voltage-rich mixture threshold for diagnosis					
C_TEG_CAT_MDL_DIAG_OC_LS_UP	1	0...7FF0H	0...2047	0.0625	°C
Minimum required exhaust gas temperature for diagnosis					

Diagnosis	Symptom	Nr	ABC type
Upstream Oxygen Sensor Signal	SYM_2	*0*	MEM
*OC*_LS_UP [i]			

A.76.4 Sensor not ready within time after operating conditions have been met

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_ERR_READY_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean error flag, upstream oxygen sensor not ready in time					
ERR_SYM_READY_LS_UP[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom describing the delayed readiness					
LV_CDN_DIAG_READY_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Status conditions of upstream oxygen sensor delayed readiness diagnosis					
LV_END_DIAG_READY_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Bit to indicate end of upstream oxygen sensor delayed readiness diagnosis					

Input data:

LV_IGK	LV_LS_UP_READY_CDN[NC_CBK_EX_NR]	LV_INH_DIAG_READY_LS_UP[NC_CBK_EX_NR]	LV_LS_UP_READY[NC_CBK_EX_NR]
LV_T_TOUT_LS_UP_READY[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_DC	

Description:

The diagnosis shall monitor if the lambda sensor gets ready in time, if it was disabled due to operating conditions as defined in the operative readiness determination. It evaluates two flags defined there. LV_LS_UP_READY[i] is set while suitable operating conditions are met and the sensor is operational, while LV_T_TOUT_LS_UP_READY[i] is set if suitable operating conditions are present for a certain time. An error is determined to be present if the second flag is set but the first one is not.

No filtering shall be applied to the symptoms.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 at LV_IGK = 0 -> 1 or reset or by clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence: ***T_sample = 100ms***

Activation / Deactivation:

LV_ST_END = 1 **and**

LV_INH_DIAG_READY_LS_UP[i] = 0 **and**

% don't run the check if the sensor is on limited operability (an error should already be entered):

LV_LS_UP_READY_CDN[i] = 0 **and**

% a diagnosis result is available once each time the sensor warmup (should have) occurred once:

LV_T_TOUT_LS_UP_READY[i] 0 → 1

Formula section:

% If the activation conditions are fulfilled, just test if the sensor is ready:

LV_END_DIAG_READY_LS_UP[i] = 1

If LV_LS_UP_READY[i] = 0

then

% sensor not ready in time. Set the error flags

LV_ERR_READY_LS_UP[i] = 1

ERR_SYM_READY_LS_UP[i] = SYM_3

else

% sensor ready in time.

LV_ERR_READY_LS_UP[i] = 0

ERR_SYM_READY_LS_UP[i] = NO_SYM

endif

Calibration data:

Diagnostic	Symptom description	Symptom	Filter type
READY_LS_UP[i]	Delayed upstream sensor readiness bank i	SYM_3	*NO*

A.77 Signal plausibility monitoring

A.77.1 Overrun fuel cut-off (PUC) oxygen sensor signal plausibility.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]	V/O/S	0...1H	0...1	1	-
Boolean error flag, PUC plausibility fault currently present on upstream oxygen sensor signal					
ERR_SYM_PUC_LS_UP[NC_CBK_EX_NR]	V/O	0H 4H	NO_SYM SYM_2	1	-
Variable indicating plausibility fault currently present on upstream oxygen sensor signal					
LV_CDN_DIAG_PUC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor upstream PUC plausibility diagnosis					
LV_END_DIAG_PUC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of complete upstream oxygen sensor signal PUC plausibility diagnosis					

Input data:

VLS_UP[NC_CBK_EX_NR]	LV_LS_UP_READY[NC_CBK_EX_NR]	LV_PUC	LV_ERR_LSH_UP[NC_CBK_EX_NR]
TEG_CAT_UP_MDL[NC_CBK_EX_NR]	MAF_INT_PUC	LV_ERR_EL_LS_UP[NC_CBK_EX_NR]	NC_CBK_EX_NR
STATE_LSH_UP[NC_CBK_EX_NR]	C_TEG_CAT_MDL_DIAG_OC_LS_UP	LV_INH_DIAG_PUC_LS_UP[NC_CBK_EX_NR]	LV_IGK
LV_DC			

Description:

Due to possible oxygen sensor defects or faults in the injection system (e.g. leaking fuel injector), the oxygen sensor may not provide the expected lean AFR signal level during an overrun fuel cut-off (PUC) condition.

Hence, in order to determine the signal fault, the oxygen sensor signal shall be checked for plausibility during the engine operating state PUC. The oxygen sensor signal voltage (VLS_UP[i]) shall be required to exceed the calibratable threshold (C_VLS_MIN_DIAG_PUC_LS_UP), and the following activation conditions shall be met::

- No electrical fault shall be present on the O_2 sensor under test (LV_ERR_EL_LS_UP[i] = 0)
- Sensor shall be in a state of operative readiness (LV_LS_UP_READY[i] =1).
- The sensor shall be determined to be at operating temperature (STATE_LSH_UP[i] = LSH_POW_CTL & LV_ERR_LSH_UP[i] = 0) or should a fault exist in the heater circuit of the particular sensor under test, then a minimum Exhaust Gas Temperature (EGT) shall be exceeded (TEG_CAT_UP_MDL[i] > C_TEG_CAT_MDL_DIAG_OC_LS_UP).
- The engine state PUC (overrun fuel cut-off) shall be determined to be active (LV_PUC = 1) and stable, as determined by comparing the mass air flow integral calculated after PUC activation (MAF_INT_PUC) within the thresholds (C_MAF_INT_MIN_DIAG_PUC_LS_UP; C_MAF_INT_MAX_DIAG_PUC_LS_UP).

If the above listed conditions and the voltage condition are met then an implausible oxygen sensor signal PUC voltage shall be determined to be present and the debounce counter incremented. Should the activation conditions be met but the voltage condition not be met then the debounce counter shall be decremented. Should any of the activation conditions not be met, then the debounce counter shall remain frozen.

The error flag LV_ERR_PUC_LS_UP[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 at LV_IGK = 0 -> 1 ot reset or by clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence: ***T_sample = 100ms***

Activation / Deactivation:

If LV_INH_DIAG_PUC_LS_UP[i] = 0
and LV_ERR_EL_LS_UP[i] = 0
and LV_LS_UP_READY[i] = 1
and STATE_LSH_UP[i] = LSH_POW_CTL
and (LV_ERR_LSH_UP[i] = 0
or TEG_CAT_UP_MDL[i] > C_TEG_CAT_MDL_DIAG_OC_LS_UP)
and LV_PUC = 1
and C_MAF_INT_MAX_DIAG_PUC_LS_UP > MAF_INT_PUC > C_MAF_INT_MIN_DIAG_PUC_LS_UP
then ***"Diagnosis available"***
LV_CDN_DIAG_PUC_LS_UP[i] = 1
else ***"Diagnosis NOT available"***
LV_CDN_DIAG_PUC_LS_UP[i] = 0
endif

Formula section:

Detection: (Assumes "Diagnosis available")

If LV_CDN_DIAG_PUC_LS_UP[i] = 1
then **If** VLS_UP[i] > C_VLS_MIN_DIAG_PUC_LS_UP
then **"PUC signal plausibility fault present"**
Increment ABC counter
ERR_SYM_PUC_LS_UP[i] = sym_2 / "SENSOR SIGNAL PLAUSIBILITY"
else **Decrement** ABC counter
ERR_SYM_PUC_LS_UP[i] = "NO_SYM"

endif

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_MAF_INT_MIN_DIAG_PUC_LS_UP	1	0...FFFFH	0...1820.393	0.028	g
Minimum threshold for MAF integral required to start upstream PUC signal plausibility					
C_MAF_INT_MAX_DIAG_PUC_LS_UP	1	0...FFFFH	0...1820.393	0.028	g
Maximum threshold for MAF integral required to start upstream PUC signal plausibility					
C_VLS_MIN_DIAG_PUC_LS_UP	1	0...3FFH	0…4.995117	4.883 e-3	V
VLS voltage threshold for PUC fault detection					

NOTE: Calibration data C_TEG_CAT_MDL_DIAG_OC_LS_UP shall be obtained from chapter above.

Diagnosis	Symptom	Nr	ABC type
Upstream Oxygen Sensor Signal Plausibility	*Sym_2*	*0*	MEM
PUC_LS_UP[i]			

A.77.2 Oxygen sensor signal voltage excursion plausibility

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_STK_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, voltage excursion plausibility fault currently present on upstream oxygen sensor signal					
ERR_SYM_STK_LS_UP[NC_CBK_EX_NR]	V/O	0H 8H	NO_SYM SYM_3	1	-
Variable indicating voltage excursion plausibility fault currently present on upstream oxygen sensor signal					
LV_CDN_DIAG_STK_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status conditions of oxygen sensor upstream voltage excursion plausibility diagnosis					
LV_END_DIAG_STK_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of complete upstream oxygen sensor signal voltage excursion plausibility diagnosis					
CTR_LAM_P_DIAG_STK_LS_UP[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
Counter indicating the number of observed p jumps reported by the lambda controller					
VLS_UP_MMV_DIF_SAVE[NC_CBK_EX_NR]	V/O/S	0…FFFFH	0…4.999924	7.6294E-5	V
Difference between Min / max oxygen sensor voltage mean value for scan tool					

Input data:

FAC_LAM_MIN[NC_CBK_EX_NR]	FAC_LAM_LIM[NC_CBK_EX_NR]	FAC_LAM_MAX[NC_CBK_EX_NR]	LV_ERR_EL_LS_UP[NC_CBK_EX_NR]
T_AFL_CYC[NC_CBK_EX_NR]	T_AFR_CYC[NC_CBK_EX_NR]	LV_VLS_UP_MMV_STK_VLD[NC_CBK_EX_NR]	CTR_RAF_CHG[NC_CBK_EX_NR]
LV_LAM_LSCL[NC_CBK_EX_NR]	VLS_UP_MMV_MAX[NC_CBK_EX_NR]	VLS_UP_MMV_MIN[NC_CBK_EX_NR]	LV_INH_DIAG_STK_LS_UP[NC_CBK_EX_NR]
STATE_LSH_UP[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]	TEG_CAT_UP_MDL[NC_CBK_EX_NR]	C_TEG_CAT_MDL_DIAG_OC_LS_UP
NC_CBK_EX_NR	LV_IGK	LV_DC	

Description:

Should the oxygen sensor internal resistance be too high or a leakage resistance exist to either GND or Vbatt on the oxygen sensor signal line, the signal voltage excursion may not be sufficient to guarantee correct lambda control.

In order to determine the signal fault, voltage excursion implausible, the difference in the measured voltage between rich AFR (VLS_UP_MMV_MAX[i]) and lean AFR (VLS_UP_MMV_MIN[i]) is compared to a calibratable threshold (C_VLS_STK_DIAG_STK_LS_UP) in conjunction with the following activation conditions.

- No electrical fault shall be present on the O_2 sensor under test (LV_ERR_EL_LS_UP[i] = 0)
- The sensor shall be determined to be at operating temperature (STATE_LSH_UP[i] = LSH_POW_CTL & LV_ERR_LSH_UP[i] = 0) or should a fault exist in the heater circuit of the particular sensor under test, then a minimum Exhaust Gas Temperature (EGT) shall be exceeded (TEG_CAT_UP_MDL[i] > C_TEG_CAT_MDL_DIAG_OC_LS_UP).
- A minimum number of lambda controller P-jumps (C_CTR_LAM_P_DIAG_STK_LS_UP) since each activation of the lambda controller shall have been executed. The P-jump counter shall be reset whenever the lambda controller values VLS_UP_MMV_MAX[i] & VLS_UP_MMV_MIN[i] are reset. This shall ensure that the values used in the voltage condition are and remain current.
- No lambda limit present. That shall exclude a failure entry in case of low amplitude because of lambda limit or electrical error of upstream O2 sensor (SCG or OL).
- The cycle time counters T_AFL_CYC[i] and T_AFR_CYC[i] must not exceed a calibratable threshold. This is to prevent a stroke error entry due to the diminishing moving mean value for long cycle times.

If the above listed conditions and the voltage condition are met then an implausible oxygen sensor signal voltage excursion shall be determined to be present and the debounce counter incremented. Should the activation conditions be met but the voltage condition not be met then the debounce counter shall be decremented. Should any of the activation conditions not be met, then the debounce counter shall remain frozen.

The error flag LV_ERR_STK_LS_UP[i] shall be set once the anti-bounce counter has reached its maximum value and reset should the anti-bounce counter reach 0.

The variables shall be defined in accordance with the exhaust gas system for [i] = 1 to NC_CBK_EX_NR.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) shall be initialised to 0 at LV_IGK = 0 -> 1 ot reset or by clearing the error memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

VLS_UP_MMV_DIF_SAVE[i] shall be initialized with its saved value. Reset only at clearing error memory.

Recurrence: ***T_sample = 1000 ms***

Activation / Deactivation:

If LV_LAM_LSCL[i] = 1
then **If** $CTR\_RAF\_CHG[I]_n \neq CTR\_RAF\_CHG[I]_{n-1}$
then CTR_LAM_P_DIAG_STK_LS_UP[i] =
CTR_LAM_P_DIAG_STK_LS_UP[i] + 1
else CTR_LAM_P_DIAG_STK_LS_UP[i] =
CTR_LAM_P_DIAG_STK_LS_UP[i]
endif
else CTR_LAM_P_DIAG_STK_LS_UP[i] = 0
endif

Note: CTR_LAM_P_DIAG_STK_LS_UP[i] shall not be restarted at 0 at overflow, but stay at maximum value. For reset see also DESCRIPTION.

If LV_INH_DIAG_STK_LS_UP[i] = 0
and LV_ERR_EL_LS_UP[i] = 0
and STATE_LSH_UP[i] = LSH_POW_CTL
and (LV_ERR_LSH_UP[i] = 0
or TEG_CAT_UP_MDL[i] > C_TEG_CAT_MDL_DIAG_OC_LS_UP)
and CTR_LAM_P_DIAG_STK_LS_UP[i] > C_CTR_LAM_P_DIAG_STK_LS_UP
and LV_VLS_UP_MMV_STK_VLD[i] = 1
and FAC_LAM_MIN[i] < FAC_LAM_LIM[i] < FAC_LAM_MAX[i]
and T_AFL_CYC[i] < C_T_CYC_MAX_DIAG_STK_LS_UP
and T_AFR_CYC[i] < C_T_CYC_MAX_DIAG_STK_LS_UP
then "Diagnosis available"
LV_CDN_DIAG_STK_LS_UP[i] = 1
else "Diagnosis NOT available"
LV_CDN_DIAG_STK_LS_UP[i] = 0
endif

Formula section:

If LV_CDN_DIAG_STK_LS_UP[i] = 1
then **If** (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) <
C_VLS_STK_DIAG_STK_LS_UP

then **“Voltage excursion plausibility fault present”**

Increment ABC counter

ERR_SYM_STK_LS_UP[i] = sym_3 / "SENSOR SIGNAL EXCURSION"

else **Decrement** ABC counter

ERR_SYM_STK_LS_UP[i] = "NO_SYM"

endif

endif

NOTE: The difference (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) shall be limited to 0 to prevent occurrence of negative values!

If LV_END_DIAG_STK_LS_UP[i] = 0 to 1

then VLS_UP_MMV_DIF_SAVE[i] = (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i])

% Mode 06 test value shall be updated only once

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_CTR_LAM_P_DIAG_STK_LS_UP	1	0...FFH	0...255	1	-
Number of P-Jumps after LSCL=1 for diagnosis begin					
C_VLS_STK_DIAG_STK_LS_UP	1	0...FFFFH	0...4.99	76.3e-6	V
Difference threshold of the L-mean value for error detection					
C_T_CYC_MAX_DIAG_STK_LS_UP	1	0...FFH	0...2.55	10e-3	s
Maximum cycle time to activate the diagnosis					

NOTE: Calibration data C_TEG_CAT_MDL_DIAG_OC_LS_UP shall be obtained from chapter above.

Diagnosis	Symptom	Nr	ABC type
Upstream Oxygen Sensor Signal Plausibility	*Sym_0*	*0*	MEM
STK_LS_UP[i]			

A.78 Application Incidences for upstream O2-Sensor Signal Monitoring

A.78.1.1 Application conditions for the O2 sensor diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_OC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for LS_UP open line diagnosis not met					
LV_INH_DIAG_SCG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for LS_UP short to ground diagnosis not met					
LV_INH_DIAG_SCP_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for LS_UP to plus diagnosis not met					
LV_INH_DIAG_READY_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean flag indicating project specific application conditions for LS_UP delayed readiness diagnosis not met					
LV_INH_DIAG_STK_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for STK_LS_UP diagnosis not met					
LV_INH_DIAG_PUC_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating project specific application conditions for PUC_LS_UP diagnosis not met					

Input data:

NC_CBK_EX_NR	LV_IGK		LV_ERR_CPS
LV_ERR_IV[NC_CYL_NR]	LV_ERR_MEC_CPS	LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]
LV_ERR_FSD_LAM_LIM_i	LV_ERR_STK_LS_UP	LV_ERR_FRQ_LS_UP	LV_ERR_SWT_LS_UP
LV_MIS_STATE_A	LV_MIS_STATE_B	CONF_LAM	LV_CDN_VB_OBD1
CP_STATE	LAM_DIF_CP		

FUNCTION DESCRIPTION:

General information:

According the exhaust gas system shall the variables be defined from [i]=1 to NC_CBK_EX_NR.

Description:

The application incidences is applicable to each upstream oxygen sensor signal monitoring diagnosis function.

Note: The status of the Emissions Testing LV_ET_ACT shall be taken into account, altough this is not explicitly included in the formula section in order to make this specification more flexible. Should an emissions test be determined to be active, then no further STK diagnosis shall take place as false air is introduced into the system and may cause a lean mixture at the oxygen sensor location to be measured.

Application conditions:

Initialisation: *All inhibition flags shall be initialised to 1 (inhibit diagnosis until conditions met) at the beginning of a new driving cycle.*

Recurrence: *100ms*

Activation / Deactivation: **if** LV_IGK = 1

then *enable corresponding diagnosis*

else *disable corresponding diagnosis*

Formula section:

Inhibition Conditions for OC/SCP/SC_LS_UP Diagnoses:

if LV_IGK = 1
and CONF_LAM = 1
and LV_CDN_VB_OBD1 = 1
then LV_INH_DIAG_OC_LS_UP[i] = 0
LV_INH_DIAG_SCP_LS_UP[i] = 0
LV_INH_DIAG_SCG_LS_UP[i] = 0
else LV_INH_DIAG_OC_LS_UP[i] = 1
LV_INH_DIAG_SCP_LS_UP[i] = 1
LV_INH_DIAG_SCG_LS_UP[i] = 1

Inhibition Conditions for READY_LS_UP Diagnoses:

if LV_IGK = 1
and CONF_LAM = 1
and LV_ERR_LSH_UP[i] = 0
then LV_INH_DIAG_READY_LS_UP[i] = 0
else LV_INH_DIAG_READY_LS_UP[i] = 1

Inhibition Conditions for PUC_LS_UP Diagnosis:

if LV_ERR_CPS = 1 **or**
LV_ERR_IV[NC_CYL_NR] = 1 **or**
LV_ERR_MEC_CPS = 1 **or**
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR] = 1 **or**
LV_ERR_STK_LS_UP[i] = 1 **or**
LV_ERR_FRQ_LS_UP[1] = 1 **or**
LV_ERR_SWT_LS_UP[i] = 1 **or**
LV_MIS_STATE_A = 1 **or**
LV_MIS_STATE_B = 1 **or**
CONF_LAM = 0
then LV_INH_DIAG_PUC_LS_UP[i] = 1
else LV_INH_DIAG_PUC_LS_UP[i] = 0

Inhibition Conditions for STK_LS_UP Diagnosis:

if LV_ERR_PUC_LS_UP[NC_CBK_EX_NR] = 1 **or**
LV_ERR_FSD_LAM_LIM_i = 1 **or**
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR] = 1 **or**
CONF_LAM = 0 **or**
[(CP_STATE = RAMP_OPEN or MIN_PUREG) **and** LAM_DIF_CP > 0]
then LV_INH_DIAG_STK_LS_UP[i] = 1

else LV_INH_DIAG_STK_LS_UP[i] = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_MAX_OC_LS_UP	1	0...FFH	0...255	1	-
Debounce counter maximum for electrical check open line					
C_ABC_INC_OC_LS_UP	1	0...FFH	0...255	1	-
Debounce counter increment for electrical check ine					
C_ABC_MAX_SCG_LS_UP	1	0...FFH	0...255	1	-
Debounce counter maximum for electrical check short to ground					
C_ABC_INC_SCG_LS_UP	1	0...FFH	0...255	1	-
Debounce counter increment for electrical check short to ground					
C_ABC_MAX_SCP_LS_UP	1	0...FFH	0...255	1	-
Debounce counter maximum for electrical check short to plus					
C_ABC_INC_SCP_LS_UP	1	0...FFH	0...255	1	-
Debounce counter increment for electrical check short to plus					
C_ABC_MAX_PUC_LS_UP	1	0...FFH	0...255	1	-
Debounce counter maximum for PUC plausibility					
C_ABC_INC_PUC_LS_UP	1	0...FFH	0...255	1	-
Debounce counter increment for PUC plausibility					
C_ABC_MAX_STK_LS_UP	1	0...FFH	0...255	1	-
Debounce counter maximum for voltage excursiosibility					
C_ABC_INC_STK_LS_UP	1	0...FFH	0...255	1	-
Debounce counter increment for voltage excursion plausibility					

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_OC_LS_UP_1		SYM_0	*MEM*
		SYM_1	
	OL	SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_PUC_LS_UP_1		SYM_0	*MEM*
		SYM_1	
	Plausibility fault	SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_SCG_LS_DOWN_1		SYM_0	*MEM*
	SCG	SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_SCG_LS_UP_1		SYM_0	*MEM*
	SCG	SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_SCP_LS_UP_1	SCB	SYM_0	*MEM*
		SYM_1	

		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_STK_LS_UP_1		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Sensor signal excursion	SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_READY_LS_UP_1		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Plausibility fault	SYM_3	

A.78.1.2 RBM interface for the O2 sensor signal excursion plausibility

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
STATE_RBM_STK_LS_UP[1]	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor of O2 sensor signal excursion plausability for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_STK_LS_UP[1]	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of TCO Signal Stuck diagnosis, for application purposes					

Input data:

LV_END_DIAG_STK_LS_UP[1]	CTR_COMP_RBM_STK_LS_UP[1]	CTR_CDN_RBM_STK_LS_UP[1]	LV_DC
LV_ERR_SCP_LS_UP[1]	LV_ERR_SCG_LS_UP[1]	LV_ERR_OC_LS_UP[1]	
CTR_ERR_DYN_NR			

FUNCTION DESCRIPTION:

General information:

With this module the interface between the O2 sensor signal excursion plausability and the Rate-Based Monitoring statistics is defined with STATE_RBM_STK_LS_UP[1] data.

Within STATE_RBM_STK_LS_UP[1], three different informations are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
  (general information depending on monitor could be added)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
  (general information depending on monitor could be added)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (general information depending on monitor could be added)

Application conditions:

Initialisation :

at ECU reset: bit 0, bit 1 and bit 2 of STATE_RBM_STK_LS_UP[1] = 0

at transition LV_DC 0 → 1: bit 0, bit 1 and bit 2 of STATE_RBM_STK_LS_UP[1] = 0

on failure memory reset : bit 1 of STATE_RBM_STK_LS_UP[1] = 0

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_SCP_LS_UP[1], LV_ERR_SCG_LS_UP[1], LV_ERR_OC_LS_UP[1]

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1) **While** bit 1 of STATE_RBM_STK_LS_UP[1] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2) bit 1 of STATE_RBM_STK_LS_UP[1] = 1

Endwhile

Else(1) { the dynamic failure memory is empty }

No action

Every 1 s :

If bit 0 of STATE_RBM_STK_LS_UP[1] = 0

Then **If** LV_END_DIAG_STK_LS_UP[1] = 1

Then bit 0 of STATE_RBM_STK_LS_UP[1] = 1

If bit 1 of STATE_RBM_STK_LS_UP[1] = 0

Then **If** LV_ERR_SCP_LS_UP[1] = 1 **or**

LV_ERR_SCG_LS_UP[1] = 1 **or**

LV_ERR_OC_LS_UP[1] = 1

Then bit 1 of STATE_RBM_STK_LS_UP[1] = 1

No specific condition for STK_LS_UP[1] :

bit 2 of STATE_RBM_STK_LS_UP[1] always equal to 1

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_STK_LS_UP[1]_DIAG = 0

then RATE_RBM_STK_LS_UP[1] = FFFFH

else RATE_RBM_STK_LS_UP[1] = (CTR_COMP_RBM_STK_LS_UP[1] / CTR_CDN_RBM_STK_LS_UP[1])

A.78.1.3 RBM interface for the O2 sensor signal plausability in Fuel Cut-Off

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
STATE_RBM_PUC_LS_UP[1]	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor of O2 sensor signal plausability in PUC for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_PUC_LS_UP[1]	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of TCO Signal Stuck diagnosis, for application purposes					

Input data:

LV_END_DIAG_PUC_LS_UP[1]	CTR_COMP_RBM_PUC_LS_UP[1]	CTR_CDN_RBM_PUC_LS_UP[1]	LV_DC
LV_ERR_SCP_LS_UP[1]	LV_ERR_SCG_LS_UP[1]	LV_ERR_OC_LS_UP[1]	
CTR_ERR_DYN_NR	LV_ERR_LSH_UP[1]		

FUNCTION DESCRIPTION:

General information:

With this module the interface between the O2 sensor signal plausability in PUC and the Rate-Based Monitoring statistics is defined with STATE_RBM_PUC_LS_UP[1] data.

Within STATE_RBM_PUC_LS_UP[1], three different informations are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
  (general information depending on monitor could be added)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
  (general information depending on monitor could be added)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (general information depending on monitor could be added)

Application conditions:

Initialisation :

at ECU reset : bit 0, bit 1 and bit 2 of STATE_RBM_PUC_LS_UP[1] = 0

at transition LV_DC 0 → 1: bit 0, bit 1 and bit 2 of STATE_RBM_PUC_LS_UP[1] = 0

on failure memory reset : bit 1 of STATE_RBM_PUC_LS_UP[1] = 0

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_SCP_LS_UP[1], LV_ERR_SCG_LS_UP[1], LV_ERR_OC_LS_UP[1], LV_ERR_LSH_UP[1]

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1) **While** bit 1 of STATE_RBM_PUC_LS_UP[1] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2) bit 1 of STATE_RBM_PUC_LS_UP[1] = 1

Endwhile

Else(1) { the dynamic failure memory is empty }

No action

<u>*Every 1 s :*</u>

If bit 0 of STATE_RBM_PUC_LS_UP[1] = 0

Then **If** LV_END_DIAG_PUC_LS_UP[1] = 1

Then bit 0 of STATE_RBM_PUC_LS_UP[1] = 1

If bit 1 of STATE_RBM_PUC_LS_UP[1] = 0

Then **If** LV_ERR_SCP_LS_UP[1] = 1 **or**

LV_ERR_SCG_LS_UP[1] = 1 **or**

LV_ERR_OC_LS_UP[1] = 1 **or**

LV_ERR_LSH_UP[1] = 1

Then bit 1 of STATE_RBM_PUC_LS_UP[1] = 1

<u>No specific condition for STK_LS_UP[1] :</u>

bit 2 of STATE_RBM_PUC_LS_UP[1] always equal to 1

<u>***Calculation of actual RBM rate (for application purposes)***</u>

If CTR_CDN_RBM_PUC_LS_UP[1]_DIAG = 0

then RATE_RBM_PUC_LS_UP[1] = FFFFH

else RATE_RBM_PUC_LS_UP[1] = (CTR_COMP_RBM_PUC_LS_UP[1] / CTR_CDN_RBM_PUC_LS_UP[1])

A.78.1.4 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VLS_UP_MIN_TOT_DC	V/O/S	0...3FFH	0...4.99512	0.0048828	V
Former / current driving cycle minimum upstream oxygen sensor voltage					
VLS_UP_MAX_TOT_DC	V/O/S	0...3FFH	0...4.99512	0.0048828	V
Former / current driving cycle maximum upstream oxygen sensor voltage					

Input data:

VLS_UP[NC_CBK_EX_NR]	LV_IGK		

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

VLS_UP_MIN_TOT_DC = 3FFH

VLS_UP_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: 10ms

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If VLS_UP[1] < VLS_UP_MIN_TOT_DC

Then VLS_UP_MIN_TOT_DC = VLS_UP[1]

Endif

If VLS_UP[1] > VLS_UP_MAX_TOT_DC

Then VLS_UP_MAX_TOT_DC = VLS_UP[1]

Endif

A.79 Shut off valve electrical diagnosis (SOV)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SOV	V/O	0...1H	0...1	1	-
Boolean for error currently present on shut off valve command signal.					
LV_ERR_SCG_SOV	O	0...1H	0...1	1	-
Boolean for SCG error currently present on shut off valve command signal.					
LV_ERR_SCP_SOV	O	0...1H	0...1	1	-
Boolean for SCP error currently present on shut off valve command signal.					
LV_ERR_OC_SOV	O	0...1H	0...1	1	-
Boolean for OC error currently present on shut off valve command signal.					
LV_CDN_DIAG_SOV	V/O	0...1H	0...1	1	-
Diagnosis condition SOV diagnosis					
ERR_SYM_SOV	V/O	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Detected error symptom SOV diagnosis					
LV_END_DIAG_SOV	V/O	0...1H	0...1	1	-
End of diagnosis SOV diagnosis					
LV_INH_DIAG_SOV	V/O	0...1H	0...1	1	-
Diagnosis inhibition SOV diagnosis					
CDN_DIAG_SOV	V	0...7H	0...7	1	-
Diagnosis condition for each symptom					

Input data:

LV_IGK	VB	CONF_DIAGCP	LV_ST_END
LV_DC			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the shut off valve signal from the driver which controls the shut off valve.

Description:

For error detection algorithm see “Electrical diagnosis of powerstage outputs ATIC39”.

Error-symptoms are defined to this diagnosis function as following :

```
[ | | | ]
 │  │  │  └─> Short circuit to Vbatt    SCP     (= SYM_0)
 │  │  └────> Short circuit to GND      SCG     (= SYM_1)
 │  └───────> Open load                 OL      (= SYM_2)
 └──────────> -                                 (= SYM_3)
```

Application conditions:

Initialisation: At reset or at LV_IGK = 0->1 or at clearing FMY:

- all (except LV_END_DIAG_xx) 0
- LV_INH_DIAG_SOV = 1

At reset or at LV_DC = 0->1 or at clearing FMY:

- LV_END_DIAG_xx = 0

Recurrence: 300 ms

Activation: at every engine operating state

Deactivation: activation condition not fulfilled

Formula section:

If LV_IGK = 1
and VB < C_VB_MAX_DIAG_SOV
and VB > C_VB_MIN_DIAG_SOV
and CONF_DIAGCP > 0
and LV_ST_END = 1
then LV_INH_DIAG_SOV = 0
else LV_INH_DIAG_SOV = 1
endif

- LV_INH_DIAG_SOV is used in chapter “Electrical diagnosis of power stage outputs” to activat/deactivate the ATIC39 diagnosis for SOV
- SOV diagnosis pulse time (noise treatment)

  To reduce SOV activation noise during SOV closed state, SOV diagnosis pulse time will be changed according to SOV status.

  1. SOV activation frequency at every SOV status : 7.6 Hz
  2. Normal SOV open state : Diagnosis pulse time 0.45 ms
  3. Normal SOV closed state (for EVAP, FUC_MISS_1) : Diagnosis pulse time 0 ms
  4. Electrical error at SOV closed : Diagnosis pulse time 0.45 ms

Short circuit to Vbatt:

If ERR_SYM_SOV = 1H (SYM_0)
and LV_ERR_SOV
then LV_ERR_SCP_SOV = 1
else LV_ERR_SCP_SOV = 0
endif

Short circuit to Vbatt:

If ERR_SYM_SOV = 2H (SYM_1)

and LV_ERR_SOV

then LV_ERR_SCG_SOV = 1

else LV_ERR_SCG_SOV = 0

endif

Open load:

If ERR_SYM_SOV = 4H (SYM_2)

and LV_ERR_SOV

then LV_ERR_OC_SOV = 1

else LV_ERR_OC_SOV = 0

endif

Global error bit LV_ERR_SOV:

If one of the symptoms is detected faulty after debounce, LV_ERR_SOV is set to 1. Else LV_ERR_SOV is set to 0.

Calculation of LV_END_DIAG_SOV and antibounce counter:

- see chapter "Anti-bounce Algorithm: Calculation of the end of diagnosis"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_SOV	1	0...FFH	0...255	1	-
Debounce counter increment - SOV diagnosis					
C_ABC_MAX_SOV	1	1...FFH	1...255	1	-
Debounce counter maximum value - SOV diagnosis					
C_VB_MIN_DIAG_SOV	1	0...FFH	0...26	0.102	V
Minimum Battery voltage threshold for SOV diagnosis					
C_VB_MAX_DIAG_SOV	1	0...FFH	0...26	0.102	V
Maximum Battery voltage threshold for SOV diagnosis					

A.79.1 SOV diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Evap. system leak errors	SCB	SYM_0	*STD*
	SCG	SYM_1	
	OL	SYM_2	
SOV		SYM_3	

List of Environmental Data to store in Failure Memory:: TAM
CPPWM_ENVD
DTP_ENVD
CP_STATE

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limp home for the concerned function and inhibed the concerned diagnosis).

List of impacted function and diagnosis :

In case of STG only:

- Evaporative system control (NO_PURGE)
- Evaporative system monitoring
- Downstram fuel trim regulation
- Lambda adaptation
- Idle speed adaptation
- Segment time adaptation
- ISA plausibilty diagnosis
- DTP sensor diagnosis
- Lambda limit diagnosis
- Catalist efficiency diagnosis
- O2-sensor diagnosesFuel system diagnosis
- MAP/TPS rationality check

Whatever the symptom, is disabled:
- Evaporative system monitoring

A.80 Idle Speed Control Diagnosis

A.80.1 Plausibility check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_ISC	V/O	0...01H	0...1	1	-
Boolean for fault memory “Idle Control System”					
LV_ERR_ISC_CST	V/O	0...01H	0...1	1	-
Boolean for fault memory Idle Control System: Unplausible speed at coldstart					
T_DLY_ISC_DIAG_1	V	0...FFFF H	0 ... 6553.5	0.1	s
Delay timer for diagnosis activation					
T_DLY_ISC_DIAG_2	V	0...FFFF H	0 ... 6553.5	0.1	s
Delay timer for safe ISC diagnosis					
T_DLY_ISC_DIAG_3	V	0...FF H	0 ... 25.5	0.1	s
Delay timer for safe ISC CST diagnosis – check of I term					
ERR_SYM_ISC	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom idle speed control diagnosis					
ERR_SYM_ISC_CST	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom idle speed control diagnosis at coldstart					
LV_CDN_DIAG_ISC	V/O	0...1H	0...1	1	-
Diagnosis conditions fulfilled					
LV_CDN_DIAG_ISC_CST	V/O	0...1H	0...1	1	-
Diagnosis conditions fulfilled					
N_DIF_MIN_DIAG_ISC_CST	V	0...1FE0H	0...8160	1	[rpm]
Lower engine speed deviation of idle speed control diagnosis at coldstart					
N_DIF_MAX_DIAG_ISC_CST	V	0...1FE0H	0...8160	1	[rpm]
Upper engine speed deviation of idle speed control diagnosis at coldstart					

Input data:

LV_INH_DIAG_ISC	FLOW_CPS	LV_TCO_AD_CDN_IS	N_SP_IS
N	VS	MAF	LV_ES
LV_REQ_ISC	LV_AST_END	LV_CT	LV_ST
LV_IGK	LV_MTC_CUR_OFF	T_VS_MIN_DIAG	LV_CDN_VB_OBD2
LV_N_SP_IS_CH	TCO	LV_DRI	LV_RLY_ACCOUT
TCO_ST	T_AST	LV_CH	LV_AT
TQ_DIF_I_IS	C_TQ_TOL_I_IS	C_TQ_BOL_I_IS	

FUNCTION DESCRIPTION:

General information:

Engine speed deviation from the nominal engine speed is monitored when the vehicle is stopped and the throttle is settled after closing.

If the difference to the nominal idle speed keeps too high or too low in spite of an activated idle speed controller, an idle speed control plausibility error is detected.

A second diagnosis is available for cold start idle monitoring.

Application conditions:

Initialization: *transition* LV_IGK 0 → 1 or reset
LV_ERR_ISC = 0
LV_ERR_ISC_CST = 0

ERR_SYM_ISC = NO_SYM
ERR_SYM_ISC_CST = NO_SYM
LV_CDN_DIAG_ISC = 0
LV_CDN_DIAG_ISC_CST = 0
T_DLY_ISC_DIAG_1 = 0
T_DLY_ISC_DIAG_2 = 0
T_DLY_ISC_DIAG_3 = 0

Recurrence: 100ms

Activation:

Each plausibility diagnosis is performed only if the conditions are fulfilled (CDN bit set):

IF {LV_CT = 1 (driver passive)
and (VS = 0 and T_VS_MIN_DIAG < C_T_VS_MIN_IS_DIAG)}
and LV_REQ_ISC = 1
and $LV\_DRI_{(n)} = LV\_DRI_{(n-1)}$
and $LV\_RLY\_ACCOUT_{(n)} = LV\_RLY\_ACCOUT_{(n-1)}$
THEN increment T_DLY_ISC_DIAG_1
ELSE T_DLY_ISC_DIAG_1 = 0
ENDIF

IF {
LV_IGK = 1
AND LV_ES = 0
AND LV_ST = 0
AND T_DLY_ISC_DIAG_1 >= C_T_DLY_ISC_DIAG_1
AND LV_REQ_ISC = 1
AND LV_CDN_VB_OBD2 = 1
AND FLOW_CPS < C_FLOW_CPS_MAX_DIAG_ISC
AND LV_MTC_CUR_OFF = 0
AND **LV_INH_DIAG_ISC = 0 }**

THEN
IF{
LV_TCO_AD_CDN_IS = 1 **and**
(This bit is set in the ISC adaptation condition module when TCO is within a certain range)
MAF < C_MAF_MAX_DIAG_ISC **and**
LV_AST_END = 1 **and**
LV _CH = 0 **and**
}
THEN LV_CDN_DIAG_ISC = 1

```
    ELSEIF {
            LV_CH = 1 and
            LV_ERR_ISC_CST = 0 and
            TCO > C_TCO_MIN_ISC_CST and
            T_AST > IP_T_ISC_DIAG_AST_INH(TCO) and
            (LV_AT = 1 or LC_ISC_CST_MT_ENA = 1)
    THEN    LV_CDN_DIAG_ISC_CST = 1
    ELSE
            LV_CDN_DIAG_ISC = 0
            LV_CDN_DIAG_ISC_CST = 0
            T_DLY_ISC_DIAG_2 = 0
            T_DLY_ISC_DIAG_3 = 0
    ENDIF
ELSE
            LV_CDN_DIAG_ISC = 0
            LV_CDN_DIAG_ISC_CST = 0
            T_DLY_ISC_DIAG_2 = 0
            T_DLY_ISC_DIAG_3 = 0
ENDIF
```

Formula section:

1) Idle diagnosis (only performed if LV_CDN_DIAG_ISC = 1)

```
IF          N > N_SP_IS + C_N_DIF_MAX_DIAG_ISC
THEN
        if      T_DLY_ISC_DIAG_2 >= C_T_DLY_ISC_DIAG_2
        then ERR_SYM_ISC = SYM_0
                and LV_ERR_ISC = 1        (after debounce)
            "Idle Control System RPM higher than expected" is detected
        else  increment T_DLY_ISC_DIAG_2
ELSE
        If          N < N_SP_IS - C_N_DIF_MIN_DIAG_ISC
        Then
                if          T_DLY_ISC_DIAG_2 >= C_T_DLY_ISC_DIAG_2
                then        ERR_SYM_ISC = SYM_1
                            and LV_ERR_ISC = 1          (after debounce)
            "Idle Control System RPM lower than expected" is detected
                else  increment T_DLY_ISC_DIAG_2


        Else        ERR_SYM_ISC = NO_SYM          and     T_DLY_ISC_DIAG_2 = 0
                    LV_ERR_ISC = 0                (after healing)
        End

END
```

2) Cold start Idle diagnosis (only performed if LV_CDN_DIAG_ISC_CST = 1)

N_DIF_MAX_DIAG_ISC_CST = IP_N_DIF_MAX_DIAG_ISC_CST(TCO, TCO_ST)
* IP_FAC_DIF_DIAG_ISC_CST(T_AST)

N_DIF_MIN_DIAG_ISC_CST = IP_N_DIF_MIN_DIAG_ISC_CST(TCO, TCO_ST)
* IP_FAC_DIF_DIAG_ISC_CST(T_AST)

```
IF        N > N_SP_IS + N_DIF_MAX_DIAG_ISC_CST  (overspeed)
THEN
          IF TQ_DIF_I_IS <= C_TQ_BOL_I_IS
                IF T_DLY_ISC_DIAG_3 >= C_T_DLY_ISC_DIAG_3
                THEN
                      ERR_SYM_ISC_CST = SYM_0
                      LV_ERR_ISC_CST = 1         (after debounce)
                      "Idle Control System RPM higher than expected" is detected
                ELSE
                      increment T_DLY_ISC_DIAG_3
                ENDIF
          ELSE
                T_DLY_ISC_DIAG_3 = 0
          END

ELSEIF    N < N_SP_IS - N_DIF_MIN_DIAG_ISC_CST (underspeed)
THEN
          IF TQ_DIF_I_IS >= C_TQ_TOL_I_IS
                IF T_DLY_ISC_DIAG_3 >= C_T_DLY_ISC_DIAG_3
                THEN
                      ERR_SYM_ISC = SYM_1
                      LV_ERR_ISC_CST = 1         (after debounce)
                      "Idle Control System RPM lower than expected" is detected
                ELSE
                      increment T_DLY_ISC_DIAG_3
                ENDIF
          ELSE
                T_DLY_ISC_DIAG_3 = 0
          END
ELSE
          ERR_SYM_ISC = NO_SYM
          LV_ERR_ISC = 0                  (after healing)
          T_DLY_ISC_DIAG_3 = 0

END
```

For detailed information see antibounce algorithm, respectively the generic error management.

Calibration data:

Name	Dim	Hex. limits	Phys.Lim.	Resol.	Unit
C_MAF_MAX_DIAG_ISC	-	0...FFH	0...1389	5.4471	mg/stk
Maximum mass air flow to perform idle speed Plausibility check					
C_FLOW_CPS_MAX_DIAG_ISC	V	0H...FFFFH	0...8	0.000122	kg/h
Maximum canister purge fuel flow from CPS to cylinder to perform idle speed plausibility check					
C_N_DIF_MAX_DIAG_ISC	-	0...1FE0H	0...8160	1	rpm
Maximum deviation above nominal idle speed					
C_N_DIF_MIN_DIAG_ISC	-	0...1FE0H	0...8160	1	rpm
Maximum deviation below nominal idle speed					
C_ABC_INC_ISC	-	0...FFH	0...255	1	-
Anti-bounce counter increment for idle speed Plausibility check					
C_ABC_MAX_ISC	-	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for idle speed Plausibility check					
C_ABC_DEC_ISC	-	0...FFH	0...255	1	-
Anti-bounce counter decrement for idle speed Plausibility check					
C_T_DLY_ISC_DIAG_1	1	0...FFFFH	0 ... 6553.5	0.1	s
Time delay for diagnosis activation					
C_T_DLY_ISC_DIAG_2	1	0...FFFFH	0 ... 6553.5	0.1	s
Time delay for safe ISC diagnosis					
C_T_DLY_ISC_DIAG_3	1	0...FFH	0 ... 25.5	0.1	s
Time delay for safe ISC CST diagnosis– check of I term					
C_T_VS_MIN_IS_DIAG	1	0...FFFFH	0 ... 655.35	0.01	s
Max. time with VS unplausible to detect ISA plausibility error					
C_ABC_INC_ISC_CST	1	0...FFH	0...255	1	[-]
Anti-bounce counter increment for idle speed Plausibility check at coldstart					
C_ABC_MAX_ISC_CST	1	1...FFH	1...255	1	[-]
Maximum value of the anti-bounce counter for idle speed Plausibility check at coldstart					
C_TCO_MIN_ISC_CST	1	0...FEH	-48...142.5	0.75	[°C]
Min TCO threshold for idle speed diagnosis at coldstart					
LC_ISC_CST_MT_ENA	1	0...1H	0...1	1	[-]
Flag to enable the cost start idle diagnosis in an MT vehicle					
IP_FAC_DIF_DIAG_ISC_CST	6	0...FFH	0...1.99218	0.0078125	[-]
LDP_T_AST_IP_FAC_DIAG_ISC_CST	6	0...FFFFH	0...6553.5	0.1	[s]
Influence of time after start for limits of idle speed controller diagnosis at coldstart					
IP_T_ISC_DIAG_AST_INH	6	0...FFFFH	0...6553.5	0.1	[s]
LDPM_TCO_IP_DIAG_ISC_CST	6	0...FEH	-48...142.5	0.75	[°C]
Time after start to inhibit cold start diagnosis					
IP_N_DIF_MAX_DIAG_ISC_CST	6*6	0...1FE0H	0...8160	1	[rpm]
LDPM_TCO_IP_DIAG_ISC_CST	6	0...FEH	-48...142.5	0.75	[°C]
LDPM_TCO_ST_IP_DIAG_ISC_CST	6	0...FEH	-48...142.5	0.75	[°C]
Upper engine speed deviation for limits idle speed control diagnosis at coldstart					
IP_N_DIF_MIN_DIAG_ISC_CST	6*6	0...1FE0H	0...8160	1	[rpm]
LDPM_TCO_IP_DIAG_ISC_CST	6	0...FEH	-48...142.5	0.75	[°C]
LDPM_TCO_ST_IP_DIAG_ISC_CST	6	0...FEH	-48...142.5	0.75	[°C]
Lower engine speed deviation for limits idle speed control diagnosis at coldstart					

A.81 Idle Speed Control Diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_ISC	O/V	0...1H	0...1	1	[-]
Flag for inhibitation of the Idle Speed Control Diagnosis					

Input data:

LV_ERR_CAM_DE_IVVT_EX[NC_NR_CBK_IVVT]	LV_ERR_CAM_DE_IVVT_IN[NC_NR_CBK_IVVT]	LV_ERR_CPS	LV_ERR_CRK
LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_FSD[NC_CBK_EX_NR]	LV_ERR_FSD_LAM_LIM[NC_CBK_EX_NR]	LV_ERR_ISA_i
	LV_ERR_ISC	LV_ERR_IV[NC_CYL_NR]	LV_ERR_LOAD_PLAUS
LV_ERR_LOAD_TPS_PLAUS	LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_VS	LV_ERR_MAF
LV_ERR_MAP	LV_ERR_MEC_CPS	LV_ERR_MEC_IVVT_EX[NC_NR_CBK_IVVT]	LV_ERR_MEC_IVVT_IN[NC_NR_CBK_IVVT]
LV_ERR_MWSS	LV_IGK		
LV_MIS_STATE_B	LV_ERR_TPS	LV_ERR_SLV_IVVT_EX[NC_NR_CBK_IVVT]	LV_ERR_SLV_IVVT_IN[NC_NR_CBK_IVVT]
LV_IS_AD_INH_OBD	LV_MIS_STATE_A	LV_ERR_TPS_PLAUS	

FUNCTION DESCRIPTION:

General information:

These application incidences give the possibility to define customer specific conditions for inhibiting the diagnosis of the idle speed.

Application conditions:

Initialisation: LV_INH_DIAG_ISC = 1

Recurrence: 100 ms

Activation: LV_IGK = 1

Formula section:

If LV_ERR_CAM_DE_IVVT_IN_i = 1 or
LV_ERR_CAM_DE_IVVT_EX_i = 1 or
LV_ERR_CRK = 1 or
LV_ERR_MAF = 1 or
LV_ERR_MAP = 1 or
LV_ERR_CPS = 1 or
LV_ERR_MEC_CPS =1 or

	LV_ERR_SLV_IVVT_IN_i = 1	or
	LV_ERR_SLV_IVVT_EX_i = 1	or
	LV_ERR_ISA_i = 1	or
	LV_ERR_FSD_i = 1	or
	LV_ERR_FSD_LAM_LIM_i = 1	or
	LV_MIS_STATE_A = 1	or
	LV_MIS_STATE_B = 1	or
	LV_ERR_LOAD_PLAUS = 1	or
	LV_ERR_TPS = 1	or
	LV_ERR_TPS_PLAUS = 1	or
	LV_ERR_IV = 1	or
	LV_ERR_MEC_IVVT_IN_i = 1	or
	LV_ERR_MEC_IVVT_EX_i = 1	or
	LV_ERR_MWSS = 1	or
	LV_ERR_VS = 1	or
	LV_ERR_LOAD_TPS_PLAUS = 1	or
	LV_IS_AD_INH_OBD = 1	or
	LV_ERR_LS_UP[NC_CBK_EX_NR] = 1	
Then	LV_INH_DIAG_ISC_= 1	
Else	LV_INH_DIAG_ISC = 0	
Endif		

Error treatment:

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> Idle Control System RPM higher than expected  (= SYM_0)
│ │ └───> Idle Control System RPM lower than expected   (= SYM_1)
│ └─────> -                                             (= SYM_2)
└───────> -                                             (= SYM_3)
```

Application Assistances :

The following diagnosis functions are inhibited :

- MAF / TPS Plausability
- DTP Noisy signal diagnosis
- EVAP diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Idle speed control diagnosis ERR_ISC	RPM higher than expected	SYM_0	*STD_INI*
	RPM lower than expected	SYM_1	
		SYM_2	
		SYM_3	

Diagnostic	Symptom Description	Symptom	Filter type
Idle speed control diagnosis ERR_ISC_CST	RPM higher than expected	SYM_0	*STD_INI*
	RPM lower than expected	SYM_1	
		SYM_2	
		SYM_3	

A.82 Main Relay Diagnosis (RLY_MAIN)

A.82.1 Main Relay not switched on / not switched off Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_RLY_MAIN	O/V	0...1h	0...1	1	-
Boolean for error detected at the main relay.					
T_RLY_MAIN_DIAG	V	0...FFh	0...2550	10	ms
Time window to perform the main relay diagnosis.					

Input data:

LV_IGK	V_IGK	VB	LV_RLY_MAIN_EXT_ADJ
LV_ACT_RLY_MAIN_EXT_ADJ			

FUNCTION DESCRIPTION:

General information:

The Main Relay Diagnosis is performed to detect if the Main Relay has switched and remains on after Key-On and if it has switched off after the Power Latch Phase.

Description:

The diagnosis is delayed for C_T_DLY_RLY_MAIN_DIAG to allow the relay to switch on/off. After this the actual VB is compared with a diagnostic threshold. If the error conditions are fulfilled the error is debounced.

If an error is detected after the Power Latch Phase the ECU is switched off after T_RLY_MAIN_DIAG has ellapsed.

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> Main relais has not switched off     (= SYM_0)
│ │ └───> Main relais has not switched on      (= SYM_1)
│ └─────> -                                    (= SYM_2)
└───────> -                                    (= SYM_3)
```

Application conditions:

Initialisation: restored out of the NVMY

Recurrence:

The application recurrence for the Diagnosis is **10 msec**;
The „Main Relay Off"-Detection is performed only once after power latch.

Activation: *see Formula section*

Formula section:

Detection if Main Relay is on during Key-On:

If(1) LV_RLY_MAIN_EXT_ADJ = 0

Then(1) **If(2)** LV_IGK = 1 **and**

V_IGK > C_V_IGK_RLY_MAIN_DIAG

(direct connection to the battery,
to ensure a battery voltage that would be able to switch the relay)

Then(2) start time: C_T_DLY_RLY_MAIN_DIAG
(time to allow the relay to switch, set only once. For continuos diagnosis set to 0)

If(3) C_T_DLY_RLY_MAIN_DIAG has run out *(if started, see above)*

Then(3) **If(4)** VB < C_VB_RLY_MAIN_DIAG
(connection to battery via main relay)

Then(4) LV_ERR_RLY_MAIN = 1 (after debounce)

Else(4) LV_ERR_RLY_MAIN = 0 (after rebounce)

Endif(4)

Else(3) C_T_DLY_RLY_MAIN_DIAG is still running

Endif(3)

Else(2) no Diagnosis

Endif(2)

Else(1) no Diagnosis; the main relay is controlled by LV_ACT_RLY_MAIN_EXT_ADJ

Endif(1)

<u>Detection if Main Relay is off after Power Latch Phase:</u>

If(1) Power Latch Phase has run out after Key-Off

Then(1) start time: C_T_DLY_RLY_MAIN_DIAG
(time to allow the relay to switch)

If(2) LV_IGK is still 0

Then(2) start T_RLY_MAIN_DIAG with C_T_RLY_MAIN_DIAG

(start the time window during which the diagnosis is performed)

If(3) T_RLY_MAIN_DIAG is still running **and**

VB > C_VB_RLY_MAIN_DIAG
(connection to battery via main relay)

Then(3) LV_ERR_RLY_MAIN = 1 (after debounce)

Else(3) LV_ERR_RLY_MAIN = 0 (after rebounce)

Endif(3)

Else(2) no switched off Diagnosis

Endif(2)

Else(1) no Diagnosis

Endif(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_RLY_MAIN_DIAG	1	0...FFH	0...2550	10	ms
Delay Time for the diagnosis to allow switching of the relay.					
C_V_IGK_RLY_MAIN_DIAG	1	0...FFH	0...25.8984	0.1015625	Volt
voltage threshold for detection of main relay on.					
C_VB_RLY_MAIN_DIAG	1	0...FFH	0...25.8984	0.1015625	Volt
voltage threshold for detection of battery voltage present.					
C_T_RLY_MAIN_DIAG	1	0..FFH	0...2550	10	ms
Time window to perform the main relay diagnosis					
C_ABC_INC_RLY_MAIN	1	0...FFH	0...255	1	-
Debounce counter increment – RLY_MAIN diagnosis					
C_ABC_MAX_RLY_MAIN	1	01...FFH	1...255	1	-
Debounce counter maximum value – RLY_MAIN diagnosis					

A.82.2 Main Relay switched on too slow diagnosis for pre-drive check RLY_MAIN_DLY

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_RLY_MAIN_DLY	V	0...1H	0...1	1	[-]
Diagnosis condition RLY_MAIN_DLY					
CTR_DIAG_RLY_MAIN_DLY	V	0...FFH	0...255	1	[-]
Debounce counter - main relay diagnosis for "switch on too slow" diagnosis					
CTR_END_DIAG_RLY_MAIN_DLY	V	0...FFH	0...255	1	[-]
counter for END_DIAG - main relay diagnosis for "switch on too slow" diagnosis					
ERR_SYM_RLY_MAIN_DLY	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Detected symptom RLY_MAIN_DLY					
LV_ERR_RLY_MAIN_DLY	V/O	0...1H	0...1	1	[-]
Error detected switch on relay					
LV _RLY_MAIN_DLY_ERR	V/O	0...1H	0...1	1	[-]
Error detected switch on relay (flag not for error management)					
LV_DIAG_END_RLY_MAIN_DLY	V/O	0...1H	0...1	1	[-]
End of Diagnosis RLY_MAIN_DLY (flag not for error management)					
LV_END_DIAG_RLY_MAIN_DLY	V	0...1H	0...1	1	[-]
End of Diagnosis RLY_MAIN_DLY					

Input data:

LV_IGK	V_IGK	VB	LV_RLY_MAIN_EXT_ADJ

FUNCTION DESCRIPTION:

General information:

The main relay diagnosis is performed at reset to detect if the main relay has been switched on within the correct timing. This diagnostic is comparable to main relay switch on, but is separated for having faster reaction time.

Once no failure has been found, diagnosis is then disabled.

Description:

The diagnosis starts after LV_IGK 0->1. After this the actual VB is compared with a diagnostic threshold. If the error conditions are fulfilled the error is debounced.

Error-symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
│ │ │ └─> -                                   (= SYM_0)
│ │ └───> -                                   (= SYM_1)
│ └─────> Main relay switched on too slow     (= SYM_2)
└───────> -                                   (= SYM_3)
```

Application conditions:

Initialisation: ERRM data according filter type: **NO**

At reset : LV_CDN_DIAG_RLY_MAIN_DLY = 0

LV_ERR_RLY_MAIN_DLY = 0

LV_RLY_MAIN_DLY_ERR = 0

ERR_SYM_RLY_MAIN_DLY = NO_SYM

CTR_END_DIAG_RLY_MAIN_DLY = 0

CTR_DIAG_RLY_MAIN_DLY = 0

LV_DIAG_END_RLY_MAIN_DLY = 0

LV_END_DIAG_RLY_MAIN_DLY = 0

Recurrence: 5 ms

Activation: see Formula section

Formula section:

If(1) LV_IGK = 1

and LV_RLY_MAIN_EXT_ADJ = 0 (Actuator test is not active)

and (LV_DIAG_END_RLY_MAIN_DLY = 0

Or LV_RLY_MAIN_DLY_ERR = 1) "healing should be possible"

and V_IGK > C_V_IGK_RLY_MAIN_DLY

Then(1) LV_CDN_DIAG_RLY_MAIN_DLY = 1

If(2) VB >= C_VB_RLY_MAIN_DLY_DIAG
(connection to battery via main relay)

Then(2) ERR_SYM_RLY_MAIN_DLY = NO_SYM

CTR_END_DIAG_RLY_MAIN_DLY =
CTR_END_DIAG_RLY_MAIN_DLY +
C_CTR_INC_END_DIAG_RLY_MAIN_DLY

If(3a) CTR_END_DIAG_RLY_MAIN_DLY >=
C_CTR_MAX_END_DIAG_RLY_MAIN_DLY

Then(3a) LV_END_DIAG_RLY_MAIN_DLY = 1

LV_DIAG_END_RLY_MAIN_DLY = 1

LV_ERR_RLY_MAIN_DLY = 0

LV_RLY_MAIN_DLY_ERR = 0

Endif(3a)

Else(2) ERR_SYM_RLY_MAIN_DLY = SYM_2

If(3b) CTR_DIAG_RLY_MAIN_DLY >=

C_CTR_MAX_DIAG_RLY_MAIN_DLY

Then(3b) LV_END_DIAG_RLY_MAIN_DLY = 1

LV_DIAG_END_RLY_MAIN_DLY = 1

LV_ERR_RLY_MAIN_DLY = 1

LV_RLY_MAIN_DLY_ERR = 1

CTR_END_DIAG_RLY_MAIN_DLY = 0

Else(3b) CTR_DIAG_RLY_MAIN_DLY =
CTR_ DIAG_RLY_MAIN_DLY +
C_CTR_INC_DIAG_RLY_MAIN_DLY

Endif(3b)

Else(1) LV_CDN_DIAG_RLY_MAIN_DLY = 0

CTR_END_DIAG_RLY_MAIN_DLY = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VB_RLY_MAIN_DLY_DIAG	1	0...FFH	0...25.89843	0.1015625	[V]
Voltage threshold for main relay "too slow" diagnosis					
C_CTR_INC_DIAG_RLY_MAIN_DLY	1	0...FFH	0...255	1	[-]
Debounce counter increment - main relay "too slow" diagnosis					
C_CTR_MAX_DIAG_RLY_MAIN_DLY	1	1...FFH	1...255	1	[-]
Debounce counter maximum value - main relay "too slow" diagnosis					
C_CTR_INC_END_DIAG_RLY_MAIN_DLY	1	0...FFH	0...255	1	[-]
counter increment for END_DIAG - main relay "too slow" diagnosis					
C_CTR_MAX_END_DIAG_RLY_MAIN_DLY	1	1...FFH	1...255	1	[-]
counter maximum value for END_DIAG - main relay "too slow" diagnosis					
C_V_IGK_RLY_MAIN_DLY	1	0...FFH	0...25.89843	0.1015625	[V]
voltage threshold for detection of battery voltage present for main relay "too slow" diagnosis					

A.82.3 Main relay diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Main relay diagnosis	Not switched off	SYM_0	*STD*
	Not switched on	SYM_1	
		SYM_2	
RLY_MAIN		SYM_3	

List of Environmental Data to store in Failure Memory: LV_IGK
ENG_STATE
T_AST_ENVD
VS

Diagnostic	Symptom Description	Symptom	Filter type
Main relay switched too slow diagnosis		SYM_0	*NO*
		SYM_1	
	Main relay switched too slow	SYM_2	
RLY_MAIN_DLY		SYM_3	

List of Environmental Data to store in Failure Memory: LV_IGK
ENG_STATE
T_AST_ENVD
VS

A.83 Oil Temperature Sensor Diagnosis

A.83.1 TOIL electrical diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TOIL	V/O/S	0...1H	0...1	1	-
Boolean for error currently present on oil temperature sensor signal					
LV_CDN_DIAG_TOIL	O/V	0...01H	0...1	1	-
Diagnosis condition TOIL diagnosis					
ERR_SYM_TOIL	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom TOIL diagnosis					
LV_END_DIAG_TOIL	O/V	0...01H	0...1	1	-
End of diagnosis TOIL diagnosis					

Input data:

TOIL_MES	TCO	LV_IGK	TOIL_MDL_FSD
LV_CDN_VB_MIN_DIAG	LV_ST	LV_IS	LV_PL
CONF_TOIL_MDL	TOIL	LV_ERR_TCO	

FUNCTION DESCRIPTION:

General information:

The purpose of this diagnostic function is to diagnose the oil temperature sensor signal.

Application Recurrence: **1 sec.**

Application conditions:

Activation: **If** LV_IGK = 1
and LV_CDN_VB_MIN_DIAG = 1
and CONF_TOIL_MDL = 0

Deactivation: Activation conditions not met

Formula section:

* Failure Detection:

If (1) TCO < C_TCO_MAX_TOIL_DIAG
and LV_ERR_TCO = 0
and TOIL_MES > C_TOIL_MAX_DIAG

then (1) ERR_SYM_TOIL = SYM 0 (*'Signal Line Short to Ground' is detected*)
LV_ERR_TOIL = 1 (after debounce)

else (1) ERR_SYM_TOIL = NO_SYM
LV_ERR_TOIL = 0 (after rebounce)

If (2) C_T_TCO_TOIL_DIAG has run out after transition from ST to IS or PL**)**
or (TCO >= C_TCO_MIN_TOIL_DIAG **and** LV_ERR_TCO = 0 **)**
and TOIL_MES < C_TOIL_MIN_DIAG

then (2) ERR_SYM_TOIL = SYM 1 (*'Signal Line Short to Battery' or 'Line Break' is detected*)
LV_ERR_TOIL = 1 (after debounce)

Else (2) ERR_SYM_TOIL = NO_SYM
LV_ERR_TOIL = 0 (after rebounce)

Emergency operation :

* Limp Home:

- TOIL = TOIL_MDL_FSD (from Fuel System Diagnosis)
- CAM control and short trip deviations diagnosis (P0010) disabled
- TOIL stuck and plausability diagnosis disabled

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MAX_TOIL_DIAG	1	0...FEH	-48...142,5	0,75	°C
Coolant temperature threshold for oil temperature signal diagnosis.					
C_TCO_MIN_TOIL_DIAG	1	0...FEH	-48...142,5	0,75	°C
Coolant temperature threshold for oil temperature signal diagnosis.					
C_T_TCO_TOIL_DIAG	1	1...FFFFH	0,1...6553,5	0,1	s
Time delay after start to enable the TOIL signal diagnosis					
C_TOIL_MIN_DIAG	1	0...C8H	-40...160	1	°C
Oil temperature threshold for signal line short to battery voltage or signal line break detection					
C_TOIL_MAX_DIAG	1	0...C8H	-40...160	1	°C
Oil temperature threshold for signal line short to ground detection					
C_ABC_INC TOIL	1	0...FFH	0...255	1	-
Anti-bounce counter increment for oil temperature sensor diagnosis.					
C_ABC_MAX_TOIL	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for oil temperature sensor diagnosis.					

A.83.2 TOIL stuck signal diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TOIL_STUCK	V/S/O	0...1H	0...1	1	-
Boolean for error currently present on oil temperature signal (stuck signal detected).					
LV_CDN_DIAG_TOIL_STUCK	O/V	0...01H	0...1	1	-
Diagnosis condition TOIL_STUCK diagnosis					
ERR_SYM_TOIL_STUCK	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom TOIL_STUCK diagnosis					
LV_END_DIAG_TOIL_STUCK	V/O	0...1H	0...1	1	-
Boolean for oil temperature stuck signal diagnosis finished					
TOIL_MES_DIF_DIAG	V	0...C8H	0...200	1	°C
Difference between the highest and lowest measured oil temperature (absolute value)					
TOIL_MES_MIN_DIAG	V	0...C8H	-40...160	1	°C
Lowest measured oil temperature					
TOIL_MES_MAX_DIAG	V	0...C8H	-40...160	1	°C
Highest measured oil temperature					
TOIL_MDL_DIF_DIAG	V	0...C8H	0...200	1	°C
Difference between the highest and lowest modelized oil temperature (absolute value)					
TOIL_MDL_MIN_DIAG	V	0...C8H	-40...160	1	°C
Lowest modelized oil temperature					
TOIL_MDL_MAX_DIAG	V	0...C8H	-40...160	1	°C
Highest modelized oil temperature					

Input data:

TOIL_MDL_FSD	LV_IGK	LV_CDN_VB_MIN_DIAG	TOIL_MES
TCO_ST	LV_ERR_TCO	LV_ERR_TOIL	LV_ERR_TCO_PLAUS
LV_ERR_TCO_GRD	LV_ES	LV_ST	LV_ERR_TOIL_PLAUS_L
LV_ERR_TOIL_PLAUS_H	TCO_MES	TCO_SUB	CONF_TOIL_MDL
C_TCO_TH_OPEN_DIAG_TH			

FUNCTION DESCRIPTION:

General information:

The purpose of this diagnosis is to detect a stuck oil temperature signal.

The diagnostic function checks whether after a variation of the calculated oil temperature TOIL_MDL_FSD also a variation of the measured oil temperature TOIL_MES is detected.

The diagnosis is performed only once per driving cycle.

Error-symtoms are defined to this diagnosis function as following :

```
|_|_|_|_|
 |  |  |  └─> -                        (= SYM_0)
 |  |  └────> -                        (= SYM_1)
 |  └───────> Sensor signal stuck  (= SYM_2)
 └──────────> -                        (= SYM_3)
```

Application conditions:

Initialisation (after initialisation of TOIL_MDL_FSD and TOIL_MES):

TOIL_MES_MIN_DIAG = TOIL_MES_MAX_DIAG = TOIL_MES

TOIL_MDL_MIN_DIAG = TOIL_MDL_MAX_DIAG = TOIL_MDL_FSD

TOIL_MES_DIF_DIAG = TOIL_MDL_DIF_DIAG = 0

LV_END_DIAG_TOIL_STUCK = 0

Recurrence: **1 sec**

Activation: TCO_ST < C_TCO_ST_MAX_TOIL_STUCK_DIAG

and LV_ES = 0

and LV_ST = 0

and CONF_TOIL_MDL = 0

Deactivation: Activation conditions not met

or LV_END_DIAG_TOIL_STUCK = 1

Formula section:

If LV_ERR_TCO = 0

and LV_ERR_TCO_PLAUS = 0

and LV_ERR_TCO_GRD = 0

and LV_ERR_TOIL = 0

and LV_ERR_TOIL_PLAUS_L = 0

and LV_ERR_TOIL_PLAUS_H = 0

and LV_IGK = 1

and LV_CDN_VB_MIN_DIAG = 1

then $TOIL\_MES\_MIN\_DIAG_N = \min(TOIL\_MES, TOIL\_MES\_MIN\_DIAG_{N-1})$

$TOIL\_MES\_MAX\_DIAG_N = \max(TOIL\_MES, TOIL\_MES\_MAX\_DIAG_{N-1})$

$TOIL\_MDL\_MIN\_DIAG_N = \min(TOIL\_MDL\_FSD, TOIL\_MDL\_MIN\_DIAG_{N-1})$

$TOIL\_MDL\_MAX\_DIAG_N = \max(TOIL\_MDL\_FSD, TOIL\_MDL\_MAX\_DIAG_{N-1})$

$TOIL\_MES\_DIF\_DIAG = | TOIL\_MES\_MAX\_DIAG_N - TOIL\_MES\_MIN\_DIAG_N |$

$TOIL\_MDL\_DIF\_DIAG = | TOIL\_MDL\_MAX\_DIAG_N - TOIL\_MDL\_MIN\_DIAG_N |$

If TOIL_MDL_DIF_DIAG > IP_TOIL_MDL_DIF_MIN_DIAG(TCO_ST)

and TCO_SUB > C_TCO_TH_OPEN_DIAG_TH **or** TCO_MES > C_TCO_TH_OPEN_DIAG_TH

then

if TOIL_MES_DIF_DIAG<IP_TOIL_MES_DIF_MIN_DIAG(TCO_ST)

then LV_ERR_TOIL_STUCK = 1 *(TOIL Stuck Signal Failure detected)*

LV_END_DIAG_TOIL_STUCK = 1

else LV_ERR_TOIL_STUCK = 0 *(TOIL Stuck Signal Failure healed)*

LV_END_DIAG_TOIL_STUCK = 1

endif

endif

else LV_END_DIAG_TOIL_STUCK = 1

endif

Emergency operation:

The Limp Home is activated if LV_ERR_TOIL_STUCK = 1 and Diagnosis is finished (LV_END_DIAG_TOIL_STUCK = 1)

- TOIL = TOIL_MDL_FSD (from Fuel System Diagnosis)
- CAM control and short trip deviations diagnosis (P0010) disabled
- TOIL plausability diagnosis disabled

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TOIL_MDL_DIF_MIN_DIAG	6	0...C8H	0...200	1	°C
LDPM_TCO_ST__TOIL_DIF_DIAG	6	0...FEH	-48...142.5	0.75	°C
Minimum oil temperature model increase to end the stuck signal diagnosis					
IP_TOIL_MES_DIF_MIN_DIAG	6	0...C8H	0...200	1	°C
LDPM_TCO_ST__TCO_DIF_DIAG	6	0...FEH	-48...142.5	0.75	°C
Minimum measured oil temperature increase to not detect a stuck signal					
C_TCO_ST_MAX_TOIL_STUCK_DIAG	1	0...FEH	-48...142.5	0.75	°C
Maximum coolant temperature at start for oil temperature stuck signal diagnosis					

A.83.3 TOIL plausability diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TOIL_PLAUS_L	V/S/O	0...1H	0...1	1	-
Boolean for error currently present on oil temperature signal (signal unplausible Low).					
LV_CDN_DIAG_TOIL_PLAUS_L	O/V	0...01H	0...1	1	-
Diagnosis condition TOIL_PLAUS_L diagnosis					
ERR_SYM_TOIL_PLAUS_L	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom TOIL_PLAUS_L diagnosis					
LV_END_DIAG_TOIL_PLAUS_L	V/O	0...1H	0...1	1	-
Diagnosis for TOIL_PLAUS_L diagnosis finished					
LV_ERR_TOIL_PLAUS_H	V/S/O	0...1H	0...1	1	-
Boolean for error currently present on oil temperature signal (signal unplausible High).					
LV_CDN_DIAG_TOIL_PLAUS_H	O/V	0...01H	0...1	1	-
Diagnosis condition TOIL_PLAUS_H diagnosis					
ERR_SYM_TOIL_PLAUS_H	O/V	0H 1H 2H 4H	NO_SYM SYM_0 SYM_1 SYM_2	1	-
Error symptom TOIL_PLAUS_H diagnosis					
LV_END_DIAG_TOIL_PLAUS_H	V/O	0...1H	0...1	1	-
Diagnosis for TOIL_PLAUS_H diagnosis finished					

Input data:

LV_IGK	LV_CDN_VB_MIN_DIAG	LV_ES	LV_ST
LV_ERR_TCO	LV_ERR_TCO_PLAUS	TOIL	TOIL_MES
LV_ERR_TH	LV_ERR_TCO_STUCK	LV_ERR_TOIL_STUCK	LV_ERR_TOIL
TCO_ST	TCO	TOIL_MDL_FSD	CONF_TOIL_MDL

FUNCTION DESCRIPTION:

General information:

To be sure that some diagnosis functions are not disabled by unplausible TOIL value (i.e. Camshaft control deviation is disabled in case of high or low TOIL), this diagnosis function check if :

- TOIL_MES is not unplausibly low when TOIL_MDL_FSD is high
- TOIL_MES is not unplausibly high when TCO is low (and no relevant failure)

Error-symtoms are defined to this diagnosis function as following :

```
| | | └─> Sensor signal unplausible low   (= SYM_0)
| | └───> -                               (= SYM_1)
| └─────> -                               (= SYM_2)
└───────> -                               (= SYM_3)
```

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> -                                  (= SYM_0)
 │  │  └────> Sensor signal unplausible high     (= SYM_1)
 │  └───────> -                                  (= SYM_2)
 └──────────> -                                  (= SYM_3)
```

Application conditions:

Recurrence: **1 sec**

Activation: LV_IGK = 1
and LV_CDN_VB_MIN_DIAG = 1
and LV_ES = 0
and LV_ST = 0
and LV_ERR_TCO = 0
and LV_ERR_TCO_PLAUS = 0
and LV_ERR_TCO_STUCK = 0
and LV_ERR_TH = 0
and LV_ERR_TOIL = 0
and LV_ERR_TOIL_STUCK = 0
and CONF_TOIL_MDL = 0

Deactivation: One of above condition is not true

Formula section:

- TOIL sensor signal unplausible low :

if TCO_ST **<** C_TCO_ST_MIN_TOIL_PLAUS_L
and TOIL_MDL_FSD **>** C_TOIL_MDL_PLAUS_DIAG_L
and TOIL_MES **<** C_TOIL_MIN_DIAG_CAM_DE_IVVT
(Minimum TOIL to activate Camshaft control deviation diagnosis)
then ERR_SYM_TOIL_PLAUS_L = SYM_0 (Unplausible low TOIL sensor signal is detected)
LV_ERR_TOIL_PLAUS_L = 1 (after debounce)
else ERR_SYM_TOIL_PLAUS_L = NO_SYM
LV_ERR_TOIL_PLAUS_L = 0 (after rebounce)
endif

- TOIL sensor signal unplausible high :

if TCO **<** C_TCO_MIN_TOIL_PLAUS_H

and TOIL_MES **>** C_TOIL_MAX_DIAG_CAM_DE_IVVT

(Maximum TOIL to activate Camshaft control deviation diagnosis)

then ERR_SYM_TOIL_PLAUS_H = SYM_1 (Unplausible high TOIL sensor signal is detected)

LV_ERR_TOIL_PLAUS_H = 1 (after debounce)

else ERR_SYM_TOIL_PLAUS_H = NO_SYM

LV_ERR_TOIL_PLAUS_H = 0 (after rebounce)

endif

Emergency operation:

The Limp Home is activated if LV_ERR_TOIL_PLAUS_L/(_H) = 1 .

- TOIL = TOIL_MDL_FSD (from Fuel System Diagnosis)
- CAM control and short trip deviations diagnosis (P0010) disabled
- TOIL stuck diagnosis disabled

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_TCO_ST_MIN_TOIL_PLAUS_L	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature at Start to check TOIL sensor unplausible Low signal					
C_TOIL_MDL_PLAUS_DIAG_L	1	0...C8H	-40...160	1	°C
Oil temperature model threshold to check TOIL sensor unplausible Low signal					
C_TCO_MIN_TOIL_PLAUS_H	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature to check TOIL sensor unplausible High signal					
C_ABC_INC_TOIL_PLAUS_L	1	0...FFH	0...255	1	-
Anti-bounce counter increment for oil temperature sensor signal plausability (Low)					
C_ABC_MAX_TOIL_PLAUS_L	1	0...FFH	0...255	1	-
Maximum value of the anti-bounce counter for oil temperature signal plausability (Low)					
C_ABC_INC_TOIL_PLAUS_H	1	0...FFH	0...255	1	-
Anti-bounce counter increment for oil temperature sensor signal plausability (High)					
C_ABC_MAX_TOIL_PLAUS_H	1	0...FFH	0...255	1	-
Maximum value of the anti-bounce counter for oil temperature signal plausability (High)					
C_TOIL_MIN_DIAG_CAM_DE_IVVT	1	0...C8H	-40...160	1	°C
Min. oil temperature for steady deviation diagnosis					
C_TOIL_MAX_DIAG_CAM_DE_IVVT	1	0...C8H	-40...160	1	°C
Max. oil temperature for steady deviation diagnosis					

A.83.4 Oil temperature sensor diagnosis (Appl. Inc.)

A.83.4.1 Oil temperature signal range diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Engine Oil Temperature Sensor Circuit Malfunction	SCG	SYM_0	*STD*
	SCB + OL	SYM_1	
		SYM_2	
TOIL		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
TIA_MES
TOIL_MDL
V_TOIL

Application Assistances:

- Calculation of the IVVT Intake Camshaft Drift Speed disabled

A.83.4.2 Oil temperature stuck signal diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Engine Oil Temperature Circuit Range/ Performance		SYM_0	*STD*
		SYM_1	
	Signal stuck	SYM_2	
TOIL_STUCK		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
TOIL_MES
TOIL_MES_DIF_DIAG
TOIL_MDL_DIF_DIAG

A.83.4.3 Oil temperature signal plausibility diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Engine Oil Temperature Sensor Circuit Malfunction		SYM_0	*STD*
	Unplausible high signal	SYM_1	
		SYM_2	
TOIL_PLAUS_H		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
TOIL_MES
TOIL_MES_DIF_DIAG
TOIL_MDL_DIF_DIAG

Engine Oil Temperature Sensor Circuit Malfunction	Unplausible low signal	SYM_0	*STD*
		SYM_1	
		SYM_2	
TOIL_PLAUS_L		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
TOIL_MES
TOIL_MES_DIF_DIAG
TOIL_MDL_DIF_DIAG

Application Assistances:

A.83.4.4 RBM interface for the "TOIL Signal Stuck" diagnosis

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
STATE_RBM_TOIL_STUCK	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor of TOIL Signal Stuck diagnosis for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TOIL_STUCK	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of TCO Signal Stuck diagnosis, for application purposes					

Input data:

LV_END_DIAG_TOIL_STUCK	CTR_COMP_RBM_TOIL_STUCK	CTR_CDN_RBM_TOIL_STUCK	LV_DC
LV_ERR_TCO_EL	LV_ERR_TCO_PLAUS	LV_ERR_TCO_GRD	
CTR_ERR_DYN_NR	LV_ERR_TOIL	LV_ERR_TCO_STUCK	
LV_ERR_TOIL_PLAUS_L	LV_ERR_TOIL_PLAUS_H		

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TOIL_STUCK monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TOIL_STUCK data.

Within STATE_RBM_TOIL_STUCK, three different informations are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
  (general information depending on monitor could be added)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
  (general information depending on monitor could be added)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (general information depending on monitor could be added)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_TOIL_STUCK = 0

at LV_DC 0 → 1 transition :

bit 0, bit 1 and bit 2 of STATE_RBM_TOIL_STUCK = 0

on failure memory reset :

bit 1 of STATE_RBM_TOIL_STUCK = 0

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_TCO_EL, LV_ERR_TCO_PLAUS, LV_ERR_TCO_GRD, LV_ERR_TCO_STUCK, LV_ERR_TOIL, LV_ERR_TOIL_PLAUS_L, LV_ERR_TOIL_PLAUS_H

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TOIL_STUCK = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TOIL_STUCK = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TOIL_STUCK = 0
Then
If LV_END_DIAG_TOIL_STUCK = 1
Then
bit 0 of STATE_RBM_TOIL_STUCK = 1
Endif
Endif

If bit 1 of STATE_RBM_TOIL_STUCK = 0
Then
If LV_ERR_TCO_EL = 1 **or**
LV_ERR_TCO_GRD = 1 **or**
LV_ERR_TCO_PLAUS = 1 **or**
LV_ERR_TCO_STUCK = 1 **or**
LV_ERR_TOIL = 1 **or**
LV_ERR_TOIL_PLAUS_L = 1 **or**
LV_ERR_TOIL_PLAUS_H = 1 **or**
Then
bit 1 of STATE_RBM_TOIL_STUCK = 1
Endif
Endif

No specific condition for TOIL_STUCK :

bit 2 of STATE_RBM_TOIL_STUCK always equal to 1

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TOIL_STUCK = 0
then RATE_RBM_TOIL_STUCK = FFFFH
else RATE_RBM_TOIL_STUCK
= (CTR_COMP_RBM_TOIL_STUCK / CTR_CDN_RBM_TOIL_STUCK)
endif

A.84 Differential Tank Pressure Sensor Diagnosis (DTP)

A.84.1 General activation condition

Input data:

C_CONF_DIAGCP			

Application conditions:

Activation: C_CONF_DIAGCP ≠ 0

Deactivation: C_CONF_DIAGCP = 0

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the differential tank pressure sensor output signal. Additional to a short circuit or a line break it could occur that the chip inside the pressure sensor fails and the signal which is then provided at the sensor output could be in the valid range, but it is not fluctuating anymore. To detect this failure the plausibility of the signal is checked by looking at the maximum fluctuation.

The DTP diagnosis is performed by three errors**.**

Electrical error:

"DTP Short to Ground" :**MIN_DTP (ABC)**

"DTP Short to Plus or line break" :**MAX_DTP (ABC)**

Plausibility error:

"DTP signal not plausible" :**PLAUS_DTP (STC)**

Noisy error:

"DTP signal Noisy":**NOISE_DTP (STC)**

Freeze frame, DTC, error code number, symptom number, recurrence, CARB-Info, Successive error management and data for MIL-management see chapters for general diagnosis informations.

A.84.2 DTP electrical diagnosis and SOV stuck diagnosis:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_DTP_EL	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (Electrical Diagnosis).					
LV_CDN_DIAG_DTP_EL	V/O	0...1H	0...1	1	-
Diagnosis condition for electrical diagnosis (DTP_MIN and DTP_MAX)					
ERR_SYM_DTP_EL	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom DTP_EL diagnosis					
LV_END_DIAG_DTP_EL	V/O	0...1H	0...1	1	-
End of electrical diagnosis (DTP_MIN and DTP_MAX) diagnosis					

Input data:

CONF_DIAGCP	LV_CDN_VB_OBD2	LV_IGK	V_DTP
DTP			LV_ERR_VCC_SENS_SUB

Description:

Error-symtoms are defined to this diagnosis function as following :

Short circuit to Vbatt or open line (= SYM_0)
Short circuit to GND (= SYM_1)

Remark: Calculation of LV_END_DIAG_DTP_EL see generic calculation “End of diagnosis” in anti bounce algorithm.

Application conditions:

Initialisation: at new DC

- LV_ERR_DTP_EL = 0
- LV_ERR_MEC_SOV = 0

Recurrence:

- 40ms,

Activation:

If LV_IGK = 1
and LV_CDN_VB_OBD2 = 1
and CONF_DIAGCP > 0
and LV_ERR_VCC_SENS_SUB = 0
and LV_ERR_MEC_SOV = 0

then the diagnosis for the symptoms ´**MIN_DTP**´ and ´**MAX_DTP**´ is **available**

else the diagnosis for the symptoms ´**MIN_DTP**´ and ´**MAX_DTP**´ is **not available**

Symptom 'Signal Line Short to Battery voltage' or 'Signal line break' ***MAX_DTP****:*

If V_DTP > C_V_DTP_MAX_DIAG

then 'Signal line short to battery voltage' or 'Signal line break' is detected

and Symptom **MAX_DTP** is active, anti bounce counter is incremented

and ERR_SYM_DTP_EL = SYM_0

and LV_ERR_DTP_EL = 1 (after debounce)

Symptom 'Signal line short to ground' ***MIN_DTP****:*

elseif V_DTP < C_V_DTP_MIN_DIAG

then 'Signal line short to ground' is detected

and symptom **MIN_DTP** is active, anti bounce counter is incremented

and ERR_SYM_DTP_EL = SYM_1

and LV_ERR_DTP_EL = 1 (after debounce)

else

then symptom **MAX_DTP** and **MIN_DTP** are inactive

and ERR_SYM_DTP_EL = NO_SYM

and LV_ERR_DTP_EL = 0 (after rebounce)

endif

Remark : For the calibration data for SOV stuck, refer to 'Charcoal-canister shutoff valve (SOV) stuck' chapter

- If no symptom is active, anti bounce counter is decremented

Anti-bounce counter increment: C_ABC_INC_DTP_EL
Maximum value of anti-bounce counter: C_ABC_MAX_DTP_EL

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_DTP_EL	1	0...FFH	0...255	1	-
Anti-bounce counter increment for DTP signal diagnosis.					
C_ABC_MAX_DTP_EL	1	0...FFH	0...255	1	-
Maximum value of the anti-bounce counter for DTP signal diagnosis.					
C_V_DTP_MIN_DIAG	1	0...3FFH	0...4.995	0.00488	V
Minimum threshold for DTP signal diagnosis.					
C_V_DTP_MAX_DIAG	1	0...3FFH	0...4.995	0.00488	V
Maximum threshold for DTP signal diagnosis.					

A.84.3 Plausibility Diagnosis /signal stuck / stuck High or Low

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
V_DTP_MAX_DIAG	V/O	0...3FFH	0...4.995	0.00488	V
Maximum voltage of the DTP for Plausibility Diagnosis					
V_DTP_MIN_DIAG	V/O	0...3FFH	0...4.995	0.00488	V
Minimum voltage of the DTP for Plausibility Diagnosis					
LV_ERR_DTP_PLAUS	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (Plausibility Diagnosis).					
LV_DTP_PLAUS_READY	V/O	0...1H	0...1	1	-
Boolean for finished DTP Plausibility Diagnosis					
LV_CDN_DIAG_DTP_PLAUS	V/O	0...1H	0...1	1	-
Diagnosis condition for plausibility diagnosis (DTP_PLAUS)					
ERR_SYM_DTP_PLAUS	V/O	0H 1H	NO_SYM SYM_0	1	-
Detected error symptom DTP_PLAUS diagnosis					
LV_END_DIAG_DTP_PLAUS	V/O	0...1H	0...1	1	-
End of DTP plausibility diagnosis (DTP_PLAUS) diagnosis					
LV_ERR_DTP_PLAUS_H	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (DTP_PLAUS_H)					
LV_DTP_PLAUS_H_READY	V/O	0...1H	0...1	1	-
Boolean for finished DTP Plausibility Diagnosis (DTP_PLAUS_H)					
LV_CDN_DIAG_DTP_PLAUS_H	V/O	0...1H	0...1	1	-
Diagnosis condition for plausibility diagnosis (DTP_PLAUS_H)					
ERR_SYM_DTP_PLAUS_H	V/O	0H 1H	NO_SYM SYM_1	1	-
Detected error symptom diagnosis (DTP_PLAUS_H)					
LV_END_DIAG_DTP_PLAUS_H	V/O	0...1H	0...1	1	-
End of DTP plausibility diagnosis (DTP_PLAUS_H)					
LV_ERR_DTP_PLAUS_L	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (DTP_PLAUS_L)					
LV_DTP_PLAUS_L_READY	V/O	0...1H	0...1	1	-
Boolean for finished DTP Plausibility Diagnosis (DTP_PLAUS_L)					
LV_CDN_DIAG_DTP_PLAUS_L	V/O	0...1H	0...1	1	-
Diagnosis condition for plausibility diagnosis (DTP_PLAUS_L)					
ERR_SYM_DTP_PLAUS_L	V/O	0H 1H	NO_SYM SYM_2	1	-
Detected error symptom diagnosis (DTP_PLAUS_L)					
LV_END_DIAG_DTP_PLAUS_L	V/O	0...1H	0...1	1	-
End of DTP plausibility diagnosis (DTP_PLAUS_L)					
V_DTP_DIF_PLAUS_MOD_6	V/O/S	0...3FFH	0...4.995	0.00488	V
Saved value of DTP varaiation for mode $06					
DTP_PLAUS_H_MOD_6	V/O/S	8000...7FFFH	-40.96...40.96	0.00125	hPa
Saved value of DTP for PLAUS_H monitor in mode $06					
DTP_PLAUS_L_MOD_6	V/O/S	8000...7FFFH	-40.96...40.96	0.00125	hPa
Saved value of DTP for PLAUS_L monitor in mode $06					

Input data:

V_DTP	CP_STATE	CL_MMV	MAF_CPS
LV_ST_END	C_ABC_MAX_DTP_EL	LV_ERR_DTP_EL	TAM
C_TAM_MIN_DIAGCP	LV_ERR_CPS	LV_ERR_MEC_CPS	LV_ERR_SOV
LV_ERR_MEC_SOV	C_V_DTP_MIN_DIAG	C_DTP_MIN_SOV	C_DTP_MIN_DIAG
C_DTP_MAX_DIAG	DTP	TCO	C_TCO_MIN_DIAGCP
LV_ERR_VCC_SENS_SUB	LV_ERR_FTL	FTL_MMV	LV_DTP_PLAUS_READY
LV_DTP_PLAUS_L_READY	LV_DTP_PLAUS_H_READY	CONF_DIAGCP	C_V_DTP_MAX_DIAG

LV_CDN_VB_OBD1	LV_ERR_TAM	LV_END_DIAG_VAP_LEAK_10	LV_END_DIAG_FUC_MISS

Description:

Error-symtoms are defined to this diagnosis function as following :

DTP signal not plausible (= SYM_0)
DTP signal not plausible high (= SYM_1)
DTP signal not plausible low (= SYM_2)

Remark: Calculation of LV_END_DIAG_DTP_PLAUS(_H/_L) see generic calculation "End of diagnosis"

Application conditions:

Initialisation: at new DC

- LV_ERR_DTP_PLAUS = 0
- LV_ERR_DTP_PLAUS_H = 0
- LV_ERR_DTP_PLAUS_L = 0
- LV_DTP_PLAUS_READY = 0 (DTP Plausibility runs every DC)
- LV_DTP_PLAUS_H_READY = 0
- LV_DTP_PLAUS_L_READY = 0

Recurrence: 1 sec

After engine start both the minimum and the maximum value of V_DTP are monitored. If all conditions for the Plausibility Diagnosis are fulfilled, the difference between the minimum and the maximum value of V_DTP is compared with a threshold.

- Determination of signal fluctuation:

1. Initialization: (at ECU reset) V_DTP_MAX_DIAG = 0V (0 hex)
V_DTP_MIN_DIAG = 4.995V (3FF hex)

2.Initialization: (one second after ECU reset) V_DTP_MAX_DIAG = V_DTP
V_DTP_MIN_DIAG = V_DTP

All following calculations must only be done after 2. Initialisation of V_DTP_MIN/MAX_DIAG.

- MIN / MAX determination of V_DTP:

If V_DTP_n > V_DTP_MAX_DIAG
then V_DTP_MAX_DIAG = V_DTP_n

If V_DTP_n < V_DTP_MIN_DIAG
then V_DTP_MIN_DIAG = V_DTP_n

Activation 1 : for DTP stuck

If LV_DTP_PLAUS_READY = 0

then **If** LV_ST_END = 1

and CONF_DIAGCP > 0

and LV_CDN_VB_OBD1 = 1

and LV_ERR_DTP_EL = 0

and LV_ERR_VCC_SENS_SUB = 0 (no Error on DTP Sensor supply voltage)

and LV_ERR_CPS = 0

and LV_ERR_SOV = 0

and LV_ERR_MEC_CPS = 0

and LV_ERR_MEC_SOV = 0

and C_ABC_MAX_DTP_EL < 255

and TAM >= C_TAM_MIN_DIAGCP

and FTL_MMV < C_FTL_MAX_DTP_PLAUS_H

and LV_END_DIAG_DTP_PLAUS = 0

and LV_ERR_FTL = 0

and LV_ERR_TAM = 0

then the diagnosis for the symptom **DTP_PLAUS** is **available**
LV_CDN_DIAG_DTP_PLAUS = 1

else the diagnosis for the symptom **DTP_PLAUS** is **not available**
LV_CDN_DIAG_DTP_PLAUS = 0

endif

else the diagnosis for the symptom **DTP_PLAUS** is **not available**
LV_CDN_DIAG_DTP_PLAUS = 0

endif

Activation 2 : for DTP stuck high & low

if LV_ST_END = 1

and CONF_DIAGCP > 0

and LV_CDN_VB_OBD1 = 1

and LV_ERR_DTP_EL = 0

and LV_ERR_VCC_SENS_SUB = 0 (no Error on DTP Sensor supply voltage)

and LV_ERR_CPS = 0

and LV_ERR_SOV = 0

and LV_ERR_MEC_CPS = 0

and LV_ERR_MEC_SOV = 0

and TAM >= C_TAM_MIN_DIAGCP

and TCO >= C_TCO_MIN_DIAGCP

and FTL_MMV < C_FTL_MAX_DTP_PLAUS_H

and LV_ERR_FTL = 0

and LV_ERR_TAM = 0

then **If** LV_DTP_PLAUS_H_READY = 0

then the diagnosis for the symptom **DTP_PLAUS** is **available**
LV_CDN_DIAG_DTP_PLAUS_H = 1

else the diagnosis for the symptom **DTP_PLAUS** is **not available**
LV_CDN_DIAG_DTP_PLAUS_H = 0

endif

If LV_DTP_PLAUS_L_READY = 0
then the diagnosis for the symptom **DTP_PLAUS** is **available**
LV_CDN_DIAG_DTP_PLAUS_L = 1

else the diagnosis for the symptom **DTP_PLAUS** is **not available**
LV_CDN_DIAG_DTP_PLAUS_L = 0

endif

else the diagnosis for the symptom **DTP_PLAUS** is **not available**
LV_CDN_DIAG_DTP_PLAUS_H = 0

LV_CDN_DIAG_DTP_PLAUS_L = 0

endif

__Formula section:__

1) Symptom 'Plausibility Check Failure DTP' **PLAUS_DTP** :
The symptom **PLAUS_DTP** is entered directly into failure memory

If (V_DTP_MAX_DIAG - V_DTP_MIN_DIAG) >= C_V_DTP_THD_DIAG

then the symptom **PLAUS_DTP** is **not active**

ERR_SYM_DTP_PLAUS = NO_SYM

LV_ERR_DTP_PLAUS = 0

LV_DTP_PLAUS_READY = 1

LV_END_DIAG_DTP_PLAUS = 1

else **if** (V_DTP_MAX_DIAG - V_DTP_MIN_DIAG) < C_V_DTP_THD_DIAG

and C_V_DTP_MIN_DIAG <= V_DTP <= C_V_DTP_MAX_DIAG

and VS >= C_VS_MIN_DTP_PLAUS_DIAG __at least once__ since beginning of driving cycle

and V_DTP > C_V_DTP_PLAUS_MIN_DIAG

and { LV_END_DIAG_VAP_LEAK_10 = 1

or LV_END_DIAG_FUC_MISS = 1 }

then the symptom **PLAUS_DTP** is **active**

ERR_SYM_DTP_PLAUS = SYM_0
LV_ERR_DTP_PLAUS = 1 (immediately)
LV_DTP_PLAUS_READY = 1
LV_END_DIAG_DTP_PLAUS = 1

endif

endif

2) Symptom 'Plausibility high check Failure DTP' **PLAUS_DTP_H** :
The symptom **PLAUS_DTP_H** is entered directly into failure memory

if DTP < C_DTP_MAX_DIAG

then the symptom **PLAUS_DTP_H** is **not active**

ERR_SYM_DTP_PLAUS_H = NO_SYM

LV_ERR_DTP_PLAUS_H = 0

LV_DTP_PLAUS_H_READY = 1

LV_END_DIAG_DTP_PLAUS_H = 1

else **if** DTP >= C_DTP_MAX_DIAG

and C_V_DTP_MIN_DIAG <= V_DTP <= C_V_DTP_MAX_DIAG

and **{** (CP_STATE = MAX_PURGE
and CL_MMV < C_CL_MMV_MAX_DTP_DIAG
and MAF_CPS > C_MAF_CPS_MIN_DTP_DIAG)
for C_T_MIN_DTP_PLAUS_H uninterrupted seconds}
(T_MIN_DTP_PLAUS_H starts from 0 if one of above conditions in "{}" is not true)

then the symptom **PLAUS_DTP_H** is **active**
ERR_SYM_DTP_PLAUS_H = SYM_1
LV_ERR_DTP_PLAUS_H = 1 (immediately)
LV_DTP_PLAUS_H_READY = 1
LV_END_DIAG_DTP_PLAUS_H = 1

endif

endif

3) Symptom 'Plausibility high check Failure DTP' **PLAUS_DTP_L** :
The symptom **PLAUS_DTP_L** is entered directly into failure memory

if DTP > C_DTP_MIN_DIAG

then the symptom **PLAUS_DTP_L** is **not active**

ERR_SYM_DTP_PLAUS_L = NO_SYM

LV_ERR_DTP_PLAUS_L = 0

LV_DTP_PLAUS_L_READY = 1

LV_END_DIAG_DTP_PLAUS_L = 1

else **if** C_DTP_MIN_SOV <= DTP <= C_DTP_MIN_DIAG

and **{** CP_STATE = CP_NOT_ACTIVE **or** NO_PURGE
for C_T_MIN_DTP_PLAUS_L uninterrupted seconds **}**
(T_MIN_DTP_PLAUS_L starts from 0 if one of above conditions in "{}" is not true)

then the symptom **PLAUS_DTP_L** is **active**
ERR_SYM_DTP_PLAUS_L = SYM_2
LV_ERR_DTP_PLAUS_L = 1 (immediately)
LV_DTP_PLAUS_L_READY = 1
LV_END_DIAG_DTP_PLAUS_L = 1

endif

endif

A.84.3.1 Scantool SAE 1979 Mode $06

Description:

The following calculations of output values for Scan Tool SAE 1979 Mode $06 take place only at the end of a complete diagnostic cycle.

Formula section:

If LV_DTP_PLAUS_READY transition 0 -> 1

Then V_DTP_DIF_PLAUS_MOD_6 = V_DTP_MAX_DIAG – V_DTP_MIN_DIAG

Endif

If LV_DTP_PLAUS_L_READY transition 0 -> 1

Then DTP_PLAUS_L_MOD_6 = DTP

Endif

If LV_DTP_PLAUS_H_READY transition 0 -> 1

Then DTP_PLAUS_H_MOD_6 = DTP

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_DTP_THD_DIAG	1	0...3FFH	0...4.995	0.00488	V
Pressure fluctuation threshold for Plausibility Diagnosis					
C_VS_MIN_DTP_PLAUS_DIAG	1	0 ...FFH	0 .. 255	1	Km/h
Min. value for vehicle speed for DTP Plauscheck					
C_V_DTP_PLAUS_MIN_DIAG	1	0...3FFH	0...4.995	0.00488	V
Minimum voltage of DTP to enable Plausibility Diagnosis					
C_CL_MMV_MAX_DTP_DIAG	1	0...FFFFH	0...8	0.000122	-
Max. CL_MMV for DTP Plausicheck					
C_MAF_CPS_MIN_DTP_DIAG	1	0....FFFFH	0....8	0.000122	kg/h
minimum MAF_CPS for DTP Plausicheck					
C_T_MIN_DTP_PLAUS_H	1	0 ...FFH	0 .. 255	1	s
Time condition for DTP high Plausicheck					
C_T_MIN_DTP_PLAUS_L	1	0 ...FFH	0 .. 255	1	s
Time condition for DTP low Plausicheck					
C_FTL_MAX_DTP_PLAUS_H	1	0...FFH	0...100	0.39	%
FTL_MMV maximal condition for DTP stuck high diagnosis					

A.84.4 Plausibility Diagnosis /signal noisy

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_DTP_NOISE	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (Signal Noisy); irreversible.					
LV_CDN_DIAG_DTP_NOISE	V/O	0...1H	0...1	1	-
Diagnosis condition for DTP_NOISE diagnosis					
ERR_SYM_DTP_NOISE	V/O	0H 1H	NO_SYM SYM_0	1	-
Detected error symptom DTP_NOISE diagnosis					
LV_END_DIAG_DTP_NOISE	V/O	0...1H	0...1	1	-
End of DTP_NOISE diagnosis					
CTR_ABC_NOISE_DTP	V/O	0...FFH	0...255	1	-
anti bounce counter of diagnosis DTP_NOISE					

Input data:

LV_DTP_NOISE	LV_DC	STATE_DIAGCP	EOL_STATE_DIAGCP
LV_ERR_VCC_SENS_SUB			

Description:

Error-symtoms are defined to this diagnosis function as following :

DTP signal noisy (= SYM_0)

Remark: Calculation of LV_END_DIAG_DTP_NOISE see generic calculation “End of diagnosis”

Application conditions:

Initialisation: at new DC

- LV_ERR_DTP_NOISE = 0
- all 0 for DTP_NOISE diagnosis

Recurrence: 40ms

Activation:

If LV_DC = 1

and LV_ERR_DTP_NOISE = 0

and LV_ERR_VCC_SENS_SUB = 0

and **{** STATE_DIAGCP = 1H (PRS_DYN state)

or STATE_DIAGCP = 2H (CLOSE_CPS state)

or transition : STATE_DIAGCP = 1H -> 11H (T_DLY_3) **}**

then the diagnosis for the symptom NOISE_DTP is available
LV_CDN_DIAG_DTP_NOISE = 1

else the diagnosis for the symptom NOISE_DTP is not available
LV_CDN_DIAG_DTP_NOISE = 0

endif

Formula section:

Symptom **NOISE_DTP**; noisy DTP signal

If LV_DTP_NOISE **changes** 0 --> 1 (DTP signal noisy)
and LV_CDN_DIAG_DTP_NOISE = 1
then symptom **NOISE_DTP** is **active**
ERR_SYM_DTP_NOISE = SYM_0
One increment anti-bounce counter with C_ABC_INC_NOISE_DTP per diagnosis
if anti-bounce counter reaches C_ABC_MAX_NOISE_DTP
or EOL_STATE_DIAGCP <> EOL_PAS
then LV_ERR_DTP_NOISE = 1 (irreversible)
LV_END_DIAG_DTP_NOISE = 1
endif
elseif LV_DTP_NOISE = 0
and LV_CDN_DIAG_DTP_NOISE **changes** 1 --> 0
and STATE_DIAGCP = 3H (DLY_SOV state)
then symptom **NOISE_DTP** is **not available**
ERR_SYM_DTP_NOISE = NO_SYM

if CTR_ABC_NOISE_DTP > 0
then One decrement anti-bounce counter with 1
else LV_ERR_DTP_NOISE = 0
LV_END_DIAG_DTP_NOISE = 1
endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_NOISE_DTP	1	0 ...FFH	0 .. 255	1	-
Maximum value for frequency counter for DTP Noise diagnosis					
C_ABC_MAX_NOISE_DTP	1	0 ...FFH	0 .. 255	1	-
Maximum value for test cycle counter for DTP Noise diagnosis					

A.84.5 Global error flag

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_DTP	V/O	0...1H	0...1	1	-
Boolean for error currently present on differential tank pressure sensor (EL or PLAUS).					

Input data:

LV_ERR_DTP_NOISE	LV_ERR_DTP_PLAUS	LV_ERR_DTP_EL	LV_ERR_VCC_SENS_SUB

Application conditions:

Initialisation: at new DC

- LV_ERR_DTP = 0

Formula section:

If LV_ERR_DTP_PLAUS = 1 **or**
LV_ERR_DTP_PLAUS_H = 1 **or**
LV_ERR_DTP_PLAUS_L = 1 **or**
LV_ERR_DTP_NOISE = 1 **or**
LV_ERR_DTP_EL = 1 **or**
LV_ERR_VCC_SENS_SUB = 1 (Error on DTP Sensor supply voltage)
then LV_ERR_DTP = 1
else LV_ERR_DTP = 0
endif

A.84.6 Electrical diagnosis

Configuration or calibration data:

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_DTP_EL	SCB + OL	SYM_0	*STD*
	SCG	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
CP_STATE
CPPWM_ENVD
V_DTP_ENVD

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limp home for the concerned function and inhibed the concerned diagnosis).

List of impacted function and diagnosis :

Are inhibited:
- DTP sensor plausibility diagnosis
- Evaporative system monitoring
- SOV stuck closed diagnosis

A.84.6.1 List of environmental data to be stored in failure memory :

(For more details to refer 'Environmental data' specification)

A.84.6.1.1 List of specific environmental data (FRF_SPECIFIC_CUS) :

-

A.84.7 Constant Sensor detection

Configuration or calibration data:

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_DTP_PLAUS	Signal not plausible	SYM_0	*STC*
	Signal not plausible high	SYM_1	
	Signal not plausible low	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
DTP_DIF_ENVD_H
DTP_DIF_ENVD_L
V_DTP_ENVD

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limp home for the concerned function and inhibed the concerned diagnosis).

List of impacted function and diagnosis :

Are inhibited:
- DTP sensor plausibility diagnosis
- Evaporative system monitoring
- SOV stuck closed diagnosis

A.84.7.1 List of environmental data to be stored in failure memory :

(For more details to refer ‘Environmental data’ specification)

A.84.7.1.1 List of specific environmental data (FRF_SPECIFIC_CUS) :

-

A.84.7.2 Noisy sensor detection

Configuration or calibration data:

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_DTP_NOISE	Signal noisy	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: DTP_DIF_ENVD_H
DTP_DIF_ENVD_L
CPPWM_ENVD
V_DTP_ENVD

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limp home for the concerned function and inhibed the concerned diagnosis).

List of impacted function and diagnosis :

Are inhibited:
- DTP sensor plausibility diagnosis
- Evaporative system monitoring
- SOV stuck closed diagnosis

A.84.7.3 List of environmental data to be stored in failure memory :

(For more details to refer ‘Environmental data’ specification)

A.84.7.3.1 List of specific environmental data (FRF_SPECIFIC_CUS) :

-

A.84.8 SOV stuck open diagnosis

Configuration or calibration data:

Diagnostic	Symptom Description	Symptom	Filter type
LV_ERR_MEC_SOV	Stuck in closed position	SYM_0	*STD*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory. TIA_ST
TAM
CPPWM_ENVD
V_DTP_ENVD

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limp home for the concerned function and inhibed the concerned diagnosis).

List of impacted function and diagnosis :

Disabled
- Evaporative system monitoring

A.84.8.1 List of environmental data to be stored in failure memory :

(For more details to refer 'Environmental data' specification)

A.84.8.1.1 List of specific environmental data (FRF_SPECIFIC_CUS) :

-

A.84.9 Lambda control activation monitoring diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Insufficient Coolant Temperature for closed-loop Fuel Control		SYM_0	STD
		SYM_1	
		SYM_2	
TCO_PLAUS	Plausibility fault	SYM_3	

List of Environmental Data to store in Failure Memory: T_AST_ENVD
TIA_MES
TCO_MES
TIA_ST

*** Application Assistances:**

- TIA/TCO - Limp Home Calculation modified
- Evaporative Emission Control - Activation of MIN_PURGE instead of MAX_PURGE
- Fans Management set to MAX
- The following Control Functions are inhibited:
  - Engine Overheating Protection at VS = 0
  - Progressive Fuel Cut Off and Rewetting at Part Load
  - Downstream Fuel Trim Regulation
- The following Adaptation Functions are inhibited:
  - Lambda Adaptation
  - Knock Adaptation
  - Idle Speed Adaptation
- The following Diagnosis Functions are inhibited:
  - Differential Tank Pressure sensor Plausibility Diagnosis (Noisy signal)
  - Differential Tank Pressure sensor Plausibility Diagnosis (Signal constant)
  - Evaporative System Diagnosis
  - Fuel System Diagnosis
  - Catalyst Efficiency Diagnosis
  - MAP/TPS Plausibility Diagnosis
  - O2-Sensor Upstream Switching Time Diagnosis
  - O2-Sensor Upstream Frequency Diagnosis
  - O2-Sensor Downstream Switching Time Diagnosis
  - O2-Sensor Downstream PUC end Diagnosis
  - O2-Sensor Downstream SCG Diagnosis
  - Thermostat Diagnosis
  - TCO Stuck Diagnosis

A.85 Diagnosis of the cruise control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CRU_SWI_1	V/O	0...1H	0...1	1	-
logical variable cruise switch button “out of range” acquisition error					
LV_ERR_CRU_SWI_2	V/O	0...1H	0...1	1	-
logical variable cruise switch SET/COAST stuck button acquisition error					
LV_ERR_CRU_SWI_3	V/O	0...1H	0...1	1	-
logical variable cruise switch RES/ACC stuck button acquisition error					
ERR_SYM_CRU_SWI_1	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for cruise switch button “out of range” acquisition error					
LV_CDN_DIAG_CRU_SWI_1	V/O	0...1H	0...1	1	-
Diagnisis condition for cruise switch button “out of range” acquisition error					
LV_END_DIAG_CRU_SWI_1	V/O	0...1H	0...1	1	-
End of cruise switch button “out of range” acquisition diagnosis					
ERR_SYM_CRU_SWI_2	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for cruise switch SET/COAST stuck button acquisition error					
LV_CDN_DIAG_CRU_SWI_2	V/O	0...1H	0...1	1	-
Diagnisis condition for cruise switch SET/COAST stuck button acquisition error					
LV_END_DIAG_CRU_SWI_2	V/O	0...1H	0...1	1	-
End of cruise switch SET/COAST stuck button acquisition diagnosis					
ERR_SYM_CRU_SWI_3	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Error symptom for cruise switch RES/ACC stuck button acquisition error					
LV_CDN_DIAG_CRU_SWI_3	V/O	0...1H	0...1	1	-
Diagnisis condition for cruise switch RES/ACC stuck button acquisition error					
LV_END_DIAG_CRU_SWI_3	V/O	0...1H	0...1	1	-
End of cruise switch RES/ACC stuck button acquisition diagnosis					
LV_CRU_SWI_SET_PAS	V	0...1H	0...1	1	-
Bit indicating that the SET button is pressed out of the STATE_CRU = “PASSIVE”					
LV_CRU_SWI_RES_PAS	V	0...1H	0...1	1	-
Bit indicating that the RES button is pressed out of the STATE_CRU = “PASSIVE”					

Input data:

LV_ST_END	CRU_SWI_DRIV_STATE	LV_IGK	CRU_BAS
STATE_CRU	CONF_CRU		

FUNCTION DESCRIPTION:

General Information:

The purpose of the diagnosis is to detect faults in the cruise switches. The signal of the cruise switches are checked for stuck position of the button for the SET/COAST and RES/ACC switches. Invalid voltage ranges and short circuit to positve, are also checked on the LAV since the switches are directly wired to the ECU.

Description:

The bits carrying the error-symptom is described in "Table of failure chapter" as well as the ABC counter type.

Application conditions:

Initialisation: by ERRM: all 0 according filter type

Recurrence: 100ms

Activation: (LV_IGK = on (key on) **and** LV_ST_END = 1 (out of start phase)
and LC_ERR_CRU_SWI_AUTH = 1 **and** CONF_CRU = 1)

Deactivation: LV_IGK = off (key off)
at deactivation: LV_CDN_DIAG_CRU_SWI_1/2/3 = 0

Formula section:

If **((**CRU_BAS > C_V_CRU_BAS_HI over C_T_CRU_BAS_HI_DIAG time delay**) or**
(CRU_SWI_DRIV_STATE = "OUT OF RANGE" over C_T_CRU_OUT_DIAG time delay**))**

Then Invalid voltage range
LV_CDN_DIAG_CRU_SWI_1 = 1
ERR_SYM_CRU_SWI_1 = 1H
LV_ERR_CRU_SWI_1 = 1 (invalid voltage range; after debounce)

Else **If** CRU_BAS < C_V_CRU_BAS_HI and CRU_SWI_DRIV_STATE ≠ "OUT OF RANGE"
Then No failure
LV_CDN_DIAG_CRU_SWI_1 = 1
ERR_SYM_CRU_SWI_1 = 0H
LV_ERR_CRU_SWI_1 = 0 (after rebounce)

Endif

When the "SET/COAST" buttton is pressed out of the STATE_CRU = "PASSIVE" for more than C_T_CRU_SET_COAST_DIAG, an error is detected.

If CRU_SWI_DRIV_STATE = "SET/COAST" and STATE_CRU (n-1) = "PASSIVE"
Then LV_CRU_SWI_SET_PAS = 1
Else **If** CRU_SWI_DRIV_STATE ≠ "SET/COAST"
Then LV_CRU_SWI_SET_PAS = 0
Endif
Endif

If **(**CRU_SWI_DRIV_STATE = “SET/COAST” over C_T_CRU_SET_COAST_DIAG time delay
and LV_CRU_SWI_SET_PAS = 1**)**
Then <u>SET/COAST switch stuck</u>
LV_CDN_DIAG_CRU_SWI_2 = 1
ERR_SYM_CRU_SWI_2 = 2H
LV_ERR_CRU_SWI_2 = 1 for remainder of ignition cycle
(set/coast switch stuck; after debounce)
Else **if** **(**CRU_SWI_DRIV_STATE ≠ “SET/COAST” over C_T_CRU_SET_OFF_DIAG time delay**)**
then <u>No failure</u>
LV_CDN_DIAG_CRU_SWI_2 = 1
ERR_SYM_CRU_SWI_2 = 0H
LV_ERR_CRU_SWI_2 = 0 (after rebounce)
endif
Endif

When the “RES/ACC” buttton is pressed out of the STATE_CRU = “PASSIVE” for more than C_T_CRU_RES_ACC_DIAG, an error is detected.

If CRU_SWI_DRIV_STATE = “RES/ACC” and STATE_CRU (n-1) = “PASSIVE”
Then LV_CRU_SWI_RES_PAS = 1
Else **If** CRU_SWI_DRIV_STATE ≠ “RES/ACC”
Then LV_CRU_SWI_RES_PAS = 0
Endif
Endif

If **(**CRU_SWI_DRIV_STATE = “RES/ACC” over C_T_CRU_RES_ACC_DIAG time delay
and LV_CRU_SWI_RES_PAS = 1**)**
Then <u>RES/ACC switch stuck</u>
LV_CDN_DIAG_CRU_SWI_3 = 1
ERR_SYM_CRU_SWI_3 = 4H
LV_ERR_CRU_SWI_3 = 1 for remainder of ignition cycle
(res/acc switch stuck; after debounce)
Else **if** **(**CRU_SWI_DRIV_STATE ≠ “RES/ACC”
over C_T_CRU_RES_OFF_DIAG time delay**)**
then <u>No failure</u>
LV_CDN_DIAG_CRU_SWI_3 = 1
ERR_SYM_CRU_SWI_3 = 0H
LV_ERR_CRU_SWI_3 = 0 (after rebounce)
endif
Endif

<u>Filtering :</u>

Apply filter on current symptoms
If filtering result available *{ after debounce }*
LV_ERR_CRU_SWI_1/2/3 = filtering result
LV_END_DIAG_CRU_SWI_1/2/3 = 1
Delivery the result to Error Management

Endif

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Cruise switch button "out of range" diagnosis	Invalid voltage range	SYM_0	*STD_INI*
		SYM_1	
		SYM_2	
CRU_SWI_1		SYM_3	
Cruise switch SET/COAST stuck button diagnosis		SYM_0	*STD_INI*
	SET/COAST switch stuck	SYM_1	
		SYM_2	
CRU_SWI_2		SYM_3	
Cruise switch RES/ACC stuck button diagnosis		SYM_0	*STD_INI*
		SYM_1	
	RES/ACC switch stuck	SYM_2	
CRU_SWI_3		SYM_3	

List of Environmental Data to store in Failure Memory: CRU_BAS_ENVD_H
CRU_BAS_ENVD_L
CRU_SWI_DRIV_STATE
STATE_CRU

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_T_CRU_BAS_HI_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to detect a short to battery, 5V; typical 1 second					
C_V_CRU_BAS_HI	1	0...3FFH	0...4.99512	.00488	V
Voltage level to detect a short to battery or 5V					
C_T_CRU_SET_COAST_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to detect a stuck SET/COAST switch; typical 90 seconds					
C_T_CRU_RES_ACC_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to detect a stuck RES/ACC switch; typical 90 seconds					
C_T_CRU_OUT_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to detect “out of range”; typical 1 second					
C_T_CRU_SET_OFF_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to reset failure once SET button is released					
C_T_CRU_RES_OFF_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Time delay to reset failure once RESUME button is released					
LC_ERR_CRU_SWI_AUTH	1	0...1H	0...1	1	-
Flag to activate the diagnosis on the cruise switch					
C_ABC_INC_CRU_SWI_1	1	0...FFH	0...255	1	-
Anti bounce increment for cruise failure detection					
C_ABC_MAX_CRU_SWI_1	1	0...FFH	1...255	1	-
Anti bounce max value for cruise failure detection					
C_ABC_INC_CRU_SWI_2	1	0...FFH	0...255	1	-
Anti bounce increment for cruise failure detection					
C_ABC_MAX_CRU_SWI_2	1	0...FFH	1...255	1	-
Anti bounce max value for cruise failure detection					
C_ABC_INC_CRU_SWI_3	1	0...FFH	0...255	1	-
Anti bounce increment for cruise failure detection					
C_ABC_MAX_CRU_SWI_3	1	0...FFH	1...255	1	-
Anti bounce max value for cruise failure detection					

A.86 Clutch switch diagnosis (LV_ERR_CS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_CS	V/O	0...1H	0...1	1	[-]
Diagnosis condition CS					
ERR_SYM_CS	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	0	[-]
Detected CS Symptom					
LV_ERR_CS	V/O	0...1H	0...1	1	[-]
Active CS error					
LV_END_DIAG_CS	V/O	0...1H	0...1	1	[-]
End of CS diagnosis					
LV_CS_DIAG	V	0...1H	0...1	1	[-]
Flag set when vehicle was stopped = pre-condition for diagnosis cycle					
LV_CS_SWI	V	0...1H	0...1	1	[-]
Flag set when CS change its state					
T_MIN_MAF_CS_DIAG	V	0...FFH	0...255	1	[s]
Minimum diagnosis time for MAF condition					

Input data:

LV_IGK	LV_IM_CS_PN	N_32	LV_AT
MAF	VS	C_VS_MIN_RUN	LV_ERR_MWSS
LV_ERR_VS	LV_ST	LV_ES	CONF_CRU

FUNCTION DESCRIPTION:

General information:

This module performs the diagnosis of the clutch switch. This is valid only in case of manual transmission and if vehicle is equiped with cruise-control.

The diagnosis is based on the verification of the presence of the expected clutch action to shift the gears necessary to bring the vehicle to a certain speed. It is highly improbable that the vehicle is accelerated several consecutive times to a certain speed with the clutch always depressed or never depressed.

Description:

If there's a a change in LV_IM_CS_PN, the CS is diagnosed as error-free and the diagnosis is switched off if end of diagnosis is reached.

If there's no change in LV_IM_CS_PN then each time the vehicle is stopped, the diagnosis is enabled. Until the vehicle gets to a certain speed, LV_IM_CS_PN is monitored for value change (clutch pedal actuation). If, for several times (Anti bounce is inrementing by one inrement per diagnosis cycle) the vehicle gets to speed and the clutch switch state has never changed (always 0 or 1) then a failure is recognized.

Depending of status LV_IM_CS_PN 0/1 at beginning of monitoring period there is the symtom OL/SCB or SCG.

Application conditions:

Initialisation: according filter type: ***"STD-INI",***
all other outputs 0 at LV_IGK 0->1 **or** reset **or** at clearing error memory

Recurrence: 1s (10 ms for LV_IM_CS_PN detection)

Activation:

If	LV_IGK = 0	**or**	
	LV_ES = 1	**or**	
	LV_ST = 1	**or**	
	LV_AT = 1	**or**	
	CONF_CRU = 0	**or**	
	LV_END_DIAG_CS = 1	**or**	
	LV_ERR_VS = 1	**or**	*// VS error*
	LV_ERR_MWSS = 1		*// MWSS error*
Then	Function passive LV_CDN_DIAG_CS is reset to 0		
Else	Function active		
Endif			

Formula section:

LV_CDN_DIAG_CS = 0

Errorfree case: CS activitation detected:

If LV_IM_CS_PN switched once from 1 to 0 **or** 0 to 1 (managed each 10 ms)
Then LV_CDN_DIAG_CS = 1
ERR_SYM_CS = NO_SYM
LV_CS_SWI = 1
LV_END_DIAG_CS = 1 *Calculated by error management*

Else

Error case: High vehicle speed without CS activation

If VS ≤ C_VS_MIN_RUN *Init when vehicle is stopped*
Then LV_CS_DIAG = 1
Else **If** VS > C_VS_CS_DIAG **and**
MAF > C_MAF_CS_DIAG **and** *above zero no load line*
N_32 > C_N_32_CS_DIAG *out of idle speed*
Then increment T_MIN_MAF_CS_DIAG
If T_MIN_MAF_CS_DIAG >= C_T_MIN_MAF_CS_DIAG **and**
LV_CS_SWI = 0 **and** *the clutch did not switched*
LV_CS_DIAG = 1 *vehicle was stopped*
Then LV_CDN_DIAG_CS = 1
If LV_IM_CS_PN = 1

Then ERR_SYM_CS = SYM_0 */short circuit to battery or open circuit*

If LV_IM_CS_PN = 0

Then ERR_SYM_CS = SYM_1 */short circuit to ground*

Endif

LV_CS_DIAG = 0

// vehicle must stop again before next diagnosis

$CTR\_ABC\_CS_n = CTR\_ABC\_CS_{n-1} + C\_ABC\_INC\_CS$ *only once*

Endif

Else T_MIN_MAF_CS_DIAG = 0 *reset*

Endif

End if

Endif

Calculation of LV_END_DIAG_CS and LV_ERR_CS

LV_ERR_CS Calculated by error management : CTR_ABC_CS reach maximum

LV_END_DIAG_CS Calculated by error management

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_CS_DIAG	1	0...FFH	0...255	1	[km/h]
Speed threshold for activation of diagnosis cycle					
C_T_MIN_MAF_CS_DIAG	1	0...FFH	0...255	1	[s]
Minimum diagnosis time for MAF condition					
C_MAF_CS_DIAG	1	0...FFFFH	0...1389	2.12E-02	[mg/stk]
MAF threshold for error detection					
C_N_32_CS_DIAG	1	0...FFH	0...8160	32	[rpm]
N32 threshold for error detection					
C_ABC_INC_CS	1	0...FFH	0...255	1	[-]
Anti bouce increment CS diagnosis					
C_ABC_MAX_CS	1	1...FFH	1...255	1	[-]
Anti bouce maximum CS diagnosis					

A.87 Clutch switch diagnosis (appl. inc.)

Configuration for diagnostic symptoms:

Diagnosis	Symptom description	Symptom	Filter-Type
Clutch switch diagnosis	Short circuit to Vbatt or open load	SYM_0	STD_INI
	Short circuit to ground	SYM_1	
		SYM_2	
CS		SYM_3	

List of Environmental Data to store in failure memory: TAM

VS

LV_BLS

LV_BTS

A.88 RCL actuator electrical power stage diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_EL_RCL_ACR	O/V	0...1H	0...1	1	-
Present RCL electrical failure after filtering of diagnosis					
ERR_SYM_EL_RCL_ACR	O/V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of RCL electrical detected failure (SYM_0 = SCP, SYM_1 = SCG, SYM_2 = OC)					
CDN_DIAG_EL_RCL_ACR	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of EL_RCL_ACR					
CTR_INH_DIAG_EL_RCL_ACR	-	0...FFH	0...255	1	-
Delay counter for RCL electrical diagnosis					
ERR_DIAG_EL_RCL_ACR	-	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Raw value of error symptom for EL_RCL_ACR (only parameter)					

Input data:

LC_TCHA_CONF	LV_INH_DIAG_EL_RCL_ACR	LV_CDN_VB_OBD1	LV_IGK
NC_IDX_DIAG_EL_RCL_ACR			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_EL_RCL_ACR	1	0...FFH	0...255	1	-
Anti - bounce counter increment for RCL actuator electrical failure diagnosis					
C_ABC_MAX_EL_RCL_ACR	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce for RCL actuator electrical failure diagnosis					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PSD_DLY_EL_RCL_ACR	1	1...FH	1...15	1	-
Number of recurrences with diagnosis deactivation after diagnosis activation conditions are met (= 2)					

Import actions:

ACTION_ERRM_FilterMulticondition(IN <XX>, IN <cdn_diag_XX>, IN <err_diag_XX>, IN <c_abc_inc_XX>, IN <c_abc_max_XX>, OUT <lv_err_XX>)
This action computes the elementary anti-bounce filter for one failure treatment and returns filter result
ACTION_ERRM_GetErrSym(IN <XX>, OUT <err_sym_XX>)
This action returns the error-symptom
ACTION_INFR_GetElDiagEl_rcl_acr(OUT <CDN_DIAG_EL_RCL_ACR>, OUT <ERR_DIAG_EL_RCL_ACR>)
This action reads the failure and the conditions information for each symptom from the infrastructure

A.88.1 General information

The RCL actuator is driven by the ECU via an output driver. The failure detection is done by ECU hardware. The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

After activation conditions are met, the diagnosis activation is delayed for NC_PSD_DLY_EL_RCL_ACR executions, to avoid the usage of wrong infrastructure information.

Error-symptoms and conditions are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> SYM_0: Short circuit to battery (SCP)
 │ │ └───> SYM_1: Short circuit to ground (SCG)
 │ └─────> SYM_2: Open circuit (OC)
 └───────> SYM_3
```

Configuration for diagnostic symptoms:

Diagnostic **RCL Actuator**	**Symptoms**	**Nb**	**Filter type**
RCL actuator diagnosis	SCP	*0*	*MPL_STD_INI*
	SCG	*1*	
	OC	2	
		3	

The data LV_END_DIAG_EL_RCL_ACR and LV_CDN_DIAG_EL_RCL_ACR are not present in output data, since they are not used by other functions. Nevertheless; they are provided in error management and always visible.

The error management variables are initialized according to filter type.

Application Condition

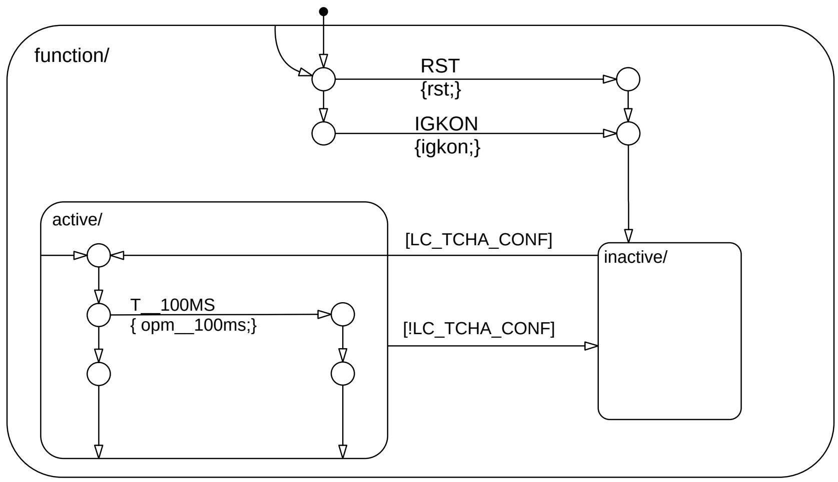


Figure 59 CHRG_RCLACRELDIAG0/ APP_CDN/ Chart

Function Description

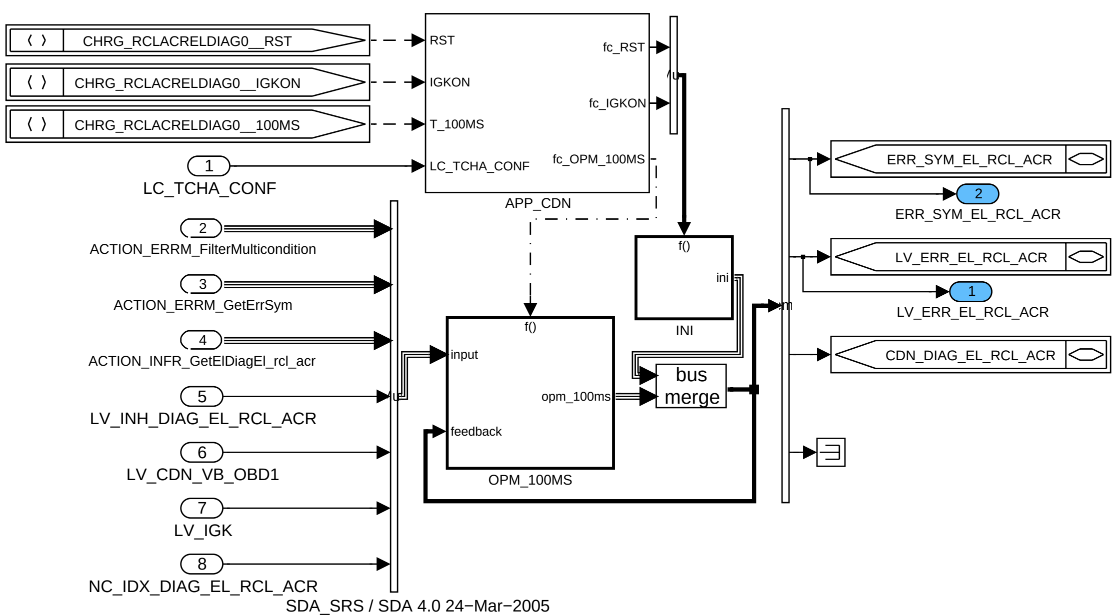


Figure 60 CHRG_RCLACRELDIAG0

A.88.1.1 Initialization

At reset and ignition-key-on

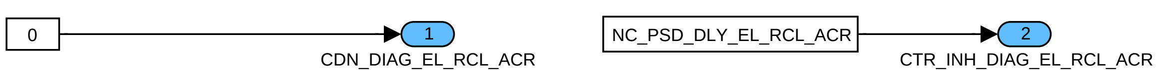


Figure 61 CHRG_RCLACRELDIAG0/ INI/ INI

A.88.1.2 Formula section

Validity of diagnosis and call of error management

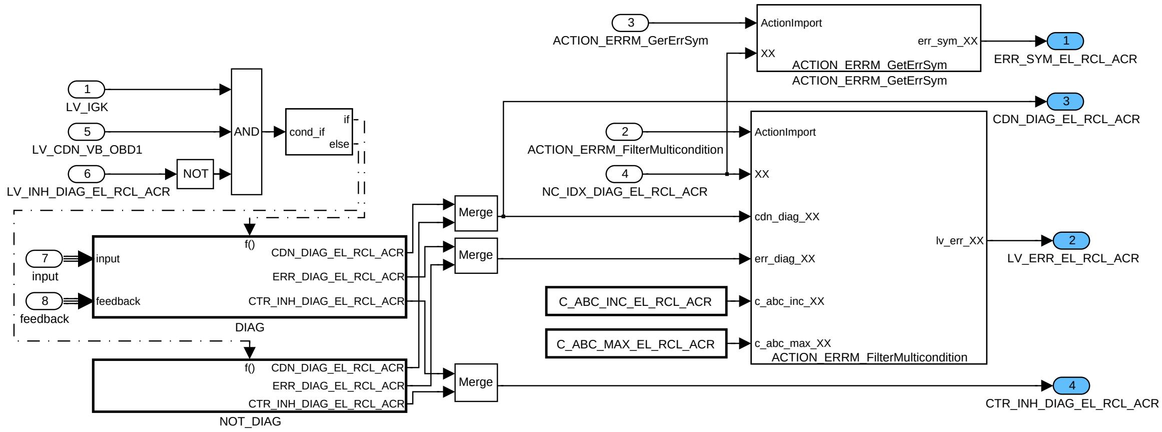


Figure 62 CHRG_RCLACRELDIAG0/ OPM_100MS/ DIAGNOSIS

Valid diagnosis

The diagnosis information (failure symptoms – raw value – ERR_DIAG_EL_RCL_ACR and basic diagnosis conditions CDN_DIAG_EL_RCL_ACR) is received from the infrastructure by action-call.

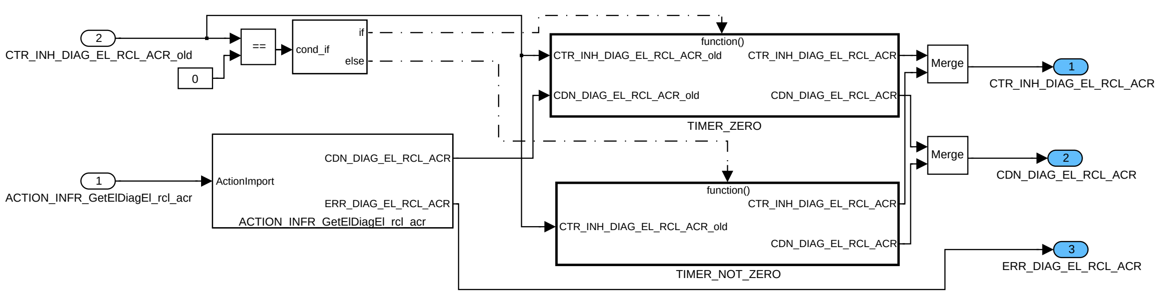


Figure 63 CHRG_RCLACRELDIAG0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG

Timer elapsed

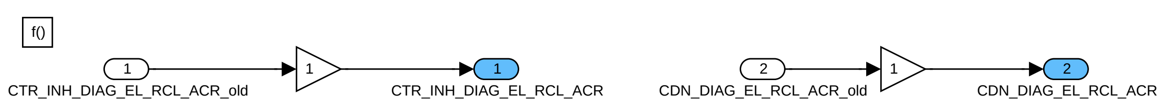


Figure 64 CHRG_RCLACRELDIAG0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG/ TIMER_ZERO

Timer not yet elapsed

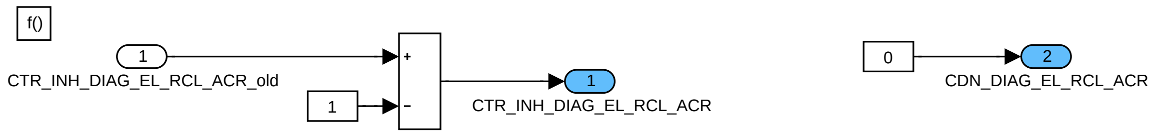


Figure 65 CHRG_RCLACRELDIAG0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG/ TIMER_NOT_ZERO

Invalid diagnosis

CONTENT

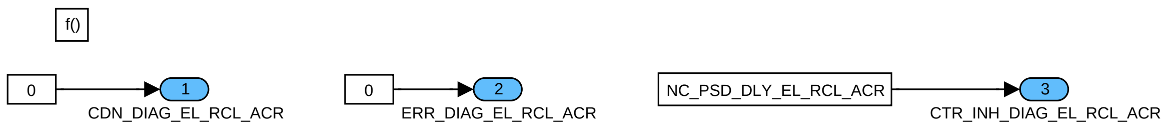


Figure 66 CHRG_RCLACRELDIAG0/ OPM_100MS/ DIAGNOSIS/ NOT_DIAG

A.89 RCL actuator diagnosises (Application Incidences)

A.89.1 General:

Input data:

LC_TCHA_CONF	LV_DC
LV_ERR_TAM	LV_ERR_AMP
LV_ERR_TPS_MAF_1	LV_ERR_CRK_PLAUS
LV_ERR_AMP_PLAUS	LV_ERR_PVS
LV_ERR_EL_RCL_ACR	LV_ERR_CAN_BUS_OFF
LV_ERR_PVS_H_1	LV_ERR_CAP_H
LV_ERR_TPS_2	LV_ERR_PVS_L_1
LV_ERR_PVS_H_2	LV_ERR_PVS_RATIO
LV_ERR_TPS	LV_ERR_TPS_1
LV_ERR_TPS_MAF_2	LV_ERR_TPS_AD
LV_ERR_TPS_ST_CHK	
LV_ES	LV_ERR_CRK_SYN
LV_ERR_PUT	LV_ERR_WG
LV_ERR_CRK_TOOTH_PER	LV_ERR_VS
LV_ERR_PVS_L_2	LV_LIH_ERR_CRK
LV_ERR_PLAUS_CLOSE_RCL	LV_ERR_TPS_AD_SPR
LV_ERR_MAP_PLAUS	LV_ERR_TIMEOUT_TCU1
	LV_END_DIAG_PLAUS_OPEN_RCL
LV_ERR_PUT_PLAUS	LV_ERR_TPS_AD_GAIN
LV_ERR_PLAUS_OPEN_RCL	LV_END_DIAG_PLAUS_CLOSE_RCL
CTR_ERR_DYN_NR	LV_ERR_CAM
CTR_CDN_RBM_PLAUS_OPEN_RCL	GEAR
CTR_COMP_RBM_PLAUS_CLOSE_RCL	LV_ERR_CRK_TOOTH
CTR_COMP_RBM_PLAUS_OPEN_RCL	CTR_CDN_RBM_PLAUS_CLOSE_RCL
LV_ST	LV_ERR_EL_PUT
LV_ERR_VCC_SENS_SUB	LV_CDN_VB_OBD1

A.89.2 Project specific inhibition conditions:

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_EL_RCL_ACR	O/V	0...1H	0...1	1	[-]
one indicates inhibition of the electrical RCL actuator Diagnosis					
LV_INH_DIAG_PLAUS_CLOSE_RCL	O/V	0...1H	0...1	1	[-]
"1" indicates inhibition of the plausibility RCL -stuck close - actuator Diagnosis					
LV_INH_DIAG_RBM_PLAUS_CLOSE_RCL	V	0...1H	0...1	1	[-]
RCL plausibility - stuck close - diagnosis inhibition condition for RBM interface definition					
LV_INH_DIAG_PLAUS_OPEN_RCL	O/V	0...1H	0...1	1	[-]
"1" indicates inhibition of the plausibility RCL -stuck open - actuator Diagnosis					
LV_INH_DIAG_RBM_PLAUS_OPEN_RCL	V	0...1H	0...1	1	[-]
RCL plausibility - stuck open - diagnosis inhibition condition for RBM interface definition					
STATE_RBM_PLAUS_CLOSE_RCL	O/V	0...FFH	0...255	1	[-]
Interface of RCL plausibility - stuck close - diagnosis monitor with the Rate-Based Monitoring statistics					
RATE_RBM_PLAUS_CLOSE_RCL	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Actual RBM ratio of RCL plausibility - stuck close - diagnosis, for application purposes					
STATE_RBM_PLAUS_OPEN_RCL	O/V	0...FFH	0...255	1	[-]
Interface of RCL plausibility - stuck open - diagnosis monitor with the Rate-Based Monitoring statistics					
RATE_RBM_PLAUS_OPEN_RCL	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Actual RBM ratio of RCL plausibility - stuck open - diagnosis, for application purposes					

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_x_RCL allows to deactivate the corresponding RCL diagnostic.

The Formula section must be updated by the Project. The Calibration data list must be completed.

The project can add additionnal conditions if desired.

FUNCTION DESCRIPTION:

Application conditions:

Initialization: **At** Reset **or** Transition LV_ES = 0 to 1:

LV_INH_DIAG_EL_RCL_ACR = 0
LV_INH_DIAG_PLAUS_CLOSE_RCL = 0
LV_INH_DIAG_PLAUS_OPEN_RCL = 0

Reccurence: **10 ms**

Activation: LC_CHA_CONF = 1

Deactivation: LC_CHA_CONF = 0

Formula section:

EL diag:

If LC_INH_DIAG_EL_RCL_ACR = 1
Or LV_ST = 1
Or LV_CDN_VB_OBD1 = 0
Then LV_INH_DIAG_EL_RCL_ACR = 1
Else LV_INH_DIAG_EL_RCL_ACR = 0
Endif

PLAUS_CLOSE diag:

If LV_ERR_EL_RCL_ACR = 0 *[RCL]*
And LV_ERR_PLAUS_**OPEN**_RCL = 0 *[RCL]*
And LV_ERR_TAM = 0 (global: AIRT) *[TAM]*
And LV_ERR_VS = 0 *(Global failure: Project specific)*
And LV_ERR_AMP = 0 (el.: INSY) *[AMP]*
And LV_ERR_AMP_PLAUS = 0 (plaus. – partial global: INSY) *[AMP]*
And LV_ERR_PUT = 0 (el.: CHRG) *[PUT]*
And LV_ERR_PVS = 0 (global: project specific) *[PV_AV_GRD]*
And LV_ERR_WG = 0 (el.: CHRG) *[WG]*
And LV_ERR_CAP_H = 0 (plaus.: CHRG) *[WG]*
Then LV_INH_DIAG_RBM_PLAUS_CLOSE_RCL = 0
Else LV_INH_DIAG_RBM_PLAUS_CLOSE_RCL = 1
Endif

If LC_INH_DIAG_PLAUS_RCL_AS_ACT = 1
Then LV_INH_DIAG_PLAUS_CLOSE_RCL = LC_INH_DIAG_PLAUS_RCL_AS
Else **If** LV_INH_DIAG_RBM_PLAUS_CLOSE_RCL = 0
    And GEAR <> F**E**
    Then LV_INH_DIAG_PLAUS_CLOSE_RCL = 0
    Else LV_INH_DIAG_PLAUS_CLOSE_RCL = 1
    Endif
Endif

PLAUS_OPEN diag:

If LV_ERR_EL_RCL_ACR = 0 *[RCL]*
And LV_ERR_PLAUS_**CLOSE**_RCL = 0 *[RCL]*
And LV_ERR_TAM = 0 (global: AIRT) *[TAM]*
And LV_ERR_AMP = 0 (el.: INSY) *[AMP]*
And LV_ERR_AMP_PLAUS = 0 (plaus. – partial global: INSY) *[AMP]*
And LV_ERR_PUT = 0 (el.: CHRG) *[PUT]*
And LV_ERR_TPS = 0 (global – project specific) [TPS]
And LV_LIH_ERR_CRK = 0 *(global _ 1st part: ENSD) [N]*
And LV_ERR_CAM = 0 *(global _ 2nd part: ENSD) [N]*
And LV_ERR_WG = 0 (el.: CHRG) *[WG]*
And LV_ERR_CAP_H = 0 (plaus.: CHRG) *[WG]*
Then LV_INH_DIAG_RBM_PLAUS_OPEN_RCL = 0
Else LV_INH_DIAG_RBM_PLAUS_OPEN_RCL = 1
Endif

If LC_INH_DIAG_PLAUS_RCL_AS_ACT = 1
Then LV_INH_DIAG_PLAUS_OPEN_RCL = LC_INH_DIAG_PLAUS_RCL_AS
Else LV_INH_DIAG_PLAUS_ OPEN_RCL = LV_INH_DIAG_RBM_PLAUS_OPEN_RCL

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_PLAUS_CLOSE_RCL	1	0...FFH	0...255	1	[-]
Anti - bounce counter increment for RCL actuator plausibility failure -stuck close- diagnosis					
C_ABC_MAX_PLAUS_CLOSE_RCL	1	1...FFH	1...255	1	[-]
Maximum value of the anti - bounce for RCL actuator plausibility failure -stuck close- diagnosis					
C_ABC_INC_PLAUS_OPEN_RCL	1	0...FFH	0...255	1	[-]
Anti - bounce counter increment for RCL actuator plausibility failure -stuck open- diagnosis					
C_ABC_MAX_PLAUS_OPEN_RCL	1	1...FFH	1...255	1	[-]
Maximum value of the anti - bounce for RCL actuator plausibility failure -stuck open- diagnosis					
LC_INH_DIAG_EL_RCL_ACR	1	0...1H	0...1	1	[-]
If set to 1 then the RCL actuator Electrical diagnosis is inhibited					
LC_INH_DIAG_PLAUS_RCL_AS	1	0...1H	0...1	1	[-]
LV_INH_DIAG_PLAUS_CLOSE[OPEN]_RCL value set via application system					
LC_INH_DIAG_PLAUS_RCL_AS_ACT	1	0...1H	0...1	1	[-]
LC_INH_DIAG_PLAUS_RCL_AS_ACT=1: LV_INH_DIAG_PLAUS_CLOSE[OPEN]_RCL is set via appli. system					

Configuration for diagnostic symptoms:

Electrical diagnosis:

Diagnostic RCL	Symptoms	Nb	Filter type
RCL actuator Electrical diagnosis	SYM_0 = SCP	*0*	*MPL_STD_INI*
	SYM_1 = SCG	*1*	
	SYM_2 = OC	2	

Diagnostic RCL	Symptoms	Nb	P-Code/ Symptom	P-Code/ Global 1	P-Code/ Global 2	Failure class A/B	Readiness/ Code
RCL actuator Electrical diagnosis	signal to high	*0*					
	signal to low	*1*					
	no signal	*2*					
		3					

Plausibility – stuck close - diagnosis:

Diagnostic RCL	Symptoms	Nb	Filter type
RCL actuator Plausibility – stuck close - diagnosis		*0*	*MEM*
		1	
		2	
		3	

Diagnostic RCL	Symptoms	Nb	P-Code/ Symptom	P-Code/ Global 1	P-Code/ Global 2	Failure class A/B	Readiness/ Code
RCL actuator Plausibility – stuck close - diagnosis	Plausibility - stuck close	*0*					
		1					
		2					
		3					

Plausibility – stuck open - diagnosis:

Diagnostic RCL	Symptoms	Nb	Filter type
RCL actuator Plausibility – stuck open - diagnosis		*0*	*MEM*
		1	
		2	
		3	

Diagnostic RCL	Symptoms	Nb	P-Code/ Symptom	P-Code/ Global 1	P-Code/ Global 2	Failure class A/B	Readiness/ Code
RCL actuator Plausibility – stuck open - diagnosis	Plausibility _ stuck open	*0*					
		1					
		2					
		3					

- The antibounce type may be changed by the project.
- The recommended P-code value may be the P-code defined by the Carb. The project may changed this parameter if the customer or the project decides to take into account others values.
- The failure class parameter allows to configure the behaviour of the error management for this failure. This value may be recommended and the customer or the project decides to take into account others values.
- The failure class are described in the specification Failure Class.

Application assistance:

In case of the function is declared damaged by the diagnosis then the impacted function or diagnosis must be taken into account (applied a correction or a limphome for the concerned function and inhibed the concerned diagnosis).

List of impacted functions and diagnoses :

A.89.3 RBM interface for RCL Plausibilty - stucked close - diagnosis:

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the RCL plausibility - stuck close - monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_PLAUS_CLOSE_RCL data.

Within STATE_RBM_PLAUS_CLOSE_RCL, three different informations are defined:

_ Conditions for monitoring are met long enough to detect malfunction (bit 0)
(no intrusive operation, no short trip)
_ Monitor disabled because of system malfunction (bit 1)
(depending on failure status: pending)
_ Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_CLOSE_RCL = 0
at LV_DC 0 → 1 transition :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_CLOSE_RCL = 0
on failure memory reset :
bit 1 of STATE_RBM_PLAUS_CLOSE_RCL = 0

Recurrence: **1 s**

Activation: LC_TCHA_CONF =1 **And** (LV_DC 0 → 1 transition **Or** LV_DC = 1)

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

{ Specific, the project shall list hereafter all failures that can inhibit the RCL plausibility - stuck close - monitor }
{ Global failures which aren't managed by Error Management shall not appear here, because their pending status doesn't exist }

LV_ERR_EL_RCL_ACR *[RCL]*
LV_ERR_PLAUS_**OPEN**_RCL *[RCL]*
LV_ERR_CAN_BUS_OFF
LV_ERR_TIMEOUT_TCU1
LV_ERR_VS (*Project specific – Global failure – pending status available) [VS]*
LV_ERR_AMP (el.: INSY) *[AMP]*

LV_ERR_AMP_PLAUS (plaus. – check: INSY) *[AMP]*
LV_ERR_MAP_PLAUS (plaus. – check: INSY) *[MAP]*
LV_ERR_EL_PUT (el.: CHRG) *[PUT]*
LV_ERR_VCC_SENS_SUB *(supply voltage for PUT sensor)*
LV_ERR_PUT_PLAUS (plaus. – check: INSY) *[PUT]*
LV_ERR_PVS_L_1 (project specific) *[PV_AV_GRD]*
LV_ERR_PVS_L_2 (project specific) *[PV_AV_GRD]*
LV_ERR_PVS_H_1 (project specific) *[PV_AV_GRD]*
LV_ERR_PVS_H_2 (project specific) *[PV_AV_GRD]*
LV_ERR_PVS_RATIO (project specific) *[PV_AV_GRD]*
LV_ERR_WG (el.: CHRG) *[WG]*
LV_ERR_CAP_H (plaus.: CHRG) *[WG]*

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }
Then(1) **While** bit 1 of STATE_RBM_PLAUS_CLOSE_RCL = 0 **do**
with each RCL plausibility - stuck close - failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>,
OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
If(2) **XX** has a pending status
Then(2) bit 1 of STATE_RBM_PLAUS_CLOSE_RCL = 1
Endif(2)
Endwhile
Else(1) { the dynamic failure memory is empty }
No action
Endif(1)

Every 1 s :

- ***STATE_RBM determination:***

If bit 0 of STATE_RBM_PLAUS_CLOSE_RCL = 0
Then **If** LV_END_DIAG_PLAUS_CLOSE_RCL = 1
Then bit 0 of STATE_RBM_PLAUS_CLOSE_RCL = 1
Endif
Endif

If bit 1 of STATE_RBM_PLAUS_CLOSE_RCL = 0
Then **If** LV_INH_DIAG_RBM_PLAUS_CLOSE_RCL = 1
{ Cross dependency of the monitor defined by the project }
{ List here failures that can inhibit RCL plausibility - stuck close - monitor }
Then bit 1 of STATE_RBM_PLAUS_CLOSE_RCL = 1
Endif
Endif

bit 2 of STATE_RBM_PLAUS_CLOSE_RCL = 1

- ***RATE_RBM determination (to easier the calibration):***

RATE_RBM_PLAUS_CLOSE_RCL
= CTR_COMP_RBM_PLAUS_CLOSE_RCL / CTR_CDN_RBM_PLAUS_CLOSE_RCL

A.89.4 RBM interface for RCL Plausibilty - stucked open - diagnosis:

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the RCL plausibility - stuck open - monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_PLAUS_OPEN_RCL data.

Within STATE_RBM_PLAUS_OPEN_RCL, three different informations are defined:

_ Conditions for monitoring are met long enough to detect malfunction (bit 0)
(no intrusive operation, no short trip)
_ Monitor disabled because of system malfunction (bit 1)
(depending on failure status: pending)
_ Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_OPEN_RCL = 0
at LV_DC 0 → 1 transition :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_OPEN_RCL = 0
on failure memory reset :
bit 1 of STATE_RBM_PLAUS_OPEN_RCL = 0

Recurrence: **1 s**

Activation: LC_TCHA_CONF =1 **And** (LV_DC 0 → 1 transition **Or** LV_DC = 1)

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

{ Specific, the project shall list hereafter all failures that can inhibit the RCL plausibility - stuck open - monitor }
{ Global failures which aren't managed by Error Management shall not appear here, because their pending status doesn't exist }

LV_ERR_EL_RCL_ACR *[RCL]*
LV_ERR_PLAUS_**CLOSE**_RCL *[RCL]*
LV_ERR_CAN_BUS_OFF
LV_ERR_TIMEOUT_TCU1
LV_ERR_AMP = 0 (el.: INSY) *[AMP]*
LV_ERR_AMP_PLAUS (plaus. – check: INSY) *[AMP]*
LV_ERR_MAP_PLAUS (plaus. – check : INSY) *[MAP]*
LV_ERR_EL_PUT (el.: CHRG) *[PUT]*
LV_ERR_VCC_SENS_SUB *(supply voltage for PUT sensor)*
LV_ERR_PUT_PLAUS (plaus. – check: INSY) *[PUT]*

LV_ERR_TPS_1 (Project specific) *[TPS]*
LV_ERR_TPS_2 (Project specific) *[TPS]*
LV_ERR_TPS_MAF_1 (Project specific) *[TPS]*
LV_ERR_TPS_MAF_2 (Project specific) *[TPS]*
LV_ERR_TPS_AD (Project specific) *[TPS]*
LV_ERR_TPS_AD_SPR (Project specific) *[TPS]*
LV_ERR_TPS_AD_GAIN (Project specific) *[TPS]*
LV_ERR_TPS_ST_CHK (Project specific) *[TPS]*
LV_ERR_CRK_TOOTH (ENSD) *[N]*
LV_ERR_CRK_TOOTH_PER (ENSD) *[N]*
LV_ERR_CRK_SYN (ENSD) *[N]*
LV_ERR_CRK_PLAUS (ENSD) *[N]*
LV_ERR_WG (el.: CHRG) *[WG]*
LV_ERR_CAP_H (plaus.: CHRG) *[WG]*

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }
Then(1) **While** bit 1 of STATE_RBM_PLAUS_OPEN_RCL = 0 **do**
with each RCL plausibility - stuck open - failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
If(2) **XX** has a pending status
Then(2) bit 1 of STATE_RBM_PLAUS_OPEN_RCL = 1
Endif(2)
Endwhile
Else(1) { the dynamic failure memory is empty }
No action
Endif(1)

Every 1 s :

- ***STATE_RBM determination:***

If bit 0 of STATE_RBM_PLAUS_OPEN_RCL = 0
Then **If** LV_END_DIAG_PLAUS_OPEN_RCL = 1
Then bit 0 of STATE_RBM_PLAUS_OPEN_RCL = 1
Endif
Endif

If bit 1 of STATE_RBM_PLAUS_OPEN_RCL = 0
Then **If** LV_INH_DIAG_RBM_PLAUS_OPEN_RCL = 1
{ Cross dependency of the monitor defined by the project }
{ List here failures that can inhibit RCL plausibility - stuck open - monitor }
Then bit 1 of STATE_RBM_PLAUS_OPEN_RCL = 1
Endif
Endif

bit 2 of STATE_RBM_PLAUS_OPEN_RCL = 1

- ***RATE_RBM determination (to easier the calibration):***

RATE_RBM_PLAUS_OPEN_RCL
= CTR_COMP_RBM_PLAUS_OPEN_RCL / CTR_CDN_RBM_PLAUS_OPEN_RCL

A.90 Electrical diagnosis of Waste Gate power stage

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_WG	O/V	0...1H	0...1	1	-
Present failure after filtering of diagnosis WG					
CDN_DIAG_WG	V	0...7H	0...7	1	-
Diagnosis condition for each symptom of WG					
CTR_INH_DIAG_WG	-	0...FFH	0...255	1	-
Delay counter for WG electrical diagnosis					
ERR_DIAG_WG	-	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Raw value of error symptom for WG (only parameter)					

Input data:

LC_TCHA_CONF	PWM_WG	LV_INH_DIAG_WG	LV_CDN_VB_OBD1
LV_IGK	NC_IDX_DIAG_WG		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_WG	1	0...FFH	0...255	1	-
Antibounce counter increment					
C_ABC_MAX_WG	1	1...FFH	1...255	1	-
Maximum value for antibounce counter					
C_WG_EL_PWM_DIAG_MAX_SCG	1	0...FFH	0...99.609375	0.390625	%
Maximum threshold for SCG diagnosis window					
C_WG_EL_PWM_DIAG_MIN_OC	1	0...FFH	0...99.609375	0.390625	%
Minimum threshold for OC diagnosis window					
C_WG_EL_PWM_DIAG_MIN_SCP	1	0...FFH	0...99.609375	0.390625	%
Minimum threshold for SCP diagnosis window					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PSD_DLY_WG	1	1...FH	1...15	1	-
Number of recurrences with diagnosis deactivation after diagnosis activation conditions are met (= 2)					

Import actions:

ACTION_ERRM_FilterMulticondition(IN <XX>, IN <cdn_diag_XX>, IN <err_diag_XX>, IN <c_abc_inc_XX>, IN <c_abc_max_XX>, OUT <lv_err_XX>)
This action computes the elementary anti-bounce filter for one failure treatment and returns filter result
ACTION_INFR_GetElDiagWg(OUT <CDN_DIAG_WG>, OUT <ERR_DIAG_WG>)
This action reads the failure and the conditions information for each symptom from the infrastructure

A.90.1 General information

The Waste Gate (WG) is driven by the ECU via an output driver. The failure detection is done by ECU hardware. The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

After activation conditions are met, the diagnosis activation is delayed for NC_PSD_DLY_WG executions, to avoid the usage of wrong infrastructure information.

Error-symptoms and conditions are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> SYM_0: Short circuit to battery (SCP)
 │ │ └───> SYM_1: Short circuit to ground (SCG)
 │ └─────> SYM_2: Open circuit (OC)
 └───────> SYM_3
```

Configuration for diagnostic symptoms:

Diagnostic WG	Symptoms	Nb	Filter type
WG	SCP	*0*	*MPL_STD_INI*
	SCG	*1*	
	OC	2	
		3	

The data LV_END_DIAG_WG and LV_CDN_DIAG_WG are not present in output data, since they are not used by other functions. Nevertheless; they are provided in error management and always visible.

The error management variables are initialized according to filter type.

Application Condition

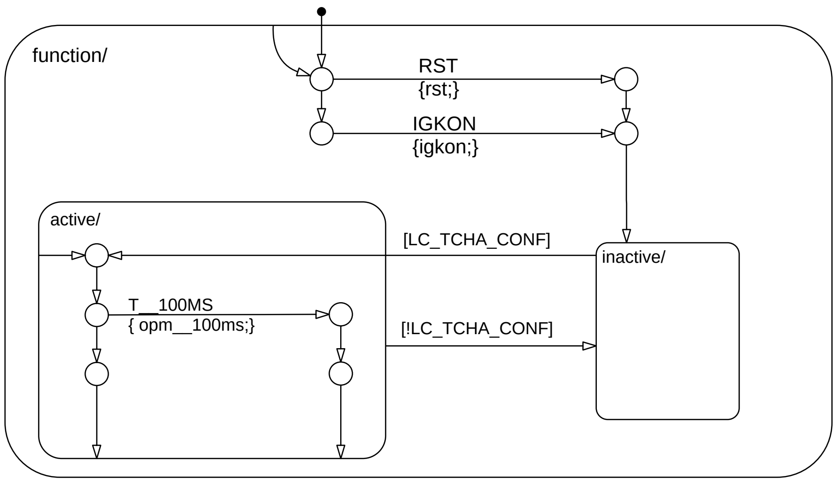


Figure 67 CHRG_WGPSD0/ APP_CDN/ Chart

Function Description

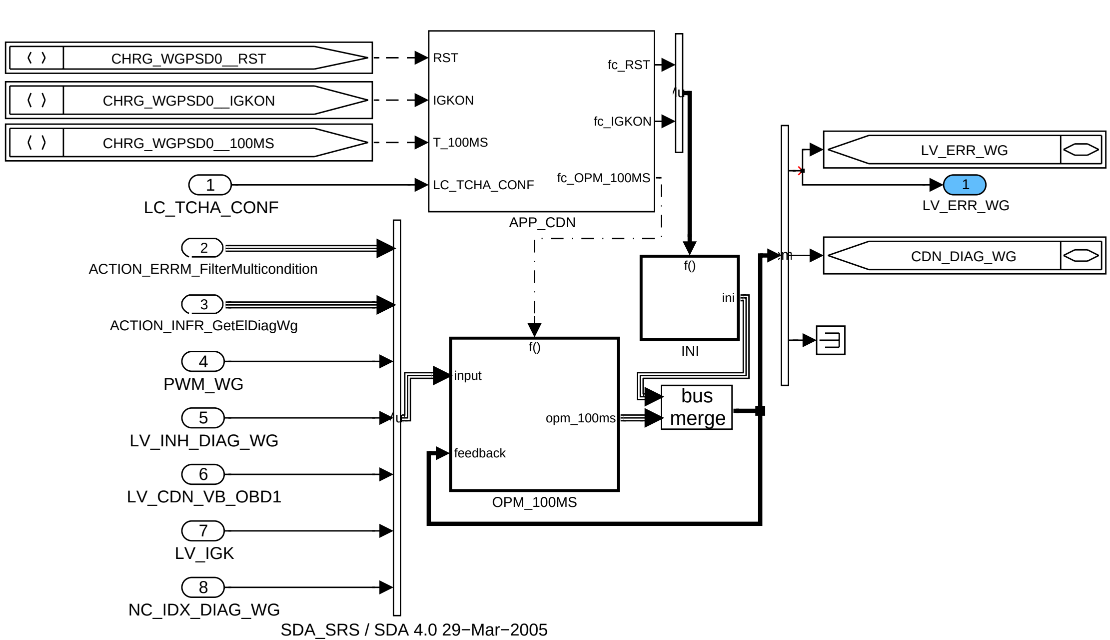


Figure 68 CHRG_WGPSD0

A.90.1.1 Initialization

At reset and ignition-key-on

Figure 69 CHRG_WGPSD0/ INI/ INI

A.90.1.2 Formula section

Validity of diagnosis and call of error management

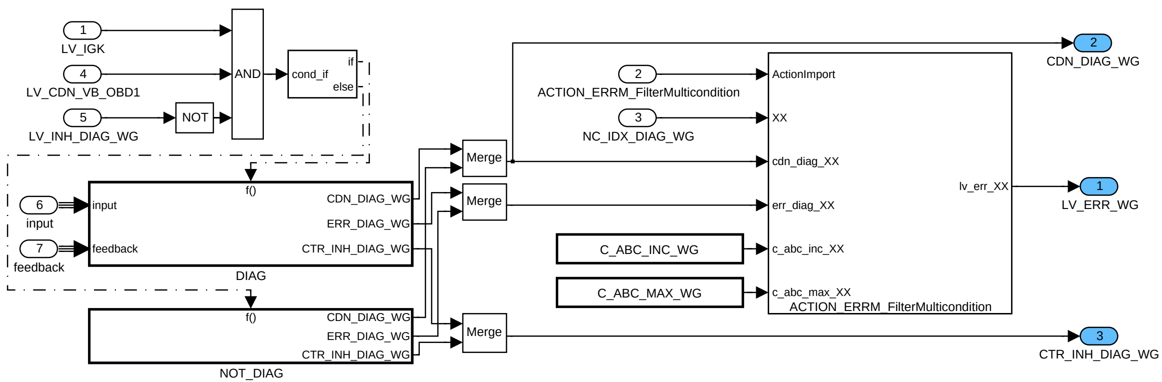


Figure 70 CHRG_WGPSD0/ OPM_100MS/ DIAGNOSIS

Valid diagnosis

The diagnosis information (failure symptoms – raw value – ERR_DIAG_WG and basic diagnosis conditions CDN_DIAG_WG) is received from the infrastructure by action-call.

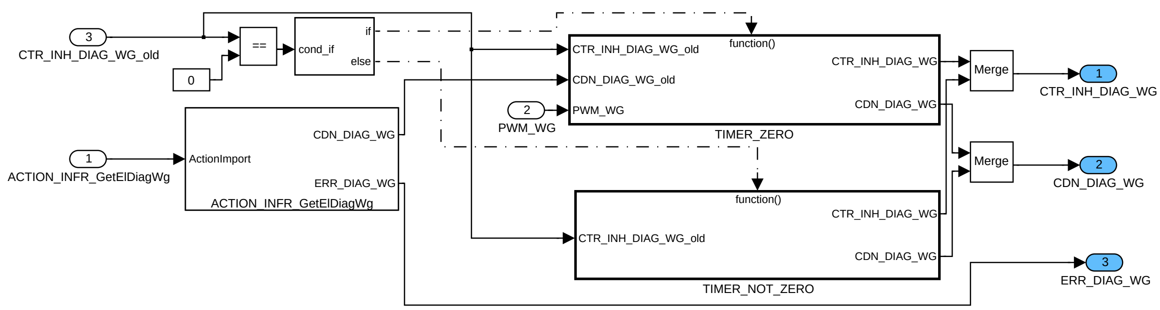


Figure 71 CHRG_WGPSD0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG

Timer elapsed

Additional diagnosis conditions have to be considered. Depending on the value of PWM_WG the diagnosis of SCP (bit 8), SCG (bit 7) or OC (bit 6) is not possible.

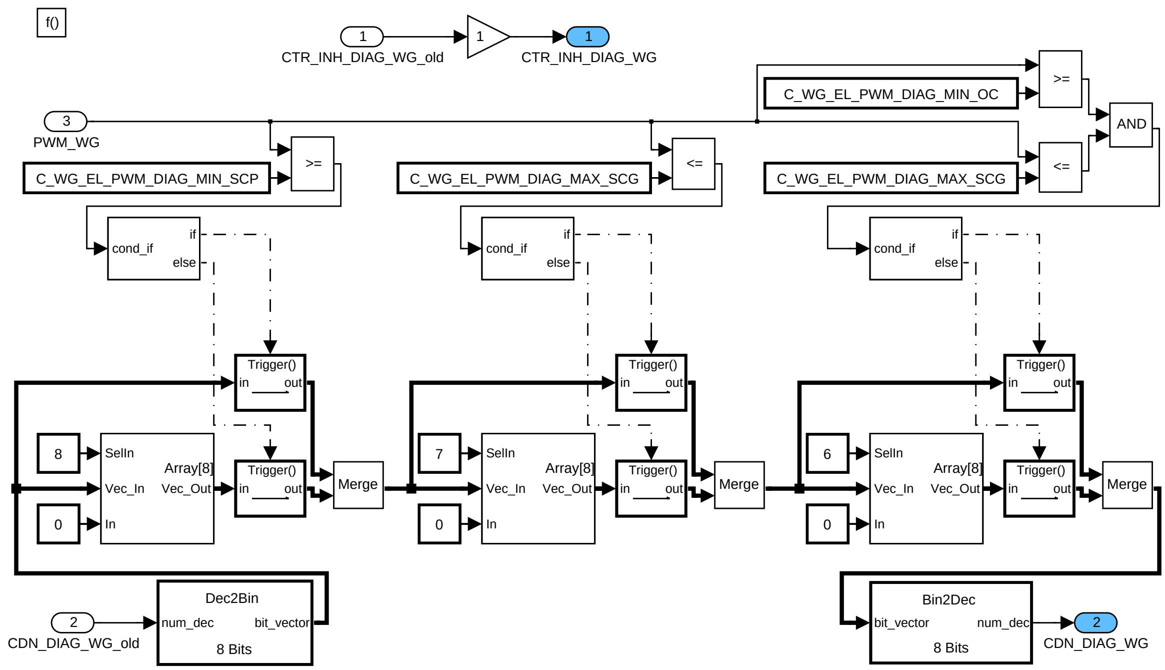


Figure 72 CHRG_WGPSD0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG/ TIMER_ZERO

Timer not yet elapsed

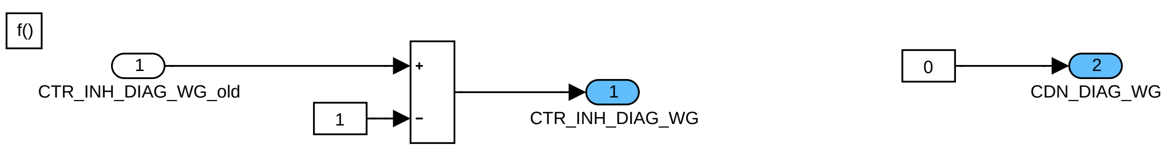


Figure 73 CHRG_WGPSD0/ OPM_100MS/ DIAGNOSIS/ DIAG/ DIAG/ TIMER_NOT_ZERO

Invalid diagnosis

Figure 74 CHRG_WGPSD0/ OPM_100MS/ DIAGNOSIS/ NOT_DIAG

A.91 Turbo charger diagnosis (over speed and over temperature)

Overview

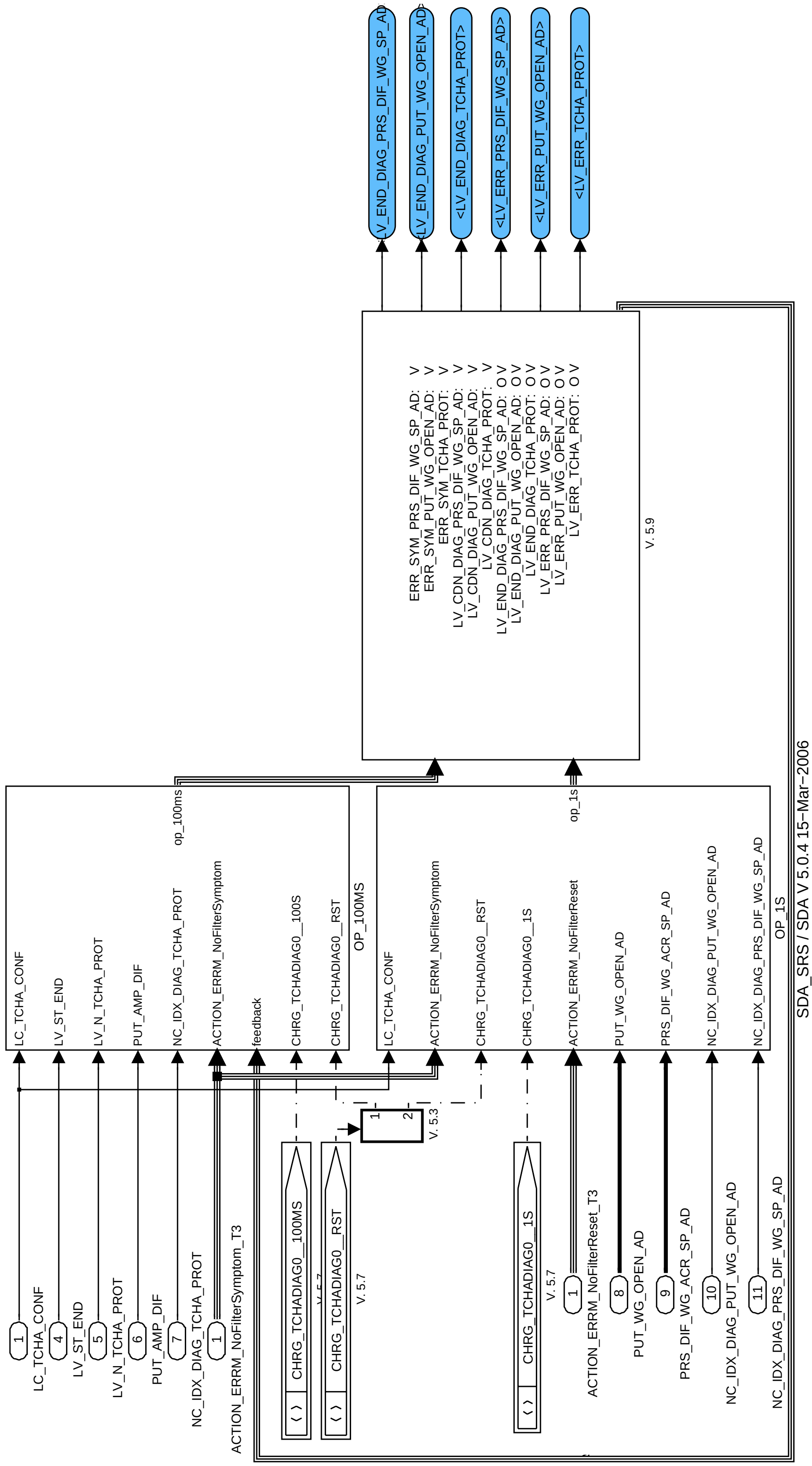


Figure 75:
Path: CHRG_TCHADIAG0

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
ERR_SYM_PRS_DIF_WG_SP_AD	V	0 1 2 4 8	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	-	[-]
Symptom for the adapted values of the pressure control upstream of throttle (PUT_CTL)					
ERR_SYM_PUT_WG_OPEN_AD	V	0 1 2 4 8	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	-	[-]
Symptom for the adapted values of the basic charge air pressure					
ERR_SYM_TCHA_PROT	V	0 1 2 4 8	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	-	[-]
Symptom for disabled charged engine operation					
LV_CDN_DIAG_PRS_DIF_WG_SP_AD	V	0... 1H	0... 1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_CDN_DIAG_PUT_WG_OPEN_AD	V	0... 1H	0... 1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_CDN_DIAG_TCHA_PROT	V	0... 1H	0... 1	1	[-]
Boolean for disabled charged engine operation diagnosis conditions					
LV_END_DIAG_PRS_DIF_WG_SP_AD	O/V	0... 1H	0... 1	1	[-]
Diagnosis performed at least one time					
LV_END_DIAG_PUT_WG_OPEN_AD	O/V	0... 1H	0... 1	1	[-]
Diagnosis performed at least one time					
LV_END_DIAG_TCHA_PROT	O/V	0... 1H	0... 1	1	[-]
Boolean for end of disabled charged engine operation diagnosis					
LV_ERR_PRS_DIF_WG_SP_AD	O/V	0... 1H	0... 1	1	[-]
Present failure after filtering of diagnosis pressure control upstream of throttle (PUT_CTL)					
LV_ERR_PUT_WG_OPEN_AD	O/V	0... 1H	0... 1	1	[-]
Present failure after filtering of diagnosis basic charge air pressure adaptation					
LV_ERR_TCHA_PROT	O/V	0... 1H	0... 1	1	[-]
Charged engine operation disabled because of turbo charger protection					

Input Data:

LC_TCHA_CONF	LV_ST_END	LV_N_TCHA_PROT	PUT_AMP_DIF
NC_IDX_DIAG_TCHA_PROT	PUT_WG_OPEN_AD [6]	PRS_DIF_WG_ACR_SP_AD [9] [9]	NC_IDX_DIAG_PUT_WG_OPEN_AD
NC_IDX_DIAG_PRS_DIF_WG_SP_AD			

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
C_PRS_DIF_WG_ACR_SP_AD_BOL_DIAG	1	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Bottom limit of PUT control adaptation range for diagnosis					
C_PRS_DIF_WG_ACR_SP_AD_TOL_DIAG	1	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Top limit of PUT control adaptation range for diagnosis					

C_PUT_AMP_DIF_DIAG_TCHA_PROT	1	8000... 7FFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
Minimum pressure difference to set end flag TCHA_PROT in the error free system					
C_PUT_WG_OPEN_AD_BOL_DIAG	1	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Bottom limit of PUT controller adaptation map entries for diagnosis					
C_PUT_WG_OPEN_AD_TOL_DIAG	1	0... FFFFH	-2717... 2716.91708374	0.0829163	[hPa]
Top limit of PUT controller adaptation map entries for diagnosis					

Import Actions:

ACTION_ERRM_NoFilterSymptom(**IN** <IDX_DIAG>, **IN** <LV_CDN_DIAG>, **IN** <ERR_SYM>, **IN** <LV_ERR_SET>, **IN** <LV_ERR_RST>, **IN** <LV_END_DIAG>, **OUT** <LV_ERR>)
ACTION_ERRM_NoFilterReset(**IN** <IDX_DIAG>, **OUT** <LV_ERR>)

General Information

This module performs three diagnosis operations from three different error sources. The first diagnosis, which tests if the turbo charger speed is too high, is called every 100 ms.

The second diagnosis tests if the adapted values of the basic charge air pressure (PUT_WG_OPEN_AD) are above or below defined thresholds.

Similar to the second diagnosis, the third diagnosis tests if the adapted values of the pressure control upstream of throttle (PUT_CTL) are below or above defined thresholds.

The second and third diagnoses are called every second.

A.91.1 General information

Turbo charger speed diagnosis

For turbo charged engines charged engine operation is disabled for the rest of the current driving cycle if charger speed exceeds a critical threshold. If this happens the error bit LV_ERR_TCHA_PROT is set. The symptom of the error code is not handled by anti-bouncing.

Error symptoms for the diagnosis function are defined as following:

> SYM_0
> SYM_1: Too high charger speed
> SYM_2
> SYM_3

Configuration for diagnostic symptoms:

Diagnostic TCHA_PROT	Symptoms	Nb	Filter type
Turbo charger diagnosis			*NO_FIL*
	Too high charger speed	*SYM_I*	

The error management variables are initialized according to filter type.

Application Conditions

Initialization: RST
Recurrence: 100MS
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

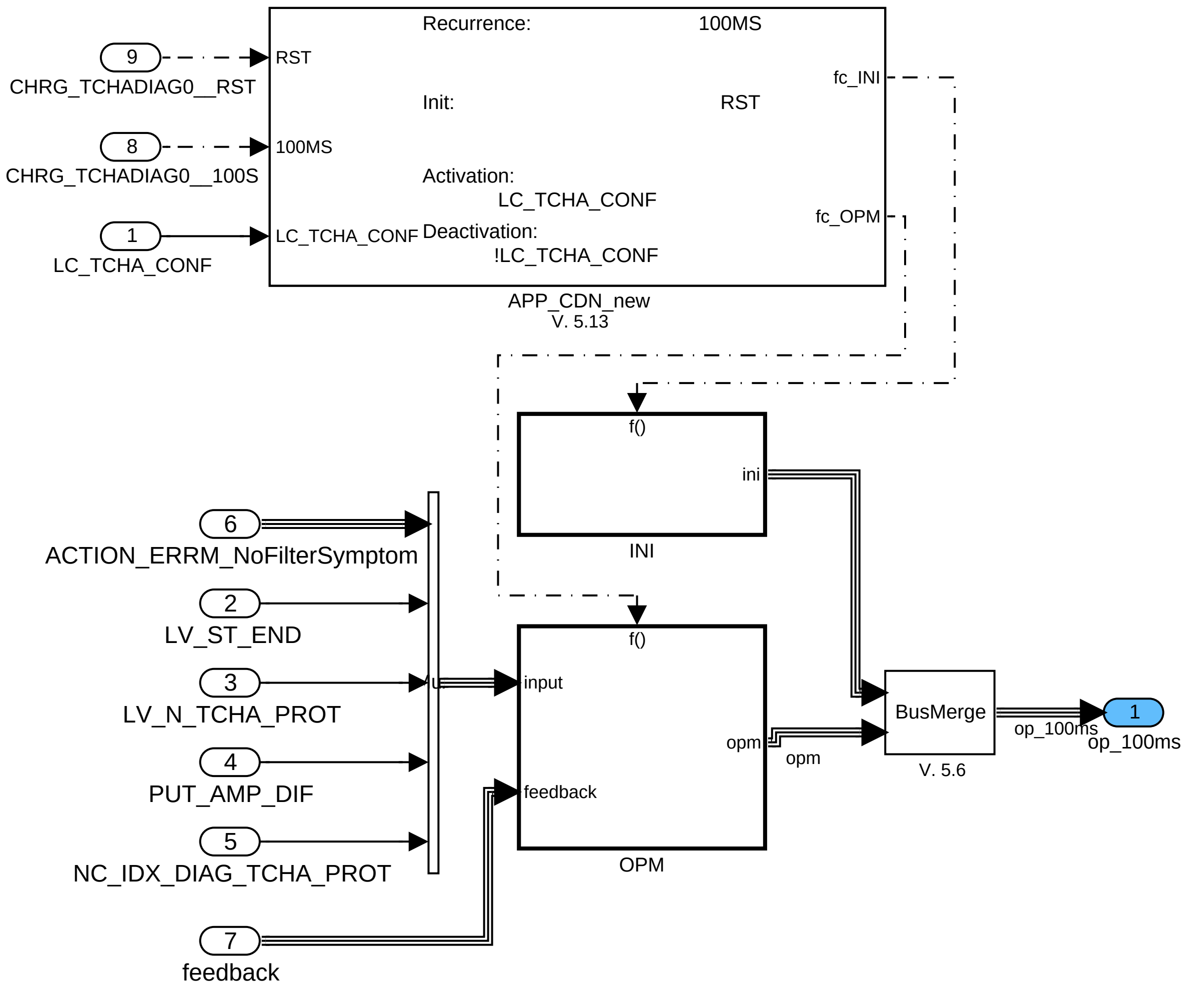


Figure 76:
Path: CHRG_TCHADIAG0/OP_100MS

A.91.1.1 Initialization

A.91.1.1.1 At EXIT_ST

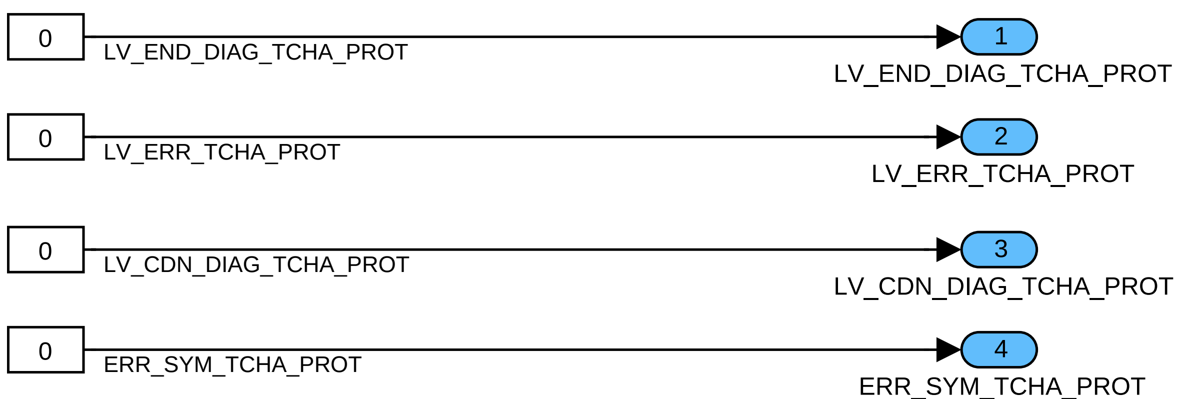


Figure 77:
Path: CHRG_TCHADIAG0/OP_100MS/INI/CLC_INI

A.91.1.2 Formula section

A.91.1.2.1 Validity of diagnosis and call of error-management

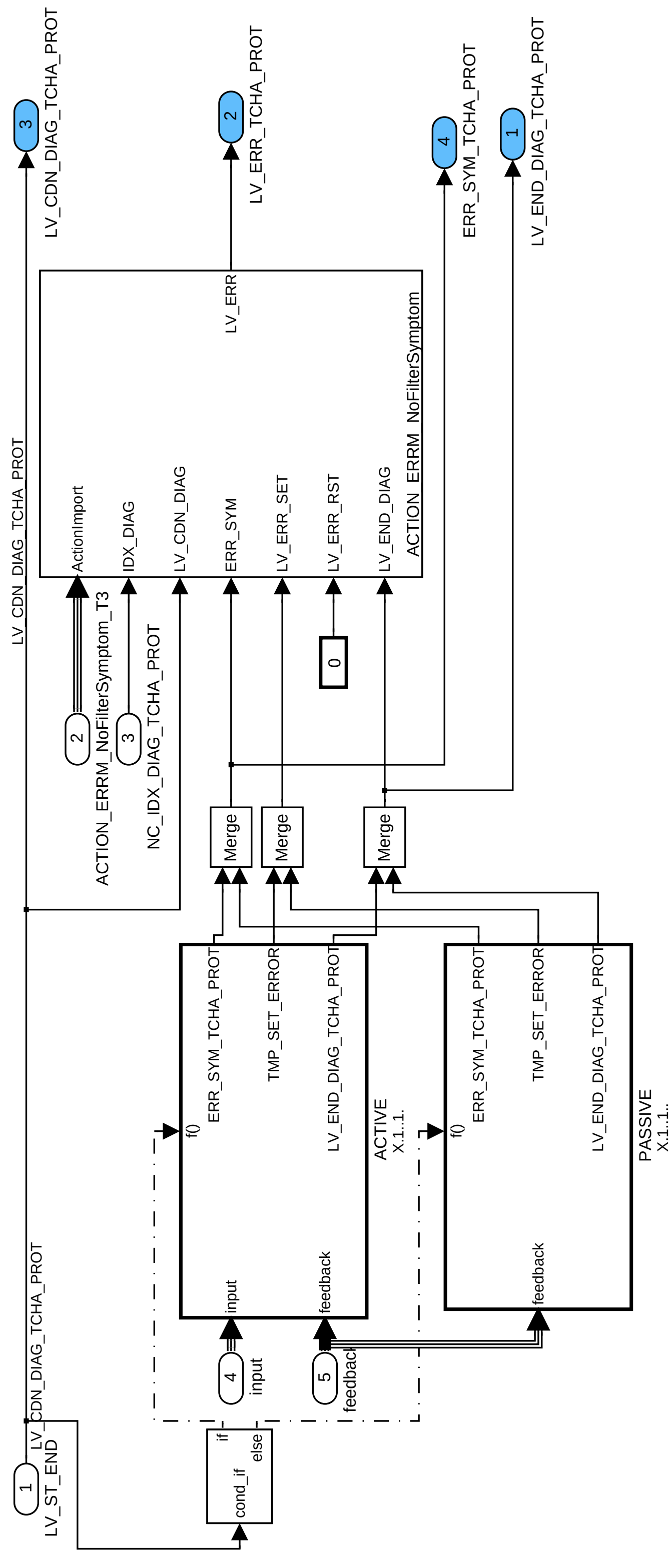


Figure 78:
Path: CHRG_TCHADIAG0/OP_100MS/OPM/CLC_OPM

A.91.1.2.1.1 CHRG_TCHADIAG0/OP_100MS/OPM/CLC_OPM/ACTIVE

A.91.1.2.1.1.1 Valid diagnosis

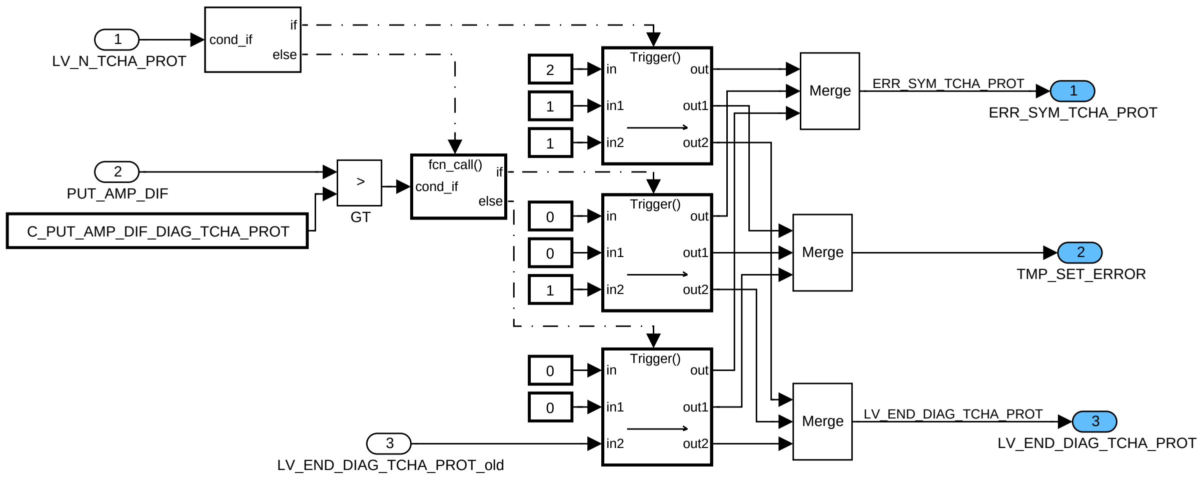


Figure 79:
Path: CHRG_TCHADIAG0/OP_100MS/OPM/CLC_OPM/ACTIVE/ACTIVE

A.91.1.2.1.2 CHRG_TCHADIAG0/OP_100MS/OPM/CLC_OPM/PASSIVE

A.91.1.2.1.2.1 Invalid diagnosis

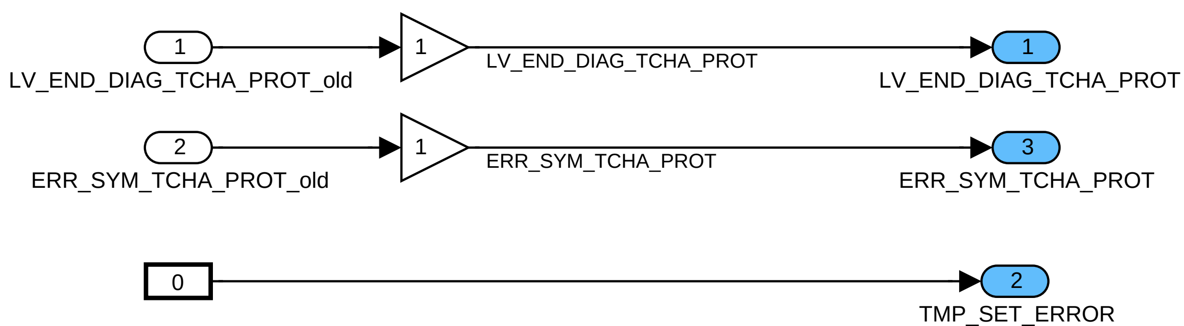


Figure 80:
Path: CHRG_TCHADIAG0/OP_100MS/OPM/CLC_OPM/PASSIVE/PASSIVE

A.91.2 CHRG_TCHADIAG0/OP_1S

Basic charge air pressure adaptation diagnosis and PUT_CTL diagnosis

The module performs the diagnoses for basic charge air pressure (PUT_WG_OPEN) adaptation and for PUT_CTL adaptation every second.

Basic charge air pressure adaptation diagnosis (PUT_WG_OPEN_AD)

Tests if the adapted values of the basic charge air pressure (PUT_WG_OPEN_AD) are above or below defined thresholds (C_PUT_WG_OPEN_AD_BOL and C_PUT_WG_OPEN_AD_TOL, respectively). If at least one value in PUT_WG_OPEN_AD is outside of the thresholds, the error flag LV_ERR_PUT_WG_OPEN_AD is set. No anti-bounce filters are used.

Error-symptoms for the diagnosis function are defined as following:

> SYM_0: PUT_WG_OPEN adaptation values below lower threshold
> SYM_1: PUT_WG_OPEN adaptation values above upper threshold
> SYM_2
> SYM_3

Configuration for diagnostic symptoms:

Diagnostic PUT_WG_OPEN _AD	Symptoms	Nb	Filter type
Turbo charger diagnosis			*NO_FIL*
	PUT_WG_OPEN adaptation values below lower threshold	*SYM_0*	
	PUT_WG_OPEN adaptation values above upper threshold	SYM_1	

The error management variables are initialized according to filter type.

PUT_CTL diagnosis

Tests if the adapted values of the pressure control upstream of throttle (PUT_CTL) are below or above defined thresholds (C_PRS_DIF_WG_ACR_SP_AD_BOL and C_PRS_DIF_WG_ACR_SP_AD_TOL, respectively). If at least one value in PRS_DIF_WG_ACR_SP_AD is outside of the thresholds, the error flag LV_ERR_PRS_DIF_WG_SP_AD is set. No anti-bounce filters are used.

Error-symptoms for the diagnosis function are defined as following:

> SYM_0: PUT_CTL adaptation values below lower threshold
> SYM_1: PUT_CTL adaptation values above upper threshold
> SYM_2
> SYM_3

Configuration for diagnostic symptoms:

Diagnostic PUT_CTL	Symptoms	Nb	Filter type
Turbo charger diagnosis			*NO_FIL*
	PUT_CTL adaptation values below lower threshold	*SYM_0*	
	PUT_CTL adaptation values above upper threshold	SYM_1	

The error management variables are initialized according to filter type.

Application Conditions

Initialization: RST
Recurrence: 1S
Activation: LC_TCHA_CONF
Deactivation: !LC_TCHA_CONF

Function description

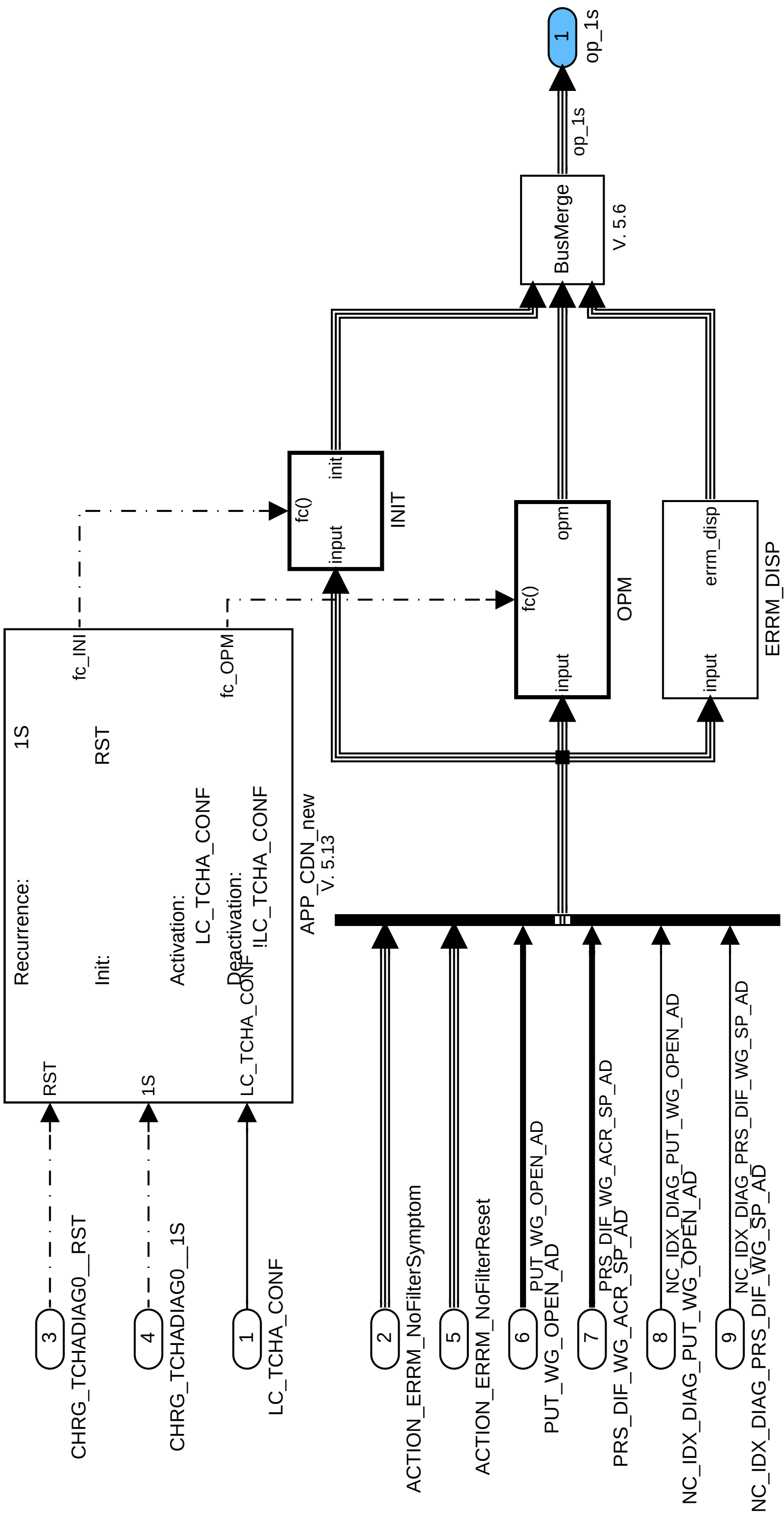


Figure 81:
Path: CHRG_TCHADIAG0/OP_1S

A.91.2.1 SUBFUNCTION: INIT

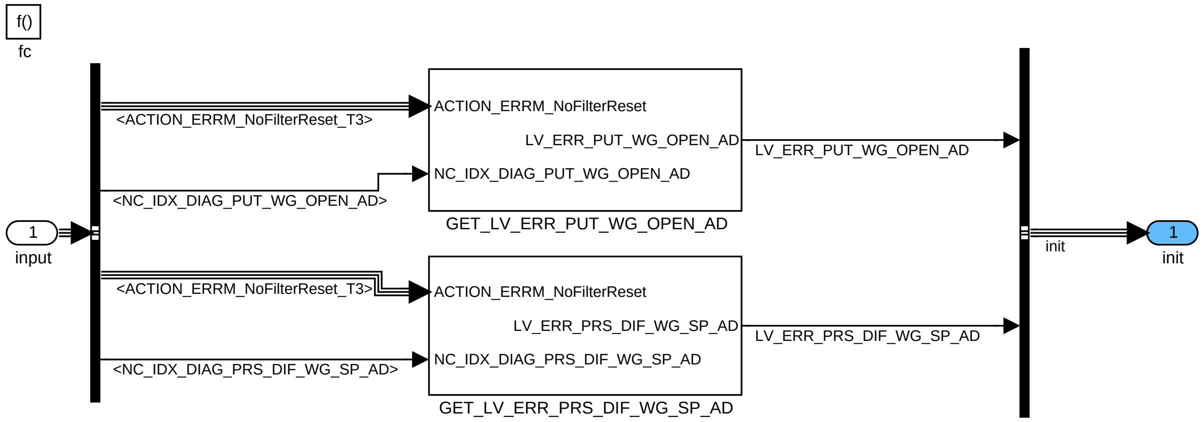


Figure 82:
Path: CHRG_TCHADIAG0/OP_1S/INIT

A.91.2.1.1 No title given

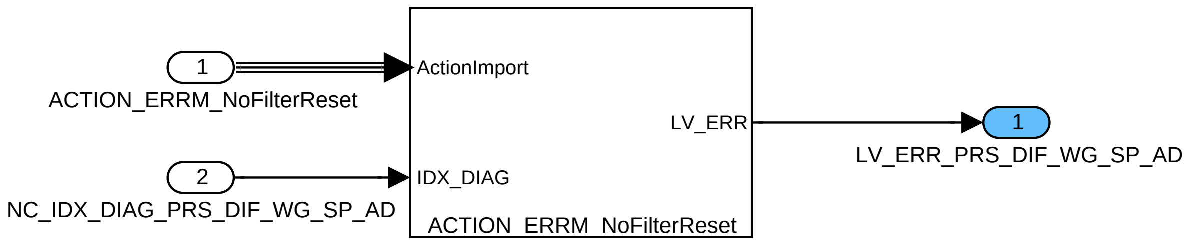


Figure 83:
Path: CHRG_TCHADIAG0/OP_1S/INIT/GET_LV_ERR_PRS_DIF_WG_SP_AD

A.91.2.1.2 No title given

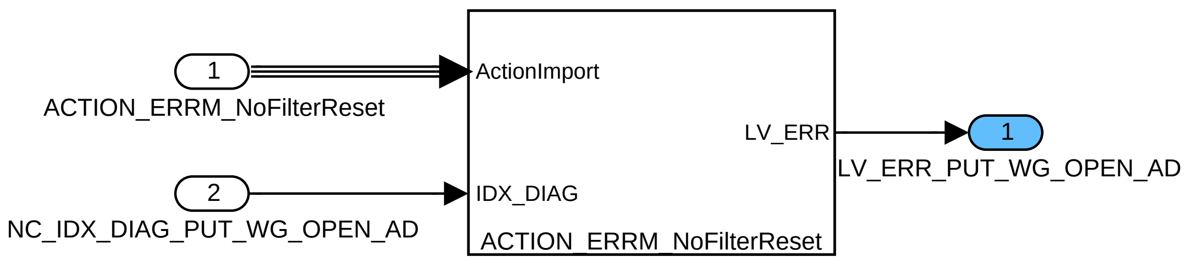


Figure 84:
Path: CHRG_TCHADIAG0/OP_1S/INIT/GET_LV_ERR_PUT_WG_OPEN_AD

A.91.2.2 SUBFUNCTION: OPM

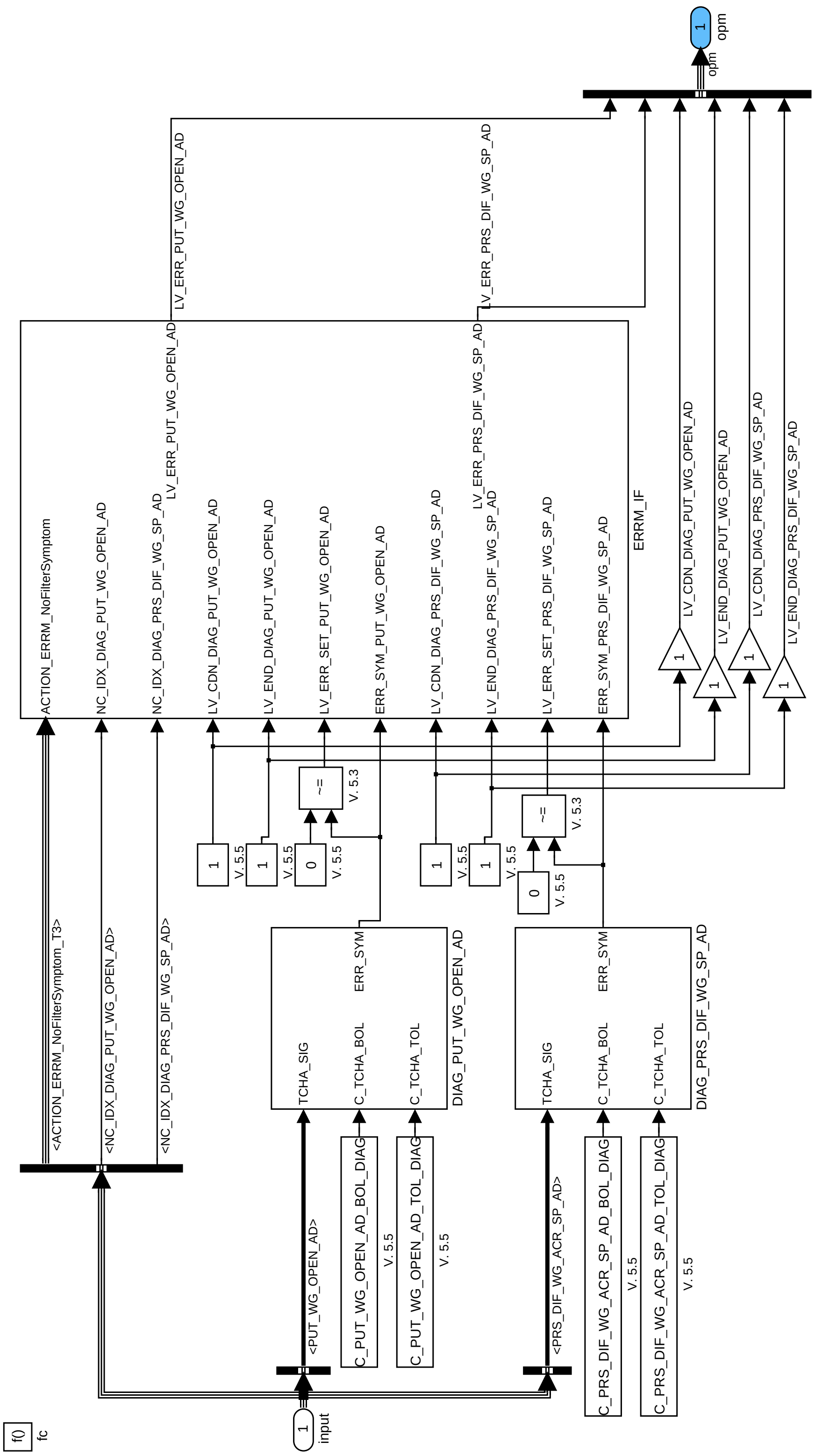


Figure 85:
Path: CHRG_TCHADIAG0/OP_1S/OPM

A.91.2.2.1 No title given

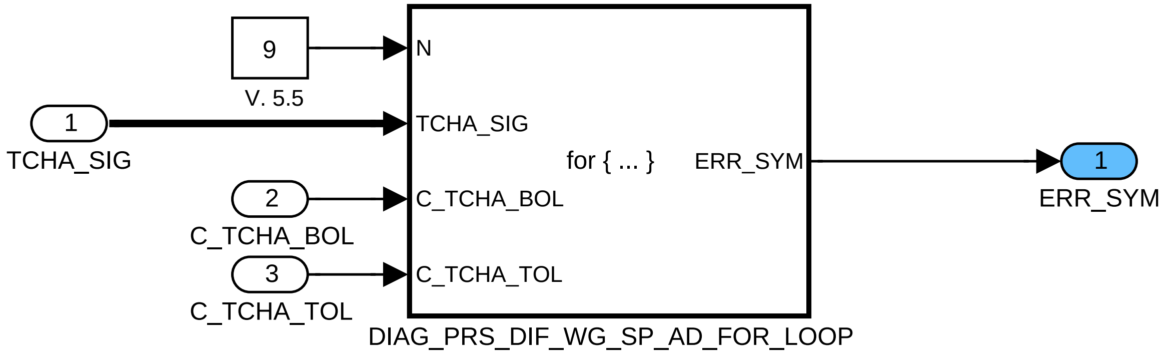


Figure 86:
Path: CHRG_TCHADIAG0/OP_1S/OPM/DIAG_PRS_DIF_WG_SP_AD

A.91.2.2.1.1 No title given

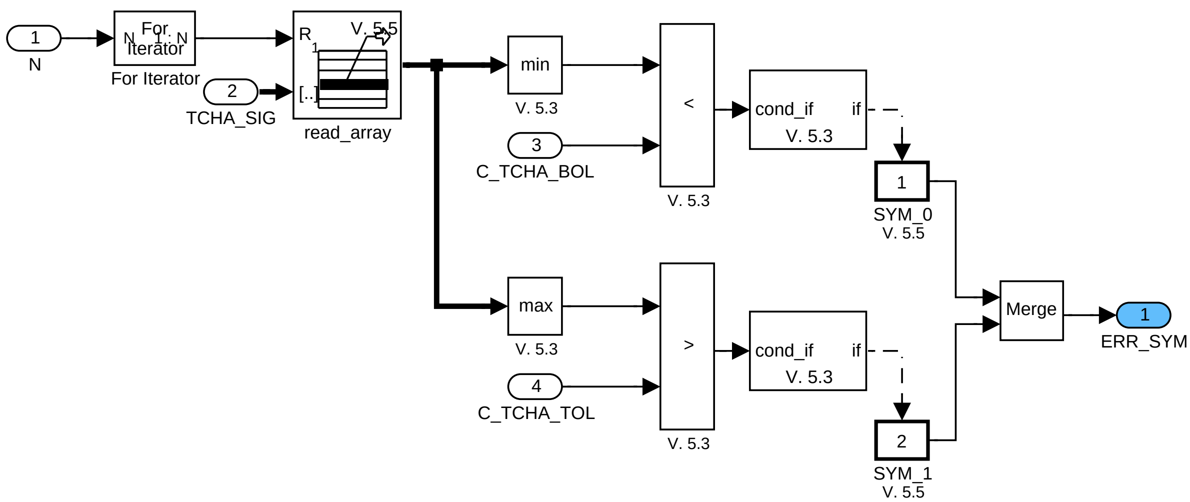


Figure 87:
Path: CHRG_TCHADIAG0/OP_1S/OPM/DIAG_PRS_DIF_WG_SP_AD/DIAG_PRS_DIF_WG_SP_AD_FOR_LOOP

A.91.2.2.2 No title given

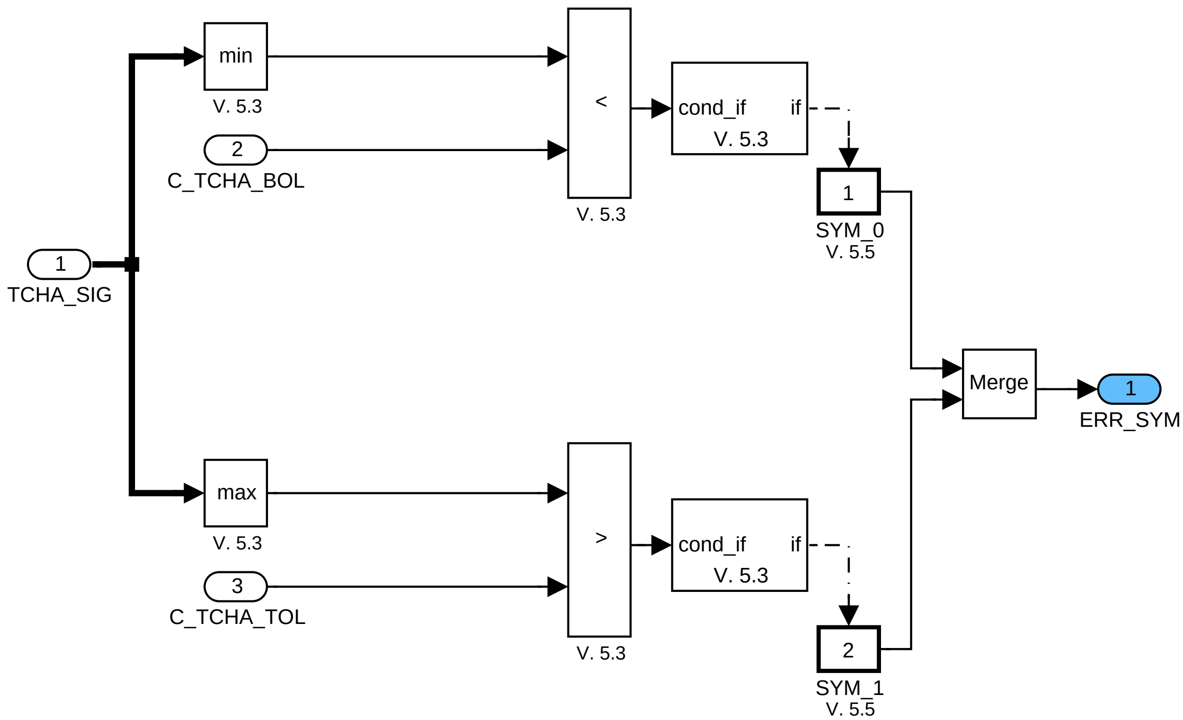


Figure 88:
Path: CHRG_TCHADIAG0/OP_1S/OPM/DIAG_PUT_WG_OPEN_AD

A.91.2.2.3 No title given

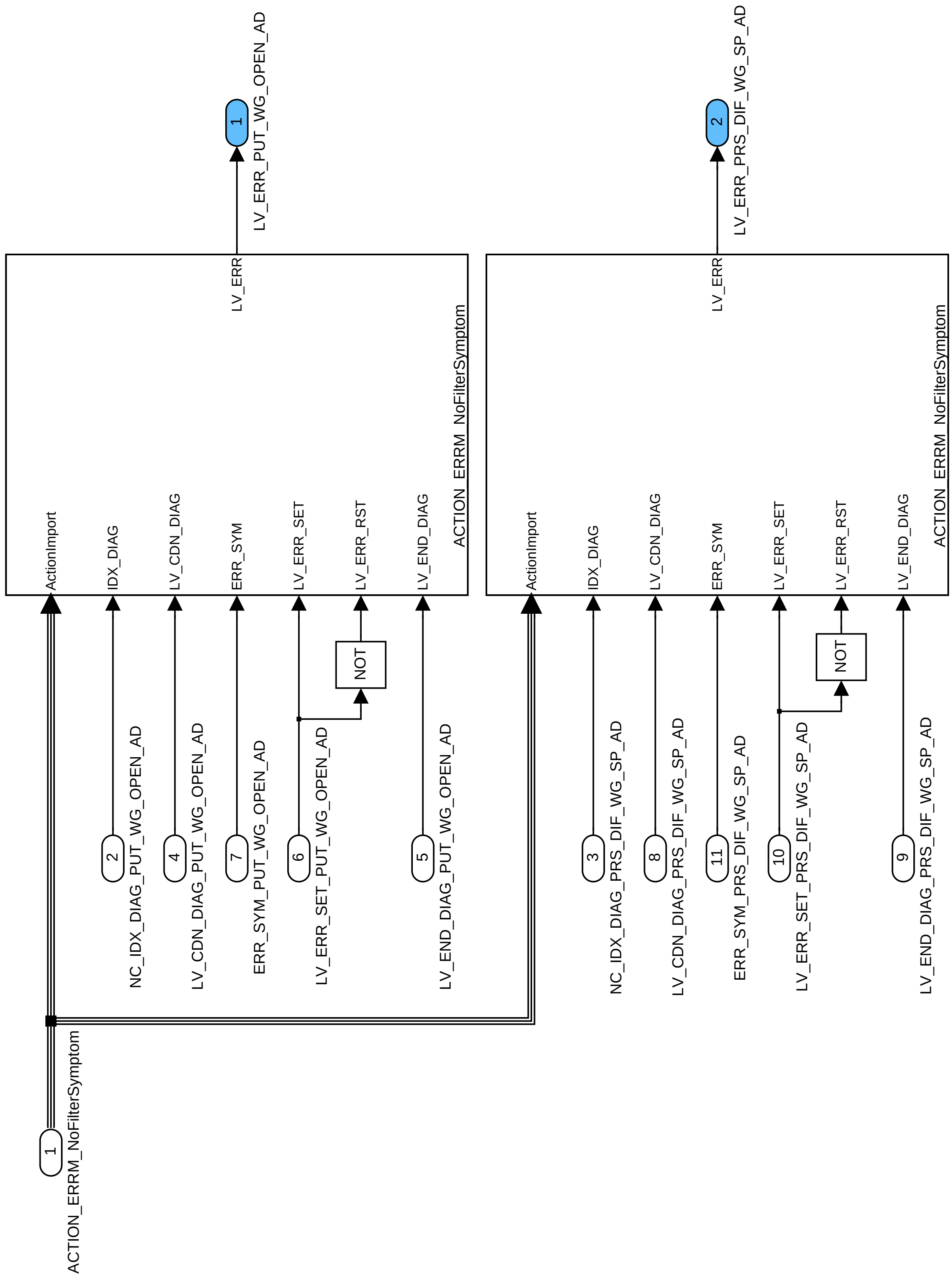


Figure 89:
Path: CHRG_TCHADIAG0/OP_1S/OPM/ERRM_IF

A.92 Waste gate electrical diagnosis (Appl. Inc.)

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_INH_DIAG_WG	O/V	0... 1H	0... 1	1	[-]
Inhibition condition for the WG diagnostic					

Input Data:

LV_IGK	LC_TCHA_CONF	LV_ST

General Information

The flag LV_INH_DIAG_WG permit to deactivate the corresponding diagnostic.

Application Conditions

Initialization: RST
Recurrence: 100MS
Activation: LV_IGK==1
Deactivation: LV_IGK==0

Function description

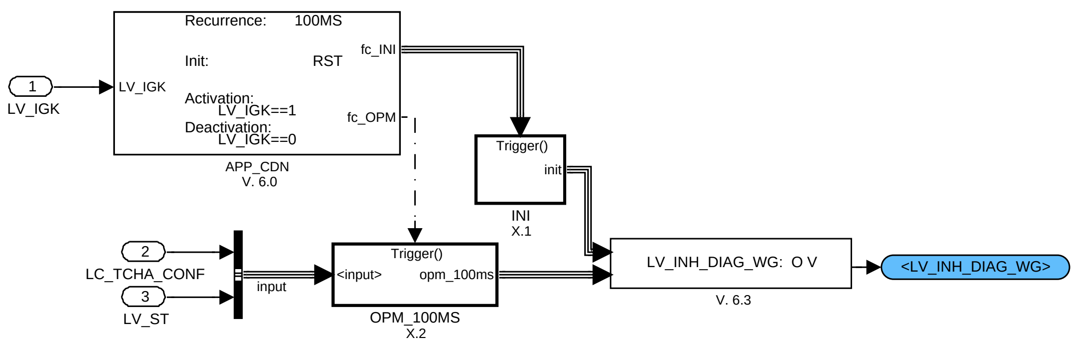


Figure 90:

A.92.1 Initialisation

A.92.1.1 At reset

LV_INH_DIAG_WG is set to zero.

A.92.2 Recurrence 100 ms

A.92.2.1 Calculation of LV_INH_DIAG_WG

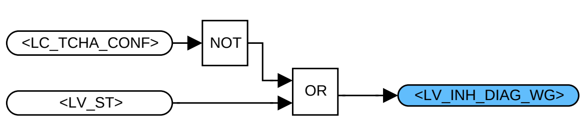


Figure 91:

A.93 Brake Switch Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_BLS_BTS	V / O	0...1H	0..1	1	-
Logic variable for detected BLS-BTS-plausibility-error					
LV_END_DIAG_BLS_BTS	V / O	0...1H	0..1	1	-
BLS-BTS-plausibility diagnosis done at least one time					
LV_CDN_DIAG_BLS_BTS	V / O	0...1H	0..1	1	-
BLS-BTS-plausibility diagnosis condition to start symptom detection					
ERR_SYM_BLS_BTS	V / O	0H 4H	NO_SYM SYM_2	1	-
BLS-BTS-Plausibility detected symptom					
T_1_BLS_BTS_NOT_PLAUS	V	0...FFH	0...25,5	0,1	s
Timer 1 for BLS-BTS-Plausibility detection					
T_2_BLS_BTS_NOT_PLAUS	V	0...FFH	0...25,5	0,1	s
Timer 2 for BLS-BTS-Plausibility detection					
LV_BTS_PUSH	O	0...1	0...1	1	-
Flag to indicate pushed BTS detected					
LV_BTS_RLS	O	0...1	0...1	1	-
Flag to indicate released BTS detected					
LV_BLS_PUSH	O	0...1	0...1	1	-
Flag to indicate pushed BLS detected					
LV_BLS_RLS	O	0...1	0...1	1	-
Flag to indicate released BLS detected					

Input data:

LV_BLS	LV_BTS	LV_IGK	LV_ES
LV_INH_DIAG_BLS_BTS	LC_ENA_BLS_BTS_ENG_STOP	LV_AT	

General information:

The brake light switch (BLS) and the brake test switch (BTS) are checked for plausibility versus each other. The anti-bounce counter is incremented, if BLS and BTS are not plausible longer than C_T_1_BLS_BTS_NOT_PLAUS_MIN or C_T_2_BLS_BTS_NOT_PLAUS_MIN sec. depending on case, or if BTS is reased and BLS was not detected as pushed.

The anti-bounce counter is decremented if "braking" and "not braking" are clearly detected.

BLS - BTS error symptoms :

```
[ | | | ]
 | | | └─> (not used)            (= SYM_0)
 | | └───> (not used)            (= SYM_1)
 | └─────> BLS-BTS not plausible (= SYM_2)
 └───────> (not used)            (= SYM_3)
```

Application conditions:

Initialisation: LV_ERR_BLS_BTS, LV_END_DIAG_BLS_BTS, LV_CDN_DIAG_BLS_BTS, ERR_SYM_BLS_BTS:

according filter type: **"STD-INI"**at reset :

T_1_BLS_BTS_NOT_PLAUS, T_2_BLS_BTS_NOT_PLAUS, LV_BTS_PUSH, LV_BTS_RLS, LV_BLS_PUSH, LV_BLS_RLS:

all equal to "0"

Recurrence: 100 ms (note: the formula section is not calculated during the first recurrence, when the activation conditions are fulfilled for the first time)

Activation: LV_IGK = 1

and (LV_ES = 0 **or** (LC_ENA_BLS_BTS_ENG_STOP = 1 **and** LV_AT = 1))

and LV_INH_DIAG_BLS_BTS = 0

Deactivation: otherwise

Formula section:

If LV_BTS 0->1

Then LV_BTS_PUSH = 1

Endif

If LV_BTS 1->0

Then LV_BTS_RLS = 1

Endif

If LV_BLS 0->1

Then LV_BLS_PUSH = 1

Endif

If LV_BLS 1->0

Then LV_BLS_RLS = 1

Endif

If LV_BTS_PUSH = 0

and LV_BTS = 1

and LV_BLS = 0

Then $T\_1\_BLS\_BTS\_NOT\_PLAUS_{(n)} = T\_1\_BLS\_BTS\_NOT\_PLAUS_{(n-1)} + 1$

Endif

If LV_BLS = 1 **and** LV_BTS = 0

Then $T\_2\_BLS\_BTS\_NOT\_PLAUS_{(n)} = T\_2\_BLS\_BTS\_NOT\_PLAUS_{(n-1)} + 1$

Else T_2_BLS_BTS_NOT_PLAUS = 0

Endif

Remark: calculation of all xx_PUSH and xx_RLS Flags and all timers must be done prior to symptom calculation

Symptom calculation :

```
If(1)     LV_BLS_RLS = 1
Then(1)   If(2)      LV_BLS_PUSH = 1
                and  LV_BTS_PUSH = 1
                and  LV_BTS_RLS = 1
          Then(2)    LV_CDN_DIAG_BLS_BTS = 1
                     ERR_SYM_BLS_BTS = 0H        (NO_SYM)
                     LV_BLS_PUSH = 0
                     LV_BLS_RLS = 0
                     LV_BTS_PUSH = 0
                     LV_BTS_RLS = 0
          Else(2)    LV_BLS_PUSH = 0
                     LV_BLS_RLS = 0
                     LV_BTS_PUSH = 0
                     LV_BTS_RLS = 0
Else(1)   If(3)      LV_BTS_RLS = 1
                and  LV_BLS_PUSH = 0
                and  T_2_BLS_BTS_NOT_PLAUS = 0
          Then(3)    LV_CDN_DIAG_BLS_BTS = 1
                     ERR_SYM_BLS_BTS = 4H              (SYM_2)
                     LV_BTS_PUSH = 0
                     LV_BTS_RLS = 0
          Else(3)
                     If(4)      T_2_BLS_BTS_NOT_PLAUS > C_T_2_BLS_BTS_NOT_PLAUS_MIN
                     Then(4)    LV_CDN_DIAG_BLS_BTS = 1
```

ERR_SYM_BLS_BTS = 4H (SYM_2)

T_2_BLS_BTS_NOT_PLAUS = 0

LV_BTS_PUSH = 0

LV_BTS_RLS = 0

Else(4) **If(5)** T_1_BLS_BTS_NOT_PLAUS > C_T_1_BLS_BTS_NOT_PLAUS_MIN

Then(5) LV_CDN_DIAG_BLS_BTS = 1

ERR_SYM_BLS_BTS = 4H (SYM_2)

T_1_BLS_BTS_NOT_PLAUS = 0

Else(5) LV_CDN_DIAG_BLS_BTS = 0

Endif(5)

Endif(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_1_BLS_BTS_NOT_PLAUS_MIN	1	0...FFH	0...25,5	0,1	s
minimum time 1 for non-plausibility BLS - BTS					
C_T_2_BLS_BTS_NOT_PLAUS_MIN	1	0...FFH	0...25,5	0,1	s
minimum time 2 for non-plausibility BLS - BTS					

A.94 Brake Switch Diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_BLS_BTS	V / O	0...1H	0...1	1	-
Inhibition condition for 'Brake switch diagnosis'					

Input data:

LV_IGK	LV_ES	LV_AT	LC_ENA_BLS_BTS_ENG_STOP
LV_CDN_VB_OBD1			

General information:

The flag LV_INH_DIAG_BLS_BTS inhibits the corresponding diagnosis (Brake Switch Diagnosis).

Application conditions:

Initialisation: at reset

LV_INH_DIAG_BLS_BTS = 1

Recurrence: 100 ms

Activation: LV_IGK = 1

and *(LV_ES = 0* ***or*** *(LC_ENA_BLS_BTS_ENG_STOP = 1 and LV_AT = 1))*

Deactivation: otherwise

Formula section:

If LC_BLS_BTS_PLAUS_DIAG_ENA = 1 (plausibility check BLS / BTS enabled)

and LV_CDN_VB_OBD1 = 1

then LV_INH_DIAG_BLS_BTS = 0

else LV_INH_DIAG_BLS_BTS = 1

Configuration for diagnostic symptoms:

Diagnostic XX	Symptoms	Nb	Filter type
Brake Switch Diagnosis	-	*0*	*STD_INI*
	-	*1*	
	BLS - BTS not plausible	*2*	
BLS_BTS	-	*3*	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_BLS_BTS	1	0...FFH	0...255	1	-
anti-bounce counter increment					
C_ABC_MAX_BLS_BTS	1	1...FFH	1...255	1	-
anti-bounce counter maximum					
LC_BLS_BTS_PLAUS_DIAG_ENA	1	0...1H	0...1	1	-
Plausibility check BLS / BTS enabled					

A.95 Power steering pressure sensor diagnosis

A.95.1 Powersteering pressure electrical diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PSP	O/V	0H...1H	0...1	1	-
Error currently present on power steering pressure signal (after debounce)					
LV_CDN_DIAG_PSP	O/V	0...01H	0...1	1	-
Diagnosis condition for PSP diagnosis					
ERR_SYM_ PSP	O/V	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Error symptom PSP diagnosis					
LV_END_DIAG_PSP	O/V	0...01H	0...1	1	-
End of diagnosis PSP diagnosis					
LV_INH_DIAG_PSP	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition PSP diagnosis					

Input data:

V_PSP_MES	CONF_PSTE	C_PSP_MIN_DIAG	C_PSP_MAX_DIAG
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the analog input signal (V_PSP_MES) from power steering pressure sensor to the micro-controller via the multiplexed.
The application recurrence is **100 ms**.

Description:

Error symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> Short circuit to Vbatt (SCB)                 (= SYM_0)
 │  │  └────> Short circuit to GND (SCG) or Open load (OL)  (= SYM_1)
 │  └───────>                                             (= SYM_2)
 └──────────> -                                           (= SYM_3)
```

Application conditions:

The error detection is performed only if the power steering pressure sensor option has been recognized (CONF_PSTE has been set to 1).

Initialisation: at LV_IGK 0->1 or Reset LV_INH_DIAG_PSP = 0

Activation: **If** LV_IGK = 1

And LV_INH_DIAG_PSP = 0

Then LV_CDN_DIAG_PSP = 1

Else LV_CDN_DIAG_PSP = 0 (*Deactivation)*

Recurrence: 100 ms

Formula section:

IF LV_ERR_VCC_SENS_SUB = 1

THEN LV_INH_DIAG_PSP = 1

ELSE LV_INH_DIAG_PSP = 0

If V_PSP_MES > C_PSP_MAX_DIAG

then *`Short Battery Voltage´ is detected.*
and the anti-bounce counter is incremented with C_ABC_INC_PSP_DIAG
ERR_SYM_ PSP = SYM_0

If V_PSP_MES < C_PSP_MIN_DIAG

then *`Short to Ground´ or `Line Break´ is detected.*
and the anti-bounce counter is incremented with C_ABC_INC_PSP_DIAG
ERR_SYM_PSP = SYM_1

If the anti-bounce counter reaches C_ABC_MAX_PSP_DIAG
Then LV_ERR_PSP = 1
If the anti-bounce counter reaches 0
Then LV_ERR_PSP = 0

* Debounce :

Anti - bounce counter increment : C_ABC_INC_PSP_DIAG
Maximum value of anti - bounce counter : C_ABC_MAX_PSP_DIAG

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_PSP_DIAG	1	0...FFH	0...255	1	-
Anti-bounce counter increment for PSP diagnosis					
C_ABC_MAX_PSP_DIAG	1	0...FFH	0...255	1	-
Maximum value of the anti – bounce counter for PSP diagnosis					

A.95.2 Powersteering pressure plausibility diagnosis(CONF_PSTE = 1)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PSP_PLAUS	O/V	0H...1H	0...1	1	-
Plausibility error of power steering pressure sensor currently present(after debounce)					
LV_CDN_DIAG_PSP_PLAUS	O/V	0...01H	0...1	1	-
Diagnosis condition for PSP plausibility diagnosis					
ERR_SYM_ PSP_PLAUS	O/V	0H 1H	NO_SYM SYM_0	1	-
Error symptom PSP plausibility diagnosis					
LV_END_DIAG_PSP_PLAUS	O/V	0...01H	0...1	1	-
End of diagnosis PSP plausibility diagnosis					
LV_INH_DIAG_PSP_PLAUS	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition PSP plausibility diagnosis					
T_PSP_PLAUS_DIAG	V	0....FFFFH	0...6553.5	0.1	S
Power steering pressure plausibility diagnosis timer					

Input data:

CONF_PSTE	LV_IGK	LV_ES	LV_ERR_VS
LV_ERR_MWSS	TCO	VS	V_PSP_MES
LV_ERR_PSP	LV_ERR_VCC_SENS_SUB		

FUNCTION DESCRIPTION:

General information:

A.95.3 The purpose of this diagnosis is to check the plausibility of the power steering pressure sensor signal.

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> Signal implausible      (= SYM_0)
│ │ └───> -                       (= SYM_1)
│ └─────> -                       (= SYM_2)
└───────> -                       (= SYM_3)
```

Application conditions:

Initialisation: at reset and at transition LV_IGK 0→1; all outputs = 0

Recurrence: 100 ms

Activation: **If** CONF_PSTE = 1 **and**

LV_IGK = 1 **and**

LV_ES = 0 **and**

LV_ST = 0 **and**

LV_ERR_VS = 0 **and**

LV_ERR_MWSS = 0 **and**

LV_ERR_PSP = 0 **and**

LV_ERR_VCC_SENS_SUB = 0 (no Error on Voltage Supply for PSP-Sensor) **and**

VS > C_VS_MIN_PSP_PLAUS_DIAG **and**

TCO > C_TCO_MIN_PSP_PLAUS_DIAG

Then LV_CDN_DIAG_PSP_PLAUS = 1

Else LV_CDN_DIAG PSP_PLAUS = 0

Endif

Deactivation: -

Formula section:

```
If      LV_CDN_DIAG_PSP_PLAUS = 1          and
        V_PSP_MES > C_THD_PSP_PLAUS_DIAG
Then    If      T_PSP_PLAUS_DIAG = 0
        Then    start timer T_PSP_PLAUS_DIAG
        Else    If      T_PSP_PLAUS_DIAG >= C_T_PSP_PLAUS_DIAG
                Then    ERR_SYM_PSP_PLAUS = SYM_0
                        LV_ERR_PSP_PLAUS = 1
                Else    increase timer T_PSP_PLAUS_DIAG
Else    reset timer T_PSP_PLAUS_DIAG
        If      V_PSP_MES < C_THD_PSP_PLAUS_DIAG
        Then    ERR_SYM_PSP_PLAUS = NO_SYM
                LV_ERR_PSP_PLAUS = 0
Endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_PSP_PLAUS_DIAG	1	0...FEH	-48…142,5	0.75	°C
Minimum coolant temperature condition for power steering switch diagnosis					
C_VS_MIN_PSP_PLAUS_DIAG	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for error detection					
C_T_PSP_PLAUS_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Maximum time for T_PSTE with vehicle speed above threshold without error detection					
C_THD_PSP_PLAUS_DIAG	1	0...3FFH	0...5	0.0195	V
Power steering pressure sensor votage threshold for PSP_PLAUS diagnosis					

A.95.4 Powersteering pressure plausibility diagnosis(CONF_PSTE = 0)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PSP_SWI_PLAUS	O/V	0H...1H	0...1	1	-
Plausibility error of power steering pressure switch currently present(after debounce)					
LV_CDN_DIAG_PSP_SWI_PLAUS	O/V	0...01H	0...1	1	-
Diagnosis condition for PSP switch plausibility diagnosis					
ERR_SYM_ PSP_SWI_PLAUS	O/V	0H 1H	NO_SYM SYM_0	1	-
Error symptom PSP switch plausibility diagnosis					
LV_END_DIAG_PSP_SWI_PLAUS	O/V	0...01H	0...1	1	-
End of diagnosis PSP switch plausibility diagnosis					
LV_INH_DIAG_PSP_SWI_PLAUS	O/V	0 ... 1H	0 ... 1	1	-
Diagnosis inhibition PSP switch plausibility diagnosis					
T_PSP_SWI_PLAUS_DIAG	V	0....FFFFH	0...6553.5	0.1	S
Power steering pressure switch plausibility diagnosis timer					

Input data:

CONF_PSTE	LV_IGK	LV_ES	LV_ERR_VS
LV_ERR_MWSS	TCO	VS	LV_PSTE
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

General information:

A.95.5 The purpose of this diagnosis is to check the plausibility of the power steering pressure sensor signal.

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> Signal implausible          (= SYM_0)
│ │ └───> -                           (= SYM_1)
│ └─────> -                           (= SYM_2)
└───────> -                           (= SYM_3)
```

Application conditions:

Initialisation: at reset and at transition LV_IGK 0→1; all outputs = 0

Recurrence: 100 ms

Activation: **If** CONF_PSTE = 0 **and**
LV_IGK = 1 **and**
LV_ES = 0 **and**
LV_ST = 0 **and**
LV_ERR_VS = 0 **and**
LV_ERR_MWSS = 0 **and**

LV_ERR_VCC_SENS_SUB = 0 (no Error on Voltage Supply for PSP-Sensor) **and**

VS > C_VS_MIN_PSP_SWI_PLAUS_DIAG **and**

TCO > C_TCO_MIN_PSP_SWI_PLAUS_DIAG

Then LV_CDN_DIAG_PSP_SWI_PLAUS = 1

Else LV_CDN_DIAG PSP_SWI_PLAUS = 0

Endif

Deactivation: -

Formula section:

If LV_CDN_DIAG_PSP_SWI_PLAUS = 1 **and**

LV_PSTE = 1

Then **If** T_PSP_SWI_PLAUS_DIAG = 0

Then start timer T_PSP_SWI_PLAUS_DIAG

Else **If** T_PSP_SWI_PLAUS_DIAG >= C_T_PSP_SWI_PLAUS_DIAG

Then ERR_SYM_PSP_SWI_PLAUS = SYM_0

LV_ERR_PSP_SWI_PLAUS = 1

Else increase timer T_PSP_SWI_PLAUS_DIAG

Else reset timer T_PSP_SWI_PLAUS_DIAG

If LV_PSTE = 0

Then ERR_SYM_PSP_SWI_PLAUS = NO_SYM

LV_ERR_PSP_SWI_PLAUS = 0

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_PSP_SWI_PLAUS_DIAG	1	0...FEH	-48…142,5	0.75	°C
Minimum coolant temperature condition for power steering switch diagnosis					
C_VS_MIN_PSP_SWI_PLAUS_DIAG	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for error detection					
C_T_PSP_SWI_PLAUS_DIAG	1	0...FFFFH	0...6553.5	0.1	s
Maximum time for T_PSTE with vehicle speed above threshold without error detection					

A.96 Power steering pressure sensor diagnosis (Appl. Inc.)

A.96.1 Powersteering pressure electrical diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Power steering pressure diagnosis	SCB + OL	SYM_0	*STD*
	SCG	SYM_1	
		SYM_2	
PSP		SYM_3	

- Environmental data stored: **VS**
  V_PSP_MES_ENVD_H
  V_PSP_MES_ENVD_L
  TAM

A.96.2 Powersteering pressure plausibility diagnosis(CONF_PSTE = 1)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Power steering pressure plausibility diagnosis	Signal implausible	SYM_0	*STD*
		SYM_1	
		SYM_2	
PSP_PLAUS		SYM_3	

- Environmental data stored: **VS**
  V_PSP_MES_ENVD_H
  V_PSP_MES_ENVD_L
  TAM

A.96.3 Powersteering pressure plausibility diagnosis(CONF_PSTE = 0)

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Power steering switch plausibility diagnosis	Signal implausible	SYM_0	*STD*
		SYM_1	
		SYM_2	
PSP_SWI_PLAUS		SYM_3	

- Environmental data stored: **VS**
  V_PSP_MES_ENVD_H
  V_PSP_MES_ENVD_L
  TAM

A.97 Generator load PWM diagnosis (GEN_LOAD)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_GEN_LOAD_ES	V/O	0...1H	0...1	1	-
Failure state currently present on alternator load PWM GEN_LOAD at engine stop					
ERR_SYM_GEN_LOAD_ES	V	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
symptom: failure without filtering of diagnosis for GEN_LOAD at engine stop					
LV_DIAG_CDN_GEN_LOAD_ES	V	0...1H	0...1	1	-
Conditions of diagnosis fulfilled at engine stop					
LV_END_DIAG_GEN_LOAD_ES	V	0...1H	0...1	1	-
End of diagnosis					
T_DIAG_GEN_LOAD_MAX_ES	V	0...FFFFH	0...6553.5	0.1	sec
Diagnosis counter for SCB and OL error detection at engine stop					
T_DIAG_GEN_LOAD_MIN_ES	V	0...FFFFH	0...6553.5	0.1	sec
Diagnosis counter for SCG error detection at engine stop					
T_DIAG_GEN_LOAD_ES	V	0...FFFFH	0...6553.5	0.1	sec
Diagnosis counter for error healing at engine stop					
LV_ERR_GEN_LOAD	V/O	0...1H	0...1	1	-
Failure state currently present on alternator load PWM GEN_LOAD					
ERR_SYM_GEN_LOAD	V	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
symptom: failure without filtering of diagnosis for GEN_LOAD					
LV_DIAG_CDN_GEN_LOAD	V	0...1H	0...1	1	-
Conditions of diagnosis fulfilled					
LV_END_DIAG_GEN_LOAD	V	0...1H	0...1	1	-
End of diagnosis					
T_DIAG_GEN_LOAD_MIN	V	0...FFFFH	0...65535.5	0.1	sec
Diagnosis counter for SCB and OL error detection					
T_DIAG_GEN_LOAD	V	0...FFFFH	0...6553.5	0.1	sec
Diagnosis counter for error healing					

Input data:

GEN_LOAD	LV_IGK	LV_ST_END	LV_ES
GEN_LOAD_MMV	VB_MMV	N	TCO
CONF_GEN_LOAD	LV_ERR_RLY_MAIN	LV_CDN_VB_OBD1	CONF_BAT

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the input signal (PWM) from the alternator about the current load condition.

This diagnosis is performed in two conditions.

1) During Pre-driving

At key On and engine stop condition, the PWM value from alternator is kept to pre-defined specific value(Valeo : 29%, Denso : 19%) and if the PWM value show 0% or 100%, it is defined as error.

Error-symptoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> Short circuit to GND (SCG)                      (= SYM_0)
 │  │  └────> Short circuit to Vbatt or Open load (SCB/OL)    (= SYM_1)
 │  └───────> -                                               (= SYM_2)
 └──────────> -                                               (= SYM_3)
```

2) During driving

Because short circuit to ground could not be distinguished to normal 100% GEN_LOAD, the diagnosis is performed only to detect open circuit & short circuit battery error.

Error-symptoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │  │  │  └─> -                                               (= SYM_0)
 │  │  └────> Short circuit to Vbatt or Open load (SCB/OL)    (= SYM_1)
 │  └───────> -                                               (= SYM_2)
 └──────────> -                                               (= SYM_3)
```

Application conditions:

Initialisation: *at reset*

If CONF_BAT = 0

then GEN_LOAD_MAX_DIAG_ES = C_GEN_LOAD_MAX_DIAG_ES

GEN_LOAD_MIN_DIAG_ES = C_GEN_LOAD_MIN_DIAG_ES

Different PWM signals at stopped Alternator between BEM and NON BEM

else GEN_LOAD_MAX_DIAG_ES = C_GEN_LOAD_MAX_DIAG_ES_BEM

GEN_LOAD_MIN_DIAG_ES = C_GEN_LOAD_MIN_DIAG_ES_BEM

endif

Activation: LV_IGK = 1 **and**

CONF_GEN_LOAD = 1

Recurrence: 100 ms

Formula section:

1. During pre-driving

If(1) LV_IGK = 1

and run out C_T_DLY_GEN_LOAD after LV_IGK = 1

and	LV_ES = 1	
and	N = 0	
and	LV_ERR_RLY_MAIN = 0	
and	LV_CDN_VB_OBD1 = 1	
Then	LV_DIAG_CDN_GEN_LOAD_ES = 1	
Else	LV_DIAG_CDN_GEN_LOAD_ES = 0	

If	LV_DIAG_CDN_GEN_LOAD_ES = 1	
Then	**If**	GEN_LOAD > GEN_LOAD_MAX_DIAG_ES
	Then	T_DIAG_GEN_LOAD_MAX_ES is decreased.
	Else	T_DIAG_GEN_LOAD_MAX_ES = C_T_DIAG_GEN_LOAD_ES
	Or	
	If	GEN_LOAD < GEN_LOAD_MIN_DIAG_ES
	Then	T_DIAG_GEN_LOAD_MIN_ES is decreased.
	Else	T_DIAG_GEN_LOAD_MIN_ES = C_T_DIAG_GEN_LOAD_ES
	If	GEN_LOAD >= GEN_LOAD_MIN_DIAG_ES
	and	GEN_LOAD <= GEN_LOAD_MAX_DIAG_ES
	Then	T_DIAG_GEN_LOAD_ES is increased
	Else	T_DIAG_GEN_LOAD_ES = 0
Else	T_DIAG_GEN_LOAD_MAX_ES = C_T_DIAG_GEN_LOAD_ES	
and	T_DIAG_GEN_LOAD_MIN_ES = C_T_DIAG_GEN_LOAD_ES	

If	T_DIAG_GEN_LOAD_MAX_ES = 0
Then	ERR_SYM_GEN_LOAD_ES = SYM_0 : SCG
and	the anti-bounce counter is incremented with C_ABC_INC_GEN_LOAD_ES
If	T_DIAG_GEN_LOAD_MIN_ES = 0
Then	ERR_SYM_GEN_LOAD_ES = SYM_1 : SCB or Open Load
and	the anti-bounce counter is incremented with C_ABC_INC_GEN_LOAD_ES

If	T_DIAG_GEN_LOAD_ES >= C_T_DIAG_GEN_LOAD_ES
Then	ERR_SYM_GEN_LOAD_ES = NO_SYM
and	the anti-bounce counter is decremented by 1

If	the anti-bounce counter reaches C_ABC_MAX_GEN_LOAD_ES

Then LV_ERR_GEN_LOAD_ES = 1

If the anti-bounce counter reaches 0

Then LV_ERR_GEN_LOAD_ES = 0

The error bit LV_ERR_GEN_LOAD_ES and the end of diagnosis bit LV_END_DIAG_GEN_LOAD_ES are set by the failure memory.

2. During Driving

If(1) LV_IGK = 1
and LV_ES = 0
and LV_ST_END = 1
and LV_ERR_GEN_LOAD_ES = 0
and VB_MMV < C_VB_MMV_GEN_LOAD_DIAG
and C_N_MIN_GEN_LOAD_DIAG < N < C_N_MAX_GEN_LOAD_DIAG
and TCO > C_TCO_MIN_GEN_LOAD_DIAG
and LV_CDN_VB_OBD1 = 1
Then LV_DIAG_CDN_GEN_LOAD = 1
Else LV_DIAG_CDN_GEN_LOAD = 0
T_DIAG_GEN_LOAD = 0

If LV_CDN_DIAG_GEN_LOAD = 1
and GEN_LOAD_MMV < C_GEN_LOAD_MIN_DIAG
Then T_DIAG_GEN_LOAD_MIN is decreased.
T_DIAG_GEN_LOAD = 0
Else T_DIAG_GEN_LOAD_MIN = C_T_DIAG_GEN_LOAD
T_DIAG_GEN_LOAD is increased

If T_DIAG_GEN_LOAD_MIN = 0
Then ERR_SYM_GEN_LOAD = SYM_1 : SCB or Open Load
and the anti-bounce counter is incremented with C_ABC_INC_GEN_LOAD

If T_DIAG_GEN_LOAD >= C_T_DIAG_GEN_LOAD
Then ERR_SYM_GEN_LOAD = NO_SYM
and the anti-bounce counter is decremented by 1

If the anti-bounce counter reaches C_ABC_MAX_GEN_LOAD

Then LV_ERR_GEN_LOAD = 1

If the anti-bounce counter reaches 0

Then LV_ERR_GEN_LOAD = 0

The error bit LV_ERR_GEN_LOAD and the end of diagnosis bit LV_END_DIAG_GEN_LOAD are set by the failure memory.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_GEN_LOAD_ES	1	0...FFH	0...255	1	-
increment of debounce counter					
C_ABC_MAX_GEN_LOAD_ES	1	1...FFH	1...255	1	-
Max. increment of debounce counter					
C_ABC_INC_GEN_LOAD	1	0...FFH	0...255	1	-
increment of debounce counter					
C_ABC_MAX_GEN_LOAD	1	1...FFH	1...255	1	-
Max. increment of debounce counter					
C_T_DLY_GEN_LOAD	1	0...FFH	0...25.5	0.1	sec
Time delay to start GEN_LOAD diagnosis at ES					
C_GEN_LOAD_MAX_DIAG_ES_BEM	1	0...FFH	0...99.6	0.396	%
Max threshold on GEN_LOAD to detect a failure at ES for VALEO Alternator BEM					
C_GEN_LOAD_MIN_DIAG_ES_BEM	1	0...FFH	0...99.6	0.396	%
Min threshold on GEN_LOAD to detect a failure at ES for VALEO Alternator BEM					
C_GEN_LOAD_MAX_DIAG_ES	1	0...FFH	0...99.6	0.396	%
Max threshold on GEN_LOAD to detect a failure at ES					
C_GEN_LOAD_MIN_DIAG_ES	1	0...FFH	0...99.6	0.396	%
Min threshold on GEN_LOAD to detect a failure at ES					
C_T_DIAG_GEN_LOAD_ES	1	1...FFFFH	0...6553.5	0.1	sec
Diagnosis counter to detect GEN_LOAD error at engine stop					
C_GEN_LOAD_MIN_DIAG	1	0...FFH	0...99.6	0.4	%
Min threshold on GEN_LOAD to detect a failure during engine running					
C_T_DIAG_GEN_LOAD	1	1...FFFFH	0...6553.5	0.1	sec
Diagnosis counter to detect GEN_LOAD error during engine running					
C_VB_MMV_GEN_LOAD_DIAG	1	0...FFH	0...26	0.102	V
Max VB for GEN_LOAD diagnosis					
C_N_MIN_GEN_LOAD_DIAG	1	0...1FE0H	0...8160	1	rpm
Min engine speed for GEN_LOAD diagnosis					
C_N_MAX_GEN_LOAD_DIAG	1	0...1FE0H	0...8160	1	rpm
Max engine speed for GEN_LOAD diagnosis					
C_TCO_MIN_GEN_LOAD_DIAG	1	0...FEH	-48...142.5	0.75	deg.C
Min TCO for GEN_LOAD diagnosis					

A.97.1 Application incidences for generator load signal diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
GEN_LOAD		SYM_0	*STD*
	SCB + OL	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_IGK
GEN_LOAD
GEN_LOAD_MMV
VS

Diagnostic	Symptom Description	Symptom	Filter type
GEN_LOAD_ES	SCG	SYM_0	*STD*
	SCB+OL	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: V_IGK
GEN_LOAD
GEN_LOAD_MMV
VS

A.98 Fuel Tank Level (FTL) input Signal Diagnosis

General information:

The purpose is to diagnose invalid values of the fuel tank level signal coming from the sender.
No fuel level sensor diagnosis in case of engine operating states start (LV_ST) and engine stopped (LV_ES) and CONF_DIAGCP = 0 and CONF_DIAGCP_VOL = 0.

A.98.1 FTL electrical diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FTL_EL	V/O/S	0...1H	0...1	1	-
Boolean for Fuel Tank Level diagnosis (main)					
ERR_SYM_FTL_EL	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom FTL_EL diagnosis (main)					
CTR_DLY_FTL_DIAG	V	0...FFFFH	0...65535	1	-
Inhibition counter for Fuel Tank Level diagnosis					
LV_CDN_DIAG_FTL_EL	V	0...1H	0...1	1	-
Diagnosis condition for electrical diagnosis of FTL (main)					
LV_END_DIAG_FTL_EL	V	0...1H	0...1	1	-
End of electrical diagnosis of FTL (main)					
LV_ERR_FTL_SUB_EL	V/O/S	0...1H	0...1	1	-
Boolean for fuel tank level diagnosis (sub)					
ERR_SYM_FTL_SUB_EL	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected error symptom FTL_EL diagnosis (sub)					
LV_CDN_DIAG_FTL_SUB_EL	V	0...1H	0...1	1	-
Diagnosis condition for electrical diagnosis of FTL (sub)					
LV_END_DIAG_FTL_SUB_EL	V	0...1H	0...1	1	-
End of electrical diagnosis of FTL (sub)					
CTR_DLY_FTL_SUB_DIAG	V	0...FFFFH	0...65535	1	-
Inhibition counter for fuel tank level diagnosis (sub)					

Input data:

LV_IGK	V_FTL	LV_ST	LV_ES
LV_CDN_VB_OBD1	VB	CONF_DIAGCP_VOL	CONF_DIAGCP
V_FTL_SUB	LV_ST_END		

DESCRIPTION:

Error-symtoms are defined to this diagnosis function as following :

Short circuit to Vbatt (=SYM_0)

Short circuit to GND or Open line (=SYM_1)

Remark: Calculation of LV_END_DIAG_FTL_EL and LV_END_DIAG_FTL_SUB_EL see generic calculation "End of diagnosis" in anti bounce algorithm.

Application conditions:

Initialisation: *at new DC*

- LV_ERR_FTL_EL = 0
- LV_CDN_DIAG_FTL_EL= 0
- LV_END_DIAG_FTL_EL = 0
- LV_ERR_FTL_SUB_EL = 0
- LV_CDN_DIAG_FTL_SUB_EL = 0
- LV_END_DIAG_FTL_SUB_EL = 0

```
If    LV_ST_END = 0 -> 1
Then  CTR_DLY_FTL_DIAG = C_CTR_DLY_FTL_DIAG
 and  CTR_DLY_FTL_SUB_DIAG = C_CTR_DLY_FTL_DIAG
endif
```

Recurrence: *0.1 s*

Activation:

The diagnosis is enabled only if LV_IGK = 1 and LV_CDN_VB_OBD1 = 1 and CONF_DIAGCP > 0 and CONF_DIAGCP_VOL > 0 and LV_ST = 0 and LV_ES = 0.

LV_CDN_DIAG_FTL_EL condition:

```
If      LV_IGK = 1
and     LV_ST = 0
and     LV_ES = 0
and     LV_CDN_VB_OBD1 = 1
and     CONF_DIAGCP > 0
then    if    CONF_DIAGCP_VOL = 1    (single FTL sensor)
        and   CTR_DLY_FTL_DIAG = 0
        then  LV_CDN_DIAG_FTL_EL = 1
              LV_CDN_DIAG_FTL_SUB_EL = 0
        else  if    CONF_DIAGCP_VOL = 2        (dual FTL sensors)
              and   CTR_DLY_FTL_SUB_DIAG = 0
              then  LV_CDN_DIAG_FTL_EL = 1
                    LV_CDN_DIAG_FTL_SUB_EL = 1
              endif
        endif
endif
```

Detection:

Due to no clamping voltage with fuel level sensor, electrical diagnosis is performed after the delay counter CTR_DLY_FTL_DIAG has run out with C_CTR_DLY_FTL_DIAG since V_FTL (or V_FTL_SUB) is above(or below) the threshold.

1) main FTL electrical diagnosis

'Short to battery Voltage' :

If $V\_FTL_{(n)}$ > C_V_FTL_MAX_DIAG
and $V\_FTL_{(n-1)}$ <= C_V_FTL_MAX_DIAG
then CTR_DLY_FTL_DIAG has set to C_CTR_DLY_FTL_DIAG
endif

If CTR_DLY_FTL_DIAG has run out
and VB < C_VB_MAX_FTL_DIAG
and $V\_FTL_{(n)}$ > C_V_FTL_MAX_DIAG
then anti-bounce counter is incremented
ERR_SYM_FTL_EL = SYM_0
LV_ERR_FTL_EL = 1 (after debounce)
LV_END_DIAG_FTL_EL = 1 (after debounce)
else **if** V_FTL <= C_V_FTL_MAX_DIAG
and CTR_DLY_FTL_DIAG = 0
then anti-bounce counter is decremented
ERR_SYM_FTL_EL = NO_SYM
LV_ERR_FTL_EL = 0 (after debounce)
LV_END_DIAG_FTL_EL = 1 (after debounce)
endif
endif

'Short to ground or line break' :

If $V\_FTL_{(n)}$ < C_V_FTL_MIN_DIAG
and $V\_FTL_{(n-1)}$ >= C_V_FTL_MIN_DIAG
then CTR_DLY_FTL_DIAG has set to C_CTR_DLY_FTL_DIAG
endif

If CTR_DLY_FTL_DIAG has run out
and $V\_FTL_{(n)}$ < C_V_FTL_MIN_DIAG
then anti-bounce counter is incremented
ERR_SYM_FTL_EL = SYM_1
LV_ERR_FTL_EL = 1 (after debounce)

LV_END_DIAG_FTL_EL = 1 (after debounce)

else **if** V_FTL >= C_V_FTL_MIN_DIAG

and CTR_DLY_FTL_DIAG = 0

then anti-bounce counter is decremented

ERR_SYM_FTL_EL = NO_SYM

LV_ERR_FTL_EL = 0 (after debounce)

LV_END_DIAG_FTL_EL = 1 (after debounce)

endif

endif

2) sub FTL electrical diagnosis

'Short to battery Voltage' :

If $V\_FTL\_SUB_{(n)}$ > C_V_FTL_MAX_DIAG

and $V\_FTL\_SUB_{(n-1)}$ <= C_V_FTL_MAX_DIAG

then CTR_DLY_FTL_SUB_DIAG has set to C_CTR_DLY_FTL_DIAG

endif

If CTR_DLY_FTL_SUB_DIAG has run out

and VB < C_VB_MAX_FTL_DIAG

and $V\_FTL\_SUB_{(n)}$ > C_V_FTL_MAX_DIAG

then anti-bounce counter is incremented

ERR_SYM_FTL_SUB_EL = SYM_0

LV_ERR_FTL_SUB_EL = 1 (after debounce)

LV_END_DIAG_FTL_SUB_EL = 1 (after debounce)

else **if** V_FTL_SUB <= C_V_FTL_MAX_DIAG

and CTR_DLY_FTL_SUB_DIAG = 0

then anti-bounce counter is decremented

ERR_SYM_FTL_SUB_EL = NO_SYM

LV_ERR_FTL_SUB_EL = 0 (after debounce)

LV_END_DIAG_FTL_SUB_EL = 1 (after debounce)

endif

endif

'Short to ground or line break' :

If $V\_FTL\_SUB_{(n)}$ < C_V_FTL_MIN_DIAG

and $V\_FTL\_SUB_{(n-1)}$ >= C_V_FTL_MIN_DIAG

then CTR_DLY_FTL_SUB_DIAG has set to C_CTR_DLY_FTL_DIAG

endif

If CTR_DLY_FTL_SUB_DIAG has run out
and $V\_FTL\_SUB_{(n)}$ < C_V_FTL_MIN_DIAG
then anti-bounce counter is incremented
ERR_SYM_FTL_SUB_EL = SYM_1
LV_ERR_FTL_SUB_EL = 1 (after debounce)
LV_END_DIAG_FTL_SUB_EL = 1 (after debounce)
else **if** V_FTL_SUB >= C_V_FTL_MIN_DIAG
and CTR_DLY_FTL_SUB_DIAG = 0
then anti-bounce counter is decremented
ERR_SYM_FTL_SUB_EL = NO_SYM
LV_ERR_FTL_SUB_EL = 0 (after debounce)
LV_END_DIAG_FTL_SUB_EL = 1 (after debounce)
endif
endif

- In case of error detection the anti-bounce counter is incremented with C_ABC_INC_FTL_DIAG.
- If the anti-bounce counter reaches C_ABC_MAX_FTL_DIAG then LV_ERR_FTL_EL = 1 or LV_ERR_FTL_SUB_EL = 1.
- If no symptom is active, anti bounce counter is decremented.

Anti-bounce counter increment : C_ABC_INC_FTL_DIAG
Maximum value of anti-bounce counter : C_ABC_MAX_FTL_DIAG

Emergency operation **:**

* Limp home:

1 . FTL = C_FTL_LIH (recommended value 50%)
2. Evap monitoring reference volume calculation will be done with DTP gradient volume estimation after LV_ERR_FTL_EL = 1 or LV_ERR_FTL_SUB_EL = 1

Calibration data:

Name	Dim.	Hex. Limits	Phys. Limits	Resol.	Unit
C_V_FTL_MIN_DIAG	1	0…FFH	0...4.985	0.0195	V
Min diagnostic threshold for FTL signal range diagnosis					
C_V_FTL_MAX_DIAG	1	0...FFH	0...4.985	0.0195	V
Max diagnostic threshold for FTL signal range diagnosis					
C_VB_MAX_FTL_DIAG	1	0...FFH	0...25.8984	26/256	V
Max VB threshold for FTL signal diag. for short circuit to battery					
C_ABC_INC_FTL_DIAG	1	0...FFH	0...255	1	-
Anti – bounce counter increment for FTL signal range diagnosis					
C_ABC_MAX_FTL_DIAG	1	0...FFH	0...255	1	-
Maximum value of anti-bounce counter for FTL signal range diagnosis					
C_FTL_LIH	1	0...FFH	0...100	0.39	%
Lim home value of FTL					
C_CTR_DLY_FTL_DIAG	1	0...FFFFH	0...65535	1	-
Minimum delay counter for FTL signal electrical check (recurrence 0.1 sec.)					

A.98.2 FTL Plausibility diagnosis – Stuck

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FTL_STUCK	V/O	0...1H	0...1	1	-
Boolean for stuck error currently present on FTL					
LV_CDN_DIAG_FTL_STUCK	V	0...1H	0...1	1	-
Boolean for stuck error condition of Fuel Tank Level signal (stuck)					
LV_FTL_STUCK_DIAG_ACT	V	0...1H	0...1	1	-
Boolean for FTL stuck diagnosis activated					
LV_END_DIAG_FTL_STUCK	V	0...1H	0...1	1	-
Boolean for FTL stuck diagnosis finished					
ERR_SYM_FTL_STUCK	V/O	0H 1H	NO_SYM SYM_0	1	-
Detected error symptom for FTL_STUCK diagnosis					
FTL_MMV_MV_PER	V	0...FFFFH	0...100	0.0015	%
Average FTL_MMV value during C_T_DIF_FTL_MMV					
FTL_MMV_REF_INI	V/S/O	0...FFFFH	0...100	0.0015	%
Reference FTL_MMV at FTL stuck diagnosis start					
FTL_MMV_POS_DIAG	V/S/O	0...FFFFH	0...100	0.0015	%
FTL_MMV value toward positive direction					
FTL_MMV_NEG_DIAG	V/S/O	0...FFFFH	0...100	0.0015	%
FTL_MMV value toward negative direction					
FTL_MMV_DIF_POS_DIAG	V/O	0...FFFFH	0...100	0.0015	%
Difference between positive measured and reference FTL_MMV value					
FTL_MMV_DIF_NEG_DIAG	V/O	0...FFFFH	0...-100	0.0015	%
Difference between negative measured and reference FTL_MMV value					
CTR_FCO_FTL_STUCK	V/S/O	0…FFFFFFFFH	0…549763678,2	0.128	uL
Integrated fuel consumption for FTL_STUCK and FTL_STUCK_H					
LV_ERR_FTL_SUB_STUCK	V/O	0...1H	0...1	1	-
Boolean for stuck error currently present on FTL (sub sensor)					
LV_END_DIAG_FTL_SUB_STUCK	V	0...1H	0...1	1	-
Boolean for FTL stuck diagnosis finished (sub sensor)					
ERR_SYM_FTL_SUB_STUCK	V/O	0H 1H	NO_SYM SYM_0	1	-
Detected error symptom for FTL_STUCK diagnosis (sub sensor)					
V_FTL_SUB_POS_DIAG	V/S/O	0...FFH	0...4.985	0.0195	Volt
V_FTL_SUB value toward positive direction (sub sensor)					
V_FTL_SUB_NEG_DIAG	V/S/O	0...FFH	0...4.985	0.0195	Volt
V_FTL_SUB value toward negative direction (sub sensor)					
V_FTL_SUB_DIF_POS_DIAG	V/O	0...FFH	0...4.985	0.0195	Volt
Difference between positive measured and reference V_FTL_SUB value (sub sensor)					
V_FTL_SUB_DIF_NEG_DIAG	V/O	0...FFH	0...-4.985	0.0195	Volt
Difference between negative measured and reference V_FTL_SUB value (sub sensor)					
V_FTL_SUB_REF_INI	V/S/O	0...FFH	0...4.985	0.0195	Volt
Reference V_FTL_SUB_MMV at FTL_SUB stuck diagnosis start (sub sensor)					
CTR_FCO_FTL_SUB_STUCK	V/S/O	0…FFFFFFFFH	0…549763678,2	0.128	uL
Integrated fuel consumption for FTL_SUB_STUCK (sub sensor)					
LV_FTL_SUB_STUCK_DIAG_ACT	V	0...1H	0...1	1	-
Boolean for sub FTL stuck diagnosis activated (sub sensor)					
LV_CDN_DIAG_FTL_SUB_STUCK	V	0...1H	0...1	1	-
Boolean for stuck error condition of Fuel Tank Level signal (sub sensor stuck)					

Input data:

LV_IGK	LV_ERR_FTL_EL	FTL_MMV	LV_ERR_VS
CONF_DIAGCP_VOL	LV_ST	LV_CDN_VB_OBD2	LV_ERR_MWSS
LV_END_DIAG_FTL_STUCK_H	LV_ES	C_FTL_LIH	LV_ERR_FTL_INTM
LV_FTL_STUCK_H_DIAG_ACT	LV_ERR_FTL	FCO_SUM	LV_ERR_FTL_STUCK_H
LC_ERR_FMY_CLR	CONF_DIAGCP	V_FTL_SUB_MMV	LV_ERR_FTL_SUB_EL
LV_ERR_FTL_SUB_INTM	LV_INH_DIAG_FTL_STUCK		

Description:

The purpose is to diagnose the fuel tank level signal from fuel level sender.

The fuel level signal stuck is detected when the filtered FTL signal (FTL_MMV) is not changed during accumulated mileage with engine running. The mileage is accumulated regardless driving cycle and an error is detected when the mileage counter reaches to calibrated constant.

The diagnosis takes place only if LV_IGK = 1 and LV_CDN_VB_OBD2 = 1 and CONF_DIAGCP > 0 and CONF_DIAGCP_VOL > 0.

Error-symptoms are defined to this diagnosis function as following :

FTL signal stuck (=SYM_0)

FTL_SUB signal stuck (=SYM_0)

Remark : Calculation of LV_END_DIAG_FTL_STUCK, LV_END_DIAG_FTL_SUB_STUCK, LV_CDN_DIAG_FTL_STUCK and LV_CDN_DIAG_FTL_SUB_STUCK

see generic calculation "End of diagnosis" in anti bounce algorithm.

Application conditions:

Initialisation: *at new DC*

- LV_CDN_DIAG_FTL_STUCK = 0
- LV_CDN_DIAG_FTL_SUB_STUCK = 0
- LV_END_DIAG_FTL_STUCK = 0
- LV_ERR_FTL_STUCK = 0
- LV_FTL_STUCK_DIAG_ACT = 0
- FTL_MMV_DIF_POS_DIAG = 0
- FTL_MMV_DIF_NEG_DIAG = 0
- LV_END_DIAG_FTL_SUB_STUCK = 0
- LV_ERR_FTL_SUB_STUCK = 0
- LV_FTL_SUB_STUCK_DIAG_ACT = 0
- V_FTL_SUB_DIF_POS_DIAG = 0
- V_FTL_SUB_DIF_NEG_DIAG = 0

```
        If      Loss of NVMY or NVMY reset
                or failure memory reset (LC_ERR_FMY_CLR)
        then    FTL_MMV_REF_INI               = 0
                FTL_MMV_POS_DIAG              = 0
                FTL_MMV_NEG_DIAG              = 0
                CTR_FCO_FTL_STUCK             = 0
                FTL_MMV_DIF_POS_DIAG          = 0
                FTL_MMV_DIF_NEG_DIAG          = 0
                LV_FTL_STUCK_DIAG_ACT         = 0
                LV_END_DIAG_FTL_STUCK         = 0
                V_FTL_SUB_POS_DIAG            = 0
                V_FTL_SUB_NEG_DIAG            = 0
                V_FTL_SUB_DIF_POS_DIAG        = 0
                V_FTL_SUB_DIF_NEG_DIAG        = 0
                CTR_FCO_FTL_SUB_STUCK         = 0
```

Recurrence : 0.1 sec.

Activation:

LV_CDN_DIAG_FTL_STUCK, LV_CDN_DIAG_FTL_SUB_STUCK condition:

```
If          LV_IGK = 1
     and    LV_ST = 0
     and    LV_ES = 0
     and    LV_INH_DIAG_FTL_STUCK = 0
     and    LV_CDN_VB_OBD2 = 1
     and    CONF_DIAGCP > 0
     and    C_T_DLY_FTL_STUCK_DIAG is expired since engine start(ST)
then        if      CONF_DIAGCP_VOL = 1     (single FTL)
            then    LV_CDN_DIAG_FTL_STUCK = 1
            else    if      CONF_DIAGCP_VOL = 2          (dual FTL)
                    then    LV_CDN_DIAG_FTL_STUCK = 1
                            LV_CDN_DIAG_FTL_SUB_STUCK = 1
                    endif
            endif
else        LV_CDN_DIAG FTL_STUCK= 0
            LV_CDN_DIAG_FTL_SUB_STUCK = 0
endif
```

Formula section:

FTL_MMV_REF_INI calculation:

*Activation condition :

if LV_CDN_DIAG_FTL_STUCK = 1
and LV_FTL_STUCK_DIAG_ACT = 0
and LV_FTL_SUB_STUCK_DIAG_ACT = 0
and LV_FTL_STUCK_H_DIAG_ACT = 0
and LV_END_DIAG_FTL_STUCK = 0
and LV_END_DIAG_FTL_STUCK_H = 0
and CTR_FCO_FTL_STUCK = 0
and CTR_FCO_FTL_SUB_STUCK = 0
then FTL_MMV_REF_INI calculation

* FTL_MMV_REF_INI calculation :

1. SUM (FTL_MMV i) = Sum of all FTL_MMV values during C_T_DIF_FTL_MMV
2. FTL_MMV_MV_PER value is mean value during C_T_DIF_FTL_MMV and SUM(FTL_MMVi) is calculated every **0.1 sec**.
   FTL_MMV_MV_PER = [SUM (FTL_MMV i)] / (C_T_DIF_FTL_MMV / 0.1 [s])
3. The reference FTL initial value is average value during C_NR_FTL_REF_INI.

$$\mathrm{FTL\_MMV\_REF\_INI} = \frac{\sum_{0}^{\mathrm{C\_NR\_FTL\_REF\_INI}} \mathrm{FTL\_MMV\_MV\_PER}}{\mathrm{C\_NR\_FTL\_REF\_INI} + 1}$$

If there is an interruption during FTL_MMV_REF_INI calculation, this calculation routine is repeated again.

If	FTL_MMV_REF_INI calculation is finished (C_NR_FTL_REF_INI is passed)		
or	Delay time (= C_T_DIF_FTL_MMV * C_NR_FTL_REF_INI) has passed since LV_CDN_DIAG_FTL_STUCK = 1		
then	LV_FTL_STUCK_DIAG_ACT = 1		
	FTL_MMV_POS_DIAG = FTL_MMV_REF_INI		
	FTL_MMV_NEG_DIAG = FTL_MMV_REF_INI		
	CTR_FCO_FTL_STUCK = 0		
endif			

If	FTL_MMV_REF_INI calculation is finished (C_NR_FTL_REF_INI is passed)
or	Delay time (= C_T_DIF_FTL_MMV * C_NR_FTL_REF_INI) has passed since LV_CDN_DIAG_FTL_SUB_STUCK = 1
then	LV_FTL_SUB_STUCK_DIAG_ACT = 1
	V_FTL_SUB_REF_INI = V_FTL_SUB_MMV
	V_FTL_SUB_POS_DIAG = V_FTL_SUB_MMV
	V_FTL_SUB_NEG_DIAG = V_FTL_SUB_MMV
	CTR_FCO_FTL_SUB_STUCK = 0
endif	

At the end of DC, the FTL_MMV_REF_INI & V_FTL_SUB_REF_INI are saved in NVMY.

FTL_MMV_DIF_POS(NEG)_DIAG calculation:

An error (fuel sender stuck) is detected if the average fuel level FTL_MMV does not change (up ⇒ tank fill-up, down ⇒ fuel is consumed) for a certain mileage with the engine running.

1) main FTL

If	LV_FTL_STUCK_DIAG_ACT = 1		
or	LV_FTL_STUCK_H_DIAG_ACT = 1		
then	**if**	LV_ERR_FTL = 0	
	then	**if**	FTL_MMV < FTL_MMV_NEG_DIAG
		then	FTL_MMV_NEG_DIAG = FTL_MMV
		endif	
		if	FTL_MMV > FTL_MMV_POS_DIAG
		then	FTL_MMV_POS_DIAG = FTL_MMV
		endif	
	else	$FTL\_MMV\_NEG\_DIAG_{(n)} = FTL\_MMV\_NEG\_DIAG_{(n-1)}$	
		$FTL\_MMV\_POS\_DIAG_{(n)} = FTL\_MMV\_POS\_DIAG_{(n-1)}$	

```
            endif
            FTL_MMV_DIF_POS_DIAG = FTL_MMV_POS_DIAG – FTL_MMV_REF_INI
            FTL_MMV_DIF_NEG_DIAG = FTL_MMV_NEG_DIAG – FTL_MMV_REF_INI
endif
```

2) sub FTL

```
If      LV_FTL_SUB_STUCK_DIAG_ACT = 1
then    if      LV_ERR_FTL = 0
        then    if      V_FTL_SUB_MMV < V_FTL_SUB_NEG_DIAG
                then    V_FTL_SUB_NEG_DIAG = V_FTL_SUB_MMV
                endif
                if      V_FTL_SUB_MMV > V_FTL_SUB_POS_DIAG
                then    V_FTL_SUB_POS_DIAG = V_FTL_SUB_MMV
                endif
        else    V_FTL_SUB_NEG_DIAG(n) = V_FTL_SUB_NEG_DIAG(n-1)
                V_FTL_SUB_POS_DIAG(n) = V_FTL_SUB_POS_DIAG(n-1)
        endif
        V_FTL_SUB_DIF_POS_DIAG = V_FTL_SUB_POS_DIAG – V_FTL_SUB_REF_INI
        V_FTL_SUB_DIF_NEG_DIAG = V_FTL_SUB_NEG_DIAG – V_FTL_SUB_REF_INI
endif
```

If the end of driving cycle, FTL_MMV_POS(NEG)_DIAG & V_FTL_SUB_POS(NEG) are stored in the NVMY.

CTR_FCO_FTL_STUCK calculation

1) main FTL fuel consumption

```
If      (       LV_FTL_STUCK_DIAG_ACT = 1
        or      LV_FTL_STUCK_H_DIAG_ACT = 1   )
and     LV_END_DIAG_FTL_STUCK = 0
and     LV_END_DIAG_FTL_STUCK_H = 0
then    CTR_FCO_FTL_STUCK(n) = CTR_FCO_FTL_STUCK(n-1)
                        + (FCO_SUM(n) -  FCO_SUM(n-1) )
else    CTR_FCO_FTL_STUCK = 0
endif
```

2) sub FTL fuel consumption

```
If      CONF_DIAGCP_VOL = 2     (dual FTL)
and     LV_FTL_SUB_STUCK_DIAG_ACT = 1
and     LV_END_DIAG_FTL_SUB_STUCK = 0
```

then $CTR\_FCO\_FTL\_SUB\_STUCK_n = CTR\_FCO\_FTL\_SUB\_STUCK_{n-1} + (FCO\_SUM_n - FCO\_SUM_{n-1})$

else CTR_FCO_FTL_SUB_STUCK = 0

endif

The CTR_FCO_FTL_STUCK & CTR_FCO_FTL_SUB_STUCK are saved in NVMY at the end of driving cycle.

Detection:

1) main FTL stuck diagnosis

```
If(1)     LV_END_DIAG_FTL_STUCK = 0
then
   If(2)        FTL_MMV_REF_INI <= C_FTL_MMV_REF_INI_DIAG
   then   if    [ (FTL_MMV_DIF_POS_DIAG > C_FTL_MMV_DIF_POS_DIAG)
                or (FTL_MMV_DIF_NEG_DIAG < C_FTL_MMV_DIF_NEG_DIAG) ]
          then  LV_ERR_FTL_STUCK = 0          (FTL Signal Stuck Failure healed)
                ERR_SYM_FTL_STUCK = NO_SYM
                LV_END_DIAG_FTL_STUCK = 1
                LV_FTL_STUCK_DIAG_ACT = 0
                CTR_FCO_FTL_STUCK = 0
          else  If     CTR_FCO_FTL_STUCK >= C_CTR_FCO_MAX_FTL_STUCK
                then   LV_ERR_FTL_STUCK = 1   (FTL Signal Stuck Failure detected)
                       LV_END_DIAG_FTL_STUCK = 1
                       LV_FTL_STUCK_DIAG_ACT = 0
                       ERR_SYM_FTL_STUCK = SYM_0
                       CTR_FCO_FTL_STUCK = 0
                else   LV_FTL_STUCK_DIAG_ACT = 1
                endif
          endif
   else(2) if   LV_ERR_FTL_STUCK_H = 0
          and   LV_END_DIAG_FTL_STUCK_H = 1        (no high stuck error)
          then  LV_ERR_FTL_STUCK = 0               (FTL stuck failure healed)
                ERR_SYM_FTL_STUCK = NO_SYM
                LV_END_DIAG_FTL_STUCK = 1
          endif
   endif(2)
endif(1)
```

2) sub FTL stuck diagnosis

```
If(1)       LV_END_DIAG_FTL_SUB_STUCK = 0
then
    If(2)          FTL_MMV_REF_INI <= C_FTL_MMV_REF_INI_DIAG
    then  if       [ (V_FTL_SUB_DIF_POS_DIAG > C_V_FTL_SUB_DIF_POS_DIAG)
                   or (V_FTL_SUB_DIF_NEG_DIAG < C_V_FTL_SUB_DIF_NEG_DIAG) ]
          then     LV_ERR_FTL_SUB_STUCK = 0      (FTL Signal Stuck Failure healed)
                   ERR_SYM_FTL_SUB_STUCK = NO_SYM
                   LV_END_DIAG_FTL_SUB_STUCK = 1
                   LV_FTL_SUB_STUCK_DIAG_ACT = 0
                   CTR_FCO_FTL_SUB_STUCK = 0
          else     If      CTR_FCO_FTL_SUB_STUCK >= C_CTR_FCO_MAX_FTL_STUCK
                   then    LV_ERR_FTL_SUB_STUCK = 1
                           (FTL Signal Stuck Failure detected)
                           LV_END_DIAG_FTL_SUB_STUCK = 1
                           LV_FTL_SUB_STUCK_DIAG_ACT = 0
                           ERR_SYM_FTL_SUB_STUCK = SYM_0
                           CTR_FCO_FTL_SUB_STUCK = 0
                   else    LV_FTL_SUB_STUCK_DIAG_ACT = 1
                   endif
          endif
    endif(2)
endif(1)
```

Emergency operation **:**

* Limp home:

1. FTL = C_FTL_LIH (recommended value 50%)
2. Evap monitoring reference volume calculation will be done with DTP gradient volume estimation after LV_ERR_FTL_STUCK = 1 or LV_ERR_FTL_SUB_STUCK = 1

Calibration data:

Name	Dim.	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_FTL_STUCK_DIAG	1	0...FFH	0...255	1	s
Time delay of FTL_STUCK diagnosis after engine start					
C_T_DIF_FTL_MMV	1	0...FFH	0...25.5	0.1	s
Time duration for FTL_MMV_MV_PER calculation					
C_NR_FTL_REF_INI	1	0...FFH	0...255	1	-
Number of FTL_MMV_MV_PER to determine FTL_MMV_REF_INI					
C_FTL_MMV_DIF_POS_DIAG	1	0...FFFFH	0...100	0.0015	%
Threshold for FTL stuck error detection					
C_FTL_MMV_DIF_NEG_DIAG	1	0...FFFFH	0...-100	0.0015	%
Threshold for FTL stuck error detection					
C_FTL_MMV_REF_INI_DIAG	1	0...FFFFH	0...100	0.0015	%
Threshold for FTL stuck & stuck high error detection					
C_CTR_FCO_MAX_FTL_STUCK	1	0...FFFFFFFFH	0...549763678.2	10.128	uL
Maximum counter for consumed fuel for FTL stuck detection					
C_V_FTL_SUB_DIF_POS_DIAG	1	0...FFH	0...4.985	0.0195	Volt
Threshold for sub V_FTL stuck error detection					
C_V_FTL_SUB_DIF_NEG_DIAG	1	0...FFH	0...-4.985	0.0195	Volt
Threshold for sub V_FTL stuck error detection					

A.98.3 FTL Plausibility diagnosis – Stuck high

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FTL_STUCK_H	V/O	0...1H	0...1	1	-
Boolean for stuck high error currently present on FTL					
LV_END_DIAG_FTL_STUCK_H	V	0...1H	0...1	1	-
Boolean for FTL stuck diagnosis finished					
ERR_SYM_FTL_STUCK_H	V/O	0H 2H	NO_SYM SYM_1	1	-
Detected error symptom for FTL_STUCK_H diagnosis					
LV_FTL_STUCK_H_DIAG_ACT	V	0...1H	0...1	1	-
Boolean for FTL stuck high diagnosis activated					

Input data:

	FTL_MMV_DIF_POS_DIAG	FTL_MMV_REF_INI	FTL_MMV
CTR_FCO_FTL_STUCK	FTL_MMV_DIF_NEG_DIAG	FTL_MMV_MV_PER	C_FTL_LIH
C_CTR_FCO_MAX_FTL_STUCK	C_FTL_MMV_REF_INI_DIAG	C_T_DIF_FTL_MMV	C_NR_FTL_REF_INI
LV_ERR_FTL_STUCK	LV_END_DIAG_FTL_STUCK	LC_ERR_FMY_CLR	CONF_DIAGCP
CONF_DIAGCP_VOL			

Description:

The purpose is to diagnose the fuel tank level signal from fuel level sender.

The fuel level signal stuck is detected when the filtered FTL signal (FTL_MMV) is not changed during accumulated mileage with engine running. The mileage is accumulated regardless driving cycle and an error is detected when the mileage counter reaches to calibrated constant.

The diagnosis takes place only if LV_IGK = 1 and LV_CDN_VB_OBD2 = 1 and CONF_DIAGCP > 0 and CONF_DIAGCP_VOL > 0.

Error-symptoms are defined to this diagnosis function as following :

FTL signal stuck high (=SYM_1)

Remark : Calculation of LV_END_DIAG_FTL_STUCK_H see generic calculation "End of diagnosis" in anti bounce algorithm.

Application conditions:

Initialisation: *at new DC*

- LV_END_DIAG_FTL_STUCK_H = 0
- LV_FTL_STUCK_H_DIAG_ACT = 0
- LV_ERR_FTL_STUCK_H = 0

If Loss of NVMY or NVMY reset
or failure memory reset (LC_ERR_FMY_CLR)
then LV_FTL_STUCK_H_DIAG_ACT = 0
LV_END_DIAG_FTL_STUCK_H = 0
endif

Recurrence : 0.1 sec.

Activation:

LV_CDN_DIAG_FTL_STUCK condition : same with "FTL plausibility diagnosis – stuck"

Formula section:

```
If      FTL_MMV_REF_INI calculation is finished (C_NR_FTL_REF_INI is passed)
   or   Delay time (= C_T_DIF_FTL_MMV * C_NR_FTL_REF_INI ) has passed since
        LV_CDN_DIAG_FTL_STUCK = 1
then    LV_FTL_STUCK_H_DIAG_ACT = 1
endif
```

Detection:

```
If(1)  LV_END_DIAG_FTL_STUCK_H = 0
then
   If(2)   FTL_MMV_REF_INI > C_FTL_MMV_REF_INI_DIAG
           If     [ (FTL_MMV_DIF_POS_DIAG > C_FTL_MMV_DIF_POS_DIAG_H)
                  or (FTL_MMV_DIF_NEG_DIAG < C_FTL_MMV_DIF_NEG_DIAG_H) ]
           then   LV_ERR_FTL_STUCK_H = 0        (FTL signal stuck high failure healed)
                  ERR_SYM_FTL_STUCK_H = NO_SYM
                  LV_END_DIAG_FTL_STUCK_H = 1
                  LV_FTL_STUCK_H_DIAG_ACT = 0
           else   If     CTR_FCO_FTL_STUCK >= C_CTR_FCO_MAX_FTL_STUCK
                  then   LV_ERR_FTL_STUCK_H = 1 (FTL signal stuck high failure detected)
                         LV_END_DIAG_FTL_STUCK_H = 1
                         LV_FTL_STUCK_H_DIAG_ACT = 0
                         ERR_SYM_FTL_STUCK_H = SYM_1
                  else   LV_FTL_STUCK_H_DIAG_ACT = 1
                  endif
           endif
   else(2) if     LV_ERR_FTL_STUCK = 0
           and    LV_END_DIAG_FTL_STUCK = 1          (no stuck error)
           then   LV_ERR_FTL_STUCK_H = 0             (FTL stuck high failure healed)
                  ERR_SYM_FTL_STUCK_H = NO_SYM
                  LV_END_DIAG_FTL_STUCK = 1
           endif
   endif(2)
endif (1)
```

Emergency operation **:**

* Limp home:

1. FTL = C_FTL_LIH (recommended value 50%)
2. Evap monitoring reference volume calculation will be done with DTP gradient volume estimation after LV_ERR_FTL_STUCK_H = 1

Calibration data:

Name	Dim.	Hex. limits	Phys. limits	Resol.	Unit
C_FTL_MMV_DIF_POS_DIAG_H	1	0...FFFFH	0...100	0.0015	%
Threshold for FTL stuck high detection					
C_FTL_MMV_DIF_NEG_DIAG_H	1	0...FFFFH	0...-100	0.0015	%
Threshold for FTL stuck high detection					

A.98.4 Plausibility component diagnosis - Intermittent

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FTL_INTM	V/S	0...01H	0...1	1	-
Boolean for error currently present on FTL plausibility -intermittent					
LV_CDN_DIAG_FTL_INTM	V	0...01H	0...1	1	-
Boolean for activation condition on FTL plausibility -intermittent					
ERR_SYM_FTL_INTM	V/O	0H 1H	NO_SYM SYM_0	1	
Detected error symptom FTL_INTM diagnosis					
CTR_DLY_WHEEL_GRD	V	0...FFH	0...25.5	0.1	s
Delay time after wheel speed gradient above the threshold					
LV_END_DIAG_FTL_INTM	V	0...1H	0...1	1	-
Boolean for FTL intermittent diagnosis finished					
LV_ERR_FTL_SUB_INTM	V/S	0...01H	0...1	1	-
Boolean for error currently present on sub FTL plausibility –intermittent (sub sensor)					
LV_CDN_DIAG_FTL_SUB_INTM	V	0...01H	0...1	1	-
Boolean for activation condition on sub FTL plausibility –intermittent (sub sensor)					
ERR_SYM_FTL_SUB_INTM	V/O	0H 1H	NO_SYM SYM_0	1	
Detected error symptom sub FTL_INTM diagnosis (sub sensor)					
LV_END_DIAG_FTL_SUB_INTM	V	0...1H	0...1	1	-
Boolean for sub FTL intermittent diagnosis finished (sub sensor)					

Input data:

FTL_MES	LV_IGK	LV_CDN_VB_OBD2	LV_LDC_CAT_1
LV_ERR_FTL_EL	LV_ERR_FTL_STUCK	WHEEL_GRD_MMV	CONF_DIAGCP_VOL
LV_ERR_FTL_STUCK_H	C_FTL_LIH	LV_PL	VS
FTL_MMV	LV_ERR_MWSS	T_AST	CONF_DIAGCP
V_FTL_SUB	V_FTL_SUB_MMV	LV_ERR_FTL_SUB_EL	LV_ERR_VS
LV_ERR_FTL_SUB_STUCK	LV_END_DIAG_FTL_STUCK	LV_END_DIAG_FTL_SUB_STUCK	

FUNCTION DESCRIPTION:

General information:

The purpose of this diagnosis is to detect an implausible gradient on the FTL_MES or V_FTL_SUB signal.

The diagnostic function checks wether the difference between one measured fuel tank level value FTL_MES and the succeeding value is too big. To avoid mis-detection of FTL signal-jump by fuel sloshing, limited dynamic conditions must be fulfilled and sudden vehicle movement should be avoided.

The diagnosis is continuously active if LV_IGK = 1 and LV_CDN_VB_OBD2 = 1 and CONF_DIAGCP > 0 and CONF_DIAGCP_VOL > 0.

Remark : Calculation of LV_END_DIAG_FTL_INTM or LV_END_DIAG_FTL_SUB_INTM see generic calculation "End of diagnosis" in anti bounce algorithm.

Application conditions:

Initialization : *at new DC*
- LV_ERR_FTL_INTM = 0
- LV_CDN_DIAG_FTL_INTM = 0
- LV_END_DIAG_FTL_INTM = 0
- LV_ERR_FTL_SUB_INTM = 0
- LV_CDN_DIAG_FTL_SUB_INTM = 0
- LV_END_DIAG_FTL_SUB_INTM = 0
- CTR_DLY_WHEEL_GRD = 0 (at LV_IGK = 0 -> 1)

Recurrence : 0.1 sec.

Activation :

```
If          LV_IGK = 1
      and   LV_PL = 1
      and   LV_ERR_FTL_EL = 0
      and   LV_ERR_FTL_SUB_EL = 0
      and   LV_ERR_FTL_STUCK = 0
      and   LV_ERR_FTL_SUB_STUCK = 0
      and   LV_ERR_FTL_STUCK_H = 0
      and   LV_ERR_MWSS = 0
      and   LV_ERR_VS = 0
      and   LV_CDN_VB_OBD2 = 1
      and   CONF_DIAGCP > 0
      and   T_AST > C_T_AST_FTL_INTM
      and   VS >= C_VS_MIN_FTL_INTM
      and   VS <= C_VS_MAX_FTL_INTM
      and   CTR_DLY_WHEEL_GRD = 0
      and   LV_LDC_CAT_1 = 1          (Limited dynamic is fulfilled)
      and   LV_END_DIAG_FTL_STUCK = 1
then
      if    CONF_DIAGCP_VOL = 1                          (single FTL)
      then  LV_CDN_DIAG_FTL_INTM = 1
            LV_CDN_DIAG_FTL_SUB_INTM = 0
      else  if    CONF_DIAGCP_VOL = 2                    (dual FTL)
            and   LV_END_DIAG_FTL_SUB_STUCK = 1
            then  LV_CDN_DIAG_FTL_INTM = 1
                  LV_CDN_DIAG_FTL_SUB_INTM = 1
            endif
      endif
else  LV_CDN_DIAG_FTL_INTM = 0
```

LV_CDN_DIAG_FTL_SUB_INTM = 0

endif

CTR_DLY_WHEEL_GRD calculation.

If WHEEL_GRD_MMV >= C_WHEEL_GRD_MMV_FTL_INTM
or LV_END_DIAG_FTL_STUCK = 0
then CTR_DLY_WHEEL_GRD = C_CTR_DLY_WHEEL_GRD
else CTR_DLY_WHEEL_GRD is decrement by zero
endif

Formula section:

1) main FTL noise diagnosis

If | FTL_MMV_{n-1} – FTL_MES_{n} | > C_FTL_MES_GRD_MAX_DIAG
then increment the anti-bounce counter by C_ABC_INC_FTL_INTM_DIAG
  if the anti-bounce counter reached C_ABC_MAX_FTL_INTM_DIAG
  then LV_ERR_FTL_INTM = 1 *(FTL Signal Intermittent Failure detected)*
    ERR_SYM_FTL_INTM = SYM_0
    LV_END_DIAG_FTL_INTM = 1
  endif
else decrement the anti-bounce counter
  if the anti-bounce counter reached 0
  then LV_ERR_FTL_INTM = 0 *(FTL Signal Intermittent Failure healed)*
    ERR_SYM_FTL_INTM = NO_SYM
    LV_END_DIAG_FTL_INTM = 1
  endif
endif

2) sub FTL noise diagnosis

If | $V\_FTL\_SUB\_MMV_{n-1}$ – $V\_FTL\_SUB_{n}$ | > C_V_FTL_MES_GRD_MAX_DIAG
then increment the anti-bounce counter by C_ABC_INC_FTL_INTM_DIAG
  if the anti-bounce counter reached C_ABC_MAX_FTL_INTM_DIAG
  then LV_ERR_FTL_SUB_INTM = 1 *(FTL Signal Intermittent Failure detected)*
    ERR_SYM_FTL_SUB_INTM = SYM_0
    LV_END_DIAG_FTL_SUB_INTM = 1
  endif
else decrement the anti-bounce counter
  if the anti-bounce counter reached 0
  then LV_ERR_FTL_SUB_INTM = 0 *(FTL Signal Intermittent Failure healed)*
    ERR_SYM_FTL_SUB_INTM = NO_SYM
    LV_END_DIAG_FTL_SUB_INTM = 1
  endif
endif

Anti-bounce counter increment: C_ABC_INC_FTL_INTM_DIAG

Maximum value of anti-bounce counter: C_ABC_MAX_FTL_INTM_DIAG

Emergency operation :

* Limp home:

1. FTL = C_FTL_LIH (recommended value 50%)
2. Evap monitoring reference volume calculation will be done with DTP gradient volume estimation after LV_ERR_FTL_INTM = 1 or LV_ERR_FTL_SUB_INTM = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FTL_MES_GRD_MAX_DIAG	1	0...FFH	0...100	0,39	%
Maximum allowed gradient for the fuel tank level signal					
C_VS_MIN_FTL_INTM	1	0...FFH	0...255	1	Km/h
Minimum vehicle speed for FTL signal intermittent diagnosis.					
C_CTR_DLY_WHEEL_GRD	1	0...FFH	0...25.5	0.1	S
Delay time after wheel speed gradient above the threshold					
C_WHEEL_GRD_MMV_FTL_INTM	1	0...FFFFH	0...999.85	0.01526	‰
Minimum wheel speed gradient for FTL signal intermittent diagnosis.					
C_ABC_INC_FTL_INTM_DIAG	1	0...FFH	0...255	1	-
Anti-bounce counter increment for FTL signal gradient diagnosis.					
C_ABC_MAX_FTL_INTM_DIAG	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for FTL signal gradient diagnosis.					
C_T_AST_FTL_INTM	1	1...FFFFH	0...6553.5	0.1	s
FTL signal stabilization time after start and delay time before FTL intermittent diagnosis activation					
C_V_FTL_MES_GRD_MAX_DIAG	1	0...FFH	0...4.985	0.195	Volt
Maximum allowed gradient for the sub fuel tank level signal (sub sensor)					
C_VS_MAX_FTL_INTM	1	0...FFH	0...255	1	Km/h
Maximum vehicle speed for FTL signal intermittent diagnosis.					

A.98.5 Global error flag

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FTL	V/O	0...1H	0...1	1	-
Boolean for error currently present on FTL sensor (EL , STUCK , INTM)					

Input data:

LV_ERR_FTL_EL	LV_ERR_FTL_STUCK	LV_ERR_FTL_INTM	LV_ERR_FTL_STUCK_H
LV_ERR_FTL_SUB_EL	LV_ERR_FTL_SUB_STUCK	LV_ERR_FTL_SUB_INTM	

Application conditions:

Initialization : *at new DC* LV_ERR_FTL = 0

Formula section:

If LV_ERR_FTL_EL = 1
or LV_ERR_FTL_SUB_EL = 1
or LV_ERR_FTL_STUCK = 1
or LV_ERR_FTL_SUB_STUCK = 1
or LV_ERR_FTL_STUCK_H = 1
or LV_ERR_FTL_INTM = 1
or LV_ERR_FTL_SUB_INTM = 1
then LV_ERR_FTL = 1
else LV_ERR_FTL = 0
endif

A.98.6 Electrical diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel level sensor circuit error	SCB	SYM_0	*STD*
	SCG + OL	SYM_1	
FTL_EL			

List of Environmental Data to store in Failure Memory : V_FTL
FTL_MMV_ENVD
CONF_DIAGCP_VOL
VS

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Plausibility Diagnosis – Stuck
  - Fuel Level Sensor Plausibility Diagnosis – Intermittent

A.98.7 Sub FTL electrical diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Sub Fuel level sensor circuit error	SCB	SYM_0	STD
	SCG + OL	SYM_1	
FTL_SUB_EL			

List of Environmental Data to store in Failure Memory : V_FTL_SUB
V_FTL_SUB_MMV_ENVD
CONF_DIAGCP_VOL
VS

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Plausibility Diagnosis – Stuck
  - Fuel Level Sensor Plausibility Diagnosis – Intermittent

A.98.8 Stuck diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel level sensor signal stuck	Signal stuck	SYM_0	STD
FTL_STUCK			

List of Environmental Data to store in Failure Memory :
FTL_MMV_ENVD
FTL_MMV_REF_INI_ENVD
FTL_MMV_DIF_POS_DIAG_ENVD
FTL_MMV_DIF_NEG_DIAG_ENVD

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Intermittent Diagnosis
  - EVAP monitoring volume calculation : DTP gradient method for volume estimation

A.98.9 Sub FTL Stuck diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel level sensor signal stuck	Signal stuck	SYM_0	STD
FTL_SUB_STUCK			

List of Environmental Data to store in Failure Memory :
V_FTL_SUB_ENVD
V_FTL_SUB_REF_INI_ENVD
V_FTL_SUB_DIF_POS_DIAG_ENVD
V_FTL_SUB_DIF_NEG_DIAG_ENVD

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Intermittent Diagnosis
  - EVAP monitoring volume calculation : DTP gradient method for volume estimation

A.98.10 Stuck high diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel level sensor signal stuck			*STD*
	Signal stuck high	SYM_1	
FTL_STUCK_H			

List of Environmental Data to store in Failure Memory :

FTL_MMV_ENVD
FTL_MMV_REF_INI_ENVD
FTL_MMV_DIF_POS_DIAG_ENVD
FTL_MMV_DIF_NEG_DIAG_ENVD

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Intermittent Diagnosis
  - EVAP monitoring volume calculation : DTP gradient method for volume estimation

A.98.11 Plausibility-intermittent diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel level sensor signal Intermittent	Signal intermittent	SYM_0	*MEM*
FTL_INTM			

List of Environmental Data to store in Failure Memory :

WHEEL_GRD_MMV_ENVD_H
WHEEL_GRD_MMV_ENVD_L
FTL_MES
FTL_MMV_ENVD

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Plausibility Diagnosis – Stuck
  - EVAP monitoring volume calculation : DTP gradient method for volume estimation

A.98.12 Sub FTL plausibility-intermittent diagnosis

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Sub Fuel level sensor signal Intermittent	Signal intermittent	SYM_0	*MEM*
FTL_SUB_INTM			

List of Environmental Data to store in Failure Memory : WHEEL_GRD_MMV_ENVD_H
WHEEL_GRD_MMV_ENVD_L
V_FTL_SUB
V_FTL_SUB_MMV_ENVD

Application assistance:

- EVAP monitoring volume calculation : DTP gradient method for volume estimation
- The following diagnosis functions are inhibited :
  - Fuel Level Sensor Plausibility Diagnosis – Stuck
  - EVAP monitoring volume calculation : DTP gradient method for volume estimation

A.99 Fuel tank level diagnosis (Applic. Inc.)

A.99.1 Application Incidences:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_FTL_STUCK	O/V	0H...1H	0...1	1	-
Inhibition flag for FTL_STUCK diagnosis					

Input data:

LV_ERR_FTL_EL	LV_ERR_FTL_SUB_EL	LV_ERR_FTL_INTM	LV_ERR_FTL_SUB_INTM
LV_ERR_VS	LV_ERR_MWSS	LV_DC	

FUNCTION DESCRIPTION:

Application Conditions:

Initialization: **at reset** LV_INH_DIAG_FTL_STUCK = 0

Reccurence: **100 ms**

Activation: at every Engine State

Deactivation: -

Formula section:

If LV_ERR_FTL_EL = 1
or LV_ERR_FTL_SUB_EL = 1
or LV_ERR_FTL_INTM = 1
or LV_ERR_FTL_SUB_INTM = 1
or LV_ERR_VS = 1
or LV_ERR_MWSS = 1
then LV_INH_DIAG_FTL_STUCK = 1
else LV_INH_DIAG_FTL_STUCK = 0
endif

A.100 Alternator power management diagnosis

A.100.1 Alternator electric diagnostic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CDN_DIAG_ALTER	-	0...7H	0...7	1	[-]
Diagnosis condition for each symptom of XX bit 0: diagnosis condition for symptom SYM_0 bit 1: diagnosis condition for symptom SYM_1 bit 2: diagnosis condition for symptom SYM_2					
ERR_DIAG_ALTER	-	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Raw value of error symptom for alternator (only parameter)					
LV_ERR_ALTER	O/V	0...1H	0...1	1	[-]
Present failure : failure after filtering of alternator diagnosis					

Input data:

CONF_BAT	LV_INH_DIAG_ALTER	NC_IDX_DIAG_ALTER	LV_CDN_VB_OBD1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_ALTER	1	0...FFH	0...255	1	[-]
Debounce counter increment value for Alternator diagnosis					
C_ABC_MAX_ALTER	1	1...FFH	1...255	1	[-]
Debounce counter maximum value for Alternator diagnosis					
LC_ALTER_DIAG_OC_INH	1	0...1H	0...1	1	[-]
Inhibition of OC symptom detection (typical value 1 : inhibited)					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PSD_DLY_ALTER	1	1...FH	1...15	1	[-]
Number of recurrences with diagnosis deactivation after diagnosis activation conditions are met (typical 2)					

Import actions:

ACTION_ERRM_FilterMulticondition(IN<XX>, IN< CDN_DIAG_ALTER >, IN< ERR_DIAG_ALTER >, IN<C_ABC_INC_ALTER>, IN<C_ABC_MAX_ALTER>, OUT<LV_ERR_ALTER>)

FUNCTION DESCRIPTION:

General information:

The Alternator is driven by the ECU via an output driver. The failure detection is done by ECU Hardware.
The purpose of the diagnosis is to detect electrical faults as defined by OBD I requirements.

Description

Error-symptoms are defined to this diagnosis function as following :

ERR_SYM_ALTER:

```
□□□□
 │ │ │ └─> Short circuit to Vbatt    SCP     (= SYM_0)
 │ │ └───> Short circuit to GND      SCG     (= SYM_1)
 │ └─────> Open load                 OL      (= SYM_2)
 └───────> -                                 (= SYM_3)
```

Application conditions:

Initialisation: LV_ERR_ALTER, ERR_SYM_ALTER, LV_END_DIAG_ALTER are initialized according Filter-type.
at reset LV_CDN_DIAG_ALTER = 0

Recurrence: 100 ms

Activation: CONF_BAT = 1 **and** (LV_ERR_ALTER=0 **or** LV_END_DIAG_ALTER=0) **and** LV_CDN_VB_OBD1=1

Deactivation: When the activation condition is not fulfilled

Formula section:

If LV_INH_DIAG_ALTER = 0
Then
Basic diagnosis conditions CDN_DIAG_ALTER is set according driver information
Failure symptoms ERR_DIAG_ALTER is set according driver information

If LV_INH_DIAG_ALTER change from 1 to 0 (Activation conditions is set) it's necessary to set CDN_DIAG_ALTER=0 during NC_PSD_DLY_ALTER recurrences
Then
CDN_DIAG_ALTER = 0
Else
{ Diagnosis info valid: no inhibition during NC_PSD_DLY_ALTER recurrences }

{ Inhibition of open load symptom (SYM_2) : }

If LC_ALTER_DIAG_OC_INH = 1
Then
bit 2 of ERR_DIAG_ALTER = 0 { The open load (SYM_2) is removed }
Endif

Endif
Else
CDN_DIAG_ALTER = 0
Endif

Failure filtering and error management treatment:

For failure and error management treatment the anti-bounce mechanism is called with the parameters CDN_DIAG_ALTER and ERR_DIAG_ALTER.

ACTION_ERRM_FilterMulticondition (IN<NC_IDX_DIAG_ALTER>, IN< CDN_DIAG_ALTER >, IN< ERR_DIAG_ALTER >, IN<C_ABC_INC_ALTER>, IN<C_ABC_MAX_ALTER>, OUT<LV_ERR_ALTER>)
This algorithm determines:

ERR_SYM_ALTER	(Detected error symptom for alternator diagnosis)
LV_ERR_ALTER	(Error flag for debounced error of alternator)
LV_CDN_DIAG_ALTER	(Diagnostics condition information)
LV_END_DIAG_ALTER	(End of diagnosis information)

A.100.2 Alternator plausibility diagnostic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_ALTER_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
For each symptom : status of failure (set to 1 when failure symptom detected)					
LV_CDN_DIAG_ALTER_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_END_DIAG_ALTER_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic performed at least one time					
LV_ERR_ALTER_PLAUS	O/V	0...1H	0...1	1	[-]
Present failure after filtering of alternator plausibility diagnostic					
CTR_DIAG_ALTER_PLAUS	V	0..FFH	0…25.5	0.1	[s]
Counter to manage the plausibility diagnostic duration					
CTR_DIAG_ALTER_OK	V	0...FFH	0…255	1	[-]
Counter of number time battery voltage conform					

Input data:

CONF_BAT	LV_ES	ALTPWM	VB
VS	LV_ERR_ALTER		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_ALTER_PLAUS	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_ALTER_PLAUS	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					
C_DLY_ALTER_DIAG_PLAUS	1	0..FFH	0…25.5	0.1	[s]
Delay to determinate plausibility diagnostic failure (typical value 10)					
C_CTR_DIAG_ALTER_OK	1	1..FFH	1…255	1	[-]
Number minimum of time the battery voltage test has to be good (typical value 3)					
C_ALTPWM_DIAG_ENA_MIN	1	0...7FFH	0...99.89	0.0488	%
Minimum alternator voltage value threshold to start the plausibility diagnostic					
C_ALTPWM_DIAG_ENA_MAX	1	0...7FFH	0...99.89	0.0488	%
Maximum alternator voltage value threshold to start the plausibility diagnostic					
C_VB_ALTER_DIAG_MAX	1	0... FFH	0... 26	0.1019608	[V]
Battery voltage threshold to detect a failure					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_IDX_DIAG_ALTER_PLAUS	1	1...FFFFH	1...65535	1	[-]
Failure index for plausibility alternator diagnostic					

Import actions:

ACTION_ERRM_FilterSymptom(IN<IDX_DIAG>, IN< LV_CDN_DIAG_XX >, IN< ERR_SYM_XX >, IN< C_ABC_INC_XX >, IN< C_ABC_MAX_XX >, IN< C_ABC_DEC_XX >, OUT< LV_ERR_XX >)

Error treatment

Diagnostic Instance		Symptoms		ABC type	CARB class
Description	Name (XX)	Description	Nr		
Alternator plausibility diagnosis	ALTER_PLAUS	-	0	MEM	NO
		-	1		
		-	2		
		Plausibility failure	3		

FUNCTION DESCRIPTION:

General information:

Error detection in case of alternator voltage is low during a delay, but the battery voltage stays at the same voltage. The MEM filter type is used to avoid vehicle effect in case of intermittent error.

Description:

Error-symptoms are defined to this diagnosis function as following :

ERR_SYM_ALTER_PLAUS

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> -                          (= SYM_0)
 │  │  └────> -                          (= SYM_1)
 │  └───────> -                          (= SYM_2)
 └──────────> -Plausibility failure      (= SYM_3)
```

Application conditions:

Initialization at reset, at key transition and at CLRFMY event :

CTR_DIAG_ALTER_PLAUS = C_DLY_ALTER_DIAG_PLAUS
CTR_DIAG_ALTER_OK =0
The other diagnostic data are initialized according filtering configuration.

Recurrence: 100 ms

Activation: CONF_BAT = 1 **and** LV_ES=0 **and** VS>0 **and** LV_ERR_ALTER=0

Deactivation: When activation function is not filling in
In case of activation condition is false then init treatment is applied :
CTR_DIAG_ALTER_PLAUS = C_DLY_ALTER_DIAG_PLAUS
CTR_DIAG_ALTER_OK = 0
LV_CDN_DIAG_ALTER_PLAUS = 0

Formula section:

Diagnostic condition calculation :

```
If  C_ALTPWM_DIAG_ENA_MIN < ALTPWM < C_ALTPWM_DIAG_ENA_MAX
Then
   { When the battery voltage is low the number of good result is increased }
   If      VB < C_VB_ALTER_DIAG_MAX
   Then
           CTR_DIAG_ALTER_OK = CTR_DIAG_ALTER_OK + 1
   Endif

   { When the delay is elapsed the condition is set }
   CTR_DIAG_ALTER_PLAUS = CTR_DIAG_ALTER_PLAUS - 0.1s

   If      CTR_DIAG_ALTER_PLAUS=0
   Then
           LV_CDN_DIAG_ALTER_PLAUS=1
   Else
           LV_CDN_DIAG_ALTER_PLAUS=0
   Endif
Else
    { When the condition on ALTPWM is not fill in the diagnostic is re-initialized }
    LV_CDN_DIAG_ALTER_PLAUS = 0
    CTR_DIAG_ALTER_PLAUS = C_DLY_ALTER_DIAG_PLAUS
    CTR_DIAG_ALTER_OK = 0
Endif
```

Symptoms calculation :

```
If   LV_CDN_DIAG_ALTER_PLAUS=1
Then
    { When the counter of good diagnostic is too low the failure is set }
    If  CTR_DIAG_ALTER_OK < C_CTR_DIAG_ALTER_OK
    Then
           ERR_SYM_ALTER_PLAUS = SYM_3
    Else
           ERR_SYM_ALTER_PLAUS = NO_SYM
    Endif

    { The diagnostic is re-initialized }
    CTR_DIAG_ALTER_PLAUS = C_DLY_ALTER_DIAG_PLAUS
    CTR_DIAG_ALTER_OK =0

Endif
```

For failure and error management treatment the anti-bounce mechanism is called :

```
ACTION_ERRM_FilterSymptom( IN<NC_IDX_DIAG_ALTER_PLAUS>,
IN<LV_CDN_DIAG_ALTER_PLAUS>, IN<ERR_SYM_ALTER_PLAUS>,
IN<C_ABC_INC_ALTER_PLAUS>, IN<C_ABC_MAX_ALTER_PLAUS>, IN<1>,
OUT<LV_ERR_ALTER_PLAUS>)
```

A.100.3 Alternator charge efficiency plausibility diagnostic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_ALTER_CHA_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
For each symptom : status of failure (set to 1 when failure symptom detected)					
LV_CDN_DIAG_ALTER_CHA_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_END_DIAG_ALTER_CHA_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic performed at least one time					
LV_ERR_ALTER_CHA_PLAUS	O/V	0...1H	0...1	1	[-]
Present failure after filtering of alternator charge efficiency plausibility diagnostic					
CTR_DIAG_ALTER_CHA_PLAUS	V	0..FFH	0…25.5	0.1	[s]
Counter to manage the plausibility diagnostic duration					
CTR_DIAG_CHA_ALTER_OK	V	0...FFH	0…255	1	[-]
Counter of number time level current charge conform					

Input data:

CONF_BAT	CUR_BAT_EFF_MMV	LV_ES	ALTPWM
SOC	LV_ERR_ALTER	LV_ERR_BAT_SENS	LV_ERR_ALTER_PLAUS
LV_SENS_LIN_READY	GEN_LOAD_MMV	TBAT_CLC	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_ALTER_CHA_PLAUS	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_ALTER_CHA_PLAUS	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					
C_DLY_ALTER_DIAG_CHA_PLAUS	1	0..FFH	0…25.5	0.1	[s]
Delay to determinate plausibility diagnostic failure					
C_ALTPWM_DIAG_CHA_ENA_MIN	1	0...7FFH	0...99.89	0.0488	%
Minimum alternator voltage value threshold to start the plausibility diagnostic					
C_ALTPWM_DIAG_CHA_ENA_MAX	1	0...7FFH	0...99.89	0.0488	%
Maximum alternator voltage value threshold to start the plausibility diagnostic					
C_SOC_DIAG_CHA_ENA_MAX	1	0...FFFFH	0...99.9985	1.525 e-3	%
Maximum SOC value threshold to start the plausibility diagnostic					
C_TBAT_CLC_CHA_DIAG	1	0... FEH	-48... 142.5	0.75	[°C]
Minimum temperature battery threshold to start the plausibility diagnostic					
C_GEN_LOAD_MMV_DIAG_MAX	1	0...FFH	0...99.6	0.4	%
Maximum filtered electrical load PWM signal from Alternator to start the plausibility diagnostic					
C_CUR_BAT_EFF_MMV_DIAG_MIN	1	8000... 7FFFH	-120 ...120	3.66217e-3	[A]
Current threshold to detect a failure					
C_CTR_DIAG_CHA_ALTER_OK	1	1..FFH	1…255	1	[-]
Number minimum of time the battery current threshold test has to be good					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_IDX_DIAG_ALTER_CHA_PLAUS	1	1...FFFFH	1...65535	1	[-]
Failure index for alternator charge efficiency plausibility diagnostic					

Import actions:

ACTION_ERRM_FilterSymptom(IN<IDX_DIAG>, IN< LV_CDN_DIAG_XX >, IN< ERR_SYM_XX >, IN< C_ABC_INC_XX >, IN< C_ABC_MAX_XX >, IN< C_ABC_DEC_XX >, OUT< LV_ERR_XX >)

Error treatment

Diagnostic Instance		Symptoms		ABC type	CARB class
Description	Name (XX)	Description	Nr		
Alternator charge efficiency plausibility diagnostic	ALTER_CHA_PLAUS	-	0	MEM	NO
		-	1		
		-	2		
		Plausibility failure	3		

FUNCTION DESCRIPTION:

General information:

Error detection in case of alternator is not efficiency charging the battery during a delay.

Description:

Error-symptoms are defined to this diagnosis function as following :

ERR_SYM_ALTER_CHA_PLAUS

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │  │  │  └─> -                              (= SYM_0)
 │  │  └────> -                              (= SYM_1)
 │  └───────> -                              (= SYM_2)
 └──────────> -Plausibility charge failure   (= SYM_3)
```

Application conditions:

Initialization at reset, at key transition and at CLRFMY event :

CTR_DIAG_ALTER_CHA_PLAUS = C_DLY_ALTER_DIAG_CHA_PLAUS
CTR_DIAG_CHA_ALTER_OK =0
The other diagnostic data are initialized according filtering configuration.

Recurrence: 100 ms

Activation: CONF_BAT=1 **and** SOC <= C_SOC_DIAG_CHA_ENA_MAX
and LV_ES=0 **and** LV_ERR_BAT_SENS=0
and LV_ERR_ALTER=0
and LV_ERR_ALTER_PLAUS=0 **and** LV_SENS_LIN_READY=1
and GEN_LOAD_MMV <= C_GEN_LOAD_MMV_DIAG_MAX
and TBAT_CLC >= C_TBAT_CLC_CHA_DIAG

Deactivation: *When activation function is not filling in*
In case of activation condition is false then init treatment is applied :
CTR_DIAG_ALTER_CHA_PLAUS=C_DLY_ALTER_DIAG_CHA_PLAUS
CTR_DIAG_CHA_ALTER_OK = 0
LV_CDN_DIAG_ALTER_CHA_PLAUS = 0

Formula section:

Diagnostic condition calculation :

If C_ALTPWM_DIAG_CHA_ENA_MIN<=ALTPWM<=C_ALTPWM_DIAG_CHA_ENA_MAX
Then
{ When the battery current is high the number of good result is increased }
If CUR_BAT_EFF_MMV > C_CUR_BAT_EFF_MMV_DIAG_MIN
Then
CTR_DIAG_CHA_ALTER_OK = CTR_DIAG_CHA_ALTER_OK + 1
Endif

{ When the delay is elapsed the condition is set }
CTR_DIAG_ALTER_CHA_PLAUS = CTR_DIAG_ALTER_CHA_PLAUS - 0.1s

If CTR_DIAG_ALTER_CHA_PLAUS=0
Then
LV_CDN_DIAG_ALTER_CHA_PLAUS=1
Else
LV_CDN_DIAG_ALTER_CHA_PLAUS=0
Endif
Else
{ When the condition on ALTPWM is not fill in the diagnostic is re-initialized }
LV_CDN_DIAG_ALTER_CHA_PLAUS = 0
CTR_DIAG_ALTER_CHA_PLAUS = C_DLY_ALTER_DIAG_CHA_PLAUS
CTR_DIAG_CHA_ALTER_OK = 0
Endif

Symptoms calculation :

If LV_CDN_DIAG_ALTER_CHA_PLAUS=1
Then
{ When the counter of good diagnostic is too low the failure is set }
If CTR_DIAG_CHA_ALTER_OK < C_CTR_DIAG_CHA_ALTER_OK
Then
ERR_SYM_ALTER_CHA_PLAUS = SYM_3
Else
ERR_SYM_ALTER_CHA_PLAUS = NO_SYM
Endif

{ The diagnostic is re-initialized }
CTR_DIAG_ALTER_CHA_PLAUS = C_DLY_ALTER_DIAG_CHA_PLAUS
CTR_DIAG_CHA_ALTER_OK =0

Endif

For failure and error management treatment the anti-bounce mechanism is called :

ACTION_ERRM_FilterSymptom(IN<NC_IDX_DIAG_ALTER_CHA_PLAUS>, IN<LV_CDN_DIAG_ALTER_CHA_PLAUS>, IN<ERR_SYM_ALTER_CHA_PLAUS>, IN<C_ABC_INC_ALTER_CHA_PLAUS>, IN<C_ABC_MAX_ALTER_CHA_PLAUS>, IN<1>, OUT<LV_ERR_ALTER_CHA_PLAUS>)

A.101 Alternator power management diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_ALTER	O/V	0...1H	0...1	1	[-]
Alternator diagnosis inhibition					

Input data:

CONF_BAT	LV_IGK	LV_ST_END	VB

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VB_MAX_DIAG_ALTER	1	0...FFH	0...25.89843	0.1015625	[V]
Maximum Battery voltage threshold for Alternator diagnosis					
C_VB_MIN_DIAG_ALTER	1	0...FFH	0...25.89843	0.1015625	[V]
Minimum Battery voltage threshold for Alternator diagnosis					

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_IDX_DIAG_ALTER	1	1...FFFFH	1...65535	1	[-]
Failure index for alternator diagnostic					

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_ALTER is used in chapter “Electrical diagnosis of power stage outputs” to activat/deactivate the ATIC39 diagnosis for alternator management

Description:

Application conditions:

Initialisation:
At reset event : LV_INH_DIAG_ALTER = 0

Recurrence:
Application incidence recurrence = 100ms
Diagnostic recurrence = 100ms

Activation: CONF_BAT = 1

Deactivation: When the activation condition is not fulfilled

Formula section:

If LV_IGK = 1
and VB < C_VB_MAX_DIAG_ALTER
and VB > C_VB_MIN_DIAG_ALTER
and LV_ST_END = 1
then LV_INH_DIAG_ALTER = 0
else LV_INH_DIAG_ALTER = 1
endif

Error treatment

Diagnostic Instance		Symptoms		ABC type	CARB class
Description	**Name (XX)**	**Description**	**Nr**		
Diagnostic alternator	ALTER	Short circuit to battery	0	MPL_STD_INI	NO
		Short circuit to ground	1		
		Open circuit	2		
			3		

A.102 CAN-Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DLY_CAN_DIAG	-	0...FFH	0...2550	10	ms
Time period when CAN diagnosis is forbidden					
LV_CAN_DIAG_ACT	V	0...1H	0...1	1	-
Boolean for CAN diagnosis active					
LV_ERR_CAN	V	0...1H	0...1	1	-
Boolean for CAN error					
LV_CDN_DIAG_CAN_BUS_OFF	V	0...1H	0...1	1	-
Diagnosis conditions for CAN BUS OFF diagnosis fulfilled					
ERR_SYM_CAN_BUS_OFF	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of CAN_BUS_OFF diagnosis					
LV_ERR_CAN_BUS_OFF	V	0...1H	0...1	1	-
Boolean for transfer problem in CAN-Communication					
LV_ERR_CAN_FRF	V	0...1H	0...1	1	-
Boolean for Freeze Frame requested by TCU					
LV_INH_DIAG_TIMEOUT_TCU1	O/V	0...1H	0...1	1	-
Boolean for TIMEOUT_TCU1 diagnosis inhibition					
LV_CDN_DIAG_TIMEOUT_TCU1	V	0...1H	0...1	1	-
Diagnosis conditions for TIMEOUT TCU1 diagnosis fulfilled					
ERR_SYM_TIMEOUT_TCU1	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of TIMEOUT_TCU1 diagnosis					
LV_ERR_TIMEOUT_TCU1	V/O	0...1H	0...1	1	-
Boolean for Timeout in CAN-Communication with TCU					
LV_TCU1_MSG_MISS_ONCE	-	0...1H	0...1	1	-
Flag indicating first time TCU1 message was not received					
LV_TCU2_MSG_MISS_ONCE	-	0...1H	0...1	1	-
Flag indicating first time TCU2 message was not received					
T_DLY_TIMEOUT_TCU1	-	0...255H	0...255	1	??
Timer for TCU1 diagnosis					
T_DLY_TIMEOUT_TCU2	-	0...255H	0...255	1	??
Timer for TCU2 timeout diagnosis					

Input data:

VB	LV_IGK	N_32	LV_STE_TCU1_CAN_IN
TOUT_CTR_TCU1	CONF_TCU		

FUNCTION DESCRIPTION:

General information:

The diagnosis function checks the receiving of messages from the TCU via the CAN plus the correct transmission of messages via the CAN.

Error symptoms are defined to the diagnosis functions as following:

TIMEOUT_TCU1:

```
□□□□
│ │ │ └─> -                                   (= SYM_0)
│ │ └───> -                                   (= SYM_1)
│ └─────> -                                   (= SYM_2)
└───────> No Message received from TCU        (= SYM_3)
```

CAN_BUS_OFF:

```
□□□□
│ │ │ └─> -                                   (= SYM_0)
│ │ └───> -                                   (= SYM_1)
│ └─────> Communication Problem on the CAN    (= SYM_2)
└───────>-                                    (= SYM_3)
```

CAN_FRF:

```
□□□□
│ │ │ └─> Freeze frame request via CAN        (= SYM_0)
│ │ └───> -                                   (= SYM_1)
│ └─────> -                                   (= SYM_2)
└───────> -                                   (= SYM_3)
```

Application conditions:

Initialisation:

Recurrence: 10ms

Activation:

If LV_IGK = 1
and VB > C_VB_MIN_CAN_DIAG
and N_32 > C_N_MIN_CAN_DIAG
then **if** T_DLY_CAN_DIAG >= C_T_DLY_MIN_CAN_DIAG
then LV_CAN_DIAG_ACT = 1
else LV_CAN_DIAG_ACT = 0
$T\_DLY\_CAN\_DIAG_{(n)} = T\_DLY\_CAN\_DIAG_{(n-1)} + 1$
else T_DLY_CAN_DIAG = 0
LV_CAN_DIAG_ACT = 0

Before the diagnosis starts after ignition on, a time of 140 ms is waited for initialization TCU.

After activation of the CAN-diagnosis the timer T_DLY_TIMEOUT_TCU1 and T_DLY_TIMEOUT_TCU2 are started. If the TCU1 timer reaches the maximum value

C_MAX_TIMEOUT_TCU1 without having received a message, the anti-bounce counter is incremented with C_ABC_INC_TIMEOUT_TCU1 and the timer starts again.
If the TCU2 timer reaches the maximum value C_MAX_TIMEOUT_TCU1 without having received a message all TCU2 values are set to init values and the timer starts again

If again no message arrives, this process repeats until the anti-bounce counter reaches C_ABC_MAX_TIMEOUT_TCU1 and the failure bit LV_ERR_TIMEOUT_TCU1 is set.

If a message is transfered incorrect on the bus to one of the CAN-connected components (e.g. due to electrical interference on the CAN), the anti-bounce counter is incremented with C_ABC_INC_CAN_BUS_OFF.

The receiving of a message (in case of a Timeout-Error) or the correct transfer of a message (in case of a transfer error) decrements the corresponding anti-bounce counter with 1.
As soon as one of the CAN-Diagnosis related error bits is set to one then LV_ERR_CAN = 1. It is reset to 0 if all the corresponding anti-bounce counters are 0.

If CAN message transimission is suppressed by key-word service request (disableNormalMessageTransmission (28h)) or CAN_BUS_OFF error is detected (LV_ERR_CAN_BUS_OFF =1) TIMEOUT_TCU1 diagnosis is inhibited (LV_INH_DIAG_TIMEOUT_TCU1 = 1).

If CAN_FRF Diagnosis is enabled (LC_ENA_CAN_FRF_DIAG = 1) and the TCU requests a freeze frame & MIL ON via CAN(TCU_OBD) via TCU_OBD, the LV_ERR_CAN_FRF is set to 1, else LV_ERR_CAN_FRF is set to 0. If LC_ENA_CAN_FRF_DIAG = 0 then diagnosis is deactivated.

The Freeze Frame which is requested by TCU has to be cleared if there is no more freeze frame request from TCU regardless of current warm-up cycle counter in EMS ECU.

<u>* Emergency operation:</u>

If the error CAN_BUS_OFF occurs
then initialization values for each message will be set

If the error TIMEOUT_TCU1 occurs
then initialization values for the TCU-message will be set

For more detailed information please refer to the CAN-specification.

<u>Application conditions:</u>
* Debounce :

Anti-bounce counter increment:	C_ABC_INC_CAN_BUS_OFF C_ABC_INC_TIMEOUT_TCU1
Maximum value of anti-bounce counter:	C_ABC_MAX_CAN_BUS_OFF C_ABC_MAX_TIMEOUT_TCU1

Formula section:

 # TCU Diagnosis

```
- TCU1:
If      (CONF_TCU = 00h (AT)  or  CONF_TCU = 0Ah (CVT))
    and LV_INH_DIAG_TIMEOUT_TCU1 = 0
    and (CCP Transmission is NOT inhibited via KWP Service DNMT)
Then If    LV_STE_TCU1_CAN_IN = 1
     Then If    LV_CAN_DIAG_ACT = 1
          Then If    TOUT_CTR_TCU1 = 0
               Then LV_CDN_DIAG_TIMEOUT_TCU1 = 1
                    T_DLY_TIMEOUT_TCU1 = 0
                    LV_TCU1_MSG_MISS_ONCE = 0
               Else If    LV_TCU1_MSG_MISS_ONCE = 1
                    Then T_DLY_TIMEOUT_TCU1(n)=T_DLY_TIMEOUT_TCU1(n – 1)+1
                         If    T_DLY_TIMEOUT_TCU1 >= C_MAX_TIMEOUT_TCU1
                         Then LV_CDN_DIAG_TIMEOUT_TCU1 = 1
                              ERR_SYM_TIMEOUT_TCU1 = SYM_3
                              T_DLY_TIMEOUT_TCU1 = 0
                         Endif
                    Else LV_TCU1_MSG_MISS_ONCE = 1
                    Endif
               Endif
          Else T_DLY_TIMEOUT_TCU1 = 0
               LV_TCU1_MSG_MISS_ONCE = 0
          Endif
     Else LV_TCU1_MSG_MISS_ONCE = 0
     Endif
     If    LV_ERR_TIMEOUT_TCU1 = 1
     Then set all TCU1 variables to init. values
Endif
```

- TCU2:

```
If      (CONF_TCU = 00h (AT)  or  CONF_TCU = 0Ah (CVT))
   and  (CCP Transmission is NOT inhibited via KWP Service DNMT)
Then If     LV_STE_TCU1_CAN_IN = 1
     Then If     LV_CAN_DIAG_ACT = 1
          Then If     TOUT_CTR_TCU2 = 0
               Then T_DLY_TIMEOUT_TCU2 = 0
                    LV_TCU2_MSG_MISS_ONCE = 0
               Else If     LV_TCU2_MSG_MISS_ONCE = 1
                    Then T_DLY_TIMEOUT_TCU2(n)=T_DLY_TIMEOUT_TCU2(n – 1)+1
                         If     T_DLY_TIMEOUT_TCU2 >= C_MAX_TIMEOUT_TCU1
                         Then T_DLY_TIMEOUT_TCU2 = 0
                              Set default values for TCU2 variables
                         Endif
                    Else LV_TCU2_MSG_MISS_ONCE = 1
                    Endif
               Endif
          Else T_DLY_TIMEOUT_TCU2 = 0
               LV_TCU2_MSG_MISS_ONCE = 0
          Endif
     Else LV_TCU2_MSG_MISS_ONCE = 0
     Endif
Endif
```

 # CAN-Bus off Diagnosis

```
If      LV_IGK = 1
Then If     LV_CAN_DIAG_ACT = 1
     Then LV_CDN_DIAG_CAN_BUS_OFF = 1
          If     CAN_BUS_OFF error detected in BSW
          Then ERR_SYM_CAN_BUS_OFF = SYM_2
               Set init values for EMSx variables
          Endif
     Endif
     If     LV_ERR_CAN_BUS_OFF = 1
     Then LV_INH_DIAG_TIMEOUT_TCU1 = 1
     Endif
```

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_CAN_BUS_OFF	1	0...FFH	0...255	1	-
Anti-bounce counter increment for can bus controller check.					
C_ABC_INC_TIMEOUT_TCU1	1	0...FFH	0...255	1	-
Anti-bounce counter message timeout for TCU1.					
C_ABC_MAX_CAN_BUS_OFF	1	1...FFH	1...255	1	-
Maximum value of the anti-bounce counter for can bus controller check.					
C_ABC_MAX_TIMEOUT_TCU1	1	1...FFH	1...255	1	-
Maximum value for the anti-bounce counter for TCU1.					
C_T_DLY_MIN_CAN_DIAG	1	0...FFH	0...2550	10	ms
Delay time after voltage drop for CAN-Diagnosis activating					
C_VB_MIN_CAN_DIAG	1	0...FFH	0...26	0.1	V
Minimum battery voltage for active CAN-diagnosis					
C_N_MIN_CAN_DIAG	1	0...FFH	0...8160	32	rpm
Minimum engine speed for active CAN-diagnosis					
C_MAX_TIMEOUT_TCU1	1	0...FFH	0...2550	10	ms
Maximum waiting time for TCU-message.					
LC_ENA_CAN_FRF_DIAG	1	0...1H	0...1	1	-
Logical constant to enable CAN_FRF diagnosis					

A.102.1 CAN diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms:

Diagnosis	Symptom description	Symptom	Filter-Type
Serial communication Link Malfunction		SYM_0	STD
		SYM_1	
	Communication Problem on the CAN	SYM_2	
CAN_BUS_OFF		SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
T_AST_ENVD
V_IGK
DIAG_INST_ENVD

Diagnosis	Symptom description	Symptom	Filter-Type
TCU Request for MIL On / Freeze Frame to ECU via CAN	Freeze frame requested via CAN	SYM_0	NO
		SYM_1	
		SYM_2	
CAN_FRF		SYM_3	

List of Environmental Data to store in failure memory: TAM
TCU_OBD
TCO_ST
VS

Diagnosis	Symptom description	Symptom	Filter-Type
Serial communication problem with TCU (Timeout)		SYM_0	STD
		SYM_1	
		SYM_2	
TIMEOUT_TCU1	No Message received from TCU	SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
T_AST_ENVD
V_IGK
DIAG_INST_ENVD

* Application Incidence:

- The following Adaptation Functions are inhibited:
  - Idle Speed Adaptation (only in case of TCU-Error)
- The following Control Functions are inhibited:
  - Torque Intervention for Gear Shift (only in case of Timeout TCU, Serial Communication Link Malfunction Error)

A.103 Torque Request for Safety Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_TQ_LIM_INTV	O/V	0...1H	0...1	1	-
Diagnosis condition (0: Passive, 1: Active)					
ERR_SYM_TQ_LIM_INTV	O/V	0H 1H	NO_SYM SYM_0	1	-
Symptoms of the failure					
LV_ERR_TQ_LIM_INTV	O/V	0...1H	0...1	1	-
Present failure (0: Passive, 1: Active)					
LV_END_DIAG_TQ_LIM_INTV	O/V	0...1H	0...1	1	-
Result of diagnosis (0: Passive, 1: Active)					

Input data:

LV_TQ_LIM_INTV	LV_MTC_CUR_OFF	LV_IGK	LV_ST
LV_ES			

FUNCTION DESCRIPTION:

General information:

The objective of this function is to determine the number of intervention actions concerning the torque request for safety and to set the failure table entry if the counter exceeds a calibratable threshold.

The monitoring level generates a desired indicated torque TQI_SP_MON. If the actual torque TQI_AV exceeds this value for more than a calibratable threshold a torque demand for the slow and the fast path is generated. The flag LV_TQ_LIM_INTV shows whether this torque intervention is active or not (see chapter D – "Torque request for safety").

Application conditions:

Initialisation: ECU reset **OR** transition LV_IGK = 0 → 1
LV_CDN_DIAG_TQ_LIM_INTV = 0
ERR_SYM_TQ_LIM_INTV = NO_SYM
LV_ERR_TQ_LIM_INTV = 0

Recurrence: 100 ms

Activation: LV_MTC_CUR_OFF == 0 **AND** LV_ST == 0 **AND** LV_ES == 0
Set LV_CDN_DIAG_TQ_LIM_INTV = 1

Deactivation: LV_MTC_CUR_OFF == 1 **OR** LV_ST == 1 **OR** LV_ES == 1
Set LV_CDN_DIAG_TQ_LIM_INTV = 0

Formula section:

IF LV_TQ_LIM_INTV == 1
THEN ERR_SYM_TQ_LIM_INTV = 01h (Maximum allowed value exceeded)
ELSE ERR_SYM_TQ_LIM_INTV = 0h
ENDIF

After debounce LV_ERR_TQ_LIM_INTV = 1

Configuration data :

Diagnosis	Symptom	Nr	ABC type
Torque intervention for safety active TQ_LIM_INTV	Actual or requested torque higher than allowed limit	0	See remarks below

Remarks:

- decrement must be calibrated with maximum value
- if failure is present, no healing possible
- anitbounce counter is initialized with zero at reset
- LV_END_DIAG_TQ_LIM_INTV is calculated in antibounce algorithm

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQ_LIM_INTV	1	0...FFH	0...255	1	-
Debounce counter increment value for torque request for safety diagnosis					
C_ABC_DEC_TQ_LIM_INTV	1	0...FFH	0...255	1	-
Debounce counter decrement value for torque request for safety diagnosis					
C_ABC_MAX_TQ_LIM_INTV	1	01...FFH	1...255	1	-
Debounce counter maximum value for the for torque request for safety diagnosis					

A.104 Reset resistant ETC- Limp Home

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_N_LIM_REQ_RST_CHK	V / O	0H...1H	0...1	1	-
Flag to be set if LIH_ACT is set after reset					
LIH_ACT	V/S	0H...FFFFH	0...65536	1	-
0H = no ETC-Limp Home / AF50H = ETC-Limp Home active – Word is stored every 100 ms to reset resistant memory and stored twice as complement					

Input data:

LV_ERR_PVS	LV_ERR_TPS	LV_N_LIM_REQ_RST_CHK	LV_IGK
LV_N_LIM_REQ_MON	LV_ERR_TQI_N_MAX_MON		

FUNCTION DESCRIPTION:

General information:

If ETC-Limp home is active, performance is limited for the driver. After a reset of the ECU this limitation of performance is lost from one second to another. To prevent this dangerous situation the word LIH_ACT is saved to reset independant memory every 100 ms. This special part of memory is not deleted in case of warm reset (software reset). In case of cold reset (hardware reset) even information in this reset resistant memory is lost.

While initializing the ECU the word LIH_ACT is been read out of the reset resistant memory. If it is set, LV_N_LIM_REQ_RST_CHK is set to 1 for the current driving cycle. At transition LV_IGK = 1 → 0 or if the failure memory is been deleted, LV_N_LIM_REQ_RST_CHK and LIH_ACT are reset to 0.

Application Conditions:

Recurrence: 100ms

Initialisation:

at reset (but not in case of warm-reset) **OR**
at transition LV_IGK = 1 → 0 **OR**
after clearing failure memory

LV_N_LIM_REQ_RST_CHK = 0
LIH_ACT = 0

The following initialization is only used after warm-reset :

IF after warm-reset LIH_ACT = AF50 hex (44880 dec)
(read after warm-reset from reset resistant memory)

THEN LV_N_LIM_REQ_RST_CHK = 1
(to be written only once while initializing ECU)

ELSE LV_N_LIM_REQ_RST_CHK = 0
(to be written only once while initializing ECU)

ENDIF.

Activation:

LV_IGK = 1

Deactivation:

LV_IGK = 0

<u>Formula section:</u>

IF LV_ERR_PVS = 1 **OR**
LV_ERR_TPS = 1 **OR**
LV_N_LIM_REQ_MON = 1 **OR**
LV_ERR_TQI_N_MAX_MON = 1 **OR**
LV_N_LIM_REQ_RST_CHK = 1

THEN LIH_ACT = AF50 hex (44880 dec)
(to be written to reset resistant memory every 100ms)

ELSE LIH_ACT = 0 (to be written to reset resistant memory every 100ms)

ENDIF.

A.105 Exhaust gas temperature sensor diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TEG_MES[NC_NR_TEG_SENS]	O	0...1H	0...1	1	-
Global error flag for each sensor					

Input data:

NC_NR_TEG_SENS

A.105.1 General info:

This module is just to define LV_ERR_TEG_MES even if no sensor is available because this value is used as input in other functions.

Function Description

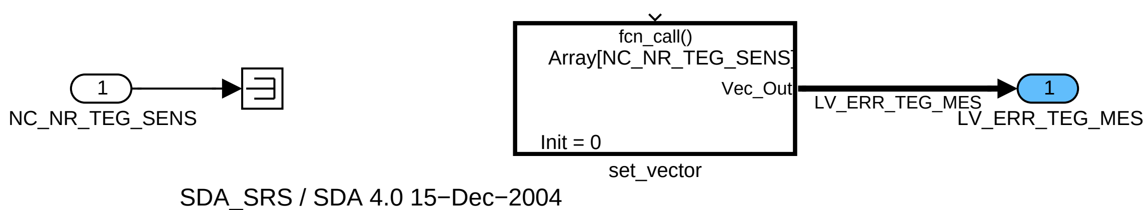


Figure 92 EXTD_SIGDG0

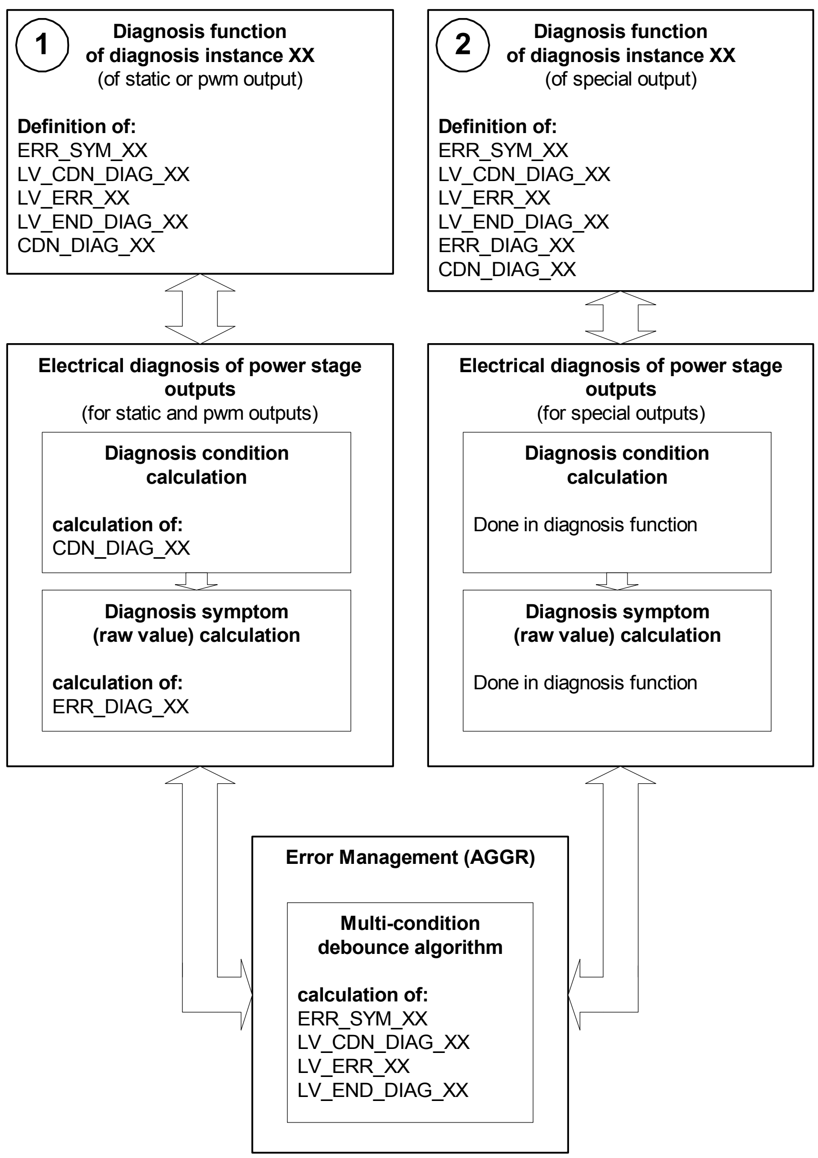
A.106 Electrical diagnosis of power stage outputs

FUNCTION DESCRIPTION:

General information:

In this function description the electrical diagnosis of power stage outputs is specified. Because of the possibility of different hardware configurations, the specification is splited into three parts plus one common part for the symptom debounce and error management treatment.

- Diagnosis of static outputs (e.g. relays)
- Diagnosis of pwm outputs (e.g. VVT actuator)
- Diagnosis of special outputs (other hardware configurations)



According these three different kind of driver outputs the management of the electrical diagnosis is different between static/pwm outputs and special outputs (see previous picture case 1 and 2).

For all types of these diagnoses three main calculations must be done:

- Diagnosis condition calculation
- Diagnosis symptom (raw value) calculation
- anti-bounce mechanism with error management treatments

In the following chapter these treatments are described.

A.106.1 Diagnosis of static outputs

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ERR_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis symptom (raw value) for each symptom for XX					
CDN_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis condition for each symptom for XX					

Input data:

LV_IGK	LV_INH_DIAG_XX		

Import actions:

ACTION_ERRM_FilterMulticondition(IN< *XX* >, IN< CDN_DIAG_XX>, IN< ERR_DIAG_XX >, OUT< LV_END_DIAG_XX >, OUT< LV_ERR_XX >, OUT< LV_CDN_DIAG_XX >, OUT< ERR_SYM_XX >)

General information:

The purpose is to diagnose electrical errors detected by the hardware for static outputs. The signal is controlled by the ATIC 39.

The driver can distinguish between three symptoms: ‘Short to battery’ SCB, ‘Short to ground’ SCG and ‘Open circuit’ OC.

Description:

By the diagnosis condition calculation the information given by the BSW (‘Diagnosis of symptom ... is valid’) that shows, that a diagnosis of a symptom was possible, since the last recurrence of the diagnosis function, is displayed for each symptom within the value CDN_DIAG_XX.

The information of the failure symptom is also delivered by the BSW (‘Failure symptom ... detected by HW’). According this information the diagnosis symptom (raw value) calculation is performed to set ERR_DIAG_XX.

The calculation of LV_ERR_XX, LV_CDN_DIAG_XX, ERR_SYM_XX and LV_END_DIAG_XX is done in the Multi-condition debounce algorithm (part of Error Management AGGR).

Application conditions:

Initialization: -

Recurrence: -

Activation: every ECU state

reactivation in the NC_PSD_DLY recurrence after LV_INH_DIAG_XX 1->0 and LV_IGK 0->1 and reset and FMY reset

Deactivation: LV_INH_DIAG_XX = 1

at deactivation: CDN_DIAG_XX = 0

Formula section:

Diagnosis condition calculation:

CDN_DIAG_XX = 0

Diagnosis condition calculation for symptom 'Short to battery' SCB:

If 'Diagnosis of symptom SCB is valid' (BSW information)

Then bit 0 of CDN_DIAG_XX = 1

Endif

Diagnosis condition calculation for symptom 'Short to ground' SCG:

If 'Diagnosis of symptom SCG is valid' (BSW information)

Then bit 1 of CDN_DIAG_XX = 1

Endif

Diagnosis condition calculation for symptom 'Open circuit' OC:

If 'Diagnosis of symptom OC is valid' (BSW information)

Then bit 2 of CDN_DIAG_XX = 1

Endif

Diagnosis symptom (raw value) calculation:

Symptom calculation for 'Short to battery' SCB:

If	'Failure symptom SCB detected by HW'	(BSW information)
Then	bit 0 of ERR_DIAG_XX = 1	
Else	bit 0 of ERR_DIAG_XX = 0	
Endif		

Symptom calculation for 'Short to ground' SCG:

If	'Failure symptom SCG detected by HW'	(BSW information)
Then	bit 1 of ERR_DIAG_XX = 1	
Else	bit 1 of ERR_DIAG_XX = 0	
Endif		

Symptom calculation for 'Open circuit' OC:

If	'Failure symptom OC detected by HW'	(BSW information)
Then	bit 2 of ERR_DIAG_XX = 1	
Else	bit 2 of ERR_DIAG_XX = 0	
Endif		

ACTION_ERRM_FilterMulticondition (*XX*, CDN_DIAG_XX, ERR_DIAG_XX, LV_END_DIAG_XX, LV_ERR_XX, LV_CDN_DIAG_XX, ERR_SYM_XX)

Remark: For failure debouncing and error management treatment the algorithm Multi-condition debounce algorithm (part of Error Management AGGR) is used with the parameters CDN_DIAG_XX and ERR_DIAG_XX.

A.106.2 Diagnosis of PWM outputs

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ERR_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis symptom (raw value) for each symptom for XX					
CDN_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis condition for each symptom for XX					

Input data:

LV_IGK	LV_INH_DIAG_XX	XXPWM	

Import actions:

ACTION_ERRM_FilterMulticondition(IN< *XX* >, IN< CDN_DIAG_XX>, IN< ERR_DIAG_XX >,
OUT< LV_END_DIAG_XX >, OUT< LV_ERR_XX >, OUT< LV_CDN_DIAG_XX >, OUT< ERR_SYM_XX >)

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose electrical errors detected by the hardware for PWM outputs. The signal is controlled by ATIC39.
The driver can distinguish between three symptoms: 'Short to battery' SCB, 'Short to ground' SCG and 'Open circuit' OC.

Because of the different filter times for SCB, SCG and OC detection, the diagnosis window, where a symptom can be detected, is defined for each symptom. A diagnosis window is defined by a minimum and a maximum duty cycle of the PWM signal controlled by the output of the driver.

SCB diagnosis window:

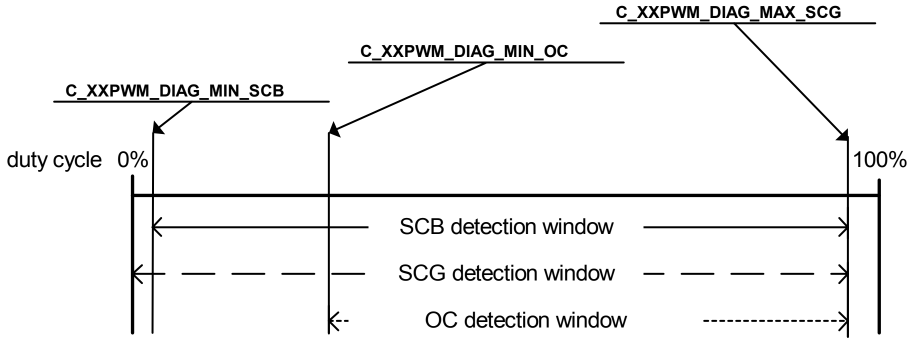
C_XXPWM_DIAG_MIN_SCB <-------> C_XXPWM_DIAG_MAX_SCG

SCG diagnosis window:

0 <-------> C_XXPWM_DIAG_MAX_SCG

OC diagnosis window:

C_XXPWM_DIAG_MIN_OC <-------> C_XXPWM_DIAG_MAX_SCG



For some diagnosis it's not necessary to define different diagnosis windows for each symptom. This is possible for all diagnosis, where a limitation of the duty cycle range is made within the function which controls the corresponding output. The limitation must be made in this way, that a diagnosis of all three symptoms can always (if LV_INH_DIAG_XX= 0) be made (CDN_DIAG_XX = 7H, LV_CDN_DIAG_XX = 1).

These two possibilities of a diagnosis condition calculation are described in the following chapter. It's distinguished between a "open range" (different diagnosis windows for each symptom necessary) and a "limited" duty cycle calculation.

A.106.2.1 Diagnosis condition calculation

A.106.2.1.1 "Open range" duty cycle

Description:

For an output with an "open range" duty cycle range, a diagnosis result of a symptom can be considered as valid, if the duty cycle of the PWM output was in the right window at least one time between two calculations of the diagnosis function and not only if diagnosis function is called. To have a better reflection of this behavior, the diagnosis condition is calculated independently of the diagnosis function.

If the recurrence of the 'diagnosis condition' calculation is the same like the 'symptom' calculation, the 'diagnosis condition' calculation must be treated after the 'symptom' calculation to have a new diagnosis condition CDN_DIAG_XX information of the following diagnosis window for the next symptom calculation.

Application conditions:

Initialization: -

Recurrence: -

Activation: every ECU state

reactivation in the NC_PSD_DLY recurrence after LV_INH_DIAG_XX 1->0 and LV_IGK 0->1 and reset and FMY reset

Deactivation: LV_INH_DIAG_XX = 1

at deactivation CDN_DIAG_XX = 0

Formula section:

Diagnosis window calculation:

If 0 <= XXPWM < C_XXPWM_DIAG_MIN_SCB
Then bit 1 of CDN_DIAG_XX = 1
Diagnosis of SCG is valid

Else **If** XXPWM < C_XXPWM_DIAG_MIN_OC
Then bit 0 and bit 1 of CDN_DIAG_XX = 1
Diagnosis of SCB, SCG is valid

Else **If** XXPWM <= C_XXPWM_DIAG_MAX_SCG
Then bit 0, bit 1 and bit 2 of CDN_DIAG_XX = 1
Diagnosis of SCB, SCG and OC is valid
Endif
Endif
Endif

A.106.2.1.2 "Limited" duty cycle

Description:

For an output with a "limited" duty cycle range, the diagnosis result of a symptom can always be considered as valid, because of the current duty cycle. It only can be invalid because special conditions managed by the BSW.

Application conditions:

Initialization: -

Recurrence: -

Activation: every ECU state

reactivation in the NC_PSD_DLY recurrence after LV_INH_DIAG_XX 1->0 and LV_IGK 0->1 and reset and FMY reset

Deactivation: LV_INH_DIAG_XX = 1

at deactivation: CDN_DIAG_XX = 0

Formula section:

bit 0, bit 1 and bit 2 of CDN_DIAG_XX = 1

Diagnosis of SCB, SCG and OC is valid

A.106.2.2 Diagnosis symptom (raw value) calculation

FUNCTION DESCRIPTION:

Application conditions:

Initialization: -

Recurrence: -

Activation: every ECU state

reactivation in the NC_PSD_DLY recurrence after LV_INH_DIAG_XX 1->0 and LV_IGK 0->1 and reset and FMY reset

Deactivation: LV_INH_DIAG_XX = 1

Formula section:

Symptom calculation for 'Short to battery' SCB:

If 'Failure symptom SCB detected by HW' (BSW information)

Then bit 0 of ERR_DIAG_XX = 1

Else bit 0 of ERR_DIAG_XX = 0

Endif

Symptom calculation for 'Short to ground' SCG:

If 'Failure symptom SCG detected by HW' (BSW information)

Then bit 1 of ERR_DIAG_XX = 1

Else bit 1 of ERR_DIAG_XX = 0

Endif

Symptom calculation for 'Open circuit' OC:

If 'Failure symptom OC detected by HW' (BSW information)

Then bit 2 of ERR_DIAG_XX = 1

Else bit 2 of ERR_DIAG_XX = 0

Endif

Diagnosis condition from BSW verification:

Diagnosis condition calculation for symptom 'Short to battery' SCB:

If 'Diagnosis of symptom SCB is not valid' (BSW information)

Then bit 0 of CDN_DIAG_XX = 0

Endif

Diagnosis condition calculation for symptom 'Short to ground' SCG:

If 'Diagnosis of symptom SCG is not valid' (BSW information)

Then bit 0 of CDN_DIAG_XX = 0

Endif

Diagnosis condition calculation for symptom 'Open circuit' OC:

If 'Diagnosis of symptom OC is not valid' (BSW information)

Then bit 0 of CDN_DIAG_XX = 0

Endif

ACTION_ERRM_FilterMulticondition (*XX*, CDN_DIAG_XX, ERR_DIAG_XX, LV_END_DIAG_XX, LV_ERR_XX, LV_CDN_DIAG_XX, ERR_SYM_XX)

Remark: For failure debouncing and error management treatment the algorithm Multi-condition debounce algorithm (part of Error Management AGGR) is used with the parameters CDN_DIAG_XX and ERR_DIAG_XX.

CDN_DIAG_XX = 0 (after failure debouncing and error management treatment)

A.106.3 Diagnosis of special outputs

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ERR_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis symptom (raw value) for each symptom for XX					
CDN_DIAG_XX	-	0 ... 7H	0 ... 7	1	-
Diagnosis condition for each symptom for XX					

Input data:

LV_IGK	ERR_DIAG_XX	CDN_DIAG_XX	

Import actions:

ACTION_ERRM_FilterMulticondition(IN< *XX* >, IN< CDN_DIAG_XX>, IN< ERR_DIAG_XX >, OUT< LV_END_DIAG_XX >, OUT< LV_ERR_XX >, OUT< LV_CDN_DIAG_XX >, OUT< ERR_SYM_XX >)

General information:

The purpose is to diagnose electrical errors detected by the hardware for special outputs. Special outputs are outputs, which cannot be treated like "normal" static or pwm outputs, like described in the previous chapters.

Description:

The diagnosis condition calculation for the driver output is done in the diagnosis function of the diagnosis instance XX. It's displayed for each symptom within the value CDN_DIAG_XX. Also the diagnosis symptom (raw value) calculation (ERR_DIAG_XX) is performed in the diagnosis function for special outputs.

The calculation of LV_ERR_XX, LV_CDN_DIAG_XX, ERR_SYM_XX and LV_END_DIAG_XX is done in the Multi-condition debounce algorithm (part of Error Management AGGR).

Application conditions:

Initialization: -

Recurrence: -

Activation: every ECU state

reactivation in the NC_PSD_DLY recurrence after CDN_DIAG_XX > 0 and LV_IGK 0->1 and reset and FMY reset

Deactivation: -

Formula section:

Diagnosis condition / Diagnosis symptom (raw value) calculation:

CDN_DIAG_XX and ERR_DIAG_XX are calculated in the diagnosis function.

ACTION_ERRM_FilterMulticondition (*XX*, CDN_DIAG_XX, ERR_DIAG_XX, LV_END_DIAG_XX, LV_ERR_XX, LV_CDN_DIAG_XX, ERR_SYM_XX)

Remark: For failure debouncing and error management treatment the algorithm Multi-condition debounce algorithm (part of Error Management AGGR) is used with the parameters CDN_DIAG_XX and ERR_DIAG_XX.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_PSD_DLY	1	0 ... FFH	0 ... 255	1	-
The diagnosis is reactivated after (LV_INH_DIAG_XX 1->0 or CDN_DIAG_XX >0 (depending on case)) and LV_IGK 0->1 and reset and FMY reset after NC_PSD_DLY recurrences.					

A.107 Battery Sensor diagnostic

A.107.1 Sensor diagnostic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_BAT_SENS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
For each symptom : status of failure (set to 1 when failure symptom detected)					
LV_CDN_DIAG_BAT_SENS	V	0...1H	0...1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_END_DIAG_BAT_SENS	V	0...1H	0...1	1	[-]
Diagnostic performed at least one time					
LV_ERR_BAT_SENS	O/V	0...1H	0...1	1	[-]
Present failure after filtering of battery sensor diagnostic					
CTR_DLY_BAT_SENS_ENA	V	0...FFH	0...25.5	0.1	[s]
Counter for battery sensor information availability delay					

Input data:

CONF_BAT	CONF_BAT_LIN	LV_DFT_SENS_BAT_LIN	LV_SENS_BAT_LIN_FIRST_VLD
LV_SENS_BAT_LIN_VLD			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_BAT_SENS	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_BAT_SENS	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					
C_DLY_BAT_SENS_ENA	1	0...FFH	0...25.5	0.1	[s]
Maximum time allowed to have the battery sensor information available					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_IDX_DIAG_BAT_SENS	1	1...FFFFH	1...65535	1	[-]
Failure index for battery sensor diagnostic					

Import actions:

ACTION_ERRM_FilterSymptom(IN<IDX_DIAG>, IN< LV_CDN_DIAG_XX >, IN< ERR_SYM_XX >, IN< C_ABC_INC_XX >, IN< C_ABC_MAX_XX >, IN< C_ABC_DEC_XX >, OUT< LV_ERR_XX >)

Error treatment

Diagnostic Instance		Symptoms		ABC type	CARB class
Description	Name (XX)	Description	Nr		
Battery sensor diagnosis	BAT_SENS	Time out after reset	0	MEM	NO
		Invalid message	1		
		Error response	2		
		-	3		

FUNCTION DESCRIPTION:

General information:

Error detection on the battery sensor depends of the smart battery sensor message received and the time to get a message valid after power on. The MEM type filter is used to avoid vehicle effect in case of intermittent error.

Description:

Error-symptoms are defined to this diagnosis function as following

ERR_SYM_BAT_SENS

```
□□□□
│ │ │ └─> Time out after reset          (= SYM_0)
│ │ └───> Invalid message               (= SYM_1)
│ └─────> Error response                (= SYM_2)
└───────> -                             (= SYM_3)
```

Application conditions:

Initialization at reset :

CTR_DLY_BAT_SENS_ENA = C_DLY_BAT_SENS_ENA

The other standard diagnostic data are initialized according filtering configuration.

Recurrence: 100 ms

Activation: CONF_BAT = 1 **and** CONF_BAT_LIN = 2 (Hella sensor)

Deactivation: When activation function is not filling in

Formula section:

Diagnostic condition calculation :

```
If  CTR_DLY_BAT_SENS_ENA  > 0
Then
     CTR_DLY_BAT_SENS_ENA  = CTR_DLY_BAT_SENS_ENA  - 1
     LV_CDN_DIAG_BAT_SENS=0
Else
     LV_CDN_DIAG_BAT_SENS=1
Endif
```

Symptoms calculation :

```
If(1)  LV_CDN_DIAG_BAT_SENS=1
Then(1)

     { SYM_2 : Error response  }
     If (2)    LV_DFT_SENS_BAT_LIN=1
     Then(2)
               ERR_SYM_BAT_SENS = SYM_2
      Else(2)

          { SYM_1 : Invalid message }
          If(3)         LV_SENS_BAT_LIN_VLD=0
          Then(3)
                        ERR_SYM_BAT_SENS = SYM_1
          Else(3)

             { SYM_0 : Time out after reset }
             If(4)      LV_SENS_BAT_LIN_FIRST_VLD=0
             Then(4)
                        ERR_SYM_BAT_SENS = SYM_0
             Else(4)
                        ERR_SYM_BAT_SENS = NO_SYM { No failure detected  }
             Endif(4)
          Endif(3)
     Endif(2)
Else(1)
     ERR_SYM_BAT_SENS unchanged
Endif(1)
```

For failure and error management treatment the anti-bounce mechanism is called :

ACTION_ERRM_FilterSymptom(IN<NC_IDX_DIAG_BAT_SENS>, IN<LV_CDN_DIAG_BAT_SENS>, IN<ERR_SYM_BAT_SENS>, IN<C_ABC_INC_BAT_SENS>, IN<C_ABC_MAX_BAT_SENS>, IN<1>, OUT<LV_ERR_BAT_SENS>)

A.107.2 System voltage plausibility diagnostic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_VB_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
For each symptom : status of failure (set to 1 when failure symptom detected)					
LV_CDN_DIAG_VB_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled)					
LV_END_DIAG_VB_PLAUS	V	0...1H	0...1	1	[-]
Diagnostic performed at least one time					
LV_ERR_VB_PLAUS	O/V	0...1H	0...1	1	[-]
Present failure after filtering of system voltage plausibility diagnostic					
CTR_DIAG_VB_PLAUS	V	0..FFH	0…25.5	0.1	[s]
Counter to manage the plausibility diagnostic duration					
CTR_DIAG_VB_PLAUS_OK	V	0...FFH	0…255	1	[-]
Counter of number time battery voltage conform					

Input data:

CONF_BAT	CONF_BAT_LIN	VB	VB_H_MMV
LV_ERR_BAT_SENS	LV_END_DIAG_BAT_SENS	LV_CDN_VB_OBD1	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_VB_PLAUS	1	0...FFH	0...255	1	[-]
Antibounce counter increment					
C_ABC_MAX_VB_PLAUS	1	1...FFH	1...255	1	[-]
Maximum value for antibounce counter					
C_VB_H_MMV_DELTA_MAX	1	0... 3FFH	0... 26	0.0254154	[V]
Delta voltage to detect battery system voltage plausibility failure					
C_DLY_VB_PLAUS	1	0..FFH	0…25.5	0.1	[s]
Delay to determinate plausibility diagnostic failure (typical value 10)					
C_CTR_DIAG_VB_PLAUS_OK	1	1..FFH	1…255	1	[-]
Number minimum of time the battery voltage test has to be good (typical value 3)					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_IDX_DIAG_VB_PLAUS	1	1...FFFFH	1...65535	1	[-]
Failure index for system voltage plausibility diagnostic					

Import actions:

ACTION_ERRM_FilterSymptom(IN<IDX_DIAG>, IN< LV_CDN_DIAG_XX >, IN< ERR_SYM_XX >, IN< C_ABC_INC_XX >, IN< C_ABC_MAX_XX >, IN< C_ABC_DEC_XX >, OUT< LV_ERR_XX >)

Error treatment

Diagnostic Instance		Symptoms		ABC type	CARB class
Description	Name (XX)	Description	Nr		
System voltage plausibility diagnostic	VB_PLAUS	-	0	MEM	NO
		-	1		
		-	2		
		System voltage plausibility	3		

FUNCTION DESCRIPTION:

General information:

Error detection in case of there is a different voltage between battery voltage measured on the ECM and received from the LIN battery sensor.

Description:

Error-symptoms are defined to this diagnosis function as following :

ERR_SYM_VB_PLAUS

```
□□□□
│ │ │ └─> -                              (= SYM_0)
│ │ └───> -                              (= SYM_1)
│ └─────> -                              (= SYM_2)
└───────> System voltage plausibility    (= SYM_3)
```

Application conditions:

Initialization at reset, at key transition and at CLRFMY event :

CTR_DIAG_VB_PLAUS = C_DLY_VB_PLAUS
CTR_DIAG_VB_PLAUS_OK =0
The other diagnostic data are initialized according filtering configuration.

Recurrence: 100 ms

Activation: CONF_BAT = 1 **and** CONF_BAT_LIN = 2

Deactivation: *never*

Formula section:

Diagnostic condition calculation :

{ The diagnostic condition is set when the sensor is a smart sensor, the battery sensor LIN diagnostic is passed and battery voltage OBD1 condition fullfil }

```
if        LV_ERR_BAT_SENS = 0
    and   LV_END_DIAG_BAT_SENS = 1
    and   LV_CDN_DIAG_OBD1 = 1
Then

   { When the two battery voltage values are very different }
   If     |VB - VB_H_MMV| <= C_VB_H_MMV_DELTA_MAX
   Then
          CTR_DIAG_VB_PLAUS_OK = CTR_DIAG_VB_PLAUS_OK + 1
   Endif

   { When the delay is elapsed the condition is set }
   CTR_DIAG_VB_PLAUS = CTR_DIAG_VB_PLAUS - 0.1s

   If     CTR_DIAG_VB_PLAUS=0
   Then
          LV_CDN_DIAG_VB_PLAUS=1
   Else
          LV_CDN_DIAG_VB_PLAUS=0
   Endif
Else
    { When the condition is not fill in the diagnostic is re-initialized }
    LV_CDN_DIAG_VB_PLAUS = 0
    CTR_DIAG_VB_PLAUS = C_DLY_VB_PLAUS
    CTR_DIAG_VB_PLAUS_OK = 0
Endif
```

Symptom calculation :

If(1) LV_CDN_DIAG_VB_PLAUS=1
Then(1)

{ When the counter of good diagnostic is too low the failure is set }
If CTR_DIAG_VB_PLAUS_OK < C_CTR_DIAG_VB_PLAUS_OK
Then
ERR_SYM_VB_PLAUS = SYM_3
Else
ERR_SYM_VB_PLAUS = NO_SYM
Endif

{ The diagnostic is re-initialized }
CTR_DIAG_VB_PLAUS = C_DLY_VB_PLAUS
CTR_DIAG_VB_PLAUS_OK =0
Else(1)
ERR_SYM_VB_PLAUS unchanged
Endif(1)

For failure and error management treatment the anti-bounce mechanism is called :

ACTION_ERRM_FilterSymptom(IN<NC_IDX_DIAG_VB_PLAUS>, IN<LV_CDN_DIAG_VB_PLAUS>, IN<ERR_SYM_VB_PLAUS>, IN<C_ABC_INC_VB_PLAUS>, IN<C_ABC_MAX_VB_PLAUS>, IN<1>, OUT<LV_ERR_VB_PLAUS>)

A.108 Port Flap Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PORT	V/O	0...1H	0...1	1	-
Error currently present on port flap					
PORTPWM_DIAG	V/O	8000...7FFFH	-100...99.99	0.003	%
Port flap pulse width modulation in case of port flap error					
T_PORT_DLY3	V	0...FFH	2.55	0.01	s
Delay time before Port Flap Pulse width modulation is switched off in case of port flap error					

Input data:

LV_ERR_PORT_POTI	LV_ERR_PORT_AD	LV_ERR_PORT_STUCK_CLOSE	LV_IGK
PORTPWM	LV_ERR_PORT_RNG	LV_ERR_VCC_SENS_SUB	CONF_PORT
LV_ERR_PORT_EL		LV_ERR_PORT_STUCK_OPEN	

FUNCTION DESCRIPTION:

Application conditions:

Initialization: At reset or at transition LV_IGK 0 -> 1:

LV_ERR_PORT = 0

PORTPWM_DIAG = 0%

Recurrence: 10 ms

Activation: at CONF_PORT = 1 **and** LV_IGK = 1

Deactivation: at LV_IGK = 0 **or** CONF_PORT = 0

General information:

The Port flap diagnosis is performed by electrical diagnosis for Power Stage and Potentiometer and Plausibility diagnosis.

The electrical diagnosis of the Potentiometer is divided in two symptoms:

- Signal line „shorted to Plus“or „line broken“
- Signal line „short to Ground“

There are four plausibility diagnosis :

- Adaptation failed
- Flap stuck opened
- Flap stuck closed
- Flap moves to slow or to unplausible range (Range/Performance)

Formula section:

Global error flag for PORT diagnosis:

if(1) LV_ERR_PORT_POTI = 1
or LV_ERR_PORT_STUCK_OPEN = 1
or LV_ERR_PORT_STUCK_CLOSE = 1
or LV_ERR_PORT_RNG = 1
or LV_ERR_PORT_EL = 1
or LV_ERR_PORT_AD = 1
or LV_ERR_VCC_SENS_SUB = 1
then(1) LV_ERR_PORT(n) = 1
else(1) LV_ERR_PORT(n) = 0
endif(1)

Moment of general port flap error setting:

if(1) LV_ERR_PORT(n-1) = 0
and LV_ERR_PORT(n) = 1
then(1) T_PORT_DLY3 = C_T_PORT_DLY3
endif(1)

Port flap pulse width modulation for limp home position:

if(1) LV_ERR_PORT(n) = 1
then(1) if(2) T_PORT_DLY3 = 0
then(2) PORTPWM_DIAG = 0 %
else(2) PORTPWM_DIAG = C_PORTPWM_DIAG
decrement T_PORT_DLY3 by 1H every 10ms
endif(2)
else(1) PORTPWM_DIAG = 0%
endif(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PORTPWM_DIAG	1	8000...7FFFH	-100...99.99	0.003	%
port flap pulse width modulation to reach Limp home position					
C_T_PORT_DLY3	1	0...FFH	0...2.55	0.01	s
Delay time before Port Flap Pulse width modulation is switched off in case of port flap error					

A.108.1 Potentiometer Diagnosis (LV_ERR_PORT_POTI)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PORT_POTI	V/O	0...1H	0...1	1	-
Error currently present on potentiometer of Port flap					
LV_CDN_DIAG_PORT_POTI	V/O	0..1H	0..1	1	-
Diagnosis condition PORT POTI diagnosis					
LV_END_DIAG_PORT_POTI	V/O	0...1 H	0...1	1	-
End of diagnosis PORT_POTI diagnosis					
ERR_SYM_PORT_POTI	V/O	0H 1H 2H 4H 8H	NO_SYM SYM 0 SYM 1 SYM 2 SYM 3	1	-
Error Symptom PORT POTI diagnosis					

Input data:

V_PORT	LV_CDN_VB_OBD1	LV_IGK	CONF_PORT
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the analog input signal from the PORT-potentiometer to the microcontroller.

Error-symptoms: are defined to this diagnosis function as following

```
□□□□
 │ │ │ └─> Short circuit to Plus or Line break      (= SYM_0)
 │ │ └───> Short circuit to GND                     (= SYM_1)
 │ └─────> -                                        (= SYM_2)
 └───────> -                                        (= SYM_3)
```

Application conditions:

Initialisation: at Reset, at LV_IGK= 0--> 1, set all variables = 0

Recurrence: *10 ms*

Activation:
if LV_IGK = 1
and LV_CDN_VB_OBD1 = 1
and CONF_PORT = 1
and LV_ERR_VCC_SENS_SUB = 0
then LV_CDN_DIAG_PORT_POTI = 1
else LV_CDN_DIAG_PORT_POTI = 0
endif

Deactivation: -

Formula section:

if(1) V_PORT > C_V_MAX_PORT_DIAG

then(1) ERR_SYM_PORT_POTI = "SYM_0"

symptom "line shorted to Plus" or "line broken" is active

LV_ERR_PORT_POTI = 1 (after debounce)

else(1)

if(2) V_PORT < C_V_MIN_PORT_DIAG

then(2) ERR_SYM_PORT_POTI = "SYM_1"

symptom "line shorted to Ground" is active

LV_ERR_PORT_POTI = 1 (after debounce)

else(2) ERR_SYM_PORT_POTI = "NO_SYM"

endif(2)

endif(1)

The bit LV_ERR_PORT_POTI is set irreversible, reset at transition LV_IGK 0 -> 1

Calculation of LV_END_DIAG_PORT_POTI and antibounce counter:
- see chapter " Anti bounce Algorithm"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_MIN_PORT_DIAG	1	0...3FFH	0...5	0.0048	V
Voltage threshold for detection of PORT-poti signal line shorted to ground					
C_V_MAX_PORT_DIAG	1	0...3FFH	0...5	0.0048	V
Voltage threshold for detection of PORT-poti signal line shorted to battery voltage or broken					
C_ABC_INC_PORT_POTI	1	0...FFH	0...255	1	-
Antibounce Counter Increment for Port Potentiometer Diagnosis					
C_ABC_MAX_PORT_POTI	1	0...FFH	0...255	1	-
Maximum value of Antibounce Counter Increment for Port Potentiometer Diagnosis					

A.108.2 Mechanical Diagnosis: Port Flap stuck closed

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_PORT_STUCK_CLOSE	V/O	0...1H	0...1	1	-
PORT_Flap stuck in Closed Position (after debounce)					
LV_CDN_DIAG_PORT_STUCK_CLOSE	V/O	0..1H	0..1	1	-
Diagnosis condition PORT_STUCK_CLOSE diagnosis					
LV_END_DIAG_PORT_STUCK_CLOSE	V/O	0...1 H	0...1	1	-
End of diagnosis PORT_STUCK_CLOSE diagnosis					
ERR_SYM_PORT_STUCK_CLOSE	V/O	0H 1H 2H 4H 8H	NO_SYM SYM 0 SYM 1 SYM 2 SYM 3	1	-
Error Symptom PORT_STUCK_CLOSE diagnosis					
T_PORT_DLY4	V/O	0...FFH	0..2.55	0.01	s
timer for Port Stuck Closed diagnosis					

Input data:

LV_IGK	LV_CDN_VB_OBD2	PORT_SP	CONF_PORT
LV_ERR_PORT_POTI	V_PORT	LV_PORT_AD_VLD	LV_ERR_PORT_EL
LV_ERR_VCC_SENS_SUB			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose whether the port flap stucks in closed position. This is performed with the Potentiometer value.

If the Port flap is required to open a timer T_PORT_DLY4 is decremented. If the counter reaches 0 and the Potentiometer value is still near the adapted Closed Position, then the Port Flap is determined to stuck closed.

Error-symptoms: are defined to this diagnosis function as following

```
□□□□
| | | └─> Port Flap stuck close          (= SYM_0)
| | └───> -                              (= SYM_1)
| └─────> -                              (= SYM_2)
└───────> -                              (= SYM_3)
```

Application conditions:

Initialization: At reset, at transition LV_IGK 0 -> 1:

T_PORT_DLY4 = C_T_PORT_DLY4

all other variables = 0

Recurrence: 10 ms

Activation: **if** LV_ IGK = 1

and CONF_PORT = 1

and LV_PORT_AD_VLD = 1

Formula section:

if(1) PORT_SP = 0
and LV_CDN_VB_OBD2 = 1
and LV_ERR_PORT_POTI = 0
and LV_ERR_PORT_EL = 0
and LV_ERR_VCC_SENS_SUB = 0
then(1) if(2) T_PORT_DLY4 = 0
then(2) LV_CDN_DIAG_PORT_STUCK_CLOSE = 1
Diagnosis is available
if(3) V_PORT > C_V_PORT_STUCK_CLOSE
then(3) ERR_SYM_PORT_STUCK_CLOSE = "SYM_0"
symptom "Port flap stuck closed" is active
else(3) ERR_SYM_PORT_STUCK_CLOSE = "NO_SYM"
endif(3)
else(2) *Diagnose is not available*
LV_CDN_DIAG_PORT_STUCK_CLOSE = 0
T_PORT_DLY4 = T_PORT_DLY4 - 1H every 10 ms
endif(2)
else(1) T_PORT_DLY4 = C_T_PORT_DLY4
LV_CDN_DIAG_PORT_STUCK_CLOSE = 0
Diagnose is not available
endif(1)

After debounce, the bit LV_ERR_PORT_STUCK_CLOSE is set to 1

Calculation of LV_END_DIAG_PORT_STUCK_CLOSE and antibounce counter:
- see chapter " Anti bounce Algorithm"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_PORT_DLY4	1	0...FFH	0..2.55	0.01	s
timer for mechanical diagnosis Stuck Close					
C_V_PORT_ STUCK_CLOSE	1	0...3FFH	0...5	0.0048	V
V_PORT threshold to detect a stucked close Port Flap					
C_ABC_INC_PORT_STUCK_CLOSE	1	0...FFH	0...255	1	-
Antibounce Counter Increment for Port Stuck Close Diagnosis					
C_ABC_MAX_PORT_STUCK_CLOSE	1	0...FFH	0...255	1	-
Maximum value of Antibounce Counter Increment for Port Stuck Close Diagnosis					

A.108.3 Mechanical Diagnosis: Port Flap stuck open

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_PORT_STUCK_OPEN	V/O	0...1H	0...1	1	-
PORT_Flap stucks in OPEN Position (after debounce)					
LV_CDN_DIAG_PORT_STUCK_OPEN	V/O	0..1H	0..1	1	-
Diagnosis condition PORT_STUCK_OPEN diagnosis					
LV_END_DIAG_PORT_STUCK_OPEN	V/O	0...1 H	0...1	1	-
End of diagnosis PORT_STUCK_OPEN diagnosis					
ERR_SYM_PORT_STUCK_OPEN	V/O	0H 1H 2H 4H 8H	NO_SYM SYM 0 SYM 1 SYM 2 SYM 3	1	-
Error Symptom PORT Stuck OPEN diagnosis					
T_PORT_DLY5	V/O	0...FFH	0..2.55	0.01	s
timer for Port Stuck OPEN diagnosis					

Input data:

LV_IGK	LV_CDN_VB_OBD2	PORT_SP	CONF_PORT
LV_ERR_PORT_POTI	V_PORT	LV_PORT_AD_VLD	LV_ERR_PORT_EL
LV_ERR_VCC_SENS_SUB	LV_ERR_PORT_RNG	LV_ERR_PORT_STUCK_CLOSE	

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose wheter the Port flap stucks in opened position. This is performed with the Potentiometer value.

If the Port flap is required to close a timer T_PORT_DLY5 is decremented. If the timer reaches 0 and the Potentiometer value is still near the adapted Open position, then the Port flap is determined to stuck open. As the H-bridge output diagnosis cannot detect open circuit this can be also a symtom for an open circuit at supply line.

Error-symptoms: are defined to this diagnosis function as following

```
[ ][ ][ ][ ]
 |  |  |  └─> Port Flap stuck open or open circuit of DC-motor supply   (= SYM_0)
 |  |  └────> -                                                        (= SYM_1)
 |  └───────> -                                                        (= SYM_2)
 └──────────> -                                                        (= SYM_3)
```

Application conditions:

Initialization: At reset, at transition LV_IGK 0 -> 1:

T_PORT_DLY5 = C_T_PORT_DLY5

all other variables = 0

Recurrence: 10 ms

Activation: **if** LV_ IGK = 1

and CONF_PORT = 1

and LV_PORT_AD_VLD = 1

Formula section:

if(1) PORT_SP = 1

and LV_CDN_VB_OBD2 = 1

and LV_ERR_PORT_POTI = 0

and LV_ERR_PORT_EL = 0

and LV_ERR_PORT_RNG = 0

and LV_ERR_PORT_STUCK_CLOSE = 0

and LV_ERR_VCC_SENS_SUB = 0

then(1) if(2) T_PORT_DLY5 = 0

then(2) LV_CDN_DIAG_PORT_STUCK_OPEN = 1

Diagnosis is available

if(3) V_PORT < C_V_PORT_STUCK_OPEN

then(3) ERR_SYM_PORT_STUCK_OPEN = "SYM_0"

symptom "Port flap stuck open or open circuit at DC-motor supply " is active

else(3) ERR_SYM_PORT_STUCK_OPEN = "NO_SYM"

endif(3)

else(2) LV_CDN_DIAG_PORT_STUCK_OPEN = 0

Diagnose is not available

T_PORT_DLY5 = T_PORT_DLY5 - 1H every 10 ms

endif(2)

else(1) LV_CDN_DIAG_PORT_STUCK_OPEN = 0

T_PORT_DLY5 = C_T_PORT_DLY5

Diagnose is not available

endif(1)

After debounce, the bit LV_ERR_PORT_STUCK_OPEN is set to 1

Calculation of LV_END_DIAG_PORT_STUCK_OPEN and antibounce counter:
see chapter " Anti bounce Algorithm"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_PORT_DLY5	1	0...FFH	0..2.55	0.01	s
timer for PORT_STUCK_OPEN diagnosis					
C_V_PORT_ STUCK_OPEN	1	0...3FFH	0...5	0.0048	V
V_PORT threshold to detect stucked open Port Flap					
C_ABC_INC_PORT_STUCK_OPEN	1	0...FFH	0...255	1	-
Antibounce Counter Increment for PORT stuck OPEN Diagnosis					
C_ABC_MAX_PORT_STUCK_OPEN	1	0...FFH	0...255	1	-
Maximum value of Antibounce Counter Increment for Port stuck OPEN Diagnosis					

A.108.4 Mechanical diagnosis: Port Flap Range/Performance

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_PORT_RNG	V/O	0...1H	0...1	1	-
PORT_Flap cannot Range Performance error (after debounce)					
LV_CDN_DIAG_PORT_RNG	V/O	0..1H	0..1	1	-
Diagnosis condition PORT_RNG diagnosis					
LV_END_DIAG_PORT_RNG	V/O	0...1 H	0...1	1	-
End of diagnosis PORT_RNG diagnosis					
ERR_SYM_PORT_RNG	V/O	0H 1H 2H 4H 8H	NO_SYM SYM 0 SYM 1 SYM_2 SYM_3	1	-
Error Symptom PORT_RNG diagnosis					

Input data:

PORT_SP	V_PORT		CONF_PORT
LV_ERR_PORT_POTI	C_V_PORT_STUCK_OPEN	LV_PORT_AD_VLD	LV_ERR_PORT_EL
LV_ERR_VCC_SENS_SUB	C_V_PORT_STUCK_CLOSE	T_PORT_DLY4	T_PORT_DLY5
PORTPWM_DIAG			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose wheter the Port flap is moving to slow or if the shaft is not connected properly to the Port Flap motor. This is performed with the Potentiometer value.

If a Port flap setpoint change is required and the related timer is decremented to 0 then the Potentiometer value is checked. If the value is between the learned open and closed value, then this indicates in a slow moving Port Flap or stucked Flap (Setpoint could not be reached in the required time), if the value is below the learned open or above the learned close value, this is an indication that the Flap shaft is not connected properly to the DC motor unit.

Error-symptoms: are defined to this diagnosis function as following

```
□□□□
│ │ │ └─> Port Flap moves too slow or stucked between open and close   (= SYM_0)
│ │ └───> Port Flap shaft not connected properly to DC-Motor unit       (= SYM_1)
│ └─────> -                                                             (= SYM_2)
└───────> -                                                             (= SYM_3)
```

Application conditions:

Initialization:

Recurrence: 10 ms

Activation: **if** LV_ IGK = 1
and CONF_PORT = 1
and LV_PORT_AD_VLD = 1

Deactivation: -

Formula section:

```
if(1)
     and   LV_ERR_PORT_POTI = 0
     and   LV_ERR_PORT_EL = 0
     and   LV_ERR_VCC_SENS_SUB = 0
then(1) if(2)        PORT_SP = 0
             and     T_PORT_DLY4 = 0
        then(2)      LV_CDN_DIAG_PORT_RNG =1
        else(2) if(3)          PORT_SP = 1
                      and      T_PORT_DYL5 = 0
                then(3)        LV_CDN_DIAG_PORT_RNG = 1
                else(3)        LV_CDN_DIAG_PORT_RNG = 0
                endif(3)
        endif (2)
else(1) LV_CDN_DIAG_PORT_RNG = 0
endif(1)

if(1)      LV_CDN_DIAG_PORT_RNG = 1
then(1) if(2)     V_PORT < C_V_PORT_MIN_RNG
        then(2)   ERR_SYM_PORT_RNG = "SYM_1"
                  symptom " Port Flap shaft not connected properly to DC Motor" is active
        else(2)  if(3)         V_PORT >=  C_V_PORT_STUCK_OPEN
```

and V_PORT < = C_V_PORT_ STUCK_CLOSE

and PORTPWM_DIAG = 0%

(no error detection due to Limp Home reaction)

then(3) ERR_SYM_PORT_RNG = "SYM_0"

symptom "Port flap moves to slow or stucked between open and close"is active

else(3) ERR_SYM_PORT_RNG = "NO_SYM"

endif(3)

endif(2)

endif(1)

After debounce, the bit LV_ERR_PORT_RNG is set to 1

Calculation of LV_END_DIAG_PORT_RNG and antibounce counter:
see chapter " Anti bounce Algorithm"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_PORT_MIN_RNG	1	0...3FFH	0...5	0.0048	V
Port Flap voltage threshold to detect shaft not connected properly					
C_ABC_INC_PORT_RNG	1	0...FFH	0...255	1	-
Antibounce Counter Increment for Port RNG Diagnosis					
C_ABC_MAX_PORT_RNG	1	0...FFH	0...255	1	-
Maximum value of Antibounce Counter Increment for Port RNG Diagnosis					

A.108.5 Port Flap Adaptation diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PORT_AD	V/O	0...1H	0...1	1	-
Error currently present on Port flap, error symptom „Port adaptation“ (after debounce)					
LV_CDN_DIAG_PORT_AD	V/O	0..1H	0..1	1	-
Diagnosis condition PORT Adaption diagnosis					
LV_END_DIAG_PORT_AD	V/O	0...1 H	0...1	1	-
End of diagnosis PORT Adaption diagnosis					
ERR_SYM_PORT_AD	V/O	0H 1H 2H 4H 8H	NO_SYM SYM 0 SYM 1 SYM 2 SYM 3	1	-
Error Symptom PORT Adaption diagnosis					

Input data:

LV_PORT_AD_ERR	CONF_PORT	LV_IGK	LV_PORT_AD_VLD

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose the adaptation procedure of the Port flap.

Error-symptoms:

are defined to this diagnosis function as following

```
┌─┬─┬─┬─┐
│ │ │ └─> -                              (= SYM_0)
│ │ └───> -                              (= SYM_1)
│ └─────> -                              (= SYM_2)
└───────> Port Flap adaptation failed    (= SYM_3)
```

Freeze frame, DTC, error code number, symptom number, CARB-Info, Successive error management and data for MIL-management see chapters for general diagnosis informations.

Application conditions:

Initialisation: at reset, at transition LV_IGK 0 -> 1: set all variables = 0

Recurrency: 10 ms

Activation: **If** LV_IGK = 1
And CONF_PORT = 1

Deactivation:

Formula section:

if LV_PORT_AD_VLD = 1
or LV_PORT_AD_ERR = 1

then LV_CDN_DIAG_PORT_AD = 1

if LV_PORT_AD_ERR = 1

then ERR_SYM_PORT_AD = "SYM_3"

symptom "adaptation failed" is active

LV_ERR_PORT_AD = 1

else ERR_SYM_PORT_AD = "NO_SYM"

symptom "adaptation failed" is inactive

LV_ERR_PORT_AD = 0

endif

else LV_CDN_DIAG_PORT_AD = 0

endif

Calculation of LV_END_DIAG_PORT_AD and antibounce counter:
- see chapter " Anti bounce Algorithm"

A.108.6 Application incidences for Port flap diagnosis

A.108.6.1 Port Flap Potentiometer diagnosis

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Port Flap Potentiometer	Short Circuit to Battery or Open Circuit	SYM_0	*MEM*
	Short Circuit to Ground	SYM_1	
	-	SYM_2	
PORT_POTI	-	SYM_3	

A.108.6.2 Port Flap stucked in CLOSE Position

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Port Flap stucked in CLOSE position	Port Flap stuck close	SYM_0	*MEM*
		SYM_1	
	-	SYM_2	
PORT STUCK CLOSE	-	SYM_3	

A.108.6.3 Port Flap stucked in OPEN Position

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Port Flap stucked in Open position	Port Flap stuck open or open circuit of DC-Motor Supply	SYM_0	*MEM*
		SYM_1	
	-	SYM_2	
PORT STUCK OPEN	-	SYM_3	

A.108.6.4 Port Flap Range/Performance problem

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Port Flap Range/Performance	Port Flap moves to slow or stucked between open and close	SYM_0	*MEM*
	Port Flap shaft not connected properly to DC-Motor unit	SYM_1	
	-	SYM_2	
PORT RNG	-	SYM_3	

A.108.6.5 Port Flap Adaptation failure

Configuration for Diagnostic Symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Port Flap Adaptation	-	SYM_0	*MEM*
	-	SYM_1	
	-	SYM_2	
PORT_AD	Port Flap adaptation failed	SYM_3	

A.109 Port Flap: Electrical Power Stage Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PORT_EL	V/O	0...1H	0...1	1	-
Error currently present on electrical part of Port flap					
LV_CDN_DIAG_PORT_EL	V/O	0...1H	0...1	1	-
Diagnosis condition PORT_EL					
ERR_SYM_PORT_EL	V/O	0H 8H	NO_SYM SYM_3	1	-
LV_END_DIAG_PORT_EL	V/O	0...1H	0...1	1	-

Input data:

LV_IGK	LV_INH_DIAG_PORT_EL		

FUNCTION DESCRIPTION:

General information:

Description:

The following errors symptoms leads to an error. They can not be distinguished by the software:

- Undervoltage
- Overcurrent
- Overtemperature
- H-bridge hardware disable

Error-symptoms are defined to this diagnosis function as following :

```
┌──┬──┬──┬──┐
└──┴──┴──┴──┘
 │  │  │  └──> -                          (= SYM_0)
 │  │  └─────> -                          (= SYM_1)
 │  └────────> -                          (= SYM_2)
 └───────────> PORT power stage error     (= SYM_3)
```

Application conditions:

Initialisation: all to zero at LV_IGK = 0 --> 1 or reset

Recurrence: 10 ms

Activation: **if** LV_IGK **and**
LV_INH_DIAG_PORT_EL == 0
then LV_CDN_DIAG_PORT_EL = 1
else LV_CDN_DIAG_PORT_EL = 0
endif

Deactivation: When the activation condition is not fulfilled.

Formula section:

Call BSW routine to read out the error state lv_port_el_sf of the power stage !

If lv_port_el_sf *{Port H-bridge error detected }*
then ERR_SYM_PORT_EL = SYM_3
LV_ERR_PORT_EL = 1 { after debounce, set for this DC }
else ERR_SYM_PORT_EL = NO_SYM
endif

Calculation end of diagnosis:

LV_END_DIAG_PORT_EL is calculated by error management.

A.110 Port Flap power stage diagnosis (appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_PORT_EL	V/O	0h...1h	0...1	1	-
Inhibition condition for the power stage diagnosis					

Input data:

LV_IGK	LV_ES	LV_ST	VB

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_PORT_EL permit to deactivate the corresponding diagnostic.

Description:

Application conditions:

Initialisation: at LV_IGK = 0 → 1 or reset,
LV_INH_DIAG_PORT_EL = 1

Recurrence: 10 ms

Activation: LV_IGK == 1

Deactivation: When the activation condition is not fulfilled

Formula section:

LV_INH_DIAG_PORT_EL = **(**LV_ES = 1 **or**
LV_ST = 1 **or**
VB < C_VB_MIN_DIAG_PORT_EL**)**

Configuration for diagnostic symptoms:

Diagnostic PORT_EL	Symptoms	Nb	Filter type
PORT FLAP power stage diagnosis		*0*	*MEM*
		1	
		2	
	PORT FLAP power stage error	3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_PORT_EL	1	0..FFH	0...255	1	-
Debounce counter increment, Port Flap driver diagnosis					
C_ABC_MAX_PORT_EL	1	1..FFH	1...255	1	-
Debounce counter maximum value, Port Flap driver diagnosis					
C_VB_MIN_DIAG_PORT_EL	1	0 ... FFH	0 ... 26	0.102	V
Minimum voltage threshold for Port Flap diagnosis					

B OBD II functions

Table of Contents

Index

A

D

F

G

I

T

W

B.1 Misfire detection

B.1.1 Global fade out switch for misfire detection and monitoring

Description:

There is one global switch which can be used for inhibiting all the MISF aggregate modules:

- Generic misfire parameters & fade-out conditions : Aggr MISF, chapter OBDII
- Legal misfire detection fade-out conditions : Aggr MISF, chapter OBDII
- Rough road detection : Aggr MISF, chapter OBDII
- Crankshaft oscillation detection : Aggr MISF, chapter OBDII
- Misfire detection : Aggr MISF, chapter OBDII
- Appl. Inc. for misfire detection : Aggr MISF, chapter OBDII
- Misfire rate and criterions determination : Aggr MISF, chapter OBDII
- Appl. Inc. for Misfire rate and criterions determination : Aggr MISF, chapter OBDII

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
LC_MIS_INH	1	0...1H	0...1	1	[-]
Global switch to disable all misfire monitoring related modules (Inhibition with = 1)					

B.1.2 Misfire detection thresholds determination for engine roughness index

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
FAC_THD_ER	V	0...400H	0...4	3.9063e-3	[–]
Global gain multiplicative correction applied to basic misfire detection thresholds					
THD_ER	V/O	8000...7FFFH	-32768...32767	1	[µs]
Misfire detection threshold for current ER value					
THD_ER_BUF[NC_SIZE_THD_ER_BUF]	V	8000...7FFFH	-32768...32767	1	[µs]
Misfire detection threshold stack for ER evaluation					
THD_ER_CYL[NC_CYL_NR]	V	8000...7FFFH	-32768...32767	1	[µs]
Cylinder Misfire detection threshold for current ER value					

Input data:

N_32	TCO	LV_AT	FAC_THD_APP_ER
N	LV_IS	LOAD_MIS	INH_INJ
SEG_NR_ER	LV_ENA_SEG_T_MES	NC_CMB_CONF	LV_S_ACT
LV_HOM_AFL_ACT	THD_ER_CLC	LV_DRI	

Function description:

ER strategy misfire detection is based on a comparison between a cylinder specific ER value (nominal or normalised) and a threshold who is relative to the engine parameters when this same cylinder was in intake phase (threshold relative to combustion conditions image).

The engine roughness value ER depends strongly on the current operating point, both in its spread during undisturbed engine operation, and in the signal amplitude in case of an actual misfire.

ER misfire detection thresholds are defined according basic threshold maps and threshold corrections :

- *A basic threshold map:*

This map is function of engine load and speed which are calibrated for engine at operating temperature as well as for adapted transmitter gear. One map is assigned through the vehicle transmission type, idle speed condition & the combustion mode.

- *A threshold temperature correction:*

A temperature correction is applied to compensate larger signal amplitudes arise of ER values at lower engine temperatures. This correction is relative to TCO value.

- *A threshold correction during active injection valve cut-off:*

If one or several injection valves are cut off selectively by the engine management system (IV diagnoses ...), then the cylinder or cylinders are excluded from the misfire check. Due to the system, the signal amplitude of the ER value is reduced with a misfire at another cylinder. To improve the detection for this period, the absolute amount of the threshold can be reduced multiplicatively with the factor IP_FAC_THD_IV_ER(N_32) or IP_FAC_THD_IV_S_ER(N_32) *(in stratifed mode)*. *FAC_THD_IV_ER is a temporary data.*

- *A threshold correction defined in application incidences:*

FAC_THD_APP_ER is a threshold multiplicative correction who is defined in the misfire detection application incidences file.

B.1.2.1 Multiplicative correction for misfire thresholds

Application conditions:

Initialisation: *FAC_THD_IV_ER = 1*

Recurrence: every ENRD segment task

Activation: *LV_ENA_SEG_T_MES = 1*

And *LC_MIS_INH = 0*

Deactivation: *LV_ENA_SEG_T_MES = 0*

Or *LC_MIS_INH = 1*

Formula section:

#IF NC_CMB_CONF = AFS_S Or AFS_AFL_S

If INH_INJ = 0

Then FAC_THD_IV_ER = 1

ElseIf LV_S_ACT = 1

Then FAC_THD_IV_ER = IP_FAC_THD_IV_S_ER(N_32)

Else FAC_THD_IV_ER = IP_FAC_THD_IV_ER(N_32)

EndIf

#ELSE

If INH_INJ = 0

Then FAC_THD_IV_ER = 1

Else FAC_THD_IV_ER = IP_FAC_THD_IV_ER(N_32)

EndIf

#ENDIF

FAC_THD_ER = IP_FAC_THD_TCO_ER(TCO) * FAC_THD_IV_ER * FAC_THD_APP_ER

Remark : Given the update rate of TCO is 100ms, the value of ip_fac_thd_tco_er(tco) can be updated at this rate (CPU Load).

B.1.2.2 Misfire detection threshold buffers management

To manage the delay between cylinder intake phase and cylinder ER values, buffers are used to store threshold values before using for detection.

THD_ER_BUF[NC_SIZE_THD_ER_BUF] buffer is managed as a FIFO stack.

THD_ER_BUF[0] threshold value is the one obtained at current segment.

THD_ER is threshold value delayed and who will be used for detection with current segment ER value. This delay between ER value and cylinder corresponding threshold depends on engine cylinder number (NC_CYL_NR).

B.1.2.3 Misfire detection threshold based on engine roughness value (THD_ER)

By using the ER value for misfire detection, the engine speed has the greatest influence on basic thresholds definition. On the other hand, by using the ER_STND values for misfire detection, the engine load has the greatest influence on basic thresholds definition.

The influence of the engine load on the detection thresholds can change rapidly during instationary operation, it is urgently necessary to determine and to temporarily store the threshold values already at the time of the actual combustion before the calculatory check.

note: at each ER segment occurence, THD_ER_BUF is shifted with one memory unit (FIFO management) before determination process.

Application conditions:

Initialisation: at ECU reset **Or** engine stop

THD_ER = 0x8000

// ER Threshold buffer init on ER buffer reinitialisation

For k = 0 : NC_SIZE_THD_ER_BUF-1

THD_ER_BUF[k] = 0x8000 *// For all buffer cells*

EndFor

For x = 0 : NC_CYL_NR-1

THD_ER_CYL[x] = 0x8000 *// For all cylinders*

EndFor

Recurrence: every ENRD segment task

Activation: *LV_ENA_SEG_T_MES = 1*

***And** LC_MIS_INH = 0*

Deactivation: *LV_ENA_SEG_T_MES = 0*

***Or** LC_MIS_INH = 1*

Actions on Deactivation event:

THD_ER = 0x8000

// ER Threshold buffer init on ER buffer reinitialisation

For k = 0 : NC_SIZE_THD_ER_BUF-1

THD_ER_BUF[k] = 0x8000 *// For all buffer cells*

EndFor

For x = 0 : NC_CYL_NR-1

THD_ER_CYL[x] = 0x8000 *// For all cylinders*

EndFor

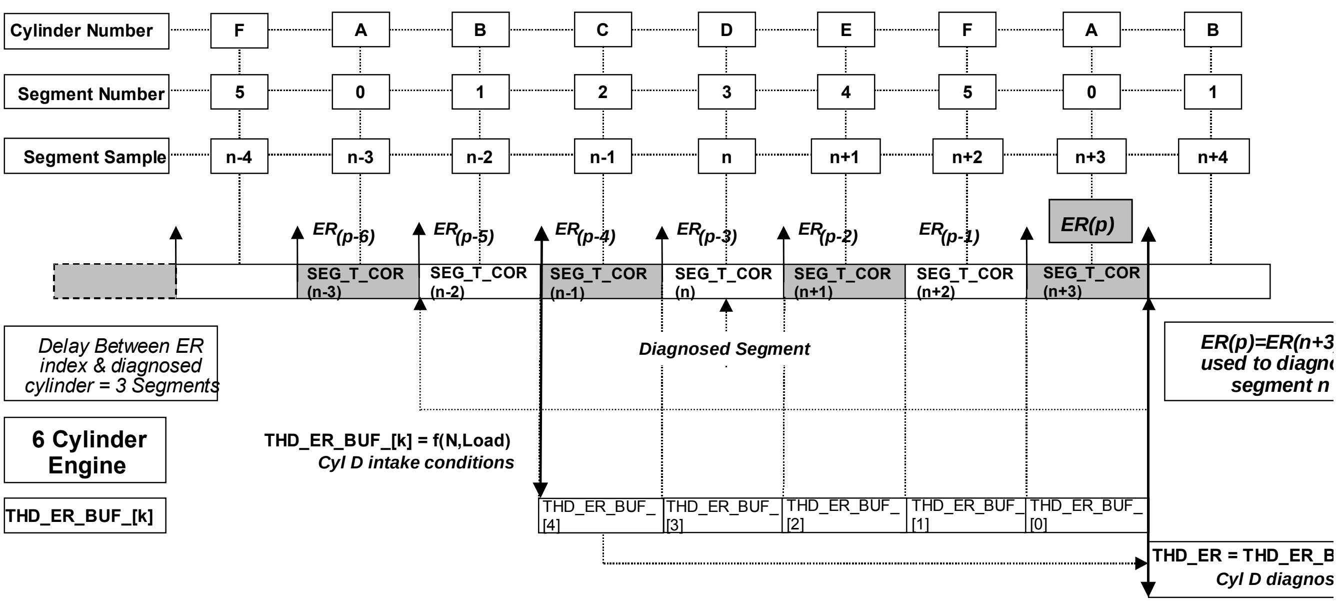
Formula section:

THD_ER_BUF[0] = THD_ER_CLC * FAC_THD_ER

THD_ER = THD_ER_BUF[NC_SIZE_THD_ER_BUF-1] *// Last cell of THD_ER buffer*

THD_ER_CYL[SEG_NR_ER] = THD_ER

6 cylinder engine example



Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
IP_FAC_THD_TCO_ER	8	0...1FFH	0...1.99609	3.9063e-3	[-]
LDP_TCO_IP_FAC_THD_TCO_ER	8	0...FEH	-48...142.5	0.75	[°C]
Temperature-dependent threshold reduction					
IP_FAC_THD_IV_ER	6	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_N_32_IP_FAC_THD_IV_ER	6	0...FFH	0...8160	32	[rpm]
Factor for adaptation of the threshold in case of active cylinder cut-off					
#IF NC_CMB_CONF = AFS_S Or AFS_AFL_S					
IP_FAC_THD_IV_S_ER	6	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_N_32_IP_FAC_THD_IV_S_ER	6	0...FFH	0...8160	32	[rpm]
Factor for adaptation of the threshold in case of active cylinder cut-off in stratified combustion mode					
#ENDIF					

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_SIZE_THD_ER_BUF	-	0...8H	0...8	1	[-]
Misfire detection threshold buffer size					

B.1.3 Misfire detection based on engine roughness evaluation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DET_MIS	O/V	0...1H	0...1	1	[-]
Single misfire detected and confirmed (after fade out test)					
LV_INH_DET_MIS	O/V	0...1H	0...1	1	[-]
General misfire detection fade out					
LV_DET_THD_FIL	V	0...1H	0...1	1	[-]
Single misfire detected based only on ER_FIL criterion (before fade out test)					
LV_DET_THD_MIS	V	0...1H	0...1	1	[-]
Single misfire detected based only on threshold criterion (before fade out test)					
MIS_STATE_ER	V	0...FFFH	0...4095	1	[-]
Status carrier byte of actual detected misfire through engine roughness index - 1 bit / cylinder					
MIS_DET_CYL_INH	V	0...FFFH	0...4095	1	[-]
Cylinder misfiring detection inhibition carrier byte when a fade out condition occurred (after masking & delay)					

Input data:

LV_DET_THD_DELTA	LV_ENA_ER	LV_DIAG_MIS	NC_CYL_NR
LV_ZDLY_DIAG_MIS	SEG_NR_ER	LV_DET_THD_DELTA_FIL_MEM	LV_REQ_INH_MIS
ER_FIL	NLC_CONF_ER_MV		

General information:

The system identifies misfiring by monitoring engine roughness index. Misfiring causes the angular velocity of the crankshaft to drop in an angular range specific to the cylinder in question.

Misfire detection is based on nominal engine roughness (ER) by using THD_ER detection threshold.

The identification of the cylinder(s) detected in misfire is realised according SEG_NR_ER segment reference (see definition and scheme in ENRD aggregate) *(Chapter system variables)*.

THD_ER is already phased according ER delay *(see detection thresholds management)*.

Application conditions:

Initialisation: at ECU reset, at Engine Stop ***Or*** *at Deactivation Event*

LV_DET_MIS = LV_DET_THD_MIS = LV_DET_THD_FIL = ~~LV_DET_THD_DELTA =~~ 0

LV_INH_DET_MIS = 1

MIS_STATE_ER = 0

MIS_DET_CYL_INH = $2^{NC\_CYL\_NR}$ - 1

Recurrence: every segment task

Activation: *LV_ENA_ER = 1*

And *LC_MIS_INH = 0*

Deactivation: *LV_ENA_ER = 0*

Or *LC_MIS_INH = 1*

Update rate:

ENRD segment task

Forumla section:

Misfire detection criterion

If (ER_FIL < THD_ER **)**

Then LV_DET_THD_FIL = 1

Else LV_DET_THD_FIL = 0

EndIf

If ((LV_DET_THD_FIL = 1) OR (LV_DET_THD_DELTA = 1)

#IF NLC_CONF_ER_MV =1

OR (LV_DET_THD_DELTA_FIL_MEM = 1)

#ENDIF

)

Then LV_DET_THD_MIS = 1

Else LV_DET_THD_MIS = 0

EndIf

General fade-out management

If LV_REQ_INH_MIS = 0 *// no fade-out request on going (legal and generic)*

And **(** LV_ZDLY_DIAG_MIS = 1 **Or** LV_DIAG_MIS = 1 **)** *// diagnosis phase started*

Then LV_INH_DET_MIS = 0

MIS_DET_CYL_INH[SEG_NR_ER] = 0

(corresponding cylinder bit in MIS_DET_CYL_INH structure is set to 0,
misfire detection will be enabled on this cylinder

Else LV_INH_DET_MIS = 1

MIS_DET_CYL_INH[SEG_NR_ER] = 1

(corresponding cylinder bit in MIS_DET_CYL_INH structure is set to 1,
misfire detection will be disabled on this cylinder

EndIf

Misfire detection & cylinder identification

If LV_DET_THD_MIS = 1

And LV_INH_DET_MIS = 0 *// no fade out on monitored cylinder*

Then LV_DET_MIS = 1 *// an instantaneous misfiring is detected*

MIS_STATE_ER[SEG_NR_ER] = 1

Cylinder specific bit in MIS_STATE_ER carrier structure is set to 1 according SEG_NR_ER segment reference.

Else LV_DET_MIS = 0

MIS_STATE_ER[SEG_NR_ER] = 0

Cylinder specific bit in MIS_STATE_ER carrier structure is set to 0 according SEG_NR_ER segment reference.

EndIf

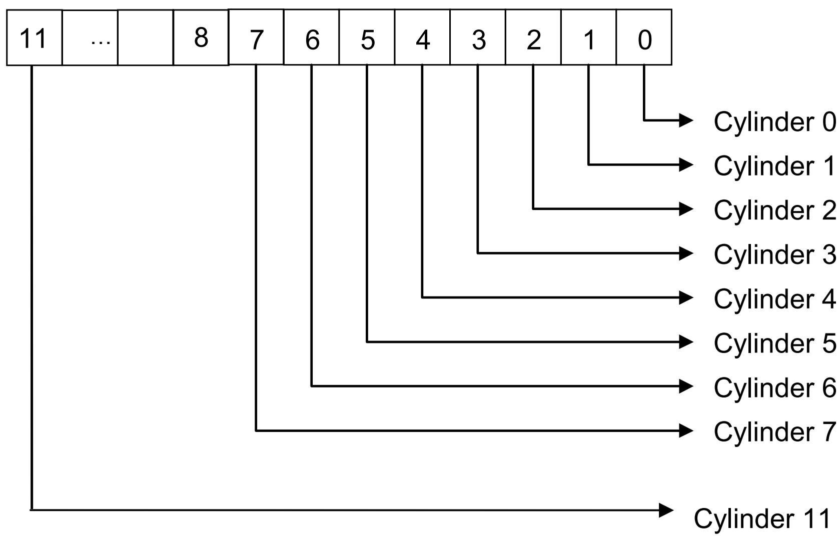
Misfire cylinder fade out carrier definition:

MIS_DET_CYL_INH : Carrier used for cylinder fade-out reference
Shows the cylinders that will be in fade-out during the detection phase

MIS_STATE_ER *:* Carrier used for identification of the cylinder detected in misfire
Shows the cylinders that will be detected in misfire

Carrier stucture valid for

MIS_DET_CYL_INH and **MIS_STATE_ER**



Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_USE_ER_STND_MIS	-	0...1H	0...1	1	[-]
Misfire detection integration mode (based on ER index = 0, based on ER_STND index = 1)					

B.2 Misfire detection - Application incidences

B.2.1 Misfire detection threshold correction gain specific to application

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_THD_APP_ER	O/V	0...1FFH	0...1.996093	0.003906	-
Misfire detection threshold correction gain specific to application					
FAC_THD_CH_ER	V	0...1FFH	0...1.996093	0.003906	-
Misfire detection threshold correction gain for catalyst heating					
FAC_THD_FQ_ER	V	0...1FFH	0...1.996093	0.003906	-
Misfire detection threshold correction gain for fuel quality adaptation					

Input data:

LV_ENA_SEG_T_MES	LC_MIS_INH	TQ_ADD_CH	TCO
LV_AT	TI_IS_MMV_MAX	FAC_FQ_ST_AD	

Description:

In case of heavy catalyst heating (large ignition retard) misfire detection threshold also has to be corrected due to delayed combustion: correction versus amount of catalyst heating torque reserve.

Application conditions:

Initialization: *at reset: FAC_THD_APP_ER = 1*
FAC_THD_CH_ER = 1
FAC_THD_FQ_ER = 1

Activation: *LV_ENA_SEG_T_MES = 1*
and *LC_MIS_INH = 0*

Deactivation: *LV_ENA_SEG_T_MES= 0*
or *LC_MIS_INH = 1*

Actions on Deactivation event: *FAC_THD_APP_ER = 1*
FAC_THD_CH_ER = 1
FAC_THD_FQ_ER = 1

Reccurrence : Update every segment before misfire detection function

Formula section:

If LV_AT = 1
then FAC_THD_CH_ER = IP_FAC_THD_APP_ER_CH_AT
else FAC_THD_CH_ER = IP_FAC_THD_APP_ER_CH_MT
endif

If TI_IS_MMV_MAX > C_TI_IS_MMV_THD_ER
or FAC_FQ_ST_AD > C_FAC_FQ_ST_AD_THD_ER
then FAC_THD_FQ_ER = IP_FAC_THD_FQ_ER
else FAC_THD_FQ_ER = 1

endif

FAC_THD_APP_ER = FAC_THD_CH_ER * FAC_THD_FQ_ER

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_THD_APP_ER_CH_MT	8 x 4	0...1FF H	0...1.996093	0.003906	-
LDPM_TCO__FAC_THD_APP_ER	8	0...FE H	-48...142.5	0.75	°C
LDPM_TQ_ADD_CH__FAC_THD_APP_ER	4	0...FFFFH	-1024...1023.97	0.03125	Nm
Weighting factor for determination of the engine roughness threshold for compensation TCO influence, catalyst heating torque reserve / MT; init = 1.0					
IP_FAC_THD_APP_ER_CH_AT	8 x 4	0...1FF H	0...1.996093	0.003906	-
LDPM_TCO__FAC_THD_APP_ER	8	0...FE H	-48...142.5	0.75	°C
LDPM_TQ_ADD_CH__FAC_THD_APP_ER	4	0...FFFFH	-1024...1023.97	0.03125	Nm
Weighting factor for determination of the engine roughness threshold for compensation TCO influence, catalyst heating torque reserve / AT; init = 1.0					
IP_FAC_THD_FQ_ER	8	0...1FF H	0...1.996093	0.003906	-
LDPM_TCO__FAC_THD_APP_ER	8	0...FE H	-48...142.5	0.75	°C
Weighting factor for desensibilisation of the engine roughness threshold due to low fuel quality					
C_TI_IS_MMV_THD_ER	1	80...7F H	-0.5...0.496	1/256	-
Threshold on TI_IS_MMV_MAX to activate desensibilisation of Misfire detection					
C_FAC_FQ_ST_AD_THD_ER	1	0...FF H	0...1.992	0.0078	-
Threshold on FAC_FQ_ST_AD to activate desensibilisation of Misfire detection					

B.2.2 Correction gain for engine load specific to application

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_GAIN_LOAD_MIS	V/O	0...1FFH	0...1.99609	3.91E-03	[-]
Correction gain of the normalised engine load for misfire, specific to application					

Input data:

LV_ENA_SEG_T_MES	LC_MIS_INH		

Description:

Here can be defined a gain who can change the normalised engine load dedicated to misfire monitoring according specific load corrections.

This correction can be applied during torque reduction sequences specific to project.

Application conditions:

Activation: *LV_ENA_SEG_T_MES = 1*

And *LC_MIS_INH = 0*

Deactivation: *LV_ENA_SEG_T_MES= 0*

Or *LC_MIS_INH = 1*

Actions on Deactivation event:

FAC_GAIN_LOAD_MIS = 1

Recurrency: updated every segment before misfire engine load determination

Formula section:

FAC_GAIN_LOAD_MIS = 1

B.2.3 Misfire detection inhibition related to OBDI diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_OBD_DET_MIS	V/O	0...1H	0...1	1	-
OBD I misfire detection process inhibition flag					

Input data:

LV_ERR_CRK	LV_ERR_CAM_IN [NC_NR_CAM_CBK]	LV_ERR_MAF	LV_ERR_TPS
LV_ENA_ER	LC_MIS_INH	LV_MTC_CUR_OFF	LV_ERR_LOAD_TPS_PLAUS
LV_ERR_CAM_EX [NC_NR_CAM_CBK]			

FUNCTION DESCRIPTION:

Misfire detection process is inhibited when one of the following OBD I errors occurs.

(see project system integration)

Application conditions:

Activation: LV_ENA_ER = 1
And LC_MIS_INH = 0

Deactivation: LV_ENA_ER = 0
Or LC_MIS_ INH = 1

Initialisation: at ECU reset, at Engine Stop **Or** at LC_MIS_INH 0 to 1 transition
LV_INH_OBD_DET_MIS = 1

Recurrence: updated every segment before misfire detection function

Formula section:

```
If      LV_ERR_CRK = 1                  // crankshaft signal error
   Or   LV_ERR_CAM_IN = 1               // camshaft signal error
   Or   LV_ERR_CAM_EX = 1               // camshaft signal error
   Or   LV_ERR_MAF = 1                  // mass air flow signal error detected
   Or   LV_ERR_MAP = 1                  // Manifold absolute pressure sensor error
   Or   LV_ERR_LOAD_TPS_PLAUS = 1       // MAF or TPS signal implausibility detected
   Or   LV_ERR_TPS = 1                  // throttle position sensor signal error
   Or   LV_MTC_CUR_OFF = 1              // ETC limp-home requested
Then    LV_INH_OBD_DET_MIS  = 1
Else    LV_INH_OBD_DET_MIS  = 0
EndIf
```

B.2.4 Misfire detection inhibition related to application

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_APP_DET_MIS	V/O	0...1H	0...1	1	-
Misfire detection process inhibition flag *(application specific)*.					
LV_FIRST_EOL_FDOUT	V/O/S	0H...1H	0...1	1	-
Boolean for the misfire monitoring inhibition during the first engine start at EOL					
LV_STATE_MIS_ON	V/O	0H...1H	0...1	1	-
Boolean for the misfire monitoring ON/OFF status, for KWP					
LV_IV_OFF_DET_MIS_APP	V	0H...1H	0...1	1	-
Misfire detection fade out flag due to cylinder shut-off, phased with misfire detection index (App.)					
LV_INH_IV_OFF_DET_MIS_APP	V	0...1H	0...1	1	-
Misfire detection fade out request flag due to cylinder shut-off condition (App.)					

Input data:

LV_SEG_AD_DIF_MAX_ER	LV_ENA_ER	LC_MIS_INH	C_SUM_INH_IV_MAX_MIS
LV_PUC_DET_MIS	SEG_NR	PREV_STATE_IV	SUM_INH_INJ
LV_INH_N_MAX_DET_MIS	INH_IV_MIS	INH_IV_MIS_GEN	LC_REQ_MIS_GEN

FUNCTION DESCRIPTION:

Misfire detection process can be inhibited when one of the following function requires an inhibition.

Application conditions:

Activation: LV_ENA_ER = 1

And LC_MIS_INH = 0

Deactivation: LV_ENA_ER = 0

Or LC_MIS_ INH = 1

Initialisation: at ECU reset, at Engine Stop ***Or*** *at LC_MIS_INH 0 to 1 transition LV_INH_APP_DET_MIS = 1*

Update rate: *updated every segment before misfire detection function*

Formula section:

If LV_SEG_AD_DIF_MAX_ER = 1

Or LV_FIRST_EOL_FDOUT = 1

Or LV_INH_IV_OFF_DET_MIS_APP =1

Then LV_INH_APP_DET_MIS = 1

Else LV_INH_APP_DET_MIS = 0

EndIf

B.2.4.1 Misfire detection inhibition at End of line test

B.2.5 FUNCTION DESCRIPTION:

When the engine start is occurred for the first time at the production line, the misfire error may be detected because the fuel is not fully delivered to fuel line from the fuel tajnk. To avoid the misfire detection at EOL, the misfire monitoring shall be inhibited(LV_FIRST_EOL_FDOUT = 1) during the first engine running time(NC_MAX_T_AST_FIRST_EOL_FDOUT) after the first ECU powered-up.

This misfire monitoring ON / OFF state information (LV_STATE_MIS_ON) is sent to a scan tool via KWP.

Application conditions:

Initialization : At ECU- first time powered-up

LV_FIRST_EOL_FDOUT = 1

LV_STATE_MIS_ON = 0

Activation : LV_IGK = 1

Recurrence : 1 sec

Formula section:

If LV_FIRST_EOL_FDOUT = 0 (EOL check already done)

Then LV_STATE_MIS_ON = 1 (Misfire mon. is ON, bit for KWP support)

Else

IF LV_ST_END = 1

And T_AST >= NC_MAX_T_AST_FIRST_EOL_FDOUT

THEN

LV_FIRST_EOL_FDOUT = 0 (Saved into ROM at the end of DC, not any more set to 1 during the whole the ECU life time)

ENDIF

EndIf

B.2.5.1 Inhibition of MISF diagnosis for specific cylinder shut-off.(App.)

Inhibition of MISF diagnosis for specific cylinder shut-off.(App.) can be activated by LC_INH_IV_OFF_DET_MIS_APP=0. When this module is activated, bit for generic inhibition for specific cylinder shut-off should be deactivated. This can be done by zero setting for generic inhibition bit of C_MASK_INH_DET_MIS which enables to inhibit the flag in the carrier MIS_DET_CDN_INH ("Legal misfire detection fade-out").

<u>Application conditions:</u>

Initialisation: *at Reset, Engine Stop,* ***Or*** *on Deactivation event*

LV_IV_OFF_DET_MIS_APP = 0

LV_INH_IV_OFF_DET_MIS_APP = 0

Recurrence: *every segment task*

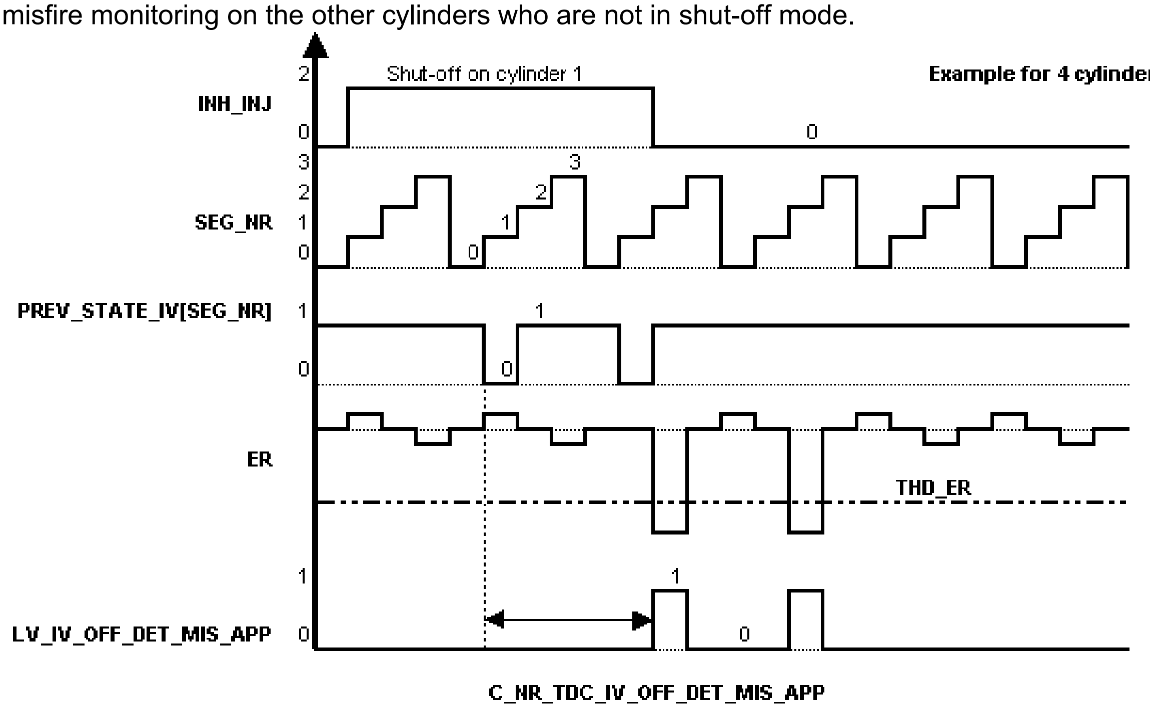
Activation: *LC_MIS_INH = 0* ***AND*** *LC_INH_IV_OFF_DET_MIS_APP=0*

Deactivation: *LC_MIS_INH = 1* ***OR*** *LC_INH_IV_OFF_DET_MIS_APP=1*

Misfire detection can be suppressed for the cylinder(s), for which injection has been shut-off.

<u>Cylinder specific shut-off:</u>

If the current cylinder checked by misfire detection is shut-off by any EMS function a fade-out can be applied for the misfire detection of this cylinder.
Due to the delay between the misfire index and the injection current informations, a calibration delay (C_NR_TDC_IV_OFF_DET_MIS_APP) has been introduced on LV_IV_OFF_DET_MIS_APP.
This delay synchronises MISF & INJR informations in a way to apply correctly the fade-out on the proper cylinder during individual cylinder shut-off operations, and to be able the perform the misfire monitoring on the other cylinders who are not in shut-off mode.



Formula section:

Cylinder fuel shut-off:

If PREV_STATE_IV[SEG_NR] = 0 (*indexed by cylinder bit position)*

And INH_IV_MIS = 0 *(no cylinder shut-off due to misfire detection)*

And LC_REQ_MIS_GEN = 0 *(no cylinder shut-off due to internal misfire generator)*

Then LV_IV_OFF_DET_MIS_APP = 1 after a C_NR_TDC_IV_OFF_DET_MIS_APP delay

Else LV_IV_OFF_DET_MIS_APP = 0 after a C_NR_TDC_IV_OFF_DET_MIS_APP delay

EndIf

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_IV_OFF_DET_MIS_APP = 1

Formula section:

Information for General Fuel Shut-Off and Limit number of Cylinder Shut-Off can be refer to Section " Legal Misfire Detection Fade-Out Conditions".

General definition:

If LV_PUC_DET_MIS = 1 (General Fuel Shut-Off)

Or LV_IV_OFF_DET_MIS_APP = 1 (Specific Cylinder Shut-Off)

Or **{**SUM_INH_INJ >= C_SUM_INH_IV_MAX_MIS (Limit Number of Cylinder Shut-Off)

And **[(**INH_IV_MIS = 0 *(no cylinder shut-off due to misfire detection)*

And INH_IV_MIS_GEN = 0 *(no cylinder shut-off due to internal misfire generator)*

And LC_DISA_INH_IV_OFF_DET_MIS_APP = 1**)**

Or LC_DISA_INH_IV_OFF_DET_MIS_APP = 0 **]}**

Then LV_INH_IV_OFF_DET_MIS_APP = 1

(and reset C_NR_TDC_IV_OFF_DET_MIS_DLY delay counter)

Else LV_INH_IV_OFF_DET_MIS_APP = 0 after C_NR_TDC_IV_OFF_DET_MIS_DLY delay

EndIf

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_NR_TDC_IV_OFF_DET_MIS_APP	1	0...7H	0...7	1	-
TDC delay to synchronise cylinder shut-off information with the misfire detection index (App.)					
C_NR_TDC_IV_OFF_DET_MIS_DLY	1	0...FFH	0...255	1	-
TDC delay to re-start misfire detection after cylinder shut-off (App.)					
LC_INH_IV_OFF_DET_MIS_APP	1	0...1H	0...1H	1	-
Switch to enable inhibition of MISF diagnosis after specific cylinder shut-off (App)					
LC_DISA_INH_IV_OFF_DET_MIS_APP	1	0...1H	0...1H	1	-
Switch to disable inhibition of MISF diagnosis after specific cylinder shut-off (App)					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_MAX_T_AST_FIRST_EOL_FDOUT	1	0H...FFH	0...255	1	Sec
Maximum misfire detection inhibition time after first engine run					

B.2.6 Rough road detection application incidences fade out

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_INH_APP_RR	V/O	0...01H	0...1	1	-
Rough road detection appl. Inc. fade out					

Input data:

LV_ERR_MWSS	LV_ERR_VS	LC_MIS_INH	

FUNCTION DESCRIPTION:

Rough road detection process can be inhibited when one of the following function and/or error requires an inhibition :

WSS solution : OBDI wheel speed sensor(s) error(s), Braking ...

CAN solution : General CAN time out, ABS module CAN time out, ABS in action, Rationality error, ...

(see project system integration)

Application conditions:

Activation: LC_MIS_INH = 0

Deactivation: LC_MIS_ INH = 1

Initialisation: ECU reset ***Or*** *on Deactivation event*

LV_INH_APP_RR = 0

Update rate: same as rough road detection update rate

Formula section:

(WSS solution example)

If LV_ERR_MWSS = 1

Or LV_ERR_VS = 1

Then LV_INH_APP_RR = 1

Else LV_INH_APP_RR = 0

B.2.7 Crankshaft oscillation detection fade-out

#IF NLC_USE_CRK_OSC_MIS = 1

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_INH_CRK_OSC_DET	V/O	0...01H	0...1	1	-
Crankshaft oscillation detection fade out					

Input data:

LC_MIS_INH	LV_AT		

FUNCTION DESCRIPTION:

Crankshaft oscillation detection process can be inhibited when one function and/or error requires an inhibition.

Note : This bit is necessary only if the crankshaft oscillation detection module is used in the project (NLC_USE_CRK_OSC_MIS = 1)

Application conditions:

Activation: LC_MIS_INH = 0

Deactivation: LC_MIS_ INH = 1

Initialisation: ECU reset ***Or*** *on Deactivation event*

LV_INH_CRK_OSC_DET = 1

Recurrence: updated every segment before misfire detection function

Formula section:

If (LV_AT = 1

And LC_ INH_CRK_OSC_DET_AT = 1)

or (LV_AT = 0

And LC_ INH_CRK_OSC_DET_MT = 1)

Then LV_INH_CRK_OSC_DET = 1

Else LV_INH_CRK_OSC_DET = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_INH_CRK_OSC_DET_MT	1	0...1 H	0...1	1	-
Inhibition of crankshaft oscillation detection for MT-vehicle					
LC_INH_CRK_OSC_DET_AT	1	0...1 H	0...1	1	-
Inhibition of crankshaft oscillation detection for AT-vehicle					

#ENDIF

B.2.8 Misfire low fuel tank level informations

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_FTL_L_DET_MIS	V/O	0...1	0...1H	1	-
Misfire detection inhibition flag in case of low fuel tank level					

LV_INH_FTL_L_DET_MIS = 0

B.2.9 Definition of dummy values for ER-calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ER_FIL	O/V	FF800000 ...7FFFFFH	-2000000... 1999999.76158	0.2384186	[µs]
Filtered Engine roughness					
ER_DELTA	O	FF800000 ...7FFFFFH	-2000000... 1999999.76158	0.2384186	[µs]
Delta Engine roughness on same physical segment					
LV_DET_THD_DELTA_FIL_MEM	O	0...1H	0...1	1	[-]
Single misfire detected based only on threshold criterion on delta mean values (continuous and stored)					
LV_DET_THD_DELTA	O	0...1H	0...1	1	[-]
Single misfire detected based only on ER_DELTA criterion (before fade out test)					

Input data:

ER	LV_ENA_ER		

FUNCTION DESCRIPTION:

This function calculates all the ER signals used for detection

Application conditions:

Initialisation: at ECU reset, at Engine Stop To Runing

ER_FIL = ER_DELTA = 0

Recurrence: updated every segment **before** misfire detection function

Activation: LV_ENA_ER = 1

Deactivation: LV_ENA_ER = 0

Formula section:

ER_FIL = ER

ER_DELTA = 0

LV_DET_THD_DELTA_FIL_MEM = 0

LV_DET_THD_DELTA = 0

Configuration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
NLC_CONF_ER_MV	1	0...1H	0...1	1	[-]
Mean ER correction enable					

B.2.10 Misfire detection calibration ease

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ER_THD_RATIO	V	0...FFFFH	0...399.99389	6.1035e-3	[%]
clipped ratio between engine roughness index and detection threshold					

Input data:

LV_ENA_ER	ER	THD_ER	LC_MIS_INH

General information:

For calibration ease, a ratio between ER value and the ER detection threshold can be integrated. ER values are proportionnal to engine speed, thus the detection and fade-out calibration in transients engine operating points are choosey. Data visualisation and scalling have always to be updated according current engine operating point.

After precalibration, with the introduction of such ratio, the misfire detection check and validation will be much more easier for tuning team.

Ratio close to zero : no risk of misfire detected.
Ratio close to 100% or above : misfire detected by threshold *(before fade-out check)*.

Description:

Application conditions:

Activation: *LV_ENA_ER = 1*
And *LC_MIS_INH = 0*

Deactivation: *LV_ENA_ER = 0*
Or *LC_MIS_INH = 1*

Initialisation: ECU reset ***Or*** *on Deactivation event*

ER_THD_RATIO = 0

Update rate: updated every segment after misfire detection function

Formula section:

If ER < 0

ER_THD_RATIO = **min** (ER / THD_ER , C_ER_THD_RATIO_MAX)

Else ER_THD_RATIO = 0

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ER_THD_RATIO_MAX	1	0...FFFFH	0...399.99389	6.1035e-3	[%]
clipping maximum value of ratio between engine roughness index and detection threshold					

B.2.11 Current Misfire Detection Threshold

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
THD_ER_CLC	V/O	8000...7FFFH	-32768...32767	1	[µs]
Value of the table used for Misfire detection threshold at current segment					

Input data:

N_32	LV_DRI	LV_AT	LC_MIS_INH
N	LV_IS	LOAD_MIS	LV_ENA_SEG_T_MES

General information:

THD_ER_CLC represents the current value used for determination of the misfire detection threshold. This data must be calculated before THD_ER calculation and after LOAD_MIS calculation.

Application conditions:

Initialisation: at ECU reset **Or** engine stop
THD_ER_CLC = 0x8000

Recurrence: every ENRD segment task

Activation: *LV_ENA_SEG_T_MES = 1*
And *LC_MIS_INH = 0*

Deactivation: *LV_ENA_SEG_T_MES = 0*
Or *LC_MIS_INH = 1*

Actions on Deactivation event : THD_ER_CLC = 0x8000

Formula section:

Convention for transmission type determination:

If LV_AT = 1
Then yy = "AT"
Else yy = "MT"
EndIf

If(1) LV_IS = 1
And N_32 < C_N_32_MAX_IS_MIS
Then(1)
  If(2) (LV_AT = 1 **And** LV_DRI = 1)
  Then(2) THD_ER_CLC = IP_THD_AFS_ER_IS_DRI (N,LOAD_MIS)
  Else(2) THD_ER_CLC = IP_THD_AFS_ER_IS_yy(N,LOAD_MIS)
  EndIf(2)
Else(1) THD_ER_CLC = IP_THD_AFS_ER_yy(N,LOAD_MIS)
EndIf(1)

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_N_32_MAX_IS_MIS	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed to apply specific ER detection threshold in idle speed					
IP_THD_AFS_ER_AT	12*8	0...FFFFH	-32768...32767	1	[µs]
LDP_N_IP_THD_AFS_ER_AT	12	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_MIS_IP_THD_AFS_ER_AT	8	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER threshold for misfire detection in AFS mode, AT vehicle transmission type					
IP_THD_AFS_ER_MT	12*8	0...FFFFH	-32768...32767	1	[µs]
LDP_N_IP_THD_AFS_ER_MT	12	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_MIS_IP_THD_AFS_ER_MT	8	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER threshold for misfire detection in AFS mode, MT vehicle transmission type					
IP_THD_AFS_ER_IS_MT	4*6	0...FFFFH	-32768...32767	1	[µs]
LDP_N_IP_THD_AFS_ER_IS_MT	4	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_IP_THD_AFS_ER_IS_MT	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER threshold for misfire detection in AFS combustion mode & in idle speed, MT vehicle					
IP_THD_AFS_ER_IS_AT	4*6	0...FFFFH	-32768...32767	1	[µs]
LDP_N_IP_THD_AFS_ER_IS_AT	4	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_IP_THD_AFS_ER_IS_AT	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER threshold for misfire detection in AFS combustion mode & in idle speed, AT vehicle					
IP_THD_AFS_ER_IS_DRI	4*6	0...FFFFH	-32768...32767	1	[µs]
LDP_N_IP_THD_AFS_ER_IS_DRI	4	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_IP_THD_AFS_ER_IS_DRI	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
ER threshold for misfire detection in AFS combustion mode & in idle speed, AT vehicle, gear engaged					

B.2.12 Zero Load Line for Misfire detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LOAD_MIN_MIS	V/O	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine "zero" load for misfire detection					
FAC_APP_LOAD_MIN_MIS	V	0...1FFH	0...1.99609	3.9063e-3	[-]
Correction of the normalised engine"zero" load for misfire, specific to application					

Input data:

N	TCO	LV_AT	AMP
LC_MIS_INH	T_AST	N_32	LV_ENA_SEG_T_MES
VB			

FUNCTION DESCRIPTION:

The zero load line is a function of the engine speed and depends on the coolant temperature, atmospheric pressure and generator load.

Application conditions:

Initialisation: LOAD_MIN_MIS is initialised to 0 at reset, at engine stop Or at LC_MIS_INH 0 to 1 transition

FAC_APP_LOAD_MIN_MIS is initialised to 1 at reset, at engine stop Or at LC_MIS_INH 0 to 1 transition

Recurrence: every segment task

Activation: LV_ENA_SEG_T_MES = 1

And LC_MIS_INH = 0

Deactivation: LV_ENA_SEG_T_MES = 0

Or LC_MIS_INH = 1

Formula section:

Minimum engine load and zero load lines

If T_AST < C_T_AST_MAX_LOAD_MIN_MIS_COR
And TCO > C_TCO_MAX_LOAD_MIN_MIS_COR
Then FAC_APP_LOAD_MIN_MIS = IP_FAC_LOAD_MIN_MIS_COR_VB
Else FAC_APP_LOAD_MIN_MIS = 1
Endif

If (LV_AT = 0)

Then LOAD_MIN_MIS = [IP_LOAD_MIN_MIS_AFS_MT(N, TCO)
+ IP_LOAD_MIN_OFS_AMP_MIS_AFS_MT(N, AMP)]
* FAC_APP_LOAD_MIN_MIS

Else LOAD_MIN_MIS = [IP_LOAD_MIN_MIS_AFS_AT(N, TCO)
+ IP_LOAD_MIN_OFS_AMP_MIS_AFS_AT(N, AMP)]
* FAC_APP_LOAD_MIN_MIS

Endif

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_T_AST_MAX_LOAD_MIN_MIS_COR	1	0...FFFFH	0...6553.5	0.1	[s]
Maximum time after start for correction of LOAD_MIS in case of low generator load					
C_TCO_MAX_LOAD_MIN_MIS_COR	1	0...FEH	-48...142.5	0.75	[°C]
Maximum coolant temperature for correction of LOAD_MIS in case of low generator load					
IP_FAC_LOAD_MIN_MIS_COR_VB	4*2	0...1FFH	0...1.99609	3.9063e-3	[-]
LDP_VB_FAC_LOAD_MIN_MIS_COR	4	0...FFH	0...26	0.102	[V]
LDP_N_32_FAC_LOAD_MIN_MIS_COR	2	0...FFH	0...8160	32	[rpm]
Weighting factor for correction of LOAD_MIS in case of low generator load					
IP_LOAD_MIN_MIS_AFS_MT	8*8	0...7FFFH	0...99.99694	3.0518e-3	[%]
LDPM_N_32_LOAD_MIN_MIS_AFS_MT	8	0...FFH	0...8160	32	[rpm]
LDPM_TCO_LOAD_MIN_MIS_AFS	8	0...FEH	-48...142.5	0.75	[°C]
Zero load curve vs. engine speed and coolant temperature in AFS combustion mode, with MT vehicle					
IP_LOAD_MIN_MIS_AFS_AT	8*8	0...7FFFH	0...99.99694	3.0518e-3	[%]
LDPM_N_32_LOAD_MIN_MIS_AFS_AT	8	0...FFH	0...8160	32	[rpm]
LDPM_TCO_ LOAD_MIN_MIS_AFS	8	0...FEH	-48...142.5	0.75	[°C]
Zero load curve vs. engine speed and coolant temperature in AFS combustion mode, with AT vehicle					
IP_LOAD_MIN_OFS_AMP_MIS_AFS_MT	8*8	0...7FFFH	-50...49.99389	3.0518e-3	[%]
LDPM_N_32_LOAD_MIN_MIS_AFS_MT	8	0...FFH	0...8160	32	[rpm]
LDPM_AMP_LOAD_AMP_MIS_AFS	8	0...FFFFH	0...5434	0.0829175	[hPa]
Extra load from zero load curve vs. engine speed and atmospheric pressure in AFS with MT vehicle					
IP_LOAD_MIN_OFS_AMP_MIS_AFS_AT	8*8	0...7FFFH	-50...49.99389	3.0518e-3	[%]
LDPM_N_32_LOAD_MIN_MIS_AFS_AT	8	0...FFH	0...8160	32	[rpm]
LDPM_AMP_LOAD_AMP_MIS_AFS	8	0...FFFFH	0...5434	0.0829175	[hPa]
Extra load from zero load curve vs. engine speed and atmospheric pressure in AFS with AT vehicle					

B.3 Generic misfire parameters and fade-out conditions

B.3.1 Misfire detection engine parameters

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LOAD_MIS	V/O	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine load for misfire detection - sampled at current misfire segment					
LOAD_MIS_1	-	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine load for misfire detection - 1 misfire segment before					
LOAD_MIS_2	-	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine load for misfire detection - 2 misfire segment before					
LOAD_MIS_3	V	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine load for misfire detection - 3 misfire segment before					
LOAD_GRD_MIS	V/O	0...7FFFH	0...99.99694	3.0518e-3	[%]
Normalised engine load gradient for misfire detection					

Input data:

AMP	TCO	MAF	LV_ENA_SEG_T_MES
FAC_GAIN_LOAD_MIS	TQI_AV	LC_MIS_INH	MAP

FUNCTION DESCRIPTION:

In order to cover all the combustion states of an engine (Hom. AFS, AFL & Strat.) in torque based engine control, a general variable LOAD_MIS that represents the ratio of engine load – whatever should be the load definition depending on engine combustion state (*eg. MAF in Hom. and TQI_AV in Strat.*) – is defined for the purpose of the Misfire function and the calculation of the ER-misfire detection thresholds.

Depending on the project choice configuration (*with C_CONF_LOAD_MIS*), in Homogenous combustion mode this Misfire load variable should be either MAF, MAP or TQI_AV. In the other hand this Misfire load variable is always TQI_AV in Stratified combustion mode.

Legally, the presence of engine misfire in the engine operating region is bounded by the positive torque line (i.e. engine load with the transmission in neutral) and an engine speed value.

The zero load line is a function of the engine speed and depends on the engine combustion mode (for GDI application), coolant temperature and atmospheric pressure.

Application conditions:

Initialisation: All variables are initialised to 0 at reset, at engine stop **Or** at LC_MIS_INH 0 to 1 transition

LOAD_GRD_MIS is first time calculated when LOAD_MIS stack have been fill in (4 segments).

Recurrence: every segment task

Activation: LV_ENA_SEG_T_MES = 1

And LC_MIS_INH = 0

Deactivation: LV_ENA_SEG_T_MES = 0

Or LC_MIS_INH = 1

Formula section:

Normalised engine load dedicated to misfire detection

For each misfire segment, the following stack is serviced:

LOAD_MIS_3 =LOAD_MIS_2

LOAD_MIS_2 =LOAD_MIS_1

LOAD_MIS_1 = LOAD_MIS

If C_CONF_LOAD_MIS = 0

Or LV_S_ACT = 1

Then *LOAD_MIS_Temp* = TQI_AV / IP_TQI_AV_MAX_MIS(N_32) *// TQI_AV based*

ElseIf C_CONF_LOAD_MIS = 1

Then *LOAD_MIS_Temp* = MAF / IP_MAF_MAX_MIS(N32) *// MAF based*

Else *LOAD_MIS_Temp* = MAP / C_MAP_MAX_MIS *// MAP based*

EndIf

LOAD_MIS = FAC_GAIN_LOAD_MIS * *LOAD_Temp* *// Application incidence correction*

LOAD_GRD_MIS = | LOAD_MIS - LOAD_MIS_3 |

Note : LOAD_MIS calculation steps are checked against division by zero
LOAD_MIS_Temp is a non visible, temporary value

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_MAP_MAX_MIS	1	0...FFFFH	0...5434	0.0829175	[hPa]
Maximum manifold air pressure value to determine engine load ratio scale for misfire detection					
C_CONF_LOAD_MIS	1	0...2H	0...2	1	[-]
LOAD_MIS switch configuration versus engine combustion mode: (=0) LOAD_MIS always relative to TQI_AV (=1) LOAD_MIS relative to MAF in homog. & rel. to TQI_AV in stratif. mode, (=2) LOAD_MIS relative to MAP in homog. & rel. to TQI_AV in stratif. mode					
IP_TQI_AV_MAX_MIS	8	0...7FFFH	0...1023.97	0.03125	[Nm]
LDP_N_32_IP_TQI_AV_MAX_MIS	8	0...FFH	0...8160	32	[rpm]
Maximum torque value to determine engine load ratio scale for misfire detection					
IP_MAF_MAX_MIS	6	0...FFFFH	0...1389	0.0211948	[mg/stk]
LDP_N_32_IP_MAF_MAX_MIS	6	0...FFH	0...8160	32	[rpm]
Maximum air mass value to determine engine load ratio scale for misfire detection					

B.3.2 Generic fade out conditions for misfire detection

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
MIS_DET_CDN_APP_INH	V	0...FFFFH	0...65535	1	[-]
Misfiring APP INC detection fade out requests carrier word (before masking)					
MIS_DET_CDN_APP_INH_NR	V	0...FFH	0...255	1	[-]
Misfiring APP INC detection fade out requests number (after masking)					
MAP_DIF_MIS	V	0...FFFFH	0...5434	8.29E-02	[hPa]
Manifold air pressure difference between 2 misfire segment tasks					
IGA_DIF_MIS	V	0...FFH	0...95.625	0.375	[°CRK]
Absolute Ignition angle difference between 2 misfire segment tasks					
IGA_BAS_DIF_MIS	V	0...FFH	-35.625...60	0.375	[°CRK]
Ignition angle difference between IGA_BAS_COR_MV and IGA_AV, segment sample					
IGA_MIS_2	-	0...FFH	-35.625...60	0.375	[°CRK]
Ignition angle 2 misfire segment before					
IGA_MIS_1	-	0...FFH	-35.625...60	0.375	[°CRK]
Ignition angle 1 misfire segment before					
IGA_MIS	V	0...FFH	-35.625...60	0.375	[°CRK]
Ignition angle at current misfire segment					
LV_REQ_APP_INH_MIS	V/O	0...1H	0...1	1	[-]
Misfire detection APP INC fade out request flag					
LV_INH_MAP_DIF_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to MAP_DIF condition					
LV_INH_LOAD_GRD_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to LOAD_GRD_MIS condition					
LV_INH_TPS_GRD_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to TPS gradient condition					
LV_INH_IGA_DIF_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to IGA_DIF condition					
LV_INH_ACC_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to Air Conditionner transient condition					
LV_INH_CMB_TRA_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to combustion mode transients					

Input data:

MAP	MAF	TPS_AV	LV_INH_APP_DET_MIS
LOAD_MIS	N_32	LV_S_ACT	LV_ACCOUT_RLY
LOAD_GRD_MIS	LV_STATE_RR	LV_INH_FTL_L_DET_MIS	IGA_AV[NC_CYL_NR]
CTR_T_ZDLY_MIS	TPS_GRD	LV_HOM_AFL_ACT	NC_CMB_CONF
LV_STATE_CRK_OSC	LV_INH_OBD_DET_MIS	LV_ENA_ER	SEG_NR
LC_MIS_INH	LV_IS	OPM_AV	TCO
LV_IGA_GRD_ACT	IGA_BAS_COR_MV		

General information:

Generic fade out (LV_REQ_APP_INH_MIS) is based on all conditions who could create unreliable conditions for misfire detection :

- Engine gradients (engine load, throttle, combustion mode & ignition angle transients).
- Crankshaft jolt effects who can distrub detection process (rough road, drivetrain oscillations, air conditionner & accessoiries activation/deactivation).
- OBDI errors on one of the sensors used by the misfire monitoring process.

All these different conditions are merge into a carrier (MIS_DET_CDN_APP_INH).

Application conditions:

Initialisation: at ECU reset, at Engine Stop **Or** at LC_MIS_INH 0 to 1 transition

LV_REQ_APP_INH_MIS = 1

MIS_DET_CDN_APP_INH = 2047

MIS_DET_CDN_APP_INH_NR = 11

MAP_DIF_MIS = 0

IGA_DIF_MIS = 0

IGA_BAS_DIF_MIS = 0

IGA_MIS_2 = 0

IGA_MIS_1 = 0

IGA_MIS = 0

Recurrence: segment task for data process

10 ms for free running timers

Activation: LV_ENA_ER = 1

And LC_MIS_INH = 0

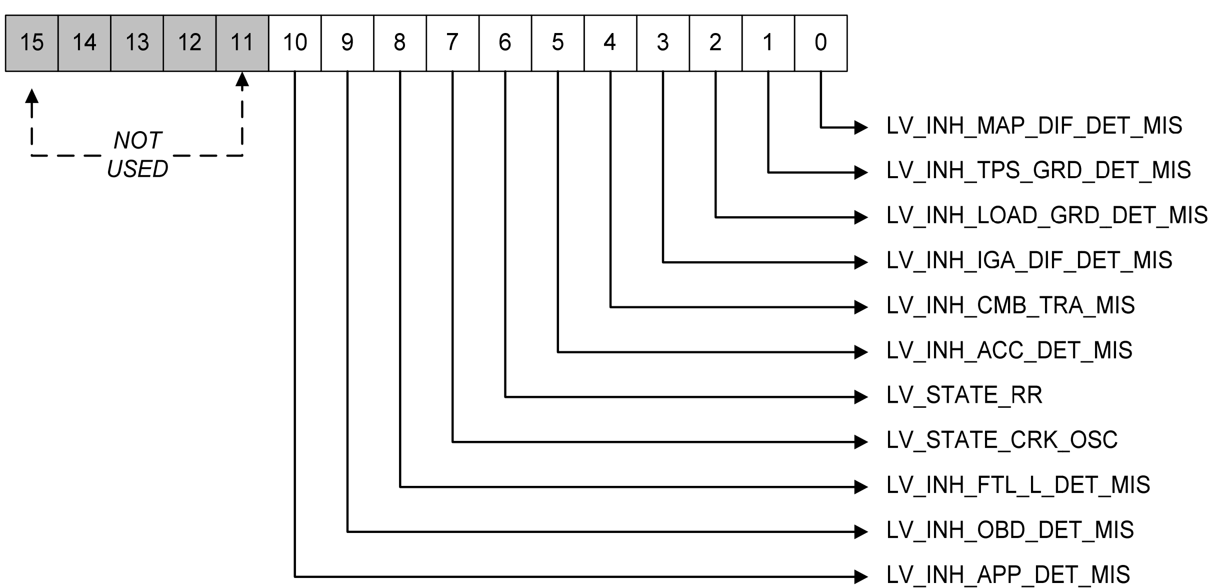
Deactivation: LV_ENA_ER = 0

Or LC_MIS_INH = 1

Remark : Timers used for delay are 10 ms free running decounters, they are not linked to activation/deactivation conditions described above.

B.3.2.1 Generic fade out condition carrier definition

MIS_DET_CDN_APP_INH



remark : MIS_DET_CDN_APP_INH is updated at current misfire task

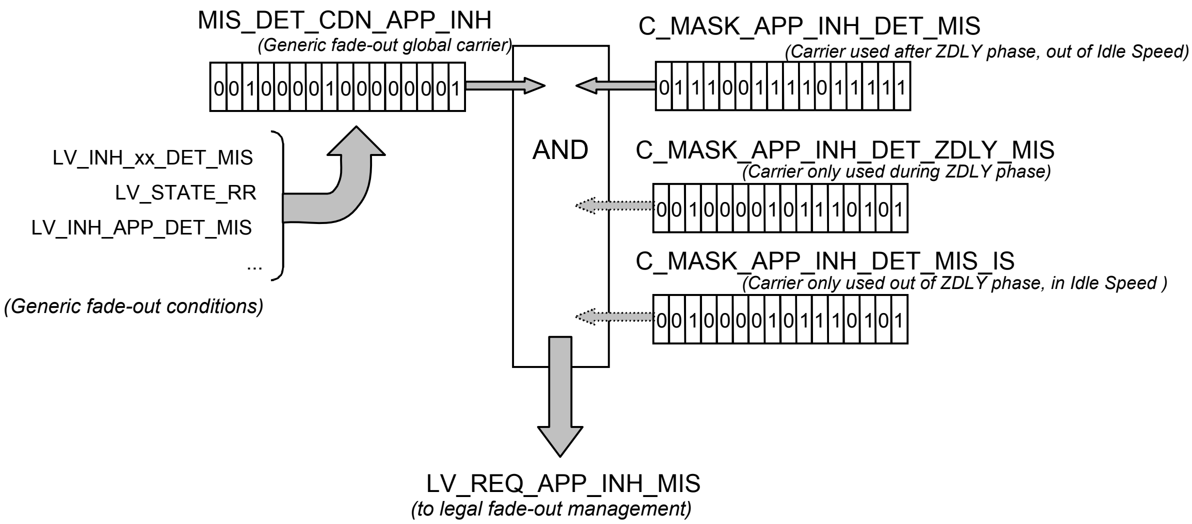
B.3.2.2 Configurable generic fade-out management

Generic fade out management during zero delay phase, nominal phase & idle speed engine state can be configurated via 3 bitfield masks that allow to take in account or not some conditions in a phase and not in the other.

If the corresponding bit in the MIS_DET_CDN_APP_INH carrier structure is set to 0 in C_MASK_APP_INH_DET_ZDLY_MIS, C_MASK_APP_INH_DET_MIS or C_MASK_APP_INH_DET_MIS_IS calibration, then the corresponding condition will not fade-out misfire detection.

For the definition of C_MASK_APP_INH_DET_ZDLY_MIS, C_MASK_APP_INH_DET_MIS & C_MASK_APP_INH_DET_MIS_IS masks please refer to the legal requirements & customer recommendations.

Overview:



Formula section:

If(1) CTR_T_ZDLY_MIS != 0

Then(1) *// ZDLY phase*

MIS_DET_CDN_APP_INH_NR =

sum(MIS_DET_CDN_APP_INH & C_MASK_APP_INH_DET_ZDLY_MIS)

(bitfield operations)

Else(1)

If(2) LV_IS = 1

Then(2) *// Idle Speed engine state*

MIS_DET_CDN_APP_INH_NR =

sum(MIS_DET_CDN_APP_INH & C_MASK_APP_INH_DET_MIS_IS)

(bitfield operations)

Else(2) *// Out of idle speed engine state*

MIS_DET_CDN_APP_INH_NR =

sum(MIS_DET_CDN_APP_INH & C_MASK_APP_INH_DET_MIS)

(bitfield operations)

EndIf(2)

EndIf(1)

If MIS_DET_CDN_APP_INH_NR != 0

Then LV_REQ_APP_INH_MIS = 1

Else LV_REQ_APP_INH_MIS = 0

EndIf

B.3.2.3 Maximum manifold air-pressure gradient

Due to trailing throttle / acceleration transition problems, it could be necessary to disable misfire detection for a short period when the manifold air-pressure gradient exceeds an applicable value.

Application conditions:

In all homogeneous combustion mode, Misfire detection is disabled when the amount of the MAP gradient exceeds IP_MAP_DIF_MAX_MIS(MAP)

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_MAP_DIF_DET_MIS = 1

Formula section:

MAP_DIF_MIS = | $MAP_{(n)}$ - $MAP_{(n-2)}$ |

If LV_S_ACT = 0

And MAP_DIF_MIS > IP_MAP_DIF_MAX_MIS(MAP)

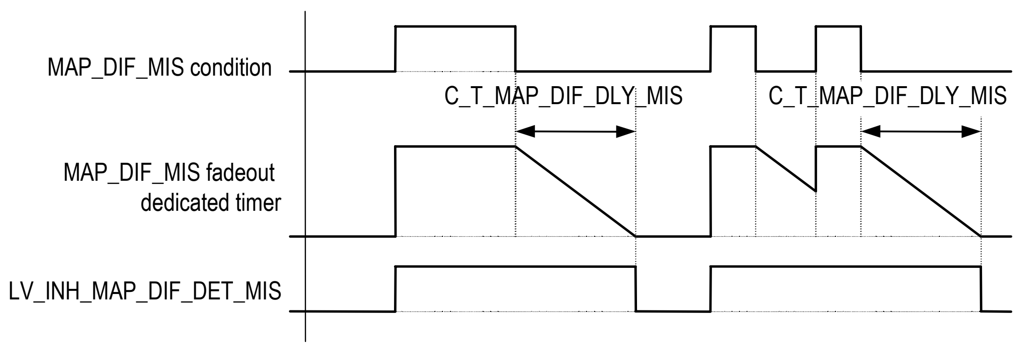
Then

After condition rising edge triggering, LV_INH_MAP_DIF_DET_MIS flag is set to 1 as long as condition is true.

After condition falling edge triggering, LV_INH_MAP_DIF_DET_MIS flag is hold to 1 for a period of C_T_MAP_DIF_DLY_MIS.

EndIf

Fade out behaviour summary:



B.3.2.4 Maximum throttle gradient

Due to trailing throttle / acceleration transient problems at low load, it is necessary to disable misfire detection for a short period when the throttle gradient exceeds an applicable value.

Application conditions:

Misfire detection can be disabled when the amount of the throttle gradient exceeds the applicable value hereunder mentioned. But during the 0-delay activation of the misfire monitoring, driver-induced fade-out is allowed:

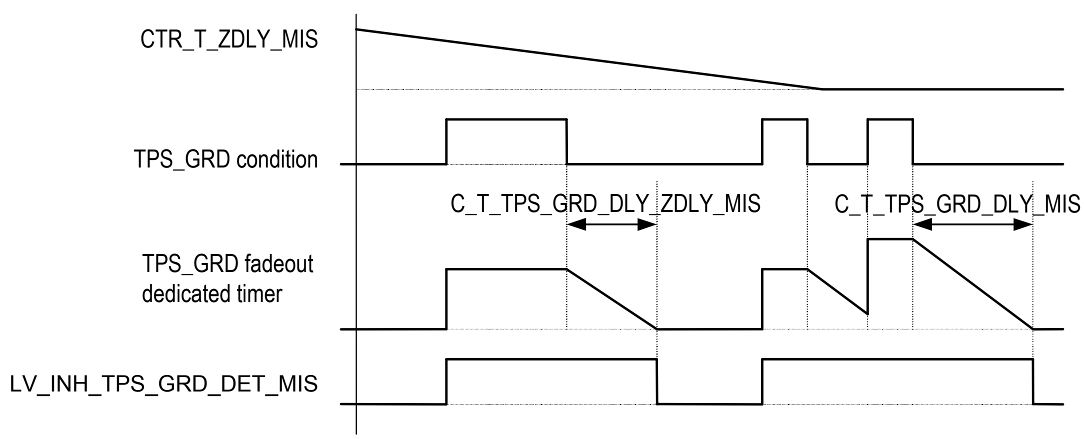
Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_TPS_GRD_DET_MIS = 1

Formula section:

If(1) CTR_T_ZDLY_MIS = 0
Then(1)
  If(2) TPS_GRD > IP_TPS_GRD_MAX_MIS(TPS_AV)
  Then(2)
    After condition rising edge triggering, LV_INH_TPS_GRD_DET_MIS flag is set to 1 as long as condition is true.
    After condition falling edge triggering, LV_INH_TPS_GRD_DET_MIS flag is hold to 1 for a period of C_T_TPS_GRD_DLY_MIS.
  EndIf(2)
Else(1)
  If(2) TPS_GRD > IP_TPS_GRD_ZDLY_MIS(TPS_AV)
  Then(2)
    After condition rising edge triggering, LV_INH_TPS_GRD_DET_MIS flag is set to 1 as long as condition is true.
    After condition falling edge triggering, LV_INH_TPS_GRD_DET_MIS flag is hold to 1 for a period of C_T_TPS_GRD_DLY_ZDLY_MIS.
  EndIf(2)
EndIf(1)

Fade out behaviour summary:



B.3.2.5 Maximum engine load gradient

Due to trailing throttle / acceleration transient problems at low load, it could be necessary to disable misfire detection for a short period when the engine load exceeds an applicable value.

Application conditions:

Misfire detection can be disabled when the amount of engine load gradient exceeds a calibration value, this value is switched according combustion mode. Fade out delay is also specific to combustion mode.

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_LOAD_GRD_DET_MIS = 1

Formula section:

If(1) LV_S_ACT = 0

Then(1)

> **If**(2) LOAD_GRD_MIS > IP_LOAD_GRD_MIS(LOAD_MIS)
>
> **Then**(2)
>
> > After condition rising edge triggering, LV_INH_LOAD_GRD_DET_MIS flag is set to 1 as long as condition is true.
> >
> > After condition falling edge triggering, LV_INH_LOAD_GRD_DET_MIS flag is hold to 1 for a period of C_T_LOAD_GRD_HOM_DLY_MIS.
>
> **EndIf**(2)

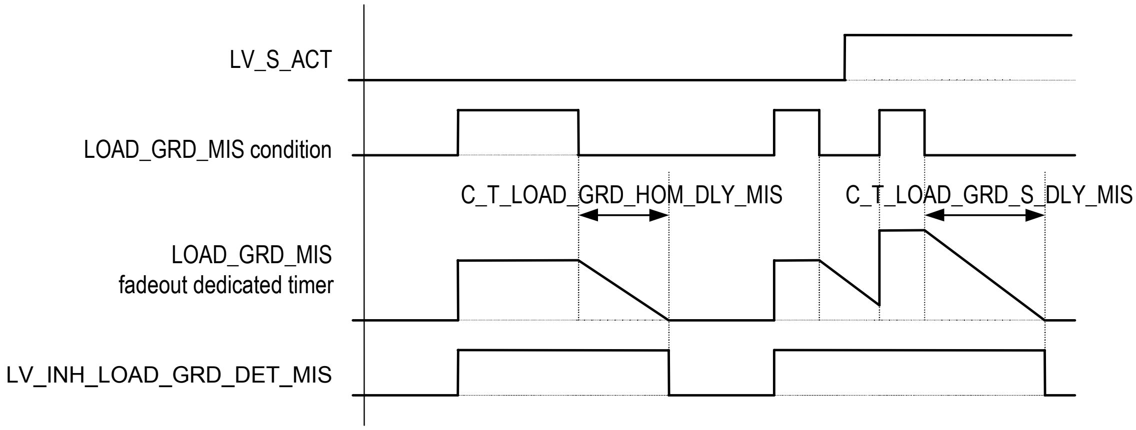
EndIf(1)

#If NC_CMB_CONF = AFS_S Or AFS_AFL_S

> **If**(1) LV_S_ACT = 1
>
> **Then**(1)
>
> > **If**(2) LOAD_GRD_MIS > IP_LOAD_GRD_S_MIS(LOAD_MIS)
> >
> > **Then**(2)
> >
> > > After condition rising edge triggering, LV_INH_LOAD_GRD_DET_MIS flag is set to 1 as long as condition is true.
> > >
> > > After condition falling edge triggering, LV_INH_LOAD_GRD_DET_MIS flag is hold to 1 for a period of C_T_LOAD_GRD_S_DLY_MIS.
> >
> > **EndIf**(2)
>
> **EndIf**(1)

#EndIf

Fade out behaviour summary:



B.3.2.6 Ignition retardation without change limitation

Various functions or CAN requests (e.g.. gear-shift signal. torque reduction. etc...) cause an ignition - timing retardation, without change limitation. Misfire detection can be disabled when the amount of the ignition timing retardation is too important.

Several criterion are available to disable the misfire monitoring according the ignition angle behaviour. Two relative criterions dedicated to focus on IGA transients, one in idle speed and one in part load, plus one absolute criterion dedicated to focus on too important IGA values compared to the

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_IGA_DIF_DET_MIS = 1

Formula section:

For each TDC, the following stacks are serviced after having calculated IGA_AV_x.

IGA_DIF_MIS = | IGA_MIS_2 - IGA_MIS |

IGA_BAS_DIF_MIS = IGA_BAS_COR_MV - IGA_MIS

IGA_MIS_2 = IGA_MIS_1

IGA_MIS_1 = IGA_MIS

IGA_MIS = IGA_AV[SEG_NR]

If(1) LV_S_ACT = 0

Then(1)

If(2) LV_IS = 1

And N_32 < C_N_32_MAX_IGA_IS_MIS

Then(2)

If(3) IGA_DIF_MIS > IP_DELTA_IGA_IS_MIS(TCO, LOAD_MIS_2)

Then(3) After condition rising edge triggering, LV_INH_IGA_DIF_DET_MIS flag is set to 1 as long as condition is true.

Else(3) After condition falling edge triggering, LV_INH_IGA_DIF_DET_MIS flag is hold to 1 for a period of C_NR_TDC_IGA_DIF_IS_MIS.

When this period is exceeded without retriggering, LV_INH_IGA_DIF_DET_MIS releases to 0.

EndIf(3)

Else(2) *// out of idle speed conditions*

If(3) IGA_DIF_MIS > IP_DELTA_IGA_MIS(N_32, LOAD_MIS)

Then(3) After condition rising edge triggering, LV_INH_IGA_DIF_DET_MIS flag is set to 1 as long as condition is true.

Else(3) After condition falling edge triggering, LV_INH_IGA_DIF_DET_MIS flag is hold to 1 for a period of C_NR_TDC_IGA_DIF_MIS.

When this period is exceeded without retriggering, LV_INH_IGA_DIF_DET_MIS releases to 0.

EndIf(3)

EndIf(2)

If(2) LV_IGA_GRD_ACT = 0 *// out of IGA engine start calculation phase*

And IGA_BAS_DIF_MIS > IP_DELTA_IGA_BAS_MIS(N_MIS,LOAD_MIS)

Then(2) After condition rising edge triggering, LV_INH_IGA_DIF_DET_MIS flag is set to 1 as long as condition is true.

Else(2) After condition falling edge triggering, LV_INH_IGA_DIF_DET_MIS flag is hold to 1 for a period of C_NR_TDC_IGA_BAS_DIF_MIS.

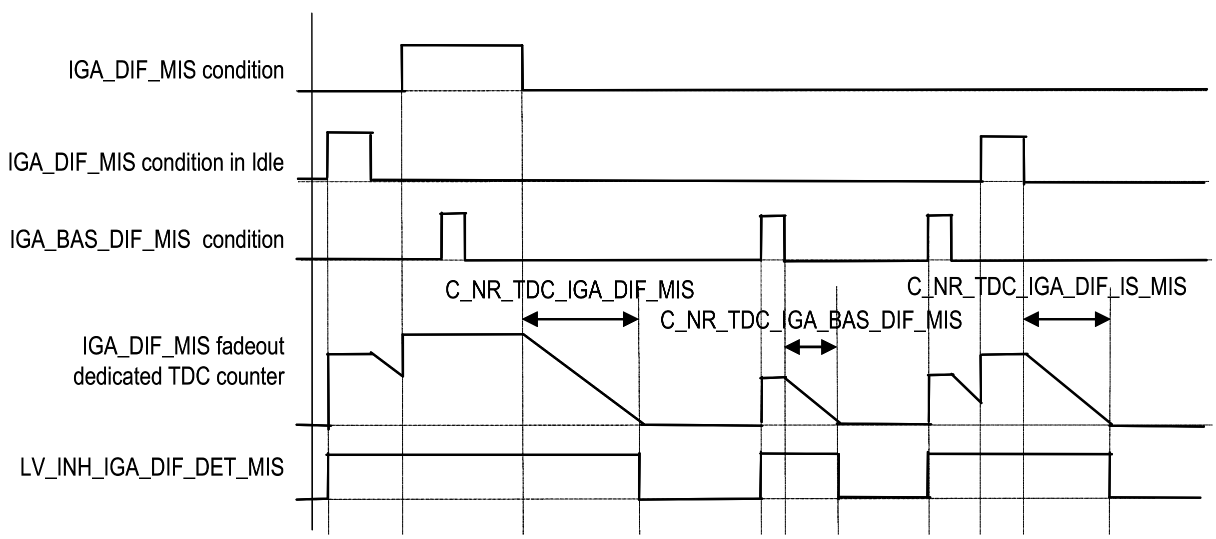
When this period is exceeded without retriggering, LV_INH_IGA_DIF_DET_MIS releases to 0.

EndIf (2)

EndIf(1)

// note: the TDC decounter is retriggered only if the considered preload calibration is greater than the decounter actual value.

Fade out behaviour summary:



B.3.2.7 Combustion mode transients

Application conditions:

While the combustion manager is in transient state, the misfire detection can be disabled to have a proper base for evaluation of the ER-misfire detection thresholds, which are defined for the stationary modes (AFS, AFL & S).

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_CMB_TRA_MIS = 1

Formula section:

#If NC_CMB_CONF = AFS_S Or AFS_AFL_S

If OPM_AV = x -> 1 (x is different of 1)

Then LV_INH_CMB_TRA_MIS is set to 1 during C_NR_TDC_CMB_TRA_S_MIS tdc's

EndIf

#EndIf

#If NC_CMB_CONF = AFS_AFL Or AFS_AFL_S

If OPM_AV = x -> 3 (x is different of 3)

Then LV_INH_CMB_TRA_MIS is set to 1 during C_NR_TDC_CMB_TRA_AFL_MIS tdc's

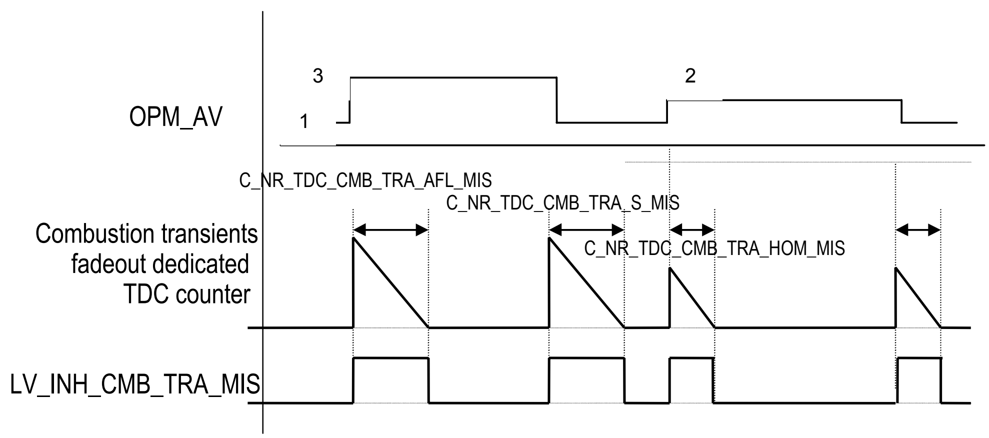
EndIf

#EndIf

If OPM_AV = x -> 2 (x is different of 2)

Then LV_INH_CMB_TRA_MIS is set to 1 during C_NR_TDC_CMB_TRA_HOM_MIS tdc's

EndIf



B.3.2.8 Air - conditioning compressor activation

When the air - conditioning compressor is switched on, an additional load is briefly applied to the engine.
This load jump can cause a segment period jump and crankshaft vibration, depending on the engine operating state and load request.

Application conditions:

Misfire detection can be suppressed for the applicable constant period C_T_ACCOUT_DLY_MIS, starting if the air - conditioning compressor is switched on or off (LV_ACCOUT_RLY changes from 0 to 1 or from 1 to 0), during this period LV_INH_ACC_DET_MIS is set to 1.

Initialisation: *at ECU reset, at Engine Stop* ***Or*** *at LC_MIS_INH 0 to 1 transition*

LV_INH_ACC_DET_MIS = 1

Formula section:

If LV_ACCOUT_RLY = 0 –> 1
Or LV_ACCOUT_RLY = 1 –> 0
Then LV_INH_ACC_DET_MIS is set to 1 during C_T_ACCOUT_DLY_MIS
EndIf

B.3.2.9 Rough road detection

Application conditions:

Misfire detection can be suppressed when the status of the *rough road detection* is active (LV_STATE_RR = 1).

B.3.2.10 Crankshaft oscillation detection

Application conditions:

Misfire detection can be suppressed when the status of the *crankshaft oscillation detection* is active (LV_STATE_CRK_OSC = 1).

B.3.2.11 Low fuel level

Application conditions:

Misfire detection can be suppressed when the low fuel level is detected (LV_INH_FTL_L_DET_MIS=1).

B.3.2.12 OBDI diagnosis fade-out

Application conditions:

LV_INH_OBD_DET_MIS combines all OBD project specific error bits to generate an input for the generic misfire detection fade-out conditions. *See definition in Misfire detection - Application Incidences file.* Misfiring detection can be suppressed if LV_INH_OBD_DET_MIS = 1

B.3.2.13 Application incidences fade-out

Application conditions:

LV_INH_APP_DET_MIS combines all project specific special conditions to generate an input for the generic misfire detection fade-out conditions. *See definition in Misfire detection - Application Incidences file*. Misfiring detection can be suppressed if LV_INH_APP_DET_MIS = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MASK_APP_INH_DET_ZDLY_MIS	1	0...FFFFH	0...65535	1	[-]
MIS_DET_CDN_APP_INH carrier structure fade out configuration mask (misfire detection in zero delay phase)					
C_MASK_APP_INH_DET_MIS	1	0...FFFFH	0...65535	1	[-]
MIS_DET_CDN_APP_INH carrier structure fade out configuration mask (detection in nominal phase out of IS)					
C_MASK_APP_INH_DET_MIS_IS	1	0...FFFFH	0...65535	1	[-]
MIS_DET_CDN_APP_INH carrier structure fade out configuration mask (detection in nominal phase & in IS)					
C_T_MAP_DIF_DLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when manifold air pressure gradient has been detected.					
C_T_TPS_GRD_DLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum throttle gradient has been detected.					
C_T_TPS_GRD_DLY_ZDLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum throttle gradient has been detected during zero delay starting phase					
C_NR_TDC_IGA_DIF_MIS	1	0...FFH	0...255	1	[-]
Fade out duration when IGA difference per TDC has been detected.					
C_NR_TDC_IGA_DIF_IS_MIS	1	0...FFH	0...255	1	[-]
Fade out duration when relative IGA criterion in idle speed has been detected					
C_NR_TDC_IGA_BAS_DIF_MIS	1	0...FFH	0...255	1	[-]
Fade out duration when absolute IGA criterion has been detected					
C_T_LOAD_GRD_HOM_DLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum LOAD_GRD_MIS has been detected in homogeneous combustion modes					
C_T_ACCOUT_DLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when air - conditioning compressor has been switched on.					
C_N_32_MAX_IGA_IS_MIS	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed to apply iga_dif_mis in idle speed					
IP_TPS_GRD_MAX_MIS	10	0...FFH	0...2987.5	11.7156863	[°TPS/s]
LDP_TPS_AV_IP_TPS_GRD_MAX_MIS	10	0...3FFFH	0...119.5	7.29E-03	[°TPS]
Maximum throttle gradient threshold for fade out condition.					
IP_TPS_GRD_ZDLY_MIS	6	0...FFH	0...2987.5	11.7156863	[°TPS/s]
LDP_TPS_AV_IP_TPS_GRD_ZDLY_MIS	6	0...3FFFH	0...119.5	7.29E-03	[°TPS]
Maximum throttle gradient threshold for fade out condition during zero delay misfire monitoring activation					
IP_LOAD_GRD_MIS	12	0...7FFFH	0...99.99694	3.05E-03	[%]
LDP_LOAD_MIS_IP_LOAD_GRD_MIS	12	0...7FFFH	0...99.99694	3.05E-03	[%]
Maximum LOAD_MIS gradient for fade out condition in homogeneous combustion modes					
IP_MAP_DIF_MAX_MIS	12	0...FFFFH	0...5434	8.29E-02	[hPa]
LDP_MAP_IP_MAP_DIF_MAX_MIS	12	0...FFFFH	0...5434	8.29E-02	[hPa]
Maximum MAP gradient for fade out condition.					
IP_DELTA_IGA_MIS	8*8	0...FFH	0...95.625	0.375	[°CRK]
LDP_N_32_IP_DELTA_IGA_MIS	8	0...FFH	0...8160	32	[rpm]
LDP_LOAD_MIS_IP_DELTA_IGA_MIS	8	0...7FFFH	0...99.99694	3.0518e-3	[%]
Spark advance angle difference at part load for misfire inhibition					
IP_DELTA_IGA_IS_MIS	4*4	0...FFH	0...95.625	0.375	[°CRK]
LDP_TCO_IP_DELTA_IGA_IS_MIS	4	0...FEH	-48...142.5	0.75	[°C]
LDP_LOAD_IP_DELTA_IGA_IS_MIS	4	0...7FFFH	0...99.99694	3.0518e-3	[%]
Spark advance angle difference in idle speed for misfire inhibition					
IP_DELTA_IGA_BAS_MIS	8*8	0...FFH	-35.625...60	0.375	[°CRK]
LDP_N_32_IP_DELTA_IGA_BAS_MIS	8	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_DELTA_IGA_BAS_MIS	8	0...7FFFH	0...99.99694	3.0518e-3	[%]
Absolute spark advance angle difference for misfire inhibition					
#IF NC_CMB_CONF = AFS_S Or AFS_AFL_S					

IP_LOAD_GRD_S_MIS	12	0...7FFFH	0...99.99694	3.05E-03	[%]
LDP_LOAD_MIS_IP_LOAD_GRD_S_MIS	12	0...7FFFH	0...99.99694	3.05E-03	[%]
Maximum LOAD_MIS gradient for fade out condition in stratified combustion mode					
C_T_LOAD_GRD_S_DLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when maximum LOAD_GRD_MIS has been detected in stratified combustion mode					
C_NR_TDC_CMB_TRA_S_MIS	1	0...FFH	0...255	1	[-]
Fade out TDC's duration when engine combustion manager enter in stratified combustion mode					
#ENDIF					
#IF NC_CMB_CONF = AFS_AFL Or AFS_AFL_S					
C_NR_TDC_CMB_TRA_AFL_MIS	1	0...FFH	0...255	1	[-]
Fade out TDC's duration when engine combustion manager enter in air/fuel lean combustion mode					
#ENDIF					
C_NR_TDC_CMB_TRA_HOM_MIS	1	0...FFH	0...255	1	[-]
Fade out TDC's duration when engine combustion manager enter in homogen combustion mode					

B.4 Legal Misfire Detection Fade-Out Conditions

B.4.1 General information

This chapter is relative to misfire detection fade out management as defined by legal texts linked to functionnal OBD diagnosis (OBDII, EOBD...).

Continuous misfire monitoring from engine starting instant:

- ***Engine start instant*** *: the point when the engine reaches a speed 150 rpm below the normal, warmed-up idle speed (as determined in the drive position for vehicles equipped with an automatic transmission) (CARB definition).*
  - *One engine cycle (2 crankshaft revolutions) is allowed to initialise misfire detection process (see CTR_TDC_ZDLY_MIS using).*

Fade out conditions may disable misfire monitoring with the following conditions:

- **Minimum engine speed** : N_IS_SP - 150 rpm *(US)*
- **Maximum engine speed** : Redline engine speed *(US)* or 4500 rpm *(EC)*
- **Minimum engine load** :

  Zero load line (engine positive torque line)

  A line joining the point (3000 rpm, zero load) and a point (6000 rpm, 135 hPa *(or 4 inches Hg)* above the zero load line)
- **Minimum coolant or ambiant temperature**

  Below 20°F (US) or -7°C (EC) *(equivalent),*

  Or If Engine start coolant temperature is below 20°F, until current temperature reaches warm up engine temperature 70°F.
- **Minimum atmospheric pressure**

  Elevations above 8000 feet above sea level (US) or 2 500 meters (EC)
- **Time after start**

  A 5s time after start fade-out can be allowed for EC market

  Such kind of fade out is prohibited for US market (see CARB requirement for zero delay misfire monitoring activation)
- **Cylinder shut-off**

  During fuel cut-off phase, on a specific cylinder shut-off or the number of cylinder in shut-off is too important, misfire diagnosis may be fade out *(no injection/combustion occurs)*

B.4.2 Misfire detection fade out management

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
CTR_TDC_ZDLY_MIS	V	0...FFFFH	0...65535	1	[-]
TDC decounter for disabled diagnostic					
CTR_T_ZDLY_MIS	V/O	0...FFFFH	0...655.35	0.01	[s]
10 ms decounter for diagnostic fade out (general purpose)					
LV_DIAG_MIS	V/O	0...1H	0...1	1	[-]
Nominal Misfire Monitoring Phase					
LV_ZDLY_DIAG_MIS	V/O	0...1H	0...1	1	[-]
Zero Delay Misfire Monitoring Phase					
STATE_DIAG_MIS	V	0H 1H 2H 3H 4H	INI_PHA PREP_0_DLY 0_DLY_MON PREP_MON NOM_MON	1	[-]
Misfire Detection Phase State Machine					

Input data:

N	LV_RUN_ENG	LV_ES	

FUNCTION DESCRIPTION:

CARB requests an activation of the misfire monitoring and diagnosis as off engine start. Particular fade-out conditions are required during the first 5 seconds since engine running. In this section the flag representing the request is defined, and is used in several other modules that are concerned with this requirement *(LC_REQ_ZDLY_MIS = 1).*

For non-CARB applications, this special monitoring is not required and has to be de-activated *(LC_REQ_ZDLY_MIS = 0).*

A state-based approach is taken for the activation of the misfire detection algorithms, distinguishing the separate monitoring during the first 5 seconds and the 'nominal' monitoring afterwards, and preparatory phases. During the monitoring phases, the normal fade-out mechanisms are used, corrected for some fade-outs that are not allowed during the first 5 seconds.

Application conditions:

Activation:

Engine Start LV_RUN_ENG = 1

Deactivation:

Engine Stop LV_RUN_ENG = 0 **Or** ECU reset

Remark: At deactivation, ALL internal and external variables related to the legal fade out misfire detection conditions must be reset to 0.
In nominal mode (after engine start and zero delay period), the state machine executes no operations.

Formula section:

At engine start-up (LV_RUN_ENG switches from 0 to 1) the system starts in STATE_DIAG_MIS = 0 (initialisation phase).

In the case of a CARB-application, special monitoring takes placed during C_T_ZDLY_MIS seconds. The timer is started at detection of engine running:

If LC_REQ_ZDLY_MIS = 1
And STATE_DIAG_MIS = 0
Then CTR_T_ZDLY_MIS = C_T_ZDLY_MIS
EndIf

Transition initialisation state to preparation states for 0-delay OR to nominal monitoring

If LC_REQ_ZDLY_MIS = 1
And STATE_DIAG_MIS = 0
And N > C_N_MIN_ZDLY_MIS
Then ***Zero-Delay Activation of Misfire Detection***
STATE_DIAG_MIS = 1
CTR_TDC_ZDLY_MIS = C_NR_TDC_ZDLY_MIS
ElseIf LC_REQ_ZDLY_MIS = 0
And STATE_DIAG_MIS = 0
And N > C_N_MIN_ZDLY_MIS
Then ***Standard Activation of Misfire Detection***
STATE_DIAG_MIS = 3
CTR_TDC_ZDLY_MIS = C_NR_TDC_ZDLY_MIS
Else ***Initialisation Phase***
LV_DIAG_MIS = 0
LV_ZDLY_DIAG_MIS = 0
EndIf

Transition back to the initialisation phase

If LV_ES = 1
And STATE_DIAG_MIS <> 0
Then ***In case of engine stop, return to initialisation phase***
STATE_DIAG_MIS = 0
EndIf

Transition preparation state 0-delay monitoring to 0-delay monitoring state

If	LC_REQ_ZDLY_MIS = 1	CARB application
And	STATE_DIAG_MIS = 1	0-delay preparation state
And	N > C_N_MIN_ZDLY_MIS	engine speed over threshold
And	CTR_TDC_ZDLY_MIS = 0	1 engine cycle passed
Then	STATE_DIAG_MIS = 2	
	LV_ZDLY_DIAG_MIS = 1	
EndIf		

Transition preparation nominal state to nominal monitoring state

If	LC_REQ_ZDLY_MIS = 0	standard application
And	STATE_DIAG_MIS = 3	preparation monitoring state
And	N > C_N_MIN_ZDLY_MIS	engine speed over threshold
And	CTR_TDC_ZDLY_MIS = 0	1 engine cycle passed
Then	STATE_DIAG_MIS = 4	
	LV_DIAG_MIS = 1	
EndIf		

Transition 0-delay monitoring state to nominal monitoring state

If	LC_REQ_ZDLY_MIS = 1	*CARB application*
And	STATE_DIAG_MIS = 2	*0-delay monitoring state*
And	N > C_N_MIN_ZDLY_MIS	*engine speed over threshold*
And	CTR_TDC_ZDLY_MIS = 0	*1 engine cycle passed*
And	CTR_T_ZDLY_MIS = 0	*timer finished (typically 5 seconds)*
Then	STATE_DIAG_MIS = 4	
	LV_DIAG_MIS = 1	
	LV_ZDLY_DIAG_MIS = 0	
EndIf		

The counter CTR_T_ZDLY_MIS is decremented at every time instance of 10 ms when STATE_DIAG_MIS != 0.

The counter CTR_TDC_ZDLY_MIS is decremented with 1 at every TDC.

(both counters are saturated to 0)

Rermark: During the 0-delay misfire detection, no fade-outs are allowed for crankshaft oscillation, rough road detection and zero-load line conditions (see appropriate sections). However, it is allowed to have so-called driver-induced fade-outs (e.g. gradient on pedal value). This latter section has been added.

At hot resets, the system starts again in the initialisation phase.

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
LC_REQ_ZDLY_MIS	1	0...1H	0...1	1	[-]
Activation of CARB-required 0-delay monitoring					
C_N_MIN_ZDLY_MIS	1	0...1FE0H	0...8160	1	[rpm]
Minimum monitoring engine speed for engine start monitoring phase					
C_T_ZDLY_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Duration for engine start specific misfire monitoring (5s according CARB)					
C_NR_TDC_ZDLY_MIS	1	0...FFH	0...255	1	[-]
Standard TDC count disablement tdc number before monitoring (engine start phase)					

B.4.3 Misfire detection legal fade out conditions

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
MIS_DET_CDN_INH	V	0...3FH	0...63	1	[-]
Misfiring detection fade out requests carrier byte (before masking)					
MIS_DET_CDN_INH_NR	V	0...FFH	0...255	1	[-]
Misfiring detection fade out requests number (after masking)					
LV_REQ_INH_MIS	V/O	0...1H	0...1	1	[-]
Misfire detection fade out request flag					
LV_PUC_DET_MIS	V	0...1H	0...1	1	[-]
Fuel Cut-Off dedicated to misfire detection, including a delay time for fuel reactivation					
LV_INH_N_MAX_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to maximum engine speed condition					
LV_INH_N_MIN_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to minimum engine speed condition					
LV_INH_LOAD_MIN_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to zero load line condition					
LV_INH_TCO_MIN_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flagdue to minimum temperature condition					
LV_INH_AMP_MIN_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flagdue to minimum atmospheric pressure condition					
LV_INH_ST_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to engine start condition					
LV_INH_IV_OFF_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out request flag due to cylinder shut-off condition					
LV_IV_OFF_DET_MIS	V	0...1H	0...1	1	[-]
Misfire detection fade out flag due to cylinder shut-off, phased with misfire detection index					

Input data:

N	LOAD_MIS	LOAD_MIN_MIS	TCO
AMP	LV_ST	SEG_NR	LC_MIS_INH
LV_AT	LV_DC	INH_INJ	TCO_ST
LV_REQ_APP_INH_MIS	LV_INJ_CUT	SUM_INH_INJ	C_N_MIN_ZDLY_MIS

FUNCTION DESCRIPTION:

General information:

Legal fade out (LV_REQ_INH_MIS) is based on all fade out conditions allowed by the legal texts for continuous misfire monitoring.

All these different fade-out conditions are merge into a global carrier (MIS_DET_CDN_INH).

Timers used for fade-out conditions delays are free running timers who are not deactivated according following conditions.

Application conditions: *(global to Misfire detection legal fade out conditions chapter)*

Initialisation: at Reset, Engine Stop, **Or** on Deactivation event

LV_PUC_DET_MIS = 0

LV_IV_OFF_DET_MIS = 1

LV_REQ_INH_MIS = 1

MIS_DET_CDN_INH = 127

MIS_DET_CDN_INH_NR = 7

Recurrence: every segment task

Activation: *LC_MIS_INH = 0*

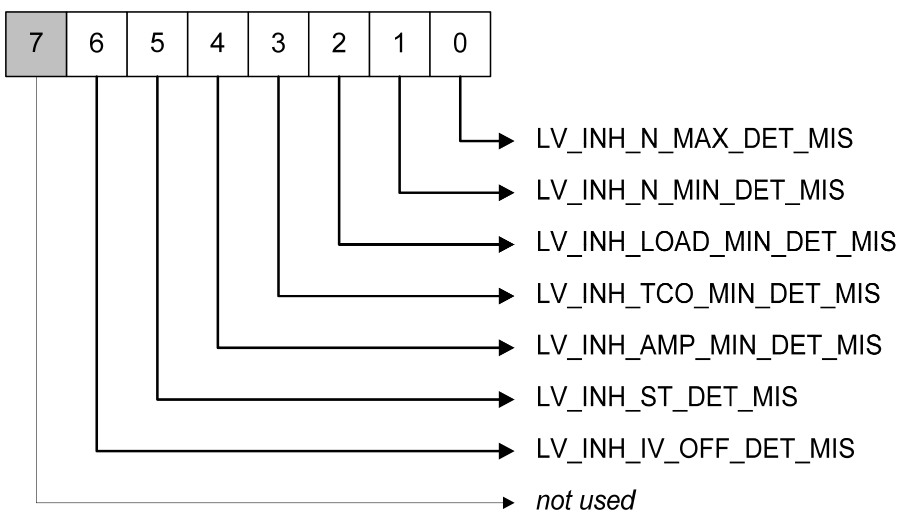
Deactivation: *LC_MIS_INH = 1*

B.4.3.1 Fade out carrier definition

Misfire fade out conditions carrier:

MIS_DET_CDN_INH

Carrier used for fade-out conditions merge
(legal fade-out overview)



B.4.3.2 Configurable Fade-out Management

Application conditions:

Fade out management during zero delay phase & nominal phase can be configurated via 2 bitfield masks that allow to take in account or not some conditions in a phase and not in the other.

If the corresponding bit in the MIS_DET_CDN_INH carrier structure is set to 0 in C_MASK_INH_DET_ZDLY_MIS or C_MASK_INH_DET_MIS calibration, then the corresponding condition will not fade-out misfire detection.

For the definition of C_MASK_INH_DET_ZDLY_MIS & C_MASK_INH_DET_MIS masks please refer to the legal requirements & customer recommendations.

Formula section:

If CTR_T_ZDLY_MIS ≠ 0

Then

MIS_DET_CDN_INH_NR = sum(MIS_DET_CDN_INH & C_MASK_INH_DET_ZDLY_MIS)

(bitfield operations)

Else

MIS_DET_CDN_INH_NR = sum(MIS_DET_CDN_INH & C_MASK_INH_DET_MIS)

(bitfield operations)

EndIf

If MIS_DET_CDN_INH_NR ≠ 0

Or LV_REQ_APP_INH_MIS = 1 *// Fade out request linked to misfire detection method*

Then LV_REQ_INH_MIS = 1 *// General fade out request*

Else LV_REQ_INH_MIS = 0

EndIf

B.4.3.3 Maximum engine speed fade-out

Application conditions:

The misfire monitoring can be inhibated when the engine speed reachs the maximum engine speed imposed by the market target : Redline engine speed for US applications or 4500 rpm for EC applications.

Moreover this information is use to stop misfire monitoring process for high speed if not required (limitation of cpu load especially for european application).

Formula section:

If (N > C_N_MAX_MIS)

Then LV_INH_N_MAX_DET_MIS = 1

Else LV_INH_N_MAX_DET_MIS = 0

EndIf

B.4.3.4 Minimum engine speed fade-out

The irregular engine operation test is performed as soon as the above-mentioned condition is disabled.

Important:

- C_N_MIN_MIS is defined by the current configuration of the index used for misfire detection in the current application. C_N_MIN_MIS should be lower than C_N_MIN_ZDLY_MIS.
- C_N_MIN_ZDLY_MIS is defined as the engine speed set-point for a warm engine minus 150 rpm. After passing below the C_N_MIN_ZDLY_MIS threshold, one is allowed one engine cycle (meaning for example for a 4-cylinder engine 4 tdc's) before reactivating the output of the misfire detection algorithm. Misfire detection is suppressed for a period of C_NR_TDC_INH_MIS tdc's.
- It should be checked in each application that within the allowed number of TDC's proper misfire detection index values have been re-obtained. Typically, it should be possible to restart misfire detection earlier.

Application conditions:

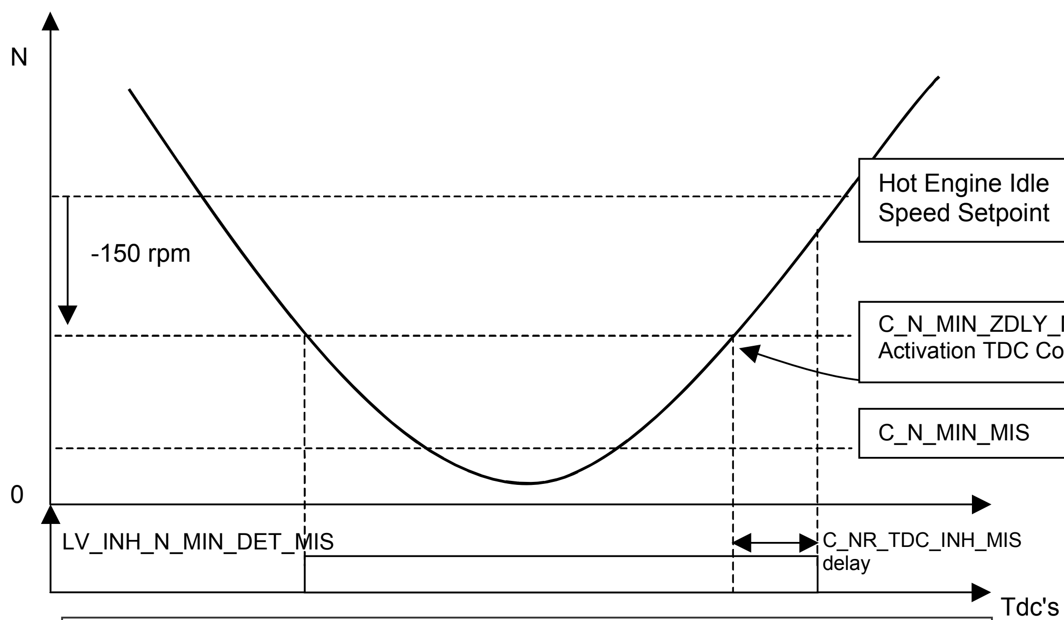
N < C_N_MIN_ZDLY_MIS

Engine speed below CARB requirements *(used for US applications)*.

Typically, once N < C_N_MIN_ZDLY_MIS, the output of the misfire detection is inhibited, allthough the calculation is still valid. C_NR_TDC_ZDLY_MIS tdc's decounter is started at passing engine speed C_N_MIN_ZDLY_MIS again. Once the decounter is at 0, misfire detection index output is allowed.

N < C_N_MIN_MIS

Engine speed below misfire detection index resolution *(used for CE applications)*. If the engine speed falls below C_N_MIN_MIS.



Explanation on the mechanism to re-activate the misfire detection after an engine speed drop

As the misfiring test is not mandatory and risky over a certain engine speed threshold, a fade out condition is tested on engine rpm (not allowed for US market).

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_N_MIN_DET_MIS = 1

Formula section:

If N < C_N_MIN_ZDLY_MIS
 Or N < C_N_MIN_MIS
Then LV_INH_N_MIN_DET_MIS = 1
Else LV_INH_N_MIN_DET_MIS = 0
 caution : see specific reactivation conditions above when N > C_N_MIN_ZDLY_MIS, C_NR_TDC_INH_MIS delay used
EndIf

B.4.3.5 Minimum engine load

Application conditions:

Below a specific load (zero load), fluctuations in engine speed cannot be detected, even in the event of misfire. In this case, the misfire detection is disabled.

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_LOAD_MIN_DET_MIS = 1

Formula section:

If LOAD_MIS < LOAD_MIN_MIS
Then LV_INH_LOAD_MIN_DET_MIS = 1
Else LV_INH_LOAD_MIN_DET_MIS = 0
EndIf

B.4.3.6 Minimum coolant temperature

Application conditions:

Misfire detection can be disabled when the coolant temperature is below a threshold (20°F / -7°C). Additionally when the coolant temperature at engine start is below this threshold, the misfire monitoring can be disable on that driving cycle until the current engine temperature reaches a warm up temperature threshold (70°F / 21°C):

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_TCO_MIN_DET_MIS = 1

Formula section:

LV_TEMP : temporary bit used to check only one occurence per engine start

Initialisation : LV_TEMP = 0 at ECU reset ***Or*** *LV_DC 0 -> 1*

If TCO < C_TCO_MIN_MIS

Or (TCO_ST < C_TCO_MIN_MIS

And TCO < C_TCO_WUP_MIN_MIS

And *LV_TEMP = 0*)

Then LV_INH_TCO_MIN_DET_MIS = 1

Else LV_INH_TCO_MIN_DET_MIS = 0

LV_TEMP = 1

EndIf

B.4.3.7 Minimum atmospheric pressure

Application conditions:

Misfire detection can be disabled when the atmospheric pressure is below a threshold:

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_AMP_MIN_DET_MIS = 1

Formula section:

If AMP < C_AMP_MIN_MIS

Then LV_INH_AMP_MIN_DET_MIS = 1

Else LV_INH_AMP_MIN_DET_MIS = 0

EndIf

B.4.3.8 After start

Application conditions:

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_ST_DET_MIS = 1

Misfire detection can be suppressed (see legal requirements - forbidden for CARB zero delay requirements) for a C_T_DLY_ST_MIS duration after the engine exits the operating state start (LV_ST), during this period LV_INH_ST_DET_MIS is set to 1, even if C_T_DLY_ST_MIS = 0.

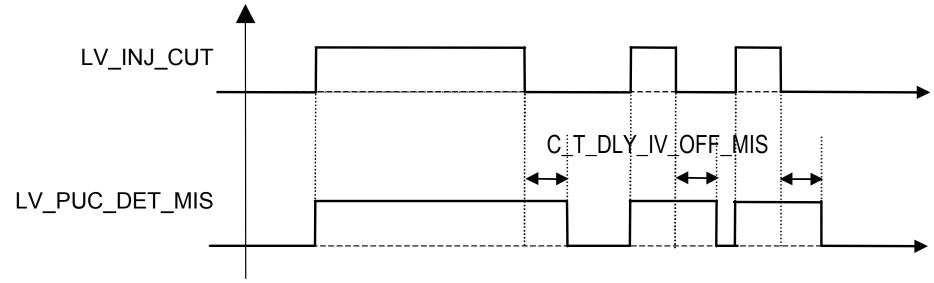
B.4.3.9 Injection shut-off

Application conditions:

Misfire detection can be suppressed for the cylinder(s), for which injection has been shut-off.

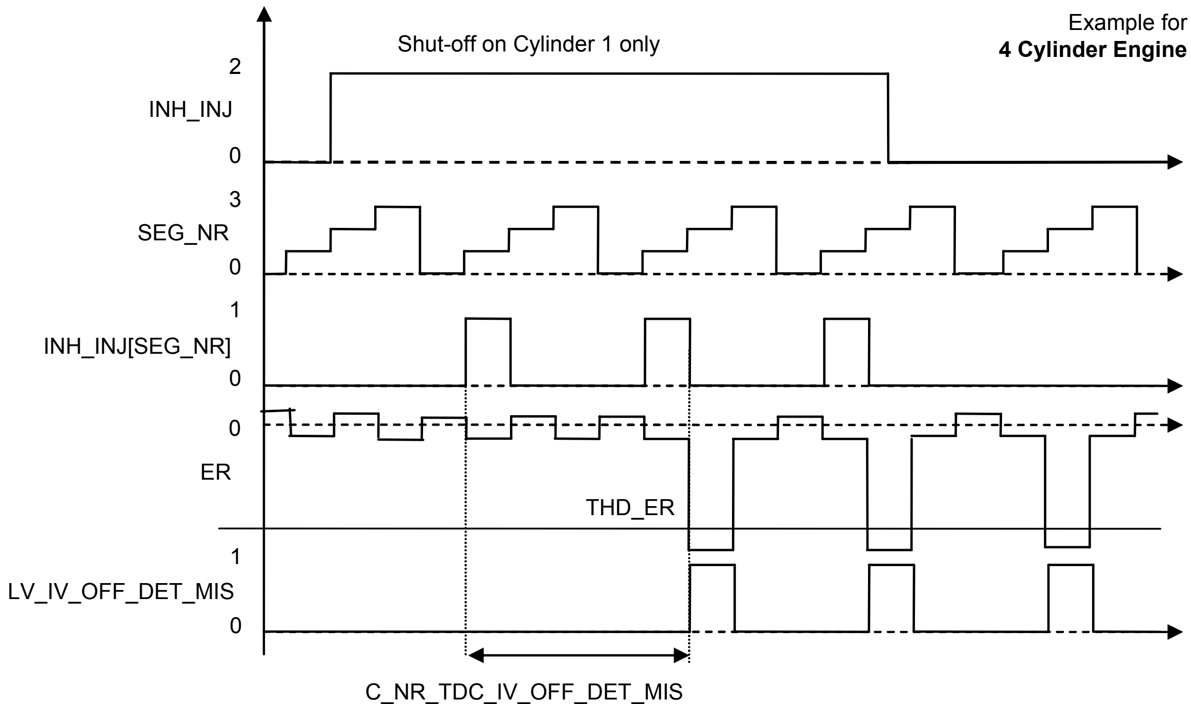
General fuel shut-off:

LV_INJ_CUT indicates a general fuel shut-off. LV_PUC_DET_MIS bit includes a calibration delay time (C_T_DLY_IV_OFF_MIS) triggered at fuel cut-off condition falling edge, in a way to start the misfire detection only after crankshaft oscillations due to fuel reactivation.



Cylinder specific shut-off:

If the current cylinder checked by misfire detection is shut-off by any EMS function even MISF limp home functionnality), a fade-out can be apply for the misfire detection of this cylinder. Due to the delay between the misfire index and the injection current informations, a calibration delay (C_NR_TDC_IV_OFF_DET_MIS) has been introduced on LV_IV_OFF_DET_MIS. This delay synchronises MISF & INJR informations in a way to apply correctly the fade-out on the proper cylinder during individual cylinder shut-off operations, and to be able the perform the misfire monitoring on the other cylinders who are not in shut-off mode.



Limit number of cylinders shut-off:

If the number of shut-off cylinders reaches a calibration data (C_SUM_INH_IV_MAX_DET_MIS), reliable misfire detection is impossible. In such case is preferable to inhibate misfire detection.

Formula section:

General fuel shut-off:

If LV_INJ_CUT = 1 *// All cylinders are shut off*

Then LV_PUC_DET_MIS = 1

Else LV_PUC_DET_MIS = 0 after C_T_DLY_IV_OFF_MIS delay time

EndIf

Cylinder fuel shut-off:

If INH_INJ[SEG_NR] = 1 *// indexed by cylinder bit position*

Then LV_IV_OFF_DET_MIS = 1 after a C_NR_TDC_IV_OFF_DET_MIS delay

Else LV_IV_OFF_DET_MIS = 0 after a C_NR_TDC_IV_OFF_DET_MIS delay

EndIf

Activation: LV_INH_N_MAX_DET_MIS = 0

Deactivation: LV_INH_N_MAX_DET_MIS = 1

Action on deactivation event: LV_INH_IV_OFF_DET_MIS = 1

Formula section:

General definition:

If LV_PUC_DET_MIS = 1

Or LV_IV_OFF_DET_MIS = 1

Or SUM_INH_INJ >= C_SUM_INH_IV_MAX_MIS

Then LV_INH_IV_OFF_DET_MIS = 1

Else LV_INH_IV_OFF_DET_MIS = 0

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_MAX_MIS	1	0...1FE0H	0...8160	1	[rpm]
Maximum engine speed for misfire detection					
C_MASK_INH_DET_MIS	1	0...FFH	0...255	1	[-]
MIS_DET_CDN_INH carrier structure fade out configuration mask (misfire detection nominal phase)					
C_AMP_MIN_MIS	1	0...FFFFH	0...5434	0.0829175	[hPa]
Minimum atmospheric pressure for misfire detection					
C_N_MIN_MIS	1	0...1FE0H	0...8160	1	[rpm]
Minimum engine speed for misfire detection.					
C_N_MAX_MIS	1	0...1FE0H	0...8160	1	[rpm]
Maximum engine speed for misfire detection					
C_TCO_MIN_MIS	1	0...FEH	-48...142.5	0.75	[°C]
Minimum coolant temperature for misfire detection					
C_TCO_WUP_MIN_MIS	1	0...FEH	-48...142.5	0.75	[°C]
Minimum warm up coolant temperature for misfire detection					
C_T_DLY_ST_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration when engine operating state start LV_ST has been exited.					
C_T_DLY_IV_OFF_MIS	1	0...FFFFH	0...655.35	0.01	[s]
Fade out duration after injection is back on.					
C_NR_TDC_INH_MIS	1	0...FFH	0...255	1	[-]
Fade out duration in TDC when engine speed rise up to C_N_MIN_ZDLY_MIS					
C_SUM_INH_IV_MAX_MIS	1	0...8H	0...8	1	[-]
Maximum number of cylinders in shut-off to fade-out					
C_NR_TDC_IV_OFF_DET_MIS	1	0...7H	0...7	1	[-]
TDC delay to synchronise cylinder shut-off information with the misfire detection index					

B.5 Misfire rate determination and error management

General information:

The misfire rate determination and misfire specific error management are applied to the following CARB misfire legal definitions:

- **CARB A misfire failure criterion** *(LV_MIS_STATE_A)*:

Risk of catalyst damage, monitoring interval over 200 crankshaft revolutions (= 200 * NC_CYL_NR / 2 TDCs) during the driving cycle.

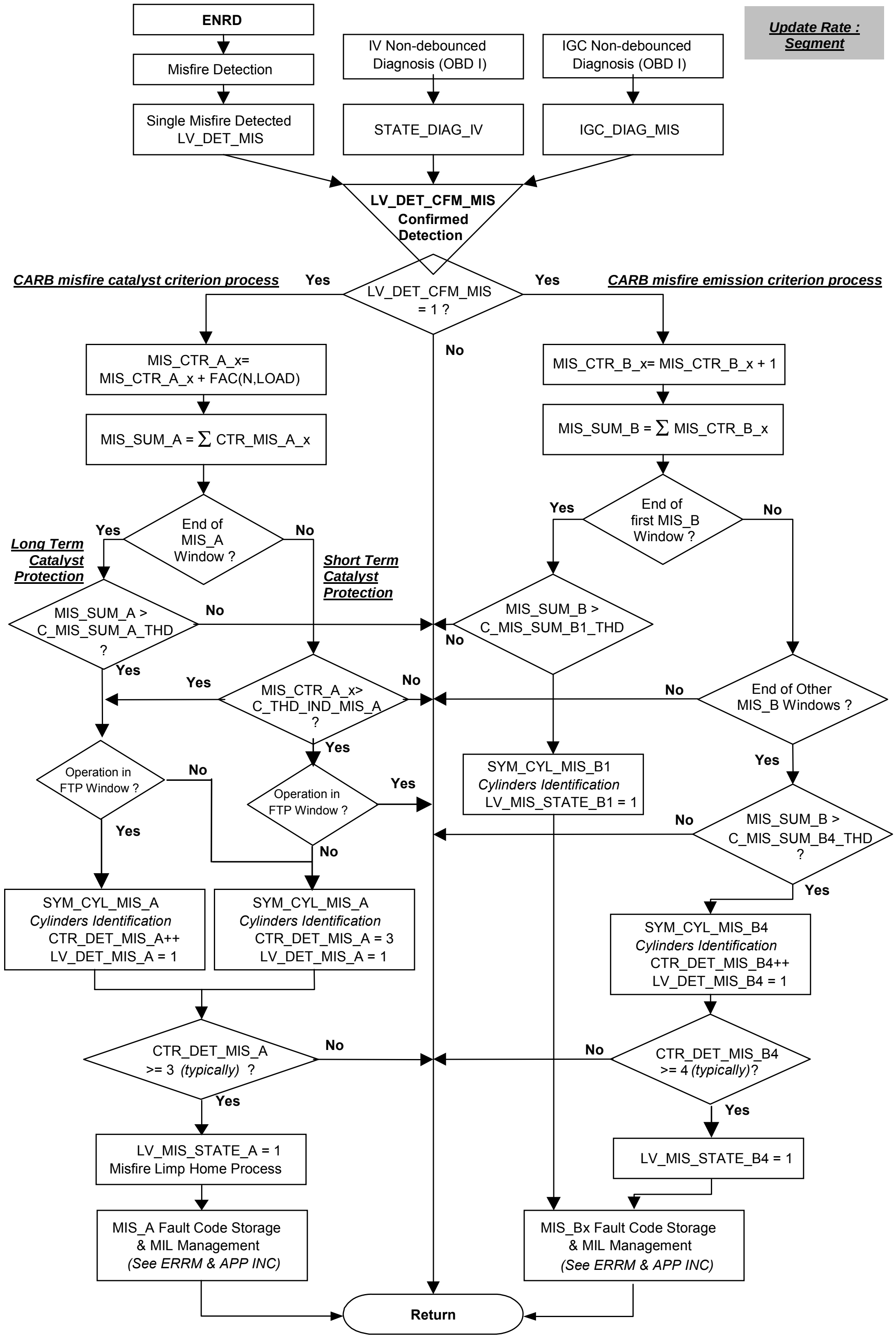
- **CARB B1 misfire failure criterion** *(LV_MIS_STATE_B1):*

Emission increase, monitoring interval over the first 1000 crankshaft revolutions (= 1000 * NC_CYL_NR / 2 TDCs) of the driving cycle.

- **CARB B4 misfire failure criterion** *(LV_MIS_STATE_B4):*

Emission increase, monitoring interval over 1000 crankshaft revolutions (= 1000 * NC_CYL_NR / 2 TDCs). For criterion confirmation, misfire must be take place for 4 monitoring intervals (consecutive or not).

MIS_A, MIS_B1 & MIS_B4 Handling Flowchart diagram:



B.5.1 Combination of different diagnosis reliability levels (misfire detection, ignition, injection)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SUM_FAC_DIAG_MIS	V	0...2FDH	0...5.97656	0.0078125	[-]
Sum of the weighting factor from diagnosis combinations for final misfire decision.					
FAC_ER_DIAG_MIS	V	0...FFH	0...1.99218	0.0078125	[-]
Reliability weighting factor for detection based on misfire detection index (ER)					
FAC_IGC_DIAG_MIS	V	0...FFH	0...1.99218	0.0078125	[-]
Reliability weighting factor for ignition diagnosis (ignition failure detected)					
FAC_IV_DIAG_MIS	V	0...FFH	0...1.99218	0.0078125	[-]
Reliability weighting factor for injection valve (injection valve failure detected)					
LV_DET_CFM_MIS	V/O	0...1H	0...1	1	[-]
Flag for individual confirmed misfire detection after crossed diagnosis					

Input data:

LV_DET_MIS	STATE_DIAG_IV	IGC_DIAG_MIS	LOAD_MIS
N_32	INH_IV_MIS	SEG_NR_ER	LV_HOM_AFL_ACT
LV_S_ACT	LV_INH_IGC_DIAG_MIS	LV_INH_IV_DIAG_MIS	NC_CMB_CONF
LC_MIS_INH			

FUNCTION DESCRIPTION:

General information:

Due to their physical principle, the individual diagnostic functions have a different diagnostic relevance depending on the operating point.

To achieve the best possible detection of misfires, the individual diagnostic methods are combined. Here with an assumed misfire of a part function, an operating point dependent weighting factor, which reflects the reliability of this part function in the current operating point, is taken from a map. If the sum of these three values now exceeds a threshold for one combustion, then misfire is detected.

The value which is not debounced is used as input value of the IV and IGCFB diagnoses. The injection for a cylinder is cut off exclusively by the CARB error mechanism and not by the IV and IGCFB diagnostic functions.

Application conditions:

Misfire segment after detection, before CARB diagnosis

Application conditions:

Update rate: every segment

Activation:

LC_MIS_INH = 0

Deactivation:

LC_MIS_INH = 1

Initialisation on ECU reset, on LV_DC 0 to 1 transition and on LC_MIS_INH 0 to 1 transition:

SUM_FAC_DIAG_MIS = 0

FAC_ER_DIAG_MIS = 0

FAC_IGC_DIAG_MIS= 0

FAC_IV_DIAG_MIS = 0

LV_DET_CFM_MIS = 0

Formula section:

The output from the misfire detection, injection and ignition diagnosis functions is influenced by weighting interpolated table, giving the reliability of the individual functions.

- A weighting value of **0** means that detection is not possible for this engine operating point.

- A value equal to or greater than **1** indicates reliable detection.

- The values in between represent the corresponding levels of reliability.

- Irregular engine operation weighting (LV_DET_MIS = 1):

If misfiring is suspected by the irregular engine operation test, a weighting factor for the corresponding engine operating point is applied. This factor is derived from tables according combustion mode.

Convention for combustion mode determination:

#If NC_CMB_CONF = AFS_AFL Or AFS_AFL_S

If OPM_AV = 3 (homogeneous-stratified)

Then xx = "HOM_S"

EndIf

#EndIf

#If NC_CMB_CONF = AFS_S Or AFS_AFL_S

If OPM_AV = 1 (stratified)

Then xx = "S"

EndIf

#EndIf

If OPM_AV <> 1 and OPM_AV <> 3 (homogeneous)

Then xx = "AFS"

EndIf

If LV_DET_MIS = 1

Then FAC_ER_DIAG_MIS = IP_FAC_ER_xx_DIAG_MIS(N_32, LOAD_MIS)

Else FAC_ER_DIAG_MIS = 0

EndIf

- Ignition diagnosis weighting (IGC_DIAG_MIS):

IGC_DIAG_MIS is the not debounced carrier from Ignition diagnosis, which has to be delayed due to misfire detection *(according segment reference SEG_NR_ER-1, cylinder x compression phase)*. If misfiring is suspected by the ignition diagnosis function, a weighting factor for the corresponding engine operating point is applied.

This factor is derived from IP_FAC_IGC_DIAG_MIS table.

For the time the number of cylinders shut off due to misfire detection limp home reaches or exceeds the applicable number of C_MIS_NR_OFF_IV_IGCFB, FAC_IGC_DIAG factor is forced to ‘’1’’ *(test based on INH_IV_MIS carrier)*.

If(1) IGC_DIAG_MIS[SEG_NR_ER] = 1

And LV_INH_IGC_DIAG_MIS = 0

Then(1)

If(2) number of bits set to "1" in INH_IV_MIS carrier structure >= C_MIS_NR_OFF_IV_IGCFB

Then(2) FAC_IGC_DIAG_MIS = 1

Else(2) FAC_IGC_DIAG_MIS = IP_FAC_IGC_DIAG_MIS(N_32, LOAD_MIS)

EndIf(2)

Else(1) FAC_IGC_DIAG_MIS = 0

EndIf(1)

- Injection diagnosis weighting (STATE_DIAG_IV) :

STATE_DIAG_IV is a not debounced carrier from Injection valves diagnosis.

Injection valve in default is identifed according segment reference SEG_NR_ER within the bitfield structure STATE_DIAG_IV.

If misfiring is suspected by the injection valve diagnosis function, a weighting factor for the corresponding engine combustion mode is applied.

If STATE_DIAG_IV[SEG_NR_ER] = 1

And LV_INH_IV_DIAG_MIS = 0

Then FAC_IV_DIAG_MIS = IP_FAC_IV_DIAG_MIS(N_32, LOAD_MIS)

Else FAC_IV_DIAG_MIS = 0

EndIf

- Formation of a cylinder - specific diagnosis weighting sum :

The sum of the three weighting values SUM_FAC_DIAG_MIS is performed for the individual cylinder. as soon as one of the three diagnosis function suspects misfiring.

SUM_FAC_DIAG_MIS = FAC_ER_DIAG_MIS

+ FAC_IGC_DIAG_MIS

+ FAC_IV_DIAG_FAC

If SUM_FAC_DIAG_MIS >= 1

Then LV_DET_CFM_MIS = 1 *misfire status is confirmed*

Else LV_DET_CFM_MIS = 0 *no misfire*

EndIf

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_MIS_NR_OFF_IV_IGCFB	1	0...8H	0...8	1	[-]
Threshold for number of cylinders shut off due to misfire					
IP_FAC_ER_AFS_DIAG_MIS	6*6	0...FFH	0...1.99218	0.0078125	[-]
LDP_N_32_IP_FAC_ER_AFS_DIAG_MIS	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_ER_AFS_DIAG_MIS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for irregular engine operation (engine speed analysis) in AFS combustion mode					
IP_FAC_IGC_DIAG_MIS	6*6	0...FFH	0...1.99218	0.0078125	[-]
LDP_N_32_IP_FAC_IGC_DIAG_MIS	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_IGC_DIAG_MIS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for ignition diagnosis (ignition failure detected).					
IP_FAC_IV_DIAG_MIS	6*6	0...FFH	0...1.99218	0.0078125	[-]
LDP_N_32_IP_FAC_IGC_DIAG_MIS	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_MIS_IP_FAC_IV_DIAG_MIS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for injection valve diagnosis (injection valve failure detected).					
#IF NC_CMB_CONF = AFS_AFL Or AFS_AFL_S					
IP_FAC_ER_HOM_S_DIAG_MIS	6*6	0...FFH	0...1.99218	0.0078125	[-]
LDP_N_32_IP_FAC_ER_HOM_S_MIS	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_ER_HOM_S_MIS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for irregular engine operation (engine speed analysis) in HOM_S combustion mode					
#ENDIF					
#IF NC_CMB_CONF = AFS_S Or AFS_AFL_S					
IP_FAC_ER_S_DIAG_MIS	6*6	0...FFH	0...1.99218	0.0078125	[-]
LDP_N_32_IP_FAC_ER_S_DIAG_MIS	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_IP_FAC_ER_S_DIAG_MIS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Weighting factor for irregular engine operation (engine speed analysis) in S combustion mode					
#ENDIF					

B.5.2 Misfire detection counters (ISO 15031 Data)

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
CTR_MIS_TOT_NVMY	V/O/S	0...FFFFFFFFH	0...4294967295	1	[-]
Overall misfire detection counter					
CTR_MIS_TOT_DC	V/O/S	0...FFFFH	0...65535	1	[-]
Former / current driving cycle overall misfire detection counter					
CTR_MIS_DC_CYL[NC_CYL_NR]	V/O/S	0...FFFFH	0...65535	1	[-]
Former / current driving cycle misfire detection cylinder counters (ISO15031 Data $0C)					
CTR_MIS_DC_MMV_CYL[NC_CYL_NR]	V/O/S	0...FFFFH	0...65535	1	[-]
Exponential weighted moving average (EWMA) misfire counters (ISO15031 Data $0B)					

Input data:

LV_DET_CFM_MIS	LV_ENA_ER	LC_MIS_INH	SEG_NR_ER
LV_DC			

FUNCTION DESCRIPTION:

General information:

These counters are incremented at each misfire detection based on misfire index evaluation *(after crossed diagnosis with IGC & IV OBDI diags evaluation)*. Thus, it's possible to determine the total number of misfire detected during engine lifetime and during current driving cycle.

Moreover, cylinder misfire counters have been introduced for Mode 6 communication service, as described in last ISO specification available (ISO/DIS 15031-5.8).

Note: CTR_MIS_TOT_DC & CTR_MIS_DC_CYL[NC_CYL_NR] are initialised only at engine running to keep the information after engine stop.

B.5.2.1 NVMY Data Formating

On non volatile memory formatting **or** on NV memory corruption detection:

CTR_MIS_TOT_NVMY = 0

On non volatile memory formatting **or** on NV memory corruption detection **or** failure memory clearing :

CTR_MIS_TOT_DC = 0

For(1) x = 0 : NC_CYL_NR-1

  CTR_MIS_DC_CYL[x] = 0

  CTR_MIS_DC_MMV_CYL[x] = 0

EndFor(1)

B.5.2.2 Initialisation of former / current driving cycle counters

Initialisations:

- at ECU reset
  NVMY data loading
- at LV_DC 0 -> 1 transition event
  CTR_MIS_TOT_DC = 0
  For(1) x = 0 : NC_CYL_NR-1
  CTR_MIS_DC_CYL[x] = 0
  EndFor(1)

B.5.2.3 Increment of misfire counters during current driving cycle

Application conditions:

Activation:	*LV_ENA_ER = 1*	
	And	*LC_MIS_INH = 0*
Deactivation:	*LV_ENA_ER = 0*	
	Or	*LC_MIS_INH = 1*
Update rate:	Segment task	

Formula section:

If LV_DET_CFM_MIS = 1

Then CTR_MIS_TOT_DC = CTR_MIS_TOT_DC + 1 *// with saturation*

CTR_MIS_TOT_NVMY = CTR_MIS_TOT_NVMY + 1 *// with saturation*

CTR_MIS_DC_CYL[SEG_NR_ER] = CTR_MIS_DC_CYL[SEG_NR_ER] + 1 *// with saturation*

EndIf

B.5.2.4 Update of the EWMA misfire counters

Exponential weighted moving average (EWMA) misfire counters are updated at the beginning of the driving cycle based on former driving cycle values.

Update rate: at LV_DC 0 -> 1 transition event *before CTR_MIS_DC_CYL[x] reset*

For x = 0 : NC_CYL_NR-1

CTR_MIS_DC_MMV_CYL[x] =

0.9 * CTR_MIS_DC_MMV_CYL[x] + 0.1 * CTR_MIS_DC_CYL[x]

EndFor

B.5.3 Exhaust cylinder bank reference for misfire detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SEG_NR_CBK_ER	V	0...FFH	0...255	1	[-]
Exhaust cylinder bank reference for misfire monitoring					

Input data:

SEG_NR_ER	LV_SYN_ENG	NC_CBK_EX_NR	NC_CYL_NR

FUNCTION DESCRIPTION:

For catalyst damage and/or emission criterion, misfire diagnosis functions can be set via calibration data in a way to be able to detect criterions according exhaust cylinder bank phasing instead of a global criterion detection.

Exhaust cylinder bank reference for misfire detection allows to increase these criterions on the dedicated cylinder bank weighting factor if the misfire detected has effect on the impacted cylinder bank and not on the other (if multiple exhaust cylinder bank design).

Description:

The exhaust cylinder bank reference SEG_NR_CBK_ER is determinate according the segment reference SEG_NR_ER. It already includes the delay introduced by the misfire index chosen.

Application conditions:

Update rate: Segment task

Activation: *LV_SYN_ENG = 1*

Deactivation: *LV_SYN_ENG = 0*

Initialisation on Deactivation condition:

SEG_NR_CBK_ER = 0

Initialisation: at ECU reset

SEG_NR_CBK_ER = 0

Formula section:

SEG_NR_CBK_ER = NC_CBK_EX_NR_MISF[SEG_NR_ER]

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CBK_EX_NR_MISF[NC_CYL_NR]	-	0...FFH	0...255	1	[-]
Exhaust cylinder bank allocation according misfire segment phasing					

B.5.4 Determination of CARB A misfire criterion, causing catalyst damage

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DET_MIS_A	V	0...1H	0...1	1	[-]
CARB A misfire criterion detected in actual 200 rev. window (not debounced)					
LV_MIS_STATE_A	V/O	0...1H	0...1	1	[-]
CARB A misfire criterion confirmed (debounced)					
LV_INH_IV_MIS_A	V/O	0...1H	0...1	1	[-]
At least one cylinder is shut-off by a CARB A misfire criterion confirmed					
CTR_MIS_A_LIH_CYL[NC_CYL_NR]	V	0...FFH	0...255	1	[-]
Cylinder dedicated counters for CARB A Limp-home phases on current DC					
MIS_A_FAC	V	0...64H	0...100	1	[-]
Catalyst damage weighting factor to increment cylinder specific counters when misfire is detected					
MIS_A_FAC_BUF[NC_SIZE_THD_ER_BUF]	V	0...64H	0...100	1	[-]
Catalyst damage weighting factor buffer					
INH_IV_MIS	V/O	0...FFH	0...255	1	[-]
Identification of cylinders shut off by misfire CARB A					
MIS_CTR_A[NC_CYL_NR]	V	0...FFFFH	0...65535	1	[-]
Misfire sums (cylinder individual) after combination and weighting in CARB A window.					
MIS_NR_TDC_A	V	0...FFFFH	0...65535	1	[-]
TDC counter CARB A window (200 crankshaft revolutions).					
MIS_SUM_A	V	0...FFFFH	0...65535	1	[-]
CARB A misfire weighted sum (global)					
MIS_SUM_A_CBK[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[-]
CARB A misfire weighted sum (dedicated to Exhaut cylinder bank)					
CTR_CHK_MIS_A	V	0...FFH	0...255	1	[-]
CARB A windows occurrence counter					
CTR_FTP_CDN_MIS_A	V	0...FFFFH	0...65535	1	[-]
TDC counter for FTP condition detection					
CTR_DET_MIS_A	V	0...FFH	0...255	1	[-]
CARB A misfire criterion detected counter					
LV_FTP_MIS_A	V	0...1H	0...1	1	[-]
Boolean for FTP emission cycle condition status at the end of CARB A window					
LV_END_MIS_A	V/O	0...1H	0...1	1	[-]
CARB A misfire criterion determination end					
LV_END_WIN_MIS_A	V/O	0...1H	0...1	1	[-]
End of CARB A window for similar condition					
LV_ERR_IN_WIN_MIS_A	V/O	0...1H	0...1	1	[-]
End during CARB A window for similar condition					
SYM_CYL_MIS_A	V/O	0...FFFFH	0...65535	1	[-]
CARB A misfire criterion symptoms					

Input data:

N	LOAD_MIS	LV_DET_CFM_MIS	LV_CDN_MIS_A
LV_DC	NC_CYL_NR	LV_REQ_APP_CTR_MIS_A	INH_INJ
SEG_NR_ER	SEG_NR_CBK_ER	LV_ES	LV_PUC
LC_MIS_INH	LV_MIS_A_DIAG_REQ_APP	FAC_MIS_A_APP	LV_AT
NC_CBK_EX_NR	FAC_MIS_A_THD_IND_APP	FAC_MIS_SUM_A_THD_APP	

Description:

With a CARB A misfire rate criterion, the emissions are increased and there is a risk of catalyst damage if the cylinder causing the failure criterion can not be shut off.

Detection:

The check for misfire rate violation **CARB A** comprises 200 crankshaft revolutions *(100 * NC_CYL_NR tdc's)* for statistic misfiring and cyclewise monitoring for best possible catalyst protection.

If a confirmed misfiring is detected on ER current TDC (LV_DET_CFM_MIS=1), a weighting factor (MIS_A_FAC) is added to the cylinder-specific counter MIS_CTR_A[SEG_NR_ER], extracted from an interpolated table *(see APP. INC.)* who depends on engine combustion mode and current operating point. The increment represents the severity of catalyst damage.

Statistic:

The monitoring for Misfire **CARB A** is performed continuously in intervals of 200 crankshaft revolutions.

Monitoring during CARB A window:

To guarantee best possible catalyst protection, the cylinder-specific counters are monitored each cycle. For that purpose, MIS_CTR_A[SEG_NR_ER] is compared to the applicable value C_MIS_A_IND_THD representing the critical threshold for single cylinder misfiring.

If the threshold is reached outside of the FTP cycle, the concerning cylinder is shut off immediately. CARB A misfire criterion is confirmed, else the criterion will be evaluated at the end of the 200 crankshaft revolution window.

Monitoring at the end of 200 crankshaft revolutions:

At the end of each monitoring interval the cylinder-specific sums MIS_CTR_A[NC_CYL_NR] and the global sum (MIS_SUM_A) are available.

If the sum MIS_SUM_A is greater than the threshold C_MIS_SUM_A_THD, the cylinder with the highest misfire rate is determined (by comparing the different MIS_CTR_A[NC_CYL_NR] values).

Then operation inside/outside of the FTP-area is checked by comparison of the counter CTR_FTP_CDN_MIS_A / MIS_NR_TDC_A with the threshold C_RATIO_FTP_CDN_MIS_A. If the FTP tdc's ratio is lower than the threshold, not enough operating points have been inside the FTP-window and therefore operation outside FTP-area is detected (LV_FTP_MIS_A=0). This is done in order to fulfil the CARB-regulations, which require MIL-illumination at first threshold exceeding outside the FTP-area and at third exceeding inside the FTP-area.

Therefore the counter CTR_DET_MIS_A is incremented inside the FTP-area at each threshold exceeding. That means it reaches the maximum of C_NR_DET_MAX_MIS_A = 3 only if the threshold is exceeded in 3 succeeding monitoring intervals.

Outside the FTP-area the failure CARB A is validated immediately. The counter is reset only after an ECU-reset.

Even if we are inside the FTP-area, a specific application request (according catalyst temperature or else) LV_REQ_APP_CTR_MIS_A can force to validate immediately the failure CARB A *(see legal requirements & customer wishes).* This request is defined in associated Application Incidences file.

Note: cylinder and Exhaut cylinder bank specific counters are respectively indexed with SEG_NR_ER and SEG_NR_CBK_ER phase reference counters to take in account ERND delay in misfire monitoring process

Application conditions:

Initialisation: *at ECU reset, at LV_DC 0 - > 1 transition, at LC_MIS_INH 0 to 1 transition & at MIS_A or MIS_[NC_CYL_NR] errors clearing*

SYM_CYL_MIS_A = 0

INH_IV_MIS = 0

MIS_NR_TDC_A = 0

MIS_SUM_A = 0

CTR_FTP_CDN_MIS_A = 0

CTR_DET_MIS_A = 0

CTR_CHK_MIS_A = 0

LV_END_WIN_MIS_A = 0

LV_END_MIS_A = 0

LV_FTP_MIS_A = 0

LV_MIS_STATE_A = 0

LV_INH_IV_MIS_A = 0

LV_DET_MIS_A = 0

LV_ERR_IN_WIN_MIS_A = 0

For x = 0 : NC_CYL_NR-1

MIS_CTR_A[x] = 0 *// All cylinder specific counters*

CTR_MIS_A_LIH_CYL[x] = 0 *// All cylinder specific counters*

EndFor

For i = 0 : NC_CBK_EX_NR-1

MIS_SUM_A_CBK[i] = 0 *// All Exhaut cylinder bank specific counters*

EndFor

Update rate: segment task

Activation:

LV_DC = 1

And LC_MIS_INH = 0

Deactivation:

LV_DC = 0

Or LC_MIS_INH = 1

B.5.4.1 Determination of catalyst damage weighting factor

Catalyst damage weighting factor is determinated according engine combustion mode & buffered to take in account admission phase (SEG_NR_ER - 2).

MIS_A_FAC_BUF is managed as a rotary buffer, MIS_A_FAC_BUF[0] is the weighting factor for the actual segment value.

At each ER segment recurrence:

Convention for combustion mode determination:

#If NC_CMB_CONF = AFS_AFL Or AFS_AFL_S

If OPM_AV = 3 (homogeneous-stratified)

Then xx = "HOM_S"

EndIf

#EndIf

#If NC_CMB_CONF = AFS_S Or AFS_AFL_S

If OPM_AV = 1 (stratified)

Then xx = "S"

EndIf

#EndIf

If OPM_AV <> 1 and OPM_AV <> 3 (homogeneous)

Then xx = "AFS"

EndIf

MIS_A_FAC_BUF is managed as a rotary buffer

If LV_MIS_A_DIAG_REQ_APP = 0

Then MIS_A_FAC_BUF[current_position] = IP_MIS_A_FAC_xx(N, LOAD_MIS)

Else MIS_A_FAC_BUF[current_position] = FAC_MIS_A_APP

EndIf

MIS_A_FAC = MIS_A_FAC_BUF[oldest_position] *// points on the oldest value of the buffer*

B.5.4.2 CARB A window management (main process)

If(1) LV_CDN_MIS_A = 1 *// MIS_A monitoring active (see App Inc file)*

Then(1)

If(2) LV_DET_CFM_MIS = 1 *// Individual misfire detected*

Then(2) Call **'CARB A Misfire counters increment process'** *(see description below)*

Else(2) MIS_A Misfire counters unchanged

EndIf(2)

If(2) MIS_NR_TDC_A < 100 * NC_CYL_NR

Then(2) MIS_NR_TDC_A = MIS_NR_TDC_A + 1

LV_END_WIN_MIS_A = 0 *// End of window not reached*

If(3) LOAD_MIS < C_LOAD_MIS_FTP

And N < C_N_MIS_FTP *// FTP window conditions*

Then(3) CTR_FTP_CDN_MIS_A = CTR_FTP_CDN_MIS_A + 1

EndIf(3)

call **'Monitoring process during CARB A window'** *(see description below)*

Else(2) call **'Monitoring process at CARB A window end'** *(see description below)*

call **'CARB A criterion end detection process'** *(see description below)*

call **'CARB A window reset process'** *(see description below)*

EndIf(2)

Else(1) *no process* *// Counters & flags unchanged*

EndIf(1)

B.5.4.3 CARB A Misfire counters increment process

MIS_SUM_A = MIS_SUM_A + MIS_A_FAC

MIS_SUM_A_CBK[SEG_NR_CBK_ER] =

MIS_SUM_A_CBK[SEG_NR_CBK_ER] + MIS_A_FAC

// Assign only ER current specific Exhaut cylinder bank counter

MIS_CTR_A[SEG_NR_ER] = MIS_CTR_A[SEG_NR_ER] + MIS_A_FAC

// Assign only ER current specific cylinder counter

B.5.4.4 Monitoring process during CARB A window *(short term catalyst protection & limp home)*

If(1) **{ [** LV_MIS_A_DIAG_REQ_APP = 0

And ((LV_AT = 0 **And** MIS_CTR_A[SEG_NR_ER] >= C_MIS_A_IND_THD_MT)

Or

(LV_AT = 1 **And** MIS_CTR_A[SEG_NR_ER] >=C_MIS_A_IND_THD_AT) **)]**

// Test based only on ER current specific cylinder counter

Or [LV_MIS_A_DIAG_REQ_APP = 1

And ((LV_AT = 0 **And** MIS_CTR_A[SEG_NR_ER] >=

FAC_MIS_A_THD_IND_APP * C_MIS_A_IND_THD_MT)

Or (LV_AT = 1 **And** MIS_CTR_A[SEG_NR_ER] >=

FAC_MIS_A_THD_IND_APP * C_MIS_A_IND_THD_AT) **)] }**

Then(1)

Short term FTP conditions check

If(2) **[** LV_AT = 1 **And**

CTR_FTP_CDN_MIS_A/MIS_NR_TDC_A < C_RATIO_FTP_CDN_IND_MIS_AT **]**

Or [LV_AT = 0 **And**

CTR_FTP_CDN_MIS_A/MIS_NR_TDC_A < C_RATIO_FTP_CDN_IND_MIS_MT **]**

// Out of FTP cycle inside CARB A window

Then(2) Set concerned cylinder bit in structure SYM_CYL_MIS_A to 1

LV_DET_MIS_A = 1 *// Misfire CARB A criterion detected*

LV_MIS_STATE_A = 1 *// Misfire CARB A criterion confirmed*

LV_ERR_IN_WIN_MIS_A = 1

CTR_DET_MIS_A = max(CTR_DET_MIS_A+1, C_NR_DET_MAX_MIS_A)

LV_END_MIS_A = 1

// Substract Current cylinder MIS_A & MIS_SUM_A_CBK counters on global & cylinder bank MIS_A sums to avoid wrong global MIS_A detection at end of current Long term diagnosis

MIS_SUM_A = MIS_SUM_A - MIS_CTR_A[SEG_NR_ER]

MIS_SUM_A_CBK[SEG_NR_CBK_ER] =

MIS_SUM_A_CBK[SEG_NR_CBK_ER] - MIS_CTR_A[SEG_NR_ER]

MIS_CTR_A[SEG_NR_ER] = 0

// Avoid to retrigger criterion until end of the current window

If(3) LC_ENA_MIS_A_IND_LIH = 1

Then(3)

Shut off cylinder indexed by SEG_NR_ER via INH_IV_MIS carrier

Evaluate **'Misfire Cylinder Limp Home process'**

(see description afterwards)

EndIf(3)

Else(2) no operation *// MIS_A criterion evaluated at the end of MIS_A window or*
// at the next cylinder evaluation

EndIf(2)

Else(1) *flags & data unchanged*

EndIf(1)

#If(1) NLC_ENA_SCDN_NEW = 1

In this case, we do exactly the same operations that those done in "**'Monitoring process at CARB A window end'.**

#EndIf(1)

B.5.4.5 Monitoring process at CARB A window end *(long term catalyst protection & limp home)*

Global CARB A FTP conditions check

If **[** LV_AT = 1 **And**
CTR_FTP_CDN_MIS_A / MIS_NR_TDC_A >= C_RATIO_FTP_CDN_MIS_AT **]**
Or **[** LV_AT = 0 **And**
CTR_FTP_CDN_MIS_A / MIS_NR_TDC_A >= C_RATIO_FTP_CDN_MIS_MT **]**

Then LV_FTP_MIS_A = 1

Else LV_FTP_MIS_A = 0

EndIf

Criterion detection process

// Be carefull, the counter of window A (CTR_CHK_MIS_A) is incremented only at the end of the window and not during.

CTR_CHK_MIS_A = CTR_CHK_MIS_A + 1

If(1) **{** LV_MIS_A_DIAG_REQ_APP = 0
And **[** **(** LV_AT = 0 **And**
(MIS_SUM_A >= C_MIS_SUM_A_THD_MT
Or MIS_SUM_A_CBK[0...NC_CBK_EX_NR-1]
>= C_MIS_SUM_A_CBK_THD_MT) *// Check for each exhaust CBK* **)**
Or **(** LV_AT = 1 **And**
(MIS_SUM_A >= C_MIS_SUM_A_THD_AT
Or MIS_SUM_A_CBK[0...NC_CBK_EX_NR-1]
>= C_MIS_SUM_A_CBK_THD_AT) *// Check for each exhaust CBK* **)] }**
Or **{** LV_MIS_A_DIAG_REQ_APP = 1
And **[** **(** LV_AT = 0 **And**
MIS_SUM_A >= FAC_MIS_SUM_A_THD_APP * C_MIS_SUM_A_THD_MT
Or **(** LV_AT = 1 **And**
MIS_SUM_A >= FAC_MIS_SUM_A_THD_APP * C_MIS_SUM_A_THD_AT**)] }**

Then(1)

If(2) LV_ERR_IN_WIN_MIS_A = 0

Then(2)

LV_DET_MIS_A = 1

LV_ERR_IN_WIN_MIS_A = 1

As soon as the misfire criterion detected counter reached his threshold, we directly set the error. To avoid to stored it in the same window a second time, we used *lv_err_in_win_mis_a flag.*

If(3) LV_FTP_MIS_A = 0 *// Outside of FTP cycle*

Or LV_REQ_APP_CTR_MIS_A = 1 *// Application specific request*

Then(3) CTR_DET_MIS_A=max(CTR_DET_MIS_A+1, C_NR_DET_MAX_MIS_A)

Else(3) CTR_DET_MIS_A = CTR_DET_MIS_A + 1

EndIf(3)

If(3) CTR_DET_MIS_A >= C_NR_DET_MAX_MIS_A

Then(3) ***// Global MIS_A default***

LV_MIS_STATE_A = 1

// MIS_A cylinder identification

For(4) x = 0 to NC_CYL_NR - 1 *// Check for each cylinder x*

If(5) MIS_CTR_A[x] > C_MIS_A_MIN_NR * MIS_SUM_A

Then(5) Set in SYM_CYL_MIS_A the cylinder x concerned bit to 1

If(6) LC_ENA_MIS_A_LIH = 1 **And**

[LV_AT = 0 **And**

CTR_DET_MIS_A >= ID_NR_DET_ENA_MIS_A_LIH_MT(N_32) **]**

Or **[** LV_AT = 1 **And**

CTR_DET_MIS_A >= ID_NR_DET_ENA_MIS_A_LIH_AT(N_32) **]**

Then(6) Shut off cylinder x via INH_IV_MIS carrier

Evaluate **'Misfire Cylinder Limp Home process'**

(see description afterwards)

Else(6) *No operation*

EndIf(6)

EndIf(5)

EndFor(4)

// MIS_A random cylinder pattern

If(4) No specific cylinder counters comply with C_MIS_A_MIN_NR * MIS_SUM_A criterion

Then(4)

If(5) LC_CONF_RND_DET_MIS_A = 0

Then(5) Evaluate Cylinder with the highest MIS_CTR_A[x] value

Set concerned cylinder bit to 1 in SYM_CYL_MIS_A carrier

Set RDN bit *(random)* to 0 in SYM_CYL_MIS_A carrier

Else(5) Set RDN bit *(random)* to 1 in SYM_CYL_MIS_A carrier

Endif(5)

Else(4) Set RDN bit *(random)* to 0 in SYM_CYL_MIS_A carrier

EndIf(4)

EndIf(3)

EndIf(2)

Else(1)

// Be carefull, the 3 following datas are reseted only at the end of the window and not during.

SYM_CYL_MIS_A = 0

LV_DET_MIS_A = 0

LV_MIS_STATE_A = 0

EndIf(1)

// MIS_A default management in case of cylinder shut-off

If INH_IV_MIS ≠ 0 **And** LC_CONF_MIS_A_LIH = 1

Then LV_DET_MIS_A = 1

LV_MIS_STATE_A = 1

Set in SYM_CYL_MIS_A the cylinder bit(s) who are set in INH_IV_MIS *(OR mask)*

Else no action

EndIf

// MIS_A multiple cylinder identification

If LC_CONF_MPL_DET_MIS_A = 1

And Number of cylinder(s) set to 1 in SYM_CYL_MIS_A is equal or over C_NR_CYL_MPL_MIS_A

Then Set MPL bit *(multiple)* to 1 in SYM_CYL_MIS_A carrier

Else Set MPL bit *(multiple)* to 0 in SYM_CYL_MIS_A carrier

End

B.5.4.6 CARB A end detection process

If LV_FTP_MIS_A = 0

Or CTR_DET_MIS_A >= C_NR_DET_MAX_MIS_A

Or CTR_CHK_MIS_A >= C_NR_DET_MAX_MIS_A

Then LV_END_MIS_A = 1

EndIf

LV_END_WIN_MIS_A = 1 *// Set for one segment*

B.5.4.7 CARB A window reset process

MIS_NR_TDC_A = 0

MIS_SUM_A = 0

CTR_FTP_CDN_MIS_A = 0

For x = 0 : NC_CYL_NR-1

MIS_CTR_A[x] = 0 *// All cylinder specific counters*

EndFor

For i = 0 : NC_CBK_EX_NR-1

MIS_SUM_A_CBK[i] = 0 *// All exhaust cylinder bank specific counters*

EndFor

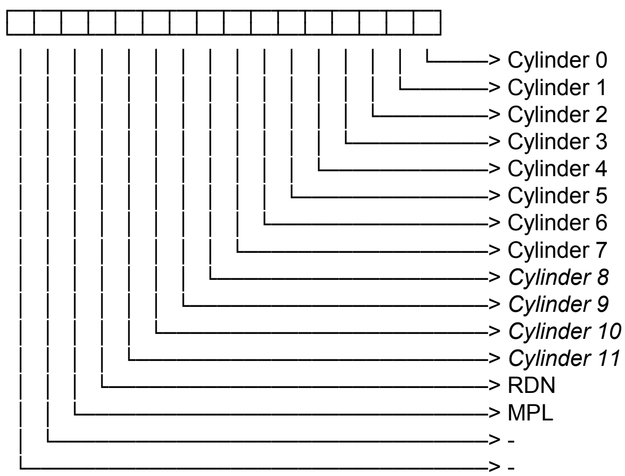
<u>Reset of Misfire Status</u>*:*

The Misfire Status CARB A is resetted at the end of the driving cycle → see formula section

B.5.4.8 Misfire Cylinder Identification SYM_CYL_MIS_A

The location of detected cylinders with misfire status CARB A is coded in a carrier word:

SYM_CYL_MIS_A



B.5.4.9 Misfire Cylinder Limp Home process:

Cylinder shut off is limited by a C_MIS_MAX_NR_OFF_IV maximum number of cylinders that can be shut-off simultaneously

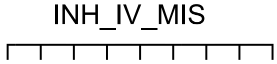
Each time, the misfire cylinder limp home process on one (or more) cylinder(s) is triggered, the corresponding cylinder counter(s) CTR_MIS_A_LIH_CYL[NC_CYL_NR] is(are) unitary incremented.

The location of switched off Cylinder(s) x is set to 1 in INH_IV_MIS, other cylinder bits are set to 0.

<u>On the same driving cycle</u>, if after a cylinder shut-off cancellation, a new cylinder MIS_A criterion occurs during CARB A window, the sytem must shut-off this cylinder once more.

<u>Carrier structure :</u>

INH_IV_MIS



```
└─┴─┴─┴─┴─┴─┴─┴─┘
 │ │ │ │ │ │ │ └──────> Cylinder 0
 │ │ │ │ │ │ └────────> Cylinder 1
 │ │ │ │ │ └──────────> Cylinder 2
 │ │ │ │ └────────────> Cylinder 3
 │ │ │ └──────────────> Cylinder 4
 │ │ └────────────────> Cylinder 5
 │ └──────────────────> Cylinder 6
 └────────────────────> Cylinder 7
```

B.5.4.10 Conditions for Cylinder switch back on

Update rate: *every 100ms, independently of MIS_A window process & conditions*

Cylinder(s) x can be switched on according two different methods:

1. Cylinder(s) x can be switched back on at the end of the driving cycle

   LC_CONF_INH_IV_MIS = 0

2. Cylinder(s) x can be switched back on after a defined number of fuel cut-off phases **and** at the end of the driving cycle

   LC_CONF_INH_IV_MIS = 1

   For this purpose, the system performs a test for the applicable number C_MIS_MIN_NR_PUC of trailing throttle fuel cut-off LV_PUC cycles, since the last cylinder(s) shutoff.

   Therefore, the trailing throttle fuel cut-off LV_PUC flag must have been active for the applicable minimum period C_T_MIS_MIN_PUC.

   If the condition concerning the minimum number of trailing throttle fuel cut-off events is met. Cylinder shut-off is cancelled at the end of the last trailing throttle fuel cut-off phase only for cylinders who answer to the conditions CTR_MIS_A_LIH_CYL[x] < C_NR_MAX_MIS_A_LIH_CYL *(check for all cylinders),* all other cylinders stay in fuel cut-off up to end of the driving cycle.

Cylinder shut-off is also cancelled after engine stopped LV_ES.

B.5.4.11 Process on cylinder shut-off cancellation

Update rate: *every TDC, independently of MIS_A window process & conditions*

If Any 1 -> 0 cylinder bit(s) transition within INH_IV_MIS

And LC_CONF_MIS_A_LIH = 1

Then Corresponding bit(s) within SYM_CYL_MIS_A is set to 0

- **If** LC_CONF_MPL_DET_MIS_A = 1
- **And** Number of remaining cylinder bit(s) set to 1 in SYM_CYL_MIS_A is less than C_NR_CYL_MPL_MIS_A cylinders
- **Then** Set MPL bit *(multiple)* to 0 in SYM_CYL_MIS_A carrier
- **EndIf**
- **If** No more cylinder bit(s) or RND bit are set to 1 whitin SYM_CYL_MIS_A structure
- **Then** LV_DET_MIS_A = 0

LV_MIS_STATE_A = 0

EndIf

Else *No action*

EndIf

If Any cylinder bit(s) within INH_IV_MIS is set to 1

And Corresponding bit(s) within INH_INJ is (are) also set to 1

Then LV_INH_IV_MIS_A = 1

Else LV_INH_IV_MIS_A = 0

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LOAD_MIS_FTP	1	0...7FFFH	0...99.99694	3.0518e-3	[%]
FTP cycle engine load threshold					
C_N_MIS_FTP	1	0...1FE0H	0...8160	1	[rpm]
FTP cycle engine speed threshold					
C_MIS_A_IND_THD_MT	1	0...FFFFH	0...65535	1	[-]
Cylinder individual threshold used for cylinder shut-off during CARB A criterion window with MT					
C_MIS_A_IND_THD_AT	1	0...FFFFH	0...65535	1	[-]
Cylinder individual threshold used for cylinder shut-off during CARB A criterion window with AT					
C_RATIO_FTP_CDN_MIS_AT	1	0...FFH	0...99.60937	0.390625	[%]
Covering ratio to detect FTP conditions for long term protection with AT vehicle					
C_RATIO_FTP_CDN_MIS_MT	1	0...FFH	0...99.60937	0.390625	[%]
Covering ratio to detect FTP conditions for long term protection with MT vehicle					
C_RATIO_FTP_CDN_IND_MIS_AT	1	0...FFH	0...99.60937	0.390625	[%]
Covering ratio to detect FTP conditions for short term protection with AT vehicle					
C_RATIO_FTP_CDN_IND_MIS_MT	1	0...FFH	0...99.60937	0.390625	[%]
Covering ratio to detect FTP conditions for short term protection with MT vehicle					
C_MIS_A_MIN_NR	1	0...FFH	0...99.60937	0.390625	[%]
Cylinder identification ratio for CARB A criterion detection					
C_MIS_SUM_A_THD_MT	1	0...FFFFH	0...65535	1	[-]
Cylinder global threshold for misfire status CARB A detection with MT					
C_MIS_SUM_A_CBK_THD_MT	1	0...FFFFH	0...65535	1	[-]
Exhaust cylinder bank threshold for misfire status CARB A detection with MT					
C_MIS_SUM_A_THD_AT	1	0...FFFFH	0...65535	1	[-]
Cylinder global threshold for misfire status CARB A detection with AT					
C_MIS_SUM_A_CBK_THD_AT	1	0...FFFFH	0...65535	1	[-]
Exhaust cylinder bank threshold for misfire status CARB A detection with AT					
C_MIS_MAX_NR_OFF_IV	1	0...4H	0...4	1	[-]
Maximum number of allowed cylinder to be shut-off after detection of the misfire status CARB A.					
C_NR_DET_MAX_MIS_A	1	0...FFH	0...255	1	[-]
Maximum value to confirm misfire CARB A status detection (typical: 3) for long term protection					
C_MIS_MIN_NR_PUC	1	1...FFH	1...255	1	[-]
Minimum number of trailing throttle fuel cut-off phases LV_PUC before cancelling cylinder(s) shut-off.					
C_T_MIS_MIN_PUC	1	1...3E8H	0.1...100	0.1	[s]
Minimum duration of trailing throttle fuel cut-off phase to be validated like this.					
C_NR_CYL_MPL_MIS_A	1	0...FFH	0...255	1	[-]
Number of cylinder in MIS_A to set the multiple misfire CARB A status (typical: 2)					
C_NR_MAX_MIS_A_LIH_CYL	1	0...FFH	0...255	1	[-]
Maximum number of MIS_A cylinder limp-home cycle to disable fuel reactivation during current DC					
LC_CONF_INH_IV_MIS	1	0...1H	0...1	1	[-]
Cylinder switch back on configuration mode : switch back on at the end of driving cycle (=0) or after a defined number of fuel cut-off phases (=1)					
LC_CONF_RND_DET_MIS_A	1	0...1H	0...1	1	[-]
Configuration for random cylinder detection when no specific cylinder identified, (=0) Cyl. with the highest value is identified in SYM_CYL_MIS_A or (=1) No identification is realised and RDN bit is set to 1 in SYM_CYL_MIS_A					
LC_CONF_MPL_DET_MIS_A	1	0...1H	0...1	1	[-]
Configuration for multiple cylinder status set in SYM_CYL_MIS_A (multiple misfire pattern Pcode saved) (=1)					
LC_CONF_MIS_A_LIH	1	0...1H	0...1	1	[-]
Configuration to select MIL mode in case of cylinder shut-off (=0, LV_MIS_STATE_A=0, MIL is not set by cylinder shut-off function), (=1, LV_MIS_STATE_A=1, MIL is ON until end of cylinder shut-off, see mode chosen with LC_CONF_INH_IV_MIS)					
LC_ENA_MIS_A_IND_LIH	1	0...1H	0...1	1	[-]
Enable the misfire cylinder limp home process during the CARB A window (short term catalyst protection) (=1, active)					
LC_ENA_MIS_A_LIH	1	0...1H	0...1	1	[-]
Enable the misfire cylinder limp home process at the end of the CARB A window (long term catalyst protection) (=1, active)					

ID_NR_DET_ENA_MIS_A_LIH_MT	8	0...FFH	0...255	1	[-]
LDPM_N_32_1_MISF	8	0...FFH	0...8160	32	[rpm]
Number of CARB-A detections to enable MIS_A cylinder switch-off limp-home for MT					
ID_NR_DET_ENA_MIS_A_LIH_AT	8	0...FFH	0...255	1	[-]
LDPM_N_32_1_MISF	8	0...FFH	0...8160	32	[rpm]
Number of CARB-A detections to enable MIS_A cylinder switch-off limp-home for AT					
IP_MIS_A_FAC_AFS	6*6	0...64H	0...100	1	[-]
LDP_N__MIS_A_FAC_AFS	6	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_MIS__MIS_A_FAC_AFS	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Catalyst damage weighting factor used for MIS_A counters increment (AFS combustion mode)					
#IF NC_CMB_CONF = AFS_AFL Or AFS_AFL_S					
IP_MIS_A_FAC_HOM_S	6*6	0...64H	0...100	1	[-]
LDP_N__MIS_A_FAC_HOM_S	6	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_MIS__MIS_A_FAC_HOM_S	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Catalyst damage weighting factor used for MIS_A counters increment (HOM_S combustion mode)					
#ENDIF					
#IF NC_CMB_CONF = AFS_S Or AFS_AFL_S					
IP_MIS_A_FAC_S	6*6	0...64H	0...100	1	[-]
LDP_N__MIS_A_FAC_S	6	0...1FE0H	0...8160	1	[rpm]
LDP_LOAD_MIS__MIS_A_FAC_S	6	0...7FFFH	0...99.99694	3.0518e-3	[%]
Catalyst damage weighting factor used for MIS_A counters increment (Stratified combustion mode)					
#ENDIF					

B.5.5 Determination of CARB B1 & B4 misfire criterions, causing increased emissions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MIS_STATE_B1	V/O	0...1H	0...1	1	[-]
CARB B1 misfire criterion confirmed					
LV_DET_MIS_B4	V	0...1H	0...1	1	[-]
CARB B4 misfire criterion detected in actual 1000 rev. window, (not debounced)					
LV_MIS_STATE_B4	V/O	0...1H	0...1	1	[-]
CARB B4 misfire criterion confirmed (debounced)					
LV_MIS_STATE_B	V/O	0...1H	0...1	1	[-]
CARB B (B1 or B4) misfire criterion confirmed (debounced)					
MIS_CTR_B[NC_CYL_NR]	V	0...FFFFH	0...65535	1	[-]
Misfire sums (cylinder individual) after combination in CARB B window					
MIS_NR_TDC_B	V	0...FFFFH	0...65535	1	[-]
TDC counter CARB B window (1000 crankshaft revolutions)					
MIS_SUM_B	V	0...FFFFH	0...65535	1	[-]
CARB B misfire sum (global)					
MIS_SUM_B_CBK[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[-]
CARB B misfire sum (exhaut cylinder bank dedicated)					
CTR_CHK_MIS_B4	V	0...FFH	0...255	1	[-]
CARB B4 window occurrence counter					
CTR_DET_MIS_B4	V	0...FFH	0...255	1	[-]
CARB B4 misfire criterion detected counter					
LV_END_MIS_B1	V/O	0...1H	0...1	1	[-]
CARB B1misfire criterion determination end					
LV_END_MIS_B4	V/O	0...1H	0...1	1	[-]
CARB B4 misfire criterion determination end					
LV_END_WIN_MIS_B1	V/O	0...1H	0...1	1	[-]
End of CARB B1 window for similar condition					
LV_ERR_IN_WIN_MIS_B1	V	0...1H	0...1	1	[-]
Error during CARB B1 window for similar condition					
LV_ERR_IN_WIN_MIS_B4	V	0...1H	0...1	1	[-]
Error during CARB B4 window for similar condition					
LV_ERR_MIS_A_IN_WIN_B	V/O	0...1H	0...1	1	[-]
Error MIS_A during CARB B1/B4 window for similar condition					
LV_END_WIN_MIS_B4	V/O	0...1H	0...1	1	[-]
End of CARB B4 window for similar condition					
SYM_CYL_MIS_B1	V/O	0...FFFFH	0...65535	1	[-]
CARB B1 misfire criterion symptoms					
SYM_CYL_MIS_B4	V/O	0...FFFFH	0...65535	1	[-]
CARB B4 misfire criterion symptoms					

Input data:

LV_DET_CFM_MIS	LV_DC	LV_AT	LV_CDN_MIS_B1
LV_CDN_MIS_B4	SEG_NR_ER	SEG_NR_CBK_ER	LC_MIS_INH
NC_CBK_EX_NR	NC_CYL_NR	LV_MIS_B_DIAG_REQ_APP	FAC_MIS_SUM_B1_THD_APP
FAC_MIS_SUM_B4_THD_APP	NLC_ENA_SCDN_NEW	LV_ERR_IN_WIN_MIS_A	

Description:

The purpose is to detect a misfire rate causing an emission increase. The failure criterion entry is performed by direct statistical evaluation.

Detection:

The check for misfire rate violation CARB comprises 1.000 crankshaft revolutions (500*NC_CYL_NR tdc's).

Each combustion during these 1.000 crankshaft revolutions is monitored for misfiring.

If a confirmed misfiring is detected on ER current TDC (LV_DET_CFM_MIS=1), the cylinder-specific misfire counter MIS_CTR_B[SEG_NR_ER] is incremented by one.

In every driving cycle a distinction is made between misfires detected during the first 1000 crankshaft revs after engine start (**CARB B1**) and misfires detected in later observing intervals. In the second case, a misfire rate violation must take place during four monitoring intervals to recognise misfire (**CARB B4**).

Statistic:

At the end of the first 1.000 crankshaft revolutions, the added cylinder-specific sums are evaluated. If the sum exceeds the applicable value C_THD_MIS_B1_AT/MT, **CARB B1** misfire criterion is detected.

At the end of each following 1.000 crankshaft revolutions interval, the added cylinder-specific sums are evaluated. If the sum exceeds the applicable value C_THD_MIS_B1_AT/MT the anti-bounce counter CTR_DET_MIS_B4 will be incremented till its maximum C_NR_DET_MAX_MIS_B4 (typical = 3) is reached. If the sum does not exceed C_THD_MIS_B1_AT/MT, CTR_DET_MIS_B4 remains unchanged.

To filter out those cylinders with low misfire rates, the cylinder-specific counters MIS_CTR_B[NC_CYL_NR] shall be compared to the applicable value C_THD_MIS_B_MIN before storage takes place.

If none of the cylinder-specific counters is higher than C_THD_MIS_B_MIN, a maximum choice is done and the cylinder with the highest misfire rate is entered into failure memory.

Application conditions:

Initialisation: *at ECU reset, at LV_DC 0 -> 1 transition, at LC_MIS_INH 0 to 1 transition & at MIS_B1/B4 or MIS_[NC_CYL_NR] errors clearing*

LV_MIS_STATE_B1	= 0
LV_DET_MIS_B4	= 0
LV_MIS_STATE_B4	= 0
LV_MIS_STATE_B	= 0
SYM_CYL_MIS_B1	= 0
SYM_CYL_MIS_B4	= 0
MIS_NR_TDC_B	= 0
MIS_SUM_B	= 0
CTR_DET_MIS_B4	= 0
CTR_CHK_MIS_B4	= 0
LV_END_MIS_B1	= 0
LV_END_MIS_B4	= 0
LV_END_WIN_MIS_B1	= 0
LV_END_WIN_MIS_B4	= 0

LV_ERR_IN_WIN_MIS_B1 = 0

LV_ERR_IN_WIN_MIS_B4 = 0

LV_ERR_MIS_A_IN_WIN_B = 0

For x = 0 : NC_CYL_NR-1

MIS_CTR_B[x] = 0 *// All cylinder specific counters*

EndFor

For i = 0 : NC_CBK_EX_NR-1

MIS_SUM_B_CBK[i] = 0 *// All Exhaut cylinder bank specific counters*

EndFor

Update rate: every segment

Activation:

LV_DC = 1

And LC_MIS_INH = 0

Deactivation:

LV_DC = 0

Or LC_MIS_INH = 1

Formula section:

B.5.5.1 CARB B1/B4 windows management *(main process)*

If(1) LV_CDN_MIS_B1 = 1 *// CARB B1 monitoring active (see App Inc file)*

Or LV_CDN_MIS_B4 = 1 *// CARB B4 monitoring active (see App Inc file)*

Then(1)

If(2) LV_DET_CFM_MIS = 1 *// Misfire detected*

Then(2) call **'CARB B1/B4 Misfire counters increment process'**

(see description below)

Else(2) *MIS_B1/B4 counters unchanged*

EndIf(2)

If(2) LV_ERR_IN_WIN_MIS_A = 1

Then(2) LV_ERR_MIS_A_IN_WIN_B = 1

EndIf(2)

If(2) MIS_NR_TDC_B < 500 * NC_CYL_NR

Then(2) MIS_NR_TDC_B= MIS_NR_TDC_B + 1

LV_END_WIN_MIS_B1 = 0

LV_END_WIN_MIS_B4 = 0

If(3) NLC_ENA_SCDN_NEW = 1

'Monitoring process during CARB B1/B4 window'

EndIf(3)

(End of windows not reached)

Else(2) call **'Monitoring process at CARB B1/B4 window end'**

(see description below)

call **'CARB B1/B4 ends detection process'**

(see description below)

call **'CARB B1/B4 window reset process'** *(see description below)*

EndIf(2)

If(1) **[** LV_MIS_STATE_B1 = 1

And (LV_END_MIS_B4 = 0 **Or** LC_CONF_MIS_STATE_B = 1) **]**

Or LV_MIS_STATE_B4 = 1

Then(1) LV_MIS_STATE_B = 1

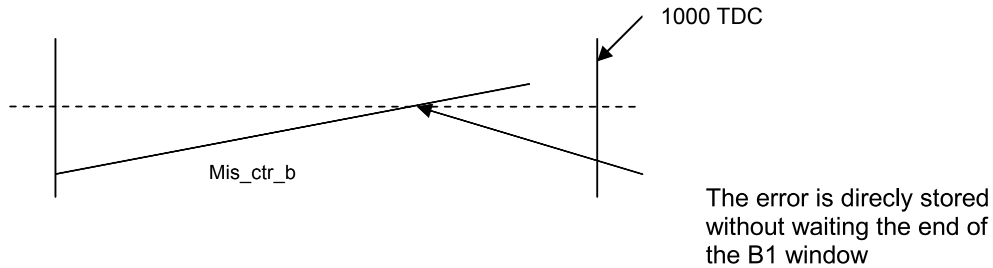
Else(1) LV_MIS_STATE_B = 0

EndIf(1)

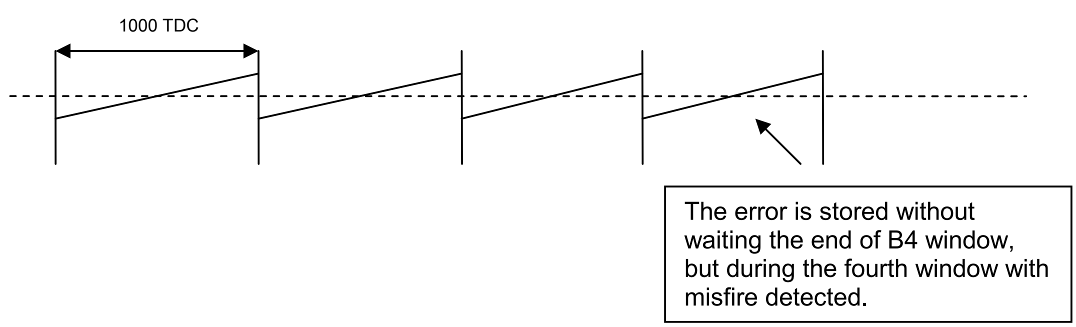
Else(1) *no process (Counters & flags unchanged)*

EndIf(1)

MIS_B1 Management



MIS_B4 Management



B.5.5.2 CARB B1/B4 Misfire counters increment process

MIS_SUM_B = MIS_SUM_B + 1

MIS_SUM_B_CBK[SEG_NR_CBK_ER] = MIS_SUM_B_CBK[SEG_NR_CBK_ER] + 1

// Assign only ER current specific exhaut cylinder bank counter

MIS_CTR_B[SEG_NR_ER] = MIS_CTR_B[SEG_NR_ER] + 1

// Assign only ER current specific cylinder counter

B.5.5.3 Monitoring process at CARB B1/B4 window end

If(1) LV_END_MIS_B1 = 0 *// End of CARB B1 window*

Then(1)

If(2) **{** LV_MIS_B_DIAG_REQ_APP = 0 **And**

[(MIS_SUM_B >= C_MIS_SUM_B1_THD_MT **And** LV_AT = 0 **)**

Or **(** MIS_SUM_B >= C_MIS_SUM_B1_THD_AT **And** LV_AT = 1 **)**

Or MIS_SUM_B_CBK[0...NC_CBK_EX_NR-1]

>= C_MIS_SUM_B1_CBK_THD **] }**

// Check for each exhaust cylinder banks

Or

{ LV_MIS_B_DIAG_REQ_APP = 1 **And**

[(MIS_SUM_B >=

(FAC_MIS_SUM_B1_THD_APP * C_MIS_SUM_B1_THD_MT)

And LV_AT = 0 **)**

Or **(** MIS_SUM_B >=

(FAC_MIS_SUM_B1_THD_APP * C_MIS_SUM_B1_THD_AT)

And LV_AT = 1 **)] }**

Then(2) call **'CARB B1 criterion determination process'** *(see description below)*

Else(2) *no criterion detection*

LV_MIS_STATE_B1 = 0

EndIf(2)

Else(1)

// Be carefull, the counter of window B4 (CTR_CHK_MIS_B4) is incremented only at the end of the window and not during.

CTR_CHK_MIS_B4 = CTR_CHK_MIS_B4 + 1 *// at the End of CARB B4 window*

If(2) **{** LV_MIS_B_DIAG_REQ_APP = 0 **And**

[(MIS_SUM_B >= C_MIS_SUM_B4_THD_MT **And** LV_AT = 0 **)**

Or **(** MIS_SUM_B >= C_MIS_SUM_B4_THD_AT **And** LV_AT = 1 **)**

Or MIS_SUM_B_CBK[0...NC_CBK_EX_NR-1]

>= C_MIS_SUM_B4_CBK_THD **] }**

// Check for each exhaust cylinder banks

Or

{ LV_MIS_B_DIAG_REQ_APP = 1 **And**

[(MIS_SUM_B >=

(FAC_MIS_SUM_B4_THD_APP * C_MIS_SUM_B4_THD_MT)

And LV_AT = 0 **)**

Or (MIS_SUM_B >=

(FAC_MIS_SUM_B4_THD_APP * C_MIS_SUM_B4_THD_AT)

And LV_AT = 1 **)] }**

Then(2) call **'CARB B4 criterion determination process'** *(see description below)*

Else(2) *no criterion detection*

// Be carefull, the 3 following datas are reseted only at the end of the window and not during.

LV_DET_MIS_B4 = 0

LV_MIS_STATE_B4 = 0

SYM_CYL_MIS_B4 = 0

EndIf(2)

EndIf(1)

B.5.5.4 Monitoring process during CARB B1/B4 window

This part is realized during CARB B1/B4 window when NLC_ENA_SCDN_NEW=1.

In this case, we do exactly the same operations that those done in "**'Monitoring process at CARB B1/B4 window end'.**

B.5.5.5 CARB B1 criterion determination process:

// Global CARB B1 default

As soon as an error occurs during B1 window, we set the error.

But if the error enters again, we do not want to store it again. This is realized by lv_err_in_win_mis_b1 flag.

if(1) (LV_ERR_IN_WIN_MIS_B1 = 0)

Then(1)

LV_MIS_STATE_B1 = 1

LV_ERR_IN_WIN_MIS_B1 = 1

// CARB B1 cylinder identification

For x = 0 to NC_CYL_NR - 1 *// Check for each cylinder x*

If(2) MIS_CTR_B[x] > C_MIS_B1_MIN_NR * MIS_SUM_B

Then(2) Set in SYM_CYL_MIS_B1 the cylinder x concerned bit to 1

EndIf(2)

EndFor

// CARB B1 random cylinder pattern

If No specific cylinder counters comply with C_MIS_B1_MIN_NR * MIS_SUM_B criterion

Then

If(3) LC_CONF_RND_DET_MIS_B = 0

Then(3) Evaluate Cylinder with the highest MIS_CTR_B[x] value

Set concerned cylinder bit to 1 in SYM_CYL_MIS_B1 carrier

Set RDN bit *(random)* to 0 in SYM_CYL_MIS_B1 carrier

Else(3) Set RDN bit *(random)* to 1 in SYM_CYL_MIS_B1 carrier

Endif(3)

Else Set RDN bit *(random)* to 0 in SYM_CYL_MIS_B1 carrier

EndIf

// CARB B1 multiple cylinder identification

If LC_CONF_MPL_DET_MIS_B = 1

And number of cylinder(s) set to 1 in SYM_CYL_MIS_B1 is equal or over C_NR_CYL_MPL_MIS_B

Then Set MPL bit *(multiple)* to 1 in SYM_CYL_MIS_B1 carrier

Else Set MPL bit *(multiple)* to 0 in SYM_CYL_MIS_B1 carrier

End

Endif(1)

B.5.5.6 CARB B4 criterion determination process

If(1) (LV_ERR_IN_WIN_MIS_B4 = 0)

Then(1)

CTR_DET_MIS_B4 = CTR_DET_MIS_B4 + 1

LV_DET_MIS_B4 = 1

LV_ERR_IN_WIN_MIS_B4 = 1

As soon as the misfire criterion detected counter reached his threshold, we directly set the error. To avoid to stored it in the same window a second time, we used *lv_err_in_win_mis_b4 flag.*

If(2) (CTR_DET_MIS_B4 >= C_NR_DET_MAX_MIS_B4)

Then(2)

// Global CARB B4 default

LV_MIS_STATE_B4 = 1

// CARB B4 cylinder identification

For(3) x = 0 to NC_CYL_NR-1 *// Check for each cylinder x*

If(4) MIS_CTR_B[x] > C_MIS_B4_MIN_NR * MIS_SUM_B

Then(4) Set in SYM_CYL_MIS_B4 the cylinder x concerned bit to 1

EndIf(4)

EndFor(3)

// CARB B4 random cylinder pattern

If(3) No specific cylinder counters comply with C_MIS_B4_MIN_NR * MIS_SUM_B criterion

Then (3)

If(4) LC_CONF_RND_DET_MIS_B = 0

Then(4) Evaluate Cylinder with the highest MIS_CTR_B[x] value

Set concerned cylinder bit to 1 in SYM_CYL_MIS_B4 carrier

Set RDN bit *(random)* to 0 in SYM_CYL_MIS_B4 carrier

Else(4) Set RDN bit *(random)* to 1 in SYM_CYL_MIS_B4 carrier

Endif(4)

Else(3) Set RDN bit *(random)* to 0 in SYM_CYL_MIS_B4 carrier

EndIf(3)

// CARB B4 multiple cylinder identification

If(3) LC_CONF_MPL_DET_MIS_B = 1

And number of cylinder(s) set to 1 in SYM_CYL_MIS_B4 is equal or over C_NR_CYL_MPL_MIS_B

Then(3) Set MPL bit *(multiple)* to 1 in SYM_CYL_MIS_B4 carrier

Else(3) Set MPL bit *(multiple)* to 0 in SYM_CYL_MIS_B4 carrier

End(3)

Else(2)

SYM_CYL_MIS_B4 = 0

LV_MIS_STATE_B4 = 0

EndIf(2)

EndIf(1)

B.5.5.7 CARB B1/B4 ends detection process

If(1) LV_END_MIS_B1 = 0

Then(1) LV_END_MIS_B1 = 1

LV_END_WIN_MIS_B1 = 1 *(set for one segment)*

Else(1)

LV_END_WIN_MIS_B4 = 1 *(set for one segment)*

If(2) CTR_DET_MIS_B4 >= C_NR_DET_MAX_MIS_B4

Or CTR_CHK_MIS_B4 >= C_NR_DET_MAX_MIS_B4

Then(2) LV_END_MIS_B4 = 1

EndIf(2)

EndIf(1)

B.5.5.8 CARB B1/B4 window reset process

MIS_NR_TDC_B = 0

MIS_SUM_B = 0

LV_ERR_IN_WIN_MIS_B1 = 0

LV_ERR_IN_WIN_MIS_B4 = 0

~~LV_ERR_IN_WIN_MIS_A = 0~~

LV_ERR_MIS_A_IN_WIN_B = 0

For x = 0 : NC_CYL_NR-1

MIS_CTR_B[x] = 0 *// All cylinder specific counters*

EndFor

For i = 0 : NC_CBK_EX_NR-1

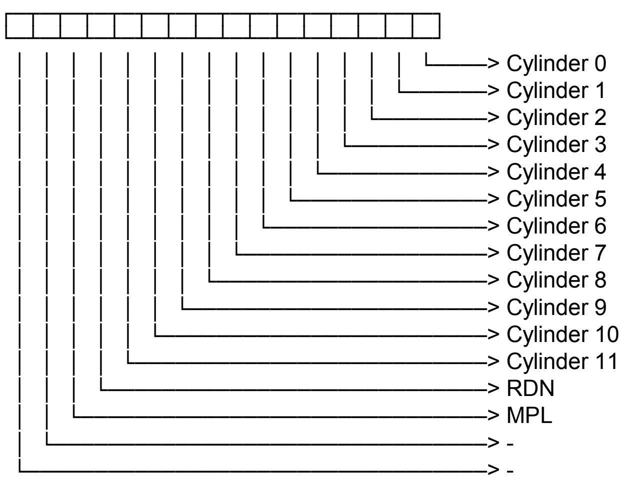
MIS_SUM_B_CBK[i] = 0 *// All exhaust cylinder bank specific counters*

EndFor

B.5.5.9 SYM_CYL_MIS_B1 and SYM_CYL_MIS_B4 carriers definition

The location of detected cylinders with misfire status CARB B is coded in two carrier bytes:

SYM_CYL_MIS_B1, SYM_CYL_MIS_B4



Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_MIS_SUM_B1_THD_MT	1	0...FFFFH	0...65535	1	[-]
Global threshold for CARB B1 misfire criterion detection, LV_AT = 0					
C_MIS_SUM_B1_THD_AT	1	0...FFFFH	0...65535	1	[-]
Global threshold for CARB B1 misfire criterion detection, LV_AT = 1					
C_MIS_SUM_B4_THD_MT	1	0...FFFFH	0...65535	1	[-]
Global threshold for CARB B4 misfire criterion detection, LV_AT = 0					
C_MIS_SUM_B4_THD_AT	1	0...FFFFH	0...65535	1	[-]
Global threshold for CARB B4 misfire criterion detection, LV_AT = 1					
C_MIS_B1_MIN_NR	1	0...FFH	0...99.60937	0.390625	[%]
Cylinder identification ratio for CARB B1 criterion detection					
C_MIS_B4_MIN_NR	1	0...FFH	0...99.60937	0.390625	[%]
Cylinder identification ratio for CARB B4 criterion detection					
C_MIS_SUM_B1_CBK_THD	1	0...FFFFH	0...65535	1	[-]
Exhaut cylinder bank threshold for CARB B1 misfire criterion detection					
C_MIS_SUM_B4_CBK_THD	1	0...FFFFH	0...65535	1	[-]
Exhaut cylinder bank threshold for CARB B4 misfire criterion detection					
C_NR_CYL_MPL_MIS_B	1	0...FFH	0...255	1	[-]
Number of cylinder in MIS_Bx to set the multiple misfire CARB Bx status (typical: 2)					
C_NR_DET_MAX_MIS_B4	1	0...FFH	0...255	1	[-]
Maximum value of CTR_DET_MIS_B4 for misfire CARB B4 status detection (typical: 4)					
LC_CONF_MIS_STATE_B	1	0...1H	0...1	1	[-]
Configuration for LV_MIS_STATE_B setting (=0, setting by MIS_B1 only when end of MIS_B4 isn't reached, then only set by MIS_B4) (=1, setting by MIS_B1 or MIS_B4 in any case)					
LC_CONF_MPL_DET_MIS_B	1	0...1H	0...1	1	[-]
Configuration for multiple cylinder status set in SYM_CYL_MIS_Bx (multiple misfire pattern Pcode saved) (=1)					
LC_CONF_RND_DET_MIS_B	1	0...1H	0...1	1	[-]
Configuration for random cylinder detection when no specific cylinder identified,: (=0) Cyl. with the highest value is identified in SYM_CYL_MIS_B1 or SYM_CYL_MIS_B4 or (=1) No identification is realised and RDN bit is set to 1 in SYM_CYL_MIS_B1/B4					

B.6 Misfire Rate Determination and Error Management - Application Incidences

B.6.1 Application Incidences data for diagnosis

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_MIS_A_DIAG_REQ_APP	O/V	0...1H	0...1	1	-
Specific request to evaluate CARB A criterion with project specific calibration set according external conditions (bad fuel quality...)					
LV_MIS_B_DIAG_REQ_APP	O/V	0...1H	0...1	1	-
Specific request to evaluate CARB B1 & B4 criterions with project specific calibration set according external conditions (bad fuel quality...)					
FAC_MIS_A_APP	O/V	0..64H	0...100	1	-
Catalyst damage weighting factor used to increment counters for MIS_A criterion pending application specific conditions					
FAC_MIS_A_THD_IND_APP	O/V	0...1FFH	0...1.9961	0.0039	-
Cylinder individual threshold used for cylinder shut-off during CARB A criterion window pending application specific conditions					
FAC_MIS_SUM_A_THD_APP	O/V	0...1FFH	0...1.9961	0.0039	-
Global threshold used for CARB A criterion pending application specific conditions					
FAC_MIS_SUM_B1_THD_APP	O/V	0...1FFH	0...1.9961	0.0039	-
Global threshold used for CARB B1 criterion pending application specific conditions					
FAC_MIS_SUM_B4_THD_APP	O/V	0...1FFH	0...1.9961	0.0039	-
Global threshold used for CARB B4 criterion pending application specific conditions					
LV_INH_IGC_DIAG_MIS	O/V	0..1H	0.... 1	1	-
Inhibition of misfire crossed diagnosis with IGC OBD I errors					
LV_INH_IV_DIAG_MIS	O/V	0..1H	0.... 1	1	-
Inhibition of misfire crossed diagnosis with IV OBD I errors					
LV_REQ_APP_CTR_MIS_A	O/V	0..1H	0.... 1	1	-
Specific request to allow out of FTP counting mode even if we are within FTP area					

Input data:

TEG_CAT_UP_MDL_MAX	LC_MIS_INH	LV_DC	NC_CYL_NR
LV_INH_DET_MIS	LV_CDN_DIAG_IGC_OL[NC_CYL_NR]	LV_CDN_DIAG_IGC_SCG[NC_CYL_NR]	LV_CDN_DIAG_IGC_SCP[NC_CYL_NR]
LV_CDN_DIAG_IV[NC_CYL_NR]			

FUNCTION DESCRIPTION:

Application conditions:

Initialisation : At Reset, all output data are set to 0

Activation: LV_DC = 1

And LC_MIS_INH = 0

Recurrence: ENRD segment synchronous,
After misfire detection, Before misfire diagnosis

Formula section:

B.6.1.1 Inputs for project specific diagnosis functions:

If TEG_CAT_UP_MDL_MAX > C_TEG_CAT_UP_MIS_A_MAX

Then LV_REQ_APP_CTR_MIS_A = 1

Else LV_REQ_APP_CTR_MIS_A = 0

EndIf

Inhibition of misfire crossed detection with IV and IGC non debounced errors:

If LV_INH_DET_MIS = 1
Or LV_CDN_DIAG_IGC_SCP[NC_CYL_NR] = 0
Or LV_CDN_DIAG_IGC_SCG[NC_CYL_NR] = 0
Or LV_CDN_DIAG_IGC_OL[NC_CYL_NR] = 0
Then LV_INH_IGC_DIAG_MIS = 1
Else LV_INH_IGC_DIAG_MIS = 0
EndIf

If LV_INH_DET_MIS = 1
Or LV_CDN_DIAG_IV [NC_CYL_NR] = 0
Then LV_INH_IV_DIAG_MIS = 1
Else LV_INH_IV_DIAG_MIS = 0
EndIf

B.6.1.2

B.6.1.3 Request to evaluate CARB A criterion with project specific calibration set

LV_MIS_A_DIAG_REQ_APP = 0

FAC_MIS_A_APP = 0

FAC_MIS_A_THD_IND_APP = 1

FAC_MIS_SUM_A_THD_APP = 1

B.6.1.4 Request to evaluate CARB B1 & B4 criterions with project specific calibration set

LV_MIS_B_DIAG_REQ_APP = 0

FAC_MIS_SUM_B1_THD_APP = 1

FAC_MIS_SUM_B4_THD_APP = 1

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TEG_CAT_UP_MIS_A_MAX	1	0...7FF0H	0...2.047E3	0.0625	°C
TEG Threshold for Misfire A counter management					

B.6.2 Determination of CARB misfire criterion monitoring conditions *(ER algorithm integration, NC_MISF_VERS = 1)*

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_MIS_A	O/V	0...1H	0...1	1	-
CARB A misfire criterion monitoring condition					
LV_CDN_MIS_B1	O/V	0...1H	0...1	1	-
CARB B1 misfire criterion monitoring condition					
LV_CDN_MIS_B4	O/V	0...1H	0...1	1	-
CARB B4 misfire criterion monitoring condition					

Input data:

LV_END_MIS_B1	LV_INH_DET_MIS	LV_STATE_CRK_OSC	LC_MIS_INH
NC_MISF_VERS	LV_DC		

General information:

Depending on customers & legal requirements reached, two Tdc's counting modes are available :

- counting each TDC's even if some aren't diagnosed *(LC_CTR_CONF_DIAG_MIS = 0)*
- counting only diagnosed Tdc's *(LC_CTR_CONF_DIAG_MIS = 1)*

Integration:

#If NC_MISF_VERS = 1

Module Integrated

#Else

Module Not Integrated

#EndIf

Application conditions:

Recurrence: ENRD segment synchronous,
After misfire detection, Before misfire diagnosis

Activation: LV_DC = 1
And LC_MIS_INH = 0

Deactivation: LV_DC = 0
Or LC_MIS_INH = 1

Initialisation: *on ECU reset, LV_DC 0 to 1 transition and on LC_MIS_INH 0 to 1 transition*

LV_CDN_MIS_A = 0

LV_CDN_MIS_B1 = 0

LV_CDN_MIS_B4 = 0

Formula section:

```
If(1)  LC_CTR_CONF_DIAG_MIS = 0                          // Counting each TDC's
       Or
       { ( LC_CTR_CONF_DIAG_MIS = 1 And LV_INH_DET_MIS = 0 )
       And
       [ LC_CTR_CONF_OSC_MIS = 1
           Or ( LC_CTR_CONF_OSC_MIS = 0 And LV_STATE_CRK_OSC = 0) ] }
           // if Crankshaft oscillation detection module integrated, to be deleted if not
Then(1)
       LV_CDN_MIS_A = 1                                  // Carb A diagnosis active
       If(2)  LV_END_MIS_B1 = 0
       Then(2)
              LV_CDN_MIS_B1 = 1                          // Carb B1 diagnosis active
              LV_CDN_MIS_B4 = 0                          // Carb B4 diagnosis inactive
       Else(2)
              LV_CDN_MIS_B1 = 0                          // Carb B1 diagnosis inactive
              LV_CDN_MIS_B4 = 1                          // Carb B4 diagnosis active
       EndIf(2)
Else(1)
       LV_CDN_MIS_A = 0          // Carb A diagnosis inactive
       LV_CDN_MIS_B1 = 0         // Carb B1 diagnosis inactive
       LV_CDN_MIS_B4 = 0         // Carb B4 diagnosis inactive
EndIf(1)
```

LV_DC	LC_CTR_CONF_DIAG_MIS	LV_INH_DET_MIS	LC_CTR_CONF_OSC_MIS	LV_STATE_CRK_OSC	LV_CDN_MIS_A LV_CDN_MIS_B1 LV_CDN_MIS_B4	MIS_A, MIS_B1 & MIS_B4 Monitoring Modes
0	x	x	x	x	0	Out of Driving Cycle
1	0	x	x	x	1	All TDC's Counting Mode
1	1	1	x	x	0	Diagnosed TDC's Counting Mode / Classical Fade Out
1	1	0	0	0	1	Diagnosed TDC's Counting Mode / No Fade Out
1	1	0	0	1	0	Classical Fade Out **with** CRK_OSC / Diagnosed TDC's Counting Mode
1	1	0	1	0	1	Diagnosed TDC's Counting Mode / No Fade Out
1	1	0	1	1	1	Classical Fade Out **without** CRK_OSC / Diagnosed TDC's Counting Mode

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_CTR_CONF_DIAG_MIS	1	0...1H	0...1	1	-
Diagnosis TDC counting mode : counting all TDCs (=0) or only diagnosed TDC's (=1)					
LC_CTR_CONF_OSC_MIS	1	0...1H	0...1	1	-
Diagnosis TDC counting mode when crankshaft oscillations occurs: stop counting TDC's (=0) or counting allowed TDC's (=1)					

B.6.3 Specific cylinder misfire errors

B.6.4 Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_MIS[NC_CYL_NR]	V/O	0...1H	0...1	1	[-]
Diagnosis condition specific cylinder misfire					
LV_ERR_MIS[NC_CYL_NR]	V/O	0...1H	0...1	1	[-]
Present error flag specific cylinder misfire					
ERR_SYM_MIS[NC_CYL_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	0	[-]
Error symptom specific cylinder misfire					
LV_END_DIAG_MIS[NC_CYL_NR]	V/O	0...1H	0...1	1	[-]
End of diagnosis flag					

Import actions:

ACTION_ERRM_CdnDiagScdn (IN <XX>)
This action is used to calculate a ratio to recognise the similar condition with or without failure
ACTION_ERRM_EndWinScdn (IN <XX>, IN <XX>)
This is used for recognition of similar condition with or without failure

Input data:

SYM_CYL_MIS_A	SYM_CYL_MIS_B1	SYM_CYL_MIS_B4	LV_MIS_STATE_A
LV_MIS_STATE_B1	LV_MIS_STATE_B4	LV_END_WIN_MIS_B1	LV_END_WIN_MIS_B4
LV_CDN_MIS_B4	LV_END_MIS_B4	LV_CDN_MIS_A	LV_CDN_MIS_B1
LC_MIS_INH	NLC_TREAT_DIAG_MIS	LV_DC	LV_ERR_MIS_A_IN_WIN_B
LV_ERR_IN_WIN_MIS_B4	LV_END_WIN_MIS_A	NLC_ENA_SCDN_NEW	

Description:

If Misfire criterion is detected on one cylinder with at least one of the CARB_A, CARB_B1 or CARB4 default criterions, the corresponding cylinder error is set.

Each cylinder supply at least one P-code (error is set directly without debounce).

The symptom is always corresponding to the last error occurence, and the error is reseted only when all misfire criterions on this cylinder have disappeared.

Error-symptoms are defined to this diagnosis function as following :

```
[ | | | ]
 |  |  |  └─> - Cylinder in misfire, Criterion default : MIS_A     (= SYM_0)
 |  |  └────> - Cylinder in misfire, Criterion default : MIS_B1    (= SYM_1)
 |  └───────> - Cylinder in misfire, Criterion default : MIS_B4    (= SYM_2)
 └──────────> -                                                    (= SYM_3)
```

Integration

#If NLC_TREAT_DIAG_MIS = 1

Module integrated

#Else : Module not integrated

#Endif

Formula section:

Recurrence: ENRD segment synchronous,
After misfire detection, After misfire diagnosis

Activation:

```
If(1) LV_DC = 1
      And LC_MIS_INH = 0
      And   ( LV_CDN_MIS_A = 1
              Or    LV_CDN_MIS_B4 = 1
              Or    LV_CDN_MIS_B1 = 1 )
Then(1)     For(2)  x = 0 ... NC_CYL_NR-1
                    LV_CDN_DIAG_MIS[x] = 1        // Diagnosis is active for all cylinders
                    If(3) NLC_ENA_SCDN_NEW=1
                    Then(3) ACTION_ERRM_CdnDiagScdn (MIS[x])
                    EndIf(3)
            EndFor(2)
Else(1)     For(2)  x = 0 ... NC_CYL_NR-1
                    LV_CDN_DIAG_MIS[x] = 0        // Diagnosis is passive for all cylinders
            EndFor(2)
EndIf(1)
```

Initialisation: *at ECU reset, LV_DC 0 to 1 transition and LC_MIS_INH 0 to 1 transition*

For all cylinders

```
For  x = 0 ... NC_CYL_NR-1
      LV_CDN_DIAG_MIS[x]       = 0
      LV_END_DIAG_MIS[x]       = refer to filter configuration for the initialisation value
      LV_ERR_MIS[x]            = 0
      ERR_SYM_MIS[x]           = 0
EndFor
```

Diagnosis:

Note : CARRIER*[x]* = 1, stands for bit dedicated to cylinder x inside CARRIER structure is set to 1

ERR_SYM[x]*[0]* = 1, stands for symptom 0 inside ERR_SYM[x] structure is set to 1

[x] (italic) stands for bit assignement within a data, [x] stands for data assignement within an array

Individual cylinder errors set in case of MIS_A failure

```
If(1)   LV_MIS_STATE_A = 1                               // MIS_A failure criterion
    For(2)  x = 0 to NC_CYL_NR - 1                       // Check for each cylinder x
        If(3)       SYM_CYL_MIS_A[x] = 1
        Then(3)     LV_ERR_MIS[x] = 1                    // MIS failure on Cyl x
                    ERR_SYM_MIS[x][0] = 1                // MIS_A symptom on Cyl x
                    Delivery the result to Error Management
                    If(4) (NLC_ENA_SCDN_NEW=1 and LV_ERR_MIS[x]=0 ->1)
                    Then(4)     ACTION_ERRM_EndWinScdn(MIS[x],1)
                    End If(4)
        Else(3)     If(4)   SYM_CYL_MIS_B1[x] = 0
                        And SYM_CYL_MIS_B4[x] = 0
                    Then(4)     LV_ERR_MIS[x] = 0
                                ERR_SYM_MIS[x] = 0H
                                Delivery the result to Error Management
                    Else(4)     ERR_SYM_MIS[x][0] = 0    // MIS_A symptom on Cyl x
                                Delivery the result to Error Management
                    EndIf(4)
        EndIf(3)
    EndFor(2)
Else(1)     // No MIS_A failure criterion
    For(2)  x = 0 to NC_CYL_NR - 1                       // Check for each cylinder x
        If(3)   SYM_CYL_MIS_B1[x] = 0
            And SYM_CYL_MIS_B4[x] = 0
        Then(3)     LV_ERR_MIS[x] = 0
                    ERR_SYM_MIS[x] = 0H
                    Delivery the result to Error Management
        Else(3)     ERR_SYM_MIS[x][0] = 0                // Symptom MIS_A erased
                    Delivery the result to Error Management
        EndIf(3)
    EndFor(2)
EndIf(1)
```

Individual cylinder errors set in case of MIS_B1 failure

If(1) (LV_END_MIS_B1 = 0 **or** LV_END_WIN_MIS_B1 = 1) *// Diagnosis during window MIS_B1*

Then(1)

If(2) LV_MIS_STATE_B1 = 1 *// MIS_B1 failure criterion*

For(3) x = 0 to NC_CYL_NR - 1 *// Check for each cylinder x*

If(4) SYM_CYL_MIS_B1*[x]* = 1

Then(4) LV_ERR_MIS[x] = 1 *// MIS failure on Cyl x*

ERR_SYM_MIS[x]*[1]* = 1 *// MIS_B1 symptom on Cyl x*

Delivery the result to Error Management

EndIf(4)

EndFor(3)

Else(2)

// No MIS_B1 failure criterion, MIS_B1 symptom on Cyl x evaluated once per driving cycle

No operation

EndIf(2)

If(5) NLC_ENA_SCDN_NEW=1

Then(5)

For(6) x = 0 to NC_CYL_NR – 1

If(7) **(**LV_ERR_MIS[x]=0→1 **or** LV_END_WIN_MIS_B1 = 1**)**

ACTION_ERRM_EndWinScdn(MIS[x],LV_ERR_MIS[x]+ LV_ERR_MIS_A_IN_WIN_Bold) // Check for each cylinder

EndIf(7)

EndFor(6)

EndIf(5)

EndIf(1)

Individual cylinder errors set in case of MIS_B4 failure

```
If(1) LV_MIS_STATE_B4 = 1                              // MIS_B4 failure criterion
    For(2)  x = 0 to NC_CYL_NR - 1                     // Check for each cylinder x
        If(3)       SYM_CYL_MIS_B4[x] = 1
        Then(3)     LV_ERR_MIS[x] = 1                  // MIS failure on Cyl x
                    ERR_SYM_MIS[x][2] = 1                   // MIS_B4 symptom on Cyl x
                    Delivery the result to Error Management

        Else(3)     If(4)  SYM_CYL_MIS_A[x] = 0
                           And  SYM_CYL_MIS_B1[x] = 0
                    Then(4)     LV_ERR_MIS[x] = 0
                                ERR_SYM_MIS[x] = 0H
                               Delivery the result to Error Management
                    Else(4)    ERR_SYM_MIS[x][2] = 0   // MIS_B4 symptom on Cyl x
                               Delivery the result to Error Management
                    EndIf(4)
        EndIf(3)
    EndFor(2)
Else(1)     // No MIS_B4 failure criterion
    For(2)  x = 0 to NC_CYL_NR - 1                     // Check for each cylinder x
        If(3)   SYM_CYL_MIS_A[x] = 0
                And  SYM_CYL_MIS_B1[x] = 0
        Then(3)     LV_ERR_MIS[x] = 0
                    ERR_SYM_MIS[x] = 0H
                    Delivery the result to Error Management
        Else(3)     ERR_SYM_MIS[x][2] = 0                   // Symptom MIS_B4 erased
                    Delivery the result to Error Management
        EndIf(3)
    EndFor(2)
EndIf(1)
```

If(1) **(**NLC_ENA_SCDN_NEW=1**)**

Then(1)

For(2) x = 0 to NC_CYL_NR - 1

If(3) **(**LV_ERR_MIS[x]=0 **and** LV_END_WIN_MIS_B4=1 **)**

Then(3) ACTION_ERRM_EndWinScdn (MIS[x], LV_ERR_IN_WIN_MIS_B4old + LV_ERR_MIS_A_IN_WIN_Bold) // previous values of lv_err_mis_a_in_win_b and lv_err_in_win_mis_b4.

Else If(4) **(**LV_ERR_MIS[x]=1 **and** LV_ERR_IN_WIN_MIS_B4=0->1 **)**

ACTION_ERRM_EndWinScdn (MIS[x], LV_ERR_IN_WIN_MIS_B4)

End If(4)

End If(3)

EndFor(2)

End If(1)

If **LV_ERR_MIS[x] = 0 -> 1**

then **Update N_MAX(_MIN)_SCDN_EQU_MIS_x_ENVD**

endif

End of Diagnosis:

If LV_ERR_MIS[x] = 1

Then LV_END_DIAG_MIS[x] = 1 *// for all cylinders*

Else LV_END_DIAG_MIS[x] = LV_END_MIS_B4 *// for all cylinders*

EndIf

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_TREAT_DIAG_MIS	-	0...1H	0...1	1	[-]
Misfire default failures generated per symptoms CARB_A, CARB_B1 & CARB_B4 (NLC_TREAT_DIAG_MIS = 0) or per specific cylinder misfire errors (NLC_TREAT_DIAG_MIS = 1)					

B.6.5 Random/Multiple cylinder misfire failure

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_MIS_MPL	V/O	0...1H	0...1	1	-
Diagnosis condition multiple cylinder misfire					
ERR_SYM_MIS_MPL	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Error symptom multiple cylinder misfire					
LV_ERR_MIS_MPL	V/O	0...1H	0...1	1	-
Present error flag multiple cylinder misfire					
LV_END_DIAG_MIS_MPL	V/O	0...1H	0...1	1	-
End of diagnosis flag					

Input data:

LV_MIS_STATE_A	LV_DC	LV_MIS_STATE_B1	LV_MIS_STATE_B4
SYM_CYL_MIS_A	SYM_CYL_MIS_B1	SYM_CYL_MIS_B4	LV_END_WIN_MIS_A
LV_END_WIN_MIS_B1	LV_END_WIN_MIS_B4	LC_MIS_INH	LV_CDN_MIS_A
LV_END_MIS_A	LV_END_MIS_B1	LV_END_MIS_B4	

Description:

If at the end of the concerned checking range (LV_END_WIN_MIS_xx), misfire is detected on two or more cylinders, then the “Multiple cylinder misfire” is detected.
The error is set directly without debounce
Error-symptoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> Multiple cylinder misfire detected     (= SYM_0)
│ │ └───> Random cylinder misfire detected       (= SYM_1)
│ └─────> -                                      (= SYM_2)
└───────> -                                      (= SYM_3)
```

Application conditions:

Initialisation: all outputs with 0 at LV_DC 0 −> 1 **or** ECU reset

Recurrence: ENRD segment synchronous,
After misfire detection, After misfire diagnosis

Activation:

If LV_DC = 1
And LC_MIS_INH = 0
And LV_CDN_MIS_A = 1
Then LV_CDN_DIAG_MIS_MPL = 1 *// diagnosis is active*
Else LV_CDN_DIAG_MIS_MPL = 0 *// diagnosis is passive*
Endif

Formula section:

Error detection

If **(** LV_MIS_STATE_A = 0 −> 1
And symptom MPL inside SYM_CYL_MIS_A is set to 1**)**
Or **(** LV_MIS_STATE_B1 = 1
And LV_END_WIN_MIS_B1 = 1
And symptom MPL inside SYM_CYL_MIS_B1 is set to 1**)**
Or **(** LV_MIS_STATE_B4 = 1
And LV_END_WIN_MIS_B4 = 1
And symptom MPL inside SYM_CYL_MIS_B4 is set to 1**)**
Then LV_ERR_MIS_MPL = 1
ERR_SYM_MIS_MPL = SYM_0
Delivery the result to Error Management
EndIf

If **(** LV_MIS_STATE_A = 0 −> 1
And symptom RDN inside SYM_CYL_MIS_A is set to 1**)**
Or **(** LV_MIS_STATE_B1 = 1
And LV_END_WIN_MIS_B1 = 1
And symptom RDN inside SYM_CYL_MIS_B1 is set to 1**)**
Or **(** LV_MIS_STATE_B4 = 1
And LV_END_WIN_MIS_B4 = 1
And symptom RDN inside SYM_CYL_MIS_B4 is set to 1**)**
Then LV_ERR_MIS_MPL = 1
ERR_SYM_MIS_MPL = SYM_1
Delivery the result to Error Management
Endif

If MPL bit is set to 0 in symptom SYM_CYL_MIS_A
And MPL bit is set to 0 in symptom SYM_CYL_MIS_B1
And MPL bit is set to 0 in symptom SYM_CYL_MIS_B4
Then SYM_0 symptom is removed from ERR_SYM_MIS_MPL
EndIf

If RDN bit is set to 0 in symptom SYM_CYL_MIS_A

And RDN bit is set to 0 in symptom SYM_CYL_MIS_B1

And RDN bit is set to 0 in symptom SYM_CYL_MIS_B4

Then SYM_1 symptom is removed from ERR_SYM_MIS_MPL

EndIf

If ERR_SYM_MIS_MPL = NO_SYM

Then LV_ERR_MIS_MPL = 0

Delivery the result to Error Management

EndIf

End of Diagnosis

(MIS_B4 is the longer failure criterion between MIS_A, MIS_B1 & MIS_B4)

If (LV_END_MIS_A = 1 **or** LV_END_MIS_B1 = 1)

And LV_ERR_MIS_MPL = 1

Then LV_END_DIAG_MIS_MPL = 1

Else LV_END_DIAG_MIS_MPL = LV_END_MIS_B4

EndIf

B.6.6 Misfire Diagnosis Failure Class

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
Misfire Cylinder 0 MIS_0	MIS_A	SYM_0	*NO*
	MIS_B1	SYM_1	
	MIS_B4	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
FAC_LAM_COR_1_ENVD
N_MIN_SCDN_EQU_MIS_0_ENVD
N_MIN_SCDN_EQU_MIS_0_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Misfire Cylinder 1 MIS_1	MIS_A	SYM_0	*NO*
	MIS_B1	SYM_1	
	MIS_B4	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
FAC_LAM_COR_1_ENVD
N_MIN_SCDN_EQU_MIS_1_ENVD
N_MIN_SCDN_EQU_MIS_1_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Misfire Cylinder 2 MIS_2	MIS_A	SYM_0	*NO*
	MIS_B1	SYM_1	
	MIS_B4	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
FAC_LAM_COR_1_ENVD
N_MIN_SCDN_EQU_MIS_2_ENVD
N_MIN_SCDN_EQU_MIS_2_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Misfire Cylinder 3 MIS_3	MIS_A	SYM_0	*NO*
	MIS_B1	SYM_1	
	MIS_B4	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TCO_ST
FAC_LAM_COR_1_ENVD
N_MIN_SCDN_EQU_MIS_3_ENVD
N_MIN_SCDN_EQU_MIS_3_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Random/ Multiple Misfire MIS_MPL	Multiple cylinder misfire	SYM_0	*NO*
	Random cylinder misfire	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory:

TCO_ST

FAC_LAM_COR_1_ENVD

TIA

VS

B.7 Crankshaft oscillation detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_STATE_CRK_OSC	V/O	0...1H	0...1	1	[-]
Status of crankshaft oscillation condition					
LV_CRK_OSC_DET_ACT	V	0...1H	0...1	1	[-]
Crankshaft oscillation detection active					
DRV0_ER_SUM_OSC	V	0...FFFFH	0...65535	1	[µs]
Integrated DRV0_ER value for amplitude criterion reseted every sign change or DRV0					
DRV0_ER_SUM_THD_OSC	V	0...FFFFH	0...65535	1	[µs]
Threshold for DRV0_ER_SUM_OSC to fulfill amplitude criteria					
SEG_SUM_FRQ_CRK_OSC	V	0...FFH	0...255	1	[-]
Segment counter for frequency criterion reseted every sign change or DRV0					
RATIO_VS_N_CRK_OSC	V	0...FFH	0...0.05	1.96E-04	[(km/h)/rpm]
Ratio VS / N for gear area determination					
CTR_TDC_CRK_OSC_ACT	V	0...FFH	0...255	1	[-]
TDC counter to enable crankshaft oscillation detection after misfiring detection					
LV_FRQ_CRIT_CRK_OSC	V	0...FFH	0...255	1	[-]
Critical crankshaft oscillaction frequency detected					
LV_AMPL_CRIT_CRK_OSC	V	0...FFH	0...255	1	[-]
Critical crankshaft oscillaction amplitude detected					

Input data:

N_32	LV_DET_CFM_MIS	VS	DRV0_ER
N	LOAD_MIS	LV_ENA_ER	LC_MIS_INH
LV_INH_CRK_OSC_DET			

B.7.1 General information

The misfire detection based on engine roughness index (ER) may be disturbed by crankshaft oscillations when single misfire (random) occurs, especially for front drive vehicles crankshaft in the low engine speed / high engine load area (in combination with a 3rd, 4th or 5th gear ratio). A single misfire acts as a drivetrain / crankshaft oscillations trigger, in this case the ER index is distrubed by these oscillations, in some conditions it is practically not possible to distinguish on ER index, real misfire towards speed drop caused through crankshaft speed oscillations.

The trigger for such oscillations can be a single misfire, an obstacle, a big torque change or others instantaneous conditions.

The principle of this crankshaft / drivetrain oscillation detection is to detect such an oscillation and then to fade-out the misfire detection for a short period to avoid over-detection.

Initialisation: *on ECU reset, on LC_MIS_INH 0 to 1 transition*

SEG_SUM_FRQ_CRK_OSC = 0

DRV0_ER_SUM_OSC = 0

LV_STATE_CRK_OSC = 0

Application conditions:

Activation/Deactivation:

RATIO_VS_N_CRK_OSC = VS / N

If LV_DET_CFM_MIS = 1

Then CTR_TDC_CRK_OSC_ACT = C_NR_TDC_MAX_CRK_OSC

ElseIf CTR_TDC_CRK_OSC_ACT ≠ 0

Then CTR_TDC_CRK_OSC_ACT = CTR_TDC_CRK_OSC_ACT - 1

EndIf

If LV_ENA_ER = 1

And LC_MIS_INH = 0

And LV_INH_CRK_OSC_DET = 0

And N_32 **<** C_N_MAX_CRK_OSC

And RATIO_VS_N_CRK_OSC **>** C_RATIO_VS_N_MIN_CRK_OSC

And CTR_TDC_CRK_OSC_ACT ≠ 0

Then LV_CRK_OSC_DET_ACT = 1 *// Crankshaft oscillation detection active*

Else LV_CRK_OSC_DET_ACT = 0 *// Crankshaft oscillation detection inactive*

SEG_SUM_FRQ_CRK_OSC = 0

DRV0_ER_SUM_OSC = 0

EndIf

Update rate: Segment

B.7.2 Detection description

The final decision about crankshaft oscillation consists of two parts:

- The detection of a critical engine roughness oscillation frequency.
- The detection of a critical engine roughness oscillation amplitude.

If both parts fulfilled at the same time their own detection criterion, a crankshaft oscillation is detected. The reason therefore is, that even at constant speed due to normal little speed variations a high frequency or at a normal acceleration phase a big amplitude is possible.

B.7.2.1 Determination of a critical engine roughness frequency

Application conditions:

Activation: LV_CRK_OSC_DET_ACT = 1

Deactivation: LV_CRK_OSC_DET_ACT = 0

Formula section:

If Bit sign between $DRV0\_ER_{(n-1)}$ and $DRV0\_ER_{(n)}$ changes

Then SEG_SUM_FRQ_CRK_OSC = IP_SEG_MIN_FRQ_CRK_OSC(N_32)

ElseIf SEG_SUM_FRQ_CRK_OSC ≠ 0

Then SEG_SUM_FRQ_CRK_OSC = SEG_SUM_FRQ_CRK_OSC - 1

EndIf

If SEG_SUM_FRQ_CRK_OSC ≠ 0

Then LV_FRQ_CRIT_CRK_OSC = 1 *// Critical crankshaft frequency status is active*

Else LV_FRQ_CRIT_CRK_OSC = 0

EndIf

B.7.2.2 Determination of a critical engine roughness oscillation amplitude

Application conditions:

Activation: LV_CRK_OSC_DET_ACT = 1

Deactivation: LV_CRK_OSC_DET_ACT = 0

Formula section:

If Bit sign between $DRV0\_ER_{(n-1)}$ and $DRV0\_ER_{(n)}$ changes

Then $DRV0\_ER\_SUM\_OSC_{(n)} = |DRV0\_ER_{(n)}|$

Else $DRV0\_ER\_SUM\_OSC_{(n)} = DRV0\_ER\_SUM\_OSC_{(n-1)} + |DRV0\_ER_{(n)}|$

// with saturation

EndIf

DRV0_ER_SUM_THD_OSC = IP_DRV0_ER_THD_OSC(N_32, LOAD_MIS)

If DRV0_ER_SUM_OSC > DRV0_ER_SUM_THD_OSC

Then LV_AMPL_CRIT_CRK_OSC = 1 *// Critical crankshaft amplitude status is active*

Else LV_AMPL_CRIT_CRK_OSC = 0

EndIf

B.7.2.3 Determination of crankshaft oscillation final status

Application conditions:

Activation: LV_CRK_OSC_DET_ACT = 1

Deactivation: LV_CRK_OSC_DET_ACT = 0

Formula section:

If in the same time, both conditions are active, a crankshaft oscillation status is triggered :

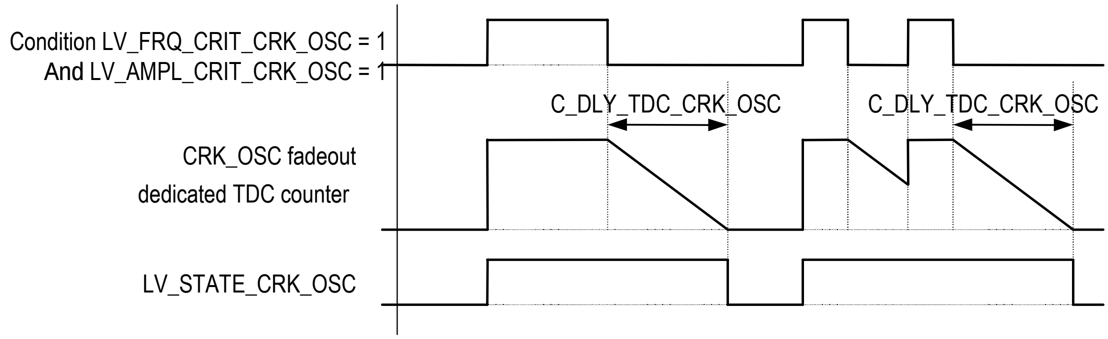
If LV_FRQ_CRIT_CRK_OSC = 1

And LV_AMPL_CRIT_CRK_OSC = 1

Then After condition rising edge triggering, LV_STATE_CRK_OSC flag is set to 1 as long as condition is true.

After condition falling edge triggering, LV_STATE_CRK_OSC flag is hold to 1 for a period of C_DLY_TDC_CRK_OSC tdc's, even if LV_CRK_OSC_DET_ACT is then set to 0.

EndIf



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_MAX_CRK_OSC	1	0...FFH	0...8160	32	[rpm]
Maximum speed for crankshaft oscillation calculation					
C_DLY_TDC_CRK_OSC	1	0...FFH	0...255	1	[-]
Active TDC's of crankshaft oscillation status after detection triggering					
C_NR_TDC_MAX_CRK_OSC	1	0...FFH	0...255	1	[-]
Maximum TDC after detected misfire to activate crankshaft oscillation detection process					
C_RATIO_VS_N_MIN_CRK_OSC	1	0...FFH	0...0.05	1.96E-04	[(km/h)/rpm]
Minimum VS / N ratio to activate crankshaft oscillation detection					
IP_SEG_MIN_FRQ_CRK_OSC	4	0...FFH	0...255	1	[-]
LDP_N_32_IP_SEG_MIN_FRQ_CRK_OSC	4	0...FFH	0...8160	32	[rpm]
Minimum number of segment after turn of speed direction for detecting critical high frequency					
IP_DRV0_ER_THD_OSC	6*6	0...FFFFH	0...65535	1	[µs]
LDP_N_32_IP_DRV0_ER_THD_OSC	6	0...FFH	0...8160	32	[rpm]
LDP_LOAD_MIS_IP_DRV0_ER_THD_OSC	6	0...7FFFH	0...99.99694	3.05E-03	[%]
DRV0 amplitude threshold to detect critical crankshaft oscillation amplitude					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_USE_CRK_OSC_MIS	1	0...1H	0...1	1	[-]
Crankshaft oscillation detection module used (=1) or stub version (=0)					

B.8 Rough road detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
WHEEL_GRD_MMV	V	0...FFFFH	0...999.98474	1.53E-02	[°/oo]
Filtered rough road value					
SUM_RR	V	0...FFH	0...255	1	[-]
Counter of rough road detection events.					
WHEEL_GRD_MMV_THD	V	0...FFFFH	0...999.98474	1.53E-02	[°/oo]
Threshold for rough road value					
CTR_T_DLY_RR	V	0...FFFFH	0...655350	10	[ms]
Fade out timer for engine roughness after rough road detection					
LV_STATE_RR	V/O	0...1H	0...1	1	[-]
Boolean for state of rough road detection (No : Yes).					
LV_ENA_T_SEG_RR	V	0...1H	0...1	1	[-]
Two consecutive WSS segment periods are valid					

Input data:

LV_AT	LV_INH_APP_RR	T_SEG_RR_0	T_SEG_RR_1
VS	LC_MIS_INH		

FUNCTION DESCRIPTION:

General information:

Rough road conditions must be detected to prevent erroneous misfire detection that could be caused by jollity tracks influence on crankshaft via the drive train.

The speed disturbance of vehicle wheels is used to evaluate the rough road conditions. Therefore, the EMS control unit uses an acquired signal coming from wheel speed sensors *(WHEEL_GRD)*.

The basically integration possibility consists by using magnetic sensor mounted on driven wheel on the right side (*preferred for cars with left driver*) where the presence probability of a jollity track is greater.

However, transmission from the left to the right side has also been observed depending on the chassis and other circumstances, so that a limited detection of unilateral rough-road conditions on the left side is also possible.

The number of integrated teeth for one segment depends on the number of teeth per revolution of the wheel speed. An angle of about 30 ° is preferable (e.g. an integration of 4 teeth is used for a wheel with 48 teeth).

Rough road event task:

The rough road detection task is triggered at the end of each wheel target segment.

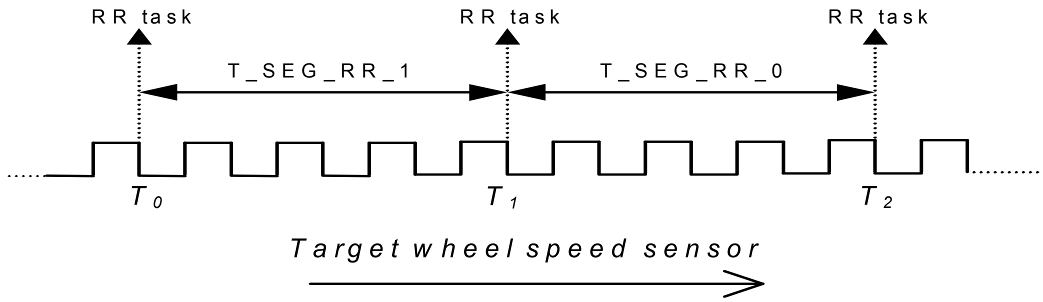
Process sequence:

- *Wheel target segment time acquisition*
- *Filtered & raw wheel gradient determination*
- *Rough road detection*

B.8.1.1 Wheel segment time acquisition control

General information:

T_SEG_RR_1 is the segment period preceding T_SEG_RR_0.



To be calculated in reliable conditions, rough road detection index needs a least two successive wheel segments.

Accordingly, the wheel segment acquisition process provide a control flag (LV_ENA_T_SEG_RR) who indicates that:

- no timer overflow and/or acquisition trouble occurs during last two consecutive wheel segments acquisitions
- two last wheel segments acquisitions are included within safe timing range *(speed sensor maximum timing acquirement & disablement for very high speed who reduce dedicated CPU load)*

Formula section:

If T_SEG_RR_0 is acquired & valid *(no dedicated timer overflow)*

And T_SEG_RR_0 <= C_T_SEG_RR_MAX

And T_SEG_RR_0 >= C_T_SEG_RR_MIN

And T_SEG_RR_1 is acquired & valid *(no dedicated timer overflow)*

And T_SEG_RR_1 <= C_T_SEG_RR_MAX

And T_SEG_RR_1 >= C_T_SEG_RR_MIN *(divide by zero protection)*

Then LV_ENA_T_SEG_RR = 1

Else LV_ENA_T_SEG_RR = 0

EndIf

Segment limitation reasons:

- C_T_SEG_RR_MAX

In lower vehicle speed range, the ABS sensor may no longer supply a valid output signal. In this case, the rough road value is computed only if two valid & consecutive segment periods have been measured in sequence.

- C_T_SEG_RR_MIN

For rough road index calculation, segment time have to be non-null to prevent division by zero in WHEEL_GRD calculation. It also allows limiting dedicated rough road detection CPU load at very high vehicle speed.

B.8.1.2 Rough road indexes determination

General information:

Rough road detection is based on a wheel gradient WHEEL_GRD evaluation. The raw values WHEEL_GRD are used in a floating averaging, to restrict suppression to actual rough road conditions. The current rough road value WHEEL_GRD_MMV from floating averaging is retained.
Note: To allow a fast filter initialisation, in case of LV_ENA_T_SEG_RR = 0 → 1 transition, $WHEEL\_GRD\_MMV_n$ and $WHEEL\_GRD\_MMV_{n-1}$, have to be initialised to WHEEL_GRD current value. Then for following valid samples, filtering is processed as described.

Update rate: rough road event task

Formula section:

If(1) LV_ENA_T_SEG_RR = 1

And LC_MIS_INH = 0

Then(1)

$$WHEEL\_GRD = \left| \frac{T\_SEG\_RR\_1 - T\_SEG\_RR\_0}{T\_SEG\_RR\_1} \right| * 1000 [‰]$$

If(2) LV_AT = 1

$WHEEL\_GRD\_MMV_n = WHEEL\_GRD\_MMV_{n-1}$

$+ C\_WHEEL\_GRD\_CRLC\_RR\_AT * (WHEEL\_GRD_n - WHEEL\_GRD\_MMV_{n-1})$

Else(2)

$WHEEL\_GRD\_MMV_n = WHEEL\_GRD\_MMV_{n-1}$

$+ C\_WHEEL\_GRD\_CRLC\_RR\_MT * (WHEEL\_GRD_n - WHEEL\_GRD\_MMV_{n-1})$

EndIf(2)

Else(1)

WHEEL_GRD = 0

WHEEL_GRD_MMV = 0

EndIf(1)

B.8.2 Detection of rough road status

Description:

The rough road detection is active within a valid engine speed range and when no error on rough road information is detected. If such error occurs, a calibration limp home rough road status can be applied according SI/customer requirements:

- LC_STATE_RR_LIH = 0, the misfire monitoring will not be inhibated if such error occurs. A specific Pcode can be saved if a misfire error is detected and if at the same time an error on CAN rough road information is set.
- LC_STATE_RR_LIH = 1, the misfire monitoring will be inhibated if such error occurs.

Update rate: rough road event task

Initialisation: At ECU reset event

LV_STATE_RR = 0

SUM_RR = 0

Activation condition:

	VS > C_VS_MIN_RR
And	VS < C_VS_MAX_RR
And	LV_INH_APP_RR = 0
And	LV_ENA_T_SEG_RR = 1
And	LC_MIS_INH = 0

Limp home:

If LV_INH_APP_RR = 1 *// OBD I wheel speed sensors diagnosis or any other*
// fade out conditions (ABS acting, ASR ...)

Then LV_STATE_RR = LC_STATE_RR_LIH *(see customer requirements)*

SUM_RR counter is frozen

CTR_T_DLY_RR = 0

EndIf

Formula section:

Rough road detection threshold determination:

If LV_AT = 1

Then WHEEL_GRD_MMV_THD = IP_WHEEL_THD_AT(VS)

Else WHEEL_GRD_MMV_THD = IP_WHEEL_THD_MT(VS)

EndIf

If WHEEL_GRD_MMV >= WHEEL_GRD_MMV_THD

Then LV_STATE_RR = 1

CTR_T_DLY_RR = C_T_DLY_RR

SUM_RR = SUM_RR + 1 *(with saturation)*

ElseIf CTR_T_DLY_RR != 0

Then LV_STATE_RR = 1

Else LV_STATE_RR = 0

EndIf

Remarks:
When the detection of rough road is effective, the logical value LV_STATE_RR is activated for a period of C_T_DLY_RR. This duration is triggered each time that WHEEL_GRD_MMV exceeds the threshold. The rough road events number is saved in the counter SUM_RR that is initialised at the ECU reset.

B.8.3 Rough Road Fade Out Timer Management

The timer dedicated to apply a delay on rough road status at each rough road detection occurence is management with a time task.

Update rate: 10 ms

Initialisation: At ECU reset event

CTR_T_DLY_RR = 0

Formula Section:

If CTR_T_DLY_RR != 0

Then CTR_T_DLY_RR = CTR_T_DLY_RR - 1 *(hexadecimal operation)*

Else LV_STATE_RR = 0

EndIf

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_WHEEL_GRD_CRLC_RR_MT	1	0...FFFFH	0...0.99998	1.53E-05	[-]
Correlation factor for floating averaging of the rough road raw values - MT vehicle					
C_WHEEL_GRD_CRLC_RR_AT	1	0...FFFFH	0...0.99998	1.53E-05	[-]
Correlation factor for floating averaging of the rough road raw values - AT vehicle					
C_T_DLY_RR	1	0...FFFFH	0...655350	10	[ms]
Fade out time for engine roughness after rough road detection.					
C_T_SEG_RR_MAX	1	0...FFFFH	0...0.26214	0.000004	[s]
Maximum segment duration for rough road detection.					
C_T_SEG_RR_MIN	1	0...FFFFH	0...0.26214	0.000004	[s]
Minimum segment duration for rough road detection.					
C_VS_MAX_RR	1	0...FFH	0...255	1	[km/h]
Maximum vehicle speed for rough road detection.					
C_VS_MIN_RR	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed for rough road detection.					
IP_WHEEL_THD_MT	9	0...FFFFH	0...999.98474	1.53E-02	[°/oo]
LDP_VS_IP_WHEEL_THD_MT	9	0...FFH	0...255	1	[km/h]
Threshold versus vehicle speed for rough road detection with the segment period (MT version)					
IP_WHEEL_THD_AT	9	0...FFFFH	0...999.98474	1.53E-02	[°/oo]
LDP_VS_IP_WHEEL_THD_AT	9	0...FFH	0...255	1	[km/h]
Threshold versus vehicle speed for rough road detection with the segment period (AT version)					
LC_STATE_RR_LIH	1	0...1H	0...1	1	[-]
Limp home rough road detection status (0...1)					

B.9 Additional functions for Evap. Syst. Monitoring

B.9.1 General activation condition

Input data:

CONF_DIAGCP			

Application conditions:

Activation: **If** CONF_DIAGCP > 0

Recurrence: 1000ms

General Information:

The additional functions for Evap. Syst. Monitoring must only be calculated if CONF_DIAGCP > 0

B.9.2 Minimum drive time and external inhibit bit

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAGCP_EXT_FCT	V/O	0 ... 1H	0 ... 1	1	-
Boolean for inhibitation of EVAP_DIAG by Additional functions for Evap. Syst. Monitoring					
T_VS_MIN_INI_DIAGCP	V	0...FFFFH	0...65535	1	s
Timer for VS-condition fulfilled after start of DC					
T_ES_DIAGCP	V	0...FFFFH	0...65535	1	min
ENG_OFF_TIME at DC begin					
TCO_DIF_DIAGCP	V	0...FEH	-48...142.5	0.75	°C
Coolant temperature difference from TCO_STOP to TCO_ST					
TAM_DIF_DIAGCP	V	0...FEH	-48...142.5	0.75	°C
Ambient air temperature difference from TAM_ST to TIA_ST					

Input data:

LV_DC	LV_AMP_GRD_DIAGCP_INH	LV_INH_T_ERU_DIAGCP	VS
T_AST	LV_DIAGCP_ACT_EXT_ADJ	T_VS_MIN_DIAG	CTR_ABC_SOV
T_ES	LV_INH_FTL_NEW_DIAGCP	LV_T_ES_NOT_PLAUS	LV_ERR_TIA
TCO_STOP	TCO_ST	LV_ERR_TCO	
TAM_ST	TIA_ST	CONF_TAM	LV_TAM_VLD_DIAGCP

Application conditions:

Initialisation: **If** LV_DC = 0 → 1
then T_VS_MIN_INI_DIAGCP = C_T_VS_MIN_INI_DIAGCP
LV_INH_DIAGCP_EXT_FCT = 1
If LC_T_ES_ACT = 1
and LV_T_ES_NOT_PLAUS = 0
then T_ES_DIAGCP = T_ES

```
                  else   if     LV_ERR_TCO = 0    and    LV_ERR_TIA = 0
                         then   TCO_DIF_DIAGCP = TCO_STOP – TCO_ST
                                TAM_DIF_DIAGCP = TAM_ST – TIA_ST
                                T_ES_DIAGCP = IP_T_ES_DIAGCP* IP_FAC_T_ES_DIAGCP
                         else   T_ES_DIAGCP = C_T_ES_DIAGCP
                         endif
                         endif
              endif
Activation:       If     LV_DC = 1
```

Formula section:

```
If       T_VS_MIN_INI_DIAGCP > 0
then     if     VS >= C_VS_MIN_INI_DIAGCP
         then   T_VS_MIN_INI_DIAGCP is decremented
         else   T_VS_MIN_INI_DIAGCP = C_T_VS_MIN_INI_DIAGCP
         endif
endif
```

B.9.2.1 Coordiation of LV_INH_DIAGCP_EXT_FCT

```
If       LV_DIAGCP_ACT_EXT_ADJ = 0
then     if     [ (T_VS_MIN_INI_DIAGCP = 0 and T_AST > C_T_AST_INI_DIAGCP)
                and ( CONF_TAM <> 0 or LV_TAM_VLD_DIAGCP = 1 )
                and LV_AMP_GRD_DIAGCP_INH = 0
                and LV_INH_T_ERU_DIAGCP = 0
                and LV_INH_FTL_NEW_DIAGCP = 0
                and T_VS_MIN_DIAG = 0
                and CTR_ABC_SOV = 0 ]
         then   LV_INH_DIAGCP_EXT_FCT = 0
         else   LV_INH_DIAGCP_EXT_FCT = 1
         endif
else     LV_INH_DIAGCP_EXT_FCT = 0
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MIN_INI_DIAGCP	1	0...FFH	0...255	1	km/h
VS threshold for initialisation					
C_T_VS_MIN_INI_DIAGCP	1	0...FFFFH	0...65535	1	s
Maximum time threshold for initialisation					
C_T_AST_INI_DIAGCP	1	0...FFFFH	0...6553.5	0.1	s
Time threshold for initialisation after start					
C_T_ES_DIAGCP	1	0...FFFFH	0...65535	1	min
Time threshold for initialisation after start					
LC_T_ES_ACT	1	0...1H	0...1	1	-
Manual switch to select source between T_ES or TCO_STOP for engine off timer					
IP_T_ES_DIAGCP	8*6	0...FFFFH	0...65535	1	min
LDP_TCO_DIF_DIAGCP	8	0...FEH	-48...142.5	0.75	°C
LDPM_TAM	6	0...FEH	-48...142.5	0.75	°C
Substituent engine off timer without timer circuit					
IP_FAC_T_ES_DIAGCP	8	0...FFFFH	0...1.999969	3.052E-5	-
LDP_TAM_DIF_DIAGCP	8	0...FEH	-48...142.5	0.75	°C
Substituent engine off timer correction without timer circuit					

B.9.3 Inhibition in case of high altitude and high fuel temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_AMP_TEMP_FUEL_DIAGCP	V/O	0 ... 1H	0 ... 1	1	-
Boolean for inhibitation of EVAP_DIAG in case of high fuel temperature and high altitude					

Input data:

TEMP_COR_DIAGCP	AMP		

Application conditions:

Initialisation: at reset: *LV_INH_AMP_TEMP_FUEL_DIAGCP = 0*

Activation: LV_DC = 1

Formula section:

LV_INH_AMP_TEMP_FUEL_DIAGCP = ID_AMP_TEMP_FUEL

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_AMP_TEMP_FUEL	8*8	0...1H	0...1	1	-
LDP_TEMP_COR_DIAGCP	8	0...FEH	-48...142.5	0.75	°C
LDP_AMP_ID_AMP_TEMP_FUEL	8	0...FFFFH	0...5434	0.083	hPa
Map due to inhibit DIAGCP in case of high fuel temperature and high altitude					

B.9.4 Refuelling detection / Inhibition of leak detection in case of refuelling

B.9.4.1 Refuelling detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FTL_NEW	V	0...1H	0...1	1	-
Boolean for refuelling detection (high FTL-difference)					
T_MAX_FTL_NEW	V	0...FFFFH	0...65535	1	sec
Timer after detected refuelling					
FTL_MMV_OLD	V/S	0...FFFFH	0...100	0.0015	%
Fuel level in previous driving cycle (used for refuelling-detect at stopped engine)					
FTL_MMV_ST	V	0...FFFFH	0...100	0.0015	%
Starting Fuel level at new DC					

Input data:

TAM	FTL_MMV	LV_DT	T_VS_MIN_INI_DIAGCP
LV_FTL_NEW_END	LV_ST_END	LV_DC	T_AST

General information:

The purpose of the refulling detection is to correct the fuel temperature in case of refuelling.

Initialisation:

If there are no stored values or a checksum error in adaptation values has occured
then set default value (FTL_MMV_OLD = FTL_MMV)
else use previously stored values
endif

If LV_DC = 0 →1
then LV_FTL_NEW = 0
T_MAX_FTL_NEW = 0
endif

If T_AST ≥ C_T_MAX_FTL_NEW
and LV_DC = 1
then FTL_MMV_ST = FTL_MMV (once)
endif

If LV_DC = 1 → 0
then LV_FTL_NEW = 0
T_MAX_FTL_NEW = 0
if LV_FTL_NEW_END = 1 *(FTL_MMV_OLD is saved*
then FTL_MMV_OLD = FTL_MMV *in a non volatile memory after*
endif *refueling detection calculation)*
endif

FTL_MMV_OLD value is allowed if the calculation of a detected refuelling (LV_FTL_NEW_END = 1) is finished in the last driving cycle. And it is updated with FTL_MMV after refuelling (stopped or running) is detected (see formular section).

Activation: **If** LV_DC =1

and TAM ≥ C_TAM_MIN_FTL_NEW

and T_AST ≥ C_T_MAX_FTL_NEW + 1 [sec]

Formula section:

- Refuelling at stopped engine happened (between current and last driving-cycle)

```
If      FTL_MMV_ST – FTL_MMV_OLD > C_FTL_MMV_DIF
then    LV_FTL_NEW = 1
        FTL_MMV_OLD = FTL_MMV_ST
endif
```

- Refuelling at running engine

```
If      LV_DT = 0
then    if     FTL_MMV – FTL_MMV_ST > C_FTL_MMV_DIF
        then   LV_FTL_NEW = 1
               FTL_MMV_ST = FTL_MMV
               FTL_MMV_OLD = FTL_MMV
        endif
endif
```

- Timer for refuelling detection:

```
If      LV_FTL_NEW  = 1
then    T_MAX_FTL_NEW is incremented
endif

If      T_MAX_FTL_NEW  > C_T_MAX_FTL_NEW
then    LV_FTL_NEW = 0
        T_MAX_FTL_NEW  = 0
        "perform once  T_AST_COR correction;  see next chapter 1.5.3 Calculation
        of fuel temperature"
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TAM_MIN_FTL_NEW	1	0...FEH	-48...142.5	0.75	°C
Minimum TAM threshold for refuelling detection					
C_FTL_MMV_DIF	1	0...FFFFH	0...100	0.0015	%
FTL-threshold for refuelling detection (only positive values)					
C_T_MAX_FTL_NEW	1	0...FFFFH	0...6553.5	100	ms
Timer for reset refuelling detection and offset-correction of T_AST_COR					

B.9.4.2 Inhibition of leak detection in case of refuelling

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_FTL_NEW_DIAGCP	V/O/S	0...1H	0...1	1	-
Boolean for inhibition of leak detection diagnosis in case of refuelling					
T_INH_FTL_NEW_DIAGCP	V/S	0...FFFFH	0...65535	1	sec
Timer for inhibition of leak detection diagnosis					

Input data:

LV_DC	T_ES_DIAGCP	TAM	C_TAM_MIN_FTL_NEW
LV_FTL_NEW	LV_T_ES_NOT_PLAUS		

Application conditions:

Initialisation:

If there are no stored values or a checksum error in adaptation values has occured
then set default values
(LV_INH_FTL_NEW_DIAGCP = 0)
(T_INH_FTL_NEW_DIAGCP = 0)
else use previously stored values
endif

If LV_DC = 0 → 1
and LV_T_ES_NOT_PLAUS = 0
then **if** T_ES_DIAGCP >= C_T_ES_INH_FTL min
then LV_INH_FTL_NEW_DIAGCP = 0
T_INH_FTL_NEW_DIAGCP = 0
else LV_INH_FTL_NEW_DIAGCP saved values remain
T_INH_FTL_NEW_DIAGCP saved values remain
endif
endif

Recurrence: 1000ms

Activation: **If** LV_DC =1 **and** TAM ≥ C_TAM_MIN_FTL_NEW

Formula section:

If LV_FTL_NEW = 0 → 1
then T_INH_FTL_NEW_DIAGCP = C_T_INH_FTL_NEW
endif

If LV_DC = 1
then **if** T_INH_FTL_NEW_DIAGCP > 0 [sec]
then T_INH_FTL_NEW_DIAGCP is decremented to 0 [sec]
LV_INH_FTL_NEW_DIAGCP = 1
else LV_INH_FTL_NEW_DIAGCP = 0
endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_ES_INH_FTL	1	0...FFFFH	0...65535	1	min
Engine off time threshold for inhibition of leak detection function in case of refuelling					
C_T_INH_FTL_NEW	1	0...FFFFH	0...65535	1	sec
Timer for inhibition of leak detection diagnosis					

B.9.5 Fuel temperature model

B.9.5.1 Ambient temperature for fuel temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TAM_FUEL_TEMP	V/S	0...FEH	-48...142.5	0.75	°C
Temperature ambient for fuel temperature model					
T_VS_MIN_TAM_FUEL_TEMP	V	0...FFFFH	0...65535	1	s
Timer for VS-condition fulfilled (after initialisation of fuel temperature model)					
T_FTL_MMV_NEW_END	V	0...FFFFH	0...65535	1	[s]
Timer for calculation of refuelling detection					

Input data:

TAM	LV_DC	T_VS_MIN_INI_DIAGCP	VS
C_VS_MIN_INI_DIAGCP			

General information:

The most important input data for the fuel temperature model is the ambient temperature TAM. The ambient temperature sensor is exposed to rediation heat from the engine and the asphalt road. Therefore, an approximately reliable ambient temperature from the sensor is only useful over a determined engine speed (C_VS_MIN_INI_DIAGCP). Under this engine speed threshold the last TAM value is frozen.

Application conditions:

Initialisation: **If** LV_DC = 1
and T_VS_MIN_INI_DIAGCP 1 → 0
then TAM_FUEL_TEMP = TAM
T_VS_MIN_TAM_FUEL_TEMP = C_T_VS_MIN_TAM_FUEL_TEMP
T_FTL_MMV_NEW_END = 1 [sec]
endif

Formula section:

If T_VS_MIN_INI_DIAGCP = 0
and VS ≥ C_VS_MIN_INI_DIAGCP
then T_VS_MIN_TAM_FUEL_TEMP is decremented
else T_VS_MIN_TAM_FUEL_TEMP
= C_T_VS_MIN_TAM_FUEL_TEMP
endif

If T_VS_MIN_TAM_FUEL_TEMP = 0
then TAM_FUEL_TEMP = TAM
else TAM_FUEL_TEMP keeps old value (frozen last TAM_FUEL_TEMP - value)
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_VS_MIN_TAM_FUEL_TEMP	1	0...FFFFH	0...65535	1	s
Minimum time for VS-condition (after initialisation of fuel temperature model)					

B.9.5.2 Engine running time until initialisation of fuel temperature model

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_AST_INIT_COR	V	0...FFFFH	0...65535	1	sec
Engine running time until initialisation of fuel temperature model					
T_AST_INIT_COR_OLD	V/S	0...FFFFH	0...65535	1	sec
Saved engine running time until initialisation of fuel temperature model					

Input data:

LV_DC	T_VS_MIN_INI_DIAGCP	LV_T_ES_NOT_PLAUS	T_ES_DIAGCP

General information:

The engine running time from engine start until the initialisation of the fuel temperature model must be taken into account.

Application conditions:

Initialisation:

If there are no stored values or a checksum error in adaptation values has occured
then set default values
(T_AST_INIT_COR_OLD = 0 [sec])
else use previously stored values
endif

If LV_DC = 0 → 1
and LV_T_ES_NOT_PLAUS = 0
then T_AST_INIT_COR = T_AST_INIT_COR_OLD
* IP_FAC_T_AST_INIT_COR_OLD (T_ES_DIAGCP)
endif

If LV_DC = 1 → 0
then **if** T_VS_MIN_INI_DIAGCP > 0 [sec]
then T_AST_INIT_COR_OLD = T_AST_INIT_COR
else T_AST_INIT_COR_OLD = 0 [sec]
(T_AST_INIT_COR_OLD is saved in a non volatile memory)
endif
endif

Formula section:

If LV_DC = 1 **and** T_VS_MIN_INI_DIAGCP > 0 s
then T_AST_INIT_COR = T_AST_INIT_COR + 1s
else **if** LV_DC = 1 **and** T_VS_MIN_INI_DIAGCP = 1 → 0
then T_AST_INIT_COR = T_AST_INIT_COR * C_FAC_T_AST_INIT_COR
last T_AST_INIT_COR value is frozen (no further calculation)
endif
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_T_AST_INIT_COR	1	0...FFH	0...1.99218	0.0078125	-
Factor for engine running time until initialisation of fuel temperature model					
IP_FAC_T_AST_INIT_COR_OLD	8	0...FFH	0...1.99218	0.0078125	-
LDPM_T_ES_DIAGCP	8	0...FFFFH	0...6.5535E+4	1	min
Correction factor for T_AST_INIT_COR_OLD					

B.9.5.3 Calculation of fuel temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEMP_COR_DIAGCP	V/O	0...FEH	-48...142.5	0.75	°C
Temperature correction dependent on fuel temperature					
TEMP_COR_DIAGCP_OLD	V/S	0...FEH	-48...142.5	0.75	°C
Saved fuel temperature					
TEMP_COR_DIAGCP_IT	V	0...FE000000H	-48...142.5	44.70E-9	°C
Temperature correction dependent on fuel temperature - 16 bit					
TEMP_COR_DIAGCP_STAT	V	0...FEH	-48...142.5	0.75	°C
Steady state fuel temperature					
T_AST_COR	V	8000...7FFFH	-32768...32767	1	s
Corrected time after start					
T_AST_ADD	V	8000...7FFFH	-32768...32767	1	s
initial additive time after start					
T_AST_ADD_OLD	V/S	8000...7FFFH	-32768...32767	1	s
Saved engine running time from last DC					
T_AST_COR_FTL_NEW	V	8000...7FFFH	-32768...32767	1	sec
Correction time in case of refuelling					
LV_FTL_NEW_END	V	0...1H	0...1	1	-
Refuelling calculation finished					
T_CRLC_H	V	0...FFH	0...255	1	sec
Timer for using high correlation constant for fuel temperature decrease or increase					
CRLC_TEMP_FUEL	V	0...FFFFH	0...0.9998	1.53E-5	-
Correlation factor for fuel temperature filtering					

Input data:

LV_ERR_TAM	LV_DC	TCO	TAM_FUEL_TEMP
LV_T_ES_NOT_PLAUS	VS	T_VS_MIN_INI_DIAGCP	T_AST_INIT_COR
T_FTL_MMV_NEW_END	T_ES_DIAGCP	T_MAX_FTL_NEW	C_T_MAX_FTL_NEW

Application conditions:

Initialisation: ***If*** there are no stored values or a checksum error in adaptation values has occured

then set default values
(T_AST_ADD_OLD = 0 [sec]
; TEMP_COR_DIAGCP_OLD = 0 °C)

else use previously stored values

If LV_DC 0 → 1 *(initialisation in 1.2 is already done)*
and LV_T_ES_NOT_PLAUS = 0
then T_AST_ADD = T_AST_ADD_OLD * IP_FAC_T_AST_ADD_OLD
T_AST_COR = T_AST_ADD (set start-value for T_AST_COR)
LV_FTL_NEW_END = 0
T_CRLC_H = 0
If T_ES_DIAGCP ≥ C_MAX_T_ES_DIAGCP
then TEMP_COR_DIAGCP = TCO
else TEMP_COR_DIAGCP = TEMP_COR_DIAGCP_OLD
* IP_FAC_TEMP_COR_DIAGCP_OLD
endif
TEMP_COR_DIAGCP_IT = TEMP_COR_DIAGCP
endif

If LV_DC = 1 → 0
then T_AST_ADD_OLD = T_AST_COR
(T_AST_ADD_OLD is saved in a non volatile memory)
TEMP_COR_DIAGCP_OLD = TEMP_COR_DIAGCP
(TEMP_COR_DIAGCP_OLD is saved in a non volatile memory)

Activation: **If** LV_DC =1 **and** T_VS_MIN_INI_DIAGCP = 0

<u>Formula section:</u>

1. *Start value of T_AST_COR (with engine running time from engine start to initialisation of fuel temperature model)*

   T_AST_COR = T_AST_COR + T_AST_INIT_COR

   <u>Note:</u> The start value of T_AST_COR is calculated only once since T_VS_MIN_INI_DIAGCP changes to 0 [sec] !

2. *Corrected engine running time*

If T_AST_COR < C_T_AST_COR_MAX
then T_AST_COR + 1s
else T_AST_COR keeps old value

3. *Correction (once) in case of refuelling if timer is elapsed after refuelling detection*

If T_MAX_FTL_NEW > C_T_MAX_FTL_NEW
then T_AST_COR_FTL_NEW =
IP_T_AST_COR_FTL_NEW (T_AST_COR, TAM_FUEL_TEMP)

LV_FTL_NEW_END = 1

T_AST_COR = T_AST_COR + T_AST_COR_FTL_NEW

endif

4. Timer for high correlation constant after refuelling

If T_MAX_FTL_NEW > C_T_MAX_FTL_NEW

then T_CRLC_H = C_T_CRLC_H

endif

If T_CRLC_H > 0

then T_CRLC_H is decremented

endif

5. Calculation of refuelling detection is finished without refuelling detection

If T_FTL_MMV_NEW_END

> C_T_MAX_FTL_NEW + 1[sec] *(once fulfilled!)*

then LV_FTL_NEW_END = 1

T_FTL_MMV_NEW_END keeps old value

else T_FTL_MMV_NEW_END is incremented

endif

LV_FTL_NEW_END is also set, if no refuelling is detected but the point of time for the calculation of refuelling is finished

6. Steady state fuel temperature

If LV_ERR_TAM = 0

and LV_T_ES_NOT_PLAUS = 0

then TEMP_COR_DIAGCP_STAT = IP_TEMP_FUEL (T_AST_COR, TAM_FUEL_TEMP)

else TEMP_COR_DIAGCP_STAT = C_TEMP_FUEL_LIH

endif

7. Calculation of the estimated actual fuel temperature

If T_CRLC_H > 0

then CRLC_TEMP_FUEL = C_CRLC_TEMP_FUEL_INC_DEC

else **if** $TEMP\_COR\_DIAGCP\_STAT_{(N)} \geq TEMP\_COR\_DIAGCP\_IT_{(N)}$

then CRLC_TEMP_FUEL = ID_CRLC_TEMP_FUEL_INC (VS, FTL_MMV)

else CRLC_TEMP_FUEL = ID_CRLC_TEMP_FUEL_DEC (VS, FTL_MMV)

endif

endif

$$TEMP\_COR\_DIAGCP\_IT_{(N)} = TEMP\_COR\_DIAGCP\_IT_{(N-1)} + CRLC\_TEMP\_FUEL$$

* (TEMP_COR_DIAGCP_STAT$_{(N)}$ - TEMP_COR_DIAGCP_IT$_{(N-1)}$)

- ***Conversion to 8 bit***

If LV_ERR_TAM = 0
and LV_T_ES_NOT_PLAUS = 0
then TEMP_COR_DIAGCP = TEMP_COR_DIAGCP_IT (high byte of high word)
else TEMP_COR_DIAGCP = C_TEMP_FUEL_LIH
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TEMP_FUEL	12*12	0...FEH	-48...142.5	0.75	°C
LDP_T_AST_COR_IP_TEMP_FUEL	12	0...FFFFH	-32768...32767	1	s
LDPM_TAM_FUEL_TEMP	12	0...FEH	-48...142.5	0.75	°C
Fuel temperature					
C_TEMP_FUEL_LIH	1	0...FEH	-48...142.5	0.75	°C
Fuel temperature limp home value					
ID_CRLC_TEMP_FUEL_INC	6*8	0...FFFFH	0...0.99998	1.53E-5	-
LDPM_VS_EVAM_1	6	0...FFH	0...255	1	km/h
LDPM_FTL_MMV	8	0...FFFFH	0...100	0.0015	%
Correlation constant for fuel temperature increase					
ID_CRLC_TEMP_FUEL_DEC	6*8	0...FFFFH	0...0.99998	1.53E-5	-
LDPM_VS_EVAM1	6	0...FFH	0...255	1	km/h
LDPM_FTL_MMV	8	0...FFFFH	0...100	0.0015	%
Correlation constant for fuel temperature decrease					
C_T_AST_COR_MAX	1	8000...7FFFH	-32768...32767	1	s
Maximum time for T_AST_COR					
C_CRLC_TEMP_FUEL_INC_DEC	1	0...FFFFH	0...0.99998	1.53E-5	-
Correlation constant for fuel temperature decrease					
IP_FAC_T_AST_ADD_OLD	8	0...FFH	0...1.99218	0.0078125	-
LDPM_T_ES_DIAGCP	8	0...FFFFH	0...6.5535E+4	1	min
Correction factor for T_AST_ADD_OLD depending on T_ES					
IP_T_AST_COR_FTL_NEW	8*12	0...FFH	-32768...32512	256	sec
LDP_T_AST_COR_IP_T_AST_COR	8	0...FFFFH	-32768...32767	1	sec
LDPM_TAM_FUEL_TEMP	12	0...FEH	-48...142.5	0.75	°C
Time correction in case of refuelling					
C_MAX_T_ES_DIAGCP	1	0...FFFFH	0...65535	1	min
Maximum engine stop time threshold for initialisation of start fuel temperature value					
IP_FAC_TEMP_COR_DIAGCP_OLD	8	0...FFH	0...1.99218	0.0078125	-
LDPM_T_ES_DIAGCP	8	0...FFFFH	0...6.5535E+4	1	min
Correction factor for TEMP_COR_DIAGCP_OLD					
C_T_CRLC_H	1	0...FFH	0...255	1	sec
Maximum timer for using high correlation constant for fuel temperature decrease or increase					

B.9.6 AMP_GRD

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_AMP_GRD_DIAGCP_INH	V	0 ... 1H	0 ... 1	1	-
Boolean for detection of a high ambient pressure gradient					
AMP_GRD	V	8000 ... 7FFFH	-2717 ...2717	0.0829	hPa
Ambient pressure gradient					
AMP_OLD	V	0 ... FFFFH	0 ... 5434	0.0829	hPa
Ambient pressure value - old					
TEMP_COR_GRD	V	80...7FH	-96...95.25	0.75	°C
Temperature correction gradient					
TEMP_COR_OLD	V	0...FEH	-48...142.5	0.75	°C
Temperature correction value - old					
T_AMP_GRD_SUM	V	0...FFFFH	0...65535	1	sec
Timer between two AMP measurements					
T_INH_AMP_GRD_DIAGCP	V	0...FFFFH	0...65535	1	sec
Timer for disabling DIAGCP					

Input data:

AMP_AD	T_VS_MIN_INI_DIAGCP	TEMP_COR_DIAGCP	

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to detect high ambient pressure changes in a defined time. In this case the evaporative system diagnosis is not allowed.

Application conditions:

Initialisation: LV_DC = 1 **and** T_VS_MIN_INI_DIAGCP 1-> 0

AMP_GRD = 0 hpa

AMP_OLD = AMP_AD

TEMP_COR_OLD = TEMP_COR_DIAGCP (must be calculated once before)

TEMP_COR_GRD = 0

T_AMP_GRD_SUM = 0 [sec]

T_INH_AMP_GRD_DIAGCP = 0 [sec]

LV_AMP_GRD_DIAGCP_INH = 0

Recurrence: 1000ms

Activation: **If** LV_DC =1 **and** T_VS_MIN_INI_DIAGCP = 0

Deactivation: Activation condition not met

Formula section:

```
If      T_AMP_GRD_SUM   >= C_T_AMP_GRD_SUM_MAX
then    AMP_GRD =  AMP_AD – AMP_OLD
        TEMP_COR_GRD = TEMP_COR_DIAGCP - TEMP_COR_OLD
        T_AMP_GRD_SUM = 0
        AMP_OLD = AMP_AD
        TEMP_COR_OLD = TEMP_COR_DIAGCP
else    T_AMP_GRD_SUM is incremented
endif

If      T_AMP_GRD_SUM = 0
and   ( AMP_GRD > C_AMP_GRD_MAX_DIAGCP
   or   AMP_GRD < C_AMP_GRD_MIN_DIAGCP
   or   TEMP_COR_GRD > C_TEMP_COR_GRD_MAX_DIAGCP
   or   TEMP_COR_GRD < C_TEMP_COR_GRD_MIN_DIAGCP )
then    T_INH_AMP_GRD_DIAGCP = C_T_INH_AMP_GRD_DIAGCP
endif

If      T_INH_AMP_GRD_DIAGCP > 0
then    LV_AMP_GRD_DIAGCP_INH = 1
        T_INH_AMP_GRD_DIAGCP is decremented
else    LV_AMP_GRD_DIAGCP_INH = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_AMP_GRD_SUM_MAX	1	0...FFFFH	0...65535	1	s
Time between two AMP measurements					
C_T_INH_AMP_GRD_DIAGCP	1	0...FFFFH	0...65535	1	s
Inhibit time for leak detection function					
C_AMP_GRD_MAX_DIAGCP	1	8000 ... 7FFFH	-2717 ...2717	0.0829	hPa
Maximum AMP_GRD for leak detection function					
C_AMP_GRD_MIN_DIAGCP	1	8000 ... 7FFFH	-2717 ...2717	0.0829	hPa
Minimum AMP_GRD for leak detection function					
C_TEMP_COR_GRD_MAX_DIAGCP	1	80...7FH	-96...95.25	0.75	°C
Maximal temperature correction gradient					
C_TEMP_COR_GRD_MIN_DIAGCP	1	80...7FH	-96...95.25	0.75	°C
Minimal temperature correction gradient					

B.9.7 Inhibition Evaporative System Monitoring after defined engine running time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_T_ERU_DIAGCP	V	0...1H	0...1	1	-
Boolean for inhibition of evaporative system monitoring due to long engine running time					
LV_INH_T_ERU_DIAGCP_OLD	V/S	0...1H	0...1	1	-
Saved boolean for inhibition of evaporative system monitoring due to long engine running time					

Input data:

T_AST_COR	LV_DC	T_ES_DIAGCP	

General information:

Application conditions:

Initialisation: LV_INH_T_ERU_DIAGCP = 0

If LV_DC = 1 → 0
then LV_INH_T_ERU_DIAGCP_OLD = LV_INH_T_ERU_DIAGCP

Activation: LV_DC = 1

Formula section:

If LV_INH_T_ERU_DIAGCP_OLD = 0 (LV_INH_T_ERU_DIAGCP_OLD is saved in a non volatile memory)

then **if** T_AST_COR ≥ C_T_AST_COR_MAX_ERU
then LV_INH_T_ERU_DIAGCP = 1
else LV_INH_T_ERU_DIAGCP = 0
endif

else **if** T_ES_DIAGCP ≥ C_T_MAX_T_ES_ERU
and T_AST_COR < C_T_AST_COR_MAX_ERU
then LV_INH_T_ERU_DIAGCP = 0
else LV_INH_T_ERU_DIAGCP = 1
endif

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_AST_COR_MAX_ERU	1	8000...7FFFH	-32768...32767	1	s
Maximum allowed engine running time					
C_T_MAX_T_ES_ERU	1	0...FFFFH	0...65535	1	min
Maximum allowed engine off time					

B.9.8 Correction factor due to fuel quality

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TI_ST_AD_DIAGCP	O/V	0...FFH	0...1.992	0.0078	-
Fuel quality correction factor					
FAC_TI_ST_AD_DIAGCP_OLD	O/V/S	0...FFH	0...1.992	0.0078	-
Saved fuel quality correction factor					

Input data:

FAC_FQ_ST_AD	LV_FQ_ST_AD_END_1	TCO_ST	

FUNCTION DESCRIPTION:

General information:

The purpose of the function is to correct the actual leakage area (see chapter Evaporative System Monitoring) in case of fuels with high fuel evaporation or fuels with high vapor pressure. Fuels with high vapor pressure (low values of FAC_TI_ST_AD_DIAGCP) leads to overdetection in the leak detection diagnosis. Therefore the actual leakage area must be reduced accordingly.

Application conditions:

Initialisation: **If** there are no stored values or a checksum error in adaptation values has occured
then set default value (FAC_TI_ST_AD_DIAGCP_OLD = 1)
else use previously stored values

Formula section:

If LV_FQ_ST_AD_END_1 = 0 → 1

then **if** C_TCO_ST_MIN_ST_AD_DIAGCP ≤ TCO_ST ≤ C_TCO_ST_MAX_ST_AD_DIAGCP

then FAC_TI_ST_AD_DIAGCP = IP_FAC_FQ_ST_AD_DIAGCP

else FAC_TI_ST_AD_DIAGCP = FAC_TI_ST_AD_DIAGCP_OLD

If LV_DC 1 → 0

then FAC_TI_ST_AD_DIAGCP_OLD = FAC_TI_ST_AD_DIAGCP

FAC_TI_ST_AD_DIAGCP_OLD is saved in a non volatile memory!

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_ST_MIN_ST_AD_DIAGCP	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO_ST threshold					
C_TCO_ST_MAX_ST_AD_DIAGCP	1	0...FEH	-48...142.5	0.75	°C
Maximum TCO_ST threshold					
IP_FAC_FQ_ST_AD_DIAGCP	1*8	0...FFH	0...1.992	0.0078	-
LDP_FAC_FQ_ST_AD_IP_FAC_FQ_ST	8	0...FFH	0...1.992	0.0078	-
Weighted fuel quality factor for evaporative system monitoring					

B.10 Evaporative System Monitoring – Vacuum Leak Detection based on Model Parameter Estimation

B.10.1 Summary

The leak detetction function (DIAGCP) permits the detection of leaks in the evaporative emission control system with a diameter of 0.5 mm and more.

The evaporative emission control system is hermetically sealed off from the atmosphere by means of a shutoff valve (SOV) on the charcoal canister. When the evaporative emission control valve opens, the engine generates a vacuum in the fuel tank, and the pressure differential curve in the evaporative emission control system is measured by means of a tank pressure sensor in the evaporative emission control system.

If the vacuum generated within a monitoring period increases above a defined threshold, a leak is detected.

The monitoring algorithm is based on a model of leakage in a tank, which is suitable for parameter estimation. The method is very robust against harmonic and white noise pressure-signal-disturbances and gives insight into the functional dependences of fuel volume, temperature and ambient pressure.

It is sensitive to a pressure-signal offset. Hence, an offset determination is included.

A correction parameter is calculated during an evaporation test phase to consider the increase of pressure due to unsaturated HC vapour.

An information about fuel volume is necessary.

Recurrence : *40 ms for all States with exception of :*
“*Monitoring Conditions (see application incidences)“. 1000 ms*
“*Actuator test for SOV”. 20 ms*

B.10.2 Applied tank pressure sensor characteristic

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
Differential pressure tank - ambient					
DTP_MOD_6	V/O/S	8000...7FFFH	-40.96...40.96	0.00125	hPa
Differential pressure tank – ambient, stored for mode 06 communication					
DTP_RAW	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
Differential pressure tank – raw value; without offset correction					
DTP_MES	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
Differential pressure tank – filtered value; without offset correction					

Input data:

V_DTP	DTP_OFS	LV_END_DIAG_MEC_SOV	

The DTP and DTP_RAW sample rate is **40ms.**

Exception: During normal engine run (STATE_DIAGCP = START, PAS, T_DLY_3, CLD and LV_FUC_MISS_DIAG_CDN = 0) DTP is determined **every 1000 ms**.

Formula section:

The pressure value is derived from a map:

DTP_RAW = IP_DTP_V_DTP

DTP_MES $_{(n)}$ = DTP_MES $_{(n-1)}$ + C_DTP_MES_CRLC * (DTP_RAW $_{(n)}$ - DTP_MES $_{(n-1)}$)

DTP = DTP_MES - DTP_OFS

⇒ At initialisation DTP_OFS is resetted.

⇒ Determination of DTP_OFS (*see chapter "Determination of Sensor Offset*")

Special treatment for diagnostic Test Mode:

The following calculation of output values for Scan Tool SAE 1979 Mode $06 takes place only at the end of a complete diagnostic-cycle (LV_END_DIAG_MEC_SOV = 1)

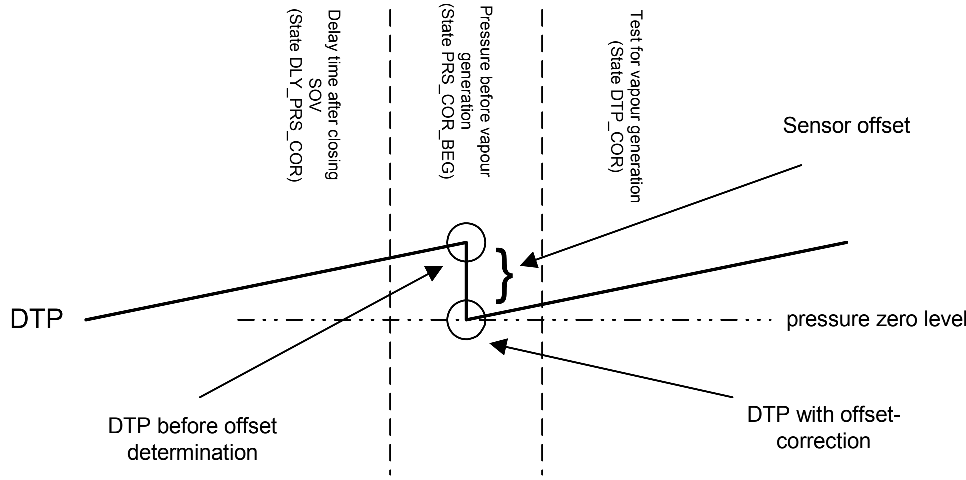
DTP_MOD_6 = DTP

The value DTP_MOD_6 should be initialized with 0 in case of failure memory is cleared.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_DTP_V_DTP	6	0...FFFFH	-40.96...40.96	0.00125	hPa
LDP_V_DTP_IP_DTP_V_DTP	6	0…3FF	0…5V	4.88e-3	V
Characteristic tank-pressure-sensor					
C_DTP_MES_CRLC	1	0...FFH	0...0.996	3.9e-3	-
Dynamic filter factor for the DTP_MES calculation					

B.10.2.1 Overview about ´DTP Sensor Offset Correction´:



B.10.2.2 Determination of DTP_MV

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_MV	V	8000...7FFFH	-40.96...40.96	0.00125	hpa
Pressure mean value					

Input data:

DTP	STATE_DIAGCP	LV_DIAGCPS_ACT	LV_FUC_MISS_DIAG_CDN

Description:

A mean value (DTP_MV) out of 4 pressure values is calculated (every 40ms) during LV_DIAGCPS_ACT = 1 or LV_FUC_MISS_DIAG_CDN = 1

Formula section:

If LV_DIAGCPS_ACT = 1

or LV_FUC_MISS_DIAG_CDN = 1

then $DTP\_MV_i = (DTP_i + DTP_{i-1} + DTP_{i-2} + DTP_{i-3}) / 4$

else *no calculation of DTP_MV*

endif

B.10.3 Determination of “Limited Dynamics DTP” condition

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
DTP_MV_DYW	V	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Mean value of DTP					
DTP_DIF_DYW	V	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Mean value of the differential pressure between tank and ambient for limited dynamic condition					
LV_PRS_LIM_DYN_DIAGCP	V	0...1H	0...1	1	-
Boolean for limited pressure curve dynamics condition existing					

Input data:

DTP	STATE_DIAGCP		

General information:

The “limited dynamics DTP” condition is introduced in order to filter out pressure values resulting from pressure fluctuations in the tank (caused by a door being slammed or by abrupt braking of a slowly rolling vehicle, etc.) that might cause false error signalling.

First, the mean value of the current pressure DTP_i is formed with the value DTP_{i-1} last measured: $DTP\_MV\_DYW_i = (DTP_i + DTP_{I-1}) / 2$

The dynamics window value DTP_DYW depends on the current state:

- C_DTP_DYW_1 — during test for vapour generation (DLY_PRS_COR state and DTP_COR state, except PRS_COR_BEG state)
- C_DTP_DYW_2 — during evacuation phase (DTP_EVAC state) and model parameter estimation (DLY_DIAG state);
- C_DTP_DYW_3 — during diagnosis (DTP_DIAG state)
- C_DTP_DYW_4 — during VAP_CHK state

The limited dynamics condition (LV_PRS_LIM_DYN_DIAGCP = 1) is monitored from “DLY_PRS_COR” state to “DTP_DIAG” state and it continuous to exist, while the pressure DTP stays within the dynamic window DTP_DYW, except during determination of sensor offset (PRS_COR_BEG – state).

If T_DTP_GRD_VAP_CHK
$| DTP\_MV\_DYW_i - DTP_i = DTP\_DIF\_DYW | <$ "DTP DYW value"

then LV_PRS_LIM_DYN_DIAGCP = 1

else LV_PRS_LIM_DYN_DIAGCP = 0; switch to *´Wait Period´*

endif

Depending on the current state the DTP DYW value is C_DTP_DYW_1, C_DTP_DYW_2 or C_DTP_DYW_3

The function switches immediately to the *'Wait Period'* mode if the limited dynamics condition is violated.

Only during *´Test for vapour generation´* the function waits until the time C_T_1_DIAGCP has passed, before the *´Wait Period ´*is activated.

Only during ´Evacuation´ the function switches to the "ERR_CLOSE_CPS" before the *´wait period´* is activated

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_DYW_1	-	0...7FFFH	0...40.96	0.00125	hPa
Dynamic window during test for vapour generation					
C_DTP_DYW_2	-	0...7FFFH	0...40.96	0.00125	hPa
Dynamic window during evacuation phase and model parameter estimation					
C_DTP_DYW_3	-	0...7FFFH	0...40.96	0.00125	hPa
Dynamic window during diagnosis period					
C_DTP_DYW_4	-	0...7FFFH	0...40.96	0.00125	hPa
Dynamic window during VAP_CHK					

B.10.4 Monitoring Algorithm

B.10.4.1 Purging in engine operating state part load (PL)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_PL_DIAGCP	V	0...FFFFH	0...6553.5	0.100	s
Minimum purge time during MAX_PURGE in part-load condition					

Input data:

CL_MMV	REL_FLOW_CPS	STATE_DIAGCP	LV_PL
LV_IS			

Description:

High canister saturation during the vacuum generation period may cause problems with the idle-speed.

Therefore, depending on the result of the determination of the degree of charcoal-canister saturation, purging must take place for a period T_PL_DIAGCP in the engine operating state *´part load´* before the diagnosis function is started. This avoids high canister saturation during evacuation.

In the engine operating states *trailing throttle* (PU), *trailing throttle fuel cut-off* (PUC), *idle* (IS) and *wide-open throttle* (FL), the counter for T_PL_DIAGCP is stopped:

T_PL_DIAGCP is set after each determination of the degree of charcoal-canister saturation (transition of STATE_CP from RAMP_OPEN to MAX_PURGE) or after each interruption of EVAP-Diagnosis.

In the engine operating states *´Part Load´* (PL) **and** (STATE_CP = MAX_PURGE **or** RAMP_CLOSE), the counter for T_PL_DIAGCP is decremented, otherwise it is stopped.

Recurrence: *100ms*

Formula section:

Initialisation:

If Transition of STATE_CP = RAMP_OPEN to STATE_CP = MAX_PURGE

or (interruption of DIAGCP and transition to wait period, i.e. STATE_DIAGCP=T_DLY_3)

then **If** CL_MMV < C_CL_MMV_PL_DIAGCP

and REL_FLOW_CPS > C_REL_FLOW_MIN_CP_DIAGCP

then T_PL_DIAGCP = C_T_1_PL_DIAGCP (*short time*)

else T_PL_DIAGCP = C_T_2_PL_DIAGCP (*long time*)

endif

Decrementation of T_PL_DIAGCP:

If LV_PL = 1

and STATE_CP = MAX_PURGE

or STATE_CP = RAMP_CLOSE

then T_PL_DIAGCP = decremented

else no decrementation of T_PL_DIAGCP

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_REL_FLOW_MIN_CP_DIAGCP	-	0...FFFFH	0...1	0.000015	-
Min. REL_FLOW_CPS threshold for determination of T_PL_DIAGCP					
C_CL_MMV_PL_DIAGCP	-	0...FFFFH	0...8	0.000122	-
CL_MMV threshold for determination of T_PL_DIAGCP					
C_T_1_PL_DIAGCP	-	0...FFFFH	0...6553.5	0.100	s
Minimum purging time with no high canister saturation					
C_T_2_PL_DIAGCP	-	0...FFFFH	0...6553.5	0.100	s
Minimum purging time with high canister saturation					

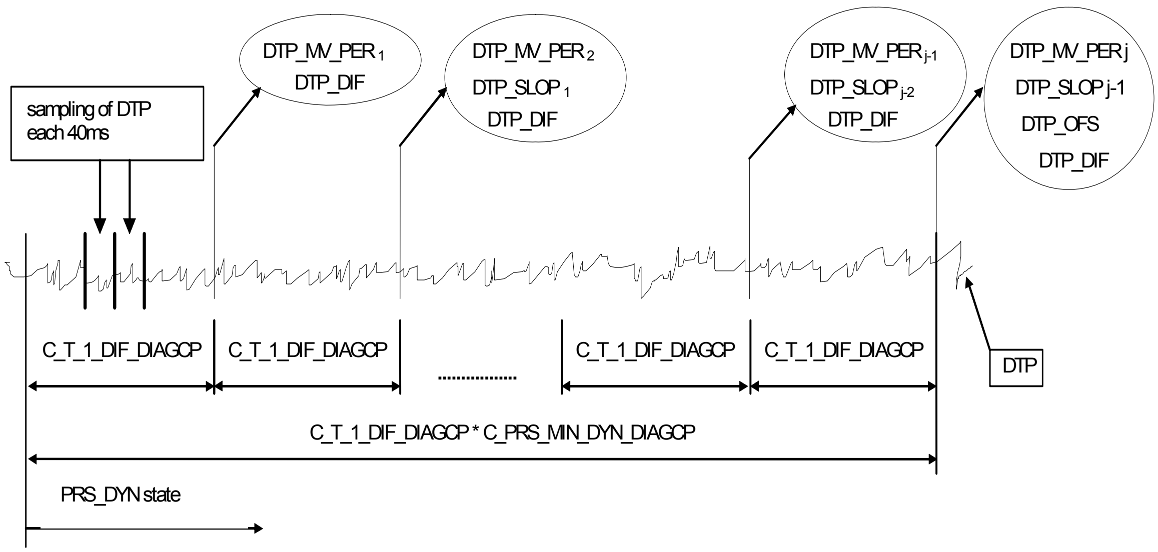
B.10.4.2 Determination of Signal fluctuation and Pressure Slope (PRS_DYN state)

Description:

At the beginning of the diagnostic algorithm the pressure signal fluctuation shall be as small as possible.

The function tests for *´signal fluctuation´* as soon as the monitoring algorithm is released, in the same time the *‘pressure-slope’* determination and the *´plausibility check´* for DTP signal noisy error (refer to chapter A: *DTP Sensor Diagnosis*) are performed.

Overview of the ´Signal Fluctuation Cycle´:



Where j = 0,1,2,3,...C_PRS_MIN_DYN_DIAGCP

Explanation:

⇒ After each C_T_1_DIF_DIAGCP – cycle, the counter for C_PRS_MAX_DYN_DIAGCP is incremented, independently of a violation of the limited dynamics.

⇒ The counter for C_PRS_MIN_DYN_DIAGCP is only incremented if no limited dynamics violation was present during the C_T_1_DIF_DIAGCP- cycle. The counter is reset to zero if the limited dynamics condition was not fulfilled. That means, this counter represents the number of cycles without limited dynamics vialation (´good cycles´).

This C_T_1_DIF_DIAGCP – cycle is repeated until either:

C_PRS_MIN_DYN_DIAGCP periods are reached - in case of no limited dynamic violation

or

C_PRS_MAX_DYN_DIAGCP periods are reached - in case of limited dynamic violation and C_PRS_MIN_DYN_DIAGCP could not be reached

⇒ In fact, the minimum physical value of C_PRS_MAX_DYN_DIAGCP must be the calibrated value of C_PRS_MIN_DYN_DIAGCP. If it is not the case, the watchdog made by C_PRS_MAX_DYN_DIAGCP * 0.04 [s] will be always reached before the determination of DTP_OFS.

B.10.4.2.1 Determination of signal fluctuation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_DIF	V/O	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Maximum DTP range during C_T_1_DIF_DIAGCP					
DTP_DIF_MOD_6	V/O/S	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Maximum DTP range during C_T_1_DIF_DIAGCP stored for mode 06-communication					
DTP_MAX	V	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Max DTP - value during C_T_1_DIF_DIAGCP					
DTP_MIN	V	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Min DTP - value during C_T_1_DIF_DIAGCP					
LV_DTP_NOISE	V/O	0...1H	0...1	1	-
Boolean for Noisy DTP signal					

Input data:

DTP			

Description:

The minimum and maximum pressure (DTP_MIN and DTP_MAX) are determined during the period C_T_1_DIF_DIAGCP * C_PRS_MIN_DYN_DIAGCP. If the pressure difference DTP_DIF = DTP_MAX - DTP_MIN is smaller than C_DTP_DIF_MAX the pressure signal fluctuation is low and the system switches to *´CLOSE_CPS´* state.

DTP_DIF is determined again, if the pressure difference between DTP_MAX and DTP_MIN is greater than C_DTP_DIF_MAX. This cycle is repeated until the condition is met.

The algorithm is stepped to the "*Wait Period*" if C_PRS_MAX_DYN_DIAGCP periods of C_T_1_DIF_DIAGCP is reached. In this case the signal noisy of the "Differential Tank Pressure Sensor" is too high (LV_DTP_NOISE = 1).

Formula section:

If DTP_DIF = DTP_MAX - DTP_MIN ≤ C_DTP_DIF_MAX
for each C_T_1_DIF_DIAGCP
during C_PRS_MIN_DYN_DIAGCP *subsequent time periods*

then *step to ´CLOSE_CPS´ state*

else repeat until the conditions are met

If C_PRS_MAX_DYN_DIAGCP periods of C_T_1_DIF_DIAGCP are reached

then LV_DTP_NOISE = 1 ;step to *´Wait Period´*

endif

⇒ LV_DTP_NOISE is resetted at the end of *´Wait Period´* and at the beginning of a new driving cycle

⇒ The measurement of signal fluctuation is repeated by C_PRS_MIN_DYN_DIAGCP

⇒ The *´Conditions for DIAGCP´* must be met continuously

⇒ During the PRS_DYN state the canister purge function is in normal operation and the CPPWM is calculated according to this

Special treatment for Diagnostic Test Mode:

DTP_DIF is stored as DTP_DIF_MOD_6, if C_PRS_MAX_DYN_DIAGCP periods are reached or STATE_DIAGCP changes 1 → 2 or LV_DTP_NOISE transition 0->1.

The value DTP_DIF_MOD_6 should be initialized with 0 in case of failure memory is cleared.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_DIF_MAX	-	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Maximum allowed pressure-signal fluctuation to start DIAGCP					
C_T_1_DIF_DIAGCP	-	0...FFH	0...25.5	0.1	s
Time-period to determine signal fluctuation and pressure slope just before vapour generation					
C_PRS_MIN_DYN_DIAGCP	-	2…FFH	2…255	1	-
Minimum number of C_T_1_DIF_DIAGCP periods					
C_PRS_MAX_DYN_DIAGCP	-	2…FFH	2…255	1	-
Maximum number of C_T_1_DIF_DIAGCP periods					

B.10.4.2.2 Determination of pressure slope

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_MV_PER	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure mean-value during C_T_1_DIF_DIAGCP					
DTP_SLOP_END	V/O	8000...7FFFH	-409,6...409,587	12,5E-3	hPa/s
Pressure slope					
LV_DTP_SLOP_END	V/O	0...1H	0...1	-	-
Flag for stable pressure slope					

Input data:

DTP			

Description:

The pressure slope is determined in [hPa/sec]. For this purpose the pressure mean-value during the period C_T_1_DIF_DIAGCP is calculated.

The sampling rate is **40 ms**.

If DTP_MV_PER $_j$ and DTP_MV_PER $_{j-1}$ are valid (after 2 measurements cycles) then the pressure slope determination is possible:

Formula section:

Pressure mean-value:

DTP_MV_PER $_j$ = [sum (DTP $_i$)] / (C_T_1_DIF_DIAGCP / 0.04[s])

sum (DTP $_i$) := Sum of all pressure values during C_T_1_DIF_DIAGCP

Pressure slope:

DTP_SLOP_END j-1 = (DTP_MV_PER j - DTP_MV_PER j-1)
/ (C_T_1_DIF_DIAGCP - 0.1[s])

Where i and j are two different counters :

i : every **40** ms, DTP sampling rate.

j : number of C_T_1_DIF_DIAGCP times during ´*PRS_DYN*´ state.

Formula section:

Detection:

If | DTP_SLOP_END j | < C_DTP_SLOP_MAX

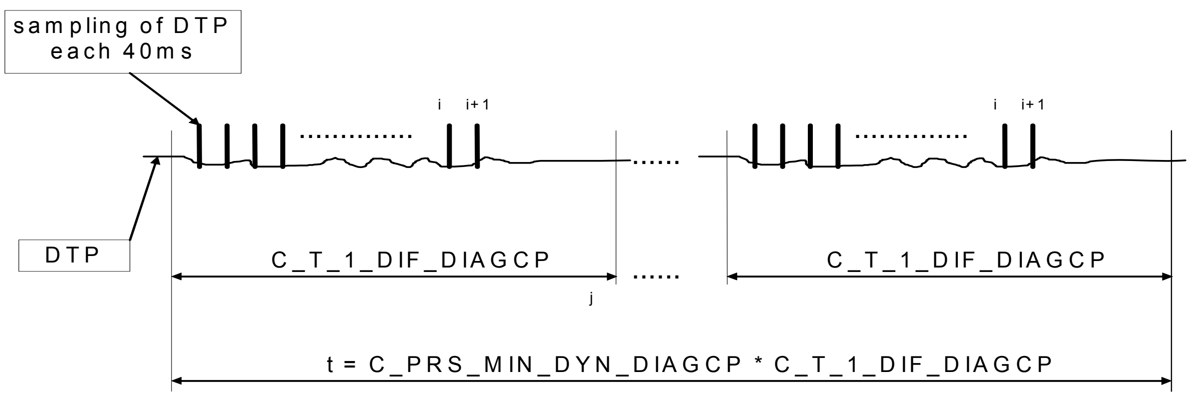
for each C_T_1_DIF_DIAGCP
during C_PRS_MIN_DYN_DIAGCP *subsequent time-periods*

then LV_DTP_SLOP_END = 1 (pressure-slope is stable); switch to *´CLOSE_CPS´* state

else LV_DTP_SLOP_END = 0; step to *´Wait Period´*

endif

⇒ LV_DTP_SLOP_END = 1 at the beginning of each C_T_1_DIF_DIAGCP



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_SLOP_MAX	-	0000...7FFFH	0...409.5875	0.0125	hPa/s
Maximum pressure slope threshold used to determine sensor offset					

B.10.4.3 CPS closing with a ramp (CLOSE_CPS state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAGCPS_ACT	V/O	0...1H	0...1	1	-
Flag indicating CP not active and DIAGCP active (1), or the opposite (0)					

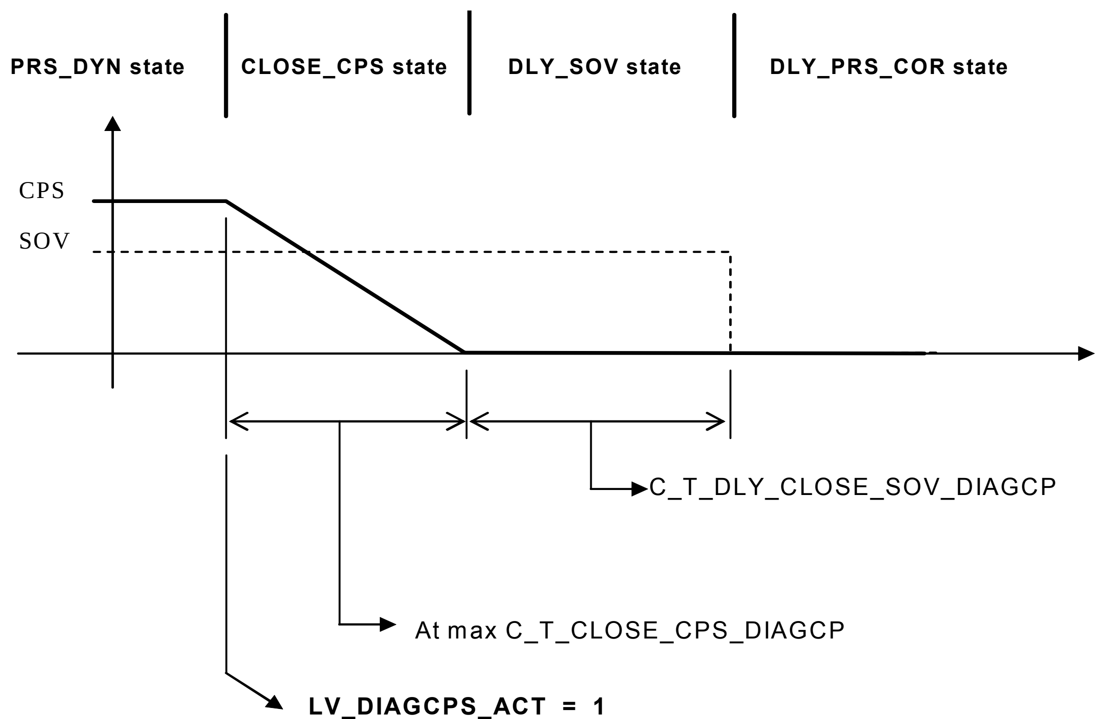
Input data:

FLOW_SP_CPS	FLOW_SP_CPS_EXT	STATE_DIAGCP	

Description:

The transition from ´*PRS_DYN*´ state to ´*CLOSE_CPS*´ state is only possible if no noisy DTP signal is present (LV_DTP_NOISE=0) and the pressure slope is stable (LV_DTP_SLOP=1).

The evaporative emission control is switched to the ´evaporative emission control valve closed mode´ (NO_PURGE) and disabled during diagnosis. Therefore FLOW_SP_CPS_EXT is set to FLOW_SP_CPS and then the CPS is closing with a ramp given by C_FLOW_CPS_SP_CLOSE_DIAGCP. If the CPS is not totally closed after C_T_CLOSE_CPS_DIAGCP then the CPS must be closed suddenly. The lambda adaptation is also disabled.



Formula section:

If STATE_DIAGCP = CPS_CLOSE
then LV_DIAGCPS_ACT = 1 **and**
FLOW_SP_CPS_EXT = FLOW_SP_CPS

The CPS-closing is performed by:
$FLOW\_SP\_CPS\_EXT_{n}$ = $FLOW\_SP\_CPS\_EXT_{n-1}$ - C_FLOW_CPS_SP_CLOSE_DIAGCP

If C_T_CLOSE_CPS_DIAGCP is reached
then $FLOW\_SP\_CPS\_EXT_{n} = 0$

If the CPS is not totally closed after C_T_CLOSE_CPS_DIAGCP then the CPS must be closed suddenly.

If $FLOW\_SP\_CPS\_EXT_{n} = 0$
then STATE_DIAGCP = DLY_SOV

In VAP_CHK the CPS is also controlled by the Monitoring Function

If STATE_DIAGCP = VAP_CHK
then LV_DIAGCPS_ACT = 1

Note:

⇒ FAC_LAM_MV_MMV[1] correction during LV_DIAGCPS_ACT = 1 through calculation of the additive fuel flow correction MFF_ADD_DIAGCP *(see chapter "Injection Time Correction")*

⇒ Disabling of the lambda adaptation: see ´*Auxiliary Functions*´, chapter "Lambda Adaption" and "Evaporative Emission Control"

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_CLOSE_CPS_DIAGCP	-	0...FFH	0...25.5	0.100	s
Time for CPS closing ramp after transition from purge function to evap. monitoring					
C_FLOW_CPS_SP_CLOSE_DIAGCP	-	0...FFFFH	0...7.999	0.000122	kg/h
Closing ramp of the CPS at the beginning of the evap. monitoring					

B.10.4.4 SOV closing delay State (DLY_SOV state)

Description:

The charcoal canister shutoff valve is closed after waiting C_T_DLY_CLOSE_SOV_DIAGCP. The status of the shut-off-valve is represented by LV_SOV_REQ.

Formula section:

If	C_T_DLY_CLOSE_SOV_DIAGCP has passed	
then	LV_SOV_REQ = 1	(Request to close SOV)
else	LV_SOV_REQ = 0	(Request to open SOV)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_CLOSE_SOV_DIAGCP	-	0...FFH	0...25.5	0.100	s
Delay time between closing the CPS and closing the SOV					

B.10.4.5 Delay time after closing SOV (DLY_PRS_COR state)

Input data:

DTP			

Description:

If the shutoff valve is closed, a delay time C_T_DLY_COR_BEG_DIAGCP is started. After this delay time, the algorithm switches to PRS_COR_BEG (determination of the sensor offset).

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_COR_BEG_DIAGCP	-	0...FFH	0...25.5	0.100	s
Delay time before determination of start pressure					

B.10.4.6 Determination of sensor offset (PRS_COR_BEG state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_OFS	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure sensor offset					

Input data:

DTP_MV			

Description:

To get a reliable monitoring result, it's necessary to determine the actual sensor-offset. During the whole monitoring algorithm the pressure-signal will be corrected by the sensor-offset for all necessary calculations.
Because of the DTP-correction

The limited pressure curve dynamics is not checked in this state.

Formula Section:

⇒ Determination of Sensor Offset

$$DTP\_OFS_{n} = DTP\_OFS_{n-1} + DTP\_MV$$

DTP_OFS is valid for the whole engine run or until the next determination of this offset is performed. DTP_OFS is resetted at next engine run.

B.10.4.7 Test for vapour generation (DTP_COR state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_COR	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
DTP corrected by C_DTP_BAS_COR in the vapour generation phase					
DTP_DIF_COR	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
DTP difference during the vapour generation phase					
DTP_DIF_COR_MOD_6	V/O/S	8000...7FFFH	-40.96...40.96	0.00125	hPa
DTP difference during the vapour generation phase stored for mode 06-communication					
FAC_DIAGCP_MDL_ESTIM	V	0000...FFFFH	0...0.467	7.126e-6	-
Estimated DIAGCP leakage-parameter due to vapour generation in DTP_DIAG					
TAM_DIAGCP	V/O	0...FEH	-48...142.5	0.75	°C
Ambient temperature at the moment of EVAP monitoring performance					

Input data:

DTP	FAC_DIAGCP_MDL_BAS	C_DTP_DIF_CPS_DIAGCP	TAM

Description:

In order not to take in account DTP jittering, DTP is kept to 0 if it is increasing by less than C_DTP_HYS_POS or decreasing by less than C_DTP_HYS_NEG.

If DTP i >= C_DTP_HYS_NEG

and DTP i <= C_DTP_HYS_POS

then DTP i = 0

endif

During *'Test for vapour generation'* the actual pressure value has to be corrected by C_DTP_BAS_COR (virtual pressure shift to negative direction). This mathematical offset is used to have the same reference as the calculation of the FAC_DIAGCP_MDL_END.

DTP_COR i = DTP i - C_DTP_BAS_COR

During the time-period C_T_1_DIAGCP (DTP_COR state) the '*Model Parameter Estimation'* for *'Vapour generation'* is active (see chapter "*Model Parameter Estimation*").

The result FAC_DIAGCP_MDL_BAS is the correction value for the 'Diagnosis' period due to vapour generation.

If FAC_DIAGCP_MDL_BAS > C_FAC_DIAGCP_MDL_BAS_MAX

then the diagnosis is aborted and the system switches to "wait period".
At the same time a new NORMAL_PURGE State of the CP-Function is initiated (LV_NORM_PURGE_END_1 = 0, see chapter Cooperation with the caniter purge function

If LV_NORM_PURGE_END_1 is set to 1 by Canister purge function (MAX_PURGE function finished), the diagnosis algortihm starts from the beginning.

DTP difference during the vapour generation phase:

During DTP_COR state, the increase or decrease of DTP is calculated as follows:

DTP_DIF_COR = DTP_MV

- If during *test of vapour generation* the pressure drops more than C_DTP_DIF_CPS_DIAGCP, the debouncing of the error 'evaporative emission control valve jammed in open position' is started (see chapter "*Mechanical evaporative emission control valve error*").
- If no errors are detected, the function switches to the 'evacuation' state.

Special treatment for Diagnostic Test Mode:

If readyness of symptom "MEC_CPS" is detected or MEC_CPS error is detected, DTP_DIF_COR is stored as DTP_DIF_COR_MOD_6. (STATE_DIAGCP changes 6 $\rightarrow$ 7 or LV_ERR_MEC_CPS transition 0 -> 1, the last value of DTP_DIF_COR is stored as DTP_DIF_COR_MOD_6)

The value DTP_DIF_COR_MOD_6 should be initialized with 0 in case of failure memory is cleared.

Correction factor ´ FAC_DIAGCP_MDL_ESTIM´:

When FAC_DIAGCP_MDL_BAS is obtained (that represents the slope of the DTP square root), a correction factor is applied by using the map IP_FAC_DIAGCP_MDL_ESTIM, to estimate the slope of the DTP square root due to the vapour generation we have in the DTP_DIAG state:

FAC_DIAGCP_MDL_ESTIM = IP_FAC_DIAGCP_MDL_ESTIM

TAM used for EVAP monitoring:

At the moment of FAC_DIAGCP_MDL_ESTIM determination during DTP_COR, need to store TAM as TAM_DIAGCP which is used for diagnosis:

TAM_DIAGCP = TAM

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_1_DIAGCP	1	0...FFH	0...25.5	0.1	S
Duration of test for vapour generation					
C_DTP_BAS_COR	1	8000...7FFFH	-40.96...40.96	0.00125	Hpa
DTP correction value during vapour generation period					
C_FAC_DIAGCP_MDL_BAS_MAX	1	0000...FFFFh	0...0.467	7.126e-6	-
Maximum allowed leakage parameter during vapour generation period					
C_DTP_HYS_POS	1	0000...7FFFH	0...40.96	0.00125	Hpa
Hysteresis not to take in account small DTP variation during vapor generation in case of positive values					
C_DTP_HYS_NEG	1	8000...7FFFH	-40.96...40.96	0.00125	Hpa
Hysteresis not to take in account small DTP variation during vapor generation in case of negative values					
IP_FAC_DIAGCP_MDL_ESTIM	4*6	0000...FFFFH	0...0.467	7.126e-6	-
LDP_FAC_DIAGCP_MDL_BAS	4	0000...FFFFH	0...0.467	7.126e-6	
LDPM_TAM	6	00..FEH	-48...142.5	0.75	°C
Vapour generation correction factor					

B.10.4.8 Pressure before evacuation (PRS_EVAC_BEG state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_BEG_DIAGCP	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
Mean DTP value before the evacuation phase					
FTL_MMV_DIAGCP	V/O	0...FFFFH	0...100	0.0015	%
Fuel mean volume from FTL sensor at moment of EVAP monitoring performance					

Input data:

DTP_MV	STATE_DIAGCP	ERR_SYM_MEC_CPS	FLOW_SP_CPS_EXT
C_FLOW_CPS_SP_INC_DIAGCP			

Description:

The system switches to the ´*PRS_EVAC_BEG*´ state if the period C_T_1_DIAGCP is reached. The pressure mean value (DTP_MV) out of the first 4 pressure values – is stored as ´*DTP_BEG_DIAGCP*´, the pressure before evacuation.

Before the pressure mean value DTP_BEG_DIAGCP is calculated, the evacuation is started, i.e. the CPS valve is opened at least one increment.

In case of DTP decrease during Vapour Generation phase due to environmental condition, the diagnosis switches to the ´Wait Period´. The diagnosis will be performed when monitoring condition will be fulfilled again.

Formula Section:

If C_T_1_DIAGCP is reached

then STATE_DIAGCP = PRS_EVAC_BEG

and DTP_BEG_DIAGCP = DTP_MV

and FTL_MMV_DIAGCP = FTL_MMV (at the moment of EVAP monitoring)

end

If DTP_BEG_DIAGCP ≤ C_DTP_HYS_NEG

or ERR_SYM_MEC_CPS = SYM_0

then the diagnosis is aborted and the system switches to the "*Wait Period*"

end

B.10.4.9 Evacuation phase (DTP_EVAC state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FLOW_SP_CPS_EXT	V/O	0...FFFFH	0...8	0.000122	kg/h
Setpoint for MAF_CPS at evacuation					
DTP_DIF_ACT	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure difference during evacuation					
DTP_DIF_ACT_MOD_6	V/O/S	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure difference during evacuation stored for mode 06-communication					
LV_DTP_EVAC_INC	V/O	0...1H	0...1	1	-
Boolean for fast large leak error is detected					
LV_T_DLY_DTP_EVAC_INC	V/O	0...1H	0...1	1	-
Boolean for time delay setting					
DTP_MV_REF	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
Reference tank pressure for tank cap open detection					
T_DLY_DTP_EVAC_INC	V/O	0...FFH	0...25.5	0.1	S
Time delay for large leak detection					
LV_DTP_EVAC_CPS_END	V/O	0...1H	0...1	1	-
Boolean to show the end of CPS fully opening					
DTP_DIF_EVAC	V/O	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure difference before reaching DTP evacuation target					

Input data:

LAM_MV_i	STATE_DIAGCP	FAC_LAM_MV_MMV[NC_CBK_EX_NR]	DTP_BEG_DIAGCP
LV_LAM_LIM_MIN_i	DTP	FAC_LAM_DIAGCP	DTP_MV
FLOW_SP_CPS_VAP_CHK	LV_DTP_SLOP_CLC_END	FTL_MMV	N_DIF

B.10.4.9.1 Opening and closing of the CPS:

To open and close the canister purge valve for controlling the evacuation period, a value FLOW_SP_CPS_EXT is calculated. It is a setpoint for the desired mass air flow through the CPS performed by the Evaporative emission control function.

At the beginning of evacuation FLOW_SP_CPS_EXT is incremented to C_FLOW_CPS_SP_DIAGCP using an increment C_FLOW_CPS_SP_INC_DIAGCP.

At the end of DTP slop calculation, FLOW_SP_CPS_EXT is updated accordring to DTP differences between DTP_DIF_ACT & C_DTP_DIF_DIAGCP.If STATE_DIAGCP = VAP_CHK then FLOW_SP_CPS_EXT = FLOW_SP_CPS_VAP_CHK

Initialization :

Initialization at the beginning of DTP_EVAC

LV_DTP_EVAC_CPS_END = 0

FLOW_SP_CPS_EXT = 0

Formula section:

If STATE_DIAGCP = 8h (DTP_EVAC)

then **if** FLOW_SP_CPS_EXT < C_FLOW_CPS_SP_DIAGCP

and LV_DTP_EVAC_CPS_END = 0

then FLOW_SP_CPS_EXT is incremented with C_FLOW_CPS_SP_INC_DIAGCP until C_FLOW_CPS_SP_DIAGCP

endif

if FLOW_SP_CPS_EXT = C_FLOW_CPS_SP_DIAGCP

then LV_DTP_EVAC_CPS_END = 1 (kept until next initialization)

endif

if LV_DTP_EVAC_CPS_END = 1 (after CPS once fully opening)

and DTP_MV < C_DTP_DIF_MIN_SLOP

and LV_DTP_SLOP_CLC_END = 1 (after double slop calculation)

then DTP_DIF_EVAC = DTP_DIF_ACT - C_DTP_DIF_DIAGCP

FLOW_SP_CPS_EXT = IP_FLOW_CPS_SP_DIAGCP (FTL_MMV, DTP_DIF_EVAC)

endif

else if STATE_DIAGCP = Fh (VAP_CHK)

then FLOW_SP_CPS_EXT = FLOW_SP_CPS_VAP_CHK

else FLOW_SP_CPS_EXT = 0

endif

B.10.4.9.2 “Regular” - Closing of the CPS (CLOSE_CPS state):

In the following cases FLOW_SP_CPS_EXT is set to 0 without change limitation (further actions see “Injection time correction”)

End of evacuation period

If no impermissible deviations from FAC_LAM_MV_MMV[1] are registered, the system waits until the pressure has dropped by C_DTP_DIF_DIAGCP. Then FLOW_SP_CPS_EXT is set to 0 (suddenly), before advancing to “Release of Parameter Estimation (DLY_DIAG-state)”.

Function interruption:

If the bits LV_STATE_CDN_DIAGCP (activation conditions violated) or LV_PRS_LIM_DYN_DIAGCP (limited dynamics violated) changing from 1 to 0 during evacuation, FLOW_SP_CPS_EXT is set to 0 (suddenly) and the *´Wait Period´* is activated.

B.10.4.9.3 "Irregular" - Closing of the CPS (ERR_CLOSE_CPS state)

In the following cases, FLOW_SP_CPS_EXT is reduced to 0 via ramp with the slope C_FLOW_CPS_SP_DEC_DIAGCP (STATE_DIAGCP = ERR_CLOSE_CPS)

Lambda deviation at evacuation

The lambda controller value (FAC_LAM_MV_MMV[1]) is shifted for more than C_FAC_LAM_MAX_DIAGCP (for leaning the mixture), respectively for more than C_FAC_LAM_MIN_DIAGCP (enrich the mixture) from the lambda control mean value at the beginning of evacuation (FAC_LAM_DIAGCP),

```
If  {   FAC_LAM_DIAGCP – FAC_LAM_MV_MMV[1] >= C_FAC_LAM_MAX_DIAGCP
    or  FAC_LAM_DIAGCP – FAC_LAM_MV_MMV[1] <= C_FAC_LAM_MIN_DIAGCP
        // Conditions for breaking off
    or  LV_LAM_LIM_MIN_i = 1              }
        // the lambda controller is limited in lean direction
and {   DTP_DIF_ACT <= C_DTP_DIF_MIN_DIAGCP
        // missing tank cap not suspected, see next paragraph
    or  N_DIF > C_N_DIF_MAX_DIAGCP   }
        // RPM oscillations are limited for engine stability
```

the diagnosis is aborted (´ERR_CLOSE_CPS´ state). That means FLOW_SP_CPS_EXT is reduced to 0 via a ramp with the slope C_FLOW_CPS_SP_DEC_DIAGCP. When FLOW_SP_CPS_EXT reaches 0 then LV_DIAGCPS_ACT is set to 0 and the system switches to "Wait period".

As soon as the diagnosis is aborted the SOV is opened (for SOV-diagnosis reasons):

If STATE_DIAGCP = ERR_CLOSE_CPS

then LV_SOV_REQ = 0 (Request to open SOV)

The system waits for the next determination of the degree of charcaol-canister saturation. (see chapter 1.5.12 "Cooperation with the canister purge function")

Tank cap missing

If the pressure does not drop by at least C_DTP_DIF_MIN_DIAGCP (STATE_DIAGCP=ERR_CLOSE_CPS) during the maximum possible time period C_T_MAX_CPS, *'tank cap missing'* is detected. In that case FLOW_SP_CPS_EXT is reduced to 0 via a ramp with the slope C_FLOW_CPS_SP_DEC_DIAGCP. *(see chapter "Tank cap missing during leak detection algorithm")*

Large leak

If the pressure drops by more then C_DTP_DIF_DIAGCP during the above period, a 'large leak´ is detected and registered, and the diagnosis is disabled for this driving cycle. In that case FLOW_SP_CPS_EXT is reduced to 0 via a ramp with the slope C_FLOW_CPS_SP_DEC_DIAGCP. (see chapter "*Large leak*")

If the pressure increases more than the previous pressure, timer (=C_T_DTP_MV_EVAC_INC) will be set and the previous pressure is memorised to detect large leak early. If not, evaporative system monitoring can be stopped due to too much lambda deviation at hot condition.

Formula section:

DTP_DIF_ACT = DTP_MV - DTP_BEG_DIAGCP

If DTP_DIF_ACT ≤ C_DTP_DIF_DIAGCP

then switch to 'DLY_DIAG' state (see above: "*End of evacuation period*")

elseif LV_DTP_EVAC_INC = 1

then error detection ´large leak´ (see above: "*Large leak*")

else

if C_T_MAX_CPS has passed

then

if DTP_DIF_ACT >= C_DTP_DIF_MIN_DIAGCP

then error detection 'tank cap missing' (see above: "*Tank cap missing during leak detection algorithm*")

else error detection ´large leak´ (see above: "*Large leak*")

endif

else continue the state ´DTP_EVAC´

endif

endif

Formular section – Fast large leak error detection :

Activation :

If STATE_DIAGCP = 8 (DTP_EVAC)

and FLOW_SP_CPS_EXT = C_FLOW_CPS_SP_DIAGCP

and LV_DTP_EVAC_INC = 0

then

if LV_T_DLY_DTP_EVAC_INC = 0

then

if DTP_DIF_ACT <= C_DTP_MV_EVAC_DIF

then

T_DLY_DTP_EVAC_INC = C_T_DTP_MV_EVAC_INC

LV_T_DLY_DTP_EVAC_INC = 1

DTP_MV_REF = DTP_MV_{N-1}

endif

endif

if T_DLY_DTP_EVAC_INC is elapsed

then **if** │DTP_MV – DTP_MV_REF│ <= C_DTP_MV_EVAC_INC

then

LV_DTP_EVAC_INC = 1

endif

endif

else

LV_T_DLY_DTP_EVAC_INC = 0

T_DLY_DTP_EVAC_INC = 0

DTP_MV_REF = 0

endif

Special treatment for Diagnostic Test Mode:

If readyness of symptom ´LARGE_LEAK´ or ´FUC_MISS´ is detected, DTP_DIF_ACT is stored as DTP_DIF_ACT_MOD_6

The value DTP_DIF_ACT_MOD_6 should be initialized with 0 in case of failure memory is cleared.

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_DTP_DIF_DIAGCP	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Required pressure-drop to finish evacuation period					
C_DTP_DIF_MIN_DIAGCP	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Threshold for detection of missing tank cap					
C_FAC_LAM_MAX_DIAGCP	1	8000...7FFFH	-50...50	0.0015	%
Maximum allowed lambda integrator deviation during evacuation					
C_FLOW_CPS_SP_DIAGCP	1	0...FFFFH	0...7.999	0.000122	kg/h
Setpoint for MAF_CPS of CP-function during evap.-monitoring					
C_FLOW_CPS_SP_INC_DIAGCP	1	0...FFFFH	0...7.999	0.000122	kg/h
Increment for MAF_CPS setpoint of CP-function during evap.-monitoring					
C_FLOW_CPS_SP_DEC_DIAGCP	1	0...FFFFH	0...7.999	0.000122	kg/h
Decrement for MAF_CPS setpoint of CP-function during evap.-monitoring					
C_T_MAX_CPS	1	0...FFH	0...25.5	0.1	s
Maximum time period for evacuation					
C_FAC_LAM_MIN_DIAGCP	1	8000...7FFFH	-50...50	0.0015	%
Minimum allowed lambda integrator deviation during evacuation					
C_DTP_MV_EVAC_DIF	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Minimum pressure difference to start fast large leak detection					
C_T_DTP_MV_EVAC_INC	1	1...FFH	0.1...25.5	0.1	s
Maximum time period to detect fast large leak					
C_DTP_MV_EVAC_INC	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Maximum pressure difference to detect fast large leak					
C_DTP_DIF_MIN_SLOP	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Required pressure-drop to start the CPS slow closing during evacuation					
IP_FLOW_CPS_SP_DIAGCP	4*8	0...FFFFH	0...7.999	0.000122	kg/h
LDP_DTP_DIF_EVAC	4	0...FFFFH	-40.96...40.96	1.25E-3	hPa
LDP_FTL_MMV	8	0...FFFFH	0...100	0.0015	%
Setpoint for MAF_CPS of CP-function to reach DTP EVAC target smoothly during evacuation					
C_N_DIF_MAX_DIAGCP	1	8000...7FFFH	-32768...32767	1	rpm
RPM deviation threshold for engine stability during evacuation					

B.10.4.9.4 DTP slope estimation during evacuation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DTP_SLOP	V/O	0…1H	0…1	-	-
Boolean for stable slope (0) or unstable (1) during evac					
DTP_MMV	V	8000…7FFFH	-40.96…40.96	0.00125	hPa
DTP moving mean value during evacuation					
DTP_SLOP	V	8000…7FFFH	-409.6…409.6	0.0125	hPa/s
DTP slope value during the evacuation					
DTP_SLOP_MMV	V	8000…7FFFH	-409.6…409.6	0.0125	hPa/s
DTP slope moving mean value during evacuation					
DTP_SLOP_MMV_MAX	V	8000…7FFFH	-409.6…409.6	0.0125	hPa/s
Maximum DTP slope moving mean value stored during evacuation					
DTP_SLOP_MMV_MIN	V	8000…7FFFH	-409.6…409.6	0.0125	hPa/s
Minimum DTP slope moving mean value stored during evacuation					
DTP_SLOP_MMV_DIF	V	0…FFFFH	0…819.2	0.0125	hPa/s
Variation of DTP slope moving mean value during evacuation					
LV_DTP_SLOP_CLC	V/O	0…1H	0…1	-	-
Boolean for slope calculation during evac					
LV_DTP_SLOP_CLC_END	V/O	0…1H	0…1	-	-
Boolean for slope calculation ending during evac					
DTP_SLOP_MMV_DIF_MMV	V	0…FFFFH	0…819.2	0.0125	hPa/s
Mean value variation of DTP slope to determine DTP slope calculation ending during evac					

Input data:

DTP	STATE_DIAGCP	TCO_ST	T_AST
C_T_DTP_GRD_DIAGCP			

General information:

High ambient temperatures have a big influence on the DTP behavior as well as on the components.

Above a certain temperature DTP behavior becomes unlinear and there is a risk of leak detection even with a tight system. Therefore we analyze the DTP behavior during evacuation to have a good picture of the temperature. The diagnosis is aborted if a 0.5 mm leak is detected when DTP slope variation during evacuation is greater than a threshold (see also chapter “Evaporative emission control diagnosis error - 0.5 mm leak”).

Formula section:

a.) During the complete 'Evacuation period' the following check is performed every 40 msec:

If STATE_DIAGCP = DTP_EVAC

then

$DTP\_MMV_{(n+1)} = DTP\_MMV_{(n)} + C\_DTP\_MMV\_CRLC * (DTP_{(n+1)} - DTP\_MMV_{(n)})$

$DTP\_SLOP_{(n+1)} = (DTP\_MMV_{(n+1)} - DTP\_MMV_{(n)}) / 0.04 [s]$

$DTP\_SLOP\_MMV_{(n+1)} = DTP\_SLOP\_MMV_{(n)} + C\_DTP\_SLOP\_MMV\_CRLC * (DTP\_SLOP_{(n+1)} - DTP\_SLOP\_MMV_{(n)})$

Initialisation:

$DTP\_MMV_{(n)} = DTP_{(n)}$

$DTP\_SLOP\_MMV_{(n)} = 0$

$DTP\_SLOP\_MMV\_DIF\_MMV_{(n)} = 0$

LV_DTP_SLOP = 0

LV_DTP_SLOP_CLC = 0

LV_DTP_SLOP_CLC_END = 0

b.) C_T_MAX_FAC_SLOP_MMV seconds after beginning of evacuation the following calculations are done:

If $DTP\_SLOP\_MMV_{(n)} > DTP\_SLOP\_MMV\_MAX$
then $DTP\_SLOP\_MMV\_MAX = DTP\_SLOP\_MMV_{(n)}$

If $DTP\_SLOP\_MMV_{(n)} < DTP\_SLOP\_MMV\_MIN$
then $DTP\_SLOP\_MMV\_MIN = DTP\_SLOP\_MMV_{(n)}$

$DTP\_SLOP\_MMV\_DIF = |DTP\_SLOP\_MMV\_MAX - DTP\_SLOP\_MMV\_MIN|$

$DTP\_SLOP\_MMV\_DIF\_MMV_{(n)} = DTP\_SLOP\_MMV\_DIF\_MMV_{(n-1)}$

$+ C\_DTP\_SLOP\_DIF\_MMV\_CRLC * (DTP\_SLOP\_MMV\_DIF_{(n)} - DTP\_SLOP\_MMV\_DIF\_MMV_{(n-1)})$

If $| DTP\_SLOP\_MMV\_DIF\_MMV_{(n-1)} - DTP\_SLOP\_MMV\_DIF\_MMV_{(n)} | < C\_DTP\_SLOP\_DIF\_MMV\_DIF$

then LV_DTP_SLOP_CLC_END = 1

else LV_DTP_SLOP_CLC_END = 0

endif

If *DTP_SLOP_MMV_DIF after C_T_DTP_GRD_DIAGCP is elapsed*

then LV_DTP_SLOP_CLC = 1

else LV_DTP_SLOP_CLC = 0 *(0.5mm evacuation error is valid)*

endif

c.) Stable slop is determined at DLY_DIAG state :

If STATE_DIAGCP = 09H (DLY_DIAG state)

and { (DTP_SLOP_MMV_DIF >= IP_DTP_MAX_DIF_SLOP

and EOL_STATE_DIAGCP = EOL_PAS)

or

(DTP_SLOP_MMV_DIF >= IP_DTP_MAX_DIF_SLOP_EOL

and EOL_STATE_DIAGCP <> EOL_PAS) }

// EOL_STATE_DIAGCP <> EOL_PAS means EVAP monitoring was done at EOL test,

and LV_DTP_SLOP_CLC = 1

or {T_AST < C_T_AST_MIN_DIAGCP_1

and C_TCO_ST_MIN_DIAGCP_1 < TCO_ST < C_TCO_ST_MAX_DIAGCP_1}

then LV_DTP_SLOP = 1

else LV_DTP_SLOP = 0 *(0.5mm evacuation error is valid)*

endif

Remark: If IP_DTP_MAX_DIF_SLOP = 0, the symptom LEAK_SMALL_05MM is not available in the whole driving cycle.

In addition the availability of the symptom LEAK_SMALL_05MM can be shifted an arbitrarily time after start (dependently of TCO_ST).

LV_DTP_SLOP = 1 is kept after first transition 0→1

LV_DTP_SLOP_CLC = 1 is kept after first transition 0→1

Initialisation:

a) C_T_MAX_FAC_SLOP_MMV seconds after beginning of evacuation:

$DTP\_SLOP\_MMV\_MAX_{(n)} = DTP\_SLOP\_MMV_{(n)}$

$DTP\_SLOP\_MMV\_MIN_{(n)} = DTP\_SLOP\_MMV_{(n)}$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_MMV_CRLC	1	0…FFH	0…0.996	3.9e-3	-
Dynamic filter factor for the DTP_MMV calculation					
C_DTP_SLOP_MMV_CRLC	1	0…FFH	0…0.996	3.9e-3	-
Dynamic filter factor for the DTP_SLOP_MMV calculation					
C_T_MAX_FAC_SLOP_MMV	1	0…FFH	0…25.5	0.1	s
Time at the beginning of DTP_EVAC before calculating DTP_SLOP_MMV_MAX/MIN					
C_T_AST_MIN_DIAGCP_1	1	0…FFFFH	0…6553.5	0.1	s
Minimum time after start for activation of 0.5mm leak detection					
C_TCO_ST_MIN_DIAGCP_1	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature minimum threshold for activation of 0.5mm leak detection					
C_TCO_ST_MAX_DIAGCP_1	1	0...FEH	-48...142.5	0.75	°C
Coolant temperature maximum threshold for activation of 0.5mm leak detection					
IP_DTP_MAX_DIF_SLOP_EOL	12	0…FFFFH	0…819.2	0.0125	hPa/s
LDPM_FAC_DIAGCP_VOL	12	0…FFH	0…99.61	0.39	%
Maximum slop variation during evacuation based on fuel volume at EOL test					
IP_DTP_MAX_DIF_SLOP	12	0…FFFFH	0…819.2	0.0125	hPa/s
LDPM_FAC_DIAGCP_VOL	12	0…FFH	0…99.61	0.39	%
Maximum slop variation during evacuation based on fuel volume					
C_DTP_SLOP_DIF_MMV_CRLC	1	0…FFFFH	0…0.999984	1.5258e-5	-
Dynamic filter factor for the DTP_SLOP_MMV_DIF_MMV calculation					
C_DTP_SLOP_DIF_MMV_DIF	1	0...FFFFH	0...819.2	0.0125	hPa/s
DTP slop minimum difference to determine DTP slope calculation ending during evac					

B.10.4.10 Fuel flow correction during opened CPS

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FAC_LAM_DIAGCP	V/O	0...1H	0...1	-	-
Boolean information for Lambda control to set FAC_LAM_MV_MMV to FAC_LAM_DIAGCP					
FAC_LAM_DIAGCP	V/O	8000...7FFFH	-50...50	0.0015	%
LAM_MV value at the beginning of CPS opening					
CL_INTER	V/O	0H...FFFFH	0...8	0.000122	-
Last CL for injection time correction					
MFF_ADD_DIAGCP	V/O	0...FFFFH	0...1389	0.02119	mg/stk
Mass fuel flow correction during evacuation period (interface reasons)					

Input data:

CL_MMV	CL_MMV_CLC_END	MAF_CPS	LV_NORM_PURGE_END_1
N	LV_DIAGCPS_ACT	STATE_DIAGCP	LV_FAC_LAM_DIAGCP_END[NC_CBK_EX_NR]

Calculation of the additive fuel flow correction MFF_ADD_DIAGCP

General information:

During several phases the CPS will be opened more than the normal purge-function allows for idle-speed condition. The additive fuel flow correction MFF_ADD_DIAGCP is calculated by help of the actual canister load CL_MMV and supports the lambda controller to keep the A/F Ratio at 1.

Formula section:

If LV_NORM_PURGE_END_1 = 1
then CL_INTER = C_FAC_TI_DIAGCP *CL_MMV_CLC_END
else CL_INTER = C_FAC_TI_DIAGCP *CL_MMV
endif

Calculation of MFF_ADD_DIAGCP; Necessary for interface reasons; kg/h – mg/stk):

If LV_DIAGCPS_ACT = 1
then **if** STATE_DIAGCP = DLY_DIAG (09h)
then **if** MFF_ADD_DIAGCP > 0
then MFF_ADD_DIAGCP (n) = MFF_ADD_DIAGCP (n-1) – C_MFF_DEC_DIAGCP
endif
else MFF_ADD_DIAGCP = CL_INTER * MAF_CPS
endif
else MFF_ADD_DIAGCP = 0
endif

Remark:

- Factor required for conversion *kg/h - mg/stk:* 10^5 / (N * NC_CYL_NR * 3)

FAC_LAM_MV_MMV[1] correction during Evacuation:

Although applying an additive MFF-Correction during Evacuation, the Lambda controller has to correct the rest of the lambda deviation.

To avoid Lambda-Jumps at the “Closing Step” of the CPS (LV_FAC_LAM_DIAGCP = 1) the system shifts the lambda controller *to the Lambda controller mean value before CPS opening (FAC_LAM_DIAGCP)*, but only if FAC_LAM_DIAGCP – FAC_LAM_COR[1] > 0 (Lambda controller difference in lean direction)

See also chapter 7, injection-Lambda controller.

As soon as the lambda shift has been performed by LACO, LV_FAC_LAM_DIAGCP_END[i] is set by LACO.

See also LACO aggregate.

If LV_FAC_LAM_DIAGCP_END[i] = 1
then LV_FAC_LAM_DIAGCP = 0
FAC_LAM_DIAGCP[i] = 0
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_TI_DIAGCP	-	0...FFH	0...1	0,0039	-
Scaling factor for MFF_ADD_DIAGCP					
C_MFF_DEC_DIAGCP	-	0...FFFFH	0...1389	0.02119	mg/stk
Mass fuel flow correction decrement after evacuation period (interface reasons)					

B.10.4.11 Dynamic window on MAF fluctuation during ´Evacuation´ period:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_MMV_DIAGCP	V	0...FFFFH	0...1389	0.0212	mg/stk
MAF dynamic window value during the evacuation					

Input data:

MAF			

Description:

During the *´Evacuation´* period the diagnosis is aborted (ERR_CLOSE_CPS) and switched to “wait period” state., if the MAF fluctuation is bigger than C_MAF_MMV_DIAGCP (e.g. “air conditionning” activation).

Formula section :

During the complete 'Evacuation period' the following check is performed:

$MAF\_MMV\_DIAGCP_{(n)} =$

$MAF\_MMV\_DIAGCP_{(n-1)} + C\_MAF\_MMV\_CRLC\_DIAGCP * (MAF_{(n)} - MAF\_MMV\_DIAGCP_{(n-1)})$

If $\left| MAF\_MMV\_DIAGCP - MAF \right| < C\_MAF\_MMV\_DIAGCP$

then continue 'Evacuation'

else perform 'ERR_CLOSE_CPS'

switch to 'Wait period' (automatically after 'ERR_CLOSE_CPS')

MAF_MMV_DIAGCP is initialised with MAF at beginning of 'Evacuation period'.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_MMV_DIAGCP	1	0...FFFFH	0...1389	0.0212	mg/stk
Maximum threshold for MAF dynamic window value during the evacuation					
C_MAF_MMV_CRLC_DIAGCP	1	0...FFH	0...0.996	3.9e-3	-
Dynamic filter coefficient for the MAF_MMV_DIAGCP calculation					

B.10.4.12 Cooperation with the canister purge function

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_STATE_RST_CLC_END	V/O	0…1H	0…1	1	
Bit signalling high vapour generation or high lambda deviation at evacuation phase					
T_CLC_END	V/O	0...FFH	0…255	1	s
time since LV_NORM_PURGE_END_1 was set to 1					

Input data:

STATE_INTR_DIAGCP	LV_NORM_PURGE_END_1		

Description:

Because of exorbitant vapour generation or too high lambda deviation at the evacuation phase, a new MAX_PURGE phase of the canister purge function must be awaited. This is realized by setting the bit LV_STATE_RST_CLC_END.

Recurrence : 1000 ms

Formula section:

If Bit # 1 **or** Bit # 2 of STATE_INTR_DIAGCP = **changes** to 1
then LV_STATE_RST_CLC_END = 1 ;new MAX_PURGE mode is needed
If LV_NORM_PURGE_END_1 = 1
then increment timer T_CLC_END

If LV_NORM_PURGE_END_1 **changes** 0 → 1
or new Driving cycle detetcted
then LV_STATE_RST_CLC_END = 0 ;MAX_PURGE mode finished
reset timer T_CLC_END

B.10.4.13 Gradient method

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
DTP_GRD	V	8000...7FFFH	-40.96...40.96	0.00156	hPa/s
Tank pressure gradient for determination of tank level					
FAC_DIAGCP_VOL	V	0..FFH	0...99.61	0.39	%
Fuel tank level evaluation obtained by gradient method					
DTP_DIF_ACT_GRD_DIAGCP	V	8000...7FFFH	-40.96...40.96	0.00125	hPa
Differential DTP during C_T_DTP_GRD_DIAGCP					
T_DTP_GRD_DIAGCP	V	0..FFFFH	0...2621.4	0.04	s
Duration time for fuel tank level evaluation					

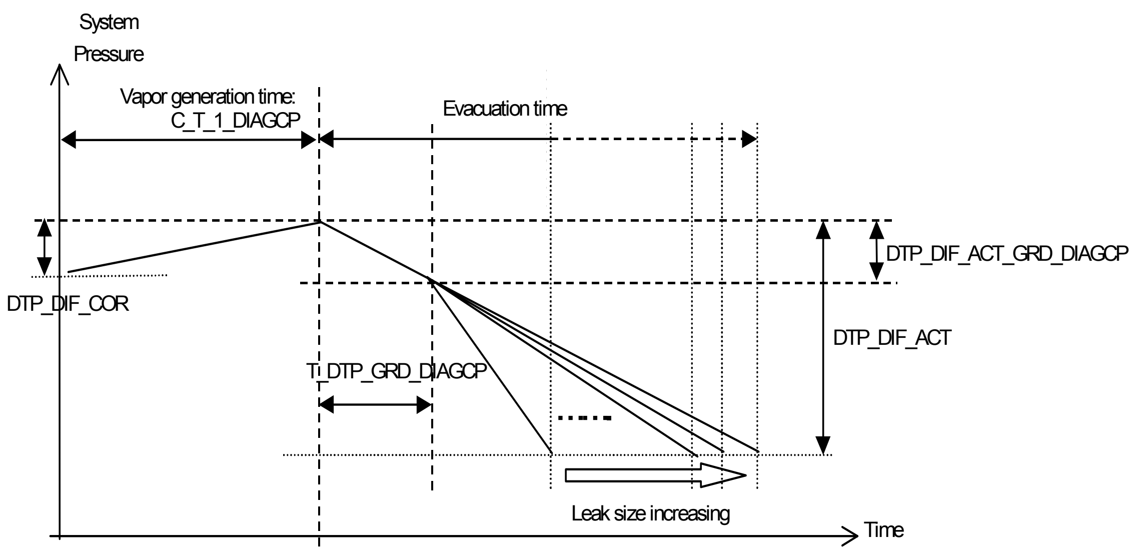
Input data:

DTP_DIF_COR	DTP_DIF_ACT	FAC_DIAGCP_MDL_BAS	C_T_1_DIAGCP
CONF_DIAGCP_VOL	LV_ERR_FTL	FTL_MMV	

Description:

For determination of the tank level the pressure gradient during ‘test for vapour generation’ and during ‘evacuation’ is analyzed. With increasing fuel level the gradient will be higher due to less air volume which has to be evacuated. The gradient during ‘test for vapour generation’ is taken into acount because high evaporation influences the evacuation time.

The duration of the evacuation phase for pressure gradient calculation is defined by C_T_DTP_GRD_DIAGCP. The corresponding pressure decrease is stored as DTP_DIF_ACT_GRD_DIAGCPi.



Formula section:

During C_T_DTP_GRD_DIAGCP seconds the following is calculated for every sampling step:

For i = 0 to n

DTP_DIF_ACT_GRD_DIAGCPi = DTP_DIF_ACTi

DTP_GRDi = (DTP_DIF_ACT_GRD_DIAGCPi / T_DTP_GRD_DIAGCP) -
(DTP_DIF_COR / C_T_1_DIAGCP)

where T_DTP_GRD_DIAGCP is the current evacuation time for gradient calculation since beginning of evacuation.

Calculation of FAC_DIAGCP_VOL (if no FTL available) :

If CONF_DIAGCP_VOL = 0 (no FTL information)

or LV_ERR_FTL = 1 (FTL sensor error)

then FAC_DIAGCP_VOL = IP_FAC_DIAGCP_VOL (DTP_GRD) *
IP_FAC_DIAGCP_VOL_COR (FAC_DIAGCP_MDL_BAS)

else FAC_DIAGCP_VOL = FTL_MMV

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_DIAGCP_VOL	1*8	0..FFH	0...99.61	0.39	%
LDP_DTP_GRD__IP_FAC_DIAGCP_VOL	8	0...FFFFH	-40.96...40.96	0.00125	hPa/s
FAC_DIAGCP_VOL information in % provided by the gradient method					
IP_FAC_DIAGCP_VOL_COR	1*4	0..FFH	0...2	0.0078	-
LDP_FAC_DIAGCP_VOL_COR_MDL_BAS	4	0...FFFFH	0...0.467	7.126e-6	-
DIAGCP_VOL correction depending on the vapor generation during DTP_COR					
C_T_DTP_GRD_DIAGCP	1	1...FFFFH	0.04...2621.4	0.04	s
Duration of the gradient calculation during ´evacuation phase´					

B.10.4.14 Release of 'Model Parameter Estimation' (DLY_DIAG state)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FAC_DIAGCP_MDL	V	0...1H	0...1	-	-
Signal for start parameter estimation : (0) for final stop, (1) for start					

Input data:

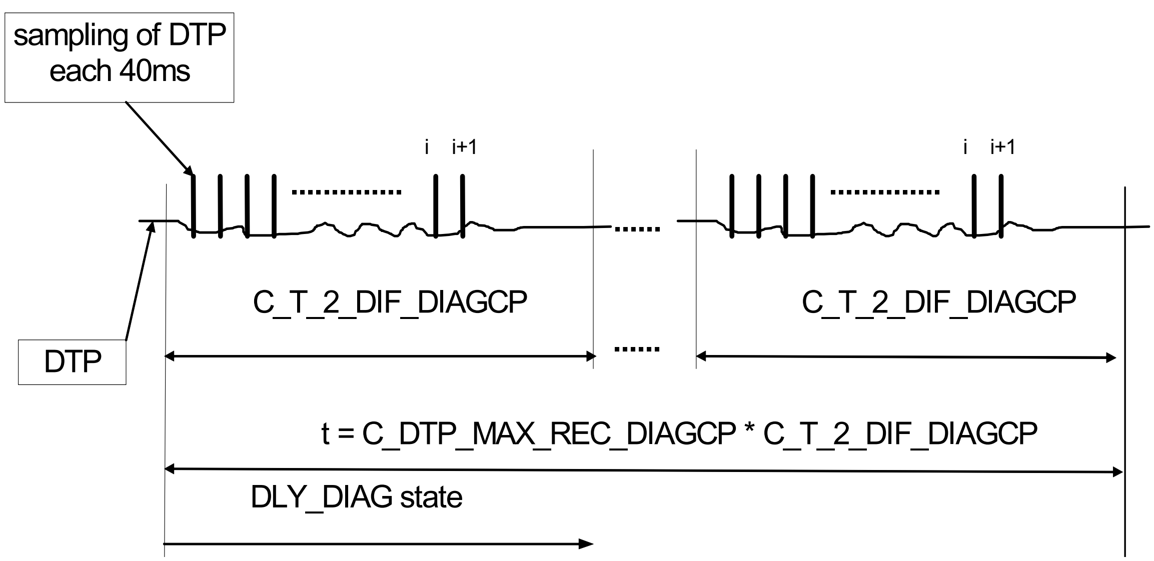
DTP_MV	STATE_DIAGCP	DTP_MV_PER	DTP_SLOP
FAC_DIAGCP_VOL	EOL_STATE_DIAGCP		

General Information:

After closing the CPS at the end of the *´Evacuation Period´* the pressure will continue to drop due to storing effects in the tube system. The pressure slope determination is introduced in order to recognize the reversal of the pressure (DTP_SLOP > C_DTP_SLOP_MIN_DIAGCP). The system will be checked for tightness and the fuel volume must be within a certain range (see chapter 1.2 *´Monitoring Conditions´*).

The pressure slope is determined in [hPa/sec]. For this purpose the pressure mean-value (DTP_MV_PER) during the period C_T_2_DIF_DIAGCP is calculated, starting at the end of 'Evacuation' period.

Overview of ´DLY_DIAG´ state:



The fuel-volume check is be by-passed in case of end-of-line test, i.e. EOL_STATE_DIAGCP must be passive:

If EOL_STATE_DIAGCP <> EOL_ACT

then

if C_FAC_DIAGCP_VOL_MIN ≤ FAC_DIAGCP_VOL ≤ C_FAC_DIAGCP_VOL_MAX

then continue ‘DLY_DIAG’ state until start of ‘Model Parameter Estimation’

else switch to ’Wait period'

Formula section:

If DTP_MV_PER $_{j}$ and DTP_MV_PER $_{j-1}$ are valid (after 2 measurements cycles) then the pressure slope determination is possible:

DTP_MV_PER $_{j}$ = [SUM (DTP $_{i}$)] / (C_T_2_DIF_DIAGCP / 0.040 [s])

SUM (DTP $_{i}$) = Sum of all pressure values during C_T_2_DIF_DIAGCP

Pressure slope:

DTP_SLOP j-1 = (DTP_MV_PER j - DTP_MV_PER j-1)
/ (C_T_2_DIF_DIAGCP - 0.1 [s])

Start of Model Parameter Estimation:

The Model Parameter Estimation starts if the pressure increases again above the start pressure C_DTP_DIAGCP with a positive slope.

IF DTP_SLOP >= C_DTP_SLOP_MIN_DIAGCP

then **if** DTP_MV > C_DTP_DIAGCP

then LV_FAC_DIAGCP_MDL = 1 (start parameter estimation; ´DTP_DIAG´ state)

else **if** C_DTP_MAX_REC_DIAGCP is reached

then EVAP-system detection 'leak-free'

else next calculation of ´DTP_SLOP´

endif

endif

else switch to *´Wait Period´*

endif

Additionally it is permanentely checked, if during ´Model Parameter Estimation´ DTP_MV falls below C_DTP_MV_MIN_DIAGCP. This case of DTP-undershoot the diagnosis is immediately aborted:

If STATE_DIAGCP = DLY_DIAG
and DTP_MV < C_DTP_MV_MIN_DIAGCP
then switch to *´Wait Period´*
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_DIAGCP	-	8000...7FFFH	-40.96...40.96	0.00125	hPa
Minimum pressure for diagnosis period					
C_DTP_MAX_REC_DIAGCP	-	1...FFH	1...255	1	-
Maximum number of C_T_2_DIF_DIAGCP periods before starting dtp_diag state					
C_DTP_SLOP_MIN_DIAGCP	-	8000...7FFFH	-40.96...40.96	0.00125	hPa/s
Minimum allowed pressure slope threshold					
C_FAC_DIAGCP_VOL_MAX	1	0...FFH	0...99.61	0.39	%
Maximum fuel volume allowed to do the diagnosis					
C_FAC_DIAGCP_VOL_MIN	1	0...FFH	0...99.61	0.39	%
Minimum fuel volume allowed to do the diagnosis					
C_T_2_DIF_DIAGCP	-	0...FFH	0...25.5	0.1	s
Time period used to determine pressure slope before diagnosis					
C_DTP_MV_MIN_DIAGCP	-	8000...7FFFH	-40.96...40.96	0.00125	hpa
Minimum pressure threshold to stop parameter estimation					

B.10.4.15 Diagnosis (DTP_DIAG state)

B.10.4.15.1 Diagnosis ‘Model Parameter Estimation’

For parameter estimation the actual pressure value $DTP_{(i)}$ is used.

The parameter estimation is performed if LV_FAC_DIAGCP_MDL = 1 and lasts C_T_2_DIAGCP seconds.

The estimation-result **FAC_DIAGCP_MDL_DIAG** is used for the actual leakage determination. (see chapter “Model Parameter Estimation”).

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_2_DIAGCP	-	0...FFH	0...25.5	0.1	s
Duration time to diagnosis model parameter estimation					

B.10.4.15.2 Fuel tank level (FTL) information

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FTV_COR_DIAGCP	V	0...FFH	0...99.61	0.39	liters
Air-fuel volume correction (included tank elasticity correction)					

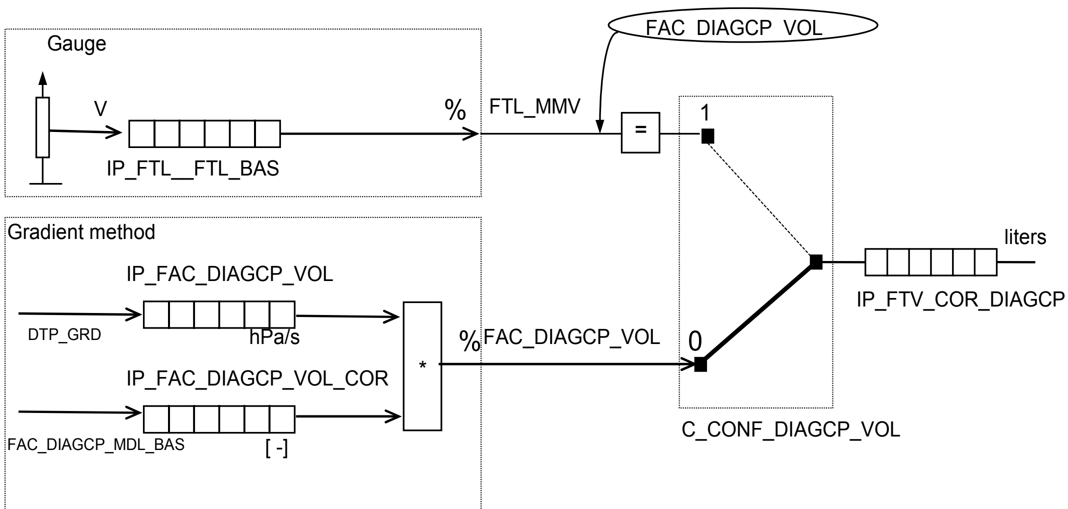
Input data:

FAC_DIAGCP_VOL	FTL_MMV	CONF_DIAGCP_VOL	LV_ERR_FTL

General information:

It is helpful to get approximately the fuel tank level because the diagnosis depends on the volume of the *fuel-air mixture* in the tank. As many possibilities to determine the FTL exist, we use a calibrated switch to select the source of the FTL information. To select the input to use for FTL information see C_CONF_DIAGCP_VOL in general chapter.

Description:



In case of using the gauge input for FTL information (CONF_DIAGCP_VOL = 1), the FAC_DIAGCP_VOL value is freezed at the transition of DTP_EVAC to DLY_DIAG.

When FTL is malfunction, the FAC_DIAGCP_VOL is determined by gradient method (see next chapter) and not switched to FTL_MMV at the transition of DTP_EVAC to DLY_DIAG with C_CONF_DIAGCP_VOL = 1.

FTV_COR_DIAGCP represents the fuel-air mixture in the tank.

Formula Section:

FTV_COR_DIAGCP = IP_FTV_COR_DIAGCP

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FTV_COR_DIAGCP	12	0...FFH	0...99.61	0.39	liters
LDPM_FAC_DIAGCP_VOL	12	0...FFH	0...99.61	0.39	%
Aif-fuel volume correction function of FAC_DIAGCP_VOL (%)					

B.10.4.15.3 Determination of actual leakage area

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_DIAGCP_MDL_END	V	0000...FFFFH	0...0.467	7.126e-6	-
Final leakage parameter (including correction due to vapour generation)					
AR_DIAGCP	V	0000...FFFFH	0...0.1348	2.05e-6	cm²
Reduced leakage area estimated					

Input data:

FAC_DIAGCP_MDL_DIAG	FAC_DIAGCP_MDL_ESTIM	AMP	FTV_COR_DIAGCP
TEMP_COR_DIAGCP	FAC_TI_ST_AD_DIAGCP		

Description:

The estimated parameter from 'Diagnosis Period' has to be corrected due to vapour generation:

=> FAC_DIAGCP_MDL_END = FAC_DIAGCP_MDL_DIAG - FAC_DIAGCP_MDL_ESTIM

FAC_DIAGCP_MDL_END = f(fuel-tank-air-volume, ambient-pressure, ambient-temperature, leaksize)

If DIAG_MDL_END < 0

then DIAG_MDL_END = 0

Now the actual leakage can be determined:

AR_DIAGCP = ((FTV_COR_DIAGCP * FAC_DIAGCP_MDL_END) / IP_AMP_SQRT) * IP_FAC_DIAGCP

Where AR_DIAGCP is the reduced leakage area : alpha*A.

In this formula, the volume correction is applied in m^3 whereas as output it is defined in liters (see chapter "Fuel tank level information").

Theoritically : $\text{IP\_FAC\_DIAGCP} = \sqrt{\frac{2*\rho*T_0}{P_0*ambient-temperature}}$

Where the ambient temperature is equal to 293 K (20°C).

In the map IP_AMP_SQRT the square-root of the ambient-pressure is calibrated.

IP_AMP_SQRT = sqrt (AMP)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_AMP_SQRT	6	0...FFFFH	0...34.641	5.286E-4	√hPa
LDP_AMP_IP_AMP_SQRT	6	0...FFFFH	0...5434	0.083	hPa
Square root of AMP					
IP_FAC_DIAGCP	6*8	0...FFH	0...0.996	3.9E-3	-
LDP_TEMP_COR_DIAGCP_IP_FAC	8	0...FEH	-48...142.5	0.75	°C
LDP_FAC_TI_ST_AD_DIAGCP_IP_FAC	6	0...FFH	0...1.992	0.0078	-
Fuel temperature and fuel quality factor					

B.10.4.15.4 Determination of actual leakage-diameter

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_DIAM_DIAGCP	V/O	0...FFH	0...4.127	16.184e-3	Mm
Reduced leakage diameter					
FAC_DIAM_DIAGCP_MOD_6	V/O/S	0...FFH	0...4.127	16.184e-3	Mm
Reduced leakage diameter stored for mode 06-communication					

Input data:

AR_DIAGCP	FAC_DIAGCP_MDL_BAS		

The existing leakage (reduced-) diameter can be determined:

FAC_DIAM_DIAGCP = IP_FAC_SQRT_AR_DIAGCP

Where :

IP_FAC_SQRT_AR_DIAGCP = 200 * sqrt (AR_DIAGCP / pi)

IF C_DIAM_MIN_DIAGCP_VAP_CHK < FAC_DIAM_DIAGCP
< C_DIAM_MAX_DIAGCP_VAP_CHK
and FAC_DIAGCP_MDL_BAS < C_FAC_DIAGCP_MAX_VAP_CHK
and C_T_2_DIAGCP is elapsed
and LC_VAP_CHK_INH = 0
then the system switches to DLY_VAP_CHK state
else the system switches to “control diagnostic error – 0.5 mm”
endif

Special treatment for Diagnostic Test Mode:

If readyness of symptom LEAK_SMALL_05MM or LEAK_SMALL_1MM is detected, FAC_DIAM_DIAGCP is stored as FAC_DIAM_DIAGCP_MOD_6

The value FAC_DIAM_DIAGCP_MOD_6 should be initialized with 0 in case of failure memory is cleared.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_DIAGCP_MAX_VAP_CHK	1	0000...FFFFh	0...0.467	7.126e-6	-
Maximum FAC_DIAGCP_MDL_BAS value to start VAP_CHK					
C_DIAM_MIN_DIAGCP_VAP_CHK	1	00...FFH	0...4.127	16.184e-3	Mm
Minimum FAC_DIAM_DIAGCP value to start VAP_CHK					
C_DIAM_MAX_DIAGCP_VAP_CHK	1	00...FFH	0...4.127	16.184e-3	Mm
Maximum FAC_DIAM_DIAGCP value to start VAP_CHK					
LC_VAP_CHK_INH	-	0...1h	0...1	1	-
Deactivate VAP_CHK					
IP_FAC_SQRT_AR_DIAGCP	20	0...FFH	0...4.127	16.184e-3	mm
LDP_AR_DIAGCP_IP_FAC_SQRT	20	0...FFFFH	0...0.1348	2.05e-6	cm²
Square root of DIAGP_AR					

B.10.4.16 Delay time before changing to VAP_CHK state

Input data:

STATE_DIAGCP			

Description:

At transition into this state, a delay time C_T_DLY_VAP_CHK is started. After this delay time, the algorithm switches to VAP_CHK.

Formula section:

If STATE_DIAGCP = DLY_VAP_CHK
then LV_SOV_REQ = 0
endif

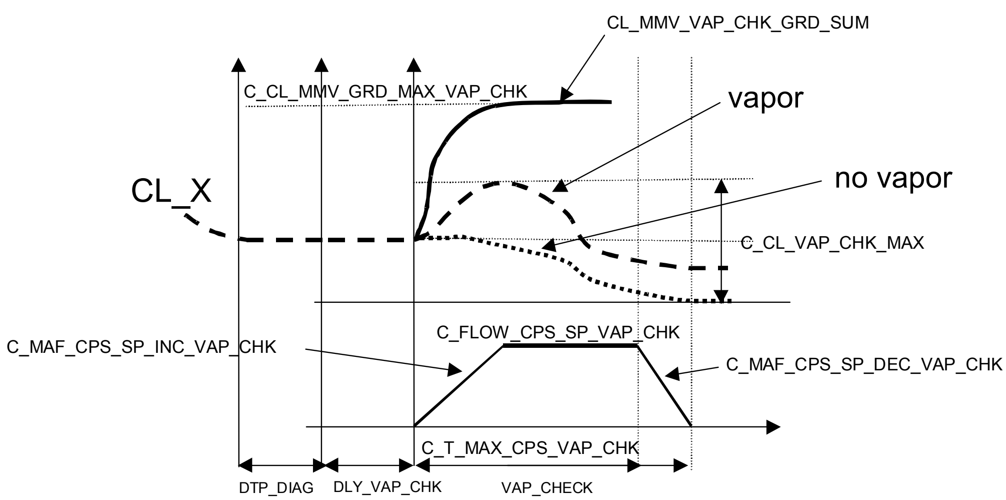
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_VAP_CHK	1	0...FFH	0…10.20	0.04	s
Delay time before vapor check					

B.10.4.17 Vapor Monitoring (STATE_DIAGCP = VAP_CHK)

Summary:

At certain ambient conditions (t_{amb}, p_{amb}, Δp_{amb}, fuel vapor pressure) the fuel tends to evaporate only at reduced system pressure although with the result of FAC_DIAGCP_MDL_BAS below a threshold. This can lead to an over detection in the diagnosis phase, due to no or incorrect correction of the DTP slope. The evaporated fuel that is in the fuel tank system can be visualised by purging the system immediately after the diagnosis. A calculated Canister Load CL_VAP_CHK (depending on the lambda control behaviour and analogue the CP – function) will show the amount of fuel vaporised in the tank system in terms of gradient (CL_MMV_VAP_CHK_GRD) and absolute value (CL_MMV_VAP_CHK).



Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FLOW_SP_CPS_VAP_CHK	V/O	0...FFFFH	0...8	0.000122	kg/h
flow setpoint for MAF_CPS at evacuation for CL gradient check					
LV_VAP_CHK_END	V	0...1H	0...1	-	-
Flag indicating that VAP_CHK is finished					
T_VAP_CHK	V	0...FFH	0…10.20	0.04	s
Timer during purge on for CL gradient check					
CL_VAP_CHK	V	0H...FFFFH	0...8	0.000122	-
Canister load CL calculated for vapor check					
CL_MMV_VAP_CHK	V	0H...FFFFH	0...8	0.000122	-
Canister load (mean value) calculated for gradient check					
CL_MMV_VAP_CHK_GRD	V	0H...FFFFH	0...8	0.000122	-
Canister load (mean value) gradient					
CL_VAP_CHK_TMP	V	0H...FFFFH	0...8	0.000122	-
Canister load temporary value					
T_DTP_GRD_VAP_CHK	V	0…FFH	0…10.20	0.04	s
Timer for GRD calculation					
MFF_VAP_CHK_DIAGCP	V	8000H...7FFFH	-8...8	0.000244	kg/h
Fuel flow from the ACF to check CL behaviour					
FAC_LAM_DIF_CP_VAP_CHK	V	8000H...7FFFH	-50...50	0.0015	%
Lambda deviation calculated during vapor check					
CL_MMV_VAP_CHK_GRD_SUM	V	0H...FFFFH	0...8	0.000122	-
Sum of Canister load (mean value) gradient					
FAC_LAM_0_CP_VAP_CHK	V	8000H...7FFFH	-50...50	0.0015	%
Initial lambda value at the beginning of VAP_CHK state					

Input data:

C_RAF_CLC_CP	FAC_LAM_MV_MMV[NC_CBK_EX_NR]	LV_LAM_LIM_MIN_i	LV_PRS_LIM_DYN_DIAGCP
MAF	FAC_LAM_DIAGCP	MFF_ADD_CP_KGH	N_32
LAM_MV_i	MAF_TOT_CP_DLY_MMV	CL_INTER	C_FAC_DIAGCP_MAX_VAP_CHK
FLOW_DLY_MMV_CP			

General information:

To check the VAP_CHK gradient after evacuation, a value FLOW_SP_CPS_VAP_CHK is calculated to purge the system. It is a setpoint for the requested mass flow through the CPS performed by the Evaporative Emission Control function.

Application conditions:

Initialisation by transition into state

T_VAP_CHK = 0

FLOW_SP_CPS_VAP_CHK = 0

LV_VAP_CHK_END = 0

FAC_LAM_DIF_CP_VAP_CHK = 0

FAC_LAM_0_CP_VAP_CHK = LAM_MV_I

MFF_VAP_CHK_DIAGCP = 0

CL_VAP_CHK = 0

CL_VAP_CHK_TMP = CL_VAP_CHK (=0)

T_DTP_GRD_VAP_CHK = C_T_DTP_GRD_VAP_CHK

CL_MMV_VAP_CHK = 0

CL_MMV_VAP_CHK_GRD = 0

CL_MMV_VAP_CHK_GRD_SUM = 0

Opening of the valve:

At the beginning of the VAP_CHK state, FLOW_SP_CPS_VAP_CHK is increased to C_FLOW_CPS_SP_VAP_CHK by C_FLOW_CPS_SP_INC_VAP_CHK

FLOW_SP_CPS_VAP_CHK is increased or set to C_FLOW_CPS_SP_VAP_CHK for max. C_T_MAX_CPS_VAP_CHK.

Closing of the valve:

In the following cases, FLOW_SP_CPS_VAP_CHK is reduced to 0 via the slope C_FLOW_CPS_SP_DEC_VAP_CHK (STATE_DIAGCP = VAP_CHK):

Normal end of purge phase or if the lambda controller value (FAC_LAM_MV_MMV[1]) is shifted for more than C_FAC_LAM_MAX_VAP_CHK (for lean mixture), respectively for more than C_FAC_LAM_MIN_VAP_CHK (rich mixture) from the lambda control mean value at the beginning of evacuation (FAC_LAM_DIAGCP),

Formula section:

If FAC_LAM_DIAGCP - FAC_LAM_MV_MMV[1] >=C_FAC_LAM_MAX_VAP_CHK
or FAC_LAM_DIAGCP - FAC_LAM_MV_MMV[1] <= C_FAC_LAM_MIN_VAP_CHK
or the lambda controller is limited in lean direction (LV_LAM_LIM_MIN_i = 1)
or { (CL_MMV_VAP_CHK_GRD_SUM$_{(n)}$ > C_CL_MMV_GRD_MAX_VAP_CHK
    or CL_MMV_VAP_CHK > C_CL_VAP_CHK_MAX)
    and FAC_DIAGCP_MDL_BAS < C_FAC_DIAGCP_MAX_VAP_CHK }
then FLOW_SP_CPS_VAP_CHK is reduced by C_FLOW_CPS_SP_DEC_VAP_CHK to 0 (and also if trigger conditions not furthermore fulfilled).
    if FLOW_SP_CPS_VAP_CHK = 0
    then system switches to "wait period"
    endif
else **if** T_VAP_CHK(n) < C_T_MAX_CPS_VAP_CHK
    then T_VAP_CHK(n) = T_VAP_CHK(n-1) + 1
        if (CL_MMV_VAP_CHK_GRD_SUM$_{(n)}$ > C_CL_MMV_GRD_MAX_VAP_CHK
        or CL_MMV_VAP_CHK > C_CL_VAP_CHK_MAX)
        and FAC_DIAGCP_MDL_BAS < C_FAC_DIAGCP_MAX_VAP_CHK

then FLOW_SP_CPS_VAP_CHK is reduced by C_FLOW_CPS_SP_DEC_VAP_CHK to 0 and switch to STATE_DIAGCP = T_DLY_3 and set STATE_INTR_DIAGCP = 14

else

if FLOW_SP_CPS_VAP_CHK < C_FLOW_CPS_SP_VAP_CHK

then FLOW_SP_CPS_VAP_CHK is increased by C_FLOW_CPS_SP_INC_VAP_CHK to C_FLOW_CPS_SP_VAP_CHK

else FLOW_SP_CPS_VAP_CHK= C_FLOW_CPS_SP_VAP_CHK

endif

endif

else FLOW_SP_CPS_VAP_CHK is reduced by C_FLOW_CPS_SP_DEC_VAP_CHK to 0 (and also if trigger conditions not furthermore fulfilled).

if (FLOW_SP_CPS_VAP_CHK = 0

and T_VAP_CHK >= C_T_MAX_CPS_VAP_CHK)

then LV_VAP_CHK_END = 1

endif

endif

Setting of STATE_INTR_DIAGCP

If wait period is due to FAC_LAM_DIAGCP or LV_LAM_LIM conditions, then STATE_INTR_DIAGCP = 3 (Limited dynamics not fulfilled)

If wait period is due to CL_MMV_VAP_CHK_GRD_SUM or CL_MMV_VAP_CHK conditions, then STATE_INTR_DIAGCP = 14 (VAP_CHK too high)

The injection time is corrected as in chapter “Injection time correction during opened CPS” described.

The SOV is open during this complete state LV_SOV_REQ = 0

Function interruption:

LV_PRS_LIM_DYN_DIAGCP (limited dynamics violated) changes from 1 to 0 during cps opened, FLOW_SP_CPS_VAP_CHK is set to 0 (suddenly) and the ´Wait Period´ is activated.

When the valve is opened to purge the system, the degree of HC saturation VAP_CHK is calculated. The gradient of this value and the delta to the last valid loading degree can be an indicator for high vapour generation in the evacuation phase. A detected leak can be suppressed and the diagnosis start again (transition into STATE_DIAGCP = T_DLY_3).

Calculation of FAC_LAM_DIF_CP_VAP_CHK:

if FAC_LAM_0_CP_VAP_CHK - FAC_LAM_MV_MMV[1] <= C_FAC_LAM_COR_MIN_VAP_CHK

then FAC_LAM_DIF_CP_VAP_CHK = FAC_LAM_0_CP_VAP_CHK - FAC_LAM_MV_MMV[1] - C_FAC_LAM_COR_MIN_VAP_CHK

elseif FAC_LAM_0_CP_VAP_CHK - FAC_LAM_MV_MMV[1] >= C_FAC_LAM_COR_MAX_VAP_CHK

then FAC_LAM_DIF_CP_VAP_CHK = FAC_LAM_0_CP_VAP_CHK - FAC_LAM_MV_MMV[1] - C_FAC_LAM_COR_MAX_VAP_CHK

endif

Calculation of MFF_VAP_CHK_DIAGCP:

MFF_VAP_CHK_DIAGCP = MAF_TOT_CP_DLY_MMV * FAC_LAM_DIF_CP_VAP_CHK / C_RAF_CLC_CP+ MFF_ADD_CP_KGH

Calculation of CL_VAP_CHK, CL_MMV_VAP_CHK:

if FLOW_DLY_MMV_CP = 0

then CL_VAP_CHK = 0

else CL_VAP_CHK = MFF_VAP_CHK_DIAGCP / FLOW_DLY_MMV_CP

endif

if CL_VAP_CHK < 0 THEN CL_VAP_CHK = 0

elseif CL_VAP_CHK > C_CL_MAX_VAP_CHK

then CL_VAP_CHK = C_CL_MAX_VAP_CHK

else CL_VAP_CHK = CL_VAP_CHK

endif

Calculation of CL_MMV_GRD , CL_MMV_VAP_CHK

CL_MMV_VAP_CHK(n) = (1- IP_CL_CRLC_RAMP_OPEN_VAP_CHK) * CL_MMV_VAP_CHK(n-1) + IP_CL_CRLC_RAMP_OPEN_VAP_CHK *CL_VAP_CHK (n)

DEC(T_DTP_GRD_VAP_CHK)

if T_DTP_GRD_VAP_CHK = 0

then CL_MMV_VAP_CHK_GRD = CL_VAP_CHK - CL_VAP_CHK_TMP

$CL\_MMV\_VAP\_CHK\_GRD\_SUM_{(n)} = CL\_MMV\_VAP\_CHK\_GRD\_SUM_{(n-1)} + CL\_MMV\_VAP\_CHK\_GRD$

CL_VAP_CHK_TMP = CL_VAP_CHK

T_DTP_GRD_VAP_CHK = C_T_DTP_GRD_VAP_CHK (re-init timer)

endif

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
IP_CL_CRLC_RAMP_OPEN_VAP_CHK	8*8	0H...FFFFH	0...1	0.000015	-

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
LDP_N_32_IP_CL_CRLC_RAMP_OPEN	8	0H...FFH	0...8160	32	rpm
LDP_MAF_IP_CL_CRLC_RAMP_OPEN	8	0H...FFH	0...1389	5.447	mg/stk
Correlation constant for CL-filter					
C_FAC_LAM_COR_MAX_VAP_CHK	1	8000H...7FFFH	-50...50	0.0015	%
max. correction value for FAC_LAM_DIF_CP_VAP_CHK window					
C_FAC_LAM_COR_MIN_VAP_CHK	1	8000H...7FFFH	-50...50	0.0015	%
min. correction value for FAC_LAM_DIF_CP_VAP_CHK window					
C_CL_MAX_VAP_CHK	1	0H...FFFFH	0...8	0.000122	-
Canister load threshold for vapour detected					
C_CL_MMV_GRD_MAX_VAP_CHK	1	0H...FFFFH	0...8	0.000122	-
Maximum CL threshold for “vapour detected”					
C_CL_VAP_CHK_MAX	1	0H...FFFFH	0...8	0.000122	-
Maximum CL delta for “vapour detected”					
C_T_DTP_GRD_VAP_CHK	1	0…FFH	0…10.20	0.04	s
Time range of CL gradient determination					
C_FLOW_CPS_SP_INC_VAP_CHK	1	0...FFFFH	0...7.999	0.000122	kg/h
Increment for MAF_CPS setpoint of CP-function during purging for CL check					
C_FLOW_CPS_SP_DEC_VAP_CHK	1	0...FFFFH	0...7.999	0.000122	kg/h
Decrement for MAF_CPS setpoint of CP-function during purging for CL check					
C_T_MAX_CPS_VAP_CHK	1	0...FFH	0…10.20	0.04	s
Maximum time period for purging during CL check					
C_FAC_LAM_MAX_VAP_CHK	1	8000...7FFFH	-50...50	0.0015	%
Maximum allowed lambda integrator deviation for purging during CL check					
C_FAC_LAM_MIN_VAP_CHK	1	8000...7FFFH	-50...50	0.0015	%
Minimum allowed lambda integrator deviation for purging during CL check					
C_FLOW_CPS_SP_VAP_CHK	1	0...FFFFH	0...7.999	0.000122	kg/h
Setpoint for MAF_CPS of CP-function for purging during CL check					

B.10.5 Model Parameter Estimation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_DIAGCP_MDL_BAS	V	0000...FFFFH	0...0.467	7.126e-6	-
DIAGCP leakage-parameter vapour generation period					
FAC_DIAGCP_MDL_DIAG	V	0000...FFFFH	0...0.467	7.126e-6	-
DIAGCP leakage-parameter diagnosis period					
DTP_TOT_DIF_BAS	V	80000000H... 7FFFFFFFH	- 417900...417900	1.946e-4	-
Sum of the differencial product in leakage parameter term during vapour generation phase					
DTP_TOT_DIF_DIAG	V	80000000H... 7FFFFFFFH	- 417900...417900	1.946e-4	-
Sum of the differencial product in leakage parameter term during diagnosis phase					
DTP_TOT_SUM	V	0...FFFFH	0...65535	1	-
Total number of DTP samples during C_T_y_DIAGCP					
CTR_SUM_DTP	V	0...FFFFH	0...65535	1	-
actual number of calculation operations beginning with 0					

Input data:

DTP	DTP_COR		

General information:

In order to fulfilled the need for DIAGCP_MDL_x range and resolution, TOT_DIF_SIG_x is limited for the final result to -52237.54...+52237.54.

The parameter estimation delivers two parameters which describe the characteristic of the pressure curve, FAC_DIAGCP_MDL_BAS and FAC_DIAGCP_MDL_DIAG.

These parameters are a function of leakage-size, fuel-level, ambient-pressure and ambient-temperature

The parameters determined during test for 'Vapor Generation' are marked with **_BAS**.

In this case the input is DTP_COR.

The parameters determined during test for 'Diagnosis Period' are marked with **_DIAG**.

In this case the input is DTP.

Formula section:

$$FAC\_DIAGCP\_MDL\_x = \frac{6}{N^3 \bullet SAMPLE} \bullet \left| DTP\_TOT\_DIF\_x \right|$$

with :

TOT_DIF_SIG_x (n) = $\sum_{n=0}^{N-1} \left[(N-1) - 2 \bullet n \right] \bullet IP\_DTP\_MDL\_DIAGCP(n)$

IP_DTP_MDL_DIAGCP = sqrt [DTP(n)]

N := Total number of DTP samples during C_T_y_DIAGCP

(DTP_TOT_SUM in the software)

SAMPLE := sample-rate of pressure signal DTP(n) [s]

n := actual number of calculation operations beginning with 0.

(CTR_SUM_DTP in the software)

y := 1; 2

x := BAS; DIAG

Remark : the increment in the sum TOT_DIF_SIG_x, is the value ONE :

=> For each recurrence n = n+1

DIAGCP_MDL_x is calculated only once with the last value of TOT_DIF_SIG_x.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_DTP_MDL_DIAGCP	20	0...FFFFH	-6.377...6.376	1.946e-4	√hPa
LDP_DTP_IP_DTP_MDL_DIAGCP	20	0...FFFFH	-40.96...40.96	1.25E-3	hPa
Signed square root of DTP					

B.10.6 Wait period

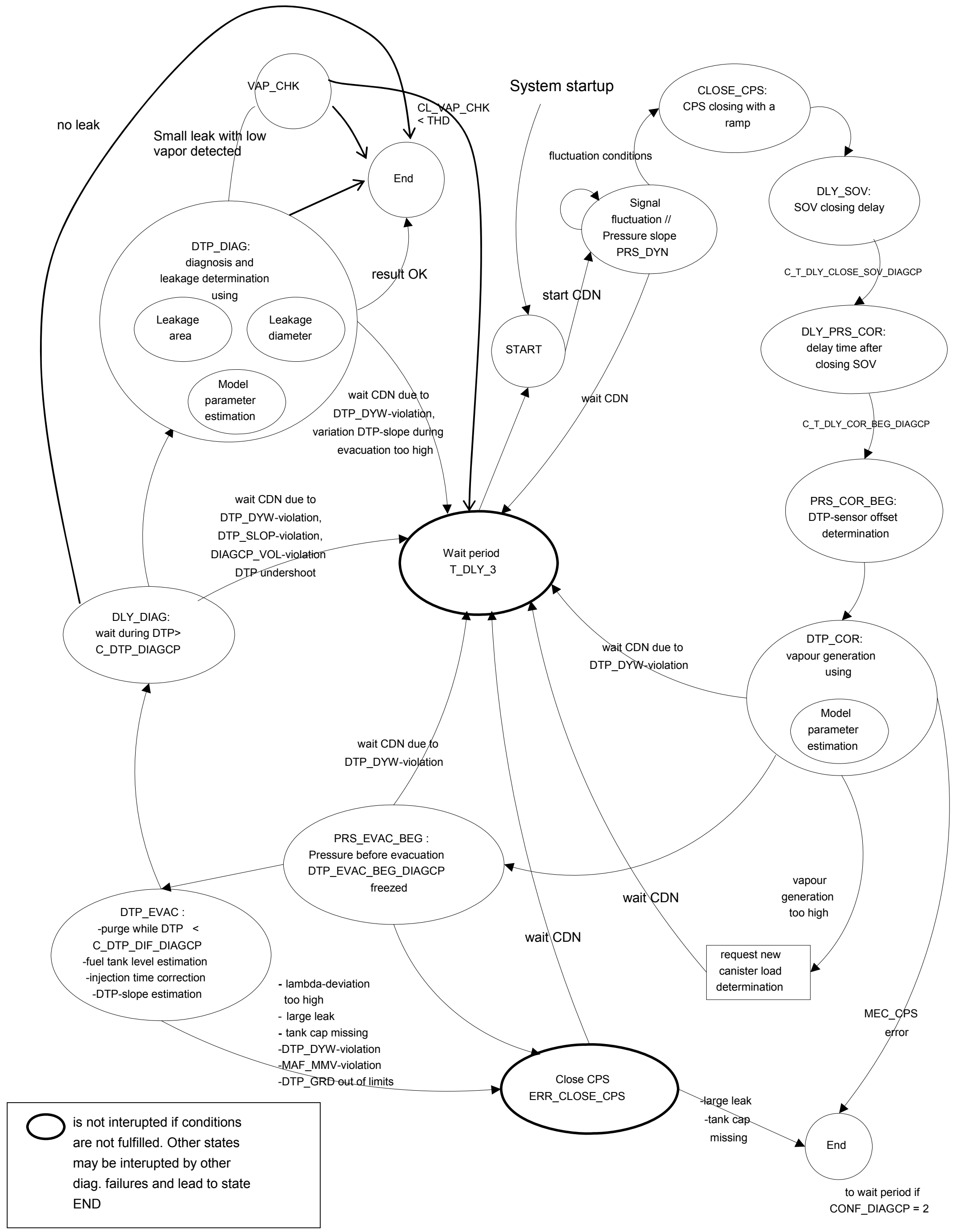
At interruption of DIAGCP without result the delay time C_T_DLY_3_DIAGCP is started. If C_T_DLY_3_DIAGCP has expired, and all diagnosis conditions exist, the diagnosis function is restarted with '*determination of sensor offset*'.

The wait period can be used to assure stabilisation of the pressure in the tank before the diagnosis is restarted. During the wait period the timer T_DIAGCP is stopped.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_3_DIAGCP	1	0...FFFFH	0...6553.5	0.100	s
Delay time for restart of DIAGCP					

Description of the status transitions:



B.10.7 Actuator test for SOV

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_SOV	V/O	0...1H	0...1	-	-
Boolean for SOV closed					
LV_SOV_REQ	V	0...1H	0...1	-	-
Request for switching the SOV valve					

Input data:

LV_SOV_ACT_EXT_ADJ	LV_SOV_EXT_ADJ	CTR_ABC_SOV	

General information:

The diagnosis tool intervention for SOV actuator test has first priority and interrupts the DIAGCP algortithm. If the actuator test for SOV is not active the SOV is switched according to the demands of the evaporative system monitoring function (LV_SOV_REQ).

Application conditions:

Initialization: at start of driving cycle

LV_SOV_REQ = 0

Recurrence: 20 ms

Formula section:

```
If      LV_SOV_ACT_EXT_ADJ = 1
then    external adjustment active
        If    LV_SOV_EXT_ADJ = 1
        then  LV_SOV = 1
        else  LV_SOV = 0
        endif
else    external adjustment not active at the moment
        LV_SOV = LV_SOV_REQ
endif
```

with LV_SOV also the HW-pin "SOV" must be switched:

```
If    CTR_ABC_SOV = 0
Then  no error symptom detected yet -> normal operation
      LV_SOV = 0  => diagnosis pulse time 0.45ms (=0.34% duty @ 7.6 Hz)
      LV_SOV = 1  => diagnosis pulse time 0ms (=100% duty)
Else  error symptom detected -> send additional test pulses for improved diagnosis
      LV_SOV = 0  => diagnosis pulse time 0.45ms (=0.34% duty @ 7.6 Hz)
      LV_SOV = 1  => diagnosis pulse time 0.45ms (=99.66% duty @ 7.6 Hz)
Endif
```

B.10.8 Fuel cap missing

B.10.8.1 Fuel cap missing detection out of leak detection algorithm

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_FUC_MISS_DIAG	V	0...1H	0...1	1	-
FUC_MISS diagnosis is in process					
LV_FUC_MISS_DIAG_CDN	V	0...1H	0...1	1	-
All conditions for “Fuel Cap Missing Diagnosis” are fulfilled					
LV_FUC_MISS_DIAG_READY	V/O	0...1H	0...1	1	-
FUC_MISS diagnosis is finished					
T_DLY_1_FUC_MISS_DIAG	V	0...FFFFH	0...2621.4	0.04	s
Delay time after LV_FUC_MISS_DIAG_CDN = 1					
T_DLY_2_FUC_MISS_DIAG	V	0...FFFFH	0...2621.4	0.04	s
Time during FUC_MISS diagnosis is in process					
T_DLY_3_FUC_MISS_DIAG	V	0...FFFFH	0...6553.5	0.1	s
Time duration in idle speed					
T_DLY_4_FUC_MISS_DIAG	V	0...FFFFH	0...6553.5	0.1	s
Time duration between two fuel cap missing diagnosis cycles					
DTP_MV_FUC_MISS_DIAG	V	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Pressure mean value before FUC_MISS diagnosis					
DTP_DIF_FUC_MISS_DIAG	V/O	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Pressure difference during FUC_MISS diagnosis					
DTP_DIF_FUC_MISS_DIAG_MOD_6	V/O/S	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Pressure difference during FUC_MISS diagnosis stored for mode 06-communication					
LV_FUC_MISS_DIAG_END	V	0...1H	0...1	1	-
Function completed					
LV_DIAG_CHK_MPL_FUC_MISS	V	0...1H	0...1	1	-
Multiple check active					
NR_CYC_INTR_FUC_MISS_DIAG	V	0...FFH	0...255	1	-
Total number of FUC_MISS_DIAG interruptions before completing cycle					
NR_CYC_FUC_MISS_DIAG	V	0...FFH	0...255	1	-
Total number of completed FUC_MISS_DIAG cycles					

Input data:

LV_INH_DIAGCP	STATE_CP	MAF_CPS	LV_IS
DTP	LV_DC	LV_CPPWM_EXT_ADJ	LV_SOV_ACT_EXT_ADJ
LV_STATE_CDN_DIAGCP	DTP_MV	VS	CONF_DIAGCP
STATE_DIAGCP	LV_DTP_PLAUS_READY	CL_MMV	AMP_AD
FAC_DIAGCP_VOL	LV_STATE_RR	LV_ERR_FUC_MISS_1	LV_ST_END
FTL_MMV	C_FTL_MAX_DIAGCP	PQ	

General information:

The purpose of the function is to detect a missing fuel cap independently from the vakuum leak detection function. Therefore the *charcoal-canister shutoff valve (SOV)* is closed during a MAX_PURGE state. This operation generates a vacuum in the fuel system. If it is not possible to evacuate the fuel system, a missing tank cap is detected.

Application conditions:

Initialisation:

⇒ *T_DLY_1_FUC_MISS_DIAG:*

The timer T_DLY_1_FUC_MISS_DIAG is resetted, if at least one of the activation condition is not met (LV_FUC_MISS_DIAG_CDN = 0) or at start of a new driving cycle (LV_DC 0→1).

⇒ *T_DLY_2_FUC_MISS_DIAG:*

The timer T_DLY_2_FUC_MISS_DIAG is resetted, if at least one of the activation condition is not met (LV_FUC_MISS_DIAG_CDN = 0) or at start of a new driving cycle (LV_DC 0→1).

⇒ *T_DLY_3_FUC_MISS_DIAG:*

If LV_IS = 1→ 0 (transition) or VS > 0 or LV_FUC_MISS_DIAG_END = 0 then T_DLY_3_FUC_MISS_DIAG = 0

⇒ *T_DLY_4_FUC_MISS_DIAG:*

If LV_FUC_MISS_DIAG_READY = 0→1 then T_DLY_4_FUC_MISS_DIAG = C_T_DLY_4_FUC_MISS_DIAG

At start of a new driving cycle (LV_DC = 0→1) T_DLY_4_FUC_MISS_DIAG = 0

⇒ NR_CYC_FUC_MISS_DIAG, NR_CYC_INTR_FUC_MISS_DIAG, LV_FUC_MISS_DIAG_END, LV_FUC_MISS_DIAG_READY and LV_DIAG_CHK_MPL_FUC_MISS are resetted to 0 at start of a new driving cycle (LV_DC 0→1).

Recurrence: *1000ms, if LV_FUC_MISS_DIAG_CDN = 0*
40ms, if LV_FUC_MISS_DIAG_CDN = 1

Activation:

If STATE_DIAGCP = START, CLD or PAS
and LV_ST_END = 1
and CONF_DIAGCP > 0
then Activation of FUC_MISS diagnosis
endif

Formula section:

- *Conditions for FUC_MISS Diagnosis (LV_FUC_MISS_DIAG_CDN = 1)*

If LV_INH_DIAGCP = 0
and LV_DTP_PLAUS_READY = 1
and LV_SOV_ACT_EXT_ADJ = 0
and LV_CPPWM_EXT_ADJ = 0
and LV_CDN_VB_OBD2 = 1
and LV_FUC_MISS_DIAG_END = 0
and LV_STATE_RR = 0
and T_DLY_4_FUC_MISS_DIAG = 0

and N_32 < C_N_MAX_FUC_MISS_DIAG

and VS > C_VS_MIN_FUC_MISS_DIAG

and FTL_MMV < C_FTL_MAX_DIAGCP

and PQ <= C_PQ_MAX_FUC_MISS_DIAG

and STATE_CP = MAX_PURGE

and C_MAF_CPS_MIN_FUC_MISS_DIAG ≤ MAF_CPS ≤ C_MAF_CPS_MAX_FUC_MISS_DIAG

and CL_MMV < C_CL_MMV_FUC_MISS_DIAG

and AMP_AD > C_AMP_AD_FUC_MISS_DIAG

and C_DTP_MIN_FUC_MISS_DIAG < DTP < C_DTP_MAX_FUC_MISS_DIAG

then LV_FUC_MISS_DIAG_CDN = 1

and T_DLY_1_FUC_MISS_DIAG timer is incremented

else LV_FUC_MISS_DIAG_CDN = 0

and T_DLY_1_FUC_MISS_DIAG = 0

endif

Configuration byte C_CONF_DIAGCP:

The fuel cap missing diagnosis is only running if C_CONF_DIAGCP ≠ 0

(CONF_DIAGCP > 0, not passive).

B.10.8.1.1 Fuel Cap Diagnosis sequence

- *Determination of the pressure mean value (DTP_MV_FUC_MISS_DIAG) before starting the diagnosis*

After C_T_DLY_1_FUC_MISS_DIAG has elapsed, DTP_MV_FUC_MISS_DIAG = DTP_MV This is the pressure before starting the FUC_MISS diagnosis:

If LV_FUC_MISS_DIAG_CDN = 1
and C_T_DLY_1_FUC_MISS_DIAG has elapsed
then DTP_MV_FUC_MISS_DIAG = DTP_MV
else no determination of DTP_MV_FUC_MISS_DIAG
endif

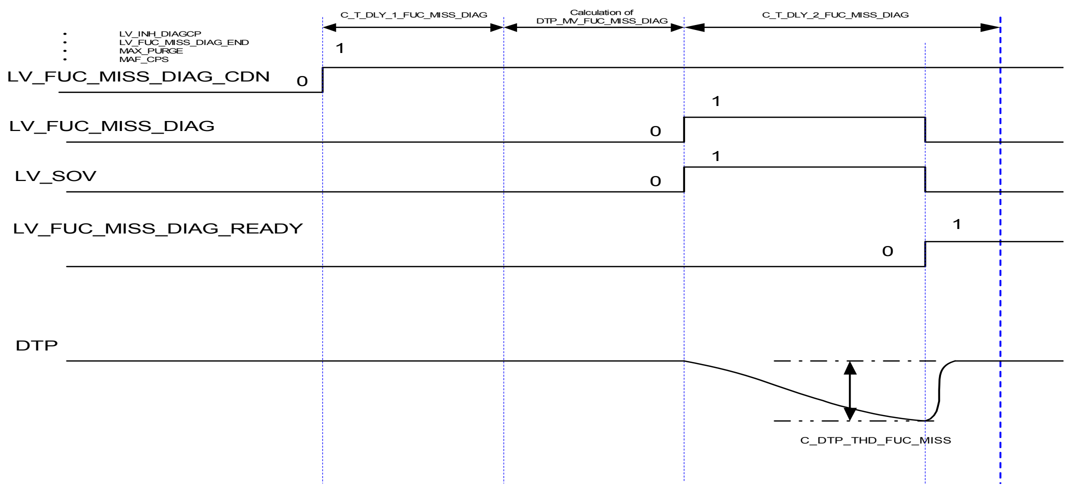
- *Start of “Missing Fuel Cap Diagnosis”*

If LV_FUC_MISS_DIAG_CDN = 1
and C_T_DLY_1_FUC_MISS_DIAG is reached
then LV_FUC_MISS_DIAG = 1
and T_DLY_2_FUC_MISS_DIAG timer is incremented
and LV_SOV_REQ = 1
else LV_FUC_MISS_DIAG = 0
and T_DLY_2_FUC_MISS_DIAG = 0
and LV_SOV_REQ = 0
endif

- *End of “Missing Fuel Cap Diagnosis” (one FUC_MISS diagnosis cycle)*

If LV_FUC_MISS_DIAG = 1
and C_T_DLY_2_FUC_MISS_DIAG has elapsed
or DTP - DTP_MV_FUC_MISS_DIAG = DTP_DIF_FUC_MISS_DIAG ≤ C_DTP_THD_FUC_MISS
then LV_FUC_MISS_DIAG = 0
LV_SOV_REQ = 0
LV_FUC_MISS_DIAG_READY = 1
endif

B.10.8.1.2 Function Diagram:



B.10.8.1.3 Triggering a renewed “FUC_MISS Diagnosis” after a longer Ídle State Duration´

After a longer idle duration (e.g. refuelling with running engine), the FUC_MISS diagnosis algorithm is activated again (only, if multiple check is not running)

If LV_IS = 1
and VS = 0
} for longer than C_T_DLY_3_FUC_MISS_DIAG seconds
and LV_FUC_MISS_DIAG_END = 1 (Multiple check is not running)
then LV_FUC_MISS_DIAG_END = 0 (diagnosis algorithm is started again)
NR_CYC_FUC_MISS_DIAG = 0
LV_DIAG_CHK_MPL_FUC_MISS = 0
LV_FUC_MISS_DIAG_READY = 0
T_DLY_4_FUC_MISS_DIAG = 0
endif

B.10.8.1.4 Multiple Check:

Because of statistical reasons, the fuel cap diagnosis algorithm should be performed more times per driving cycle, if the first diagnosis cycle in driving cycle has been finished with a diagnostic error.

- **Activation of multiple check:**

If FUC_MISS_DIAG algorithm is completed
(LV_FUC_MISS_DIAG_READY = 0→1)
then NR_CYC_FUC_MISS_DIAG = NR_CYC_FUC_MISS_DIAG + 1
endif

If LV_FUC_MISS_DIAG_READY = 1
and LV_ERR_FUC_MISS_1 = 1
then LV_DIAG_CHK_MPL_FUC_MISS = 1
endif

If LV_DIAG_CHK_MPL_FUC_MISS = 1
and LV_FUC_MISS_DIAG_READY = 1
and NR_CYC_FUC_MISS_DIAG < C_NR_CYC_FUC_MISS_DIAG_MAX
then LV_FUC_MISS_DIAG_READY = 0
else LV_FUC_MISS_DIAG_END = 1
endif

- **Delay time between FUC_MISS_DIAG cycles during running multiple check**

If T_DLY_4_FUC_MISS_DIAG > 0
then T_DLY_4_FUC_MISS_DIAG is decremented
else no decrementation
endif

B.10.8.1.5 Cancelling because of too many interruptions (LV_SOV = 1 at missing fuel cap diagnosis)

If LV_FUC_MISS_DIAG = 1→ 0 (transition)
and LV_FUC_MISS_DIAG_CDN = 1→ 0 (transition)
then NR_CYC_INTR_FUC_MISS_DIAG = NR_CYC_INTR_FUC_MISS_DIAG + 1
if NR_CYC_INTR_FUC_MISS_DIAG >= C_NR_CYC_INTR_FUC_MISS_DIAG
then LV_FUC_MISS_DIAG_END = 1
(FUC_MISS diagnosis algorithm is finished)
endif

<u>Special treatment for Diagnostic Test Mode:</u>

If LV_FUC_MISS_DIAG_READY = 1 the value of DTP_DIF_FUC_MISS_DIAG is stored as DTP_DIF_FUC_MISS_DIAG_MOD_6 for mode 06-communication

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_1_FUC_MISS_DIAG	-	0...FFFFH	0.. 2621.2	0.04	s
Min. delay time after LV_FUC_MISS_DIAG_CDN = 1 to start FUC_MISS diagnosis LV_FUC_MISS_DIAG = 1					
C_T_DLY_2_FUC_MISS_DIAG	-	0...FFFFH	0...2621.2	0.04	s
Time for the duration of the FUC_MISS diagnosis					
C_T_DLY_3_FUC_MISS_DIAG	-	0...FFFFH	0...6553.5	0.1	s
Time for Idle State Duration due to trigger a next FUC_MISS diagnosis					
C_T_DLY_4_FUC_MISS_DIAG	-	0...FFFFH	0...6553.5	0.1	s
Time duration between two FUC_MISS diagnosis cycles					
C_MAF_CPS_MIN_FUC_MISS_DIAG	-	0...FFFFH	0...8	0.000122	kg/h
Minimum threshold for MAF_CPS					
C_MAF_CPS_MAX_FUC_MISS_DIAG	-	0...FFFFH	0...8	0.000122	kg/h
Maximum threshold for MAF_CPS					
C_VS_MIN_FUC_MISS_DIAG	-	0...FFH	0...255	1	km/h
Minimum VS threshold for starting the monitoring cycle					
C_NR_CYC_FUC_MISS_DIAG_MAX	-	0...FFH	0...255	1	-
Maximum number of FUC_MIS_DIAG cycles per driving cycle					
C_NR_CYC_INTR_FUC_MISS_DIAG	-	0...FFH	0...255	1	-
Maximum number of FUC_MISS_DIAG diagnosis interruptions					
C_N_MAX_FUC_MISS_DIAG	-	0...FFH	0...8160	32	rpm
Maximum engine speed for missing fuel cap diagnosis					
C_CL_MMV_FUC_MISS_DIAG	-	0...FFFFH	0...8	0.000122	-
Maximum canister load for starting missing fuel cap diagnosis					
C_AMP_AD_FUC_MISS_DIAG	-	0...FFFFH	0...5434	0.083	hpa
Altitude threshold for starting missing fuel cap diagnosis					
C_DTP_MAX_FUC_MISS_DIAG	-	8000H...7FFFH	-40.96...40.96	0.00125	hpa
Maximum DTP level for starting missing fuel cap diagnosis					
C_DTP_MIN_FUC_MISS_DIAG	-	8000H...7FFFH	-40.96...40.96	0.00125	hpa
Minimum DTP level for starting missing fuel cap diagnosis					
C_DTP_THD_FUC_MISS	1	8000H...FFFFH	-40.96...40.96	0.00125	hpa
Pressure threshold for "Fuel Cap Missing" detection					
C_PQ_MAX_FUC_MISS_DIAG	1	0…FFFFH	0…0.99609	0.0039	-
Pressure quotient for "Fuel Cap Missing" detection					

B.10.9 Total diagnosis time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DIAGCP	V	0...FFH	0...255	1	s
Total duration of evap. monitoring					

Input data:

STATE_DIAGCP			

The time from the beginning of the *´determination of the signal fluctuation´* to the end of the ´diagnosis´ is displayed by the ramcell T_DIAGCP.

The timer doesn' t run **IF** STATE_DIAGCP = T_DLY_3
= START
= PAS
= CLD

The timer is resetted at the beginning of the evap. monitoring (STATE_DIAGCP changes to PRS_DYN).

B.10.10 Function control, Configuration and State Display

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
STATE_DIAGCP	V/O	0...FFH	0...255	1	-
States during evaporative system monitoring					
STATE_INTR_DIAGCP	V	0...FFFFH	0...65535	1	-
Information byte at interruption of DIAGCP					
LV_DIAGCP_CHK_MPL	V	0...1H	0...1	1	-
Multiple check active					
NR_CYC_DIAGCP	V	0...FFH	0...255	1	-
Total number of completed DIAGCP Cycles					
NR_CYC_INTR_DIAGCP	V/O	0...FFH	0...255	1	-
Total number of DIAGCP Interruptions befor completing one cycle					
LV_DIAGCP_END	V	0...1H	0...1	1	-
Diagnosis completed for this driving cycle					

Input data:

LV_ERR_VAP_LEAK_1	LV_ERR_VAP_LEAK_2	LV_ERR_MEC_CPS	C_CONF_DIAGCP
LV_ERR_VAP_LEAK_10	LV_ERR_FUC_MISS		

Function control:
Because of statistical reasons, the leak detection algorithm should be performed more times per driving cycle, if the first diagnosis cycle per driving cycle was finished with a diagnostic error.
Should the algorithm be interrupted too much before completing one diagnostic cycle, it should be aborted until end of driving cycle.

Formula section:

Activation of Multiple Check

If DIAGCP algorithm completed
then NR_CYC_DIAGCP = NR_CYC_DIAGCP + 1
NR_CYC_INTR_DIAGCP = 0

If LV_ERR_VAP_LEAK_1 = 1 (Only very small leak detetcted)
and LV_ERR_VAP_LEAK_2 = 0
and LV_ERR_VAP_LEAK_10 = 0
and LV_ERR_FUC_MISS = 0
then LV_DIAGCP_CHK_MPL = 1 *(Multiple check activated)*
else **if** LV_ERR_VAP_LEAK_2 = 1
or LV_ERR_VAP_LEAK_10 = 1
or LV_ERR_FUC_MISS = 1
then LV_DIAGCP_CHK_MPL = 0 *(Multiple check deactivated)*
endif

If NR_CYC_DIAGCP < C_NR_CYC_DIAGCP_MAX
and LV_DIAGCP_CHK_MPL = 1
then enable new DIAGCP algorithm
else Function completed for this DC, STATE_DIAGCP = PAS, LV_DIAGCP_END = 1

Cancelling because of too many interruptions

If DIAGCP algorithm interrupted (at every jump to State T_DLY_3)
then NR_CYC_INTR_DIAGCP = NR_CYC_INTR_DIAGCP + 1

If NR_CYC_INTR_DIAGCP = C_NR_CYC_INTR_DIAGCP_MAX
then abort DIAGCP algorithm for this DC, STATE_DIAGCP = CLD

If DIAGCP algorithm completed
then reset NR_CYC_INTR_DIAGCP

Initialisation:

NR_CYC_DIAGCP , NR_CYC_INTR_DIAGCP, LV_DIAGCP_CHK_MPL and LV_DIAGCP_END are set to 0 at Start of a new Driving cycle

Configuration byte C_CONF_DIAGCP:

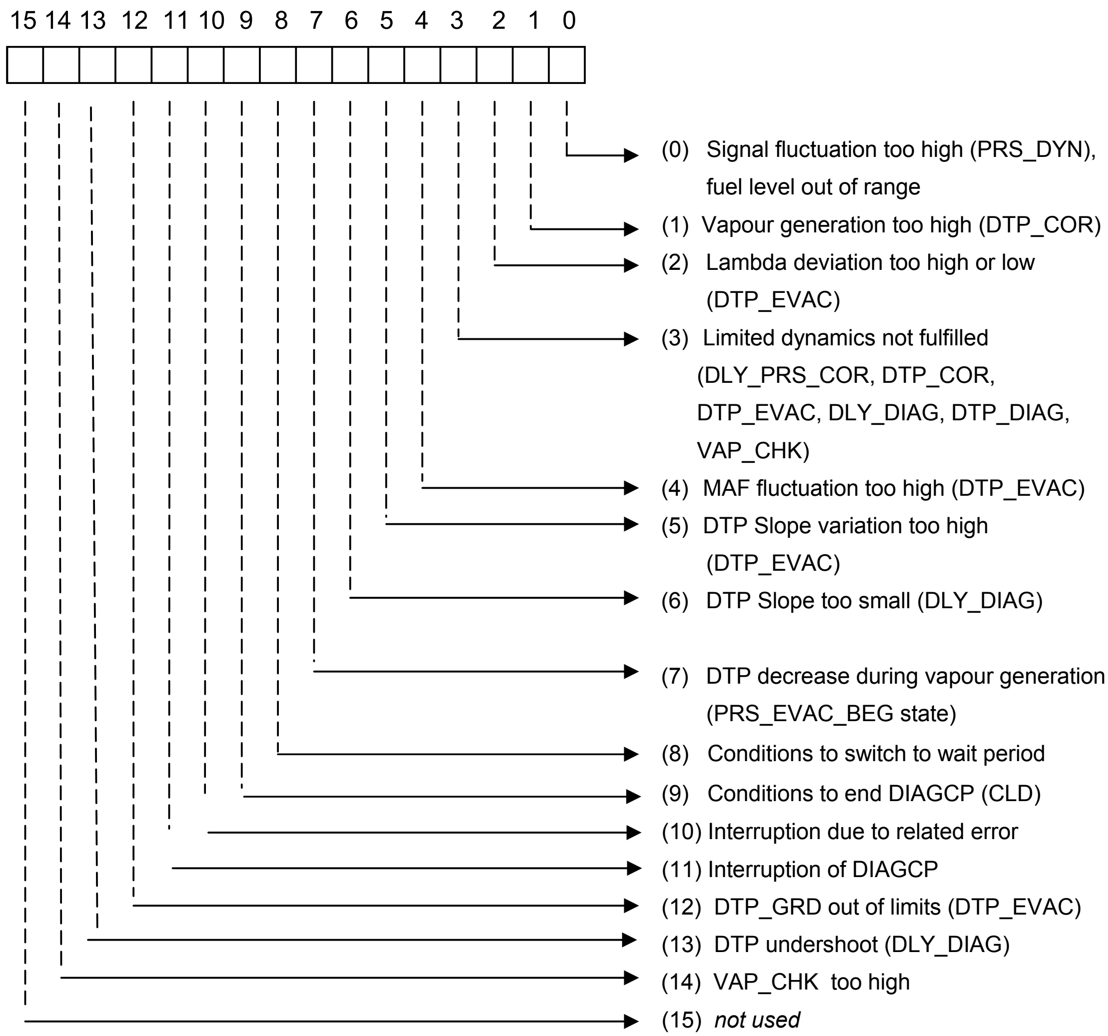
C_CONF_DIAGCP is used to configure the DIAGCP-Algorithm (for application purposes).

0: passive
1: until NR_CYC_DIAGCP_MAX >= C_NR_CYC_DIAGCP_MAX
(LV_DIAGCP_CHK_MPL = 1)
2: Continuous run (algorithm is performed continuously only interrupted by the wait period)

DIAGCP States display:

STATE_DIAGCP =	0 - ST	(START)
	1 - PDYN	(PRS_DYN)
	2 - CLSCPS	(CLOSE_CPS)
	3 - DLYSOV	(DLY_SOV)
	4 - DLYPCOR	(DLY_PRS_COR)
	5 - PCORBEG	(PRS_COR_BEG)
	6 - PCOR	(DTP_COR)
	7 - PEVBEG	(PRS_EVAC_BEG)
	8 - PEV	(DTP_EVAC)
	9 - DLYDIA	(DLY_DIAG)
	A - PDIA	(DTP_DIAG)
	B - T_DLY_3	
	C - ERRCLSCPS	(ERR_CLOSE_CPS)
	D - PAS	
	E - CLD	
	F - VAPCHK	(VAP_CHK)
	10 - DLY_VAP_CHK	

STATE_INTR_DIAGCP : Information at interruption of the DIAGCP algorithm:

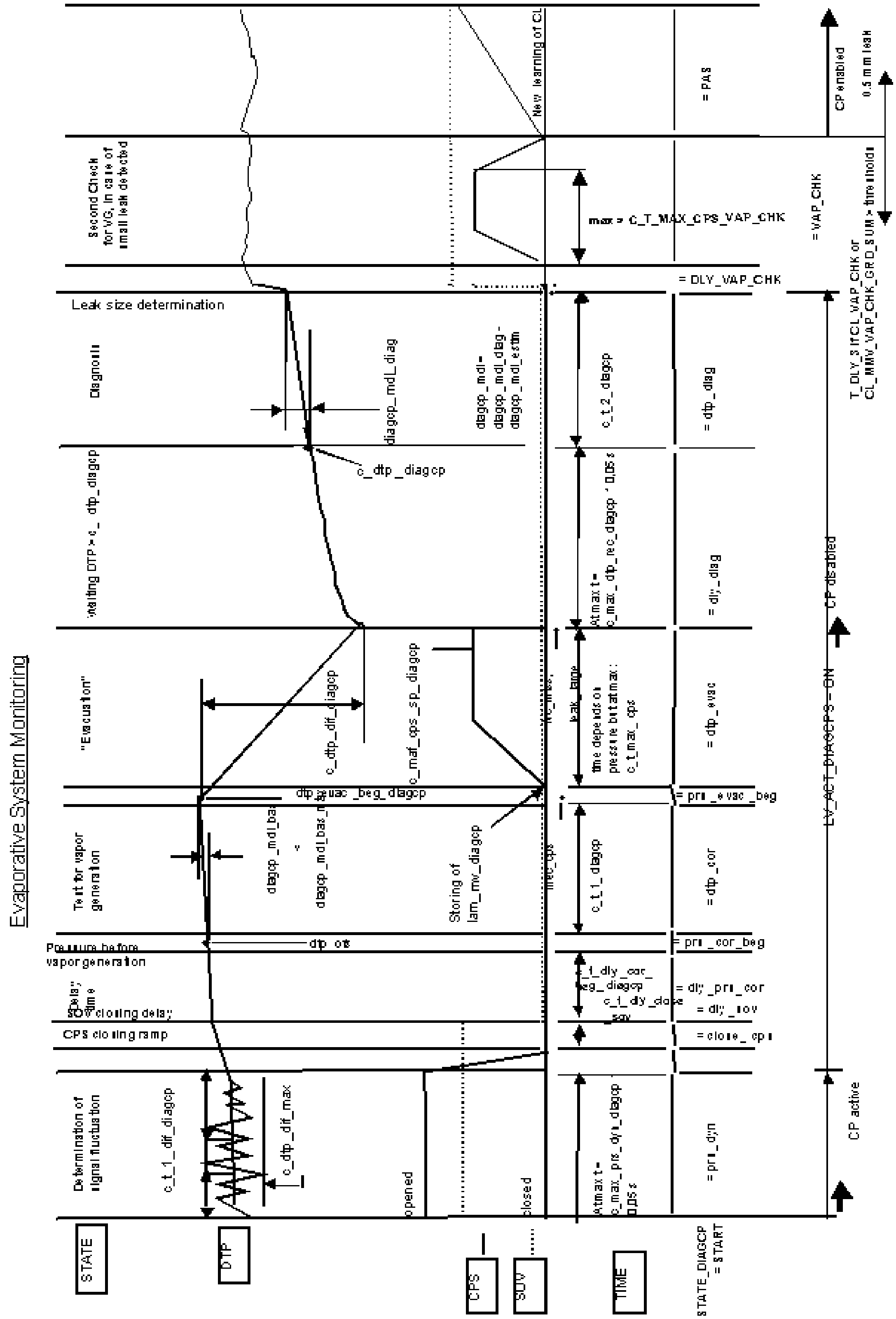


Remark:
It is possible that more than one bit is set at the same time, as some of the above mentioned states can happen at the same time.
(Example: If the `limited dynamics´ are not fulfilled then bit 3 and bit 11 is active)

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_NR_CYC_DIAGCP_MAX	-	0... FFH	0...255	1	-
Max number of DIAGCP cycles per driving cycle					
C_NR_CYC_INTR_DIAGCP_MAX	-	0... FFH	0...255	1	-
Max number of DIAGCP interruptions					

B.10.11 Algorithm overview


Evaporative System Monitoring
Determination of signal fluctuation
CPS closing ramp
SOV closing delay
Pressure before vapor generation
Test for vapor generation
"Evacuation"
Waiting DTP > c_dtp_diagcp
Diagnosis
Leak size determination
Second Check for VG, in case of small leak detected
STATE
DTP
CPS
SOV
TIME
STATE_DIAGCP = START
= prs_dyn
= close_cps
= dly_sov
= dly_prs_cor
= prs_cor_beg
= dtp_cor
= prs_evac_beg
= dtp_evac
= dly_diag
= dtp_diag
= DLY_VAP_CHK
= VAP_CHK
= PAS
c_t_1_dif_diagcp
c_dtp_dif_max
c_dtp_dif_diagcp
c_maf_cps_sp_diagcp
c_dtp_diagcp
diagcp_mdl_diag
dtp_cor
dtp_evac_beg_diagcp
c_t_1_diagcp
c_t_2_diagcp
opened
closed
New learning of CL
max = C_T_MAX_CPS_VAP_CHK
CP active
CP disabled
CP enabled
LV_ACT_DIAGCPS = ON

B.11 Error Management for Evaporative System Monitoring

B.11.1 Component diagnosis

B.11.1.1 Mechanical evaporative emission control valve (CPS) error

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MEC_CPS	V/O	0...1H	0...1	1	-
Boolean for mechanical CPS error (irreversible)					
LV_CDN_DIAG_MEC_CPS	V/O	0...1H	0...1	1	-
Diagnosis condition for mechanical cps error diagnosis (MEC_CPS)					
ERR_SYM_MEC_CPS	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for MEC_CPS diagnosis					
LV_END_DIAG_MEC_CPS	V/O	0...1H	0...1	1	-
End of mechanical cps error diagnosis (MEC_CPS)					
CTR_ABC_CPS_MEC	V/O	0...FFH	0...255	1	-
anti bounce counter of diagnosis MEC_CPS					

Input data:

DTP_DIF_COR	STATE_DIAGCP	EOL_STATE_DIAGCP	

FUNCTION DESCRIPTION:

General information:

The mechanical CPS error is represented by the symptom **MEC_CPS**

Frezze frame, DTC, error code number, symptom number, CARB-Info and Data for MIL-Management see chapters for general Diagnosis informations

Description:

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
│ │ │ └─> -    CPS stucks in open position        (= SYM_0)
│ │ └───> -                                       (= SYM_1)
│ └─────> -                                       (= SYM_2)
└───────> -                                       (= SYM_3)
```

Remark: Calculation of LV_END_DIAG_MEC_CPS see generic calculation “End of diagnosis” in anti bounce algorithm.

Application conditions:

Initialisation: at new driving cycle

- LV_ERR_MEC_CPS = 0

Activation:

⇒ Only during ´test of vapour generation" the diagnosis for the symptom **MEC_CPS** is available.

⇒ The symptom MEC_CPS is handled by statistical evaluation

If STATE_DIAGCP = 6H (DTP_COR state)
and LV_ERR_MEC_CPS = 0
then diagnosis for the symptom **MEC_CPS** is available
LV_CDN_DIAG_MEC_CPS = 1
else diagnosis for the symptom **MEC_CPS** is not available
LV_CDN_DIAG_MEC_CPS = 0
endif

Formula section:

If DTP_DIF_COR $\leq$ C_DTP_DIF_CPS_DIAGCP
then symptom **MEC_CPS** is active
ERR_SYM_MEC_CPS = SYM_0
One increment anti-bounce counter with C_ABC_INC_CPS_MEC per diagnosis

  If anti-bounce counter reaches C_ABC_MAX_CPS_MEC
  or EOL_STATE_DIAGCP <> EOL_PAS
  then LV_ERR_MEC_CPS = 1 (irreversible)
  LV_END_DIAG_MEC_CPS = 1
  endif

else symptom **MEC_CPS** is not active
  if LV_CDN_DIAG_MEC_CPS changes 1 -> 0
  and { STATE_DIAGCP = 7H (PRS_EVAC_BEG state)
  or STATE_DIAGCP = 8H (DTP_EVAC state) }
  then ERR_SYM_MEC_CPS = NO_SYM

    If CTR_ABC_CPS_MEC > 0
    then One decrement anti-bounce counter with 1
    else LV_ERR_MEC_CPS = 0
    LV_END_DIAG_MEC_CPS = 1
    endif
  endif
endif

Application assistances:

In the case of error, the following functions are concerned:
- Evaporative Emission Contol: disabled
- Leak detetction : disabled indirectly by disabled EVAP function
- Diagnosis MEC_CPS : disabled indirectly by disabled leak detetction
- Misfire detection/segment adaptation: disabled
- Secondary air System diagnosis: disabled
- Fuel system diagnosis: disabled
- Lambda sensor upstream/downstream signal diagnosis/Plausibility in PUC: disabled
- Lambda adaptation: disabled

- Fuel trim regulation: disabled
- idle air demand adaptation: disabled

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_DIF_CPS_DIAGCP	1	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure threshold for mechanical CPS error					
C_ABC_INC_CPS_MEC	-	0...FFH	0...255	1	-
Maximum value of frequency counter for MEC_CPS					
C_ABC_MAX_CPS_MEC	-	0...FFH	0...255	1	-
Maximum value of test cycle counter for CPS error					

B.11.1.2 Charcoal-canister shutoff valve (SOV) stuck

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_MEC_SOV	V/O	0...1H	0...1	1	-
Diagnosis condition for mechanical SOV error diagnosis (MEC_SOV)					
LV_END_DIAG_MEC_SOV	V/O	0...1H	0...1	1	-
End of mechanical SOV error diagnosis (MEC_SOV)					
T_DTP_MES	V	0...FFH	0...10.2	0.04	sec
Counter for the detection of DTP short circuit to ground error					
LV_ERR_MEC_SOV	V/O	0...1H	0...1	1	-
Boolean for SOV stucks open					
ERR_SYM_MEC_SOV	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for MEC_SOV diagnosis					

Input data:

DTP	VB	LV_DTP_PLAUS_READY	LV_ERR_DTP
LV_ERR_SOV	LV_CDN_VB_OBD2		LV_SOV
LV_ST_END	C_VB_MIN_DIAGCP	LV_ERR_MEC_SOV	CONF_DIAGCP
V_DTP	C_V_DTP_MAX_DIAG	C_V_DTP_MIN_DIAG	
LV_DIAGCPS_ACT			

FUNCTION DESCRIPTION:

General information:

The purpose is to diagnose mechanical damage of the SOV. symptom **MEC_SOV.** The diagnosis is performed by anti-bouncing.

Description:

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
|  |  |  └─> -    SOV stucks in closed position        (= SYM_0)
|  |  └────> -                                         (= SYM_1)
```

Remark: Calculation of LV_END_DIAG_MEC_SOV see generic calculation “End of diagnosis” in anti bounce algorithm.

Application conditions:

Activation:

```
If      LV_ST_END = 1
and     VB > C_VB_MIN_DIAGCP
and     CONF_DIAGCP > 0
and     LV_DTP_PLAUS_READY = 1      (DTP Plausibility check performed)
and     LV_ERR_DTP = 0              (no DTP error present)
and     LV_ERR_SOV = 0              (no electr. SOV error present)
and     LV_CDN_VB_OBD2 = 1
and     LV_SOV = 0
then    diagnosis for the symptom MEC_SOV is available
        LV_CDN_DIAG_MEC_SOV = 1
else    diagnosis for the symptom MEC_SOV is not available
        LV_CDN_DIAG_MEC_SOV = 0
endif
```

Formula section:

Delay time counter calculation before MEC_SOV detection :

```
If      DTP < C_DTP_MIN_SOV
then    if      T_DTP_MES < C_T_DTP_MES
        then    if      V_DTP > C_V_DTP_MAX_DIAG
                or      V_DTP < C_V_DTP_MIN_DIAG
                then    Stop  T_DTP_MES
                else    T_DTP_MES increase
                endif
        else    Stop  T_DTP_MES
        endif
else    Reset  T_DTP_MES
endif
```

MEC_SOV failure detection :

```
if       DTP < C_DTP_MIN_SOV
and      T_DTP_MES >= C_T_DTP_MES
and      LV_ST_END = 1
and      LV_DIAGCPS_ACT = 0
and      LV_ERR_DTP = 0 (no tank pressure sensor error)
and      C_V_DTP_MIN_DIAG <= V_DTP <= C_V_DTP_MAX_DIAG
         (no DTP electrical sensor error)
then     'Canister shut off valve (SOV) stuck' is detected
         and     symptom MEC_SOV is active, anti bounce counter is incremented
                 (with C_ABC_INC_MEC_SOV)
         and     ERR_SYM_MEC_SOV = SYM_0
         and     LV_ERR_MEC_SOV = 1 (after debounce)
else if  DTP > C_DTP_NOM_SOV
then     the anti bounce counter of MEC_SOV is decremented
         and     ERR_SYM_MEC_SOV = NO_SYM
         and     LV_ERR_MEC_SOV = 0 (after rebounce)
endif
```

Remark : For the SOV stuck diagnosis formular, refer to 'DTP electrical diagnosis and SOV stuck diagnosis' chapter.

Increment of debounce counter: C_ABC_INC_MEC_SOV
Maximum value of debounce counter: C_ABC_MAX_MEC_SOV

Additional measures:

Disable evaporative emission control for the time the error is present and
disable evaporative emission control diagnosis for this driving cycle

Application assistances:

In the case of error , the following functions are concerned:

- Evaporative Emission Control: disabled
- Leak detetction : disabled indirectly by disabled EVAP function
- Diagnosis MEC_CPS : disabled indirectly by disabled leak detetction

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DTP_MIN_SOV	-	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure threshold for SOV jammed in closed position					
C_ABC_INC_MEC_SOV	-	0...FFH	0...255	1	-
Increment debounce counter					
C_ABC_MAX_MEC_SOV	-	0...FFH	0...255	1	-
Maximum value debounce counter					
C_T_DTP_MES	1	0...FFH	0...10.2	0.04	sec
Counter for the detection of DTP short circuit to battery or line break					
C_DTP_NOM_SOV	-	8000...7FFFH	-40.96...40.96	0.00125	hPa
Pressure threshold for SOV jammed failure healing					

B.11.2 Leakage in the EVAP system

B.11.2.1 Small leak

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VAP_LEAK_1	V/O	0...1H	0...1	1	-
Flag indicating a small leak in the evap system (reversible)					
LV_CDN_DIAG_VAP_LEAK_1	V/O	0...1H	0...1	1	-
Diagnosis condition for 05mm error detection					
ERR_SYM_VAP_LEAK_1	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for 05mm error					
LV_END_DIAG_VAP_LEAK_1	V/O	0...1H	0...1	1	-
End of diagnosis for 05mm error detection					
LV_ERR_VAP_LEAK_2	V/O	0...1H	0...1	1	-
Flag indicating a small leak in the evap system (reversible)					
LV_CDN_DIAG_VAP_LEAK_2	V/O	0...1H	0...1	1	-
Diagnosis condition for 1mm error detection					
ERR_SYM_VAP_LEAK_2	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for 1mm error					
LV_END_DIAG_VAP_LEAK_2	V/O	0...1H	0...1	1	-
End of diagnosis for 1mm error detection					
LV_ERR_VAP_LEAK_1_MIL	V/O	0...1H	0...1	1	-
Flag indicating a small leak in the evap system (reversible)					
LV_CDN_DIAG_VAP_LEAK_1_MIL	V/O	0...1H	0...1	1	-
Diagnosis condition for 05mm error detection					
ERR_SYM_VAP_LEAK_1_MIL	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for 05mm error					
LV_END_DIAG_VAP_LEAK_1_MIL	V/O	0...1H	0...1	1	-
End of diagnosis for 05mm error detection					
SUM_DIAG_VAP_LEAK_1	V/O	0...FFH	0...255	1	[-]
Number of all small leak diagnosis (good and bad results)					
SUM_ERR_VAP_LEAK_1	V	0...FFH	0...255	1	[-]
Number of detected leak diagnosis					
RATIO_ERR_VAP_LEAK_1	V	0...FFH	0...0.99609	3.9063e-3	[-]
Ratio between SUM_DIAG_VAP_LEAK_1 and SUM_ERR_VAP_LEAK_1					

Input data:

FAC_DIAM_DIAGCP	LV_DTP_SLOP	LV_VAP_CHK_END	C_T_2_DIAGCP
STATE_DIAGCP	C_DTP_MAX_REC_DIAGCP	DTP_MV	C_DTP_DIAGCP
CTR_DC_x	CONF_DIAGCP	LV_END_DIAG_VAP_LEAK_1	LV_END_DIAG_VAP_LEAK_1_MIL
EOL_STATE_DIAGCP			

B.11.2.1.1 Evaporative emission control diagnostic error – 0.5 mm (not MIL relevant)

General information:

The symptom name is **LEAK_1 (0.5mm)**. The diagnosis is performed by statistics. The error detection is “hard”. The error is detected based on the statistic result when the leakage diameter (FAC_DIAM_DIAGCP) is within certain limits.

For VAP_LEAK_2 (1.0mm) MIL management during VAP_LEAK_1 diagnosis, software refers to DTC (P0442 : small leak).

Description:

Error-symtoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ └─> -   VAP_LEAK_1      (= SYM_0)
│ │ └───> -                   (= SYM_1)
│ └─────> -                   (= SYM_2)
```

Application conditions:

Initialisation: at new driving cycle and at reset

- LV_ERR_VAP_LEAK_1 = 0
- all 0 for VAP_LEAK_1 diagnosis
- SUM_DIAG_VAP_LEAK_1 = 0
- SUM_ERR_VAP_LEAK_1 = 0

Activation:

If STATE_DIAGCP = 1 – 10H (leak diagnosis); 15-16H (VAP_CHK)
then diagnosis for the symptom **VAP_LEAK_1** is available
LV_CDN_DIAG_VAP_LEAK_1 = 1
else diagnosis for the symptom **VAP_LEAK_1** is not available
LV_CDN_DIAG_VAP_LEAK_1 = 0
endif

Formula section:

If C_T_2_DIAGCP is elapsed **or** LV_VAP_CHK_END = 1
or (STATE_DIAGCP = DLY_DIAG (9H)
and DTP_MV <= C_DTP_DIAGCP
and C_DTP_MAX_REC_DIAGCP is reached)
(state DTP_DIAG is finished or state VAP_CHK is finished or 'leak-free')

then

If { (C_DIAM_MAX_DIAGCP > FAC_DIAM_DIAGCP > C_DIAM_MIN_DIAGCP and
EOL_STATE_DIAGCP = EOL_PAS) // at normal test
or
(C_DIAM_MAX_DIAGCP_EOL > FAC_DIAM_DIAGCP > C_DIAM_MIN_DIAGCP_EOL and
EOL_STATE_DIAGCP <> EOL_PAS) } // at EOL test

then **if** LV_DTP_SLOP = 0
then symptom **VAP_LEAK_1** is active
ERR_SYM_VAP_LEAK_1 = SYM_0
LV_ERR_VAP_LEAK_1 = 1
LV_END_DIAG_VAP_LEAK_1 = 1
$SUM\_ERR\_VAP\_LEAK\_1_{(n)} = SUM\_ERR\_VAP\_LEAK\_1_{(n-1)}+1$
$SUM\_DIAG\_VAP\_LEAK\_1_{(n)} = SUM\_DIAG\_VAP\_LEAK\_1_{(n-1)}+1$
if SUM_ERR_VAP_LEAK_1 = 1
then ACTION_ERRM_StorePrevFrf (VAP_LEAK_1_MIL)

This action cause entry in the prestored freeze frame for VAP_LEAK_1_MIL

endif
if CTR_DC_x (for VAP_LEAK_2) > 0
then symptom **VAP_LEAK_2** is active
ERR_SYM_VAP_LEAK_2 = SYM_0
LV_ERR_VAP_LEAK_2 = 1
LV_END_DIAG_VAP_LEAK_2 = 1
else LV_ERR_VAP_LEAK_2 = 0 (reset error bit; if already set)
endif

else the system switches to the “wait period”

endif

elseif { (FAC_DIAM_DIAGCP <= C_DIAM_MIN_DIAGCP and EOL_STATE_DIAGCP = EOL_PAS)
or
(FAC_DIAM_DIAGCP <= C_DIAM_MIN_DIAGCP_EOL and EOL_STATE_DIAGCP <> EOL_PAS) }
// EOL_STATE_DIAGCP <> EOL_PAS means EVAP monitoring was done at EOL test

then symptom **VAP_LEAK_1** is not active
System is "leak free"
ERR_SYM_VAP_LEAK_1 = NO_SYM
LV_ERR_VAP_LEAK_1 = 0 (reset error bit; if already set)
LV_ERR_VAP_LEAK_2 = 0 (reset error bit; if already set)
LV_END_DIAG_VAP_LEAK_1 = 1
SUM_DIAG_VAP_LEAK_1$_{(n)}$ = SUM_DIAG_VAP_LEAK_1$_{(n-1)}$+1
if SUM_DIAG_VAP_LEAK_1 = 1
then LV_END_DIAG_VAP_LEAK_1_MIL = 1
endif

endif

endif

<u>Special treatment for readyness generation:</u>

If STATE_DIAGCP changes 10 → 11 (State DTP_DIAG finished without interruptions)
or STATE_DIAGCP changes 15 → 11 (State VAP_CHK finished without interruptions)

then symptom **VAP_LEAK_1** is **available** and **inactive**
and readyness detection for this symptom has been performed in this driving cycle

Application assistances:

No other functions concerned

B.11.2.1.2 Evaporative emission control diagnostic error – 0.5 mm (MIL relevant)

Description:

Error-symtoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ └─> -   VAP_LEAK_1_MIL          (= SYM_0)
│ │ └───> -                           (= SYM_1)
│ └─────> -                           (= SYM_2)
```

Application conditions:

Initialisation: at new driving cycle and at reset

– ACTION_ERRM_NoFilterReset(IN<VAP_LEAK_1_MIL>, OUT<LV_ERR_VAP_LEAK_1_MIL>)

This action erases filter data (LV_CDN_DIAG_VAP_LEAK_1_MIL=0, LV_ERR_VAP_LEAK_1_MIL=0, ERR_SYM_VAP_LEAK_1_MIL=NO_SYM)

Activation:

If CONF_DIAGCP > 0
and **{** LV_DC = 1 → 0 (before error management treatments done at LV_DC = 1 → 0)
or SUM_ERR_VAP_LEAK_1 >= C_DIAGCP_SUM_STC_MIN
or EOL_STATE_DIAGCP = EOL_RDY **}**
then diagnosis for the symptom **VAP_LEAK_1_MIL** is available
LV_CDN_DIAG_VAP_LEAK_1_MIL = 1
else diagnosis for the symptom **VAP_LEAK_1_MIL** is not available
LV_CDN_DIAG_VAP_LEAK_1_MIL = 0
endif

Formula section:

If SUM_DIAG_VAP_LEAK_1 = 0
then no ERRM calculation
else RATIO_ERR_VAP_LEAK_1 =
SUM_ERR_VAP_LEAK_1 / SUM_DIAG_VAP_LEAK_1

If RATIO_ERR_VAP_LEAK_1 >= C_MAX_RATIO_ERR_VAP_LEAK_1
then symptom **VAP_LEAK_1_MIL** is active
ERR_SYM_VAP_LEAK_1_MIL = SYM_0
LV_ERR_VAP_LEAK_1_MIL = 1
LV_END_DIAG_VAP_LEAK_1_MIL = 1
else symptom **VAP_LEAK_1_MIL** is not active
ERR_SYM_VAP_LEAK_1_MIL = NO_SYM
LV_ERR_VAP_LEAK_1_MIL = 0
LV_END_DIAG_VAP_LEAK_1_MIL = 1
endif

endif

Application assistances:

No other functions concerned

B.11.2.1.3 Evaporative emission control diagnostic error – 1.0 mm

General information:

The symptom name is **VAP_LEAK_2 (1mm)**. The error is detected as soon as the leakage diameter (FAC_DIAM_DIAGCP) exceeds a limit. And no more leak check will be performed at the same driving cycle after 1.0 mm leak is detected.

For VAP_LEAK_1 (0.5mm) MIL management during VAP_LEAK_2 diagnosis, software refers to DTC (P0456 : very small leak).

Description:

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
 │ │ │ └─> -    VAP_LEAK_2          (= SYM_0)
 │ │ └───> -                        (= SYM_1)
```

Application conditions:

Initialisation: at new driving cycle and at reset

- LV_ERR_VAP_LEAK_2 = 0
- all 0 for VAP_LEAK_2 diagnosis

Activation:

If STATE_DIAGCP = 1 – 10H (leak diagnosis); 15 – 16H (VAP_CHK)
then diagnosis for the symptom **VAP_LEAK_2** is available
LV_CDN_DIAG_VAP_LEAK_2 = 1
else diagnosis for the the symptom **VAP_LEAK_2** is not available
LV_CDN_DIAG_VAP_LEAK_2 = 0
endif

Formula section:

If C_T_2_DIAGCP is elapsed **or** LV_VAP_CHK_END = 1
or (STATE_DIAGCP = DLY_DIAG (9H)
and DTP_MV <= C_DTP_DIAGCP
and C_DTP_MAX_REC_DIAGCP is reached)
(state DTP_DIAG is finished or state VAP_CHK is finished or 'leak-free')
then
If {(FAC_DIAM_DIAGCP >= C_DIAM_MAX_DIAGCP and EOL_STATE_DIAGCP = EOL_PAS)

```
                or
                ( FAC_DIAM_DIAGCP >= C_DIAM_MAX_DIAGCP_EOL and EOL_STATE_DIAGCP <> EOL_PAS )}
                // EOL_STATE_DIAGCP <> EOL_PAS means EVAP monitoring was done at EOL test
        then    symptom VAP_LEAK_2 is active

                if      LC_CONF_SLOP_VAP_LEAK_2 = 1
                then    if      LV_DTP_SLOP = 0
                        then    ERR_SYM_VAP_LEAK_2 = SYM_0
                                LV_ERR_VAP_LEAK_2 = 1
                                LV_END_DIAG_VAP_LEAK_2 = 1
                                LV_END_DIAG_VAP_LEAK_1 = 1
                                LV_END_DIAG_VAP_LEAK_1_MIL = 1
                                if      CTR_DC_x (for VAP_LEAK_1_MIL) > 0
                                then    symptom VAP_LEAK_1_MIL is active
                                        ERR_SYM_VAP_LEAK_1_MIL = SYM_0
                                        LV_ERR_VAP_LEAK_1_MIL = 1
                                        LV_END_DIAG_VAP_LEAK_1_MIL =1
                                endif
                        else    the system switches to the 'wait period'
                        endif
                else
                then    ERR_SYM_VAP_LEAK_2 = SYM_0
                        LV_ERR_VAP_LEAK_2 = 1
                        LV_END_DIAG_VAP_LEAK_2 = 1
                        LV_END_DIAG_VAP_LEAK_1 = 1
                        LV_END_DIAG_VAP_LEAK_1_MIL = 1
                        if      CTR_DC_x (for VAP_LEAK_1_MIL) > 0
                        then    symptom VAP_LEAK_1_MIL is active
                                ERR_SYM_VAP_LEAK_1_MIL = SYM_0
                                LV_ERR_VAP_LEAK_1_MIL = 1
                                LV_END_DIAG_VAP_LEAK_1_MIL =1
                        endif
                endif

        else    symptom VAP_LEAK_2 is not active
                ERR_SYM_VAP_LEAK_2 = NO_SYM
                LV_ERR_VAP_LEAK_2 = 0           (reset error bit; if already set)
                LV_END_DIAG_VAP_LEAK_2 = 1
        endif
endif
```

<u>Special treatment for readyness generation:</u>

If STATE_DIAGCP changes 10 → 11 (State DTP_DIAG finished without interruptions)
then symptom **VAP_LEAK_2** is **available** and **inactive**
and readyness detection for this symptom has been performed in this driving cycle

Application assistances:

No other functions concerned

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_DIAM_MIN_DIAGCP	-	00...FFH	0...4.127	16.184e-3	mm
Minimum threshold for leakage detection					
C_DIAM_MAX_DIAGCP	-	00...FFH	0...4.127	16.184e-3	mm
Maximum threshold for leakage detection					
C_DIAM_MIN_DIAGCP_EOL	-	00...FFH	0...4.127	16.184e-3	mm
Minimum threshold for leakage detection at EOL test					
C_DIAM_MAX_DIAGCP_EOL	-	00...FFH	0...4.127	16.184e-3	mm
Maximum threshold for leakage detection at EOL test					
C_DIAGCP_SUM_STC_MIN	1	0...FFH	0...255	1	[-]
Minimum number of diagnosis cycles to activate MIL statistic					
C_MAX_RATIO_ERR_VAP_LEAK_1	1	0...FFH	0...0.99609	3.9063e-3	[-]
Maximum VAP_LEAK_1 (0.5mm) threshold to activate MIL					
LC_CONF_SLOP_VAP_LEAK_2	-	0...1h	0...1	1	-
Activate double slop abortion condition for 1.0mm leak detection : YES/NO					

B.11.2.2 Large leak

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_VAP_LEAK_10	V/O	0...1H	0...1	1	-
Flag indicating a large leak (reversible)					
LV_CDN_DIAG_VAP_LEAK_10	V/O	0...1H	0...1	1	-
Diagnosis condition for large leak error detection					
ERR_SYM_VAP_LEAK_10	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for large leak error					
LV_END_DIAG_VAP_LEAK_10	V/O	0...1H	0...1	1	-
End of diagnosis for large leak error detection					

Input data:

DTP_DIF_ACT	C_DTP_DIF_DIAGCP	C_DTP_DIF_MIN_DIAGCP	C_T_MAX_CPS
STATE_DIAGCP	LV_END_DIAG_VAP_LEAK_1	LV_END_DIAG_VAP_LEAK_2	LV_END_DIAG_VAP_LEAK_1_MIL
V_DTP_MAX_DIAG	V_DTP_MIN_DIAG	C_V_DTP_THD_DIAG	

General Information:

The symptom name is **VAP_LEAK_10 (large Leak)**.

Error detection:

The pressure does not drop at least C_DTP_DIF_DIAGCP, but more than C_DTP_DIF_MIN_DIAGCP, during the maximum duration or the evacuation period (C_T_MAX_CPS).

Description:

Error-symtoms are defined to this diagnosis function as following :

```
□□□□
 |  |  |  └> -                                  (= SYM_0)
 |  |  └──> -                                   (= SYM_1)
 |  └────> -     VAP_LEAK_10                    (= SYM_2)
```

Application conditions:

Initialisation: at new driving cycle and at reset

- LV_ERR_VAP_LEAK_10 = 0
- all 0 for VAP_LEAK_10 diagnosis

Activation:

If STATE_DIAGCP = 1 – 9H
then diagnosis for the symptom **VAP_LEAK_10** is available
LV_CDN_DIAG_VAP_LEAK_10 = 1
else diagnosis for the symptom **VAP_LEAK_10** is not available
LV_CDN_DIAG_VAP_LEAK_10 = 0
endif

Formula section:

If C_T_MAX_CPS has elapsed (transition from 1s -> 0s)
then **If** DTP_DIF_ACT > C_DTP_DIF_DIAGCP
and DTP_DIF_ACT < C_DTP_DIF_MIN_DIAGCP
and V_DTP_MAX_DIAG – V_DTP_MIN_DIAG >= C_V_DTP_THD_DIAG
then symptom VAP_LEA**K_10** is active
ERR_SYM_VAP_LEAK_10 = SYM_2
LV_ERR_VAP_LEAK_10 = 1

LV_END_DIAG_VAP_LEAK_10 = 1

LV_END_DIAG_VAP_LEAK_1 = 1
LV_END_DIAG_VAP_LEAK_2 = 1
LV_END_DIAG_VAP_LEAK_1_MIL = 1

else symptom **VAP_LEAK_10** is not active
ERR_SYM_VAP_LEAK_10 = NO_SYM
LV_ERR_VAP_LEAK_10 = 0 (reset error bit; if already set)

LV_END_DIAG_VAP_LEAK_10 = 1

endif

endif

<u>Special treatment for readyness generation:</u>

If STATE_DIAGCP changes 8 → 9 (State DTP_EVAC passed)

then symptom **VAP_LEAK_10** is **available** and **inactive**
and readyness detection for this symptom has been performed in this driving cycle

Application assistances:

No other functions concerned

B.11.3 Fuel Cap Missing Error Management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FUC_MISS	V/O	0...1H	0...1	1	-
Boolean for Evap. system: Fuel cap missing (detected during evaporative system monitoring), reversible					
LV_CDN_DIAG_FUC_MISS	V/O	0...1H	0...1	1	-
Diagnosis condition for fuel cap missing (detected during evaporative system monitoring)					
ERR_SYM_FUC_MISS	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for fuel cap missing during evaporative system monitoring					
LV_END_DIAG_FUC_MISS	V/O	0...1H	0...1	1	-
End of diagnosis for fuel cap missing detection (during evaporative system monitoring)					
LV_ERR_FUC_MISS_1	V/O	0...1H	0...1	1	-
Flag indicating a missing fuel cap (out of EVAP algorithm), reversible					
LV_CDN_DIAG_FUC_MISS_1	V/O	0...1H	0...1	1	-
Diagnosis condition for fuel cap missing (detection out of EVAP algorithm)					
ERR_SYM_FUC_MISS_1	V/O	0H 1H	NO_SYM SYM_0	1	-
Error symptom for fuel cap missing (detection out of EVAP algorithm)					
LV_END_DIAG_FUC_MISS_1	V/O	0...1H	0...1	1	-
End of diagnosis for fuel cap missing detection (detection out of EVAP algorithm)					
LV_ERR_DIAGCP_2	V/O	0...1H	0...1	1	-
Global error flag for missing fuel cap (for interface reasons), reversible					

Input data:

DTP_DIF_ACT	DTP_DIF_FUC_MISS_DIAG	STATE_DIAGCP	C_DTP_THD_FUC_MISS
T_DLY_2_FUC_MISS_DIAG	LV_END_DIAG_VAP_LEAK_1	LV_END_DIAG_VAP_LEAK_2	LV_END_DIAG_VAP_LEAK_1_MIL
V_DTP_MAX_DIAG	V_DTP_MIN_DIAG	C_V_DTP_THD_DIAG	

B.11.3.1 Fuel cap missing detection during leak detection algorithm

General information:

The symptom name is **FUC_MISS.**

Error detection:

The pressure in the tank system DTP does not drop by at least C_DTP_DIF_MIN_DIAG during evacuation. Alternatives to a missing tank cap is a very large leakage, the evaporative emission control valve may stuck in the closed position or the DTP signal “clamps” (DTP plausibility error)

Description:

Error-symtoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> -   FUC_MISS (detection during EVAP)          (= SYM_0)
 │ │ └───> -                                             (= SYM_1)
```

Application conditions:

Initialisation: at new driving cycle

– LV_ERR_FUC_MISS = 0

Activation:

If STATE_DIAGCP = 8H (DTP_EVAC state)
then diagnosis for the symptom **FUC_MISS** is available
LV_CDN_DIAG_FUC_MISS = 1
else diagnosis for the symptom **FUC_MISS** is not available
LV_CDN_DIAG_FUC_MISS = 0
endif

Formula section:

If DTP_DIF_ACT >= C_DTP_DIF_MIN_DIAGCP
and V_DTP_MAX_DIAG – V_DTP_MIN_DIAG >= C_V_DTP_THD_DIAG
then symptom **FUC_MISS** is active
ERR_SYM_FUC_MISS = SYM_0
LV_ERR_FUC_MISS = 1
LV_ERR_DIAGCP_2 = 1 (global error bit is active)
LV_END_DIAG_FUC_MISS = 1
LV_END_DIAG_VAP_LEAK_1 = 1
LV_END_DIAG_VAP_LEAK_2 = 1
LV_END_DIAG_VAP_LEAK_1_MIL = 1

else symptom **FUC_MISS** is not active
ERR_SYM_FUC_MISS = NO_SYM
ERR_SYM_FUC_MISS_1 = NO_SYM
LV_ERR_FUC_MISS = 0 (reset error bit; if already set)
LV_ERR_FUC_MISS_1 = 0 (reset error bit; if already set)
LV_ERR_DIAGCP_2 = 0 (reset error bit; if already set)
LV_END_DIAG_FUC_MISS = 1
endif

Special treatment for readyness generation:

If STATE_DIAGCP changes 8 $\rightarrow$ 9 (State DTP_EVAC passed)
then symptom **FUC_MISS** and **FUC_MISS_1** is **available** and **inactive**
and readyness detection for these symptoms has been performed in this driving cycle

Application assistances:

No other functions concerned.

B.11.3.2 Fuel Cap Missing detection out of leak detection algorithm – Error Management

General information:

The symptom name is **FUC_MISS_1.**

Description:

Error-symtoms are defined to this diagnosis function as following :

```
┌─┬─┬─┬─┐
│ │ │ └─> -                (= SYM_0)
│ │ └───> FUC_MISS_1       (= SYM_1)
```

Application conditions:

Initialisation: at new driving cycle

– LV_ERR_FUC_MISS_1 = 0

Activation:

If T_DLY_2_FUC_MISS_DIAG is running
then diagnosis for symptom **FUC_MISS_1** is available
LV_CDN_DIAG_FUC_MISS_1 = 1
else diagnosis for symptom **FUC_MISS_1** is not available
LV_CDN_DIAG_FUC_MISS_1 = 0
endif

Formula section:

```
If       DTP_DIF_FUC_MISS_DIAG > C_DTP_THD_FUC_MISS
         (during T_DLY_2_FUC_MISS_DIAG is running)
then     LV_ERR_DIAGCP_2 = 1     (global error bit is active)

    if   LC_FMY_ACT_FUC_MISS_DIAG = 1
    then symptom FUC_MISS_1 is active
         ERR_SYM_FUC_MISS_1 = SYM_1
         LV_ERR_FUC_MISS_1 = 1
         LV_END_DIAG_FUC_MISS_1 = 1
    else symptom FUC_MISS_1 is not active
         ERR_SYM_FUC_MISS_1 = NO_SYM
              if    T_DLY_2_FUC_MISS_DIAG is elapsed
              then  LV_END_DIAG_FUC_MISS_1 = 1
              endif
    endif

else     symptom FUC_MISS_1 is inactive
         ERR_SYM_FUC_MISS_1 = NO_SYM
         LV_ERR_FUC_MISS_1 = 0
         if    T_DLY_2_FUC_MISS_DIAG is elapsed
         then  LV_END_DIAG_FUC_MISS_1 = 1
               if    LC_FMY_ACT_FUC_MISS_DIAG = 1
               then  symptom FUC_MISS is inactive
                     ERR_SYM_FUC_MISS = NO_SYM
                     LV_ERR_FUC_MISS = 0
                     LV_ERR_DIAGCP_2 = 0     (global error bit is inactive)
               endif
         endif
endif
```

Application hint for LC_FMY_ACT_FUC_MISS_DIAG:

⇒ LC_FMY_ACT_FUC_MISS_DIAG = 1: Failure memory and MIL evaluation are active
⇒ LC_FMY_ACT_FUC_MISS_DIAG = 0: Failure memory and MIL evaluation are not active

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_FMY_ACT_FUC_MISS_DIAG	-	0...1h	0...1	1	-
Activate failure memory: YES/NO					

B.12 Application incidences for Leak detection function

B.12.1 Conditions for EVAP-Monitoring

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Auflös.	Einh.
LV_STATE_CDN_DIAGCP	V/O	0…1H	0…1	1	-
conditions for DIAGCP fulfilled: YES/NO					
LV_DIAGCP_ACT	O	0…1H	0…1	1	-
only for interface reason: LAM-P-jump delay time of the catalyst purge function					
CPPWM_CP_DIAGCP	O	0...FFFFH	0…99.99	0.0015	%
only for interface reason : CPPWM output for CPS Driving					

Input data:

LV_INH_AMP_TEMP_FUEL_DIAGCP	TCO	LV_INH_DIAGCP	CL_MMV
CL_MMV_CLC_END	VS	LV_LSCL_i	DTP
LV_NORM_PURGE_END_1	TAM	LV_CPPWM_EXT_ADJ	AMP_AD
T_CLC_END	LV_IS	LV_SOV_ACT_EXT_ADJ	STATE_CP
LV_CP_SET_CLOSE	T_AST		LV_ST_END
TCO_ST	VB	STATE_SAV_DIAG	T_PL_DIAGCP
LV_INH_DIAGCP_EXT_FCT	FTL_MMV	STATE_DIAGCP	

Application conditions:

Recurrence: 40 ms (during EVAP diagnosis) / 1000 ms (Monitoring conditions), activated by main-function for EVAP-system-monitoring

Formula section:

Common conditions (LV_STATE_CDN_DIAGCP):

If LV_IS = 1
and VS ≤ C_VS_MAX_DIAGCP (No Fuel slosh due to vehicle movement)
and VB > C_VB_MIN_DIAGCP
and LV_LSCL_i = 1 (Lambda controller active)
and LV_CP_SET_CLOSE = 0 (No Oxygen sensor Diag. active)
and AMP_AD > C_AMP_AD_MIN_DIAGCP (altitude threshold)
and STATE_SAV_DIAG =NOT_ACTIVE (0H) (No SA Diagnosis active)
and LV_SOV_ACT_EXT_ADJ = 0 (No serv. tool intervention)
and LV_CPPWM_EXT_ADJ = 0 (No serv. tool intervention)
and (VS = 0 **or** STATE_DIAGCP < 8 **or** STATE_DIAGCP > A)
then
if STATE_DIAGCP = DLY_DIAG (9H)
and CL_MMV > C_CL_MMV_CLC_DIAGCP_MAX
then LV_STATE_CDN_DIAGCP = 0
else LV_STATE_CDN_DIAGCP = 1 **endif**
else LV_STATE_CDN_DIAGCP = 0
endif

⇒ *Activation:*

If LV_INH_DIAGCP = 0 (see next chapter : inhibit EVAP by errors)

and LV_STATE_CDN_DIAGCP = 1 (common conditions fulfilled, see above)

and LV_INH_AMP_TEMP_FUEL_DIAGCP = 0
and LV_ST_END = 1
and FUC_MISS_DIAG algorithm is not running

and TCO > C_TCO_MIN_DIAGCP
and TCO ≤ C_TCO_MAX_DIAGCP **and** TCO_ST ≤ IP_TCO_ST_DIAGCP
and TAM < C_TAM_MAX_DIAGCP (max. TAM threshold)
and TAM > C_TAM_MIN_DIAGCP

and FTL_MMV < C_FTL_MAX_DIAGCP

and STATE_CP = NO_PURGE (1H)

and LV_NORM_PURGE_END_1 = 1
and T_CLC_END ≤ C_T_CLC_PURGE_END_MAX
and T_AST > ID_T_AST_MIN_DIAGCP_2
and CL_MMV_CLC_END ≤ IP_CL_MMV_CLC_DIAGCP

or STATE_CP = MAX_PURGE (4H)

and CL_MMV ≤ IP_CL_MMV_CLC_DIAGCP
and T_AST > ID_T_AST_MIN_DIAGCP

and T_PL_DIAGCP = 0 (Time for purging in part load)
and C_DTP_MIN_DIAG < DTP < C_DTP_MAX_DIAG

and LV_INH_DIAGCP_EXT_FCT = 0 (External funcion inhibitation)

then *evaporative system monitoring is started*
endif

⇒ *Deactivation:*

The evaporative system monitoring is ***stopped for this Driving cycle***

If LV_INH_DIAGCP = 1
or TCO > C_TCO_MAX_DIAGCP
or TAM ≤ C_TAM_MIN_DIAGCP

} At least once per driving cycle (checked continuously)

then *evaporative system monitoring is stopped for this Driving cycle*

⇒ *Wait Period:*

The algorithm switches to *´Wait period´ (T_DLY_3),* if during activation of DIAGCP LV_STATE_CDN_DIAGCP = 0 is met.

Remark: T_DIAGCP (see chapter “Total diagnosis time”) is stopped during ´Wait period´.

Variables for interface-reasons:

LV_DIAGCP_ACT = 0

CPPWM_CP_DIAGCP= 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_AD_MIN_DIAGCP	-	0...FFFFH	0...5434	0.083	hPa
Altitude threshold for activation of DIAGCP					
C_DTP_MIN_DIAG	-	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Minimum DTP for activation of DIAGCP					
C_DTP_MAX_DIAG	-	8000H...7FFFH	-40.96...40.96	0.00125	hPa
Maximum DTP for activation of DIAGCP					
C_TAM_MIN_DIAGCP	-	0...FEH	-48...142.5	0.75	°C
Minimum TAM for activation of DIAGCP					
C_TAM_MAX_DIAGCP	-	0...FEH	-48...142.5	0.75	°C
Maximum TAM for activation of DIAGCP					
C_TCO_MIN_DIAGCP	-	0…FFH	-48...142.5	0.75	°C
Minimum TCO for activation of DIAGCP					
C_TCO_MAX_DIAGCP	-	0...FEH	-48...142.5	0.75	°C
Maximum TCO for activation of DIAGCP					
C_VB_MIN_DIAGCP	-	0...FEH	0....26	0.102	V
Minimum VB for DIAGCP					
C_VS_MAX_DIAGCP	-	0...FFH	0....255	1	km/h
Maximum VS for DIAGCP					
C_T_CLC_PURGE_END_MAX	-	0…FFH	0...255	1	s
Max. allowed time after setting of LV_NORM_PURGE_END_1 for activation of DIAGCP					
ID_T_AST_MIN_DIAGCP	4	0...FFFFH	0...6553.5	0.1	S
LDP_TCO_ST_ID_T_AST_MIN_DIAGCP	4	0...FEH	-48...142.5	0.75	°C
Time threshold for activation of DIAGCP, STATE_CP = MAX_PURGE					
ID_T_AST_MIN_DIAGCP_2	4	0...FFFFH	0...6553.5	0.1	S
LDP_TCO_ST_ID_T_AST_MIN_DIAGCP2	4	0...FEH	-48...142.5	0.75	°C
Time threshold for activation of DIAGCP; STATE_CP = MIN_PURGE					
C_FTL_MAX_DIAGCP	1	0...FFH	0...100	0.39	%
Maximum FTL_MMV for DIAGCP function activation (recommand 85%)					
C_CL_MMV_CLC_DIAGCP_MAX	-	0...FFFFH	0...8	0.000122	-
Maximum canister load for DIAGCP function activation (for small leak detection)					
IP_CL_MMV_CLC_DIAGCP	1*6	0...FFFFH	0...8	0.000122	-
LDPM_TAM	6	0...FEH	-48...142.5	0.75	°C
Maximum canister load for DIAGCP function activation					
IP_TCO_ST_DIAGCP	1*6	0...FEH	-48...142.5	0.75	°C
LDP_AMP_IP_TCO_ST_DIAGCP	6	0...FFFFH	0...5434	0.08292	hPa
TCO_ST condition for DIAGCP function activation					

B.12.2 Inhibition of Evaporative System Monitoring by errors

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAGCP	V/O	0...1H	0...1	1	-
Boolean for EVAP System diagnosis inhibitation by project specific conditions					

Input data:

LV_ERR_CPS	LV_ERR_FTL	LV_ERR_VS	LV_ERR_TCO
LV_ERR_TPS	LV_ERR_SOV	LV_ERR_MEC_SOV	LV_ERR_DTP
LV_ERR_FSD_i	LV_OFF_MTC_MON		
LV_ERR_TAM	LV_ERR_AMP	LV_ERR_IGC	LV_ERR_ISC
LV_ERR_IV[NC_CYL_NR]	LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_MAF	LV_ERR_MWSS
LV_ERR_ISA_i	LV_ERR_FSD_LAM_LIM_i	LV_MIS_STATE_A	LV_MIS_STATE_B
LV_ERR_CRK	LV_ERR_FTL	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAP

FUNCTION DESCRIPTION:

LV_INH_DIAGCP combines all project specific error bits and other special conditions to generate an input for the EVAP System Diagnosis Application conditions.

Application conditions:

Initialisation: at reset: LV_INH_DIAGCP = 0

Recurrence: 100 ms

Activation: In all engine states

Formula section:

```
If  LV_ERR_CPS = 1                  or   ; CPS electrical error
    LV_ERR_VS = 1                   or   ; Vehicle speed error
    LV_ERR_TCO = 1                  or   ; Coolant temperature error
    LV_ERR_SOV = 1                  or   ; SOV electrical error
    LV_ERR_MEC_SOV = 1              or   ; SOV mechanical  error
    LV_ERR_TPS = 1                  or   ; Throttle position sensor error
    LV_ERR_DTP = 1                  or   ; Tank pressure sensor error
    LV_ERR_FSD_i = 1                or   ; Lambda control error
    LV_ERR_TAM= 1                   or   ; Error TAM sensor
    LV_ERR_AMP = 1                  or   ; Error AMP sensor
    LV_OFF_MTC_MON = 1              or   ; ETC safety concept rel. error
    LV_ERR_IGC = 1                  or   ; Ignition Coil error
    LV_ERR_ISC = 1                  or   ; Idle speed control error
    LV_ERR_IV[NC_CYL_NR] = 1        or   ; Injection valve error
    LV_ERR_LS_UP[NC_CBK_EX_NR] =1   or   ; upstream O2 sensor signal error
    LV_ERR_MAF = 1                  or   ; MAF sensor error
    LV_ERR_MAP = 1                  or   ; MAP sensor error
    LV_ERR_MWSS =1                  or   ; Wheelspeed sensor error
    LV_ERR_ISA_i = 1                or   ; ISA error
    LV_ERR_FSD_LAM_LIM_i = 1        or   ; Lambda control error
```

LV_MIS_STATE_A = 1 **or** ; CARB A misfire
LV_MIS_STATE_B = 1 **or** ; CARB B misfire
LV_ERR_CRK = 1 **or** ; Crank Sensor error
LV_ERR_FTL = 1 **or** ; Fuel tank level sensor error
LV_ERR_LOAD_TPS_PLAUS = 1 ; MAF sensor implausible

then LV_INH_DIAGCP = 1
else LV_INH_DIAGCP = 0

B.12.3 Error and P-Code information

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Evap. system leak errors VAP_LEAK_1	VAP_LEAK_1	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
FAC_DIAM_DIAGCP
FAC_DIAGCP_MDL_ESTIM_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. system leak errors VAP_LEAK_1_MIL	VAP_LEAK_1_MIL	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
FAC_DIAM_DIAGCP
FAC_DIAGCP_MDL_ESTIM_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. System leak errors VAP_LEAK_2	VAP_LEAK_2	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
FAC_DIAM_DIAGCP
FAC_DIAGCP_MDL_ESTIM_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. system leak errors VAP_LEAK_10		SYM_0	*STC*
		SYM_1	
	VAP_LEAK_10	SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
DTP_DIF_ACT_ENVD
DTP_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. system leak errors FUC_MISS	FUC missing (detection during EVAP)	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
DTP_DIF_ACT_ENVD
DTP_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. System leak errors FUC_MISS_1		SYM_0	*STC*
	FUC missing	SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TAM
DTP_DIF_ACT_ENVD
DTP_ENVD
FAC_DIAGCP_VOL

Diagnostic	Symptom Description	Symptom	Filter type
Evap. System leak errors MEC_CPS	Stuck in open position	SYM_0	*STC*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in Failure Memory: TIA
DTP_DIF_COR_ENVD_H
DTP_DIF_COR_ENVD_L
DTP_ENVD

B.12.4 Interface to Rate Base Monitoring Module

B.12.4.1 Definition of common Bit 2 setting for all diagnosis performed during Evaporative System Monitoring

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_STATE_RBM_EVAP_BIT_2	O/V	0 ... 7H	0 ... 7H	1	-
Boolean to indicate that individual RBM conditions for EVAP Monitoring are fullfilled					
T_AST_EVAP_RBM	V	0...FFFFH	0...65535	1	s
Comulative time (depending on ambient temperature) since engine start					

Input data:

TAM	TCO_ST	TAM_ST	

FUNCTION DESCRIPTION

Description:

For the interface to RBM module, Bit 2 of STATE_RBM_xxx is used to show if individual conditions to detect an “RBM driving cycle” are set. In this chapter are defined conditions to set Bit2 for STATE_RBM of EVAP monitoring.

Application conditions:

Initialisation: LV_STATE_RBM_EVAP_BIT_2 = 0

T_AST_EVAP_RBM = 0

Recurrence: 1 sec.

Formula section:

Cumulative time (depending on ambient temperature) since engine start:

If C_TAM_MIN_EVAP_RBM <= TAM <= C_TAM_MAX_EVAP_RBM
(ambient temperature greater than or equal to 40 degrees Fahrenheit but less than or equal to 95 degrees Fahrenheit)

then **if** T_AST_EVAP_RBM >= C_T_AST_EVAP_RBM

then T_AST_EVAP_RBM keeps old value

else T_AST_EVAP_RBM = T_AST_EVAP_RBM + 1s

endif

else no incrementation of T_AST_EVAP_RBM

endif

Bit 2 - Coordination of STATE_RBM_xx for diagnosis running during EVAP monitoring:

If LV_STATE_RBM_EVAP_BIT_2 = 0

then **if** T_AST_EVAP_RBM >= C_T_AST_EVAP_RBM
(time since start is greater than or equal to 600 seconds)

and C_TCO_ST_MIN_EVAP_RBM <= TCO_ST <= C_TCO_ST_MAX_EVAP_RBM
(engine coolant temperature at engine start is greater than or equal to 40

degrees Fahrenheit but less than or equal to 95 degrees Fahrenheit)

and TCO_ST <= (TAM_ST + C_TAM_ST_HYS_EVAP_RBM)
(less than or equal to 12 degrees Fahrenheit higher than ambient temperature at engine start

then LV_STATE_RBM_EVAP_BIT_2 = 1

endif

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TAM_MIN_EVAP_RBM	-	0...FEH	-48...142.5	0.75	°C
Minimum ambient temperature threshold for EVAP_RBM					
C_TAM_MAX_EVAP_RBM	-	0...FEH	-48...142.5	0.75	°C
Maximum ambient temperature threshold for EVAP_RBM					
C_T_AST_EVAP_RBM	-	0...FFFFH	0...65535	1	s
Minimum comulative time since engine start for incrementation of the evaporative system monitor denominator					
C_TCO_ST_MIN_EVAP_RBM	-	0...FEH	-48...142.5	0.75	°C
Minimum engine start temperature threshold for EVAP_RBM					
C_TCO_ST_MAX_EVAP_RBM	-	0...FEH	-48...142.5	0.75	°C
Maximum engine start temperature threshold for EVAP_RBM					
C_TAM_ST_HYS_EVAP_RBM	-	0...FEH	0...190. 5	0.75	°C
Ambient temperature hysteresis for EVAP_RBM					

B.12.4.2 RBM interface for the detection of a 'noisy' DTP sensor signal (ERR_DTP_NOISE)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_DTP_NOISE	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor 'noisy' DTP sensor signal for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_DTP_NOISE	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of noisy DTP sensor failure, for application purposes					

Input data:

STATE_DIAGCP	CTR_COMP_RBM_DTP_NOISE	CTR_CDN_RBM_DTP_NOISE	
LV_INH_DIAGCP	LV_END_DIAG_DTP_NOISE		

FUNCTION DESCRIPTION:

General information:

With this module the interface between the CPS Mechanical failure and the Rate-Based Monitoring statistics is defined with STATE_RBM_DTP_NOISE data.

Within STATE_RBM_DTP_NOISE three different information are defined:

Information about bit 0 of STATE_RBM_DTP_NOISE:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_DTP_NOISE:

Monitor disabled because of system malfunction

Information about bit 2 of STATE_RBM_DTP_NOISE:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation: *at ECU reset and LV_DC 0 → 1 transition :*
STATE_RBM_DTP_NOISE = 0
on failure memory reset :
bit 1 of STATE_RBM_DTP_NOISE = 0

Recurrence: 1 sec.

Activation: LV_DC = 1

De-activation: LV_DC = 0

Formula section:

```
If      bit 0 of STATE_RBM_DTP_NOISE = 0    (Numerator increment condition)
then    if      {       STATE_DIAGCP >= 2h     (PRS_DYN (1h)check finished)
                and     STATE_DIAGCP <> Bh        (No wait period)        }
                or      LV_END_DIAG_DTP_NOISE = 1
        then    bit 0 of STATE_RBM_DTP_NOISE = 1
        endif
endif

If      bit 1 of STATE_RBM_DTP_NOISE = 0
then    if      LV_INH_DIAGCP = 1  (Diagnosis inhibited)
        then    bit 1 of STATE_RBM_DTP_NOISE = 1
        endif
endif
```

- No special condition defined in CARB Mail-out for DTP_NOISE diagnosis

Bit 2 of STATE_RBM_DTP_NOISE = 1 (always)

Calculation of actual RBM rate (for application purposes)

```
If      CTR_CDN_RBM_DTP_NOISE_DIAG = 0
then    RATE_RBM_DTP_NOISE = FFFFH
else    RATE_RBM_DTP_NOISE
                  = ( CTR_COMP_RBM_DTP_NOISE /
                      CTR_CDN_RBM_DTP_NOISE )
endif
```

B.12.4.3 RBM interface for CPS failure (ERR_MEC_CPS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_MEC_CPS	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor CPS Mechanical failure for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_MEC_CPS	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of CPS Mechanical failure, for application purposes					

Input data:

STATE_DIAGCP	CTR_COMP_RBM_MEC_CPS	CTR_CDN_RBM_MEC_CPS	
LV_INH_DIAGCP	LV_END_DIAG_MEC_CPS		

FUNCTION DESCRIPTION:

General information:

With this module the interface between the CPS Mechanical failure and the Rate-Based Monitoring statistics is defined with STATE_RBM_MEC_CPS data.

Within STATE_RBM_MEC_CPS three different information are defined:

Information about bit 0 of STATE_RBM_MEC_CPS:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_MEC_CPS:

Monitor disabled because of system malfunction

Information about bit 2 of STATE_RBM_MEC_CPS:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation: *at ECU reset and LV_DC 0 → 1 transition :*

STATE_RBM_MEC_CPS = 0

on failure memory reset :

bit 1 of STATE_RBM_MEC_CPS = 0

Recurrence: 1 sec.

Activation: LV_DC = 1

De-activation: LV_DC = 0

Formula section:

```
If      bit 0 of STATE_RBM_MEC_CPS = 0    (Numerator increment condition)
then    if    {     STATE_DIAGCP >= 7h    (Vapor generation check (6h) finished)
              and   STATE_DIAGCP <> Bh      (No wait period)       }
              or    LV_END_DIAG_MEC_CPS = 1
        then  bit 0 of STATE_RBM_MEC_CPS = 1
        endif
endif

If      bit 1 of STATE_RBM_MEC_CPS = 0
then    if    LV_INH_DIAGCP = 1  (Diagnosis inhibited)
        then  bit 1 of STATE_RBM_MEC_CPS = 1
        endif
endif
```

- No special condition defined in CARB Mail-out for MEC_CPS diagnosis

Bit 2 of STATE_RBM_MEC_CPS = 1 (always)

Calculation of actual RBM rate (for application purposes)

```
If      CTR_CDN_RBM_MEC_CPS_DIAG = 0
then    RATE_RBM_MEC_CPS = FFFFH
else    RATE_RBM_MEC_CPS
                 = ( CTR_COMP_RBM_MEC_CPS / CTR_CDN_RBM_MEC_CPS )
endif
```

B.12.4.4 RBM interface for “Small Leak” (0,5 and 1mm Leak)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_VAP_LEAK_1	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor EVAP Small Leakage for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_VAP_LEAK_1	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of EVAP Small Leakage, for application purposes					

Input data:

	CTR_COMP_RBM_VAP_LEAK_1	CTR_CDN_RBM_VAP_LEAK_1	
LV_STATE_RBM_EVAP_BIT_2	LV_INH_DIAGCP	SUM_DIAG_VAP_LEAK_1	

FUNCTION DESCRIPTION:

General information:

With this module the interface between the EVAP Small Leakage failure and the Rate-Based Monitoring statistics is defined with STATE_RBM_VAP_LEAK_1 data.

Within STATE_RBM_VAP_LEAK_1 three different information are defined:

Information about bit 0 of STATE_RBM_VAP_LEAK_1:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_VAP_LEAK_1:

Monitor disabled because of system malfunction

Information about bit 2 of STATE_RBM_VAP_LEAK_1:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation: *at ECU reset and LV_DC 0 → 1 transition :*

STATE_RBM_VAP_LEAK_1 = 0

on failure memory reset :

bit 1 of STATE_RBM_VAP_LEAK_1 = 0

Recurrence: 1 sec.

Activation: LV_DC = 1

De-activation: LV_DC = 0

Formula section:

If bit 0 of STATE_RBM_VAP_LEAK_1 = 0 (Numerator increment condition)
then **if** SUM_DIAG_VAP_LEAK_1 = 1 (1st diagnosis at multiple check)
then bit 0 of STATE_RBM_VAP_LEAK_1 = 1
endif
endif

If bit 1 of STATE_RBM_VAP_LEAK_1 = 0
then **if** LV_INH_DIAGCP = 1 (Diagnosis inhibited)
then bit 1 of STATE_RBM_VAP_LEAK_1 = 1
endif
endif

- As special conditions for EVAP monitoring are defined in CARB Mail-out, common flag LV_STATE_RBM_EVAP_BIT_2 is used to set Bit 2 :

Bit 2 of STATE_RBM_VAP_LEAK_1 = LV_STATE_RBM_EVAP_BIT_2

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_VAP_LEAK_1_DIAG = 0
then RATE_RBM_VAP_LEAK_1 = FFFFH
else RATE_RBM_VAP_LEAK_1
= (CTR_COMP_RBM_VAP_LEAK_1 / CTR_CDN_RBM_VAP_LEAK_1)
endif

B.12.4.5 RBM interface for “Large Leak” (Large Leak or Tank Cap missing out of EVAP monitoring)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_FUC_MISS_1	O/V	0 ... 7H	0 ... 7H	1	-
Interface of monitor EVAP Large Leakage for the rate based monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_FUC_MISS_1	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM rate of EVAP Large Leakage, for application purposes					

Input data:

LV_END_DIAG_FUC_MISS_1	CTR_COMP_RBM_FUC_MISS_1	CTR_CDN_RBM_FUC_MISS_1	
LV_STATE_RBM_EVAP_BIT_2			

FUNCTION DESCRIPTION:

General information:

With this module the interface between the Tank Cap missing failure and the Rate-Based Monitoring statistics is defined with STATE_RBM_FUC_MISS_1 data.

Within STATE_RBM_FUC_MISS_1 three different information are defined:

Information about bit 0 of STATE_RBM_FUC_MISS_1:

Conditions for monitoring are met long enough to detect malfunction (no intrusive operation, no short trip)

Information about bit 1 of STATE_RBM_FUC_MISS_1:

Monitor disabled because of system malfunction

Information about bit 2 of STATE_RBM_FUC_MISS_1:

Monitor individual RBM conditions encountered within this DC

Application conditions:

Initialisation: *at ECU reset and LV_DC 0 → 1 transition :*
STATE_RBM_FUC_MISS_1 = 0
on failure memory reset :
bit 1 of STATE_RBM_FUC_MISS_1 = 0

Recurrence: 1 sec.

Activation: LV_DC = 1

De-activation: LV_DC = 0

Formula section:

```
If      bit 0 of STATE_RBM_FUC_MISS_1 = 0    (Numerator increment condition)
then    if      LV_END_DIAG_FUC_MISS_1 = 1
        then    bit 0 of STATE_RBM_FUC_MISS_1 = 1
        endif
endif

If      bit 1 of STATE_RBM_FUC_MISS_1 = 0
then    if      LV_INH_DIAGCP = 1  (Diagnosis inhibited)
        then    bit 1 of STATE_RBM_FUC_MISS_1 = 1
        endif
endif
```

- As no special conditions for Tank Cap missing are defined in CARB Mail-out :
Bit 2 of STATE_RBM_FUC_MISS_1 = 1 Always

Calculation of actual RBM rate (for application purposes)

```
If      CTR_CDN_RBM_FUC_MISS_1_DIAG = 0
then    RATE_RBM_FUC_MISS_1 = FFFFH
else    RATE_RBM_FUC_MISS_1
                  = ( CTR_COMP_RBM_FUC_MISS_1 /
                    CTR_CDN_RBM_FUC_MISS_1 )
endif
```

B.12.4.6 Counters to determine why EVAP monitoring not running in case of Idle phase longer than 30 sec.

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_EVAP_T_IS	V/S	0...FFH	0...255	1	-
Total number of 30 sec. Idle phases with vehicle stopped to perform EVAP monitoring					
CTR_EVAP_T_DLY_3	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP monitoring disabled because system still in “Wait” period					
CTR_EVAP_DTP_DIAG	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. still running (DTP_DIAG) after 30 sec.					
CTR_EVAP_DIAGCP_START	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. disabled due to system not ready					
CTR_EVAP_INTR_DIAGCP	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is started but aborted before it is finished					
CTR_EVAP_TCO	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is not started because TCO is too low					
CTR_EVAP_T_AST	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is not started because time after Start is too low					
CTR_EVAP_T_PL_DIAGCP	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is not started because T_PL_DIAGCP > 0					
CTR_EVAP_INTR_1	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to too high vapor generation					
CTR_EVAP_INTR_2	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to too big Lambda cotnroller deviation					
CTR_EVAP_INTR_4	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to too high MAF fluctuation					
CTR_EVAP_INTR_5	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to too high DTP slope variation					
CTR_EVAP_INTR_12	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to fuel level out of range					
CTR_EVAP_INTR_10	V/S	0...FFH	0...255	1	-
Number of Idle phases where EVAP mon. is aborted due to other condition					
LV_T_DLY_3	V	0...1H	0...1	1	-
Boolean flag to indicate if STATE_DIAGCP = “T_DLY_3” at the beginning of an Idle phase with VS = 0					
LV_INC_CTR_EVAP	V	0...1H	0...1	1	-
Boolean flag to indicate that CTR_EVAP_xx counters can be incremented					

Input data:

T_IS	STATE_DIAGCP	C_TAM_MIN_EVAP_RBM	TAM
C_T_IS_RBM	T_AST	C_TAM_MAX_EVAP_RBM	VS
T_PL_DIAGCP	ID_T_AST_MIN_DIAGCP	C_TCO_MIN_DIAGCP	TCO
STATE_INTR_DIAGCP	ID_T_AST_MIN_DIAGCP_2	TCO_ST	

FUNCTION DESCRIPTION:

General information:

Target of this module is to monitor inside ECU why is EVAP monitoring not peformed if there is an Idle phase with vehicle stopped longer than 30 sec. To improve as much as possible in-use performance ratio, reason why EVAP monitoring is not performed is determined via some counters (14) :

- Global counter CTR_EVAP_T_IS is counting number of Idle phases with vehicle stopped longer than 30 sec.

- Main counters (CTR_EVAP_T_DLY_3, CTR_EVAP_DTP_DIAG, CTR_EVAP_DIAGCP_START, CTR_EVAP_INTR_DIAGCP) are checking if EVAP monitoring is :
    - Not started because system in "Wait" state (CTR_EVAP_T_DLY_3)
    - Not started because system is not ready (CTR_EVAP_DIAGCP_START)
    - Started but not finished (CTR_EVAP_DTP_DIAG -> Diagnosis still on-going after 30 sec., CTR_EVAP_INTR_DIAGCP -> Diagnosis aborted)

- Sub-counters give more informations why was EVAP monitoring not started or aborted (TCO too low, Time after Start too low, Purge time in Part Load too low, Vapor generation too high, Lambda deviation too high, MAF variation too high, DTP slope variation during evacuation too high, Fuel level out of range....)

Application conditions:

Initialisation: If ECU is brand new or LC_CTR_EVAP_RST = 1, then all CTR_EVAP_xxx are initialised to 0

Recurrence: 500 msec.

Activation: LV_DC = 1

De-activation: LV_DC = 0

Formula section:

- Definition of LV_T_DLY_3 (*check if STATE_DIAGCP = "T_DLY_3" at the beginning of Idle phase with VS = 0*) :

If $TI\_IS_{(n)} < T\_IS_{(n-1)}$ (*TI_IS reseted -> New Idle phase with VS = 0*)
Then
  If STATE_DIAGCP = "T_DLY_3"
  Then LV_T_DLY_3 = 1
  Else LV_T_DLY_3 = 0
  Endif
Endif

- Check there is a new Idle phase with vehicle stopped longer than 30 sec. -> CTR_EVAP_xx to be incremented (LV_INC_CTR_EVAP) :

If STATE_DIAGCP <> "PAS" (*EVAP monitoring not yet performed*)
  and T_IS > C_T_IS_RBM (*Idle phase with vehicle stopped of more than 30sec.)*
  and C_TAM_MIN_EVAP_RBM < TAM < C_TAM_MAX_EVAP_RBM
  (*Ambient temperature limits to set Bit 2 of STATE_RBM for EVAP monitoring*)
Then
  LV_INC_CTR_EVAP = 1
Else
  LV_INC_CTR_EVAP = 0

- Check why EVAP monitoring was not completed in case of long Idle phase with vehicle stopped:

If (1) Transition LV_INC_CTR_EVAP = 0 -> 1
Then (1) CTR_EVAP_T_IS = CTR_EVAP_T_IS + 1
(*Total number of Idle phases where possible to perform EVAP monitoring)*
If (2) LV_T_DLY_3 = 1 (*STATE_DIAGCP = "T_DLY_3 since begin of Idle*)
Then (2) CTR_EVAP_T_DLY_3 = CTR_EVAP_T_DLY_3 + 1
Else (2)
If (3) STATE_DIAGCP = "DLY_DIAG" **or** "DTP_DIAG" (*diagnosis on-going at take-off*)
Then (3) CTR_EVAP_DTP_DIAG = CTR_EVAP_DTP_DIAG + 1
Else (3)
If (4) STATE_DIAGCP = "START" (*diagnosis not ready to Start*)
Then (4) CTR_EVAP_DIAGCP_START = CTR_EVAP_DIAGCP_START+1
(*Total number of EVAP monitoring disabled due to system not ready*)
If TCO < C_TCO_MIN_DIAGCP
Then CTR_EVAP_TCO = CTR_EVAP_TCO + 1
If T_AST < max (ID_T_AST_MIN_DIAGCP, ID_T_AST_MIN_DIAGCP_2)
Then CTR_EVAP_T_AST = CTR_EVAP_T_AST + 1
If T_PL_DIAGCP > 0
Then CTR_EVAP_T_PL_DIAGCP=CTR_EVAP_T_PL_DIAGCP+1
Else (4)
If (5) STATE_DIAGCP = "T_DLY_3" **or** "ERR_CLOSE_CPS"
(*Diagnosis was started but aborted before finished*)
Then (5) CTR_EVAP_INTR_DIAGCP = CTR_EVAP_INTR_DIAGCP+1
If Bit 1 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_1 = CTR_EVAP_INTR_1 + 1
(*vapor generation too high*)
If Bit 2 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_2 = CTR_EVAP_INTR_2 + 1
(*Lambda deviation too high during evacuation*)
If Bit 4 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_4 = CTR_EVAP_INTR_4 + 1
(*MAF variation too high during evacuation*)
If Bit 5 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_5 = CTR_EVAP_INTR_5 + 1
(*DTP slope variation during evacuation too high*)
If Bit 12 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_12 = CTR_EVAP_INTR_12 + 1
(*Fuel level out of range*)
If Bit 10 of STATE_INTR_DIAGCP = 1
Then CTR_EVAP_INTR_10 = CTR_EVAP_INTR_10 + 1
(*Interruption due to other condition*)
Endif (5)
Endif (4)
Endif (3)
Endif (2)
Endif (1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_CTR_EVAP_RST	1	0...1H	0...1	1	-
Logical constant to request reset of CTR_EVAP_xx counters					

B.12.5 Variables for fleet monitoring

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
FAC_DIAM_DIAGCP_MAX_TOT_DC	V/O/S	0…FFH	0…4.127	16.184e-3	mm
Former / current driving cycle maximum reduced leakage diameter					

Input data:

FAC_DIAM_DIAGCP_MAX_TOT_DC	LV_END_DIAG_VAP_LEAK_1	LV_END_DIAG_VAP_LEAK_2	

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

FAC_DIAM_DIAGCP_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: 40 ms

Activation: LV_END_DIAG_VAP_LEAK_1 = 0 → 1

or LV_END_DIAG_VAP_LEAK_2 = 0 → 1

Deactivation: -

Formula section:

If FAC_DIAM_DIAGCP > FAC_DIAM_DIAGCP_MAX_TOT_DC

Then FAC_DIAM_DIAGCP_MAX_TOT_DC = FAC_DIAM_DIAGCP

Endif

B.13 Fuel system diagnosis

B.13.1 General

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_FSD_i	V/O	0...1H	0...1	1	-
Diagnosis conditions - relevant to malfunction indication light					
LV_CDN_DIAG_FSD_FAC_H_i	V/O	0...1H	0...1	1	-
Diagnosis conditions in upper multiplicative adaptation learning area					
LV_CDN_DIAG_FSD_LAM_LIM_i	V/O	0...1H	0...1	1	-
Diagnosis conditions in the area where the lambda control output concerned - relevant to MIL					
ERR_SYM_FSD_i	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected failure of each symptom - relevant to malfunction indication light					
ERR_SYM_FSD_FAC_H_i	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected failure of each symptom in upper multiplicative adaptation learning area					
ERR_SYM_FSD_LAM_LIM_i	V/O	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Detected failure of each symptom in case of lambda control in dead stop - relevant to MIL					
LV_ERR_FSD_i	V/O	0...1H	0...1	1	-
Failure - relevant to malfunction indication light					
LV_ERR_FSD_FAC_H_i	V/O	0...1H	0...1	1	-
Failure in upper multiplicative adaptation learning area					
LV_ERR_FSD_LAM_LIM_i	V/O	0...1H	0...1	1	-
Failure - lambda control in dead stop					
LV_END_DIAG_FSD_i	V/O	0...1H	0...1	1	-
End of first diagnosis cycle - relevant to malfunction indication light					
LV_END_DIAG_FSD_FAC_H_i	V/O	0...1H	0...1	1	-
End of first diagnosis cycle in upper multiplicative adaptation learning area					
LV_END_DIAG_FSD_LAM_LIM_i	V/O	0...1H	0...1	1	-
End of first diagnosis cycle in the area where the lambda control output concerned					
LV_END_DIAG_WIN_FSD_i	V/O	0...1H	0...1	1	-
End of diagnosis cycle for similar conditions window in additive adaptation learning area					
LV_END_DIAG_WIN_FSD_LAM_LIM_i	V/O	0...1H	0...1	1	-
End of diagnosis cycle for similar conditions window in case of lambda control in dead stop					
LV_LAM_LIM_MFF_AD_i	V/O	0...1H	0...1	1	-
Request for forced lambda adaptation					
CTR_STOP_FSD	V/S	0...FFH	0...255	1	-
Counter value as a measure of the oil dilution under cold start conditions					
T_SUM_MAX_FSD_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_MFF_AD_ADD_LIM_MAX_i = 1 or LV_MFF_AD_FAC_L_LIM_MAX_i = 1					
T_SUM_MIN_FSD_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_MFF_AD_ADD_LIM_MIN_i = 1 or LV_MFF_AD_FAC_L_LIM_MIN_i = 1					
T_SUM_MAX_FSD_FAC_H_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_MFF_AD_FAC_H_LIM_MAX_i = 1					
T_SUM_MIN_FSD_FAC_H_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_MFF_AD_FAC_H_LIM_MIN_i = 1					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_SUM_MAX_FSD_LAM_LIM_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_LAM_LIM_MAX_i = 1					
T_SUM_MIN_FSD_LAM_LIM_i	V	0...FFFFH	0...1310.7	0.02	s
Total duration of LV_LAM_LIM_MIN_i = 1					
T_SUM_RST_FSD_i	V	0...FFFFH	0...1310.7	0.02	s
Timer to reset the counter of total duration in MIL - area					
T_SUM_RST_FSD_FAC_H_i	V	0...FFFFH	0...1310.7	0.02	s
Timer to reset the counter of total duration in FAC_H - area					
T_SUM_RST_FSD_LAM_LIM_i	V	0...FFFFH	0...1310.7	0.02	s
Timer to reset the counter of total duration for lambda control in dead stop					
T_SUM_END_DIAG_WIN_FSD_i	V	0...FFFFH	0...1310.7	0.02	s
Counter for the diagnosis window - relevant to MIL					
T_SUM_END_DIAG_WIN_FSD_FAC_H_i	V	0...FFFFH	0...1310.7	0.02	s
Counter for the diagnosis window - upper multiplicative adaptation learning area					
T_SUM_END_DIAG_WIN_FSD_LAM_i	V	0...FFFFH	0...1310.7	0.02	s
Counter for the diagnosis window - lambda control in dead stop					
T_LAM_LIM_MFF_AD_i	V	0...FFFFH	0...1310.7	0.02	s
Timer for forced lambda adaptation					
T_MFF_AD_ACT_FSD_i	V	0...FFFFH	0...1310.7	0.02	s
Timer indicating the forced lambda adaptation is running					
TOIL_MAX	V	0...C8H	-40...160	1	°C
Maximum Oil Temperature from the last engine running					

Input data:

LV_ST	LV_IGK	LV_LSCL_i	LV_MFF_AD_ADD_i
TIA	TOIL_MDL_FSD	TCO	TCO_ST
LV_INH_DIAG_FSD	LV_ES	MAF	AMP
LV_MFF_AD_FAC_L_i	LV_MFF_AD_FAC_H_i	LV_LAM_LIM_MIN_i	LV_LAM_LIM_MAX_i
LV_INH_DIAG_FSD_CP	N_32	TI_LAM_COR_i	LV_DC
LV_MFF_AD_FAC_L_LIM_MIN_i	LV_MFF_AD_FAC_L_LIM_MAX_i	LV_MFF_AD_FAC_H_LIM_MIN_i	
LV_MFF_AD_FAC_H_LIM_MAX_i	LV_MFF_AD_ADD_LIM_MIN_i	LV_MFF_AD_ADD_LIM_MAX_i	

General information:

The objective function of the fuel system diagnosis is to monitor the lambda control output and the lambda adaptation values in various areas. It should also cover the PCV monitoring where the lambda control output is considered in the idle range. Breaking the adaptation and lambda controller limits for a long time, which may have been caused by failures in the fuel or intake system will involve emission rise and therefore shall be diagnosed by fuel system diagnosis. Fuel system diagnosis is active as soon as the general activation conditions are fulfilled and the lambda controller is activated. That means the function monitors the lambda controller output independent on whether the lambda adaptation is activated. As soon as the lambda controller limits are exceeded (after an applicable time), the request for lambda adaptation will be set (LV_LAM_LIM_MFF_AD_i = 1). Figure 1 shows the monitoring area of fuel system diagnosis where the activation area of lambda adaptation is subset of lambda controller activation area. The diagnosis action is exemplary shown in Figure 2 for lambda controller output limited.

FSD errors are located in three failure memories with two and four symptoms each:

ERR_SYM_FSD_i:

NO_SYM:	no symptom
SYM_0:	maximum limit of additive adaptation value reached
SYM_1:	minimum limit of additive adaptation value reached
SYM_2:	maximum limit of multiplicative adaptation value (lower area) reached
SYM_3:	minimum limit of multiplicative adaptation value (lower area) reached

ERR_SYM_FSD_FAC_H_i:

NO_SYM:	no symptom
SYM_0:	maximum limit of multiplicative adaptation value (upper area) reached
SYM_1:	minimum limit of multiplicative adaptation value (upper area) reached

ERR_SYM_FSD_LAM_LIM_i:

NO_SYM:	no symptom
SYM_0:	lambda control in dead stop (upper limit)
SYM_1:	lambda control in dead stop (lower limit)

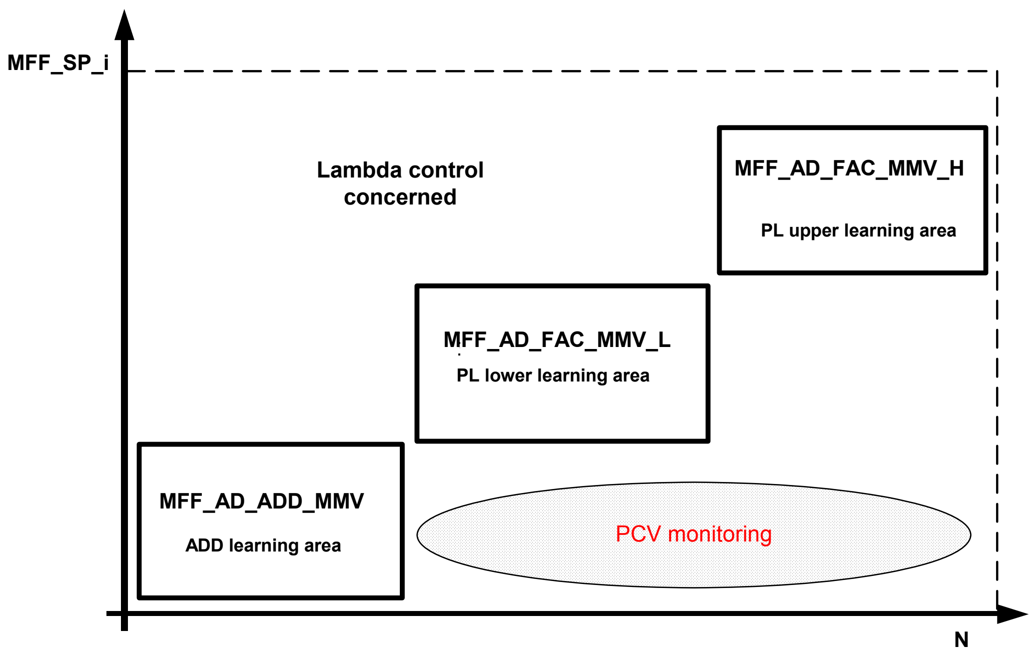
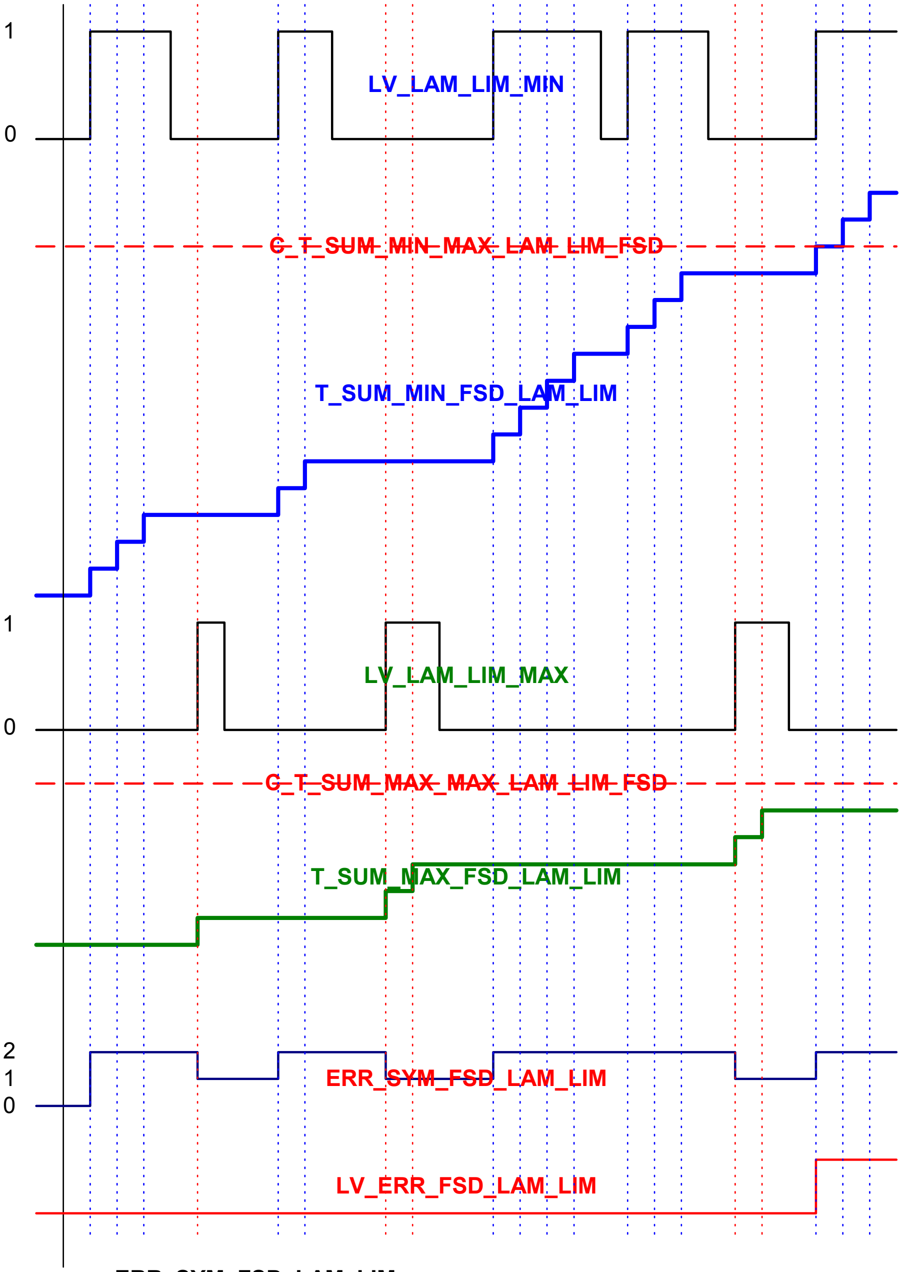


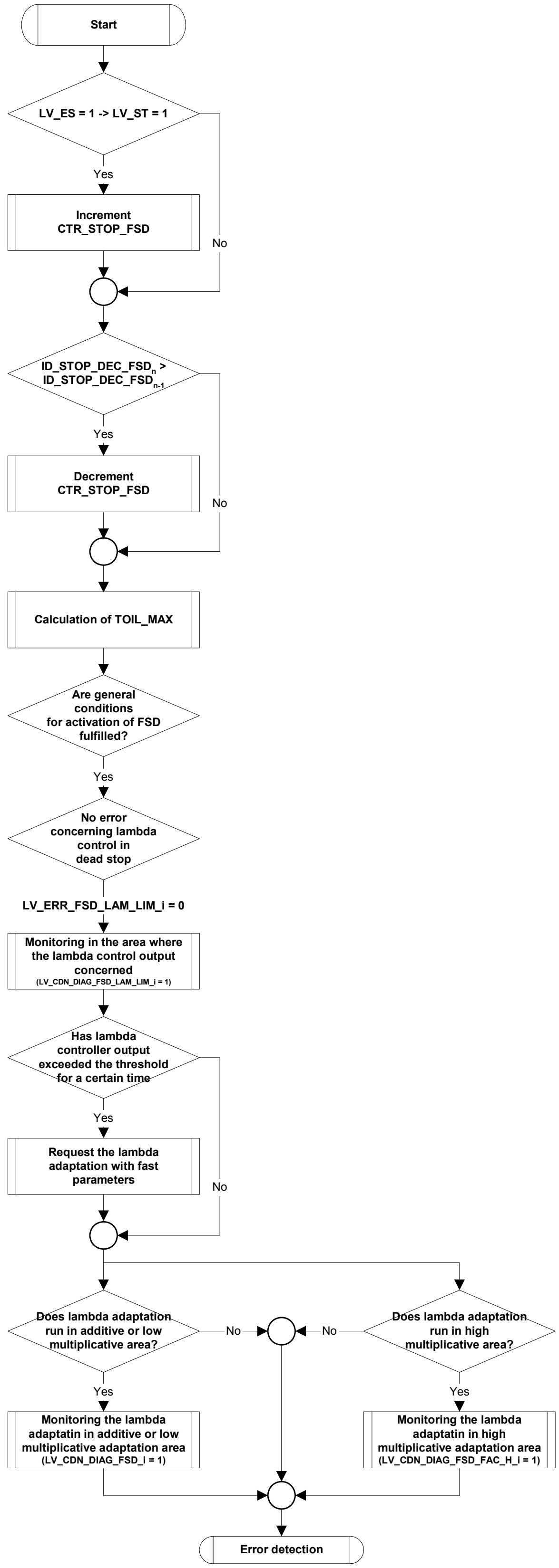
Figure 1: Monitoring area of fuel system diagnosis

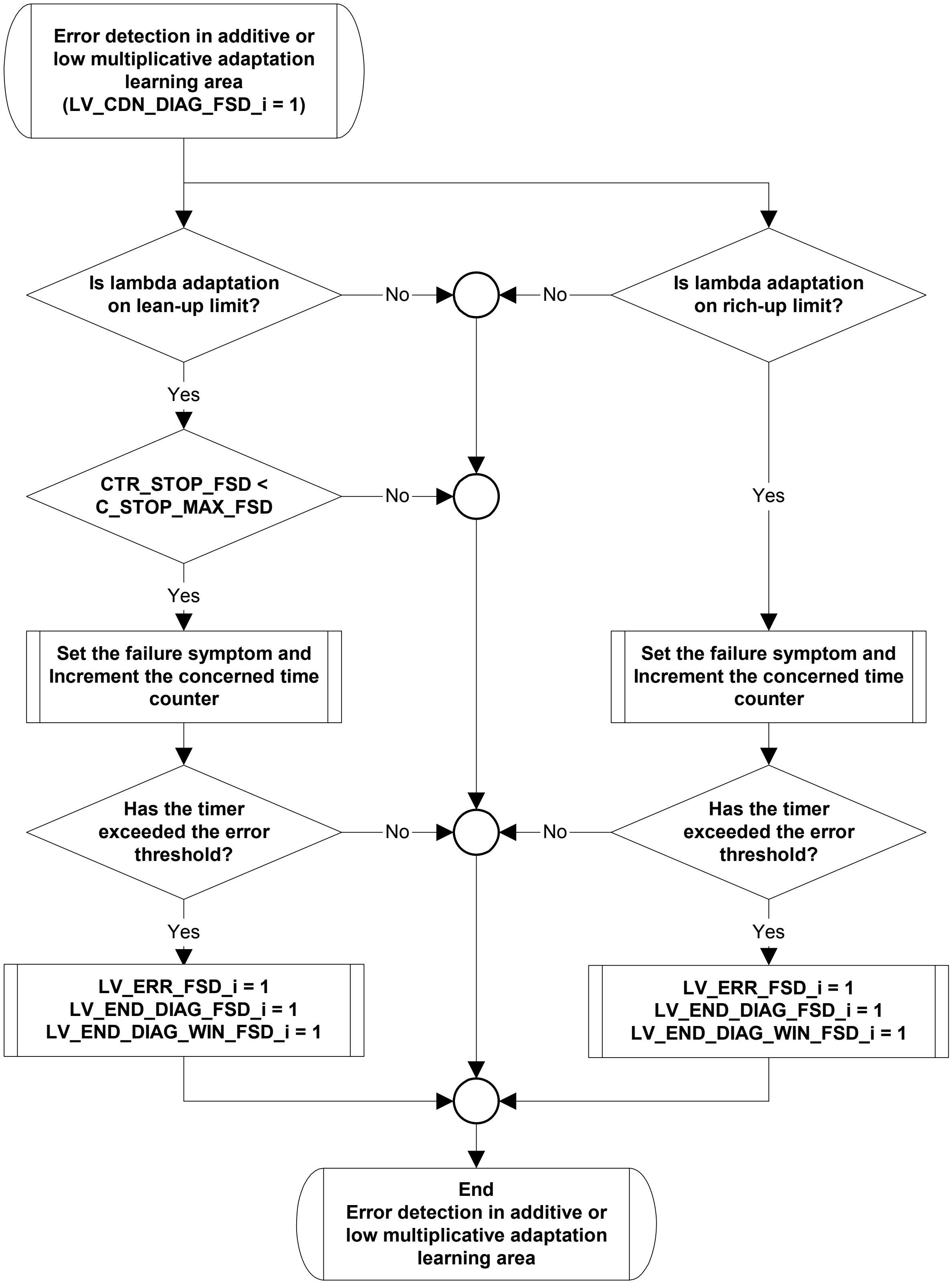


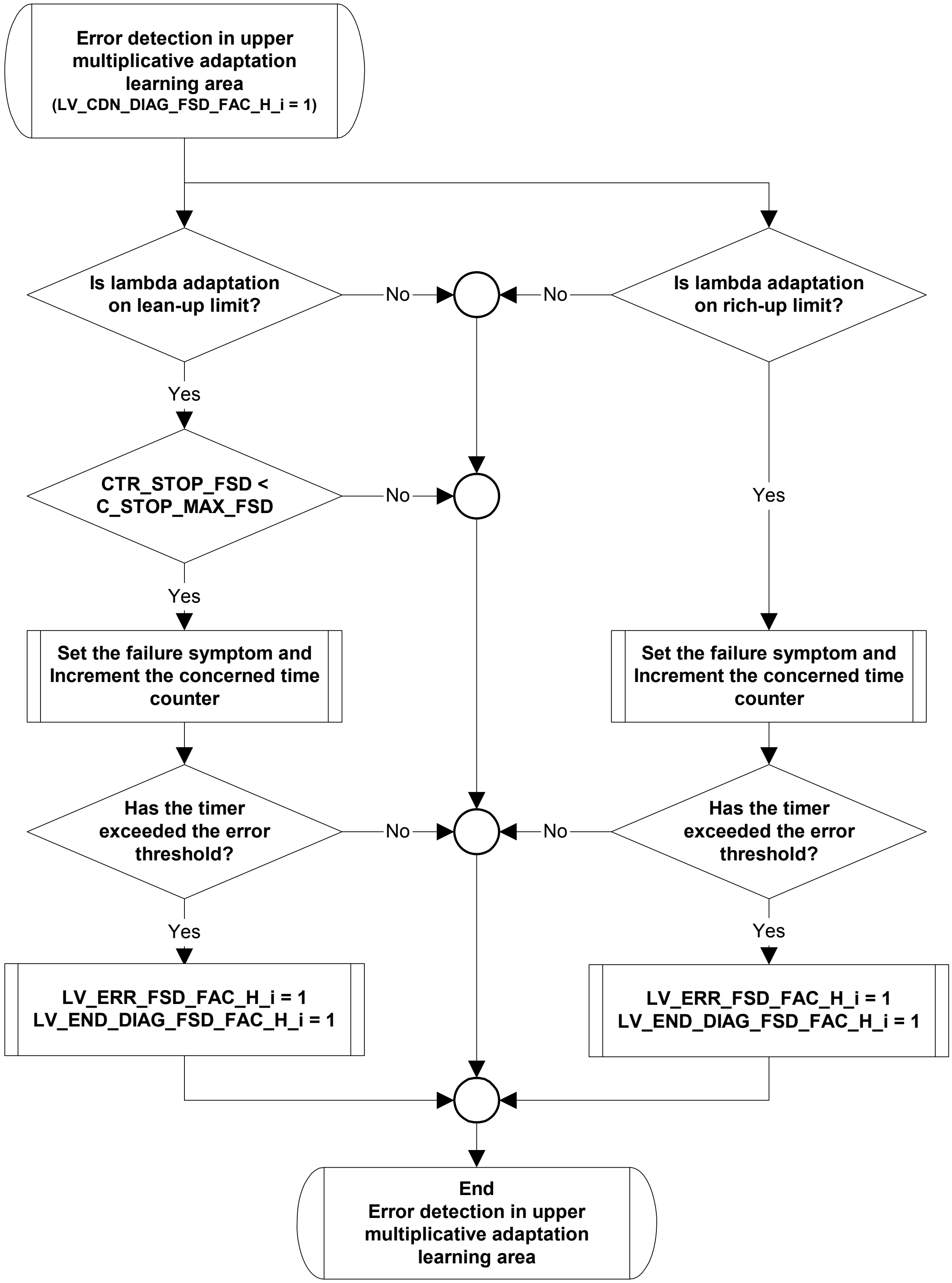
ERR_SYM_FSD_LAM_LIM=:
0: no symptom
1: lambda control in dead stop (upper limit)
2: lambda control in dead stop (lower limit)

Figure 2: Example of diagnosis action in case of lambda control in dead stop

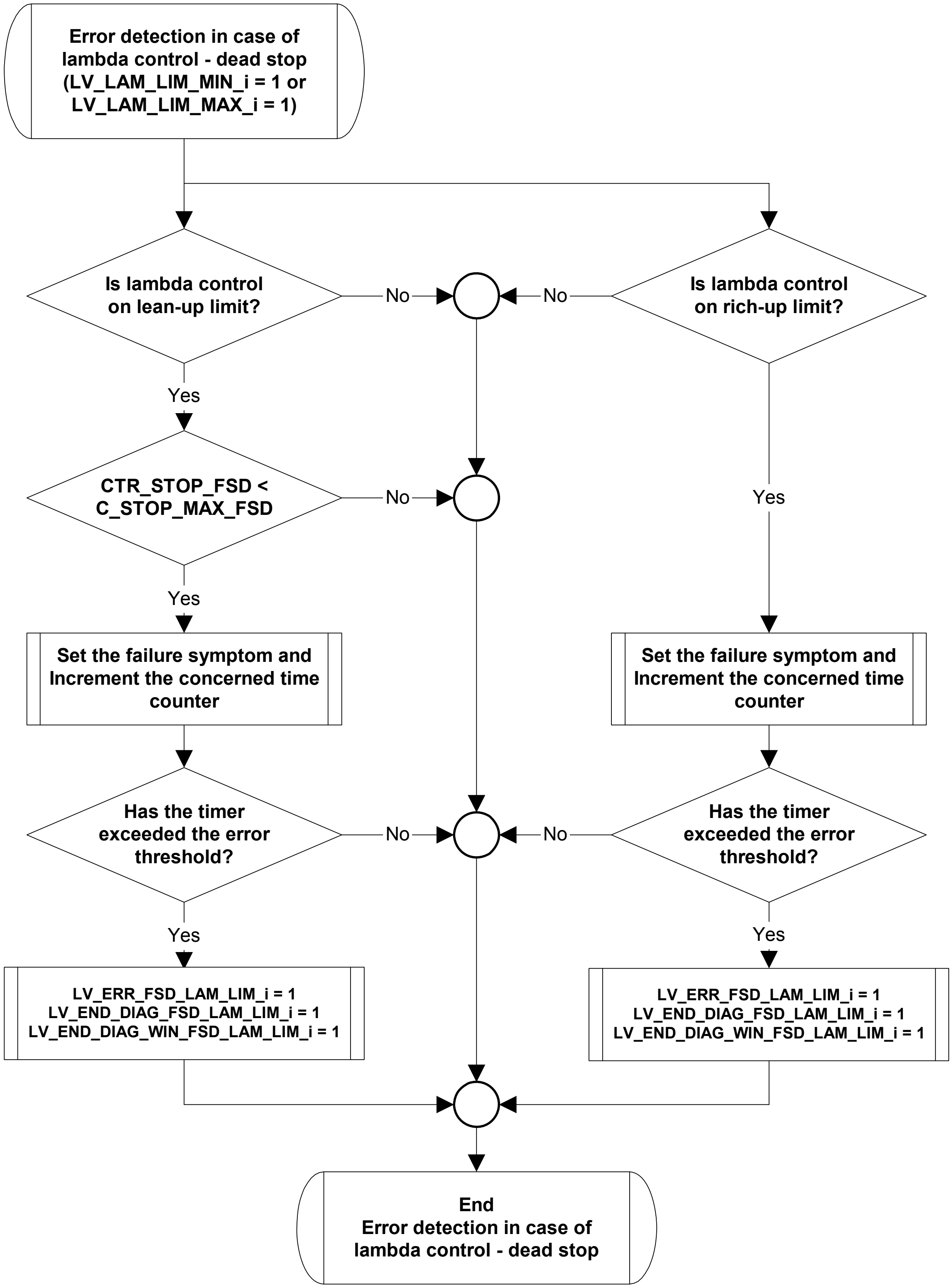
Signal Flow Diagram:




Error detection in additive or low multiplicative adaptation learning area (LV_CDN_DIAG_FSD_i = 1)
Is lambda adaptation on lean-up limit?
No
No
Is lambda adaptation on rich-up limit?
Yes
CTR_STOP_FSD < C_STOP_MAX_FSD
No
Yes
Yes
Set the failure symptom and Increment the concerned time counter
Set the failure symptom and Increment the concerned time counter
Has the timer exceeded the error threshold?
No
No
Has the timer exceeded the error threshold?
Yes
Yes
LV_ERR_FSD_i = 1
LV_END_DIAG_FSD_i = 1
LV_END_DIAG_WIN_FSD_i = 1
LV_ERR_FSD_i = 1
LV_END_DIAG_FSD_i = 1
LV_END_DIAG_WIN_FSD_i = 1
End
Error detection in additive or low multiplicative adaptation learning area



Error detection in upper multiplicative adaptation learning area (LV_CDN_DIAG_FSD_FAC_H_i = 1)
Is lambda adaptation on lean-up limit?
Is lambda adaptation on rich-up limit?
No
No
Yes
CTR_STOP_FSD < C_STOP_MAX_FSD
No
Yes
Yes
Set the failure symptom and Increment the concerned time counter
Set the failure symptom and Increment the concerned time counter
Has the timer exceeded the error threshold?
No
No
Has the timer exceeded the error threshold?
Yes
Yes
LV_ERR_FSD_FAC_H_i = 1
LV_END_DIAG_FSD_FAC_H_i = 1
LV_ERR_FSD_FAC_H_i = 1
LV_END_DIAG_FSD_FAC_H_i = 1
End
Error detection in upper multiplicative adaptation learning area


Error detection in case of lambda control - dead stop (LV_LAM_LIM_MIN_i = 1 or LV_LAM_LIM_MAX_i = 1)
Is lambda control on lean-up limit?
No
No
Is lambda control on rich-up limit?
Yes
CTR_STOP_FSD < C_STOP_MAX_FSD
No
Yes
Yes
Set the failure symptom and Increment the concerned time counter
Set the failure symptom and Increment the concerned time counter
Has the timer exceeded the error threshold?
No
No
Has the timer exceeded the error threshold?
Yes
Yes
LV_ERR_FSD_LAM_LIM_i = 1
LV_END_DIAG_FSD_LAM_LIM_i = 1
LV_END_DIAG_WIN_FSD_LAM_LIM_i = 1
LV_ERR_FSD_LAM_LIM_i = 1
LV_END_DIAG_FSD_LAM_LIM_i = 1
LV_END_DIAG_WIN_FSD_LAM_LIM_i = 1
End
Error detection in case of lambda control - dead stop

Application conditions:

Initialisation:

At reset, LV_IGK = 0 → 1, LV_ES → LV_ST and clear error memory:

T_SUM_RST_FSD_i	=	0
T_SUM_RST_FSD_FAC_H_i	=	0
T_SUM_RST_FSD_LAM_LIM_i	=	0
T_SUM_END_DIAG_WIN_FSD_i	=	0
T_SUM_END_DIAG_WIN_FSD_FAC_H_i	=	0
T_SUM_END_DIAG_WIN_FSD_LAM_i	=	0
T_SUM_MIN_FSD_i	=	0
T_SUM_MIN_FSD_LAM_LIM_i	=	0
T_SUM_MIN_FSD_FAC_H_i	=	0
T_SUM_MAX_FSD_i	=	0
T_SUM_MAX_FSD_LAM_LIM_i	=	0
T_SUM_MAX_FSD_FAC_H_i	=	0
T_LAM_LIM_MFF_AD_i	=	0
T_MFF_AD_ACT_FSD_i	=	0
LV_ERR_FSD_i	=	0
LV_ERR_FSD_FAC_H_i	=	0
LV_ERR_FSD_LAM_LIM_i	=	0
LV_END_DIAG_WIN_FSD_i	=	0
LV_END_DIAG_WIN_FSD_LAM_LIM_i	=	0
LV_LAM_LIM_MFF_AD_i	=	0
LV_CDN_DIAG_FSD_i	=	0
LV_CDN_DIAG_FSD_FAC_H_i	=	0
LV_CDN_DIAG_FSD_LAM_LIM_i	=	0
ERR_SYM_FSD_i	=	0
ERR_SYM_FSD_FAC_H_i	=	0
ERR_SYM_FSD_LAM_LIM_i	=	0

CTR_STOP_FSD is restored out of the non-volatile memory.

At reset, LV_DC = 0 → 1 and clear error memory:

LV_END_DIAG_FSD_i	=	0
LV_END_DIAG_FSD_FAC_H_i	=	0
LV_END_DIAG_FSD_LAM_LIM_i	=	0

Only at reset:

TOIL_MAX = TOIL_MDL_FSD **and** calculate $ID\_STOP\_DEC\_FSD_{n-1}$ based on TOIL_MAX

B.13.2 Fuel system diagnosis inhibition due to fuel dilution in engine oil

For high rate of evaporated fuel (diluted in the engine oil) during the rich warm-up phase under cold start conditions, the lambda controller may reach the rich side limitation C_LAM_MIN. In this case, in order to prevent an error display, the fuel system diagnosis shall be deactivated.
According to TCO_ST the corresponding counter CTR_STOP_FSD is incremented once at every transition from engine stop to engine start. According to the maximum oil temperature during the last engine run the counter is decremented at every engine stop. The maximum oil temperature during last engine run is always calculated out of TOIL_MDL_FSD.

Incremental feed:

if LV_ES = 1 → LV_ST = 1 (*engine operating state: "engine stopped → start"*)

then

$CTR\_STOP\_FSD_n = CTR\_STOP\_FSD_{n-1} + ID\_STOP\_INC\_FSD\_\_TCO\_ST$

endif

Decrementing:

The decremental action of CTR_STOP_FSD shall be updated every 1 second on following condition:

if $ID\_STOP\_DEC\_FSD_n > ID\_STOP\_DEC\_FSD_{n-1}$

then

$CTR\_STOP\_FSD_n = CTR\_STOP\_FSD_{n-1} - (ID\_STOP\_DEC\_FSD_n - ID\_STOP\_DEC\_FSD_{n-1})$

endif

Recurrence: 20 ms

Activation:

- all engine operating states

Deactivation:

-

Formula section:

B.13.3 Oil temperature calculation (TOIL_MAX)

TOIL_MAX **=** max (TOIL_MDL_FSD) during the last engine running (LV_ES = 0)

B.13.4 Detection of diagnosis area and request for forced lambda adaptation (LV_LAM_LIM_MFF_AD_i = 1)

if LV_LSCL_i = 1 *(lambda controller shall be active)* **and**

LV_INH_DIAG_FSD = 0 *(diagnosis is not inhibited)* **and**

LV_INH_DIAG_FSD_CP = 0

(no inhibition due to high charcoal canister saturation degree) **and**

N_32 **>** C_N_MIN_FSD **and**

MAF **>** C_MAF_MIN_FSD **and**

TCO **>** C_TCO_MIN_FSD *(no diagnosis in case of warm-up)* **and**

AMP **>** C_AMP_MIN_FSD *(no diagnosis in case of high altitude)* **and**

TIA **>** C_TIA_MIN_FSD *(no diagnosis in case of low intake air temperature)*

then

if (LV_ERR_FSD_LAM_LIM_i = 0)

then

LV_CDN_DIAG_FSD_LAM_LIM_i = 1

else

LV_CDN_DIAG_FSD_LAM_LIM_i = 0

LV_END_DIAG_WIN_FSD_LAM_LIM_i = 0

endif

if (TI_LAM_COR_i > C_FAC_LAM_OUT_MAX_FSD **or**
TI_LAM_COR_i < C_FAC_LAM_OUT_MIN_FSD)

then

if T_LAM_LIM_MFF_AD_i ≥ C_T_LAM_LIM_MFF_AD

then

LV_LAM_LIM_MFF_AD_i = 1

else

increment (T_LAM_LIM_MFF_AD_i)

LV_LAM_LIM_MFF_AD_i = 0

endif

else

if T_MFF_AD_ACT_FSD_i ≥ C_T_MFF_AD_ACT_MIN_FSD

then

LV_LAM_LIM_MFF_AD_i = 0

T_LAM_LIM_MFF_AD_i = 0

T_MFF_AD_ACT_FSD_i = 0

endif

endif

```
if      LV_LAM_LIM_MFF_AD_i = 1
then
        if      (LV_MFF_AD_ADD_i = 1 or LV_MFF_AD_FAC_L_i = 1 or
                                                LV_MFF_AD_FAC_H_i = 1)
        then
                increment (T_MFF_AD_ACT_FSD_i)
        endif
endif

if      (LV_MFF_AD_ADD_i = 1 or LV_MFF_AD_FAC_L_i = 1)
then
        LV_CDN_DIAG_FSD_i = 1
        LV_CDN_DIAG_FSD_FAC_H_i = 0

elseif  LV_MFF_AD_FAC_H_i = 1
then
        LV_CDN_DIAG_FSD_FAC_H_i = 1
        LV_CDN_DIAG_FSD_i = 0
        LV_END_DIAG_WIN_FSD_i = 0

else
        LV_CDN_DIAG_FSD_FAC_H_i = 0
        LV_CDN_DIAG_FSD_i = 0
        LV_END_DIAG_WIN_FSD_i = 0
endif

if      LV_ERR_FSD_i = 1
then
        LV_CDN_DIAG_FSD_i = 0
        LV_END_DIAG_WIN_FSD_i = 0
endif

if      LV_ERR_FSD_FAC_H_i = 1
then
        LV_CDN_DIAG_FSD_FAC_H_i = 0
endif
```

```
else
    LV_CDN_DIAG_FSD_FAC_H_i = 0
    LV_CDN_DIAG_FSD_i = 0
    LV_CDN_DIAG_FSD_LAM_LIM_i = 0
    LV_END_DIAG_WIN_FSD_i = 0
    LV_END_DIAG_WIN_FSD_LAM_LIM_i = 0
endif
```

B.13.5 Error detection monitoring

B.13.5.1 Error detection in additive adaptation learning area

```
if  (LV_CDN_DIAG_FSD_i = 1 and LV_MFF_AD_ADD_LIM_MIN_i = 1 and
                                                    LV_MFF_AD_ADD_i = 1)
then
    if      CTR_STOP_FSD < C_STOP_MAX_FSD
    then
            increment (T_SUM_MIN_FSD_i)
            ERR_SYM_FSD_i = 2
            if      T_SUM_MIN_FSD_i > C_T_SUM_MIN_MAX_FSD
            then
                    LV_ERR_FSD_i = 1
                    LV_END_DIAG_FSD_i = 1
                    LV_END_DIAG_WIN_FSD_i = 1
            endif
    endif

elseif  (LV_CDN_DIAG_FSD_i = 1 and LV_MFF_AD_ADD_LIM_MAX_i = 1 and
                                                    LV_MFF_AD_ADD_i = 1)
then
    increment (T_SUM_MAX_FSD_i)
    ERR_SYM_FSD_i = 1
    if      T_SUM_MAX_FSD_i > C_T_SUM_MAX_MAX_FSD
    then
            LV_ERR_FSD_i = 1
            LV_END_DIAG_FSD_i = 1
            LV_END_DIAG_WIN_FSD_i = 1
    endif

endif
```

B.13.5.2 Error detection in lower multiplicative adaptation learning area

```
if    (LV_CDN_DIAG_FSD_i = 1 and LV_MFF_AD_FAC_L_LIM_MIN_i = 1 and
                                                    LV_MFF_AD_FAC_L_i = 1)
then
      if      CTR_STOP_FSD < C_STOP_MAX_FSD
      then
              increment (T_SUM_MIN_FSD_i)
              ERR_SYM_FSD_i = 8
              if      T_SUM_MIN_FSD_i > C_T_SUM_MIN_MAX_FSD
              then
                      LV_ERR_FSD_i = 1
                      LV_END_DIAG_FSD_i = 1
                      LV_END_DIAG_WIN_FSD_i = 1
              endif
      endif

elseif  (LV_CDN_DIAG_FSD_i = 1 and LV_MFF_AD_FAC_L_LIM_MAX_i = 1 and
                                                    LV_MFF_AD_FAC_L_i = 1)
then
      increment (T_SUM_MAX_FSD_i)
      ERR_SYM_FSD_i = 4
      if      T_SUM_MAX_FSD_i > C_T_SUM_MAX_MAX_FSD
      then
              LV_ERR_FSD_i = 1
              LV_END_DIAG_FSD_i = 1
              LV_END_DIAG_WIN_FSD_i = 1
      endif

endif
```

B.13.5.3 Error detection in case of lambda control - dead stop

```
if    (LV_CDN_DIAG_FSD_LAM_LIM_i = 1 and LV_LAM_LIM_MIN_i = 1)
then
      if      CTR_STOP_FSD < C_STOP_MAX_FSD
      then
              increment (T_SUM_MIN_FSD_LAM_LIM_i)
              ERR_SYM_FSD_LAM_LIM_i = 2
              if      T_SUM_MIN_FSD_LAM_LIM_i > C_T_SUM_MIN_MAX_LAM_LIM_FSD
              then
                      LV_ERR_FSD_LAM_LIM_i = 1
                      LV_END_DIAG_FSD_LAM_LIM_i = 1
                      LV_END_DIAG_WIN_FSD_LAM_LIM_i = 1
              endif
      endif

elseif  (LV_CDN_DIAG_FSD_LAM_LIM_i = 1 and LV_LAM_LIM_MAX_i = 1)
then
      increment (T_SUM_MAX_FSD_LAM_LIM_i)
      ERR_SYM_FSD_LAM_LIM_i = 1
      if      T_SUM_MAX_FSD_LAM_LIM_i > C_T_SUM_MAX_MAX_LAM_LIM_FSD
      then
```

```
            LV_ERR_FSD_LAM_LIM_i = 1
            LV_END_DIAG_FSD_LAM_LIM_i = 1
            LV_END_DIAG_WIN_FSD_LAM_LIM_i = 1
    endif

endif
```

B.13.5.4 Error detection in upper multiplicative adaptation learning area

```
if    (LV_CDN_DIAG_FSD_FAC_H_i = 1 and LV_MFF_AD_FAC_H_LIM_MIN_i = 1)
then
    if      CTR_STOP_FSD < C_STOP_MAX_FSD
    then
            increment (T_SUM_MIN_FSD_FAC_H_i)
            ERR_SYM_FSD_FAC_H_i = 2
            if      T_SUM_MIN_FSD_FAC_H_i > C_T_SUM_MIN_MAX_FSD
            then
                    LV_ERR_FSD_FAC_H_i = 1
                    LV_END_DIAG_FSD_FAC_H_i = 1
            endif
    endif

elseif (LV_CDN_DIAG_FSD_FAC_H_i = 1 and LV_MFF_AD_FAC_H_LIM_MAX_i = 1)
then
    increment (T_SUM_MAX_FSD_FAC_H_i)
    ERR_SYM_FSD_FAC_H_i = 1
    if      T_SUM_MAX_FSD_FAC_H_i > C_T_SUM_MAX_MAX_FSD
    then
            LV_ERR_FSD_FAC_H_i = 1
            LV_END_DIAG_FSD_FAC_H_i = 1
    endif

endif
```

B.13.6 Handling of operation counters

T_SUM_RST_FSD_xxx_i:anti wind-up counter

T_SUM_END_DIAG_WIN_FSD_xxx_i: counter for the diagnosis window

T_SUM_MIN_FSD_xxx_i: debouncing counter versus LV_MFF_AD_xxx_LIM_MIN_i

T_SUM_MAX_FSD_xxx_i: debouncing counter versus LV_MFF_AD_xxx_LIM_MAX_i

```
if    LV_CDN_DIAG_FSD_i = 1
then
    if      (ERR_SYM_FSD_i ≠ 0 and T_SUM_RST_FSD_i = 0)
    then
            T_SUM_END_DIAG_WIN_FSD_i = 0
    endif
    if      (ERR_SYM_FSD_i ≠ 0 or T_SUM_RST_FSD_i > 0}
    then
```

increment (T_SUM_RST_FSD_i)

endif

increment (T_SUM_END_DIAG_WIN_FSD_i)

if T_SUM_END_DIAG_WIN_FSD_i = C_T_SUM_RST_MAX_FSD

then

LV_END_DIAG_WIN_FSD_i = 1

LV_END_DIAG_FSD_i = 1

T_SUM_END_DIAG_WIN_FSD_i = 0

else

if LV_ERR_FSD_i = 0

then

LV_END_DIAG_WIN_FSD_i = 0

endif

endif

if (T_SUM_RST_FSD_i = C_T_SUM_RST_MAX_FSD **and** LV_ERR_FSD_i = 0)

then

T_SUM_MAX_FSD_i = 0

T_SUM_MIN_FSD_i = 0

T_SUM_RST_FSD_i = 0

endif

elseif LV_CDN_DIAG_FSD_FAC_H_i = 1
then

LV_END_DIAG_WIN_FSD_i = 0

if (ERR_SYM_FSD_FAC_H_i ≠ 0 **and** T_SUM_RST_FSD_FAC_H_i = 0)

then

T_SUM_END_DIAG_WIN_FSD_FAC_H_i = 0

endif

if (ERR_SYM_FSD_FAC_H_i ≠ 0 **or** T_SUM_RST_FSD_FAC_H_i > 0)

then

increment (T_SUM_RST_FSD_FAC_H_i)

endif

increment (T_SUM_END_DIAG_WIN_FSD_FAC_H_i)

if T_SUM_END_DIAG_WIN_FSD_FAC_H_i = C_T_SUM_RST_MAX_FSD
then

LV_END_DIAG_FSD_FAC_H_i = 1

T_SUM_END_DIAG_WIN_FSD_FAC_H_i = 0

endif

if (T_SUM_RST_FSD_FAC_H_i = C_T_SUM_RST_MAX_FSD **and** LV_ERR_FSD_FAC_H_i = 0)

then

T_SUM_MAX_FSD_FAC_H_i = 0

T_SUM_MIN_FSD_FAC_H_i = 0

T_SUM_RST_FSD_FAC_H_i = 0

endif

else

LV_END_DIAG_WIN_FSD_i = 0

endif

if LV_CDN_DIAG_FSD_LAM_LIM_i = 1
then

if (ERR_SYM_FSD_LAM_LIM_i ≠ 0 **and** T_SUM_RST_FSD_LAM_LIM_i = 0)
then

T_SUM_END_DIAG_WIN_FSD_LAM_i = 0

endif

if (ERR_SYM_FSD_LAM_LIM_i ≠ 0 **or** T_SUM_RST_FSD_LAM_LIM_i > 0)
then

increment (T_SUM_RST_FSD_LAM_LIM_i)

endif

increment (T_SUM_END_DIAG_WIN_FSD_LAM_i)

if T_SUM_END_DIAG_WIN_FSD_LAM_i = C_T_SUM_RST_MAX_FSD

then

LV_END_DIAG_FSD_LAM_LIM_i = 1

LV_END_DIAG_WIN_FSD_LAM_LIM_i = 1

T_SUM_END_DIAG_WIN_FSD_LAM_i = 0

else

if LV_ERR_FSD_LAM_LIM_i = 0

then

LV_END_DIAG_WIN_FSD_LAM_LIM_i = 0

endif

endif

if (T_SUM_RST_FSD_LAM_LIM_i = C_T_SUM_RST_MAX_FSD **and** LV_ERR_FSD_LAM_LIM_i = 0)

then

T_SUM_MAX_FSD_LAM_LIM_i = 0

T_SUM_MIN_FSD_LAM_LIM_i = 0

T_SUM_RST_FSD_LAM_LIM_i = 0

endif

else

LV_END_DIAG_WIN_FSD_LAM_LIM_i = 0

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_STOP_INC_FSD__TCO_ST	1*4	0...FFH	0...255	1	-
LDP_TCO_ST__STOP_INC_FSD	4	0...FEH	-48...142,5	0.75	°C
Increasing value for the debounce counter STOP_FSD					
ID_STOP_DEC_FSD__TOIL_MAX	1*4	0...FFH	0...255	1	-
LDP_TOIL_MAX__STOP_DEC_FSD	4	0...C8H	-40...160	1	°C
Decreasing value for the debounce counter STOP_FSD					
C_STOP_MAX_FSD	1	0...FFH	0...255	1	-
Maximum value for the debounce counter STOP_FSD					
C_T_SUM_MAX_MAX_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Maximum value of counter for ADD- , FAC_L- and FAC_H adaptation area (upper limit)					
C_T_SUM_MIN_MAX_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Maximum value of counter for ADD- , FAC_L- and FAC_H adaptation area (lower limit)					
C_T_SUM_MAX_MAX_LAM_LIM_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Maximum value of counter in case of lambda control - dead stop (upper limit)					
C_T_SUM_MIN_MAX_LAM_LIM_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Maximum value of counter in case of lambda control - dead stop (lower limit)					
C_T_SUM_RST_MAX_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Time counter threshold for T_SUM_RST_FSD_xxx_i and T_SUM_END_DIAG_WIN_FSD_xxx_i					
C_T_LAM_LIM_MFF_AD	1	1...FFFFH	0.02...1310.7	0.02	s
Time counter threshold for forced activation of lambda adaptation					
C_T_MFF_AD_ACT_MIN_FSD	1	1...FFFFH	0.02...1310.7	0.02	s
Minimum threshold of active forced lambda adaptation to reset the request flag					
C_FAC_LAM_OUT_MAX_FSD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
The upper limit of lambda controller output to request forced lambda adaptation					
C_FAC_LAM_OUT_MIN_FSD	1	8000...7FFFH	-50...49.998474	1.5259E-3	%
The lower limit of lambda controller output to request forced lambda adaptation					
C_N_MIN_FSD	1	0...FFH	0...8160	32	rpm
Engine speed threshold for fuel system diagnosis					
C_MAF_MIN_FSD	1	0...FFH	0...1389	5.45	mg/str.
Mass air flow threshold for fuel system diagnosis					
C_TCO_MIN_FSD	1	0...FEH	-48...142.5	0.75	°C
Minimum threshold of TCO for fuel system diagnosis					
C_AMP_MIN_FSD	1	0...FFFFH	0...5434	0.083	hPa
Minimum ambient pressure threshold for fuel system diagnosis					
C_TIA_MIN_FSD	1	0...FEH	-48...142.5	0.75	°C
Minimum intake air temperature threshold for fuel system diagnosis					

B.14 Application Incidences for fuel system diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_FSD	V/O	0...1H	0...1	1	-
Fuel system diagnosis: application incidence to inhibit the diagnosis					
LV_INH_DIAG_FSD_CP	V/O	0...1H	0...1	1	-
Deactivation bit for fuel system diagnosis due to internal states of the CP-function					
TOIL_MDL_FSD	V/O	0...C8H	-40...160	1	°C
calculated Oil Temperature Model For FSD					
LV_END_DIAG_WIN_FSD_FAC_H_i	V/O	0...1H	0...1	1	-
End of diagnosis cycle for similar conditions window in upper multiplicative adaptation learning area					

Input data:

LV_ERR_TPS	LV_ERR_TCO	LV_ERR_SA_SAV	LV_ERR_MAF
LV_ERR_CAM_IN	LV_ERR_SAP	LV_ERR_SAV	LV_ERR_MEC_IVVT_EX_i
LV_ERR_EL_CPS	LV_ERR_RATIO_CHK	LV_MIS_STATE_A	LV_IGK
LV_ERR_MEC_IVVT_IN_i	CL_MMV	LV_MIS_STATE_B1	LV_ERR_CAM_DE_IVVT_EX_i
LV_ERR_MAP	LV_ERR_CRK	LV_MIS_STATE_B4	LV_ERR_IGC
CP_STATE	LV_ERR_SLV_IVVT_IN_i	LV_ERR_SLV_IVVT_EX_i	LV_ERR_MEC_OPEN_CPS
LV_ERR_CAM_DE_IVVT_IN_i	TCO_ST		LV_ERR_IV[NC_CYL_NR]
FAC_LAM_COR_1_ENVD	MFF_ADD_LAM_AD_ENVD_L	FAC_L_RNG_LAM_AD_ENVD_H	FAC_H_RNG_LAM_AD_ENVD_H
VLS_UP_1_ENVD_L	LV_LAM_AD_ENA	LV_ERR_RLY_EFP	LV_END_DIAG_FSD_FAC_H_i

FUNCTION DESCRIPTION:

General information:

LV_INH_DIAG_FSD represents the influence by errors deactivating the Fuel system diagnosis.
LV_INH_DIAG_FSD_CP respects a maximum threshold for the charcoal canister load in specific states of STATE_CP.
If LC_INH_FSD_MAN_DEAC is 1 the FSD diagnosis is not inhibited by a present error.

Description:

Application conditions:

Initialisation: *at LV_IGK = 0 -> 1, reset and clear error memory*

all variables are initialised with 0

Recurrency: *20ms*

Activation: *at all engine states*

Formula section:

if	**(**LV_ERR_SAV = 1	(Secondary air valve electrical error) **or**
	LV_ERR_SAP = 1	(Secondary air pump electrical error) **or**
	LV_ERR_SA_SAV = 1	(Secondary air mechanical jammed valve) **or**
	LV_ERR_IGC = 1	(Ignition coil error) **or**
	LV_ERR_MAP = 1	(Manifold air pressure sensor error) **or**
	LV_ERR_SLV_IVVT_IN_i = 1	(IVVT inlet solenoid error) **or**
	LV_ERR_MEC_IVVT_IN_i = 1	(IVVT mechanics violation inlet error) **or**
	LV_ERR_CAM_DE_IVVT_IN_i = 1	(IVVT enduring inlet CAM deviation error) **or**
	LV_ERR_RATIO_CHK = 1	(Error - MAF or TPS ratio check) **or**
	LV_ERR_SLV_IVVT_EX_i = 1	(IVVT exhaust solenoid error) **or**
	LV_ERR_MEC_IVVT_EX_i = 1	(IVVT mechanics violation exhaust error) **or**
	LV_ERR_CAM_DE_IVVT_EX_i = 1	(IVVT enduring exhaust CAM deviation e) **or**
	LV_ERR_TPS = 1	(Throttle position sensor error) **or**
	LV_ERR_TCO = 1	(Coolant temperature sensor error) **or**
	LV_ERR_MAF = 1	(Mass airflow sensor error) **or**
	LV_ERR_CAM_IN = 1	(Camshaft sensor error) **or**
	LV_MIS_STATE_A = 1	(Misfire Catalyst damage) **or**
	LV_MIS_STATE_B1 = 1	(CARB B1 misfire criterion) **or**
	LV_MIS_STATE_B4 = 1	(CARB B4 misfire criterion) **or**
	LV_ERR_EL_CPS = 1	(Canister purge valve electrical error) **or**
	LV_ERR_CRK = 1	(Crank shaft sensor error) **or**

(LV_ERR_RLY_EFP = 1 AND LC_ENA_FSD_INH_RLY_EFP = 1)

(Fuel pump relay error; LC_ENA_FSD_INH_RLY_EFP allows to inhibit FSD by RLY_EFP error. If the RLY_EFP error is not MIL relevant, this switch must be set to 0, because a non MIL relevant error can not inhibit a MIL relevant error) **or**

LV_ERR_MEC_OPEN_CPS = 1	(Canister purge valve mechanical error) **or**	
LV_ERR_IV[NC_CYL_NR] = 1**)**	(Injection valve error) **or**	**and**

LC_INH_FSD_MAN_DEAC = 0

then

LV_INH_DIAG_FSD = 1 (Fuel system diagnosis inhibited)

else

LV_INH_DIAG_FSD = 0 (Fuel system diagnosis not inhibited)

Endif

LV_END_DIAG_WIN_FSD_FAC_H_i = LV_END_DIAG_FSD_FAC_H_i

Consideration of the charcoal canister saturation

if **{(**CP_STATE = MAX_PURGE (phase of NORMAL_PURGE)

and CL_MMV ≤ C_CL_MMV_MAX_FSD *(Normal State of CP-Function active and **low** canister saturation determined)**)***

or CP_STATE = CP_NOT_ACTIVE

or CP_STATE = NO_PURGE

or (CP_STATE = RAMP_CLOSE

and LV_LAM_AD_ENA = 1)**}**

then

LV_INH_DIAG_FSD_CP = 0

else

LV_INH_DIAG_FSD_CP = 1

endif

Oil temperature calculation (TOIL_MDL_FSD)

The oil temperature model calculation is performed through an interpolated table depending on engine operating point (N_32, MAF) because of the high thermal inertia.

Application conditions:

Initialisation : At ECU-Reset

TOIL_MDL_FSD = TCO

Recurrence : 1 Second

Activation : LV_IGK = 1

Deactivation : LV_IGK = 0

Formula section:

IF LV_ES = 1
THEN
(at engine stop, TOIL_MDL_FSD = TCO_ST)
IF TCO_ST ≥ C_TOIL_MDL_FSD_INI
THEN
TOIL_MDL_FSD = C_TOIL_MDL_FSD_INI
ELSE
TOIL_MDL_FSD = TCO_ST
ENDIF
ELSE

(at engine running, calculate TOIL_MDL_FSD)

$TOIL\_MDL\_FSD_n$ =

$TOIL\_MDL\_FSD_{n-1}$ + C_TOIL_MDL_FSD_CRLC*

(IP_TOIL_MDL_FSD__N_32__MAF - $TOIL\_MDL\_FSD_{n-1}$)

ENDIF

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Fuel System Diagnosis relevant to malfunction indication light	Max. limit (additive)	SYM_0	*NO*
	Min. limit (additive)	SYM_1	
	Max. limit (multiplicative; lower area)	SYM_2	
FSD_1	Min. limit (multiplicative; lower area)	SYM_3	

Diagnostic FSD_FAC_H_1	Symptom description	Symptom	Filter type
Diagnosis in upper multiplicative adaptation learning area	maximum limit of multiplicative adaptation value (upper area) reached	SYM_0	*NO*
	minimum limit of multiplicative adaptation value (upper area) reached	SYM_1	
		SYM_2	
		SYM_3	

Diagnostic FSD_LAM_LIM_1	Symptom description	Symptom	Filter type
Diagnosis in the area where the lambda control output concerned - relevant to MIL	lambda control in dead stop (upper limit)	SYM_0	*NO*
	lambda control in dead stop (lower limit)	SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CL_MMV_MAX_FSD	1	0H...FFFFH	0...8	0.000122	-
Maximum value for CL_MMV to permit FSD					
C_TOIL_MDL_FSD_CRLC	1	0H ... FFFFH	0...0.9999	1.5258E-5	-
Filtering factor for oil temperature model calculation					
C_TOIL_MDL_FSD_INI	1	0H...C8H	-40...160	1	°C
Oil temperature initialisation					
IP_TOIL_MDL_FSD__N_32__MAF	4*4	0H ...C8H	-40...160	1	°C
LDP_N_32_IP_TOIL_MDL_FSD	4	0H ... FFH	0 ... 8160	32	rpm
LDP_MAF_IP_TOIL_MDL_FSD	4	0H ... FFH	0 ... 1389	5.45	mg/stk
Oil temperature model in steady engine operating conditions					
LC_ENA_FSD_INH_RLY_EFP	1	0H...1H	0...1	1	-
Switch to enable FSD inhibition when RLY_EFP error is set					
LC_INH_FSD_MAN_DEAC	1	0H...1H	0...1	1	-
Manual deactivation of fuel system diagnosis inhibition					

B.15 Limited dynamic for catalyst efficiency and oxygen sensor diagnosis

Output data:

Name	V/S	Hex. limits	Phys. limits	Resol.	Unit
LV_LDC_CAT_i	V/O	0...1H	0...1	1	-
status limited dynamic					
N_MMV_CAT	V	0...1FE0H	0...8160	1	rpm
floating mean value for limited engine speed					
MAF_MMV_CAT	V	0...FFFFH	0...1389	0,021	mg/stk
floating mean value for limited MAF					
LAM_MV_MMV_CAT_i	V	8000...7FFFH	-50...50	1,5*10-3	%
floating mean value for limited lambda deviation					
TPS_GRD_SUM_i	V	0...FFFFH	0...767916	11,72	°TPS/s
sum. value for limited TPS gradient					
MAF_INT_LDC_CAT_i	V/O	0...FFFFH	0...1820.393	0,028	g
integral of mass air flow since limited dynamic conditions fulfilled					
LV_N_LDC	V	0...1H	0...1	1	-
Status limited dynamic N					
LV_MAF_LDC	V	0...1H	0...1	1	-
Status limited dynamic MAF					
LV_LAM_MV_LDC_i	V	0...1H	0...1	1	-
Status limited dynamic LAM_MV_i					
LAM_P_CTR_LDC_CAT	V	0...FFH	0...255	1	-
P-jump delay counter for LV_LDC_CAT					

Input data:

N	MAF	TPS_GRD	LAM_MV_i
LV_IGK	LV_ST_END	MAF_KGH	LV_AFL[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

For a twin branch exhaust line the calculation of the limited dynamic condition LV_LDC_CAT_i has the same structure for both exhaust lines and is calculated for each line separately. To differentiate variables and calibration data the index i is used.

i = 1, for cylinder bank 1
i = 2, for cylinder bank 2

The limited dynamic conditions LV_LDC_CAT_i for catalyst efficiency and OBDII oxygen sensor diagnosis exists and is set high if the limited dynamic conditions for engine speed (1.1), mass air flow (1.2) and LAM_MV_i (1.3) exist and the threshold C_MAF_INT_MIN_CAT for the integral of the air flow is exceeded. Moreover the summation of the trottel position gradient TPS_GRD_SUM_i must not exceed C_TPS_GRD_SUM_MAX.

Application conditions:

Recurrence: Ta = 0.02 s

Activation: LV_ES = 0 **AND** LV_ST = 0 % after leaving the engine state ST

Deactivation: LV_ES = 1

B.15.1 Limited engine speed (N) dynamics

The floating mean value N_MMV_CAT is computed using the averaging constant C_N_CRLC_CAT. The limited dynamics condition only exist while the engine speed N lies within the dynamics window around N_MMV_CAT:
If the above limited dynamics condition is violated, the floating mean value N_MMV_CAT is set to the current engine speed in order to reach the limited dynamics condition faster:

Formula section:
N_MMV_CAT = N_MMV_CAT * (1 -C_N_CRLC_CAT) + C_N_CRLC_CAT * N

```
IF | N - N_MMV_CAT | < C_N_DYW_CAT
THEN
    LV_N_LDC = 1
ELSE
    LV_N_LDC = 0
    N_MMV_CAT = N
END
```

B.15.2 Limited mass air flow (MAF) dynamics

The floating mean value MAF_MMV_CAT is computed using the averaging constant C_MAF_CRLC_CAT. The limited dynamics condition only exist while the air mass MAF stays within the dynamics window around MAF_MMV_CAT:
If the above limited dynamics condition is violated, the floating mean value MAF_MMV_CAT is set to the current air-mass value in order to reach the limited dynamics condition faster:

Formula section:
MAF_MMV_CAT = MAF_MMV_CAT * (1 - C_MAF_CRLC_CAT) +
C_MAF_CRLC_CAT * MAF

```
IF | MAF - MAF_MMV_CAT | < C_MAF_DYW_CAT
THEN
    LV_MAF_LDC = 1
ELSE
    LV_MAF_LDC = 0
    MAF_MMV_CAT = MAF
END
```

B.15.3 Limited mean oxygen value (LAM_MV_i) dynamics

The floating mean value LAM_MV_MMV_CAT_i is computed using the averaging constant C_LAM_MV_CRLC_CAT:

The limited dynamics condition continues to exist while the mean oxygen value LAM_MV_i stays within the dynamics window around LAM_MV_MMV_CATi:
If the above limited dynamics condition is violated, the floating mean value LAM_MV_MMV_CAT_i is set to the current mean oxygen value in order to reach the limited dynamics condition faster:

Formula section:

```
LAM_MV_MMV_CAT_i  =  LAM_MV_MMV_CAT_i * (1 - C_LAM_MV_CRLC_CAT) +
                     C_LAM_MV_CRLC_CAT * LAM_MV_i


IF | LAM_MV_i - LAM_MV_MMV_CAT_i | < C_LAM_MV_DYW_CAT
THEN
    LV_LAM_MV_LDC_i = 1
ELSE
    LV_LAM_MV_LDC_i = 0
    LAM_MV_MMV_CAT_i  = LAM_MV_i
END
```

B.15.4 Limited dynamic bit: LV_LDC_CAT_i

If C_TPS_GRD_SUM_MIN is larger than C_TPS_GRD_SUM_MAX the TPS gradient summation TPS_GRD_SUM will be no conditon to determine the limited dynamic bit LV_LDC_CAT_i

```
IF   LV_N_LDC = 1 AND LV_MAF_LDC = 1 AND LV_LAM_MV_LDC_i = 1
THEN
     IF       MAF_INT_LDC_CAT_i =<  C_MAF_INT_MIN_CAT
     THEN LAM_P_CTR_LDC_CAT       = 0
              MAF_INT_LDC_CAT_i[n+1]  = MAF_INT_LDC_CAT_i[n] +
                                        MAF_KGH * NC_FAC_MAF_INT
                  % NC_FAC_MAF_INT = 20ms/3,6
     ELSE
              TPS_GRD_SUM_i[n+1] = TPS_GRD_SUM_i[n] + |TPS_GRD|
              T_TPS_GRD_DEC_SUM_i = T_TPS_GRD_DEC_SUM_i + 1

              IF  T_TPS_GRD_DEC_SUM_i > C_TPS_GRD_DEC_SUM
              THEN
                  TPS_GRD_SUM_i[n+1] = TPS_GRD_SUM_i[n] –
                                             C_TPS_GRD_DEC
                          % The decrementation by C_TPS_GRD_DEC is done all
                          % C_TPS_GRD_DEC_SUM calculation intervalls
                  T_TPS_GRD_DEC_SUM_i = 0
              END

              IF      C_TPS_GRD_SUM_MIN < C_TPS_GRD_SUM_MAX
              THEN  % TPS_GRD_SUM is condition for limited dynamic bit
                      IF      LV_LDC_CAT_i = 0
                      THEN  %  limited dynamic bit deactive, activation possible
                              IF      TPS_GRD_SUM_i <= C_TPS_GRD_SUM_MIN
                              THEN  if       LV_AFL = 0 → 1 or LV_AFL = 1 → 0
                                    then     LAM_P_CTR_LDC_CAT(n) =
                                             LAM_P_CTR_LDC_CAT(n-1)  + 1
                                             until C_LAM_P_CTR_LDC_CAT
                                    if       LAM_P_CTR_LDC_CAT ≥ C_LAM_P_CTR_LDC_CAT
                                             % activation of limited dynamic bit
                                    then     LV_LDC_CAT_i = 1
                              END
                      ELSE  % limited dynamic bit active, deactivation possible
                              IF      TPS_GRD_SUM_i > C_TPS_GRD_SUM_MAX
                              THEN  % deactivation of limited dynamic bit
                                    LV_LDC_CAT_i = 0
                                    LAM_P_CTR_LDC_CAT       = 0
                              END
                      END
              ELSE
                      LV_LDC_CAT_i = 1
              ENDIF
     ENDIF
ELSE
     LV_LDC_CAT_i              = 0
     TPS_GRD_SUM               = 0
     MAF_INT_LDC_CAT_i         = 0
     T_TPS_GRD_DEC_SUM_i   = 0
```

LAM_P_CTR_LDC_CAT = 0
ENDIF

*** Remark**
The update rate of LV_AFL is 10 ms and LAM_P_CTR_LDC_CAT is 20 ms.
But, it will not make a problem because normally lean/rich half duration will be surely longer than 20ms and it will be even better to do not count for LV_AFL jittering.

The P-jump counter to set LV_LDC_CAT is used only if C_TPS_GRD_SUM_MIN < C_TPS_GRD_SUM_MAX as otherwise LV_LDC_CAT is set immediately

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_CRLC_CAT	-	0...FFFFH	0...1	1,5e-5	-
floating mean value calculation engine speed					
C_MAF_CRLC_CAT	-	0...FFFFH	0...1	1,5e-5	-
floating mean value calculation MAF					
C_LAM_MV_CRLC_CAT	-	0...FFFFH	0...1	1,5e-5	-
floating mean value calculation lambda					
C_N_DYW_CAT	-	0...1FE0H	0...8160	1	rpm
threshold limited dynamic engine speed					
C_MAF_DYW_CAT	V	0...FFFFH	0...1389	0,021	mg/stk
threshold limited dynamic engine MAF					
C_LAM_MV_DYW_CAT	-	0...FFFFH	0...100	1,5e-3	%
threshold limited dynamic lambda					
C_TPS_GRD_DEC	-	0...FFH	0...2988	11,72	°TPS/s
decrement for calculation TPS gradient					
C_TPS_GRD_SUM_MAX	-	0...FFFFH	0...767916	11,72	°TPS/s
max threshold limited dynamic TPS gradient					
C_TPS_GRD_SUM_MIN	-	0...FFFFH	0...767916	11,72	°TPS/s
min threshold limited dynamic TPS gradient					
C_TPS_GRD_DEC_SUM	-	0...FFH	0...5,1	0,02	s
time periods for calculation limited dynamic TPS gradient					
C_MAF_INT_MIN_CAT	-	0...FFFFH	0...1820,393	0,028	g
MAF integral after setting limited dynamic conditions LV_LDC_CATi before starting the monitoring cycle					
C_LAM_P_CTR_LDC_CAT	-	0...FFH	0...255	1	-
P-jump counter delay for LV_LDC_CAT					

B.16 Oxygen Sensor Diagnosis Management

B.16.1 Upstream oxygen sensor final diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean error flag, fault currently present on upstream oxygen sensor signal					

Input data:

LV_ERR_EL_LS_UP[NC_CBK_EX_NR]	LV_IGK	LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]	LV_ERR_STK_LS_UP[NC_CBK_EX_NR]
LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]
NC_CBK_EX_NR	LV_ST_END	LV_ERR_CHG_LS_UP[NC_CBK_EX_NR]	

FUNCTION DESCRIPTION:

This function summarizes the result of all diagnosis related to the upstream oxygen sensor, which are:

- The upstream sensor heater diagnosis:
  - Electrical failure LV_ERR_LSH_UP[i];
  - OBD2 Failure LV_ERR_OBD_LSH_UP[i];

- The oxygen sensor signal electrical diagnosis (bit LV_ERR_EL_LS_UP[i]), which shall detect the following faults:
  - Short circuit of the oxygen sensor signal line to GND
  - Short circuit of the oxygen sensor signal line to Vbatt
  - Open circuit (line break) in the connection to oxygen sensor element

- The following signal plausibility checks shall be carried out:
  - Overrun fuel cut-off (PUC) oxygen sensor signal plausibility
  - Oxygen sensor signal voltage excursion plausibility

- The OBD2 signal diagnosis:
  - Oxygens sensor switching time monitoring
  - Oxygens sensor control frequency check

- The oxygen sensor swapped signal diagnosis. This diagnosis is intended to detect swapped upstream sensors for systems with two exhaust banks. For one bank system this diagnosis is not active, i.e. LV_ERR_CHG_LS_UP[i] = 0.

LV_DIAG_PLAUS_SYM_LSL_UP[i] is generated to serve a common interface to Appl. Inc. of heater management.

Application conditions:

Initialisation:

At LV_IGK = 0 to 1, reset **or** <u>at clearing error memory</u>:

LV_ERR_LS_UP[i] = 0

Recurrence:

This function shall be carried out every 100ms.

Activation / Deactivation:

The diagnosis will be carried out only if LV_ST_END = 1

Formula section:

If LC_LS_UP_DIAG_ACT[i] = 1
and (LV_ERR_EL_LS_UP[i] = 1
or LV_ERR_PUC_LS_UP[i] = 1
or LV_ERR_STK_LS_UP[i] = 1
or LV_ERR_SWT_LS_UP[i] = 1
or LV_ERR_FRQ_LS_UP[i] = 1
or LV_ERR_LSH_UP[i] = 1
or LV_ERR_OBD_LSH_UP[i] = 1
or LV_ERR_CHG_LS_UP[i] = 1
or LC_LS_UP_ERR[i] = 1)
then LV_ERR_LS_UP[i] = 1
else LV_ERR_LS_UP[i] = 0
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_LS_UP_DIAG_ACT[i]	1	0...1H	0...1	1	-
Manual activation bit for upstream oxygen sensor final diagnosis					
LC_LS_UP_ERR[i]	1	0...1H	0...1	1	-
Opportunity for manual setting of LV_ERR_LS_UP[i] = 1					

B.16.2 Downstream oxygen sensor final diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_LS_DOWN[NC_CBK_EX_NR]	V/O	0H...1H	0...1	1	-
Final diagnostic of the downstream oxygen sensor					

Input data:

LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]	LV_ST_END	LV_ERR_SWT_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_PUE_LS_DOWN[NC_CBK_EX_NR]
NC_CBK_EX_NR	LV_IGK	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]	LV_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

This function summarizes the result of all diagnosis related to the downstream oxygen sensor, which are:

- The downstream sensor heater diagnosis:
  - Electrical failure LV_ERR_LSH_DOWN[i];
  - OBD2 Failure LV_ERR_OBD_LSH_DOWN[i];

- The oxygen sensor signal electrical diagnosis (bit LV_ERR_EL_LS_DOWN[i]), which shall detect the following faults:
  - Short circuit of the oxygen sensor signal line to GND
  - Short circuit of the oxygen sensor signal line to Vbatt
  - Open circuit (line break) in the connection to oxygen sensor element

- The following signal dynamic and plausibility checks shall be carried out:
  - Monitoring the rich-lean switching times in the trailing throttle fuel cut-off (LV_ERR_SWT_LS_DOWN[i])
  - Monitoring the sensor voltage after leaving trailing throttle fuel cut-off (LV_ERR_PUE_LS_DOWN[i])

Application conditions:

Initialisation:

At LV_IGK = 0 to 1, reset **or** at clearing error memory:

LV_ERR_LS_DOWN[i] = 0

Recurrence:

The function should be carried out once every 100 ms.

Activation / Deactivation:

The diagnosis will be carried out only if LV_ST_END = 1

Formula section:

If	LC_LS_DOWN_DIAG_ACT[i] = 1
and	(LV_ERR_EL_LS_DOWN[i]= 1
or	LV_ERR_SWT_LS_DOWN[i] = 1
or	LV_ERR_PUE_LS_DOWN[i] = 1
or	LV_ERR_LSH_DOWN[i] = 1
or	LV_ERR_OBD_LSH_DOWN[i] = 1
or	LC_LS_DOWN_ERR[i] = 1)
then	LV_ERR_LS_DOWN[i] = 1
else	LV_ERR_LS_DOWN[i] = 0
endif	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_LS_DOWN_DIAG_ACT[i]	1	0...1H	0...1	1	-
Manual activation bit for downstream oxygen sensor final diagnosis					
LC_LS_DOWN_ERR [i]	1	0...1H	0...1	1	-
Opportunity for manual setting of LV_ERR_LS_DOWN[i] = 1					

B.16.3 Definition of interface flags

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_OBD_VLD_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating the validation of correspondent symptom (transmitted to the error management)					
LV_ERR_TTIP_OBD_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that fault is present and has been debounced					

Input data:

NC_CBK_EX_NR	LV_ST_END		

FUNCTION DESCRIPTION:

General information:

Heater OBD2 TTIP error can occur not only due to heater malfunctioning ageing but also due to an *open circuit* in certain sensor lines, therefore this function shall be consulted to validate a symptom given by the WRAF sensor dynamic diagnosis. For BIN/BIN systems the flag is therefore set to 0 as dummy flag.

Application conditions:

Initialisation: at reset or LV_ST_END = 1 all variables are set = 0

Recurrence: This function shall run every 1s (if formula text is added).

Activation / Deactivation:

The diagnosis will be carried out only if LV_ST_END = 1

Formula section:

No formula text defined up to now

B.17 Oxygen Sensor Interchanged Signal Diagnosis

B.17.1 Upstream Sensor Banks 1 and 2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CHG_LS_UP	V/0	0...1H	0...1	1	-
Debounced error: upstream sensor of banks 1and 2 interchanged					

FUNCTION DESCRIPTION:

General information:

This function is a dummy specification for one bank systems.

Application conditions:

Initialisation: at reset

LV_ERR_CHG_LS_UP = 0

B.17.2 Downstream Sensor Banks 1 and 2

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAG_SYM_CHG_LS_DOWN[NC_CBK_EX_NR]	V/0	0...1H	0...1	1	-
Debounced error: downstream sensor of banks 1and 2 interchanged					

FUNCTION DESCRIPTION:

General information:

This function is a dummy specification for one bank systems.

Application conditions:

Initialisation: at reset

LV_DIAG_SYM_CHG_LS_DOWN[i] = 0

B.18 Appl. Incidences of “Oxygen Sensor Interchanged Signal Diagnosis”

B.18.1 “Upstream sensor banks 1 and 2”

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_CHG_LS_UP	V/O	0...1H	0...1	1	-
Inhibit diagnosis interchaged upstream sensors					

Application conditions:

Initialisation:

At reset all variables shall be initialised = 0

B.19 Upstream oxygen sensor heater OBDII monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that conditions for diagnosis met					
ERR_SYM_OBD_LSH_UP[NC_CBK_EX_NR]	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Variable indicating status of each symptom, updated each recurrence & at diagnosis completion					
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that fault is present and has been debounced					
LV_END_DIAG_OBD_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean flag indicating that diagnosis has been completed					
STATE_OBD_LSH_UP[NC_CBK_EX_NR]	V	00H 01H 02H 03H 04H 05H 06H 07H 08H	DIAG_OFF DIAG_INIT LS_READY OBD_1_DLY OBD_1_CHK TEG_THD POW_INT DIAG_ACT DIAG_END	1	-
Boolean flag indicating that diagnosis has been completed					
TEMP_DIF_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...07FFH	0...2047	1	K
Indicates difference in temperature between set operating temperature & exhaust gas at sensor location					
POW_INT_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	J
Integral indicates measure of cooling energy of exhaust gas at sensor location					
TCC_VLD_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Number of completed diagnosis cycles					
TCC_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Number of diagnosis cycles in which temperature deviation exceeding limits present					
CTR_CYCNR_R_IT_OBD_LS_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
Number of diagnosis cycles in which temperature deviation exceeding limits present					
R_IT_THD_OBD_LSH_UP[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...65535	1	Ω
Threshold for most current upstream oxygen sensor internal resistance used for diagnosis cycle, Mode 6 information					
R_IT_OBD_LSH_UP[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...65535	1	Ω
Most current upstream internal resistance used for diagnosis cycle (Mode 6 information)					
R_IT_L_MES_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω
Latest measured upstream oxygen sensor internal resistance value for no error case, mode 6 information					
R_IT_L_REF_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω
Latest reference value for measured internal resistance upstream in no error case, mode 6 information					
R_IT_H_MES_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω
Latest measured upstream oxygen sensor internal resistance value for error case, mode 6 information					
R_IT_H_REF_OBD_LSH_UP[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	Ω

Latest reference value for measured internal resistance upstream in error case, mode 6 information

Input data:

LV_ST_END	R_IT_LS_UP[NC_CBK_EX_NR]]	MAF_KGH	LV_INH_DIAG_OBD_LSH_UP[NC_CBK_EX_NR]
STATE_LSH_UP[NC_CBK_EX_NR]	LV_LS_UP_READY[NC_CBKEX_NR]	LV_VB_CDN_OBD_2	LSHPWM_UP[NC_CBK_EX_NR]
	LV_ERR_LSH_UP[NC_CBK_EX_NR]	TEG_DYN_LS_UP[NC_CBK_EX_NR]	CTR_CYCNR_R_IT_LS_UP_VLD[NC_CBK_EX_NR]
LV_IGK	R_IT_MDL_LS_UP_NEW[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_ERR_SCP_LS_UP[NC_CBK_EX_NR]
LV_ERR_SCG_LS_UP[NC_CBK_EX_NR]	LV_ERR_OC_LS_UP[NC_CBK_EX_NR]	LV_DC	

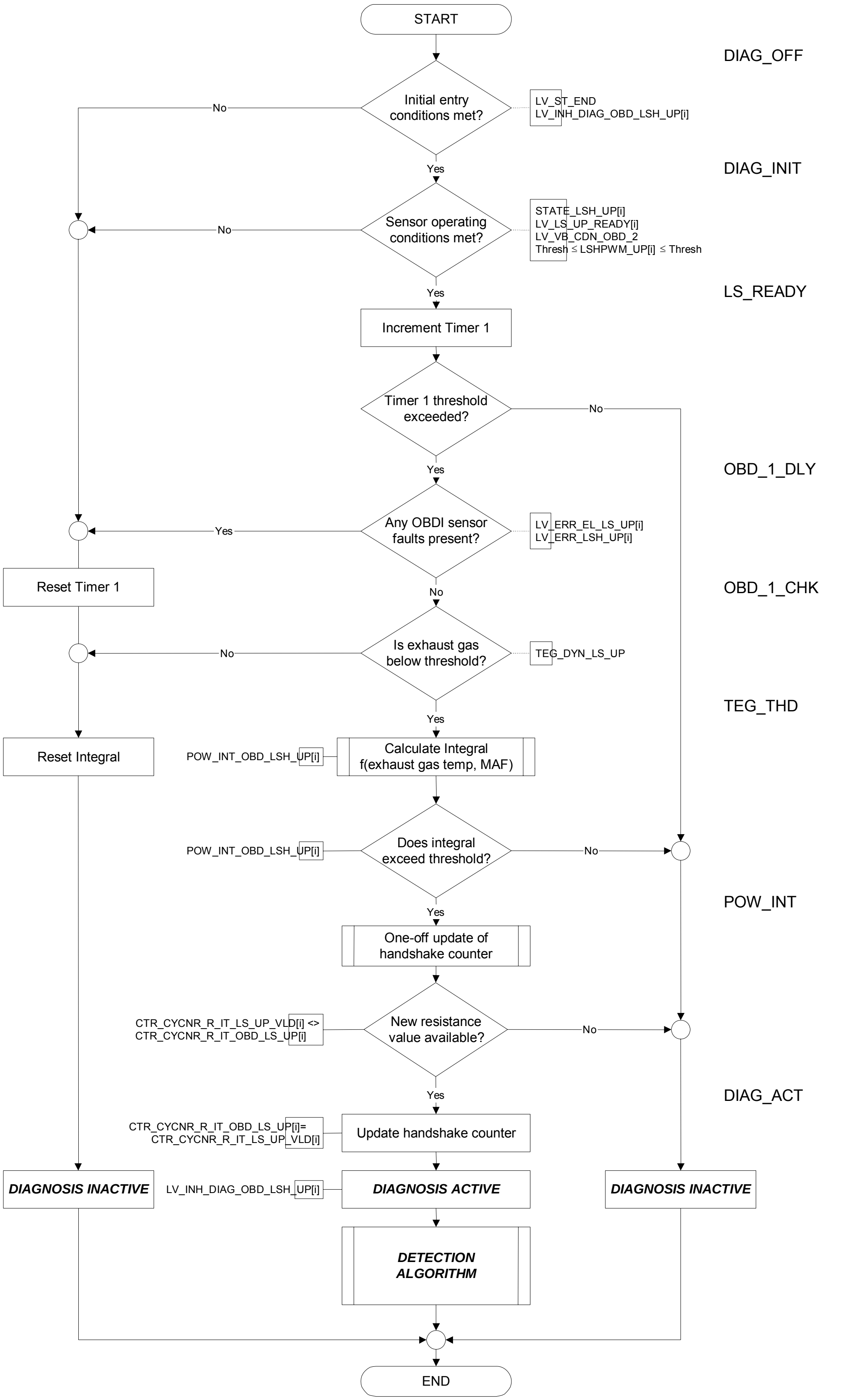
FUNCTION DESCRIPTION:

General information:

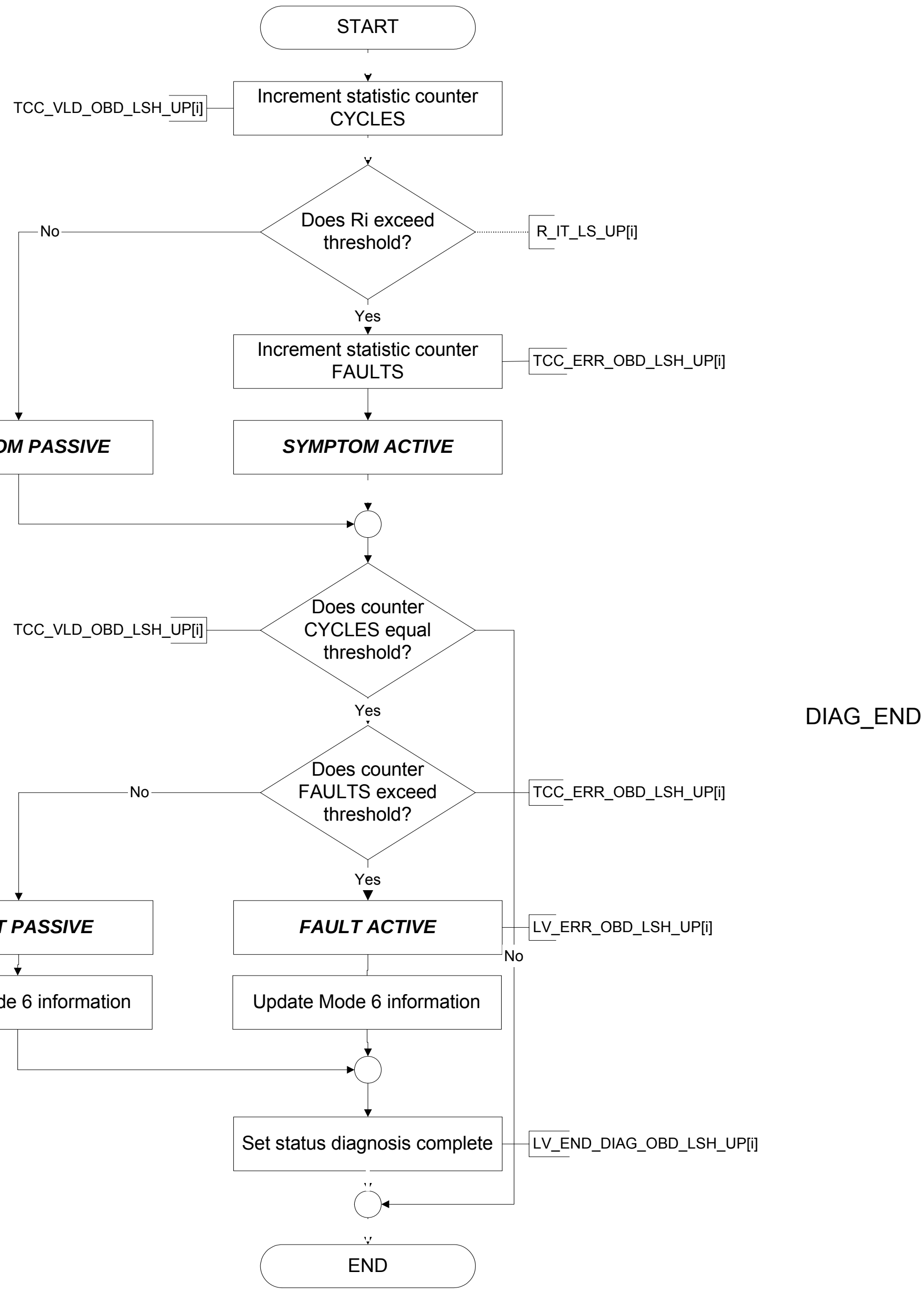
The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR= 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank.

Signal flow diagram:

Activation conditions:


START
DIAG_OFF
Initial entry conditions met?
LV_ST_END
LV_INH_DIAG_OBD_LSH_UP[i]
No
Yes
DIAG_INIT
Sensor operating conditions met?
STATE_LSH_UP[i]
LV_LS_UP_READY[i]
LV_VB_CDN_OBD_2
Thresh ≤ LSHPWM_UP[i] ≤ Thresh
No
Yes
LS_READY
Increment Timer 1
Timer 1 threshold exceeded?
No
Yes
OBD_1_DLY
Any OBDI sensor faults present?
LV_ERR_EL_LS_UP[i]
LV_ERR_LSH_UP[i]
Yes
No
Reset Timer 1
OBD_1_CHK
Is exhaust gas below threshold?
TEG_DYN_LS_UP
No
Yes
TEG_THD
Reset Integral
POW_INT_OBD_LSH_UP[i]
Calculate Integral f(exhaust gas temp, MAF)
POW_INT_OBD_LSH_UP[i]
Does integral exceed threshold?
No
Yes
POW_INT
One-off update of handshake counter
CTR_CYCNR_R_IT_LS_UP_VLD[i] <>
CTR_CYCNR_R_IT_OBD_LS_UP[i]
New resistance value available?
No
Yes
DIAG_ACT
CTR_CYCNR_R_IT_OBD_LS_UP[i]=
CTR_CYCNR_R_IT_LS_UP_VLD[i]
Update handshake counter
DIAGNOSIS INACTIVE
LV_INH_DIAG_OBD_LSH_UP[i]
DIAGNOSIS ACTIVE
DIAGNOSIS INACTIVE
DETECTION ALGORITHM
END

Detection Algorithm:



Description:

The upstream oxygen sensor heater circuit shall detect any loss in heater power that would cause a drop in the sensor operating temperature, thereby possibly causing exhaust gas emissions to rise above the applicable standards or prevent the sensor signal from being used as a diagnostic system monitoring device.

Losses in heater power may occur due to, for example, ageing of the heater element, defective wiring, increased heater cct. connector contact resistance, defective heater driver etc.

The hereafter mentioned diagnosis strategy is based on the comparison of the oxygen sensor internal resistance to an absolute threshold during operating conditions where the exhaust gas temperature has been determined to be sufficiently low as to cause the sensor ceramic temperature to fall outside normal operating levels, in cases where a power is insufficient.

The functionality of the diagnosis may be described in further detail below:

The initial diagnosis application condition; LV_ST_END shall ensure that the engine is running and has left the start phase but not yet entered the engine stop phase, LV_INH_DIAG_OBD_LSH_UP[i] shall ensure that the project specific application conditions (Application Incidences) have been met and LV_END_DIAG_OBD_LSH_UP[i] shall ensure that the diagnosis only be carried out once per driving cycle.

In order to ensure that the sensor has reached its normal operating temperature, the heater management state STATE_LSH_UP[i] shall be determined to be in the open loop control state LSH_POW_CTL, the sensor shall be determined to be in a state of operative readiness via LV_LS_UP_READY[i], the battery voltage VB shall be in a range determined to be fit for carrying out OBD2 diagnosis shown by LV_VB_CDN_OBD_2 and the heater duty-cycle LSHPWM_UP[i] shall be determined to be within calibrateable range C_LSHPWM_MIN_LSH_UP & C_LSHPWM_MAX_LSH_UP.

Should the above initial and operative conditions have been determined to be met, a timer, denoted herein as *TIMER_1*, shall be started. The timer shall run until a calibrateable threshold C_T_DLY_POW_CTL_OBD_LSH_UP is equalled or exceeded. Once this delay has passed, the status of the OBDI error bits LV_ERR_EL_VLS_UP[i] & LV_ERR_LSH_UP[i] shall be read to ensure that no signal electrical or plausibility faults and no heater electrical faults are present in the sensor cct. under test. The delay shall ensure that sufficient time passes to permit completion of the OBDI monitoring prior to the start of OBDII monitoring.

Should no OBDI faults be present, the modelled exhaust gas temperature TEG_DYN_LS_UP[i] shall be compared to a calibrateable threshold C_TEG_DYN_MAX_OBD_LSH_UP. Due tolerances in the determination of the sensor internal resistance and in the case of a functional heater, the maximum resistance value may be given by a minimum tolerance sensor at a much lower temperature than for a maximum tolerance sensor at a higher temperature. Therefore, in the case of diagnosis based on an absolute internal resistance threshold, a certain temperature tolerance exists. The above mentioned temperature threshold shall lie below the minimum temperature determined from analysis of the internal resistance tolerances and be sufficient to cool the sensor ceramic element to below normal operating conditions, in the case of a heater fault.

Should the exhaust gas temperature be determined to equal or fall below the threshold, the monitoring strategy shall ensure that the sensor be cooled sufficiently long by the exhaust gas to enable differentiation between a functional and non-functional heater cct. (The time required is dependent on the thermal capacity of the system at the sensor location and the forced convection cooling of the sensor location). This shall be achieved by integrating the heat flux of the forced convection thermal transfer between sensor and exhaust gas, this being a measure for the cooling energy of the exhaust gas. The heat flux shall be calculated by determining the difference between the typical set sensor operating temperature C_TEMP_SP_OBD_LSH_UP and the modelled exhaust gas temperature TEG_DYN_LS_UP[i] and multiplying the result by the factor IP_FAC_MAF_OBD_LSH_UP representing the heat transfer coefficient and surface area concerned in the thermal transfer. As the heat transfer coefficient is a function of the velocity of the exhaust gas, the factor shall be mapped and dependent on the current mass airflow MAF_KGH.

The resultant integral POW_INT_OBD_LSH_UP[i] shall be compared to a calibrateable threshold C_POW_INT_MIN_OBD_LSH_UP. Each time that the threshold be exceeded for the first time, e.g. first time after engine start or after the power integral has been reset, the CTR_CYCNR_R_IT_OBD_LS_UP[i] shall be updated once only with the value CTR_CYCNR_R_IT_LS_UP_VLD[i]. This shall ensure that only new internal resistance values be used for diagnosis purposes that have been determined since all the activation conditions have been met.

It shall be determined whether a new internal resistance has been determined by comparing the contents of counter CTR_CYCNR_R_IT_OBD_LS_UP[i] with that of CTR_CYCNR_R_IT_LS_UP_VLD[i]. Should the counters be unequal, then a new value is available and CTR_CYCNR_R_IT_OBD_LS_UP[i] shall be updated with CTR_CYCNR_R_IT_LS_UP_VLD[i]. If this conditions and all the above conditions be met, LV_END_DIAG_OBD_LSH_UP[i] shall be set and the diagnosis shall be considered to be active and the current monitoring cycle valid.

Once the current monitoring cycle has been determined to be valid, a counter TCC_VLD_OBD_LSH_UP[i] shall be incremented, denoting the number of valid monitoring cycles carried out to date. The internal resistance R_IT_LS_UP[i] shall be compared to a threshold IP_R_IT_THD_OBD_LSH_UP which is dependent on the modelled tip temperature of a new sensor (including safety margin) and the number of valid diagnosis cycles TCC_VLD_OBD_LSH_UP. Should the resistance equal or exceed the threshold, a heater fault shall be determined to be present in the current cycle and a counter TCC_ERR_OBD_LSH_UP[i] shall be incremented, denoting the number of heater faults within the valid monitoring cycles carried out to date.

In order to comply with OBDII requirements, the current R_IT_LS_UP[i] value used shall be made available to an external tester. This information is Mode 6 information. Hence once the comparison to the threshold has been carried out, the latest measured internal resistance and threshold values (R_IT_L_MES_OBD_LSH_UP[i] and R_IT_L_REF_OBD_LSH_UP[i]) shall be copied to the respective Mode 6 interface variables R_IT_OBD_LS_UP[i] and R_IT_THD_OBD_LSH_UP[i] for the no error case. In the error case the variables R_IT_H_MES_OBD_LSH_UP[i] and R_IT_H_REF_OBD_LSH_UP[i] shall be handed out.

Once a calibrateable number of valid monitoring cycles C_TCC_THD_OBD_LSH_UP has been determined to have been carried out, the diagnosis shall carry out a n-out-of-m statistical evaluation of the results. Should the number of determined heater faults exceed a calibrateable threshold C_TCC_THD_ERR_OBD_LSH_UP, the oxygen sensor heater under test shall be considered to be faulty for the current driving cycle and flag LV_ERR_OBD_LSH_UP[i] shall be set.

If the number of faults does not exceed the threshold, LV_ERR_OBD_LSH_UP[i] shall be reset.

Once the statistical evaluation has been carried out, no further monitoring cycles shall be executed in the remaining driving cycle. This is achieved by setting LV_END_DIAG_OBD_LSH_UP[i].

NOTES:

A. The integral POW_INT_OBD_LSH_UP[i] shall be reset to 0 should the exhaust gas temperature exceed the applicable threshold. This would cause the sensor to be re-heated from the exhaust gas once again and a certain delay would be required prior to the sensor being cooled long enough to permit valid diagnosis to occur.

B. The timer *TIMER_1* and integral POW_INT_OBD_LSH_UP[i] shall be reset should any of the initial entry or operative readiness conditions no longer be met or an OBDI fault be detected in the sensor under test. This may cause the sensor to leave its normal operating conditions and the OBDII diagnosis shall be required to start from the beginning as the sensor operating conditions would require time to stabilise should the conditions be met again.

C. The statistic counters TCC_VLD_OBD_LSH_UP[i] and TCC_ERR_OBD_LSH_UP[i] shall not be reset by changing conditions and therefore shall only increment or remain frozen during any given monitoring cycle. The counters shall only be reset at the beginning of a new driving cycle.

D. Projects that provide for an internal statistical error management shall use their internal error management and apply this wherever reference to TCC_VLD_OBD_LSH_UP[i], TCC_ERR_OBD_LSH_UP[i], C_TCC_THD_OBD_LSH_UP & C_TCC_THD_ERR_OBD_LSH_UP is made.

E. The variable STATE_OBD_LSH_UP[i] shall indicate, at the end of the each monitoring cycle, the status of the diagnosis within that cycle. The following correlation of variable content to description shall apply:

0:	DIAG_OFF	Initialisation state; no conditions met
1:	DIAG_INIT	Initial conditions & application incidences met only
2:	LS_READY	Sensor operatively ready conditions met + 1,
3:	OBD_1_DLY	OBDI monitoring completion delay exceeded + 1 & 2
4:	OBD_1_CHK	No OBDI faults present +1, 2 & 3
5:	TEG_THD	Exhaust gas below threshold + 1, 2, 3 & 4
6:	POW_INT	Power integral exceeded + 1,2,3,4 & 5
7:	DIAG_ACT	***Diagnosis detection active***
8:	DIAG_END	***Diagnosis detection complete***

Application conditions:

Initialisation:

The following variables (Mode "S") shall be initialised at checksum error:

LV_ERR_OBD_LSH_UP[i] = 0

R_IT_THD_OBD_LSH_UP[i] = 65535

R_IT_OBD_LSH_UP[i] = 65535

The following variables shall be initialised at reset, at the transition ignition key on (i.e. LV_IGK = 0 -> 1) and upon clearing error memory:

LV_CDN_DIAG_OBD_LSH_UP[i] = 0
ERR_SYM_OBD_LSH_UP[i] = 0
LV_ERR_OBD_LSH_UP[i] = 0
STATE_OBD_LSH_UP[i] = 0
TEMP_DIF_OBD_LSH_UP[i] = 0
POW_INT_OBD_LSH_UP[i] = 0
TCC_VLD_OBD_LSH_UP[i] = 0
TCC_ERR_OBD_LSH_UP[i] = 0
CTR_CYCNR_R_IT_OBD_LS_UP[i] = 0
R_IT_L_MES_OBD_LSH_UP[i] = 0
R_IT_L_REF_OBD_LSH_UP[i] = 0
R_IT_H_MES_OBD_LSH_UP[i] = 0
R_IT_H_REF_OBD_LSH_UP[i] = 0

The following variable shall be initialised at reset, at every new driving cycle (i.e. LV_DC = 0 -> 1) and upon clearing error memory:

LV_END_DIAG_OBD_LSH_UP[i] = 0

Reset *TIMER_1*

R_IT_OBD_LSH_UP[i], R_IT_THD_OBD_LSH_UP[i] shall be initialised with the last value from the previous driving cycle stored in non-volatile memory after RESET.

NOTE: TIMER_1 refers to a program specific timer and is not intended to represent a variable name!

Recurrence:

The OBDII oxygen sensor heater diagnosis shall be carried out once per driving cycle. The diagnosis shall require a number of monitoring cycles to be carried out to permit statistical evaluation of the results. These monitoring cycles shall occur every 1 s, until a calibrateable number of valid diagnosis cycles has been exceeded.

Once this threshold has been completed, the OBDII oxygen sensor heater diagnosis shall no longer be carried out until a new driving cycle is initiated.

Should an OBDII heater fault be detected, the application assistances (Functions that are defined to be affected by the state of the oxygen sensor signal) shall remain active for the remainder of the driving cycle. The restarting of the affected functions during the next driving cycle shall be carried out according to the project philosophy.

Activation:

If (LV_ST_END = 1) **&**
(LV_INH_DIAG_OBD_LSH_UP[i] = 0)

then If (LV_LS_UP_READY[i] = 1) **&**
(STATE_LSH_UP[i] = LSH_POW_CTL) **&**
(LV_VB_CDN_OBD_2 = 1) **&**
(C_LSHPWM_MIN_LSH_UP ≤ LSHPWM_UP[i]) **&**
(LSHPWM_UP[i] ≤ C_LSHPWM_MAX_LSH_UP)

then Increment *TIMER_1*
If (*TIMER_1* ≥ C_T_DLY_POW_CTL_OBD_LSH_UP)

then If LV_ERR_SCP_LS_UP[i] = 0 **&**
LV_ERR_SCG_LS_UP[i] = 0 **&**
LV_ERR_OC_LS_UP[i] = 0 **&**
(LV_ERR_LSH_UP[i] = 0)

then If (TEG_DYN_LS_UP[i] ≤
C_TEG_DYN_MAX_OBD_LSH_UP)

then ***Calculate*** POW_INT_OBD_LSH_UP[i]
If (POW_INT_OBD_LSH_UP[i] ≥
C_POW_INT_MIN_OBD_LSH_UP)

then ***One-off update*** CTR_CYCNR_R_IT_OBD_LS_UP[i]
% see note below

If (CTR_CYCNR_R_IT_LS_UP_VLD[i] <>
CTR_CYCNR_R_IT_OBD_LS_UP[i])

then LV_CDN_DIAG_OBD_LSH_UP[i] = 1
CTR_CYCNR_R_IT_OBD_LS_UP[i] =
CTR_CYCNR_R_IT_LS_UP_VLD[i]
„Diagnosis detection active“

else STATE_OBD_LSH_UP[i] = POW_INT
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = TEG_THD
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = OBD1_CHK
Reset POW_INT_OBD_LSH_UP[i]
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = OBD_1_DLY
Reset *TIMER_1*
Reset POW_INT_OBD_LSH_UP[i]
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = LS_READY
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = DIAG_INIT
Reset *TIMER_1*
Reset POW_INT_OBD_LSH_UP[i]
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif

else STATE_OBD_LSH_UP[i] = DIAG_OFF
Reset *TIMER_1*
Reset POW_INT_OBD_LSH_UP[i]
LV_CDN_DIAG_OBD_LSH_UP[i] = 0

endif.

Deactivation:

If (LV_END_DIAG_OBD_LSH_UP[i] = 1)

then „Diagnosis detection complete“

endif.

NOTES: The state variable STATE_OBD_LSH_UP[i] *shall only be validated at the completion of the current monitoring cycle, i.e. the state should not vary within the current 1 s monitoring cycle, but indicate at the end of the current cycle , the last state reached, dependant upon the branches.*

The variable CTR_CYCNR_R_IT_OBD_LS_UP[i] shall be updated with the current value of the input variable CTR_CYCNR_R_IT_LS_UP_VLD[i] on each occasion that the integral POW_INT_OBD_LSH_UP[i] exceeds the threshold for the first time. For further function calls where the integral condition is met, the one-off update shall not be carried out. This shall continue until the integral condition is no longer met.

Formula section:

Calculation of cooling energy integral:

Determine difference between typical sensor operating temperature and exhaust gas temperature:

TEMP_DIF_OBD_LSH_UP[i] = (C_TEMP_SP_OBD_LSH_UP - TEG_DYN_LS_UP[i])

(NOTE: Resolution of temperature difference is reduced from 0.0625 °C, resolution of the individual temperature variables, to 1 °C; The difference shall be limited to provide only positive values, i.e. ≥ 0)

The integral of heat flux (in Watts) provides a measure for the total cooling energy of the exhaust gas (in Joules):

$POW\_INT\_OBD\_LSH\_UP[i]_{(n)}$ =($POW\_INT\_OBD\_LSH\_UP[i]_{(n-1)}$ +
TEMP_DIF_OBD_LSH_UP[i] *
IP_FAC_MAF_OBD_LSH_UP)

(NOTE: Integral shall be limited to its maximum hex limit, i.e. FFFFH or 65535 digits, to prevent overflow.)

Detection: (Assumes "Diagnosis detection active") **See also NOTE D above!**

Increment TCC_VLD_OBD_LSH_UP[i]

If (R_IT_LS_UP[i] ≥ IP_R_IT_THD_OBD_LSH_UP)

then Symptom "Oxygen sensor heater fault, bank i" active

ERR_SYM_OBD_LSH_UP = "SYM_0"

R_IT_H_MES_OBD_LSH_UP[i] = R_IT_LS_UP[i]

R_IT_H_REF_OBD_LSH_UP[i] = IP_R_IT_THD_OBD_LSH_UP

Increment TCC_ERR_OBD_LSH_UP[i]

else Symptom "Oxygen sensor heater fault, bank i" passive

ERR_SYM_OBD_LSH_UP = "NO_SYM"

R_IT_L_MES_OBD_LSH_UP[i] = R_IT_LS_UP[i]

R_IT_L_REF_OBD_LSH_UP[i] = IP_R_IT_THD_OBD_LSH_UP

endif.

If TCC_VLD_OBD_LSH_UP[i] ≥ C_TCC_THD_OBD_LSH_UP

then STATE_OBD_LSH_UP[i] = DIAG_END

If (TCC_ERR_OBD_LSH_UP[i] ≥ C_TCC_THD_ERR_OBD_LSH_UP)

then Symptom "Oxygen sensor heater fault, bank i" active

LV_ERR_OBD_LSH_UP[i] = 1

ERR_SYM_OBD_LSH_UP = "SYM_0"

R_IT_OBD_LSH_UP[i] = R_IT_H_MES_OBD_LSH_UP[i]

R_IT_THD_OBD_LSH_UP[i] = R_IT_H_REF_OBD_LSH_UP[i]

Deliver the result to Error Management

else Symptom "Oxygen sensor heater fault, bank i" passive
LV_ERR_OBD_LSH_UP[i] = 0

ERR_SYM_OBD_LSH_UP = "NO_SYM"

R_IT_OBD_LSH_UP[i] = R_IT_L_MES_OBD_LSH_UP[i]

R_IT_THD_OBD_LSH_UP[i] = R_IT_L_REF_OBD_LSH_UP[i]

Deliver the result to Error Management

endif

LV_END_DIAG_OBD_LSH_UP[i] = 1

Deliver the result to Error Management

else STATE_OBD_LSH_UP[i] = DIAG_ACT

endif.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LSHPWM_MIN_LSH_UP	1	0...FFH	0...99.609375	3.891E-1	%
Minimum permitted heater effective voltage PWM for upstream LSH OBD2 diagnosis					
C_LSHPWM_MAX_LSH_UP	1	0...FFH	0...99.609375	3.891E-1	%
Maximum permitted heater effective voltage PWM for upstream LSH OBD2 diagnosis					
C_T_DLY_POW_CTL_OBD_LSH_UP	1	0...FFH	0...255	1	s
Minimum delay prior to checking status of OBD1 heater /signal faults					
C_TEG_DYN_MAX_OBD_LSH_UP	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Temperature threshold of TEG_DYN_LS_UP under which LSH OBD2 activated					
C_TEMP_SP_OBD_LSH_UP	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Setpoint temperature for upstream operation used to generate power integral					
C_POW_INT_MIN_OBD_LSH_UP	1	0...FFFFH	0...65535	1	J
Mimimum required cooling energy at upstream position prior checking Ri threshold					
IP_R_IT_THD_OBD_LSH_UP	4*4	0...FFFFH	0...65535	1	Ω
LDP_R_IT_LS_NEW_IP_LSH_UP	4	0...FFFFH	0...65535	1	Ω
LDP_TCC_VLD_OBD_LSH_UP	4	0...FFFFH	0...65535	1	-
Internal resistance threshold, when exceeded, heater is determined to be defective for current driving cycle; according to reference value of internal resistance					
IP_FAC_MAF_OBD_LSH_UP	1*6	0...FFH	0...15.9375	6.25E-2	J/K
LDP_MAF_KGH_3_EGCP	6	0...FFFFH	0...2047.96875	3.125E-2	kg/h
Factor reflecting forced convention influence of cooling area, gas velocity and heat transfer coefficient					
C_TCC_THD_OBD_LSH_UP	1	0...FFFFH	0...65535	1	-
Number of valid LSH OBD2 diagnosis cycles over which stzatistic evaluation to be carried out					
C_TCC_THD_ERR_OBD_LSH_UP	1	0...FFFFH	0...65535	1	-
Fault detection threshold, when exceeded, heater is determined to be defective during driving cycle					

See also NOTE D above!

B.19.1 Upstream oxygen sensor heater OBDII monitoring (Appl. inc.)

B.19.1.1 Inhibition of oxygen sensor upstream heater diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_OBD_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean inhibit flag for O2 sensor heater OBDII monitoring					
LV_INH_DIAG_RBM_OBD_LSH_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean inhibit flag for Rate Based Monitoring of O2 sensor heater OBDII diagnosis					

Input data:

LV_IGK	LV_ERR_MAF	LV_ERR_RATIO_CHK	NC_CBK_EX_NR
LV_ST_END	LV_ERR_SCG_LS_UP[NC_CBK_EX_NR]	LV_ERR_OC_LS_UP[NC_CBK_EX_NR]	LV_ERR_SCP_LS_UP[NC_CBK_EX_NR]
LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_CDN_VB_OBD2	TCO_ST	TCO_STOP
CONF_LAM	LV_ERR_MAP		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

This function shall evaluate the application conditions applicable to the oxygen sensor heater OBDII monitoring and provide the result in a single boolean flag. This flag shall represent the interface to the main diagnosis function. Additionally the inhibition bit for Rate Based Monitoring is evaluated.

Description:

The oxygen sensor heater OBD II monitoring shall be ihibited if one of the following listed errors was recognized:

- LV_ERR_MAF/MAP shall indicate a fault in the mass air flow measurement which may have an effect on the modelled exhaust gas temperature (TEG_DYN). Such a faulty value may cause the heating to be incorrectly performed leading to a secondary fault entry.
- LV_ERR_RATIO_CHK shall indicate a fault due to the load – throttle position sensor diagnosis. Such a fault may cause incorrect heating to be performed.
  - LV_ERR_SCG_LS_UP, LV_ERR_OC_LS_UP and LV_ERR_SCP_LS_UP shall indicate a fault of the oxygen upstream sensor probe. Such a fault may lead to wrong internal resistance values.

- LV_ERR_LSH_UP shall indicate a fault of the oxygen upstream sensor heater. Such a fault may cause insufficient heating and may lead to false error entries.
- If TCO_ST has not decreased sufficiently compared with TCO_STOP the exhaust system is still warm and a heater error cannot be detected, therefore the OBD2 diagnosis is inhibited.

Application conditions:

Initialisation:

The following variables shall be initialised at reset and at the ignition key transition on (i.e. LV_IGK = 0 -> 1):

LV_INH_DIAG_OBD_LSH_UP[i] = 0

LV_INH_DIAG_RBM_OBD_LSH_UP[i] = 0

Recurrence:

The function shall be carried out once every 1s.

Activation:

LV_ST_END = 1

Deactivation:

LV_ST_END = 0

Formula section:

If (LV_ERR_MAF = 0) **&**
(LV_ERR_MAP = 0) **&**
(LV_ERR_RATIO_CHK = 0) **&**
(LV_ERR_LSH_UP[i] = 0) **&**
(LV_ERR_SCG_LS_UP[i] = 0) **&**
(LV_ERR_OC_LS_UP[i] = 0) **&**
(LV_ERR_SCP_LS_UP[i] = 0) **&**
(CONF_LAM = 1)

then

LV_INH_DIAG_RBM_OBD_LSH_UP[i] = 0

else

LV_INH_DIAG_RBM_OBD_LSH_UP[i] = 1

endif.

If (LV_INH_DIAG_RBM_OBD_LSH_UP[i] = 0) **&**
(LV_CDN_VB_OBD2 = 1) **&**
TCO_STOP – TCO_ST > C_TCO_DIF_MIN_OBD_LSH_UP

then

LV_INH_DIAG_OBD_LSH_UP[i] = 0

else

LV_INH_DIAG_OBD_LSH_UP[i] = 1

endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Upstream O2 sensor heater OBDII monitoring	O2 sensor heater circuit malfunction, bank 1, sensor 1	SYM_0	*NO*
		SYM_1	
		SYM_2	
OBD_LSH_UP[1]		SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
LSHPWM_UP_1
R_IT_LS_UP_1_ENVD_H
R_IT_LS_UP_1_ENVD_L

Application assistances:

Recommendation: The Application Assistances shall be maintained for the remaining duration of the driving cycle.

- Evaporative emission system:
  Control - activation of Minimum Operation
  System diagnosis - Inhibit
- Lambda control (Only for exhaust gas temperatures lower than O2 sensor lowest operating temperature):
  Control - Return to open-loop mode (Bank selectively)
  NOTE: Closed loop control may be permitted if the signal voltage excursion is large enough
  Adaptation - Inhibit (Bank selectively)
  Downstream trim regulation - Inhibit (Bank selectively)
- Oxygen sensor / Lambda controller diagnosis:
  O2 sensor upstream voltage excursion diagnosis - Inhibit (Bank selectively)
  O2 sensor upstream response time diagnosis - Inhibit (Bank selectively)
  O2 sensor lambda control limit diagnosis - Inhibit (Bank selectively)
  O2 sensor upstream frequency diagnosis - Inhibit (Bank selectively)
  O2 sensor upstream signal voltage diagnosis (Bank selectively)
  O2 sensor upstream signal plausibility diagnosis (Bank selectively)
  O2 sensor downstream response time diagnosis (Bank selectively)
  O2 sensor downstream signal voltage diagnosis (Bank selectively)
  O2 sensor downstream PUC end diagnosis (Bank selectively)
- Idle speed adaptation - Inhibit
- Knock adaptation - Inhibit
- MAF / TPS-Plausibility diagnosis - Inhibit
- Fuel system diagnosis - Inhibit (indirectly via LV_LSCL)

- Catalyst efficiency diagnosis - Inhibit (Bank selectively) (indirectly via LV_LSCL)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_DIF_MIN_OBD_LSH_UP	1	0 ... FEH	-48 ... 142.5	0.75	°C
Minimum drop in TCO_ST compared to TCO_STOP for OBD_LSH_UP activation					

B.19.1.21.2. Interface for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_OBD_LSH_UP[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of OBD_LSH_UP monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_OBD_LSH_UP_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Upstream O2 sensor OBDII heater diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_OBD_LSH_UP[NC_CBK_EX_NR]	LV_INH_DIAG_RBM_OBD_LSH_UP[NC_CBK_EX_NR]
NC_CBK_EX_NR	CTR_CDN_RBM_OBD_LSH_UP_1	CTR_COMP_RBM_OBD_LSH_UP_1	LV_ERR_EL_MAP
LV_ERR_VCC_PVS_2	LV_ERR_MAP_PLAUS		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the OBD_LSH_UP[i] monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_OBD_LSH_UP[i] data.

Within STATE_RBM_OBD_LSH_UP[i], three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_OBD_LSH_UP[i] = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1 of STATE_RBM_OBD_LSH_UP[i] = 0

on failure memory reset :

bit 1 of STATE_RBM_OBD_LSH_UP[i]= 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_OBD_LSH_UP[i] = 1

The pending status of the following failures has to be checked only once:

Dependence	Error			
NC_CBK_EX_NR	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_SCG_LS_UP[i]	LV_ERR_SCP_LS_UP[i]	LV_ERR_OC_LS_UP[i]
	LV_ERR_MAF_SCG_OC	LV_ERR_MAF_SCP	LV_ERR_LSH_UP[i]	LV_ERR_VCC_PVS_2
	LV_ERR_MAP_PLAUS	LV_ERR_EL_MAP		

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_OBD_LSH_UP[i] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_OBD_LSH_UP[i] = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_OBD_LSH_UP_1 = 0
then RATE_RBM_OBD_LSH_UP_1 = FFFFH
else RATE_RBM_OBD_LSH_UP_1
= (CTR_COMP_RBM_OBD_LSH_UP_1 / CTR_CDN_RBM_OBD_LSH_UP_1)
endif

Every 1 s :

If bit 0 of STATE_RBM_OBD_LSH_UP[i] = 0

Then

If LV_END_DIAG_OBD_LSH_UP[i] = 1

Then

bit 0 of STATE_RBM_OBD_LSH_UP[i] = 1

Endif

Endif

If bit 1 of STATE_RBM_OBD_LSH_UP[i] = 0

Then

If LV_INH_DIAG_RBM_OBD_LSH_UP[i] = 1

Then

bit 1 of STATE_RBM_OBD_LSH_UP[i] = 1

Endif

Endif

B.20 OBD II diagnosis of the lambda trim control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Diagnosis condition of lambda trim control					
STATE_ERR_LAM_ADJ [NC_CBK_EX_NR]	V/O	0H 1H 2H	NO_ERROR UPSTREAM ERROR DOWNSTREAM ERROR	1	-
Determination of faulty sensor which caused the trim controller error					
ERR_SYM_LAM_ADJ [NC_CBK_EX_NR]	V	0H 1H 2H	NO_SYM SYM_0 SYM_1	1	-
Symptoms of the lambda trim control errors					
LV_ERR_LAM_ADJ [NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status diagnostic result of lambda trim control					
LV_END_DIAG_LAM_ADJ[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Diagnostic cycle for lambda trim control complete					
LV_END_DIAG_WIN_LAM_ADJ[NC_CBK_EX_NR]	O/V	0…1H	0…1	1	[-]
End of Diagnostic cycle for similar conditions window for lambda trim control					
T_DLY_LAM_END_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	s
Counter to set LV_END_DIAG_LAM_ADJ [NC_CBK_EX_NR]					
T_DLY_LAM_SUM_MAX_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	s
Total duration over C_T_LAM_DLY_MAX_DIAG					
T_DLY_LAM_SUM_MIN_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	s
Total duration over C_T_LAM_DLY_MIN_DIAG					
T_DLY_LAM_SUM_RST_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	s
Time counter to reset T_DLY_LAM_SUM_MIN_DIAG[NC_CBK_EX_NR] and T_DLY_LAM_SUM_MAX_DIAG[NC_CBK_EX_NR]					

Input data:

LV_INH_DIAG_DLY_LAM[NC_CBK_EX_NR]	LV_CP_RAMP_OPEN_ACT	LV_LAM_ADJ_I_ACT[NC_CBK_EX_NR]	LV_IGK
T_DLY_I_LAM_ADJ[NC_CBK_EX_NR]	LV_LAM_ADJ_PER_VLD[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_DC

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

To fulfill the legal requirements of the OBDII lambda sensor diagnosis, the trim of the lambda control by the lambda sensor signal downstream of the catalytic converter must be monitored in addition to the diagnosis of the lambda sensor upstream of the catalytic converter (by determining the switching times and the control frequency). In the case of a diagnostic error, a plausibility check of the trim control diagnosis can be performed to assign the faulty behavior to the control sensor or the monitoring sensor.
Faulty behavior of the lambda sensors upstream or downstream the catalytic converter exists if sensor limits are reached and in this way one of the emission values in the FTP test is exceeded by 1.5 fold.

Description:

If the I-share of the adaptation value T_DLY_I_LAM_ADJ[i] exceeds the diagnostic threshold C_T_DLY_LAM_MAX_DIAG for the first time the time counters T_DLY_LAM_SUM_RST_DIAG[i] and T_DLY_LAM_SUM_MAX_DIAG[i] are started. T_DLY_LAM_SUM_RST_DIAG[i] is incremented as long as the diagnosis conditions are fulfilled. T_DLY_LAM_SUM_MAX_DIAG[i] is incremented also as long as the diagnostic threshold is exceeded. If T_DLY_I_LAM_ADJ[i] falls again below the diagnostic threshold, then the T_DLY_LAM_SUM_MAX_DIAG[i] is stopped. A failure is detected as soon as T_DLY_LAM_SUM_MAX_DIAG[i] is above the threshold C_T_DLY_LAM_SUM_MAX_DIAG. Exceeds T_DLY_LAM_SUM_RST_DIAG[i] the calibrateable limit and no failure was detected (ERR_SYM_LAM_ADJ[i] = "NO_SYM) all time counters are reseted. Incrementation starts again, if the threshold for T_DLY_I_LAM_ADJ[i] is exceeded again. If the engine is stopped the time counters are reseted also.
For the minimum threshold T_DLY_LAM_SUM_MIN_DIAG[i] the procedure is analogous.

Is there a failure present ERR_SYM_LAM_ADJ[i] ≠ "NO_SYM as described above the validity of the controller period is checked. If no complete controller period of the I component has occurred after the trim control was active (LV_LAM_ADJ_PER_VLD[i]), the malfunction is assigned to the lambda sensor downstream of the catalytic converter otherwise to the upstream sensor.

Application conditions:

Initialisation: all variables (except LV_END_DIAG_xx) are initialised with 0 at every LV_IGK = 0 ->1 and reset
LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 , at reset and at FMY clear.

Recurrence: 1s

Activation:

If LV_INH_DIAG_DLY_LAM[i] = 0
and LV_CP_RAMP_OPEN_ACT = 0
and LV_LAM_ADJ_I_ACT[i] = 1
then LV_CDN_DIAG_LAM_ADJ[i] = 1
activate the function
else LV_CDN_DIAG_LAM_ADJ[i] = 0
deactivate the function
LV_END_DIAG_WIN_LAM_ADJ[i] = 0

endif

Formula section:

If(1) LV_CDN_DIAG_LAM_ADJ[i] = 1
and(1) T_DLY_LAM_SUM_RST_DIAG[i] < C_T_DLY_LAM_RST_SUM_DIAG
then(1) **If**(2) LV_END_DIAG_LAM_ADJ[i] = 0
and(2) T_DLY_LAM_END_DIAG[i] < C_T_DLY_LAM_END_DIAG
then(2) *increment* T_DLY_LAM_END_DIAG[i]
LV_END_DIAG_WIN_LAM_ADJ[i] = 0
else(2) LV_END_DIAG_LAM_ADJ[i] = 1
LV_END_DIAG_WIN_LAM_ADJ[i] = 1
endif(2)
If(3) T_DLY_I_LAM_ADJ[i] > C_T_DLY_LAM_MAX_DIAG
then(3) *increment* T_DLY_LAM_SUM_MAX_DIAG[i]
ERR_SYM_LAM_ADJ[i] = "SYM_1"
else(3) *freeze* T_DLY_LAM_SUM_MAX_DIAG[i]
endif(3)
If(4) T_DLY_I_LAM_ADJ[i] < C_T_DLY_LAM_MIN_DIAG
then(4) *increment* T_DLY_LAM_SUM_MIN_DIAG[i]
ERR_SYM_LAM_ADJ[i] = "SYM_0"
else(4) *freeze* T_DLY_LAM_SUM_MIN_DIAG[i]
endif(4)
If(5) T_DLY_LAM_SUM_RST_DIAG[i] > 0
or(5) ERR_SYM_LAM_ADJ[i] ≠ "NO_SYM"
then(5) *increment* T_DLY_LAM_SUM_RST_DIAG[i]
else(5) *freeze* T_DLY_LAM_SUM_RST_DIAG[i]
endif(5)
If(6) (T_DLY_LAM_SUM_MIN_DIAG[i] >
C_T_DLY_LAM_SUM_MIN_DIAG
or(6) T_DLY_LAM_SUM_MAX_DIAG[i] >
C_T_DLY_LAM_SUM_MAX_DIAG)
and(6) C_T_DLY_LAM_MAX_DIAG > C_T_DLY_LAM_MIN_DIAG
then(6) LV_ERR_LAM_ADJ[i] = 1
LV_END_DIAG_LAM_ADJ[i] = 1
LV_END_DIAG_WIN_LAM_ADJ[i] = 1
If(7) LV_LAM_ADJ_PER_VLD[i] = 1
then(7) STATE_ERR_LAM_ADJ[i] = "UPSTREAM ERROR"

else(7) STATE_ERR_LAM_ADJ[i] = "DOWNSTREAM ERROR"

endif(7)

endif(6)

else(1) T_DLY_LAM_SUM_MAX_DIAG[i] = 0

T_DLY_LAM_SUM_MIN_DIAG[i] = 0

T_DLY_LAM_SUM_RST_DIAG[i] = 0

ERR_SYM_LAM_ADJ[i] = "NO_SYM"

STATE_ERR_LAM_ADJ[i] = “NO ERROR”

endif(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DLY_LAM_END_DIAG	1	0...FFFFH	0...65535	1	s
Time limit to set LV_END_DIAG_LAM_ADJ[NC_CBK_EX_NR]					
C_T_DLY_LAM_MAX_DIAG	1	8000...7FFFH	640..639.980469	1.95E-2	ms
Maximum limit of T_DLY_I_LAM_ADJ[NC_CBK_EX_NR]					
C_T_DLY_LAM_MIN_DIAG	1	8000...7FFFH	640..639.980469	1.95E-2	ms
Minimum limit of T_DLY_I_LAM_ADJ[NC_CBK_EX_NR]					
C_T_DLY_LAM_SUM_MAX_DIAG	1	0...FFFFH	0...65535	1	s
Total duration over C_T_DLY_LAM_MAX_DIAG					
C_T_DLY_LAM_SUM_MIN_DIAG	1	0...FFFFH	0...65535	1	s
Total duration over C_T_DLY_LAM_MIN_DIAG					
C_T_DLY_LAM_RST_SUM_DIAG	1	0...FFFFH	0...65535	1	s
Time counter to reset T_DLY_LAM_SUM_MIN_DIAG[NC_CBK_EX_NR] and T_DLY_LAM_SUM_MAX_DIAG[NC_CBK_EX_NR]					

Configuration data:

Diagnosis	Symptom	Nr	ABC type
Dynamic fuel trim	*Fuel Trim Malfunction upstream (Bank 1/2)*	*0*	NO
	Fuel Trim Malfunction downstream (Bank 1/2)	*1*	
LAM_ADJ[NC_CBK_EX_NR]			

B.21 Application Incidences of "Dynamic fuel trim diagnosis"

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_DLY_LAM[NC _CBK_EX_NR]	V/O	0...1H	0...1	1	-
Inhibition condition for the dynamic fuel trim diagnosis					

Input data:

LV_DC	LV_MIS_STATE_B4	LV_ERR_SA_SAV	LV_ERR_TPS
LV_ERR_CRK	LV_ERR_TCO	LV_IGK	LV_MIS_STATE_B1
LV_ERR_LS_UP[NC _CBK_EX_NR]	LV_ERR_LSH_DOWN[NC _CBK_EX_NR]	LV_ERR_EL_LS_DOWN[NC _CBK_EX_NR]	LV_ERR_SWT_LS_DOWN [NC _CBK_EX_NR]
LV_ERR_PUE_LS_DOWN[NC _CBK_EX_NR]	LV_ERR_MEC_OPEN_CPS	LV_ERR_OBD_LSH_DOWN[NC _CBK_EX_NR]	LV_ERR_MAP
LV_ERR_RATIO_CHK	LV_ERR_MEC_IVVT_EX_i	LV_ERR_MAF	LV_ERR_CAM
LV_ERR_EL_CPS	LV_ERR_CAT_DIAG[NC _CBK_EX_NR]	LV_ERR_FSD[NC _CBK_EX_NR]	LV_ERR_IGC
LV_ERR_IV[NC_CYL_NR]	LV_ERR_MEC_IVVT_IN_i		LV_ERR_PUC_LS_DOWN
LV_ERR_FSD_LAM_LIM_i	LV_CDN_VB_OBD2	LV_ERR_CAM_IN[NC_NR _CAM_CBK]	LV_ERR_CAM_EX[NC_NR _CAM_CBK]

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC _CBK_EX_NR = 2) then
   i = 1, for exhaust cylinder bank 1
   i = 2, for exhaust cylinder bank 2
otherwise (NC _CBK_EX_NR = 1)
   i = 1, for single exhaust cylinder bank.

Application conditions:

Initialisation: LV_INH_DIAG_DLY_LAM[I] is initialised with 1 at ervery LV_IGK = 0 -> 1 and at reset

Recurrence: 1s

Activation:: at every engine state

Formula section:

If	LV_DC = 1	
and	LV_ERR_TPS = 0	(Throttle position sensor error)
and	LV_ERR_RATIO_CHK = 0	(Throttle position sensor plausibility error)
and	LV_ERR_CAT_DIAG[i] = 0	(Catalysator diagnosis error)
and	LV_ERR_SA_SAV = 0	(Secondary air valve mechanical error)
and	LV_ERR_EL_CPS = 0	(Canister purge solenoid error)
and	LV_ERR_MEC_OPEN_CPS = 0	(Functional check Canister purge solenoid)
and	LV_ERR_TCO = 0	(Coolant temperature sensor error)
and	LV_ERR_MAF = 0	(Mass air flow sensor error)
and	LV_ERR_MAP = 0	(Manifold pressure sensor error)
and	LV_ERR_CRK = 0	(Crankshaft sensor error)
and	LV_ERR_CAM = 0	(Camshaft sensor error)
and	LV_ERR_FSD[i] = 0	(Fuel system diagnosis error)
and	LV_ERR_LS_UP[i] = 0	(O2 sensor up output stage OBD1 error)
and	LV_ERR_EL_LS_DOWN[I] = 0	(O2 sensor down OBD1 output stage error)
and	LV_ERR_LSH_DOWN[i] = 0	(O2 sensor heater down OBD1 output stage error)
and	LV_ERR_SWT_LS_DOWN[I] = 0	(O2 sensor down OBD2 switching time error, in case of a SWT error in the last driving cycle, this condition is faded out)
and	LV_ERR_PUE_LS_DOWN[I] = 0	(O2 sensor down OBD2 PUC end error, in case of a PUE error in the last driving cycle, this condition is faded out)
and	LV_ERR_OBD_LSH_DOWN[i] = 0	(O2 sensor heater down OBD2 error)
and	LV_MIS_STATE_B1 = 0	(misfire carb B error)
and	LV_MIS_STATE_B4 = 0	(misfire carb B error)
and	LV_ERR_IGC = 0	(Ignition Coil error)
and	LV_ERR_IV[NC_CYL_NR] = 0	(Injection Valve error)
and	LV_ERR_MEC_IVVT_IN_i = 0	(crankshaft to camshaft mec. violation error)
and	LV_ERR_MEC_IVVT_EX_i = 0	(crankshaft to camshaft mec. violation error)
and	LV_ERR_CAM_IN_i = 0	(IVVT position deviation)
and	LV_ERR_CAM_EX_i = 0	(IVVT position deviation)
and	LV_CDN_VB_OBD2 = 1	(Battery Voltage conditions fulfilled)
and	LV_ERR_FSD_LAM_LIM_i = 0	(FSD error with influence on lamda control)
and	LV_ERR_PUC_LS_DOWN_i = 0	(O2 sensor down error during PUC)
then		
	LV_INH_DIAG_DLY_LAM[I] = 0	(Dynamic fuel trim diagnosis enabled)
else		
	LV_INH_DIAG_DLY_LAM[I] = 1	(Dynamic fuel trim diagnosis disabled)
endif		

Application assistances:

For binary systems only:

Due to the increasing P jump delay times of the trim control, the control frequency of the lambda controller will decrease simultaneously. In the case of a faulty behaviour of the trim control, it must be ensured that the control frequency diagnosis of the lambda sensor upstream of the catalytic converter does not detect any successive error. The control frequency is determined by the mixture change detection of the lambda sensor upstream of the catalytic converter, i.e. the delay times of the trim controls are not considered separately.

Configuration for diagostic symptoms:

List of Environmental Data to store in Failure Memory: EFF_CAT_DIAG_1
VLS_DOWN_1_ENVD_H
VLS_DOWN_1_ENVD_L
ENG_STATE

B.22 Diagnosis of monitor sensors

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAG_MPL_CDN_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Status of permission for diagnosis					
LV_DIAG_MPL_END_LS_DOWN[NC_CBK_EX_NR]	O	0...1H	0...1	1	[-]
Bit to indicate end of complete monitor sensor diagnosis					

Input data:

LV_DIAG_MPL_INH_LS_DOWN[NC_CBK_EX_NR]	LV_SAP	LV_SAV	TCO
LV_LS_DOWN_READY[NC_CBK_EX_NR]	VS	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	LV_IGK
LV_END_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	LV_END_DIAG_SWT_LS_DOWN[NC_CBK_EX_NR]	NC_CBK_EX_NR	STATE_LSH_DOWN[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

The monitor sensor diagnosis consists of three separate tests. These tests are carried out at the beginning or at the end of the pull fuel cut-off (PUC). In detail there are the tests monitoring of the rich-lean switching time, monitoring of the signal voltage during PUC and monitoring of the signal voltage motion after PUC. The purpose of these tests is the recognition of emission relevant sensor errors caused by changes in the sensor behaviour or by electrical errors like open circuits.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
  i = 1, for exhaust cylinder bank 1
  i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
  i = 1, for single exhaust cylinder bank.

Application conditions:

Initialisation:

If At LV_IGK 0->1 **or** reset **or** at clearing error memory

then reset of all variables and bits to 0

endif

Recurrence: 20 ms

Activation / Deactivation:

If LV_DIAG_MPL_INH_LS_DOWN[i] = 0
(No inhibition reason for diagnosis)

and LV_SAP = 0

and LV_SAV = 0
(No secondary air injection)

and TCO > C_TCO_MIN_DIAG_MPL_LS_DOWN

and LV_LS_DOWN_READY[i] = 1
(Operating readiness of the monitor sensor must be detected)

and C_VS_MIN_DIAG_MPL_LS_DOWN ≤ VS ≤ C_VS_MAX_DIAG_MPL_LS_DOWN
(Vehicle speed in range)

and TEMP_CAT_DYN_MDL[i] > C_TEMP_CAT_MIN_DIAG_MPL_LS_DOWN
(Catalytic converter at operating temperature)

and **STATE_LSH_DOWN[i] = "LSH_POW_CTL"**

then LV_DIAG_MPL_CDN_LS_DOWN[i] = 1

else LV_DIAG_MPL_CDN_LS_DOWN[i] = 0

endif

If LV_DIAG_MPL_CDN_LS_DOWN[i] = 1

and (LV_END_DIAG_PUE_LS_DOWN[i] = 0

or LV_END_DIAG_SWT_LS_DOWN[i] = 0)

then LV_DIAG_MPL_CDN_LS_DOWN[i] = 1

else LV_DIAG_MPL_CDN_LS_DOWN[i] = 0

endif

If LV_END_DIAG_SWT_LS_DOWN[i] = 1

and LV_END_DIAG_PUE_LS_DOWN[i] = 1

then LV_DIAG_MPL_END_LS_DOWN[i] = 1

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_MIN_DIAG_MPL_LS_DOWN	1	0...FEH	-48...142.5	0.75	[°C]
Minimum coolant temperature for diagnosis of the monitor sensors					
C_VS_MIN_DIAG_MPL_LS_DOWN	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed for diagnosis of the monitor sensors					
C_VS_MAX_DIAG_MPL_LS_DOWN	1	0...FFH	0...255	1	[km/h]
Maximum vehicle speed for diagnosis of the monitor sensors					
C_TEMP_CAT_MIN_DIAG_MPL_LS_DOWN	1	0...7FFFH	-273.15... 1.77479E+3	0.0625	[°C]
Catalyst temperature threshold for diagnosis of the monitor sensors					

B.22.1 Monitoring the rich-lean switching times in the trailing throttle fuel cut-off

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_SWT_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Status of permission for diagnosis of switching time check					
VLS_DOWN_TRAN_PUC[NC_CBK_EX_NR]	V/O	0...3FFH	0...4.99511	4.8828e-3	[V]
Monitor sensor voltage on activating the trailing throttle fuel cut-off, Bank i					
CTR_SWT_ACT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFH	0...5100	20	[ms]
Cycle counter for switching time determination, Bank i					
CTR_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFH	0...255	1	[-]
Number of valid switching times from rich to lean					
CTR_QUO_SUM_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFFFH	0...255.99609	3.9063e-3	[-]
Sum of weighted CTR_SWT_ACT_LS_DOWN[NC_CBK_EX_NR], Bank i					
CTR_CYCNR_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFH	0...1.99218	0.0078125	[-]
Average of weighted CTR_SWT_ACT_LS_DOWN[NC_CBK_EX_NR], Bank i					
CTR_SAVE_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O/S	0...FFH	0...1.99218	0.0078125	[-]
Mean value of the ratio between the monitor sensor switching time and the threshold value for scantool					
CTR_QUO_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...FFFFH	0...255.99609	3.9063e-3	[-]
weighted value of CTR_SWT_ACT_LS_DOWN[NC_CBK_EX_NR]					
MAF_KGH_MIN_PUC	V	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Minimum value of MAF_KGH while switching time calculation					
LV_END_DIAG_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
bit to indicate end of switching time diagnosis					
LV_ERR_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Error bit of the switching time diagnosis					
LV_SWT_DIAG_VLD_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Validity bit of the current switching time check					
ERR_SYM_SWT_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Determined error of the switching time diagnosis					
CTR_SWT_ACT_RBM_LS_DOWN[NC_CBK_EX_NR]	V	0...FFH	0...5100	20	[ms]
Cycle counter for RBM switching time determination, Bank i					
CTR_QUO_SWT_RBM_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...255.99609	3.9063e-3	[-]
weighted value of CTR_SWT_ACT_RBM_LS_DOWN[NC_CBK_EX_NR] for RBM purpose					

Input data:

MAF_KGH	VLS_DOWN[NC_CBK_EX_NR]	LV_PUC	LV_IGK
LV_DIAG_MPL_CDN_LS_DOWN[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_DC	

FUNCTION DESCRIPTION:

General information:

Changes in the dynamic behaviour of the monitor sensor can be detected by the recognition of the rich-lean switching time after fuel cut-off.

Description:

After detection of the PUC engine operating state, the rich-lean switching time of the lambda sensor is determined and weighted depending upon the mass air flow in the intake system. The functioning of the sensor is diagnosed by comparison of the mean value of the weighted switching time with a limit. The functioning is guaranteed in this test if the switching time is below the limit.
The switching time diagnosis shall be finished for the current driving cycle, if C_CTR_SWT_DIAG_LS_DOWN valid switching times are calculated.
The rich-lean switching time of the sensor is determined in defined ranges of the mass air flow. A prerequisite for performing the diagnosis is that the voltage of the monitor sensor at the time of activating the trailing throttle fuel cut-off VLS_DOWN_TRAN_PUC[i] is above the threshold C_VLS_PUC_MIN_DIAG_SWT_LS_DOWN. The voltage value of the sensor at the time of trailing throttle fuel cut-off VLS_DOWN_TRAN_PUC[i] is required for the calculation of the dynamic limits for switching point determination and must be stored for the diagnostic period.
If the sensor voltage drops below the value C_FAC_VLS_MAX_DIAG_SWT_LS_DOWN * VLS_DOWN_TRAN_PUC[i], the switching time determination starts and the cycle counter CTR_SWT_ACT_LS_DOWN[i] has to be started. The counting process is ended if the limit C_FAC_VLS_MIN_DIAG_SWT_LS_DOWN * VLS_DOWN_TRAN_PUC[i] is exceeded downwards. The value of the cycle counter CTR_SWT_ACT_LS_DOWN[i] is a measure of the rich-lean switching time of the monitor sensor.
If a rich peak which exceeds C_VLS_HYS_DIAG_SWT_LS_DOWN occurs during determination of the switching time, its determination is interrupted and the relevant switching time is no longer further processed. The determination of the sensor switching times must also be interrupted if PUC is left.
Simultaneously to incrementation of CTR_SWT_ACT_LS_DOWN[i] the counter CTR_SWT_ACT_RBM_LS_DOWN[i] shall be incremented until the corresponding ratio CTR_QUO_SWT_RBM_LS_DOWN[i] exceeds the threshold for a defective lambda probe C_CTR_SWT_LS_DOWN. After this condition has been fulfilled the valid cycle counter CTR_SWT_LS_DOWN[i] shall be incremented. This procedure ensures that within each diagnostic cycle and after completion of the required total number of diagnostic cycles a probe malfunction can be detected as is demanded by ARB.
A valid value of the switching times for the monitor sensor must be weighted depending upon the minimum value for the mass flow during the determination of the switching time of the lambda sensor MAF_KGH_MIN_PUC. The weighting factor is determined via the map IP_FAC_CYCNR_MAX_LS_DOWN_ACT.
If for the determined switching time of the monitor sensor doesn't exceed the limit the sensor is diagnosed as functioning in this part of the diagnosis. The corresponding value must be saved for the scantool SAE 1979 independly if there is an error or not.
After the valid determination of switching time, the end of switching time diagnosis is indicated.

Application conditions:

Initialisation:

If At LV_IGK 0->1 **or** reset **or** at clearing error memory

then reset of all variables and bits to 0, except LV_END_DIAG_xx and:

MAF_KGH_MIN_PUC = *"max phys. limit"*

CTR_SAVE_SWT_LS_DOWN[i] which shall be initialized with its saved Value. Reset only at clearing error memory.

endif

If At LV_DC = 0->1 **or** reset **or** at clearing error memory

then reset LV_END_DIAG_xx:

endif

Recurrence: 20 ms

Activation / Deactivation:

If LV_DIAG_MPL_CDN_LS_DOWN[i] =1

and LV_END_DIAG_SWT_LS_DOWN[i] = 0

then LV_CDN_DIAG_SWT_LS_DOWN[i] =1
(switching time check enabled)

else LV_CDN_DIAG_SWT_LS_DOWN[i] =0
(switching time check disabled)

VLS_DOWN_TRAN_PUC[i] = 0

LV_SWT_DIAG_VLD_LS_DOWN[i] = 0

CTR_SWT_ACT_LS_DOWN[i] = 0

CTR_SWT_ACT_RBM_LS_DOWN[i] = 0

MAF_KGH_MIN_PUC = *"max phys. limit"*

endif

Formula section:

If LV_PUC = 0 ⇒ 1

then VLS_DOWN_TRAN_PUC[i] = VLS_DOWN[i]

endif

If LV_PUC = 1

and VLS_DOWN_TRAN_PUC[i] > C_VLS_PUC_MIN_DIAG_SWT_LS_DOWN

and C_MAF_PUC_MIN_DIAG_SWT_LS_DOWN < MAF_KGH < C_MAF_PUC_MAX_DIAG_SWT_LS_DOWN

and $VLS\_DOWN[i]_n < VLS\_DOWN[i]_{n-1}$ + C_VLS_HYS_DIAG_SWT_LS_DOWN

then **If** VLS_DOWN[i]<C_FAC_VLS_MAX_DIAG_SWT_LS_DOWN *

VLS_DOWN_TRAN_PUC[i]

then *increment* CTR_SWT_ACT_RBM_LS_DOWN[i]

If C_FAC_VLS_MIN_DIAG_SWT_LS_DOWN *
VLS_DOWN_TRAN_PUC[i] < VLS_DOWN[i]

then *increment* CTR_SWT_ACT_LS_DOWN[i]
MAF_KGH_MIN_PUC = MIN(MAF_KGH; MAF_KGH_MIN_PUC)

else *freeze* CTR_SWT_ACT_LS_DOWN[i]
$MAF\_KGH\_MIN\_PUC_{N}$ = $MAF\_KGH\_MIN\_PUC_{N-1}$
CTR_QUO_SWT_RBM_LS_DOWN[i] =
CTR_SWT_ACT_RBM_LS_DOWN[i] /
IP_FAC_CYCNR_MAX_LS_DOWN_ACT

If (CTR_QUO_SWT_RBM_LS_DOWN[i] >=
C_CTR_SWT_LS_DOWN)

and LV_SWT_DIAG_VLD_LS_DOWN[i] = 0

then LV_SWT_DIAG_VLD_LS_DOWN[i] = 1
increment CTR_SWT_LS_DOWN[i]
CTR_QUO_SWT_LS_DOWN[i] =
CTR_SWT_ACT_LS_DOWN[i] /
IP_FAC_CYCNR_MAX_LS_DOWN_ACT
CTR_QUO_SUM_SWT_LS_DOWN[i] =
CTR_QUO_SUM_SWT_LS_DOWN[i] +
CTR_QUO_SWT_LS_DOWN[i]
CTR_CYCNR_SWT_LS_DOWN[i] =
CTR_QUO_SUM_SWT_LS_DOWN[i] /
CTR_SWT_LS_DOWN[i]

endif

endif

endif

else VLS_DOWN_TRAN_PUC[i] = 0
LV_SWT_DIAG_VLD_LS_DOWN[i] = 0
CTR_SWT_ACT_LS_DOWN[i] = 0
CTR_SWT_ACT_RBM_LS_DOWN[i] = 0
MAF_KGH_MIN_PUC = *"max phys. limit"*

endif

If CTR_SWT_LS_DOWN[i] ≥ C_CTR_SWT_DIAG_LS_DOWN
then **If** CTR_CYCNR_SWT_LS_DOWN[i] < C_CTR_SWT_LS_DOWN
then LV_ERR_SWT_LS_DOWN[i] = 0
ERR_SYM_SWT_LS_DOWN[i] = "NO_SYM"
else LV_ERR_SWT_LS_DOWN[i] = 1
ERR_SYM_SWT_LS_DOWN[i] = " SYM _3"
endif
LV_END_DIAG_SWT_LS_DOWN[i] = 1
CTR_SAVE_SWT_LS_DOWN[i] = CTR_CYCNR_SWT_LS_DOWN[i]
else *switching time check going on*
endif

Filtering :

Apply filter on current symptoms
If filtering result available (after internal debounce; see Appl. Inc. ABC type: No)
then **Delivery** the result to Error Management
Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_PUC_MIN_DIAG_SWT_LS_DOWN	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Mass air flow threshold for diagnosis of the switching times					
C_MAF_PUC_MAX_DIAG_SWT_LS_DOWN	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Mass air flow threshold for diagnosis of the switching times					
C_VLS_PUC_MIN_DIAG_SWT_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
Voltage threshold for monitor sensor diagnosis					
C_FAC_VLS_MAX_DIAG_SWT_LS_DOWN	1	0...FFH	0...0.99609	3.91E-03	[-]
Reduction factor for start of switching time determination					
C_FAC_VLS_MIN_DIAG_SWT_LS_DOWN	1	0...FFH	0...0.99609	3.91E-03	[-]
Reduction factor for end of switching time determination					
C_VLS_HYS_DIAG_SWT_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
Monitor sensor voltage hysteresis					
C_CTR_SWT_DIAG_LS_DOWN	1	0...FFH	0...255	1	[-]
Condition for the start of the lambda sensor switching time check					
C_CTR_SWT_LS_DOWN	1	0...FFH	0...1.99218	0.0078125	[-]
Diagnostic threshold for CTR_CYCNR_SWT_LS_DOWN[NC_CBK_EX_NR]					
IP_FAC_CYCNR_MAX_LS_DOWN_ACT	8	0...FFH	0...5100	20	[ms]
LDP_MAF_KGH_IP_FAC_LS_DOWN	8	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Weighting factor for CTR_SWT_ACT_LS_DOWN[NC_CBK_EX_NR]					

B.22.2 Monitoring the sensor voltage after leaving trailing throttle fuel cut-off

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Status of permission for diagnosis of leaving PUC check					
VLS_DOWN_PUE[NC_CBK_EX_NR]	V/O	0...3FFH	0...4.99511	4.88E-03	[V]
Monitor sensor voltage at the time of leaving PUC, Bank i (i=1.2)					
LV_ERR_PUE_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Debounced diagnosis result of pull end plausibility					
LV_DIAG_PUE_VLD_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	[-]
Logical variable indicating validity of previous diagnostic cycle					
ERR_SYM_PUE_LS_DOWN[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error symptom of pull end plausibility					
LV_END_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Bit to indicate end of pull end plausibility diagnosis					
CTR_SYM_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	V	0...FFH	0...255	1	[-]
Counter for anti-bounce of PUE_LS_DOWN diagnosis					
CTR_END_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[-]
Counter to set LV_END_DIAG_VLS_DOWN_PUE_i					
VLS_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	V	0...3FFH	0...4.99511	4.88E-03	[V]
Maximum VLS_DOWN_i value from the beginning to the end of the MAF_INT_PUC_NOT_ACT calculation					
VLS_SAVE_DIAG_LS_DOWN[NC_CBK_EX_NR]	V/O/S	0...3FFH	0...4.99511	4.88E-03	[V]
Maximum VLS_DOWN_i value from the beginning to the end of the MAF_INT_PUC_NOT_ACT calculation for scantool					
LV_PURGE_VLD_PUE_LS_DOWN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Flag to indicate rich purge valid for PUE plausibility check of LS_DOWN					
VLS_UP_PCAT_PURGE_MIN[NC_CBK_EX_NR]	V	0...3FFH	0...4.99511	4.88E-03	-
Minimum VLS_UP during pre catalyst purge.					
LV_VLS_DOWN_AFL[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Monitoring sensor has detected "lean"					
LV_VLS_DOWN_AFR[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
Monitoring sensor has detected "rich"					
LV_MAF_INT_PUC_NOT_ACT_RST	V	0...1H	0...1	1	[-]
Logical variable indicating that MAF_INT_PUC_NOT_ACT integral has been reset					

Input data:

VLS_DOWN[NC_CBK_EX_NR]	LV_PUC	LV_DIAG_MPL_CDN_LS_DOWN[NC_CBK_EX_NR]	MAF_INT_PUC_NOT_ACT
MAF_INT_PUC_ACT	LV_IGK	NC_CBK_EX_NR	VLS_UP[NC_CBK_EX_NR]
LV_PCAT_PURGE_PUC_i	MAF_INT_PUE	LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_INH_INT_PUC_PUE_i
LV_DC			

FUNCTION DESCRIPTION:

General information:

With level monitoring of the monitor sensor after leaving PUC unplausible voltage values due to short circuit to ground can be detected and a plausibility check upstream / downstream sensor can be made.

Description:

If the general conditions for activating the diagnostic function are fulfilled and If MAF_INT_PUC_NOT_ACT exceeds the value C_MAF_INT_DIAG_PUE_LS_DOWN, then the diagnosis must be made.

If no unplausible lambda sensor voltage of the downstream sensor after PUC is detected ERR_SYM_PUE_LS_DOWN[i] = "NO_SYM" shall be set and the counter is decremented. After calibratable number of checks C_CTR_END_DIAG_PUE_LS_DOWN the end of diagnosis flag LV_END_DIAG_PUE_LS_DOWN[i] shall be set to 1 and VLS_DIF_SAVE_ DIAG_LS_DOWN[i] equal to VLS_DIF_MIN_DIAG_PUE_LS_DOWN[i].

Application conditions:

Initialisation:

If At LV_IGK 0->1 **or** reset **or** <u>at clearing error memory</u>

then reset of all variables and bits to 0 except LV_END_DIAG_xx and:

- VLS_SAVE_DIAG_LS_DOWN which shall be initialized with its saved value. Reset only at clearing error memory.
- LV_DIAG_PUE_VLD_LS_DOWN[i] = 1

endif

If At LV_DC = 0->1 **or** reset **or** <u>at clearing error memory</u>

then reset LV_END_DIAG_xx:

endif

Recurrence: 20 ms

Activation / Deactivation:

If LV_DIAG_MPL_CDN_LS_DOWN[i] =1

and LV_END_DIAG_PUE_LS_DOWN[i] = 0

then LV_CDN_DIAG_PUE_LS_DOWN[i] =1

(leaving PUC check enabled)

else LV_CDN_DIAG_PUE_LS_DOWN[i] =0

(leaving PUC check disabled)

LV_DIAG_PUE_VLD_LS_DOWN[i] = 1

endif

Formula section:

Check if MAF_INT_PUC_NOT_ACT has been reset:

if MAF_INT_PUC_NOT_ACT$_n$ < MAF_INT_PUC_NOT_ACT$_{n-1}$
then LV_MAF_INT_PUC_NOT_ACT_RST = 1

Check if VLS_DOWN[i] was active in the past (once it is set it is not reset during the DC)

if VLS_DOWN[i] < C_VLS_AFL_THD_DIAG_LS_DOWN
then LV_VLS_DOWN_AFL[i] = 1 *// Lean voltage reached*

if VLS_DOWN[i] > C_VLS_AFR_THD_DIAG_LS_DOWN
then LV_VLS_DOWN_AFR[i] = 1 *// Rich voltage reached*

Valid catalyst purge for PUE_LS_DOWN diagnosis:

```
If      LV_PUC = 0 ⇒ 1
then    LV_PURGE_VLD_PUE_LS_DOWN = 0
        VLS_UP_PCAT_PURGE_MIN[i] = 4.99511
endif

If      LV_ERR_LS_UP[i] = 0
  and   LV_PCAT_PURGE_PUC_i = 1
  and   MAF_INT_PUE > C_MAF_INT_PUE_PURGE_VLD
then    VLS_UP_PCAT_PURGE_MIN[i] = MIN (VLS_UP[i],
                                        VLS_UP_PCAT_PURGE_MIN[i])
endif
if      LV_PCAT_PURGE_PUC_i = 1 ⇒ 0
then    if      VLS_UP_PCAT_PURGE_MIN[i] > C_VLS_UP_MIN_PURGE_VLD
          and   LV_INH_INT_PUC_PUE_i = 0
        then    LV_PURGE_VLD_PUE_LS_DOWN = 1
        else    LV_PURGE_VLD_PUE_LS_DOWN = 0
        endif
endif
```

PUC End Diagnosis Algorithm:

```
If      LV_PUC = 1 ⇒ 0
then    VLS_DOWN_PUE[i] = VLS_DOWN[i]
        LV_DIAG_PUE_VLD_LS_DOWN[i] = 0
        VLS_DIAG_PUE_LS_DOWN[i] = 0V
endif
```

```
If      LV_PUC = 0
and     MAF_INT_PUC_ACT > C_MAF_INT_PUC_DIAG_PUE_LS_DOWN
and     LV_DIAG_PUE_VLD_LS_DOWN[i] = 0
and     VLS_DOWN_PUE < C_VLS_DOWN_PUE_DIAG_LS_DOWN
then    VLS_DIAG_PUE_LS_DOWN[i] = MAX(VLS_DOWN[i],
                                        VLS_DIAG_PUE_LS_DOWN[i])
        If      MAF_INT_PUC_NOT_ACT > C_MAF_INT_DIAG_PUE_LS_DOWN
        and     LV_MAF_INT_PUC_NOT_ACT_RST = 1
        and     LV_PURGE_VLD_PUE_LS_DOWN = 1
        then    LV_MAF_INT_PUC_NOT_ACT_RST = 0
                CTR_END_DIAG_PUE_LS_DOWN[i] =
                        CTR_END_DIAG_PUE_LS_DOWN[i] + 1
                LV_DIAG_PUE_VLD_LS_DOWN[i] = 1
                if      VLS_DIAG_PUE_LS_DOWN[i] >
                                C_VLS_DOWN_MIN_DIAG_PUE_LS_DOWN
                then    CTR_SYM_DIAG_PUE_LS_DOWN[i] =
                                CTR_SYM_DIAG_PUE_LS_DOWN[i] – 1
                        ERR_SYM_PUE_LS_DOWN = "NO_SYM"
                else    CTR_SYM_DIAG_PUE_LS_DOWN[i] =
                                CTR_SYM_DIAG_PUE_LS_DOWN[i]
                                + C_CTR_SYM_INC_DIAG_PUE_LS_DOWN
                        ERR_SYM_PUE_LS_DOWN = "SYM_3"
                endif
        endif
endif

If      CTR_END_DIAG_PUE_LS_DOWN[i] ≥ C_CTR_END_DIAG_PUE_LS_DOWN
then    LV_END_DIAG_PUE_LS_DOWN[i] = 1
        if      CTR_SYM_DIAG_PUE_LS_DOWN[i] ≥
                                C_CTR_SYM_MAX_DIAG_PUE_LS_DOWN
        and     LV_VLS_DOWN_AFR[i] = 0
        then    LV_ERR_PUE_LS_DOWN[i] = 1
                VLS_SAVE_DIAG_LS_DOWN[i] = VLS_DIAG_PUE_LS_DOWN[i]
        else    LV_ERR_PUE_LS_DOWN[i] = 0
                VLS_ SAVE_DIAG_LS_DOWN[i] = VLS_ DIAG_PUE_LS_DOWN[i]
        endif
endif
```

Filtering:

Apply filter on current symptoms

If filtering result available (after internal debounce; see Appl. Inc. ABC type: No)

then Deliver the result to Error Management

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_INT_PUC_DIAG_PUE_LS_DOWN	1	0...FFFFH	0...2912.66666	4.44E-02	[g]
Threshold of MAF integral during PUC for activation of the diagnosis					
C_MAF_INT_DIAG_PUE_LS_DOWN	1	0...FFFFH	0...2912.66666	4.44E-02	[g]
Threshold of MAF integral outside PUC for start of the diagnosis					
C_CTR_END_DIAG_PUE_LS_DOWN	1	0...FFFFH	0...65535	1	[-]
Counter threshold for CTR_END_DIAG_PUE_LS_DOWN_[NC_CBK_EX_NR]					
C_CTR_SYM_INC_DIAG_PUE_LS_DOWN	1	0...FFH	0...255	1	[-]
Counter increment for PUE_VLS_DOWN diagnosis					
C_CTR_SYM_MAX_DIAG_PUE_LS_DOWN	1	0...FFH	0...255	1	[-]
Maximum value of counter for PUE_VLS_DOWN diagnosis					
C_VLS_DOWN_MIN_DIAG_PUE_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
Threshold for VLS_DOWN for implausible signal recognition.					
C_VLS_DOWN_PUE_DIAG_LS_DOWN	1	0...3FFH	0...4.99511	4.88E-03	[V]
Threshold for valid VLS_DOWN signal at PUC end.					
C_VLS_UP_MIN_PURGE_VLD	1	0...3FFH	0...4.99512	0.0048828	V
Minimum VLS_UP for valid (rich) pre catalyst purge recognition.					
C_MAF_INT_PUE_PURGE_VLD	1	0...FFFFH	0...1820.4167	0.0278	g
Minimum MAF_INT_PUE for VLS_UP validity check during pre catalyst purge.					
C_VLS_AFL_THD_DIAG_LS_DOWN	1	0...3FFH	0...4.995117	4.8828E-3	V
Lambda sensor voltage "lean" detected					
C_VLS_AFR_THD_DIAG_LS_DOWN	1	0...3FFH	0...4.995117	4.8828E-3	V
Lambda sensor voltage "rich" detected					

B.23 Activation conditions for diagnosis of the monitor sensors

1.1. Calculation of inhibition for diagnosis and for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAG_MPL_INH_LS_DOWN[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
OBD II function dowmstream sensor: application incidences for inhibition					
LV_INH_DIAG_RBM_MPL_LS_DOWN[NC_CBK_EX_NR]	O/V	0...1H	0...1	1	-
OBD II function dowmstream sensor: application incidences for inhibition of Rate Based Monitoring					

Input data:

LV_ERR_CAM	LV_ERR_CRK	LV_ERR_TCO	LV_ERR_TPS
LV_ERR_VS	LV_ERR_RATIO_CHK	LV_ERR_FSD[NC_CBK_EX_NR]	LV_IGK
LV_ERR_MEC_OPEN_CPS	LV_ERR_EL_CPS	LV_ERR_PUE_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_SWT_LS_DOWN[NC_CBK_EX_NR]
LV_ERR_LS_UP[NC_CBK_EX_NR]	LV_ERR_LSH_DOWN[NC_CBK_EX_NR	LV_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]	LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]
LV_MIS_STATE_B	LV_ST_END	STATE_ERR_IV	LV_SAV
LV_ERR_MAF	LV_LS_DOWN_OBD_2_MAN_DEAC[NC_CBK_EX_NR]	LV_SAP	NC_CBK_EX_NR
LV_ERR_IV[NC_CYL_NR]	LV_ERR_MWSS	LV_CDN_VB_OBD2	LV_ERR_MAP
LV_MIS_STATE_A	CONF_LAM	CONF_CAT_EFF	LV_ERR_CAM_IN[NC_NR_CAM_CBK]
LV_ERR_CAM_EX[NC_NR_CAM_CBK]	LV_ERR_MEC_IVVT_IN_i	LV_ERR_MEC_IVVT_EX_i	

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
    i = 1, for exhaust cylinder bank 1
    i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
    i = 1, for single exhaust cylinder bank.

If one of the below mentioned conditions is not fulfilled, the inhibit bit interrupts the downstream oxygen sensor diagnosis. The inhibition bit for Rate Based Monitoring is set if one of the listed errors below is recognized.

The logical variable LV_LS_DOWN_OBD_2_MAN_DEAC[i] is set via LC_LS_DOWN_OBD_2_MAN_DEAC[i] which provides the opportunity of manual deactivation of the OBDII diagnostic package for downstream sensors.

Application conditions:

Initialization: *at every LV_IGK = 0-> 1 and reset all variables are initialized with 0*

Recurrence: *100 ms*

Activation: *after leaving engine start LV_ST_END = 1*

Formula section:

Inhibition of Rate Based Monitoring

If	LV_ERR_TPS = 0	(Throttle position sensor error)
and	LV_ERR_RATIO_CHK = 0	(Throttle position sensor plausibility error)
and	LV_ERR_EL_CPS = 0	(Canister purge solenoid error)
and	LV_ERR_MEC_OPEN_CPS = 0	(Canister purge solenoid mechanical error)
and	LV_ERR_TCO = 0	(Coolant temperature sensor error)
and	LV_ERR_MAF = 0	(MAF sensor error)
and	LV_ERR_MAP = 0	(Manifold air pressure sensor**)**
and	LV_ERR_CRK = 0	(Crankshaft sensor error)
and	LV_ERR_CAM = 0	(Camshaft sensor error)
and	LV_ERR_CAM_IN = 0	(Camshaft sensor error)
and	LV_ERR_CAM_EX = 0	(Camshaft sensor error)
and	LV_ERR_FSD[i] = 0	Fuel system diagnosis error)
and	LV_ERR_VS = 0	(Vehicle speed sensor error)
and	LV_ERR_LS_UP[i] = 0	(O2 sensor up output stage OBD1 error)
and	LV_ERR_EL_LS_DOWN[i] = 0	(O2 sensor down OBD1 output stage error)
and	LV_ERR_LSH_DOWN[i] = 0	(O2 sensor down heater OBD1 error)
and	LV_ERR_SWT_LS_DOWN[i] = 0	(O2 sensor down OBD2 switching time error)
and	LV_ERR_PUE_LS_DOWN[i] = 0	(O2 sensor down OBD2 PUC end error)
and	LV_ERR_OBD_LSH_DOWN[i] = 0	(O2 sensor heater down OBD2 error)
and	LV_ERR_IV[NC_CYL_NR] = 0	(Injection valve error)
and	LV_ERR_MWSS = 0	(Magnetic wheelspeed sensor error)
and	LV_ERR_MEC_IVVT_IN = 0	(Camshaft shifter error)
and	LV_ERR_MEC_IVVT_EX = 0	(Camshaft shifter error)
and	LV_CDN_VB_OBD2 = 1	(Battery voltage conditions fulfilled)
and	LV_MIS_STATE_A = 0	(CARB A Misfire error)
and	LV_MIS_STATE_B = 0	(CARB B1 or B4 Misfire error)
and	CONF_LAM = 1	(closed loop lambda control configuration)
then		
	LV_INH_DIAG_RBM_MPL_LS_DOWN[i] = 0	
else		
	LV_INH_DIAG_RBM_MPL_LS_DOWN[i] = 1	
endif		

Inhibition of diagnosis

If STATE_ERR_IV = 0 (Injection valve error)
and LV_SAV = 0
and LV_SAP = 0
and LV_INH_DIAG_RBM_MPL_LS_DOWN[i] = 0
and LV_LS_DOWN_OBD_2_MAN_DEAC[i] = 0
and CONF_CAT_EFF > 0
then
LV_DIAG_MPL_INH_LS_DOWN [i] = 0
else
LV_DIAG_MPL_INH_LS_DOWN [i] = 1
endif

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Oxygen sensor downstream rich lean switch time check		*SYM_0*	*NO*
		SYM_1	
		SYM_2	
SWT_LS_DOWN_1	Rich/lean switch time error	*SYM_3*	

List of Environmental Data to store in Failure Memory: ENG_STATE
CTR_CYCNR_SWT_LS_DOWN
LSHPWM_DOWN_1
VLS_DOWN_MMV_MIN_1_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Oxygen sensor downstream monitoring sensor voltage after PUC check		*SYM_0*	*STD*
		SYM_1	
		SYM_2	
PUE_LS_DOWN_1	Invalid PUC check after signal	*SYM_3*	

List of Environmental Data to store in Failure Memory: LV_VLS_DOWN_AFR
LV_VLS_DOWN_AFL
VLS_DOWN_1_ENVD_H
VLS_DOWN_1_ENVD_L

1.2. Interface for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_SWT_LS_DOWN[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of SWT_LS_DOWN monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
STATE_RBM_PUE_LS_DOWN[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of PUE_LS_DOWN monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_SWT_LS_DOWN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Downstream O2 sensor Switching Time Diagnosis , for application purpose					
RATE_RBM_PUE_LS_DOWN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Downstream O2 sensor PUE Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_INH_DIAG_RBM_MPL_LS_DOWN[NC_CBK_EX_NR]	NC_CBK_EX_NR
LV_END_DIAG_SWT_LS_DOWN[NC_CBK_EX_NR]	LV_END_DIAG_PUE_LS_DOWN[NC_CBK_EX_NR]	NC_NR_CAM_CBK	NLC_CAM_IN
CTR_CDN_RBM_SWT_LS_DOWN_1	CTR_COMP_RBM_SWT_LS_DOWN_1	CTR_CDN_RBM_PUE_LS_DOWN_1	CTR_COMP_RBM_PUE_LS_DOWN_1
LV_ERR_EL_MAP	LV_ERR_VCC_PVS_2	LV_ERR_MAP_PLAUS	

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the SWT_LS_DOWN[i] and PUE_LS_DOWN[i] monitors and the Rate-Based Monitoring statistics is defined with STATE_RBM_SWT_LS_DOWN[i] and STATE_RBM_PUE_LS_DOWN[i] data, respectively.

Within STATE_RBM_SWT_LS_DOWN[i] and STATE_RBM_PUE_LS_DOWN[i], three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_SWT_LS_DOWN[i] = 0

bit 0, bit 1 and bit 2 of STATE_RBM_PUE_LS_DOWN[i] = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 0

bit 0 and bit 1 of STATE_RBM_PUE_LS_DOWN[i] = 0

on failure memory reset :

bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 0

bit 1 of STATE_RBM_PUE_LS_DOWN[i] = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_SWT_LS_DOWN[i] = 1

bit 2 of STATE_RBM_PUE_LS_DOWN[i] = 1

The pending status of the following failures has to be checked only once:

For ETC and non_ETC variants:

Dependence	Error			
	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_VCC_PVS_2		
	LV_ERR_MAF_SCG_OC	LV_ERR_MAF_SCP	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS
	LV_ERR_TCO_EL	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_ERR_TCO_PLAUS
	LV_ERR_CRK_SYN	LV_ERR_CRK_TOOTH_PER	LV_ERR_CRK_TOOTH	LV_ERR_CRK_PLAUS
	LV_ERR_CPS	LV_ERR_MEC_CPS		
	LV_ERR_VS			
NC_NR_CAM_CBK	LV_ERR_SYN_CAM_IN_i	LV_ERR_PER_CAM_IN_i	LV_ERR_PLAUS_CAM_IN_i	LV_ERR_REF_CRK_CAM_IN_i
	LV_ERR_SYN_CRK_CAM_IN_i			
NC_CBK_EX_NR	LV_ERR_SCG_LS_UP[i]	LV_ERR_SCP_LS_UP[i]	LV_ERR_OC_LS_UP[i]	LV_ERR_PUC_LS_UP[i]
	LV_ERR_STK_LS_UP[i]	LV_ERR_LSH_UP[i]	LV_ERR_OBD_LSH_UP[i]	LV_ERR_SWT_LS_UP[i]
	LV_ERR_FRQ_LS_UP[i]			
	LV_ERR_OBD_LSH_DOWN[i]	LV_ERR_SWT_LS_DOWN[i]	LV_ERR_PUE_LS_DOWN[i]	LV_ERR_LSH_DOWN[i]
	LV_ERR_SCG_LS_DOWN[i]	LV_ERR_SCP_LS_DOWN[i]	LV_ERR_OC_LS_DOWN[i]	
	LV_ERR_FSD[i]	LV_ERR_MEC_IVVT_IN[i]	LV_ERR_MEC_IVVT_EX[i]	

Additional diagnosis to be checked only for ETC variant :

LV_ERR_TPS_1	LV_ERR_TPS_2	LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2
LV_ERR_MTC_DR	LV_ERR_MTC_CTL		

Additional diagnosis to be checked only for non-ETC variant :

LV_ERR_TPS_EL	LV_ERR_VCC_TPS		

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

The following failures must be checked in dependence on #NC_NR_CAM_CBK:

#IF (NC_NR_CAM_CBK = 1)
#THEN
#IF (NLC_CAM_IN = 1)
#THEN
LV_ERR_SYN_CAM_IN_1 = 1 **OR**
Present intake camshaft sensor failure

LV_ERR_PER_CAM_IN_1 = 1 **OR**
Present failure intake camshaft period too short

LV_ERR_PLAUS_CAM_IN_1 = 1 **OR**
Present intake camshaft failure synchronization impossiple

LV_ERR_REF_CRK_CAM_IN_1 = 1 **OR**
Present intake camshaft failure reference violation

LV_ERR_SYN_CRK_CAM_IN_1 = 1
Present intake camshaft failure for crankshaft synchronization

#ENDIF

#ELSE

#IF (NLC_CAM_IN = 1)
#THEN

LV_ERR_SYN_CAM_IN_1 = 1 **OR**
Present intake camshaft sensor failure

LV_ERR_PER_CAM_IN_1 = 1 **OR**
Present failure intake camshaft period too short

LV_ERR_PLAUS_CAM_IN_1 = 1 **OR**
Present intake camshaft failure synchronization impossiple

LV_ERR_REF_CRK_CAM_IN_1 = 1 **OR**
Present intake camshaft failure reference violation

LV_ERR_SYN_CRK_CAM_IN_1 = 1
Present intake camshaft failure for crankshaft synchronization

AND

LV_ERR_SYN_CAM_IN_2 = 1 **OR**
Present intake camshaft sensor failure

LV_ERR_PER_CAM_IN_2 = 1 **OR**
Present failure intake camshaft period too short

LV_ERR_PLAUS_CAM_IN_2 = 1 **OR**
Present intake camshaft failure synchronization impossiple

LV_ERR_REF_CRK_CAM_IN_2 = 1 **OR**
Present intake camshaft failure reference violation

LV_ERR_SYN_CRK_CAM_IN_2 = 1
Present intake camshaft failure for crankshaft synchronization

#ENDIF

#ENDIF

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 1

Endif(2)

Endwhile

bit 1 of STATE_RBM_PUE_LS_DOWN[i] =

bit 1 of STATE_RBM_SWT_LS_DOWN[i]

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_SWT_LS_DOWN[i] = 0
and LV_END_DIAG_SWT_LS_DOWN[i] = 1
Then bit 0 of STATE_RBM_SWT_LS_DOWN[i] = 1
Endif

If bit 0 of STATE_RBM_PUE_LS_DOWN[i] = 0
and LV_END_DIAG_PUE_LS_DOWN[i] = 1
Then bit 0 of STATE_RBM_PUE_LS_DOWN[i] = 1
Endif

If bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 0
and LV_ERR_TPS = 1
or LV_ERR_RATIO_CHK = 1
or LV_ERR_EL_CPS = 1
or LV_ERR_MEC_OPEN_CPS = 1
or LV_ERR_TCO = 1
or LV_ERR_MAF = 1
or LV_ERR_MAP = 1
or LV_ERR_CRK = 1
or LV_ERR_CAM = 1
or LV_ERR_CAM_IN = 1
or LV_ERR_CAM_EX = 1
or LV_ERR_FSD[i] = 1
or LV_ERR_VS = 1
or LV_ERR_LS_UP[i] = 1
or LV_ERR_EL_LS_DOWN[i] = 1
or LV_ERR_LSH_DOWN[i] = 1
or LV_ERR_SWT_LS_DOWN[i] = 1
or LV_ERR_PUE_LS_DOWN[i] = 1
or LV_ERR_OBD_LSH_DOWN[i] = 1
or LV_ERR_IV[NC_CYL_NR] = 1
or LV_ERR_MWSS = 1
or LV_ERR_MEC_IVVT_IN = 1
or LV_ERR_MEC_IVVT_EX = 1
or LV_ERR_VB = 1
or LV_MIS_STATE_A = 1
or LV_MIS_STATE_B = 1
Then bit 1 of STATE_RBM_SWT_LS_DOWN[i] = 1
Endif

If bit 1 of STATE_RBM_PUE_LS_DOWN[i] = 0

and LV_ERR_TPS = 1
or LV_ERR_RATIO_CHK = 1
or LV_ERR_EL_CPS = 1
or LV_ERR_MEC_OPEN_CPS = 1
or LV_ERR_TCO = 1
or LV_ERR_MAF = 1
or LV_ERR_MAP = 1
or LV_ERR_CRK = 1
or LV_ERR_CAM = 1
or LV_ERR_CAM_IN = 1
or LV_ERR_CAM_EX = 1
or LV_ERR_FSD[i] = 1
or LV_ERR_VS = 1
or LV_ERR_LS_UP[i] = 1
or LV_ERR_EL_LS_DOWN[i] = 1
or LV_ERR_LSH_DOWN[i] = 1
or LV_ERR_SWT_LS_DOWN[i] = 1
or LV_ERR_PUE_LS_DOWN[i] = 1
or LV_ERR_OBD_LSH_DOWN[i] = 1
or LV_ERR_IV[NC_CYL_NR] = 1
or LV_ERR_MEC_IVVT_IN = 1
or LV_ERR_MEC_IVVT_EX = 1
or LV_ERR_MWSS = 1
or LV_ERR_VB = 1
or LV_MIS_STATE_A = 1
or LV_MIS_STATE_B = 1

Then bit 1 of STATE_RBM_PUE_LS_DOWN[i] = 1

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_SWT_LS_DOWN_1 = 0
then RATE_RBM_SWT_LS_DOWN_1 = FFFFH
else RATE_RBM_SWT_LS_DOWN_1
= (CTR_COMP_RBM_SWT_LS_DOWN_1 / CTR_CDN_RBM_SWT_LS_DOWN_1)
endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_PUE_LS_DOWN_1 = 0
then RATE_RBM_PUE_LS_DOWN_1 = FFFFH
else RATE_RBM_PUE_LS_DOWN_1
= (CTR_COMP_RBM_PUE_LS_DOWN_1 / CTR_CDN_RBM_PUE_LS_DOWN_1)
endif

B.24 Downstream oxygen sensor heater OBDII monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean flag indicating that conditions for diagnosis met					
ERR_SYM_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Variable indicating status of each symptom, updated each recurrence & at diagnosis completion					
LV_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean flag indicating that fault is present and has been debounced					
LV_END_DIAG_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	[-]
Boolean flag indicating that diagnosis has been completed					
STATE_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0H 1H 2H 3H 4H 5H 6H 7H 8H	DIAG_OFF DIAG_INIT LS_READY OBD1_DLY OBD1_CHK TEG_THD POW_INT DIAG_ACT DIAG_END	1	[-]
Indicates current phase of oxygen sensor heater OBDII diagnosis					
TEMP_DIF_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...7FFH	0...2047	1	[K]
Indicates difference in temperature between set operating temperature & exhaust gas at sensor location					
POW_INT_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[J]
Integral indicates measure of cooling energy of exhaust gas at sensor location					
TCC_VLD_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...3FFH	0...1023	1	[-]
Number of completed valid diagnosis cycles					
TCC_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...3FFH	0...1023	1	[-]
Number of faults detected during valid diagnosis cycles					
CTR_CYCNR_R_IT_OBD_LS_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[-]
Value of handshake counter for last valid resistance value used for diagnosis					
R_IT_THD_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...65535	1	[Ohm]
Threshold for most current downstream oxygen sensor internal resistance used for diagnosis cycle, Mode 6 information					
R_IT_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...65535	1	[Ohm]
Most current downstream oxygen sensor internal resistance used for diagnosis cycle, Mode 6 information					
T_DLY_OBD_LSH_DOWN_TMP	-	0...FFH	0...255	1	[s]
Minimum delay time prior to checking status of OBDI heater / signal faults					
POW_INT_MIN_OBD_LSH_DOWN_TMP	-	0...FFFFH	0...65535	1	[J]
Minimum required cooling energy required at downstream sensor prior to checking Ri threshold					
TCC_THD_OBD_LSH_DOWN_TMP	-	0...3FFH	0...1023	1	[-]
Number of valid downstream LSH OBDII diagnosis cycles after which statistic evaluation is carried out					
TCC_THD_ERR_OBD_LSH_DOWN_TMP	-	0...3FFH	0...1023	1	[-]
Fault detection threshold for determination of defective heater					
R_IT_L_MES_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[Ohm]

Latest measured downstream oxygen sensor internal resistance value for no error case, mode 6 information					
R_IT_L_REF_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[Ohm]
Latest reference value for measured internal resistance downstream in error case, mode 6 information					
R_IT_H_MES_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[Ohm]
Latest measured downstream oxygen sensor internal resistance value for error case, mode 6 information					
R_IT_H_REF_OBD_LSH_DOWN[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	[Ohm]
Latest reference value for measured internal resistance downstream in error case, mode 6 information					

Input data:

LV_ST_END	R_IT_LS_DOWN[NC_CBK_EX_NR]	STATE_LSH_DOWN[NC_CBK_EX_NR]	LV_INH_DIAG_OBD_LSH_DOWN[NC_CBK_EX_NR]
MAF_KGH	LV_LS_DOWN_READY[NC_CBK_EX_NR]	LV_VB_CDN_OBD_2	LSHPWM_DOWN[NC_CBK_EX_NR]
LV_IGK	LV_ERR_EL_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]	CTR_CYCNR_R_IT_LS_DOWN_VLD[NC_CBK_EX_NR]
R_IT_MDL_LS_DOWN_NEW[NC_CBK_EX_NR]	NC_CBK_EX_NR	LV_DIAG_EOL_REQ_OBD_LSH_DOWN[NC_CBK_EX_NR]	TEG_DYN_LS_DOWN[NC_CBK_EX_NR]
LV_DIAG_EOL_END_LS_UP_DOWN[NC_CBK_EX_NR]	NC_STATE_LSL_UP_IF	LV_ERR_LS_DOWN[NC_CBK_EX_NR]	LV_DC

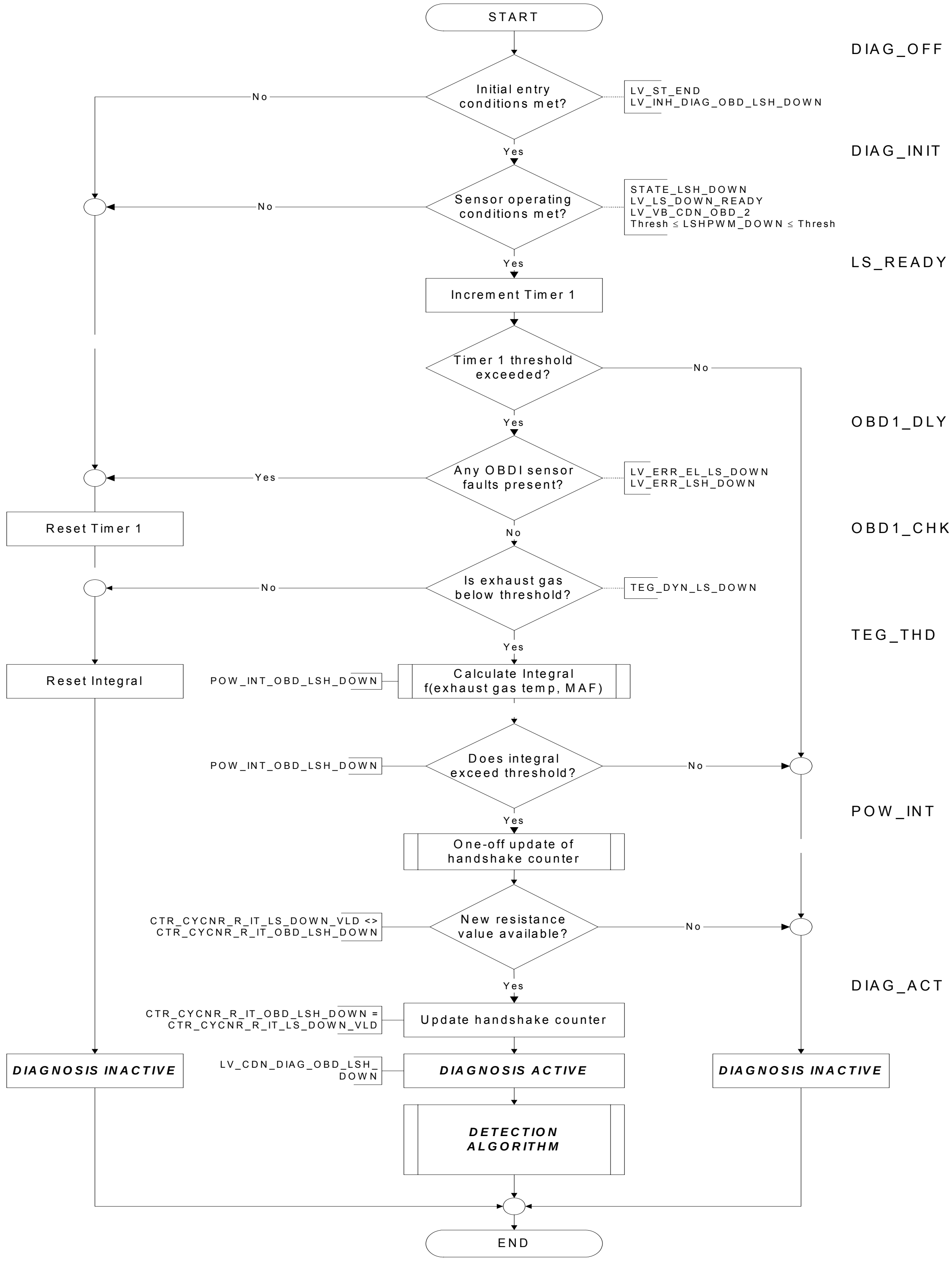
FUNCTION DESCRIPTION:

General information:

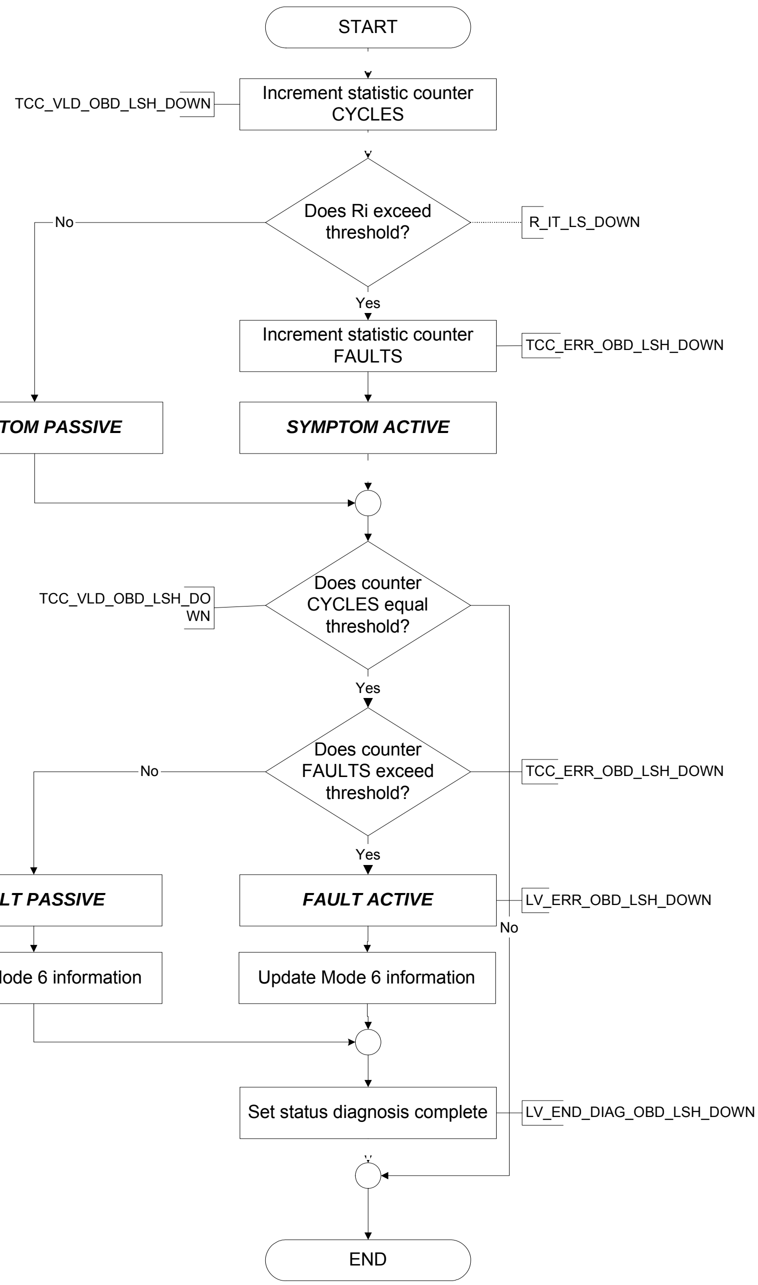
The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Signal flow diagram:

Activation conditions:



Detection Algorithm:



Description:

The downstream oxygen sensor heater circuit shall detect any loss in heater power that would cause a drop in the sensor operating temperature, thereby possibly causing exhaust gas emissions to rise above the applicable standards or prevent the sensor signal from being used as a diagnostic system monitoring device.

Losses in heater power may occur due to, for example, ageing of the heater element, defective wiring, increased heater cct. connector contact resistance, defective heater driver etc.

The hereafter mentioned diagnosis strategy is based on the comparison of the oxygen sensor internal resistance to an absolute threshold during operating conditions where the exhaust gas temperature has been determined to be sufficiently low as to cause the sensor ceramic temperature to fall outside normal operating levels, in cases where a power is insufficient.

The functionality of the diagnosis may be described in further detail below:

The initial diagnosis application condition; LV_ST_END shall ensure that the engine is running and has left the start phase but not yet entered the engine stop phase, LV_INH_DIAG_OBD_LSH_DOWN[i] shall ensure that the project specific application conditions (Application Incidences) have been met and LV_END_DIAG_OBD_LSH_DOWN[i] shall ensure that the diagnosis only be carried out once per driving cycle.

In order to ensure that the sensor has reached its normal operating temperature, the heater management state STATE_LSH_DOWN[i] shall be determined to be in the open loop control state LSH_POW_CTL, the sensor shall be determined to be in a state of operative readiness via LV_LS_DOWN_READY[i], the battery voltage VB shall be in a range determined to be fit for carrying out OBD2 diagnosis shown by LV_VB_CDN_OBD_2 and the heater duty-cycle LSHPWM_DOWN[i] shall be determined to be within calibrateable range C_LSHPWM_MIN_LSH_DOWN & C_LSHPWM_MAX_LSH_DOWN.

Should the above initial and operative conditions have been determined to be met, a timer, denoted herein as *TIMER_1*, shall be started. The timer shall run until the threshold T_DLY_OBD_LSH_DOWN_TMP is equalled or exceeded. Once this delay has passed, the status of the OBDI error bits LV_ERR_EL_LS_DOWN[i] & LV_ERR_LSH_DOWN[i] shall be read to ensure that no signal electrical or plausibility faults and no heater electrical faults are present in the sensor cct. under test. The delay shall ensure that sufficient time passes to permit completion of the OBDI monitoring prior to the start of OBDII monitoring.

Should no OBDI faults be present, the modelled exhaust gas temperature TEG_DYN_LS_DOWN[i] shall be compared to a calibrateable threshold C_TEG_DYN_MAX_OBD_LSH_DOWN. Due tolerances in the determination of the sensor internal resistance and in the case of a functional heater, the maximum resistance value may be given by a minimum tolerance sensor at a much lower temperature than for a maximum tolerance sensor at a higher temperature. Therefore, in the case of diagnosis based on an absolute internal resistance threshold, a certain temperature tolerance exists. The above mentioned temperature threshold shall lie below the minimum temperature determined from analysis of the internal resistance tolerances and be sufficient to cool the sensor ceramic element to below normal operating conditions, in the case of a heater fault.

Should the exhaust gas temperature be determined to equal or fall below the threshold, the monitoring strategy shall ensure that the sensor be cooled sufficiently long by the exhaust gas to enable differentiation between a functional and non-functional heater cct. (The time required is dependent on the thermal capacity of the system at the sensor location and the forced convection cooling of the sensor location). This shall be achieved by integrating the heat flux of the forced convection thermal transfer between sensor and exhaust gas, this being a measure for the cooling energy of the exhaust gas. The heat flux shall be calculated by determining the difference between the typical set sensor operating temperature C_TEMP_SP_OBD_LSH_DOWN and the modelled exhaust gas temperature TEG_DYN_LS_DOWN[i] and multiplying the result by the factor IP_FAC_MAF_OBD_LSH_DOWN representing the heat transfer coefficient and surface area concerned in the thermal transfer. As the heat transfer coefficient is a function of the velocity of the exhaust gas, the factor shall be mapped and dependent on the current mass airflow MAF_KGH.

The resultant integral POW_INT_OBD_LSH_DOWN[i] shall be compared to the threshold POW_INT_MIN_OBD_LSH_DOWN_TMP. Each time that the threshold be exceeded for the first time, e.g. first time after engine start or after the power integral has been reset, the CTR_CYCNR_R_IT_OBD_LS_DOWN[i] shall be updated once only with the value CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. This shall ensure that only new internal resistance values be used for diagnosis purposes that have been determined since all the activation conditions have been met.

It shall be determined whether a new internal resistance has been determined by comparing the contents of counter CTR_CYCNR_R_IT_OBD_LS_DOWN[i] with that of CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. Should the counters be unequal, then a new value is available and CTR_CYCNR_R_IT_OBD_LS_DOWN[i] shall be updated with CTR_CYCNR_R_IT_LS_DOWN_VLD[i]. If this conditions and all the above conditions be met, LV_CDN_DIAG_OBD_LSH_DOWN[i] shall be set and the diagnosis shall be considered to be active and the current monitoring cycle valid.

Once the current monitoring cycle has been determined to be valid, a counter TCC_VLD_OBD_LSH_DOWN[i] shall be incremented, denoting the number of valid monitoring cycles carried out to date. The internal resistance R_IT_LS_DOWN[i] shall be compared to a calibrateable threshold IP_R_IT_THD_OBD_LSH_DOWN which is dependent on the modelled tip temperature of a new sensor (including safety margin) and the number of valid diagnosis cycles completed TCC_VLD_OBD_LSH_DOWN. Should the resistance equal or exceed the threshold, a heater fault shall be determined to be present in the current cycle and a counter TCC_ERR_OBD_LSH_DOWN[i] shall be incremented, denoting the number of heater faults within the valid monitoring cycles carried out to date.

In order to comply with OBDII requirements, the current R_IT_LS_DOWN[i] value used shall be made available to an external tester. This information is Mode 6 information. Hence once the comparison to the threshold has been carried out, the latest measured internal resistance and threshold values (R_IT_L_MES_OBD_LSH_DOWN[i] and R_IT_L_REF_OBD_LSH_DOWN[i]) shall be copied to the respective Mode 6 interface variables R_IT_OBD_LS_DOWN[i] and R_IT_THD_OBD_LSH_DOWN[i] for the no error case. In the error case the variables R_IT_H_MES_OBD_LSH_DOWN[i] and R_IT_H_REF_OBD_LSH_DOWN[i] shall be handed out.

Once the number of valid monitoring cycles TCC_THD_OBD_LSH_DOWN_TMP has been determined to have been carried out, the diagnosis shall carry out a n-out-of-m statistical evaluation of the results. Should the number of determined heater faults exceed the threshold TCC_THD_ERR_OBD_LSH_DOWN_TMP, the oxygen sensor heater under test shall be considered to be faulty for the current driving cycle and flag LV_ERR_OBD_LSH_DOWN[i] shall be set.

If the number of faults does not exceed the threshold, LV_ERR_OBD_LSH_DOWN[i] shall be reset.

Once the statistical evaluation has been carried out, no further monitoring cycles shall be executed in the remaining driving cycle. This is achieved by setting LV_END_DIAG_OBD_LSH_DOWN[i].

Should, according to project philosophy, the diagnosis be activated for an end of line (EOL) test, specific threshold conditions for the diagnosis delay timer, the power integral, the required number of test and error indicating cycles may be chosen to shorten the time until recognition of diagnosis end.

NOTES:

F. The integral POW_INT_OBD_LSH_DOWN[i] shall be reset to 0 should the exhaust gas temperature exceed the applicable threshold. This would cause the sensor to be re-heated from the exhaust gas once again and a certain delay would be required prior to the sensor being cooled long enough to permit valid diagnosis to occur.

G. The timer *TIMER_1* and integral POW_INT_OBD_LSH_DOWN[i] shall be reset should any of the initial entry or operative readiness conditions no longer be met or an OBDI fault be detected in the sensor under test. This may cause the sensor to leave its normal operating conditions and the OBDII diagnosis shall be required to start from the beginning as the sensor operating conditions would require time to stabilise should the conditions be met again.

H. The statistic counters TCC_VLD_OBD_LSH_DOWN[i] and TCC_ERR_OBD_LSH_DOWN[i] shall not be reset by changing conditions and therefore shall only increment or remain frozen during any given monitoring cycle. The counters shall only be reset at the beginning of a new driving cycle.

I. Projects that provide for an internal statistical error management shall use their internal error management and apply this wherever reference to TCC_VLD_OBD_LSH_DOWN[i], TCC_ERR_OBD_LSH_DOWN[i], TCC_THD_OBD_LSH_DOWN_TMP & TCC_THD_ERR_OBD_LSH_DOWN_TMP is made.

J. The variable STATE_OBD_LSH_DOWN[i] shall indicate, at the end of the each monitoring cycle, the status of the diagnosis within that cycle. The following correlation of variable content to description shall apply:

0:	DIAG_OFF	Initialisation state; no conditions met
1:	DIAG_INIT	Initial conditions & application incidences met only
2:	LS_READY	Sensor operatively ready conditions met + 1,
3:	OBD1_DLY	OBDI monitoring completion delay exceeded + 1 & 2
4:	OBD1_CHK	No OBDI faults present +1, 2 & 3
5:	TEG_THD	Exhaust gas below threshold + 1, 2, 3 & 4
6:	POW_INT	Power integral exceeded + 1,2,3,4 & 5
7:	DIAG_ACT	***Diagnosis detection active***
8:	DIAG_END	***Diagnosis detection complete***

Application conditions:

Initialisation:

The following variables shall be initialised at reset, at the transition ignition key on (i.e. LV_IGK = 0 -> 1) and at clearing the error memory:

LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0
ERR_SYM_OBD_LSH_DOWN[i] = 0
LV_ERR_OBD_LSH_DOWN[i] = 0
STATE_OBD_LSH_DOWN[i] = 0
TEMP_DIF_OBD_LSH_DOWN[i] = 0
POW_INT_OBD_LSH_DOWN[i] = 0
TCC_VLD_OBD_LSH_DOWN[i] = 0
TCC_ERR_OBD_LSH_DOWN[i] = 0
CTR_CYCNR_R_IT_OBD_LS_DOWN[i] = 0
R_IT_L_MES_OBD_LSH_DOWN[i] = 0
R_IT_L_REF_OBD_LSH_DOWN[i] = 0
R_IT_H_MES_OBD_LSH_DOWN[i] = 0
R_IT_H_REF_OBD_LSH_DOWN[i] = 0

The following variable shall be initialised at reset, at every new driving cycle (i.e. LV_DC = 0 -> 1) and upon clearing error memory:

LV_END_DIAG_OBD_LSH_DOWN[i] = 0

Reset *TIMER_1*

R_IT_OBD_LSH_DOWN[i] and R_IT_THD_OBD_LSH_DOWN[i] shall be initialised with the last value from the previous driving cycle stored in non-volatile memory after RESET.

NOTE: TIMER_1 refers to a program specific timer and is not intended to represent a variable name!

Recurrence:

The OBDII oxygen sensor heater diagnosis shall be carried out once per driving cycle. The diagnosis shall require a number of monitoring cycles to be carried out to permit statistical evaluation of the results. These monitoring cycles shall occur every 1 s, until a calibrateable number of valid diagnosis cycles has been exceeded.

Once this threshold has been completed, the OBDII oxygen sensor heater diagnosis shall no longer be carried out until a new driving cycle is initiated.

Should an OBDII heater fault be detected, the application assistances (Functions that are defined to be affected by the state of the oxygen sensor signal) shall remain active for the remainder of the driving cycle. The restarting of the affected functions during the next driving cycle shall be carried out according to the project philosophy.

Activation:

If (LV_ST_END = 1) **&**
(LV_INH_DIAG_OBD_LSH_DOWN[i] = 0)

then If LV_DIAG_EOL_REQ_OBD_LSH_DOWN[i] = 0

then T_DLY_OBD_LSH_DOWN_TMP = C_T_DLY_OBD_LSH_DOWN

POW_INT_MIN_OBD_LSH_DOWN_TMP = C_POW_INT_MIN_OBD_LSH_DOWN

TCC_THD_OBD_LSH_DOWN_TMP = C_TCC_THD_OBD_LSH_DOWN

TCC_THD_ERR_OBD_LSH_DOWN_TMP = C_TCC_THD_ERR_OBD_LSH_DOWN

else T_DLY_OBD_LSH_DOWN_TMP = C_T_DLY_OBD_LSH_DOWN_EOL

POW_INT_MIN_OBD_LSH_DOWN_TMP = C_POW_INT_MIN_OBD_LSH_DOWN_EOL

TCC_THD_OBD_LSH_DOWN_TMP = C_TCC_THD_OBD_LSH_DOWN_EOL

TCC_THD_ERR_OBD_LSH_DOWN_TMP = C_TCC_THD_ERR_OBD_LSH_DOWN_EOL

endif

If (LV_LS_DOWN_READY[i] = 1) **&**
(STATE_LSH_DOWN[i] = LSH_POW_CTL) **&**
(LV_VB_CDN_OBD_2 = 1) **&**
(C_LSHPWM_MIN_LSH_DOWN ≤ LSHPWM_DOWN[i]) **&**
(LSHPWM_DOWN[i] ≤ C_LSHPWM_MAX_LSH_DOWN)

then Increment *TIMER_1*
If (*TIMER_1* ≥ T_DLY_OBD_LSH_DOWN_TMP)

```
then If  (LV_ERR_EL_LS_DOWN[i] = 0) &
         (LV_ERR_LSH_DOWN[i] = 0)

     then If  (TEG_DYN_LS_DOWN[i] ≤
              C_TEG_DYN_MAX_OBD_LSH_DOWN)

          then Calculate POW_INT_OBD_LSH_DOWN[i]
               If (POW_INT_OBD_LSH_DOWN[i] ≥
                  POW_INT_MIN_OBD_LSH_DOWN_TMP)

               then One-off update CTR_CYCNR_R_IT_OBD_LS_DOWN[i]

                    % see note below

                    If  (CTR_CYCNR_R_IT_LS_DOWN_VLD[i] <>

                        CTR_CYCNR_R_IT_OBD_LS_DOWN[i])

                    then LV_CDN_DIAG_OBD_LSH_DOWN[i] = 1

                         CTR_CYCNR_R_IT_OBD_LS_DOWN[i] =

                         CTR_CYCNR_R_IT_LS_DOWN_VLD[i]

                         „Diagnosis detection active“

                    else STATE_OBD_LSH_DOWN[i] = POW_INT
                         LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

                    endif

               else STATE_OBD_LSH_DOWN[i] = TEG_THD
                    LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

               endif

          else STATE_OBD_LSH_DOWN[i] = OBD1_CHK
               Reset POW_INT_OBD_LSH_DOWN[i]
               LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

          endif

     else STATE_OBD_LSH_DOWN[i] = OBD1_DLY
          Reset TIMER_1
          Reset POW_INT_OBD_LSH_DOWN[i]
          LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

     endif

else STATE_OBD_LSH_DOWN[i] = LS_READY
     LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

endif

else STATE_OBD_LSH_DOWN[i] = DIAG_INIT
     Reset TIMER_1
     Reset POW_INT_OBD_LSH_DOWN[i]
     LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0

endif

else STATE_OBD_LSH_DOWN[i] = DIAG_OFF
     Reset TIMER_1
     Reset POW_INT_OBD_LSH_DOWN[i]
     LV_CDN_DIAG_OBD_LSH_DOWN[i] = 0
```

endif.

Deactivation:

If (LV_END_DIAG_OBD_LSH_DOWN[i] = 1)

then „Diagnosis detection complete“

endif.

NOTES: The state variable STATE_OBD_LSH_DOWN[i] *shall only be validated at the completion of the current monitoring cycle, i.e. the state should not vary within the current 1 s monitoring cycle, but indicate at the end of the current cycle , the last state reached, dependant upon the branches.*

The variable CTR_CYCNR_R_IT_OBD_LS_DOWN[i] shall be updated with the current value of the input variable CTR_CYCNR_R_IT_LS_DOWN_VLD[i] on each occasion that the integral POW_INT_OBD_LSH_DOWN[i] exceeds the threshold for the first time. For further function calls where the integral condition is met, the one-off update shall not be carried out. This shall continue until the integral condition is no longer met.

Formula section:

Calculation of cooling energy integral:

Determine difference between typical sensor operating temperature and exhaust gas temperature:

TEMP_DIF_OBD_LSH_DOWN[i] = (C_TEMP_SP_OBD_LSH_DOWN - TEG_DYN_LS_DOWN[i])

(NOTE: Resolution of temperature difference is reduced from 0.0625 °C, resolution of the individual temperature variables, to 1 °C; The difference shall be limited to provide only positive values, i.e. ≥ 0)

The integral of heat flux (in Watts) provides a measure for the total cooling energy of the exhaust gas (in Joules):

$POW\_INT\_OBD\_LSH\_DOWN[i]_{(n)}$ = ($POW\_INT\_OBD\_LSH\_DOWN[i]_{(n-1)}$ + TEMP_DIF_OBD_LSH_DOWN[i] * IP_FAC_MAF_OBD_LSH_DOWN)

(NOTE: Integral shall be limited to its maximum hex limit, i.e. FFFFH or 65535 digits, to prevent overflow.)

Detection: (Assumes "Diagnosis detection active") **See also NOTE D above!**

Increment TCC_VLD_OBD_LSH_DOWN[i]

If (R_IT_LS_DOWN[i] ≥ IP_R_IT_THD_OBD_LSH_DOWN)

then Symptom "Oxygen sensor heater fault, bank i" active

ERR_SYM_OBD_LSH_DOWN[i] = "SYM_0"

R_IT_H_MES_OBD_LSH_DOWN[i] = R_IT_LS_DOWN[i]

R_IT_H_REF_OBD_LSH_DOWN[i] = IP_R_IT_THD_OBD_LSH_DOWN

Increment TCC_ERR_OBD_LSH_DOWN[i]

else Symptom "Oxygen sensor heater fault, bank i" passive

ERR_SYM_OBD_LSH_DOWN[i] = "NO_SYM"

R_IT_L_MES_OBD_LSH_DOWN[i] = R_IT_LS_DOWN[i]

R_IT_L_REF_OBD_LSH_DOWN[i] = IP_R_IT_THD_OBD_LSH_DOWN

endif.

If (TCC_VLD_OBD_LSH_DOWN[i] ≥ TCC_THD_OBD_LSH_DOWN_TMP)

then STATE_OBD_LSH_DOWN[i] = DIAG_END

  If (TCC_ERR_OBD_LSH_DOWN[i] ≥ TCC_THD_ERR_OBD_LSH_DOWN_TMP)

  then **Symptom “Oxygen sensor heater fault, bank i” active**

    LV_ERR_OBD_LSH_DOWN[i] = 1

    ERR_SYM_OBD_LSH_DOWN[i] = " SYM_0"

    R_IT_OBD_LSH_DOWN[i] = R_IT_H_MES_OBD_LSH_DOWN[i]

    R_IT_THD_OBD_LSH_DOWN[i] = R_IT_H_REF_OBD_LSH_DOWN[i]

  else **Symptom “Oxygen sensor heater fault, bank i” passive**

    LV_ERR_OBD_LSH_DOWN[i] = 0

    ERR_SYM_OBD_LSH_DOWN[i] = "NO_SYM"

    R_IT_OBD_LSH_DOWN[i] = R_IT_L_MES_OBD_LSH_DOWN[i]

    R_IT_THD_OBD_LSH_DOWN[i] = R_IT_L_REF_OBD_LSH_DOWN[i]

  endif

else STATE_OBD_LSH_DOWN[i] = DIAG_ACT

endif.

The diagnosis end bit shall be calculated each recurrence independently on STATE_OBD_LSH_DOWN[i] until diagnosis deactivation:

If STATE_OBD_LSH_DOWN[i] = DIAG_END

or ... **#If** NC_STATE_LSL_UP_IF > 0
#then **...** (LV_DIAG_EOL_END_LS_UP_DOWN[i] = 1
and LC_DIAG_EOL_END_LSH_DOWN[i] = 1
and LV_ERR_LS_DOWN[i] = 0)
#endif

then LV_END_DIAG_OBD_LSH_DOWN[i] = 1
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LSHPWM_MIN_LSH_DOWN	1	0...FFH	0...99.60937	0.390625	[%]
Minimum permitted heater effective voltage PWM for downstream LSH OBDII diagnosis					
C_LSHPWM_MAX_LSH_DOWN	1	0...FFH	0...99.60937	0.390625	[%]
Maximum permitted heater effective voltage PWM for downstream LSH OBDII diagnosis					
C_T_DLY_OBD_LSH_DOWN	1	0...FFH	0...255	1	[s]
Minimum delay prior to checking status of OBDI heater / signal faults					
C_T_DLY_OBD_LSH_DOWN_EOL	1	0...FFH	0...255	1	[s]
Minimum delay prior to checking status of OBDI heater / signal faults when EOL test required					
C_TEG_DYN_MAX_OBD_LSH_DOWN	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Temperature threshold under which downstream LSH OBDII diagnosis activated					
C_TEMP_SP_OBD_LSH_DOWN	1	0...7FFFH	-273.15... 1774.7875	0.0625	[°C]
Set temperature for downstream LSH operation used to generate power integral					
C_POW_INT_MIN_OBD_LSH_DOWN	1	0...FFFFH	0...65535	1	[J]
Measure of minimum required cooling energy required at downstream sensor prior to checking Ri threshold					
C_POW_INT_MIN_OBD_LSH_DOWN_EOL	1	0...FFFFH	0...65535	1	[J]
Measure of minimum required cooling energy required at downstream sensor prior to checking Ri threshold when EOL test required					
IP_R_IT_THD_OBD_LSH_DOWN	4*4	0...FFFFH	0...65535	1	[Ohm]
LDP_R_IT_LS_DOWN_NEW_LSH_DOWN	4	0...FFFFH	0...65535	1	[Ohm]
LDP_TCC_VLD_OBD_LSH_DOWN	4	0...FFFFH	0...65535	1	-
Internal resistance threshold, when exceeded, heater is determined to be defective for current diagnosis cycle; including reference to modelled value for new sensor					
IP_FAC_MAF_OBD_LSH_DOWN	6	0...FFH	0...15.9375	0.0625	[J/K]
LDPM_MAF_KGH_3_EGCP	6	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Factor reflecting forced convection influence of cooling area, gas velocity and heat transfer coefficient					
C_TCC_THD_OBD_LSH_DOWN	1	0...3FFH	0...1023	1	[-]
Number of valid downstream LSH OBDII diagnosis cycles over which statistic evaluation to be carried out					
C_TCC_THD_OBD_LSH_DOWN_EOL	1	0...3FFH	0...1023	1	[-]
Number of valid downstream LSH OBDII diagnosis cycles over which statistic evaluation to be carried out when EOL test required					
C_TCC_THD_ERR_OBD_LSH_DOWN	1	0...3FFH	0...1023	1	[-]
Fault detection threshold, when exceeded, heater is determined to be defective during driving cycle					
C_TCC_THD_ERR_OBD_LSH_DOWN_EOL	1	0...3FFH	0...1023	1	[-]
Fault detection threshold, when exceeded, heater is determined to be defective during driving cycle when EOL test required					
LC_DIAG_EOL_END_LSH_DOWN[NC_CBK_EX_NR]	1	0...1H	0...1	1	[-]
Locigal calibration data to enforce end of diagnosis when EOL test has passed					

See also NOTE D above!

Configuration for diagnostic symptoms:

Diagnostic XX	Symptom description	Symptom	Filter type
OBD_LSH_DOWN[NC_CBK_EX_NR]	*Heater fault*	SYM_0	NO

B.24.1 Downstream oxygen sensor heater OBDII monitoring (Appl. Inc.)

B.24.1.1 Inhibition of diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Boolean inhibit flag for O2 sensor heater OBDII monitoring					
LV_INH_DIAG_RBM_OBD_LSH_DOWN[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag to inhibit rate based monitoring for oxygen sensor heater OBDII diagnosis					

Input data:

LV_ERR_MAP	LV_ERR_MAF	LV_ERR_RATIO_CHK	LV_ST_END
LV_IGK	NC_CBK_EX_NR	NLC_LSH_RLY_EFP	LV_EFP
LV_ERR_RLY_EFP	TCO_ST	TCO_STOP	CONF_LAM
CONF_CAT_EFF			

FUNCTION DESCRIPTION:

General information:

This function shall evaluate the application conditions applicable to the oxygen sensor heater OBDII monitoring and provide the result in a single boolean flag. Additionally the interfaces for Rate Based Monitoring RBM are provided.

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
 i = 1, for exhaust cylinder bank 1
 i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
 i = 1, for single exhaust cylinder bank.

Description:

The oxygen sensor heater OBD II monitoring shall be ihibited if one of the following listed errors was recognized:

- LV_ERR_MAF shall indicate a fault in the mass air flow measurement which may have an effect on the modelled exhaust gas temperature. Such a faulty value may cause the heating to be incorrectly performed leading to a secondary fault entry.
- LV_ERR_MAP shall indicate a fault in the manifold air pressure measurement which may have an effect on the modelled exhaust gas temperature (TEG_DYN). Such a fault value may cause the heating to be incorrectly performed leading to a secondary fault entry.
- In some systems the positive feed of the oxygen sensor heater may be connected to the electrical fuel pump relay as indicated by NLC_LSH_RLY_EFP = 1. In this case, if the electrical fuel pump is switched off or if the fuel pump relay is defective, no heating will occur and the heater driver will recognize an incorrect heater fault. Hence the OBD II heater diagnosis shall be inhibited if LV_EFP = 0 or if LV_ERR_RLY_EFP = 1.
- LV_ERR_RATIO_CHK shall indicate a fault due to the load – throttle position sensor diagnosis. Such a fault may cause incorrect heating to be performed.

- If TCO_ST has not decreased sufficiently compared with TCO_STOP the exhaust system is still warm and a heater error cannot be detected, therefore the OBD2 diagnosis is inhibited

The same OBD I errors are evaluated for inhibition of Rate Based Monitoring.

Application conditions:

Initialisation:

The following variables shall be initialised at reset and at the ignition key transition on (i.e. LV_IGK = 0 -> 1):

LV_INH_DIAG_OBD_LSH_DOWN[i] = 0

Recurrence:

The function shall be carried out once every 1s.

Activation:

LV_ST_END = 1

Deactivation:

LV_ST_END = 0

Formula section:

Inhibition of Rate Based Monitoring

If LV_ERR_MAF = 1 **or**
LV_ERR_MAP = 1 **or**
LV_ERR_RATIO_CHK = 1
then
LV_INH_DIAG_RBM_OBD_LSH_DOWN[i] = 1
else
LV_INH_DIAG_RBM_OBD_LSH_DOWN[i] = 0
endif.

Inhibition of oxygen sensor heater OBD II diagnosis

If TCO_STOP – TCO_ST < C_TCO_DIF_MIN_OBD_LSH_DOWN **or**
LV_ERR_MAF = 1 **or**
LV_ERR_MAP = 1 **or**
LV_ERR_RATIO_CHK = 1 **or**
CONF_LAM = 0 **or**
CONF_CAT_EFF = 0
#if (NLC_LSH_RLY_EFP = 1)
#then LV_EFP = 0 **or**

LV_ERR_RLY_EFP = 1

#endif

then

LV_INH_DIAG_OBD_LSH_DOWN[i] = 1

else

LV_INH_DIAG_OBD_LSH_DOWN[i] = 0

endif.

Configuration for diagnostic symptoms :

Diagnostic	Symptom description	Symptom	Filter type
Downstream O2 sensor heater OBDII monitoring	O2 sensor heater circuit malfunction, bank 1, sensor 2	SYM_0	*NO*
		SYM_1	
		SYM_2	
OBD_LSH_DOWN[1]		SYM_3	

List of Environmental Data to store in failure memory: TCO_ST
LSHPWM_DOWN_1
R_IT_LS_DOWN_1_ENVD_H
R_IT_LS_DOWN_1_ENVD_L

Application assistances:

The Application Assistances shall be maintained for the remaining duration of the driving cycle.

- Lambda control (Only for exhaust gas temperatures lower than O2 sensor lowest operating temperature):
  Downstream trim regulation - Inhibit (Bank selectively)
- Oxygen sensor / Lambda controller diagnosis:
  O2 sensor downstream signal diagnosis – Inhibit (Bank selectively, OC & SCB)
  O2 sensor downstream response time diagnosis – Inhibit (Bank selectively)
  O2 sensor downstream signal voltage diagnosis – Inhibit (Bank selectively)
  O2 sensor downstream PUC end diagnosis – Inhibit (Bank selectively)
- Catalyst efficiency diagnosis - Inhibit (Bank selectively) (indirectly via LV_LSCL)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_DIF_MIN_OBD_LSH_DOWN	1	0 ... FEH	-48 ... 142.5	0.75	°C
Minimum drop in TCO_ST compared to TCO_STOP for OBD_LSH_DOWN activation					

B.24.1.2 Interface for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_OBD_LSH_DOWN[NC_CBK_EX_NR]	O/V	0 ... 7H	0 ... 7	1	-
Interface of OBD_LSH_DOWN monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_ OBD_LSH_DOWN_1	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Downstream O2 sensor OBDII heater diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_OBD_LSH_DOWN[NC_CBK_EX_NR]	LV_INH_DIAG_RBM_OBD_LSH_DOWN[NC_CBK_EX_NR]
NC_CBK_EX_NR	NLC_LSH_RLY_EFP	CTR_CDN_RBM_OBD_LSH_DOWN_1	CTR_COMP_RBM_OBD_LSH_DOWN_1
LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2	

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the OBD_LSH_DOWN[i] monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_OBD_LSH_DOWN[i] data.

Within STATE_RBM_OBD_LSH_DOWN[i], three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_OBD_LSH_DOWN[i] = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 0

on failure memory reset :

bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_OBD_LSH_DOWN[i] = 1

The pending status of the following failures has to be checked only once:

LV_ERR_EL_MAP	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAF_SCP	LV_ERR_MAF_SCG_OC
LV_ERR_RLY_EFP	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2	

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 0 **do**

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

with all XX failures of the above list.

The necessity of monitoring LV_ERR_RLY_EFP depends on system configuration, i.e. whether NLC_LSH_RLY_EFP is set or not (see section 1.1).

#If (NLC_LSH_RLY_EFP = 1)

#Then LV_ERR_RLY_EFP

#Endif

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If (bit 0 of STATE_RBM_OBD_LSH_DOWN[i] = 0 **and**
LV_END_DIAG_OBD_LSH_DOWN[i] = 1)
Then bit 0 of STATE_RBM_OBD_LSH_DOWN[i] = 1
Endif

If (bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 0 **and**
LV_INH_DIAG_RBM_OBD_LSH_DOWN[i] = 1)
Then bit 1 of STATE_RBM_OBD_LSH_DOWN[i] = 1
Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_OBD_LSH_DOWN_1 = 0
then RATE_RBM_OBD_LSH_DOWN_1 = FFFFH
else RATE_RBM_OBD_LSH_DOWN_1
= (CTR_COMP_RBM_OBD_LSH_DOWN_1 / CTR_CDN_RBM_OBD_LSH_DOWN_1)
endif

B.25 Upstream oxygen sensor diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DIAG_PLAUS_SYM_LSL_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Debounced diagnosis result of the signal plausibility for the wide range A/F sensor					
LV_DIAG_MPL_END_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit to indicate end of complete upstream sensor diagnosis					
LV_CDN_DIAG_MPL_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Status of permission for multiple diagnosis upostream					

Input data:

LV_END_DIAG_SWT_LS_UP[NC_CBK_EX_NR]	LV_ST_END	LV_END_DIAG_FRQ_LS_UP[NC_CBK_EX_NR]	LV_IGK
NC_CBK_EX_NR	LV_INH_DIAG_LS_UP[NC_CBK_EX_NR]		

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

This function indicates the completion of the upstream oxygen sensor diagnosis by setting the boolean variable LV_DIAG_MPL_END_LS_UP[i]. LV_DIAG_PLAUS_SYM_LSL_UP[i] is generated to serve a common interface to Appl. Inc. of heater management.

Application conditions:

Initialisation:

Recurrence:

This function shall be carried out every 100ms.

Activation:

LV_ST_END = 1

Deactivation:

LV_ST_END = 0

Formula section:

If LV_END_DIAG_SWT_LS_UP[i] = 1
and LV_END_DIAG_FRQ_LS_UP[i] = 1
then LV_DIAG_MPL_END_LS_UP[i] = 1
endif

If LV_INH_DIAG_LS_UP [i] = 0
then LV_CDN_DIAG_MPL_LS_UP[i] = 1
else LV_CDN_DIAG_MPL_LS_UP[i] = 0
endif

B.26 OBDII Oxygen Sensor diagnosis with multiple check

B.26.1 Oxygen sensor switching time monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
status diagnostic result of switch-time monitoring					
ERR_SYM_SWT_LS_UP[NC_CBK_EX_NR]	V/O	0H 8H	NO_SYM Sym_3	1	-
status of determined diagnostic result of switch-time monitoring					
LV_END_DIAG_SWT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
diagnostic cycle for switching time complete					
LV_END_DIAG_RBM_SWT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
diagnostic cycle for limit switching time complete					
LV_CDN_DIAG_SWT_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
status conditions of switch time diagnosis of lambda sensor upstream					
VLS_THD_MAX[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828E-3	V
top limit diagnostic threshold for switching time monitoring					
VLS_THD_MIN[NC_CBK_EX_NR]	V	0...3FFH	0...4.995117	4.8828E-3	V
bottom limit diagnostic threshold for switching time monitoring					
SWT_THD_MAX_MIN[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
actual switching time					
SWT_THD_MAX_MIN_OLD[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
saved switching time value of last measured switching event					
SWT_THD_RBM_MAX_MIN[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
additional switching time for limit lambda probe					
SWT_SUM_RBM_MAX_MIN[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of measured switching time and SWT_THD_RBM_MAX_MIN[i] for limit lambda probe					
RATIO_MV_CYCNR_AFL[NC_CBK_EX_NR]	V/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed switching times from rich to lean					
RATIO_MV_CYCNR_AFR[NC_CBK_EX_NR]	V/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed switching times from lean to rich					
CYCNR_AFL_SUM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured switching times from rich to lean					
CYCNR_RBM_AFL_SUM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured RBM limit switching times from rich to lean					
CYCNR_AFR_SUM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured switching times from lean to rich					
CYCNR_RBM_AFR_SUM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured RBM limit switching times from lean to rich					
RATIO_CYCNR_AFL[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio between measured and max. allowed switching times from rich to lean					
RATIO_CYCNR_RBM_AFL[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio for limit rich to lean switching time					
RATIO_MV_SAVE_CYCNR_AFL[NC_CBK_EX_NR]	V/S/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed switching times from rich to lean for scantool					
RATIO_CYCNR_AFR[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio between measured and max. allowed switching times from lean to rich					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
RATIO_CYCNR_RBM_AFR[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio for limit lean to rich switching time					
RATIO_MV_SAVE_CYCNR_AFR[NC_CBK_EX_NR]	V/S/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed switching times from lean to rich for scantool					
RATIO_SUM_CYCNR_AFL[NC_CBK_EX_NR]	V	0...FFFFH	0...1023.984375	1.56*10-2	-
total value of switching time from rich to lean					
RATIO_SUM_CYCNR_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...1023.984375	1.56*10-2	-
total value of switching time from lean to rich					
T_SUM_SWI_AFR_AFL[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of SWT rich to lean					
T_SUM_MAX_SWI_AFR_AFL[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of max SWT rich to lean					
T_SUM_SWI_AFL_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of SWT lean to rich					
T_SUM_MAX_SWI_AFL_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of max SWT lean to rich					
VLS_SAVE_THD_MIN[NC_CBK_EX_NR]	V/O/S	0...3FFH	0...4.995117	4.8828E-3	V
SWT lean threshold for scantool					
VLS_SAVE_THD_MAX[NC_CBK_EX_NR]	V/O/S	0...3FFH	0...4.995117	4.8828E-3	V
SWT rich threshold for scantool					
T_SAVE_SWI_AFR_AFL[NC_CBK_EX_NR]	V/O/S	0...FFH	0...2.55	0.01	s
SWT rich to lean for scantool					
T_MAX_SAVE_SWI_AFR_AFL[NC_CBK_EX_NR]	V/O/S	0...FFH	0...2.55	0.01	s
max. SWT rich to lean for scantool					
T_SAVE_SWI_AFL_AFR[NC_CBK_EX_NR]	V/O/S	0...FFH	0...2.55	0.01	s
SWT lean to rich for scantool					
T_MAX_SAVE_SWI_AFL_AFR[NC_CBK_EX_NR]	V/O/S	0...FFH	0...2.55	0.01	s
max. SWT lean to rich for scantool					
LV_DIAG_SWT_LS_UP_AFR_ENA[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag to crosslock switching time determination					
LV_DIAG_SWT_LS_UP_RBM_AFR[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that limit switching time determination for lean to rich transition active					
LV_DIAG_SWT_LS_UP_RBM_AFL[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that limit switching time determination for rich to lean transition active					
LV_DIAG_SWT_LS_UP_AFR_CDN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Logical variable indicating rich air fuel ratio before start of switching time determination					
LV_DIAG_SWT_LS_UP_AFL_CDN[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Logical variable indicating lean air fuel ratio before start of switching time determination					

Input data:

VLS_UP_MMV_MAX[NC_CBK_EX_NR]	VLS_UP_MMV_MIN[NC_CBK_EX_NR]	VLS_UP[NC_CBK_EX_NR]	MAF_KGH
LV_CDN_DIAG_MPL_LS_UP[NC_CBK_EX_NR]	LV_DC		

FUNCTION DESCRIPTION:

General information:

If an error has been registered within the corresponding engine operating cycle, which disables the oxygen sensor switching time monitoring, the checking algorithm for this engine operating cycle is aborted without rendering a result. If an error has been debounced for the oxygen sensor switching time monitoring, catalyst efficiency monitoring must be aborted for the corresponding cylinder group in the corresponding engine operating cycle.

Description:

In order to detect oxygen sensor switching problems, the sensor signals are monitored for reversal points while the switching times are being measured. If a reversal point is detected, this oxygen sensor signal change is not used to determine the sensor switching time. Since the sensor switching times fluctuate considerably during normal engine operation, the switching times are averaged.

The oxygen sensor signals are sampled every 10 ms. (This precludes an accurate determination of the individual sensor switching times. Only greater deviations of the oxygen sensor switching times from the normal value can be detected.)

When the oxygen sensor switching time monitoring has been completed rendering a result, the function indicates to the error storage mechanism that it is ready for the corresponding engine operating cycle.

Simultaneously to measuring the switching times from rich to lean the switching time for a limit sensor is determined out of SWT_THD_MAX_MIN[i] and an additional timer SWT_THD_RBM_MAX_MIN[i] which is increased until RATIO_CYCNR_RBM_AFL[i] fulfills the error criterion. The same holds for the switching time from lean to rich. Each time the error criterion is reached the counters CYCNR_RBM_AFL_SUM[i] and CYCNR_RBM_AFR_SUM[i] are increased. The end bit for Rate Based Monitoring purpose is set if the diagnosis end threshold C_CYCNR_SUM_MIN is exceeded. This procedure ensures that within each voltage transition a switching time limit probe could have been detected as is demanded by ARB.

Application conditions:

Initialisation: At LV_IGK = 0 -> 1, reset or by clearing the error memory shall all variables be initialised to 0 except LV_END_DIAG_xx and the variables saved in the non-volatile memory.

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence:

This function shall be carried out every 10ms, except the “Determination of relative thresholds” which shall be carried out every 1s.

Activation / Deactivation:

If LV_CDN_DIAG_MPL_LS_UP[i] = 1

and LV_END_DIAG_RBM_SWT_LS_UP[i] = 0

then If C_CYCNR_SUM_MIN = 0

then *Switching time diagnosis switched off*

LV_END_DIAG_SWT_LS_UP[i] = 1

LV_END_DIAG_RBM_SWT_LS_UP[i] = 1

(*no further calculations for switching time diagnosis must be done)*

ERR_SYM_SWT_LS_UP[i] = "NO_SYM"

LV_ERR_SWT_LS_UP[i] = 0

else *Switching time diagnosis active*

LV_CDN_DIAG_SWT_LS_UP[i] = 1

endif

else *Switching time diagnosis passive*

LV_CDN_DIAG_SWT_LS_UP[i] = 0

LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 0

LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 0

LV_DIAG_SWT_LS_UP_AFR_CDN[i] = 0

LV_DIAG_SWT_LS_UP_AFL_CDN[i] = 0

endif

B.26.1.1 Determination of relative thresholds

Description:

The relative thresholds VLS_THD_MAX[i] and VLS_THD_MIN[i] (10-90% signal excursion of VLS_UP_MMV_MAX[i] and VLS_UP_MMV_MIN[i]) are used to determine the oxygen sensor switching times from lean to rich mixture operation and back.

Formula section:

// updated every 1s

VLS_THD_MAX[i] = VLS_UP_MMV_MIN[i] + (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) * 0.9

VLS_THD_MIN[i] = VLS_UP_MMV_MIN[i] + (VLS_UP_MMV_MAX[i] - VLS_UP_MMV_MIN[i]) * 0.1

B.26.1.2 Switching time from rich to lean

While the oxygen sensor signal VLS_UP[i] is greater than VLS_THD_MAX[i], the counter SWT_THD_MAX_MIN[i] is set to 0. When the signal drops below the threshold VLS_THD_MAX[i], the counter SWT_THD_MAX_MIN[i] is started. The logical variable LV_DIAG_SWT_LS_UP_AFR_CDN[i] ensures that VLS_UP[i] has exceeded VLS_THD_MAX[i] before switching time determination when set. While VLS_UP[i] > VLS_THD_MIN[i], the counter is incremented with each sensor sampling cycle. The counter SWT_THD_RBM_MAX_MIN[i] is incremented until the ratio RATIO_CYCNR_RBM_AFL[i] exceeds the limit for error detection.

If the oxygen sensor signal drops below the threshold VLS_THD_MIN[i], the counter is stopped.

If a „rich peak“ which exceeds $VLS\_UP[i]_{n-1}$ + C_VLS_HYS occurs while the switching time is being determined, the switching time determination is aborted.

Formula section:

```
If      LV_DIAG_SWT_LS_UP_AFR_ENA[i] = 1
Then  If     VLS_UP[i] > VLS_THD_MAX[i]
      then LV_DIAG_SWT_LS_UP_AFR_CDN[i] = 1
      endif
      if    LV_DIAG_SWT_LS_UP_AFR_CDN[i] = 1
      then  If    VLS_THD_MAX[i] > VLS_UP[i]
            then  if    VLS_UP[i]n < VLS_UP[i]n-1 + C_VLS_HYS
                  then  if    VLS_UP[i] > VLS_THD_MIN[i]
                        then  SWT_THD_MAX_MIN[i] = SWT_THD_MAX_MIN[i] + 1
                        else  stop switching time determination
                              calculation of average switching time rich/lean
                              LV_DIAG_SWT_LS_UP_AFR_CDN[i] = 0
                              LV_DIAG_SWT_LS_UP_AFR_ENA[i] = 0
                              if    LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 0
                              and   LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 0
                              and   (CYCNR_RBM_AFR_SUM[i] >
                                          CYCNR_RBM_AFL_SUM[i])
                              then  LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 1
                                    SWT_THD_MAX_MIN_OLD[i] =
                                          SWT_THD_MAX_MIN[i]
                              endif
                        endif
                  else  stop switching time determination
                        LV_DIAG_SWT_LS_UP_AFR_CDN[i] = 0
                  endif
            else  SWT_THD_MAX_MIN[i] = 0
            endif
      endif
endif


if          LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 1
and         LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 0
```

then SWT_THD_RBM_MAX_MIN[i] = SWT_THD_RBM_MAX_MIN[i] +1

SWT_SUM_RBM_MAX_MIN[i] =

SWT_THD_MAX_MIN_OLD[i] + SWT_THD_RBM_MAX_MIN[i]

RATIO_CYCNR_RBM_AFL[i] =

SWT_SUM_RBM_MAX_MIN[i] / ID_T_MAX_CYCNR_AFL[i]

if RATIO_CYCNR_RBM_AFL[i] >= C_RATIO_MV_CYCNR_AFL

then CYCNR_RBM_AFL_SUM[i] = CYCNR_RBM_AFL_SUM[i] +1

LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 0

SWT_THD_RBM_MAX_MIN[i] = 0

RATIO_CYCNR_RBM_AFL[i] = 0

endif

else SWT_THD_RBM_MAX_MIN[i] = 0

SWT_SUM_RBM_MAX_MIN[i] = 0

RATIO_CYCNR_RBM_AFL[i] = 0

endif

B.26.1.3 Switching time from lean to rich

Description:

While the oxygen sensor signal VLS_UP[i] is smaller than VLS_THD_MIN[i], the counter SWT_THD_MAX_MIN[i] is set to 0. The logical variable LV_DIAG_SWT_LS_UP_AFL_CDN[i] ensures that VLS_UP[i] has been smaller than VLS_THD_MIN[i] before switching time determination when set. When the signal exceeds the threshold VLS_THD_MIN[i], the counter SWT_THD_MAX_MIN[i] is started. While VLS_UP[i] < VLS_THD_MAX[i], the counter is incremented with each sensor sampling cycle. The counter SWT_THD_RBM_MAX_MIN[i] is incremented until the ratio RATIO_CYCNR_RBM_AFR[i] exceeds the limit for error detection.

If the oxygen sensor signal exceeds the threshold VLS_THD_MAX[i], the counter is stopped.

If a „lean peak“ which falls below $VLS\_UP[i]_{n-1}$ - C_VLS_HYS occurs while the switching time is being determined, the switching time determination is aborted.

Formula section:

If LV_DIAG_SWT_LS_UP_AFR_ENA[i] = 0

then **if** VLS_UP[i] < VLS_THD_MIN[i]

then LV_DIAG_SWT_LS_UP_AFL_CDN[i] = 1

endifif LV_DIAG_SWT_LS_UP_AFL_CDN[i] = 1

then **If** VLS_THD_MIN[i] < VLS_UP[i]

then **if** $VLS\_UP[i]_n > VLS\_UP[i]_{n-1}$ - C_VLS_HYS

then **if** VLS_UP[i] < VLS_THD_MAX[i]

then SWT_THD_MAX_MIN[i] = SWT_THD_MAX_MIN[i] + 1

```
                  else  stop switching time determination
                        calculation of average switching time lean/rich
                        LV_DIAG_SWT_LS_UP_AFL_CDN[i] = 0
                        LV_DIAG_SWT_LS_UP_AFR_ENA[i] = 1
                        if    LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 0
                        and   LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 0
                        and   (CYCNR_RBM_AFR_SUM[i] =
                                        CYCNR_RBM_AFL_SUM[i])
                        then  LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 1
                              SWT_THD_MAX_MIN_OLD[i] =
                                        SWT_THD_MAX_MIN[i]
                        endif
                  endif
            else  stop switching time determination
                  LV_DIAG_SWT_LS_UP_AFL_CDN[i] = 0
            endif
      else  SWT_THD_MAX_MIN[i] = 0
      endif
   endif
endif
if      LV_DIAG_SWT_LS_UP_RBM_AFL[i] = 0
and     LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 1
then    SWT_THD_RBM_MAX_MIN[i] = SWT_THD_RBM_MAX_MIN[i] +1
        SWT_SUM_RBM_MAX_MIN[i] =
                  SWT_THD_MAX_MIN_OLD[i] + SWT_THD_RBM_MAX_MIN[i]
        RATIO_CYCNR_RBM_AFR[i] =
                SWT_SUM_RBM_MAX_MIN[i] / ID_T_MAX_CYCNR_AFR[i]
        if    RATIO_CYCNR_RBM_AFR[i] >= C_RATIO_MV_CYCNR_AFR
        then  CYCNR_RBM_AFR_SUM[i] = CYCNR_RBM_AFR_SUM[i] +1
              LV_DIAG_SWT_LS_UP_RBM_AFR[i] = 0
              SWT_THD_RBM_MAX_MIN[i] = 0
              RATIO_CYCNR_RBM_AFR[i] = 0
        endif
else    SWT_THD_RBM_MAX_MIN[i] = 0
        SWT_SUM_RBM_MAX_MIN[i] = 0
        RATIO_CYCNR_RBM_AFR[i] = 0
```

endif

B.26.1.4 Averaging of sensor switching times

Description:

Since the sensor switching times fluctuate considerably, the valid sensor switching times are averaged during the check cycle.

In addition to this, the limit for the switching time is read from a map for each valid oxygen sensor switching operation.

Formula section:

```
If      LV_END_DIAG_SWT_LS_UP[i] = 0
then    if      determination of switching time rich/lean was OK
        then    RATIO_CYCNR_AFL[i] =
                SWT_THD_MAX_MIN[i] / ID_T_MAX_CYCNR_AFL
                CYCNR_AFL_SUM[i]  = CYCNR_AFL_SUM[i]  + 1
                RATIO_SUM_CYCNR_AFL[i] =
                RATIO_SUM_CYCNR_AFL[i] + RATIO_CYCNR_AFL[i]
                T_SUM_SWI_AFR_AFL[i] =
                T_SUM_SWI_AFR_AFL[i] + SWT_THD_MAX_MIN[i]
                T_SUM_MAX_SWI_AFR_AFL[i] = T_SUM_MAX_SWI_AFR_AFL[i] +
                ID_T_MAX_CYCNR_AFL
        endif
        If      determination of switching time lean/rich was OK
        then    RATIO_CYCNR_AFR[i] =
                SWT_THD_MAX_MIN[i] / ID_T_MAX_CYCNR_AFR
                CYCNR_AFR_SUM[i]  = CYCNR_AFR_SUM[i]  + 1
                RATIO_SUM_CYCNR_AFR[i] =
                RATIO_SUM_CYCNR_AFR[i] + RATIO_CYCNR_AFR[i]
                T_SUM_SWI_AFL_AFR[i] =
                T_SUM_SWI_AFL_AFR[i] + SWT_THD_MAX_MIN[i]
                T_SUM_MAX_SWI_AFL_AFR[i] = T_SUM_MAX_SWI_AFL_AFR[i] +
                ID_T_MAX_CYCNR_AFR
        endif
endif
```

B.26.1.5 Error detection for switching time diagnosis

Description:

The oxygen sensor switching times are ok when the switching time ratio from rich to lean as well as the switching time ratio from lean to rich are smaller than the corresponding limits. At the end of a complete diagnostic cycle the ratio values are evaluated and LV_END_DIAG_SWT_LS_UP[i] is set to 1. That means, that the diagnosis is stopped for this driving cycle.

Formula section:

```
If      LV_END_DIAG_SWT_LS_UP[i] = 0
then    If      CYCNR_AFL_SUM[i]   > C_CYCNR_SUM_MIN
        and     CYCNR_AFR_SUM[i]   > C_CYCNR_SUM_MIN
        then    RATIO_MV_CYCNR_AFL[i] =
                RATIO_SUM_CYCNR_AFL[i] / CYCNR_AFL_SUM[i]
                RATIO_MV_CYCNR_AFR[i] =
                RATIO_SUM_CYCNR_AFR[i] / CYCNR_AFR_SUM[i]
                LV_END_DIAG_SWT_LS_UP[i] = 1
                evaluation for scantool communication
        endif
        if      RATIO_MV_CYCNR_AFR[i] < C_RATIO_MV_CYCNR_AFR
        and     RATIO_MV_CYCNR_AFL[i] < C_RATIO_MV_CYCNR_AFL
        and     C_RATIO_MIN_DIF < (RATIO_MV_CYCNR_AFL[i] -
                RATIO_MV_CYCNR_AFR[i] ) < C_RATIO_MAX_DIF
        then    LV_ERR_SWT_LS_UP[i] = 0
                ERR_SYM_SWT_LS_UP[i] = "NO_SYM"
        else    LV_ERR_SWT_LS_UP[i] = 1
                ERR_SYM_SWT_LS_UP[i] = "sym_3"
        endif
else    if      LV_ERR_SWT_LS_UP[i] = 1
        then    ERR_SYM_SWT_LS_UP[i] = "sym_3"
        endif
endif
If      CYCNR_RBM_AFL_SUM[i] > C_CYCNR_SUM_MIN
and     CYCNR_RBM_AFR_SUM[i]  > C_CYCNR_SUM_MIN
or      LV_ERR_SWT_LS_UP[i] = 1
then    LV_END_DIAG_RBM_SWT_LS_UP[i] = 1
else    LV_END_DIAG_RBM_SWT_LS_UP[i] = 0
```

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_T_MAX_CYCNR_AFL	1*6	0...FFH	0...2.55	0.01	s
LDPM_MAF_KGH_2_EGCP	6	0...FFFFH	0...2047.98	0.03125	kg/h
max. time interval for switching time monitoring (rich to lean)					
ID_T_MAX_CYCNR_AFR	1*6	0...FFH	0...2.55	0.01	s
LDPM_MAF_KGH_2_EGCP	6	0...FFFFH	0...2047.98	0.03125	kg/h
max. time interval for switching time monitoring (lean to rich)					
C_VLS_HYS	1	0...3FFH	0...4.995117	4.8828E-3	V
max. allowed sensor output peak while switching					
C_CYCNR_SUM_MIN	1	0...FFFFH	0...65535	1	-
min. number diagnostic cycles for switching time monitoring					
C_RATIO_MV_CYCNR_AFL	1	0...7FH	0...1.984375	1.56*10-2	-
limit for total ratio between measured and max. allowed switching times from rich to lean					
C_RATIO_MV_CYCNR_AFR	1	0...7FH	0...1.984375	1.56*10-2	-
max. limit for total ratio between measured and max. allowed switching times from lean to rich					
C_RATIO_MAX_DIF	1	80...7FH	-2...1.984375	1.56*10-2	-
max. difference between lean and rich switching times ratio					
C_RATIO_MIN_DIF	1	80...7FH	-2...1.984375	1.56*10-2	-
min. difference between lean and rich switching times ratio					

B.26.1.6 Evaluation for Scantool SAE 1979 communication:

Description:

The following calculations of output values for Scan Tool SAE 1979 Mode $05 (i = 1) take place only at the end of a complete diagnostic-cycle (**LV_END_DIAG_SWT_LS_UP[i] = 1**) .

Formula section:

$03, Low sensor voltage for switch time calculation:

VLS_SAVE_THD_MIN[i] = VLS_THD_MIN[i]

$04, High sensor voltage for switch time calculation:

VLS_SAVE_THD_MAX[i] = VLS_THD_MAX[i]

$05, Rich to Lean switch time (limited to 1.02 sec as this is the maximum required value of the scantool):

If (T_SUM_SWI_AFR_AFL[i] / CYCNR_AFR_SUM) > 66h

then T_SAVE_SWI_AFR_AFL[i] = 66h

else T_SAVE_SWI_AFR_AFL[i] =
T_SUM_SWI_AFR_AFL[i] / CYCNR_AFR_SUM

endif

If (T_SUM_MAX_SWI_AFR_AFL[i] / CYCNR_AFR_SUM) > 66h

then T_MAX_SAVE_SWI_AFR_AFL[i] = 66h

else T_MAX_SAVE_SWI_AFR_AFL[i] =

T_SUM_MAX_SWI_AFR_AFL[i] / CYCNR_AFR_SUM

endif

$06, Lean to Rich switch time:

If (T_SUM_SWI_AFL_AFR[i] / CYCNR_AFL_SUM[i]) > 66h

then T_SAVE_SWI_AFL_AFR[i] = 66h

else T_SAVE_SWI_AFL_AFR[i] =

T_SUM_SWI_AFL_AFR[i] / CYCNR_AFL_SUM[i]

endif

If (T_SUM_MAX_SWI_AFL_AFR[i] / CYCNR_AFL_SUM[i]) > 66h

then T_MAX_SAVE_SWI_AFL_AFR[i] = 66h

else T_MAX_SAVE_SWI_AFL_AFR[i] =

T_SUM_MAX_SWI_AFL_AFR[i] / CYCNR_AFL_SUM[i]

endif

The following calculations of output values for Scan Tool SAE 1979 Mode $06 (i = 1) take place only at the end of a complete diagnostic-cycle(**LV_END_DIAG_SWT_LS_UP[i] = 1**) .

RATIO_MV_SAVE_CYCNR_AFR[i] = RATIO_MV_CYCNR_AFR[i]

RATIO_MV_SAVE_CYCNR_AFL[i] = RATIO_MV_CYCNR_AFL[i]

Recommended P- Code:

The generic P-code is described as follows.

Diagnostic XX	Symptom description	Symptom	Filter type
SWT_LS_UP[NC_CBK_EX_NR]	Switching time too long	SYM_3	NO

B.26.2 Oxygen sensor control frequency check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
status diagnostic result oxygen sensor frequency monitoring					
ERR_SYM_FRQ_LS_UP[NC_CBK_EX_NR]	V/O	0H 8H	NO_SYM Sym_3	1	-
Detected failure of oxygen sensor frequency monitoring					
LV_END_DIAG_FRQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
diagnostic cycle for frequency check complete					
LV_END_DIAG_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
diagnostic cycle for RBM frequency check complete					
LV_CDN_DIAG_FRQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
status conditions of frequency diagnosis of lambda sensor upstream					
LV_DIAG_FRQ_LS_UP_RBM_AFR[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
status conditions of calculation of rich limit dwell time					
LV_DIAG_FRQ_LS_UP_RBM_AFL[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
status conditions of calculation of lean limit dwell time					
LV_FRQ_DIAG_REQ_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Request of multiple diagnostic of frequency check					
LV_FRQ_DIAG_SYM_LS_UP[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean flag indicating that fault is present before rendering result to error manager					
CTR_CYC_SUM[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured complete controller periods					
CTR_CYC_SUM_RBM_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured limit rich periods					
CTR_CYC_SUM_RBM_AFL[NC_CBK_EX_NR]	V	0...FFFFH	0...65535	1	-
number of measured limit lean periods					
CTR_FRQ_VLD_DIAG_MPL_LS_UP[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
number of valid frequency checks for multiple check					
RATIO_CYC_AFL[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio between measured and max. allowed lean time					
RATIO_CYC_RBM_AFL[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
Stored value of ratio between measured and max. allowed lean time					
RATIO_CYC_RBM_AFL_SUM[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
Calculated limit ratio for lean air fuel phase					
RATIO_CYC_AFR[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
ratio between measured and max. allowed rich time					
RATIO_CYC_RBM_AFR[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
Stored value of ratio between measured and max. allowed rich time					
RATIO_CYC_RBM_AFR_SUM[NC_CBK_EX_NR]	V	0...7FH	0...1.984375	1.56*10-2	-
Calculated limit ratio for rich air fuel phase					
RATIO_SUM_CYC_AFL[NC_CBK_EX_NR]	V	0...FFFFH	0...1023.984375	1.56*10-2	-
total value of lean time					
RATIO_SUM_CYC_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...1023.984375	1.56*10-2	-

total value of rich time					
RATIO_MV_CYC_AFL[NC_CBK_EX_NR]	V/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed lean time					
RATIO_MV_CYC_AFR[NC_CBK_EX_NR]	V/O	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed rich time, saved for SAE Mode $6 output					
RATIO_MV_SAVE_CYC_AFL[NC_CBK_EX_NR]	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed lean time, saved for SAE Mode $6 output					
RATIO_MV_SAVE_CYC_AFR[NC_CBK_EX_NR]	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
total ratio between measured and max. allowed rich time, saved for SAE Mode $6 output					
T_SUM_CYC_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of rich time					
T_SUM_MAX_CYC_AFR[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
sum of max rich time					
T_SAVE_CYC_AFR[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...655.35	0.01	s
rich dwell time for scantool					
T_MAX_SAVE_CYC_AFR[NC_CBK_EX_NR]	V/O/S	0...FFFFH	0...655.35	0.01	s
max. rich dwell time for scantool					
T_AFR_CYC_RBM[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
Additional rich time for limit ratio calculation					
T_AFL_CYC_RBM[NC_CBK_EX_NR]	V	0...FFFFH	0...655.35	0.01	s
Additional lean time for limit ratio calculation					
LV_DIAG_FRQ_CYC_AFR_VLD[NC_CBK_EX_NR]	V	0...1H	0...1	1	-
Boolean variable indicating that last lambda controller period was rich phase					

Input data:

MAF	T_AFR_CYC[NC_CBK_EX_NR]	T_AFL_CYC[NC_CBK_EX_NR]	N_32
T_DLY_NEG_LAM[NC_CBK_EX_NR]	T_DLY_POS_LAM[NC_CBK_EX_NR]	T_DLY_LAM_P_OLD[NC_CBK_EX_NR]	T_DLY_BAS_POS_LAM[NC_CBK_EX_NR]
LV_AFL[NC_CBK_EX_NR]	LV_CDN_DIAG_MPL_LS_UP[NC_CBK_EX_NR]	LV_O2L_LAM_REQ_CAT_DIAG[NC_CBK_EX_NR]	LV_DC

FUNCTION DESCRIPTION:

General Information:

The check cycle can be started following an oxygen controller lambda transition.

The diagnosis threshold C_CTR_CYC_SUM_MIN must be applied such that the checking algorithm for the oxygen sensors upstream from the catalyst is terminated before the catalyst efficiency algorithm. This is done in order to permit the backup trip recognition for catalyst efficiency monitoring to be set in time in the event of an oxygen sensor error.

When the oxygen sensor control frequency check has been completed rendering a result, the function indicates to the error storage mechanism that it is ready for the corresponding engine operating cycle. If an error has been registered within the corresponding engine operating cycle which disables the oxygen sensor control frequency check, the checking algorithm for this engine operating cycle is aborted without rendering a result. If an error has been debounced for the oxygen sensor control frequency check, catalyst efficiency monitoring must be aborted for the corresponding cylinder group in the corresponding engine operating cycle.
In order to ensure that a lambda probe malfunction could have been detected for Rate Based Monitoring additional timer conditions have been implemented. Each time the lambda voltage

switches from rich to lean the timer T_AFR_CYC_RBM[i] is incremented until the ratio RATIO_CYC_RBM_AFR_SUM[i] exceeds the failure criterion. The same holds of course for the transition from lean to rich. If the required number of diagnostic cycles has been exceeded in this way the "RBM end bit" is set. This procedure does not lead to a prolonged diagnosis time.

Description:

The rich T_AFR_CYC[i] and lean T_AFL_CYC[i] dwell times and the check cycle counter CTR_CYC_SUM[i] are evaluated to check the oxygen sensor control frequency. These are also used to determine the catalyst efficiency.
In addition to this, the limits for the maximum rich and lean dwell times are derived from a map for each valid oxygen controller cycle.

Application conditions:

Initialisation: At LV_IGK = 0 -> 1, reset or by clearing the error memory shall all variables be initialised to 0 except LV_END_DIAG_xx and the variables saved in the non-volatile memory

LV_END_DIAG_xx is initialised with 0 at every LV_DC = 0 ->1 or reset or at clearing the error memory

Recurrence:

This function shall be carried out every 10ms.

Activation / Deactivation:

If LV_CDN_DIAG_MPL_LS_UP[i] = 1
and LV_AFL[i] *changed at least one time*
and **(** LV_END_DIAG_RBM_FRQ_LS_UP[i] = 0
or LV_FRQ_DIAG_REQ_LS_UP[i] = 1 **)**
and **(**LC_O2L_LAM_REQ_CAT_DIAG = 0
or LV_O2L_LAM_REQ_CAT_DIAG = 0**)**
then */Frequency check diagnosis active*
LV_CDN_DIAG_FRQ_LS_UP[i] = 1
else */Frequency check diagnosis passive*
LV_CDN_DIAG_FRQ_LS_UP[i] = 0
LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 0
LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 0
endif

Initialisation: *additional to initialisation conditions of chapter 1*

If LV_FRQ_DIAG_REQ_LS_UP[i] = 1

and CTR_CYC_SUM[i] > C_CTR_CYC_SUM_MIN

then reset of all variables and bits, **but not :**

LV_FRQ_DIAG_REQ_LS_UP[i]

CTR_FRQ_VLD_DIAG_MPL_LS_UP[i]

LV_FRQ_DIAG_SYM_LS_UP[i]

SAVE variables saved in non-volatile memory

endif

B.26.2.1 Dwell time calculation

Description:

The ***T_DLY_LAM_ADJ[i]*** value must be the value witch was active at the last p-jump, because this value was used for the p-jump delay T_DLY_POS_LAM[i] or T_DLY_NEG_LAM[i]. See Calculation of T_DLY_LAM_P_OLD[i] below.

Formula section:

Rich dwell time

If LV_AFL[i] = 0 ⇒ 1

and LV_DIAG_FRQ_CYC_AFR_VLD[i] = 0

then LV_DIAG_FRQ_CYC_AFR_VLD[i] = 1

If (T_DLY_POS_LAM[i] > 0)

then RATIO_CYC_AFR[i] = (T_AFR_CYC[i] -

T_DLY_LAM_P_OLD[i] * C_T_DLY_LAM_DIAG_LS_UP) /

IP_T_MAX_CYC_AFR

(consideration of delay time from dynamic fuel trim)

else RATIO_CYC_AFR[i] = (T_AFR_CYC[i] +

T_DLY_BAS_POS_LAM[i] * C_T_DLY_LAM_DIAG_LS_UP) /

IP_T_MAX_CYC_AFR

endif

if LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 0

and LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 0

and CTR_CYC_SUM_RBM_AFR[i] = CTR_CYC_SUM_RBM_AFL[i]

then LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 1

RATIO_CYC_RBM_AFR[i] = RATIO_CYC_AFR[i]

endif

endif

Lean dwell time

If LV_AFL[i] = 1 ⇒ 0

and LV_DIAG_FRQ_CYC_AFR_VLD[i] = 1

then LV_DIAG_FRQ_CYC_AFR_VLD[i] = 0

CTR_CYC_SUM[i] = CTR_CYC_SUM[i] + 1 until CTR_CYC_SUM[i] = C_CTR_CYC_SUM_MIN + 1

If T_DLY_POS_LAM[i] = 0

and T_DLY_NEG_LAM[i] > 0

and *T_DLY_LAM_P_OLD[i]* < 0

then RATIO_CYC_AFL[i] = (T_AFL_CYC[i] +

[***T_DLY_LAM_P_OLD[i]*** + T_DLY_BAS_POS_LAM[i]] *

C_T_DLY_LAM_DIAG_LS_UP) /

IP_T_MAX_CYC_AFL

(consideration of delay time from dynamic fuel trim)

else RATIO_CYC_AFL[i] = T_AFL_CYC[i] / IP_T_MAX_CYC_AFL

endif

if LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 0

and LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 0

and CTR_CYC_SUM_RBM_AFR[i] > CTR_CYC_SUM_RBM_AFL[i]

then LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 1

RATIO_CYC_RBM_AFL[i] = RATIO_CYC_AFL[i]

endif

endif

Calculations for Rate Based Monitoring

Rich dwell time

If LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 1

and LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 0

then *increment* T_AFR_CYC_RBM[i]

RATIO_CYC_RBM_AFR_SUM[i] = RATIO_CYC_RBM_AFR[i] +

(T_AFR_CYC_RBM[i] + C_T_DLY_LAM_ADD_AFR) / IP_T_MAX_CYC_AFR

If RATIO_CYC_RBM_AFR_SUM[i] >= C_RATIO_MAX_VLS_AFR_MV

then CTR_CYC_SUM_RBM_AFR[i] = CTR_CYC_SUM_RBM_AFR[i] +1

LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 0

RATIO_CYC_RBM_AFR_SUM[i] = 0

T_AFR_CYC_RBM[i] = 0

endif

else T_AFR_CYC_RBM[i] = 0

RATIO_CYC_RBM_AFR_SUM[i] = 0

RATIO_CYC_RBM_AFR[i] = 0

endif

Lean dwell time

If LV_DIAG_FRQ_LS_UP_RBM_AFR[i] = 0

and LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 1

then *increment* T_AFL_CYC_RBM[i]

RATIO_CYC_RBM_AFL_SUM[i] = RATIO_CYC_RBM_AFL[i] +

(T_AFL_CYC_RBM[i] + C_T_DLY_LAM_ADD_AFL) / IP_T_MAX_CYC_AFL

If RATIO_CYC_RBM_AFL_SUM[i] >= C_RATIO_MAX_VLS_AFL_MV

then CTR_CYC_SUM_RBM_AFL[i] = CTR_CYC_SUM_RBM_AFL[i] +1

LV_DIAG_FRQ_LS_UP_RBM_AFL[i] = 0

RATIO_CYC_RBM_AFL_SUM[i] = 0

T_AFL_CYC_RBM[i] = 0

endif

else T_AFL_CYC_RBM[i] = 0

RATIO_CYC_RBM_AFL_SUM[i] = 0

RATIO_CYC_RBM_AFL[i] = 0

endif

B.26.2.2 Calculations at end of AF cycle

Description:

The following calculations take place only at the end of a complete AF cycle until diagnosis end is not reached.

Formula section:

If LV_END_DIAG_FRQ_LS_UP[i] = 0

Then **If** $CTR\_CYC\_SUM[i]_n > CTR\_CYC\_SUM[i]_{n-1}$

then

RATIO_SUM_CYC_AFR[i] =

RATIO_SUM_CYC_AFR[i] + RATIO_CYC_AFR[i]

RATIO_SUM_CYC_AFL[i] =

RATIO_SUM_CYC_AFL[i] + RATIO_CYC_AFL[i]

T_SUM_CYC_AFR[i] = T_SUM_CYC_AFR[i] + T_AFR_CYC[i]

T_SUM_MAX_CYC_AFR[i] =

T_SUM_MAX_CYC_AFR[i] + IP_T_MAX_CYC_AFR

RATIO_MV_CYC_AFR[i] =

RATIO_SUM_CYC_AFR[i] / CTR_CYC_SUM[i]

RATIO_MV_CYC_AFL[i] =

RATIO_SUM_CYC_AFL[i] / CTR_CYC_SUM[i]

endif

endif

B.26.2.3 Error detection of oxygen sensor frequency check

Description:

The oxygen sensor is ok when the oxygen controller dwell period measured in the rich or lean range is below the corresponding limit.

An error is detected if one or both of these two times exceed(s) the corresponding limit, an error is entered in the error memory and the diagnostic result ERR_SYM_FRQ_LS_UP[i] and is set. LV_ERR_FRQ_LS_UP[i] is set high when multiple check has been finished and error was present **at all checks**.

Formula section:

If LV_END_DIAG_FRQ_LS_UP[i] = 0

Then **If** CTR_CYC_SUM[i] > C_CTR_CYC_SUM_MIN

then *increment* CTR_FRQ_VLD_DIAG_MPL_LS_UP[i]

RATIO_MV_SAVE_CYC_AFR[i] = RATIO_MV_CYC_AFR[i]

RATIO_MV_SAVE_CYC_AFL[i] = RATIO_MV_CYC_AFL[i]

If RATIO_MV_CYC_AFR[i] < C_RATIO_MAX_VLS_AFR_MV

and RATIO_MV_CYC_AFL[i] < C_RATIO_MAX_VLS_AFL_MV

and C_RATIO_MIN_CYC_AFR <

(RATIO_MV_CYC_AFL[i] - RATIO_MV_CYC_AFR[i])

and C_RATIO_MAX_CYC_AFR >

(RATIO_MV_CYC_AFL[i] - RATIO_MV_CYC_AFR[i])

then LV_FRQ_DIAG_SYM_LS_UP[i] = 0

LV_END_DIAG_FRQ_LS_UP[i] = 1

LV_FRQ_DIAG_REQ_LS_UP[i] = 0

else LV_FRQ_DIAG_SYM_LS_UP[i] = 1

If CTR_FRQ_VLD_DIAG_MPL_LS_UP[i] ≥

C_CTR_FRQ_VLD_DIAG_MPL_LS_UP

then LV_FRQ_DIAG_REQ_LS_UP[i] = 0

(multiple diagnosis finished and error detected)

```
                LV_END_DIAG_FRQ_LS_UP[i] = 1
          else  LV_FRQ_DIAG_REQ_LS_UP[i] = 1
          ´     (multiple diagnosis for O2 sensor is still requested)
          endif
      endif

    endif
endif
If    LV_FRQ_DIAG_SYM_LS_UP[i] = 1
then        ERR_SYM_FRQ_LS_UP[i] = "sym_3"
            If      LV_END_DIAG_FRQ_LS_UP[i] = 1
            then    LV_ERR_FRQ_LS_UP[i] = 1
            endif
else        ERR_SYM_FRQ_LS_UP[i] = "NO_SYM"
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_T_MAX_CYC_AFL	6*6	0...3FFH	0...10.23	0.01	s
LDPM_N_32_4_EGCP	6	0...FFH	0...8160	32	rpm
LDPM_MAF_2_EGCP	6	0...FFFFH	0...1389	0.0212	mg/stk
max. lean time interval for frequency monitoring					
IP_T_MAX_CYC_AFR	6*6	0...3FFH	0...10.23	0.01	s
LDPM_N_32_4_EGCP	6	0...FFH	0...8160	32	rpm
LDPM_MAF_2_EGCP	6	0...FFFFH	0...1389	0.0212	mg/stk
max. rich time interval for frequency monitoring					
C_CTR_CYC_SUM_MIN	1	0...FFFFH	0...65535	1	-
Min. total diagnostic cycles for frequency monitoring					
C_RATIO_MAX_VLS_AFL_MV	1	0...7FH	0...1.984375	1.56*10-2	-
max. limit for total ratio between measured and max. allowed lean time					
C_RATIO_MAX_VLS_AFR_MV	1	0...7FH	0...1.984375	1.56*10-2	-
max. limit for total ratio between measured and max. allowed rich time					
C_RATIO_MAX_CYC_AFR	1	80...7FH	-2...1.984375	1.56*10-2	-
max. difference between lean and rich time ratio					
C_RATIO_MIN_CYC_AFR	1	80...7FH	-2...1.984375	1.56*10-2	-
min. difference between lean and rich time ratio					
C_T_DLY_LAM_DIAG_LS_UP	1	0...FFH	0...1.992	7.8E-3	-
weighting factor for considering adaptation value of downstream fuel trim					
C_T_DLY_LAM_ADD_AFR	1	0...FFFFH	-655.35...655.35	0.02	s
Additional delay time for calculation of limit rich ratio					
C_T_DLY_LAM_ADD_AFL	1	0...FFFFH	-655.35...655.35	0.02	s
Additional delay time for calculation of limit lean ratio					
C_CTR_FRQ_VLD_DIAG_MPL_LS_UP	1	0...FFH	0...255	1	-
Threshold for multiple check cycle counter for O2 sensor diagnosis					
LC_O2L_LAM_REQ_CAT_DIAG	1	0...1H	0...1	1	-
Switch to indicate that an additional O2L is used during CAT_DIAG preventing parallel FRQ_DIAG.					

B.26.2.4 Scantool SAE 1979 Mode $05:

Description:

The following calculations of output values for Scan Tool SAE 1979 Mode $05 (i = 1) take place only at the end of a complete diagnostic-cycle(**LV_END_DIAG_FRQ_LS_UP[i] = 1**) .

Formula section:

If LV_END_DIAG_FRQ_LS_UP[i] = 0 ⇒ 1

then T_SAVE_CYC_AFR[i] = T_SUM_CYC_AFR[i] / CTR_CYC_SUM[i]

T_MAX_SAVE_CYC_AFR[i] =

T_SUM_MAX_CYC_AFR[i] / CTR_CYC_SUM[i]

Endif

B.26.2.5 Calculation of end bit for Rate Based Monitoring:

Description:

The end bit for Rate Based Monitoring shall be calculated either if both RBM cycle counters exceed the minimum required number of diagnostic cycles or as soon as frequency error has been detected.

Formula section:

If CTR_CYC_SUM_RBM_AFR[i] > C_CTR_CYC_SUM_MIN

and CTR_CYC_SUM_RBM_AFL[i] > C_CTR_CYC_SUM_MIN

and LV_END_DIAG_FRQ_LS_UP[i] = 1

or LV_ERR_FRQ_LS_UP[i] = 1

then LV_END_DIAG_RBM_FRQ_LS_UP[i] = 1

else LV_END_DIAG_RBM_FRQ_LS_UP[i] = 0

endif

Recommended P- Code:

The generic P-code is described as follows.

Diagnostic XX	Symptom description	Symptom	Filter type
FRQ_LS_UP[NC_CBK_EX_NR]	Frequency too small	SYM_3	NO

B.26.3 Calculation of T_DLY_LAM_P_OLD[i]

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DLY_LAM_P_OLD[NC_CBK_EX_NR]	V/O	8000...7FFF	-640.02...639.98	0.0195	ms
T_DLY_LAM_ADJ[i] value witch was active at the last p-jump					
T_DLY_LAM_P[NC_CBK_EX_NR]	V	8000...7FFF	-640.02...639.98	0.0195	ms
The T_DLY_LAM_ADJ[i] value of actual p-jump					

Input data:

T_DLY_LAM_ADJ[NC_CBK_EX_NR]	LV_LAM_LSCL[NC_CBK_EX_NR]	LV_AFL[NC_CBK_EX_NR]	

Description:

In order to observe the ***T_DLY_LAM_ADJ[i]*** value, witch was active at the last p-jump; T_DLY_LAM_P_OLD[i] is noted and transferred from last recurrence.

Application conditions:

Initialisation:

At LV_IGK = 0 -> 1, reset or by clearing the error memory shall all variables be initialised to 0 excepted the variables saved in the non-volatile memory.

Recurrence:

This function shall be carried out every 10ms.

Activation:

If the frequency check diagnosis is activated.

Formula section:

If LV_LAM_LSCL[i]$_n$ = 1
and LV_LAM_LSCL[i]$_{n-1}$ = 1
then **If** LV_AFL[i] = 0 $\Rightarrow$ 1 *or* LV_AFL[i] = 1 $\Rightarrow$ 0
then T_DLY_LAM_P_OLD[i] = T_DLY_LAM_P[i]
T_DLY_LAM_P[i] = T_DLY_LAM_ADJ[i]
endif
else **If** LV_AFL[i] = 0 $\Rightarrow$ 1 *or* LV_AFL[i] = 1 $\Rightarrow$ 0
then T_DLY_LAM_P_OLD[i] = T_DLY_LAM_ADJ[i]
T_DLY_LAM_P[i] = T_DLY_LAM_ADJ[i]
endif
endif

B.27 Application incidence for binary oxygen sensor diagnosis

B.27.1 Definition of diagnosis inhibition

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Inhibit upstream O2 sensor Frequency and Switching time diagnosis					
LV_INH_DIAG_RBM_MPL_LS_UP[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Bit indicating inhibition of Rate Based Monitoring of upstream O2 sens. Frequency and Switching time diagnosis					

Input data:

LV_ST_END	LV_IGK	LV_CDN_VB_OBD2	LV_ERR_TPS
LV_ERR_RATIO_CHK	LV_ERR_TCO	LV_ERR_MAF	LV_ERR_CRK
LV_ERR_CAM	LV_ERR_IV[NC_CYL_NR]	LV_ERR_CPS	LV_ERR_MEC_CPS
LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]	LV_ERR_STK_LS_UP[NC_CBK_EX_NR]	LV_ERR_FSD_i	LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]
LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_IS	LV_MIS_STATE_B4	LV_LAM_LSCL[NC_CBK_EX_NR]
LV_ERR_IGC		LV_ERR_MEC_IVVT_IN_i	LV_ERR_LS_UP[NC_CBK_EX_NR]
LV_CAT_DIAG_CDN_ACT[NC_CBK_EX_NR]		CONF_LAM	LV_ERR_MAP
LV_ERR_MEC_IVVT_EX_i	LV_ERR_CAM_IN[NC_NR_CAM_CBK]	LV_ERR_CAM_EX[NC_NR_CAM_CBK]	

FUNCTION DESCRIPTION:

Within this function the inhibition bit for the binary oxygen sensor multiple diagnosis is defined.

Description:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank.

Application conditions:

Initialisation: **at LV_IGK = 0-> 1 and at reset all variables are initialized with 0**

Recurrence: **10 ms**

Activation / Deactivation:

If **N_32 > engine speed for leaving start -> LV_ST_END =1**

and **LV_IGK = 1**

then **enable corresponding diagnosis**

else **disable corresponding diagnosis**

endif

Formula section:

Definition of the Inhibit Bit for Rate Based Monitoring:

If	**LV_ERR_TPS = 1**	**(Throttle position sensor error)**
or	**LV_ERR_RATIO_CHK = 1**	**(Throttle position sensor plausibility error)**
or	**LV_ERR_TCO = 1**	**(Coolant temperature sensor error)**
or	**LV_ERR_MAF = 1**	**(Mass Air Flow sensor error)**
or	**LV_ERR_MAP = 1**	**(Manifold air pressure sensor)**
or	**LV_ERR_CRK = 1**	**(Crankshaft sensor error)**
or	**LV_ERR_CAM = 1**	**(Camshaft sensor error)**
or	**LV_ERR_PUC_LS_UP[i] = 1**	**(O2-sensor upstream PUC error)**
or	**LV_ERR_STK_LS_UP[i] = 1**	**(O2-sensor upstream STK error)**
or	**LV_ERR_CPS = 1**	**(canister purge valve error)**
or	**LV_ERR_MEC_CPS = 1**	**(canister purge valve mechanical error)**
or	**LV_ERR_FSD[i] = 1**	**(fuel system diagnosis error)**
or	**LV_ERR_OBD_LSH_UP[i] = 1**	**(upstream O2-heater OBD2 error)**
or	**LV_ERR_LSH_UP[i] = 1**	**(upstream O2-heater error)**
or	**LV_ERR_IGC = 1**	**(ignition coil error)**
or	**LV_ERR_IV[i] = 1**	**(injection valve error)**
or	LV_ERR_MEC_IVVT_IN_i = 1	(crankshaft to inlet camshaft mech. violation)
or	LV_ERR_MEC_IVVT_EX_i = 1	(crankshaft to inlet camshaft mech. violation)
or	LV_ERR_CAM_IN_i = 1	(IVVT position deviation)
or	LV_ERR_CAM_EX_i = 1	(IVVT position deviation)
or	LV_ERR_LS_UP[i] = 1	(upstream O2-sensor error)
or	CONF_LAM = 0	(open loop lambda control configuration)

then **LV_INH_DIAG_RBM_MPL_LS_UP[i] = 1**
(Rate Based Monitoring of OBD II Upstream oxygen sensor diagnosis inhibited)

else **LV_INH_DIAG_RBM_MPL_LS_UP[i] = 0**
(Rate Based Monitoring of OBD II Upstream oxygen sensor diagnosis enabled)

Endif

Definition of the oxygen sensor inhibit bit :

If	**LV_INH_DIAG_RBM_MPL_LS_UP[i] = 1**	
or	**LV_CAT_DIAG_CDN_ACT[i] = 0**	**(catalyst efficiency diagnosis not active)**
or	**LV_IS = 1**	**(idle)**
or	**LV_MIS_STATE_B4 = 1**	**(Misfire CARB B4 detected)**
or	**LV_LAM_LSCL[i] = 0**	**(lambda controller not active)**
or	**LC_LS_UP_DIAG = 0**	**(manual deactivation of multiple check)**
or	**LV_CDN_VB_OBD2 = 0**	

then **LV_INH_DIAG_LS_UP[i] = 1**
(OBD II Upstream oxygen sensor diagnosis inhibited)
else **LV_INH_DIAG_LS_UP[i] = 0**
(OBD II Upstream oxygen sensor diagnosis enabled)
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_LS_UP_DIAG	1	0...1H	0...1	1	-
Activation bit for upstream oxygen sensor diagnosis					

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
Oxygen sensor upstream switch time		SYM_0	*STD*
		SYM_1	
		SYM_2	
SWT_LS_UP_1	Rich/ lean switch time error	SYM_3	

List of Environmental Data to store in Failure Memory: DIAG_INST_ENVD
RATIO_CYC_AFR
VLS_UP_MMV_MIN_1_ENVD
VLS_UP_MMV_MAX_1_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
Oxygen sensor upstream frequency check		SYM_0	*STD*
		SYM_1	
		SYM_2	
FRQ_LS_UP_1	Slow response	SYM_3	

List of Environmental Data to store in Failure Memory: DIAG_INST_ENVD
RATIO_CYC_AFR
VLS_UP_MMV_MIN_1_ENVD
VLS_UP_MMV_MAX_1_ENVD

B.27.2 Interface for Rate Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_SWT_LS_UP[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of SWT_LS_UP monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
STATE_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	O/V	0 ... FFH	0 ... 255	1	-
Interface of FRQ_LS_UP monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for O2 sensor frequency Diagnosis , for application purpose					
RATE_RBM_SWT_LS_UP[NC_CBK_EX_NR]	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for O2 sensor Switching time Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR		NC_CBK_EX_NR
LV_END_DIAG_SWT_LS_UP[NC_CBK_EX_NR]	LV_END_DIAG_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	NC_NR_CAM_CBK	NLC_CAM_IN
CTR_CDN_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	CTR_COMP_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	CTR_CDN_RBM_SWT_LS_UP[NC_CBK_EX_NR]	CTR_COMP_RBM_SWT_LS_UP[NC_CBK_EX_NR]
LV_ERR_TPS	LV_ERR_RATIO_CHK	LV_ERR_TCO	LV_ERR_MAF
LV_ERR_MAP	LV_ERR_CRK	LV_ERR_CAM	LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]
LV_ERR_STK_LS_UP[NC_CBK_EX_NR]	LV_ERR_CPS	LV_ERR_MEC_CPS	LV_ERR_FSD_i
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]	LV_ERR_IGC	LV_ERR_IV[NC_CYL_NR]
LV_ERR_MEC_IVVT_IN_i	LV_ERR_EL_LS_UP[NC_CBK_EX_NR]	LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]
LV_ERR_CHG_LS_UP	LV_ERR_VCC_PVS_2	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the SWT_LS_UP[i] and FRQ_LS_UP[i] monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_SWT_LS_UP[i] and STATE_RBM_FRQ_LS_UP[i] data, respectively.

Within STATE_RBM_SWT_LS_UP[i] and STATE_RBM_FRQ_LS_UP[i], three different information are defined:

- **Conditions for monitoring are met long enough to detect malfunction (bit 0) (no intrusive operation, no short trip)**
- **Monitor disabled because of system malfunction (bit 1) (depending on failure status: pending)**
- **Monitor individual RBM conditions encountered within this DC (bit 2)**

Application conditions:

Initialisation:

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_SWT_LS_UP[i] = 0
bit 0, bit 1 and bit 2 of STATE_RBM_FRQ_LS_UP[i] = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1 of STATE_RBM_SWT_LS_UP[i] = 0
bit 0 and bit 1 of STATE_RBM_FRQ_LS_UP[i] = 0

on failure memory reset :

bit 1 of STATE_RBM_SWT_LS_UP[i]= 0
bit 1 of STATE_RBM_FRQ_LS_UP[i]= 0

Recurrence: **1s**

Activation: ***LV_DC = 1***

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_SWT_LS_UP[i] = 1

bit 2 of STATE_RBM_FRQ_LS_UP[i] = 1

The pending status of the following failures has to be checked only once:

For ETC and non-ETC variants :

Dependence	Error			
NC_CBK_EX_NR	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_TCO_EL
	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_ERR_TCO_PLAUS	LV_ERR_MAF_SCG_OC
	LV_ERR_MAF_SCP	LV_ERR_CRK_SYN	LV_ERR_CRK_PLAUS	LV_ERR_CRK_TOOTH_PER
	LV_ERR_EL_MAP	LV_ERR_CRK_TOOTH	LV_ERR_PER_CAM_IN_i	LV_ERR_PER_CAM_EX_i
	LV_ERR_PLAUS_CAM_IN_i	LV_ERR_PLAUS_CAM_EX_i	LV_ERR_SYN_CAM_IN_i	LV_ERR_SYN_CAM_EX_i
	LV_ERR_REF_CRK_CAM_IN_i	LV_ERR_REF_CRK_CAM_EX_i	LV_ERR_SYN_CRK_CAM_IN_i	LV_ERR_SYN_CRK_CAM_EX_i
	LV_ERR_CPS	LV_ERR_MEC_CPS	LV_ERR_FSD[NC_CBK_EX_NR]	LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]
	LV_ERR_LSH_UP[NC_CBK_EX_NR]			

Additionnal diagnosis to be checked only for ETC variant :

LV_ERR_TPS_VCC	LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2	LV_ERR_TPS_1
LV_ERR_TPS_2			

Additionnal diagnosis to be checked only for ISA variant :

LV_ERR_TPS_EL	LV_ERR_ISA_1	LV_ERR_ISA_2	LV_ERR_VCC_TPS

If(1) **{ CPU optimization at LV_DC 0 → 1 transition }**

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While **bit 1 of STATE_RBM_SWT_LS_UP[i] = 0 do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) **XX has a pending status**

Then(2) **bit 1 of STATE_RBM_SWT_LS_UP[i] = 1**

Endif(2)

Endwhile

bit 1 of STATE_RBM_FRQ_LS_UP[i] = bit 1 of STATE_RBM_SWT_LS_UP[i]

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

<u>Every 1 s :</u>

If **bit 0 of STATE_RBM_SWT_LS_UP[i] = 0**

and **LV_END_DIAG_SWT_LS_UP[i] = 1**

Then **bit 0 of STATE_RBM_SWT_LS_UP[i] = 1**

Endif

If **bit 0 of STATE_RBM_FRQ_LS_UP[i] = 0**

and **LV_END_DIAG_RBM_FRQ_LS_UP[i] = 1**

Then **bit 0 of STATE_RBM_FRQ_LS_UP[i] = 1**

Endif

If **bit 1 of STATE_RBM_SWT_LS_UP[i] = 0**
and **(LV_ERR_TPS = 1**
or LV_ERR_RATIO_CHK = 1
or LV_ERR_TCO = 1
or LV_ERR_MAF = 1
or LV_ERR_MAP = 1
or LV_ERR_CRK = 1
or LV_ERR_CAM = 1
or LV_ERR_PUC_LS_UP[NC_CBK_EX_NR] = 1
or LV_ERR_STK_LS_UP[NC_CBK_EX_NR] = 1
or LV_ERR_CPS = 1
or LV_ERR_MEC_CPS = 1
or LV_ERR_FSD[NC_CBK_EX_NR]
or LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR] = 1
or LV_ERR_LSH_UP[NC_CBK_EX_NR] = 1
or LV_ERR_IGC = 1
or LV_ERR_IV[NC_CYL_NR] = 1
or LV_ERR_MEC_IVVT_IN[i] = 1
or LV_ERR_MEC_IVVT_EX[i] = 1
or LV_ERR_CAM_IN[i] = 1
or LV_ERR_CAM_EX[i] = 1
or LV_ERR_EL_LS_UP[NC_CBK_EX_NR] = 1
or LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR] = 1
or LV_ERR_CHG_LS_UP = 1)

Then **bit 1 of STATE_RBM_SWT_LS_UP[i] = 1**
Endif

If **bit 1 of STATE_RBM_FRQ_LS_UP[i] = 0**
and **(LV_ERR_TPS = 1**
or LV_ERR_RATIO_CHK = 1
or LV_ERR_TCO = 1
or LV_ERR_MAF = 1
or LV_ERR_MAP = 1

or LV_ERR_CRK = 1

or LV_ERR_CAM = 1

or LV_ERR_PUC_LS_UP[NC_CBK_EX_NR] = 1

or LV_ERR_STK_LS_UP[NC_CBK_EX_NR] = 1

or LV_ERR_CPS = 1

or LV_ERR_MEC_CPS = 1

or LV_ERR_FSD[NC_CBK_EX_NR]

or LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR] = 1

or LV_ERR_LSH_UP[NC_CBK_EX_NR] = 1

or LV_ERR_IGC = 1

or LV_ERR_IV[NC_CYL_NR] = 1

or LV_ERR_MEC_IVVT_IN[i] = 1

or LV_ERR_MEC_IVVT_EX[i] = 1

or LV_ERR_CAM_IN[i] = 1

or LV_ERR_CAM_EX[i] = 1

or LV_ERR_EL_LS_UP[NC_CBK_EX_NR] = 1

or LV_ERR_SWT_LS_UP[NC_CBK_EX_NR] = 1

or LV_ERR_CHG_LS_UP = 1)

Then **bit 1 of STATE_RBM_FRQ_LS_UP[i] = 1**

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_FRQ_LS_UP_i = 0
then RATE_RBM_FRQ_LS_UP_i = FFFFH
else RATE_RBM_FRQ_LS_UP_i
= (CTR_COMP_RBM_FRQ_LS_UP_i / CTR_CDN_RBM_FRQ_LS_UP_i)
endif

If CTR_CDN_RBM_SWT_LS_UP_i = 0
then RATE_RBM_SWT_LS_UP_i = FFFFH
else RATE_RBM_SWT_LS_UP_i
= (CTR_COMP_RBM_SWT_LS_UP_i / CTR_CDN_RBM_SWT_LS_UP_i)
endif

B.27.3 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
RATIO_MV_CYC_AFL_MAX_TOT_DC	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
Former / current driving cycle maximum total ratio between measured and Max allowed lean time					
RATIO_MV_CYC_AFR_MAX_TOT_DC	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
Former / current driving cycle maximum total ratio between measured and Max allowed rich time					
RATIO_MV_CYCNR_AFL_MAX_TOT_DC	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
Former / current driving cycle maximum total ratio between measured and Max allowed swiching times from rich to lean					
RATIO_MV_CYCNR_AFR_MAX_TOT_DC	V/O/S	0...7FH	0...1.984375	1.56*10-2	-
Former / current driving cycle maximum total ratio between measured and Max allowed swiching times from lean to rich					

Input data:

RATIO_MV_CYC_AFL[NC_CBK_EX_NR]	RATIO_MV_CYC_AFR[NC_CBK_EX_NR]	RATIO_MV_CYCNR_AFL[NC_CBK_EX_NR]	RATIO_MV_CYCNR_AFR[NC_CBK_EX_NR]
LV_IGK			

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

RATIO_MV_CYC_AFL_MAX_TOT_DC = 0H

RATIO_MV_CYC_AFR_MAX_TOT_DC = 0H

RATIO_MV_CYCNR_AFL_MAX_TOT_DC = 0H

RATIO_MV_CYCNR_AFR_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: 10ms

Activation: LV_IGK = 1

Deactivation: -

Formula section:

If RATIO_MV_CYC_AFL[1] > RATIO_MV_CYC_AFL_MAX_TOT_DC

Then RATIO_MV_CYC_AFL_MAX_TOT_DC = RATIO_MV_CYC_AFL[1]

Endif

If RATIO_MV_CYC_AFR[1] > RATIO_MV_CYC_AFR_MAX_TOT_DC
Then RATIO_MV_CYC_AFR_MAX_TOT_DC = RATIO_MV_CYC_AFR[1]
Endif

If RATIO_MV_CYCNR_AFL[1] > RATIO_MV_CYCNR_AFL_MAX_TOT_DC
Then RATIO_MV_CYCNR_AFL_MAX_TOT_DC = RATIO_MV_CYCNR_AFL[1]
Endif

If RATIO_MV_CYCNR_AFR[1] > RATIO_MV_CYCNR_AFR_MAX_TOT_DC
Then RATIO_MV_CYCNR_AFR_MAX_TOT_DC = RATIO_MV_CYCNR_AFR[1]
Endif

B.28 Catalyst Efficiency Diagnosis (OSC Method) for binary lambda control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LAMB_CAT_DOWN[NC_CBK_EX_NR]	V	0..FFFFH	0....1.99997	3.051758* 10-5	-
Lambda downstream of Cat., linearization from downstream LS signal					
LAMB_CAT_DOWN_SUM[NC_CBK_EX_NR]	V	0...7FFFFFFFH	0...65535.99997	3.051758* 10-5	-
integral of downstream Lambda through 1diagnosis period for calculation of mean value					
LAMB_CAT_DOWN_MV[NC_CBK_EX_NR]	V	0..FFFFH	0....1.99997	3.051758* 10-5	-
downstream lambda mean value for complete diagnosis period					
LAMB_CAT_DOWN_ST_CYC[NC_CBK_EX _NR]	V	0..FFFFH	0....1.99997	3.051758* 10-5	-
LAMB_CAT_DOWN[i] value at start of monitoring cycle					
LAMB_CAT_DOWN_MIN[NC_CBK_EX_NR]	-	0..FFFFH	0....1.99997	3.051758* 10-5	-
Minimum of O2_CAT[NC_CBK_EX_NR] over diagnosis period					
LAMB_CAT_DOWN_MAX[NC_CBK_EX_NR]	-	0..FFFFH	0....1.99997	3.051758* 10-5	-
Maximum of O2_CAT[NC_CBK_EX_NR] over diagnosis period					
FLOW_O2_CAT_DIF[NC_CBK_EX_NR]	V	0...FFFFH	0...2.04441	3.11957*1 0-5	g/s
downstream O2 Flow difference to mean value					
MASS_O2_CAT_DIF_CYC[NC_CBK_EX_N R]	V	0...FFFFH	0...1.30842456	1.996528* 10-5	g
Mass O2 downsteam catalyst to mean value through 1 diagnosis period					
MASS_O2_CAT_DIF_RATIO[NC_CBK_EX_ NR]	V	0... FFFFH	0....1.99997	3.051758* 10-5	-
Weighted MASS_O2_CAT_DIF_CYC[i] value related to O2-load upstream catalyst from lambda controller					
MASS_O2_CAT_DIF_RATIO_SUM[NC_CB K_EX_NR]	V	0...FFFFFFFFH	0...131071,9999 7	3.051758* 10-5	-
Summation of MASS_O2_CAT_DIF_RATIO[i] over all valid monitoring cycles					
CTR_LAMB_CAT_DOWN_SUM[NC_CBK_E X_NR]	V	0...FFFFH	0..65535	1	-
Counter for the summation of LAMB_CAT_DOWN_SUM for each period					
CTR_CAT_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0..65535	1	-
monitoring cycle counter					
CTR_CAT_DIAG_TOT[NC_CBK_EX_NR]	V/S	0 .... FFH	0 ... 255	1	-
Total sum of all completed diagnosis cycles since ECU programming					
STATE_CAT_DIAG[NC_CBK_EX_NR]	V/O	0H 1H 2H 3H 4H 5H	PASSIVE WAIT CYC_LEAN CYC_RICH END RAMP_UP	1	-
State of diagnosis					
EFF_CAT_DIAG[NC_CBK_EX_NR]	V/S/O	0..FFH	0...1.992188	7.8125*10- 3	-
At ended diagnosis final value for catalyst conversion capability					
LV_O2L_LAM_REQ_CAT_DIAG[NC_CBK_E X_NR]	V/O	0...1H	0...1	1	-
Flag to switch lambda controler to realize O2load for OSC Cat. Diagnosis					
LV_LAM_ADJ_CAT_DIAG[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag to switch trim control parameters for OSC Cat. Diagnosis					

CTR_O2L_TRA_CAT_DIAG[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
Counter of half periods of lambda controller in transition of O2load from not active to active Cat. Diagnosis					
CTR_CAT_DIAG_LAMB_OUT[NC_CBK_EX_NR]	V	0...FFH	0...255	1	-
Counter of non valid cycle due to LAMB_CAT_DOWN_MV[i] is not in right range, for switch off Cat Diag					
TEMP_CAT_DIF_DYN_STAT[NC_CBK_EX_NR]	V	0...FFH	-256...254	2	[°C]
Temperature difference between dynamic and stationary catalyst temperature (from cat. temperature model)					
STATE_VLD_CAT_DIAG[NC_CBK_EX_NR]	V	0...3FH	0...63Phys	1	[-]
Status word of catalyst diagnosis cycle validity (sum of separate validity bits), 1H = 1CROSS, 2H = MVDEL, 4H = 2CROSS, 8H = DIFMV, 10H = RANGE, 20H = TEMP					
O2L_SP_CAT_DIAG[NC_CBK_EX_NR]	V/O	0...7FFFH	0...1.3084	3.99305*10-5	g
O2-load setpoint for lambda controller at Cat diag active					
O2L_AFL_LAM_HLD[NC_CBK_EX_NR]	V	0...7FFFH	0...1.3084	3.99305*10-5	g
Stored O2-load from current lean-halfperiod of lambda controller, before reset in lambda controller					
O2L_LAM_BEG[NC_CBK_EX_NR]	V	0...7FFFH	0...1.3084	3.99305*10-5	g
Reached O2-load from lambda controller before switch to Cat diag active in state wait					
O2L_SP_TRA[NC_CBK_EX_NR]	V	0...7FFFH	0...1.3084	3.99305*10-5	g
O2-load setpoint in transition phase non active to active Cat diag in state WAIT					
O2L_DELTA_SP_TRA[NC_CBK_EX_NR]	V	0...7FFFH	0...1.3084	3.99305*10-5	g
Increase of O2-load setpoint for each half period at transition phase non active to active Cat diag					

Input data:

LV_FAC_P_POS_LAM[NC_CBK_EX_NR]	LV_IGK	LV_O2L_LAM_RLS[NC_CBK_EX_NR]
LV_INH_DIAG_CAT_DIAG[NC_CBK_EX_NR]	LV_AT	O2L_LAM[NC_CBK_EX_NR]
LV_CAT_DIAG_CDN_ACT[NC_CBK_EX_NR]	MAF_CYL	O2L_AFL_LAM[NC_CBK_EX_NR]
VLS_SP_LAM_ADJ(NC_CBK_EX_NR)	N_32	LV_MAF_INT_O2L_TRO_THD[NC_CBK_EX_NR]
VLS_DELTA_LAM_ADJ_CAT_DIAG(NC_CBK_EX_NR)	LV_DC	VLS_DOWN[NC_CBK_EX_NR]
LC_AD_CLR_CAT_DIAG	NC_CBK_EX_NR	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]
CTR_SAVE_SWT_LS_DOWN[NC_CBK_EX_NR]	MAF_HB	TEMP_CAT_STAT_MDL[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank
The signals from the control sensor upstream and the associated monitoring sensor downstream from the catalyst are used for catalyst diagnosis.

If a catalyst has good conversion properties, the oscillating lambda signal upstream from the catalyst - generated by the lambda controller (which shall ensure defined O2-load of catalyst) – is smoothed by the ***O****xygen* ***S****torage* ***C****apacity* of the catalyst.

If the conversion provided by the catalyst is low due to aging, poisoning from leaded fuel or misfiring, then the oscillating lambda signal upstream of the catalyst also exist downstream of the catalyst. The lambda oscillation downstream of the catalyst is detected by an oxygen sensor (monitoring sensor).

Every monitored catalyst or combination of catalysts shall be displayed when the HC emission exceeds the applicable emission test threshold.

Application conditions:

Recurrence: T_SAMPLE = 0.02 sec

Activation: LV_IGK transition 0 -> 1 (ignition key on) **OR** LC_AD_CLR_CAT_DIAG transition 0 -> 1 (reset of CAT-diagnosis values)

Deactivation: *LV_IGK = 1 -> 0 (ignition off)*

Initialization:
At activation and clearing CAT-Diagnosis values: reset of all variables and bits except those described below:
CTR_O2L_TRA_CAT_DIAG[i] = **1** %set transition counter at first to 1

IF EEPROM does not work)

THEN % default values are set

EFF_CAT_DIAG[i] = 0

CTR_CAT_DIAG_TOT[i] = 0

ELSE

IF EFF_CAT_DIAG[i] ≠ 0 **OR** CTR_CAT_DIAG_TOT ≠ 0

THEN no initialisation % New DC has been started from PWL and values from previous DC are still available / not yet stored in EEPROM.

ELSE % previously stored value is assigned

EFF_CAT_DIAG[i] = EFF_CAT_DIAG[i] stored in EEPROM

CTR_CAT_DIAG_TOT[i] = CTR_CAT_DIAG_TOT[i] stored in EEPROM

ENDIF

ENDIF

% Catalyst diagnosis should be reactivated not only at new driving cycle, but at clearing CAT-diagnosis values too.

Description:

The diagnosis algorithm is realized with a state machine in order to improve specification understanding and coherence between specification and SW.

Monitoring cycle:(STATE_CAT_DIAG[i]: CYC_LEAN and CYC_RICH)
The first monitoring cycle is started following the rich-lean change (LV_FAC_P_POS_LAM[i] 1->0). A monitoring cycle lasts one complete lambda controller period (2 subsequent lambda richness changes). Several monitoring cycles are needed to finish the diagnosis.

VLS_DOWN signal from monitoring lambda sensor is converted with inverse sensor characteristic to LAMB_CAT_DOWN[i].

In the Wait State before the first cycle after every reactivation (rich-lean change LV_FAC_P_POS_LAM) only LAMB_CAT_DOWN_MV[i] is calculated. LAMB_CAT_DOWN_SUM[i] is obtained by the integration (therefore factor T_SAMPLE used) of the LAMB_CAT_DOWN[i] over a complete monitoring cycle.

LAMB_CAT_DOWN_MV[i] is the mean value of the LAMB_CAT_DOWN[i] - lambda downstream the catalyst.

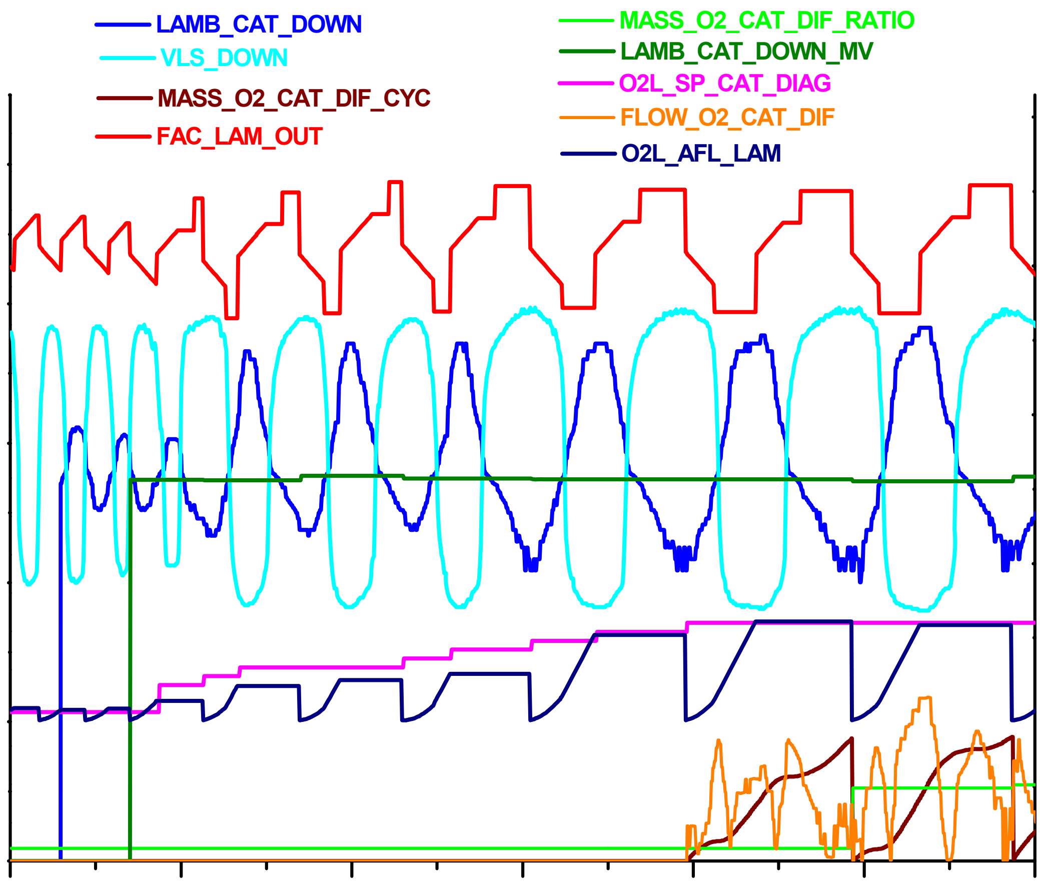
It will be observed, if LAMB_CAT_DOWN_MV[i] is in the right range, than the lambda and trim controller will be switch over to Cat Diag active. Trim control is switched to Cat Diag parameters (P-share to 0, I-share with own calibration and VLS setpoint depends from EFF_CAT_DIAG value)

At activation of catalyst diagnosis the O2-load of catalyst is switched to higher O2L_SP_CAT_DIAG (transition soft switch with ramp up from beginning O2-load (lambda controller switched in not active catalyst diagnosis) to aim O2-load with active cat diag in C_CTR_O2L_TRA_CAT_DIAG cycles so, that the mean value O2-load remains constant).

The calculation of MASS_O2_CAT_DIF_CYC[i] is based on LAMB_CAT_DOWN_MV[i] of the previous monitoring cycle. MASS_O2_CAT_DIF_CYC[i] is the integrated difference between current LAMB_CAT_DOWN[i] signal and LAMB_CAT_DOWN_MV[i] of the previous monitoring cycle multiplied with MAF_CYL so it means the O2 mass comming out of catalyst during 1 diagnosis period.

MASS_O2_CAT_DIF_CYC[i] is related to product of lambda controller (to O2-load of catalyst – comming into catalyst) and weighted with factor IP_FAC_MASS_O2_CAT_DIAG_xx .. to normalize the influence of working point. Result is MASS_O2_CAT_DIF_RATIO[i] - the diagnosis value of one monitoring cycle.

Illustration monitoring cycle



Validity check: (STATE_CAT_DIAG_i: Transition CYC_RICH -> CYC_LEAN)
The obtained MASS_O2_CAT_DIF_CYC[i] of a monitoring cycle is only used for the calculation of the diagnosis value if the validity check was passed (*STATE_VLD_CAT_DIAG_i reach the minimum value for validity*)

To pass the validity check it is necessary that:

- LAMB_CAT_DOWN[i] must cross at least once LAMB_CAT_DOWN_MV[i] of the previous monitoring: 0.bit of STATE_VLD_CAT_DIAG_i is set.

OR

- The deviation between LAMB_CAT_DOWN[i] and LAMB_CAT_DOWN_MV[i] of the last monitoring cycle is smaller than C_LAMB_CAT_MV_DELTA_MAX: 1.bit of STATE_VLD_CAT_DIAG_i is set.

OR

- LAMB_CAT_DOWN[i] cross two times LAMB_CAT_DOWN_MV[i] of the previous monitoring (periodically braking down of VLS_DOWN occurs): 2.bit of STATE_VLD_CAT_DIAG_i is set.

AND

- Difference between actual and the LAMB_CAT_DOWN_MV[i] of the last monitoring cycle is not too high (if LAMB_CAT_DOWN[i] cross two times LAMB_CAT_DOWN_MV[i], higher difference allowed): 3.bit of STATE_VLD_CAT_DIAG_i is set.

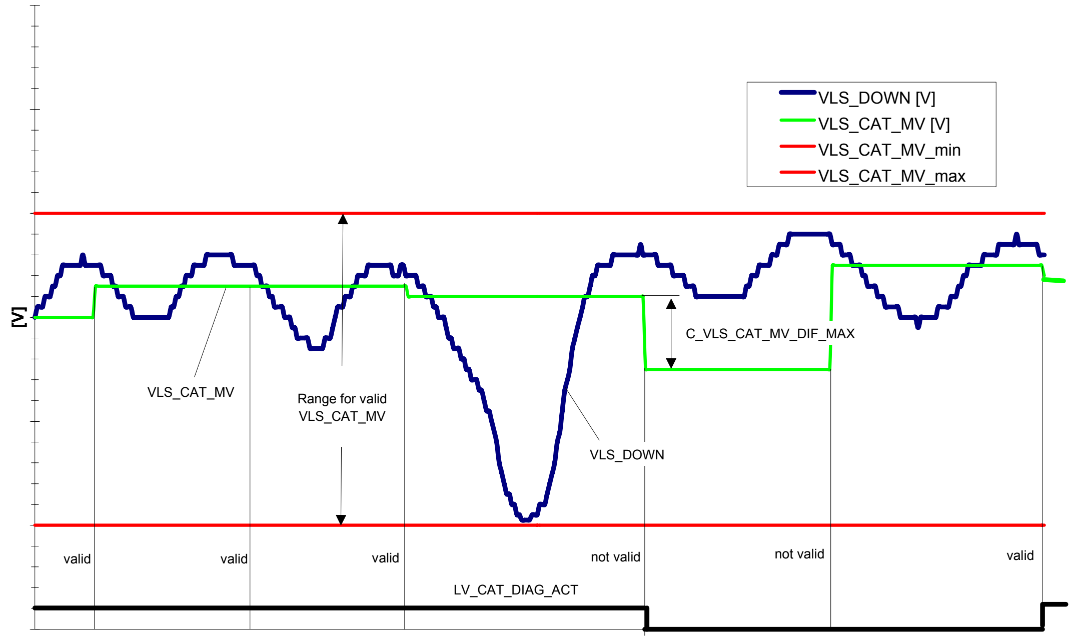
AND

- LAMB_CAT_DOWN_MV[i] is in the right range 4.bit of STATE_VLD_CAT_DIAG_i is set.

AND

- Catalyst temperature dynamic is limited: 5.bit of STATE_VLD_CAT_DIAG_i is set.

Illustration validity check



Calculation of diagnosis value: (STATE_CAT_DIAG[i]: Transition CYC_RICH -> CYC_LEAN)

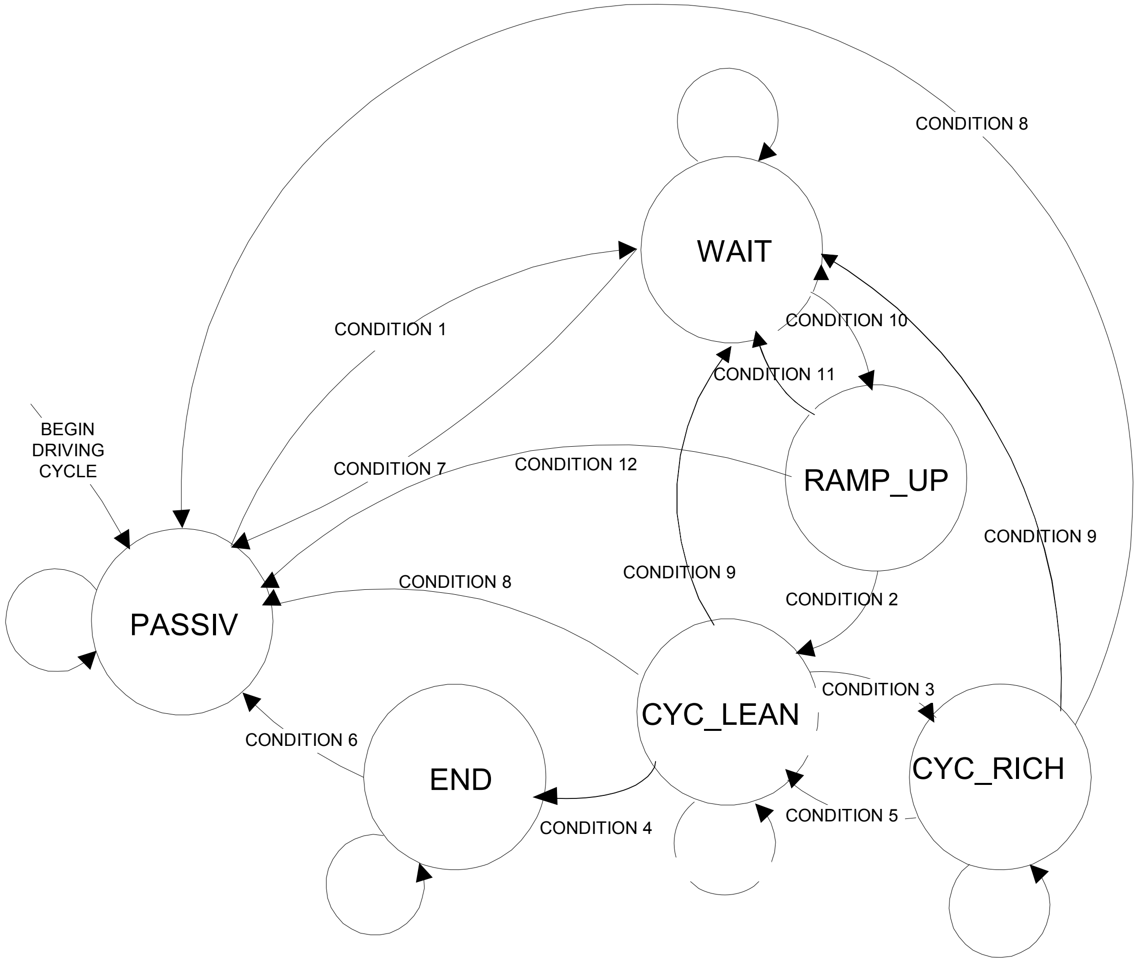
Monitoring cycle counter CTR_CAT_DIAG[i] is incremented for each valid monitoring cycle.

MASS_O2_CAT_DIF_CYC[i] is weighted by engine speed and load (IP_FAC_MASS_O2_CAT_DIAG_XX). From that calculation MASS_O2_CAT_DIF_RATIO[i] results. MASS_O2_CAT_DIF_RATIO[i] is summed up for all monitoring cycles. That summation give MASS_O2_CAT_DIF_RATIO_SUM[i]. Finally, the average of MASS_O2_CAT_DIF_RATIO_SUM[i] over all valid monitoring cycles give the diagnosis value EFF_CAT_DIAG[i]

End of diagnosis: (STATE_CAT_DIAG[i]: Transition CYC_LEAN -> END)

If the necessary number of valid monitoring cycles CTR_CAT_DIAG[i] reaches C_CTR_MIN_CAT_DIAG the diagnosis is ended for that driving cycle and the catalyst diagnosis value EFF_CAT_DIAG[i] is calculated and stored in the EEPROM.
The monitoring counter CTR_CAT_DIAG[i] is used to set the readiness code. For this reason C_CTR_MIN_CAT_DIAG must be reached in the emission test cycle.

State diagram: STATE_CAT_DIAG[i]



Note: The priorities of the conditions to change between states is defined by order described below.

B.28.1 Diagnosis algorithm

STATE_CAT_DIAG[i] "PASSIVE"

Actions:

Condition 1: *"PASSIVE to WAIT"*

LV_CAT_DIAG_CDN_ACT[i] = 1 **AND** LV_INH_DIAG_CAT_DIAG[i] = 0 **AND**

N_32 < C_N_32_CAT_DIAG_DEAC

% To save ECU performance the diagnosis is only active if the engine speed N_32 is
% lower than C_N_32_CAT_DIAG_DEAC.

Transition actions:

STATE_CAT_DIAG[i] = WAIT

TMP_CYC_WAIT[i] = 0 % the first following cycle in WAIT have no current

% LAMB_CAT_DOWN_SUM for mean value calculation

LAMB_CAT_DOWN_SUM[i] = 0

LAMB_CAT_DOWN_MV[i] = 0

STATE_CAT_DIAG[i]: "WAIT"

Actions:
VLS_TEMP = VLS_DOWN[i]

LAMB_CAT_DOWN[i] = IP_LAMB_CAT(VLS_TEMP) % linearization of VLS_DOWN to lambda

1IF LV_FAC_P_POS_LAM[i] 1 -> 0 % transition rich->lean;**monitoring cycle could start**,

1THEN

2IF TMP_CYC_WAIT[i] = 1% second wait cycle with valid LAMB_CAT_DOWN_SUM

2THEN LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] / CTR_LAMB_CAT_DOWN_SUM[i]

2ELSE TMP_CYC_WAIT[i] = 1 % follow. cyc. with valid LAMB_CAT_DOWN_SUM

2END % second wait cycle with valid O2_CAT_SUM?

LAMB_CAT_DOWN_SUM[i] = 0 % new sum after start monitoring cycle

CTR_LAMB_CAT_DOWN_SUM[i] = 0 % reset counter after start monitoring cycle

1END % transition rich -> lean, start of monitoring cycle?

LAMB_CAT_DOWN_SUM[i] (n+1) = LAMB_CAT_DOWN_SUM[i] (n) +

LAMB_CAT_DOWN[i]

CTR_LAMB_CAT_DOWN_SUM[i] +=1 %increment counter

Condition 7: *"WAIT to PASSIVE"*

LV_CAT_DIAG_CDN_ACT[i] = 0 **OR** LV_INH_DIAG_CAT_DIAG[i] = 1 **OR**

N_32 > C_N_32_CAT_DIAG_DEAC

Transition actions:

STATE_CAT_DIAG[i] = PASSIVE

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0 % lambda and trim control param.

LV_LAM_ADJ_CAT_DIAG[i] = 0 %'without CAT_DIAG'

STATE_VLD_CAT_DIAG[i] = 0

Condition 10: *"WAIT to RAMP_UP"*

1IF LV_FAC_P_POS_LAM[i] 1 -> 0 & TMP_CYC_WAIT[i] = 1

% transition rich->lean;**monitoring cycle could start**,

1THEN VLS_TEMP = VLS_SP_LAM_ADJ[i]

TEMP_CAT_DIF_DYN_STAT[i] = TEMP_CAT_DYN_MDL[i] - TEMP_CAT_STAT_MDL[i]

LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] /
CTR_LAMB_CAT_DOWN_SUM[i]

2IF (| LAMB_CAT_DOWN_MV[i] – IP_LAMB_CAT(VLS_TEMP) |) ≤ C_LAMB_CAT_DELTA_MAX

2THEN STATE_VLD_CAT_DIAG[i] = bitwise or RANGE: 01 0000

2ELSE STATE_VLD_CAT_DIAG[i] = bitwise and RANGE=0: 10 1111

2END

3IF I TEMP_CAT_DIF_DYN_STAT[i] **I** <= C_TEMP_CAT_DIF_DYN_ST

3THEN STATE_VLD_CAT_DIAG[i] = bitwise or TEMP: 10 0000

3ELSE STATE_VLD_CAT_DIAG[i] = bitwise and TEMP=0: 01 1111

3END

4IF (STATE_VLD_CAT_DIAG[i] bitwise AND 11 0000) = 110000 % logical enquiry

4THEN RAMP_UP

4END

1END

Transition actions:

STATE_CAT_DIAG[i] = RAMP_UP

LV_LAM_ADJ_CAT_DIAG[i] = 1

LV_O2L_LAM_REQ_CAT_DIAG[i] = 1

CTR_CAT_DIAG_LAMB_OUT[i] = 0 % counter of non valid cycle reset

O2L_LAM_BEG[i] = O2L_LAM[i] %store of beginnig O2load

O2L_DELTA_SP_TRA[i] = IP_O2L_SP_CAT_DIAG – O2L_LAM_BEG[i]
%difference for O2load ramping up

O2L_SP_TRA[i] = O2L_LAM_BEG[i] + O2L_DELTA_SP_TRA[i] /
C_CTR_O2L_TRA_CAT_DIAG

% only for first calculation step at O2load ramping up in tranmsition from WAIT

IF C_CTR_O2L_TRA_CAT_DIAG > 2

THEN O2L_SP_CAT_DIAG[i] = O2L_SP_TRA[i]

ELSE O2L_SP_CAT_DIAG[i] = 0

ENDIF

CTR_O2L_TRA_CAT_DIAG[i] = **2**% set ramping-up counter for **next** calculation

STATE_CAT_DIAG[i]: "RAMP_UP"

Actions:
VLS_TEMP = VLS_DOWN[i]

LAMB_CAT_DOWN[i] = IP_LAMB_CAT(VLS_TEMP) % linearization of VLS_DOWN to lambda

IF LV_O2L_LAM_RLS = 0

THEN LV_LAM_ADJ_CAT_DIAG = 0

ELSE LV_LAM_ADJ_CAT_DIAG = 1

END

1IF LV_FAC_P_POS_LAM[i] 1 -> 0 % transition rich->lean;**monitoring cycle could start**,

1THEN

LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] / CTR_LAMB_CAT_DOWN_SUM[i]

VLS_TEMP = VLS_SP_LAM_ADJ[i] – (VLS_DELTA_LAM_ADJ_CAT_DIAG[i] * CTR_O2L_TRA_CAT_DIAG[i] / C_CTR_O2L_TRA_CAT_DIAG)

2IF (│LAMB_CAT_DOWN_MV[i] – IP_LAMB_CAT(VLS_TEMP) │) < C_LAMB_CAT_DELTA_MAX %LAMB_CAT_DOWN_MV[i] is in the right range

2THEN CTR_CAT_DIAG_LAMB_OUT[i]=0 %reset non valid cycle counter

STATE_VLD_CAT_DIAG[i] = bitwise or RANGE: 01 0000

O2L_DELTA_SP_TRA[i] = IP_O2L_SP_CAT_DIAG – O2L_LAM_BEG[i]

O2L_SP_TRA[i] = O2L_LAM_BEG[i] + (O2L_DELTA_SP_TRA[i] * CTR_O2L_TRA_CAT_DIAG[i]) / C_CTR_O2L_TRA_CAT_DIAG

IF C_CTR_O2L_TRA_CAT_DIAG > 2

THEN O2L_SP_CAT_DIAG[i] = O2L_SP_TRA[i]

ELSE O2L_SP_CAT_DIAG[i] = 0

ENDIF

IF LV_O2L_LAM_RLS[i] = 1 **THEN** CTR_O2L_TRA_CAT_DIAG[i] += 1

%increment counter for next calculation cycle

2ELSE % LAMB_CAT_DOWN_MV[i] is not in the right range

CTR_CAT_DIAG_LAMB_OUT[i] += 1 % increment counter

STATE_VLD_CAT_DIAG_i = bitwise and RANGE=0: 10 1111

2END % LAMB_CAT_DOWN_MV[i] is? in the right range

LAMB_CAT_DOWN_SUM[i] = 0 % new sum after start monitoring cycle

CTR_LAMB_CAT_DOWN_SUM[i] = 0 % reset counter after start monitoring cycle

1ELSE

IF LV_FAC_P_POS_LAM[i] 0 -> 1 **AND** LV_O2L_LAM_RLS[i] = 1% transition lean -> rich; start of rich halfperiod at released O2load from lambda controller

THEN

IF CTR_CAT_DIAG_LAMB_OUT[i]=0

THEN

O2L_DELTA_SP_TRA[i] = IP_O2L_SP_CAT_DIAG – O2L_LAM_BEG[i]

O2L_SP_TRA[i] = O2L_LAM_BEG[i] + (O2L_DELTA_SP_TRA[i]

*CTR_O2L_TRA_CAT_DIAG[i]) / C_CTR_O2L_TRA_CAT_DIAG

IF C_CTR_O2L_TRA_CAT_DIAG > 2

THEN O2L_SP_CAT_DIAG[i] = O2L_SP_TRA[i]

ELSE O2L_SP_CAT_DIAG[i] = 0

ENDIF

CTR_O2L_TRA_CAT_DIAG[i] += 1 %increment counter

END

END

1END

LAMB_CAT_DOWN_SUM[i] (n+1) = LAMB_CAT_DOWN_SUM[i] (n) +

LAMB_CAT_DOWN[i]

CTR_LAMB_CAT_DOWN_SUM[i] +=1 %increment counter

Condition 12: *“RAMP_UP to PASSIVE” %ramp up cycle interrupted because of activation conditions not fullfilled*

LV_CAT_DIAG_CDN_ACT[i] = 0 **OR** LV_INH_DIAG_CAT_DIAG[i] = 1 **OR**

N_32 > C_N_32_CAT_DIAG_DEAC

Transition actions:

STATE_CAT_DIAG[i] = PASSIVE

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0 % lambda and trim control param.

LV_LAM_ADJ_CAT_DIAG[i] = 0 %‘without CAT_DIAG’

STATE_VLD_CAT_DIAG[i] = 0

O2L_SP_CAT_DIAG[i] = 0

Condition 11 :“RAMP_UP to WAIT” % monitoring cycle interrupted, O2-load not realized

(LV_O2L_LAM_RLS[i] = 0 **AND** LV_MAF_INT_O2L_TRO_THD[i] = 1) **OR**

CTR_CAT_DIAG_LAMB_OUT[i] > C_CTR_CAT_DIAG_LAMB_OUT

% counter of non valid cycle due to O2_CAT _MV[i] is not in the right range over threshold

Transition actions:

STATE_CAT_DIAG[i] = WAIT

LV_LAM_ADJ_CAT_DIAG[i] = 0 % trimm control 'without CAT_DIAG'

CTR_O2L_TRA_CAT_DIAG [i] = 0 % reset transition counter of rich/lean crosses

STATE_VLD_CAT_DIAG[i] = 0
LV_O2L_LAM_REQ_CAT_DIAG[i] = 0
O2L_SP_CAT_DIAG[i] = 0

Condition 2:*" RAMP_UP to CYC_LEAN"*

LV_FAC_P_POS_LAM[i] 1 -> 0 **AND** CTR_O2L_TRA_CAT_DIAG[i] >=

C_CTR_O2L_TRA_CAT_DIAG **AND** LV_O2L_LAM_RLS[i] = 1

% transition rich -> lean; transition of lambda controller after switch to 'with CAT_DIAG' ended–counter exceeds C_CTR_O2L_TRA_CAT_DIAG ,**start monitoring cycle**

Transition actions:

STATE_CAT_DIAG[i] = *CYC_LEAN*

O2L_SP_CAT_DIAG[i] = IP_O2L_SP_CAT_DIAG (N_32, MAF_HB)

%setpoint of the O2-load 'with CAT DIAG' for the lambda controller

LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] / CTR_LAMB_CAT_DOWN_SUM[i]

%first the mean value from the sum of previous diag cycle will be calculated, then the sum and the summation counter will be set to 0 for summation of following diagnosis cycle.

LAMB_CAT_DOWN_SUM[i] = 0

CTR_LAMB_CAT_DOWN_SUM[i] = 0

MASS_O2_CAT_DIF_CYC[i] = 0

LAMB_CAT_DOWN_ST_CYC[i] = LAMB_CAT_DOWN[i] %value at start of new monitoring cycle is stored

STATE_VLD_CAT_DIAG[i] = bitwise and 1CROSS=2CROSS=0:111010

STATE_VLD_CAT_DIAG[i] = bitwise or MVdel:000010

% reset of both CROSS bits and set MVdel bit.

LAMB_CAT_DOWN_MAX[i] = LAMB_CAT_DOWN_MIN[i] = LAMB_CAT_DOWN[i]

STATE_CAT_DIAG[i]: "CYC_LEAN"

Actions:

VLS_TEMP = VLS_DOWN[i]

LAMB_CAT_DOWN[i] = IP_LAMB_CAT(VLS_TEMP)

% linearization of VLS_DOWN to lambda down the Catalyst

10IF LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MAX[i]

10THEN LAMB_CAT_DOWN_MAX[i] = LAMB_CAT_DOWN[i]

10ELSE

11IF LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MIN[i]

11THEN LAMB_CAT_DOWN_MIN[i] = LAMB_CAT_DOWN[i]

11END

10END

IF LV_O2L_LAM_RLS = 0

THEN LV_LAM_ADJ_CAT_DIAG = 0

ELSE LV_LAM_ADJ_CAT_DIAG = 1

END

LAMB_CAT_DOWN_SUM[i] (n+1) = LAMB_CAT_DOWN_SUM[i] (n) + LAMB_CAT_DOWN[i]

CTR_LAMB_CAT_DOWN_SUM[i] +=1 %increment counter

FLOW_O2_CAT_DIF[i] = 0.23 * **I**1/LAMB_CAT_DOWN_MV[i] - 1/LAMB_CAT_DOWN[i]**I**

* (MAF_CYL[kg/h] / NC_CBK_EX_NR) * (1000/3600)[g/kg] * [h/s]

% 0.23*2/65536*0.03125*1000/3600*512=3.11957*10-5 [g/s]

MASS_O2_CAT_DIF_CYC[i](n) = MASS_O2_CAT_DIF_CYC[i](n-1) + FLOW_O2_CAT_DIF[i] * T_SAMPLE

% 3.11957*10-5 [g/s]*0.02s*32 = 1.996528*10-5g

%for validity of this diagnosis period LAMB_CAT_DOWN[i] must cross at least once LAMB_CAT_DOWN_MV[i] of the previous monitoring cycle, then STATE_VLD_CAT_DIAG[i] bitwise or 1CROSS: 00 0001.

12IF (STATE_VLD_CAT_DIAG[i] bitwise and 1CROSS: 00 0001)

12THEN

13IF ((LAMB_CAT_DOWN_ST_CYC[i] > LAMB_CAT_DOWN_MV[i]) **AND** LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MV[i])) **OR**

((LAMB_CAT_DOWN_ST_CYC[i] < LAMB_CAT_DOWN_MV[i]) **AND** (LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MV[i]))

13THEN STATE_VLD_CAT_DIAG[i] = bitwise or 2CROSS 00 0100

13END

12ELSE

14IF ((LAMB_CAT_DOWN_ST_CYC[i] > LAMB_CAT_DOWN_MV[i]) **AND** (LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MV[i])) **OR**

((LAMB_CAT_DOWN_ST_CYC[i] < LAMB_CAT_DOWN_MV[i]) **AND**

(LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MV[i]))

14THEN STATE_VLD_CAT_DIAG[i] = bitwise or 1CROSS 00 0001

14END

12END

% or the difference between LAMB_CAT_DOWN[i] and LAMB_CAT_DOWN_MV[i] of the last monitoring cycle is limited.

IF C_LAMB_CAT_MV_DELTA_MAX < **I** LAMB_CAT_DOWN[i] - LAMB_CAT_DOWN_MV[i] **I**

THEN STATE_VLD_CAT_DIAG[i] = bitwise and MVdel = 0: 11 1101

END

Condition 4: "CYC_LEAN to END"

CTR_CAT_DIAG[i] >= C_CTR_MIN_CAT_DIAG

Transition actions:

EFF_CAT_DIAG[i] = MASS_O2_CAT_DIF_RATIO_SUM[i] / CTR_CAT_DIAG[i]

* IP_FAC_EFF_CAT_SWT_COR (CTR_SAVE_SWT_LS_DOWN[i])

%3.051758*10-5*256 = 7.8125*10-3

CTR_CAT_DIAG_TOT[i] = CTR_CAT_DIAG_TOT[i] + 1

STATE_CAT_DIAG[i] = END % diagnosis finished for that driving-cycle

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0 % lambda and trim control param.

LV_LAM_ADJ_CAT_DIAG[i] = 0 'withoutCATDIAG'

STATE_VLD_CAT_DIAG[i] = 0

O2L_SP_CAT_DIAG[i] = 0

Condition 8 :"CYC_LEAN to PASSIVE" % monitoring cycle interrupted

LV_CAT_DIAG_CDN_ACT[i] = 0 **OR** LV_INH_DIAG_CAT_DIAG[i] = 1

OR

N_32 > C_N_32_CAT_DIAG_DEAC **OR**

CTR_CAT_DIAG_LAMB_OUT[i] > C_CTR_CAT_DIAG_LAMB_OUT

% counter of non valid cycle due to O2_CAT _MV[i] is not in the right range over threshold

Transition actions:

STATE_CAT_DIAG[i] = PASSIVE

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0 % lambda and trim control param.

LV_LAM_ADJ_CAT_DIAG[i] = 0 ‘withoutCATDIAG’

CTR_O2L_TRA_CAT_DIAG[i] = 1 % reset counter of rich/lean crosses to 1

STATE_VLD_CAT_DIAG[i] = 0

O2L_SP_CAT_DIAG[i] = 0

Condition 9 :“CYC_LEAN to WAIT” % monitoring cycle interrupted, O2-load not realized

LV_O2L_LAM_RLS[i] = 0 **AND** LV_MAF_INT_O2L_TRO_THD[i] = 1

Transition actions:

STATE_CAT_DIAG[i] = WAIT

CTR_O2L_TRA_CAT_DIAG[i] = 1 % set counter of rich/lean crosses at O2-load transition

LV_LAM_ADJ_CAT_DIAG[i] = 0 trim control parameters ‘withoutCATDIAG’

STATE_VLD_CAT_DIAG[i] = 0
O2L_SP_CAT_DIAG[i] = 0

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0

Condition 3:“ CYC_LEAN to CYC_RICH”

LV_FAC_P_POS_LAM[i] 0 -> 1 % transition lean -> rich

Transition actions:

STATE_CAT_DIAG[i] = CYC_RICH
O2L_SP_CAT_DIAG[i] = IP_O2L_SP_CAT_DIAG (N_32, MAF_HB)
O2L_AFL_LAM_HLD[i] = O2L_AFL_LAM[i] %stored O2load before reset in lambda ctrl

STATE_CAT_DIAG[i]: “CYC_RICH”
Actions:

VLS_TEMP = VLS_DOWN[i]

LAMB_CAT_DOWN[i] = IP_LAMB_CAT(VLS_TEMP)

% linearization of VLS_DOWN to lambda down the catalyst

10IF LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MAX[i]

10THEN LAMB_CAT_DOWN_MAX[i] = LAMB_CAT_DOWN[i]

10ELSE

11IF LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MIN[i]

11THEN LAMB_CAT_DOWN_MIN[i] = LAMB_CAT_DOWN[i]

11END

10END

IF LV_O2L_LAM_RLS = 0

THEN LV_LAM_ADJ_CAT_DIAG = 0

ELSE LV_LAM_ADJ_CAT_DIAG = 1

END

LAMB_CAT_DOWN_SUM[i] (n+1) = LAMB_CAT_DOWN_SUM[i] (n) + LAMB_CAT_DOWN[i]

CTR_LAMB_CAT_DOWN_SUM[i] +=1 %increment counter

FLOW_O2_CAT_DIF[i] = 0.23 * **I**1/LAMB_CAT_DOWN_MV[i] - 1/LAMB_CAT_DOWN[i]**I** * (MAF_CYL[kg/h] / NC_CBK_EX_NR) * (1000/3600)[g/kg] * [h/s]

MASS_O2_CAT_DIF_CYC[i](n)= MASS_O2_CAT_DIF_CYC[i](n-1) + FLOW_O2_CAT_DIF[i] * T_SAMPLE

% for validity of this diagnosis period LAMB_CAT_DOWN[i] must cross at least once LAMB_CAT_DOWN_MV[i] of the previous monitoring cycle, then STATE_VLD_CAT_DIAG_i bitwise or 00 0001.

18IF STATE_VLD_CAT_DIAG[i] bitwise and 1CROSS: 00 0001

18THEN

19IF ((LAMB_CAT_DOWN_ST_CYC[i] > LAMB_CAT_DOWN_MV[i]) **AND**

(LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MV[i])) **OR**

((LAMB_CAT_DOWN_ST_CYC[i] < LAMB_CAT_DOWN_MV[i]) **AND**

(LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MV[i]))

19THEN STATE_VLD_CAT_DIAG[i] = bitwise or 2CROSS: 00 0100

19END

18ELSE

20IF ((LAMB_CAT_DOWN_ST_CYC[i] > LAMB_CAT_DOWN_MV[i]) **AND**

(LAMB_CAT_DOWN[i] < LAMB_CAT_DOWN_MV[i])) **OR**

((LAMB_CAT_DOWN_ST_CYC[i] < LAMB_CAT_DOWN_MV[i]) **AND**

(LAMB_CAT_DOWN[i] > LAMB_CAT_DOWN_MV[i]))

20THEN STATE_VLD_CAT_DIAG[i] = bitwise or 1CROSS: 00 0001

20END

18END

% or the difference between LAMB_CAT_DOWN[i] and LAMB_CAT_DOWN_MV[i] of the last monitoring cycle is limited.

IF C_LAMB_CAT_MV_DELTA_MAX < **I**LAMB_CAT_DOWN[i] - LAMB_CAT_DOWN_MV[i]**I**

THEN STATE_VLD_CAT_DIAG_i = bitwise and MVdel:11 1101

END

Condition 8 :“CYC_RICH to PASSIVE” % monitoring cycle interrupted

LV_CAT_DIAG_CDN_ACT[i] = 0 **OR** LV_INH_DIAG_CAT_DIAG[i] = 1

OR

N_32 > C_N_32_CAT_DIAG_DEAC **OR** CTR_CAT_DIAG_LAMB_OUT[i] > C_CTR_CAT_DIAG_LAMB_OUT

% counter of non valid cycle due to O2_CAT _MV[i] is not in the right range over threshold

Transition actions:

STATE_CAT_DIAG[i] = PASSIVE
LV_O2L_LAM_REQ_CAT_DIAG[i] = 0 % lambda and trim control param.

LV_LAM_ADJ_CAT_DIAG[i] = 0 ‘withoutCATDIAG’

CTR_O2L_TRA_CAT_DIAG[i] = 1 % reset counter of rich/lean crosses to 1

STATE_VLD_CAT_DIAG[i] = 0

O2L_SP_CAT_DIAG[i] = 0

Condition 9 :“CYC_RICH to WAIT” % monitoring cycle interrupted, O2-load not realized

LV_O2L_LAM_RLS[i] = 0 **AND** LV_MAF_INT_O2L_TRO_THD[i] = 1

Transition actions:

STATE_CAT_DIAG[i] = WAIT
CTR_O2L_TRA_CAT_DIAG[i] = 1 % reset counter of rich/lean crosses to 1

STATE_VLD_CAT_DIAG[i] = 0

LV_LAM_ADJ_CAT_DIAG[i] = 0

O2L_SP_CAT_DIAG[i] = 0

LV_O2L_LAM_REQ_CAT_DIAG[i] = 0

Condition 5: “CYC_RICH to CYC_LEAN”

LV_FAC_P_POS_LAM[i] 1 -> 0 **% transition rich -> lean; monitoring cycle finished**

Transition actions:

LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] / CTR_LAMB_CAT_DOWN_SUM[i]

%first the mean value from the sum of previous diag cycle will be calculated

VLS_TEMP = VLS_SP_LAM_ADJ[i] – VLS_DELTA_LAM_ADJ_CAT_DIAG[i]

TEMP_CAT_DIF_DYN_STAT[i] = TEMP_CAT_DYN_MDL[i] - TEMP_CAT_STAT_MDL[i]

IF LC_EFF_CAT_DIAG_CLC_LAM = 0

THEN

MASS_O2_CAT_DIF_RATIO[i] = MASS_O2_CAT_DIF_CYC[i] *

IP_FAC_TEMP_CAT_DIF_STAT /

(O2L_AFL_LAM_HLD[i] * IP_FAC_MASS_O2_CAT_DIAG_xx)

%Xx = MT / AT according LV_AT

% 1.996528*10-5g/3.99305*10-5/3.051758*10-5 = 16384/2e14/128/256

= 7.8125*10-3/256 =3.051758*10-5

ELSE

MASS_O2_CAT_DIF_RATIO[i] = MASS_O2_CAT_DIF_CYC[i] *

IP_FAC_TEMP_CAT_DIF_STAT / (O2L_SP_CAT_DIAG [i] *

IP_FAC_MASS_O2_CAT_DIAG_xx)

END

IF (│LAMB_CAT_DOWN_MV[i] – IP_LAMB_CAT(VLS_TEMP) │) < C_LAMB_CAT_DELTA_MAX

%LAMB_CAT_DOWN_MV[i] is in the right range

THEN STATE_VLD_CAT_DIAG[i] = bitwise or RANGE: 01 0000

CTR_CAT_DIAG_LAMB_OUT[i] = 0   % counter of non valid cycle reseted

ELSE STATE_VLD_CAT_DIAG[i] = bitwise and RANGE=0: 10 1111

CTR_CAT_DIAG_LAMB_OUT[i] += 1   % counter of non valid cycle incremented

END

23IF I TEMP_CAT_DIF_DYN_STAT[i] **I** <= C_TEMP_CAT_DIF_DYN_MAX

23THEN STATE_VLD_CAT_DIAG[i] = bitwise or TEMP: 10 0000

23ELSE STATE_VLD_CAT_DIAG[i] = bitwise and TEMP=0: 1101 1111

23END

% Validity check of last monitoring cycle

24IF (C_LAMB_CAT_MV_DIF_MAX +

((LAMB_CAT_DOWN_MAX[i] - LAMB_CAT_DOWN_MIN[i]) /8)

* ((STATE_VLD_CAT_DIAG[i] bitwise and 2CROSS = 00 0100) /4)

>= **I** LAMB_CAT_DOWN_MV[i](n) – LAMB_CAT_DOWN_MV[i] (n-1) **I)**

% result from bitwise or shall be the logical value =1, if the 00 0100 bit was set

24THEN STATE_VLD_CAT_DIAG[i] = bitwise or difMV: 00 1000

24ELSE STATE_VLD_CAT_DIAG[i] = bitwise and difMV=0: 11 0111

24END

1IF LV_O2L_LAM_RLS[i] = 1

1THEN LV_LAM_ADJ_CAT_DIAG[i] = 1

2IF STATE_VLD_CAT_DIAG_i >= 58

AND O2L_AFL_LAM_HLD[i] >= C_O2L_AFL_LAM_THD

% TEMP&RANGE&difMV&(2CROSS or MVdel) & released O2load is not to small

2THEN %Validity check passed. Calculation of diagnosis value

MASS_O2_CAT_DIF_RATIO_SUM[i](n+1) =

MASS_O2_CAT_DIF_RATIO_SUM[i](n) + MASS_O2_CAT_DIF_RATIO[i]

CTR_CAT_DIAG[i](n+1) = CTR_CAT_DIAG[i](n) + 1

% Monitoring cycle counter SUM_CAT_DIAG is incremented

2END

1ELSE

LV_LAM_ADJ_CAT_DIAG[i] = 0

1END

STATE_CAT_DIAG[i] = CYC_LEAN

O2L_SP_CAT_DIAG[i] = IP_O2L_SP_CAT_DIAG (N_32, MAF_HB)

%setpoint of the O2-load 'with CAT DIAG' for the lambda controller

LAMB_CAT_DOWN_SUM[i] = 0 %then the sum will be set to 0 for summation of following diagnosis cycle.

CTR_LAMB_CAT_DOWN_SUM[i] = 0

MASS_O2_CAT_DIF_CYC[i] = 0

STATE_VLD_CAT_DIAG[i] = bitwise and 1CROSS=2CROSS=0:111010

STATE_VLD_CAT_DIAG[i] = bitwise or MVdel:000010

% reset of both CROSS bits and set MVdel bit.

LAMB_CAT_DOWN_MAX[i] = LAMB_CAT_DOWN_MIN[i] = LAMB_CAT_DOWN[i]

LAMB_CAT_DOWN_ST_CYC[i] = LAMB_CAT_DOWN[i] %value at start of new monitoring cycle is stored

STATE_CAT_DIAG[i]: "END"

Actions:

Condition 6:"END to PASSIVE"

LV_DC = 0 -> 1

Transition actions:

STATE_CAT_DIAG[i] = PASSIVE

B.28.2 Application Assistance

For OBD II O2 sensor diagnosis: C_CTR_MIN_CAT_DIAG has to be applied higher then the thresholds C_SUM_CYC_MIN_DIAG and C_SUM_CYCNR_MIN_DIAG (LV_LS_FRQ_DIAG_READY -> 1 and LV_LS_SWT_DIAG_READY -> 1).

They have to be reached before STATE_CAT_DIAG[i] is set to END.

Hint:
An active catalyst diagnosis can disable the module “Downstream Fuel Trim Regulation” also a additional lambda offset can be imposed on the basic lambda setpoint in the module “Basic Lambda setpoint”. When the catalyst diagnosis is completed the above functions return to their normal operation.

B.28.3 Overview: Chronological order of diagnosis algorithm:

Triggering of catalyst diagnosis calculation by p-jump of lambda controller LV_FAC_P_POS_LAM[i] 0->1 and 1->0...

O2L_SP_CAT_DIAG[i] = IP_O2L_SP_CAT_DIAG (N_32, MAF_HB)

%setpoint of the O2-load 'with CAT DIAG' for the lambda controller (output, not used here)

1) LAMB_CAT_DOWN[i] = IP_LAMB_CAT(VLS_DOWN[i]) %every recurrence

2) $LAMB\_CAT\_DOWN\_SUM[i]_{n+1}$ = $LAMB\_CAT\_DOWN\_SUM[i]_{n}$ + LAMB_CAT_DOWN[i]

%every recurrence

3) LAMB_CAT_DOWN_MV[i] = LAMB_CAT_DOWN_SUM[i] / CTR_LAMB_CAT_DOWN_SUM[i]

(after end of current diagnosis period)

4) FLOW_O2_CAT_DIF[i] = 0.23 * (1000/3600)[g/s] * I1/LAMB_CAT_DOWN_MV[i] - 1/LAMB_CAT_DOWN[i] I * (MAF_CYL[kg/h] / NC_CBK_EX_NR)

5) $MASS\_O2\_CAT\_DIF\_CYC[i]_{n+1}$ = $MASS\_O2\_CAT\_DIF\_CYC[i]_{n}$ + FLOW_O2_CAT_DIF[i]*T_SAMPLE[ms] %every recurrence

6) MASS_O2_CAT_DIF_RATIO[i] = MASS_O2_CAT_DIF_CYC[i] * IP_FAC_TEMP_CAT_DIF_STAT / (O2L_AFL_LAM_HLD[i] * IP_FAC_MASS_O2_CAT_DIAG_XX) $\Rightarrow$ with XX for AT or MT (after end of current diagnosis period)

7) $MASS\_O2\_CAT\_DIF\_RATIO\_SUM[i]_{n+1}$ = $MASS\_O2\_CAT\_DIF\_RATIO\_SUM[i]_{n}$ + MASS_O2_CAT_DIF_RATIO[i]

(after end of current diagnosis period and passed validity check)

8) EFF_CAT_DIAG[i] = MASS_O2_CAT_DIF_RATIO_SUM[i] / CTR_CAT_DIAG[i]

(Catalyst diagnosis value after diagnosis finished)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LAMB_CAT_DELTA_MAX	1	0..FFFFH	0....1.99997	3.051758*10-5	-
Max. deviation between LAMB_CAT_MV and setpoint for a valid monitoring cycle					
C_LAMB_CAT_MV_DIF_MAX	1	0..FFFFH	0....1.99997	3.051758*10-5	-
max. allowed O2_CAT_MV fluctuation between 2 subsequent monitoring cycles					
C_LAMB_CAT_MV_DELTA_MAX	1	0..FFFFH	0....1.99997	3.051758*10-5	-
max. allowed fluctuation between O2_CAT_MV and O2_CAT[i] in the same monitoring cycle					
IP_FAC_MASS_O2_CAT_DIAG_AT	8*8	0..FFFFH	0....1.99997	3.051758*10-5	-
LDPM_N_32_1_EGTR	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_EGTR	8	0...FFH	0...1389	5.426	mg/stk
weight factor of downstream sensor signal integral for automatic transmission					
IP_FAC_MASS_O2_CAT_DIAG_MT	8*8	0..FFFFH	0....1.99997	3.051758*10-5	-
LDPM_N_32_1_EGTR	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_EGTR	8	0...FFH	0...1389	5.426	mg/stk
weight factor of downstream sensor signal integral for manual transmission					
C_CTR_MIN_CAT_DIAG	1	0...FFFFH	0...65535	1	-
min. number of required monitoring cycles for catalyst diagnosis					
C_EFF_CAT_DIAG_THD	1	0...FFH	0...1.992188	7.8125*10-3	-
threshold for CAT_DIAG to detect a catalyst with malfunction					
C_N_32_CAT_DIAG_DEAC	1	0 ... FFH	0 ... 8160	32	rpm
Engine speed threshold to deactivate catalyst diagnosis to save ECU performance					
IP_LAMB_CAT	1*12	0..FFFFH	0....1.99997	3.051758*10-5	-
LDP_VLS_TEMP_IP_LAMB_CAT	12	0...3FFH	0...4.995	4.8876*10-3	V
O2_load linearization from downstream LS signal					
C_CTR_CAT_DIAG_LAMB_OUT	1	0…FFH	0...255	1	-
Threshold to switch off Cat Diag due to to O2_CAT _MV is not in the right range					
C_CTR_O2L_TRA_CAT_DIAG	1	0…FFH	0...255	1	-
Threshold of transition cycles after Lambda controller switched to Cat Diag O2-load to evaluate Cat Diag cycle					
IP_O2L_SP_CAT_DIAG	8*8	0...7FFFH	0...1.3084	3.99305*10-5	g
LDPM_N_32_1_EGTR	8	0...FFH	0...8160	32	rpm
LDPM_MAF_HB_1_EGTR	8	0...FFH	0...1389	5.447	mg/stk
O2-load setpoint for lambda controller at Cat diag active					
LC_EFF_CAT_DIAG_CLC_LAM	1	0...1	0...1	1	-
Logical switch to relate Cat.Diag value to realized O2load or to O2load setpoint					
IP_FAC_TEMP_CAT_DIF_STAT	8*8	0...FFH	0...3.984375	0.015625	[-]
LDP_TEMP_CAT_DYN_MDL	8	0...FFH	-273.15...1766.85	8	[°C]
LDP_TEMP_CAT_DIF_DYN_STAT	8	0...FFH	-256..254	2	[°C]
Factor to correct cat.diag. value at catalyst temperature different to stationary temperature in this working point					
C_TEMP_CAT_DIF_DYN_ST	1	0...FFH	-256..254	2	[°C]
Max. Temperature difference for catalyst diagnosis begin					
C_TEMP_CAT_DIF_DYN_MAX	1	0...FFH	-256..254	2	[°C]
Max. Temperature difference for catalyst diagnosis cycle validity					
C_O2L_AFL_LAM_THD	1	0...7FFFH	0...1.30840	0.0399e-3	[g]
Threshold for stored O2-load from current lean-halfperiod of lambda controller, before reset in lambda controller.					
IP_FAC_EFF_CAT_SWT_COR	8	0..FFH	0...3.984375	0.015625	-
LDP_CTR_SAVE_SWT_LS_DOWN	8	0..FFH	0...1.992188	7.84*10-3	-
Down O2 sensor switiching time effect correction for catalyst diagnosis					

B.28.4 Application Incidence for Catalyst Efficiency Diagnosis (OSC Method)

B.28.4.1 Diagnosis inhibition flag

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_INH_DIAG_CAT_DIAG_i	V	0...1H	0...1	1	
Flag to inhibit catalyst diagnosis function if there is a component failure					
LV_CL_CALC_AVL	V/O	0...1H	0...1	1	-
Flag to inhibit Cat. Dianosis in case of Min-Purge (pre-defined to 0)					

Input data:

LV_DC	LV_ERR_CRK	LV_ERR_CAM_IN_i	LV_ERR_MAP
LV_ERR_MAF	LV_ERR_TCO	LV_ERR_TPS	LV_CDN_DIAG_CAT_DIAG_i
LV_ERR_VS	LV_MIS_STATE_B1	LV_MIS_STATE_A	LV_MIS_STATE_B4
LV_ERR_LS_DOWN_[NC_CBK_EX_NR]	LV_ERR_FSD[NC_CBK_EX_NR]	LV_ERR_LS_UP_[NC_CBK_EX_NR]	LV_FAC_P_POS_LAM[NC_CBK_EX_NR]
LV_ERR_AMP		LV_ERR_MEC_IVVT_EX_i	LV_ERR_CPS
LV_ERR_IGC	LV_ERR_IV[NC_CYL_NR]	LV_ERR_MEC_IVVT_IN_i	
LV_CDN_VB_OBD2	LV_ERR_MEC_CPS	LV_ERR_FSD_LAM_LIM_i	LV_ERR_LOAD_TPS_PLAUS
STATE_CP	CL_MMV	LV_ERR_PUC_LS_DOWN	LV_ERR_CAM_EX[NC_NR_CAM_CBK]

FUNCTION DESCRIPTION:

If one of the following components fail the catalyst efficiency diagnosis function is stopped by LV_INH_DIAG_CAT_DIAG_i= 1.

The inhibition of the catalyst diagnosis can be deactivated by LC_INH_DIAG_CAT_DIAG_MAN_DEAC = 1.

For LC_INH_DIAG_CAT_DIAG_MAN_DEAC = 1 the catalyst diagnosis function is not inhibited by a component failure. In that way the diagnosis conditions can be ensured for application reasons.

T_VLS_CYC_EFC_i is defined to have an interface to the OSC cat. diagnosis that fits for different definitions (different physical limit) of the lean / rich dwell time of the AFR.

Application conditions:

Recurrence: 0.02 sec

Activation: LV_DC transition 0 -> 1 (start driving cycle)

Deactivation: LV_DC transition 1 -> 0 (driving cycle terminated)

Formula section:

```
IF
        STATE_CP = MIN_PURGE AND CL_MMV < C_CL_ MAX_CAT_DIAG
        THEN LV_CL_CALC_AVL = 1
        ELSE LV_CL_CALC_AVL = 0
ENDIF

IF      (Inhibition due to OBD-I error)
        LV_ERR_CAM_IN_i = 1
   OR   LV_ERR_CAM_EX_i = 1
   OR   LV_ERR_CRK = 1
   OR   LV_ERR_TCO = 1
   OR   LV_ERR_MAF = 1
   OR   LV_ERR_MAP = 1
   OR   LV_ERR_VS = 1
   OR   LV_ERR_TPS = 1
   OR   LV_ERR_AMP =1
   OR   LV_ERR_CPS = 1
   OR   LV_ERR_IGC = 1
   OR   LV_ERR_IV[NC_CYL_NR] = 1
   OR   LV_ERR_MEC_IVVT_IN_i = 1
   OR   LV_ERR_MEC_IVVT_EX_i = 1
   OR   LV_CDN_VB_OBD2 = 0
   OR   LV_ERR_PUC_LS_DOWN_i = 1

        (Inhibition due to OBD-II error)
   OR   LV_ERR_MEC_CPS = 1

   OR   LV_ERR_FSD_LAM_LIM_i =1

   OR   LV_ERR_LOAD_TPS_PLAUS = 1
   OR   LV_ERR_LS_UP[NC_CBK_EX_NR] = 1
        (summary error bit of upstr. Lambda sensor due to OBD I and OBD II error)

   OR   LV_ERR_LS_DOWN[NC_CBK_EX_NR] = 1
        (summary error bit of downstr. Lambda sensor due to OBD I and OBD II error)

   OR   LV_ERR_FSD[NC_CBK_EX_NR] = 1
   OR   LV_MIS_STATE_A = 1
   OR   LV_MIS_STATE_B1 = 1
   OR   LV_MIS_STATE_B4 = 1

 AND    LC_INH_DIAG_CAT_DIAG_MAN_DEAC = 0
THEN    LV_INH_DIAG_CAT_DIAG_i= 1
ELSE    LV_INH_DIAG_CAT_DIAG_i = 0
```

Remark:
The calculation of lean/rich dwell time is done in the Lambda Control and imported into the Catalyst diagnosis function directly from there.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_INH_DIAG_CAT_DIAG_MAN_DEAC	1	0 ... 1H	0 ... 1	1	-
Manual deactivation of catalyst diagnosis inhibition conditions					
C_CL_ MAX_CAT_DIAG	1	0 ... FFFFH	0 ... 8	0.000122	-
Maximum CL for catalyst diagnosis in STATE_CP = MIN_PURGE.					

Configuration or calibration data

Diagnostic	Symptom Description	Symptom	Filter type
CAT_DIAG_1	CAT damaged	SYM_0	*NO*
		SYM_1	
		SYM_2	
		SYM_3	

B.28.4.2 O2 sensors diagnosis end flag grouping

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_LS_UP_DIAG_END[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	
Flag for Upstream O2 sensor diagnosis (relevant for Catalyst Diagnosis) end status					
LV_LS_DOWN_DIAG_END[NC_CBK_EX_NR]	V/O	0...1H	0...1	1	-
Flag for Downstream O2 sensor diagnosis (relevant for Catalyst Diagnosis) end status					

Input data:

LV_END_DIAG_SWT_LS_UP[NC_CBK_EX_NR]	LV_END_DIAG_FRQ_LS_UP[NC_CBK_EX_NR]		

FUNCTION DESCRIPTION:

As an Upstream O2 sensor failure can cause high Catalyst Diagnosis result (i.e. slow control frequency of upstream O2 sensor), it is checked if all relevant Upstream O2 sensor diagnosis are finished before to decide for Catalyst Diagnosis failure.

As a Downstream O2 sensor failure can lead only to a non-detection of a faulty Catalyst, LV_LS_DOWN_DIAG_ END[NC_CBK_EX_NR] = 1.

Application conditions:

Recurrence: 1 sec

Activation: LV_DC transition 0 -> 1 (start driving cycle)

Deactivation: LV_DC transition 1 -> 0 (driving cycle terminated)

Formula section:

If LV_END_DIAG_SWT_LS_UP[NC_CBK_EX_NR] = 1 **and**
LV_END_DIAG_FRQ_LS_UP[NC_CBK_EX_NR] = 1
Then
LV_LS_UP_DIAG_END[NC_CBK_EX_NR] = 1
Else
LV_LS_UP_DIAG_END[NC_CBK_EX_NR] = 0

LV_LS_DOWN_DIAG_ END[NC_CBK_EX_NR] = 1 always.

B.28.4.3 Interface to Rate-Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_CAT_DIAG[NC_CBK_EX_NR]	O/V	0 ... 7H	0 ... 7H	1	-
Interface of Catalyst Diagnosis with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_CAT_DIAG[NC_CBK_EX_NR]	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Catalyst Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_CAT_DIAG[NC_CBK_EX_NR]	NC_CBK_EX_NR
LV_INH_DIAG_CAT_DIAG_i	CTR_CDN_RBM_CAT_DIAG[NC_CBK_EX_NR]	CTR_COMP_RBM_CAT_DIAG[NC_CBK_EX_NR]	LV_ERR_EL_MAP
LV_ERR_VCC_PVS_2	LV_ERR_MAP_PLAUS		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the Catalyst Diagnosis and the Rate-Based Monitoring statistics is defined with STATE_RBM_CAT_DIAG[NC_CBK_EX_NR].

Within STATE_RBM_CAT_DIAG[NC_CBK_EX_NR], three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation:

at ECU reset :
bit 0, bit 1 and bit 2 of STATE_RBM_ CAT_DIAG[NC_CBK_EX_NR] = 0
at LV_DC 0 → 1 transition :
bit 0 and bit 1 of STATE_RBM_ CAT_DIAG[NC_CBK_EX_NR] = 0
on failure memory reset :
bit 1 of STATE_RBM_ CAT_DIAG[NC_CBK_EX_NR] = 0

Recurrence: 1s

Activation: *LV_DC = 1*

Formula section:

At LV_DC 0 → 1 transition

bit 2 of STATE_RBM_ CAT_DIAG[NC_CBK_EX_NR] = 1

The pending status of the following failures has to be checked only once:

For ETC and non-ETC variants :

LV_ERR_PUC_LS_DOWN	LV_ERR_SYN_CAM_IN_i	LV_ERR_PER_CAM_IN_i	LV_ERR_PLAUS_CAM_IN_i
LV_ERR_REF_CRK_CAM_IN_i	LV_ERR_SYN_CRK_CAM_IN_i	LV_ERR_CRK_SYN	LV_ERR_CRK_TOOTH_PER
LV_ERR_SYN_CAM_EX_i	LV_ERR_PER_CAM_EX_i	LV_ERR_PLAUS_CAM_EX_i	LV_ERR_REF_CRK_CAM_EX_i
LV_ERR_SYN_CRK_CAM_EX_i	LV_ERR_MEC_IVVT_EX_i	LV_ERR_MEC_IVVT_IN_i	
LV_ERR_CRK_TOOTH	LV_ERR_CRK_PLAUS	LV_ERR_EL_MAP	LV_ERR_TCO_EL
LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_ERR_TCO_PLAUS	LV_ERR_MAF_SCP
LV_ERR_MAF_SCG_OC	LV_ERR_VS	LV_ERR_FSD[NC_CBK_EX_NR]	LV_ERR_MIS[NC_CYL_NR]
LV_ERR_SCG_LS_UP[NC_CBK_EX_NR]	LV_ERR_SCP_LS_UP[NC_CBK_EX_NR]	LV_ERR_OC_LS_UP[NC_CBK_EX_NR]	LV_ERR_PUC_LS_UP[NC_CBK_EX_NR]
LV_ERR_STK_LS_UP[NC_CBK_EX_NR]	LV_ERR_SWT_LS_UP[NC_CBK_EX_NR]	LV_ERR_FRQ_LS_UP[NC_CBK_EX_NR]	LV_ERR_LSH_UP[NC_CBK_EX_NR]
LV_ERR_OBD_LSH_UP[NC_CBK_EX_NR]	LV_ERR_LAM_ADJ[NC_CBK_EX_NR	LV_ERR_OBD_LSH_DOWN[NC_CBK_EX_NR]	LV_ERR_LSH_DOWN[NC_CBK_EX_NR]
LV_ERR_SCG_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_SCP_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_OC_LS_DOWN[NC_CBK_EX_NR]	LV_ERR_PUE_LS_DOWN[NC_CBK_EX_NR]
LV_ERR_VCC_PVS_2	LV_ERR_MAP_PLAUS		

Additional diagnosis to be checked only for ETC variant :

LV_ERR_TPS_VCC	LV_ERR_TPS_1	LV_ERR_TPS_2	LV_ERR_TPS_MAF_1
LV_ERR_TPS_MAF_2			

Additional diagnosis to be checked only for non-ETC variant :

LV_ERR_TPS_EL	LV_ERR_ISA_1	LV_ERR_ISA_2	LV_ERR_VCC_TPS

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 0

and LV_END_DIAG_CAT_DIAG[NC_CBK_EX_NR] = 1

Then bit 0 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 1

Endif

If bit 1 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 0

and LV_ERR_CAM_IN_i = 1
or LV_ERR_CAM_EX_i = 1
or LV_ERR_CRK = 1
or LV_ERR_TCO = 1
or LV_ERR_MAF = 1
or LV_ERR_MAP = 1
or LV_ERR_VS = 1
or LV_ERR_TPS = 1
or LV_ERR_AMP =1
or LV_ERR_CPS = 1
or LV_ERR_IGC = 1
or LV_ERR_IV[NC_CYL_NR] = 1
or LV_ERR_MEC_IVVT_IN_i = 1
or LV_ERR_MEC_IVVT_EX_i = 1
or LV_ERR_VB = 1
or LV_ERR_PUC_LS_DOWN_i = 1
or LV_ERR_MEC_CPS = 1
or LV_ERR_FSD_LAM_LIM_i =1
or LV_ERR_LOAD_TPS_PLAUS = 1
or LV_ERR_LS_UP[NC_CBK_EX_NR] = 1
or LV_ERR_LS_DOWN[NC_CBK_EX_NR] = 1

or LV_ERR_FSD[NC_CBK_EX_NR] = 1
or LV_MIS_STATE_A = 1
or LV_MIS_STATE_B1 = 1
or LV_MIS_STATE_B4 = 1

Then bit 1 of STATE_RBM_CAT_DIAG[NC_CBK_EX_NR] = 1

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_CAT_DIAG_i = 0
then RATE_RBM_CAT_DIAG_i = FFFFH
else RATE_RBM_CAT_DIAG_i
= (CTR_COMP_RBM_CAT_DIAG_i / CTR_CDN_RBM_CAT_DIAG_i)
endif

B.29 Catalyst efficiency diagnosis (error interface for one cylinder bank)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_CAT_DIAG_OBD[NC_CBK_EX_NR]	V/O/S	0...FFH	0...1,992188	7.8125*10-3	-
Cat. Diag. value for OBD Scantool Mod 6 output					
ERR_SYM_CAT_DIAG[NC_CBK_EX_NR]	V	0H 1H	NO_SYM SYM_0	1	-
Detected failure of each symptom: failure without filtering of diagnosis value catalyst diagnose					
LV_ERR_CAT_DIAG[NC_CBK_EX_NR]	V/O	0...1H	0 ... 1	1	-
Present failure: failure after filtering diagnosis value catalyst diagnose					
LV_END_DIAG_CAT_DIAG[NC_CBK_EX_NR]	V/O	0...1H	0 ... 1	1	-
Flag catalyst diagnosis finished for that driving cycle					
EFF_CAT_MAX_DIAG_OBD[NC_CBK_EX_NR]	V/O/S	0...FFH	0...1,992188	7.8125*10-3	-
Actually threshold for Scantool Mod 6 Output					
LV_CDN_DIAG_CAT_DIAG[NC_CBK_EX_NR]	V/O	0...1H	0 ... 1	1	-
Flag condition for catalyst diagnosis active					

Input data:

LV_ERR_LS_DOWN[NC_CBK_EX_NR]	LV_LS_DOWN_DIAG_END[NC_CBK_EX_NR]	LV_LS_UP_DIAG_END[NC_CBK_EX_NR]	LV_DC
LV_ERR_LS_UP[NC_CBK_EX_NR]	STATE_CAT_DIAG[NC_CBK_EX_NR]	EFF_CAT_DIAG[NC_CBK_EX_NR]	NC_CBK_EX_NR
CONF_CAT_EFF		LV_CAT_DIAG_REQ_EOL	

FUNCTION DESCRIPTION:

General information:

The function is valid only for one exhaust cylinder bank system.
(NC_CBK_EX_NR = 1) then
i = 1, for exhaust cylinder bank 1

Application conditions:

Initialisation: LV_DC 0 -> 1

By activation at start driving cyclethe OBD output values must be initialised as described below. All other variables must be set to 0.

Initialisation of OBD Variables:

IF (the EEPROM does not work)

THEN EFF_CAT_DIAG_OBD[i] = 0 % default values are set

EFF_CAT_MAX_DIAG_OBD = 0

ELSE

IF EFF_CAT_DIAG_OBD[i] ≠ 0 OR EFF_CAT_MAX_DIAG_OBD ≠ 0

THEN no initialisation % New DC has been started from PWL and values from previous DC are still available / not yet stored in EEPROM.

ELSE EFF_CAT_DIAG_OBD[i] = EFF_CAT_DIAG_OBD[i] stored in EEPROM

EFF_CAT_MAX_DIAG_OBD = EFF_CAT_MAX_DIAG_OBD[i] stored in EEPROM

ENDIF

ENDIF

Recurrence: T_SAMPLE = 0.02 sec

Activation:

IF LV_DC = 1 (start driving cycle)

AND STATE_CAT_DIAG[i] != PASSIVE

AND LV_END_DIAG_CAT_DIAG[i] = 0

AND CONF_CAT_EFF > 0

THEN LV_CDN_DIAG_CAT_DIAG[i] = 1

% function is active

ELSE LV_CDN_DIAG_CAT_DIAG[i] = 0

% function is not active

END

Description:

Formula section:

1IF STATE_CAT_DIAG[i] = END & LV_END_DIAG_CAT_DIAG_i = 0

%cat. diag. of bank i finished

1THEN

2IF LV_CAT_DIAG_REQ_EOL = 1%for EOL Test separate bank calculation

2THEN %EOL

3IF EFF_CAT_DIAG[i] > C_EFF_CAT_MAX_DIAG_EOL

3THEN

ERR_SYM_CAT_DIAG[i] = SYM_0

% not sufficient catalyst conversion efficiency (detected symptom)

LV_ERR_CAT_DIAG[i] = 1 (failure flag)

3END

EFF_CAT_MAX_DIAG_OBD[i] = C_EFF_CAT_MAX_DIAG_EOL %for sctool

EFF_CAT_DIAG_OBD[i] = EFF_CAT_DIAG[i] %for scantool

2ELSE %no EOL

Function call "separate bank calculation"

2END **%EOL?**

Function call “final destincness”

4IF STATE_CAT_DIAG[i] (n-1) ≠ END **AND** STATE_CAT_DIAG[i] = END

4THEN Function call “intermediate save”

4END

1END

Function call: “separate bank calculation”:

IF EFF_CAT_DIAG[i] > C_EFF_CAT_MAX_DIAG

THEN

ERR_SYM_TMP_CAT_DIAG[i] = SYM_0

LV_ERR_TMP_CAT_DIAG[i] = 1

END

%end of function call: “separate bank calculation”

Function call “intermediate save”: Action_ERRM_StorePrevFrf(CAT_DIAG)

Function call “final destincness”:

IF **(**(LV_LS_UP_DIAG_END[i] = 1**AND** LV_ERR_LS_UP[i] = 0) **AND**

(LV_LS_DOWN_DIAG_END[i] = 1**AND** LV_ERR_LS_DOWN[i] = 0)**) OR**

LV_DC = 1 -> 0

THEN LV_END_DIAG_CAT_DIAG[i] = 1

ERR_SYM_ CAT_DIAG[i] = ERR_SYM_TMP_CAT_DIAG[i]

LV_ERR_CAT_DIAG[i] = LV_ERR_TMP_CAT_DIAG[i]

EFF_CAT_MAX_DIAG_OBD[i] = C_EFF_CAT_MAX_DIAG %for scantool

EFF_CAT_DIAG_OBD[i] = EFF_CAT_DIAG[i] %for scantool

END

%**function call for delivery the result to Error Manager:**

clc_fmyabc_mngf_by_diag (cat_diag_1_ERR_IDX , err_sym_tmp, 1);

Configuration for diagnostic symptoms :

Diagnostic EFF_CAT_DIAG[i]	Symptom description	Symptom	Filter type
Catalyst diagnosis	´CAT_EFFIC_LOW´	SYM_0	NO

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_EFF_CAT_MAX_DIAG_EOL	1	0..FFH	0...1.992188	7.8125*10-3	-
max. threshold for EFF_CAT_DIAG to detect a separate catalyst with OBD limit emissions at EOL Test					
C_EFF_CAT_MAX_DIAG	1	0..FFH	0...1.992188	7.8125*10-3	-
max. threshold for EFF_CAT_DIAG to detect a separate catalyst with OBD limit emissions					

B.30 Activation conditions for catalyst efficiency and O2 sensor diagnosis(bin/bin)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CAT_DIAG_CDN_ACT[NC_CBK_EX_NR]	V/O	0 ... 1H	0 ... 1	1	
Activation conditions for O2 sensor and catalyst efficiency diagnosis (OSC-method)					
MAF_INT_DLY_CAT_DIAG[NC_CBK_EX_NR]	V	0...FFFFH	0...1820.42	0.028	g
Air mass flow integral after conditions fullfilled for cat diag activation					

Input data:

AMP_AD	N_32	LV_CL_CALC_AVL	T_DLY_P_LAM_ADJ[NC_CBK_EX_NR]
LV_SAWUP	LV_IGK	LV_LAM_LSCL[NC_CBK_EX_NR]	LV_LAM_STOP[NC_CBK_EX_NR]
STATE_CP	CL_MMV		STATE_LSH_DOWN[NC_CBK_EX_NR]
	TCO		LV_LS_DOWN_READY[NC_CBK_EX_NR]
MAF_CYL	MAF		LV_CAT_PURGE_ACT[NC_CBK_EX_NR]
TEMP_CAT	VS	NC_CBK_EX_NR	LV_LDC_CAT[NC_CBK_EX_NR]

FUNCTION DESCRIPTION:

General information:

The calculation shall be done for all exhaust cylinder banks.
For instance, if two separate catalyst systems are concerned (NC_CBK_EX_NR = 2) then
i = 1, for exhaust cylinder bank 1
i = 2, for exhaust cylinder bank 2
otherwise (NC_CBK_EX_NR = 1)
i = 1, for single exhaust cylinder bank

That function determines the common activation flag LV_CAT_DIAG_CDN_ACT[i] for catalyst efficiency (OSC method) and O2 sensor diagnosis. The common activation flag LV_CAT_DIAG_CDN_ACT[i] can be set manually by LC_CAT_DIAG_CDN_ACT_MAN.

Application conditions:

Recurrence: T_SAMPLE = 0.02 sec

Activation: LV_IGK transition 0 -> 1 (ignition key on) **OR** new activation after Fault memory cleared (CLR_FMY)

Deactivation: LV_IGK = 1 -> 0 (ignition off)

MAF_INT_DLY_CAT_DIAG[i] is initialized by 0 at each activation and at Failure memory reset.

Formula section:

% First group of conditions (except for temp_cat-conditions):
IF **(**TCO > C_TCO_MIN_CAT
%coolant at operation temperature
AND C_VS_MIN_CAT <= VS <= C_VS_MAX_CAT
% vehicle speed must lie in the following range:

AND AMP_AD >= C_AMP_MIN_CAT_LS_DIAG (750mbar)
% ambient pressure large enough

AND LV_LDC_CAT[i] = 1
% limited dynamics conditions must exist

AND IT_DLY_P_LAM_ADJ[i]**I**< C_T_DLY_P_LAMB_MAX_CAT_DIAG**)**
% P-share from trim control will be set to 0 at Cat Diag active, so it couldn't be high in absolute value

THEN TMP_CND_1 = 1

ELSE TMP_CND_1 = 0

ENDIF

% Second group of conditions - after fullfilling will a MAF_CYL integral be started:
IF (LV_LAM_LSCL[i] = 1 **AND** LV_LAM_STOP[i] = 0
% lambda controller must be active (controlling!)

AND
LV_CAT_PURGE_ACT[i] = 0
% catalyst purge function not active

AND ((STATE_CP = NO_PURGE) **OR** (STATE_CP = CP_NOT_ACT)

OR (STATE_CP = WAIT_RAMP_OPEN)

OR ((STATE_CP = MAX_PURGE) **AND** (CL_MMV <= C_CL_MAX_CAT_LS_DIAG)**)**
% canister purge not active or active in max purge state and canister load must be equal or smaller than threshold C_CL_MAX_CAT_LS_DIAG

OR ((STATE_CP = MIN_PURGE) **AND** (LV_CL_CALC_AVL = 1)**)**

%or active in min purge due to low (not unknown) canister load (LV_CL_CALC_AVL)

AND LV_LS_DOWN_READY[i] = 1 **AND** STATE_LSH_DOWN[i] = LSH_POW_CTR
% downstream lambda sensor is ready for operation and is heated

AND LV_SAWUP = 0**)**
% no secondary-air injection active.

THEN TMP_CND_2 = 1

ELSE TMP_CND_2 = 0

ENDIF

OLD_CAT_DIAG_CDN [i] = LV_CAT_DIAG_CDN_ACT[i]

Activation or deactivation

IF TMP_CND_2 = 0
THEN MAF_INT_DLY_CAT_DIAG[i] = 0

LV_CAT_DIAG_CDN_ACT[i] = 0

%if one of second group of conditions is not fullfilled, the air mass flow integral will be set to 0 and cat diag inhibited.

ELSE

IF ((STATE_CP = MAX_PURGE **AND** STATE_CP_OLD[i] = NO_PURGE) **OR**

(STATE_CP = NO_PURGE **AND** STATE_CP_OLD[i] = MAX_PURGE)**)**

THEN MAF_INT_DLY_CAT_DIAG[i] = C_MAF_INT_SW_CAT_DIAG

LV_CAT_DIAG_CDN_ACT[i] = 0

% the beginning MAF_INT_DLY_CAT_DIAG[i] will not be set to zero, but initialized to C_MAF_INT_SW_CAT_DIAG, so the offset-free MAFintegral is the difference of MAF_INT_DLY_CAT_DIAG and _SW.

ENDIF

IF (MAF_INT_DLY_CAT_DIAG[i] < C_MAF_INT_DLY_CAT_DIAG)

THEN MAF_INT_DLY_CAT_DIAG[i](n) = MAF_INT_DLY_CAT_DIAG [i](n-1) +

MAF_CYL * NC_FAC_MAF_INT_20

%Air mass flow integral will be integrated after fulfilling all the activation conditions,

f it is lower, then threshold.

ELSE LV_CAT_DIAG_CDN_ACT[i] = 1 % Air mass flow integral exceeds the

threshold and stopped, cat diag could be activated

ENDIF

IF **(**(TMP_CND_1 = 1 **AND** OLD_CAT_DIAG_CDN [i] = 0) **AND** **(**(TEMP_CAT < C_TEMP_MIN_CAT) **OR** (TEMP_CAT > C_TEMP_MAX_CAT)**)** **OR** **(**(MAF < IP_MAF_LAM_CAT_MIN) **OR** (MAF > IP_MAF_LAM_CAT_MAX)**))**

THEN LV_CAT_DIAG_CDN_ACT[i] = 0

ELSE

IF **(**(TMP_CND_1 = 1 **AND** OLD_CAT_DIAG_CDN [i] = 1) **AND** **(**(TEMP_CAT < C_TEMP_MIN_CAT - C_TEMP_HYS_CAT) **OR** (TEMP_CAT > C_TEMP_MAX_CAT + C_TEMP_HYS_CAT)**)** **OR** **(**(IP_MAF_LAM_CAT_MIN - C_MAF_LAM_HYS_CAT > MAF) **OR** (IP_MAF_LAM_CAT_MAX + C_MAF_LAM_HYS_CAT < MAF)**))**

% engine speed and load conditions out of range, including hysteresis:

THEN LV_CAT_DIAG_CDN_ACT[i] = 0

ELSE

IF TMP_CND_1 = 0

THEN LV_CAT_DIAG_CDN_ACT[i] = 0

ENDIF

ENDIF

ENDIF

ENDIF

STATE_CP_OLD[i] = STATE_CP

%if one of first group of conditions is not fullfilled, the air mass flow need not be set to 0, cat diag inhibited.

The activation of the cat diagnosis is carried out (LV_CAT_DIAG_CDN_ACT[i] = 1), if the both groups of conditions are fulfilled and air mass flow integral after fullfilling the second group of conditions exceeds the treshold C_MAF_INT_ CAT_DIAG _DLY. If one of the second group of conditions is not fullfilled, air mass flow integral is set to 0 and after fullfilling of all starts the integration.

The value NC_FAC_MAF_INT_20 considers the sampling rate T_SAMPLE and converts MAF_CYL in g / s (NC_FAC_MAF_INT_20 = T_SAMPLE / 3.6 = 20 ms / 3.6).

IF LC_CAT_DIAG_CDN_ACT_MAN = 1

% diagnosis condition can be set manually without consideration of above conditions

THEN LV_CAT_DIAG_CDN_ACT[i] = 1

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_LAM_CAT_MIN	6	0...FFFFH	0...1389	0.021	mg/stk
LDPM_N_32_3_EGTR	6	0...FFH	0...8160	32	rpm
min. MAF threshold					
IP_MAF_LAM_CAT_MAX	6	0...FFFFH	0...1389	0.021	mg/stk
LDPM_N_32_3_EGTR	6	0...FFH	0...8160	32	rpm
max. MAF threshold					
C_TCO_MIN_CAT	1	0...FEH	-48...142.5	0.75	°C
Min TCO threshold					
C_TEMP_MIN_CAT	1	0...FFH	-33...990	4	°C
min. TEMP_CAT theshold					
C_TEMP_MAX_CAT	1	0...FFH	-33...990	4	°C
max. TEMP_CAT threshold					
C_TEMP_HYS_CAT	1	0...FFH	0...1023	4	°C
hysterese after exceeding TEMP_CAT threshold					
C_VS_MIN_CAT	1	0...FFH	0...255	1	km/h
min. VS threshold					
C_VS_MAX_CAT	1	0...FFH	0...255	1	km/h
max. VS threshold					
C_AMP_MIN_CAT_LS_DIAG	1	0 ... FFFFH	0 ... 5434	0.083	hPa
min. ambient pressure threshold					
C_CL_MAX_CAT_LS_DIAG	1	0 ... FFFFH	0 ... 2	3.05 e-5	-
Maximum canister load to allow catalyst and O2 sensor diagnosis					
LC_CAT_DIAG_CDN_ACT_MAN	1	0 ... 1H	0 ... 1	1	-
Flag to set diagnosis condition manually					
C_MAF_INT_DLY_CAT_DIAG	1	0...FFFFH	0...1820.42	0.028	g
Air mass flow threshold for cat diag activation after PUC, CL max purge, force stim. on, LS_down ready					
C_MAF_INT_SW_CAT_DIAG	1	0...FFFFH	0...1820.42	0.028	g
Air mass flow initial for cat diag activation after MAX to NO_PURGE and back					
C_MAF_LAM_HYS_CAT	1	0...FFH	0...21.5	0.084	mg/stk
hysterese after exceeding MAF_LAM_CAT threshold					
C_T_DLY_P_LAMB_MAX_CAT_DIAG	1	0...FFH	0...2.55	0.01	s
Max. P-share from Trim controller threshold					

B.31 Load / TPS Plausibility Check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_LOAD_TPS_PLAUS	O/V	0...1H	0...1	1	-
Boolean that indicates inconsistencies between actual load and throttle position (if set)					
LV_CDN_DIAG_LOAD_TPS_PLAUS	V	0...1H	0...1	1	-
Status of diagnosis flag for load/TPS plausibility check					
ERR_SYM_LOAD_TPS_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom load/TPS plausibility check					
LV_END_DIAG_LOAD_TPS_PLAUS	V	0...1H	0...1	1	-
End of diagnosis flag for load/TPS plausibility check					
PUT_MDL_DIF_MMV	V	8000...7FFFH	-2717...2716.967	0.083	hPa
Filtered pressure controller excitation					
PUT_MDL_DIF_MMV_REL	V/O	8000...7FFFH	-50...49.99847	1.52e-3	%
Filtered pressure controller excitation divided by ambient pressure					
AR_RED_DIF_REL_MMV	V/O	8000...7FFFH	-50...49.99847	1.52e-3	%
Filtered reduced area controller excitation (without adaptation values)					
LV_PUT_CTL_RST	V/O	0...01H	0...1	1	-
Control bit for resetting of pressure controller					
LV_AR_RED_CTL_RST	V/O	0...01H	0...1	1	-
Control bit for resetting of reduced area controller					
LV_ERR_RATIO_CHK	V/O	0...01H	0...1	1	-
Boolean for actual value MAF or TPS ratio check is present (yet available only for interface)					
AR_RED_SUM_COR_ACT_REL	V/O	8000...7FFFH	-100...99.996948	3.0518e-3	%
Active relative ar_red correction (included ar_red adaptation and ar_red controller)					

Input data:

PUT	PUT_MDL_DIF	AR_RED_AD_ADD	AR_RED_AD_FAC
AR_RED_DIF_REL	LV_IGK	AMP	LV_INH_LOAD_TPS_PLAUS
N_32	PQ	AR_RED_AD_ADD_MMV	MAP_DRV1
C_PQ_PUT_CTL	LV_ST_END	LV_MAF_SWI	LV_INH_MAP_CTL
AR_RED_BAS	AR_RED_AD_FAC_COR		

FUNCTION DESCRIPTION:

General information:

The load/TPS plausibility check proofs the consistency between load and throttle position. The plausibility is checked by comparing the excitation of the controller (reduced area if PQ <= C_PQ_PUT_CTL (which is defined in IMM, 30404T01.00A), pressure if PQ > C_PQ_PUT_CTL) with the corresponding "basic" value (AR_RED_BAS if area controlled, PUT if pressure controlled). In addition, the area adaptation values (AR_RED_AD_ADD and AR_RED_AD_FAC) are checked seperatly. This is necessary, because at closed throttle small absolute values (e.g. due to leakage air adaptation) may result in large relative values that pretend a bad system.

If the current load and throttle position signals are not consistent (inplausible) a flag is set, that indicates problems in the plausibility check (LV_ERR_LOAD_TPS_PLAUS=1).

If the controller excitation (or area adaptation value) is above a calibratable threshold, i.e. it is large compared to the corresponding basic reduced area, it is assumed, that an error in the system occured. The unplausibility detected only states, that the actual load and throttle position do not fit together and therefore a large controller excitation is required to bring them into line.

If the fuel system diagnosis is available it is possible to identify the reason of the unplausibility (i.e. error in the load signal or throttle position). To get reasonable results from the FSD, a certain delay time is required before accessing the FSD results.

Application conditions:

Initialization:
LV_CDN_DIAG_LOAD_TPS_PLAUS = 0
LV_END_DIAG_ LOAD_TPS_PLAUS = *Refer to filtering configuration for the initialisation value*
LV_ERR_ LOAD_TPS_PLAUS = *Refer to filtering configuration for the initialisation value*
ERR_SYM_ LOAD_TPS_PLAUS = *Refer to filtering configuration for the initialisation value*
If *LV_IGK = 1 and LV_ST_END= 0--->1*

Then

LV_PUT_CTL_RST=LV_AR_RED_CTL_RST=LV_ERR_RATIO_CHK=PUT_MDL_DIF_MMV=PUT_MDL_DIF_MMV_REL=AR_RED_DIF_REL_MMV=0

endif

Recurrence: ***20 ms*** all tasks except the following

Activation:

if **(**LV_IGK = 1 and LV_ST_END = 1) **and**
LV_ERR_LOAD_TPS_PLAUS=0 **and**
LV_INH_LOAD_TPS_PLAUS = 0 **and**
C_N_MIN_LOAD_TPS_PLAUS < N_32 < C_N_MAX_LOAD_TPS_PLAUS **and**
C_PQ_MIN_LOAD_TPS_PLAUS < PQ < C_PQ_MAX_LOAD_TPS_PLAUS **and**
(C_MAP_DRV1_MIN_LOAD_TPS_PLAUS < MAP_DRV1 < C_MAP_DRV1_MAX_LOAD_TPS_PLAUS)

then

LV_CDN_DIAG_LOAD_TPS_PLAUS=1

else

LV_CDN_DIAG_LOAD_TPS_PLAUS=0

endif

All the calculations for the plausibility checks are only done, if the function is activated, i.e. LV_CDN_DIAG_LOAD_TPS_PLAUS = 1, otherwise the plausibility check is stopped and all values are stored until reactivation.

B.31.1 Plausibility check of area adaptation values

Description:

The additive and multiplicative adaptation values of the reduced area are checked seperatly. If one of them exceeds the corresponding threshold, a related error flag is set, if the value is inside the thresholds the respective flag is reset. Because adaptation is done with a slow recurrence, no filtering of the adaptation values is required and they are directly used.

Formula section:

Check the additive adaptation mean moving value of the reduced area.

If (AR_RED_AD_ADD-AR_RED_AD_ADD_MMV) < C_AR_RED_AD_ADD_PLAUS_MIN **or**

(AR_RED_AD_ADD -AR_RED_AD_ADD_MMV > C_AR_RED_AD_ADD_PLAUS_MAX

then

Symptom "additive adaption out of range" is active, anti-bounce counter increment

ERR_SYM_LOAD_TPS_PLAUS = 1H

else

Symptom "additive adaption out of range" is passive, anti-bounce counter decrement

ERR_SYM_LOAD_TPS_PLAUS = 0H

Endif

Check the multiplicative adaptation value of the reduced area.

If AR_RED_AD_FAC < C_AR_RED_AD_FAC_PLAUS_MIN **or**
AR_RED_AD_FAC > C_AR_RED_AD_FAC_PLAUS_MAX

then

Symptom "multiplicative adaption out of range" is active, anti-bounce counter increment

ERR_SYM_LOAD_TPS_PLAUS = 2H

else

Symptom "multiplicative adaption out of range" is passive, anti-bounce counter decrement

ERR_SYM_LOAD_TPS_PLAUS = 0H

endif

Filtering :

Apply filter on current symptoms
If filtering result available (after debounce)
LV_ERR_LOAD_TPS_PLAUS = filtering result
LV_END_DIAG_LOAD_TPS_PLAUS = 1
Delivery the result to Error Management
Endif

B.31.2 Plausibility check of reduced area controller excitation

Description:

In the plausibility check only the steady state conditions are of interest. Therefore the controller excitations fo the reduced area controller is smoothed with a low pass filter. This controller output is only calculated as long as the corresponding controller is active. If the plausibility check is deactivated (LV_CDN_DIAG_LOAD_TPS_PLAUS = 0) the calculation of the moving mean value from the reduced area controller output is stopped (but not reset).

In case of no AMP sensor is available, diagnosis threshold must be set versus AMP in order too take in account possible variation of Ambient Pressure (ECU reset at high altitude, vehicle transportation on a truck from high to low altitude or vice-versa).

If the filtered reduced area controller AR_RED_DIF_REL_MMV reaches the thresholds AR_RED_DIF_REL_PLAUS_MIN or AR_RED_DIF_REL_PLAUS_MAX the plausibility check error is set to active.

Formula section:

if PQ <= C_PQ_PUT_CTL

then

$$AR\_RED\_DIF\_REL\_MMV_N = AR\_RED\_DIF\_REL\_MMV_{N-1} * (1 - C\_CRLC\_AR\_RED\_DIF\_REL) + AR\_RED\_DIF\_REL_N * C\_CRLC\_AR\_RED\_DIF\_REL$$

endif

AR_RED_SUM_COR_ACT_REL = AR_RED_DIF_REL_MMV + AR_RED_AD_FAC_COR +
((AR_RED_AD_ADD – AR_RED_AD_ADD_MMV) * 100 / AR_RED_BAS)

(converted to phys. resolution of AR_RED_SUM_COR_ACT_REL)

Calculation of diagnosis thresholds

AR_RED_DIF_REL_PLAUS_MIN = IP_AR_RED_DIF_REL_PLAUS_MIN

AR_RED_DIF_REL_PLAUS_MAX = IP_AR_RED_DIF_REL_PLAUS_MAX

Check the reduced area controller excitation

if PQ <= C_PQ_PUT_CTL **and**
(AR_RED_SUM_COR_ACT_REL < AR_RED_DIF_REL_PLAUS_MIN **or**
AR_RED_SUM_COR_ACT_REL > AR_RED_DIF_REL_PLAUS_MAX **)**
then

Symptom “filtered reduced area controller out of range” is active, anti-bounce counter increment

ERR_SYM_LOAD_TPS_PLAUS = 4H

else

Symptom “filtered reduced area controller out of range” is passive, anti-bounce counter decrement

ERR_SYM_LOAD_TPS_PLAUS = 0H

Endif

Filtering :

Apply filter on current symptoms
If filtering result available (after debounce)
LV_ERR_LOAD_TPS_PLAUS = filtering result
LV_END_DIAG_LOAD_TPS_PLAUS = 1
Delivery the result to Error Management
Endif

B.31.3 Plausibility check of pressure controller excitation

Description:

In the plausibility check only the steady state conditions are of interest. Therefore the controller excitation of the pressure controller is smoothed with a low pass filter. This filtered controller output is only calculated as long as the controller itself is active. If the plausibility check is deactivated (LV_CDN_DIAG_LOAD_TPS_PLAUS = 0) the calculation of the moving mean value from the pressure controller output is stopped (but not reset).

Like in the algorithm for reduced area plausibility check, in case of no AMP sensor is available, diagnosis threshold must be set versus AMP in order too take in account possible variation of Ambient Pressure (ECU reset at high altitude, vehicle transportation on a truck from high to low altitude or vice-versa).

If the ratio of the filtered pressure controller and the (adapted) ambient pressure PUT_MDL_DIF_MMV_REL reaches the threshold PUT_MDL_DIF_REL_PLAUS_MIN or PUT_MDL_DIF_REL_PLAUS_MAX the plausibility check error is set to active.

Formula section:

if PQ > C_PQ_PUT_CTL
then

$$PUT\_MDL\_DIF\_MMV_{N} = PUT\_MDL\_DIF\_MMV_{N-1} * (1 - C\_CRLC\_PUT\_MDL\_DIF) + PUT\_MDL\_DIF_{N} * C\_CRLC\_PUT\_MDL\_DIF$$

$$PUT\_MDL\_DIF\_MMV\_REL = \frac{PUT\_MDL\_DIF\_MMV}{PUT}$$

endif

Calculation of diagnosis thresholds

PUT_MDL_DIF_REL_PLAUS_MIN = IP_PUT_MDL_DIF_REL_PLAUS_MIN

PUT_MDL_DIF_REL_PLAUS_MAX = IP_PUT_MDL_DIF_REL_PLAUS_MAX

Check the ambient controller excitation

if PQ > C_PQ_PUT_CTL **and**
(PUT_MDL_DIF_MMV_REL < PUT_MDL_DIF_REL_PLAUS_MIN **or**
PUT_MDL_DIF_MMV_REL > PUT_MDL_DIF_REL_PLAUS_MAX **)**
then

Symptom “filtered ambient controller out of range” is active, anti-bounce counter increment

ERR_SYM_LOAD_TPS_PLAUS = 8H

else

Symptom “filtered ambient controller out of range” is passive, anti-bounce counter decrement

ERR_SYM_LOAD_TPS_PLAUS = 0H

Endif

Filtering :

Apply filter on current symptoms
If filtering result available (after debounce)
LV_ERR_LOAD_TPS_PLAUS = filtering result
LV_END_DIAG_LOAD_TPS_PLAUS = 1
Delivery the result to Error Management
Endif

B.31.4 Reset and additional measures to be taken

Description:

If a debounced unplausibility is detected the reduced area controller and the pressure controller must be reset (also for the rest of the driving cycle). Precontrolled state of the intake manifold model (without controller excitation) is precondition to be able to decide which of the sensors caused the error.

Temporary the old flag indicating an unplausibility´, LV_ERR_RATIO_CHK, will be provided too.

Yet only a few generic actions taken on plausibility error are defined:

- The Intake Manifold Model goes from closed to open loop and in addition the controllers are reset.
- The adaptation of the ambient pressure is forbidden.

Additional project-specific actions can be defined in the "Application incidence for load/TPS plausibility check".

Formula section:

if LV_ERR_LOAD_TPS_PLAUS = 1
then

if LV_MAF_SWI = 1 **or** LV_INH_MAP_CTL = 1
then LV_AR_RED_CTL_RST = 1 all set for this DC
LV_PUT_CTL_RST = 1

endif

LV_ERR_RATIO_CHK = 1

endif

B.32 Identification of reason for unplausibility (only possible if LC_USE_FSD = 1)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TPS_PLAUS	O/V	0...1H	0...1	1	-
Boolean that indicates that TPS sensor is faulty					
LV_ERR_LOAD_PLAUS	O/V	0...1H	0...1	1	-
Boolean that indicates that load sensor is faulty					
LV_CDN_DIAG_TPS_PLAUS	V	0...1H	0...1	1	-
Status of diagnosis flag for TPS plausibility check					
LV_CDN_DIAG_LOAD_PLAUS	V	0...1H	0...1	1	-
Status of diagnosis flag for load plausibility check					
ERR_SYM_TPS_PLAUS	O/V	0H 8H	NO_SYM SYM_3	1	-
Detected symptom flag for TPS plausibility check					
ERR_SYM_LOAD_PLAUS	O/V	0H 8H	NO_SYM SYM_3	1	-
Detected symptom flag for load plausibility check					
LV_END_DIAG_TPS_PLAUS	V	0...1H	0...1	1	-
End of diagnosis flag for TPS plausibility check					
LV_END_DIAG_LOAD_PLAUS	V	0...1H	0...1	1	-
End of diagnosis flag for load plausibility check					

Input data:

LV_DC	LV_END_DIAG_WIN_FSD[NC_CBK_EX_NR]	LV_ERR_FSD[NC_CBK_EX_NR]	LV_ERR_LOAD_TPS_PLAUS
LV_END_DIAG_LOAD_TPS_PLAUS			

Application conditions:

Initialization:
LV_CDN_DIAG_TPS_PLAUS = 0, LV_CDN_DIAG_LOAD_PLAUS=0
LV_END_DIAG_TPS_PLAUS , LV_END_DIAG_LOAD_PLAUS= *Refer to filtering configuration for the initialisation value*
LV_ERR_TPS_PLAUS, LV_ERR_LOAD_PLAUS = *Refer to filtering configuration for the initialisation value*
ERR_SYM_TPS_PLAUS, ERR_SYM_LOAD_PLAUS = *Refer to filtering configuration for the initialisation value*

Every 20ms:

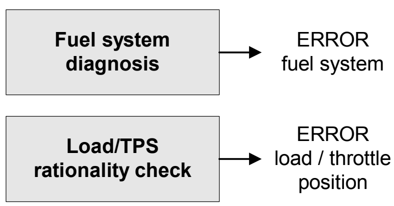
LV_FSD_TMP =1
LV_TPS_TMP = 0
LV_END_FSD_TMP = 1

Activation:

if LV_DC=1
and LC_USE_FSD = 1
and LV_END_DIAG_LOAD_TPS_PLAUS = 1
and LV_ERR_LOAD_PLAUS = 0
and LV_ERR_TPS_PLAUS = 0
then LV_CDN_DIAG_TPS_PLAUS=1 diagnosis is switched on
LV_CDN_DIAG_LOAD_PLAUS=1 "
Else LV_CDN_DIAG_TPS_PLAUS=0 diagnosis is switched off
LV_CDN_DIAG_LOAD_PLAUS=0
endif

Description:

The load/TPS rationality check can detect unplausibilities in the system, but only in combination with the output of the FSD (fuel system diagnosis) it is possible to identify the faulty sensor (load or throttle position). Usually the reduced area and the pressure controller will be deactivated (and reset) if a plausibility error occurs. This means, the IMM goes to precontrolled (the system will be α-N-controlled) and the only signal used is the throttle position, it means MAF is calculated based on TPS signal but no sensor signal. Therefore if the FSD detects an error in the fuel path, this means that the TPS signal is bad. If the FSD detects no error this means, the unplausibility was caused from a faulty load sensor.



ERROR fuel system	0	1	0	1
ERROR load / throttle position	0	0	1	1
	-	-	load	TPS

The diagnosis can be started if the FSD-diagnosis has been finished a diagnosis window (LV_END_DIAG_WIN_FSD[i] = 1).

If there are two banks (i=0,1), it is necessary to take into account both of them.

<u>Formula section:</u>

```
For    i=0    to  (NC_CBK_EX_NR - 1)           then
    if  [LV_END_DIAG_WIN_FSD[i]=1 or LV_ERR_FSD[i]=1] and  LV_FSD_TMP = 1

    then    LV_FSD_TMP=1
    else    LV_FSD_TMP = 0
    endif
endfor
```

```
if      LV_FSD_TMP = 1
then    if      LV_ERR_LOAD_TPS_PLAUS=0
        then    LV_END_DIAG_TPS_PLAUS=1
                LV_END_DIAG_LOAD_PLAUS=1
        else re-initialization of end of diagnosis
                LV_END_DIAG_TPS_PLAUS=0
                LV_END_DIAG_LOAD_PLAUS=0
        Endif
endif

if  LV_ERR_LOAD_TPS_PLAUS=1
then
    For    (i=0 to (NC_CBK_EX_NR – 1)       and     LV_TPS_TMP =0)     then
        If        LV_ERR_FSD[i]=1
        then      LV_TPS_TMP = 1
        endif
    endfor
    if   LV_TPS_TMP = 1
    then    ERR_SYM_TPS_PLAUS= 8H                          directly set for this DC
            ERR_SYM_LOAD_ PLAUS = 0H
            LV_ERR_TPS_PLAUS = 1
            LV_ERR_LOAD_PLAUS = 0
    Else
            For    i=0     to      (NC_CBK_EX_NR – 1)        then
                if        LV_END_DIAG_WIN_FSD[i]=1       and LV_END_FSD_TMP= 1
                then      LV_END_FSD_TMP =1
                else      LV_END_FSD_TMP = 0
                endif
            endfor
            if      LV_END_FSD_TMP = 1
            then     ERR_SYM_LOAD_ PLAUS = 8H             directly set for this DC
                     ERR_SYM_TPS_PLAUS= 0H
                     LV_ERR_TPS_PLAUS = 0
                     LV_ERR_LOAD_PLAUS = 1
            Endif
    endif
endif
```

```
if      LV_ERR_TPS_PLAUS = 1      or      LV_ERR_LOAD_PLAUS = 1
then    LV_END_DIAG_TPS_PLAUS=1
        LV_END_DIAG_LOAD_PLAUS=1
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PUT_MDL_DIF_REL_PLAUS_MAX	6	0000...FFFFH	-50 ... 49.998	0.0015	%
LDPM_AMP_LOAD_TPS_PLAUS	6	0...FFFFH	0...5434	0.083	hPa
Upper threshold for pressure controller plausibility check					
IP_PUT_MDL_DIF_REL_PLAUS_MIN	6	0000...FFFFH	-50 ... 49.998	0.0015	%
LDPM_AMP_LOAD_TPS_PLAUS	6	0...FFFFH	0...5434	0.083	hPa
Lower threshold for pressure controller plausibility check					
C_AR_RED_AD_ADD_PLAUS_MAX	1	8000...7FFFH	-29.29 ... 29.29	8.94e-4	cm^2
Maximum allowed deviation of additive adaptation value of reduced area					
C_AR_RED_AD_ADD_PLAUS_MIN	1	8000...7FFFH	-29.29 ... 29.29	8.94e-4	cm^2
Minimum allowed deviation of additive adaptation value of reduced area					
C_AR_RED_AD_FAC_PLAUS_MAX	1	8000...7FFFH	-50 ... 49.998	0.0015	%
Maximum allowed deviation of multiplicative adaptation value of reduced area					
C_AR_RED_AD_FAC_PLAUS_MIN	1	8000...7FFFH	-50 ... 49.998	0.0015	%
Minimum allowed deviation of multiplicative adaptation value of reduced area					
IP_AR_RED_DIF_REL_PLAUS_MAX	6x4	0000...FFFFH	-100...99.996948	3.0518e-3	%
LDPM_AMP_LOAD_TPS_PLAUS	6	0...FFFFH	0...5434	0.083	hPa
LDPM_AR_RED_BAS_LOAD_TPS_PLAUS	4	0...FFFFH	0...58.59285	0.8941e-3	cm^2
Upper threshold for reduced area controller plausibility check					
IP_AR_RED_DIF_REL_PLAUS_MIN	6x4	0000...FFFFH	-100...99.996948	3.0518e-3	%
LDPM_AMP_LOAD_TPS_PLAUS	6	0...FFFFH	0...5434	0.083	hPa
LDPM_AR_RED_BAS_LOAD_TPS_PLAUS	4	0...FFFFH	0...58.59285	0.8941e-3	cm^2
Lower threshold for reduced area controller plausibility check					
C_CRLC_PUT_MDL_DIF	1	0...FFH	0...0.996	0.0039	-
Constant for pressure controller output filtering					
C_CRLC_AR_RED_DIF_REL	1	0...FFH	0...0.996	0.0039	-
Constant for reduced area controller output filtering					
C_MAP_DRV1_MAX_LOAD_TPS_PLAUS	1	8000...7FFFH	-82.9...82.9	0.00253	hPa/msec
Maximum allowed pressure change in time to allow plausibility check					
C_MAP_DRV1_MIN_LOAD_TPS_PLAUS	1	8000...7FFFH	-82.9...82.9	0.00253	hPa/msec
Minimum required pressure change in time to allow plausibility check					
C_N_MAX_LOAD_TPS_PLAUS	1	00...FFH	0...8160	32	rpm
Maximum engine speed allowed for load/TPS plausibility check					
C_N_MIN_LOAD_TPS_PLAUS	1	00...FFH	0...8160	32	rpm
Minimum engine speed required for load/TPS plausibility check					
C_PQ_MAX_LOAD_TPS_PLAUS	1	0000...FFFFH	0...0.9999847	1.52e-5	-
Maximum allowed pressure ratio at the throttle for load/TPS plausibility check					
C_PQ_MIN_LOAD_TPS_PLAUS	1	0000...FFFFH	0...0.9999847	1.52e-5	-
Minimum required pressure ratio at the throttle for load/TPS plausibility check					
LC_USE_FSD	1	0...01H	0...1	1	-
Flag for usage of FSD error to distinguish between load or TPS error					

B.33 Application incidences for Load-/TPS Plausibility Check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_LOAD_TPS_PLAUS	O/V	0...1H	0...1	1	-
Flag to stop the load/TPS plausibility check					

Input data:

LV_ ERR_TPS	LV_DC	LV_ENA_MAF	LV_ENA_MAP
LV_ERR_TCO	LV_ERR_CAM_DE_IVVT_EX_i	LV_ERR_TIA	LV_ERR_CAM_DE_IVVT_IN_i
TPS	LV_ERR_CPS	LV_ERR_MEC_CPS	LV_ERR_MEC_IVVT_IN_i
LV_ERR_SLV_IVVT_IN_i	LV_ERR_TPS_AD_GAIN		LV_ERR_MEC_IVVT_EX_i
LV_ERR_SLV_IVVT_EX_i	LV_ERR_AMP	LV_ERR_AMP_PLAUS	LV_ERR_PUT
LV_N_LIM_ETC_LIH	LV_PUC	CTR_AR_RED_AD_ADD_FAST	TPS_AV_ENVD_H
MAP_ENVD	ISAPWM_ISA_ENVD	MAP_MES_BAS_ENVD	TPS_SP_ENVD
TPS_AV_1_ENVD	TPS_AV_2_ENVD	VCC_TPS_DIAG_ENVD	LV_ERR_ISA_1
LV_ERR_ISA_2	LV_ERR_VIM_PLAUS		

FUNCTION DESCRIPTION:

General information:

Depending on projects specific requirements, the load/TPS plausibility check can be inhibited by setting LV_INH_LOAD_TPS_PLAUS = 1.

Application conditions:

Initialization: LV_INH_LOAD_TPS_PLAUS is set 0 at reset
or at transition LV_DC: 0 →1

Recurrence: **20 ms**

Activation: **if** LV_DC=1

Diagnostic	Symptom Description	Symptom	Filter type
LOAD_TPS_PLAUS	additive adaption out of range	SYM_0	*MEM*
	multiplicative adaption out of range	SYM_1	
	filtered reduced area controller out of range	SYM_2	
	filtered ambient controller out of range	SYM_3	

List of Environmental Data to store in Failure Memory: for ISA system:

TPS

TPS_AV_ENVD_H

ISAPWM_ISA_ENVD

MAP_MES_BAS_ENVD

For ETC system:

TPS_SP_ENVD
TPS_AV_1_ENVD
TPS_AV_2_ENVD
MAP_MES_BAS_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
LOAD_PLAUS		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Unplausibility caused by faulty load sensor	SYM_3	

List of Environmental Data to store in Failure Memory: for ISA system:
TPS
TPS_AV_ENVD_H
ISAPWM_ISA_ENVD
MAP_MES_BAS_ENVD
For ETC system:
TPS_SP_ENVD
TPS_AV_1_ENVD
TPS_AV_2_ENVD
MAP_MES_BAS_ENVD

Diagnostic	Symptom Description	Symptom	Filter type
TPS_PLAUS		SYM_0	*MEM*
		SYM_1	
		SYM_2	
	Unplausibility caused by faulty throttle sensor	SYM_3	

List of Environmental Data to store in Failure Memory: for ISA system:
VCC_TPS_DIAG_ENVD
MAP_ENVD
TPS_AV_ENVD_H
ISAPWM_ISA_ENVD
For ETC system:
VCC_TPS_DIAG_ENVD
MAP_ENVD
TPS_AV_1_ENVD
TPS_AV_2_ENVD

B.33.1 Inhibition of the load/TPS plausibility check

If an error is marked in the throttle channel plausibility flag and yet no channel is detected as faulty, the load/TPS plausibility check will be inhibited until the plausibility check for throttle position sensor is finished.

```
if    LV_ERR_TPS = 1                                                       or
      (LV_ENA_MAF = 0 And LV_ENA_MAP = 0)  (no working load sensor)        or
      LV_ERR_CAM_DE_IVVT_IN_i = 1                                          or
      LV_ERR_CAM_DE_IVVT_EX_i = 1                                          or
      LV_ERR_MEC_IVVT_IN_i = 1                                             or
      LV_ERR_MEC_IVVT_EX_i = 1                                             or
      LV_ERR_SLV_IVVT_IN_i = 1                                             or
      LV_ERR_SLV_IVVT_EX_i = 1                                             or
      LV_ERR_CPS = 1                                                       or
      LV_ERR_MEC_CPS = 1                                                   or
      LV_ERR_TPS_AD_GAIN = 1                                               or
      LV_ERR_TCO = 1                                                       or
      LV_ERR_TIA = 1                                                       or
      LV_ERR_AMP = 1                                                       or
      LV_ERR_AMP_PLAUS = 1                                                 or
      LV_ERR_PUT = 1                                                       or
      LV_ERR_ISA_1 = 1                                                     or
      LV_ERR_ISA_2 = 1                                                     or
      LV_N_LIM_ETC_LIH = 1                                                 or
      LV_PUC = 1                                                           or
      LV_ERR_VIM_PLAUS = 1                                                 or
      CTR_AR_RED_AD_ADD_FAST > 0
then
      LV_INH_LOAD_TPS_PLAUS = 1
else
      LV_INH_LOAD_TPS_PLAUS = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_LOAD_TPS_PLAUS	1	0...FFH	0...255	1	-
Increment of the load/TPS plausibility check anti-bounce counter					
C_ABC_MAX_LOAD_TPS_PLAUS	1	1...FFH	1...255	1	-
Threshold to be reached, before permanently activating load/TPS plausibility error					

B.34 Ambient and Manifold pressure plausibility diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_END_DIAG_AMP_PLAUS	O/V	0...1H	0...1	1	-
End of diagnosis flag for Ambient pressure plausibility diagnosis					
LV_END_DIAG_PUT_PLAUS	O/V	0...1H	0...1	1	-
End of diagnosis PUT plausibility diagnosis					
LV_END_DIAG_MAP_PLAUS	O/V	0...1H	0...1	1	-
End of diagnosis flag for Manifold pressure plausibility diagnosis					
LV_ERR_AMP_PLAUS	O/V	0...1H	0...1	1	-
Boolean for detected error Ambient pressure plausibility diagnosis					
LV_ERR_PUT_PLAUS	O/V	0...1H	0...1	1	-
Boolean for error currently present on PUT plausibility diagnosis					
LV_ERR_MAP_PLAUS	O/V	0...1H	0...1	1	-
Boolean for detected error of Manifold pressure plausibility diagnosis					
LV_ERR_AMP_PLAUS_DIAG	O/V	0...1H	0...1	1	-
Boolean for detected error of AMP_MES plausibility					
LV_ERR_MAP_PLAUS_DIAG	O/V	0...1H	0...1	1	-
Boolean for detected error of MAP_MES plausibility					
PUT_MES_PLAUS_DIAG_PU	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Value of PUT_MES during plausibility check in engine state PULL					
STATE_PRS_AIR_PLAUS	V	0H 1H 2H 3H 4H 5H	DIAG_NOT_ACT PRS_AIR_PLAUS_ES PRS_AIR_PLAUS_PWL PRS_AIR_PLAUS_FL PRS_AIR_PLAUS_PU PRS_AIR_PLAUS_IS	1	-
State of AMP-MAP-PUT plausibility diagnosis					
MAP_MES_PLAUS_DIAG_ES	V	0...FFFFH	0...5.434E+3	0.08291752	hPa
Value of MAP_MES during plausibility check in engine state STOP					
LV_IS_PRS_AIR_PLAUS_ERR	V	0...1H	0...1	1	-
Indicator that engine was in idle speed during current driving cycle					
LV_SYM_AMP_PUT_PLAUS_FL	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of AMP_PUT plausibility check in engine state FULL LOAD					
T_DLY_AMP_PUT_PLAUS_PU	V	0...FFFFH	0...6.5535E+3	0.1	s
Delay time before starting diagnosis for AMP-PUT plausibility when engine in PULL					
AMP_MAP_DIF_DIAG	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Difference ambient vs. manifold pressure (AMP_MES ~ MAP_MES)					
AMP_PUT_DIF_DIAG	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Difference ambient vs. pressure up throttle (AMP_MES ~ PUT_MES)					
PUT_MAP_DIF_DIAG	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Difference pressure up throttle vs. manifold pressure (PUT_MES ~ MAP_MES)					
CTR_MAP_MAF_PLAUS	V	0...FFFFH	0...6.5535E+4	1	-
Delay counter before setting diagnosis flag for MAP-MAF plausibility					
CTR_MAP_PLAUS	V	0...FFFFH	0...6.5535E+4	1	-
Delay counter before setting diagnosis flag for MAP plausibility					
T_DLY_AMP_PUT_PLAUS_FL	V	0...FFFFH	0...6.5535E+3	0.1	s
Delay Time before seting PRS_AIR_PLAUS_FL Mode					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_AMP_PLAUS	V	0...1H	0...1	1	-
Status of diagnosis flag for Ambient pressure plausibility					
ERR_SYM_AMP_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom Ambient pressure plausibility					
ERR_SYM_MAP_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom Manifold pressure plausibility diagnosis					
LV_CDN_DIAG_PUT_PLAUS	V	0...1H	0...1	1	-
Boolean for diagnosis condition present on PUT plausibility diagnosis					
LV_CDN_DIAG_MAP_PLAUS	V	0...1H	0...1	1	-
Status of diagnosis flag for Manifold pressure plausibility					
ERR_SYM_PUT_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected error PUT plausibility diagnosis					
CTR_AMP_PUT_PLAUS	V	0...FFFFH	0...6.5535E+4	1	-
Delay counter before setting diagnosis flag for AMP-PUT plausibility					
CTR_PUT_MAP_PLAUS	V	0...FFFFH	0...6.5535E+4	1	-
Delay counter before setting diagnosis flag for PUT-MAP plausibility					
LV_SYM_AMP_PUT_PLAUS_ES	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of AMP_PUT plausibility check in engine state STOP					
LV_SYM_PUT_MAP_PLAUS_ES	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of PUT_MAP plausibility check in engine state STOP					
PUT_MES_PLAUS_DIAG_ES	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Value of PUT_MES during plausibility check in engine state STOP					
LV_SYM_AMP_MAP_PLAUS_FL	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of AMP_MAP plausibility check in engine state FULL LOAD					
LV_SYM_PUT_MAP_PLAUS_FL	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of PUT_MAP plausibility check in engine state FULL LOAD					
PUT_MES_PLAUS_DIAG_FL	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Value of PUT_MES during plausibility check while in full load or high part load					
MAP_MES_PLAUS_DIAG_FL	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Value of MAP_MES during plausibility check in engine state FULL_LOAD					
MAP_MES_PLAUS_DIAG_IS	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Value of MAP_MES during plausibility check while in idle or low part load					
LV_FL_PRS_AIR_PLAUS_ERR	V	0...1H	0...1	1	-
Indicator that engine was in full load or high part load during current driving cycle					
CTR_AMP_MAP_PLAUS	V	0...FFFFH	0...6.5535E+4	1	-
Delay counter before setting diagnosis flag for AMP-MAP plausibility					
LV_SYM_AMP_PUT_PLAUS_PU	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of AMP_PUT plausibility check in engine state PULL					
LV_SYM_AMP_MAP_PLAUS_ES	V	0...1H	0...1	1	-
Logical bit indicating preliminary status of AMP_MAP plausibility check in engine state STOP					
CTR_MAP_MAF_PLAUS_MAX	-	0...FFFFH	0...6.5535E+4	1	-
Maximum of CTR_MAP_MAF_PLAUS					
CTR_PRS_AIR_PLAUS_MAX	-	0...FFFFH	0...6.5535E+4	1	-
Maximum of CTR_xx_PLAUS_ES					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_AMP_ENA_TMP	-	0...1H	0...1	1	-
Temporary variable indicating that AMP sensor is used and that there is no electrical error					
LV_MAP_ENA_TMP	-	0...1H	0...1	1	-
Temporary variable indicating that MAP sensor is used and that there is no electrical error					
LV_PUT_ENA_TMP	-	0...1H	0...1	1	-
Temporary variable indicating that PUT sensor is used and that there is no electrical error					
LV_STATE_PRS_AIR_PLAUS_NEW	-	0...1H	0...1	1	-
Bit indicating that STATE_PRS_AIR_PLAUS has just changed					

Input data:

LV_ES	AMP_MES_BAS	MAP_MES_BAS	PQ_SP
N_32	TPS_AV	TPS_SP_MDL_MAX	PUT_MES_BAS
NC_CHRG_CONF	LV_INH_DIAG_PRS_AIR_PLAUS_IS	LV_IS	CTR_DLY_AMP_AD
T_PWL	NC_PUT_CONF	NC_MAP_CONF	NC_AMP_CONF
LV_TCHA_CONF	LV_PL	LV_INH_DIAG_PRS_AIR_PLAUS	MAP_SP
LV_PUC	LV_PU	LV_ST_END	LV_INH_DIAG_PRS_AIR_PLAUS_ES
LV_INH_DIAG_PRS_AIR_PLAUS_FL	LV_INH_DIAG_PRS_AIR_PLAUS_PWL	LV_INH_DIAG_PRS_AIR_PLAUS_PU	NC_IDX_DIAG_AMP_PLAUS
NC_IDX_DIAG_MAP_PLAUS	NC_IDX_DIAG_PUT_PLAUS	LV_IGK	AMP
VP_AMP	VP_MAP_MV	VP_PUT_MV	VP_MAP_MV_MAX_DIAG
VP_MAP_MV_MIN_DIAG	VP_PUT_MV_MAX_DIAG_ES	VP_PUT_MV_MIN_DIAG_ES	LV_AMP_SWI
LV_PUT_SWI	LV_MAP_SWI	LV_ERR_AMP	LV_ERR_PUT
LV_ERR_MAP	VS	MAF_THR	MAF_CYL
MAF	MAP	LV_ENA_MAF	STATE_CTL_INSY
NC_MAF_CONF	VP_AMP_MAX_DIAG	VP_AMP_MIN_DIAG	C_AMP_MIN
C_AMP_MAX	PWM_WG		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_PUT_MAX_DIAG_PLAUS_PU	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Maximum difference between AMP and PUT sensors during pull					
C_CTR_MAP_MAF_PLAUS_INI	1	1...FFH	1...255	1	-
Initial value of delay counter for MAP-MAF plausibility before setting of error symptom					
C_CTR_PRS_AIR_PLAUS_INI	1	1...FFH	1...255	1	-
Initial value of delay counter before setting of error symptom for air pressure sensor					
C_MAF_KGH_DIF_MAX_MAP_PLAUS	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Maximum mass air flow difference for MAP plausibility detection					
C_MAF_MAX_MAP_PLAUS	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
MAF maximum threshold for MAP plausibility detection					
C_MAF_MIN_MAP_PLAUS	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
MAF minimum threshold for MAP plausibility detection					
C_MAP_MAX_DIF_MAP_PLAUS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum difference for MAP plausibility detection					
C_MAP_MES_DIF_PLAUS_DIAG	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimal difference between MAP_MES and MAP measured during engine stop or full load					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAP_MES_DIF_PLAUS_DIAG_IS_ES	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Maximum difference between MAP_MES during engine stop and idle					
C_MAP_MES_MIN_MAP_PLAUS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum MAP_MES for MAP plausibility detection					
C_PRS_MAX_DIAG_PLAUS_ES	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximal difference between two pressure sensors while standing engine					
C_PRS_MAX_DIAG_PLAUS_RUN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximal difference between two pressure sensors while running engine or full load					
C_PRS_MIN_DIAG_PLAUS_RUN	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Minimal difference between two pressure sensors while running engine or full load					
C_PUT_MAP_MAX_DIAG_PLAUS_FL	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Maximum difference between PUT and MAP sensors during high part load or full load					
C_PUT_MES_DIF_PLAUS_DIAG	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimal difference between PUT measured during engine stop and full load					
C_PWM_WG_DIAG_PLAUS_FL	1	0...FFFFH	0...99.9984741	0.00152588	%
Allowed PWM_WG deviation to detect engine state Full load					
C_TPS_SP_MDL_MAX_DE	1	0...3FFFH	0...119.5	0.00729415	°TPS
Allowed throttle position deviation to detect engine state FULL LOAD					
C_T_DLY_AMP_PUT_PLAUS_FL	1	0...FFFFH	0...6.5535E+3	0.1	s
Delay Time before starting diagonsis for air pressure plausiblity in FL					
C_T_DLY_AMP_PUT_PLAUS_PU_INI	1	0...FFFFH	0...6.5535E+3	0.1	s
Delay time before starting diagnosis for AMP-PUT plausibility when engine at PULL - init value					
C_T_DLY_PRS_AIR_PLAUS_PWL	1	0...FFFFH	0...6.5535E+3	0.1	s
Delay time before starting diagnosis for air pressure plausibility in power latch					
C_VS_MAX_ES_DIAG_PLAUS	1	0...FFH	0...255	1	km/h
Vehicle speed threshold at vehicle stopped for plausibility diagnosis					
LC_PRS_AIR_SENS_2_ERR_SAVE	1	0...1H	0...1	1	-
Switch whether a detected error of a pair of sensors is sent to ERRM if it can not be assigned to one sensor					
IP_AMP_PUT_MAX_PLAUS_DIAG_FL	6	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDP_N_32_IP_AMP_PUT_MAX_DIAG	6	0...FFH	0...8.16E+3	32	rpm
Maximum Difference between AMP and PUT at Full load or high part load dependent on engine speed					

Import actions:

ACTION_ERRM_GetLvEndDiag(IN <IDX_DIAG>, OUT <LV_END_DIAG>)
Action that returns the status of the failure availability
ACTION_ERRM_NoFilterReset(IN <IDX_DIAG>, OUT <LV_ERR>)
This action resets data filter in case of no filter usage
ACTION_ERRM_NoFilterSymptom(IN <IDX_DIAG>, IN <LV_CDN_DIAG>, IN <ERR_SYM>, IN <LV_ERR_SET>, IN <LV_ERR_RST>, IN <LV_END_DIAG>, OUT <LV_ERR>)
This action returns the result on symptoms detected at each diagnostic recurrence, when no filter is used

B.34.1 General Information:

For systems equipped with more air pressure sensors the plausibility diagnosis is done by comparing the measured pressures at stopped engine, when there is ambient pressure everywhere, also at the throttle and in the intake manifold. If two sensors show the same pressure and a third differs significantly, the third sensor is implausible.

If there are only two pressure sensors or if the implausibility occurs in an engine state when only two pressures are equal or close to each other, then only an implausible pair on sensors can be pointed to. In another engine state is tried to find out, which of the two sensors is faulty.

Implausibilities are detected based on the following relations

AMP = PUT in ES and PWL

AMP = MAP in ES and PWL

PUT = MAP in ES and PWL

AMP = PUT in FL for non charged engines

PUT = MAP in FL for charged engines

MAP is higher in ES than in IS

MAP is higher in FL than in IS

PUT is higher in FL than in ES for charged engines

PUT > AMP in FL for charged engines

PUT = AMP in PU/PUC for charged engines

For naturally aspirated engines, when the engine is standing or in full load, if the difference between the measured AMP and MAP values is above a certain threshold, a counter is started and incremented up to a certain maximum value. If this maximum is reached, a preliminary error flag, LV_SYM_AMP_MAP_PLAUS_ES and the corresponding error symptoms are set. However, the setting of the preliminary bit is not registered in the error memory. This is done in order to detect the exact origin of the deviation i.e. is the MAP sensor or AMP sensor faulty.

Once the preliminary error bit is set, the value of MAP_MES is checked when engine is in state idle or part load. If there is no change in the value of the measured manifold pressure compared to the pressure measured during engine stop or full load, it can be concluded that the MAP sensor is faulty. If however the value of MAP_MES does change, it is most probably a defective AMP sensor but it could also be due to a faulty(e.g. clogging up) air cleaner. In such a case, before the AMP sensor is replaced in the car, it should be made sure there is no clogging in the air cleaner.

For MAF-systems equipped with a MAP sensor the plausibility diagnosis for the manifold air pressure sensor is done as following:

At stopped engine or PWL, when there is ambient pressure everywhere, also at the throttle and in the intake manifold, it is monitored whether the measured manifold air pressure is within the maximum possible and the minimum possible ambient pressure (calibratable thresholds coming from the module ambient pressure adaptation).

When the engine is running the plausibility is checked between the MAP value measured from the MAP sensor and the MAP value calculated by the air path model (from the engine speed, the TPS value and the MAF sensor value). If the difference between these two values is over a calibratable threshold then it is supposed that the MAP sensor is faulty (assumed that the MAF sensor is all right).

If a system is not equipped with one of the pressure sensors, the function parts concerning this sensor are omitted in the software by means of compiler switches

Application Condition

Function Description

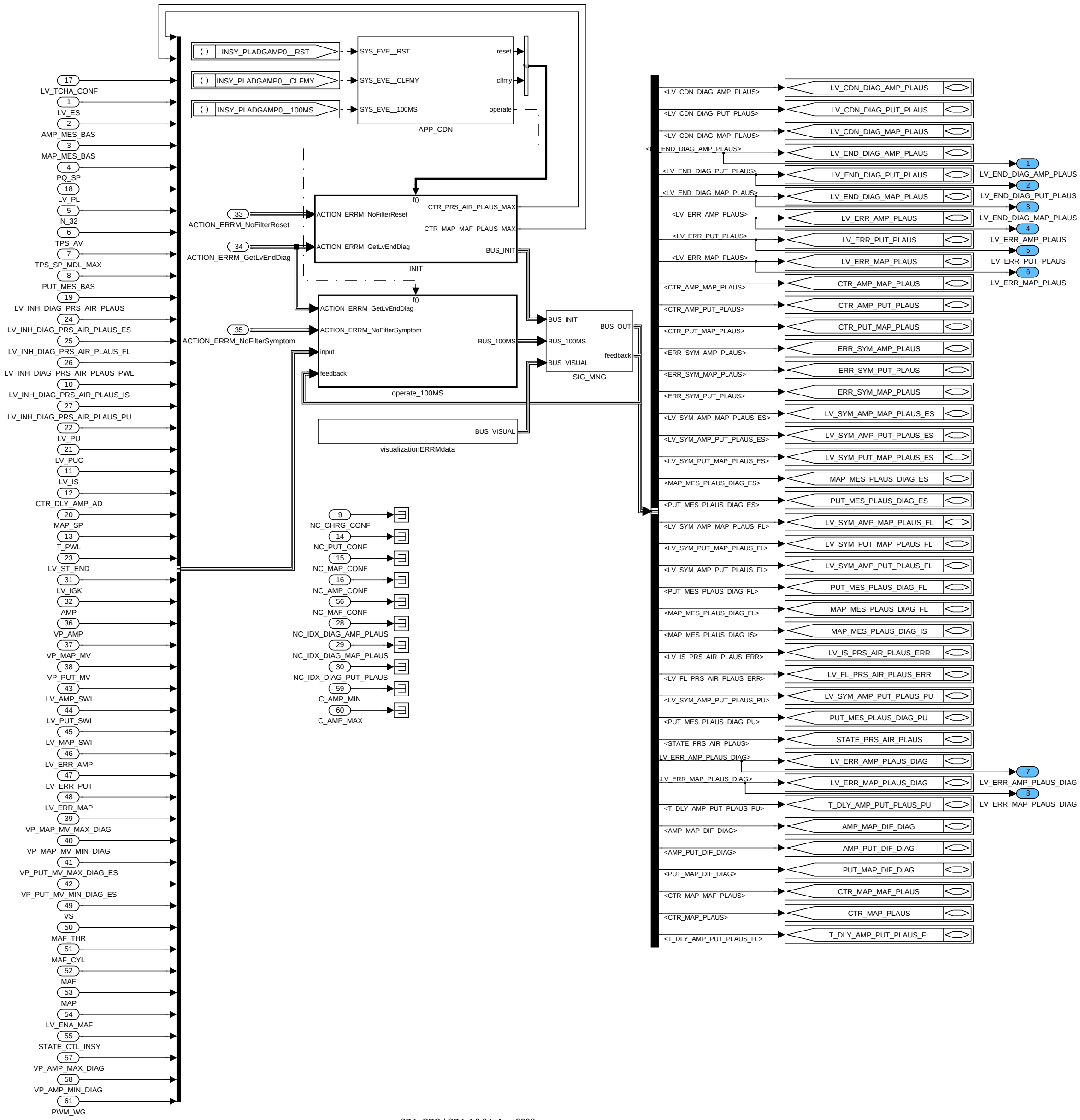


Figure 3 INSY_PLADGAMP0

B.34.1.1 Initialization at reset or clear failure memory:

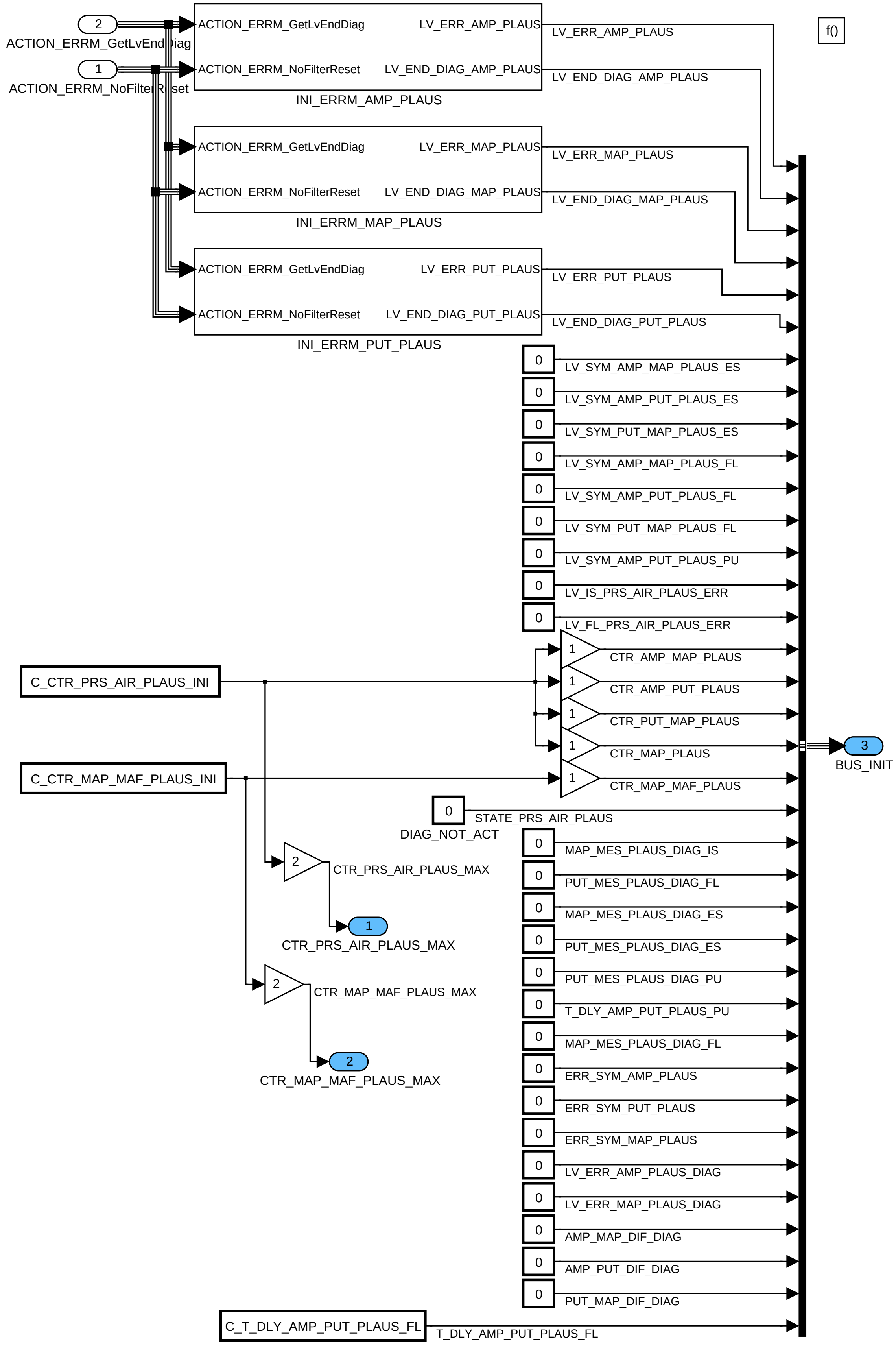

2
ACTION_ERRM_GetLvEndDiag
1
ACTION_ERRM_NoFilterReset
ACTION_ERRM_GetLvEndDiag
LV_ERR_AMP_PLAUS
ACTION_ERRM_NoFilterReset
LV_END_DIAG_AMP_PLAUS
INI_ERRM_AMP_PLAUS
ACTION_ERRM_GetLvEndDiag
LV_ERR_MAP_PLAUS
ACTION_ERRM_NoFilterReset
LV_END_DIAG_MAP_PLAUS
INI_ERRM_MAP_PLAUS
ACTION_ERRM_GetLvEndDiag
LV_ERR_PUT_PLAUS
ACTION_ERRM_NoFilterReset
LV_END_DIAG_PUT_PLAUS
INI_ERRM_PUT_PLAUS
f()
LV_ERR_AMP_PLAUS
LV_END_DIAG_AMP_PLAUS
LV_ERR_MAP_PLAUS
LV_END_DIAG_MAP_PLAUS
LV_ERR_PUT_PLAUS
LV_END_DIAG_PUT_PLAUS
LV_SYM_AMP_MAP_PLAUS_ES
LV_SYM_AMP_PUT_PLAUS_ES
LV_SYM_PUT_MAP_PLAUS_ES
LV_SYM_AMP_MAP_PLAUS_FL
LV_SYM_AMP_PUT_PLAUS_FL
LV_SYM_PUT_MAP_PLAUS_FL
LV_SYM_AMP_PUT_PLAUS_PU
LV_IS_PRS_AIR_PLAUS_ERR
LV_FL_PRS_AIR_PLAUS_ERR
C_CTR_PRS_AIR_PLAUS_INI
C_CTR_MAP_MAF_PLAUS_INI
CTR_AMP_MAP_PLAUS
CTR_AMP_PUT_PLAUS
CTR_PUT_MAP_PLAUS
CTR_MAP_PLAUS
CTR_MAP_MAF_PLAUS
3
BUS_INIT
0
DIAG_NOT_ACT
STATE_PRS_AIR_PLAUS
2
CTR_PRS_AIR_PLAUS_MAX
1
CTR_PRS_AIR_PLAUS_MAX
2
CTR_MAP_MAF_PLAUS_MAX
2
CTR_MAP_MAF_PLAUS_MAX
MAP_MES_PLAUS_DIAG_IS
PUT_MES_PLAUS_DIAG_FL
MAP_MES_PLAUS_DIAG_ES
PUT_MES_PLAUS_DIAG_ES
PUT_MES_PLAUS_DIAG_PU
T_DLY_AMP_PUT_PLAUS_PU
MAP_MES_PLAUS_DIAG_FL
ERR_SYM_AMP_PLAUS
ERR_SYM_PUT_PLAUS
ERR_SYM_MAP_PLAUS
LV_ERR_AMP_PLAUS_DIAG
LV_ERR_MAP_PLAUS_DIAG
AMP_MAP_DIF_DIAG
AMP_PUT_DIF_DIAG
PUT_MAP_DIF_DIAG
C_T_DLY_AMP_PUT_PLAUS_FL
T_DLY_AMP_PUT_PLAUS_FL

Figure 4 INSY_PLADGAMP0/ INIT

B.34.1.2 Recurrence: 100ms

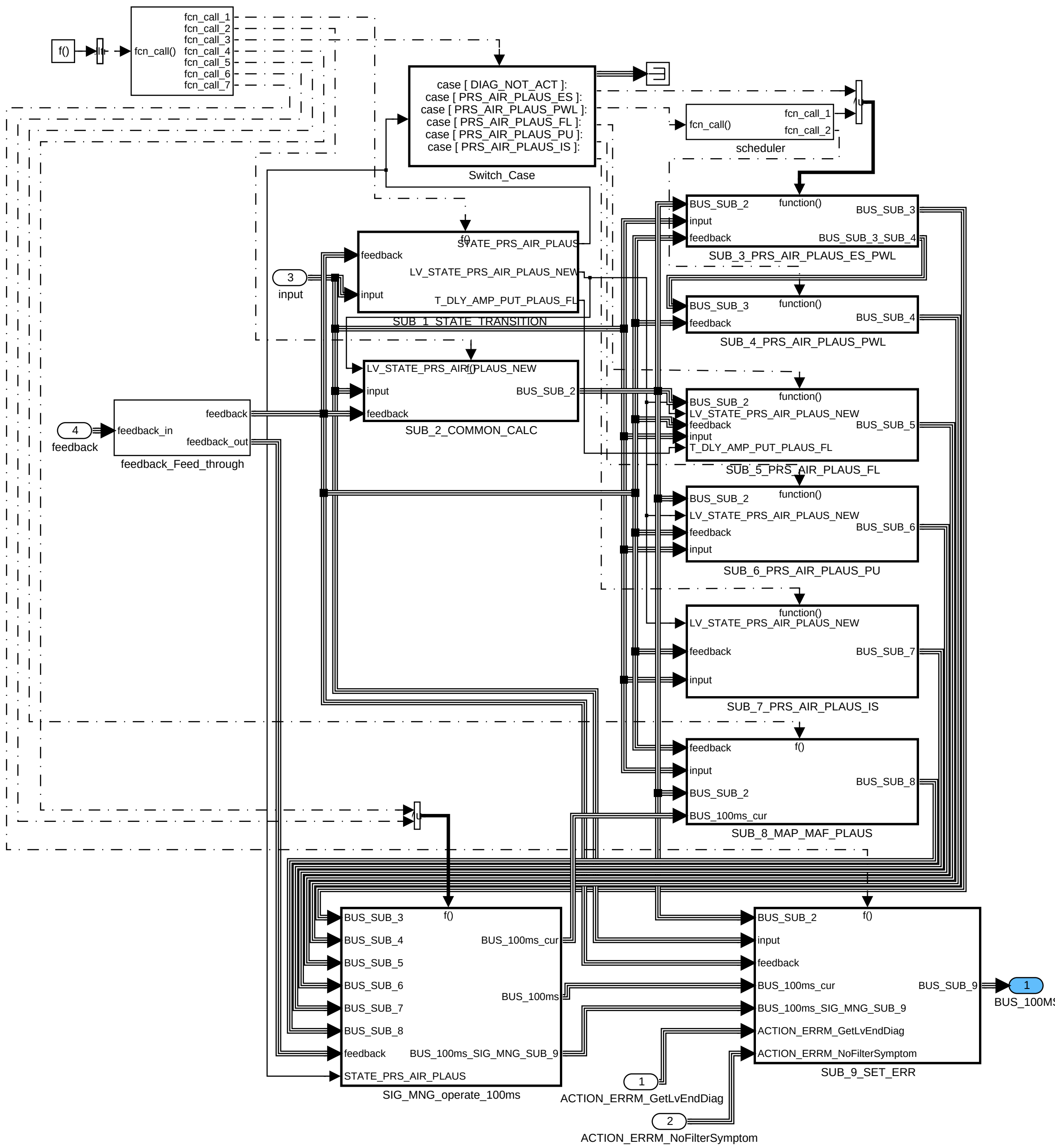


Figure 5 INSY_PLADGAMP0/ operate_100MS

<u>SUB_1: Transition between different states of the plausibility diagnosis:</u>

Parts of the plausibility diagnosis are only active if the engine is in its corresponding load range. The activation of the different states of the plausibility diagnosis and the transition between them is done in this subsystem.

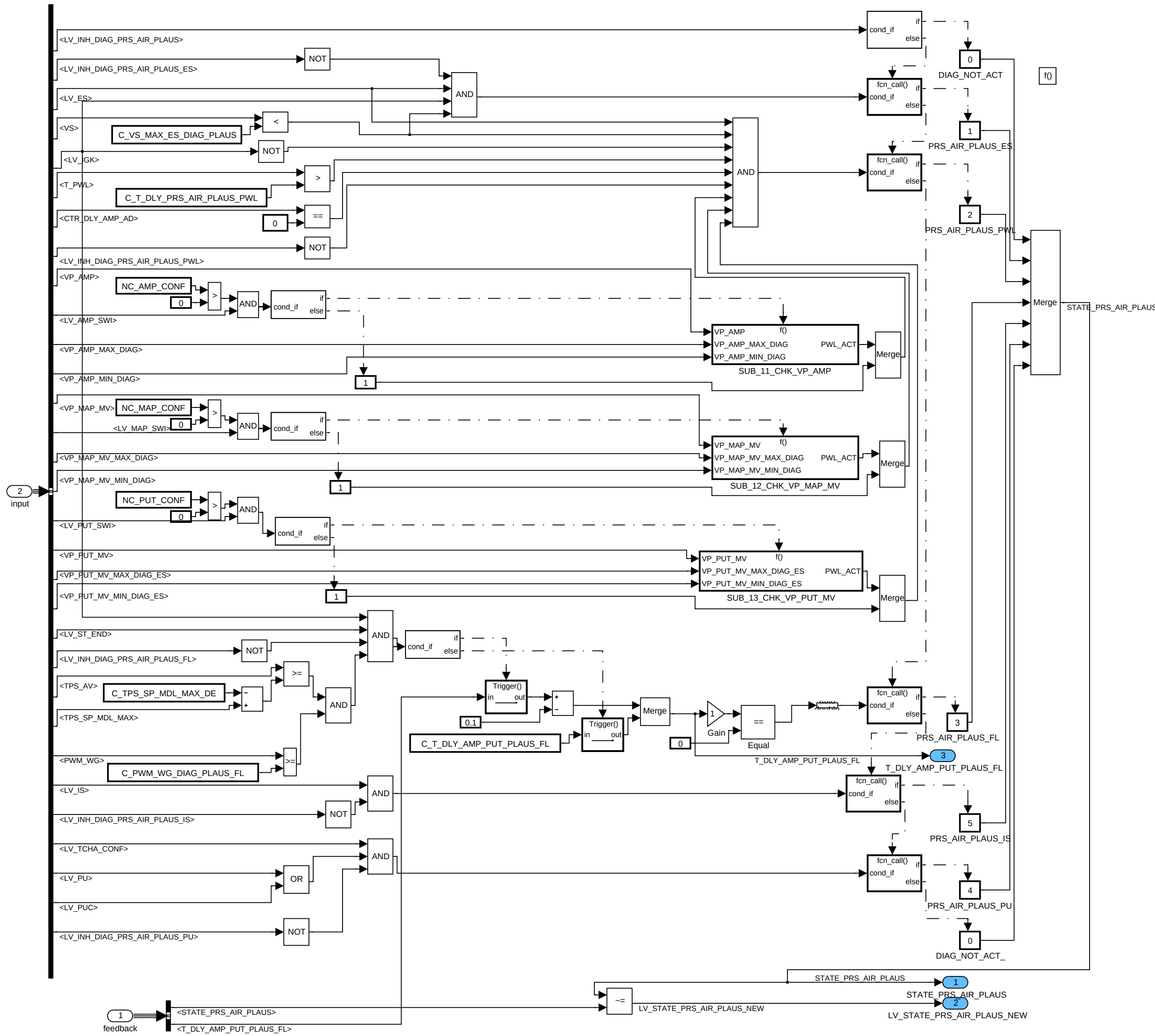


Figure 6 INSY_PLADGAMP0/ operate_100MS/ SUB_1_STATE_TRANSITION

SUB_11: Check whether VP_AMP is within his thresholds:

At Power latch phase the electrical sensor diagnosis is not working any more, so the sensor voltage has to be monitored here again separately.

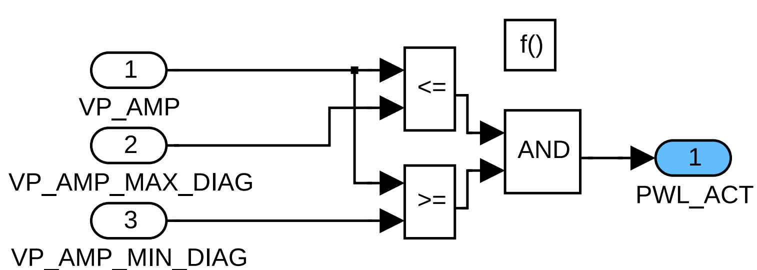


Figure 7 INSY_PLADGAMP0/ operate_100MS/ SUB_1_STATE_TRANSITION/ SUB_11_CHK_VP_AMP

SUB_12: Check whether VP_MAP_MV is within his thresholds:

At Power latch phase the electrical sensor diagnosis is not working any more, so the sensor voltage has to be monitored here again separately.

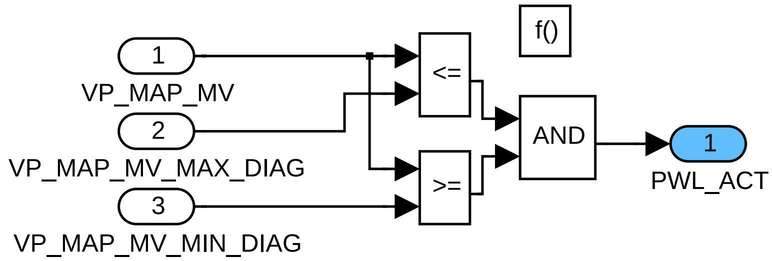


Figure 8 INSY_PLADGAMP0/ operate_100MS/ SUB_1_STATE_TRANSITION/ SUB_12_CHK_VP_MAP_MV

SUB_13: Check whether VP_PUT_MV is within his thresholds:

At Power latch phase the electrical sensor diagnosis is not working any more, so the sensor voltage has to be monitored here again separately.

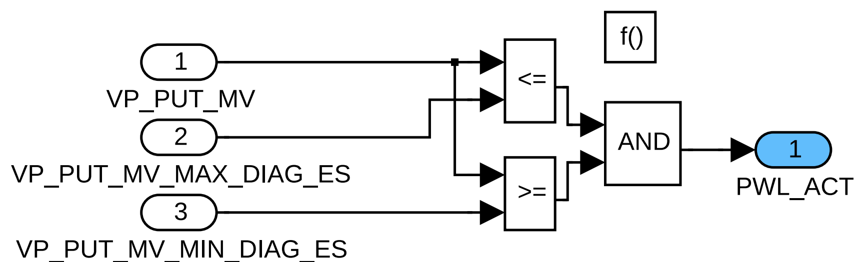


Figure 9 INSY_PLADGAMP0/ operate_100MS/ SUB_1_STATE_TRANSITION/ SUB_13_CHK_VP_PUT_MV

SUB_2: Calculations of variables which are used in every state of the diagnosis (previous calculation):

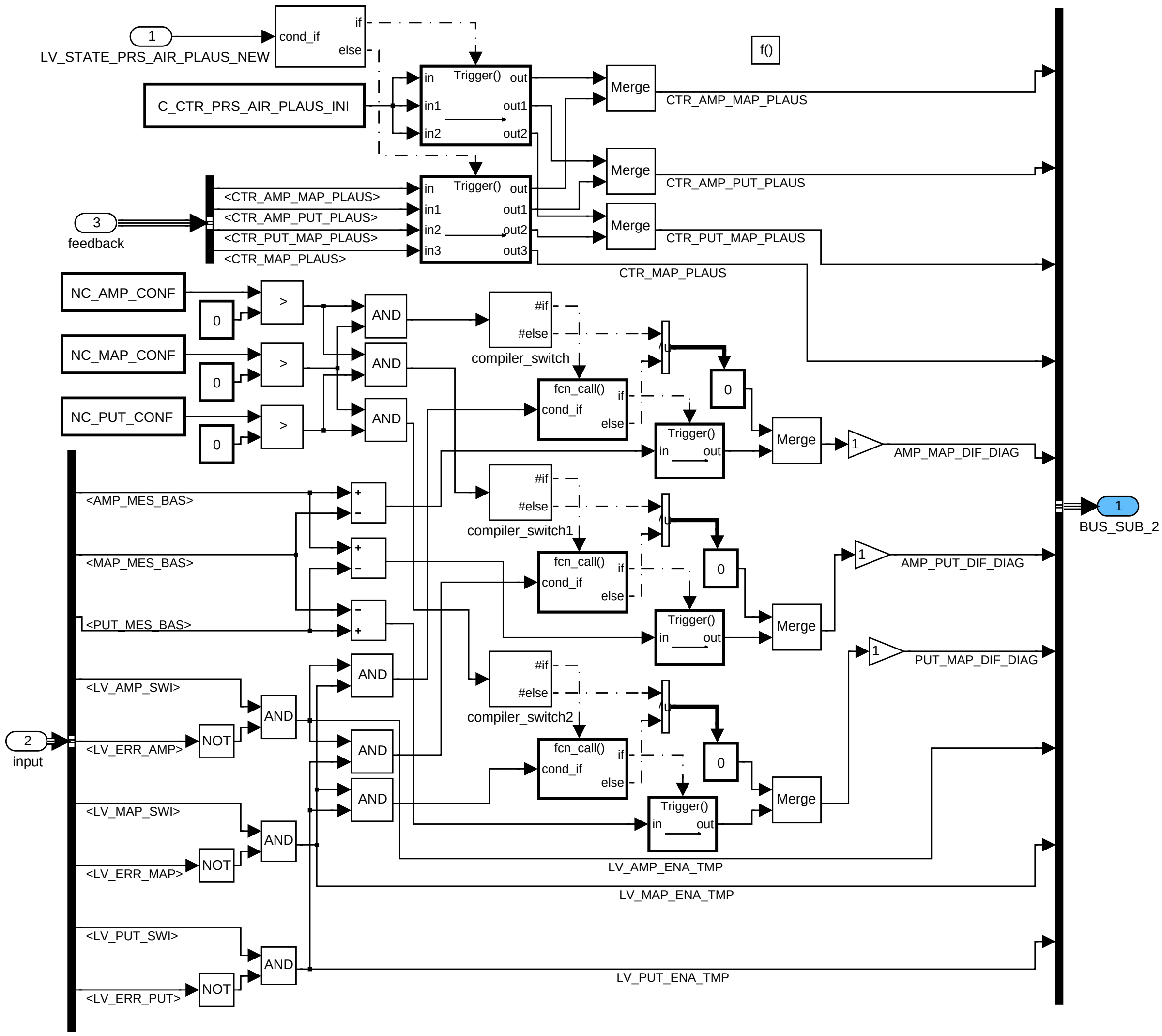


Figure 10 INSY_PLADGAMP0/ operate_100MS/ SUB_2_COMMON_CALC

SWITCH_CASE

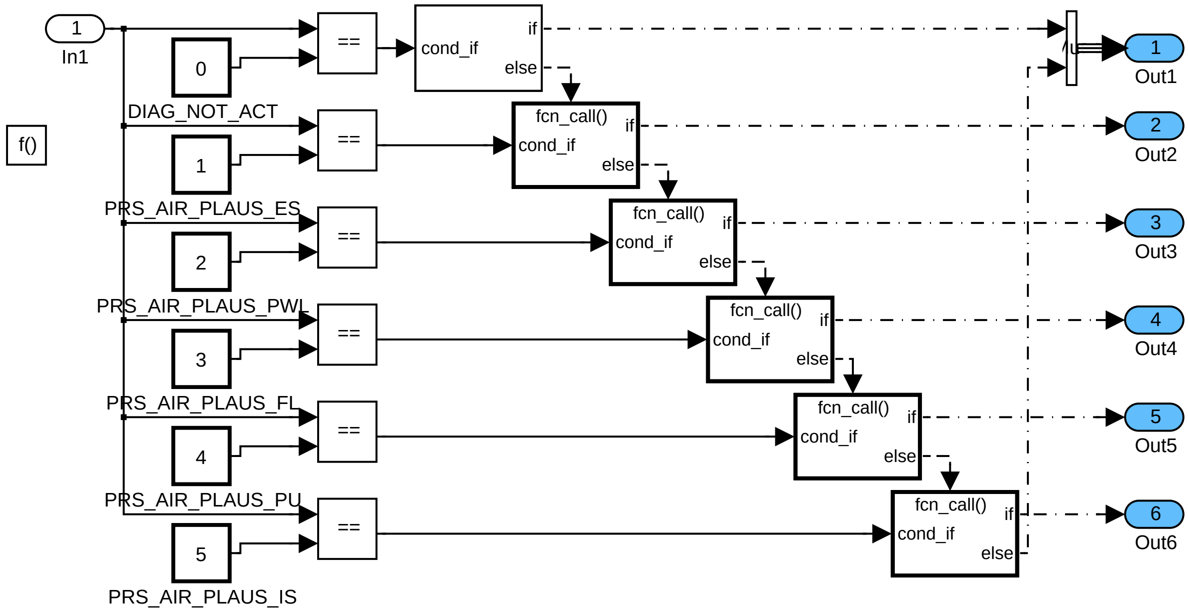


Figure 11 INSY_PLADGAMP0/ operate_100MS/ Switch_Case

SUB_3: Overview over all possible pressure sensor plausibilities in ES or PWL:

For systems equipped with more air pressure sensors the plausibility diagnosis is done by comparing the measured pressures at stopped engine, when there is ambient pressure everywhere, also at the throttle and in the intake manifold. If two sensors show the same pressure and a third differs significantly, the third sensor is implausible.

If there are only two pressure sensors or if the implausibility occurs in an engine state when only two pressures are equal or close to each other, then only an implausible pair on sensors can be pointed to. In another engine state it is tried to find out, which of the two sensors is faulty.

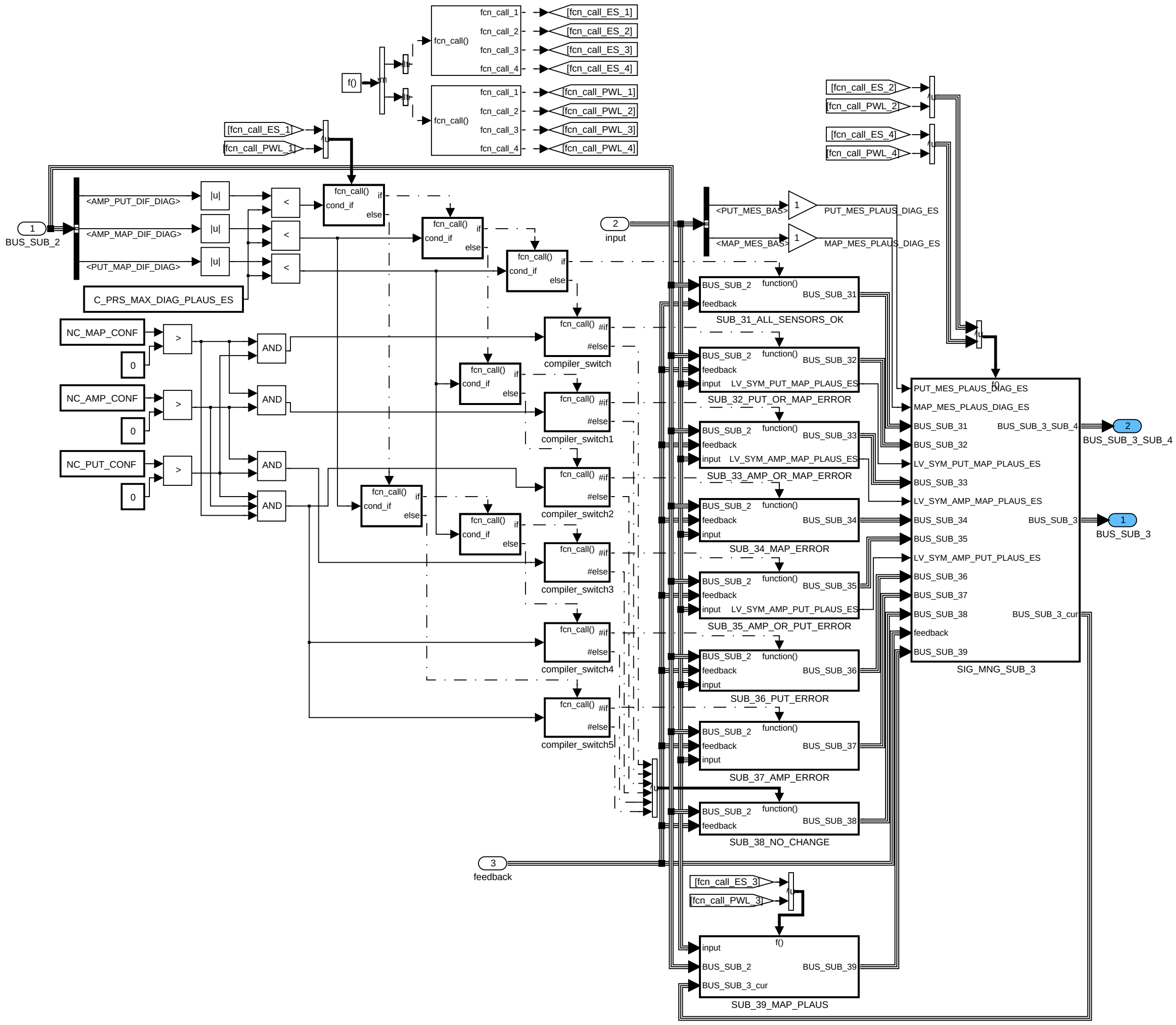


Figure 12 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL

SUB_31: All available pressure sensors are O.K.:

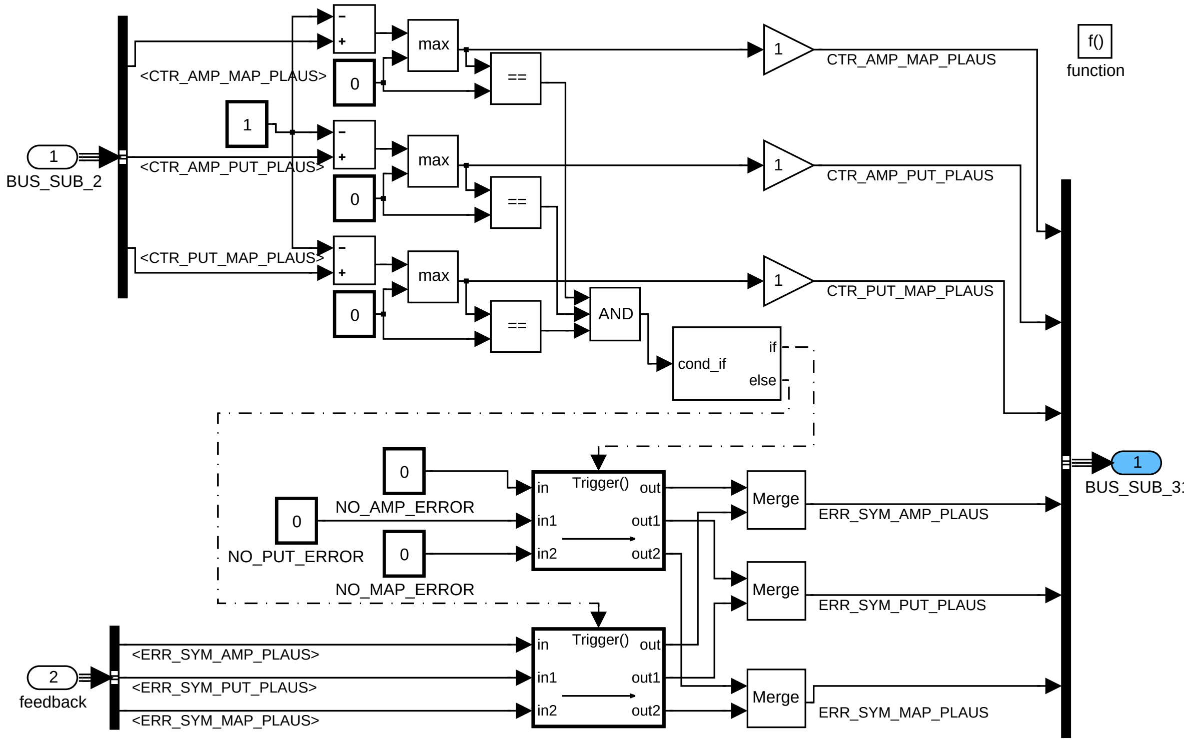


Figure 13 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_31_ALL_SENSORS_OK

SUB_32: Detection that PUT-sensor or MAP-sensor is defective:

Only these two pressure sensors are available. A difference between the two pressures is detected in the current state ES or PWL. If you have been in state idle speed before entering the current state ES or PWL then the defective sensor can be pointed out and the corresponding error symptom is set. But if you haven´t been in state idle speed before, only an implausible pair of sensors can be pointed out and a preliminary flag is set: LV_SYM_PUT_MAP_PLAUS_ES. Another engine state is necessary to point out which sensor is defective.

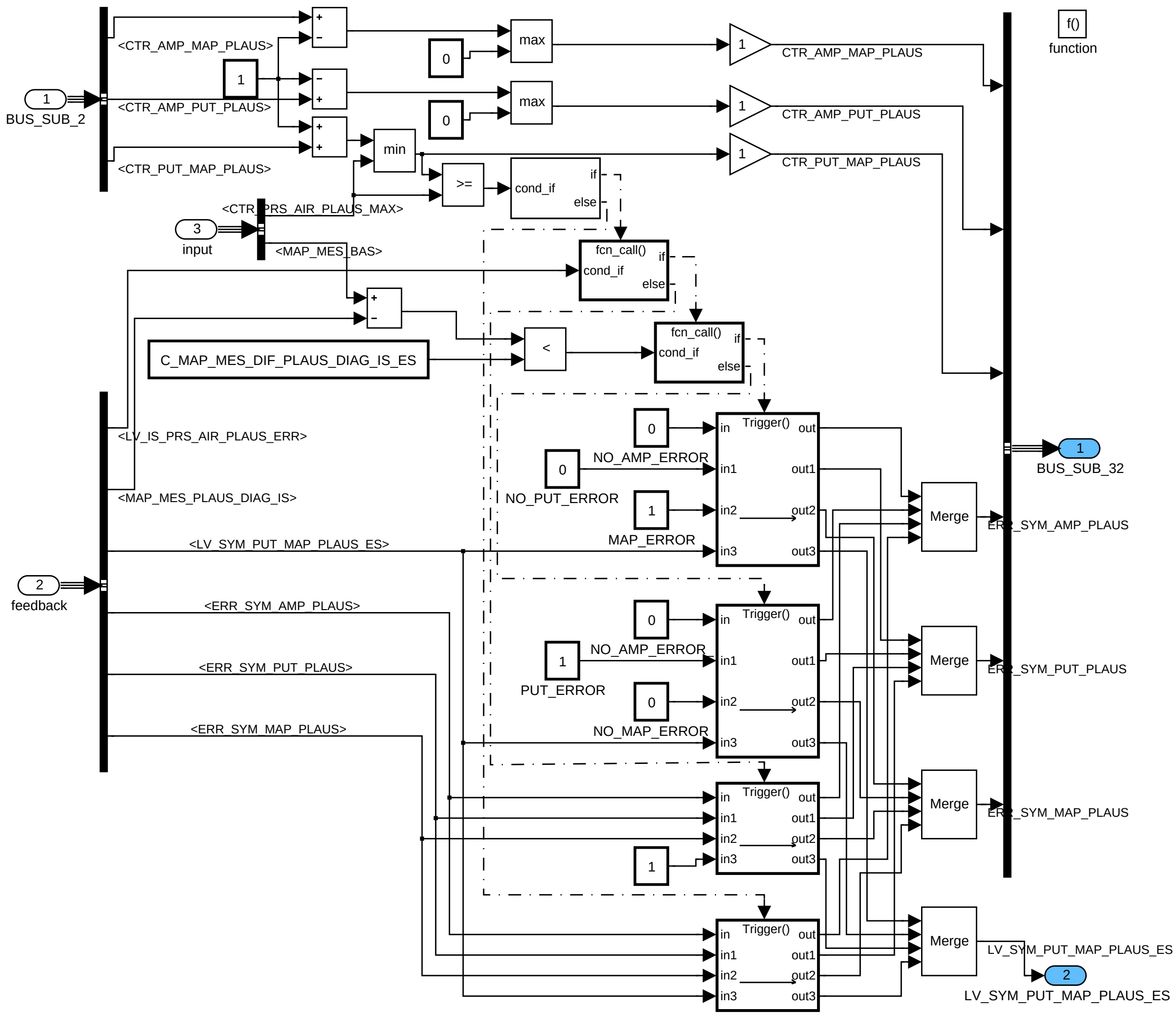


Figure 14 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_32_PUT_OR_MAP_ERROR

SUB_33: Detection that AMP-sensor or MAP-sensor is defective:

Only these two pressure sensors are available. A difference between the two pressures is detected in the current state ES or PWL. If you have been in state idle speed before entering the current state ES or PWL then the defective sensor can be pointed out and the corresponding error symptom is set. But if you haven´t been in state idle speed before, only an implausible pair of sensors can be pointed out and a preliminary flag is set: LV_SYM_AMP_MAP_PLAUS_ES. Another engine state is necessary to point out which sensor is defective.

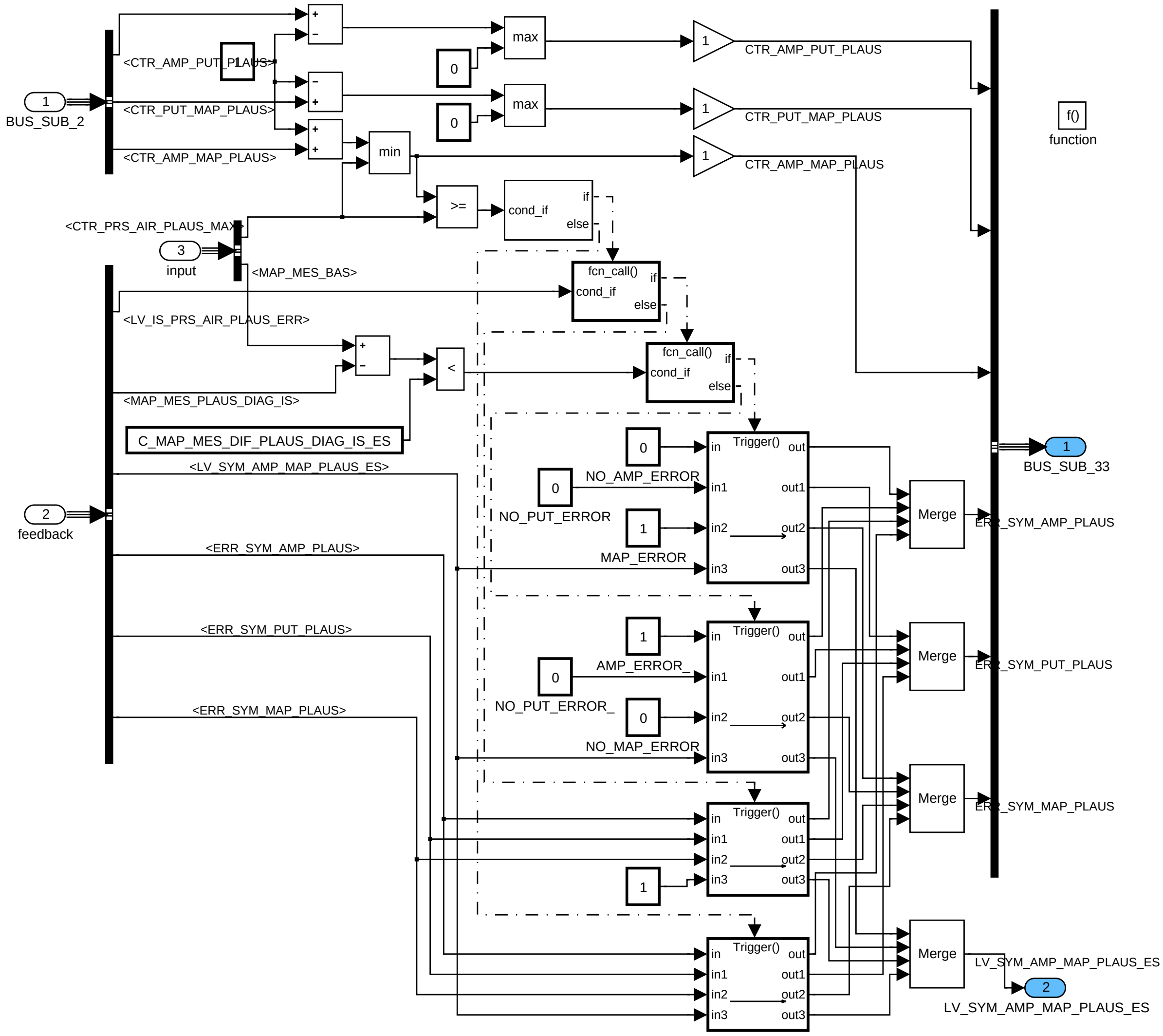


Figure 15 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_33_AMP_OR_MAP_ERROR

SUB_34: Detection that MAP-sensor is defective:

Three pressure sensors are available. The AMP-sensor and the PUT-sensor show the same pressure in the current state ES or PWL and the third sensor (MAP) differs significantly. So the MAP-sensor is supposed to be defective and the corresponding error symptom is set.

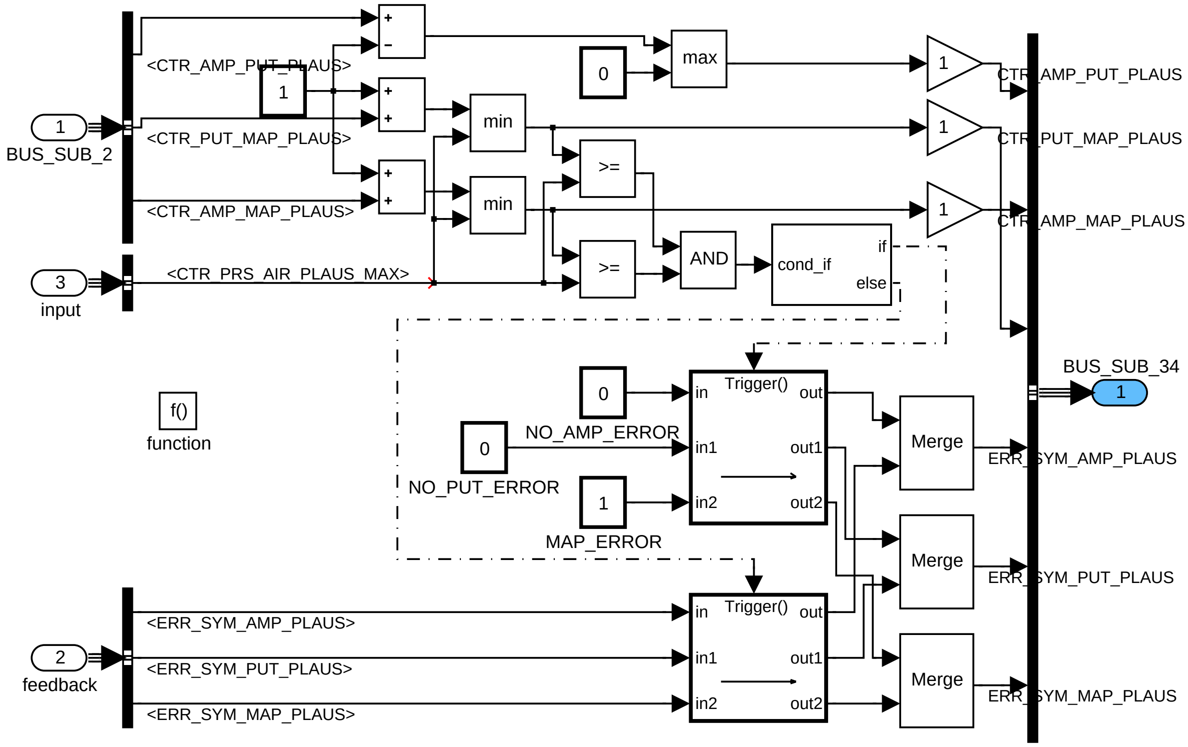


Figure 16 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_34_MAP_ERROR

SUB_35: Detection that AMP-sensor or PUT-sensor is defective:

Only these two pressure sensors are available. A difference between the two pressures is detected in the current state ES or PWL. If you have been in state full load before entering the current state ES or PWL then the defective sensor can be pointed out and the corresponding error symptom is set. But if you haven´t been in state full load before, only an implausible pair of sensors can be pointed out and a preliminary flag is set: LV_SYM_AMP_PUT_PLAUS_ES. Another engine state is necessary to point out which sensor is defective

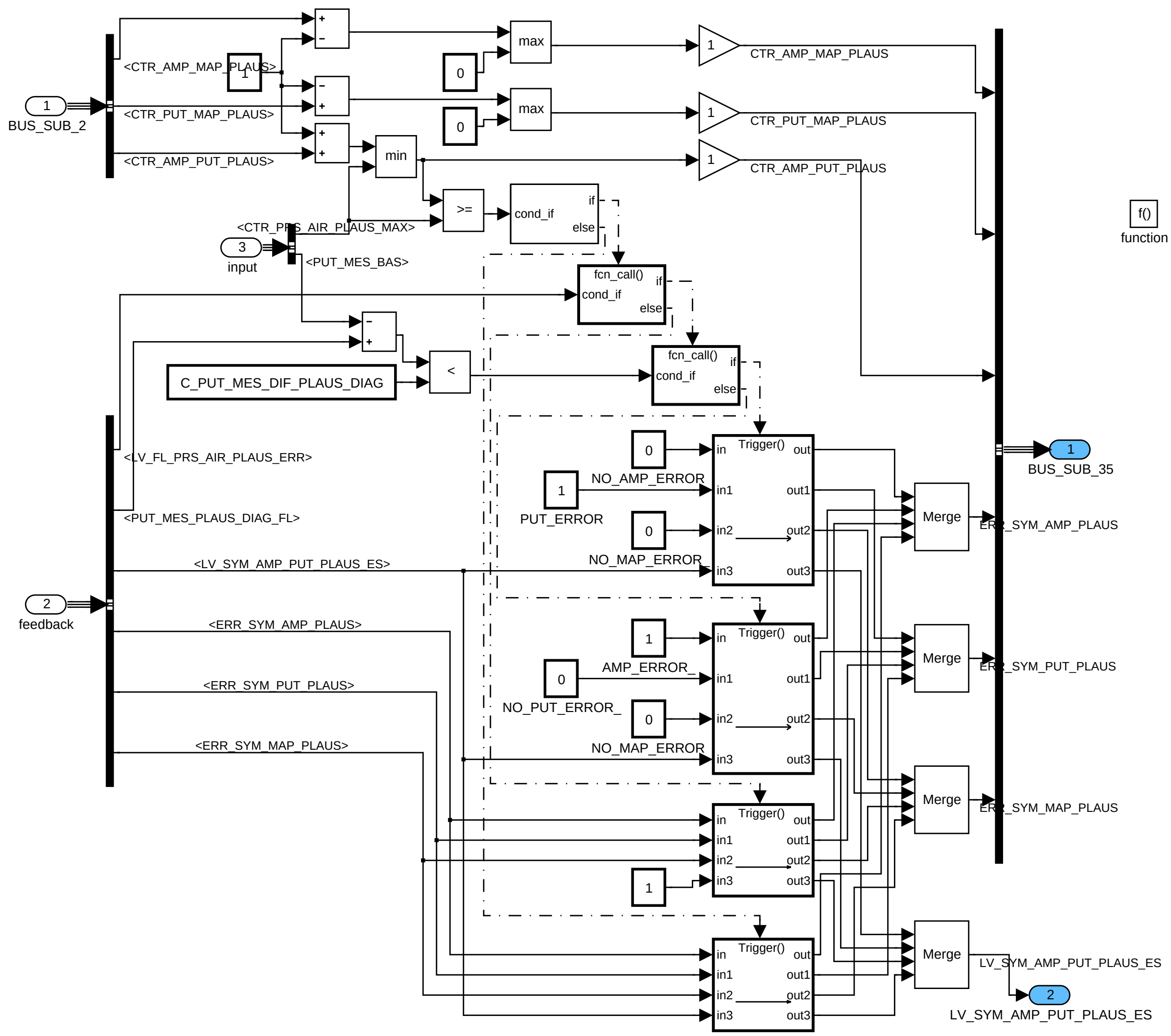


Figure 17 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_35_AMP_OR_PUT_ERROR

<u>SUB_36: Detection that PUT-sensor is defective:</u>

Three pressure sensors are available. The AMP-sensor and the MAP-sensor show the same pressure in the current state ES or PWL and the third sensor (PUT) differs significantly. So the PUT-sensor is supposed to be defective and the corresponding error symptom is set.

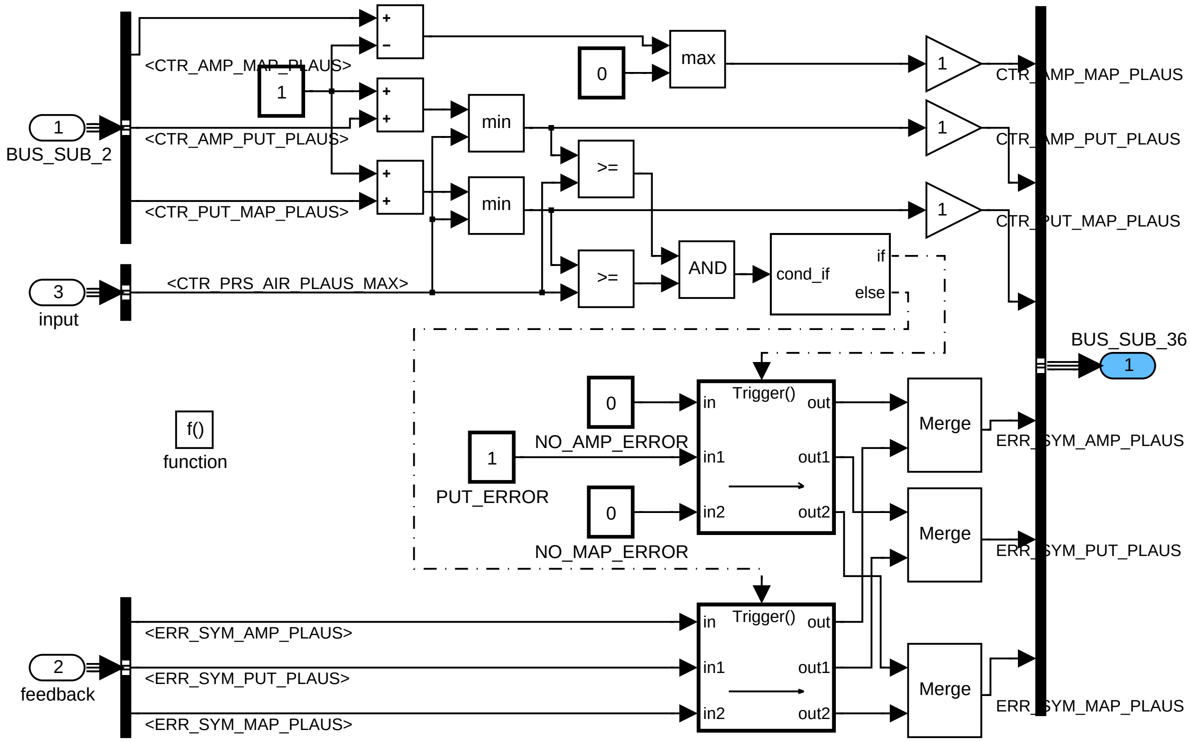


Figure 18 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_36_PUT_ERROR

SUB_37: Detection that AMP-sensor is defective:

Three pressure sensors are available. The PUT-sensor and the MAP-sensor show the same pressure in the current state ES or PWL and the third sensor (AMP) differs significantly. So the AMP-sensor is supposed to be defective and the corresponding error symptom is set.

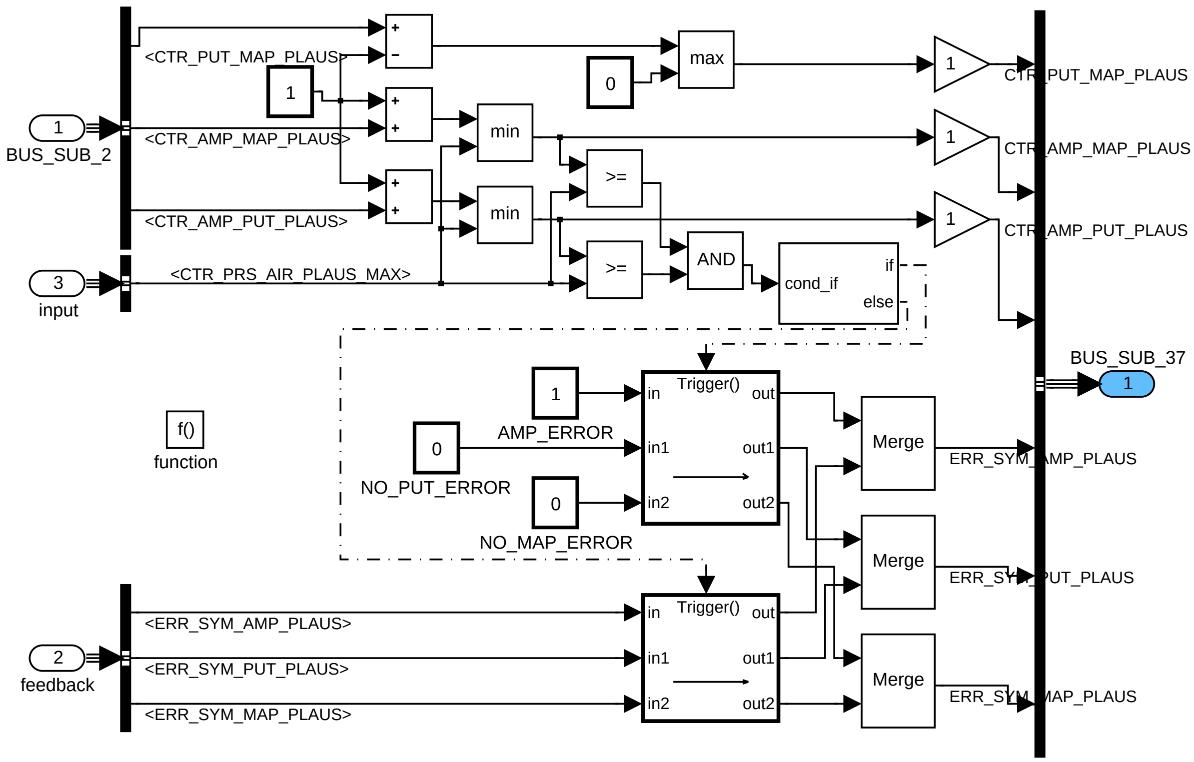


Figure 19 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_37_AMP_ERROR

<u>SUB_38: No change of error symptoms requested in state ES or PWL:</u>

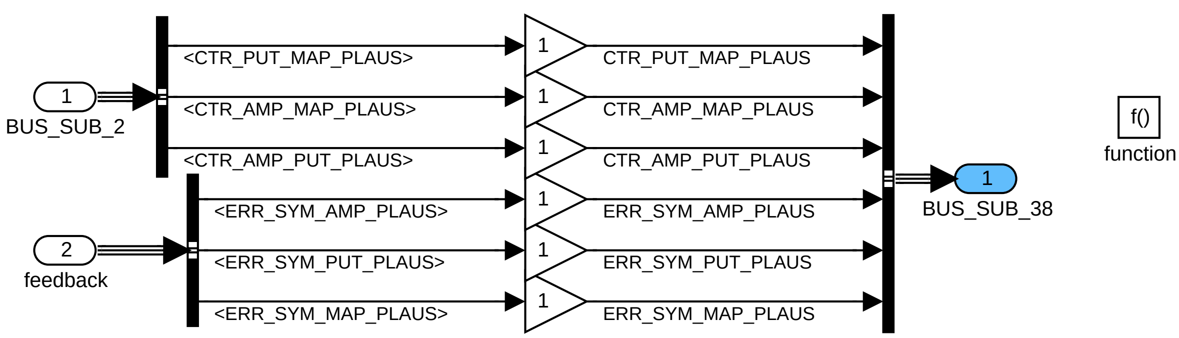


Figure 20 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_38_NO_CHANGE

<u>SUB_39: MAP plausibility for MAF-systems equipped with a MAP-sensor:</u>

<u>SUB_39: MAP plausibility for MAF-systems equipped with a MAP-sensor (at state ES or PWL):</u>

At engine standing or at PWL it is monitored whether the measured manifold air pressure is within the maximum possible and the minimum possible ambient pressure. If not then the corresponding error symptom is set.

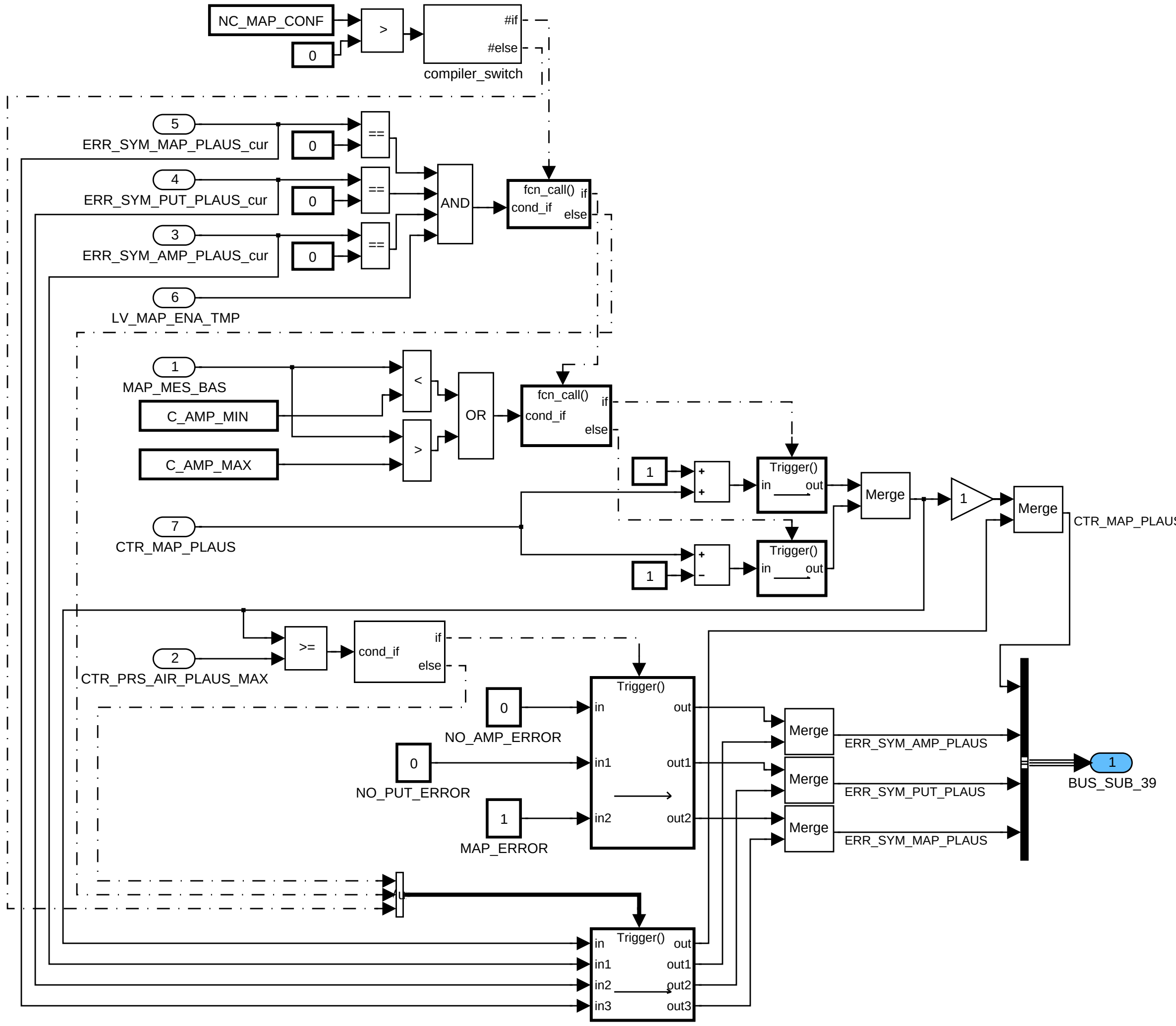


Figure 21 INSY_PLADGAMP0/ operate_100MS/ SUB_3_PRS_AIR_PLAUS_ES_PWL/ SUB_39_MAP_PLAUS/ SUB_39_MAP_PLAUS

<u>SUB_4: Implausible pair of pressure sensors (detected in PWL) registered in error management:</u>

Depending on LC_PRS_AIR_SENS_2_ERR_SAVE it can be decided whether an implausible pair of pressure sensors, that was detected in state PWL, should be registered in the error management and an LV_ERR should be set or not.

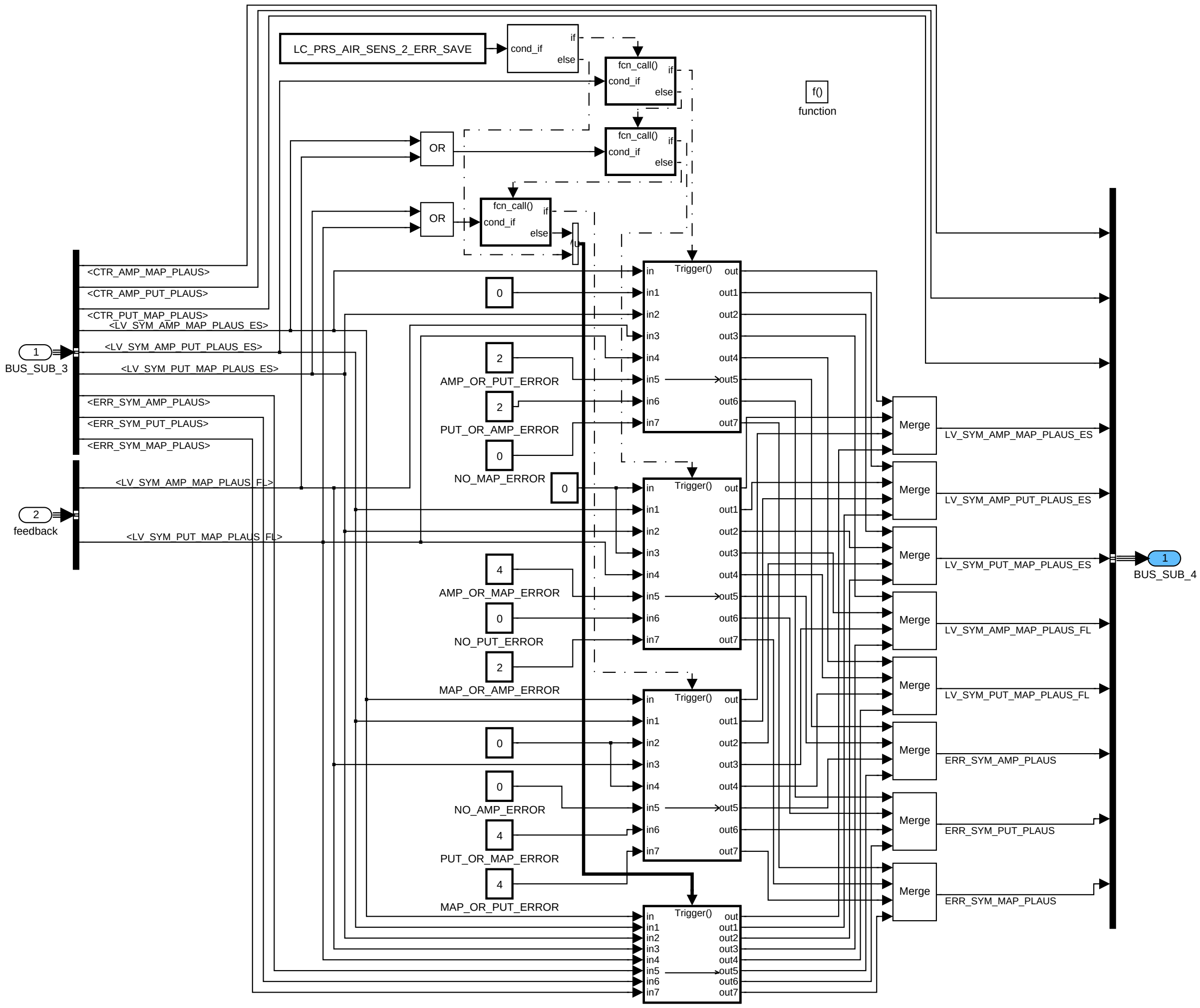


Figure 22 INSY_PLADGAMP0/ operate_100MS/ SUB_4_PRS_AIR_PLAUS_PWL

SUB_5: Overview over all possible pressure sensor plausibilities in FL:

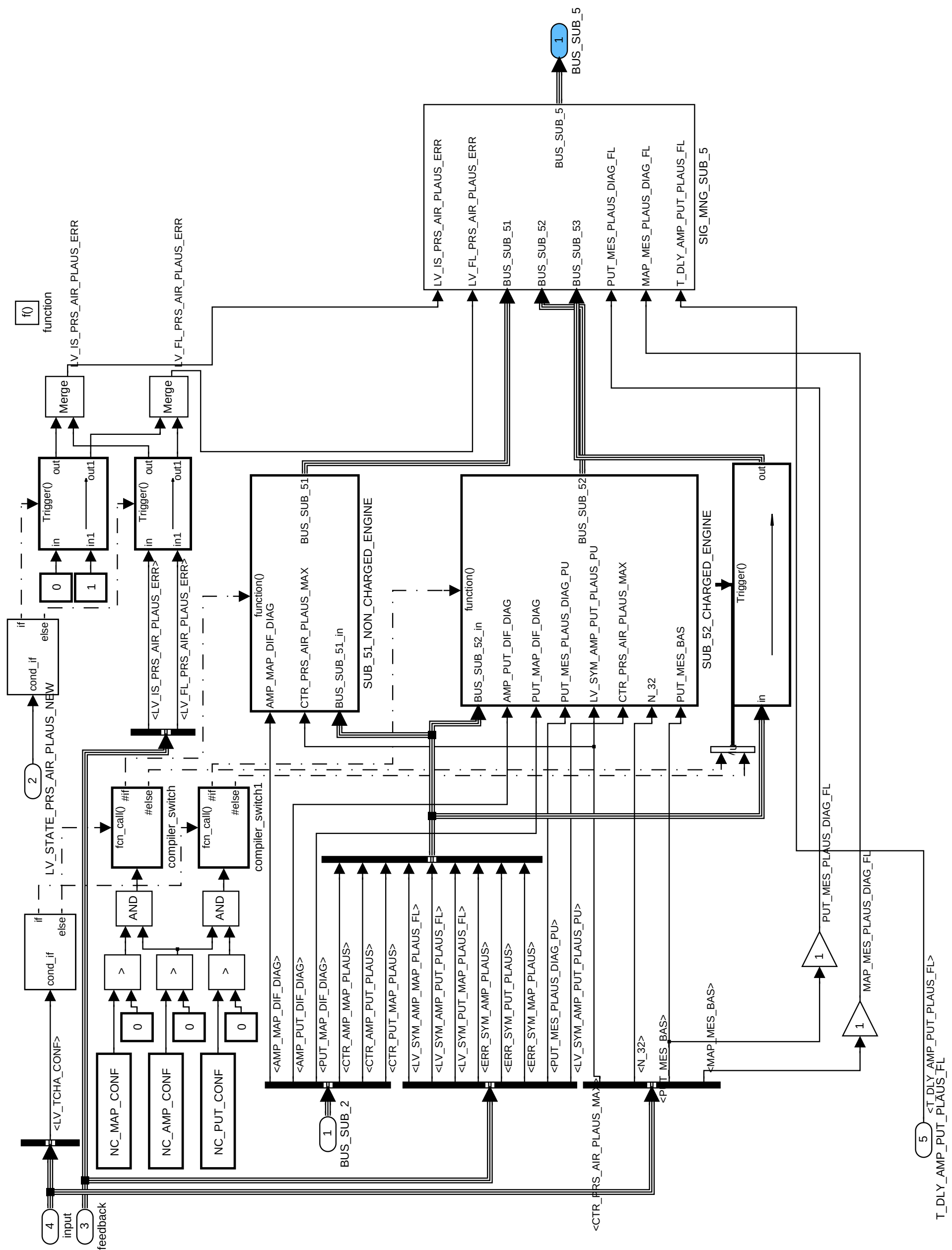


Figure 23 INSY_PLADGAMP0/ operate_100MS/ SUB_5_PRS_AIR_PLAUS_FL

SUB_51: Detection of implausible pressure sensors at FL for non charged engines:

At full load there is ambient pressure at the throttle and also nearly in the intake manifold for non charged engines, so the AMP-sensor and the MAP-sensor should show nearly the same pressure. If not, a preliminary flag is set LV_SYM_AMP_MAP_PLAUS_FL.

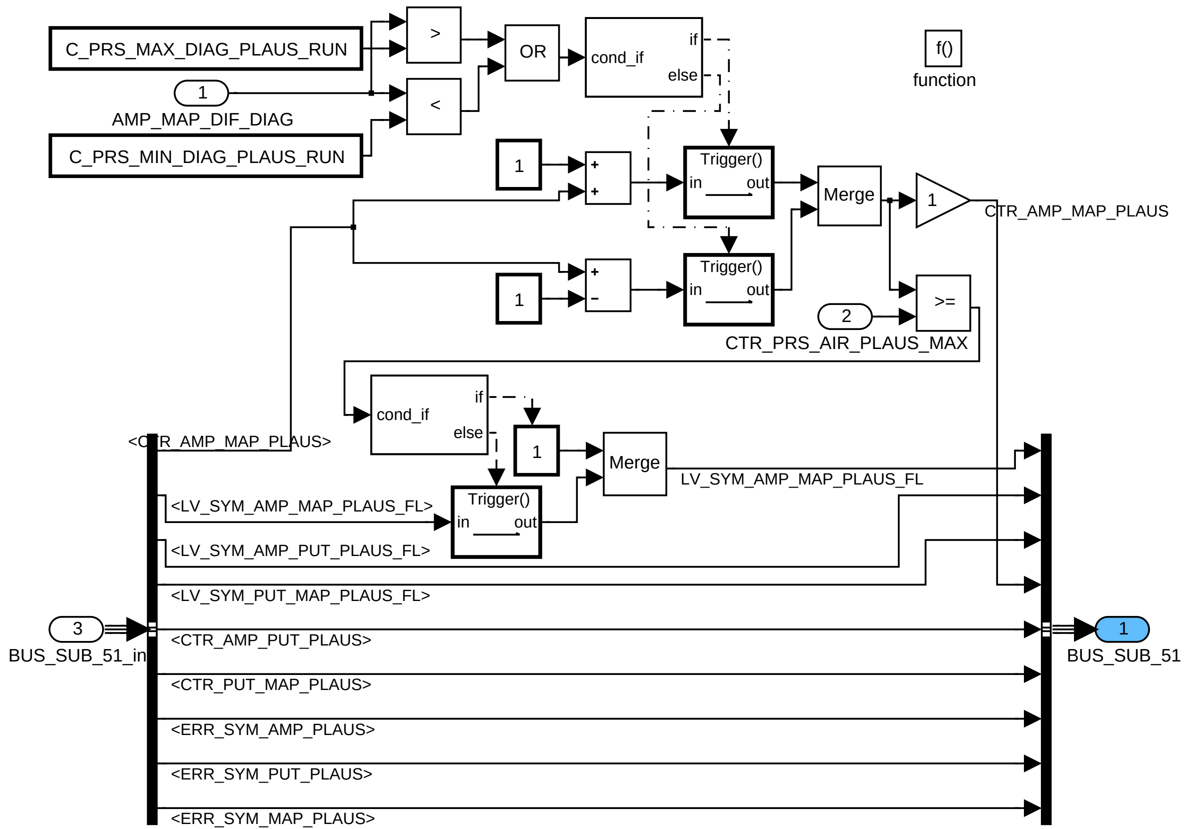


Figure 24 INSY_PLADGAMP0/ operate_100MS/ SUB_5_PRS_AIR_PLAUS_FL/ SUB_51_NON_CHARGED_ENGINE

SUB_52: Detection of implausible pressure sensors at FL for charged engines:

At full load for charged engines, the pressure upstream throttle is equal to the intake manifold pressure, so the PUT-sensor and the MAP-sensor should show nearly the same pressure. If not, a preliminary flag is set LV_SYM_PUT_MAP_PLAUS_FL.

The pressure upstream throttle is bigger than the ambient pressure for charged engines at full load, so the PUT-sensor and the AMP-sensor should show a certain pressure difference. If the difference is small and you have been in state pull before then the defective sensor can be pointed out and the corresponding error symptom is set. But if you haven´t been in state pull before, only an implausible pair of sensors can be pointed out and a preliminary flag is set: LV_SYM_AMP_PUT_PLAUS_FL. Another engine state is necessary to point out which sensor is defective.

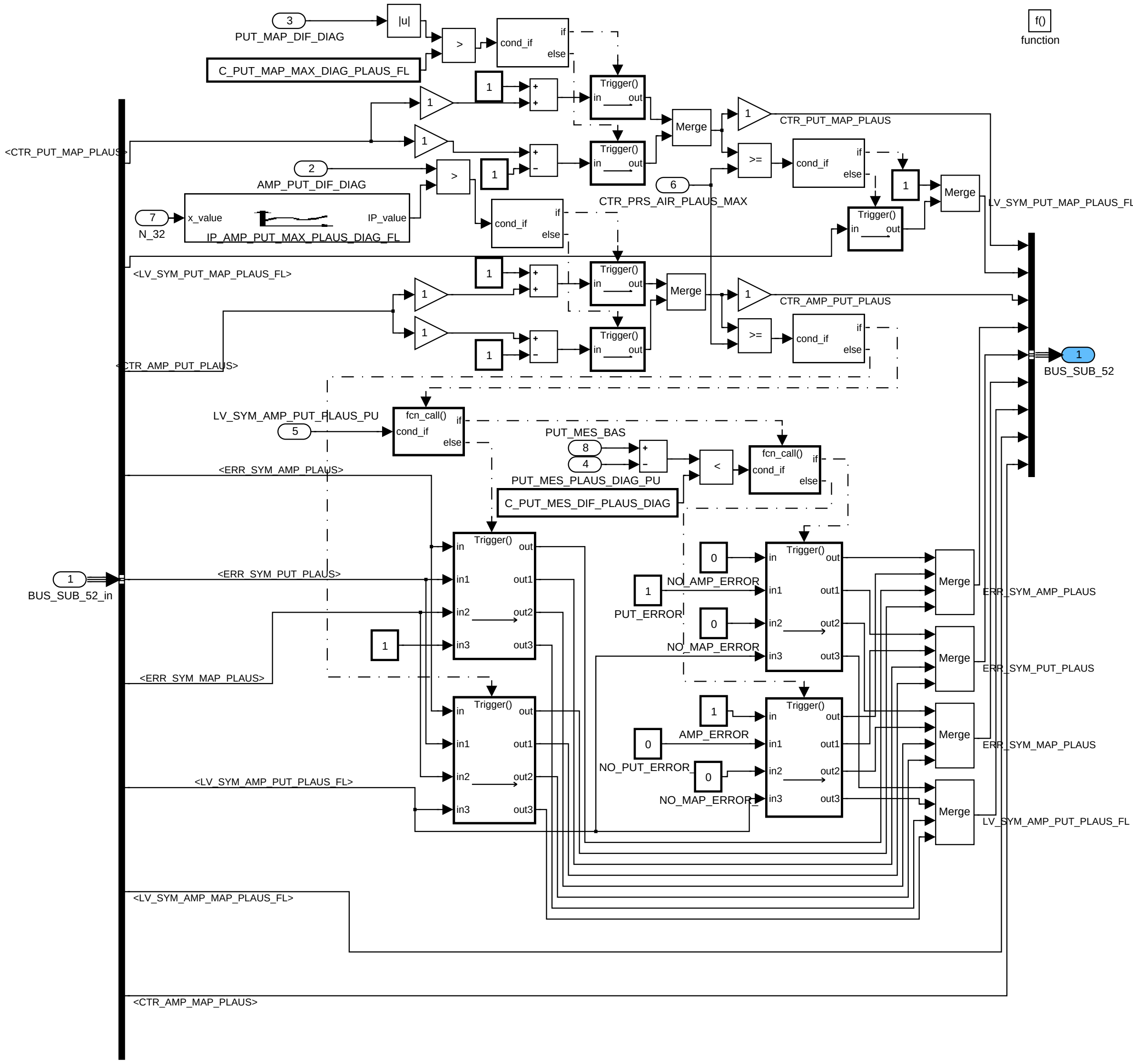


Figure 25 INSY_PLADGAMP0/ operate_100MS/ SUB_5_PRS_AIR_PLAUS_FL/ SUB_52_CHARGED_ENGINE

SUB_6: Overview over all possible pressure sensor plausibilities in engine state pull:

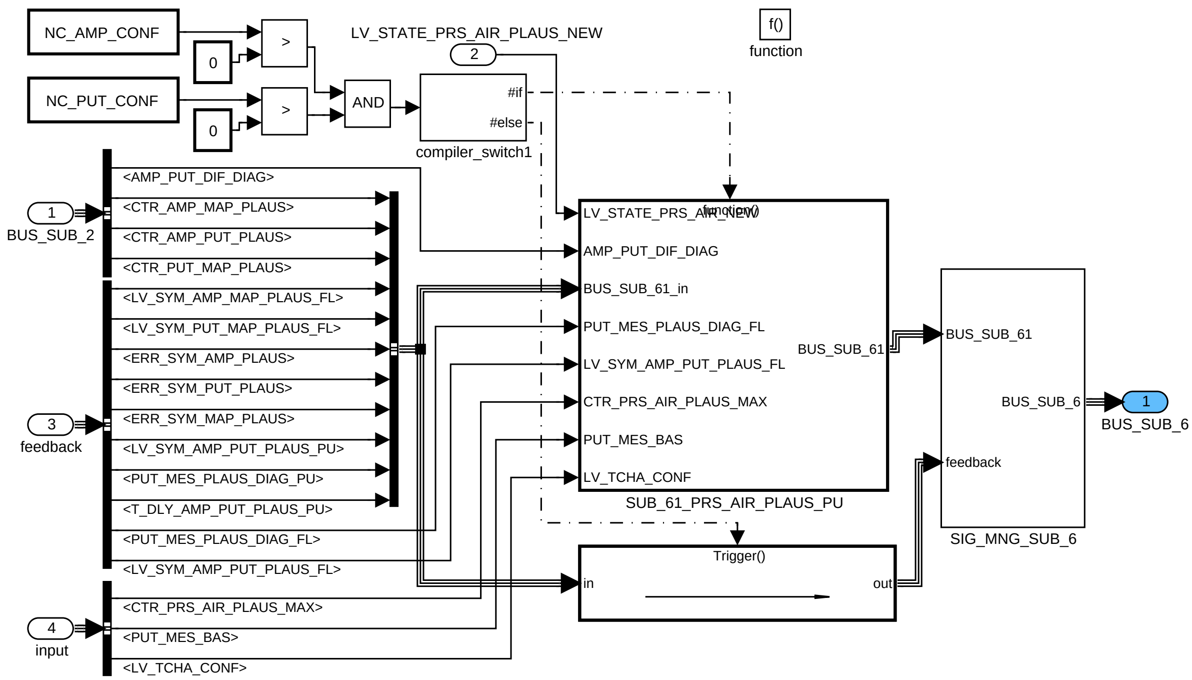


Figure 26 INSY_PLADGAMP0/ operate_100MS/ SUB_6_PRS_AIR_PLAUS_PU

SUB_61: Detection of implausible pressure sensors in engine state pull:

The pressure upstream throttle is equal to the ambient pressure in engine state pull, so the PUT-sensor and the AMP-sensor should show nearly the same pressure. If there is a difference and you have been in state full load before then the defective sensor can be pointed out and the corresponding error symptom is set. But if you haven´t been in state full load before, only an implausible pair of sensors can be pointed out and a preliminary flag is set: LV_SYM_AMP_PUT_PLAUS_PU. Another engine state is necessary to point out which sensor is defective.

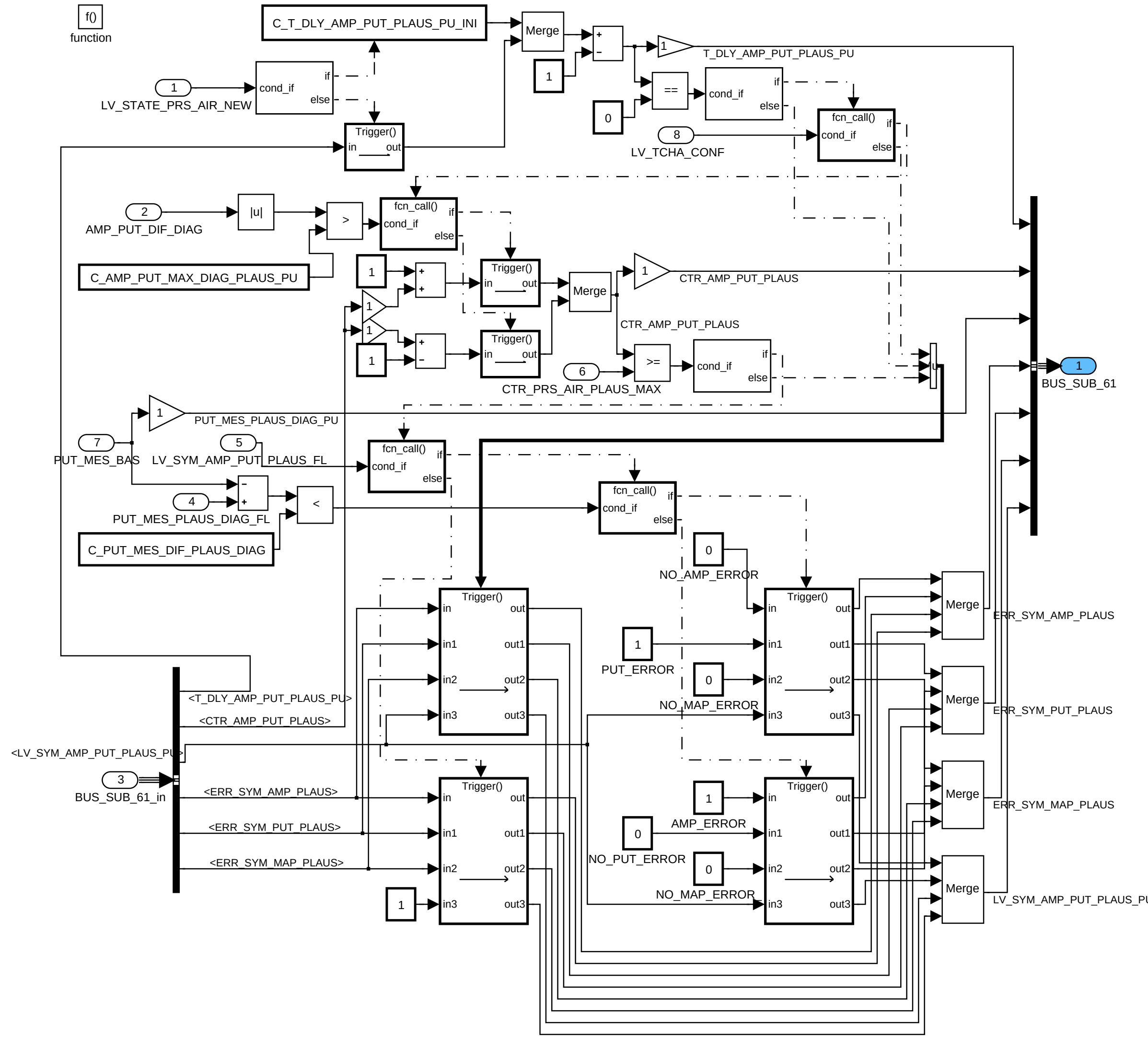


Figure 27 INSY_PLADGAMP0/ operate_100MS/ SUB_6_PRS_AIR_PLAUS_PU/ SUB_61_PRS_AIR_PLAUS_PU

SUB_7: Overview over all possible pressure sensor plausibilities in engine state idle speed:

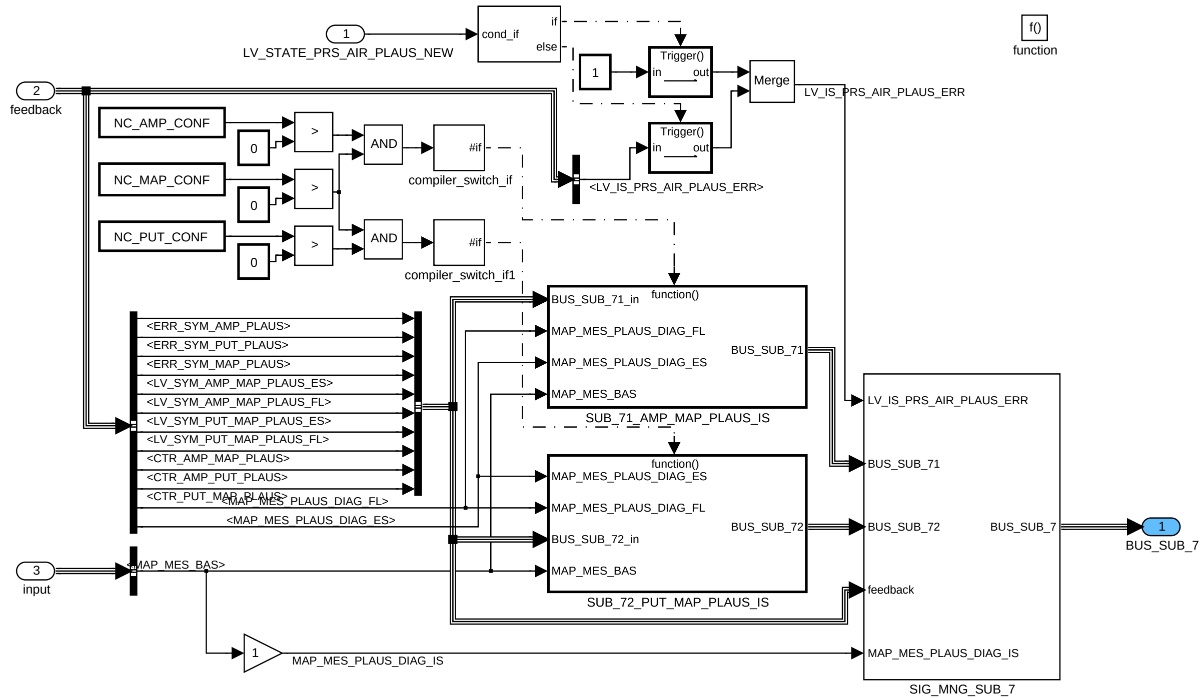


Figure 28 INSY_PLADGAMP0/ operate_100MS/ SUB_7_PRS_AIR_PLAUS_IS

SUB_71: Detection of defective AMP-sensor or MAP-sensor in engine state idle speed:

If the preliminary error bit LV_SYM_AMP_MAP_PLAUS_ES or LV_SYM_AMP_MAP_PLAUS_FL is set, the value of MAP_MES is checked when engine is in state idle or part load. If there is no change in the value of the measured manifold pressure compared to the pressure measured during engine stop or full load, it can be concluded that the MAP sensor is faulty. If however the value of MAP_MES does change, it is most probably a defective AMP sensor.

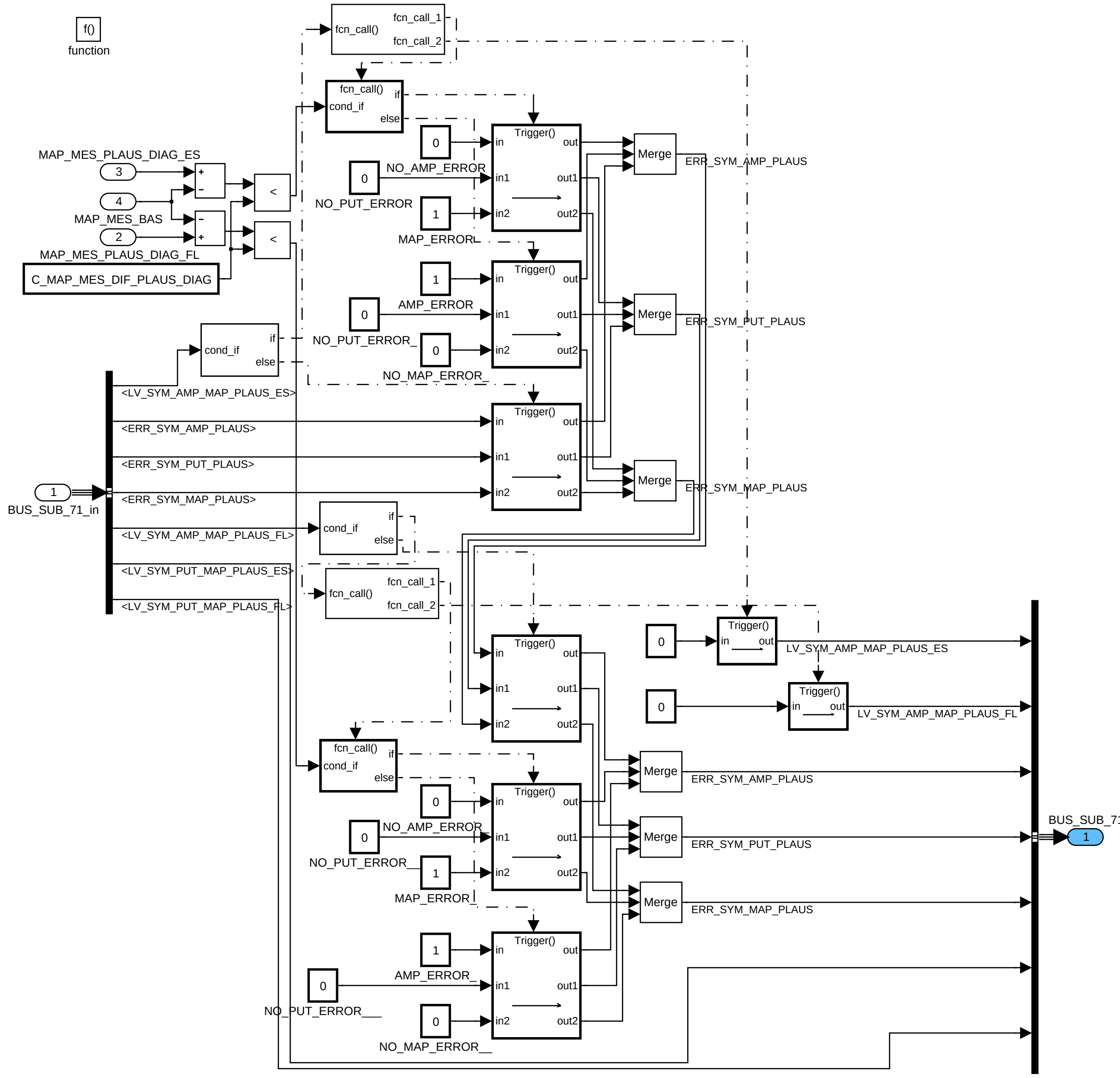


Figure 29 INSY_PLADGAMP0/ operate_100MS/ SUB_7_PRS_AIR_PLAUS_IS/ SUB_71_AMP_MAP_PLAUS_IS

SUB_72: Detection of defective PUT-sensor or MAP-sensor in engine state idle speed:

If the preliminary error bit LV_SYM_PUT_MAP_PLAUS_ES or LV_SYM_PUT_MAP_PLAUS_FL is set, the value of MAP_MES is checked when engine is in state idle or part load. If there is no change in the value of the measured manifold pressure compared to the pressure measured during engine stop or full load, it can be concluded that the MAP sensor is faulty. If however the value of MAP_MES does change, it is most probably a defective PUT sensor.

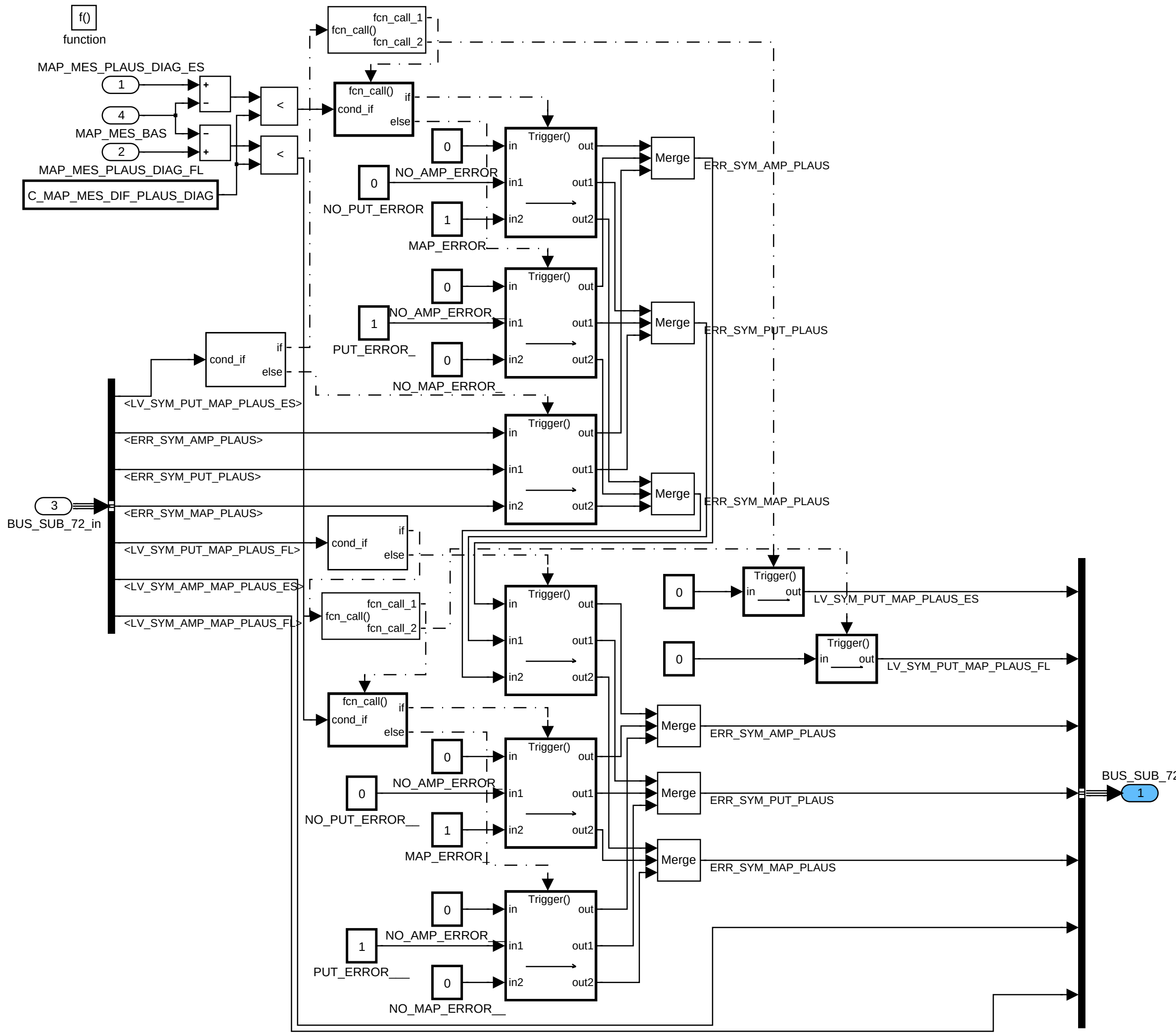


Figure 30 INSY_PLADGAMP0/ operate_100MS/ SUB_7_PRS_AIR_PLAUS_IS/ SUB_72_PUT_MAP_PLAUS_IS

<u>SUB_8: MAP plausibility for MAF-systems equipped with a MAP-sensor (when engine is running):</u>

This test is for MAF-systems with only one pressure sensor, the MAP sensor. The purpose of this test is to check the plausibility between the MAP value measured from the MAP sensor and the MAP value calculated by the air path model (from the engine speed, the TPS value and the MAF sensor value).

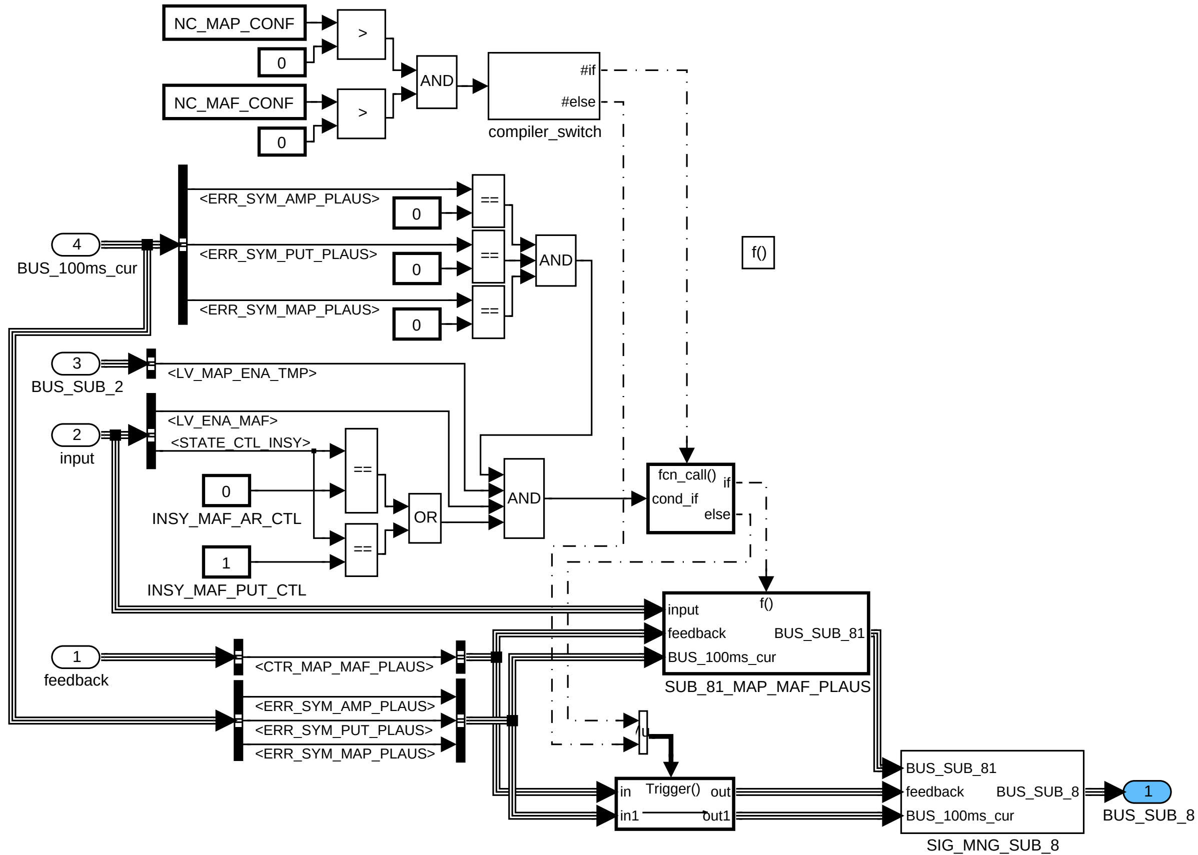


Figure 31 INSY_PLADGAMP0/ operate_100MS/ SUB_8_MAP_MAF_PLAUS

SUB_81: MAP plausibility for MAF-systems equipped with a MAP-sensor (when engine is running):

SUB_811: MAP plausibility for MAF-systems equipped with a MAP-sensor (when engine is running):

This plausibility check is for MAF-systems with only one pressure sensor, the MAP sensor. The purpose of this test is to check the plausibility between the MAP value measured from the MAP sensor and the MAP value calculated by the air path model (from the engine speed, the TPS value and the MAF sensor value).

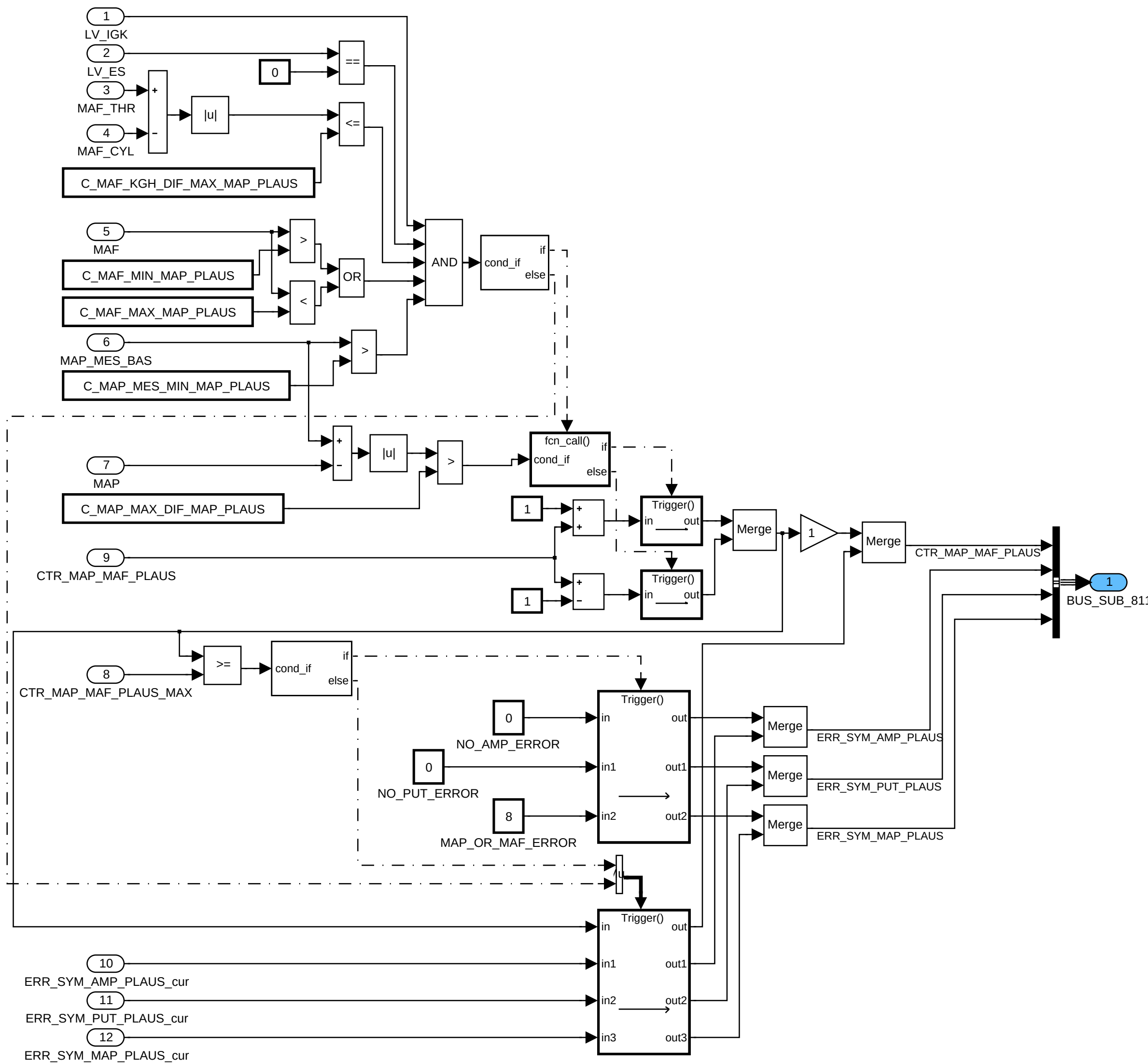


Figure 32 INSY_PLADGAMP0/ operate_100MS/ SUB_8_MAP_MAF_PLAUS/ SUB_81_MAP_MAF_PLAUS/ SUB_811_ENG_RUN

SUB_9: Setting of error management variables:

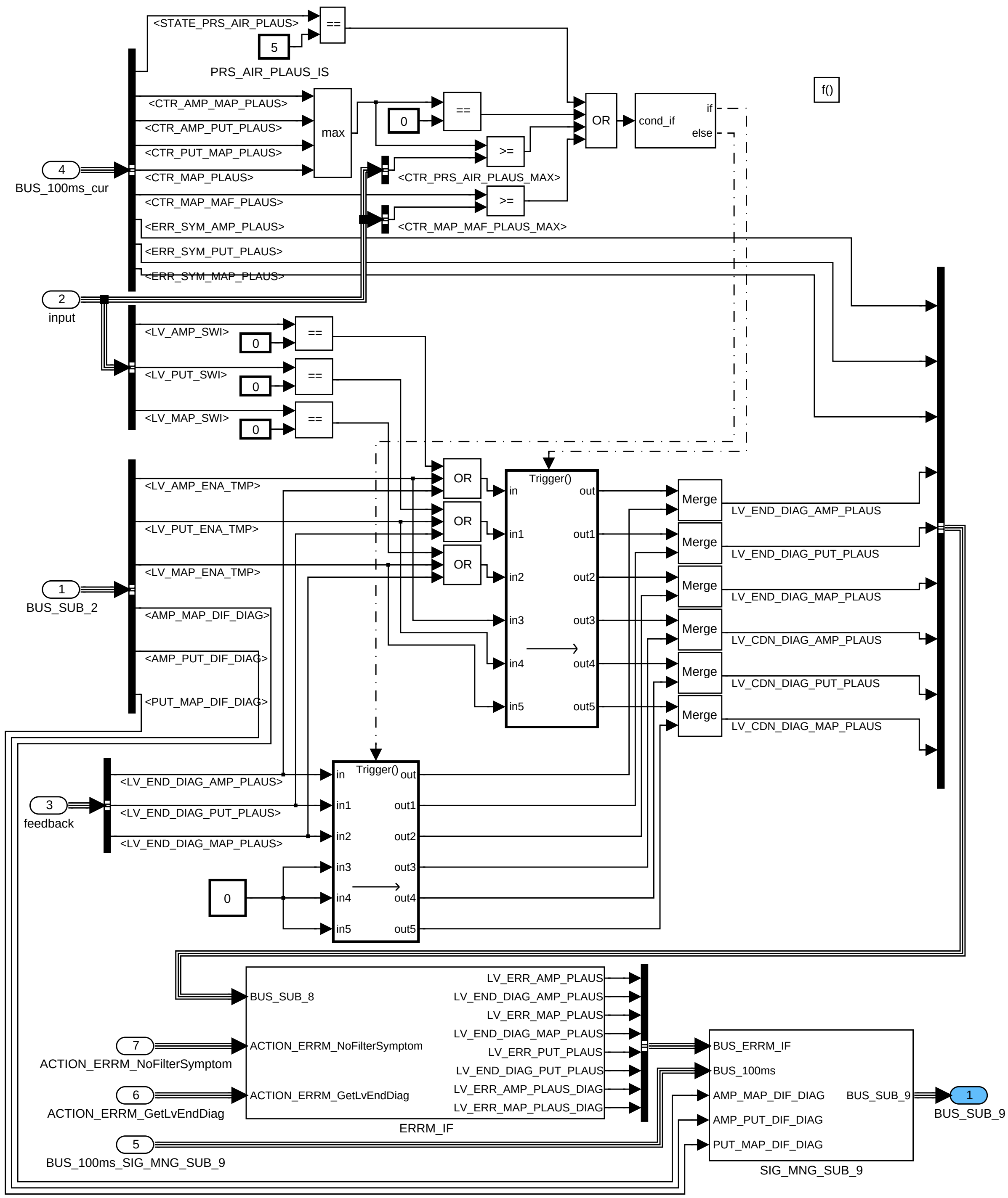


Figure 33 INSY_PLADGAMP0/ operate_100MS/ SUB_9_SET_ERR

B.35 Application incidences for Ambient and Manifold Pressure Plausibility Diagnosis

Output Data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CRK_PRS_AIR_PLAUS	O/V	0... 1H	0... 1	1	[-]
Indicator that engine has cranked since last ECU reset					
LV_INH_DIAG_PRS_AIR_PLAUS	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis					
LV_INH_DIAG_PRS_AIR_PLAUS_ES	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis at engine stop					
LV_INH_DIAG_PRS_AIR_PLAUS_FL	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis in full load					
LV_INH_DIAG_PRS_AIR_PLAUS_IS	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis in idle speed					
LV_INH_DIAG_PRS_AIR_PLAUS_PU	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis in pull					
LV_INH_DIAG_PRS_AIR_PLAUS_PWL	O/V	0... 1H	0... 1	1	[-]
Flag to stop the air pressure sensor plausibility diagnosis in power latch					

Input Data:

LV_ERR_TPS	LV_ERR_AMP_PLAUS	LV_ERR_MAP_PLAUS	LV_ERR_PUT_PLAUS
LV_ERR_WG			

Calibration Data:

Name	Dim	Hex.Limits	Phys.Limits	Resol.	Unit
LC_INH_DIAG_PRS_AIR_PLAUS	1	0... 1H	0... 1	1	[-]
Logical constant to stop the air pressure sensor plausibility diagnosis					

General Information

Depending on project specific requirements, the air pressure plausibility diagnosis can be inhibited for all states by LV_INH_DIAG_PRS_AIR_PLAUS or for individual states by the respective bits.

Error configuration data:

Diagnosis	Symptom	Nr	ABC type
AMP_PLAUS	Measured AMP not plausible	0	NO
AMP_PLAUS	Measured AMP or PUT not plausible	1	NO
AMP_PLAUS	Measured AMP or MAP not plausible	2	NO

Diagnosis	Symptom	Nr	ABC type
MAP_PLAUS	Measured MAP not plausible	0	NO
MAP_PLAUS	Measured MAP or AMP not plausible	1	NO
MAP_PLAUS	Measured MAP or PUT not plausible	2	NO
MAP_PLAUS	Measured MAP or MAF not plausible	3	NO

Diagnosis	Symptom	Nr	ABC type
PUT_PLAUS	Measured PUT not plausible	0	NO
PUT_PLAUS	Measured PUT or AMP not plausible	1	NO
PUT_PLAUS	Measured PUT or MAP not plausible	2	NO

Application Conditions

Initialization: RST, FIRST_TOOTH
Recurrence: 100MS
Activation: always
Deactivation: never

Function description

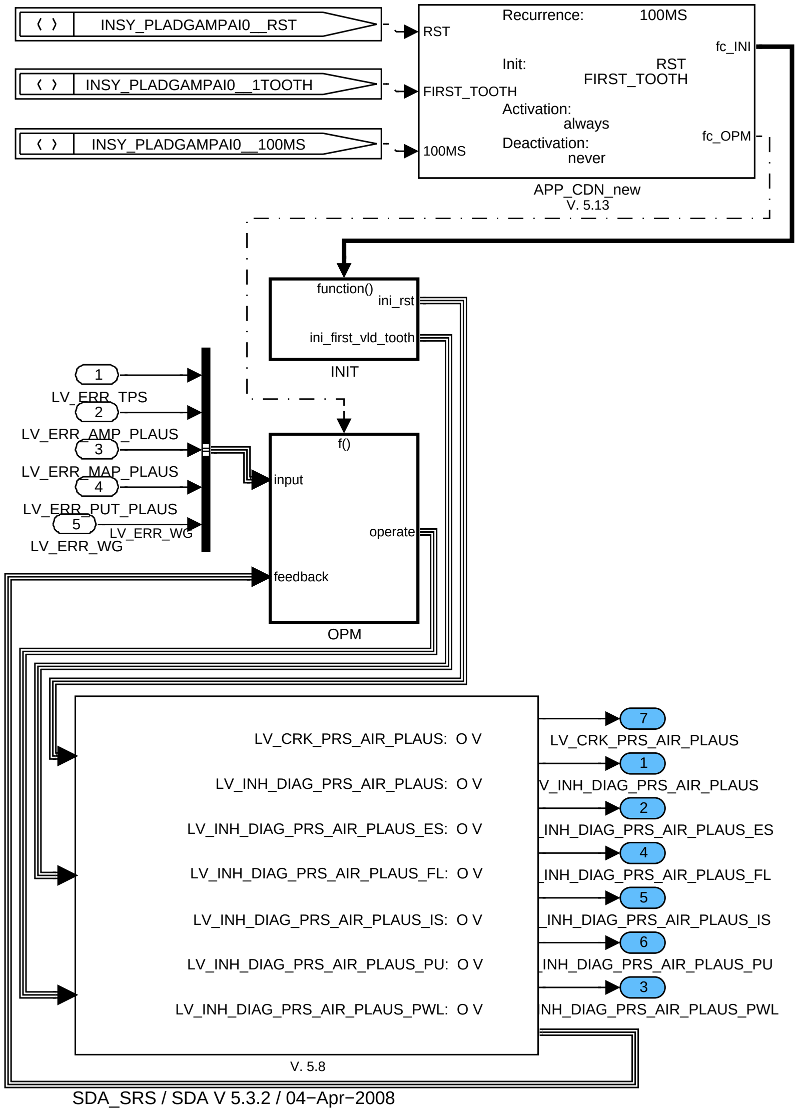


Figure 34:
Path: INSY_PLADGAMPAI0

B.35.1 Initialization:

B.35.1.1 Initialization at ECU reset:

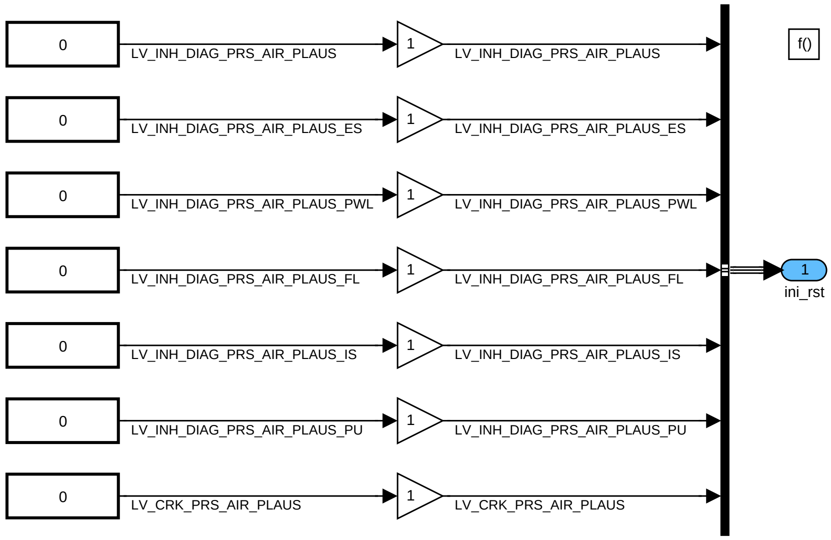


Figure 35:
Path: INSY_PLADGAMPAI0/INIT/RST

B.35.1.2 Initialization at first valid tooth event:

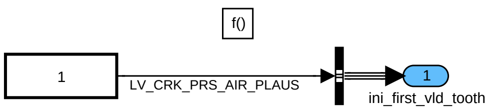


Figure 36:
Path: INSY_PLADGAMPAI0/INIT/FIRST_VLD_TOOTH

B.35.2 operate_100ms:

B.35.2.1 Calculation of LV_INH_DIAG_PRS_AIR_PLAUS:

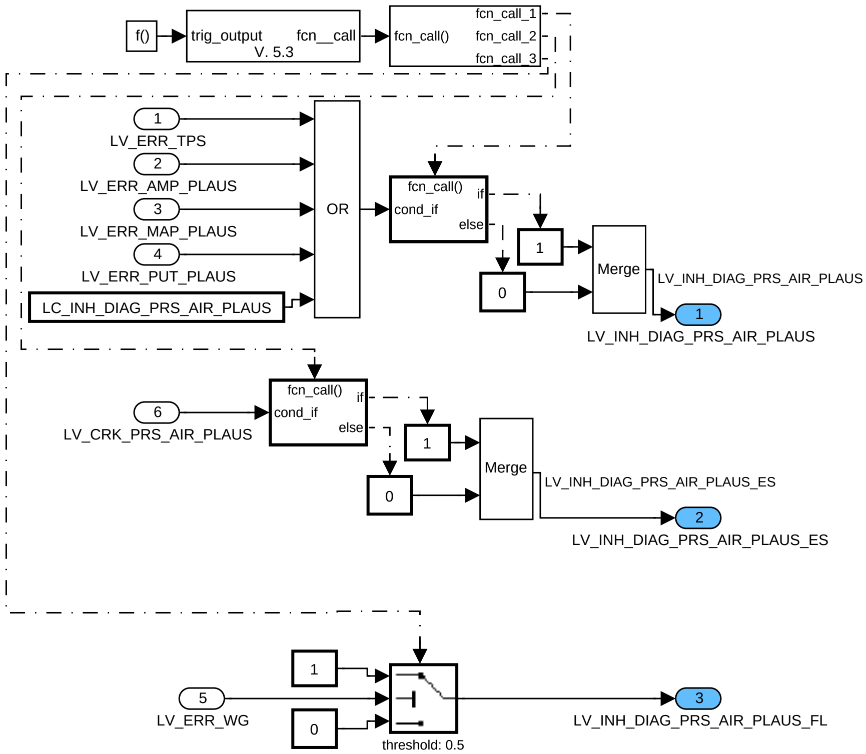


Figure 37:
Path: INSY_PLADGAMPAI0/OPM/CLC_LV_INH_DIAG_PRS_AIR_PLAUS

B.36 Charge Air Pressure and PWM_WG Plausibility Diagnosis (CAP)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CAP_H	O/V	0...1H	0...1	1	-
Boolean that indicates charge air pressure is too high (CAP_H)					
LV_END_DIAG_CAP_H	O/V	0...1H	0...1	1	-
End of diagnosis flag for charge air pressure is too high (CAP_H)					
LV_ERR_CAP_L	O/V	0...1H	0...1	1	-
Boolean that indicates charge air pressure is too low (CAP_L)					
LV_END_DIAG_CAP_L	O/V	0...1H	0...1	1	-
End of diagnosis flag for charge air pressure too low (CAP_L)					
LV_ERR_CAP_L_BAS	O/V	0...1H	0...1	1	-
Boolean that indicates basic charge air pressure too low (CAP_L_BAS)					
LV_END_DIAG_CAP_L_BAS	O/V	0...1H	0...1	1	-
End of diagnosis flag for basic charge air pressure too low (CAP_L_BAS) diagnosis					
ERR_SYM_CAP_H	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom inconsistencies charge air pressure is too high (CAP_H)					
LV_CDN_DIAG_CAP_H	V	0...1H	0...1	1	-
Status of diagnosis flag for inconsistencies charge air pressure is too high (CAP_H)					
PUT_MAX_CAP_H_DIAG	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Maximum pressure ratio at charger for plausibility diagnosis					
PUT_MIN_CAP_L_DIAG_LIM	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Minimum pressure ratio at charger for plausibility diagnosis with change limitation					
PUT_MAX_CAP_H_DIAG_LIM	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Maximum pressure ratio at charger for plausibility diagnosis with change limitation					
PUT_MIN_CAP_L_DIAG	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Minimum pressure ratio at charger for plausibility diagnosis					
T_RCL_CLOSE_DLY_CAP_DIAG	V	0...FFH	0...25.5	0.1	s
Delay time for activation of closing of recirculation for charge air pressure diagnosis					
PUT_AMP_DIF_CAP_L_BAS_DIAG	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.0829175 2	hPa
Minimun pressure difference for basic charge air pressure too low (CAP_L_BAS) diagnosis					
T_DLY_CAP_L_BAS_DIAG	V	0...FFH	0...25.5	0.1	s
Delay time for activation of basic charge air pressure too low (CAP_L_BAS) diagnosis					
T_DLY_CAP_L_DIAG	V	0...FFH	0...25.5	0.1	s
Delay time for activation of charge air pressure LOW diagnosis after exceeding PQ_CHA_SP_MIN condition					
ERR_SYM_CAP_L	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom charge air pressure too low (CAP_L)					
LV_CDN_DIAG_CAP_L	V	0...1H	0...1	1	-
Status of diagnosis flag for charge air pressure is too low (CAP_L)					
ERR_SYM_CAP_L_BAS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Detected symptom basic charge air pressure too low (CAP_L_BAS) diagnosis					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_CAP_L_BAS	V	0...1H	0...1	1	-
Status of diagnosis flag for basic charge air pressure too low (CAP_L_BAS)					

Input data:

LC_TCHA_CONF	C_ABC_INC_CAP_H	C_ABC_INC_CAP_L	C_ABC_INC_CAP_L_BAS
C_ABC_MAX_CAP_H	C_ABC_MAX_CAP_L	C_ABC_MAX_CAP_L_BAS	NC_IDX_DIAG_CAP_H
NC_IDX_DIAG_CAP_L	NC_IDX_DIAG_CAP_L_BAS	LV_INH_DIAG_CAP	N_32
AMP	PUT_AMP_DIF	LV_IN_PROT	PQ_CHA_SP
LV_ST_END	MAF_KGH_TCHA	LV_CMD_RCL_OPEN	PWM_WG
PUT_MES	PUT_SP	PUT_DIF	LV_PUT_CTL_ACT
TCO	LV_FCUT_IND	FTL	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_MIN_CAP_L_DIAG	1	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Minimum Ambient pressure for CAP Low Diagnosis					
C_FTL_MIN_CAP_L_DIAG	1	0...FFH	0...99.609375	0.390625	%
Minimum fuel level for CAP_L and CAP_L_BAS diagnosis					
C_MAF_KGH_MIN_CAP_L_BAS_DIAG	1	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Minimum mass air flow for basic charge air pressure too low (CAP_L_BAS) diagnosis					
C_PQ_CHA_SP_MIN_CAP_L_BAS_DIAG	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Minimun pressure quotient at charger setpoint to activate basic charge air pressure too low (CAP_L_BAS) diag					
C_PQ_CHA_SP_MIN_CAP_L_DIAG	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Minimun pressure quotient at charger setpoint to activate Charge air pressure diagnosis					
C_PUT_AMP_DIF_MIN_CAP_H_DIAG	1	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.0829175 2	hPa
Minimum pressure difference to open the waste gate					
C_PUT_DIF_MAX_CAP_L_DIAG	1	8000...7FFFH	-5.434E+3 ... 5.43383E+3	0.1658325 2	hPa
PUT deviation for for basic charge air pressure too low (CAP_L) diagnosis (init value +200 hPa)					
C_PUT_DIF_MIN_CAP_H_DIAG	1	8000...7FFFH	-5.434E+3 ... 5.43383E+3	0.1658325 2	hPa
PUT deviation for for basic charge air pressure too high (CAP_H) diagnosis (init value -200 hPa)					
C_PUT_MAX_CAP_H_DIAG_LGRD_NEG	1	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Negative Change Limitation for PUT_MAX_CAP_H_DIAG					
C_PWM_WG_MAX_CAP_L_BAS_DIAG	1	0...FFH	0...99.609375	0.390625	%
Maximum PWM_WG for basic charge air pressure too low (CAP_L_BAS) diagnosis					
C_TCO_MIN_CAP_L_DIAG	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO threshold for CAP Low diagnosis					
C_T_DLY_CAP_L_BAS_DIAG	1	0...FFH	0...25.5	0.1	s
Delay time for activation of charge air pressure LOW diagnosis after exceeding PQ_CHA_SP_MIN condition					
C_T_RCL_CLOSE_DLY_CAP_DIAG	1	0...FFH	0...25.5	0.1	s
Delay time for activation of closing of recirculation for charge air pressure diagnosis					
IP_N_32_MIN_CAP_L_BAS_DIAG	2	0...FFH	0...8.16E+3	32	rpm
LDPM_AMP_IP_N_32_MIN_CAP_DIAG	2	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Minimum engine speed for basic charge air pressure too low (CAP_L_BAS) diagnosis					
IP_N_32_MIN_CAP_L_DIAG	2	0...FFH	0...8.16E+3	32	rpm
LDPM_AMP_IP_N_32_MIN_CAP_DIAG	2	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Minimum engine speed for CAP Low Diagnosis					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PUT_AMP_DIF_CAP_L_BAS_DIAG	4	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.0829175 2	hPa
LDP_MAF_KGH_TCHA_IP_PUT_AMP_DIF	4	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Minimum pressure difference for basic charge air pressure too low (CAP_L_BAS) diagnosis					
IP_PUT_MIN_CAP_L_DIAG_LGRD_POS	4	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDPM_N_32_CAP_DIAG	4	0...FFH	0...8.16E+3	32	rpm
Positive Change Limitation for PUT_MIN_CAP_L_DIAG					
IP_T_DLY_CAP_L_DIAG	4	0...FFH	0...25.5	0.1	s
LDPM_N_32_CAP_DIAG	4	0...FFH	0...8.16E+3	32	rpm
Delay time for activation of charge air pressure LOW diagnosis after exceeding PQ_CHA_SP_MIN condition					
IP_PUT_MAX_CAP_H_DIAG	6x6	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDPM_PUT_SP_IP_PUT_CAP_DIAG	6	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDPM_N_32_IP_PUT_CAP_DIAG	6	0...FFH	0...8.16E+3	32	rpm
Maximum charge air pressure quotient for charge air pressure too high (CAP_H) diagnosis					
IP_PUT_MIN_CAP_L_DIAG	6x6	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDPM_PUT_SP_IP_PUT_CAP_DIAG	6	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
LDPM_N_32_IP_PUT_CAP_DIAG	6	0...FFH	0...8.16E+3	32	rpm
Minimum charge air pressure quotient for charge air pressure too low (CAP_L) diagnosis					

Import actions:

ACTION_ERRM_FilterSymptom(IN <XX>, IN <lv_cdn_diag_XX>, IN <err_sym_XX>, IN <c_abc_inc_XX>, IN <c_abc_dec_XX>, IN <c_abc_max_XX>, OUT <lv_err_XX>)
This action computes the elementary anti-bounce filter for one failure treatment and returns filter result
ACTION_ERRM_GetLvEndDiag(IN <XX>, OUT <lv_end_diag_XX>)
This action returns the end-of-diagnosis flag

B.36.1 General information

The purpose of this diagnosis is to detect several failures related with the charge air pressure and the wastegate behaviour.

In case the charge air pressure is **too high** any of the following reasons could cause this bad behaviour:

wastegate stuck closed

defect engine

closed exhaust system

defect EPC pressure supply or stuck closed

If this failure is present, the turbo charge protection will react to avoid to damage the engine and to ensure a good driveability of the car.(see Turbo Protection Specification)

In case the charge air pressure is **too low** any of the following reasons could cause this bad behaviour:

defect charger

leaky air tube or intercooler

recirculation valve stuck open

defect EPC stuck open

Error-symptoms for the Charge air pressure (CAP) diagnoses are defined as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> SYM_0
 │ │ └───> SYM_1
 │ └─────> SYM_2
 └───────> SYM_3: CAP too high (CAP_H)/ CAP too low (CAP_L)/ Basic CAP too low (CAP_L_BAS)
```

The error-management variables are initialized according to filter-type.

Function Description

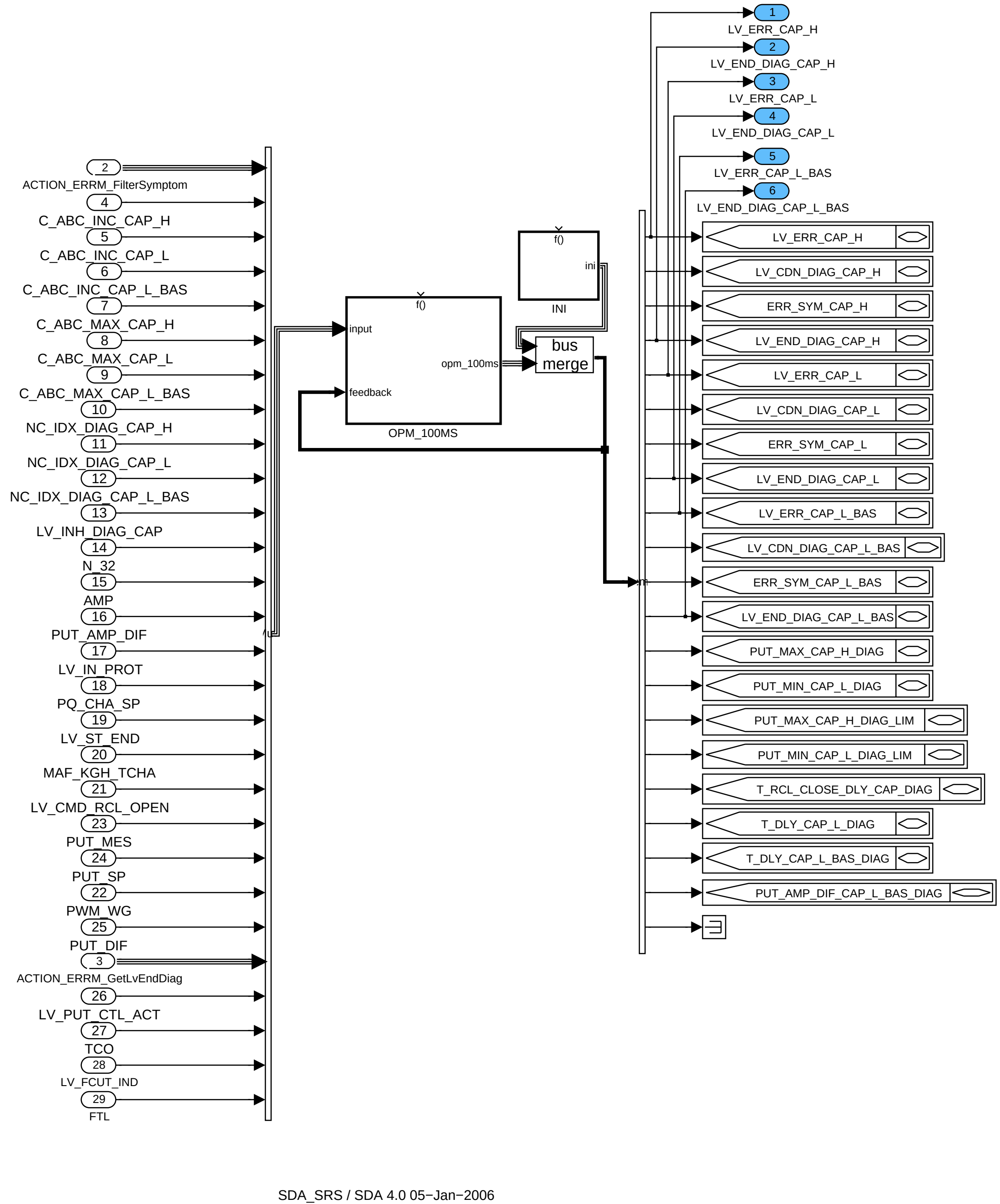


Figure 38 CHRG_PLADGPWMWG0

B.36.1.1 Initialization

At reset and LV_ST_END 0 -> 1

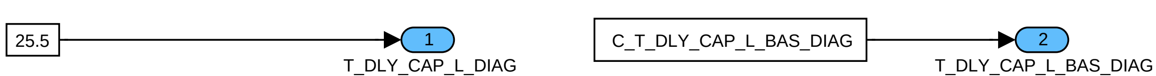


Figure 39 CHRG_PLADGPWMWG0/ INI/ INI

B.36.1.2 Formula section

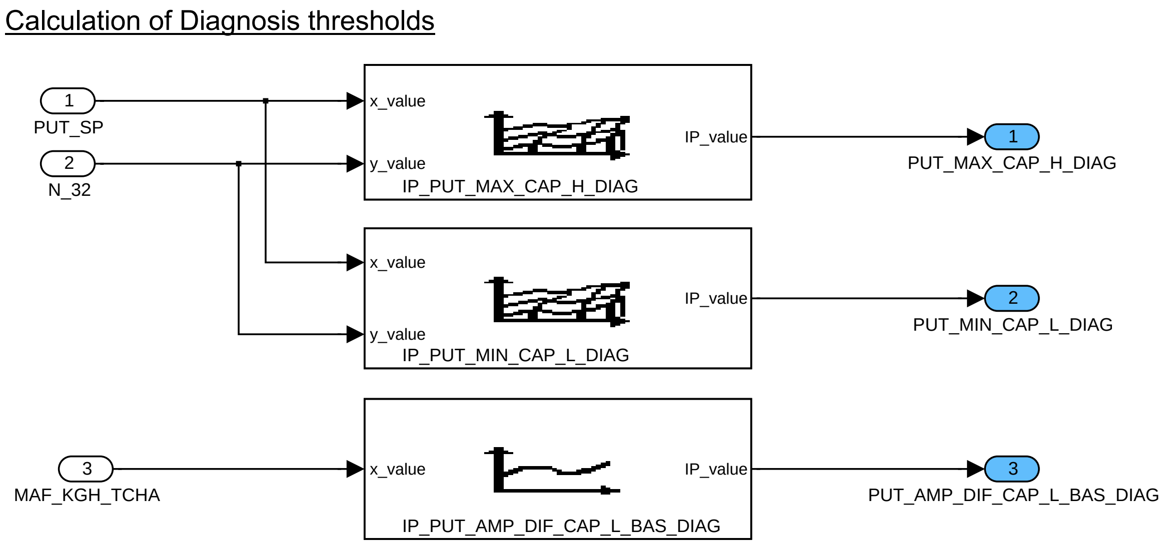


Figure 40 CHRG_PLADGPWMWG0/ OPM_100MS/ DIAGNOSIS_THRESHOLDS

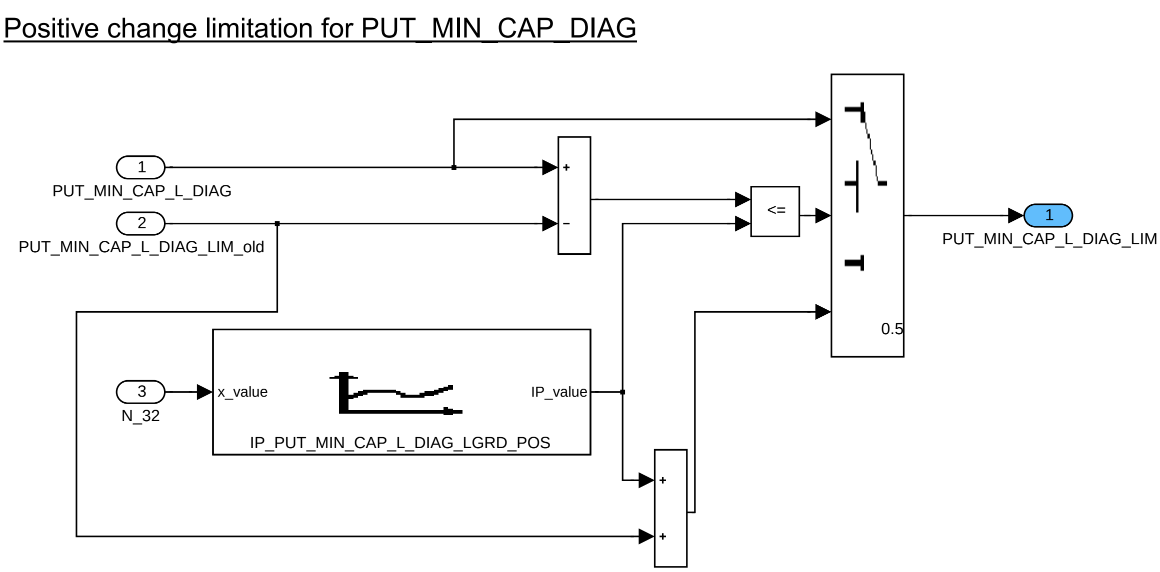


Figure 41 CHRG_PLADGPWMWG0/ OPM_100MS/ POS_CHANGE_LIMITATION

Negative change limitation for PUT_MAX_CAP_DIAG

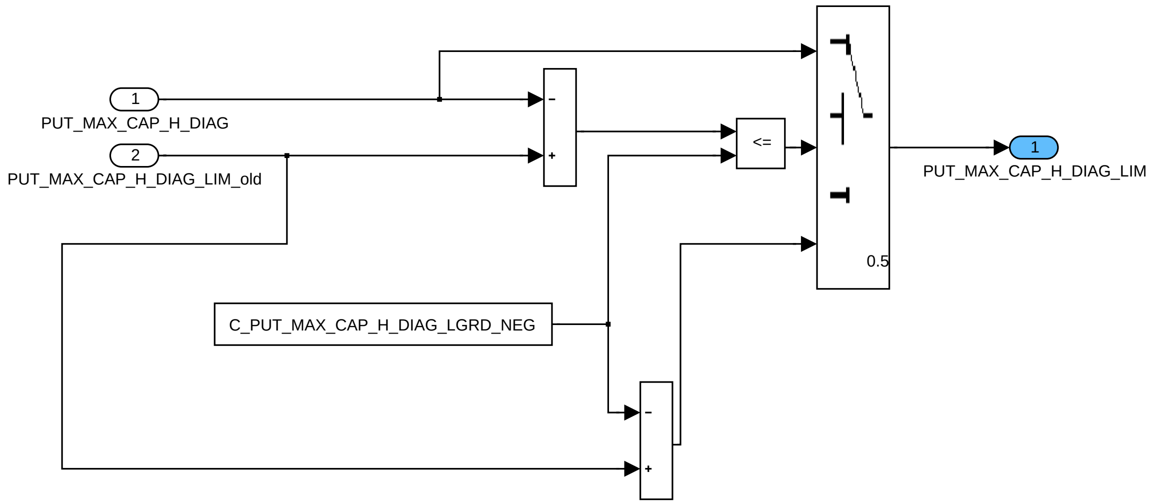


Figure 42 CHRG_PLADGPWMWG0/ OPM_100MS/ NEG_CHANGE_LIMITATION

Time delay after closing of the recirculation valve

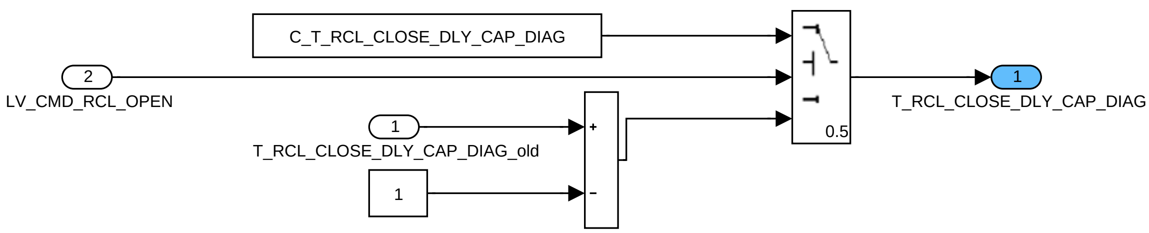


Figure 43 CHRG_PLADGPWMWG0/ OPM_100MS/ T_DLY_AFTER_CLOSING_RCL

Time delay after increasing PQ_CHA_SP jump

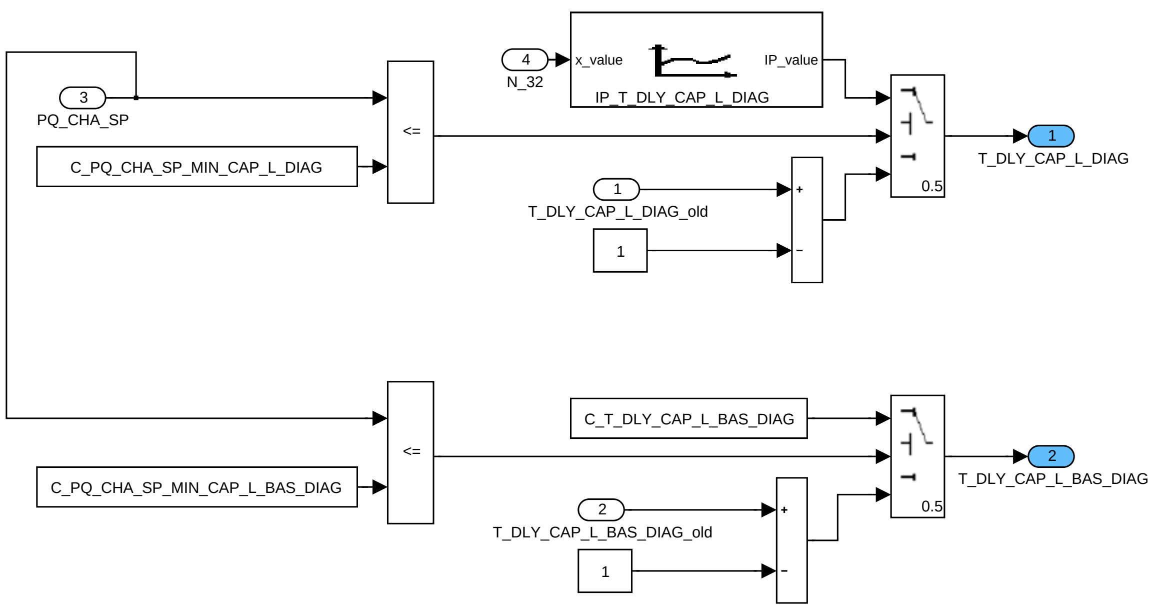


Figure 44 CHRG_PLADGPWMWG0/ OPM_100MS/ T_DLY_AFTER_INCR_PQ_CHA_SP_JUMP

Diagnosis for "Charge Air Pressure too high" (CAP_H)

In case of LV_IN_PROT = 1 the calibration C_ABC_MAX_CAP_H is taken instead of C_ABC_INC_CAP_H as anti bounce counter. This ensures an immediate error entry without debouncing.

HINT: If the error flag is set, a torque reduction must be executed.

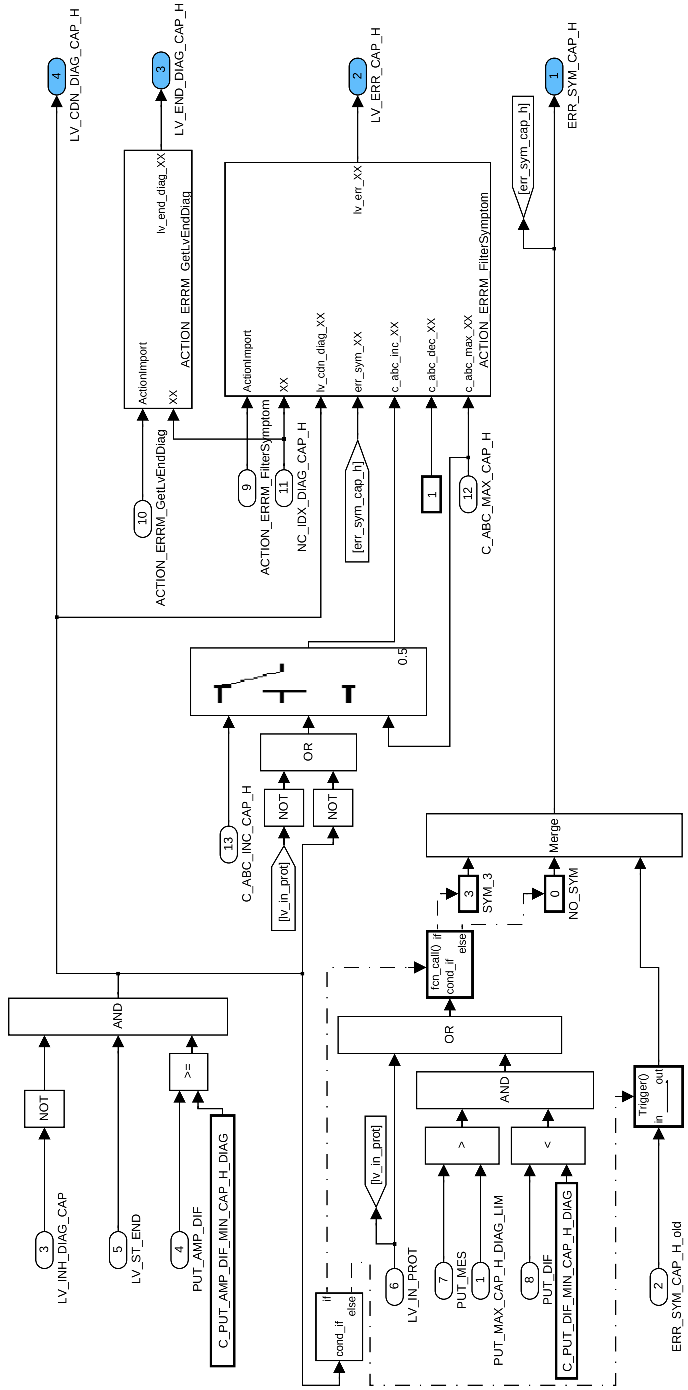


Figure 45 CHRG_PLADGPWMWG0/ OPM_100MS/ DIAGNOSIS__PRESSURE_TOO_HIGH

Diagnosis for "Charge Air Pressure too low" (CAP_L)

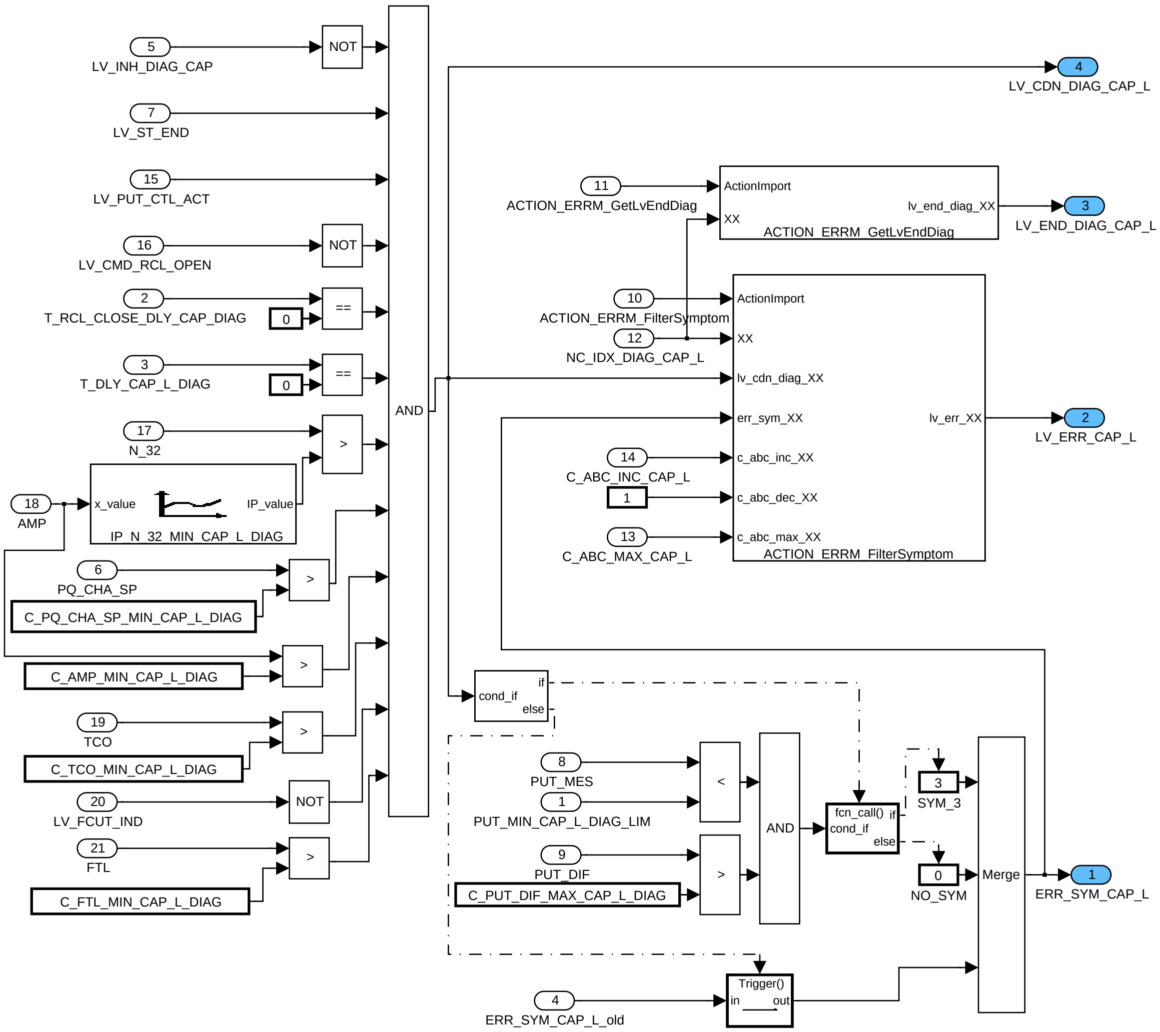


Figure 46 CHRG_PLADGPWMWG0/ OPM_100MS/ DIAGNOSIS__PRESSURE_TOO_LOW

Diagnosis for "Basic Charge Air Pressure too low" (CAP_L_BAS)

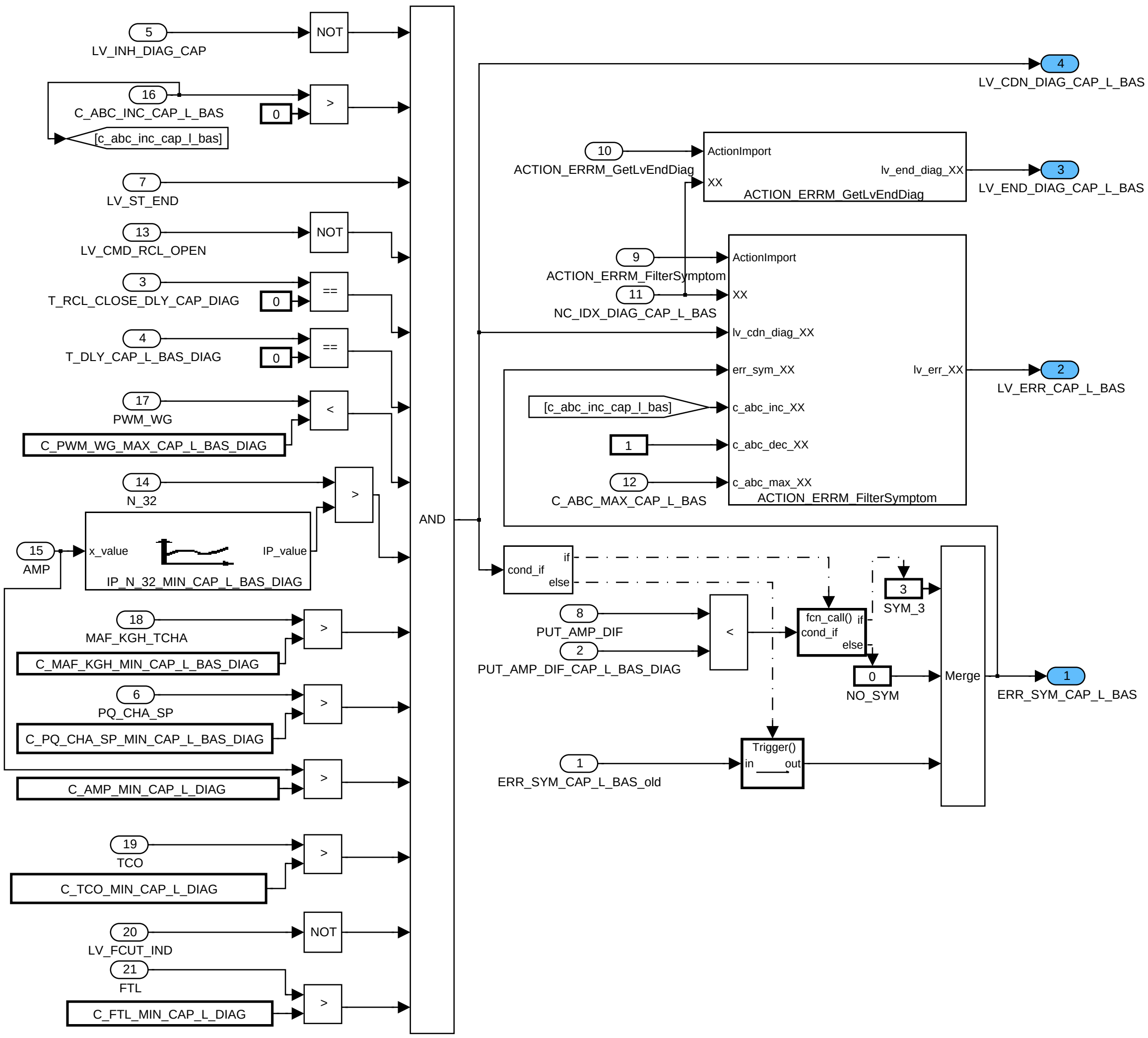


Figure 47 CHRG_PLADGPWMWG0/ OPM_100MS/ DIAGNOSIS__BASIC_PRESSURE_TOO_LOW

Saving old input values

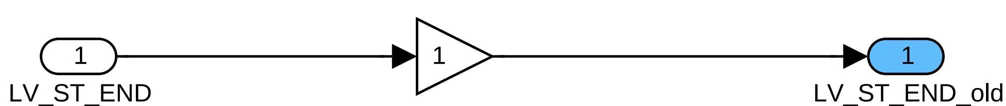


Figure 48 CHRG_PLADGPWMWG0/ OPM_100MS/ OLD_INPUT_VALUES

B.37 Application incidences for Charge Air Pressure and PWM_WG Plausibility Diagnosis (CAP)

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_CAP	O/V	0...1H	0...1	1	[-]
Inhibition condition for the Charge Air Pressure and PWM_WG Plausibility diagnosis					
LV_INH_DIAG_RBM_CAP	-	0...1H	0...1	1	[-]
Inhibition condition for the Charge Air Pressure and PWM_WG Plausibility RBM diagnosis					
STATE_RBM_CAP_L	O/V	0...7H	0...7	1	[-]
Interface of CAP_L monitor with RBM statistics Bit 0: conditions are met long enough to detect malfunction, Bit 1: inhibition of the monitor because of system failure(s), Bit 2: individual RBM conditions of the monitor were encountered within this DC					
RATE_RBM_CAP_L	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Actual RBM rate of CAP_L , for application purposes					
STATE_RBM_CAP_L_BAS	O/V	0...7H	0...7	1	[-]
Interface of CAP_L_BAS monitor with RBM statistics Bit 0: conditions are met long enough to detect malfunction Bit 1: inhibition of the monitor because of system failure(s) Bit 2: individual RBM conditions of the monitor were encountered within this DC					
RATE_RBM_CAP_L_BAS	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Actual RBM rate of CAP_L_BAS , for application purposes					
STATE_RBM_CAP_H	O/V	0...7H	0...7	1	[-]
Interface of CAP_H monitor with RBM statistics Bit 0: conditions are met long enough to detect malfunction Bit 1: inhibition of the monitor because of system failure(s) Bit 2: individual RBM conditions of the monitor were encountered within this DC					
RATE_RBM_CAP_H	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Actual RBM rate of CAP_H , for application purposes					

Input data:

CTR_CDN_RBM_CAP_H	CTR_CDN_RBM_CAP_L	CTR_CDN_RBM_CAP_L_BAS	CTR_COMP_RBM_CAP_H
CTR_COMP_RBM_CAP_L	CTR_COMP_RBM_CAP_L_BAS	CTR_ERR_DYN_NR	LC_TCHA_CONF
LV_DC	LV_ENA_PUT	LV_END_DIAG_CAP_H	LV_END_DIAG_CAP_L
LV_END_DIAG_CAP_L_BAS	LV_ERR_AMP	LV_ERR_AMP_PLAUS	LV_ERR_EL_PUT
LV_ERR_PUT	LV_ERR_PUT_PLAUS	LV_ERR_VCC_SENS_SUB	

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

This flag inhibits the diagnosis in case that the pressure up throttle sensor has any problem (electrical or plausibility).

Application conditions:

Initialisation: LV_INH_DIAG_CAP = 1 at reset or at transition LV_DC: 0 →1

Recurrence: 100 ms

Activation: LC_TCHA_CONF =1

Deactivation: LC_TCHA_CONF =0

Formula section:

```
If      LV_ERR_PUT = 1
   or   LV_ERR_AMP = 1
   or   LV_ERR_AMP_PLAUS = 1

then    LV_INH_DIAG_RBM_CAP = 1
else    LV_INH_DIAG_RBM_CAP = 0
endif

If      LV_ENA_PUT = 0
   or   LV_INH_DIAG_RBM_CAP = 1
Then    LV_INH_DIAG_CAP  = 1
Else    LV_INH_DIAG_CAP  = 0
Endif
```

Configuration data :

Diagnostic CAP_H	Symptoms	Nb	Filter type
CAP_H Plausibility		*0*	*MEM*
		1	
		2	
	Plausibility error charge air pressure is too high	3	

Diagnostic CAP_L	Symptoms	Nb	Filter type
CAP_L Plausibility		*0*	*MEM*
		1	
		2	
	Plausibility error charge air pressure is too low	3	

Diagnostic CAP_L	Symptoms	Nb	Filter type
CAP_L _BAS Plausibility		*0*	*MEM*
		1	
		2	
	Plausibility error basic charge air pressure is too low	3	

B.37.1 Interface for RBM – Charge air pressure LOW diagnosis (CAP_L)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the CAP_L monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_CAP_L data.

Within STATE_RBM_CAP_L, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (not valid for this diagnosis)

Application conditions:

Initialisation:

at ECU reset:

bit 0, bit 1 and bit 2 of STATE_RBM_CAP_L = 0 at LV_DC 0 → 1 transition:

bit 0 and bit 1of STATE_RBM_CAP_L = 0
bit 2 of STATE_RBM_CAP_L = 1

on failure memory reset:

bit 1 of STATE_RBM_CAP_L = 0

Recurrence: 1 s

Activation: LV_DC = 1 **and** LC_TCHA_CONF = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_PUT_PLAUS	LV_ERR_EL_PUT	LV_ERR_VCC_SENS_SUB	

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }
Then(1) **While** bit 1 of STATE_RBM_CAP_L = 0 **do**
with each XX failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
If(2) XX has a pending status
Then(2) bit 1 of STATE_RBM_CAP_L = 1
Endif(2)
Endwhile
Else(1) { the dynamic failure memory is empty }
No action
Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_CAP_L = 0
Then **If** LV_END_DIAG_CAP_L = 1
Then bit 0 of STATE_RBM_CAP_L = 1
Endif
Endif
If bit 1 of STATE_RBM_CAP_L = 0
Then **If** LV_INH_DIAG_RBM_CAP = 1
Then bit 1 of STATE_RBM_CAP_L = 1
Endif
Endif

- Calculation of actual RBM rate (for application purposes)

recurrence: 1 s

If CTR_CDN_RBM_CAP_L = 0
then RATE_RBM_CAP_L = FFFFH
else RATE_RBM_CAP_L
= (CTR_COMP_RBM_CAP_L / CTR_CDN_RBM_CAP_L)
endif

B.37.2 Interface for RBM – Charge air pressure LOW diagnosis (CAP_L_BAS)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the CAP_L_BAS monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_CAP_L_BAS data.

Within STATE_RBM_CAP_L_BAS, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0) (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1) (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2) (not valid for this diagnosis)

Application conditions:

Initialisation:

at ECU reset:
bit 0, bit 1 and bit 2 of STATE_RBM_CAP_L_BAS = 0

at LV_DC 0 → 1 transition:
bit 0 and bit 1of STATE_RBM_CAP_L_BAS = 0
bit 2 of STATE_RBM_CAP_L_BAS = 1

on failure memory reset:
bit 1 of STATE_RBM_CAP_L_BAS = 0

Recurrence: 1 s

Activation: LV_DC = 1 **and** LC_TCHA_CONF = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_PUT_PLAUS	LV_ERR_EL_PUT	LV_ERR_VCC_SENS_SUB	

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1) **While** bit 1 of STATE_RBM_CAP_L_BAS = 0 **do**
with each XX failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
If(2) XX has a pending status

Then(2) bit 1 of STATE_RBM_CAP_L_BAS = 1

Endif(2)

Endwhile

Else(1) { the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_CAP_L_BAS = 0

Then **If** LV_END_DIAG_CAP_L_BAS = 1

Then bit 0 of STATE_RBM_CAP_L_BAS = 1

Endif

Endif

If bit 1 of STATE_RBM_CAP_L_BAS = 0

Then **If** LV_INH_DIAG_RBM_CAP = 1

Then bit 1 of STATE_RBM_CAP_L_BAS = 1

Endif

Endif

- Calculation of actual RBM rate (for application purposes)

recurrence: 1 s

If CTR_CDN_RBM_CAP_L_BAS = 0
then RATE_RBM_CAP_L_BAS = FFFFH
else RATE_RBM_CAP_L_BAS
= (CTR_COMP_RBM_CAP_L_BAS / CTR_CDN_RBM_CAP_L_BAS)
endif

B.37.3 Interface for RBM – Charge air pressure HIGH diagnosis (CAP_H)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the CAP_H monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_CAP_H data.

Within STATE_RBM_CAP_H, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
(no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
(depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
(not valid for this diagnosis)

Application conditions:

Initialisation:

at ECU reset:
bit 0, bit 1 and bit 2 of STATE_RBM_CAP_H = 0

at LV_DC 0 → 1 transition:
bit 0 and bit 1of STATE_RBM_CAP_H = 0
bit 2 of STATE_RBM_CAP_H = 1

on failure memory reset:
bit 1 of STATE_RBM_CAP_H = 0

Recurrence: 1 s

Activation: LV_DC = 1 **and** LC_TCHA_CONF = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_PUT_PLAUS	LV_ERR_EL_PUT	LV_ERR_VCC_SENS_SUB	

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1) **While** bit 1 of STATE_RBM_CAP_H = 0 **do**
with each XX failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
If(2) XX has a pending status
Then(2) bit 1 of STATE_RBM_CAP_H = 1
Endif(2)
Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_CAP_H = 0

Then **If** LV_END_DIAG_CAP_H = 1

Then bit 0 of STATE_RBM_CAP_H = 1

Endif

Endif

If bit 1 of STATE_RBM_CAP_H = 0

Then **If** LV_INH_DIAG_RBM_CAP = 1

Then bit 1 of STATE_RBM_CAP_H = 1

Endif

Endif

- Calculation of actual RBM rate (for application purposes)

recurrence: 1 s

If CTR_CDN_RBM_CAP_H = 0
then RATE_RBM_CAP_H = FFFFH
else RATE_RBM_CAP_H
= (CTR_COMP_RBM_CAP_H / CTR_CDN_RBM_CAP_H)
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_CAP_H	1	0...FFH	0...255	1	[-]
Increment of the Plausibility error to detect waste gate stuck in a fully closed position anti-bounce counter					
C_ABC_MAX_CAP_H	1	1...FFH	1...255	1	[-]
Threshold to be reached, before permanently activating Plausibility error to detect waste gate stuck in a fully closed position					
C_ABC_INC_CAP_L	1	0...FFH	0...255	1	[-]
Increment of the Plausibility error to detect waste gate stuck in a fully open position anti-bounce counter					
C_ABC_MAX_CAP_L	1	1...FFH	1...255	1	[-]
Threshold to be reached, before permanently activating Plausibility error to detect waste gate stuck in a fully open position					
C_ABC_INC_CAP_L_BAS	1	0...FFH	0...255	1	[-]
Increment of the Plausibility error to detect waste gate stuck in a fully open position anti-bounce counter					
C_ABC_MAX_CAP_L_BAS	1	1...FFH	1...255	1	[-]
Threshold to be reached, before permanently activating Plausibility error to detect waste gate stuck in a fully open position					

B.37.4 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_TCHA_MAX_TOT_DC	V/O/S	0...FFFFH	0...4E+5	6.10360876	rpm
Former / current driving cycle maximum turbo charger rotational speed					

Input data:

N_TCHA	LV_IGK		

Description:

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost:

N_TCHA_MAX_TOT_DC = 0H

- otherwise: restored from non-volantile memory

Recurrence: *100 ms*

Activation: *LV_IGK = 1*

Deactivation: -

Formula section:

If N_TCHA > N_TCHA_MAX_TOT_DC

Then N_TCHA_MAX_TOT_DC = N_TCHA

Endif

B.38 Coolant temperature plausibility diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TCO_PLAUS	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature plausibility error					
ERR_SYM_TCO_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom for coolant temperature plausibility error					
LV_CDN_DIAG_TCO_PLAUS	V	0...1H	0...1	1	[-]
Boolean for coolant temperature plausibility diagnosis conditions					
LV_END_DIAG_TCO_PLAUS	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature plausibility diagnosis					
LV_END_DIAG_RBM_TCO_PLAUS	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature plausibility diagnosis for RBM					
T_MIN_DIAG_TCO_PLAUS	V	0...FFFFH	0...65535	1	[s]
Minimum time to activate the coolant temperature plausibility diagnosis					
T_AST_DIAG_TCO_PLAUS	V	0...FFFFH	0...65535	1	[s]
Time after start of the performed coolant temperature plausibility diagnosis					
T_AST_MAX_DIAG_TCO_PLAUS	V	0...FFFFH	0...65535	1	[s]
Maximum time after start to reach closed-loop enable temperature for coolant temperature plausibility diagnosis					
TCO_DIF_DIAG_TCO_PLAUS	V	80...7FH	-96...95.25	0.75	[°C]
Difference between closed-loop enable- and intake air temperature at start for coolant plausibility diagnosis					

Input data:

LV_PUC	TCO	TCO_ST	TCO_SUB
LV_ST_END	TIA_THR_ST	T_DIAG_AST	RATIO_T_IS_AST
LV_CDN_VB_OBD_2	LV_INH_DIAG_TCO_PLAUS		

FUNCTION DESCRIPTION:

General information:

To monitor the activation of the lambda control after start above a coolant temperature threshold (OBD II requirement), a plausibilization between the coolant temperature increase (TCO) and the calculated coolant temperature increase (TCO_SUB) is made.

The coolant temperature plausibility diagnosis is performed once per engine run.

If the engine stalls and the coolant temperature plausibility diagnosis has not run out, then at the next engine start the timer will be initialized new and the diagnosis starts again. When the diagnosis is finished, the Boolean for end of coolant temperature plausibility diagnosis is set.

Remark: *At the time when the "TCO-plausibility diagnosis" is performed, it is not possible to distinguish between a "TCO-plausibility error" and a "Thermostat error" (which can only be detected later). Therefore a "TCO-plausibility error" is entered, even if the problem may be related to the Thermostat (in order to activate the limp home in case there is really a problem with the coolant temperature sensor).*
That means, if additionally to the "TCO- plausibility error" a "Thermostat diagnosis error" is detected, the "TCO-plausibility error" can be ignored.

Description:

Error-symptoms are defined to this diagnosis function as following:

```
□□□□
 | | | └> -                                        (= SYM_0)
 | | └───> -                                      (= SYM_1)
 | └─────> -                                      (= SYM_2)
 └───────> Coolant temperature plausibility error (= SYM_3)
```

Application conditions:

Initialization at ECU reset:

ERR_SYM_TCO_PLAUS = “NO_SYM”

LV_ERR_TCO_PLAUS = 0

LV_CDN_DIAG_TCO_PLAUS = 0

Initialisation at EXIT_ST:

LV_END_DIAG_RBM_TCO_PLAUS = 0

T_MIN_DIAG_TCO_PLAUS = ID_T_MIN_DIAG_TCO_PLAUS

T_AST_DIAG_TCO_PLAUS = 0

TCO_DIF_DIAG_TCO_PLAUS =

ID_TCO_MIN_DIAG_TCO_PLAUS - TIA_THR_ST

T_AST_MAX_DIAG_TCO_PLAUS = IP_T_MAX_DIAG_TCO_PLAUS

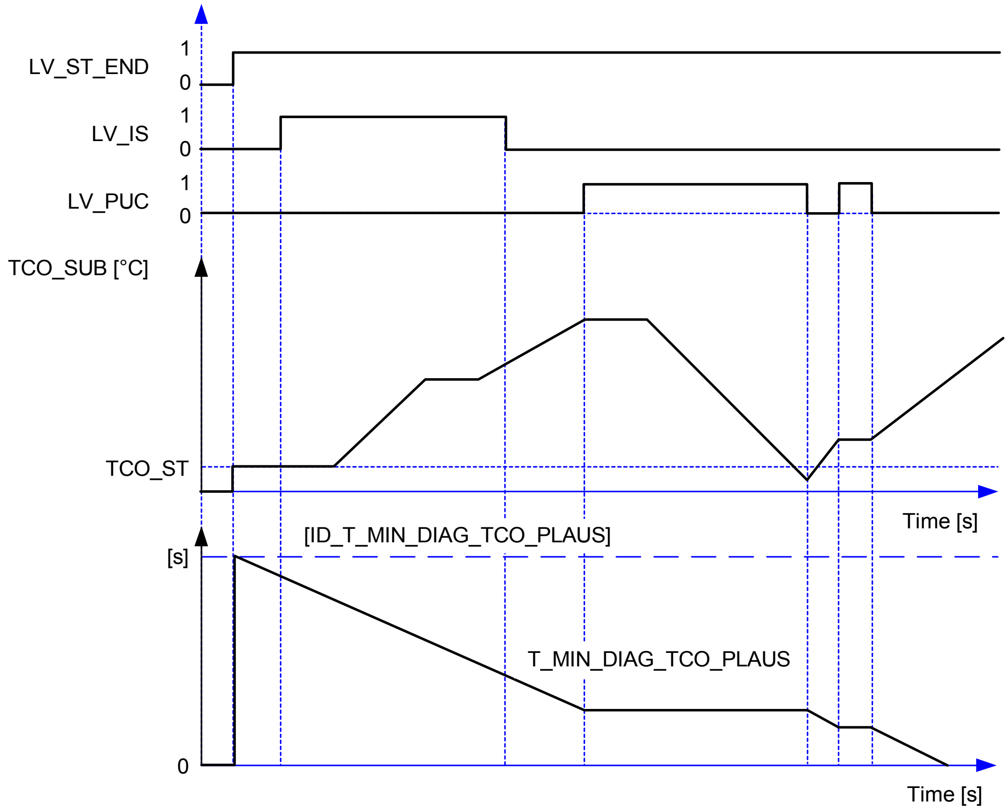
// LV_END_DIAG_TCO_PLAUS is directly initialized by ERRM

Recurrence: 1000 ms

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Signal flow diagram:



Formula section:

Calculation of plausibility diagnosis activation conditions:

If	LV_INH_DIAG_TCO_PLAUS = 0	**and**
	LV_END_DIAG_TCO_PLAUS = 0	**and**
	LV_CDN_VB_OBD_2 = 1	**and**
	TCO_ST < ID_TCO_MIN_DIAG_TCO_PLAUS	
then	LV_CDN_DIAG_TCO_PLAUS = 1	
else	LV_CDN_DIAG_TCO_PLAUS = 0	
endif		

Calculation of minimum diagnosis time after engine start and end of diagnosis for RBM:

If(1)	LV_END_DIAG_RBM_TCO_PLAUS = 0	**and**
	LV_INH_DIAG_TCO_PLAUS = 0	**and**
	LV_CDN_VB_OBD_2 = 1	**and**
	TCO_ST < ID_TCO_MIN_DIAG_TCO_PLAUS	

then(1)

if(2a) LV_PUC = 1

then(2a) $T\_MIN\_DIAG\_TCO\_PLAUS_{(n)}$ = $T\_MIN\_DIAG\_TCO\_PLAUS_{(n-1)}$

$T\_AST\_MAX\_DIAG\_TCO\_PLAUS_{(n)}$ =

$T\_AST\_MAX\_DIAG\_TCO\_PLAUS_{(n-1)}$

else(2a) $T\_MIN\_DIAG\_TCO\_PLAUS_{(n)}$ = $T\_MIN\_DIAG\_TCO\_PLAUS_{(n-1)}$ - 1 s

$T\_AST\_MAX\_DIAG\_TCO\_PLAUS_{(n)}$ =

$T\_AST\_MAX\_DIAG\_TCO\_PLAUS_{(n-1)}$ -1s

// counter minimum value limited to zero, no overflow possible

endif(2a)

if(2b) LC_T_DLY_TCO_PLAUS = 1

then(2b) *// only use of delay time based on absolute temperature at start*

if(2c) T_MIN_DIAG_TCO_PLAUS = 0 **and**

(TCO_SUB > ID_TCO_MIN_DIAG_TCO_PLAUS **or**

LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1 **)**

*// default is LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1; this **must** also be kept for US applications, exept lambda activation by TCO is circumvented*

then(2c) LV_END_DIAG_RBM_TCO_PLAUS = 1

// end of diagnosis for RBM

endif(2c)

else(2b) *// only use of delay time based on relative temperature at start*

if(2d) T_AST_MAX_DIAG_TCO_PLAUS = 0 **and**

(TCO_SUB > ID_TCO_MIN_DIAG_TCO_PLAUS **or**

LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1 **)**

*// default is LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1; this **must** also be kept for US applications, exept lambda activation by TCO is circumvented*

then(2d) LV_END_DIAG_RBM_TCO_PLAUS = 1

// end of diagnosis for RBM

endif(2d)

endif(2b)

endif(1)

Error detection:

If(1) LV_CDN_DIAG_TCO_PLAUS = 1

then(1)

If(2) TCO > ID_TCO_MIN_DIAG_TCO_PLAUS

then(2) ERR_SYM_TCO_PLAUS = "NO_SYM"

LV_ERR_TCO_PLAUS = 0 *// without debounce*

LV_END_DIAG_TCO_PLAUS = 1 *// end of diagnosis*

```
                    T_AST_DIAG_TCO_PLAUS = T_DIAG_AST
    else(2)
    if(3)           LC_T_DLY_TCO_PLAUS = 1
    then(3)         // only use of delay time based on absolute temperature at start

        if(4a)      T_MIN_DIAG_TCO_PLAUS = 0                              and
                    ( TCO_SUB > ID_TCO_MIN_DIAG_TCO_PLAUS  or
                    LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1 )
                    // default is LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1; this must also be kept
                    for US applications, exept lambda activation by TCO is circumvented
        then(4a)    ERR_SYM_TCO_PLAUS = "SYM_3"
                    LV_ERR_TCO_PLAUS = 1                     // without debounce
                    LV_END_DIAG_TCO_PLAUS = 1                // end of diagnosis
                    T_AST_DIAG_TCO_PLAUS = T_DIAG_AST
        endif(4a)
    else(3)         // only use of delay time based on relative temperature at start
        if(4b)      T_AST_MAX_DIAG_TCO_PLAUS = 0                          and
                    ( TCO_SUB > ID_TCO_MIN_DIAG_TCO_PLAUS  or
                    LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1 )
                    // default is LC_TCO_PLAUS_TCO_SUB_DIAG_INH = 1; this must also be kept
                    for US applications, exept lambda activation by TCO is circumvented
        then(4b)    ERR_SYM_TCO_PLAUS = "SYM_3"
                    LV_ERR_TCO_PLAUS = 1                     // without debounce
                    LV_END_DIAG_TCO_PLAUS = 1                // end of diagnosis
                    T_AST_DIAG_TCO_PLAUS = T_DIAG_AST
        endif(4b)
    endif(3)
    endif(2)
endif(1)
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_T_MIN_DIAG_TCO_PLAUS	6	1...FFFFH	1...65535	1	[s]
LDP_TIA_THR_ST_ID_T_TCO_PLAUS	6	0...FEH	-48...142.5	0.75	[°C]
Minimum time for coolant temperature plausibility diagnosis activation					
IP_T_MAX_DIAG_TCO_PLAUS	4	1...FFFFH	1...65535	1	[s]
LDP_TCO_DIF_DIAG_TCO_PLAUS_ID_T	4	0...FFH	-96...95.25	0.75	[°C]
Time interval to reach closed-loop enable temperature for coolant temperature plausibility diagnosis					
ID_TCO_MIN_DIAG_TCO_PLAUS	6*6	0...FEH	-48...142.5	0.75	[°C]
LDP_TCO_ST_ID_TCO_MIN_DIAG_TCO	6	0...FEH	-48...142.5	0.75	[°C]
LDP_RATIO_T_IS_AST_ID_TCO_DIAG	6	0...FFH	0...99.60937	0.390625	[%]
Minimum coolant temperature threshold for coolant temperature plausibility diagnosis					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TCO_PLAUS_TCO_SUB_DIAG_INH	1	0...1H	0...1	1	[-]
Logical calibration data to inhibit the use of TCO_SUB condition for error detection					
LC_T_DLY_TCO_PLAUS	1	0...1H	0...1	1	[-]
Logical calibration data to switch between delay timer based on absolute or relative temperature at start					

B.39 Coolant temperature sensor plausibility diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TCO_PLAUS	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature plausibility diagnosis inhibit					
LV_INH_DIAG_RBM_TCO_PLAUS	V	0...1H	0...1	1	[-]
Boolean for coolant temperature plausibility diagnosis inhibit due to OBD error					

Input data:

LV_ERR_TCO_EL	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_IGK
LV_ST_END	LV_ERR_TIA_THR	LV_ERR_TCO_STUCK_H	

FUNCTION DESCRIPTION:

General information:

The coolant temperature sensor plausibility diagnosis is performed at transition from the logical value for “Start” from “1” to “0”. If the engine stalls or one of the conditions for inhibiting the diagnosis is present, it will be stopped and starts again at the next engine start.

The diagnosis is finished as soon as the boolean for end of coolant temperature sensor plausibility diagnosis is set to “1”. In this case the boolean for coolant temperature plausibility error indicates either a not plausible (= 1) or a normal (= 0) coolant temperature sensor signal.

Depending on project specific requirements, the coolant temperature sensor plausibility diagnosis can be inhibited by setting of LV_INH_DIAG_TCO_PLAUS.

Application conditions:

*Initialisation at RESET **or** EXIT_ST **or** LV_IGK = 0 -> 1 **or** FMY clear:*

LV_INH_DIAG_TCO_PLAUS= 0

LV_INH_DIAG_RBM_TCO_PLAUS = 0

Recurrence: 1000 ms

Activation: LV_ST_END = 1

(Calculation of the TCO plausi. diagnosis interface parameter enabled)

Deactivation: LV_ST_END = 0

(Calculation of the TCO plausi. diagnosis interface parameter disabled)

Formula section:

Calculation of diagnosis interface parameter:

Note: the projects have to adapt their requirements to the interface parameter !

Default:

If	LV_ERR_TCO_EL = 1	**or**
	LV_ERR_TCO_GRD = 1	**or**
	LV_ERR_TCO_STUCK = 1	**or**
	LV_ERR_TCO_STUCK_H = 1	**or**
	LV_ERR_TIA_THR = 1	
then	LV_INH_DIAG_RBM_TCO_PLAUS = 1	
endif		

If	LV_IGK = 0	**or**
	LV_INH_DIAG_RBM_TCO_PLAUS = 1	**or**
	// project specific requirements = 1	
then	LV_INH_DIAG_TCO_PLAUS = 1	
endif		

Configuration for diagnostic symptoms :

Diagnostic TCO_PLAUS	Symptom description	Symptom	Filter type
Coolant temperature sensor plausibility diagnosis		*SYM_0*	*NO*
		SYM_1	
		SYM_2	
	Coolant temperature plausibility error	*SYM_3*	

Diagnosis	Symptom	Nr	P-Code/ Failure	P-Code/ Symptom	Recurence	Failure class A/B
Diagnosis description	*1st symptom description*	*0*				
	2nd symptom description	*1*				
	3rd symptom description	*2*				
Diagnosis name	*4th symptom description*	*3*				

B.39.1 Interface for Rate – Based - Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_TCO_PLAUS	V/O	0...7H	0...7	1	[-]
Interface of TCO_PLAUS monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TCO_PLAUS	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for TCO Plausibility Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_RBM_TCO_PLAUS	LV_INH_DIAG_RBM_TCO_PLAUS
		CTR_CDN_RBM_TCO_PLAUS	CTR_COMP_RBM_TCO_PLAUS

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TCO_PLAUS monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TCO_PLAUS data.

Within STATE_RBM_TCO_PLAUS, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (not valid for catalyst diagnosis)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_TCO_PLAUS = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1of STATE_RBM_TCO_PLAUS = 0

bit 2 of STATE_RBM_TCO_PLAUS = 1

on failure memory reset :

bit 1 of STATE_RBM_TCO_PLAUS = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_TCO_EL	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_ERR_TCO_STUCK_H

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TCO_PLAUS = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TCO_PLAUS = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TCO_PLAUS = 0

Then

If LV_END_DIAG_RBM_TCO_PLAUS = 1

Then bit 0 of STATE_RBM_TCO_PLAUS = 1

Endif

Endif

If bit 1 of STATE_RBM_TCO_PLAUS = 0

Then

If LV_INH_DIAG_RBM_TCO_PLAUS = 1

Then bit 1 of STATE_RBM_TCO_PLAUS = 1

Endif

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TCO_PLAUS = 0
then RATE_RBM_TCO_PLAUS = FFFFH
else RATE_RBM_TCO_PLAUS
= (CTR_COMP_RBM_TCO_PLAUS / CTR_CDN_RBM_TCO_PLAUS)
endif

B.40 Coolant temperature high sided rationality check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TCO_STUCK_H	V/O	0...1H	0...1	1	-
Boolean for coolant temperature stuck high error					
LV_CDN_DIAG_TCO_STUCK_H	V	0...1H	0...1	1	-
Boolean for coolant temperature stuck high diagnosis conditions					
ERR_SYM_TCO_STUCK_H	V	0H 8H	NO_SYM SYM_3	1	-
Symptom for coolant temperature stuck high error					
LV_END_DIAG_TCO_STUCK_H	V/O	0...1H	0...1	1	-
Boolean for end of coolant temperature stuck high diagnosis					
LV_END_DIAG_RBM_TCO_STUCK_H	V/O	0...1H	0...1	1	-
Boolean for end of coolant stuck high diagnosis for RBM					
LV_CDN_DIAG_RBM_TCO_STUCK_H	V	0...1H	0...1	1	-
Boolean for coolant temperature stuck high diagnosis conditions for RBM					
TCO_STUCK_DIF_L	V	0…FEH	-48…142.5	0.75	°C
Minimum detected TCO for coolant temperature stuck high diagnosis					
TCO_STUCK_DIF_H	V	0…FEH	-48…142.5	0.75	°C
Maximum detected TCO for coolant temperature stuck high diagnosis					
TCO_DIF_TCO_STUCK_H	V	0…FEH	0…190.5	0.75	°C
Detected TCO difference for coolant temperature stuck high diagnosis					
LV_TCO_STUCK_H_ENA_1	V	0…1H	0…1	1	-
Enable condition 1 (higher load) for coolant temperature stuck high diagnosis					
LV_TCO_STUCK_H_ENA_2	V	0…1H	0…1	1	-
Enable condition 2 (lower load) for coolant temperature stuck high diagnosis					
CTR_DLY_1_TCO_STUCK_H	V	0…FFFFH	0...65535	1	-
Enable condition 1 (higher load) delay counter for coolant temperature stuck high diagnosis					
CTR_DLY_2_TCO_STUCK_H	V	0…FFFFH	0...65535	1	-
Enable condition 2 (lower load) delay counter for coolant temperature stuck high diagnosis					

Input data:

TCO	TCO_ST	TIA_THR_ST	LV_ST_END
LV_CDN_VB_OBD_2	LV_INH_DIAG_TCO_STUCK_H	TAM	T_DIAG_AST
LV_PL	N	VS	MAF_KGH
LV_IS	LV_PUC	LV_FL	

FUNCTION DESCRIPTION:

General information:

The diagnosis is based on monitoring the alteration of the coolant temperature signal (TCO) (positive and negative change) to determine a malfunctioning sensor stuck signal at high coolant temperature values.

Even at high coolant temperature values at engine start, a smooth change of the coolant temperature signal can be observed. Either at special engine operating conditions after a specific time interval or due to temperature changes caused by the coolant thermostat.

The diagnosis is performed only once per engine run. The symptom of the error code is not handled by anti-bouncing.

Description:

Error-symptoms are defined to this diagnosis function as following:

```
┌─┬─┬─┬─┐
│ │ │ │ │
└─┴─┴─┴─┘
 │ │ │ └─> -                                        (= SYM_0)
 │ │ └───> -                                        (= SYM_1)
 │ └─────> -                                        (= SYM_2)
 └───────> Coolant temperature signal stuck high error  (= SYM_3)
```

Application conditions:

Initialisation at EXIT_ST:

LV_ERR_TCO_STUCK_H = 0

LV_CDN_DIAG_TCO_STUCK_H = 0

LV_CDN_DIAG_RBM_TCO_STUCK_H = 0

ERR_SYM_TCO_STUCK_H = “NO_SYM”

LV_END_DIAG_RBM_TCO_STUCK_H = 0

TCO_DIF_TCO_STUCK_H = 0°C

TCO_STUCK_DIF_L = 142.5°C

TCO_STUCK_DIF_H = -48°C

LV_TCO_STUCK_H_ENA_1 = 0

LV_TCO_STUCK_H_ENA_2 = 0

CTR_DLY_1_TCO_STUCK_H = IP_CTR_DLY_1_TCO_STUCK_H

CTR_DLY_2_TCO_STUCK_H = IP_CTR_DLY_2_TCO_STUCK_H

// LV_END_DIAG_TCO_STUCK_H is directly initialized by ERRM

Recurrence: 1000 ms

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Formula section:

Calculation of RBM diagnosis condition

IF	LV_INH_DIAG_TCO_STUCK_H = 0	**AND**
	LV_END_DIAG_RBM_TCO_STUCK_H = 0	**AND**
	LV_CDN_VB_OBD_2 = 1	**AND**
	TCO_ST > C_TCO_ST_MIN_TCO_STUCK_H	**AND**
	TIA_THR_ST > C_TIA_THR_ST_MIN_TCO_STUCK_H	**AND**
	TIA_THR_ST < C_TIA_THR_ST_MAX_TCO_STUCK_H	**AND**
	TAM > C_TAM_MIN_TCO_STUCK_H	

THEN LV_CDN_DIAG_RBM_TCO_STUCK_H = 1

ELSE LV_CDN_DIAG_RBM_TCO_STUCK_H = 0

ENDIF

Calculation of diagnosis condition

IF LV_CDN_DIAG_RBM_TCO_STUCK_H = 1 **AND**
LV_END_DIAG_TCO_STUCK_H = 0

THEN LV_CDN_DIAG_TCO_STUCK_H = 1

ELSE LV_CDN_DIAG_TCO_STUCK_H = 0

ENDIF

Calculation of min. / max. TCO value and TCO deviation

IF TCO > TCO_STUCK_DIF_H

THEN TCO_STUCK_DIF_H = TCO

ENDIF

IF TCO < TCO_STUCK_DIF_L

THEN TCO_STUCK_DIF_L = TCO

ENDIF

TCO_DIF_TCO_STUCK_H = TCO_STUCK_DIF_H - TCO_STUCK_DIF_L

Calculation of Diagnosis enable condition:

IF(1) LV_CDN_DIAG_RBM_TCO_STUCK_H = 1

THEN(1) (*calculation of formula section A, B, C, D – "SYM_3" error check)*

A Comparison of TCO_deviation to end the diagnosis

IF(2a) LV_CDN_DIAG_TCO_STUCK_H = 1 **AND**
TCO_DIF_TCO_STUCK_H >= C_TCO_DIF_MIN_TCO_STUCK_H

THEN(2a) LV_END_DIAG_TCO_STUCK_H = 1 *// end of diagnosis*

ENDIF(2a)

B Calculation of delay counter 1 to reach diagnosis enable conditions

IF(2b) LV_TCO_STUCK_H_ENA_1 = 0

THEN(2b)

IF(3b) **(** LV_PL = 1 **OR** LV_FL = 1 **)** **AND**
N > C_N_MIN_TCO_STUCK_H **AND**
VS > C_VS_MIN_TCO_STUCK_H **AND**
MAF_KGH > C_MAF_KGH_MIN_TCO_STUCK_H

THEN(3b) CTR_DLY_1_TCO_STUCK_H **--**

```
                    // conditions 1 causing change in TCO are satisfied, decrement counter
            IF(4b)          CTR_DLY_1_TCO_STUCK_H = 0
            THEN(4b)        LV_TCO_STUCK_H_ENA_1 = 1
            ENDIF(4b)
        ELSE(3b)        CTR_DLY_1_TCO_STUCK_H = CTR_DLY_1_TCO_STUCK_H
                            + C_CTR_DLY_INC_TCO_STUCK_H
                    // conditions 1 causing change in TCO not fulfilled, increment counter
            IF(5b)          CTR_DLY_1_TCO_STUCK_H >=
                                IP_CTR_DLY_1_TCO_STUCK_H
            THEN(5b)        CTR_DLY_1_TCO_STUCK_H =
                                IP_CTR_DLY_1_TCO_STUCK_H
            ENDIF(5b)
        ENDIF(3b)
    ELSE(2b)        CTR_DLY_1_TCO_STUCK_H = CTR_DLY_1_TCO_STUCK_H   //freeze
    ENDIF(2b)
```

C Calculation of delay counter 2 to reach diagnosis enable conditions

```
    IF(2c)                  LV_TCO_STUCK_H_ENA_2 = 0
    THEN(2c)
        IF(3c)          ( LV_IS = 1 AND VS < C_VS_MAX_TCO_STUCK_H )     OR
                        LV_PUC = 1
        THEN(3c)        CTR_DLY_2_TCO_STUCK_H --
                    // conditions 2 causing change in TCO are satisfied, decrement counter
            IF(4c)          CTR_DLY_2_TCO_STUCK_H = 0
            THEN(4c)        LV_TCO_STUCK_H_ENA_2 = 1
            ENDIF(4c)
        ELSE(3c)        CTR_DLY_2_TCO_STUCK_H = CTR_DLY_2_TCO_STUCK_H
                            + C_CTR_DLY_INC_TCO_STUCK_H
                    // conditions 2 causing change in TCO not fulfilled, increment counter
            IF(5c)          CTR_DLY_2_TCO_STUCK_H >=
                                IP_CTR_DLY_2_TCO_STUCK_H
            THEN(5c)        CTR_DLY_2_TCO_STUCK_H =
                                IP_CTR_DLY_2_TCO_STUCK_H
            ENDIF(5c)
        ENDIF(3c)
    ELSE(2c)        CTR_DLY_2_TCO_STUCK_H = CTR_DLY_2_TCO_STUCK_H   //freeze
    ENDIF(2c)
```

D Start diagnostic window / Diagnostic check:

```
IF(2d)       LV_TCO_STUCK_H_ENA_1 = 1                                AND
             LV_TCO_STUCK_H_ENA_2 = 1                                AND
             T_DIAG_AST > C_T_DIAG_AST_MIN_TCO_STUCK_H
THEN(2d)
    IF(3d)       LV_END_DIAG_TCO_STUCK_H = 0
    THEN(3d)     ERR_SYM_TCO_STUCK_H = "SYM_3"
                 LV_ERR_TCO_STUCK_H = 1          // without debounce
                 LV_END_DIAG_TCO_STUCK_H = 1     // end of diagnosis
    ENDIF(4d)
    LV_END_DIAG_RBM_TCO_STUCK_H = 1              // end of diagnosis for RBM
ENDIF(2d)
```

ELSE(1) *(TCO_STUCK_H diagnosis finished or diagnosis conditions not met)*

ENDIF(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_CTR_DLY_1_TCO_STUCK_H	6	0…FFFFH	0...65535	1	-
LDPM_TIA_THR_ST_IP_CTR_STUCK_H	6	0…FEH	-48…142.5	0.75	°C
Delay counter 1 start value for TCO_STUCK_H rationality check					
IP_CTR_DLY_2_TCO_STUCK_H	6	0…FFFFH	0...65535	1	-
LDPM_TIA_THR_ST_IP_CTR_STUCK_H	6	0…FEH	-48…142.5	0.75	°C
Delay counter 2 start value for TCO_STUCK_H rationality check					
C_TCO_ST_MIN_TCO_STUCK_H	1	0…FEH	-48…142.5	0.75	°C
Min. TCO_ST to activate TCO_STUCK_H rationality check					
C_TIA_THR_ST_MIN_TCO_STUCK_H	1	0…FEH	-48…142.5	0.75	°C
Min. TIA_THR_ST to activate TCO_STUCK_H rationality check					
C_TIA_THR_ST_MAX_TCO_STUCK_H	1	0…FEH	-48…142.5	0.75	°C
Max. TIA_THR_ST to activate TCO_STUCK_H rationality check					
C_TAM_MIN_TCO_STUCK_H	1	0...FEH	-48...142.5	0.75	°C
Min. TAM to activate TCO_STUCK_H rationality check					
C_TCO_DIF_MIN_TCO_STUCK_H	1	0…FEH	0…190.5	0.75	°C
Min. TCO difference to enable TCO_STUCK_H rationality check					
C_N_MIN_TCO_STUCK_H	1	0…1FE0H	0…8160	1	rpm
Min. N for calculation of enable condition 1 for TCO_STUCK_H rationality check					
C_VS_MIN_TCO_STUCK_H	1	0…FFH	0…255	1	km/h
Min. VS for calculation of enable condition 1 for TCO_STUCK_H rationality check					
C_MAF_KGH_MIN_TCO_STUCK_H	1	0...FFFFH	0...2047.96875	0.03125	kg/h
Min. MAF_KGH for calculation of enable condition 1 for TCO_STUCK_H rationality check					
C_VS_MAX_TCO_STUCK_H	1	0…FFH	0…255	1	km/h
Min. VS for calculation of enable condition 2 for TCO_STUCK_H rationality check					
C_CTR_DLY_INC_TCO_STUCK_H	1	0...FFH	0...255	1	-
Delay counter increment value for TCO_STUCK_H rationality check					
C_T_DIAG_AST_MIN_TCO_STUCK_H	1	0…FFFFH	0...65535	1	s
Min. T_DIAG_AST to enable TCO_STUCK_H rationality check					

B.41 Coolant temperature high sided rationality check (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TCO_STUCK_H	O/V	0...1H	0...1	1	-
Boolean for coolant temperature stuck high diagnosis conditions inhibit					
LV_INH_DIAG_RBM_TCO_STUCK_H	V	0...1H	0...1	1	-
Boolean for coolant temperature stuck high diagnosis conditions inhibit due to OBD error					

Input data:

LV_IGK	LV_ERR_TCO_PLAUS	LV_ERR_TCO_EL	LV_ERR_TCO_GRD
LV_ERR_TCO_STUCK	LV_ERR_TIA_THR	LV_ERR_TAM	LV_ST_END
LV_ERR_VS	LV_ERR_MAF	LV_ERR_CAM	LV_ERR_CRK
LV_ERR_TCO_STUCK_RNG	LV_ERR_MAP		

FUNCTION DESCRIPTION:

General information:

Task of the coolant temperature high sided rationality check (Appl. Inc.) is to allow the projects to adapt their specific requirements to generic TCO sensor stuck high diagnosis. The setting of the interface output variables have to be provided by the projects.

Application conditions:

Initialisation: see separate chapter
Recurrence: see separate chapter
Activation: at every engine operating state
Deactivation: -

B.41.1 Coolant temperature high sided rationality check - interface parameter

FUNCTION DESCRIPTION:

General information:

Depending on project specific requirements, the coolant temperature high sided rationality check can be inhibited by setting of the logical variable LV_INH_DIAG_TCO_STUCK_H. As a default value LV_INH_DIAG_TCO_STUCK_H is set as shown below.

Application conditions:

Initialisation at RESET ***or*** *EXIT_ST* ***or*** *LV_IGK = 0 -> 1* ***or*** *FMY clear:*

LV_INH_DIAG_TCO_STUCK_H = 0

LV_INH_DIAG_RBM_TCO_STUCK_H = 0

Recurrence: 1000 ms

Activation: LV_ST_END = 1

(Calculation of the TCO_STUCK_H diag. interface parameter enabled)

Deactivation: LV_ST_END = 0

(Calculation of the TCO_STUCK_H diag. interface parameter disabled)

Formula section:

Calculation of diagnosis interface parameter:

If LV_ERR_TCO_EL = 1 **or**
LV_ERR_TCO_GRD = 1 **or**
LV_ERR_TCO_PLAUS = 1 **or**
LV_ERR_TCO_STUCK = 1 **or**
LV_ERR_TCO_STUCK_RNG = 1 **or**
LV_ERR_TIA_THR = 1 **or**
LV_ERR_TAM = 1 **or**
LV_ERR_VS = 1 **or**
LV_ERR_MAF = 1 **or**
LV_ERR_MAP = 1 **or**
LV_ERR_CAM = 1 **or**
LV_ERR_CRK = 1
then LV_INH_DIAG_RBM_TCO_STUCK_H = 1
endif

If LV_IGK = 0 **or**
LV_INH_DIAG_RBM_TCO_STUCK_H = 1
then LV_INH_DIAG_TCO_STUCK_H = 1
endif

Configuration for diagnostic symptoms :

Diagnostic TCO_STUCK_H	**Symptom description**	**Symptom**	**Filter type**
Coolant temperature high sided rationality check		*SYM_0*	*NO*
		SYM_1	
		SYM_2	
	Coolant temperature signal stuck high error	*SYM_3*	

List of Environmental Data to store in Failure Memory: TCO_ST
DIAG_INST_ENVD
TCO_MES_DIF_DIAG_TCO_STUCK
TCO_SUB_DIF_DIAG_TCO_STUCK

B.42 Interface for RBM – TCO sensor stuck high diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_TCO_STUCK_H	O/V	0...7H	0...7	1	-
Interface of TCO_STUCK_H monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TCO_STUCK_H	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for TCO Stuck High Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_RBM_TCO_STUCK_H	LV_INH_DIAG_RBM_TCO_STUCK_H
CTR_CDN_RBM_TCO_STUCK_H	CTR_COMP_RBM_TCO_STUCK_H		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TCO_STUCK_H monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TCO_STUCK_H data.

Within STATE_RBM_TCO_STUCK_H, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (not valid for catalyst diagnosis)

Application conditions:

Initialisation:

at ECU reset:

bit 0, bit 1 and bit 2 of STATE_RBM_TCO_STUCK_H = 0

at LV_DC 0 → 1 transition:

bit 0 and bit 1of STATE_RBM_TCO_STUCK_H = 0

bit 2 of STATE_RBM_TCO_STUCK_H = 1

on failure memory reset:

bit 1 of STATE_RBM_TCO_STUCK_H = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once:

LV_ERR_TCO_EL	LV_ERR_TCO_PLAUS	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK
LV_ERR_TCO_STUCK_RNG			

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TCO_STUCK_H = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TCO_STUCK_H = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TCO_STUCK_H = 0

Then

If LV_END_DIAG_RBM_TCO_STUCK_H = 1

Then bit 0 of STATE_RBM_TCO_STUCK_H = 1

Endif

Endif

If bit 1 of STATE_RBM_TCO_STUCK_H = 0

Then

  If LV_INH_DIAG_RBM_TCO_STUCK_H = 1

  Then bit 1 of STATE_RBM_TCO_STUCK_H = 1

  Endif

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TCO_STUCK_H = 0
then RATE_RBM_TCO_STUCK_H = FFFFH
else RATE_RBM_TCO_STUCK_H
= (CTR_COMP_RBM_TCO_STUCK_H / CTR_CDN_RBM_TCO_STUCK_H)
endif

B.43 Coolant temperature low sided rationality check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TCO_STUCK	V/O	0...1H	0...1	1	[-]
TCO Stuck signal detected					
ERR_SYM_TCO_STUCK	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom for coolant temperature stuck signal error					
LV_CDN_DIAG_TCO_STUCK	V	0...1H	0...1	1	[-]
Boolean for coolant temperature stuck signal diagnosis conditions					
LV_END_DIAG_TCO_STUCK	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature stuck signal diagnosis					
TCO_MES_DIF_DIAG_TCO_STUCK	V	0...FEH	0...190.5	0.75	[°C]
Difference of measured coolant temperature values for coolant temperature stuck signal diagnosis					
TCO_MES_MIN_DIAG_TCO_STUCK	V	0...FEH	-48...142.5	0.75	[°C]
Measured coolant temperature minimum value for coolant temperature stuck signal diagnosis					
TCO_MES_MAX_DIAG_TCO_STUCK	V	0...FEH	-48...142.5	0.75	[°C]
Measured coolant temperature maximum value for coolant temperature stuck signal diagnosis					
TCO_SUB_DIF_DIAG_TCO_STUCK	V	0...FEH	0...190.5	0.75	[°C]
Difference of substitute coolant temperature values for coolant temperature stuck signal diagnosis					
TCO_SUB_MIN_DIAG_TCO_STUCK	V	0...FEH	-48...142.5	0.75	[°C]
Substitute coolant temperature minimum value for coolant temperature stuck signal diagnosis					
TCO_SUB_MAX_DIAG_TCO_STUCK	V	0...FEH	-48...142.5	0.75	[°C]
Substitute coolant temperature maximum value for coolant temperature stuck signal diagnosis					

Input data:

TCO_MES	TCO_ST	TCO_SUB	LV_ST_END
LV_CDN_VB_MIN_DIAG	LV_INH_DIAG_TCO_STUCK		

FUNCTION DESCRIPTION:

General information:

The purpose of the signal stuck diagnosis is to detect a stuck measured coolant temperature sensor signal. The diagnosis function checks if after a variation of the calculated coolant temperature substitute value (TCO_SUB) also a variation of the measured coolant temperature value (TCO_MES) is detected.

The diagnosis is performed only once per engine run. The symptom of the error code is not handled by anti-bouncing.

Description:

Error-symptoms are defined to this diagnosis function as following:

```
□□□□
│ │ │ └─> -                                        (= SYM_0)
│ │ └───> -                                        (= SYM_1)
│ └─────> -                                        (= SYM_2)
└───────> Coolant temperature signal stuck error   (= SYM_3)
```

Application conditions:

Initialisation at Exit start (EXIT_ST):

LV_ERR_TCO_STUCK = 0
LV_CDN_DIAG_TCO_STUCK = 0
ERR_SYM_TCO_STUCK = “NO_SYM”
TCO_MES_DIF_DIAG_TCO_STUCK = 0°C
TCO_SUB_DIF_DIAG_TCO_STUCK = 0°C
TCO_MES_MIN_DIAG_TCO_STUCK = 142.5°C
TCO_MES_MAX_DIAG_TCO_STUCK = -48°C
TCO_SUB_MIN_DIAG_TCO_STUCK = TCO_SUB
TCO_SUB_MAX_DIAG_TCO_STUCK = TCO_SUB

// LV_END_DIAG_TCO_STUCK is directly initialized by ERRM

Recurrence: 1000 ms

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Formula section:

If LV_CDN_VB_MIN_DIAG = 1 **and**
LV_INH_DIAG_TCO_STUCK = 0 **and**
LV_END_DIAG_TCO_STUCK = 0 **and**
TCO_ST < C_TCO_ST_MAX_TCO_STUCK
then LV_CDN_DIAG_TCO_STUCK = 1
else LV_CDN_DIAG_TCO_STUCK = 0
endif

IF(1) LV_CDN_DIAG_TCO_STUCK = 1
THEN(1)
Calculation of the coolant temperature-boundary values:
If(2) TCO_MES < TCO_MES_MIN_DIAG_TCO_STUCK
then(2) TCO_MES_MIN_DIAG_TCO_STUCK = TCO_MES
else(2) TCO_MES_MIN_DIAG_TCO_STUCK = TCO_MES_MIN_DIAG_TCO_STUCK
endif(2)
If(2) TCO_MES > TCO_MES_MAX_DIAG_TCO_STUCK
then(2) TCO_MES_MAX_DIAG_TCO_STUCK = TCO_MES
else(2) TCO_MES_MAX_DIAG_TCO_STUCK = TCO_MES_MAX_DIAG_TCO_STUCK
endif(2)
If(2) TCO_SUB > TCO_SUB_MAX_DIAG_TCO_STUCK
then(2) TCO_SUB_MAX_DIAG_TCO_STUCK = TCO_SUB

else(2) TCO_SUB_MAX_DIAG_TCO_STUCK = TCO_SUB_MAX_DIAG_TCO_STUCK
endif(2)

Calculation of the coolant temperature-difference values:

TCO_MES_DIF_DIAG_TCO_STUCK =
| TCO_MES_MAX_DIAG_TCO_STUCK – TCO_MES_MIN_DIAG_TCO_STUCK **|**

TCO_SUB_DIF_DIAG_TCO_STUCK =
| TCO_SUB_MAX_DIAG_TCO_STUCK – TCO_SUB_MIN_DIAG_TCO_STUCK **|**

Error detection:

If (2) TCO_SUB_DIF_DIAG_TCO_STUCK >
IP_TCO_SUB_DIF_DIAG_TCO_STUCK
then (2)
if (3) TCO_MES_DIF_DIAG_TCO_STUCK <
IP_TCO_MES_DIF_DIAG_TCO_STUCK
then (3) LV_ERR_TCO_STUCK = 1 (without debounce)
ERR_SYM_TCO_STUCK = “SYM_3”
LV_END_DIAG_TCO_STUCK = 1
else (3) LV_ERR_TCO_STUCK = 0 (without debounce)
ERR_SYM_TCO_STUCK = “NO_SYM”
LV_END_DIAG_TCO_STUCK = 1
endif (3)
endif (2)
ENDIF(1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TCO_SUB_DIF_DIAG_TCO_STUCK	6	0...FEH	0...190.5	0.75	[°C]
LDPM_TCO_ST_1_ENTE	6	0...FEH	-48...142.5	0.75	[°C]
Minimum increase of the TCO_SUB value for the coolant temperature stuck signal diagnosis					
IP_TCO_MES_DIF_DIAG_TCO_STUCK	6	0...FEH	0...190.5	0.75	[°C]
LDPM_TCO_ST_1_ENTE	6	0...FEH	-48...142.5	0.75	[°C]
Minimum increase of the TCO value for the coolant temperature stuck signal diagnosis					
C_TCO_ST_MAX_TCO_STUCK	1	0...FEH	-48...142.5	0.75	[°C]
Maximum TCO_ST value to activate temperature stuck signal diagnosis					

B.44 Coolant temperature low sided rationality check (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TCO_STUCK	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature stuck signal diagnosis inhibit					
LV_INH_DIAG_RBM_TCO_STUCK	V	0...1H	0...1	1	[-]
Boolean for temperature stuck signal diagnosis inhibit due to OBD error					

Input data:

LV_IGK	LV_ERR_TCO_EL	LV_ERR_TCO_GRD	LV_ERR_TCO_PLAUS
LV_ST_END	LV_ERR_TCO_STUCK_H	LV_ERR_TCO_STUCK_RNG	

FUNCTION DESCRIPTION:

General information:

Task of the coolant temperature sensor diagnosis (Appl. Inc.) is to allow the projects to adapt their specific requirements to generic TCO sensor stuck low diagnosis. The setting of the interface output variables have to be provided by the projects.

Application conditions:

Initialisation: see separate chapter

Recurrence: see separate chapter

Activation: at every engine operating state

Deactivation: -

B.44.1 Coolant temperature low sided rationality check - interface parameter

FUNCTION DESCRIPTION:

General information:

Depending on project specific requirements, the signal stuck diagnosis of the main coolant temperature sensor can be inhibited by setting of LV_INH_DIAG_TCO_STUCK.

Application conditions:

*Initialisation at RESET **or** EXIT_ST **or** LV_IGK = 0 -> 1 **or** FMY clear:*

LV_INH_DIAG_TCO_STUCK = 0

LV_INH_DIAG_RBM_TCO_STUCK = 0

Recurrence: 1000 ms

Activation: LV_ST_END = 1
(Calculation of the TCO stuck diagnosis interface parameter enabled)

Deactivation: LV_ST_END = 0
(Calculation of the TCO stuck diagnosis interface parameter disabled)

Formula section:

Calculation of diagnosis interface parameter:

Note: **the projects have to adapt their requirements to the interface parameter !**

Default:

If LV_ERR_TCO_EL = 1 **or**
LV_ERR_TCO_GRD = 1 **or**
LV_ERR_TCO_PLAUS = 1 **or**
LV_ERR_TCO_STUCK_H = 1 **or**
LV_ERR_TCO_STUCK_RNG = 1
then LV_INH_DIAG_RBM_TCO_STUCK = 1
endif

If LV_INH_DIAG_RBM_TCO_STUCK = 1
then LV_INH_DIAG_TCO_STUCK = 1
endif

Configuration for diagnostic symptoms :

Diagnostic TCO_STUCK	Symptom description	Symptom	Filter type
Coolant temperature low sided rationality check		*SYM_0*	*NO*
		SYM_1	
		SYM_2	
	Coolant temperature signal stuck error	*SYM_3*	

B.44.2 Interface for Rate – Based - Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_TCO_STUCK	V/O	0...7H	0...7	1	[-]
Interface of TCO_STUCK monitor with the Rate-Based Monitoring statistics					
RATE_RBM_TCO_STUCK	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for TCO Stuck Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_TCO_STUCK	LV_INH_DIAG_RBM_TCO_STUCK
CTR_CDN_RBM_TCO_STUCK	CTR_COMP_RBM_TCO_STUCK		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TCO_STUCK monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TCO_STUCK data.

Within STATE_RBM_TCO_STUCK, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0) (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1) (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2) (not valid for catalyst diagnosis)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_TCO_STUCK = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1of STATE_RBM_TCO_STUCK = 0

bit 2 of STATE_RBM_TCO_STUCK = 1

on failure memory reset :

bit 1 of STATE_RBM_TCO_STUCK = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_TCO_EL	LV_ERR_TCO_PLAUS	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK_H
LV_ERR_TCO_STUCK_RNG			

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TCO_STUCK = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TCO_STUCK = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TCO_STUCK = 0

Then

If LV_END_DIAG_TCO_STUCK = 1

Then bit 0 of STATE_RBM_TCO_STUCK = 1

Endif

Endif

If bit 1 of STATE_RBM_TCO_STUCK = 0

Then

If LV_INH_DIAG_RBM_TCO_STUCK = 1

Then bit 1 of STATE_RBM_TCO_STUCK = 1

Endif

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TCO_STUCK = 0
then RATE_RBM_TCO_STUCK = FFFFH
else RATE_RBM_TCO_STUCK
= (CTR_COMP_RBM_TCO_STUCK / CTR_CDN_RBM_TCO_STUCK)
endif

B.45 TCO sensor stuck in range plausibility diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TCO_STUCK_RNG	V/O	0...1H	0...1	1	[-]
Boolean for coolant temperature signal stuck in range detected					
LV_TCO_STUCK_RNG_CHK_END	V	0...1H	0...1	1	[-]
Boolean for coolant temperature stuck in range check finished					
LV_END_DIAG_TCO_STUCK_RNG	V/O	0...1H	0...1	1	[-]
Boolean for end of coolant temperature signal stuck in range diagnosis					
ERR_SYM_TCO_STUCK_RNG	V/O	0H 8H	NO_SYM SYM_3	1	[-]
Boolean for end of coolant temperature signal stuck in range diagnosis					

Input data:

LV_INH_DIAG_TCO_STUCK_RNG	LV_ST_END	TOIL_STOP	TCO_STOP
LV_T_ES_NOT_PLAUS	TOIL_MES	TCO_MES	T_ES
CONF_TOIL_MDL	TIA_ST	CYC_ST	

FUNCTION DESCRIPTION:

General information:

For OBDII systems, it is required by CARB to check if Coolant Temperature " Stuck in Range Above the Lowest Maximum Enable Temperature". That means, ECU shall detect a malfunction if the ECT sensor inappropriately indicates a temperature above the lowest maximum enable temperature required by the OBD II system to enable other diagnostics (e.g., an OBD II system that requires ECT to be less than 90 degrees Fahrenheit at engine start to enable a diagnostic must detect malfunctions that cause the ECT sensor to inappropriately indicate a temperature above 90 degrees Fahrenheit).

In case the system is equipped with an Oil Temperature sensor this diagnostic compares coolant temperature and oil / intake air temperature at Start : if both oil and intake air temperature indicate a low value (e.g. 35°C), but coolant temperature indicates a higher value (e.g. 55°C), then a failure is detected.
This diagnostic must be used only for systems where it is ensured that TOIL is cooling down quicker than TCO.
In case the system is not equipped with TOIL sensor the cooling down behavior of the engine is checked depending on the engine off time. At cooled down engine the Coolant Temperature may not exceed a threshold dependent on TIA_ST.

Application conditions:

Initialisation: *At RESET,*

LV_ERR_TCO_STUCK_RNG = 0

LV_END_DIAG_TCO_STUCK_RNG = 0

LV_TCO_STUCK_RNG_CHK_END = 0

Recurrence: 100 msec

Activation: every engine state

Deactivation: Activation condition not true

Formula section:

```
if          (LV_ST_END = 0-->1  or  CYC_ST > C_CYC_ST_MAX_TCO_STUCK_RNG)
    and     LV_TCO_STUCK_RNG_CHK_END = 0
then        LV_TCO_STUCK_RNG_CHK_END = 1
        if          CONF_TOIL_MDL = 0
                        engine is equipped with TOIL Sensor
        then        if          LV_INH_DIAG_TCO_STUCK_RNG = 0
                        and     TOIL_STOP > C_TOIL_STOP_MIN_STUCK_RNG
                        and     TCO_STOP > C_TCO_STOP_MIN_STUCK_RNG
                        and     TIA_ST > C_TIA_ST_MIN_TCO_STUCK_RNG
                        and     TIA_ST < C_TIA_ST_MAX_TCO_STUCK_RNG
                        and     TOIL_MES < C_TOIL_MES_MAX_STUCK_RNG
                    then    if          TCO_MES > C_TCO_MES_DIAG_STUCK_RNG
                            then        LV_ ERR_TCO_STUCK_RNG = 1
                                        ERR_SYM_TCO_STUCK_RNG = "SYM_3"
                                        LV_END_DIAG_TCO_STUCK_RNG = 1
                            else        LV_ ERR_TCO_STUCK_RNG = 0
                                        ERR_SYM_TCO_STUCK_RNG = "NO_SYM"
                                        LV_END_DIAG_TCO_STUCK_RNG = 1
                            endif
                    else        No failure detection possible (LV_ERR_TCO_STUCK_RNG
                                / LV_END_DIAG_TCO_STUCK_RNG remains equal to 0)
                    endif
        endif
        if          CONF_TOIL_MDL =! 0
                    engine is not equipped with TOIL-Sensor
        then        if          LV_INH_DIAG_TCO_STUCK_RNG = 0
                        and     TCO_STOP > C_TCO_STOP_MIN_STUCK_RNG
                        and     TIA_ST  > C_TIA_ST_MIN_TCO_STUCK_RNG
                        and     TIA_ST  < C_TIA_ST_MAX_TCO_STUCK_RNG
                        and     LV_T_ES_NOT_PLAUS = 0
                        and     T_ES > C_T_ES_MIN_TCO_STUCK_RNG
                    then        if      TCO_MES > IP_TCO_STUCK_RNG
                                 then   LV_ ERR_TCO_STUCK_RNG = 1
                                        ERR_SYM_TCO_STUCK_RNG = "SYM_3"
                                        LV_END_DIAG_TCO_STUCK_RNG = 1
```

else LV_ ERR_TCO_STUCK_RNG = 0

ERR_SYM_TCO_STUCK_RNG = "NO_SYM"

LV_END_DIAG_TCO_STUCK_RNG = 1

endif

else No failure detection possible *(LV_ERR_TCO_STUCK_RNG / LV_END_DIAG_TCO_STUCK_RNG remains equal to 0)*

endif

endif

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TOIL_STOP_MIN_STUCK_RNG	1	0…C8H	-40…160	1	°C
Minimum TOIL engine stop temperature to perform TCO Stuck in range diagnosis					
C_TCO_STOP_MIN_STUCK_RNG	1	0…FEH	-48…142.5°C	0.75	°C
Minimum TCO engine stop temperature to perform TCO Stuck in range diagnosis					
C_TOIL_MES_MAX_STUCK_RNG	1	0…C8H	-40…160	1	°C
Maximum engine oil temperature to perform TCO Stuck in range diagnosis					
C_TCO_MES_DIAG_STUCK_RNG	1	0…FEH	-48…142.5°C	0.75	°C
Failure detection threshold for TCO Stuck in range					
C_T_ES_MIN_TCO_STUCK_RNG	1	0...FFFFH	0...65535	1	min
Constant for minimum requried engine off time to perform TCO Stuck in range diagnosis					
C_TIA_ST_MIN_TCO_STUCK_RNG	1	0...FEH	-48...142.5	0.75	°C
Constant for minimun Air Temperature at Start to perform TCO_Stuck in range diagnosis					
C_TIA_ST_MAX_TCO_STUCK_RNG	1	0...FEH	-48...142.5	0.75	°C
Constant for maximum Air Temperature at Start to perform TCO_Stuck in range diagnosis					
C_CYC_ST_MAX_TCO_STUCK_RNG	1	0...FFH	0...255	1	-
Maximum Cycles during start before TCO_STUCK_RNG diagnosis is performed					
IP_TCO_STUCK_RNG	8*8	0...FEH	-48...142.5	0.75	°C
LDP_TIA_ST_IP_TCO_STUCK_RNG	8	0...FEH	-48...142.5	0.75	°C
LDP_T_ES_IP_TCO_STUCK_RNG	8	0...FFFFH	0...65535	1	min
Failure detection threshold for TCO Stuck in range dependent of TIA_ST and T_ES					

B.46 TCO sensor stuck in range plausibility diagnosis (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TCO_STUCK_RNG	V/O	0...1H	0...1	1	[-]
Boolean for inhibition of coolant temperature signal stuck in range diagnosis					

Input data:

ERR_SYM_TOIL	ERR_SYM_TCO_EL	ERR_SYM_EL_TIA	ERR_SYM_INTM_TIA
LV_ST	ERR_SYM_TCO_GRD	LV_ERR_TOIL_PLAUS_L	LV_ERR_TOIL_PLAUS_H
LV_ERR_TOIL_STUCK			

B.46.1 Coolant temperature sensor stuck in range diagnosis - interface parameter

FUNCTION DESCRIPTION:

General information:

Application conditions:

Initialisation: LV_INH_DIAG_TCO_STUCK_RNG = 0

Recurrence: 100 ms

Activation: LV_ST = 1

Deactivation: LV_ST = 0

Formula section:

If ERR_SYM_TOIL = "NO_SYM"
and ERR_SYM_TCO_EL = "NO_SYM"
and ERR_SYM_TCO_GRD = "NO_SYM"
and ERR_SYM_EL_TIA_IM_CYL = "NO_SYM"
and ERR_SYM_INTM_TIA_IM_CYL = "NO_SYM"
and LV_ERR_TOIL_PLAUS_L = 0
and LV_ERR_TOIL_PLAUS_H = 0
and LV_ERR_TOIL_STUCK = 0

Then LV_INH_DIAG_TCO_STUCK_RNG = 0

Else LV_INH_DIAG_TCO_STUCK_RNG = 1

Configuration for diagnostic symptoms:

Diagnostic TCO_STUCK_RNG	Symptom description	Symptom	Filter type
TCO sensor stuck in range diagnosis			No
	TCO sensor stuck in range	*SYM_3*	

B.47 Interface to RBM - MAF deviation monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_TCO_STUCK_RNG	O/V	0...FFH	0...255	1	-
Interface of TCO_STUCK_RNG monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TCO_STUCK_RNG	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for TCO Stuck in Range Diagnosis , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_TCO_STUCK_RNG	LV_INH_DIAG_TCO_STUCK_RNG
CTR_CDN_RBM_TCO_STUCK_RNG	CTR_COMP_RBM_TCO_STUCK_RNG		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TCO_STUCK_RNG monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TCO_STUCK_RNG data.

Within STATE_RBM_TCO_STUCK_RNG, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (not valid for catalyst diagnosis)

Application conditions:

Initialisation:

at ECU reset:

bit 0, bit 1 and bit 2 of STATE_RBM_TCO_STUCK_RNG = 0

at LV_DC 0 → 1 transition:

bit 0 and bit 1of STATE_RBM_TCO_STUCK_RNG = 0

bit 2 of STATE_RBM_TCO_STUCK_RNG = 1

on failure memory reset:

bit 1 of STATE_RBM_TCO_STUCK_RNG = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once:

LV_ERR_TOIL	LV_ERR_TCO_EL	LV_ERR_EL_TIA_IM_CYL	LV_ERR_INTM_TIA_IM_CYL
LV_ERR_TCO_GRD			

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TCO_STUCK_RNG = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TCO_STUCK_RNG = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TCO_STUCK_RNG = 0

Then

If LV_END_DIAG_TCO_STUCK_RNG = 1

Then bit 0 of STATE_RBM_TCO_STUCK_RNG = 1

Endif

Endif

If bit 1 of STATE_RBM_TCO_STUCK_RNG = 0

Then

If LV_INH_DIAG_TCO_STUCK_RNG = 1

Then bit 1 of STATE_RBM_TCO_STUCK_RNG = 1

Endif

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TCO_STUCK_RNG = 0
then RATE_RBM_TCO_STUCK_RNG = FFFFH
else RATE_RBM_TCO_STUCK_RNG
= (CTR_COMP_RBM_TCO_STUCK_RNG / CTR_CDN_RBM_TCO_STUCK_RNG)
endif

B.48 Coolant thermostat monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TH	V	0...1H	0...1	1	-
Boolean for coolant thermostat error					
ERR_SYM_TH	V	0H 8H	NO_SYM SYM_3	1	-
Symptom for coolant thermostat error					
LV_CDN_DIAG_TH	V	0...1H	0...1	1	-
Boolean for coolant thermostat diagnosis conditions					
LV_END_DIAG_TH	O/V	0...1H	0...1	1	-
Boolean for end of coolant thermostat diagnosis					
LV_END_DIAG_RBM_TH	O/V	0...1H	0...1	1	-
Boolean for end of coolant thermostat diagnosis for RBM					
LV_END_DIAG_RBM_TH_TMP	-	0...1H	0...1	1	-
Temporary bit for end of coolant thermostat diagnosis for RBM					
LV_CNL_DIAG_TH	V	0...1H	0...1	1	-
Boolean for coolant thermostat diagnosis cancelled					
LV_ACT_DIAG_TH	O/V	0...1H	0...1	1	-
Boolean for coolant thermostat diagnosis activated					
LV_ACT_DIAG_RBM_TH	V	0...1H	0...1	1	-
Boolean for coolant thermostat diagnosis activated for RBM					
TIA_DE_DIAG_TH	V	0...FEH	-48...142.5	0.75	°C
Intake air temperature deviation value for coolant thermostat diagnosis					
TIA_MIN_DIAG_TH	V	0...FEH	-48...142.5	0.75	°C
Minimum Intake air temperature value during coolant thermostat diagnosis					
T_TCO_DE_DIAG_TH	V	0...FFFFH	0...65535	1	s
Timer for coolant temperature deviation for coolant thermostat diagnosis					
TCO_DE_INI_DIAG_TH	V	0...FEH	-48...142.5	0.75	°C
Initialization value for coolant temperature deviation for coolant thermostat diagnosis					
TCO_DE_DIAG_TH	V	0...FEH	-48...142.5	0.75	°C
Coolant temperature deviation value for coolant thermostat diagnosis					
LV_TCO_DE_MAX_DIAG_TH	V	0...1H	0...1	1	-
Boolean for high coolant temperature deviation for coolant thermostat diagnosis					
TCO_DIAG_TH	V	0...FEH	-48...142.5	0.75	°C
Coolant temperature of the performed coolant thermostat diagnosis					

Input data:

TIA_THR	TIA_THR_ST	TCO_MES	TCO_ST
RATIO_T_PUC_AST	RATIO_T_LOAD_MIN_AST	RATIO_T_VS_MIN_AST	RATIO_T_VS_MAX_AST
TCO_SUB	N	LV_INH_DIAG_TH	LV_CDN_VB_OBD_2
LV_ST_END			

FUNCTION DESCRIPTION:

General information:

Task of the coolant thermostat is to effect a quick engine warm up after start. The thermostat is closed after engine start to limit the cooling-liquid circulation until the thermostat regulating temperature is reached. The thermostat opens at this temperature and the limitation of the cooling-liquid circulation is finished. If an opened stuck thermostat occurs, the cooling-liquid circulation will not be limited after start. That means an increase of the engine warm up time and can cause emission increase as well.

To take this behaviour into consideration, an opened stuck thermostat must be detected if either other diagnosis functions are inhibited or the coolant temperature does not reach its operating temperature within 20°F. The conditions for inhibiting the diagnosis are described in the chapter “Application incidences for coolant thermostat monitoring”. The coolant temperature model (TCO_SUB) must simulate the worst engine warm up temperature behaviour for a normal working thermostat (heating on. A/C on. etc.) after start until the thermostat opening temperature is reached.

Description:

The error detection is done by a comparison between the raw value of the coolant temperature sensor and the value of the coolant temperature substitute model.

Error-symptoms are defined to this diagnosis function as following:

```
□□□□
│ │ │ └─> -                 (= SYM_0)
│ │ └───> -                 (= SYM_1)
│ └─────> -                 (= SYM_2)
└───────> TH open stuck     (= SYM_3)
```

B.48.1 Activation condition for thermostat monitoring

FUNCTION DESCRIPTION:

General information:

The initialization of the function is performed as soon as EXIT_ST is detected. In this case it is regardless if the actual thermostat diagnosis status is set to enabled or canceled.

Thermostat diagnosis is performed as soon as all conditions below are fulfilled at the same time. The function is enabled at once if LV_ST_END is set to “1”. A minimum and a maximum threshold for the coolant temperature at start are limiting the diagnosis activation range as well as a minimum threshold for the intake air temperature at start.

Thermostat monitoring is finished if either the logical value for “thermostat diagnosis finished” or the logical value for “thermostat diagnosis cancelled” equals to “1”. In this case the diagnosis activation bit is set to “0” (LV_ACT_DIAG_TH = 0).

Application conditions:

Initialisation at EXIT_ST:

LV_ERR_TH = 0

LV_CDN_DIAG_TH = 0

ERR_SYM_TH = “NO_SYM”

LV_END_DIAG_RBM_TH = 0

LV_END_DIAG_RBM_TH_TMP = 0

LV_ACT_DIAG_RBM_TH = 0

LV_CNL_DIAG_TH = 0

	LV_ACT_DIAG_TH = 0	
	TCO_DIAG_TH = -48°C	
Recurrence:	1000 ms	
Activation:	LV_ST_END = 1	
Deactivation:	LV_ST_END = 0	

Formula section:

Calculation of the functionality check (RBM) enable condition:

If	TCO_ST > C_TCO_ST_MIN_DIAG_TH	**and**
	TCO_ST < C_TCO_ST_MAX_DIAG_TH	**and**
	TIA_THR_ST > C_TIA_ST_MIN_DIAG_TH	**and**
	LV_CDN_VB_OBD_2 = 1	**and**
	LV_INH_DIAG_TH = 0	**and**
	LV_CNL_DIAG_TH = 0	
then	LV_ACT_DIAG_RBM_TH = 1	
else	LV_ACT_DIAG_RBM_TH = 0	
endif		

Calculation of the functionality check enable condition:

If	LV_ACT_DIAG_RBM_TH = 1	**and**
	LV_END_DIAG_TH = 0	
then	LV_ACT_DIAG_TH = 1	
else	LV_ACT_DIAG_TH = 0	
endif		

Calculation of the intake air temperature deviation:

If (1)	TCO_SUB <= C_TCO_TH_OPEN_DIAG_TH	**and**
	LV_ACT_DIAG_RBM_TH = 1	
then (1)		
if (2)	TIA_THR < TIA_MIN_DIAG_TH $_{(n-1)}$	
then (2)	TIA_MIN_DIAG_TH $_{(n)}$ = TIA_THR	
else (2)	TIA_MIN_DIAG_TH $_{(n)}$ = TIA_MIN_DIAG_TH $_{(n-1)}$	
endif (2)		
else (1)	TIA_DE_DIAG_TH = TIA_MIN_DIAG_TH - TIA_THR_ST	
endif (1)		

Calculation of the coolant temperature deviation during high engine speed:

If (1) TCO_SUB <= C_TCO_TH_OPEN_DIAG_TH **and**
LV_ACT_DIAG_RBM_TH = 1

then (1)

if (2) N >= C_N_MAX_DIAG_TH

then (2) (Trigger the coolant temperature deviation calculation)

1st. step: "Initialization" : No re-triggering until timer ends

TCO_DE_INI_DIAG_TH = TCO_MES

TCO_DE_DIAG_TH = TCO_MES - TCO_DE_INI_DIAG_TH

T_TCO_DE_DIAG_TH = C_T_TCO_DE_DIAG_TH

Timer T_TCO_DE_DIAG_TH starts

2nd. step: "Check for coolant temperature deviations until timer ends"

TCO_DE_DIAG_TH = TCO_MES - TCO_DE_INI_DIAG_TH

if (3) TCO_DE_DIAG_TH < C_TCO_DE_MAX_DIAG_TH

then (3) LV_TCO_DE_MAX_DIAG_TH = 1

endif (3)

endif (2)

endif (1)

Calculation of the diagnosis result:

If (1) LV_ACT_DIAG_TH = 1

then (1) *// Coolant thermostat functionality check enabled*

// Calculation of chapter 1.2

endif (1)

Calculation of the diagnosis condition:

If (1) LV_END_DIAG_TH = 1 **or**

LV_INH_DIAG_TH = 1 **or**

LV_CNL_DIAG_TH = 1

then (1) LV_CDN_DIAG_TH = 0

If(2) Transition LV_END_DIAG_TH = 0 to LV_END_DIAG_TH = 1

then(2) TCO_DIAG_TH = TCO_MES

endif (2)

endif (1)

Calculation of end of diagnosis for RBM:

```
If (1)          LV_ACT_DIAG_RBM_TH = 1                              and
                TCO_SUB > C_TCO_TH_OPEN_DIAG_TH                     and
                LV_END_DIAG_RBM_TH_TMP = 0                          and
                LV_CNL_DIAG_TH = 0
then (1)
                LV_END_DIAG_RBM_TH_TMP = 1
    if (2)      TCO_MES >= C_TCO_WUP_MIN_DIAG_TH
    then (2)
        if (3a)     RATIO_T_VS_MIN_AST <=
                        C_RATIO_T_VS_MIN_THD_DIAG_TH                and
                    RATIO_T_LOAD_MIN_AST <=
                        C_RATIO_T_LOAD_MAX_THD_DIAG_TH
        then (3a)   LV_END_DIAG_RBM_TH = 1
        endif (3a)
    else (2)
        if (3b)     RATIO_T_PUC_AST <=
                        C_RATIO_T_PUC_THD_DIAG_TH                   and
                    RATIO_T_LOAD_MIN_AST <=
                        C_RATIO_T_LOAD_MIN_THD_DIAG_TH              and
                    RATIO_T_VS_MAX_AST <=
                        C_RATIO_T_VS_MAX_THD_DIAG_TH                and
                    TIA_DE_DIAG_TH >= C_TIA_DE_MAX_DIAG_TH          and
                    LV_TCO_DE_MAX_TH_DIAG = 0
        then (3b)   LV_END_DIAG_RBM_TH = 1
        endif (3b)
    endif (2)
endif (1)
```

B.48.2 Coolant thermostat functionality check

FUNCTION DESCRIPTION:

General information:

Task of the functionality check is to detect a stuck open thermostat. Correct functionality is not guaranteed if the coolant temperature does not reach a specified warmed up temperature.

The thermostat functionality check is enabled as long as the boolean for "coolant thermostat diagnosis activated" is set to "1". As soon as EXIT_ST is detected, all timers and initialization values are set to neutral values.

Application conditions:

Initialisation at EXIT_ST:

TIA_DE_DIAG_TH = 0°C

T_TCO_DE_DIAG_TH = 0s

TCO_DE_INI_DIAG_TH = 0°C

TCO_DE_DIAG_TH = 0°C

LV_TCO_DE_MAX_DIAG_TH = 0

TIA_MIN_DIAG_TH = TIA_THR_ST

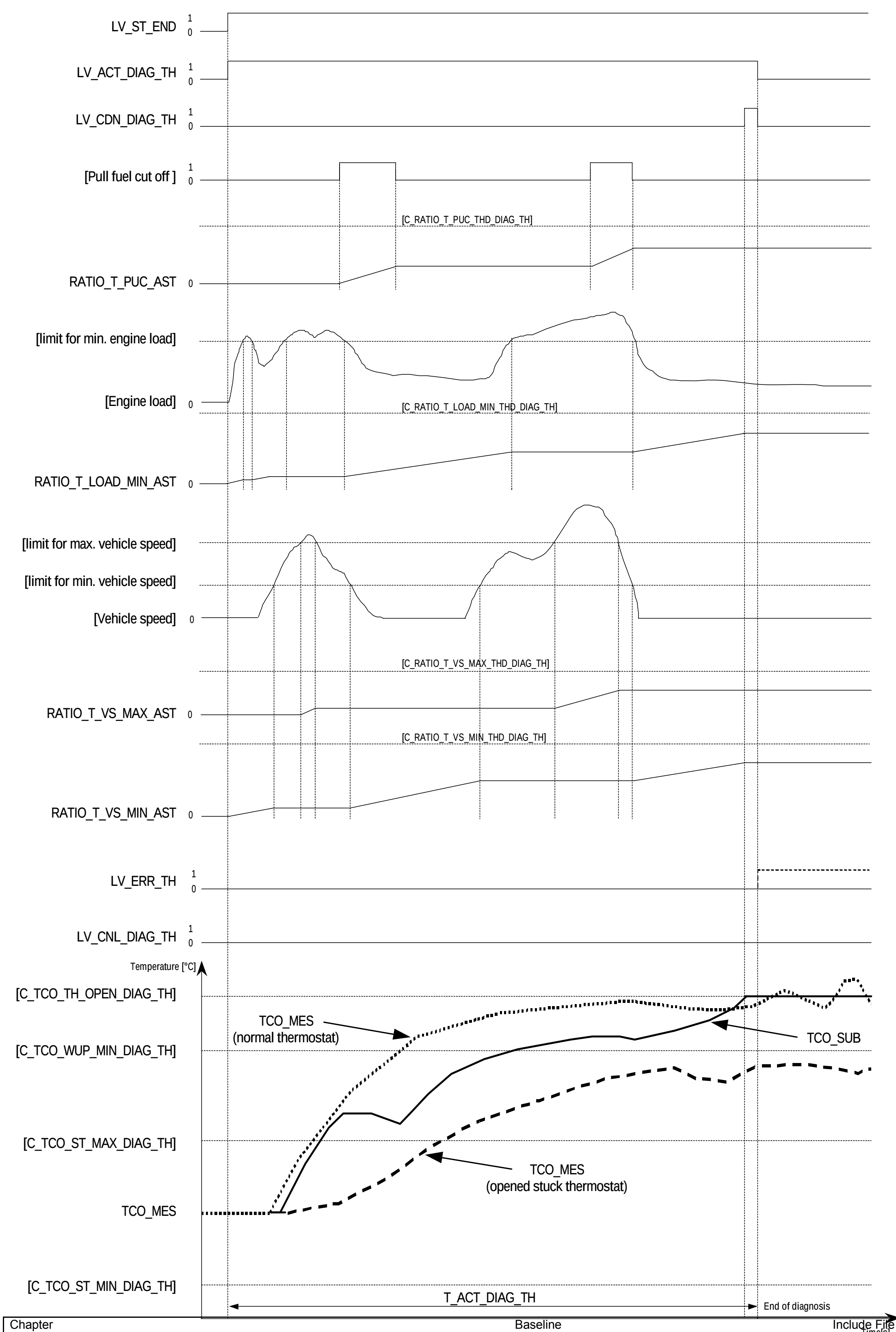
Recurrence: 1000 ms

Activation: LV_ACT_DIAG_TH = 1

(Coolant thermostat functionality check enabled)

Deactivation: LV_ACT_DIAG_TH = 0

(Coolant thermostat functionality check disabled)



Signal flow diagram:

Formula section:

Calculation of the diagnosis result:

```
If (1)          TCO_MES > C_TCO_TH_OPEN_DIAG_TH
then (1)
    if (2)      RATIO_T_VS_MIN_AST > C_RATIO_T_VS_MIN_THD_DIAG_TH
           or   RATIO_T_LOAD_MIN_AST > C_RATIO_T_LOAD_MAX_THD_DIAG_TH
    then (2)    LV_CDN_DIAG_TH = 0
                LV_CNL_DIAG_TH = 1
    else (2)    LV_CDN_DIAG_TH = 1
                ERR_SYM_TH = “NO_SYM”
                LV_CNL_DIAG_TH = 0
    endif (2)
else (1)
    if (2)      TCO_SUB > C_TCO_TH_OPEN_DIAG_TH
    then (2)
        if (3)  TCO_MES >= C_TCO_WUP_MIN_DIAG_TH
        then (3)
          if    RATIO_T_VS_MIN_AST > C_RATIO_T_VS_MIN_THD_DIAG_TH
             or RATIO_T_LOAD_MIN_AST > C_RATIO_T_LOAD_MAX_THD_DIAG_TH
          then  LV_CDN_DIAG_TH = 0
                LV_CNL_DIAG_TH = 1
          else  LV_CDN_DIAG_TH = 1
                ERR_SYM_TH = “NO_SYM”
                LV_CNL_DIAG_TH = 0
          endif
        else (3) (In case of "TCO_MES less than C_TCO_WUP_MIN_DIAG_TH ")
          if    ( RATIO_T_PUC_AST > C_RATIO_T_PUC_THD_DIAG_TH
             or RATIO_T_LOAD_MIN_AST > C_RATIO_T_LOAD_MIN_THD_DIAG_TH
             or RATIO_T_VS_MAX_AST > C_RATIO_T_VS_MAX_THD_DIAG_TH
             or TIA_DE_DIAG_TH < C_TIA_DE_MAX_DIAG_TH
             or LV_TCO_DE_MAX_TH_DIAG = 1 )
          then  LV_CDN_DIAG_TH = 0
                LV_CNL_DIAG_TH = 1
          else  LV_CDN_DIAG_TH = 1
                ERR_SYM_TH = “SYM_3”
                LV_ERR_TH = 1                         (after debounce)
```

LV_CNL_DIAG_TH = 0

endif

endif (3)

endif (2)

endif (1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_ST_MIN_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant temperature at start for coolant thermostat diagnosis					
C_TCO_ST_MAX_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Maximum coolant temperature at start for coolant thermostat diagnosis					
C_TIA_ST_MIN_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Minimum intake air temperature at start for coolant thermostat diagnosis					
C_RATIO_T_PUC_THD_DIAG_TH	1	0...FFH	0...99.609	0.3906	%
"Pull fuel cut off" ratio threshold in per cent for coolant thermostat diagnosis					
C_RATIO_T_VS_MIN_THD_DIAG_TH	1	0...FFH	0...99.609	0.3906	%
"Vehicle speed minimum" ratio threshold in per cent for coolant thermostat diagnosis					
C_RATIO_T_VS_MAX_THD_DIAG_TH	1	0...FFH	0...99.609	0.3906	%
"Vehicle speed maximum" ratio threshold in per cent for coolant thermostat diagnosis					
C_RATIO_T_LOAD_MIN_THD_DIAG_TH	1	0...FFH	0...99.609	0.3906	%
"Engine load minimum" minimum ratio threshold in per cent for coolant thermostat diagnosis					
C_RATIO_T_LOAD_MAX_THD_DIAG_TH	1	0...FFH	0...99.609	0.3906	%
"Engine load minimum" maximum ratio threshold in per cent for coolant thermostat diagnosis					
C_N_MAX_DIAG_TH	1	0...1FE0H	0...8160	1	rpm
Maximum engine speed for high coolant temperature deviation detection for coolant thermostat diagnosis					
C_T_TCO_DE_DIAG_TH	1	0...FFFFH	0...65535	1	s
Time for high coolant temperature deviation detection for coolant thermostat diagnosis					
C_TCO_DE_MAX_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Maximum coolant temperature deviation for coolant thermostat diagnosis					
C_TIA_DE_MAX_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Maximum intake air temperature deviation for coolant thermostat diagnosis					
C_TCO_TH_OPEN_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Regulation temperature for coolant thermostat opening					
C_TCO_WUP_MIN_DIAG_TH	1	0...FEH	-48...142.5	0.75	°C
Minimum coolant warm up temperature for coolant thermostat diagnosis					

B.49 Coolant thermostat monitoring (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_TH	O/V	0...1H	0...1	1	-
Boolean for coolant thermostat diagnosis inhibit					
LV_INH_DIAG_RBM_TH	V	0...1H	0...1	1	-
Boolean for coolant temperature plausibility diagnosis inhibit due to OBD error					
STATE_DIAG_TH	O/V	0H 1H 2H 3H	ST END CNL INH	-	-
Status for coolant thermostat diagnosis					

Input data:

LV_IGK	LV_ST_END	LV_ERR_TCO_EL	LV_ERR_TCO_GRD
LV_ERR_TCO_STUCK	LV_ERR_MAF	LV_ERR_TIA_THR	LV_ERR_TPS
LV_ERR_VS	LV_ERR_CRK	LV_ERR_MWSS	LV_ERR_TCO_PLAUS
LV_ES	TAM	LV_ERR_LOAD_TPS_PLAUS	LV_END_DIAG_TH
LV_INH_DIAG_TH	LV_CNL_DIAG_TH	LV_ERR_TCO_STUCK_H	LV_ERR_TCO_STUCK_RNG
LV_ERR_MAP			

FUNCTION DESCRIPTION:

General information:

The coolant thermostat diagnosis is performed at transition from the logical value for “engine stop” from “1” to “0”. If the engine stalls or one of the conditions for inhibiting the diagnosis is present, it will be stopped and starts again at the next engine start.

The diagnosis is finished as soon as the boolean for end of coolant thermostat diagnosis is set to “1”. In this case the boolean for coolant thermostat error indicates either an open stuck thermostat (= 1) or a normal working thermostat (= 0).

Depending on project specific requirements, the coolant thermostat diagnosis can be inhibited by setting of LV_INH_DIAG_TH.

Application conditions:

Initialisation at RESET or EXIT_ST or LV_IGK = 0 -> 1 ***or*** *FMY clear:*

LV_INH_DIAG_TH = 0

LV_INH_DIAG_RBM_TH = 0

STATE_DIAG_TH="ST"

Recurrence: 1000 ms

Activation: LV_ST_END = 1

(Coolant thermostat monitoring (Appl. Inc.)” function enabled)

Deactivation: LV_ST_END = 0

(Coolant thermostat monitoring (Appl. Inc.)” function disabled)

Formula section:

Calculation of diagnosis interface parameter:

If	LV_ERR_TCO_EL = 1	**or**
	LV_ERR_TCO_GRD = 1	**or**
	LV_ERR_TCO_STUCK = 1	**or**
	LV_ERR_TCO_STUCK_H = 1	**or**
	LV_ERR_TCO_STUCK_RNG = 1	**or**
	LV_ERR_MAF = 1	**or**
	LV_ERR_MAP = 1	**or**
	LV_ERR_TIA_THR = 1	**or**
	LV_ERR_TPS = 1	**or**
	LV_ERR_VS = 1	**or**
	LV_ERR_CRK = 1	**or**
	LV_ERR_MWSS = 1	**or**
	LV_ERR_TCO_PLAUS = 1	**or**
	LV_ERR_LOAD_TPS_PLAUS = 1	
then	LV_INH_DIAG_RBM_TH = 1	
endif		

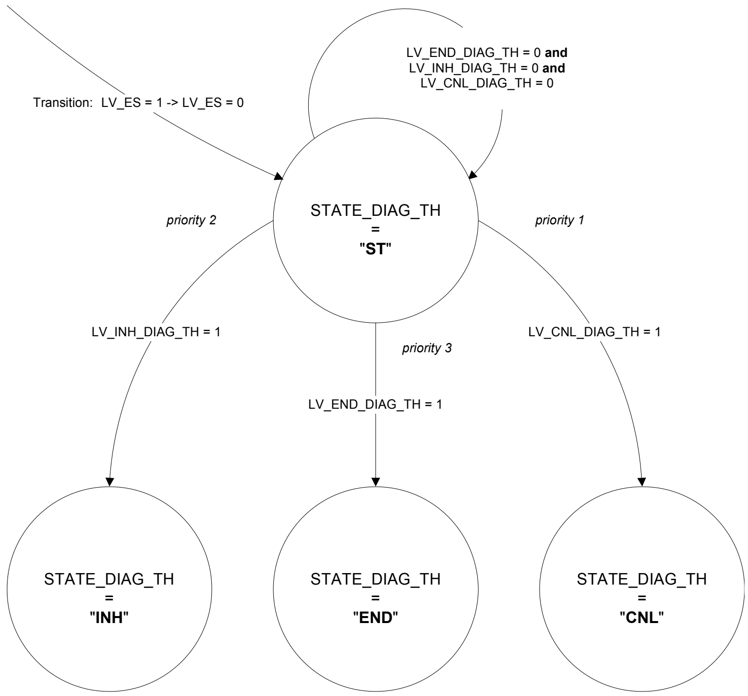
If	LV_INH_DIAG_RBM_TH = 1	**or**
	TAM < C_TAM_MIN_INH_TH	
then	LV_INH_DIAG_TH = 1	
endif		

Error treatment:

Error debounce:

Debounce counter increment: C_ABC_INC_TH
Debounce counter maximum value: C_ABC_MAX_TH

Signal flow diagram:



Configuration for diagnostic symptoms:

Diagnosis	Symptom description	Symptom	Filter type
Coolant thermostat monitoring		SYM_0	STD
		SYM_1	
		SYM_2	
TH	Thermostat stuck open	SYM_3	

List of Environmental Data to store in Failure Memory: T_AST_ENVD
TAM
TCO_ST
TIA_ST

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TH	1	0...FFH	0...255	1	-
Debounce counter increment value for the coolant thermostat diagnosis					
C_ABC_MAX_TH	1	01...FFH	1...255	1	-
Debounce counter maximum value for the coolant thermostat diagnosis					
C_TAM_MIN_INH_TH	1	0...FEH	-48...142.5	0.75	°C
Maximum Ambient Temperature to perform EVAP Monitoring					

B.50 Rate Base Monitoring interface for thermostat monitoring.

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_TH	O/V	0 ... 7H	0...7	1	-
Interface of thermostat monitoring with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_TH	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for Thermostat Monitoring , for application purpose					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_RBM_TH	LV_INH_DIAG_RBM_TH
CTR_CDN_RBM_TH	CTR_COMP_RBM_TH		

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the thermostat monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_TH data.

Within STATE_RBM_TH, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (not valid for catalyst diagnosis)

Application conditions:

Initialisation :

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_TH = 0

at LV_DC 0 → 1 transition :

bit 0 and bit 1of STATE_RBM_TH = 0

bit 2 of STATE_RBM_TH = 1

on failure memory reset :

bit 1 of STATE_RBM_TH = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_TCO_EL	LV_ERR_TCO_GRD	LV_ERR_TCO_STUCK	LV_ERR_TCO_STUCK_H
LV_ERR_TCO_STUCK_RNG			

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_TH = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_TH = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If bit 0 of STATE_RBM_TH = 0

Then

If LV_END_DIAG_RBM_TH = 1

Then bit 0 of STATE_RBM_TH = 1

Endif

Endif

If bit 1 of STATE_RBM_TH = 0

Then

If LV_INH_DIAG_RBM_TH = 1

Then bit 1 of STATE_RBM_TH = 1

Endif

Endif

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_TH = 0
then RATE_RBM_TH = FFFFH
else RATE_RBM_TH
= (CTR_COMP_RBM_TH / CTR_CDN_RBM_TH)
endif

B.51 Dynamic Error Management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_DC[NC_NR_ERR_DYN]	O/V/S	0...FFH	0...255	1	[-]
Counter for driving cycle					
CTR_ERR_DYN_NR	O/V/S	0...FFH	0...255	1	[-]
Number of failures stored in dynamic memory (2nd layer)					
CTR_FRC[NC_NR_ERR_DYN]	O/V/S	0...FFH	0...255	1	[-]
Frequency counter : number of occurrences of present failure					
CTR_WUP_CYC[NC_NR_ERR_DYN]	O/V/S	0...FFH	0...255	1	[-]
Counter for warm-up cycle					
DIAG_INST[NC_NR_ERR_DYN]	V	0...FFFFH	0...65535	1	[-]
Index of diagnosis instance XX with XX = F(IDX)					
LV_DC_MAX[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Flag for the MIL state (on or off) related to failure IDX					
LV_ERR_CFM[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Confirmed failure status flag					
LV_ERR_DC[NC_NR_ERR_DYN]	O/V	0...1H	0...1	1	[-]
Failure present status for this driving cycle					
LV_ERR_DISA[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Disappeared failure status flag					
LV_ERR_LST_CLR_XX	O/V/S	0...1H	0...1	1	[-]
Failure set at least once since last failure memory clear					
LV_ERR_MEM_XX	O/V/S	0...1H	0...1	1	[-]
Failure is stored in dynamic memory (2nd layer)					
LV_ERR_MKD[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Marked failure status flag					
LV_ERR_PND[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Pending failure status flag					
LV_ERR_TMP[NC_NR_ERR_DYN]	O/V/S	0...1H	0...1	1	[-]
Temporary failure status flag					

Input data:

DC_DEC_XX	DC_INC_XX	DC_MAX_XX	LC_ENA_SCDN
LV_DC	LV_END_DIAG_XX	LV_ERR_XX	LV_MKD_MOD
LV_WUP_CYC	NLC_ENA_SCDN		

Export actions:

ACTION_ERRM_*EraseErr* (IN <IDX>)
This action erases the failure IDX in the second layer memory
ACTION_ERRM_*IncrementDCctrScdn* (IN <XX>)
This action is used to increment the driving cycle counter for failures using similar conditions functionality
ACTION_ERRM_*CheckPendingStatus* (IN <PRM_IDX_DIAG>, OUT <PRM_LV_ERR_PND>)
This action shall be called to determine if a failure is pending or not
ACTION_ERRM_*GetErrLastClr* (<PRM_IDX_ERR>, <PRM_LV_ERR_LST_CLR>)
This action indicates if a failure has been set already once since the last failure memory clear.

Import actions:

ACTION_ERRM_*StoreDtc* (IN <IDX>)
ACTION_ERRM_*StoreDtcLst* (IN <IDX>)
ACTION_ERRM_*EraseDtc* (IN <IDX>)
ACTION_ERRM_*StoreFrf* (IN <IDX>)
ACTION_ERRM_*EraseFrf* (IN <IDX>)

ACTION_ERRM_*StoreHistory* (IN <IDX>)
ACTION_ERRM_*DecrementDCctrScdn* (OUT <DCdec>, IN <XX>)
ACTION_ERRM_*ConfirmErrScdn* (OUT <DCconf>, IN <XX>)
ACTION_ERRM_*EraseScdn* (IN <XX>)
ACTION_ERRM_*PrioRule* (OUT <RESP> , OUT <IDX>)
ACTION_ERRM_*TrigErrDyn* (IN <XX>)
ACTION_ERRM_*StorePermanentCode* (IN <PRM_IDX_ERR>)
ACTION_ERRM_*ErasePermanentCode* (IN <PRM_IDX_ERR>)
ACTION_ERRM_*ClcPermanentIniErrm* ( )
ACTION_ERRM_*TrigFarm* (IN <XX>)
ACTION_ERRM_*GetMilRelevant* (IN <PRM_IDX_ERR>, OUT <PRM_LV_MIL_RLV>)

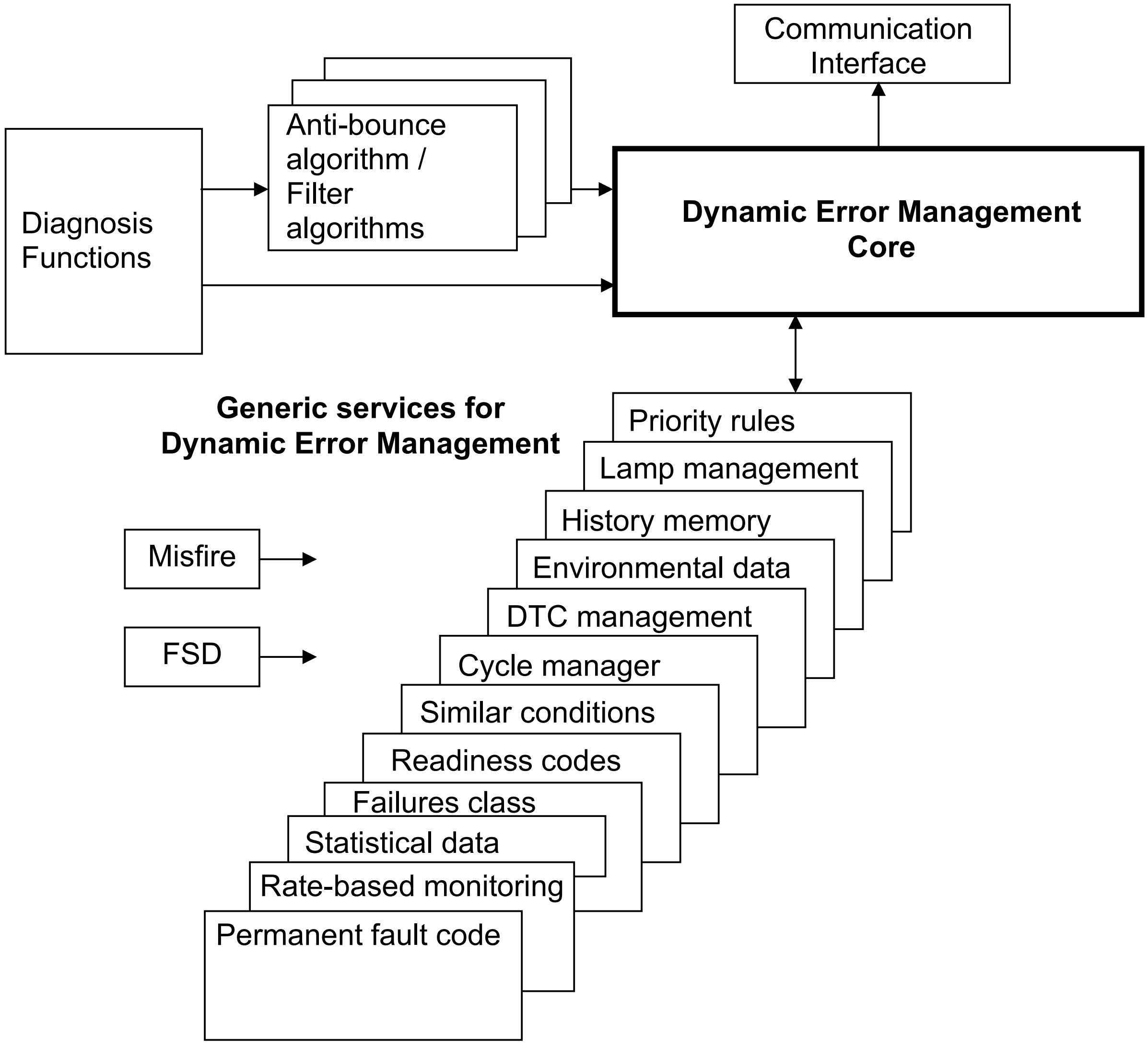
FUNCTION DESCRIPTION:

General information:

This module is the heart of the dynamic error management function. It defines statement of principles and mechanism of dynamic error management function.

Dynamic error management receives results from diagnosis function (may be filtered by antibounce algorithm). These results are, in this module, managed as diagnosis failure following **CARB** and **EOBD** standards.

In addition, this module uses generic services defined by others modules (priority rules for example).


Communication Interface
Anti-bounce algorithm / Filter algorithms
Diagnosis Functions
Dynamic Error Management Core
Generic services for Dynamic Error Management
Priority rules
Lamp management
History memory
Environmental data
DTC management
Cycle manager
Similar conditions
Readiness codes
Failures class
Statistical data
Rate-based monitoring
Permanent fault code
Misfire
FSD

B.51.1 Diagnosis failure description

B.51.1.1 Failures definition

Description:

Dynamic error management manages failures, which have different states.

Detected failure

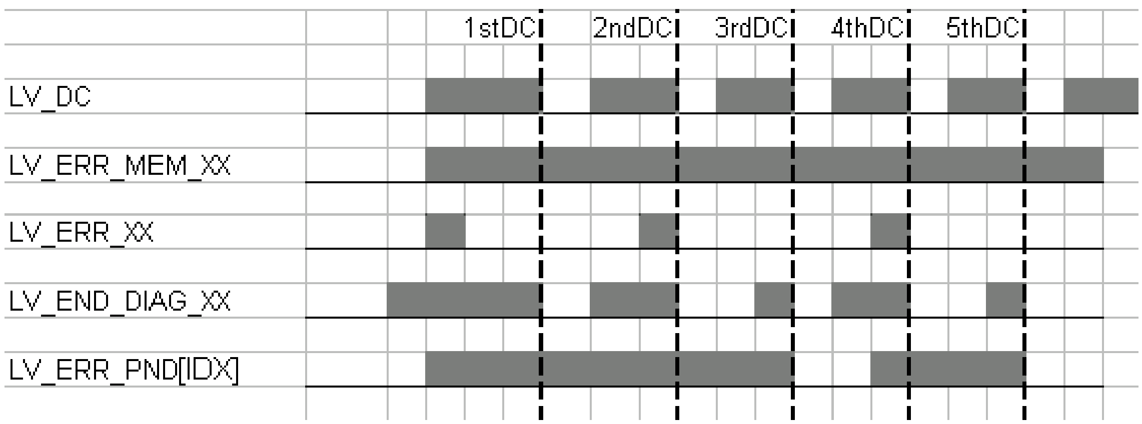
The detected failure is provided by a defective system (the error provided by this system is basic). It occurs when the diagnosis test and diagnosis condition of at least one symptom or more are true and disappears in other cases.

Present failure

The present failure is the detected failure after filtering. The real entry point of the error management is the present failure.

Pending failure

The pending failure is the detected failure after filtering. The failure shall stay pending until detection of a whole driving cycle without the failure present and the diagnosis done.



Temporary failure

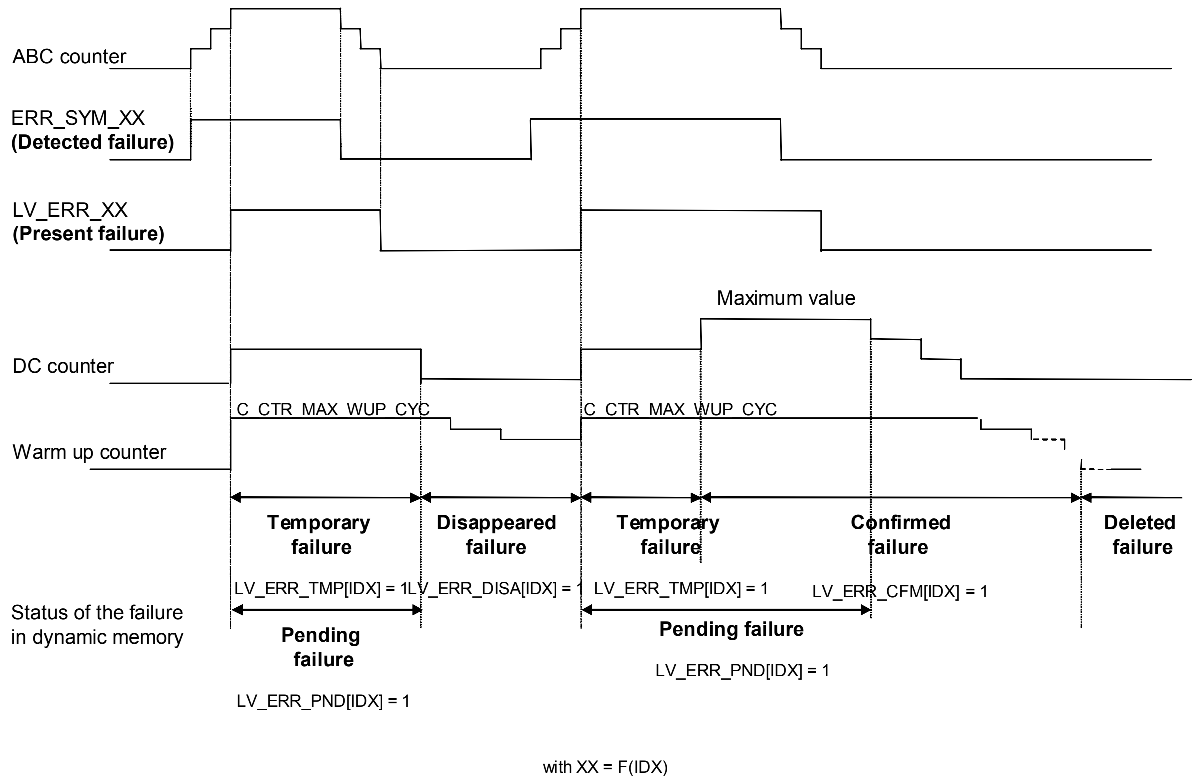
A failure is considered as temporary when this failure is present and leaves this state either when this failure is confirmed or disappeared (in this case, this failure isn't confirmed).

Disappeared failure

A failure is considered as disappeared, when no failure was detected during a complete driving cycle and the failure wasn't confirmed. It leaves this state either when the failure reoccurs or when the warm-up counter reaches 0.

Confirmed failure

A failure is considered as confirmed when the driving cycle counter reaches the maximum value and leaves this state when the warm-up counter reaches 0.



B.51.1.2 Failures information

Description:

Each failure is constituted of informations, which are stored in the memory.

Some informations are allocated in the static memory. It means that these informations are available for each failure.

Static memory: XX represents any failure defined in the application (index of XX∈[1...65535])

- ABC counter	CTR_ABC_XX
- End of diagnosis counter	CTR_ABC_END_DIAG_XX
- End of diagnosis flag	LV_END_DIAG_XX
- Readiness flag	LV_READY_XX
- Diagnosis condition	LV_CDN_DIAG_XX
- Detected symptom	ERR_SYM_XX
- Present failure	LV_ERR_XX
- Failure is stored in dynamic memory	LV_ERR_MEM_XX

- Failure present at least one time since failure memory cleared LV_ERR_LST_CLR_XX

Some informations are allocated in the dynamic memory. It means that these informations are created in memory in case of a new dynamic failure entry (limited number of dynamic entries managed with priority rules).

Dynamic memory : IDX∈ [0...NC_NR_ERR_DYN-1]

- Frequency counter	CTR_FRC[IDX]
- Driving cycle counter	CTR_DC[IDX]
- Warm up counter	CTR_WUP_CYC[IDX]
- Memorized the first symptom (DTC)	ERR_SYM_MEM[IDX]
- Memorized the last symptom (DTC)	ERR_SYM_LST[IDX]
- Environmental data (Freeze frame)	ENVD_OBD[u][m]
	ENVD_CUS_CMN[v][m]
	ENVD_CUS_SET_CMN[W][x][m]
	ENVD_CUS_SET_SPC[y][z][m]
- Confirmed failure	LV_ERR_CFM[IDX]
- Disappeared failure	LV_ERR_DISA[IDX]
- Memorized MIL activation flag	LV_DC_MAX[IDX]
- Failure during DC	LV_ERR_DC[IDX]
- Temporary failure	LV_ERR_TMP[IDX]
- Pending failure	LV_ERR_PND[IDX]
- Marked failure	LV_ERR_MKD[IDX]

Remark :

To keep the link between the index of the failure in dynamic memory and the index of the failure in static memory, a transfer function is defined by : XX = F (IDX)

with m=IDX_FRF[IDX]

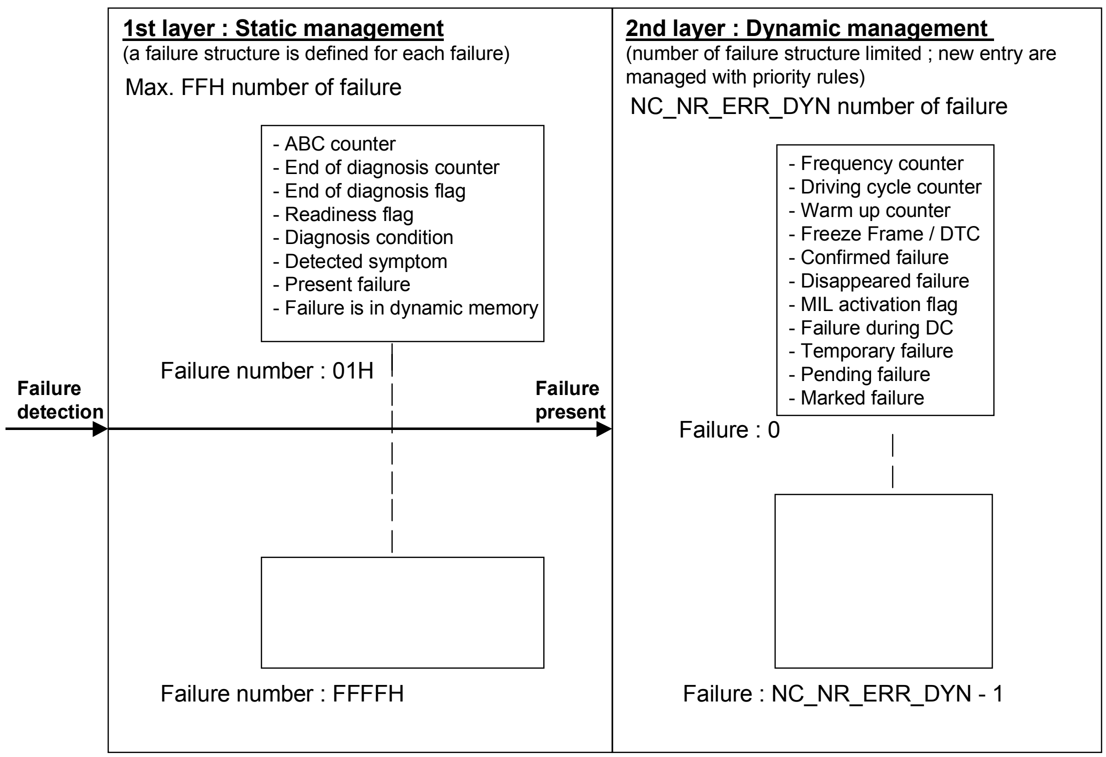
B.51.2 Memory management

Description:

As explained above, the memory is managed as two layers.

The first layer is managed as static memory. According to failure detection, ABC counter and end of diagnosis counter are incremented or decremented. Present failure information is the output of this layer. All failures are managed in the first layer.

When a present failure appears, it may provoke a dynamic entry in the second layer (dynamic) of the memory. In this second layer, only present failures are managed. The number of failure structure is limited to a maximum value defined by NC_NR_ERR_DYN.



B.51.2.1 Failures configuration

Description:

For every diagnosis two bytes exist to configure the failure. The following items can be influenced :

- control of dashboard lamp (MIL and/or Warning lamp 1 and/or Warning lamp 2)
- the emission relevance of a failure
- failure priority, used for failure storage

The definition of these bytes is done in Failure Class module.

B.51.2.2 Store / Erase a failure in 2nd layer memory

Description:

This module manages the "Entry" (storage) and "Exit" (deleting) of failures in the 2nd layer memory.

Storage :

Syntax : ACTION_ERRM_StoreErr (IN <XX>)

Parameter (in) : XX number of the failure to store in dynamic memory

Parameter (out) : -

Short description :

If other modules call the action to store the failure XX with ACTION_ERRM_StoreErr (IN<XX>) then the failure XX (with XX = F(IDX)) is taken into the 2nd layer memory and :

- frequency counter (number of occurrences of present failure) is updated ;
- the DTC is stored with ACTION_ERRM_StoreDtc (IN<IDX>) (if needed) ;
- the Freeze Frame is stored with ACTION_ERRM_StoreFrf (IN<IDX>) ;
- the index of diagnosis instance XX is stored in DIAG_INST[IDX] for visualisation

Deleting :

Syntax : ACTION_ERRM_EraseErr (IN <IDX>)

Parameter (in) : IDX index of failure in 2nd layer memory to erase

Parameter (out) : -

Short description :

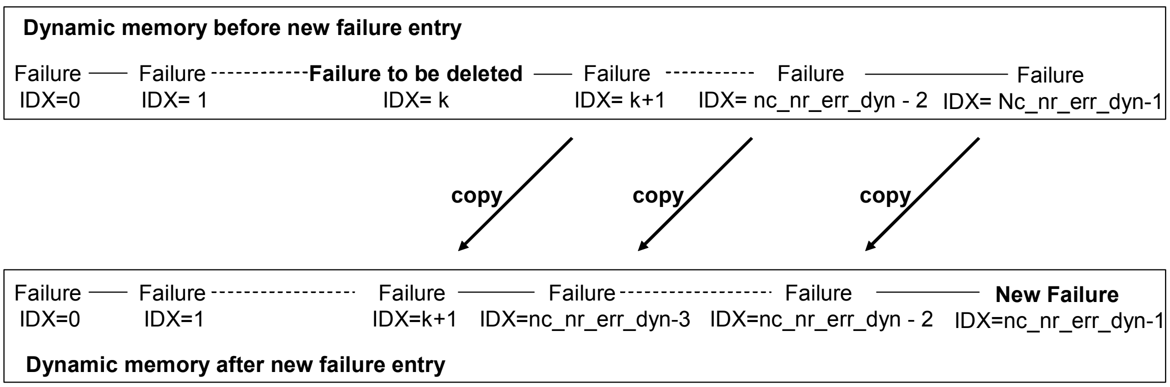
If other modules call the action to erase the failure XX with ACTION_ERRM_EraseErr (IN<IDX>) then the failure IDX is taken out of the 2nd layer memory and :

- the failure is stored in history memory with ACTION_ERRM_StoreHistory (IN<IDX>) ;
- the DTC of the failure is erased with ACTION_ERRM_EraseDtc (IN<IDX>) ;
- the Freeze Frame of the failure is erased with ACTION_ERRM_EraseFrf (IN<IDX>) ;
- similar conditions of the failure are erased with ACTION_ERRM_EraseScdn (IN<XX>) if existing

2nd layer memory description :

It is useful to sort the failures in the sequence in which they occurred ; first failure has index (IDX) number zero, second failure has index (IDX) number one, ... For example, when a new failure entry is accepted (memory is full), one failure is deleted, sorting algorithm is respected and the new failure entry is stored in last position.

In any case, sorting algorithm must be always respected.



Application conditions:

Initialization : On saved RAM lost

CTR_ERR_DYN_NR = 0 ; LV_ERR_MEM_XX = 0

CTR_FRC[IDX] = 0; LV_ERR_TMP[IDX] = 0; LV_ERR_MKD[IDX] = 0

DIAG_INST[IDX] = 0; LV_ERR_LST_CLR_XX = 0

Recurrence: -

Activation: at Action request

Formula section:

Store the failure in 2nd layer memory :

ACTION_ERRM_StoreErr (IN <XX>) :

LV_ERR_LST_CLR_XX = 1

If LV_ERR_MEM_XX = 0

Then Failure XX is stored in 2nd layer memory in position IDX = CTR_ERR_DYN_NR

LV_ERR_MEM_XX = 1

LV_ERR_TMP[IDX] = 1

DIAG_INST[IDX] = index of diagnosis instance XX (with XX=F(IDX))

ACTION_ERRM_StoreDtc (IN<IDX>)

CTR_ERR_DYN_NR = CTR ERR_DYN_NR +1

If LV_MKD_MOD = 1

Then LV_ERR_MKD[IDX] = 1

Endif

Else (IDX compliant with XX = F(IDX))

Endif

CTR_FRC[IDX] = CTR_FRC[IDX] + 1

If CTR_FRC[IDX] > 254 (saturation to 254 : value 255 forbidden)

Then CTR_FRC[IDX] = 254

Endif

ACTION_ERRM_StoreFrf (IN<IDX>)

(previous failure status (if disappeared) necessary for freeze frame calculation)

If LV_ERR_DISA[IDX] = 1

Then LV_ERR_DISA[IDX] = 0

LV_ERR_TMP[IDX] = 1

ACTION_ERRM_StoreDtc (IN<IDX>)

(Store DTC corresponding of the first occurrence of the failure)

Endif

LV_ERR_PND[IDX] = 1

ACTION_ERRM_StoreDtcLst (IN<IDX>)

(Store DTC corresponding of the last occurrence of the failure)

Clear the failure out of 2nd layer memory

ACTION_ERRM_EraseErr (IN<IDX>) :

ACTION_ERRM_StoreHistory (IN<IDX>)

ACTION_ERRM_EraseDtc (IN<IDX>)

ACTION_ERRM_EraseFrf (IN<IDX>)

If failure is using similar conditions functionality (see Similar conditions Appli.Inc.) (LC_ENA_SCDN=1 and NLC_ENA_SCDN=1)

Then ACTION_ERRM_EraseScdn (IN<XX>)

(with XX=F(IDX))

Endif

Failure XX (with XX=F(IDX)) is cleared (all data in 2nd layer memory related to failure IDX are filled with 0 ; all data in static memory related to failure XX are unchanged)

LV_ERR_MEM_XX = 0 (with XX = F(IDX))

CTR_ERR_DYN_NR = CTR_ERR_DYN_NR –1

Temporary flag management :

For each failure in 2nd layer memory

If LV_ERR_CFM[IDX] = 1
or LV_ERR_DISA[IDX] = 1
Then LV_ERR_TMP[IDX] = 0
Endif

B.51.2.3 Clear failure memory

Description:

For development conveniences, it is often useful to reinit all failure in the memory. This is possible by setting the calibration bit LC_ERR_FMY_CLR to 1.

Application conditions:

Initialisation: -

Recurrence: -

Activation: LC_ERR_FMY_CLR 0 → 1 transition

Formula section:

Treatment performed at global clearing request (e.g. Mode$04) :

If LC_ERR_FMY_CLR = 1

Then Reset failure memory (all data in 2nd layer memory and all data in static memory, e.g. LV_ERR_LST_CLR)

CTR_ERR_DYN_NR = 0

ACTION_ERRM_*ClcPermanentIniErrm*()

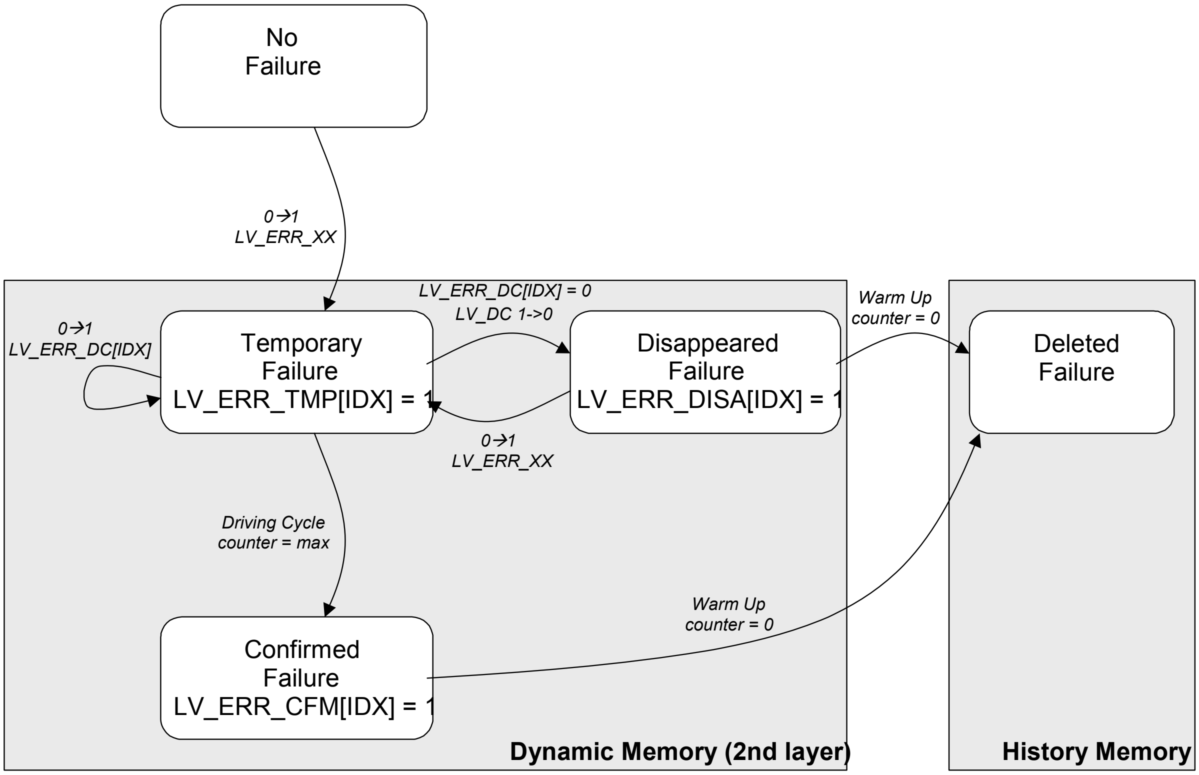
Endif

B.51.3 Failure management

B.51.3.1 State diagram

Description:

In the following diagram, the failure state progress is described.



There are also additional transitions possible due to :

- Overwriting/Erasement by priority rule or Tool Request :

  Temporary Failure → Deleted Failure

  Disappeared Failure → Deleted Failure

  Confirmed Failure → Deleted Failure

	Driving Cycle counter value	Warm Up counter value
No Failure	0	0
Temporary Failure	[0, Max [without having reached Max	Max
Confirmed Failure	[0, Max] with having reached Max	] 0, Max]
Disappeared Failure	0	] 0, Max]
Deleted Failure	0	0

B.51.3.2 Present failure occurred

Description:

When a failure occurs (becomes present), this failure shall be stored in the 2nd layer memory (if possible). If the 2nd layer memory is full and marked mode is not set, a choice is made in the module “Priority rules” :

- RESP is set to OK by priority rules, if the failure can be stored in the 2nd layer memory ; in this case IDX gives the index of the failure to be replaced into dynamic memory;
- RESP is set to NOK by priority rules if the failure can’t be stored in the 2nd layer memory.

To monitor the failure, ACTION_ERRM_TrigErrDyn shall be called when the failure occurs (the failure is evaluated or re-evaluated) or the failure is removed.

Application conditions:

Initialisation: -

Recurrence: -

Activation: at LV_END_DIAG_XX 0 → 1 transition

with (LV_ERR_XX = 1 **and** LV_ERR_MEM_XX = 1)

or

at LV_ERR_XX 1 → 0 transition

(due to the physical default disappears only: it means that a clear failure memory or diagnosis initialization aren’t activation condition)

Formula section:

ACTION_ERRM_TrigErrDyn (IN<XX>)
This action is called to indicate that a new failure has occurred, although failure is not stored.

Application conditions:

Initialisation: -

Recurrence: -

Activation: LV_ERR_XX 0 → 1 transition **with** LV_END_DIAG_XX = 1

or

LV_END_DIAG_XX 0 → 1 transition

with (LV_ERR_XX = 1 **and** LV_ERR_MEM_XX = 0)

Formula section:

If (1) CTR_ERR_DYN_NR < NC_NR_ERR_DYN

or LV_ERR_MEM_XX = 1

Then (1) ACTION_ERRM_StoreErr (IN<XX>)

Else (1) **If (2)** LV_MKD_MOD = 0

Then (2) ACTION_ERRM_PrioRule (OUT<RESP>, OUT<IDX>)

If (3) RESP = OK

Then (3) ACTION_ERRM_EraseErr (IN<IDX>)

ACTION_ERRM_StoreErr (IN<XX>)

Endif (3)

Endif (2)

Endif (1)

ACTION_ERRM_TrigErrDyn (IN<XX>)
This action is called to indicate that a new failure has occurred.

B.51.3.3 Present failure occurred

Application conditions:

Initialisation: -

Recurrence: -

Activation: at LV_END_DIAG_XX 0 → 1 transition (independant of LV_ERR value)

Formula section:

call ACTION_ERRM_TrigFarm(XX)

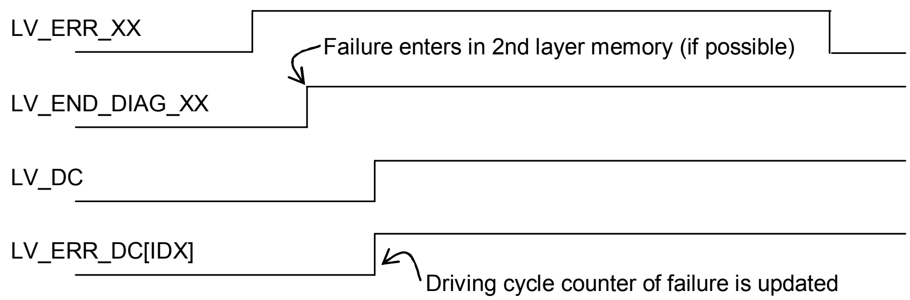
B.51.3.4 Error occurred during present driving cycle

Description:

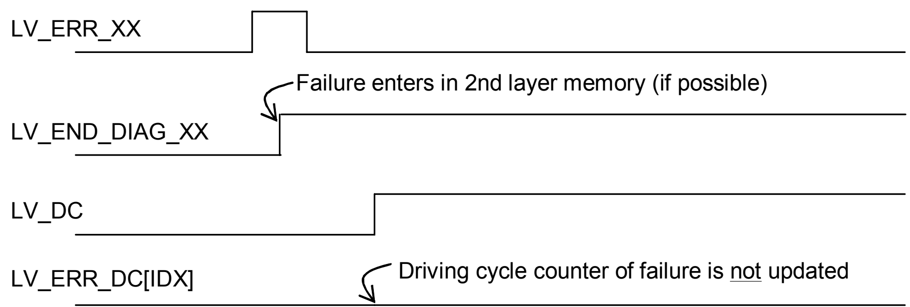
An active flag called LV_ERR_DC[IDX] indicates that an error XX (with XX = F(IDX)) has occurred during a driving cycle.

Even if the error disappears during the driving cycle, the flag remains active during the whole driving cycle.

Failure present, then driving cycle is recognized, then failure not present



Failure present, then failure not present, then driving cycle is recognized



Application conditions:

Initialisation: at ECU reset **or** transition LV_DC 1->0 after treatment done at this event

LV_ERR_DC[IDX] = 0

Recurrence: same as the diagnosis routine

all failures are updated at transition 0->1 of LV_DC

Activation: at every engine operating state

Formula section:

If LV_ERR_XX = 1 (failure is present)

and LV_END_DIAG_XX =1 (current diagnosis is available)

	and LV_DC = 1	(within driving cycle)
	and LV_ERR_MEM_XX =1	(failure is stored in dynamic memory)
Then	LV_ERR_DC[IDX] = 1	(it remains 1 until the next initialization)
Endif		

B.51.3.5 Driving Cycle Counter management

Description:

The driving cycle counter CTR_DC[IDX] is associated to each failure which has a structure defined in second layer of memory.

This counter is incremented at each driving cycle in the following way:

- for all failures not using SCDN functionality (withLC_ENA_SCDN=0 or NLC_ENA_SCDN=0), when the associated failure gets present (LV_ERR_DC[IDX]).
- for failures using SCDN functionality (with LC_ENA_SCDN=1 and NLC_ENA_SCDN=1)
    - when the failure gets present the first time (and stored)
    - when the failure gets present again in a following driving cycle
    - when the failure gets present again in another driving cycle (not previous) (driving cycle counter of this failure is != 0/max before new occurrence), if the similar conditions functionality decides, that the incrementation is possible

The driving cycle counter is decremented at the end of a driving cycle in the following cases:

- for all failures not using SCDN functionality (or LC_ENA_SCDN=0 or NLC_ENA_SCDN=0), when the associated failure was not present the complete DC (LV_ERR_DC[IDX] = 0).
- for failures using SCDN functionality (with LC_ENA_SCDN=1 and NLC_ENA_SCDN=1)
    - when similar conditions are encountered without failure (for failure with present MIL illumination)
    - when the similar conditions functionality decides that the decrementation is possible (for temporary failure or confirmed failure with no present MIL illumination)

The driving cycle counter is useful for MIL management and warm-up counter management (see specific chapter).

With the calibration LC_ENA_SCDN it’s possible to enable/disable the similar conditions functionality. If LC_ENA_SCDN = 0 the failures using SCDN functionality (see similar conditions Appl.Inc) are treated like normal failures in the driving cycle counter management.

Application conditions:

Initialization: On saved RAM lost

LV_DC_MAX[IDX] = 0

CTR_DC[IDX] = 0

LV_ERR_DISA[IDX] = 0

LV_ERR_CFM[IDX] = 0

Recurrence: -

Activation: -

Formula section:

Treatment for all failures not using SCDN functionality (with LC_ENA_SCDN=0 or NLC_ENA_SCDN=0) stored in 2nd layer memory :

Conditions to increment the driving cycle counter :

At LV_ERR_DC[IDX] 0 → 1 transition

XX=f(IDX)

LV_ERR_DISA[IDX] = 0

If LV_DC_MAX[IDX] =1 (for confirmed failure with MIL ON)

Then

CTR_DC[IDX] = DC_MAX_XX (counter is forced to max.)

ACTION_ERRM_*GetMILRelevant*(XX,PRM_LV_MIL_RLV)

If PRM_LV_MIL_RLV= 1 (MIL illuminated and Failure OBD relevant)

Then

Permanent fault code has to be stored, if not yet done due to memory full

ACTION_ERRM_*StorePermanentCode* (XX)

Endif

Else

CTR_DC[IDX] = CTR_DC[IDX] + DC_INC_XX

If CTR_DC[IDX] ≥ DC_MAX_XX

Then

CTR_DC[IDX] = DC_MAX_XX

LV_DC_MAX[IDX] = 1 (failure becomes confirmed)

LV_ERR_CFM[IDX] = 1

ACTION_ERRM_*GetMILRelevant*(XX,PRM_LV_MIL_RLV)

If PRM_LV_MIL_RLV= 1 (MIL illuminated and Failure OBD relevant)

Then

Permanent fault code has to be stored

ACTION_ERRM_*StorePermanentCode* (XX)

Endif

Endif

Endif

Conditions to decrement the driving cycle counter :

At transition LV_DC 1 -> 0:

If LV_ERR_DC[IDX] = 0
and LV_END_DIAG_XX = 1
Then LV_ERR_PND[IDX] = 0 (Pending fault is erased)
If LV_ERR_CFM[IDX] = 1 (for confirmed failure)
Then **If** LV_DC_MAX[IDX] =1
Then CTR_DC[IDX] = CTR_DC[IDX] - DC_DEC_XX
If CTR_DC[IDX] $\leq$ 0
Then LV_DC_MAX[IDX] = 0
CTR_DC[IDX] = 0
ACTION_ERRM_*ErasePermanentCode* (XX)
Endif
Else CTR_DC[IDX] = 0
Endif
Else (for temporary failure)
CTR_DC[IDX] = 0
LV_ERR_DISA[IDX] = 1 (failure disappeared)
Endif
Endif

Treatment for failures using SCDN functionality (LC_ENA_SCDN=1 and NLC_ENA_SCDN=1) (running within DC) stored in 2nd layer memory.

Conditions to increment the driving cycle counter :

Done upon ACTION_ERRM_IncrementDCctrSCDN call:

(see also similar conditions specification)

Description:

Syntax : ACTION_ERRM_IncrementDCctrScdn (IN <XX>)

Parameter (in) : XX failure in static memory with XX=F(IDX)

Parameter (out) :

Short description:

With the ACTION_ERRM_IncrementDCctrScdn (IN <XX>) the driving cycle counter for the failure XX stored in 2nd layer memory at position IDX, which is using similar conditions functionality, is incremented. This action is called at the end of a diagnosis window in the following cases :

- failure occurred and similar conditions are recorded
- failure already stored in a previous driving cycle. Now failure gets present again and was also present in last driving cycle.
- failure already stored in a previous driving cycle. Now failure gets present again with SCDN recognition for detected failure.
- failure already stored in a previous driving cycle and LV_DC_MAX[IDX] = 1 (MIL is illuminated, if LC_MIL_ON =1).

For details see similar conditions specification.

Application conditions:

Initialization: -

Recurrence: -

Activation: at action request

Formula section:

ACTION_ERRM_IncrementDCctrScdn (IN<XX>) :

XX=f(IDX)

If LV_ERR_MEM_XX = 1 (failure XX is stored in 2nd layer memory)

Then LV_ERR_DISA[IDX] = 0

If LV_DC_MAX[IDX] =1 (for confirmed failure with MIL on)

Then

CTR_DC[IDX] = DC_MAX_XX (counter is forced to max.)

ACTION_ERRM_*GetMILRelevant*(XX,PRM_LV_MIL_RLV)

If PRM_LV_MIL_RLV= 1

(MIL illuminated and Failure OBD relevant)

Then

Permanent fault code has to be stored, if not yet done due to memory full

ACTION_ERRM_*StorePermanentCode* (XX)

Endif

Else ACTION_ERRM_ConfirmErrScdn (OUT<DCconf>, IN<XX>)

(allows direct failure confirmation; see similar conditions Appl.Inc.)

If DCconf = YES

Then

CTR_DC[IDX] = DC_MAX_XX (counter is forced to max.)

Else

CTR_DC[IDX] = CTR_DC[IDX] + DC_INC_XX

Endif

If CTR_DC[IDX] ≥ DC_MAX_XX

Then

CTR_DC[IDX] = DC_MAX_XX

LV_DC_MAX[IDX] = 1 (failure confirmed)

LV_ERR_CFM[IDX] = 1

LV_ERR_TMP[IDX] = 0

ACTION_ERRM_*GetMILRelevant*(XX,PRM_LV_MIL_RLV)

If PRM_LV_MIL_RLV= 1

(MIL illuminated and Failure OBD relevant)

Then

Permanent fault code has to be stored

ACTION_ERRM_*StorePermanentCode* (XX)

Endif

Endif

Endif

Endif

<u>Conditions to decrement the driving cycle counter :</u>

At transition LV_DC 1 -> 0:

If LV_ERR_DC[IDX] = 0

and LV_END_DIAG_XX = 1

Then ACTION_ERRM_DecrementDCctrSCDN (OUT<DCdec>, IN<XX>)

(with XX=F(IDX))

The similar conditions functionality decides, if it's possible to decrement the driving cycle counter.

```
If      DCdec = YES                            (decrementing is possible)
Then    LV_ERR_PND[IDX] = 0                    (Pending fault is erased)
        If      LV_ERR_CFM[IDX] = 1            (for confirmed failure)
        Then    If      LV_DC_MAX[IDX] = 1
                Then    CTR_DC[IDX] = CTR_DC[IDX] - DC_DEC_XX
                        If      CTR_DC[IDX] ≤ 0
                        Then    LV_DC_MAX[IDX] = 0
                                CTR_DC[IDX] = 0
                                ACTION_ERRM_ErasePermanentCode (XX)
                        Endif
                Else
                        CTR_DC[IDX] = 0
                Endif
        Else                                   (for temporary failure)
           CTR_DC[IDX] = 0
           LV_ERR_DISA[IDX] = 1                (failure disappeared)
        Endif
Endif
Endif
```

B.51.3.6 Warm-Up counter management

Description:

As soon as the failure gets present and stored in the memory, the warm-up cycle counter is initialized.

With every valid warm-up cycle with no error occurrence, the warm-up cycle counter is decremented by 1. If the warm-up cycle counter reaches zero the error is deleted from the 2nd layer memory.

Application conditions:

Initialization:	on saved ram lost CTR_WUP_CYC[IDX]= 0
Recurrence:	-
Activation:	only at transition LV_DC 1 -> 0

Formula section:

Treatment for all failures not using SCDN functionality (or LC_ENA_SCDN = 0) stored in 2nd layer memory :

Condition to decrement the warm up counter :

The following treatment is done for all failure stored in 2nd layer memory **and only at transition LV_DC 1 -> 0** (**before** treatment of driving cycle counter decrementing at end of driving cycle):

If (LV_WUP_CYC = 1 **and** C_CTR_MAX_WUP_CYC != 255)
the warm-up cycle is recognised and failure healing is possible
Then **If** (CTR_DC[IDX] = 0 **and** LV_ERR_DC[IDX] = 0)
the MIL is extinguished and no failure in this driving cycle
or
(CTR_DC[IDX] != 0 **and** LV_DC_MAX[IDX] = 0
case before the MIL may be illuminated
and LV_ERR_DC[IDX] = 0 **and** LV_END_DIAG_XX = 1)
the failure is no more pending
Then CTR_WUP_CYC[IDX] = CTR_WUP_CYC[IDX] - 1
If CTR_WUP_CYC[IDX] = 0
Then ACTION_ERRM_EraseErr (IN<IDX>)
Endif
Endif
Endif

Treatment for failures using SCDN functionality (and LC_ENA_SCDN = 1) (running within DC) stored in 2nd layer memory.

Condition to decrement the warm up counter :

The following treatment is done for all failure stored in 2nd layer memory **and only at transition LV_DC 1 -> 0** (**before** treatment of driving cycle counter decrementing at end of driving cycle):

If LV_WUP_CYC = 1
Then **If** CTR_DC_IDX = 0
and LV_ERR_DC_IDX = 0
Then CTR_WUP_CYC_IDX = CTR_WUP_CYC_IDX - 1
If CTR_WUP_CYC_IDX = 0
Then ACTION_ERRM_EraseErr (IDX)
Endif
Endif
Endif

Condition to reload the warm up counter :

The following treatment is done for failure XX when it is stored in 2nd layer memory **when ACTION_ERRM_StoreErr (IN<XX>) is called :**

CTR_WUP_CYC[IDX] = C_CTR_MAX_WUP_CYC

B.51.4 Miscellaneous services to get information

B.51.4.1 API for determination of pending status of failures

General information:

This API shall be called to determine if any XX failure is pending or not.

Description:

Syntax: ACTION_ERRM_CheckPendingStatus (IN <PRM_IDX_DIAG>, OUT <PRM_LV_ERR_PND>)

Parameter(in): PRM_IDX_DIAG Diagnostic instance of the XX failure

Parameter(out): PRM_LV_ERR_PND State of the failure :

= 1 failure is pending (or it's impossible to determine its status)

= 0 failure is not pending

Short Description: This API shall be used to verify if a failure stored in the dynamic memory has the pending status or not. When it impossible to determine if the fault is pending or not (failure not store because the dynamic memory is full), this failure should be considered as pending anyway.

Application conditions:

Initialisation: -

Recurrence: -

Activation: at action call

Deactivation: -

Formula section:

If (1) *{ the XX failure analysed is present in the dynamic memory }*

LV_ERR_MEM_XX = 1 (with XX = F(IDX))

Then (1)

If (2) *{ IDX is the failure index, in the dynamic memory }*

LV_ERR_PND[IDX] = 1

Then (2)

PRM_LV_ERR_PND = 1

Else (2)

PRM_LV_ERR_PND = 0

Endif (2)

Else (1)

{ failure not stored in the dynamic memory }

If (2) LV_ERR_XX = 1

Then (2)

{ the failure is stored in the static memory but it's not present in the dynamic memory : impossible to determine its status }

{ then the failure shall be considered as pending }

PRM_LV_ERR_PND = 1

Else (2)

{ the failure is neither stored in the static memory nor in the dynamic memory }

PRM_LV_ERR_PND = 0

Endif (2)

Endif (1)

B.51.4.2 Get the "Failure set since last clear" state

Description for ACTION_ERRM_*GetErrLastClr*

ACTION_ERRM_*GetErrLastClr* (<PRM_IDX_ERR>, <PRM_LV_ERR_LST_CLR>)					
This action indicates if a failure has been set already once since the last failure memory clear.					
Parameter	**Type**	**Hex. Limits**	**Phys. limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Diagnostic failure instance					
PRM_LV_ERR_LST_CLR	OUT	0...1H	0...1	1	[-]
Failure set at least once since last failure memory clear					

General information:

This API shall be called to determine if any XX failure has been set at least one time since memory clear

Formula section:

If PRM_IDX_ERR != 0

Then

with XX = PRM_IDX_ERR

PRM_LV_ERR_LST_CLR = LV_ERR_LST_CLR_XX

Else

PRM_LV_ERR_LST_CLR = 0

Endif

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_ERR_DYN	1	1...FFH	1...255	1	[-]
Maximum number of failure defined in dynamic structure (typical value : 10)					

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CTR_MAX_WUP_CYC	1	0...FFH	0...255	1	[-]
Initialization value for the warm-up cycle of diagnostics					
LC_ERR_FMY_CLR	1	0...1H	0...1	1	[-]
Boolean to clear failure memory when set to 1					

B.52 Dynamic Error Management (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_WUP_CYC	V/O	0...1H	0...1	1	[-]
Warm-up cycles calculation inhibition					
TCO_DSL_CMN	V/O	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature (used only in ERRM for GS-DS diversity)					
TCO_ST_DSL_CMN	V/O	0...FEH	-48...142.5	0.75	[°C]
Coolant temperature at start (used only in ERRM for GS-DS diversity)					

Input data:

TCO	TCO_ST		

Export actions:

ACTION_ERRM_TrigErrDyn (IN <XX>)
This action is called by the error management function when a new failure occurs

B.52.1 Warm-up cycle inhibition

FUNCTION DESCRIPTION:

Description:

Application conditions:

Initialisation: at ECU reset

LV_INH_WUP_CYC = 0

TCO_DSL_CMN = TCO

TCO_ST_DSL_CMN = TCO_ST

Recurrence: 100 ms

Activation: LV_IGK = 1

Deactivation: -

Formula section:

Project shall describe hereafter all conditions to inhibit warm-up cycle counter computation (e.g. with TCO failure, the warm-up cycle evaluation isn't possible).

By default configuration, the warm-cycle computation is not inhibited.

LV_INH_WUP_CYC = 0

The following data are used for different data type between DS and GS:

TCO_DSL_CMN = TCO

TCO_ST_DSL_CMN = TCO_ST

B.52.2 API for Trigger Event

General information:

This file defines the Action "ACTION_ERRM_TrigErrDyn" called by the error management function when a failure occurs or disappears.

The contain of this action should be provided by the project team

Description:

Syntax: ACTION_ERRM_TrigErrDyn (IN <XX>)

Parameter (in): XX diagnostic instance of failure ≡ index of failure in static memory

Parameter (out): -

Short description: This action allows to inform the communication tools in line as soon as a failure occurs

Application conditions:

Initialisation: -

Recurrence: -

Activation: at action request

Formula section:

ACTION_ERRM_TrigErrDyn (XX):

B.53 Cycle manager

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DC	V/O	0...1H	0...1	1	[-]
driving cycle status flag					
LV_WUP_CYC	V/O	0...1H	0...1	1	[-]
Warm-up cycle status flag					
LV_STATE_WUP	V/O	0...1H	0...1	1	[-]
tco state for exceeding the warm-up cycle tco threshold					

Input data:

LV_IGK	LV_ST_END	LV_ES	TCO_DSL_CMN
TCO_ST_DSL_CMN	LV_INH_WUP_CYC		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_WUP_THD	1	0...FEH	-48...142.5	0.75	[°C]
Minimum value for warm-up cycle detection					
C_TCO_WUP_INC	1	0...FEH	-48...142.5	0.75	[°C]
Minimum increase of the coolant temperature for warm-up cycle detection					
C_TCO_ST_MAX_INH_WUP_CYC	1	0...FEH	-48...142.5	0.75	[°C]
maximum engine temperature at start to detect warm up cycle (customer requirement 71,1°C)					

Import actions:

ACTION_ERRM_*ClcPermanentResetDC* ( )

B.53.1 Driving cycle

FUNCTION DESCRIPTION:

General information:

Description:

A driving cycle consists of a vehicle operation phase from engine startup to engine shutoff and includes the power-lach phase after the engine running phase (In order manage all OBD failures set only in PWL phase).

After key is on, the flag LV_DC is set when the engine exits from “engine start” state.

The flag LV_DC is reset at the end of the power-latch phase before the NVMY management. In case of a LV_IGK 0→1 transition during the powerlatch phase or at ECU reset LV_DC is also reseted.

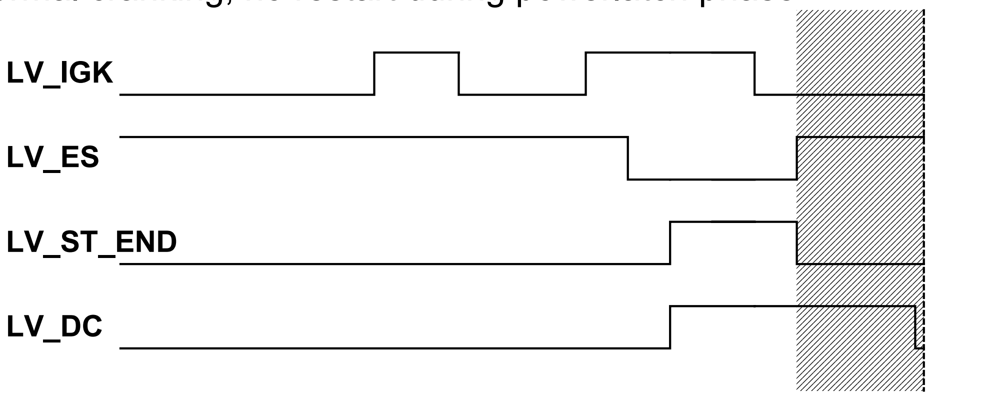
Remark :

For simplification and global coherence, the following rule is applied:

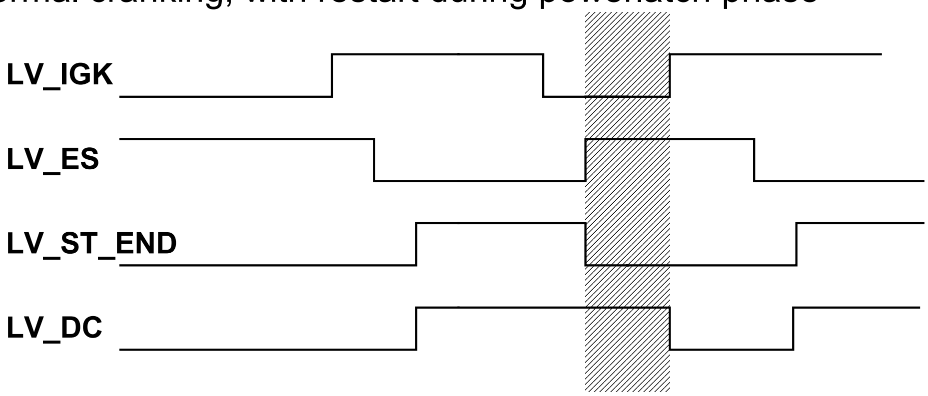
Engine stalling is ignored (if engine restarts after stalling, without key-off/key-on transition, the cycle is considered as the same driving cycle).

Signal flow diagram:

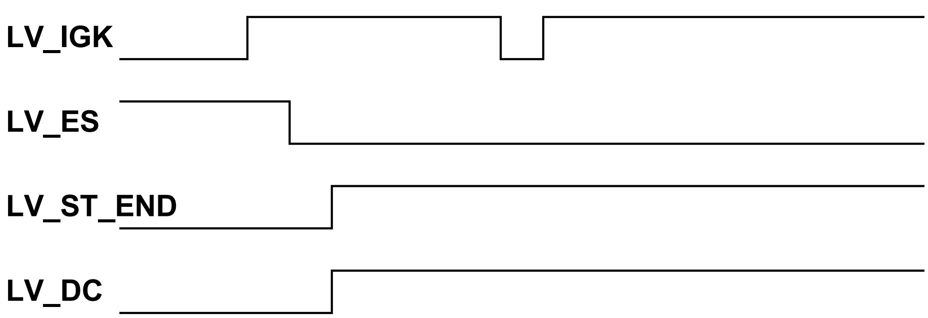
Normal cranking, no restart during powerlatch phase



Normal cranking, with restart during powerlatch phase



Short key off/on while engine is running



Application conditions:

Initialisation: At ECU reset

LV_DC = 0

Before storage of data of the error management within the NVMY the driving cycle is finished.

Recurrence: -

Activation: see Formula section

Formula section:

Activation:

If LV_ST_END 0→1 transition

Then

LV_DC = 1

Endif

Deactivation:

If LV_IGK 0→1 transition

and

LV_ES =1

Then

LV_DC = 0

Before clearing static and dynamic memory (e.g. LV_END_DIAG_XX) call :

ACTION_ERRM_*ClcPermanentResetDC*()

Endif

If End of power-latch phase is reached (before NVMY storing)

Then

LV_DC = 0

Before clearing static and dynamic memory (e.g. LV_END_DIAG_XX) call :

ACTION_ERRM_*ClcPermanentResetDC*()

Endif

B.53.2 Warm-Up Cycle and Warm-Up Status

FUNCTION DESCRIPTION:

General information:

The detection of the warm-up cycle is based on the coolant temperature. The warm-up cycle is detected as soon as the coolant temperature reaches C_TCO_WUP_THD and the change with regard to the starting temperature exceeds C_TCO_WUP_INC. These values are defined by law, so a change of these values is only for development validation commodity.

The application must guarantee warm-up cycle detection when a coolant temperature sensor error exists.

Therefore if a coolant temperature sensor error has been debounced, the threshold for warm-up cycle detection are referred to the backup coolant temperature value computed by the engine control (see chapter “Diagnosis and emergency operation“).

The warm-up status is needed for similar driving conditions test.

Application conditions:

Initialisation: At ECU reset or LV_DC 1→0 transition (after warm-up cycle counter management)

LV_WUP_CYC =0
LV_STATE_WUP = 0

Recurrence: 1 s

Activation: LV_DC = 1

Formula section:

```
If    LV_INH_WUP_CYC = 0
Then
        If    TCO_DSL_CMN > C_TCO_WUP_THD
        Then
              LV_STATE_WUP = 1
              If    TCO_DSL_CMN > (TCO_ST_DSL_CMN + C_TCO_WUP_INC)
                    and
                    TCO_ST_DSL_CMN < C_TCO_ST_MAX_INH_WUP_CYC
              Then
                          LV_WUP_CYC = 1
              Endif
        Endif
Endif
```

Calibration hint:

C_TCO_WUP_THD =	at least 72 °C for gasoline engines	(OBD II requirement)
	at least 70 °C for gasoline engines	(EOBD requirement)
C_TCO_WUP_INC =	at least 23 °C for gasoline engines	(OBD II requirement)
	at least 22 °C for gasoline engines	(EOBD requirement)

In case of Diesel systems, recommended values are the same, except :

C_TCO_WUP_THD = at least 60 °C for diesel engines (OBD II requirement)

C_TCO_ST_MAX_INH_WUP_CYC = 71,1 °C (customer specific requirement)

B.54 Environmental data

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ENVD_OBD[NC_NR_ENVD_OBD][NC_NR_ERR_DYN]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_OBD : environmental data (fixed by law)					
ENVD_CUS_CMN[NC_NR_ENVD_CUS_CMN][NC_NR_ERR_DYN]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_CMN : environmental data common to all failure (not fixed by law)					
ENVD_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN][NC_NR_FRF_SET][NC_NR_ERR_DYN]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_SET_CMN : environmental data common to all failure (not fixed by law) stored in set					
ENVD_CUS_SET_SPC[NC_NR_ENVD_CUS_SET_SPC][NC_NR_FRF_SET][NC_NR_ERR_DYN]	V/O/S	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_SET_SPC : environmental data specific to each failure (not fixed by law) stored in set					
ENVD_PREV_OBD[NC_NR_ENVD_OBD][NC_NR_ENVD_PREV]	O/S	0...FFH	0...255	1	[-]
Prestored freeze frame FRF_PREV_OBD as content for FRF_OBD;					
ENVD_PREV_CUS_CMN[NC_NR_ENVD_CUS_CMN][NC_NR_ENVD_PREV]	O/S	0...FFH	0...255	1	[-]
Prestored freeze frame FRF_PREV_CUS_CMN as content for FRF_CUS_CMN;					
ENVD_PREV_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN][NC_NR_ENVD_PREV]	O/S	0...FFH	0...255	1	[-]
Prestored Freeze frame FRF_PREV_CUS_SET_CMN as content for FRF_CUS_SET_CMN					
ENVD_PREV_CUS_SET_SPC[NC_NR_ENVD_CUS_SET_SPC][NC_NR_ENVD_PREV]	O/S	0...FFH	0...255	1	[-]
Prestored freeze frame FRF_PREV_CUS_SET_SPC :as content for FRF_CUS_SET_SPC					
IDX_FRF[NC_NR_ERR_DYN]	V/O/S	0...FFH	0...255	1	[-]
Pointer to make the link between the failure and the freeze frame					

Note:

The link between the failure stored in the dynamic error mangement at position *IDX* is made with IDX_FRF[*IDX*]. E. g. a failure stored at the position *IDX* = 2 belongs to freeze frame ENVD_..._[m] with m = IDX_FRF[2]

Input data:

ENVD_CONF_OBD[NC_NR_ENVD_OBD]	ENVD_CONF_CUS_CMN[NC_NR_ENVD_CUS_CMN]	ENVD_CONF_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN]
CTR_FRC[NC_NR_ERR_DYN]	NC_NR_ERR_DYN	LV_ERR_DISA[NC_NR_ERR_DYN]

Export actions:

ACTION_ERRM_StoreFrf (IN <IDX>)
This action stores the freeze frame of the failure IDX
ACTION_ERRM_EraseFrf (IN <IDX>)
This action erases the freeze frame of the failure IDX
ACTION_ERRM_StorePrevFrf (IN <XX>)
This action stores the prestored freeze frame for the diagnosis instance XX

FUNCTION DESCRIPTION:

General information:

This module describes the structure of the freeze frame and manages the storage and deleting of the freeze frame.

A freeze frame is a group of vehicle environment data catched when a failure occurs.

Freeze frame is filled up with variables, which are not limp home values : the variables are set up before going to the limp home.

A freeze frame is made of many parts :

- FRF_OBD: Environmental data, which are common to all failures and stored on time. It's only updated, if the failure has disappeared before and occurs again.

  The content is defined by law. It's called usually CARB Freeze Frame and the content is according to SAE J1979.

  Size of this part is defined by configuration data NC_NR_ENVD_OBD affected value is in the Environmental data (Appl. Inc.) module.

  Contents of this part is fixed and defined in Environmental data (Appl. Inc.) module.

- FRF_CUS_CMN : Environmental data, which are common to all failure.

  It's stored one time, if NC_ ENVD_CUS_CMN_UPD = 0.

  With NC_ ENVD_CUS_CMN_UPD = 1, FRF_CUS_CMN is updated, if the failure was disappeared before and occurs again.

  The content is not required by law.

  Size of this part is defined by configuration data NC_NR_ENVD_CUS_CMN; affected value is in the Environmental data (Appl. Inc.) file.

  Contents of this part is fixed and defined in the Environmental data (Appl. Inc.) module.

- FRF_CUS_SET_CMN : Environmental data, which are common to all failure and stored many times. The content is not required by law. For each failure occurrence several catches (called sets) can be done.

  Size of this part is defined by configuration data, affected value is in the Environmental data (Appl. Inc.) module :

  NC_NR_FRF_SET for number of sets per freeze frame

  NC_NR_ENVD_CUS_SET_CMN for number of common data (in bytes) in each set

  Contents of this part is fixed and defined in the Environmental data (Appl. Inc.) module.

- FRF_CUS_SET_SPC : Environmental data, which are specific to each failure and stored many times. The content is not required by law. For each failure occurrence several catches (called sets) can be done.

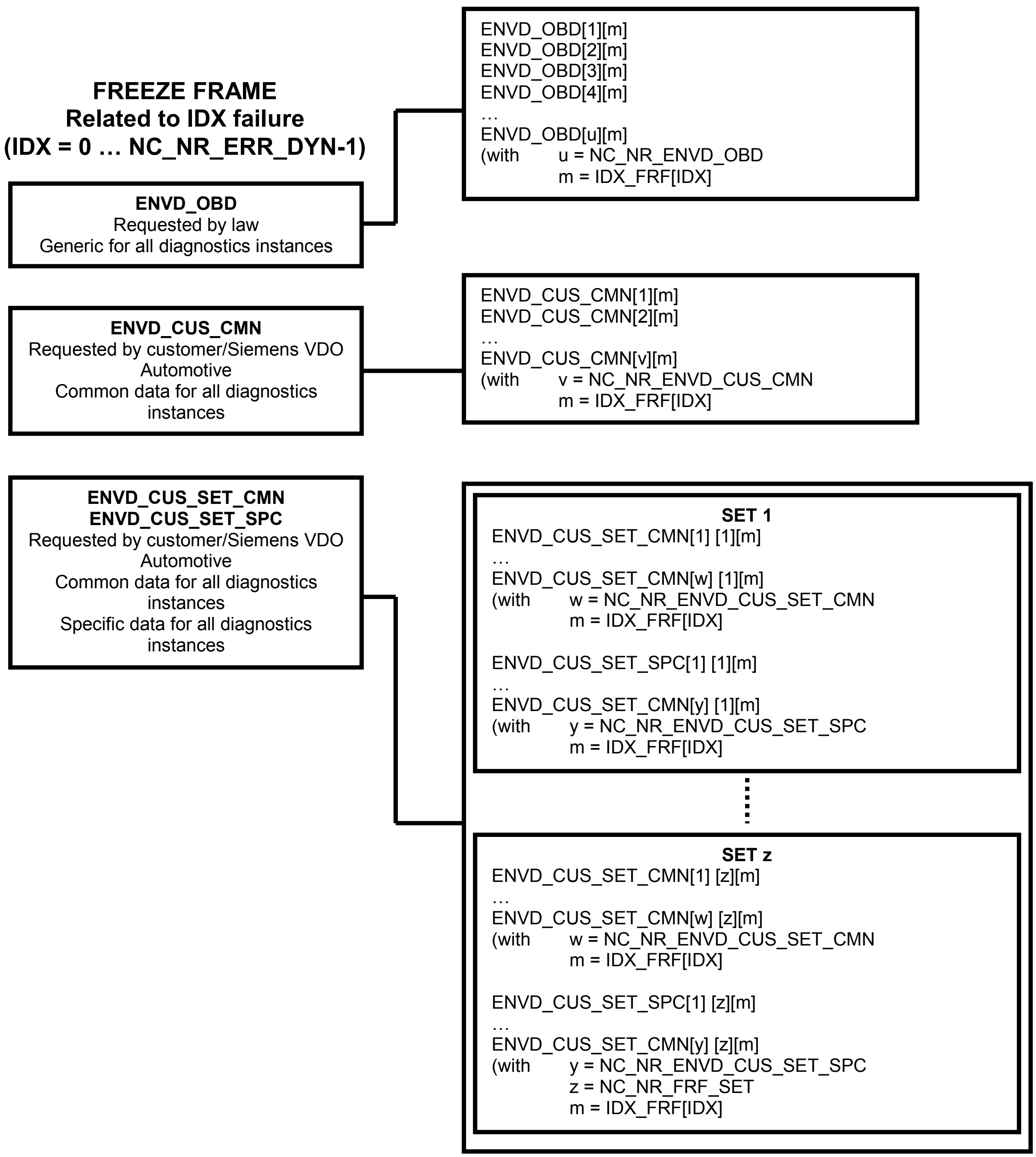
  Size of this part is defined by configuration data, affected value is in Environmental data (Appl. Inc.) module :

  NC_NR_FRF_SET for number of sets per freeze frame

  NC_NR_ENVD_CUS_SET_SPC for number of specific data (in bytes) in each set

  Contents of this part is defined by calibration, process is described in the last chapter of this module.

Signal flow diagram:



Remark: This signal flow diagram doesn't include the prestored freeze frame functionality described in chapter 1.4.

B.54.1 Freeze frame storage

Description:

Syntax : ACTION_ERRM_StoreFrf (IN <IDX>).

Parameter (in) : IDX Index of failure in 2nd layer memory to store the freeze frame

Parameter (out) : -

Short description :

When the failure IDX gets present, the storage of a freeze frame is done in the 2nd layer memory by calling the action ACTION_ERRM_StoreFrf (IDX).

If several failures occur (TCO sensor, injector 1, injector 4), a freeze frame is stored for each failure. For ram memory limitation, the number of freeze frame is limited to NC_NR_ERR_DYN.

If same failure occurs several times (occurrence of the failure is defined by the frequency counter CTR_FRC[IDX]), the following rules are applied :

- 1st occurrence :

  ⇨ Store FRF_OBD +FRF_CUS_CMN +FRF_CUS_SET_CMN + FRF_CUS_SET_SPC

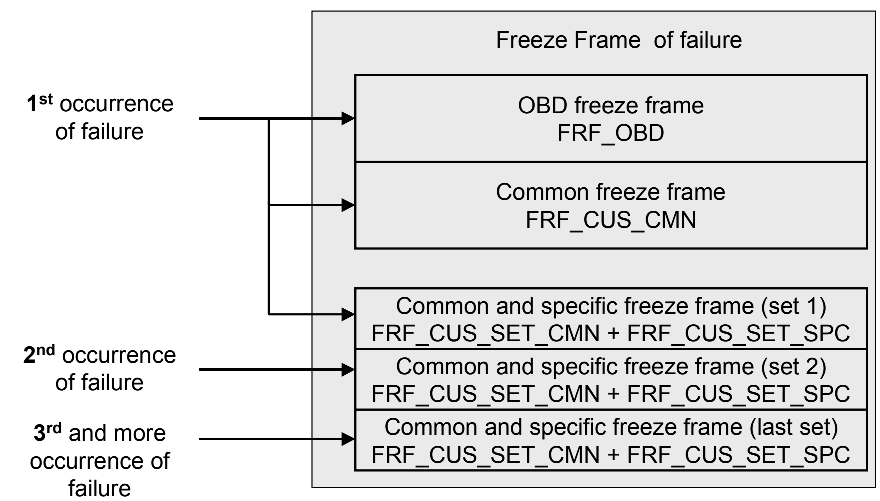
- occurrence ∈ [2 to NC_NR_FRF_SET]

  ⇨ Store FRF_CUS_SET_CMN + FRF_CUS_SET_SPC

- occurrence > NC_NR_FRF_SET

  ⇨ Replace last FRF_CUS_SET_CMN + FRF_CUS_SET_SPC

Example with NC_NR_FRF_SET set to 3



Freeze frame informations are available as long as the failure is not erased in 2nd layer memory.

Remark: Special behavior for FRF_OBD and FRF_CUS_CMN in case of failure status change from disappeared to temporary/confirmed.

Application conditions:

Initialization: after system initialization the freeze frame is restored from non volatile memory or in case of a damaged non volatile memory the freeze frame is initialized with zero

Recurrence: -

Activation: at Action request

Formula section:

ACTION_ERRM_StoreFrf (IDX) :

If CTR_FRC[IDX] = 1

Then (1st occurrence : storage

With u = 1 ... NC_NR_ENVD_OBD

v = 1 ... NC_NR_ENVD_CUS_CMN

w = 1 ... NC_NR_ENVD_CUS_SET_CMN

y = 1 ... NC_NR_ENVD_CUS_SET_SPC)

If (NC_NR_ENVD_PREV != 0 and

XX = diagnosis instance XX using prestored freeze frame functionality (see Environmental data (Appl. Inc.) module) with XX = F(IDX))

Then ENVD_OBD[u][m] = ENVD_PREV_OBD[u][t]

ENVD_CUS_CMN[v][m] = ENVD_PREV_CUS_CMN[v][t]

ENVD_CUS_SET_CMN[w][1][m] = ENVD_PREV_CUS_SET_CMN[w][t]

ENVD_CUS_SET_SPC[y][1][m] = ENVD_PREV_CUS_SET_SPC[y][t]

(with **m=IDX_FRF[IDX]**

with t related to failure XX, see Environmental data Appli.Inc.)

Else ENVD_OBD[u][m] = ENVD_CONF_OBD[u]

ENVD_CUS_CMN[v][m] = ENVD_CONF_CUS_CMN[v]

ENVD_CUS_SET_CMN[w][1][m] = ENVD_CONF_CUS_SET_CMN[w]

ENVD_CUS_SET_SPC[y][1][m] = ID_ERR_ENVD_XX[y]

(with XX = F(IDX))

Endif

Else (occurrence 2 to NC_NR_FRF_SET

With w = 1 .. NC_NR_ENVD_CUS_SET_CMN

y = 1 ... NC_NR_ENVD_CUS_SET_SPC

z = CTR_FRC[IDX], if CTR_FRC[IDX] ≤ NC_NR_FRF_SET : storage

z = NC_NR_FRF_SET, if CTR_FRC[IDX] > NC_NR_FRF_SET : update

(with **m=IDX_FRF[IDX]**)

If (NC_NR_ENVD_PREV != 0 **and**

XX = diagnosis instance XX using prestored freeze frame functionality

(see Environmental data (Appl. Inc.) module) with XX = F(IDX))

Then **If** (LV_ERR_DISA[IDX] = 1)

failure was in disappeared state before this occurrence

Then ENVD_OBD[u][m] = ENVD_PREV_OBD[u][t]

FRF_OBD is updated

If (NC_ ENVD_CUS_CMN_UPD = 1)

Then ENVD_CUS_CMN[v][m] = ENVD_PREV_CUS_CMN[v][t]

FRF_CUS_CMN is updated

Endif

Endif

ENVD_CUS_SET_CMN[w][z][m] = ENVD_PREV_CUS_SET_CMN[w][t]

ENVD_CUS_SET_SPC[y][z][m] = ENVD_PREV_CUS_SET_SPC[y][t]

(with t related to failure XX, see Environmental data Appli.Inc.)

Else **If** (LV_ERR_DISA[IDX] = 1)

Failure was in disappeared state before this occurrence

Then ENVD_OBD[u][m] = ENVD_CONF_OBD[u]

Freeze frame ENVD_OBD is updated

If (NC_ ENVD_CUS_CMN_UPD = 1)

Then ENVD_CUS_CMN[v][m]= ENVD_CONF_CUS_CMN[v]

FRF_CUS_CMN is updated

Endif

Endif

ENVD_CUS_SET_CMN[w][z][m] = ENVD_CONF_CUS_SET_CMN[w]

ENVD_CUS_SET_SPC[y][z][m] = ID_ERR_ENVD_XX [y]

(with XX = F(IDX))

Endif

Endif

B.54.2 Freeze frame deleting

Description:

Syntax : ACTION_ERRM_EraseFrf (IN <IDX>).

Parameter (in) : IDX Index of failure in 2nd layer memory to store the freeze frame

Parameter (out) : -

Short description :

When the failure IDX is erased from 2nd layer memory, the freeze frame of this failure is also erased by calling the action ACTION_ERRM_EraseFrf (IDX).

Application conditions:

Initialization: -

Recurrence: -

Activation: at Action request

Formula section:

ACTION_ERRM_EraseFrf (IDX) :

(Erase OBD freeze frame, common freeze frame and all specific freeze frame

With u = 1 ... NC_NR_ENVD_OBD

v = 1 ... NC_NR_ENVD_CUS_CMN

w = 1 ... NC_NR_ENVD_CUS_SET_CMN

y = 1 ... NC_NR_ENVD_CUS_SET_SPC

z = 1 ... NC_NR_FRF_SET)

ENVD_OBD[u][m] = 0

ENVD_CUS_CMN[v][m] = 0

ENVD_CUS_SET_CMN[w][z][m] = 0

ENVD_CUS_SET_SPC[y][z][m] = 0

B.54.3 Calibratable set of environmental data

Description:

The freeze frame part called FRF_CUS_SET_SPC contains environmental data, which are :

- specific to each failure, not required by law ;
- and calibratable.

This chapter describes "how to define by calibration the contents of FRF_CUS_SET_SPC".

The calibratable system of environmental data is managed via tables defined as following:

ID_ENVD_FAC : Configuration table, which describes all possible environmental data, which can be stored in a freeze frame (specific part) ;

Affected value defined in Environmental data (Appl. Inc.) module.

ID_ERR_ENVD_XX : Calibration table, which defines for each failure XX a maximum of NC_NR_ENVD_CUS_SET_SPC environmental data, which should be stored.

Example :

To store the environmental data TIA and TCO_ST in case of TCO failure ;

MAP and AMP in case of EGR failure ;

First, you should defined the possible environmental data in ID_ENVD_FAC table :

ID_ENVD_FAC[1] = &TIA

ID_ENVD_FAC[2] = &TCO_ST

ID_ENVD_FAC[3] = &MAP

ID_ENVD_FAC[4] = &AMP

Then, you should declare for each failure, the environmental data to stored :

ID_ERR_ENVD_TCO[1] = 1 ; ID_ERR_ENVD_TCO [2] = 2

ID_ERR_ENVD_EGR[1] = 3 ; ID_ERR_ENVD_EGR [2] = 4

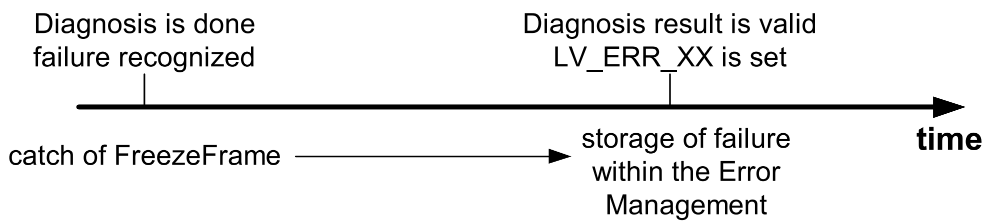
B.54.4 Prestored freeze frame

FUNCTION DESCRIPTION:

General information:

Usually the content for the freeze frame is catched, when the failure is stored within the failure memory. In special cases, when the entry of a failure within the failure memory is delayed, it's necessary to catch the content for the freeze frame at the same time as the failure information is available.

Example:



Description:

Syntax : ACTION_ERRM_StorePrevFrf (IN <XX>).

Parameter (in) : XX diagnosis instance

Parameter (out) : -

Short description :

If ACTION_ERRM_StorePrevFrf (XX) is called by the diagnosis instance XX, a "prestored" freeze frame for diagnosis instance XX is stored. It's used as content for the freeze frame, if the failure is entered in the failure memory later.

Application conditions:

Initialization: -

Recurrence: -

Activation: at Action request

Formula section:

ACTION_ERRM_StorePrevFrf (XX) :

with t related to failure XX, see Environmental data Appli.Inc.

ENVD_PREV_OBD[u][t] = ENVD_CONF_OBD[u]

ENVD_PREV_CUS_CMN[v][t] = ENVD_CONF_CUS_CMN[v]

ENVD_PREV_CUS_SET_CMN[w][t] = ENVD_CONF_CUS_SET_CMN[w]

ENVD_PREV_CUS_SET_SPC[y][t] = ID_ERR_ENVD_XX[y] (with XX = F(IDX))

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
ID_ERR_ENVD_XX	NC_NR_ENVD_CUS_SET_SPC	1...FFH	1...255	1	[-]
LDP_1_ID_ERR_ENVD_XX	NC_NR_ENVD_CUS_SET_SPC	0...FFH	0...255	1	[-]
Freeze frame FRF_CUS_SET_SPC : environmental data contents for failure XX					

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_NR_ENVD_OBD	-	1...FFH	1...255	1	[-]
Number of different environment data (in bytes) which are fixed by the law (stored one time)					
NC_NR_ENVD_CUS_CMN	-	1...FFH	1...255	1	[-]
Number of different environment data (in bytes) which are common for all diagnosis instance (stored one time)					
NC_NR_ENVD_CUS_SET_CMN	-	0...FFH	0...255	1	[-]
Number of different environment data (in bytes) which are common for all diagnosis instance (stored many time)					
NC_NR_ENVD_CUS_SET_SPC	-	1...FFH	1...255	1	[-]
number of different environment data (in bytes) which are specific to each diagnosis instance (stored many time)typical value is 4					
NC_NR_FRF_SET	-	1...FFH	1...255	1	[-]
Number of different environment data groups (set) which can be store for one freeze frametypical value is 3 ; the value 0 is forbidden					
NC_ENVD_CUS_CMN_UPD	-	0...1H	0...1	1	[-]
Selection of FRF_CUS_CMN update method					
NC_NR_ENVD_PREV	-	1...FFH	1...255	1	[-]
Number of different failure instances XX using prestored freeze frameIf NC_NR_ENVD_PREV = 0 , then no prestored freeze frame functionality required					
NC_ID_ENVD_FAC	-	1...FFH	1...255	1	[-]
Size of ID_ENVD_FAC table					
ID_ENVD_FAC[255]	-	1...FFH	1...255	1	[-]
Possible stored environmental data in FRF_CUS_SET_SPC					

Configuration data detailed description:

NC_NR_ENVD_CUS_SET_SPC : Typical value is 4.

NC_NR_FRF_SET : Typical value is 3.
The value 0 is forbidden.

NC_ ENVD_CUS_CMN_UPD 0: FRF_CUS_CMN is not updated when failure status changes from "disappeared" to "temporary/confirmed".
1: FRF_CUS_CMN is updated when failure status changes from "disappeared" to "temporary/confirmed".

NC_NR_ENVD_PREV 0: No pre-stored freeze frame functionality required

Size of ID_ENVD_FAC is defined according number of data in the table ; maximum size is 255.

B.55 Environmental data (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD	O/V	0...FFH	0...200.54	0.786	[° CRK]
CAM_DIF_INT_DIAG_IVVT_EX with lower resolution					
CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD	O/V	0...FFH	0...200.54	0.786	[° CRK]
CAM_DIF_INT_DIAG_IVVT_IN with lower resolution					
CPPWM_ENVD	O/V	0...FFH	0...99.61	0.3906	[%]
CPPWM with lower resolution					
CPU_LOAD_MAX_ENVD_H	O/V	0...04H	0...0	25	[%]
CPU_LOAD_MAX high byte					
CPU_LOAD_MAX_ENVD_L	O/V	0...FFH	0...0	0.09765	[%]
CPU_LOAD_MAX low byte					
CRU_BAS_ENVD_H	O/V	0...3H	0...3.75	1.245	[V]
CRU_BAS high Byte					
CRU_BAS_ENVD_L	O/V	0...FFH	0...1.245	0.0048	[V]
CRU_BAS low Byte					
DIAG_INST_ENVD	O/V	0...FFH	0...255	1	[-]
Diagnosis instance causing current ENVD request					
DTP_DIF_ACT_ENVD	O/V	80...7FH	-40.96...40.64	0.32	[hPa]
DTP_DIF_ACT with lower resolution					
DTP_DIF_COR_ENVD_H	O/V	80...7FH	-40.96...40.64	0.32	[hPa]
DTP_DIF_COR high Bye					
DTP_DIF_COR_ENVD_L	O/V	80...7FH	0.16...0.159	0.00125	[hPa]
DTP_DIF_COR low Bye					
DTP_DIF_ENVD_H	O/V	80...7FH	-40.96...40.64	0.32	[hPa]
DTP_DIF_ high Byte					
DTP_DIF_ENVD_L	O/V	80...7FH	0.16...0.159	0.00125	[hPa]
DTP_DIF_ low Byte					
DTP_ENVD	O/V	80...7FH	-40.96...40.64	0.32	[hPa]
DTP with lower resolution					
ENVD_CONF_CUS_CMN[NC_NR_ENVD_CUS_CMN]	O	0...FFH	0...255	0	[-]
Freeze frame FRF_CUS_CMN : environmental data contents					
ENVD_CONF_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN]	O	0...FFH	0...255	0	[-]
Freeze frame FRF_CUS_SET_CMN : environmental data contents					
ENVD_CONF_OBD[NC_NR_ENVD_OBD]	O	0...FFH	0...255	0	[-]
Freeze frame FRF_OBD : environmental data contents					
FAC_DIAGCP_MDL_ESTIM_ENVD	O/V	0...FFH	0...0.4652	0.00182	[-]
FAC_DIAGCP_MDL_ESTIM with lower resolution					
FAC_H_RNG_LAM_AD_ENVD_H	O/V	80...7FH	-50...49.609	0.389	[%]
FAC_H_RNG_LAM_AD High Byte					
FAC_L_RNG_LAM_AD_ENVD_H	O/V	80...7FH	-50...49.609	0.389	[%]
FAC_L_RNG_LAM_AD High Byte					
FAC_LAM_COR_1_ENVD	O/V	80...7FH	-50...49.61	0.3906	[%]
FAC_LAM_COR with lower resolution					
FAC_LAM_COR_H_1_ENVD	O/V	80...7FH	-100...99.219	0.78125	[%]
FAC_LAM_COR with higher resolution					
FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_H	O/V	0...7FH	0...31.74999	0.249	[-]
Mean value of normalised single sensor signal amplitude high byte					
FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_L	O/V	0...FFH	0...0.249	0.0009766	[-]
Mean value of normalised single sensor signal amplitude low byte					
FTL_MMV_DIF_NEG_DIAG_ENVD	O/V	0...FFH	0...100	0.39	[%]
FTL_MMV_DIF_NEG_DIAG with lower resolution					
FTL_MMV_DIF_POS_DIAG_ENVD	O/V	0...FFH	0...100	0.39	[%]
FTL_MMV_DIF_POS_DIAG with lower resolution					
FTL_MMV_ENVD	O/V	0...FFH	0...100	0.39	[%]

FTL_MMV with lower resolution					
FTL_MMV_REF_INI_ENVD	O/V	0...FFH	0...100	0.39	[%]
FTL_MMV_REF_INI with lower resolution					
ISAPWM_ISA_ENVD	O/V	0...FFH	0...99.61	0.3906	[%]
ISAPWM_ISA with lower resolution					
IVVTPWM_EX_1_ENVD	O/V	0...FFH	0...99.609	0.3906	[%]
IVVTPWM_EX with lower resolution					
IVVTPWM_IN_1_ENVD	O/V	0...FFH	0...99.609	0.3906	[%]
IVVTPWM_IN with lower resolution					
LAMB_DELTA_I_LAM_ADJ_1_ENVD_H	O/V	F8...8H	0.125...0.125	0.015625	[-]
I share from trim control high byte					
LAMB_DELTA_LAM_ADJ_1_ENVD_H	O/V	F8...8H	0.125...0.125	0.015625	[-]
total output (P share + I share) from trim control high byte					
LAMB_LS_UP_1_ENVD_H	O/V	0...7FH	0...31.74999	0.249	[-]
Lambda signal value of the WRAF sensor high byte					
LAMB_LS_UP_1_ENVD_L	O/V	0...FFH	0...0.249	0.0009766	[-]
Lambda signal value of the WRAF sensor low byte					
LAMB_SP_1_ENVD_H	O/V	0...7FH	0...31.74999	0.249	[-]
lambda setpoint high byte					
LAMB_SP_1_ENVD_L	O/V	0...FFH	0...0.249	0.0009766	[-]
lambda setpoint low byte					
MAF_KGH_ENVD	O/V	0...FFH	0...2040	8	[kg/h]
MAF_KGH with lower resolution					
MAF_MES_ENVD	O/V	0...FFH	0...1389	5.447	[mg/stk]
MAF_MES with lower resolution					
MAF_SP_ENVD	O/V	0...FFH	0...1389	5.447	[mg/stk]
MAF_SP with lower resolution					
MAP_ENVD	O/V	0...FFH	0...5412.856	21.2269	[hPa]
MAP with lower resolution					
MAP_MES_BAS_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
MAP_MES_BAS with lower resolution					
MAP_MES_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
MAP_MES with lower resolution					
MFF_ADD_LAM_AD_ENVD_L	O/V	80...7FH	-2.713...2.692	0.021195	[mg/stk]
MFF_ADD_LAM_AD Low Byte					
MTCPWM_ENVD	O/V	80...7FH	-100...99.219	0.78125	[%]
MTCPWM with lower resolution					
MTCPWM_MMV_ENVD	O/V	0...FFH	0...99.609	0.3906	[%]
MTCPWM_MMV with lower resolution					
N_MAX_SCDN_EQU_MIS_I_ENVD	O/V	0...FFH	0...8160	32	[rpm]
N_MAX_SCDN_EQU_MIS_i with lower resolution					
N_MIN_SCDN_EQU_MIS_I_ENVD	O/V	0...FFH	0...8160	32	[rpm]
N_MIN_SCDN_EQU_MIS_i with lower resolution					
N_SP_IS_ENVD	O/V	0...FFH	0...8160	32	[rpm]
N_SP_IS with lower resolution					
NL_2_ENVD	O/V	0...FFH	0...4.9805	0.019531	[V]
NL_2 with lower resolution					
NL_3_ENVD	O/V	0...FFH	0...4.9805	0.019531	[V]
NL_3 with lower resolution					
PORT_AV_ENVD	O/V	0...FFH	-100...99.219	0.78125	[°PORT]
Actual port flap position					
PORTPWM_CTL_ENVD	O/V	0...FFH	-100...99.219	0.78125	[%]
Pulse width modulation directly after port position controller					
PORTPWM_ENVD	O/V	0...FFH	-100...99.219	0.78125	[%]
Pulse width modulation for Port Flap electric motor energization					
PRS_AIC_DOWN_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
Pressure downstream the air cleaner					
PUT_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
Pressure upstream throttle					
PUT_SP_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
pressure before throttle set point (prs_up_thr-control)					

PUT_WG_OPEN_ENVD	O/V	0...FFH	0...5434	21.309	[hPa]
Pressure upstream throttle when WG opening requested with lower resolution					
PV_AV_SEL_ENVD	O/V	0...FFH	0...99.609	0.3906	[%]
PV_AV_SEL with lower resolution					
PWM_WG_ENVD	O/V	0...FFH	0...99.609	0.3906	[%]
Pulse width modulation for Port Flap electric motor energization with lower resolution					
R_IT_LS_DOWN_ENVD_H	O/V	0...FFH	0...65280	256	[Ohm]
R_IT_LS_DOWN high Byte					
R_IT_LS_DOWN_ENVD_L	O/V	0...FFH	0...255	1	[Ohm]
R_IT_LS_DOWN low Byte					
R_IT_LS_UP_ENVD_H	O/V	0...FFH	0...65280	256	[Ohm]
R_IT_LS_UP high Byte					
R_IT_LS_UP_ENVD_L	O/V	0...FFH	0...255	1	[Ohm]
R_IT_LS_UP low Byte					
SEG_AD_MMV_ER_X_ENVD	O/V	80...7FH	-7.81...7.752	0.061	[‰]
SEG_AD_MMV_ER with lower resolution					
T_AST_ENVD	O/V	0...FFH	0...6528	25.6	[s]
T_AST with lower resolution					
T_DLY_I_AD_LAM_ADJ_1_ENVD	O/V	80...7FH	-640...635	5	[ms]
T_DLY_I_AD_LAM_ADJ					
T_ES_DIAG_ENVD_H	O/V	0...FFH	0...65280	256	[s]
Engine off timer high byte					
T_ES_DIAG_ENVD_L	O/V	0...FFH	0...255	1	[s]
Engine off timer low byte					
TEG_DYN_LS_DOWN_1_ENVD_H	O/V	0...7FH	-273.15...1758.85	16	[°C]
Exhaust gas temperatures at the lambda sensor downstream catalyst with lower resolution					
TEG_DYN_LS_UP_1_ENVD_H	O/V	0...7FH	-273.15...1758.85	16	[°C]
Exhaust gas temperatures at the lambda sensor upstream catalyst high byte					
TEG_DYN_LS_UP_1_ENVD_L	O/V	0...FFH	0...16	0.06274	[°C]
Exhaust gas temperatures at the lambda sensor upstream catalyst high byte					
TEMP_CAT_DYN_MDL_1_ENVD_H	O/V	0...7FH	-273.15...1758.85	16	[°C]
Modelled catalyst temperature under dynamic conditions with lower resolution					
TEMP_CAT_ENVD	O/V	0...7FH	-273.15...1758.85	16	[°C]
Catalyst temperature with lower resolution					
TI_1_X_ENVD	O/V	0...FFH	0...261.12	1.024	[ms]
TI_1_i with lower resolution					
TI_1_X_ENVD_H	O/V	0...FFH	0...261.112	1.02	[ms]
TI_1_x High Byte					
TI_1_X_ENVD_L	O/V	0...FFH	0...1.02	0.004	[ms]
TI_1_x Low Byte					
TI_2_MES_X_ENVD_H	O/V	0...FFH	0...261.112	1.02	[ms]
TI_2_MES_x High Byte					
TPS_AV_1_ENVD	O/V	0...FFH	0...119.5	0.4686	[°]
TPS_AV_1 with lower resolution					
TPS_AV_2_ENVD	O/V	0...FFH	0...119.5	0.4686	[°]
TPS_AV_2 with lower resolution					
TPS_AV_ENVD	O/V	0...FFH	0...119.5	0.4686	[°]
TPS_AV_with lower resolution					
TPS_AV_ENVD_H	O/V	0...3FH	0...117.64	1.8673	[V]
TPS_AV High Byte					
TPS_LIH_ENVD	O/V	0...FFH	0...119.5	0.4686	[°]
TPS_LIH with lower resolution					
TPS_SP_ENVD	O/V	0...FFH	0...119.5	0.4686	[°]
TPS_SP with lower resolution					
TQI_TCHA_PROT_ENVD	O/V	0...FFH	0...1023.97	4.015568	[Nm]
Indicated torque for turbo charger protection as an interface to minimum / maximum torque request selection with lower resolution					

TTIP_MES_LS_UP_1_ENVD_H	O/V	80...7FH	-2048...2032	16	[°C]
Oxygen sensor ceramic temperature high byte					
V_ACP_MES_ENVD_H	O/V	0...3H	0...3.733	1.245	[V]
V_ACP_MES high Byte					
V_ACP_MES_ENVD_L	O/V	0...FFH	0...1.245	0.0048	[V]
V_ACP_MES low Byte					
V_DTP_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_DTP with lower resolution					
V_FTL_SUB_DIF_NEG_DIAG_ENVD	O/V	0...FFH	-4.985...0	0.0195	[V]
Sub V_FTL negative difference from reference					
V_FTL_SUB_DIF_POS_DIAG_ENVD	O/V	0...FFH	0...4.985	0.0195	[V]
Sub V_FTL positive difference from reference					
V_FTL_SUB_ENVD	O/V	0...FFH	0...4.985	0.0195	[V]
Sub V_FTL voltage for Diagnosis					
V_FTL_SUB_MMV_ENVD	O/V	0...FFH	0...4.985	0.0195	[V]
Sub V_FTL_MMV voltage for Diagnosis					
V_FTL_SUB_REF_INI_ENVD	O/V	0...FFH	0...4.985	0.0195	[V]
Sub V_FTL reference voltage for Diagnosis					
V_PORT_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
Port flap potentiometer signal					
V_PSP_MES_ENVD_H	O/V	0...3H	0...3.733	1.245	[V]
V_PSP_MES high Byte					
V_PSP_MES_ENVD_L	O/V	0...FFH	0...1.245	0.0048	[V]
V_PSP_MES low Byte					
V_PVS_1_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_PVS_1 with lower resolution					
V_PVS_2_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_PVS_1 with lower resolution					
V_TIA_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_TIA with lower resolution					
V_TPS_1_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_TPS_1 with lower resolution					
V_TPS_2_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
V_TPS_1 with lower resolution					
V_VIM_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
Voltage value for position feedback sensor at VIM with lower resolution					
VCC_PVS_1_DIAG_ENVD	O/V	0...FFH	0...9.9609	0.03906	[V]
Supply Voltage of PVS_1 channel for Diagnosis with lower resolution					
VCC_PVS_2_DIAG_ENVD	O/V	0...FFH	0...9.9609	0.03906	[V]
Supply Voltage of PVS_1 channel for Diagnosis with lower resolution					
VCC_SENS_SUB_DIAG_ENVD	O/V	0...FFH	0...9.9609	0.03906	[V]
Supply Voltage of Sensors for Diagnosis with lower resolution					
VCC_TPS_DIAG_ENVD	O/V	0...FFH	0...9.9609	0.03906	[V]
VCC_TPS_DIAG with lower resolution					
VLS_DIF_LAM_ADJ_1_ENVD_H	O/V	FC...3H	-5...3.75	1.25	[V]
difference between set point and actual downstream LS signal high byte					
VLS_DOWN_1_ENVD_H	O/V	0...3H	0...3.741	1.247	[V]
VLS_DOWN high Byte					
VLS_DOWN_1_ENVD_L	O/V	0...FFH	0...1.245	0.0049	[V]
VLS_DOWN low Byte					
VLS_DOWN_MMV_MIN_1_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VLS_DOWN_MMV_MIN with lower resolution					
VLS_OFS_LSL_1_ENVD_H	O/V	80...7FH	-5...4.96093	0.03906	[V]
Output signal offset of linear lambda sensor for Ip gain16 high byte					
VLS_UP_1_ENVD_H	O/V	0...3H	0...3.741	1.247	[V]
VLS_UP high Byte					
VLS_UP_1_ENVD_L	O/V	0...FFH	0...1.245	0.0049	[V]
VLS_UP low Byte					
VLS_UP_DIAG_1_ENVD_H	O/V	80...7FH	-160...158.75	1.25	[V]
Offset and exhaust pressure compensated VLS_UP[i] which is used for sensor diagnoses high byte					
VLS_UP_MMV_MAX_1_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]

VLS_UP_MMV_MAX with lower resolution					
VLS_UP_MMV_MIN_1_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VLS_UP_MMV_MIN with lower resolution					
WHEEL_GRD_MMV_ENVD_H	O/V	0...FFH	0...999.85	3.921	[‰]
WHEEL_GRD_MMV high byte					
WHEEL_GRD_MMV_ENVD_L	O/V	0...FFH	0...3.921	0.01537	[‰]
WHEEL_GRD_MMV low byte					
VP_MAP_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VP_MAP with lower resolution					
VP_PUT_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VP_PUT with lower resolution					
VP_AMP_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VP_AMP with lower resolution					
VP_TAM_ENVD	O/V	0...FFH	0...4.9805	0.01953	[V]
VP_TAM with lower resolution					

Input data:

N_32	CP_STATE	DTP_DIF_ACT	NC_NR_ENVD_CUS_SET_SPC
MAF	V_DTP	NC_NR_FRF_SET	NC_NR_ENVD_CUS_SET_CMN
TCO	MAP	NC_NR_ENVD_OBD	NC_NR_ENVD_CUS_CMN
LV_LSCL_i		R_IT_LS_UP[NC_CBK_EX_NR]	MFF_AD_FAC_MMV_i
TIA	IGA_AV_MV	NC_ID_ENVD_FAC	CAM_AV_IVVT_IN[NC_NR_CBK_IVVT]
MAF_HB	CAM_IN[NC_NR_CBK_IVVT]	CAM_SP_IVVT_IN	CAM_DIF_INT_DIAG_IVVT_IN_i
CPPWM	STATE_IVVT	DTP_DIF_COR	FAC_DIAGCP_MDL_ESTIM
DTP	TCU_OBD	MAF_KGH	FAC_DIAGCP_VOL
MAF_SP	DIST	IVVTPWM_IN_i	KNKS_CMD_GAIN_AD_x
LOAD_CLC	FAC_DIAM_DIAGCP	TIA_MES[NC_SENS_NR_TIA]	CTR_EDGE_CAM_IN_i
MAF_MES	NL_x	LSHPWM_DOWN[NC_CBK_EX_NR]	N_MAX_SCDN_EQU[NC_NR_WIN_SCDN]
TCO_ST	N_SP_IS	FAC_LAM_COR[NC_CBK_EX_NR]	EFF_CAT_DIAG[NC_CBK_EX_NR]
TAM	STATE_LS[NC_CBK_EX_NR]	TI_LAM_COR_i	TCO_MES_DIF_DIAG_TCO_STUCK
T_AST		VLS_DOWN_MMV_MIN[NC_CBK_EX_NR]	T_DLY_I_AD_LAM_ADJ
TCO_MES	TOIL_MES	LSHPWM_UP[NC_CBK_EX_NR]	TIA_MES_ST[NC_SENS_NR_TIA]
DTP_DIF	TI_1_x	TOIL_MDL_DIF_DIAG	N_MIN_SCDN_EQU[NC_NR_WIN_SCDN]
TIA_ST	TPS_LIH	LV_ERR_PVS_RATIO	TOIL_MES_DIF_DIAG
TPS	TPS_AV_1	TPS_AV_2	VCC_TPS_DIAG
	V_TOIL	VP_TIA[NC_SENS_NR_TIA]	RATIO_CYC_AFR[NC_CBK_EX_NR]
VB	VLS_UP[NC_CBK_EX_NR]	VLS_UP_MMV_MAX[NC_CBK_EX_NR]	NC_ENVD_CUS_CMN_UPD
VS	ISAPWM_ISA	ANS_ERR_COM_IMOB	TCO_SUB_DIF_DIAG_TCO_STUCK
OBD_TCO	OBD_N	OBD_TPS_AV_1	WHEEL_GRD_MMV
OBD_TIA	OBD_VB	OBD_IGA_IGC	R_IT_LS_DOWN[NC_CBK_EX_NR]

OBD_MAF	OBD_REL_CT	LV_ERR_VCC_PVS_1	OBD_LAM_COR_i
V_FTL	T_AST_OBD	OBD_CPPWM	OBD_REL_TPS_SP
TRT	V_PVS_1	PSN_AD_CAM_IN[NC_NR_CAM_CBK]	V_MIN_MWSS_DIAG
MAP	MTCPWM_MMV	OBD_TPS_AV_2	VCC_PVS_1_DIAG
MTCPWM	TPS_SP	TPS_MES	GEN_LOAD_MMV
V_PVS_2	TPS_AV	LV_ERR_VCC_PVS_2	FTL_MMV_DIF_NEG_DIAG
V_TPS_2	ENG_STATE	PV_AV_SEL	LV_VLS_DOWN_AFR[NC_CBK_EX_NR]
LV_BTS	ENVD_1_MON	V_MAX_MWSS_DIAG	CTR_CYCNR_SWT_LS_DOWN[NC_CBK_EX_NR]
TI_2_MES_x	LV_PVS_H_R_2	LV_RLY_ACCOUT_CTRL	LV_VLS_DOWN_AFR[NC_CBK_EX_NR]
V_TPS_1	OBD_LAMB_SP	FTL_MMV_DIF_POS_DIAG	FAC_H_RNG_LAM_AD[NC_CBK_EX_NR]
VB_MMV	VLS_DOWN[NC_CBK_EX_NR]	FTL_MMV_REF_INI	FAC_L_RNG_LAM_AD[NC_CBK_EX_NR]
STATE_CRU	LV_PVS_H_R_1	VCC_PVS_2_DIAG	SEG_AD_MMV_ER[NC_CYL_NR]
V_PSP_MES	GEN_LOAD	OBD_LAM_AD_i	MFF_ADD_LAM_AD[NC_CBK_EX_NR]
V_ACP_MES	CRU_BAS	CRU_SWI_DRIV_STATE	CONF_DIAGCP_VOL
TOIL	ENVD_2_MON	LV_RLY_ACCOUT	RATIO_CYC_AFL[NC_CBK_EX_NR]
LV_BLS	ENVD_3_MON	ECU_DIAG_SF_HB	ECU_DIAG_SF_LB
FTL_MMV	ENVD_0_MON	TIA_MES_DIF_DIAG_INTM[NC_SENS_NR_TIA]	TIA_MES_DIF_DIAG_DYN[NC_SENS_NR_TIA]
TOIL_MDL	LV_ACCIN	ERR_INTM_DIAG_INST_ACT	CPU_LOAD_MAX
FTL_MES	TECU_MAX	OBD_FTL	OBD_PV_1
OBD_PV_2	OBD_AMP	LOAD_ABSV	OBD_MAP
OBD_TAM	CAM_EX[NC_NR_CBK_IVVT]	CAM_AV_IVVT_EX[NC_NR_CBK_IVVT]	CAM_DIF_INT_DIAG_IVVT_EX_i
MAP_MES_BAS	IVVTPWM_EX_i	CAM_SP_IVVT_EX	PSN_AD_CAM_EX[NC_NR_CAM_CBK]
V_PORT	PORT_AV		PORTPWM_CTL
LV_VIM_SP		V_FTL_SUB_MMV	V_FTL_SUB_DIF_NEG_DIAG
V_FTL_SUB	TECU	V_FTL_SUB_REF_INI	V_FTL_SUB_DIF_POS_DIAG
LAMB_SP[NC_CBK_EX_NR]	VLS_OFS_LSL[NC_CBK_EX_NR]	FAC_MV_DIAG_DYN_LSL_UP[NC_CBK_EX_NR]	STATE_ERR_EL_LSL_UP[NC_CBK_EX_NR]
TTIP_MES_LS_UP[NC_CBK_EX_NR]	ERR_SYM_OC_LSL_UP[NC_CBK_EX_NR]	STATE_SYM_DIAG_PUC_LSL_UP[NC_CBK_EX_NR]	STATE_SYM_OBD_LSL_LSH_UP[NC_CBK_EX_NR]
TEG_DYN_LS_UP[NC_CBK_EX_NR]	VLS_UP_DIAG[NC_CBK_EX_NR]	ERR_DIAG_CTL_LSL_UP[NC_CBK_EX_NR]	CTR_ERR_LSL_IF_SPI_WR[NC_CBK_EX_NR]
LAMB_LS_UP[NC_CBK_EX_NR]	LAMB_DELTA_LAM_ADJ[NC_CBK_EX_NR]	LAMB_DELTA_I_LAM_ADJ[NC_CBK_EX_NR]	VLS_DIF_LAM_ADJ[NC_CBK_EX_NR]
PORTPWM	VCC_SENS_SUB_DIAG	ERR_SYM_OFS_LSL_UP[NC_CBK_EX_NR]	ERR_SYM_LSL_UP_IF[NC_CBK_EX_NR]
T_ES_DIAG	V_VIM	PRS_AIC_DOWN	LV_CMD_RCL_OPEN
PWM_WG	PUT	TQI_TCHA_PROT	PUT_WG_OPEN
PUT_SP	LV_SURGE	OBD_AD_LAM_ADJ	TEG_DYN_LS_DOWN[NC_CBK_EX_NR]
TEMP_CAT[NC_NR_CAT_MDL]	OBD_LAM_ADJ	TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	MAP_MES
LOAD_MAX_SCDN_EQU[NC_NR_WIN_SCDN]	VP_TAM	T_ES_DIF	VP_MAP

LOAD_MIN_SCDN_EQU[NC_NR_WIN_SCDN]	VP_AMP	VP_PUT	

B.55.1 Configuration of size and content of Environmental data (freeze frame)

FUNCTION DESCRIPTION:

General information:

Description:

This file defines sizes and contents of Environmental data (freeze frame) catched when a failure occurs.

All data in italic are comments.

Freeze frame part	Characteristics	Value
FRF_OBD	Number of data (in bytes)	NC_NR_ENVD_OBD = 33
	Contents	ENVD_CONF_OBD[1] = LOAD_CLC **(PID 04h)** *Calculated load value (%)*
		ENVD_CONF_OBD[2] = OBD_TCO **(PID 05h)** *Engine coolant temperature*
		ENVD_CONF_OBD[3] = OBD_N (high byte) **(PID 0Ch)** *Engine rpm (high byte)*
		ENVD_CONF_OBD[4] = OBD_N (low byte) **(PID 0Ch)** *Engine rpm (low byte)*
		ENVD_CONF_OBD[5] = OBD_TPS_AV_1 **(PID 11h)** *Absolute throttle position sensor 1*
		ENVD_CONF_OBD[6] = OBD_TPS_AV_2 **(PID 47h)** *Absolute throttle position sensor 2*
		ENVD_CONF_OBD[7] = VS **(PID 0Dh)** *Vehicle speed*
		ENVD_CONF_OBD[8] = STATE_LS_1 **(PID 03h)** *Fuel system status bank 1*
		ENVD_CONF_OBD[9] = OBD_LAM_COR_1 **(PID 06h)** *Short term fuel trim bank 1*
		ENVD_CONF_OBD[10] = OBD_LAM_AD_1 **(PID 07h)** *Long term fuel trim bank 1*
		ENVD_CONF_OBD[11] = OBD_IGA_IGC **(PID 0Eh)** *Ignition timing advance cylinder 1*

		ENVD_CONF_OBD[12] = OBD_TIA **(PID 0Fh)** *Intake Air temperature*
		ENVD_CONF_OBD[13] = OBD_MAP **(PID 0Bh)** *Intake manifold absolute pressure*
		ENVD_CONF_OBD[14] = OBD_FTL **(PID 2Fh)** *Fuel level input*
		ENVD_CONF_OBD[15] = OBD_REL_CT **(PID 45h)** *Relative throttle position at closed throttle*
		ENVD_CONF_OBD[16] = OBD_VB (high byte) **(PID 42h)** *Engine control module voltage (high byte)*
		ENVD_CONF_OBD[17] = OBD_VB (low byte) **(PID 42h)** *Engine control module voltage (low byte)*
		ENVD_CONF_OBD[18] = OBD_LAMB_SP (high Byte) **(PID 44h)** *Commanded equivalence ratio (high byte)*
		ENVD_CONF_OBD[19] = OBD_LAMB_SP (low Byte) **(PID 44h)** *Commanded equivalence ratio (low byte)*
		ENVD_CONF_OBD[20] = T_AST_OBD (high byte) **(PID 1Fh)** *Time since engine start (high byte)*
		ENVD_CONF_OBD[21] = T_AST_OBD (low byte) **(PID 1Fh)** *Time since engine start (low byte)*
		ENVD_CONF_OBD[22] = OBD_CPPWM **(PID 2Eh)** *Commanded evaprative purge*
		ENVD_CONF_OBD[23] = OBD_PV_1 **(PID 49h)** *Accelerator pedal position (sensor 1)*
		ENVD_CONF_OBD[24] = OBD_REL_TPS_SP **(PID 4Ch)** *Commanded throttle actuator control*
		ENVD_CONF_OBD[25] = OBD_AMP **(PID 33h)** *Barometric pressure*
		ENVD_CONF_OBD[26] = LOAD_ABSV (high byte) **(PID 43h)** *Absolute load value (high byte)*
		ENVD_CONF_OBD[27] = LOAD_ABSV (low byte) **(PID 43h)** *Absolute load value (low byte)*
		ENVD_CONF_OBD[28] = OBD_PV_2 **(PID 4Ah)** *Accelerator pedal position (sensor 2)*
		ENVD_CONF_OBD[29] = OBD_TAM **(PID 46h)** *Ambient air temperature*

		ENVD_CONF_OBD[30] = OBD_MAF (high Byte) **(PID 10h)** *Air flow rate (high byte)*
		ENVD_CONF_OBD[31] = OBD_MAF (low Byte) **(PID 10h)** *Air flow rate (low byte)*
		ENVD_CONF_OBD[32] = OBD_LAM_ADJ **(PID 55h)** *Short term Secondary O2 Sensor fuel trim Bank 1*
		ENVD_CONF_OBD[33] = OBD_AD_LAM_ADJ **(PID 56h)** *Long term Secondary O2 Sensor fuel trim Bank 1*
Freeze frame part	**Characteristics**	**Value**
FRF_CUS_CMN	Number of data (in bytes)	NC_NR_ENVD_CUS_CMN = 0
	Contents	ENVD_CONF_CUS_CMN[1] =
		ENVD_CONF_CUS_CMN[2] =
		....
		ENVD_CONF_CUS_CMN[v] = (v = NC_NR_ENVD_CUS_CMN)
"Set of Freeze Frame"	Number of set	NC_NR_FRF_SET = 3
FRF_CUS_SET_CMN	Number of common data per set (in bytes)	NC_NR_ENVD_CUS_SET_CMN = 12
	Contents	ENVD_CONF_CUS_SET_CMN[1] =TRT (byte 1) *Total running time of EMS*
		ENVD_CONF_CUS_SET_CMN[2] =TRT (byte 2) *Total running time of EMS*
		ENVD_CONF_CUS_SET_CMN[3] =TRT (byte 3) *Total running time of EMS*
		ENVD_CONF_CUS_SET_CMN[4] =TRT (byte 4) *Total running time of EMS*
		ENVD_CONF_CUS_SET_CMN[5] =DIST (byte 1) *DIST(1) for mileage counter*
		ENVD_CONF_CUS_SET_CMN[6] =DIST (byte 2) *DIST (2) for mileage counter*
		ENVD_CONF_CUS_SET_CMN[7] =DIST (byte 3) *DIST (3) for mileage counter*
		ENVD_CONF_CUS_SET_CMN[8] =DIST (byte 4) *DIST (4) for mileage counter*

		ENVD_CONF_CUS_SET_CMN[9] =N_32 *Engine speed*
		ENVD_CONF_CUS_SET_CMN[10] = MAF_HB *Mass air flow (mg/Stk) (high byte)*
		ENVD_CONF_CUS_SET_CMN[11] = TCO *Coolant temperature (°C)*
		ENVD_CONF_CUS_SET_CMN[12] = VB *Battery Voltage*
		...
		ENVD_CONF_CUS_SET_CMN[w] = (w = NC_NR_ENVD_CUS_SET_CMN)
		NC_NR_ENVD_CUS_SET_SPC = 4
FRF_CUS_SET_SPC	Number of specific data per set (in bytes)	Define by calibration ID_ERR_ENVD_XX (see Environmental data module) Possible stored environmental data : NC_ID_ENVD_FAC [1... 236] =see Table below ... NC_ID_ENVD_FAC [255]
	Contents	

		NC_ID_ENVD_FAC [...]	**Comment**
[1]	=	CAM_AV_IVVT_IN	
[2]	=	CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD	
[3]	=	CAM_IN_1	
[4]	=	CAM_SP_IVVT_IN_1	
[5]	=	CP_STATE	
[6]	=	CPPWM_ENVD	
[7]	=	CTR_EDGE_CAM_IN_1	
[8]	=	DTP_DIF_COR_ENVD_H	
[9]	=	DTP_DIF_COR_ENVD_L	
[10]	=	DTP_DIF_ENVD_H	
[11]	=	DTP_DIF_ENVD_L	
[12]	=	DTP_ENVD	
[13]	=	EFF_CAT_DIAG 1	
[14]	=	FAC_DIAGCP_MDL_ESTIM_ENVD	
[15]	=	FAC_DIAGCP_VOL	
[16]	=	FAC_DIAM_DIAGCP	
[17]	=	FAC_LAM_COR_1_ENVD	
[18]	=	IVVTPWM_IN_1_ENVD	
[19]	=	KNKS_CMD_GAIN_AD_2	
[20]	=	LSHPWM_DOWN_1	
[21]	=	LSHPWM_UP_1	
[22]	=	MAF_HB	

[23]	=	MAF_KGH_ENVD	
[24]	=	MAF_MES_ENVD	
[25]	=	MAF_SP_ENVD	
[26]	=	N_32	
[27]	=	N_MAX_SCDN_EQU_MIS_0_ENVD	
[28]	=	N_MAX_SCDN_EQU_MIS_1_ENVD	
[29]	=	N_MAX_SCDN_EQU_MIS_2_ENVD	
[30]	=	N_MAX_SCDN_EQU_MIS_3_ENVD	
[31]	=	N_MIN_SCDN_EQU_MIS_0_ENVD	
[32]	=	N_MIN_SCDN_EQU_MIS_1_ENVD	
[33]	=	N_MIN_SCDN_EQU_MIS_2_ENVD	
[34]	=	N_MIN_SCDN_EQU_MIS_3_ENVD	
[35]	=	N_SP_IS_ENVD	
[36]	=	NL_2_ENVD	
[37]	=	NL_3_ENVD	
[38]	=	R_IT_LS_DOWN_1_ENVD_H	
[39]	=	R_IT_LS_DOWN_1_ENVD_L	
[40]	=	R_IT_LS_UP_1_ENVD_H	
[41]	=	R_IT_LS_UP_1_ENVD_L	
[42]	=	STATE_IVVT	
[43]	=	T_AST_ENVD	
[44]	=	T_DLY_I_AD_LAM_ADJ_1_ENVD	
[45]	=	TAM	
[46]	=	TCO	
[47]	=	TCO_MES	
[48]	=	TCO_MES_DIF_DIAG_TCO_STUCK	
[49]	=	TCO_SUB_DIF_DIAG_TCO_STUCK	
[50]	=	TCO_ST	
[51]	=	TCU_OBD	
[52]	=	PWM_WG_ENVD	(only if NC_CHRG_CONF = 1)
[53]	=	PUT_ENVD	(only if NC_CHRG_CONF = 1)
[54]	=	PRS_AIC_DOWN_ENVD	(only if NC_CHRG_CONF = 1)
[55]	=	LV_CMD_RCL_OPEN	(only if NC_CHRG_CONF = 1)
[56]	=	TIA	
[57]	=	TIA_MES	
[58]	=	TIA_ST	
[59]	=	TOIL_MDL_DIF_DIAG	
[60]	=	TOIL_MES	
[61]	=	TOIL_MES_DIF_DIAG	
[62]	=	TPS	
[63]	=	TPS_AV_1_ENVD	
[64]	=	TPS_AV_2_ENVD	
[65]	=	V_DTP_ENVD	
[66]	=	V_IGK	
[67]	=	V_TIA_ENVD	
[68]	=	V_TOIL	
[69]	=	VB	
[70]	=	VLS_DOWN_1_ENVD_L	
[71]	=	VLS_DOWN_MMV_MIN_1_ENVD	
[72]	=	VLS_UP_1_ENVD_L	
[73]	=	VLS_UP_MMV_MAX_1_ENVD	
[74]	=	VLS_UP_MMV_MIN_1_ENVD	

[75]	=	VS	
[76]	=	ISAPWM_ISA_ENVD	
[77]	=	ANS_ERR_COM_IMOB	
[78]	=	VCC_TPS_DIAG_ENVD	
[79]	=	V_TPS_1_ENVD	
[80]	=	V_TPS_2_ENVD	
[81]	=	MAP_ENVD	
[82]	=	MTCPWM_MMV_ENVD	
[83]	=	VB_MMV	
[84]	=	TPS_AV_ENVD	
[85]	=	MTCPWM_ENVD	
[86]	=	TPS_SP_ENVD	
[87]	=	TPS_AD_STEP	(only if NC_ETC_CONF = 1)
[88]	=	TPS_LIH_ENVD	
[89]	=	V_PVS_1_ENVD	
[90]	=	V_PVS_2_ENVD	
[91]	=	LV_ERR_VCC_PVS_1	(only if NC_ETC_CONF = 1)
[92]	=	LV_ERR_VCC_PVS_2	(only if NC_ETC_CONF = 1)
[93]	=	LV_PVS_H_R_1	(only if NC_ETC_CONF = 1)
[94]	=	LV_PVS_H_R_2	(only if NC_ETC_CONF = 1)
[95]	=	LV_ERR_PVS_RATIO	(only if NC_ETC_CONF = 1)
[96]	=	PV_AV_SEL_ENVD	
[97]	=	LV_BLS	(only if NC_ETC_CONF = 1)
[98]	=	LV_BTS	(only if NC_ETC_CONF = 1)
[99]	=	RATIO_CYC_AFL	(only if NC_STATE_LSL_UP_IF=0)
[100]	=	RATIO_CYC_AFR	(only if NC_STATE_LSL_UP_IF=0)
[101]	=	TOIL_MDL	
[102]	=	KNKS_CMD_GAIN_AD_3	
[103]	=	TPS_AV_ENVD_H	
[104]	=	ENG_STATE	
[105]	=	TI_1_0_ENVD_H	
[106]	=	TI_1_1_ENVD_H	
[107]	=	TI_1_2_ENVD_H	
[108]	=	TI_1_3_ENVD_H	
[109]	=	TI_1_0_ENVD_L	
[110]	=	TI_1_1_ENVD_L	
[111]	=	TI_1_2_ENVD_L	
[112]	=	TI_1_3_ENVD_L	
[113]	=	TI_2_MES_0_ENVD_H	
[114]	=	TI_2_MES_1_ENVD_H	
[115]	=	TI_2_MES_2_ENVD_H	
[116]	=	TI_2_MES_3_ENVD_H	
[117]	=	TIA_MES_ST	
[118]	=	CTR_CYCNR_SWT_LS_DOWN	
[119]	=	LV_VLS_DOWN_AFR	(only if NC_STATE_LSL_UP_IF=0)
[120]	=	LV_VLS_DOWN_AFL	(only if NC_STATE_LSL_UP_IF=0)
[121]	=	FAC_H_RNG_LAM_AD_ENVD_H	
[122]	=	FAC_L_RNG_LAM_AD_ENVD_H	
[123]	=	MFF_ADD_LAM_AD_ENVD_L	
[124]	=	DTP_DIF_ACT_ENVD	
[125]	=	PSN_AD_CAM_IN	
[126]	=	V_MIN_MWSS_DIAG	

[127]	=	V_MAX_MWSS_DIAG	
[128]	=	LV_RLY_ACCOUT_CTRL	
[129]	=	LV_PLAUS_TIA_DEC	
[130]	=	VCC_PVS_1_DIAG_ENVD	
[131]	=	VCC_PVS_2_DIAG_ENVD	
[132]	=	VLS_UP_1_ENVD_H	
[133]	=	VLS_DOWN_1_ENVD_H	
[134]	=	SEG_AD_MMV_ER_1_ENVD	
[135]	=	SEG_AD_MMV_ER_3_ENVD	
[136]	=	V_PSP_MES_ENVD_H	
[137]	=	V_PSP_MES_ENVD_L	
[138]	=	GEN_LOAD	
[139]	=	GEN_LOAD_MMV	
[140]	=	LV_ACCIN	
[141]	=	LV_RLY_ACCOUT	
[142]	=	V_ACP_MES_ENVD_H	
[143]	=	V_ACP_MES_ENVD_L	
[144]	=	CRU_BAS_ENVD_H	
[145]	=	CRU_BAS_ENVD_L	
[146]	=	CRU_SWI_DRIV_STATE	(only if NC_ETC_CONF = 1)
[147]	=	STATE_CRU	(only if NC_ETC_CONF = 1)
[148]	=	TOIL	
[149]	=	ENVD_0_MON	
[150]	=	ENVD_1_MON	
[151]	=	ENVD_2_MON	
[152]	=	ENVD_3_MON	
[153]	=	DIAG_INST_ENVD	
[154]	=	ECU_DIAG_SF_HB	
[155]	=	ECU_DIAG_SF_LB	
[156]	=	TIA_MES_DIF_DIAG_INTM	
[157]	=	TIA_MES_DIF_DIAG_DYN	
[158]	=	V_FTL	
[159]	=	FTL_MMV_ENVD	
[160]	=	FTL_MMV_REF_INI_ENVD	
[161]	=	FTL_MMV_DIF_POS_DIAG_ENVD	
[162]	=	FTL_MMV_DIF_NEG_DIAG _ENVD	
[163]	=	WHEEL_GRD_MMV_ENVD_H	
[164]	=	WHEEL_GRD_MMV_ENVD_L	
[165]	=	FTL_MES	
[166]	=	CONF_DIAGCP_VOL	
[167]	=	ERR_INTM_DIAG_INST_ACT	
[168]	=	CPU_LOAD_MAX_ENVD_H	
[169]	=	CPU_LOAD_MAX_ENVD_L	
[170]	=	TECU_MAX	
[171]	=	CAM_AV_IVVT_EX	
[172]	=	CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD	
[173]	=	CAM_EX_1	
[174]	=	CAM_SP_IVVT_EX_1	
[175]	=	IVVTPWM_EX_1_ENVD	
[176]	=	PSN_AD_CAM_EX	
[177]	=	MAP_MES_BAS_ENVD	
[178]	=	V_PORT_ENVD	

[179]	=	PORT_AV_ENVD	
[180]		MAP_MES_ENVD	
[181]		V_VIM_ENVD	
[182]	=	TEMP_CAT_DYN_MDL_1_ENVD_H	
[183]	=	PORTPWM_CTL_ENVD	
[184]	=	V_FTL_SUB_MMV_ENVD	
[185]	=	V_FTL_SUB_ENVD	
[186]	=	V_FTL_SUB_REF_INI_ENVD	
[187]	=	V_FTL_SUB_DIF_POS_DIAG_ENVD	
[188]	=	V_FTL_SUB_DIF_NEG_DIAG_ENVD	
[189]	=	TTIP_MES_LS_UP_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[190]	=	STATE_ERR_EL_LSL_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[191]	=	STATE_SYM_OBD_LSL_LSH_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[192]	=	STATE_SYM_DIAG_PUC_LSL_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[193]	=	ERR_SYM_OC_LSL_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[194]	=	ERR_DIAG_CTL_LSL_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[195]	=	CTR_ERR_LSL_IF_SPI_WR[1]	(only if NC_STATE_LSL_UP_IF=1)
[196]	=	VLS_OFS_LSL_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[197]	=	VLS_UP_DIAG_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[198]	=	TECU	
[199]	=	FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[200]	=	FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_L	(only if NC_STATE_LSL_UP_IF=1)
[201]	=	LAMB_DELTA_I_LAM_ADJ_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[202]	=	LAMB_DELTA_LAM_ADJ_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[203]	=	VLS_DIF_LAM_ADJ_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[204]	=	LAMB_SP_1_ENVD_H	
[205]	=	LAMB_SP_1_ENVD_L	
[206]	=	TEG_DYN_LS_UP_1_ENVD_H	
[207]	=	TEG_DYN_LS_UP_1_ENVD_L	
[208]	=	LAMB_LS_UP_1_ENVD_H	(only if NC_STATE_LSL_UP_IF=1)
[209]	=	LAMB_LS_UP_1_ENVD_L	(only if NC_STATE_LSL_UP_IF=1)
[210]	=	ERR_SYM_LSL_UP_IF[1]	(only if NC_STATE_LSL_UP_IF=1)
[211]	=	ERR_SYM_OFS_LSL_UP[1]	(only if NC_STATE_LSL_UP_IF=1)
[212]	=	FAC_LAM_COR_H_1_ENVD	(only if NC_STATE_LSL_UP_IF=1)
[213]	=	PORTPWM_ENVD	
[214]	=	VCC_SENS_SUB_DIAG_ENVD	
[215]	=	T_ES_DIAG_ENVD_H	
[216]	=	T_ES_DIAG_ENVD_L	
[217]	=	PUT_SP_ENVD	(only if NC_CHRG_CONF = 1)
[218]	=	LV_SURGE	(only if NC_CHRG_CONF = 1)
[219]	=	TQI_TCHA_PROT_ENVD	(only if NC_CHRG_CONF = 1)
[220]	=	PUT_WG_OPEN_ENVD	(only if NC_CHRG_CONF = 1)
[221]	=	TEMP_CAT_ENVD	
[222]	=	TEG_DYN_LS_DOWN_1_ENVD_H	
[223]	=	LV_VIM_SP	
[224]	=	LOAD_MAX_SCDN_EQU[1]	
[225]	=	LOAD_MAX_SCDN_EQU[2]	
[226]	=	LOAD_MAX_SCDN_EQU[3]	
[227]	=	LOAD_MAX_SCDN_EQU[4]	
[228]	=	LOAD_MIN_SCDN_EQU[1]	
[229]	=	LOAD_MIN_SCDN_EQU[2]	
[230]	=	LOAD_MIN_SCDN_EQU[3]	

[231]	=	LOAD_MIN_SCDN_EQU[4]	
[232]	=	VP_MAP_ENVD	
[233]	=	VP_PUT_ENVD	
[234]	=	VP_TAM_ENVD	(only if NC_CHRG_CONF = 1)
[235]	=	VP_AMP_ENVD	
[236]	=	T_ES_DIF	

Formula section:

CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD = CAM_DIF_INT_DIAG_IVVT_IN_1 with resolution 0.786 and range 0...200.54 °CRK

CAM_DIF_INT_DIAG_IVVT_EX_1_ENVD = CAM_DIF_INT_DIAG_IVVT_EX_1 with resolution 0.786 and range 0...200.54 °CRK

FAC_LAM_COR_1_ENVD = FAC_LAM_COR with resolution 0.3906 and range -50...49.6

FAC_LAM_COR_H_1_ENVD = FAC_LAM_COR[1] with resolution 0.78125 and range -100...99.219

CPPWM_ENVD = CPPWM with resolution 0.3906 and range 0...99.6 %

{Remark: When the EVAC Aggregate (e.g.6H1) is used CPPWM is replaced by CPPWM_CPS which has resolution 0...FFH, 0...99.6% (same as CPPWM_ENVD).}

MAF_KGH_ENVD = MAF_KGH with resolution 8.0 and range 0...2040.0 kg/h

V_DTP_ENVD = V_DTP with resolution 0.0195 and range 0...4,980 V

DTP_DIF_ENVD_L = DTP_DIF with resolution 0.00125 and range -0.16...0.159 hPa

DTP_DIF_ENVD_H = DTP_DIF with resolution 0.32 and range -40.96...40.64 hPa

DTP_DIF_COR_ENVD_L = DTP_DIF_COR with resolution 0.00125 and range -0.16...0.159 hPa

DTP_DIF_COR_ENVD_H = DTP_DIF_COR with resolution 0.32 and range -40.96...40.64 hPa

R_IT_LS_DOWN_ENVD_L = R_IT_LS_DOWN with resolution 1 and range 0...255 Ohm

R_IT_LS_DOWN_ENVD_H = R_IT_LS_DOWN with resolution 256 and range 0...65280 Ohm

R_IT_LS_UP_ENVD_L = R_IT_LS_UP with resolution 1 and range 0...255 Ohm

R_IT_LS_UP_ENVD_H = R_IT_LS_UP with resolution 256 and range 0...65280 Ohm

VLS_DOWN_1_ENVD_L = VLS_DOWN with resolution 0.0049 and range 0...1.245 V

VLS_UP_1_ENVD_L = VLS_UP with resolution 0.0049 and range 0...1.245 V

FAC_DIAGCP_MDL_ESTIM_ENVD = FAC_DIAGCP_MDL_ESTIM with resolution 0.00182 and range 0...0.4652

DTP_ENVD = DTP with resolution 0.32 and range -40.96...40.64 hPa

IVVTPWM_IN_1_ENVD = IVVTPWM_IN with resolution 0.3906 and range 0...99.609 %

IVVTPWM_EX_1_ENVD = IVVTPWM_EX with resolution 0.3906 and range 0...99.609 %

MAF_MES_ENVD = MAF_MES with resolution 5.447 and range 0...1389 mg/stk

MAF_SP_ENVD = MAF_SP with resolution 5.447 and range 0...1389 mg/stk

N_MAX_SCDN_EQU_MIS_0_ENVD = N_MAX_SCDN_EQU[1] with resolution 32 and range 0...8160 rpm

N_MAX_SCDN_EQU_MIS_1_ENVD = N_MAX_SCDN_EQU[2] with resolution 32 and range 0...8160 rpm

N_MAX_SCDN_EQU_MIS_2_ENVD = N_MAX_SCDN_EQU[3] with resolution 32 and range 0...8160 rpm

N_MAX_SCDN_EQU_MIS_3_ENVD = N_MAX_SCDN_EQU[4] with resolution 32 and range 0...8160 rpm

N_MIN_SCDN_EQU_MIS_0_ENVD = N_MIN_SCDN_EQU[1] with resolution 32 and range 0...8160 rpm

N_MIN_SCDN_EQU_MIS_1_ENVD = N_MIN_SCDN_EQU[2] with resolution 32 and range 0...8160 rpm

N_MIN_SCDN_EQU_MIS_2_ENVD = N_MIN_SCDN_EQU[3] with resolution 32 and range 0...8160 rpm

N_MIN_SCDN_EQU_MIS_3_ENVD = N_MIN_SCDN_EQU[4] with resolution 32 and range 0...8160 rpm

N_SP_IS_ENVD = N_SP_IS with resolution 32 and range 0...8160 rpm

NL_2_ENVD = NL_2 with resolution 0.019531 and range 0...4.9805 V

NL_3_ENVD = NL_3 with resolution 0.019531 and range 0...4.9805 V

T_DLY_I_AD_LAM_ADJ_1_ENVD = T_DLY_I_AD_LAM_ADJ with resolution 5 and range -640...635 ms

TPS_AV_1_ENVD = TPS_AV_1 (for NC_ETC_CONF=1) with resolution 0.4686 and range 0...119.5 °
= 0 (for NC_ETC_CONF=0)

TPS_AV_2_ENVD = TPS_AV_2 (for NC_ETC_CONF=1) with resolution 0.4686 and range 0...119.5 °
= 0 (for NC_ETC_CONF=0)

VLS_DOWN_MMV_MIN_1_ENVD = VLS_DOWN_MMV_MIN with resolution 0.01953 and range 0...4.9805 V

VLS_UP_MMV_MAX_1_ENVD = VLS_UP_MMV_MAX (for NC_STATE_LSL_UP_IF=0) with resolution 0.01953 and range 0...4.9805 V
= 0 (for NC_STATE_LSL_UP_IF=1)

VLS_UP_MMV_MIN_1_ENVD = VLS_UP_MMV_MIN (for NC_STATE_LSL_UP_IF=0) with resolution 0.01953 and range 0...4.9805 V
= 0 (for NC_STATE_LSL_UP_IF=1)

V_TIA_ENVD = VP_TIA with resolution 0.01953 and range 0...4.9805 V

T_AST_ENVD = T_AST with resolution 25.6 and range 0...6528 s

TI_1_x_ENVD = TI_1_x with resolution 1.024 and range 0...261.12 ms

ISAPWM_ISA_ENVD = ISAPWM_ISA (for NC_ETC_CONF=0)with resolution 0.390 and range 0...99.6%
= 0 (for NC_ETC_CONF=1)

VCC_TPS_DIAG_ENVD = VCC_TPS_DIAG with resolution 0.03906 and range 0...9.9609 V

V_TPS_1_ENVD = V_TPS_1 (for NC_ETC_CONF=1) with resolution 0.01953 and range 0...4.9805 V
= TPS_MES (for NC_ETC_CONF=0)

V_TPS_2_ENVD = V_TPS_2 (for NC_ETC_CONF=1) with resolution 0.01953 and range 0...4.9805 V
= 0 (for NC_ETC_CONF=0)

MAP_ENVD = MAP with resolution 21.2269 and range 0...5412,8 hPa

MTCPWM_MMV_ENVD = MTCPWM_MMV (for NC_ETC_CONF=1) with resolution 0.3906 and range 0...99.609 %
= 0 (for NC_ETC_CONF=0)

TPS_AV_ENVD = TPS_AV (for NC_ETC_CONF=1) with resolution 0.4686 and range 0...119.5 °
= 0 (for NC_ETC_CONF=0)

MTCPWM_ENVD = MTCPWM (for NC_ETC_CONF=1)with resolution 0.7812 and range -100...99,219 %
= 0 (for NC_ETC_CONF=0)

TPS_SP_ENVD = TPS_SP (for NC_ETC_CONF=1) with resolution 0.4686 and range 0...119.5 °
= 0 (for NC_ETC_CONF=0)

TPS_LIH_ENVD = TPS_LIH (for NC_ETC_CONF=1) with resolution 0.4686 and range 0...119.5 °
= 0 (for NC_ETC_CONF=0)

V_PVS_1_ENVD = V_PVS_1 (for NC_ETC_CONF=1) with resolution 0.01953 and range 0...4.9805 V
= 0 (for NC_ETC_CONF=0)

V_PVS_2_ENVD = V_PVS_2 (for NC_ETC_CONF=1) with resolution 0.01953 and range 0...4.9805 V
= 0 (for NC_ETC_CONF=0)

PV_AV_SEL_ENVD = PV_AV_SEL (for NC_ETC_CONF=1) with resolution 0.3906 and range 0...99.609 %
= 0 (for NC_ETC_CONF=0)

TPS_AV_ENVD_H = TPS_AV with resolution 1.86 and range 0...117.64 V

TI_1_x_ENVD_H = TI_1_x with resolution 1.02 and range 0...261.11 ms

TI_1_x_ENVD_L = TI_1_x with resolution 0.004 and range 0...1.02 ms

TI_2_MES_x_ENVD_H = TI_2_MES_x with resolution 1.02 and range 0...261.11 ms

FAC_H_RNG_LAM_AD_ENVD_H=FAC_H_RNG_LAM_AD with resolution 0.389 and range -50...49.609

FAC_L_RNG_LAM_AD_ENVD_H=FAC_L_RNG_LAM_AD with resolution 0.389 and range -50...49.609

MFF_ADD_LAM_AD_ENVD_L=MFF_ADD_LAM_AD with resolution 0.021195 and range -2.713...2.692

DTP_DIF_ACT_ENVD = DTP_DIF_ACT with resolution 0.32 and range -40.96...40.64 hPa

VLS_UP_1_ENVD_H = VLS_UP with resolution 1.247 and range 0...3.741 V

VLS_DOWN_1_ENVD_H = VLS_DOWN with resolution 1.247 and range 0...3.741 V

SEG_AD_MMV_ER_x_ENVD = SEG_AD_MMV_ER with resolution 0.061 and range -7.81... 7.752 ‰

V_PSP_MES_ENVD_H = V_PSP_MES with resolution 1.245 and range 0...3.733 V

V_PSP_MES_ENVD_L = V_PSP_MES with resolution 0.0048 and range 0...1.245 V

V_ACP_MES_ENVD_H = V_ACP_MES with resolution 1.245 and range 0...3.733 V

V_ACP_MES_ENVD_L = V_ACP_MES with resolution 0.0048 and range 0...1.245 V

CRU_BAS_ENVD_H = CRU_BAS (for NC_ETC_CONF=1) with resolution 1.245 and range 0...3.733 V
= 0 (for NC_ETC_CONF=0)

CRU_BAS_ENVD_L = CRU_BAS (for NC_ETC_CONF=1) with resolution 0.0048 and range 0...1.245 V
= 0 (for NC_ETC_CONF=0)

ENVD_x_MON = ENVD_x_MON (for NC_ETC_CONF=1)
= 0 (for NC_ETC_CONF=0)

DIAG_INST_ENVD = diagnosis instance causing current ENVD request

FTL_MMV_ENVD = FTL_MMV with resolution 0.39 and range 0...100%

FTL_MMV_REF_INI_ENVD = FTL_MMV_REF_INI with resolution 0.39 and range 0...100%

FTL_MMV_DIF_POS_DIAG_ENVD = FTL_MMV_DIF_POS_DIAG with resolution 0.39 and range 0...100%

FTL_MMV_DIF_NEG_DIAG_ENVD = FTL_MMV_DIF_NEG_DIAG with resolution 0.39 and range 0...100%

V_FTL_SUB_MMV_ENVD = V_FTL_SUB_MMV with resolution 0.0195 and range 0…4.985V

V_FTL_SUB_ENVD = V_FTL_SUB with resolution 0.0195 and range 0…4.985V

V_FTL_SUB_REF_INI_ENVD = V_FTL_SUB_REF_INI with resolution 0.0195 and range 0…4.985V

V_FTL_SUB_DIF_POS_DIAG_ENVD = V_FTL_SUB_DIF_POS_DIAG with resolution 0.0195 and range 0…4.985V

V_FTL_SUB_DIF_NEG_DIAG_ENVD = V_FTL_SUB_DIF_NEG_DIAG with resolution 0.0195 and range -4.985…0V

WHEEL_GRD_MMV_ENVD_H = WHEEL_GRD_MMV with resolution 3.921 and range 0...999.98‰

WHEEL_GRD_MMV_ENVD_L = WHEEL_GRD_MMV with resolution 0.0153 and range 0...3.921‰

CPU_LOAD_MAX_ENVD_H = CPU_LOAD_MAX with resolution 25 and range 0...100%

CPU_LOAD_MAX_ENVD_L = CPU_LOAD_MAX with resolution 0,09765625 and range 0...24.90%

VCC_PVS_1_DIAG_ENVD = VCC_PVS_1_DIAG (for NC_ETC_CONF=1) with resolution 0.03906 and
= 0 (for NC_ETC_CONF=0) range 0...9.9609 V

VCC_PVS_2_DIAG_ENVD = VCC_PVS_2_DIAG with resolution 0.03906 and range 0...9.9609 V

MAP_MES_BAS_ENVD = MAP_MES_BAS with resolution 21.309 and range 0...5434 hPa

V_PORT_ENVD = V_PORT with resolution 0.01953 and range 0...4.9805 V

PORT_AV_ENVD = PORT_AV with resolution 0.78125 and range -100...99.219 °PORT

PORTPWM_CTL_ENVD_ = PORTPWM_CTL with resolution 0.78125 and range -100...99.219 %

TTIP_MES_LS_UP_1_ENVD_H = TTIP_MES_LS_UP[1] with resolution 16 and range -2048...2032 °C

VLS_OFS_LSL_1_ENVD_H = VLS_OFS_LSL[1] with resolution 0.039215 and range -5...4.96093 V

VLS_UP_DIAG_1_ENVD_H = VLS_UP_DIAG[1] with resolution 1.25 and range -160...158.75 V

FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_H = FAC_MV_DIAG_DYN_LSL_UP[1] with resolution 0.249 and range 0....31.74999

FAC_MV_DIAG_DYN_LSL_UP_1_ENVD_L = FAC_MV_DIAG_DYN_LSL_UP[1] with resolution 9.7656e-4 and range 0....0.249

LAMB_DELTA_I_LAM_ADJ_1_ENVD_H = LAMB_DELTA_I_LAM_ADJ[1] with resolution 1.5625e-2 and range -0.125...0.125

LAMB_DELTA_LAM_ADJ_1_ENVD_H = LAMB_DELTA_LAM_ADJ[1] with resolution 1.5625e-2 and range -0.125...0.125

VLS_DIF_LAM_ADJ_1_ENVD_H = VLS_DIF_LAM_ADJ[1] with resolution 1.25 and range -5...3.75 V

LAMB_SP_1_ENVD_H = LAMB_SP[1] with resolution 0.249 and range 0....31.74999

LAMB_SP_1_ENVD_L = LAMB_SP[1] with resolution 9.7656e-4 and range 0...0.249

TEG_DYN_LS_UP_1_ENVD_H = TEG_DYN_LS_UP[1] with resolution 16 and range -273.15...1758.85 °C

TEG_DYN_LS_UP_1_ENVD_L = TEG_DYN_LS_UP[1] with resolution 0.0627 and range 0...16 °C

LAMB_LS_UP_1_ENVD_H = LAMB_LS_UP[1] with resolution 0.249 and range 0...31.74999

LAMB_LS_UP_1_ENVD_L = LAMB_LS_UP[1] with resolution 9.7656e-4 and range 0....0.249

PORTPWM_ENVD = PORTPWM with resolution 0.78125 and range -100...99.219 %

VCC_SENS_SUB_DIAG_ENVD = VCC_SENS_SUB_DIAG with resolution 0.03906 and range 0...9.9609 V

PWM_WG_ENVD = PWM_WG with resolution 0.3906 and range 0...99.609 % (only if NC_CHRG_CONF = 1) else PWM_WG_ENVD = 0

PUT_ENVD = PUT with resolution 21.309 and range 0...5434 hPa (only if NC_CHRG_CONF = 1) else PUT_ENVD = 0

PRS_AIC_DOWN_ENVD = PRS_AIC_DOWN with resolution 21.309 and range 0...5434 hPa (only if NC_CHRG_CONF = 1) else PRS_AIC_DOWN_ENVD = 0

PUT_SP_ENVD = PUT_SP with resolution 21.309 and range 0...5434 hPa (only if NC_CHRG_CONF = 1) else PUT_SP_ENVD = 0

TQI_TCHA_PROT_ENVD = TQI_TCHA_PROT with resolution 4.015568 and range 0...1023.97 Nm (only if NC_CHRG_CONF = 1) else TQI_TCHA_PROT_ENVD = 0

PUT_WG_OPEN_ENVD = PUT_WG_OPEN with resolution 21.309 and range 0...5434 hPa (only if NC_CHRG_CONF = 1) else PUT_WG_OPEN_ENVD = 0

TEMP_CAT_ENVD = TEMP_CAT[NC_NR_CAT_MDL] with resolution 16 and range -273.15...1758.85 °C

T_ES_DIAG_ENVD_H = T_ES_DIAG with resolution 256 and range 0...65280 sec

T_ES_DIAG_ENVD_L = T_ES_DIAG with resolution 1 and range 0...255 sec

TEG_DYN_LS_DOWN_1_ENVD_H = TEG_DYN_LS_DOWN[1] with resolution 16 and range -273.15...1758.85 °C

TEMP_CAT_DYN_MDL_1_ENVD_H = TEMP_CAT_DYN_MDL[1] with resolution 16 and range -273.15...1758.85 °C

MAP_MES_ENVD = MAP_MES with resolution 21.309 and range 0...5434 hPa

V_VIM_ENVD = V_VIM with resolution 0.01953 and range 0...4.9805 V

VP_MAP_ENVD = VP_MAP with resolution 0.01953 and range 0...4.9805 V

VP_PUT_ENVD = VP_PUT with resolution 0.01953 and range 0...4.9805 V

VP_AMP_ENVD = VP_AMP with resolution 0.01953 and range 0...4.9805 V

VP_TAM_ENVD = VP_TAM with resolution 0.01953 and range 0...4.9805 V (only if NC_CHRG_CONF = 1) else VP_TAM_ENVD = 0

B.55.2 Configuration of update of FRF_CUS_CMN

Description:

This file defines the update of the FRF_CUS_CMN in case of failure status change from disappeared to temporary/confirmed. To follow specific customer requirements, this file is a template to be fulfilled by project team.

Update of FRF_CUS_CMN in case of failure status change from disappeared to temporary/confirmed.	NC_ENVD_CUS_CMN_UPD = 1 0: no update 1: update

B.55.3 Configuration for prestored freeze frame

Description:

This file defines the usage of the prestored freeze frame functionality. To follow specific customer requirements, this file is a template to be fulfilled by project team.

Number of diagnosis instances XX using prestored freeze frame	NC_NR_ENVD_PREV = 2

Diagnosis instance XX	**prestored freeze frame number (t)**
CAT_DIAG_1	1
VAP_LEAK_1_MIL	2

Remark: The usage of the prestored freeze frame functionality must be limited for special diagnosis functions, because of memory consumption.

B.56 Statistical data

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DTC_CLR	V/O/S	0...FFFFH	0...65535	1	[min]
Time since diagnostic trouble codes cleared					
DIST_ACT_MIL	V/O/S	0...FFFFH	0...65535	1	[km]
Global distance traveled while the MIL is illuminated					
DIST_REL_ACT_MIL	- /S	0...FFFFFFFFH	0...4294967295	1	[km]
Relative distance to calculate distance traveled while the MIL is illuminated					
CTR_ACT_MIL	- /S	0...FFH	0...255	1	[-]
Counter for warm up cycle with MIL Off					
T_ACT_MIL	V/O/S	0...FFFFH	0...65535	1	[min]
Minutes run by the engine while MIL activated					
T_ACT_MIL_60	V/S	0...3CH	0...60	1	[s]
Seconds run by the engine while MIL activated for T_ACT_MIL calculation (0 ... 59)					
DIST_DTC_CLR	V/O/S	0...FFFFH	0...65535	1	[km]
Distance since diagnostic trouble codes cleared					
DIST_REL_DTC_CLR	V/S	0...FFFFFFFFH	0...4294967295	1	[km]
Distance at following event : diagnostic trouble codes cleared					
CTR_WUP_DTC_CLR	V/O/S	0...FFH	0...255	1	[-]
Number of warm-ups since diagnostic trouble codes cleared					
STATE_ENA_OBD	V/O	0...FFFFH	0...65535	1	[-]
Monitor enable status for the current driving cycle					
STATE_CMPL_OBD	V/O	0...FFFFH	0...65535	1	[-]
Monitor completion status for the current driving cycle					

Input data:

DIST	C_CTR_MAX_WUP_CYC	STATE_MIL	LV_DC
C_STATE_READY_OBD_1	C_STATE_READY_OBD_2	LV_END_DIAG_XX	WAL_CONF_XX
LV_WUP_CYC	CONF_KOBD	LV_MIS_STATE_A	LV_MIS_STATE_B1
LV_MIS_STATE_B4	LV_END_DIAG_CAT_DIAG [NC_CBK_EX_NR]	LV_ERR_FSD_LAM_LIM [NC_CBK_EX_NR]	LV_ERR_FSD [NC_CBK_EX_NR]
LV_ES	LV_END_DIAG_VAP_LEAK_1	NLC_OBD_DSL	LV_MIL_ACT_REQ

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_MIL_ACT_REQ	1	0...1H	0...1	1	[-]
External MIL request is taken or not into account for DIST_ACT_MIL and T_ACT_MIL calculation (0 : not taken and 1 : taken)					

Import actions:

ACTION_ERRM_MonitorEnableStatus (INOUT < MonitorEnableStatus>)
This action allows to read the MonitorEnableStatus information

FUNCTION DESCRIPTION:

General information:

EOBD/CARB official requirements include a list of statistical data related to Error Management. These statistical data are counters and timers related to Error Management events. You can find below, in the following chapters, a description of all these requirements.

B.56.1 Time since diagnostic trouble codes cleared

Description:

Purpose of this strategy is to compute the time accumulated by the vehicle since diagnostic trouble codes cleared (referenced in ISO15031 as Modes 01h & 02h PID4Eh). It is simply an indication for I/M (Inspection/Maintenance), of the last time an external test equipment was used to clear DTCs.

Application conditions:

Initialization: on non-volatil memory reset or lost
or on failure erase service received
T_DTC_CLR = 0

Recurrence : 1 minute ±1 second, after event : on “failure erase” service received

Activation : -

Deactivation: -

Formula section :

{ T_DTC_CLR is updated at required recurrence }
{ T_DTC_CLR shall saturate to its maximal value 65535 }

If T_DTC_CLR < 65535
Then
T_DTC_CLR = T_DTC_CLR + 1
Endif

B.56.2 Distance counters since MIL activation and Minutes run by the engine with MIL activated

Description:

According to ISO/DIS 15031-5.8 (Road vehicles — Communication between vehicle and external equipment for emissions-related diagnostics — Part 5: Emissions-related diagnostic services), ECU must be able to indicate :

- the distance travelled while the MIL is activated
- the number of minutes run by the engine while MIL is activated

Distance Travelled While MIL is Activated

The purpose is to compute the covered distance while the MIL is illuminated (MIL blinking is considered as a MIL illumination). This mileage (DIST_ACT_MIL) is a global distance :

- which is not linked with any failure and is independent from the Pre-drive check
- which is dependent from the global inhibition MIL boolean
- which is dependent of the MIL request
- which is readable with Scan-Tool with kilometer as unit

Conditions for “Distance travelled” counter :

- reset to 0 when MIL state changes from deactivated to activated by this ECU
- accumulate counts in km if MIL is activated (ON)
- do not change value while MIL is not activated (OFF)
- reset to 0 if diagnostic information is cleared either by service 04h or 40 warm-up cycles without MIL activated
- do not wrap to 0 if value is FFFFh

Minutes run by the engine while MIL activated

The purpose is to compute the time accumulated while MIL is activated (Modes 01h & 02h PID4Dh). This duration shall be cleared when failure memory is erased and after 40 warm-up cycles.

Conditions for “Minutes run by the engine while MIL activated” counter :

- reset to 0 when MIL state changes from deactivated to activated by this ECU
- accumulate counts in minutes if MIL is activated (ON)
- do not change value while MIL is not activated (OFF)
- reset to 0 if diagnostic information is cleared either by service 04h or 40 warm-up cycles without MIL activated
- do not wrap to 0000h if value is FFFFh

Application conditions:

Initialization: On saved ram lost

or [(STATE_MIL transition OFF -> MIL_FLL or OFF -> ON)

and LV_MIL_ACT_REQ = 0]
or (NLC_MIL_ACT_REQ = 1
and LV_MIL_ACT_REQ transition 0 -> 1
and bit 1 of C_CONF_MIL = 1
and STATE_MIL = OFF)
or failure erase service received
DIST_REL_ACT_MIL = DIST
DIST_ACT_MIL = 0
CTR_ACT_MIL = C_CTR_MAX_WUP_CYC
T_ACT_MIL = 0
T_ACT_MIL_60 = 0

Recurrence: upon LV_DC 1→0 transition and 1 s

Activation: LV_ES = 0

Formula section:

Each 1s:

```
If(1)   STATE_MIL = MIL_FLL
        or
        STATE_MIL = ON
        or
        (NLC_MIL_ACT_REQ = 1 and LV_MIL_ACT_REQ = 1
        and bit 1 of C_CONF_MIL = 1)
Then(1)
        If(2)   ( DIST – DIST_REL_ACT_MIL ) < 65535
        Then(2)
                DIST_ACT_MIL = DIST – DIST_REL_ACT_MIL
        Else(2)
                DIST_ACT_MIL = 65535
        Endif(2)
        T_ACT_MIL_60 = T_ACT_MIL_60 + 1 s
        If(2)   (T_ACT_MIL_60 = 60)
        Then(2)
                T_ACT_MIL_60 = 0
                If(3)   T_ACT_MIL < 65535
                Then(3)
                        T_ACT_MIL = T_ACT_MIL + 1 min
                Else(3)
```

T_ACT_MIL = 65535

Endif(3)

Endif(2)

Endif(1)

At LV_DC 1→0 transition :

If(1) LV_WUP_CYC = 1

and

(LV_MIL_ACT_REQ = 0 during all the warm-up cycle

or bit 1 of C_CONF_MIL = 0 **or** NLC_MIL_ACT_REQ = 0**)**

and

STATE_MIL was OFF during all the warm-up cycle

and

CTR_ACT_MIL > 0

Then(1)

CTR_ACT_MIL = CTR_ACT_MIL –1

If(2) CTR_ACT_MIL = 0

Then(2)

DIST_REL_ACT_MIL = DIST

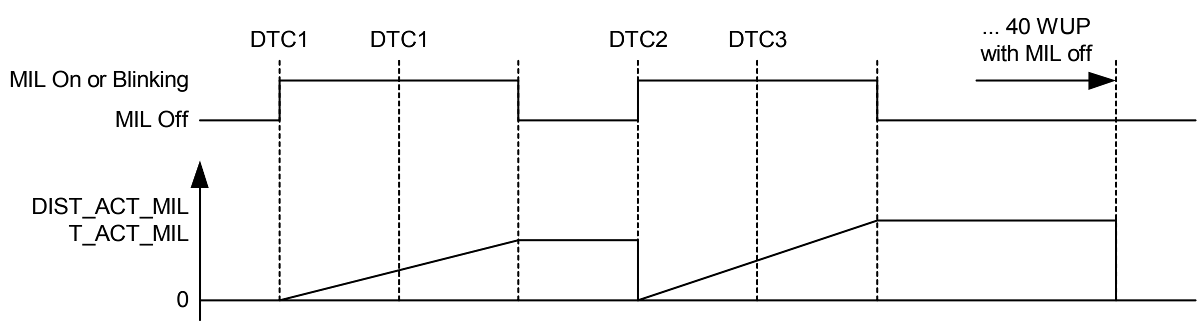
DIST_ACT_MIL = 0

T_ACT_MIL = 0

T_ACT_MIL_60 = 0

Endif(2)

Endif(1)



B.56.3 Distance since diagnostic trouble codes cleared

Description :

general specification

Purpose of this strategy is to compute the distance accumulated by the vehicle since diagnostic trouble codes cleared (referenced in ISO15031 as mode01h&02hPID31h). It is simply an indication for Inspection/Maintenance, of the last time an external test equipment was used to clear DTCs.

Application conditions:

Initialization:	on non-volatil memory reset or lost
	or
	on failure erase service received DIST_DTC_CLR = 0 DIST_REL_DTC_CLR = DIST
Recurrence :	1 second
Activation :	-
Deactivation:	-

Formula section :

{ DIST_DTC_CLR is updated regularly regarding DIST }
{ DIST_DTC_CLR shall saturate to its maximal value 65535 km}

```
If      DIST - DIST_REL_DTC_CLR < 65535
Then
        DIST_DTC_CLR = DIST - DIST_REL_DTC_CLR
Else
        DIST_DTC_CLR = 65535
Endif
```

B.56.4 Number of warm-ups since diagnostic trouble codes cleared

Description:

Purpose of this strategy is to compute number of warm-ups since diagnostic trouble codes cleared (referenced in ISO15031 as Modes 01h & 02h PID30h). It's just an indication for Inspection/Maintenance, to determine the last time an external test equipment was used to clear DTCs.

Application conditions:

Activation : On LV_WUP_CYC 0 $\rightarrow$ 1 transition

Initialization: On non-volatil memory reset or lost
or on failure erase service received
CTR_WUP_DTC_CLR=0

Recurrence : none, executed single time on activation.

Formula section:

{ on detection of a warm-up cycle CTR_WUP_DTC_CLR is incremented }
{ CTR_WUP_DTC_CLR shall saturate to its maximal value 255 }

If CTR_WUP_DTC_CLR < 255
Then
CTR_WUP_DTC_CLR = CTR_WUP_DTC_CLR + 1
Endif

B.56.5 Monitor enable/completion status for the current driving cycle

General information:

Purpose of this strategy is to get the follwoing informations :
- if a diagnosis (a monitor) is disabled for rest of this monitoring cycle or not (referenced in ISO15031 as Modes 01h & 02h PID41h).
- if a diagnosis (a monitor) is completed for the current monitoring cycle or not (referenced in ISO15031 as Modes 01h & 02h PID41h).

Each concerned diagnosis is part of a generic list : Misfire, Fuel system, Comprehensive component, Catalyst, Heated catalyst, Evaporative system, Secondary air system, A/C system refrigerant, Oxygen sensor, Oxygen sensor heater, EGR system monitoring.

MonitorEnableStatus :
Each bit of STATE_ENA_OBD variable contains status for each predefined diagnosis as defined below :
Enable status of a diagnosis this monitoring cycle:
- NO (STATE_ENA_OBD[n]=0) means disabled for rest of this monitoring cycle
  or not supported in C_STATE_READY_OBD_1 & C_STATE_READY_OBD_2
- YES (STATE_ENA_OBD[n]=1) means enabled for this monitoring cycle.

MonitorCompletionStatus :
Each bit of STATE_CMPL_OBD variable contains status for each predefined diagnosis as defined below :
Completion status of a diagnosis this monitoring cycle:
- YES (STATE_CMPL_OBD[n]=0) means monitor complete this monitoring cycle, or not supported in C_STATE_READY_OBD_1 & C_STATE_READY_OBD_2
- NO (STATE_CMPL_OBD[n]=1) means monitor not complete this monitoring cycle.

Bit 0 to 7 of STATE_ENA_OBD[0..7] :

- bit0: Misfire monitoring
- bit1: Fuel system monitoring
- bit2: Comprehensive component monitoring
- bit3: GS/DS vehicle (NLC_OBD_DSL = 1 for DS, 0 for GS)
- bit4: reserved
- bit5: reserved
- bit6: reserved
- bit7: reserved

Bit 8 to 15 of STATE_ENA_OBD[8..15] :

- bit8: Catalyst monitoring
- bit9: Heated catalyst monitoring
- bit10: Evaporative system monitoring
- bit11: Secondary air system monitoring
- bit12: A/C system refrigerant monitoring
- bit13: Oxygen sensor monitoring
- bit14: Oxygen sensor heater monitoring
- bit15: EGR system monitoring monitoring

Bit 0 to 7 of STATE_CMPL_OBD[0..7] :

- bit0: Misfire monitoring
- bit1: Fuel system monitoring
- bit2: Comprehensive component monitoring
- bit3: GS/DS vehicle (NLC_OBD_DSL = 1 for DS, 0 for GS)
- bit4: reserved
- bit5: reserved
- bit6: reserved
- bit7: reserved

Bit 8 to 15 of STATE_CMPL_OBD[8..15] :

- bit8: Catalyst monitoring
- bit9: Heated catalyst monitoring
- bit10: Evaporative system monitoring
- bit11: Secondary air system monitoring
- bit12: A/C system refrigerant monitoring
- bit13: Oxygen sensor monitoring
- bit14: Oxygen sensor heater monitoring
- bit15: EGR system monitoring monitoring

Application conditions:

Initialization: on LV_DC 1 $\rightarrow$ 0 transition

STATE_ENA_OBD = 0

STATE_CMPL_OBD = 0

Recurrence : 10 seconds

Activation : -

Deactivation: -

<u>Formula section :</u>

{

{ *Action call definition shall define strategy related to each supported diagnosis* }
{ *each bit of STATE_ENA_OBD shall be computed according diagnosis strategy* }

{ *STATE_ENA_OBD computation* }
{ *STATE_ENA_OBD is the result returned by ACTION_ERRM_MonitorEnableStatus action* }
{ Please refer to ACTION_ERRM_MonitorEnableStatus definition in }
{ the Application Incidence file }

ACTION_ERRM_MonitorEnableStatus(INOUT<STATE_ENA_OBD>,
SYNCHRONIZATION<CALL>)

{ *STATE_CMPL_OBD computation* }
{ *Each bit of STATE_CMPL_OBD shall be computed according end diagnosis flags* }

STATE_CMPL_OBD[0..7] = C_STATE_READY_OBD_1
STATE_CMPL_OBD[8..15] = C_STATE_READY_OBD_2

<u>Distinction between GS vehicle and DS vehicle</u>

Bit 3 of STATE_ENA_OBD = NLC_OBD_DSL

Bit 3 of STATE_CMPL_OBD = NLC_OBD_DSL

Misfire monitoring status bit :

If CONF_KOBD = 0
Then
If Bit 0 of C_STATE_MONI_CMPL_OBD_1 = 0
(this option is used to have similar behaviour of readiness and completion status of misfire monitoring)
Then

```
                    If      LV_END_DIAG_XX=1 for all diagnosis XX defined below
                            Take into account all LV_END_DIAG_XX which fulfill the following
                            condition:
                            this XX failure has the status CARB_MIS (see tables of failures)
                            and
                            the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
                    Then    Bit 0 of STATE_CMPL_OBD = 0
                    Endif
            Else    Monitoring status for misfire monitoring is always set
                    Bit 0 of STATE_CMPL_OBD = 0
            EndIf

Else        (CONF_KOBD = 1)
            if      LV_END_DIAG_CAT_DIAG_1 = 1
            then    Bit 0 of STATE_CMPL_OBD = 0
            else
                    if      (LV_MIS_STATE_A = 1
                    or      LV_MIS_STATE_B1 = 1
                    or      LV_MIS_STATE_B4 = 1)
                    then    Bit 0 of STATE_CMPL_OBD = 0
                    endif
            EndIf
EndIf
```

Fuel system monitoring status bit :

```
If          CONF_KOBD = 0
Then
            If      Bit 1 of C_STATE_MONI_CMPL_OBD_1 = 0
                    (this option is used to have similar behaviour of readiness and completion
                     status of fuel system monitoring)
            Then
                    If      LV_END_DIAG_XX=1 for all diagnosis XX defined below
                            Take into account all LV_END_DIAG_XX which fulfill the following
                            condition:
                            this XX failure has the status CARB_FSD (see tables of failures)
                            and the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
                    Then    Bit 1 of STATE_CMPL_OBD = 0
                    Endif
            Else    Monitoring status for fuel system monitoring is always set
                    Bit 1 of STATE_CMPL_OBD = 0
            EndIf

Else        (CONF_KOBD = 1)
            if      LV_END_DIAG_CAT_DIAG_1 = 1
            then    Bit 1 of STATE_CMPL_OBD = 0
            else
                    if      LV_ERR_FSD_1 = 1
                    or      LV_ERR_FSD_LAM_LIM_1= 1
                    then    Bit 1 of STATE_CMPL_OBD = 0
                    endif
            EndIf
```

EndIf

Comprehensive component monitoring status bit :

If CONF_KOBD = 0
Then
  If Bit 2 of C_STATE_MONI_CMPL_OBD_1 = 0
  (this option is used to have similar behaviour of readiness and completion status of comprehensive components monitoring)
  Then
    If LV_END_DIAG_XX = 1 for all diagnosis XX defined below
    Take into account all LV_END_DIAG_XX which fulfill the following condition:
    this XX failure has the status CARB_CC (see tables of failures)
    and the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
    Then Bit 2 of STATE_CMPL_OBD = 0
    Endif
  Else Readiness for Comprehensive component status is always set
  Bit 2 of STATE_CMPL_OBD = 0
  EndIf
Else (CONF_KOBD = 1)
Readiness for Comprehensive component status is always set
Bit 2 of STATE_CMPL_OBD = 0
EndIf

Catalyst monitoring bit :

If LV_END_DIAG_XX = 1 for all diagnosis XX defined below
Take into account all LV_END_DIAG_XX which fulfill the following condition:
this XX failure has the status CARB_CAT (see tables of failures) **and** the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
Then Bit 8 of STATE_CMPL_OBD = 0
Endif

Heated catalyst bit :

If LV_END_DIAG_XX = 1 for all diagnosis XX defined below
Take into account all LV_END_DIAG_XX which fulfill the following condition:
this XX failure has the status CARB_HC (see tables of failures) **and** the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
Then Bit 9 of STATE_CMPL_OBD = 0
Endif

Evaporative system monitoring bit :

If (LV_END_DIAG_XX = 1 for all diagnosis XX defined below **or** LV_ERR_XX = 1 for one of diagnosis XX defined below)
Take into account all LV_END_DIAG_XX or LV_ERR_XX = 1 which fulfill the following condition:
this XX failure has the status CARB_EVAP(see tables of failures) **and**

the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 10 of STATE_CMPL_OBD = 0
Else
If LV_END_DIAG_VAP_LEAK_1 = 1
then Bit 10 of STATE_CMPL_OBD = 0
Endif

Endif
Remark: If once Bit 10 of STATE_CMPL_OBD is set to 0, it is kept to 0 in the driving cycle.

Secondary air system monitoring bit :

If LV_END_DIAG_XX = 1 for all diagnosis XX defined below
Take into account all LV_END_DIAG_XX which fulfill the following condition:
this XX failure has the status CARB_SA (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 11 of STATE_CMPL_OBD = 0
Endif

A/C system refrigerant monitoring bit :

If LV_END_DIAG_XX = 1 for all diagnosis XX defined below
Take into account all LV_END_DIAG_XX which fulfill the following condition:
this XX failure has the status CARB_AC (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 12 of STATE_CMPL_OBD = 0
Endif

Oxygen sensor monitoring bit :

If (LV_END_DIAG_XX = 1 for all diagnosis XX defined below **or**
LV_ERR_XX = 1 for one of diagnosis XX defined below)

Take into account all LV_END_DIAG_XX or LV_ERR_XX = 1 which fulfill the following condition:
this XX failure has the status CARB_LS (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 13 of STATE_CMPL_OBD = 0
Endif
Remark: If once Bit 13 of STATE_CMPL_OBD is set to 0, it is kept to 0 in the driving cycle.

Oxygen sensor heater monitoring bit :

If (LV_END_DIAG_XX = 1 for all diagnosis XX defined below **or**
LV_ERR_XX = 1 for one of diagnosis XX defined below)

Take into account all LV_END_DIAG_XX or LV_ERR_XX = 1 which fulfill the following condition:
this XX failure has the status CARB_LSH (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 14 of STATE_CMPL_OBD = 0

Endif

Remark: If once Bit 14 of STATE_CMPL_OBD is set to 0, it is kept to 0 in the driving cycle.

EGR system monitoring bit :

If (LV_END_DIAG_XX = 1 for all diagnosis XX defined below **or**
LV_ERR_XX = 1 for one of diagnosis XX defined below).
Take into account all LV_END_DIAG_XX or LV_ERR_XX = 1 which fulfill the following condition:
this XX failure has the status CARB_EGR (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 15 of STATE_CMPL_OBD = 0
Endif

Remark: If once Bit 15 of STATE_CMPL_OBD is set to 0, it is kept to 0 in the driving cycle.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_STATE_MONI_CMPL_OBD_1	1	0...FFH	0...255	1	[-]
Direct setting of monitoring completion status for MIS, FSD, CC					

Calibration data detailled description for C_STATE_MONI_CMPL_OBD_1 :

Completion status calculation for CARB_MIS, CARB_FSD, CARB_CC diagnosis :

0: completion status of concerned group is calculated based on the end of diagnosis flag of each diagnosis of the group
1: completion status of concerned group always indicates "complete"

C_STATE_MONI_CMPL_OBD_1:

```
|_|_|_|_|_|_|_|_|
 | | | | | | | └─> bit0 : set directly completion status of misfire monitoring
 | | | | | | └───> bit1 : set directly completion status of fuel system monitoring
 | | | | | └─────> bit2 : set directly completion status of compr. component monitoring
 | | | | └───────> bit3 : 0
 | | | └─────────> bit4 : 0
 | | └───────────> bit5 : 0
 | └─────────────> bit6 : 0
 └───────────────> bit7 : 0
```

B.57 Statistical data (Appl. Inc.)

FUNCTION DESCRIPTION:

General information:

EOBD/CARB regulation and ISO standards requirements include a list of statistical data related to Error Management. These statistical data are counters and timers related to Error Management events. You can find below in following chapters a description of all these requirements (depends on customer needs).

B.57.1 Monitor enable status for the current driving cycle

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_ENA_OBD_STAT	V	0...FFFFH	0...65535	1	[-]
Monitor enable status for the current driving cycle. Stays 0 for current DC, if disable status is reached					

Input data:

C_STATE_READY_OBD_1	C_STATE_READY_OBD_2	AMP	C_AMP_MIN_MIS
C_AMP_MIN_FSD	CTR_STOP_FSD	C_STOP_MAX_FSD	AMP_AD
C_AMP_MIN_CAT_LS_DIAG	C_AMP_AD_MIN_DIAGCP	NLC_OBD_DSL	

Export actions:

ACTION_ERRM_MonitorEnableStatus (IN,OUT < MonitorEnableStatus>)
This action allows to read the MonitorEnableStatus information

Description :

Purpose of this strategy is to get the follwoing informations :
- if a diagnosis (a monitor) is disabled for rest of this monitoring cycle or not (referenced as Mode01h&02hPID41h ISO15031).

Each concerned diagnosis is part of a generic list : Misfire, Fuel system, Comprehensive component, Catalyst, Heated catalyst, Evaporative system, Secondary air system, A/C system refrigerant, Oxygen sensor, Oxygen sensor heater, EGR system monitoring.

MonitorEnableStatus :
Each bit of MonitorEnableStatus variable contains status for each predefined diagnosis as defined below :
Enable status of a diagnosis this monitoring cycle: NO (MonitorEnableStatus[n]=0) means disabled for rest of this monitoring cycle, YES (MonitorEnableStatus[n]=1) means enabled for this monitoring cycle.

Bit 0 to 7 of MonitorEnableStatus[0..7] :

```
┌┬┬┬┬┬┬┬┐
└┴┴┴┴┴┴┴┘
 | | | | | | | └─> bit0: Misfire monitoring
 | | | | | | └───> bit1: Fuel system monitoring
 | | | | | └─────> bit2: Comprehensive component monitoring
 | | | | └───────> bit3: GS/DS vehicle (NLC_OBD_DSL = 1 for DS, 0 for GS)
 | | | └─────────> bit4: reserved
 | | └───────────> bit5: reserved
 | └─────────────> bit6: reserved
 └───────────────> bit7: reserved
```

Bit 8 to 15 of MonitorEnableStatus[8..15] :

```
┌┬┬┬┬┬┬┬┐
└┴┴┴┴┴┴┴┘
 | | | | | | | └─> bit8: Catalyst monitoring
 | | | | | | └───> bit9: Heated catalyst monitoring
 | | | | | └─────> bit10: Evaporative system monitoring
 | | | | └───────> bit11: Secondary air system monitoring
 | | | └─────────> bit12: A/C system refrigerant monitoring
 | | └───────────> bit13: Oxygen sensor monitoring
 | └─────────────> bit14: Oxygen sensor heater monitoring
 └───────────────> bit15: EGR system monitoring monitoring
```

Description :

Syntax :	ACTION_ERRM_MonitorEnableStatus (IN,OUT < MonitorEnableStatus >)
Parameter(in) :	No parameter
Parameter(out) :	MonitorEnableStatus with the same format as here described before
Short Description :	This API calculate and returns the monitor enable status result information in the software structure.

Application conditions:

Activation : -

Initialization: at LV_DC 1 -> 0 and reset
STATE_ENA_OBD_STAT = FFFFh
STATE_ENA_OBD = C_STATE_READY_OBD_2 (bit 8...15) +
C_STATE_READY_OBD_1 (bit 0...7)

Recurrence : -

Formula section:

{ action call definition shall define the strategy related to each supported diagnosis }
{ each bit of MonitorEnableStatus shall be computed according diagnosis strategy }

Project specific MonitorEnableStatus computation

Databyte B:

Misfire monitoring status bit:

If Bit 0 of STATE_ENA_OBD_STAT = 0
then Bit 0 of STATE_ENA_OBD = 0 (Once Monitor Status is disabled it remains so.)
elseif AMP > C_AMP_MIN_MIS
then Bit 0 of MonitorEnableStatus = 1
else Bit 0 of MonitorEnableStatus = 0
Bit 0 of STATE_ENA_OBD_STAT = 0
endif
endif

B.57.2 Fuel system monitoring status bit:

If Bit 1 of STATE_ENA_OBD_STAT = 0
then Bit 1 of STATE_ENA_OBD = 0 (Once Monitor Status is disabled it remains so.)
elseif AMP > C_AMP_MIN_FSD
and CTR_STOP_FSD < C_STOP_MAX_FSD
then Bit 1 of MonitorEnableStatus = 1
else Bit 1 of MonitorEnableStatus = 0
Bit 1 of STATE_ENA_OBD_STAT = 0
endif
endif

B.57.3 Comprehensive component monitoring status bit:

B.57.4 Bit 2 of MonitorEnableStatus = 1

Databyte C:

B.57.5 Catalyst monitoring status bit:

If Bit 8 of STATE_ENA_OBD_STAT = 0

then Bit 8 of STATE_ENA_OBD = 0 (Once Monitor Status is disabled it remains so.)

elseif AMP_AD > C_AMP_MIN_CAT_LS_DIAG

then Bit 8 of MonitorEnableStatus = 1

else Bit 8 of MonitorEnableStatus = 0

Bit 8 of STATE_ENA_OBD_STAT = 0

endif

endif

B.57.6 Evaporative system monitoring status bit:

If Bit 10 of STATE_ENA_OBD_STAT = 0

then Bit 10 of STATE_ENA_OBD = 0 (Once Monitor Status is disabled it remains so.)

elseif AMP_AD > C_AMP_AD_MIN_DIAGCP

then Bit 10 of MonitorEnableStatus = 1

else Bit 10 of MonitorEnableStatus = 0

Bit 10 of STATE_ENA_OBD_STAT = 0

endif

endif

Oxygen sensor monitoring status bit:

Bit 13 of MonitorEnableStatus = 1

Oxygen sensor heater monitoring status bit:

Bit 14 of MonitorEnableStatus = 1

EGR system monitoring status bit:

Bit 15 of MonitorEnableStatus = 1

MonitorEnableStatus [0..7] = MonitorEnableStatus [0..7] and C_STATE_READY_OBD_1
MonitorEnableStatus [8..15] = MonitorEnableStatus [8..15] and C_STATE_READY_OBD_2

B.57.7 Driving cycles counters since DTC present

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_DC_ERR_DET[NC_NR_ERR_DYN]	O/V/S	0 ... FFH	0 ... 255	1	-
Number of driving cycles since DTC detection, for the failure XX=F(IDX)					
CTR_DC_ERR_DET_DIAG_CDN[NC_NR_ERR_DYN]	O/V/S	0 ... FFH	0 ... 255	1	-
Number of driving cycle since DTC detection, with diag started within these driving, cycles for the failure XX=F(IDX)					
LV_CDN_DIAG_DC[NC_NR_ERR_DYN]	V	0 .... 1H	0 ... 1	1	-
This flag permits to detect that diag. conditions have been met during driving cycle, for the failure XX=F(IDX)					
CTR_DC_ERR_DET_DIAG_CMPL[NC_NR_ERR_DYN]	O/V/S	0 ... FFH	0 ... 255	1	-
Number of driving cycle since DTC detection, with diag completed during these driving cycles, for the failure XX=F(IDX)					

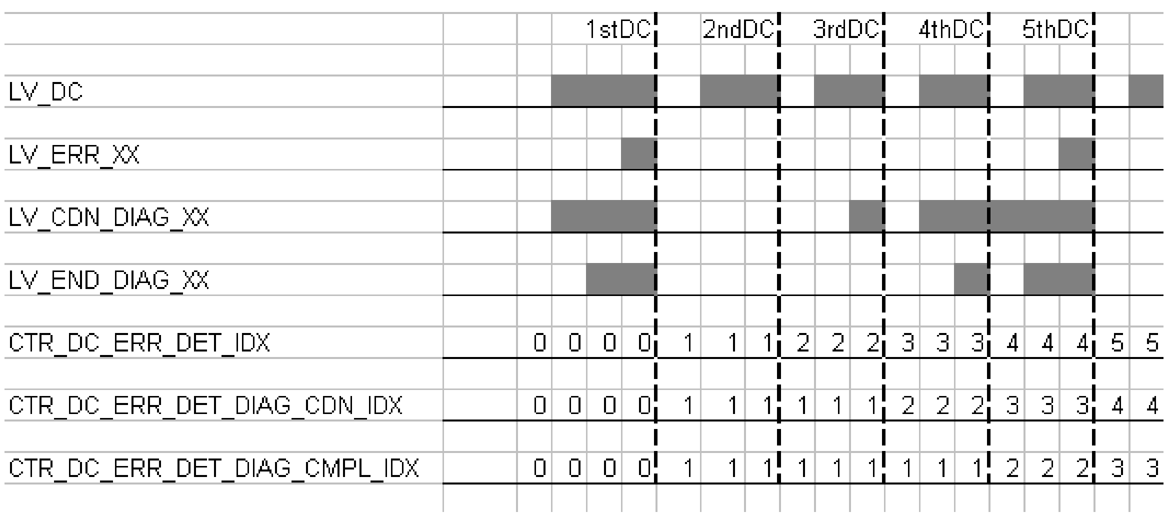
Input data:

LV_DC	LV_ERR_DC[NC_NR_ERR_DYN]	LV_END_DIAG_XX	LV_CDN_DIAG_XX
	LV_ERR_MEM_XX		

FUNCTION DESCRIPTION:

General information:

Purpose of this strategy is to compute number of driving cycles since detection of diagnostic trouble codes.



Signal flow diagram:

None

Description:

Purpose of this formula section is to detect that diag. conditions have been met during driving cycle.

Application conditions:

Initialisation: at LV_DC 1 → 0 transition
LV_CDN_DIAG_DC[IDX]=0

Recurrence: none, executed only once on activation

Activation: at LV_CDN_DIAG_XX 0 → 1 transition
and
LV_ERR_MEM_XX=1

Deactivation: -

Formula section:

{ on detection of diagnosis conditions LV_CDN_DIAG_DC[IDX] is updated}

LV_CDN_DIAG_DC[IDX]=1

Description :

Purpose of this section is to compute the number of driving cycles regarding the following events :
- detection of diagnostic trouble codes (action call)
- detection of an error in last driving cycle (LV_ERR_DC[IDX])
- detection of diag. conditions have been met during last driving cycle (LV_CDN_DIAG_DC[IDX])
- detection of diag. completion have been met during last driving cycle (LV_END_DIAG_XX)

Application conditions:

Activation : at LV_DC 1 → 0 transition

Initialization: on non-volatil memory reset or lost
or on failure erase service received
or on dynamic failure memory erase
CTR_DC_ERR_DET[IDX] = 0
CTR_DC_ERR_DET_DIAG_CDN[IDX] = 0
CTR_DC_ERR_DET_DIAG_CMPL[IDX] = 0

Recurrence : none, executed only once on activation

Formula section:

{ detection of an error in last driving cycle shall start counters }

{ on detection of an error in driving cycle counters are updated }
{ counters shall saturate to there maximal value 255 }

```
If        LV_ERR_DC[IDX]= 1
Then

    If    CTR_DC_ERR_DET[IDX] <MAX(CTR_DC_ERR_DET[IDX])
    Then
          CTR_DC_ERR_DET[IDX] = CTR_DC_ERR_DET[IDX] + 1
    Endif

    If    CTR_DC_ERR_DET_DIAG_CDN[IDX] <MAX(CTR_DC_ERR_DET_DIAG_CDN[IDX])
    Then
          CTR_DC_ERR_DET_DIAG_CDN[IDX] = CTR_DC_ERR_DET_DIAG_CDN[IDX]+1
    Endif

    If    CTR_DC_ERR_DET_DIAG_CMPL[IDX] <MAX(CTR_DC_ERR_DET_DIAG_CMPL[IDX])
    Then
          CTR_DC_ERR_DET_DIAG_CMPL[IDX] = CTR_DC_ERR_DET_DIAG_CMPL[IDX]+1
    Endif
```

Else { no detection of an error in driving cycle }

{ CTR_DC_ERR_DET[IDX] update }

{ after error detection CTR_DC_ERR_DET[IDX] is updated every driving cycle }
{ CTR_DC_ERR_DET[IDX] shall saturate to its maximal value 255 }

```
If        CTR_DC_ERR_DET[IDX] > 0
    And   CTR_DC_ERR_DET[IDX] <MAX(CTR_DC_ERR_DET[IDX])
Then
    CTR_DC_ERR_DET[IDX] = CTR_DC_ERR_DET[IDX] + 1
Endif
```

{ CTR_DC_ERR_DET_DIAG_CDN[IDX] update }

{ on detection of diag. conditions in driving cycle after error detection }
{ CTR_DC_ERR_DET_DIAG_CDN[IDX] is updated }
{ CTR_DC_ERR_DET_DIAG_CDN[IDX] shall saturate to its maximal value 255 }

If CTR_DC_ERR_DET_DIAG_CDN[IDX]>0
and
LV_CDN_DIAG_DC[IDX]= 1
and
CTR_DC_ERR_DET_DIAG_CDN[IDX] <MAX(CTR_DC_ERR_DET_DIAG_CDN[IDX])
Then
CTR_DC_ERR_DET_DIAG_CDN[IDX] = CTR_DC_ERR_DET_DIAG_CDN[IDX] + 1
Endif

{ CTR_DC_ERR_DET_DIAG_CMPL[IDX] update }

{ on detection of diag. conditions in driving cycle after error detection }
{ CTR_DC_ERR_DET_DIAG_CMPL[IDX] is updated }
{ CTR_DC_ERR_DET_DIAG_CMPL[IDX] shall saturate to its maximal value 255 }

If CTR_DC_ERR_DET_DIAG_CMPL[IDX]>0
and
LV_END_DIAG_XX= 1
and
CTR_DC_ERR_DET_DIAG_CMPL[IDX] <MAX(CTR_DC_ERR_DET_DIAG_CMPL[IDX])
Then
CTR_DC_ERR_DET_DIAG_CMPL[IDX] = CTR_DC_ERR_DET_DIAG_CMPL[IDX] + 1
Endif

Endif **{ detection of an error in driving cycle }**

B.58 Rate-Based Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_COMP_RBM[NC_NR_DIAG_RBM]	V/O/S	0...FFFFH	0...65535	1	[-]
Monitor individual numerator: Number of DC with monitor done since first power up					
CTR_CDN_RBM[NC_NR_DIAG_RBM]	V/O/S	0...FFFFH	0...65535	1	[-]
Monitor individual denominator: Number of DC with convenient vehicle operation for monitor since first power up					
STATE_CTR_RBM[NC_NR_DIAG_RBM]	V	0...3H	0...3	1	[-]
Information data for CTR_COMP_RBM					
CTR_IGK_CYC_RBM	V/O/S	0...FFFFH	0...65535	1	[-]
Ignition cycle counter: Number of DC since first ECU power up					
LV_IGK_CYC_RBM	V	0...1H	0...1	1	[-]
Boolean to indicate, that incrementation of CTR_IGK_CYC_RBM done this DC					
CTR_CDN_OBD_RBM	V/O/S	0...FFFFH	0...65535	1	[-]
General denominator: Number of DC with valid standardized vehicle operations since first ECU power up					
LV_DC_RBM	V/O	0...1H	0...1	1	[-]
DC including standardized vehicle operations for Rate-Based Monitoring					
STATE_DC_RBM	V	0...FFH	0...255	1	[-]
Status information about all conditions necessary to set LV_DC_RBM					
T_AST_RBM	V	0...FFFFH	0...6553.5	0.1	[s]
Cumulated time since engine start with Rate-Based Monitoring conditions					
T_VS_RBM	V	0...FFFFH	0...6553.5	0.1	[s]
Cumulated vehicle operation with vehicle speed >= C_T_VS_RBM with Rate-Based Monitoring conditions					
T_IS_RBM	V	0...FFFFH	0...6553.5	0.1	[s]
Continuous vehicle operation in idle >= C_T_IS_RBM with Rate-Based Monitoring conditions					
LV_DC_CDN_CST_RBM	O/V	0...1H	0...1	1	[-]
Cold start conditions for individual denominator calculation					
T_AST_CST_RBM	V	0...FFFFH	0...6553.5	0.1	[s]
Cumulative time since engine start (based on ambient temperature conditions)					
TAM_ST_DSL_CMN	V	0...FEH	-48...142.5	0	[°C]
Ambient temperature at engine start (used within ERRM only to handle GS/DS data types diversity)					

Output data detailed description:

STATE_CTR_RBM[NC_NR_DIAG_RBM] : Information data for CTR_COMP_RBM[NC_NR_DIAG_RBM] and CTR_CDN_RBM[NC_NR_DIAG_RBM] calculations.

bit 0: incrementation of monitor individual numerator done in this DC
bit 1: incrementation of monitor individual denominator done in this DC

Input data:

LV_DC	T_AST	STATE_RBM[NC_NR_DIAG_RBM]	LV_INH_DC_RBM
NC_NR_DIAG_RBM	T_IS	TAM_DSL_CMN	AMP
LV_IS	LV_ES	VS	PV
TCO_ST_DSL_CMN			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_MIN_RBM	1	0...FFFFH	0...5434	0.0829175	[hPa]
Minimum ambient pressure for Rate-Based Monitoring (relation to altitude)					
C_DLY_AST_RBM	1	0...FFFFH	0...6553.5	0.1	[s]
Time after start, necessary to increment the ignition cycle counter					
C_PV_IS_RBM	1	0...3FFH	0...99.90234	0.0976562	[%]
Minimum threshold to detect accelerator pedal released					
C_T_AST_CST_RBM	1	0...FFFFH	0...6553.5	0	[s]
Minimum comulative time since engine start for cold start condition detection					
C_T_AST_RBM	1	0...FFFFH	0...6553.5	0.1	[s]
Minimum trip length for Rate-Based Monitoring					
C_T_IS_RBM	1	0...FFFFH	0...6553.5	0.1	[s]
Minimum time of continuous idle operation for Rate-Based Monitoring					
C_T_VS_RBM	1	0...FFFFH	0...6553.5	0.1	[s]
Minimum time with VS_H_RES greater or equal than C_VS_THD_RBM for Rate-Based Monitoring					
C_TAM_MAX_CST_RBM	1	0...FEH	-48...142.5	0	[°C]
Maximum ambient temperature threshold for for RBM cold start condition					
C_TAM_MIN_CST_RBM	1	0...FEH	-48...142.5	0	[°C]
Minimum ambient temperature threshold for RBM cold start condition					
C_TAM_MIN_RBM	1	0...FEH	-48...142.5	0.75	[°C]
Minimum ambient temperature for Rate-Based Monitoring					
C_TAM_ST_HYS_CST_RBM	1	0...FEH	-48...142.5	0	[°C]
Ambient temperature hysteresis for RBM cold start condition					
C_TCO_ST_MAX_CST_RBM	1	0...FEH	-48...142.5	0	[°C]
Maximum engine start temperature threshold for RBM cold start condition					
C_TCO_ST_MIN_CST_RBM	1	0...FEH	-48...142.5	0	[°C]
Minimum engine start temperature threshold for RBM cold start condition					
C_VS_THD_IS_CDN_RBM	1	0...FFH	0...25.5	0.1	[km/h]
Vehicle speed threshold for continuous idle speed operation, for Rate-Based Monitoring					
C_VS_THD_RBM	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed threshold to start timer T_VS_RBM					

FUNCTION DESCRIPTION:

General information:

To gain more information about the monitoring performance under real world conditions, CARB required to implement software algorithms to track and report in-use performance for the following monitors in a standardized format: Catalyst, Oxygen sensor, Evaporating system, EGR system, VVT system and Secondary air system.

For this kind of statistic different counters are necessary:

- Ignition cycle counter

  Counter that indicates the number of ignition cycles a vehicle has experienced

- General denominator

  Measures the number of times a vehicle is operated under standardized conditions

- Monitor individual numerator

  Measures the number of driving cycles in which the monitor was done

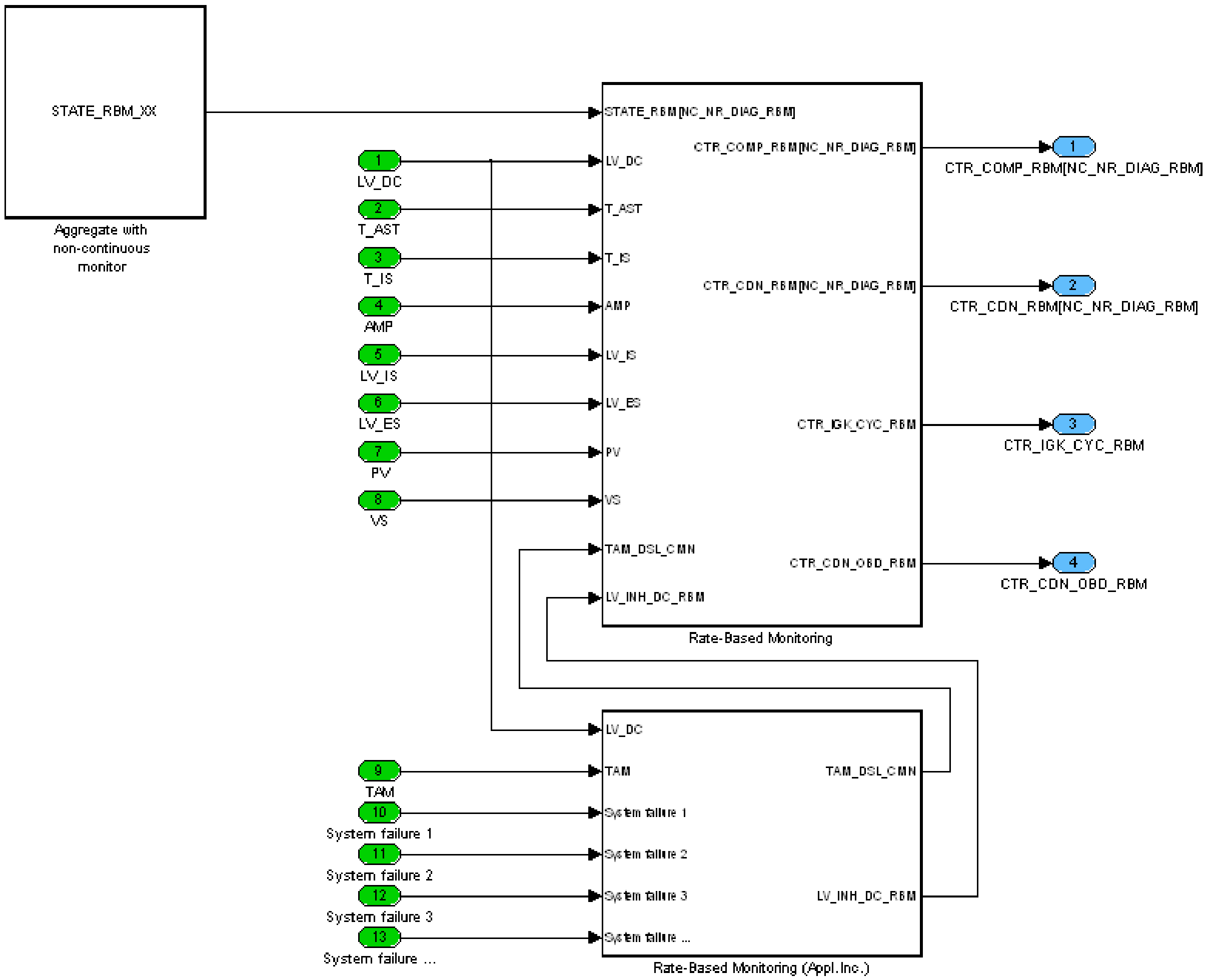
- Monitor individual denominator

  Counter that indicates the number of times a vehicle is operated under monitor individual conditions.

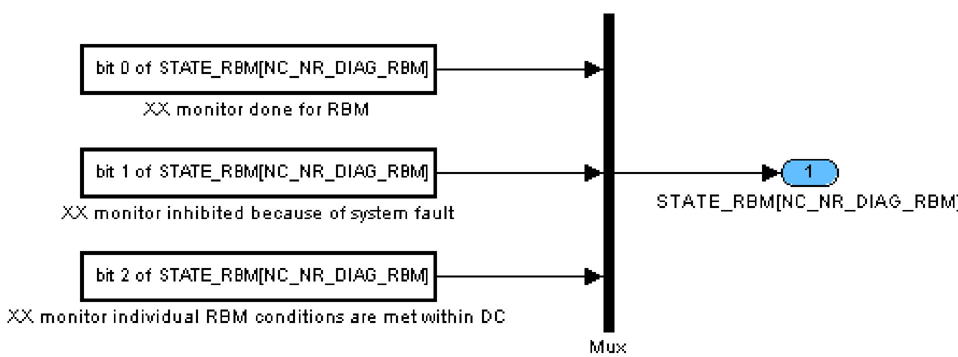
All these counters are set to zero only when a non-volatile memory reset occurs (e.g. reprogramming). They will not be initialized with zero on any other circumstances including when a scan-tool command to clear fault codes is received.

Signal flow diagram:

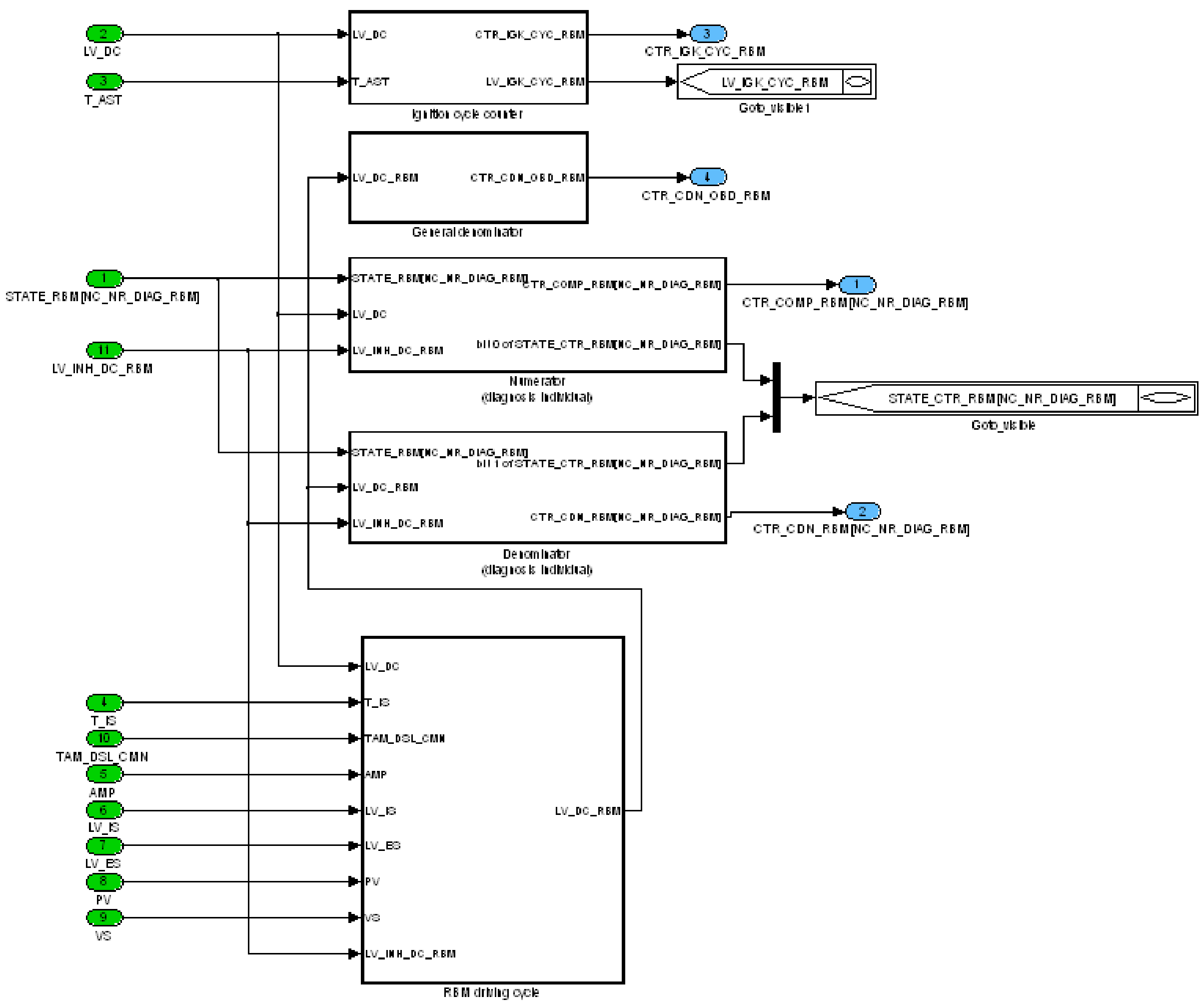
Overview :



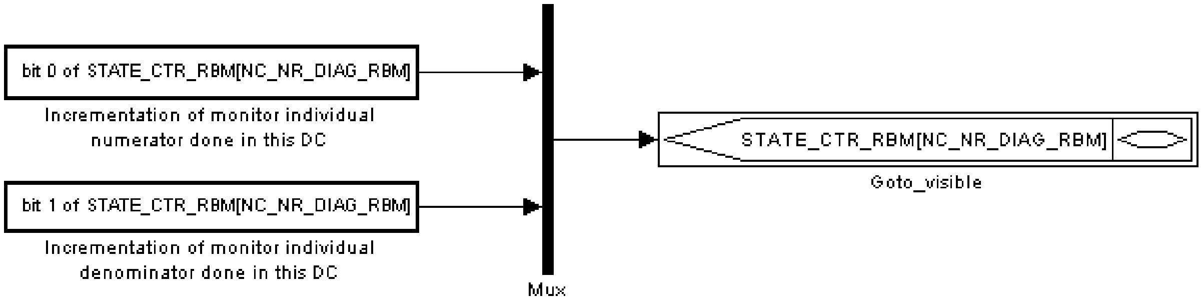
STATE_RBM[NC_NR_DIAG_RBM] definition :



Modules of Rate-Based Monitoring :



STATE_CTR_RBM[NC_NR_DIAG_RBM] definition:



B.58.1 Ignition cycle counter

Description:

The number of ignition cycles since the first ECU power up are counted within CTR_IGK_CYC_RBM. The counter is incremented each time a driving cycle with engine running for at least 2 seconds (±1s) (C_DLY_AST_RBM) is detected. This value is only initialized with 0 on saved RAM lost (or reprogramming). If CTR_IGK_CYC_RBM is incremented LV_IGK_CYC_RBM is set to 1 to indicate that counter is already incremented within this driving cycle.

If the counter reaches its maximum value, the value is set to 0 on the next calculation.

Application conditions:

Initialisation: at first ECU power up / on saved RAM lost (or reprogramming)

CTR_IGK_CYC_RBM = 0

at reset

CTR_IGK_CYC_RBM = restored from NVMY

at LV_DC 0 → 1 transition and ECU reset

LV_IGK_CYC_RBM = 0

Recurrence: 1 s

Activation: LV_DC = 1 **and** LV_IGK_CYC_RBM = 0

Formula section:

Quantity of driving cycles calculation:

If T_AST >= C_DLY_AST_RBM (time after start at least C_DLY_AST_RBM)

Then

If CTR_IGK_CYC_RBM = 65535 (maximum value of CTR_IGK_CYC_RBM)

Then

CTR_IGK_CYC_RBM = 0

Else

CTR_IGK_CYC_RBM = CTR_IGK_CYC_RBM + 1

Endif

LV_IGK_CYC_RBM = 1 (CTR_IGK_CYC_RBM incremented this DC)

Endif

B.58.2 General denominator calculation

Description:

The counter for the general denominator CTR_CDN_OBD_RBM is incremented each time a valid DC with standardized vehicle operation (LV_DC_RBM = 1) is recognized. The numbers of driving cycles are counted since the first ECU power up. This value is only initialized with 0 on saved RAM lost (or reprogramming).

If the counter reaches its maximum value, the value is set to 0 on the next calculation.

Application conditions:

Initialisation: at first ECU power up / on saved RAM lost (or reprogramming)

CTR_CDN_OBD_RBM = 0

at reset

CTR_CDN_OBD_RBM = restored from NVMY

Recurrence: -

Activation: LV_DC_RBM 0→1 transition

Formula section:

If CTR_CDN_OBD_RBM = 65535 (maximum value of CTR_CDN_OBD_RBM)

Then

CTR_CDN_OBD_RBM = 0

Else

General denominator calculation:

CTR_CDN_OBD_RBM = CTR_CDN_OBD_RBM + 1

Endif

B.58.3 Monitor individual numerator calculation

Description:

For all monitors requiring Rate-Based Monitoring (m = 0 to (NC_NR_DIAG_RBM – 1)) a numerator must be calculated. This numerator CTR_COMP_RBM[m] is incremented, if the conditions for the monitor were present for a sufficient time (bit 0 of STATE_RBM[m] = 1), but only one time within the driving cycle. This value is only initialized with 0 on saved RAM lost (or reprogramming)

If the counter CTR_CDN_RBM[m] is halved, because of maximum value limitation, the value CTR_COMP_RBM[m] is also divided by 2 (positive rounded).

Application conditions:

Initialisation: at first ECU power up / on saved RAM lost (or reprogramming)

CTR_COMP_RBM[m] = 0

at reset

CTR_COMP_RBM[m] = restored from NVMY

at LV_DC 0 →1 transition or ECU reset

STATE_CTR_RBM[m] = 0

with m = 0 to (NC_NR_DIAG_RBM – 1)

Recurrence: 10 s **or** LV_DC 1 → 0 transition

Activation: LV_INH_DC_RBM = 0 **and** LV_DC = 1

Formula section:

For m = 0 to (NC_NR_DIAG_RBM – 1)
all monitor requiring Rate-Based Monitoring
{ see specification "Rate-Based Monitoring (Appl. Inc.)" }
If bit 0 of STATE_CTR_RBM[m] = 0
(Incrementation of CTR_COMP_RBM[m] not yet done in this DC)

Then **If** bit 0 of STATE_RBM[m] = 1
(monitor is done for RBM)

and bit 1 of STATE_RBM[m] = 0
(monitor isn't inhibited because of a system fault)

Then CTR_COMP_RBM[m] = CTR_COMP_RBM[m] + 1

bit 0 of STATE_CTR_RBM[m] = 1
(Incrementation of CTR_COMP_RBM[m] done in DC)

Endif

Endif

Endfor

B.58.4 Monitor individual denominator calculation

Description:

In comparison to the counter CTR_IGK_CYC_RBM, counting the numbers of driving cycles (longer C_DLY_AST_RBM) since the first ECU power-up, the monitor individual denominator is counting the number of DC in which special vehicle/driving and ambient conditions are met individually for each monitor.

The counter CTR_CDN_RBM[m] is managed individually for each monitor (m = 0 to (NC_NR_DIAG_RBM – 1)), for which the monitoring of the real in-use performance is made. To increment the monitor denominator, some common conditions must be fulfilled (LV_DC_RBM = 1) but also monitor individual conditions. These monitor individual conditions are managed within the monitor function (bit 2 of STATE_RBM[m]). The value is only initialized with 0 on saved RAM lost (or reprogramming).

If the counter CTR_COMP_RBM[m] is halved, because of maximum value limitation, the value CTR_CDN_RBM[m] is also divided by 2 (positive rounded).

Application conditions:

Initialisation: at first ECU power up / on saved RAM lost (or reprogramming)

CTR_CDN_RBM[m] = 0

at reset

CTR_CDN_RBM[m] = restored from NVMY

with m = 0 to (NC_NR_DIAG_RBM – 1)

Recurrence: 10 s **or** LV_DC 1 → 0 transition (before LV_DC_RBM initialisation performed at LV_DC 1 → 0)

Activation: LV_INH_DC_RBM = 0 **and** LV_DC_RBM = 1

Formula section:

For m = 0 to (NC_NR_DIAG_RBM – 1)
all monitor requiring Rate-Based Monitoring
{ see specification "Rate-Based Monitoring (Appl. Inc.)" }
If bit 1 of STATE_CTR_RBM[m] = 0
(incrementation is not done yet)

Then **If** bit 1 of STATE_RBM[m] = 0
(monitor is not inhibit, because of a system fault)

and bit 2 of STATE_RBM[m] = 1
(monitor individual RBM conditions are met within this DC)

Then CTR_CDN_RBM[m] = CTR_CDN_RBM[m] +1

bit 1 of STATE_CTR_RBM[m] = 1
(incrementation is done within DC)

Endif
Endif
Endfor

B.58.5 Counter overflow treatment for monitor individual numerator and denominator

Description:

If either the numerator or denominator for a specific component reaches the maximum value (65535), both numbers shall be divided by two before either is incremented again to avoid overflow problems.

Application conditions:

Initialisation: -

Recurrence: -

Activation: LV_DC 0 → 1 transition

Formula section:

For m = 0 to (NC_NR_DIAG_RBM – 1)
all monitor requiring Rate-Based Monitoring
{ see specification "Rate-Based Monitoring (Appl. Inc.)" }
If CTR_CDN_RBM[m] = 65535 (maximum value of CTR_CDN_RBM[m])

or

CTR_COMP_RBM[m] = 65535 (maximum value of CTR_COMP_RBM[m])

Then CTR_CDN_RBM[m] = CTR_CDN_RBM[m] / 2

CTR_COMP_RBM[m] = CTR_COMP_RBM[m] / 2

Endif

Endfor

B.58.6 Standardized vehicle operation for Rate-Based Monitoring

<u>FUNCTION DESCRIPTION:</u>

<u>General information:</u>

For the calculation of the denominators for Rate-Based Monitoring, CARB standardized vehicle operations, which must be met to increment the denominator counter.

<u>Description:</u>

A special driving cycle (LV_DC_RBM) for Rate-Based Monitoring is defined, which is set, if the following standardized vehicle operations are fulfilled:

- Cumulated trip length (time since engine start) T_AST_RBM >= C_T_AST_RBM

  (CARB : 600 s)
- Cumulative vehicle operation >= C_VS_THD_RBM for minimum of C_T_VS_RBM

  (CARB : >= 40 km/h (25mph) for at least 300 s)
- Continuous vehicle operation in idle equal or longer than C_T_IS_RBM (accelerator pedal released and vehicle speed less or equal than 1 mph)

  (CARB : 30 s)

with

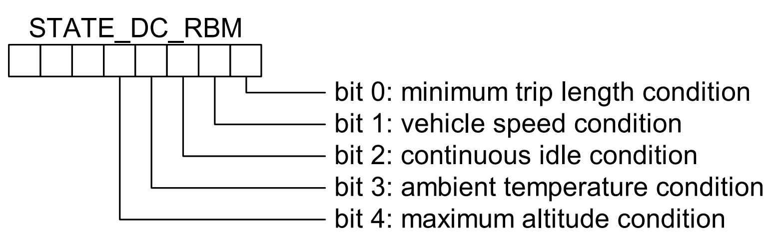
- altitude such as ambient pressure AMP > C_AMP_MIN_RBM

  (CARB : 8000 feet)
- ambient temperature (TAM_DSL_CMN) >= C_TAM_MIN_RBM

  (CARB : 20 °F)

Remark: to manage data type diversity between GS and DS divisions, TAM_DSL_CMN variable is updated with ambient temperature information every 100 ms.

Within STATE_DC_RBM the different conditions necessary to set LV_DC_RBM are visualized.



bit value = 0: condition not met yet; bit value = 1: condition met

Application conditions:

Initialisation: at LV_DC 0 → 1 transition **or** reset

LV_DC_RBM = 0

STATE_DC_RBM = 0H

T_AST_RBM = 0 s

T_VS_RBM = 0 s

T_IS_RBM = 0 s

at LV_DC 1 → 0 transition

after individual numerator/denominator computations:

LV_DC_RBM = 0

STATE_DC_RBM = 0H

Recurrence: 100 ms

Activation: LV_DC = 1

and

LV_DC_RBM = 0

and
LV_INH_DC_RBM = 0

Formula section:

Ambient temperature

If (1) TAM_DSL_CMN >= C_TAM_MIN_RBM (ambient temperature condition)

Then (1) bit 3 of STATE_DC_RBM = 1 (ambient temperature condition reached)

Else (1) bit 3 of STATE_DC_RBM = 0 (ambient temperature condition not reached)

Endif (1)

Maximum altitude

If (1) AMP > C_AMP_MIN_RBM (maximum altitude condition)

Then (1) bit 4 of STATE_DC_RBM = 1 (maximum altitude condition reached)

Else (1) bit 4 of STATE_DC_RBM = 0 (maximum altitude condition not reached)

Endif (1)

If (1) bit 3 and bit 4 of STATE_DC_RMB = 1

Then (1)

Minimum trip length

If (2) bit 0 of STATE_DC_RBM = 0 (minimum trip length not reached yet)

and

LV_ES = 0

Then (2)

T_AST_RBM = T_AST_RBM + 100 ms

If (3) T_AST_RBM = C_T_AST_RBM

Then (3)

bit 0 of STATE_DC_RBM = 1

Endif (3)

Endif (2)

Vehicle speed

If (2) bit 1 of STATE_DC_RBM = 0 (vehicle speed condition not met yet)

and

VS >= C_VS_THD_RBM

Then (2)

T_VS_RBM = T_VS_RBM + 100 ms

If (3) T_VS_RBM = C_T_VS_RBM

Then (3)

bit 1 of STATE_DC_RBM = 1

Endif (3)

Endif (2)

Continuous idle

If (2) bit 2 of STATE_DC_RBM = 0 (continuous idle cond. not met yet)

Then (2)

If (3) LV_IS = 1 (engine state = ‘idle’)

and

PV <= C_PV_IS_RBM (accelerator pedal released)

and

VS <= C_VS_THD_IS_CDN_RBM (CARB: 1 mile/h)

Then (3)

If (4) T_IS < T_IS_RBM (new idle phase detected)

Then (4)

T_IS_RBM = 0

Else (4)

T_IS_RBM = T_IS_RBM + 100 ms

If (5) T_IS_RBM = C_T_IS_RBM

Then (5)

bit 2 of STATE_DC_RBM = 1

Endif (5)

Endif (4)

Else (3)

T_IS_RBM = 0

Endif (3)

Endif (2)

Else (1)

T_IS_RBM = 0 (ambient temperature and maximum altitude conditions not met during the whole continuous idle speed condition)

Endif (1)

<u>Global information about DC including standardized vehicle operations</u>

If (1) STATE_DC_RBM = 1FH (all conditions are met)

Then (1) LV_DC_RBM = 1

Endif (1)

B.58.7 Cold start conditions detection

<u>FUNCTION DESCRIPTION:</u>

<u>General information:</u>

The individual monitor condition established for evaporative system denominator calculation (see §1968.2 - 4.3.2, section D) can be applied for other diagnoses (bit 2 calculation of STATE_RBM_xx) that require extended monitoring evaluation (e.g. coolant or intake air rationality).

<u>Description:</u>

Cold start conditions are set within the driving cycle when: (see §1968.2 - 4.3.2, section D)

- Cumulative time since engine start is greater than or equal to 600 seconds while at an ambient temperature of greater than or equal to 40 degrees Fahrenheit but less than or equal to 95 degrees Fahrenheit
- Engine cold start occurs with engine coolant temperature at engine start greater than or equal to 40 degrees Fahrenheit but less than or equal to 95 degrees Fahrenheit and less than or equal to 12 degrees Fahrenheit higher than ambient temperature at engine start.

Application conditions:

Initialisation: at RST system event **or** at LV_DC 0 → 1 transition

LV_DC_CDN_CST_RBM = 0

T_AST_CST_RBM = 0 s

TAM_ST_DSL_CMN = TAM_DSL_CMN

Recurrence: 100ms

Activation: LV_DC = 1

and

LV_DC_CDN_CST_RBM = 0

and

LV_INH_DC_RBM = 0

Dectivation: LV_DC = 0

or

LV_DC_CDN_CST_RBM = 1

or

LV_INH_DC_RBM = 1

Formula section:

If(1) C_TAM_MIN_CST_RBM <= TAM_DSL_CMN <= C_TAM_MAX_CST_RBM
(ambient temperature greater than or equal to 40 degrees Fahrenheit but less than or equal to 95 degrees Fahrenheit)

Then(1)

If(2) T_AST_CST_RBM >= C_T_AST_CST_RBM
(time since start is greater than or equal to 600 seconds)

Then(2)

T_AST_CST_RBM is frozen

If(3) C_TCO_ST_MIN_CST_RBM <= TCO_ST_DSL_CMN <= C_TCO_ST_MAX_CST_RBM
(engine coolant temperature at engine start is greater than or equal to 40 degrees but less than or equal to 95 degrees Fahrenheit)

and
TCO_ST_DSL_CMN <= (TAM_ST_DSL_CMN + C_TAM_ST_HYS_CST_RBM)
(less than or equal to 12 degrees Fahrenheit higher than ambient temperature at engine start)

Then(3)

LV_DC_CDN_CST_RBM = 1

Endif(3)

Else(2)

T_AST_CST_RBM = T_AST_CST_RBM + 100ms

Endif(2)

Else(1)

T_AST_CST_RBM is frozen

Endif(1)

B.59 Rate-Based Monitoring (Appl. Inc.)

B.59.1 Definition of diagnosis using RBM

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM[NC_NR_DIAG_RBM]	O	0 ... FH	0 ... 15	1	-
Monitor interface for the Rate-Based Monitoring statistics					

Input data:

STATE_RBM_CAT_DIAG[NC_CBK_EX_NR]	STATE_RBM_STK_LS_UP[1]	STATE_RBM_PUC_LS_UP[1]	STATE_RBM_TCO_PLAUS
STATE_RBM_TOIL_STUCK	STATE_RBM_TCO_STUCK	STATE_RBM_OBD_LSH_UP[NC_CBK_EX_NR]	STATE_RBM_OBD_LSH_DOWN[NC_CBK_EX_NR]
STATE_RBM_FRQ_LS_UP[NC_CBK_EX_NR]	STATE_RBM_SWT_LS_UP[NC_CBK_EX_NR]	STATE_RBM_TH	STATE_RBM_CAM_DYN_IVVT_IN_i
STATE_RBM_SWT_LS_DOWN[NC_CBK_EX_NR]	STATE_RBM_PUE_LS_DOWN[NC_CBK_EX_NR]	STATE_RBM_PLAUS_TIA[NC_SENS_NR_TIA]	STATE_RBM_DTP_NOISE
STATE_RBM_MEC_CPS	STATE_RBM_FUC_MISS_1	STATE_RBM_VAP_LEAK_1	STATE_RBM_TIA_DYN
STATE_RBM_TCO_STUCK_H	STATE_RBM_MAF_DIF_CH_INT	STATE_RBM_TCO_STUCK_RNG	STATE_RBM_CAM_STAT_IVVT_IN_i
STATE_RBM_CAM_STAT_IVVT_EX_i	STATE_RBM_CAM_DYN_IVVT_EX_i	STATE_RBM_PLAUS_OPEN_RCL	STATE_RBM_PLAUS_CLOSE_RCL
STATE_RBM_CAP_L_BAS	STATE_RBM_PLAUS_TAM	STATE_RBM_CAP_H	STATE_RBM_CAP_L
LV_DC	STATE_RBM_CAM_PAS_IVVT_IN_i	STATE_RBM_CAM_PAS_IVVT_EX_i	STATE_RBM_VIM_MEC_LONG
STATE_RBM_VIM_MEC_SHO			

FUNCTION DESCRIPTION:

Description:

Depending on the system configuration and the customer requirements, the diagnosis individual counters for Rate-Based Monitoring, the numerator and denominator, must be calculated for specific monitors.

Within this module the monitors using RBM statistics must be defined.

Application conditions:

Initialisation : *at ECU reset and LV_DC 0 → 1 transition :*

For i = 0 to (NC_NR_DIAG_RBM - 1) **do**

STATE_RBM[i] = 0

Endfor

Recurrence: 10 s **and** at LV_DC 1 → 0 transition

{ always executed prior to Rate-Based Monitoring main module }

Activation: -

Formula section:

	Interface byte	Group Name
0	STATE_RBM_CAT_DIAG[1]	CAT_1
1	0 (not used)	CAT_2
2	STATE_RBM_STK_LS_UP[1]	LS_UP_1
3	STATE_RBM_FRQ_LS_UP[1]	LS_UP_1
4	STATE_RBM_SWT_LS_UP[1]	LS_UP_1
5	0 (not used)	LS_UP_2
6	STATE_RBM_CAM_STAT_IVVT_IN_1	EGR_VVT
7	STATE_RBM_CAM_DYN_IVVT_IN_1	EGR_VVT
8	STATE_RBM_CAM_PAS_IVVT_IN_1	EGR_VVT
9	STATE_RBM_CAM_STAT_IVVT_EX_1	EGR_VVT
10	STATE_RBM_CAM_DYN_IVVT_EX_1	EGR_VVT
11	STATE_RBM_CAM_PAS_IVVT_EX_1	EGR_VVT
12	0 (not used)	SA
13	STATE_RBM_VAP_LEAK_1	EVAP
14	STATE_RBM_SWT_LS_DOWN[1]	LS_DOWN_1
15	STATE_RBM_PUC_LS_DOWN[1]	LS_DOWN_1
16	STATE_RBM_PUE_LS_DOWN[1]	LS_DOWN_1
17	0 (not used)	LS_DOWN_2
18	STATE_RBM_TCO_PLAUS	CUSTOMER
19	STATE_RBM_TOIL_STUCK	CUSTOMER
20	STATE_RBM_TCO_STUCK	CUSTOMER
21	STATE_RBM_PUC_LS_UP[1]	CUSTOMER
22	STATE_RBM_OBD_LSH_DOWN[1]	CUSTOMER
23	STATE_RBM_OBD_LSH_UP[1]	CUSTOMER
24	STATE_RBM_PLAUS_TIA_IM_CYL	CUSTOMER
25	STATE_RBM_DTP_NOISE	CUSTOMER
26	STATE_RBM_MEC_CPS	CUSTOMER
27	STATE_RBM_FUC_MISS_1	CUSTOMER
28	STATE_RBM_TH	CUSTOMER
29	STATE_RBM_TIA_DYN	CUSTOMER
30	STATE_RBM_TCO_STUCK_H	CUSTOMER
31	STATE_RBM_TCO_STUCK_RNG	CUSTOMER
32	STATE_RBM_MAF_DIF_CH_INT	CUSTOMER
33	STATE_RBM_PLAUS_OPEN_RCL	CUSTOMER
34	STATE_RBM_PLAUS_CLOSE_RCL	CUSTOMER

35	STATE_RBM_CAP_H	CUSTOMER
36	STATE_RBM_CAP_L	CUSTOMER
37	STATE_RBM_CAP_L_BAS	CUSTOMER
38	STATE_RBM_PLAUS_TAM	CUSTOMER
39	STATE_RBM_VIM_MEC_LONG	CUSTOMER
40	STATE_RBM_VIM_MEC_SHO	CUSTOMER
NC_NR_DIAG_RBM = 41	Number of monitors connected to rate-based monitoring	

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_DIAG_RBM	1	1 ... FFH	1 ... 255	1	-
Instance number of monitor using RBM statistics					

B.59.2 Inhibition of standardized vehicle operation or all numerators/denominators calculations

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DC_RBM	V/O	0 ... 1H	0 ... 1	1	-
Inhibition of standardized vehicle operation and all numerators/denominators, due to a system failure that influences the conditions required to set LV_DC_RBM, the conditions on the Secondary Air system, and the conditions for Evaporative System Monitor					

Input data:

LV_DC	LV_DC_RBM	LV_ERR_AMP	LV_ERR_VS
LV_ERR_TCO_PLAUS	LV_ERR_TCO_STUCK	LV_ERR_TCO_EL	LV_ERR_TCO_GRD
LV_ERR_TCO	LV_ERR_EL_TIA[NC_SENS_NR_TIA]	CTR_ERR_DYN_NR	

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT <PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With the value LV_INH_DC_RBM the inhibition of the standardized vehicle operation calculation and the disablement of all numerators and denominators are controlled.

Description:

The OBD II system shall disable all numerators and denominators calculations if a malfunction of any component used to determine if the following criteria (i.e., vehicle speed, ambient temperature, elevation, idle operation, engine cold start, time of operation, ...) are satisfied has been detected. Calculations shall resume as soon as the malfunction is no longer present.

- Criteria required to set LV_DC_RBM (standardized vehicle operation)

  - Cumulated trip length (time since engine start) T_AST_RBM ≥ C_T_AST_RBM (CARB : 600 s)
  - Cumulative vehicle operation ≥ C_VS_THD_RBM for minimum of C_T_VS_RBM (CARB : 40 km/h (25 mph) for at least 300 s)
  - Continuous vehicle operation in idle of C_T_IS_RBM or longer (accelerator pedal pedal released and vehicle speed less or equal than 1 mph) (CARB : 30 s)

with
  - Altitude such as ambient pressure AMP > C_AMP_MIN_RBM (CARB : 8000 feet)
  - Ambient temperature (TAM) ≥ C_TAM_MIN_RBM (CARB : 20 °F)

- Criteria required for Secondary Air system

  The Secondary Air system has been operated for a time greater than or equal to 10 s. Short trips and intrusive operations (end of line tests, repair tests, ...) shall not be included.

- Criteria required for Evaporative System Monitor

  - Cumulative time since engine start is greater or equal to 600 s while at an ambient temperature of greater than or equal to 40 °F but less than or equal to 95 °F.
  - Engine cold start occurs with engine coolant temperature at engine start greater than or equal to 40°F but less than or equal to 95°F and less than or equal to 12 °F higher than ambient temperature at engine start.

Application conditions:

Initialisation: *at ECU reset and LV_DC 0 → 1 transition :*

LV_INH_DC_RBM = 0

Recurrence: 1 s

Activation: LV_DC = 1

and

LV_INH_DC_RBM = 0

Formula section:

At LV_DC 0 → 1 transition :

The pending status of the following failures has to be checked only once :

LV_ERR_EL_TIA[NC_IDX_TIA_IM_CYL]
LV_ERR_VS
LV_ERR_TCO_PLAUS
LV_ERR_TCO_STUCK
LV_ERR_TCO_EL
LV_ERR_TCO_GRD

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While LV_INH_DC_RBM = 0 **do**

with each XX failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

LV_INH_DC_RBM = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s :

If LV_ERR_AMP = 1
or LV_ERR_EL_TIA[NC_IDX_TIA_IM_CYL] = 1
or LV_ERR_VS = 1
or LV_ERR_TCO = 1
Then LV_INH_DC_RBM = 1

Endif

B.59.3 Clear rate base monitoring statistics

Import actions:

ACTION_ERRM_ClearRbmStatistics (OUT < ResultClrInfo>)

Description:

For development conveniences, it is often useful to reinit the rate base monitoring statistics. This is possible by setting the calibration bit LC_RBM_CLR to 1.

Otherwise, like requested by CARB, RBM statistics must be reseted in case of ECU reprogramming (Application SW or Calibration data). During development phase RBM statistics shall not be reset at calibration reprogramming (using LC_RBM_NOT_CLR_DATA_FLASH = 1), but this must be disbaled for serial production (LC_RBM_NOT_CLR_DATA_FLASH = 0).

Application conditions:

Initialisation: -

Recurrence: 1s

Activation: 0 → 1 transition of LC_RBM_CLR

Formula section:

If LC_RBM_CLR = 1
Or ECU reprogramming of Application SW
Or (ECU reprogramming of Calibration data
And LC_RBM_NOT_CLR_DATA_FLASH = 0)
Then ACTION_ERRM_ClearRbmStatistics (OUT < ResultClrInfo>, SYNCHRONIZATION<CALL>)
Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_RBM_CLR	1	0...1h	0...1	1	-
reinit the rate base monitoring statistics at LC_RBM_CLR 0->1					
LC_RBM_NOT_CLR_DATA_FLASH	1	0...1h	0...1	1	-
No deletion of the rate base monitoring statistics at calibration reprogramming					

B.59.4 Ambient air temperature adaptation to handle GS/DS data types differences

Output data:

Name	Mode	Hex. limits	Phys. Limitslimits	Resol.	Unit
TAM_DSL_CMN	O	0...FEH	-48...142.5	0.75	[°C]
Ambient temperature (used within ERRM only to support GS/DS data types diversity)					

Input data:

TAM			

FUNCTION DESCRIPTION:

Description:

Some temperatures values can be delivered from Diesel or Gasoline system:

Ambient temperature TAM

As data types may be different, the following fomula section shall be used to allow adaptation to each environment.

Application conditions:

Initialisation: *at ECU reset*

TAM_DSL_CMN = TAM

Recurrence: 1 s

Activation: always

Formula section:

TAM_DSL_CMN = TAM

B.60 Failure Classes

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
WAL_CONF_XX	O	0...FFH	0...255	1	[-]
Lamp configuration for failure XX					
PRI_CONF_XX	O	0...7H	0...7	1	[-]
Priority configuration for failure XX					
CTR_DC_INC_XX	O	0...FFH	0...255	1	[-]
Increment of the driving cycle counter for failure XX					
CTR_DC_DEC_XX	O	0...FFH	0...255	1	[-]
Decrement of the driving cycle counter for failure XX					
CTR_DC_MAX_XX	O	0...FFH	0...255	1	[-]
Maximum value of the driving cycle counter for failure XX					

FUNCTION DESCRIPTION:

General information:

Because the diagnosis system becomes more and more complex, to avoid multiplication of calibrations and to help the tuning team for simplifying the calibration process, each failure is defined via classes of failure.

Each class defines the type of this failure as emission relevant or not, as MIL handling or not, as using driving cycle for MIL illumination, and so on.

The configuration of a failure is defined by WAL_CONF_XX, PRI_CONF_XX, CTR_DC_INC_XX, CTR_DC_DEC_XX and CTR_DC_MAX_XX.

Format of WAL_CONF_XX		
Bit	Logical value	Description
0	LC_WAL_1_ON	Enable WAL_1 (0 : Off / 1 : On) See details in “Lamp management” module.
1	LC_WAL_2_ON	Enable WAL_2 (0 : Off / 1 : On) See details in “Lamp management” module.
2	LC_MIL_ON	Enable MIL (0 : Off / 1 : On) See details in “Lamp management” module.
3	LC_MIL_FLL	Enable flash mode for MIL (0 : Off / 1 : On) See details in “Lamp management” module.
4	LC_OBD_ERR	Defines failure as a “CARB/EOBD failure” (emission relevant). (failure visible via scantool) 0 : failure is not considered as a “CARB/EOBD failure” 1 : failure is considered as a “CARB/EOBD failure”

Format of PRI_CONF_XX		
Bit	**Logical value**	**Description**
0	LC_ERR_PRI_1	This option defines the priority of the XX diagnosis.
1	LC_ERR_PRI_2	It is used for storage of failure and freeze frame.
2	LC_ERR_PRI_3	000: lowest priority for errors which are not significant for emission 011: highest priority for errors, which are not significant for emission 100: lowest priority for errors, which are significant for emission 111: highest priority which are significant for emission See details in “Priority rules” module. For emission relevant (EOBD, CARB) failure : LC_ERR_PRI_3 = 1 For non emission relevant failure : LC_ERR_PRI_3 = 0

B.60.1 Failure assignation

For each failure XX, a failure class is allocated. The failure class number makes this affectation, which is a calibration named C_ERR_CLAS_XX.

This table is defined in the “General Diagnosis Information”.

Diagnosis instance	Failure Class number Class A / Class B
Failure XX	C_ERR_CLAS_XX
Failure YY	C_ERR_CLAS_YY
...	...

The failure class number is a byte divided in two subclasses named class B (defined by the most significant quartet) and class A (defined by the less significant quartet).

Failure class number for failure XX

C_ERR_CLAS_XX =

(	Class B		)	(	Class A		)

B.60.2 Failure class definition

A failure class defines for a failure, which belongs to this failure class :

Part sub-class A :

- if the failure have impact on MIL illumination or not ;
- if the failure have impact on MIL blinking or not ;
- if the failure have impact on Warning Lamp 1 or not ;
- if the failure have impact on Warning lamp 2 or not ;
- if the failure is considered as a CARB/EOBD failure (emission relevant)
  (failure visible with scantool, failure can set OBD freeze frame for mode 2h)

Part sub-class B :

- the increment value for driving cycle counter management ;
- the maximum value of driving cycle counter ;
- the decrement value for driving cycle counter management ;
- the priority of the failure.

Each subclass of a failure class is composed of 4 bits, so 16 different values are possible for each subclass. For each subclass, some values are predefined and some values are reserved for improvement/validation and are managed by calibration.

Sub Class A definition : ID_ERR_CLAS_A_FMT (1 byte x 16)

Subclass A Value (i) in hex / Name	**Description**	**ID_ERR_CLAS_A_FMT [i]**
0 / NO_LAMP	*- Predefined value –* Failure without impact on any lamp	%0000 0000 (in binary)
1 / WAL_1_ON	*- Predefined value -* Failure with impact on WAL1 lamp	%0000 0001 (in binary)
2 / WAL_2_ON	*- Predefined value –* Failure with impact on WAL2 lamp	%0000 0010 (in binary)
3 / MIL_ON	*- Predefined value -* Failure with impact on Mil (on)	%0001 0100 (in binary)
4 / MIL_ON_FLL	*- Predefined value –* Failure with impact on Mil (on & blinking)	%0001 1100 (in binary)
5 / LAMP_CUS_1	*Free for customer definition*	*Free for customer definition*
6 / LAMP_CUS_2	*Free for customer definition*	*Free for customer definition*
7 / LAMP_CUS_3	*Free for customer definition*	*Free for customer definition*
8 / LAMP_CUS_4	*Free for customer definition*	*Free for customer definition*
9 / LAMP_CUS_5	*Free for customer definition*	*Free for customer definition*
A / LAMP_CUS_6	*Free for customer definition*	*Free for customer definition*
B / LAMP_CUS_7	*Free for customer definition*	*Free for customer definition*
C / LAMP_CUS_8	*Free for customer definition*	*Free for customer definition*
D / LAMP_CUS_9	*Free for customer definition*	*Free for customer definition*
E / LAMP_CUS_10	*Free for customer definition*	*Free for customer definition*
F / LAMP_CUS_11	*Free for customer definition*	*Free for customer definition*

Formula section:

For each failure XX allocated to the failure class number C_ERR_CLAS_XX :

WAL_CONF_XX = ID_ERR_CLAS_A_FMT [4 Less Significant Bit of C_ERR_CLAS_XX]

Sub Class B definition : ID_ERR_CLAS_B_FMT (4 bytes x 16)

		ID_ERR_CLAS_B_FMT			
Subclass B Value (i) in hex/ Name	**Description**	**[i] [0] DC inc**	**[i] [1] DC max**	**[i] [2] DC dec**	**[i] [3] Priority**
0 / NO_LAW_L	*- Predefined value –* Non Carb/EOBD failure with low priority	1	1	1	0
1 / NO_LAW_M	*- Predefined value -* Non Carb/EOBD failure with medium priority	1	1	1	1
2 / NO_LAW_H	*- Predefined value –* Non Carb/EOBD failure with high priority	1	1	1	2
3 / LAW_CC_L	*- Predefined value -* Carb/EOBD failure with low priority Comprehensive component	3	3	1	4
4 / LAW_L	*- Predefined value -* Carb/EOBD failure with low priority	3 / 2 *)	6	2	4
5 / LAW_M	*- Predefined value -* Carb/EOBD failure with medium priority	3 / 2 *)	6	2	5
6 / LAW_H	*- Predefined value -* Carb/EOBD failure with high priority	3 / 2 *)	6	2	6
7 / VALID	For validation	1	1	1	4
8 / NO_ERASE	*- Predefined value -* failure with high priority, not erasable by warm-up cycles.	3	3	0	6
9 / DC_CUS_1	*Free for customer definition*	-	-	-	-
A / DC_CUS_2	*Free for customer definition*	-	-	-	-
B / DC_CUS_3	*Free for customer definition*	-	-	-	-
C / DC_CUS_4	*Free for customer definition*	-	-	-	-
D / DC_CUS_5	*Free for customer definition*	-	-	-	-
E / DC_CUS_6	*Free for customer definition*	-	-	-	-
F / DC_CUS_7	*Free for customer definition*	-	-	-	-

Formula section:

For each failure XX allocated to the failure class number C_ERR_CLAS_XX :

CTR_DC_INC_XX = ID_ERR_CLAS_B_FMT[4 Most Significant Bit of C_ERR_CLAS_XX][0]

CTR_DC_MAX_XX = ID_ERR_CLAS_B_FMT[4 Most Significant Bit of C_ERR_CLAS_XX][1]

CTR_DC_DEC_XX = ID_ERR_CLAS_B_FMT[4 Most Significant Bit of C_ERR_CLAS_XX][2]

Bit 0,1,2 of PRI_CONF_XX = ID_ERR_CLAS_B_FMT [4 Most Significant Bit of C_ERR_CLAS_XX][3]

Remark: The increment of the DC counter (*) for the failures defined by the failure classes LAW_L, LAW_M and LAW_H depends on the required regulation (Carb/EOBD).

B.60.3 Example

To define the failure MIS_A (Misfiring A) as :

- a failure with high priority
- with only impact on MIL (On and Blinking)

Set the failure MIS_A in class number 64 (in hex) with C_ERR_CLAS_MIS_A = 64H.

That means :

- The value of sub class B is 6H => failure with high priority ;
- The value of sub class A is 4H => Failure with impact on MIL (On and blinking) ;

= > Failure is a "CARB/EOBD failure"

DC_INC_MIS_A = 3

DC_MAX_MIS_A = 6

DC_DEC_MIS_A = 2

WAL_CONF_MIS_A = %0001 1100 (in binary)

PRI_CONF_MIS_A = %0000 0110 (in binary)

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_ERR_CLAS_XX	1	0...FFH	0...255	1	[-]
Failure class number for failure XX					
ID_ERR_CLAS_A_FMT	16	0...FFH	0...255	1	[-]
LDPM_CLAS_A_ID_CLAS_FMY	16	0...FFFFH	0...65535	1	[-]
Sub Class A definition					
ID_ERR_CLAS_B_FMT	4*16	0...FFH	0...255	1	[-]
LDP_CLAS_B_ID_CLAS_FMY	4	0...FFFFH	0...65535	1	[-]
LDPM_CLAS_A_ID_CLAS_FMY	16	0...FFFFH	0...65535	1	[-]
Sub Class B definition					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ERR_PRI_H	1	0...7H	0...7	1	[-]
Failure with priority ≥ NC_ERR_PRI_H are considered as high priority failures					

B.60.4 Action for MIL relevance

Export actions:

ACTION_ERRM_*GetMILRelevant* (IN <PRM_IDX_ERR >, OUT <PRM_LV_MIL_RLV>)
This action is used to know if the failure is MIL relevant or not

Description for actions:

ACTION_ERRM_*GetMILRelevant* (<PRM_IDX_ERR >, <PRM_LV_MIL_RLV>)					
This actions returns: - 0 if the failure is not MIL relevant - 1 if the failure is MIL relevant					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Diagnostic failure instance					
PRM_LV_MIL_RLV	OUT	0...1H	0...1	1	[-]
Mil relevant failure					

Formula section:

PRM_LV_MIL_RLV = 0

If PRM_IDX_ERR ! = 0

Then

MIL relevant means enable MIL or enable flash mode for MIL and CARB/EOBD failure (visible via scantool)

If (bit 2 **or** bit 3 of WAL_CONF_XX = 1) **and** bit 4 of WAL_CONF_XX = 1

Then PRM_LV_MIL_RLV = 1

Endif

Endif

B.60.5 Failure Class Definition (Customer Specific)

In addition to the standard pre defined failure classes also customer specific failure classes can be defined (Sub class B: Value from 9h to Fh).

Sub Class B definition : ID_CLAS_B_FMT (4 bytes x 16) continued

		ID_CLAS_B_FMT			
Subclass B **Value (i) in hex/ Name**	**Description**	**[i] [0]** **DC inc**	**[i] [1]** **DC max**	**[i] [2]** **DC dec**	**[i] [3]** **Priority**
9 / DC_CUS_1	*Free for customer definition* Carb/EOBD failure with high priority (for debounce in 1 DC)	3	3	1	6
A / DC_CUS_2	*Free for customer definition*	-	-	-	-
B / DC_CUS_3	*Free for customer definition*	-	-	-	-
C / DC_CUS_4	*Free for customer definition*	-	-	-	-
D / DC_CUS_5	*Free for customer definition*	-	-	-	-
E / DC_CUS_6	*Free for customer definition*	-	-	-	-
F / DC_CUS_7	*Free for customer definition*	-	-	-	-

B.61 Lamp management

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_MIL	V/O/S	0...2H	0...2	1	[-]
State of MIL illumination commanded by error management (ON (1), OFF (0), MIL_FLL (2))					
STATE_WAL_1	V/O/S	0...1H	0...1	1	[-]
State of WAL_1 illumination commanded by error management (ON (1), OFF (0))					
STATE_WAL_2	V/O/S	0...1H	0...1	1	[-]
State of WAL_2 illumination commanded by error management (ON (1), OFF (0))					
LV_MIL	V/O	0...1H	0...1	1	[-]
Boolean indicating MIL physical output (0 : off / 1 : on)					
LV_WAL_1	V/O	0...1H	0...1	1	[-]
Boolean indicating WAL_1 physical output (0 : off / 1 : on)					
LV_WAL_2	V/O	0...1H	0...1	1	[-]
Boolean indicating WAL_2 physical output (0 : off / 1 : on)					
LV_WAL_ST	V/O	0...1H	0...1	1	[-]
Boolean indicating Pre-drive Check mode (0 : off / 1 : on)					
DLY_MIL_READY	V	0...FFFFH	0...32767.5	0.5	[s]
Delay time to unable MIL blinking during pre-drive check mode					
LV_MIL_FLL_READY	V/O	0...1H	0...1	1	[-]
Boolean indicating MIL shall blink during pre-drive check mode phase 2 (0 : not blink / 1 : blink)					
T_WAL_ST	V	0...FFH	0...127.5	0.5	[s]
Timer for pre-drive check duration					

Input data:

LV_ES	SYM_CYL_MIS_A	LV_WAL_1_ACT_REQ	LV_WAL_2_ACT_REQ
LV_MIS_STATE_A	LV_DC_MAX[NC_NR_ERR_DYN]	INH_IV_MIS	WAL_CONF_XX
LV_IGK	LV_MIL_ACT_REQ	LV_ERR_XX	STATE_READY_OBD_1
STATE_READY_OBD_2	LV_ST		

FUNCTION DESCRIPTION:

General information:

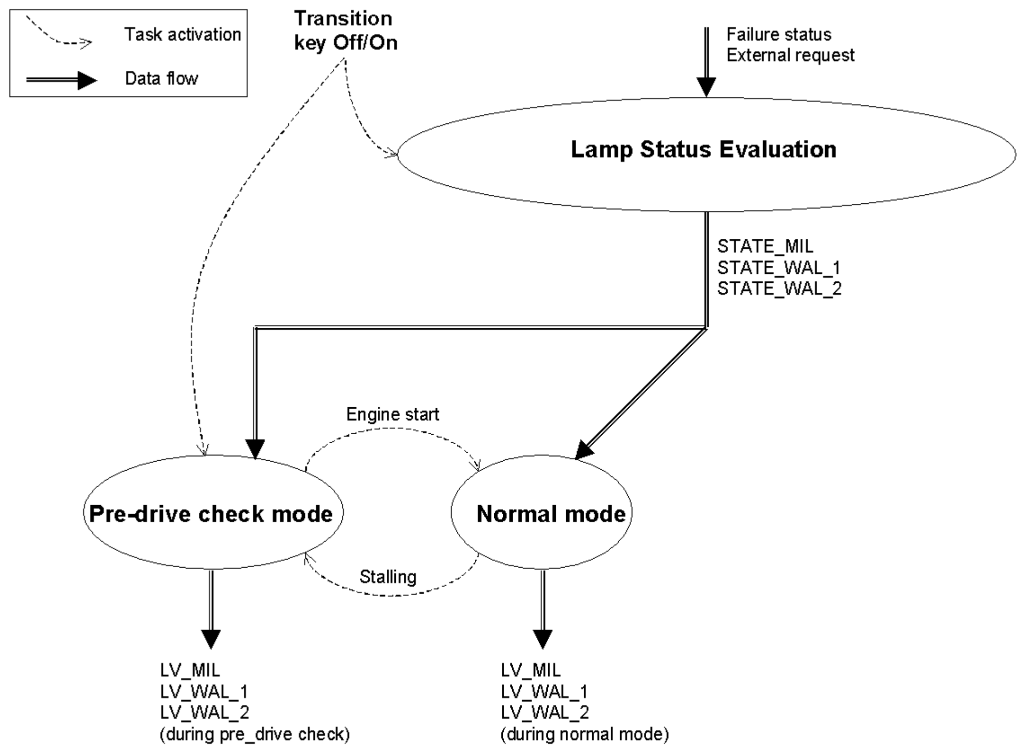
It is possible to manage up to three different lamps. They are called:

- MIL: Malfunction Indicator Lamp; this lamp is used to inform the vehicle driver in case of failures which can affect emissions.
- WAL_1: Warning Lamp 1 and WAL_2: Warning Lamp 2; these lamps are used to inform the vehicle driver in case of others failures.

For each lamp, a task evaluated the state of MIL, WAL_1 and WAL_2. These states STATE_MIL, STATE_WAL_1 and STATE_WAL_2 are respectively what error management commanded to the lamps regard to failure status.

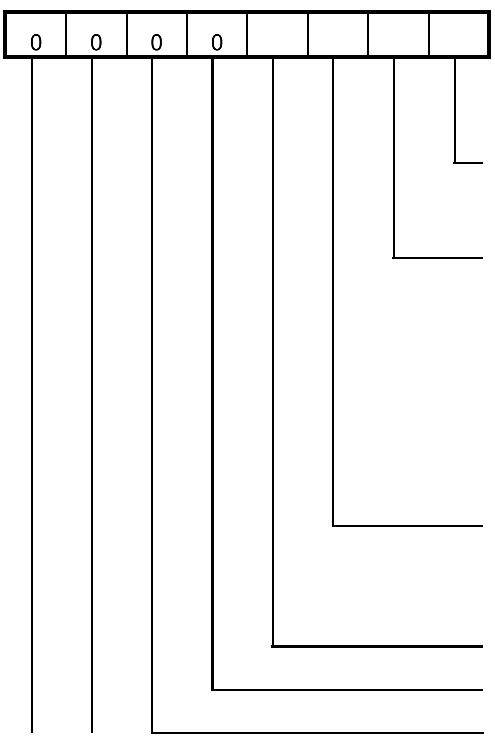
Then, the real physical state of each lamp (LV_MIL for MIL, LV_WAL_1 for WAL_1 and LV_WAL_2 for WAL_2) is build related to the mode in which they are. Lamps behavior is based on two modes:

- A predrive check mode between transition LV_IGK 0->1 and engine start; this mode is used to check the correct lamps working.
- A normal mode for the rest of the time.



A configuration byte is defined for each lamp: C_CONF_XXX (with XXX = MIL C_CONF_MIL, WAL_1 or WAL_2, C_CONF_WAL_1 C_CONF_WAL_2).

It is then possible to inhibit a lamp illumination, to allow or not external request for lamp illumination and to link lamp illumination each other's.



bit 0 of C_CONF_XXX : Global lamp XXX inhibition bit
0 : lamp XXX illumination is authorized
1 : lamp XXX is never illuminated except by an external request for lamp XXX

bit 1 of C_CONF_XXX : Lamp XXX request enable bit
0 : lamp XXX request input is never taken into account
1 : lamp XXX request input is taken into account

For MIL :
bit 2 of C_CONF_XXX: Special behavior for the Misfiring A failure in case of cylinder cut off.
0 : Without management of cylinder cut off
1 : With management of cylinder cut off
For WAL_1 and WAL_2 :
bit 2 of C_CONF_XXX : Illumination of MIL instead of lamp XXX
0 : no links between lamp XXX and MIL
1 : MIL is illuminated if illumination of lamp XXX is requested

bit 3 of C_CONF_XXX :
Inhibition of Predrive check for lamp XXX
0: Predrive check for lamp XXX enabled
1: Predrive check for lamp XXX disabled

Not used

Not used

Not used

Not used

Impact of each failure on lamp is defined in WAL_CONF_XX byte:

- bit 0 : if set, the present failure may caused a WAL_1 illumination ;
- bit 1 : if set, the present failure may caused a WAL_2 illumination ;
- bit 2 : if set, the failure may caused a MIL illumination when driving cycle counter reaches maximum ;
- bit 3: if set, the present failure may cause a MIL blinking.

B.61.1 Lamp Status Evaluation

Description:

Management of WAL_1 and WAL_2

Behavior of WAL_1 and WAL_2 are identical. WAL_1 and WAL_2 may be illuminated (state is ON) or not (state is OFF). When the global lamp inhibition bit is not set, the lamp is requested to illuminate:

- if at least one failure is present (LV_ERR_XX) with the corresponding bit of WAL_CONF_XX (bit 0 for WAL_1, bit 1 for WAL_2) in enable position and this failure is stored in 2nd layer memory ;
- or in case of allowed lamp request.

Management of MIL

MIL internal management is based on 3 states:

- STATE_MIL = OFF (MIL is requested continuously off)
- STATE_MIL = ON (MIL is requested continuously on)
- STATE_MIL = MIL_FLL (MIL is requested blinking at 1 Hz)

When the global MIL inhibition bit is not set, the MIL is requested to illuminate (ON):

- if at least one failure has reached its driving cycle maximum value and is unequal to zero (LV_DC_MAX[IDX]) with the corresponding bit of WAL_CONF_XX (bit 2) in enable position and this failure is stored in 2nd layer memory ;
- or in case of allowed lamp request ;
- or WAL_1 is requested to be illuminated and WAL_1 is link with MIL ;
- or WAL_2 is requested to be illuminated and WAL_2 is link with MIL ;

When the global MIL inhibition bit is not set, the MIL is requested to be blinking at 1hz (MIL_FLL):

- if at least one failure is present (LV_ERR_XX) with the corresponding bit of WAL_CONF_XX (bit 3) in enable position and this failure is stored in 2nd layer memory ;

A special calculation can be enabled for Misfire A failure with the management of cylinder shut off (bit 2 of C_CONF_MIL):

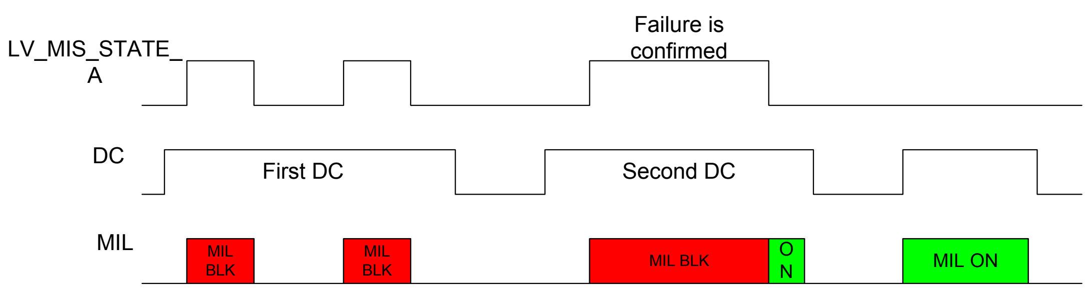
- bit 2 of C_CONF_MIL = 0: When a misfire A failure (LV_MIS_STATE_A) is present, the MIL is blinking.
- bit 2 of C_CONF_MIL = 1: If Misfire A failure (LV_MIS_STATE_A) is present with cylinder cut off, the MIL is blinking only for a calibratable time (C_DLY_INH_IV_MIS). After this time, the MIL is ON again.

Beware: The “blinking mode” is priority mode compared to “continuous on mode”.

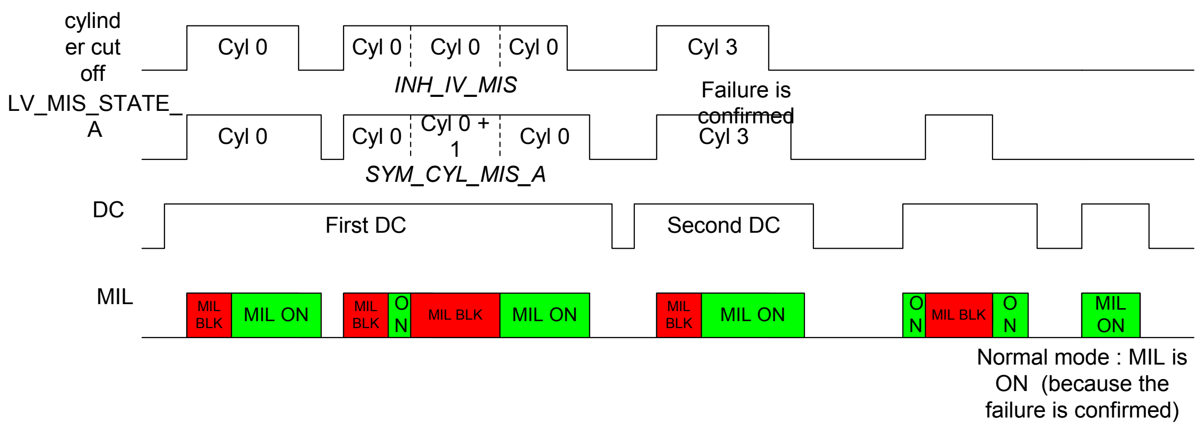
Signal flow diagram:

Special behavior for Misfire A failure with bit LC_MIL_INH_IV_MIS of C_CONF_MIL:

Behavior of Misfiring failure MIS_A without management of cylinder cut off (LC_MIL_INH_IV_MIS = 0)



Behavior of Misfiring A failure with management of cylinder cut off (LC_MIL_INH_IV_MIS = 1) :



Application conditions:

Initialization: -

Recurrence: 500 ms

Activation: -

Formula section:

If bit 0 of C_CONF_WAL_1 = 1 (lamp WAL_1 illumination not authorized)
Then STATE_WAL_1 = OFF (lamp WAL_1 is off)
Else **If** at least one failure in 2nd layer memory has:
(LV_ERR_XX and bit 0 of WAL_CONF_XX) = 1
(lamp WAL_1 illumination by a failure)
Then STATE_WAL_1 = ON (lamp WAL_1 is on)
Else STATE_WAL_1 = OFF (lamp WAL_1 is off)
Endif
Endif

If bit 0 of C_CONF_WAL_2 = 1 (lamp WAL_2 illumination not authorized)
Then STATE_WAL_2 = OFF (lamp WAL_2 is off)
Else **If** at least one failure in 2nd layer memory has:
(LV_ERR_XX and bit 1 of WAL_CONF_XX) = 1
(lamp WAL_2 illumination by a failure)
Then STATE_WAL_2 = ON (lamp WAL_2 is on)
Else STATE_WAL_2 = OFF (lamp WAL_2 is off)
Endif
Endif

If bit 0 of C_CONF_MIL = 1 (MIL illumination not authorized)

Then STATE_MIL = OFF (MIL is requested continuously off)

Else **If** at least one failure in 2nd layer memory has:

(LV_DC_MAX[IDX] and bit 2 of WAL_CONF_XX) = 1

(at least one failure request to illuminate the MIL)

or (STATE_WAL_1 = ON and bit 2 of C_CONF_WAL_1 = 1)

(a failure illuminates WAL_1 and WAL1 is linked with MIL)

or (STATE_WAL_2 = ON and bit 2 of C_CONF_WAL_2 = 1)

(a failure illuminates WAL_2 and WAL_2 is linked with MIL)

Then STATE_MIL = ON (MIL is requested continuously on)

Else STATE_MIL = OFF (MIL is requested continuously off)

Endif

If LV_MIS_STATE_A = 1

(CARB A misfire failure criterion)

then **if** bit 2 of C_CONF_MIL = 0

(Without management of cylinder shut off for the misfiring A)

then STATE_MIL = MIL_FLL

(MIL is requested blinking at 1Hz)

else (Management of cylinder shut off for the misfiring A)

If (any cylinder bit within SYM_CYL_MIS_A 0 -> 1

and corresponding bit within INH_IV_MIS is set to 1)

Then STATE_MIL = MIL_FLL only during C_DLY_INH_IV_MIS

After C_DLY_INH_IV_MIS, STATE_MIL = ON

mode = ‘blink mode during a delay + continuous ON’

Endif

if any cylinder bit within SYM_CYL_MIS_A 0 -> 1

and corresponding bit within INH_IV_MIS is set to 0

then mode = ‘blink mode’

Endif

If mode = ‘blink mode’

Then **if** any cylinder bit SYM_CYL_MIS_A is set

And

corresponding bit within INH_IV_MIS remains to 0

(no transition 1-> 0 of the corresponding bit during the failure)

Then mode = ‘blink mode’

Else mode =‘blink mode during a delay + continuous ON’

```
                        Endif
                Endif
                If      mode = 'blink mode'
                Then    STATE_MIL = MIL_FLL
                        (MIL is requested blinking at 1Hz)
                else    if      C_DLY_INH_IV_MIS isn't achieved
                        Then    STATE_MIL = MIL_FLL
                                (MIL is requested blinking at 1Hz)
                        Else    STATE_MIL = ON
                                (MIL is requested continuously on)
                        Endif
                Endif
        Endif
Endif
If      at least one failure in 2nd layer memory has :
        (LV_ERR_XX and bit 3 of WAL_CONF_XX) = 1
        (at least one failure request to blink the MIL)
Then    STATE_MIL = MIL_FLL
        (MIL is requested blinking at 1Hz)
Endif
```

Endif

Remark:

For the misfiring A failure (treated as a particular case), the Bit 3 of WAL_CONF_XX mustn't be set to 1 by calibration (only bit 2 for continuous MIL illumination).

B.61.2 Lamp mode

B.61.2.1 Predrive Check mode

Description:

Predrive check mode is used to enable a visual check of the correct lamp working. Additionally it can be used to display readiness status as an option (MIL blinking).

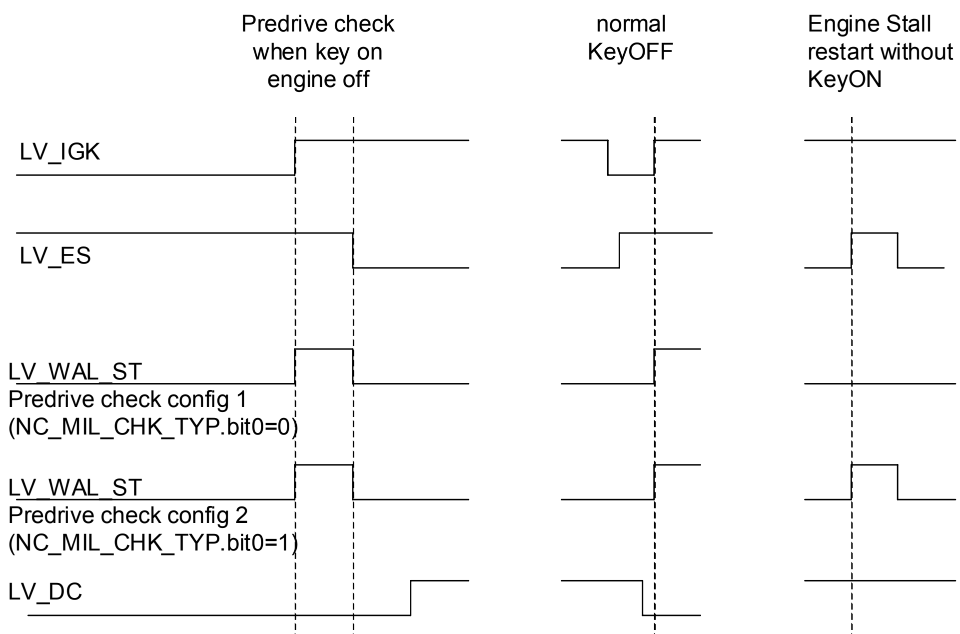
Nevertheless a failure may also illuminate a lamp. Illumination of lamp during pre-drive check mode is performed only when :

- the global lamp inhibition bit is not set (bit 0 of C_CONF_XXX = 1) and the lamp individual predrive check inhibition is not set (bit 3 of C_CONF_XXX = 1)
- or error management commands a lamp

A) Configuration of predrive check behaviour in case of stalling event:

with NC_MIL_CHK_TYP (bit 0)

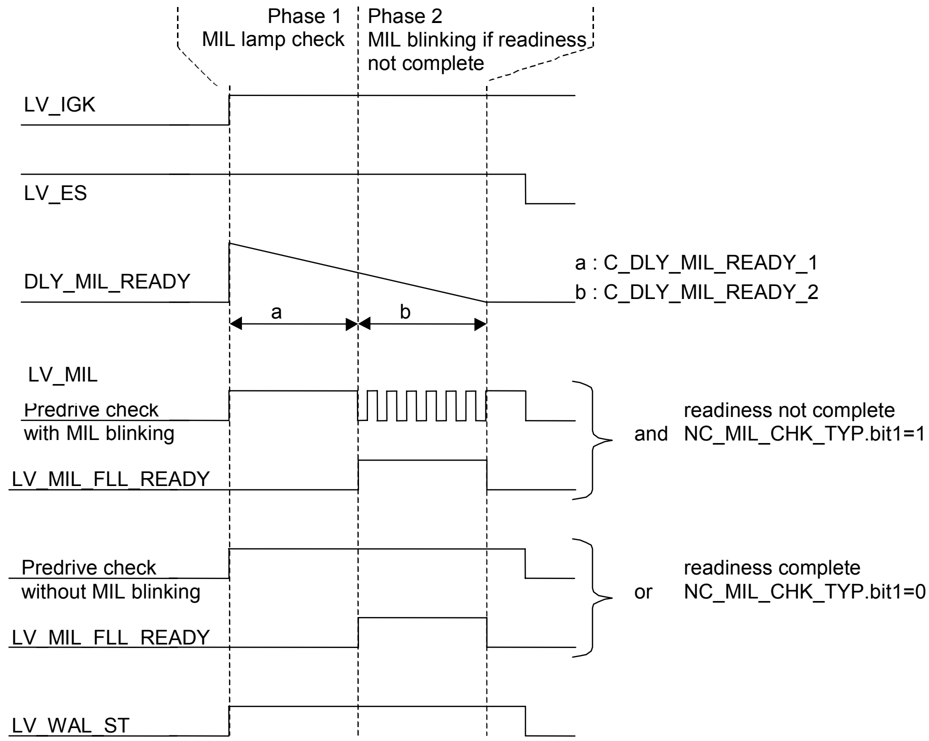
Predrive check mode phase can be defined in 2 ways:
- between key on transition LV_IGK 0->1 and engine start (bit 0 of NC_MIL_CHK_TYP=0)
- or between key on transition LV_IGK 0->1 and engine start, plus between an engine stall event and engine start (bit 0 of NC_MIL_CHK_TYP=1)



B) Configuration of MIL blinking readiness status functionality:

with NC_MIL_CHK_TYP (bit 1)

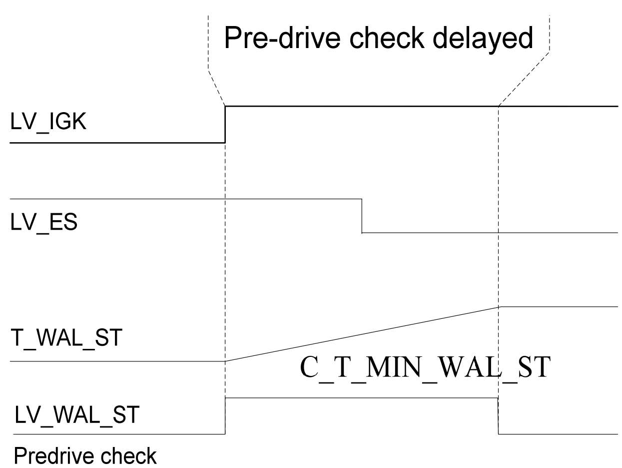
During predrive check mode, you can display the status of readiness codes through the MIL warning lamp : the MIL is blinking if readiness not complete.
This strategy is part of predrive check and does not affect the MIL status (STATE_MIL).



C) Management of minimum time of predrive check duration

-> only relevant after engine cranking (LV_ES 1->0)

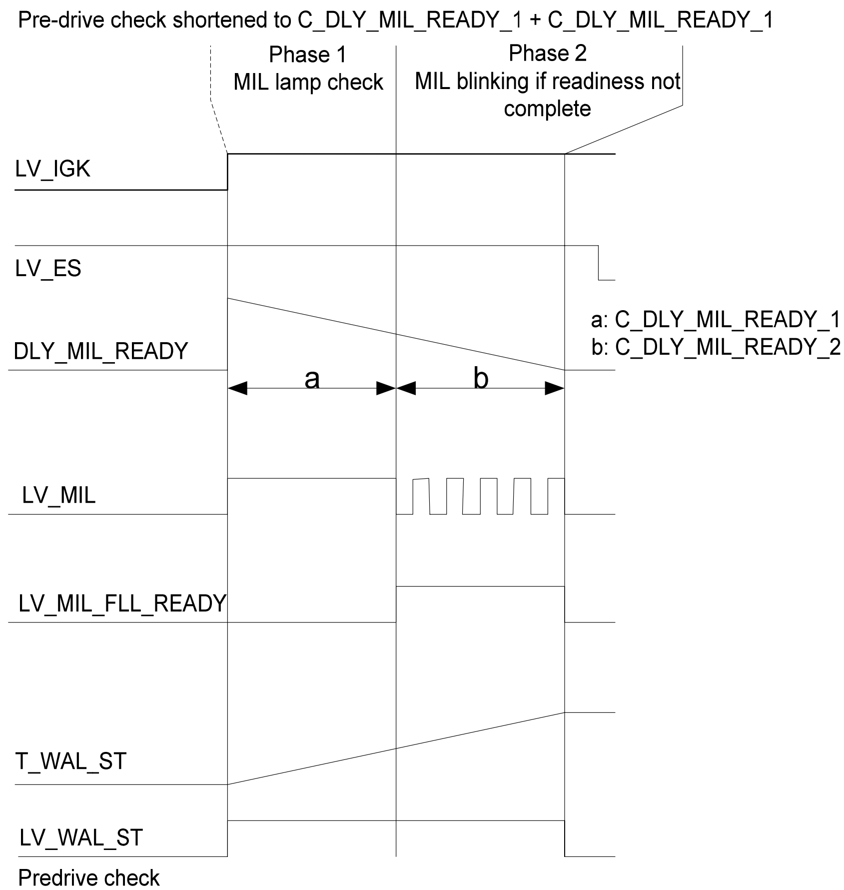
Wih the calibration C_T_MIN_WAL_ST it's possible to set a minimum time for the predrive check duration independant of engine cranking (LV_ES 1->0). This gives the possibility to illuminate the Mil for predrive check (e.g. in case of a quick start).



D) Management of maximum time of predrive check duration

-> only relevant before engine cranking (LV_ES = 0)

With the calibration LC_T_MAX_WAL_ST_ENA, it's possible to activated the maximum time of predrive check functionality. If this functionality is activated (LC_T_MAX_WAL_ST_ENA = 1) the maximum predrive check duration before engine cranking is C_DLY_MIL_READY_1 + C_DLY_MIL_READY_2. This gives the possibility to finish the predrive check before engine cranking e.g. in case of a long key on engine stop phase.



Application conditions:

Initialization: At LV_IGK transition 0 to 1
T_WAL_ST = 0 **and** LV_MIL_FLL_READY=0

At LV_ES transition 0 to 1
T_WAL_ST = 0

At LV_IGK transition 1 to 0 or if reset
LV_WAL_ST = 0 (Pre-drive check mode deactivated)

Recurrence: 500 ms and at LV_IGK 0 -> 1 transition

Activation: (pre-drive check enable between “stalling event/engine start”)

bit 0 of NC_MIL_CHK_TYP=1

and **[(**LV_IGK = 1 **and** LV_ES = 1**)**

or **(**LV_IGK = 1 **and** LV_ST = 1

and C_T_ MIN_WAL_ST >0 **and**

T_WAL_ST < C_T_ MIN_WAL_ST**)]**

and (T_WAL_ST < C_DLY_READY_1 + C_DLY_READY_2

or LC_T_MAX_WAL_ST_ENA = 0)

and (bit 0 of C_CONF_WAL_1 = 0

or bit 0 of C_CONF_WAL_2 = 0

or bit 0 of C_CONF_MIL = 0)

or

(pre-drive check disable between “stalling event/engine start”)

bit 0 of NC_MIL_CHK_TYP=0
and transition LV_IGK 0 ->1 **and** LV_ES = 1

and (T_WAL_ST < C_DLY_READY_1 + C_DLY_READY_2

or LC_T_MAX_WAL_ST_ENA = 0)

and (bit 0 of C_CONF_WAL_1 = 0

or bit 0 of C_CONF_WAL_2 = 0

or bit 0 of C_CONF_MIL = 0)

Deactivation: **(**LV_ES = 0 (engine not stopped)

and T_WAL_ST >= C_T_ MIN_WAL_ST**)**

(minimum time of predrive check reached)

or

((LV_ES = 1 (engine stopped)

and T_WAL_ST >= C_DLY_READY_1 + C_DLY_READY_2

and LC_T_MAX_WAL_ST_ENA = 1**)**

(maximum time of predrive check reached)

or LV_IGK=0 (or key off)

Formula section:

LV_WAL_ST=1 during activation of this formula section.
LV_WAL_ST=0 and LV_MIL_FLL_READY=0 during deactivation of this formula section (end of predrive check).

T_WAL_ST = T_WAL_ST + 0.5 s

```
If    bit 1 of NC_MIL_CHK_TYP = 1                                  (if readiness display enable)
Then

      If        transition of LV_WAL_ST from 0 to 1            (on beginning of predrive check)
      Then                                                              (initialize time delay)
                Initialize Delay time DLY_MIL_READY with
                C_DLY_MIL_READY_1+C_DLY_MIL_READY_2
      Endif
      DLY_MIL_READY = DLY_MIL_READY - 0.5 s

                                                   (check if delay is within the timing window)

      If              DLY_MIL_READY < C_DLY_MIL_READY_2
                And   DLY_MIL_READY > 0
      Then
                                                                                        (checl

           If         (bit 4 to 7 of STATE_READY_OBD_1=0)
                And   (STATE_READY_OBD_2=0)

           Then
                LV_MIL_FLL_READY=0                     (if "complete" then MIL does not blink)
           Else
                LV_MIL_FLL_READY=1                 (if "not complete" then MIL lamp shall blink)
           Endif

      Else
                LV_MIL_FLL_READY = 0             (the readiness status phase is finished)
      Endif

Endif

If    bit 0 of C_CONF_MIL = 0                   (MIL not inhibit)
and   bit 3 of C_CONF_MIL = 0                   (predrive check for MIL enable)
and   LV_MIL_FLL_READY = 1                      (readiness not complete)
Then
      LV_MIL shall blink at 1Hz
Endif
```

("**predrive check**": **switch on warning lamps**)

If bit 0 of C_CONF_WAL_1 = 0 (WAL_1 not inhibit)
and bit 3 of C_CONF_WAL_1 = 0 (predrive check for WAL_1 enabled)
Then LV_WAL_1 = 1
Endif
If bit 0 of C_CONF_WAL_2 = 0 (WAL_2 not inhibit)
and bit 3 of C_CONF_WAL_2 = 0 (predrive check for WAL_2 enabled)
Then LV_WAL_2 = 1
Endif
If bit 0 of C_CONF_MIL = 0 (MIL not inhibit)
and bit 3 of C_CONF_MIL = 0 (predrive check for MIL enabled)
and LV_MIL_FLL_READY = 0 (MIL not blinking readiness status)
Then LV_MIL = 1
Endif

B.61.2.2 Normal mode

The normal mode is defined as "not to be in pre-drive check mode".

B.61.2.3 All modes

Description:

The following treatment is done in pre-drive check mode and in normal mode.

Application conditions:

Initialization: at transition LV_IGK 1->0 or if reset
LV_MIL = 0 (MIL is off)
LV_WAL_1 = 0 (WAL_1 is off)
LV_WAL_2 = 0 (WAL_2 is off)

Recurrence: 500 ms and at LV_IGK 0 -> 1 transition

Activation: LV_IGK=1

Formula section:

If STATE_MIL = MIL_FLL
Then LV_MIL blink at 1Hz
Else If (STATE_MIL = ON **or** (LV_MIL_ACT_REQ = 1 **and** bit 1 of C_CONF_MIL = 1))
(allowed MIL request)
AND (LV_MIL_FLL_READY=0 **or** bit 3 of C_CONF_MIL = 1)
(case MIL blinking during the pre-drive check: readiness not completed)
Then LV_MIL = 1

Else If we are in normal mode **or** bit 3 of C_CONF_MIL = 1
(predrive check not authorized for MIL)
Then LV_MIL = 0
Endif
Endif
Endif

If STATE_WAL_1 = ON **or** (LV_WAL_1_ACT_REQ = 1 **and** bit 1 of C_CONF_WAL_1 = 1)
(allowed lamp WAL_1 request)

Then LV_WAL_1 = 1
Else If we are in normal mode **or** bit 3 of C_CONF_WAL_1 = 1
(pre drivecheck not authorized for WAL_1)
Then LV_WAL_1 = 0
Endif
Endif

If STATE_WAL_2 = ON **or** (LV_WAL_2_ACT_REQ = 1 **and** bit 1 of C_CONF_WAL_2 = 1)
(allowed lamp WAL_2 request)

Then LV_WAL_2 = 1
Else If we are in normal mode **or** bit 3 of C_CONF_WAL_2 = 1
(pre drivecheck not authorized for WAL_2)

Then LV_WAL_2 = 0
Endif
Endif

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_MIL_CHK_TYP	-	0...3H	0...3	1	[-]
Bit 0 = 0 : disable pre-drive check between "stalling event/engine start"Bit 0 = 1 : enable pre-drive check between "stalling event/engine start"Bit 1 = 0 : disable readiness status display during pre-drive check (MIL blinking)Bit 1 = 1 : enable readiness					

Configuration data detailed description:

Bit 0 = 0 : disable pre-drive check between “stalling event/engine start”
Bit 0 = 1 : enable pre-drive check between “stalling event/engine start”
Bit 1 = 0 : disable readiness status display during pre-drive check (MIL blinking)
Bit 1 = 1 : enable readiness
Typical value bit 0 = 0, bit 1 = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CONF_MIL	1	0...FFH	0...255	1	[-]
Configuration of MIL					
C_CONF_WAL_1	1	0...FFH	0...255	1	[-]
Configuration of warning lamp 1 (WAL_1)					
C_CONF_WAL_2	1	0...FFH	0...255	1	[-]
Configuration of warning lamp 2 (WAL_2)					
C_DLY_MIL_READY_1	1	0...FFH	0...127.5	0.5	[s]
Delay to start MIL blinking for readiness display during pre-drive checkShall be in between 15 and 20 sec. Typical value 20sec					
C_DLY_MIL_READY_2	1	0...FFH	0...127.5	0.5	[s]
MIL blinking duration for readiness display during pre-drive checkShall be in between 5 and 10 sec. Typical value 10sec					
C_DLY_INH_IV_MIS	1	0...FFH	0...127.5	0.5	[s]
Delay for MIL blinking in case of misfiring present failure with cylinder cut off					
C_T_MIN_WAL_ST	1	0...FFH	0...127.5	0.5	[s]
Minimum time of pre-drive check duration					
LC_T_MAX_WAL_ST_ENA	1	0...1H	0...1	1	[-]
Activation of maximum time of pre-drive check					

Calibration data detailed description:

C_DLY_MIL_READY_1: Delay to start MIL blinking for readiness display during pre-drive check. Shall be tuned between 15 s and 20 s. Typical value 20 s.

C_DLY_MIL_READY_2: MIL blinking duration for readiness display during pre-drive check. Shall be tuned between 5 s and 10 s. Typical value 10 s.

LC_T_MAX_WAL_ST_ENA:
= 1: maximum time of predrive check with LV_ES = 1 is C_DLY_MIL_READY_1 + C_DLY_MIL_READY_2

= 0: maximum time of predrive check with LV_ES = 1 is unlimited

B.62 Lamp Management (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_WAL_1_ACT_REQ	V/O	0...1H	0...1	1	[-]
Request of WAL 1 activation by others					
LV_WAL_2_ACT_REQ	V/O	0...1H	0...1	1	[-]
Request of WAL 2 activation by others					

B.62.1 External request management

FUNCTION DESCRIPTION:

General information:

With below formula :

- No direct MIL or warning lamp (WAL_1, WAL_2) request is used.
- All lamps requests (connected to the final output variables LV_MIL and LV_WAL_1 and LV_WAL_2) result from diagnostic or safety routines are treated by the driving cycle management of the dynamic error memory.

Application conditions:

Activation: -

Deactivation: -

Initialization: At ECU reset:

LV_WAL_1_ACT_REQ = 0

LV_WAL_2_ACT_REQ = 0

Recurrence: -

Formula section:

B.62.2 Connection between lamps and infrastructure

FUNCTION DESCRIPTION:

General information:

Purpose of this module is to make the connection between lamps management dataflow and CAN inter-system. This strategy controls warning lamps through CAN inter-system. Warning lamps comprise MIL(Malfunction Lamp Indicator) and customer specific warning lamps 1 and 2.

Signal flow diagram:

-

Description:

Purpose of this strategy is to switch on and off lamps according LV_MIL, LV_WAL_1, LV_WAL_2 state of warning lamps.

Application conditions:

Initialisation:

Recurrence: 500ms

Activation: at ECU reset

Deactivation: -

Formula section:

-

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_LAMP	1	0...3H	0...3	1	[-]
Quantity of warning lamps					

B.63 Priority rules

Input data:

PRI_CONF_XX	WAL_CONF_XX	LV_ERR_DISA[NC_NR_ERR_DYN]	LV_DC_MAX[NC_NR_ERR_DYN]
LV_ERR_CFM[NC_NR_ERR_DYN]	NC_ERR_PRI_H		

Export actions:

ACTION_ERRM_PrioRule (OUT <RESP>, OUT <IDX>)
This action performs priority rules, when failure memory is full, to say if the new failure is accepted or not.

FUNCTION DESCRIPTION:

General information:

The dynamic error management module calls this module in case of present failure occurrence when the 2nd layer memory is full.

It defines criteria to store or not this failure in the 2nd layer memory (dynamic memory). Only one failure occurrence is stored.

Results of this module is either:

- "storage of new failure not possible"
- or ("storage of new failure possible" ; index of failure to delete in 2nd layer memory)

In case of new failure entry whereas the dynamic memory is full (number of stored failure = NC_NR_ERR_DYN), different criteria are applied to prioritize the failure:

- A new failure is more important than a disappeared failure
- Failure priority as defined in PRI_CONF_XX ;
  - Chronological failure order

    If NLC_OLD_ERR_PRI = 1: old failure has priority regards of new failure;

    (Failure i older than failure j means failure i has entered 2nd layer memory before failure j)

    If NLC_OLD_ERR_PRI = 0: new failure has priority regards of old failure;

    (Failure i newer than failure j means failure i has entered 2nd layer memory after failure j)

Additional features of this functionality:

- MIL can never be switched off
- Failure setting the OBD freeze frame (e.g. used for diagnostic communication with scantool / mode 2h) cannot be erased.
- The "increased priority" can be enabled with the configuration NLC_INC_ERR_PRI. This priority is a final rule, which is used, if it's not possible to enter a failure with any of the other possibilities. Then it's possible (only for failure with priority >= 4) to store the failure, if a confirmed failure not of the highest priority (priority < NC_ERR_PRI_H), which is not illumination the MIL anymore, can erased instead.

If the new failure entry is not accepted because of the priority, this failure will not be stored in the 2nd layer memory but limp home is activated, thus the freeze frame is not memorized and also no lamp management for this failure is possible.

Description:

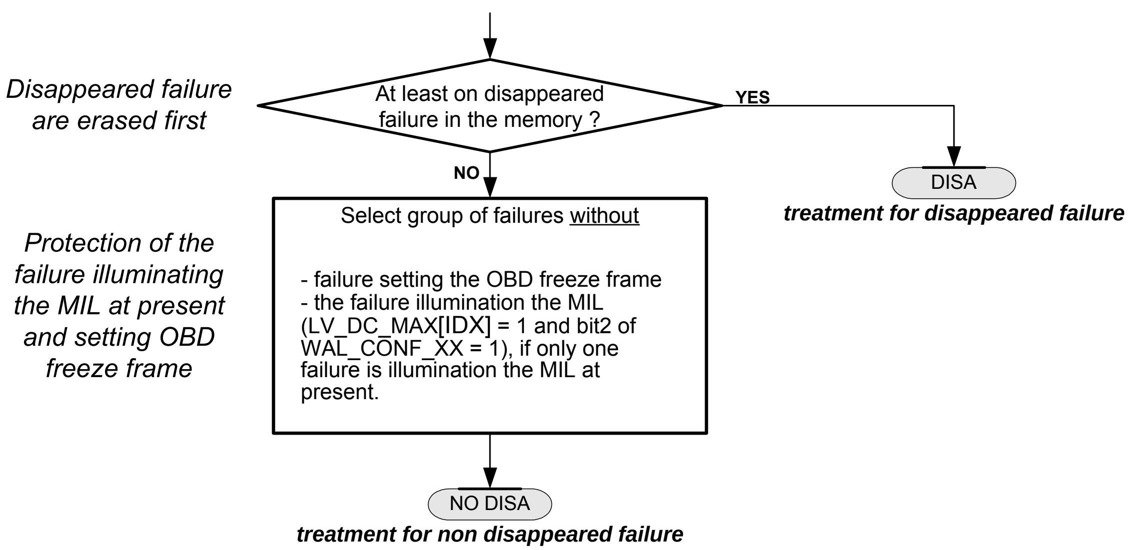
Syntax :	ACTION_ERRM_PrioRule (OUT <RESP>, OUT <IDX>)	
Parameter (in) :	-	
Parameter (out) :	RESP	OK if storage is possible; NOK if storage is not possible
	IDX	Index of failure in 2nd layer memory to delete
Short description :	This action performs priority rules, when failure memory is full, to say if the new failure is accepted or not.	

Application conditions:

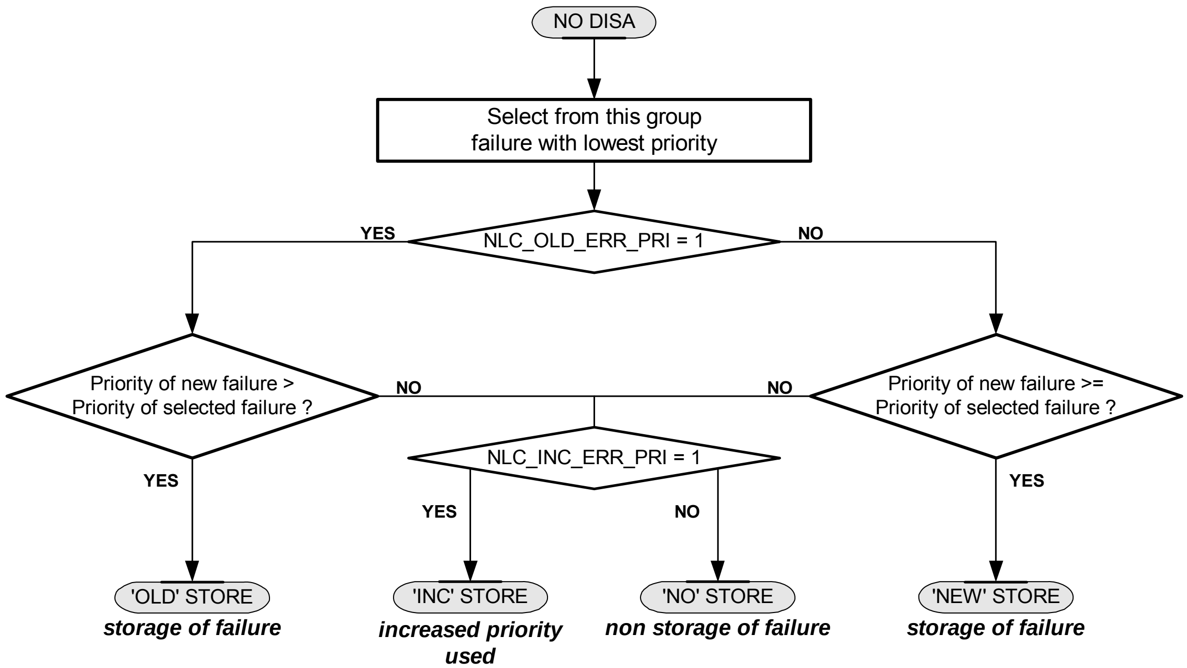
Activation:	-
Recurrence:	-
Deactivation:	at action request

Formula section:

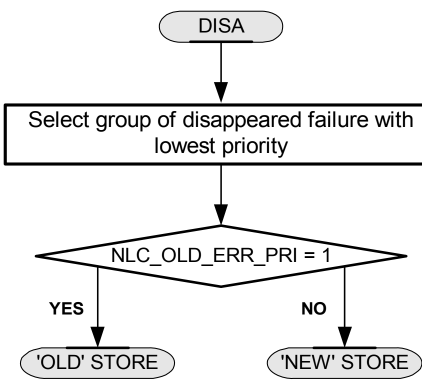
ACTION_ERRM_PrioRule (RESP, IDX) :



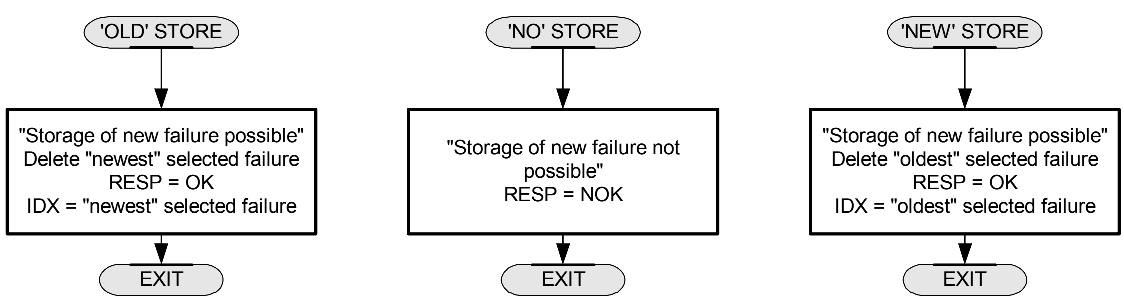
Treatment for non disappeared failure:



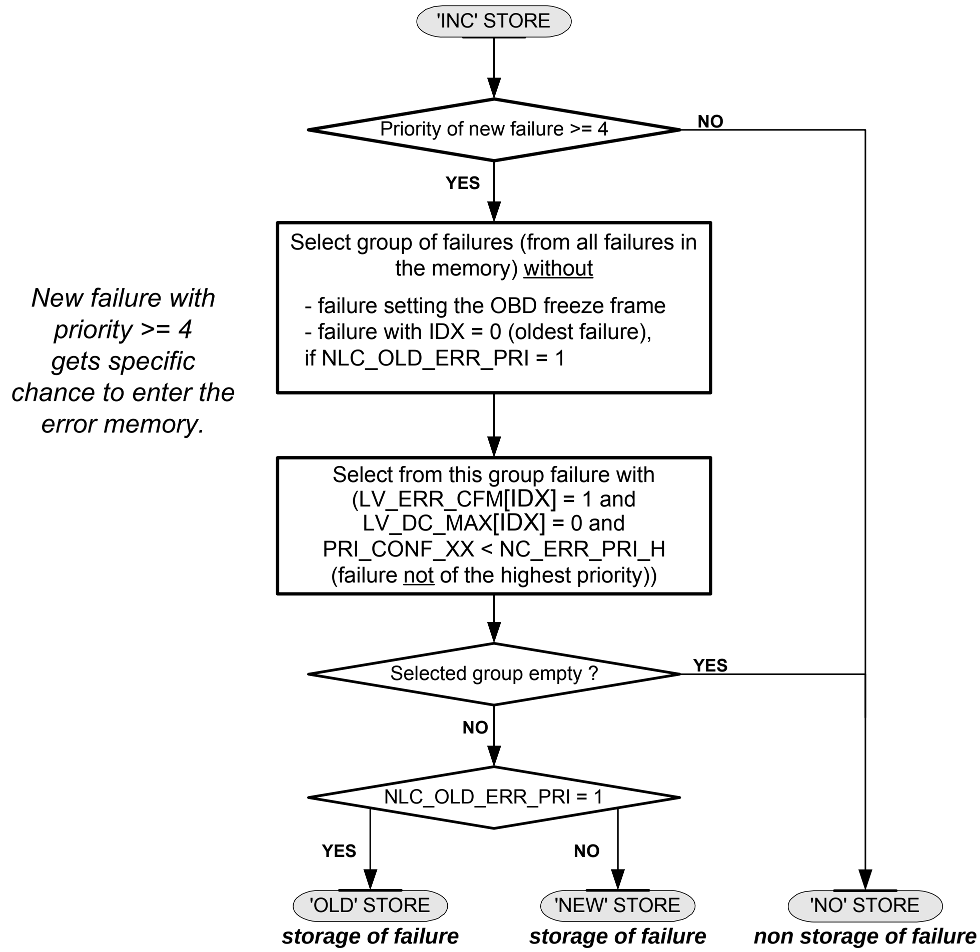
Treatment for disappeared failure:



Storage / non storage of failure:



Increased priority (with NLC_INC_ERR_PRI = 1):



Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_OLD_ERR_PRI	1	0...1H	0...1	1	[-]
Selection of old failure priority face to new failure					
NLC_INC_ERR_PRI	1	0...1H	0...1	1	[-]
Enable/disable increased failure priority					

Configuration data detailed description:

NLC_OLD_ERR_PRI : Set this bit to 1 to give priority to old failure regarding new failure.
Set this bit to 0 to give priority to new failure regarding old failure.

NLC_INC_ERR_PRI : Set this bit to 1 to enable increased failure priority.
Set this bit to 0 to disable increased failure priority.

B.64 Readiness codes

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_READY_XX	V/O/S	0...1H	0...1	1	[-]
Readiness flag related to diagnosis XX					

Input data:

LV_END_DIAG_XX	LV_ERR_DC[NC_NR_ERR_DYN]	LV_DC_MAX[NC_NR_ERR_DYN]	LV_DC
LV_ERR_MEM_XX	LV_EOL_OBD_DC	LV_ERR_XX	

Export actions:

ACTION_ERRM_InitReadiness ( )
This action initialise all the readiness flag

FUNCTION DESCRIPTION:

General information:

Readiness flag allows to know if a full diagnostic check (not in EOL phase) has been done or not (minimum number of checks necessary for MIL illumination if failure is present).

A readiness flag is defined for each diagnosis (LV_READY_XX). Based on these flags, readiness code status (2 bytes STATE_READY_OBD_1 and STATE_READY_OBD_2) are generated in the Communication Interface (API) file for Carb mode.

Readiness flags are mainly used by after market service and inspections maintenance. When after market achieves a repair, the system should be controlled before to give back the car to its owner.

Because some failure need a long time (and may be 2 DC) to be diagnosed, the repair operator need an as quick as possible information meaning the repairs is OK.

The readiness flag indicates when it is set to 0 that :

- the diagnosis related to failure XX is done since the last “clear DTC” service received.
- and if a failure XX is present, MIL would be illuminated.

For after market time optimisation, in case of no failure at all, the readiness flag is set to 0.

The readiness flag indicates when it is set to 1 that :

- the diagnosis related to failure XX is not done since the last “clear DTC” service received.
- or if a failure XX is present, MIL is not still illuminated (may takes many driving cycle).

After failure erase, all bits associated to supported functions are set to 1.

When the system has executed enough driving cycles for determining if MIL should be switch On or not, all the readiness flags are in 0 state. At this occurrence, failures which may have switch on the MIL could be considered as repaired if MIL is Off.

B.64.1 Readiness flag set to 0

Description:

The readiness flag LV_READY_XX is set to 0 when enough driving cycle has been done without failure XX.

Application conditions:

Initialization: On saved ram lost

LV_READY_XX = 1

Recurrence: at the recurrence of activation condition

Activation: at transition LV_DC 0->1
or at transition LV_END_DIAG_XX 0->1

Formula section:

```
If    LV_READY_XX = 1                              (diag XX not still ready)
      and LV_DC = 1                                (driving cycle valid)
            and LV_END_DIAG_XX = 1                       (diag done)
            and LV_EOL_OBD_DC = 0                        (not in EOL tests)
Then        If    LV_ERR_MEM_XX = 1                (failure in 2nd layer memory)
            Then  If    LV_ERR_DC[IDX] = 0         (no failure present this DC)
                  Then  LV_READY_XX = 0            (diag XX is ready)
                  Else  If    LV_DC_MAX[IDX] = 1
                        Then  LV_READY_XX = 0      (diag XX is ready)
                        Endif
                  Endif
            Else  If    LV_ERR_XX = 0
                  Then  LV_READY_XX = 0            (diag XX is ready)
                  Endif
            Endif
Endif
```

B.64.2 Readiness flag reinitialisation (set to 1)

Description:

Syntax : ACTION_ERRM_InitReadiness ( )

Parameter(in) : No parameter

Parameter(out) : No parameter

Short description :

The readiness flag LV_READY_XX is reinitialized (set to 1) upon communication tool request via the action ACTION_ERRM_InitReadiness ( ).

This reinitilisation is always done when all failure are erased from failure memory.

Application conditions:

Initialization: -

Recurrence: -

Activation: at action request

Formula section:

For every failure : LV_READY_XX = 1

B.64.3 Readiness code information update

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_READY_OBD_1	V/O/S	0...FFH	0...255	1	[-]
Readiness code completion status 1					
STATE_READY_OBD_2	V/O/S	0...FFH	0...255	1	[-]
Readiness code completion status 2					
STATE_READY_OBD_3	V/O/S	0...FFH	0...255	1	[-]
Readiness code completion status 3					

Input data:

LV_DC			

Import actions:

ACTION_ERRM_ReadReadinessCode(INOUT<ReadinessCode>, OUT<ResultReadinessCode>)
ACTION_ERRM_ReadReadinessCodCus(INOUT<ReadinessCode>, OUT<ResultReadinessCode>)

Description:

Purpose of this paragraph is to update readiness code completion status. This update shall be done :

- at end of driving cycle, for readiness status display,
- on action call, for customer tool and Scan Tool coherency with STATE_READY_OBD_X.

Important remark: each time ACTION_ERRM_ReadReadinessCode is called Readiness codes completion status (STATE_READY_OBD_1 and STATE_READY_OBD_2) are calculated.

Application conditions:

Activation: at LV_DC 1→ 0 transition

Initialization: on non-volatil memory reset/lost **or** on failure erase service received

STATE_READY_OBD_1 = FFh,

STATE_READY_OBD_2 = FFh
{FFh value indicates that readiness code is not yet computed}

Recurrence: none, executed single time on activation.

Formula section:

{ Action call to update STATE_READY_OBD_1 and STATE_READY_OBD_2 }

Call ACTION_ERRM_ReadReadinessCode(INOUT<STATE_READY_OBD_1, STATE_READY_OBD_2>, OUT<ResultReadinessCode>)

Application conditions:

Activation : at LV_DC 1→ 0 transition

Initialization: on non-volatil memory reset/lost

or

on failure erase service received
STATE_READY_OBD_3 = FFh
{ *FFh value permits to indicate that readiness is not yet computed* }

Recurrence : none, executed single time on activation.

Formula section:

{ Action call to update STATE_READY_OBD_3 }

Call ACTION_ERRM_ReadReadinessCodCus(INOUT<STATE_READY_OBD_3>, OUT<ResultReadinessCode>)

B.65 Similar conditions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_SCDN[NC_NR_WIN_SCDN]	V/S	0...3FH	0...63	1	[-]
Similar condition status					
CTR_SCDN_EQU_DC[NC_NR_WIN_SCDN]	V/S	0...FFH	0...255	1	[-]
Counter of non occurrence of similar condition					
N_MAX_SCDN_EQU[NC_NR_WIN_SCDN]	V/S	0...1FE0H	0...8160	1	[rpm]
maximal engine speed of similar conditions window					
N_MIN_SCDN_EQU[NC_NR_WIN_SCDN]	V/S	0...1FE0H	0...8160	1	[rpm]
minimal engine speed of similar conditions window					
LOAD_MAX_SCDN_EQU[NC_NR_WIN_SCDN]	V/S	0...FFH	0...99.60937	0.390625	[%]
maximal load of similar conditions window					
LOAD_MIN_SCDN_EQU[NC_NR_WIN_SCDN]	V/S	0...FFH	0...99.60937	0.390625	[%]
minimal load of similar conditions window					
LV_WUP_SCDN_EQU[NC_NR_WIN_SCDN]	V/S	0...1H	0...1	1	[-]
Warm-up status for similar conditions for OBDII continuous error					
CTR_SCDN_EQU[NC_NR_WIN_SCDN]	V	0...FFFFH	0...65535	1	[-]
Number of times N/LOAD is within SCDN window					
CTR_SCDN_SUM[NC_NR_WIN_SCDN]	V	0...FFFFH	0...65535	1	[-]
Number of times the diagnosis condition are set					

STATE_SCDN[k] : Similar conditions status for diagnostic instance k

bit 0: 0: SCDN will be recorded (updated) at next failure occurrence
1: SCDN are recorded
bit 1: Recognition of similar conditions with failure detection this DC
bit 2: Recognition of similar condition without failure/exceedance detection this DC
It’s also set after C_CTR_SCDN_EQU_DC_MAX driving cycles
without similar conditions without error
bit 3: Failure XX was present last DC
bit 4: Request to increment driving cycle counter of failure XX

bit 5: " exceedance" of specific limit was reported for failure XX

Input data:

LV_DC	LV_STATE_WUP	N	LOAD_SCDN
LC_ENA_SCDN	NLC_ENA_SCDN	LV_ERR_DC[NC_NR_ERR_DYN]	CTR_DC[NC_NR_ERR_DYN]
LV_CDN_DIAG_XX			

Export actions:

ACTION_ERRM_DecrementDCctrScdn (OUT <DCdec>, IN <XX>)
This action calculates the possibility to decrement the driving cycle counter related to failure XX
ACTION_ERRM_EraseScdn(IN <XX>)
This action erase the similar conditions related to failure XX (if existing)
ACTION_ERRM_CdnDiagScdn (IN <XX>)
This action is used to calculate a ratio to recognize the similar condition without failure
ACTION_ERRM_EndWinScdn (IN <XX>, IN <EXC>)
This is used to store similar conditions and recognize similar condition with or without failure

Import actions:

ACTION_ERRM_IncrementDCctrScdn (IN <XX>)

FUNCTION DESCRIPTION:

General information:

The different states of a failure are managed regards their driving cycle and warm up cycle counters values (see Dynamic error management core for details).

For Misfire and Fuel system failures, some additional conditions should be fulfilled to increment and decrement their driving cycle counter. These additional conditions are related to engine conditions and are called similar conditions.

When similar conditions without failure are fulfilled for the failure XX already stored in the dynamic memory in a previous driving cycle, bit 2 of STATE_SCDN[k] is set to 1. That means, that the driving cycle counter of failure XX could be decremented at the end of the driving cycle if the failure XX will not get present till this treatment. If an exceedance is reported by the diagnostic function (bit 5 of STATE_SCDN[k] is set to 1), the driving cycle counter is not decremented at the end of the DC. Exception: Driving cycle counter decrementation linked with C_CTR_SCDN_EQU_DC_MAX DCs, see description below.

Also the incrementation of the driving cycle counter of a failure stored in dynamic memory is controlled by the similar conditions functionality. The incrementation done in the error management core is triggered by the ACTION_ERRM_IncrementDCctrScdn (XX) in the following cases:

- Similar conditions are recorded

  (bit 0 of STATE_SCDN[k] 0->1).
- A failure already stored in memory occurs again in the following driving cycle (bit 2 of STATE_SCDN[k] = 1).
- A temporary failure or a confirmed failure with MIL off (LV_DC_MAX[IDX] = 0) occurs again not in the following driving cycle but with similar conditions with failure (bit 1 of STATE_SCDN[k] = 1).
- A confirmed failure with MIL on (LV_DC_MAX[IDX] = 1) occurs again.

In all of these cases, the driving cycle counter is incremented (bit 4 of STATE_SCDN[k] 0→1).

For a temporary failure or confirmed failure with MIL off (LV_DC_MAX[IDX] = 0), even if similar conditions are not recognized during a C_CTR_CDN_EQU_DC_MAX number of driving cycles, the driving cycle counter of the failure can be decremented after C_CTR_CDN_EQU_DC_MAX driving cycles.

Similar conditions functionality is inhibited when LC_ENA_SCDN=0 or NLC_ENA_SCDN=0.

Remark:

To keep the link between the index of the failure using the similar condition and the index of the failure in static memory, a transfer function is defined: k = Scdn (XX). The link between the diagnosis instance and the similar conditions instance k is described in the application incidence of similar condition.

B.65.1 Ratio calculation to recognize the similar condition without failure

Description:

For the recognition of similar conditions the current engine speed / load conditions are compared with the recorded similar conditions window.

Based on two counters calculated with this algorithm the SCDN recognition at end of the diagnostic window is done. (see next chapter)

- CTR_SCDN_EQU[k]

  counting the number of times engine speed / load conditions are within the recorded similar conditions window (with LV_CDN_DIAG_XX = 1)

- CTR_SCDN_SUM[k]

  counting the number of times the diagnosis conditions are set (LV_CDN_DIAG_XX = 1)

The calculation of the counters is stopped, if an exceedance is reported from diagnostic function.

Syntax : ACTION_ERRM_CdnDiagScdn (IN <XX>)

Parameter (in) : XX failure XX using similar condition instance XX

Parameter (out) : -

Short description:

This action is called at each diagnosis condition. It is used to calculate a ratio to recognize the similar condition with or without failure.

Application conditions:

Initialization: at transition LV_DC 0 -> 1 or reset

bit 0, bit 3 of STATE_SCDN[k] = restored from NVMY
bit 1, bit 2, bit 4, bit 5 of STATE_SCDN[k] = 0

CTR_SCDN_EQU[k] = 0
CTR_SCDN_SUM[k] = 0

Recurrence:

Activation: At action request **and** LV_DC = 1
and LC_ENA_SCDN = 1 and NLC_ENA_SCDN = 1
(similar conditions enable)

Formula section:

k = Scdn (XX)

If bit 0 of STATE_SCDN[k] = 1 (similar conditions are recorded for failure XX)

and bit 5 of STATE_SCDN[k] = 0 (no exceedance reported from diagnostic function)

Then **If** LV_CDN_DIAG_XX = 1

Then CTR_SCDN_SUM[k] ++

(check if current N, LOAD_SCDN point is within SCDN window)

If N <= N_SCDN_MAX_EQU[k]

and N >= N_SCDN_MIN_EQU[k]

and LOAD_SCDN <= LOAD_SCDN_MAX_EQU[k]

and LOAD_SCDN >= LOAD_SCDN_MIN_EQU[k]

Then CTR_SCDN_EQU[k] ++

Endif

Endif

B.65.2 Record and recognition of similar conditions at end of diagnosis window

Description:

The similar conditions are stored for a failure XX (or updated) in the following cases:

- The similar conditions are frozen within the window N/LOAD_MAX/MIN_SCDN_EQU[k], if the corresponding failure is stored in the memory (failure gets present the first time)
- The similar conditions are stored again (updated) within the window N/LOAD_MAX/MIN_SCDN_EQU[k], if the corresponding failure occurs again and the driving cycle counter of this failure was 0 before.

In both cases bit 0 of STATE_SCDN[k] is 0 before the storage.

The recognition of similar conditions for detected failure is done by comparing, if the actual N/LOAD point is within the stored SCDN window. If also the same warmup status is reached at this time, similar conditions with failure are recognized. This calculation is done at the action call depending of the monitoring function, only in the following case:

- Similar conditions are stored in a previous driving cycle (bit 0 of STATE_SCDN[k] = 1)
- and the same failure XX was not present the last driving cycle. In case of same failure present last driving cycle, the driving cycle counter is incremented without asking for similar conditions.
- and failure is not illuminating the MIL. In case failure is illuminating the MIL at present the driving cycle counter is set to the maximum without asking for similar conditions.

- and driving cycle counter not already incremented in this driving cycle

If no error is present at the end of the diagnosis window, the ratio (CTR_SCDN_EQU[k] / CTR_SCDN_SUM[k]) is compared with the threshold C_CTR_SCDN_NOT_ERR_THD[k]. If the threshold and the same warmup status are reached at this time, similar conditions without failure are recognized.

This calculation is done at the end of a diagnosis window, only if no failure XX is present in the following cases:

- Similar conditions are already recorded for failure XX (bit 0 of STATE_SCDN[k] = 1) in a previous driving cycle
- Similar conditions were not already encountered within this driving cycle with no failure
- no exceedance reported from diagnostic function (bit 5 of STATE_SCDN[k] = 0)
- Driving cycle counter was not incremented this driving cycle yet.

Syntax :	ACTION_ERRM_EndWinScdn (IN <XX>, <EXC>)	
Parameter (in) :	XX	failure XX using similar condition instance XX
	EXC	Exceedance reported from Diagnostic function
Parameter (out) :	-	

Short description:

This action is called at each end of diagnosis windows if there's no failure or when the failure gets present or when an exceedance is recognized by the diagnostic function. This usage of this function depends on the diagnostic function. It is used for recognition of similar condition with or without failure

Application conditions:

Initialization: -

Recurrence: -

Activation: At action request **and** LV_DC = 1
and LC_ENA_SCDN = 1 **and** NLC_ENA_SCDN = 1
(similar conditions enable)

Formula section:

k = Scdn(XX)

If EXC = 1 (Exceedance reported from Diagnostic function)

Then bit 5 of STATE_SCDN[k] = 1

Endif

If bit 4 of STATE_SCDN[k] = 0 (request to increment driving cycle counter of failure XX not made)

and LV_ERR_XX = 1 (failure present)

Then **If** bit 0 of STATE_SCDN[k] = 1 (similar conditions recorded for failure XX)

Then **If** bit 3 of STATE_SCDN[k] = 1
(failure present the last driving cycle)

or LV_DC_MAX[IDX] = 1
(failure is confirmed with MIL on)

Then ACTION_ERRM_IncrementDCctrScdn (XX)
(request to increment driving cycle counter for failure XX)

bit 4 of STATE_SCDN[k] = 1
(request to increment driving cycle counter of was made)

Else (recognition of similar conditions with failure)

(check if ratio to recognize SCDN is reached)

If N <= N_MAX_SCDN_EQU[k]

and N >= N_MIN_SCDN_EQU[k]

and LOAD_SCDN <= LOAD_MAX_SCDN_EQU[k]

and LOAD_SCDN >= LOAD_MIN_SCDN_EQU[k]

and LV_WUP_SCDN_EQU[k] = LV_STATE_WUP

Then bit 1 of STATE_SCDN[k] = 1
(similar conditions encountered with failure window)

bit 2 of STATE_SCDN[k] = 0
(similar conditions not encountered with no failure)

ACTION_ERRM_IncrementDCctrScdn (XX)
(request to increment driving cycle counter for failure XX)

bit 4 of STATE_SCDN[k] = 1
(request to increment driving cycle counter of was made)

Endif

Endif

Else (SCDN engine speed/load point frozen)

N_SCDN_MAX_EQU[k] = N + C_N_SCDN_EQU

N_SCDN_MIN_EQU[k] = N - C_N_SCDN_EQU

LOAD_SCDN_MAX_EQU[k] = LOAD_SCDN + C_LOAD_SCDN_EQU

LOAD_SCDN_MIN_EQU[k] = LOAD_SCDN - C_LOAD_SCDN_EQU

LV_WUP_SCDN_EQU[k] = LV_STATE_WUP (warm-up status stored)

CTR_SCDN_EQU_DC[k] = C_CTR_SCDN_EQU_DC_MAX

bit0 of STATE_SCDN[k] = 1 (similar conditions recorded for failure XX)

ACTION_ERRM_IncrementDCctrScdn (XX)
(request to increment driving cycle counter for failure XX)

bit 4 of STATE_SCDN[k] = 1
(request to increment driving cycle counter was made)

Endif

Else (recognition of similar conditions without failure)

If bit 0 of STATE_SCDN[k] = 1 (similar conditions recorded for failure XX)

and bit 4 of STATE_SCDN[k] = 0 (request to increment driving cycle counter of failure XX was not made)

and bit 2 of STATE_SCDN[k] = 0 (similar conditions not encountered with no failure)

and LV_ERR_XX = 0 (no failure present)

and bit 5 of STATE_SCDN[k] = 0 (no exceedance reported from Diagnostic function)

Then **If** CTR_SCDN_EQU[k] / CTR_SCDN_SUM[k]>= C_CTR_SCDN_NOT_ERR_THD[k]

and LV_STATE_WUP = LV_WUP_SCDN_EQU[k] (warm-up status)

Then bit 2 of STATE_SCDN[k] = 1 (similar conditions encountered with no failure)

Endif

Endif

Endif

If bit 0 of STATE_SCDN[k] = 1 (similar conditions recorded for failure XX)

Then CTR_SCDN_EQU[k] = 0

CTR_SCDN_SUM[k] = 0

Endif

B.65.3 SCDN storage and recognition for detected failure performed last DC

Description:

According regulation the information, if a failure XX (using SCDN functionality) was present the last driving cycle, must be respected to increment the DC counter. In case of a new failure occurrence, if the same failure was present the last driving cycle, the driving cycle counter must be incremented (if not already incremented this driving cycle) independently of similar conditions.

Application conditions:

Initialization: -

Recurrence:

Activation: At LV_DC transition 1 -> 0 **and**

LC_ENA_SCDN = 1 **and** NLC_ENA_SCDN = 1
(similar conditions enable)

Formula section:

For k = 0 to NC_NR_WIN_SCDN - 1

k = Scdn(XX); XX = F(IDX)

If LV_ERR_DC[IDX] = 1 (failure present in this DC)

Then bit 3 of STATE_SCDN[k] = 1
(failure present the last driving cycle)

CTR_SCDN_EQU_DC[k] = C_CTR_SCDN_EQU_DC_MAX
(re-initialization of similar conditions counters)

Else bit 3 of STATE_SCDN[k] = 0
(failure not present the last driving cycle)

Endif

EndFor

B.65.4 SCDN usage to decrement driving cycle counter

Description:

Syntax : ACTION_ERRM_DecrementDCctrScdn (OUT <DCdec>, IN <XX>)

Parameter (in) : XX number of failure to store in dynamic memory

Parameter (out) : DCdec YES if decrementation of DC counter is possible

NO if decrementation of DC counter is not possible

Short description:

At the end of a driving cycle (LV_DC 1->0) the driving cycle counter for the failure XX can be decremented if similar conditions without failure are recognized and no exceedance reported from diagnostic function (bit 5 of STATE_SCDN[k] = 0) (used for misfire monitoring). For temporary failure or confirmed failure with MIL off, even if similar conditions are not recognized during a C_CTR_CDN_EQU_DC_MAX number of driving cycle, the driving cycle counter of the failure can be decremented after C_CTR_CDN_EQU_DC_MAX driving cycles.

Application conditions:

Initialization: -

Recurrence: -

Activation: at action request

Formula section:

k = Scdn(XX)

DCdec = NO (decrementation of DC counter is not possible)

If bit 2 of STATE_SCDN[k] = 1 (similar conditions encountered with no failure)

and bit 5 of STATE_SCDN[k] = 0 (no exceedance reported)

Then DCdec = YES (decrementation of DC counter is possible)

Else **If** LV_DC_MAX[IDX] = 0 (failure is temporary or confirmed with MIL off)

Then CTR_SCDN_EQU_DC[k] = CTR_SCDN_EQU_DC[k] – 1

If CTR_SCDN_EQU_DC[k] = 0

bit 2 of STATE_SCDN[k] = 1

DCdec = YES (decrementation of DC counter is possible)

Endif

Endif

Endif

If DCdec = YES

and **(**LV_DC_MAX[IDX] = 0 (failure is temporary or confirmed with MIL off and DC counter of failure will be 0 after decrementation)

or **(**LV_DC_MAX[IDX] = 1 **and** CTR_DC[IDX] <= DC_DEC_XX**))**

(failure is confirmed with MIL on and MIL will be switched off after decrementation)

Then bit 0 of STATE_SCDN[k] = 0 (The SCDN window of failure XX will be updated With new engine conditions (N / LOAD (min/max)) at next failure occurrence)

Endif

B.65.5 Similar conditions erase

Description:

Syntax : ACTION_ERRM_EraseScdn (IN <XX>)

Parameter (in) : XX number of failure to store in dynamic memory

Parameter (out) : -

Short description:

The similar conditions are erased by calling the action ACTION_ERRM_EraseScdn :
- when saved ram lost (at initialization)
- when corresponding failure is erased (see dynamic error management core module)

Application conditions:

Initialization: -

Recurrence: -

Activation: at action request

Formula section:

k = Scdn (XX)

STATE_SCDN[k] = 0

N_MAX_SCDN_ EQU [k] = 0 [rpm]

N_MIN_SCDN_EQU[k] = 8160 [rpm]

LOAD_MAX_SCDN_EQU[k] = 0 [%]

LOAD_MIN_SCDN_EQU[k] = 99.60937 [%]

CTR_SCDN_EQU_DC[k] = 0

LV_WUP_SCDN_EQU[k] = 0

CTR_SCDN_EQU[k] = 0

CTR_SCDN_SUM[k] = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_SCDN_EQU	1	0...1FE0H	0...8160	1	[rpm]
engine speed threshold for similar conditions detection (375 rpm)					
C_LOAD_SCDN_EQU	1	0...FFH	0...99.60937	0.390625	[%]
LOAD value for similar conditions detection (typical 20%)					
C_CTR_SCDN_EQU_DC_MAX	1	0...FFH	0...255	1	[-]
maximum value of the driving cycle counter for similar conditions erase					
C_CTR_SCDN_NOT_ERR_THD[NC_NR_WIN_SCDN]	1	0...FFH	0...99.60937	0.390625	[%]
threshold for ratio CTR_SCDN_EQU[NC_NR_WIN_SCDN] / CTR_SCDN_SUM[NC_NR_WIN_SCDN] to recognize SCDN without failure					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_WIN_SCDN	1	1...FFH	1...255	1	[-]
Number of instances used for similar conditions calculation					

B.66 Similar conditions (Appl. Inc.)

B.66.1 Similar conditions window definition

Input data:

NC_NR_WIN_SCDN	LV_CDN_DIAG_FSD[NC_CBK_EX_NR]	LV_CDN_DIAG_FSD_LAM_LIM[NC_CBK_EX_NR]	LV_END_DIAG_WIN_FSD[NC_CBK_EX_NR]
LV_END_DIAG_WIN_FSD_LAM_LIM[NC_CBK_EX_NR]	LV_CDN_DIAG_DELTA_I_LAM [NC_CBK_EX_NR]	LV_END_DIAG_WIN_DELTA_I_LAM[NC_CBK_EX_NR]	NC_STATE_LSL_UP_IF
LV_END_DIAG_WIN_LAM_ADJ[NC_CBK_EX_NR]	LV_CDN_DIAG_LAM_ADJ [NC_CBK_EX_NR]		

Import actions:

ACTION_ERRM_CdnDiagScdn (IN <XX>)
ACTION_ERRM_EndWinScdn (IN <XX>, IN <EXC>)

FUNCTION DESCRIPTION:

General information:

This file defines the usage of the similar conditions functionality. To follow specific customer requirements, this file is a template to be fulfilled by project team.

B.66.1.1 Similar conditions data definition

There is only some failure using the similar condition so this table allows to do the link between the index of the failure in static memory and the index of the failure using the similar condition:

- for NC_STATE_LSL_UP_IF = 0

Diagnosis instances XX using similar conditions functionality	NC_NR_WIN_SCDN = 7 *(Note: see ERRM configuration data)*
Diagnosis instance XX	**similar conditions instance (k)**
FSD_1	0
MIS_0	1
MIS_1	2
MIS_2	3
MIS_3	4
FSD_LAM_LIM_1	5
LAM_ADJ[1]	6
	(NC_NR_WIN_SCDN - 1)

- for NC_STATE_LSL_UP_IF = 1

Diagnosis instances XX using similar conditions functionality	NC_NR_WIN_SCDN = 7 *(Note: see ERRM configuration data)*

Diagnosis instance XX	similar conditions instance (k)
FSD_1	0
MIS_0	1
MIS_1	2
MIS_2	3
MIS_3	4
FSD_LAM_LIM_1	5
DELTA_I_LAM_1	6
	(NC_NR_WIN_SCDN - 1)

One function is defined to do the link between the index of the failure in static memory and the index of the failure using the similar condition: k = Scdn(XX)

B.66.1.2 Treatment activated by the fuel system diagnosis for the similar condition (for both NC_STATE_LSL_UP_IF = 0 and NC_STATE_LSL_UP_IF = 1)

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Application conditions:

Initialisation: -

Recurrence: fuel system diagnosis recurrency

Activation: -

Formula section:

Adaptation to calculate a ratio used to recognize the similar condition with or without failure:

If LV_CDN_DIAG_FSD_1 = 1

Then ACTION_ERRM_CdnDiagScdn (FSD_1)
Endif

Adaptation for the record and recognition of similar conditions:

If LV_END_DIAG_WIN_FSD_1 = 0 ->1

Then

Record and recognition of similar conditions at the end of diagnosis or when the failure is present:

ACTION_ERRM_EndWinScdn (FSD_1, 0)

Endif

If LV_CDN_DIAG_FSD_LAM_LIM_1 = 1

Then ACTION_ERRM_CdnDiagScdn (FSD_LAM_LIM_1)
Endif

Adaptation for the record and recognition of similar conditions:

If LV_END_DIAG_WIN_FSD_LAM_LIM_1 = 0 ->1

Then

Record and recognition of similar conditions at the end of diagnosis or when the failure is present:

ACTION_ERRM_EndWinScdn (FSD_LAM_LIM_1, 0)

Endif

B.66.1.3 Treatment activated by the dyanmic fuel trim diagnosis for the similar condition (only for NC_STATE_LSL_UP_IF = 1)

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Application conditions:

Only valid for NC_STATE_LSL_UP_IF = 1

Initialisation: -

Recurrence: dynamic fuel trim diagnosis recurrency (100 ms)

Activation: -

Formula section:

Adaptation to calculate a ratio used to recognize the similar condition with or without failure:

If LV_CDN_DIAG_DELTA_I_LAM_1 = 1

Then ACTION_ERRM_CdnDiagScdn (DELTA_I_LAM_1)

Endif

Adaptation for the record and recognition of similar conditions:

If LV_END_DIAG_WIN_DELTA_I_LAM_1 = 0 ->1

Then

Record and recognition of similar conditions at the end of diagnosis or when the failure is present:

ACTION_ERRM_EndWinScdn (DELTA_I_LAM_1, 0)

Endif

B.66.1.4 Treatment activated by the dyanmic fuel trim diagnosis for the similar condition (only for NC_STATE_LSL_UP_IF = 0)

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Description:

This treatement allows to call the two generic ACTION_ERRM_CdnDiagScdn (IN <XX>), ACTION_ERRM_EndWinScdn (IN <XX>>, IN <EXC>) actions using for the similar condition.

Application conditions:

Only valid for NC_STATE_LSL_UP_IF = 0

Initialisation: -

Recurrence: dynamic fuel trim diagnosis recurrency (1s)

Activation: -

Formula section:

Adaptation to calculate a ratio used to recognize the similar condition with or without failure:

If LV_CDN_DIAG_LAM_ADJ[1] = 1

Then ACTION_ERRM_CdnDiagScdn (LAM_ADJ[1])
Endif

Adaptation for the record and recognition of similar conditions:

If LV_END_DIAG_WIN_LAM_ADJ[1] = 0 ->1

Then

Record and recognition of similar conditions at the end of diagnosis or when the failure is present:

ACTION_ERRM_EndWinScdn (LAM_ADJ[1], 0)

Endif

B.66.2 SCDN usage for direct failure confirmation

Export actions:

ACTION_ERRM_ConfirmErrScdn (OUT <DCconf>, IN <XX>)
This action calculates the possibility to confirm a failure XX directly

Description:

Syntax : ACTION_ERRM_ConfirmErrScdn (OUT <DCconf>, IN <XX>)

Parameter (in) : XX number of failure to store in dynamic memory

Parameter (out) : DCconf 1 if failure must be confirmed directly

0 if failure should not be confirmed directly

Short description:

With the ACTION_ERRM_ConfirmErrScdn (OUT <DCconf>, IN <XX>) the value DCconf is transmitted to the driving cycle counter management to give the possibility to confirm a failure directly by incrementing the driving cycle counter for the failure XX to the maximum. This treatment must be done in the following situation:

A temporary failure XX caused by misfire B is in the memory and a failure XX caused by misfire A occures (not stored before). Then the misfire B depending failure can be erased and the misfire A failure can be confirmed directly.

Application conditions:

Initialization: -

Recurrence: -

Activation: at action request

Formula section:

ACTION_ERRM_ConfirmErrScdn (DCconf, XX):

DCconf = 0 (direct confirmation of failure XX is not possible)

B.66.3 Definition of similar conditions load

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LOAD_SCDN	O/V	0...FFH	0...99.60937	0.390625	%
Engine load for SCDN calculation					

Input data:

LOAD_CLC			

Description:

The calculated load SAE1979 LOAD_CLC is used for similar condition.

Application conditions:

Initialisation: At reset:

LOAD_SCDN = LOAD_CLC

Recurrence: 100 ms

Activation: -

Formula section:

LOAD_SCDN = LOAD_CLC

B.66.4 Definition of switch for use of similar conditions load

Short description:

As this switch was not defined in ERRM core module, it is defined in application incidences. It always has to be calibrated to **LC_ENA_SCDN = 1** for OBD-projects**.**

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ENA_SCDN	1	00H...FFH	0...255	1	-
Switch to enable similar conditions					

B.67 Error management communication interface

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_MKD_MOD	O	0...1H	0...1	1	[-]
Marked Mode					
ERR_SYM_DTC[NC_NR_ERR_DYN]	-	0...FH	0...15	1	[-]
Symtom calculation used by the API communication					
#IF NLC_TREAT_DIAG_MIS = 0					
SYM_CYL_DTC_MIS_A	-	0...FFFFH	0...65535	1	[-]
Cylinder calculation with misfire A used by the API communication					
SYM_CYL_DTC_MIS_B1	-	0...FFFFH	0...65535	1	[-]
Cylinder calculation with misfire B1 used by the API communication					
SYM_CYL_DTC_MIS_B4	-	0...FFFFH	0...65535	1	[-]
Cylinder calculation with misfire B4 used by the API communication					
#ENDIF					
IDX_TMP_RBM	V	0...FFH	0...255	1	[-]
Temporary current position, within the list					
RATIO_TMP_RBM	V	0...FFFFH	0...7.99987	0.1221e-3	[-]
Temporary variable used to store the lowest Numerator/Denominator ratio of the monitor located at IDX_TMP_RBM position					
LV_READY_FSD_KOBD	V/S	00..01H	0...1	1	-
Readiness bit for FSD in FMY					
LV_READY_MISF_KOBD	V/S	00..01H	0...1	1	-
Readiness bit for MISF in FMY					

Input data:

ERR_SYM_MEM[NC_NR_ERR_DYN]	SYM_CYL_MEM_MIS_A	SYM_CYL_MEM_MIS_B1	SYM_CYL_MEM_MIS_B4
ERR_SYM_LST[NC_NR_ERR_DYN]	SYM_CYL_LST_MIS_A	SYM_CYL_LST_MIS_B1	SYM_CYL_LST_MIS_B4
ID_ERR_DTC_XX	ID_ERR_DTC_MIS	NC_CYL_NR	NC_NR_DTC_FMT
LV_ERR_XX	LV_ERR_MEM_XX	LV_ERR_TMP[NC_NR_ERR_DYN]	WAL_CONF_XX
LV_ERR_CFM[NC_NR_ERR_DYN]	LV_ERR_DISA[NC_NR_ERR_DYN]	PRI_CONF_XX	ENVD_OBD[NC_NR_ENVD_OBD][NC_NR_ERR_DYN]
ENVD_CUS_CMN[NC_NR_ENVD_CUS_CMN][NC_NR_ERR_DYN]	ENVD_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN][NC_NR_FRF_SET][NC_NR_ERR_DYN]	ENVD_CUS_SET_SPC[NC_NR_ENVD_CUS_SET_SPC][NC_NR_FRF_SET][NC_NR_ERR_DYN]	C_ERR_CLAS_XX
CTR_ABC_XX	CTR_ABC_END_DIAG_XX	LV_END_DIAG_XX	LV_READY_XX
LV_CDN_DIAG_XX	ERR_SYM_XX	CTR_FRC[NC_NR_ERR_DYN]	CTR_DC[NC_NR_ERR_DYN]
CTR_WUP_CYC[NC_NR_ERR_DYN]	CTR_ERR_DYN_NR	LV_DC_MAX[NC_NR_ERR_DYN]	LV_ERR_DC[NC_NR_ERR_DYN]
NLC_TREAT_DIAG_MIS	DC_DEC_XX	DC_INC_XX	DC_MAX_XX
CTR_IGK_CYC_RBM	CTR_CDN_OBD_RBM	CTR_CDN_RBM[NC_NR_DIAG_RBM]	CTR_COMP_RBM[NC_NR_DIAG_RBM]
NC_NR_DIAG_RBM	NLC_OBD_RBM_ENA	STATE_READY_OBD_1	STATE_READY_OBD_2
STATE_READY_OBD_3	LV_DC	CONF_KOBD	LV_READY_CAT_DIAG
NLC_OBD_DSL	ERR_SYM_PERM[NC_NR_ERR_PERM]	NC_RBM_CAT_1	NC_RBM_CAT_2
NC_RBM_EGR_VVT	NC_RBM_EVAP	NC_RBM_LS_DOWN_1	NC_RBM_LS_DOWN_2
NC_RBM_LS_UP_1	NC_RBM_LS_UP_2	NC_RBM_SA	SYM_CYL_PERM_MIS_A
SYM_CYL_PERM_MIS_B1	SYM_CYL_PERM_MIS_B4		

Export actions:

ACTION_ERRM_ReadDtcByTypeOfDtc (IN <TypeOfDtc>, IN <LevelOfDtc>, INOUT <ListOfDtc>, OUT <ResultDtc>)
This action returns a list of DTC. All the DTC returns have the same type defined by the parameter TypeOfDtc
ACTION_ERRM_ReadDtcByDtc (IN <DtcIdentifier>, IN <LevelOfDtc>, OUT <ResultDtc>)
This action returns a result to learn if a DTC number of a certain level is stored in memory. ACTION_ERRM_ReadInfoByTypeOfDtc (IN <TypeOfDtc>, IN <InfoIdentifier>, INOUT <ListOfDtcInfo>, OUT <ResultDtc>)
This action returns a structure of data of DTCs. All the data returned are related to DTC, which have the same type defined by the parameter TypeOfDtc and which are stored in dynamic memory.
ACTION_ERRM_ReadInfoByDtc (IN <DtcIdentifier>, IN <LevelOfDtc>, IN <InfoIdentifier>, INOUT <ListOfDtcInfo>, OUT <ResultDtc>)
This action returns a structure of data of a DTC, which is stored in dynamic memory.
ACTION_ERRM_ReadDtcLevelByDtcLevel (IN <DtcIdentifier>, IN <LevelOfDtc>, INOUT <ListofDtc>, OUT <ResultDtc>)
This action returns a Customer DTCLevel of Law DTCLevel which is stored in dynamic memory or returns a Law DTCLevel of Customer DTCLevel which is stored in dynamic memory.
ACTION_ERRM_ReadQuantityOfDtc (IN <TypeOfDtc>, IN <LevelOfDtc>, INOUT <Quantity>, OUT <ResultQuantity>)
This action returns the quantity of DTCs with a certain type, which are stored in memory.
ACTION_ERRM_ReadFrfByDtc (IN <TypeOfFF>, IN <FFIdentifier>, IN <DtcIdentifier>, IN <LevelOfDtc>, INOUT <Frf>, OUT <ResultFrf>)
This action returns a Freeze_Frame (with a particular type) related to the DTC given in parameter.
ACTION_ERRM_ClrInfoByTypeOfDtc (IN <TypeOfDtc>, OUT <ResultClrInfo>)
This action clear all the failure in dynamic memory associated to DTC with a certain type.
ACTION_ERRM_ClrInfoByDtc (IN <DtcIdentifier>, IN <LevelOfDtc>, OUT <ResultClrInfo>)
This action clear the failure in dynamic memory associated to a DTC number.
ACTION_ERRM_ControlDtcSettings (IN <MarkedMode>)
This action allows to activate or deactivate the marked mode. In this mode all new failure which appears are marked and then erased when we leave this mode.
ACTION_ERRM_ActivateMarkedMode (IN <MarkedMode>)

This API shall be used to activate/deactivate the marked mode.
ACTION_ERRM_ReadReadinessCode (INOUT <ReadinessCode>, OUT <ResultReadinessCode>)
This action allows to read the readiness code information
ACTION_ERRM_ReadReadinessCodCus (INOUT <ReadinessCode>, OUT <ResultReadinessCode>)
This action allows to read the customer specific readiness code information CARB_OTHER/NO_CARB
ACTION_ERRM_SelectRbmData (INOUT <ListRbmData>, OUT < ResultRbmData >)
This action is used to request the selection of data to be transmitted by the Communication Mode $09
ACTION_ERRM_SelectRbmByGroup (IN < GroupName >, INOUT < ListOfRbmDataByGroup >, OUT < ResultRbmDataSelection >)
This action determines within one single group, the data with lowest ratios.
ACTION_ERRM_ClearRbmStatistics (OUT < ResultClrInfo>)
This action clears all the Rate-Based Monitoring statistics

Import actions:

ACTION_ERRM_EraseErr (IN <IDX>)
ACTION_ERRM_InitReadiness ( )
ACTION_ERRM_ClcPermanentByErr (IN <PRM_IDX_ERR>)
ACTION_ERRM_ClcPermanentIniErrm ( )

FUNCTION DESCRIPTION:

General information:

This chapter provides an open interface for other module which have link with error management functionality (communication module for instance). This interface is described with API (Application Programming Interface) to provide access to dynamic data flow in contrast to static data flow accessed directly through variable names.

By calling API, you can access in failure memory data related to Error Management such freeze frame and DTC stored, readiness code state and so on.

An API can have some input parameters.

Results returns by the API are defined as output parameters.

To returns functional results, a software structure (buffer) is used that allows to returns some data with not fixed size (a list of data for instance).

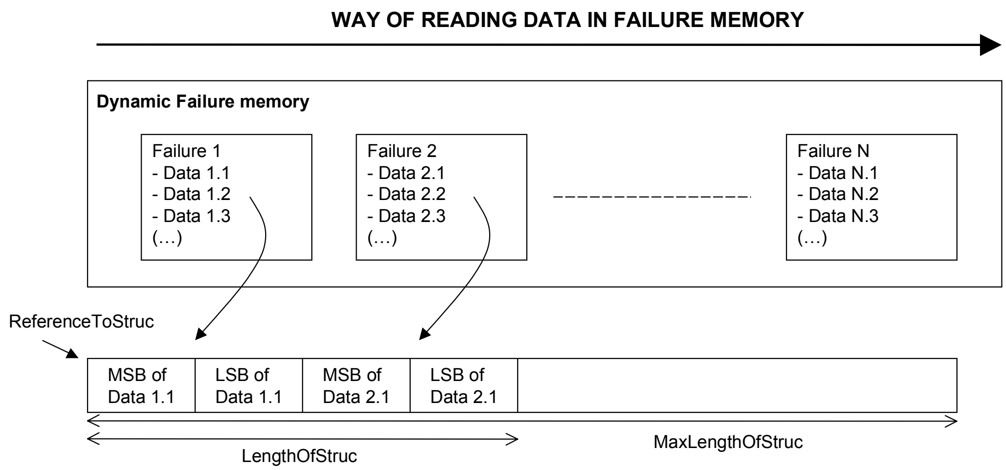
This structure is defined by the parameter in INOUT.

In the output parameter, a flag indicate if the software buffer is full or not.

Way of reading the failure memory :

API has to read through the failure memory to access some data related to Error Management.

In this case the failure memory is read in the order following the first failure entry, that means the oldest one. Thus the software structure is fill in first with data related to first failure entry (if need), then with data related to second failure entry (if need) and so on.



GENERAL IMPORTANT REMARK:

Please take care that the parameter PERMANENT is only linked to permanent memory (e.g. no impact on dynamic memory). For all services err_sym_dtc is still calculated with err_sym_mem and err_sym_lst (no link to err_sym_perm)

Fot the APIs :

- ACTION_ERRM_ReadDtcByDtc,
- ACTION_ERRM_ReadInfoByTypeOfDtc,
- ACTION_ERRM_ReadInfoByDtc,
- ACTION_ERRM_ReadDtcLevelByDtcLevel,
- ACTION_ERRM_ClrInfoByTypeOfDtc

It's forbidden to use the parameter PERMANENT for TypeOfDTC

For Mode 0Ah, the two APIs which must be used are

- ACTION_ERRM_ReadQuantityOfDtc with PERMANENT parameter for TypeOfDtc
- ACTION_ERRM_ReadDtcByTypeOfDtc with PERMANENT parameter for TypeOfDtc

Description of parameters IN, OUT, INOUT used by API:

The INPUT parameters used by API and their authorised value are defined below :

TypeOfDtc : Type of Dtc

Authorised values for TypeOfDtc are :

- ALL : all the DTC stored in Dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
- ALL FIRST: all the DTC stored at the first time in Dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
- ALL LAST: all the DTC stored at the last time in Dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
- PRESENT : all the DTC of present failure stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_XX = 1
- PENDING : all the DTC of pending failure stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_PND[IDX] = 1
- TEMPORARY : all the DTC of temporary failure stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_TMP[IDX] = 1
- CONFIRMED : all the DTC of confirmed failure stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_CFM[IDX] = 1
- DISAPPEARED : all the DTC of disappeared failure stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_DISA[IDX] = 1
- MARKED : all DTC of failure marked and stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_MKD[IDX] = 1
- NOT MARKED : all DTC of failure not marked and stored in dynamic memory.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and LV_ERR_MKD[IDX] = 0
- OBD : all DTC related to emission relevant failure.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and a Carb/EOBD failure :
  (emission relevant)
  (definition see failure class specification)

- OBD FIRST: all the DTC stored at the first time in Dynamic memory related to emission relevant failure.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and a Carb/EOBD failure :
  (emission relevant)
  (definition see failure class specification)
- OBD LAST: all the DTC stored at the last time in Dynamic memory.related to emission relevant failure.
  Failures concerned in error management : LV_ERR_MEM_XX = 1
  and a Carb/EOBD failure :
  (emission relevant)
  (definition see failure class specification)
- PERMANENT : all the DTC of permanent fault memory
  Failures concerned in error management : LV_ERR_PERM[IDX_PERM] = 1
- Logical AND combination between these values.
  For instance : PRESENT and CONFIRMED are failure in error management with flags : LV_ERR_MEM_XX = 1 and LV_ERR_XX = 1 and LV_ERR_CFM[IDX] = 1.

Calculation of ERR_SYM_DTC[IDX] and SYM_CYL_DTC_XX following the TypeOfDtc parameter:

For all failure except the misfiring failure treated by type :

case TypeOfDtc

ALL : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX] **or** ERR_SYM_MEM[IDX]

ALL FIRST: ERR_SYM_DTC[IDX] = ERR_SYM_MEM[IDX]

ALL LAST : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX]

PRESENT : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX]

PENDING : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX]

TEMPORARY : ERR_SYM_DTC[IDX] = ERR_SYM_MEM[IDX]

CONFIRMED : ERR_SYM_DTC[IDX] = ERR_SYM_MEM[IDX]

DISAPPEARED : ERR_SYM_DTC[IDX] = ERR_SYM_MEM[IDX]

MARKED : ERR_SYM_DTC[IDX] =
ERR_SYM_LST[IDX] **or** ERR_SYM_MEM[IDX]

NOT MARKED : ERR_SYM_DTC[IDX] =
ERR_SYM_LST[IDX] **or** ERR_SYM_MEM[IDX]

OBD : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX] **or** ERR_SYM_MEM[IDX]

OBD FIRST : ERR_SYM_DTC[IDX] = ERR_SYM_MEM[IDX]

OBD LAST : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX]

(!! independant of dynamic memory !!)

PERMANENT : ERR_SYM_PERM[IDX_PERM]

End case

For the misfiring failure treated by type (XX = MIS_A, MIS_B1 or MIS_B4)

IF NLC_TREAT_DIAG_MIS = 0

case TypeOfDtc

ALL : SYM_CYL_DTC_XX = SYM_CYL_LST_XX **or** SYM_CYL_MEM_XX

ALL FIRST : SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

ALL LAST: SYM_CYL_DTC_XX = SYM_CYL_LST_XX

PRESENT: SYM_CYL_DTC_XX = SYM_CYL_LST_XX

PENDING : SYM_CYL_DTC_XX = SYM_CYL_LST_XX

TEMPORARY : SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

CONFIRMED : SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

DISAPPEARED : SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

MARKED : SYM_CYL_DTC_XX = SYM_CYL_LST_XX **or** SYM_CYL_MEM_XX

NOT MARKED : SYM_CYL_DTC_XX =
SYM_CYL_LST_XX **or** SYM_CYL_MEM_XX

OBD : SYM_CYL_DTC_XX = SYM_CYL_LST_XX **or** SYM_CYL_MEM_XX

OBD FIRST: SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

OBD LAST: SYM_CYL_DTC_XX = SYM_CYL_LST_XX

PERMANENT : SYM_CYL_DTC_XX = SYM_CYL_MEM_XX

End case

ENDIF { NLC_TREAT_DIAG_MIS = 0 }

For the logical AND combination between the different TypeOfDtc values, the ERR_SYM_DTC[IDX] or SYM_CYL_DTC_XX will be calculated with a binary logical AND from the associated carrier symptom.

For instance : PRESENT and CONFIRMED are failure in error management with flags : LV_ERR_MEM_XX = 1 and LV_ERR_XX = 1 and LV_ERR_CFM[IDX] = 1.
It means : ERR_SYM_DTC[IDX] = ERR_SYM_LST[IDX] **&** ERR_SYM_MEM[IDX],
SYM_CYL_DTC_XX = SYM_CYL_LST_XX **&** SYM_CYL_MEM_XX.

Calculation of ERR_SYM_DTC[IDX] and SYM_CYL_DTC_XX in case the API doesn't use the TypeOfDtc parameter :

ERR_SYM_DTC[IDX] = ERR_SYM_LST **or** ERR_SYM_MEM[IDX]

SYM_CYL_DTC_XX = SYM_CYL_LST_XX **or** SYM_CYL_MEM_XX

DtcIdentifier : Number of the DTC (identifier)

Authorised value for DtcIdentifier are : NO_DTC, ...<integer>

TypeOfFF : Type of Freeze Frame

Authorised values for TypeOfFF are :

- LAW : part of the freeze frame defined by the law (Carb)
- CUS_CMN : part of the freeze frame defined for the customer and not in the set
- CUS_SET : part of the freeze frame defined for the customer and in the set
- All combination of these values.

LevelOfDtc : Allowing to access to OBD error codes or customer error codes.

Authorised values for LevelOfDtc are :

- LAW : the DTC corresponds to a OBD DTC (Carb)
- CUS : the DTC corresponds to a specific customer DTC

FFIdentifier : Number of FF (identifier)

Authorised values for FFIdentifier are :

- FIRST : first freeze frame is requested
- SECOND : second freeze frame is request
- ...

MarkedMode : State of the marked mode

Authorised values for MarkedMode are :

- ON : marked mode activation : all failure stored are marked.
- OFF : marked mode deactivation (normal mode)

InfoIdentifier : Identifier for the diagnosis related information which is returned by the API (only words are returned)

Authorised information for InfoIdentifier are (only one information is returned by the API):

- CTR_ABC_XX
- CTR_ABC_END_DIAG_XX
- LV_END_DIAG_XX
- LV_READY_XX
- LV_CDN_DIAG_XX
- ERR_SYM_XX
- LV_ERR_XX
- LV_ERR_MEM_XX
- CTR_FRC[IDX]
- CTR_DC[IDX]
- CTR_WUP_CYC[IDX]
- ERR_SYM_MEM[IDX]
- ERR_SYM_LST[IDX]
- ERR_SYM_DTC[IDX]
- LV_ERR_CFM[IDX]
- LV_ERR_DISA[IDX]
- LV_ERR_TMP[IDX]

- LV_ERR_PND[IDX]
- LV_ERR_PERM[IDX]
- LV_DC_MAX[IDX]
- LV_ERR_DC[IDX]
- LV_ERR_MKD[IDX]
- C_ERR_CLAS_XX
- WAL_CONF_XX
- PRI_CONF_XX
- DC_INC_XX
- DC_DEC_XX
- DC_MAX_XX
- ID_ERR_DTC_XX[0][0]
- ID_ERR_DTC_XX[0][1]
- ID_ERR_DTC_XX[0][2]
- ID_ERR_DTC_XX[0][3]
- ID_ERR_DTC_XX[0][4]

if NC_NR_DTC_FMT = 0

(Manage 6 DTC per failure)

- ID_ERR_DTC_XX[0][5]

else

(Manage 10 DTC per failure)

- ID_ERR_DTC_XX[1][0]
- ID_ERR_DTC_XX[1][1]
- ID_ERR_DTC_XX[1][2]
- ID_ERR_DTC_XX[1][3]
- ID_ERR_DTC_XX[1][4]

endif

Please note that some application incidences can provide additional project specific informations.

The OUTPUT parameters used by API and their authorised value are defined below :

ResultDtc : Result to say if there is some DTC in memory corresponding to the API request

Authorised values for ResultDtc are :

1st bit of ResultDtc :

- (0) NO_DTC_PRESENT : there is no DTC in memory corresponding to the API request
- (1) DTC_PRESENT : there is some DTC in memory corresponding to the API request

2nd bit of ResultDtc :

- (0) NO_BUFFER_FULL : The software buffer used to return data is not full
- (1) BUFFER_FULL : The software buffer used to return data is full

ResultFrf : Result to say if there is some Freeze Frame in memory corresponding to the API request

Authorised values for ResultFrf are :

1st bit of ResultFrf :

- (0) NO_FRF_PRESENT : there is no freeze frame in memory corresponding to the API request
- (1) FRF_PRESENT : there is some freeze frame in memory corresponding to the API request

2nd bit of ResultFrf :

- (0) NO_BUFFER_FULL : The software buffer used to return data is not full
- (1) BUFFER_FULL : The software buffer used to return data is full

ResultQuantity : Result to say if the result of the API is valid.

Authorised values for ResultQuantity are :

1st bit of ResultQuantity :

- (0) NO_DTC_PRESENT : there is no DTC in memory corresponding to the API request
- (1) DTC_PRESENT : there is some DTC in memory corresponding to the API request

ResultClrInfo : Result to say if the clear DTC made by the API is done or not.

Authorised values for ResultClrInfo are :

1st bit of ResultClrInfo :

- (0) INFO_CLEARED : DTC are cleared by the call of the API
- (1) INFO_NOT_CLEARED : no DTC is cleared by the call of the API

ResultReadinessCode : Result to say if the API is done or not.

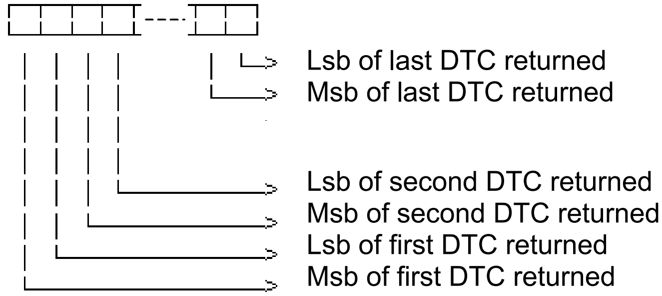
Authorised values for ResultReadinessCode are :

1st bit of ResultReadinessCode :

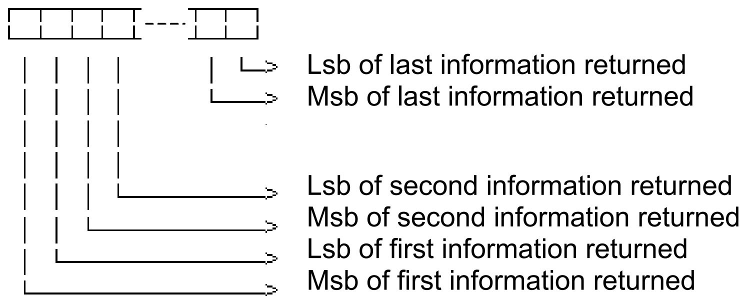
- (0) POSITIVE_RESPONSE : the call of the API is successful
- (1) NEGATIVE_RESPONSE : the call of the API is not successful

The INPUT/OUTPUT parameters used by API and their authorised value are defined below :

- ListOfDtc : Software structure filled up with a list of DTC

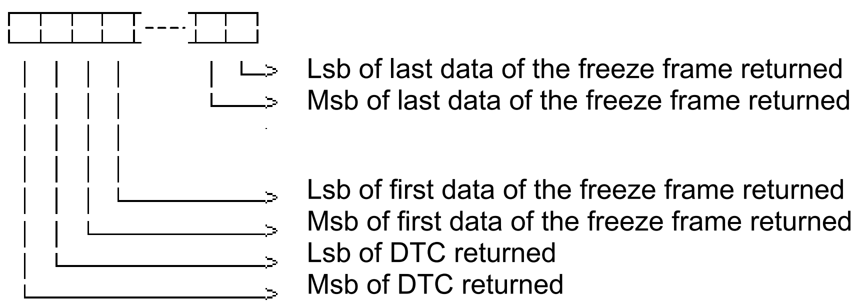


- ListOfDtcInfo : Software structure filled up with a list of DTCInfo



- Quantity : Software structure filled up with a quantity of DTC

- Frf : Software structure filled up with a freeze frame



- ReadinessCode : Software structure filled up with readiness code

B.67.1 API for reading Diagnostic Trouble Code (DTC)

B.67.1.1 Request a list of DTCs by type of DTC.

Description :

Syntax : ACTION_ERRM_ReadDtcByTypeOfDtc (
IN <TypeOfDtc>,
IN <LevelOfDtc>,
INOUT <ListOfDtc>,
OUT <ResultDtc>)

Parameter(in) :	TypeOfDtc	Type of DTC which is requested by the API.
	LevelOfDtc	OBD or customer DTC is requested by the API
Parameter(out) :	ResultDtc	Boolean to say if some DTC is in memory
Parameter(inout) :	ListOfDtc	Software structure fills up with a ListOfDtc

Short Description : This API returns a list of DTCs. All the DTC returns have the same type defined by the parameter TypeOfDtc and the same level defined in LevelOfDtc.

For each failure, if NC_NR_DTC_FMT = 0 then 6 DTC are defined : one DTC for each symptom (OBD or Customer), one OBD global DTC, one customer global DTC else 10 DTC are defined : one OBD DTC for each symptom, one OBD global DTC, one customer specific DTC number for each symptom and one customer global DTC.

When LevelOfDtc = LAW : OBD Dtc is request

If NC_ERR_DTC_REQ_OBD = SYMPTOM, the DTC is build up following the detected symptom. If more than one symptom is detected, a DTC per symptom detected is returned. There is a particular case for Misfiring failure. For this failure, the DTC number is built up following the cylinder where the misfiring is detected.

If NC_ERR_DTC_REQ_OBD = FAILURE, the OBD global DTC is returned.

When LevelOfDtc = CUS : Customer Dtc is request

If NC_ERR_DTC_REQ_CUS = SYMPTOM, the DTC is build up following the detected symptom. If more than one symptom is detected, a DTC per symptom detected is returned. There is a particular case for Misfiring failure. For this failure, the DTC number is built up following the cylinder where the misfiring is detected.

If NC_ERR_DTC_REQ_CUS = FAILURE, the customer global DTC is returned.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1st bit of ResultDtc = NO_DTC_PRESENT

2nd bit of ResultDtc = NO_BUFFER_FULL

If TypeOfDtc = PERMANENT

Then

If CTR_ERR_PERM_NR > 0

Then

1st bit of ResultDtc = DTC_PRESENT

Endif

For IDX_PERM = 0 to CTR_ERR_PERM_NR -1

with XX = ERR_PERM[IDX_PERM]

If failure is not with status = CARB_MIS

Then Condition for failure treatment is true

Endif

If NLC_TREAT_DIAG_MIS = 1

Then Condition for failure treatment is true

Endif

If Condition for failure treatment is true

Then **If** LevelOfDtc = LAW (OBD DTC is request)

Then **If** NC_ERR_DTC_REQ_OBD=SYMPTOM (access by symptom)

Then **For** each bit i which is set in
ERR_SYM_PERM[IDX_PERM]
software structure fill up with
ID_ERR_DTC_XX[0] [i]

Endfor

Else (access by failure; NC_ERR_DTC_REQ_OBD = FAILURE)
software structure fill up with ID_ERR_DTC_XX[0][4]

Endif

Else (Customer DTC is request ; LevelOfDtc = CUS)

If NC_ERR_DTC_REQ_CUS = SYMPTOM (access by symptom)

Then **For** each bit i which is set in
ERR_SYM_PERM[IDX_PERM]
software structure fill up with
ID_ERR_DTC_XX[NC_NR_DTC_FMT] [i]

Endfor

Else (access by failure; NC_ERR_DTC_REQ_CUS = FAILURE)
software structure fill up with
ID_ERR_DTC_XX
[NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]

```
                  Endif
              Endif
          Else            (particular case for misfiring failure NLC_TREAT_DIAG_MIS = 0)
              XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
              If      LevelOfDtc = LAW                          (OBD DTC is request)
              Then    If      SYM_CYL_PERM_XX [12]=1            (Random)
                      Then    software structure fill up with
                              ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR+1]
                      Endif
                      If      SYM_CYL_PERM_XX [13]=1            (Multiple cylinder)
                      Then    software structure fill up with
                              ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR ]
                      Endif
                      For each cylinder bit i which is set in SYM_CYL_PERM_XX
                                software structure fill up with
                                ID_ERR_DTC_MIS[ 0 ] [ i ]
                      Endfor
              Else    (Customer DTC is request ; LevelOfDtc = CUS)
                      If      SYM_CYL_PERM_XX [12]=1            (Random)
                      Then    software structure fill up with
                              ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR+1]
                      Endif
                      If      SYM_CYL_PERM_XX [13]=1            (Multiple cylinder)
                      Then    software structure fill up with
                              ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR ]
                      Endif
                      For each cylinder bit i which is set in SYM_CYL_PERM_XX
                              software structure fill up with
                              ID_ERR_DTC_MIS[ 1 ] [ i ]
                      Endfor
              Endif
          Endif
      Endfor
  Else
      For each failure IDX stored in dynamic memory,
         If           the failure belongs to failures group defined by the parameter TypeOfDtc
            Then   1st bit of ResultDtc = DTC_PRESENT
              If failure is not with status = CARB_MIS
              Then Condition for failure treatment is true
              Endif
              If NLC_TREAT_DIAG_MIS = 1
              Then Condition for failure treatment is true
              Endif
              If Condition for failure treatment is true
```

```
        Then If   LevelOfDtc = LAW              (OBD DTC is request)
            Then  If  NC_ERR_DTC_REQ_OBD=SYMPTOM  (access by symptom)
                  Then  For each bit i which is set in ERR_SYM_DTC[IDX]
                                software structure fill up with
                                ID_ERR_DTC_XX[ 0 ] [ i ]
                        Endfor
                  Else  (access by failure; NC_ERR_DTC_REQ_OBD = FAILURE)
                        software structure fill up with ID_ERR_DTC_XX[0][4]
                  Endif
            Else  (Customer DTC is request ; LevelOfDtc = CUS)
                  If  NC_ERR_DTC_REQ_CUS = SYMPTOM (access by symptom)
                  Then  For each bit i which is set in ERR_SYM_DTC[IDX]
                                software structure fill up with
                                ID_ERR_DTC_XX[ NC_NR_DTC_FMT ] [ i ]
                        Endfor
                  Else  (access by failure; NC_ERR_DTC_REQ_CUS = FAILURE)
                        software structure fill up with
                        ID_ERR_DTC_XX
                        [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
                  Endif
            Endif
      Else        (particular case for misfiring failure NLC_TREAT_DIAG_MIS = 0)
            XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
            If    LevelOfDtc = LAW                          (OBD DTC is request)
            Then  If    SYM_CYL_DTC_XX [12]=1  (Random)
                  Then  software structure fill up with
                        ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR+1]
                  Endif
                  If    SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
                  Then  software structure fill up with
                        ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR ]
                  Endif
                  For each cylinder bit i which is set in SYM_CYL_DTC_XX
                          software structure fill up with
                          ID_ERR_DTC_MIS[ 0 ] [ i ]
                  Endfor
            Else  (Customer DTC is request ; LevelOfDtc = CUS)
                  If    SYM_CYL_DTC_XX [12]=1  (Random)
                  Then  software structure fill up with
                        ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR+1]
                  Endif
                  If    SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
                  Then  software structure fill up with
                        ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR ]
                  Endif
```

```
                For each cylinder bit i which is set in SYM_CYL_DTC_XX
                        software structure fill up with
                        ID_ERR_DTC_MIS[ 1 ] [ i ]
                Endfor
            Endif
        Endif
      Endif
   Endfor
 Endif
```

During all the software structure filling-in, if the software buffer is full then 2nd bit of ResultDtc is set to BUFFER_FULL, software process is aborted.

If there is duplicated DTC then only the oldest is returned

B.67.1.2 Request a DTC by DTC number

Description :

Syntax : ACTION_ERRM_ReadDtcByDtc (
IN <DtcIdentifier>,
IN <LevelOfDtc>,
OUT <ResultDtc>)

Parameter(in) :	DtcIdentifier	Number of the DTC (identifier) which is requested.
	LevelOfDtc	OBD or customer DTC is requested by the API
Parameter(out) :	ResultDtc	Boolean to say if there is the DTC in memory

Short Description : This API returns a result to learn if a DTC number of a certain level is stored in memory.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1^{st} bit of ResultDtc = NO_DTC_PRESENT

2^{nd} bit of ResultDTC = NO_BUFFER_FULL

```
For each failure IDX stored in dynamic memory,
    If   failure is not with status = CARB_MIS
    Then Condition for failure treatment is true
    Endif
    If   NLC_TREAT_DIAG_MIS = 1
    Then Condition for failure treatment is true
    Endif
    If   Condition for failure treatment is true
    Then If    LevelOfDtc = LAW                      (OBD DTC is request)
         Then If    NC_ERR_DTC_REQ_OBD = SYMPTOM   (access by symptom)
              Then  For each bit i which is set in ERR_SYM_DTC[IDX]
                          If    DtcIdentifier = ID_ERR_DTC_XX [0][i]
                          Then  1st bit of ResultDtc = DTC_PRESENT
                          Endif
                    Endfor
              Else  (access by failure ; NC_ERR_DTC_REQ_OBD = FAILURE)
                    If    DtcIdentifier = ID_ERR_DTC_XX [0][4]
                    Then  1st bit of ResultDtc = DTC_PRESENT
                    Endif
              Endif
         Else  (Customer DTC is request ; LevelOfDtc = CUS)
              If    NC_ERR_DTC_REQ_CUS = SYMPTOM   (access by symptom)
              Then  For each bit i which is set in ERR_SYM_DTC[IDX]
                    If    DtcIdentifier = ID_ERR_DTC_XX [NC_NR_DTC_FMT][i]
                    Then  1st bit of ResultDtc = DTC_PRESENT
                    Endif
                    Endfor
              Else  (access by failure ; NC_ERR_DTC_REQ_CUS = FAILURE)
                    If    DtcIdentifier = ID_ERR_DTC_XX
                          [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
                    Then  1st bit of ResultDtc = DTC_PRESENT
                    Endif
              Endif
         Endif
```

```
        Else      (particular case for misfiring failure NLC_TREAT_DIAG_MIS = 0)
          XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
          If        LevelOfDtc = LAW                        (OBD DTC is request)
          Then If       SYM_CYL_DTC_XX [12]=1               (Random)
               Then If       DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR+1]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Endif
               If       SYM_CYL_DTC_XX [13]=1               (Multiple cylinder)
               Then If       DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Endif
               For each cylinder bit i which is set inSYM_CYL_DTC_XX
                    If       DtcIdentifier = ID_ERR_DTC_MIS[0][i]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Endfor
          Else      (Customer DTC is request ; LevelOfDtc = CUS)
               If       SYM_CYL_DTC_XX [12]=1               (Random)
               Then If       DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR+1]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Else
               If       SYM_CYL_DTC_XX [13]=1               (Multiple cylinder)
               Then If       DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Endif
               For each cylinder bit i which is set inSYM_CYL_DTC_XX
                    If       DtcIdentifier = ID_ERR_DTC_MIS[1][i]
                    Then     1st bit of ResultDtc = DTC_PRESENT
                    Endif
               Endfor
          Endif
     Endif
Endfor
```

B.67.2 API for reading the status of Diagnostic Trouble Code

B.67.2.1 Request a list of status of DTCs by type of DTC

Description :

Syntax : ACTION_ERRM_ReadInfoByTypeOfDtc (
IN <TypeOfDtc>,
IN <InfoIdentifier>,
INOUT <ListOfDtcInfo>,
OUT < ResultDtc>)

Parameter(in) :	TypeOfDtc	Type of DTC which is requested by the API.
	InfoIdentifier	Information which is returned by the API
Parameter(out) :	ResultDtc	Boolean to say if there is the DTC in memory
Parameter(inout) :	ListOfDtcInfo	Software structure fill up with ListOfDtcInfo

Short Description : This API returns a diagnosis information. All the information returned are related to DTC which have the same type defined by the parameter TypeOfDtc and which are stored in memory.

Diagnosis information returned is defined by InfoIdentifier parameter.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1^{st} bit of ResultDtc = NO_DTC_PRESENT

2^{nd} bit of ResultDtc = NO_BUFFER_FULL

For each failure stored in dynamic memory,

If the failure belongs to group defined by the parameters TypeOfDtc

Then 1^{st} bit of ResultDtc = DTC_PRESENT

ListOfDtcInfo is fill up with data of this failure corresponding to InfoIdentifier.

Endif

EndFor

During all the software structure filling-in, if the software buffer is full then 2^{nd} bit of ResultDtc is set to BUFFER_FULL, software process is aborted.

B.67.2.2 Request a status of a DTC by DTC number

Description :

Syntax : ACTION_ERRM_ReadInfoByDtc (
IN <DtcIdentifier>,
IN <LevelOfDtc>,
IN <InfoIdentifier>,
INOUT <ListOfDtcInfo>,
OUT <ResultDtc>)

Parameter(in) :	DtcIdentifier	Number of the DTC (identifier) which is requested.
	LevelOfDtc	OBD or customer DTC is requested by the API
	InfoIdentifier	Information which is returned by the API
Parameter(out) :	ResultDtc	Boolean to say if there is the DTC in memory
Parameter(inout) :	ListOfDtcInfo	Software structure fill up with ListOfDtcInfo

Short Description : This API returns a structure of diagnosis information related to a DTC which is stored in dynamic memory.

Diagnosis information returned is defined by InfoIdentifier parameter.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1st bit of ResultDtc = NO_DTC_PRESENT

2nd bit of ResultDtc = NO_BUFFER_FULL

```
For each failure IDX stored in dynamic memory,
    If    failure is not with status = CARB_MIS
    Then Condition for failure treatment is true
    Endif
    If    NLC_TREAT_DIAG_MIS = 1
    Then Condition for failure treatment is true
    Endif
    If    Condition for failure treatment is true
    Then If     LevelOfDtc = LAW                      (OBD DTC is request)
         Then If     NC_ERR_DTC_REQ_OBD = SYMPTOM   (access by symptom)
              Then For each bit i which is set in ERR_SYM_DTC[IDX]
                        If     DtcIdentifier = ID_ERR_DTC_XX [0][i]
                        Then   1st bit of ResultDtc = DTC_PRESENT
                               ListOfDtcInfo is fill up with data of this failure
                               corresponding to InfoIdentifier.
                        Endif
                   Endfor
              Else (access by failure ; NC_ERR_DTC_REQ_OBD = FAILURE)
                   If     DtcIdentifier = ID_ERR_DTC_XX [0][4]
                   Then   1st bit of ResultDtc = DTC_PRESENT
                          ListOfDtcInfo is fill up with data of this failure
                          corresponding to InfoIdentifier.
                   Endif
              Endif
         Else (Customer DTC is request ; LevelOfDtc = CUS)
              If     NC_ERR_DTC_REQ_CUS = SYMPTOM   (access by symptom)
              Then For each bit i which is set in ERR_SYM_DTC[IDX]
                   If     DtcIdentifier = ID_ERR_DTC_XX [NC_NR_DTC_FMT][i]
                   Then   1st bit of ResultDtc = DTC_PRESENT
                          ListOfDtcInfo is fill up with data of this failure
                          corresponding to InfoIdentifier.
                   Endif
                   Endfor
              Else (access by failure ; NC_ERR_DTC_REQ_CUS = FAILURE)
                   If     DtcIdentifier = ID_ERR_DTC_XX
                          [NC_NR_DTC_FMT] [5-NC_NR_DTC_FMT]
                   Then   1st bit of ResultDtc = DTC_PRESENT
                          ListOfDtcInfo is fill up with data of this failure
                          corresponding to InfoIdentifier.
                   Endif
              Endif
         Endif
```

```
Else            (particular case for misfiring failure NLC_TREAT_DIAG_MIS = 0)
      XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
      If    LevelOfDtc = LAW                        (OBD DTC is request)
      Then  If    SYM_CYL_DTC_XX [13]=1             (Multiple cylinder)
            Then  If    DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR]
                  Then  1st bit of ResultDtc = DTC_PRESENT
                        ListOfDtcInfo is fill up with data of this failure
                        corresponding to InfoIdentifier.
                  Endif
            Endif
            If    SYM_CYL_DTC_XX [12]=1             (Random)
            Then  If    DtcIdentifier=ID_ERR_DTC_MIS[0][NC_CYL_NR+1]
                  Then  1st bit of ResultDtc = DTC_PRESENT
                        ListOfDtcInfo is fill up with data of this failure
                        corresponding to InfoIdentifier.
                  Endif
            Endif
            For each cylinder bit i which is set in SYM_CYL_DTC_XX
                  If DtcIdentifier = ID_ERR_DTC_MIS[0][i]
                  Then  1st bit of ResultDtc = DTC_PRESENT
                        ListOfDtcInfo is fill up with data of this failure
                        corresponding to InfoIdentifier.
                  Endif
            Endfor
```

```
        Else    (Customer DTC is request ; LevelOfDtc = CUS)
                If      SYM_CYL_DTC_XX [13]=1          (Multiple cylinder)
                Then    If      DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR]
                        Then    1st bit of ResultDtc = DTC_PRESENT
                                ListOfDtcInfo is fill up with data of this failure
                                corresponding to InfoIdentifier.
                        Endif
                Endif
                If      SYM_CYL_DTC_XX [12]=1          (Random)
                Then    If      DtcIdentifier=ID_ERR_DTC_MIS[1][NC_CYL_NR+1]
                        Then    1st bit of ResultDtc = DTC_PRESENT
                                ListOfDtcInfo is fill up with data of this failure
                                corresponding to InfoIdentifier.
                        Endif
                Endif
                For each cylinder bit i which is set in SYM_CYL_DTC_XX
                        If DtcIdentifier = ID_ERR_DTC_MIS[1][i]
                        Then    1st bit of ResultDtc = DTC_PRESENT
                                ListOfDtcInfo is fill up with data of this failure
                                corresponding to InfoIdentifier.
                        Endif
                Endfor
        Endif
    Endif
Endfor
```

During all the software structure filling-in, if the software buffer is full then 2nd bit of ResultDtc is set to BUFFER_FULL, software process is aborted.

When a failure is found in dynamic memory with the right DtcIdentifier, the treatment is stopped. That means that only the status of the first DTC found (oldest DTC) is returned.

B.67.2.3 Request a DTCLevel by an another DTCLevel

Description :

Syntax : ACTION_ERRM_ReadDtcLevelByDtcLevel (
IN <DtcIdentifier>,
IN <LevelOfDtc>,
INOUT <ListOfDtc>,
OUT <ResultDtc>)

Parameter(in) :	DtcIdentifier	Number of the DTC (identifier) which is requested.
	LevelOfDtc	OBD or customer DTC is requested by the API
Parameter(out) :	ResultDtc	Boolean to say if there is the DTC in memory
Parameter(inout) :	ListOfDtcInfo	Software structure fill up with ListOfDtcInfo

Short Description : This API returns the customer DTC from the Law DTC or the DTC Law from the DTC customer following the 'levelofDTC' parameter.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1st bit of ResultDtc = NO_DTC_PRESENT

2nd bit of ResultDtc = NO_BUFFER_FULL

For each failure IDX stored in dynamic memory,
  If failure is not with status = CARB_MIS
  Then Condition for failure treatment is true
  Endif
  If NLC_TREAT_DIAG_MIS = 1
  Then Condition for failure treatment is true
  Endif
  If Condition for failure treatment is true
  Then If LevelOfDtc = LAW (OBD DTC is request)
    Then If NC_ERR_DTC_REQ_OBD = SYMPTOM (access by symptom)
      Then For each bit i which is set in ERR_SYM_DTC[IDX]
        If DtcIdentifier = ID_ERR_DTC_XX [0][i]
        Then 1st bit of ResultDtc = DTC_PRESENT
          ListOfDtcInfo is fill up with
          If NC_NR_DTC_FMT = 0
          Then ID_ERR_DTC_XX[0][5]
          Else if NC_ERR_DTC_REQ_CUS=SYMPTOM
            Then ID_ERR_DTC_XX[1][i]

```
                        Else    ID_ERR_DTC_XX[1][4]
                        Endif
                    Endif
                    Exit
                Endif
            Endfor
        Else    (access by failure ; NC_ERR_DTC_REQ_OBD = FAILURE)

            If      DtcIdentifier = ID_ERR_DTC_XX [0][4]
            Then    1st bit of ResultDtc = DTC_PRESENT
                    ListOfDtc is fill up with
                    If      NC_ERR_DTC_REQ_CUS = SYMPTOM
                    (access by symptom for the customer code)
                    Then    If      only one bit is set in ERR_SYM_MEM[IDX]
                            Then    i = bit number which is set
                                    in ERR_SYM_MEM[IDX]
                                    ID_ERR_DTC_XX[NC_NR_DTC_FMT][i]
                            Else(many or no bit is set in ERR_SYM_MEM[IDX])
                                ID_ERR_DTC_XX
                                [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
                            Endif
                    else    (access by failure for the customer code)
                            ID_ERR_DTC_XX
                            [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
                    Endif
            Endif

        Endif
    Else    (Customer DTC is request ; LevelOfDtc = CUS)
        If      NC_ERR_DTC_REQ_CUS = SYMPTOM    (access by symptom)

        Then    For each bit i which is set in ERR_SYM_DTC[IDX]
            If      DtcIdentifier = ID_ERR_DTC_XX [NC_NR_DTC_FMT][i]
            Then    1st bit of ResultDtc = DTC_PRESENT
                    ListOfDtcInfo is fill up with
                    If      NC_NR_DTC_FMT = 0
                    Then    ID_ERR_DTC_XX[0][4]
                    Else    if      NC_ERR_DTC_REQ_OBD = SYMPTOM
                            Then    ID_ERR_DTC_XX[0][i]
                            Else    ID_ERR_DTC_XX[0][4]
                            Endif
                    Endif
                    Exit
            Endif
            Endfor

        Else    (access by failure ; NC_ERR_DTC_REQ_CUS = FAILURE)

            If      DtcIdentifier = ID_ERR_DTC_XX
                    [NC_NR_DTC_FMT] [5-NC_NR_DTC_FMT]

            Then    1st bit of ResultDtc = DTC_PRESENT
                    ListOfDtc is fill up with
                    If      NC_ERR_DTC_REQ_OBD = SYMPTOM
                            (access by symptom for the DTC law)
```

```
                                If      only one bit is set in ERR_SYM_MEM[IDX]
                                Then    i = bit number which is set
                                        in ERR_SYM_MEM[IDX]
                                        ID_ERR_DTC_XX[0][i]
                                Else    (many or no bit is set
                                        in ERR_SYM_MEM[IDX])
                                        ID_ERR_DTC_XX[0][4]
                                Endif
                        else    (access by failure for the customer code)
                                ID_ERR_DTC_XX[0][4]
                        Endif

                Endif

        Endif

    Endif


Else            (particular case for misfiring failure)

    XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4

    If      LevelOfDtc = LAW                        (OBD DTC is request)

    Then    If      SYM_CYL_DTC_XX [12]=1  (Random)
            Then    If     DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR+1]
                    Then    1st bit of ResultDtc = DTC_PRESENT
                            ListOfDtc is fill up with ID_ERR_DTC_MIS[1]
                            [NC_CYL_NR+1]
                            Exit
                    Endif
            Endif
            If      SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
            Then    If      DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR]
                    Then    1st bit of ResultDtc = DTC_PRESENT
                            ListOfDtc is fill up with ID_ERR_DTC_MIS[1]
                            [NC_CYL_NR]
                            Exit
                    Endif
            Endif

            For each cylinder bit i which is set inSYM_CYL_DTC_XX

                    If DtcIdentifier = ID_ERR_DTC_MIS[0][i]

                    Then    1st bit of ResultDtc = DTC_PRESENT
                            ListOfDtc is fill up with ID_ERR_DTC_MIS[1][i]
                            Exit
                    Endif
            Endfor
    Else    (Customer DTC is request ; LevelOfDtc = CUS)

            If      SYM_CYL_DTC_XX [12]=1  (Random)
            Then    If     DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR+1]
                    Then    1st bit of ResultDtc = DTC_PRESENT
                            ListOfDtc is fill up with ID_ERR_DTC_MIS[0]
                            [NC_CYL_NR+1]
                            Exit
```

```
                Endif
         Endif
         If     SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
         Then   If      DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR]
                Then    1st bit of ResultDtc = DTC_PRESENT
                        ListOfDtc is fill up with ID_ERR_DTC_MIS[0] [NC_CYL_NR]
                        Exit
                Endif
         Endif
         For each cylinder bit i which is set inSYM_CYL_DTC_XX
                If DtcIdentifier = ID_ERR_DTC_MIS[1][i]
                Then    1st bit of ResultDtc = DTC_PRESENT
                        ListOfDtc is fill up with ID_ERR_DTC_MIS[0][i]
                        Exit
                Endif
         Endfor
      Endif
   Endif
Endfor
```

During all the software structure filling-in, if the software buffer is full then 2nd bit of ResultDtc is set to BUFFER_FULL, software process is aborted.

When a failure is found in dynamic memory with the right DtcIdentifier, the treatment is stopped. That means that only the status of the first DTC found (oldest DTC) is returned.

B.67.3 API for reading the number of Diagnostic Trouble Code

B.67.3.1 Request the quantity of DTC with a certain type stored in memory

Description :

Syntax :	ACTION_ERRM_ReadQuantityOfDtc (IN <TypeOfDtc>, IN <LevelOfDtc>, INOUT <Quantity>, OUT <ResultQuantity>)	
Parameter(in) :	TypeOfDtc LevelOfDtc	Type of DTC which is requested by the API. OBD or customer DTC is requested by the API
Parameter(out) :	ResultQuantity	Boolean to say if there is at least 1 DTC in memory
Parameter(inout) :	Quantity	Software structure fill up with quantity of Dtc
Short Description :	This API returns the quantity of DTCs with a certain type, which are stored in memory. Quantity of DTCs is the number of failures in memory. If failures have same DTC values then they are counted once.	

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

Quantity = 0

ResultQuantity = NO_DTC_PRESENT

ttt

If PRM_TypeOfDtc = PERMANENT

Then

If CTR_ERR_PERM_NR > 0

Then

1st bit of ResultDtc = DTC_PRESENT

For IDX_PERM = 0 to CTR_ERR_PERM_NR -1

with XX = ERR_PERM[IDX_PERM]

If failure is not with status = CARB_MIS

Then Condition for failure treatment is true

Endif

If NLC_TREAT_DIAG_MIS = 1

Then Condition for failure treatment is true

Endif

If Condition for failure treatment is true

Then **If** LevelOfDtc = LAW (OBD DTC is request)

Then **If** NC_ERR_DTC_REQ_OBD=SYMPTOM (access by symptom)

Then Quantity = Quantity + j

with j = number of bit set in ERR_SYM_PERM[IDX_PERM]

Else (access by failure; NC_ERR_DTC_REQ_OBD = FAILURE)
Quantity = Quantity +1

Endif

Else (Customer DTC is request ; LevelOfDtc = CUS)

If NC_ERR_DTC_REQ_CUS = SYMPTOM (access by symptom)

Then Quantity = Quantity + j

with j = number of bit set in ERR_SYM_PERM[IDX_PERM]

Else (access by failure; NC_ERR_DTC_REQ_CUS = FAILURE)
Quantity = Quantity +1

Endif

Endif

Else (particular case for misfiring failure NLC_TREAT_DIAG_MIS = 0)
XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4

If LevelOfDtc = LAW (OBD DTC is request)
Then **If** SYM_CYL_PERM_XX [12]=1 (Random)
Then Quantity = Quantity +1

Endif

```
                    If      SYM_CYL_PERM_XX [13]=1        (Multiple cylinder)
                    Then    Quantity = Quantity +1
                    Endif
                    If      (many cylinder bit is set in SYM_CYL_PERM_XX)
                            Quantity = Quantity + j
                            with j = number of cylinder bit set in SYM_CYL_PERM_XX
                    Endif
                Else    (Customer DTC is request ; LevelOfDtc = CUS)
                    If      SYM_CYL_PERM_XX [12]=1        (Random)
                            Then   Quantity = Quantity +1
                    Endif
                    If      SYM_CYL_PERM_XX [13]=1        (Multiple cylinder)
                    Then    Quantity = Quantity +1
                    Endif
                    If      (many cylinder bit is set in SYM_CYL_PERM_XX)
                            Quantity = Quantity + j
                            with j = number of cylinder bit set in SYM_CYL_PERM_XX
                    Endif
                Endif
            Endif
          Endfor
       Endif
  Else
       For each failure IDX stored in dynamic memory,
          If        the failure belongs to failures group defined by the parameter TypeOfDtc
          Then      ResultQuantity = DTC_PRESENT
               If failure is not with status = CARB_MIS
               Then Condition for failure treatment is true
               Endif
               If  NLC_TREAT_DIAG_MIS = 1
               Then Condition for failure treatment is true
               Endif
               If Condition for failure treatment is true
               Then    If LevelOfDtc = LAW                (OBD DTC is request)
                 Then    If NC_REQ_OBD = SYMPTOM          (access by symptom)
                    Then    Quantity  = Quantity + j
                            with j = number of bit set in ERR_SYM_DTC_IDX
                    Else    (access by failure ; NC_REQ_OBD = FAILURE )
                            Quantity = Quantity +1
                    Endif
                 Else    (Customer DTC is request ; LevelOfDtc = CUS)
                    If      NC_REQ_CUS = SYMPTOM          (access by symptom)
```

```
                    Then  Quantity  = Quantity + j
                          with j = number of bit set in ERR_SYM_DTC_IDX
                    Else  (access by failure ; NC_REQ_OBD = FAILURE )
                          Quantity = Quantity +1
                    Endif
              Endif

        Else  (particular case for misfiring failure)
              XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
              If    LevelOfDtc = LAW                    (OBD DTC is request)
              Then  If    SYM_CYL_DTC_XX [12]=1  (Random)
                    Then  Quantity = Quantity +1
                    Endif
                    If    SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
                    Then  Quantity = Quantity +1
                    Endif
                    If    (many cylinder bit is set in SYM_CYL_DTC_XX)
                          Quantity = Quantity + j
                          With j  = number of cylinder bit set in SYM_CYL_DTC_XX
                    Endif

              Else  (Customer DTC is request ; LevelOfDtc = CUS)
                    If    SYM_CYL_DTC_XX [12]=1  (Random)
                    Then  Quantity = Quantity +1
                    Endif
                    If    SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
                    Then  Quantity = Quantity +1
                    Endif
                    If    (many cylinder bit is set in SYM_CYL_DTC_XX)
                          Quantity = Quantity + j
                          With j  = number of cylinder bit set in SYM_CYL_DTC_XX
                    Endif
              Endif
        Endif
     Endif
   EndFor
Endif
```

For each duplicated DTC, the corresponding quantity will be subtracted. Duplicated DTC will be counted once.

B.67.4 API for reading the Freeze Frame

B.67.4.1 Request a freeze frame with a certain type by DTC

Description :

Syntax : ACTION_ERRM_ReadFrfByDtc(
IN <TypeOfFF>,
IN <FFIdentifier>,
IN <DtcIdentifier>,
IN < LevelOfDtc>
INOUT <Frf>
OUT <ResultFrf>)

Parameter(in) :	TypeOfFF	Type of freeze frame which are request by the API
	FFIdentifier	Identifier of freeze frame requested
	DtcIdentifier	Identifier of DTC of the freeze frame requested
	LevelOfDtc	OBD or customer DTC is requested by the API
Parameter(out) :	ResultFrf	Boolean to say if requested freeze frame is present
Parameter(inout) :	Frf	Software structure fill with Freeze frame

Short Description : This API returns a freeze frame or a part of freeze frame.

With the parameter TypeOfFF, some different part of a freeze frame (defined by the law or by the customer - including set or not) can be returned.

The parameter FFIdentifier is not used (reserved for future improvement).

The freeze frame can be requested :

- by DTC ; in this case the parameters DtcIdentifier and LevelOfDtc should be fulfilled.
- for the CARB/EOBD request (mode 02h) ; in this case, the parameter DtcIdentifier has to be set to NO_DTC, the parameter TypeOfFF has to be set to LAW, and the parameter FFIdentifier has to be set to FIRST.

  The freeze frame associated to the oldest Carb failure among highest priority Carb failure is returned.

See definitions of freeze frame in module “Environmental data” module in “Error management” aggregate.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

1st bit of ResultFrf = NO_FRF_PRESENT

2nd bit of ResultFrf = NO_BUFFER_FULL

If (FFIdentifier = FIRST **and** TypeOfFF = LAW **and** DtcIdentifier = NO_DTC)

Then (Law freeze frame is requested)

If NLC_OBD_FRF_PND = 0

Then

Find oldest Carb/EOBD failure (emission relevant, LC_OBD_ERR = 1) failure stored in dynamic memory and confirmed (LV_ERR_CFM[IDX] = 1). If a misfire or fuel system failure is in the memory with LV_ERR_CFM[IDX] = 1, then the oldest of these failures must be chosen.

Else

Find oldest Carb/EOBD failure (emission relevant, LC_OBD_ERR = 1) failure stored in dynamic memory and confirmed (LV_ERR_CFM[IDX] = 1). If a misfire or fuel system failure is in the memory with LV_ERR_CFM[IDX] = 1, then the oldest of these failures must be chosen.

If this failure isn't existing, find oldest Carb/EOBD failure (emission relevant, LC_OBD_ERR = 1) failure stored in dynamic memory and pending (LV_ERR_PND[IDX] =1). If a misfire or fuel system failure is in the memory with LV_ERR_PND[IDX] = 1, then the oldest of these failures must be chosen.

Endif

If failure is not with status = CARB_MIS

Then Condition for failure treatment is true

Endif

If NLC_TREAT_DIAG_MIS = 1

Then Condition for failure treatment is true

Endif

If Condition for failure treatment is true

Then **If** NC_ERR_DTC_REQ_OBD = SYMPTOM (access by symptom)

Then **If** only one bit is set in ERR_SYM_MEM[IDX]

Then i = bit number which is set in ERR_SYM_MEM[IDX]
ReturnDtc = ID_ERR_DTC_XX [0][i]

Else (many or no bit is set in ERR_SYM_MEM[IDX])
ReturnDtc = ID_ERR_DTC_XX [0][4]

Endif

Else (access by failure ; NC_ERR_DTC_REQ_OBD = FAILURE)

ReturnDtc = ID_ERR_DTC_XX [0][4]

Endif

Else (particular case for misfiring failure)

XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4

If SYM_CYL_MEM_XX [12]=1 (Random)

Then ReturnDtc = ID_ERR_DTC_MIS[0][NC_CYL_NR+1]

Else **if** SYM_CYL_MEM_XX [13]=1 (Multiple cylinder)
Or many cylinder bits are set in
Then ReturnDtc = ID_ERR_DTC_MIS[0][NC_CYL_NR]

Else (only one cylinder bit is set in SYM_CYL_MEM_XX)
i = bit number which is set in SYM_CYL_MEM_XX
ReturnDtc = ID_ERR_DTC_MIS[0][i]
Endif
Endif

Endif

Software structure is fill up with ReturnDtc + ENVD_OBD[u][IDX]

1st bit of ResultFrf = FRF_PRESENT

Else (others freeze frame is requested)

For each failure IDX stored in dynamic memory,

If failure is not with status = CARB_MIS

Then Condition for failure treatment is true

Endif

If NLC_TREAT_DIAG_MIS = 1

Then Condition for failure treatment is true

Endif

If Condition for failure treatment is true

Then **If** LevelOfDtc = LAW (OBD DTC is request)

Then **If** NC_ERR_DTC_REQ_OBD=SYMPTOM (access by symptom)

Then **For** each bit i which is set in ERR_SYM_DTC[IDX]

If DtcIdentifier = ID_ERR_DTC_XX[0][i]

Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Exit
Endif
Endfor
Else (access by failure; NC_ERR_DTC_REQ_OBD = FAILURE)

If DtcIdentifier = ID_ERR_DTC_XX [0][4]
Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Endif

Endif

Else (Customer DTC is request ; LevelOfDtc = CUS)

If NC_ERR_DTC_REQ_CUS=SYMPTOM (access by symptom)

Then **For** each bit i which is set in ERR_SYM_DTC[IDX]

If DtcIdentifier = ID_ERR_DTC_XX[NC_NR_DTC_FMT][i]

Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Exit
Endif
Endfor
Else (access by failure; NC_ERR_DTC_REQ_CUS = FAILURE)

If DtcIdentifier = ID_ERR_DTC_XX [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Endif

Endif

Endif

Else (particular case for misfiring failure)
XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4

If LevelOfDtc = LAW (OBD DTC is request)
Then **If** SYM_CYL_DTC_XX [12]=1 (Random)
Then **If** DtcIdentifier=ID_ERR_DTC_MIS[0][NC_CYL_NR+1]
Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Exit
Endif
Endif

If SYM_CYL_DTC_XX [13]=1 (Multiple cylinder)
Then **If** DtcIdentifier=ID_ERR_DTC_MIS[0][NC_CYL_NR]
Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Exit
Endif
Endif
For each cylinder bit i which is set in SYM_CYL_DTC_XX

If DtcIdentifier = ID_ERR_DTC_MIS[0][i]

Then 1st bit of ResultFrf = FRF_PRESENT
Software structure is filled-up with DtcIdentifier plus freeze frame of failure according parameter TypeOfFF
Exit
Endif
Endfor

```
                    Else    (Customer DTC is request ; LevelOfDtc = CUS)
                            If      SYM_CYL_DTC_XX [12]=1  (Random)
                            Then    If DtcIdentifier=ID_ERR_DTC_MIS[1][NC_CYL_NR+1]
                                    Then    1st bit of ResultFrf = FRF_PRESENT
                                            Software structure is filled-up with DtcIdentifier plus
                                            freeze frame of failure according parameter
                                            TypeOfFF
                                            Exit
                                    Endif
                            Endif
                            If       SYM_CYL_DTC_XX [13]=1  (Multiple cylinder)
                            Then    If DtcIdentifier=ID_ERR_DTC_MIS[1][NC_CYL_NR]
                                    Then    1st bit of ResultFrf = FRF_PRESENT
                                            Software structure is filled-up with DtcIdentifier plus
                                            freeze frame of failure according parameter
                                            TypeOfFF
                                            Exit
                                    Endif
                            Endif
                            For each cylinder bit i which is set in SYM_CYL_DTC_XX
                                    If DtcIdentifier = ID_ERR_DTC_MIS[1][i]
                                    Then    1st bit of ResultFrf = FRF_PRESENT
                                            Software structure is filled-up with DtcIdentifier plus
                                            freeze frame of failure according parameter
                                            TypeOfFF
                                            Exit
                                    Endif
                            Endfor
                    Endif
            Endif
    Endfor
Endif
```

"Software structure is filled-up with freeze frame of failure according parameter TypeOfFF" means :

(for failure IDX)

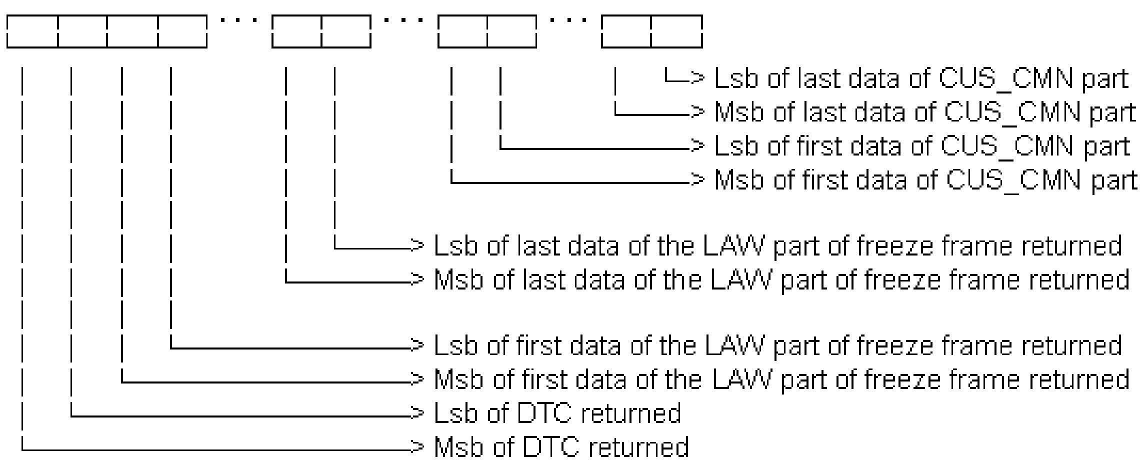
Software structure is filled-up with DTCIdentifier plus the softwate structure defined below :

If TypeOfFF = LAW
Then Software structure is fill up with ENVD_OBD[u][IDX]
Endif
If TypeOfFF = CUS_CMN
Then Software structure is fill up with ENVD_CUS_CMN[v][IDX]
Endif
If TypeOfFF = CUS_SET
Then Software structure is fill up with
ENVD_CUS_SET_CMN[w][z][IDX] + ENVD_CUS_SET_SPC[y][z][IDX]
Endif

Combination of parameter Type OfFF is possible. For instance if TypeOfFF is equal to LAW and CUS_CMN, both part are returned. In case of multiple part of freeze frame to returned, , the following order is respected :

- Part LAW of the freeze frame is returned first ;
- Part CUS_CMN of the freeze frame is then returned ;
- Part CUS_SET of the freeze frame is then returned .

Example : In the case of a request with TypeOfFF = LAW and CUS_CMN then the following structure is returned :



When a failure is found in dynamic memory with the right DtcIdentifier, the treatment is stopped. That means that only the freeze frame of the first DTC is found (oldest DTC) and returned by the API.

During all the software structure filling-in, if the software buffer is full then 2nd bit of ResultFrf is set to BUFFER_FULL, software process is aborted.

B.67.5 API for erasing Diagnostic information

B.67.5.1 Clear the failure associated to DTC with a certain type.

Description :

Syntax : ACTION_ERRM_ClrInfoByTypeOfDtc (
IN <TypeOfDtc>,
OUT < ResultClrInfo>)

Parameter(in) : TypeOfDtc Type of DTC which is requested by the API

Parameter(out) : ResultClrInfo Indicate if some DTC are cleared or not

Short Description : This API clear all the failure in dynamic memory associated to DTC with a certain type. It permits also to reinitialised all readiness code.

Remark :

For reason of design, and time resource, it's recommended to activate this API when engine stop. (Ref. SAE Mode $04).

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

ResultClrInfo = INFO_NOT_CLEARED

For each failure stored in dynamic memory,

If the failure belongs to failures group defined by the parameter TypeOfDtc

Then ACTION_ERRM_EraseErr (IN<IDX>, SYNCHRONIZATION<CALL>)

All data in static memory related to failure XX (with XX = F(IDX)) is also cleared

ACTION_ERRM_ClcPermanentByErr(XX)

ResultClrInfo = INFO_CLEARED

Endif

Endfor

In the case TypeOfDTC = All, then, the whole static memory shall be cleared (same behavior as clear failure memory using LC_ERR_FMY_CLR calibration bit)

{ Permanent calculation at ERRM clearing }

In case of TypeOfDTC = All, then call also ACTION_ERRM_ClcPermanentIniErrm( )

{ All readiness bit are reinitialised }

ACTION_ERRM_InitReadiness (SYNCHRONIZATION<CALL>)

Retransmit "failure erase service received"

B.67.5.2 Clear the failure associated to a DTC

Description :

Syntax :

```
ACTION_ERRM_ClrInfoByDtc (
        IN <DtcIdentifier>,
        IN <LevelOfDtc>,
        OUT < ResultClrInfo>)
```

Parameter(in) :	DtcIdentifier	Number of DTC which is requested (identifier)
	LevelOfDtc	OBD or customer DTC is requested by the API
Parameter(out) :	ResultClrInfo	to say if some DTC are cleared or not

Short Description : This API clear all the failure in dynamic memory associated to a DTC. The erasing is also applicable when there are identical DTC identifiers in the memory.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

```
ResultClrInfo = INFO_NOT_CLEARED
{ All readiness bit are reinitialised }
ACTION_ERRM_InitReadiness ( SYNCHRONIZATION<CALL> )
For each failure stored in dynamic memory do
    If   failure is not with status = CARB_MIS
    Then Condition for failure treatment is true
    Endif
    If   NLC_TREAT_DIAG_MIS = 1
    Then Condition for failure treatment is true
    Endif
    If   Condition for failure treatment is true
    Then If     LevelOfDtc = LAW                          (OBD DTC )
         Then If     NC_ERR_DTC_REQ_OBD = SYMPTOM    (access by symptom)
              Then   For each bit i which is set in ERR_SYM_DTC[IDX]
                         If DtcIdentifier = ID_ERR_DTC_XX[0][i]
                         Then ACTION_ERRM_EraseErr ( IN<IDX>,
                              SYNCHRONIZATION<CALL> )
                              All data in static memory related to failure XX (with
                              XX = F(IDX)) is also cleared
                              ACTION_ERRM_ClcPermanentByErr(XX)
                              ResultClrInfo = INFO_CLEARED
                              Exit
                         Endif
                     Endfor
```

```
                Else    { access by failure : NC_ERR_DTC_REQ_OBD = FAILURE }
                        If      DtcIdentifier = ID_ERR_DTC_XX[ 0 ][ 4 ] )
                        Then    ACTION_ERRM_EraseErr( IN<IDX>,
                                SYNCHRONIZATION<CALL> )
                                All data in static memory related to failure XX (with XX =
                                F(IDX)) is also cleared
                                ACTION_ERRM_ClcPermanentByErr(XX)
                        ResultClrInfo = INFO_CLEARED
                        Endif
                Endif
        Else    ( Customer DTC : LevelOfDtc = CUSTOMER)
                If      NC_ERR_DTC_REQ_CUS = SYMPTOM    (access by symptom)
                Then    For each bit i which is set in ERR_SYM_DTC[IDX]

                                If DtcIdentifier = ID_ERR_DTC_XX[NC_NR_DTC_FMT][i]

                                Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                                        SYNCHRONIZATION<CALL> )
                                        All data in static memory related to failure XX (with
                                        XX = F(IDX)) is also cleared
                                        ACTION_ERRM_ClcPermanentByErr(XX)
                                        ResultClrInfo = INFO_CLEARED
                                        Exit
                                Endif
                        Endfor
                Else    (access by failure : NC_ERR_DTC_REQ_OBD = FAILURE)
                        If      DtcIdentifier = ID_ERR_DTC_XX
                                [NC_NR_DTC_FMT][5-NC_NR_DTC_FMT]
                        Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                                SYNCHRONIZATION<CALL> )
                                All data in static memory related to failure XX (with XX =
                                F(IDX)) is also cleared
                                ACTION_ERRM_ClcPermanentByErr(XX)
                ResultClrInfo = INFO_CLEARED
                        Endif
                Endif
        Endif
Else    (particular case for misfiring failure)
        XX = F (IDX) with XX = MIS_A, MIS_B1 or MIS_B4
        If      LevelOfDtc = LAW                        (OBD DTC)
        Then    If      SYM_CYL_DTC_XX [12]=1           (Random)
                Then    If    DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR+1]
                        Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                                SYNCHRONIZATION<CALL> )
                                All data in static memory related to failure XX (with XX =
                                F(IDX)) is also cleared
                                ACTION_ERRM_ClcPermanentByErr(XX)
                                ResultClrInfo = INFO_CLEARED
                                Exit
                        Endif
                Endif
```

```
        If      SYM_CYL_DTC_XX [13]=1          (Multiple cylinder)
        Then    If      DtcIdentifier=ID_ERR_DTC_MIS [0] [NC_CYL_NR]
                Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                        SYNCHRONIZATION<CALL> )
                        All data in static memory related to failure XX (with XX =
                        F(IDX)) is also cleared
                        ACTION_ERRM_ClcPermanentByErr(XX)
                        ResultClrInfo = INFO_CLEARED
                        Exit
                Endif
        Endif
        For each cylinder bit i which is set inSYM_CYL_DTC_XX

                If DtcIdentifier = ID_ERR_DTC_MIS[0][i]

                Then    ACTION_ERRM_EraseErr( IN<IDX>,
                        SYNCHRONIZATION<CALL> )
                        All data in static memory related to failure XX (with XX =
                        F(IDX)) is also cleared
                        ACTION_ERRM_ClcPermanentByErr(XX)
                        ResultClrInfo = INFO_CLEARED
                        Exit
                Endif
        Endfor
Else     (Customer DTC )

        If      SYM_CYL_DTC_XX [12]=1          (Random)
        Then    If     DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR+1]
                Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                        SYNCHRONIZATION<CALL> )
                        All data in static memory related to failure XX (with XX =
                        F(IDX)) is also cleared
                        ACTION_ERRM_ClcPermanentByErr(XX)
                        ResultClrInfo = INFO_CLEARED
                        Exit
                Endif
        Endif

        If      SYM_CYL_DTC_XX [13]=1          (Multiple cylinder)
        Then    If      DtcIdentifier=ID_ERR_DTC_MIS [1] [NC_CYL_NR]
                Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                        SYNCHRONIZATION<CALL> )
                        All data in static memory related to failure XX (with XX =
                        F(IDX)) is also cleared
                        ACTION_ERRM_ClcPermanentByErr(XX)
                        ResultClrInfo = INFO_CLEARED
                        Exit
                Endif
        Endif
        For each cylinder bit i which is set inSYM_CYL_DTC_XX

                If DtcIdentifier = ID_ERR_DTC_MIS[1][i]

                Then    ACTION_ERRM_EraseErr ( IN<IDX>,
                        SYNCHRONIZATION<CALL> )
                        All data in static memory related to failure XX (with XX =
                        F(IDX)) is also cleared
```

```
                    ACTION_ERRM_ClcPermanentByErr(XX)
                    ResultClrInfo = INFO_CLEARED
                    Exit
                Endif
            Endfor
        Endif
    Endif
Endfor
```

When a failure is found in dynamic memory with the right DtcIdentifier, the treatment is not stopped. That means that all failure with the right DTC are erased.

B.67.6 API to manage the entry or exit of “Marked Mode”

B.67.6.1 Activate/deactivate the marked mode with clear failure

Description :

Syntax : ACTION_ERRM_ControlDtcSettings (IN <MarkedMode>)

Parameter(in) : MarkedMode State of marked mode

Short Description : This API permits to activate or deactivate the marked mode. In this mode all new failure which appears are marked and then erased when we leave this mode (useful in End Of Line phase). For errm inhibition, please refer to ACTION_ERRM_ActivateMarkedMode ().

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

If MarkedMode = ON

Then LV_MKD_MOD = 1

Else LV_MKD_MOD = 0

For each failure (IDX) stored in dynamic memory,

If LV_ERR_MKD[IDX] = 1

Then ACTION_ERRM_EraseErr (IN<IDX>)

All data in static memory related to failure XX (with XX = F(IDX)) is also cleared
ACTION_ERRM_ClcPermanentByErr(XX)

Endif

EndFor

Endif

B.67.6.2 Activate/deactivate the marked mode without clearing failures

Description :

Syntax : ACTION_ERRM_ActivateMarkedMode (IN <MarkedMode>)

Parameter(in) : MarkedMode State of marked mode ON or OFF

Short Description : This API shall be used to activate/deactivate the marked mode. In this mode, each new occuring failure is marked.
This is useful for ERRM inhibition. This inhibition may be used during a reprogramming phase.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

If MarkedMode = ON

Then LV_MKD_MOD = 1

Else LV_MKD_MOD = 0

Endif

B.67.6.3 Deactivation of marked mode (security mechanism)

Description:Purpose of this module is to add a security mechanism regarding marked mode. Because marked mode permits to inhibit error management, a strategy shall be introduced to avoid wrong inhibition.

This strategy is to deactivate marked mode when a at the end of driving cycle is detected. After driving cycle ends, reactivation of marked mode is the not supportedor at the next reset.

Application conditions:

Initialisation: none

Recurrence: none

Activation: at DCOFF event **or** at RST event

Deactivation: none

Formula section:

If LV_MKD_MOD = ON

Then

MarkedMode = OFF

ACTION_ERRM_ ControlDtcSettings (IN <MarkedMode >)

B.67.7 API for reading readiness code

B.67.7.1 Read the readiness code information

Description :

Syntax : ACTION_ERRM_ReadReadinessCode (
INOUT <ReadinessCode>,
OUT <ResultReadinessCode>)

Parameter(inout) : ReadinessCode   Software structure fill with readiness code

Short Description : This API calculate and returns the readiness code information in the software structure.

Readiness code information are build from readiness flags.

Please see “Readiness Code” module of the “Error management” aggregate for more details.

Application conditions:

Deactivation : -

Initialization: -

At failure erasure and saved ram lost
LV_READY_MISF_KOBD = 1, LV_READY_FSD_KOBD = 1

Recurrence : -

Formula section :

ResultReadinessCode = NEGATIVE_RESPONSE

(Calculation of STATE_READY_OBD_1)

STATE_READY_OBD_1 :

```
 ┌─┬─┬─┬─┬─┬─┬─┬─┐
 └─┴─┴─┴─┴─┴─┴─┴─┘
  | | | | | | | └─> bit0 : Misfire monitoring supported
  | | | | | | └───> bit1 : Fuel system monitoring supported
  | | | | | └─────> bit2 : Comprehensive component monitoring supported
  | | | | └───────> bit3 : GS/DS vehicle (NLC_OBD_DSL = 1 for DS, 0 for GS)
  | | | └─────────> bit4 : Misfire monitoring status
  | | └───────────> bit5 : Fuel system monitoring status
  | └─────────────> bit6 : Comprehensive component monitoring status
  └───────────────> bit7 : Reserved
```

4 less significant bit of STATE_READY_OBD_1 = 4 less significant bit of C_STATE_READY_OBD_1
4 most significant bit of STATE_READY_OBD_1 = 4 less significant bit of C_STATE_READY_OBD_1

STATE_READY_OBD_2:

```
[ | | | | | | | ]    GS name / DS name
 | | | | | | | └─>   bit 0 : Catalyst / NMHC Catalyst monitoring
 | | | | | | └───>   bit 1 : Heated catalyst / NOx Cat-Adsorber monitoring
 | | | | | └─────>   bit 2 : Evaporative system monitoring / reserved
 | | | | └───────>   bit 3 : Secondary air system / Boost pressure monitoring
 | | | └─────────>   bit 4 : A/C system refrigerant monitoring
 | | └───────────>   bit 5 : Oxygen sensor / Exhaust Gas sensor monitoring
 | └─────────────>   bit 6 : Oxygen sensor heater / PM Filter monitoring
 └───────────────>   bit 7 : EGR system monitoring
```

STATE_READY_OBD_2 = C_STATE_READY_OBD_2

<u>The readiness code calculation is performed in two steps:</u>

1. setting of readiness of a specific group CARB_YY (e.g. CARB_SA, CARB_EVAP), depending on the dynamic Error Management. If at least one failure XX is confirmed with bit 4 of WAL_CONF_XX = 1 of the group CARB_YY, then the readiness of this group can be set to ready.
2. setting of readiness of a specific group CARB_YY, depending on the failure XX individual readiness information LV_READY_XX. E.g. in case of a failure free system only step 2 is used for the readiness code calculation.

<u>Step 1:</u>

If CTR_ERR_DYN_NR != 0 (at least one failure stored in Error Management)

Then **For** IDX = 0 to CTR_ERR_DYN_NR – 1

If bit 4 of WAL_CONF_XX = 1 (failure is emission relevant)
with XX = DIAG_INST[IDX]

and LV_ERR_CFM[IDX] = 1 (failure is confirmed)

Then case CARB_YY for failure XX

CARB_MIS: Bit 4 of STATE_READY_OBD_1 = 0
CARB_FSD: Bit 5 of STATE_READY_OBD_1 = 0
CARB_CC: Bit 6 of STATE_READY_OBD_1 = 0
CARB_CAT: Bit 0 of STATE_READY_OBD_2 = 0
CARB_HC: Bit 1 of STATE_READY_OBD_2 = 0
CARB_EVAP: Bit 2 of STATE_READY_OBD_2 = 0
CARB_SA: Bit 3 of STATE_READY_OBD_2 = 0
CARB_AC: Bit 4 of STATE_READY_OBD_2 = 0
CARB_LS: Bit 5 of STATE_READY_OBD_2 = 0

CARB_LSH: Bit 6 of STATE_READY_OBD_2 = 0

CARB_EGR: Bit 7 of STATE_READY_OBD_2 = 0

Endif

Endfor

Endif

<u>Step 2:</u>

<u>Distinction between GS vehicle and DS vehicle</u>

Bit 3 of STATE_READY_OBD_1 = NLC_OBD_DSL

Misfire monitoring status bit4 :

```
If          CONF_KOBD = 0
Then
            If      Bit 4 of C_STATE_READY_CMPL_OBD_1 = 0
            Then
                    if      ∑ (LV_READY_XX) = 0
                    Take into account all LV_READY_XX which fulfill the following condition:
                    this XX failure has the status CARB_MIS (see tables of failures) AND
                    the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

                    then    Bit 4 of STATE_READY_OBD_1 = 0
                    endif
            Else    Readiness for Misfire monitoring status is always set
                    Bit 4 of STATE_READY_OBD_1 = 0
            EndIf
Else        (CONF_KOBD = 1)
            if      LV_READY_CAT_DIAG = 0
            then    LV_READY_MISF_KOBD = 0
            else

                    if    Bit 4 of STATE_READY_OBD_1 = 0(from step 1)

                    then    LV_READY_MISF_KOBD = 0
            endif

            if      LV_READY_MISF_KOBD = 0
            then    Bit 4 of STATE_READY_OBD_1 = 0
            endif
endif
```

Fuel system monitoring status bit5 :

```
If          CONF_KOBD = 0
Then
            If      Bit 5 of C_STATE_READY_CMPL_OBD_1 = 0
            Then
                    if      ∑ (LV_READY_XX) = 0
                    Take into account all LV_READY_XX which fulfill the following condition:
                    this XX failure has the status CARB_FSD (see tables of failures) AND
                    the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
```

then Bit 5 of STATE_READY_OBD_1 = 0
endif
Else Readiness for Fuel system monitoring status is always set
Bit 5 of STATE_READY_OBD_1 = 0
EndIf
Else (CONF_KOBD = 1)
if LV_READY_CAT_DIAG = 0
then LV_READY_FSD_KOBD = 0
else

if Bit 5 of STATE_READY_OBD_1 = 0(from step 1)

then LV_READY_FSD_KOBD = 0
endif

if LV_READY_FSD_KOBD = 0
then Bit 5 of STATE_READY_OBD_1 = 0
endif
endif

Comprehensive component monitoring status bit6 :

If CONF_KOBD = 0
Then
If Bit 6 of C_STATE_READY_CMPL_OBD_1 = 0
Then
if ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_CC (see tables of failures) **AND**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
then Bit 6 of STATE_READY_OBD_1 = 0
endif
Else Readiness for Comprehensive component status is always set
Bit 6 of STATE_READY_OBD_1 = 0
EndIf
Else (CONF_KOBD = 1)
Readiness for Comprehensive component status is always set
Bit 6 of STATE_READY_OBD_1 = 0
endif

Bit 7 of STATE_READY_OBD_1 is unchanged
(Calculation of STATE_READY_OBD_2)

Catalyst monitoring bit 0 :
If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_CAT (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)
Then Bit 0 of STATE_READY_OBD_2 = 0
Endif

Heated catalyst bit 1 :

If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_HC (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 1 of STATE_READY_OBD_2 = 0
Endif

Evaporative system monitoring bit2 :
If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_EVAP(see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 2 of STATE_READY_OBD_2 = 0
Endif

Secondary air system monitoring bit3 :
If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_SA (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 3 of STATE_READY_OBD_2 = 0
Endif

A/C system refrigerant monitoring bit4 :
If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_AC (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 4 of STATE_READY_OBD_2 = 0
Endif

Oxygen sensor monitoring bit5 :
If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_LS (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 5 of STATE_READY_OBD_2 = 0
Endif

Oxygen sensor heater monitoring bit6 :
If LV_READY_XX = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_LSH (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 6 of STATE_READY_OBD_2 = 0
Endif

EGR system monitoring bit7 :
If LV_READY_XX = 0
Take into account all LV_READY_XX which fulfill the following condition:

this XX failure has the status CARB_EGR (see tables of failures) **and** the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 7 of STATE_READY_OBD_2 = 0
Endif

ReadinessCode structure is filled up with : C_STATE_READY_OBD_2, STATE_READY_OBD_1 and STATE_READY_OBD_2.

ResultReadinessCode = POSITIVE_RESPONSE

Definition of Readiness Code:

In C_STATE_READY_OBD_1, a bit set to 1 means that corresponding test is supported according to the J1979 standard.

C_STATE_READY_OBD_1 :

```
┌─┬─┬─┬─┬─┬─┬─┬─┐
└─┴─┴─┴─┴─┴─┴─┴─┘
 │ │ │ │ │ │ │ └─>  bit0 : Misfire monitoring supported
 │ │ │ │ │ │ └───>  bit1 : Fuel system monitoring supported
 │ │ │ │ │ └─────>  bit2 : Comprehensive component monitoring supported
 │ │ │ │ └───────>  bit3 : GS/DS vehicle (NLC_OBD_DSL = 1 for DS, 0 for GS)
 │ │ │ └─────────>  bit4 = 0 (reserved)
 │ │ └───────────>  bit5 = 0 (reserved)
 │ └─────────────>  bit6 = 0 (reserved)
 └───────────────>  bit7 = 0 (reserved)
```

With C_STATE_READY_OBD_CMPL_OBD_1, it's possible to set directly Misfire, Fuel system, and comprehensive component readiness codes to "ready" status.

In C_STATE_READY_OBD_2, a bit set to 1 means that corresponding test is supported according to the J1979 standard.

C_STATE_READY_OBD_2:

```
┌─┬─┬─┬─┬─┬─┬─┬─┐   GS name / DS name
└─┴─┴─┴─┴─┴─┴─┴─┘
 │ │ │ │ │ │ │ └─>  bit 0 : Catalyst / NMHC Catalyst monitoring
 │ │ │ │ │ │ └───>  bit 1 : Heated catalyst / NOx Cat-Adsorber monitoring
 │ │ │ │ │ └─────>  bit 2 : Evaporative system monitoring / reserved
 │ │ │ │ └───────>  bit 3 : Secondary air system / Boost pressure monitoring
 │ │ │ └─────────>  bit 4 : A/C system refrigerant monitoring
 │ │ └───────────>  bit 5 : Oxygen sensor / Exhaust Gas sensor monitoring
 │ └─────────────>  bit 6 : Oxygen sensor heater / PM Filter monitoring
 └───────────────>  bit 7 : EGR system monitoring
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_STATE_READY_OBD_1	1	0...FFH	0...255	1	[-]
Status configuration for readiness code (continuous tests)					
C_STATE_READY_OBD_2	1	0...FFH	0...255	1	[-]
Status configuration for readiness code (non continuous tests)					
C_STATE_READY_CMPL_OBD_1	1	0...FFH	0...255	1	[-]
Readiness MIS, FSD, CC directly set to ready					

Calibration data detailled description for C_STATE_READY_CMPL_OBD_1 :

Readiness code calculation for CARB_MIS, CARB_FSD, CARB_CC diagnosis :

0: readiness of conserned group is calculated based on the readiness of each comprehensive component diagnostic

1: readiness of conserned group always indicates "ready"

C_STATE_READY_CMPL_OBD_1:

- bit0 : 0
- bit1 : 0
- bit2 : 0
- bit3 : 0
- bit4 : set directly readiness for Misfire monitoring
- bit5 : set directly readiness Fuel system monitoring
- bit6 : set directly readiness Comprehensive component monitoring
- bit7 : 0

B.67.7.2 Read the readiness code information for customer specific purpose

Description :

Syntax : ACTION_ERRM_ReadReadinessCodCus(
INOUT <ReadinessCode>,
OUT <ResultReadinessCode>)

Parameter(inout): ReadinessCode — Software structure filled-up with customer specific readiness code

Short Description : This API calculates and returns the readiness code information in the software structure.

Readiness code information are built thanks to readiness flags.

Please see “Readiness Code” module of the ERRM aggregate for more details.

Application conditions:

Deactivation : -

Initialization: -

Recurrence : -

Formula section :

ResultReadinessCode = NEGATIVE_RESPONSE

(Calculation of STATE_READY_OBD_3)

STATE_READY_OBD_3 :

```
|_|_|_|_|_|_|_|_|
 | | | | | | | └─> bit0 : for diagnosis classified as CARB_OTHER
 | | | | | | └───> bit1 : for diagnosis classified as NO_CARB
 | | | | | └─────> bit2 : 0
 | | | | └───────> bit3 : 0
 | | | └─────────> bit4 : 0
 | | └───────────> bit5 : 0
 | └─────────────> bit6 : 0
 └───────────────> bit7 : 0
```

STATE_READY_OBD_3 = C_STATE_READY_OBD_3

Diagnosis classified as CARB_OTHER bit0:

If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status CARB_OTHER (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 1 (CARB/EOBD failure)

Then Bit 0 of STATE_READY_OBD_3 = 0
Endif

Diagnosis classified as NO_CARB bit1

If ∑ (LV_READY_XX) = 0
Take into account all LV_READY_XX which fulfill the following condition:
this XX failure has the status NO_CARB (see tables of failures) **and**
the bit 4 of WAL_CONF_XX = 0 (CARB/EOBD failure)

Then Bit 1 of STATE_READY_OBD_3 = 0
Endif

ReadinessCode structure is filled-up with : C_STATE_READY_OBD_3, STATE_READY_OBD_3.

ResultReadinessCode = POSITIVE_RESPONSE

<u>Definition of Readiness Code:</u>

This readiness code does not follow existing J1979 standard.

C_STATE_READY_OBD_3 :

```
|_|_|_|_|_|_|_|_|
 | | | | | | | └─> bit0 : for diagnosis classified as CARB_OTHER
 | | | | | | └───> bit1 : for diagnosis classified as NO_CARB
 | | | | | └─────> bit2 : 0
 | | | | └───────> bit3 : 0
 | | | └─────────> bit4 : 0
 | | └───────────> bit5 : 0
 | └─────────────> bit6 : 0
 └───────────────> bit7 : 0
```

<u>Calibration data:</u>

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_STATE_READY_OBD_3	1	0...FFH	0...255	1	[-]
Status configuration for customer specific readiness code					

B.67.8 Rate-Based Monitoring - Communication interface

FUNCTION DESCRIPTION:

Description:

The Rate-Based Monitoring statistics shall be reported to the Scan-Tool via the Mode $09.

The data to be transmitted are :

- Ignition cycle counter
- General denominator
- For each group of monitors, individual numerator and denominator of the monitor which has the lowest In-Use Performance ratios.

These statistics are required and transmitted within Mode $09 for the monitors of the following groups :

- Catalyst bank 1
- Catalyst bank 2
- Oxygen sensor upstream bank 1
- Oxygen sensor upstream bank 2
- EGR system and/or VVT system
- Secondary air system
- Evaporating system
- Oxygen sensor downstream bank 1
- Oxygen sensor downstream bank 2

The selection of data to be displayed on Scan-Tool shall respect some rules :

- For each group of monitors, the individual numerator and denominator of monitor which has the lowest numerical In-Use Performance ratio shall be reported.
- If two or more specific monitors of the same group have identical ratios, the corresponding numerator and denominator for the specific monitor that has the highest denominator shall be reported.
- Depending on system configurations, some components might not exist (e.g. O2 Sensor for bank 2 on vehicles with 1 bank engine). In this case the numerator and denominator shall be set to 0.

B.67.8.1 API to select Rate-Based Monitoring data to be transmitted via Mode $09

The following paragraph descibes the data selection algorithm to be executed only once (for CPU load saving reason), upon external tool request via Mode $09, by calling an API.

Syntax : ACTION_ERRM_SelectRbmData (INOUT < ListRbmData >, OUT < ResultRbmData >)

Parameter (in) : No parameter

Parameter (out) : ResultRbmData

Parameter(inout) : ListRbmData

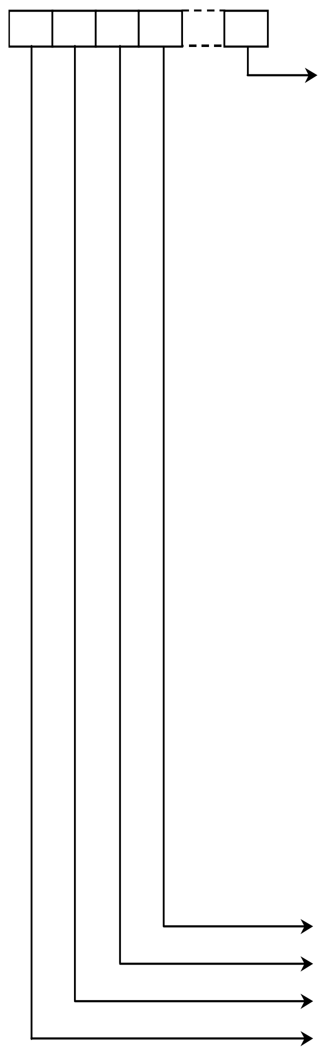
Short description : This API calculates and returns the Rate-Based Monitoring data to be transmitted to the Scan-Tool, when requested by the Mode 09$ (InfoType $08).

ResultRbmData : Indicates if the API has been executed or not.

Authorised values for ResultRbmDataSelection are :

- POSITIVE_RESPONSE : the call of the API is successful
- NEGATIVE_RESPONSE : the call of the API is unsuccessful

ListRbmData : Software structure filled-up with 16 counters of 2 bytes coming from Rate-Based Monitoring



Sec. O2 Sensor Monitor Cond. Encountered Counts Bank 2 (denominator – LSB byte)
Sec. O2 Sensor Monitor Cond. Encountered Counts Bank 2 (denominator - MSB byte)
Sec. O2 Sensor Monitor Completion Counts Bank 2 (numerator – LSB byte)
Sec. O2 Sensor Monitor Completion Counts Bank 2 (numerator - MSB byte)
Sec. O2 Sensor Monitor Cond. Encountered Counts Bank 1 (denominator - LSB byte)
Sec. O2 Sensor Monitor Cond. Encountered Counts Bank 1 (denominator - MSB byte)
Sec. O2 Sensor Monitor Completion Counts Bank 1 (numerator – LSB byte)
Sec. O2 Sensor Monitor Completion Counts Bank 1 (numerator – MSB byte)
EVAP Monitor Conditions Encountered Counts (denominator - LSB byte)
EVAP Monitor Conditions Encountered Counts (denominator - MSB byte)
EVAP Monitor Completion Condition Counts (numerator - LSB byte)
EVAP Monitor Completion Condition Counts (numerator - MSB byte)
AIR Monitor Conditions Encountered Counts (denominator - LSB byte)
AIR Monitor Conditions Encountered Counts (denominator - MSB byte)
AIR Monitor Completion Condition Counts (numerator - LSB byte)
AIR Monitor Completion Condition Counts (numerator - MSB byte)
EGR Monitor Conditions Encountered Counts (denominator - LSB byte)
EGR Monitor Conditions Encountered Counts (denominator – MSB byte)
EGR Monitor Completion Condition Counts (numerator - LSB byte)
EGR Monitor Completion Condition Counts (numerator - MSB byte)
O2 Sensor Monitor Conditions Encountered Counts Bank 2 (denominator – LSB byte)
O2 Sensor Monitor Conditions Encountered Counts Bank 2 (denominator - MSB byte)
O2 Sensor Monitor Completion Counts Bank 2 (numerator – LSB byte)
O2 Sensor Monitor Completion Counts Bank 2 (numerator - MSB byte)
O2 Sensor Monitor Conditions Encountered Counts Bank 1 (denominator - LSB byte)
O2 Sensor Monitor Conditions Encountered Counts Bank 1 (denominator - MSB byte)
O2 Sensor Monitor Completion Counts Bank 1 (numerator – LSB byte)
O2 Sensor Monitor Completion Counts Bank 1 (numerator – MSB byte)
Catalyst Monitor Conditions Encountered Counts Bank 2 (denominator - LSB byte)
Catalyst Monitor Conditions Encountered Counts Bank 2 (denominator - MSB byte)
Catalyst Monitor Completion Counts Bank 2 (numerator - LSB byte)
Catalyst Monitor Completion Counts Bank 2 (numerator - MSB byte)
Catalyst Monitor Conditions Encountered Counts Bank 1 (denominator - LSB byte)
Catalyst Monitor Conditions Encountered Counts Bank 1 (denominator - MSB byte)
Catalyst Monitor Completion Counts Bank 1 (numerator - LSB byte)
Catalyst Monitor Completion Counts Bank 1 (numerator - MSB byte)
Ignition Counter (LSB byte)
Ignition Counter (MSB byte)
OBD Monitoring Conditions Encountered Counts (general denominator - LSB byte)
OBD Monitoring Conditions Encountered Counts (general denominator - MSB byte)

Application conditions:

Initialization : Initialize the ListRbmData software structure with all counters set to 00h

ResultRbmData = NEGATIVE_RESPONSE

Recurrence : -

Activation : at Action call

Deactivation : -

Formula section:

If NLC_OBD_RBM_ENA = 1

Then

Fill-in ListRbmData software structure with *Ignition counter* = CTR_IGK_CYC_RBM and *General denominator* = CTR_CDN_OBD_RBM

Call ACTION_ERRM_SelectRbmByGroup (IN<NC_RBM_CAT_1>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN<NC_RBM_CAT_2>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_LS_UP_1>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_LS_UP_2>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_EGR_VVT>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_SA>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListOfRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_EVAP>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_LS_DOWN_1>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListRbmData with ListOfRbmDataByGroup

Call ACTION_ERRM_SelectRbmByGroup (IN< NC_RBM_LS_DOWN_2>, INOUT<ListOfRbmDataByGroup>, OUT<ResultRbmDataSelection>, SYNCHRONIZATION <CALL>)

Fill-in ListRbmData with ListOfRbmDataByGroup

Else

ListRbmData = 00000000000000000000000000000000h

Endif

ResultRbmData = POSITIVE_RESPONSE

B.67.8.2 API to select RBM statistic with lowest in-use performance ratio within one single group

General information:

This API shall be used to determine, within one single group (see specification “Rate-Based Monitoring (Appl. Inc.)”), the counters of the monitor which has the lowest in-use performance ratio.

Description:

Syntax : ACTION_ERRM_SelectRbmByGroup (

IN < GroupName >

INOUT < ListRbmDataByGroup >,

OUT < ResultRbmDataByGroup >)

Parameter (in) : GroupName

Parameter (out) : ResultRbmDataByGroup

Parameter(inout) : ListRbmDataByGroup

Short description : This API calculates and returns the Rate-Based Monitoring data with lowest ratios, within one single group.

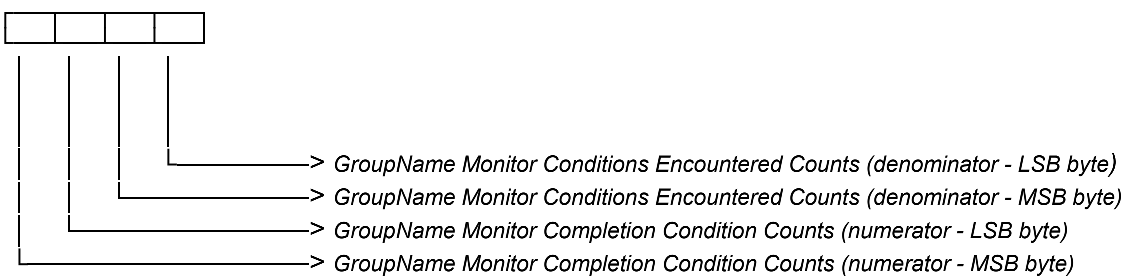
GroupName : Group name requested. See specification “Rate-Based Monitoring (Appl. Inc.)”

ResultRbmDataByGroup : Indicates if the API has been executed or not.

Authorised values for ResultRbmDataByGroup are :

- POSITIVE_RESPONSE : the call of the API is successful
- NEGATIVE_RESPONSE : the call of the API is unsuccessful

ListRbmDataByGroup : Software structure filled-up with the Rate-Based Monitoring counters of 2 bytes related to the requested group name.



Application conditions:

Initialization : -

Recurrence : -

Activation : at Action call

Deactivation : -

Formula section:

{ For CPU optimization reasons, this implementation assumes that Groups Names are ordered as indicated in the Rate-Based Monitoring (Appl. Inc.) file : Groups Names order must be respected and monitors belonging to a same group shall be joined }

{ If this requirement isn't respected, data sent to the Scan-Tool might be erroneous }

ResultRbmDataByGroup = NEGATIVE_RESPONSE

m = 0

While m ≤ (NC_NR_DIAG_RBM – 1) **do**

If(1) m monitor belongs to GroupName

Then(1)

If(2) CTR_CDN_RBM[m] ≠ 0 { denominator is different from 0 }

Then(2)

RATIO_TMP_RBM = CTR_COMP_RBM[m] / CTR_CDN_RBM[m]

Fill-in ListRbmDataByGroup with *numerator* = CTR_COMP_RBM[m] and *denominator* = CTR_CDN_RBM[m]

Else(2)

RATIO_TMP_RBM = FFFFh

Fill-in ListRbmDataByGroup with *numerator* = CTR_COMP_RBM[m] and *denominator* = 0

Endif(2)

IDX_TMP_RBM = m

For j = (m+1) to (NC_NR_DIAG_RBM – 1) **do**

If(3) j monitor belongs to GroupName

Then(3)

If(4) CTR_CDN_RBM[j] ≠ 0 { denominator is different from 0 }

Then(4)

{ case of one monitor with lower ratio than previous one found }

If(5) ((CTR_COMP_RBM[j] / CTR_CDN_RBM[j]) < RATIO_TMP_RBM)

Then(5)

RATIO_TMP_RBM = CTR_COMP_RBM[j] / CTR_CDN_RBM[j]

IDX_TMP_RBM = j

Fill-in ListRbmDataByGroup with *numerator* = CTR_COMP_RBM[j] and *denominator* = CTR_CDN_RBM[j]

Else(5)

{ case of two monitors with same ratios }

If(6) ((CTR_COMP_RBM[j] / CTR_CDN_RBM[j]) = RATIO_TMP_RBM)

Then(6)

If(7) CTR_CDN_RBM[j] > CTR_CDN_RBM[IDX_TMP_RBM]

Then(7)

Fill-in ListRbmDataByGroup with *numerator* = CTR_COMP_RBM[j] and *denominator* = CTR_CDN_RBM[j]

IDX_TMP_RBM = j

Endif(7)

Endif(6)

Endif(5)

Else(4)

{ denominator equals 0 }

Endif(4)

Endif(3)

Endfor

Exit while loop

Else(1)

{ m monitor doesn't belong to GroupName }

m = m +1

Fill-in ListRbmDataByGroup with *numerator* = 0 and *denominator* = 0

Endif(1)

Endwhile

ResultRbmDataByGroup = POSITIVE_RESPONSE

B.67.8.3 API to clear all Rate-Based Monitoring statistics

General information:

Regulation text :
"Each number shall be reset to zero only when a non-volatile memory reset occurs (e.g., reprogramming event, etc.) or, if the numbers are stored in keep-alive memory (KAM), when KAM is lost due to an interruption in electrical power to the control module (e.g., battery disconnect, etc.). Numbers may not be reset to zero under any other circumstances including when a scan tool command to clear fault codes or reset KAM is received."

The service provided in this paragraph shall be used to clear all the Rate-Based Monitoring statistics after a reprogramming session of ECU software and/or calibration data.

This service shall not be called when a Scan Tool command to clear faults (Mode 04$) is received.

Description:

Syntax : ACTION_ERRM_ClearRbmStatistics (OUT < ResultClrInfo>)

Parameter(in) : No parameter.

Parameter(out) : ResultClrInfo to inform if RBM statistics are cleared or not

Short Description : This API clear all the Rate-Based Monitoring statistics

Remark : For design and CPU resource reasons, this API shall be called while engine is stopped.

Application conditions:

Initialization : -

Recurrence : -

Activation : at Action call

Deactivation : -

Formula section:

ResultClrInfo = STATISTICS_NOT_CLEARED

CTR_IGK_CYC_RBM = 0

CTR_CDN_OBD_RBM = 0

For m = 0 to (NC_NR_DIAG_RBM –1) **do**

CTR_CDN_RBM[m] = 0

CTR_COMP_RBM[m] = 0

Endfor

ResultClrInfo = STATISTICS_CLEARED

Configuration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
NC_ERR_DTC_REQ_OBD	-	0...1H	0...1	1	[-]
This bit shall be set to 0 (SYMPTOM) to read OBD DTC by symptomThis bit shall be set to 1 (FAILURE) to read OBD DTC by failure					
NC_ERR_DTC_REQ_CUS	-	0...1H	0...1	1	[-]
This bit shall be set to 0 (SYMPTOM) to read customer DTC by symptomThis bit shall be set to 1 (FAILURE) to read customer DTC by failure					
NLC_OBD_FRF_PND	-	0...1H	0...1	1	[-]
Defines the EOBD/CARB freeze frame strategy the following ways :0 : EOBD/CARB freeze frame data of Mode 02h will be returned after detection of a confirmed failure .1 : EOBD/CARB freeze frame data of Mode 02h will be returned :after detection of a confir					

Configuration data detailed description:

NC_ERR_DTC_REQ_OBD : This bit shall be set to 0 (SYMPTOM) to read OBD DTC by symptom.
This bit shall be set to 1 (FAILURE) to read OBD DTC by failure.

NC_ERR_DTC_REQ_CUS : This bit shall be set to 0 (SYMPTOM) to read Customer DTC by symptom.
This bit shall be set to 1 (FAILURE) to read Customer DTC by failure.

NLC_OBD_FRF_PND : Definition the EOBD/CARB freeze frame strategy, as follow :
0 : EOBD/CARB freeze frame data of Mode 02h will be returned after detection of a confirmed failure.
1 : EOBD/CARB freeze frame data of Mode 02h will be returned :
- after detection of a confirmed emission relevant failure if confirmed in memory
- after pending emission relevant detection if not confirmed in memory

B.68 Communication Interface (Appl. Inc.)

Input data:

NC_ERR_DTC_REQ_OBD	NC_ERR_DTC_REQ_CUS		

FUNCTION DESCRIPTION:

General information:

Description:

This file defines which DTC number to sent to the communication tool in case of DTC request. To follow specific customer, this file is a template to be fullfilled by project team.

All data in italic are comments.

Caracteristics	Value
DTC number to sent to CARB communication tool	NC_ERR_DTC_REQ_OBD = 0 Example : set this bit to 0 to sent symptom OBD DTC set this bit to 1 to sent global OBD DTC
DTC number to sent to customer communication tool	NC_ERR_DTC_REQ_CUS = 0 Example : set this bit to 0 to sent symptom customer DTC set this bit to 1 to sent global customer DTC

B.69 Diagnosis Trouble Code Management (DTC)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_SYM_MEM[NC_NR_ERR_DYN]	V/O/S	0...FH	0...15	1	[-]
Memorized detected symptom of the failure IDX (for the DTC building)					
ERR_SYM_LST[NC_NR_ERR_DYN]	V/O/S	0...FH	0...15	1	[-]
Memorized the last detected symptom of the failure IDX (for the DTC building)					
#IF NLC_TREAT_DIAG_MIS = 0					
SYM_CYL_MEM_MIS_A	V/O/S	0...FFFFH	0...65535	1	[-]
memorized cylinder with misfire A (for the DTC building)					
SYM_CYL_MEM_MIS_B1	V/O/S	0...FFFFH	0...65535	1	[-]
memorized cylinder with misfire B1 (for the DTC building)					
SYM_CYL_MEM_MIS_B4	V/O/S	0...FFFFH	0...65535	1	[-]
memorized cylinder with misfire B4 (for the DTC building)					
SYM_CYL_LST_MIS_A	V/O/S	0...FFFFH	0...65535	1	[-]
Memorized the last cylinder of the misfire A failure (for the DTC building)					
SYM_CYL_LST_MIS_B1	V/O/S	0...FFFFH	0...65535	1	[-]
Memorized the last cylinder of the misfire B1 failure (for the DTC building)					
SYM_CYL_LST_MIS_B4	V/O/S	0...FFFFH	0...65535	1	[-]
Memorized the last cylinder of the misfire B4 failure (for the DTC building)					
#ENDIF					
ERR_DTC[NC_NR_ERR_DYN]	V/O/S	0...FFFFH	0...65535	1	[-]
DTC code stored in memory for failure IDX					

Input data:

ERR_SYM_XX	SYM_CYL_MIS_A	SYM_CYL_MIS_B1	SYM_CYL_MIS_B4
NC_CYL_NR	NLC_TREAT_DIAG_MIS		

Export actions :

ACTION_ERRM_StoreDtc (IN <IDX>)
This action stores the memorized symptom of the first occurrence of the failure IDX to build the DTC
ACTION_ERRM_StoreDtcLst (IN <IDX>)
This action stores the memorized symptom of the last occurrence of the failure IDX to build the DTC
ACTION_ERRM_EraseDtc (IN <IDX>)
This action erases the memorized symptom of the failure IDX

FUNCTION DESCRIPTION:

General information:

Purpose of this module is to catch some data (ERR_SYM_XX) when a failure occurs. Then, these data (ERR_SYM_MEM[IDX] and ERR_SYM_LST_[IDX]) are used to generate a code called DTC upon communication tool request (see Communication interface module). Additionally, for visualization (for communication tool, developpment purposes), a DTC code is stored in memory (ERR_DTC_[IDX]).

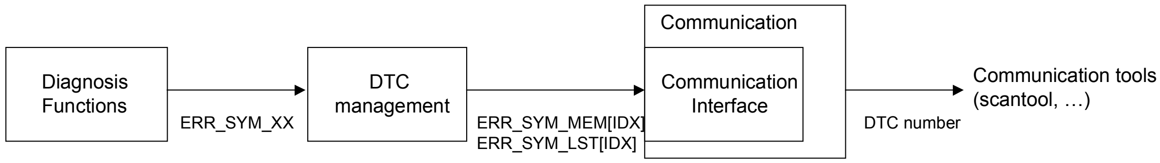
The DTC is a code used to identify the symptom of the failure or the failure (global failure). It is an interface between the diagnosis and the communication.

The customer and/or EOBD/CARB according to the standard J2012 define this DTC.

DTC encoding for regulation EOBD/CARB shall be done using J2012 definition (2 bytes interpreted [0000h to FFFFh] in hex, [P0000 to U3FFF] in J2012).

DTC encoding for customer is free. It can be in decimal, or applying J2012 definition, or any other encoding definition.

Signal flow diagram:



B.69.1 DTC Storage

B.69.1.1 DTC Storage / First occurence

Description:

When the failure IDX gets present (1^{st} occurrence), the detected of the first symptom is stored in the 2^{nd} layer memory by calling the action ACTION_ERRM_StoreDtc (IN<IDX>). Additionally, a DTC code is stored in dynamic memory for visualization :

- related to the symptom DTC if NC_ERR_DTC_CONF = 0
- related to the failure OBD DTC if NC_ERR_DTC_CONF = 1
- related to the customer specific symptom DTC if NC_ERR_DTC_CONF = 2
- related to the customer specific failure DTC if NC_ERR_DTC_CONF = 3

Syntax :	ACTION_ERRM_StoreDtc (IN <IDX>)	
Parameter (in) :	IDX	Index of failure in 2^{nd} layer memory to store the DTC
Parameter (out) :	-	
Short description :	This action stores the memorised symptom of the first occurrence of the failure IDX to build the DTC	

Application conditions:

Initialization: ERR_SYM_MEM[IDX] is restored from the NVMY

SYM_CYL_MEM_MIS_A, SYM_CYL_MEM_MIS_B1 and

SYM_CYL_MEM_MIS_B4 are restored from NVMY

In case of damaged NVMY, ERR_SYM_MEM[IDX] = 0

SYM_CYL_MEM_MIS_A = 0,

SYM_CYL_MEM_MIS_B1 = 0,

SYM_CYL_MEM_MIS_B4 = 0,

ERR_DTC[IDX] = 0

Recurrence: -

Activation: at Action request

Formula section:

```
If    NLC_TREAT_DIAG_MIS = 0
      (Misfire are treated by type (LV_ERR_MIS_A, MIS_B1 and MIS_B4 is defined))
Then
      For a misfiring diagnosis:
      (XX stands for MIS_A, MIS_B1, and MIS_B4) (Misfire by type NLC_TREAT_DIAG_MIS=0)
            SYM_CYL_MEM_XX = SYM_CYL_XX

            If    NC_ERR_DTC_CONF = 0 or 1 or 4                (J2012 DTC is stored)
            Then  If    SYM_CYL_MEM_XX [12] = 1        (Random)
                  Then  ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR+1 ]
                  Else  If    SYM_CYL_MEM_XX [13] = 1        (Multiple cylinder)
                              Or many cylinder bits are set in SYM_CYL_MEM_XX
                        Then  ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR ]
                        Else  (only one cylinder bit is set in SYM_CYL_MEM_XX)
                                    i = bit number which is set in SYM_CYL_MEM_XX
                                    ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ i ]
                        Endif
                  Endif
            Endif

            If    NC_ERR_DTC_CONF = 2 or 3 or 5 (customer encoded DTC is stored)
            Then  If    SYM_CYL_MEM_XX [12] = 1        (Random)
                  Then  ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR+1 ]
                  Else  If    SYM_CYL_MEM_XX [13] = 1        (Multiple cylinder)
                              Or many cylinder bits are set in SYM_CYL_MEM_XX
                        Then  ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR ]
                        Else  (only one cylinder bit is set in SYM_CYL_MEM_XX)
                                    i = bit number which is set in SYM_CYL_MEM_XX
                                    ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ i ]
                        Endif
                  Endif
            Endif
Endif
```

For others diagnosis :

(also for misfire diagnosis if treated by cylinder (LV_ERR_MIS_x is defined))

ERR_SYM_MEM[IDX] = ERR_SYM_XX

If NC_ERR_DTC_CONF = 0 or 4 (1st symptom J2012 DTC is stored)

Then **If** only one bit is set in ERR_SYM_MEM[IDX]

Then i = bit number which is set in ERR_SYM_MEM[IDX]

ERR_DTC[IDX] = ID_ERR_DTC_XX [0] [i] with XX = F(IDX)

Else (many bit is set in ERR_SYM_MEM[IDX])

ERR_DTC[IDX] = ID_ERR_DTC_XX [0] [4] with XX = F(IDX)

Endif

Endif

If NC_ERR_DTC_CONF = 1 (global J2012 DTC is stored)

Then ERR_DTC[IDX] = ID_ERR_DTC_XX [0] [4] with XX = F(IDX)

Endif

If NC_ERR_DTC_CONF = 2 or 5 (customer encoded DTC is stored)

Then **If** only one bit is set in ERR_SYM_MEM[IDX]

Then i = bit number which is set in ERR_SYM_MEM[IDX]

ERR_DTC[IDX] =

ID_ERR_DTC_XX [NC_NR_DTC_FMT] [i] with XX = F(IDX)

Else (many bit is set in ERR_SYM_MEM[IDX])

ERR_DTC[IDX] =

ID_ERR_DTC_XX [NC_NR_DTC_FMT] [5 - NC_NR_DTC_FMT]

with XX = F(IDX)

Endif

Endif

If NC_ERR_DTC_CONF = 3 (global customer encoded DTC is stored)

Then ERR_DTC[IDX] =

ID_ERR_DTC_XX [NC_NR_DTC_FMT] [5 - NC_NR_DTC_FMT]
with XX = F(IDX)

Endif

B.69.1.2 DTC Storage / Last occurence

Description:

For each occurence of the failure IDX (Last occurrence), the detected of the last symptom is stored in the 2nd layer memory by calling the action ACTION_ERRM_StoreDtcLst (IN<IDX>).

Syntax :	ACTION_ERRM_StoreDtcLst (IN<IDX>)	
Parameter (in) :	IDX	Index of failure in 2nd layer memory to store the DTC
Parameter (out) :	-	
Short description :	This action stores the memorised symptom of the last occurrence of the failure IDX to build the DTC	

Application conditions:

Initialization: ERR_SYM_LST[IDX] is restored from the NVMY

SYM_CYL_LST_MIS_A, SYM_CYL_LST_MIS_B1 and

SYM_CYL_LST_MIS_B4 are restored from NVMY

In case of damaged NVMY, ERR_SYM_LST[IDX] = 0

SYM_CYL_LST_MIS_A = 0,

SYM_CYL_LST_MIS_B1 = 0,

SYM_CYL_LST_MIS_B4 = 0,

Recurrence: -

Activation: at Action request

Formula section:

ERR_SYM_LST[IDX] = ERR_SYM_XX

If NLC_TREAT_MIS_DIAG = 0

(Misfire are treated by type)

then For a misfiring diagnosis :

(XX stands for MIS_A, MIS_B1 and MIS_B4)

SYM_CYL_LST_XX = SYM_CYL_ XX

endif

If NLC_TREAT_DIAG_MIS = 0

(Misfire are treated by type (LV_ERR_MIS_A, MIS_B1 and MIS_B4 is defined))

Then <u>For a misfiring diagnosis:</u>

```
If     NC_ERR_DTC_CONF = 4                    (J2012 DTC is stored)
Then   If     SYM_CYL_LST_XX [12] = 1 (Random)
       Then   ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR+1 ]
       Else   If     SYM_CYL_LST_XX [13] = 1 (Multiple cylinder)
                     Or many cylinder bits are set in SYM_CYL_LST_XX
              Then   ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ NC_CYL_NR ]
              Else   (only one cylinder bit is set in SYM_CYL_LST_XX)
                            i = bit number which is set in SYM_CYL_LST_XX
                            ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 0 ] [ i ]
              Endif
       Endif
Endif

If     NC_ERR_DTC_CONF = 5                    (customer DTC is stored)
Then   If     SYM_CYL_LST_XX [12] = 1 (Random)
       Then   ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR+1 ]
       Else   If     SYM_CYL_LST_XX [13] = 1 (Multiple cylinder)
                     Or many cylinder bits are set in SYM_CYL_LST_XX
              Then   ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ NC_CYL_NR ]
              Else   (only one cylinder bit is set in SYM_CYL_LST_XX)
                            i = bit number which is set in SYM_CYL_LST_XX
                            ERR_DTC[IDX] = ID_ERR_DTC_MIS [ 1 ] [ i ]
              Endif
       Endif
Endif
```

For others diagnosis :

(also for misfire diagnosis if treated by cylinder (LV_ERR_MIS_x is defined))

If NC_ERR_DTC_CONF = 4 (last symptom J2012 DTC is stored)

Then **If** only one bit is set in ERR_SYM_LST[IDX]

Then i = bit number which is set in ERR_SYM_LST[IDX]

ERR_DTC[IDX] = ID_ERR_DTC_XX [0] [i] with XX = F(IDX)

Else (many bit is set in ERR_SYM_LST[IDX])

ERR_DTC[IDX] = ID_ERR_DTC_XX [0] [4] with XX = F(IDX)

Endif

Endif

If NC_ERR_DTC_CONF = 5 (customer encoded DTC is stored)

Then **If** only one bit is set in ERR_SYM_LST[IDX]

Then i = bit number which is set in ERR_SYM_LST[IDX]

ERR_DTC[IDX] =

ID_ERR_DTC_XX [NC_NR_DTC_FMT] [i] with XX = F(IDX)

Else (many bit is set in ERR_SYM_LST[IDX])

ERR_DTC[IDX] =

ID_ERR_DTC_XX [NC_NR_DTC_FMT] [5 - NC_NR_DTC_FMT]

with XX = F(IDX)

Endif

Endif

B.69.2 DTC storage for the Misfiring (with NLC_TREAT_DIAG_MIS = 0, Misfire by type)

Description:

As describe above, the cylinder in default is memorised at the first occurrence. In this part each new cylinder in default will be memorised.

Application conditions:

Initialization:	treated above
Recurrence:	For Misfire A, B1 and B4:
	Misfire segment synchronous
Activation:	-

Formula section:

For a misfiring A diagnosis :

SYM_CYL_MEM_MIS_A is updated with the new additional bits (cylinder, multiple and random) of SYM_CYL_MIS_A

For a misfiring B1 diagnosis :

SYM_CYL_MEM_MIS_B1 is updated with the new additional bits (cylinder, multiple and random) of SYM_CYL_MIS_B1

For a misfiring B4 diagnosis :

SYM_CYL_MEM_MIS_B4 is updated with the new additional bits (cylinder, multiple and random) of SYM_CYL_MIS_B4.

B.69.3 DTC deleting

Description:

When the failure IDX is erased from 2nd layer memory, the memorised detected symptom is also erased by calling the action ACTION_ERRM_EraseDtc (IN <IDX>).

Syntax :	ACTION_ERRM_EraseDtc (IN <IDX>)	
Parameter (in) :	IDX	Index of failure in 2nd layer memory to store the DTC
Parameter (out) :	-	
Short description :	This action erases the memorised symptom of the failure IDX	

Application conditions:

Initialization: -

Recurrence: -

Activation: at Action request

Formula section:

If NLC_TREAT_DIAG_MIS = 0

(Misfire are treated by type (LV_ERR_MIS_A, MIS_B1 and MIS_B4 is defined))

Then For a misfiring A diagnosis :

SYM_CYL_MEM_MIS_A = 0

SYM_CYL_LST_MIS_A = 0

ERR_DTC[IDX] = 0

For a misfiring B1 diagnosis :

SYM_CYL_MEM_MIS_B1 = 0

SYM_CYL_LST_MIS_B1 = 0

ERR_DTC[IDX] = 0

For a misfiring B4 diagnosis :

SYM_CYL_MEM_MIS_B4 = 0

SYM_CYL_LST_MIS_B4 = 0

ERR_DTC[IDX] = 0

Endif

For other diagnoses, including misfire diagnoses if treated by cylinder (LV_ERR_MIS_x is defined) :

ERR_SYM_MEM[IDX] = 0

ERR_SYM_LST[IDX] = 0

ERR_DTC[IDX] = 0

B.69.4 Calibratable DTC number

Description:

The DTC numbers are defined by a calibration table called ID_ERR_DTC_XX.

This table defined for each failure XX :

a) with NC_NR_DTC_FMT = 0

- a DTC number for each symptom of the failure XX
- a global OBD DTC number for the failure XX
- a global customer specific DTC number

b) with NC_NR_DTC_FMT = 1

- a OBD DTC number for each symptom of the failure XX
- a global OBD DTC number for the failure XX
- a customer specific DTC number for each symptom of the failure XX
- a global customer specific DTC number

When a DTC is request by a diagnostic tool, the Communication Interface module build up the DTC number with information :

- ERR_SYM_MEM[IDX] and / or ERR_SYM_LST[IDX] stored in 2^{nd} layer memory ;
- Contents of table ID_ERR_DTC_XX .

There is a particular case for misfiring (MIS_A, MIS_B1, MIS_B4) diagnosis. The DTC numbers are defined by cylinder in a calibration table called ID_ERR_DTC_MIS.

(used if NLC_TREAT_DIAG_MIS = 0; Misfire by type)

- ID_ERR_DTC_MIS[0] [i] is the DTC for misfiring detection on the cylinder number i +1
- ID_ERR_DTC_MIS[0] [NC_CYL_NR] is the DTC for multiple misfiring
- ID_ERR_DTC_MIS[0] [NC_CYL_NR +1] is the DTC for random misfiring

- ID_ERR_DTC_MIS[1] [i] is the customer DTC for misfiring detection on the cylinder number i+1
- ID_ERR_DTC_MIS[1] [NC_CYL_NR] is the customer DTC for multiple misfiring
- ID_ERR_DTC_MIS[1] [NC_CYL_NR +1] is the customer DTC for random misfiring

Example 1:

The failure TCO (coolant temperature) is stored in the 2nd layer memory. The calibratable table of DTC has the following value :

a) with NC_NR_DTC_FMT = 0

DTC of the 1st symptom	DTC of the 2nd symptom	DTC of the 3rd symptom	DTC of the 4th symptom	Global DTC of the failure	Specific customer code
ID_ERR_DTC_ TCO [0] [0] = 0118h (P0118)	ID_ERR_DTC_ TCO [0] [1] = 0117h (P0117)	ID_ERR_DTC_ TCO [0] [2] = 0119h (P0119)	ID_ERR_DTC_ TCO [0] [3] = 0116h (P0116)	ID_ERR_DTC_ TCO [0] [4] = 0115h (P0115)	ID_ERR_DTC_ TCO [0] [5] = 30

b) with NC_NR_DTC_FMT = 1

	DTC of the 1st symptom	DTC of the 2nd symptom	DTC of the 3rd symptom	DTC of the 4th symptom	Global DTC of the failure
OBD DTC	ID_ERR_DTC_ TCO [0] [0] = 0118h (P0118)	ID_ERR_DTC_ TCO [0] [1] = 0117h (P0117)	ID_ERR_DTC_ TCO [0] [2] = 0119h (P0119)	ID_ERR_DTC_ TCO [0] [3] = 0116h (P0116)	ID_ERR_DTC_ TCO [0] [4] = 0115h (P0115)
Customer DTC	ID_ERR_DTC_ TCO [1] [0] = 31	ID_ERR_DTC_ TCO [1] [1] = 32	ID_ERR_DTC_ TCO [1] [2] = 33	ID_ERR_DTC_ TCO [1] [3] = 34	ID_ERR_DTC_ TCO [1] [4] = 30

Upon request from the diagnosis tool, the DTC number sent to the tool will be one of the numbers defined in ID_ERR_DTC_TCO regards the symptom detected.

Example 2 :

One failure misfiring (LV_ERR_MIS_A, LV_ERR_MIS_B1 or LV_ERR_MIS_B4) is stored in the 2nd layer memory. The application is a NC_CYL_NR -cylinder engine. The calibratable table of DTC has the following value :

	Misfiring on 1st cylinder	Misfiring on 2nd cylinder	Misfiring on 3rd cylinder	Misfiring on 4th cylinder	...	Misfiring on NC_CYL_NR th cylinder	Multiple Misfiring	Random Misfiring
OBD DTC	ID_ERR_ DTC_MIS [0] [0] = 0301h (P0301)	ID_ERR_ DTC_MIS [0] [1] = 0302h (P0302)	ID_ERR_ DTC_MIS [0] [2] = 0303h (P0303)	ID_ERR_ DTC_MIS [0] [3] = 0304h (P0304)	...	ID_ERR_ DTC_MIS [0] [NC_CYL_NR - 1] = 03xxh (P03xx)	ID_ERR_ DTC_MIS [0] [NC_CYL_NR] = 0300h (P0300)	ID_ERR_ DTC_MIS [0] [NC_CYL_NR + 1] = 0300h (P0300)
Customer DTC	ID_ERR_ DTC_MIS [1] [0] = 11	ID_ERR_ DTC_MIS [1] [1] = 12	ID_ERR_ DTC_MIS [1] [2] = 13	ID_ERR_ DTC_MIS [1] [3] = 14	...	ID_ERR_ DTC_MIS [1] [NC_CYL_NR - 1] = 16	ID_ERR_ DTC_MIS [1] [NC_CYL_NR] = 10	ID_ERR_ DTC_MIS [1] [NC_CYL_NR + 1] = 10

Upon request from the diagnosis tool, the DTC number sent to the tool will be one of the numbers defined in ID_ERR_DTC_MIS regards the symptom detected.

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
ID_ERR_DTC_XX	NC_LDP_1_DTC_TABLE_SIZE * NC_LDP_2_DTC_TABLE_SIZE	0...FFFFH	0...65535	1	[-]
LDP_NR_1_ID_ERR_DTC_XX	NC_LDP_1_DTC_TABLE_SIZE	0...FFH	0...255	1	[-]
LDP_NR_2_ID_ERR_DTC_XX	NC_LDP_2_DTC_TABLE_SIZE	0...FFH	0...255	1	[-]
DTC table declaration for the diagnosis instance XX, J2012 shall be applied on OBD/CARB part					
ID_ERR_DTC_MIS	2* NC_LDP_2_DTC_MIS_TABLE_SIZE	0...FFFFH	0...65535	1	[-]
LDP_NR_1_ID_ERR_DTC_MIS	2	0...FFH	0...255	1	[-]
LDP_NR_2_ID_ERR_DTC_MIS	NC_LDP_2_DTC_MIS_TABLE_SIZE	0...FFH	0...255	1	[-]
DTC table declaration for misfiring diagnosis, J2012 shall be applied on OBD/CARB part					

With :

NC_LDP_1_DTC_TABLE_SIZE = NC_NR_DTC_FMT +1

NC_LDP_2_DTC_TABLE_SIZE = 6 - NC_NR_DTC_FMT

NC_LDP_2_DTC_MIS_TABLE_SIZE = NC_CYL_NR +2

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_ERR_DTC_CONF	1	0...5H	0...5	1	[-]
Type of DTCs displayed through ERR_DTC[IDX]					
NC_NR_DTC_FMT	1	0...1H	0...1	1	[-]
Selection of 6 or 10 DTCs configurations					
NC_LDP_1_DTC_TABLE_SIZE	1	0...FH	0...15	1	[-]
Table size definition for ID_ERR_DTC_XX; set to NC_NR_DTC_FMT+1					
NC_LDP_2_DTC_TABLE_SIZE	1	0...FH	0...15	1	[-]
Table size definition for ID_ERR_DTC_XX; set to 6-NC_NR_DTC_FMT					
NC_LDP_2_DTC_MIS_TABLE_SIZE	1	0...FH	0...15	1	[-]
Table size definition for ID_ERR_DTC_MIS; set to NC_CYL_NR+2					

Configuration data detailed description:

NC_NR_DTC_FMT = 0 6 DTCs configuration.
= 1 10 DTCs configuration.

B.70 Diagnosis Trouble Code Management (DTC) (Appl. Inc.)

Input data:

NC_CONF_DTC_COD	NC_CHG_DTC_FMT	NC_TREAT_MIS_DIAG	

FUNCTION DESCRIPTION:

General information:

Description:

This file defines which DTC number to store for visualisation. To follow specific customer requirement and this file is to be fullfilled by project team.

All data initalic are comments.

Caracteristics	Value
DTC number to store for visualisation	NC_CONF_DTC_COD = **0** *Example : set to 0 for storage of symptom DTC* *set to 1 for storage of global DTC* *set to 2 for storage of symptom customer DTC* *set to 3 for storage global customer DTC*
Number of calibratable DTCs	NC_CHG_DTC_FMT = **0** *Example : set to 0 for 6 DTCs* *set to 1 for 10 DTCs* *(6DTCs + symptom customer DTCs)*

Two ways of the treatment of misfire failures are possible within the error management:

- Misfire by type (LV_ERR_MIS_A, MIS_B1 and MIS_B4 is defined)
- Misfire by cylinder (LV_ERR_MIS_x is defined)

Characteristics	Value
Way of treatment of misfire failure	NC_TREAT_MIS_DIAG = **1** *Example : set to 0 to treat Misfire by type* (LV_ERR_MIS_A, LV_ERR_MIS_B1 and MIS_B4 is defined) *set to 1 to treat Misfire by cylinder* *(LV_ERR_MIS_X is defined)*

B.71 History memory

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
CTR_ERR_HIS_NR	V/O/S	0...FFH	0...255	1	[-]
Number of failures stored in the history memory					
ERR_HIS_DTC[NC_NR_ERR_HIS]	V/O/S	0...FFFFH	0...65535	1	[-]
DTC of failure in history memory					
ERR_HIS[NC_NR_ERR_HIS][NC_NR_HIS]	O/V/S	0...FFH	0...255	1	[-]
Array for data to be stored in history memory					

Output data detailed description:

ERR_HIS_DTC[NC_NR_ERR_HIS] array counts NC_NR_ERR_HIS elements (1… NC_NR_ERR_HIS).

ERR_HIS[NC_NR_ERR_HIS][NC_NR_HIS] array counts NC_NR_ERR_HIS x NC_NR_HIS elements (1…NC_NR_ERR_HIS, 1… NC_NR_HIS).

Input data:

ERR_HIS_CONF[NC_NR_HIS]	ERR_DTC[NC_NR_ERR_DYN]		

Export actions:

ACTION_ERRM_StoreHistory (IN <IDX>)
This action stores the failure IDX in the history memory

FUNCTION DESCRIPTION:

General information:

This module is called by the dynamic error management module just before erasing a failure in the 2nd layer memory (dynamic memory).

The principle is to copy some data related to this failure in a separate memory, called history memory. So the failure become historic failure.

The number of historic failure is limited to a maximum of NC_NR_ERR_HIS. When this maximum number is reach, new historic failure entry deletes the oldest historic failure in memory (“first in first out” principle).

History memory is defined by :

- ERR_HIS_DTC[x] with : x = index of failure in history memory

  A DTC number related to the failure.

- ERR_HIS[x][y] with : x = index of failure in history memory

  y = index of data stored for failure x in history memory

  Some others data.

Contents of history memory is the same for each failure and defined in History memory (Appl. Inc.) module.

The history memory can be read and erased separately by dedicated diagnosis tool service.

The history memory can be deleted by setting the calibration LC_ERR_FMY_HIS_CLR to 1.

B.71.1 Store a failure in history memory

Description:

Syntax :	ACTION_ERRM_StoreHistory (IN <IDX>)
Parameter (in) :	IDX Index of failure to store in history memory
Parameter(out) :	-
Short description :	This action stores a failure in history memory

Application conditions:

Initialization:	after system initialization the history memory is restored from non-volatile memory or in case of a damaged non volatile memory the history memory is initialized with zero
Recurrence:	-
Activation:	at action request

Formula section:

If CTR_ERR_HIS_NR < NC_NR_ERR_HIS

Then CTR_ERR_HIS_NR = CTR_ERR_HIS_NR + 1

Else ERR_HIS_DTC[x] = ERR_HIS_DTC[x+1] (“first in first out” principle)

ERR_HIS[x][y] = ERR_HIS[x+1][y] (with x = 1...CTR_ERR_HIS_NR–1, y = 1..NC_NR_HIS)

Endif

ERR_HIS_DTC[x] = ERR_DTC[IDX]

ERR_HIS[x][y] = ERR_HIS_CONF[y]

(with x = 1.. NC_NR_HIS, y = CTR_ERR_HIS_NR)

B.71.2 Clear the History Memory

Description:

The history memory can be erased by setting the calibration bit LC_ERR_FMY_HIS_CLR at 1.

Application conditions:

Initialization: after system initialization the history memory is restored from non-volatile memory or in case of a damaged non volatile memory the history memory is initialized with zero

Recurrence: -

Activation: at LC_ERR_FMY_HIS_CLR 0→1 transition

Formula section:

ERR_HIS_DTC[x] = 0

ERR_HIS[x][y] = 0

(with x = 1 ... NC_NR_ERR_HIS

y = 1 ... NC_NR_HIS)

CTR_ERR_HIS_NR = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ERR_FMY_HIS_CLR	1	0...1H	0...1	1	[-]
Calibration to clear the history memory (when set to 1)					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_ERR_HIS	1	0...FFH	0...255	1	[-]
Max number of failure in the historic memory					
NC_NR_HIS	1	0...FFH	0...255	1	[-]
Max number of data (in bytes) to stored for a failure in the historic memory					

B.72 History memory (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_HIS_CONF[NC_NR_HIS]	O	0 ... FFH	0 ... 255	-	-
Contents of history memory					

Input data:

NC_NR_ERR_HIS	NC_NR_HIS	CTR_FRC[NC_NR_ERR_DYN]	TRT
ENVD_CUS_SET_CMN[NC_NR_ENVD_CUS_SET_CMN][NC_NR_FRF_SET][NC_NR_ERR_DYN]	CTR_WUP_CYC[NC_NR_ERR_DYN]	DIST	CTR_DC[NC_NR_ERR_DYN]
ENVD_CUS_SET_SPC[NC_NR_ENVD_CUS_SET_SPC][NC_NR_FRF_SET][NC_NR_ERR_DYN]	CTR_ABC_XX		

FUNCTION DESCRIPTION:

General information:

Description:

This file defines size and contents of History memory. To follow specific customer requirement, this file is a template to be fullfilled by project team.

All data in italic are comments.

History memory	Caracteristics	Value
	Max number of failure	NC_NR_ERR_HIS = 10
	Number of data to store for a failure (in bytes)	NC_NR_HIS = 22
	Contents	ERR_HIS_CONF_1 = TRT (byte 1) *Total running time of EMS*
		ERR_HIS_CONF_2 = TRT (byte 2) *Total running time of EMS*
		ERR_HIS_CONF_3 = TRT (byte 3) *Total running time of EMS*
		ERR_HIS_CONF_4 = TRT (byte 4) *Total running time of EMS*
		ERR_HIS_CONF_5 = DIST (byte 1) *DIST (1) for mileage counter*
		ERR_HIS_CONF_6 = DIST (byte 2) *DIST (2) for mileage counter*
		ERR_HIS_CONF_7 = DIST (byte 3) *DIST (3) for mileage counter*
		ERR_HIS_CONF_8 = DIST (byte 4) *DIST (4) for mileage counter*
		ERR_HIS_CONF_9 = StatusOfDTC *Current DTC status of failure (see spec. 2KI05Q01)*
		ERR_HIS_CONF_10 = CTR_ABC_XX *Counter of anti bounce counter of present failure*

		ERR_HIS_CONF_11 = CTR_FRC[NC_NR_ERR_DYN] *Frequency counter of present failure*
		ERR_HIS_CONF_12 = CTR_WUP_CYC[NC_NR_ERR_DYN] *Counter of warm-up cycles of present failure*
		ERR_HIS_CONF_13 = CTR_DC[NC_NR_ERR_DYN] *Counter of driving cycles of present failure*
		ERR_HIS_CONF_14 = CARB Failure Status *Current CARB status of present failure (see spec. 2KI05Q01)*
		ERR_HIS_CONF_15 = ENVD_CUS_SET_SPC_1_1 *Failure specific freeze frame data 1*
		ERR_HIS_CONF_16 = ENVD_CUS_SET_SPC_1_2 *Failure specific freeze frame data 2*
		ERR_HIS_CONF_17 = ENVD_CUS_SET_SPC_1_3 *Failure specific freeze frame data 3*
		ERR_HIS_CONF_18 = ENVD_CUS_SET_SPC_1_4 *Failure specific freeze frame data 4*
		ERR_HIS_CONF_19 = ENVD_CUS_SET_CMN_1_9 *Freeze frame data engine speed*
		ERR_HIS_CONF_20 = ENVD_CUS_SET_CMN_1_10 *Freeze frame data mass-airflow (high byte)*
		ERR_HIS_CONF_21 = ENVD_CUS_SET_CMN_1_11 *Freeze frame data coolant temperature*
		ERR_HIS_CONF_22 = ENVD_CUS_SET_CMN_1_12 *Freeze frame data battery voltage*

B.73 TIA Plausibility diagnosis (Turbo)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_PLAUS_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	[-]
Condition for TIA plausibility diagnosis: 0 = not fullfilled, 1 = fullfilled					
ERR_SYM_PLAUS_TIA[NC_SENS_NR_TIA]	V/O	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Symptom status for TIA plausibility diagnosis					
LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA]	V/O	0...1H	0...1	1	[-]
Present failure for TIA plausibility diagnosis					
LV_END_DIAG_PLAUS_TIA[NC_SENS_NR_TIA]	V/O	0...1H	0...1	1	[-]
Result of TIA plausibility diagnosis					
LV_CDN_DIAG_PLAUS_TIA_PREL[NC_SENS_NR_TIA]	V	0...1H	0...1	1	[-]
Preliminary condition for TIA plausibility diagnosis: 0 = not fullfilled, 1 = fullfilled					
TIA_MES_DIAG[NC_SENS_NR_TIA]	V	0...FEH	-48...142.5	0.75	[°C]
Reference measured TIA value for diagnosis (different from TIA_MES only for test purpose)					
STATE_PLAUS_TIA	V	0H 1H	INC_CHK DEC_CHK	1	[-]
TIA check type (increase or decrease) currently performed					
MAF_INT_PLAUS_TIA	V	0...FFFFH	0...116.49	1.7775e-3	[kg]
Mass air flow integral for TIA plausibility diagnosis (warm up criteria)					
T_PLAUS_TIA	V	0...FFFFH	0...655.35	**0.01**	[s]
Timer for TIA plausibility diagnosis					
T_PLAUS_TIA_DIF	V	0...FFFFH	0...655.35	**0.01**	[s]
Timer difference value used for timer definition for TIA plausibility diagnosis					
T_PLAUS_TIA_INC_MAX	V	0...FFFFH	0...655.35	**0.01**	[s]
Maximum timer value in case TIA increase check					
T_PLAUS_TIA_DEC_MAX	V	0...FFFFH	0...655.35	**0.01**	[s]
Maximum timer value in case TIA decrease check					
TIA_DIF_PLAUS_TIA[NC_SENS_NR_TIA]	V	80...7FH	-96...95.25	0.75	[°C]
TIA variation (decrease / increase) value for TIA plausibility diagnosis					
LV_TIA_PREP	V	0...1H	0...1	1	[-]
LV_TIA_PREP = 1: "preparation (preconditionning) period" is over and symptom detection is from now possible					
TIA_DIF_PLAUS_TIA_THD_MIN	V	80...7FH	-96...95.25	0.75	[°C]
Minimum TIA variation threshold value for TIA plausibility symptom determination					
TIA_DIF_PLAUS_TIA_THD_MAX	V	80...7FH	-96...95.25	0.75	[°C]
Maximum TIA variation threshold value for TIA plausibility symptom determination					
TIA_DIF_PLAUS_TIA_DEC_CAL_MIN[NC_SENS_NR_TIA]	V	80...7FH	-96...95.25	0.75	[°C]
Minimum TIA decrease value recorded within the driving cycle ¿ used only for calibration ease					
TIA_DIF_PLAUS_TIA_DEC_CAL_MAX[NC_SENS_NR_TIA]	V	80...7FH	-96...95.25	0.75	[°C]
Maximum TIA decrease value recorded within the driving cycle ¿ used only for calibration ease					
TIA_DIF_PLAUS_TIA_INC_CAL_MIN[NC_SENS_NR_TIA]	V	80...7FH	-96...95.25	0.75	[°C]
Minimum TIA increase value recorded within the driving cycle ¿ used only for calibration ease					
TIA_DIF_PLAUS_TIA_INC_CAL_MAX[NC_SENS_NR_TIA]	V	80...7FH	-96...95.25	0.75	[°C]
Maximum TIA increase value recorded within the driving cycle ¿ used only for calibration ease					
T_PLAUS_TIA_REF	-	0...FFFFH	0...655.35	**0.01**	[s]
Reference timer value for TIA plausibility diagnosis					
TIA_REF_PLAUS_TIA[NC_SENS_NR_TIA]	-	0...FEH	-48...142.5	0.75	[°C]

Reference TIA value for TIA_DIF_PLAUS_TIA determination

Input data:

TIA_MES[NC_SENS_NR_TIA]	STATE_ENG	LV_IGK
LV_INH_DIAG_PLAUS_TIA[NC_SENS_NR_TIA]	NC_SENS_NR_TIA	VS
ERR_SYM_INTM_TIA[NC_SENS_NR_TIA]	LV_CDN_VB_MIN_DIAG	MAF_KGH
ERR_SYM_DYN_TIA[NC_SENS_NR_TIA]	ERR_SYM_EL_TIA[NC_SENS_NR_TIA]	PQ_CHA[NC_NR_TCHA]

FUNCTION DESCRIPTION:

General information:

Version valid for:

NC_CHRG_CONF <> 0 And NC_EGR_CONF = 0 And NC_TIA_CONF = 10,13

Important:

. Project with TIA sensor only (NC_TAM_CAN_USE = 0):
The aim of this functionnality is to verify the right dynamic [not low/not too high] of a TIA sensor under dedicated driving conditions. This check is the only feasable when an additionnal TAM sensor is not available.

. Project with TIA sensor + TAM sensor via CAN (NC_TAM_CAN_USE = 1):
An additionnal functionnality must be implemented in the SW to detect an offset (temperature shift) on TIA value. This check is verified within "TAM plausibility diagnosis" (Refer to module xxB05xxx).

NC_SENS_NR_TIA has been defined depending on the TIA hardware definition (NC_TIA_CONF value).

The diagnosis detects failures on air intake temperature signal from a dedicated sensor. This diagnosis is necessary as a number of OBD II functionnalities and/or diagnosises are based on air intake temperature value (e.g.: generic TCO plausibility diagnosis)

The intake air temperature is expected to decrease significantly when the vehicule is run at high vehicule speed and intermediate load and limited charge pressure ratio during a sufficient period of time. If this decrease of temperature is too small/big (or if the temperature increases instead) then the symptom “PLAUS_DEC” (“DEC” for “decrease”) is set.
Respectively, the intake air temperature is expected to increase significantly when:
. the vehicule is run at low vehicule speed and low load or
. the vehicule is run in charge mode at high load

and this during a sufficient period of time. If this increase of temperature is too small/big (or if the temperature decreases instead) then the symptom “PLAUS_INC” (“INC” for “increase”) is set.

In engine warm conditions, the two tests are performed successively (test "INC_CHK" --> test "DEC_CHK" --> test "INC_CHK" --> ...etc ...) and the failure counter is incremented each time the symptom “PLAUS_DEC” or “PLAUS_INC” is set.

During the first part of the engine warm up phase, the “temperature increase ” and “temperature decrease” checks are delayed until the dynamic (increase and decrease) of TIA sensor is sufficient (detectable).

Remark: TIA is the main parameter for TCO and TAM plausibility diagnosis. Consequently:
_ TIA plausibility diagnosis has been designed independantely of TCO and TAM.
_ The engine warm up is evaluated by computing an integral of mass air flow (instead of TCO)

Signal flow diagram:

Description:

Application conditions:

Initialisation:

At Reset **Or** Clear of FMY

STATE_PLAUS_TIA = "DEC_CHK"

For m = 1 to NC_SENS_NR_TIA

TIA_REF_PLAUS_TIA[m] = TIA_MES[m]
LV_CDN_DIAG_ PLAUS_TIA[m] = *Refer to filtering configuration for the initialisation value*
LV_END_DIAG_ PLAUS_TIA[m] = *Refer to filtering configuration for the initialisation value*
LV_ERR_PLAUS_TIA[m] = *Refer to filtering configuration for the initialisation value*
ERR_SYM_ PLAUS_TIA[m] = *Refer to filtering configuration for the initialisation value*

TIA_MES_DIAG[m] = TIA_MES[m]

TIA_DIF_PLAUS_TIA_DEC_CAL_MIN[m] = TIA_DIF_PLAUS_TIA_INC_CAL_MIN[m] = +95.25°C
TIA_DIF_PLAUS_TIA_DEC_CAL_MAX[m] = TIA_DIF_PLAUS_TIA_INC_CAL_MAX[m] = -96°C

All other variales from Output data list = 0H

End for m

Recurrence: 100ms

Activation: always active

Deactivation: -

Formula section:

integral maf:

If STATE_ENG <> PUC
Then MAF_INT_PLAUS_TIA
= MAF_INT_PLAUS_TIA(n-1) + (MAF_KGH / 3600) * 0.1 *(100ms)*
remark: MAF_INT_PLAUS_TIA must be internally calculated on 32 bits
Else MAF_INT_PLAUS_TIA(n) = MAF_INT_PLAUS_TIA(n-1)
Endif

Timer and state flag:

T_PLAUS_TIA_INC_MAX = IP_T_PLAUS_TIA_INC_MAX(MAF_INT_PLAUS_TIA)

```
If LV_TIA_PREP(n-1) = 0
Then T_PLAUS_TIA_DEC_MAX = C_T_PLAUS_TIA_DEC_MAX_INI
Else T_PLAUS_TIA_DEC_MAX = IP_T_PLAUS_TIA_DEC_MAX(MAF_INT_PLAUS_TIA)
Endif

If MAF_INT_PLAUS_TIA >= C_MAF_INT_PLAUS_TIA_MIN
Then If (STATE_PLAUS_TIA(n-1) = "DEC_CHK"
            And C_MAF_PLAUS_TIA_DEC_L <= MAF_KGH <= C_MAF_PLAUS_TIA_DEC_H
            And PQ_CHA[NC_NR_TCHA] <= C_PQ_CHA_PLAUS_TIA_DEC
            And VS >= C_VS_PLAUS_TIA_DEC)
        Or {STATE_PLAUS_TIA(n-1) = "INC_CHK"
            And [(MAF_KGH >= C_MAF_PLAUS_TIA_INC_H
                And PQ_CHA[NC_NR_TCHA] >= C_PQ_CHA_PLAUS_TIA_INC)
              Or (MAF_KGH <= C_MAF_PLAUS_TIA_INC_L
                And VS <= C_VS_PLAUS_TIA_INC)]}
        Then T_PLAUS_TIA_REF = T_PLAUS_TIA(n-1) + 0.1 (100ms)
             T_PLAUS_TIA = T_PLAUS_TIA_REF
        Else T_PLAUS_TIA_REF = T_PLAUS_TIA_REF(n-1)
             T_PLAUS_TIA = T_PLAUS_TIA(n-1) –
                          ID_T_PLAUS_TIA_DEC(T_PLAUS_TIA_DIF(n-1))
        Endif
        T_PLAUS_TIA_DIF = T_PLAUS_TIA_REF – T_PLAUS_TIA
Endif

If T_PLAUS_TIA = 0
Then T_PLAUS_TIA_REF = 0 (reset)
Endif


For m = 1 to NC_SENS_NR_TIA

If LC_TIA_MES_DIAG_CAL = 0
Then TIA_MES_DIAG[m] = TIA_MES[m]
Else TIA_MES_DIAG[m] = C_TIA_MES_DIAG_CAL
Endif
```

Temperature variations and Thresholds:

```
If LV_TIA_PREP(n-1) = 1
Then If STATE_PLAUS_TIA(n-1) = "DEC_CHK"
        Then TIA_REF_PLAUS_TIA[m]
               = MAX(TIA_MES_DIAG[m], TIA_REF_PLAUS_TIA[m](n-1))
        Else TIA_REF_PLAUS_TIA[m]
               = MIN(TIA_MES_DIAG[m], TIA_REF_PLAUS_TIA[m](n-1))
        Endif
Else TIA_REF_PLAUS_TIA[m] = TIA_MES_DIAG[m]
Endif

If STATE_PLAUS_TIA(n-1) = "DEC_CHK"
Then TIA_DIF_PLAUS_TIA[m] = MIN(TIA_MES_DIAG[m] - TIA_REF_PLAUS_TIA[m],
        TIA_DIF_PLAUS_TIA[m](n-1))
        (TIA_DIF_PLAUS_TIA <=0 always)
        TIA_DIF_PLAUS_TIA_THD_MIN
         = IP_TIA_DIF_PLAUS_DEC_THD_MIN(MAF_INT_PLAUS_TIA)
        TIA_DIF_PLAUS_TIA_THD_MAX
```

```
            = C_TIA_DIF_PLAUS_TIA_DEC_THD_MAX
Else TIA_DIF_PLAUS_TIA[m] = MAX(TIA_MES_DIAG[m] - TIA_REF_PLAUS_TIA[m],
      TIA_DIF_PLAUS_TIA[m](n-1))
      (TIA_DIF_PLAUS_TIA >=0 always)
      TIA_DIF_PLAUS_TIA_THD_MIN
      = IP_TIA_DIF_PLAUS_INC_THD_MIN(MAF_INT_PLAUS_TIA)
      TIA_DIF_PLAUS_TIA_THD_MAX
      = C_TIA_DIF_PLAUS_TIA_INC_THD_MAX
Endif
```

Condition flag and symptom:

```
If LV_IGK = 1
And LV_CDN_VB_MIN_DIAG = 1
And ERR_SYM_EL_TIA[m] = 0
And ERR_SYM_INTM_TIA[m] = 0
And ERR_SYM_DYN_TIA[m] = 0
And LV_INH_DIAG_PLAUS_TIA[m] = 0
And (LV_END_DIAG_PLAUS_TIA[m] = 0 Or LC_END_DIAG_PLAUS_TIA_SUPP = 1)
Then LV_CDN_DIAG_PLAUS_TIA_PREL[m] = 1
Else  LV_CDN_DIAG_PLAUS_TIA_PREL[m] = 0
Endif

If LV_TIA_PREP(n-1) = 1
And LV_CDN_DIAG_PLAUS_TIA_PREL[m] = 1
And {[ STATE_PLAUS_TIA(n-1) = "INC_CHK"
        And T_PLAUS_TIA >= T_PLAUS_TIA_INC_MAX
        And (MAF_INT_PLAUS_TIA > C_MAF_INT_PLAUS_TIA_INC_MIN)]
      Or
      [STATE_PLAUS_TIA(n-1) = "DEC_CHK"
        And T_PLAUS_TIA >= T_PLAUS_TIA_DEC_MAX
        And (MAF_INT_PLAUS_TIA > C_MAF_INT_PLAUS_TIA_DEC_MIN)]}
Then LV_CDN_DIAG_PLAUS_TIA[m] = 1
      If STATE_PLAUS_TIA(n-1) = "INC_CHK"
      And (TIA_DIF_PLAUS_TIA[m] < TIA_DIF_PLAUS_TIA_THD_MIN
           Or TIA_DIF_PLAUS_TIA[m] > TIA_DIF_PLAUS_TIA_THD_MAX)
      Then  "test: temperature increase "
             ERR_SYM_PLAUS_TIA[m] = "PLAUS_INC"
      Else  If STATE_PLAUS_TIA(n-1) = "DEC_CHK"
            And (TIA_DIF_PLAUS_TIA[m] > TIA_DIF_PLAUS_TIA_THD_MIN
                 Or TIA_DIF_PLAUS_TIA[m] < TIA_DIF_PLAUS_TIA_THD_MAX)
            Then "test: temperature decrease "
                   ERR_SYM_PLAUS_TIA[m] = "PLAUS_DEC"
            Else   ERR_SYM_PLAUS_TIA[m] = "NO_SYM"
            Endif
      Endif
Else LV_CDN_DIAG_PLAUS_TIA[m] = 0
Endif
```

Additionnal variables displayed on application tool to ease calibration:

```
If LV_CDN_DIAG_PLAUS_TIA[m] = 1
Then If STATE_PLAUS_TIA(n-1) = "DEC_CHK"
       Then
```

TIA_DIF_PLAUS_TIA_DEC_CAL_MIN[m](n)
= MIN[TIA_DIF_PLAUS_TIA[m], TIA_DIF_PLAUS_TIA_DEC_CAL_MIN[m](n-1)]
TIA_DIF_PLAUS_TIA_DEC_CAL_MAX[m](n)
= MAX[TIA_DIF_PLAUS_TIA[m], TIA_DIF_PLAUS_TIA_DEC_CAL_MAX[m](n-1)]
Else
TIA_DIF_PLAUS_TIA_INC_CAL_MIN[m](n)
= MIN[TIA_DIF_PLAUS_TIA[m], TIA_DIF_PLAUS_TIA_INC_CAL_MIN[m](n-1)]
TIA_DIF_PLAUS_TIA_INC_CAL_MAX[m](n)
= MAX[TIA_DIF_PLAUS_TIA[m], TIA_DIF_PLAUS_TIA_INC_CAL_MAX[m](n-1)]
Endif
Endif

End for m

<u>*Other:*</u>

If STATE_PLAUS_TIA(n-1) = "DEC_CHK"
And T_PLAUS_TIA >= T_PLAUS_TIA_DEC_MAX
Then STATE_PLAUS_TIA = "INC_CHK"
Else If STATE_PLAUS_TIA(n-1) = "INC_CHK"
And T_PLAUS_TIA >= T_PLAUS_TIA_INC_MAX
Then STATE_PLAUS_TIA = "DEC_CHK"
Endif *(STATE_PLAUS_TIA unchanged)*
Endif *(STATE_PLAUS_TIA unchanged)*

If STATE_PLAUS_TIA = "INC_CHK" *(ie: 1st transition "DEC_CHK" to "INC_CHK" as init = "DEC_CHK")*
Then LV_TIA_PREP = 1 *(used to define from when sym detection is possible)*
Endif *(LV_TIA_PREP unchanged)*

If STATE_PLAUS_TIA <> STATE_PLAUS_TIA(n-1)
Then T_PLAUS_TIA = 0 *(timer reset)*
T_PLAUS_TIA_REF = 0 *(timer reset)*
Endif *(T_PLAUS_TIA and T_PLAUS_TIA_REF unchanged)*

Symptoms are **unchanged** if condition for doing symptom detection is not fulfilled.

<u>Filtering :</u>

Apply filter on current symptoms
If filtering result available (after debounce)
Then LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA] = filtering result
LV_END_DIAG_PLAUS_TIA[NC_SENS_NR_TIA] = 1
Delivery of the result to Error Management
Endif

<u>Remark:</u>
After debouncing,
LV_END_DIAG_PLAUS_TIA[NC_SENS_NR_TIA],
LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA]
are automatically managed.
Note: The type and the calibration of the anti-bounce are defined in Application Incidence specification.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TIA_MES_DIAG_CAL	1	0...FEH	-48...142.5	0.75	[°C]
Application value for TIA_MES value: to be used for TIA diagnosis calibration purpose only					
C_MAF_INT_PLAUS_TIA_DEC_MIN	1	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum air mass flow integral value to allow the temperature decrease test					
C_MAF_INT_PLAUS_TIA_INC_MIN	1	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum air mass flow integral value to allow the temperature increase test					
C_MAF_INT_PLAUS_TIA_MIN	1	0...FFFFH	0...116.49	1.7775e-3	[kg]
Commun minimum necessary air mass flow integral value for TIA decrease/increase test					
C_MAF_PLAUS_TIA_DEC_L	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Minimum mass air flow value for TIA decrease					
C_MAF_PLAUS_TIA_DEC_H	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Maximum mass air flow value for TIA decrease					
C_MAF_PLAUS_TIA_INC_L	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Minimum mass air flow value for TIA increase					
C_MAF_PLAUS_TIA_INC_H	1	0...FFFFH	0...2047.96875	0.03125	[kg/h]
Maximum mass air flow value for TIA increase					
C_PQ_CHA_PLAUS_TIA_DEC	1	0...FFFFH	0...15.99975	0.2441e-3	[-]
Maximum charge pressure quotient value for TIA decrease					
C_PQ_CHA_PLAUS_TIA_INC	1	0...FFFFH	0...15.99975	0.2441e-3	[-]
Minimum charge pressure quotient value for TIA increase					
C_VS_PLAUS_TIA_DEC	1	0...FFH	0...255	1	[km/h]
Minimum vehicle speed value for TIA decrease					
C_VS_PLAUS_TIA_INC	1	0...FFH	0...255	1	[km/h]
Maximum vehicle speed value for TIA increase					
C_T_PLAUS_TIA_DEC_MAX_INI	1	0...FFFFH	0...655.35	0.01	[s]
Iniitial maximum timer value in case TIA decrease check during reconditionning period					
C_TIA_DIF_PLAUS_TIA_INC_THD_MAX	1	0...FFH	-96...95.25	0.75	[°C]
Maximum TIA variation threshold for symptom ¿PLAUS_INC¿ detection					
C_TIA_DIF_PLAUS_TIA_DEC_THD_MAX	1	0...FFH	-96...95.25	0.75	[°C]
Maximum TIA variation threshold for symptom ¿PLAUS_DEC¿ detection					
ID_T_PLAUS_TIA_DEC	6	0...FFFFH	0...655.35	0.01	[s]
LDP_T_PLAUS_TIA_DIF_IP_T_PLAUS	6	0...FFFFH	0...655.35	0.01	[s]
Timer decrerment value for TIA plausibility diagnosis					
IP_T_PLAUS_TIA_INC_MAX	4	0...FFFFH	0...655.35	0.01	[s]
LDPM_MAF_INT_PLAUS_TIA_1_AIRT	4	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum necessarry time to allow TIA increase test					
IP_T_PLAUS_TIA_DEC_MAX	4	0...FFFFH	0...655.35	0.01	[s]
LDPM_MAF_INT_PLAUS_TIA_1_AIRT	4	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum necessary time to allow TIA decrease test					
IP_TIA_DIF_PLAUS_DEC_THD_MIN	4	0...FFH	-96...95.25	0.75	[°C]
LDPM_MAF_INT_PLAUS_TIA_1_AIRT	4	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum TIA variation threshold for symptom PLAUS_DEC detection					
IP_TIA_DIF_PLAUS_INC_THD_MIN	4	0...FFH	-96...95.25	0.75	[°C]
LDPM_MAF_INT_PLAUS_TIA_1_AIRT	4	0...FFFFH	0...116.49	1.7775e-3	[kg]
Minimum TIA variation threshold for symptom PLAUS_INC detection					
LC_END_DIAG_PLAUS_TIA_SUPP	1	0...1H	0...1	1	[-]
Switch to suppress the effect of LV_END_DIAG_PLAUS_TIA on TIA plausibility diagnosis activations conditions					
LC_TIA_MES_DIAG_CAL	1	0...1H	0...1	1	[-]
Switch to fix manually TIA_MES_DIAG: for calibration purpose only ¿ must be set to 0 for serial production !!					

B.74 TIA Plausibility Diagnostic (application incidence)

B.74.1 General:

The functionnality described in this module is necessary only if TIA plausibilty diagnosis functionnality is used by the Project.

This specification is a template. It has to be adapted to the specifics needs of each Project.

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_PLAUS_TIA[NC_SENS_NR_TIA]	LV_ERR_DYN_TIA[NC_SENS_NR_TIA]
NC_SENS_NR_TIA	LV_ERR_MAF	LV_ERR_VS	LV_ERR_EL_TIA[NC_SENS_NR_TIA]
LV_ERR_INTM_TIA[NC_SENS_NR_TIA]	LV_ERR_LOAD_PLAUS	LV_ERR_MAF_SCG_OC	LV_ERR_MAF_SCP
CTR_CDN_RBM_PLAUS_TIA_IM_CYL	CTR_COMP_RBM_PLAUS_TIA_IM_CYL	LV_ERR_MAP	LV_ERR_EL_MAP
LV_ERR_VCC_PVS_2	LV_ERR_MAP_PLAUS		

B.74.2 Project specific inhibition conditions:

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_PLAUS_TIA[NC_SENS_NR_TIA]	V/O	0..1H	0..1	1	-
Inhibition condition for TIA plausibility diagnosis					
LV_INH_DIAG_RBM_PLAUS_TIA[NC_SENS_NR_TIA]	V	0...1H	0...1	1	-
TIA plausibility diagnosis inhibition condition for RBM interface definition					

FUNCTION DESCRIPTION:

General information:

The flag LV_INH_DIAG_PLAUS_TIA allows to deactivate the corresponding diagnostic.

The Formula section must be updated by the Project. The Calibration data list must be completed.

The project can add additionnal conditions if desired.

FUNCTION DESCRIPTION:

Application conditions:

Initialization: **At** Reset: LV_INH_DIAG_PLAUS_TIA = 0

Reccurence: **100 ms**

Activation: always active

Deactivation: -

Formula section:

For m = 1 to NC_SENS_NR_TIA

If LV_ERR_MAF = 0 (el.)
And LV_ERR_MAP = 0 (el.)
And LV_ERR_LOAD_PLAUS = 0 (MAF or MAP plaus.)
And LV_ERR_VS = 0
And LV_ERR_EL_TIA[m] = 0
And LV_ERR_INTM_TIA[m]= 0
And LV_ERR_DYN_TIA[m]= 0
Then LV_INH_DIAG_RBM_PLAUS_TIA[m] = 0
Else LV_INH_DIAG_RBM_PLAUS_TIA[m] = 1
Endif

If LC_INH_DIAG_PLAUS_TIA = 0
And LV_INH_DIAG_RBM_PLAUS_TIA[m] = 0
Then LV_INH_DIAG_PLAUS_TIA[m] = 0
Else LV_INH_DIAG_PLAUS_TIA[m] = 1
Endif

End for m

Configuration for diagnostic symptoms :

Diagnostic	Symptom Description	Symptom	Filter type
PLAUS_TIA_IM_CYL	TIA increase error	SYM_0	*STD_INI*
	TIA decrease error	SYM_1	
		SYM_2	
		SYM_3	

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_ABC_INC_PLAUS_TIA	1	0..FFH	0..255	-	
Antibounce counter increment for TIA plausibility diagnosis					
C_ABC_MAX_PLAUS_TIA	1	1..FFH	1..255	-	
Maximum value for antibounce counter for TIA plausibility diagnosis					
LC_INH_DIAG_PLAUS_TIA	1	0..1H	0..1	-	
Inhibition switch of TIA plausibility diagnosis					

B.74.3 RBM interface:

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_PLAUS_TIA[NC_SENS_NR_TIA]	O/V	0 ... FFH	0 ... 255	1	-
Interface of TIA plausibility diagnosis monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_PLAUS_TIA_IM_CYL	V	0 ... FFFFH	0 ... 7,99572	0,000122	-
Actual RBM rate for TIA Plausibility Diagnosis , for application purpose					

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

NC_SENS_NR_TIA has been defined depending on the TIA hardware definition (NC_TIA_CONF value).

The RBM interface has to be defined for the NC_SENS_NR_TIA sensor(s).
(See “Loop For m = 1 to NC_SENS_NR_TIA” below).

With this module the interface between the TIA plausibility monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_PLAUS_TIA[NC_SENS_NR_TIA] data.

Within STATE_RBM_PLAUS_TIA[NC_SENS_NR_TIA], three different informations are defined:

_ Conditions for monitoring are met long enough to detect malfunction (bit 0)

(no intrusive operation, no short trip)

_ Monitor disabled because of system malfunction (bit 1)

(depending on failure status: pending)

_ Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

For m = 1 to NC_SENS_NR_TIA

at ECU reset :

bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_TIA[m] = 0

at LV_DC 0 → 1 transition :

bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_TIA[m] = 0

on failure memory reset :

bit 1 of STATE_RBM_PLAUS_TIA[m] = 0

End for m

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition and LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

LV_ERR_VS (Vehicle Speed Sensor “B” Range/Performance - OBD-market)
LV_ERR_LOAD_PLAUS (Mass or Volume Air Flow Circuit Range / Performance)
LV_ERR_MAF_SCG_OC (Mass or Volume Air Flow Circuit Low Input)
LV_ERR_MAF_SCP (Mass or Volume Air Flow Circuit high Input)
LV_ERR_EL_MAP (Manifold air pressure electrical failure)
LV_ERR_MAP_PLAUS
LV_ERR_VCC_PVS_2
LV_ERR_EL_TIA[NC_SENS_NR_TIA]
LV_ERR_INTM_TIA[NC_SENS_NR_TIA]
LV_ERR_DYN_TIA[NC_SENS_NR_TIA]

For m = 1 to NC_SENS_NR_TIA
  If(1) { CPU optimization at LV_DC 0 → 1 transition }
      CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn’t empty }
  Then(1) **While** bit 1 of STATE_RBM_PLAUS_TIA[m] = 0 **do**
      with each TIA plausibility failure of the above list :
      ACTION_ERRM_CheckPendingStatus(IN<XX>,
      OUT<PendingStatus>, SYNCHRONIZATION<CALL>)
      If(2) **XX** has a pending status
      Then(2) bit 1 of STATE_RBM_PLAUS_TIA[m] = 1
      Endif(2)
    Endwhile
  Else(1) { the dynamic failure memory is empty }
    No action
  Endif(1)
End for m

Every 1 s :

For m = 1 to NC_SENS_NR_TIA

  If bit 0 of STATE_RBM_PLAUS_TIA[m] = 0
  Then **If** LV_END_DIAG_PLAUS_TIA[m] = 1
      Then bit 0 of STATE_RBM_PLAUS_TIA[m] = 1
      Endif
  Endif

  If bit 1 of STATE_RBM_PLAUS_TIA[m] = 0
  Then **If** LV_INH_DIAG_RBM_PLAUS_TIA[m] = 1
    { Cross dependency of the monitor defined by the project }
    { List here failures that can inhibit TIA plausibility monitor }

Then bit 1 of STATE_RBM_PLAUS_TIA[m] = 1
Endif
Endif

bit 2 of STATE_RBM_PLAUS_TIA[m] = 1

End for m

Calculation of actual RBM rate (for application purposes)

If CTR_CDN_RBM_PLAUS_TIA_IM_CYL = 0
then RATE_RBM_PLAUS_TIA_IM_CYL = FFFFH
else RATE_RBM_PLAUS_TIA_IM_CYL
= (CTR_COMP_RBM_PLAUS_TIA_IM_CYL / CTR_CDN_RBM_PLAUS_TIA_IM_CYL)
endif

B.75 TAM Plausibility diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_PLAUS_TAM	V	0...1	0...1	1	-
Condition for TAM plausibility diagnosis: 0 = not fulfilled, 1 = fulfilled					
ERR_SYM_PLAUS_TAM	V/O	00H 01H 02H	NO_SYM PLAUS_H PLAUS_L	1	-
Symptom status for TAM plausibility diagnosis					
LV_ERR_PLAUS_TAM	V/O	0...1	0...1	1	-
Present failure for TAM plausibility diagnosis					
LV_END_DIAG_PLAUS_TAM	V/O	0...1	0...1	1	-
Result of TAM plausibility diagnosis					
LV_CDN_DIAG_PLAUS_TAM_PREL	V	0...1	0...1	1	-
Preliminary condition for TAM plausibility diagnosis: 0 = not fulfilled, 1 = fulfilled					
STATE_PLAUS_TAM	V	00H 01H 02H	NO_CHK COLD_CHK HOT_CHK	1	-
Type of TAM plausibility diagnosis check to be performed					
LV_CDN_DIAG_PLAUS_TAM_COLD	V	0...1	0...1	1	-
Condition for TAM plausibility diagnosis (engine cold check): 0 = not fulfilled, 1 = fulfilled					
ERR_SYM_PLAUS_TAM_COLD	V	00H 01H 02H	NO_SYM PLAUS_H PLAUS_L	1	-
Symptom status for TAM plausibility (engine cold check) diagnosis					
LV_END_DIAG_PLAUS_TAM_COLD	V	0...1	0...1	1	-
Result of TAM plausibility diagnosis (engine cold check)					
T_PLAUS_TAM_COLD	V	0...FFFFH	0...6553.5	0.1	s
Timer for TAM plausibility diagnosis (engine cold check)					
TIA_DIF_PLAUS_TAM_COLD	V	0...FEH	0 ... 190.5	0.75	°C
TIA variation value for TIA stability criteria (engine cold check)					
TAM_DIF_PLAUS_TAM_COLD	V	0...FEH	0 ... 190.5	0.75	°C
TAM variation value for TAM stability criteria (engine cold check)					
T_HEAT_PLAUS_HOT	V	0...FFFFH	0...1310.7	0.02	s
Timer value for intake manifold heat storage quantity evaluation (engine hot check)					
T_HEAT_PLAUS_HOT_DIF	V	8000...7FFFH	-655.36...655.34	0.02	s
Timer difference value used for intake manifold heat storage quantity evaluation (engine hot check)					
TAM_MDL_PLAUS_TAM_HOT	V/S	0...FFFFH	-48 ... 335.995	0.75 / 128	°C
Modeled TAM value for TAM plausibility diagnosis (engine hot check)					
TAM_MDL_PLAUS_TAM_HOT_BAS	V	0...FFFFH	-48 ... 335.995	0.75 / 128	°C
Modeled TAM raw value for TAM plausibility diagnosis (engine hot check)					
TAM_MDL_PLAUS_TAM_HOT_DIF	V	8000...7FFFH	-192 ... 191.994140625	0.75 / 128	°C
Ambient air temp. diff. Between TAM_MDL_PLAUS_TAM_HOT_BAS and $TAM\_MDL\_PLAUS\_TAM\_HOT_{(n-1)}$					
TIA_DIF_PLAUS_TAM_HOT	V	0...FEH	0 ... 190.5	0.75	°C
TIA variation value for TIA stability criteria (engine hot check)					
TAM_DIF_PLAUS_TAM_HOT	V	0...FEH	0 ... 190.5	0.75	°C
TAM variation value for TAM stability criteria (engine hot check)					
CTR_FIRST_CDN	V	0...2H	0 ... 2	1	-
Counter for 1^{st} condition met detected (engine hot check)					
LV_TIA_PLAUS_TAM_HOT_STAB	V	0...1	0...1	1	-
TIA stability criterium result (engine hot check): 0 = TIA stability not fulfilled, 1 = TIA stability fulfilled					
LV_TAM_PLAUS_TAM_HOT_STAB	V	0...1	0...1	1	-
TAM stability criterium result (engine hot check): 0 = TAM stability not fulfilled, 1 = TAM stability fulfilled					
LV_T_HEAT_INC	V	0...1H	0...1	1	[-]
Timer increase[decrease] flag for heat quantity storage - TAM plausibility diagnosis (engine hot check)					
LV_T_HEAT_PLAUS_HOT	V	0...1	0...1	1	-
Evaluation of Intake system heat quantity storage level: 0 = too high level, 1 = acceptable level					

LV_TAM_WIN_PLAUS_TAM_HOT	V	0...1	0...1	1	-
Vehicle/Engine conditions criterium result (engine hot check): 0 = criterium not fulfilled, 1 = criterium fulfilled					
T_PLAUS_TAM_HOT	V	0...FFFFH	0...6553.5	0.1	s
Timer value for TAM plausibility diagnosis (engine hot check)					
LV_CDN_DIAG_PLAUS_TAM_HOT	V	0...1	0...1	1	-
Condition for TAM plausibility diagnosis (engine hot check): 0 = not fulfilled, 1 = fulfilled					
ERR_SYM_PLAUS_TAM_HOT	V	00H 01H 02H	NO_SYM PLAUS_H PLAUS_L	1	-
Symptom status for TAM plausibility diagnosis (engine hot check)					
ERR_SYM_PLAUS_TAM_HOT_TMP	V	00H 01H 02H	NO_SYM PLAUS_H PLAUS_L	1	-
Temporary symptom status for TAM plausibility diagnosis (engine hot check)					
LV_END_DIAG_PLAUS_TAM_HOT	V	0...1	0...1	1	-
Result of TAM plausibility diagnosis (engine hot check)					
TEMP_DIF_PLAUS_TAM_REF_CAL_MIN	V	80...7FH	-96...95.25	0.75	°C
Minimum TEMP_DIF_PLAUS_TAM_REF recorded within the driving cycle – used only for calibration ease					
TEMP_DIF_PLAUS_TAM_REF_CAL_MAX	V	80...7FH	-96...95.25	0.75	°C
Maximum TEMP_DIF_PLAUS_TAM_REF recorded within the driving cycle – used only for calibration ease					
T_HEAT_PLAUS_HOT_REF	V	0...FFFFH	0...1310.7	0.02	s
T_HEAT_PLAUS_HOT reference value (engine hot check)					
TEMP_DIF_PLAUS_TAM_REF	-	80...7FH	-96...95.25	0.75	°C
Temperature difference reference for TAM plausibility diagnosis (common variable for engine cold + hot checks)					
T_PLAUS_TAM_HOT_REF	-	0...FFFFH	0...6553.5	0.1	s
Reference timer value for TAM plausibility diagnosis (engine hot check)					
CTR_TIA_PLAUS_TAM_HOT	-	0...FFH	0...255	1	-
Counter for TIA stability criterium definition (engine hot check)					
CTR_TAM_PLAUS_TAM_HOT	-	0...FFH	0...255	1	-
Counter for TAM stability criterium definition (engine hot check)					
CTR_FIRST_TAM	-	0...1H	0 ... 1	1	-
Counter for 1st TAM_DIF_PLAUS_TAM_COLD[HOT] calculation					
T_VS_DIAG_PLAUS_TAM_COLD	V	0...FFFFH	0...6553.5	0.1	s
Timer value for TAM acquisition above a calibrateable vehicle speed threshold					
TEMP_DIF_PLAUS_TAM_REF_MIN	V	80...7FH	-96...95.25	0.75	°C
Mimum value of TEMP_DIF_PLAUS_TAM_REF and TEMP_DIF_PLAUS_TAM_REF_VS					
TEMP_DIF_PLAUS_TAM_REF_VS	V	80...7FH	-96...95.25	0.75	°C
Temperature difference reference for TAM plausibility diagnosis for driving vehicle					

Input data:

LV_CDN_VB_MIN_DIAG	VS	ERR_SYM_PLAUS_TIA[NC_SENS_NR_TIA]	TCO
LV_TEMP_COLD_PLAUS_TAM	TAM_MES_MMV	ERR_SYM_EL_TIA[NC_SENS_NR_TIA]	N_32
LV_INH_DIAG_PLAUS_TAM	TIA_TMP	ERR_SYM_INTM_TIA[NC_SENS_NR_TIA]	T_AST
CONF_TAM	MAF_KGH	ERR_SYM_DYN_TIA[NC_SENS_NR_TIA]	TIA_THR_ST
NC_SENS_NR_TIA	LV_CT	LV_TAM_CAN_FIRST_VLD	
LC_TCHA_CONF	LV_ST_END	C_PQ_CHA_PLAUS_TIA_INC	
C_VS_PLAUS_TIA_INC	PQ_CHA[NC_NR_TCHA]	C_MAF_PLAUS_TIA_INC_H	
C_MAF_PLAUS_TIA_DEC_L	MAF	C_PQ_CHA_PLAUS_TIA_DEC	
C_VS_PLAUS_TIA_DEC	LV_IGK	C_MAF_PLAUS_TIA_DEC_H	

FUNCTION DESCRIPTION:

General information:

Version valid for: NC_TAM_CAN_USE = 1 And NC_EGR_CONF = 0

Important:

. Project with TIA sensor only (NC_TAM_CAN_USE = 0):
This diagnosis is not needed

. Project with TIA sensor + TAM sensor via CAN (NC_TAM_CAN_USE = 1):
This diagnosis can detect:

- an offset (temperature shift) on TAM value
- an offset (temperature shift) on TIA value

The result of the diagnosis must be "TAM or TIA unplausible" (1 unique Pcode)

The aim is to check whether the measured ambient temperature signal is plausible or not. This OBDII diagnosis is necessary as the ambient temperature information is used by other OBDII strategies (ex: Evap Syst. Monitoring).

Two checks are considered:

_ "engine cold check":

this test is performed only:

. if cold start conditions are verified and
. if the first TAM_MES_MMV valid value is available
. if the intake air temperature and ambient air temperatures are stable during a predefined period of time [in order to avoid false detection in case of following scenarios: use of an engine block heater / vehicle driven from a warm location to a cold location – and vice versa ...]. Additionally a temperature difference calculation above a vehicle speed threshold (for a calibrateable period of time) avoids error entries caused by warming-up in the sun.

The ambient air temperature is defined as not plausible if the difference between the ambient temperature and the air intake temperature is too high. Intermediate "cold check" symptom "PLAUS_H" is set.
The ambient temperature value is defined as plausible if it is similar to the intake air temperature value (intermediate "cold check" symptom "NO_SYM").

_ "engine hot check":

this test is performed:

. if "engine cold check" was not possible (conditions not met) and
. if the heat quantity stored in the Manifold (high storage occurs during long idle/engine shut off periods, or in high charge mode) has been sufficiently decreased
. if engine warm conditions are verified (nominal values for engine coolant temperature) and
. if the vehicle is run under defined conditions (load, engine speed, vehicle speed, ...) for a sufficient time and
. if intake air temperatures and ambiant temperatures values stability is sufficient

Then the ambient air temperature can be approached by:
TAM_MDL = TIA - offset(mass air flow, engine speed, air and engine temperature)
The ambient air temperature is defined as not plausible if the difference between the ambient temperature TAM_MES_MMV and the modeled ambient air temperature TAM_MDL is too low [respectively too high]. Intermediate "hot check" symptom "PLAUS_L" [respectively "PLAUS_H"] is set.

The ambient temperature value is defined as plausible if TAM_MDL and TAM_MES_MMV values are similar (intermediate “hot check” symptom “NO_SYM”).

Remark: the "hot check" result is available when a same symptom has been detected C_CTR_SYM_PLAUS_TAM_HOT_MAX consecutives times.
(e.g: with C_CTR_SYM_PLAUS_TAM_HOT_MAX = 3: "PLAUS_H" can be set only when this symptom has been detected 3 consecutive times).

_ TAM plausibility error definition:

. if “engine cold check” was possible:

End of diagnosis condition is set
Final error value = result from "cold test"

. if “engine cold check” was not possible and “engine hot check” was possible:

End of diagnosis condition is set
Final error value = result from "hot test"

. if “engine cold check” and “engine hot check” were both not possible:

End of diagnosis condition is not set (diagnosis could not be done within this driving cycle)
Final error value = "NO_SYM"

Signal flow diagram:

Description:

Application conditions:

Initialisation:

At Clear of FMY

LV_CDN_DIAG_PLAUS_TAM = 0
LV_END_DIAG_PLAUS_TAM = 0
ERR_SYM_PLAUS_TAM = NO_SYM
LV_ERR_PLAUS_TAM = 0

T_HEAT_PLAUS_HOT = T_HEAT_PLAUS_HOT_REF = C_T_HEAT_PLAUS_HOT_INI

TIA_PLAUS_TAM_COLD_MIN = TIA_PLAUS_TAM_COLD_MAX
= TIA_PLAUS_TAM_HOT_MIN = TIA_PLAUS_TAM_HOT_MAX
= TAM_MDL_PLAUS_TAM_HOT
= TIA_TMP

TAM_PLAUS_TAM_COLD_MIN = TAM_PLAUS_TAM_COLD_MAX
= TAM_PLAUS_TAM_HOT_MIN = TAM_PLAUS_TAM_HOT_MAX
= TAM_MES_MMV

TEMP_DIF_PLAUS_TAM_REF_CAL_MIN = +95.25°C
TEMP_DIF_PLAUS_TAM_REF_CAL_MAX = -96.00°C

All other variables from Output data list - excepted "specific variable list" (see below) must be initialized to 0H

At *LV_ES = 0 -> 1*

T_VS_DIAG_PLAUS_TAM_COLD = C_T_VS_DIAG_PLAUS_TAM_COLD

At *Reset*

Same initialization variable list/procedure as at "Clear of FMY" (see above) + specific variable list (see below):

Specific variable list:

TAM_MDL_PLAUS_TAM_HOT = TAM_MDL_PLAUS_TAM_HOT_BAS are initialized with:

If there is no previously TAM_MDL_PLAUS_TAM_HOT value saved in the NVMY
Or the EEPROM does not work
Then 20.25°C
Else Min(TAM_MDL_PLAUS_TAM_HOT saved in NVMY; TIA_TMP)
Endif

TAM_MDL_PLAUS_TAM_HOT_DIF = 0

(comment: these 3 variables must not be reseted at "Clear of FMY" otherwise it leads to false detection – due to the filtering on TAM_MDL_PLAUS_TAM_HOT)

T_VS_DIAG_PLAUS_TAM_COLD = C_T_VS_DIAG_PLAUS_TAM_COLD

Recurrence: 100ms *(necessary as OBD2 diag. / even if TAM calc. Rec. = 1s only)*

Activation: CONF_TAM ≠ 0

Deactivation: activation condition not met

Formula section:

Remark: recurrence for TAM_MES_MMV = 100ms

➢ ***Common preliminary conditions:***

If LV_IGK = 1
And LV_ST_END = 1
And LV_CDN_VB_MIN_DIAG = 1
And LV_INH_DIAG_PLAUS_TAM = 0
And (LV_TAM_CAN_FIRST_VLD = 1 **Or** CONF_TAM = 2)
And TCO > C_TCO_PLAUS_TAM_MIN
(remark: TAM is not consummed in TCO plaus diag)
And All ERR_SYM_EL_TIA[m] = 0 [NC_SENS_NR_TIA values]
And All ERR_SYM_INTM_TIA[m] = 0 [NC_SENS_NR_TIA values]
And All ERR_SYM_DYN_TIA[m] = 0 [NC_SENS_NR_TIA values]
And All ERR_SYM_PLAUS_TIA[m] = 0 [NC_SENS_NR_TIA values]
Then LV_CDN_DIAG_PLAUS_TAM_PREL = 1
Else LV_CDN_DIAG_PLAUS_TAM_PREL = 0
T_PLAUS_TAM_COLD = 0
T_PLAUS_TAM_HOT = 0
T_PLAUS_TAM_HOT_REF = 0
Endif

If LV_CDN_DIAG_PLAUS_TAM_PREL = 1
Then Computation of Section "common part"
Endif

--

➢ *Section "common part"*

- *Intake system heat quantity storage:*

If [(LV_CT = 1
    And VS <= C_VS_PLAUS_TIA_INC)
Or (MAF_KGH >= C_MAF_PLAUS_TIA_INC_H
      And PQ_CHA[NC_NR_TCHA] >= C_PQ_CHA_PLAUS_TIA_INC)]
Then LV_T_HEAT_INC = 1
Else **If** (C_MAF_PLAUS_TIA_DEC_L <= MAF_KGH <= C_MAF_PLAUS_TIA_DEC_H
        And PQ_CHA[NC_NR_TCHA] <= C_PQ_CHA_PLAUS_TIA_DEC
        And VS >= C_VS_PLAUS_TIA_DEC)
        Then LV_T_HEAT_INC = 0
        Endif
Endif

If LV_T_HEAT_INC <> LV_T_HEAT_INC(n-1)
Then T_HEAT_PLAUS_HOT_REF = T_HEAT_PLAUS_HOT(n-1)
Endif

If LV_T_HEAT_INC = 1
Then T_HEAT_PLAUS_HOT = T_HEAT_PLAUS_HOT(n-1)
        + ID_T_HEAT_PLAUS_HOT_INC(T_HEAT_PLAUS_HOT_DIF, TCO)
Else T_HEAT_PLAUS_HOT = T_HEAT_PLAUS_HOT(n-1)
        – ID_T_HEAT_PLAUS_HOT_DEC(T_HEAT_PLAUS_HOT_DIF, TCO)
Endif

T_HEAT_PLAUS_HOT
= Min(T_HEAT_PLAUS_HOT, IP_T_HEAT_PLAUS_HOT_TOL)

If T_HEAT_PLAUS_HOT = 0
Then T_HEAT_PLAUS_HOT_REF = 0 *(Counter Reset)*
Endif *(T_HEAT_PLAUS_HOT_REF is unchanged)*

T_HEAT_PLAUS_HOT_DIF = T_HEAT_PLAUS_HOT_REF – T_HEAT_PLAUS_HOT

- *TAM model for TAM plausibility diagnosis:*

If LC_TCHA_CONF = 0
Then TAM_MDL_OFS_PLAUS_HOT = IP_TAM_MDL_OFS_PLAUS_0(N_32, MAF)
Else TAM_MDL_OFS_PLAUS_HOT = IP_TAM_MDL_OFS_PLAUS_1(N_32, MAF)
Endif

TAM_MDL_PLAUS_TAM_HOT_BAS
= TIA_TMP – {[TAM_MDL_OFS_PLAUS_HOT
        + IP_TAM_MDL_OFS_PLAUS_VS(VS)]
        * IP_FAC_TAM_MDL_PLAUS_TAM_HOT(TIA_TMP, TCO)}
TAM_MDL_PLAUS_TAM_HOT_DIF
= TAM_MDL_PLAUS_TAM_HOT_BAS - TAM_MDL_PLAUS_TAM_HOT(n-1)
TAM_MDL_PLAUS_TAM_HOT(n)
= TAM_MDL_PLAUS_TAM_HOT(n-1)
+ [IP_CRLC_TAM_MDL_PLAUS_TAM_HOT(TAM_MDL_PLAUS_TAM_HOT_DIF)

* TAM_MDL_PLAUS_TAM_HOT_DIF]

- *Cold or hot check selection:*

If LV_TEMP_COLD_PLAUS_TAM = 1 //COLD CHECK
Then **If** T_VS_DIAG_PLAUS_TAM_COLD > 0
Then TEMP_DIF_PLAUS_TAM_REF_VS = TAM_MES_MMV – TIA_THR_ST

If VS > C_VS_DIAG_PLAUS_TAM_COLD
Then T_VS_DIAG_PLAUS_TAM_COLD -- //timer is decremented
Else T_VS_DIAG_PLAUS_TAM_COLD= C_T_VS_DIAG_PLAUS_TAM_COLD
Endif
Endif

// Temperature difference calculation in order to avoid error entries caused by warming-up in the sun.

If T_PLAUS_TAM_COLD(n-1) < C_T_PLAUS_TAM_COLD_MAX
(remark: as currently TAM rec. = 1s, it makes only sense to calibrate C_T_PLAUS_TAM_COLD_MAX with a multiple of 1)
Then TEMP_DIF_PLAUS_TAM_REF = TAM_MES_MMV - TIA_THR_ST
T_PLAUS_TAM_COLD = T_PLAUS_TAM_COLD(n-1) + 0.1 (100ms)

- *Stability check:*

TIA_PLAUS_TAM_COLD_MIN
= MIN(TIA_TMP, TIA_PLAUS_TAM_COLD_MIN(n-1))
TIA_PLAUS_TAM_COLD_MAX
= MAX(TIA_TMP, TIA_PLAUS_TAM_COLD_MAX(n-1))
TIA_DIF_PLAUS_TAM_COLD
= |TIA_PLAUS_TAM_COLD_MAX -TIA_PLAUS_TAM_COLD_MIN|

If CTR_FIRST_TAM = 0
Then TAM_PLAUS_TAM_COLD_MIN = TAM_MES_MMV
TAM_PLAUS_TAM_COLD_MAX = TAM_MES_MMV
CTR_FIRST_TAM = CTR_FIRST_TAM + 1
Else TAM_PLAUS_TAM_COLD_MIN
= MIN(TAM_MES_MMV, TAM_PLAUS_TAM_COLD_MIN(n-1))
TAM_PLAUS_TAM_COLD_MAX
= MAX(TAM_MES_MMV, TAM_PLAUS_TAM_COLD_MAX(n-1))
Endif
TAM_DIF_PLAUS_TAM_COLD
= |TAM_PLAUS_TAM_COLD_MAX -TAM_PLAUS_TAM_COLD_MIN|

Else If TIA_DIF_PLAUS_TAM_COLD <= C_TIA_DIF_PLAUS_TAM_COLD_MAX
And
TAM_DIF_PLAUS_TAM_COLD <= C_TAM_DIF_PLAUS_TAM_COLD_MAX
Then STATE_PLAUS_TAM = "COLD_CHK" //computation of section "cold check"
Else STATE_PLAUS_TAM = "HOT_CHK" //computation of section "hot check"
Endif
Endif
Else STATE_PLAUS_TAM = HOT_CHK //computation of section "hot check"
Endif

➢ Section "cold check"

(0) If STATE_PLAUS_TAM = "COLD_CHK"

(only "cold check" will be performed – if conditions met)

(0) Then (cold check)

- *Cold check result:*

```
(1) If        T_AST < C_T_AST_TAM_PLAUS_COLD_MAX
                                         //Engine was started a short time ago
(1) Then   (2)If     T_VS_DIAG_PLAUS_TAM_COLD = 0    //vehicle was driven
           (2)Then TEMP_DIF_PLAUS_TAM_REF_MIN =
             MIN [ |TEMP_DIF_PLAUS_TAM_REF|, |TEMP_DIF_PLAUS_TAM_REF_VS| ]

                // min temperature difference value of standing and driven vehicle

                (3) If          LV_END_DIAG_PLAUS_TAM_COLD = 0
                (3)Then         LV_CDN_DIAG_PLAUS_TAM_COLD = 1

                     (4) If      TEMP_DIF_PLAUS_TAM_REF_MIN
                                  >= C_TEMP_DIF_PLAUS_TAM_COLD_THD_1
                     (4) Then ERR_SYM_PLAUS_TAM_COLD =  "PLAUS_H"
                                         // means TAM too high or TIA too low
                     (4) Else  ERR_SYM_PLAUS_TAM_COLD =  "NO_SYM"
                       // means TAM and TIA have a delta lower than error threshold
                     (4) Endif
                     LV_END_DIAG_PLAUS_TAM_COLD = 1
                     Computation of Section "Final error definition"
                (3) Else LV_CDN_DIAG_PLAUS_TAM_COLD = 0
                (3) Endif
           (2) Endif

(1) Else   (5) If       LV_END_DIAG_PLAUS_TAM_COLD = 0
           (5) Then    STATE_PLAUS_TAM = "HOT_CHK" //computation of section "hot check"
           (5) Endif
(1) Endif
```

Hint: ERR_SYM_PLAUS_TAM_COLD = "PLAUS_L" is still defined but no longer used

Additionnal variables displayed on application tool to ease calibration:

TEMP_DIF_PLAUS_TAM_REF_CAL_MIN
= MIN[TEMP_DIF_PLAUS_TAM_REF,TEMP_DIF_PLAUS_TAM_REF_CAL_MIN$_{(n-1)}$]
TEMP_DIF_PLAUS_TAM_REF_CAL_MAX
= MAX[TEMP_DIF_PLAUS_TAM_REF,TEMP_DIF_PLAUS_TAM_REF_CAL_MAX$_{(n-1)}$]

(0) Else LV_CDN_DIAG_PLAUS_TAM_COLD = 0
(0) Endif

➢ Section "hot check"

If STATE_PLAUS_TAM = "HOT_CHK" (only "hot check" will be performed - if conditions met)

Then (hot check)

TEMP_DIF_PLAUS_TAM_REF = TAM_MES_MMV - TAM_MDL_PLAUS_TAM_HOT

- *TIA stability criterium:*

CTR_TIA_PLAUS_TAM_HOT = CTR_TIA_PLAUS_TAM_HOT(n-1) +1
TIA_PLAUS_TAM_HOT_MIN = MIN(TIA_TMP, TIA_PLAUS_TAM_HOT_MIN(n-1))
TIA_PLAUS_TAM_HOT_MAX = MAX(TIA_TMP, TIA_PLAUS_TAM_HOT_MAX(n-1))
TIA_DIF_PLAUS_TAM_HOT =
|TIA_PLAUS_TAM_HOT_MAX -TIA_PLAUS_TAM_HOT_MIN|
If CTR_TIA_PLAUS_TAM_HOT < C_CTR_TEMP_PLAUS_TAM_HOT_MAX
Then If TIA_DIF_PLAUS_TAM_HOT > C_TIA_DIF_PLAUS_TAM_HOT_MAX
    Then LV_TIA_PLAUS_TAM_HOT_STAB = 0
        CTR_TIA_PLAUS_TAM_HOT = 0
        TIA_PLAUS_TAM_HOT_MIN = TIA_TMP
        TIA_PLAUS_TAM_HOT_MAX = TIA_TMP
        TIA_DIF_PLAUS_TAM_HOT = 0
    Endif
Else If TIA_DIF_PLAUS_TAM_HOT > C_TIA_DIF_PLAUS_TAM_HOT_MAX
    Then LV_TIA_PLAUS_TAM_HOT_STAB = 0
    Else LV_TIA_PLAUS_TAM_HOT_STAB = 1
    Endif
    CTR_TIA_PLAUS_TAM_HOT = 0
    TIA_PLAUS_TAM_HOT_MIN = TIA_TMP
    TIA_PLAUS_TAM_HOT_MAX = TIA_TMP
    TIA_DIF_PLAUS_TAM_HOT = 0
Endif

- *TAM stability criterium::*

CTR_TAM_PLAUS_TAM_HOT = CTR_TAM_PLAUS_TAM_HOT(n-1) +1
If CTR_FIRST_TAM = 0
Then TAM_PLAUS_TAM_HOT_MIN = TAM_MES_MMV
    TAM_PLAUS_TAM_HOT_MAX = TAM_MES_MMV
    CTR_FIRST_TAM = CTR_FIRST_TAM + 1
Else TAM_PLAUS_TAM_HOT_MIN
    = MIN(TAM_MES_MMV, TAM_PLAUS_TAM_HOT_MIN(n-1))
    TAM_PLAUS_TAM_HOT_MAX
    = MAX(TAM_MES_MMV, TAM_PLAUS_TAM_HOT_MAX(n-1))
Endif
TAM_DIF_PLAUS_TAM_HOT =
|TAM_PLAUS_TAM_HOT_MAX -TAM_PLAUS_TAM_HOT_MIN|
If CTR_TAM_PLAUS_TAM_HOT < C_CTR_TEMP_PLAUS_TAM_HOT_MAX
(remark: as currently TAM rec. = 1s, it makes only sense to calibrate C_CTR_TEMP_PLAUS_TAM_HOT_MAX *with a multiple of 10: 10*100ms=1s)*
Then If TAM_DIF_PLAUS_TAM_HOT > C_TAM_DIF_PLAUS_TAM_HOT_MAX
    Then LV_TAM_PLAUS_TAM_HOT_STAB = 0
        CTR_TAM_PLAUS_TAM_HOT = 0
        TAM_PLAUS_TAM_HOT_MIN = TAM_MES_MMV
        TAM_PLAUS_TAM_HOT_MAX = TAM_MES_MMV

```
            TAM_DIF_PLAUS_TAM_HOT = 0
        Endif
Else  If TAM_DIF_PLAUS_TAM_HOT > C_TAM_DIF_PLAUS_TAM_HOT_MAX
        Then LV_TAM_PLAUS_TAM_HOT_STAB = 0
        Else LV_TAM_PLAUS_TAM_HOT_STAB = 1
        Endif
        CTR_TAM_PLAUS_TAM_HOT = 0
        TAM_PLAUS_TAM_HOT_MIN = TAM_MES_MMV
        TAM_PLAUS_TAM_HOT_MAX = TAM_MES_MMV
        TAM_DIF_PLAUS_TAM_HOT = 0
Endif
```

- *Evaluation of Intake system heat quantity storage level:*

```
If T_HEAT_PLAUS_HOT > IP_T_HEAT_PLAUS_HOT_MIN(TCO)
Then LV_T_HEAT_PLAUS_HOT = 0
Else If T_HEAT_PLAUS_HOT <= IP_T_HEAT_PLAUS_HOT_MAX(TCO)
        Then LV_T_HEAT_PLAUS_HOT = 1
        Endif
Endif
```

- *Vehicle/Engine conditions criteria:*

```
If C_TCO_PLAUS_TAM_HOT_MIN < TCO < C_TCO_PLAUS_TAM_HOT_MAX
And C_N_PLAUS_TAM_HOT_MIN < N_32 < C_N_PLAUS_TAM_HOT_MAX
And C_MAF_PLAUS_TAM_HOT_MIN < MAF_KGH < C_MAF_PLAUS_TAM_HOT_MAX
And PQ_CHA[NC_NR_TCHA] <= C_PQ_CHA_PLAUS_TAM_HOT_MAX
And C_VS_PLAUS_TAM_HOT_MIN =< VS < C_VS_PLAUS_TAM_HOT_MAX
Then LV_TAM_WIN_PLAUS_TAM_HOT = 1
Else  LV_TAM_WIN_PLAUS_TAM_HOT = 0
Endif
```

- *Different criteria combination:*

```
If LV_TAM_WIN_PLAUS_TAM_HOT = 1
And LV_TIA_PLAUS_TAM_HOT_STAB = 1
And LV_TAM_PLAUS_TAM_HOT_STAB = 1
And LV_T_HEAT_PLAUS_HOT = 1 (low level of Heat quantity stored in the Intake Manifold)
Then If T_PLAUS_TAM_HOT(n-1) < C_T_PLAUS_TAM_HOT_MAX
        Then T_PLAUS_TAM_HOT_REF = T_PLAUS_TAM_HOT(n-1) + 0.1 (100ms)
            T_PLAUS_TAM_HOT = T_PLAUS_TAM_HOT_REF
        Endif
Else  T_PLAUS_TAM_HOT_REF = T_PLAUS_TAM_HOT_REF(n-1)
        If (T_PLAUS_TAM_HOT_REF - T_PLAUS_TAM_HOT(n-1) ) <
          C_T_PLAUS_TAM_HOT_REF_DIF
        Then T_PLAUS_TAM_HOT
             = T_PLAUS_TAM_HOT(n-1) – C_T_PLAUS_TAM_HOT_DEC
        Else  T_PLAUS_TAM_HOT = 0
        Endif
Endif

If T_PLAUS_TAM_HOT = 0
Then T_PLAUS_TAM_HOT_REF = 0 (reset)
Endif (T_PLAUS_TAM_HOT_REF is unchanged)
```

- *Hot check result:*

If [(LV_END_DIAG_PLAUS_TAM_HOT(n-1) = 0
Or LC_END_DIAG_PLAUS_TAM_SUPP = 1)
And T_PLAUS_TAM_HOT >= C_T_PLAUS_TAM_HOT_MAX]
Then If [|TEMP_DIF_PLAUS_TAM_REF| >= C_TEMP_DIF_PLAUS_TAM_HOT_THD_1
Or |TEMP_DIF_PLAUS_TAM_REF| <= C_TEMP_DIF_PLAUS_TAM_HOT_THD_0]
Then LV_CDN_DIAG_PLAUS_TAM_HOT = 1
If TEMP_DIF_PLAUS_TAM_REF >=
C_TEMP_DIF_PLAUS_TAM_HOT_THD_1
Then ERR_SYM_PLAUS_TAM_HOT_TMP = "**PLAUS_H**"
Else If TEMP_DIF_PLAUS_TAM_REF <=
- C_TEMP_DIF_PLAUS_TAM_HOT_THD_1
Then ERR_SYM_PLAUS_TAM_HOT_TMP = "**PLAUS_L**"
Else ERR_SYM_PLAUS_TAM_HOT_TMP= "**NO_SYM**"
(as |TEMP_DIF_..| <= C_TEMP_DIF_.._HOT_THD_0)
Endif
Endif
Symptom filtering for "hot check":
T_PLAUS_TAM_HOT = 0 *(Reset)*
CTR_CDN_FIRST = CTR_CDN_FIRST +1
If ERR_SYM_PLAUS_TAM_HOT_TMP
= ERR_SYM_PLAUS_TAM_HOT_TMP(n-1)
(allows counter increase when SYM is confirmed)
Or CTR_CDN_FIRST = 1
(allows counter increase the 1st time CDN are met whatever SYM value)
Then CTR_SYM_PLAUS_TAM_HOT = CTR_SYM_PLAUS_TAM_HOT + 1
Else CTR_SYM_PLAUS_TAM_HOT = 0 *(Reset compulsory)*
CTR_CDN_FIRST = 0 *(Reset – to prepare next test: _TMP=_TMP(n-1)?)*
Endif
If CTR_SYM_PLAUS_TAM_HOT >= C_CTR_SYM_PLAUS_TAM_HOT_MAX
Then ERR_SYM_PLAUS_TAM_HOT = ERR_SYM_PLAUS_TAM_HOT_TMP
LV_END_DIAG_PLAUS_TAM_HOT = 1
CTR_CDN_FIRST = 0
(Reset – allows the possibility of a continuous diagnosis – hot check part)
CTR_SYM_PLAUS_TAM_HOT = 0
(Reset – allows the possibility of a continuous diagnosis – hot check part)
Computation of Section "Final error definition"
Endif
Else LV_CDN_DIAG_PLAUS_TAM_HOT = 0
(no conclusion possible from "hot check": next attempt when retrieving the conditions for "hot check" – or - at next DC)
Endif

Additionnal variables displayed on application tool to ease calibration:
TEMP_DIF_PLAUS_TAM_REF_CAL_MIN
= MIN[TEMP_DIF_PLAUS_TAM_REF,TEMP_DIF_PLAUS_TAM_REF_CAL_MIN$_{(n-1)}$]
TEMP_DIF_PLAUS_TAM_REF_CAL_MAX
= MAX[TEMP_DIF_PLAUS_TAM_REF,TEMP_DIF_PLAUS_TAM_REF_CAL_MAX$_{(n-1)}$]

Else LV_CDN_DIAG_PLAUS_TAM_HOT = 0
Endif
Endif
--

➢ ***Section "Final error definition"***

If (LV_END_DIAG_PLAUS_TAM(n-1) = 0
Or LC_END_DIAG_PLAUS_TAM_SUPP = 1)
Then If (LV_END_DIAG_PLAUS_TAM_**COLD** = 1
Or LV_END_DIAG_PLAUS_TAM_**HOT** = 1)
Then LV_CDN_DIAG_PLAUS_TAM = 1
If LV_END_DIAG_PLAUS_TAM_**COLD** = 1
Then ERR_SYM_PLAUS_TAM = ERR_SYM_PLAUS_TAM_**COLD**
Endif
If LV_END_DIAG_PLAUS_TAM_**HOT** = 1
Then ERR_SYM_PLAUS_TAM = ERR_SYM_PLAUS_TAM_**HOT**
Endif
If ERR_SYM_PLAUS_TAM = “NO_SYM”
Then LV_ERR_PLAUS_TAM = 0
Else LV_ERR_PLAUS_TAM = 1
Endif
LV_END_DIAG_PLAUS_TAM = 1
Else LV_CDN_DIAG_PLAUS_TAM = 0
Endif
Else LV_CDN_DIAG_PLAUS_TAM = 0
Endif
--

Remark:
ERR_SYM_PLAUS_TAM_HOT, ERR_SYM_PLAUS_TAM_COLD as well as LV_END_DIAG_PLAUS_TAM_HOT, LV_END_DIAG_PLAUS_TAM_COLD are only intermediate variables (they are not handled by Error Management).
The final error result is characterized by LV_CDN_DIAG_PLAUS_TAM , ERR_SYM_PLAUS_TAM, LV_ERR_PLAUS_TAM and LV_END_DIAG_PLAUS_TAM

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_HEAT_PLAUS_HOT_INI	1	0...FFFFH	0...1310.7	0.02	s
Initialization value for T_HEAT_PLAUS_HOT[REF] counters					
C_TCO_PLAUS_TAM_MIN	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO value for TAM plausibility diagnosis activation					
C_T_PLAUS_TAM_COLD_MAX	1	0...FFFFH	0...6553.5	0.1	s
Maximum value of counter T_PLAUS_TAM_COLD					
C_TIA_DIF_PLAUS_TAM_COLD_MAX	1	0...FEH	0 ... 190.5	0.75	°C
Maximum allowed TIA variation (engine cold check)					
C_TAM_DIF_PLAUS_TAM_COLD_MAX	1	0...FEH	0 ... 190.5	0.75	°C
Maximum allowed TAM variation (engine cold check)					
C_TIA_DIF_PLAUS_TAM_HOT_MAX	1	0...FEH	0 ... 190.5	0.75	°C
Maximum allowed TIA variation (engine hot check)					
C_TAM_DIF_PLAUS_TAM_HOT_MAX	1	0...FEH	0 ... 190.5	0.75	°C
Maximum allowed TAM variation (engine hot check)					
C_TEMP_DIF_PLAUS_TAM_COLD_THD_1	1	0...FEH	0 ... 190.5	0.75	°C
Offset temperature threshold value (engine cold check) commun for PLAUS_H/PLAUS_L - failure - definition					
C_TCO_PLAUS_TAM_HOT_MIN	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_TCO_PLAUS_TAM_HOT_MAX	1	0...FEH	-48...142.5	0.75	°C
Maximum TCO value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_N_PLAUS_TAM_HOT_MIN	1	0...FFH	0...8160	32	rpm
Minimum N value for timer T_PLAUS_TAM_HOT increment (engine hot check)					

C_N_PLAUS_TAM_HOT_MAX	1	0...FFH	0...8160	32	rpm
Maximum N value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_MAF_PLAUS_TAM_HOT_MIN	1	0...FFFFH	0...2047.9688	0.03125	kg/h
Minimum MAF value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_MAF_PLAUS_TAM_HOT_MAX	1	0...FFFFH	0...2047.9688	0.03125	kg/h
Maximum MAF value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_PQ_CHA_PLAUS_TAM_HOT_MAX	1	0...FFFFH	0 ..15.999756	244.14e-6	-
Maximum PQ_CHA value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_VS_PLAUS_TAM_HOT_MIN	1	0...FFH	0...255	1	Km/h
Minimum VS value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_VS_PLAUS_TAM_HOT_MAX	1	0...FFH	0...255	1	Km/h
Maximum VS value for timer T_PLAUS_TAM_HOT increment (engine hot check)					
C_T_PLAUS_TAM_HOT_REF_DIF	1	0...FFFFH	0...6553.5	0.1	s
Reference T_PLAUS_TAM_HOT value used for decremention management of T_PLAUS_TAM_HOT					
C_T_PLAUS_TAM_HOT_DEC	1	0...FFFFH	0...6553.5	0.1	s
timer decrerment value for TAM plausibility diagnosis (engine hot check)					
C_T_PLAUS_TAM_HOT_MAX	1	0...FFFFH	0...6553.5	0.1	s
Maximum value of the timer T_PLAUS_TAM_HOT (engine hot check)					
C_TEMP_DIF_PLAUS_TAM_HOT_THD_1	1	0...FEH	0 ... 190.5	0.75	°C
Offset temperature threshold value (engine hot check) commun for PLAUS_H/PLAUS_L - failure - definition					
C_TEMP_DIF_PLAUS_TAM_HOT_THD_0	1	0...FEH	0 ... 190.5	0.75	°C
Offset temperature threshold value (engine hot check) commun for PLAUS_H/PLAUS_L - no failure - definition					
C_CTR_SYM_PLAUS_TAM_HOT_MAX	1	0...FFH	0 ... 255	1	-
Maximum value of the counter used for symptom validation (engine hot check)					
C_CTR_TEMP_PLAUS_TAM_HOT_MAX	1	0...FFH	0...255	1	-
Maximum value of the counter used for TIA and TAM stability criteria definition (engine hot check)					
IP_CRLC_TAM_MDL_PLAUS_TAM_HOT	1*6	0...FFFFH	0 ... 0.999984	0.15259e-4	-
LDP_TAM_MDL_PLAUS_TAM_HOT_DIF	6	0...FFFFH	-192 ...192	0.75 / 128	°C
TAM_MDL_PLAUS_TAM_HOT mean value filtering factor					
IP_TAM_MDL_OFS_PLAUS_0	8*8	0...FFFFH	-192 ...192	0.75 / 128	°C
LDP_N_32_IP_TAM_MDL_OFS	8	0...FFH	0...8160	32	rpm
LDP_MAF_IP_TAM_MDL_OFS	10	0...FFFFH	0...1389	0.0212	mg/stk
Offset for TAM_MDL_PLAUS_TAM – versus temperatures – for natural aspirated engine (LC_TCHA_CONF = 0)					
IP_TAM_MDL_OFS_PLAUS_1	8*8	0...FFFFH	-192 ...192	0.75 / 128	°C
LDP_N_32_IP_TAM_MDL_OFS	8	0...FFH	0...8160	32	rpm
LDP_MAF_IP_TAM_MDL_OFS	10	0...FFFFH	0...1389	0.0212	mg/stk
Offset for TAM_MDL_PLAUS_TAM – versus temperatures – for turbo engine (LC_TCHA_CONF = 1)					
IP_TAM_MDL_OFS_PLAUS_VS	1*6	0...FFFFH	-192 ...192	0.75 / 128	°C
LDP_VS_IP_TAM_MDL_OFS	6	0...FFH	0...255	1	Km/h
Offset for TAM_MDL_PLAUS_TAM determination – versus vehicle speed					
IP_FAC_TAM_MDL_PLAUS_TAM_HOT	8*8	0...FFH	0...3.984	0.0157	-
LDP_TIA_TMP_IP_FAC_TAM_MDL	8	0...FEH	-48...142.5	0.75	°C
LDP_TCO_IP_FAC_TAM_MDL	8	0...FEH	-48...142.5	0.75	°C
Correction factor mapping for TAM_MDL_OFS_PLAUS_TAM_HOT calculation. Typical value: 1					

ID_T_HEAT_PLAUS_HOT_INC	5*4	0...FFFFH	0...1310.7	0.02	s
LDPM_T_HEAT_PLAUS_DIF_ID_T_HEAT	5	0...FFFFH	-655.36...655.34	0.02	s
LDPM_TCO_ID_T_HEAT	4	0...FEH	-48...142.5	0.75	°C
Timer increrment value for T_HEAT_PLAUS_HOT counter					
ID_T_HEAT_PLAUS_HOT_DEC	5*4	0...FFFFH	0...1310.7	0.02	s
LDPM_T_HEAT_PLAUS_DIF_ID_T_HEAT	5	0...FFFFH	-655.36...655.34	0.02	s
LDPM_TCO_ID_T_HEAT	4	0...FEH	-48...142.5	0.75	°C
Timer decrerment value for T_HEAT_PLAUS_HOT counter					
IP_T_HEAT_PLAUS_HOT_MIN	1*4	0...FFFFH	0...1310.7	0.02	s
LDPM_TCO_ID_T_HEAT	4	0...FEH	-48...142.5	0.75	°C
Over this threshold value, the heat stored in the Intake manifold is too high to perform TAM plaus diagnosis					
IP_T_HEAT_PLAUS_HOT_MAX	1*4	0...FFFFH	0...1310.7	0.02	s
LDPM_TCO_ID_T_HEAT	4	0...FEH	-48...142.5	0.75	°C
Below this threshold value, the heat stored in the Intake manifold is low enough to perform TAM plaus diagnosis					
IP_T_HEAT_PLAUS_HOT_TOL	1*4	0...FFFFH	0...1310.7	0.02	s
LDPM_TCO_ID_T_HEAT	4	0...FEH	-48...142.5	0.75	°C
Maximum possible value for heat stored in the Intake manifold					
LC_END_DIAG_PLAUS_TAM_SUPP	1	0...1H	0...1	1	-
Switch to suppress the effect of LV_END_DIAG_PLAUS_TAM/TAM_HOT on TAM plaus. Diag. activations cond.					
C_VS_DIAG_PLAUS_TAM_COLD	1	0...FFH	0...255	1	Km/h
Minimum VS threshold for the acquisition of TAM during cold check					
C_T_VS_DIAG_PLAUS_TAM_COLD	1	0...FFFFH	0...6553.5	0.1	s
Maximum value of the timer T_VS_DIAG_PLAUS_TAM_COLD (timer for TAM acquisition above VS thd.)					
C_T_AST_TAM_PLAUS_COLD_MAX	1	0...FFFFH	0...6553.5	0.1	s
Time after start threshold for cold check diagnosis					

B.76 TAM Plausibility Diagnostic (application incidence)

B.76.1 General:

The functionality described in this module is necessary only if TAM plausibility diagnosis functionality is used by the Project.

Input data:

LV_ERR_EL_TIA[NC_SENS_NR_TIA]	LV_LIH_ERR_CRK
LV_ERR_PUT_PLAUS	LV_ERR_SYN_CAM_IN_i
LV_ERR_DYN_TIA[NC_SENS_NR_TIA]	LV_ERR_CRK_TOOTH_PER
LV_ERR_TCO_PLAUS	LV_ERR_TCO_STUCK
LV_ERR_CAN_BUS_OFF	CONF_TAM
LV_ERR_MAP_PLAUS	LV_ERR_AMP_PLAUS
LV_ERR_PER_CAM_IN_i	LV_ERR_REF_CRK_CAM_IN_i
LV_ERR_CRK_TOOTH	LV_ERR_SYN_CRK_CAM_IN_i
LV_ERR_CRK_PLAUS	LV_ST_END
LV_ERR_CRK_SYN	LV_ERR_LOAD_PLAUS
T_AST	LV_ERR_TCO_STUCK_H
LV_ERR_PLAUS_CAM_IN_i	NC_SENS_NR_TIA
LV_ERR_INTM_TIA[NC_SENS_NR_TIA]	T_ES
LV_IGK	LV_ERR_TCO
LV_ERR_AMP_PLAUS	LV_ERR_TCO_GRD
LV_ERR_AMP	LV_DC
LV_END_DIAG_PLAUS_TAM	LV_ERR_VS
LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA]	CTR_ERR_DYN_NR
LV_T_ES_NOT_PLAUS	TCO_ST
LV_ERR_MAF	LV_ERR_PUT
LV_ERR_CAN_ICL_TAM	LV_ERR_TAM_CAN
LV_ERR_MAP_PLAUS	LV_ERR_MAP
LV_ERR_CAM	LV_ERR_TCO_EL
CTR_CDN_RBM_PLAUS_TAM	CTR_COMP_RBM_PLAUS_TAM
TCO	TCO_STOP
LV_ERR_EL_TAM	LV_ERR_EL_MAP
LV_ERR_EL_PUT	LV_ERR_VCC_SENS_SUB
LV_ERR_VCC_PVS_2	

B.76.2 General:

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_PLAUS_TAM	V/O	0..1H	0..1	1	-
Inhibition condition for TAM plausibility diagnosis					
LV_INH_DIAG_RBM_PLAUS_TAM	V	0...1H	0...1	1	-
TAM plausibility diagnosis inhibition condition for RBM interface definition					
LV_TEMP_COLD_PLAUS_TAM	V/O	0..1H	0..1	1	-
= 1: Engine is cold (cold start condition reached) / = 0: Engine is still warm					

FUNCTION DESCRIPTION:

General information:

The inhibition flag LV_INH_DIAG_PLAUS_TAM allows to deactivate the corresponding diagnostic.

Description:

Application conditions:

Initialization: **At** Reset **Or** Transition LV_IGK=0 to 1 :
LV_TEMP_COLD_PLAUS_TAM = 0
LV_INH_DIAG_PLAUS_TAM = 0
LV_INH_DIAG_RBM_PLAUS_TAM = 0Reccurence:
100 ms

Activation: CONF_TAM ≠ 0 **And** LV_ST_END = 1

Deactivation: activation condition not met

Formula section:

- ***irreversible condition LV_TEMP_COLD_PLAUS_TAM for the cold check***

If T_ES >= C_T_ES_PLAUS_TAM_MIN
and LV_T_ES_NOT_PLAUS = 0
and T_AST < C_T_AST_MAX_PLAUS_TAM_T_ES
--------------------- Consideration of external heater start ---------------------
and TCO_STOP - TCO_ST >= C_TCO_DIF_PLAUS_TAM_MIN *(40°C)*
and TCO < C_TCO_PLAUS_TAM_MAX *(34°C)*
--------------------- *Consideration of external heater start* ---------------------
Then LV_TEMP_COLD_PLAUS_TAM = 1
Endif *(LV_TEMP_COLD_PLAUS_TAM remains unchanged)*

- ***common inhibit flag for cold and hot check***

If LV_ERR_MAP = 0 *(EL failure – MAP system: INSY) [MAF_KGH]*
And LV_ERR_MAF = 0 *(EL failure – MAF system: INSY) [MAF_KGH]*
And LV_ERR_LOAD_PLAUS = 0 *(MAP Plaus failure – 1st criteria: INSY) [MAF_KGH]*
And LV_ERR_AMP = 0 *(EL failure: INSY / to be used for CHRG version only) [PQ_CHA]*
And LV_ERR_AMP_PLAUS = 0
(Plaus failure: INSY / to be used for CHRG version only) [PQ_CHA]
And LV_ERR_PUT = 0 *(EL failure: CHRG)*
And LV_ERR_VS = 0 *(Global failure: Project specific) [VS]*
And LV_LIH_ERR_CRK = 0 *(Global failure_1st part: ENSD) [N]*
And LV_ERR_CAM = 0 *(Global failure_2nd part: ENSD) [N]*
And LV_ERR_TAM_CAN = 0 *(Global failure: input to AIRT/Def Project specific) [TAM_CAN]*
And LV_ERR_EL_TAM = 0
And LV_ERR_TCO = 0 *(Global failure: ENTE) [TCO]*
And all LV_ERR_EL_TIA[m] = 0 [NC_SENS_NR_TIA values] *(EL failure: AIRT) [TIA]*
And all LV_ERR_INTM_TIA[m]= 0 [NC_SENS_NR_TIA values] *(INTM failure: AIRT) [TIA]*
And all LV_ERR_DYN_TIA[m]= 0 [NC_SENS_NR_TIA values] *(DYN failure: AIRT) [TIA]*
And all LV_ERR_PLAUS_TIA[m]= 0 [NC_SENS_NR_TIA values] *(PLAUS failure: AIRT) [TIA]*
Then LV_INH_DIAG_RBM_PLAUS_TAM = 0
Else LV_INH_DIAG_RBM_PLAUS_TAM = 1
Endif

If LC_INH_DIAG_PLAUS_TAM_AS_ACT = 1
Then LV_INH_DIAG_PLAUS_TAM = LC_INH_DIAG_PLAUS_TAM_AS
Else LV_INH_DIAG_PLAUS_TAM = LV_INH_DIAG_RBM_PLAUS_TAM
Endif

Configuration for diagnostic symptoms :

Diagnostic TAM	Symptom description	Symptom	Filter type
ambient temperature plausibility diagnosis	*PLAUS_H*	*SYM_0*	*NO*
	PLAUS_L	*SYM_1*	

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_INH_DIAG_PLAUS_TAM_AS	1	0..1H	0..1	1	-
LC_INH_DIAG_PLAUS_TAM value set via application system					
LC_INH_DIAG_PLAUS_TAM_AS_ACT	1	0..1H	0..1	1	-
LC_INH_DIAG_PLAUS_TAM_AS_ACT = 1: LC_INH_DIAG_PLAUS_TAM value is set via application system					
C_T_ES_PLAUS_TAM_MIN	1	0...FFFFH	0...65535	1	min
Minimum necessary engine shut off time duration to cool down the engine (cold start condition reached)					
C_TCO_DIF_PLAUS_TAM_MIN	1	0...FEH	-48...142.5	0.75	°C
Minimum TCO change to enable TAM plausibility diagnostic					
C_TCO_PLAUS_TAM_MAX	1	0...FEH	-48...142.5	0.75	°C
maximum TCO value to enable TAM plausibility diagnosis					
C_T_AST_MAX_PLAUS_TAM_T_ES	1	0...FFFFH	0...6535.5	0.1	s
Maximum time after start to set the irreversible condition LV_TEMP_COLD_PLAUS_TAM					

B.76.3 RBM interface for TAM plausibility diagnosis:

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
STATE_RBM_PLAUS_TAM	O/V	0 ... FFH	0 ... 255	1	-
Interface of TAM plausibility diagnosis after start monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
RATE_RBM_PLAUS_TAM	V	0...FFFFH	0....7,99572	0,000122	-
Actual RBM ratio of TAM plausibility diagnosis, for application purposes					

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the TAM plausibility after start monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_PLAUS_TAM data.

Within STATE_RBM_PLAUS_TAM, three different informations are defined:

_ Conditions for monitoring are met long enough to detect malfunction (bit 0)
(no intrusive operation, no short trip)

_ Monitor disabled because of system malfunction (bit 1)
(depending on failure status: pending)

_ Monitor individual RBM conditions encountered within this DC (bit 2)

Application conditions:

Initialisation :

at ECU reset :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_TAM = 0

at LV_DC 0 → 1 transition :
bit 0, bit 1 and bit 2 of STATE_RBM_PLAUS_TAM = 0

on failure memory reset :
bit 1 of STATE_RBM_PLAUS_TAM = 0

Recurrence: 1 s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

{ Global failures which aren't managed by Error Management shall not appear here, because their pending status doesn't exist }

LV_ERR_EL_MAP *(EL failure – MAP application: INSY) [MAF_KGH]*
LV_ERR_VCC_PVS_2 (supply voltage for MAP-sensor)
LV_ERR_VCC_SENS_SUB (supply voltage for PUT-Sensor)
LV_ERR_MAF *(EL failure – MAF application: INSY) [MAF_KGH]*
LV_ERR_LOAD_PLAUS *(MAP Plaus failure – 1st criteria / MAP application: INSY) [MAF_KGH]*
LV_ERR_MAP_PLAUS (*MAP Plaus failure – 2nd criteria / MAP application: INSY) [MAF_KGH]*
LV_ERR_AMP *(EL failure: INSY / to be used for CHRG version) [PQ_CHA]*
LV_ERR_AMP_PLAUS *(AMP Plaus failure – to be used for MAP application: INSY) [PQ_CHA]*
LV_ERR_EL_PUT *(EL failure: CHRG) [PQ_CHA]*
LV_ERR_PUT_PLAUS *(PUT Plaus failure - to be used for MAP application: INSY) [PQ_CHA]*
LV_ERR_VS (*Project specific – Global failure – pending status available) [VS]*
LV_ERR_REF_CRK_CAM_IN_i *(ENSD) [N]*
LV_ERR_PER_CAM_IN_i *(ENSD) [N]*
LV_ERR_SYN_CAM_IN_i *(ENSD) [N]*
LV_ERR_PLAUS_CAM_IN_i *(ENSD) [N]*
LV_ERR_SYN_CRK_CAM_IN_i *(ENSD) [N]*
LV_ERR_CRK_TOOTH *(ENSD) [N]*
LV_ERR_CRK_TOOTH_PER *(ENSD) [N]*
LV_ERR_CRK_SYN *(ENSD) [N]*
LV_ERR_CRK_PLAUS *(ENSD) [N]*
LV_ERR_CAN_ICL_TAM *(EL error: Project specific) [TAM_CAN]*
LV_ERR_CAN_BUS_OFF *(CAN comm. error) [TAM_CAN]*
LV_ERR_TCO_EL *(EL failure: ENTE) [TCO]*
LV_ERR_TCO_GRD *(GRD failure: ENTE) [TCO]*
LV_ERR_TCO_STUCK *(STUCK failure: ENTE) [TCO]*
LV_ERR_TCO_STUCK_H *(STUCK HIGH failure: ENTE) [TCO]*
LV_ERR_TCO_PLAUS *(PLAUS failure: ENTE) [TCO]*
LV_ERR_EL_TIA[NC_SENS_NR_TIA] *(EL failure: AIRT) [TIA]*
LV_ERR_INTM_TIA[NC_SENS_NR_TIA] *(INTM failure: AIRT) [TIA]*
LV_ERR_DYN_TIA[NC_SENS_NR_TIA] *(DYN failure: AIRT) [TIA]*
LV_ERR_PLAUS_TIA[NC_SENS_NR_TIA] *(PLAUS failure: AIRT) [TIA]*

If(1) { CPU optimization at LV_DC 0 → 1 transition }
CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1) **While** bit 1 of STATE_RBM_PLAUS_TAM = 0 **do**
with each TAM plausibility after start failure of the above list :
ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) **XX** has a pending status
Then(2) bit 1 of STATE_RBM_PLAUS_TAM = 1
Endif(2)

Endwhile

Else(1) { the dynamic failure memory is empty }
No action

Endif(1)

Every 1 s :

- ***STATE_RBM determination:***

For m = 1 to NC_SENS_NR_TIA

```
If      bit 0 of STATE_RBM_PLAUS_TAM = 0
Then    If LV_END_DIAG_PLAUS_TAM = 1
        Then   bit 0 of STATE_RBM_PLAUS_TAM = 1
        Endif
Endif

If      bit 1 of STATE_RBM_PLAUS_TAM = 0
Then    If LV_INH_DIAG_RBM_PLAUS_TAM = 1
        { Cross dependency of the monitor defined by the project }
        { List here failures that can inhibit TIA plausibility monitor }
        Then bit 1 of STATE_RBM_PLAUS_TAM = 1
        Endif
Endif
```

bit 2 of STATE_RBM_PLAUS_TAM = 1

- ***RATE_RBM determination (to easier the calibration):***

RATE_RBM_PLAUS_TAM
= CTR_COMP_RBM_PLAUS_TAM / CTR_CDN_RBM_PLAUS_TAM

End for m

B.77 MAF deviation monitoring

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_ERR_MAF_DIF_CH_INT	V/O	0...1H	0...1	1	-
Boolean for failure detection due to mass air flow deviation integral is out of range					
ERR_SYM_MAF_DIF_CH_INT	V	0H 8H	NO_SYM SYM_3	1	-
Symptom for MAF deviation monitoring error					
LV_CDN_DIAG_MAF_DIF_CH_INT	V	0...1H	0...1	1	-
Boolean for mass air flow deviation monitoring is enabled					
LV_CNL_DIAG_MAF_DIF_CH_INT	V	0...1H	0...1	1	-
Boolean for mass air flow deviation monitoring is canceled					
LV_END_DIAG_MAF_DIF_CH_INT	V/O	0...1H	0...1	1	-
Boolean for mass air flow deviation monitoring is finished					
MAF_DIF_DIAG_CH	V	8000...7FFFH	-694.5...694.5	0.0212	mg/stk
Mass air flow deviations during catalyst heating phase					
MAF_DIF_DIAG_CH_INT	V	0... 3E7FC18H	0... 1389000	0.0212	mg
Mass air flow deviations integral during catalyst heating phase					
MAF_SP_GRD_DIAG_CH	V	8000...7FFFH	-694.5...694.5	0.0212	mg/stk²
Gradient of mass air flow setpoint during catalyst heating phase					
NR_SEG_CTR_MAF_DIF_DIAG_CH_1	V	0...FFFFH	0...65535	1	-
Number of counted segments while the mass air flow setpoint gradient must be below a threshold					
NR_SEG_CTR_MAF_DIF_DIAG_CH_2	V	0...FFFFH	0...65535	1	-
Number of valid segments after start for the calculation of the mass air flow deviation integral					

Input data:

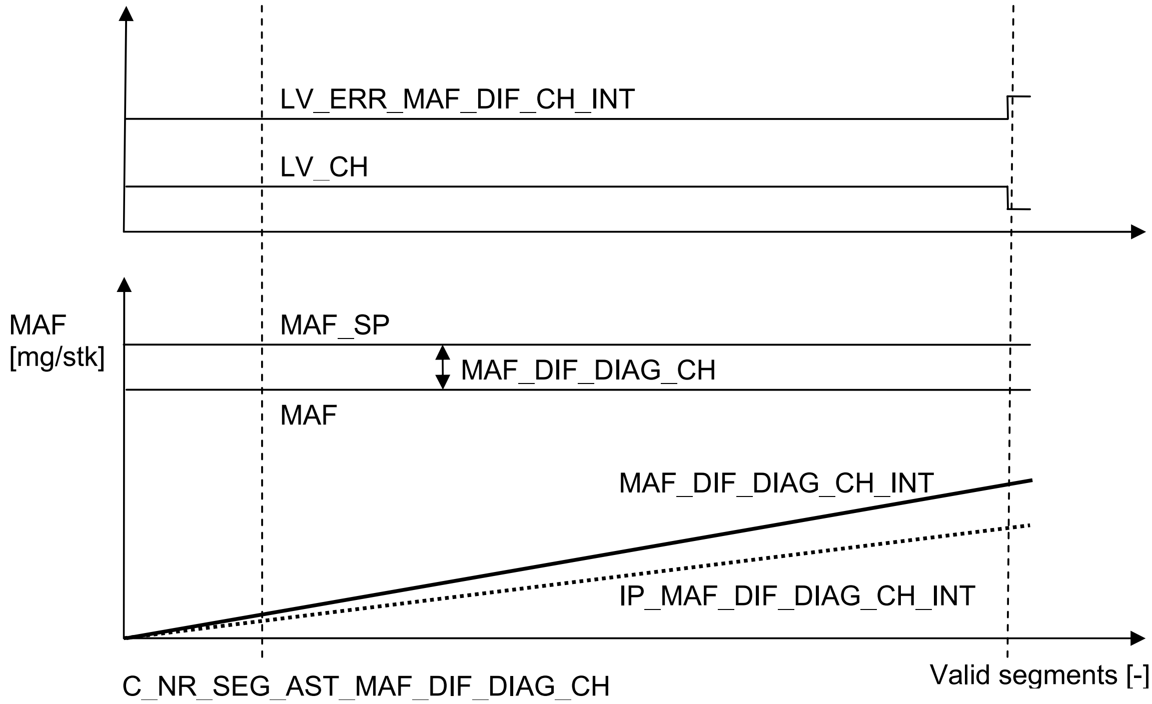
LV_ST_END	LV_CH	LV_INH_DIAG_MAF_DIF_CH_INT	MAF
MAF_SP			

FUNCTION DESCRIPTION:

General information:

This diagnostic function has been developed due to cold start strategy monitoring (CSSM) requirements.

The purpose of this monitoring is to detect MAF_SP – MAF deviations during the catalyst heating phase, as these deviations could lead to a retarded catalyst light off. Positive differences between MAF_SP and MAF are integrated, if the MAF_SP_GRD_DIAG_CH is below the threshold C_MAF_SP_GRD_DIAG_CH for a certain amount of segments C_NR_SEG_CTR_MAF_DIF_DIAG_CH. If NR_SEG_CTR_MAF_DIF_DIAG_CH_2 > C_NR_SEG_AST_MAF_DIF_DIAG_CH, then the integrated value MAF_DIF_DIAG_CH_INT is compared with IP_MAF_DIF_DIAG_CH_INT, but only at the transition LV_CH = 1 → 0. If the value MAF_DIF_DIAG_CH_INT is greater than IP_MAF_DIF_DIAG_CH_INT at the end of the catalyst heating phase, the error LV_ERR_MAF_DIF_CH_INT is detected.



General information:

Within this chapter the monitoring values for the actual diagnostic function are calculated.

Application conditions:

Initialisation at reset or LV_ST_END = 0 → 1:

LV_ERR_MAF_DIF_CH_INT = 0
LV_CDN_DIAG_MAF_DIF_CH_INT = 0
LV_CNL_DIAG_MAF_DIF_CH_INT = 0
ERR_SYM_MAF_DIF_CH_INT = "NO_SYM"
MAF_DIF_DIAG_CH = 0
MAF_DIF_DIAG_CH_INT = 0
MAF_SP_GRD_DIAG_CH = 0
NR_SEG_CTR_MAF_DIF_DIAG_CH_1 = 0
NR_SEG_CTR_MAF_DIF_DIAG_CH_2 = 0

// LV_END_DIAG_MAF_DIF_CH_INT is directly initialized by ERRM

Recurrence: every segment TDC

Activation: LV_ST_END = 1 **and**
LV_INH_DIAG_MAF_DIF_CH_INT = 0 **and**

LV_END_DIAG_MAF_DIF_CH_INT = 0 **and**

LV_CNL_DIAG_MAF_DIF_CH_INT = 0

Formula section:

IF(1) LV_CH = 1

THEN(1) LV_CDN_DIAG_MAF_DIF_CH_INT = 1

ERR_SYM_MAF_DIF_CH_INT = "NO_SYM"

(calculation of formula section **A** and **B**)

Mass air flow deviation monitoring values calculation:

A Mass air flow setpoint gradient calculation:

MAF_SP_GRD_DIAG_CH = (MAF_SP$_{(n)}$ - MAF_SP$_{(n-4)}$) / 4

B Mass air flow deviation integral calculation:

If(2b) MAF_SP_GRD_DIAG_CH =< C_MAF_SP_GRD_DIAG_CH

Then(2) NR_SEG_CTR_MAF_DIF_DIAG_CH_1$_{(n)}$ = NR_SEG_CTR_MAF_DIF_DIAG_CH_1$_{(n-1)}$ + 1

// Number of segments counter only runs until threshold is reached

If(3b) NR_SEG_CTR_MAF_DIF_DIAG_CH_1 >= C_NR_SEG_CTR_MAF_DIF_DIAG_CH

Then(3b) MAF_DIF_DIAG_CH = (MAF_SP – MAF)

If(4b) MAF_DIF_DIAG_CH > 0

Then(4b) MAF_DIF_DIAG_CH_INT$_{(n)}$ = MAF_DIF_DIAG_CH_INT$_{(n-1)}$ + MAF_DIF_DIAG_CH

NR_SEG_CTR_MAF_DIF_DIAG_CH_2$_{(n)}$ = NR_SEG_CTR_MAF_DIF_DIAG_CH_2$_{(n-1)}$ + 1

Else(3b) *//wait*

Endif(3b)

Else(2b) NR_SEG_CTR_MAF_DIF_DIAG_CH_1 = 0

Endif(2b)

ELSE(1)

If(5) LV_CH = 1 → 0

Then(5) (calculation of formula section **C**)

Mass air flow deviation monitoring error calculation:

C Mass air flow deviation integral diagnostic check:

If(6c) NR_SEG_CTR_MAF_DIF_DIAG_CH_2 > C_NR_SEG_AST_MAF_DIF_DIAG_CH

Then(6c) LV_CDN_DIAG_MAF_DIF_CH_INT = 1

If(7c) MAF_DIF_DIAG_CH_INT > IP_MAF_DIF_DIAG_CH_INT (NR_SEG_CTR_MAF_DIF_DIAG_CH_2)

Then(7c) LV_ERR_MAF_DIF_CH_INT = 1

ERR_SYM_MAF_DIF_CH_INT = "SYM_3"

LV_END_DIAG_MAF_DIF_CH_INT = 1

Else(7c) LV_ERR_MAF_DIF_CH_INT = 0

ERR_SYM_MAF_DIF_CH_INT = "NO_SYM"

LV_END_DIAG_MAF_DIF_CH_INT = 1

Endif(7c)

Else(6c) LV_CDN_DIAG_MAF_DIF_CH_INT = 0

LV_CNL_DIAG_MAF_DIF_CH_INT = 1

Endif(6c)

Else(5) LV_CDN_DIAG_MAF_DIF_CH_INT = 0

Endif(5)

ENDIF(1)

Configuration for diagnostic symptoms:

Diagnostic MAF_DIF_CH_INT	Symptom description	Symptom	Filter type
MAF deviation monitoring			*NO*
	MAF_SP - MAF deviation	*SYM_3*	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_DIF_DIAG_CH_INT	1*8	0...FFFFH	0... 1389000	21.1984	mg
LDP_NR_SEG_IP_MAF_DIF_DIAG_CH	8	0...FFFFH	0...65535	1	-
Table for integrated MAF deviations during number of valid segments					
C_MAF_SP_GRD_DIAG_CH	1	0...FFFFH	0...1389	0.0212	mg/stk²
Constant for treshold of mass air flow setpoint gradient					
C_NR_SEG_AST_MAF_DIF_DIAG_CH	1	0...FFFFH	0...65535	1	-
Constant for number of valid segments after start to begin with the monitoring					
C_NR_SEG_CTR_MAF_DIF_DIAG_CH	1	0...FFFFH	0...65535	1	-
Constant for segment counter while the mass air flow setpoint gradient must be below a threshold					

B.78 MAF deviation monitoring (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_INH_DIAG_MAF_DIF_CH_INT	V/O	0...1H	0...1	1	-
Boolean for mass air flow deviation monitoring is inhibited					

Input data:

LV_ST_END	LV_ERR_MAF	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAP
LV_ERR_TPS	LV_ERR_EL_CPS	FAC_FQ_ST_AD	TI_IS_MMV_MAX

FUNCTION DESCRIPTION:

General information:

The mass air flow deviation monitoring is performed, if the logical value LV_ST_END = 1 and LV_CH = 1. If the engine stalls or one of the conditions for inhibiting the diagnosis is present, it will be stopped and will start again, when conditions above are met.

The diagnosis is finished as soon as the boolean for end of mass air flow monitoring is set to “1”. In this case the boolean for mass air flow monitoring indicates either a faulty (= 1) or a normal (= 0) working air path.

Depending on projects specific requirements, the MAF deviation monitoring can be inhibited by setting LV_INH_DIAG_MAF_DIF_CH_INT = 1.

Application conditions:

Initialisation at reset or LV_ST_END = 0 ->1:

LV_INH_DIAG_MAF_DIF_CH_INT = 0

Recurrence: every segment TDC

Activation: LV_ST_END = 1

Deactivation: LV_ST_END = 0

Formula section:

Calculation of diagnosis interface parameter:

Note: **the projects have to adapt their requirements to the interface parameter !**

Default:

If LV_ERR_MAF = 1 **or** *{electrical load sensor error}*

LV_ERR_MAP = 1 **or** *{electrical load sensor error}*

LV_ERR_LOAD_TPS_PLAUS = 1 **or** *{plausibility error load or TPS sensor}*

LV_ERR_TPS = 1 **or** *{global TPS error}*

LV_ERR_EL_CPS = 1 **or** ***{electrical CPS error}***

FAC_FQ_ST_AD > C_FAC_FQ_ST_AD_MIN_CSSM or

{high fuel quality adaptation result}

TI_IS_MMV_MAX > C_TI_IS_MMV_MAX_MIN_CSSM

{high idle fuel quality adaptation result}

then LV_INH_DIAG_MAF_DIF_CH_INT = 1

endif

B.79 Interface to RBM - MAF deviation monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_RBM_MAF_DIF_CH_INT	O/V	0...FFH	0...255	1	-
Interface of MAF deviation monitor with the Rate-Based Monitoring statistics Bit 0: conditions for monitoring are met long enough to detect malfunction (bit 0 = 1) Bit 1: inhibition of the monitor because of system failure(s) (bit 1 = 1) Bit 2: individual RBM conditions of the monitor were encountered within this DC (bit 2 = 1)					
T_MAF_DIF_DIAG_CH	V	0...FFH	0...255	1	s
Time while catalyst heating was active during current driving cycle					

Input data:

LV_DC	CTR_ERR_DYN_NR	LV_END_DIAG_MAF_DIF_CH_INT	LV_INH_DIAG_MAF_DIF_CH_INT
LV_CH	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2

Import actions:

ACTION_ERRM_CheckPendingStatus (IN <XX>, OUT<PendingStatus>)

FUNCTION DESCRIPTION:

General information:

With this module the interface between the MAF deviation monitor and the Rate-Based Monitoring statistics is defined with STATE_RBM_MAF_DIF_CH_INT data.

Within STATE_RBM_MAF_DIF_CH_INT, three different information are defined:

- Conditions for monitoring are met long enough to detect malfunction (bit 0)
  (no intrusive operation, no short trip)
- Monitor disabled because of system malfunction (bit 1)
  (depending on failure status: pending)
- Monitor individual RBM conditions encountered within this DC (bit 2)
  (only if catalyst heating is active more than 10s)

Application conditions:

Initialisation:

at ECU reset:

bit 0, bit 1 and bit 2 of STATE_RBM_MAF_DIF_CH_INT = 0

at LV_DC 0 → 1 transition:

bit 0, bit 1 and bit 2 of STATE_RBM_MAF_DIF_CH_INT = 0

on failure memory reset:

bit 1 of STATE_RBM_MAF_DIF_CH_INT = 0

at reset or LV_ST_END = 0 → 1:

T_MAF_DIF_DIAG_CH = 0s

Recurrence: 1s

Activation: LV_DC 0 → 1 transition **and** LV_DC = 1

Formula section:

At LV_DC 0 → 1 transition

The pending status of the following failures has to be checked only once :

For ETC system (NC_ETC_CONF = 1) :

LV_ERR_MAF_SCG_OC	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAF_SCP	LV_ERR_TPS_1
LV_ERR_TPS_2	LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2	LV_ERR_TPS_VCC
LV_ERR_TPS_AD	LV_ERR_TPS_AD_SPR	LV_ERR_TPS_ST_CHK	LV_ERR_MTC_DR
LV_ERR_MTC_CTL	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2

For non-ETC-system (NC_ETC_CONF = 0) :

LV_ERR_MAF_SCG_OC	LV_ERR_LOAD_TPS_PLAUS	LV_ERR_MAF_SCP	LV_ERR_TPS_EL
LV_ERR_VCC_TPS	LV_ERR_EL_MAP	LV_ERR_MAP_PLAUS	LV_ERR_VCC_PVS_2

If(1) { CPU optimization at LV_DC 0 → 1 transition }

CTR_ERR_DYN_NR <> 0 { the dynamic failure memory isn't empty }

Then(1)

While bit 1 of STATE_RBM_MAF_DIF_CH_INT = 0 **do**

with each XX failure of the above list :

ACTION_ERRM_CheckPendingStatus(IN<XX>, OUT<PendingStatus>, SYNCHRONIZATION<CALL>)

If(2) XX has a pending status

Then(2)

bit 1 of STATE_RBM_MAF_DIF_CH_INT = 1

Endif(2)

Endwhile

Else(1)

{ the dynamic failure memory is empty }

No action

Endif(1)

Every 1 s:

If LV_CH = 1

Then T_MAF_DIF_DIAG_CH$_{(n)}$ = T_MAF_DIF_DIAG_CH$_{(n-1)}$ +1s

Else T_MAF_DIF_DIAG_CH$_{(n)}$ = 0s

Endif

If bit 0 of STATE_RBM_MAF_DIF_CH_INT = 0

Then

If LV_END_DIAG_ MAF_DIF_CH_INT = 1

Then bit 0 of STATE_RBM_MAF_DIF_CH_INT = 1

Endif

Endif

If bit 1 of STATE_RBM_MAF_DIF_CH_INT = 0

Then

If LV_INH_DIAG_MAF_DIF_CH_INT = 1

Then bit 1 of STATE_RBM_MAF_DIF_CH_INT = 1

Endif

Endif

If bit 2 of STATE_RBM_MAF_DIF_CH_INT = 0

Then

If T_MAF_DIF_DIAG_CH >= C_T_MAF_DIF_DIAG_CH

Then bit 2 of STATE_RBM_MAF_DIF_CH_INT = 1

Endif

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_MAF_DIF_DIAG_CH	1	0...FFH	0...255	1	s
Constant for time while catalyst heating was active during current driving cycle					
C_FAC_FQ_ST_AD_MIN_CSSM	1	0...FFH	0...1.992	0.0078	-
Constant for inhibition of CSSM due to fuel quality adaptation					
C_TI_IS_MMV_MAX_MIN_CSSM	1	80...7FH	-0.5...0.496	1/256	-
Constant for inhibition of CSSM due to ambient pressure					

B.80 Engine off timer plausibility diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_T_ES_PLAUS	V/O	0...1H	0...1	1	[-]
Error flag for T_ES_PLAUS diagnosis					
LV_CDN_DIAG_T_ES_PLAUS	V	0...1H	0...1	1	[-]
Activation or deactivation of the T_ES_PLAUS diagnosis					
ERR_SYM_ T_ES_PLAUS	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Detected error symptom T_ES_PLAUS diagnosis					
LV_END_DIAG_T_ES_PLAUS	V	0...1H	0...1	1	[-]
T_ES_PLAUS diagnosis done completely at least one time					
T_ES_DIF	V/O	0...FFH	0...255	1	Sec
Deviation between external Engine off timer and internal uC timer					
LV_DET_ERU_ES	V	0...1H	0...1	1	[-]
Flag indicating transition engine run to engine off was detected					
T_ES_DEC_CTR	V	0...FFH	0...255	1	-
Decrement counter before Engine off timer reset and activation check is performed					

Input data:

LV_ES	VB	T_ES_DIAG	LV_T_ES_ACT
T_ES_IGK	T_IGK_ON		

B.80.1 Engine Off Timer Runtime Error

Description:

The Diagnosis of the Engine Off Time is realised by comparing the external Engine Off timer value calculated in separate micro controller with an ECU internal timer that counts the time after Ignition key on.
Aditionally it is checked, that T_ES counter is resetted and activated again correct after Engine stop is detected. Therefore the Engine off timer value have to be inside a window after transition from running to stopped engine

Error-symtoms and conditions are defined to this diagnosis function as following:

```
□□□□
│ │ │ └─> -   Engine off timer not resetted or activated   (= SYM_0)
│ │ └───> -   (not used)                                   (= SYM_1)
│ └─────> -   (not used)                                   (= SYM_2)
└───────> -   Engine off timer inaccurate                  (= SYM_3)
```

Application conditions:

Initialisation:

at reset: LV_DET_ERU_ES = 0

at transition LV_ES = 0 --> 1:

T_ES_DEC_CTR = C_T_ES_DEC_CTR

LV_DET_ERU_ES = 1

according filter type: "STD"

Recurrence: 1000ms

Activation:

if	LC_ENA_T_ES_PLAUS_DIAG = 1	*diagnosis is enabled*
and	T_ES_DIAG < FFFFFFH	*engine off time below maximum*
and	C_VB_MIN_T_ES < VB <= C_VB_MAX_T_ES	*VB inside uC Operation range*
and	LV_T_ES_ACT = 1	*Engine off timer already activated*
then	LV_CDN_DIAG_T_ES_PLAUS = 1	*condition for DIAG fulfilled*
else	LV_CDN_DIAG_T_ES_PLAUS = 0	
endif		

Formula Section:

Accuracy check

T_ES_DIF = abs [T_ES_DIAG – T_ES_IGK - T_IGK_ON]

if LV_IGK = 1

and C_T_IGK_ON_MIN < T_IGK_ON < FFFFH

then **if** T_ES_DIF < C_T_DIF_MAX_T_ES_PLAUS + C_FAC_MAX_T_ES_PLAUS * T_IGK_ON

Comparsion between value of external uC ES timer and Main Controller value:
After IGK on, the difference has to be smaller than threshold

then ERR_SYM_T_ES_PLAUS = 0h *(NO_SYM)*

LV_ERR_T_ES_PLAUS = 0

else ERR_SYM_T_ES_PLAUS = 8h *(SYM_3)*

LV_ERR_T_ES_PLAUS = 1 *(without debounce)*

LV_END_DIAG_T_ES_PLAUS = 1

endif

Reset and Activation check:

If LV_DET_ERU_ES = 1
and T_ES_DEC_CTR = 0
then if C_T_ES_DIAG_MAX>= T_ES_DIAG >= C_T_ES_DIAG_MIN
then ERR_SYM_T_ES_PLAUS = 0h *(NO_SYM)*
LV_ERR_T_ES_PLAUS = 0
LV_DET_ERU_ES = 0
LV_END_DIAG_T_ES_PLAUS = 1
else ERR_SYM_T_ES_PLAUS = 1h *(SYM_0)*
LV_ERR_T_ES_PLAUS = 1
LV_DET_ERU_ES = 0
LV_END_DIAG_T_ES_PLAUS = 1
endif
endif

$T\_ES\_DEC\_CTR_n = T\_ES\_DEC\_CTR_{n-1} - 1$

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_DIF_MAX_T_ES_PLAUS	1	0...FFH	0...255	1	[sec]
Difference between two timers because of possible incurrancy					
C_FAC_MAX_T_ES_PLAUS	1	0...FFH	0...0,255	0.001	-
Accuracy factor for engine off timer					
C_T_ES_DIAG_MIN	1	0...FFH	0...255	1	[sec]
Minimum value for Engine off timer reset and activation					
C_T_ES_DIAG_MAX	1	0...FFH	0...255	1	[sec]
Maximum value for Engine off timer reset and activation					
C_T_ES_DEC_CTR	1	0...FFH	0...255	1	[-]
Max decrement counter before Engine off timer reset and activation check is performed					
LC_ENA_T_ES_PLAUS_DIAG	1	0...1H	0...1	1	-
Logical constant to enable T_ES_PLAUS diagnosis					
C_T_IGK_ON_MIN	1	0...FFH	0...255	1	[sec]
Minimum Time after Ignition on to perform Accuracy check					
C_VB_MIN_T_ES	1	0...FFH	0...26	0.102	V
Minimum Threshold for VB for Engine off timer Controller					
C_VB_MAX_T_ES	1	0...FFH	0...26	0.102	V
Maximum Threshold for VB for Engine off timer Controller					

B.81 Engine off timer plausibility diagnosis (Appl. Inc.)

Configuration for diagnostic symptoms :

Diagnostic T_ES_PLAUS	**Symptom description**	**Symptom**	**Filter type**
Detected error symptom T_ES_PLAUS diagnosis	*engine off timer not resetted or activated*	*SYM_0*	*(STD)*
	not used	*SYM_1*	
	not used	*SYM_2*	
	engine off timer value inaccurate	*SYM_3*	

B.82 RCL Plausibility Actuator Diagnosis

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PLAUS_CLOSE_RCL	O/V	0...1H	0...1	1	-
Present RCL plausibility - stuck close - failure					
LV_END_DIAG_PLAUS_CLOSE_RCL	O/V	0...1H	0...1	1	-
Result of RCL plausibility - stuck close - failure					
LV_ERR_PLAUS_OPEN_RCL	O/V	0...1H	0...1	1	-
Present RCL plausibility - stuck open - failure					
LV_END_DIAG_PLAUS_OPEN_RCL	O/V	0...1H	0...1	1	-
Result of RCL plausibility - stuck open - failure					
ERR_SYM_PLAUS_CLOSE_RCL	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of RCL plausibility - stuck close - detected failure					
PUT_RCL_DIF_ABSV	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Absolute PUT variation					
T_PLAUS_CLOSE_RCL	V	0...FFH	0...2.55E+3	10	ms
Time elapsed since last LV_RCL_CLOSE_CLOSE_GRD activation (plausibility - stuck close)					
PUT_RCL_MMV	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
PUT filtered value for RCL plausibility _ stuck close -diagnosis					
LV_PUT_RCL_INT	V	0...1H	0...1	1	-
Integration of PUT_RCL_INT possible (=1) / not possible (=0)					
PUT_RCL_INT	V	0...FFFFH	0...54.34	8.29175E- 4	hPa*s
Integral value of PUT_RCL_DIF_ABSV					
PUT_RCL_INT_THD_CLOSE	V	0...FFFFH	0...54.34	8.29175E- 4	hPa*s
Threshold PUT_RCL_INT value to set ERR_SYM_PLAUS_CLOSE_RCL to "STUCK CLOSE"					
PV_AV_GRD_RCL_PLAUS_MIN	V	80...7FH	-2.5E+3 ... 2.48047E+3	19.53125	%/s
Pedal gradient value used for RCL plausibility - stuck close - detected failure					
PUT_RCL_INT_CAL_MIN	V	0...FFFFH	0...54.34	8.29175E- 4	hPa*s
Minimum PUT_RCL_INT value recorded within the driving cycle - used only for calibration ease					
PUT_RCL_INT_THD_OPEN	V	0...FFFFH	0...54.34	8.29175E- 4	hPa*s
Threshold PUT_RCL_INT value to set ERR_SYM_PLAUS_CLOSE_RCL to "NO SYM"					
PSN_RCL_SP_REF	V	0...FFFFH	0...99.9984741	0.0015258 8	%
Reference RCL setpoint value for Plausibility - stuck close -diagnosis					
PUT_RCL_INT_CAL_MAX	V	0...FFFFH	0...54.34	8.29175E- 4	hPa*s
Maximum PUT_RCL_INT value recorded within the driving cycle - used only for calibration ease					
ERR_SYM_PLAUS_OPEN_RCL	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	-
Symptom of RCL plausibility - stuck OPEN - detected failure					
LV_CDN_DIAG_PLAUS_OPEN_RCL	V	0...1H	0...1	1	-
Conditions for RCL actuator plausibility - stuck open - diagnosis					
LOAD_WG_PLAUS	V	0...FFFFH	0...99.9984741	0.0015258 8	%
Wastegate load parameter for RCL actuator plausibility - stuck open - diagnosis					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CDN_DIAG_PLAUS_OPEN_RCL_PREL	V	0...1H	0...1	1	-
Preliminary conditions for RCL actuator plausibility - stuck open - diagnosis					
PUT_MDL_RCL_CLOSE_BAS	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Raw value of PUT_MDL_RCL_CLOSE for RCL actuator plausibility - stuck open - diagnosis					
PUT_MDL_RCL_CLOSE	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
PUT model when RCL valve is closed for RCL actuator plausibility - stuck open - diagnosis					
PUT_MDL_RCL_OPEN_BAS	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Raw value of PUT_MDL_RCL_OPEN for RCL actuator plausibility - stuck open - diagnosis					
LV_CDN_DIAG_PLAUS_CLOSE_RCL	V	0...1H	0...1	1	-
Conditions for RCL actuator plausibility - stuck close - failure					
PUT_MDL_RCL_OPEN	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
PUT model when RCL valve is opened for RCL actuator plausibility - stuck open - diagnosis					
PUT_MDL_RCL_CLOSE_VAR	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.0829175 2	hPa
PUT_MDL_RCL_CLOSE variation value for RCL plausibility - stuck open - diagnosis					
PUT_MDL_RCL_OPEN_VAR	V	8000...7FFFH	-2.717E+3 ... 2.71696E+3	0.0829175 2	hPa
PUT_MDL_RCL_OPEN variation value for RCL plausibility - stuck open - diagnosis					
T_PLAUS_OPEN_RCL	V	0...1FE0H	0...8.16E+4	10	ms
Time elapsed since last LV_PUT_RCL_STAB activation for RCL plausibility - stuck open - diagnosis					
LV_PUT_RCL_GAP_CLOSE_OPEN	V	0...1H	0...1	1	-
Sufficient pres.gap between PUT_MDL_RCL_CLOSE and OPEN for plausibility - stuck open - diag symptom det.					
LV_PUT_RCL_STAB	V	0...1H	0...1	1	-
RCL plausibility diagnosis - stuck open - stability criterium: (stability criterium fullfilled if =1)					
PUT_RCL_STAB	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Filtered value of PUT_MMV used to define RCL plausibility diagnosis stability criterium					
LV_DIAG_PLAUS_RCL_PREL	V	0...1H	0...1	1	-
Common preliminary conditions for all plausibility diagnosis					
LV_GR_RCL	V	0...1H	0...1	1	-
Gear shift detection flag for RCL actuator plausibility - stuck close - diagnosis					
LV_DIAG_PLAUS_CLOSE_RCL_PREL	V	0...1H	0...1	1	-
Conditions for RCL actuator plausibility - stuck close - diagnosis					
PQ_CHA_RCL_REF	V	0...FFFFH	0...15.9997559	2.44141E-4	-
Charge pressure quotient reference value for RCL plausibility - stuck close -diagnosis					
T_GR_RCL	V	0...FFH	0...2.55E+3	10	ms
Timer for gear shift detection management for RCL actuator plausibility - stuck close - diagnosis					
PUT_RCL_DIF_ABSV_MAX	V	0...FFFFH	0...5.434E+3	0.0829175 2	hPa
Maximum absolute PUT variation during counter incrementation					

Input data:

LC_TCHA_CONF	C_ABC_INC_PLAUS_CLOSE_RCL	C_ABC_MAX_PLAUS_CLOSE_RCL	C_ABC_INC_PLAUS_OPEN_RCL
C_ABC_MAX_PLAUS_OPEN_RCL	LV_INH_DIAG_PLAUS_OPEN_RCL	PSN_WG	TAM
LV_INH_DIAG_PLAUS_CLOSE_RCL	PSN_RCL_SP	TPS	LV_CDN_VB_OBD2
AMP	LV_IGK	PSN_RCL_MDL_MV	PUT_MES
LV_RCL_CLOSE_GRD	PRS_CHA_UP_MV	PWM_WG	PSN_WG_SP
GEAR	PUT_MMV	TCO	PV_AV_GRD_RCL
PUT_SP	LV_IN_PROT	N_32	PQ_CHA_MV
LV_TCHA_PROT_ACT	VS	LV_SURGE	LV_SURGE_PRED
LV_TQ_RCL_INTV	NC_IDX_DIAG_PLAUS_CL	NC_IDX_DIAG_PLAUS_O	ERR_SYM_EL_RCL_ACR

	OSE_RCL	PEN_RCL	
ERR_SYM_PUT_PLAUS_FL	ERR_SYM_PUT_PLAUS_ES	ERR_SYM_PUT	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AMP_RCL_MIN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum AMP threshold value for RCL plausibility diagnosis activation					
C_CRLC_PUT_RCL	1	0...FFH	0...0.99609375	0.00390625	-
Filter factor for PUT_RCL_MMV determination					
C_CRLC_PUT_RCL_STAB	1	0...FFFFH	0...0.99998474	1.52588E-5	-
Filter factor for PUT_RCL_STAB determination					
C_GR_RCL_MIN	1	0...FFH	0...255	1	-
Minimum gear ratio threshold value for RCL plausibility diagnosis activation					
C_LOAD_WG_PLAUS_OPEN_RCL_MAX	1	0...FFFFH	0...99.9984741	0.00152588	%
Max. necessary LOAD_WG_PLAUS value to fulfill RCL plausibility - stucked open - diagnosis conditions					
C_LOAD_WG_PLAUS_OPEN_RCL_MIN	1	0...FFFFH	0...99.9984741	0.00152588	%
Min necessary LOAD_WG_PLAUS value to fulfill RCL plausibility - stucked open - diagnosis conditions					
C_LOAD_WG_PLAUS_SRC	1	1H 2H 3H	PWM_WG PSN_WG_SP PSN_WG	1	-
Wastegate load parameter definition for RCL Plausibility - stuked open - diagnosis					
C_N_PLAUS_OPEN_RCL_MIN	1	0...FFH	0...8.16E+3	32	rpm
Min necessary N_32 value to fulfill RCL plausibility - stucked open - diagnosis conditions					
C_PQ_CHA_RCL_MAX	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Maximum charge pressure ratio threshold value to initialize the timer T_PLAUS_CLOSE_RCL to 10ms					
C_PQ_CHA_RCL_MIN	1	0...FFFFH	0...15.9997559	2.44141E-4	-
Minimum charge pressure ratio threshold value to initialize the timer T_PLAUS_CLOSE_RCL to 10ms					
C_PUT_DIF_RCL_STAB_MAX	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum possible abs. press. diff. between PUT_RCL_STAB and PUT_MMV to set LV_PUT_RCL_STAB					
C_PUT_MDL_RCL_CLOSE_BAS_AS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
PUT_MDL_RCL_CLOSE_BAS fixed via Application system					
C_PUT_MDL_RCL_CLOSE_DIF_NEG	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Negative pressure offset on PUT model for RCL valve close detection					
C_PUT_MDL_RCL_OPEN_BAS_AS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
PUT_MDL_RCL_OPEN_BAS fixed via Application system					
C_PUT_MDL_RCL_OPEN_DIF_NEG	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Negative pressure offset on PUT model for RCL valve open detection					
C_PUT_MDL_RCL_OPEN_DIF_POS	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Positive pressure offset on PUT model for RCL valve open detection					
C_PUT_MDL_RCL_OPEN_DIF_SP	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Minimum difference between PUT_SP and PUT_MMV for RCL valve open detection					
C_PUT_PLAUS_OPEN_RCL_MIN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Min. necessary PUT_MMV value to fulfill RCL Plausibility - stuck open - diagnosis conditions					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PUT_RCL_DIF_ABSV_MAX_MAX	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum allowed PUT variation for RCL plausibility - stucked close - diagnosis					
C_PUT_RCL_DIF_ABSV_THD	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Calibration used to suppress incidence of small PUT_RCL_DIF_ABSV values on PUT_RCL_INT					
C_PUT_RCL_GAP_CLOSE_OPEN_MIN	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Min. necessary . press. gap. to fulfill RCL Plausibility - stuck open diagnosis conditions					
C_PV_AV_GRD_RCL_PLAUS_MIN_MAX	1	80...7FH	-2.5E+3 ... 2.48047E+3	19.53125	%/s
Maximum PV_AV_GRD_RCL threshold value to initialize the timer T_PLAUS_CLOSE_RCL to 10 ms					
C_TAM_RCL_MIN	1	0...FEH	-48...142.5	0.75	°C
Minimum TAM threshold value for RCL plausibility diagnosis activation					
C_TCO_PLAUS_OPEN_RCL_MAX	1	0...FEH	-48...142.5	0.75	°C
Max. necessary TCO value to fulfill RCL plausibility - stuck open - diagnosis conditions					
C_TCO_PLAUS_OPEN_RCL_MIN	1	0...FEH	-48...142.5	0.75	°C
Minimum necessary TCO value to fulfill RCL plausibility - stuck open - diagnosis conditions					
C_TPS_PLAUS_OPEN_RCL_MIN	1	0...FFH	0...119.040469	0.46682537	°TPS
Min necessary TPS value to fulfill RCL plausibility - stucked open - diagnosis conditions					
C_T_GR_RCL_MAX	1	0...FFH	0...2.55E+3	10	ms
Maximum value of timer T_GR_RCL_MAX for gear shift detection					
C_T_PLAUS_CLOSE_RCL_MAX	1	0...FFH	0...2.55E+3	10	ms
Maximum necessary value of the timer T_PLAUS_CLOSE_RCL to set LV_PUT_RCL_INT					
C_T_PLAUS_CLOSE_RCL_MIN	1	0...FFH	0...2.55E+3	10	ms
Minimum necessary value of the timer T_PLAUS_CLOSE_RCL to set LV_PUT_RCL_INT					
C_T_PLAUS_OPEN_RCL_MAX	1	0...1FE0H	0...8.16E+4	10	ms
Maximum possible value for the counter T_PLAUS_RCL_OPEN					
C_VS_PLAUS_CLOSE_RCL_MIN	1	0...FFH	0...255	1	km/h
Minimum VS threshold value for RCL plausibility - stucked close - diagnosis activation					
LC_END_DIAG_PLAUS_CLOSE_SUPP	1	0...1H	0...1	1	-
Suppress the effect of LV_END_DIAG_PLAUS_CLOSE_RCL on LV_CDN_DIAG_PLAUS_CLOSE_RCL					
LC_END_DIAG_PLAUS_OPEN_SUPP	1	0...1H	0...1	1	-
Suppress the effect of LV_END_DIAG_PLAUS_OPEN_RCL on LV_CDN_DIAG_PLAUS_OPEN_RCL					
LC_PSN_RCL_SP_REF	1	0...1H	0...1	1	-
Selection of reference RCL setpoint for RCL plausibility - stucked close - diagnosis					
LC_PUT_MDL_RCL_CLOSE_BAS_AS	1	0...1H	0...1	1	-
Application system switch value for PUT_MDL_RCL_CLOSE_BAS determination					
LC_PUT_MDL_RCL_OPEN_BAS_AS	1	0...1H	0...1	1	-
Application system switch value for PUT_MDL_RCL_OPEN_BAS determination					
IP_CRLC_PUT_MDL_RCL	6	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_PUT_MDL_RCL_VAR_IP_PUT_MDL	6	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
Filter factor for PUT_MDL_RCL_OPEN and PUT_MDL_RCL_CLOSE determination					
IP_PUT_MDL_RCL_COR	6	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDP_PRS_CHA_UP_IP_PUT_MDL_RCL	6	0...FFFFH	0...5.434E+3	0.08291752	hPa
Additive ambient pressure correction term common for PUT_MDL_RCL_CLOSE/OPEN model					
IP_PUT_RCL_INT_THD_CLOSE	4	0...FFFFH	0...54.34	8.29175E-4	hPa*s
LDPM_PV_AV_GRD_1_CHRG	4	0...FFH	-2.5E+3 ... 2.48047E+3	19.53125	%/s
Threshold PUT_RCL_INT value to set ERR_SYM_PLAUS_CLOSE_RCL TO "STUCK_CLOSE"					
IP_PUT_RCL_INT_THD_OPEN	4	0...FFFFH	0...54.34	8.29175E-4	hPa*s
LDPM_PV_AV_GRD_1_CHRG	4	0...FFH	-2.5E+3 ... 2.48047E+3	19.53125	%/s
Threshold PUT_RCL_INT value to set ERR_SYM_PLAUS_CLOSE_RCL TO "NO_SYM"					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_PUT_MDL_RCL_CLOSE_OFS_1	14x10	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_TPS_1_CHRG	14	0...FFH	0...119.040469	0.46682537	°TPS
LDPM_N_32_3_CHRG	10	0...FFH	0...8.16E+3	32	rpm
1st offset term for PUT_MDL_RCL_CLOSE model					
IP_PUT_MDL_RCL_CLOSE_OFS_2	14x14	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_TPS_1_CHRG	14	0...FFH	0...119.040469	0.46682537	°TPS
LDPM_LOAD_WG_PLAUS_2_CHRG	14	0...FFFFH	0...99.9984741	0.00152588	%
2nd offset term for PUT_MDL_RCL_CLOSE model					
IP_PUT_MDL_RCL_CLOSE_OFS_3	10x14	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_N_32_3_CHRG	10	0...FFH	0...8.16E+3	32	rpm
LDPM_LOAD_WG_PLAUS_2_CHRG	14	0...FFFFH	0...99.9984741	0.00152588	%
3rd offset term for PUT_MDL_RCL_CLOSE model					
IP_PUT_MDL_RCL_OPEN_OFS_1	14x10	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_TPS_1_CHRG	14	0...FFH	0...119.040469	0.46682537	°TPS
LDPM_N_32_3_CHRG	10	0...FFH	0...8.16E+3	32	rpm
1st offset term for PUT_MDL_RCL_OPEN model					
IP_PUT_MDL_RCL_OPEN_OFS_2	14x14	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_TPS_1_CHRG	14	0...FFH	0...119.040469	0.46682537	°TPS
LDPM_LOAD_WG_PLAUS_2_CHRG	14	0...FFFFH	0...99.9984741	0.00152588	%
2nd offset term for PUT_MDL_RCL_OPEN model					
IP_PUT_MDL_RCL_OPEN_OFS_3	10x14	0...FFFFH	-2.717E+3 ... 2.71696E+3	0.08291752	hPa
LDPM_N_32_3_CHRG	10	0...FFH	0...8.16E+3	32	rpm
LDPM_LOAD_WG_PLAUS_2_CHRG	14	0...FFFFH	0...99.9984741	0.00152588	%
3rd offset term for PUT_MDL_RCL_OPEN model					

Import actions:

ACTION_ERRM_FilterSymptom(IN <XX>, IN <lv_cdn_diag_XX>, IN <err_sym_XX>, IN <c_abc_inc_XX>, IN <c_abc_dec_XX>, IN <c_abc_max_XX>, OUT <lv_err_XX>)
This action computes the elementary anti-bounce filter for one failure treatment and returns filter result
ACTION_ERRM_GetLvEndDiag(IN <XX>, OUT <lv_end_diag_XX>)
This action returns the end-of-diagnosis flag

B.82.1 General information

Stuck close diagnosis:

As long as the RCL EPC is deactivated, the RCL valve is closed (the spring force pushes the membrane). The activation of the RCL valve actuator must lead to a smooth decrease of the pressure upstream throttle (PUT_MES). If it is not verified (strong PUT oscillations noticed), we conclude that the valve remains stuck in closed position).

Stuck open diagnosis:

In part load, the RCL valve is not activated (except in case of necessity of manifold protection from overpressure) and the pressure upstream throttle reaches high values in charger mode.

If the pressure upstream throttle is not sufficient then we conclude that the RCL valve is stuck in fully open position.

This diagnosis can detect an RCL valve stuck in fully open position or any **equivalent** [if C_PUT_MDL_RCL_OPEN_DIF_NEG calibrated with a small value, e.g.: 30hPa] **– or higher** [if C_PUT_MDL_RCL_OPEN_DIF_NEG calibrated with a high value, e.g.: 200hPa] – air leakage between the output of the compressor and upstream the throttle. This leakage could be the consequence of a hose disconnected on the Intake system, a porous intercooler or a wastegate stuck in intermediate or in fully open position.

The result of the diagnosis must be "RCL open **or** Porous Intercooler **or** Hose disconnected **or** Air leakage between the output of the compressor and upstream the throttle" (1 unique Pcode).

Error-symptoms are defined to the "stuck-close" or "stuck open" diagnosis function as following:

```
┌─┬─┬─┬─┐
└─┴─┴─┴─┘
 │ │ │ └─> SYM_0: stuck close /stuck open
 │ │ └───> SYM_1
 │ └─────> SYM_2
 └───────> SYM_3
```

The error-management variables are initialized according to filter-type.

Application Condition

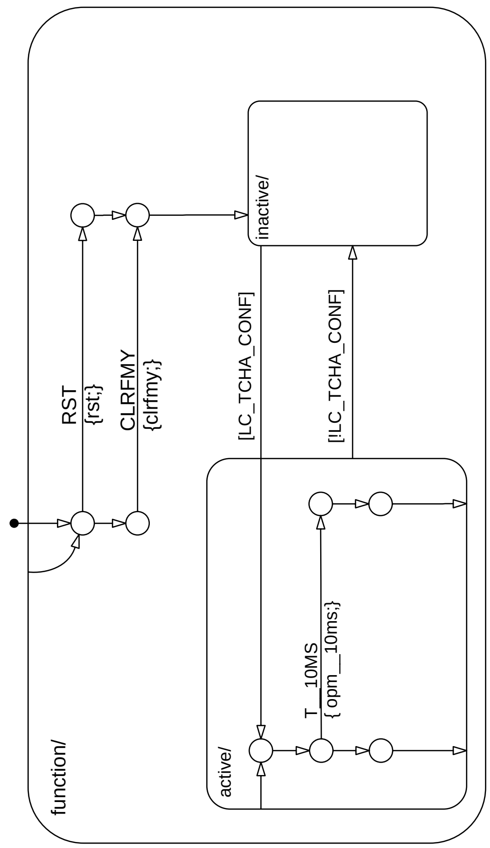


Figure 49 CHRG_RCLACRPLAUSDIAG0/ APP_CDN/ Chart

Function Description

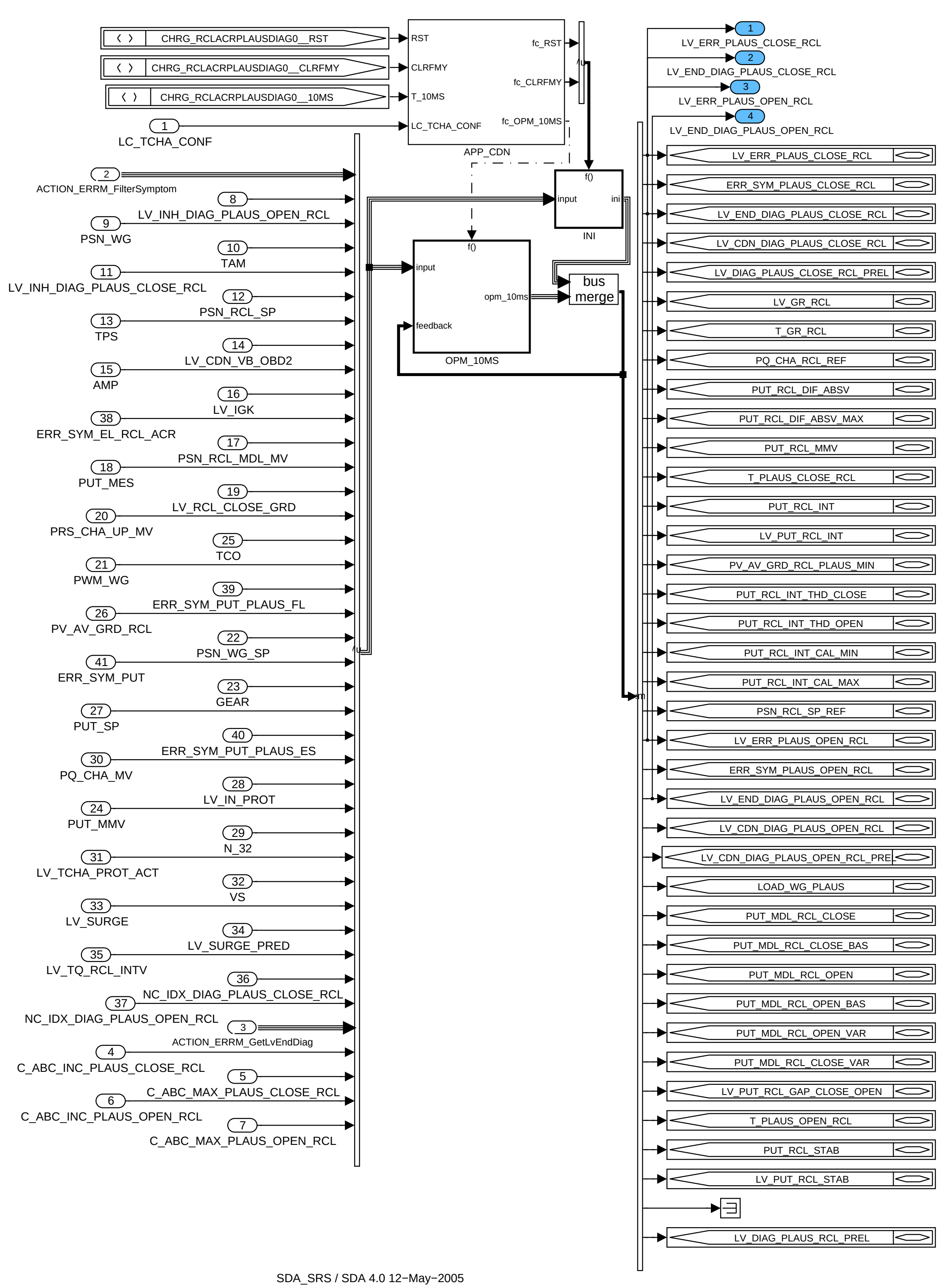


Figure 50 CHRG_RCLACRPLAUSDIAG0

B.82.1.1 Initialization

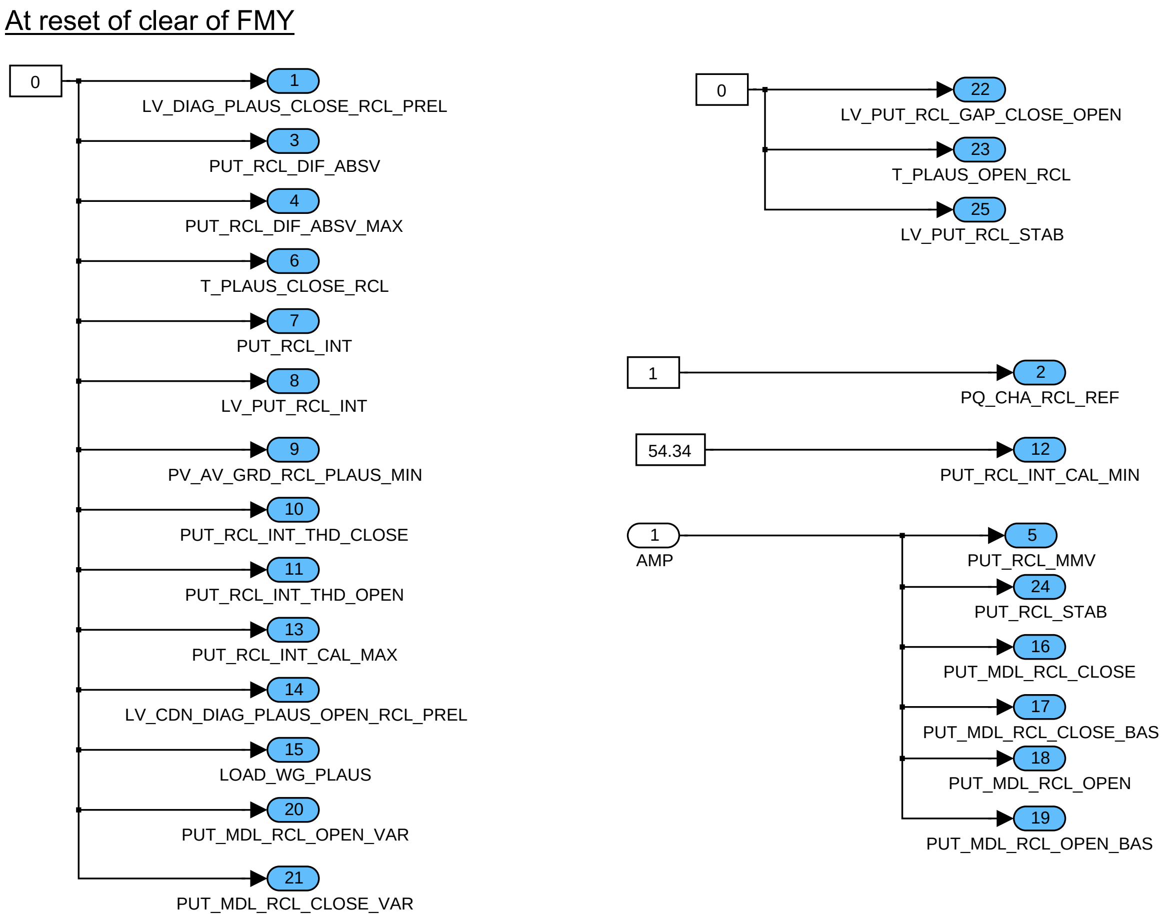


Figure 51 CHRG_RCLACRPLAUSDIAG0/ INI/ INI

B.82.1.2 Formula section

Common part for all plausibility stuck diagnoses

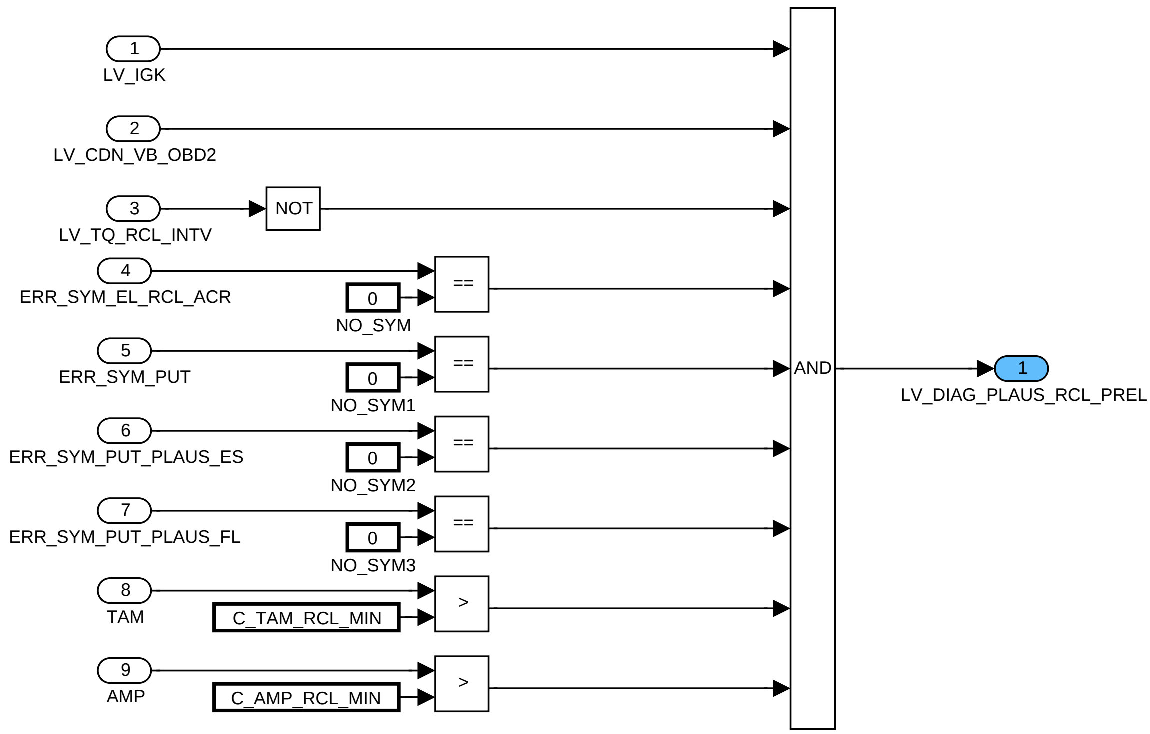


Figure 52 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ COMMON_PART

STUCK_CLOSE_DIAG: Preliminary

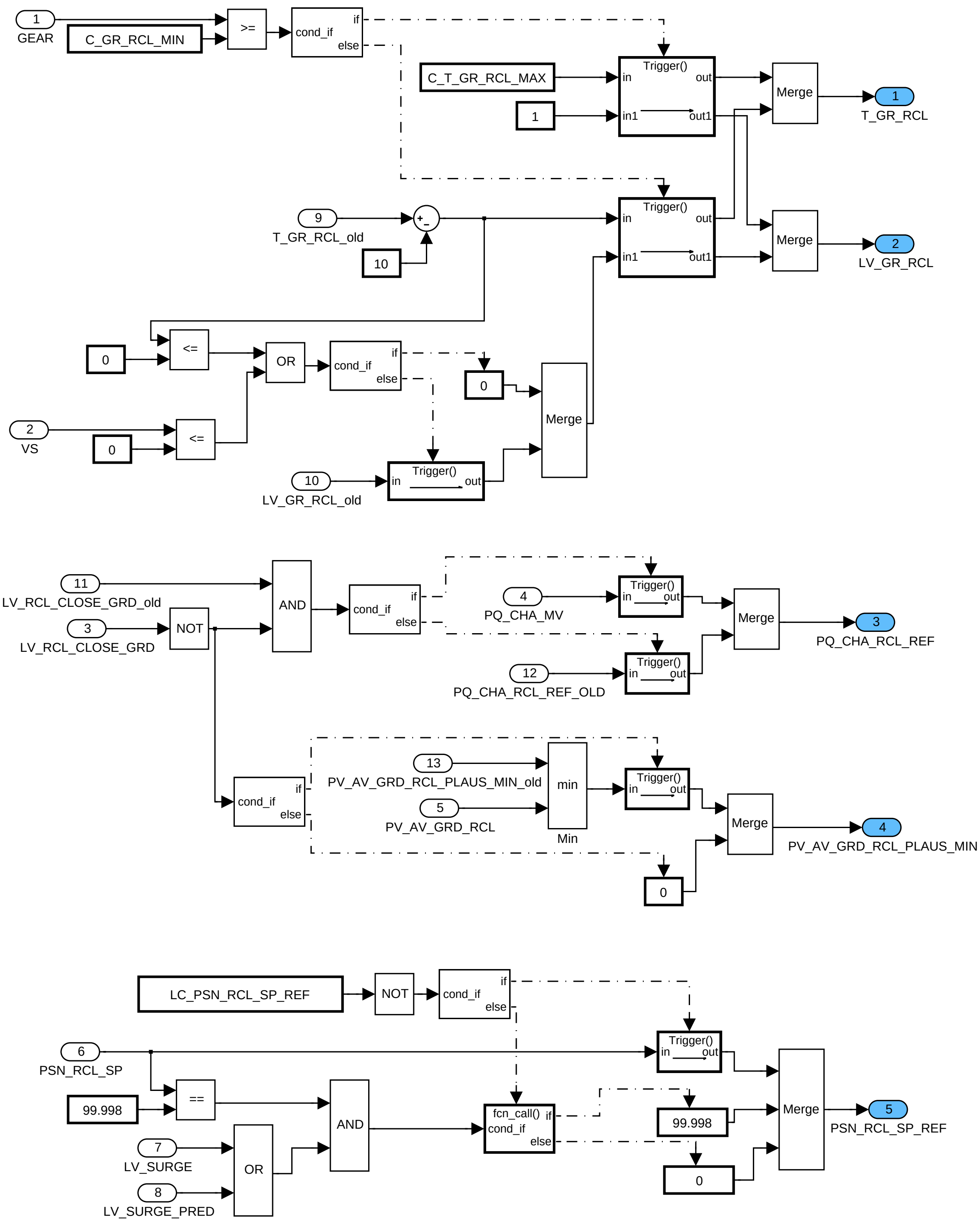


Figure 53 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__PRELIMINARY

STUCK_CLOSE_DIAG: Temporary value for T_PLAUS_CLOSE_RCL

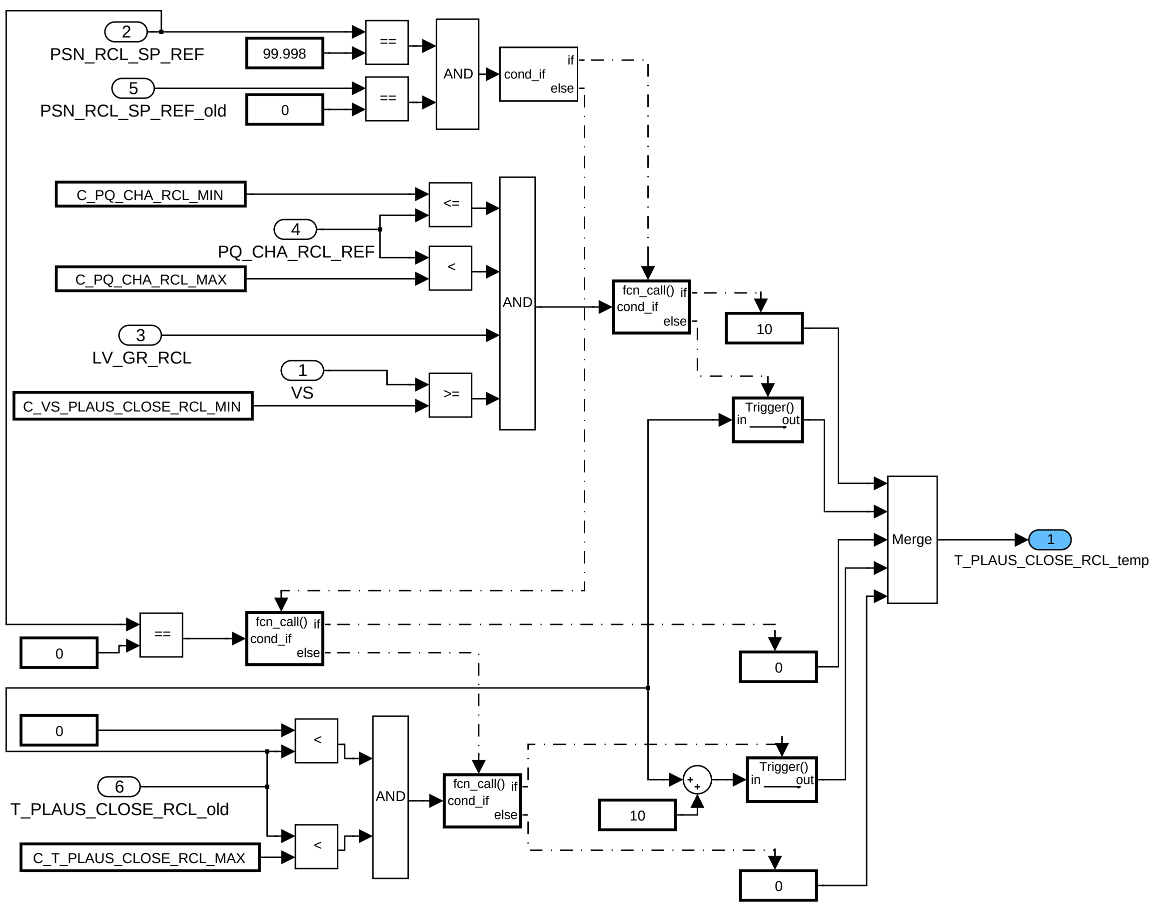


Figure 54 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__T_PLAUS_CLOSE_RCL

STUCK_CLOSE_DIAG: PUT deviation integral determination

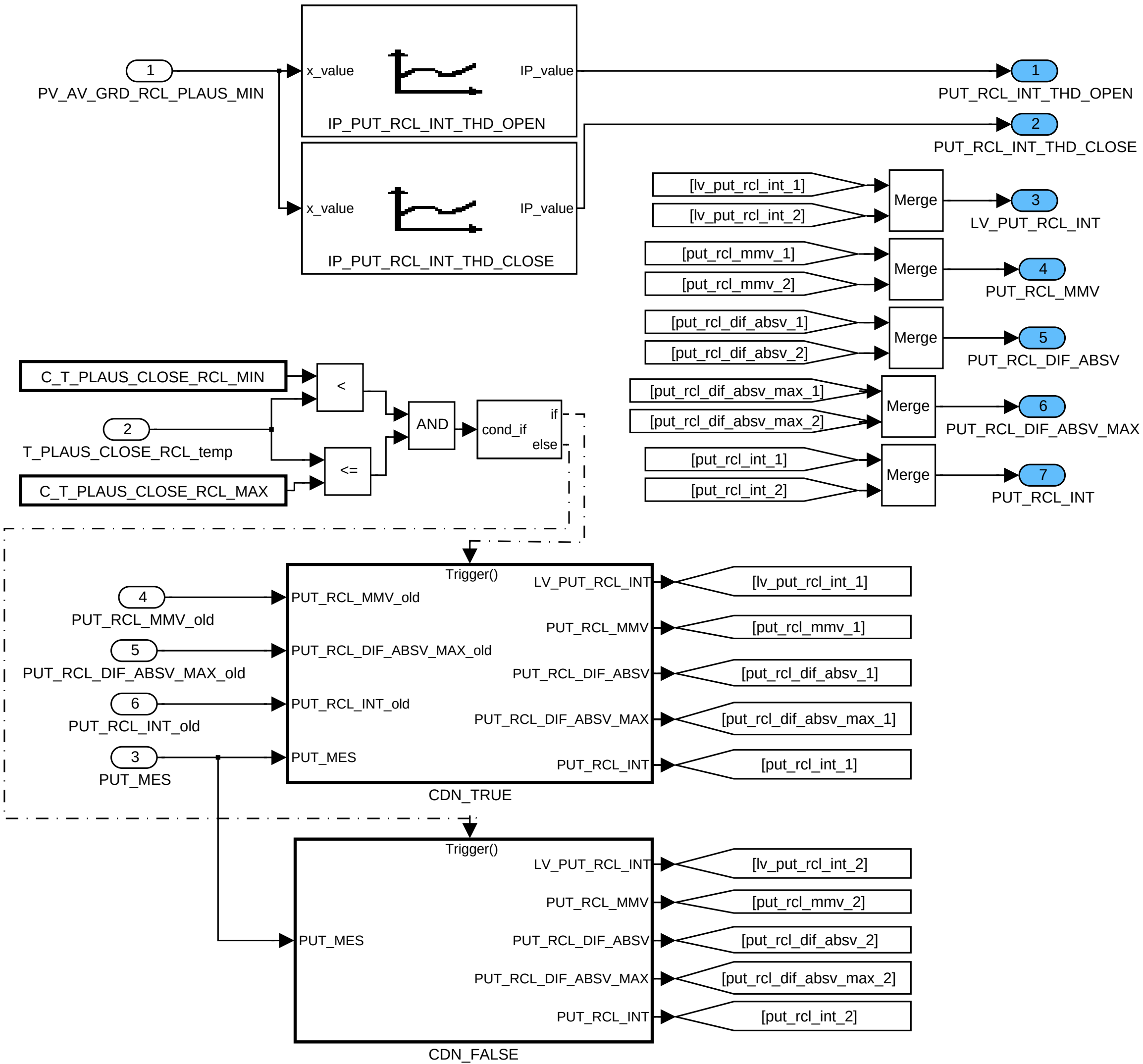


Figure 55 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__PUT_DEVIATION_INTEGRAL

Condition false

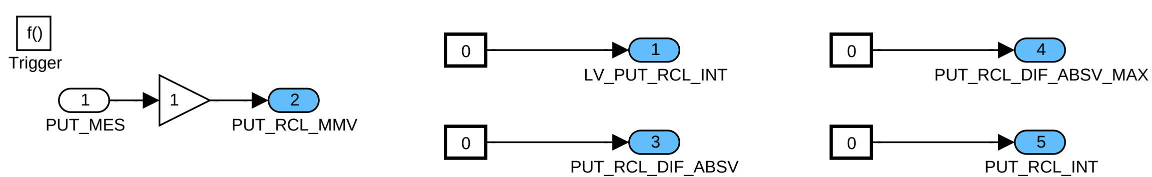


Figure 56 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__PUT_DEVIATION_INTEGRAL/ CDN_FALSE

Condition true

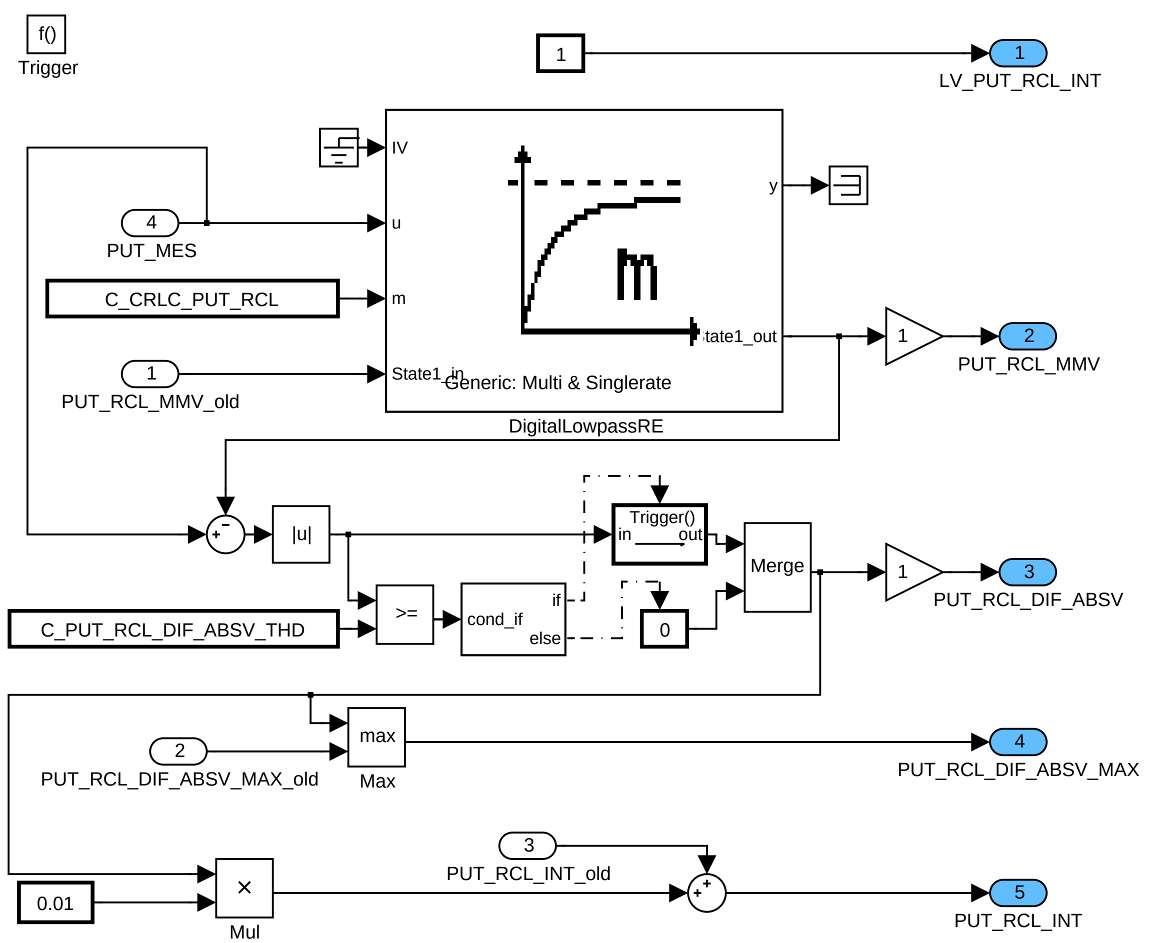


Figure 57 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__PUT_DEVIATION_INTEGRAL/ CDN_TRUE

STUCK_CLOSE_DIAG: Conditions for diagnosis

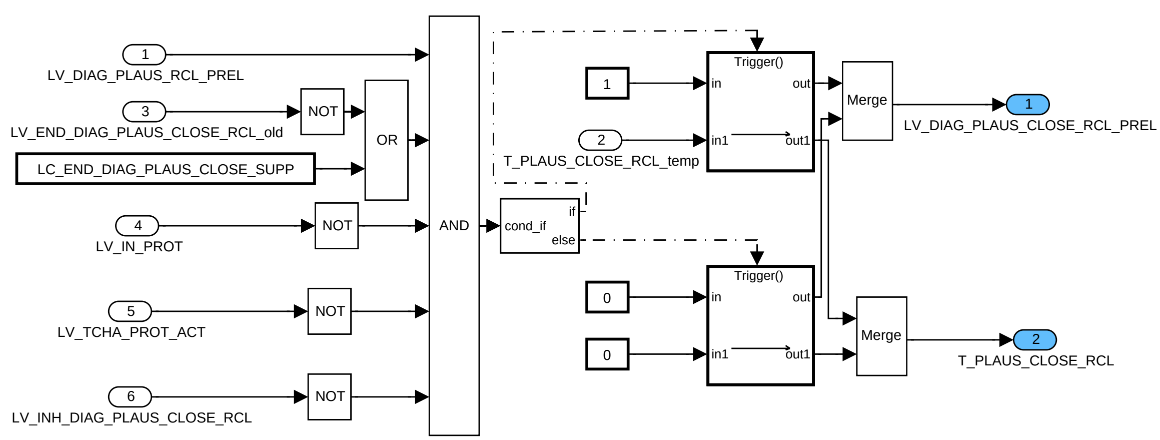


Figure 58 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__CONDITIONS_FOR_DIAGNOSIS

STUCK_CLOSE_DIAG: Diagnosis and setting of error symptom

If ERR_SYM_PLAUS_CLOSE_RCL = 0, the RCL valve opens correctly; PUT_RCL_INT is always < PUT_RCL_INT_THD_OPEN in this case because of LV_CDN_DIAG_PLAUS_CLOSE_RCL definition. In case of PUT_RCL_INT_THD_OPEN < PUT_RCL_INT < PUT_RCL_INT_THD_CLOSE the position of the valve cannot be distinguished between closed and open and LV_CDN_DIAG_PLAUS_CLOSE_RCL = 0.

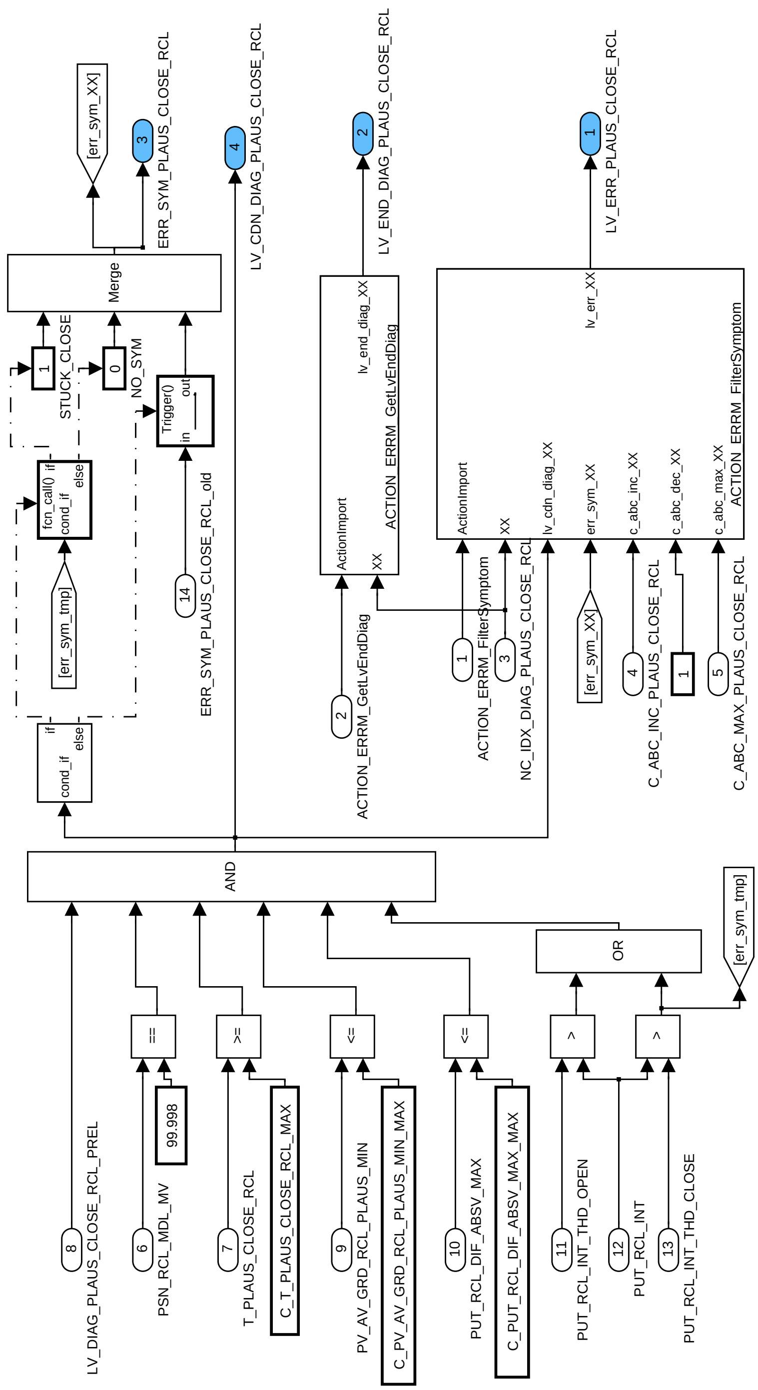


Figure 59 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__DIAGNOSIS

STUCK_CLOSE_DIAG: Calibration ease

Additional variables displayed on application tool to ease calibration

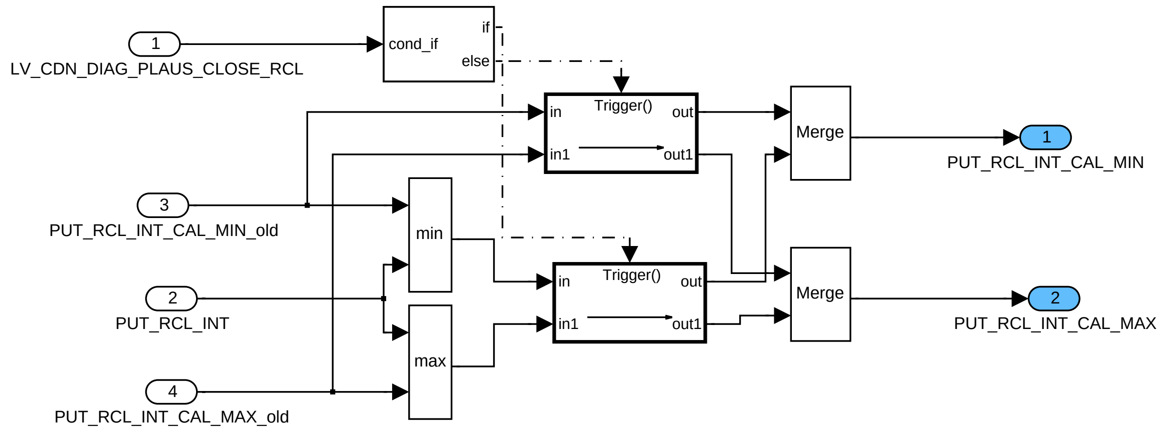


Figure 60 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ CLOSE__CALIBRATION_EASE

STUCK_OPEN_DIAG: Preliminary

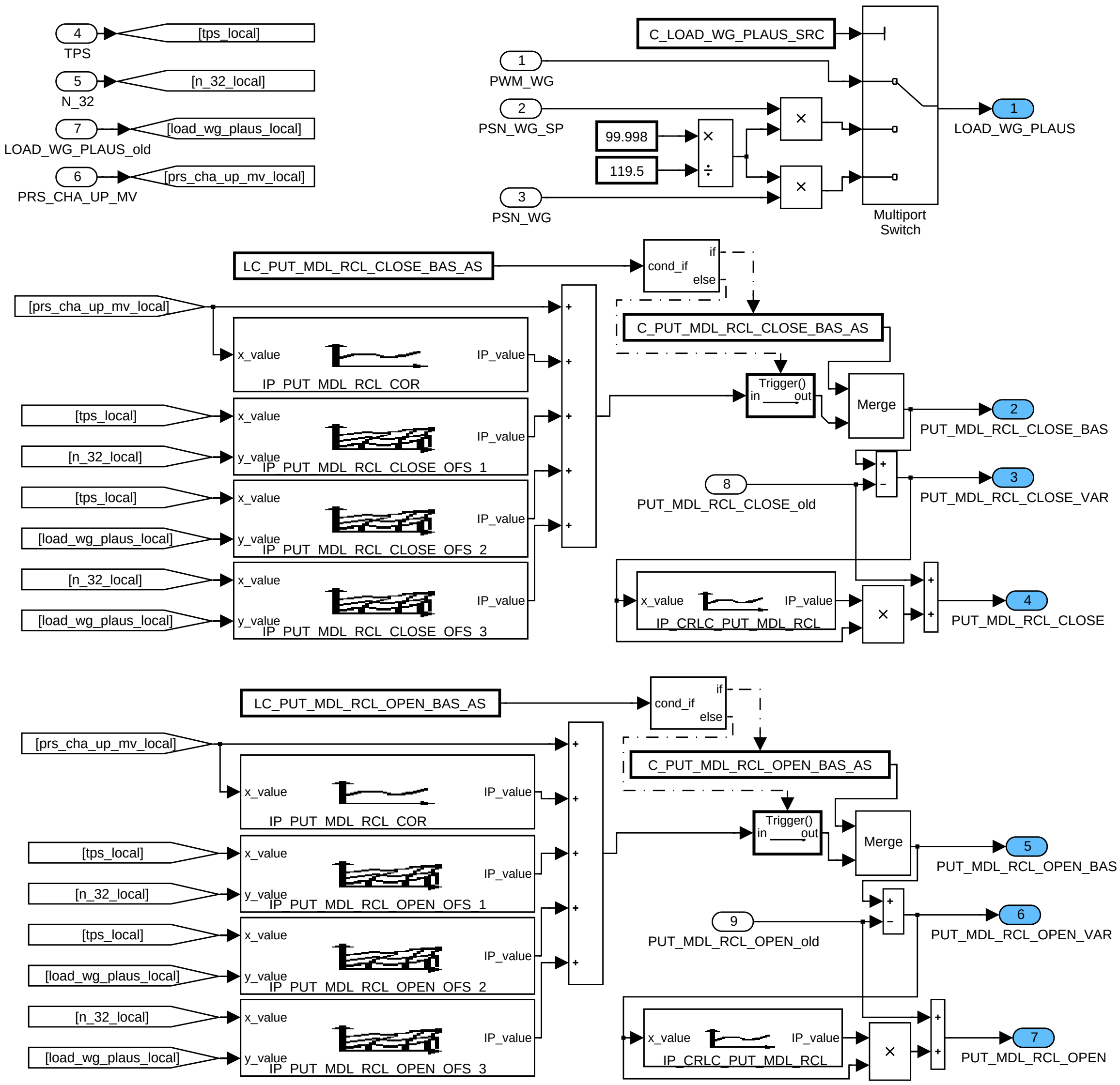


Figure 61 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ OPEN__PRELIMINARY

STUCK_OPEN_DIAG: PUT stability and minimum difference between PUT models close/open criteria

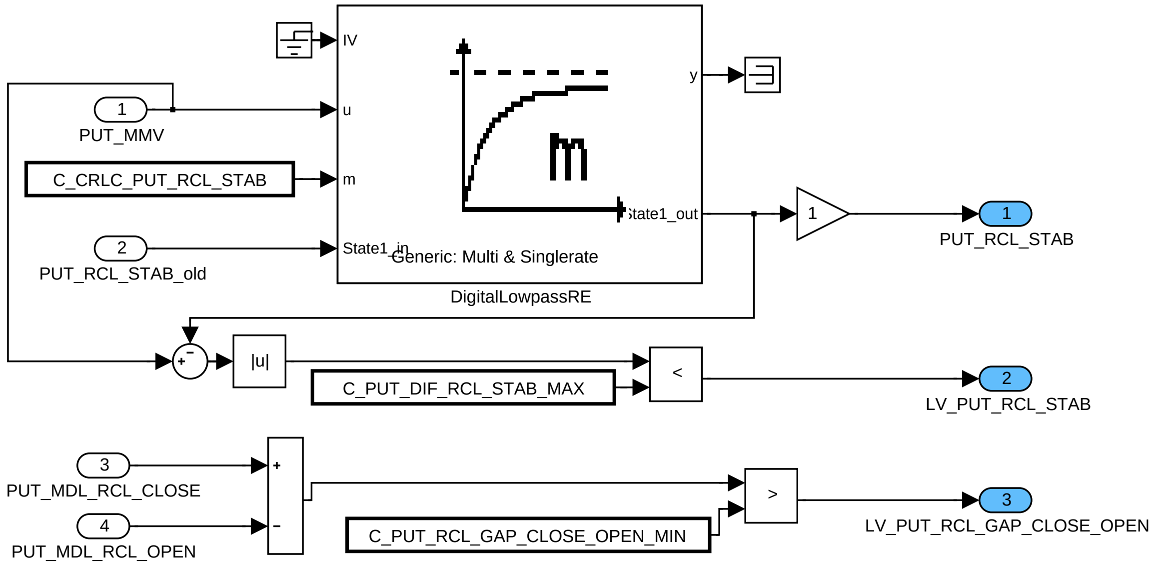


Figure 62 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ OPEN__PUT_STABILITY

STUCK_OPEN_DIAG: Conditions for diagnosis

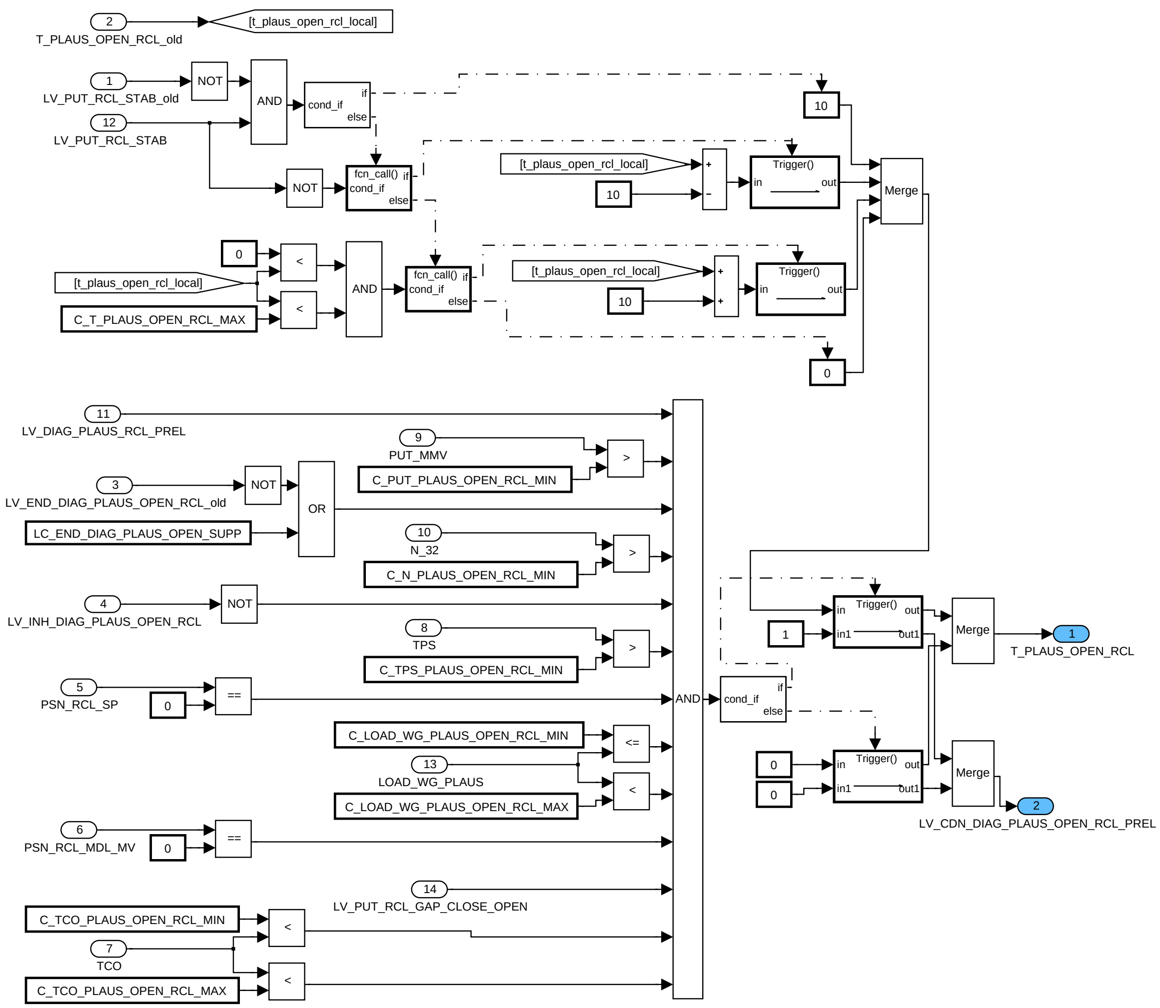


Figure 63 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ OPEN__CONDITIONS_FOR_DIAGNOSIS

STUCK_OPEN_DIAG: Diagnosis and setting of error symptom

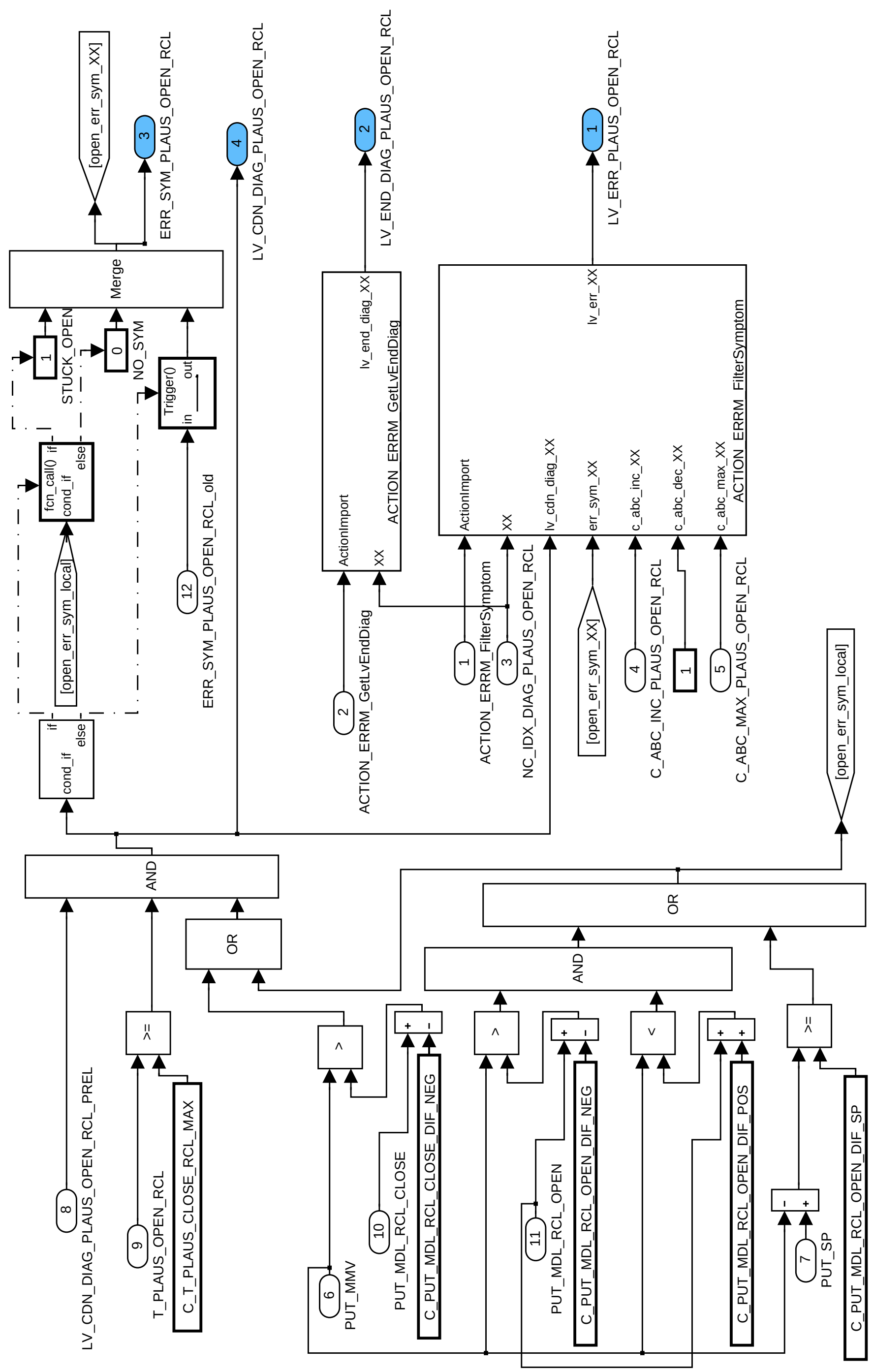


Figure 64 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ OPEN__DIAGNOSIS

Saving of old input values

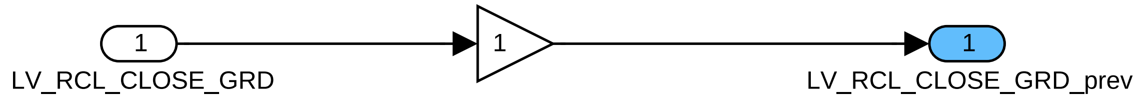


Figure 65 CHRG_RCLACRPLAUSDIAG0/ OPM_10MS/ OLD_INPUT_VALUES

B.83 Permanent fault code

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_PERM[NC_NR_ERR_PERM]	O/V/S	0...FFFFH	0...65535	1	[-]
Permanent fault index					
LV_ERR_CLR[NC_NR_ERR_PERM]	V/S	0...1H	0...1	1	[-]
Failure is only permanent due ERRM data clearing					
LV_ERR_PERM[NC_NR_ERR_PERM]	O/V/S	0...1H	0...1	1	[-]
Permanent fault code					
CTR_ERR_PERM_NR	O/V/S	0...FFH	0...255	1	[-]
Number of permanent fault codes stored					
LV_ERR_PERM_CLR_OLD	-	0...1H	0...1	1	[-]
Previous state of LC_ERR_PERM_CLR					
ERR_SYM_PERM[NC_NR_ERR_PERM]	O/V/S	0...FH	0...15	1	[-]
Permanent fault symptom					
#IF NLC_TREAT_DIAG_MIS = 0					
SYM_CYL_PERM_MIS_A	O/V/S	0...FFFFH	0...65535	1	[-]
Permanent memorized cylinder with misfire A (for the DTC building)					
SYM_CYL_PERM_MIS_B1	O/V/S	0...FFFFH	0...65535	1	[-]
Permanent memorized cylinder with misfire B1 (for the DTC building)					
SYM_CYL_PERM_MIS_B4	O/V/S	0...FFFFH	0...65535	1	[-]
Permanent memorized cylinder with misfire B4 (for the DTC building)					
#ENDIF					

Input data:

ERR_SYM_MEM[NC_NR_ERR_DYN]	NLC_TREAT_DIAG_MIS	SYM_CYL_MEM_MIS_A	SYM_CYL_MEM_MIS_B1
SYM_CYL_MEM_MIS_B4			

Export actions:

ACTION_ERRM_*StorePermanentCode* (IN <PRM_IDX_ERR>)
This action shall be called to store a permanent fault code.
ACTION_ERRM_*ErasePermanentCode* (IN <PRM_IDX_ERR>)
This action shall be called to erase a permanent fault code (self clearing).
ACTION_ERRM_*GetErrPerm* (IN <PRM_IDX_ERR>, OUT<PRM_LV_ERR_PERM>)
This action is called to know if a given failure has a permanent code stored or not.
ACTION_ERRM_*ClcPermanentIniErrm* ( )
This action is called when all error management data is cleared.
ACTION_ERRM_*ClcPermanentByErr* (IN <PRM_IDX_ERR>)
This action is called when one unique failure is cleared by a diagnostic tool.
ACTION_ERRM_*ClcPermanentResetDC* ( )
This action is called to update and sort permanent memory at end of driving cycle.

Import actions:

ACTION_ERRM_*GetErrLastClr* (IN<PRM_IDX_ERR>, OUT <PRM_LV_ERR_LST_CLR>)
ACTION_ERRM_*GetLvEndDiag*(IN<PRM_IDX_ERR>, OUT <PRM_LV_END_DIAG>)

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_NR_ERR_PERM	1	1...FFH	1...255	1	[-]
Maximum number of failure defined in permanent failure memory structure (CARB required minimum value : 4)					

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ERR_PERM_CLR	1	0...1H	0...1	1	[-]
Calibration to clear permanent failure memory (at 0->1 transition)					

FUNCTION DESCRIPTION:

General information:

Any DTC that is commanding MIL on must be logged as a permanent fault code in non-volatile memory. A Permanent DTC can only be erased by the vehicle's OBD II system.

Vehicle owners and technicians would not be able to clear or erase permanent fault codes by any generic or manufacturer specific scan tool command (or by disconnecting the battery). Instead, these fault codes would only be allowed to be self-cleared by the OBD II system itself (ERRM), once the monitor responsible for setting that fault code has run enough times to confirm that the fault was no longer present.

The permanent fault code is set as soon as the failure gets the confirmed status (MIL is switched on).
The permanent fault code is erased after self-healing of the failure, when the MIL goes off. Subsequent to a clearing of the failure memory (e.g. Mode $04), the permanent fault code is erased if the diagnostic that caused the permanent fault code to be stored has been fully executed and determined the malfunction is no longer present.

A minimum of four permanent fault codes shall be retained by by the OBD II system (ERRM). But the permanent failure memory may be adjusted over the four elements size thanks to NC_NR_ERR_PERM configuration data.
Permanent fault codes may not be erased when ECU is reprogrammed unless the readiness status for all monitored components is set to "not complete" in conjunction with the reprogramming event.

The permanent failure memory structure is defined as follow:

Index: IDX_PERM $\in$ [1...NC_NR_ERR_PERM]

- Permanent fault index ERR_PERM[IDX_PERM]
- Permanent fault code LV_ERR_PERM[IDX_PERM]
- Failure is only permanent due ERRM data clearing LV_ERR_CLR[IDX_PERM]
- Permanent Fault symptom ERR_SYM_PERM[IDX_PERM]
- Permanent memorized cylinder with misfire A, B1, B4 SYM_CYL_PERM_XX

B.83.1 Permanent failure memory

FUNCTION DESCRIPTION:

General information:

The permanent failure memory is a data structure containing NC_NR_ERR_PERM elements. Permanent faults codes are stored there, with a chronological order, but the clearing is done randomly, because it is based on the natural failure healing. When the data structure is full, no additional permanent fault code can be entered, until an empty space is freed within the structure.
The permanent failure memory must retain at least four permanent faults codes.

Application conditions:

Initialisation: *at NVMINI system event*

CTR_ERR_PERM_NR = 0

For IDX_PERM = 1 to NC_NR_ERR_PERM

ERR_PERM[IDX_PERM] = 0

LV_ERR_CLR[IDX_PERM] = 0

LV_ERR_PERM[IDX_PERM] = 0

ERR_SYM_PERM[IDX_PERM] = 0 (NO_SYM)

#IF NLC_TREAT_DIAG_MIS = 0

SYM_CYL_PERM_MIS_A = 0,

SYM_CYL_PERM_MIS_B1 = 0,

SYM_CYL_PERM_MIS_B4 = 0

#ENDIF

Endfor

Recurrence: -

Activation: -

Formula section:

No action specified.

B.83.2 Storage of permanent fault code

Description for actions:

ACTION_ERRM_*StorePermanentCode* (< PRM_IDX_ERR >)					
This action shall be called in order to store a permanent fault code. When permanent memory is full, the permament code storage is postponed until a free space within memory is released. Permanent codes are stored according the chronological appearance order.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Failure index					

Formula section:

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) ERR_PERM[IDX_PERM] = PRM_IDX_ERR

Then(1)

(Permanent failure related to PRM_IDX_ERR is already stored)

(If failure was only permanent due to a previous ERRM data clearing, this information is cleared to avoid permanent clearing at end of driving cycle)

LV_ERR_CLR[IDX_PERM] = 0

LV_ERR_PERM[IDX_PERM] = 1

Exit

Else(1)

If(2) CTR_ERR_PERM_NR <> NC_NR_ERR_PERM

Then(2)

(Permanent failure memory is not full)

If(3) LV_ERR_PERM[IDX_PERM] = 0

Then(3)

(Empty area detected)

ERR_PERM[IDX_PERM] = PRM_IDX_ERR

LV_ERR_CLR[IDX_PERM] = 0

LV_ERR_PERM[IDX_PERM] = 1

If NLC_TREAT_DIAG_MIS = 1

Then

"Conditions for treatment are true"

Endif

If failure is **not** with status = CARB_MIS

Then

"Conditions for treatment are true"

Endif

If "Conditions for treatment are true"

Then

with IDX_DYN = f (PRM_IDX_ERR)

ERR_SYM_PERM[IDX_PERM] = ERR_SYM_MEM[IDX_DYN]

Else

If PRM_IDX_ERR = MIS_A

Then

SYM_CYL_PERM_MIS_A = SYM_CYL_MEM_MIS_A

Else If PRM_IDX_ERR = MIS_B1

Then

SYM_CYL_PERM_MIS_B1 = SYM_CYL_MEM_MIS_B1

Else If PRM_IDX_ERR = MIS_B4

Then

SYM_CYL_PERM_MIS_B4 = SYM_CYL_MEM_MIS_B4

Endif

EndIf

If(4) CTR_ERR_PERM_NR < NC_NR_ERR_PERM

Then(4)

CTR_ERR_PERM_NR = CTR_ERR_PERM_NR + 1

Else(4)

CTR_ERR_PERM_NR = NC_NR_ERR_PERM

Endif(4)

Exit

Else(3)

(A permanent failure is already logged at this position)

Endif(3)

Else(2)

(Permanent failure memory is full)

(PRM_IDX_ERR failure can not be stored as permanent until a free space is released)

Endif(2)

Endif(1)

Endfor

B.83.3 Erasing permanent fault code (self-clearing)

Description for actions:

ACTION_ERRM_*ErasePermanentCode* (< PRM_IDX_ERR >)					
This action shall be called to erase a permanent fault code from memory. A permanent fault code is always self cleared. None external action can affect or delete this code.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Failure index					

Formula section:

(Erasing permanent fault code)

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) ERR_PERM[IDX_PERM] = PRM_IDX_ERR

Then(1)

(PRM_IDX_ERR error to be cleared has been found)

ERR_PERM[IDX_PERM] = 0

LV_ERR_CLR[IDX_PERM] = 0

LV_ERR_PERM[IDX_PERM] = 0

If NLC_TREAT_DIAG_MIS = 1

Then

"Conditions for treatment are true"

Endif

If failure is **not** with status = CARB_MIS

Then

"Conditions for treatment are true"

Endif

If "Conditions for treatment are true"

Then

ERR_SYM_PERM[IDX_PERM] = 0 (NO_SYM)

Else

If PRM_IDX_ERR = MIS_A

Then

SYM_CYL_PERM_MIS_A = 0

Else If PRM_IDX_ERR = MIS_B1

Then

SYM_CYL_PERM_MIS_B1 = 0

Else If PRM_IDX_ERR = MIS_B4

Then

```
                SYM_CYL_PERM_MIS_B4 = 0
            Endif
        Endif
        If(2)   CTR_ERR_PERM_NR > 0
        Then(2)
            CTR_ERR_PERM_NR = CTR_ERR_PERM_NR - 1
        Else(2)
            CTR_ERR_PERM_NR = 0
        Endif(2)
        (newly created empty place has IDX_PERM index value)
        Exit
    Endif(1)
Endfor

(Permanent failure memory sorting to remove empty space generated above)
If(1) IDX_PERM <> NC_NR_ERR_PERM
Then(1)
    For k = IDX_PERM to (NC_NR_ERR_PERM-1)
    ERR_PERM[k] = ERR_PERM[k+1]
    LV_ERR_CLR[k] = ERR_PERM[k+1]
    LV_ERR_PERM[k] = ERR_PERM[k+1]
    ERR_SYM_PERM[k] = ERR_SYM_PERM[k+1]
    Endfor
ERR_PERM[NC_NR_ERR_PERM] = 0
LV_ERR_CLR[NC_NR_ERR_PERM] = 0
LV_ERR_PERM[NC_NR_ERR_PERM] = 0
ERR_SYM_PERM[NC_NR_ERR_PERM] = 0 (NO_SYM)
Endif(1)
```

B.83.4 Impact of error management data clearing inside permanent memory

Description for actions:

ACTION_ERRM_*ClcPermanentIniErrm* ( )					
This action is called when all error management data is cleared (e.g. Mode $04 executed by scan tool). After an overall data clearing of error management data, permanent errors are marked in order to allow a self clearing at the end of next driving cycle, if the following conditiong are gathered: - diagnosis has run long enough to detect a possible malfunction - malfunction not present					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
No parameter					

Formula section:

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) LV_ERR_PERM[IDX_PERM] = 1

Then(1)

(A permanent fault code is stored at IDX_PERM position)

LV_ERR_CLR[IDX_PERM] = 1

Else(1)

(No permanent fault code stored as from position IDX_PERM until end)

Exit

Endif(1)

Endfor

B.83.5 Impact of failure clearing (with diagnostic tool) inside permanent memory

Description for actions:

ACTION_ERRM_*ClcPermanentByErr* (< PRM_IDX_ERR >)					
This action is called when one unique failure is cleared by a diagnostic tool. After clearing of any individual failure, the related permanent code is marked to allow a self clearing at the end of next driving cycle, if the following conditiong are gathered: - diagnosis has run long enough to detect a possible malfunction - malfunction not present					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Failure index					

Formula section:

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) ERR_PERM[IDX_PERM] = PRM_IDX_ERR

Then(1)

LV_ERR_CLR[IDX_PERM] = 1

Exit

Endif(1)

Endfor

B.83.6 End of driving cycle computations

Description for actions:

ACTION_ERRM_*ClcPermanentResetDC* ( )					
This action is called by dynamic error management core module to execute computations linked to the end of driving cycle.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
No parameter					

Formula section:

(Clearing of permanent failure for which diagnostic has passed)

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) (LV_ERR_PERM[IDX_PERM] = 1 **and** LV_ERR_CLR[IDX_PERM] = 1)

Then(1)

ACTION_ERRM_*GetLvEndDiag*(ERR_PERM[IDX_PERM], PRM_LV_END_DIAG)

ACTION_ERRM_*GetErrLastClr*(ERR_PERM[IDX_PERM], PRM_LV_ERR_LST_CLR)

If(2) (PRM_LV_ERR_LST_CLR = 0 **and** PRM_LV_END_DIAG = 1)

Then(2)

If NLC_TREAT_DIAG_MIS = 1

Then

"Conditions for treatment are true"

Endif

If failure is **not** with status = CARB_MIS

Then

"Conditions for treatment are true"

Endif

If "Conditions for treatment are true"

Then

ERR_SYM_PERM[IDX_PERM] = 0 (NO_SYM)

Else

If PRM_IDX_ERR = MIS_A

Then

SYM_CYL_PERM_MIS_A = 0

```
                              Else If PRM_IDX_ERR = MIS_B1
                              Then
                                    SYM_CYL_PERM_MIS_B1 = 0
                              Else If PRM_IDX_ERR = MIS_B4
                              Then
                                    SYM_CYL_PERM_MIS_B4 = 0
                              Endif
                        Endif
                        (Permanent failure may be cleared)
                        ERR_PERM[IDX_PERM] = 0
                        LV_ERR_CLR[IDX_PERM] = 0
                        LV_ERR_PERM[IDX_PERM] = 0
                        (Permanent failure counter is decreased)
                        If(3)   CTR_ERR_PERM_NR > 0
                        Then(3)
                                CTR_ERR_PERM_NR = CTR_ERR_PERM_NR – 1
                        Else(3)
                                CTR_ERR_PERM_NR = 0
                        Endif(3)
                Endif(2)
        Endif(1)
Endfor

(Permanent failure memory sorting to remove empty spaces – "heapsorting")
For IDX_PERM = 1 to (NC_NR_ERR_PERM-1)
        If(1)   ERR_PERM[IDX_PERM] = 0
        Then(1)
                For k = (IDX_PERM + 1) to NC_NR_ERR_PERM
                        If(2)   ERR_PERM[k] <> 0
                        Then(2)
                                (Swapping)
                                ERR_PERM[IDX_PERM] = ERR_PERM[k]
                                LV_ERR_CLR[IDX_PERM] = LV_ERR_CLR[k]
                                LV_ERR_PERM[IDX_PERM] = LV_ERR_PERM[k]
                                ERR_SYM_PERM[IDX_PERM] = ERR_SYM_PERM[k]
                                (Clearing of swapped data)
```

ERR_PERM[k] = 0

LV_ERR_CLR[k] = 0

LV_ERR_PERM[k] = 0

ERR_SYM_PERM[k] = 0 (NO_SYM)

Exit

Endif(2)

Endfor

Endif(1)

Endfor

B.83.7 Get failure permanent status

Description for actions:

ACTION_ERRM_*GetErrPerm* (<PRM_IDX_ERR>, <PRM_LV_ERR_PERM>)					
This action is called to know if a given failure has a permanent code stored or not.					
Parameter	**Type**	**Hex. Limits**	**Phys. Limits**	**Resol.**	**Unit**
PRM_IDX_ERR	IN	0...FFFFH	0...65535	1	[-]
Failure index					
PRM_LV_ERR_PERM	OUT	0...1H	0...1	1	[-]
Permanent status					

Formula section:

For IDX_PERM = 1 to NC_NR_ERR_PERM

If(1) ERR_PERM[IDX_PERM] = PRM_IDX_ERR

Then(1)

PRM_LV_ERR_PERM = 1

Exit

Else(2)

PRM_LV_ERR_PERM = 0

Endif(1)

Endfor

B.83.8 Clearing of permanent failure memory, in development

FUNCTION DESCRIPTION:

Description:

For development conveniences, it is often useful to reinitialise the permanent failure memory. This is possible by setting the calibration bit LC_ERR_PERM_CLR to 1.

Application conditions:

Initialisation: *at RST system event*

LV_ERR_PERM_CLR_OLD = LC_ERR_PERM_CLR

at IGKON system event

LV_ERR_PERM_CLR_OLD = LC_ERR_PERM_CLR

Recurrence: 1 s

Activation: Always

Formula section:

(Detection of LC_ERR_PERM_CLR 0 → 1 transition)

```
If    (LV_ERR_PERM_CLR_OLD = 0
      and
      LC_ERR_PERM_CLR = 1)
Then
      For IDX_PERM = 1 to NC_NR_ERR_PERM
            ERR_PERM[IDX_PERM] = 0
            LV_ERR_CLR[IDX_PERM] = 0
            LV_ERR_PERM[IDX_PERM] = 0
            ERR_SYM_PERM[IDX_PERM] = 0 (NO_SYM)
      Endfor
      If NLC_TREAT_DIAG_MIS = 0
      Then
            (XX stands for MIS_A, MIS_B1, and MIS_B4)
            SYM_CYL_PERM_XX = 0
      Endif
      CTR_ERR_PERM_NR = 0
Endif
LV_ERR_PERM_CLR_OLD = LC_ERR_PERM_CLR
```

C ETC Electronic throttle control

Table of Contents

Index

C.1 Calculation of setpoints for inverse model (MPI version)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_KGH_SP	O/V	0...FFFFH	0...2.04797E+3	0.03125	kg/h
Setpoint mass air flow into the manifold					
MAF_SP	O/V	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
MAF setpoint output for inverse air path					
MAP_SP	O/V	0...FFFFH	0...5.434E+3	0.08291752	hPa
mainfold air pressure setpoint					

Input data:

EFF_VOL_OFS_SUM	EFF_VOL_SLOP	MAF_SP_TQI	MAP_EGR
MAP_EGR_RATIO	N	MAF_MAX_COR	NC_CHRG_CONF
NC_MAF_FAC_CYL	NC_EGR_CONF		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_MAF_SP_MAN	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Manual Setpoint of the manifold pressure					
C_MAF_SP_MAX	1	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Maximum allowed MAF_SP					
C_MAP_SP_MAX	1	0...FFFFH	0...5.434E+3	0.08291752	hPa
Maximum allowed MAP_SP					
C_N_MIN_MAF_SP_LIM	1	0...1FE0H	0...8.16E+3	1	rpm
Minimal engine speed to limit MAF_SP with MAF_MAX_COR					
LC_MAF_SP_MAN	1	0...1H	0...1	1	-
Logical variable for MAF setpoint manual					

C.1.1 INSY_ISPCLLOAD0

For a MPI configuration the value of the MAF setpoint is equal to MAF_SP_TOI.

There is a switch that allows to set a manual value of MAF_SP and no calculation of MAF_SP will be done.

The output values are MAF_SP, MAF_KGH_SP and MAP_SP. The last two are input values into inverse air path for throttle setpoint calculation.

Depending on MAF_SP by multiplication with engine speed N and the scaling factor NC_MAF_FAC_CYL (= 8333.3 stroke*kg*min/mg/h for a four cylinder engine) the setpoint MAF_KGH_SP is calculated.

For the desired cylinder charge in kg/h it is possible to calculate for the actual engine speed the manifold air pressure MAP_SP necessary to relize this air mass flow.

Function Description

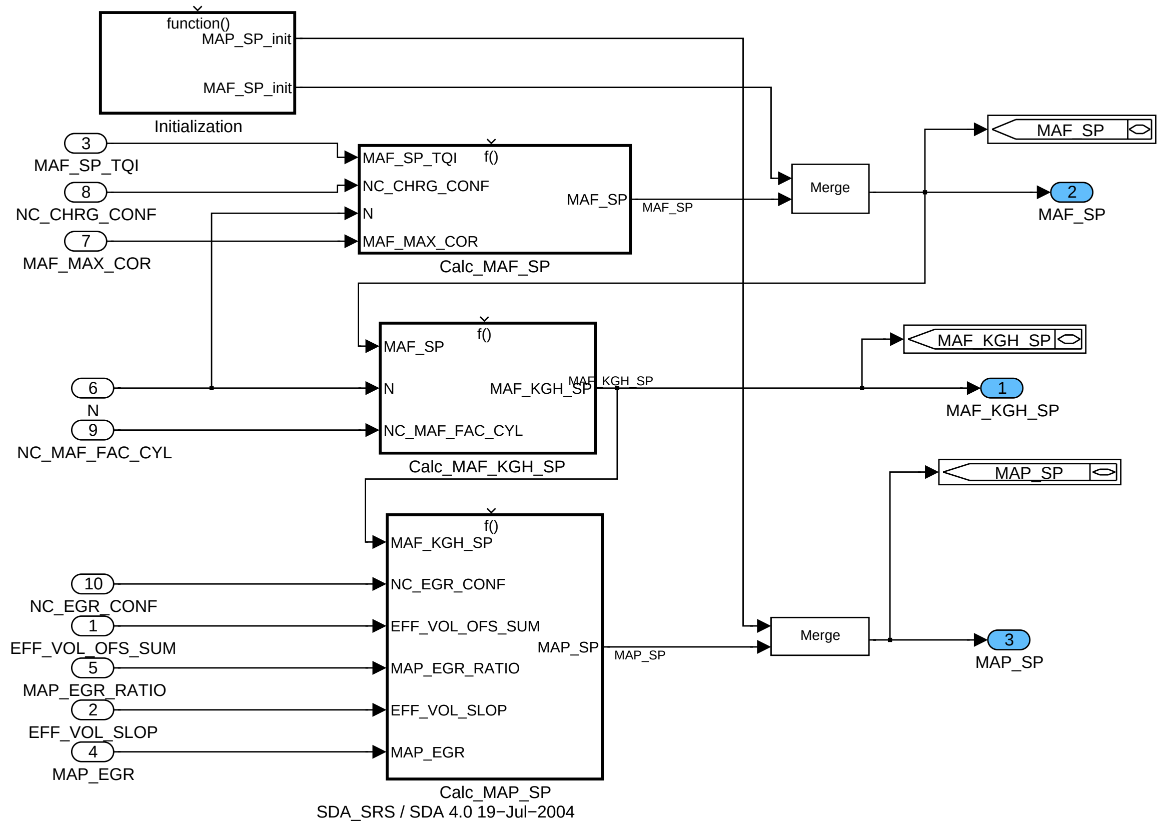


Figure 1 INSY_ISPCLLOAD0

C.1.1.1 Calculation of MAF_KGH_SP

MAF_KGH_SP is the desired air mass flow in kg/h which satisfy the adjusted torque.

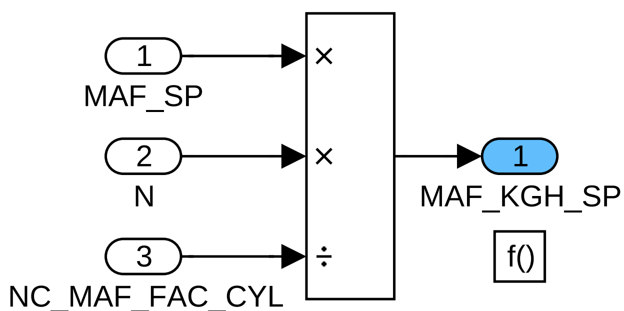


Figure 2 INSY_ISPCLLOAD0/ Calc_MAF_KGH_SP

C.1.1.2 Calculation of MAF_SP

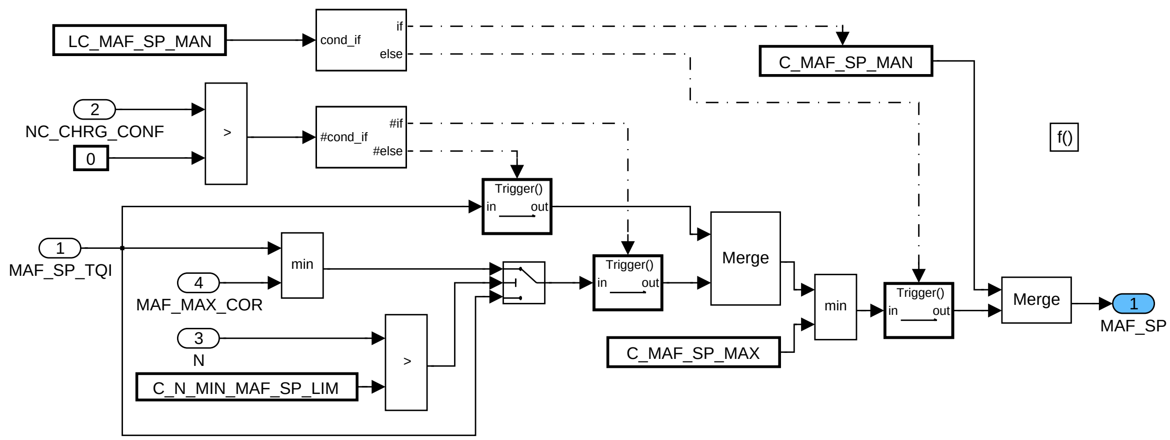


Figure 3 INSY_ISPCLLOAD0/ Calc_MAF_SP

C.1.1.3 Calculation of air flow function rate MAP_SP

For a certain engine speed the mass flow into the intake manifold is corresponding with a certain manifold pressure under steady state engine operating.

There is linear correlation between the setpoint of manifold pressure MAP_SP and the mass flow. The two interpolation tables are the same as in the intake manifold model (function: volumetric efficiency)

In case the intake pressure is not equal to zero the calculation of the setpoint of the pressure considers this intake pressure.

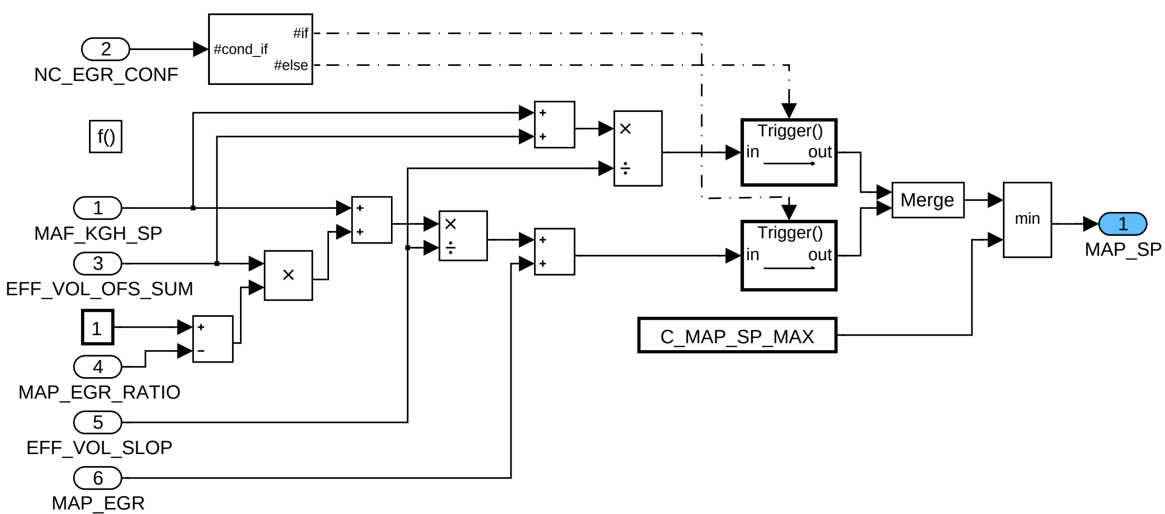


Figure 4 INSY_ISPCLLOAD0/ Calc_MAP_SP

C.1.1.4 SUBFUNCTION: Initialization

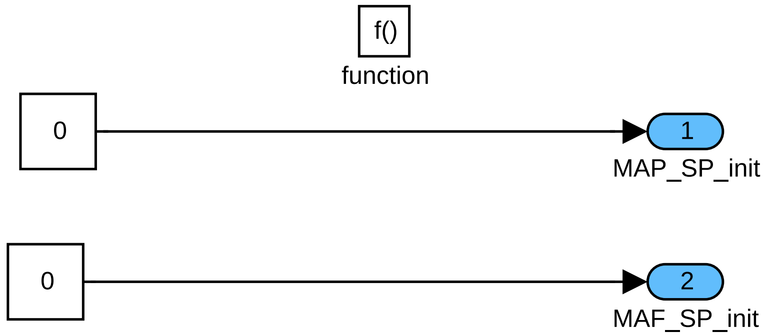


Figure 5 INSY_ISPCLLOAD0/ Initialization

C.2 Throttle Position Setpoint Calculation (Inverse Intake Manifold Model)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AR_RED_SP_1	V	0...FFFFH	0...58.59	8.94e-4	cm2
Setpoint reduced throttle area					
AR_RED_SP_MMV	V	0...FFFFH	0...58.59	8.94e-4	cm2
Moving mean value of desired reduced area					
AR_RED_SP_MMV_1	V	0...FFFFH	0...58.59	8.94e-4	cm2
Moving mean value of AR_RED_SP_1					
PQ_SP	O/V	0...FFFFH	0...0.999985	1.52e-5	-
Setpoint of the pressure quotient					
PUT_IVS	V	0...FFFFH	0...5434	0.083	hPa
Pressure upstream the throttle inverse					
PSI_SP	V	0...FFFFH	0...8.88033	1.35e-4	-
Setpoint air flow function rate					
TPS_SP_MDL	V/O	0...3FFFH	0...119.5	7.29e-3	°TPS
Setpoint throttle position					
TPS_SP_MDL_BAS	V	0...3FFFH	0...119.5	7.29e-3	°TPS
Setpoint of the throttle position before maximum check					
TPS_SP_MDL_BAS_COR	V	0...3FFFH	0...119.5	7.29e-3	°TPS
Setpoint throttle position after linear interpolation					
TPS_SP_MDL_BAS_COR_FIL	V	0...3FFFH	0...119.5	7.29e-3	°TPS
Filtered setpoint throttle position after linear interpolation					
TPS_SP_MDL_MAX	V	0...3FFFH	0...119.5	7.29e-3	°TPS
Setpoint throttle position after limitation					
OPG_SP_MDL_ISA	O	0000...FFFFH	0…99.9985	0.0015	-
Setpoint of idle speed actuator opening calculated by the inverse intake manifold model (dummy output)					
PQ_SP_1	-	0...FFFFH	0...0.999985	1.52e-5	-
Temporary Setpoint of the pressure quotient					
LV_MAP_SP_THR	V	0...1H	0 ... 1	1	-
Flag indicates MAP_SP is over the threshold					
MAF_SP_RATIO	V	0...FFFFH	0...0.999885	1.52e-5	-
Setpoint of the ratio MAF_SP / MAF_MAX_COR					
PQ_SP_LPF_AR_RED_PUT	V	0...FFFFH	0...0.99998474	1.52588E-5	-
Throttle pressure quotient setpoint gradient for INSY controller					
PQ_SP_GRD_AR_RED_PUT	O/V	8000...7FFFH	-1...0.99996948	3.05176E-5	-
Throttle pressure quotient setpoint gradient for INSY controller					
MAP_DIF_TPS_SP_CTL	V	8000... 7FFFH	-2717... 2716.91708	0.0829163	[hPa]
Difference between the MAP_SP and MAP as an input to the controller.					

TPS_SP_MDL_I_CTL	V	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
The I value of the PI controller for the calculation of TPS_SP_MDL					
TPS_SP_MDL_I_CTL_TMP	-	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
The temporary value of I controller for the calculation of TPS_SP_MDL					
TPS_SP_PQ_MAX	O/V	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
Throttle position for C_PQ_SP_MAX					
TPS_SP_MDL_MIN_TMP	-	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
The temporary minimum throttle angle value between TPS_SP_MDL_BAS and TPS_PQ_SP_MAX					
TPS_SP_MDL_P_CTL	V	0... 3FFFH	0... 119.5	7.29415e-3	[°TPS]
The P control value used for the calculation of the TPS_SP_MDL					
PSI_SP_TPS_PQ_SP_MAX	V	0...FFFFH	0...8.88033	1.35e-4	-
Setpoint air flow function rate derived for C_PQ_SP_MAX					
AR_RED_SP_PQ_MAX_1	V	0...FFFFH	0...58.59	8.94e-4	cm2
Value of desired reduced area for PQ=0.97					
AR_RED_SP_PQ_MAX_MMV	V	0...FFFFH	0...58.59	8.94e-4	cm2
Filtered value Value of desired reduced area for PQ=0.97					
AR_RED_SP_PQ_MAX_MMV_1	V	0...FFFFH	0...58.59	8.94e-4	cm2
Temporary Filtered value Value of desired reduced area for PQ=0.97					

Input data:

AMP	AR_RED_AD_ADD	AR_RED_AD_FAC_COR	AR_RED_DIF_I_REL_MMV
PV_AV	PUT_MDL_DIF_I_MMV	FAC_AR_RED_MAF	FLOW_CPS
ID_MAF_FAC_OFS	ID_MAF_FAC_SLOP	IP_PRS_LOSS_AIC	LV_IGK
PUT	MAF_KGH_SP	MAF_MAX_COR	MAF_MDL_CON_1
MAF_SP	MAP_SP	N	N_32
LC_TCHA_CONF	LV_PL	PQ	LV_AT
VS	PUT_WG_OPEN	PUT_PRED	LV_CRU_ACT
PV_CRU	LV_AT		

FUNCTION DESCRIPTION:

General information:

With the setpoint of the manifold pressure and the ambient pressure it is possible to calculate the opening of the throttle TPS_SP.

This modul will generate 2 calibratable data LC_AR_RED_FIL_ENA and LC_TPS_FIL_ENA to switch ON/OFF the filters.

Application conditions:

Initialisation: At Reset and LV_IGK = 0 -> 1

LV_MAP_SP_THR =0
TPS_SP_MDL = 0 °TPS
OPG_SP_MDL_ISA =0 (never changed)

(Calculation of throttle values PSI setpoint for C_PQ_MAX)
PSI_SP_TPS_PQ_SP_MAX = ID_MAF_FAC_OFS (C_PQ_SP_MAX)
- ID_MAF_FAC_SLOP (C_PQ_SP_MAX)* C_PQ_SP_MAX

Recurrence: 10 ms

Activation: at every engine operation state

Deactivation: -

Remark: **if** LC_TPS_SP_MAN_ENA = 1 **then**
"Only capitel1.1 will be realize"
else "go directly to1.2 till 1.4"
endif

C.2.1 Final throttle position setpoint value

If the flag LC_TPS_SP_MAN_ENA is set, the throttle position setpoint will always be the calibratable value IP_TPS_SP_MAN. If the flag is unset, the calculated TPS_SP_MDL will be realized.

If LC_TPS_SP_MAN_ENA = 1
And LV_PL = 1
Then TPS_SP_MDL = IP_TPS_SP_MAN(N_32,PV_AV)
Endif

C.2.2 Calculation of the air flow function rate PSI_SP

With the setpoint of the manifold pressure MAP_SP and the pressure upstream throttle PUT it is possible to calculate an approximation value of the air flow function rate PSI_SP. Before the pressure ratio PQ_SP (= MAP_SP / PUT_IVS) is need. For the calculation of the pressure upstream the throttle PUT_IVS the I-value of the ambient controller, delayed with a first order time delay, is included in the inverse model. PQ_SP is limited to a calibratable threshold C_PQ_SP_MAX.

If LC_TCHA_CONF =0
Then *(non turbo charger)*
PUT_IVS = PUT_MDL_DIF_I_MMV + AMP - IP_PRS_LOSS_AIC
(MAF_KGH_SP, VS)
PQ_SP_1= MAP_SP / PUT_IVS
(The input of the table IP_PRS_LOSS_AIC is the airflow MAF_KGH_SP and the vehicle speed VS. The table is identical with the table of the pressure decrease in the air cleaner.)
Else *(turbo charger)*
If LC_PUT_IVS_PUT_PRED = 1
Then PUT_IVS = PUT_MDL_DIF_I_MMV + PUT_PRED

Else PUT_IVS = PUT_MDL_DIF_I_MMV + PUT

Endif

The pressure quotient setpoint PQ_SP is kept equal to 1 as MAP_SP is bigger than the basic charge air pressure PUT_WG_OPEN minus a calibratable offset IP_PPUT_ DIF_PQ_SP_1 or when MAP_SP is decreasing till the threshold minus the hysteresis in the MAP_SP (this hysteresis drop shouldn´t allow that the throttle closes, it muss keep fully open). This threshold means the pressure up the throttle at steady engine operation with fully open wastegate (turbo charge idle speed).It depends on the engine speed and on the ambient pressure (altitude influence).

PRS_THD_PQ_SP = PUT_WG_OPEN + IP_PUT_ DIF_PQ_SP_1 (N_32, AMP)

If ((MAP_SP >PRS_THD_PQ_SP) or (MAP_SP >PRS_THD_PQ_SP – C_MAP_SP_HYS_CHRG **And** LV_MAP_SP_THR = 1))

Then PQ_SP_1 = 0.999985

LV_MAP_SP_THR =1

Else PQ_SP_1 = MAP_SP / PUT_IVS

LV_MAP_SP_THR =0

Endif

Endif

The flag LV_MAP_SP_THR defines if the MAP_SP has once exceed the corresponding threshold and keeps at least over the threshold minus the hysteresis value.

If LC_TPS_FIL_ENA = 1

Then PQ_SP = MIN (C_PQ_SP_MAX, PQ_SP_1)

Else PQ_SP = PQ_SP_1

Endif

In the INSY-controller a factor is introduced which reduce the I-part of the controller at fast load changes. To detect fast load changes the difference between the last and the current filtered PQ_SP is built.

$$PQ\_SP\_LPF\_AR\_RED\_PUT = PQ\_SP\_LPF\_AR\_RED\_PUT_{(n-1)} + IP\_CRLC\_PQ\_SP\_AR\_RED\_PUT * (PQ\_SP - PQ\_SP\_LPF\_AR\_RED\_PUT_{(n-1)})$$

$$PQ\_SP\_GRD\_AR\_RED\_PUT = PQ\_SP\_LPF\_AR\_RED\_PUT – PQ\_SP\_LPF\_AR\_RED\_PUT_{(n-1)}$$

The input of the two tables ID_MAF_FAC_SLOP and ID_MAF_FAC_OFS is the pressure ratio PQ_SP. The tables are idential with the tables of the intake manifold model (function: estimation of the pressure quotients).

PSI_SP = ID_MAF_FAC_OFS (PQ_SP) - ID_MAF_FAC_SLOP (PQ_SP)* PQ_SP

C.2.3 Calculation of the basic throttle position setpoint TPS_SP_1_BAS

With the corrected air mass flow demand MAF_KGH_SP it is possible to calculate the necessary reduced area at the throttle.

To reduce a trembling throttle a smoothing function for wide open throttle conditions is found. If the setpoint of the pressure quotient PQ_SP is close to 1, the setpoint of the reduced throttle area AR_RED_SP_1 is delayed with a first order time delay. Otherwise in fact of the negative infinite slope of the psi function the jitter of the throttle can not avoid. In this way a moving mean setpoint AR_RED_SP_MMV_1 are calculated.

The setpoint for MAF_THR has to decrease with FLOW_CPS, the air coming from canister purge valve.

AR_RED_SP_1=(MAF_KGH_SP – FLOW_CPS) / (PSI_SP * MAF_MDL_CON_1* PUT_IVS)

(Caclulation AR_RED_SP_PQ_MAX at C_PQ_MAX for charger TPSTransition Control)

AR_RED_SP_PQ_MAX_1 = (MAF_KGH_SP – FLOW_CPS)
/ (PSI_SP_TPS_PQ_SP_MAX * MAF_MDL_CON_1* PUT_IVS)

If LC_AR_RED_FIL_ENA = 1 **then**

PQ_SP_MIN = MIN(PQ, PQ_SP)

CRLC_AR_RED_SP = IP_CRLC_AR_RED_SP(PQ_SP_MIN)

$AR\_RED\_SP\_MMV\_1_{N} = AR\_RED\_SP\_MMV\_1_{N-1} * (1 - CRLC\_AR\_RED\_SP) + AR\_RED\_SP\_1_{N} * CRLC\_AR\_RED\_SP$

(Caclulation AR_RED_SP_PQ_MAX at C_PQ_MAX for charger TPSTransition Control)

$AR\_RED\_SP\_PQ\_MAX\_MMV\_1_{N} = AR\_RED\_SP\_PQ\_MAX\_MMV\_1_{N-1} * (1 - CRLC\_AR\_RED\_SP) + AR\_RED\_SP\_PQ\_MAX\_1_{N} * CRLC\_AR\_RED\_SP$

Else

$AR\_RED\_SP\_MMV\_1_{N} = AR\_RED\_SP\_1_{N}$

(Caclulation AR_RED_SP_PQ_MAX at C_PQ_MAX for charger TPS Transition Control)

$AR\_RED\_SP\_PQ\_MAX\ MMV\_1_{N} = AR\_RED\_SP\_PQ\_MAX\ \_1_{N}$

Endif

For the correction of steady state model errors the additive and multiplicative adaptation corrections AR_RED_AD_ADD and AR_RED_AD_FAC_COR and the relative I-value AR_RED_DIF_I_REL of the area reduced controller are included in the inverse model. AR_RED_DIF_I_REL is delayed with a first order time delay.

AR_RED_SP_MMV = [(AR_RED_SP_MMV_1 – AR_RED_AD_ADD) /
(1 + AR_RED_AD_FAC_COR + AR_RED_DIF_I_REL_MMV)]
* (1 / FAC_AR_RED_MAF)

(Caclulation AR_RED_SP_PQ_MAX at C_PQ_MAX for charger TPS Transition Control)

AR_RED_SP_PQ_MAX_MMV =
[(AR_RED_SP_PQ_MAX_MMV_1 – AR_RED_AD_ADD) /
(1 + AR_RED_AD_FAC_COR + AR_RED_DIF_I_REL_MMV)]
* (1 / FAC_AR_RED_MAF)

With the setpoint of the reduced area at the throttle the basic throttle position setpoint is determined. The interpolation table exists already in the intake manifold model. (Caclulation AR_RED_SP_PQ_MAX at C_PQ_MAX for charger TPS Transition Control)

If LV_AT= 1 **then**

TPS_SP_MDL_BAS = IP_TPS_SP_MDL_BAS_AT (AR_RED_SP_MMV)

(Throttle Setpoint required to give PQ = C_PQ_SP_MAX)

TPS_SP_PQ_MAX = IP_TPS_SP_MDL_BAS_AT(AR_RED_SP_PQ_MAX_MMV)

Else

TPS_SP_MDL_BAS = IP_TPS_SP_MDL_BAS_MT (AR_RED_SP_MMV)

TPS_SP_PQ_MAX = IP_TPS_SP_MDL_BAS_MT(AR_RED_SP_PQ_MAX_MMV)

Endif

MAF_SP_RATIO = Min(MAF_SP / MAF_MAX_COR , IP_MAF_SP_RATIO_MAX (N_32, PV_AV))

Remark :
If LV_CRU_ACT = 1, PV_CRU is used for MAF_SP_RATIO instead of PV_AV

The maximum value for the throttle position setpoint is depending on the engine speed N_32 and the setpoint of the mass air flow MAF_SP. This value can not be exceeded.

<u>In case of charger configuration</u> at ignition key off (LV_IGK = 0) and the Engine speed is below the calibratable threshold C_TPS_CLOSE_N_THD, the throttle position is limited to the calibratable value C_TPS_SP_IGK_OFF.

If LC_TCHA_CONF = 0

Then *(non turbo charger)*

If LV_AT = 0

Then TPS_SP_MDL_MAX
= Max(IP_TPS_SP_MDL_MAX_PV_MT(N_32,PV_AV),TPS_SP_MDL_BAS))

Else TPS_SP_MDL_MAX
= Max(IP_TPS_SP_MDL_MAX_PV_AT(N_32,PV_AV),TPS_SP_MDL_BAS))

Endif

Remark :
If LV_CRU_ACT = 1, PV_CRU is used for TPS_SP_MDL_MAX_PV_AT/MT calculation instead of PV_AV.

Else *(turbo charger)*

If LV_IGK = 1 **Or** N_32 > C_TPS_CLOSE_N_THD

Then TPS_SP_MDL_MAX = IP_TPS_SP_MDL_MAX(N_32, MAF_SP)

Else TPS_SP_MDL_MAX = C_TPS_SP_IGK_OFF

Endif

Endif

As long as MAF_SP_RATIO exceeds the threshold C_PQ_SP_MAX, the basic throttle position setpoint TPS_SP_MDL_BAS_COR is calculated as a linear function of the ratio of MAF_SP and MAF_MAX_COR. In this way it is possible to get a full open throttle position depending on the mass air flow setpoint.
During this time the threshold C_PQ_SP_MAX is needed as the correlation point of MAF_SP_RATIO. The basic throttle position setpoint TPS_SP_MDL_BAS is calculated with PQ_SP, which also is limited by C_PQ_SP_MAX.
As soon as MAF_SP_RATIO < C_PQ_SP_MAX the value TPS_SP_MDL_BAS is updated with the actual values.

For charged engines with LC_TPS_SP_MDL_PI_CTL = 1 a TPS-transition function can be activated to improve the throttle control at transition from PQ_SP = C_PQ_SP_MAX to natural boost using a PI controller to match MAP to MAP_SP.

If (LC_TPS_SP_MDL_PI_CTL = 0) *(Old Transition Function Active)*

If (LC_TPS_FIL_ENA = 1) **and** (MAF_SP_RATIO >= C_PQ_SP_MAX)

Then

TPS_SP_MDL_BAS_COR = TPS_SP_MDL_BAS +

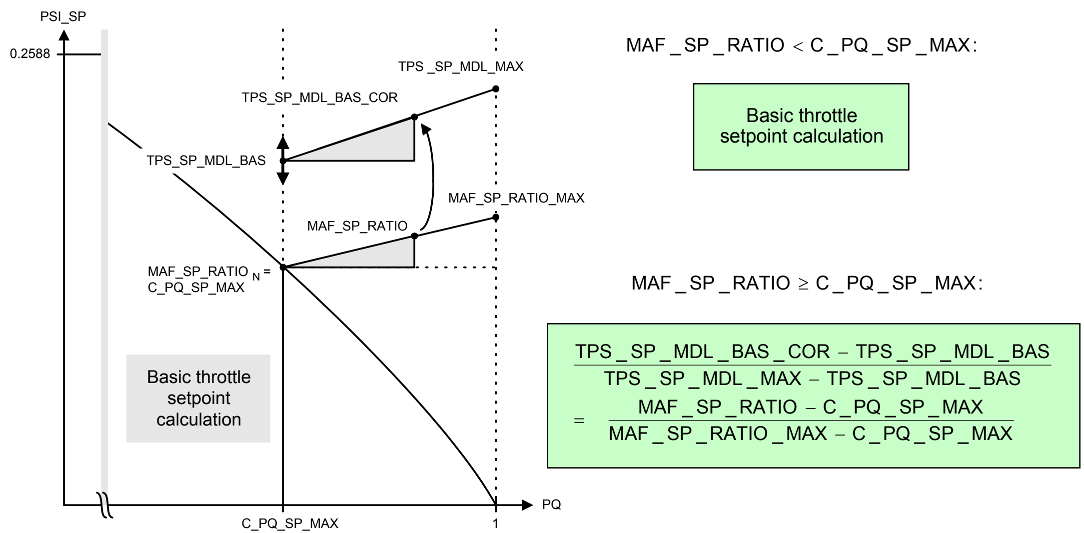
((MAF_SP_RATIO - C_PQ_SP_MAX)/
(1 - C_PQ_SP_MAX))*
(TPS_SP_MDL_MAX - TPS_SP_MDL_BAS)

If $TPS\_SP\_MDL\_BAS\_COR_n > TPS\_SP\_MDL\_BAS\_COR\_FIL_{n-1}$

Then $TPS\_SP\_MDL\_BAS\_COR\_FIL_n = TPS\_SP\_MDL\_BAS\_COR_n$

Else $TPS\_SP\_MDL\_BAS\_COR\_FIL_n$
$= TPS\_SP\_MDL\_BAS\_COR\_FIL_{n-1} + (TPS\_SP\_MDL\_BAS\_COR_n -$
$TPS\_SP\_MDL\_BAS\_COR\_FIL_{n-1}) * C\_CRLC\_TPS\_SP\_MDL\_BAS\_COR$

Else TPS_SP_MDL_BAS_COR_FIL = TPS_SP_MDL_BAS



$$\frac{TPS\_SP\_MDL\_BAS\_COR - TPS\_SP\_MDL\_BAS}{TPS\_SP\_MDL\_MAX - TPS\_SP\_MDL\_BAS} = \frac{MAF\_SP\_RATIO - C\_PQ\_SP\_MAX}{MAF\_SP\_RATIO\_MAX - C\_PQ\_SP\_MAX}$$

C.2.4 **Else** *(New Transition controller function activated)*

If PQ_SP > C_PQ_SP_MAX *(Transition controller becomes active)*

Then

TPS_SP_MDL_MIN_TMP = MIN (TPS_SP_PQ_MAX , TPS_SP_MDL_BAS)

IF LV_MAP_SP_THR = 1 *(Flag for fully boosted operation, so force throttle open using an artificial control difference)*

Then

MAP_DIF_TPS_SP_CTL = C_MAP_DIF_TPS_SP_MAX_OPEN

Else

(control difference for transition controller)

MAP_DIF_TPS_SP_CTL = MAP_SP – MAP

End

(Calculate transition controller terms)

TPS_SP_MDL_P_CTL = IP_P_CTL_TPS_SP_MDL (N , MAP_DIF_TPS_SP_CTL) * MAP_DIF_TPS_SP_CTL

TPS_SP_MDL_I_CTL_TMP =

IP_I_CTL_TPS_SP_MDL(N, MAP_DIF_SP_TPS_CTL) * MAP_DIF_TPS_SP_CTL) + TPS_SP_MDL_I_CTL(n-1)

(Anti windup logic for I Controller Term)

If (TPS_SP_MDL_MIN_TMP + TPS_SP_MDL_P_CTL + TPS_SP_MDL_I_CTL_TMP) > TPS_SP_MDL_MAX

Then

TPS_SP_MDL_I_CTL = TPS_SP_MDL_I_CTL (n-1)

Else

TPS_SP_MDL_I_CTL = TPS_SP_MDL_I_CTL_TMP

End if

(final throttle postion, before MAX limitation, TPS_SP_MDL_BAS_COR_FIL used as temporary variable)

TPS_SP_MDL_BAS_COR_FIL= TPS_SP_MDL_MIN_TMP + TPS_SP_MDL_P_CTL +TPS_SP_MDL_I_CTL

Else *(Not in transition region, force transition controller terms to zero)*

TPS_SP_MDL_BAS_COR_FIL = TPS_SP_MDL_BAS

TPS_SP_MDL_I_CTL = TPS_SP_MDL_P_CTL = 0

End

End

C.2.5 Throttle Position Setpoint Limitation

The throttle position setpoint TPS_SP_MDL results in the limitation of the corrected basic throttle position setpoint and the maximum value.

TPS_SP_MDL = MIN (TPS_SP_MDL_BAS_COR_FIL, IP_TPS_SP_MDL_MAX_XXX)

With IP_TPS_SP_MDL_MAX_XXX =

IP_TPS_SP_MDL_MAX_PV_AT (for LC_TCHA_CONF = 0 and LV_AT = 1)
IP_TPS_SP_MDL_MAX_PV_MT (for LC_TCHA_CONF = 0 and LV_AT = 0)
IP_TPS_SP_MDL_MAX (for LC_TCHA_CONF = 1)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_PQ_SP_MAX	1	0...FFFFH	0...0.999985	1.52e-5	-
Pressure quotient threshold					
IP_TPS_SP_MAN	8*12	0...3FFFH	0...119.5	7.29e-3	°TPS
LDPM_N_32_IP_TPS_SP_MDL_MAX	8	0...FFH	0...8160	32	1/min
LDPM_PV_AV_IP_TPS_SP_MDL_MAX_PV	12	0...FFH	0...99.6	0.3906	%
Manual throttle position setpoint					
IP_CRLC_AR_RED_SP	1*8	0...FFH	0...0.9969	0.0039	-
LDP_PQ_SP_IP_CRLC_AR_RED_SP	8	0...FFFFH	0...0.99998474	1.52E-5	-
Correlation constant for the moving mean value calculation of AR_RED_SP					
IP_TPS_SP_MDL_BAS_AT	16	0...3FFFH	0...119.5	7.2941e-3	[°TPS]
LDPM_AR_RED_SP_IP_TPS_SP_MDL	16	0...FFFFH	0...58.59285	0.8941e-3	[cm²]
Throttle position setpoint for A/T(! inverse of IP_AR_RED_THR !)					
LC_AR_RED_FIL_ENA	1	0...1H	0...1	1	-
Default logical variable enabled for reduced throttle area filter					
LC_TPS_FIL_ENA	1	0...1H	0...1	1	-
Default logical variable enabled for throttle position setpoint filter					
LC_TPS_SP_MAN_ENA	1	0...1H	0...1	1	-
Flag to enable manual throttle position setting					
C_TPS_CLOSE_N_THD	1	00...FFH	0...8160	32	rpm
Engine speed threshold for closing throttle at engine shut off					
C_TPS_SP_IGK_OFF	1	0...3FFFH	0..119.5	7.29e-3	°TPS
Maximum throttle position set point at ignition key off;Typical value 5°Tps					
IP_PUT_ DIF_PQ_SP_1	6*6	0...FFFFH	0...5434	0.083	hPa
LDP_N_32_IP_PUT_ DIF_PQ_SP_1	6	0...FFH	0...8160	32	1/min
LDP_AMP_IP_PUT_ DIF_PQ_SP_1	6	0...FFFFH	0...5434	0.083	hPa
PUT difference to basic charge air pressure to activate wide open throttle					
C_MAP_SP_HYS_CHRG	1	0 ... FFFFH	0...5434	0.083	hPa
Hysteresis in setpoint of manifold pressure in charger					
C_CRLC_TPS_SP_MDL_BAS_COR	1	0 ... FFH	0...0.99609	0.0039	-
Filtering factor for TPS_SP_MDL_BAS_COR					
IP_TPS_SP_MDL_MAX	8*4	0...3FFFH	0...119.5	7.29e-3	°TPS
LDPM_N_32_IP_TPS_SP_MDL_MAX	8	0...FFH	0...8160	32	1/min
LDP_MAF_SP_IP_TPS_SP_MDL_MAX	4	0...FFFFH	0...1.389E+3	0.02119478	mg/stk
Maximum throttle position setpoint at ignition key on for TCI					
IP_TPS_SP_MDL_MAX_PV_MT	8*12	0...3FFFH	0...119.5	7.29e-3	°TPS
LDPM_N_32_IP_TPS_SP_MDL_MAX	8	0...FFH	0...8160	32	1/min
LDPM_PV_AV_IP_TPS_SP_MDL_MAX_PV	12	0...FFH	0...99.6	0.3906	%
Maximum throttle position setpoint at ignition key on for non TCI and M/T					
IP_TPS_SP_MDL_MAX_PV_AT	8*12	0...3FFFH	0...119.5	7.29e-3	°TPS
LDPM_N_32_IP_TPS_SP_MDL_MAX	8	0...FFH	0...8160	32	1/min
LDPM_PV_AV_IP_TPS_SP_MDL_MAX_PV	12	0...FFH	0...99.6	0.3906	%
Maximum throttle position setpoint at ignition key on for non TCI and A/T					
IP_CRLC_PQ_SP_AR_RED_PUT	4	0...FFFFH	0...0.99998474	1.52588E-5	-
LDP_PQ_IP_CRLC_PQ_SP_AR_RED_PUT	4	0...FFFFH	0...0.99998474	1.52588E-5	-
Filter constant for filtered PQ_SP for INSY controller					
LC_PUT_IVS_PUT_PRED	1	0...1H	0...1	1	-
Logical variable for using PUT_PRED for the PUT_IVS calculation					

IP_MAF_SP_RATIO_MAX	8*12	0...FFFFH	0...0.999885	1.52e-5	-
LDP_N_32_IP MAF_SP_RATIO_MAX	8	0...FFH	0...8160	32	1/min
LDP_PV_AV_IP MAF_SP_RATIO_MAX	12	0...FFH	0...99.6	0.3906	%
Maximum MAF_SP_RATIO dependent on N_32 and PV_AV					
LC_TPS_SP_MDL_PI_CTL	1	0... 1H	0... 1	1	[-]
The logical constant which is used for the calculation of the TPS_SP_MDL through the PI Controller					
IP_I_CTL_TPS_SP_MDL	6*8	0... FFFFH	0... 0.99998474121	15.2588e-6	[°TPS/hPa]
LDPM_N_TPS_SP_MDL_CTL	6	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAP_DIF_TPS_SP_MDL_CTL	8	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
I controller gain for the TPS_SP_MDL calculation					
IP_P_CTL_TPS_SP_MDL	6*8	0... FFFFH	0... 0.99998474121	15.2588e-6	[°TPS/hPa]
LDPM_N_TPS_SP_MDL_CTL	6	0... 1FE0H	0... 8160	1	[rpm]
LDPM_MAP_DIF_TPS_SP_MDL_CTL	8	0... FFFFH	-2717.04145876 ...2716.95854124	0.0829175	[hPa]
P controller gain for the TPS_SP_MDL calculation					
C_MAP_DIF_TPS_SP_MAX_OPEN	1	0..7FFF	0..2716.91708	0.0829175	[hPa]
Constant to define the difference MAP_DIF_TPS_SP_CTL when the throttle should be opened to the maximum					
IP_TPS_SP_MDL_BAS_MT	16	0...3FFFH	0...119.5	7.2941e-3	[°TPS]
LDPM_AR_RED_SP_IP_TPS_SP_MDL	16	0...FFFFH	0...58.59285	0.8941e-3	[cm²]
Throttle position setpoint for M/T(! inverse of IP_AR_RED_THR !)					

Index tables ID_MAF_FAC_OFS and ID_MAF_FAC_SLOP are defined in module “ intake manifold model”.

Interpolation table IP_PRS_LOSS_AIC is defined in module “Pressure decrease through the air cleaner”.

C.3 Throttle setpoint calculation at start

Output data:

Name	Mode	Hex. limits	Phys. Limit	Resol.	Unit
TPS_SP_ST	O/V	0...3FFFH	0...119.5	0.0073	°TPS
TPS setpoint at start					
TPS_REL_ST	V	0...FFFFH	0...100	0.0015	%
relative TPS setpoint at start					

Input data:

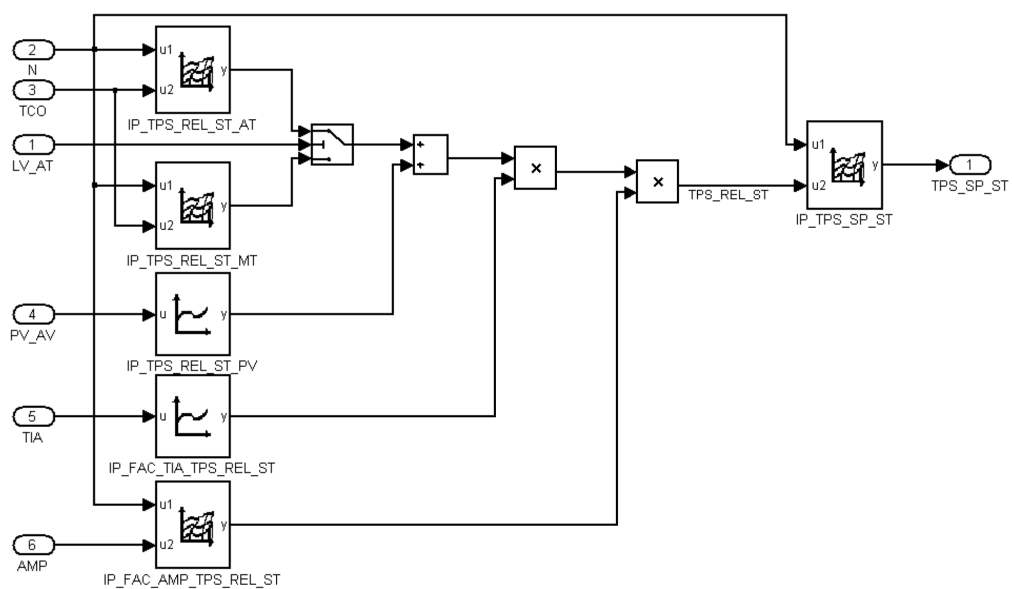
LV_AT	N	TCO	PV_AV
TIA	AMP	LV_ST	

FUNCTION DESCRIPTION:

General information:

During start a separate TPS setpoint at start is calculated. There are two basic maps (dependent on N and TCO) for manual and automatic transmission, which deliver the percentage of the opening of the throttle. This basic value is corrected by an offset dependent on the pedal value and factors dependent on intake air temperature TIA and ambient pressure AMP. The applied TPS setpoint is determined by a map dependent on the percentage and the engine speed N. This map can be used to correct the non linear behavior of the throttle (reduced area) or the behavior of the complete intake manifold.

Signal flow diagram:



Application conditions:

Initialisation: TPS_SP_ST = 0

Recurrence: 10 ms

Activation: LV_ST = 1

Formula section:

If LV_AT = 1 **then**

TPS_REL_ST_INT = IP_TPS_REL_ST_AT ! Basic map for automatic transmission

else

TPS_REL_ST_INT = IP_TPS_REL_ST_MT ! Basic map for manual transmission

endif

TPS_REL_ST = (TPS_REL_ST_INT
+ IP_TPS_REL_ST_PV) !TPS correction at start via PV_AV
* IP_FAC_TIA_TPS_REL_ST ! Air temperature correction TPS_SP at start!
* IP_FAC_AMP_TPS_REL_ST ! Altitude correction TPS_SP at start

TPS_SP_ST = IP_TPS_SP_ST

Calibration data:

Name	Dim	Hex. limits	Phys. Limit	Resol.	Unit
IP_TPS_REL_ST_AT	8x8	0...FFFFH	0...100	0.0015	%
LDPM_N_1_THRO	8	0...1FE0H	0...8160	1	1/min
LDPM_TCO_1_THRO	8	0...FEH	-48...142.5	0.75	°C
Specified relative TPS setpoint at start					
IP_TPS_REL_ST_MT	8x8	0...FFFFH	0...100	0.0015	%
LDPM_N_1_THRO	8	0...1FE0H	0...8160	1	1/min
LDPM_TCO_1_THRO	8	0...FEH	-48...142.5	0.75	°C
Specified relative TPS setpoint at start					
IP_TPS_REL_ST_PV	1x6	0...FFFFH	0...100	0.0015	%
LDP_PV_AV_IP_TPS_SP_ST_PV	6	0...FFH	0...99.6	0.39	%
correction of the ralative TPS setpoint at start via driver's demand					
IP_FAC_TIA_TPS_REL_ST	1x6	0...FFH	0...1.99	0.008	-
LDP_TIA_ IP_FAC_TIA_TPS_SP_ST	6	0...FEH	-48...142.5	0.75	°C
TIA correction of the relative TPS setpoint at start					
IP_FAC_AMP_TPS_REL_ST	8x6	0...FFH	0...1.99	0.008	-
LDPM_N_1_THRO	8	0...1FE0H	0...8160	1	1/min
LDP_AMP_IP_TPS_SP_ST_AMP	6	0...FFFFH	0...5434	0.083	hPa
Altitude correction of the relative TPS setpoint at start					
IP_TPS_SP_ST	8x12	0...FFFFH	0...119.9982	0.0018	°TPS
LDPM_N_1_THRO	8	0...1FE0H	0...8160	1	1/min
LDP_TPS_REL_ST_IP_TPS_SP_ST	12	0...FFFFH	0...100	0.0015	%
TPS setpoint at start					

C.4 Throttle Position Setpoint Selection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TPS_SP	V/O	0...3FFFh	0...119.5	0.0073	°TPS
Throttle position setpoint					
TPS_SP_SEL	V	0...3FFFh	0...119.5	0.0073	°TPS
Selected throttle position setpoint before limitation					
LV_TPS_SP_TRA_ST_MDL	V	0...1	0...1	1	-
Logical variable indicating transient phase of throttle position setpoint after start					
STATE_TPS_SP_TRA_DIR	V	0h 1h 2h	INIT UP DOWN	1	-
State variable indicating the direction of the transient phase					

Input data:

TPS_LIH	TPS_SP_PWL	TPS_SP_AD	TPS_SP_EXT_ADJ
TPS_SP_ST	TPS_SP_ES	TPS_SP_MDL	LV_TPS_AD_REQ
LV_TPS_SP_LIH	LV_TPS_PWL	LV_TPS_AD_ACT	LV_TPS_SP_EXT_ADJ
LV_ST	LV_ES	LV_PL	TCO
N			

FUNCTION DESCRIPTION:

This module describes the calculation of the throttle position setpoint TPS_SP which is used as an input variable in the module "ETC Position Controller".

The module consists of two blocks (see figure 1): In a first step the setpoint is selected depending on the engine state, in a second step the selected value is limited. This limitation acts as protection so that the throttle blade does not hit the mechanical limits with too high energy. Otherwise the throttle body could be damaged.

The setpoint before limitation TPS_SP_SEL is selected from the following variables:

- Throttle position setpoint when the position controller is switched off: TPS_LIH
- Throttle position setpoint during the post operating phase: TPS_SP_PWL
- Throttle position setpoint during adaptation: TPS_SP_AD
- Throttle position setpoint in case of external activation: TPS_SP_EXT_ADJ
- Throttle position setpoint at start: TPS_SP_ST
- Throttle position setpoint for engine stopped: TPS_SP_ES
- Basic throttle position setpoint TPS_SP_MDL, output of the inverse intake manifold model

From TPS_SP_SEL the final setpoint TPS_SP is generated by means of a limitation function which is including a limitation by absolute values.

The setpoint is determined every 10 ms, except during adaptation (5 ms).

Sign al flow diagram:

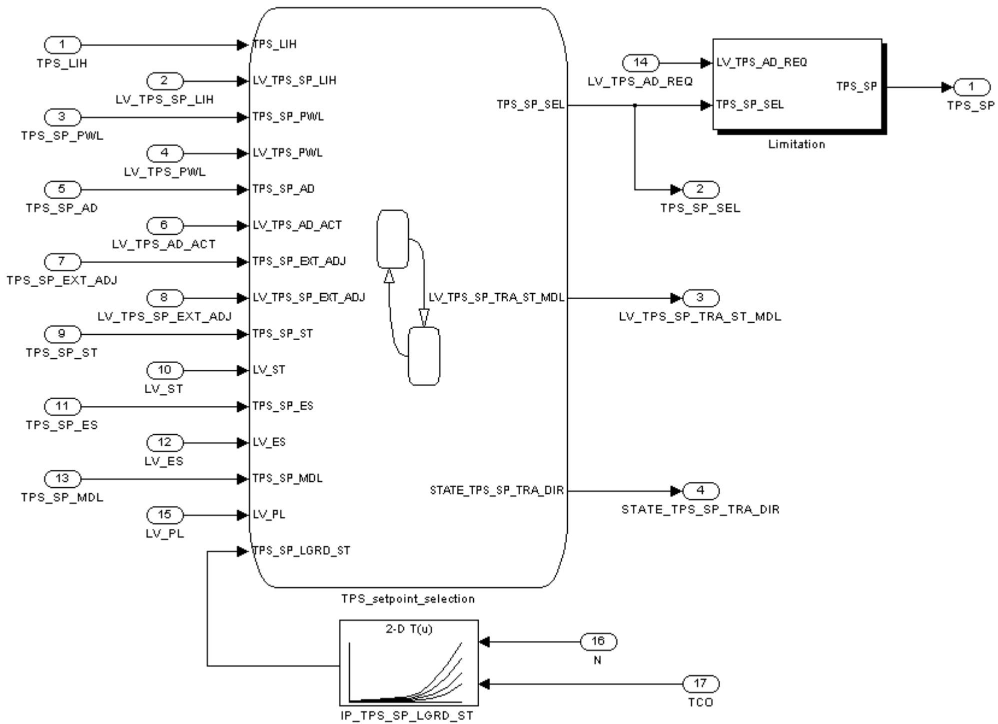


Figure 6: Throttle position setpoint selection and limitation.

Application conditions:

Activation:	at every engine state	
Deactivation:	-	
Initialisation:	*at reset:*	LV_TPS_SP_TRA_ST_MDL = 0
		STATE_TPS_SP_TRA_DIR = INIT
		TPS_SP = TPS_CTL = TPS_SP_SEL = 0;
	at exit start:	LV_TPS_SP_TRA_ST_MDL = 1
Update rate:	10 ms and 5 ms during adaptation, if TPS_SP_AD is active ! (see chapter “Throttle position setpoint during adaptation”)	

C.4.1 Throttle position setpoint selection

Description:

For the selection logic see figure 2.

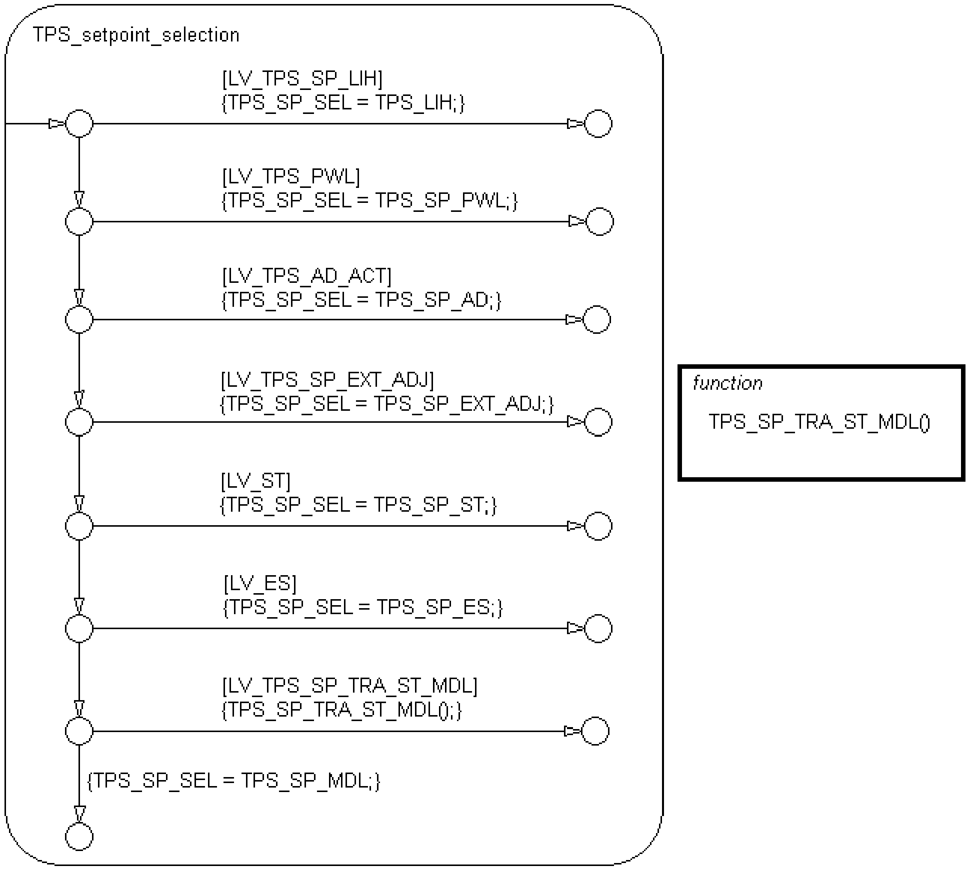


Figure 7: Throttle position setpoint selection block.

Throttle position setpoint for ETC controller switched off

In case of the ETC controller switched off the TPS setpoint is set to the limp home position TPS_LIH.

Throttle position setpoint during post operating phase

To avoid that for Key-Off condition the throttle actuator is without control current and set to limp home position, a throttle position setpoint during the post operating phase is available.

Throttle position setpoint during adaptation

During TPS adaptation the throttle position setpoint is set from the adaptation module.

Throttle position setpoint by external device

In case of request by an external device the throttle position setpoint is set by the value TPS_SP_EXT_ADJ.

Throttle position setpoint during start

During the engine state "Start" (LV_ST = 1) the throttle position setpoint is determined by the special variable TPS_SP_ST.

Throttle position setpoint for engine stopped

During the engine state "Engine stopped" (LV_ES = 1) the throttle position setpoint is determined by the special variable TPS_SP_ES.

Throttle position setpoint for transient phase after start

When the start phase is finished (LV_ST = 1 -> 0), then after a segment delay counter (to allow stabilization of the MAF_SP_RATIO during the after-start transition) a transient phase is following (LV_TPS_SP_TRA_ST_MDL = 1) where the pre-controlled setpoint from the start phase is approaching the TPS_SP_MDL from Intake Manifold Model.

Formula section:

if(1) LV_TPS_SP_LIH = 1

then(1) *(Throttle position setpoint for limp-home is requested)*

TPS_SP_SEL = TPS_LIH

else(1) **if** LV_TPS_PWL = 1

then *(Throttle position setpoint for power-latch is requested)*

TPS_SP_SEL = TPS_SP_PWL

else **if** LV_TPS_AD_ACT = 1

then *(Throttle position setpoint for TPS Adaptation is requested)*

TPS_SP_SEL = TPS_SP_AD

```
else if   LV_TPS_SP_EXT_ADJ = 1
     then (External Throttle position setpoint is requested)
          TPS_SP_SEL = TPS_SP_EXT_ADJ
     else if   LV_ST = 1
          then (Throttle position setpoint for engine start)
               TPS_SP_SEL = TPS_SP_ST
          else if   LV_ES = 1
               then (Throttle position setpoint for engine start)
                    TPS_SP_SEL = TPS_SP_ES
               else if   LV_TPS_SP_TRA_ST_MDL = 1
                    then (Transient Phase after Start)
                         call function TPS_SP_TRA_ST_MDL()
                    else (Setpoint from Intake Manifold Model)
                         TPS_SP_SEL = TPS_SP_MDL
endif(1)
```

Function TPS_SP_TRA_ST_MDL():

```
start delay timer from ID_SEG_DLY_ST_TPS_SP_TRA
if(1)      delay timer has run out
then(1)    (acitvate determination of LGRD)
    if(2)      |TPS_SP_SEL - TPS_SP_MDL| > IP_TPS_SP_LGRD_ST
      and      LV_PL = 0
    then(2)    (conditions for Transient Phase fulfilled)
               if(3)      TPS_SP_SEL > TPS_SP_MDL
                 and      STATE_TPS_SP_TRA_DIR != UP
               then(3)    (determination of ramp DOWN)
                          TPS_SP_SEL -= IP_TPS_SP_LGRD_ST
                          STATE_TPS_SP_TRA_DIR = DOWN
               else(3)    (determination of ramp UP)
                          if(4)   TPS_SP_SEL <= TPS_SP_MDL
                             and  STATE_TPS_SP_TRA_DIR != DOWN
                          then(4) TPS_SP_SEL += IP_TPS_SP_LGRD_ST
                                  STATE_TPS_SP_TRA_DIR = UP
                          else(4) (Transient Phase finished)
                                  TPS_SP_SEL = TPS_SP_MDL
                                  STATE_TPS_SP_TRA_DIR = INIT
                                  LV_TPS_SP_TRA_ST_MDL = 0
```

```
                    endif(4)
                endif(3)
        else(2)     (difference between TPS_SP_SEL and TPS_SP_MDL is small)
                    TPS_SP_SEL = TPS_SP_MDL
                    STATE_TPS_SP_TRA_DIR = INIT
                    LV_TPS_SP_TRA_ST_MDL = 0
        endif(2)
    endif(1)
end function
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TPS_SP_LGRD_ST	8 x 8	1...3FFFH	0.0073 ... 119.5	0.0073	°TPS/10 ms
LDP_N_IP_TPS_SP_LGRD_ST	8	0...1FE0H	0...8160	1	rpm
LDP_TCO_IP_TPS_SP_LGRD_ST	8	0...FEH	-48...142.5	0.75	°C
admissible gradient of throttle position setpoint for transient phase after start					
ID_SEG_DLY_ST_TPS_SP_TRA	1 x 4	0...FFH	0 ... 255	1	-
LDP_TCO_ID_SEG_DLY_ST_TPS_SP	4	0...FEH	-48...142.5	0.75	°C
segment delay counter after start before applying TP_SP_TRA logic to allow stable MAF_SP_RATIO					

C.4.2 Throttle position setpoint limitation

Description:

The task of this function is a protection mechanism for the throttle blade. Therefore, it must be evaluated after the TPS setpoint selection.

For adjusting the setpoint to the physics of the throttle the variable TPS_SP_SEL is limited by absolute values, for the non adapted throttle (LV_TPS_AD_REQ = 1 remains set after adaptation) the values indicated by "(_TPS_AD_REQ)" are used.

The throttle position setpoint may not fall below the value C_TPS_SP_MIN(_TPS_AD_REQ) or exceed C_TPS_SP_MAX(_TPS_AD_REQ), see figure 3.

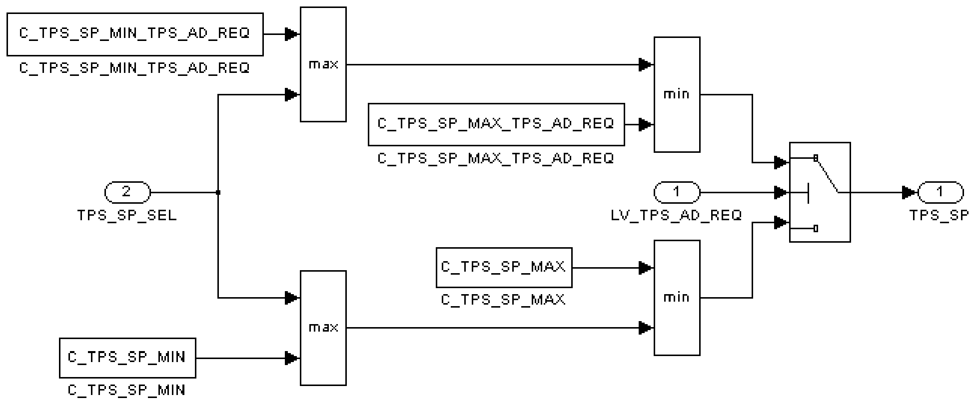


Figure 3: Throttle position setpoint limitation.

Formula section:

if (TPS_SP_SEL > C_TPS_SP_MAX(_TPS_AD_REQ))
then TPS_SP = C_TPS_SP_MAX(_TPS_AD_REQ)
else **if** TPS_SP_SEL < C_TPS_SP_MIN(_TPS_AD_REQ)
then TPS_SP = C_TPS_SP_MIN(_TPS_AD_REQ)
else TPS_SP = TPS_SP_SEL
endif
endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
C_TPS_SP_MIN	1	0...3FFFH	0...119.5	0.0073	°TPS
Lower limit of throttle position setpoint					
C_TPS_SP_MIN_TPS_AD_REQ	1	0...3FFFh	0...119.5	0.0073	°TPS
Lower limit of throttle position setpoint without throttle adaptation					
C_TPS_SP_MAX	1	0...3FFFh	0...119.5	0.0073	°TPS
Upper limit of throttle position setpoint					
C_TPS_SP_MAX_TPS_AD_REQ	1	0...3FFFh	0...119.5	0.0073	°TPS
Upper limit of throttle position setpoint without throttle adaptation					

C.5 Throttle Position Setpoint Selection (Appl. Inc.)

C.5.1 Throttle position setpoint for engine stopped

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TPS_SP_ES	V/O	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint for engine stopped					

Input data:

LV_ES	TCO		

FUNCTION DESCRIPTION:

General information:

For engine stopped TPS_SP_ES is generated from a TCO dependent map. The TPS_SP_ES value must be chosen with respect to the expected TPS_SP_ST. A sensible choice diminishes the transient time when the TPS_AV is approaching TPS_SP_ST.

Application conditions:

Initialisation: -

Recurrence: 10 ms

Activation: in every engine state

Formula section:

if LV_ES

then TPS_SP_ES = IP_TPS_SP_ES

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TPS_SP_ES	1	0...3FFFH	0...119.5	0.0073	°TPS
LDPM_TCO_1_THRO	8	0...FEH	-48...142.5	0.75	°C
Throttle position setpoint for engine stopped					

C.5.2 Throttle position setpoint by external device

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TPS_SP_EXT_ADJ	V/O	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint of external device					
LV_TPS_SP_EXT_ADJ	V/O	0...1H	0...1	1	-
Bit active for external input of throttle position setpoint					
TPS_SP_EXT_ADJ_1	V	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint of external device about PV_AV					
LV_TPS_SP_EXT_ADJ_1	V	0...1H	0...1	1	-
Bit active for external input of throttle position setpoint for PV_AV setpoint					

Input data:

LV_ES	LV_IGK	PV_AV	TPS_EXT_ADJ
LV_TPS_EXT_ADJ	LV_ENG_OFF_DMF_TPS_ISA	N_32	

Description:

For different tasks during calibration phase and for service procedures at the workshops, it is necessary to implement an external adjustment at throttle position.
When an engine is shut-off due to DMF oscillation, external adjustment can be applied as long as engine running (LV_ES = 0).

Application conditions:

Initialisation: 0

Recurrence: 10 ms

Activation: LV_IGK == 1

Formula section:

{C_TPS_SP_EXT_ADJ_ENA = 0 : no external adjustment}

{C_TPS_SP_EXT_ADJ_ENA = 1 : Only manual setpoint}

{C_TPS_SP_EXT_ADJ_ENA = 2 : only PV_AV depended setpoint}

{C_TPS_SP_EXT_ADJ_ENA = 3 : adjustment about PV_AV and manual setpoint}

if **(** PV_AV > C_PV_AV_EXT_ADJ_MIN **)** **and**
LV_ES **and**
(C_TPS_SP_EXT_ADJ_ENA == 2 **or**
C_TPS_SP_EXT_ADJ_ENA == 3 **)**
then LV_TPS_SP_EXT_ADJ_1 = 1
TPS_SP_EXT_ADJ_1 = IP_TPS_SP_EXT_ADJ
else LV_TPS_SP_EXT_ADJ_1 = 0
TPS_SP_EXT_ADJ_1 = 0
endif


```
if      LV_TPS_SP_EXT_ADJ_1            or
        C_TPS_SP_EXT_ADJ_ENA == 1      or
        C_TPS_SP_EXT_ADJ_ENA == 3      or
        LV_TPS_EXT_ADJ = 1     (for actuator test with service tester)     or
        { LV_ENG_OFF_DMF_TPS_ISA = 1 and LV_ES = 0 }
then    LV_TPS_SP_EXT_ADJ = 1
else    LV_TPS_SP_EXT_ADJ = 0
endif

if      LV_TPS_SP_EXT_ADJ = 1
then    if      LV_TPS_EXT_ADJ = 1
        then    TPS_SP_EXT_ADJ = TPS_EXT_ADJ (for actuator test with service tester)
        else
                if      LV_ENG_OFF_DMF_TPS_ISA = 1
                then
                        TPS_SP_EXT_ADJ = IP_TPS_SP_ENG_OFF_DMF
                Else
                        TPS_SP_EXT_ADJ = C_TPS_SP_AS_MAN +
                                         TPS_SP_EXT_ADJ_1
                endif
        endif
else    TPS_SP_EXT_ADJ = 0
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_TPS_SP_AS_MAN	1	0...3FFFH	0...119,5	0,0073	°TPS
Manual throttle setpoint					
IP_TPS_SP_EXT_ADJ	1x6	0...3FFFH	0...119,5	0,0073	°TPS
LDP_PV_AV__TPS_SP_EXT	6	0...FFH	0...99,6	0,39	%
External adjustment setpoint					
C_PV_AV_EXT_ADJ_MIN	1	0...FFH	0...99,6	0,39	%
Min. threshold for external ajustment about PV_AV					
C_TPS_SP_EXT_ADJ_ENA	1	0...3H	0...3	1	-
Variable enabled for throttle position setpoint manual or about PV_AV					
IP_TPS_SP_ENG_OFF_DMF	1x4	0...3FFFH	0...119.5	0.0073	°TPS
LDP_N_32_TPS_SP_ENG_OFF_DMF	4	0...FFH	0...8160	32	rpm
Throttle position setpoint at engine shut-off due to DMF oscillation					

C.5.3 Throttle position setpoint for ETC limp-home

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_TPS_SP_LIH	V/O	0...1H	0...1	1	-
ETC setpoint for the limp-home position is active					

Input data:

LV_MTC_CUR_OFF	LV_TPS_SP_LIH_REQ		

Application conditions:

Initialisation: LV_TPS_SP_LIH = 1

Recurrence: 10 ms

Activation: in every engine state

Formula section:

LV_TPS_SP_LIH = LV_MTC_CUR_OFF **or** LV_TPS_SP_LIH_REQ

C.5.4 Throttle position setpoint during post operating phase

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
TPS_SP_PWL	V/O	0...3FFFH	0...119.5	0.0073	°TPS
Throttle position setpoint during post operating phase					
LV_TPS_PWL	V/O	0...1H	0...1	1	-
Bit for throttle position setpoint in post operating phase active					
T_TPS_PWL	-	0...FFFFH	0...655.35	0.01	s
Time after engine stop					

Input data:

LV_ES	LV_IGK	N_32	

General information:

To avoid that for ‚Power Down‘ the currentless throttle is set to the limp home position, for a tuneable time period a throttle position setpoint during the post operating phase is available.
A time counter is started on detection of ignition key OFF and N < 32 rpm (engine stopped). If the counter reaches its maximum C_T_MTC_ISA_OFF, the throttle is actuated to the setpoint C_TPS_SP_PWL_UP by the current TPS_SP and then de-energized. Thus, it is ensured that the throttle are closed until a defined time after ignition key OFF and engine stopped.

If the throttle is already deenergized when ignition key OFF is detected, it is not re-energized.

Application conditions:

Initialisation: *at reset and at transition igk off to on,*

LV_TPS_PWL = 0

Recurrence: 10ms, This module is calculated before the chapter “ETC position controller (application incidences)” !

Activation: in every engine state

Formula section:

```
if (0)      transition LV_IGK from 1 to 0
then(0)     LV_TPS_PWL = 1
            T_TPS_PWL = 0
endif(0)

if (1)      LV_TPS_PWL == 1
then(1)
   if(2)    LV_ES == 0
   then(2)  TPS_SP_PWL = IP_TPS_SP_PWL
            T_TPS_PWL = 0
   else (2)
            if (3)   (T_TPS_PWL < C_T_MTC_ISA_OFF)
            then(3)       TPS_SP_PWL = IP_TPS_SP_PWL
                          T_TPS_PWL = T_TPS_PWL + 1H
            else (3)
                   if (4)   (TPS_SP_PWL>= C_TPS_SP_PWL_UP)
                   then(4)     LV_TPS_PWL = 0
                   else(4)     TPS_SP_PWL = TPS_SP_PWL + C_TPS_SP_PWL_LGRD
                   endif(4)
            endif(3)
    endif(2)
endif(1)
```

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_T_MTC_ISA_OFF	1	0...FFH	0...255	1	s
Active time period for throttle position setpoint during post operating phase					
IP_TPS_SP_PWL	1	0...3FFFH	0...119.5	0.0073	°TPS
LDP_N_32_TPS_SP_PWL	4	0...FFH	0...8160	32	rpm
Throttle position setpoint during post operating phase					
C_TPS_SP_PWL_UP	1	0...3FFFH	0...119,5	0,0073	°TPS
Threshold opening throttle after power latch					
C_TPS_SP_PWL_LGRD	1	0...3FFFH	0...119,5	0,0073	°TPS/ 10ms
Gradient limitation TPS					

C.6 ETC position controller

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MTCPWM	V/O	8000h ... 7FFFh	-100...99.997	0.00305	%
Duty cycle of the MTC control					
TPS_DIF	V/O	C000h...3FFFh	-119.51...119.5	0.0073	°TPS
Position controller deviation					
TPS_DIF_MTC	V	C000h...3FFFh	-119.51...119.5	0.0073	°TPS
TPS value before PDT_1 transfer function					
TPS_FIL_MTC	V	C000h...3FFFh	-119.51...119.5	0.0073	°TPS
TPS value after PDT_1 transfer function					
KP_COR_MTC	V	0h ... 4000h	0 ... 1024	0.0625	% / °TPS
Controller amplification, battery voltage corrected					
FAC_VB_MTC	V	200h ... 1000h	0.5 ... 4	$9.766*10^{-4}$	-
Factor battery voltage correction					
TPS_MAX_MTC	V	0h...3FFFh	0 ... 119.5	0.0073	°TPS
Limitation value of input value, PT 1 transfer function					
TPS_SP_CTL	V	C000h...3FFFh	-119.5...119.5	0.0073	°TPS
Throttle position setpoint for MTC position controller					
TPS_OFS_CTL	V	C000h...3FFFh	-119.5...119.5	0.0073	°TPS
Offset for the compensation of the control deviation					
MTCPWM_LIM	V/O	8000h ... 7FFFh	-100...99.997	0.00305	%
Limited MTCPWM					
LV_MTCPWM_RED_REQ	V/O	0h ... 1h	0 … 1	1	-
Reduced MTCPWM limitation requested					

Input data:

LV_MTC_CUR_OFF	TPS_SP	TPS_ETC	TPS_LIH
TPS_LIH_INI	VB_MMV	MTCPWM_MMV	

C.6.1 Digital position controller

Description:

The position controller is carried out as a robust digital position controller. Calibration data are determined according to Siemens internal controller design algorithms. The position controller consists of a PD – controller in backwards direction (static amplification = 1) and a amplitude correction function PDT_1 as well as amplification factor Kp in forward direction.

To avoid additional delay between AD conversion and calculation it is necessary that the routines for calculation of TPS_ETC and MTCPWM are called directly after the AD conversion is finished. Output of MTCPWM starts immediately after its calculation.

Signal flow diagram:

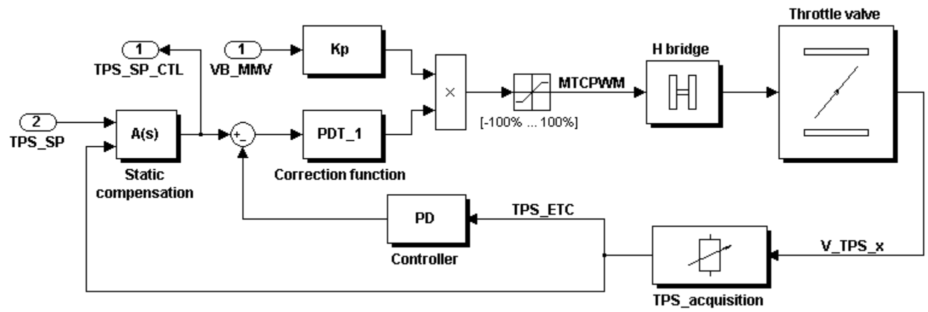


Figure 1: Electronic throttle control

Application conditions:

Initialisation:	tps_etc(k-1)	= TPS_LIH_INI
	mtc_ctl_1(k-1)	= mtc_ctl_3(k-1) = 0
	MTCPWM	= 0
Recurrence:	2 ms	
Activation:	see below	

Formula section:

{mtc_ctl_1/2/3 are 32 bit variables. They are introduced to do the calculations with higher resolution (6 bit, 0.00729 / 64 °TPS). }

if LV_MTC_CUR_OFF == 0

then *{Controller active}*

{Controller backward direction}

mtc_ctl_0(k) = TPS_ETC(k) + C_KDF_MTC * [TPS_ETC(k) – tps_etc(k-1)]

TPS_DIF_MTC = TPS_SP_CTL – mtc_ctl_0(k)

{Increase of the resolution, 6 bit}

mtc_ctl_1(k) = TPS_DIF_MTC

{Transfer function PDT$_1$}

mtc_ctl_2(k) = C_FAC_0_FIL_MTC * [mtc_ctl_1(k) - mtc_ctl_1(k-1)]
+ mtc_ctl_1(k-1)
+ C_FAC_1_FIL_MTC * [mtc_ctl_3(k-1) - mtc_ctl_1(k-1)]

mtc_ctl_3(k) = MAX(MIN(mtc_ctl_2(k), TPS_MAX_MTC), -TPS_MAX_MTC)

{Decrease of the resolution, 6 bit}

TPS_FIL_MTC = mtc_ctl_2(k)

{Controller amplification}

MTCPWM_CTL = KP_COR_MTC * TPS_FIL_MTC

{MTCPWM limitation}

MTCPWM = MAX(MIN(MTCPWM_CTL, MTCPWM_LIM), -MTCPWM_LIM)

else *{Controller inactive, throttle currentless}*

{Reset/Initialize the controller values}

tps_etc(k-1) = TPS_ETC(k)

mtc_ctl_1(k-1) = TPS_SP_CTL – TPS_ETC(k)

mtc_ctl_3(k-1) = MAX(MIN(mtc_ctl_1(k-1), TPS_MAX_MTC), -TPS_MAX_MTC)

MTCPWM = 0

endif

Update HW-Pin with new MTCPWM value

<u>Signal flow diagram:</u>

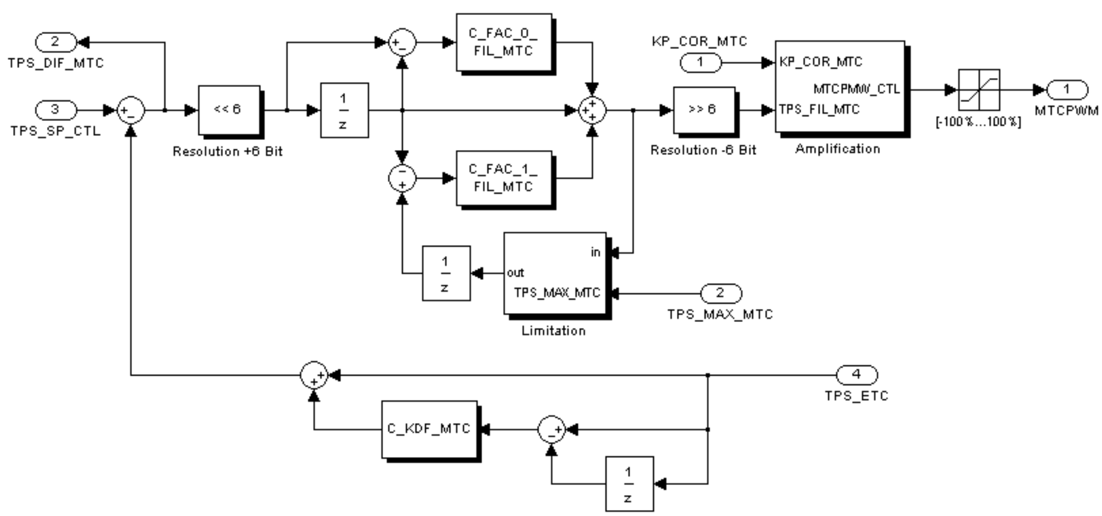


Figure 2: Technical realization, digital position controller

C.6.2 Controller amplification

<u>Description:</u>

The circuit amplification is held constantly under consideration of the battery voltage. The maximum input value of the PT_1 part of the transfer function in forward direction is calculated depend on controller amplification and battery voltage.

<u>Application conditions:</u>

Initialisation: KP_COR_MTC = C_KP_MTC

TPS_MAX_MTC = 100% / C_KP_MTC

Recurrence: 10 ms

Activation: at every engine state

Formula section:

FAC_VB_MTC = IP_FAC_VB_MTC(VB_MMV)

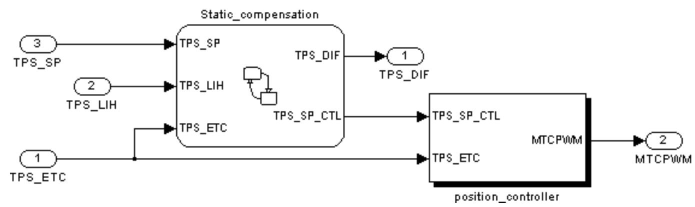
KP_COR_MTC = C_KP_MTC * FAC_VB_MTC

TPS_MAX_MTC = MTCPWM_LIM / KP_COR_MTC

C.6.3 Compensation of the system deviation

Description:

This function compensates the static control deviation of the throttle position controller. Therefore, it must be evaluated after the TPS setpoint selection and limitation. The digital position controller operates without integrator but with a high circuit amplification, consequently the position control has a small system deviation.



Signal flow diagram:

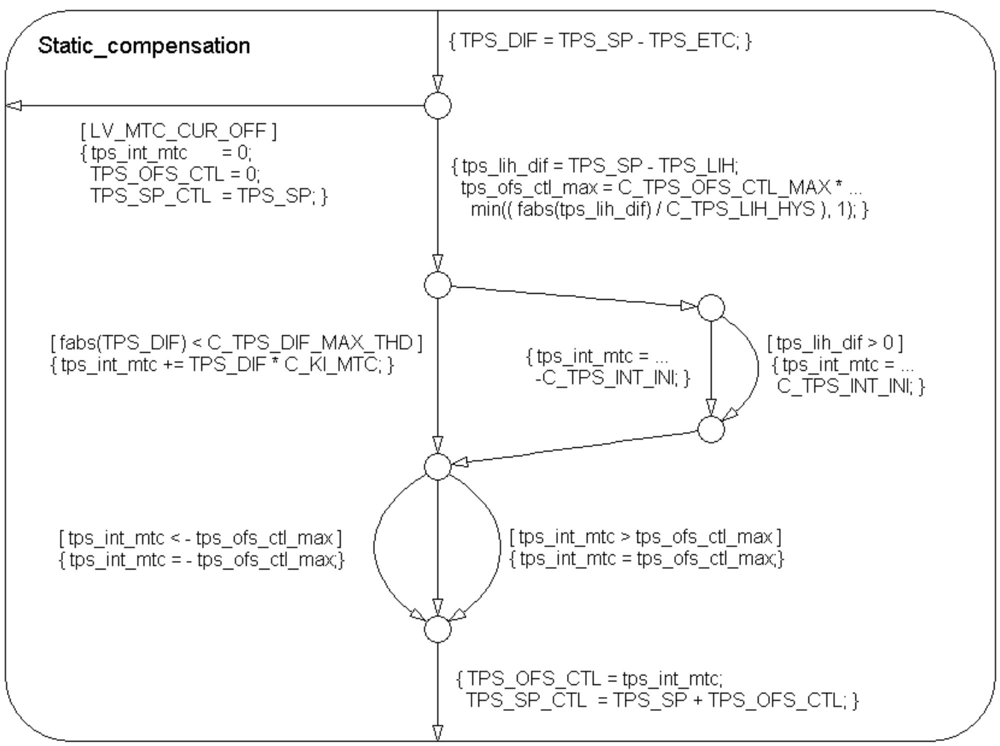


Figure2: Technical realization, static integrator

Application conditions:

Initialisation: TPS_DIF = TPS_OFS_CTL = TPS_SP_CTL = 0

Recurrence: 10 ms

Activation: at every engine state

Formula section:

```
{ Calculation of the system deviation }
TPS_DIF = TPS_SP – TPS_ETC(k)
if      LV_MTC_CUR_OFF
then    { power stage is switched off }
        tps_int_mtc      = 0
        TPS_OFS_CTL      = 0
        TPS_SP_CTL       = TPS_SP
else    { power stage is switched on }
        tps_lih_dif      = TPS_SP – TPS_LIH
        { calculation of the integrator limitation depend on the limp-home position }
        tps_ofs_ctl_max  = ...
            C_TPS_OFS_CTL_MAX * MIN[ ( | tps_lih_dif | / C_TPS_LIH_HYS ), 1 ]
        if      | TPS_DIF | < C_TPS_DIF_MAX
        then    { tps_int_mtc is calculated with a higher resolution than TPS_DIF  }
                tps_int_mtc + = TPS_DIF * C_KI_MTC
        else    { Integrator initalization }
                if      tps_lih_dif > 0
                then    tps_int_mtc = C_TPS_INT_INI
                else    tps_int_mtc = - C_TPS_INT_INI
                endif
        endif
        { limitation of the integrator output }
        if      tps_int_mtc < - tps_ofs_ctl_max
        then    tps_ int_mtc = - tps_ofs_ctl_max
        else    if      tps_int_mtc > tps_ofs_ctl_max
                then    tps_int_mtc = tps_ofs_ctl_max
                endif
        endif
        TPS_OFS_CTL      = tps_int_mtc
        { calculate the setpoint for the position controller }
        TPS_SP_CTL       = TPS_SP + TPS_OFS_CTL
endif
```

C.6.4 Reduced output limitation of the position controller

Description:

This function provides a higher availability of ETC control in case of long and high MTCPWM's (e.g. hard moving throttle blade, due to contamination or anything else).

In the diagnostic module "ETC actuator diagnosis" the moving mean value of the position controller output MTC_PWM is calculated. If the moving mean value MTCPWM_MMV exceeds a defined threshold then output limitation will be requested (LV_MTCPWM_RED_REQ = 1) by below function. The output limitation will be deactivated if the controller output falls below a second threshold (hysteresis).

Application conditions:

Initialisation: LV_MTCPWM_RED_REQ = 0, MTCPWM_LIM = 100 %

Recurrence: 10 ms

Activation: at every engine state

Formula section:

```
if      LV_MTCPWM_RED_REQ == 1
then    { Deactivate reduced output limitation of the position controller }
        if      MTCPWM _MMV < C_ MTCPWM _LIM_DI * FAC_VB_MTC
        then    LV_MTCPWM_RED_REQ = 0
        endif
else    { Activate reduced output limitation of the position controller }
        if      MTCPWM _MMV > C_ MTCPWM _LIM_ENA * FAC_VB_MTC
        then    LV_MTCPWM_RED_REQ = 1
        endif
endif

if      LV_ MTCPWM _RED_REQ == 1
then    { Reduced output of the position controller }
        MTCPWM _LIM = C_ MTCPWM _LIM * FAC_VB_MTC
else    { Limitation between -100 % and 100 % PWM output }
        MTCPWM _LIM = 100 %
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
C_KP_MTC	1	1h ... 4000h	0.0625 ... 1024	0.0625	% / °TPS
Controller amplification					
C_KI_MTC	1	0h ... 400h	0 ... 1	$9.766 \cdot 10^{-4}$	-
Static integrator amplification					
C_KDF_MTC	1	0h ... 400h	0 ... 64	0.0625	-
Controller difference amplification					
C_FAC_0_FIL_MTC	1	0h ... 400h	0 ... 1	$9.766 \cdot 10^{-4}$	-
Transfer function PDT_1 parameter b0					
C_FAC_1_FIL_MTC	1	0h ... 400h	0 ... 1	$9.766 \cdot 10^{-4}$	-
Transfer function PDT_1 parameter a1					
C_TPS_DIF_MAX	1	0h...FFh	0 ...1.86	0.0073	°TPS
Activation range for the static integrator					
C_TPS_INT_INI	1	0...89h	0 ... 0.999	0.0073	°TPS
Initialization value for the static integrator					
C_TPS_LIH_HYS	1	3h...FFh	0.022....1.86	0.0073	°TPS
Hysteresis area around limp-home position					
C_TPS_OFS_CTL_MAX	1	0...89h	0 ... 0.999	0.0073	°TPS
Maximum value of the static integrator					
C_MTCPWM_LIM	1	0h ... 7FFFh	0...99.997	0.00305	%
MTCPWM limitation value					
C_MTCPWM_LIM_DI	1	0h ... 7FFFh	0...99.997	0.00305	%
Threshold for MTCPWM limitation deactivation					
C_MTCPWM_LIM_ENA	1	0h ... 7FFFh	0...99.997	0.00305	%
Threshold for MTCPWM limitation activation					
IP_FAC_VB_MTC	2	200h ... 1000h	0.5 ... 4	$9.766 \cdot 10^{-4}$	-
LDP_VB_MMV_FAC_VB_MTC	2	0h ... FFh	0 ... 26	0.102	V
Battery voltage correction					

C.7 ETC position controller (appl. inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MTC_CUR_OFF	V/O	0H...1H	0...1	1	-
Request to switch off MTC powerstage					

Input data:

LV_ES	LV_IGK	LV_TPS_AD_ACT	LV_TPS_PWL
LV_TPS_SP_EXT_ADJ	LV_TPS_AD_CUR_OFF	LV_MTC_CUR_OFF_REQ	LV_OFF_MTC_MON
LV_ERR_MC	LV_ERR_MU	LV_ERR_TMP_MC	LV_ERR_TMP_MU
LV_PRDR_ACT			

FUNCTION DESCRIPTION:

LV_MTC_CUR_OFF contains all deactivation conditions for the output of the ETC position controller. If LV_MTC_CUR_OFF is active, the output of the ETC position controller is set to zero and the throttle actuator is currentless even with activated power stage.

Application conditions:

Initialisation: at reset :

LV_MTC_CUR_OFF = 1

{ Consider requirements of Monitoring Concept (Level 3 – PREDRIVE check) }

if LV_PRDR_ACT == 0

then *{ disable power stages }*

CALL BSW routine to disable H-bridge via disable line

else *{ enable power stages }*

CALL BSW routine to enable H-bridge via disable line

endif

Recurrence: 10 ms

Activation: all engine operating states

Formula section:

{ This application incidence example takes into consideration the LV_MTC_CUR_OFF for limp-home Mode. }

```
If  ( LV_ES                     and      { flag for engine stopped }
      LV_IGK                    and      { key on flag }
      LV_TPS_AD_ACT == 0        and      { TPS adaptation not active }
      LV_TPS_SP_EXT_ADJ == 0 )      or   { external setpoint not active }
    ( LV_IGK == 0               and      { tuneable time periode during ...
      LV_TPS_PWL == 0 )             or     post operating phase }
    LV_TPS_AD_CUR_OFF               or   { switch-off request during adaptation }
    LV_MTC_CUR_OFF_REQ              or   { bit set by monitoring level 1 }
    LV_OFF_MTC_MON                  or   { bit set by monitoring level 2 }
    LV_ERR_MC                       or   { bit set by monitoring level 3 }
```

```
        LV_ERR_MU                          or     { bit set by monitoring level 3 }
        LV_ERR_TMP_MC                      or     { bit set by monitoring level 3 }
        LV_ERR_TMP_MU                             { bit set by monitoring level 3 }
then
        LV_MTC_CUR_OFF = 1
        { Consider requirements of Monitoring Concept (Level 3 – PREDRIVE check) }
        if      LV_PRDR_ACT == 0
        then
                { disable power stages }
                CALL BSW routine to disable H-bridge via disable line
        else
                { enable power stages }
                CALL BSW routine to enable H-bridge via disable line
                { During PREDRIVE power stage must not be disabled by the main controller to
                  enable checking if MU is able to deactivate power stages ! }
        endif
else
        LV_MTC_CUR_OFF = 0
        { enable power stages }
        CALL BSW routine to enable H-bridge via disable line
endif
```

D Torque Management

Table of Contents

Index

A

C

D.1 Torque Coordination General

General information:

The aggregate *Torque Setpoint* defines the central indicated torque request to be produced at the engine. Therefore, several torque setpoints (internal and external) are considered and prioritised. The decisive outputs of the aggregate are TQI_SP_SLOW (torque setpoint for air path), TQI_SP (torque setpoint for ignition path) and TQI_SP_FAST_S (torque setpoint for stratified combustion mode). These torques are inputs to the aggregate *Torque determination and realization*.

The modules of this aggregate define the main track of the torque structure. The following figure shows the chain of the main important modules of the aggregate.

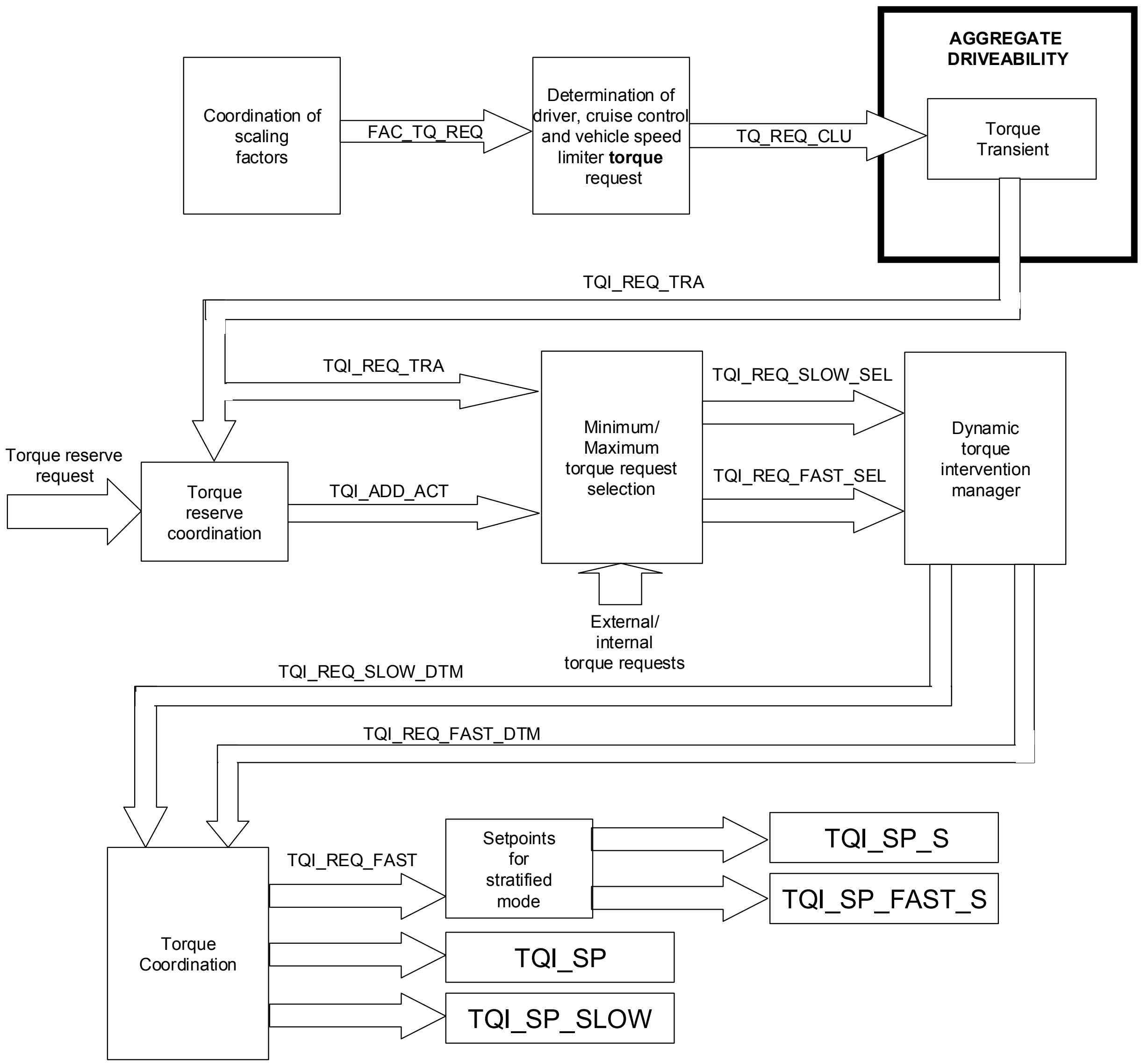


Figure 1: *Aggregate Torque Setpoint: major modul chain (not all modules of the aggregates are shown)*

The module’s attribution to the internal aggregate segmentation can be done as follows:

Coordination of scaling factors for requested torque

This module coordinates the torque scaling factors for driver, cruise control and vehicle speed limiter requests. The result is a coordinated scaling factor FAC_TQ_REQ.

Detection of driver passive

The detection of driver, cruise control or vehicle speed limiter passive is done with LV_CT (LV_CT = 1 means driver, cruise control or vehicle speed limiter passive). The bit is used (for example) in the engine operating states to detect LV_IS, LV_PL, ... = 0 or 1. The determination of LV_CT is done with the torque scaling factor FAC_TQ_REQ_CLU, which contents driver, cruise control or vehicle speed limiter requests.

Full load detection (template)

The primary task of the module “Full load detection” is to detect the full load request of the driver or - if requested – of cruise control. The indicator for the full load request is the logical variable LV_FL which constitutes the trigger for the full load enrichment in module “Full load enrichment”.
Under certain conditions it is necessary to suppress the full load request regardless of the driver demand. Such conditions can also be defined in this template.

Torque request at clutch general

With the scaling factor FAC_TQ_REQ (from *Coordination of scaling factors for requested torque*) the requested torque at clutch TQ_REQ_CLU is calculated.

Torque control at start

The engine start torque TQ_ST influences the torque at clutch during start phase; as a result it determines the engine speed transient response. TQ_ST is imported into module “Minimum Torque at Clutch” in the aggregate *Torque determination and realization* in order to increase engine speed during start phase.

Torque request for safety (Appl. Inc.) (template)

The module "Torque request for safety" needs a flag for having external torque demands, which can increase torque, as an input. Also, an activation flag for the module *Torque request for safety* is defined.

Torque request for safety

This module generates a torque request for safety for the slow (air) path and the fast (ignition) path, if the actual indicated engine torque exceeds the desired indicated engine torque, which is generated in the monitoring level, by more than a calculated threshold. Since the monitoring concept normally uses wide torque ranges to prevent permanent reactions of monitoring level 2, a limitation in level 1 can prevent failure torque developments inside the range of TQI_SP_MON and TQI_SP_MON plus offset from monitoring level 2.

Torque request for safety diagnosis

The objective of this function is to determine the number of intervention actions concerning the torque request for safety and to set the failure table entry if the counter exceeds a calibratable threshold. Under normal conditions, this module has no influence. It is meant to support the calibration. If the diagnosis leads to an entry in the failure table, this is a hint for the calibration guy that something in the calibration of the module *Torque request for safety* is still not well calibrated.

Torque based engine power limitation (template)

With the help of this function, the maximum allowed indicated engine torque can be limited (power limitation). For example, this limitation could consider a torque limitation to protect the gear box or a torque limitation in case of low fuel pressure (f.e. because the high pressure fuel pump is defective).

Torque reserve coordination

A torque reserve is a torque neutral increase of load with simultaneous retardation of ignition angle. Such torque reserves can be requested by different functions to increase the engine dynamics (i.e. idle speed controller) or to put thermodynamic energy into the exhaust system (to heat the catalyst or to desulfate the NO_X-trap). In the *Torque reserve coordination* all these torque reserve requests are coordinated and limited to a maximum allowed torque reserve, which is derived from the efficiency of the minimum allowed ignition angle. The output is a torque request TQI_ADD_ACT, which is torque request TQI_REQ_TRA plus torque reserve (TQ_ADD_REQ(_CH)).

Dynamic torque request coordination

This module coordinates project-specific dynamic requests for the realization of torque, expressed in dynamic logical bits (LV_TQ_DYN_xxx). They are used in the Dynamic torque intervention manager.

Dynamic torque intervention manager

The intention of the Dynamic Torque intervention Manager (DTM) is to have a central coordination of dynamic torque requests. It controls the realization of a torque request via the slow air path and the fast path with ignition intervention, fuel cut-off or lambda variation. It produces two kinds of outputs:

- Torque requests for slow and fast path for correct representation based on the Torque Structure.
- Logical values for switching different fast path realization variants to either their passive values or setpoint calculation via Torque Structure.

The state machine is controlled via dynamic control bits, that allow different states of intervention variants. Also the transitions from one state to the other can be influenced.

So the DTM finally adjusts the dynamic requirements from the control bits and gives the possibility to use several transitions between the different requirements.

Minimum / Maximum torque request selection (template)

This module coordinates all internal and external torque requests. It generates TQI_REQ_SLOW_SEL for the slow torque path (load) and TQI_REQ_FAST_SEL which represents the fast torque path (ignition, cylinder cut-off and lambda).

Torque coordination

The Torque Coordination module contains two functionalities.

First, there is the consideration of the limitation torques calculated in *Torque request for safety.*

Second, the module *Torque Coordination* calculates in homogeneous mode slow and fast torque setpoints (TQI_SP_SLOW for the air path and TQI_SP for ignition or also for fuel cut-off and lambda.

Torque setpoint for stratified mode

The module delivers the torque setpoint TQI_SP_FAST_S which is the decisive setpoint for the torque realization (fuel mass calculation) in the stratified combustion mode.
The second output, TQI_SP_S, is a torque setpoint that is not influenced by the fast torque interventions of the Idle Speed Controller and the Anti Jerk Controller. In the module 'Torque based fuel mass setpoint for stratified mode', this parameter will be converted into a fuel mass equivalent (MFF_SP_S) which will be utilized as load breakpoint for engine actuators with slow dynamics (e.g. FUP, EGR, PORT).
The input value TQI_REQ_FAST is delivered from the fast torque minimum/maximum selection where all internal and external torque are coordinated.

Following input values are considered:

TQ_DIF_P_D_FAST_IS; output from idle speed controller
TQ_DIF_AJ; output from anti jerk controller

D.2 Torque Control at Engine Start

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ST	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03215	Nm
Engine Start Torque					
TQ_ST_1	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03215	Nm
Engine Start Torque raw value					

Input data:

N_DIF	LV_TQ_MIN_CLU	TCO	

FUNCTION DESCRIPTION:

General information:

The engine start torque TQ_ST influences the torque at clutch during start phase as a result it determines the engine speed transient response. It is calculated from a one dimensional map versus engine speed difference N_DIF.

TQ_ST is imported into module “Minimum Torque at Clutch” in order to increase engine speed during start phase. Depending on the tuning data in IP_TQ_ST the engine speed overshoot canbe limitted to bring it to its setpoint at idle.

Activation: LV_TQ_MIN_CLU = 0

Deactivation: LV_TQ_MIN_CLU = 1

Initialization: TQ_ST = 0 Nm at reset

Update rate: 10 ms

Formula section:

TQ_ST_1 = IP_TQ_ST

if TQ_ST_1 >= TQ_ST then
TQ_ST = TQ_ST_1
else
$TQ\_ST_k = TQ\_ST_{k-1} + C\_CRLC\_TQ\_ST * (TQ\_ST\_1 - TQ\_ST_{k-1})$ (MMV calculation)
end

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ST	8x8	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_N_DIF_IP_TQ_ST	8	0 .. FFFF H	-32768 .. 32767	1	1/min
LDP_TCO_IP_TQ_ST	8	0 .. FF H	-48 .. 142.5	0.75	°C
Engine torque at start					
C_CRLC_TQ_ST	1	0 ... FF H	0 ... 0.996	0.0039	-
Correlation constant for negative PT1 filtering					

D.3 Reference and Basic Torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_TOT_BAS	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Total basic efficiency					
EFF_TOT_BAS_SLOW	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Total basic efficiency for feedback to slow torque path					
TQI_BAS	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Basic indicated engine torque					
TQI_REF	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Reference indicated engine torque					
TQI_REF_IGA	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_IGA_BAS_COR					
TQI_REF_IGA_LAMB	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_IGA_BAS_COR and EFF_LAMB_BAS					
TQI_REF_IGA_MIN_LAMB	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_IGA_MIN and EFF_LAMB_BAS					
TQI_REF_IGA_SCC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_IGA_BAS_COR and EFF_SCC_BAS					
TQI_REF_LAMB	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_LAMB_BAS					
TQI_REF_LAMB_SCC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_LAMB_BAS and EFF_SCC_BAS					
TQI_REF_SCC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque corrected by EFF_SCC_BAS					

Input data:

EFF_IGA_BAS_COR	EFF_IGA_BAS_COR_KNK_FIL	EFF_IGA_MIN	EFF_EGR_HOM
EFF_LAMB_AV	EFF_LAMB_BAS_COR	EFF_SCC_AV	EFF_SCC_BAS
LV_HOM_RUN	MAF	N_32	FAC_TQI_AD_SLOP
TQI_AD_OFS			

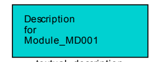
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_REF	16x12	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32_1_TQDR	16	0...FFH	0...8160	32	1/min
LDPM_MAF_1_TQDR	12	0...FFFFH	0...1389	0.0212	mg/stk
Indicated engine torque at reference conditions					

D.3.1 General Information

This module delivers the reference indicated engine torque TQI_REF, basic indicated engine torque TQI_BAS and other basic torque values for the torque model in homogeneous combustion mode.

Function Description



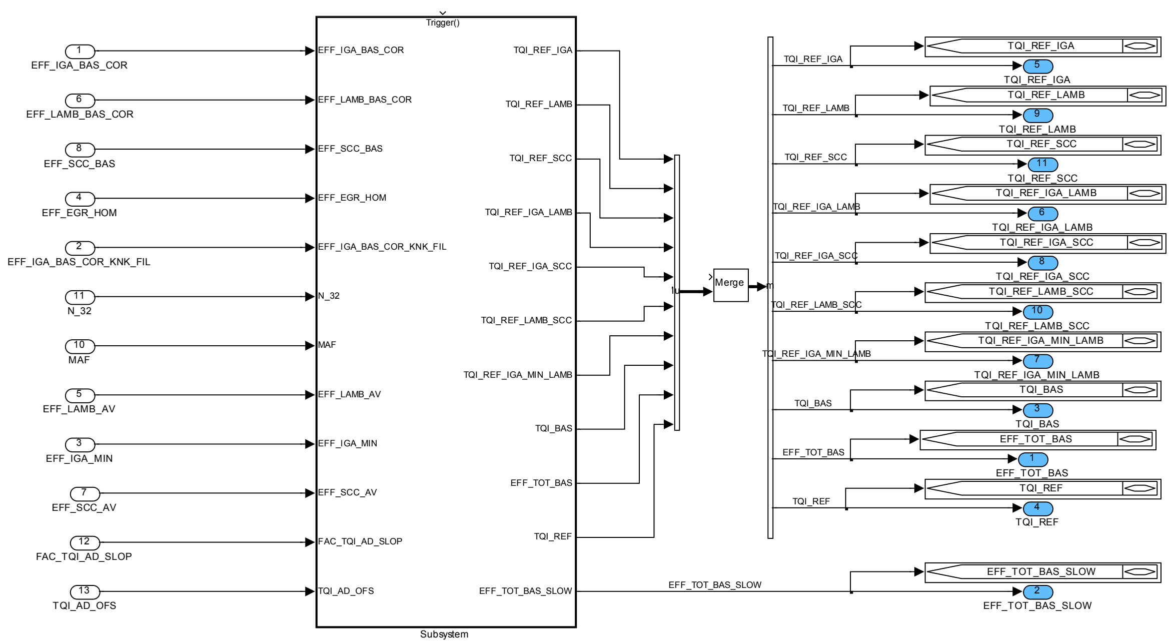


Figure 1 MD001

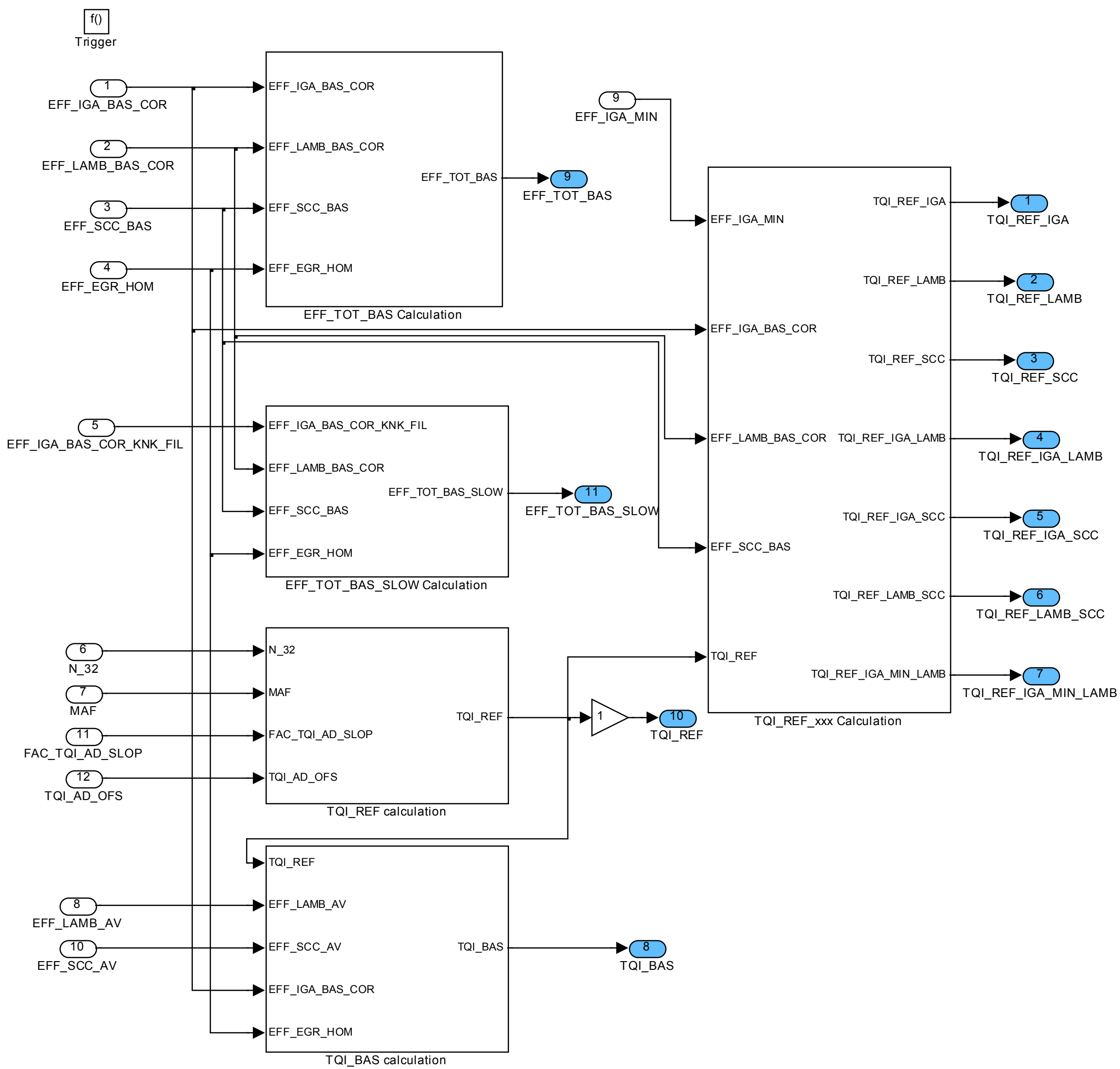


Figure 2 MD001/ Subsystem

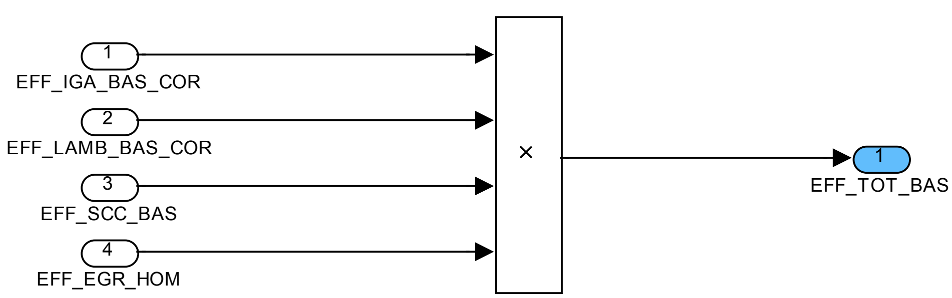


Figure 3 MD001/ Subsystem/ EFF_TOT_BAS Calculation

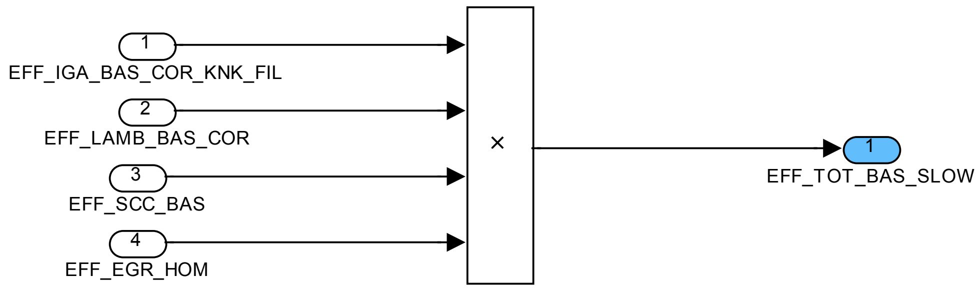


Figure 4 MD001/ Subsystem/ EFF_TOT_BAS_SLOW Calculation

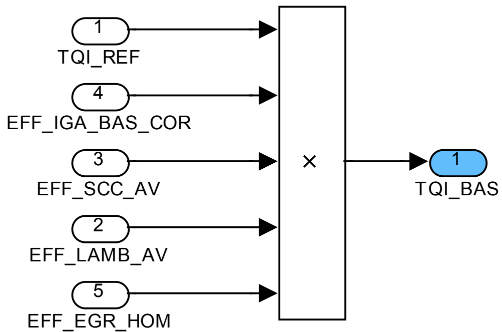


Figure 5 MD001/ Subsystem/ TQI_BAS calculation

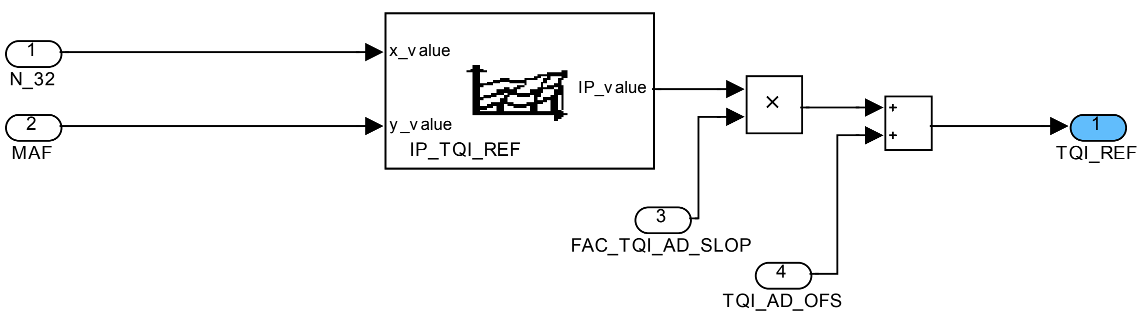


Figure 6 MD001/ Subsystem/ TQI_REF calculation

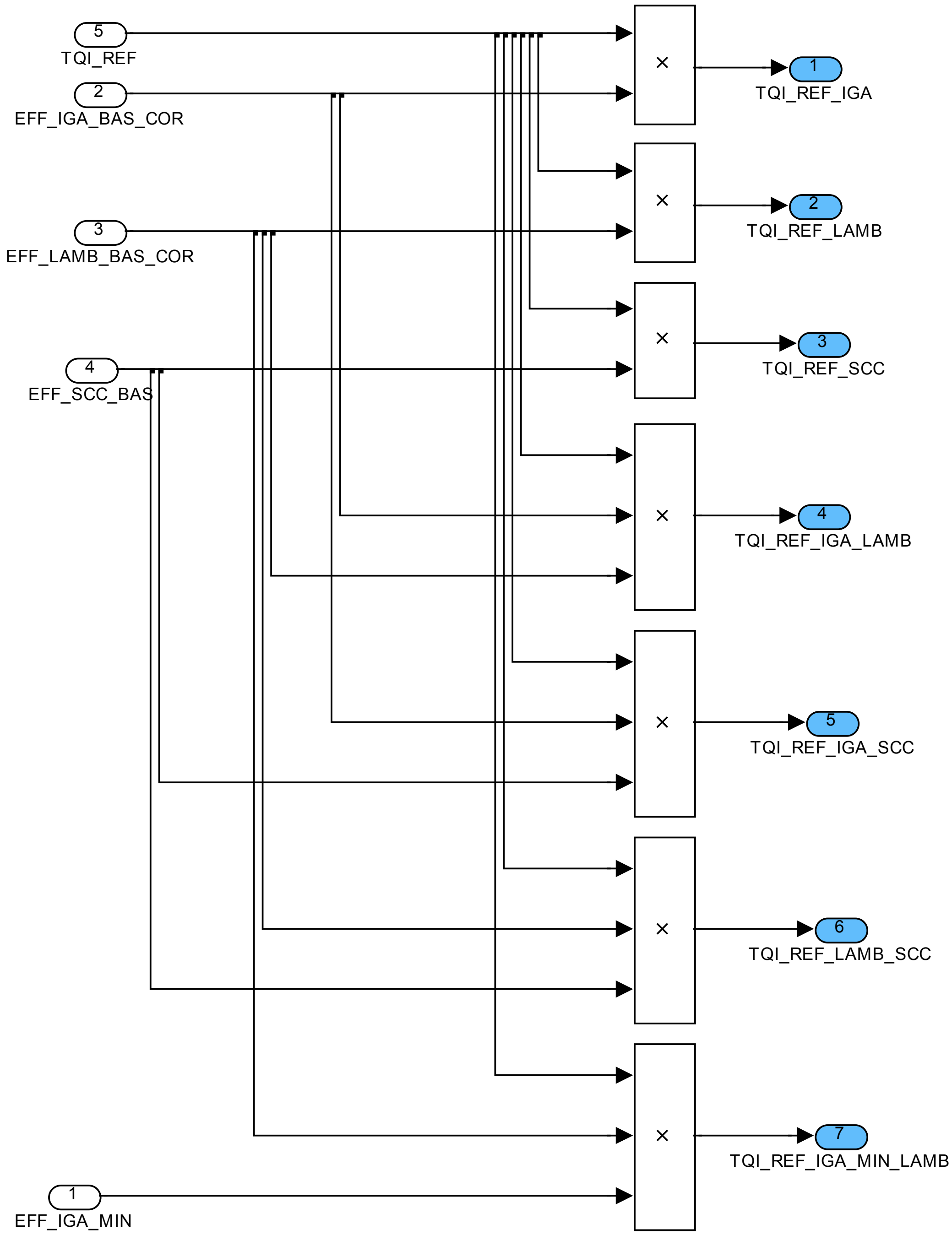


Figure 7 MD001/ Subsystem/ TQI_REF_xxx Calculation

D.4 Adaptation of indicated engine torque

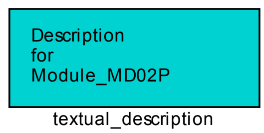
Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQI_AD_SLOP	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Adapted engine torque (slope)					
TQI_AD_OFS	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Adapted engine torque (offset value)					

D.4.1 General Information

In system configurations with a torque sensor, this module is utilized to evaluate this sensor information for an adaptation of the torque model. The torque model is adapted by means of an linear function, which is defined by a slope FAC_TQI_AD_SLOP and an offset TQI_AD_OFS.

Function Description



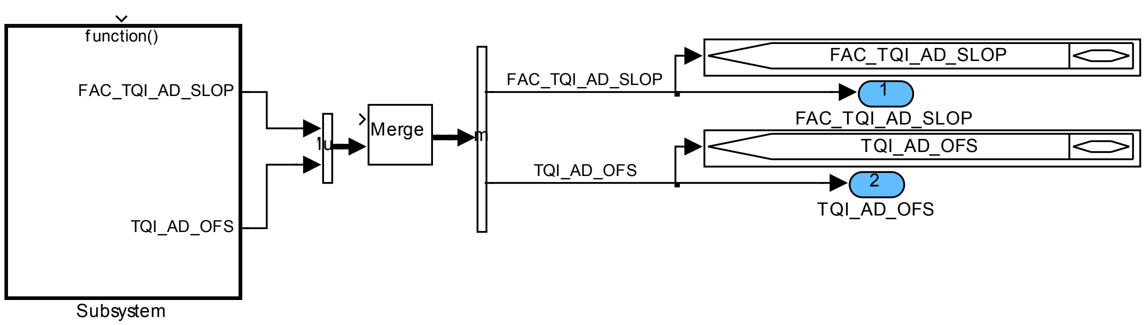


Figure 8 MD02P

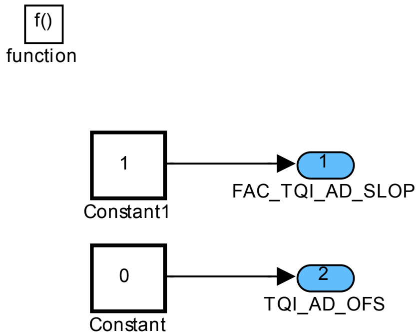


Figure 9 MD02P/ Subsystem

D.5 Torque Losses

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Overall torque losses (unfiltered value)					
TQ_LOSS_REQ_CLU	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Overall torque losses (filtered value)					
TQFR_MEC	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Mechanical friction losses of the engine					
TQFR	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Engine friction and pumping losses					
TQFR_ADD	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Temperature correction of engine friction losses					
TQFR_ST	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Start correction of engine friction losses					

Input data:

N	TCO	MAF_CYL_STK	TOIL
TCO_ST	TQ_DIF_IS_AD	TQ_LOSS_ADD	LV_ST_END
TQFR_DELTA_ADD	TQ_LOSS_ACC		

FUNCTION DESCRIPTION:

General information:

Target of the module “Torque Losses” is the determination of the overall torque losses of the engine. The torque losses mainly consist of the frictional and pumping losses and the additional torque losses caused by auxiliaries (e.g. air conditioning compressor). The model also considers the temperature dependency of the frictional losses, the break-in losses at start, project specific corrections and the adaptation of series spread.

Signal flow diagram

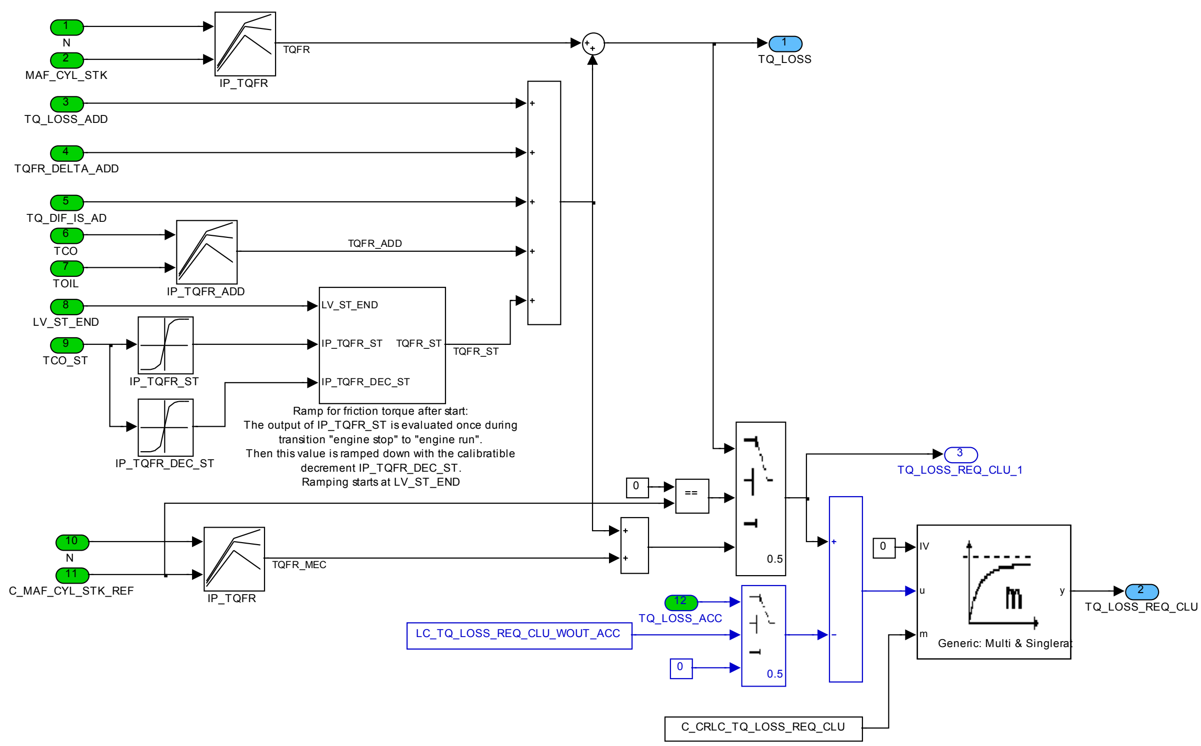


Figure 1: *Calculation of TQ_LOSS and TQ_LOSS_REQ_CLU*

Description :

The module has two main output parameters. These are TQ_LOSS and TQ_LOSS_REQ_CLU.

TQ_LOSS represents the unfiltered overall engine torque losses. It contains the frictional and pumping losses including project-specific corrections, the correction for temperature and after-start phase, the adaptation torque from the idle speed adaptation and the torque losses of auxiliaries.

TQ_LOSS_REQ_CLU represents the filtered value of the overall engine torque losses but unlike TQ_LOSS it does not depend on the mass of air flow into the cylinder. TQ_LOSS_REQ_CLU is used for the determination of the minimum and the maximum torque at clutch. The aim of this second parameter is an enhanced driveability of HPDI-vehicles during transitions between homogeneous and stratified combustion mode (please refer to tuning guide). The functionality is not HPDI-specific and can also be used for MPI-systems. Dependent on the calibration of the constant C_MAF_CYL_STK_REF TQ_LOSS_REQ_CLU derives either from TQFR_MEC or TQFR.

The single components of TQ_LOSS and TQ_LOSS_REQ_CLU can be described as follows:

1. Friction and pumping losses TQFR

TQFR represents the engine immanent friction and pumping losses. TQFR depends on engine speed N and the air mass into the cylinder MAF_CYL_STK.

2. Temperature correction TQFR_ADD

TQFR_ADD considers the coolant and oil temperature dependency of the friction losses.

3. Start correction TQFR_ST

TQFR_ST takes into consideration the increased friction losses right after start.

4. Idle speed adaptation of the engine immanent torque losses TQ_DIF_IS_AD

The idle speed adaptation torque TQ_DIF_IS_AD corrects deviations of engine immanent torque losses due to tolerances in series production. It is determined in the module “Idle speed adaptation”.

5. Torque losses of additional consumers TQ_LOSS_ADD

TQ_LOSS_ADD represents the interface to module “Coordination of additional torque losses” and expresses the torque requests of relevant auxiliaries (e.g. ACC).

6. Correction of the frictional and pumping losses TQFR_DELTA_ADD

TQFR_DELTA_ADD is a corrective parameter for the friction and pumping losses. It is calculated in the template “Correction of friction and pumping losses”. So the model for the friction and pumping losses can be corrected or extended for projectspecific purposes (e.g. Turbo, VVL,...).

7. The mechanical friction losses TQFR_MEC

TQFR_MEC represents the mechanical friction losses at wide open throttle. Although it is calculated from the same map as TQFR, it does not depend on mass of air flow and so it does not change during transitions between homogeneous and stratified combustion mode.

Application conditions:

Activation:	at every engine state
Deactivation:	-
Initialization:	TQ_LOSS = 0 Nm at reset
	TQ_LOSS_REQ_CLU = 0 at reset
	TQFR_ADD = IP_TQFR_ADD (TCO,TOIL)
Update rate:	optimised recurrency for each parameter

Formula section:

Calculation of the friction and pumping losses TQFR

TQFR = IP_TQFR (N, MAF_CYL_STK) **every 20 ms**

Calculation of TQFR_MEC :

TQFR_MEC = IP_TQFR (N, C_MAF_CYL_STK_REF) **every 20 ms**

Calculation of the torque losses of auxiliaries TQ_LOSS_ADD

TQ_LOSS_ADD is determined in the template “Coordination of additional torque losses”

Calculation of the project-specific corrections TQFR_DELTA_ADD

TQFR_DELTA_ADD is determined in the template “Correction of friction and pumping losses”

Calculation of the start correction TQFR_ST

$TQFR\_ST_0$ = IP_TQFR_ST (TCO_ST) calculated once at the beginning of engine-operating-state ST

TQFR_DEC_ST = IP_TQFR_DEC_ST (TCO_ST)

TQFR_ST is calculated once during transition from "engine stop" to "engine run". Afterwards, it is ramped down by the decrement IP_TQFR_DEC_ST, as soon as LV_ST_END = 1:

IF LV_ST_END == 1

IF $TQFR\_ST_{k-1}$ > TQFR_DEC_ST

THEN

TQFR_ST = 0

ELSE

$TQFR\_ST_k = TQFR\_ST_{k-1} - TQFR\_DEC\_ST$ **every 100 ms**

END

Calculation of the idle speed adaptation torque TQ_DIF_IS_AD

TQ_DIF_IS_AD is determined in module “Idle Speed Adaptation”

Calculation of the temperature correction TQFR_ADD

TQFR_ADD = IP_TQFR_ADD (TCO,TOIL) **every 20 ms**

Calculation of the unfiltered overall torque losses TQ_LOSS

TQ_LOSS = TQFR
+ TQFR_ADD
+ TQFR_ST
+ TQ_DIF_IS_AD
+ TQ_LOSS_ADD
+ TQFR_DELTA_ADD

every 10 ms

Calculation of TQ_LOSS_REQ_CLU and low pass filtering:

IF C_MAF_CYL_STK_REF == 0
THEN
TQ_LOSS_REQ_CLU_1 = TQ_LOSS
ELSE
TQ_LOSS_REQ_CLU_1 = TQFR_MEC
+ TQFR_ADD
+ TQFR_ST
+ TQ_DIF_IS_AD
+ TQ_LOSS_ADD
+ TQFR_DELTA_ADD
END

IF LC_TQ_LOSS_REQ_CLU_WOUT_ACC
THEN
TQ_LOSS_REQ_CLU = TQ_LOSS_REQ_CLU_1– TQ_LOSS_ACC
ELSE
TQ_LOSS_REQ_CLU = TQ_LOSS_REQ_CLU_1
END

$$TQ\_LOSS\_REQ\_CLU_{k} = TQ\_LOSS\_REQ\_CLU_{k-1} + C\_CRLC\_TQ\_LOSS\_REQ\_CLU * (TQ\_LOSS\_REQ\_CLU - TQ\_LOSS\_REQ\_CLU_{k-1})$$

every 10 ms

Remark : Convergent calculation of filter !

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQFR	12x10	0...8000H	-1024...0	0.03125	Nm
LDP_N_IP_TQFR	12	0...1FE0H	0...8160	1	1/min
LDP_MAF_CYL_STK_IP_TQFR	10	0...FFFFH	0...1389	0.0212	mg/stroke
Frictional and pumping losses of the engine					
IP_TQFR_ADD	8x8	0...FFFFH	-1024...1023.97	0.03125	Nm
LDP_TCO_IP_TQFR_ADD	8	0...FEH	-48...142.5	0.75	°C
LDP_TOIL_IP_TQFR_ADD	8	0...C8H	-40...160	1	°C
Coolant and oil temperature correction of friction losses					
IP_TQFR_ST	1x4	0...8000H	-1024...0	0.03125	Nm
LDPM_TCO_ST_1_TQLO	4	0...FEH	-48...142.5	0.75	°C
Friction torque correction during after start phase					
IP_TQFR_DEC_ST	1x4	0...8000H	-1024...0	0.03125	Nm
LDPM_TCO_ST_1_TQLO	4	0...FEH	-48...142.5	0.75	°C
Decrement for start correction of friction losses					
C_MAF_CYL_STK_REF	1	0...FFFFH	0...1389	0.0212	mg/stroke
Constant for the determination of the friction losses					
C_CRLC_TQ_LOSS_REQ_CLU	1	0...FFH	0...0.996	0.0039	-
Correlation constant for the filtering of TQ_LOSS_REQ_CLU					
LC_TQ_LOSS_REQ_CLU_WOUT_ACC	1	0...1H	0...1	1	-
Switch to remove air conditioning load from driver request scaling					

D.6 Coordination of additional torque losses

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_ADD	V/O	8000 .. 7FFFH	-1024 .. 1023.97	0.03125	Nm
additional torque losses					

Input data:

TQ_LOSS_ECF	TQ_LOSS_ACC	TQ_LOSS_PSTE	TQ_LOSS_PSTE_PSPT
TQ_LOSS_ALTER	TQ_LOSS_AMP	CONF_PSTE	

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_ADD expresses the overall additional engine torque losses. In this module the additional losses have to be coordinated.

TQ_LOSS_xxx from external consumers are considered by special torque loss input values:

TQ_LOSS_ACC	air conditioning compressor
TQ_LOSS_ECF	electric cooling fan
TQ_LOSS_ALTER	electric load from alternator
TQ_LOSS_PSTE	power steering pressure from switch
TQ_LOSS_PSTE_PSPT	power steering pressure from analog transducer

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: TQ_LOSS_ADD = 0 Nm at reset

Update Rate: 10 ms

Formula section:

TQ_LOSS_ADD = TQ_LOSS_ACC
+ TQ_LOSS_ECF
+ TQ_LOSS_ALTER
+ TQ_LOSS_AMP
+ TQ_LOSS_PSTE (if CONF_PSTE = 0)
+ TQ_LOSS_PSTE_PSPT (if CONF_PSTE = 1)

D.7 Altitude correction for torque loss

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_AMP	V/O	8000 .. 7FFFH	-1024 .. 1023.97	0.03125	Nm
Altitude correction for torque losses					
AMP_DIF	V/O	8000...7FFF H	-2717...2716.96	0.083	hPa
AMP difference compared to the reference AMP					

Input data:

AMP	PQ		

FUNCTION DESCRIPTION:

General information:

At high elevations, lower ambient pressure causes lower exhaust backpressure, so pumping losses to be lower.
To compensate this, the altitude correction for torque losses is introduced.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: TQ_LOSS_AMP = 0 Nm at reset

Update Rate: 10 ms

Formula section:

AMP_DIF = C_AMP_REF - AMP

TQ_LOSS_AMP = IP_TQ_LOSS_AMP__AMP_DIF__PQ

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_LOSS_AMP	8*8	0...FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_AMP_DIF__TQ_LOSS_AMP	8	0...FFFFH	-2717...2716.96	0.083	hPa
LDP_PQ__TQ_LOSS_AMP	8	0...FFFFH	0...0.999985	1.52e-5	-
Altitude correction for torque losses					
C_AMP_REF	1	0...FFFF H	0...5434	0.083	hPa
The reference altitude pressure					

D.8 Correction of friction and pumping losses

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQFR_DELTA_ADD	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm

Input data:

FUNCTION DESCRIPTION:

Application conditions:

Initialisation: TQFR_DELTA_ADD = 0 at reset

Recurrence: project-specific recurrency, dependent on the fastest input

Activation: At every engine state

Deactivation:

Formula section:

TQFR_DELTA_ADD = 0

D.9 Torque Loss and Reserve for ACC

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_ACC	V/O	8000...7FFF H	-1024...1023.97	0.03125	Nm
Torque reserve for switching on air conditioning compressor					
TQ_LOSS_ACC	V/O	8000 ... 0 H	-1024 ... 0	0.03125	Nm
Engine torque losses air conditioning compressor					
T_ACC_SLOW	V	0 ... FF H	0 ... 255	1	s
Timer for duration of transient TQ_LOSS_ACC correction					
T_ACC_FAST	V	0 ... FF H	0 ... 2.55	0.01	s
Timer for duration of transient TQ_LOSS_ACC correction					
FAC_TQ_LOSS_ACC_ON	V	0 ... FF H	0 ... 16	0.0627	-
TQ_LOSS_ACC weighting factor					
TQ_LOSS_ACC_AD	V	8000 ... 7FFF H	-1024 ... 1023.97	0.03125	Nm
Engine torque losses air conditioning compressor from adaption					
TQ_MAX_ACC_FATC	V/O	8000 ... 0 H	-1024 ... 0	0.03125	Nm
Maximum engine torque losses air conditioning via CAN					

Input data:

TQ_DIF_IS_AD_ACC	TIA	N_32	LV_RLY_ACCOUT_CTRL
LV_PRS_ACC	ACP	CONF_ACP	VS
T_IS_AST	TCO	TAM	LV_IS
VS	CONF_ACC_FATC	TQ_LOSS_ACC_CAN	LV_RLY_ACCOUT

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_ACC must display the air-conditioner compressor power requirements.

In this case the needed torque to drive the ACC is depending on the actual pressure of the ACC-fluid (LV_PRS_ACC, ACP), the air intake temperature and the actual pumping-speed (proportional to engine-speed).

The calculation of the torque loss starts prior to ACC activation (LV_RLY_ACCOUT_CTRL = 1).

The ACC adaptation value TQ_DIF_IS_AD_ACC, determined in the module "idle speed adaptation", is added to the calculated ACC torque.

Special correction at activation :

To take into account the gradually closing of the electro-magnetic clutch and the increased starting-torque of the ACC at activation and deactivation (fast phenomenon), a special weighting factor is multiplied to the basic torque losses.

A timer T_ACC_FAST is started at transition LV_RLY_ACCOUT_CTRL 0 -> 1 and runs until it´s maximum value. It is input for the table IP_FAC_TQ_LOSS_ACC_ON_FAST which is a weighting factor for ACC activation.

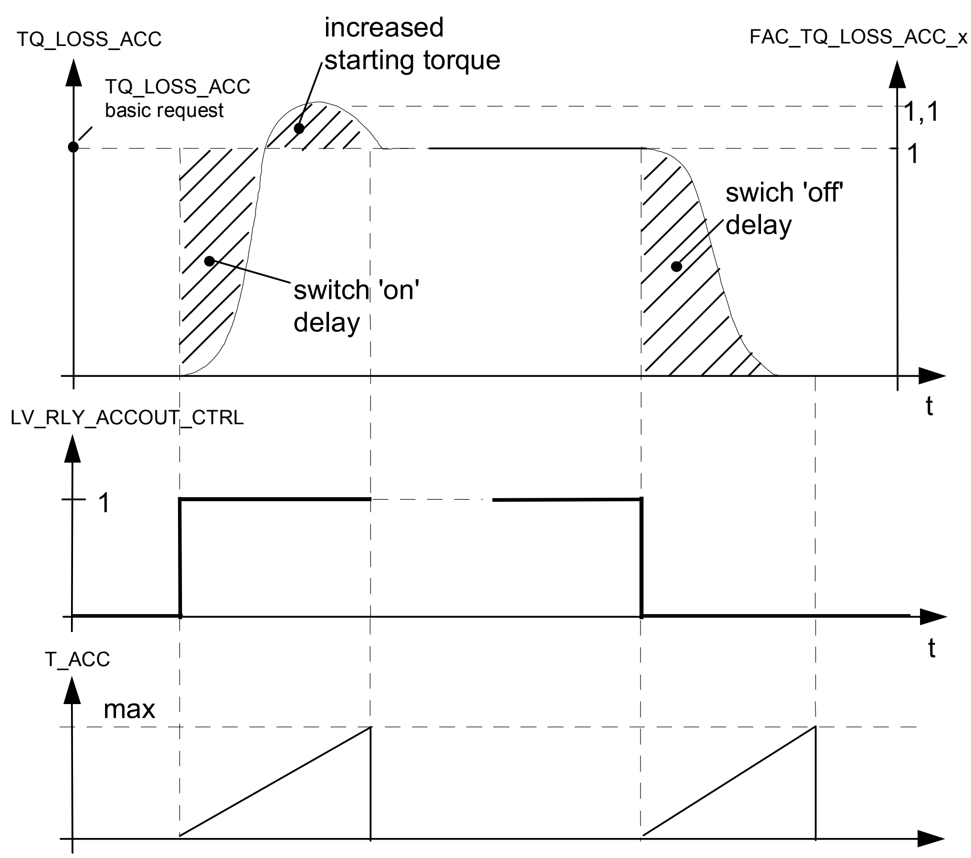
To take into account the variation of the fluid pressure inside the ACC during the first seconds of activation (slow phenomenon), a second special weighting factor is multiplied to the basic torque losses.

A timer T_ACC_SLOW is started at transition LV_RLY_ACCOUT_CTRL 0 -> 1 and runs until it´s maximum value. It is input for the table IP_FAC_TQ_LOSS_ACC_ON_SLOW which is a weighting factor for ACC activation.

Special correction at deactivation :

The timer T_ACC_FAST is also started at transition LV_RLY_ACCOUT_CTRL 1 -> 0 and runs until it´s maximum value. It is input for the table IP_FAC_TQ_LOSS_ACC_OFF which is the weighting factor for ACC deactivation.

These transient corrections are only active during a limited time after activation and deactivation of the climatic compressor.



D.9.1 Determination of Torque Loss ACC

The determination of TQ_LOSS_ACC starts parallel to the ACC relay activation.

Activation: Every engine state

Deactivation: -

Initialisation: at reset: TQ_LOSS_ACC = 0

Update Rate: 10 ms

Formula section:

activation of TQ_LOSS_ACC calculation including transient correction;

IP_FAC_TQ_LOSS_ACC_ON_FAST is regarded until T_ACC_FAST has reached it´s max. value,

IP_FAC_TQ_LOSS_ACC_ON_SLOW is regarded until T_ACC_SLOW has reached it´s max. value,

otherwise the multiplikation factor is ‘1’;

```
If     LV_IS = 1
then        if      TIA >= C_TIA_TQ_LOSS_ACC_ON_IS
     then   FAC_TQ_LOSS_ACC_ON =
              IP_FAC_TQ_LOSS_ACC_FAST_IDLE_0__TCO__T_ACC
              * IP_FAC_TQ_LOSS_ACC_SLOW_IDLE_0__TCO__T_ACC
              * IP_FAC_TQ_LOSS_ACC_IDLE_0_1__N_32__T_ACC
              * IP_FAC_TQ_LOSS_ACC_IDLE_0_2__T_IS_AST__T_ACC

     else   FAC_TQ_LOSS_ACC_ON =
              IP_FAC_TQ_LOSS_ACC_FAST_IDLE_1__TCO__T_ACC
              * IP_FAC_TQ_LOSS_ACC_SLOW_IDLE_1__TCO__T_ACC
              * IP_FAC_TQ_LOSS_ACC_IDLE_1__N_32__T_ACC
 else       FAC_TQ_LOSS_ACC_ON = IP_FAC_TQ_LOSS_ACC_ON_FAST
                                 * IP_FAC_TQ_LOSS_ACC_ON_SLOW
                                 * IP_FAC_TQ_LOSS_ACC_VS
endif


If          LV_RLY_ACCOUT_CTRL = 1
then        TQ_LOSS_ACC_AD = TQ_DIF_IS_AD_ACC * IP_TQ_LOSS_ACC_FAC
            where TQ_LOSS_ACC_AD is limited with change limitation
            C_TQ_LOSS_ACC_LGRD
else
            TQ_LOSS_ACC_AD =  0 Nm
end



If          CONF_ACC_FATC  =  0
then
            If      LV_RLY_ACCOUT_CTRL = 1 and CONF_ACP = 0
            then    TQ_LOSS_ACC = (IP_TQ_LOSS_ACC_0__TIA__LV_PRS_ACC
```

```
                                        * FAC_TQ_LOSS_ACC_ON
                                        * IP_TQ_LOSS_ACC_FAC)
                                        + TQ_LOSS_ACC_AD
          else   if     LV_RLY_ACCOUT_CTRL = 1 and CONF_ACP = 1
                 then   TQ_LOSS_ACC = (IP_TQ_LOSS_ACC_1__TIA__ACP
                                        * FAC_TQ_LOSS_ACC_ON
                                        * IP_TQ_LOSS_ACC_FAC )
                                        + TQ_LOSS_ACC_AD
                 endif
          endif
else      external air condition control uint present
          if     LV_RLY_ACCOUT_CTRL = 1
          then   TQ_LOSS_ACC = TQ_LOSS_ACC_CAN + TQ_LOSS_ACC_AD
          endif
endif
```

Note : if LV_PRS_ACC changes, the new IP_TQ_LOSS_ACC_0 value is applied with the change limitation C_TQ_LOSS_ACC_LGRD.

```
# TQ_LOSS transient correction at deactivation;
# storage of last valid TQ_LOSS_ACC
if        transition LV_RLY_ACCOUT_CTRL  1 -> 0
then      TQ_LOSS_ACC_1  =  TQ_LOSS_ACC (n-1)
endif
```

```
# deactivation transient correction until T_ACC_FAST has reached it´s max. value
if   LV_RLY_ACCOUT_CTRL = 0
          if     T_ACC_FAST < maximum value
          then   TQ_LOSS_ACC = TQ_LOSS_ACC_1 * IP_FAC_TQ_LOSS_ACC_OFF
          else   TQ_LOSS_ACC  = 0 Nm
          endif
endif
```

```
# calculation of maximum air condition torque loss via CAN

If   LV_RLY_ACCOUT = 1
          TQ_MAX_ACC_FATC = C_TQ_MAX_ACC_FATC_ACCOUT
else      TQ_MAX_ACC_FATC = C_TQ_MAX_ACC_FATC
```

endif

D.9.2 Determination of Torque Lead for ACC

To avoid drivability impact at ACC switching it´s necessary to define a torque reserve.

Activation: Every engine state

Deactivation: -

Initialisation: at reset: TQ_ADD_ACC = 0

Update Rate: 10 ms

Formula section:

if LV_RLY_ACCOUT_CTRL = 1 **and** LV_IS = 1 **and** VS < C_VS_MAX_TQ_ADD_ACC

TQ_ADD_ACC = IP_TQ_ADD_ACC (TAM) * IP_FAC_TQ_ADD_ACC(T_ACC_FAST)

else

TQ_ADD_ACC = 0 Nm

end

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MAX_TQ_ADD_ACC	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for applying torque reserve for air conditioning					
IP_TQ_LOSS_ACC_0	6*2	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_TIA__TQ_LOSS_ACC_0	6	0 .. FE H	-48 .. 142,5	0,75	°C
LDP_LV_PRS_ACC__TQ_LOSS_ACC_0	2	0 .. 1 H	0 .. 1	1	-
ACC torque loss due to intake air temperature and ACC fluid-pressure from pressure switch					
IP_TQ_LOSS_ACC_1	6*2	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_TIA__TQ_LOSS_ACC_1	6	0 .. FE H	-48 .. 142,5	0,75	°C
LDP_ACP__TQ_LOSS_ACC_1	6	0 .. FF H	0 .. 510	2	PSI
ACC torque loss due to intake air temperature and ACC fluid-pressure from APT					
IP_TQ_LOSS_ACC_FAC	6	0 .. FF H	0 .. 4	0,0157	-
LDP_N_32__TQ_LOSS_ACC	6	0 .. FF H	0 .. 8160	32	rpm
TQ_LOSS_ACC weighting factor due to engine-speed					
IP_FAC_TQ_LOSS_ACC_ON_FAST	8	0 ... FF H	0 ... 16	0,0627	-
LDP_T_ACC__TQ_LOSS_ACC_ON_FAST	8	0 ... FF H	0 ... 2.55	0.01	s
TQ_LOSS_ACC weighting factor at ACC activation – fast behaviour					
IP_FAC_TQ_LOSS_ACC_ON_SLOW	8*6	0 ... FF H	0 ... 4	0,0157	-
LDP_T_ACC__TQ_LOSS_ACC_ON_SLOW	8	0 ... FF H	0 ... 255	1	s
LDP_N_32__FAC_TQ_LOSS_ACC_ON	6	0 .. FF H	0 .. 8160	32	rpm
TQ_LOSS_ACC weighting factor at ACC activation – slow behaviour					
IP_FAC_TQ_LOSS_ACC_VS	8	0 ... FF H	0 ... 16	0,0627	-
LDP_VS__FAC_TQ_LOSS_ACC_VS	8	0 ... FF H	0 ... 255	1	Km/h
TQ_LOSS_ACC weighting factor at ACC activation – fast behaviour in idle					
IP_FAC_TQ_LOSS_ACC_FAST_IDLE_0	8	0 ... FF H	0 ... 16	0,0627	-
LDPM_TCO__TQ_LOSS_ACC_FAST	6	0 ... FE H	-48 .. 142,5	0,75	°C
LDPM_T_ACC__TQ_LOSS_ACC_FAST	8	0 ... FF H	0 ... 2.55	0.01	s
TQ_LOSS_ACC weighting factor at ACC activation for high TIA – fast behaviour in idle					
IP_FAC_TQ_LOSS_ACC_SLOW_IDLE_0	8*6	0 ... FF H	0 ... 4	0,0157	-
LDPM_TCO__TQ_LOSS_ACC_SLOW	6	0 ... FE H	-48 .. 142,5	0,75	°C
LDPM_T_ACC__TQ_LOSS_ACC_SLOW	8	0 ... FF H	0 ... 255	1	s
TQ_LOSS_ACC weighting factor at ACC activation for high TIA – slow behaviour in idle					

IP_FAC_TQ_LOSS_ACC_IDLE_0_1	8*6	0 ... FF H	0 ... 4	0,0157	-
LDPM_T_ACC__TQ_LOSS_ACC_SLOW	8	0 ... FF H	0 ... 255	1	s
LDPM_N_32__FAC_TQ_LOSS_ACC	6	0 .. FF H	0 .. 8160	32	rpm
TQ_LOSS_ACC weighting factor at ACC activation – slow behaviour in idle					
IP_FAC_TQ_LOSS_ACC_IDLE_0_2	8*6	0 ... FF H	0 ... 4	0,0157	-
LDPM_T_ACC__TQ_LOSS_ACC_FAST	8	0 ... FF H	0 ... 2.55	0.01	s
LDP_T_IS_AST__FAC_TQ_LOSS_ACC	6	0 .. FFFF H	0 .. 65535	1	s
TQ_LOSS_ACC weighting factor at ACC activation for T_IS_AST					
IP_FAC_TQ_LOSS_ACC_FAST_IDLE_1	8	0 ... FF H	0 ... 16	0,0627	-
LDPM_TCO__ TQ_LOSS_ACC_FAST	6	0 .. FE H	-48 .. 142,5	0,75	°C
LDPM_T_ACC__TQ_LOSS_ACC_FAST	8	0 ... FF H	0 ... 2.55	0.01	s
TQ_LOSS_ACC weighting factor at ACC activation – fast behaviour in idle					
IP_FAC_TQ_LOSS_ACC_SLOW_IDLE_1	8*6	0 ... FF H	0 ... 4	0,0157	-
LDPM_TCO__ TQ_LOSS_ACC_SLOW	6	0 .. FE H	-48 .. 142,5	0,75	°C
LDPM_T_ACC__TQ_LOSS_ACC_SLOW	8	0 ... FF H	0 ... 255	1	s
TQ_LOSS_ACC weighting factor at ACC activation – slow behaviour in idle					
IP_FAC_TQ_LOSS_ACC_IDLE_1	8*6	0 ... FF H	0 ... 4	0,0157	-
LDPM_T_ACC__TQ_LOSS_ACC_SLOW	8	0 ... FF H	0 ... 255	1	s
LDPM_N_32__FAC_TQ_LOSS_ACC	6	0 .. FF H	0 .. 8160	32	rpm
TQ_LOSS_ACC weighting factor at ACC activation – slow behaviour in idle					
IP_FAC_TQ_LOSS_ACC_OFF	8*6	0 ... FF H	0 ... 4	0,0157	-
LDP_T_ACC__FAC_TQ_LOSS_ACC_OFF	8	0 ... FF H	0 ... 2.55	0.01	s
LDP_N_32__FAC_TQ_LOSS_ACC_OFF	6	0 .. FF H	0 .. 8160	32	rpm
TQ_LOSS_ACC weighting factor at ACC deactivation					
C_TQ_LOSS_ACC_LGRD	1	0 .. 7FFF H	0 .. 1023.97	0.03125	Nm/10ms
Change limitation in case of LV_PRS_ACC transition					
C_TIA_TQ_LOSS_ACC_ON_IS	1	0 .. FE H	-48 ... 142.5	0.75	°C
Air temperature condition to active the special FAC_TQ_LOSS_ACC_ON for high TIA					
IP_TQ_ADD_ACC	6	0...7FFFH	0...1023.96875	0.03125	Nm
LDP_TAM__TQ_ADD_ACC	6	0...FEH	-48...142.5	0.75	°C
Torque reserve for ACC activation					
IP_FAC_TQ_ADD_ACC	8	0...FFH	0...1.99218	0.0078125	-
LDPM_T_ACC__TQ_LOSS_ACC_FAST	8	0 ... FF H	0 ... 2.55	0.01	s
Weighting factor for torque reserve after ACC activation					
C_TQ_MAX_ACC_FATC	1	8000 ... 0 H	-1024 ... 0	0.03125	Nm
Maximum engine torque losses for air conditioning via CAN with relay off					
C_TQ_MAX_ACC_FATC _ACCOUT	1	8000 ... 0 H	-1024 ... 0	0.03125	Nm
Maximum engine torque losses for air conditioning via CAN					

D.10 Torque Loss - Electric Cooling Fan

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_ECF	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Engine torque losses by electric cooling fan					
TQ_LOSS_ECF_BAS	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Basic engine torque losses by electric cooling fan in case of C_CONF_CFA = 0					
TQ_LOSS_ECF_TRA	V	8000 .. 0 H	-1024 .. 0	0.03125	Nm
Transient engine torque losses by electric cooling fan in case of C_CONF_CFA = 0					

Input data:

STATE_CFA	N_32	CFAPWM_CFA	CONF_CFA
N	STATE_CFA_IS		

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_ECF represents the engine torque loss due to power consumption by the electric cooling fan (ECF).

ECF power consumption value[W] in ID_POW_LOSS_ECF is converted to Nm by specified formula as below.

Application conditions:

Activation: at every engine operating state

Deactivation: -

Initialisation: at reset

Recurrence: 100 ms

(see: “ENTE scheduler”)

Formula section:

Definition of TQ_LOSS_ECF_BAS

TQ_LOSS_ECF_BAS = (ID_POW_LOSS_ECF_0 * 60) / (2 * 3.141828 * N)

Definition of TQ_LOSS_ECF_TRA

If Transition STATE_CFA_IS(n) > STATE_CFA_IS(n-1)

Then TQ_LOSS_ECF_TRA is set to ID_TQ_LOSS_ECF_TRA
and reset to 0 using change limitation TQ_LOSS_ECF_TRA
C_LGRD_TQ_LOSS_ECF_TRA.

If Transition STATE_CFA_IS(n) < STATE_CFA_IS(n-1)

Then TQ_LOSS_ECF_TRA = 0

```
If      N_32 > C_N_MIN_TQ_LOSS_ECF
then    if        CONF_CFA = 0
        then      TQ_LOSS_ECF = TQ_LOSS_ECF_BAS + TQ_LOSS_ECF_TRA
        else
           if     CONF_CFA = 1
           then   TQ_LOSS_ECF = ( IP_POW_LOSS_ECF_1 * 60 ) / ( 2 * 3.141828 * N )
           endif
        endif
else    TQ_LOSS_ECF = 0
endif
```

The change of TQ_LOSS_ECF is limited by limitation gradient C_LGRD_TQ_LOSS_ECF.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_POW_LOSS_ECF_0	4	0 ... FF H	0 ... -5100	20	W
LDPM_STATE_CFA__POW_LOSS_ECF_0	4	0 ... 3 H	0 ... 3	1	-
ECF power consumption versus STATE_CFA					
ID_TQ_LOSS_ECF_TRA	4	0 .. 8000 H	-1024 .. 0	0.03125	Nm
LDP_STATE_CFA_IS _ TQ_LOSS_ECF_TRA	4	0 ... 3 H	0 ... 3	1	-
ECF power consumption versus STATE_CFA					
C_LGRD_TQ_LOSS_ECF_TRA	1	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm/100ms
TQ_LOSS_ECF_TRA limitation gradient					
IP_POW_LOSS_ECF_1	8	0 ... FF H	0 ... -5100	20	W
LDP_CFAPWM_CFA__POW_LOSS_ECF_1	8	0 ... FF H	0 ... 99.6	0.39	%
ECF power consumption versus CFAPWM_CFA					
C_LGRD_TQ_LOSS_ECF	1	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm/100ms
ECF torque loss limitation gradient					
C_N_MIN_TQ_LOSS_ECF	1	0 ... FF H	0 ... 8160	32	rpm
Engine speed threshold for activation of ECF torque loss calculation					

D.11 Torque Loss for power steering

D.11.1 Torque loss compensation with pressure switch signal (CONF_PSTE = 0)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_PSTE	V/O	8000 .. 0 H	-1024 .. 0	0.03125	Nm
Engine torque losses power steering					
TQ_LOSS_PSTE_BAS	V/O	8000 .. 0 H	-1024 .. 0	0.03125	Nm
Engine torque losses power steering					
TQ_LOSS_PSTE_TRA	V/O	8000 .. 0 H	-1024 .. 0	0.03125	Nm
Engine torque losses power steering					
TQ_ADD_PSTE	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Engine torque reserve for power steering					

Input data:

LV_PSTE	N	TCO	LV_IS
CONF_PSTE	T_AST	VS	

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_PSTE describes the needed torque during activation of power steering.

When the load of the power steering changes, an additional TQ_LOSS is applied in order to ensure a rapid compensation. This is performed by the TQ_LOSS_PSTE_TRA term.

High pressure in the power steering circuit: **LV_PSTE = 1**

Low pressure in the power steering circuit: **LV_PSTE = 0**

Additionally a torque reserve term, TQ_ADD_PSTE may be used to provide futher capacity to compensate rapidly changing torque requirements.

Application conditions

Activation : all engine operating state if TCO >= C_TCO_MIN_PSTE

CONF_PSTE = 0

Update Rate: 10 ms

Initialization : TQ_LOSS_PSTE = 0

TQ_LOSS_PSTE_BAS = 0

TQ_LOSS_PSTE_TRA = 0

TQ_ADD_PSTE = 0

Formula section:

1. TQ_LOSS_PSTE_BAS calculation

if LV_PSTE = 0

then TQ_LOSS_PSTE_BAS = 0 Nm

else TQ_LOSS_PSTE_BAS = IP_TQ_LOSS_PSTE_BAS

2. TQ_LOSS_PSTE_TRA and TQ_ADD_PSTE calculation

If Transition LV_PSTE: 0 -> 1

then TQ_LOSS_PSTE_TRA is set to IP_TQ_LOSS_PSTE_TRA_ACT and reset to 0 using change limitation C_TQ_LOSS_PSTE_TRA_LGRD.

If LV_IS = 1 **and** VS < C_VS_MAX_TQ_ADD_PSTE

Then TQ_ADD_PSTE is set to IP_TQ_ADD_PSTE_TRA_ACT

Endif

TQ_ADD_PSTE is reset to 0 using change limitation C_TQ_ADD_PSTE_LGRD

If Transition LV_PSTE: 1 -> 0

then TQ_LOSS_PSTE_TRA = 0.

TQ_ADD_PSTE = 0

3. TQ_LOSS_PSTE calculation

TQ_LOSS_PSTE = (TQ_LOSS_PSTE_BAS + TQ_LOSS_PSTE_TRA) * IP_FAC_T_AST_TQ_LOSS_PSTE(T_AST)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_LOSS_PSTE_BAS	10	0...8000H	-1024...0	0.03125	Nm
LDP_N__TQ_LOSS_PSTE_BAS	10	0 .. 1FE0 H	0...8160	1	1/min
Power steering torque loss versus engine-speed					
IP_TQ_LOSS_PSTE_TRA_ACT	6	0...8000H	-1024...0	0.03125	Nm
LDPM_TCO__TQ_LOSS_PSTE	6	0...FEH	-48...142,5	0.75	°C
Power steering torque loss with LV_PSTE transition 0->1					
C_TQ_LOSS_PSTE_TRA_LGRD	1	0...8000H	0...1024	0.03125	Nm
TQ_LOSS_PSTE_TRA reset change limitation					
C_TQ_ADD_PSTE_LGRD	1	0...8000H	0...1024	0.03125	Nm
TQ_ADD_PSTE reset change limitation					
C_TCO_MIN_PSTE	1	0...FEH	-48...142,5	0.75	°C
Minimum coolant temperature for power steering correction activation					
C_VS_MAX_TQ_ADD_PSTE	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for applying torque reserve for powersteering pressure switch type					
IP_TQ_ADD_PSTE_TRA_ACT	6	0...7FFFH	0... 1023.97	0.03125	Nm
LDPM_TCO__TQ_LOSS_PSTE	6	0...FEH	-48...142,5	0.75	°C
Torque reserve for power steering					
IP_FAC_T_AST_TQ_LOSS_PSTE	6	0..FFH	0..1	3.90625e-3	-
LDP_T_AST_IP_FAC_TQ_LOSS_PSTE	6	0...FFFFH	0...6553.5	0.1	s
After start factor for power steering torque losses					

D.11.2 Torque loss compensation with analog signal from pressure transducer (CONF_PSTE = 1)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_PSTE_PSPT	V/O	8000...0 H	-1024...0	0.03125	Nm
Engine torque losses power steering					
TQ_LOSS_PSTE_BAS_PSPT	V/O	8000...0 H	-1024...0	0.03125	Nm
Engine torque losses power steering					
TQ_LOSS_PSTE_TRA_PSPT	V/O	8000...0 H	-1024...0	0.03125	Nm
Engine torque losses power steering					
TQ_ADD_PSTE_PSPT	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Engine torque reserve for power steering					

Input data:

LV_PSP	N	TCO	PSP_GRD_MMV
PSP	CONF_PSTE	T_AST	LV_IS
VS			

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_PSTE PSPT describes the needed torque during activation of power steering.

When the load of the power steering changes, an additional TQ_LOSS is applied in order to ensure a rapid compensation. This is performed by the TQ_LOSS_PSTE_TRA_PSPT term.

Additionally a torque reserve term, TQ_ADD_PSTE_PSPT may be used to provide futher capacity to compensate rapidly changing torque requirements.

Application conditions

Activation : all engine operating state TCO >= C_TCO_MIN_PSTE

CONF_PSTE = 1

Update Rate: 10 ms

Initialization : TQ_LOSS_PSTE_PSPT = 0

TQ_LOSS_PSTE_BAS_PSPT = 0

TQ_LOSS_PSTE_TRA_PSPT = 0

TQ_ADD_PSTE_PSPT = 0

Formula section:

1. TQ_LOSS_PSTE_BAS_PSPT calculation

if LV_PSP = 0
then TQ_LOSS_PSTE_BAS_PSPT = 0 Nm
else TQ_LOSS_PSTE_BAS_PSPT = IP_TQ_LOSS_PSTE_BAS_PSPT

2. TQ_LOSS_PSTE_TRA_PSPT calculation

If LV_PSP = 0
then TQ_LOSS_PSTE_TRA_PSPT = 0 Nm
else TQ_LOSS_PSTE_TRA_PSPT = IP_TQ_LOSS_PSTE_TRA_PSPT

3. TQ_LOSS_PSTE_PSPT calculation

TQ_LOSS_PSTE_PSPT = (TQ_LOSS_PSTE_BAS_PSPT
+ TQ_LOSS_PSTE_TRA_PSPT) *
IP_FAC_T_AST_TQ_LOSS_PSTE_PSPT

4. TQ_ADD_PSTE_PSPT calculation

if LV_PSP = 0 **or** LV_IS = 0 **or** VS >= C_VS_MAX_TQ_ADD_PSTE_PSPT
then TQ_ADD_PSTE_PSPT = 0 Nm
else TQ_ADD_PSTE_PSPT = IP_ TQ_ADD_PSTE_PSPT

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MAX_TQ_ADD_PSTE_PSPT	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for applying torque reserve for powersteering pressure sensor type					
IP_TQ_LOSS_PSTE_BAS_PSPT	8*6	0 .. 8000H	-1024 .. 0	0.03125	Nm
LDPM_N__TQ_LOSS_PSTE	8	0...1FE0H	0 .. 8160	1	1/min
LDP_PSP__TQ_LOSS_PSTE	6	0...FFH	0...25.5	0.1	MPa
Power steering torque loss versus engine-speed and PSP					
IP_TQ_LOSS_PSTE_TRA_PSPT	6	0 .. 8000H	-1024 .. 0	0.03125	Nm
LDPM_PSP_GRD_MMV__TQ_LOSS_PSTE	6	8000...7FFFH	-1882...1881,94	14,7/256	MPa/s
Power steering torque loss with LV_PSTE transition 0->1					
IP_TQ_ADD_PSTE_PSPT	6	0...7FFFH	0... 1023.97	0.03125	Nm
LDPM_PSP__TQ_LOSS_PSTE	6	0...FFH	0...25.5	0.1	MPa
LDPM_PSP_GRD_MMV__TQ_LOSS_PSTE	6	8000...7FFFH	-1882...1881,94	14,7/256	MPa/s
Torque reserve for power steering					
IP_FAC_T_AST_TQ_LOSS_PSTE_PSPT	6	0..FFH	0..1	3.90625e-3	-
LDP_T_AST_IP_FAC_TQ_LOSS_PSTE	6	0...FFFFH	0...6553.5	0.1	s
After start factor for power steering torque losses					

D.12 Torque Loss – Alternator load

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_ALTER	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Engine torque losses alternator					
TQ_LOSS_ALTER_BAS	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Basic engine torque losses alternator					
TQ_LOSS_ALTER_TRA	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Transient correction for engine torque losses alternator					
LV_TQ_LOSS_ALTER_TRA	V	0 ... 1H	0 ... 1	1	-
Activation condition for transient alternator torque losses					
VB_REF	V	0 ... FFH	0 ... 26	0.102	Volt
Battery Voltage before electrical load occurs to determine amplitude of VB undershoot					
VB_DIF	V	0 ... FFH	0 ... 26	0.102	Volt
Difference between actual VB and VB before electrical load (amplitude of VB undershoot)					
TQ_ADD_ALTER	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Engine torque reserve for alternator					

Input data:

GEN_LOAD_MMV	LV_ES	LV_IGK	GEN_LOAD
LV_ERR_GEN_LOAD	N	N_SP_IS	LV_IS
VB	VB_MMV	GEN_LOAD_GRD_MMV	VS

FUNCTION DESCRIPTION:

General information:

TQ_LOSS_ALTER represents the engine torque loss due to power consumption of the alternator. TQ_LOSS_ALTER is generated by the Load Signal of the generator. To avoid engine speed undershoot in Idle, a transient correction is applied in case of sudden jump of Alternator Load.

Application conditions:

Activation: LV_IGK = 1

Deactivation: LV_IGK = 0 **and** LV_ES = 1

Initialisation: at reset -> TQ_LOSS_ALTER = TQ_LOSS_ALTER_BAS = TQ_LOSS_ALTER_TRA = 0, VB_REF = 14 Volt, VB_DIF = 0

TQ_ADD_ALTER = 0

Update Rate: 10 ms

Formula section:

** TQ_LOSS_ALTER_BAS calculation :*

TQ_LOSS_ALTER_BAS = IP_TQ_LOSS_ALTER__GEN_LOAD_MMV__N
* IP_FAC_TQ_LOSS_ALTER__VB_MMV__N

The change of TQ_LOSS_ALTER_BAS is limited by limitation gradient C_LGRD_TQ_LOSS_ALTER.

** TQ_LOSS_ALTER_TRA calculation :*

TQ_LOSS_ALTER_TRA is used only in Idle and set in case of sudden jump of GEN_LOAD signal causing an engine speed undershoot.

- Activation condition of TQ_LOSS transient correction (LV_TQ_LOSS_ALTER_TRA = 1) :

If LV_TQ_LOSS_ALTER_TRA = 0
and LV_IS = 1
and GEN_LOAD_MMV < C_GEN_LOAD_MMV_ALTER_TRA_MAX
and GEN_LOAD > C_GEN_LOAD_TRA_MIN % sudden jump of GEN_LOAD
and N <= N_SP_IS – C_N_SP_IS_ALTER_TRA *% engine speed undershoot*
Then LV_TQ_LOSS_ALTER_TRA = 1

- Deactivation condition of TQ_LOSS transient correction (LV_TQ_LOSS_ALTER_TRA = 0) :

If LV_TQ_LOSS_ALTER_TRA = 1
and **(** LV_IS = 0 **or** N > N_SP_IS – C_N_SP_IS_ALTER_TRA **)**
Then LV_TQ_LOSS_ALTER_TRA = 0

- Determination of the Battery Voltage undershoot to determine TQ_LOSS transient correction :

If Transition LV_TQ_LOSS_ALTER_TRA = 0 -> 1

Then $VB\_REF_{(n)} = VB_{(n-10)}$ *% determine VB 100 msec. before electrical load occured*

Else VB_REF remains constant

If LV_TQ_LOSS_ALTER_TRA = 1

Then **If** VB < VB_REF

and VB_REF – VB > VB_DIF

then VB_DIF = VB_REF – VB *% detremine max. VB undershoot*

else VB_DIF remains unchanged *% during the transient correction*

Else VB_DIF = 0

- Determination of the Transient TQ_LOSS correction TQ_LOSS_ALTER_TRA :

If LV_TQ_LOSS_ALTER_TRA = 1

Then TQ_LOSS_ALTER_TRA is set to IP_TQ_LOSS_ALTER_TRA__VB_DIF

Else TQ_LOSS_ALTER_TRA is ramped down to 0 with change limitation C_LGRD_TQ_LOSS_ALTER_TRA.

** TQ_LOSS_ALTER calculation :*

If no failure on Generator Load signal is present, then TQ_LOSS_ALTER is sum of TQ_LOSS_ALTER_BAS and TQ_LOSS_ALTER_TRA.

If LV_ERR_GEN_LOAD = 0

Then TQ_LOSS_ALTER = TQ_LOSS_ALTER_BAS + TQ_LOSS_ALTER_TRA

Else TQ_LOSS_ALTER = 0

<u> TQ_ADD_ALTER calculation :</u>*

If a transient in generator load is detected, then a higher torque reserve can be used to improve idle quality following this transient.

If LV_IS = 1 **and** VS < C_VS_MAX_TQ_ADD_ALTER

Then TQ_ADD_ALTER = ID_TQ_ADD_ALTER(GEN_LOAD_GRD_MMV)

Else TQ_ADD_ALTER = 0 (via gradient limitation)

TQ_ADD_ALTER is gradient limited (in the negative direction only) via C_LGRD_TQ_ADD_ALTER. In the positive direction there is no requirement for gradient limitation.

<u>Calibration data:</u>

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
IP_TQ_LOSS_ALTER	8*8	0 ... FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_GEN_LOAD_MMV__TQ_LOSS_ALTER	8	0 ... FFH	0 ... 100	0.39	%
LDPM_N__TQ_LOSS_ALTER	8	0 ... 1FE0H	0 ... 8160	1	rpm
Alternator torque losses					
IP_FAC_TQ_LOSS_ALTER	6*8	0 ... FF H	0 ... 0.996	0.0039	-
LDP_VB_MMV__TQ_LOSS_ALTER	6	0 ... FFH	0 ... 26	0.102	V
LDPM_N__TQ_LOSS_ALTER	8	0 ... 1FE0H	0 ... 8160	1	rpm
Weighting factor for VB_MMV					
C_LGRD_TQ_LOSS_ALTER	1	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm/10ms
Alternator torque losses limitation gradient					
C_GEN_LOAD_MMV_ALTER_TRA_MAX	1	0 ... FFH	0 ... 99.6	0.4	%
Max. GEN_LOAD_MMV value to apply transient alternator load torque losses					
C_GEN_LOAD_TRA_MIN	1	0 ... FFH	0 ... 99.6	0.4	%
Min. GEN_LOAD value to apply transient alternator load torque losses					
C_N_SP_IS_ALTER_TRA	1	0 ... 1FE0H	0 ... 8160	1	rpm
Engine speed negative offset to Nominal Idle Speed to stop alternator transient torque losses					
C_VS_MAX_TQ_ADD_ALTER	1	0...FFH	0...255	1	km/h
Maximum vehicle speed for applying torque reserve for alternator					
IP_TQ_LOSS_ALTER_TRA	4	0 ... FFFF H	-1024 .. 1023.97	0.03125	Nm
LDP_VB_DIF__TQ_LOSS_ALTER_TRA	4	0 ... FFH	0 ... 26	0.102	V
Weighting factor for VB_MMV					
C_LGRD_TQ_LOSS_ALTER_TRA	1	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm/10ms
Transient alternator torque losses limitation gradient					
ID_TQ_ADD_ALTER	4	0 ... 7FFF H	0 .. 1023.97	0.03125	Nm
LDP_GEN_LOAD_GRD_MMV	4	0 ... FFFFH	-100 ... 99.6	3.051e-3	%/10ms
Weighting factor for VB_MMV					
C_LGRD_TQ_ADD_ALTER	1	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm/10ms
Alternator torque reserve limitation gradient					

D.13 Torque based catalyst heating

Overview

The torque model calculates the spark retardation for catalyst heating depending on air mass flow increase.

The air mass flow increase depends on the torque reserve which is generated in this module. The actually indicated torque TQI_AV will not be influenced.

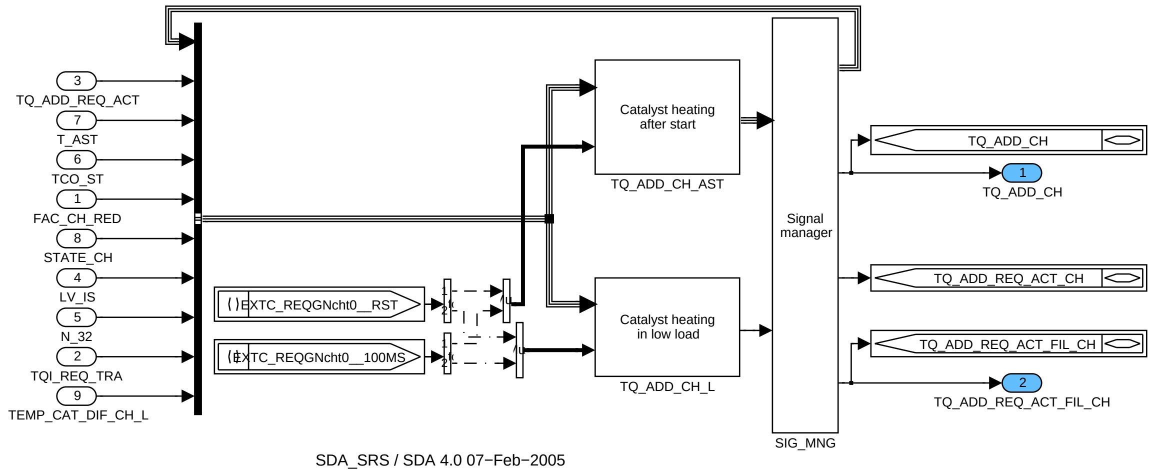


Figure 10 EXTC_REQGNcht0

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_CH	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Torque reserve for catalyst heating					
TQ_ADD_REQ_ACT_FIL_CH	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Filtered realised torque reserve request for ignition path input					
TQ_ADD_REQ_ACT_CH	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Minimum possible torque reserve request for ignition path input					

Input data:

FAC_CH_RED	TQI_REQ_TRA	TQ_ADD_REQ_ACT	LV_IS
N_32	TCO_ST	T_AST	STATE_CH
TEMP_CAT_DIF_CH_L			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_ADD_CH_FIL_LGRD	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
limitation gradient for TQ_ADD_CH_FIL (TQ_ADD_CH_FIL for the ignition path)					
C_TQ_ADD_CH_LGRD	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Limitation gradient for TQ_ADD_CH deactivation ramp					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_ADD_CH_L	8	0...FFFFH	0...1.99996948	3.05176E-5	-
LDP_TEMP_CAT_DIF_CH_L_IP_FAC_TQ	8	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Actual catalyst temperature depending correction of torque reserve for low load CH					
IP_FAC_TQ_ADD_CH_AST	8x8	0...FFH	0...0.99609375	0.00390625	-
LDPM_T_AST_2_EXTC	8	0...FFFFH	0...6.5535E+3	0.1	s
LDPM_TCO_ST_3_EXTC	8	0...FEH	-48...142.5	0.75	°C
Coolant temperature and time after start depending correction of TQ_ADD_CH_AST					
IP_TQ_ADD_CH_AST	8x8	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
LDPM_N_32_1_EXTC	8	0...FFH	0...8.16E+3	32	rpm
LDPM_TQI_REQ_TRA_1_EXTC	8	0...7FFFH	0...1.02397E+3	0.03125	Nm
Basic torque reserve for after start catalyst heating					
IP_TQ_ADD_CH_AST_IS	8x8	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
LDPM_N_32_2_EXTC	8	0...FFH	0...8.16E+3	32	rpm
LDPM_TQI_REQ_TRA_2_EXTC	8	0...7FFFH	0...1.02397E+3	0.03125	Nm
Basic torque reserve for after start catalyst heating in idle speed					
IP_TQ_ADD_CH_L	8x8	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
LDPM_N_32_1_EXTC	8	0...FFH	0...8.16E+3	32	rpm
LDPM_TQI_REQ_TRA_1_EXTC	8	0...7FFFH	0...1.02397E+3	0.03125	Nm
Basic torque reserve for low load catalyst heating					
IP_TQ_ADD_CH_L_IS	8x8	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
LDPM_N_32_2_EXTC	8	0...FFH	0...8.16E+3	32	rpm
LDPM_TQI_REQ_TRA_2_EXTC	8	0...7FFFH	0...1.02397E+3	0.03125	Nm
Basic torque reserve for low load catalyst heating in idle speed					

D.13.1 Catalyst heating after start

Application Condition

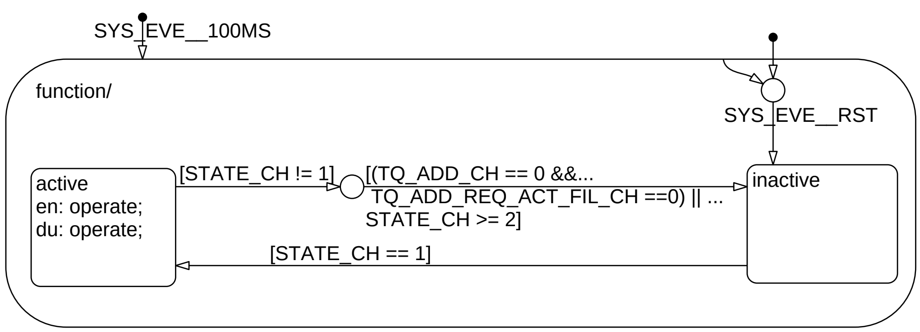


Figure 11 EXTC_REQGNcht0/ TQ_ADD_CH_AST/ APP_CDN/ Chart

Function Description

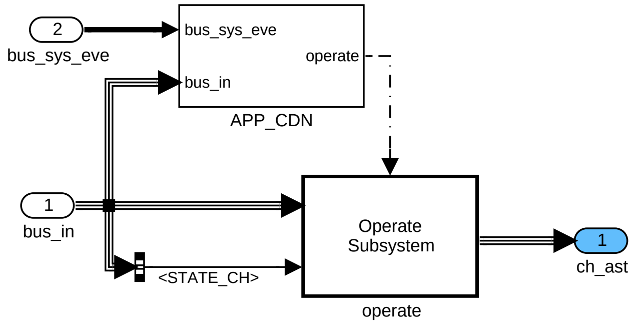


Figure 12 EXTC_REQGNcht0/ TQ_ADD_CH_AST

D.13.1.1 Operate Subsystem

The function is divided into active catalyst heating and deactive – for ramping torque reserve to 0.

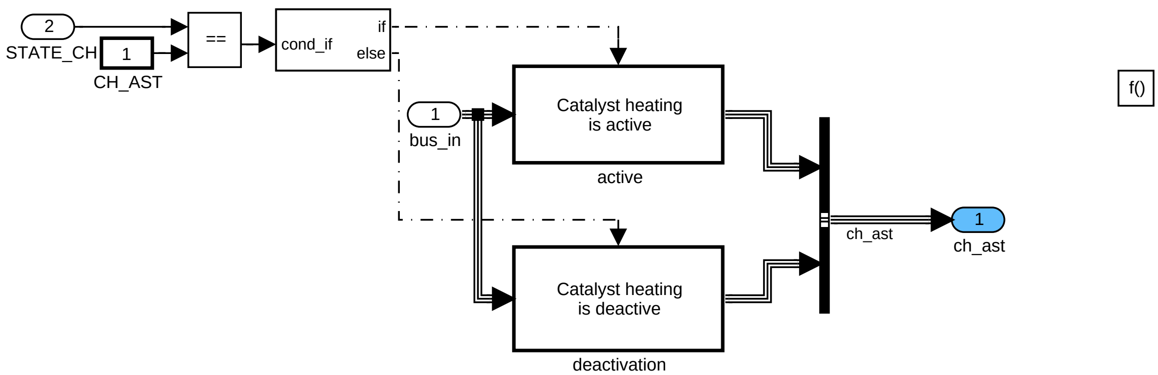


Figure 13 EXTC_REQGNcht0/ TQ_ADD_CH_AST/ operate

Catalyst heating is active

TQ_ADD_REQ_ACT_FIL_CH cannot be used for an ignition increase at dynamic torque request. It can be set to 0 with C_TQ_ADD_CH_FIL_LGRD = 0.

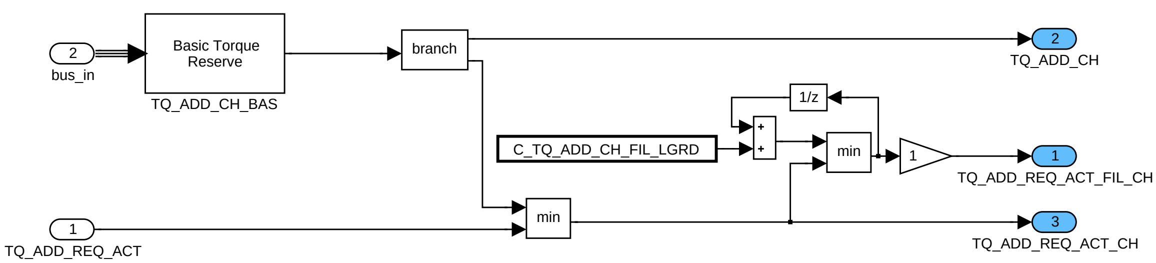


Figure 14 EXTC_REQGNcht0/ TQ_ADD_CH_AST/ operate/ active/ SUB

Basic Torque Reserve

FAC_CH_RED can be adapted by the project.

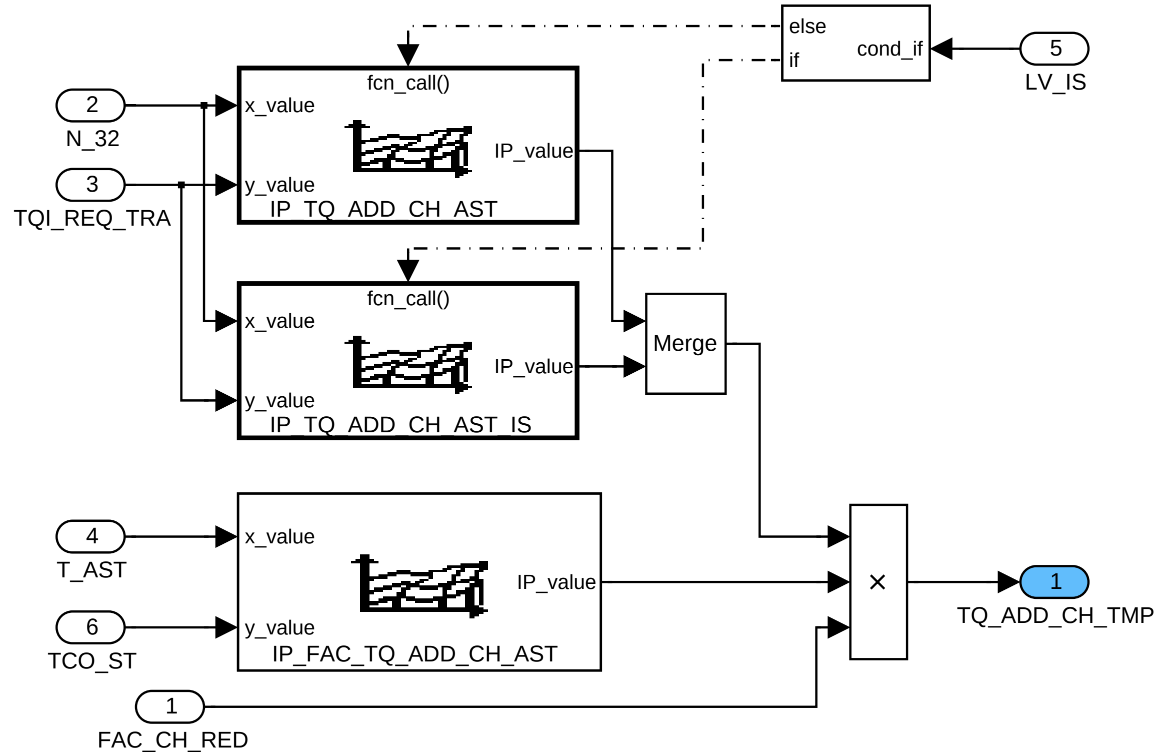


Figure 15 EXTC_REQGNcht0/ TQ_ADD_CH_AST/ operate/ active/ SUB/ TQ_ADD_CH_BAS/ SUB

Catalyst heating is deactive

TQ_ADD_CH and TQ_ADD_REQ_ACT_FIL_CH are ramped down. TQ_ADD_REQ_ACT is the actual torque reserve request.

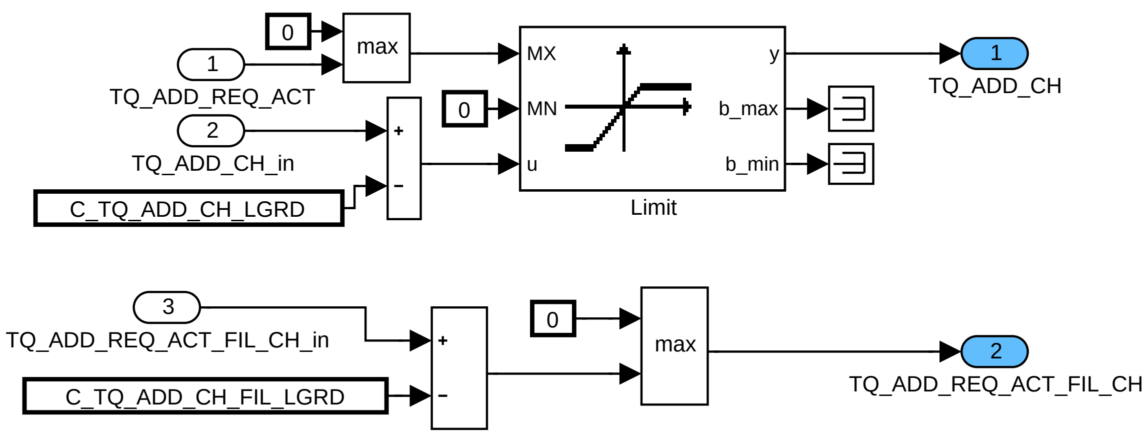


Figure 16 EXTC_REQGNcht0/ TQ_ADD_CH_AST/ operate/ deactivation/ SUB

D.13.2 Catalyst heating in low load

This function can be activated to avoid a catalyst temperature lower than its working temperature.

Application Condition

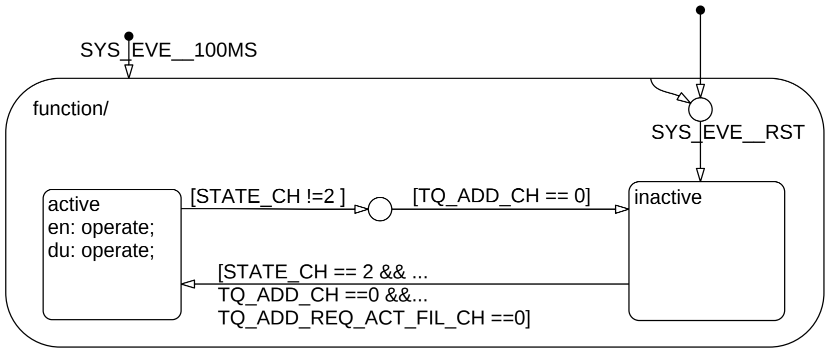


Figure 17 EXTC_REQGNcht0/ TQ_ADD_CH_L/ APP_CDN/ Chart

Function Description

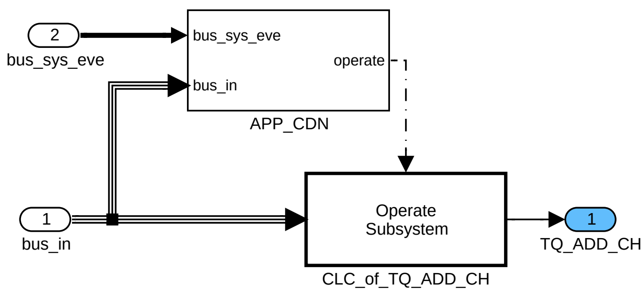


Figure 18 EXTC_REQGNcht0/ TQ_ADD_CH_L

D.13.2.1 SUBFUNCTION: CLC_of_TQ_ADD_CH

Operate Subsytem

The function is in principle a P - controller. The output is filtered, the difference from one calculation to another is limited by C_TQ_ADD_CH_L_LGRD.

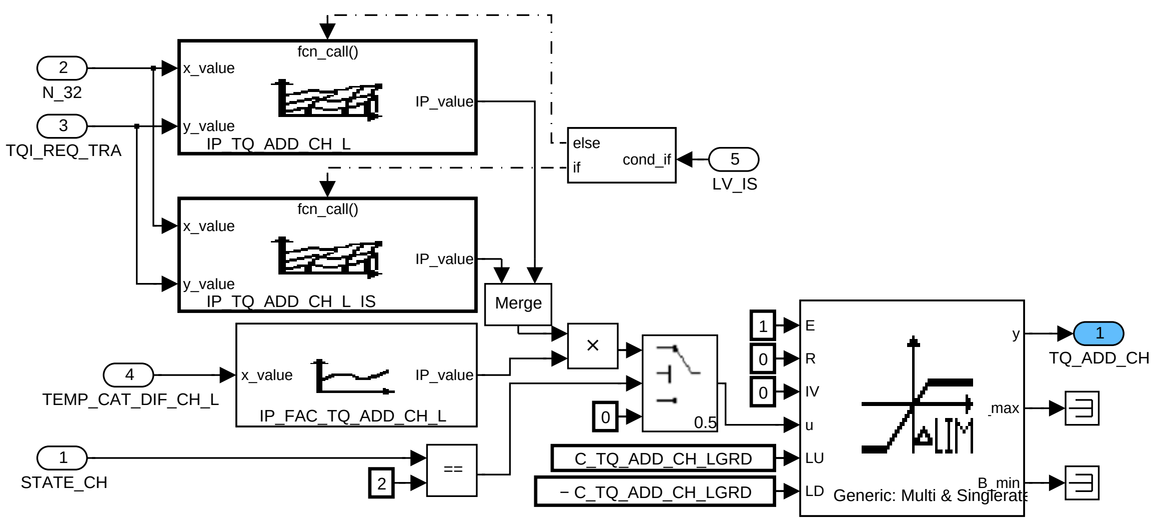


Figure 19 EXTC_REQGNcht0/ TQ_ADD_CH_L/ CLC_of_TQ_ADD_CH/ SUB

D.14 Torque based catalyst heating (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_CH_RED	V/O	0...FFH	0...4	0.015625	-
Factor to reduce the catalyst heating					
FAC_CH_RED_AT	V	0...FFH	0...4	0.015625	-
Factor to reduce the catalyst heating for automatic transmission cars					
FAC_CH_RED_DRI	V	0...FFH	0...4	0.015625	-
Factor to reduce the catalyst heating due to DRIve selected or not					
FAC_CH_RED_ACIN	V	0...FFH	0...4	0.015625	-
Factor to reduce the catalyst heating while A/C is requested					
FAC_CH_RED_COR	V	0...FFH	0...4	0.015625	-
Factor to reduce the catalyst heating due to ambient conditions (AMP, TIA)					
TQ_ADD_MIN_CH	V/O	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Minimum torque reserve for catalyst heating					
LV_TQ_ADD_MIN_CH_ACT	V/O	0...1H	0...1	1	[-]
Request for minimum torque reserve for catalyst heating is active					

Input data:

AMP	LV_DRI	LV_CH	LV_AT
TIA	LV_RLY_ACCOUT	LV_IS	TQ_ADD_CH

FUNCTION DESCRIPTION:

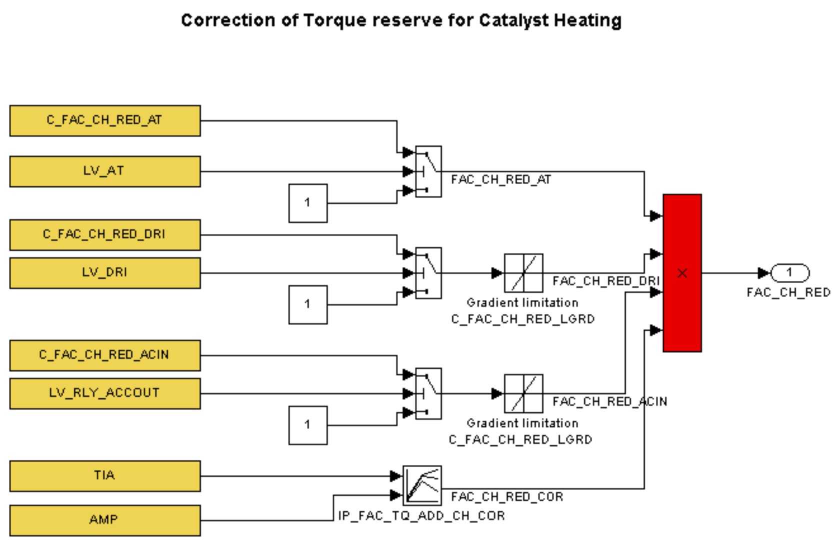
General information:

For active catalytic converter heating a spark retardation is necessary to increase the exhaust gas temperature. The reduced engine torque due to spark retardation must be compensated by increased intake mass air flow to have no influence on torque at clutch.

This function describes the output data FAC_CH_RED. This factor can reduce the torque reserve for catalyst heating for different ambient conditions and for automatic transmission depending on whether LV_DRI is selected or not, and whether the A/C is selected or not.
In order to avoid any jumps of the torque reserve as LV_DRI or LV_RLY_ACCOUT are selected, a gradient is applied on FAC_CH_RED_DRI and FAC_CH_RED_ACIN. However, FAC_CH_RED_DRI remains active even if neutral is selected anew (braking performances).

The calculation of the reduction factor starts with LV_CH=1, but does not end when LV_CH is deactivated (1->0), since the torque reserve for catalyst heating TQ_ADD_CH is still applied for a while through a deactivation ramp. Therefore in this case, the factor FAC_CH_RED must be kept frozen.

Signal flow diagramm:



Application conditions:

Initialisation: *At reset*

FAC_CH_RED = 1

FAC_CH_RED_AT=1

FAC_CH_RED_DRI=1

FAC_CH_RED_ACIN=1

FAC_CH_RED_COR=1

TQ_ADD_MIN_CH = 0

LV_TQ_ADD_MIN_CH_ACT = 0

Activation: every engine state

Deactivation:

Update Rate: *100 ms*

Formula section:

```
if        LV_CH=1
then      if      LV_AT=1
          then
                  FAC_CH_RED_AT= C_FAC_CH_RED_AT
                  if      LV_DRI=1
                  then    FAC_CH_RED_DRI =
                     max(FAC_CH_RED_DRI(n-1)-C_FAC_CH_RED_LGRD;C_FAC_CH_RED_DRI)
                  else
                          FAC_CH_RED_DRI = FAC_CH_RED_DRI(n-1)
                  endif
          else
                  FAC_CH_RED_AT = 1
          endif

          if      LV_RLY_ACCOUT = 1
          then    FAC_CH_RED_ACIN =
                  max(FAC_CH_RED_ACIN(n-1)-C_FAC_CH_RED_LGRD; C_FAC_CH_RED_ACIN)
          else    FAC_CH_RED_ACIN =
                  min(FAC_CH_RED_ACIN(n-1)+C_FAC_CH_RED_LGRD;1)
          endif

          FAC_CH_RED_COR = IP_FAC_TQ_ADD_COR_CH(TIA;AMP)
          FAC_CH_RED = FAC_CH_RED_AT * FAC_CH_RED_DRI * FAC_CH_RED_ACIN *
                                                         FAC_CH_RED_COR
          LV_TQ_ADD_MIN_CH_ACT = LC_TQ_ADD_MIN_CH_ENA
          if      LV_IS = 1
          then    TQ_ADD_MIN_CH = TQ_ADD_CH * C_FAC_TQ_ADD_MIN_CH
          else    TQ_ADD_MIN_CH = 0
          TQ_ADD_MIN_CH is gradient limited in both directions with C_TQ_ADD_MIN_CH_LGRD
else
          FAC_CH_RED=FAC_CH_RED(n-1)
          LV_TQ_ADD_MIN_CH_ACT = 0
          TQ_ADD_MIN_CH = 0 (via gradient limitation as above)
endif
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_ADD_COR_CH	6x6	0...FFH	0...4	0.015625	-
LDP_TIA_IP_FAC_TQ_ADD_COR_CH	6	0...FEH	-48...142.5	0.75	°C
LDP_AMP_IP_FAC_TQ_ADD_COR_CH	6	0...FFFFH	0...5434	0.083	HPa
Intake air temperature and ambient pressure depending correction TQ_ADD_CH					
C_FAC_CH_RED_AT	1	0...FFH	0...4	0.015625	-
Reduction of Torque reserve for CH in case of automatic transmission					
C_FAC_CH_RED_DRI	1	0...FFH	0...4	0.015625	-
Reduction of Torque reserve for CH in case of automatic transmission and LV_DRI selected					
C_FAC_CH_RED_ACIN	1	0...FFH	0...4	0.015625	-
Reduction of Torque reserve for CH in case of A/C selected					
C_FAC_CH_RED_LGRD	1	0...FFH	0...4	0.015625	-
Gradient in case of changes of reduction factor due to LV_DRI or LV_RLY_ACCOUT selected or unselected					
C_FAC_TQ_ADD_MIN_CH	1	0...FFH	0...0.99609	3.9062e-3	[-]
Factor on TQ_ADD_CH for limitation					
C_TQ_ADD_MIN_CH_LGRD	1	0...7FFFH	0...1023.96875	0.03125	[Nm]
Limitation gradient for TQ_ADD_MIN_CH					
LC_TQ_ADD_MIN_CH_ENA	1	0...1H	0...1	1	[-]
Manual switch for activation of minimum torque reserve for catalyst heating					

D.15 Torque request for gear shift intervention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_GS_ENA	V/O	0 ... 1H	0 ... 1	1	-
logical variable for gear shift intervention enabling					
LV_GS_TQI_INC_ENA	V/O	0 ... 1H	0 ... 1	1	-
logical variable for gear shift intervention enabling – Torque increment					
TQI_GS_FAST_DEC	V/O	0...7FFFH	0...1023.97	0.03125	Nm
torque decrement for fast torque intervention during gear shift					
TQI_GS_SLOW_DEC	V/O	0...7FFFH	0...1023.97	0.03125	Nm
torque decrement for slow torque intervention during gear shift					
TQI_GS_FAST_INC	V/O	0...7FFFH	0...1023.97	0.03125	Nm
torque increment for fast torque intervention during gear shift					
TQI_GS_SLOW_INC	V/O	0...7FFFH	0...1023.97	0.03125	Nm
torque increment for slow torque intervention during gear shift					
TQI_AV_WOUT_GS	V	0...7FFFH	0...1023.97	0.03125	Nm
Actual engine torque without gear shift					
TQI_DIF_GS_FAST	V	0...7FFFH	0...1023.97	0.03125	Nm
Torque difference between driver request and fast gearshift intervention					
T_GS_ENA	V	0...FFH	0...2.55	0.01	s
Time for gearshift intervention					
TQI_GS_SLOW_CLC	V	0...7FFFH	0...1023.97	0.03125	Nm
Calculated torque decrement for slow torque intervention during gear shift					
FAC_TQI_GS_SLOW	-	0...FFH	0...1	0.00392	-
Intermediate calculation of slow path torque factor					

Input data:

LV_IGK	LV_ST	LV_IS	LV_ES
LV_DRI	TQI_GS_REQ	TQI_AV	TQI_TCU_CAN
TAR_GC	N_32	MAF_HB	TQI_GS_INC_REQ
TQI_TCU_MSR_CAN	LV_ASR_REQ	LV_MSR_REQ	SPK_RTD_TCU
TQI_GS_SLOW_REQ	ETL_TCU_CAN	CONF_ETL_TCU	TQI_REQ_TRA
TQI_BAS_MAX	PV_AV		

FUNCTION DESCRIPTION:

General information:

During a gear shift operation of the automatic transmission a low engine torque helps the transmission to perform a smooth gear shift and avoids sudden changes of wheel slip that could cause problems on certain road surfaces, i.e. wet or icy roads. The transmission control unit (TCU) demands a torque reduction via CAN, which can be realized by spark retard (fast path) or throttle closing/opening (slow path).

A torque intervention for gear shift is enabled depending on the engine state, current engine torque and the value of the requersted torque.

If the GS torque reduction intervention function is enabled (LV_GS_ENA = 1), then TQI_GS_REQ, which is received from CAN, is at first limited to IP_TQI_GS_MIN_FAST_DEC (or IP_TQI_GS_MIN_FAST_DEC_ESP if there is ESP torque request) then copied into TQI_GS_FAST_DEC. Then the requested torque is adjusted only by ignition retardation.

In case of available engine torque reduction request for slow path from the TCU (CONF_ETL_TCU = 1), TQI_GS_SLOW_DEC is used to reduce the engine torque in some specific cases (ex. stall RPM check or Accel. just after N/D shift) to protect the automatic transmission. Alternatively (CONF_ETL_TCU = 2) a proportion of the fast path torque reduction can also be implemented in the slow path. The proportion used is a function of engine speed, duration of gearshift and gas pedal position.

If the GS torque increment intervention function is enabled (LV_GS_TQI_INC_ENA = 1), then TQI_GS_INC_REQ, which is received from CAN, is at first limited to IP_TQI_GS_MAX_INC then copied into TQI_GS_FAST/SLOW_INC. Then the requested torque is adjusted by ignition advance and Throttle opening.

Application conditions:

Initialisation:	at reset	LV_GS_ENA	= 0
		TQI_GS_FAST_DEC	= 1023,97 Nm
		TQI_GS_SLOW_DEC	= 1023.97 Nm
		TQI_GS_FAST_INC	= 0 Nm
		TQI_GS_SLOW_INC	= 0 Nm
		T_GS_ENA	= 0 s
Recurrence:	10 ms		
Activation:	at LV_IGK = 1		
Deactivation:	-		

Formula section:

TQI_AV_WOUT_GS = min **(** TQI_REQ_TRA, max (TQI_BAS_MAX, TQI_AV)**)**

If LV_ES = 0
and LV_ST = 0
and LV_DRI = 1
and [{ LV_IS = 0 } **or** { LV_IS = 1 **and** (TAR_GC = 3 **or** TAR_GC = 2)}
and TQI_TCU_CAN < 100% (TQ_STND) *(GS-requested torque below TQ_STND)*
or SPK_RTD_TCU <> FFH
or **(** ETL_TCU_CAN <> FFH **and** CONF_ETL_TCU = 1 **)**

then

LV_GS_ENA = 1 *(Gear shift intervention enabled)*

/ Fast path/

if LV_ASR_REQ = 1 or LV_MSR_REQ = 1

then *(ASR / MSR request active)*

TQI_GS_FAST_DEC = max **(** TQI_GS_REQ;
IP_TQI_GS_MIN_FAST_DEC_ESP(N_32,MAF_HB) **)**

else *(no ASR / MSR request active)*

```
            TQI_GS_FAST_DEC = max ( TQI_GS_REQ;
                                   IP_TQI_GS_MIN_FAST_DEC(N_32,MAF_HB) )
        endif
        /Slow path/
        if      CONF_ETL_TCU = 0         (No slow intervention)
        then
                TQI_GS_SLOW_DEC = 1023,97 Nm
        elseif CONF_ETL_TCU = 1    (Intervention with TQI_GS_SLOW_REQ)
        then
                if      LV_ASR_REQ = 1 or LV_MSR_REQ = 1
                then  (ASR / MSR request active)
                        TQI_GS_SLOW_DEC = max( TQI_GS_SLOW_REQ;
                                  IP_TQI_GS_MIN_SLOW_DEC_ESP(N_32,MAF_HB) )
                else (no ASR / MSR request active)
                        TQI_GS_SLOW_DEC = max ( TQI_GS_SLOW_REQ;
                                  IP_TQI_GS_MIN_SLOW_DEC(N_32,MAF_HB))
                endif
        elseif CONF_ETL_TCU = 2    (Proportion of fast signal TQI_GS_REQ is used)
        then
                FAC_TQI_GS_SLOW = IP_FAC_TQI_GS_SLOW_A(N_32, T_GS_ENA) *
                        IP_FAC_TQI_GS_SLOW_B(N_32, PV_AV) )
                if      FAC_TQI_GS_SLOW = 0 (no slow path intervention)
                then TQI_GS_SLOW_DEC = 1023.97 Nm
                else
                    TQI_DIF_GS_FAST =  max(0, ( TQI_AV_WOUT_GS - TQI_GS_REQ ))
                    TQI_GS_SLOW_CLC = TQI_AV_WOUT_GS -
                            ( TQI_DIF_GS_FAST * FAC_TQI_GS_SLOW)
                    TQI_GS_SLOW_DEC = max ( TQI_GS_SLOW_CLC,
                            IP_TQI_GS_SLOW_DEC(N_32,PV_AV))
                endif
                increment T_GS_ENA (10 ms)
        endif
else    LV_GS_ENA = 0                   (Gear shift intervention not enabled)
        TQI_GS_FAST_DEC = 1023,97 Nm
        TQI_GS_SLOW_DEC = 1023,97 Nm
        T_GS_ENA = 0 s
endif

if            LV_ES = 0
```

and	LV_ST = 0	
and	{ LV_IS = 1 **and** (TAR_GC = 4 **or** TAR_GC = 5)}	
and	LV_DRI = 1	*(With A/T: Drive selected)*
and	TQI_TCU_MSR_CAN >0%	

then LV_GS_TQI_INC_ENA = 1 *(Gear shift intervention enabled)*

TQI_GS_FAST_INC =

min (TQI_GS_INC_REQ; IP_TQI_GS_MAX_INC(N_32))

TQI_GS_SLOW_INC =

min (TQI_GS_INC_REQ; IP_TQI_GS_MAX_INC(N_32))

else LV_GS_TQI_INC_ENA = 0 *(Gear shift intervention not enabled)*

TQI_GS_FAST_INC = 0 Nm

TQI_GS_SLOW_INC = 0 Nm

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_GS_MIN_FAST_DEC	8x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32__TQI_GS_MIN	8	0 ... FF H	0 ... 8160	32	1/min
LDPM_MAF_HB__TQI_GS_MIN	8	0...FFH	0...1.389E3	5.447	mg/tdc
Minimum limitation for TQI_GS_FAST_DEC					
IP_TQI_GS_MIN_FAST_DEC_ESP	8x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32__TQI_GS_MIN	8	0 ... FF H	0 ... 8160	32	1/min
LDPM_MAF_HB__TQI_GS_MIN	8	0...FFH	0...1.389E3	5.447	mg/tdc
Minimum limitation for TQI_GS_FAST_DEC					
IP_TQI_GS_MIN_SLOW_DEC	8x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32__TQI_GS_MIN	8	0 ... FF H	0 ... 8160	32	1/min
LDPM_MAF_HB__TQI_GS_MIN	8	0...FFH	0...1.389E3	5.447	mg/tdc
Minimum limitation for TQI_GS_SLOW_DEC (CONF_ETL_TCU=1)					
IP_TQI_GS_MIN_SLOW_DEC_ESP	8x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32__TQI_GS_MIN	8	0 ... FF H	0 ... 8160	32	1/min
LDPM_MAF_HB__TQI_GS_MIN	8	0...FFH	0...1.389E3	5.447	mg/tdc
Minimum limitation for TQI_GS_SLOW_DEC (CONF_ETL_TCU=1)					
IP_TQI_GS_MAX_INC	1x8	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32__TQI_GS_MIN	8	0 ... FF H	0 ... 8160	32	1/min
Maximum limitation for TQI_GS_FAST/SLOW_INC					
IP_FAC_TQI_GS_SLOW_A	6 x 6	0...FFH	0...1	0.00392	-
LDPM_N_32_TQI_GS_SLOW	6	0...FFH	0...8160	32	rpm
LDP_T_GS_ENA_TQI_GS_SLOW	6	0...FFH	0...2.55	0.01	s
Scaling factor for gear shift slow path intervention (0 = no intervention)					
IP_FAC_TQI_GS_SLOW_B	6x6	0...FFH	0...1	0.00392	-
LDPM_N_32_TQI_GS_SLOW	6	0...FFH	0...8160	32	rpm
LDP_PV_AV_ TQI_GS_SLOW	6	0 ... FF H	0 ... 99.6	0.3906	%
Scaling factor based on the pedal value during gear shift intervention (0 = no intervention)					
IP_TQI_GS_SLOW_DEC	6 x 6	0...7FFFH	0...1023.97	0.03125	Nm
LDP_N_32_ TQI_GS_SLOW_DEC	6	0...FFH	0...8160	32	rpm
LDP_PV_AV_ TQI_GS_SLOW_DEC	6	0 ... FF H	0 ... 99.6	0.3906	%
Maximum limitation for TQI_GS_SLOW_DEC (CONF_ETL_TCU=2)					

D.16 Torque request for traction control (ASR)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ASR_ACT	V/O	0 ... 1H	0 ... 1	1	-
logical variable for ASR intervention enabling					
TQI_ASR_FAST	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Torque decrement for fast torque intervention during ASR					
TQI_ASR_SLOW	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Torque decrement for slow torque intervention during ASR					
TQI_ASR_BOL	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Minimum indicated torque requested during ASR intervention					
TQI_ASR_SLOW_CLC	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Calculated Torque decrement for slow torque intervention during ASR					
TQI_DIF_EMS_ASR_FAST	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Value for torque difference between TQI_EMS and TQI_ASR_FAST					
T_DLY_TQI_ASR_SLOW_FAST	V/O	0...FFH	0...2.55	0.01	S
Time for Torque Difference between TQI_ASR_SLOW and TQI_ASR_FAST					

Input data:

LV_IGK	LV_ERR_IV_x	LV_ERR_MAF	LV_ERR_LOAD_PLAUS
LV_ERR_CPS	TCO	TQI_SLW_TCS_CAN	LV_ASR_REQ
LV_ERR_CAN	TQI_TCS_CAN	LC_TCS1_OFF	LV_PL
LV_N_LIM_ETC_LIH	LV_OFF_MTC_MON	TQI_ASR_REQ	TQI_ASR_SLW_REQ
LC_TQI_ASR_REQ_MAN	N_SP_IS_RATIO_ASR	TQ_LOSS	TQ_CONV
TQI_AV	TQI_AV_EMS	CONF_ASR_SLOW	N_32
LV_ERR_MAP			

FUNCTION DESCRIPTION:

General information:

The ASR torque reduction is performed only when LV_ASR_ACT = 1 (see chapter "Minimum/ Maximum Torque Request Selection").

TQI_ASR_BOL is used to limit the minimum ASR Torque reduction request to avoid engine stall (engine speed stabilized to N_SP_IS + C_N_ADD_ASR_SP).

In case a TCS control unit can send only a fast TQ reduction request (CONF_ASR_SLOW = 0), the EMS can calculate a slow TQ reduction request out of the fast request.

Application conditions:

Initialisation: at reset LV_ASR_ACT = 0

TQI_ASR_FAST = 1023.97 Nm

TQI_ASR_SLOW = 1023.97 Nm

TQI_ASR_BOL = 1023.97 Nm

TQI_ASR_SLOW_CLC = 1023.97 Nm

TQI_DIF_EMS_ASR_FAST = 1023.97 Nm

T_DLY_TQI_ASR_SLOW_FAST = 0 s

Recurrence: 10 ms

Activation: at LV_IGK = 1

Deactivation: -

Formula section:

TQI_ASR_FAST / TQI_ASR_SLOW definition :

If LC_TQI_ASR_REQ_MAN = 1
or **(** LC_TCS1_OFF = 0
and LV_PL = 1
and LV_ERR_CAN = 0
and LV_ERR_IV_x = 0
and LV_ERR_MAF = 0
and LV_ERR_MAP = 0
and LV_ERR_LOAD_PLAUS = 0
and LV_ERR_CPS = 0
and LV_N_LIM_ETC_LIH = 0
and LV_OFF_MTC_MON = 0
and N > IP_N_MIN_ASR_ACT
and TQI_AV > TQI_ASR_BOL - C_TQI_BOL_ASR_HYS
and **(** TQI_TCS_CAN < 100% **or** TQI_SLW_TCS_CAN < 100% **)**
and LV_ASR_REQ = 1 **)**
then LV_ASR_ACT = 1 ***(ASR Torque Reduction active)***
and TQI_ASR_FAST = max (TQI_ASR_REQ ; TQI_ASR_BOL)
and **if** CONF_ASR_SLOW = 1 *(Slow Path ASR Request available)*
then TQI_ASR_SLOW = max (TQI_ASR_SLW_REQ ; TQI_ASR_BOL)
else TQI_DIF_EMS_ASR_FAST = ABS (TQI_AV_EMS - TQI_ASR_FAST)
TQI_ASR_SLOW_CLC = TQI_ASR_FAST
+(TQI_DIF_EMS_ASR_FAST * IP_FAC_TQI_ASR_SLOW_FAST)
TQI_ASR_SLOW = max (TQI_ASR_SLOW_CLC; TQI_ASR_BOL)
increment T_DLY_TQI_ASR_SLOW_FAST (10 ms)
else LV_ASR_ACT = 0 ***(ASR Torque Reduction inactive)***
and TQI_ASR_FAST = 1023.97Nm
and TQI_ASR_SLOW = 1023.97 Nm
and TQI_ASR_SLOW_CLC = 1023.97 Nm
and TQI_DIF_EMS_ASR_FAST = 1023.97 Nm
and T_DLY_TQI_ASR_SLOW_FAST = 0 s

** TQI_ASR_BOL definition :*

TQI_ASR_BOL = TQ_CONV – TQ_LOSS

+ IP_FAC_N_DIF_IS_ASR * (TQ_CONV – TQ_LOSS)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQI_BOL_ASR_HYS	1	0...7FFFH	0...1023.97	0.03125	Nm
TQI hysterisis to TQI_ASR_BOL for activation of ASR					
IP_FAC_N_DIF_IS_ASR	8	0...FFFFH	-8...7.9998	0.00024414	-
LDP_N_SP_IS_RATIO_ASR	8	0...FFFFH	0...7.9999	0.000122	-
Idle self stabilization factor in case of ASR intervention					
IP_FAC_TQI_ASR_SLOW_FAST	6 x 6	0...FFH	0...1	0.00392	-
LDP_N_32_FAC_TQI_ASR_SLOW_FAST	6	0...FFH	0...8160	32	Rpm
LDP_T_DLY_TQI_ASR_SLOW_FAST	6	0...FFH	0...2.55	0.01	s
IP_N_MIN_ASR_ACT	4	0...1FE0H	0...8160	1	rpm
LDP_TCO_N_MIN_ASR_ACT	4	0...FEH	-48...142.5	0.75	°C
Mininum engine speed for ASR intervention					

D.17 Torque Request for Cruise Control

D.17.1 Torque request of EMS internal Cruise control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_REQ_CRU	V/O	0 .. FFFF H	0 .. 1.999969	3.0517E-5	-
scaling factor for requested torque at clutch from cruise control					

Input data:

FAC_TQ_CRU	LV_CRU_ACT	VS_CRU_FIL	VS_SP_DRIV_CRU
FAC_TQ_REQ_DRIV	FAC_TQ_CRU_INI	CONF_CRU	

General information:

Depending on the cruise controller manipulated variable FAC_TQ_CRU the interpretation of torque request is calculated. Regarding the different resolutions the output of cruise control is converted to resolutions at the torque structure interface.

If the Cruise Control is deactivated by resetting CONF_CRU, then the FAC_TQ_REQ_CRU is set to 0.

Application conditions:

Activation: every engine state

CONF_CRU = 1

Recurrence: 10 ms

Initialization: FAC_TQ_REQ_CRU = 0 at reset

Deactivation: **If** CONF_CRU = 0

then FAC_TQ_REQ_CRU = 0

Formula section:

General calculation after initialisation:

$$FAC\_TQ\_REQ\_CRU = \frac{FAC\_TQ\_CRU}{99{,}6\%}$$

At initialisation of Cruise control:

To get a smooth transition into cruise controlled drive the initialisation ramp is, depending on vehicle speed, in relation to the vehicle speed demand by the driver:

This query once :

IF (1) LV_CRU_ACT = 0 to 1

and FAC_TQ_CRU / 99,6% > FAC_TQ_REQ_DRIV

Then (1.1) at the first time FAC_TQ_REQ_CRU = FAC_TQ_REQ_DRIV

This query until the end condition (until (1)) is fulfilled:

IF (2) VS_CRU_FIL ≤ VS_SP_DRIV_CRU

Then (2)

FAC_TQ_REQ_CRU = FAC_TQ_REQ_CRU + C_LGRD_TQ_CRU_INC_1

Else (2)

FAC_TQ_REQ_CRU = FAC_TQ_REQ_CRU + C_LGRD_TQ_CRU_INC_2

Until (1.1) FAC_TQ_REQ_CRU >= FAC_TQ_CRU / 99,6%

Else (1)

$$FAC\_TQ\_REQ\_CRU = \frac{FAC\_TQ\_CRU}{99{,}6\%}$$

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LGRD_TQ_CRU_INC_1	V/O	0 .. FFFF H	0 .. 1.999969	3.0517E-5	-
Change limitation for cruise control requesrt out of lower vehicle speed					
C_LGRD_TQ_CRU_INC_2	V/O	0 .. FFFF H	0 .. 1.999969	3.0517E-5	-
Change limitation for cruise control requesrt out of higher vehicle speed					

D.18 Torque request for MSR

FUNCTION DESCRIPTION:

To avoid the vehicle from skidding (on road surfaces with low friction coefficient, e.g. ice) during PU or PUC due to high engine friction torque, the drag torque has to be reduced.

By increasing the indicated engine torque, wheel slip is reduced, leading to a higher lateral stability of the wheels

D.18.1 Enable conditions for MSR intervention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MSR_ENA	V/O	0...1H	0...1	1	-
Logical variable for MSR intervention enabling					

Input data:

LV_MSR_INH_INT_MSR	LV_MSR_INH_T_MSR	LC_INH_MSR	LV_ST
LV_ASR_REQ	LV_ERR_TPS	TQI_MSR_CAN	VS
LV_MSR_INH_MON	LV_IGK	LV_ERR_CAN	LC_TQI_MSR_REQ_MAN
LV_ES	LV_N_LIM_ETC_LIH	LV_OFF_MTC_MON	

FUNCTION DESCRIPTION:

General information:

The MSR intervention function is enabled if the vehicle speed is higher than C_VS_MIN_MSR and no errors at relevant components are recognized and MSR must not be inhibited by safety relevant functions (time counter, plausibiltity check, Monitoring of MSR)

Furthermore no ASR-request and no TPS-failure is active

Application conditions:

Initialisation: at reset

LV_MSR_ENA = 0

Recurrence: 10 ms

Activation: LV_IGK = 1

Formula section:

If LC_TQI_MSR_REQ_MAN = 1

or

(LV_ES = LV_ST = 0

and LV_MSR_REQ = 1

and LV_ASR_REQ = 0

and LV_MSR_INH_T_MSR = 0

and LV_MSR_INH_INT_MSR = 0

and TQI_MSR_CAN < IP_TQI_MSR_CAN_MAX

and LC_INH_MSR = 0

and LV_ERR_TPS = 0

and LV_ERR_CAN = 0

and VS > C_VS_MIN_MSR

and LV_N_LIM_ETC_LIH = 0

and LV_OFF_MTC_MON = 0

and LV_MSR_INH_MON = 0)

Then LV_MSR_ENA = 1

Else LV_MSR_ENA = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MIN_MSR	1	0...FFH	0...255	1	km/h
Minimum vehicle speed for MSR activation (default 10 km/h)					
IP_TQI_MSR_CAN_MAX	1x6	0...FFH	0...100	0.39	%
LDP_N_32_TQI_MSR_CAN_MAX	6	0...FFH	0...8.16E3	32	rpm
Allowed Maximum of MSR request depends on engine speed					

D.18.2 MSR intervention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_MSR	V/O	0...7FFFH	0...1023.97	0.03125	Nm
Value for fast torque intervention due to MSR					
LV_MSR_ACT	V/O	0.1H	0.1	1	-
Logical variable for torque intervention due to MSR					

Input data:

LV_MSR_REQ	LV_MSR_ENA	LV_IGK	TQI_TCS_CAN
TQI_MSR_CAN	TQI_MSR_REQ	LC_TQI_MSR_REQ_MAN	

FUNCTION DESCRIPTION:

General information:

If following conditions are fulfilled, the torque intervention for MSR is performed(LV_MSR_ACT = 1),

1) a torque intervention request from the MSR is recognized through LV_MSR_REQ = 1
2) the MSR intervention function is enabled (LV_MSR_ENA = 1)
3) The sum of the complemental value of TQI_MSR_CAN through TQI_TCS_CAN and TQI_MSR_CAN is FFh.

These variables are handed over to the relevant torque coordination functions.

Application conditions:

Initialisation: at reset
LV_MSR_ACT = 0
TQI_MSR = 0 Nm

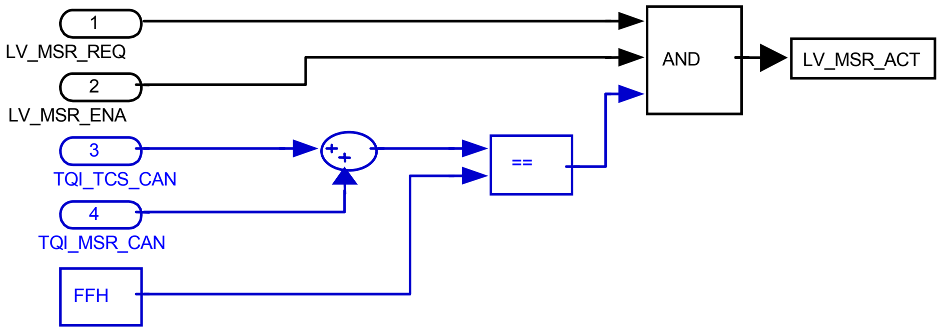
Recurrence: 10 ms

Activation: LV_IGK = 1

Formula section:

If LC_TQI_MSR_REQ_MAN = 1
or
(LV_MSR_REQ = 1
and LV_MSR_ENA = 1
and (TQI_TCS_CAN + TQI_MSR_CAN) = FFh)
Then LV_MSR_ACT = 1
and TQI_MSR = TQI_MSR_REQ
Else LV_MSR_ACT = 0
and TQI_MSR = 0 Nm

Signal flow diagram:



D.18.2.1 MSR Plausibility check

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_MSR_INT	V/O	0...FFFFH	0...655.35	0.01	Nms
MSR requested torque x Zeit					
LV_MSR_INH_INT_MSR	V/O	0...1H	0...1	1	-
Bool for non plausible MSR torque request					

Input data:

TQ_LOSS	TQI_MSR_REQ	LV_ES	LV_MSR_REQ

FUNCTION DESCRIPTION:

General information:

The MSR torque intervention has to be checked through an integral TQI_MSR_REQ x Time. This integral must not be equal or higher than a special threshold.

Application conditions:

Activation: all engine operating states

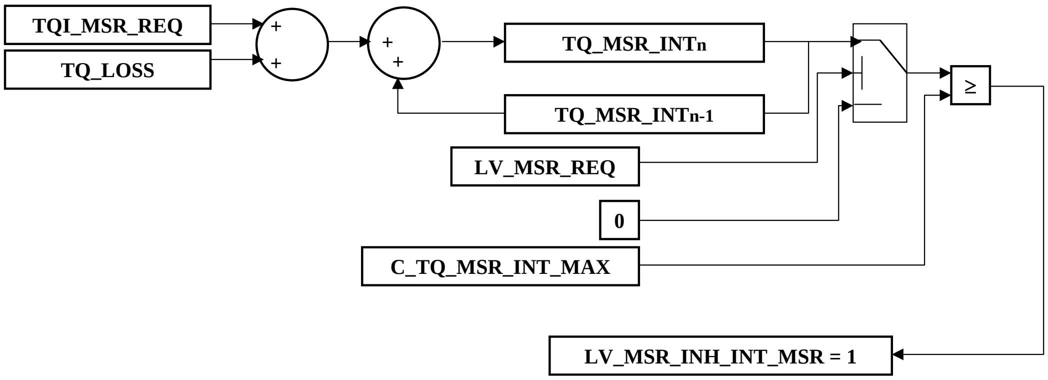
Initialisation: with LV_ES = 1

TQ_MSR_INT = 0

LV_MSR_INH_INT_MSR = 0

Recurrence: 10 ms

Signal flow diagram:



Formula section:

If LV_MSR_REQ = 0

then TQ_MSR_INT = 0

else TQ_MSR_INT (n) = TQ_MSR_INT (n-1) + 10 *ms* * (TQI_MSR_REQ + TQ_LOSS)

If TQ_MSR_INT >= C_TQ_MSR_INT_MAX

then LV_MSR_INH_INT_MSR = 1 **till** LV_MSR_REQ = 0 again.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_MSR_INT_MAX	1	0...FFFFH	0...655.35	0.01	Nms
Theshold for MSR intervention is not plausible					

D.18.2.2 Check the MSR torque intervention time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MSR_INH_T_MSR	V/O	0...1H	0...1	1	-
Flag for inhibition MSR- request due to too long MSR torque intervention request					
T_MSR	V	0...FFFFH	0...655.35	0.01	s
Integrated time during MSR torque intervention					

Input data:

LV_MSR_REQ	LV_IGK		

FUNCTION DESCRIPTION:

General information:

If the MSR request is continued longer than a threshold, and LV_MSR_INH_T_MSR is set to 1 and the MSR request is ignored until LV_MSR_REQ = 0.

Application conditions:

Initialisation: at LV_IGK = 0 → 1

LV_MSR_INH_T_MSR = 0

Recurrence: 10 ms

Activation: all engine operating states

Formula section:

If LV_MSR_REQ = 0

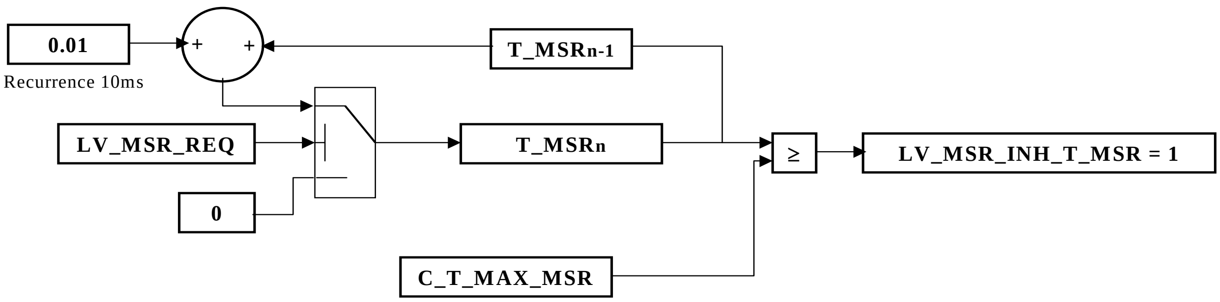
then T_MSR = 0

else T_MSR (n) = T_MSR (n-1) + 10 *ms*

If T_MSR >= C_T_MAX_MSR

then LV_MSR_INH_T_MSR = 1 **till** LV_MSR_REQ = 0 again.

Signal flow diagram:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_MAX_MSR	1	0...FFFFH	0...655.35	0.01	s
Maximum time for MSR torque intervention					

D.19 Torque based turbine overheating prevention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_TEG_MAX_TUR	O/V	0...7FFFH	0...1.02397E+3	0.03125	Nm
Indicated torque limit due to turbine overheating prevention					
LV_TEG_MAX_TCHA_DEAC	O/V	0...1H	0...1	1	-
Turbine temperature was too high - turbocharger shell be deactivated					
LV_TQI_TEG_MAX_TUR	O/V	0...1H	0...1	1	-
Boolean for active turbine overheating prevention via torque setpoint reduction					
TQI_TEG_MAX_TUR_I	V	0...7FFFFFFFH	0...1.024E+3	4.76837E-7	Nm
I-part of controller for turbine overheating prevention					
TQI_TEG_MAX_TUR_P	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
P-part of indicated torque for turbine overheating prevention					
TEG_TUR_OHP_DIF_ACT	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Turbine temperature difference threshold for activation of overheating prevention					
TEG_TUR_OHP_SP_TRA	V	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Transient turbine temperature setpoint for overheating prevention					
TEG_TUR_OHP_DIF_TRA	V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Transient temperature difference (control deviation) for overheating prevention					

Input data:

LAMB_SP[NC_CBK_EX_NR]	TEG_DYN_UP_TUR[NC_CBK_EX_NR]	TQI_ADD_ACT	TEG_TUR_OHP_GRD
LV_LAMB_TUR_OHP_BOL[NC_CBK_EX_NR]	TEG_TUR_OHP_DIF[NC_CBK_EX_NR]	TEG_TUR_OHP_SP	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_TEG_TUR_OHP_SP_TRA	1	0...FFH	0...0.99609375	0.00390625	-
Filter constant for transient turbine temperature setpoint calculation					
C_FAC_TQI_TEG_MAX_TUR_I_NEG	1	0...FFFFH	0...0.25	3.81476E-6	Nm/(K*10 ms)
I-part of indicated torque controler for negative control deviation					
C_FAC_TQI_TEG_MAX_TUR_I_POS	1	0...FFFFH	0...0.25	3.81476E-6	Nm/(K*10 ms)
I-part of indicated torque controler for positive control deviation					
C_FAC_TQI_TEG_MAX_TUR_P_NEG	1	0...FFFFH	0...32	4.88289E-4	Nm/K
P-part of indicated torque controler for negative error variable					
C_FAC_TQI_TEG_MAX_TUR_P_POS	1	0...FFFFH	0...32	4.88289E-4	Nm/K
P-part of indicated torque controler for positive error value					
C_LAMB_SP_MAX_TQI_TEG_MAX_TUR	1	0...7FFFH	0...31.9990234	9.76563E-4	-
Maximum lambda setpoint for activation of torque intervention caused by exhaust gas temperature					
C_TEG_TUR_MAX_TCHA_DEAC	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Turbine temperature to deactivate the turbocharger					

general specification

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQI_TEG_MAX_TUR_I_MIN	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Minimum value of the I-part for turbo charger overheating prevention					
C_TQI_TEG_MAX_TUR_MIN	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Minimum torque limitation for turbo charger overheating prevention					
C_TQI_TEG_MAX_TUR_TCHA_DEAC	1	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum indicated torque for turbine overheating prevention - at deactivated turbocharger					
LC_TQI_TEG_WOUT_LAMB_ENA	1	0...1H	0...1	1	-
Enables torque limitation without reaching bottom limit of enrichment					
IP_TEG_TUR_OHP_DIF_ACT	6	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
LDP_TEG_TUR_GRD_IP_OHP_DIF_ACT	6	0...FFFFH	-1.024E+3 ... 1.02397E+3	0.03125	K
Turbine temperature difference threshold for activation of overheating prevention					

D.19.1 General information

The torque based turbine overheating prevention is an additional way after the lambda enrichment to reduce the turbine temperature. Therefore the information of the lambda turbo overheating prevention will be used for the activation and calculation of the TQ- intervention. If the lambda enrichment is not sufficient to reduce the exhaust gas temperature on the turbine the calculation of the torque intervention will be activated. Additional to this torque intervention the overspeed- and wastegate influence is considered on the charger side.

Application Condition

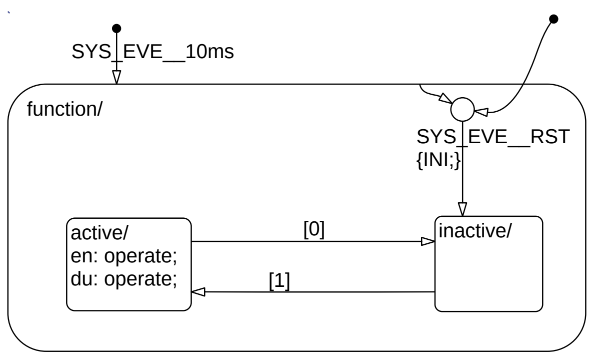


Figure 20 EXTC_ISPCLohp0/ APP_CDN/ Chart

Function Description

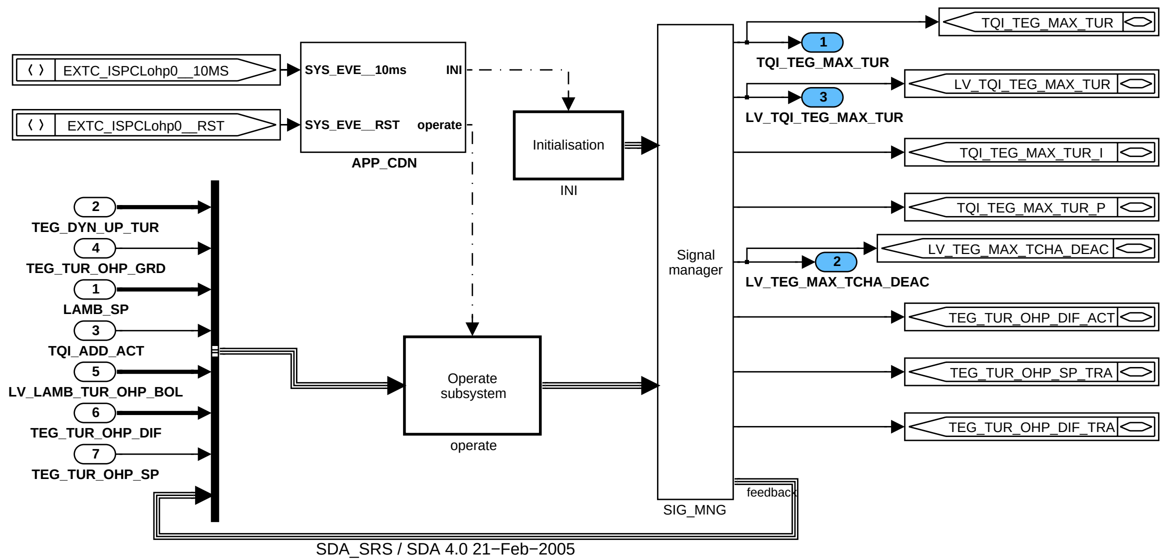


Figure 21 EXTC_ISPCLohp0

D.19.1.1 Initialisation

The outputs are initialised at reset.

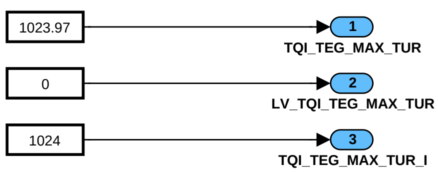


Figure 22 EXTC_ISPCLohp0/ INI/ SUB

D.19.1.2 Operate subsystem

If C_TEG_DYN_TUR_MAX_TCHA_DEAC is reached the turbo will be deactivated (function s. CHRG) and there will be a failure entry. The normal case is that the controller-function is calculated.

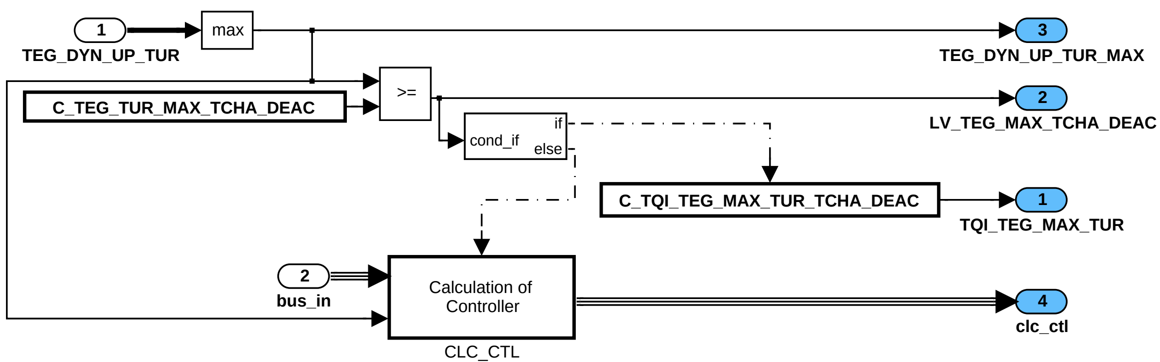


Figure 23 EXTC_ISPCLohp0/ operate/ SUB

Calculation of controller

The activation of the controller is depending on the temperature gradient. Via C_LAMB_SP_MAX_TQI_TEG_MAX_TUR you can calibrate a Lambda-setpoint that has to occur to activate the torque-limitation (1st the temperature-controller via lambda shell be active and reduce the lambda-setpoint). The I-part of the controller is ramped up after deactivation. As long as it not has reached TQI_ADD_ACT the controller stays active.

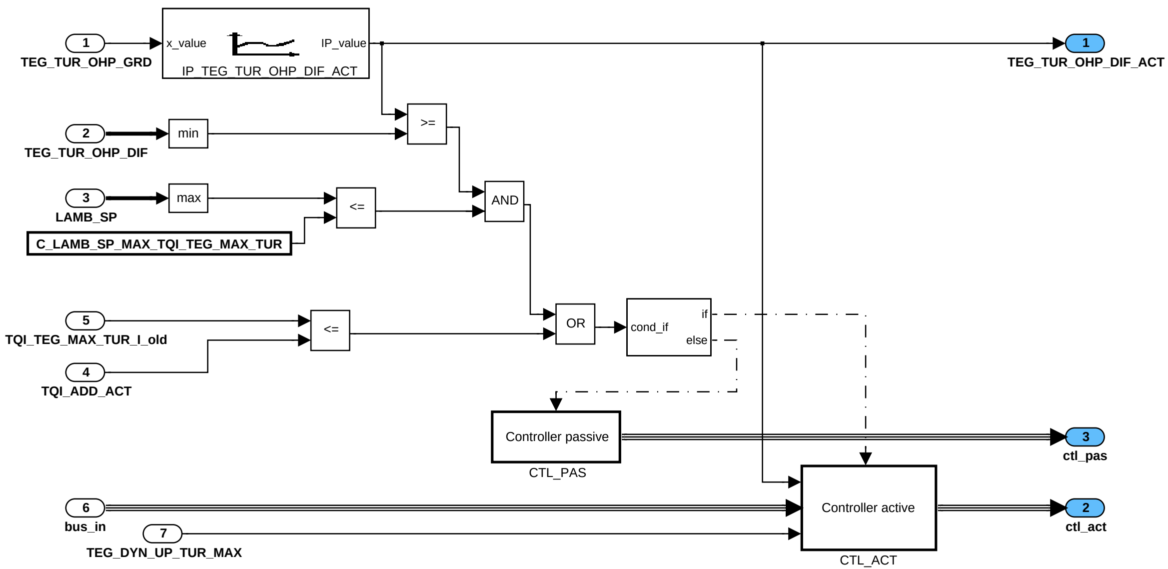


Figure 24 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB

Controller active

This function part is only calculated if the controller is active. LV_TQI_TEG_MAX_TUR is set to 1. Via [TEG_TUR_OHP_SP_TRA – turbine temperature] –the control deviation- it is decided if torque has to be reduced (TEG is too hot) or if the limitation can be increased (TEG is not too hot).

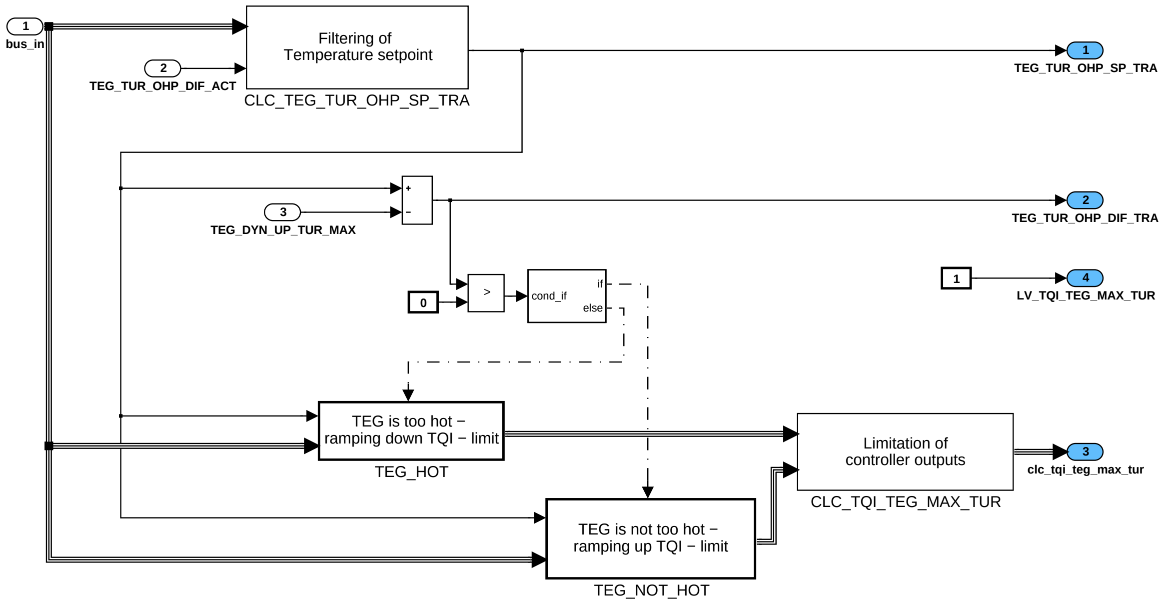


Figure 25 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_ACT/ SUB

Filtering of temperature setpoint

The temperature setpoint can be filtered. The first value after controller activation is the temperature setpoint (input from “Turbo charger overheating prevention” – via lambda enrichment) minus the gradient-depending activation-temperature-difference.

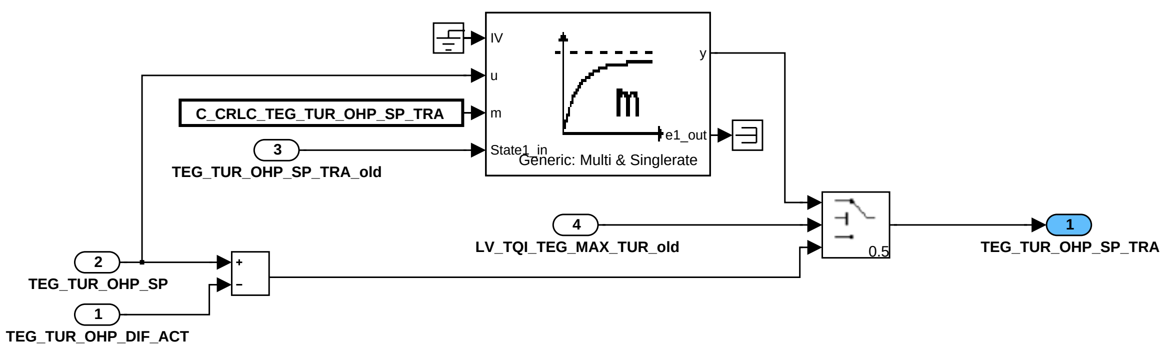


Figure 26 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_ACT/ SUB/ CLC_TEG_TUR_OHP_SP_TRA/ SUB

TEG is too hot - ramping down TQI - limit

The P-part is the product of a factor and the control deviation. The I-part is based on a minimum selection of the old value or the actual torque (inclusive torque reserve). The new term is also the product of a factor and the control deviation. With LC_TQI_TEG_WOUT_LAMB_ENA it is possible to choose to freeze the I-part or not in the case that the temperature-controller via lambda is not at its bottom limit.

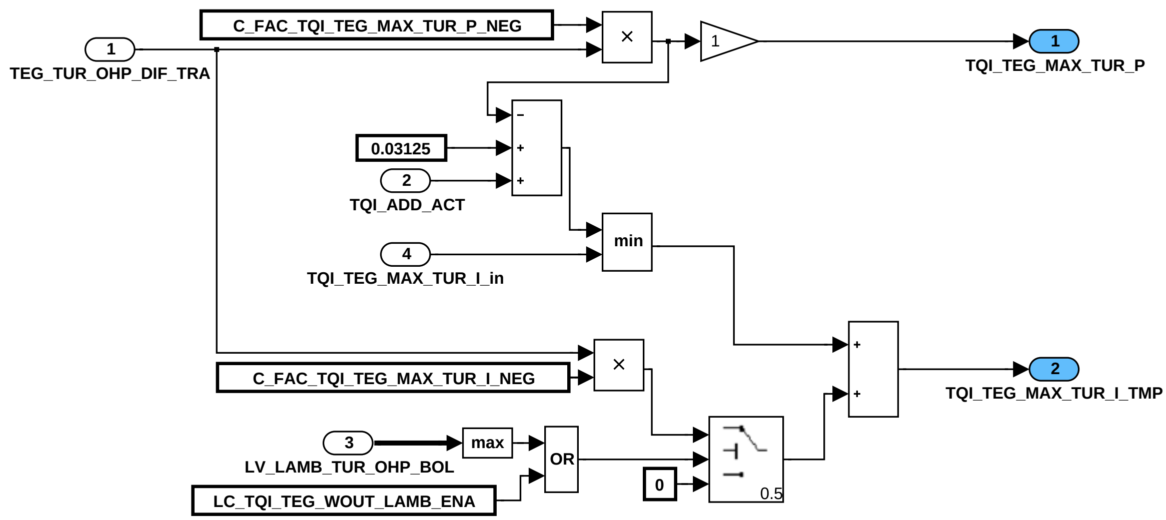


Figure 27 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_ACT/ SUB/ TEG_HOT/ SUB

TEG is not too hot - ramping up TQI - limit

There are several calibration data for down and up – ramping.

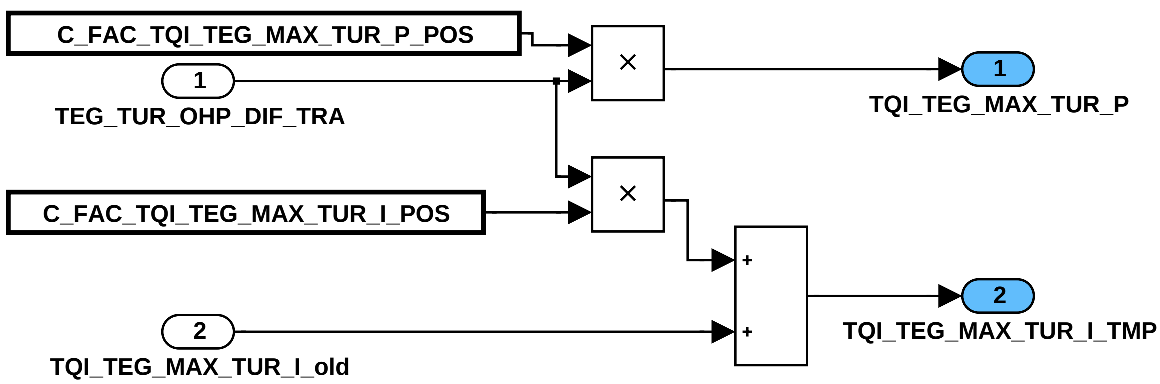


Figure 28 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_ACT/ SUB/ TEG_NOT_HOT/ SUB

Limitation of controller outputs

The I-part is limited. The addition with the P-part gives the controller output TQI_TEG_MAX_TUR. This value also is limited.

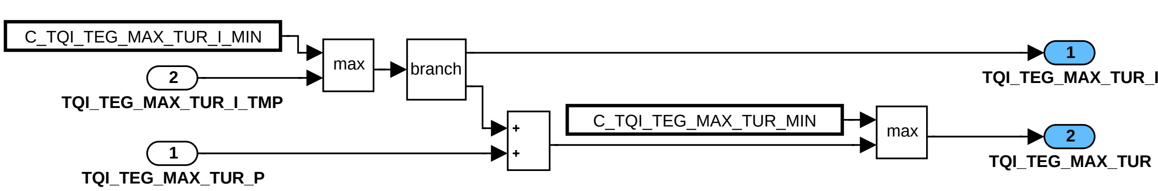


Figure 29 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_ACT/ SUB/ CLC_TQI_TEG_MAX_TUR/ SUB

Controller passive

If the temperature is in no critical range, the output variables are set to passive values.

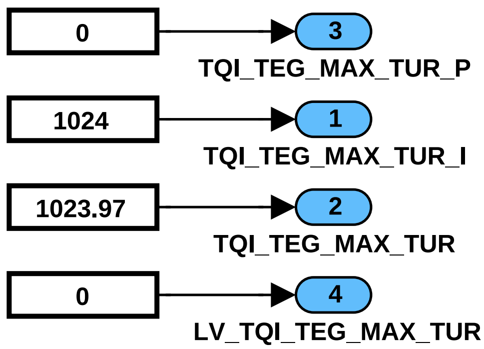


Figure 30 EXTC_ISPCLohp0/ operate/ SUB/ CLC_CTL/ SUB/ CTL_PAS/ SUB

D.20 Torque based overheating prevention

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_MAX_OHP	O/V	0...7FFFH	0...1.02397E+3	0.03125	Nm
Maximum indicated torque due to overheating prevention (of cat. or turbine)					
LV_TQI_OHP	O/V	0...1H	0...1	1	-
indicates if overheating prevention (of catalyst or turbo charger) is active					
LV_TQI_COP	V	0...1H	0...1	1	-
torque limitation due to COP requested					
POW_MAX_COP	V	0...FFFFH	0...439.22	0.0067020643	KW
maximum indicated power due to catalyst temperature					
TQI_TEG_MAX_COP	V	0...7FFFH	0...1.02397E+3	0.031250038	Nm
maximum indicated torque due to catalyst overheating prevention					

Input data:

TQI_TEG_MAX_TUR	LV_TQI_TEG_MAX_TUR	LAMB_COP[NC_CBK_EX_NR]	N_32
TEMP_CAT_DYN_MDL[NC_CBK_EX_NR]	TQI_AV		

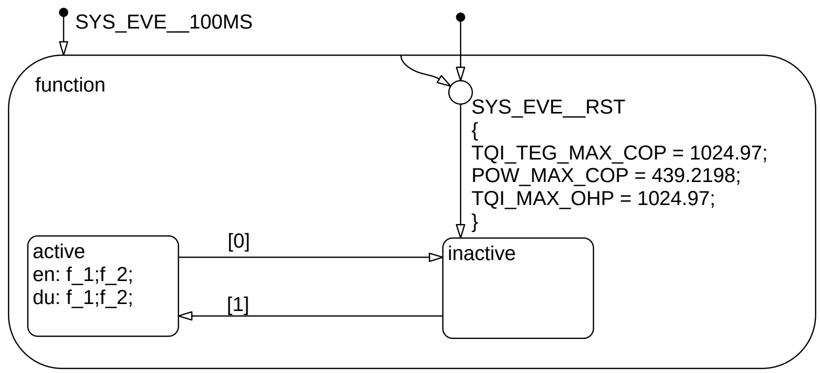
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LAMB_MIN_POW_COP	1	0...7FFFH	0...31.999	9.76563E-4	-
Lambda threshold for activation of COP power limitation					
C_POW_COP_DELTA	1	0...FFH	0...17.0903	0.0670206	kW/s
De- or increase of power limitation ramp					
C_POW_COP_MIN	1	0...FFFFH	0...439.22	0.0067020643	KW
minimum indicated power during torque reduction for COP					
C_TEG_MIN_POW_COP	1	0...7FFFH	-273.15 ... 1.77479E+3	0.0625	°C
Threshold of exhaust gas temperature for activation of COP power limitation					

D.20.1 General information

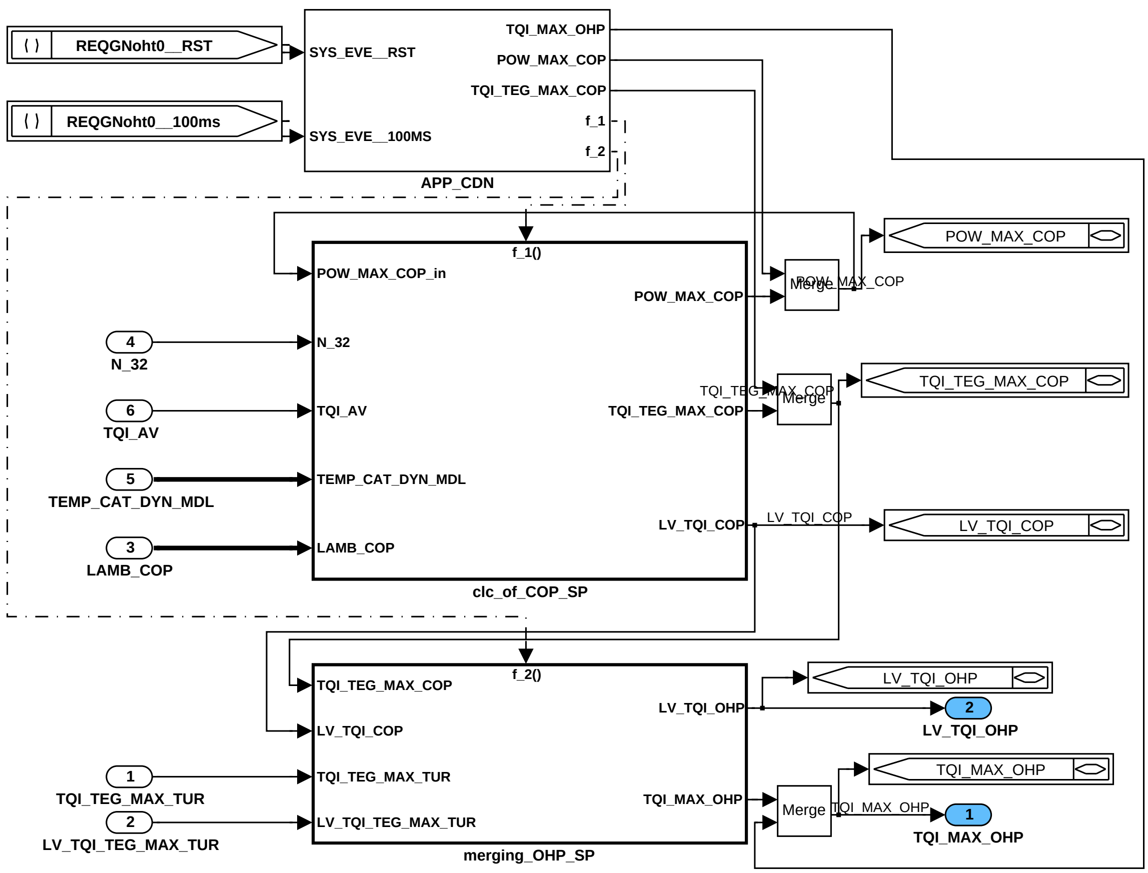
In case of critical catalyst temperatures the exhaust gas is cooled down by enrichment. If the lambda runs below the threshold C_LAMB_COP_POW_MIN the engine power has to be reduced as a second path to limit the maximum exhaust gas temperature. Once the exhaust gas temperature falls below C_TEG_POW_LIM the maximum engine power is ramped up again to the maximum value. These ramps have to calibrated slow to prevent a bad influence of driveability.

Application Condition



EXTC_REQGNoht0/APP_CDN/Chart

Function Description



SDA_SRS / SDA 3.1 22−Sep−2003

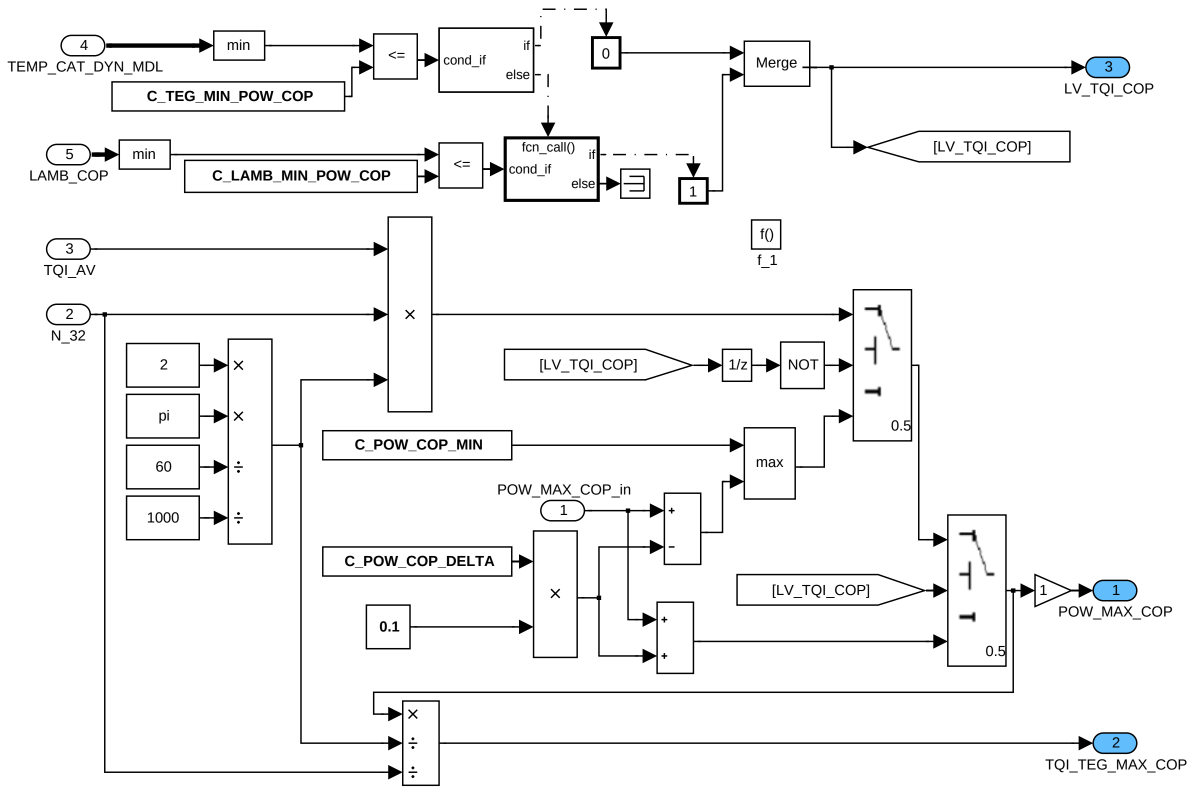
EXTC_REQGNoht0

D.20.1.1 Calculation of COP setpoints

In the case that LV_TQI_COP becomes 1, POW_MAX_COP has to be initialised with TQI_AV to get a smooth air reduction.

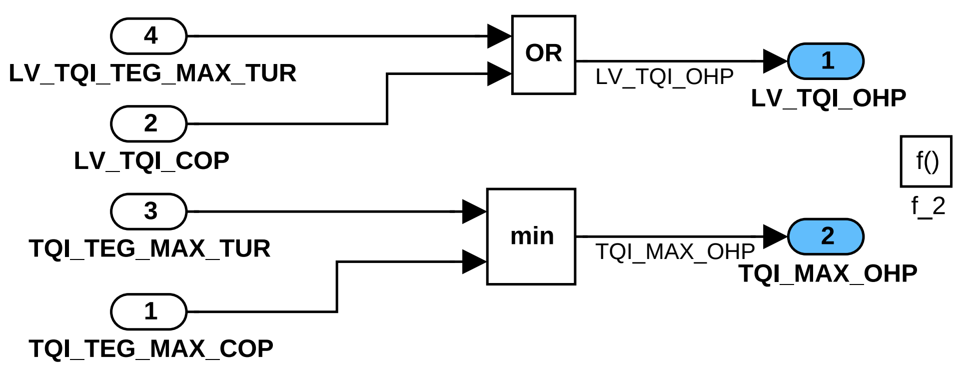
If LV_TQI_COP becomes 0 again, POW_MAX_COP is increased to its maximum value.

In order to realize the requested power limitation, it has to be converted in equivalent torque.



EXTC_REQGNoht0/clc_of_COP_SP

The maximum torque values due overheating prevention of turbine and catalyst are merged to LV_TQI_OHP and TQI_MAX_OHP.



EXTC_REQGNoht0/merging_OHP_SP

D.21 Engine Torque Limitation

D.21.1 Torque limitation coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_P_MAX_SLOW	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Maximum allowed indicated torque due to torque limitation (slow path)					
TQI_P_MAX_FAST	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Maximum allowed indicated torque due to torque limitation (fast path)					

Input data:

TQI_TCHA_PROT	TQI_MAX_OHP		

General information:

With the help of this function, the maximum allowed indicated torque can be limited (power limitation). Limitations for the turbocharger, exhaust system and overboost are coordinated here. The limitations are only introduced to the slow (air) path of the torque model.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: at reset TQI_P_MAX_SLOW = 1023.97 Nm
TQI_P_MAX_FAST = 1023.97 Nm

Update rate: 10 ms

Calculation order

1) chapter 1.2 with recurrence: 40 ms

2) chapter 1.3 with recurrance 40 ms

3) this chapter (1.1) with recurrence: 10 ms

Formula section:

TQI_P_MAX_FAST = 1023.97 Nm

TQI_P_MAX_SLOW = min (TQI_MAX_OHP, TQI_TCHA_PROT, TQI_POW_LIM, TQI_POW_IGA)

D.21.2 Permanent Power Limitation including Overboost

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_POW_MAX_REQ_CLU	V/O	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum Torque limititation					
TQI_POW_LIM	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque for power limitation used as torque limitation					
TQ_MAX_BAS_TCO	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum torque due to coolant temperature					
TQ_MAX_BAS_TOIL	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum torque due to oil temperature					
TQ_MAX_BAS	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum torque due to permanent power limitation					
TQI_MAX_BAS	V/O	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque due to permanent power limitation					
TQ_OVB_BAS	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum overboost torque range (offset to TQ_MAX_BAS)					
TQ_OFS_MAX_OVB	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum allowed torque offset for overboost					
TQ_OVB	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Actual overboost torque range (offset to TQ_MAX_BAS)					
LV_TQ_OVB_MAX	V	0...1H	0...1	1	[-]
=0 decreasing torque ramp active; =1 increasing torque ramp active					
CTR_OVB	V	0...FFFFH	0...65535	1	[-]
Overboost counter					
TQ_POW_LIM	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum torque incl. Overboost torque (overboost strategy 1)					
TQI_POW_LIM_1	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque for power limitation used as torque limitation (overboost strategy 1)					

TQI_POW_MAX_REQ_CLU_1	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque for power limitation used for the driver request calculation (overboost strategy 1)					
TQI_MAX_STAT	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum allowed indicated engine torque for stationary operation					
TQ_DIF_OVB	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Overboost torque difference					
TQ_OFS_OVB	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Torque offset for overboost					
CTR_TQ_OVB	V	0...FFFFH	0...65535	1	[-]
Counter to control the overboost activation and deactivation					
FAC_TQ_OVB_REQ	V	0...FFFFH	0...0.99998	0.0153e-3	[-]
Factor for the requested overboost torque range					
FAC_TQ_OVB_AVL	V	0...FFFFH	0...0.99998	0.0153e-3	[-]
Factor for the available overboost torque range					
LV_TQ_OVB_FDOUT	V	0...1H	0...1	1	[-]
Flag for overboost fade out					
TQI_POW_MAX_REQ_CLU_2	-	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque for power limitation used for the driver request calculation (overboost strategy 2)					
TQ_MAX_BAS_FTL	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum torque based on fuel tank level					

Input data:

TQ_AV	N_32	TQ_LOSS	TCO
VS	TOIL	LC_TCHA_CONF	TQ_LOSS_REQ_CLU
TQI_SP_SLOW	FAC_TQ_REQ	FTL	

General information:

This function provides the possibility to limit the maximum engine torque to TQ_MAX_BAS, depending on TCO and TOIL. An additional overboost torque IP_TQ_OVB_BAS (N_32) could be provided for a limited time. The transition between overboost torque und nominal torque TQ_MAX_BAS is done by ramp functions.

Application conditions:

Initialisation: TQI_POW_MAX_REQ_CLU = 1024 Nm
TQ_MAX_BAS = 1024Nm
CTR_OVB = 65535
CTR_TQ_OVB = 0
LV_TQ_OVB_FDOUT = 0
TQI_POW_LIM = 1024 Nm
FAC_TQ_OVB_AVL = 1 at reset

Recurrence: 40ms

Activation: at every engine operation state, if LC_TCHA_CONF = 1

Formula section:

TQ_MAX_BAS_TCO = IP_TQ_MAX_BAS_TCO (N_32, TCO)

TQ_MAX_BAS_TOIL = IP_TQ_MAX_BAS_TOIL (N_32, TOIL)

TQ_MAX_BAS_FTL = IP_TQ_MAX_BAS_FTL (FTL)

TQ_MAX_BAS = MIN (TQ_MAX_BAS_TCO, TQ_MAX_BAS_TOIL, TQ_MAX_BAS_FTL)
TQI_MAX_BAS = TQ_MAX_BAS - TQ_LOSS
TQ_OVB_BAS = IP_TQ_OVB_BAS (N_32) ;additional overboost ;torque range

For engine protection the overboost torque could be weighted by vehicle speed and oil temperature:

FAC_OVB_TOIL = IP_FAC_OVB_TOIL (TOIL)
FAC_OVB_VS = IP_FAC_OVB_VS (VS)
TQ_OFS_MAX_OVB = TQ_OVB_BAS * FAC_OVB_TOIL * FAC_OVB_VS

D.21.2.1 Overboost strategy 1

This strategy works with a counter to activate and deactivate the overboost. It only limits the torque setpoint without influencing the driver request calculation.

This section is *only calculated for C_OVB_MOD = 0 or C_OVB_MOD = 2*.

Calculation of de/increasing torque ramp:

If TQ_AV > TQ_MAX_BAS ;max. permanent torque exceeded?
then LV_TQ_OVB_MAX = 1 ;activate decreasing ramp
else LV_TQ_OVB_MAX = 0 ;activate increasing ramp
endif;

If LV_TQ_OVB_MAX = 1
then CTR_OVB(N) = MIN [C_CTR_OVB_INI; CTR_OVB(N-1) – C_CTR_OVB_DEC]
else CTR_OVB(N) = MIN [C_CTR_OVB_INI; CTR_OVB(N-1) + C_CTR_OVB_INC]
endif;

Calculation of the overboost torque range and maximum torque threshold:

The following formula could be explained by seperation of 3 different cases:
Counter value CTR_OVB > C_CTR_OVB_THD: full overboost torque available
Counter value 0 < CTR_OVB < C_CTR_OVB_THD: available overboost torque is ramped up/down
Counter value CTR_OVB = 0: no overboost torque available

TQ_OVB = TQ_OFS_MAX_OVB * MIN [CTR_OVB / C_CTR_OVB_THD; 1]

TQ_POW_LIM = TQ_MAX_BAS + TQ_OVB
TQI_POW_LIM_1 = TQ_POW_LIM - TQ_LOSS
(= indicated engine torque with current overboost)

TQI_POW_MAX_REQ_CLU_1 =

TQ_MAX_BAS + TQ_OFS_MAX_OVB – TQ_LOSS_REQ_CLU
(= indicated engine torque with maximum overboost)

Calibration hint: The time with maximum overboost torque TQ_OVB_BAS is determined by C_CTR_OVB_INI > C_CTR_OVB_THD.

D.21.2.2 Overboost strategy 2

To avoid a dead zone of the pedal when the overboost is over, this algorithm increases or decreases the reference torque for the driver request calculation depending on the state of the overboost function. This is done via TQI_POW_MAX_REQ_CLU_2, which limits the Maximum torque at clutch. The same value is also fed into the Minimum/Maximum torque request selection to ensure a limitation of the torque setpoint.

This section is *only calculated for C_OVB_MOD = 1 or C_OVB_MOD = 3*.

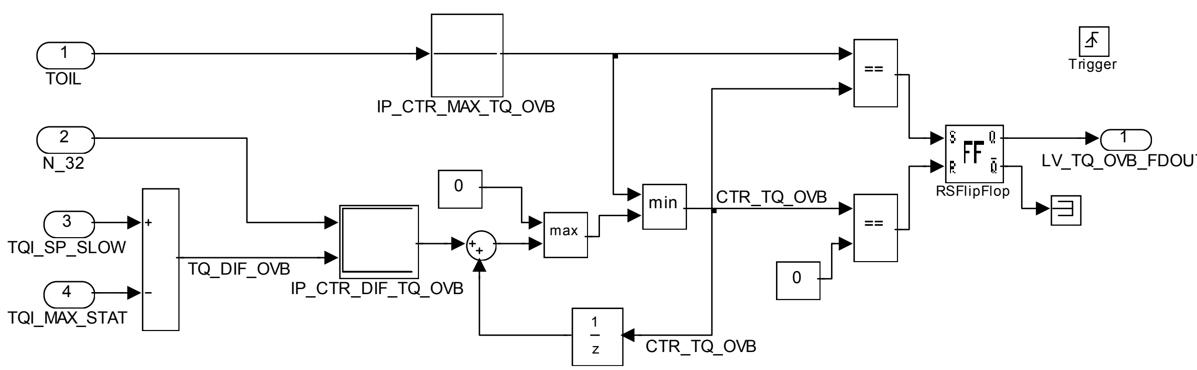
The maximum indicated torque for stationary engine operation is calculated as:

TQI_MAX_STAT = TQ_MAX_BAS - TQ_LOSS_REQ_CLU

Depending on the difference TQ_DIF_OVB between slow setpoint for the indicated engine torque and maximum torque for stationary operation, a counter is incremented and decremented. Increment and decrement are defined in IP_CTR_DIF_TQ_OVB, which must be calibrated to positive values for TQ_DIF_OVB > 0 and negative values for TQ_DIF_OVB < 0. The maximum value of the counter IP_CTR_MAX_TQ_OVB is a function of the oil temperature. When the counter reaches this maximum value, a fade out of the overboost is requested with LV_TQ_OVB_FDOUT. The flag is reset to zero when the counter reaches zero.

(Hint for SW implementation: Note data type conversions for the interpolation map IP_CTR_DIF_TQ_OVB because input and output of the map must be unsigned.)

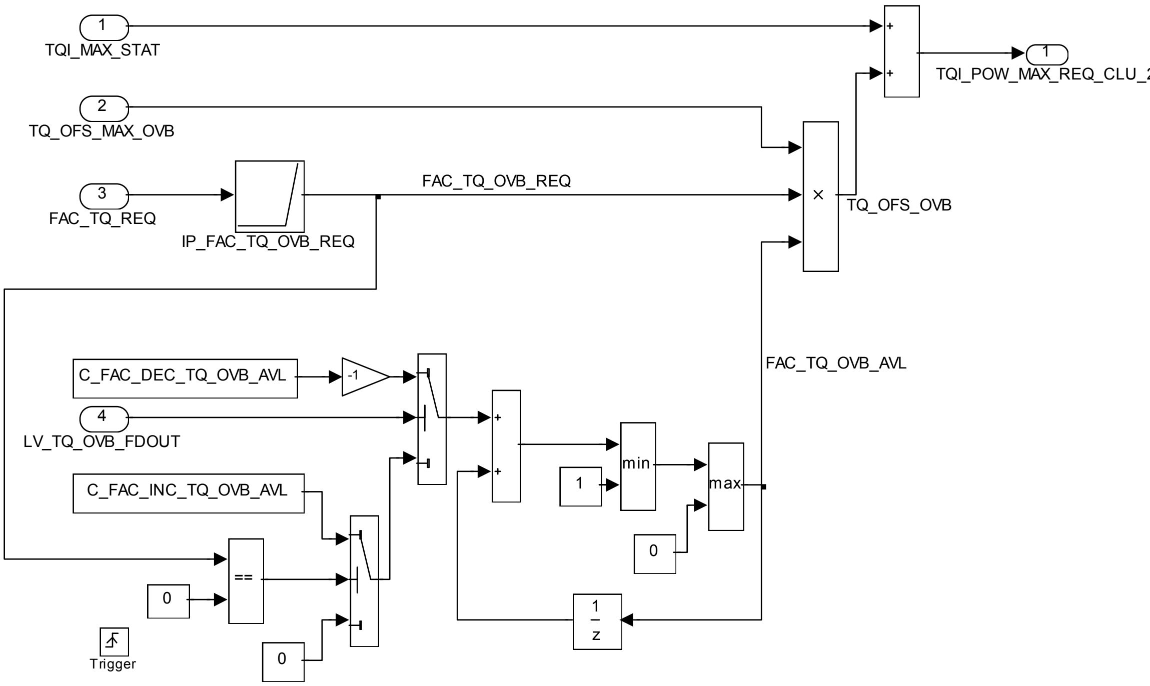
To calculate the overboost torque offset TQ_OFS_OVB, the maximum overboost offset TQ_OFS_MAX_OVB is multiplied by two factors.



The first factor FAC_TQ_OVB_REQ serves to activate the overboost depending on the driver request. This is realized with a map IP_FAC_TQ_OVB_REQ with the input FAC_TQ_REQ. The map shall be calibrated zero at small scaling factors to deactivate the overboost offset and shall increase to 1 at higher scaling factors to request the full overboost.

The second factor FAC_TQ_OVB_AVL defines which percentage of the overboost is currently available. If a regeneration is requested, this factor is ramped down to zero in order to fade out the overboost. If no regeneration is requested and if FAC_TQ_OVB_REQ is zero, FAC_TQ_OVB_AVL is ramped up to make the overboost available again. The check of FAC_TQ_OVB_REQ ensures that the overboost function does not increase the torque request without action of the driver.

The maximum allowed torque TQI_POW_MAX_REQ_CLU_2 is the maximum stationary torque TQI_MAX_STAT plus the overboost offset TQ_OFS_OVB.



D.21.2.3 Coordination of overboost output variables

The output TQI_POW_MAX_REQ_CLU is taken into account for the Maximum torque at clutch. It will therefore influence the reference of the driver request. In contrast to this, TQI_POW_LIM is taken into account in the Minimum/Maximum torque request selection and will just limit the torque request.

If C_OVB_MOD = 0

then *Use overboost strategy 1:*

TQI_POW_MAX_REQ_CLU = TQI_POW_MAX_REQ_CLU_1
TQI_POW_LIM = TQI_POW_LIM_1

elseif C_OVB_MOD = 1

Use overboost strategy 2:

TQI_POW_MAX_REQ_CLU = TQI_POW_MAX_REQ_CLU_2
TQI_POW_LIM = TQI_POW_MAX_REQ_CLU_2

elseif C_OVB_MOD = 2

Use overboost strategy 1, with no limitation of driver request:

TQI_POW_MAX_REQ_CLU = 1023.96875Nm
TQI_POW_LIM = TQI_POW_LIM_1

else *Use overboost strategy 2 with no limitation of driver request:*

(calibration of C_OVB_MOD = 3)

TQI_POW_MAX_REQ_CLU = 1023.96875 Nm
TQI_POW_LIM = TQI_POW_MAX_REQ_CLU_2

endif;

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_MAX_BAS_TCO	8*4	0...7FFFH	0...1023.96875	0.03125	[Nm]
LDPM_N_32_TQ_MAX_BAS	8	0...FFH	0...8160	32	[rpm]
LDP_TCO_TQ_MAX_BAS	4	0...FEH	-48...142.5	0.75	[°C]
Maximum torque due to coolant temperature					
IP_TQ_MAX_BAS_TOIL	8*4	0...7FFFH	0...1023.96875	0.03125	[Nm]
LDPM_N_32_TQ_MAX_BAS	8	0...FFH	0...8160	32	[rpm]
LDP_TOIL_TQ_MAX_BAS	4	0...C8H	-40...160	1	[°C]
Maximum torque due to oil temperature					
IP_TQ_OVB_BAS	8	0...7FFFH	0...1023.96875	0.03125	[Nm]
LDPM_N_32_TQ_MAX_BAS	8	0...FFH	0...8160	32	[rpm]
Overboost torque for short time application					
IP_FAC_OVB_TOIL	2	0...FFH	0...0.99609	3.9063e-3	[-]
LDP_TOIL_IP_FAC_OVB_TOIL	2	0...C8H	-40...160	1	[°C]
Overboost torque for short time application					
IP_FAC_OVB_VS	2	0...FFH	0...0.99609	3.9063e-3	[-]
LDP_VS_IP_FAC_OVB_VS	2	0...FFH	0...255	1	[km/h]
Overboost torque for short time application					
C_CTR_OVB_THD	1	0...FFFFH	0...65535	1	[-]
Threshold for overboost counter to ramp up/down overboost torque					
C_CTR_OVB_INI	1	0...FFFFH	0...65535	1	[-]
Init value for overboost counter					
C_CTR_OVB_INC	1	0...FFFFH	0...65535	1	[-]
Increasing value for overboost counter					
C_CTR_OVB_DEC	1	0...FFFFH	0...65535	1	[-]
Decreasing value for overboost counter					
IP_CTR_DIF_TQ_OVB	4*6	0...FFH	-128...127	1	[-]
LDP_N_32_IP_CTR_DIF_TQ_OVB	4	0...FFH	0...8160	32	[rpm]
LDP_TQ_DIF_IP_CTR_DIF_TQ_OVB	6	0...FFFFH	-1024... 1023.96875	0.03125	[Nm]
Increment or decrement of the overboost counter					
IP_CTR_MAX_TQ_OVB	4	0...FFFFH	0...65535	1	[-]
LDP_TOIL_IP_CTR_MAX_TQ_OVB	4	0...C8H	-40...160	1	[°C]
Maximum value of the overboost counter					
IP_FAC_TQ_OVB_REQ	4	0...FFFFH	0...0.99998	0.0153e-3	[-]
LDP_FAC_TQ_REQ_IP_FAC_TQ_OVB	4	0...FFFFH	0...1.99996	0.0305e-3	[-]
Factor for the requested overboost torque range					
C_FAC_INC_TQ_OVB_AVL	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Increment to ramp up the factor for the available overboost torque range					
C_FAC_DEC_TQ_OVB_AVL	1	0...FFFFH	0...0.99998	0.0153e-3	[-]
Decrement to ramp down the factor for the available overboost torque range					
C_OVB_MOD	1	0...3H	0...3	1	[-]
Switch to select overboost mode					
IP_TQ_MAX_BAS_FTL	4	0...7FFFH	0...1023.96875	0.03125	[Nm]
LDP_FTL_IP_TQ_MAX_BAS_FTL	4	0...FFH	0...100	0.39	[%]
Maximum torque based on fuel tank level					

D.21.3 Power Limitation due to IGA retard

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_POW_IGA	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum indicated torque due to IGA retard					
LV_POW_IGA	V	0...1H	0...1	1	[-]
power limitation due to ignition retard requested					
TQ_DELTA_POW_IGA	V	0...7FFFH	0...1023.96875	0.03125	[Nm]

torque reduction due to IGA retard					
IGA_DIF_POW_THD	V	0...80H	-48...0	0.375	[°CRK]
Activation threshold for torque reduction due to IGA retard					

Input data:

IGA_MV_ADJ_KNK	MAF_HB	N_32	TQI_AV
TQI_MAX_BAS	EFF_TOT_BAS_SLOW	TQI_REF	

Application conditions:

Initialisation: TQI_POW_IGA = 1024 Nm
TQ_DELTA_POW_IGA = 0 Nm at reset

Recurrence: 40ms

Activation: at every engine operation state, if LC_TCHA_CONF = 1

General information:

The IGA retards of due to knock events are compared to a calibrated minimum ignition angle (IP_IGA_DIF_COP_MIN). In case the minimum ignition angle is exceeded, the actual torque is reduced.

Formula section:

The following IGA condition triggers the power limitation, whereas the TQI condition usually controls the deactivation.

IGA_DIF_POW_THD = IP_IGA_DIF_COP_MIN (MAF_HB, N_32)
+ IP_IGA_ADD_COP_MIN_TIA_N_32 (TIA;N_32)
***Hint for SW**: IGA_DIF_POW_THD must be limited to -48 and 0 °CRK*

If IGA_MV_ADJ_KNK < IGA_DIF_POW_THD
then LV_POW_IGA = 1
else LV_POW_IGA = 0

The initialisation of TQI_POW_IGA_INI is made in a way that the MAF_SP course is continous and smooth:

If LV_POW_IGA = 0 -> 1
then TQI_POW_IGA_INI = TQI_REF * EFF_TOT_BAS_SLOW
TQ_DELTA_POW_IGA = 0
endif

Calculation of TQI_POW_IGA: The decrease and increase of the torque request is done by an I-Controller.

If LV_POW_IGA = 1

then $TQ\_DELTA\_POW\_IGA_{n}$ =
$TQ\_DELTA\_POW\_IGA_{n-1}$ +
[C_TQ_DELTA_POW_IGA_I * (IP_IGA_DIF_COP_MIN(MAF_HB,N_32) - IGA_MV_ADJ_KNK)];

Limitation:
0 <= TQ_DELTA_POW_IGA <= C_TQ_DELTA_POW_IGA_MAX

Maximum indicated torque:
TQI_POW_IGA = TQI_POW_IGA_INI – TQ_DELTA_POW_IGA

else Increase torque to maximum value (deactivation):
$TQI\_POW\_IGA_{n} = TQI\_POW\_IGA_{n-1} + C\_TQ\_DELTA\_POW\_IGA\_DEAC$

endif

Final limitation of TQI_POW_IGA to an absolute level below TQ_MAX_BAS

If TQI_POW_IGA < TQI_MAX_BAS – IP_TQ_DELTA_POW_IGA_MAX_ABS (TAM)
then TQI_POW_IGA = TQI_MAX_BAS – IP_TQ_DELTA_POW_IGA_MAX_ABS (TAM)
endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_DELTA_POW_IGA_DEAC	1	0...7FFFH	0...1023.96875	0.03125	[Nm]
Torque increase at deactivation					
C_TQ_DELTA_POW_IGA_MAX	1	0...7FFFH	0...1023.96875	0.03125	[Nm]
Maximum torque reduction due to IGA retard					
C_TQ_DELTA_POW_IGA_I	1	0...7FFFH	0...1023.96875	0.03125	[Nm/°CRK]
Torque slope gain due to IGA retard					
IP_TQ_DELTA_POW_IGA_MAX_ABS	4	0...7FFFH	0...1023.96875	0.03125	[Nm]
LDP_TAM_IP_TQ_DELTA_POW	4	0...FEH	-48...142.5	0.75	[°C]
Absolute maximum torque reduction due to IGA retard					
IP_IGA_DIF_COP_MIN	4*8	0...80H	-48...0	0.375	[°CRK]
LDP_MAF_HB_IP_IGA_DIF_COP_MIN	4	0...FFH	0...1389	5.4470588	[mg/stk]
LDP_N_32_IP_IGA_DIF_COP_MIN	8	0...FFH	0...8160	32	[rpm]
Permissible IGA retard before catalyst overheating prevention					
IP_IGA_ADD_COP_MIN_TIA_N_32	4*4	0...80H	-48...0	0.375	[°CRK]
LDP_TIA_IP_IGA_ADD_COP_MIN	4	0...FEH	-48...142.5	0.75	[°C]
LDP_N_32_IP_IGA_ADD_COP_MIN	4	0...FFH	0...8160	32	[rpm]
Permissible IGA retard before catalyst overheating prevention					

D.22 Torque based MAF setpoint for homogeneous mode

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_SP_TQI	O/V	0...FFFFH	0...1389	0.0212	mg / stk
Torque based MAF setpoint for homogeneous mode					
TQI_SP_MAF	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque for determination of air mass flow setpoint in homogeneous mode					

Input data:

EFF_TOT_BAS_SLOW	TQI_MIN_PU	LV_HOM_RUN	N_32
TQI_SP_SLOW	TQI_AD_OFS	FAC_TQI_AD_SLOP	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_MAF_SP	16x12	0...FFFFH	0...1389	0.0212	mg/stroke
LDPM_N_32_1_TQDR	16	0...FFH	0...8160	32	1/min
LDP_TQI_SP_MAF_IP_MAF_SP	12	0...7FFFH	0...1023.97	0.03125	Nm
Setpoint mass air flow depending on slow torque setpoint					

D.22.1 General Information

The output parameter MAF_SP_TQI represents the mass air flow setpoint for the homoge-neous combustion mode. It is utilized as an input parameter for the reverse Intake Manifold Model, where it is translated into a setpoint for the opening angle of the throttle blade. MAF_SP_TQI is obtained from the map IP_MAF_SP (inverse map of IP_TQI_REF) as a function of engine speed N_32 and the corrected torque setpoint TQI_SP_MAF.

The torque setpoint TQI_SP_MAF is principially a function of TQI_SP_SLOW and EFF_TOT_BAS_SLOW. Since the engine requires a minimum amount of air/fuel-mixture for stable combustion, TQI_SP_MAF is limited to TQI_MIN_PU at light load conditions. In the case of fuel cut-off (LV_PUC = 1), TQI_MIN_PU is utilized to control the absolute pressure in the intake manifold.

Function Description

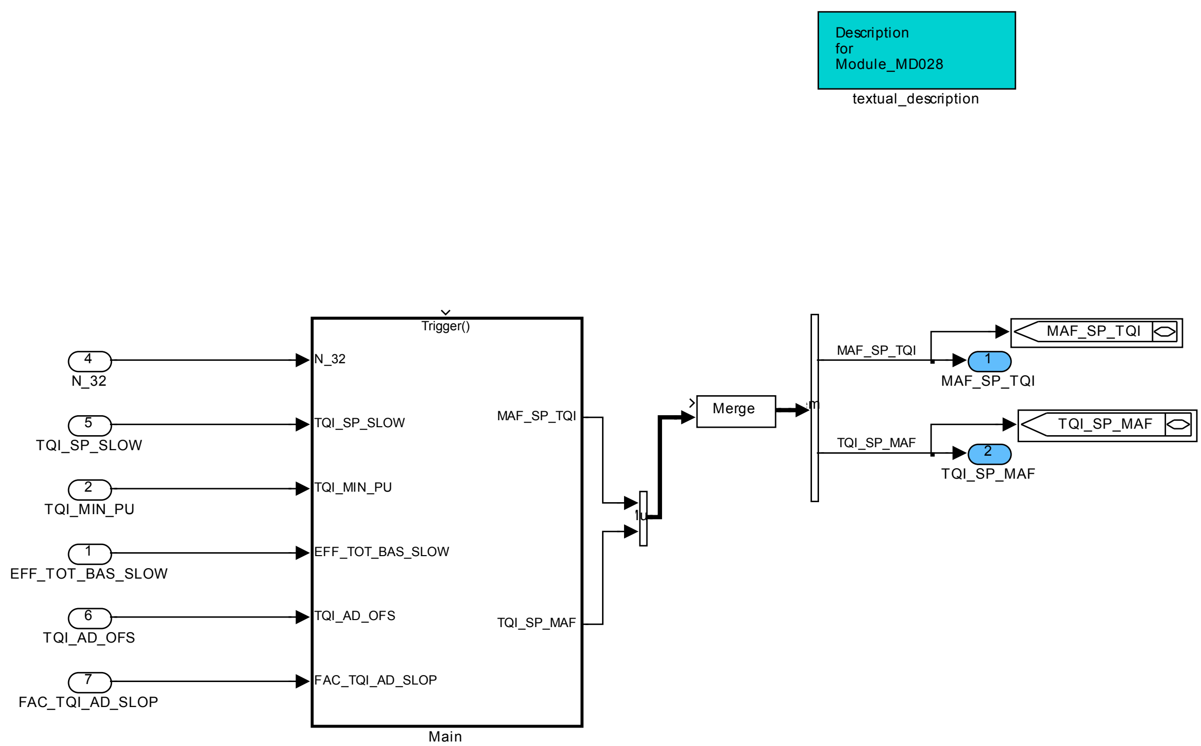


Figure 31 MD028

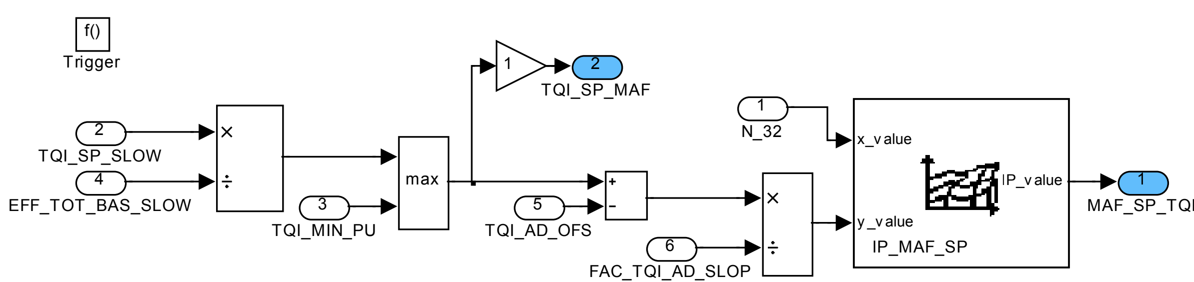


Figure 32 MD028/ Main

D.23 Torque reserve in part load

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_PL	O/V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
torqe reserve in PL					

Input data:

N	FAC_TQ_REQ_DRIV	STATE_ENG	LV_AT
VS_FIL			

FUNCTION DESCRIPTION:

Depending on engine speed and torque request of driver a torque reserve in part load can be calibrated, this torque reserve could be used to delivery enegy to the turbine to reduce turbo lag.

Application conditions:

Initialisation: TQ_ADD_PL = 0

Recurrence: 10 ms

Activation: at every engine state if LC_TCHA_CONF = 1

Deactivation: LC_TCHA_CONF = 0

Formula section:

If STATE_ENG = PL
And ((C_TQ_ADD_PL_ENA = 1 **and** LV_AT = 0)
or C_TQ_ADD_PL_ENA = 2)
Then TQ_ADD_PL = IP_TQ_ADD_PL_FAC_TQ_REQ_DRIV_N * IP_FAC_TQ_ADD_PL
Else TQ_ADD_PL = 0
Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ADD_PL_FAC_TQ_REQ_DRIV_N	12*12	0...FFFFH	-1024... 1023.96875	0.03125	[Nm]
LDP_FAC_TQ_REQ_DRI_IP_TQ_ADD_PL	12	0...FFFFH	0...1.99996	0.0305e-3	[-]
LDP_N_IP_TQ_ADD_PL	12	0...1FE0H	0...8160	1	[rpm]
torque reserve for partload					
IP_FAC_TQ_ADD_PL	4	0...FFH	0...1.99218	0.0078125	[-]
LDP_VS_FIL_IP_TQ_ADD_PL	4	0...FFFFH	0...511.99218	0.0078125	[km/h]
Weighting factor for torque reserve for partload depending on vehicle speed					
C_TQ_ADD_PL_ENA	1	0...2H	0...2	1	[-]
switch for torque reserve in PL					

D.24 Basic Fuel Cut Off Coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_SCC_BAS_REQ	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Coordinated basic cylinder cut-off efficiency					
LV_EFF_BAS_REQ	O/V	0...1H	0...1	1	-
Bit for request of calculated EFF_SCC_BAS due to fix cylinder cut-off pattern					

Input data:

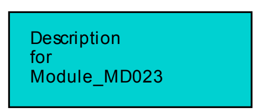
LV_ES			

D.24.1 General Information

For certain modes of fuel injection deactivation it might be necessary to have a correspond-ing basic efficiency for fuel cut-off (e.g. cylinder bank deactivation). There can also be more than one mode with a certain efficiency, so this needs to be coordinated.

This module can be adapted to project specific needs for calculation of EFF_SCC_BAS. With the logical variable LV_EFF_BAS_REQ the value EFF_SCC_BAS is obtained from EFF_SCC_BAS_REQ.

Function Description



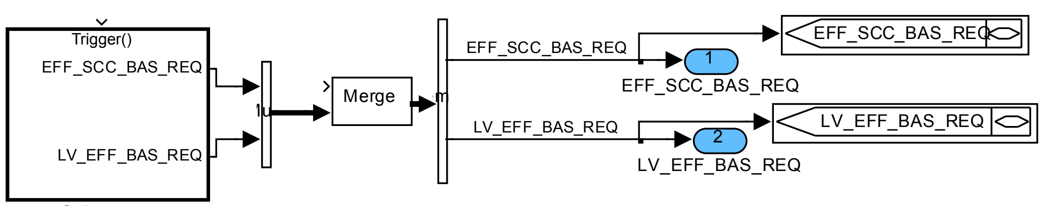


Figure 33 MD023

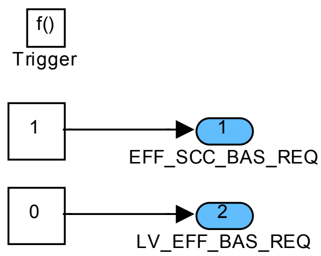


Figure 34 MD023/ Subsystem

D.25 Torque based Pattern Calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_IGA_MIN_SCC	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
minimum reference efficiency for single cylinder cut-off					
EFF_SCC_SP	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Setpoint efficiency fuel cut-off pattern					
EFF_SCC_SP_HYS	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Setpoint efficiency fuel cut-off pattern with hysteresis					
NR_PAT_SCC	O/V	0...FFH	0...255	1	-
Selected index of fuel cut-off pattern					
TQI_REF_IGA_MIN_SCC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
indicated reference torque for single cylinder cut-off					
FAC_IGA_MIN_TEG_IGA_BAS	V	0...FFFFH	0...1.9999	3.0517e-5	-
scaling factor for efficiency calculation with EFF_IGA_MIN_TEG and EFF_IGA_BAS_COR					
LV_AUTH_TQ_PAT	V	0...1H	0...1	1	-
logical variable for authorizing fuel cut-off pattern					

Input data:

EFF_IGA_BAS_COR	EFF_IGA_MIN_TEG	TQI_REF	EFF_EGR_HOM
EFF_LAMB_BAS_COR	EFF_SCC_BAS	LV_FCUT_FAST	LV_FCUT_FAST_TOT
LV_TQ_LIM_INTV	LV_TQ_SCC_INH	LV_TQ_SCC_REQ	LV_TQ_SCC_REQ_TOT
TQI_REQ_FAST	LV_S_ACT		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_INH_FCUT_FAST	1	0...1H	0...1	1	-
inhibition flag for fuel cut-off due to fast fuel cut-off					
LC_TQ_SCC_INH_MAN	1	0...1H	0...1	1	-
Inhibition flag for fuel cut-off due to torque management					
LC_TQ_SCC_ENA_MAN	1	0...1H	0...1	1	-
flag for enabling SCC by application system					
C_EFF_SCC_SP_MAN	1	0...FFFFH	0...1.9999	3.0517e-5	-
Setpoint for efficiency of fuel cut-off set by application intervention					
LC_EFF_SCC_SP_MAN	1	0...1H	0...1	1	-
flag for setting EFF_SCC_SP by application system					
C_EFF_SCC_SP_HYS	1	0...FFFFH	0...1.999	3.05e-5	-
hysteresis applied on EFF_SCC_SP for NR_PAT_SCC calculation					
C_HYS_EFF_IGA_MIN_TEG	1	0...FFFFH	0...1.9999	3.0517e-5	-
hysteresis in falling direction for IGA_MIN_TEG correction					
C_MIN_DEC_EFF_IGA_MIN_TEG_1	1	0...FFFFH	0...1.9999	3.0517e-5	-
minimum decrement for first decrease phase					
C_MIN_DEC_EFF_IGA_MIN_TEG_2	1	0...FFFFH	0...1.9999	3.0517e-5	-
minimum decrement for second decrease phase					
LC_ENA_FCUT_FAST_TOT	1	0...1H	0...1	1	-
Logical variable for setting scaling factor due to fast fuel cut-off					
LC_ENA_TQ_SCC_REQ_TOT	1	0...1H	0...1	1	-
Logical variable for setting scaling factor due to fast fuel cut-off request from dynamic torque management					
C_FAC_IGA_MIN_TEG_IGA_BAS	1	0...8000H	0...1	3.051e-5	-
Manually adjusted scaling factor for efficiency calculation with EFF_IGA_MIN_TEG and EFF_IGA_BAS_COR					
ID_NR_PAT_SCC	13	0...FFH	0...255	1	-
LDP_EFF_SCC_SP_ID_NR_PAT_SCC	13	0...FFFFH	0...1.9999	3.0517e-5	-
SCC pattern depending on SCC efficiency					

D.25.1 General Information

The objective of this module is to generate a torque intervention by fuel cut-off pattern.

This torque intervention could be caused by several conditions which are summarized in the label LV_AUTH_TQ_PAT (see chapter 1.1.1 "Authorization for single cylinder cut-off"). The label LV_FCUT_FAST can be used for project specific fast fuel cut-off conditions.

Based on the TQI_REQ_FAST and the TQI_REF corrected with the basic a/f ratio efficiency and the ignition angle efficiency an efficiency setpoint for the single cylinder cut-off EFF_SCC_SP is calculated, which leads to the pattern index NR_PAT_SCC (see chapter "Setpoint efficiency fuel cut-off").

In general, a torque reduction could be realized by spark retarding or single cylinder cut-off. With the scaling factor FAC_IGA_MIN_TEG_IGA_BAS a scaling between earliest cylinder cut-off (no spark retarding) and latest cylinder cut-off (full spark retarding up to IGA_MIN) can be realized (see chapter"Determination of scaling factor").

Function Description

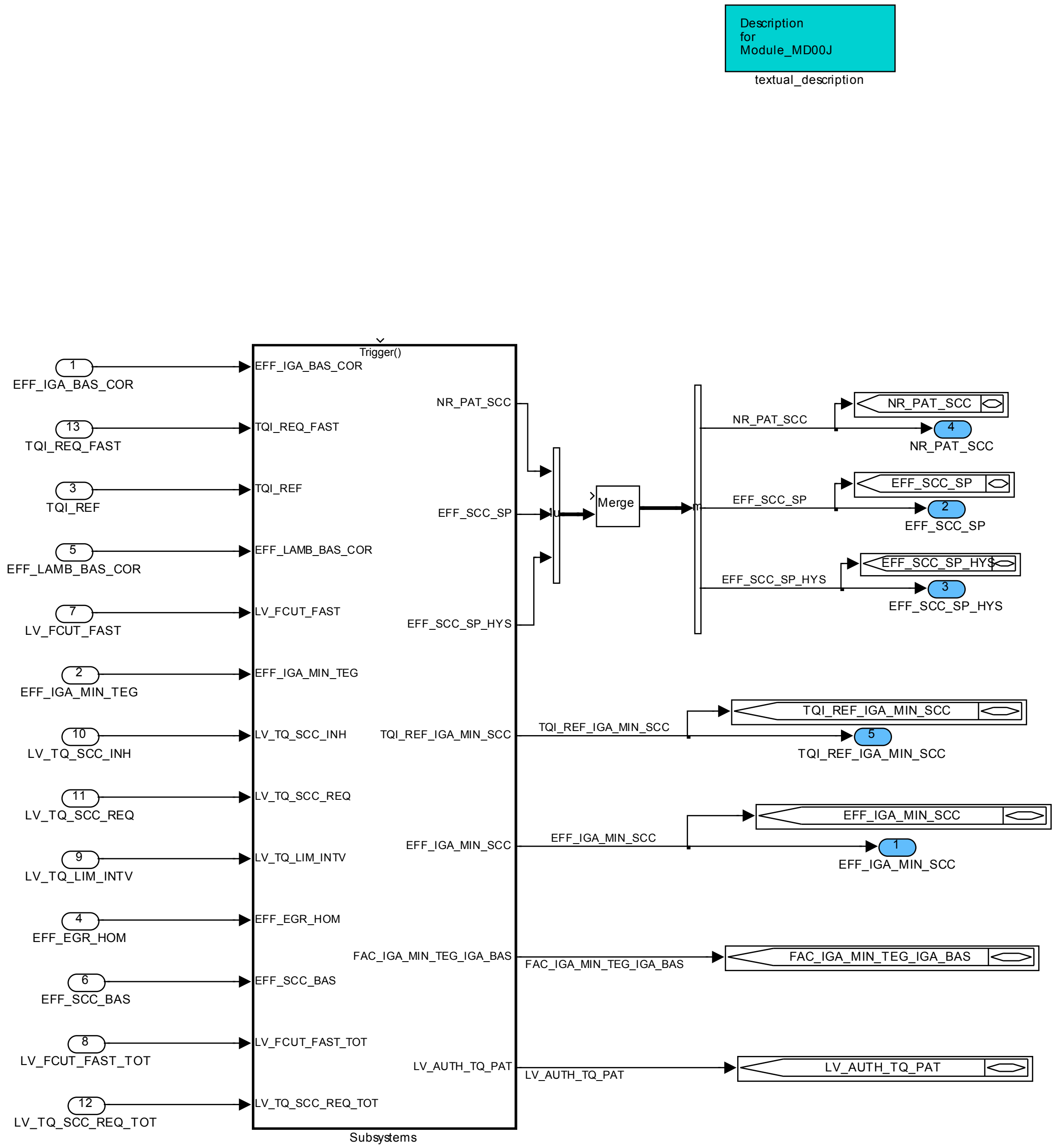


Figure 35 MD00J

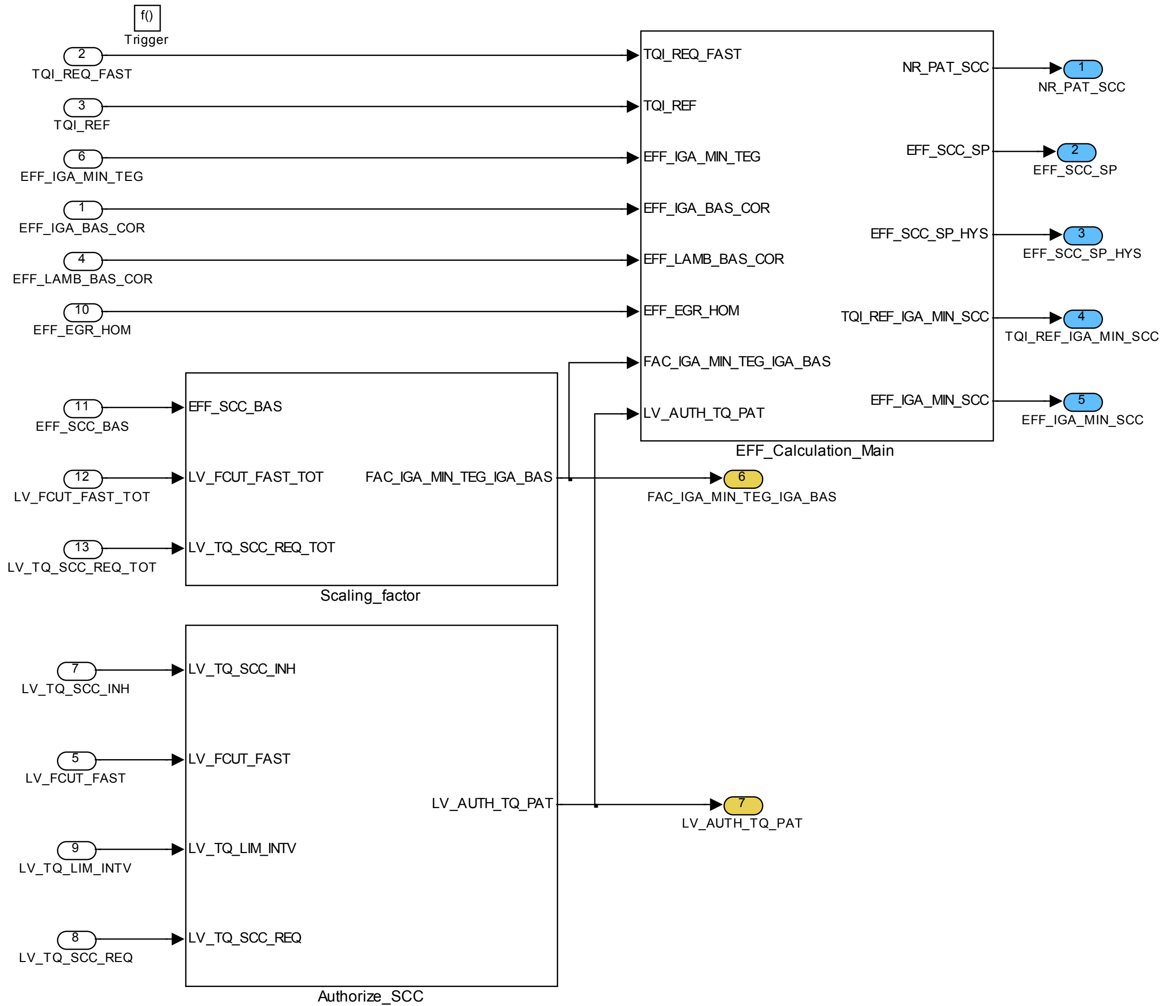


Figure 36 MD00J/ Subsystems

The authorization flag is set depending on:

- a manual enabling => LV_TQ_SCC_ENA_MAN
- a request from torque management => LV_TQ_SCC_REQ and LV_TQ_SCC_INH
- a request from safety concept => LV_TQ_LIM_INTV (see module "Torque request for safety")
- other project specific requests for fast fuel cut-off => LV_FCUT_FAST without control of dynamic torque management

Due to logical calibration constants nearly all of these request can be inhibited, except of the request from safety concept LV_TQ_LIM_INTV.

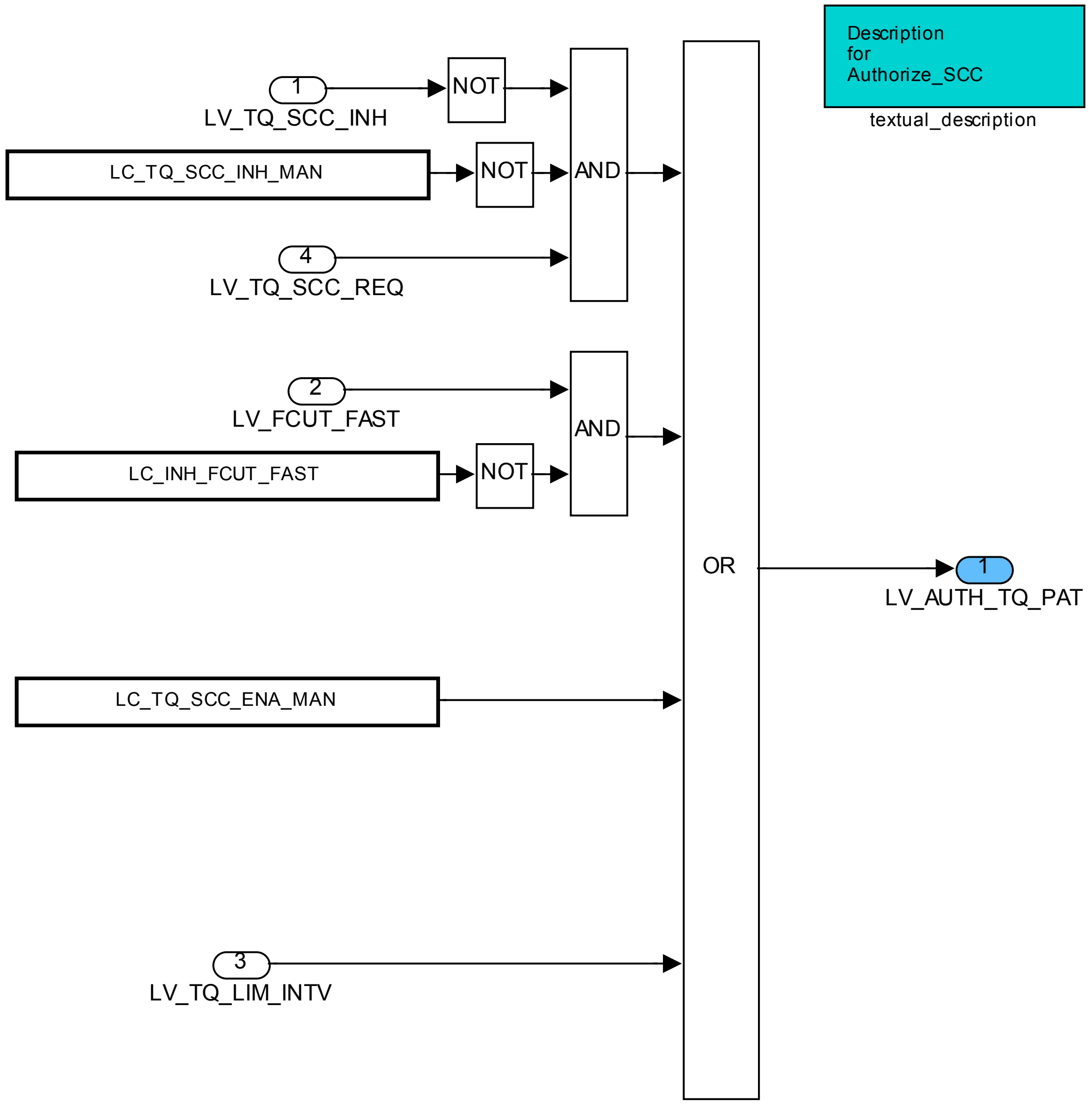
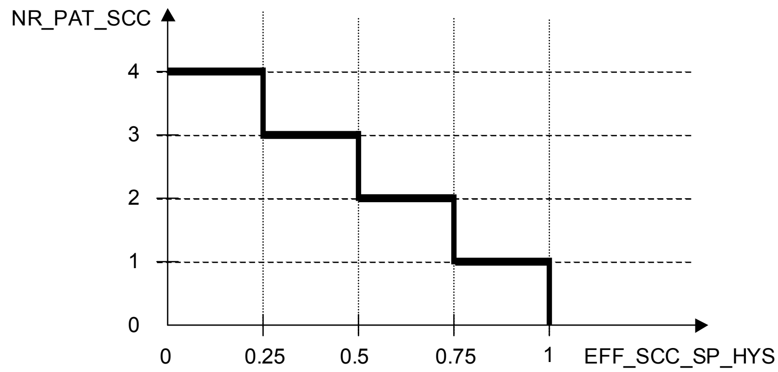


Figure 37 MD00J/ Subsystems/ Authorize_SCC

general specification

Depending on the authorization flag LV_AUTH_TQ_PAT the efficiency EFF_SCC_SP is set to the calculated value EFF_SCC_SP_1 or is set equal to "1". Additionally, two application labels allow to manipulate the efficiency setpoint (LC_EFF_SCC_SP_MAN and C_EFF_SCC_SP_MAN). Depending on the efficiency setpoint a pattern is calculated. To avoid uncomfortable jumps of pattern, if the setpoint EFF_SCC_SP jitters, the calculation of the pattern index NR_PAT_SCC is done by applying a hysteresis C_EFF_SCC_SP_HYS on EFF_SCC_SP.



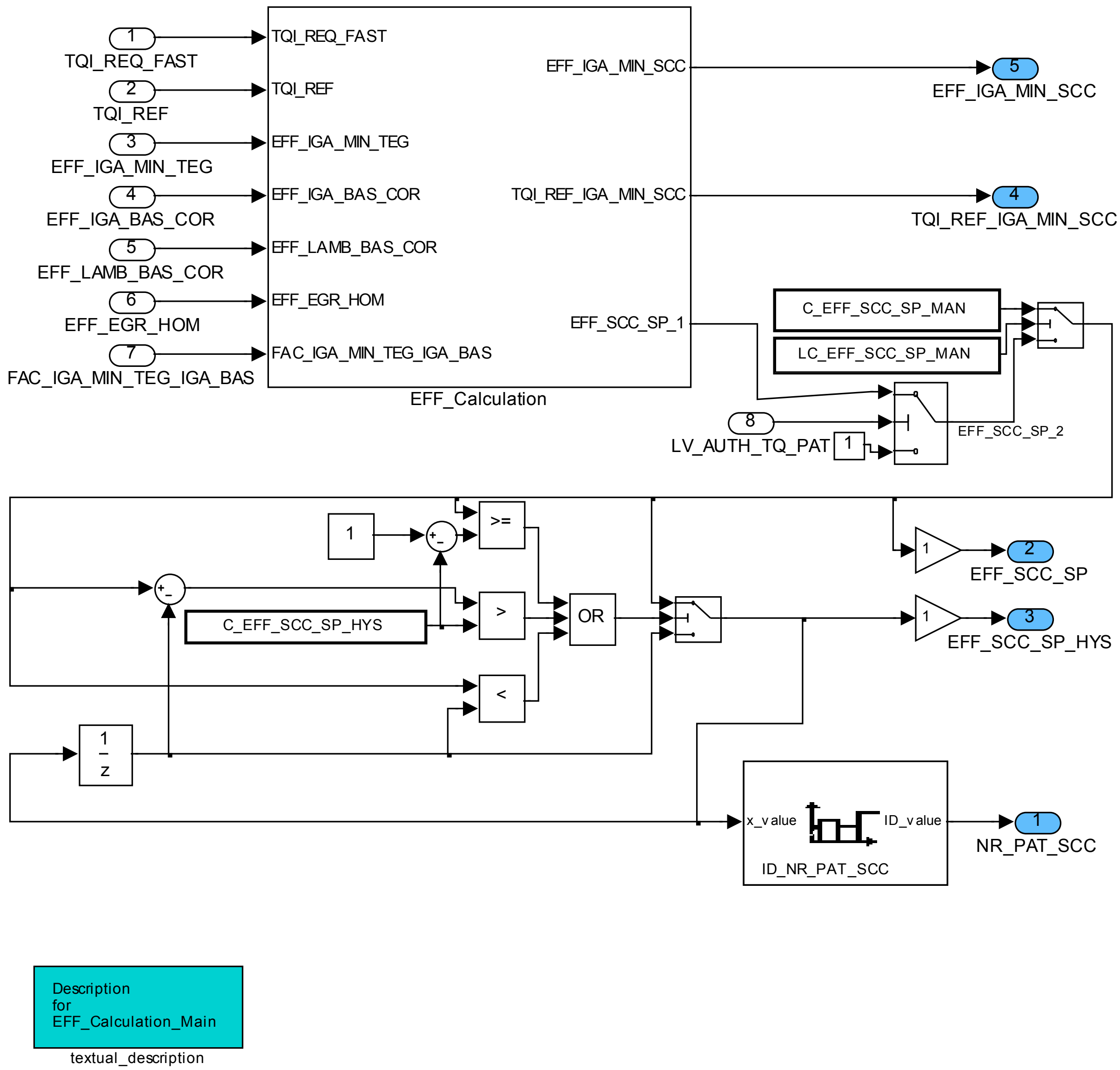


Figure 38 MD00J/ Subsystems/ EFF_Calculation_Main

general specification

The efficiency for single cylinder cut-off depends on the ratio between TQI_REQ_FAST and TQI_REF corrected with the basic efficiencies for a/f ratio and ignition angle. Generally, a scaling between EFF_IGA_MIN_TEG and EFF_IGA_BAS_COR is possible. The philosophy of this function is to give the customer the possibility to work with minimum ignition angle before fuel cut-off or to work with the basic ignition angle.

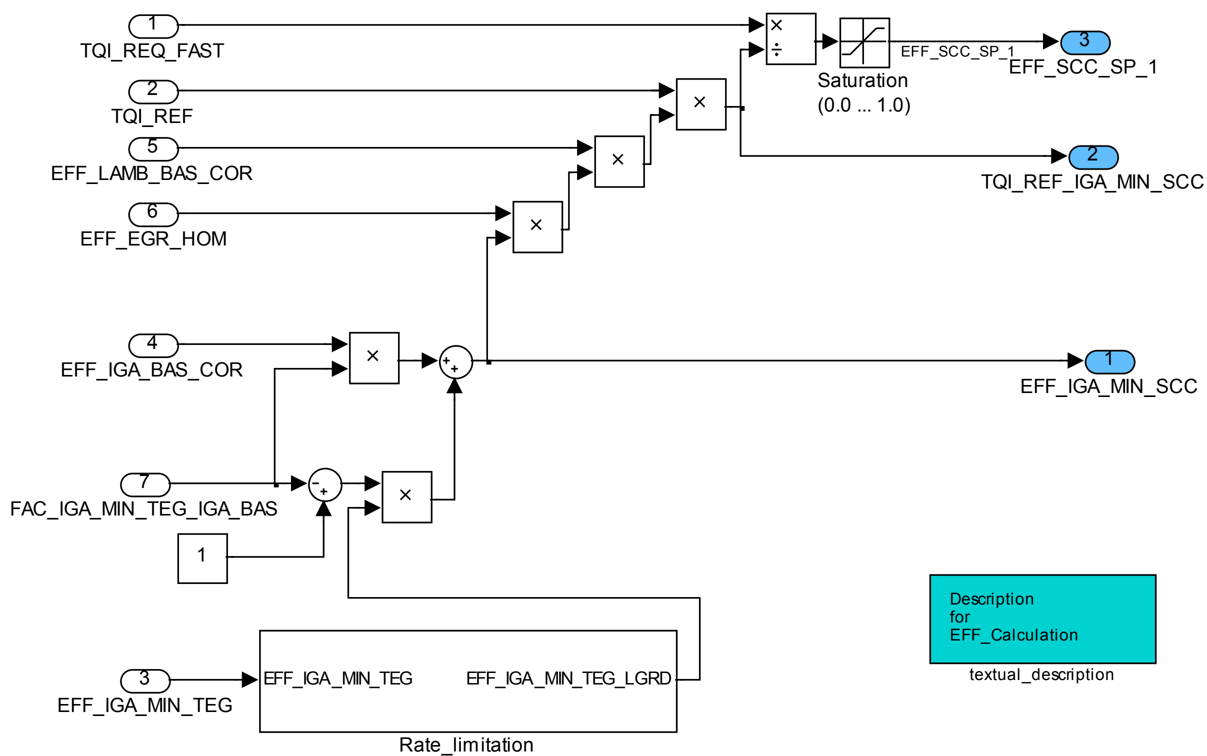
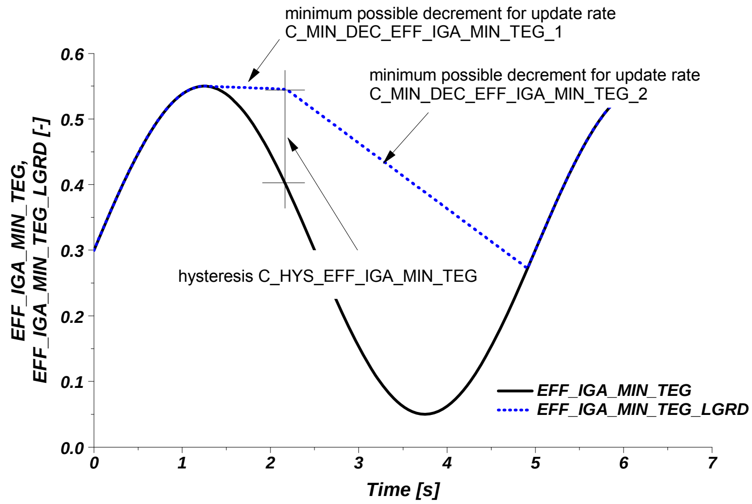


Figure 39 MD00J/ Subsystems/ EFF_Calculation_Main/ EFF_Calculation

Changes of the efficiency EFF_IGA_MIN_TEG has to be limited in falling direction to avoid high frequent jumps between pattern and spark retarding. Therefore a calibratable rate limit function is added. The principle application parameters for these rate limitations are C_MIN_DEC_EFF_IG_MIN_TEG_1, C_MIN_DEC_EFF_IG_MIN_TEG_2 and C_LGRD_EFF_IGA_MIN_TEG. The background is illustrated in figure 2.



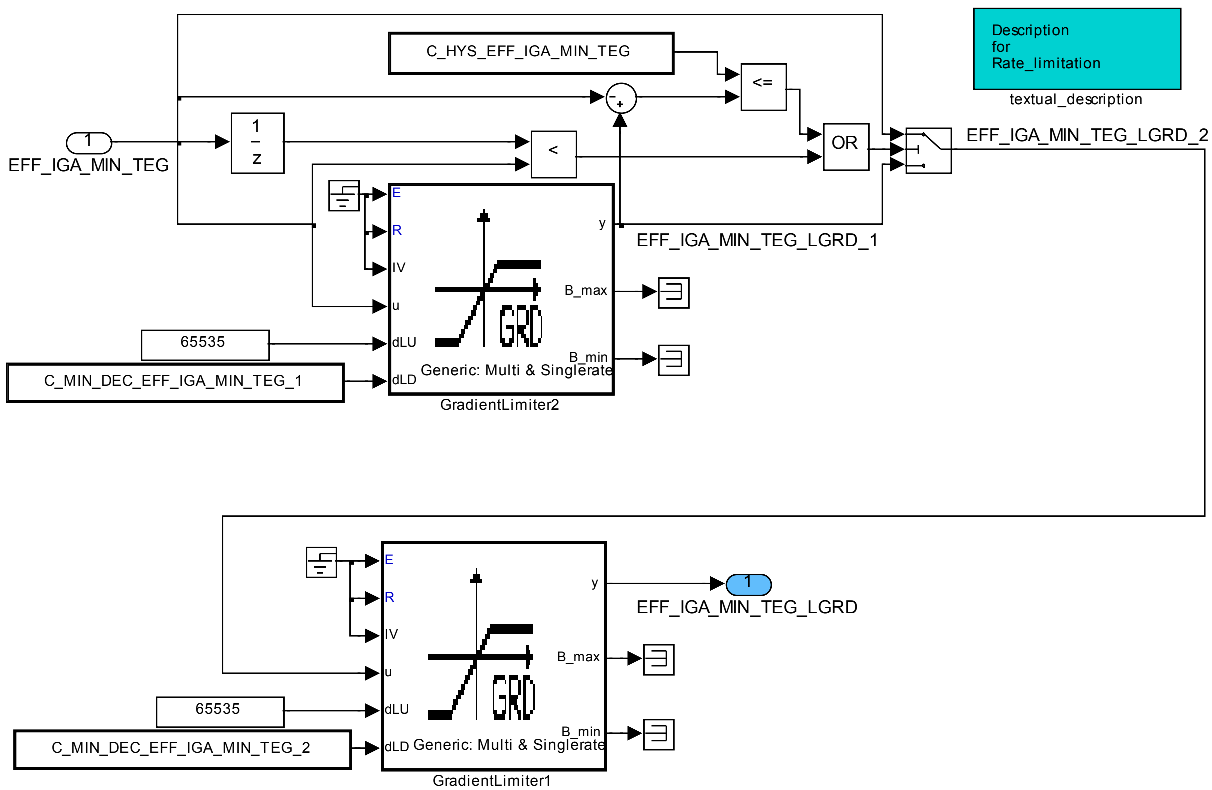


Figure 40 MD00J/ Subsystems/ EFF_Calculation_Main/ EFF_Calculation/ Rate_limitation

The factor FAC_IGA_MIN_TEG_IGA_BAS is used to scale between the efficiencies EFF_IGA_MIN_TEG (corrected basic minimum ignition angle including spark retard limitation due to exhaust gas temperature) and EFF_IGA_BAS_COR (corrected basic minimum ignition angle without spark retard limitation due to exhaust gas temperature). For a torque intervention due to gear shift, traction control, engine speed limitation and fast fuel cut-off the possibility of deactivating this scaling factor must be realized. So the scaling factor can be set equal to "1", if one of that interventions is active and if this method will be allowed (LC_ENA_FCUT_TOT_... = 1). It is also set equal to "1", if the basic efficiency for fuel cut-off is smaller than "1".

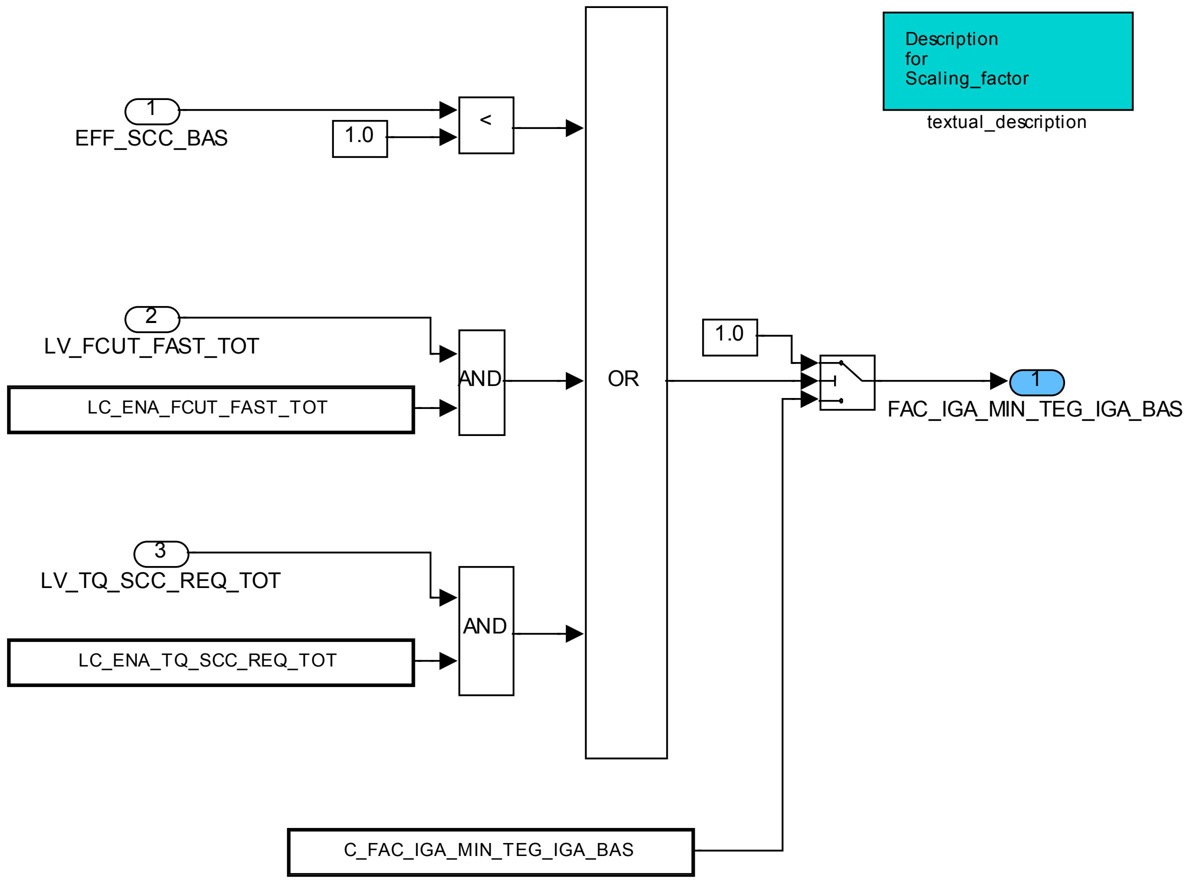


Figure 41 MD00J/ Subsystems/ Scaling_factor

D.26 Initialization Module

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_EGR_HOM	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
EGR efficiency for homogeneous mode					
LAMB_BAS_COR_MAX	O/V	0...7FFFH	0...31.999	0.0009765 6	-
Maximum value for LAMB_BAS_COR					
LAMB_SP_TQI	O/V	0...7FFFH	0...31.999	0.0009765 6	-
Lambda setpoint torque based					
LV_TQI_BOL_SET_S	O/V	0...1H	0...1	1	-
Logical variable is set if TQI_SP_S falls below the minimum allowable torque at stratified					
LV_TQ_LAMB_ACT	O/V	0...1H	0...1	1	-
Logical variable torque based lambda output is available					
TQI_AV_S	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Actual indicated engine torque for stratified					
TQI_MIN_PU_S	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum indicated engine torque at trailing throttle for stratified					

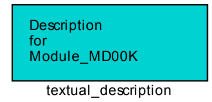
Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_LAMB_BAS_COR_MAX	1	0...7FFFH	0...31.999	0.0009765 6	-
Maximum value for LAMB_BAS_COR					

D.26.1 General Information

In this module, all output interfaces of the Aggregate TQDR are initialized that are not covered by a functionality.

Function Description



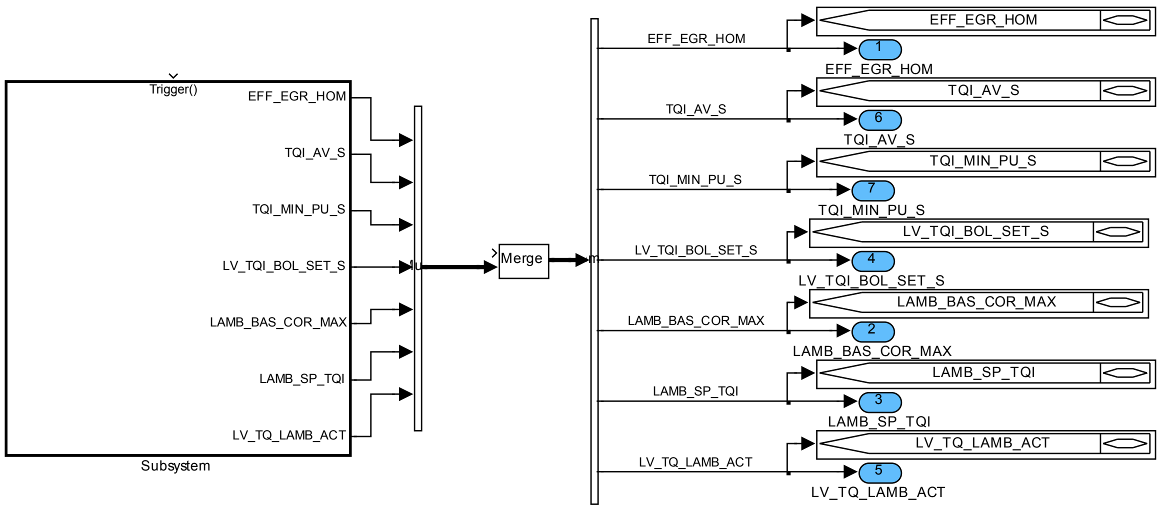


Figure 42 MD00K

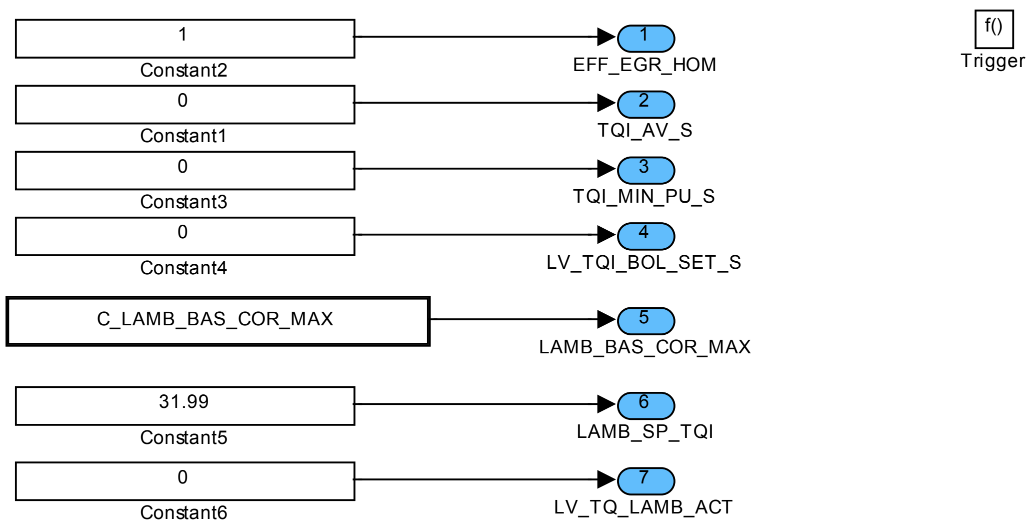


Figure 43 MD00K/ Subsystem

D.27 Coordination of Scaling Factors for Requested Torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_CRU_SUB	V	0..FFFFH	0..1.999969	3.052E-5	-
Result of the maximum selection between driver demand and cruise control demand					
FAC_TQ_REQ	O/V	0..FFFFH	0..1.999969	3.052E-5	-
Torque factor from pedal value interpretation, cruise control or vehicle speed limiter					
LV_INTV_CRU	O/V	0..1H	0..1	1	-
Boolean signals intervention of cruise control (CRU demand is equal or greater than driver demand)					
LV_INTV_VSL	O/V	0..1H	0..1	1	-
Boolean signals intervention of vehicle speed limiter (VSL demand is less than driver demand)					
LV_TQ_CRU_ACT	O/V	0..1H	0..1	1	-
Boolean signals intervention of cruise control (CRU demand greater than driver demand)					

Input data:

FAC_TQ_REQ_CRU	FAC_TQ_REQ_DRIV	FAC_TQ_REQ_VSL	LV_ACT_VSL
LV_ACT_CRU			

FUNCTION DESCRIPTION:

General information:

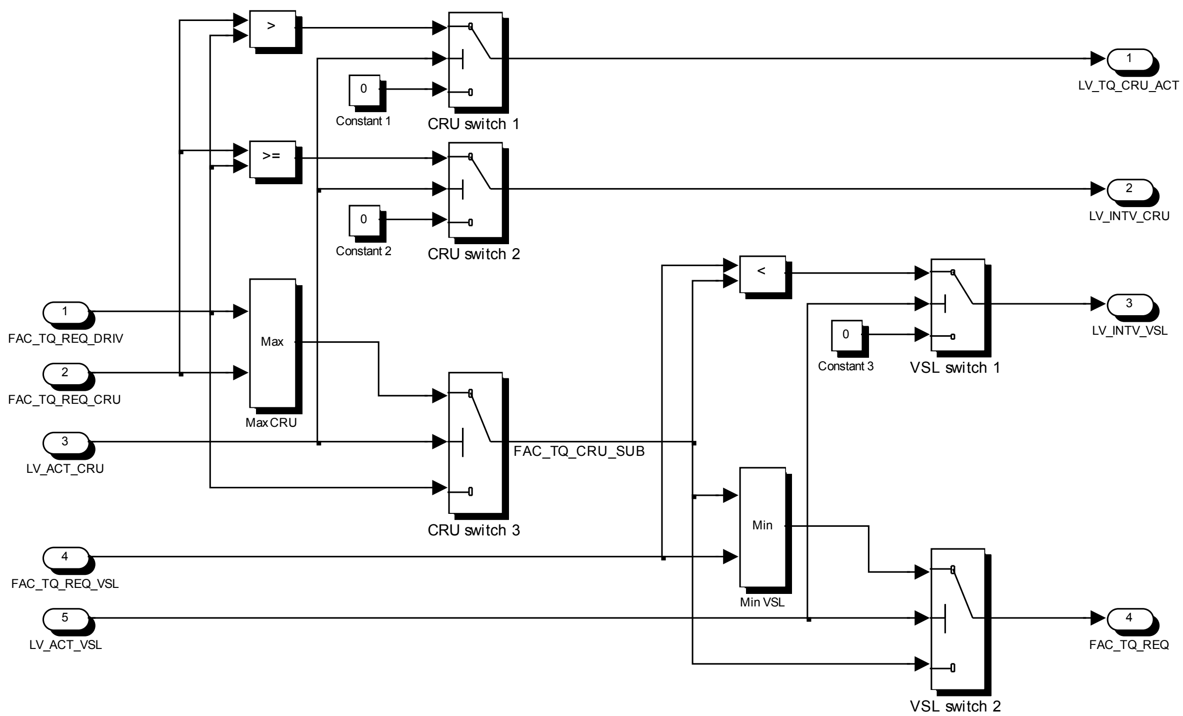
Not only the driver can provide a scaling factor but also the cruise control or the vehicle speed limiter. A coordination is necessary as all three scaling factors can request a torque demand simultaneously.

As the driver has a higher priority than the cruise control, the scaling factors FAC_TQ_REQ_DRIV (driver) and FAC_TQ_REQ_CRU (cruise control) are coordinated by a maximum selection with the output FAC_TQ_CRU_SUB. The intervention of the cruise control is indicated by the two booleans LV_TQ_CRU_ACT and LV_INTV_CRU. The coordination between the cruise control and the driver is only done if the cruise control is active (LV_ACT_CRU = 1).

As the vehicle speed limiter has a higher priority than the driver or the cruise control, the scaling factors FAC_TQ_CRU_SUB (driver or cruise control) and FAC_TQ_REQ_VSL (vehicle speed limiter) are coordinated by a minimum selection. The intervention of the speed limiter is indicated by the boolean LV_INTV_VSL. The coordination between the driver (cruise control) and the vehicle speed limiter is only done if the limiter is active (LV_ACT_VSL = 1).

FAC_TQ_REQ is exported and can also be used from the transmission control unit for calculation of gear shifting points.

Signal flow diagramm:



Application conditions:

Recurrence:	20 ms Attention: this module has to be synchronized with the 10 ms calculations.
Activation:	at every engine state
Deactivation:	-
Initialization:	at reset LV_TQ_CRU_ACT = 0 LV_INTV_CRU = 0 LV_INTV_VSL = 0 FAC_TQ_CRU_SUB = 0 FAC_TQ_REQ = 0
Hint:	If the cruise control is not available the scaling factor FAC_TQ_REQ_CRU and the boolean LV_ACT_CRU have to be set to zero. If the vehicle speed limiter is not available the scaling factor FAC_TQ_REQ_VSL has to be set to its maximum value and the boolean LV_ACT_VSL has to be set to zero.

Formula section:

Maximum selection between driver and cruise control:

```
if        LV_ACT_CRU = 1
then      FAC_TQ_CRU_SUB = Max(FAC_TQ_REQ_CRU, FAC_TQ_REQ_DRIV)
          if      FAC_TQ_REQ_CRU > FAC_TQ_REQ_DRIV
          then    LV_TQ_CRU_ACT = 1
          else    LV_TQ_CRU_ACT = 0
          endif
          if      FAC_TQ_REQ_CRU ≥ FAC_TQ_REQ_DRIV
          then    LV_INTV_CRU = 1
          else    LV_INTV_CRU = 0
          endif
else      FAC_TQ_CRU_SUB = FAC_TQ_REQ_DRIV
          LV_TQ_CRU_ACT = 0
          LV_INTV_CRU = 0
endif
```

Minimum selection between driver or cruise control and vehicle speed limiter:

```
if        LV_ACT_VSL = 1
then      FAC_TQ_REQ = Min(FAC_TQ_CRU_SUB, FAC_TQ_REQ_VSL)
          if      FAC_TQ_REQ_VSL < FAC_TQ_CRU_SUB
          then    LV_INTV_VSL = 1
          else    LV_INTV_VSL = 0
          endif
else      FAC_TQ_REQ = FAC_TQ_CRU_SUB
          LV_INTV_VSL = 0
endif
```

D.28 Torque Request at Clutch General

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_REQ_CLU	O/V	0..FFFFH	0..1.999969	3.052E-5	-
Torque scaling factor for calculation requested torque at clutch					
TQ_REQ_CLU_1	O/V	8000..7FFFH	-1024..1023.97	0.03125	Nm
Requested torque at clutch (intermediate value)					
TQI_MIN_REQ	O/V	8000...7FFFH	-1024..1023.97	0.03125	Nm
Minimum indicated engine torque which can be requested					
TQI_REQ_DRIV	V	8000..7FFFH	-1024..1023.97	0.03125	Nm
Indicated requested torque determined from TQ_REQ_CLU					

Input data:

FAC_TQ_REQ	TQ_MAX_CLU	TQ_MIN_CLU	TQ_LOSS
LV_DT	LV_IS	LV_TQ_MIN_CLU	LV_PU
LV_PUC			

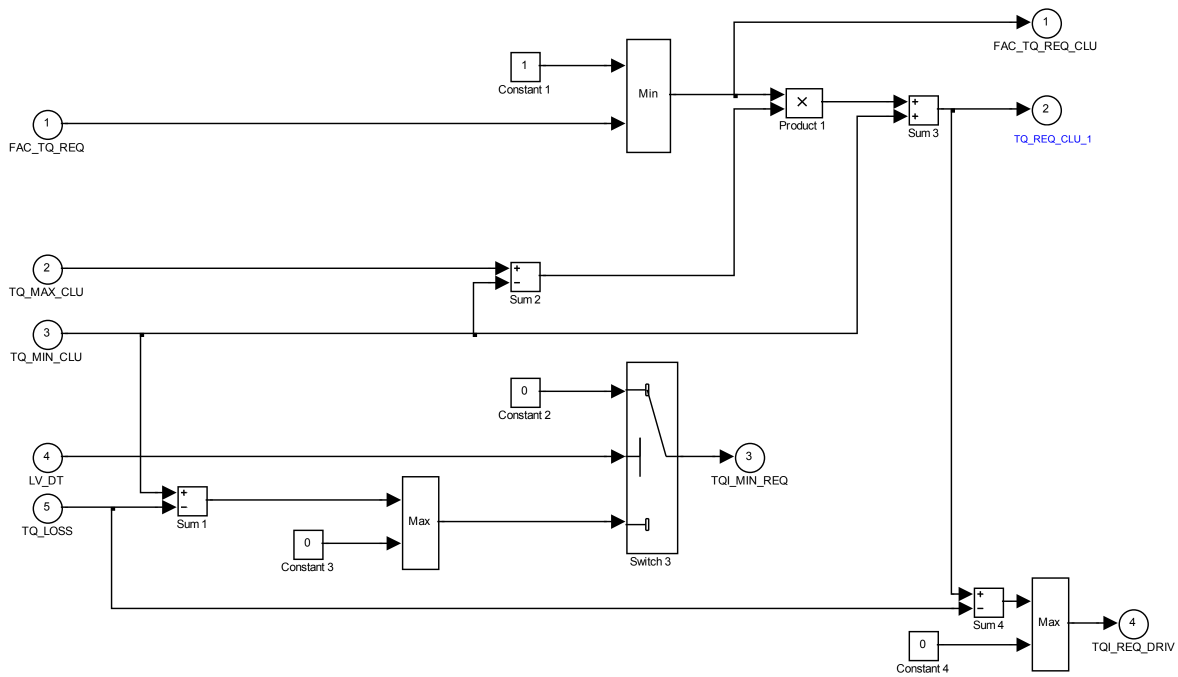
General information:

The module "Torque Request at Clutch General" supplies the torque structure with the torque request at clutch TQ_REQ_CLU_1 from the driver, the cruise control or the vehicle speed limiter.

The torque interpretation factor FAC_TQ_REQ_CLU scales between the minimum torque at clutch TQ_MIN_CLU and the maximum torque at clutch TQ_MAX_CLU.

TQI_MIN_REQ is the minimum indicated engine torque which can be requested from the driver, the cruise control or the vehicle speed limiter and is used inside the torque structure for minimum torque limitations.

FUNCTION DESCRIPTION:



Description:

Application conditions:

Recurrence: **IF** LV_IS == 1 **OR** LV_TQ_MIN_CLU == 0 **OR** LV_PU == 1 **OR** LV_PUC == 1

THEN 10 ms
ELSE 20 ms
ENDIF

Activation: at every engine state

Deactivation: -

Initialization: at reset
TQ_REQ_CLU_1 = 0 Nm
TQI_MIN_REQ = 0 Nm
FAC_TQ_REQ_CLU = 0

D.29 Pedal Value Interpretation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_REQ_DRIV	O/V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque at clutch from driver (with influence of PVS – brake torque limitation or backup value at PVS error)					
FAC_TQ_REQ_DRIV_LIM	V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque after minimum selection between driver and PVS – brake torque limitation					
FAC_TQ_REQ_PV	O/V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque at clutch from driver (without influence of PVS – brake torque limitation or backup value at PVS error)					
FAC_SLOP_DRIV	V	0...FFFFH	0...0.9999847	1.52E-5	-
Factor for switching between low VS and high VS scaling factor					
FAC_TQ_REQ_DRIV_L_VS	V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque at clutch from driver at low vehicle speeds.					
FAC_TQ_REQ_DRIV_H_VS	V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque at clutch from driver at high vehicle speeds.					
FAC_TQ_REQ_DRIV_AT	V	0...FFFFH	0...1.999969	30.517E-6	-
Scaling factor for requested torque at clutch from driver for automatic transmission vehicle					
LV_INTV_PVS_LIH	O/V	0..1H	0..1	1	-
Boolean signals an intervention by PVS – brake torque limitation or at PVS error					
N_FIL_DRIV	V	0...1FE0H	0...8160	1	rpm
Filtered value of engine speed for pedal value interpretation calculation					
FAC_TQ_REQ_DRIV_GEAR	V	0...FFFFH	0...1.999969	30.517E-6	-
Weighting factor for driver request as a function of gear					

Input data:

PV_AV_H	N	VS	LV_AT
FAC_TQ_MAX_PVS_BLS_BTS_LIM	LV_ACT_PVS_BLS_BTS_TQ_LIM	LV_ACT_PVS_LIH	FAC_TQ_MAX_PVS_LIH
GEAR	PV_GRD	TAR_GC	SWI_GS_CAN

Application conditions:

Initialization: at reset

N_FIL_DRIV = 0
FAC_SLOP_DRIV = 0
FAC_TQ_REQ_PV = 0
FAC_TQ_REQ_DRIV = 0
LV_INTV_PVS_LIH = 0

Reccurence: 20 ms

Activation: at every engine state

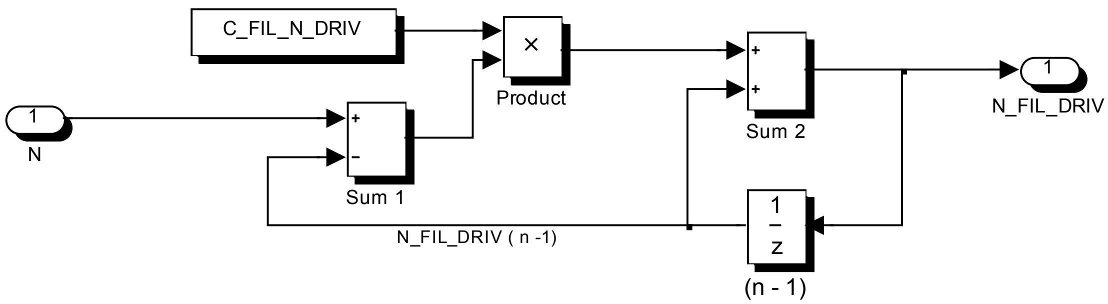
Deactivation: -

D.29.1 Engine speed filtered value

General information:

The engine speed N is filtered by an IIR filter of first order. The result N_FIL_DRIV is used in the subsequent pedal value interpredation.

Formula section:



D.29.2 Management of scaling factor for requested torque at clutch from driver

General information:

The scaling factor FAC_TQ_REQ_PV is the interpretation of the torque wish of the driver depending on engine speed and pedal value.

For variants with automatic transmission vehicle the driver interpretation map IP_FAC_TQ_REQ_DRIV_AT and IP_FAC_TQ_REQ_DRIV_R_AT are used to determine the scaling factor FAC_TQ_REQ_PV.

Determination of FAC_TQ_REQ_DRIV_AT

If	GEAR = 7
Then	FAC_TQ_REQ_DRIV_AT = IP_FAC_TQ_REQ_DRIV_R_AT
Else	FAC_TQ_REQ_DRIV_AT = IP_FAC_TQ_REQ_DRIV_AT

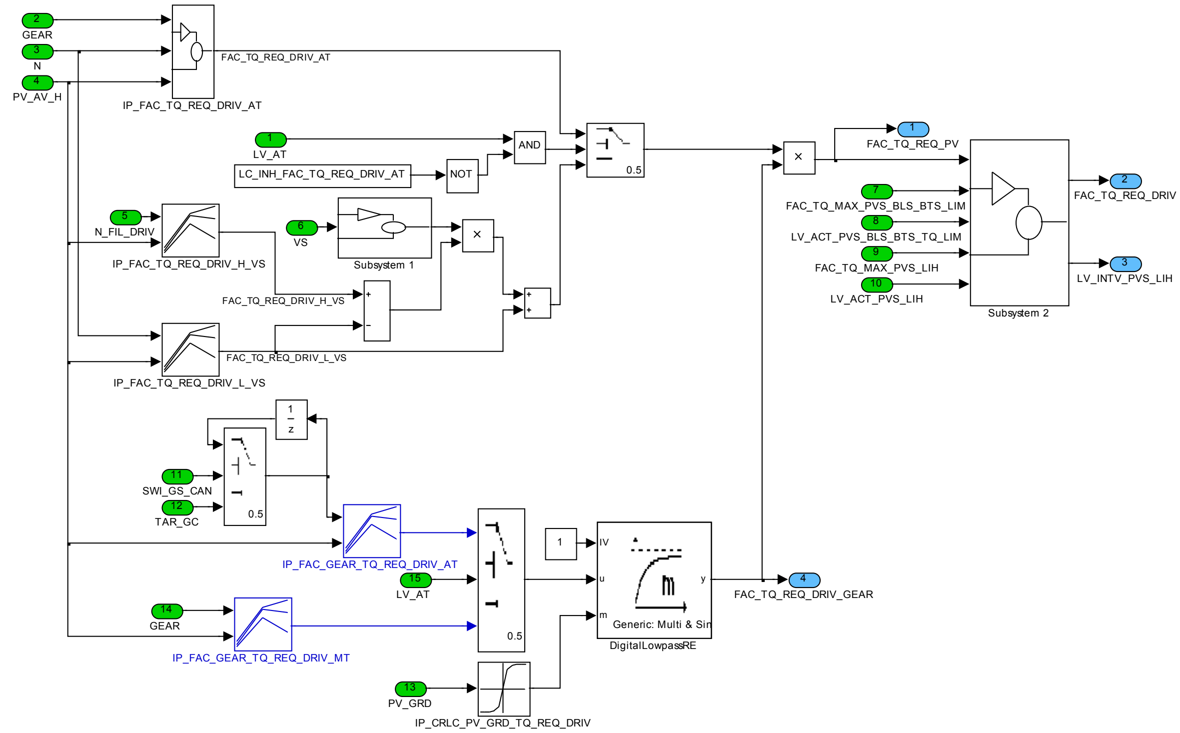
If a manual transmission is detected then the management of scaling factor is divided up in

- scaling factor for low vehicle speeds and
- scaling factor for high vehicle speeds.

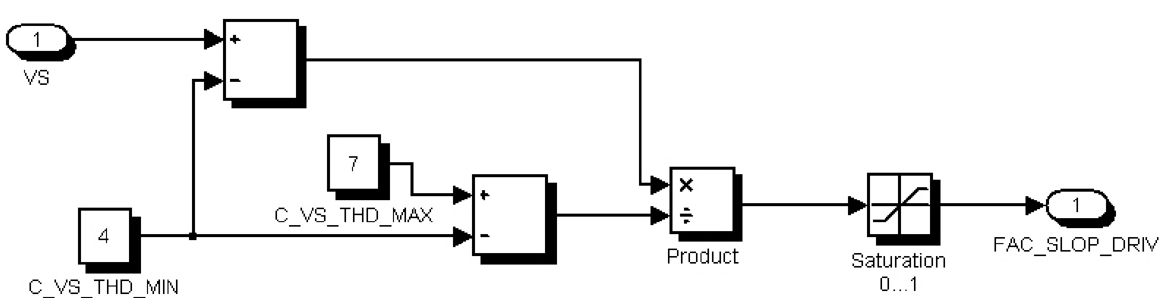
At low vehicle speeds the driver interpretation map IP_FAC_TQ_REQ_DRIV_L_VS is used and at high vehicle speeds the map IP_FAC_TQ_REQ_DRIV_H_VS is used.

An additional scaling factor that is a function of currently engaged gear and pedal position may be calibrated. This factor is filtered with the filter constant based on pedal gradient.

Signal flow diagram:



The weighing factor FAC_SLOP_DRIV between the two driver interpretation maps is set to zero at and below a vehicle speed of C_VS_THD_MIN and is set to one at and above a vehicle speed of C_VS_THD_MAX. Between the two thresholds the factor is linearly interpolated depending on the vehicle speed.
The calculation of FAC_SLOP_DRIV is done in subsystem 1 shown in the following signal flow diagram.



general specification

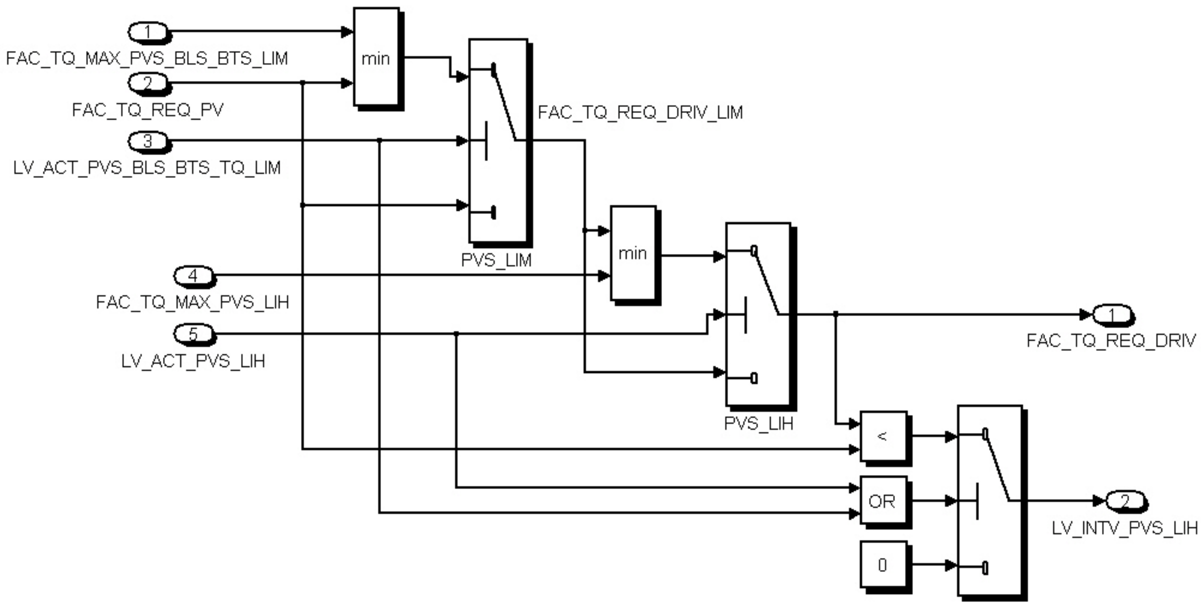
The scaling factor from driver FAC_TQ_REQ_PV can be restricted in two cases:

a) PVS-brake torque limitation, if both the accelerator pedal and the brake pedal are applied simultaneously

b) PVS error, if an error of the pedal value sensor is detected

An active constraint is indicated by the boolean LV_INTV_PVS_LIH.

The restrictions are realized by two minimum selections. The result is the scaling factor FAC_TQ_REQ_DRIV and is calculated in the subsystem 2 shown in following diagram.



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_REQ_DRIV_L_VS	12x12	0...FFFFH	0...1.999969	30.517E-6	-
LDP_N_FAC_TQ_DRIV_L	12	0...1FE0H	0...8160	1	rpm
LDP_PV_AV_H_FAC_TQ_REQ_DRIV_L	12	0...3FFH	0...99.9	0.09765	%
Driver interpretation map for low vehicle speeds.					
IP_FAC_TQ_REQ_DRIV_H_VS	12x12	0...FFFFH	0...1.999969	30.517E-6	-
LDP_N_FIL_DRIV_FAC_TQ_REQ_DRIVH	12	0...1FE0H	0...8160	1	rpm
LDP_PV_AV_H_FAC_TQ_REQ_DRIV_H	12	0...3FFH	0...99.9	0.09765	%
Driver interpretation map for high vehicle speeds.					
IP_FAC_TQ_REQ_DRIV_AT	12x12	0...FFFFH	0...1.999969	30.517E-6	-
LDP_N_FAC_TQ_REQ_DRIV_AT	12	0...1FE0H	0...8160	1	rpm
LDP_PV_AV_H_FAC_TQ_REQ_DRIV_AT	12	0...3FFH	0...99.9	0.09765	%
Driver interpretation map for automatic transmission verhicle					
IP_FAC_TQ_REQ_DRIV_R_AT	12x12	0...FFFFH	0...1.999969	30.517E-6	-
LDP_N_FAC_TQ_REQ_DRIV_R_AT	12	0...1FE0H	0...8160	1	rpm
LDP_PV_AV_H_FAC_TQ_REQ_DRIV_R_AT	12	0...3FFH	0...99.9	0.09765	%
Driver interpretation map for automatic transmission verhicle at GEAR = 7(R)					
C_VS_THD_MAX	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for switching to low VS scaling factor					
C_VS_THD_MIN	1	0...FFH	0...255	1	km/h
Vehicle speed threshold for switching to high VS scaling factor					
LC_INH_FAC_TQ_REQ_DRIV_AT	1	0...1H	0...1	1	-
Logical constant to inhibit the driver map selection of automatic transmission					
C_FIL_N_DRIV	1	0...FFH	0...1	3.92E-3	-
Filter time constant for filtering the engine speed N					
IP_FAC_GEAR_TQ_REQ_DRIV_AT	8	0...FFFFH	0...1.999969	30.517E-6	-
LDP_TAR_GC_FAC_TQ_REQ_DRIV_AT	8	0...7H	0...7	1	-
LDP_PV_AV_H_FAC_GEAR_TQ_REQ_AT	8	0...3FFH	0...99.9	0.09765	%
Gear dependent driver request weigting factor (AT)					
IP_FAC_GEAR_TQ_REQ_DRIV_MT	8	0...FFFFH	0...1.999969	30.517E-6	-
LDP_GEAR_FAC_TQ_REQ_DRIV_MT	8	0..FFH	0...255	1	-
LDP_PV_AV_H_FAC_GEAR_TQ_REQ_MT	8	0...3FFH	0...99.9	0.09765	%
Gear dependent driver request weigting factor (MT)					
IP_CRLC_PV_GRD_TQ_REQ_DRIV	8	0...FFH	0...1	3.92E-3	-
LDP_PV_GRD_TQ_REQ_DRIV	8	0...7FFH	-100...99.902	0.0976	% / 40ms
Filter constant for gear dependent driver request factor					

D.30 Maximum Torque at Clutch

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_BAS_MAX	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum available torque at basic conditions EFF_TOT_BAS					
TQI_MAX	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum indicated torque					
TQI_REF_MAX	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum indicated torque from IP_TQI_REF at MAF_MAX_COR					
TQ_MAX_CLU	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum torque at clutch					

Input data:

EFF_TOT_BAS	IP_TQI_REF	MAF_MAX_COR	N_32
TQ_LOSS_REQ_CLU	NC_ETC_CONF	TQI_POW_MAX_REQ_CLU	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQI_FAC_FL	1	0...FFFFH	0...1.9999	3.0517e-5	-
Maximum torque series variation factor					
IP_TQI_REF	16x12	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32_1_TQDR	16	0...FFH	0...8160	32	rpm
LDPM_MAF_1_TQDR	12	0...FFFFH	0...1389	0.0212	mg/stk
Indicated engine torque at reference conditions					

D.30.1 General Information

TQ_MAX_CLU is the maximum effective torque which can be requested from driver or cruise control. If the torque is not limited by other functions (see below), the value is determined from the map IP_TQI_REF and TQ_LOSS_REQ_CLU. The map IP_TQI_REF is defined in the module 'Reference and Basic Torque'. Note that the original breakpoint MAF is replaced by the breakpoint MAF_MAX_COR!

The parameter C_TQI_FAC_FL allows to increase the maximum torque which can be requested to avoid that the maximum real engine torque is not scooped out totally. But by increasing C_TQI_FAC_FL the backlash at the upper pedal position also increases.

TQI_POW_MAX_REQ_CLU is defined in the module 'Torque based engine power limitation' and describes the upper limit for the maximum indicated engine torque. This parameter can be utilized to perform temporary or permanent torque limitations (e.g. overboost, customer torque limitation) that shall not lead to a backlash at high pedal values.

TQI_BAS_MAX represents the maximum available Basic indicated torque and is utilized in the Torque Reserve Coordination.

Function Description

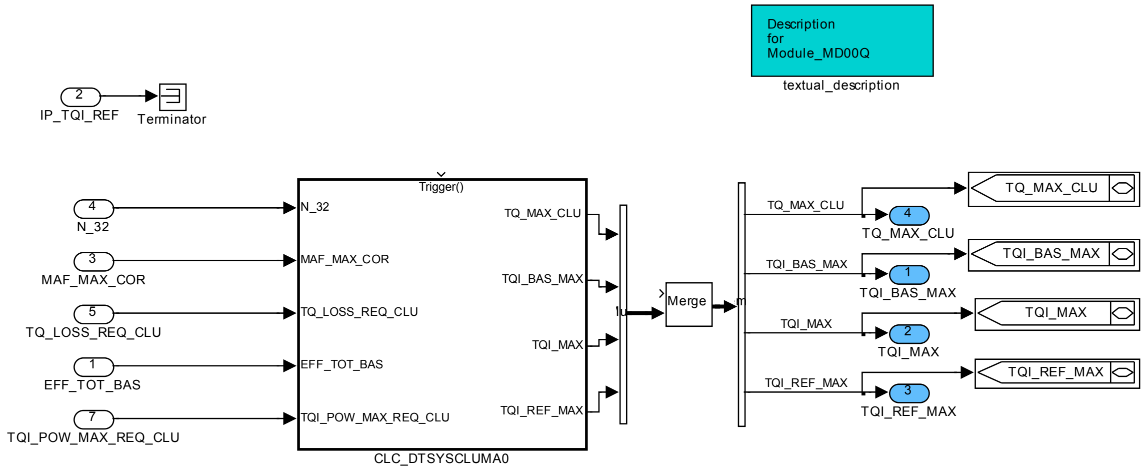


Figure 44 MD00Q

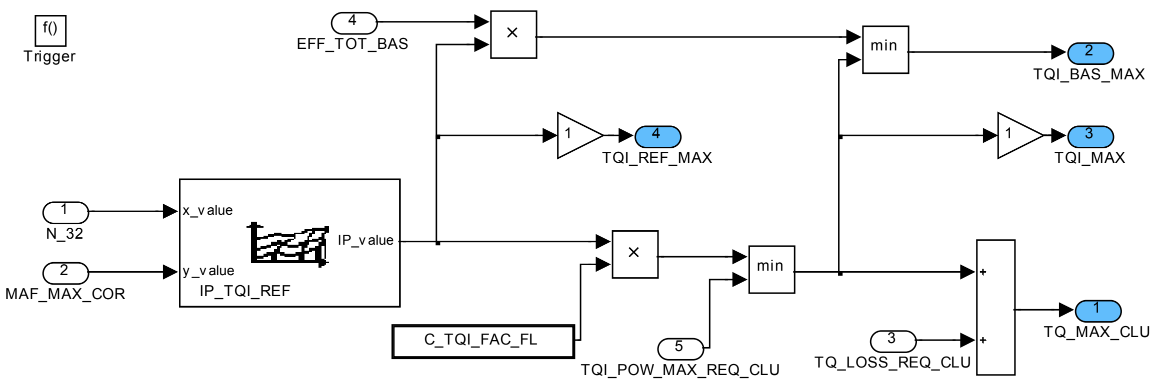


Figure 45 MD00Q/ CLC_DTSYSCLUMA0

D.31 Minimum Torque at Clutch

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TQ_MIN_CLU	O/V	0...1H	0...1	1	-
Logical variable for coordination engine start torque and minimum torque and clutch					
TQI_MIN_REQ_PU	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum requested indicated torque in PU phase					
TQ_MIN_CLU	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum torque at clutch					
FAC_N_DIF_IS	V	8000...7FFFH	-8...7.9998	0.0002441 4	-
Control factor for the engine self stabilization					
TQ_MIN_CLU_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum torque at clutch intermediate value 1					
TQ_MIN_CLU_2	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum torque at clutch intermediate value 2					
TQ_MIN_REQ_PU	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum requested torque at clutch in PU phase					
TQI_MIN_REQ_PU_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Minimum requested indicated torque in PU phase (standard conditions)					
TQI_MIN_REQ_PU_OFS_AMP	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
AMP correction offset in TQI_MIN_REQ_PU					

Input data:

LV_IS	LV_PU	LV_PUC	N_32
N_SP_IS_RATIO	TCO	TQ_CONV	TQ_DIF_I_IS
TQ_LOSS	TQ_LOSS_REQ_CLU	AMP	LV_ES
N_DIF_ST	TQ_ST		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_DIF_TQ_ST_1	1	8000...7FFFH	-32768...32767	1	1/min
N_DIF threshold 1 for deactivating TQ_ST					
C_N_DIF_TQ_ST_2	1	8000...7FFFH	-32768...32767	1	1/min
N_DIF threshold 2 for deactivating TQ_ST					
LC_INH_PU_REQ	1	0...1H	0...1	1	-
Manual inhibition to consider LV_PU for TQ_MIN_CLU calculation					
IP_FAC_N_DIF_IS	8	0...FFFFH	-8...7.9998	0.0002441 4	-
LDP_N_SP_IS_RATIO	8	0...FFFFH	0...7.9999	0.000122	-
Idle self stabilizing factor					
IP_TQI_MIN_REQ_PU_OFS_AMP	8	0...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_AMP_1_TQDR	8	0...FFFFH	0...5434	0.083	hPa
Ambient pressure correction for minimum indicated engine torque in PU phase					
IP_TQI_MIN_REQ_PU	12x4	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32_2_TQDR	12	0...FFH	0...8160	32	rpm
LDPM_TCO_1_TQDR	4	0...FEH	-48...142.5	0.75	°C
Minimum requested indicated torque in PU phase					

D.31.1 General Information

TQ_MIN_CLU is the minimum torque at clutch which can be requested from driver and cruise control.

The integral part of the idle speed controller is an input in TQ_MIN_CLU to avoid a deadzone for the driver torque demand.

The factor FAC_N_DIF_IS models the self stabilizing capability of an SI engine. The factor FAC_N_DIF_IS is a hyperbolic function versus N.

The start torque TQ_ST influences the engine speed curve during start phase. At transition to engine stopped the logical variable LV_TQ_MIN_CLU is initialized with zero.

If LV_TQ_MIN_CLU is 0 then TQ_ST and if LV_TQ_MIN_CLU = 1 then TQ_MIN_CLU_2 is considered for TQ_MIN_CLU. The evaluation of LV_TQ_MIN_CLU is done independently from the engine states PU or PUC.

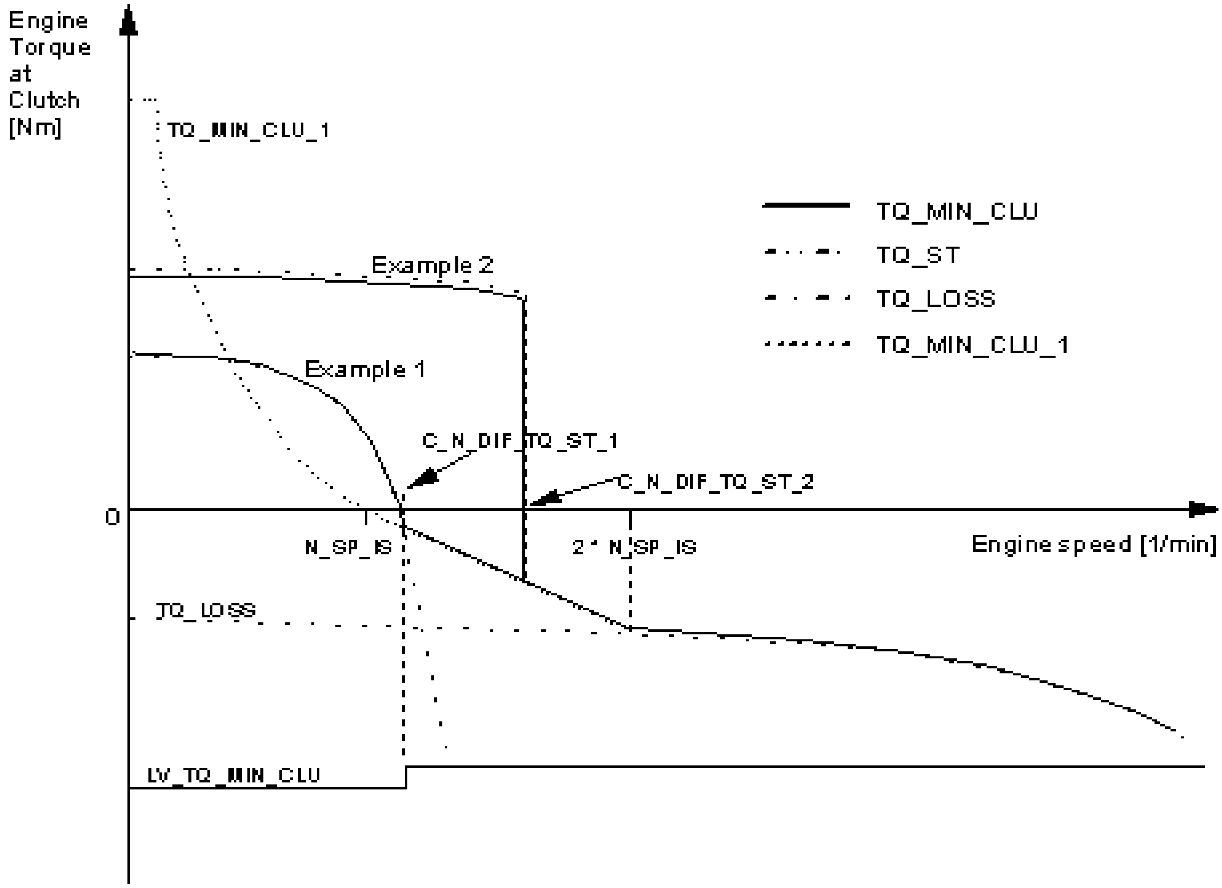
If LV_TQ_MIN_CLU is set to 1 then LV_TQ_MIN_CLU remains to this value until transition engine run to engine stop.

If trailing throttle fuel cut-off is requested (LV_PUC = 1) then TQ_MIN_CLU is set to the minimum possible engine torque TQ_LOSS. It is possible by calibration to achieve the same during trailing throttle (LV_PU = 1) by means of the switch LC_INH_PU_REQ (=0).

TQI_MIN_REQ_PU, which is calculated from the 2-d map IP_TQI_MIN_REQ_PU(N_32, TCO) and the 1-d map IP_TQI_MIN_REQ_PU_OFS_AMP(AMP), is the minimum requested indicated torque if the driver is off the pedal. The maximum between TQ_MIN_REQ_PU and TQ_MIN_CLU_1 is added to TQ_DIF_I_IS and results in TQ_MIN_CLU_2.

The map IP_TQI_MIN_REQ_PU works in combination with map IP_TQI_MIN_PU and has to be tuned in parallel with it. In PU phase both maps determines the steady state ignition angle.

IP_TQI_MIN_PU determines the load whereas IP_TQI_MIN_REQ_PU affects the ignition angle. If both are equal the ignition angle in PU is identical with IGA_BAS_COR. If IP_TQI_MIN_REQ_PU is zero and TQ_MIN_CLU_1 identical with TQ_LOSS then the ignition angle is at IGA_MIN. So the ignition angle in PU can be shifted from IGA_MIN to IGA_BAS_COR by adjusting IP_TQI_MIN_REQ_PU.


Engine
Torque
at
Clutch
[Nm]
TQ_MIN_CLU_1
TQ_MIN_CLU
TQ_ST
TQ_LOSS
TQ_MIN_CLU_1
Example 2
Example 1
C_N_DIF_TQ_ST_1
C_N_DIF_TQ_ST_2
0
N_SP_IS
2 * N_SP_IS
Engine speed [1/min]
TQ_LOSS
LV_TQ_MIN_CLU

Function Description

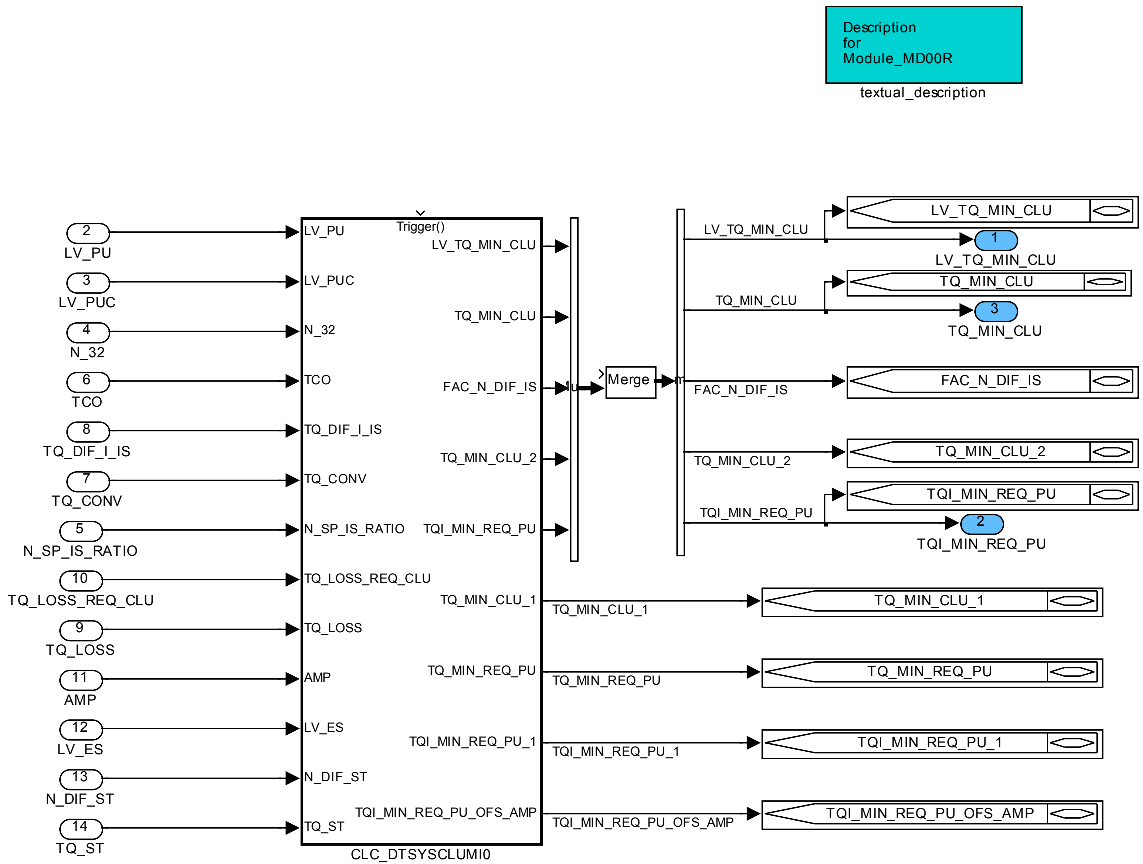


Figure 46 MD00R

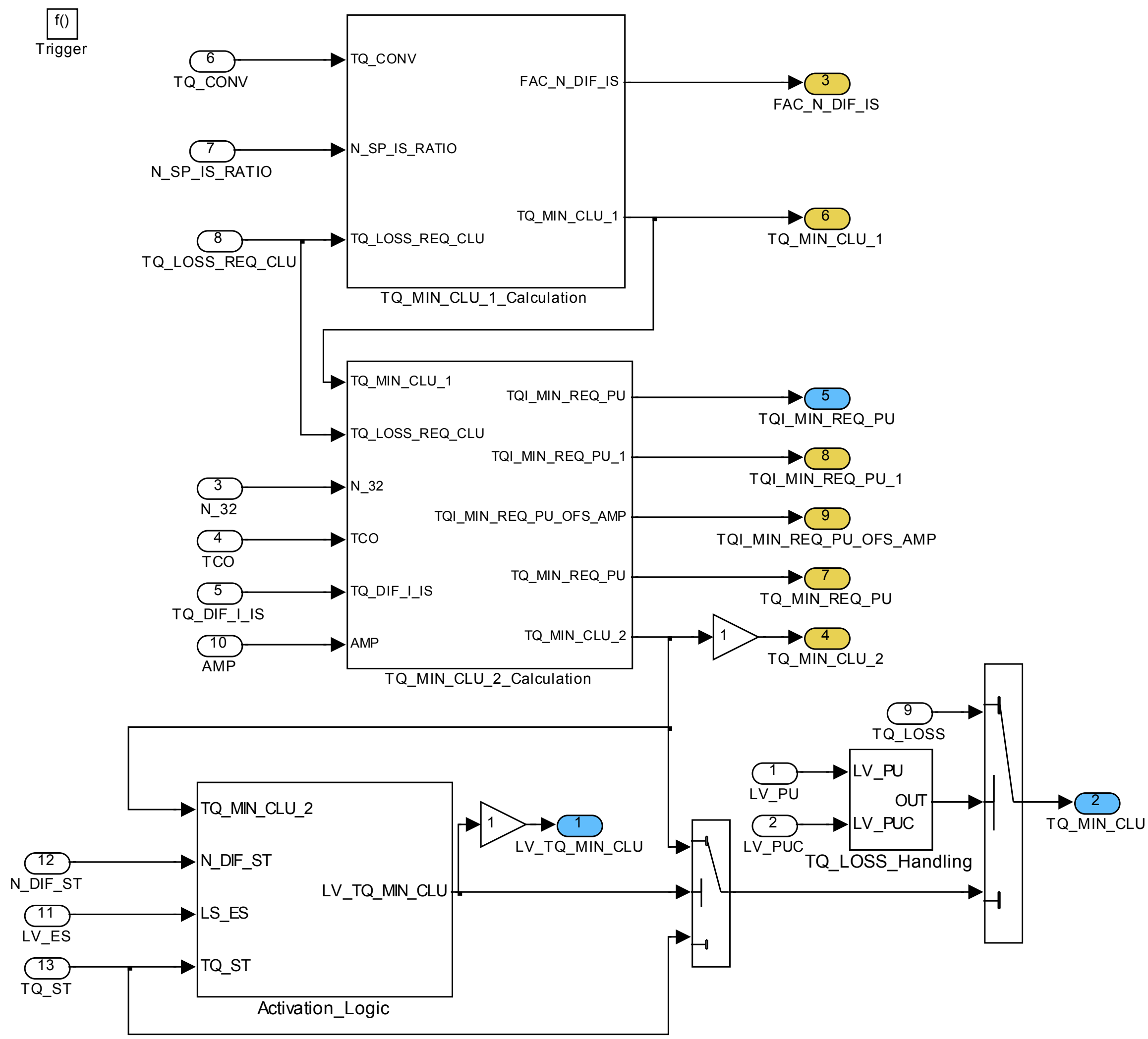


Figure 47 MD00R/ CLC_DTSYSCLUMI0

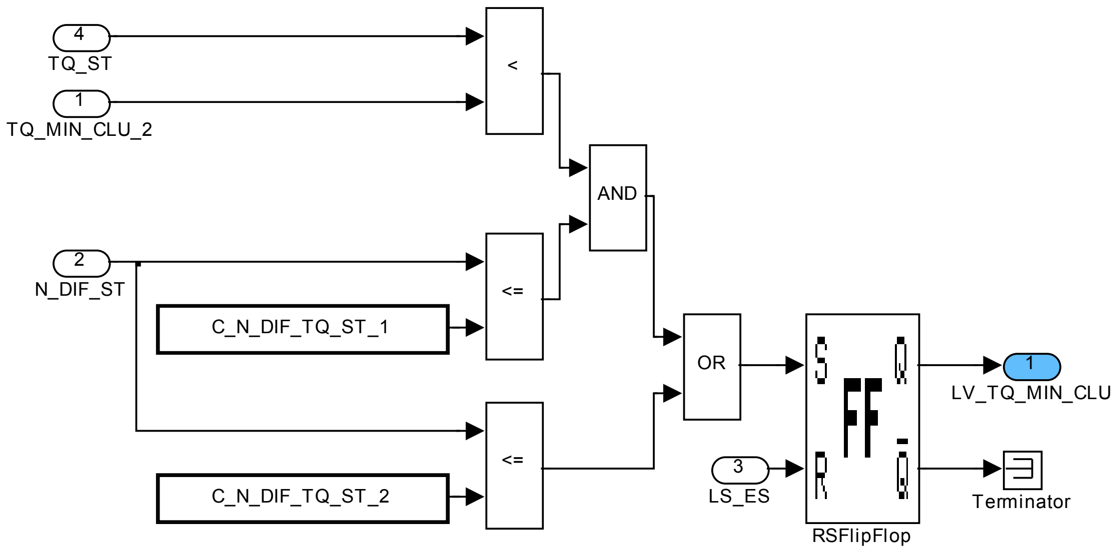


Figure 48 MD00R/ CLC_DTSYSCLUMI0/ Activation_Logic

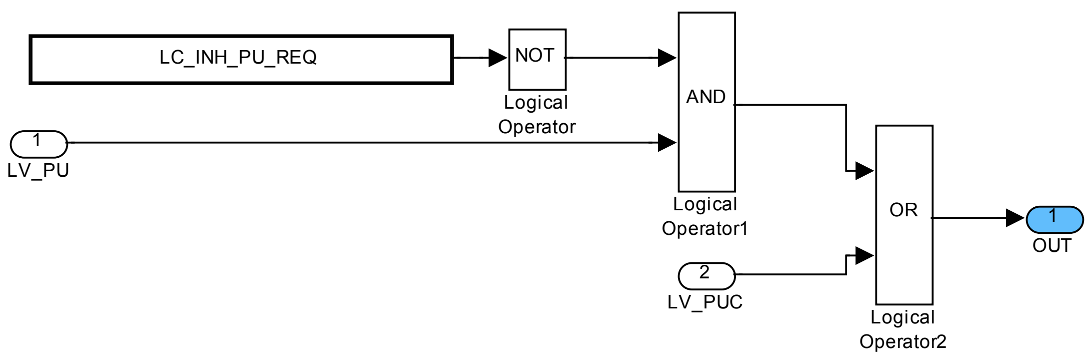


Figure 49 MD00R/ CLC_DTSYSCLUMI0/ TQ_LOSS_Handling

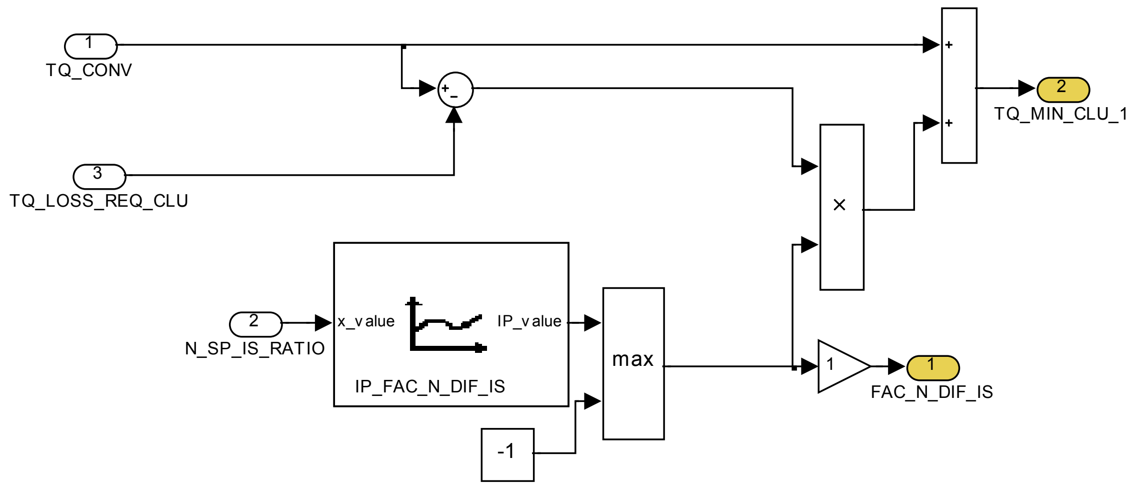


Figure 50 MD00R/ CLC_DTSYSCLUMI0/ TQ_MIN_CLU_1_Calculation

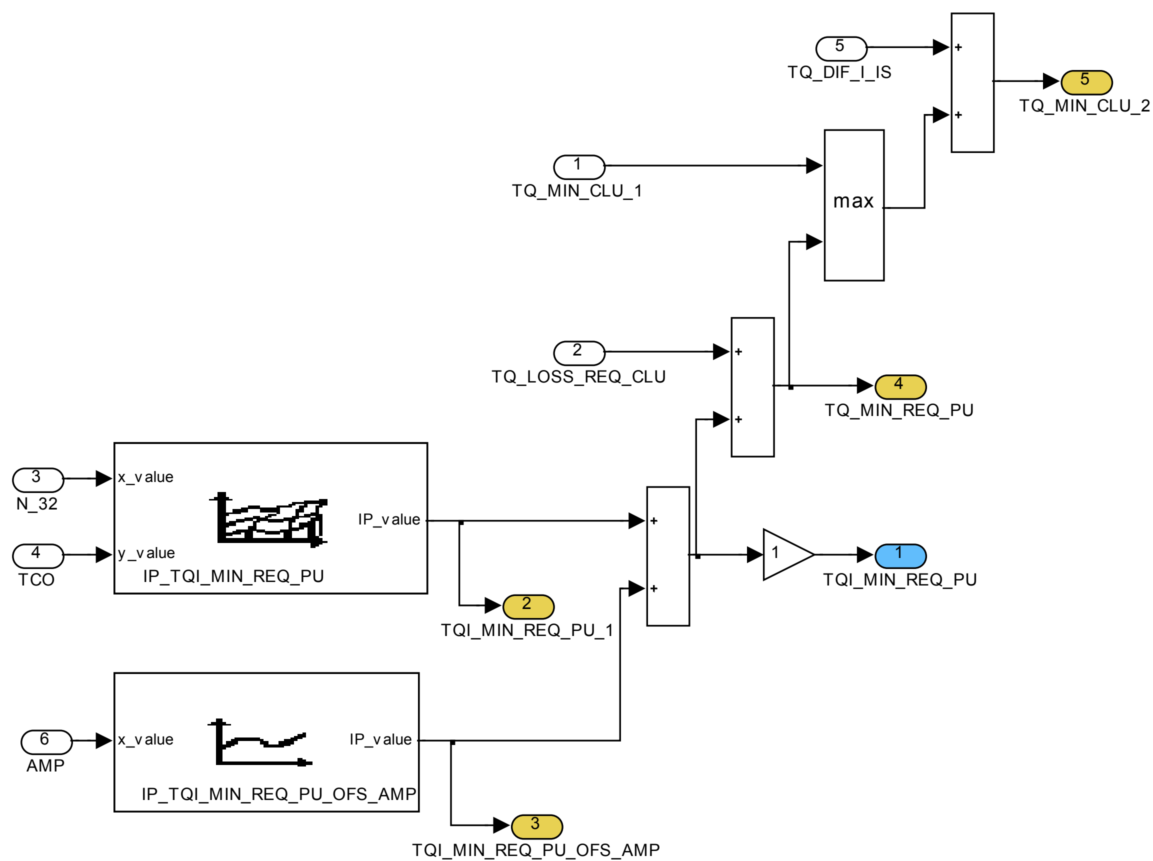


Figure 51 MD00R/ CLC_DTSYSCLUMI0/ TQ_MIN_CLU_2_Calculation

D.32 Converter Torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_CONV	V/O	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Converter Torque					
TQ_ADD_CONV	V/O	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Converter torque reserve					
LV_DLY_N_SP_IS	V/O	0...1	0...1	1	-
Flag for idle speed setpoint change					
TQ_CONV_MDL	V	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Modelled converter torque					
TQ_CONV_MDL_MEM	V	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Stored value for transition calculations					
N_TUR_CONV_INTER	V	8000 ... 7FFF H	-32768 ... 32767	1	rpm
Turbine speed calculated or directly from ETCU					
TOIL_GB_INTER	V	0...FEH	-40 ... 214	1	°C
Oil temperature from gear box (directly from ETCU) or engine oil temperature					
N_RATIO_CONV	V	0...FFH	0 ... 0.996	0.0039	-
Ratio turbine speed from converter and idle speed setpoint (N_TUR_CONV / N_SP_IS)					
TQ_CONV_MDL_BAS	V	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Value which contents the converter torque request at drivetrain opened (modelled value)					
TQ_CONV_MDL_DYN	V	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Target value in case drivetrain is engaged (modelled value)					
TQ_DIF_ADD_CONV	V	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Ramp increment for converter torque reserve at activation/deactivation of the torque reserve					
STATE_TQ_ADD_CONV	V	0 H 1 H 2 H	NO CONVERTER TORQUE RESERVE CONVERTER TORQUE RESERVE RAMPING	-	-
Status of the converter torque reserve: no conv. tq. reserve (0), converter torque reserve (1), ramping (2)					
T_FAC_TQ_CONV_ON	V	0...FFH	0 ... 2.55	0.01	s
Timer for dynamic torque increase					
T_FAC_TQ_CONV_OFF	V	0...FFFFH	0 ... 655.35	0.01	s
Timer for dynamic torque decrease					
T_DLY_N_SP_IS	V	0...FFH	0 ... 2.55	0.01	s
Timer for delaying N_SP_IS change at drivetrain engaged					
T_DLY_TQ_CONV	V	0...FFH	0 ... 2.55	0.01	s
Timer for delaying TQ_CONV increase					
STEP_T	-	0...FFH	0 ... 2.55	0.01	s
Step width for incrementing / decrementing timer					
LV_TQ_ADD_CONV_RAMP_UP	-	0...1	0....1	1	-
Flag to indicate an increasing converter torque reserve request					

Input data:

N_SP_IS	TOIL	LV_ERR_TOIL_GB_ETCU	LV_AT
VS	GR_AT	TQ_DIF_IS_AD_CONV	LV_IS
TOIL_GB	N_TUR_CONV	LV_ERR_N_TUR_CONV_ETCU	LV_DRI
TQ_CONV_ETCU	LV_ES	LV_ERR_TQ_CONV_ETCU	LV_IGK
GEAR	N_32		

D.32.1 Determination of the torque compensation for converter torque (TQ_CONV)

FUNCTION DESCRIPTION:

General information:

TQ_CONV describes the *hydraulic torque losses* inside the converter of an automatic gearbox. They are due to the speed difference between the converter turbine and pump.

In order to assist the idle speed controller, TQ_CONV is calculated for a reference condition with an engine speed at N_SP_IS. As a result, TQ_CONV is bigger than the real converter torque for N < N_SP_IS and smaller for N > N_SP_IS.

The converter torque adaptation value TQ_DIF_IS_AD_CONV is calculated in the idle speed adaptation module. It represents changes in the density of the transmission medium (oil) over the lifetime.

The change of the idle speed setpoint N_SP_IS during gearshift is synchronised with the converter torque request by application of a time delay T_DLY_N_SP_IS after engaging the gear.

It can be chosen if the information of the ETCU (electronic transmission control unit) is used (concerning turbine speed, transmission oil temperature or converter torque itself). There is the possiblility to calibrate transient effects, converter torque requests for drivetrain not engaged (park or neutral) and torque reserves, which are active when the feedforward torque requests are built up. Furthermore, the status of the converter lockup clutch is evaluated.

TQ_CONV is **not** considered in the torque losses, for these are defined as torque requests on the engine side (the converter is transmission side). This implies, that **TQ_CONV is a positive value**.

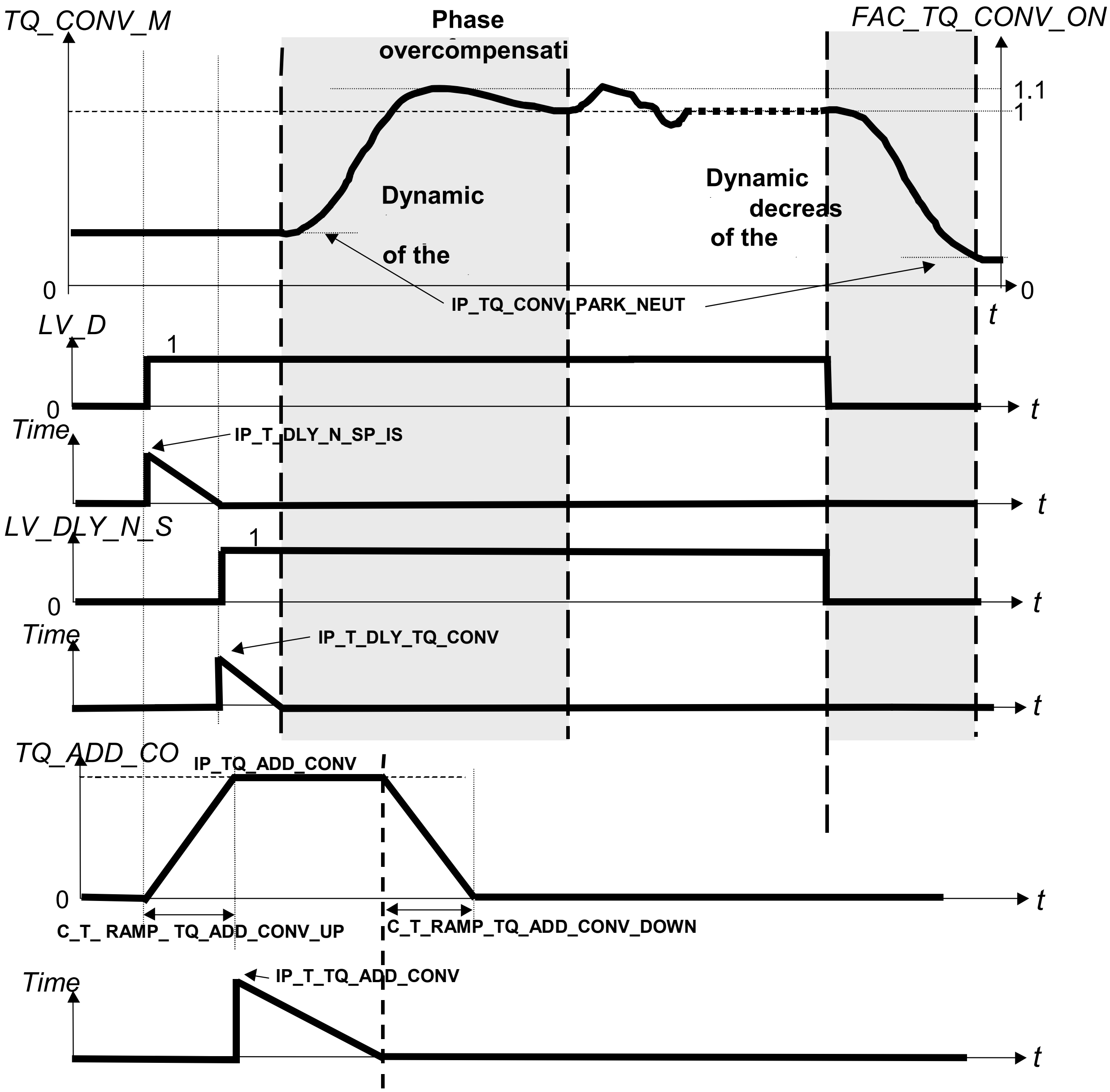


Figure 1: *Modelled converter torque and converter torque reserve*

Figure 1 shows: if LV_DRI is set (i.e. drivetrain is engaged), a calibratible delay time IP_T_DLY_N_SP_IS later the idle speed setpoint is changed (when LV_DLY_N_SP_IS = 1). Afterwards, another delay time IP_T_DLY_TQ_CONV_XX later the converter torque is built up. XX means DRI or RE, depending is Drive or Reverse gear is engaged. The converter torque reserve TQ_ADD_CONV starts ramping as soon as LV_DRI changes from 0 → 1. The final value is held until IP_T_TQ_ADD_CONV is run out **or** – a case, which is not shown in figure 1 – when LV_DRI changes from 1→ 0.

Application conditions:

Initialisation:	TQ_CONV_MDL	= 0 Nm	
	TQ_ADD_CONV	= 0 Nm	
	TQ_CONV	= 0 Nm	
	LV_DLY_N_SP_IS	= 0	
	N_TUR_CONV_INTER	= 0 rpm	
	TOIL_GB_INTER	= 0 °C	
	N_RATIO_CONV	= 0	
	TQ_CONV_MDL_MEM	= 0 Nm	
	TQ_CONV_MDL_DYN	= 0 Nm	
	T_FAC_TQ_CONV_OFF	= 655.35 s	(maximum value)
	T_FAC_TQ_CONV_ON	= 0	
	T_DLY_N_SP_IS	= 0 s	
	T_DLY_TQ_CONV	= 0 s	
	XX	= DRI	
	LV_TQ_ADD_CONV_RAMP_UP	= 0	at reset

Recurrence:
IF LV_IS = 1
THEN 20 ms
ELSE 40 ms
ENDIF

Activation: **IF** LV_AT == 1 **AND** LC_TQ_CONV_INH == 0

Deactivation:
IF LV_AT == 0 **OR** LC_TQ_CONV_INH == 1
THEN TQ_CONV = 0
ENDIF

Scheduling: This function has to be calculated **immediate before the module "Minimum torque at clutch"**

Formula section:

First, a step width for the timer has to be defined:

IF LV_IS == 1
THEN STEP_T = 2 `% STEP_T = (update rate)/10ms`
ELSE STEP_T = 4

ENDIF

If a turbine speed from the ETCU is availible and valid, then this value can be used for the calculation of the converter torque. Otherwise, a modelled value is determined:

IF LC_N_TUR_CONV_ETCU_ACT == 1
AND LV_ERR_N_TUR_CONV_ETCU == 0
AND VS >= C_VS_MIN_N_TUR_CONV_ETCU
THEN N_TUR_CONV_INTER = N_TUR_CONV
ELSE N_TUR_CONV_INTER = IP_N_TUR_CONV
ENDIF

If the oil temperature of the gear box from ETCU is availible, then this value can also be used. Otherwise, the oil temperature of the engine (TOIL) is used as the model value:

IF LC_TOIL_GB_ETCU_ACT == 1
AND LV_ERR_TOIL_GB_ETCU == 0
THEN TOIL_GB_INTER = TOIL_GB
ELSE TOIL_GB_INTER = TOIL
ENDIF

A speed relation is needed for the following calculations:

N_RATIO_CONV = N_TUR_CONV_INTER / N_SP_IS

Converter torque:

As explained before, a modelled value is used in case there is no value from ETCU availible or this value is not valid. For this model, a term IP_FAC_CONV is used which includes the of transmission oil's density and the converter's circuit diameter. The value IP_FAC_TQ_CONV(N_RATIO_CONV) is the power number of the converter. A distinction in the gear position between park/neutral and gear engaged is made:

IF(1) LC_TQ_CONV_ETCU_ACT == 1
AND LV_ERR_TQ_CONV_ETCU == 0
THEN(1) TQ_CONV = TQ_CONV_ETCU + TQ_DIF_IS_AD_CONV
ELSEIF(1) LC_TQ_CONV_ETCU_ACT == 0
OR LV_ERR_TQ_CONV_ETCU == 1
THEN(1) {

IF(2) **transition** LV_DRI 0 → 1
THEN(2) T_DLY_N_SP_IS = IP_T_DLY_N_SP_IS % start decrementing timer
% with STEP_T
ENDIF(2);

IF(3) **transition** T_DLY_N_SP_IS ≠ 0 → 0
THEN(3) LV_DLY_N_SP_IS = 1
T_DLY_TQ_CONV = IP_T_DLY_TQ_CONV_XX % start decrementing
% timer with STEP_T
ENDIF(3);

IF(4) LC_TQ_CONV_NEW_INH == 0
{
THEN(4) TQ_CONV_MDL_BAS = IP_TQ_CONV_PARK_NEUT
TQ_CONV_MDL_DYN = TQ_DIF_IS_AD_CONV +
IP_FAC_CONV * IP_FAC_TQ_CONV(N_RATIO_CONV)* $N\_SP\_IS^{2}$

IF(5) **transition** LV_DRI 1 → 0
THEN(5) T_FAC_TQ_CONV_OFF = 0;
start time counter T_FAC_TQ_CONV_OFF % start incrementing
% counter with STEP_T
TQ_CONV_MDL_MEM = $TQ\_CONV\_MDL_{k-1}$
ENDIF(5);

IF(6) LV_DRI == 0
OR **[** LV_DRI == 1 **AND** **{** T_DLY_N_SP_IS ≠ 0 **OR** T_DLY_TQ_CONV ≠ 0 **}** **]**
THEN(6) $TQ\_CONV\_MDL_{k}$ = TQ_CONV_MDL_BAS +
(TQ_CONV_MDL_MEM - TQ_CONV_MDL_BAS) * IP_FAC_TQ_CONV_OFF_XX
ENDIF(6);

IF(7) LV_DRI == 1 **AND transition {** T_DLY_TQ_CONV $\neq$ 0 $\rightarrow$ 0 **}**

THEN(7) *T_FAC_TQ_CONV_ON* = 0;

start time counter T_FAC_TQ_CONV_ON `% start incrementing`
`% counter with STEP_T`

TQ_CONV_MDL_MEM = $TQ\_CONV\_MDL_{k\text{-}1}$

ENDIF(7);

IF(8) LV_DRI == 1 **AND {** T_DLY_N_SP_IS == 0 **AND** T_DLY_TQ_CONV == 0 **}**

THEN(8) $TQ\_CONV\_MDL_k$ = TQ_CONV_MDL_MEM +
(TQ_CONV_MDL_DYN - TQ_CONV_MDL_MEM) * IP_FAC_TQ_CONV_ON_XX

ENDIF(8);

TQ_CONV = TQ_CONV_MDL;

}

ELSEIF(4) LC_TQ_CONV_NEW_INH == 1

TQ_CONV = IP_FAC_CONV * IP_FAC_TQ_CONV * N_SP_IS ^{2}
+ TQ_DIF_IS_AD_CONV

ENDIF(4) } **`% end condition part for modelled value`**

ENDIF(1)

Drive/Reverse dependent table :

Some tables (IP_T_DLY_TQ_CONV_XX, IP_FAC_TQ_CONV_ON_XX, IP_FAC_TQ_CONV_OFF_XX) are depending on which gear is selected : DRIVE or REVERSE. In case of transition to NEUTRAL, it is necessary to memorize the gear (: DRIVE or REVERSE) which was used before NEUTRAL.

If GEAR = 7 `% Reverse engaged or disengaged`

Then XX = RE

Elsif GEAR = 1 or 2 or 3 or 4 or 5 `% Drive engaged or disengaged`

Then XX = DRI

Else XX remains unchanged `% previous setting (RE or DRI)`
`% is kept in Neutral (gear = 0)`

Endif

D.32.2 Determination of the torque reserve for converter torque (TQ_ADD_CONV)

If the real torque request of the converter is higher than the feed forward controlled one (TQ_CONV), then the idle speed controller should compensate this difference. Especially in case of converter torque increase (at transition), this possible difference between reality and feed-forward control must be compensated very fast. To make this possible, a torque reserve can be built up. This torque reserve TQ_ADD_CONV can be built up and reduced with ramps. It can be defined in dependance of the transmission oil temperature and takes effect for a calibratible time IP_T_TQ_ADD_CONV.

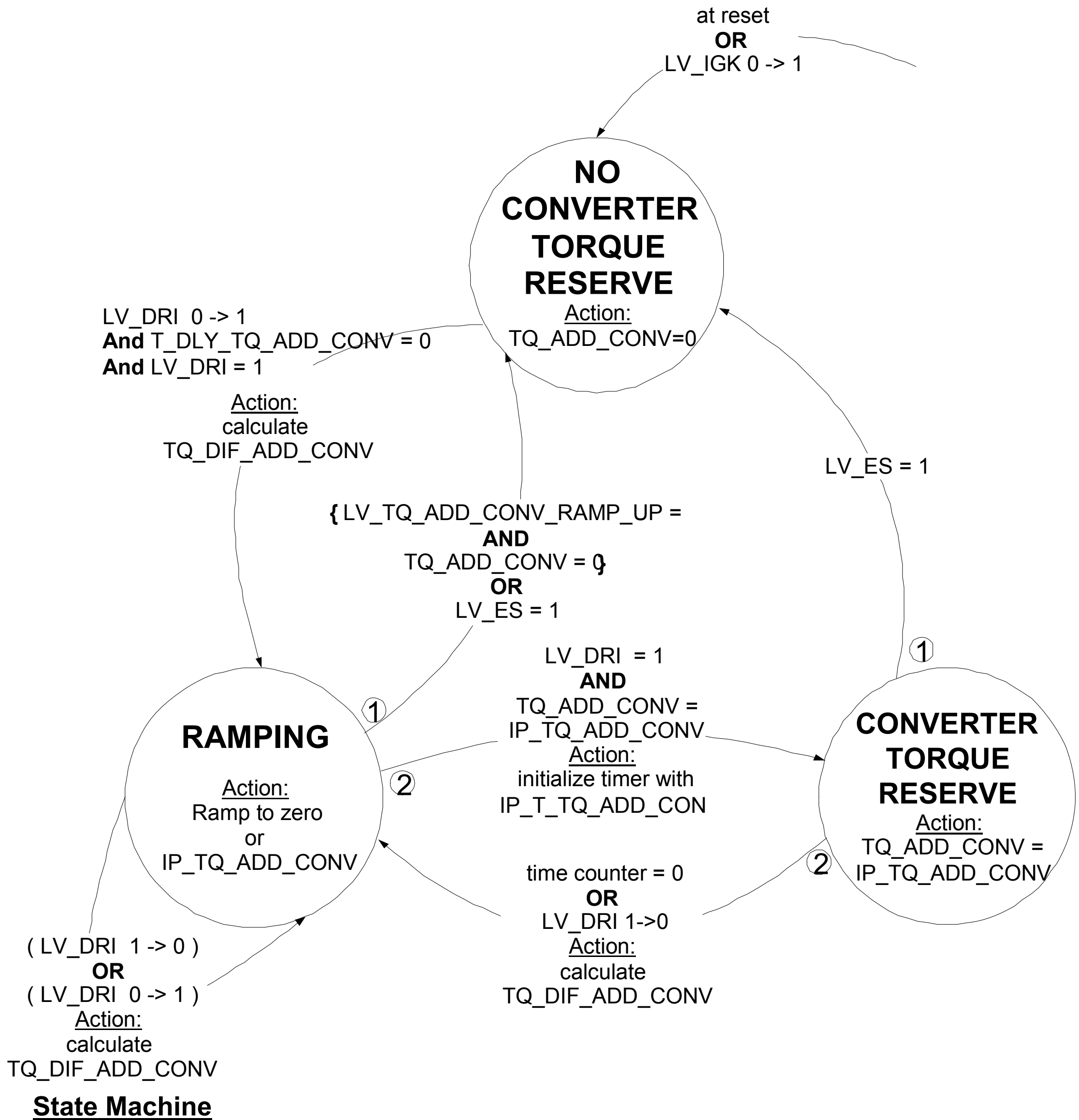


Figure 2: *State machine of the converter torque reserve*

To see each transition condition, please refer the *Figure 2:* ***State machine of the converter torque reserve***.

A. NO CONVERTER TORQUE RESERVE Mode:

Actions:

STATE_TQ_ADD_CONV = NO CONVERTER TORQUE RESERVE

TQ_ADD_CONV = 0

Actions during transition to RAMPING Mode:

Calculation of T_DLY_TQ_ADD_CONV

If GEAR = 7(reverse),

Then T_DLY_TQ_ADD_CONV = IP_T_DLY_TQ_ADD_CONV_RE

Else if GEAR = 1 or 2 or 3 or 4 or 5

Then T_DLY_TQ_ADD_CONV = IP_T_DLY_TQ_ADD_CONV_DRI

endif

After delay time T_DLY_TQ_ADD_CONV,

TQ_DIF_ADD_CONV =

IP_TQ_ADD_CONV / C_T_RAMP_TQ_ADD_CONV_UP * update rate

LV_TQ_ADD_CONV_RAMP_UP = 1

Note) TQ_DIF_ADD_CONV is only one time calculated at the transition of LV_DRI.

B. RAMPING Mode:

Actions:

STATE_TQ_ADD_CONV = RAMPING

IF(1) LV_TQ_ADD_CONV_RAMP_UP == 1

THEN(1)

IF(2) TQ_ADD_CONV $\neq$ IP_TQ_ADD_CONV

THEN(2) $TQ\_ADD\_CONV_k = TQ\_ADD\_CONV_{k-1} + TQ\_DIF\_ADD\_CONV$

IF TQ_ADD_CONV > IP_TQ_ADD_CONV

THEN TQ_ADD_CONV = IP_TQ_ADD_CONV

ENDIF

ENDIF(2)

ELSEIF(1) LV_TQ_ADD_CONV_RAMP_UP == 0

IF(3) TQ_ADD_CONV $\neq$ 0

THEN(3) $TQ\_ADD\_CONV_k = TQ\_ADD\_CONV_{k-1}$ - TQ_DIF_ADD_CONV

IF TQ_ADD_CONV < 0

THEN TQ_ADD_CONV = 0

ENDIF

ENDIF(3)

ENDIF(1)

<u>Actions during transition to CONVERTER TORQUE RESERVE Mode:</u>

Initialize timer with IP_T_TQ_ADD_CONV

<u>Actions during transition RAMPING Mode to RAMPING Mode:</u>

IF LV_DRI 0 → 1

THEN TQ_DIF_ADD_CONV = (IP_TQ_ADD_CONV – TQ_ADD_CONV) / C_T_RAMP_TQ_ADD_CONV_UP * update rate

LV_TQ_ADD_CONV_RAMP_UP = 1

ELSEIF LV_DRI 1 → 0

THEN TQ_DIF_ADD_CONV =

= TQ_ADD_CONV / C_T_RAMP_TQ_ADD_CONV_DOWN * update rate

LV_TQ_ADD_CONV_RAMP_UP = 0

ENDIF

Note) TQ_DIF_ADD_CONV is only one time calculated at the transition of LV_DRI.

<u>C. CONVERTER TORQUE RESERVE Mode :</u>

<u>Actions:</u>

STATE_TQ_ADD_CONV = CONVERTER TORQUE RESERVE

TQ_ADD_CONV = IP_TQ_ADD_CONV

ramp timer counter down to zero

Actions during transition to RAMPING Mode:

TQ_DIF_ADD_CONV =

TQ_ADD_CONV / C_T_RAMP_TQ_ADD_CONV_DOWN * update rate

LV_TQ_ADD_CONV_RAMP_UP = 0

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_CONV	8	0...FFFFH	0...0.000488	7.446E-9	$Nm*rpm^{-2}$
LDP_TOIL_GB_INTER_FAC_CONV	8	0...FEH	-40 ... 214	1	°C
Converter characteristic					
IP_TQ_CONV_PARK_NEUT	8*6	0...FFFFH	-1024...1023.97	0.03125	Nm
LDP_TOIL_GB_INTER	8	0...FEH	-40 ... 214	1	°C
LDP_N_IP_TQ_CONV_PARK_NEUT	6	0...FFH	0...8160	32	rpm
Converter characteristic in park or neutral position					
IP_FAC_TQ_CONV	8	0...FFFFH	0...1.99996	3.05E-5	-
LDP_N_RATIO_CONV_FAC_TQ_CONV	8	0...FFH	0...0.996	0.0039	-
Torque request factor					
IP_T_DLY_TQ_CONV_DRI	8	0...FFH	0...2550	10	ms
LDPM_TOIL_GB_INTER_T_DLY_CONV	8	0...FEH	-40 ... 214	1	°C
Time delay for TQ_CONV activation – DRIVE selected					
IP_T_DLY_TQ_CONV_RE	8	0...FFH	0...2550	10	ms
LDPM_TOIL_GB_INTER_T_DLY_CONV	8	0...FF H	-40 ... 214	1	°C
Time delay for TQ_CONV activation – REVERSE selected					
IP_T_DLY_N_SP_IS	8	0...FFH	0...2550	10	ms
LDP_TOIL_GB_INTER_T_DLY_N_SP_IS	8	0...FEH	-40 ... 214	1	°C
Time delay for change of idle speed setpoint					
IP_N_TUR_CONV	9x2	0...FFFFH	-32768 ... 32767	1	rpm
LDP_GR_AT_N_TUR_CONV	9	0...8H	0...8	1	-
LDP_VS_N_TUR_CONV	2	0...FFH	0...255	1	km/h
Turbine speed					
LC_N_TUR_CONV_ETCU_ACT	1	0...1	0...1	1	-
Logical calibration bit for switching between turbine speed calculated and ETCU turbine speed					
LC_TOIL_GB_ETCU_ACT	1	0...1	0...1	1	-
Logical calibration bit for switching between oil temperature from gear box (from ETCU) or the engine oil temp.					
LC_TQ_CONV_ETCU_ACT	1	0...1	0...1	1	-
Logical calibration bit for switching between modeled converter torque and converter torque from ETCU					
IP_FAC_TQ_CONV_ON_DRI	8*6	0...FFH	0...1.992	0.0078	-
LDPM_T_FAC_TQ_CONV_ON	8	0...FFH	0 ... 2.55	0.01	s
LDPM_TOIL_GB_INTER_FAC_TQ_CONV	6	0...FEH	-40 ... 214	1	°C
Weighting factor for the dynamic behaviour of the converter torque in case of transition NEUTRAL --> DRIVE					
IP_FAC_TQ_CONV_ON_RE	8*6	0...FFH	0...1.992	0.0078	-
LDPM_T_FAC_TQ_CONV_ON	8	0...FFH	0 ... 2.55	0.01	s
LDPM_TOIL_GB_INTER_FAC_TQ_CONV	6	0...FEH	-40 ... 214	1	°C
Weighting factor for the dynamic behaviour of the converter torque in case of transition NEUTRAL --> REVERSE					
IP_FAC_TQ_CONV_OFF_DRI	8*6	0...FFH	0...1.992	0.0078	-
LDPM_T_FAC_TQ_CONV_OFF	8	0...FFFFH	0 ... 655.35	0.01	S
LDPM_TOIL_GB_INTER_FAC_TQ_CONV	6	0...FEH	-40 ... 214	1	°C
Weighting factor for the dynamic behaviour of the converter torque in case of transition DRIVE --> NEUTRAL					
IP_FAC_TQ_CONV_OFF_RE	8*6	0...FFH	0...1.992	0.0078	-
LDPM_T_FAC_TQ_CONV_OFF	8	0...FFFFH	0 ... 655.35	0.01	s
LDPM_TOIL_GB_INTER_FAC_TQ_CONV	6	0...FEH	-40 ... 214	1	°C
Weighting factor for the dynamic behaviour of the converter torque in case of transition REVERSE --> NEUTRAL					
IP_TQ_ADD_CONV	8	0...FFFFH	-1024 ... 1023.97	0.03125	Nm
LDP_TOIL_GB_INTER_TQ_ADD_CONV	8	0...FEH	-40 ... 214	1	°C
Torque reserve for converter torque					
C_T_RAMP_TQ_ADD_CONV_UP	1	0...FFH	0 ... 2.55	0.01	s
Ramp time for increase of converter torque reserve					

C_T_RAMP_TQ_ADD_CONV_DOWN	1	0...FFH	0 ... 2.55	0.01	s
Ramp time for decrease of converter torque reserve					
IP_T_TQ_ADD_CONV	8	0...FFH	0 ... 2.55	0.01	s
LDP_TOIL_GB_INTER_T_TQ_ADD_CONV	8	0...FE H	-40 ... 214	1	°C
Time of active torque reserve for converter (without ramping time)					
LC_TQ_CONV_INH	1	0...1	0...1	1	-
Bit for inhibiting this function (LC_TQ_CONV_INH = 1: function disabled, TQ_CONV = 0)					
LC_TQ_CONV_NEW_INH	1	0...1	0...1	1	-
Bit for inhibiting the new concerter torque calculation (LC_TQ_CONV_NEW_INH = 1 use the old algorithms)					
C_VS_MIN_N_TUR_CONV_ETCU	1	0...FFH	0...255	1	km/h
Speed limit for switching between N_TUR_CONV from model and N_TUR_CONV from ETCU					
IP_T_DLY_TQ_ADD_CONV_DRI	8	0...FFH	0...2550	10	ms
LDPM_TOIL_GB_INTER_T_DLY_CONV	8	0...FF H	-40 ... 214	1	°C
Time delay for TQ_ADD_CONV activation – Drive selected					
IP_T_DLY_TQ_ADD_CONV_RE	8	0...FFH	0...2550	10	ms
LDPM_TOIL_GB_INTER_T_DLY_CONV	8	0...FF H	-40 ... 214	1	°C
Time delay for TQ_ADD_CONV activation – REVERSE selected					

D.33 Torque reserve for transmission

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_TRANS	O	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Central torque reserve for transmission					

Input data:

TQ_ADD_CONV			

FUNCTION DESCRIPTION:

General information:

All torque reserve requests which are needed for transmission (f.e. torque reserve requests for converter, CVT or automized manual transmission) are resumed here. The central output is TQ_ADD_TRANS, which is an input to the torque reserve coordination.

Application conditions:

Initialisation: TQ_ADD_TRANS = 0 Nm

Recurrence: 10 ms

Activation: at every engine state

Deactivation: -

Scheduling: *This module should be calculated **after** the modules to determine the*

Transmission torque reserve requests and **before the torque reserve coordination.**

Formula section:

TQ_ADD_TRANS = TQ_ADD_CONV

D.34 Torque Transient Parameters

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_TRA	V/O	0 .. FF H	0 .. 2.55	0.01	sec
Time constant for generic Torque Transient algorithm					
TQ_RANGE_SEL_OFS	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Offset for Torque range selection depending on driver torque request					
DELTA_N_TUR_ENG	V	8000...7FFFH	-32768...32767	1	rpm
Speed difference between torque converter turbine speed and engine speed					
LV_INH_TQ_FIL	V	0...1H	0...1	1	-
Inhibition flag for Torque transient filtering calculation					
T_INC_TRA_1_MT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 1 for MT					
T_INC_TRA_2_MT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 2 for MT					
T_INC_TRA_3_MT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 3 for MT					
T_INC_TRA_1_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 1 for AT					
T_INC_TRA_2_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 2 for AT					
T_INC_TRA_3_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque increase 3 for AT					
T_DEC_TRA_1_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque decrease 1 for AT					
T_DEC_TRA_2_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque decrease 2 for AT					
T_DEC_TRA_3_AT	V	0...FFH	0...2.55	0.01	Sec
Time constant for torque decrease 3 for AT					
LV_TQ_TRA_FIL_MT_GS	V	0...1H	0...1	1	-
Flag for torque transient filter in case of gear-shifting and tip-in for MT					
TQ_TRA_THD_DOWN_1	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Torque range threshold 1 (lower) for decreasing torque					
TQ_TRA_THD_DOWN_2	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Torque range threshold 2 (upper) for decreasing torque					
TQ_TRA_THD_UP_1	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Torque range threshold 1 (lower) for increasing torque					
TQ_TRA_THD_UP_2	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Torque range threshold 2 (upper) for increasing torque					

Input data:

STATE_TQ_RANGE	STATE_TQ_PHA	N_32	LV_IS
LV_AT	GEAR	TQ_REQ_CLU	PV_AV
N	N_TUR_CONV_RAW	SWI_CC	LV_CT
LV_DT	VS	LV_GS_ENA	LV_DROF_AT_ACT

FUNCTION DESCRIPTION:

General information:

The T_TRA parameter is the target value for the filtered signal when to reach the setpoint (unfiltered torque request).
This revision support manual and automated transmission. It allows variant coding.
TQ Transient Filtering for AT considering converter clutch lock-up state : Normally, the level of tip-out shock when the converter clutch is fully lock up is greater than that of no lock up control. TQ filtering time(T_INC/DEC TRA) can be adjusted separetely in order to improve the tip-out shock in case of fully lock-up (SWI_CC = 2)
TQ Transient Filtering for MT considering gear-shift and tip-in : When gear is shifted while vehicle running, Separated TQ filtering time can be applied in order to prevent vibration/shock at tip-in condition.
When TCU request TQ reduction control (spark time retard control), TQ filtering can be inhibited. (T_TRA = 0, by setting of LC_TQ_FIL_OFF_SPK_RTD_TCU = 1).

Depending on engine speed and selected gear, different thresholds can be set for the transition between the 3 torque transient ranges. It is important that before the torque transient range is determined, this threhold determination is performed. The most efficient solution is to call this spec via 2 function calls, the first one to determine only the torque transient thresholds, the second one to determine the transient time T_TRA.

For drive-off in an AT vehicle, it is possible to limit the engine torque via a different function (Driver Request Limitation) in this case it is possible to disable the torque transient function. It is possible to disable the function only for increasing torques, only for decreasing torques, or in all cases.

Application conditions:

Initialisation: *T_TRA = 0*

TQ_RANGE_SEL_OFS = 0

DELTA_N_TUR_ENG = 0

LV_INH_TQ_FIL = 0

All T_INC/DEC_TRA_AT/MT_1/2/3 = 0

LV_TQ_TRA_FIL_MT_GS = 0

TQ_TRA_THD_UP/DOWN_1/2 = 0

Recurrence: 10ms function call from "Torque Transient"

Activation: every engine state

TQ filtering inhibition in case of TQ reduction request from TCU:

If LV_AT = 1

And LC_TQ_FIL_OFF_SPK_RTD_TCU = 1

And LV_GS_ENA = 1

Then

LV_INH_TQ_FIL = 1 // Deactivate TQ filtering

T_TRA = 0

Else

LV_INH_TQ_FIL = 0 // Activate TQ filtering(Cal. T_TRA)

EndIf

Deactivation: LV_INH_TQ_FIL = 1

Formula section:

Calculation of speed difference between torque converter turbine speed and engine speed : DELTA_N_TUR_ENG

IF LV_AT = 1

THEN DELTA_N_TUR_ENG = N_TUR_CONV_RAW - N

ELSE DELTA_N_TUR_ENG = 0

TQ filtering for AT considering converter clutch lock-up condition :

```
If      LV_AT = 1               // AT vehicle
Then
    If          SWI_CC = 2 or 3        // fully locked up
        and     abs(N – N_TUR_CONV_RAW) ≤ C_N_DIF_LOCK_CONV_CS
    Then
        // Tip-in condition
        T_INC_TRA_1_AT = IP_T_INC_TRA_1_AT_LOCK_CONV_CS (N_32, GEAR)
        T_INC_TRA_2_AT = IP_T_INC_TRA_2_AT_LOCK_CONV_CS (N_32, GEAR)
        T_INC_TRA_3_AT = IP_T_INC_TRA_3_AT_LOCK_CONV_CS (N_32, GEAR)
        // Tip-out condition
        T_DEC_TRA_1_AT = IP_T_DEC_TRA_1_AT_LOCK_CONV_CS (N_32, GEAR)
        T_DEC_TRA_2_AT = IP_T_DEC_TRA_2_AT_LOCK_CONV_CS (N_32, GEAR)
        T_DEC_TRA_3_AT = IP_T_DEC_TRA_3_AT_LOCK_CONV_CS (N_32, GEAR)
    Else
        // Tip-in condition
        T_INC_TRA_1_AT = IP_T_INC_TRA_1_AT (N_32, GEAR)
        T_INC_TRA_2_AT = IP_T_INC_TRA_2_AT (N_32, GEAR)
        T_INC_TRA_3_AT = IP_T_INC_TRA_3_AT (N_32, GEAR)
        // Tip-out condition
        T_DEC_TRA_1_AT = IP_T_DEC_TRA_1_AT (N_32, GEAR)
        T_DEC_TRA_2_AT = IP_T_DEC_TRA_2_AT (N_32, GEAR)
        T_DEC_TRA_3_AT = IP_T_DEC_TRA_3_AT (N_32, GEAR)
    EndIf
EndIf
```

TQ filtering for MT considering gear-shift and tip-in :

```
If      LC_ENA_TQ_FIL_GS_MT = 1
Then    // LV_TQ_TRA_FIL_MT_GS is determined at LV_CT transition to active
   If      LV_CT = 1 -> 0
   Then    // Gear shift and tip-in situation ( tip-in during gear-shifting )
           If      ( GEAR = 0 or LV_DT = 0 )               // Drive train (Opened)
           And     ( VS > C_VS_MIN_TQ_FIL_GS_MT )          // Running vehicle
           Then    LV_TQ_TRA_FIL_MT_GS = 1
           Else    // Other situation ( just tip-in at specific gear )
                   LV_TQ_TRA_FIL_MT_GS = 0
           Endif
   Endif

   If      LV_CT = 0 -> 1
   Then    LV_TQ_TRA_FIL_MT_GS = 0
   EndIf

   // As long as gear-shift and tip-in state is active
   If      LV_TQ_TRA_FIL_MT_GS = 1
   Then    T_INC_TRA_1_MT = IP_T_INC_TRA_1_MT_GS (N_32, GEAR)
           T_INC_TRA_2_MT = IP_T_INC_TRA_2_MT_GS (N_32, GEAR)
           T_INC_TRA_3_MT = IP_T_INC_TRA_3_MT_GS (N_32, GEAR)
   Else    T_INC_TRA_1_MT = IP_T_INC_TRA_1_MT (N_32, GEAR)
           T_INC_TRA_2_MT = IP_T_INC_TRA_2_MT (N_32, GEAR)
           T_INC_TRA_3_MT = IP_T_INC_TRA_3_MT (N_32, GEAR)
   EndIf
Else    T_INC_TRA_1_MT = IP_T_INC_TRA_1_MT (N_32, GEAR)
        T_INC_TRA_2_MT = IP_T_INC_TRA_2_MT (N_32, GEAR)
        T_INC_TRA_3_MT = IP_T_INC_TRA_3_MT (N_32, GEAR)
EndIf
```

The factor for manipulating T_TRA depending on driver request :

```
If      LV_AT = 0
Then    FAC_T_INC_TRA = IP_FAC_T_INC_TRA (PV_AV)
        FAC_T_DEC_TRA = IP_FAC_T_DEC_TRA (PV_AV)
Else    FAC_T_INC_TRA = IP_FAC_T_INC_TRA _AT(PV_AV)
        FAC_T_DEC_TRA = IP_FAC_T_DEC_TRA _AT(PV_AV)
```

Calculation of T_TRA

If (LV_IS == 0 or LV_CT == 0)			Then
& STATE_TQ_PHA ==	**& STATE_TQ_RANGE ==**	**& LV_AT ==**	**T_TRA =**
INC	1	0	T_INC_TRA_1_MT * FAC_T_INC_TRA
INC	2	0	T_INC_TRA_2_MT * FAC_T_INC_TRA
INC	3	0	T_INC_TRA_3_MT * FAC_T_INC_TRA
DEC	3	0	IP_T_DEC_TRA_3_MT * FAC_T_DEC_TRA
DEC	2	0	IP_T_DEC_TRA_2_MT * FAC_T_DEC_TRA
DEC	1	0	IP_T_DEC_TRA_1_MT * FAC_T_DEC_TRA
INC	1	1	(T_INC_TRA_1_AT+IP_T_INC_TRA_DELTA_N_TUR_ENG_1) * FAC_T_INC_TRA
INC	2	1	(T_INC_TRA_2_AT+IP_T_INC_TRA_DELTA_N_TUR_ENG_2) * FAC_T_INC_TRA
INC	3	1	(T_INC_TRA_3_AT+IP_T_INC_TRA_DELTA_N_TUR_ENG_3) * FAC_T_INC_TRA
DEC	3	1	T_DEC_TRA_3_AT * FAC_T_DEC_TRA
DEC	2	1	T_DEC_TRA_2_AT * FAC_T_DEC_TRA
DEC	1	1	T_DEC_TRA_1_AT * FAC_T_DEC_TRA
If (LV_IS == 1 and LV_CT = 1) (Exceptional condition for Idle Speed)			**Then**
& STATE_TQ_PHA ==	**& STATE_TQ_RANGE ==**	**& LV_AT ==**	**T_TRA =**
INC	1	0	IP_T_INC_TRA_IS_MT
DEC	1	0	IP_T_DEC_TRA_IS_MT
INC	1	1	IP_T_INC_TRA_IS_AT
DEC	1	1	IP_T_DEC_TRA_IS_AT
If (LV_IS == 1 and LV_CT = 1) (Exceptional condition for Idle Speed)			**Then**
& STATE_TQ_PHA ==	**& STATE_TQ_RANGE ==**	**& LV_AT ==**	**T_TRA =**
INC	2	0	IP_T_INC_TRA_2_MT * FAC_T_INC_TRA
DEC	2	0	IP_T_DEC_TRA_2_MT * FAC_T_DEC_TRA
INC	2	1	IP_T_INC_TRA_2_AT * FAC_T_INC_TRA
DEC	2	1	IP_T_DEC_TRA_2_AT * FAC_T_DEC_TRA
If (LV_IS == 1 and LV_CT = 1) (Exceptional condition for Idle Speed)			**Then**
& STATE_TQ_PHA ==	**& STATE_TQ_RANGE ==**	**& LV_AT ==**	**T_TRA =**
INC	3	0	IP_T_INC_TRA_3_MT * FAC_T_INC_TRA
DEC	3	0	IP_T_DEC_TRA_3_MT * FAC_T_DEC_TRA
INC	3	1	IP_T_INC_TRA_3_AT * FAC_T_INC_TRA
DEC	3	1	IP_T_DEC_TRA_3_AT * FAC_T_DEC_TRA

IF LV_DROF_AT_ACT = 1 # Exceptional case – AT drive-off

THEN

IF C_INH_TQ_TRA_DROF_AT = 1 **and** STATE_TQ_PHA = INC

or C_INH_TQ_TRA_DROF_AT = 2 **and** STATE_TQ_PHA = DEC

or C_INH_TQ_TRA_DROF_AT = 3 # INC and DEC disabled

THEN T_TRA = 0 # Torque Transient Disabled

ENDIF

ENDIF

TQ ranges for filtering:

This section of the specification should be called by a seperate function call from the torque transient function. It is necessary to determine the correct thresholds before the range determination can take place.

The thresholds for each ramp are determined from maps against engine speed and gear for each transmission variant.

TQ_RANGE_SEL_OFS = IP_TQ_RANGE_SEL_OFS(TQ_REQ_CLU)

```
If          LV_AT = 1              // AT vehicle
     TQ_TRA_THD_DOWN_1    = IP_TQ_TRA_THD_DOWN_1_AT (N_32, GEAR)
     TQ_TRA_THD_DOWN_2    = IP_TQ_TRA_THD_DOWN_2_AT (N_32, GEAR)
     TQ_TRA_THD_UP_1      = IP_TQ_TRA_THD_UP_1_AT (N_32, GEAR)
     TQ_TRA_THD_UP_2      = IP_TQ_TRA_THD_UP_2_AT (N_32, GEAR)
Else // MT vehicle
     TQ_TRA_THD_DOWN_1    = IP_TQ_TRA_THD_DOWN_1_MT (N_32, GEAR)
     TQ_TRA_THD_DOWN_2    = IP_TQ_TRA_THD_DOWN_2_MT (N_32, GEAR)
     TQ_TRA_THD_UP_1      = IP_TQ_TRA_THD_UP_1_MT (N_32, GEAR)
     TQ_TRA_THD_UP_2      = IP_TQ_TRA_THD_UP_2_MT (N_32, GEAR)
EndIf
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_T_INC_TRA_1_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 for manual transmission					
IP_T_INC_TRA_1_MT_GS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 while gear-shift for manual transmission					
IP_T_INC_TRA_2_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 2 for manual transmission					
IP_T_INC_TRA_2_MT_GS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 2 while gear-shift for manual transmission					
IP_T_INC_TRA_3_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 3 for manual transmission					
IP_T_INC_TRA_3_MT_GS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 3 while gear-shift for manual transmission					
IP_T_DEC_TRA_1_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 1 for manual transmission					
IP_T_DEC_TRA_2_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 2 for manual transmission					

IP_T_DEC_TRA_3_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 3 for manual transmission					
IP_T_INC_TRA_IS_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 at idle for manual transmission					
IP_T_DEC_TRA_IS_MT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 1 at idle for manual transmission					
IP_T_INC_TRA_1_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 for automatic transmission					
IP_T_INC_TRA_1_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 when converter clutch is fully locked up for automatic transmission					
IP_T_INC_TRA_2_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 2 for automatic transmission					
IP_T_INC_TRA_2_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 2 when converter clutch is fully locked up for automatic transmission					
IP_T_INC_TRA_3_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 3 for automatic transmission					
IP_T_INC_TRA_3_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 3 when converter clutch is fully locked up for automatic transmission					
IP_T_DEC_TRA_1_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 1 for automatic transmission					
IP_T_DEC_TRA_1_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 1 when converter clutch is fully locked up for automatic transmission					
IP_T_DEC_TRA_2_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 2 for automatic transmission					
IP_T_DEC_TRA_2_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 2 when converter clutch is fully locked up for automatic transmission					
IP_T_DEC_TRA_3_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 3 for automatic transmission					
IP_T_DEC_TRA_3_AT_LOCK_CONV_CS	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 3 when converter clutch is fully locked up for automatic transmission					

IP_T_DEC_TRA_IS_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque decrease 1 at idle for automatic transmission					
IP_T_INC_TRA_IS_AT	8x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_N_32_1_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 at idle for automatic transmission					
IP_TQ_RANGE_SEL_OFS	8x1	0...FFFFH	-1024...1023.97	0.03125	Nm
LDP_TQ_REQ_CLU_IP_TQ_RANGE_SEL	8	0...FFFFH	-1024...1023.97	0.03125	Nm
Offset for torque range calculation depending on driver torque request					
IP_FAC_T_INC_TRA	1x6	0...FFH	0...2.55	0.01	-
LDPM_PV_AV_FAC_T_INC_TRA	6	0...FFH	0...99.6094	0.3906	%
Factor for load dependent T_TRA for increasing torque request					
IP_FAC_T_DEC_TRA	1x6	0...FFH	0...2.55	0.01	-
LDPM_PV_AV_FAC_T_DEC_TRA	6	0...FFH	0...99.6094	0.3906	%
Factor for load dependent T_TRA for decreasing torque request					
IP_FAC_T_INC_TRA_AT	1x6	0...FFH	0...2.55	0.01	-
LDPM_PV_AV_FAC_T_INC_TRA	6	0...FFH	0...99.6094	0.3906	%
Factor for load dependent T_TRA for increasing torque request for AT					
IP_FAC_T_DEC_TRA_AT	1x6	0...FFH	0...2.55	0.01	-
LDPM_PV_AV_FAC_T_DEC_TRA	6	0...FFH	0...99.6094	0.3906	%
Factor for load dependent T_TRA for decreasing torque request for AT					
IP_T_INC_TRA_DELTA_N_TUR_ENG_1	12x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_DELTA_N_TUR_ENG_1_DRVB	12	0 .. FFFF H	-32768 .. 32767	1	Rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 1 considering DELTA_N_TUR_ENG for automatic transmission					
IP_T_INC_TRA_DELTA_N_TUR_ENG_2	12x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_DELTA_N_TUR_ENG_1_DRVB	12	0 .. FFFF H	-32768 .. 32767	1	Rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 2 considering DELTA_N_TUR_ENG for automatic transmission					
IP_T_INC_TRA_DELTA_N_TUR_ENG_3	12x8	0 .. FF H	0 .. 2.55	0.01	s
LDPM_DELTA_N_TUR_ENG_1_DRVB	12	0 .. FFFF H	-32768 .. 32767	1	Rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Time constant for torque increase 3 considering DELTA_N_TUR_ENG for automatic transmission					
C_VS_MIN_TQ_FIL_GS_MT	1	0 .. FF H	0 .. 255	1	Km/h
Minimum threshold of vehicle speed to activate TQ transient filtering time while gear-shifting for MT					
C_N_DIF_LOCK_CONV_CS	1	0 .. FFFF H	0 .. 65535	1	rpm
Minimum difference between N and N_TUR_CONV_RAW to detect that converter clutch is fully locked up for automatic transmission					
LC_TQ_FIL_OFF_SPK_RTD_TCU	1	0 .. 1 H	0 .. 1	1	-
Switch to disable TQ transient filtering time calculation in case of TQ reduction request from TCU for AT					
LC_ENA_TQ_FIL_GS_MT	1	0 .. 1 H	0 .. 1	1	-
Switch to enable TQ transient filtering while gear-shift for MT					
IP_TQ_TRA_THD_DOWN_1_AT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 1 for decreasing torque AT					
IP_TQ_TRA_THD_DOWN_2_AT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 2 for decreasing torque AT					
IP_TQ_TRA_THD_UP_1_AT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 1 for increasing torque AT					
IP_TQ_TRA_THD_UP_2_AT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 2 for increasing torque AT					

IP_TQ_TRA_THD_DOWN_1_MT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 1 for decreasing torque MT					
IP_TQ_TRA_THD_DOWN_2_MT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 2 for decreasing torque MT					
IP_TQ_TRA_THD_UP_1_MT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 1 for increasing torque MT					
IP_TQ_TRA_THD_UP_2_MT	4x8	0000...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_3_DRVB	4	0...FFH	0 .. 8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FFH	0...255	1	-
Torque threshold 2 for increasing torque MT					
C_INH_TQ_TRA_DROF_AT	1	0...3H	0...3	1	-
Constant to inhibit torque transient function if drive-off function active					

D.35 Torque Transient

General Information

Due to it's physical properties the car can be considered as a two-mass-spring-system, producing oscillations on rapid torque increas. To prevent the powertrain from jerks the requested torque from driver or cruise control TQ_REQ_CLU is filtered. The torque increase or decrease is realized by torque ramps. The output value is TQ_REQ_TRA_CLU (torque at clutch) and the corresponding TQI_REQ_TRA (indicated torque).

For a more spontaneous reaction of the vehicle and to ensure that the actual torque (TQI_AV) follows best the torque setpoint (TQI_REQ_TRA) a torque reserve TQ_ADD_TRA can be tuned. This torque reserve value is added on the air path whereas a fast spark retard (if it is necessary) ensure that the actual torque and the setpoint are converging.

A second functionality is to damp uncomfortable behavior of the car because of the load cycle at zero crossing torque. In that torque range a ramp function is applied that reduces engine movement between its bearings.

See also the module Torque reserve for Torque Transient for additional functionality.

D.35.1 Generic Activation Conditions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_REQ_TRA_CLU	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Torque request at clutch after transient torque function					
TQ_REQ_DIF_TRA	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Deviation of TQ_REQ_CLU and TQ_REQ_TRA_CLU limited to “0” Nm					
TQ_REQ_TRA_OLD_CLU	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Old torque value at steady state conditions					
TQI_REQ_TRA	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Indicated torque request after transient torque function					
LV_TQ_TRA	V	0...1 H	0 ... 1	1	-
Logical variable transient torque function acitve					
LV_TQ_TRA_DYN	V / O	0...1 H	0 ... 1	1	-
Logical variable spark retard requested due to driver dynamics					
STATE_TQ_PHA	V / O	0 H 1 H 2 H	TQ_PHA_CONST TQ_PHA_INC TQ_PHA_DEC	1	-
State of torque transient phase					
STATE_TQ_RANGE	V / O	1 H 2 H 3 H	TQ_RANGE_1 TQ_RANGE_2 TQ_RANGE_3	1	-
Range of TQ_REQ_TRA_CLU					

Input data:

TQ_REQ_CLU	TQ_LOSS	N_GRD	T_TRA
TQ_RANGE_SEL_OFS	TQ_TRA_THD_UP_1	TQ_TRA_THD_UP_2	TQ_TRA_THD_DOWN_1
TQ_TRA_THD_DOWN_2			

Application conditions:

Initialisation:	at reset
	TQ_REQ_TRA_CLU = 0
	TQ_REQ_TRA_OLD_CLU = 0
Recurrence:	10 ms
Activation:	at every engine state
Deactivation:	-

Description:

The torque transient filter is activated only, if the torque request represented by TQ_REQ_CLU is not constant, in the states "INC" or "DEC". In the other case the output TQ_REQ_TRA_CLU is equal to the input TQ_REQ_CLU (state "CONST").

Formula section:

```
If      (TQ_REQ_CLU >= TQ_REQ_TRA_CLU + C_TQ_TRA_HYS) OR
        (TQ_REQ_CLU <= TQ_REQ_TRA_CLU – C_TQ_TRA_HYS) AND NOT
        (LC_TQ_TRA_INH)
then    LV_TQ_TRA = 1
        If    TQ_REQ_CLU > TQ_REQ_TRA_CLU
        then  STATE_TQ_PHA = 1           %"INC"
        else  STATE_TQ_PHA = 2           %"DEC"
        endif
else    STATE_TQ_PHA = 0                 % "CONST"
endif
```

From torque increase to steady state conditions:

```
if      TQ_REQ_TRA_CLU >= TQ_REQ_CLU
then    STATE_TQ_PHA = 0                 % "CONST"
endif
```

From torque decrease to steady state conditions:

```
if      TQ_REQ_TRA_CLU <= TQ_REQ_CLU
then    STATE_TQ_PHA = 0                 % "CONST"
endif
if      STATE_TQ_PHA == 0                % "CONST"
then    LV_TQ_TRA = 0
        TQ_REQ_TRA_CLU = TQ_REQ_CLU
        TQ_REQ_TRA_OLD_CLU = TQ_REQ_CLU
endif
```

D.35.2 TQ_TRA_THD aquisition

According to the calculation sequence here the algorithm from the module "Torque transient parameters", where the torque thresholds are calculated is called here as a function call. This MUST take place in one cycle.

TQ_TRA_THD_UP/DOWN_1 / 2 = FUNCTION CALL ("Torque Transient Parameters")

After the threshold are calculated the function returns to the next chapter.

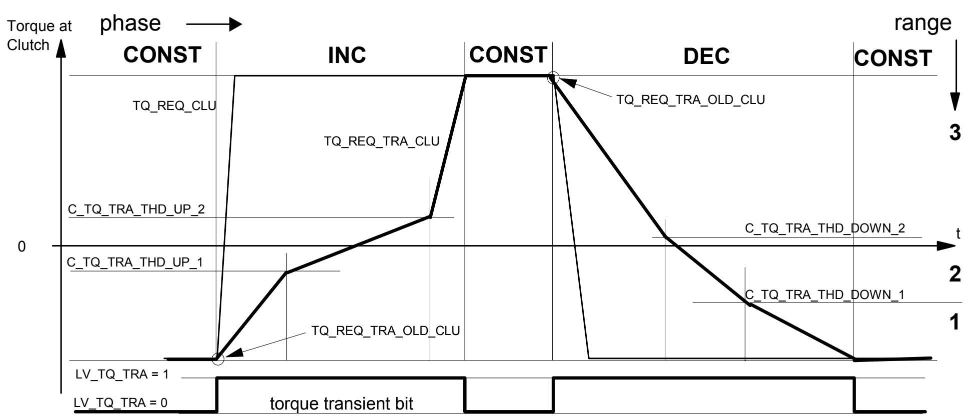
D.35.3 Phase-range detection

Description:

Since there are different requirements for the torque transient ramps depending on the range of the torque request and also on its direction it is necessary to detect these states.

The torque ranges are choosen depending on calibrateable thresholds (determined in torque transient parameter function).

The selected values are necessary to obtain a time constant information from the function module "Torque Transient parameter" and obtained via a second function call to the torque transient parameter function.



Formula section:

Criterion for range selection:

a) Torque increase

If STATE_TQ_PHA == 1 %"INC"

Then

If the N_GRD- condition was true for the last cycle, the range detection is not applied until TQ_REQ_TRA_CLU exceeds TQ_TRA_THD_UP_2 + TQ_RANGE_SEL_OFS.

STATE_TQ_RANGE =	Condition
1	TQ_REQ_TRA_CLU < TQ_TRA_THD_UP_1 + TQ_RANGE_SEL_OFS
2	TQ_TRA_THD_UP_1 + TQ_RANGE_SEL_OFS <= TQ_REQ_TRA_CLU < TQ_TRA_THD_UP_2 + TQ_RANGE_SEL_OFS
3	TQ_REQ_TRA_CLU >= TQ_TRA_THD_UP_2 + TQ_RANGE_SEL_OFS

b) Torque decrease

Elseif STATE_TQ_PHA == 2 %"DEC"

Then

If the N_GRD- condition was true for the last cycle, the range detection is not applied until TQ_REQ_TRA_CLU is below TQ_TRA_THD_DOWN_1 + TQ_RANGE_SEL_OFS.

STATE_TQ_RANGE =	Condition
1	TQ_REQ_TRA_CLU < TQ_TRA_THD_DOWN_1 + TQ_RANGE_SEL_OFS
2	TQ_TRA_THD_DOWN_1 + TQ_RANGE_SEL_OFS <= TQ_REQ_TRA_CLU < TQ_TRA_THD_DOWN_2 + TQ_RANGE_SEL_OFS
3	TQ_REQ_TRA_CLU >= TQ_TRA_THD_DOWN_2 + TQ_RANGE_SEL_OFS

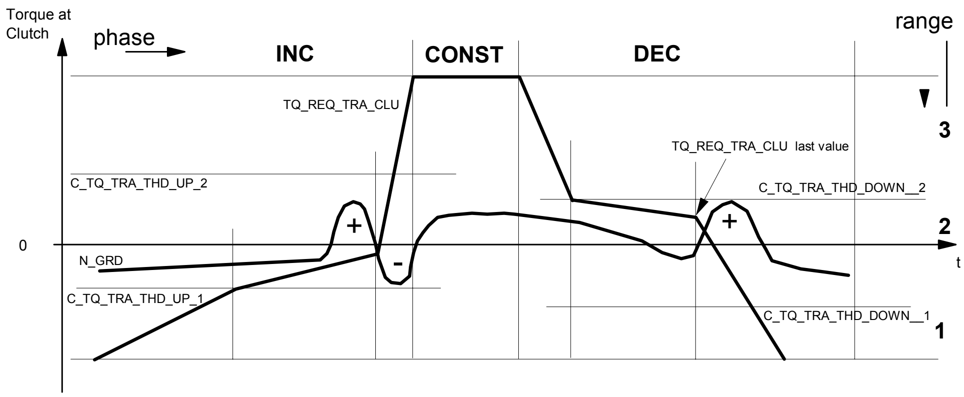
End

Under normal conditions no direct jump happens from range 1 to range 3 and vice versa, every range is passed through completely.

Special conditions are N_GRD-condition and T_TRA - condition.

Exceptional N_GRD-condition:

In range 2 only a slow ramp is performed. This range can be left immediatly, when the engine has done its movement from one stopper to the other due to load cycle, detected by the engine speed gradient.



If the N_GRD condition comes true , range 2 is left and the stored TQ_REQ_TRA_OLD_CLU is updated.

If STATE_TQ_RANGE == 2 **AND** STATE_TQ_PHA == 1
AND (N_GRD < C_N_GRD_THD_NEG)
then TQ_REQ_TRA_OLD_CLU = TQ_REQ_TRA_CLU (only once)
If TQ_REQ_CLU > TQ_TRA_THD_UP_2 + TQ_RANGE_SEL_OFS
then STATE_TQ_RANGE =3
else STATE_TQ_RANGE=2
endif
endif

If STATE_TQ_RANGE == 2 **AND** STATE_TQ_PHA == 2
AND (N_GRD > C_N_GRD_THD_POS)
then TQ_REQ_TRA_OLD_CLU = TQ_REQ_TRA_CLU (only once)
If TQ_REQ_CLU < TQ_TRA_THD_DOWN_1 + TQ_RANGE_SEL_OFS
then STATE_TQ_RANGE =1
else STATE_TQ_RANGE =2
endif
endif

If the N_GRD condition was not true in range 2, the stored value TQ_REQ_TRA_OLD_CLU is initialized at leaving range 2 and entry into a new range. TQ_REQ_TRA_OLD_CLU is set to the last filtered torque request, which contains the upper range limit incl. offset for increasing torque and the lower range limit incl. offset for decreasing torque request.

```
If      N_GRD condition was not true AND
        ((STATE_TQ_RANGE == 3 AND STATE_TQ_PHA == 1) or
        (STATE_TQ_RANGE == 1 AND STATE_TQ_PHA ==2))
Then    TQ_REQ_TRA_OLD_CLU = TQ_REQ_TRA_CLU ( only once )
End
```

If transient conditions are true and direct change from torque increase to decrease and vice versa happens then TQ_REQ_TRA_OLD_CLU = TQ_REQ_TRA_CLU and a new selection of range is done.

```
If      STATE_TQ_RANGE == 1 or STATE_TQ_RANGE == 3
Then    TQ_REQ_2 = TQ_REQ_CLU
        TQ_REQ_1 = TQ_REQ_TRA_OLD_CLU
Else    if      STATE_TQ_RANGE == 2 and STATE_TQ_PHA == 1  %"INC"
        Then    TQ_REQ_2 = TQ_TRA_THD_UP_2 + TQ_RANGE_SEL_OFS
                TQ_REQ_1 = TQ_TRA_THD_UP_1 + TQ_RANGE_SEL_OFS
        Elseif  STATE_TQ_RANGE == 2 and STATE_TQ_PHA == 2  %"DEC"
        Then    TQ_REQ_2 = TQ_TRA_THD_DOWN_1 + TQ_RANGE_SEL_OFS
                TQ_REQ_1 = TQ_TRA_THD_DOWN_2 + TQ_RANGE_SEL_OFS
        End
End
```

D.35.4 T_TRA aquisition

According to the calculation sequence here the algorithm from the module "Torque transient parameters" where the T_TRA time is calculated as a function call. It is important, that the shift of the states STATE_TQ_RANGE and STATE_TQ_PHA from phase-range-detection takes place at the same cycle as obtaining the result and handover to the next chapter. This MUST take place in one cycle.

T_TRA = FUNCTION CALL ("Torque Transient Parameters")

Where "Torque Transient parameters" consumes from here

(STATE_TQ_RANGE, STATE_TQ_PHA).

After T_TRA is calculated the function returns to the next chapter.

D.35.5 Generic torque transient filter

Description:

The torque transient filter calculates a filtered torque request. If the requested torque is faster than a certain ramp time it is filtered that the setpoint is reached within a certain time depending on phase, range and gear (and exceptional conditions).

Application conditions:

Initialisation:	at reset
	TQ_REQ_TRA_CLU = 0
	TQ_DIF_TRA = 0
	TQI_REQ_TRA = 0
Recurrence:	*10ms*
Activation:	STATE_TQ_PHA <> 0
Deactivation:	STATE_TQ_PHA == 0

Formula section:

If T_TRA == 0

then exit to STATE_TQ_PHA = 0

else TQ_DIF_TRA = ((TQ_REQ_2 - TQ_REQ_1) / T_TRA) * recurrence (10ms)

endif

Following integrator is calculated time synchonous each update rate:

TQ_REQ_TRA_CLU(n) = TQ_REQ_TRA_CLU(n-1) + TQ_DIF_TRA

Calculated after State machine:

Formula section:

Indicated requested transient torque:

TQI_REQ_TRA = **MAX**(0,(TQ_REQ_TRA_CLU - TQ_LOSS))

Deviation of unfiltered and filtered torque request at clutch:

TQ_REQ_DIF_TRA is used for indicating a torque reserve request.

TQ_REQ_DIF_TRA = TQ_REQ_CLU – TQ_REQ_TRA_CLU

Calculation of LV_TQ_TRA_DYN:

LV_TQ_TRA_DYN is used for information when high dynamic torque is requested.

If STATE_TQ_RANGE = 1

Then LV_TQ_TRA_DYN = 1

Else LV_TQ_TRA_DYN = 0

Endif

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_N_GRD_THD_NEG	1x1	80...7F H	-4096...4064	32	rpm/sec
Engine speed gradient for torque increase					
C_N_GRD_THD_POS	1x1	80...7F H	-4096...4064	32	rpm/sec
Engine speed gradient for torque decrease					
LC_TQ_TRA_INH	1x1	0...1 H	0...1	1	-
Logical parameter inhibition torque transient (0 ... torque transient active, 1 ... torque transient passive)					
C_TQ_TRA_HYS	1x1	0...FF H	0...7.96875	0.03125	Nm
Hysterisis for detection of torque transient active					

D.36 Torque Reserve for Torque Transient

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_TRA	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Torque reserve for transient trorque					
TQ_ADD_TRA_1	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Torque reserve output from map IP_TQ_ADD_TRA_MIN for transient torque					
TQ_ADD_TRA_2	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Torque reserve output from map IP_TQ_ADD_TRA_MAX for transient torque					
FAC_TQ_TRA	V	0...FFFFH	0 ... 32	0.0004883	-
Weighting factor for TQ_REQ_DIF_TRA					

Input data:

N_32	TQ_REQ_DIF_TRA		

FUNCTION DESCRIPTION:

At transient conditions the torque reserve TQ_ADD_TRA builds up a certain load potential that can be used to accelerate the slow airpath without violating the torque transient ramp. If a fast torque increase is needed for following the positive torque ramp setpoint, spark advance can be done very fast. TQ_ADD_TRA is obtained from a by FAC_TQ_REQ_DIF_TRA weighted difference. This value must be limited in both directions from the maps IP_TQ_ADD_TRA_xxx.

If ignition intervention is allowed this function can be used to accelerate the slow path with reduced ignition intervention for torque decrease.

Application conditions:

Initialisation: at reset

TQ_ADD_TRA = 0 Nm

TQ_ADD_TRA_1 = 0 Nm

TQ_ADD_TRA_2 = 0 Nm

FAC_TQ_TRA = 0

Recurrence: *10 ms*

Activation: *at every engine state*

Deactivation: -

Signal flow diagram:

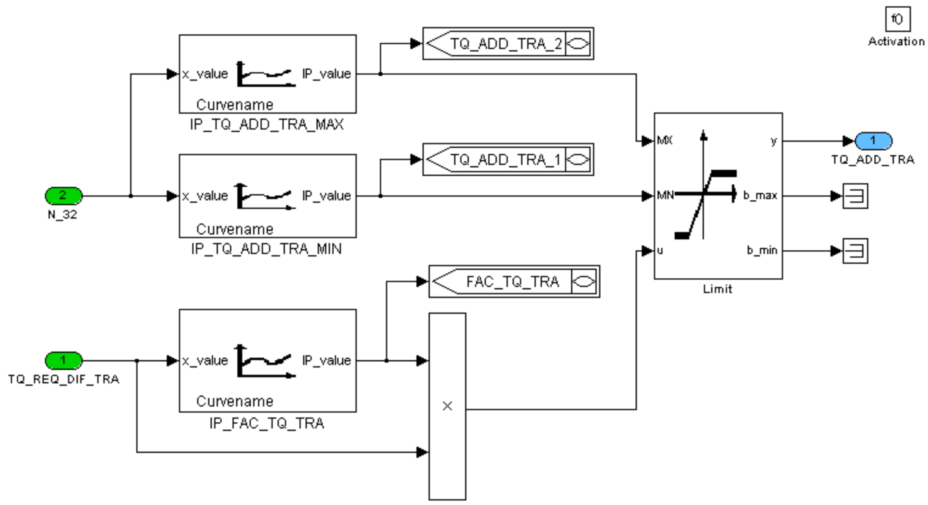


Figure 1: *Determination of the torque reserve for torque transient*

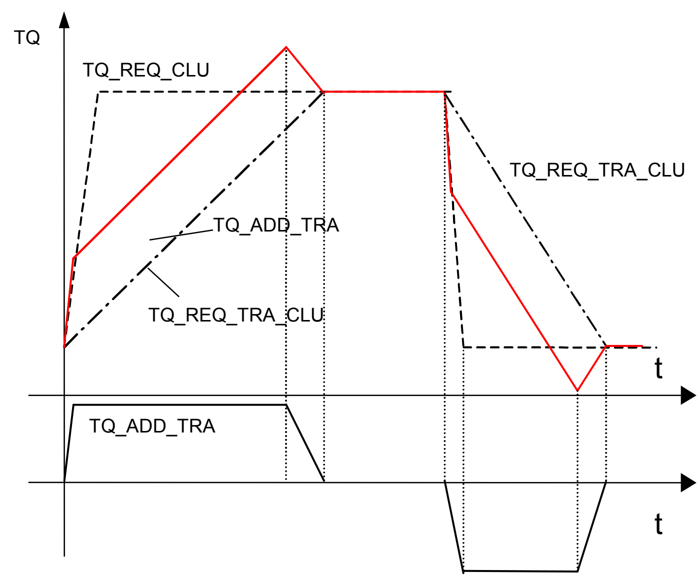


Figure 2: *Torque reserve for torque transient*

Calibration data:

Name	Dim.	Hex. limits	Phys. limits	Resol.	Unit
IP_TQ_ADD_TRA_MIN	1x8	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
LDPM_N_32_2_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
Torque reserve at increasing torque transient request					
IP_TQ_ADD_TRA_MAX	1x8	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
LDPM_N_32_2_DRVB	8	0 .. FF H	0 .. 8160	32	1/min
Torque reserve at decreasing torque transient request					
IP_FAC_TQ_TRA	1x6	0...FFFFH	0 ... 32	0.0004883	-
LDP_TQ_REQ_DIF_TRA_IP_FAC_TQ	6	0 .. FFFF H	-1024 .. 1023.97	0.03125	Nm
Torque reserve at transient torque conditions					

D.37 Torque Reserve Coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_ADD_ACT	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Indicated engine-torque request inclusive torque reserve					
TQI_ADD_ACT_TMP	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Indicated engine-torque request inclusive torque reserve - for catalyst-heating torque-reserve request					
TQI_ADD_MAX	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Maximum possible indicated engine torque inclusive torque reserve at IGA_MIN and ISC reserve limitation					
TQI_ADD_MAX_TOL	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Maximum possible indicated engine torque inclusive torque reserve at IGA_MIN					
TQI_ADD_MAX_TEG	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Maximum possible indicated engine torque inclusive torque reserve at IGA_MIN_TEG					
TQ_ADD_ACT	V / O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Difference between absolute values ot TQI_ADD_ACT and TQ_LOSS					
TQ_ADD_REQ_ACT	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Actual realized torque-reserve-request for slow torque coordination input					
TQ_ADD_REQ_CH	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Requested torque reserve for catalyt-heating					
TQI_REQ_TRA_LIM_L	V	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Limited TQI_REQ_TRA for the determination of the maximum allowed torque reserve					
TQ_ADD_REQ	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
General torque reserve request					
TQ_ADD_MAX	V/O	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Maximum possible torque reserve					
MAF_KGH_ADD_MAX	V/O	0 ... FFFF H	0 ... 2047.9678	0.03125	kg/h
Maximum possible engine load for catalyst heating - kg/h					
MAF_ADD_MAX	V/O	0 ... FFFF H	0 ... 1389	0.0212	mg/Stk
Maximum possible engine load for catalyst heating - mg/Stk					
EFF_IGA_MIN_ADD_MAX	V/O	0 ... FFFF H	0 .. 1.999969	30.517E-6	-
Minimum possible ignition efficiency for catalyst heating					
LV_TQ_ADD_REQ	V / O	0 , 1 H	0 , 1	1	-
Logical variable torque reserve request can be applied with all torque reserves					
LV_TQ_ADD_REQ_1	V	0 , 1 H	0 , 1	1	-
Logical variable torque reserve request can be applied without CH and SO2P					
TQ_ADD_TRA_TMP	-	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Coordinated torque transient torque reserve request					

Input data:

TQI_REQ_TRA	EFF_IGA_BAS_COR	EFF_IGA_MIN	EFF_IGA_MIN_TEG
TQ_ADD_SO2P	TQ_ADD_CH	TQ_ADD_TRA	TQ_ADD_CP
TQI_MIN_PU	TQ_ADD_EXT	TQ_ADD_IS_BOL	LV_HOM_RUN
TQI_BAS_MAX	TQ_LOSS	TQI_MAX	N
IP_MAF_SP	N_32	TQ_DIF_ADD_IS_TOL	NC_MAF_FAC_CYL
LV_IS	TQ_ADD_HEAT_ACC		

FUNCTION DESCRIPTION:

General information:

The torque reserve is a torque neutral increase of the cylinder charge by adding TQ_ADD_REQ or TQ_ADD_REQ_CH to the driver torque request TQI_REQ_TRA.

general specification

The torque reserve coordination delivers the torque value TQI_ADD_ACT.

TQI_ADD_ACT is the input for “Minimum/Maximum Torque Request Selection”. Without external torque intervention TQI_ADD_ACT will be transformed to TQI_REQ_SLOW. This torque request will than be used at “Torque Coordination” to determine the necessary cylinder charge torque setpoint TQI_SP_SLOW, which leads to the needed air mass flow setpoint MAF_SP_TQI.

The logical output-variable LV_TQ_ADD_REQ indicates if the requested torque reserve is applied or not.

The indicated engine torque request inclusive torque reserve TQI_ADD_ACT is always limited to the maximum possible value TQI_ADD_MAX. The limitation is necessary due to the fact that the increase of cylinder-charge must be compensated by spark retardation. The spark-retardation itself is only possible until the minimum ignition angle is reached.

For catalyst heating or for NO_X-trap desulfation the determination of TQI_ADD_MAX is done with IGA_MIN (EFF_IGA_MIN) which is independent from the actual exhaust gas temperature.

All other torque reserve-requests are limited, if high exhaust gas temperature can damage the catalytic converter or the outlet-valve. For this purpose the limitation TQI_ADD_MAX_TEG is determined as a function of IGA_MIN_TEG (EFF_IGA_MIN_TEG). The torque reserve for torque transient is treated separately, since it could become negative for reasons of air path acceleration.

In case of LV_HOM_RUN = 0 the input value TQI_REQ_TRA will directly copied to the output variable TQI_ADD_ACT.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: at reset

TQI_ADD_ACT = 0 Nm

TQ_ADD_ACT = 0 Nm

LV_TQ_ADD_REQ = 0

Update rate: 10 ms

Remark:

All TQI values are limited during calculation and output to minimum “0” Nm !

Overview:

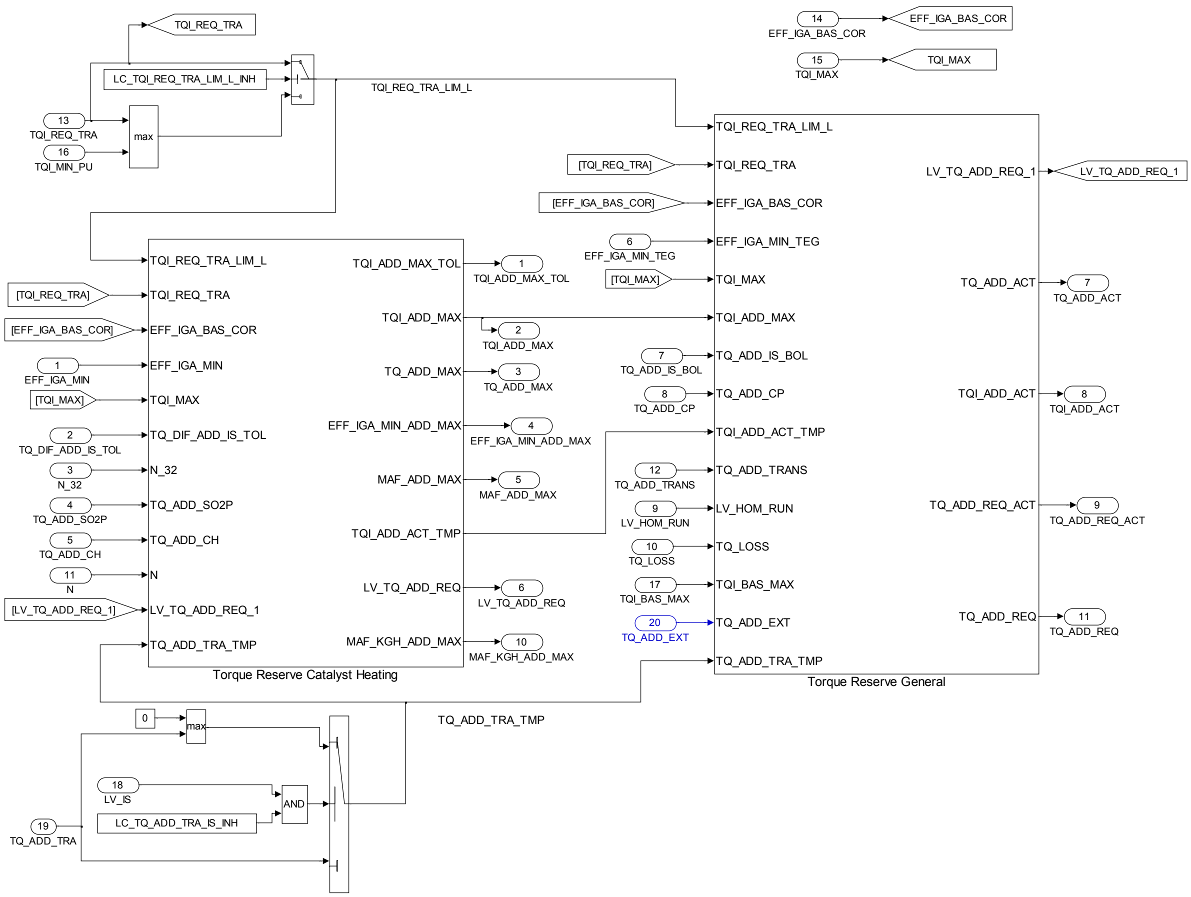


Figure 1: *Torque reserve coordination*

The torque reserve coordination is separated into the two functional blocks “Torque Reserve Catalyst Heating” and “Torque Reserve General”.

Both blocks are calculated during the 10 ms update rate.

D.37.1 Torque reserve catalyst heating

Overview:

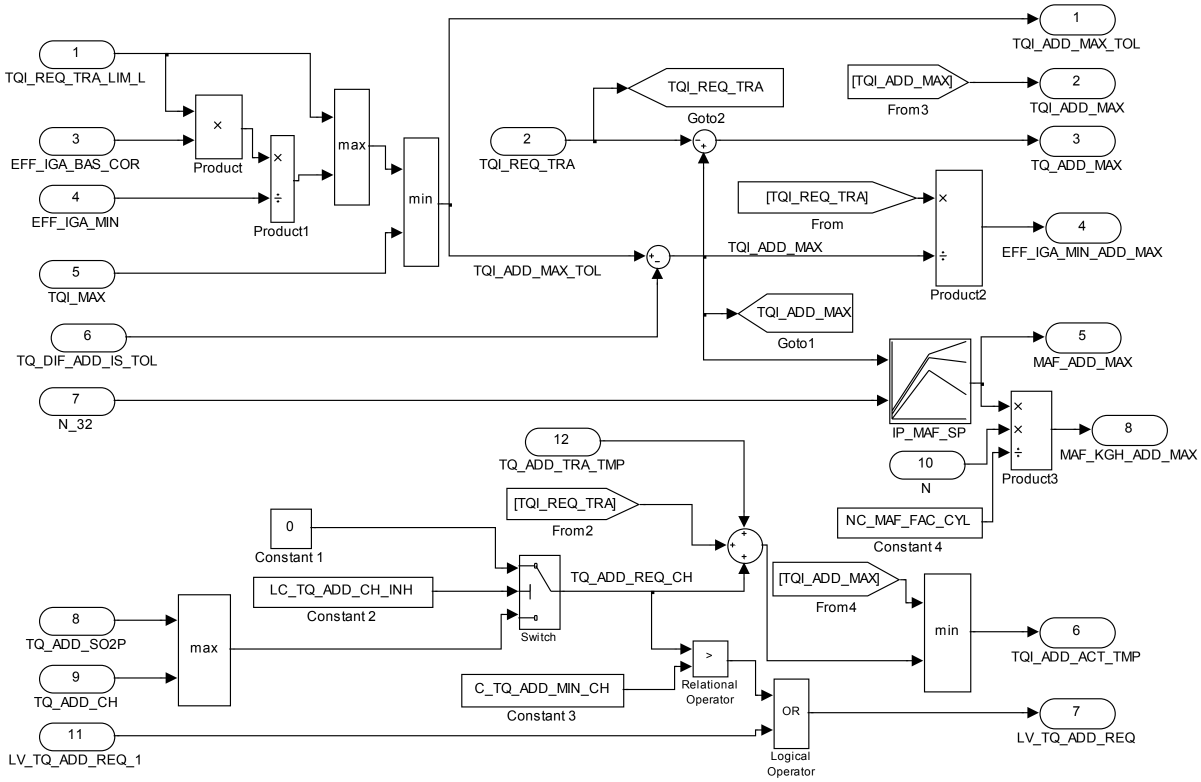


Figure 2: *Torque reserve catalyst heating*

During catalyst heating or NO_X-trap desulfation the limitation TQI_ADD_MAX is determined on basis of EFF_IGA_MIN, which is independant of the exhaust gas temperature. For the calculation of the maximum allowed torque reserve, a value TQI_REQ_TRA_LIM_L is used (see figure 1), which is TQI_MIN_PU at its minimum. Therewith, TQI_ADD_MAX can never reach zero.

For ISC request the torque-reserve TQ_DIF_ADD_IS_TOL must be taken into consideration.

The value TQ_ADD_MAX is determined as

TQ_ADD_MAX = TQI_ADD_MAX – TQI_REQ_TRA,

which is the maximum possible torque reserve.

With help of EFF_IGA_MIN_ADD_MAX and MAF_KGH_ADD_MAX (MAF_ADD_MAX) the NO_X-trap temperature model can estimate the potential of cat-heating energy which is actually available.

TQI_ADD_MAX and TQI_ADD_ACT_TMP will be used at “Torque Reserve General” to determine TQI_ADD_ACT.

Formula section:

MAF_KGH_ADD_MAX = MAF_ADD_MAX * N / NC_MAF_FAC_CYL

D.37.2 Torque Reserve General

Overview:

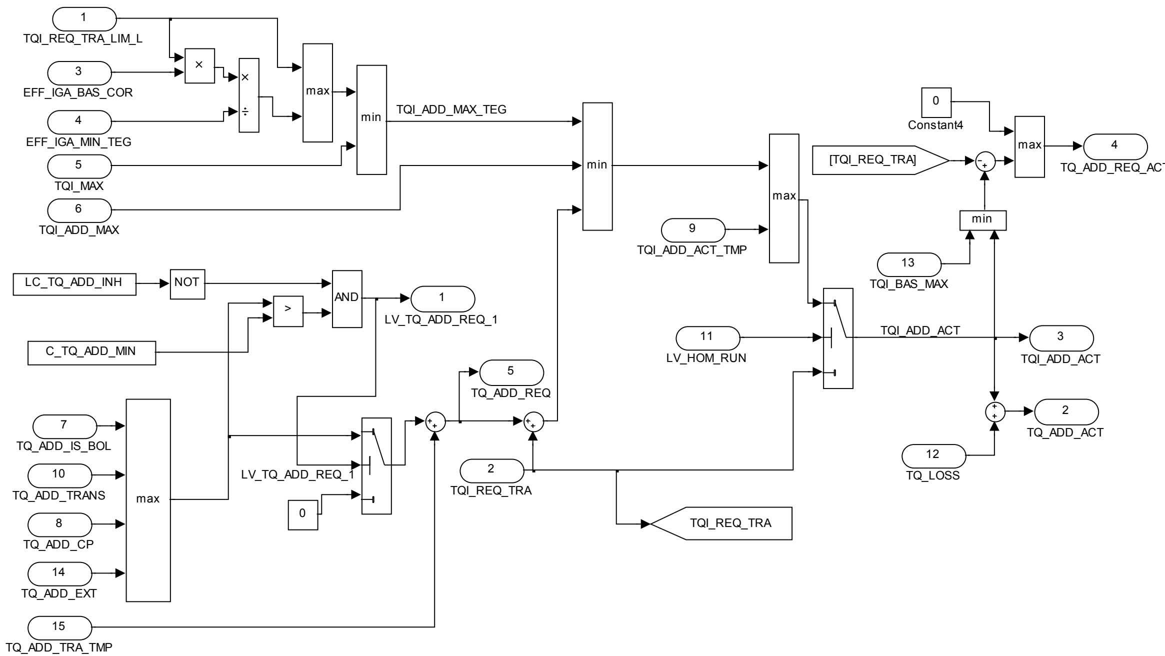


Figure 3: *Torque Reserve General*

For "General Torque Reserve Coordination" the limitation TQI_ADD_MAX_TEG is determined on basis of the EFF_IGA_MIN_TEG value which takes into consideration the actual exhaust gas temperature. For the calculation of the maximum allowed torque reserve, a value TQI_REQ_TRA_LIM_L is used (see figure 1). TQI_REQ_TRA_LIM_L is at its minimum TQI_MIN_PU, and so the maximum allowed torque reserve always unequal zero.

There must be a second limitation upto TQI_ADD_MAX because this value also includes the ISC request TQ_DIF_ADD_IS_TOL.

To prevent limitations of the catalyst heating torque reserve request, TQI_ADD_ACT_TMP is considered at the end of the torque reserve determination chain, dirctly before the definition of TQI_ADD_ACT.

At stratified combustion (LV_HOM_RUN=0), TQI_REQ_TRA will directly be copied to TQI_ADD_ACT.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TQ_ADD_CH_INH	1	0 ... 1 H	0 ...1	1	-
Logical constant for deactivation torque-reserve request for catalyst heating					
LC_TQ_ADD_INH	1	0 ... 1 H	0 ... 1	1	-
Logical constant for deactivation torque reserve request without CH and SO2P					
LC_TQI_REQ_TRA_LIM_L_INH	1	0 ... 1 H	0 ... 1	1	-
Bit for inhibiting the limitation of TQI_REQ_TRA for the calculation of the maximum allowed torque reserve					
C_TQ_ADD_MIN_CH	1	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Minimum threshold for authorizing torque reserve request for catalyst heating					
C_TQ_ADD_MIN	1	8000 .. 7FFF H	-1024 .. 1023.97	0.03125	Nm
Minimum threshold for authorizing torque reserve request without CH and SO2P					
LC_TQ_ADD_TRA_IS_INH	1	0 ... 1 H	0 ... 1	1	-
Logical constant for inhibiting the negative torque reserve request of torque transient in idle					

D.38 Torque reserve coordination (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_ADD_EXT	O/V	8000...7FFFH	-1.024E+3 ... 1.02397E+3	0.03125	Nm
Externally coordinated Torque reserve request					

Input data:

TQ_ADD_ACC	TQ_ADD_PSTE	TQ_ADD_TRANS	TQ_ADD_ALTER
TQ_ADD_PL	TQ_ADD_PSTE_PSPT		

D.38.1 FUNCTIONAL DESCRIPTION:

D.38.2 General information:

This module is used to adapt project specific torque reserve requests to the interfaces of the module 'Torque reserve coordination'.

The specific torque reserve requests are max-selected for further coordination in 'Torque reserve coordination'. They are expected to be positive and just to contain the torque reserve request only without driver request or similar. For this realization only the torque reserve requests considering a maximum torque reserve limited by exhaust gas temperature.

Since no other request exists for high exhaust gas energy, the related output value is initialized to zero.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: at reset

TQ_ADD_EXT = 0 Nm

Update rate: 10 ms

Formula section:

TQ_ADD_EXT = max (0,
TQ_ADD_ALTER,
TQ_ADD_TRANS,
TQ_ADD_ACC,
TQ_ADD_PSTE,
TQ_ADD_PSTE_PSPT,
TQ_ADD_PL)

D.39 Minimum Indicated Engine Torque Offset during PUC (Binary Lambda Sensor)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_ADD_PUC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Total additive correction indicated engine torque at trailing throttle conditions for PUC					
T_MAP_PUC_LIM	O/V	0...FFFFH	0...655.35	0.01	s
Time elapsed in PUC					
TQI_ADD_PUC_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Additive correction indicated engine torque at trailing throttle conditions for PUC					

Input data:

LV_MAP_PUC_LIM_ACT_LSL_GAIN_AD	LV_PUC	N_32	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_ADD_PUC	12	0...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_N_32_2_TQDR	12	0...FFH	0...8160	32	1/min
Minimum load limitation on indicated torque level in PUC phase					

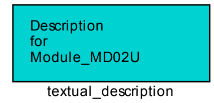
D.39.1 General Information

The output TQI_ADD_PUC is utilized as offset to the basic minimum indicated torque limitation in the module ‘Minimum indicated engine torque at trailing throttle’. The offset is active during fuel cut-off only and ensures a feasible manifold absolute pressure.

In a system configuration with binary lambda sensor, the LSL gain adapation request LV_MAP_PUC_LIM_ACT_LSL_GAIN_AD is delivered from a stub module (interface consistency). It is permanently set to zero and thus, has no functional influence.

The timer is functionally relevant only for the configuration with a linear lambda sensor. With a binary lambda sensor, the timer is not being used but has to be defined for interface consistency.

Function Description



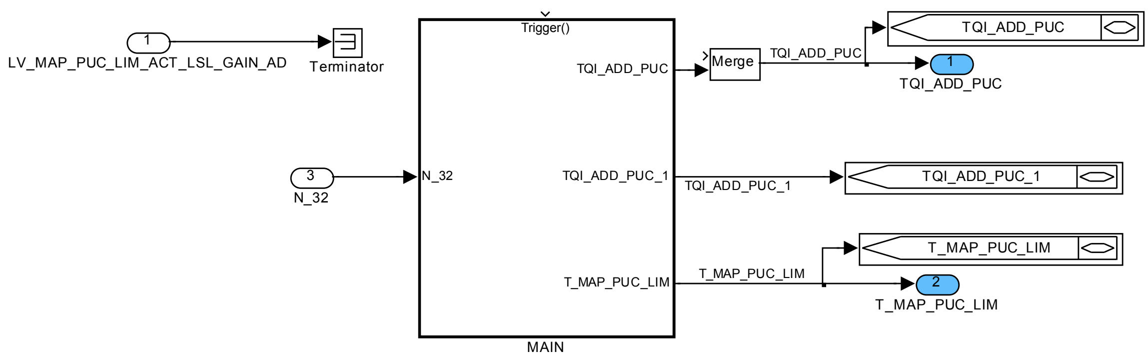


Figure 52 MD02U

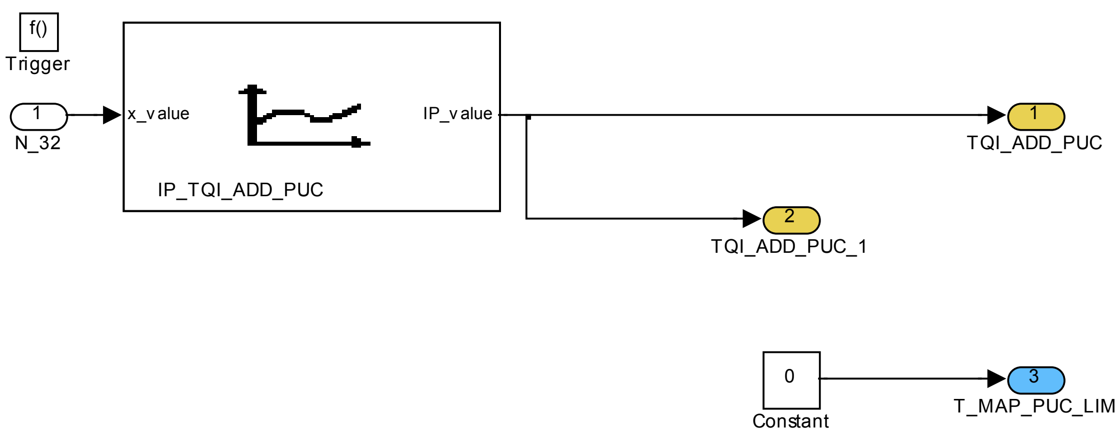


Figure 53 MD02U/ MAIN

D.40 Minimum indicated engine torque offset during PUC (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_MAP_PUC_LIM_ACT_LSL_GAIN_AD	O/V	0...1H	0...1	1	-
Logical variable is set if MAP limitation is activated					

Input data:

LV_PUC			

D.40.1 General Information

In this module, the parameter LV_MAP_PUC_LIM_ACT_LSL_GAIN_AD is stubbed for the module ‘Minimum indicated engine torque offset during PUC’.

Function Description

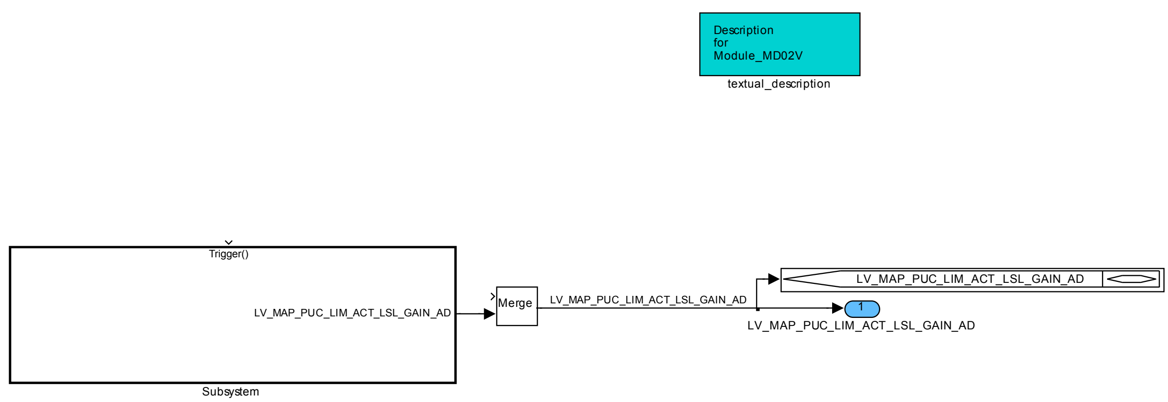


Figure 54 MD02V

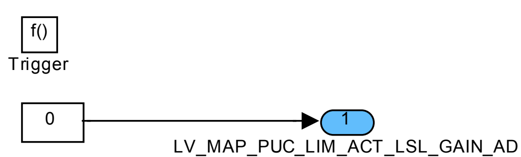


Figure 55 MD02V/ Subsystem

D.41 Minimum Indicated Engine Torque at Trailing Throttle

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TQI_BOL_MAF_SET	O/V	0...1H	0...1	1	-
Torque request is below TQI_BOL_MAF					
LV_TQI_BOL_SET	O/V	0...1H	0...1	1	-
Logical variable it is set if TQI_SP falls below TQI_BOL					
TQI_MIN_PU	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque minimum value at trailing throttle conditions for PU or PUC					
LV_TQI_BOL_IGA_SET	V	0...1H	0...1	1	-
Torque request is below TQI_BOL_IGA					
MOD_TQI_BOL_SET_HOM	V	0...2H	0...2	1	-
Calculation mode for LV_TQI_BOL_SET calculation during homogeneous combustion					
TQI_BOL_IGA	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque bottom level at minimum ignition efficiency for engine run free of misfire					
TQI_BOL_MAF	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine reference torque bottom level at minimum load for engine run free of misfire					
TQI_MIN_PU_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque minimum value at trailing throttle conditions for PU (standard conditions)					
TQI_MIN_PU_OFS_AMP	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
AMP correction offset in TQI_MIN_PU					

Input data:

EFF_IGA_MIN	TQI_REF	TQI_REF_LAMB_SCC	LV_TQI_BOL_SET_S
TQI_MIN_PU_S	TQI_MIN_REQ_PU	TQI_SP_MAF	TQI_ADD_PUC
LV_PUC	LV_S_ACT	N_32	TCO
TQI_REQ_FAST	AMP		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQI_BOL_SET_HYS_HOM	1	0...7FFFH	0...1023.97	0.03125	Nm
Hysteresis for recognition bottom level indicated engine torque (homogeneous mode)					
C_TQI_SP_MAF_OFS_TQI_BOL	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Hysteresis for deactivation of LV_TQI_BOL_MAF_SET					
C_MOD_TQI_BOL_SET_HOM	1	0...2H	0...2	1	-
Manual mode selection for LV_TQI_BOL_SET calculation					
C_TQI_BOL_OFS	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque offset for activation of LV_TQI_BOL_SET					
LC_MOD_TQI_BOL_IGA	1	0...1H	0...1	1	-
Manual mode selection for TQI_BOL_IGA calculation					
IP_TQI_MIN_PU_OFS_AMP	8	0...FFFFH	-1024...1023.97	0.03125	Nm
LDPM_AMP_1_TQDR	8	0...FFFFH	0...5434	0.083	hPa
Ambient pressure correction inTQI_MIN_PU					
IP_TQI_MIN_PU	12x4	0...7FFFH	0...1023.97	0.03125	Nm
LDPM_N_32_2_TQDR	12	0...FFH	0...8160	32	rpm
LDPM_TCO_1_TQDR	4	0...FEH	-48...142.5	0.75	°C
Minimum load limitation on indicated torque level in PU phase					

D.41.1 General Information

In case of homogeneous combustion, TQI_MIN_PU is the minimum limitation for TQI_SP_MAF at trailing throttle condition to ensure either a safe combustion process at low load conditions or a feasible manifold absolute pressure during fuel cut-off. It is formed by a basic limitation TQI_MIN_PU_1, an AMP correction offset TQI_MIN_PU_OFS_AMP, and, during trailing throttle fuel cut-off, by an additional offset TQI_ADD_PUC.

The parameter LV_TQI_BOL_SET indicates that a minimum indicated engine torque has been reached. The test for this condition can be performed in three different modes. LV_TQI_BOL_SET is utilized to trigger state transitions in the Aggegate Engine Operating States (ENOS).

The parameter LV_TQI_BOL_MAF_SET is set to 1 if the minimum engine load (air path) has been reached. It is utilized in ‘Dynamic Torque Request Coordination’.

Function Description

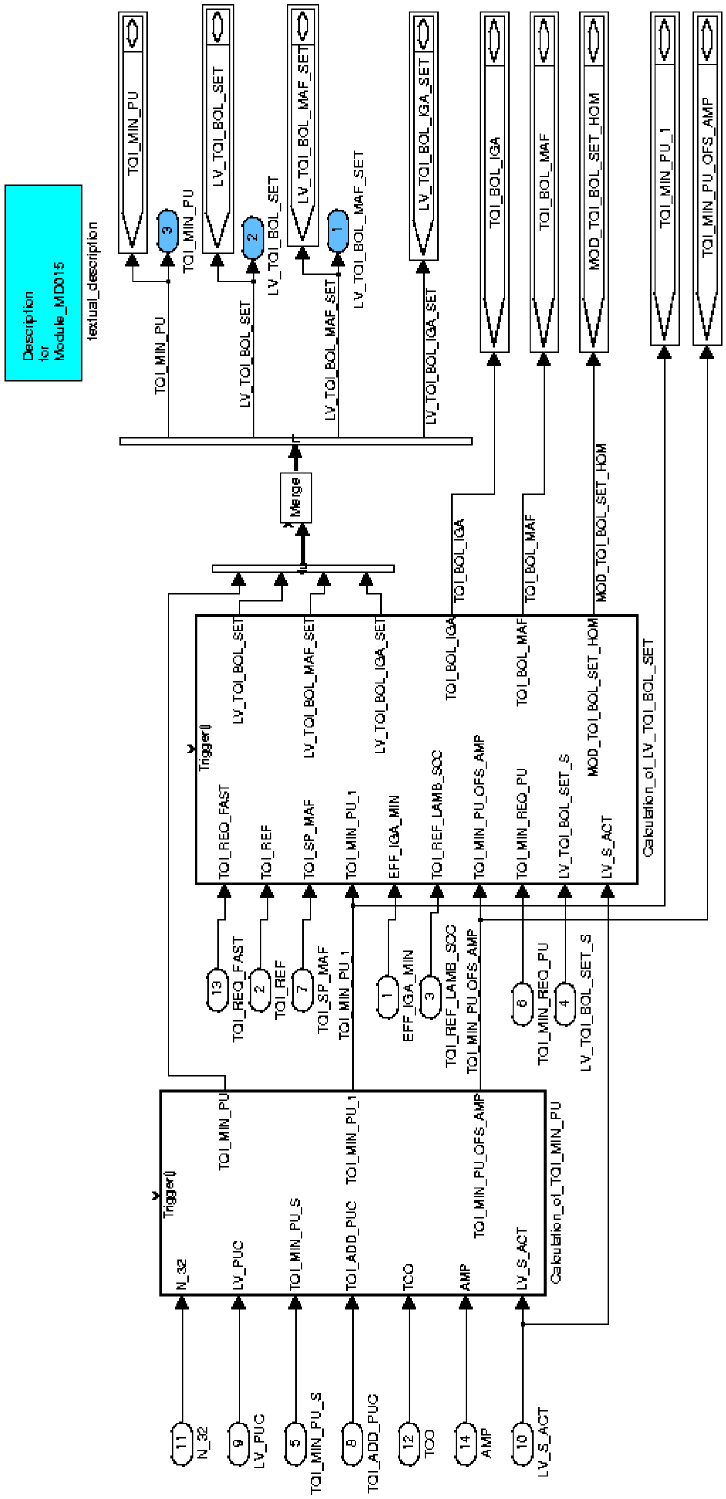


Figure 56 MD015

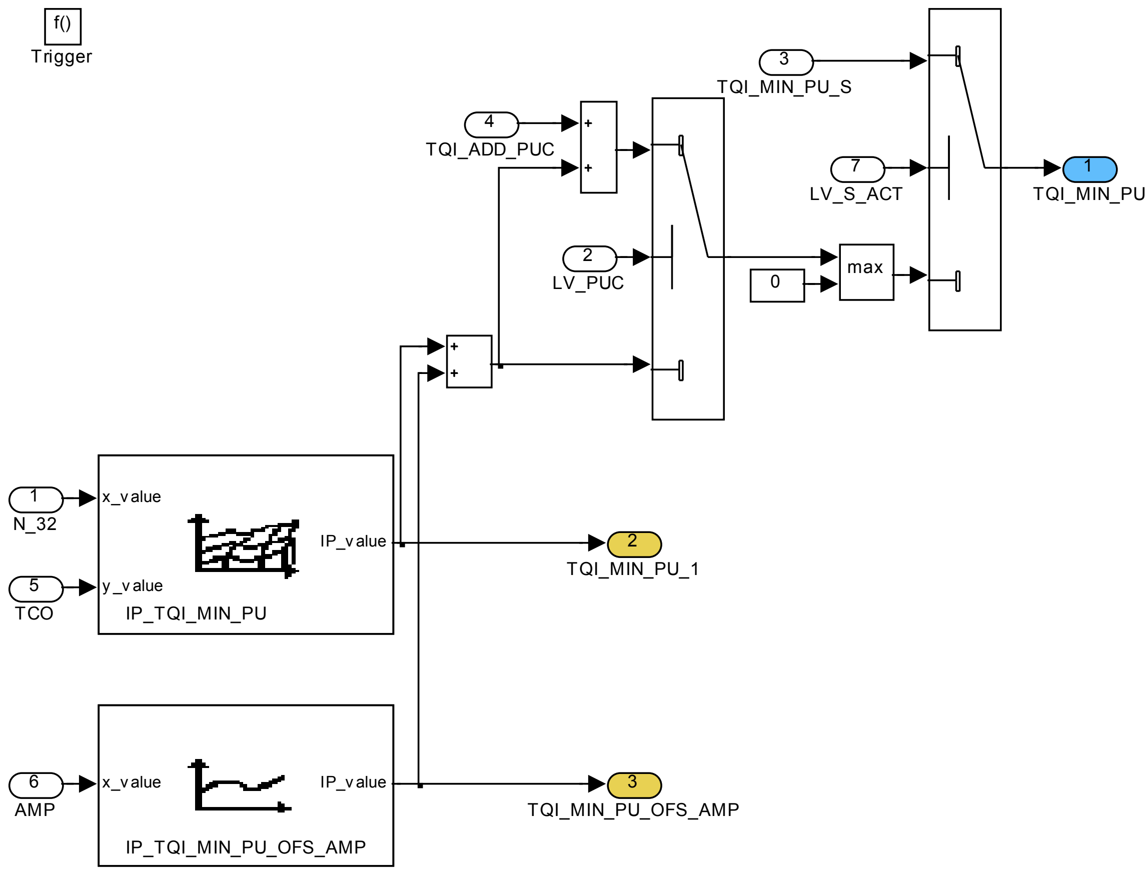


Figure 57 MD015/ Calculation_of_TQI_MIN_PU

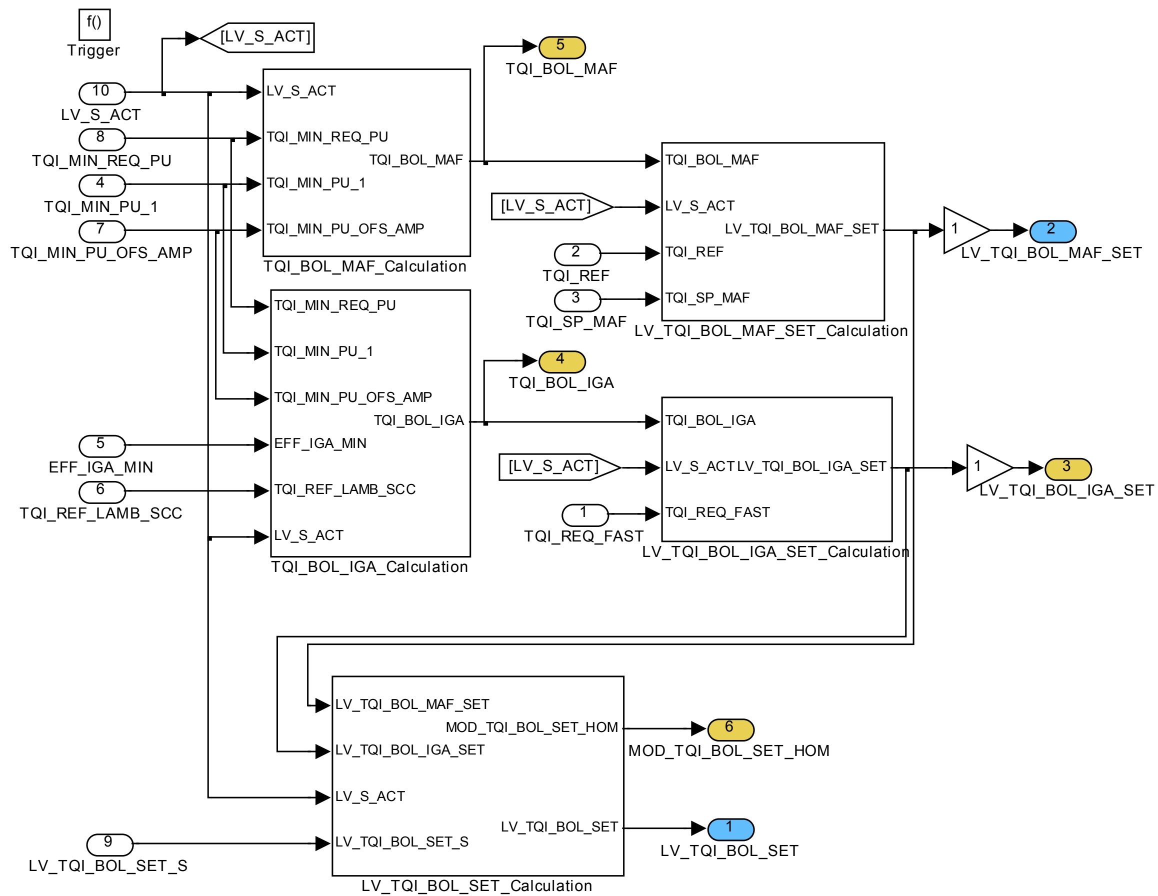


Figure 58 MD015/ Calculation_of_LV_TQI_BOL_SET

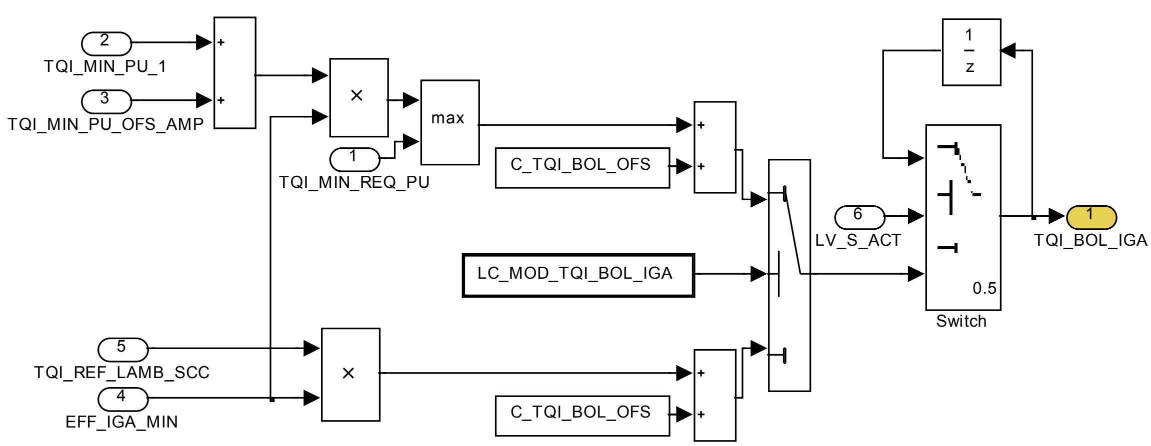


Figure 59 MD015/ Calculation_of_LV_TQI_BOL_SET/ TQI_BOL_IGA_Calculation

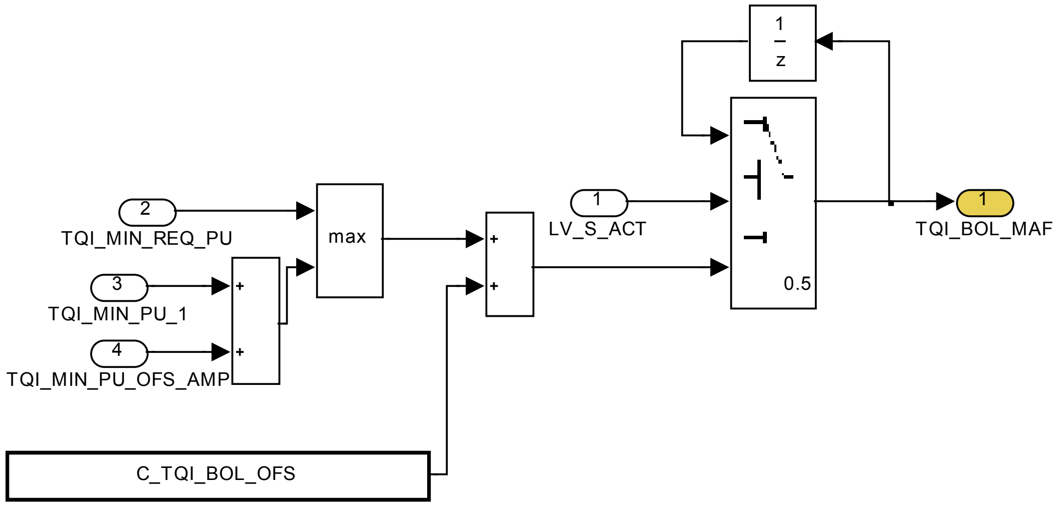


Figure 60 MD015/ Calculation_of_LV_TQI_BOL_SET/ TQI_BOL_MAF_Calculation

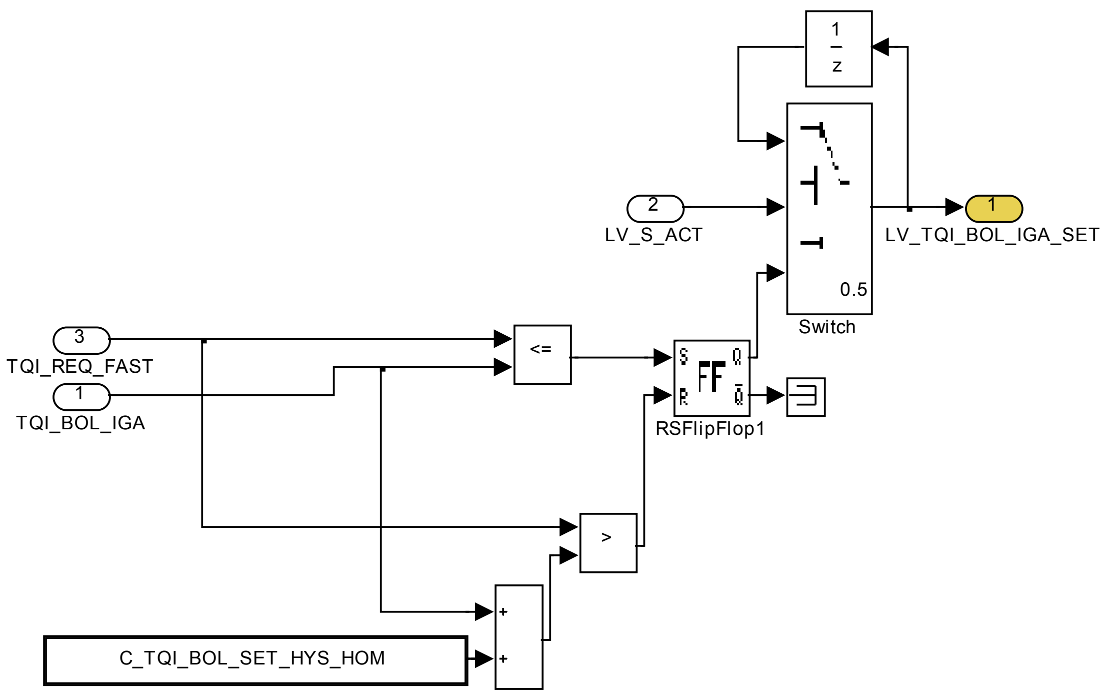


Figure 61 MD015/ Calculation_of_LV_TQI_BOL_SET/ LV_TQI_BOL_IGA_SET_Calculation

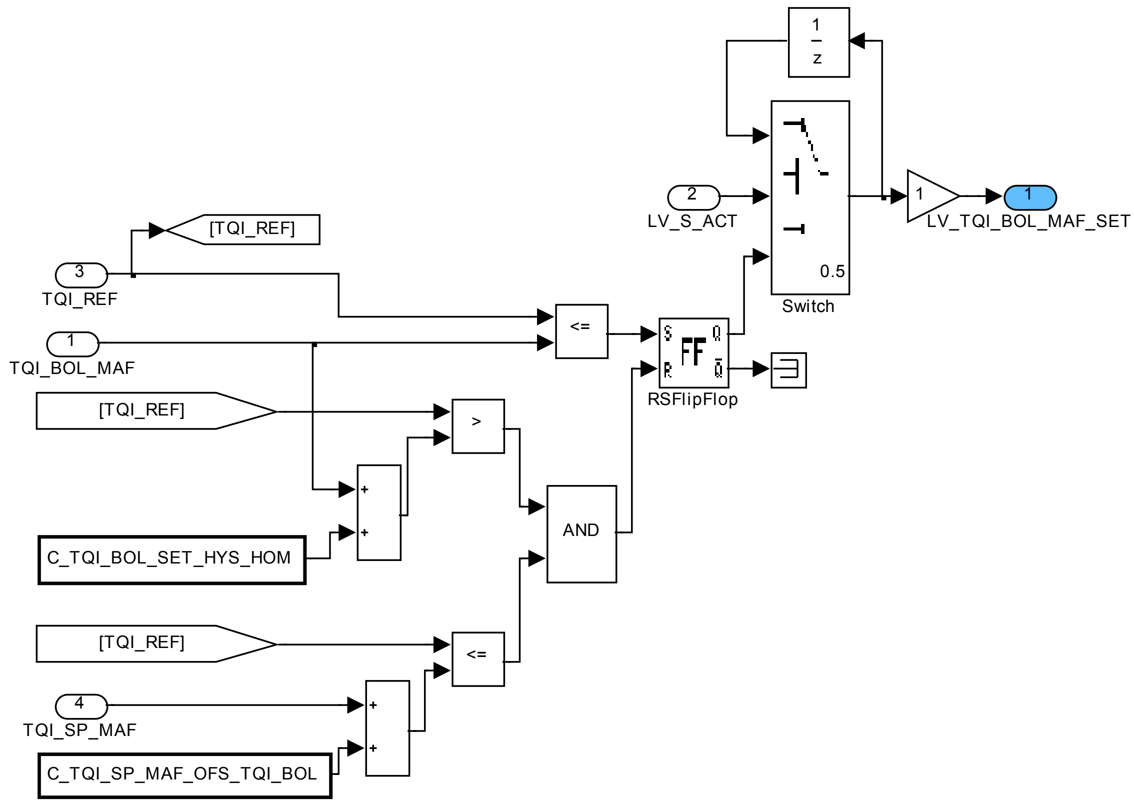


Figure 62 MD015/ Calculation_of_LV_TQI_BOL_SET/ LV_TQI_BOL_MAF_SET_Calculation

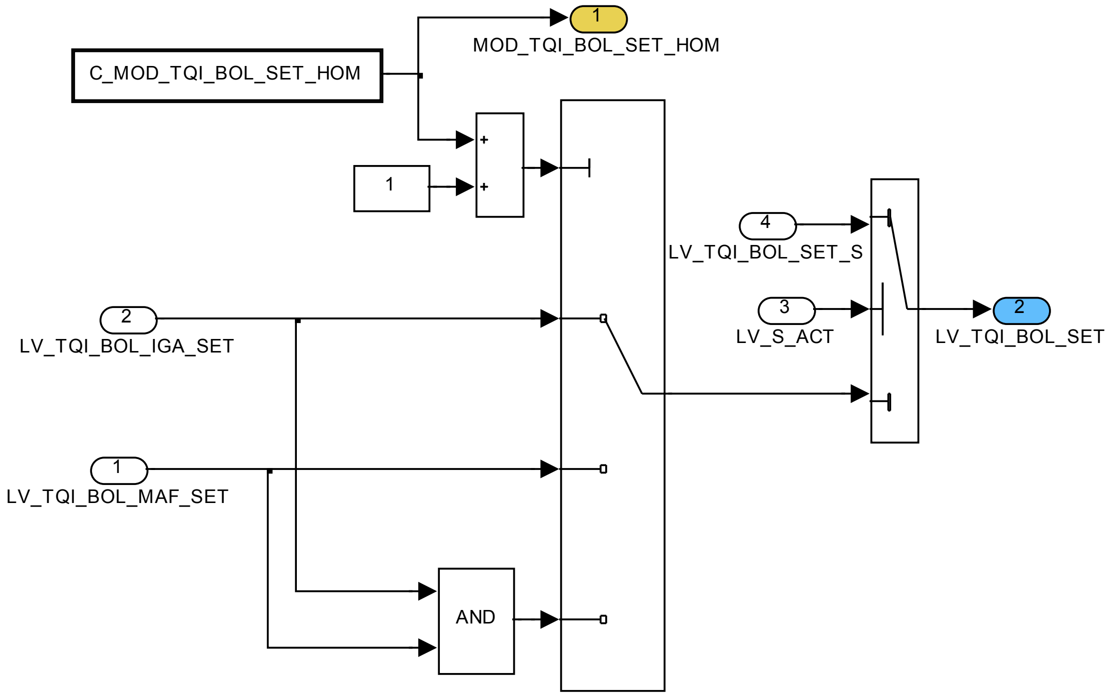


Figure 63 MD015/ Calculation_of_LV_TQI_BOL_SET/ LV_TQI_BOL_SET_Calculation

D.42 Dynamic Torque request coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TQ_DYN_0_REQ	O / V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit general					
LV_TQ_DYN_1_REQ	O / V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit general					
LV_TQ_DYN_0_EMS	-	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit EMS internal					
LV_TQ_DYN_1_EMS	-	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit EMS internal					
LV_TQ_DYN_0_DRIV	V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit (Driver)					
LV_TQ_DYN_1_DRIV	V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit (Driver)					
LV_TQ_DYN_DRIV	O / V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control bit (Driver)					
LV_TQ_DYN_SCC	O / V	0 .. 1H	0 .. 1	1	-
Logical variable for fast fuel cut-off pattern request with DTM authorization					
LV_TQ_DYN_EMS	V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control EMS internal					
LV_TQ_DYN_EXT	V	0 ... 1H	0 .. 1	1	-
Coordinated Dynamic control EMS external requests					
FAC_DYN_DRIV_DEC	O / V	0... FFFFH	0..1.999969	30.51E-6	-
Counter decrement for converging factor					
LV_FCUT_FAST	O / V	0 .. 1H	0 .. 1	1	-
Logical variable for fast fuel cut-off pattern request					
LV_FCUT_FAST_TOT	O / V	0 .. 1H	0 .. 1	1	-
Logical variable for authorizing fast fuel cut-off pattern at IGA_BAS_COR					
LV_TQ_SCC_REQ_TOT	O / V	0 .. 1H	0 .. 1	1	-
Logical variable for fuel cut-off pattern at IGA_BAS_COR					

Input data:

LV_TQ_TRA_DYN	LV_DT	VS	LV_ACT_AJ
LV_REQ_ISC	LV_PU	LV_PUC	LV_ST
LV_TQ_ADD_REQ	STATE_CMB_CTL	LV_ASR_ACT	TQI_REQ_TRA
LV_GS_ENA	LV_PL	N_32	TCO
LV_CS	LV_TQI_BOL_MAF_SET	LV_N_MAX	LV_CT
LV_MTC_CUR_OFF	LV_PAS_RAMP_ACT_I_IS	LV_PAS_RAMP_ACT_P_D_IS	LV_TQ_LIM_INTV
LV_MSR_ACT	LV_N_MAX_VS_0	LV_ERR_CS	

FUNCTION DESCRIPTION:

General information:

This module coordinates project-specific requests for the Dynamic Torque intervention Manager.

Dynamic requests from torque structure intern as well as external requests (e.g. from driving stability systems) are collected and priorised. This module description is seperated into a calculation of ignition angle intervention and a calculation for single cylinder cut-off request (SCC).

A special dynamic request is LV_N_MAX: it has an influence on both types of intervention depending on calibration.

This module contains also the calculation of for request of SCC patterns independent from Dynamic Torque intervention Manager in case of gear shift for preventing engine speed overshoot.

Application conditions:

Initialisation: *LV_TQ_DYN_0_REQ = 1*

LV_TQ_DYN_1_REQ = 0

LV_TQ_DYN_0_DRIV = 1

LV_TQ_DYN_1_DRIV = 0

LV_TQ_DYN_DRIV = 1

LV_FCUT_FAST = 0

LV_x_TOT = 0; x is any string

Recurrence: 10ms

Activation: all engine states

Deactivation: -

D.42.1 Dynamic torque requests for ignition angle intervention

Description:

To adjust the Dynamic Torque intervention Manager mode dynamic request bits are calculated. Two bits allow to adjust the mode to realize torque requests with a combustion engine at the following modes (see also the DTM function):

	LV_TQ_DYN_1_x	LV_TQ_DYN_0_x
Economy mode: realization of torque request without ignition intervention (fast path setpoint includes airpath dynamic)	0	0
High dynamic mode: identical setpoint for slow and fast path leads to ignition intervention for fastest possible torque request realization	0	1
Torque reduction only by ignition angle adjustment	1	0
Torque reserve mode: requests, that an additional torque request in slow path accelerate the slow path or for ignition angle retardation	1	1

An additional output LV_TQ_DYN_DRIV marks, if an "converging torque mode" at transition from "eco" to "high dynamic" is allowed.

Note: The priorisation of dynamic requests must be considered with priorisation of torque requests in "Minimum/Maximum torque request selection". For this realization of the module, any "high dynamic" request wins and the torque requests resulting from the "Minimum/Maximum selection module" will be realized like this, no matter if the *corresponding torque won the selection or not. This is done to ensure that the driving stabilty function have the dynamic highest priorisation.*

D.42.1.1 Dynamic requests that require convergence to high dynamic mode

FUNCTION DESCRIPTION:

Driver demand:

The driver demand can be realized in "eco" and "high dynamic" torque mode. For the transition from "eco" to "high dynamic" a "convergence function" for the torque request in fast path is allowed (see the specification "Dynamic Torque intervention Manager"). So this logic contains all the requests for transition to "high dynamic" via "convergence" function.

Formula section:

If **not** LC_TQ_DYN_DRIV_INH ; manual high dynamic request inhibited
and **[**LC_TQ_DYN_DRIV = 1 ; manual dynamic mode request
or (LV_TQ_TRA_DYN = 1 **and not** LC_TQ_DYN_TQ_TRA_INH)
or (LV_TQI_BOL_MAF_SET = 1
and not LC_TQ_DYN_TQI_BOL_MAF_INH)
; TQI_REF close to TQI_MIN_PU
or (LV_PU = 1 **and not** LC_TQ_DYN_PU_INH)
; ignition intervention during PU
or (LV_PUC = 1 **and not** LC_TQ_DYN_PUC_INH)
; ignition intervention during PUC allowed
or (LV_DT = 0 **and not** LC_TQ_DYN_DT_TRA_INH)]
; open drive train
then LV_TQ_DYN_DRIV = 1
else LV_TQ_DYN_DRIV = 0
endif

If ((LV_DT = 0 **and not** LC_TQ_DYN_DT_TRA_INH) ; open drivetrain
then If VS < C_VS_MIN_DT_FAST
then FAC_DYN_DRIV_DEC = C_FAC_DYN_DRIV_DEC_1 ; 'conv' at standing car
else FAC_DYN_DRIV_DEC = C_FAC_DYN_DRIV_DEC_2 ; 'conv' at running car
endif
else FAC_DYN_DRIV_DEC = C_FAC_DYN_DRIV_DEC ; 'conv' in all other cases
endif

D.42.1.2 Dynamic requests for direct transition to high dynamic mode EMS internal

FUNCTION DESCRIPTION:

Other intervention requests from Torque structure:

This logic collects all EMS internal requests for transition to high dynamic mode without convergence function. The bit LV_TQ_DYN_DRIV is also required as it requests "high dynamic" as well.

For engine speed limitation several intervention modes are possible:

- intervention by ETC only (depending on MAF),
- an additional ignition intervention,
- intervention by ignition and fuel cut-off patterns or
- MAF and patterns

This can be adjusted by C_STATE_N_MAX. In any case for ignition or fuel cut-off pattern LV_N_MAX requires high dynamic mode. If only MAF and pattern is allowed, the ignition retardation is prevented via an algorithm in 'Torque based pattern calculation' that controls the activation of fuel cut-off in comparison of ignition retardation.

Formula section:

```
If    LV_TQ_ADD_REQ = 1                        ; active torque reserve
      or (LV_ACT_AJ = 1 and not LC_TQ_DYN_AJ_ACT_INH) ; active anti jerk function
      or LV_REQ_ISC = 1                        ; request idle speed controller
      or (STATE_CMB_CTL &lt;> HOM_AFS and STATE_CMB_CTL &lt;> HOM_AFL)
                                               ; any other operation than HOM_AFS
                                               ; or HOM_AFL needs fast intervention mode
      or LV_ST = 1                             ; in start
      or ((LV_DT = 0 or VS < C_VS_MIN_DT_FAST )
             and LC_TQ_DYN_DT_VS_TRA_INH)      ; while gear shift or at standing car
      or (LV_N_MAX = 1 and C_STATE_N_MAX <> 0)       ; LV_N_MAX must request
                                               ;'high dyn' mode also for pattern w/o IGA
      or LV_PAS_RAMP_ACT_P_D_IS = 1            ; PD-part passive ramp active (idle speed
                                               controller)
      or (LV_PAS_RAMP_ACT_I_IS = 1
         and  LC_TQ_DYN_PAS_RAMP_ACT_I_IS_INH)
                                               ; I-part passive ramp active (idle speed
                                               controller)
      or LV_TQ_LIM_INTV = 1                    ; torque intervention for safety active
then        LV_TQ_DYN_EMS = 1
else        LV_TQ_DYN_EMS = 0
endif
```

D.42.1.3 Dynamic requests for transition to high dynamic mode EMS external

Description:

For automatic transmission a torque increase as well as a decrease can be demanded as dynamic request.

For example ASR wants the fastest possible desired torque setting when active. Dynamic requests can also require torque increase as driving stability algorithms (MSR) or automatic gear shift and external cruise controller.

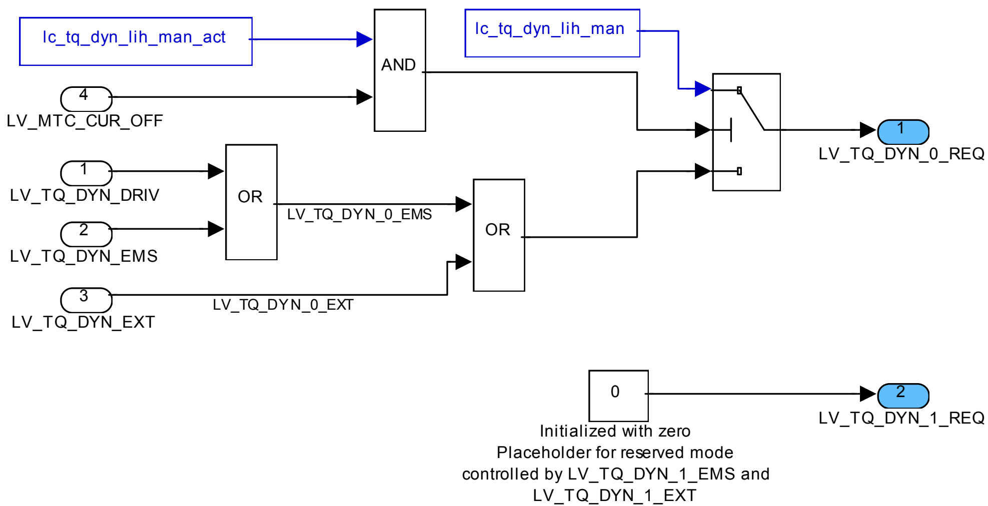
So all external requests from driving stability functions lead to immediate intervention.

Formula section:

LV_TQ_DYN_EXT =	LV_GS_ENA = 1	**or**
	LV_ASR_ACT =1	**or**
	LV_MSR_ACT =1	**or**

D.42.2 Final priorisation of dynamic requests

Signal flow diagram:



Formula section:

EMS-internal requests (reserved; LV_TQ_DYN_0_EMS used internally only)

LV_TQ_DYN_0_EMS = LV_TQ_DYN_DRIV

or LV_TQ_DYN_EMS

EMS-external requests (for info only, no calculation)

LV_TQ_DYN_0_EXT = LV_TQ_DYN_EXT

LV_TQ_DYN_1_EXT = 0

As long as all requests require the "High dynamic mode" and the reserved mode is not used no calculations are done for LV_TQ_DYN_1_EMS and LV_TQ_DYN_1_EXT to save memory.

Highest priority for Limp Home mode:

If (LV_MTC_CUR_OFF = 1 **and** LC_TQ_DYN_LIH_MAN_ACT)

Then	LV_TQ_DYN_0_REQ =	LC_TQ_DYN_LIH_MAN	
	LV_TQ_DYN_1_REQ =	0	; this bit is reserved
Else	LV_TQ_DYN_0_REQ =	LV_TQ_DYN_0_EMS **or** LV_TQ_DYN_0_EXT	
	LV_TQ_DYN_1_REQ =	0	; this bit is reserved
endif			

D.42.3 Dynamic torque requests for single cylinder cut-off

D.42.3.1 Fuel cut-off authorized by DTM

Description:

Here all dynamic requests for SCC patterns are collected that can be authorized by the DTM depending on IGA intervention state.
The LC_TQ_DYN_SCC_x_ENA defines if a fuel shut-off pattern is applied. The LC_TQ_SCC_REQ_TOT_x_ENA defines if fuel shut-off pattern is applied immediatly (1) or a calibrateable ignition retardation is allowed (0). For LV_N_MAX a state C_STATE_N_MAX is used to define the intervention mode (see calibration data section). For engine speed limitation at stopped vehicle, no ignition intevention is allowed.

Formula section:

LV_TQ_DYN_SCC = (LV_ASR_ACT **and** LC_TQ_DYN_SCC_ASR_ENA) **or**
(LV_GS_ENA **and** LC_TQ_DYN_SCC_GS_ENA) **or**
[LV_N_MAX **and**
(C_STATE_N_MAX == 2 **or** C_STATE_N_MAX == 3)]

LV_TQ_SCC_REQ_TOT =
(LV_ASR_ACT **and** LC_TQ_SCC_REQ_TOT_ASR_ENA) **or**
(LV_GS_ENA **and** LC_TQ_SCC_REQ_TOT_GS_ENA) **or**
[LV_N_MAX **and** (C_STATE_N_MAX == 3)] **or**
[LV_N_MAX_VS_0]

D.42.3.2 Fast fuel cut-off

Description:

The objective of this chapter is to define the condition for single cylinder cut-off. If fast fuel cut-off is enabled, cylinder deactivation will be managed by the torque management, but without DTM.

The fast fuel cut-off will be enabled for the following situation:

- the coolant temperature must exceed a calibrateable threshold
- the accelerator pedal must be released and the clutch has to be opened; Torque Request after Torque Transient Function has to be zero (This allows a calibrateable time delay at gear zero via Torque Transient function)

For preventing unwanted injection intervention at transition to IS a threshold can be applied, keeping distance for shut-off to reinjection in IS.

The request can be deactivated in "Torque based pattern calculation".

Formula section:

IF (LV_PL == 1 **or** LV_PU ==1) **AND** % activation condition
TCO >= C_TCO_MIN_FCUT_FAST **AND**
LV_CS == 1 **AND**
LV_ERR_CS = 0 **AND**
LV_CT == 1 **AND**
TQI_REQ_TRA <= C_TQI_REQ_TRA_THD_FCUT_FAST **AND**
N_32 >= C_N_MIN_FCUT_FAST
THEN LV_FCUT_FAST = 1
ELSE LV_FCUT_FAST = 0
END

LV_FCUT_FAST_TOT = LC_FCUT_FAST_TOT

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
LC_TQ_DYN_DRIV	1	0 .. 1H	0 .. 1	1	-
Manual switch for fast torque request enabled					
C_VS_MIN_DT_FAST	1	0 .. FFH	0 .. 255	1	Km/h
Minimum vehicle speed for fast torque mode at open drive train					
LC_TQ_DYN_DRIV_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request from driver					
LC_TQ_DYN_TQ_TRA_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request from torque transient					
LC_TQ_DYN_DT_TRA_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request from open drivetrain detection					
LC_TQ_DYN_AJ_ACT_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for choosing anti jerk active in high dynamic mode (1) or in eco mode (0)					
LC_TQ_DYN_LIH_MAN_ACT	1	0 .. 1H	0 .. 1	1	-
Switch for activation of manual mode selection for limp home					
LC_TQ_DYN_LIH_MAN	1	0 .. 1H	0 .. 1	1	-
Manual dynamic mode selection for limp home					
C_STATE_N_MAX	1	0H 1H 2H 3H	"No_INTV" "IGA_INTV" "IGA_SCC_INTV" "SCC_INTV"	1	-
Type of intervention request at active LV_N_MAX					
C_TCO_MIN_FCUT_FAST	1	0 ... FEH	-48 ... 142.5	0.75	°C
Min. coolant temperature fast fuel cut-off					
C_N_MIN_FCUT_FAST	1	0...FFH	0...8160	32	rpm
Min. engine speed for authorizing fast fuel cut-off					

LC_TQ_DYN_SCC_ASR_ENA	1	0 .. 1H	0 .. 1	1	-
Manual bit for allowing SCC at ASR request					
C_TQI_REQ_TRA_THD_FCUT_FAST	1	0...7FFFH	0...1023.97	0.03125	Nm
Threshold for filtered driver request for authorizing fast fuel cut-off					
C_FAC_DYN_DRIV_DEC	1	0... FFFFH	0..1.999969	30.51E-6	-
Counter decrement for converging factor general: *MUST NOT BE ZERO!!!*					
C_FAC_DYN_DRIV_DEC_1	1	0... FFFFH	0..1.999969	30.51E-6	-
Counter decrement for converging factor for standing vehicle w/ open drive train: *MUST NOT BE ZERO!!!*					
C_FAC_DYN_DRIV_DEC_2	1	0... FFFFH	0..1.999969	30.51E-6	-
Counter decrement for converging factor for running vehicle w/ open drive train: *MUST NOT BE ZERO!!!*					
LC_TQ_SCC_REQ_TOT_ASR_ENA	1	0 .. 1H	0 .. 1	1	-
Manual bit for priorising SCC intervention before IGA retardation for ASR					
LC_TQ_SCC_REQ_TOT_GS_ENA	1	0 .. 1H	0 .. 1	1	-
Manual bit for priorising SCC intervention before IGA retardation for gearshift					
LC_TQ_DYN_SCC_GS_ENA	1	0 .. 1H	0 .. 1	1	-
Manual bit for allowing SCC at automated gear shift request					
LC_TQ_DYN_PU_INH	1	0 .. 1H	0 .. 1	1	-
Manual bit for inhibit intervention via convergence mode					
LC_TQ_DYN_PUC_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request in PUC					
LC_TQ_DYN_TQI_BOL_MAF_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request at around TQI_BOL_MAF and below					
LC_FCUT_FAST_TOT	1	0 .. 1H	0 .. 1	1	-
LC_TQ_DYN_DT_VS_TRA_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request from open drivetrain and low vehicle speed detection					
LC_TQ_DYN_PAS_RAMP_ACT_I_IS_INH	1	0 .. 1H	0 .. 1	1	-
Manual switch for disabling high dynamic request from idle speed controller					

D.43 Authorization of ignition intervention

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_TQ_IGA_ENA	O / V	0 .. 1H	0 .. 1	1	-
Logical variable for enabling ignition intervention					

Input data:

LV_ACT_AJ	LV_REQ_ISC	LV_TQ_DYN_IGA	

FUNCTION DESCRIPTION:

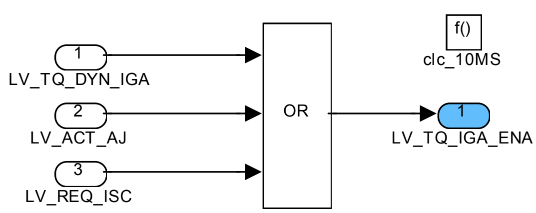
This chapter is triggered separate from the previous chapter !

This is the final instance in torque setpoint calculation for authorization of ignition intervention. If the output is active, ignition intervention is authorized. It *must* be calculated after:

- Dynamic Torque Management
- Anti jerk controller
- Idle speed controller

or any algorithm that requires immediate ignition intervention. It should be calculated directly before Torque coordination. Otherwise Anti Jerk and Idle speed controller output are cut for one sample at activation.

Signal flow diagram:



Formula section:

LV_TQ_IGA_ENA = LV_TQ_DYN_IGA
or LV_ACT_AJ
or LV_REQ_ISC

Application conditions:

Initialisation: LV_TQ_IGA_ENA = 1

Recurrence: 10ms

Activation: at every engine state

Deactivation: -

D.44 Minimum/Maximum Torque Request Selection

D.44.1 Priorisation of Torque Requests

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
TQI_REQ_SLOW_SEL	V / O	8000...7FFF H	-1024...1023.97	0.03125	Nm
Slow indicated engine torque request selected					
TQI_REQ_FAST_SEL	V / O	8000...7FFF H	-1024...1023.97	0.03125	Nm
Fast indicated engine torque request selected					
TQI_EMS	V / O	8000...7FFF H	-1024...1023.97	0.03125	Nm
Requested indicated engine torque without external interventions					
TQI_INTER_SLOW	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Intermitted value slow torque request after minimum selection					
LV_TQ_INC	V	0 , 1 H	0 , 1	1	-
Logical variable torque increase active					
TQI_INTER_FAST	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Intermitted value fast torque request after minimum selection					
TQI_REQ_FAST_INC	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Fast torque increase request					
TQI_REQ_FAST_DEC	V	8000...7FFF H	-1024...1023.97	0.03125	Nm
Fast torque decrease request					
NR_TQ_SEQ_SLOW	V	0..FF H	0..255	1	-
Number of torque request that has won the selection for slow path					
NR_TQ_SEQ_FAST	V	0..FF H	0..255	1	-
Number of torque request that has won the selection for fast path					

Input data:

TQI_ADD_ACT	TQI_GS_SLOW_DEC	TQI_N_MAX	TQI_MIN_REQ
TQI_REQ_TRA	TQI_GS_SLOW_INC	TQI_GS_FAST_DEC	TQI_GS_FAST_INC
TQI_ASR_SLOW	TQI_ASR_FAST	TQI_MSR	TQI_MAX_ETC
TQI_P_MAX_SLOW	TQI_P_MAX_FAST		

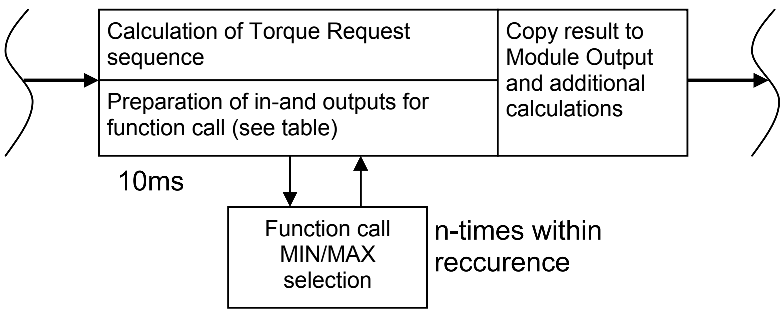
FUNCTION DESCRIPTION:

General information:

This module coordinates all internal and external torque requests. It generates TQI_REQ_SLOW_SEL for the slow torque path (load) and TQI_REQ_FAST_SEL which represents the fast torque path (ignition, cylinder cut-off and lambda).

This is done by a sequence of function calls (SW uses macro), where the last priorised torque request is compared to a torque request according to the priorisation in the sequence. The information which torque request has won the comparison is stored. Each sequence is called once in a recurrence with n function calls, where n is the number of torque requests per path to be priorised. If the torque requests have identical values, the request with higher number in the sequence wins the comparison.

The higher the number of a request in the sequence the higher it's priorisation



Application conditions:

Initialisation:	at reset:	TQI_REQ_SLOW_SEL = 0 Nm
		TQI_REQ_FAST_SEL= 0 Nm
		TQI_EMS = 0 Nm
		LV_TQ_INC = 0
Recurrence:	10 ms	
Activation:	at every engine state	
Deactivation:	-	

D.44.1.1 Slow path minimum/maximum selection

Description:

The following tables describe the priorisation of the torque inputs and the selection type Minimum or Maximum compared to the previously calculated value. It also describes when an intermediate value has to be calculated for further comparison.

There are two sequences, one for the slow path, the other one for the fast path.

The LV_MIN_MAX_SEL describes the selection type: LV_MIN_MAX_SEL = 0 means a minimum selection, if it is one it requests a maximum selection (see also the Minimum / Maximum selection function). The name of the function is "min_max_sel".

The following table shows the sequence for the slow path and the type of selection.

Priorisation Table - Slow Path (Prio_Table_Slow):

Sequence Order **nr_seq**	Torque Input **tq_inp**	Min/Max selection **lv_min_max_sel**	Reserved	Value stored in:
1	TQI_ADD_ACT	1*)		
2	TQI_P_MAX_SLOW	0		
3	TQI_GS_SLOW_DEC	0		
4	TQI_N_MAX	0		
5	TQI_MAX_ETC	0		
6	TQI_ASR_SLOW	0		
7	TQI_MIN_REQ	1		TQI_INTER_SLOW
8	TQI_GS_SLOW_INC	1		
9	TQI_MSR	1		TQI_REQ_SLOW_SEL

The sequence organizes the function calls of the selection macro and provides the input data to the function.

*) This first value is a maximum selection with the initialization value 0.

Formula section:

The variables in small letters are used internally only. See the chapter "Minimum Maximum selection function" for the parameters of the function call.

Initialisation: tq_out = 0;

nr_seq_out = 0;

Sequence:

```
For nr_seq = 1 to 9
        tq_inp1 = tq_out;
        tq_inp2 = value "tq_inp" from "Prio_Table_slow(nr_seq)";
        nr_seq_inp1 = nr_seq_out;
        nr_seq_inp2 = nr_seq;
        lv_min_max_sel = id_min_max_sel(nr_seq);
        [tq_out, nr_seq_out]  = FCN_CALL(min_max_sel)
        if nr_seq = 6
        then TQI_INTER_SLOW=tq_out
        endif
endfor
TQI_REQ_SLOW_SEL = tq_out;
NR_TQ_SEQ_SLOW = nr_seq_out;
```

D.44.1.2 Fast path minimum/maximum selection

Description:

Priorisation Table - Fast Path (Prio_Table_Fast):

Sequence Order **nr_seq**	Torque Input **tq_inp**	Min/Max selection **lv_min_max_sel**	Reserved	Value stored in:
1	TQI_REQ_TRA	1*)		
2	TQI_P_MAX_FAST	0		
3	TQI_N_MAX	0		
4	TQI_GS_FAST_DEC	0		
5	TQI_ASR_FAST	0		
6	TQI_GS_FAST_INC	1		TQI_INTER_FAST
7	TQI_MIN_REQ	1		
8	TQI_MSR	1		TQI_REQ_FAST_SEL

The calculation for the fast path is equal to the slow path. Additionally one more intermediate value is calculated.

*) This first value is a maximum selection with the initialization value 0.

Formula section:

All variables used internally are written in small letters.

Initialisation: tq_out = 0;

nr_seq_out = 0;

Sequence:

For nr_seq = 1 **to** 8

tq_inp1 = tq_out;

tq_inp2 = value "tq_inp" from "Prio_Table_Fast(nr_seq)";

nr_seq_inp1 = nr_seq_out;

nr_seq_inp2 = nr_seq;

lv_min_max_sel = id_min_max_sel(nr_seq);

[tq_out, nr_seq_out] = FCN_CALL(min_max_sel)

if nr_seq == 5

then TQI_INTER_FAST = tq_out;

endif

endfor

TQI_REQ_FAST_SEL = tq_out;

NR_TQ_SEQ_FAST = nr_seq_out;

D.44.1.3 Additional Calculations

For analyzing the minimum/maximum torque request selection additional values are needed, but they do have no affect to slow or fast path or do not come from the sequential calculation.

Formula section:

TQI_EMS = min (TQI_REQ_TRA, TQI_N_MAX, TQI_P_MAX_FAST)

If TQI_EMS < TQI_INTER_FAST

Then LV_TQ_INC = 1

Else LV_TQ_INC = 0

Endif

TQI_REQ_FAST_INC = TQI_GS_FAST_INC

TQI_REQ_FAST_DEC = min (TQI_REQ_TRA, TQI_GS_FAST_DEC)

D.44.1.4 Minimum Maximum selection function

Application conditions:

Initialisation: *tq_out = 0; nr_seq_out = 0;*

Recurrence: function call, nr_seq-times in one recurrence from the sequence

Activation: function call

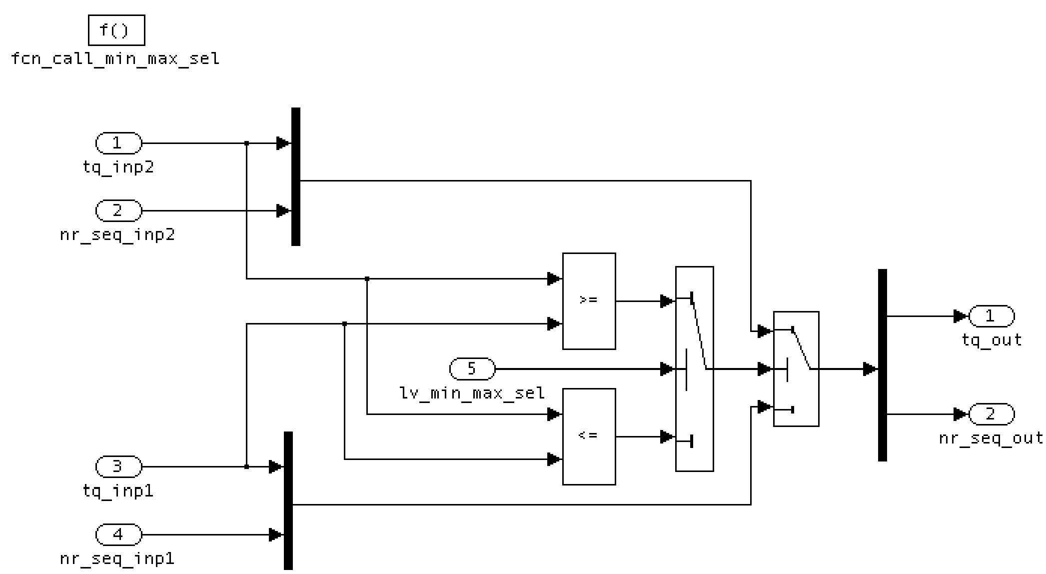
Deactivation: -

Description:

This function produces as output a torque and its corresponding position in the priorisation sequence. The result can be a minimum or a maximum of two torque request inputs, depending on a selection bit. If both torque requests have the same value, the request with higher number in the sequence wins the selection.

This function is called *n*-times within one reccurence, where *n* is the number of torque requests per sequence.

Signal flow diagram:



D.45 Dynamic Torque intervention Manager

General information:

The intention of the Dynamic Torque intervention Manager (DTM) is to have a central coordination of dynamic torque requests. It controls the realization of a torque request via the slow air path and the fast path with ignition intervention, fuel cut-off or lambda variation. It produces two kinds of outputs:

- Torque requests for slow and fast path for correct representation based on the Torque Structure.
- Logical values for switching different fast path realization variants to either their passive values or setpoint calculation via Torque Structure.

The state machine is controlled via dynamic control bits, that allow different states of intervention variants. Also the transitions from one state to the other can be influenced.
The description of the DTM is parted into three chapters: one for manipulation of the slow air path and the ignition path, another one for authorizing fuel cut-off and a last one for lambda variation. According to this is the priorisation of the different possibilities of torque intervention.
So the DTM finally adjusts the dynamic requirements from the control bits and gives the possibility to use several transitions between the different requirements.

D.45.1 Coordination ignition

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_TQ_DYN_IGA	V / O	0 .. 1 H	0 .. 1	1	-
Logical variable ignition intervention requested					
STATE_TQ_DYN_IGA	V	0H 1H 2H 3H 4H 5H	"dtm_pas" “dtm_eco” “dtm_hi_dyn” “dtm_trans” "dtm_conv" “dtm_tq_res”	1	-
Actual state of the torque dynamic state machine IGA intervention					
TQI_REQ_FAST_DTM	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated torque request fast					
TQI_REQ_SLOW_DTM	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated torque request slow					
FAC_DYN_DRIV	V	0... FFFFH	0..1.999969	30.51E-6	-
Counter for convergence factor					

Input data:

TQI_REQ_FAST_SEL	TQI_REQ_SLOW_SEL	LV_TQ_DYN_0_REQ	LV_TQ_DYN_1_REQ
TQI_BAS	LV_TQ_DYN_DRIV	LV_HOM_RUN	LV_MTC_CUR_OFF
FAC_DYN_DRIV_DEC			

FUNCTION DESCRIPTION:

General information:

This chapter contains the coordination of slow air path and ignition path. There are three states, where a certain torque input is fed through:

- "economy mode", where ignition intervention is not allowed
- "high dynamic mode", where ignition intervention is allowed
- "torque reserve mode", where the torque request for the airpath can be influenced (this mode is currently without function).

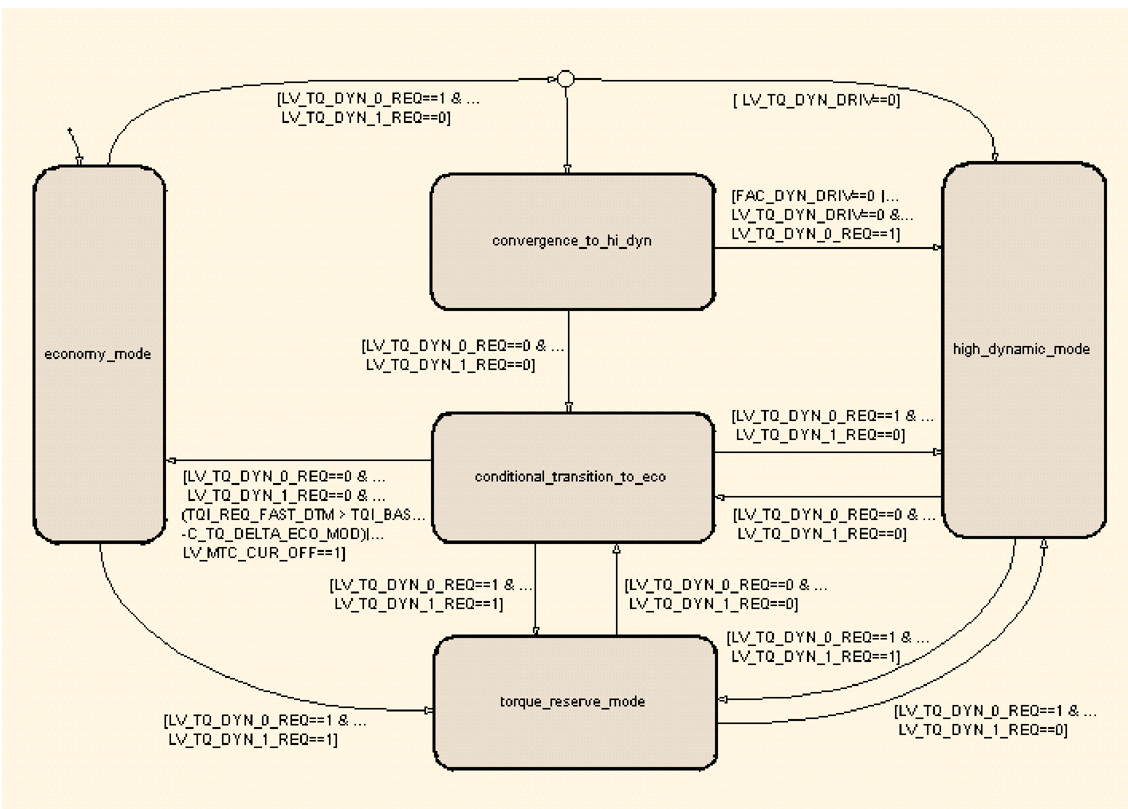
Furthermore there are transition states:

- "convergence to high dynamic mode" for smooth transition from the input to TQI_REQ_FAST_DTM in "eco" to the input in "high dynamic"
- "conditional transition to economy mode" for preventing torque jumps on exit from any mode to "economy mode" .

All these states are controlled by the dynamic control bits LV_TQ_DYN_x_REQ. For the transitions additionally a bit for low dynamic request is taken into consideration: LV_TQ_DYN_DRIV.

For correct behavior it is absolutely necessary that the logical variables for dynamic requests are coordinated correctly. See the specification of "Dynamic torque request coordination"

Signal flow diagram:



Description:

The several states define the assignment of the input TQI_REQ_SLOW_SEL and TQI_REQ_FAST_SEL to the outputs TQI_REQ_SLOW/FAST_DTM.

The state machine produces also the logical output

LV_TQ_DYN_IGA

to authorize or prohibit ignition intervention. The logical state 1 means fast intervention is allowed, the logical state=0 means the fast intervention is prohibited.

In "economy" input to the fast path is TQI_BAS, therefore no fast intervention is applied. In "high dynamic" the torque request inputs are just fed through, spark retard is allowed.

For transition from "eco" to "high dynamic" there are two possibilities:

1. A high dynamic request leads to direct entry of high dynamic.
2. A low dynamic request enters the "convergence to high dynamic". Here an convergence function performs a smooth transition for the fast path from TQI_BAS to the new target value for the fast path.

To avoid large changes in spark advance no direct entry to "economy mode" is allowed. The increment to reach the target torque request is limited to C_TQ_DELTA_MAX_COMP. A calibrateable C_TQ_DELTA_ECO_MOD defines the maximum allowed torque jump on reentry of "economy mode".

If there is a problem with the state machine when a undefined state is produced then the default state is set.

Note: The convergence factor FAC_DYN_DRIV_DEC must not be zero. It is recommended to have a bigger value then 0.001 (equals 10 sec converging time). Otherwise it is possible only to leave the convergence state with high dynamic request but not from LV_TQ_DYN_DRIV.

Application conditions:

Activation: at every engine state

Deactivation: -

Initialization: at reset

LV_TQ_DYN_IGA = 0

STATE_TQ_DYN_IGA = "passive"

FAC_DYN_DRIV = 1

Recurrence: 10 ms

Formula section:

If LC_DTM_ENA == 0

then

"DTM passive"

on entry: LV_TQ_DYN_IGA = 1

STATE TQ_DYN_IGA = "passive"

during: TQI_REQ_FAST_DTM = TQI_REQ_FAST_SEL

TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

Else DTM_State_Machine active:

"economy_mode"

on entry: LV_TQ_DYN_IGA = 0

STATE_TQ_DYN_IGA = "eco"

during: TQI_REQ_FAST_DTM = TQI_BAS

TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

"high_dynamic_mode"

on entry: LV_TQ_DYN_IGA = 1

STATE_TQ_DYN_IGA = "high_dyn"

during: TQI_REQ_FAST_DTM = TQI_REQ_FAST_SEL

TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

"conditional_transition_to_economy_mode"

on entry: LV_TQ_DYN_IGA = 1

STATE_TQ_DYN_IGA = "eco"

during: **If** [TQI_REQ_FAST_SEL (n) - TQI_REQ_FAST_DTM (n-1)] < C_TQ_DELTA_MAX_COMP

then TQI_REQ_FAST_DTM = TQI_REQ_FAST_SEL

else TQI_REQ_FAST_DTM = TQI_REQ_FAST_DTM (n-1) + C_TQ_DELTA_MAX_COMP

endif

TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

"convergence_to_high_dynamic"

on entry: LV_TQ_DYN_IGA = 1
STATE_TQ_DYN_IGA = "convergence"
FAC_DYN_DRIV = 1
during: FAC_DYN_DRIV = FAC_DYN_DRIV - FAC_DYN_DRIV_DEC
FAC_DYN_DRIV = MIN(FAC_DYN_DRIV,1)
TQI_REQ_FAST_DTM = (FAC_DYN_DRIV
* (TQI_BAS - TQI_REQ_FAST_SEL)
+ TQI_REQ_FAST_SEL)
TQI_REQ_FAST_DTM = MAX(TQI_REQ_FAST_DTM,0)
TQI_REQ_SLOW_DTM=TQI_REQ_SLOW_SEL

on exit: FAC_DYN_DRIV=1

“torque_reserve”

on entry: LV_TQ_DYN_IGA = 1
STATE_TQ_DYN_IGA = “tq_reserve”
during: TQI_REQ_FAST_DTM = TQI_REQ_FAST_SEL
TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

default case: “high_dynamic_mode”

on entry: LV_TQ_DYN_IGA = 1
STATE_TQ_DYN_IGA = “high_dyn”
during: TQI_REQ_FAST_DTM = TQI_REQ_FAST_SEL
TQI_REQ_SLOW_DTM = TQI_REQ_SLOW_SEL

end

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_TQ_DELTA_ECO_MOD	1	0...7FFF H	0...1023.97	0.03125	Nm
Maximum torque difference before transition to state ‘economy_mode’					
C_TQ_DELTA_MAX_COMP	1	0...7FFF H	0...1023.97	0.03125	Nm
Maximum torque delta for compensation in state ‘conditional_transition_mode’					
LC_DTM_ENA	1	0,1H	0,1	1	-
Enables DTM					

D.45.2 Coordination single cylinder fuel cut-off (Torque based pattern calculation)

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_TQ_SCC_REQ	V/O	0..1H	0, 1	1	-
Logical variable for enabling pattern fuel cut-off					
LV_TQ_SCC_INH	V/O	0..1H	0, 1	1	-
Logical variable for inhibit pattern fuel cut-off					

Input data:

LV_TQ_DYN_SCC	STATE_TQ_DYN_IGA	LV_ES	LV_ST

FUNCTION DESCRIPTION:

General information:

This part coordinates, if fast fuel cut-off via torque based pattern calculation is allowed or not.

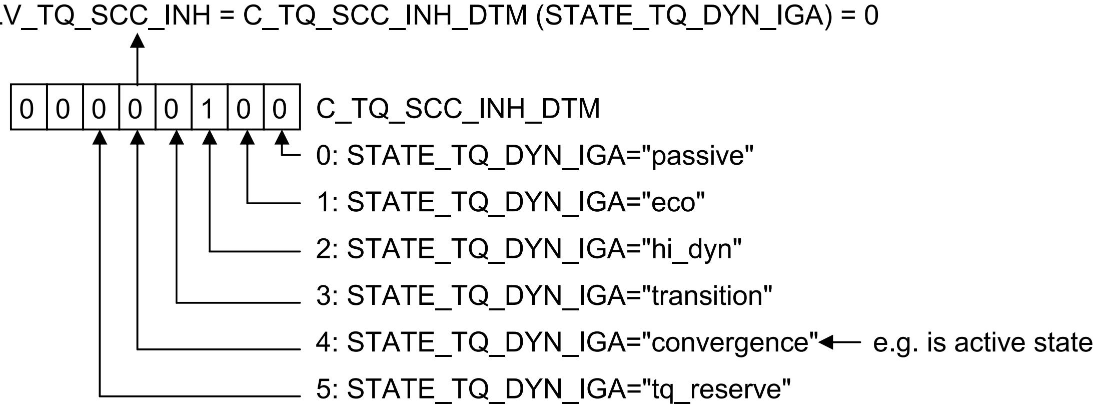
Description:

The current revision contains the following functionality (test state):

Assignment of SCC to certain states of the ignition intervention manager; SCC can only be active in the choosen states.

This is realized by a byte, where LV_TQ_SCC_INH is result of the bit of this byte (C_TQ_SCC_INH_DTM) with index STATE_TQ_DYN_IGA.

LV_TQ_SCC_INH = C_TQ_SCC_INH_DTM (STATE_TQ_DYN_IGA) = 0



Single cylinder cut-off depending on STATE_TQ_DYN_IGA is switched off at FFH for C_STATE_TQ_DYN_IGA. In this case it is authorized only by LV_TQ_DYN_SCC.

Application conditions:

Initialisation: at deactivation

LV_TQ_SCC_INH = 1

Recurrence: 10ms

Activation: at every engine state except LV_ST or LV_ES

Deactivation: -

Formula section:

LV_TQ_SCC_REQ = LV_TQ_DYN_SCC

If C_TQ_SCC_INH_DTM = FFh

then

LV_TQ_SCC_INH = NOT(LV_TQ_DYN_SCC)

else

LV_TQ_SCC_INH = BIT(STATE_TQ_DYN_IGA) from C_TQ_SCC_INH_DTM

end

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TQ_SCC_INH_DTM	8	0 ..FFH	0..255	1	-
Calibration constant for activation pattern calculation depending on DTM state for ignition intervention					

D.45.3 Coordination lambda mode

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_LAMB_TQ_CTL	V / O	0 , 1 H	0 , 1	1	-
Logical output to switch to lambda calculation torque controlled (if equal to 1)					
LV_TQ_LAMB_INH	V / O	0 , 1 H	0 , 1	1	-
Logical output to inhibit torque based lambda setpoint calculation					
LV_TQ_LAMB_ENA	V / O	0 , 1 H	0 , 1	1	-
Logical output to switch on torque based lambda setpoint calculation					
STATE_TQ_DYN_LAMB	V	0 ... FF H	0 .. 255	1	-
State dynamic torque manager lambda intervention					

Input data:

LV_HOM_RUN			

FUNCTION DESCRIPTION:

Application conditions:

Activation: LV_HOM_RUN = 1

Deactivation: LV_HOM_RUN = 0

Initialization: at reset

LV_LAMB_TQ_CTL = 0

LV_TQ_LAMB_INH = 0

LV_TQ_LAMB_ENA = 0

Recurrence: 10 ms

Formula section:

STATE_TQ_DYN_LAMB = C_STATE_TQ_DYN_LAMB

Case **STATE_TQ_DYN_LAMB** of

0: ”Lambda setpoint calculation depending on torque request”

LV_LAMB_TQ_CTL = 0
LV_TQ_LAMB_INH = 0
LV_TQ_LAMB_ENA = 0

1: “Lambda setpoint calculation switch on unconditionally”

LV_LAMB_TQ_CTL = 0
LV_TQ_LAMB_INH = 0
LV_TQ_LAMB_ENA =1

2: “Lambda setpoint calculation inhibited”

LV_LAMB_TQ_CTL = 0
LV_TQ_LAMB_INH = 1
LV_TQ_LAMB_ENA = 0

3: “Lambda setpoint unlimited torque controlled”

LV_LAMB_TQ_CTL = 1
LV_TQ_LAMB_INH = 0
LV_TQ_LAMB_ENA = 0

else default case: ”Lambda setpoint calculation depending on torque request”

LV_LAMB_TQ_CTL = 0
LV_TQ_LAMB_INH = 0
LV_TQ_LAMB_ENA = 0

end

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
C_STATE_TQ_DYN_LAMB	1	0 ... FF H	0 ... 255	1	-
Manual setting of state dynamic torque manager lambda intervention					

D.46 Torque Monitoring Limitation and Coordination

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_REQ_SLOW	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Limited torque request for slow path					
TQI_REQ_FAST	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Limited torque request for fast path					
TQI_SP_SLOW	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque set-point for slow torque path					
TQI_SP	V / O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Setpoint indicated engine torque					
TQI_FAST	V / O	8000 ... 7FFFH	-1024...1023.97	0.03125	Nm
Indicated engine torque fast request					
TQI_BAS_COR_CH	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Corrected indicated basic engine torque at active torque lead with TQI_ADD_CH_FIL					
FAC_TQI_REQ_FL	V	0...FFFFH	0...1.999969	30.51E-6	-
Factor for requested torque for fast path for full load					

Input data:

TQI_REQ_SLOW_DTM	TQI_REQ_FAST_DTM	TQI_REQ_LIM_SLOW	TQI_REQ_LIM_FAST
TQ_DIF_P_D_SLOW_IS	TQ_DIF_P_D_FAST_IS	TQI_BAS	TQ_DIF_AJ
TQ_ADD_REQ_ACT_FIL_CH	EFF_LAMB_AV	EFF_LAMB_BAS_COR	LV_HOM_AFS_ACT
LV_HOM_RUN	TQ_ADD_MIN_CH	LV_TQ_ADD_MIN_CH_ACT	LV_IGK

General information:

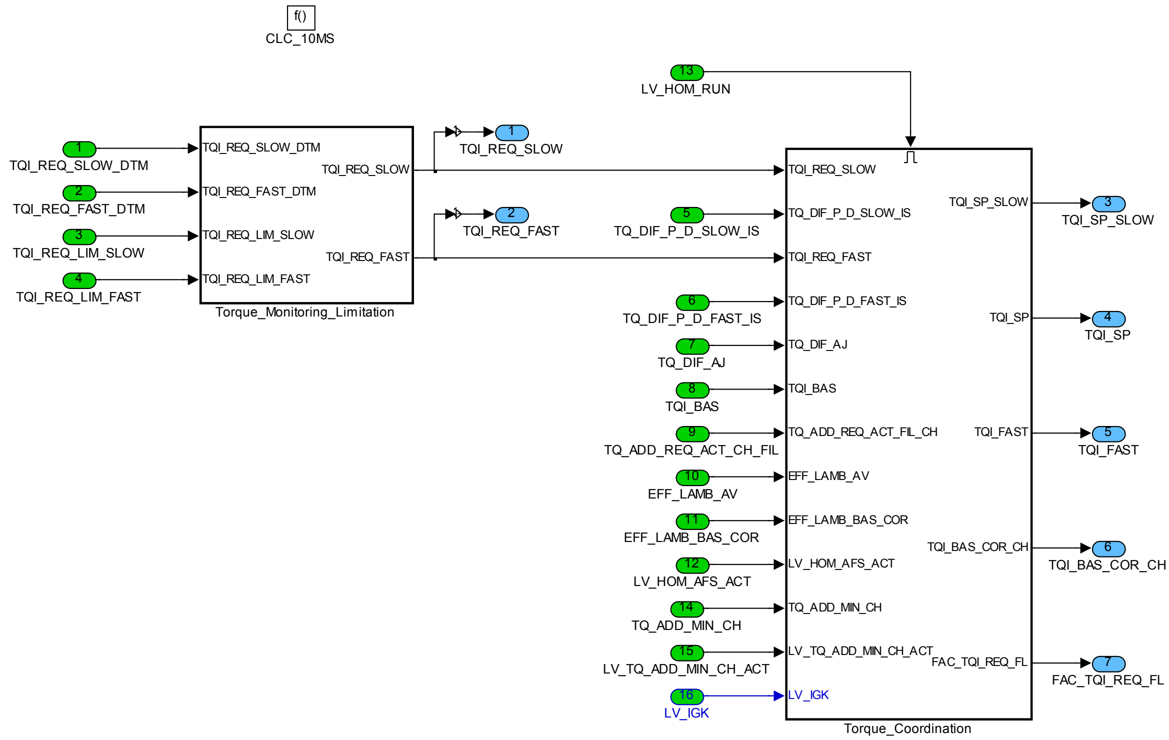
The Torque Coordination module contains two functionalities. The torque requests for coordination must be limited by the monitoring system at the latest point in sequential order to its interface to torque realization.

Application conditions:

Recurrence: 10msec

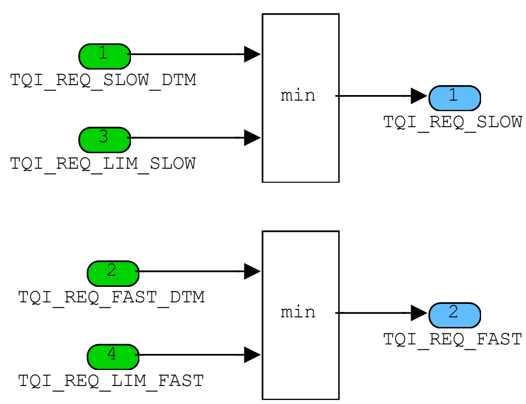
For activation / deactivation check the chapters.

Signal flow diagram:



D.46.1 Torque Monitoring Limitation

Signal flow diagram:



Formula section:

TQI_REQ_SLOW = min(TQI_REQ_SLOW_DTM, TQI_REQ_LIM_SLOW)

TQI_REQ_FAST = min(TQI_REQ_FAST_DTM, TQI_REQ_LIM_FAST)

Application conditions:

Initialisation: all zero

Recurrence: 10msec

Activation: at every engine state

D.46.2 Torque Coordination

FUNCTION DESCRIPTION:

General information:

In case of homogeneous combustion, the module “Torque coordination” coordinates slow and fast torque requests which are realized by the air path and ignition (also fuel cut-off and lambda possible).
The output value TQI_SP_SLOW of the slow path contains the request from the idle speed controller.
TQI_SP is the target torque in homogeneous combustion which has to be realized by the fast torque path. It contains additional torque request from idle speed controller and anti jerk controller. Both have the highest priority and connat be overrun by any torque request.

In case of full load a lambda efficiency quotient is applied to the fast path as a correction factor to prevent ignition angle retardation at full load conditions.

The torque setpoint C_TQI_SP_MAN can be set manually by application tool if the logical constant LC_TQI_SP_MAN_ACT = 1.

It is possible to limit the output TQI_SP to ensure that a minimum torque reserve is applied, this ensures correct catalyst heating.

When the ignition is turned off, the injection is stopped, this can result in a earlier spark angle (via EFF_SCC < 1), and due to the wall film continued combustion with a resulting engine speed increase. In order to eliminate this effect, the torque setpoint is set to 0 when the ignition is off.

Application conditions:

Initialisation: at reset:

TQI_SP = TQI_FAST = TQI_REQ_FAST = 0 Nm

TQI_SP_SLOW = TQI_REQ_SLOW = 0 Nm

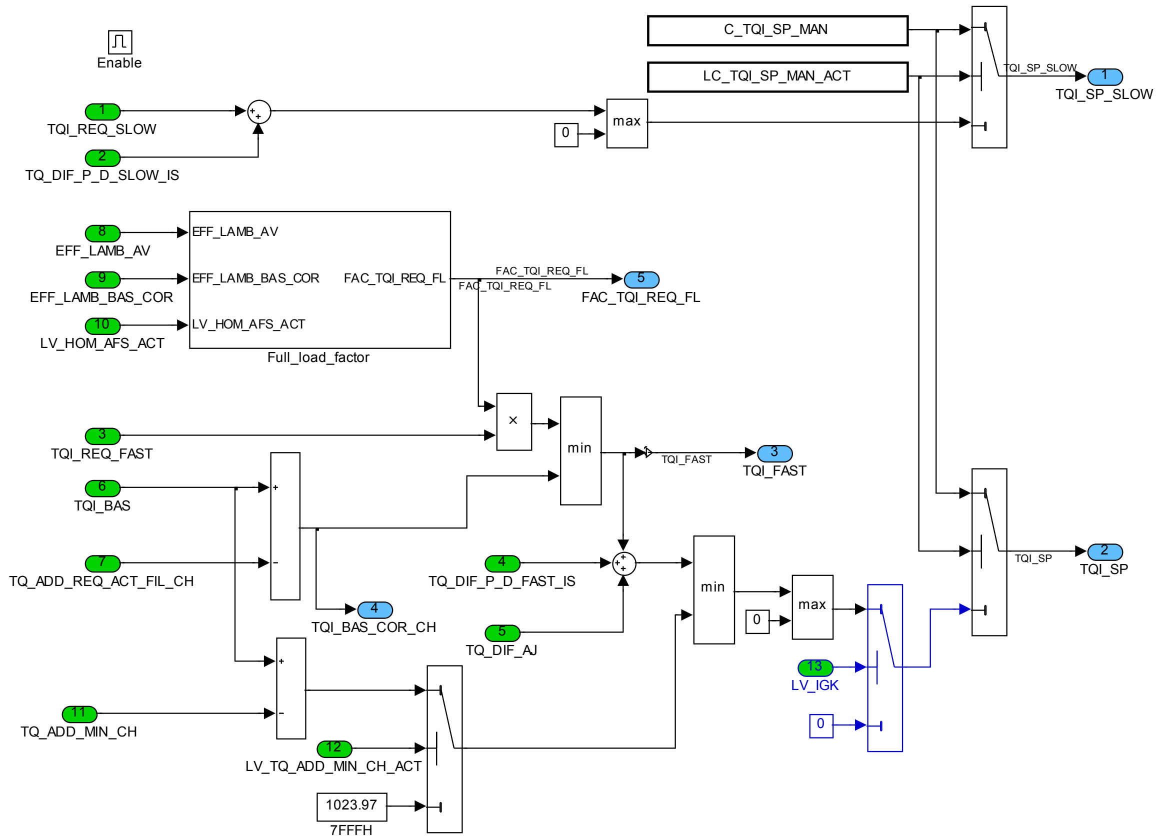
FAC_TQI_REQ_FL = 1

Recurrence: 10 ms

Activation: LV_HOM_RUN = 1

Deactivation: LV_HOM_RUN = 0

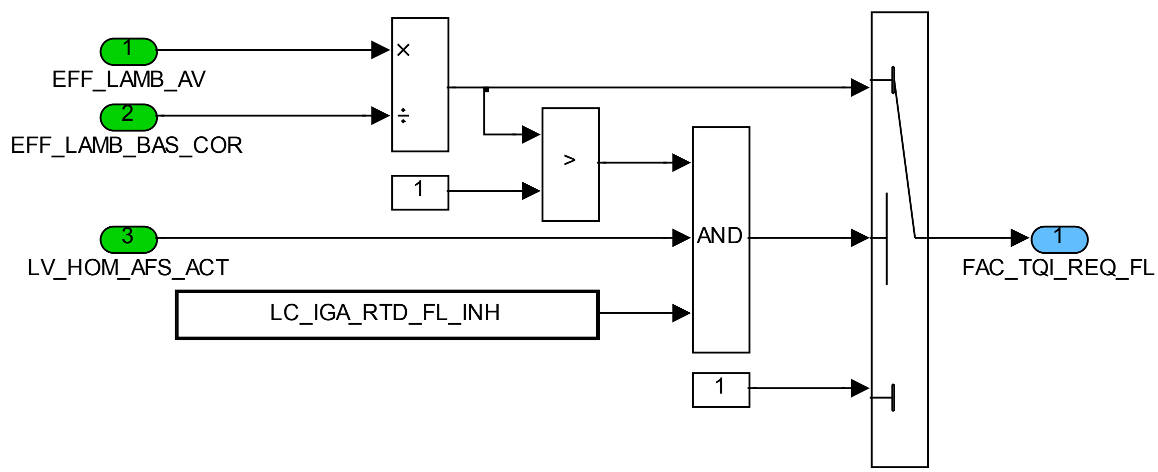
Signal flow diagram:



D.46.3 Full Load Factor

If full load is active the torque request for fast path needs to be corrected to have no ignition intervention at full load.

Signal flow diagram:



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQI_SP_MAN	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Manual torque setpoint request by application tool					
LC_TQI_SP_MAN_ACT	1	0...1 H	0...1	1	-
Logical constant manual torque active					
LC_IGA_RTD_FL_INH	1	0...1 H	0...1	1	-
Logical constant for inhibit full load correction					

D.47 Anti Jerk Controller drivetrain dependent parameters

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_AJ	O / V	8000...7FFF H	-2...1.999939	61.035E-6	Nm/rpm
Factor for converting engine speed deviation into torque correction signal					
FAC_TQ_DE_AJ	O / V	0...FFFF H	0...8	1.2207E-4	Nm/rpm
Feedback gain of engine speed model					
FAC_INV_VEH_MASS	O / V	0...FFFF H	0 ... 0.99998	1.53 E-5	rpm/Nm
Inverse vehicle mass coefficient					
N_DIF_AJ_THD_AC	O / V	8000...7FFF H	-32768...32767	1	rpm
Engine speed deviation threshold for activation anti jerk function at acceleration					
N_DIF_AJ_THD_DEACC	O / V	8000...7FFF H	-32768...32767	1	rpm
Engine speed deviation threshold for activation anti jerk function at deceleration					

Input data:

N	LV_AT	GEAR	LV_ES

FUNCTION DESCRIPTION:

This module includes the anti jerk controller and activation parameters that need to be adapted to the drivetrain configuration. Within this module it is possible to define several configurations as for example AT and MT choosen via variant coding or it could also only contain one variant to reduce calculation time and "dead" maps.

Application conditions:

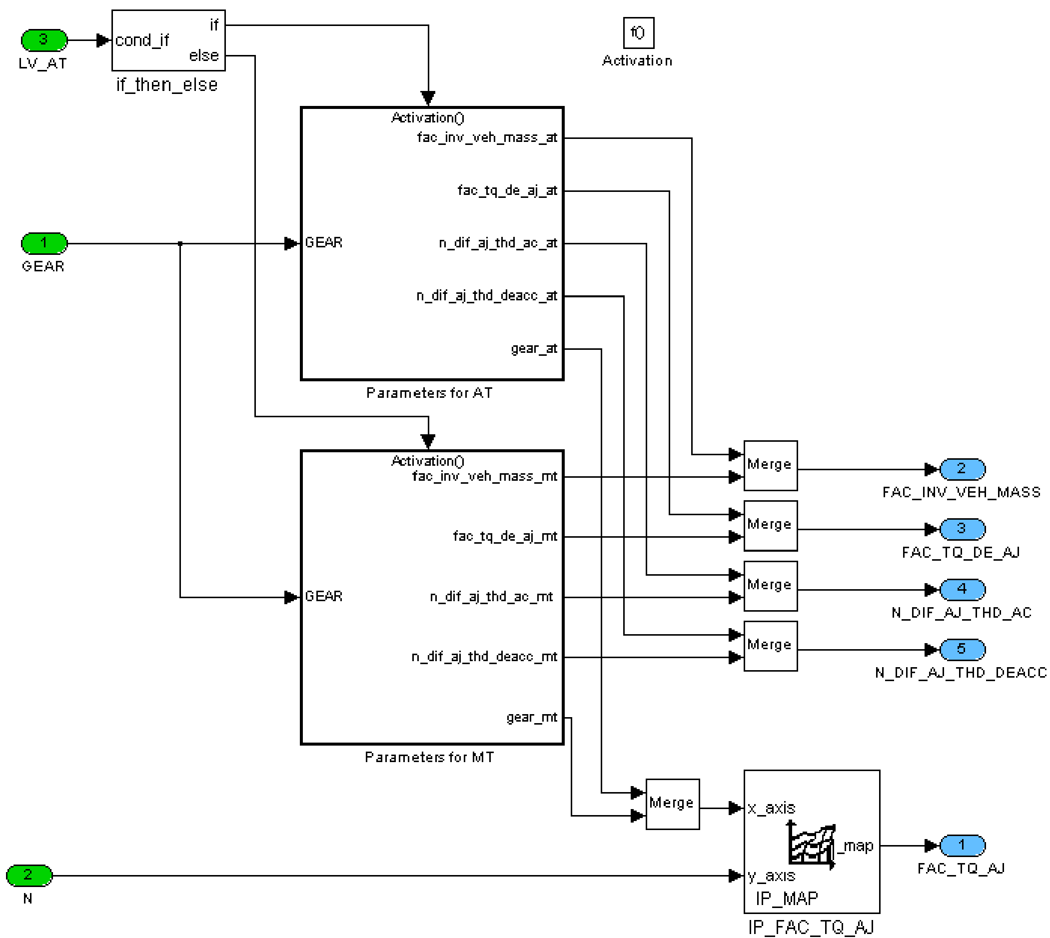
Initialisation: at reset all outputs = 0

Recurrence: 10 ms

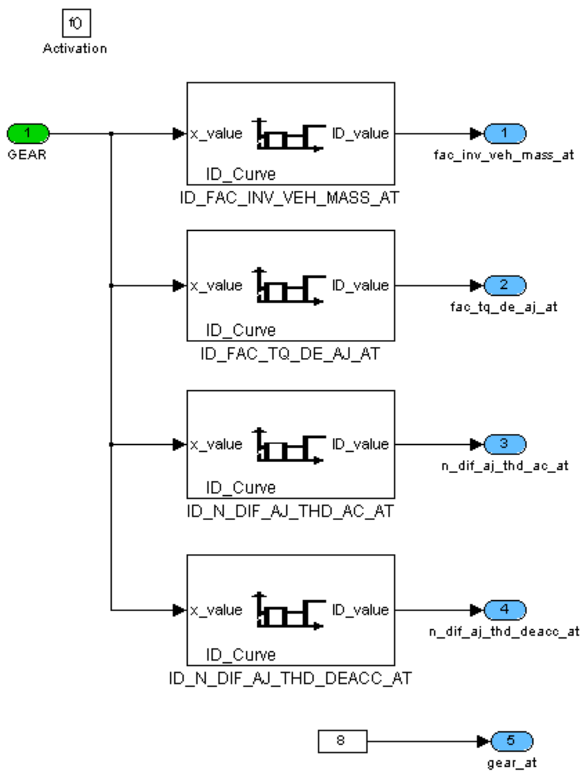
Activation: LV_ES == 0

Deactivation: LV_ES == 1

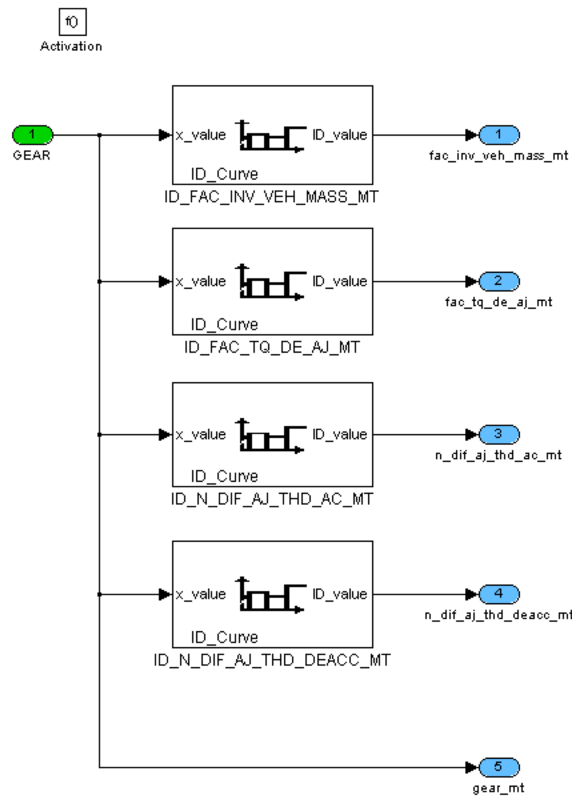
Signal flow diagram:



Block Parameters for AT



Block Parameters for MT



Description:

This project specific solution allows variant coding for manual and automatic transmission with a maximum of seven gears (R,N,1-5). Since R is treated as the 1st gear from gear ratio detection it is possible to put in *eight* gears using seven breakpoints.

For an AT the last breakpoint (8, see Signalflow diagram) from IP_FAC_TQ_AJ is reserved for calibration. Therefore for this variant FAC_TQ_AJ depends only on engine speed and not on gear. Note, that it is only necessary to have anti jerk intervention for locked converter.

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
ID_N_DIF_AJ_THD_AC_MT	8x1	8000...7FFF H	-32768...32767	1	rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Engine speed deviation threshold for activation anti jerk function at acceleration for manual transmission					
ID_N_DIF_AJ_THD_DEACC_MT	8x1	8000...7FFF H	-32768...32767	1	rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Engine speed deviation threshold for activation anti jerk function at deceleration for manual transmission					
ID_N_DIF_AJ_THD_AC_AT	8x1	8000...7FFF H	-32768...32767	1	rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Engine speed deviation threshold for activation anti jerk function at acceleration for automatic transmission					
ID_N_DIF_AJ_THD_DEACC_AT	8x1	8000...7FFF H	-32768...32767	1	rpm
LDPM_GEAR_1_DRVB	8	0 .. FF H	0 .. 255	1	-
Engine speed deviation threshold for activation anti jerk function at deceleration for manual transmission					
ID_FAC_INV_VEH_MASS_AT	8x1	0...FFFF H	0 ... 0.99998	1.53 E-5	rpm/Nm
LDPM_GEAR_1_DRVB	8	0...FF H	0...255	1	-
Inverse vehicle mass equivalent for automatic transmission					
ID_FAC_INV_VEH_MASS_MT	8x1	0...FFFF H	0 ... 0.99998	1.53 E-5	rpm/Nm
LDPM_GEAR_1_DRVB	8	0..FF H	0...255	1	-
Inverse vehicle mass equivalent for manual transmission					
ID_FAC_TQ_DE_AJ_MT	8x1	0...FFFF H	0...8	1.2207E-4	Nm/rpm
LDPM_GEAR_1_DRVB	8	0...FF H	0...255	1	-
Torque deviation compensation factor for AJ_N_DIF_REF determination for manual transmission					
ID_FAC_TQ_DE_AJ_AT	8x1	0...FFFF H	0...8	1.2207E-4	Nm/rpm
LDPM_GEAR_1_DRVB	8	0...FF H	0...255	1	-
Torque deviation compensation factor for AJ_N_DIF_REF determination for automatic transmission					
IP_FAC_TQ_AJ	8x8	0...FFFF H	-2...1.999939	61.035E-6	Nm/rpm
LDP_N_32_IP_FAC_TQ_AJ	8	0...FF H	0...8160	32	rpm
LDPM_GEAR_1_DRVB	8	0...FF H	0...255	1	-
Torque conversion factor for anti-jerk funktion					

D.48 Anti Jerk Controller Activation Conditions (Appl. Inc.)

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_AJ_INH	O / V	0 ... 1H	0. 1	1	-
LV anti-jerk function inhibited					
LV_AJ_ENA	O / V	0 ... 1H	0. 1	1	-
LV anti-jerk function enabled without initialization					
LV_PAS_AJ_INP	O / V	0 ... 1H	0. 1	1	-
LV anti-jerk function passive input (1->0 leads to anti jerk initialization)					

Input data:

LV_ES	LV_FCUT_IND	LV_GS_ACT	LV_AT
LV_ASR_ACT	LV_MSR_ACT	LV_ERR_CRK	LV_ERR_IV[NC_CYL_NR]
LV_ERR_MWSS	LV_ERR_CAM_IN_i	LV_ERR_VS	LV_MIS_STATE_A
LV_MIS_STATE_B			

FUNCTION DESCRIPTION:

General information:

The anti jerk controller must be activ only at certain activation conditions. Within this chapter the activation conditions are fitted to project specific needs, depending on engine and drivetrain configuration.

The inputs requesting anti jerk can be separated into static signal that do not require a reinitialization and signals where a reinitialization of the reference engine speed model and the anti jerk controller is necessary.

There is also the possibility the inhibit anti jerk via static signals.

Application conditions:

Initialisation: at reset all outputs = 0

Recurrence: 10 ms

Activation: LV_ES = 0

Deactivation: LV_ES = 1

Formula section:

LV_PAS_AJ_INP = LV_FCUT_IND or LV_GS_ACT or LV_ASR_ACT or LV_MSR_ACT

If LV_ERR_CRK = 1 **or**
LV_ERR_IV[NC_CYL_NR] = 1 **or**
LV_ERR_MWSS = 1 **or**
LV_ERR_CAM_IN_i =1 **or**
LV_ERR_VS =1 **or**
LV_MIS_STATE_A =1 **or**
LV_MIS_STATE_B = 1
Then LV_AJ_INH = 1Else LV_AJ_INH = LV_AT AND not(LC_AJ_ENA_AT)
Endif

LV_AJ_ENA = C_MODE_AJ_ACT

Description:

The inputs for LV_PAS_AJ_INP mark a possible reasen for jerk, where the controller may not be active and where it needs an initialization at reactivation. For LV_AJ_INH it's possible to activate AJ for AT (bridged converter).

Calibration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
LC_AJ_ENA_AT	1	0 . 1 H	0 . 1	1	-
Logical constant for anti jerk function to inhibit the AT influence on the activation conditions					
C_MOD_AJ_ACT	1	1H 2 H	‘AJ_ON’ ‘AJ_OFF’	1	-
Manual mode selection for LV_AJ_ACT_ENA calculation					

D.49 Anti Jerk Controller

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TQ_DIF_AJ	O / V	8000...7FFFH	-1024...1023.968	31.25E-3	Nm
Torque output for anti jerk					
N_DIF_AJ	O / V	8000...7FFFH	-32768...32767	1	rpm
engine-speed deviation at jerking vehicle					
N_REF_AJ	V	8000...7FFFH	-32768...32767	1	rpm
Reference engine-speed for anti-jerk function					
N_DIF_REF_AJ	V	8000...7FFFH	-32768...32767	1	rpm
limited reference engine-speed deviation for anti-jerk function					
N_DIF_REF_LPF_AJ	V	8000...7FFFH	-512...511.98	0.015625	rpm
Low-pass filtered N_DIF_REF_AJ for AJ function					
TQ_FAST_CLU	V	8000...7FFFH	-1024...1023.968	31.25E-3	Nm
Fast torque request at clutch					
TQ_MDL_DE_AJ	V	8000...7FFFH	-1024...1023.968	31.25E-3	Nm
torque deviation compensation for AJ					
N_DIF_REF_AJ_INTER	V	8000...7FFFH	-32768...32767	1	rpm
Reference engine-speed deviation for anti-jerk function without limitation					
LV_ACT_AJ	O / V	0 . 1H	0 . 1	1	-
LV_anti-jerk controller activated					
LV_PAS_AJ	V	0 . 1H	0 . 1	1	-
LV anti-jerk controller passive					
LV_REQ_AJ	V	0 . 1H	0 . 1	1	-
LV anti-jerk controller requested					
LV_INI_AJ	V	0 . 1H	0 . 1	1	-
LV initialize model reference engine speed and low pass filter					

Input data:

N	N_DIF_AJ_THD_AC	N_DIF_AJ_THD_DEACC	LV_DT
LV_PAS_AJ_INP	LV_REQ_ISC	LV_AJ_INH	LV_AJ_ENA
TQI_FAST	TQ_LOSS	FAC_TQ_AJ	FAC_TQ_DE_AJ
FAC_INV_VEH_MASS	LV_ES		

FUNCTION DESCRIPTION:

General information:

The anti-jerk function damps jerk effects of the vehicle at sudden acceleration or deacceleration. Vehicle jerking results in an engine-speed oscillation. The deviation of the actual engine-speed to a 'not jerking' modelized engine-speed is the indicator for a jerking vehicle.

The anti jerk function damps oscillation of the powertrain at vehicle acceleration or deacceleration by an torque intervention in phase with the calculated engine-speed deviation N_DIF_AJ.

According to torque structure calculation a spark intervention is applied.

The anti-jerk function consists of three main parts:

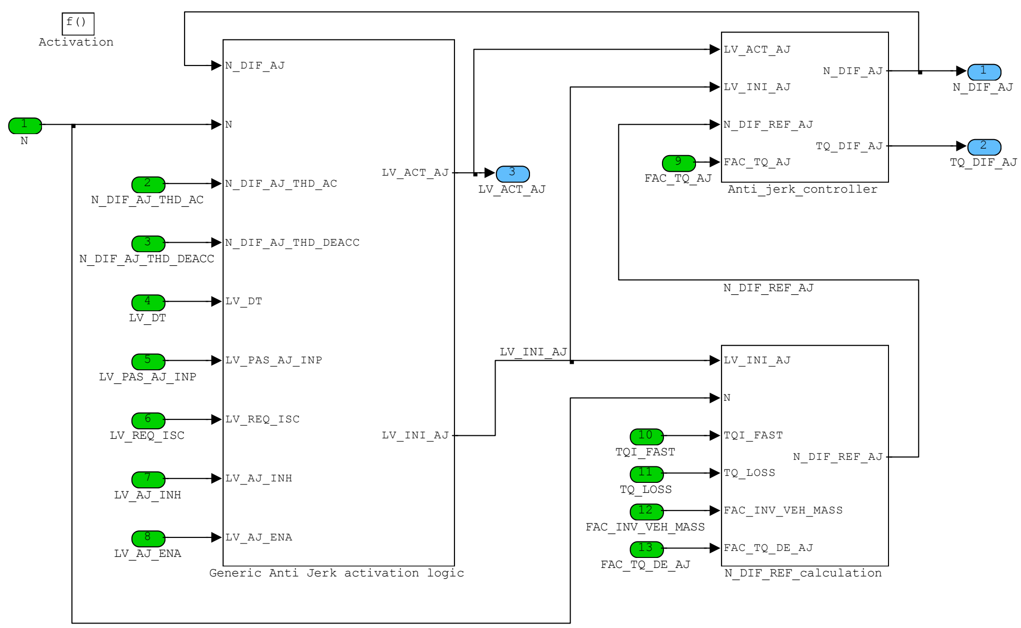
- Generic activation and trigger condition
- Determination of engine-speed oscillation N_DIF_REF_AJ
- Anti-jerk controller (with determination of N_DIF_AJ and TQ_DIF_AJ)

The engine speed oscillation is calculated from a reference engine speed signal and the real engine speed. From this deviation the output TQ_DIF_AJ for spark influence is produced.

Application conditions:

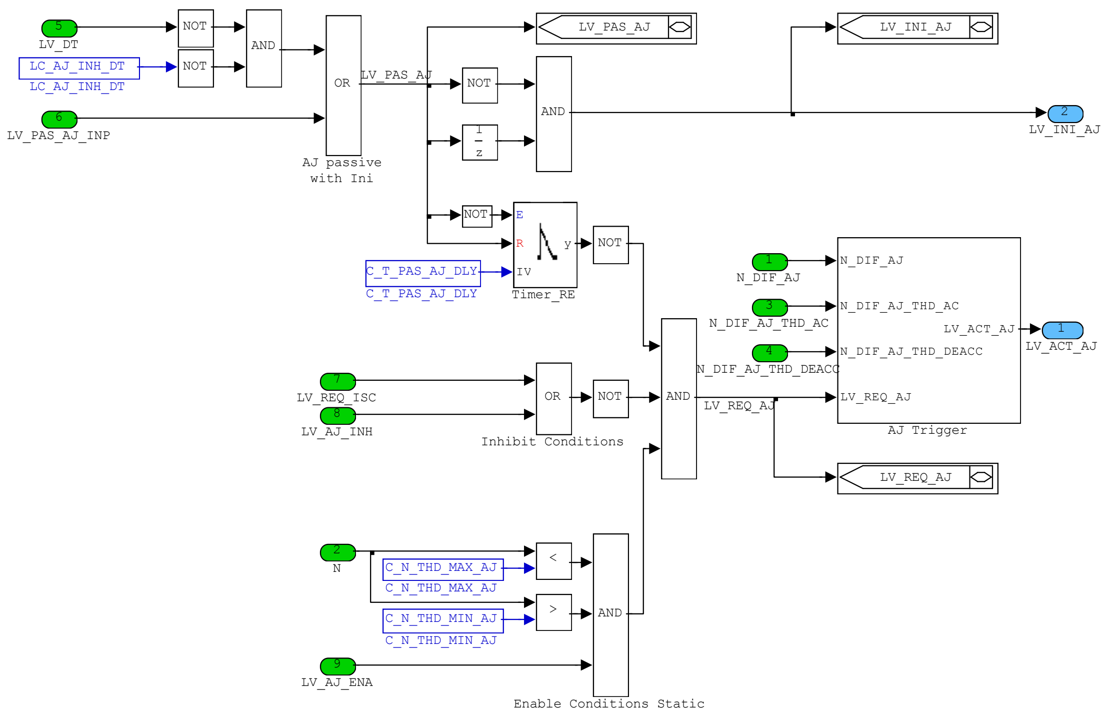
Initialisation:	at reset all outputs and intermediate values = 0
	at reset LV_INI_AJ = 1 for one recurrence
Recurrence:	10ms
Activation:	LV_ES == 0
Deactivation:	LV_ES == 1

Signal flow diagram:



D.49.1 Anti Jerk Activation and Trigger

Signal flow diagram:



Description:

The generic activation provides two inputs to mark an anti jerk request. Active LV_DT or LV_PAS_AJ_INP (input for project specific activation) lead to an initialisation of the controller. This input should be used when the modelled engine speed does not represent the real engine speed, e.g. at open drive train.

Another input , LV_AJ_ENA enables the trigger algorithm without initialisation and should be used for static signals.

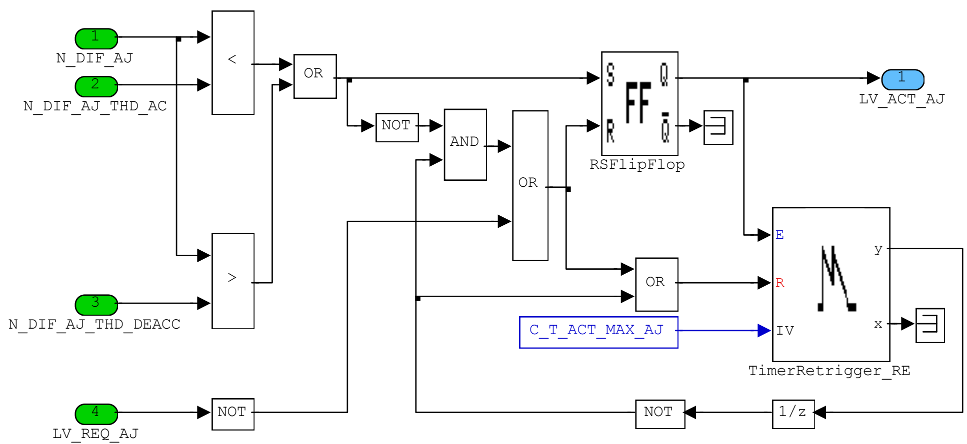
Inhibit conditions (LV_REQ_ISC, LV_AJ_INH) can prevent an anti jerk intervention. Also the anti jerk intervention can only be active within a calibrateable engine speed range.

AJ Trigger Algorithm

If an anti jerk request LV_REQ_AJ is active, the controller is activated by a trigger event depending on the deviation N_DIF_AJ of reference engine speed and real engine speed. Once the trigger occured the anti jerk activation lasts until the timer C_T_ACT_MAX_AJ is out or the LV_REQ_AJ gets inactive. If LV_REQ_AJ stays active while the timer has run out, the trigger can get active immediatly again. There should be no delay, if LV_REQ_AJ and the trigger condition are true.

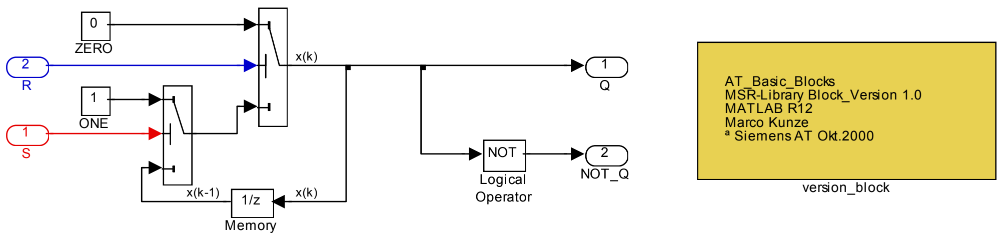
Anti Jerk is triggered immediatly at the first wave exceeding the thresholds.

Signal flow diagram:

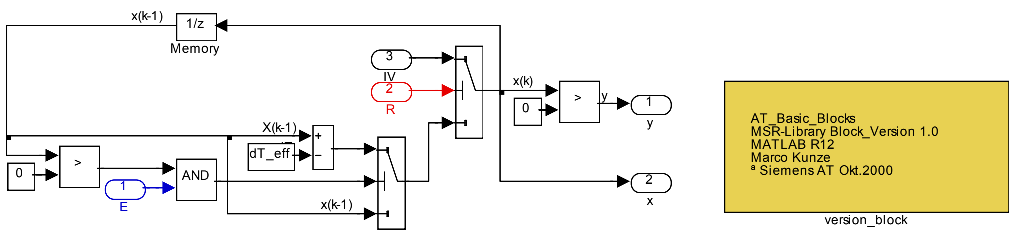


Important for RSFlipFlop and Timer: reset (R) priority is higher then set (S/E) priority.
See next page for details.

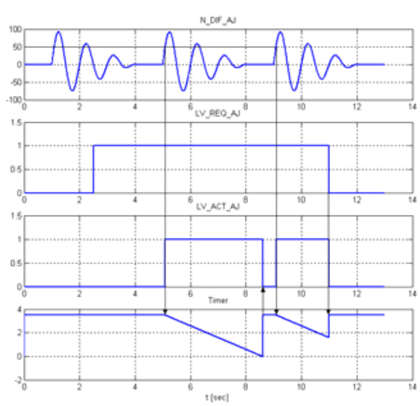
RSFlipFlop:

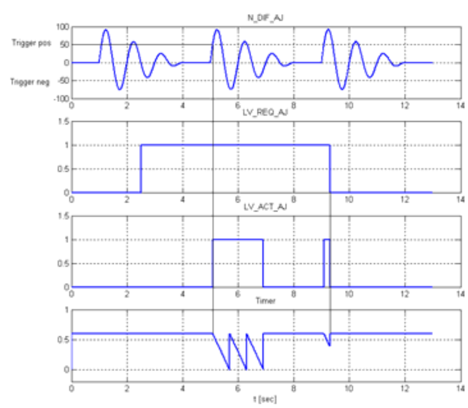


TimerRetrigger:



Signals:



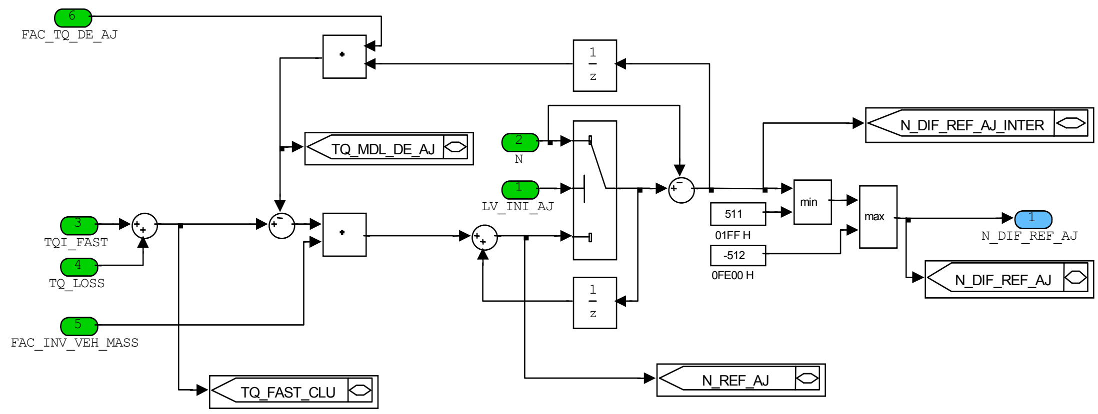


Case1: The Timer lasts longer then a jerk signal.
The LV_ACT_AJ is reset by the timer or missing LV_REQ_AJ.

Case2: The Timer lasts shorter than a jerk signal.
If the trigger condition ist fulfilled immediate restart of the timer is done. So there is no interuption of the LV_ACT_AJ.

Determination of Engine-Speed Oscillation N_DIF_REF_AJ

Signal flow diagram:



Application conditions:

Initialisation: at reset or LV_INI_AJ == 1

⇒ N_DIF_REF_AJ = 0; N_REF_AJ = N

The engine speed oscillation at jerking vehicle is determined with help of the real measured and a modelized (not jerking) engine speed (N_REF_AJ).

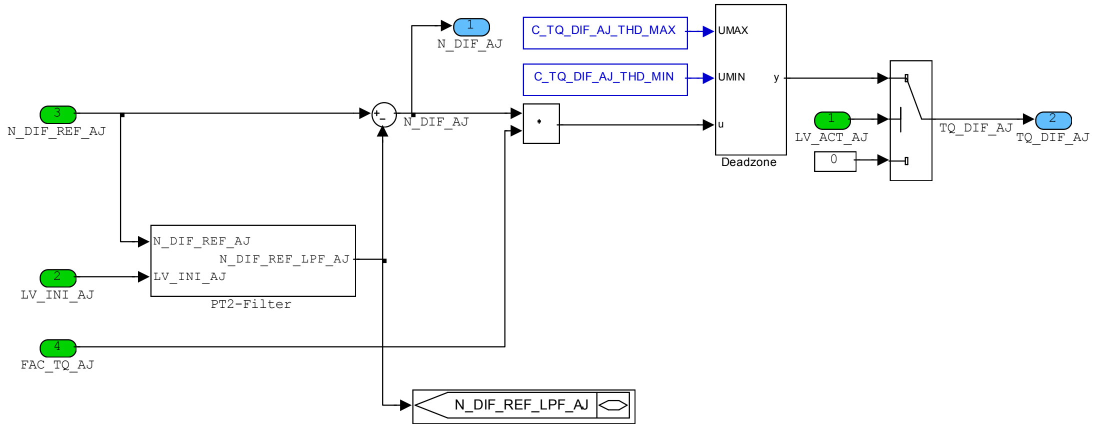
For this purpose a modelized engine speed N_REF_AJ is calculated from the actual torque request at clutch TQ_FAST_CLU = TQI_FAST + TQ_LOSS by integration. The actual vehicle-mass and moment of inertia is taken into consideration by the factor FAC_INV_VEH_MASS. This factor includes also the update rate of 10 ms.

The remaining deviation N_DIF_REF_AJ is used to determine the high frequent part of engine-speed oszillation N_DIF_AJ.

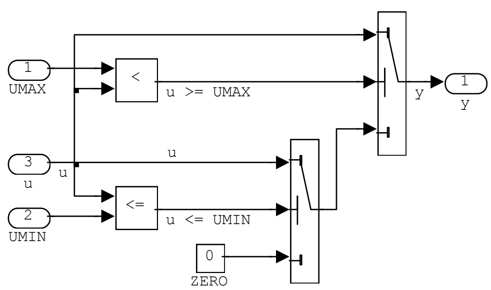
A proportional feedback loop with N_DIF_REF_AJ as input, corrects the 'input torque' TQ_FAST_CLU to adjust the reference non-jerking engine-speed to the measured engine speed N, performing an 'adaptation' of changed real-world conditions.

D.49.2 Anti-Jerk Controller

Signal flow diagram:



Dead zone:



For anti-jerk torque intervention only the high-frequent part of the engine speed oscillation is used, to damp the unpleasant oszillations (e.g. in range: 2 up to 10 Hz) but not to limit the acceleration capability of the vehicle.

The high-pass filter of N_DIF_REF_AJ is realized in two steps:

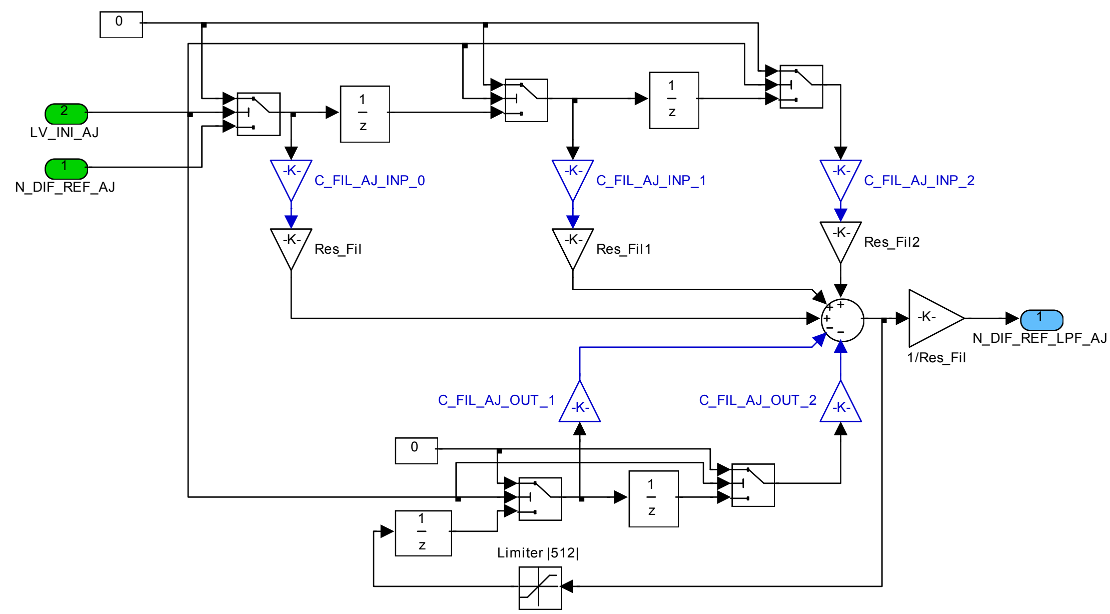
First step - low pass filtering of N_DIF_REF_AJ ⇒ N_DIF_REF_LPF_AJ represents the deviation of modelized and real engine-speed

Second step - The difference of INPUT and OUTPUT of the low-pass filter represents the high-pass filtered engine-speed oscillation which is the basis for anti-jerk torque intervention.

Proportional to this difference N_DIF_AJ the damping torque output TQ_DIF_AJ is determined.

D.49.2.1 Recursive low pass filter of second order

Signal flow diagram:



Application conditions:

Initialisation: at reset or LV_INI_AJ == 1 ⇒ N_DIF_REF_LPF_AJ = 0

Description:

The main problem for digital PT2 filter is the influence of the signal quantization to the behavior of the filter output. Bad signal quantisation can risk the stabillity of the filter or create a signal offset between filter in- and output.

For the N_DIF_REF_AJ filter the input and output is fixed to the same resolution of 1 rpm.

To minimize the error due to quantization at internal calculation, the resolution is increased by the factor 64. This increased resolution is included in the filter constants (INP_0; INP_1, INP_2).

To receive an output with resolution of 1 rpm the filtered value N_DIF_REF_LPF_AJ is divided by 64 before the determination of N_DIF_AJ is done.

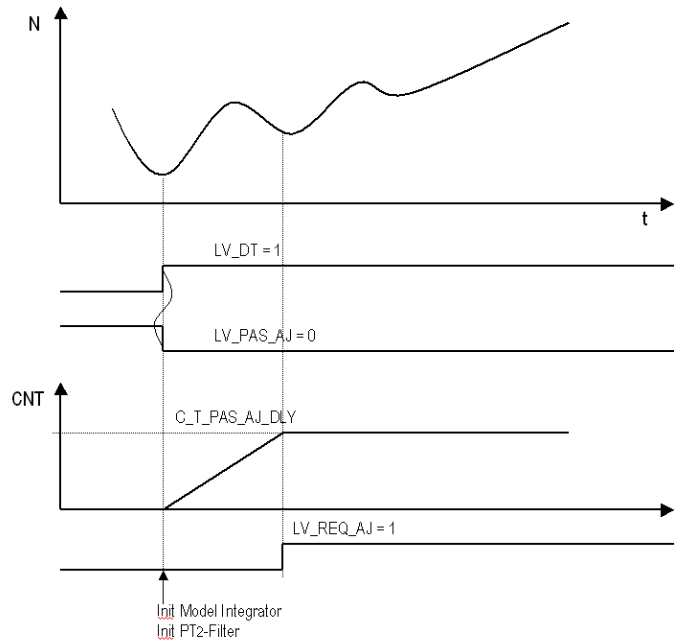
The feedback signal N_DIF_REF_AJ, with increased internal resolution, is limited to +/- 512 rpm.

Formula section

N_DIF_REF_LPF_AJ(n)	=	C_FIL_AJ_INP_0 *	N_DIF_REF_AJ(n)
	+	C_FIL_AJ_INP_1 *	N_DIF_REF_AJ(n-1)
	+	C_FIL_AJ_INP_2 *	N_DIF_REF_AJ(n-2)
	-	C_FIL_AJ_OUT_1 *	N_DIF_REF_LPF_AJ(n-1)
	-	C_FIL_AJ_OUT_2 *	N_DIF_REF_LPF_AJ(n-2)

To minimize the disturbance due to gear shift intervention, the PT2 filter in- and output is initialized with ´0´, if the gear-shift intervention is finished (LV_PAS_AJ transition 1 ⇒ 0; detected in LV_INI_AJ).

The time-delay C_T_PAS_AJ_DLY must have passed before the AJ algorithm can be released (LV_REQ_AJ = 1) again. See the module 'Anti Jerk activation conditions' for more information.



Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FIL_AJ_INP_0	1	C000...4000H	-0.015625..0.015	9.53E-7	-
Filter constant 0 applied on input value of second order low pass filter					
C_FIL_AJ_INP_1	1	C000...4000H	-0.015625..0.015	9.53E-7	-
Filter constant 1 applied on input value of second order low pass filter					
C_FIL_AJ_INP_2	1	C000...4000H	-0.015625..0.015	9.53E-7	-
Filter constant 2 applied on input value of second order low pass filter					
C_FIL_AJ_OUT_1	1	8000...7FFF H	-2...1.999939	61.035E-6	-
Filter constant 1 applied in output value of second order low pass filter					
C_FIL_AJ_OUT_2	1	8000...7FFF H	-2...1.999939	61.035E-6	-
Filter constant 2 applied on output value of second order low pass filter					
C_TQ_DIF_AJ_THD_MAX	1	8000...7FFF H	-1024...1023.968	31.25E-3	Nm
Maximum threshold of TQ_DIF_AJ deadband					
C_TQ_DIF_AJ_THD_MIN	1	8000...7FFF H	-1024...1023.968	31.25E-3	Nm
Minimum threshold of TQ_DIF_AJ deadband					
C_T_ACT_MAX_AJ	1	0...FF H	0...25.5	0.1	s
Maximum active time of anti jerk function after activation					
C_T_PAS_AJ_DLY	1	0...FF H	0...2.55	0.01	s
Delay time after closed drivetrain determination for PT2 filter actualisation					
C_N_THD_MAX_AJ	1	0...1FE0 H	0...8160	1	rpm
Maximum engine speed threshold for active anti jerk function					
C_N_THD_MIN_AJ	1	0...1FE0 H	0...8160	1	rpm
Miminum engine speed threshold for active anti jerk function					
LC_AJ_INH_DT	1	0...1 H	0...1	1	-
Logical constant for anti jerk function to inhibit the DT influence on the activation conditions					

D.50 SCC Basic and Actual Efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_SCC_AV	O/V	0...FFFFH	0...1.999969	3.05176E-5	-
Actual efficiency fuel cut-off					
EFF_SCC_BAS	O/V	0...FFFFH	0...1.999969	3.05176E-5	-
basic efficiency fuel cut-off					

Input data:

EFF_SCC_BAS_REQ	LV_EFF_BAS_REQ	NC_CYL_NR	PREV_STATE_IV
SUM_INH_INJ			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_EFF_SCC_BAS_MAN	1	0...FFFFH	0...1.999969	3.05176E-5	-
manually adjusted basic efficiency fuel cut-off					
LC_EFF_SCC_AV_MOD_SEL	1	0...1H	0...1	1	-
Mode selector for EFF_SCC_AV calculation					
LC_ENA_EFF_SCC_BAS_MAN	1	0...1H	0...1	1	-
logical variable for manually adjusting basic efficiency fuel cut-off					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_CYC_ADD_ACT	1	0...1H	0...1	1	-
Activation of calculation of EFF_SCC_AV for two cycles (current and previous)					

D.50.1 General Information

Calculation of EFF_SCC_BAS

EFF_SCC_BAS is the basic efficiency for single cylinder fuel cut-off and is utilized during permanent fuel cut-off interventions (e.g. cylinder bank specific fuel cut-off). With an external request from "Basic fuel cut-off coordination", a requested basic efficiency is applied. A calibrateable constant can be set for the basic efficiency, if enabled.

Calculation of EFF_SCC_AV

EFF_SCC_AV is the efficiency that represents the actual impact of fuel cut-off interventions on the indicated engine torque.

Every fuel cut-off intervention deriving from

- pattern (engine speed limitation, ASR, GS, etc.)
- trailing throttle fuel cut-off

- misfire
- engine stop

is considered.

The calculation of this parameter can be performed in two alternative ways (by means of LC_EFF_SCC_AV_MOD_SEL).

Mode_1 (LC_EFF_SCC_AV_MOD_SEL=0):

The value is calculated as the number of opened injectors at the ***last two or four revolutions*** divided by the number of cylinders NC_CYL_NR or twice NC_CYL_NR. The input value PREV_STATE_IV (byte) defines the status of each injector during the last finished injection cycle. If a cylinder-allocated bit of PREV_STATE_IV is active, the injection was performed for the respective cylinder.

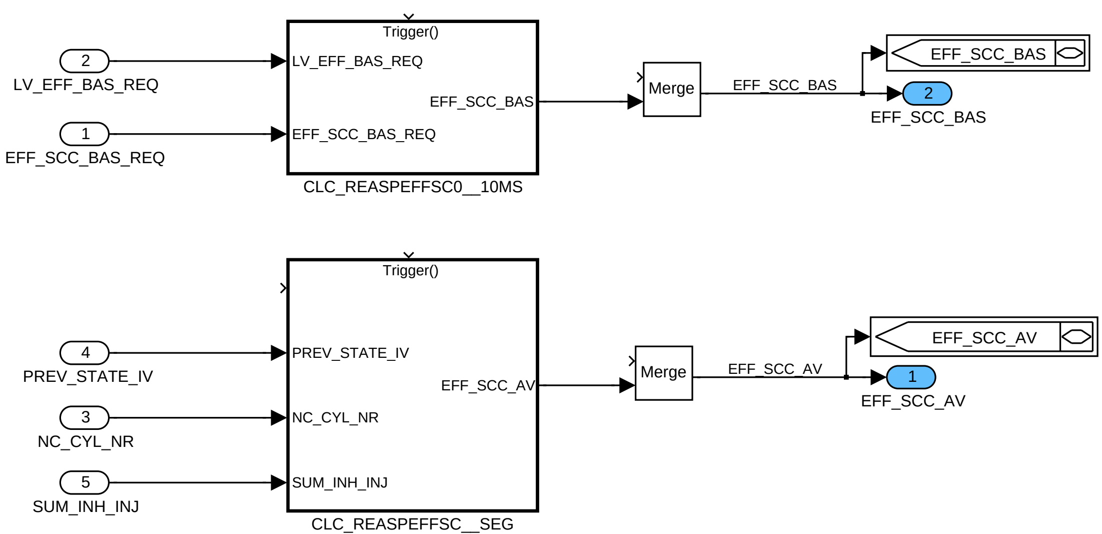
The injection information in PREV_STATE_IV is used to calculate a vertical sum. This sum is stored in SUM_PREV_STATE_IV. The value SUM_PREV_STATE_IV_OLD is the stored vertical sum from the segment n-1. With NC_CYC_ADD_ACT set to 1, the efficiency is calculated as average value over two combustion cycles.

Mode_2 (LC_EFF_SCC_AV_MOD_SEL=1):

The value is calculated as ratio between the number of injected cylinders for the ***next injection cycle*** (NC_CYL_NR - SUM_INH_INJ) and the total number of cylinders NC_CYL_NR. The value SUM_INH_INJ is calculated in the module ‘Cylinder Shut-off’ and represents the number of deactivated injectors for the following injection cycle.

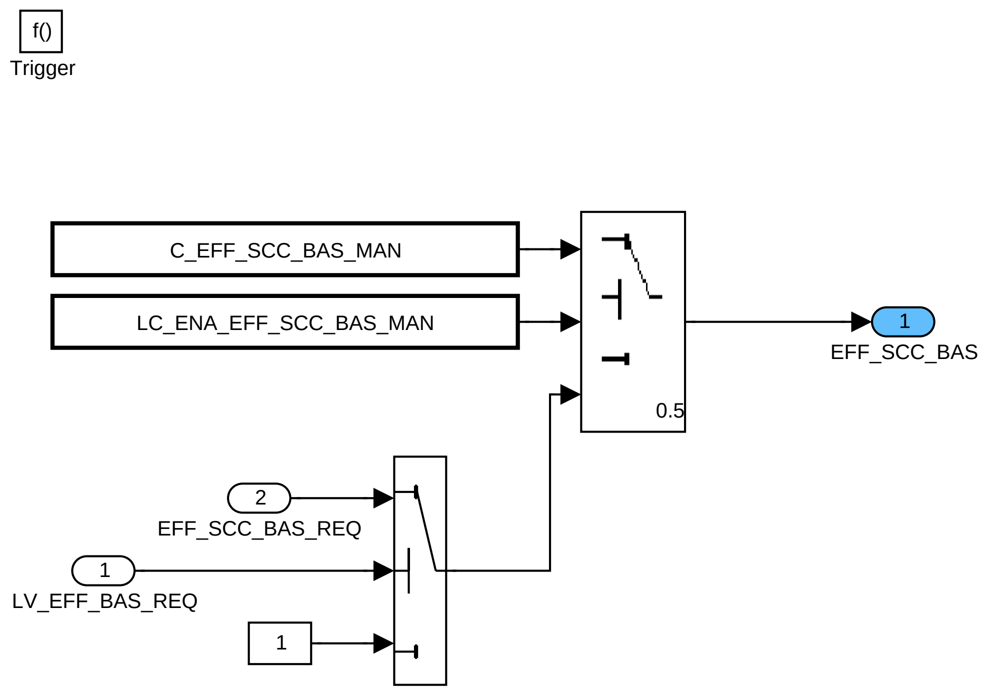
With NC_CYC_ADD_ACT set to 1, the efficiency is calculated as average value over two combustion cycles.

Function Description



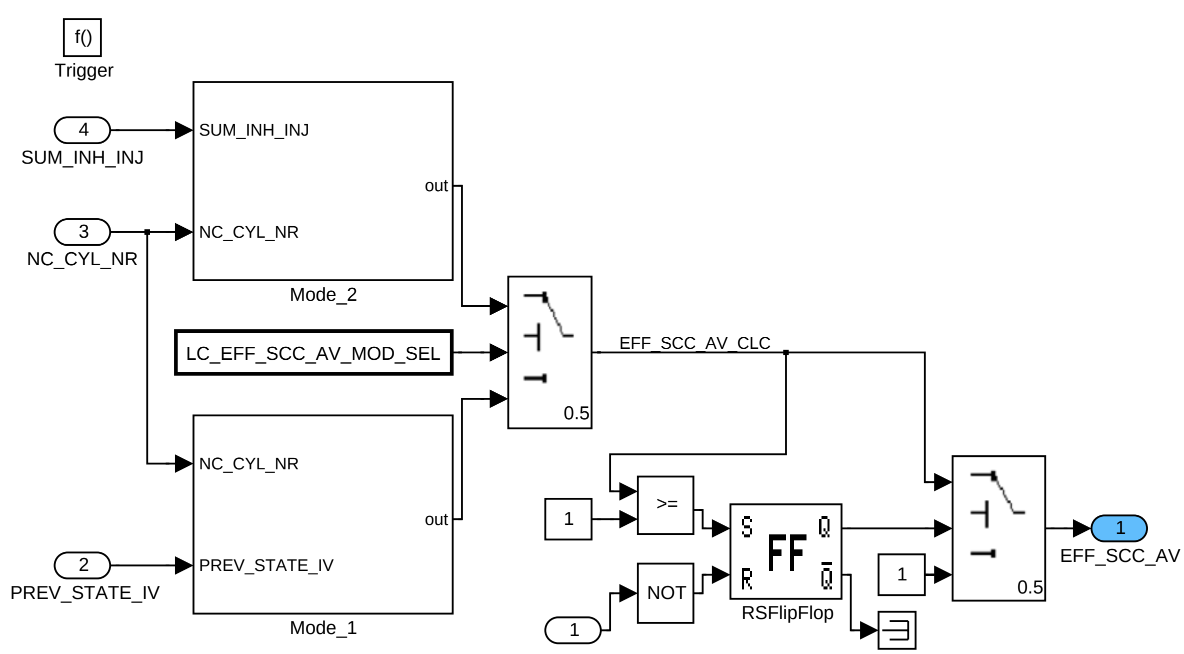
SDA_SRS / SDA 3.1 05−Dec−2003

D.50.1.1 SUBFUNCTION: CLC_REASPEFFSC0__10MS

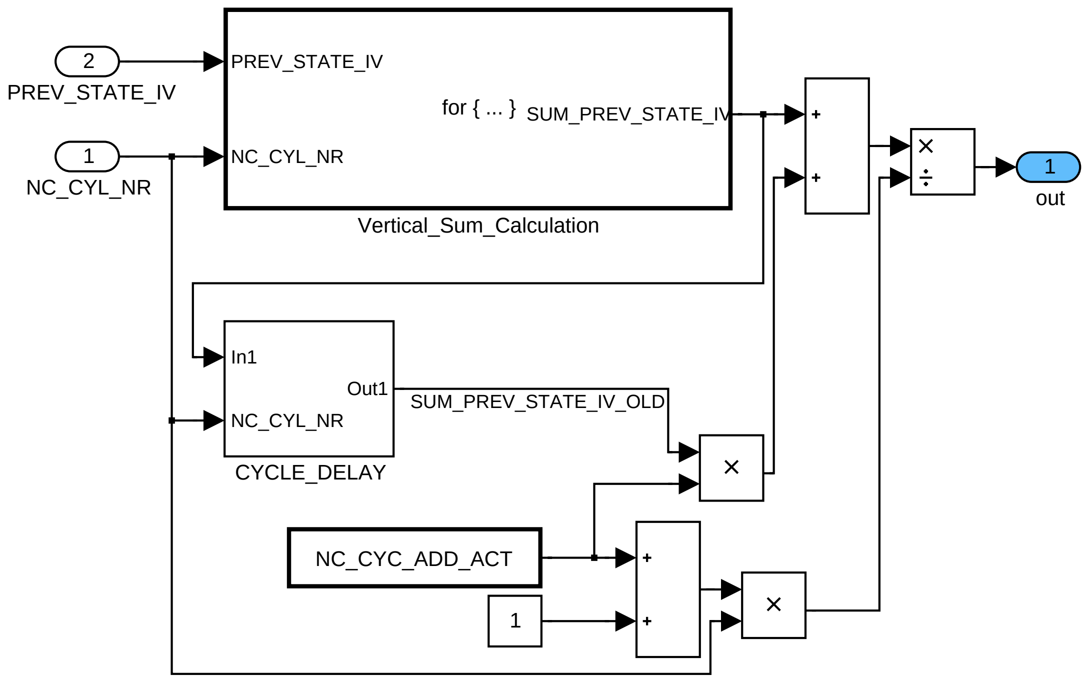


MD02G/CLC_REASPEFFSC0__10MS

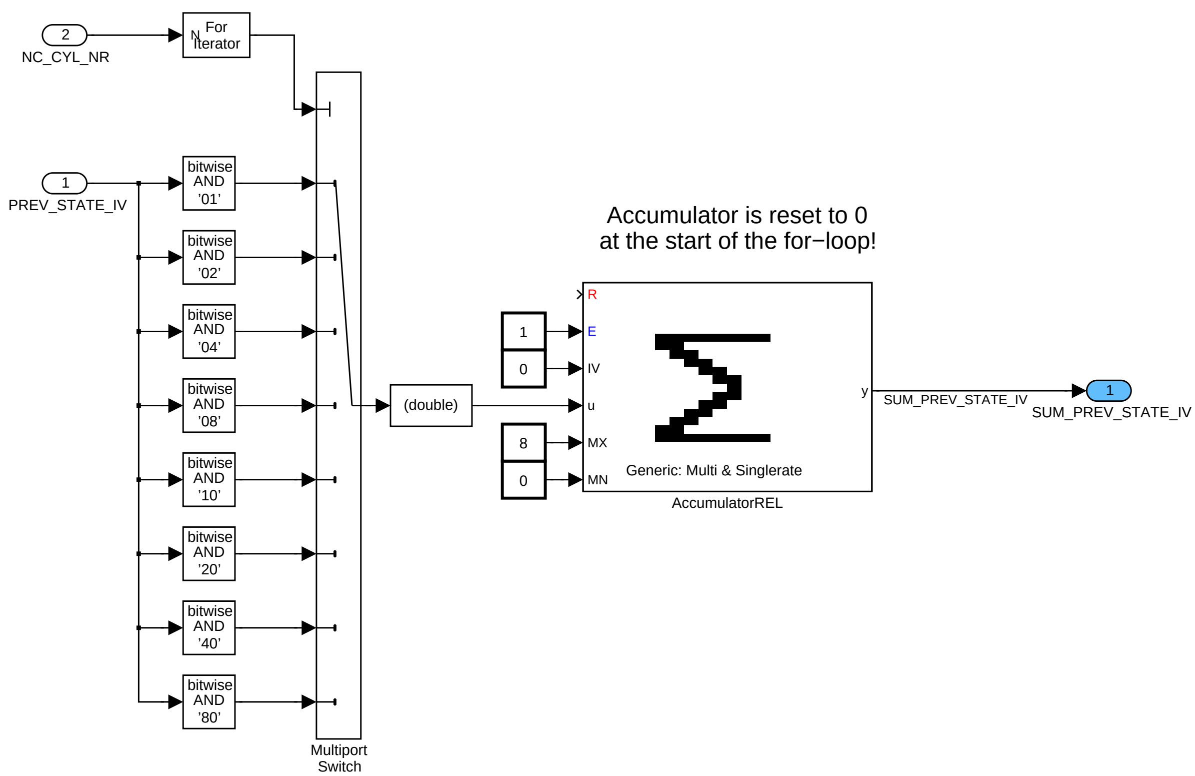
D.50.1.2 SUBFUNCTION: CLC_REASPEFFSC__SEG



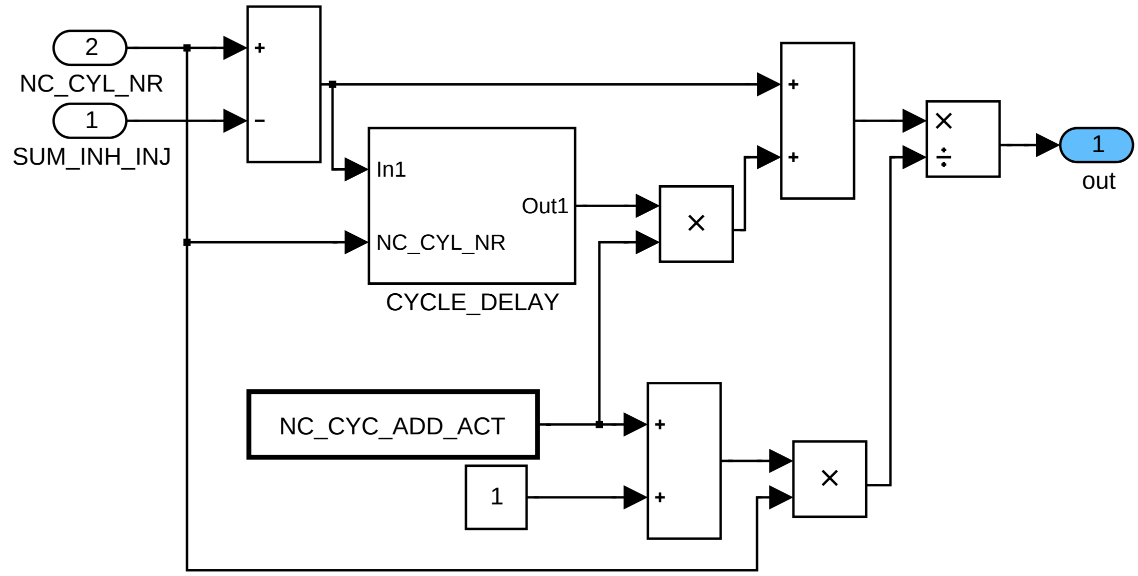
MD02G/CLC_REASPEFFSC__SEG



MD02G/CLC_REASPEFFSC__SEG/Mode_1



MD02G/CLC_REASPEFFSC__SEG/Mode_1/Vertical_Sum_Calculation



MD02G/CLC_REASPEFFSC__SEG/Mode_2

D.51 Basic, Actual, and Minimum Lambda Efficiency

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EFF_LAMB_AV	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Actual lambda efficiency					
EFF_LAMB_BAS_COR	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Basic lambda efficiency					
EFF_LAMB_MIN	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Minimum lambda efficiency					
EFF_LAMB_SP_BAS	O/V	0...FFFFH	0...1.9999	3.0517e-5	-
Basic lambda setpoint efficiency					
LAMB_REF_COR	O/V	0...7FFFH	0...31.999	0.00097656	-
Corrected lambda reference value					
LAMB_REF_ADD_TCO	V	8000...7FFFH	-32...31.999	0.00097656	-
Additive correction of reference lambda value					
LAMB_DIF_AV	V	8000...7FFFH	-32...31.999	0.00097656	-
Actual lambda difference to corrected reference lambda					
LAMB_DIF_BAS	V	8000...7FFFH	-32...31.999	0.00097656	-
Basic lambda difference to corrected reference lambda					
LAMB_DIF_MAX	V	8000...7FFFH	-32...31.999	0.00097656	-
Maximum lambda difference to corrected reference lambda					
LAMB_DIF_SP_BAS	V	8000...7FFFH	-32...31.999	0.00097656	-
Basic lambda setpoint difference to corrected reference lambda					
LAMB_REF	V	0...7FFFH	0...31.999	0.00097656	-
Lambda reference value					

Input data:

LAMB_BAS_COR_MAX	LAMB_BAS_COR_MV	LAMB_SP_BAS_MV	LAMB_SP_MV
MAF_HB	N_32	TCO	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_LAMB_REF_ADD	6	0...7FFFH	0...31.999	0.00097656	-
LDP_TCO_IP_LAMB_REF_ADD	6	0...FEH	-48...142.5	0.75	°C
Additive lambda correction versus coolant temperature					
IP_EFF_LAMB	16	0...FFFFH	0...1.9999	3.0517e-005	-
LDP_LAMB_DIF_IP_EFF_LAMB	16	0...FFFFH	-32...31.999	0.00098	-
Lambda efficiency					
IP_LAMB_REF	6x6	0...7FFFH	0...31.999	0.00097656	-
LDP_N_32_IP_LAMB_REF	6	0...FFH	0...8160	32	1/min
LDP_MAF_HB_IP_LAMB_REF	6	0...FFH	0...1389	5.45	mg/stroke
Reference lambda					

D.51.1 General Information

This module delivers torque efficiencies EFF_LAMB_xxx depending on the difference of the input lambda value (LAMB_BAS_COR_MV, LAMB_SP_MV, LAMB_BAS_COR_MAX and LAMB_SP_BAS_MV) to corrected reference lambda LAMB_REF_COR.

The map IP_EFF_LAMB is used for different inputs for LAMB_DIF, LAMB_DIF_BAS, LAMB_DIF_AV, LAMB_DIF_MAX and LAMB_DIF_SP_BAS.

Function Description

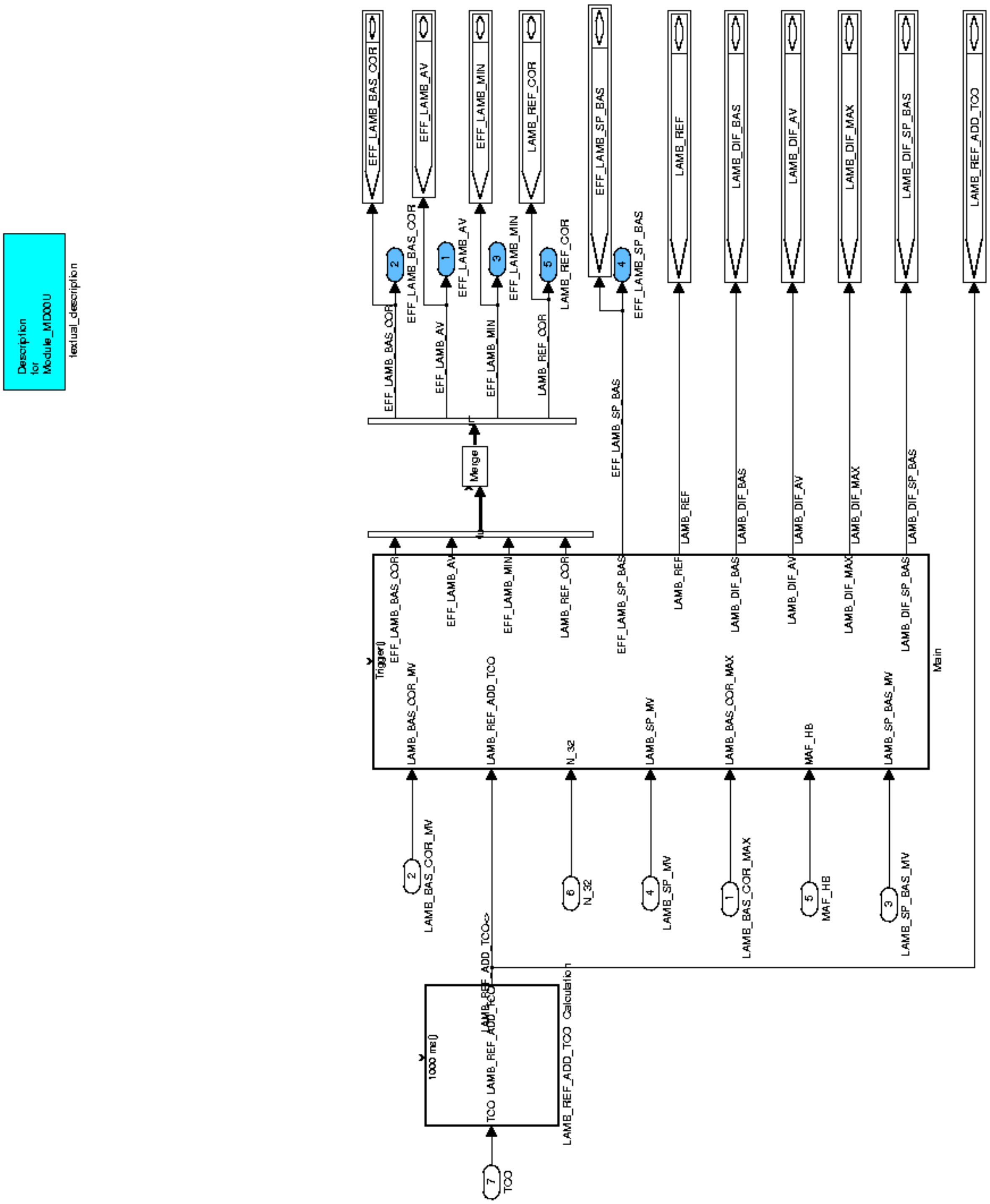


Figure 64 MD00U

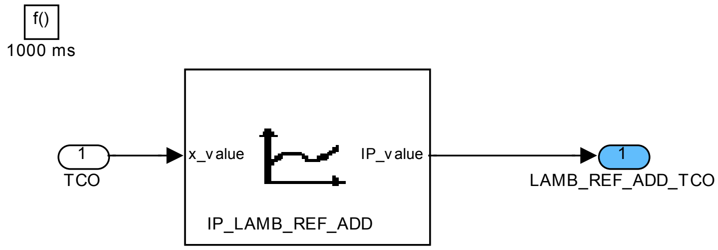


Figure 65 MD00U/ LAMB_REF_ADD_TCO Calculation

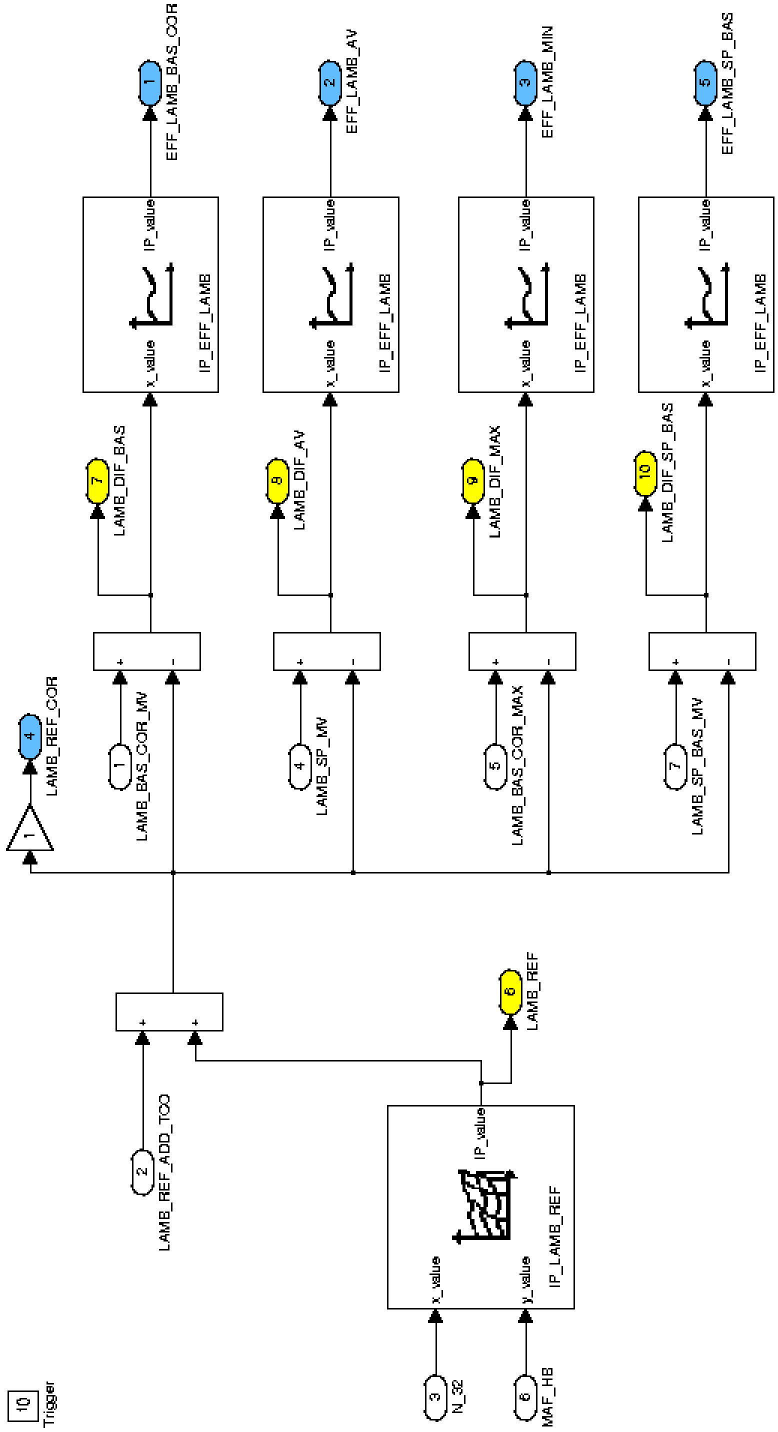


Figure 66 MD00U/ Main

D.52 Actual Engine Torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQE	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Effective engine torque at clutch					
TQI_AV	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Actual indicated engine torque					
TQ_AV	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Actual engine torque static at clutch					
EFF_TOT_AV	V	0...FFFFH	0...1.9999	3.0517e-5	-
Total actual efficiency (in homogeneous mode)					
TQE_DIF	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Difference effective engine torque at clutch					
TQI_AV_HOM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Actual indicated engine torque for homogeneous mode					

Input data:

EFF_IGA_AV	TQI_REF	EFF_EGR_HOM	TQI_AV_S
EFF_LAMB_AV	EFF_SCC_AV	LV_HOM_RUN	LV_S_ACT
N_GRD	TQ_LOSS		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_TQE	1	0...FFH	0...0.249	0.0009764 7	kg*m^2/s
Product of mass moment of inertia of rotating engine parts and PI/30					

D.52.1 General Information

TQI_AV is the actual indicated engine torque produces either in homogeneous or stratified combustion mode. The combustion mode is indicated by the logical variable LV_S_ACT (1 ... stratified combustion; 0 ... homogeneous combustion). TQ_AV is the actual static effective torque at clutch. TQE is the actual effective torque at clutch inclusive inertia.

These values are used for external units (e.g. traction control and gear box) as a feedback signal.

The dynamic torque is derived as follows:

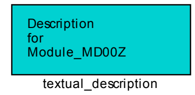
TQE_DIF = J (kg*m^2) * $d\omega/dt$ (1/s^2)

TQE_DIF = J (kg*m^2) * $\pi/30 *$ N_GRD (1/min/s)

TQE_DIF = C_FAC_TQE (kg*m^2*min) * N_GRD (1/min/s)

C_FAC_TQE = J (kg*m^2) * $\pi/30$

Function Description



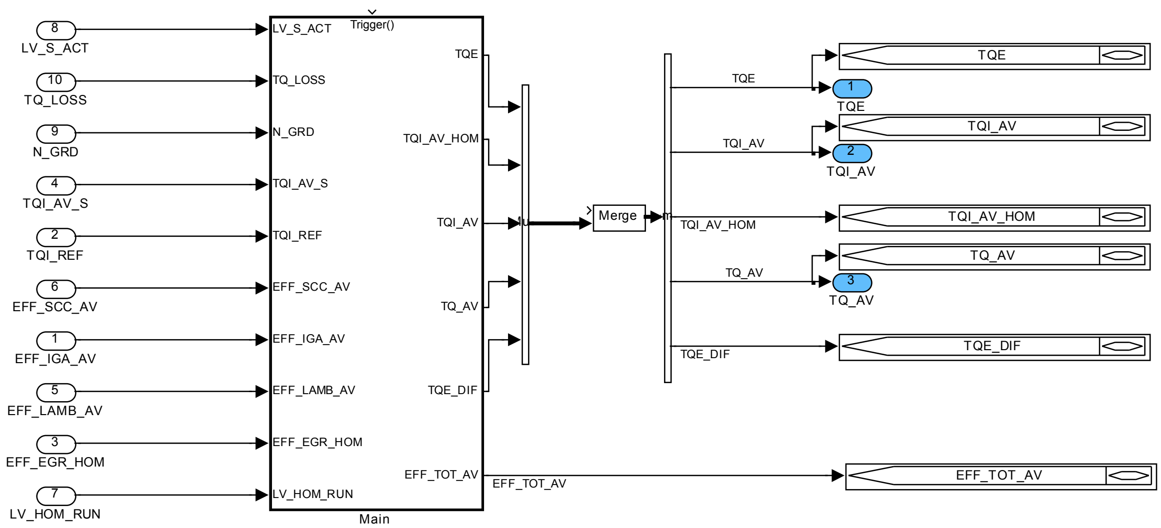


Figure 67 MD00Z

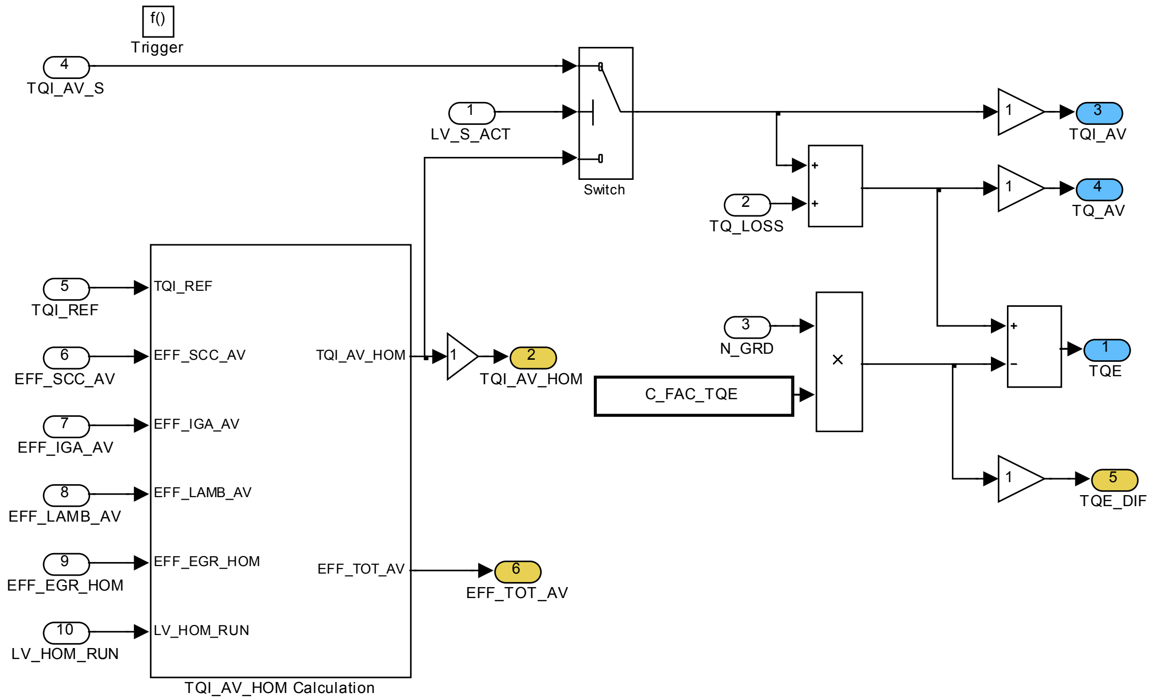


Figure 68 MD00Z/ Main

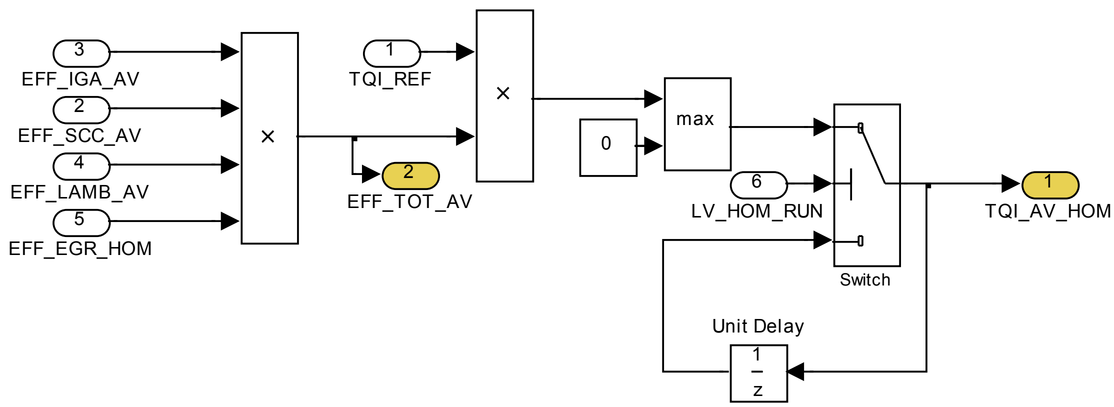


Figure 69 MD00Z/ Main/ TQI_AV_HOM Calculation

D.53 Torque Request for Safety

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
TQI_REQ_LIM_FAST	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Requested torque for safety for fast path					
TQI_REQ_LIM_SLOW	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
Requested torque for safety for slow path					
LV_TQI_LIM	V	0 ... 1H	0 ... 1	1	-
Bit for detected Torque-Error after Delay					
LV_TQ_LIM_INTV	V/O	0 ... 1H	0 ... 1	1	-
Bit for torque intervention for safety active					
SUM_TQI_REQ_LIM	S/V/O	0 ... FFFFH	0 ... 65535	1	-
Event counter of active torque limitations (stored in non volatile memory)					
TQI_DIF_LIM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque offset for recognition torque intervention active					
TQI_REQ_LIM_INC	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque offset for incrementation ramp					
TQI_REQ_LIM_DEC	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque offset for decrementation ramp					
FAC_TQI_DIF_ABS_LIM	V	0...FFH	0 ... 0.9962	0.00391	-
factor for torque offset for recognition torque intervention active					
TQI_REQ_LIM_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Intermediate value 1 for requested torque for safety					
TQI_REQ_LIM_2	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Intermediate value 2 for requested torque for safety					
TQI_REQ_LIM_3	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Intermediate value 3 for requested torque for safety					
TQI_REQ_LIM_4	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Intermediate value 4 for requested torque for safety					
TQI_SP_REL	V	0 ... FFH	0 ... 0.9962	0.00391	-
relative desired torque demand					
LV_ST_DLY	V	0 ... 1H	0 ... 1	1	-
Bit for start phase with off-delay					
T_TQ_LIM_INTV	V	0...FFFFH	0...2621.4	0.04	s
Time counter to set LV_TQ_LIM_INTV = 0 for Off -Delay					
TQI_ABS_DELTA	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
absolute torque overhead					
LV_TQI_ABS_LIM	V	0 ... 1H	0 ... 1	1	-
Bit for detected absolute Torque-Error					
TQI_ABS_LIM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Threshold for absolute Torque-Error					
TQI_DIF_ABS_LIM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
absolute Torque overhead depending on TQI_ABS_LIM					
TQI_DIF_ABS_LIM_1	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Value directly from the maps IP_TQI_DIF_ABS_LIM(_ST)					
TQI_I_LIM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Threshold to start Torque-Integration					
TQI_I_DELTA	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
this Torque will be integrated					
TQI_ABS_DELTA_REL	V	0 ... FFH	0 ... 0.9962	0.00391	-
relative Torque-Delta to absolute Threshold					
TQI_I_ DELTA_REL	V	0 ... FFH	0 ... 0.9962	0.00391	-
relative Torque-Delta to integrator Threshold					
TQI_ABS_ DELTA_REL _MEM	V	0 ... FFH	0 ... 0.9962	0.00391	-
minimal relative Torque-Delta to absolute Threshold in DC					
TQI_ABS_ DELTA_MEM	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
minimal Torque-Delta to absolute Threshold in DC					
TQI_I_ DELTA_REL _MEM	V	0 ... FFH	0 ... 0.9962	0.00391	-
minimal relative Torque-Delta to integrator Threshold in DC					
TQI_I_ DELTA_MEM	V/O	8000...7FFFH	-1024...1023.97	0.03125	Nm
minimal Torque-Delta to integrator Threshold in DC					
LV_TQI_I_LIM	V	0 ... 1H	0 ... 1	1	-
Bit for detected integration Torque-Error					
LV_TQI_DIF_RAMP_ACT	V	0 ... 1H	0 ... 1	1	-
Torque-ramping active					
TQI_I_ DELTA_SUM	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Integral of difference between actual torque and limitation torque					

Input data:

TQI_SP_MON	TQI_AV	N_32	LV_TQ_INC_EXT
LV_TQ_LIM_ACT	LV_ST	TQI_REF_MAX	FAC_TQ_REQ_MON
TQI_EMS			

FUNCTION DESCRIPTION:

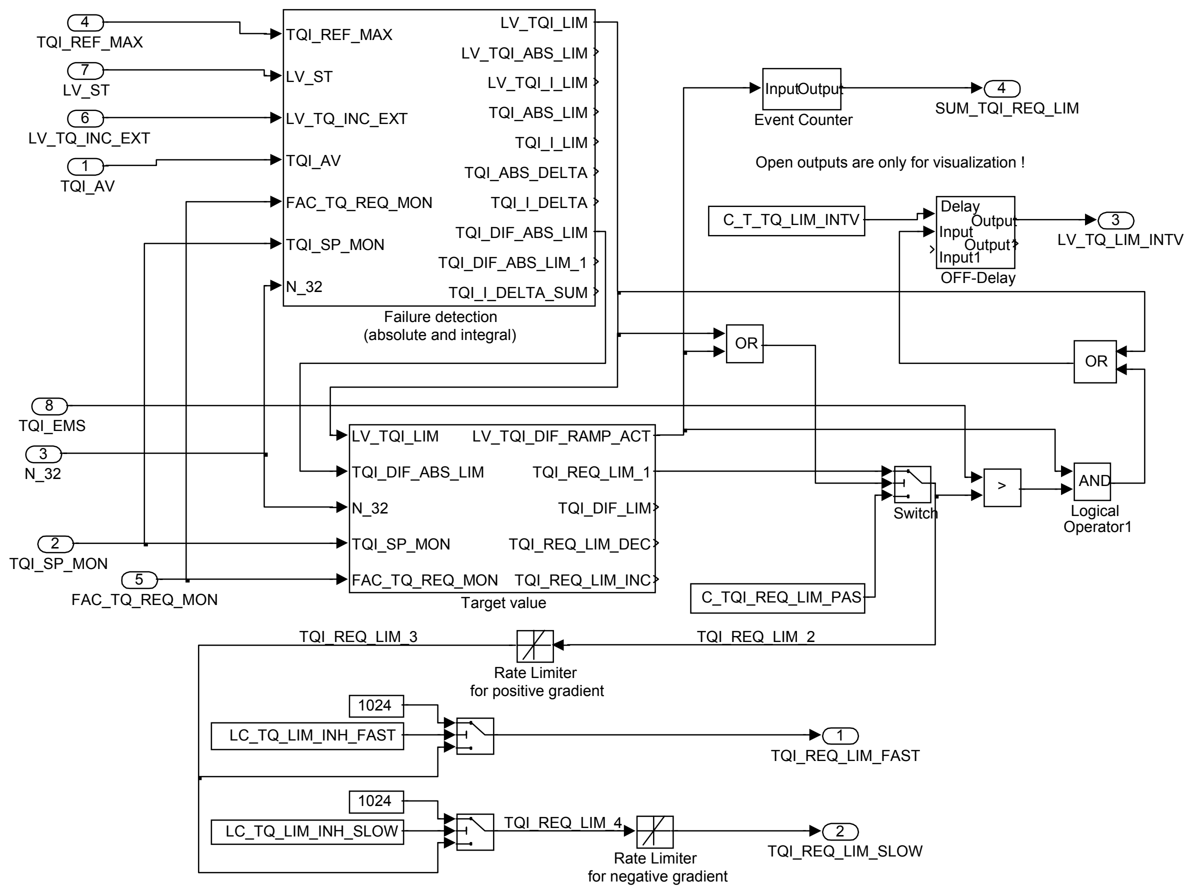
General information:

The overall ETC monitoring concept monitors the torque development of the engine by comparing the actual indicated engine torque TQI_AV_MON with the desired indicated engine torque TQI_SP_MON.

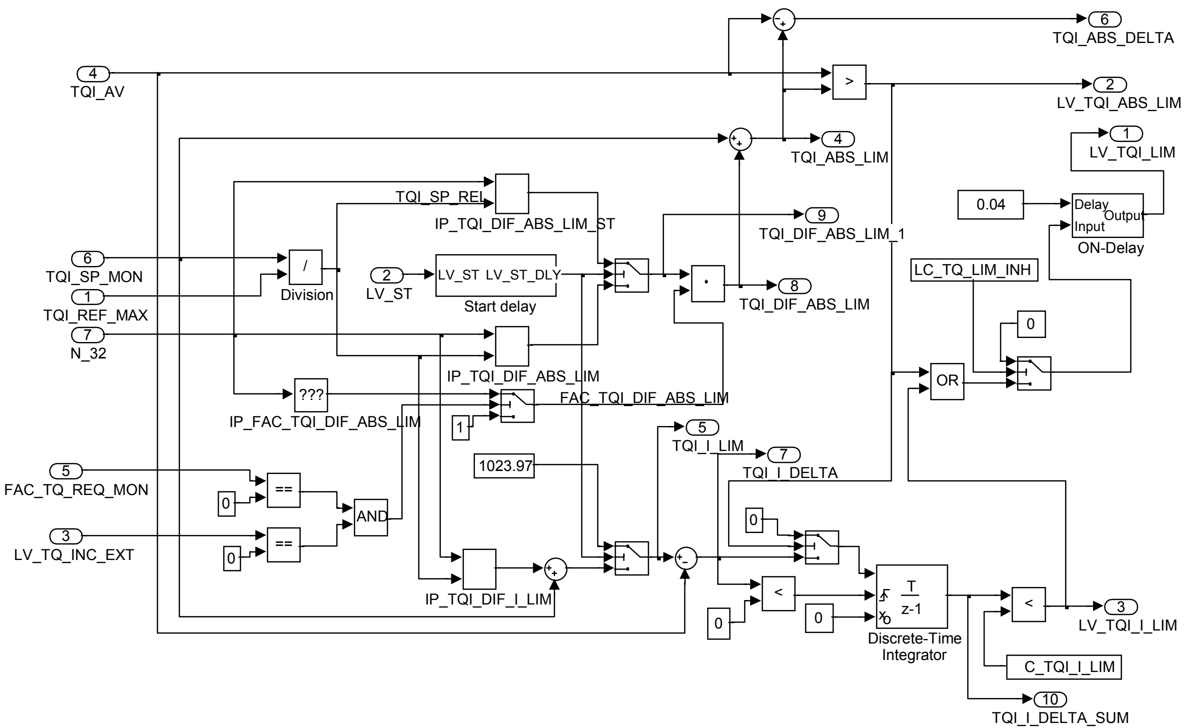
This module *Torque request for safety* generates a limitation torque request for the slow (air) path and the fast (ignition) path, if the actual indicated engine torque TQI_AV exceeds the desired indicated engine torque, which is generated in the monitoring level (TQI_SP_MON), by more than a calculated threshold TQI_DIF_LIM.

Signal flow diagram:

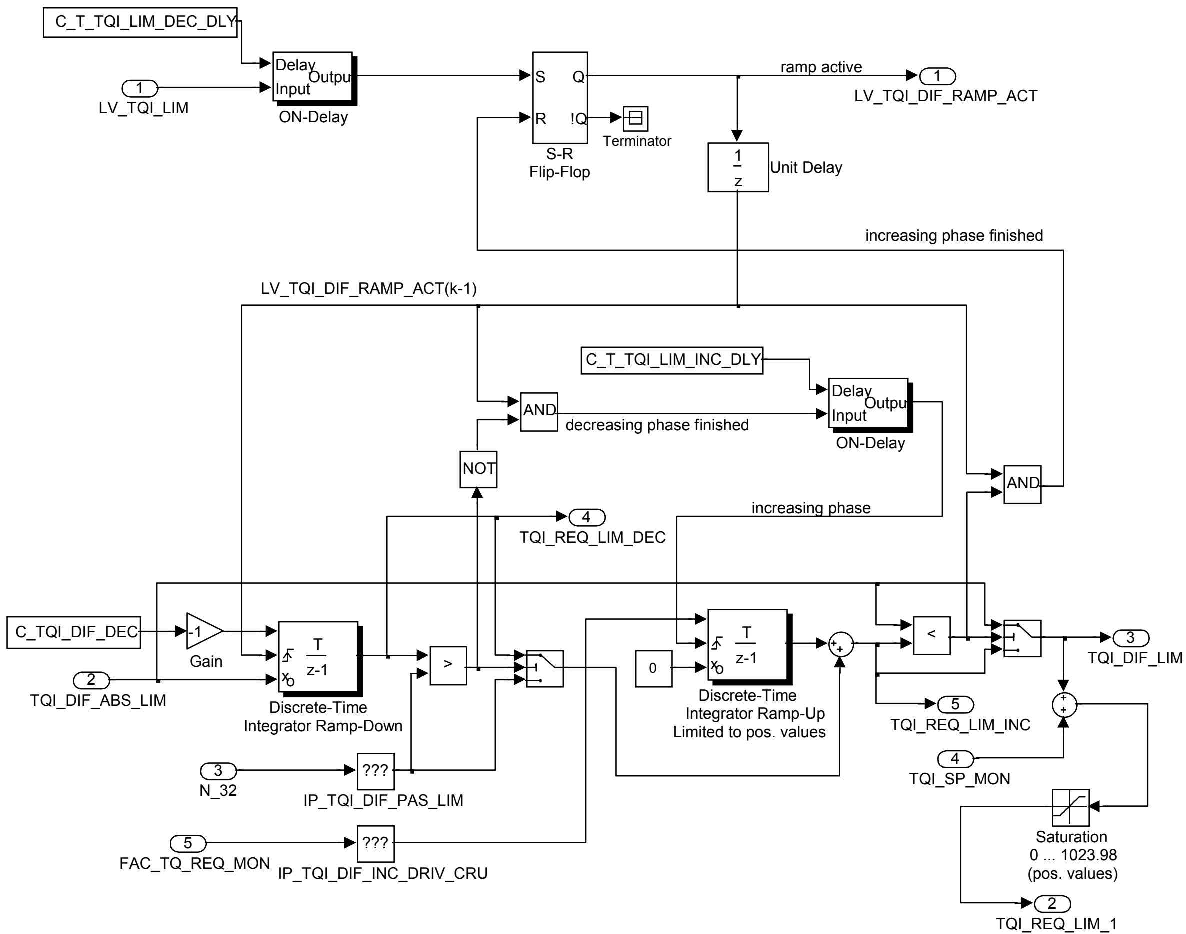
Overview:



Failure detection:



Target value:



Description:

Failure detection:

If TQI_AV exceeds a absolute threshold (TQI_SP_MON + TQI_DIF_ABS_LIM) or a integrated value of TQI_AV > TQI_I_LIM reached the threshold C_TQI_I_LIM, a torque intervention due to this module is active after a ON-Delay.

A ON-delay is necessary because of this module is not synchronized to the module ‘Desired indicated engine torque’ of level 2. No ON-delay could cause inaccurate torque limitation in case of a high positive torque gradient in level 1 while TQI_SP_MON is not updated yet.

Target value:

After the torque intervention is active, TQI_REQ_LIM_1 is ramped down.

After the target value has been reached, the target value TQI_REQ_LIM_1 is ramped up again.

Overview:

To avoid uncomfortable jumps when leaving the intervention function by choosing the passive value a rate limiter for positive gradients is used. The output of this function is TQI_REQ_LIM_3.

For the slow air path the torque request change can be limited in falling direction, to adjust a more comfortable intervention and to force the systems to react on this intervention necessacity either more with the fast or the slow path.

The bits LC_TQ_LIM_INH_SLOW/FAST allow to avoid generating torque inteventions in the slow or the fast path.

Application conditions:

Activation: LV_TQ_LIM_ACT = 1

Deactivation: otherwise

Initialization: (at reset or LV_TQ_LIM_ACT = 0)

LV_TQ_LIM_INTV	= 0
LV_TQI_LIM	= 0
TQI_REQ_LIM_FAST	= C_TQI_REQ_LIM_PAS
TQI_REQ_LIM_SLOW	= C_TQI_REQ_LIM_PAS
TQI_REQ_LIM_3	= C_TQI_REQ_LIM_PAS
TQI_I_DELTA_SUM	= 0
all other integrators	= {maximum positive value}
all timers	= {timer init value}
T_TQ_LIM_INTV	= 0

Update rate 40 ms

Scheduling: this function has to be calculated immediate after the module ‘Torque request for safety (Appl. Inc.)’

Formula section:

Overview:

```
IF      LC_TQ_LIM_INH_FAST == 1
THEN    TQI_REQ_LIM_FAST = 1023.97 Nm
ELSE    TQI_REQ_LIM_FAST = TQI_REQ_LIM_3
ENDIF
IF      LC_TQ_LIM_INH_SLOW == 1
THEN    TQI_REQ_LIM_4 = 1023.97 Nm
ELSE    TQI_REQ_LIM_4 = TQI_REQ_LIM_3
ENDIF
```

ON-Delay:

```
IF  {input}        == 0
THEN
    {output}       = 0
    {timer}        = {delay}
ELSE
    IF    {timer(k)}     == 0
    THEN
          {output}       = 1
    ELSE  {timer(k)}     = {timer(k-1)} - {update rate}
    ENDIF
ENDIF
```

Off-Delay:

```
IF      [ ( (TQI_EMS  >  TQI_REQ_LIM_2) AND (LV_TQI_DIF_RAMP_ACT == 1) )
        OR  ( LV_TQI_LIM == 1 ) ]
THEN    (LV_TQ_LIM_INTV = 1
        T_TQ_LIM_INTV = C_T_TQ_LIM_INTV)
ELSE    IF    T_TQ_LIM_INTV == 0
        THEN  LV_TQ_LIM_INTV = 0
        ELSE  T_TQ_LIM_INTV = T_TQ_LIM_INTV - [update rate]
        ENDIF
ENDIF
```

Event Counter:

IF (LV_TQI_DIF_RAMP_ACT$_{k-1}$ == 0 **AND** LV_TQI_DIF_RAMP_ACT$_k$ == 1)

THEN SUM_TQI_REQ_LIM = SUM_TQI_REQ_LIM + 1

ENDIF

{SUM_TQI_REQ_LIM must be stored in non-volatile memory until failure memory is cleaned.}

Rate limiter for positive gradient:

IF (TQI_REQ_LIM_2 > (TQI_REQ_LIM_3$_{k-1}$ + C_TQI_LIM_INC))

THEN TQI_REQ_LIM_3$_k$ = TQI_REQ_LIM_3$_{k-1}$ + C_TQI_LIM_INC

ELSE TQI_REQ_LIM_3$_k$ = TQI_REQ_LIM_2

ENDIF

Rate limiter for negative gradient:

IF (TQI_REQ_LIM_4 < (TQI_REQ_LIM_ SLOW$_{k-1}$ – C_TQI_LIM_DEC))

THEN TQI_REQ_LIM_SLOW$_k$ = TQI_REQ_LIM_SLOW$_{k-1}$ - C_TQI_LIM_DEC

ELSE TQI_REQ_LIM_SLOW$_k$ = TQI_REQ_LIM_4

ENDIF

Target value and Failure detection:

Discrete time integrator:

IF {integrator} = {active}
THEN
{integrator output(k)} = {integrator output(k-1)} + {integrator input}
ELSE
{integrator output(k)} = {integrator init}
ENDIF

The discrete time integrator for the ramp-down is limited to the entire phys. range (–1024 ... +1023.97 Nm).

The discrete time integrator for the ramp-up is limited to pos. values (0 ... +1023.97 Nm).

SR flip-flop:

It works like a normal SR-flip-flop. The following special case has to be considered:

IF S == 1 **AND** R == 1
THEN Q=0
ENDIF

Failure detection

Start delay:

IF LV_ST == 1
THEN { LV_ST_DLY = 1
$T\_MAX\_ST\_DUR_k$ = C_T_MAX_ST_DUR }
ELSE { **IF** T_MAX_ST_DUR == 0
THEN LV_ST_DLY = 0
ELSE $T\_MAX\_ST\_DUR_k$ = $T\_MAX\_ST\_DUR_{k-1}$ - [update rate]
ENDIF }
ENDIF

FAC_TQI_DIF_ABS_LIM:

IF FAC_TQ_REQ_MON == 0 **AND** LV_TQ_INC_EXT == 0
THEN FAC_TQI_DIF_ABS_LIM = IP_FAC_TQI_DIF_ABS_LIM (N_32)
ELSE FAC_TQI_DIF_ABS_LIM = 0.9962 {upper limit => no factor needed}
ENDIF

TQI_DIF_ABS_LIM = IP_TQI_DIF_LIM(_ST) * FAC_TQI_DIF_ABS_LIM

TQI_I_LIM:

IF LV_ST_DLY == 1
THEN TQI_I_LIM = 1023.97 Nm (passive value to inhibit function in the start phase)
ELSE TQI_I_LIM = TQI_SP_MON + IP_TQI_DIF_I_LIM
ENDIF

only necessary for calibration:

TQI_ABS_DELTA_REL = TQI_ABS_DELTA / TQI_ABS_LIM
TQI_I_DELTA_REL = TQI_I_DELTA / TQI_I_LIM

Initialisation: at Ignition off → on:

TQI_ABS_DELTA_REL_MEM=0.996

TQI_ABS_DELTA_MEM = 1023.97Nm

TQI_I_DELTA_REL_MEM = 0.996

TQI_I_DELTA_MEM = 1023.97Nm

IF TQI_ABS_DELTA_REL < TQI_ABS_DELTA_REL_MEM

THEN TQI_ABS_DELTA_REL_MEM = TQI_ABS_DELTA_REL

ENDIF

IF TQI_ABS_DELTA < TQI_ABS_DELTA _MEM

THEN TQI_ABS_DELTA _MEM = TQI_ABS_DELTA

ENDIF

IF TQI_I_DELTA_REL < TQI_I_DELTA_REL_MEM

THEN TQI_I_DELTA_REL_MEM = TQI_I_DELTA_REL

ENDIF

IF TQI_I_DELTA < TQI_I_DELTA _MEM

THEN TQI_I_DELTA _MEM = TQI_I_DELTA

ENDIF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_TQ_LIM_INH	1	0 ... 1 H	0 ... 1	1	-
Logical constant inhibition torque intervention for safety in general					
LC_TQ_LIM_INH_FAST	1	0 ... 1 H	0 ... 1	1	-
Logical constant inhibition torque intervention for safety for fast path					
LC_TQ_LIM_INH_SLOW	1	0 ... 1 H	0 ... 1	1	-
Logical constant inhibition torque intervention for safety for slow path					
C_TQI_REQ_LIM_PAS	1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
passive torque request for no torque intervention					
C_TQI_I_LIM	1	8000H ... 0H	-1024 0	0.03125	Nm
Maximum Integrator value to detect torque error					
C_TQI_DIF_DEC	1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
decrement of torque-ramp integrator					
C_T_TQ_LIM_INTV	1	0...FFFFH	0...2621.4	0.04	s
Time after active torque intervention to reset LV_TQ_LIM_INTV					
C_T_MAX_ST_DUR	1	0...FFFFH	0...2621.4	0.04	s
Maximum time for start phase after LV_ST was set to 0					
C_T_TQI_LIM_DEC_DLY	1	0...FFH	0...10.2	0.04	s
time for ON-delay					
C_T_TQI_LIM_INC_DLY	1	0...FFH	0...10.2	0.04	s
time for ON-delay					
IP_FAC_TQI_DIF_ABS_LIM	4	0 ... FFH	0 ... 0.9962	0.00391	-
LDP_N_32_IP_FAC_TQI_DIF_ABS_LIM	4	0 ... FFH	0 ... 8160	32	rpm
Factor for absolute torque threshold depending on N_32					
IP_TQI_DIF_I_LIM	4*4	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDPM_TQI_SP_REL_1_TQSP	4	0...FFH	0 ... 0.9962	0.00391	-
LDPM_N_32_1_TQSP	4	0 ... FFH	0 ... 8160	32	rpm
Torque threshold for integrator depending on TQI_SP_REL and N_32					
IP_TQI_DIF_ABS_LIM	4*4	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDPM_TQI_SP_REL_1_TQSP	4	0...FFH	0 ... 0.9962	0.00391	-
LDPM_N_32_1_TQSP	4	0 ... FFH	0 ... 8160	32	rpm
Torque threshold depending on TQI_SP_REL and N_32					
IP_TQI_DIF_ABS_LIM_ST	4*4	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDP_TQI_SP_REL_IP_TQI_DIF_ST	4	0...FFH	0 ... 0.9962	0.00391	-
LDP_N_32_IP_TQI_DIF_ST	4	0 ... FFH	0 ... 8160	32	rpm
Torque threshold depending on TQI_SP_REL and N_32 for start phase					
IP_TQI_DIF_PAS_LIM	1*4	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDPM_N_32_1_TQSP	4	0 ... FFH	0 ... 8160	32	rpm
Torque Taskvalue for Ramp depending N_32					
IP_TQI_DIF_INC_DRIV_CRU	1*4	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDP_FAC_TQ_REQ_MON_IP_DRIV_CRU	4	0 ... FFH	0 ... 1.992	7.8125e-3	-
Torque increment depending on FAC_TQ_REQ_MON					
C_TQI_LIM_INC	1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Increment for rate limiter for positive gradient					
C_TQI_LIM_DEC	1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Decrement for rate limiter for negative gradient					

D.54 Application Incidences for Torque Request for Safety

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_TQ_INC_EXT	V/O	0...1	0...1	1	-
Flag for having external torque demands which can increase torque					
LV_TQ_LIM_ACT	V/O	0...1	0...1	1	-
Activation flag for module 'Torque request for safety'					

Input data:

FUNCTION DESCRIPTION:

General information:

The objective of this module is to define the application incidences for the module "Torque request for safety".

Description:

The module "Torque request for safety" needs a flag for having external torque demands, which can increase torque, as an input. Up to now, only MSR can leed to increasing torque.

Application conditions:

Initialisation: none

Recurrence: 40 ms

Activation: at every engine state

Deactivation: none

Scheduling: this function has to be calculated immediate before the module 'Torque request for safety'

Formula section:

LV_TQ_LIM_ACT = 0 {Torque request for safety inactive}

LV_TQ_INC_EXT = 0

D.55 Engine torque status

D.55.1 Ratio between current and standard engine torque capabilities

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
RATIO_TQI_BAS_MAX_STND	V/O	0...FFH	0...2	0,0078	-
Standard Torque Ratio : final value sent to TCU					
RATIO_TQI_BAS_MAX_STND_1	V	0...FFH	0...2	0,0078	-
Standard Torque Ratio : AMP effect					
RATIO_TQI_BAS_MAX_STND_2	V/S	0...FFFFH	0...2	3.0517e-5	-
Standard Torque Ratio : IGA effect					
RATIO_TQI_BAS_MAX_STND_2_INTER	V	0...FFH	0...2	0,0078	-
IGA term – intermediate value					
RATIO_TQI_BAS_MAX_STND_3	V	0...FFH	0...2	0,0078	-
Standard Torque Ratio : TIA effect					
EFF_IGA_BAS_TMP	V/O	0..FFFFH	0..1.999969	3.0517e-5	-
Auxiliary ignition efficiency related to basic ignition angle					
IGA_BAS_TMP	V	0..FFH	-35.625..60	0.375	°CRK
Basic ignition angle, intermediary internal variable, only for calculation purposes					

Input data:

TIA	AMP	LAMB_SP_MV	EFF_IGA_BAS_COR
MAF_HB	IGA_REF	LV_KNK_CTL_ENA	N_32
TCO	IP_IGA_BAS[NC_CBK_IN_NR]	IP_EFF_IGA	IP_IGA_BAS_LAMB

FUNCTION DESCRIPTION:

General information:

To adapt the gear shift strategy, the Transmission Control Unit (TCU) needs the information about long-term changes of the engine torque capabilities due to special operating conditions (e.g. environmental conditions, fuel quality).

Description:

The present function calculates the output RATIO_TQI_STD, which gives the ratio of the currently available torque from the engine, compared to a standard torque. This information is based on 3 main parameters: TIA, AMP and ignition (which includes knock intervention and eventually low fuel qualities).
- Actual TIA and AMP are compared to calibratable standard values. (note : TIA < TIA_standard would lead to an efficiency bigger than 1. This is not wanted, therefore the TIA effect is set to 1 in this case)
- Actual ignition is compared to the basic ignition. As this factor is very depending of the working point, the calculation is done only in some specific N-MAF windows in which the knock effects are supposed to be the biggest, in order to get a worst case value. Additionaly, the factor is filtered to send a more steady value to the TCU.

The factor relative to ignition is stored at ECU off, because at restart it may take a certain time before the N-MAF window is met again. Is is possible to reset the store values via calibration (see specification reset of adaptative values).

Application conditions:

Initialisation at ECU reset :

RATIO_TQI_BAS_MAX_STND	= 1
RATIO_TQI_BAS_MAX_STND_1	= 1
RATIO_TQI_BAS_MAX_STND_2_INTER	= 1
RATIO_TQI_BAS_MAX_STND_2	= last stored value if available, or 1 if no saved data available

Recurrence: 100 ms

Activation: all engine states except LV_ES and LV_ST

Deactivation: LV_ES = 1 or LV_ST = 1

Formula section:

Ambient Pressure Effect

RATIO_TQI_BAS_MAX_STND_1 = (AMP / C_AMP_RATIO_TQI_BAS_MAX_STND)

Ignition Effect

IGA_BAS_TMP = IP_IGA_BAS_0 (N_32,MAF_HB) *Basic map for spark advance*
+ IP_IGA_BAS_LAMB (LAMB_SP_MV) *Correction for variable lambda*

EFF_IGA_BAS_TMP= IP_EFF_IGA (IGA_BAS_TMP – IGA_REF)

RATIO_TQI_BAS_MAX_STND_2_INTER = EFF_IGA_BAS_COR / EFF_IGA_BAS_TMP

If TCO > C_TCO_MIN_RATIO_TQI_BAS_MAX_STND
and N_32 < C_N_MAX_RATIO_TQI_BAS_MAX_STND
and N_32 > C_N_MIN_RATIO_TQI_BAS_MAX_STND
and MAF_HB > C_MAF_MIN_RATIO_TQI_BAS_MAX_STND
and MAF_HB < C_MAF_MAX_RATIO_TQI_BAS_MAX_STND

then **RATIO_TQI_BAS_MAX_STND_2** =
$RATIO\_TQI\_BAS\_MAX\_STND\_2_{n-1} + C\_CRLC\_RATIO\_TQI\_BAS\_MAX\_STND\_2 * (RATIO\_TQI\_BAS\_MAX\_STND\_2\_INTER_{n} - RATIO\_TQI\_BAS\_MAX\_STND\_2_{n-1})$

else RATIO_TQI_BAS_MAX_STND_2 is not updated

Intake Air Temperature Effect

If TIA > C_TIA_RATIO_TQI_BAS_MAX_STND

then **RATIO_TQI_BAS_MAX_STND_3** =
(C_TIA_RATIO_TQI_BAS_MAX_STND + 273,15) / (TIA + 273,15)

else RATIO_TQI_BAS_MAX_STND_3 = 1

General Result

RATIO_TQI_BAS_MAX_STND = RATIO_TQI_BAS_MAX_STND_1 *
RATIO_TQI_BAS_MAX_STND_2 *
RATIO_TQI_BAS_MAX_STND_3

General Limitation

If RATIO_TQI_BAS_MAX_STND > C_RATIO_TQI_BAS_MAX_STND_MAX
then RATIO_TQI_BAS_MAX_STND = C_RATIO_TQI_BAS_MAX_STND_MAX

If RATIO_TQI_BAS_MAX_STND < C_RATIO_TQI_BAS_MAX_STND_MIN
then RATIO_TQI_BAS_MAX_STND = C_RATIO_TQI_BAS_MAX_STND_MIN

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CRLC_RATIO_TQI_BAS_MAX_STND	1	0...FFFFH	0...0,99998	1,53e-5	-
Filtering factor for RATIO_TQI_BAS_MAX_STND					
C_AMP_RATIO_TQI_BAS_MAX_STND	1	0...FFFFH	0...5434	0,083	mbar
Value of the standard AMP for torque reduction factor calculation					
C_TIA_RATIO_TQI_BAS_MAX_STND	1	0...FEH	-48...142.5	0.75	°C
Value of the standard TIA for torque reduction factor calculation					
C_MAF_MIN_RATIO_TQI_BAS_MAX_STND	1	0...FFH	0...1389	5,447	mg/stk
minimum MAF threshold for torque reduction factor update					
C_MAF_MAX_RATIO_TQI_BAS_MAX_STND	1	0...FFH	0...1389	5,447	mg/stk
maximum MAF threshold for torque reduction factor update					
C_N_MIN_RATIO_TQI_BAS_MAX_STND	1	0...FFH	0...8160	32	rpm
minimum N_32 threshold for torque reduction factor update					
C_N_MAX_RATIO_TQI_BAS_MAX_STND	1	0...FFH	0...8160	32	rpm
maximum N_32 threshold for torque reduction factor update					
C_TCO_MIN_RATIO_TQI_BAS_MAX_STND	1	0...FEH	-48...142.5	0.75	°C
minimum TCO threshold for torque reduction factor update					
C_RATIO_TQI_BAS_MAX_STND_MIN	1	0...FFH	0...2	0,0078	-
Lower limitation of the standard torque ratio					
C_RATIO_TQI_BAS_MAX_STND_MAX	1	0...FFH	0...2	0,0078	-
Upper limitation of the standard torque ratio					

D.55.2 Actual Torque from EMS

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_AV_EMS	V / O	8000 ... 7FFF H	-1024 ... 1023.97	0.03125	Nm
Actual indicated engine torque without external interventions					

Input data:

LV_GS_ENA	SPK_RTD_TCU	TQI_GS_REQ	LV_PUC
LV_ASR_ACT	LV_ASR_REQ	TQI_ASR_REQ	TQI_AV
TQI_REF	TQI_REF_IGA	TQI_EMS	TQ_DIF_P_D_FAST_IS
EFF_IGA_MIN	EFF_LAMB_AV	NC_ETC_CONF	TQI_REQ_TRA
TQI_BAS_MAX	LV_MSR_ENA	LV_MSR_ACT	TQI_MSR_REQ
ETL_TCU_CAN			

FUNCTION DESCRIPTION:

General information:

TQI_AV_EMS is the actual indicated torque without considering the external interventions due to GS or ASR.

Description:

Non-ETC system

As long as no external inetrvention is enabled nor requested, or while the engine state LV_PUC is active, TQI_AV_EMS is forced to TQI_AV.

In the other cases, the recalculation of the actual engine torque is performed based on TQI_EMS (torque request before GS and ASR intervention) added to the torque intervention from idle speed control (TQ_DIF_P_D_FAST_IS), but limited on the one hand to the maximum available torque based on N, MAF, lambda and max spark advance considering ambient and engine temperatures as well as knock control (TQI_REF_IGA*EFF_LAMB_AV), and on the other hand to the minimum possible torque defined by TQI_REF * EFF_IGA_MIN.

ETC system

Becasue TPS is used to fulfil the torque intervention request and MAF value is directly impacted by it, we could not use the same formula like non-ETC system.

So, the TQI_AV_EMS is concluded by Minimum selection between the driver request torque (TQI_REQ_TRA) and Maximum available torque at basic condition (TQI_BAS_MAX).

Application conditions:

Initialisation at ECU reset :	TQI_AV_EMS = 0 Nm
Recurrence:	10 ms
Activation:	LV_HOM_RUN = 1
Deactivation:	LV_HOM_RUN = 0

Formula section:

```
if      [LV_GS_ENA = 0
     or TQI_GS_REQ >= TQI_AV_EMS(n-1)]
and     [LV_ASR_ACT = 0
     or LV_ASR_REQ = 0
     or TQI_ASR_REQ >= TQI_AV_EMS(n-1)]
and     [LV_MSR_ACT = 0
     or LV_MSR_ENA = 0
     or TQI_MSR_REQ =< TQI_AV_EMS(n-1)]
and     [ETL_TCU_CAN >= TQI_AV_EMS(n-1)]
or      LV_PUC = 1
then No external intervention or PUC
        TQI_AV_EMS = TQI_AV
else Recalculation of TQI_AV without considering the external requests
     if   NC_ETC_CONF = 0 (Non-ETC system)
     then TQI_AV_EMS = max [TQI_REF * EFF_IGA_MIN ;
          min (TQI_REF_IGA * EFF_LAMB_AV; TQI_EMS + TQ_DIF_P_D_FAST_IS)]
     else NC_ETC_CONF = 1 (ETC system)
          if    TQI_TCS_CAN < 100% or TQI_MSR_CAN > 0%
                 or  ETL_TCU_CAN < 100% (FFh)      % slow path request present
          then  TQI_AV_EMS = Min [TQI_REQ_TRA ; Max (TQI_BAS_MAX; TQI_AV)]
          else  TQI_AV_EMS = max [TQI_REF * EFF_IGA_MIN ;
          min (TQI_REF_IGA * EFF_LAMB_AV; TQI_EMS + TQ_DIF_P_D_FAST_IS)]
```

Remark : After the transition to no torque intervention (TQI_AV_EMS = TQI_AV) from torque intervention (TQI_AV_EMS = Formula), the TQI_AV_EMS has to be calculated by formula during 50ms (5 recurrences) to avoid a sudden drop of TQI_AV_EMS by difference of control recurrence between IGA_AV (segment recurrence) and TQI_AV (10 ms recurrence).

D.56 Indicate significant inaccuracy in torque calculation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQI	O/V	0...1H	0...1	1	-
Flag to indicate significant inaccuracy in torque calculation					

Input data:

LV_ERR_MAF	LV_ERR_CRK	LV_ERR_TPS	LV_ES
LV_ERR_MAP			

D.56.1 General Information

LV_ERR_TQI provides information about the accuracy/trustworthiness of the calculated torque value that is transmitted via CAN. In case that a failure of a relevant sensor or actuator is detected, this flag is set to 1.

Application conditions:

Initialisation :

Recurrence: 40 ms

Activation: LV_ES = 0

Deactivation: LV_ES = 1

Function Description

If LV_ERR_MAF = 1
or LV_ERR_MAP = 1
or LV_ERR_CRK = 1
or LV_ERR_TPS = 1
then LV_ERR_TQI = 1

D.57 Variables for fleet monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_AV_MAX_DC	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
maximum engine torque in current driving cycle					
TQ_AV_MAX_TOT_DC	O/V/S	8000...7FFFH	-1024...1023.97	0.03125	Nm
maximum engine torque in former / current driving cycle					

Input data:

TQ_AV	LV_ES		

Application conditions:

Initialisation: - at first ECU power up and non-volantile memory lost

TQ_AV_MAX_TOT_DC = 8000H (-1024 Nm)

- otherwiste: resored from non-volantile memory
- at LV_IGK 0->1 and reset:

TQ_AV_MAX_DC = 0

Recurrence: 10 ms

Activation: LV_ES = 0

Deactivation: -

Formula section:

If TQ_AV > TQ_AV_MAX_DC

Then TQ_AV_MAX_DC = TQ_AV

Endif

If TQ_AV > TQ_AV_MAX_TOT_DC

Then TQ_AV_MAX_TOT_DC = TQ_AV

Endif

D.58 Driver Request Limitation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_REQ_CLU	V/O	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Requested torque at clutch					
TQ_MAX_DROF_AT	V	8000 ... 7FFFH	-1024 ... 1023.97	0.03125	Nm
Maximum allowed clutch torque for drive off					
T_DROF_AT	V	0..FFFFH	0...655.35	0.01	s
Drive off timer - AT					
LV_DROF_AT_ACT	V/O	0..1H	0...1	1	-
Flag to show torque limitation active					

Input data:

VS	PV_AV_H	LV_AT	GEAR
LV_CT	TQ_REQ_CLU_1	PV_AV	N
LV_ES			

General information:

In order to gain the desired torque profile (as shown in the figure 1) during drive-off in an AT vehicle, the driver requested torque can be limited. This limitation is scheduled after the module "Torque Request at Clutch General", which now produces the output TQ_REQ_CLU_1, and before Torque Transient, which consumes the output from this spec TQ_REQ_CLU.

Whilst this function is active, the standard driveability torque transient function may be disabled, or may continue to be active.

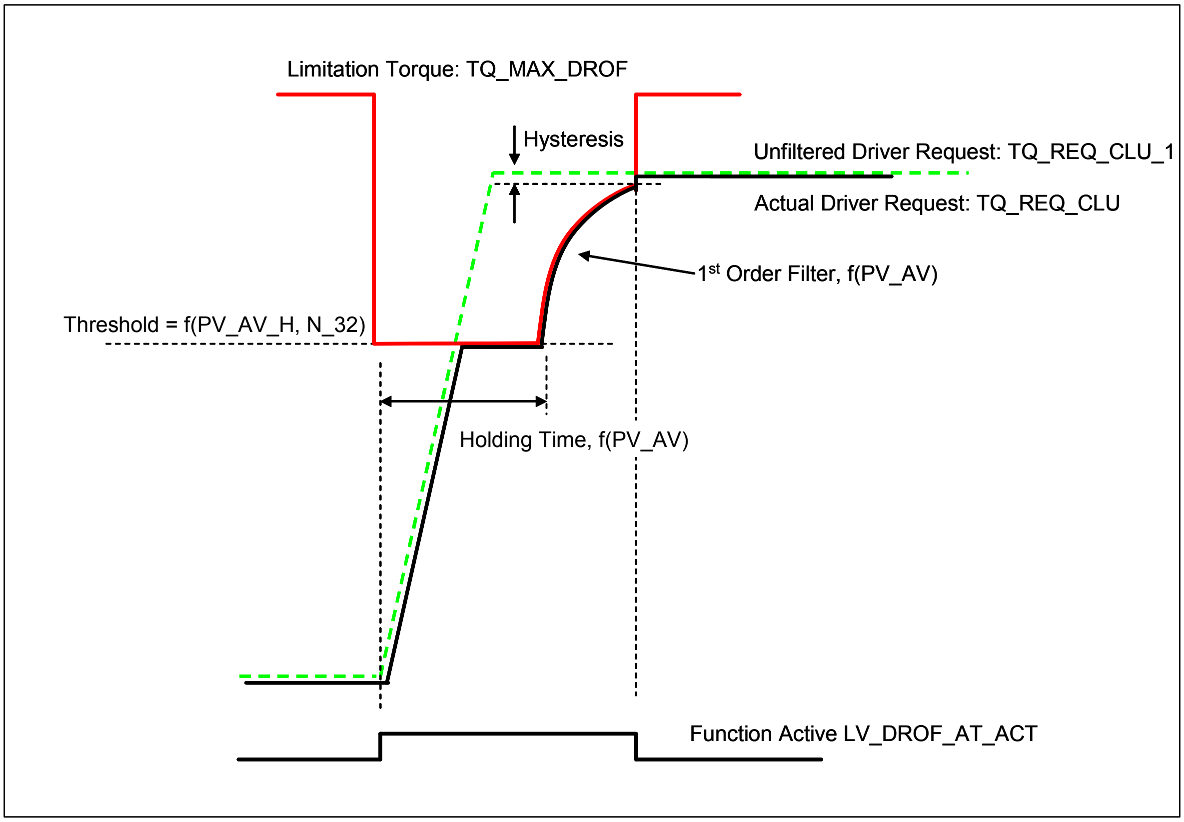


Figure 1: Desired torque profile at drive off

Application conditions:

Activation:	*at every engine state*	
Deactivation:	-	
Initialization:	at reset	TQ_REQ_CLU = 0 Nm
		TQ_MAX_DROF_AT = 1023.97 Nm
		T_DROF_AT = 0 s
		LV_DROF_AT_ACT = 0
Update rate:	10 ms	

Formula section:

```
IF    LV_ES = 1 or LV_AT = 0 or LC_DROF_AT_ENA = 0
THEN
      # Function disabled
      TQ_REQ_CLU = TQ_REQ_CLU_1
      TQ_MAX_DROF_AT = 1023.97 Nm
      LV_DROF_AT_ACT = 0

ELSE
      # Function Enabled
      IF      VS < C_VS_MAX_DROF_AT and GEAR = 1 and LV_CT = 1->0 (falling flank)
      THEN  LV_ DROF_AT_ACT = 1       # Function turned on
      ENDIF

      IF      LV_CT = 1
      THEN LV_DROF_AT_ACT = 0         # Function turned off
      ENDIF


      IF      LV_DROF_AT_ACT = 1
      THEN
              Increment T_DROF_AT
              IF      T_DROF_AT < IP_T_HLD_DROF_AT(PV_AV)
              THEN  TQ_MAX_DROF_AT = IP_TQ_MAX_DROF_AT(PV_AV_H, N)
              ELSE
                      IF      (TQ_REQ_CLU_1 - C_TQ_HYS_DROF_AT) <= TQ_MAX_DROF_AT
                      THEN
                              # Function turned off
                              LV_DROF_AT_ACT = 0
                              TQ_MAX_DROF_AT = 1023.97 Nm
                              T_DROF_AT = 0 s
                      ELSE
                              TQ_MAX_DROF_AT_n = TQ_MAX_DROF_AT_n-1
                                      + IP_CRLC_DROF_AT (PV_AV)
                                              * (TQ_REQ_CLU_1 - TQ_MAX_DROF_AT_n-1)
                      ENDIF
              ENDIF
      ELSE
              TQ_MAX_DROF_AT = 1023.97 Nm
              T_DROF_AT = 0 s
      ENDIF

      TQ_REQ_CLU = min( TQ_REQ_CLU_1, TQ_MAX_DROF_AT )

ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_DROF_AT_ENA	1	0...1H	0...1	1	-
Switch to enable driver request limitation for drive-off					
C_VS_MAX_DROF_AT	1	0...FFH	0...255	1	km/h
Maximum vehicle speed to enable driver request limitation for drive-off					
C_TQ_HYS_DROF_AT	1	0 ... 7FFFH	0 ... 1023.97	0.03125	Nm
Hysterisis to turn off driver request limitation for drive-off					
IP_CRLC_DROF_AT	8	0...FFFFH	0...1	1.526E-5	-
LDPM_PV_AV_DROF_AT	8	0...FFH	0...99.6	0.3906	%
Correlation constant for turning off driver request limitation for drive-off					
IP_TQ_MAX_DROF_AT	8	0 ... FFFFH	-1024 ... 1023.97	0.03125	Nm
LDP_PV_AV_H_DROF_AT	8	0..3FFH	0...99.6	0.09765	%
LDP_N_DROF_AT	8	0..1FE0H	0...8160	1	rpm
Torque limitation for driver request for drive-off					
IP_T_HLD_DROF_AT	8	0...FFFFH	0...655.35	0.01	s
LDPM_PV_AV_DROF_AT	8	0...FFH	0...99.6	0.3906	%
Limitation time for driver request for drive-off					

E Monitoring Concept

Table of Contents

Index

A

C

L

V

X

E.1 General

E.1.1 Basic Idea

The Monitoring Concept has to prevent an ETC system from dangerous behaviour like sudden acceleration against the driver's intent. At first faults, this must be without intervention of the driver. To avoid immobile cars at first faults, a limited limp home function is activated. This means the dynamic behaviour of the car is limited, and so in case of another (second) fault, the driver must possibly react (activation of brake pedal). The accepted fault reaction time is 500 ms. Minor deviations in the order of about 5% are possible due to run time fluctuations, aging effects and temperature influences. No or low engine power is seen as a safe state.

The Siemens ETC-Monitoring-Concept is based on the ETC-Safety-Concept of VDA Arbeitskreis E-Gas using the following main modification:

Additional, diverse AD-converter in monitoring unit (MU) for monitoring af A/D conversion. The used analog voltage must be non-constant, the frequency of the voltage changes must match to the synchronisation constraints given in the Processor Monitoring.

The monitoring concept consists of three levels, as shown in the following figure:

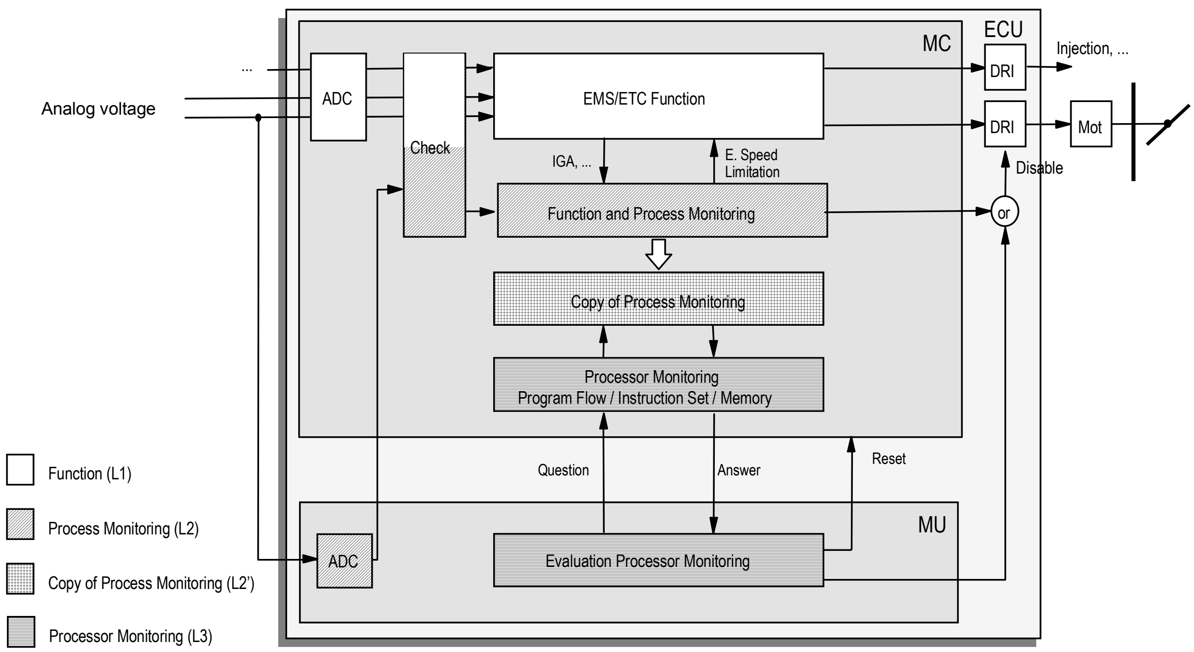


Figure 1: The three levels of the monitoring concept.

- The function level is located on the main processor (MC).
- The function/process monitoring level is also located on the main processor.
- The processor monitoring level is located on the main processor and on the monitoring processor (MU).

E.1.1.1 Function Level (VDA-name: Level 1)

The function level performs standard EMS functions including ETC functionality (description *not* to be found in this chapter):

- Determination of the drivers demand (via PVS) including standard diagnosis (signal range check, ...)
- Determination of throttle position (via TPS) including standard diagnosis (signal range check, ...)
- Receiving external torque requests (e.g. ASR, MSR, ...) including standard diagnosis (e.g. monitoring CAN by parity bit, ...). External modules must generate correct torque requests (safety requirement!) the ECU can only check, whether the transmission of this request was without faults.
- Internal calculation of torque requests (ISC, Cruise Control)
- Co-ordination and realisation of requested torque

- output: throttle position, ignition angle, injection time and cylinder cut off

E.1.1.2 Process Monitoring Level (VDA-name: Level 2)

The process monitoring performs monitoring of sensors, actuators, torque effective functions, torque co-ordination, and torque realization as a check of the function level.

The description can be found in the modules:

- Process Monitoring
- Monitoring Sensor Signals (e.g. PVS, TCO, ...)
- Monitoring Internal Signals (e.g. MAF, N_32, ...)
- Monitoring CAN Signals
- Torque Monitoring
- Fault Reaction

E.1.1.3 Processor Monitoring Level (VDA-name: Level 3)

The processor monitoring monitors proper processor function: memory, operations of the processor, program flow, analog digital conversion

The description can be found in the modules:

- Processor Monitoring
- Communication between MC and MU
- Fault Reaction

There are additional port extensions located on the monitoring unit. Note that this functionality is not a core functionality of level 3:

- Input / Output (Description of port extension facility of the monitoring unit)

E.1.2 Hardware Safety Module

The main-processor (MC) and the monitoring processor (MU, incl. port extensions) are connected via an internal communication. MU has its own oscillator (ceramic resonator).

When resetting, MC gives a reset signal to MU so it will also reset. This enables synchronisation between them.

Via the internal communication informations are excanged. This informations belongs to: test calculations (level 2'), program flow of MC, monitoring of ADC, port extensions.

For the purpose of test calculations answers are compared to known results on MU. In case of faults, the power stages of the throttle and the injectors are disabled.

One analog voltage is connected with the first AD-channel of MU and the dedicated channel on MC. The MU value is transmitted to MC only to check the ADCs. Remaining AD-channels and digital outputs can be used as port extension (no monitoring function, but it has to be assured, that no damaging voltage is fed to the ports in case of any fault).

E.2 Analog and pseudo digital inputs

E.2.1 Analog inputs on MU

Output data: (on MU)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AN_BUF_0	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN0 on MU (used for monitoring)					
AN_BUF_1	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN1 on MU					
AN_BUF_2	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN2 on MU					
AN_BUF_3	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN3 on MU					
AN_BUF_8	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN8 on MU					
AN_BUF_10	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN10 on MU					
AN_BUF_12	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN12 on MU					
AN_BUF_13	O	000h ... 3FFh	0 ... 4.995	5/1024	V
AD-value of AIN13 on MU					
LV_ERR_CONV_AN_MU	O	0 ... 1h	0 ... 1	1	1
Status flag of analog-digital conversion (indicates a fault if set)					

Note: The inputs AIN2 and AIN3 are only conducted for the 44pin package. The scope of these variables are the MU only. They are available on MC only, if the contents is transferred via SPI communication.

FUNCTION DESCRIPTION:

General information:

The MU offers up to seven 10bit AD-channels for the analog port extension. One AD-channel (AIN0) is used for Monitoring of the A/D-conversion. This signal is converted by MU directly after monitoring communication. The right behaviour of the AIN0 is checked every time the signal is converted. When the AD-conversions start, it is checked that the EOC-bit (End Of Conversion) of the control register is set and reset properly. If the bit LV_ERR_CONV_AN_MU is set, it indicates an error of the AD-conversion when converting the channel used for the monitoring of A/D conversion (AIN0).
The AD-channels are converted every communication cycle, therefore the values of the AD conversions are newer than one communication cycle.

E.2.2 Special Threshold Inputs

Output data: (on MU)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
EMUL_DIG_INP	O	00h ... FFFFh	0 ... 65535	1	1
Level of special threshold analog inputs					

Note: The scope of this variable is the MU only. It is available on MC only, if the contents is transferred via SPI communication.

FUNCTION DESCRIPTION:

General information:

The analog input pins AIN4 and AIN5 are converted for the emulation of some digital inputs only. This special threshold bits are set, if the voltage is above the upper threshold they are reset if the voltage is below the lower limit. The intermediate values form a Schmitt-triggered hyseresis.

EMUL_DIG_INP

15 (MSB)	14 ... 9	8	7	6	5	4	3	2	1	0 (LSB)
0	0	AD13-level	AD12-level	AD10-level	AD8-level	AD5-Level	AD4-Level	AD3-Level	AD2-Level	AD1-level

Application conditions:

Initialisation: Reset all bits

Recurrence: each A/D conversion

Activation: always

E.3 Digital output port extension

Input data:

OUT_PORT_MU			

FUNCTION DESCRIPTION:

General information:

Due to free resources, the MU is used for digital output port extension. The bitword OUT_PORT_MU is transmited from MC to MU, see "Communcation between MC and MU". The bits of OUT_PORT_MU represent the value of the port pins.

OUT_PORT_MU:

7 (MSB)	6	5	4	3	2	1	0 (LSB)
PF7	PF6	*PF2*	*PB2*	*PB1*	(PA7)	PA6	*PA5*

Note: The pins PB1, PB2, PF2 and PA5 only conducted for the 44pin package. Pin PA7 is reserved for the power off timer (POT), it is only available as output if the POT is permanently disabled.

After a reset or power up the MU initializes the digital ports as input first, see "Communication between MC and MU". In the initialization the pins are configurated as output ports. If the ports are not configurated as outputs, they are switched to a high impedance state.
The pins of the digital output Port Extension are updated within 1ms after communication.

E.3.1 Generation of Diagnosis Pulses on the Digital Output Port Extension

Input data:

NR_DGO_PLS_MU	T_DGO_PLS_MU		

FUNCTION DESCRIPTION:

General information:

The MU offers the possibility to generate diagnosis pulses on the digital output port extension. The MC sends two bytes to the MU. The value NR_DGO_PLS_MU shows the port pin:

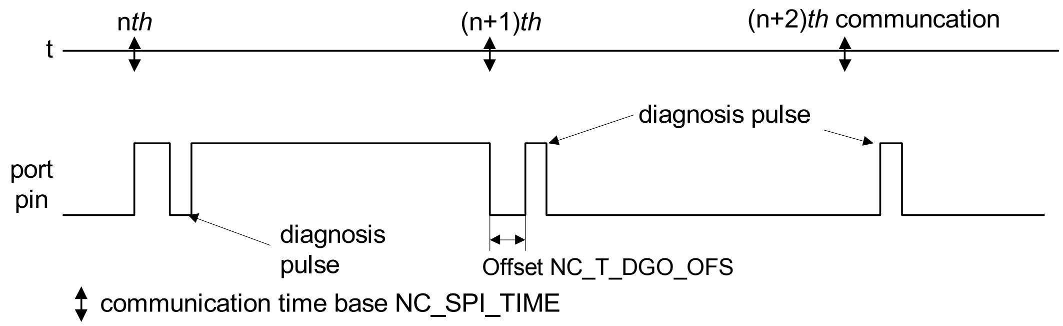
NR_DGO_PLS_MU

NR_DGO_PLS_MU	Pin
1	*PA5*
2	PA6
3	(PA7)
4	*PB1*
5	*PB2*
6	*PF2*
7	PF6
8	PF7

Note: The pins PB1, PB2, PF2 and PA5 only conducted for the 44pin package. Pin PA7 is reserved for the power off timer (POT). It is only available if the POT is permanently disabled.

All other values of NR_DGO_PLS_MU belongs to no diagnosis pulse. The value T_DGO_PLS_MU shows the length of one pulse. The allowed values for T_DGO_PLS_MU is 0 to FFh, the physical limits are 37 to 2069 µs. A fluctiation of about 50µs is possible.

The diagnosis pulse is generated NC_DGO_T_OFS after communication. The level of the pulse is the inverted output lebvel; it high if the output level is low, it is low if the output level is high.



Note:

Due to interrupt response time, T_DGO_PLS_MU has an additional offset of 5.5µs.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_T_DGO_OFS	1	0 ... FFh	0 ... 255	1	Ms
Time to wait after communcation to send diagnosis pulse					

E.4 Communication between MC and MU

Input data: (on MC, transmitted from MU to MC)

SW_VERS_MU	ERR_COD_MU	RST_CTR_MU	ABC_MON2_MU
TEST_REC_IDX_MON2	SYN_PFM_MU	SPI_CKS_MC_MU	SPI_HD_MC_MU
LV_ERR_TMP_MU	NC_ENA_PORT_MU	LV_ERR_CONV_AN_MU	

Note: These data are sent from the MU and received by the MC. They are available on MC after the dedicated communication

Input data: (on MU, from function "Analog and pseudo digital inputs")

AN_BUF_0	AN_BUF_1	AN_BUF_2	AN_BUF_3
AN_BUF_8	AN_BUF_10	AN_BUF_12	AN_BUF_13

Output data: (on MC):

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
ACQ_MU[NC_ACQ_MU_NR]	V/O	000h ... 3FFh FFFFh	0 ... 4.9951 "undefined"	5/1024 -	V -
10 bit A/D value on MC, FFFFh simply denotes "undefined" as the result of the AD conversion					
EMUL_DIG_INP	V/O	00h ... FFFFh	0 ... 65535	1	1
Level of special threshold analog inputs					

Note: On the MU, the input data for the ADC channels are AN_BUF_x which are described in "Analog and pseudo digital inputs." ACQ_MU is an array with the index ranging from 0 to NC_ACQ_MU_NR-1. During the monitoring communications (except the INIT state communication) from MU to MC, AN_BUF_0 is mapped to ACQ_MU[0], see the subsections "Predrive State of Communication", "Normal State of Communication", and "Disable State of Communication"; during the AD communications from MU to MC, the AN_BUF_x, for x > 0, are mapped to entries of ACQ_MU[y] as explictily described in the subsections "AD Communication 1" and "AD Communication 2."

Transferred data to MU, input on MC, output on MU:

SW_COMP_NR_MC	OUT_PORT_MU	NR_DGO_PLS_MU	T_DGO_PLS_MU
RESP_MON2_MC	RESP_PFM_MC	SPI_CKS_MC_MU	SPI_HD_MC_MU
RST_CTR_MC	LV_DR_OFF_MU_MON		

Note: These data are sent to MU, the values have to be assigned on MC before the dedicated communication.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_T_SPI_COM	1	00h ... FFh	0...255	1	ms
Nominal time grid for monitoring communication					
NC_T_SPI_MAX	1	00h ... FFh	0...255	1	ms
Maximum time periode between two consecutive monitoring communications					
NC_T_SPI_MIN	1	00h ... FFh	0...255	1	ms
Minimum time periode between two consecutive monitoring communications					
NC_ACQ_MU_NR	1	00h ... 07h	0 ... 7	1	1
Maximum index value of the array ACQ_MU					

E.4.1 Tranmission protocol and Monitoring communication

E.4.1.1 SPI transmission protocol

FUNCTION DESCRIPTION:

The communication is effected in full duplex mode (fdx) according to the SPI transmission protocol. MC and MU are connected to the SPI bus of the ECU. Regarding the SPI protocol, there is physically one master and one or several slaves.

As the spi communication to any other slaves must not affect the monitoring communication at all, the monitoring communication has the highest priority. The MC is only allowed to start other communications (see also “AD communications”) if this non monitoring communication does not affect timeout of the monitoring communication.

The master controls the character exchange by generation of the clock according to the baud rate. In the case of a communication between MC and MU, there is a master (MC) and a slave (MU). The MC initiates the character exchange with the chip select line.

General information:

During operation of the ECU, MC and MU permanently exchange data in terms of messages which assist the MU in monitoring the correct execution of the monitoring software in the MC.

The communication runs through an INIT-State, PREDRIVE-State, NORMAL-State. The DISABLE state is related to detected faults, see chapter “Disable State of Communication”.

An exception is the EOLT-SW (end of line test). An overview of the EOLT mode is given in chapter “Communication between MC and MU”.

Figure 1 gives an overview of all possible states of the communication.

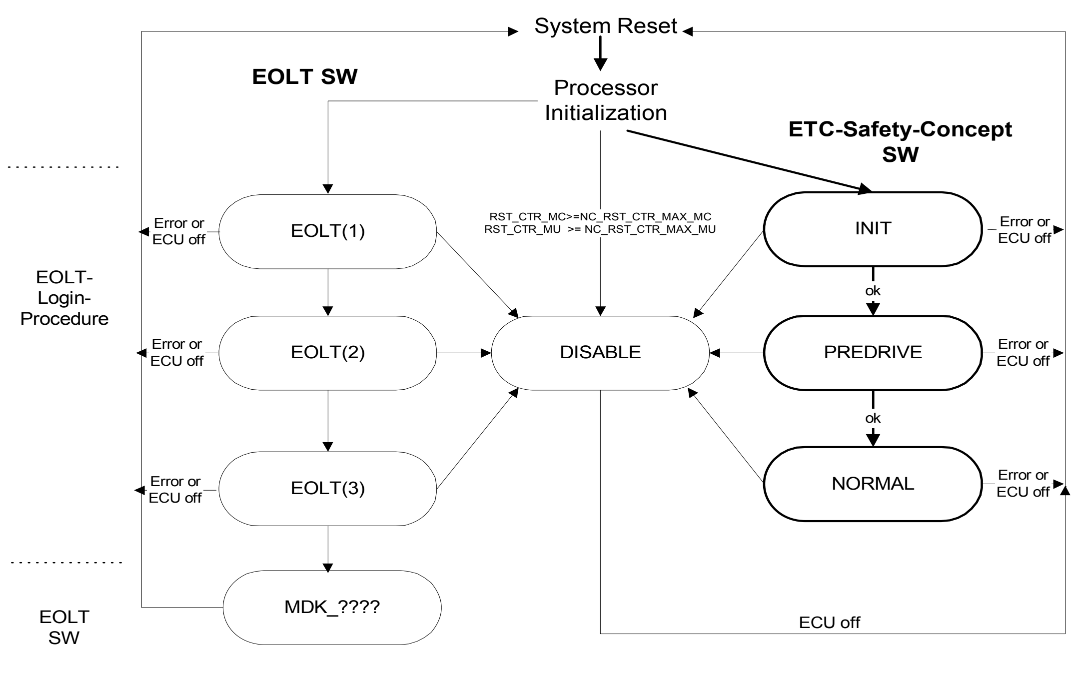


Figure 1: States of the communication

The transitions of the states are controlled by the communication between MC and MU. In the different states different actions are carried out.

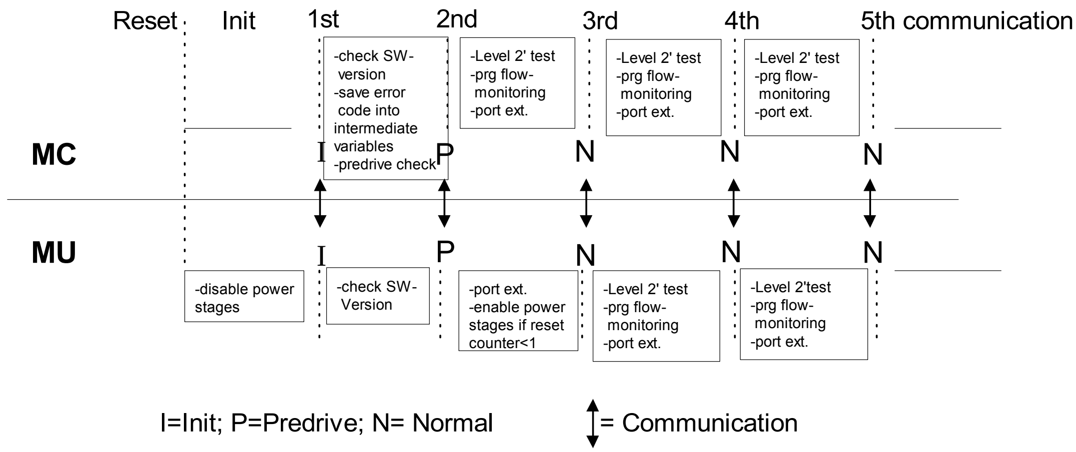


Figure 2: Time Chart of communications in fault free case

E.4.1.1.1 Structure of transferred blocks

FUNCTION DESCRIPTION:

There exist buffers for receiving data and sending data: SPI_BUF_RECV and SPI_BUF_SEND on MU, SPI_TRM_BUF_MU and SPI_RCV_BUF_MU on MC. At runtime all four of these buffers have the same size. The data to be sent have to be copied into the buffers before the communication, the reveiced data has to be copied into the variables after the fault free communication. Each communication the contents of the buffers are transferred:

- MC sends to MU / MU receives from MC (SPI_TRM_BUF_MU->SPI_BUF_RECV)
- MU sends to MC / MC receives from MU (SPI_BUF_SEND->SPI_RCV_BUF_MU)

A monitoring communication block consists of 8 bytes.

In a block sent from MC to MU byte 0 and 1 are the header, whereas bytes 2 to 6 form the data. In a block sent from MU to MC byte 0 is a dummy byte, byte 1 is the heaer byte.

According to the header the interpretation of the data block is done. Because the MC triggers the communication it is neccesary that MU interprets the first header byte during the communication. So after reception of the first byte this byte is interpreted as header by MU before the next byte is transferred. Due to full duplex communication mode the MU has to sent a dummy byte in the first byte of communication block.

The byte 7 is the checksum of the bytes 0 till 6 and is used for monitoring the communication.

Sent monitoring communication block from MC to MU:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Header	Header	Data	Data	Data	Data	Data	Chksum

Sent monitoring communication block from MU to MC:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Dummy	Header	Data	Data	Data	Data	Data	Chksum

Definition of header and dummy bytes

	Hex	Bit
INIT	12h	0001 0010
PREDRIVE	CFh	1100 1111
NORMAL	ADh	1010 1101
AD_1	7Eh	0111 1110
AD_2	7Fh	0111 1111
EOLT	85h	1000 0101
DISABLE	3Bh	0011 1011
DUMMY	00h	0000 0000

In the following sections the data bytes, byte 2 till byte 6, are not described explicitly, instead only the variable names forming the data block are listed. If not all bytes of the data block are used, the remaining have to be filled with DUMMY.

E.4.1.1.2 Init State of Communication

FUNCTION DESCRIPTION:

At the beginning of the communication (see **Figure 2**), MC and MU transmit the following data in the INIT communication:

Init State transfer data

Byte No.	MC->MU	MU->MC
0	SPI_HD_MC_MU	DUMMY
1	SPI_HD_MC_MU	SPI_HD_MU_MU
2	RST_CTR_MC, SW_COMP_NR_MC, ENA_PORT_MU	ERR_COD_MU, RST_CTR_MU, NC_SW_VERS_MU->SW_VERS_MU
3		
4		
5		
6		
7	SPI_CHKS_MC_MU	SPI_CHKS_MU_MU

The values of the header bytes are INIT

The contents of data are described in the chapters “Processor Monitoring”, “Fault Reaction of Processor Monitoring” and "Digital Output Port Extension".

E.4.1.1.3 Predrive State of Communication

FUNCTION DESCRIPTION:

In fault free case the second communication (see **Figure 2**) is the PREDRIVE communication. The following data are transmitted :

Predrive State transfer data

Byte No.	MC->MU	MU->MC
0	SPI_HD_MC_MU	DUMMY
1	SPI_HD_MC_MU	SPI_HD_MU_MU
2	ENA_PORT_MU, OUT_PORT_MU, NR_DGO_PLS_MU, T_DGO_PLS_MU	TEST_REC_IDX_MON2, ABC_MON2_MU, LV_ERR_CONF_AN_MU, AN_BUF_0->ACQ_MU[0], LV_ERR_TMP_MU, SYN_PFM_MU
3		
4		
5		
6		
7	SPI_CHKS_MC_MU	SPI_CHKS_MU_MU

The values of header bytes are PREDRIVE

E.4.1.1.4 Normal State of Communication

FUNCTION DESCRIPTION:

In fault free case the third and all following communication (see **Figure 2**) are NORMAL communications. The following data are transmitted:

Normal State transfer data

Byte No.	MC->MU	MU->MC
0	SPI_HD_MC_MU	DUMMY
1	SPI_HD_MC_MU	SPI_HD_MU_MU
2	RESP_PFM_MC, RESP_MON2_MC, LV_DR_OFF_MU_MON, OUT_PORT_MU, NR_DGO_PLS_MU, T_DGO_PLS_MU	TEST_REC_IDX_MON2, ABC_MON2_MU, LV_ERR_CONV_AN_MU, AN_BUF_0->ACQ_MU[0], LV_ERR_TMP_MU, SYN_PFM_MU
3		
4		
5		
6		
7	SPI_CHKS_MC_MU	SPI_CHKS_MU_MU

The values of the header bytes are NORMAL

E.4.1.1.5 Disable State of Communication

Input data:

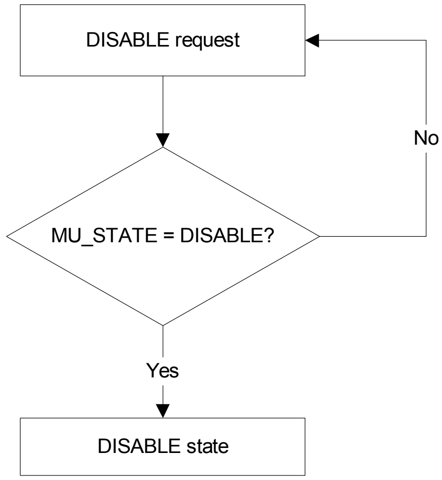
SPI_HD_MC_MU	SPI_HD_MU_MU	LV_ERR_MC	

FUNCTION DESCRIPTION:

There are several ways to change into DISABLE-State.

1. If the MC detects failures and the value of the reset counter RST_CTR_MC already reached its maximum. In this case LV_ERR_MC is set.
2. If the MU detects failures and the value of the reset counter RST_CTR_MU already reached its maximum. In this case LV_ERR_MU is set.

3. When shutting down the ECU, it is necessary to stop the processor monitoring. This is initialized by the MC, MU receives a DISABLE command next communication:



4. The MU receives a DISABLE-Command (header SPI_HD_MC_MU) from the MC.
5. The MC receives a DISABLE-Command (header SPI_HD_MU_MU) from the MU.

Disable State transfer data

<table>
<tr><th>Byte No.</th><th>MC->MU</th><th>MU->MC</th></tr>
<tr><td>0</td><td>SPI_HD_MC_MU</td><td>DUMMY</td></tr>
<tr><td>1</td><td>SPI_HD_MC_MU</td><td>SPI_HD_MU_MU</td></tr>
<tr><td>2</td><td rowspan="5">RST_CTR_MC, OUT_PORT_MU,
NR_DGO_PLS_MU, T_DGO_PLS_MU</td><td rowspan="5">ERR_COD_MU, RST_CTR_MU,
AN_BUF_0->ACQ_MU[0]</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td>SPI_CKS_MC_MU</td><td>SPI_CKS_MU_MU</td></tr>
</table>

The values of the header bytes are DISABLE

So, if either LV_ERR_MC or LV_ERR_MU is set the reason for the transition into DISABLE state is a fault detected by the Processor Monitoring. Otherwise (neither LV_ERR_MC is set nor LV_ERR_MU is set) the transition is based on shutting down the ECU.

If one of the conditions mentioned above is fulfilled, the MU disables the power stages and the following monitoring functions are deactivated:

- Program Flow Monitoring
- Communication Monitoring
- Copy of process monitoring

Still avaliable in DISABLE state are:

- Analog input port extensions, including AD-communication
- Digital output port extensions, including generation of diagnosis pulses

E.4.2 Request for Transition to Disable State

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MU_DI_STATE	O/V	0h 1h 2h 3h	MU_DI_NOT_REQ MU_DI_BUSY MU_DI_ERR MU_DI_OK	1	1
State of MU Disable request					

Export action:

ACTION_ECM3_DisableRequest(OUT <flag>)
Action to perfom a transition to DISABLE

FUNCTION DESCRIPTION:

General information:

Like given above ("Disable State of Communication") a transition to Disable can be performed without any Processor Monitoring fault. This transition has to be performed for switching off the monitoring functionality of the MU in case of entering the power latch phase or for shutting down the ECU. This transition can be triggered by any other parts of the software calling cyclical ACTION_ECM3_DisableRequest().

To initialize this transition to Disable the MC sends a DISABLE command (header) to the MU. The MU has to respond by sending a Disable header next communication.

The variable MU_DI_STATE indicates the state of the MU (seen form MC) according to the Disable request. There are several states:

MU_DI_STATE		Meaning
0h	MU_DI_NOT_REQ	No disable request
1h	MU_DI_BUSY	Disable requested. MU did not respond up till now and no timeout
2h	MU_DI_ERR	Disable requested. MU did not respond within the defined time periode
3h	MU_DI_OK	Disable requested. MU responded with Disable header

Application conditions:

Initialisation: MU_DI_STATE = 0h /* No disable request */
After each power up or reset

Description of ACTION_ECM3_DisableRequest( ):

Description:

Syntax: **ACTION_ECM3_DisableRequest(OUT <*flag*>)**

Parameter (OUT): ***flag***

Formula section:

ACTION_ECM3_DisableRequest(SYNCHRONISATION ASYNCHRONOUS)

- Requests transition to Disable
- Sets the parameter ***flag*** according to
  - set, if (MU_DI_STATE=MU_DI_OK) or (MU_DI_STATE=MU_DI_ERR)
  - reset in all other cases.

E.4.3 Communication dedicated to Port Extension

The port extensions are described in “Digital output port extension” and “Analog and pseudo digital inputs”. Within the following sections the initialisation and data transfer are described.

E.4.3.1 Initialisation of the Port Extensions

Input data: (on MU)

ENA_PORT_MU			

FUNCTION DESCRIPTION:

During initialisation the digial port extensions has to be configurated.

After power up or reset the system runs through different states, see “Processor Monitoring”.

The digital output port extensions are configurated with the transferred bit pattern ENA_PORT_MU sent to MU in the INIT and PREDRIVE communication. If these two received values are identical and plausible, the value is stored in a reset safe RAM area on MU. If the two values are not identically or any of them is not plausible, a communication error is assumed, see “Fault Reactions on Processor Monitoring”.

After all other resets running not through the INIT and PREDRIVE states this reset safe stored value is used for initialisation.

ENA_PORT_MU has the same structure as OUT_PORT_MU and indicates which pins are avaliable for digital output port extension, if the dedicated bit in ENA_PORT_MU is set. The structure of OUT_PORT_MU is described in “Digital output port extension”.

On MC the value sent in the INIT and PREDRIVE communication is a constant, NC_ENA_PORT_MU.

If it is not possible to set the digital output pins according to the received byte ENA_PORT_MU (caused i.ie. by a communication error) the MU enables all digital output pins.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ENA_PORT_MU	1	00h ... FFh	0 ... 255	1	-
Bit pattern on MC, set to MU for initializing the digital output port extension					

E.4.3.2 AD communications

After the INIT communcation the AD-communications can be performed between two monitoring communcations, if the timing of the monitoring communication is not affected, see “SPI transmission protocol”. An AD communication is triggered like every communication by the MC. Therefore the MC sends the according header bytes. With interpreting the first received header byte the MU knows the structure and contents of the block to be sent to MC. As the communication is full duplex, the MC has to sent the value DUMMY in the data bytes.

The structure and contents of the transferred data are described in "Analog and pseudo digital inputs".

Note how duing the two AD communications, the data in AN_BUF_x are mapped to the entries of ACQ_MU[y].

E.4.3.2.1 AD Communication 1

In this communication the following ADC values are transferred:

Byte No.	MC->MU	MU->MC
0	SPI_HD_MC_MU	DUMMY
1	SPI_HD_MC_MU	SPI_HD_MU_MU
2	DUMMY	AN_BUF_1 -> ACQ_MU[1], AN_BUF_8 -> ACQ_MU[2], AN_BUF_12 -> ACQ_MU[3], AN_BUF_13 -> ACQ_MU[4]
3		
4		
5		
6		
7	SPI_CKS_MC_MU	SPI_CKS_MU_MU

The values of the header bytes are AD_1

E.4.3.2.2 AD Communication 2

In this communication the following ADC values are transferred:

Byte No.	MC->MU	MU->MC
0	SPI_HD_MC_MU	DUMMY
1	SPI_HD_MC_MU	SPI_HD_MU_MU
2	DUMMY	EMUL_DIG_INP, AN_BUF_2 -> ACQ_MU[5], AN_BUF_3 -> ACQ_MU[6], AN_BUF_10 -> ACQ_MU[7]
3		
4		
5		
6		
7	SPI_CKS_MC_MU	SPI_CKS_MU_MU

The values of the header bytes are AD_2

E.4.4 Monitoring of communication

FUNCTION DESCRIPTION:

During operation, MC and MU exchange exclusively messages with a length of 8 characters. Monitoring of the communication is done on both processors, MC and MU. The contents of the received data buffer is only evaluated in case of faultless communication. The nominal value of the cycle time is NC_T_SPI_COM. The cycle time is affected by runtime fluctuations of the operating system of MC. Shorter or longer messages are recognized as inadmissible and result in error treatments. The characters within the message are transferred in direct succession. Thus, the message transfer time is essentially fixed (minor fluctuations are possible due to other activated interrupts in MU).

E.4.4.1 Communication monitoring on MC

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ABC_HD_MC	-	00h ... FFh	0...255	1	-
anti-bounce counter for command/header monitoring					
ABC_CKS_MC	-	00h ... FFh	0...255	1	-
anti-bounce counter for checksum monitoring					
SPI_CKS_MC_MU	-	00h ... FFh	0...255	1	-
Checksum of the MC ->MU communication					
SPI_CKS_CLC_MU_MU	-	00h ... FFh	0...255	1	-
Calculated checksum of the MU ->MC communication					
SPI_HD_MC_MU	-	00h ... FFh	0...255	1	-
Header of the MC ->MU communication					
MC_STATE	-	12h CFh ADh 3Bh	"INIT" "PREDRIVE" "NORMAL" "DISABLE"	1	-
State of the communication on MC					

Input data:

SPI_CKS_MU_MU	SPI_HD_MU_MU		

FUNCTION DESCRIPTION:

The MC checks the checksum of the transferred block. The MU calculates a checksum (SPI_CKS_MU_MU) and sends this checksum in the last byte, see “Structure of transferred blocks”. After block reception the MC calculates the checksum with the same algorithm (SPI_CKS_CLC_MU_MU) and compares it with the sent one.

Formula section:

IF (SPI_CKS_CLC_MU_MU <> SPI_CHKS_MU_MU) **THEN**

IF (MC_STATE==INIT) || (MC_STATE==PREDRIVE) **THEN**

ABC_CKS_MC = NC_ABC_MAX_CKS_MC

central failure reaction carried out (refer to chapter: “Fault reaction”).

ELSE

ABC_CKS_MC += NC_ABC_INC_CKS_MC

IF (ABC_CKS_MC >= NC_ABC_MAX_CKS_MC) **THEN**

central failure reaction carried out

(refer to chapter: “Fault reaction”).

ENDIF

ENDIF

ELSE

IF (ABC_CKS_MC>0) **THEN**

ABC_CKS_MC –

ENDIF

ENDIF

Furthermore the MC checks, if the MU sends the right command/header (INIT, PREDRIVE, NORMAL) at the right moment. Therefore the received header must be plausible to the state of the MC.

In case of fault free communication the MC state stored in MC_STATE is updated. Allowed values of MC_STATE are INIT, PREDRIVE, NORMAL and DISABLE. The headers AD_1 and AD_2 do not change the state.

The following transitions are allowed:

MC_STATE (before communication)	**SPI_HD_MU_MU** (received header)	**MC_STATE** (after communcation)
INIT	PREDRIVE	PREDRIVE
	DISABLE	DISABLE
	AD_1	INIT
	AD_2	INIT
PREDRIVE	NORMAL	NORMAL
	DISABLE	DISABLE
	AD_1	PREDRIVE
	AD_2	PREDRIVE
NORMAL	NORMAL	NORMAL
	DISABLE	DISABLE
	AD_1	NORMAL
	AD_2	NORMAL
DISABLE	INIT	DISABLE
	PREDRIVE	DISABLE
	NORMAL	DISABLE
	DISABLE	DISABLE
	AD_1	DISABLE
	AD_2	DISABLE

All other combinations are defined as wrong header.

Formula section:

```
IF (wrong header) THEN
    IF ( MC_STATE == INIT ) or ( MC_STATE == PREDRIVE ) THEN
        ABC_HD_MC = NC_ABC_MAX_HD_MC
        central failure reaction is carried out
        (refer to chapter: “Fault reaction”).
    ENDIF
    IF  ( MC_STATE == NORMAL)
        ABC_HD_MC+=NC_ABC_INC_HD_MC
        IF (ABC_HD_MC >= NC_ABC_MAX_HD_MC)THEN
            central failure reaction is carried out.
        ENDIF
    ENDIF
ELSE
    IF (ABC_HD_MC>0) THEN ABC_HD_MC—; ENDIF
ENDIF
```

Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_ABC_INC_HD_MC	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for monitoring of the command/header					
NC_ABC_MAX_HD_MC	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for monitoring of the command/header					
NC_ABC_INC_CKS_MC	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for monitoring of the checksum					
NC_ABC_MAX_CKS_MC	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for monitoring of the checksum					

E.4.4.2 Communication monitoring on MU

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ABC_TOUT_MAX_MU	-	00h ... FFh	0...255	1	-
Anti-bounce counter for maximum timeout monitoring					
ABC_TOUT_MIN_MU	-	00h ... FFh	0...255	1	-
Anti-bounce counter for minimum timeout monitoring					
T_SPI_COM	-	00h ... FFh	0...255	1	ms
Counter to measure the time since the last communication					
ABC_HD_MU	-	00h ... FFh	0...255	1	-
Anti-bounce counter for monitoring of the command/header					
ABC_CKS_MU	-	00h ... FFh	0...255	1	-
Anti-bounce counter for monitoring of the checksum					
SPI_CKS_MU_MU	-	00h ... FFh	0...255	1	-
Checksum of the MU ->MC communication (basic block)					
SPI_CKS_CLC_MC_MU	-	00h ... FFh	0...255	1	-
Actual checksum of the MC ->MU communication					
SPI_HD_MU_MU	-	00h ... FFh	0...255	1	-
Header of the MU ->MC communication					
MU_STATE	-	12h CFh ADh 3Bh	"INIT" "PREDRIVE" "NORMAL" "DISABLE"	1	-
State of the communication on MU					

Input data:

SPI_CKS_MC_MU	SPI_HD_MC_MU		

FUNCTION DESCRIPTION:

The communication cycle time is affected by runtime fluctuations of the operating system of MC. MU monitors the cycle time. This cycle time has to fulfill: NC_T_SPI_MIN ≤ T_SPI_COM ≤ NC_T_SPI_MAX. T_SPI_COM is the measured, elapsed time since the last monitoring communication. The timeout monitoring starts with the INIT communication, it is not carried out in DISABLE.

Application conditions:

Initialisation: ABC_TOUT_MIN_MU=ABC_TOUT_MAX_MU=
ABC_HD_MU=ABC_CKS_MU=0 after reset or power up

Formula section:

IF (MU_STATE==PREDRIVE) or (MU_STATE==NORMAL) **THEN**

    IF (T_SPI_COM > NC_T_SPI_MAX) **THEN**

        ABC_TOUT_MAX_MU += NC_ABC_INC_TOUT_MAX_MU

        IF (ABC_TOUT_MAX_MU >= NC_ABC_MAX_TOUT_MAX_MU) **THEN**

            central failure reaction is carried out

            (refer to chapter: “Fault reaction”)

```
            ENDIF
        ELSE
            IF (ABC_TOUT_MAX_MU>0) THEN ABC_TOUT_MAX_MU --; ENDIF

            IF (T_SPI_COM < NC_T_SPI_MIN) THEN
                ABC_TOUT_MIN_MU += NC_ABC_INC_TOUT_MIN_MU
                IF (ABC_TOUT_MIN_MU >= NC_ABC_MAX_TOUT_MIN_MU) THEN
                    central failure reaction is carried out.
                ENDIF
            ELSE
                IF (ABC_TOUT_MIN_MU>0) THEN ABC_TOUT_MIN_MU --; ENDIF
            ENDIF
        ENDIF
    ENDIF
    T_SPI_COM = 0
```

The MC calculates a checksum of its send message. This byte is sent as last (SPI_CHKS_MC_MU). After the reception of the message, the MU calculates the checksum (SPI_CKS_CLC_MC_MU) and compares this one with the received checksum.

Formula section:

```
IF (SPI_CHKS_MC_MU<>SPI_CKS_CLC_MC_MU) THEN
    IF ( MU_STATE == INIT ) or ( MU_STATE == PREDRIVE ) THEN
        ABC_CKS_MU = NC_ABC_MAX_CKS_MU
        central failure reaction is carried out
        (refer to chapter: “Fault reaction”).
    ENDIF
    IF  ( MU_STATE == NORMAL)
        ABC_CKS_MU+=NC_ABC_INC_CKS_MU
        IF (ABC_CKS_MU >= NC_ABC_MAX_CKS_MU)THEN
            central failure reaction is carried out.
        ENDIF
    ENDIF
ELSE
    IF (ABC_CKS_MU>0) THEN ABC_CKS_MU—; ENDIF
ENDIF
```

Furthermore the MU checks, if the MC sends the right command/header (INIT, PREDRIVE, NORMAL, AD_1, AD_2) at the right moment. Therefore the received header must be plausible to the state of the MU.

In case of fault free communication the MU state stored in MU_STATE is updated. Allowed values of MU_STATE are INIT, PREDRIVE, NORMAL and DISABLE. The headers AD_1 and AD_2 do not change the state.

The following transitions are allowed:

MU_STATE (before communication)	**SPI_HD_MC_MU** (received header)	**MU_STATE** (after communcation)
INIT	PREDRIVE	PREDRIVE
	DISABLE	DISABLE
	AD_1	INIT
	AD_2	INIT
PREDRIVE	NORMAL	NORMAL
	DISABLE	DISABLE
	AD_1	PREDRIVE
	AD_2	PREDRIVE
NORMAL	NORMAL	NORMAL
	DISABLE	DISABLE
	AD_1	NORMAL
	AD_2	NORMAL
DISABLE	INIT	DISABLE
	PREDRIVE	DISABLE
	NORMAL	DISABLE
	DISABLE	DISABLE
	AD_1	DISABLE
	AD_2	DISABLE

All other combinations are defined as wrong header.

Formula section:

IF (wrong header) **THEN**

IF (MU_STATE == INIT) or (MU_STATE == PREDRIVE) **THEN**

ABC_HD_MU = NC_ABC_MAX_HD_MU

central failure reaction is carried out

(refer to chapter: “Fault reaction”).

ENDIF

IF (MU_STATE == NORMAL)

ABC_HD_MU+=NC_ABC_INC_HD_MU

IF (ABC_HD_MU >= NC_ABC_MAX_HD_MU)**THEN**

central failure reaction is carried out.

ENDIF

ENDIF

ELSE

IF (ABC_HD_MU>0) **THEN** ABC_HD_MU—; **ENDIF**

ENDIF

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ABC_INC_TOUT_MAX_MU	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for maximum timeout monitoring					
NC_ABC_MAX_TOUT_MAX_MU	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for maximum timeout monitoring					
NC_ABC_INC_TOUT_MIN_MU	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for maximum timeout monitoring					
NC_ABC_MAX_TOUT_MIN_MU	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for maximum timeout monitoring					
NC_ABC_INC_HD_MU	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for monitoring of the command/header					
NC_ABC_MAX_HD_MU	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for monitoring of the command/header					
NC_ABC_INC_CKS_MU	1	00h ... FFh	0...255	1	-
Increment value of anti-bounce counter for monitoring of the checksum					
NC_ABC_MAX_CKS_MU	1	00h ... FFh	0...255	1	-
Maximum value of anti-bounce counter for monitoring of the checksum					

E.4.5 End of Line Test

The software of the MU comprises a test mode. It can be activated with special commands by the SPI-Communication. If the test mode is activated, the software of the Safety-Concept is not carried out. The only way to leave the test mode is a reset.

E.4.5.1 Entry into EOLT-SW

The EOLT-Mode can be reached if the first command after reset or after DISABLE-Mode is „EOLT“. The MU tests the conditions for entry into the EOLT. The MC must send the three special messages EOLT(1), EOLT(2) and EOLT(3) in a precise time window, before the MU jumps into EOLT-SW.

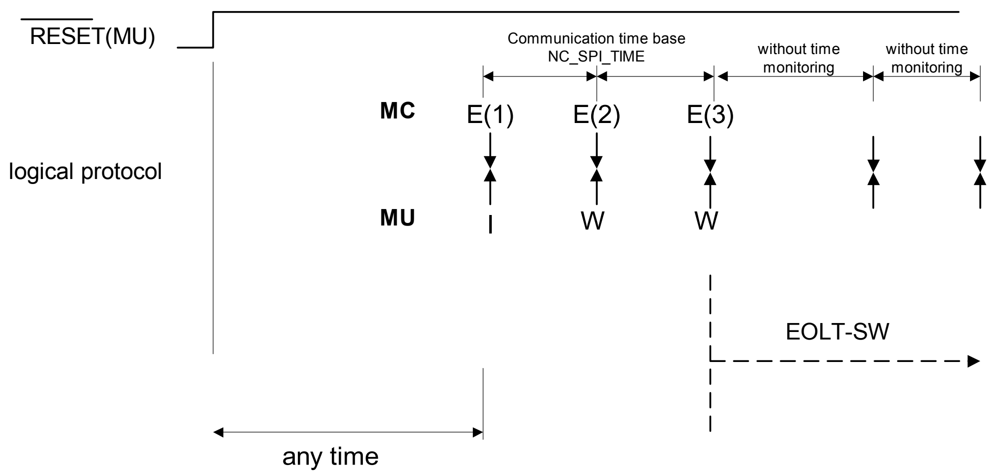


Figure 3: Time chart of entry EOLT-SW

Explanation of

Figure 3

E(1) = EOLT(1)

E(2) = EOLT(2)

E(3) = EOLT(3)

I = INIT
W = WRONG

E.4.5.2 EOLT Commands

After a successful entry procedure, different commands can be called. The communication between MC and MU is full duplex communication. Therefore it is necessary to call two communications for every command. The first to send the command to MU and the second to receive the answer from the MU.

These commands can be sent to MU during the EOLT

Command	Value	description
MDK_READ_ANA	01h	Read the four analog inputs
MDK_READ_PORT	02h	Read all digital inputs
MDK_WRITE_PORT	03h	Write all digital outputs
MDK_BIST	04h	Read the selftest-result
MDK_READ_MEMORY	05h	Read the memory of the MU

E.4.6 Availability of AD values in MU

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
ACQ_MU[NC_ACQ_MU_NR]	V/O	000h ... 3FFh FFFFh	0 ... 4.9951 “undefined”	5/1024 -	V -
10 bit A/D value on MC, FFFFh simply denotes “undefined” as the result of the AD conversion					

Input data (transferred from MU to MC):

AN_BUF_0	AN_BUF_1	AN_BUF_2	AN_BUF_3
AN_BUF_8	AN_BUF_10	AN_BUF_12	AN_BUF_13

Additionally, indicators when monitoring, AD_1 or AD_2 communications are faulty are needed; refer to section “Communication monitoring on MC.”

FUNCTION DESCRIPTION:

General information:

This function describes a mechanism to determine whether new AD values are available on the MU, and whether the AD-related communiations were faulty.
In order to achieve this, the data ACQ_MU[x] on the MC shall be initialised to FFFFh (which denotes an undefined singal by the AD converter, see above range definition of ACQ_MU[y]).
In this way, the MC can detect whether valid new data have been transmitted by the MU or not; as long as no values are available, the old values are not overwritten and so the stay undefined for as long as necessary.
Similarly, should one of the AD-related communications fail, the corresponding entries in ACQ_MU[y] (see “Predrive State of Communication”, “Normal State of Communication”, “Disable State of Communication”, “AD Communication 1”, and “AD Communication 2”) shall be set to FFFFh, again to signal to the MC that there are no currently valid AD values available on the MU.

Application conditions:

Initialisation: Preset all elements of the array ACQ_MU with FFFFh

Recurrence: each communication (Predrive, Normal, Disable, AD1 or AD2)

Activation: always

Formula section:

IF ("last communication faultless") **THEN**

"Assign the value of AN_BUF_x to the corresponding entry ACQ_MU[y] as described in "Predrive State of Communication", "Normal State of Communication", "Disable State of Communication", "AD Communication 1", and "AD Communication 2."

ELSE

IF ("monitoring communication faulty") **THEN**

ACQ_MU[0] = FFFFh

ELIF ("AD_1 communication faulty") **THEN**

ACQ_MU[1] = FFFFh

ACQ_MU[2] = FFFFh

ACQ_MU[3] = FFFFh

ACQ_MU[4] = FFFFh

ELIF ("AD_2 communication faulty") **THEN**

ACQ_MU[5] = FFFFh

ACQ_MU[6] = FFFFh

ACQ_MU[7] = FFFFh

ENDIF

ENDIF

E.5 PROCESS MONITORING

Import actions:

ACTION_ECM3_ServicePfm(IN <n>)

Note: This action is defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". This action has to be executed as last instruction after all tasks has been executed. After the first block of tasks the action has to be called with argument 1, after the second with 2, after the third with 3 and after the fourth with 4. At the very end of the scheduler finally the action has to be called with argument 6, 7, and 0.

ACTION_ECM2_LockPws()
ACTION_ECM2_UnlockPws()

Note: These two actions are defined in module "Fault Reaction of process monitoring".

Input data:

LV_RLY_MAIN_DLY_ERR	LV_DIAG_END_RLY_MAIN_DLY		

General information:

The task of process monitoring is to check whether ETC / torque functions (located in function level) work properly and to take a fault reaction, if not. The task scheduler defines the correct sequencing of all called monitoring functions within the four task slots (see figure 1) and the related action calls for program flow monitoring (see “import actions”)

Formula Section:

IF LV_DIAG_END_RLY_MAIN_DLY == 1 **AND** LV_RLY_MAIN_DLY_ERR == 0
THEN
ACTION_ECM2_UnlockPws();
/* Has to be executed exactly once! */
Tasks/ECM2-functions activated according to table 1 starting with task "1";
ELSE ACTION_ECM2_LockPws();
ENDIF

Table 1: Distribution of the process monitoring functionality over four tasks. The task scheduler is executed every 10 ms. Within one task, the different functions are executed in the order according to the rows of the table.

Task 1 (k)	Task 2 (k+10 ms)	Task 3 (k+20 ms)	Task 4 (k+30 ms)
Application incidences of process monitoring(k)	Application incidences of process monitoring(k+10ms)	Application incidences of process monitoring(k+20ms)	Application incidences of process monitoring(k+30ms)
Monitoring of A/D-conversion	Monitoring of pedal value signals	Monitoring of engine speed limitation	Actual indicated engine torque
Monitoring of TCO	Monitoring of mass air flow	Monitoring of CAN signals	Desired indicated engine torque
Monitoring of engine speed	Monitoring of torque losses	Monitoring of cruise control conditions	Monitoring of actual indicated engine torque
Monitoring of idle speed controller	Monitoring of maximum torque at clutch	Monitoring of minimum torque at clutch	Fault reaction of process monitoring
ACTION_ECM3_ ServicePfm(1)	ACTION_ECM3_ ServicePfm(2)	ACTION_ECM3_ ServicePfm(3)	ACTION_ECM3_ ServicePfm(4)
ACTION_ECM3_ ServicePfm(0)	ACTION_ECM3_ ServicePfm(0)	ACTION_ECM3_ ServicePfm(0)	Error memory management of process monitoring
			ACTION_ECM3_ ServicePfm(6)
			ACTION_ECM3_ ServicePfm(7)
			ACTION_ECM3_ ServicePfm(0)

E.6 Debounce Mechanism

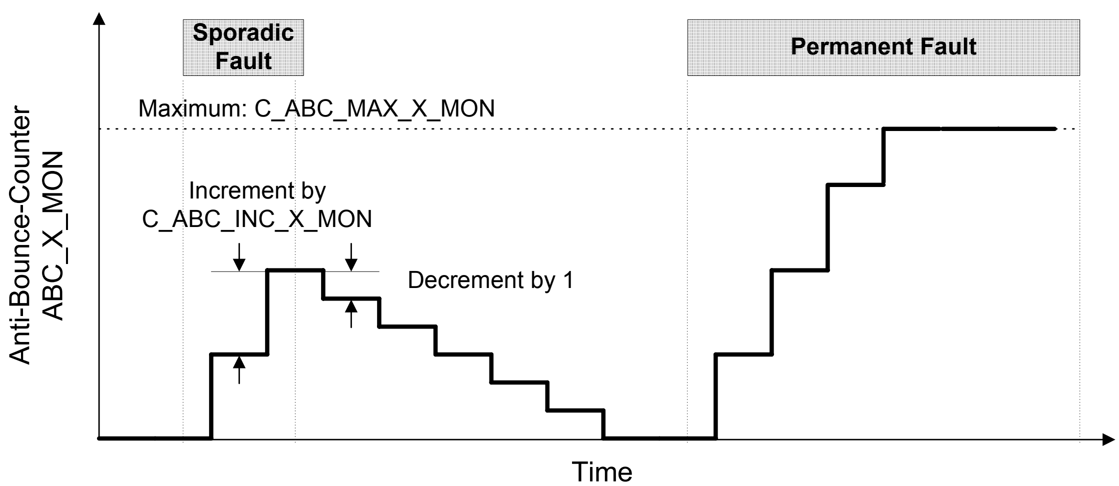
FUNCTION DESCRIPTION:

General information:

Many diagnosis algorithms need a debounce mechanism in order to avoid a faulty diagnosis caused by normal disturbances. Since the process monitoring is intended to represent redundancy to the functions of level 1, there is also a redundant debounce mechanism required. That means an algorithm with separate code, ROM- and RAM data. The debounce algorithm is used for the functions of level 2 and 2' (copy of process monitoring).

Description:

The anti-bounce-counter ABC_X_MON is increased by C_ABC_INC_X_MON, if the conditions for fault detection are fulfilled. Otherwise, the anti-bounce counter is decremented by 1. If the anti-bounce-counter ABC_X_MON exceeds its threshold C_ABC_MAX_X_MON the first time during this engine running, the relating error-flag is set and a fault can be stored in the customer error memory.



Application conditions:

The activation, deactivation and recurrency of this module depends on the data of the related diagnosis function.

Formula section:

```
IF   (Fault detection condition fulfilled)
THEN
     Increment ABC_X_MON by C_ABC_INC_X_MON
     (limited to C_ABC_MAX_X_MON)
ELSE
     Decrement ABC_X_MON by 1          (limited to 0)
```

E.7 Application Incidences and Configuration of Process Monitoring

Import actions:

ACTION_ECM3_Service0TaskPfm(0)
ACTION_ECM3_Service1TaskPfm(0)
ACTION_ECM3_Service0TaskPfm(1)
ACTION_ECM3_Service1TaskPfm(1)
ACTION_ECM3_Service0TaskPfm(2)
ACTION_ECM3_Service1TaskPfm(2)
ACTION_ECM3_Service0TaskPfm(3)
ACTION_ECM3_Service1TaskPfm(3)
ACTION_ECM3_Service0TaskPfm(4)
ACTION_ECM3_Service1TaskPfm(4)

Note: This actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 1,2,3 respective 4.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

E.7.1 Initialization conditions

FUNCTION DESCRIPTION:

General information:

The application condition for the initialization of e.g. the error bits, the anti-bounce-counters and other variables of the process monitoring is commonly defined here for all modules of the process monitoring. There is only a reference to the following definition of the application conditions in the different modules.

The error bits, anti-bounce-counters and other variables are initialized
- at reset
- at transition from LV_IGK = 0 to 1
- at clearing of the failure memory

There is one exception from the rule above. If the reset counter at the monitoring unit increased up to 7 resets, the fault reaction of the monitoring unit will be continued until the power supply of the monitoring unit is switched off.

E.8 Chapter 1

E.8.1 Application incidences for the activation conditions of process monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_N_LIM_ETC_MON	V/O	0 ... 1 H	0 ... 1	1	-
Logical variable for activation of engine speed limitation monitoring					
LV_TQI_MON_ACT_MON	V/O	0 ... 1 H	0 ... 1	1	-
Logical variable for torque monitoring active					
LV_IGK_MON	O/V	0...1H	0...1	1	[-]
Logical variable for ignition key on					

Input data:

LV_N_LIM_REQ_MON	LV_IGK	LV_ERR_PVS_1	LV_ERR_PVS_2
LV_ERR_PVS_RATIO	LV_ERR_TPS	PV_AV_MON	LV_MTC_CUR_OFF
V_TPS_1_BAS	V_TPS_2_BAS		

FUNCTION DESCRIPTION:

General information:

The logical variable LV_N_LIM_ETC_MON indicates, that any etc-safety-function of level 1, 2 or 3 has requested engine speed limitation at 1500 rpm or ignition key is switched off. The monitoring of the engine speed limitation is also active, when

- any fault of the ETC-system (pedal value sensor, throttle position sensor, throttle actuator, power stage) is present and the driver demand is zero or
- throttle is disabled and at least one throttle position signal is outside the specified limp home position

The torque monitoring can only be active, when the ignition key is on and the throttle is not disabled. Otherwise the control of the torque is not possible and torque monitoring is forbidden. This is not a risk, because engine is switched off or in limp home mode then and the monitoring of the engine speed limitation is active in most cases (at least when the driver releases the pedal).

The logical variable LV_IGK is copied to level 2 in the variable LV_IGK_MON, because it is used as a deactivation condition for the monitoring of PVS in level 2. In order to ensure the safety, the variable LV_IGK_MON is also used for the calculation of the global activation conditions LV_N_LIM_ETC_MON and LV_TQI_MON_ACT_MON.

Application conditions:

Activation: at every engine state

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'

LV_N_LIM_ETC_MON = 0

LV_TQI_MON_ACT_MON = 0

LV_IGK_MON = 0

Update Rate: 40 ms

Formula section:

ACTION_ECM3_Service0TaskPfm(1)

LV_IGK_MON = LV_IGK

Activation of monitoring of engine speed limitation:

```
IF   LV_IGK_MON              =    0     OR    {during ignition-key OFF}
     LV_N_LIM_REQ_MON        =    1     OR    {request from level 2}

     [(LV_ERR_PVS_1          =    1     OR    {engine limp home mode}
     LV_ERR_PVS_2            =    1     OR
     LV_ERR_PVS_RATIO        =    1     OR
     LV_ERR_TPS              =    1     OR
     LV_MTC_CUR_OFF          =    1)    AND

     PV_AV_MON               =    0]    OR    {request for reversible SAS}

     (LV_MTC_CUR_OFF         =    1     AND
     (V_TPS_1_BAS >  C_V_TPS_1_LIH_MON  OR    {throttle not in limp home position}
     V_TPS_2_BAS <  C_V_TPS_2_LIH_MON))

THEN
     LV_N_LIM_ETC_MON        =    1
ELSE
     LV_N_LIM_ETC_MON        =    0
ENDIF
```

Activation of torque monitoring:

```
IF   LV_IGK_MON              =    1     AND   {ignition-key ON}
     LV_MTC_CUR_OFF          =    0           {MTC NOT switched off}
THEN
     LV_TQI_MON_ACT_MON      =    1
ELSE
     LV_TQI_MON_ACT_MON      =    0
ENDIF
```

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_TPS_1_LIH_MON	1	0 ... 3FF H	0 ... 4.995117	0.0049	V
Threshold for detection of throttle position 1 outside limp home position					
C_V_TPS_2_LIH_MON	1	0 ... 3FF H	0 ... 4.995117	0.0049	V
Threshold for detection of throttle position 2 outside limp home position					

E.8.2 Aplication incidences for monitoring of torque demand from idle speed controller

E.8.2.1 Application incidences for monitoring driver request passive

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_CT_MON	V	0 ... 1H	0 ... 1	1	-
Logical bit for detecion of driver request passive					

Input data:

LV_TQI_MON_ACT_MON	LV_CT		

FUNCTION DESCRIPTION:

The aim of this chapter is the monitoring of the variable LV_CT which indicates whether a ‘driver request’ is present or not where the request can be not only from the driver but also from other sources like cruise control or vehicle speed limitation functionality.

As long as the impact between LV_CT = 0 and LV_CT = 1 can be tolerated in terms of torque thresholds, the value can be copied to L2 without plausibility check.

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: for condition see ‘Application Incidences of Process Monitoring’
LV_CT_MON = 1

Update Rate: 40 ms

Formula section:

LV_CT_MON = LV_CT

E.8.2.2 Application incidences for monitoring of idle speed setpoint

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
N_DIF_SP_IS_MON	V/O	80 ... 7FH	-4096 ... 4064	32	rpm
Deviation of actual engine speed from idle speed setpoint					
N_SP_IS_RATIO_MON	V/O	0 ... FFFFH	0 ... 7.999	0.122E-3	-
Ratio between actual engine speed and idle speed setpoint					
N_SP_IS_MON	V/O	0 ... FFH	0 ... 8160	32	rpm
Monitored value of idle speed setpoint					
N_DIF_MON	V/O	80 ... 7FH	-4096 ... 4064	32	rpm
Engine speed deviation N_SP_IS - N					
N_SP_IS_CLC_MON	V	0 ... FFH	0 ... 8160	32	rpm
Modelled value of the corrected idle-speed setpoint					
FAC_RAMP_NEG_P_D_IS_MON	V/O	0 ... FA0H	0 ... 0.5	0.000125	-
Factor for monitoring negative PD deactivation-ramp operations					
N_SP_IS_BAS_MAX_MON	V	0...FFH	0...8160	32	[rpm]
Maximum basis idle speed setpoint depending on coolant temperature					
LV_N_INC_TCU_ACT_MON	V	0...1H	0...1	1	[-]
Engine speed increasing activation for transmission (monitoring)					
T_MAX_N_INC_ACT_MON	V	0...3FFFH	0...655.32	0.04	[s]
Time of engine speed increasing activation (monitoring)					
LV_T_MAX_TCU_ACT_MON	V	0...1H	0...1	1	[-]
Engine speed increasing activation plausibility					
N_SP_IS_EXT_REQ_MON	V	0...FFH	0...8160	32	[rpm]
Plausible basis idle-speed setpoint while DROF or external tester demand active					
CTR_N_SP_IS_EXT_REQ_DEC_MON	V	0...FFH	0...255	1	[-]
Counter of iteration to allow N_SP_IS_EXT_REQ_MON decrementation					
LV_N_INC_TCU_ACT_REQ_MON	V	0...1H	0...1	1	[-]
Engine speed increasing activation for transmission request					
N_INC_OLD_MON	V	0...FFH	0...8160	32	[rpm]
Engine speed when engine speed increasing request is set (monitoring)					
LV_N_INC_TCU_MAX_MON	V	0...1H	0...1	1	[-]
Engine speed increasing, engine speed increase not plausible					

Input data:

N_32_MON	N_SP_IS	TCO_MON	N_DIF_MMV
LV_AT	N_SP_IS_COR	LV_N_INC_TCU_REQ	LV_TQI_MON_ACT_MON
LV_N_INC_TCU_ACT	LV_CT_MON	LV_ERR_CAN_BUS_OFF	LV_ERR_TIMEOUT_TCU1

FUNCTION DESCRIPTION:

General information:

The objective of this function is the monitoring of the idle speed setpoint(N_SP_IS) and the corrected idle speed setpoint. Both of these values have to be monitored because they are important inputs for the the torque monitoring functionalities. This function consists of two sections.This section includes the monitoring of the basis idle-speed setpoint , the Level 2 model value for the idle-speed setpoint and other relevant variables that are important inputs for the torque monitoring.

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'

N_DIF_SP_IS_MON = 0 rpm
N_SP_IS_RATIO_MON = 0 rpm
N_SP_IS_MON = 0 rpm
N_DIF_MON = 0 rpm
FAC_RAMP_NEG_P_D_IS_MON = 0
CTR_N_SP_IS_EXT_REQ_DEC_ MON $_{K}$ = 0
CTR_N_SP_IS_EXT_REQ_DEC_ MON $_{K-1}$ = 0
N_SP_IS_EXT_REQ_MON $_{K}$ = **min(**N_SP_IS, IP_N_SP_IS_BAS_MAX_MON**)**
N_SP_IS_EXT_REQ_MON $_{K-1}$= **min(**N_SP_IS, IP_N_SP_IS_BAS_MAX_MON**)**
LV_T_MAX_TCU_ACT_MON = 0
LV_N_INC_TCU_ACT_MON = 0
LV_N_INC_TCU_ACT_REQ_MON = 0
T_MAX_N_INC_ACT_MON $_{k}$ = 0
T_MAX_N_INC_ACT_MON $_{k-1}$ = 0
N_SP_IS_BAS_MAX_MON = **min(**N_SP_IS, IP_N_SP_IS_BAS_MAX_MON**)**
N_INC_OLD_MON = 0 rpm

Update Rate: 40 ms

E.8.2.2.1 Plausiblisation of external influences on idle-speed setpoint

Formula section:

FAC_RAMP_NEG_P_D_IS_MON = C_FAC_RAMP_P_D_MON

E.8.2.2.2 Calculation of idle-speed setpoint model value

General information:

The basis idle speed setpoint is limited by a coolant temperature dependent maximum value, IP_N_SP_IS_BAS_MAX_MON. The actual idle speed setpoint,N_SP_IS_MON, is under normal conditions equal to the basis setpoint, N_SP_IS_LIM. Depending on a LC bit, the deviation between engine speed and the idle speed setpoint is calculated in two paths:

- **LC_N_SP_IS_CLC_CALC_MON = 0** – N_SP_IS_MON and the N_DIF-moving mean value (N_DIF_MMV_MON) are relevant
- **LC_N_SP_IS_CLC_CALC_MON = 1** – Only the corrected idle speed setpoint (N_SP_IS_CLC_MON) is relevant

The corrected idle speed setpoint (N_SP_IS_COR - which is directly calculated from N_SP_IS) is limited to a maximum and minimum value i.e. engine speed (N_32_MON) and idle speed setpoint (N_SP_IS_MON) respectively. The outputs of the function are:

- ratio between idle speed setpoint and actual engine speed - **N_SP_IS_RATIO_MON**
- deviation of actual engine speed from idle speed setpoint - **N_DIF_SP_IS_MON**
- deviation of actual engine speed from N_SP_IS - **N_DIF_MON**

Formula section:
{L2 model value of idle-speed setpoint}

N_SP_IS_BAS_MAX_MON = IP_N_SP_IS_BAS_MAX_MON

IF LV_AT = 1 **AND**
LV_N_INC_TCU_REQ = 1 **AND**
LV_N_INC_TCU_ACT = 1 **AND**
LV_CT_MON = 1 **AND**
LV_ERR_CAN_BUS_OFF = 0 **AND**
LV_ERR_TIMEOUT_TCU1 = 0

THEN LV_N_INC_TCU_ACT_REQ_MON = 1
ELSE LV_N_INC_TCU_ACT_REQ_MON = 0
ENDIF

IF LV_N_INC_TCU_ACT_REQ_MON = 0 -> 1
THEN N_INC_OLD_MON = N_32_MON
ENDIF

IF LV_N_INC_TCU_ACT_REQ_MON = 1
THEN **IF** $T\_MAX\_N\_INC\_ACT\_MON_{k-1}$ > C_T_MAX_N_INC_NOT_PLAUS_MON
THEN LV_T_MAX_TCU_ACT_MON = 1
ELSE
$T\_MAX\_N\_INC\_ACT\_MON_{k} = T\_MAX\_N\_INC\_ACT\_MON_{k-1} + 40ms$
ENDIF
ELSE LV_T_MAX_TCU_ACT_MON = 0

```
        T_MAX_N_INC_ACT_MON k = 0
ENDIF

IF      N_SP_IS_TCU - N_INC_OLD_MON > C_N_INC_TCU_REL_MAX_MON
THEN    LV_N_INC_TCU_MAX_MON = 1
ELSE    LV_N_INC_TCU_MAX_MON = 0
ENDIF

IF      LV_N_INC_TCU_ACT_REQ_MON = 1 AND
        LV_T_MAX_TCU_ACT_MON = 0     AND
        LV_N_INC_TCU_MAX_MON = 0
THEN    LV_N_INC_TCU_ACT_MON = 1
ELSE    LV_N_INC_TCU_ACT_MON = 0
        N_INC_OLD_MON = 0
ENDIF

IF      LV_N_INC_TCU_ACT_MON = 1
THEN    N_SP_IS_EXT_REQ_MON K = min(N_SP_IS_TCU,
                                          C_N_INC_TCU_REQ_MAX_MON)
ELSE    N_SP_IS_EXT_REQ_MON K = min(N_SP_IS, N_SP_IS_BAS_MAX_MON)
ENDIF

{negative gradient limitation}
IF(1)       N_SP_IS_EXT_REQ_MON K < N_SP_IS_EXT_REQ_MON K-1

THEN(1)
    IF(2)          CTR_N_SP_IS_EXT_REQ_DEC_MON K-1 >=
                                      C_CTR_N_SP_IS_REQ_DEC_MAX_MON
    THEN(2)
                   CTR_N_SP_IS_EXT_REQ_DEC_MON K = 0
                   N_SP_IS_EXT_REQ_MON K =
                   min(N_SP_IS, N_SP_IS_EXT_ REQ _MON K-1 -
                                   C_N_SP_LGRD_IS_EXT_REQ_DEC_MON)
    ELSE(2)

                   CTR_N_SP_IS_EXT_REQ_DEC_MON K =
                                  CTR_N_SP_IS_EXT_REQ_DEC_MON K-1 + 1
                   N_SP_IS_EXT_REQ_MON K = N_SP_IS_EXT_REQ_MON K-1

    END IF(2)

ELSE(1)
            CTR_N_SP_IS_EXT_REQ_DEC_MON K = 0
END IF(1)
```

N_SP_IS_MON = **max(**N_SP_IS_EXT_REQ_MON $_K$,
min(N_SP_IS, N_SP_IS_BAS_MAX_MON**))**

N_SP_IS_RATIO_MON = N_32_MON / N_SP_IS_MON **{Idle-speed setpoint ratio}**

{Deviation between engine speed idle-speed setpoint}
N_DIF_MON = N_SP_IS_MON – N_32_MON

{Calculation of corrected setpoint – N_SP_IS_CLC_MON}
IF LC_N_SP_IS_CLC_CALC_MON == 1
THEN **{Plausibility check for corrected idle speed setpoint}**
N_SP_IS_CLC_1_MON = **max(**N_SP_IS_COR, N_SP_IS_MON**)**
N_SP_IS_CLC_MON = **min(**N_SP_IS_CLC_1_MON, N_SP_IS_MON + C_N_SP_IS_ADD_LPF_THD_MON**)**

{Calculation of engine speed deviation from idle setpoint}
N_DIF_CLC_MON = N_SP_IS_CLC_MON - N_32_MON
ELSE **{Plausibility check for mean moving value}**
N_DIF_MMV_MON = **min(**0, **max**(N_DIF_MMV, C_N_DIF_MIN_MMV_MON)**)**

{Calculation of engine speed deviation from idle setpoint}
N_DIF_CLC_MON = N_DIF_MON – (N_DIF_MMV_MON*C_N_DIF_FAC_MON)
ENDIF

{Deviation between engine speed and corrected idle-speed setpoint – input for ISC}
N_DIF_SP_IS_MON = N_DIF_CLC_MON

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_N_DIF_FAC_MON	1	0 ... FFH	0 ... 0.996	0.0039	-
Correlation factor for calculating N_DIFF_MMV model value					
C_N_DIF_MIN_MMV_MON	1	80 ... 0H	-4096 ... 0	32	rpm
Lower threshold value of N_DIF_MMV_MON					
LC_N_SP_IS_CLC_CALC_MON	1	0 ... 1 H	0 ... 1	1	-
Logical bit for switching between two engine-speed-deviation calculation paths(N_DIF_CLC_x_MON)					
C_N_SP_IS_ADD_LPF_THD_MON	1	0 ... FF H	0 ... 8160	32	rpm
Threshold for activation of N_SP_IS_CLC filtering					
C_FAC_RAMP_P_D_MON	1	0 ... FA0H	0 ... 0.5	0.000125	-
Factor for monitoring negative PD deactivation-ramp operations					
IP_N_SP_IS_BAS_MAX_MON	5	0 ... FFH	0 ... 8160	32	rpm
LDP_TCO_MON_IP_N_SP_IS_MON	5	0 ... FEH	-48 ... 142.5	0.75	°C
maximum basis idle speed setpoint depending on coolant temperature					

C_N_INC_TCU_REQ_MAX_MON	1	0...FFH	0...8160	32	[rpm]
Maximum engine speed rise request from TCU					
C_T_MAX_N_INC_NOT_PLAUS_MON	1	0...3FFFH	0...655.32	0.04	[s]
Maximum time of engine speed increasing activation					
C_CTR_N_SP_IS_REQ_DEC_MAX_MON	1	0...FFH	0...255	1	[-]
Maximum for counter of iteration to allow N_SP_IS_EXT_REQ_MON decrementation					
C_N_SP_LGRD_IS_EXT_REQ_DEC_MON	1	0...FFH	0...8160	32	[rpm]
Decrement for idle speed SP for external request (Monitoring)					
C_N_INC_TCU_REL_MAX_MON	1	0...FFH	0...8160	32	[rpm]
Maximum deviation of engine speed between request value from TCU and saved value at activation					

E.8.2.3 Monitoring of Idle-speed controller activation conditions

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
TQ_DIF_P_D_IS_MON	V/O	8000 … 7FFF H	-1024 ... 1023.97	0.03125	Nm
Monitored PD-part of the idle speed controller					
TQ_DIF_P_D_IS_MAX_MON	V/O	0 … 7FFF H	0 ... 1023.97	0.03125	Nm
Maximum value of PD-part					
TQ_DIF_P_D_IS_MAX_2_MON	V	0 … 7FFF H	0 ... 1023.97	0.03125	Nm
Temporary variable					
STATE_PAS_RAMP_ACT_P_D_IS_MON	V/O	0 .. FFH	0 ... 255	1	-
Circular buffer indicating statuts of PD-part deactivation ramp					
STATE_PAS_RAMP_ACT_I_IS_MON	V/O	0 .. FFH	0 ... 255	1	-
Circular buffer indicating statuts of I-part deactivation ramp					
LV_PAS_RAMP_ACT_P_D_CHG_MON	V/O	0 ... 1 H	0 ... 1	1	-
Activation bit for initialisation of PD-ramp					
LV_PAS_RAMP_ACT_I_CHG_MON	V/O	0 ... 1 H	0 ... 1	1	-
Activation bit for initialisation of I-ramp					
STATE_REQ_ISC_MON	V/O	0 H 1 H 2 H 3 H	NOT_ACTIVE IDLE TRAILING_THR PART_LOAD	-	-
State of idle speed controller monitoring					

Input data:

LV_CT_MON	N_DIF_MON	LV_REQ_ISC	LV_TQI_MON_ACT_MON
T_CTR_PU_ISC_ACT	TQ_DIF_P_D_SLOW_IS	N_32_MON	TQ_DIF_P_D_FAST_IS
STATE_PAS_RAMP_ACT_P_D_IS	STATE_PAS_RAMP_ACT_I_IS		

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: (for condition see 'Application Incidences of Process Monitoring'
OR LV_TQI_MON_ACT_MON = 0 $\rightarrow$ 1)

STATE_REQ_ISC_MON = 'NOT_ACTIVE'
TQ_DIF_P_D_IS_MAX_2_MON = **max**(TQ_DIF_P_D_SLOW_IS$_k$, TQ_DIF_P_D_FAST_IS$_k$)
TQ_DIF_P_D_IS_MON = 0 Nm
TQ_DIF_P_D_IS_MAX_MON = **max**(TQ_DIF_P_D_SLOW_IS$_k$, TQ_DIF_P_D_FAST_IS$_k$)
STATE_PAS_RAMP_ACT_P_D_IS_MON = 0
STATE_PAS_RAMP_ACT_I_IS_MON = 0
LV_PAS_RAMP_ACT_P_D_CHG_MON = 0
LV_PAS_RAMP_ACT_I_CHG_MON = 0

Update Rate: 40 ms

General information:

The ISC can be active either when the engine is in idle state(LV_REQ_ISC=1) or when the DROF functionality is active or under certain conditions in trailing-throttle state (LV_PU=1) and part-load state (LV_PL=1) and. So all of these possible "ISC-active" phases have to be monitored. The activation conditions are based on the L1 ISC activation conditions as described in the module 'Application incidences for the idle speed controller'. In addition to the normal activation states, a 'NOT_ACTIVE' state is generated in order to be able to monitor the ISC if the activation information from the function level is faulty.

Formula section:

{The L1 ISC-Torque demand signal is copied to the process monitoring (L2) RAM}
TQ_DIF_P_D_IS_MON$_k$ = **max**(TQ_DIF_P_D_SLOW_IS$_k$, TQ_DIF_P_D_FAST_IS$_k$)

{L2 threshold of 'normal' PD-part torque request}
TQ_DIF_P_D_IS_MAX_1_MON = IP_TQ_DIF_P_D_IS_MAX_MON(N_DIF_SP_IS_MON$_k$)

{Calculation of normal monitoring threshold for PD-part and corresponding filter parameters. Filtering only in falling direction.}
IF TQ_DIF_P_D_IS_MAX_1_MON$_k$ < TQ_DIF_P_D_IS_MAX_2_MON$_{k-1}$
THEN TQ_DIF_P_D_IS_MAX_2_MON$_k$ =
[C_FIL_TQ_P_D_IS_MAX_MON * TQ_DIF_P_D_IS_MAX_1_MON$_k$
+ (1- C_FIL_TQ_P_D_IS_MAX_MON) * TQ_DIF_P_D_IS_MAX_2_MON$_{k-1}$**]**
ELSE TQ_DIF_P_D_IS_MAX_2_MON$_k$ = TQ_DIF_P_D_IS_MAX_1_MON$_k$
ENDIF

$TQ\_DIF\_P\_D\_IS\_MAX\_MON_k = TQ\_DIF\_P\_D\_IS\_MAX\_2\_MON_k$
$* IP\_FAC\_TQ\_DIF\_P\_D\_IS\_MON(TCO\_MON)$

{Evaluation of ISC-deactivation-ramp status in L1 to check for activation during current recurrency or between two L2 recurrencies. To simplify the usage of individual Bit positions in the formula section, the following representation will be used:
STATE_PAS_RAMP_ACT_P_D_IS_MON(i) = STATE_PAS_RAMP_ACT_P_D_IS_MON : Bit i where i=0 ... 4
STATE_PAS_RAMP_ACT_I_IS_MON (j) = STATE_PAS_RAMP_ACT_I_IS_MON : Bit j → where j=0 ... 2 **}**

STATE_PAS_RAMP_ACT_P_D_IS_MON = STATE_PAS_RAMP_ACT_P_D_IS
IF ((STATE_PAS_RAMP_ACT_P_D_IS_MON (0) == 1
AND STATE_PAS_RAMP_ACT_P_D_IS_MON (4) == 0)
OR [(STATE_PAS_RAMP_ACT_P_D_IS_MON (0) == 1
AND STATE_PAS_RAMP_ACT_P_D_IS_MON (4) == 1)
AND (STATE_PAS_RAMP_ACT_P_D_IS_MON (1) == 0
OR STATE_PAS_RAMP_ACT_P_D_IS_MON (2) == 0
OR STATE_PAS_RAMP_ACT_P_D_IS_MON (3) == 0) **]**
OR (LV_TQI_MON_ACT_MON = 0 → 1 **AND** STATE_PAS_RAMP_ACT_P_D_IS_MON (0) == 1) **)**
THEN LV_PAS_RAMP_ACT_P_D_CHG_MON = 1
ELSE LV_PAS_RAMP_ACT_P_D_CHG_MON = 0
ENDIF

STATE_PAS_RAMP_ACT_I_IS_MON = STATE_PAS_RAMP_ACT_I_IS
IF (STATE_PAS_RAMP_ACT_I_IS_MON (0)==1
AND STATE_PAS_RAMP_ACT_I_IS_MON(2)==0)
OR [STATE_PAS_RAMP_ACT_I _IS_MON (0) ==1
AND STATE_PAS_RAMP_ACT_I _IS_MON (2) == 1
AND STATE_PAS_RAMP_ACT_I _IS_MON (1) == 0 **]**
OR (LV_TQI_MON_ACT_MON = 0 → 1 **AND** STATE_PAS_RAMP_ACT_I_IS_MON (0)==1)
THEN LV_PAS_RAMP_ACT_I_CHG_MON = 1
ELSE LV_PAS_RAMP_ACT_I_CHG_MON = 0
ENDIF

{Plausiblisation of ISC activation conditions}

IF [(N_DIF_MON ≥ C_N_DIF_MAX_IS_MON) **AND** (LV_CT_MON == 1)
AND (LV_REQ_ISC == 1)**]**
THEN STATE_REQ_ISC_MON = ‘IDLE’
ELSE IF[(N_DIF_MON < C_N_DIF_MAX_IS_MON - C_N_DIF_HYS_IS_PU_MON)
AND (LV_CT_MON = 1) **AND** (T_CTR_PU_ISC_ACT == 0) **]**
THEN STATE_REQ_ISC_MON = ‘TRAILING_THR’
ELSE IF [(N_SP_IS_MON < C_N_SP_IS_MAX_PL_ACT_MON) **AND** (LV_CT_MON==0)
AND (N_DIF_MON > C_N_DIF_MAX_PL_ACT_MON)**]**

THEN STATE_REQ_ISC_MON = ‘PART_LOAD’
ELSE STATE_REQ_ISC_MON = ‘NOT_ACTIVE’
ENDIF
ENDIF
ENDIF

ACTION_ECM3_Service1TaskPfm(1)

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_N_DIF_MAX_IS_MON	1	80 ... 7F H	-4096 ... 4064	32	rpm
Maximum idle speed setpoint deviation for active PD-part					
C_N_DIF_HYS_IS_PU_MON	1	80 ... 7F H	-4096 ... 4064	32	rpm
N_DIF_MON hysteresis before state PU					
C_N_DIF_MAX_PL_ACT_MON	1	80 ... 7F H	-4096 ... 4064	32	Rpm
Maximum idle-speed deviation allowed for active ISC in PL					
C_N_SP_IS_MAX_PL_ACT_MON	1	0 ... FFH	0 ... 8160	32	rpm
Maximum allowed idle-speed for active ISC in PL					
C_FIL_TQ_P_D_IS_MAX_MON	1	0 ... FFFF H	0 ... 0.999984	1.528e-5	-
Filter correlation constant for PD-part trorque threshold					
IP_TQ_DIF_P_D_IS_MAX_MON	6	0 ... FF H	0 ... 510	2	Nm
LDP_N_DIF_SP_IS_MON_IP_TQ_MON	6	0 ... FF H	-4096 ... 4064	32	rpm
maximum PD-part depending on speed deviation to idle speed setpoint					
IP_FAC_TQ_DIF_P_D_IS_MON	5	0 ... FF H	0 ... 1.99216	7.84e-3	-
LDP_TCO_MON_IP_FAC_TQ_P_D_MON	5	0 ... FE H	-48 ... 142.5	0.75	°C
maximum scaling factor due to separation of PD-part					

E.8.2.3.1 Circular buffer for ramp activation for ISC level 1

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
STATE_PAS_RAMP_ACT_P_D_IS	V/O	0...FFH	0...255	1	-
Circular buffer for the logical value for PD-part passive ramp active					
STATE_PAS_RAMP_ACT_I_IS	V/O	0...FFH	0...255	1	-
Circular buffer for the logical value for I-part passive ramp active					

Input data:

LV_PAS_RAMP_ACT_P_D_IS	LV_PAS_RAMP_ACT_I_IS	LV_TQI_MON_ACT_MON	

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: (for condition see 'Application Incidences of Process Monitoring'
OR LV_TQI_MON_ACT_MON = 0 → 1)

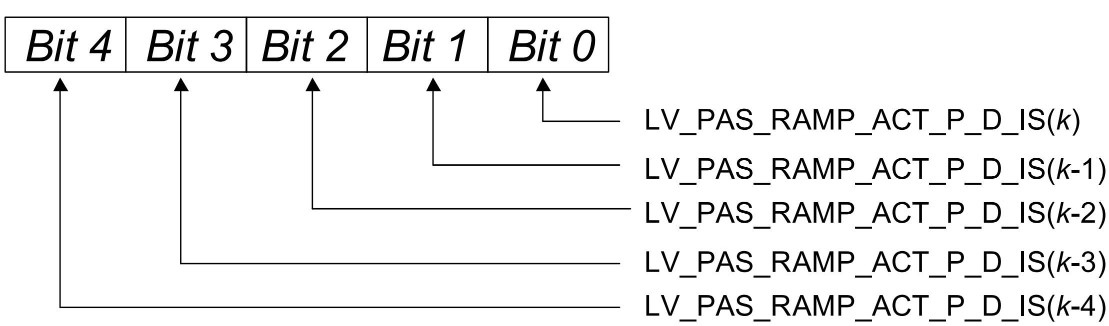
STATE_PAS_RAMP_ACT_P_D_IS = 0
STATE_PAS_RAMP_ACT_I_IS = 0

Update Rate: 10 ms

ACTION_ECM3_Service0TaskPfm(0)

Circular buffer for LV_PAS_RAMP_ACT_P_D_IS

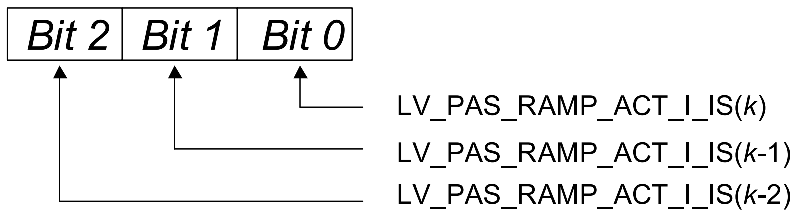
In the following circular buffer STATE_PAS_RAMP_ACT_P_D_IS, the last five values of LV_PAS_RAMP_ACT_P_D_IS are stored:



This information is needed for the monitoring of the idle speed controller. It is updated every 10ms; the values are shifted from Bit j to Bit j+1, j=0,1,2,3.

Circular buffer for LV_PAS_RAMP_ACT_I_IS

In the following circular buffer STATE_PAS_RAMP_ACT_I_IS, the last three values of LV_PAS_RAMP_ACT_I_IS are stored:



This information is needed for the monitoring of the idle speed controller. It is updated every 20ms; the values are shifted from Bit j to Bit j+1, j=0,1.

ACTION_ECM3_Service1TaskPfm(0)

E.9 Chapter 2

E.9.1 Application incidences for monitoring of pedal value signals

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_BRAKE_MON	O/V	0 ... 1 H	0 ... 1	1	-
Brake active or brake fault present					
LV_ERR_VCC_PVS_1_MON	O/V	0 ... 1 H	0 ... 1	1	-
Fault in power supply for PVS_1 present					
LV_ERR_VCC_PVS_2_MON	O/V	0 ... 1 H	0 ... 1	1	-
Fault in power supply for PVS_2 present					
LV_ERR_VCC_PVS_MON	O/V	0 ... 1 H	0 ... 1	1	-
Fault of any PVS power supply detected by level 2 diagnosis					

Input data:

LV_BLS	LV_BTS	LV_ERR_BLS_BTS	

FUNCTION DESCRIPTION:

General information:

In order to be able to use the generic module for the monitoring of the pedal value signals in level 2, different informations about the state of the brake are combined to one resulting information LV_BRAKE_MON. LV_BRAKE_MON is set, when at least one sensor (brake light switch or brake test switch) indicates an activated brake or a fault of a brake switch has been detected.

The state of the brake light switch (BLS) is checked via a direct read command on the port register. The information about the brake test switch (BTS) is evaluated, using a variable from the basic software, which is also used by the functions of level 1. A separate read command on the port register is not possible here, because the signal is measured via a multiplexer.

The usage of the error flags for the power supply voltages (LV_ERR_VCC_PVS_x_MON) of the pedal value sensor depends on the configuration of the power supply and the availability of diagnosis information in level 1. It is assumed, that faults detected in the power supply are handled in the same way as signal range faults of the corresponding pedal value sensor channels.

A separate power supply voltage diagnosis is necessary in level 2 in case of no redundant power supply is used for PVS supply and no PVS supply diagnosis is integrated in the PVS sensor.

Application conditions:

Activation: at every engine state

Deactivation: otherwise

Initialisation: for condition see ‘Application Incidences of Process Monitoring’

LV_BRAKE_MON = 0

LV_ERR_VCC_PVS_1_MON = 0

LV_ERR_VCC_PVS_2_MON = 0

LV_ERR_VCC_PVS_MON = 0

Update Rate: 40 ms

Formula section:

ACTION_ECM3_Service0TaskPfm(2)

```
IF
    [(LV_BLS = 1)            OR        {brake light switch}
    (LV_BTS = 1)             OR        {brake test switch}
    (LV_ERR_BLS_BTS = 1)]              {error of BLS/BTS}
THEN
    LV_BRAKE_MON = 1
ELSE
    LV_BRAKE_MON = 0
ENDIF
```

LV_ERR_VCC_PVS_1_MON = LV_ERR_VCC_PVS_1

LV_ERR_VCC_PVS_2_MON = LV_ERR_VCC_PVS_2

LV_ERR_VCC_PVS_MON = 0

E.9.2 Application incidences for the monitoring of torque losses

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQ_LOSS_ADD_MON	O/V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Additional torque losses					
TQ_LOSS_ACC_MON	O/V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Air conditioning compressor torque losses					
TQ_LOSS_ECF_MON	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Electric cooling fan torque losses					
TQ_LOSS_ALTER_MON	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Alternator torque losses					
TQ_LOSS_PSTE_MON	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Power steering torque losses					
TQ_LOSS_AMP_MON	V	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Altitude correction torque losses					
INT_TQ_LOSS_ACC_MON	V	0...FFFFH	0...204.79687	0.003125	[Nm*s]
Integrated air conditioning compressor torque losses					
LV_INT_TQ_LOSS_ACC_MAX_MON	V	0...1H	0...1	1	[-]
Inhibition of air conditioning compressor torque losses due to integral check					

Input data:

TQ_LOSS_ACC	TQ_LOSS_ECF	TQ_LOSS_ALTER	TQ_LOSS_PSTE
TQ_LOSS_AMP	TQ_LOSS_PSTE_PSPT		

FUNCTION DESCRIPTION:

In order to be able to use the generic module for the monitoring of torque losses in level 2 the not engine immanent additional torque losses are calculated here.

The level 1 value of each single additional torque loss will be compared with a maximum toleratable value. The larger value of the two is added to the sum TQ_LOSS_ADD_MON.

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'

TQ_LOSS_ADD_MON = 0 Nm

$INT\_TQ\_LOSS\_ACC\_MON_{k} = INT\_TQ\_LOSS\_ACC\_MON_{k-1} = 0$ Nm*s

LV_INT_TQ_LOSS_ACC_MAX_MON = 0

Update Rate: 40 ms

Formula section:

E.9.2.1 Calculation of air conditioning compressor torque losses

If TQ_LOSS_ACC < C_TQ_LOSS_ACC_MAX_MON

Then INT_TQ_LOSS_ACC_MON = min**{**C_INT_MAX_TQ_LOSS_ACC_MON;
max(0 ; INT_TQ_LOSS_ACC_MON $_{k-1}$ +
(abs(TQ_LOSS_ACC - C_TQ_LOSS_ACC_MAX_MON) *
C_CRLC_INT_TQ_LOSS_ACC_MON * 0,04s))**}**

Else INT_TQ_LOSS_ACC_MON = min**{** C_INT_MAX_TQ_LOSS_ACC_MON;
max(0 ; INT_TQ_LOSS_ACC_MON $_{k-1}$ -
C_DEC_INT_TQ_LOSS_ACC_MON)**}**

Endif

If INT_TQ_LOSS_ACC_MON >= C_INT_MAX_TQ_LOSS_ACC_MON

Then LV_INT_TQ_LOSS_ACC_MAX_MON = 1
TQ_LOSS_ACC_MON = max(TQ_LOSS_ACC; C_TQ_LOSS_ACC_MAX_MON)

Else LV_INT_TQ_LOSS_ACC_MAX_MON = 0
TQ_LOSS_ACC_MON = max(TQ_LOSS_ACC;
C_TQ_LOSS_ACC_MAX_MAX_MON)

Endif

E.9.2.2 Calculation of other torque losses

TQ_LOSS_ECF_MON = max (TQ_LOSS_ECF, C_TQ_LOSS_ECF_MAX_MON)

TQ_LOSS_ALTER_MON =max (TQ_LOSS_ALTER, C_TQ_LOSS_ALTER_MAX_MON)

TQ_LOSS_AMP_MON = max (TQ_LOSS_AMP, C_TQ_LOSS_AMP_MAX_MON)

If CONF_PSTE = 1

Then TQ_LOSS_PSTE_MON = max (TQ_LOSS_PSTE_PSPT,
C_TQ_LOSS_PSTE_PSPT_MAX_MON)

Else TQ_LOSS_PSTE_MON = max (TQ_LOSS_PSTE,
C_TQ_LOSS_PSTE_MAX_MON)

Endif

E.9.2.3 Calculation of additional torque losses

Formula section:

TQ_LOSS_ADD_MON = TQ_LOSS_ACC_MON + TQ_LOSS_ECF_MON +
TQ_LOSS_ALTER_MON + TQ_LOSS_PSTE_MON +
TQ_LOSS_AMP_MON

ACTION_ECM3_Service1TaskPfm(2)

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_TQ_LOSS_ACC_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of ACC torque loss					
C_TQ_LOSS_ECF_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of Electric cooling fan torque loss					
C_TQ_LOSS_ALTER_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of Alternator torque loss					
C_TQ_LOSS_PSTE_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of power steering torque loss (pressure switch, conf_pste=0)					
C_TQ_LOSS_PSTE_PSPT_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of power steering torque loss (pressure transducer, conf_pste=1)					
C_TQ_LOSS_AMP_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of altitude correction torque loss					
C_INT_MAX_TQ_LOSS_ACC_MON	1	0...FFFFH	0...204.79687	0.003125	[Nm*s]
Maximum value of intergrated air conditioning compressor torque losses					
C_DEC_INT_TQ_LOSS_ACC_MON	1	0...FFFFH	0...204.79687	0.003125	[Nm*s]
Decrement of intergrated air conditioning compressor torque losses					
C_CRLC_INT_TQ_LOSS_ACC_MON	1	0...FFH	0...0.99609	3.9063e-3	[-]
Correlation factor for calculation of intergrated air conditioning compressor torque losses					
C_TQ_LOSS_ACC_MAX_MAX_MON	1	8000...7FFFH	-1024... 1023.96875	0.03125	[Nm]
Maximum of ACC torque loss with integral check					

E.10 Chapter 3

E.10.1 Application incidences for monitoring of minimum torque at clutch

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
STATE_ERR_DET_TQ_MIN_MON	V/O	0 H 1 H 2 H	PU NON PU PASSIVE	-	-
Error-detection-state for monitoring of minimum torque at clutch					
TQ_ADD_MIN_CLU_MON	V/O	8000H...7FFFH	-1024...1023.97	0.03125	Nm
Additional torque request from external sources					
TQ_CONV_MAX_MDL_MON	V	0 ... FFH	0 ... 510	2	Nm
Maximum converter torque losses					
TQ_CONV_MAX_MON	V/O	0 ... FFH	0 ... 510	2	Nm
Monitored vaue of converter torque losses					
LV_SWI_TQ_MIN_CLU_MON	V/O	0 ... 1H	0 ... 1	1	-
Idle speed setpoint from external influences					

Input data:

N_DIF_MON	LV_CT_MON	LV_PUC	C_N_DIF_HYS_IS_PU_MON
EFF_SCC_MON	C_N_DIF_MAX_IS_MON	STATE_REQ_ISC_MON	TQ_CONV

FUNCTION DESCRIPTION:

General information:

The objective of this function is the determination of the error-detection paths of the minimum torque at clutch. The error-detection is divided into PU state and non-PU state. This separation is done because the biggest influence on the value of the minimum torque that is available at the clutch is the self-stabilising factor FAC_N_SP_IS_RATIO_MON and this has a parabolic trajectory where the transition point from the positive values to TQ_LOSS_MON is around the idle-speed setpoint(N_SP_IS_MON). STATE_ERR_DET_TQ_MIN_MON is an input into the module 'Monitoring of minimum torque at clutch'.

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'

STATE_ERR_DET_TQ_MIN_MON = 'PASSIVE'

TQ_ADD_MIN_CLU_MON = 0 Nm

TQ_CONV_MAX_MON = 0 Nm

LV_SWI_TQ_MIN_CLU_MON = 0

Update Rate: 40 ms

Formula section:

ACTION_ECM3_Service0TaskPfm(3)

IF (N_DIF_MON < C_N_DIF_MAX_IS_MON **-** C_N_DIF_HYS_IS_PU_MON)
AND (NOT(LC_INH_PU_REQ_MON) **OR** (LV_PUC==1 **AND** EFF_SCC_MON<1) **)**
AND (STATE_REQ_ISC_MON ≠ 'IDLE')
AND (LV_CT_MON == 1)
THEN LV_SWI_TQ_MIN_CLU_MON = 1
STATE_ERR_DET_TQ_MIN_MON = 'PU'
ELSE **IF** (N_DIF_MON ≥ C_N_DIF_MAX_IS_MON)
OR (LV_CT_MON == 0)
OR **[**(N_DIF_MON < C_N_DIF_MAX_IS_MON **-** C_N_DIF_HYS_IS_PU_MON)
AND LC_INH_PU_REQ_MON **AND** (LV_CT_MON == 1)**]**
THEN LV_SWI_TQ_MIN_CLU_MON = 0
STATE_ERR_DET_TQ_MIN_MON = 'NON PU'
ELSE STATE_ERR_DET_TQ_MIN_MON = 'PASSIVE'
ENDIF
ENDIF

{Calculation of Maximum Converter Torque. If a CVT system is used, then the torque demand/engine load resulting from the transmission has to be considered here. If this value is transmitted from the TCU via CAN, then it has to be first plausiblised in the module 'Monitoring of CAN-signals' and only then used here. If a normal automatic transmission is used, then the torque losses resulting from the converter can be calculated as given below. }

IF LV_AT == 1
THEN TQ_CONV_MAX_MDL_MON = (C_FAC_CONV_MON * N_SP_IS_MON * N_SP_IS_MON) + C_TQ_DIF_IS_AD_CONV_MAX_MON
TQ_CONV_MAX_MON = max (TQ_CONV, TQ_CONV_MAX_MDL_MON)
ELSE TQ_CONV_MAX_MON = 0 Nm
TQ_CONV_MAX_MDL_MON = 0 Nm
ENDIF

{If torque demands from external sources like AMT, CVT etc. are present, these have to be considered into the additional torque request term **TQ_ADD_MIN_CLU_MON.** In this case a sufficient monitoring of this requests must be performed.}
TQ_ADD_MIN_CLU_MON = 0 Nm

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_FAC_CONV_MON	1	0 ... FFFFH	0 ... 0.000488	7.446e-9	Nm*rpm^{-2}
Converter torque characteristics					
C_TQ_DIF_IS_AD_CONV_MAX_MON	1	0 ... FFH	0 ... 510	2	Nm

LC_INH_PU_REQ_MON	1	0 , 1H	0, 1	1	-
Logical bit to inhibit LV_PU in calculation of TQ_MIN_CLU_MDL_MON					

E.10.2 Application incidences for the actual efficiencies

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
IGA_REF_COR_EXT_MON	V/O	0 ... FF H	-35.625 ... 60	0.375	°CRK
Correction of reference ignition angle (for external functions e.g. VVT, EGR...)					
IGA_REF_TEMP_COR_MON	V	80 ... 7F H	-48 ... 47.625	0.375	°CRK
Correction of reference ignition angle (additive temperature correction)					
IGA_REF_VVT_COR_MON	V	80 ... 7F H	-48 ... 47.625	0.375	°CRK
Correction of reference ignition angle (external correction)					
IGA_REF_PORT_COR_MON	V	80 ... 7F H	-48 ... 47.625	0.375	°CRK
Correction of reference ignition angle (external correction)					

Input data:

IGA_REF_VVT_COR	IGA_REF_TEMP_COR	IGA_REF_PORT_COR	

FUNCTION DESCRIPTION:

General information:

The calculation of IGA_REF_COR_MON is a additive combination of limited values of level 1. The limitation of each single value allows to have a close tolerance range for monitoring.

Application conditions:

Activation: at every engine state

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'

IGA_REF_COR_EXT_MON = 0 °CRK

IGA_REF_TEMP_COR_MON = 0 °CRK

IGA_REF_VVT_COR_MON = 0 °CRK

IGA_REF_PORT_COR_MON = 0 °CRK

Update Rate: 40 ms

Formula section:

IGA_REF_TEMP_COR_MON = min(IGA_REF_TEMP_COR,
C_IGA_REF_TEMP_COR_LIM_MON)

IGA_REF_VVT_COR_MON = min(IGA_REF_VVT_COR,
C_IGA_REF_VVT_COR_LIM_MON)

IGA_REF_PORT_COR_MON = min(IGA_REF_PORT_COR,
C_IGA_REF_PORT_COR_LIM_MON)

IGA_REF_COR_EXT_MON = IGA_REF_TEMP_COR_MON +
IGA_REF_VVT_COR_MON +
IGA_REF_PORT_COR_MON

ACTION_ECM3_Service1TaskPfm(3)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_IGA_REF_TEMP_COR_LIM_MON	1	80 ... 7F H	-48 ... 47.625	0.375	°CRK
maximum permitted ignition angle correction for additive temperature correction					
C_IGA_REF_VVT_COR_LIM_MON	1	80 ... 7F H	-48 ... 47.625	0.375	°CRK
maximum permitted ignition angle correction for VVT					
C_IGA_REF_PORT_COR_LIM_MON	1	80 ... 7F H	-48 ... 47.625	0.375	°CRK
maximum permitted ignition angle correction for port flap					

E.11 Chapter 4

E.11.1 Application incidences for desired indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
PV_AV_AD_MON	V/O	0...FF H	0...99.6	0.39	%
Modified pedal value signal					
TQI_MIN_PU_MON	V/O	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm
Plausiblised L2 value of Minimum indicated torque in PU phase					
TQI_MIN_PU_MAX_MON	V	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm
Monitoring threshold for minimum indicated torque in PU phase					
TQI_INC_EXT_MON	V/O	8000 ... 7FFF H	-1024 .. 1023.97	0.03125	Nm
External torque demand, L2 (e.g. MSR, TCU,...)					
FAC_TQ_REQ_DRIV_TMP_MON	V	0...FFH	0...1.992	0.0078125	-
Scalling factor for monitoring driver request					
FAC_TQ_REQ_DRIV_1_MON	V	0...FFH	0...1.992	0.0078125	-
Scalling factor for monitoring driver request, without gear factor					
FAC_TQ_REQ_DRIV_GEAR_MON	V	0...FFH	0...1.992	0.0078125	-
Gear dependent driver request weigting factor					

Input data:

PV_AV_MON	TQI_MIN_PU	TQI_INC_CAN_MON	N_32_MON

FUNCTION DESCRIPTION:

General information:

The objective of this function is to plausibilise the adaptive drive dynamics factor coming from the function level. This factor is used to calculate an adapted driver request PV_AV_AD_MON which is an important input for the module 'Desired indicated engine torque'. The idea behind this changed PV_AV_MON value is to adapt the engine response to the driver request depending on his or her driving-behaviour i.e. either a very sporty driver(quite active on the gas pedal) or a relatively passive driver(not so active on the gas pedal).

The torque demands from cruise control, electronic gear shift control and the electronic stability program are set to zero, if these functions are not available. Otherwise, the torque demands have to be copied to the corresponding variable of level 2 and these variables have to be checked for plausibility (e.g. CAN monitoring, CRU monitoring, control units [ESP, EGS] with internal plausibility check).

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1

Deactivation: otherwise

Initialisation: for condition see 'Application Incidences of Process Monitoring'
PV_AV_AD_MON = 0%
TQI_MIN_PU_MON = 0 Nm

TQI_MIN_PU_MAX_MON = 0 Nm
TQI_INC_EXT_MON = 0 Nm
FAC_TQ_REQ_DRIV_1_MON = 0 Nm
FAC_TQ_REQ_DRIV_GEAR_MON = 0 Nm
FAC_TQ_REQ_DRIV_TMP_MON = 0 Nm

Update Rate: 40 ms

Formula section:

ACTION_ECM3_Service0TaskPfm(4)

The scalling factor for monitoring driver request is calculated from a maximum choice between AT and MT scalling factor of level 1 and multiplied with a weighting factor for driver request as a function of gear and pedal value.

Caution:
For this the unplausibilized level 1 information gear has been used. This can only be done, if the worst case between the different gear is not leading to a too important variation of the desired indicated engine torque.

In order to keep the old core module "Desired indicated engine torque", the calculated scalling factor for monitoring driver request, is calculated back to a modified pedal value signal, witch is used in the module "Desired indicated engine torque",to calculate the desired indicated engine torque.

FAC_TQ_REQ_DRIV_1_MON =
IP_FAC_TQ_REQ_DRIV_1_MON(N_32_MON, PV_AV_MON)

FAC_TQ_REQ_DRIV_GEAR_MON =
IP_FAC_GEAR_TQ_REQ_DRIV_MON(GEAR, PV_AV_MON)

FAC_TQ_REQ_DRIV_TMP_MON =
FAC_TQ_REQ_DRIV_GEAR_MON * FAC_TQ_REQ_DRIV_1_MON

PV_AV_AD_MON = FAC_TQ_REQ_DRIV_TMP_MON * 50

TQI_INC_EXT_MON = TQI_INC_CAN_MON

{Minimum indicated engine torque in trailing throttle}

TQI_MIN_PU_MAX_MON = IP_TQI_MIN_PU_MAX_MON(N_32_MON, TCO_MON)

TQI_MIN_PU_MON = **min**(TQI_MIN_PU, TQI_MIN_PU_MAX_MON)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_MIN_PU_MAX_MON	6x3	0 ... 7FFF H	0 ... 1023.97	0.03125	Nm
LDP_N_32_MON_IP_TQI_PU_MAX_MON	6	0 .. FF H	0 .. 8160	32	Rpm
LDP_TCO_MON_IP_TQI_PU_MAX_MON	3	0..FEH	-48..142.5	0.75	°C
Monitoring threshold for minimum indicated torque in PU phase					
IP_FAC_TQ_REQ_DRIV_1_MON	8x8	0...FFH	0...1.992	0.0078125	-
LDP_N_32_MON_IP_FAC_TQ_1_MON	8	0...FFH	0...8160	32	rpm
LDP_PV_AV_MON_IP_FAC_1_MON	8	0...FFH	0...99.6	0.39	%
Scaling factor for requested torque at clutch					
IP_FAC_GEAR_TQ_REQ_DRIV_MON	8x8	0...FFH	0...1.992	0.0078125	-
LDP_GEAR_FAC_TQ_REQ_DRIV_MON	8	0..FFH	0...255	1	-
LDP_PV_AV_MON_FAC_TQ_REQ_GEAR	8	0...FFH	0...99.6	0.39	%
Gear dependent driver request weigting factor					

E.11.2 Forced injection shut off for Level 2 - engine speed limitation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_OFF_IV_N_LIM_ETC_MON	O/V	0 ... 1 H	0 ... 1	1	-
Reversible fuel Cut Off Switch for all cylinders					

Input data:

N_32_MON	LV_N_LIM_ETC_MON		

FUNCTION DESCRIPTION:

General information:

In order to ensure a safe injection cut off while engine speed limitation is active (LV_N_LIM_ETC_MON ==1) and the engine speed is above the threshold C_N_THD_FCUT_N_LIM_ETC_MON the resulting logical variable LV_OFF_IV_N_LIM_ETC_MON is used at individual cylinder shut off functionality.
(same use and priority as LV_OFF_IV_MON).

Application conditions:

Activation: at every engine state

Deactivation: otherwise

Initialisation: for condition see ‘Application Incidences of Process Monitoring’

LV_OFF_IV_N_LIM_ETC_MON = 0

Update Rate: 40 ms

Formula section:

IF (LV_N_LIM_ETC_MON == 1)
AND (N_32_MON > C_N_THD_FCUT_N_LIM_ETC_MON)
THEN LV_OFF_IV_N_LIM_ETC_MON = 1 ;
ELSE LV_OFF_IV_N_LIM_ETC_MON = 0 ;

ACTION_ECM3_Service1TaskPfm(4)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_THD_FCUT_N_LIM_ETC_MON	1	0 ... FF H	0 ... 8160	32	rpm
Engine speed threshold for forced injection cut off in SAS mode					

E.12 Monitoring of analog-digital-conversion

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_CONV_MON	O/V	0...1H	0...1	1	-
Logical variable for ADC error					
ABC_CONV_MON	V	0...FFH	0...255	1	-
Anti bounce counter ADC error					

Input data:

ACQ_MC_0_MON	ACQ_MU_0_MON		

Import actions:

ACTION_ECM3_Service2TaskPfm(IN <n>)
ACTION_ECM3_Service3TaskPfm(IN <n>)

Note: This actions are define in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 1.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are define in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode O and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ACQ_CONV_HYS_MON	1	0...FFH	0...4.98	0.01953	V
Maximum allowed difference between the voltages on main controller and on monitoring unit					
C_ABC_INC_CONV_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment (additive value in case of ADC error)					
C_ABC_MAX_CONV_MON	1	1...FFH	1...255	1	-
Value at which ADC error is recognized, when reached					

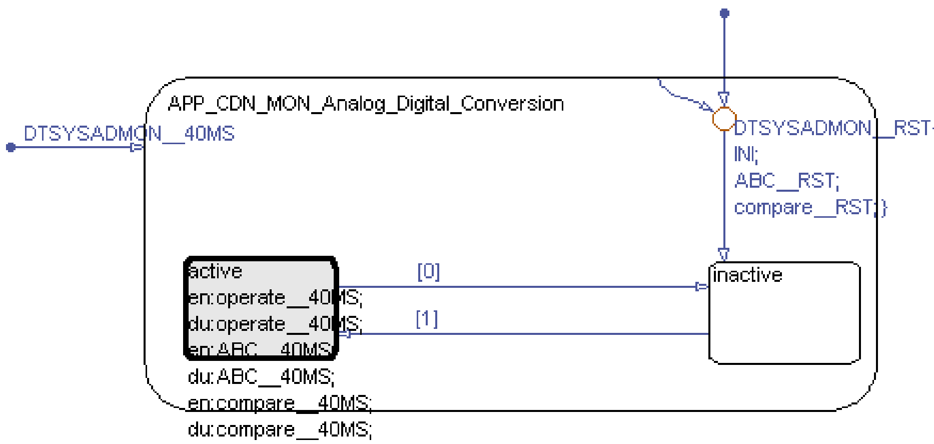
E.12.1 General Information:

The analog-digital-converter (ADC) diagnosis is performed by comparing two digital signals. The according analog signal is the same, it is digitalized once on the main controller (ACQ_MC_0_MON) and also on the monitoring unit (ACQ_MU_0_MON).

On an error free system both values are not allowed to differ more than the hysteresis C_ACQ_CONV_HYS_MON.

This hysteresis also covers differences caused by the different sampling times.

Application Condition



Note: DTSYSADMON__RST includes the function calls as defined in application incidences.

Function Description

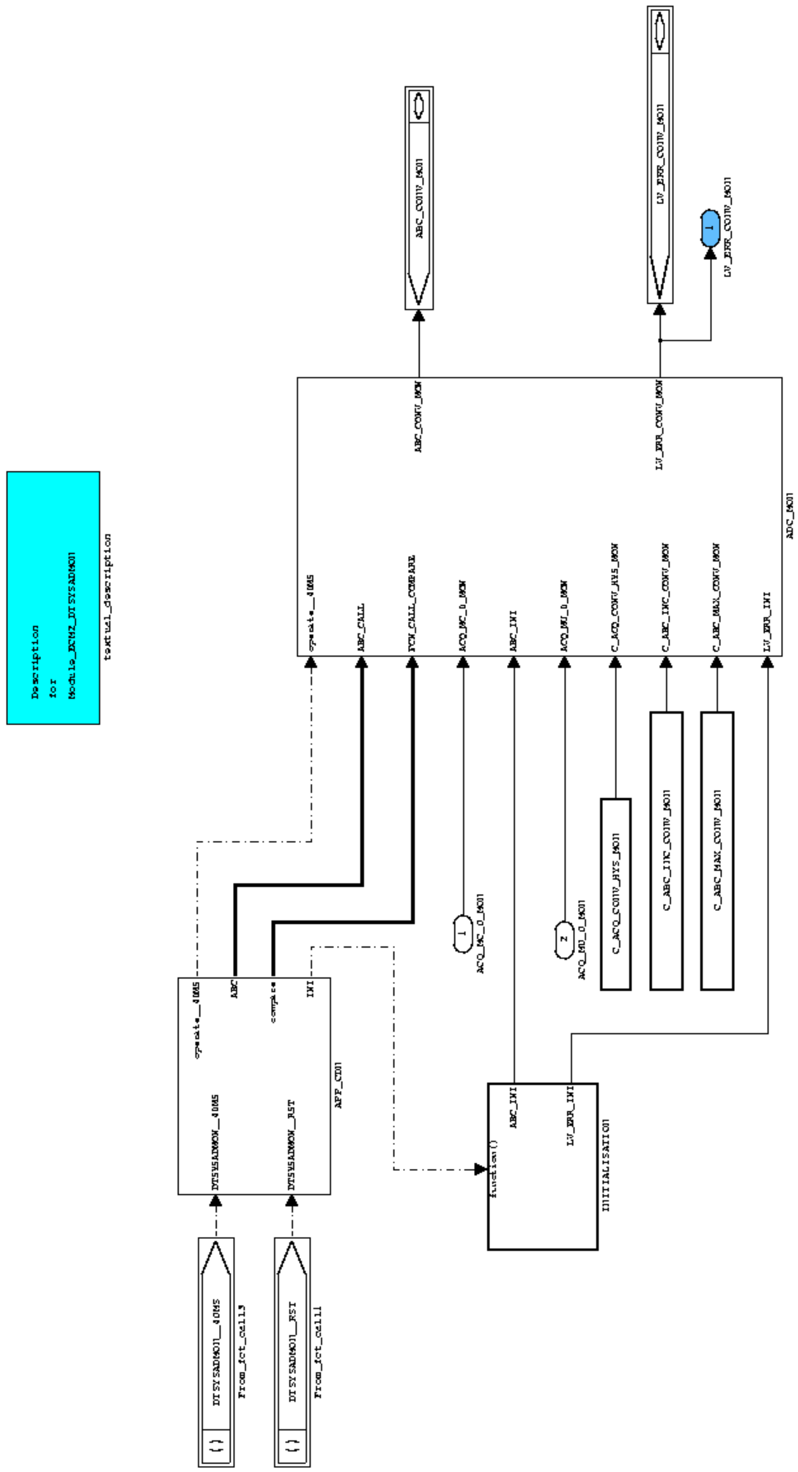


Figure 4 ECM2_DTSYSADMON

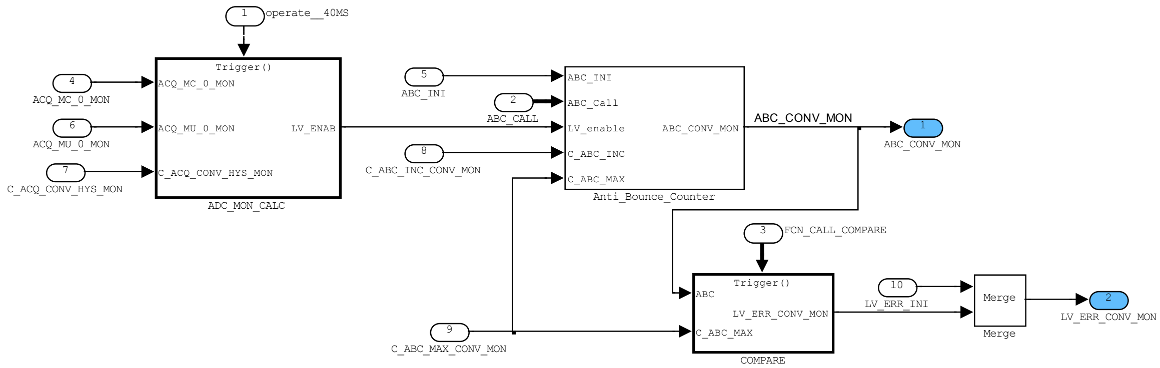


Figure 5 ECM2_DTSYSADMON/ ADC_MON

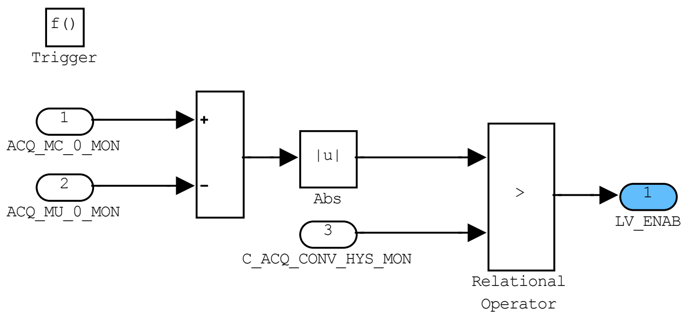


Figure 6 ECM2_DTSYSADMON/ ADC_MON/ ADC_MON_CALC

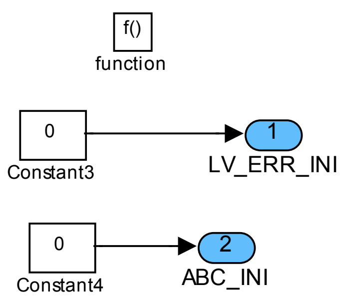


Figure 7 ECM2_DTSYSADMON/ INITIALISATION

E.13 Monitoring of coolant temperature

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TCO_MON	O/V	0...FEH	-48...142.5	0.75	°C
Coolant temperature (monitoring level)					
T_TCO_GRD_MON	V	0...FFH	0...10.2	0.04	s
Timer for negative TCO gradient limitation					

Input data:

LV_TQI_MON_ACT_MON	TCO	TCO_SUB	

Import actions:

ACTION_ECM3_Service4TaskPfm(IN <n>)
ACTION_ECM3_Service5TaskPfm(IN <n>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring Services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument n has the value 1.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with mode O and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TCO_GRD_MAX_MON	1	0...FEH	0...190.5	0.75	°C
Maximum negative gradient for the coolant temperature					
C_T_TCO_GRD_MAX_MON	1	0...FFH	0...10.2	0.04	s
Minimum time before a new change of TCO_MON is allowed					
C_TCO_THD_MIN_MON	1	0...FEH	-48...142.5	0.75	°C
Threshold for the min. TCO temperature while TCO_SUB is higher than C_TCO_SUB_THD_MAX_MON					
C_TCO_SUB_THD_MAX_MON	1	0...FE00H	-48...142.5	0.00293	°C
Threshold for the max. TCO_SUB temperature while TCO is lower than C_TCO_THD_MIN_MON					

E.13.1 FUNCTION PART: ECM2_DTSYSTCO

General information:

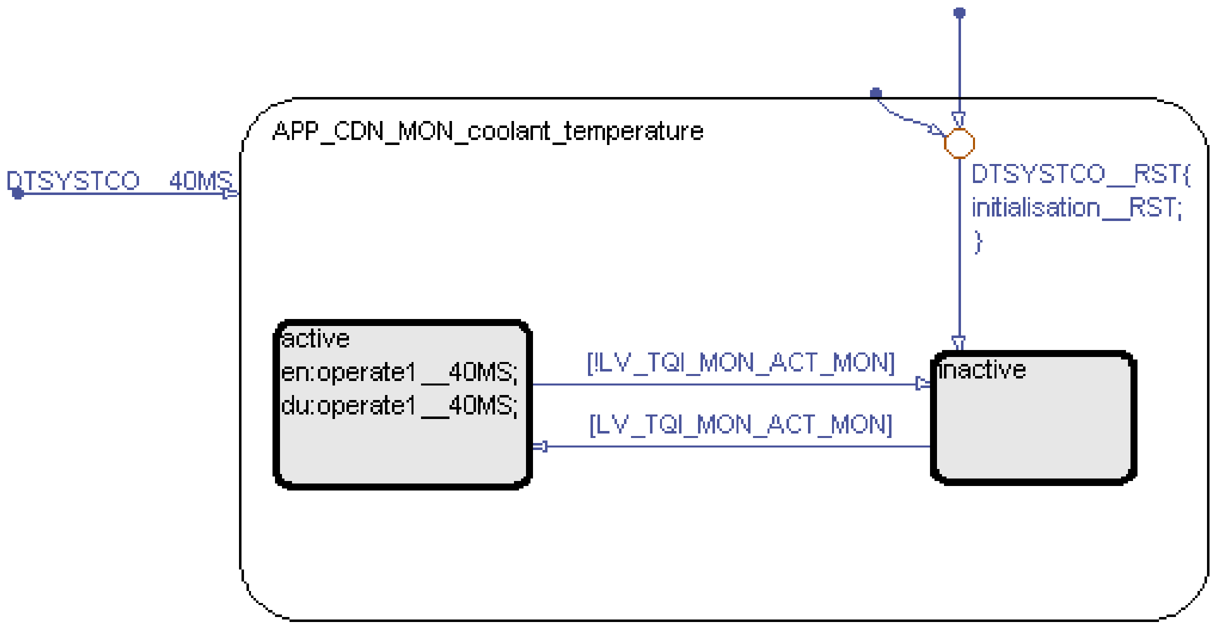
A sudden decrease in the coolant temperature TCO caused by a fault can lead to an undesired increase of the torque demand and thus to an undesired acceleration of the car because of the temperature dependent friction compensation. Therefore the negative temperature gradient of TCO shall be limited in the function level. In order to detect process faults, the gradient of the coolant temperature which is copied to the process monitoring level is also limited, i.e. to C_TCO_GRD_MAX_MON. Also the diagnosis for a too low TCO-sensor temperature is monitored. The resulting value TCO_MON is then used within the monitoring level. If a fault in TCO leads to an increased engine torque in the idle speed controller the module ‘Monitoring of idle speed controller’ detects this fault as an increased torque demand from the idle speed controller.

In case of an error free system the temperature TCO_MON used in the monitoring level should be the same as the temperature TCO used in the function level.

An additional timer is needed to compensate the different recurrencies of level 1 and level 2. The maximum counter threshold C_T_TCO_GRD_MAX_MON and the maximum negative temperature gradient C_TCO_GRD_MAX_MON need to be adapted in regard to the function recurrency and to C_TCO_GRD_MAX of level 1.

The resolution of T_TCO_GRD_MON and C_T_TCO_GRD_MAX_MON depend on the update rate of the module.

Application Condition



Note: DTSYSTCO__RST includes the function calls as defined in application incidences.

Function Description

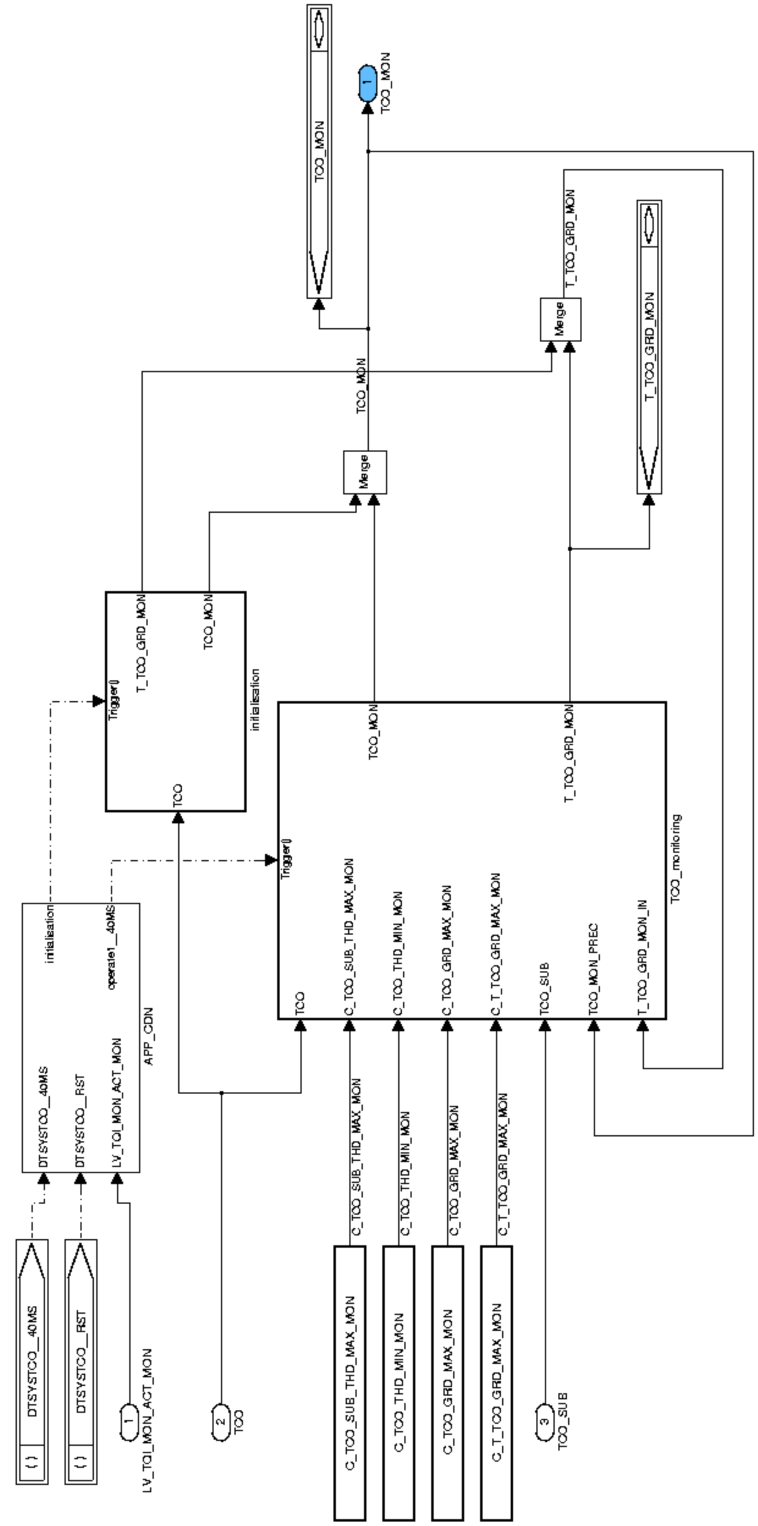


Figure 8 ECM2_DTSYSTCO

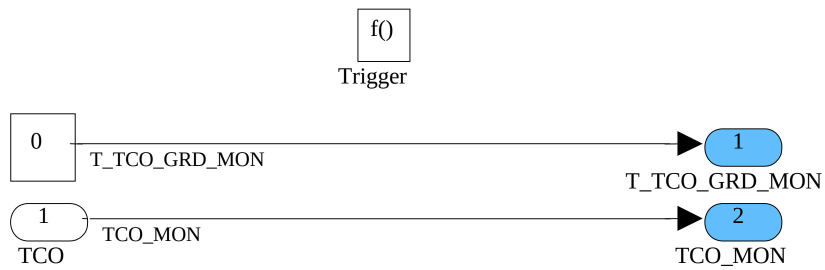


Figure 2 ECM2_DTSYSTCO/ INITIALISATION

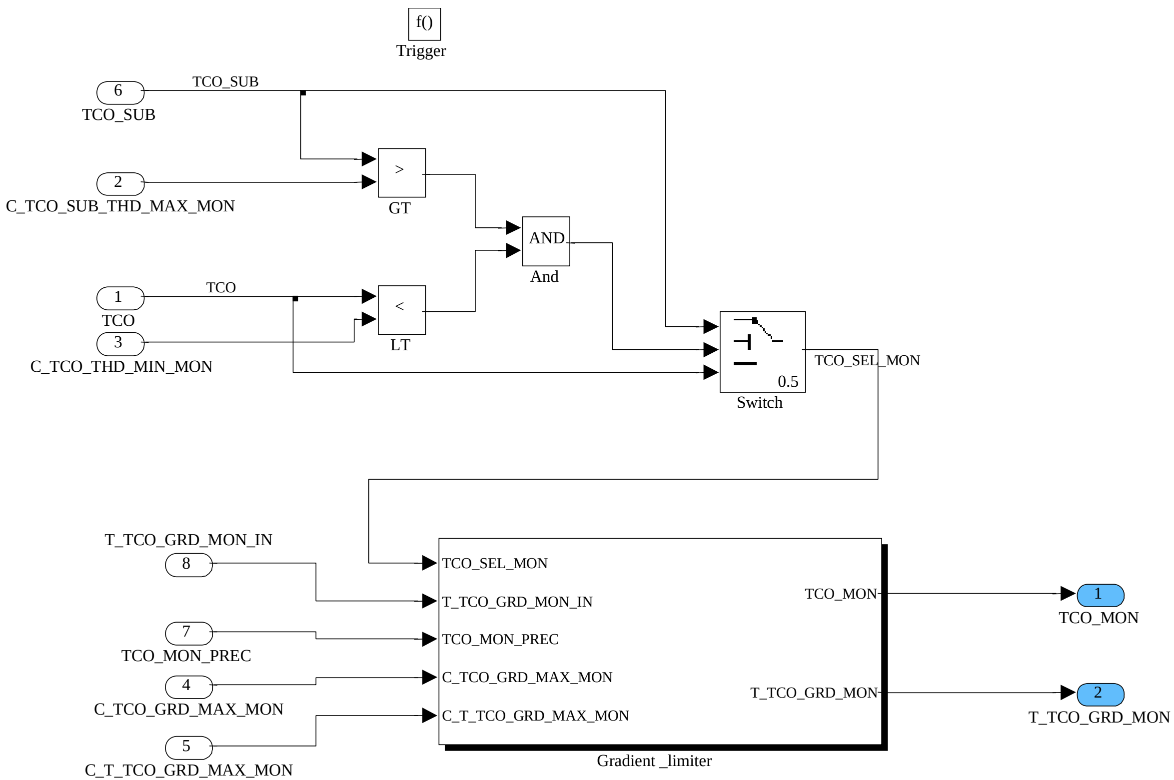


Figure 3 ECM2_DTSYSTCO/ TCO_MONITORING

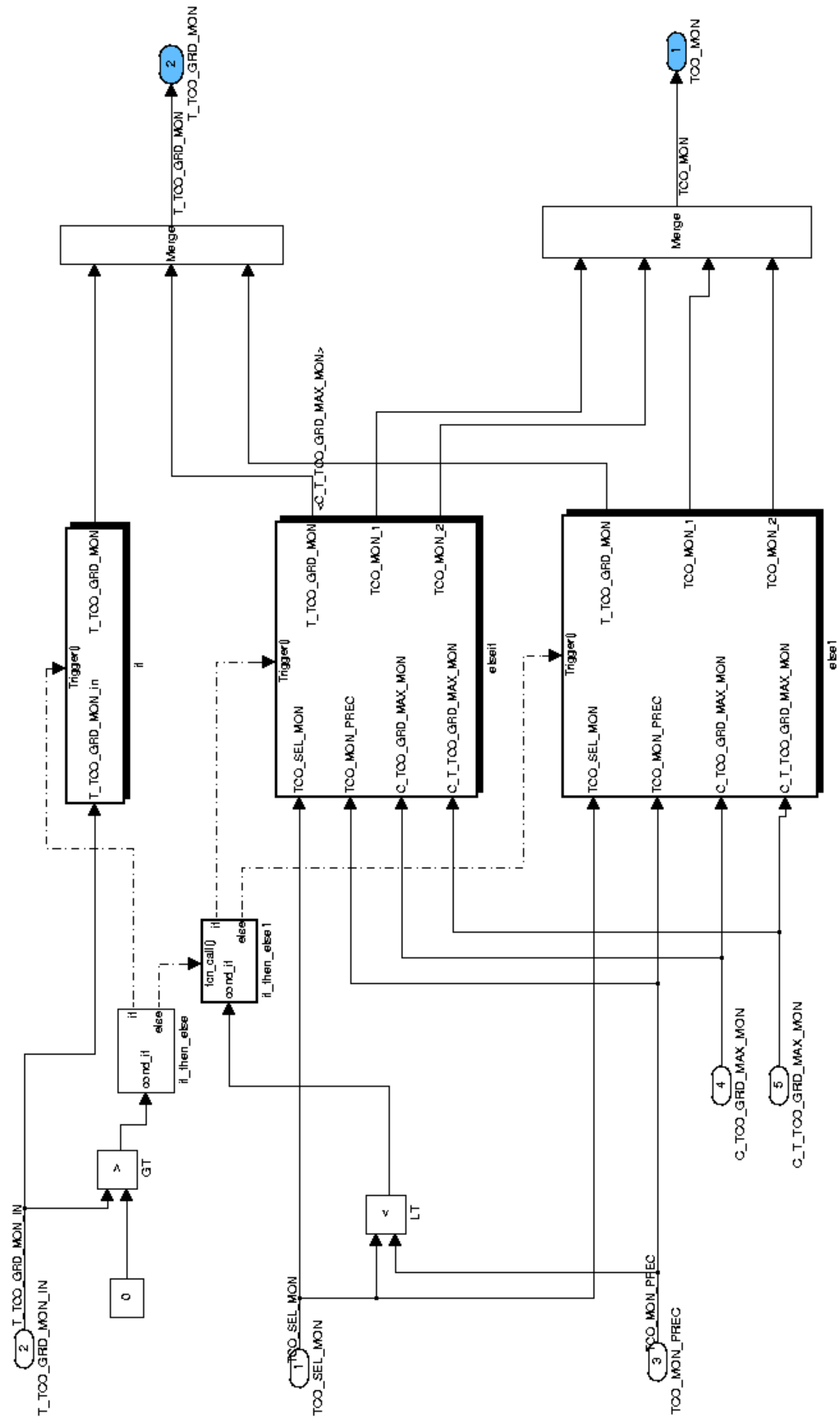


Figure 4 ECM2_DTSYSTCO/ TCO_MONITORING/ GRADIENT _LIMITER

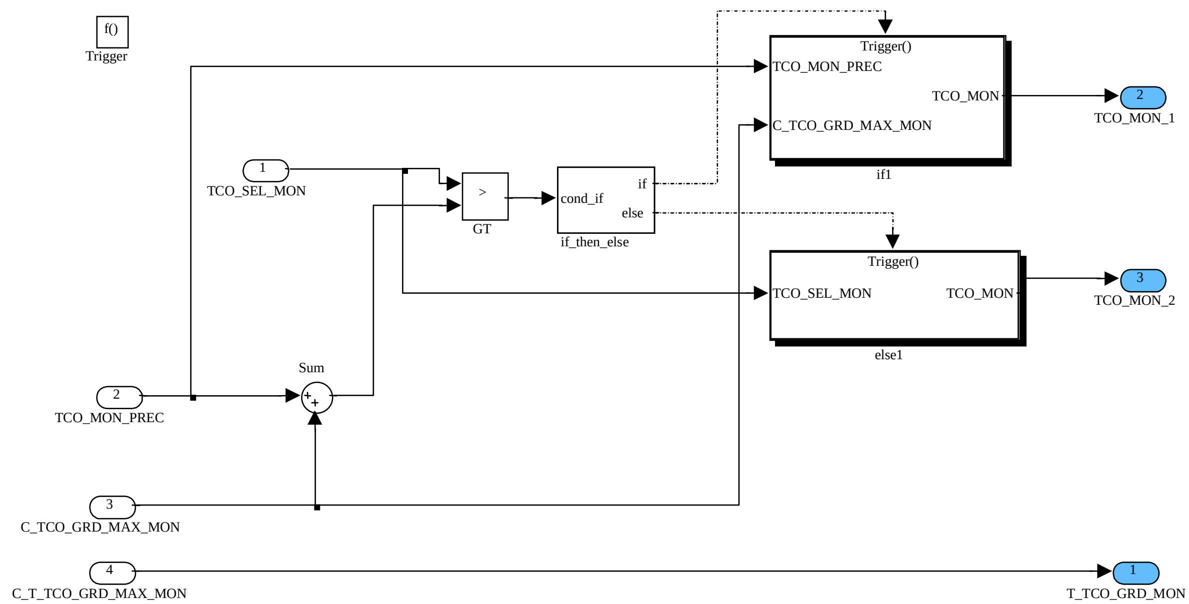


Figure 5 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ else1

Figure 6 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ else1/ else1

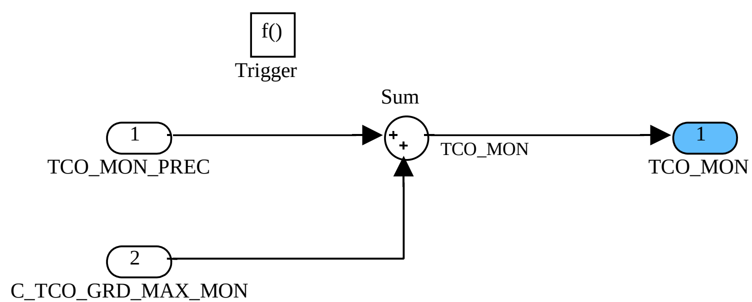


Figure 7 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ else1/ if1

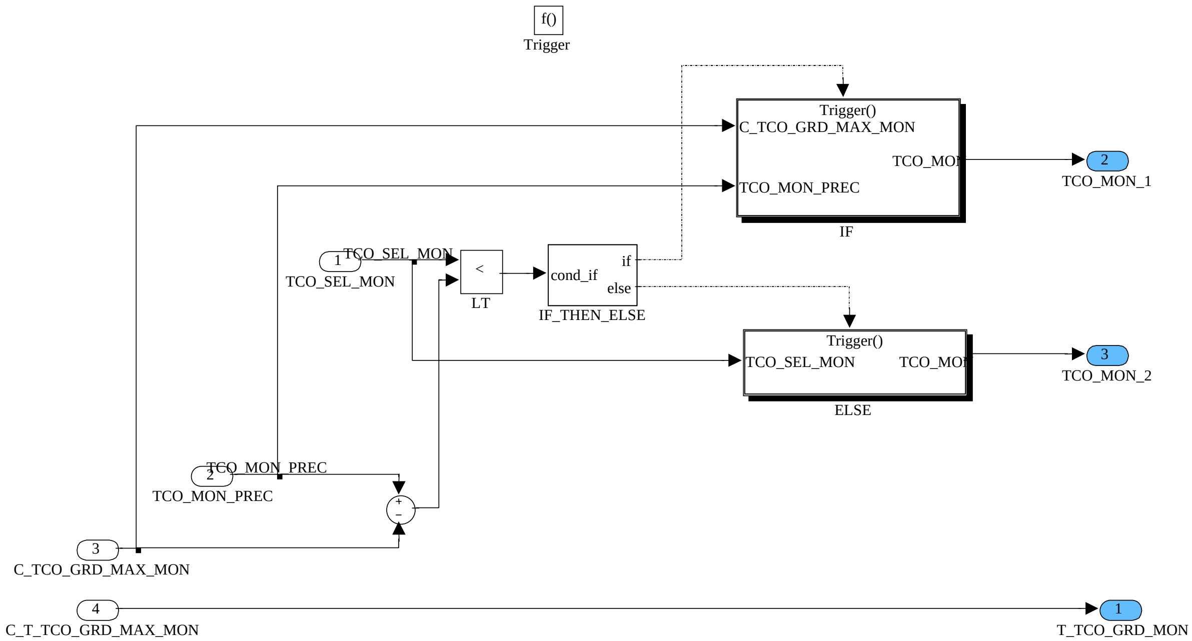


Figure 8 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ elseif

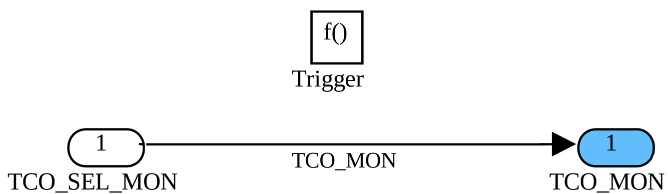


Figure 9 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ elseif/ ELSE

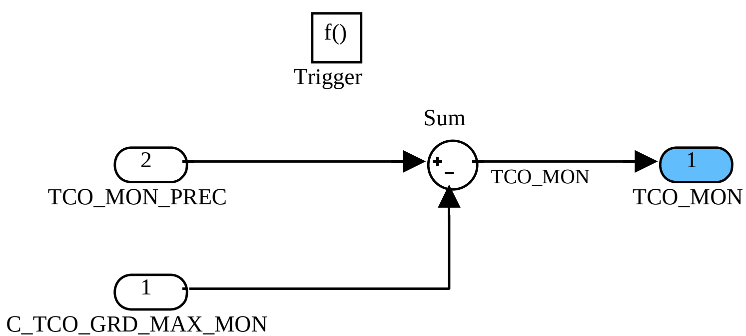


Figure 10 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ elseif/ IF

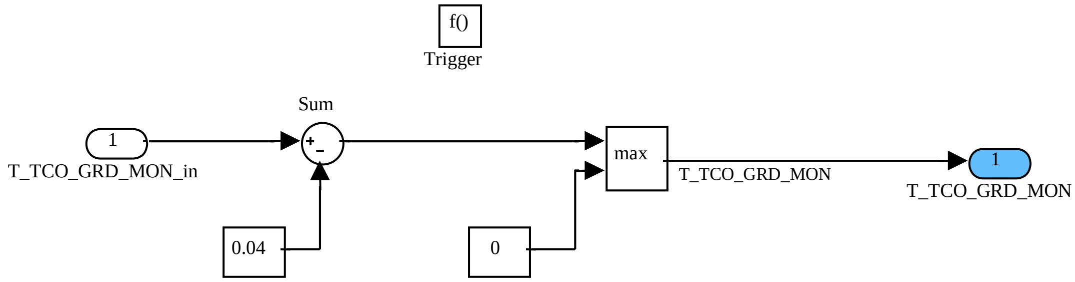


Figure 11 ECM2_DTSYSTCO/ TCO_monitoring/ Gradient _limiter/ if

E.14 Monitoring of torque demand from idle speed controller

Overview

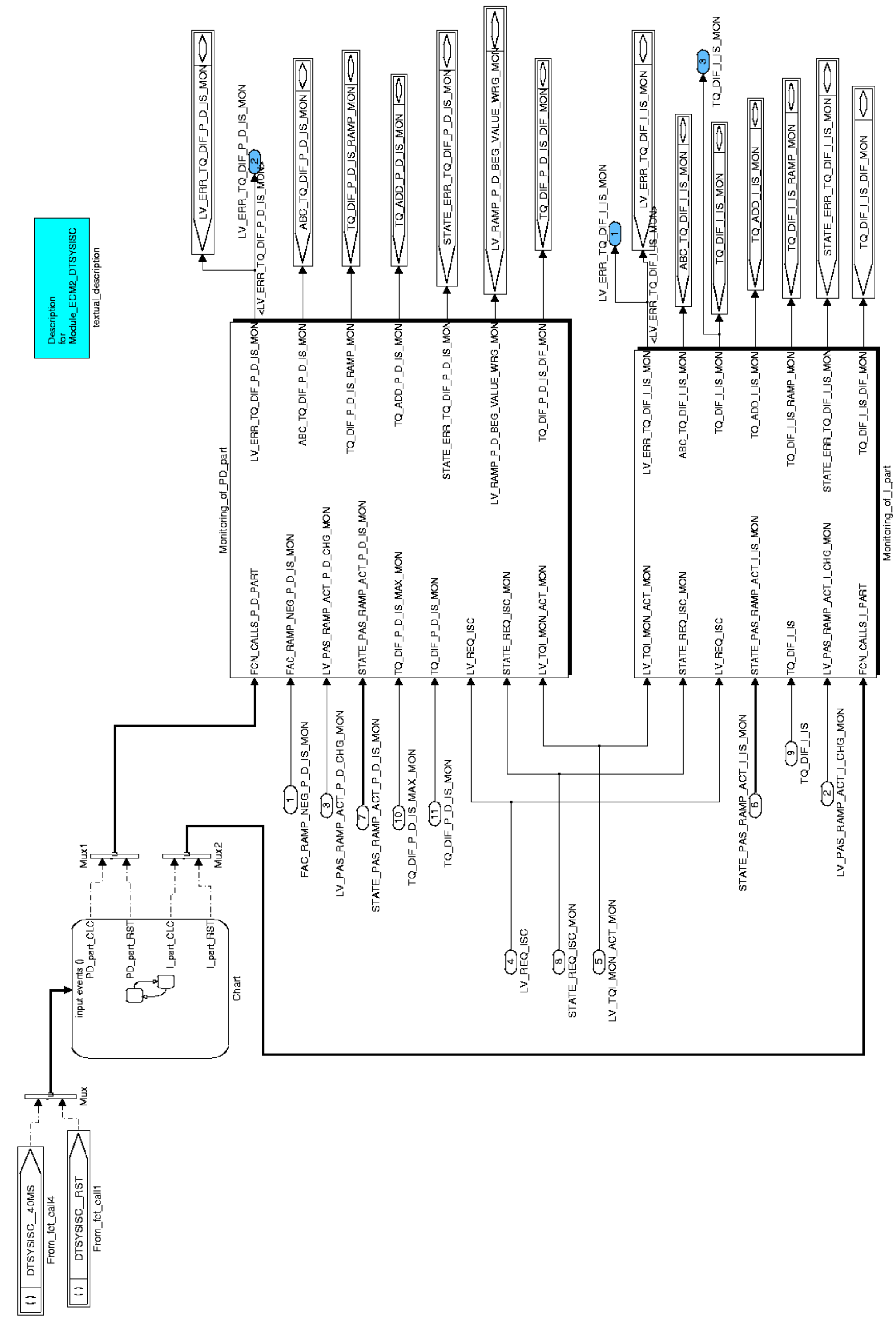


Figure 9 ECM2_DTSYSISC

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQ_DIF_I_IS_MON	O/V	0...1H	0...1	1	-
Fault currently present in torque generation, symptom ‘idle speed controller – I-part’					
LV_ERR_TQ_DIF_P_D_IS_MON	O/V	0...1H	0...1	1	-
Fault currently present in torque generation, symptom ‘idle speed controller – PD-part’					
TQ_DIF_I_IS_MON	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Monitored I-part of the idle speed controller					
ABC_TQ_DIF_I_IS_MON	V	0...FFH	0...255	1	-
Anti bounce counter for monitoring I-part of ISC					
TQ_ADD_P_D_IS_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Decrement for PD-part model value during ramp function					
TQ_DIF_I_IS_DIF_MON	V	80...7FH	-256...254	2	Nm
Difference between monitored and modelled I-part					
TQ_DIF_I_IS_RAMP_MON	V	8000...7FFFH	-256...255.992	0.0078	Nm
Model value of I-part during ramp function					
TQ_DIF_P_D_IS_DIF_MON	V	80...7FH	-256...254	2	Nm
Difference between monitored and modelled PD-part					
TQ_DIF_P_D_IS_RAMP_MON	V	0...7FFFH	0...1023.97	0.03125	Nm
Model value of PD-part during ramp function					
ABC_TQ_DIF_P_D_IS_MON	V	0...FFH	0...255	1	-
Anti bounce counter for monitoring PD-part of ISC					
LV_RAMP_P_D_BEG_VALUE_WRG_MON	V	0...1H	0...1	1	-
Logical bit indicating wrong start value of PD-ramp					
STATE_ERR_TQ_DIF_I_IS_MON	V	0H 1H 2H 3H 4H	NO_ERROR ERROR_PATH1 ERROR_PATH2 ERROR_PATH3 ERROR_PATH4	1	-
State variable for fault detection in torque generation, symptom ‘idle speed controller – I-part’					
STATE_ERR_TQ_DIF_P_D_IS_MON	V	0H 1H 2H 3H 4H	NO_ERROR ERROR_PATH1 ERROR_PATH2 ERROR_PATH3 ERROR_PATH4	1	-
State variable for fault detection in torque generation, symptom ‘idle speed controller – PD-part’					
TQ_ADD_I_IS_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Decrement for modelled I-part during ramp function					

Input data:

FAC_RAMP_NEG_P_D_IS_MON	LV_PAS_RAMP_ACT_I_CHG_MON	LV_PAS_RAMP_ACT_P_D_CHG_MON	LV_REQ_ISC
LV_TQI_MON_ACT_MON	STATE_PAS_RAMP_ACT_I_IS_MON	STATE_PAS_RAMP_ACT_P_D_IS_MON	
STATE_REQ_ISC_MON	TQ_DIF_P_D_IS_MAX_MON	TQ_DIF_P_D_IS_MON	TQ_DIF_I_IS

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQ_DIF_I_IS_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQ_DIF_I_IS_MON	1	1...FFH	1...255	1	-
Maximum of anti bounce counter					
C_TQ_DIF_I_IS_MAX_MON	1	0...FFH	0...255	1	Nm
Maximum I-part					
C_TQ_I_ISC_NOT_ACT_MAX_MON	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum I-part allowed when ISC not active					

general specification

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_TQ_PAS_I_IS_MON	1	0...FFH	0...255	1	Nm
Target value for monitoring ISC deactivation-ramp operation					
C_TQ_DIF_I_IS_ADD_MON	1	0...FFH	0...255	1	Nm
Additive constant for I-part model value					
C_FAC_RAMP_NEG_I_IS_MON	1	0...FA0H	0...0.5	0.000125	-
Factor for monitoring negative ISC deactivation-ramp of I-part					
C_ABC_INC_TQ_DIF_P_D_IS_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQ_DIF_P_D_IS_MON	1	1...FFH	1...255	1	-
Maximum of anti bounce counter					
C_TQ_P_D_ISC_NOT_ACT_MAX_MON	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Maximum PD-part allowed when ISC not active					
C_TQ_DIF_P_D_IS_ADD_MON	1	0...FFH	0...255	1	Nm
Additive constant for PD-part model value					
C_TQ_DIF_P_D_RAMP_CHK_OFS_MON	1	0...FFH	0...255	1	Nm
Offset for monitoring ramp start value					

Import actions:

ACTION_ECM3_Service8TaskPfm(1)
ACTION_ECM3_Service9TaskPfm(1)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 1.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

E.14.1 Monitoring of PD controller

General information:

The objective of this function is the monitoring of the torque demand from the PD-part of the idle speed controller function. Depending on the speed deviation to the idle speed setpoint and the coolant temperature a maximum PD-part is generated (the worst case values for the PD-part in the function level, i.e. great PD-part values are allowed for the most negative speed gradients => for N < N_SP_IS_MON and for N > N_SP_IS_MON small PD-part values are allowed). To additionally monitor the activation of the idle-speed controller in L1, a model value(STATE_REQ_ISC_MON) of the activation status is calculated in the module 'Application incidences of process monitoring'. To effectively plausiblise the values of TQ_DIF_P_D_IS and TQ_DIF_I_IS from the function level, both the monitoring thresholds as well as the activation condition model value are used.

The functionality that is depicted in the following pages consists of two fault scenarios:

Error Path – ISC not active:

Case 1: If the ISC is in the ramp limit operation due to deactivation of ISC, this ramp function is monitored. If LV_PAS_RAMP_ACT_P_D_CHG_MON is 1(i.e. indicates a change of STATE_PAS_RAMP_ACT_P_D_IS(0) from 0 to 1 or a flank change between two L2 sample times), the ramp calculation is triggered. It is first checked if the L1 ramp start-value is below the maximum PD-part. If this is the case, a ramp decrement TQ_ADD_P_D_ IS_MON is calculated and the L1 start-value,TQ_DIF_P_D_IS_MON, plus a calibratable additive constant is copied to TQ_DIF_P_D_IS_RAMP_MON. During the ramp operation, the ramp model value is calculated by adding the ramp decrement to the model value ($TQ\_DIF\_P\_D\_IS\_RAMP\ \_MON_{k-1}$) from the previous recurrency. If however the ramp start-value is greater than the threshold, TQ_DIF_P_D_IS_MAX_MON, then LV_RAMP_P_D_BEG_VALUE_WRG_MON is set to one. The difference between monitored and modelled PD-part is set as a visible variable for application work.

In the error detection part of this functionality, either $TQ\_DIF\_P\_D\_IS\_MON_k$ > TQ_DIF_P_D_IS_RAMP_MON or LV_RAMP_P_D_BEG_VALUE_WRG_MON==1 will lead to the error flag STATE_ERR_TQ_DIF_P_D_IS_MON being set.

Case 2: If the ISC is not active and the engine in the overspeed range i.e. STATE_ REQ_ISC_MON = NOT_ACTIVE, a torque demand from the PD-part of the controller greater than the passive value of zero indicates an error in the function level.

Error Path – ISC active:

Case 1: The ISC is when the engine is in idle state(LV_IS=1) or under certain conditions also in trailing-throttle(LV_PU=1) or part-load(LV_PL=1). So all of these possible "ISC-active" phases have to be monitored. A fault is set, if L1 declares the ISC be active and if L2 also detects the same based on plausiblised activation conditions (STATE_ REQ_ISC_MON = IDLE or TRAILING_THR or PART_LOAD) and if the monitored PD-part TQ_DIF_P_D_IS_MON exceeds the maximum PD-part TQ_DIF_P_D_IS_MAX_ MON.

Case 2: If in L1 the ISC is active(LV_REQ_ISC=1) but the activation conditions in L2 are not fulfilled i.e. STATE_ REQ_ISC_MON ≠ IDLE or TRAILING_THR or PART_LOAD, then if TQ_DIF_P_D_IS_MON is greater than a threshold value (C_TQ_P_D_ISC_NOT_ ACT_MAX_MON), the error-flag STATE_ERR_TQ_DIF_P_D_IS_MON is set .

If one of the errors above is detected, a fault is declared after debouncing by the idle speed controller montoring.

Function Description

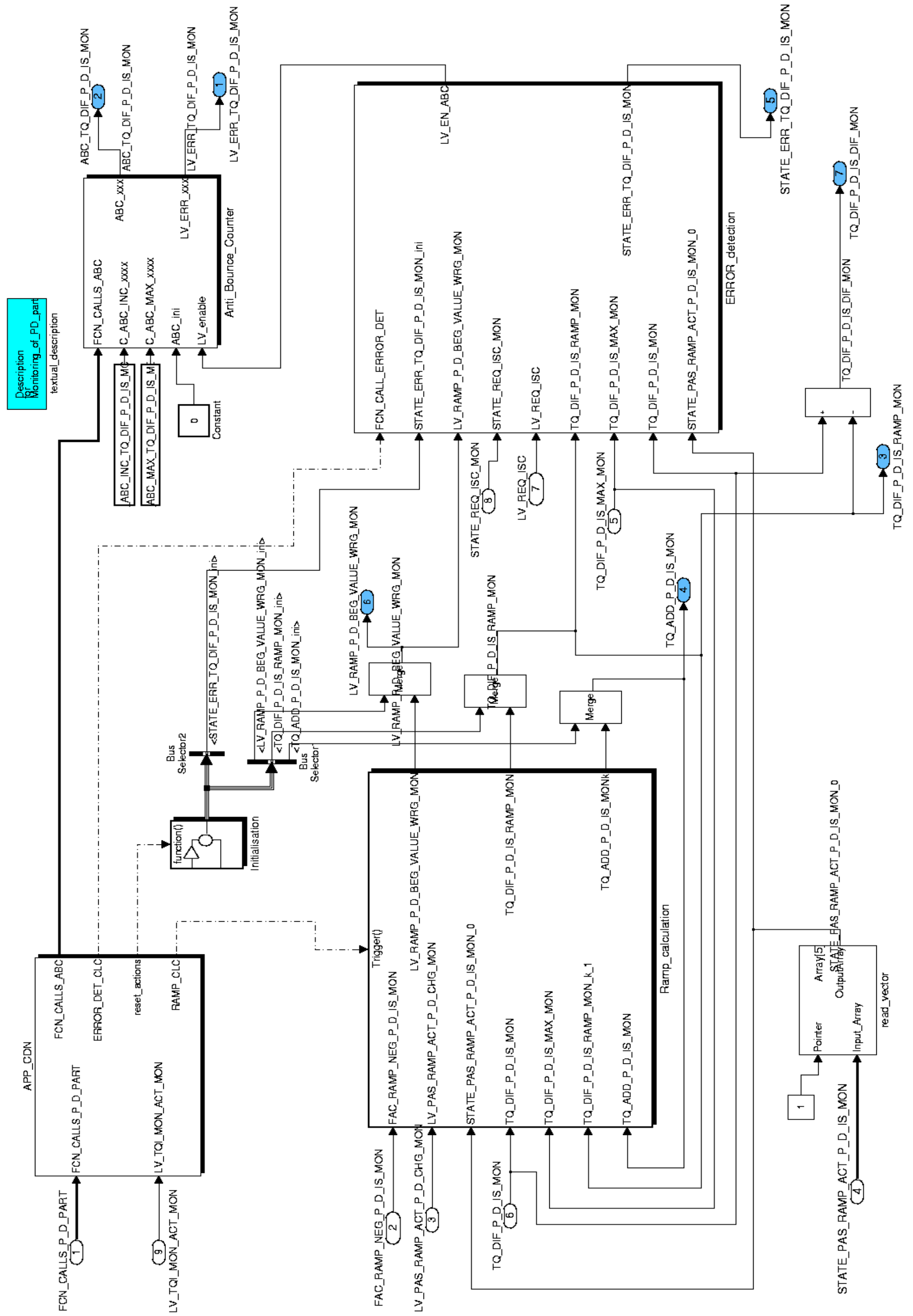
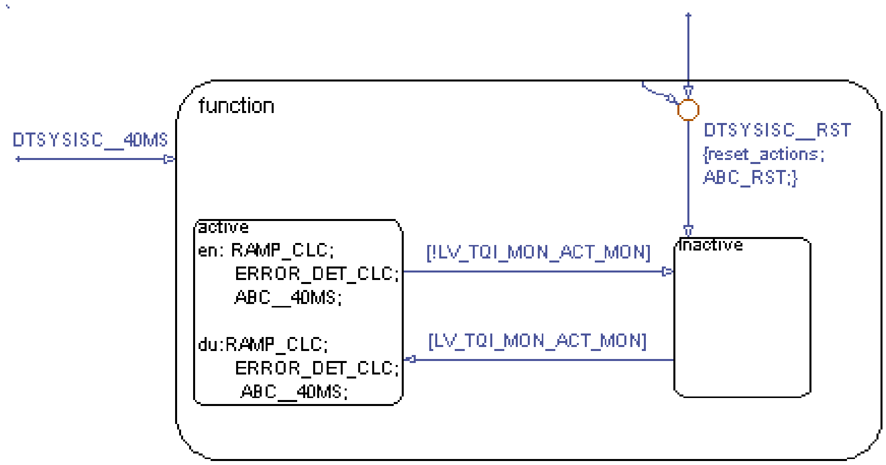
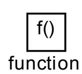


Figure 10 ECM2_DTSYSISC/ Monitoring_of_PD_part

Application Conditions



***Initialisation(DTSYSISC__RST)*:** for condition see 'Application Incidences of Process Monitoring'.



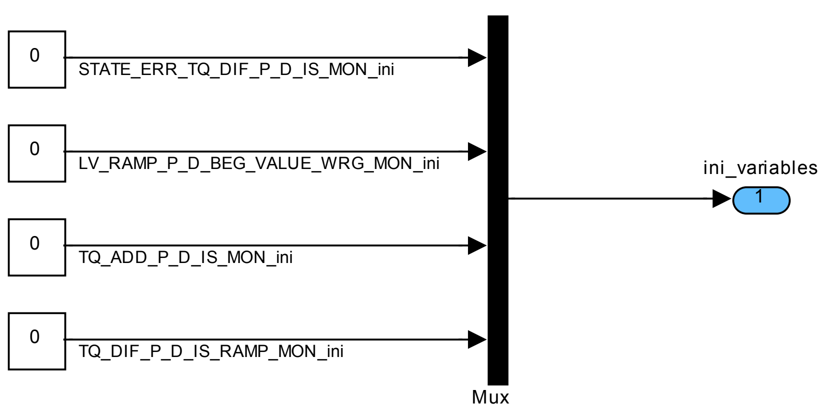


Figure 11 ECM2_DTSYSISC/ Monitoring_of_PD_part/ Initialisation

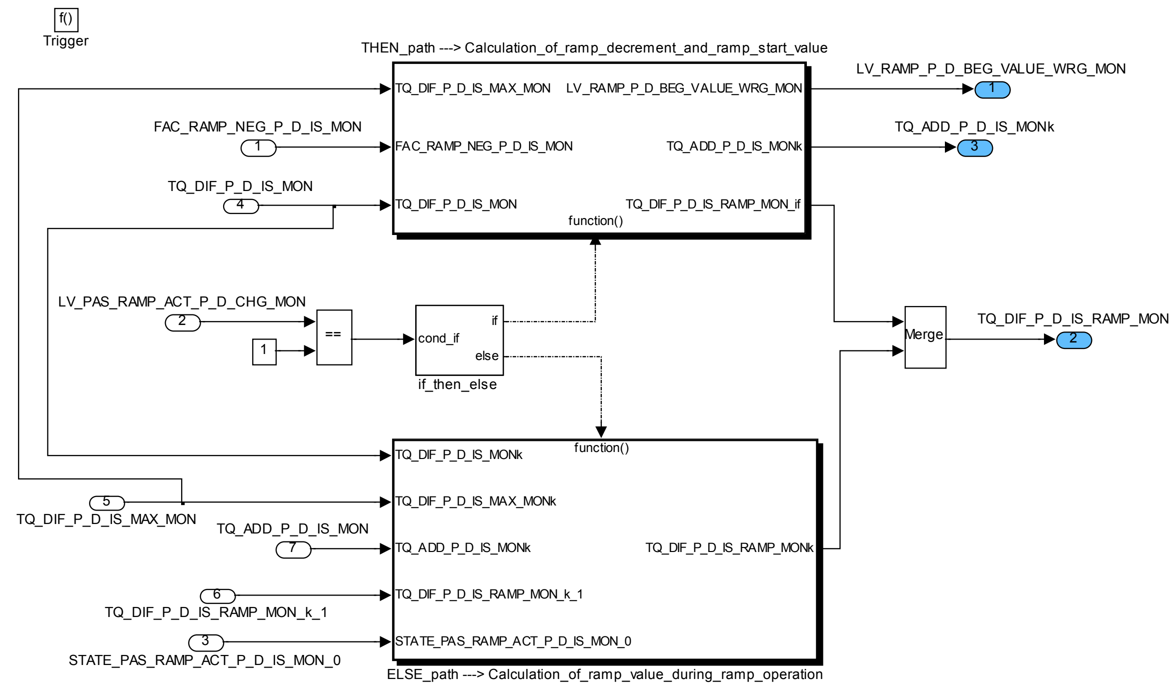


Figure 12 ECM2_DTSYSISC/ Monitoring_of_PD_part/ Ramp_calculation

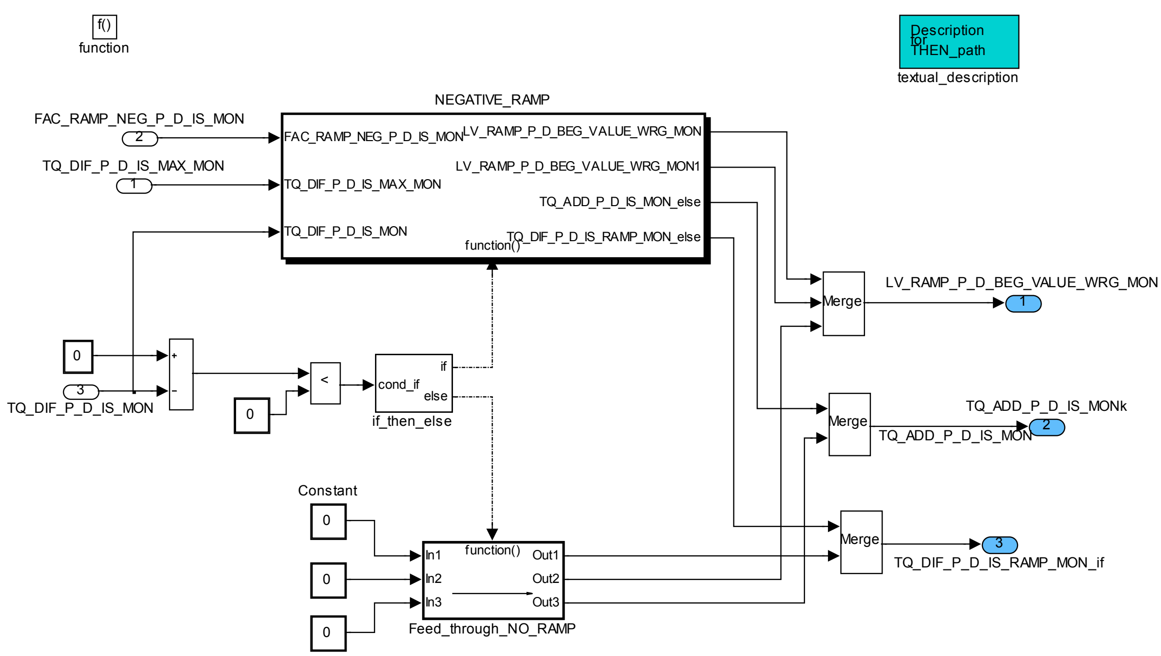
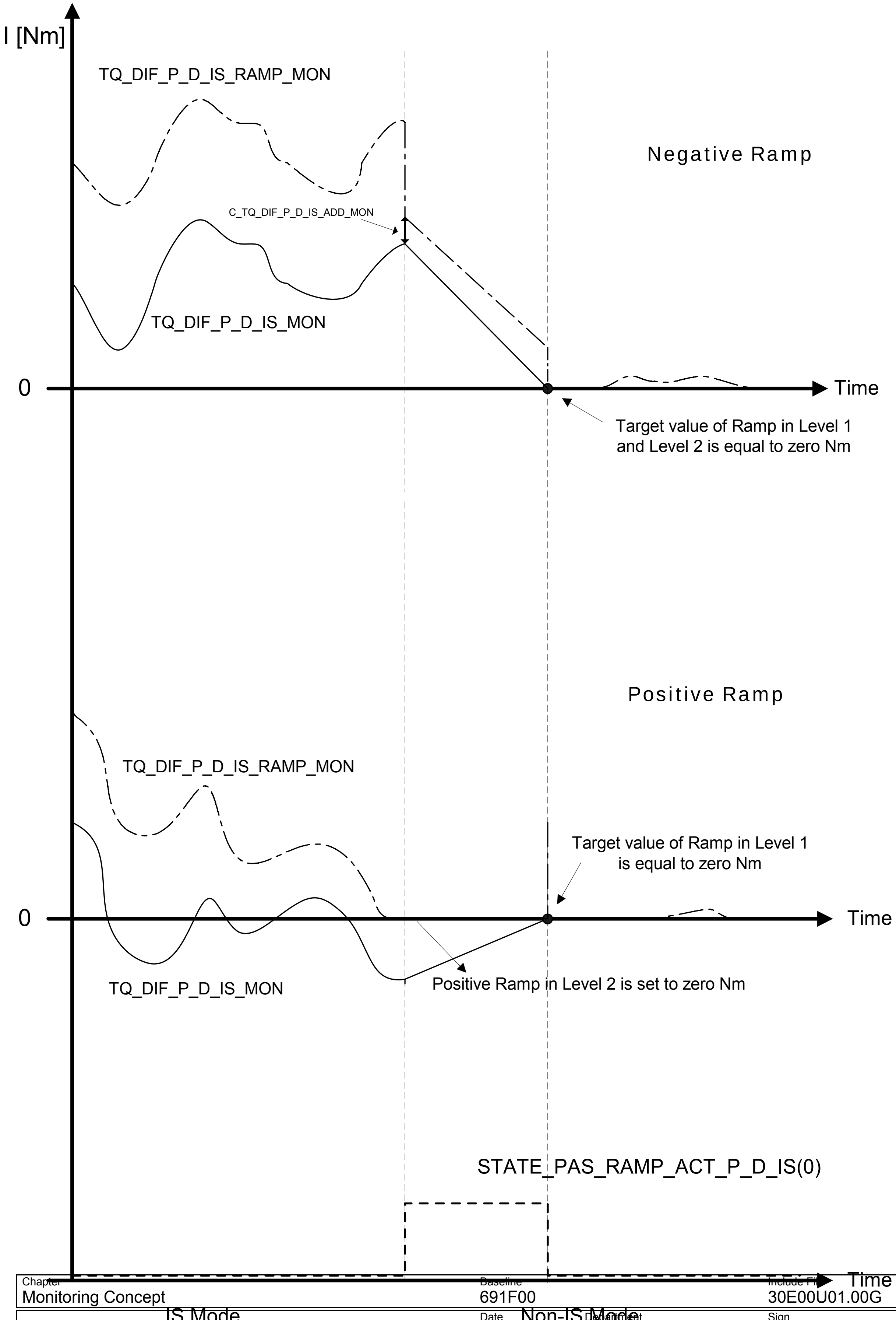


Figure 13 ECM2_DTSYSISC/ Monitoring_of_PD_part/ Ramp_calculation/ THEN_path

NOTE: The minimum PD-ramp increment/decrement is equal to the resolution of TQ_ADD_P_D_IS_MON. If the theoretically calculated *absolute value* of TQ_ADD_P_D_IS_ MON **is *smaller* than the resolution and not equal to zero**, then is to be limited to the resolution. This would lead to a


I [Nm]
TQ_DIF_P_D_IS_RAMP_MON
Negative Ramp
C_TQ_DIF_P_D_IS_ADD_MON
TQ_DIF_P_D_IS_MON
0
Time
Target value of Ramp in Level 1
and Level 2 is equal to zero Nm
Positive Ramp
TQ_DIF_P_D_IS_RAMP_MON
Target value of Ramp in Level 1
is equal to zero Nm
0
Time
TQ_DIF_P_D_IS_MON
Positive Ramp in Level 2 is set to zero Nm
STATE_PAS_RAMP_ACT_P_D_IS(0)
Time
IS Mode
Non-IS Mode

slower/faster descent of the deactivation ramp i.e. a shorter ramp time. If the theoretical value of TQ_ADD_P_D_IS_MON is equal to zero, this value is then maintained.

For e.g: theoretical value = -0.04 Nm ---> Limited value = -0.03125 Nm

theoretical value = -0.01 Nm ---> Limited value = -0.03125 Nm

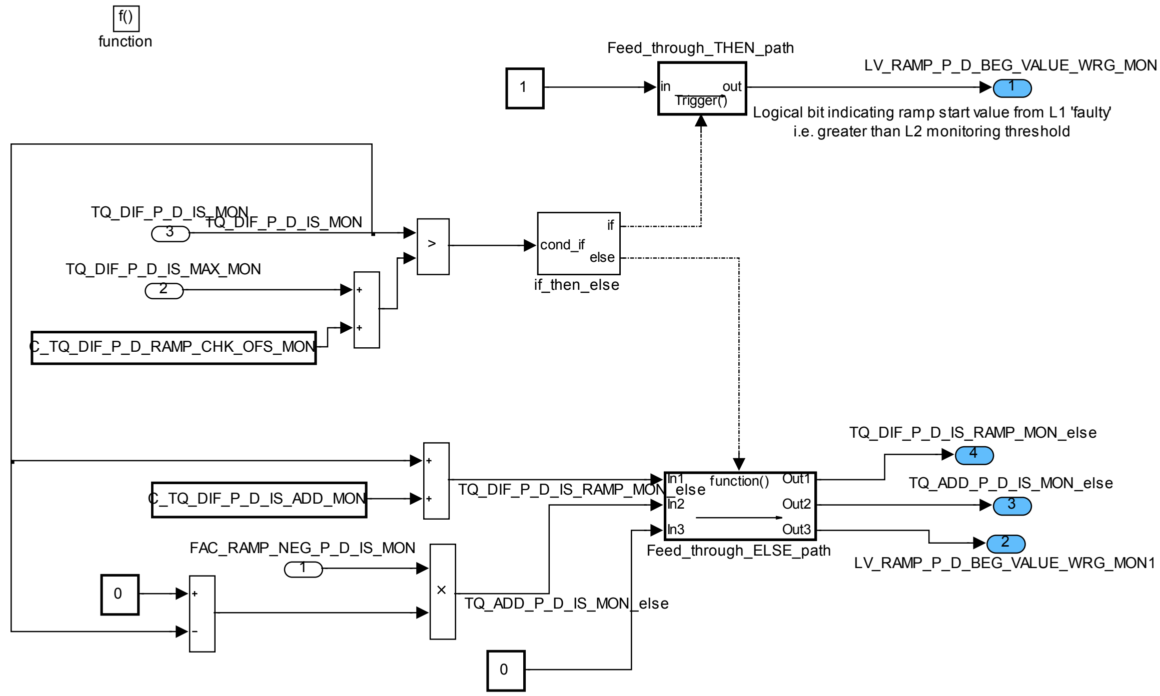


Figure 14 ECM2_DTSYSISC/ Monitoring_of_PD_part/ Ramp_calculation/ THEN_path/ NEGATIVE_RAMP

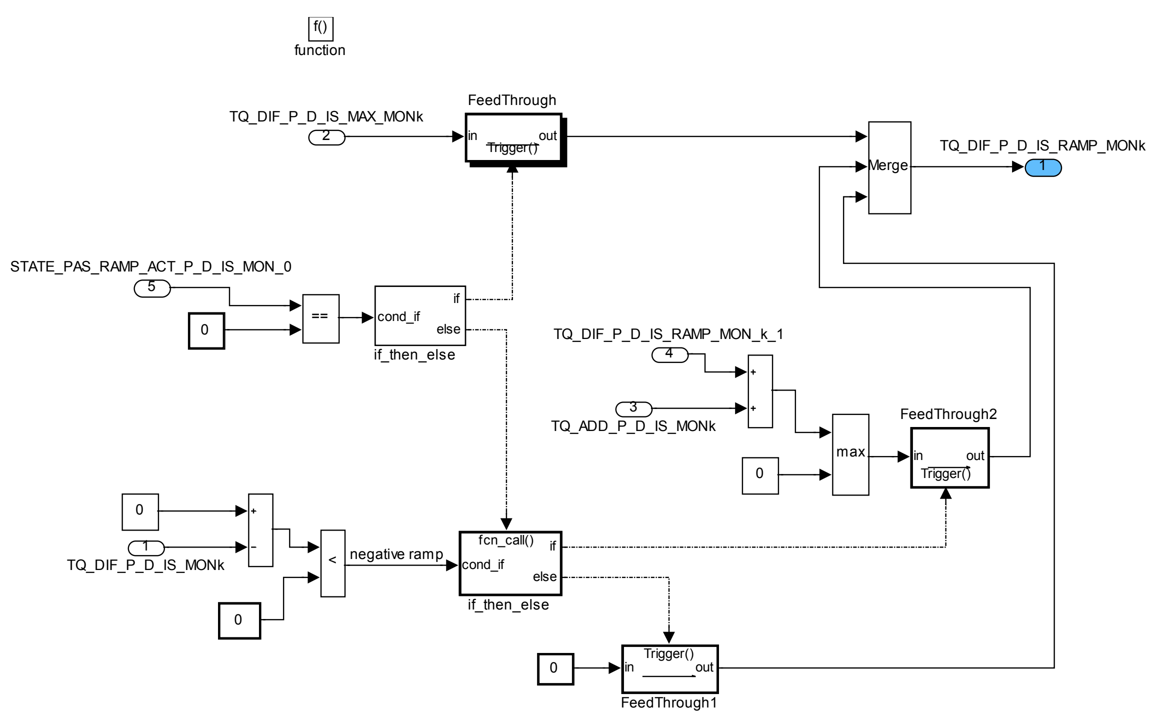


Figure 15 ECM2_DTSYSISC/ Monitoring_of_PD_part/ Ramp_calculation/ ELSE_path

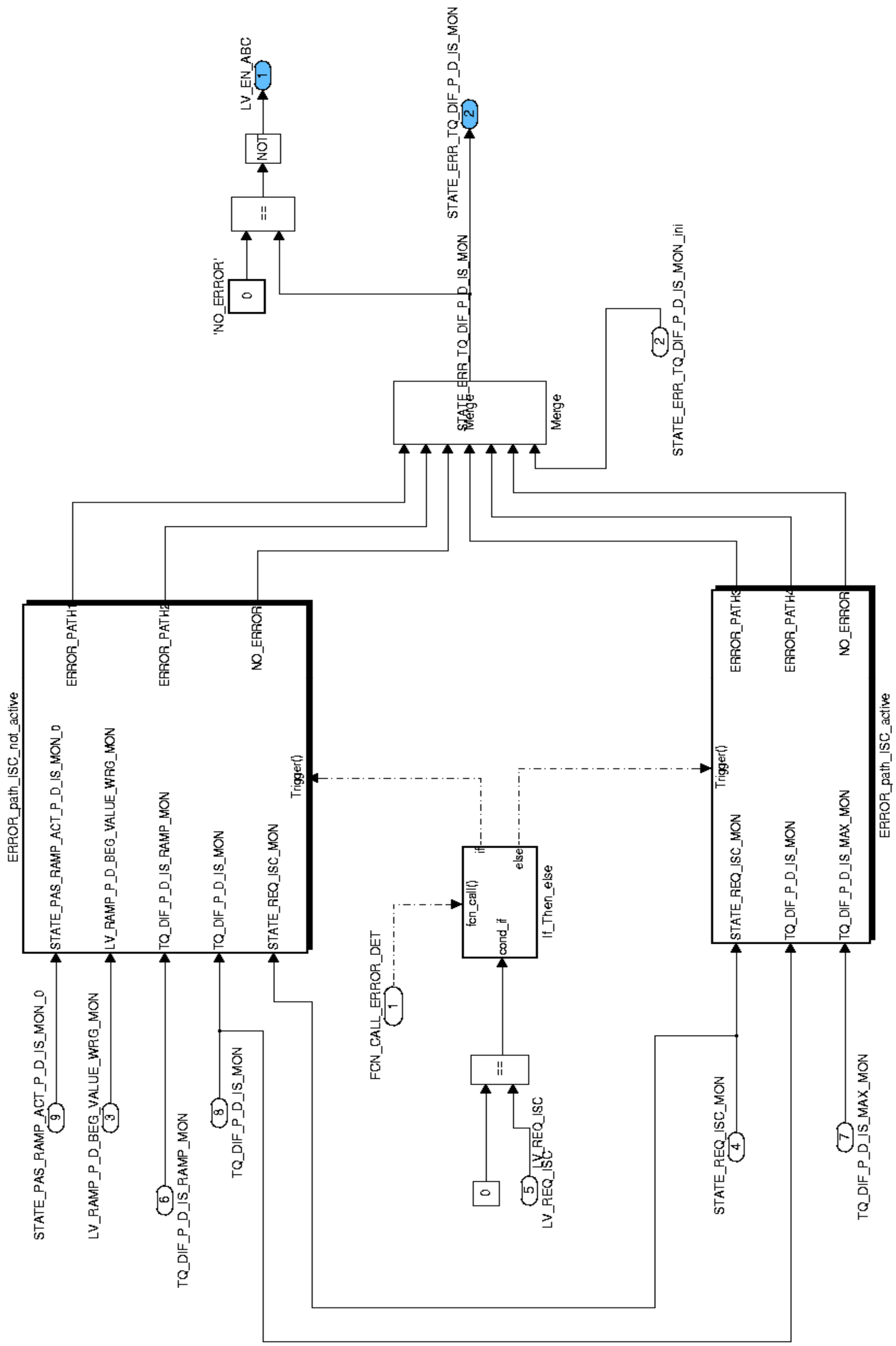


Figure 16 ECM2_DTSYSISC/ Monitoring_of_PD_part/ ERROR_detection

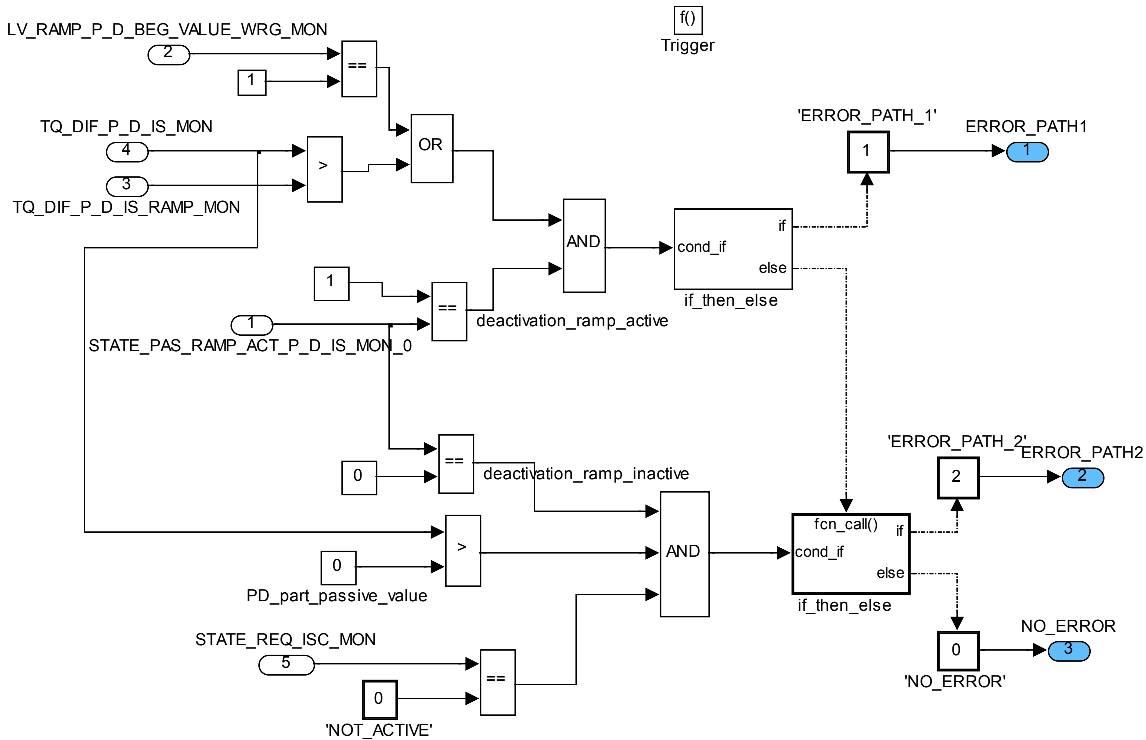


Figure 17 ECM2_DTSYSISC/ Monitoring_of_PD_part/ ERROR_detection/ ERROR_path_ISC_not_active

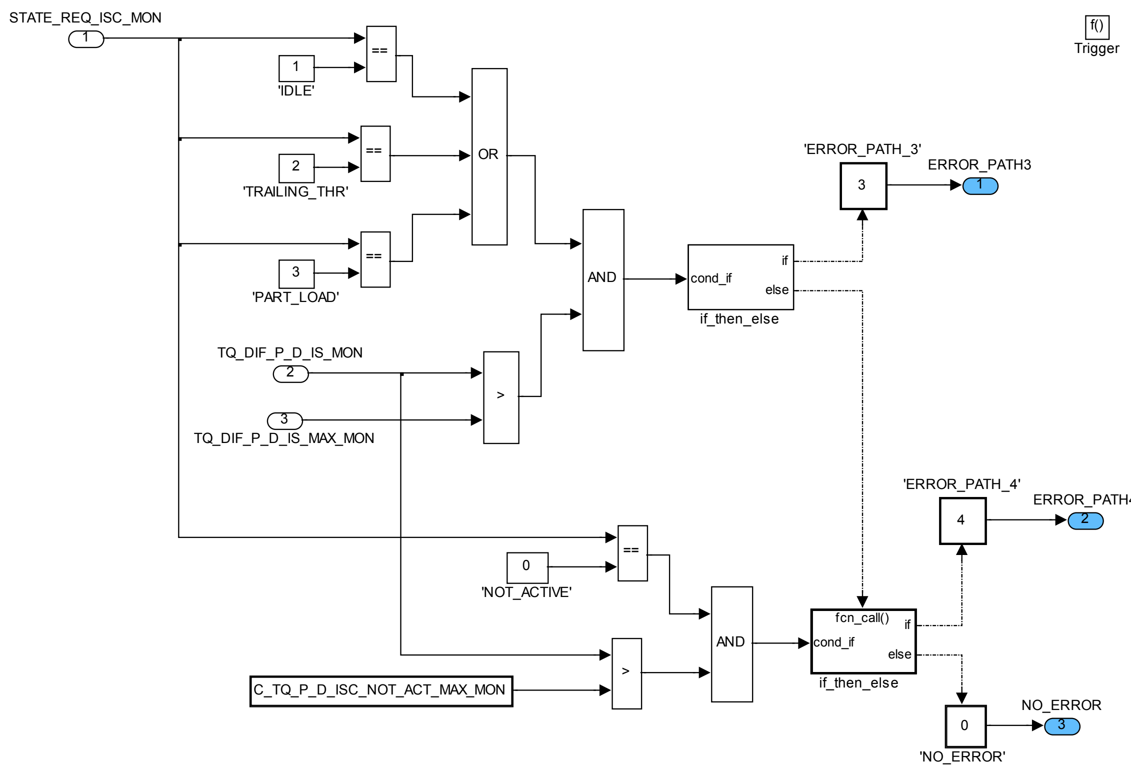


Figure 18 ECM2_DTSYSISC/ Monitoring_of_PD_part/ ERROR_detection/ ERROR_path_ISC_active

E.14.2 Monitoring of I controller

General information:

The objective of this function is the monitoring of the torque demand from the I-part of the idle speed controller function. Two different faults scenarios are considered:

Error Path – ISC not active:

Case 1: If the ISC is actually in the ramp limit operation due to deactivation of ISC, this ramp function is monitored. If LV_PAS_RAMP_ACT_I_CHG_MON is 1(i.e. indicates a change of STATE_PAS_RAMP_ACT_I_IS(0) from 0 to 1 or a flank change between two L2 sample times), a ramp decrement (TQ_ADD_I_IS_MON) is calculated and the actual I-part (TQ_DIF_I_IS_MON) plus a calibratable additive constant is copied to TQ_DIF_I_IS_RAMP_MON. Further on, TQ_DIF_I_IS_RAMP_MON is calculated by adding the calculated ramp decrement to the last delayed value. A fault is set, if TQ_DIF_I_IS_MON > TQ_DIF_I_IS_RAMP_MON or greater than the monitoring threshold C_TQ_DIF_I_IS_MAX_MON. If one of the errors is detected, a fault is declared after debouncing. The difference between monitored and modelled value is declared as a visible variable for application purposes.

Case 2: If in L2 idle speed controller is not active i.e. STATE_ REQ_ISC_MON = NOT_ ACTIVE but the I-part in the function level has not reached the passive value C_TQ_PAS_ I_IS_MON at the end the deactivation ramp, then an error is declared.

Error Path – ISC active:

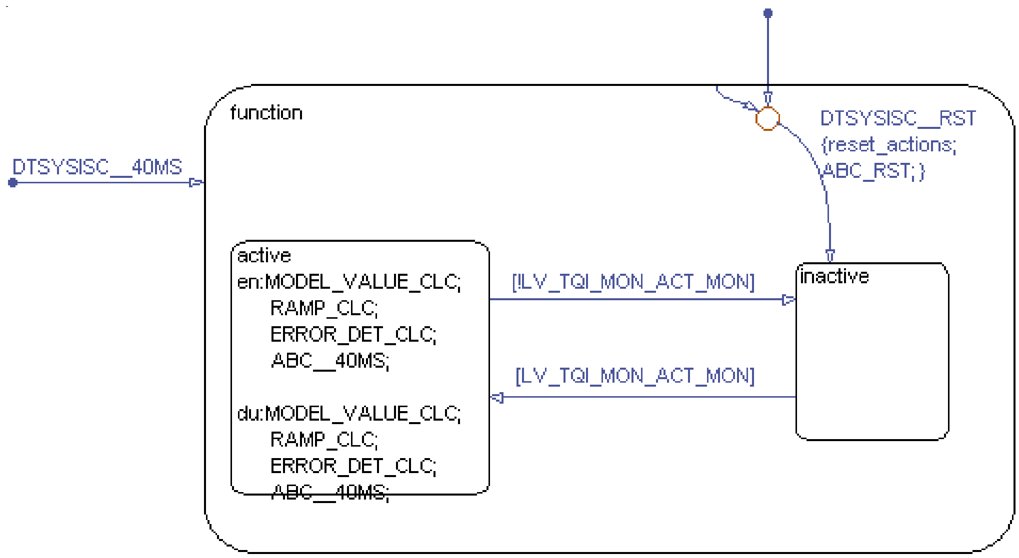
Case 1: The ISC can be active either when the engine is in idle state(LV_IS=1) or under certain conditions in trailing-throttle state (LV_PU=1) and part-load state (LV_PL=1).So all of these possible “ISC-active” phases have to be monitored. A fault is set, if L1 declares the ISC be active and if L2 also detects the same based on plausiblised activation conditions (STATE_ REQ_ISC_MON = IDLE or TRAILING_THR or PART_LOAD) and if torque demand fromL1, TQ_DIF_I_IS_MON, exceeds a threshold value C_TQ_DIF_I_IS_MAX _MON.

Case 2: If in L1 the ISC is active(LV_REQ_ISC=1) but the monitoring functionality detects this activation to be ‘faulty’ i.e. (STATE_ REQ_ISC_MON ≠ IDLE or TRAILING_THR or PART_LOAD), then an I-part torque demand from L1 greater than a threshold value (C_TQ_I_ISC_NOT _ACT_MAX_MON) will lead to an error detection.

If one of the errors above is detected, a fault is declared after debouncing by the idle speed controller montoring.

Note: The value of STATE_PAS_RAMP_ACT_I_IS(bit coded byte) and STATE_PAS_RAMP_ ACT_P_D_IS(bit coded byte) for the current recurrency (represented by STATE_PAS_RAMP_ACT_I_IS_0 and STATE_PAS_ RAMP_ACT_P_D_IS_0 respectively in the signal flow diagrams) are stored in the bit position 1.

Application Conditions



***Initialisation(DTSYSISC__RST)*:** for condition see 'Application Incidences of Process Monitoring'.

Function Description

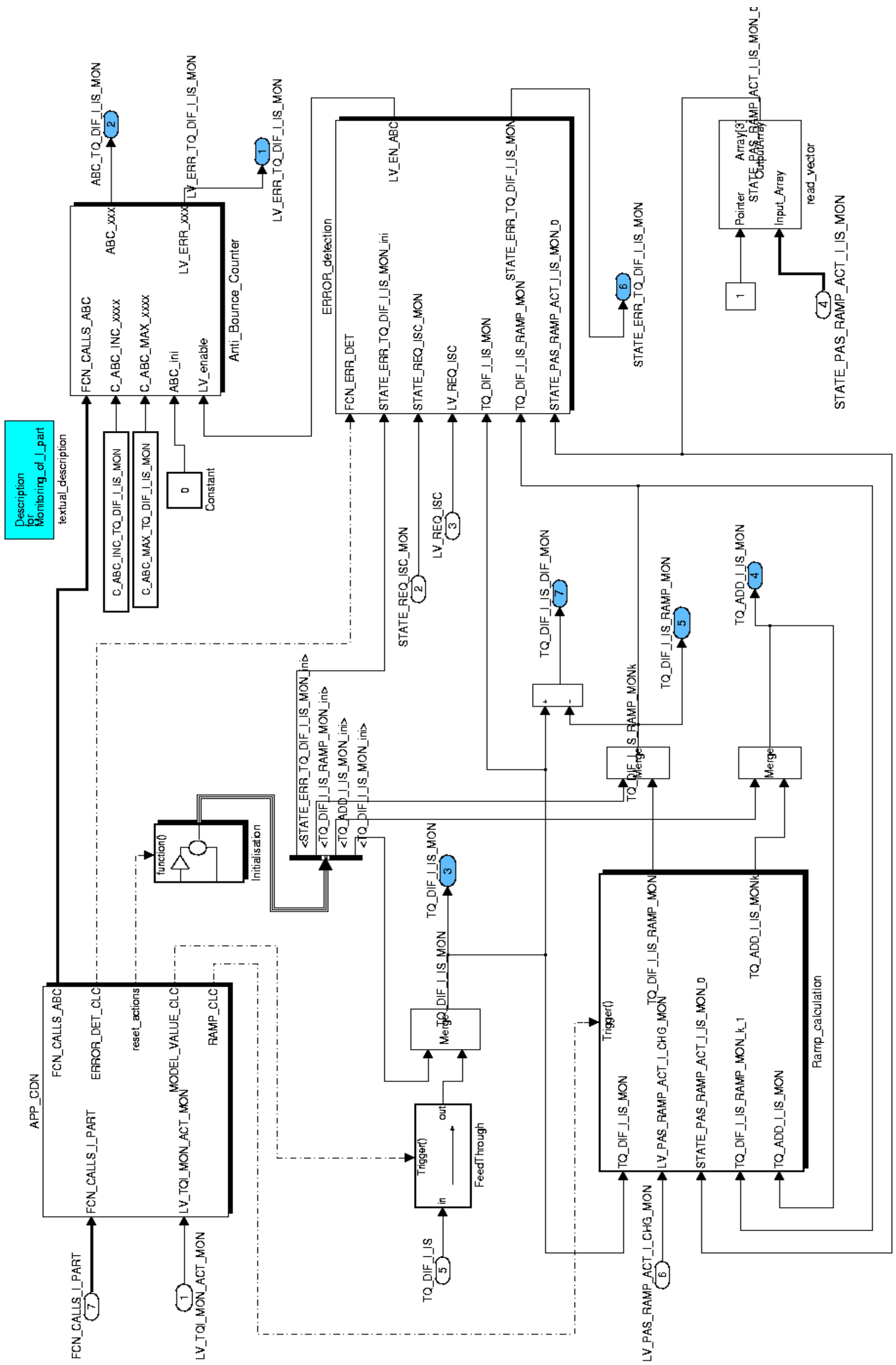


Figure 19 ECM2_DTSYSISC/ Monitoring_of_I_part

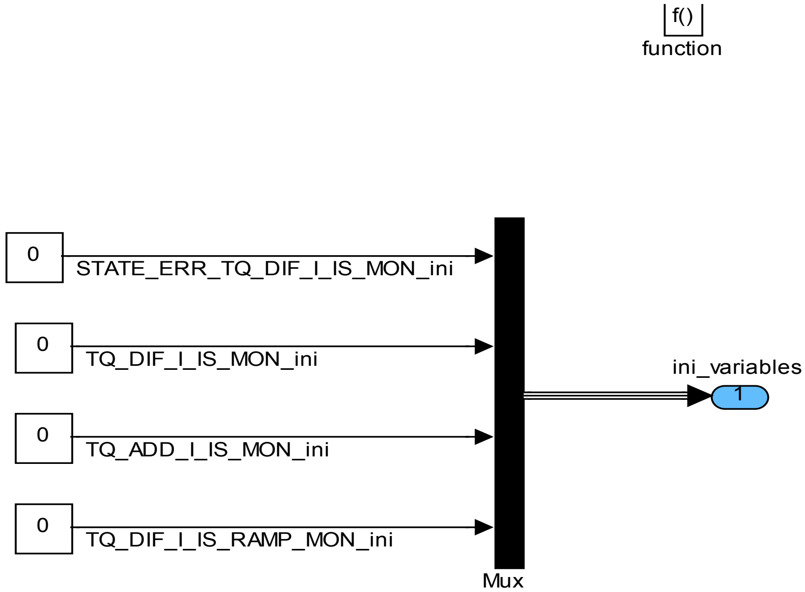


Figure 20 ECM2_DTSYSISC/ Monitoring_of_I_part/ Initialisation

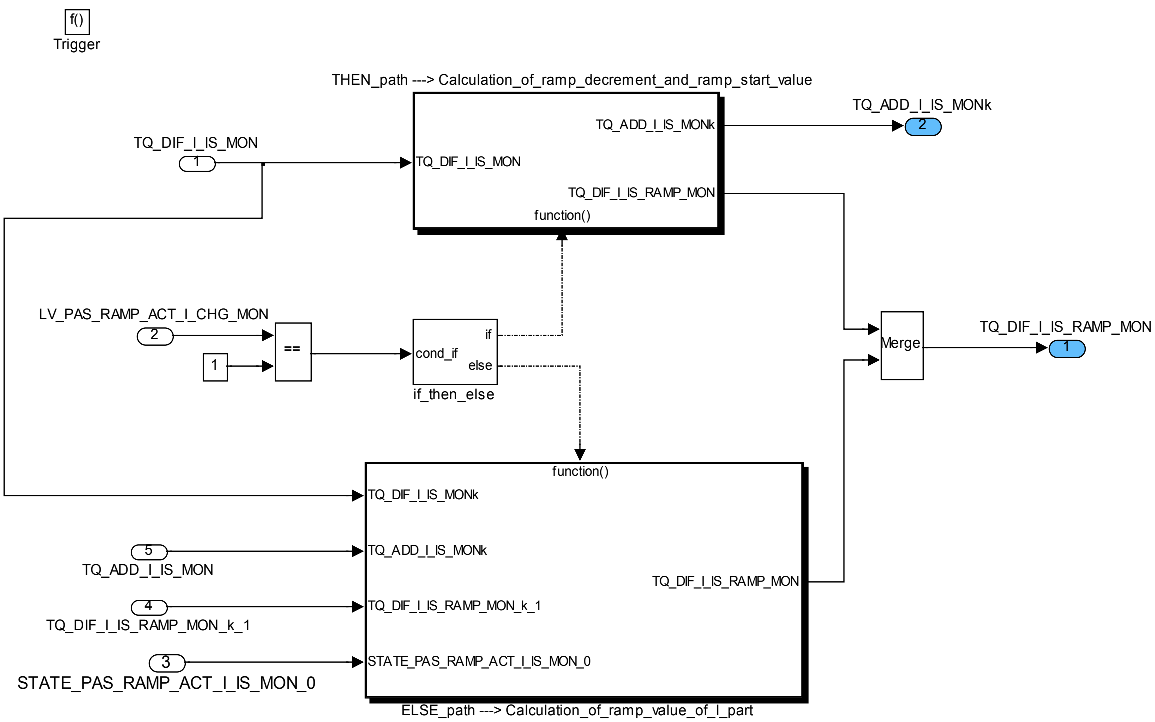


Figure 21 ECM2_DTSYSISC/ Monitoring_of_I_part/ Ramp_calculation

Note: The minimum I-ramp increment/decrement is equal to the resolution of TQ_ADD_I_IS_MON (0.03125Nm). If the theoretically calculated **absolute value** of TQ_ADD_I_IS_MON is ***smaller* than the resolution and not equal to zero**, then is to be limited to the resolution.This would lead to a faster ascent/descent of the deactivation ramp i.e. shorter ramp time. If the theoretical value of TQ_ADD_I_IS_MON is equal to zero, this value is then maintained.

For e.g: theoretical value = -0.04 Nm ---> Limited value = -0.03125 Nm
theoretical value = -0.01 Nm ---> Limited value = -0.03125 Nm
theoretical value = 0.04 Nm ---> Limited value = 0.0625 Nm
theoretical value = 0.01 Nm ---> Limited value = 0.03125 Nm

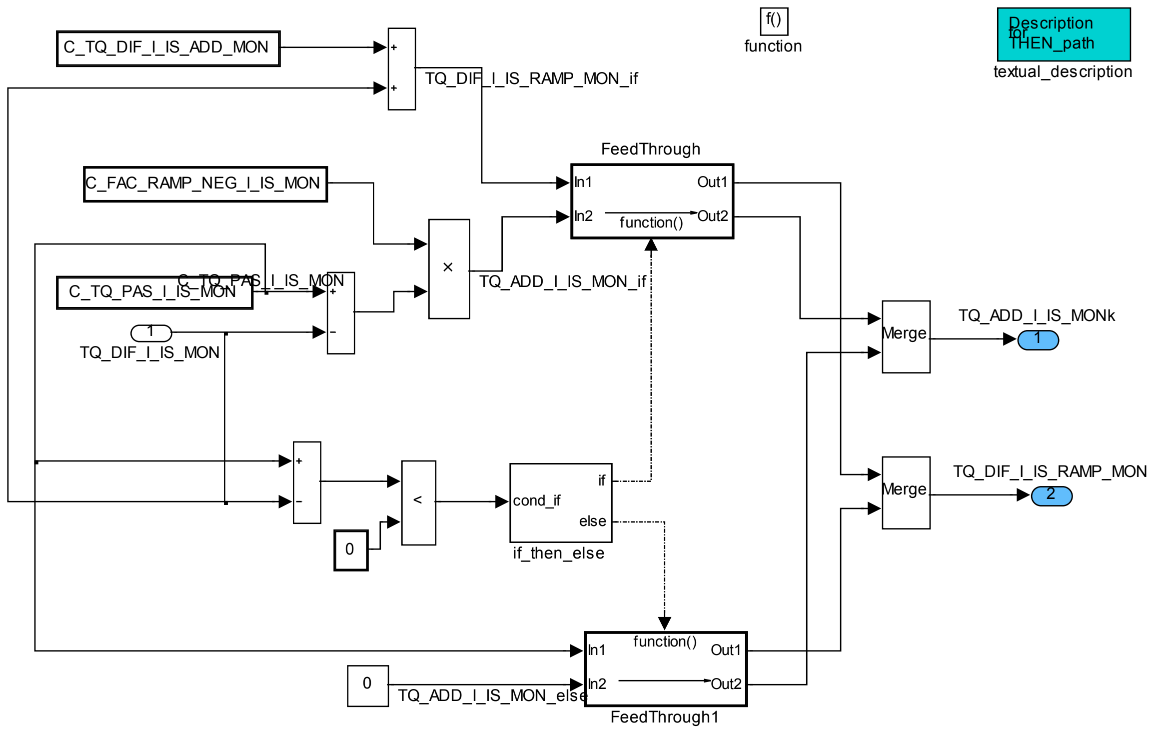


Figure 22 ECM2_DTSYSISC/ Monitoring_of_I_part/ Ramp_calculation/ THEN_path

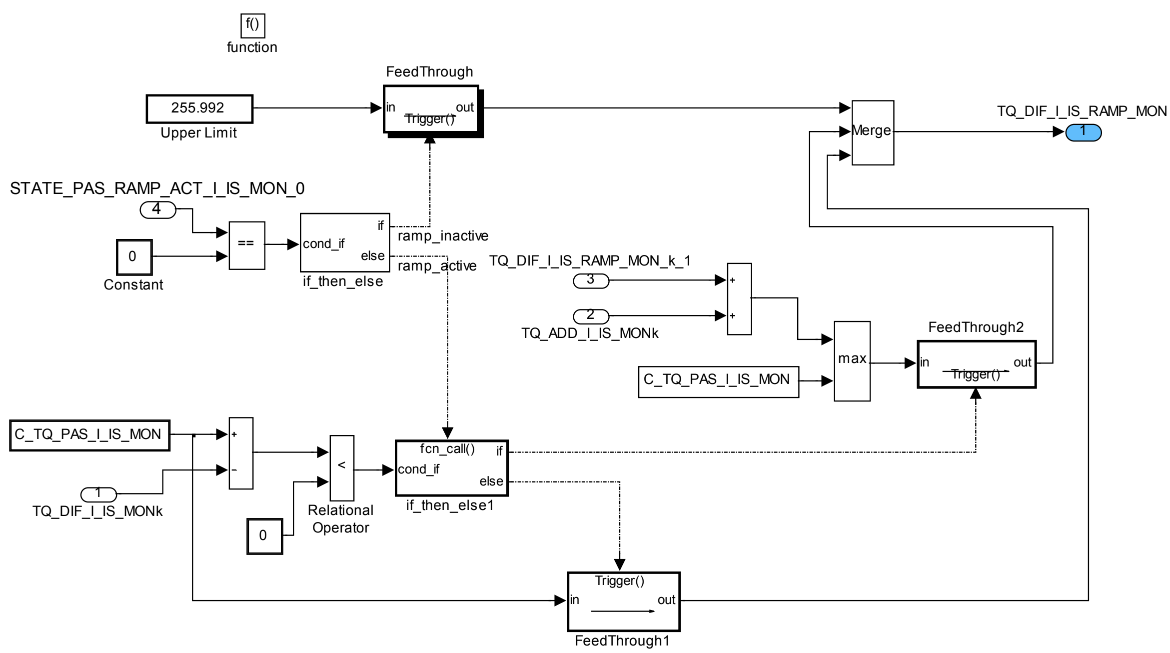
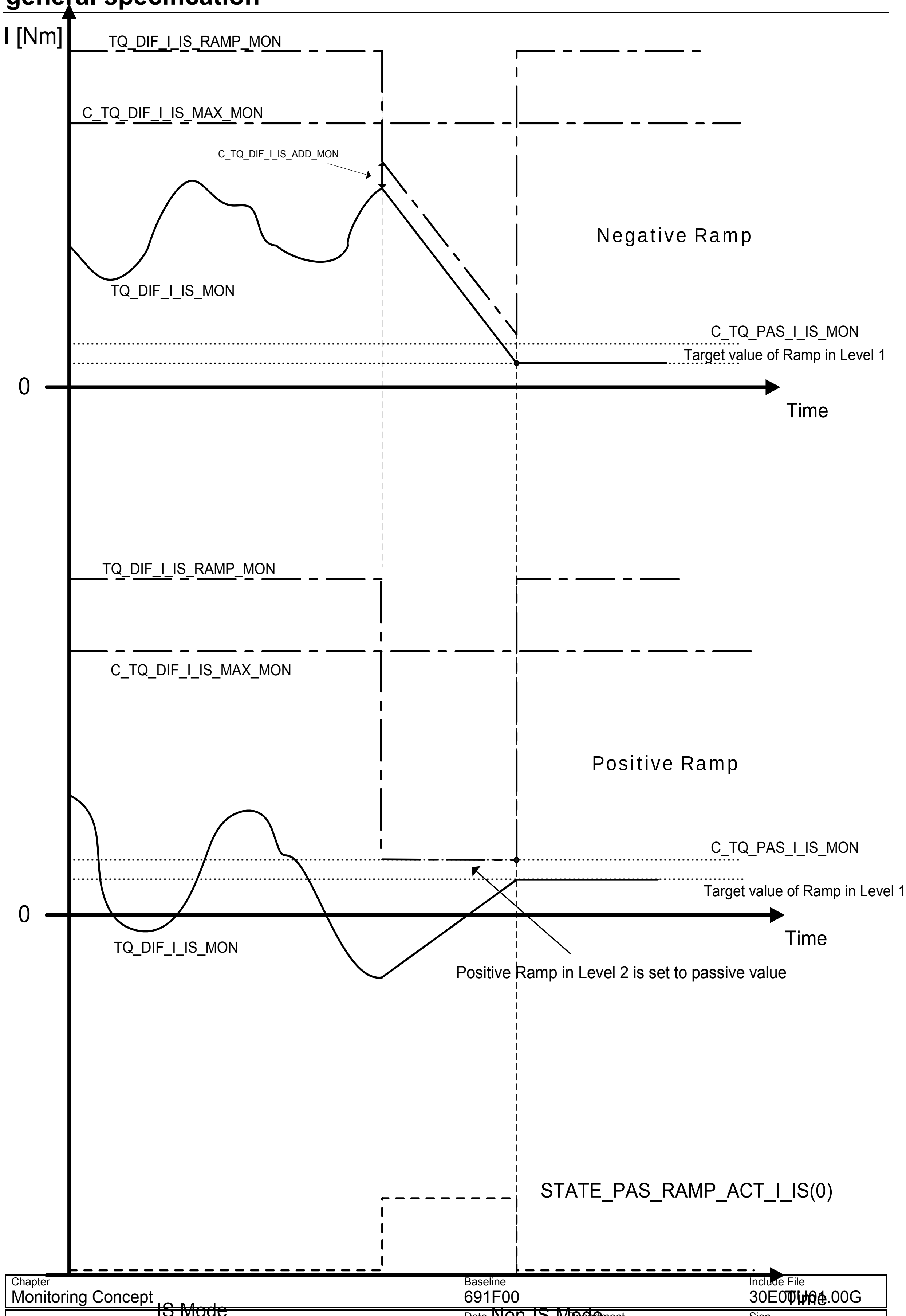


Figure 23 ECM2_DTSYSISC/ Monitoring_of_I_part/ Ramp_calculation/ ELSE_path


I [Nm]
TQ_DIF_I_IS_RAMP_MON
C_TQ_DIF_I_IS_MAX_MON
C_TQ_DIF_I_IS_ADD_MON
Negative Ramp
TQ_DIF_I_IS_MON
C_TQ_PAS_I_IS_MON
Target value of Ramp in Level 1
0
Time
TQ_DIF_I_IS_RAMP_MON
C_TQ_DIF_I_IS_MAX_MON
Positive Ramp
C_TQ_PAS_I_IS_MON
Target value of Ramp in Level 1
0
Time
TQ_DIF_I_IS_MON
Positive Ramp in Level 2 is set to passive value
STATE_PAS_RAMP_ACT_I_IS(0)
IS Mode
Non-IS Mode
Time

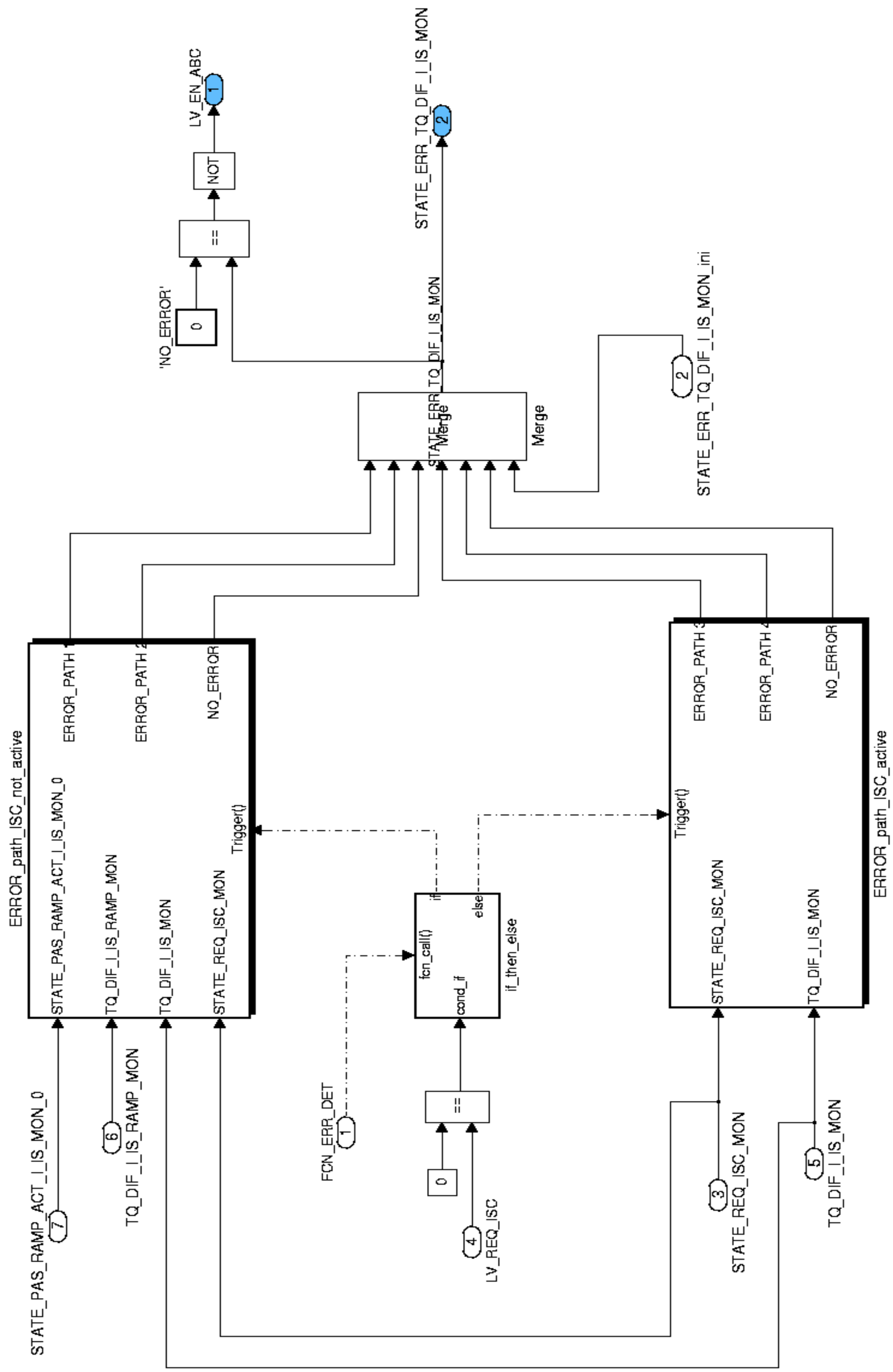


Figure 24 ECM2_DTSYSISC/ Monitoring_of_I_part/ ERROR_detection

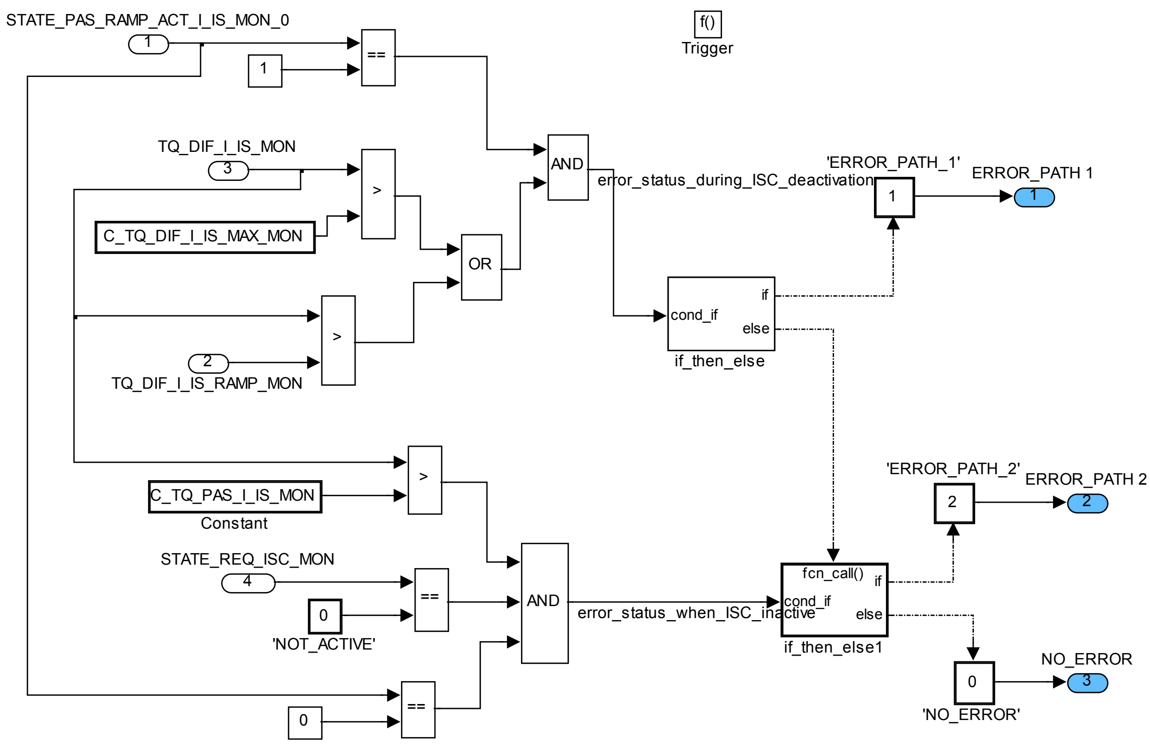


Figure 25 ECM2_DTSYSISC/ Monitoring_of_I_part/ ERROR_detection/ ERROR_path_ISC_not_active

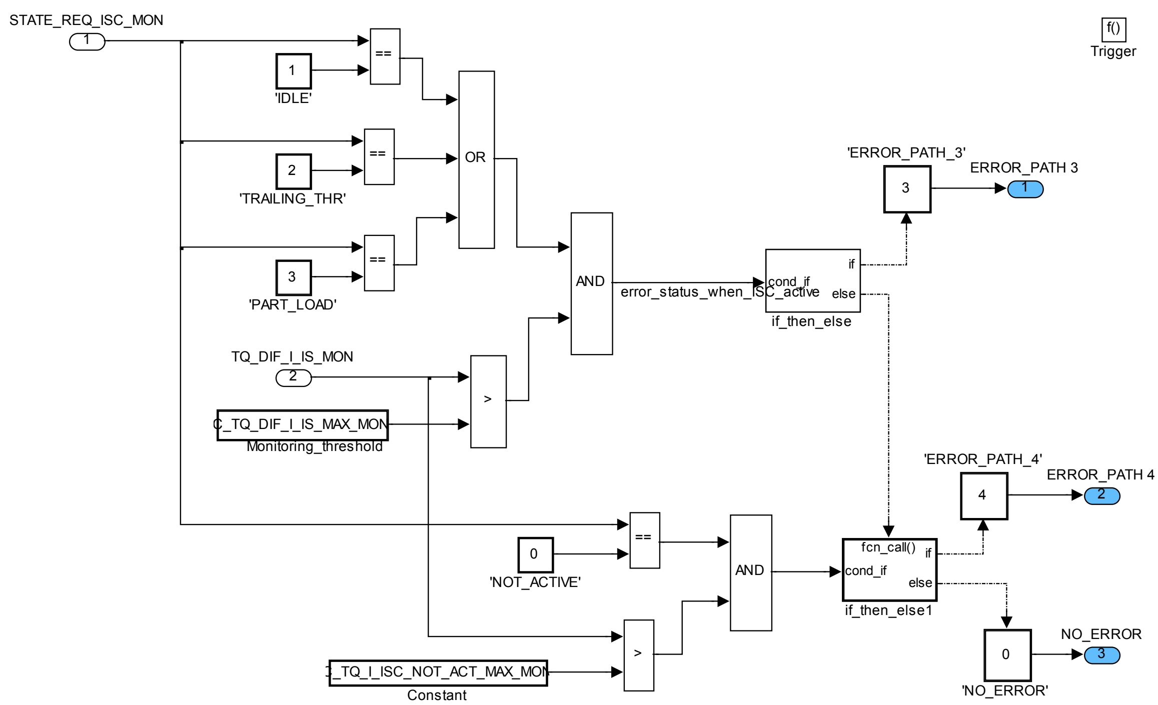


Figure 26 ECM2_DTSYSISC/ Monitoring_of_I_part/ ERROR_detection/ ERROR_path_ISC_active

E.15 Monitoring of torque losses

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQ_LOSS_MON	O/V	0...1H	0...1	1	-
Fault currently present in torque generation, symptom ‘torque losses’					
TQ_LOSS_MON	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque losses					
TQ_LOSS_REQ_CLU_MON	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Torque losses without pumping losses					
ABC_TQ_LOSS_MON	V	0...FFH	0...255	1	-
Anti bounce counter for monitoring of torque losses					
TQ_LOSS_DIF_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Difference (absolut) of monitored and modelled torque losses					
TQ_LOSS_MDL_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Modelled torque losses					
TQ_LOSS_MMV_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Filtered TQ_LOSS					

Input data:

LV_TQI_MON_ACT_MON	MAF_MON	N_32_MON	TCO_MON
TQ_DIF_IS_AD	TQ_LOSS	TQ_LOSS_ADD_MON	TQ_LOSS_REQ_CLU
TQ_LOSS_ACC_MON			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQ_LOSS_MON	1	0...FFH	0...255	1	-
anti bounce counter increment					
C_ABC_MAX_TQ_LOSS_MON	1	1...FFH	1...255	1	-
maximum of anti bounce counter					
C_TQ_LOSS_DIF_MAX_MON	1	80...7FH	-256...254	2	Nm
threshold for TQ_LOSS deviation					
C_CRLC_TQ_LOSS_REQ_CLU_MON	1	0...FFH	0...0.996	0.0039	-
Correlation constant for TQ_LOSS filtering					
C_TQ_IS_AD_MIN_MON	1	80...7FH	-256...254	2	Nm
Min. limitation of engine torque losses adaptation					
IP_TQFR_ADD_MON	1x5	0...FFH	0...510	2	Nm
LDP_TCO_MON_IP_TQFR_ADD_MON	5	0...FEH	-48...142.5	0.75	°C
Coolant temperature correction of friction torque					
IP_TQFR_MON	5x5	0...FFH	0...510	2	Nm
LDP_N_32_MON_IP_TQFR_MON	5	0...FFH	0...8160	32	rpm
LDP_MAF_MON_IP_TQFR_MON	5	0...FFH	0...1389	5.447	mg/stroke
Base friction torque and pumping losses					
LC_TQ_LOSS_REQ_CLU_WOUT_ACC_MON	1	0...1H	0...1	1	-
Switch to remove air conditioning load from driver request scaling (monitoring)					

E.15.1 ECM2_DTSYSTQLO

Import actions:

ACTION_ECM3_Service6TaskPfm(IN<n>)
ACTION_ECM3_Service7TaskPfm(IN<n>)

Note: This actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument n has the value 2.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

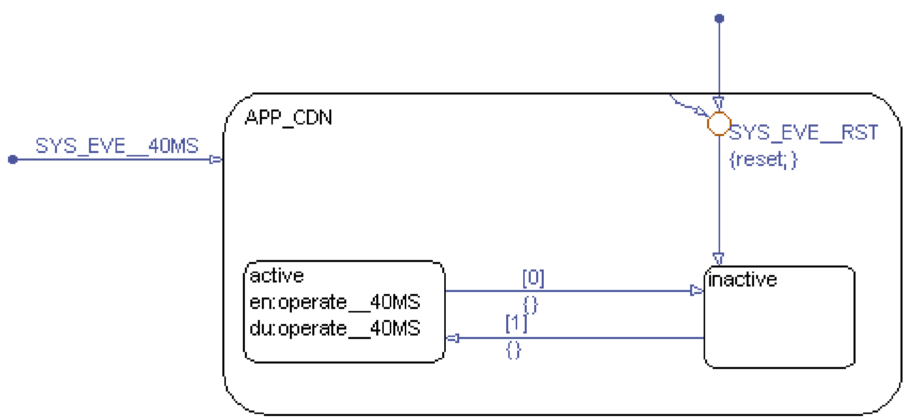
General information:

The objective of this function is the monitoring of the torque losses. It observes both level 1 values, TQ_LOSS and TQ_LOSS_REQ_CLU (filtered TQ_LOSS without pumping losses), that is exclusively used for the determination of TQ_MIN_CLU and TQ_MAX_CLU.

It has to be distinguished between engine immanent torque losses (e.g. TQFR) and not engine immanent torque losses (e.g. TQ_LOSS_ACC), that are determined in the module "Appl. Inc. and configuration of process monitoring".

A fault is detected after debouncing, if the monitored torque losses are smaller than the modelled ones or the difference between TQ_LOSS and TQ_LOSS_REQ_CLU is greater than a calibratable threshold (maximum pumping losses).

Application Condition



Function Description

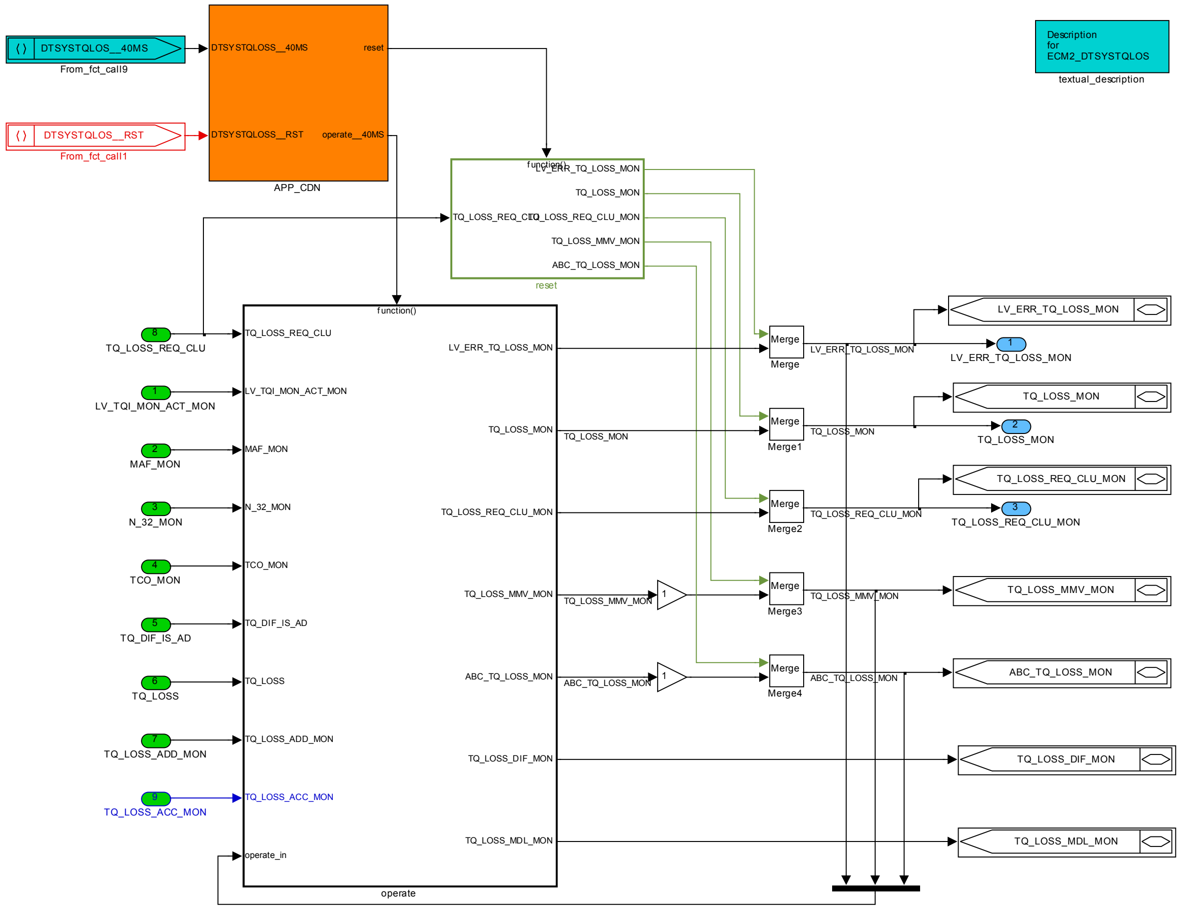


Figure 27 ECM2_DTSYSTQLOS

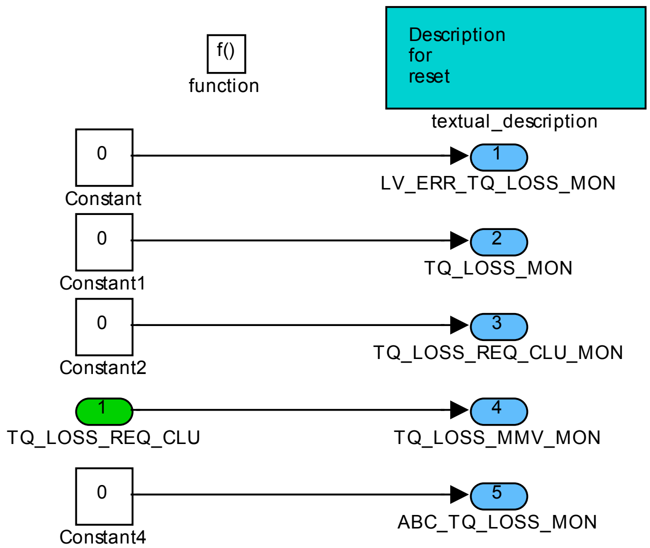


Figure 28 ECM2_DTSYSTQLOS/ reset

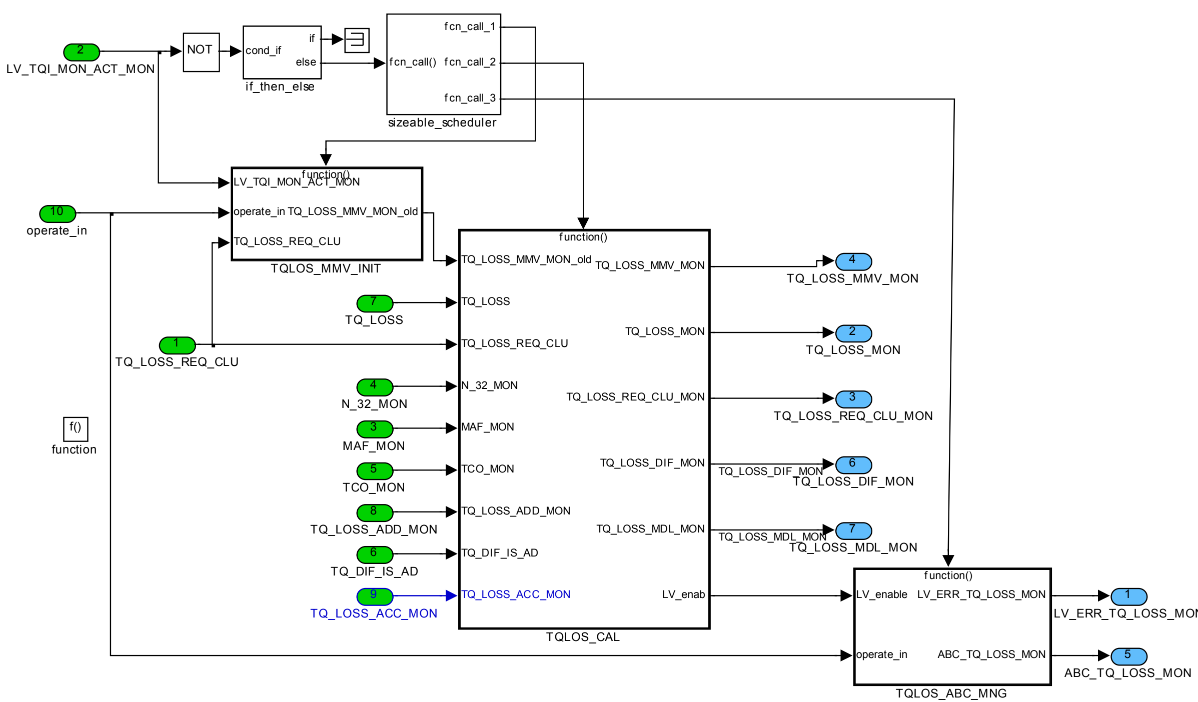


Figure 29 ECM2_DTSYSTQLOS/ operate

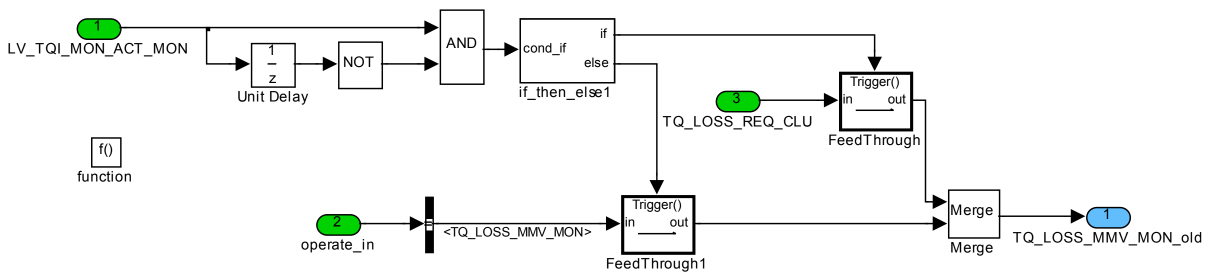


Figure 30 ECM2_DTSYSTQLOS/ operate/ TQLOS_MMV_INIT

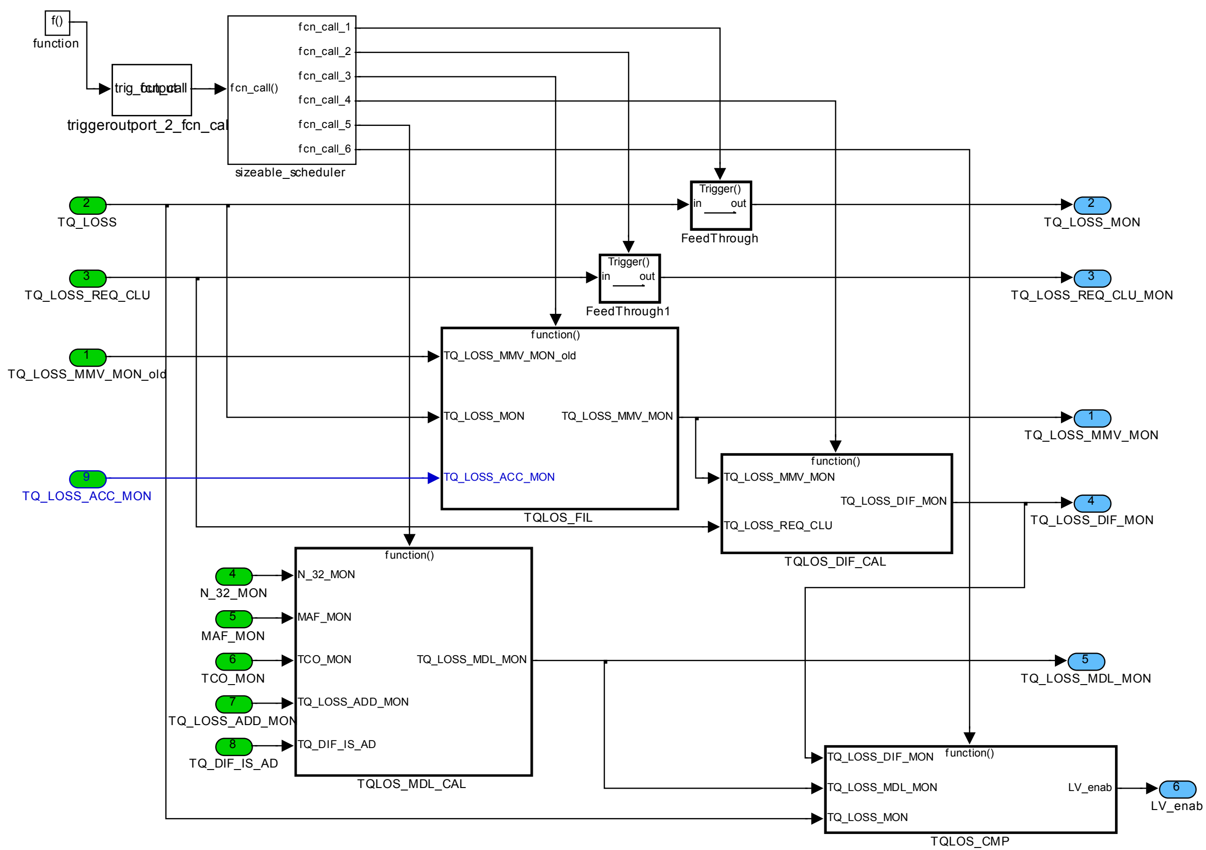


Figure 31 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL

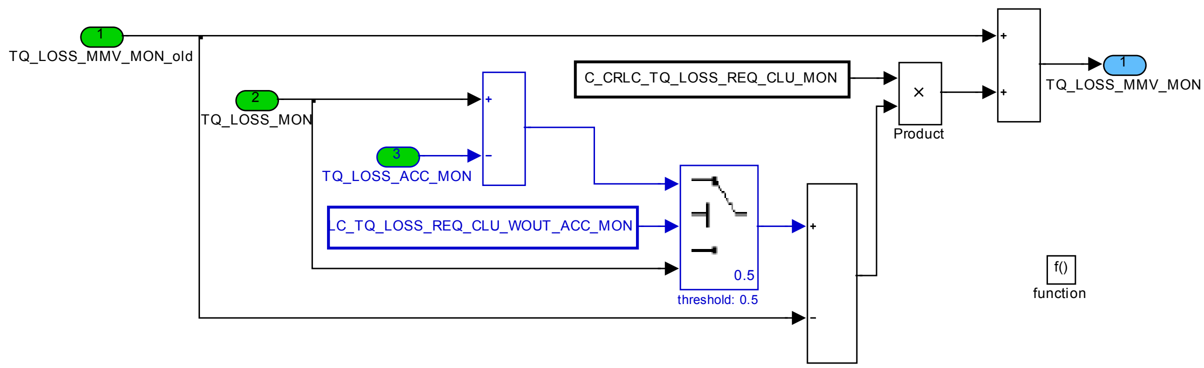


Figure 32 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL/ TQLOS_FIL

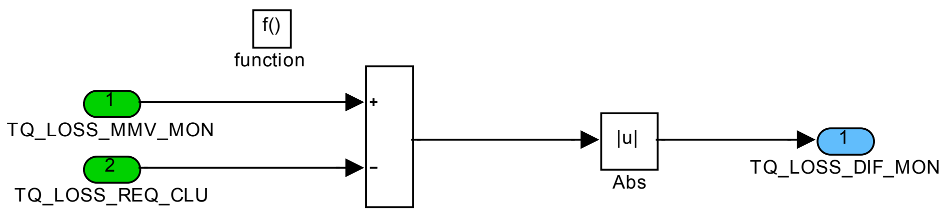


Figure 33 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL/ TQLOS_DIF_CAL

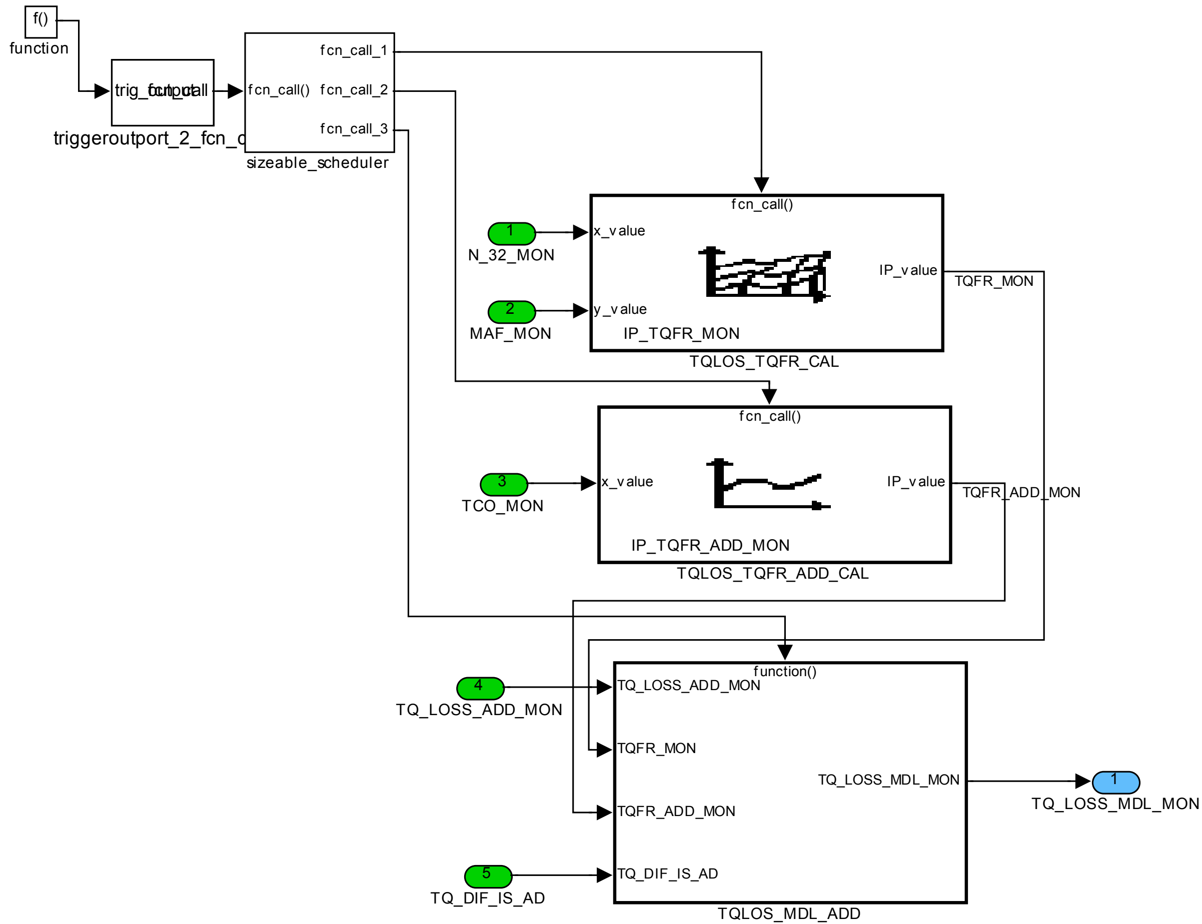


Figure 34 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL/ TQLOS_MDL_CAL

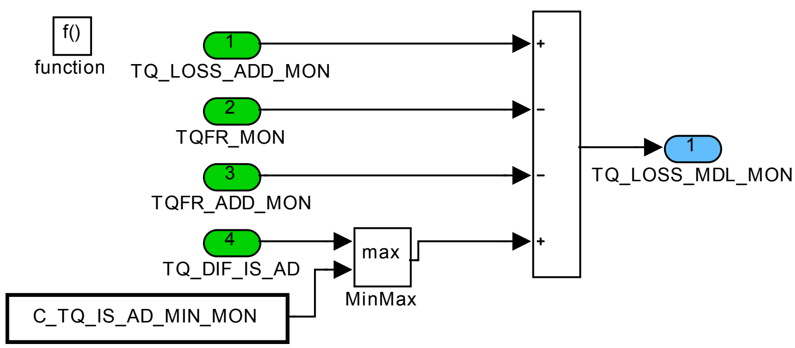


Figure 35 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL/ TQLOS_MDL_CAL/ TQLOS_MDL_ADD

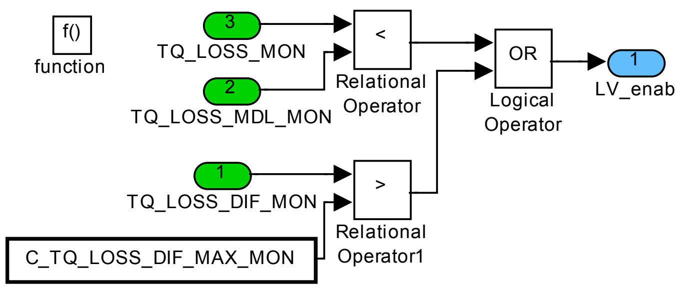


Figure 36 ECM2_DTSYSTQLOS/ operate/ TQLOS_CAL/ TQLOS_CMP

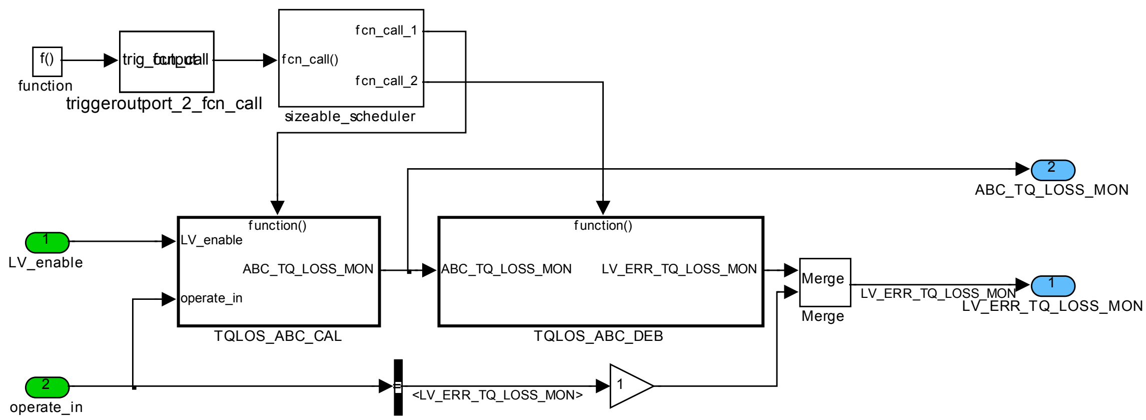


Figure 37 ECM2_DTSYSTQLOS/ operate/ TQLOS_ABC_MNG

E.16 Monitoring of minimum torque at clutch

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQ_MIN_CLU_MON	O/V	0...1H	0...1	1	-
Fault currently present in torque generation, symptom ‘minimum torque at clutch’					
TQ_MIN_CLU_MON	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
monitored minimum torque at clutch					
ABC_TQ_MIN_CLU_MON	V	0...FFH	0...255	1	-
Anti bounce counter for monitoring of minimum torque at clutch					
TQ_ST_MON	V	0...7FFFH	0...1023.97	0.03125	Nm
modelled value of start torque					
CTR_DLY_TQ_ST_MON	V	0...FFH	0...10.2	0.04	Sec
Delay time counter for holding start torque influence					
FAC_N_SP_IS_RATIO_MON	V	8000...7FFFH	-8...7.999	0.000244	-
Model of control factor for engine self-stabilising effect					
STATE_ERR_TQ_MIN_CLU_MON	V	0H 1H 2H	NO_ERROR ERROR_PATH1 ERROR_PATH2	1	-
State variable indicating origin of error resulting from monitoring of L1 torque value					
TQ_MIN_CLU_DIF_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
difference of monitored and modelled minimum torque at clutch					
TQ_MIN_CLU_MDL_1_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Temporary variable 1 for calculation of modelled minimum torque at clutch					
TQ_MIN_CLU_MDL_2_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Temporary variable 2 for calculation of modelled minimum torque at clutch					
TQ_MIN_CLU_MDL_4_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Temporary variable 3 for calculation of modelled minimum torque at clutch					
TQ_MIN_CLU_MDL_5_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Temporary variable 4 for calculation of modelled minimum torque at clutch					

Input data:

LV_SWI_TQ_MIN_CLU_MON	N_SP_IS_RATIO_MON	STATE_ERR_DET_TQ_MIN_MON	TQ_ADD_MIN_CLU_MON
TQ_CONV_MAX_MON	TQ_DIF_I_IS_MON	LV_TQI_MON_ACT_MON	LV_TQ_MIN_CLU
N_32_MON	TCO_MON	TQ_LOSS_MON	TQ_LOSS_REQ_CLU_MON
TQ_MIN_CLU			

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQ_MIN_CLU_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQ_MIN_CLU_MON	1	1...FFH	1...255	1	-
Maximum of anti bounce counter					
C_N_ST_MAX_MON	1	0...FFH	0...8160	32	rpm
maximum start end speed					
C_DLY_TQ_ST_MON	1	0...FFH	0...10.2	0.04	Sec
Time delay to hold influence of start torque on TQ_MIN_CLU model value					
C_TQ_MIN_CLU_MDL_OFS_MON	1	8000...7FFFH	-1024...1023.97	0.03125	Nm
Offset for L2 model value of minimum torque at clutch					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_N_SP_IS_RATIO_MON	5	0...FFFFH	-8...7.999	0.000244	-
LDP_N_SP_IS_RATIO_MON_IP_MON	5	0...FFFFH	0...8	0.000122	-
Idle self stabilising factor					
IP_TQI_MIN_REQ_PU_MON	6x4	0...7FFFH	0...1023.97	0.03125	Nm
LDP_N_32_MON_IP_TQI_MIN_MON	6	0...FFH	0...8160	32	rpm
LDP_TCO_MON_IP_TQI_MIN_PU_MON	4	0...FEH	-48...142.5	0.75	°C
Minimum requested indicated torque in PU phase					

E.16.1 FUNCTION PART: ECM2_DTSYSTQMIN

Import actions:

ACTION_ECM3_Service8TaskPfm(3)
ACTION_ECM3_Service9TaskPfm(3)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 3.

ACTION_ECM3_WriteChkCpl(IN<type>,INOUT<xyz_MON>, OUT<xyz_MON_CPL>, IN reg>)
ACTION_ECM3_ChkCpl(IN<type>, IN<xyz_MON>, IN<xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN<type>, OUT <reg>, IN<xyz_MON>, IN<xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

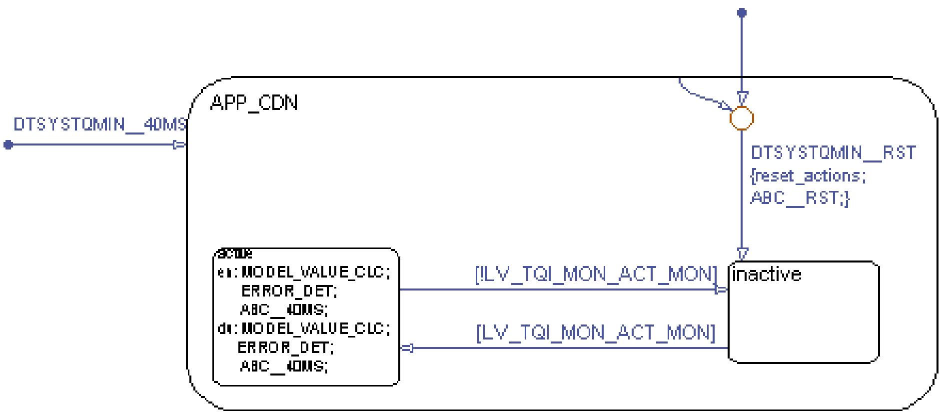
General information:

The objective of this function is the monitoring of the minimum torque at clutch. The model takes into account that a great part of the minimum torque at clutch could be the integral part of the ISC. Therefore the difference of monitored minimum torque at clutch and I-part of ISC is compared to a torque *threshold* which is dependent on:**1)** the engine self stabilising factor (function of ratio between engine speed and idle speed setpoint) **2)** TQ_LOSS_REQ_CLU_MON and **3)** TQ_CONV_MAX_MON. For engine speeds above idle speed setpoint this difference must be negative. For speeds below idle speed setpoint great values for minimum torque at clutch are allowed.

One exception of the above mentioned rule is at engine start. If the engine is in start phase, an constant additive term is taken into account for the threshold (see also "Chapter D – Minimum Torque at Clutch"). To monitor this start torque term, the actual engine speed is compared to an upper threshold for the engine start end-speed. In addition to the I-part of ISC, the two other factors influencing the minimum torque at clutch are hydraulic torque losses inside the converter of an automatic gearbox - TQ_CONV_MAX_MON (which depends on coolant temperature and idle speed setpoint) and the minimum requested indicated torque in the PU phase - TQI_MIN_REQ_PU_MON. However, if the engine is in PU or PUC mode TQ_MIN_ CLU_MDL is set to minimum possible torque that the driver can request i.e. TQ_LOSS_MON.

The plausiblisation of the L1 value, TQ_MIN_CLU, against a L2 model value is done based on current engine state. The variable STATE_ERR_DET_TQ_MIN_MON (calculated in the module 'Application incidences of process monitoring') is used to switch between the different error-detection paths. If the engine is in PU or PUC and the L1 minimum torque at clutch value is greater than TQ_LOSS_MON, then an error state is set. If however the engine is neither in PU or PUC, then the L1 value is compared to the L2 model value TQ_MIN_CLU_MDL_4_MON. If any of the error states are set, a fault is detected after debouncing.

Application Conditions



***Initialisation(DTSYSTQMIN__RST)*:** for condition see 'Application Incidences of Process Monitoring'

Function Description

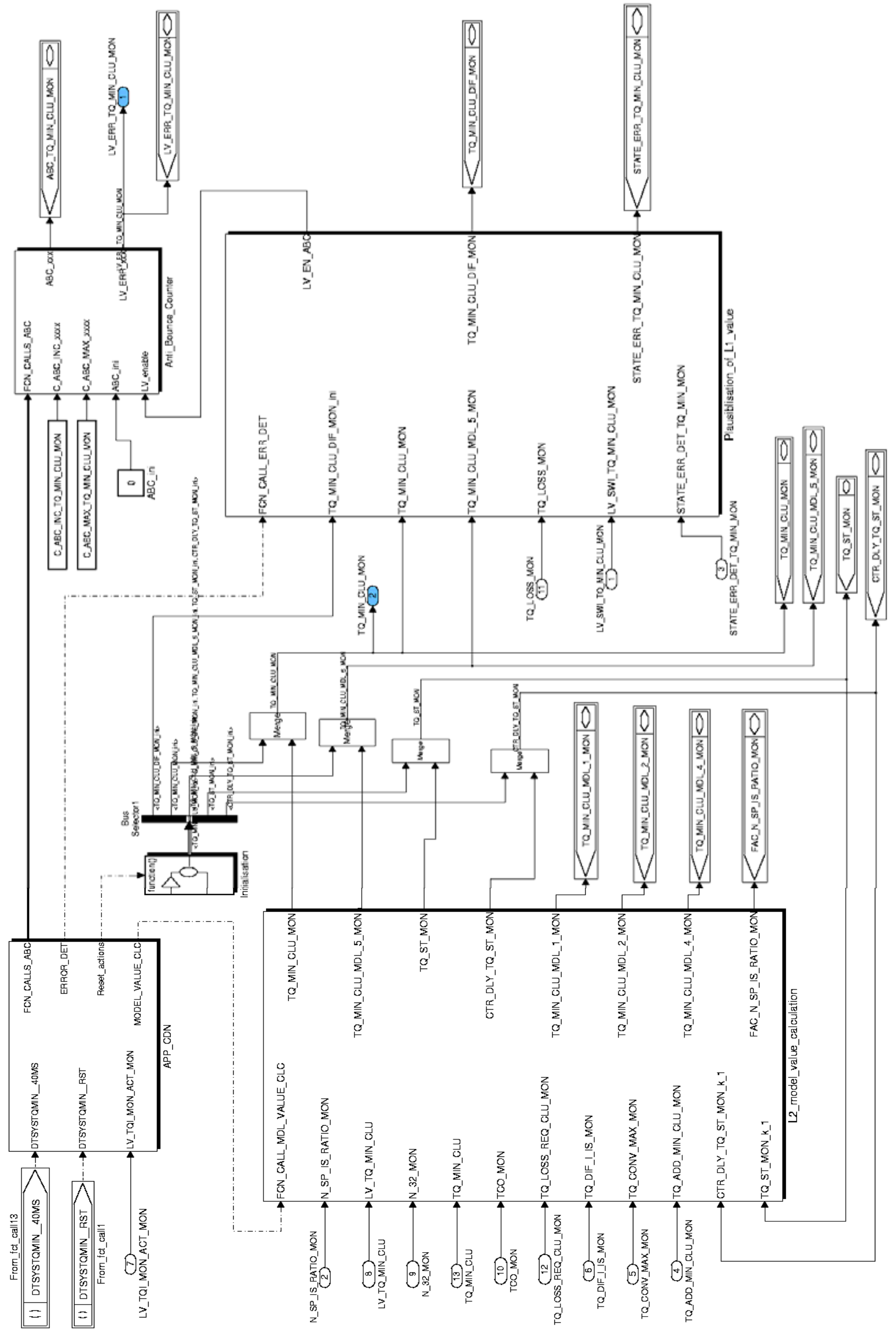


Figure 38 ECM2_DTSYSTQMIN

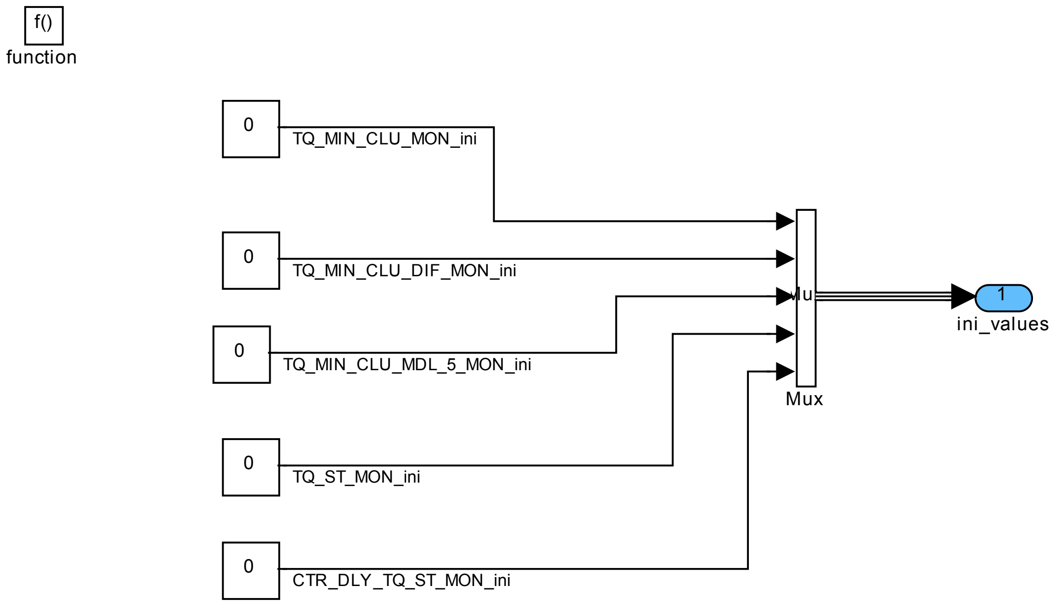


Figure 39 ECM2_DTSYSTQMIN/ INITIALISATION

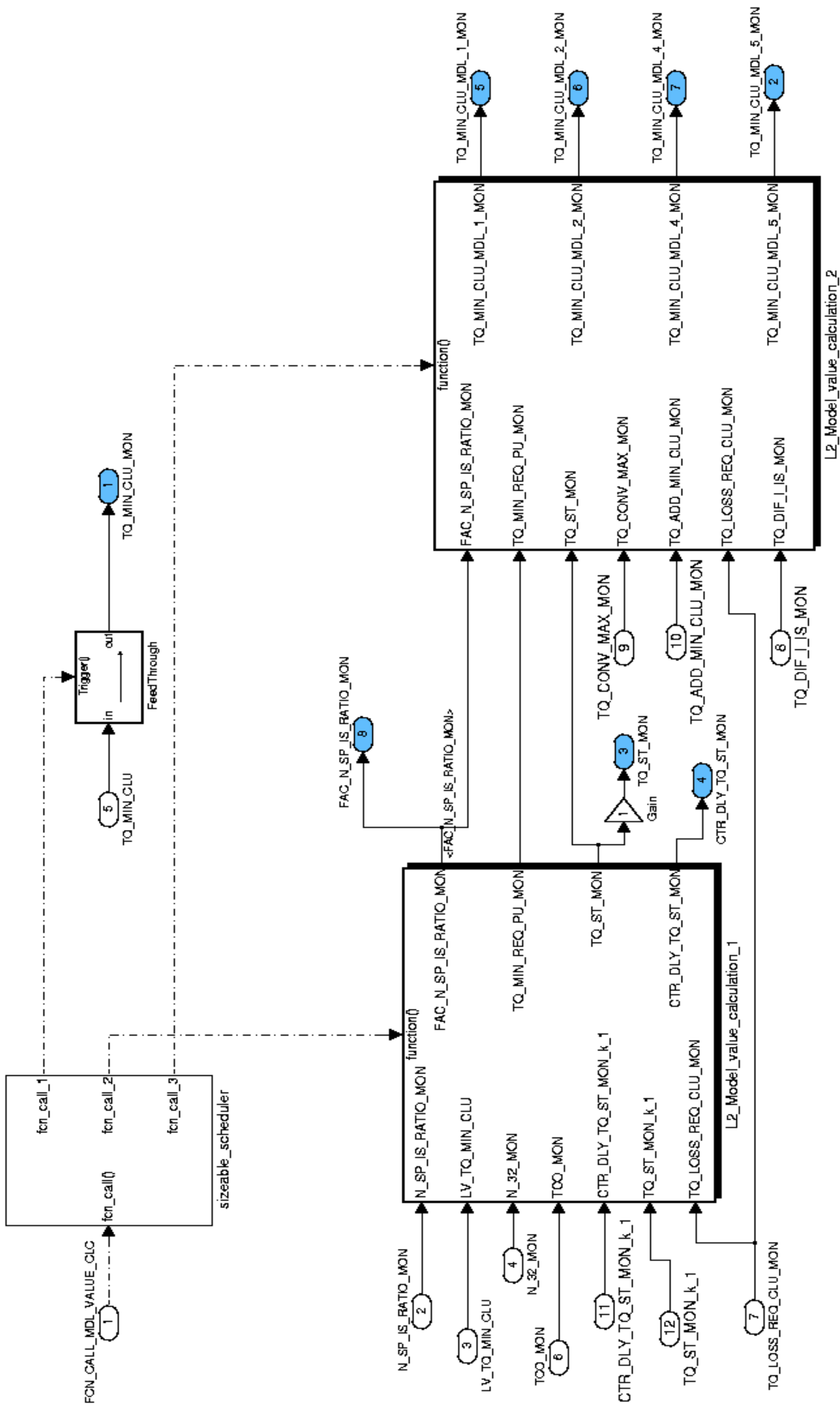


Figure 40 ECM2_DTSYSTQMIN/ L2_MODEL_VALUE_CALCULATION

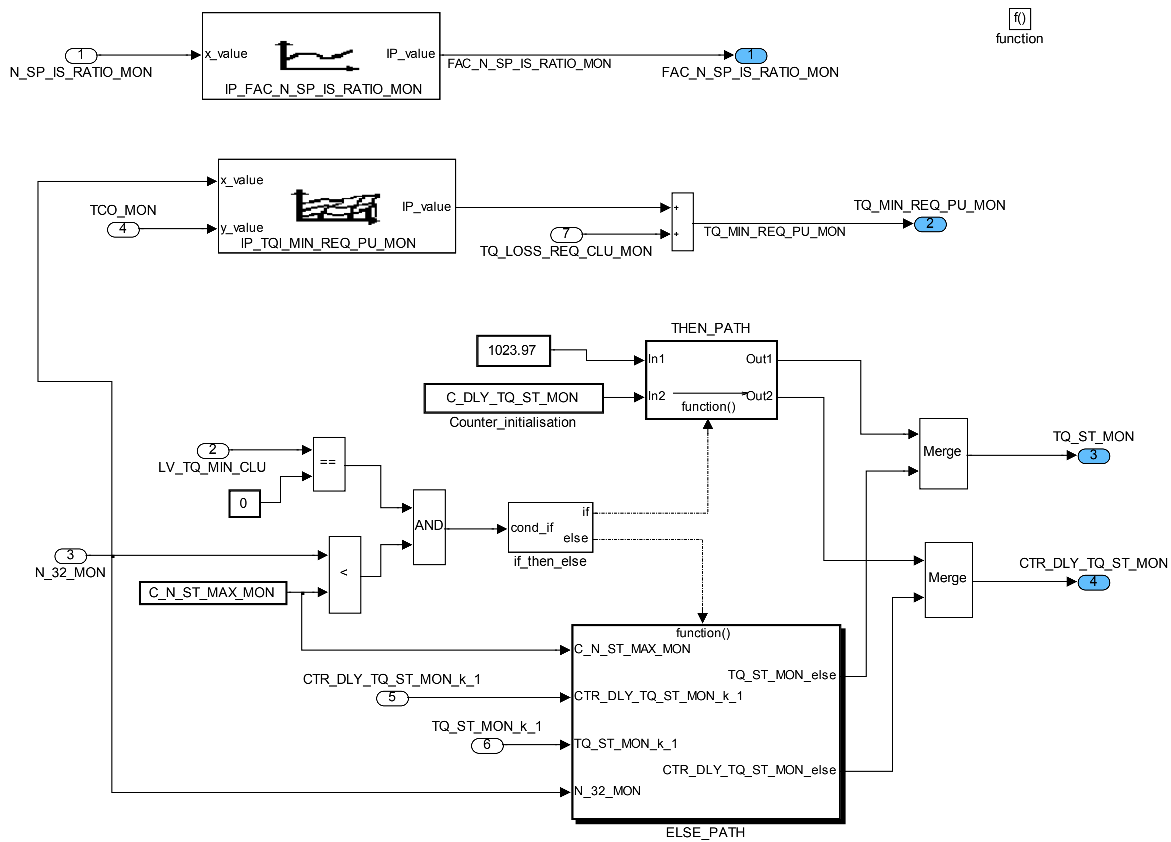


Figure 41 ECM2_DTSYSTQMIN/ L2_MODEL_VALUE_CALCULATION/ L2_MODEL_VALUE_CALCULATION_1

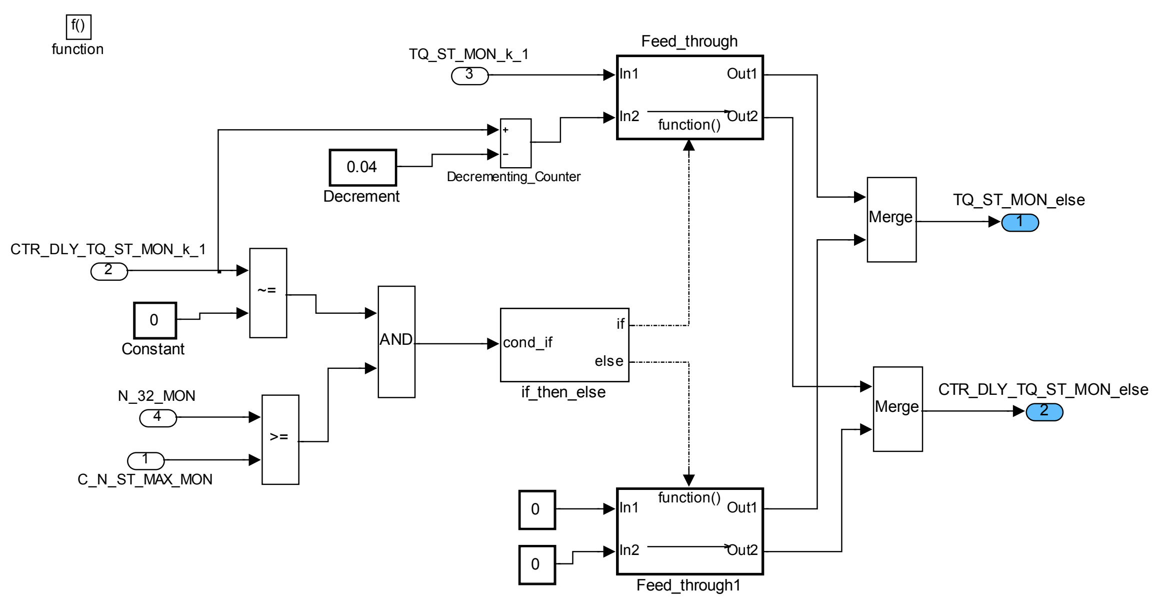


Figure 42 ECM2_DTSYSTQMIN/ L2_model_value_calculation/ L2_Model_value_calculation_1/ ELSE_PATH

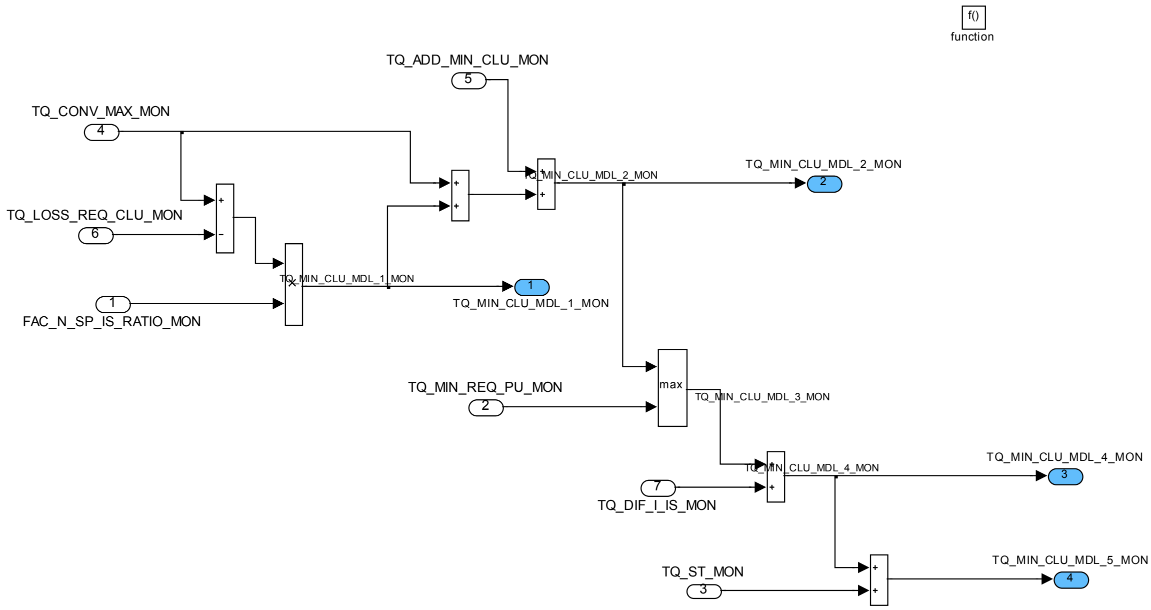


Figure 43 ECM2_DTSYSTQMIN/ L2_model_value_calculation/ L2_Model_value_calculation_2

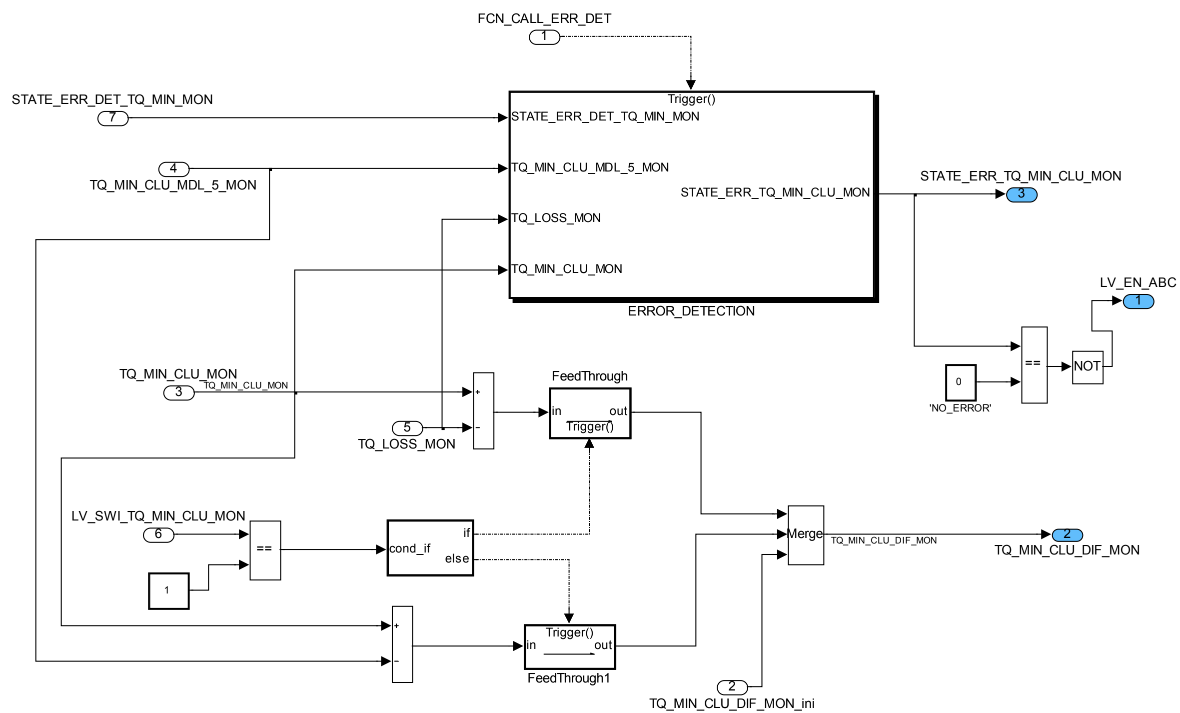


Figure 44 ECM2_DTSYSTQMIN/ PLAUSIBLISATION_OF_L1_VALUE

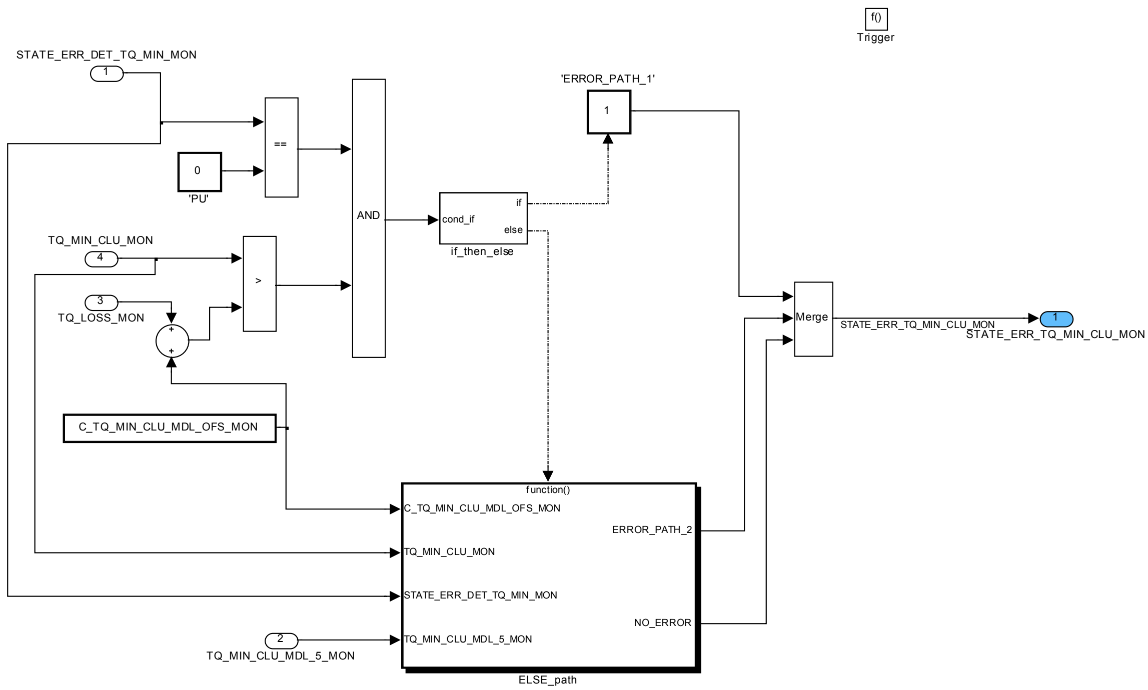


Figure 45 ECM2_DTSYSTQMIN/ PLAUSIBLISATION_OF_L1_VALUE/ ERROR_DETECTION

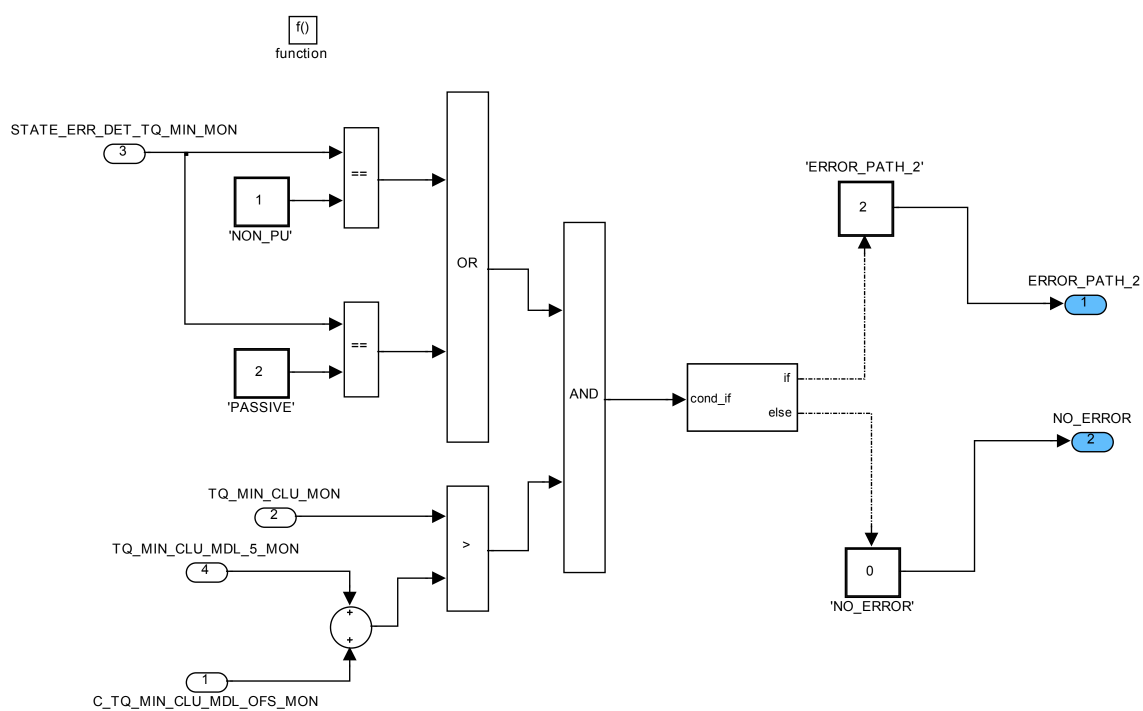


Figure 46 ECM2_DTSYSTQMIN/ Plausiblisation_of_L1_value/ ERROR_DETECTION/ ELSE_path

E.17 Monitoring of maximum torque at clutch

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQ_MAX_CLU_MON	O/V	0...1H	0...1	1	-
Fault currently present in torque generation, symptom 'maximum torque at clutch'					
TQ_MAX_CLU_MON	O/V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Monitored maximum torque at clutch					
ABC_TQ_MAX_CLU_MON	V	0...FFH	0...255	1	-
Anti bounce counter for monitoring of maximum torque at clutch					
TQ_MAX_CLU_DIF_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Difference between monitored and modelled maximum torque at clutch					
TQ_MAX_CLU_MAX_MON	V	8000...7FFFH	-1024...1023.97	0.03125	Nm
Model value for maximum torque at clutch					

Input data:

LV_TQI_MON_ACT_MON	N_32_MON	TQ_LOSS_REQ_CLU_MON	TQ_MAX_CLU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQ_MAX_CLU_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQ_MAX_CLU_MON	1	1...FFH	1...255	1	-
Maximum of anti bounce counter					
IP_TQI_REF_MAX_MON	7	0...7FFFH	0...1023.97	0.03125	Nm
LDP_N_32_MON_IP_TQI_MAX_MON	7	0...FFH	0...8160	32	rpm
Maximum reference indicated engine torque					

E.17.1 ECM2_DTSYSTQMAX

Import actions:

ACTION_ECM3_Service8TaskPfm(2)
ACTION_ECM3_Service9TaskPfm(2)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 2.

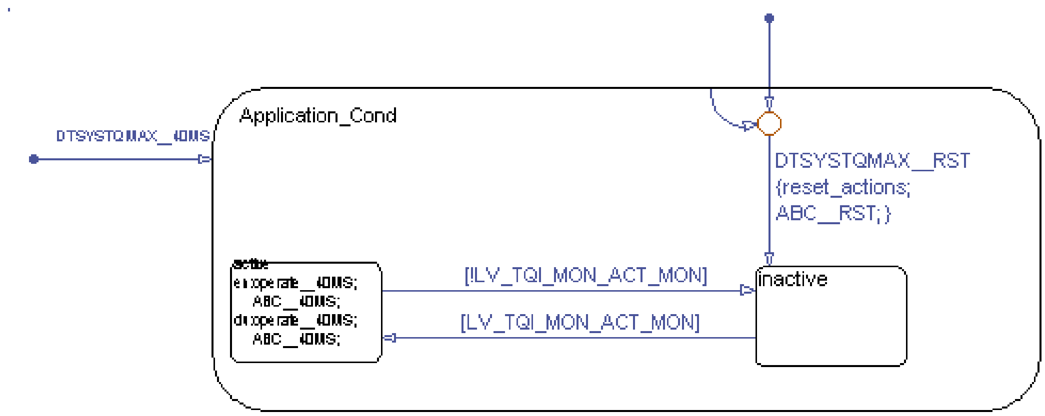
ACTION_ECM3_WriteChkCpl(IN <type>, INOUT<xyz_MON>, OUT<xyz_MON_CPL>, IN<reg>)
ACTION_ECM3_ChkCpl(IN<type>, IN<xyz_MON>, IN<xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN<type>, OUT<reg>, IN<xyz_MON>, IN<xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode "O" and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

General information:

The objective of this function is the monitoring of the maximum torque at clutch. The maximum torque at clutch is compared to a model threshold value (TQ_MAX_CLU_MAX_MON). The factors influencing this threshold are: IP_TQI_REF_MAX_MON (the worst case condition i.e. maximum possible fresh air entering in the cylinder is taken into account inside this map) and TQ_LOSS_REQ_CLU_MON. A fault is detected after debouncing, if the maximum torque at clutch exceeds this threshold.

Application Condition



***Initialisation(DTSYSTQMAX__RST)*:** for condition see 'Application Incidences of Process Monitoring'

Function Description

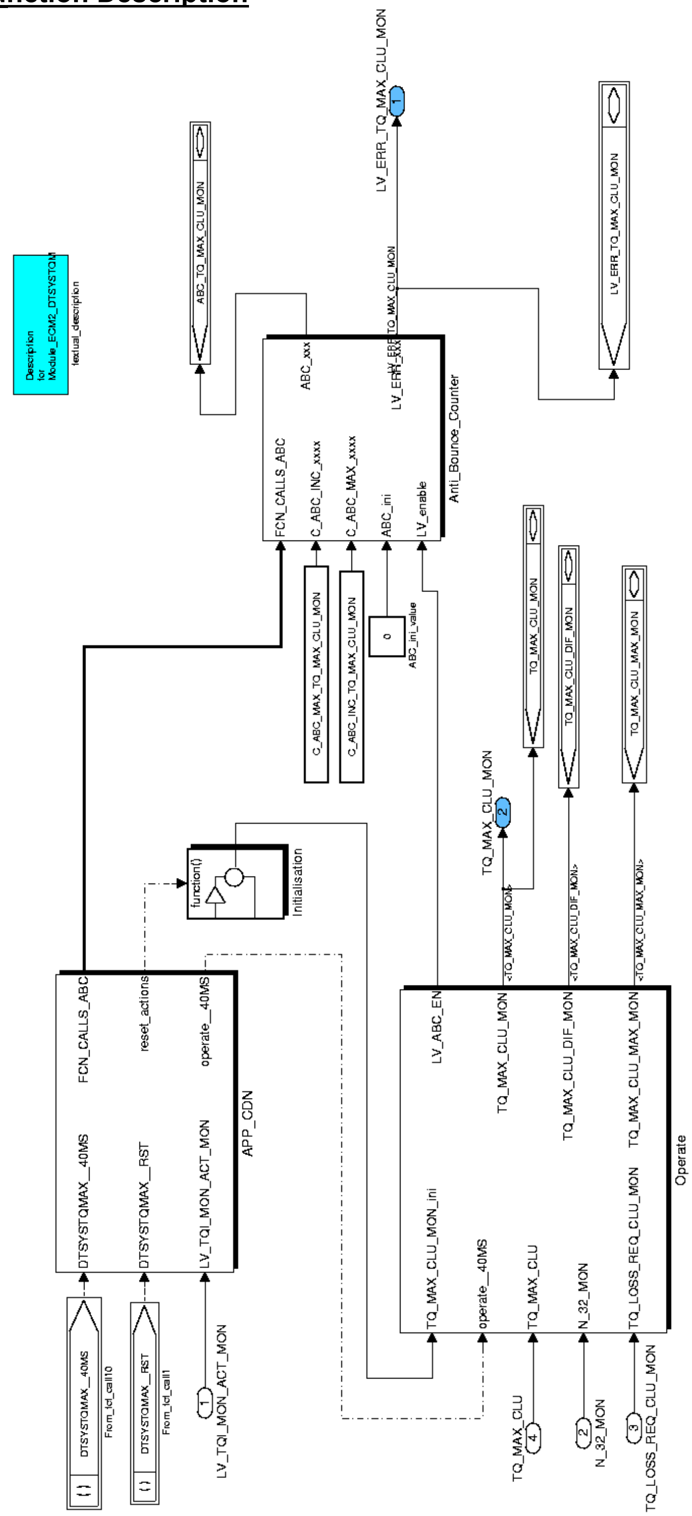


Figure 47 ECM2_DTSYSTQMAX

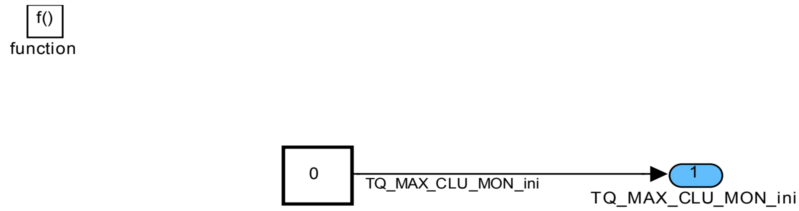


Figure 2 ECM2_DTSYSTQMAX/Initialisation

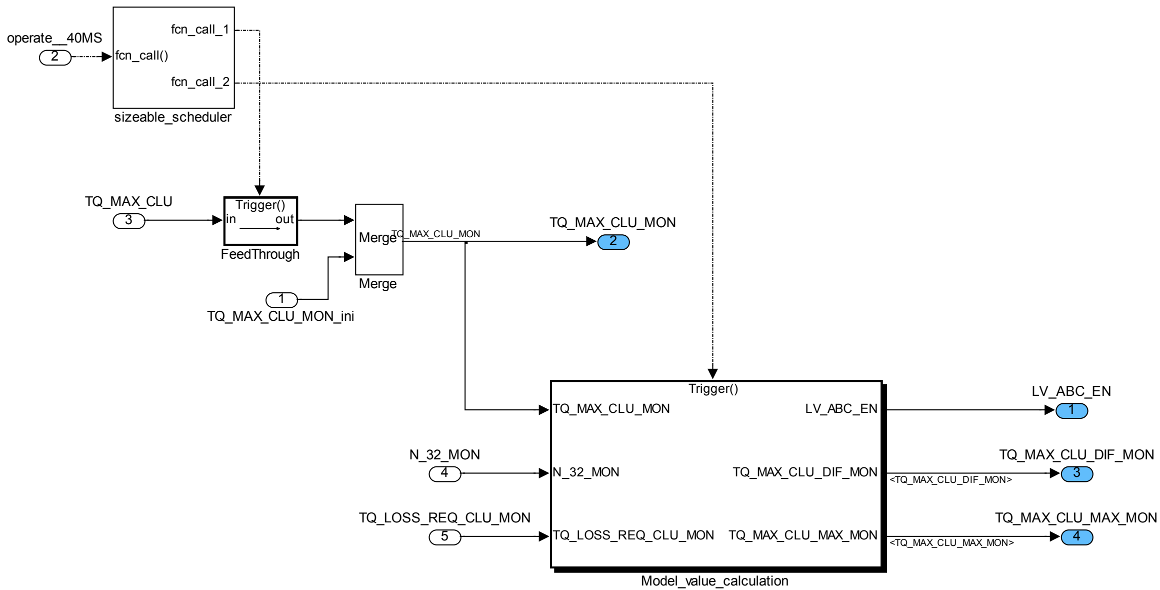


Figure 3 ECM2_DTSYSTQMAX/ Operate

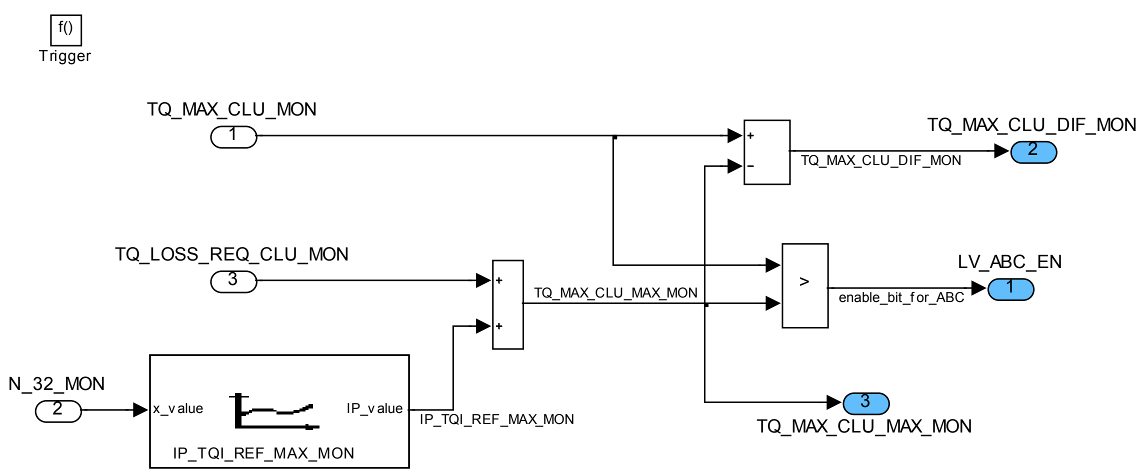


Figure 4 ECM2_DTSYSTQMAX/ Operate/ Model_value_calculation

E.18 Monitoring of CAN Signals

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_MSR_MON	O/V	0...FFH	0...510	2	Nm
Torque demand from drag control					
TQI_GS_MON	O/V	0...FFH	0...510	2	Nm
Torque demand for Gear Shift					
LV_MSR_INH_MON	O/V	0...1H	0...1	1	-
Fault in MSR present					
LV_GS_INH_MON	O/V	0...1H	0...1	1	-
Fault in GS Intervention present					
TQI_INC_CAN_MON	O/V	0...FFH	0...510	2	Nm
summarized Torque demand via CAN					
ABC_GS_MON	V	0...FFH	0...255	1	-
value of the Gear Shift anti - bounce counter					
ABC_MSR_MON	V	0...FFH	0...255	1	-
value of the MSR anti - bounce counter					
LV_CAN_INH_MSR_MON	V	0...1H	0...1	1	-
Logical Variable to inhibit MSR due to CAN fault					
LV_CAN_INH_GS_MON	V	0...1H	0...1	1	-
Logical Variable to inhibit GS due to CAN fault					
T_MSR_INH_MON	V	0...FFH	0...10.2	0.04	s
Timer for inhibition of MSR torque demand in monitoring level					
T_GS_INH_MON	V	0...FFH	0...10.2	0.04	s
Timer for inhibition of GS torque demand in monitoring level					

Input data:

LV_TQI_MON_ACT_MON	LV_MSR_ACT	TQI_MSR_CAN	LV_ERR_CAN_BUS_OFF
VB	TOUT_CTR_TCS1	LV_MSR_INH_T_MSR	TQI_TCS_CAN
TQI_TCU_MSR_CAN	LV_GS_ACT	TOUT_CTR_TCU1	F_TCU
LV_ERR_TIMEOUT_TCU1	TAR_GC_CAN		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_GS_MON	1	0...FFH	0...255	1	-
Anti - bounce counter increment					
C_ABC_INC_MSR_MON	1	0...FFH	0...255	1	-
Anti - bounce counter increment					
C_ABC_MAX_GS_MON	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter					
C_ABC_MAX_MSR_MON	1	1...FFH	1...255	1	-
Maximum value of the anti - bounce counter					
C_CAN_VB_MIN_DIAG_MON	1	0...FFH	0...25.8984375	0.1015625	V
Threshold of battery voltage for CAN diagnosis					
C_THD_MAX_GS_MISS_MON	1	0...FFH	0...255	1	-
max. allowed number for no incoming CAN- msg					

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_THD_MAX_TCS1_MISS_MON	1	0...FFH	0...255	1	-
max. allowed number for no incoming CAN- msg					
C_TQI_GS_MAX_MON	1	0...FFH	0...510	2	Nm
Maximum allowed indicated engine torque for Gear intervention					
C_TQI_STND_MON	1	0...FFH	0...510	2	Nm
Maximum indicated engine torque (scaling factor for torque demands from CAN)					
C_T_CAN_VB_VLD_MON	1	0...FFH	0...10.2	0.04	s
Delay time for VB valid for CAN diagnosis in monitoring level					
C_T_GS_INH_MON	1	0...FFH	0...10.2	0.04	s
Delay time for disable of GS torque demand in monitoring level					
C_T_MSR_INH_MON	1	0...FFH	0...10.2	0.04	s
Delay time for disable of MSR torque demand in monitoring level					

Import actions:

ACTION_ECM3_Service4TaskPfm(IN <n>)
ACTION_ECM3_Service5TaskPfm(IN <n>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring Services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument n has the value 3.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with mode O and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

FUNCTION DESCRIPTION:

General Information:

External Requests via CAN have to be monitored because they could lead to a large increase of engine torque. For this purpose the communication on the CAN Bus is checked and also the Message- content is checked for plausibility.

Application Condition

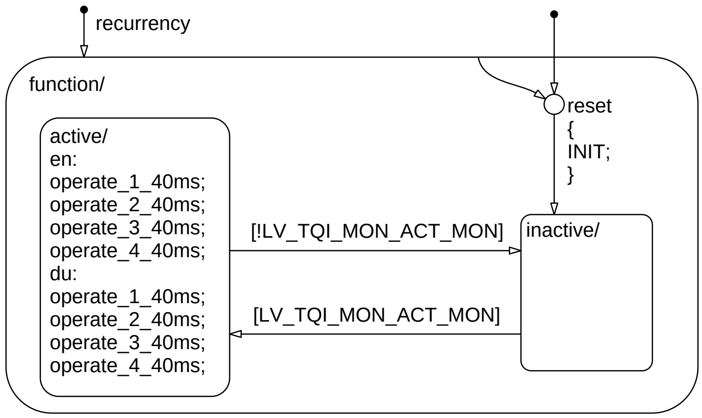


Figure 48 ECM2_DTSYSCAN_E00B/ APP_CDN/ Chart

Function Description

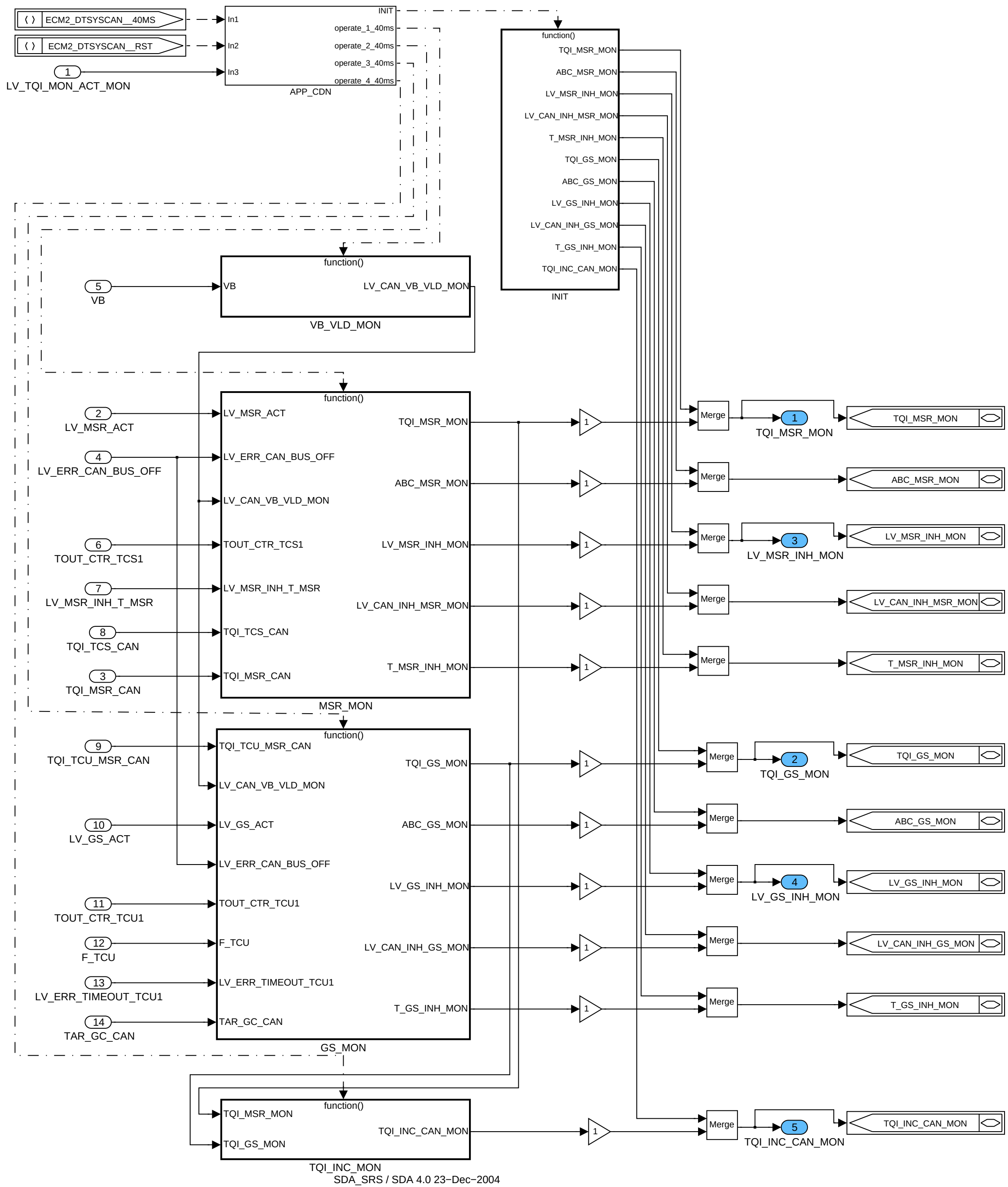


Figure 49

E.18.1.1 INIT

For condition see: 'Application Incidences of Process Monitoring'

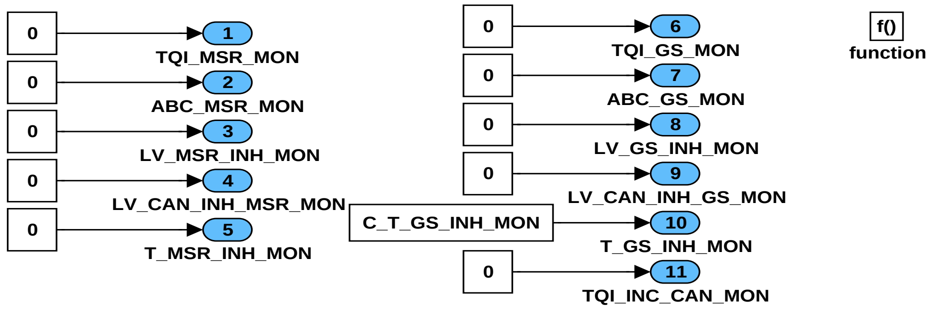


Figure 3 ECM2_DTSYSCAN_E00B/

E.18.1.2 SUBFUNCTION: GS MON

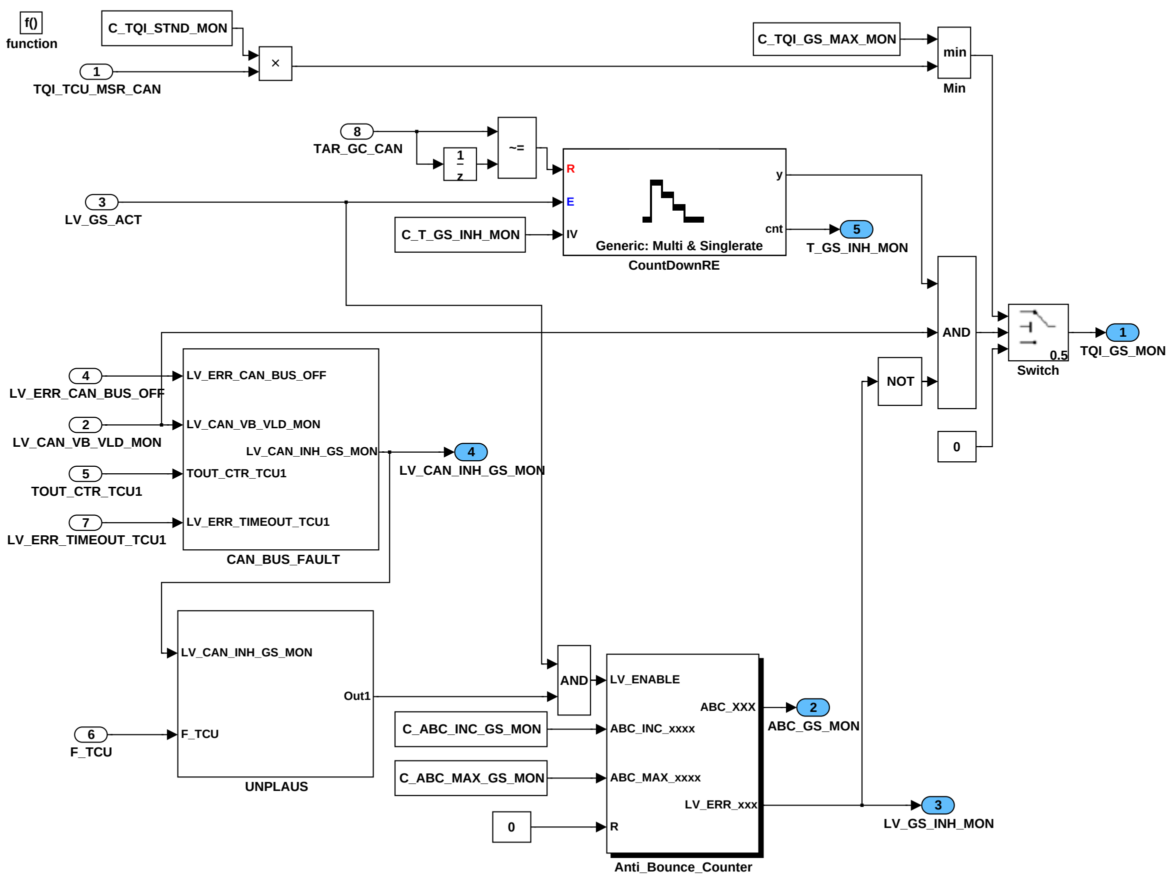


Figure 4 ECM2_DTSYSCAN_E00B/ GS_MON

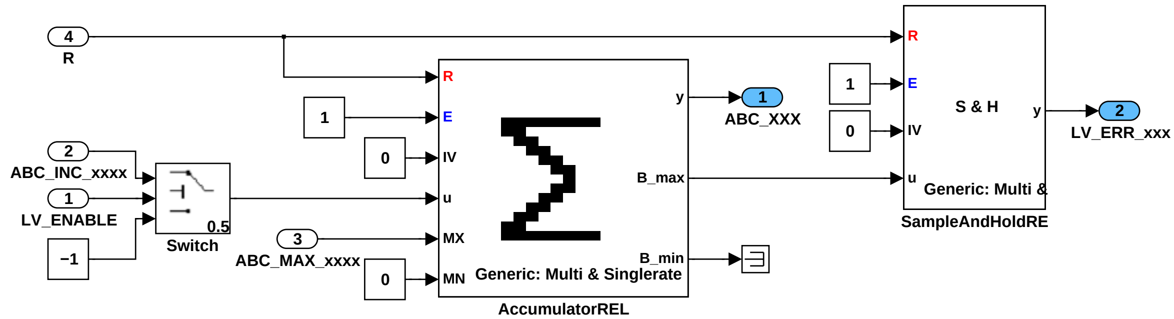


Figure 5 ECM2_DTSYSCAN_E00B/ GS_MON/ Anti_Bounce_Counter

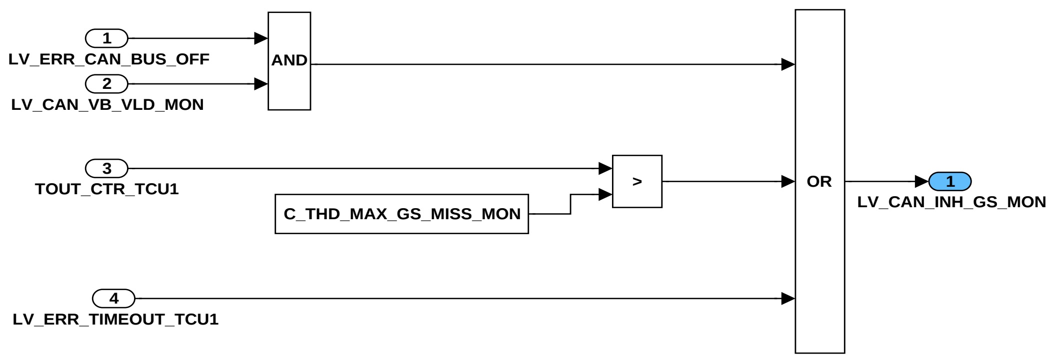


Figure 6 ECM2_DTSYSCAN_E00B/ GS_MON/ CAN_BUS_FAULT

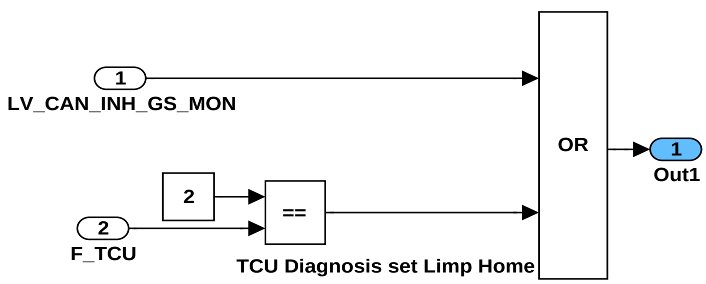


Figure 7 ECM2_DTSYSCAN_E00B/ GS_MON/ UNPLAUS

E.18.1.3 SUBFUNCTION: MSR MON

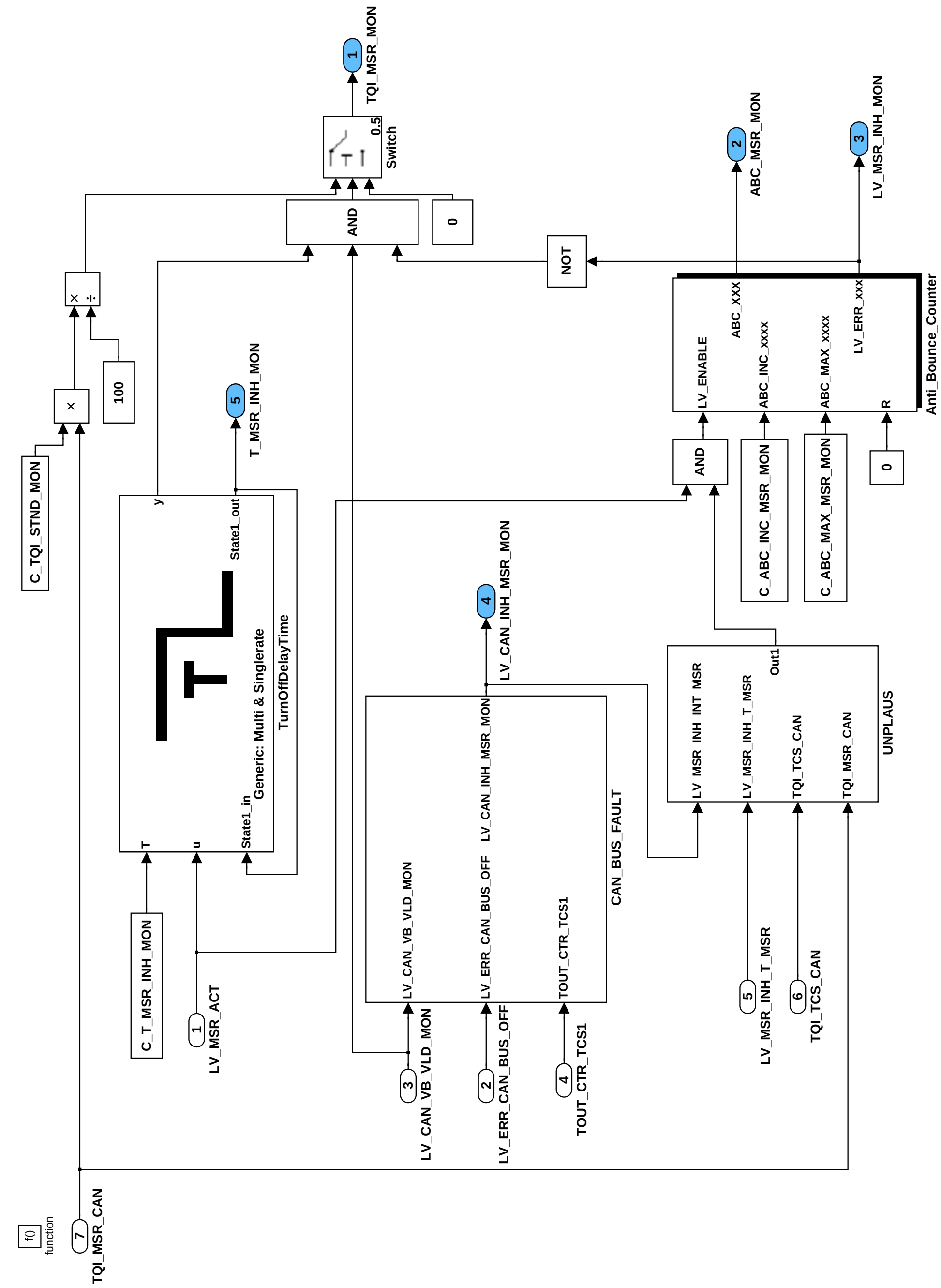


Figure 8 ECM2_DTSYSCAN_E00B/ MSR_MON

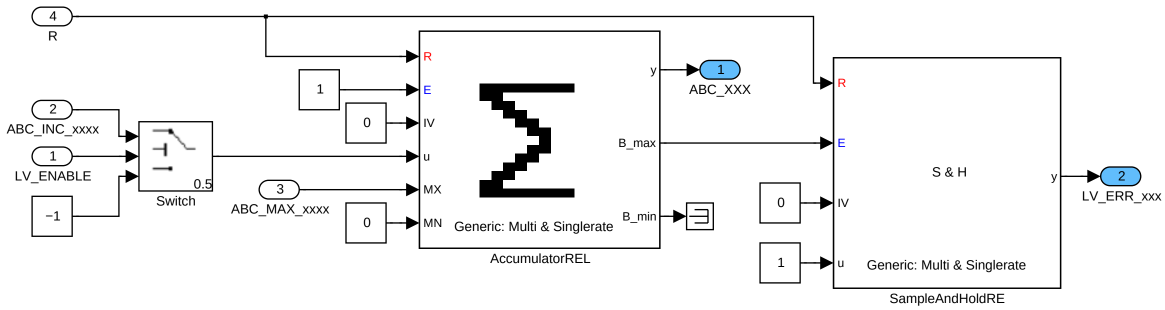


Figure 9 ECM2_DTSYSCAN_E00B/ MSR_MON/ Anti_Bounce_Counter

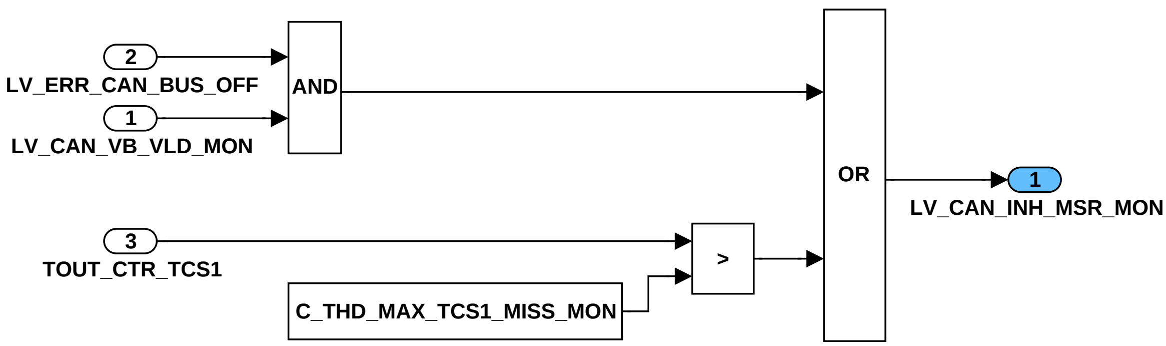


Figure 10 ECM2_DTSYSCAN_E00B/ MSR_MON/ CAN_BUS_FAULT

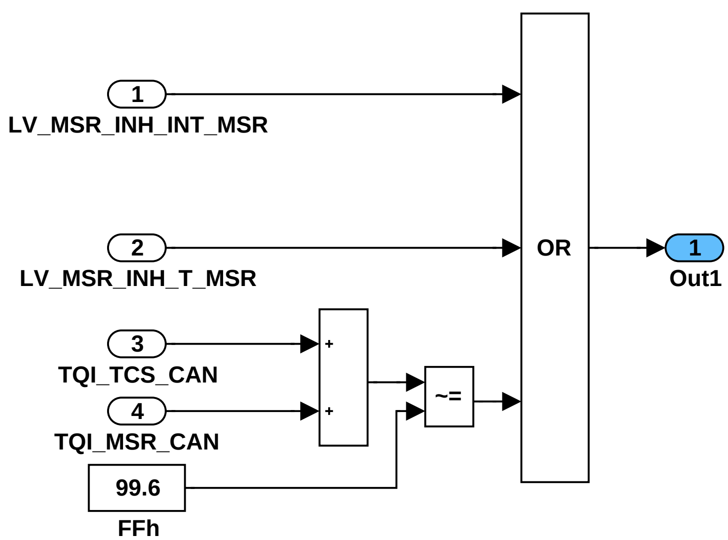


Figure 11 ECM2_DTSYSCAN_E00B/ MSR_MON/ UNPLAUS

E.18.1.4 SUBFUNCTION: TQI_INC_MON

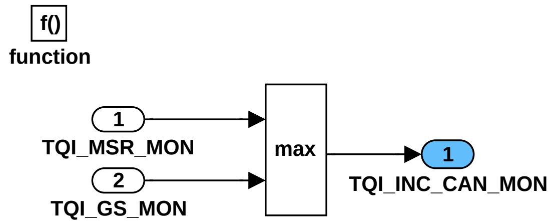


Figure 12 ECM2_DTSYSCAN_E00B/ TQI_INC_MON

E.18.1.5 SUBFUNCTION: VB VLD MON

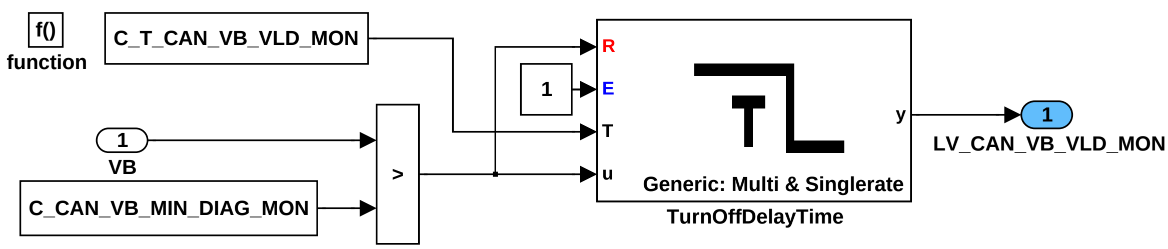


Figure 13 ECM2_DTSYSCAN_E00B/ VB_VLD_MON

E.19 Monitoring of engine speed

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_N_32_MON	O/V	0...1H	0...1	1	-
Logical variable for ratio error between N_32 and N_32_SUB_MON					
N_32_MON	O/V	0...FFH	0...8160	32	rpm
Engine speed (monitoring level)					
N_32_SUB_MON	V	0...FFH	0...8160	32	rpm
Engine speed substitute monitoring level					
ABC_N_32_MON	V	0...FFH	0...255	1	-
Anti bounce counter N_32 ratio error					
N_32_DIF_REL_MON	V	0...FFFFH	0...255	0.0039	-
Relative engine speed deviation monitoring level					

Input data:

T_SEG_SW	N_32		

Import actions:

ACTION_ECM3_Service6TaskPfm(IN <n>)
ACTION_ECM3_Service7TaskPfm(IN <n>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring Services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument n has the value 1.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with mode O and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_N_32_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment (additive value in case of N_32 ratio error)					
C_ABC_MAX_N_32_MON	1	1...FFH	1...255	1	-
Value at which N_32 ratio error is recognized, when reached					
IP_N_32_DIF_REL_MAX_MON	6	0...FFFFH	0...255	0.0039	[-]
LDP_N_32_MON_IP_N_32_DIF_MON	6	0...FFH	0...8160	32	[rpm]
Maximum relative difference between N_32 and N_32_SUB_MON					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_T_SEG_N_32_CLC	1	0...FFFFFFFFH	0 ... 1.374400e+011	32	rpm*µs
Non calibratable data for N_32_SUB_MON calculation					

E.19.1 FUNCTION PART: ECM2_DTSYSN

General information:

The comparison of the engine speed N_32 to an engine speed substitute N_32_SUB_MON, calculated in the monitoring level, performs the monitoring of the engine speed used in the monitoring level.

N_32_SUB_MON is calculated from a segment timing information that is generated by a software timer solution in the BSW (T_SEG_SW). The software timer is triggered by crankshaft segments and produces time stamps for the trigger events. From these timing informations the signal T_SEG_SW is determined (resolution: 1µs).

The engine speed signals N_32 and N_32_SUB_MON are compared to each other by means of a relative deviation. The ratio of the difference of N_32 and N_32_SUB_MON related to the minimum of the two values is not allowed to deviate more than the maximum difference IP_N_32_DIF_REL_MAX_MON, depending on MAX_N_32 value. Otherwise a process fault is assumed.

On an error free system N_32 is copied to N_32_MON. If an engine speed error is detected, the maximum value of N_32 and N_32_SUB_MON is copied to N_32_MON to prevent from a sleeping fault in the module ‚Monitoring of engine speed limitation'.

Note: This version of engine speed monitoring is for systems where a second hardware timer, which in other versions was delivering the signal SEG_T_MES, is not available on the ECU and where the missing timer is substituted by a software timer solution in the BSW.

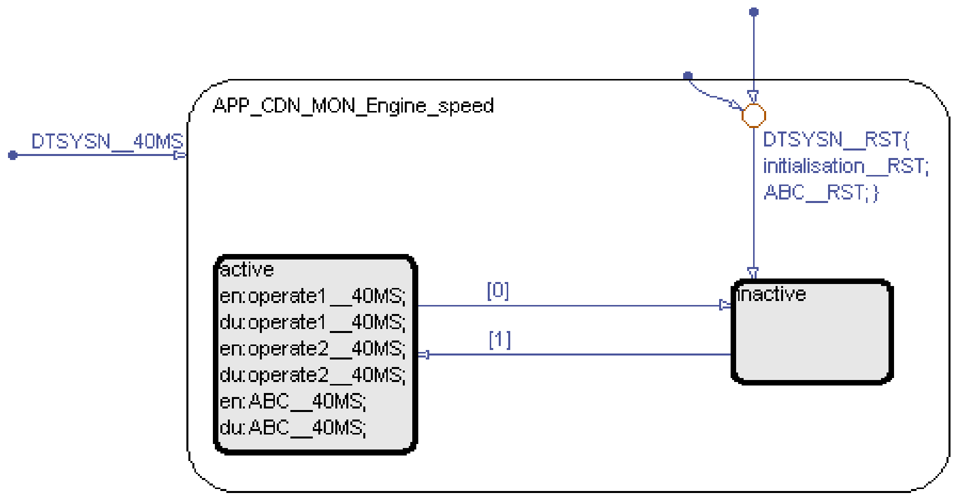
For configuration data:

NC_T_SEG_N_32_CLC is a constant for the calculation of N_32_SUB_MON from T_SEG_SW:

$$NC\_T\_SEG\_N\_32\_CLC = \frac{2*60*1000000}{NC\_CYL\_NR*\{resol.\ of\ \ N\_32\_SUB\_MON\}}$$

with *NC_CYL_NR* : number of cylinders

Application Condition



Note: DTSYSN__RST includes the function calls as defined in application incidences.

Function Description

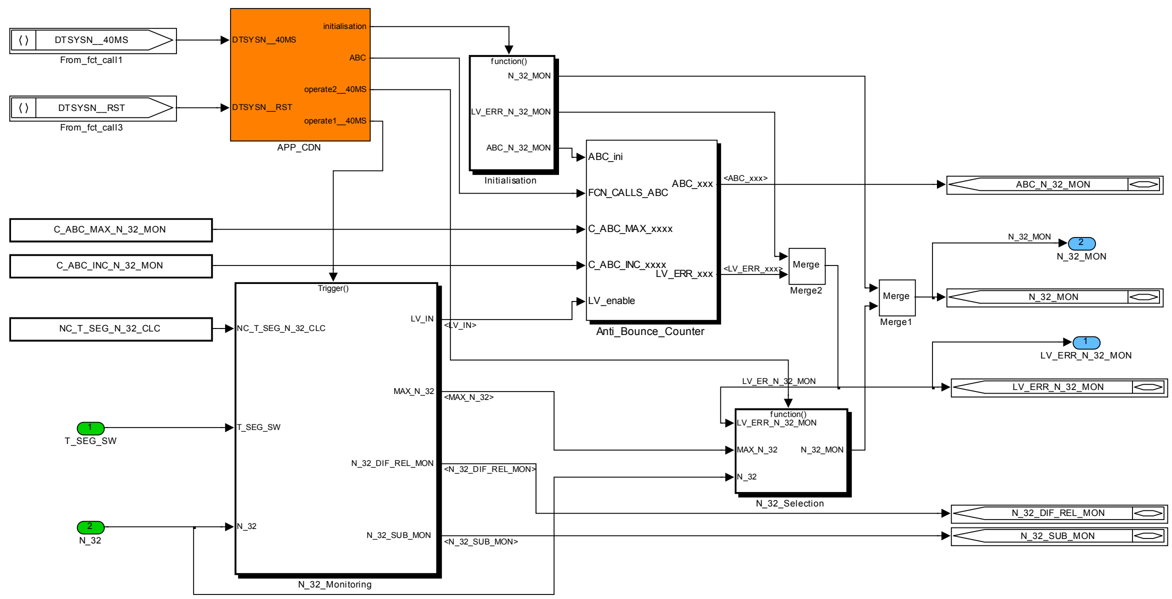


Figure 50 ECM2_DTSYSN

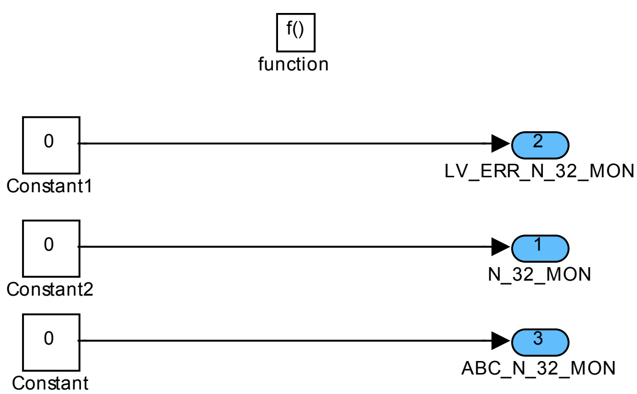


Figure 51 ECM2_DTSYSN/ INITIALISATION

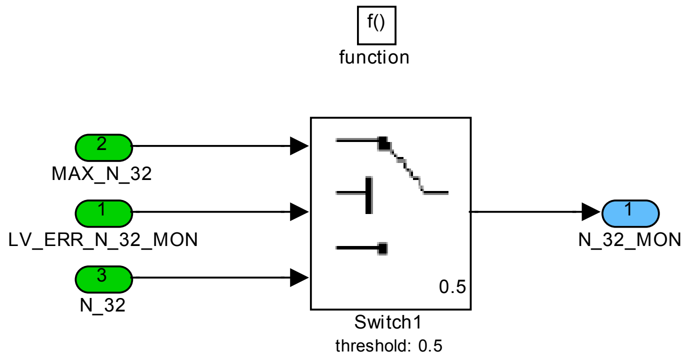


Figure 3 ECM2_DTSYSN/ N_32_MON_Selection

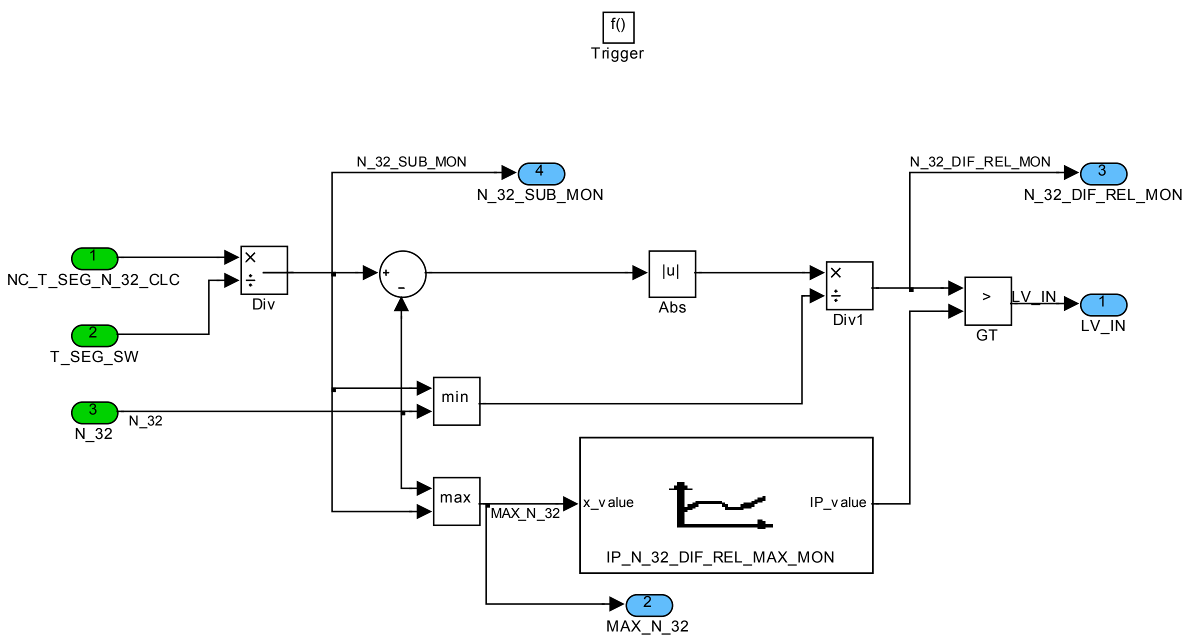


Figure 4 ECM2_DTSYSN/ N_32_MONITORING

E.20 Monitoring of pedal value signals(PVS)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_PVS_MON	V/O	0...1H	0...1	1	[-]
Logical variable for different PVS diagnosis between L1 and L2					
LV_ERR_PVS_RATIO_MON	V/O	0...1H	0...1	1	[-]
Logical variable for ratio-error between PVS channel 1 and 2					
PV_AV_MON	V/O	0...FFH	0...99.60937	0.390625	[%]
Level 2 degree of activation of the accelerator pedal					
ABC_PVS_RATIO_MON	V	0...FFH	0...255	1	[-]
Anti bounce counter PVS ratio error					
PV_AV_1_MON	V	0...FFH	0...99.60937	0.390625	[%]
Degree of activation of the accelerator pedal (channel 1)					
PV_AV_2_MON	V	0...FFH	0...99.60937	0.390625	[%]
Degree of activation of the accelerator pedal (channel 2)					

Input data:

LV_BRAKE_MON	LV_ERR_VCC_PVS_1_MON	LV_ERR_VCC_PVS_2_MON	LV_ERR_VCC_PVS_MON
LV_ERR_PVS_1	LV_ERR_PVS_2	LV_ERR_PVS_RATIO	LV_IGK_MON
N_32_MON	V_PVS_1_BAS	V_PVS_2_BAS	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_N_MIN_PVS_DIAG_MON	1	0...FFH	0...8160	32	[rpm]
Minimum engine speed for PVS diagnosis					
C_PV_LIH_MAX_MON	1	0...FFH	0...99.60937	0.390625	[%]
Max. pedal position on case of any PVS error					
C_ABC_INC_PVS_RATIO_MON	1	0...FFH	0...255	1	[-]
Anti bounce counter increment (additive value in case of error)					
C_ABC_MAX_PVS_RATIO_MON	1	1...FFH	1...255	1	[-]
Value at which ratio error is recognized, when reached					
C_V_PVS_IS_THD_1_MON	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Threshold for idle speed PVS channel 1					
C_V_PVS_IS_THD_2_MON	1	0...3FFH	0...4.99511	4.8828e-3	[V]
Threshold for idle speed PVS channel 2					
C_V_PVS_SLOP_1_MON	1	0...FFH	0...79.6875	0.3125	[%/V]
Slope of the PVS channel 1					
C_V_PVS_SLOP_2_MON	1	0...FFH	0...159.375	0.625	[%/V]
Slope of the PVS channel 2					
IP_PV_RATIO_HYS_MON	6	0...FFH	0...99.60937	0.390625	[%]
LDP_PV_AV_MIN_MON_IP_PV_MON	6	0...FFH	0...99.60937	0.390625	[%]
Maximum PV ratio error allowed					

E.20.1 ECM2_DTSYSPVS

Import actions:

ACTION_ECM3_Service2TaskPfm(2)
ACTION_ECM3_Service3TaskPfm(2)

Note: These actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 2.

ACTION_ECM3_WriteChkCpl(IN<type>, INOUT<xyz_MON>, OUT<xyz_MON_CPL>, IN<reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN<type>, OUT<reg>, IN<xyz_MON>, IN<xyz_MON_CPL>)

Note: These actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

General information:

The pedal value diagnosis is made in the monitoring level in order to check the V_PVS input value of the torque monitoring function (level 2).

After the diagnosis, a resulting pedal value PV_AV_MON is used as input for the torque monitoring function.

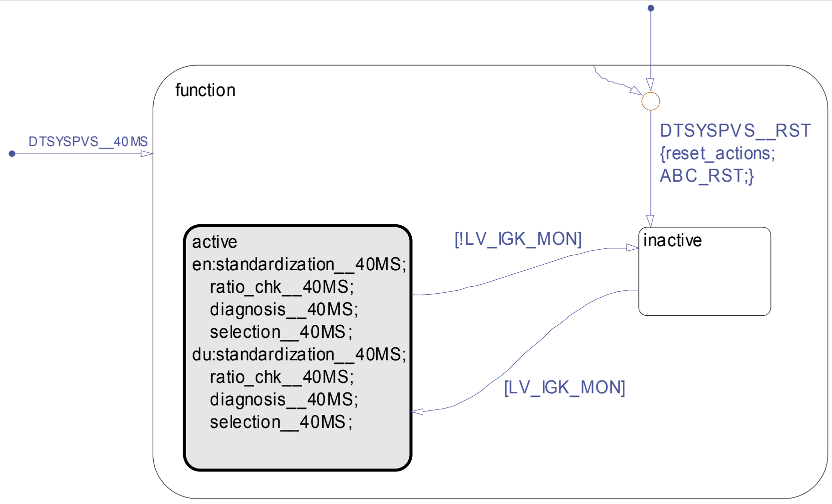
A malfunction of level 1 will be assumed, if a ratio error of the two redundant pedal values is diagnosed after debouncing, yet no PVS failure was detected previously by the PVS diagnosis function of level 1. The following errors were detected:

- The difference between the standardized values PV_AV_1_MON and PV_AV_2_MON is too large.
- The diagnosis system of level 2 detects a PVS ratio fault and no fault is detected by the diagnosis of level 1.
- Any PVS error-BIT of the PVS diagnosis in level 1 has been reset without permission.

To ensure that the PVS diagnosis of level 1 detects a fault earlier than the PVS ratio diagnosis of level 2, the fault detection time of the diagnosis function in level 2 has to be adjusted that it exceeds the fault detection time of level 1. Therefore, all level 1 PVS diagnosis thresholds, all constants of the debounce counters and the cycle time have to be taken into account.

The input values V_PVS_1_BAS and V_PVS_2_BAS are directly read from the A/D-converter buffer.

Application Condition



***Initialisation(DTSYSPVS__RST)*:** for condition see 'Application Incidences of Process Monitoring'.

Function Description

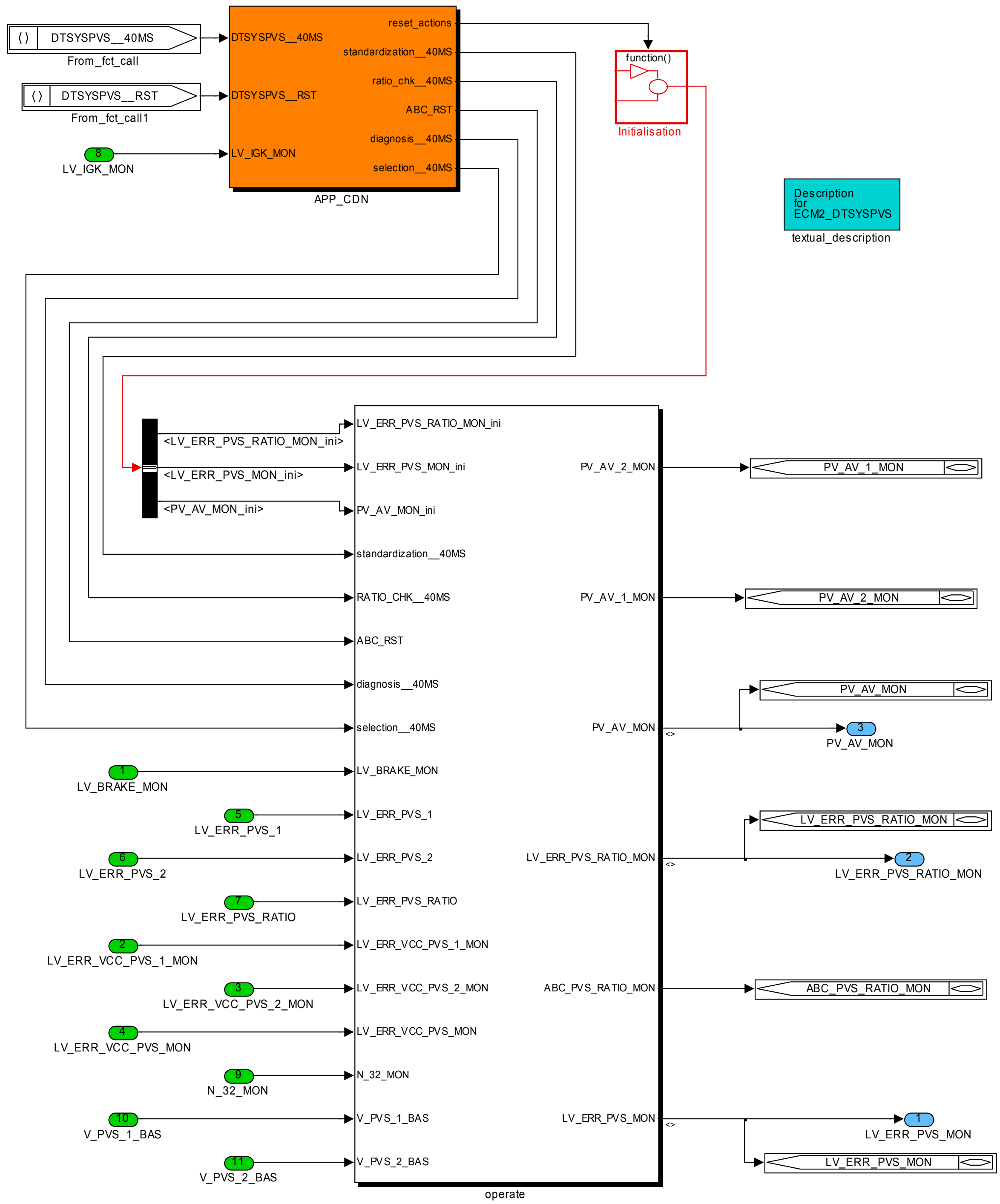


Figure 52 Monitoring functionality overview

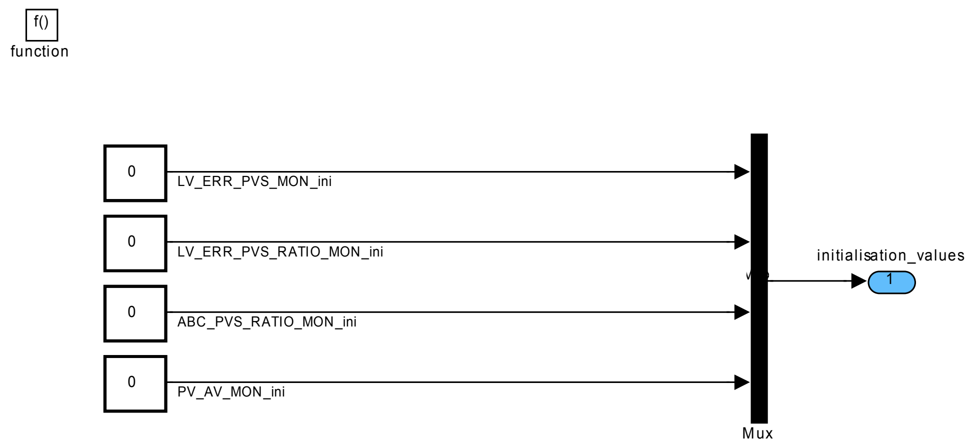


Figure 53 ECM2_DTSYSPVS/ Initialisation

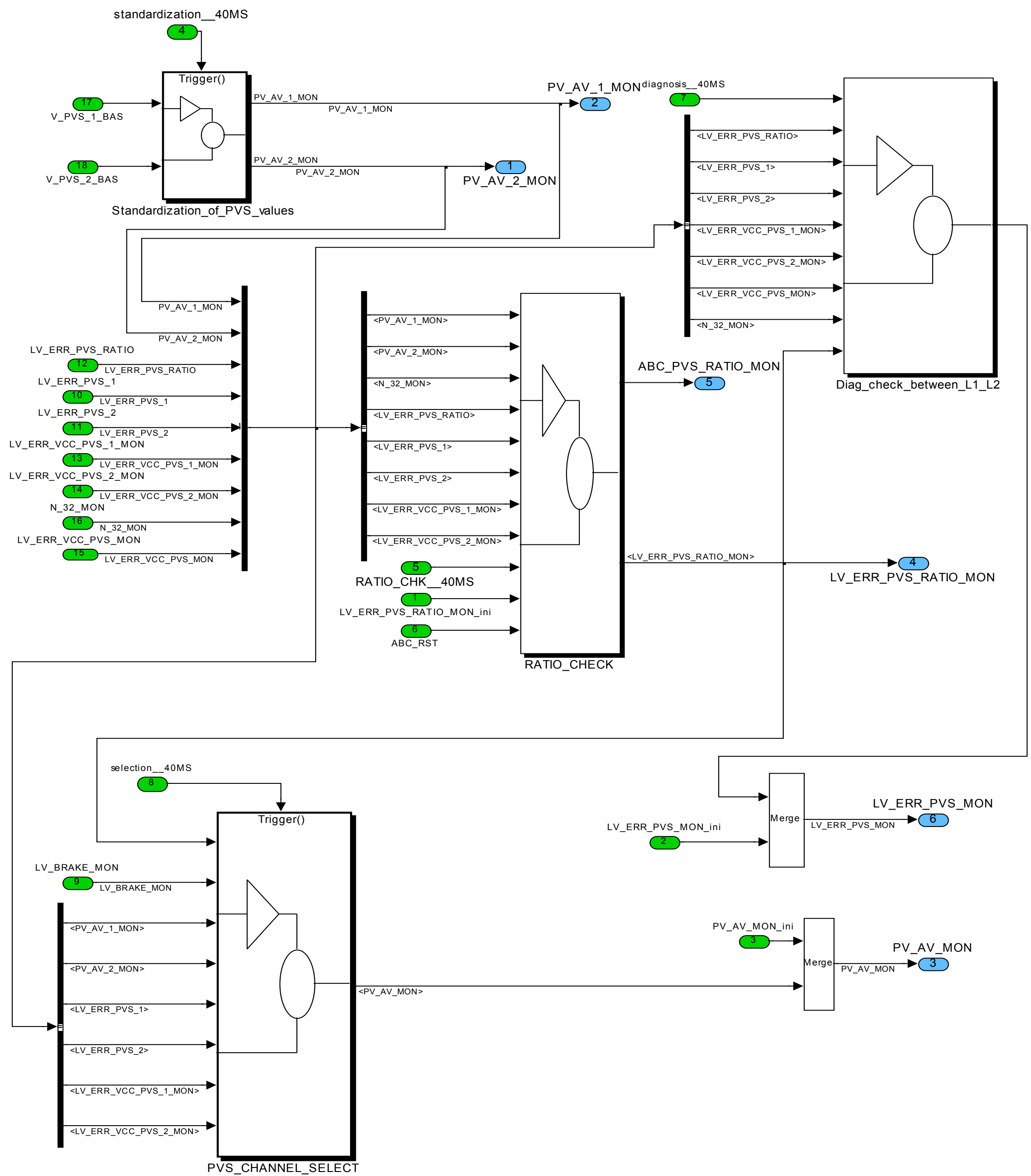


Figure 54 ECM2_DTSYSPVS/ operate

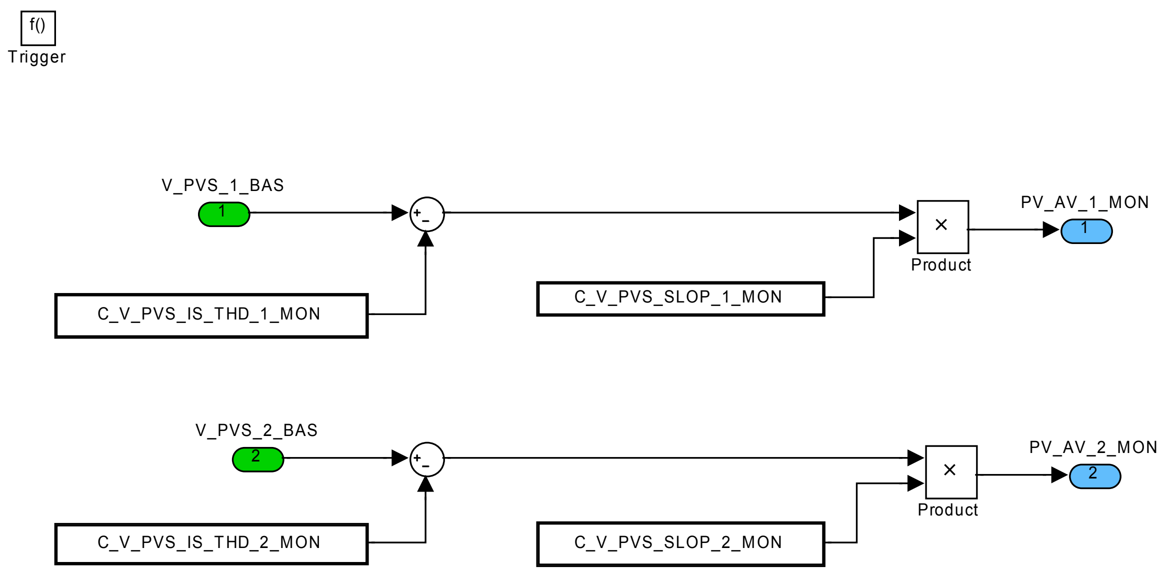


Figure 55 ECM2_DTSYSPVS/ operate/ Standardization_of_PVS_values

ECM2_DTSYSPVS/OPERATE/RATIO_CHECK

Both sensors of the PVS-system supply output signals, which are standardized between 0 and 99.6 %. In case of an error-free system (no signal range error detected), the difference between the standardized values PV_AV_1_MON and PV_AV_2_MON is not allowed to be greater than IP_PV_RATIO_HYS_MON.

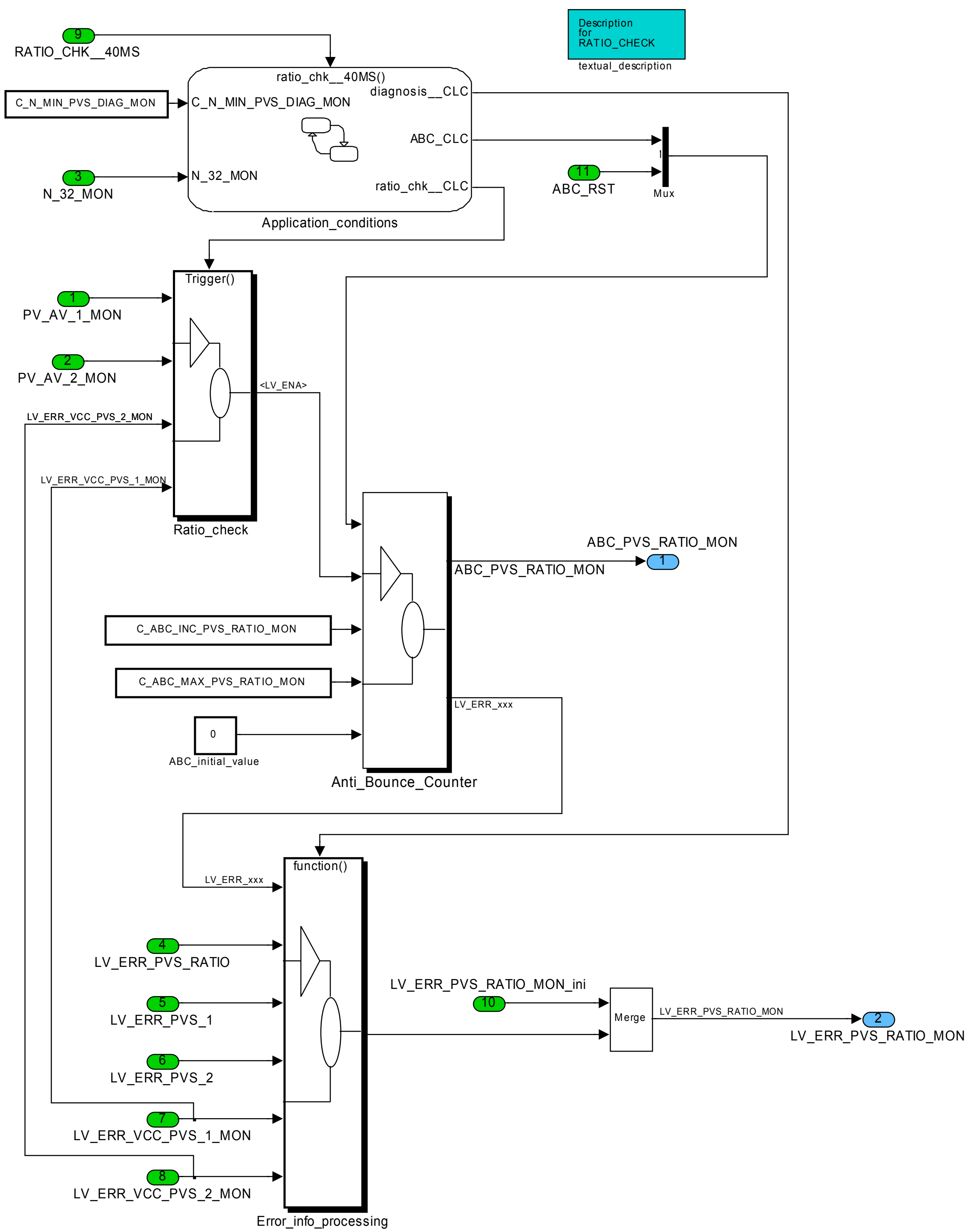


Figure 56 ECM2_DTSYSPVS/ operate/ RATIO_CHECK

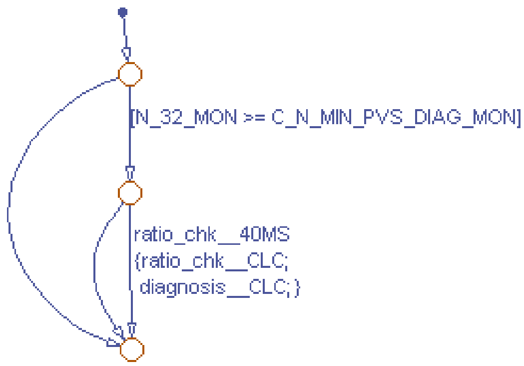


Figure: ECM2_DTSYSPVS/ operate/ RATIO_CHECK/Application_conditions

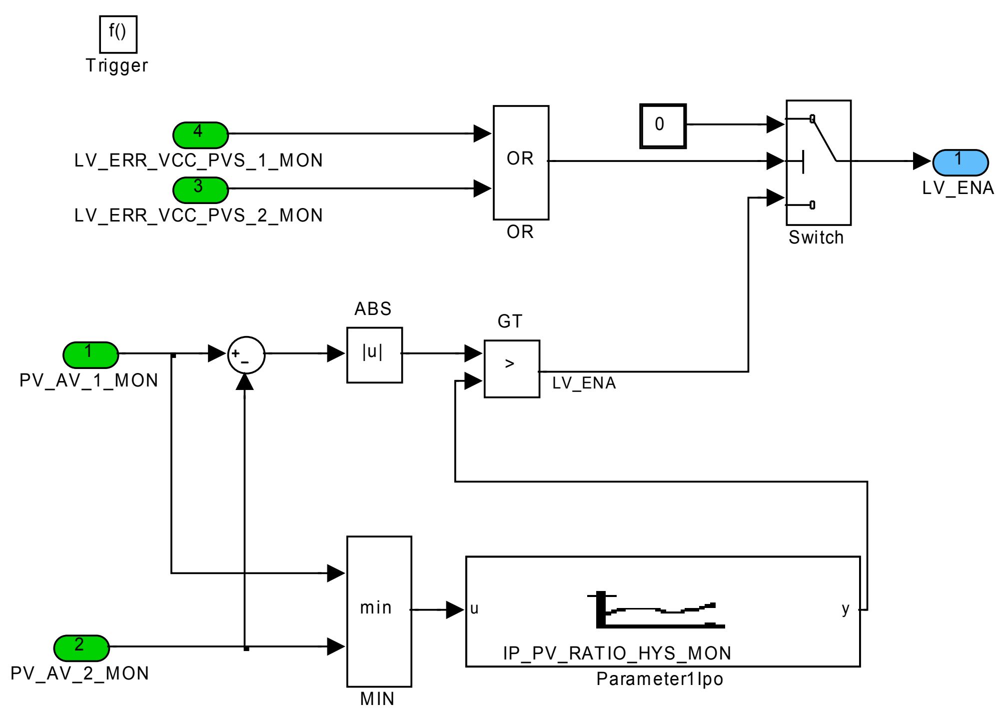


Figure 57 ECM2_DTSYSPVS/ operate/ RATIO_CHECK/ Ratio_check

ECM2_DTSYSPVS/OPERATE/RATIO_CHECK/ERROR_INFO_PROCESSING

The bits LV_ERR_VCC_PVS_1_MON and LV_ERR_VCC_PVS_2_MON are reversible. That means, the output LV_ERR_PVS_RATIO_MON has to be reversible also.
All PVS error information of the function level is considered. This allows the different process levels to be based on the same error information:

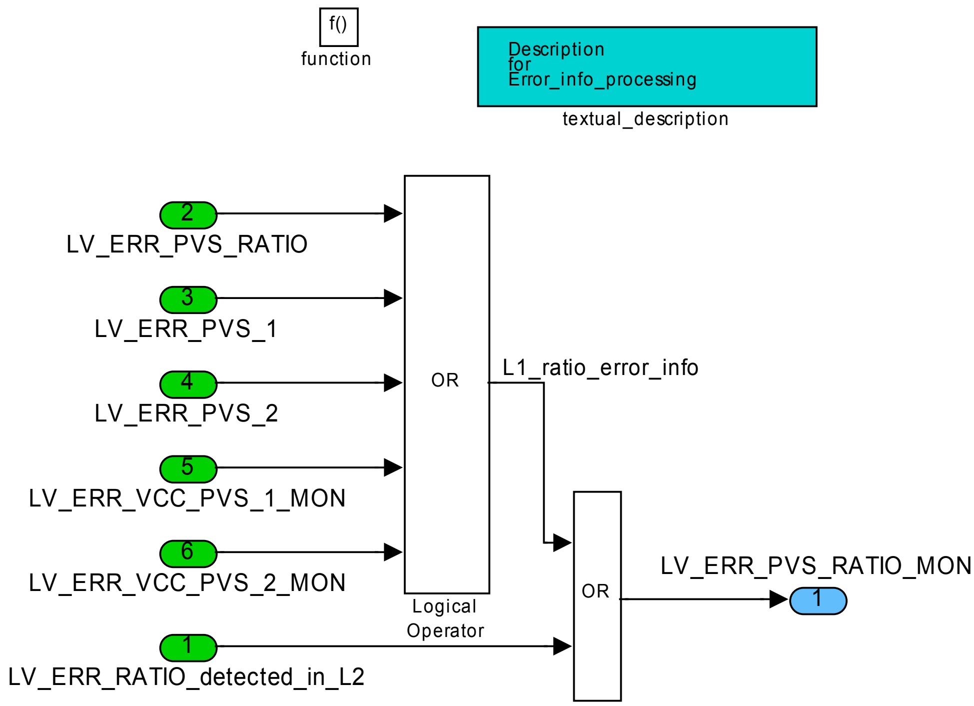


Figure 58 ECM2_DTSYSPVS/ operate/ RATIO_CHECK/ Error_info_processing

ECM2_DTSYSPVS/OPERATE/DIAG_CHECK_BETWEEN_L1_L2

If any PVS ratio error is detected in level 2 and no PVS error is detected in level 1, a fault of the PVS diagnosis in level 1 will be assumed.

There is also a check, whether an error-BIT of the PVS diagnosis in level 1 was reset without permission. This is done by using the result of LV_ERR_PVS_RATIO_MON of the last recurrence.

LV_ERR_VCC_PVS_MON is assumed to be a PVS supply voltage error flag of a level 2 diagnosis in case of any PVS supply voltage error.

LV_ERR_VCC_PVS_1_MON and LV_ERR_VCC_PVS_2_MON are error flags that result from Level 1 diagnosis of PVS-1 and PVS-2 supply voltage and are assigned to the

LV_ERR_VCC_PVS_1_MON and LV_ERR_VCC_PVS_1_MON respectively in the application incidences of process monitoring.

If an PVS monitoring error is detected and the error bit LV_ERR_PVS_MON is set, then this bit must not be reversible.

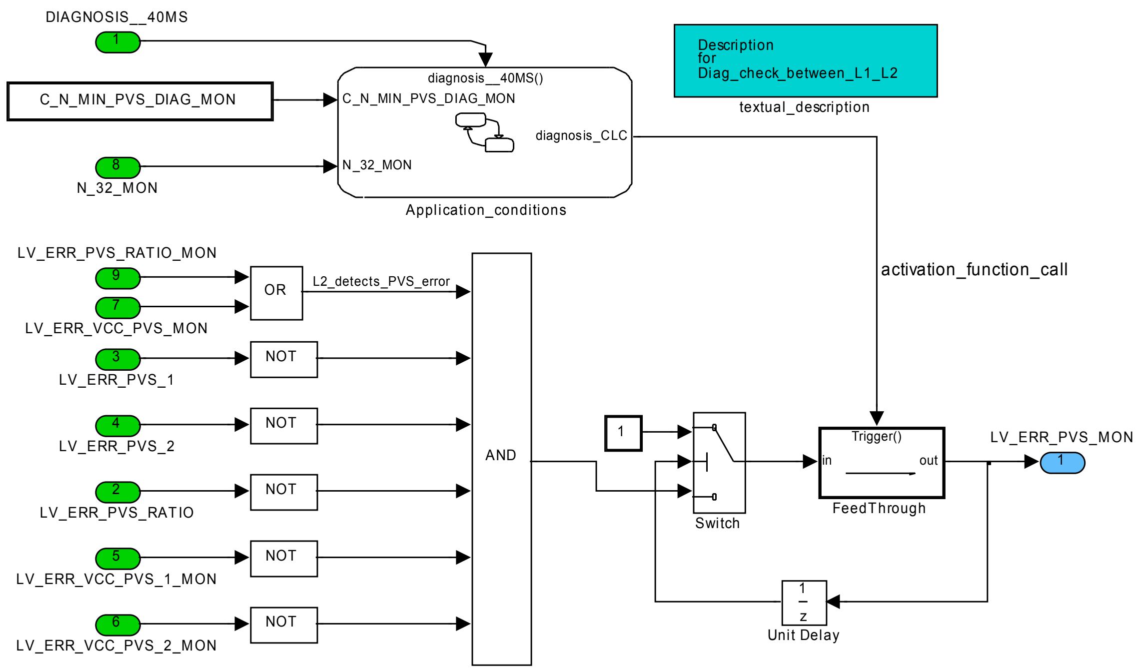


Figure 59 ECM2_DTSYSPVS/ operate/ Diag_check_between_L1_L2

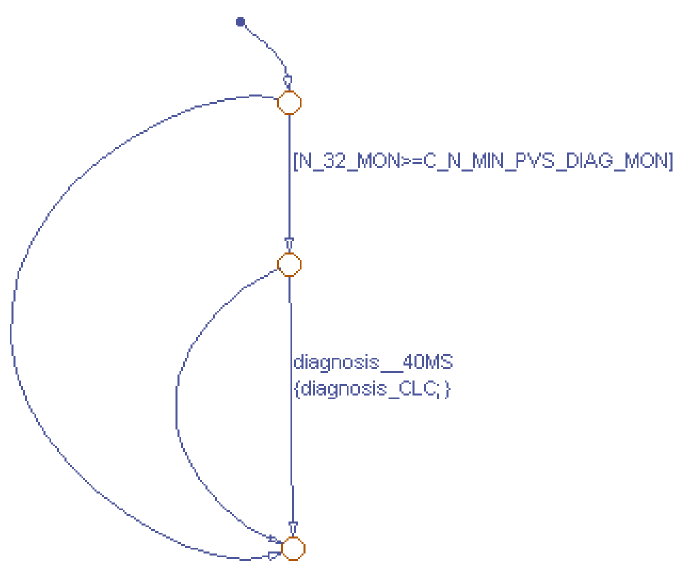


Figure: ECM2_DTSYSPVS/ operate/ Application_conditions

ECM2_DTSYSPVS/OPERATE/PVS_CHANNEL_SELECT

Depending on the plausibility, the degree of pedal activation PV_AV_MON is defined by the signals PV_AV_1_MON and PV_AV_2_MON, as follows:

Error Level	Error bits set	Analysis, error handling
0	-	PV_AV_MON = PV_AV_1_MON
1	LV_ERR_PVS_RATIO_MON	PV_AV_MON = MIN(PV_AV_1_MON, PV_AV_2_MON) (minimum)
2	LV_ERR_PVS_1 **OR** LV_ERR_VCC_PVS_1_MON	PV_AV_MON = PV_AV_2_MON (selection)
3	LV_ERR_PVS_2 **OR** LV_ERR_VCC_PVS_2_MON	PV_AV_MON = PV_AV_1_MON (selection)
4	(LV_ERR_PVS_1 OR LV_ERR_VCC_PVS_1_MON) **AND** (LV_ERR_PVS_2 OR LV_ERR_VCC_PVS_2_MON)	PV_AV_MON = 0 → driver demand not recoganizable

If any PVS error-BIT is set:

- PV_AV_MON will be limited to C_PV_LIH_MAX_MON.
- Use of the brake as a redundant driver demand:

  IF (LV_BRAKE_MON = 1)

  THEN

  PV_AV_MON = 0 (same as Error level 4)

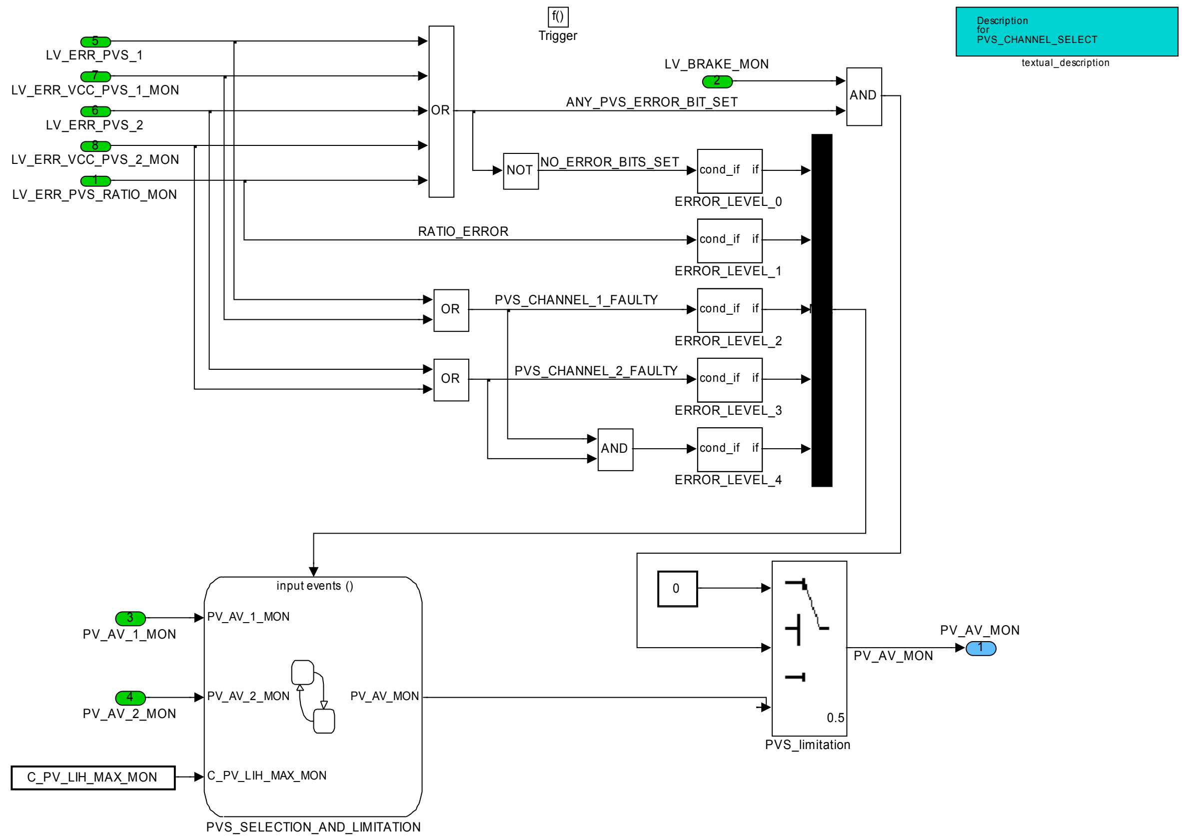


Figure 60 ECM2_DTSYSPVS/ operate/ PVS_CHANNEL_SELECT

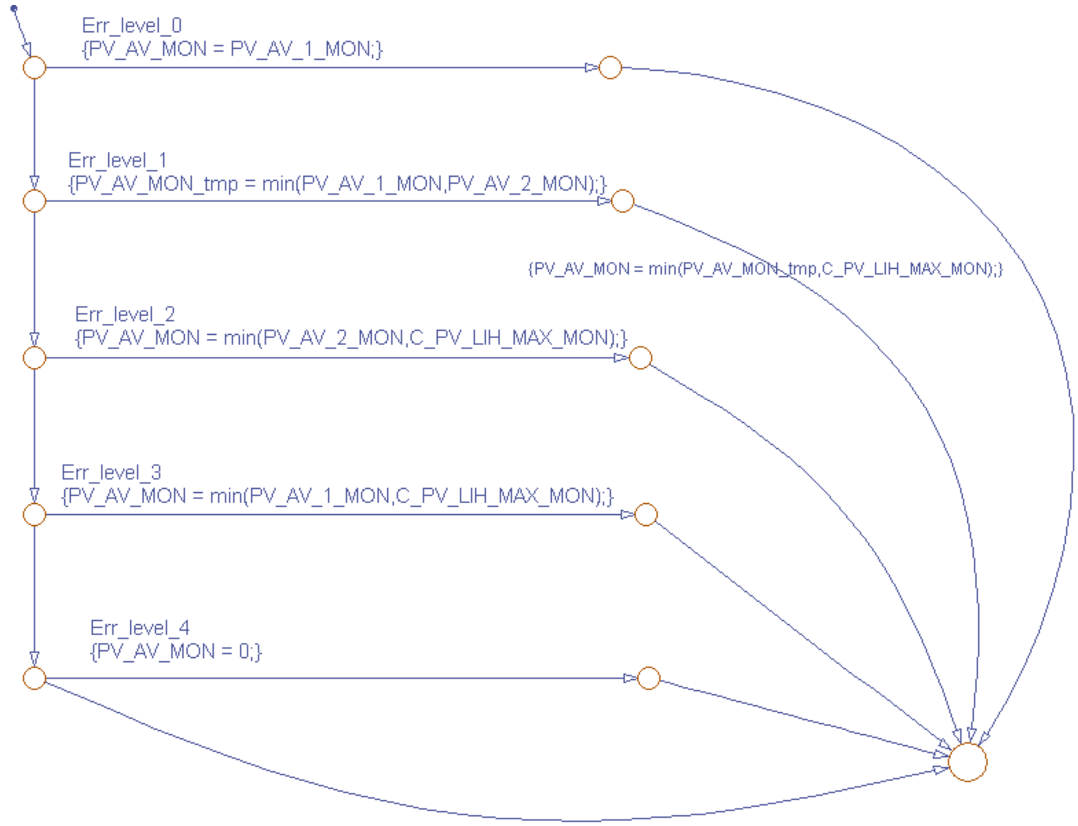


Figure: ECM2_DTSYSPVS/ operate/ PVS_CHANNEL_SELECT/ PVS_SELECTION_AND_ LIMITATION

E.21 Monitoring of mass air flow signal (MAF_MON)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MAF_SUB_MON	V	0..FFH	0..1.389E3	5.447	mg/stk
MAF substitute					
V_TPS_1_MON	V	0..FFH	0..4.980	0.0195	V
Adapted TPS signal channel 1					
V_TPS_2_MON	V	0..FFH	0..4.980	0.0195	V
Adapted TPS signal channel 2					
LV_ERR_TPS_MON	O/V	0..1H	0..1	1	-
Logical variable for different TPS diagnosis between L1 and L2					
ABC_TPS_RATIO_MON	V	0..FFH	0..255	1	-
Anti bounce counter TPS ratio diagnosis error					
MAF_SUB_COR_MMV_MON	O/V	0..FFH	0..1.389E3	5.447	mg/stk
Low pass filtered MAF_SUB_COR_MON					
MAF_MON	O/V	0..FFH	0..1.389E3	5.447	mg/stk
Mass air flow signal used monitoring level					
LV_ERR_MAF_MON	O/V	0..1H	0..1	1	-
Logical variable for error in mass air flow signal					
ABC_MAF_MON	V	0..FFH	0..255	1	-
Anti bounce counter MAF signal					

Input data:

V_TPS_1_BAS	V_TPS_2_BAS	MAF	LV_ERR_TPS_RATIO
N_32_MON	LV_ERR_TPS_2	LV_ERR_TPS_1	V_TPS_AD_EL_BOL_1
V_TPS_AD_EL_BOL_2	PUT	TIA_THR	LV_TQI_MON_ACT_MON
LV_ERR_TPS_MAF_1	LV_ERR_TPS_MAF_2		

Import actions:

ACTION_ECM3_Service4TaskPfm(2)
ACTION_ECM3_Service5TaskPfm(2)

Note: This actions are defined in chapter "Processor Monitroing", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 2.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Note: the output variables V_TPS_x_MON are not checked by the RAM test of process monitoring (level 2') but are only used for error memory storage.

FUNCTION DESCRIPTION:

General information:

The ETC safety concept checks whether the level 1 mass air flow signal MAF is determined properly by the intake manifold model. In case that the value is correct it always exceeds a the level 2 value MAF_SUB_MON.

MAF_MON is equal to the high byte of the level 1 value MAF.

The variable MAF_MON is used for calculating the actual indicated engine torque TQI_AV_MON. Therefore it has to be checked that MAF_MON does not fall under the value of the physically present mass air flow. In case that the MAF_MON is smaller than the faultless one the torque monitoring function of the ETC safety concept (refer to the module "Monitoring of actual indicated engine torque" becomes less sensitive.

An additional TPS ratio check is performed. This function checks if any TPS failure has been detected in level 1 when a TPS ratio error is recognized in level 2.

Signal flow diagram:

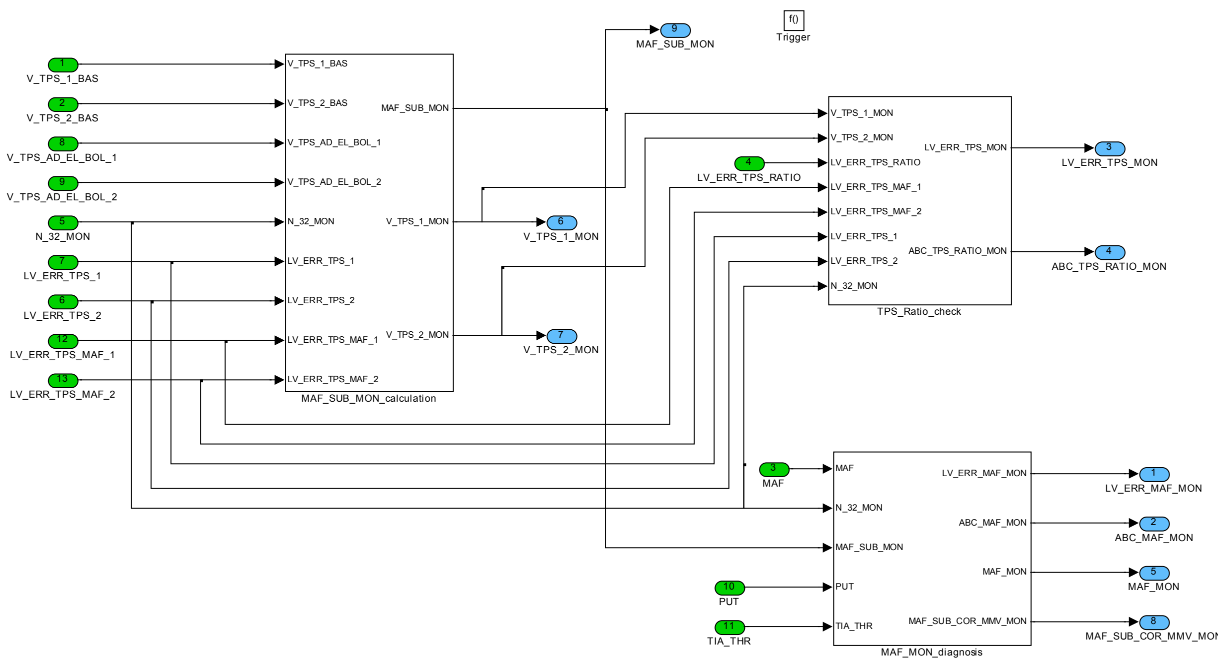


Figure 1: Monitoring of mass air flow signal (MAF_MON)Description:

Application conditions:

Activation:	*LV_TQI_MON_ACT_MON = 1*
Deactivation:	*otherwise*
Initialization:	*for condition see 'Application Incidences of Process Monitoring'*
	LV_ERR_TPS_MON = 0
	ABC_TPS_RATIO_MON = 0
	LV_ERR_MAF_MON = 0
	ABC_MAF_MON = 0
	MAF_MON = 0
	MAF_SUB_COR_MMV_MON = 0
	At deactivation:
	MAF_SUB_COR_MMV_MON(k) = 0
Update Rate:	*40 ms*

E.21.1 Mass air flow substitute (MAF_SUB_MON) calculation

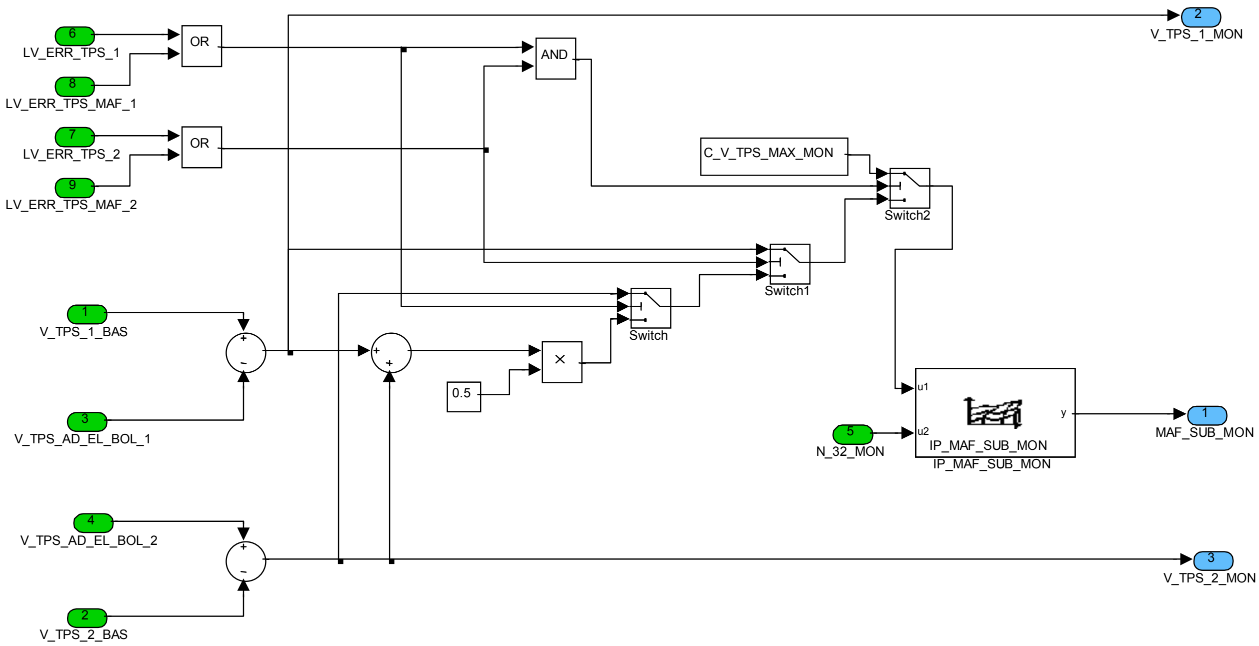


Figure 2: MAF_SUB_MON calculation

To reduce the tolerance of the MAF substitute signal MAF_SUB_MON the adapted voltage (V_TPS_AD_EL_BOL_x) of the lower limit of the TPS actuator is taken into consideration.

The TPS channel used for the mass air flow substitute calculation MAF_SUB_MON is selected according to the following table.

Error level	Error bits set	TPS channel used
0	-	AVERAGE(V_TPS_1_MON. V_TPS_2_MON)
1	LV_ERR_TPS_1 **or** LV_ERR_TPS_MAF_1	V_TPS_2_MON (selection)
2	LV_ERR_TPS_2 **or** LV_ERR_TPS_MAF_2	V_TPS_1_MON (selection)
3	(LV_ERR_TPS_1 **or** LV_ERR_TPS_MAF_1) and (LV_ERR_TPS_2 **or** LV_ERR_TPS_MAF_2)	TPS voltage set to: C_V_TPS_MAX_MON (max. voltage)

A higher error level always has higher priority to any lower error level: in case that LV_ERR_TPS_1 and LV_ERR_TPS_2 occur at the same time, the current TPS value is set to a substitute value C_V_TPS_MAX_MON. In case that level 1 has not detected any TPS failures, TPS is set to the mean average value calculated on basis of the measured TPS values V_TPS_1_MON and V_TPS_2_MON (refer to the module TPS_SEG).

E.21.2 MAF_MON diagnosis

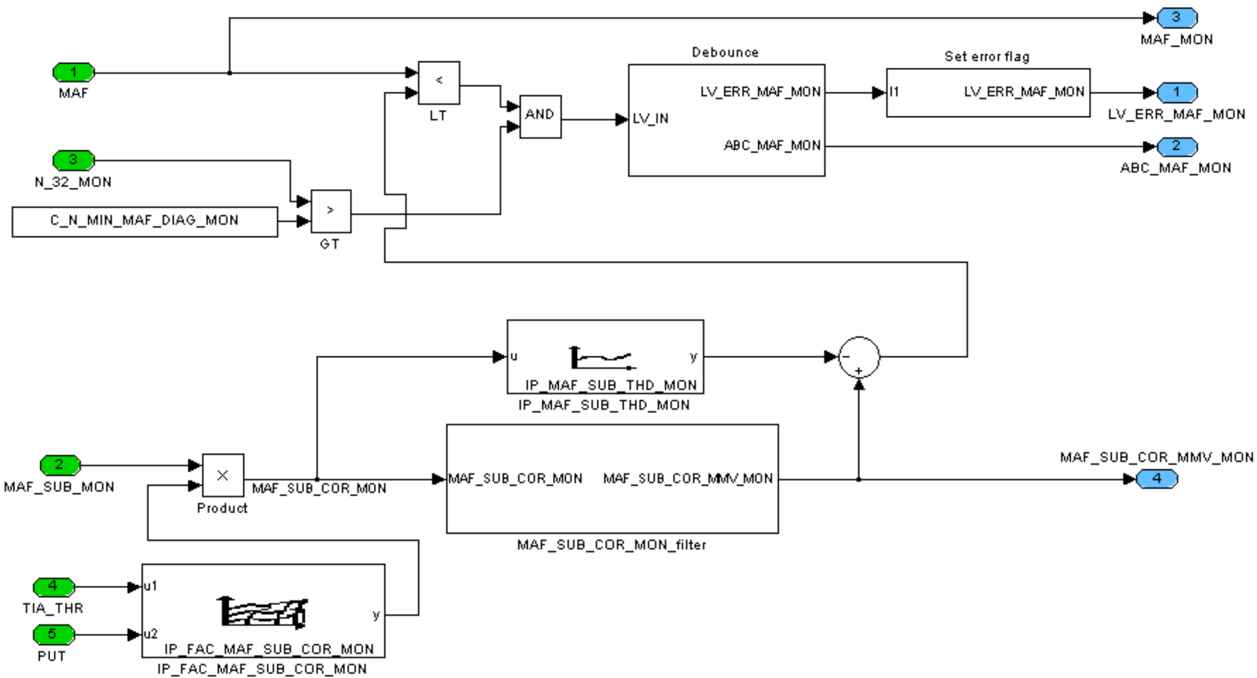


Figure 3: MAF_MON diagnosis

The high byte of the MAF signal being calculated by the intake-manifold-model is copied to MAF_MON for the use in further modules of the process monitoring (MAF_HB which is provided by the intake-manifold model as well is not used for the MAF monitoring to ensure that the level 1 functions and the related level 2 functions are working with consistent values).

For the diagnosis the MAF_MON signal is compared to a mass air flow substitute value MAF_SUB_COR_MMV_MON which is affected by the current engine speed N_32_MON, the current throttle position signal and ambient conditions (intake air temperature TIA_THR, pressure at throttle PUT). MAF_MON is not allowed to be smaller than a value MAF_SUB_COR_MMV_MON reduced by a correction value determined by IP_MAF_SUB_THD_MON.

The failure diagnosis becomes active in case that a certain engine speed has been reached to eliminate the inaccuracy both of the measured mass air flow meter in case of low engine speed and the inaccuracy of the reduced map (IP_MAF_SUB_MON).

To adapt the calculated signal to the dynamics of the intake manifold system (the air mass flow) a low pass filter for rising signals has been installed.

Formula section

The MAF signal is copied to the process monitoring (Level 2) RAM:

MAF_MON = MAF

MAF_SUB_COR_MON = MAF_SUB_MON • IP_FAC_MAF_SUB_COR_MON(TIA_THR.PUT)

BLOCK 1: MAF_SUB_COR_MON filter:

IF

(MAF_SUB_COR_MON > MAF_SUB_COR_MMV_MON(k-1))

THEN

MAF_SUB_COR_MMV_MON(k) =
MAF_SUB_COR_MMV_MON(k-1) * (1-C_MAF_SUB_CRLC_MON)
+ MAF_SUB_COR_MON(k) * C_MAF_SUB_CRLC_MON

ELSE

MAF_SUB_COR_MMV_MON(k) = MAF_SUB_COR_MON

Error recognition:

IF

(N_32_MON > C_N_MIN_MAF_DIAG_MON)
AND
(MAF_MON < (MAF_SUB_COR_MMV_MON – IP_MAF_SUB_THD_MON))

THEN

After debounce. a MAF_MON error is recognized and

LV_ERR_MAF_MON is set.

Increment: C_ABC_INC_MAF_MON

Maximum value: C_ABC_MAX_MAF_MON

E.21.3 TPS ratio check

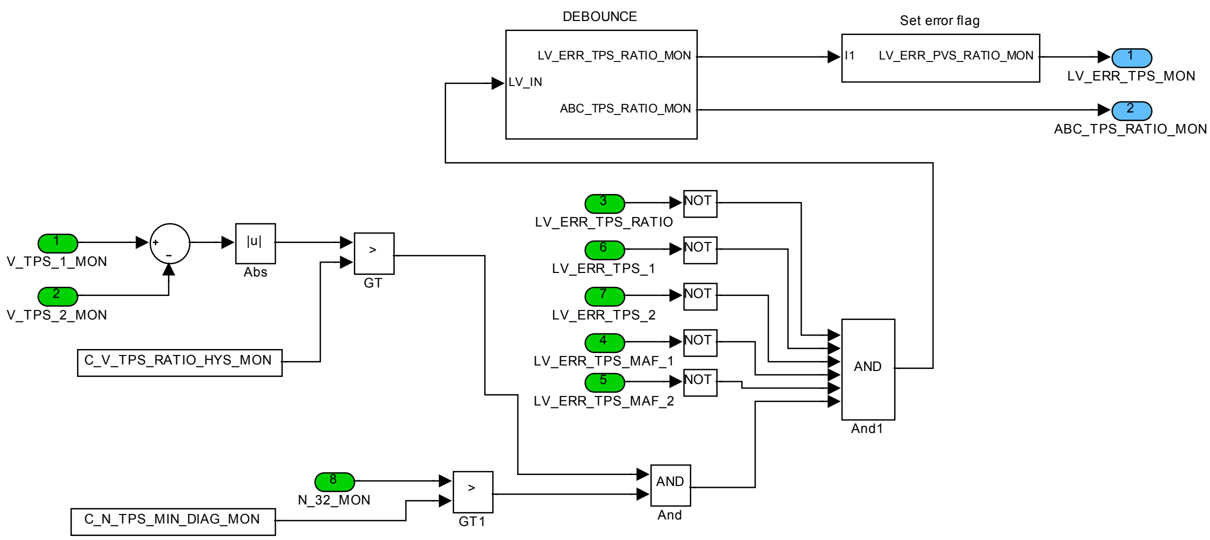


Figure 4: TPS ratio check

In a properly working system the difference between TPS channel 1 and TPS channel 2 is not allowed to exceed the threshold C_V_TPS_RATIO_HYS_MON.

Formula section:

Error recognition:

IF

(ABS(V_TPS_1_MON - V_TPS_2_MON) > C_V_TPS_RATIO_HYS_MON) AND
(N_32_MON > C_N_MIN_TPS_DIAG_MON) AND
(LV_ERR_TPS_RATIO = 0) AND
(LV_ERR_TPS_1 = 0) AND
(LV_ERR_TPS_2 = 0) AND
(LV_ERR_TPS_MAF_1 = 0) AND
(LV_ERR_TPS_MAF_2 = 0)

THEN

After debounce. LV_ERR_TPS_MON is set.

Increment: C_ABC_INC_TPS_RATIO_MON

Maximum value: C_ABC_MAX_TPS_RATIO_MON

Level 1 is assumed to detect a TPS ratio error earlier than Level 2. In case that error detection of level 1 fails a process monitoring fault is assumed and the error bit LV_ERR_TPS_MON is set.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_V_TPS_MAX_MON	1	0...FFH	0...4.98	0.01953	V
Maximum nominal voltage of the TPS fully opened					
C_V_TPS_RATIO_HYS_MON	1	0...FFH	0...4.98	0.01953	V
Maximum TPS ratio error allowed					
C_ABC_INC_TPS_RATIO_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment (additive value in case of ratio diagnosis error)					
C_ABC_MAX_TPS_RATIO_MON	1	1...FFH	1...255	1	-
Value at which ratio diagnosis error is recognized. when reached					
C_N_MIN_TPS_DIAG_MON	1	0...FFH	0...8.16E3	32	rpm
Minimum engine speed for diagnosis of TPS ratio error					
C_ABC_INC_MAF_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment (additive value in case of MAF diagnosis error)					
C_ABC_MAX_MAF_MON	1	1...FFH	1...255	1	-
Value at which MAF diagnosis error is recognized. when reached					
C_N_MIN_MAF_DIAG_MON	1	0...FFH	0...8.16E3	32	rpm
Minimum engine speed threshold for diagnosis of the MAF signal					
C_MAF_SUB_CRLC_MON	1	0...FFH	0...0.9960938	0.003906	-
Correlation constant for MAF_SUB_MON filter					
IP_MAF_SUB_THD_MON	2	0...FFH	0...1.389E3	5.447	mg/stk
LDP_MAF_SUB_IP_MAF_SUB_THD_MON	2	0...FFH	0...1.389E3	5.447	mg/stk
Threshold for MAF_MON error detection					
IP_MAF_SUB_MON	8 x 8	0...FFH	0...1.389E3	5.447	mg/stk
LDP_N_32_MON_IP_MAF_SUB_MON	8	0...FFH	0...8.16E3	32	rpm
LDP_V_TPS_IP_MAF_SUB_MON	8	0...FFH	0...4.98	0.01953	V
MAF substitute					
IP_FAC_MAF_SUB_COR_MON	6 x 4	0...FFH	0...1.992188	0.007813	-
LDP_TIA_IP_FAC_MAF_SUB_COR_MON	6	0...FEH	-48...142.5	0.75	°C
LDP_PUT_IP_FAC_MAF_SUB_COR_MON	4	0...FFFFH	0...5.434E3	0.08292	hPa
Correction factor for MAF_SUB_COR_MON calculation					

E.22 Monitoring of cruise control conditions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ABC_CRU_INH_MON	V	0...FFH	0...255	1	[-]
Anti bounce counter for fault in cruise control conditions present					
FAC_TQ_REQ_CRU_MON	O/V	0...FFH	0...1.9922	0.0078125	[-]
Charge demand from cruise control					
LV_CRU_INH_MON	O/V	0...1H	0...1	1	[-]
Logical variable for inhibition of cruise control					
T_CRU_INH_MON	O/V	0...FFH	0...10.2	0.04	[s]
Timer for inhibition of cruise control charge demand in monitoring level					

Input data:

FAC_TQ_REQ_CRU	LV_BRAKE_MON	LV_CRU_ACT	VS
LV_TQI_MON_ACT_MON	LV_CRU_MAIN_SWI	CONF_CRU	

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_ABC_INC_CRU_INH_MON	1	0 ... FF H	0 ... 255	1	-
Anti bounce counter increment					
C_ABC_MAX_CRU_INH_MON	1	0 ... FF H	0 ... 255	1	-
Maximum value of anti-bounce counter					
C_T_CRU_INH_MON	1	0 ... FF H	0 ... 10.2	0.04	s
Delay time for disable of cruise control charge demand in monitoring level					
C_VS_MIN_CRU_MON	1	0 ... FF H	0 ... 255	2	km/h
Minimum threshold for vehicle speed control active					

FUNCTION DESCRIPTION:

General information:

The cruise control function is able to request a lot of torque and to accelerate the car up to high speed. In order to avoid dangerous situations caused by a fault of the cruise control function, the main activation/deactivation conditions are monitored within the etc safety concept. Therefore the basic signals from the main switch and the brake are evaluated.

Application conditions:

Activation: LV_TQI_MON_ACT_MON = 1 and CONF_CRU = 1

Deactivation: otherwise

At deactivation, FAC_TQ_REQ_CRU_MON = 0

Initialisation (Must be done before activation of the function):

LV_CRU_INH_MON = 0

ABC_CRU_INH_MON = 0

T_CRU_INH_MON = 0

FAC_TQ_REQ_CRU_MON = 0

Update Rate: 40 ms

Formula section:

FAC_TQ_REQ_CRU_MON = FAC_TQ_REQ_CRU

IF LV_CRU_ACT = 1

THEN

  IF LV_CRU_INH_MON = 0

  THEN

    Initialize Timer: T_CRU_INH_MON = C_T_CRU_INH_MON

    IF LV_CRU_MAIN_SWI = 0 **OR**

      LV_BRAKE_MON = 1 **OR**

      VS < C_VS_MIN_CRU_MON

    THEN after debouncing with ABC_CRU_MON

      Inhibit cruise control non reversible until the initialization conditions for error bits are fulfilled (see application incidences for process monitoring)

      LV_CRU_INH_MON = 1

      FAC_TQ_REQ_CRU_MON = 0

    ENDIF

ELSE

  IF T_CRU_INH_MON = 0

  THEN

    FAC_TQ_REQ_CRU_MON = 0

  ELSE

    Decrement Timer T_CRU_INH_MON

  ENDIF

ENDIF

Anti-bounce counter: ABC_CRU_INH_MON

Anti-bounce counter increment: C_ABC_INC_CRU_INH_MON

Anti-bounce counter maximum: C_ABC_MAX_CRU_INH_MON

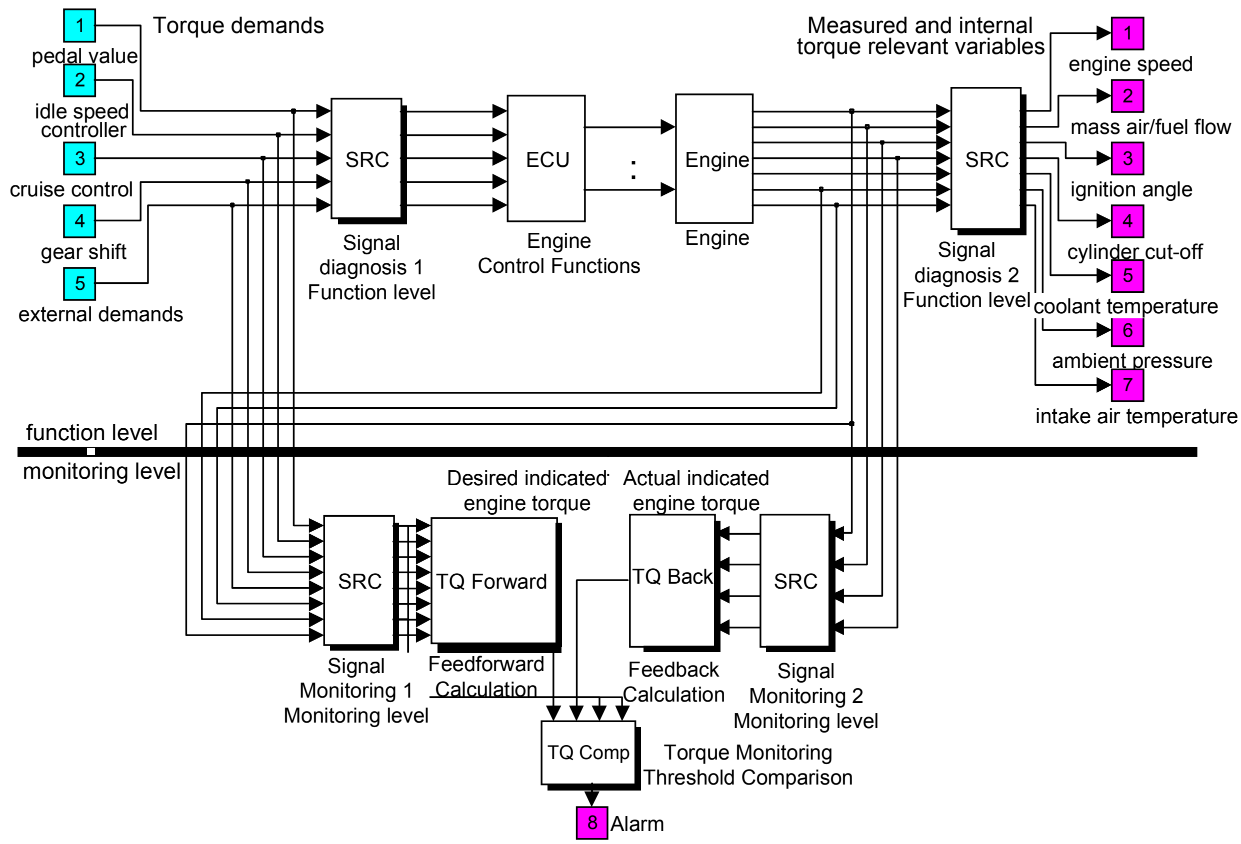
E.23 Torque Monitoring Overview

In the function level the various torque demands coming from the driver, the idle speed controller etc. are co-ordinated to a resulting quantfied charge setpoint. At the same time the actual indicated engine torque is calculated based on several measured and internal torque relevant variables.

The aim of the torque monitoring is to detect an increased actual indicated engine torque compared to the desired indicated engine torque. Therefore the co-ordination of the various torque demands and the calculation of the actual indicated engine torque done in the function level (level 1) is performed once again in the monitoring level (level 2), however with a simplified structure. Then the deviation between the actual (TQI_AV_MON) and the desired indicated engine torque (TQI_SP_MON) is checked for exceeding a threshold.

Signal flow diagram:

Overview



E.24 Desired indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_REQ_DRIV_MON	O/V	0...FFH	0...1.992	0.0078125	-
scaling factor for requested torque at clutch from driver					
FAC_TQ_REQ_MON	O/V	0...FFH	0...1.992	0.0078125	-
scaling factor for requested torque at clutch from driver and cruise control					
TQI_SP_MON	O/V	0...FFH	0...510	2	Nm
desired indicated engine torque					
TQI_REQ_TOT_MON	V	0...FFH	0...510	2	Nm
desired non-filtered indicated engine torque					
T_TQI_MON	V	0...FFH	0...10.2	0.04	s
dead time counter for torque decrease					

Input data:

PV_AV_AD_MON	TQI_INC_EXT_MON	TQI_MIN_PU_MON	TQ_DIF_P_D_IS_MON
TQ_MAX_CLU_MON	TQ_MIN_CLU_MON	FAC_TQ_REQ_CRU_MON	LV_TQI_MON_ACT_MON
N_32_MON	TQ_LOSS_MON		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_TQI_MON	1	0...FFH	0...10.2	0.04	s
Dead time for torque decrease					
C_FIL_TQI_SP_INP_0_MON	1	8000...7FFFH	-1...0.99994	3.05e-5	-
filter constant for realization of first order filter					
C_FIL_TQI_SP_INP_1_MON	1	8000...7FFFH	-1...0.99994	3.05e-5	-
filter constant for realization of first order filter					
C_FIL_TQI_SP_OUT_1_MON	1	8000...7FFFH	-1...0.99994	3.05e-5	-
filter constant for realization of first order filter					
IP_FAC_TQ_REQ_DRIV_MON	8x8	0...FFH	0...1.992	0.0078125	-
LDP_N_32_MON_IP_FAC_TQ_MON	8	0...FFH	0...8160	32	rpm
LDP_PV_AV_AD_MON_IP_FAC_MON	8	0...FFH	0...99.6	0.39	%
scaling factor for requested torque at clutch					

E.24.1 ECM2_DTSYSTQISP

Import actions:

ACTION_ECM3_Service4TaskPfm(4)
ACTION_ECM3_Service5TaskPfm(4)

Note: This actions are defined in chapter "Processor Monitroing", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 4.

ACTION_ECM3_WriteChkCpl(IN<type>,INOUT<xyz_MON>,OUT<xyz_MON_CPL>,IN<reg>)
ACTION_ECM3_ChkCpl(IN<type>, IN<xyz_MON>, IN<xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN<type>,OUT<reg>,IN<xyz_MON>, IN<xyz_MON_CPL>)

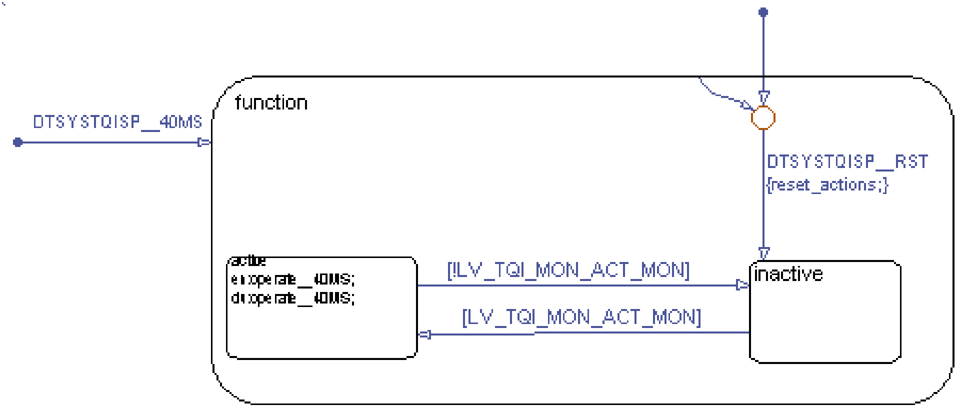
Note: This actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

General information:

The objective of this module is to calculate the desired indicated engine torque which derives from the function level (level 1). All torque increasing demands are taken into account. The differents parts are:

- calculation of scaling factor for requested torque at clutch from driver(FAC_TQ_REQ_MON)
- depending on engine speed(N_32_MON) and adapted accelerator pedal activation degree (PV_AV_AD_MON)
- calculation of requested torque at clutch from driver
- adding internal increasing torque demands (TQ_DIF_P_D_IS_MON)
- switch from brake torque to indicated torque demands
- filtering the raw torque demand in falling direction to take into account the system response time (see also figure 1)
- maximum choice between internal and external torque demands
- If the engine is in the trailing throttle state i.e. no driver demand but limitation on indicated engine torque to ensure a safe combustion at low load conditions, then the torque setpoint in L2 is limited by TQI_MIN_PU_MON.

Application Condition



Initialisation(DTSYSTQISP__RST)**:** for condition see 'Application Incidences of Process Monitoring'.

Function Description

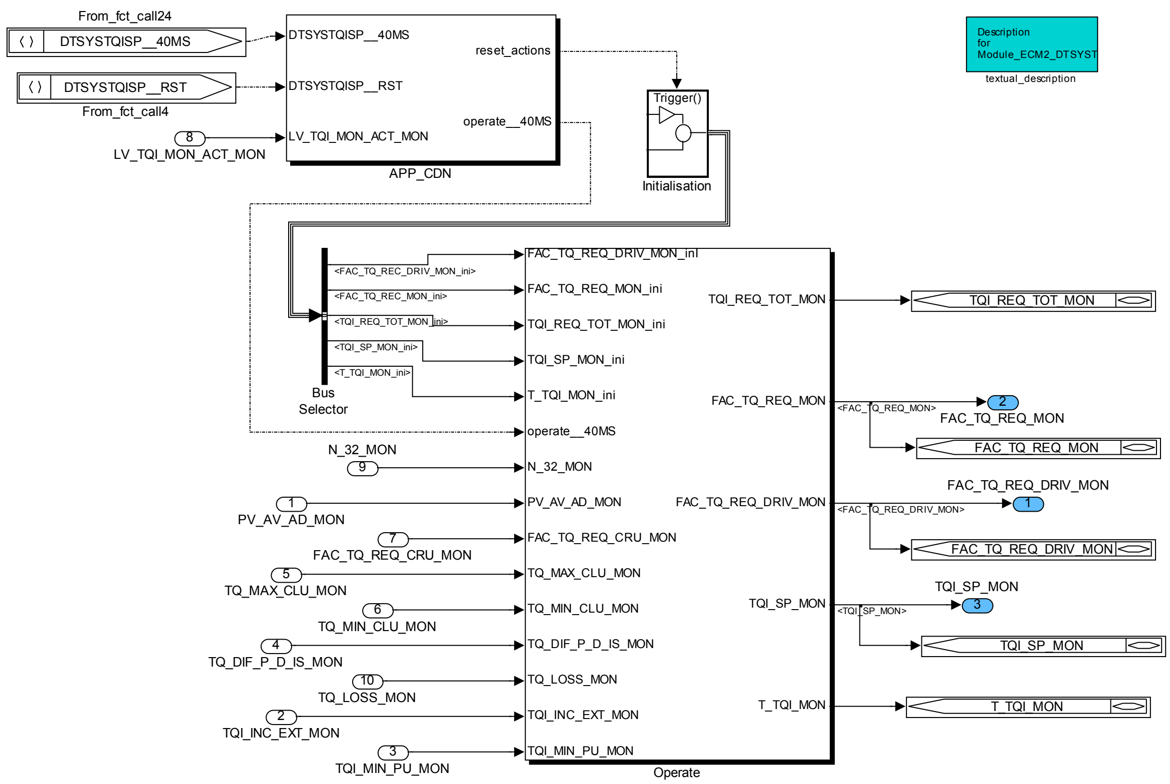


Figure 61 ECM2_DTSYSTQISP

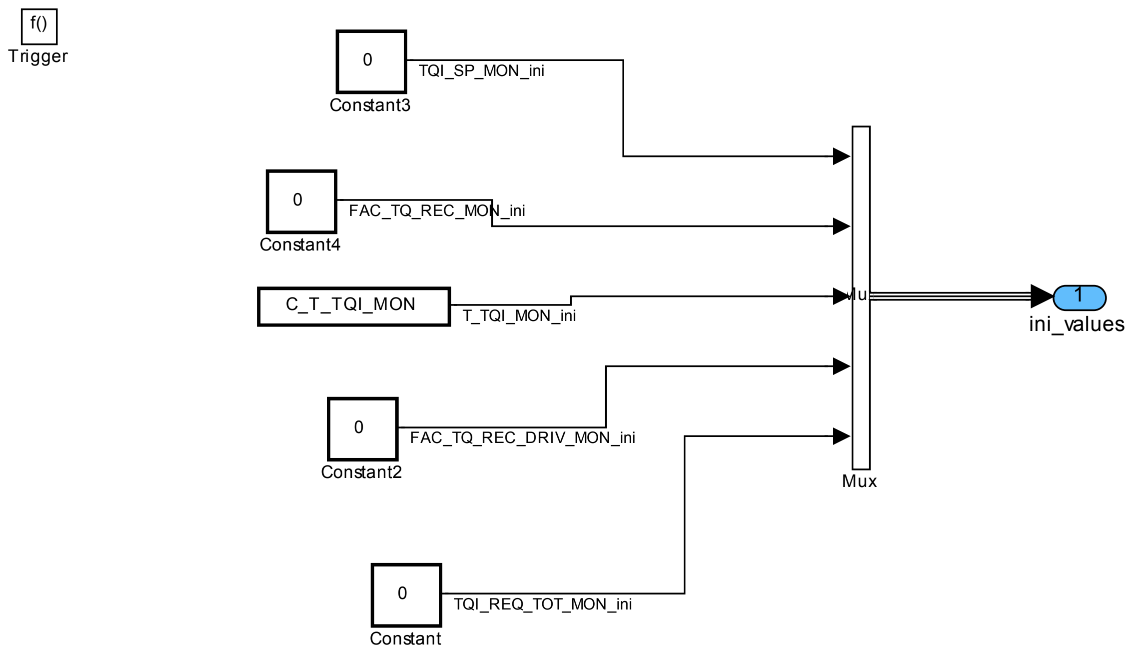


Figure 62 ECM2_DTSYSTQISP/ Initialisation

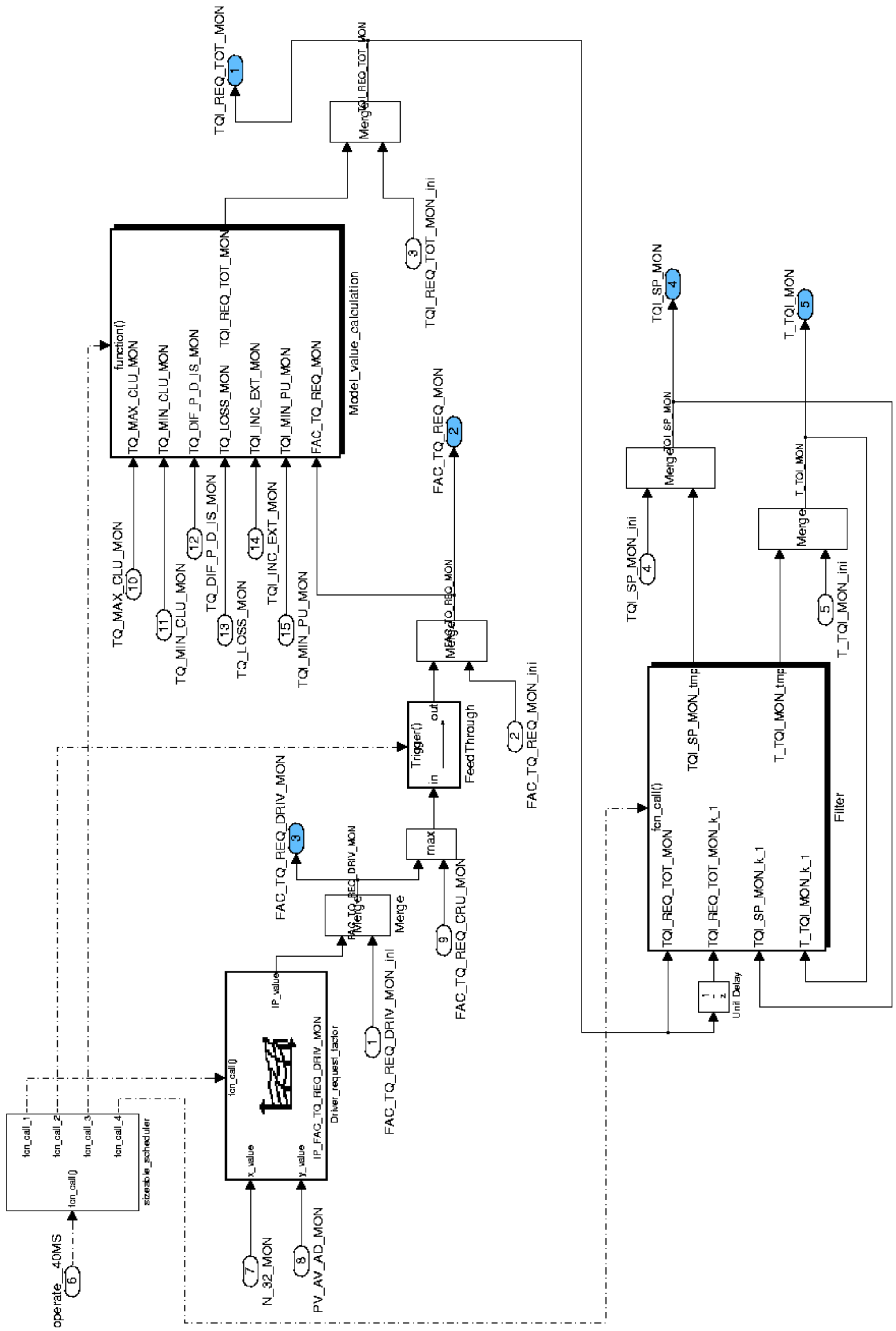


Figure 63 ECM2_DTSYSTQISP/ Operate

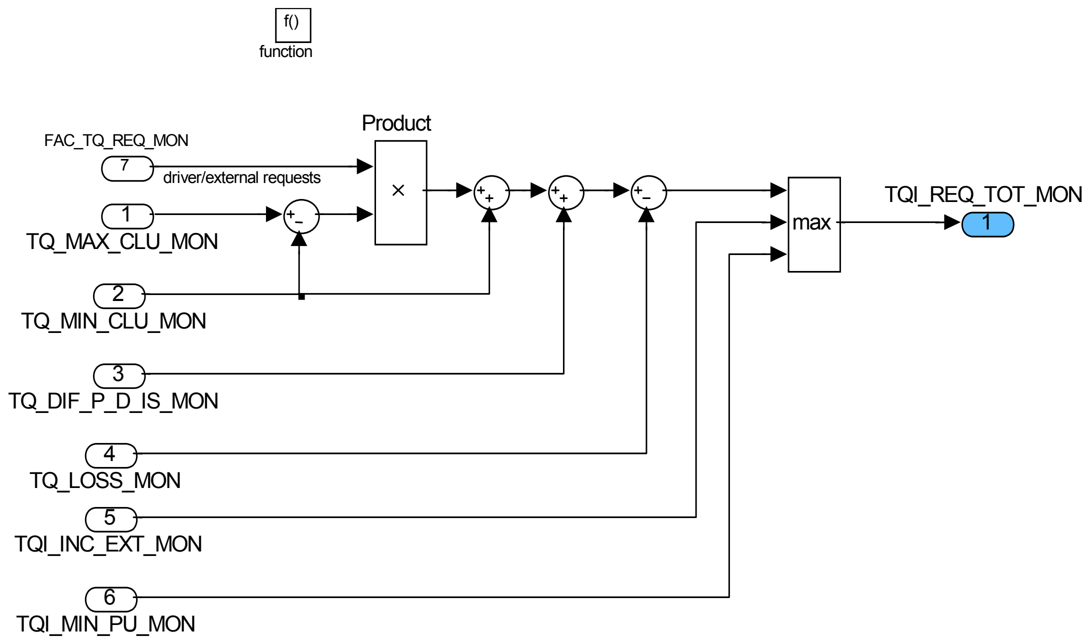


Figure 64 ECM2_DTSYSTQISP/ Operate/ Model_value_calculation

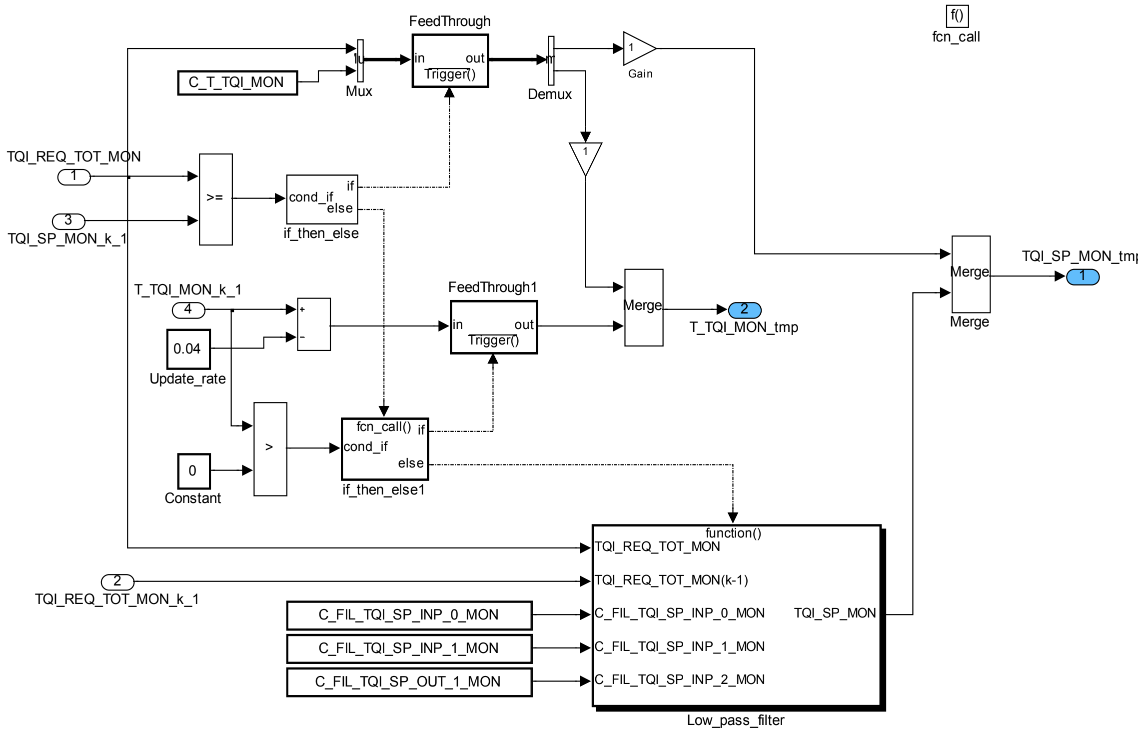


Figure 65 ECM2_DTSYSTQISP/ Operate/ Filter

E.25 Actual indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TQI_AV_MON	V/O	0 ... FF H	0 ... 510	2	Nm
actual indicated engine torque (process monitoring)					
EFF_SCC_MON	V/O	0 ... FF H	0 ... 1.992156	7.843e-3	-
actual efficiency of cylinder cut-off for torque monitoring					
TQI_REF_MON	V	0 ... FF H	0 ... 510	2	Nm
reference indicated engine torque (process monitoring)					
EFF_IGA_MON	V	0 ... FF H	0 ... 1.992156	7.843e-3	-
actual ignition efficiency for torque monitoring					
IGA_DIF_AV_MON	V	0 ... FF H	0 ... –95.625	0.375	°CRK
difference of IGA_AV_MV_MON and IGA_REF_COR_MON					

Input data:

N_32_MON	MAF_MON	IGA_IGC_x	IGA_REF_COR_EXT_MON
PREV_STATE_IV	SUM_INH_INJ	LV_TQI_MON_ACT_MON	

Import actions:

ACTION_ECM3_Service2TaskPfm(4)
ACTION_ECM3_Service3TaskPfm(4)

Note: This actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 4.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
IP_TQI_REF_MON	6x8	0 ... FF H	0 ... 510	2	Nm
LDP_N_32_MON_IP_TQI_REF_MON	6	0 ... FF H	0 ... 8160	32	rpm
LDP_MAF_MON_IP_TQI_REF_MON	8	0 ... FF H	0 ... 1389	5.43	mg/stk
Indicated engine torque at reference conditions (process monitoring)					
IP_IGA_REF_MON	8x8	0 ... FF H	-35.625 ...60	0.375	°CRK
LDP_N_32_MON_IP_IGA_REF_MON	8	0 ... FF H	0 ... 8160	32	rpm
LDP_MAF_MON_IP_IGA_REF_MON	8	0 ... FF H	0 ... 1389	5.43	mg/stk
Reference ignition angle (process monitoring)					
IP_EFF_IGA_MON	8	0 ... FF H	0 ... 1.992156	7.843e-3	-
LDP_IGA_DIF_AV_MON_IP_EFF_MON	8	0 ... FF H	0 ... –95.625	0.375	°CRK

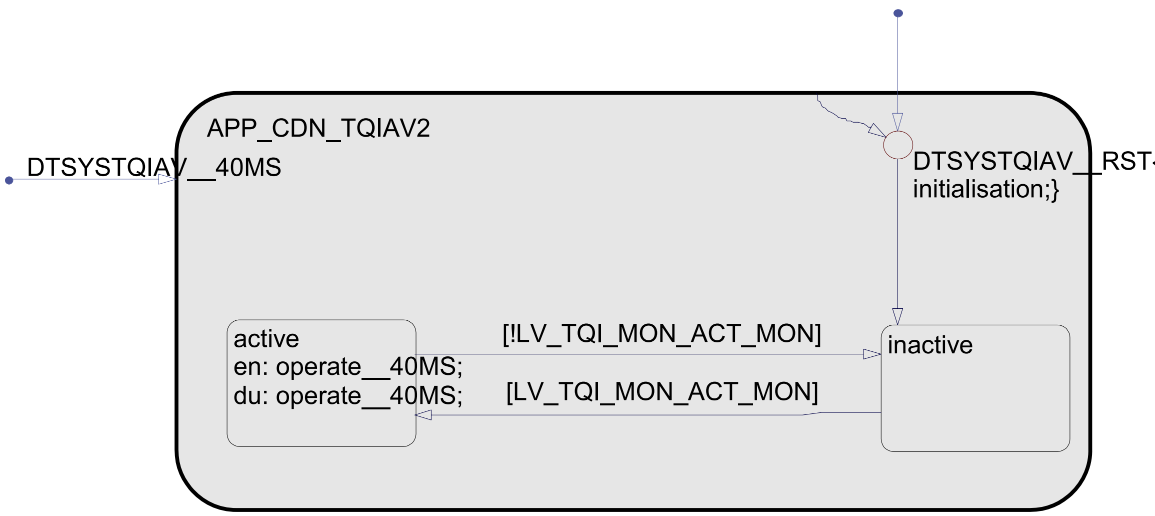
Basic ignition efficiency (process monitoring)

General information:

This module calculates the actual indicated engine torque TQI_AV_MON. Therefore the indicated engine torque TQI_REF_MON at reference conditions is corrected by the mean value IGA_AV_MV_MON of the actual ignition angles IGA_IGC_x. The maps and characteristic curves for reference torque and IGA efficiency represent simplified versions of the original calibration data of the corresponding functions in the control level. Additionally, the efficiency for cylinder cut-off is calculated from severe injection state information and is taken into account.

The calculation of the intermediate value EFF_SCC_1_MON is based on the function level. To monitor the input PREV_STATE_IV, the SUM_INH_INJ as the number for inhibited cylinders is used in EFF_SCC_2_MON. The maximum selection results into SCC – efficiency. To fade out invalid states during engine stop, efficiency calculation only works at engine speed above 32 rpm.

Application conditions



Note: DTSYSTQIAV__RST includes the function calls as defined in application incidences.

Function Description

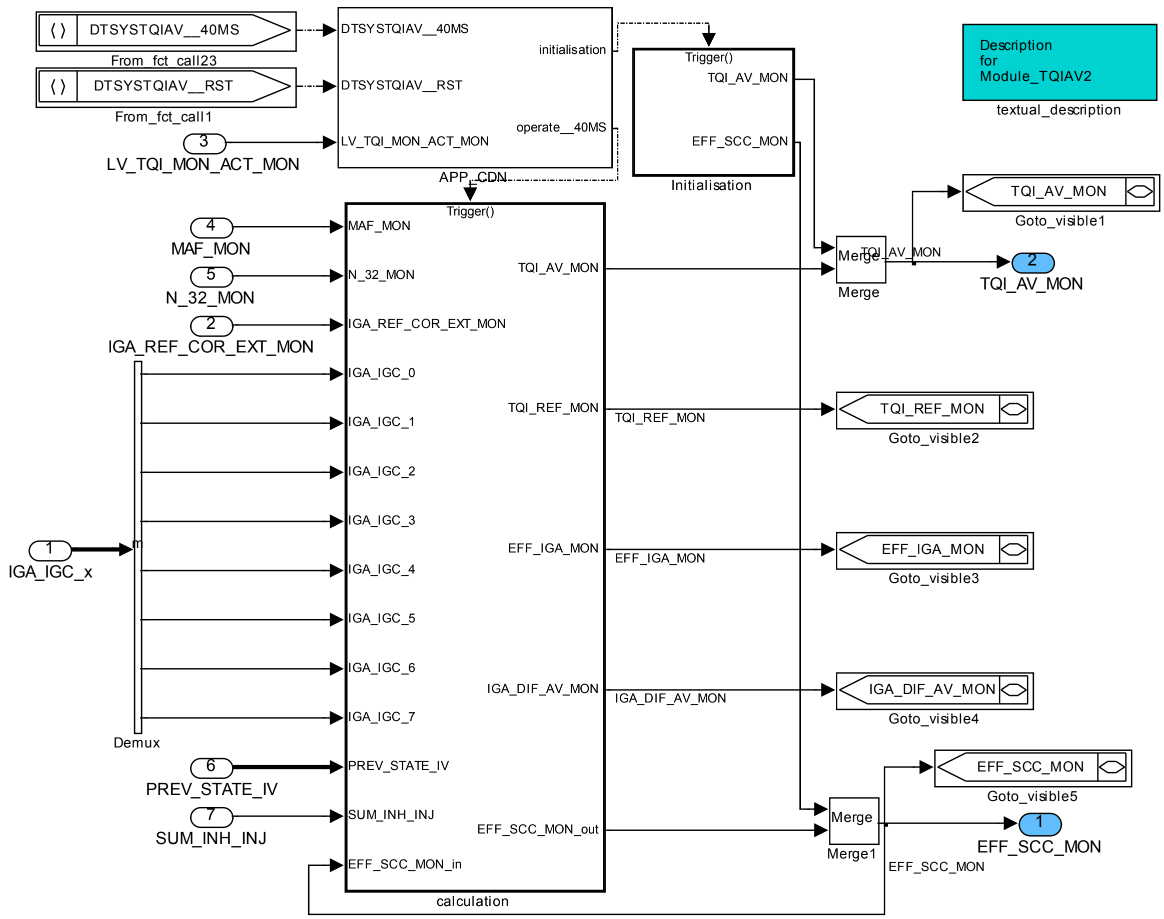


Figure 66 TQIAV2

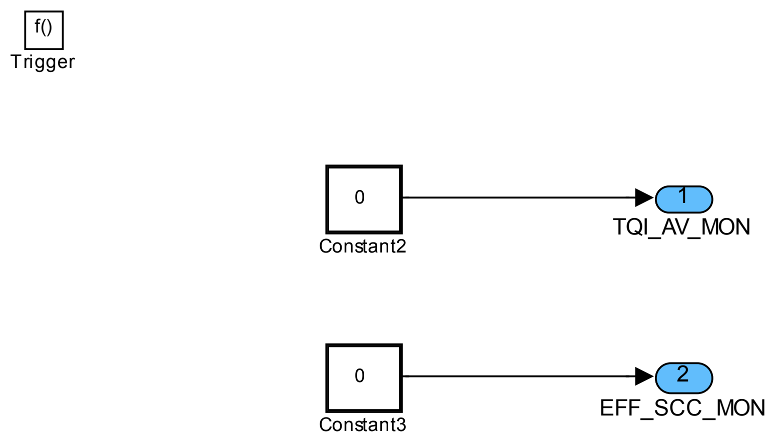


Figure 67 TQIAV2/ Initialisation

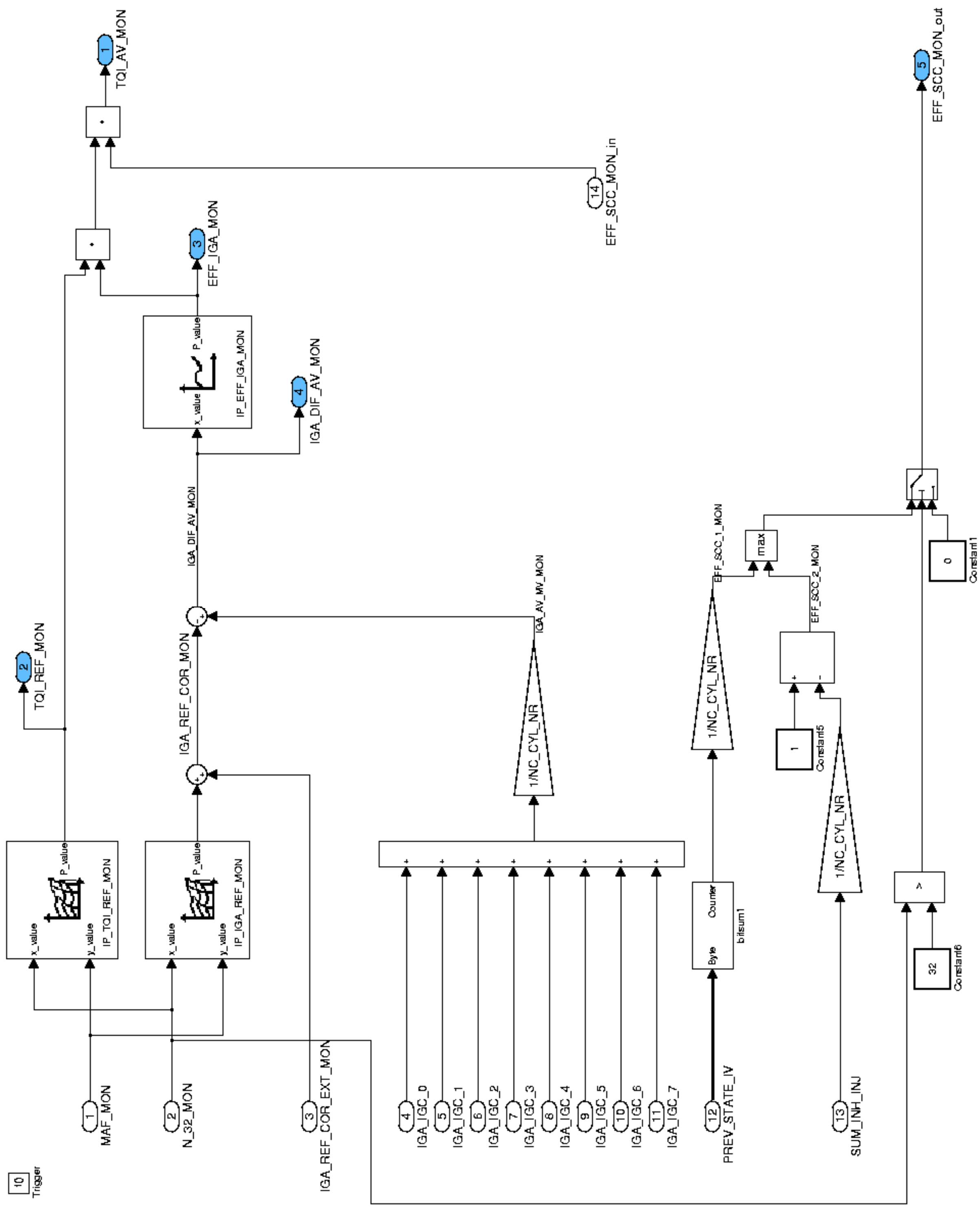


Figure 68 TQIAV2/ calculation

E.26 Monitoring of actual indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_TQI_AV_MON	V/O	0 ... 1 H	0 ... 1	1	-
Fault currently present in torque generation					
TQI_DIF_AV_MON	V	0 ... FF H	0 ... 510	2	Nm
Deviation between actual and desired indicated engine torque					
TQI_DIF_MAX_MON	V	0 ... FF H	0 ... 510	2	Nm
Permitted deviation between actual and desired indicated engine torque					
ABC_TQI_AV_MON	V	0 ... FF H	0 ... 255	1	-
Anti-bounce-counter for fault in torque generation					
TQI_DIF_SP_MON	V	0 ... FF H	0 ... 510	2	Nm
Deviation between the L1 and L2 actual indicated engine torque					

Input data:

TQI_SP_MON	TQI_AV_MON	N_32_MON	LV_TQI_MON_ACT_MON
TQI_SP			

Import actions:

ACTION_ECM3_Service6TaskPfm(4)
ACTION_ECM3_Service7TaskPfm(4)

Note: This actions are defined in chapter "Processor Monitroing", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 4.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

Calibration data:

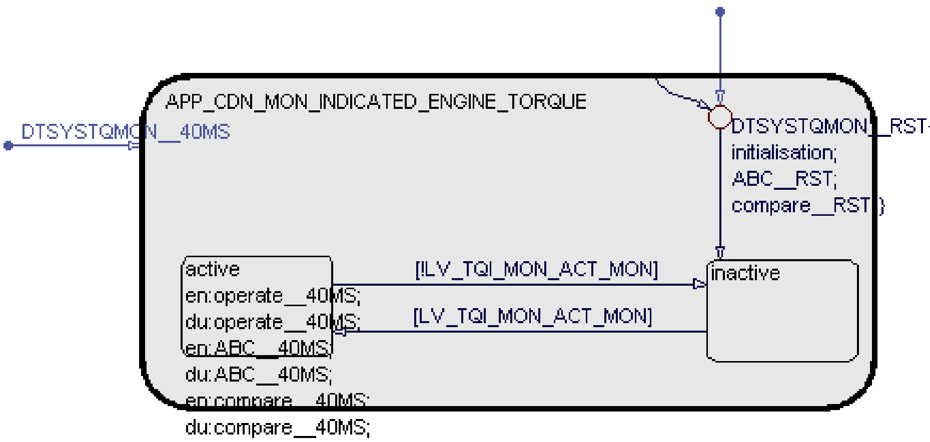
Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_DIF_MAX_MON	4x4	0 ... FF H	0 ... 510	2	Nm
LDP_N_32_IP_TQI_DIF_MAX_MON	4	0 ... FF H	0 ... 8160	32	rpm
LDP_TQI_SP_IP_TQI_DIF_MAX_MON	4	0 ... FF H	0 ... 510	2	Nm
Threshold for diagnosis of engine torque overflow					
C_GAIN_TQI_DIF_MAX_MON	1	0 ... FF H	0 ... 63.75	0.25	-
Factor for threshold diagnosis with TQI_SP					
C_ABC_INC_TQI_AV_MON	1	0 ... FF H	0 ... 255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQI_AV_MON	1	1 ... FF H	1 ... 255	1	-
Maximum value of anti-bounce counter					

General information:

The objective of this module is the comparison of the desired indicated engine torque TQI_SP_MON (feed-forward path) and the actual indicated engine torque TQI_AV_MON (feedback path). In the fault free case, the actual indicated engine torque should not exceed the desired indicated engine torque. A fault is detected after debouncing, when the actual indicated engine torque exceeds the setpoint value by more than a threshold value TQI_DIF_MAX_MON depending on engine speed and the desired torque.

TQI_AV_MON depends on the dynamics of the intake manifold and on the dynamics of the lambda sensor which means that a possible fault cannot be detected before a certain time. Hence for a fast diagnosis TQI_AV_MON is not suitable but can be replaced by TQI_SP which is compared to TQI_SP_MON. In case that TQI_SP_MON differs from TQI_SP by more than a certain value the error bit LV_ERR_TQI_AV_MON is set.

Application conditions



Note: DTSYSTQMON__RST includes the function calls as defined in application incidences.

Function Description

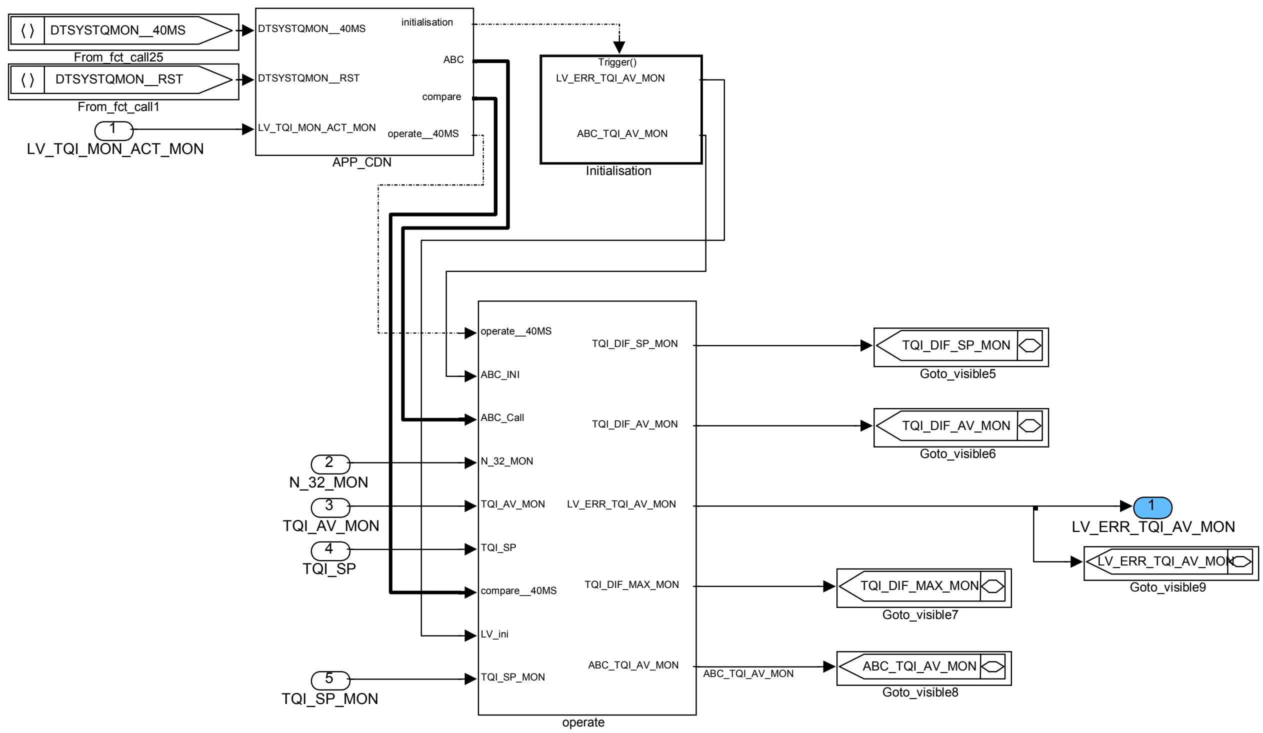


Figure 69 TQMON2

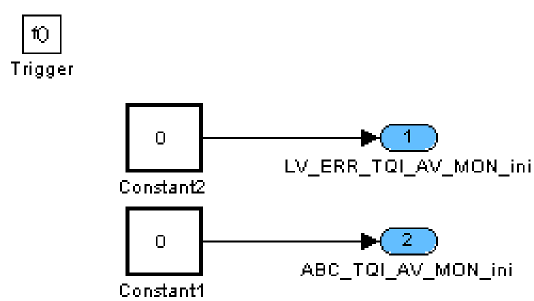


Figure 70 TQMON2/ INITIALISATION

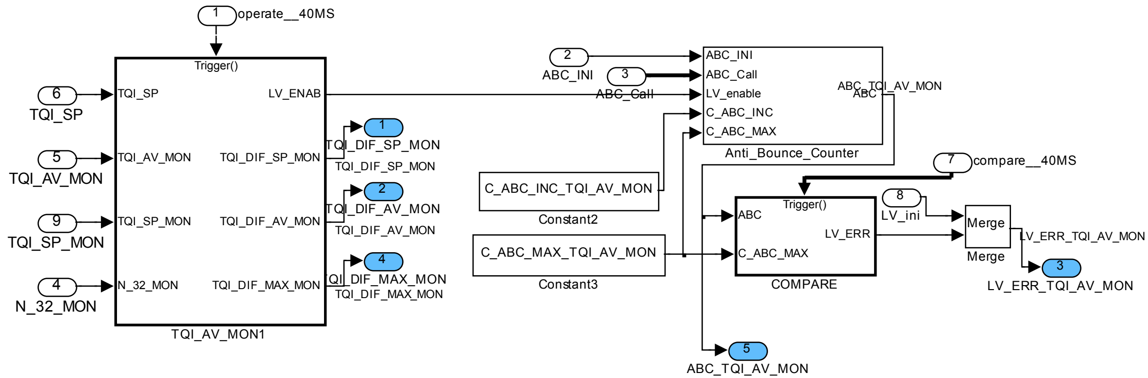


Figure 71 TQMON2/ OPERATE

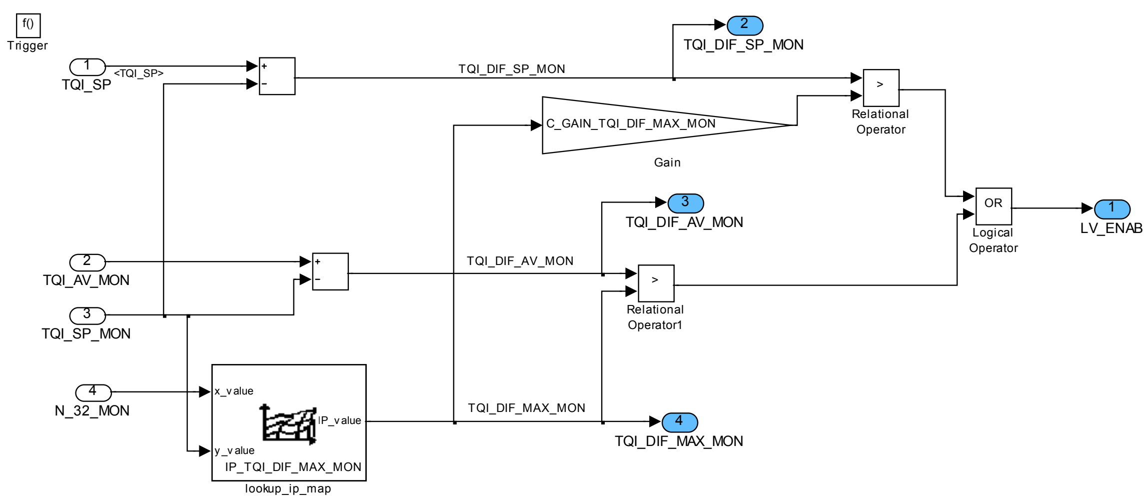


Figure 4 TQMON2/ operate/ TQI_AV_MON1

E.27 Monitoring of engine speed limitation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQI_N_MAX_MON	O/V	0...1H	0...1	1	-
Fault currently present in engine speed limitation					
ABC_TQI_N_MAX_MON	V	0...FFH	0...255	1	-
Anti-bounce-counter for fault in engine speed limitation					

Input data:

LV_N_LIM_ETC_MON	N_32_MON	PREV_STATE_IV	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ABC_INC_TQI_N_MAX_MON	1	0...FFH	0...255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQI_N_MAX_MON	1	1...FFH	1...255	1	-
Maximum value of anti-bounce counter					
C_N_MAX_MTC_LIH_THD_MON	1	0...FFH	0...8160	32	rpm
Speed threshold for the detection of a fault in the engine speed limitation					

Import actions:

ACTION_ECM3_Service2TaskPfm(3)
ACTION_ECM3_Service3TaskPfm(3)

Note: This actions are defined in chapter "Processor Monitroing", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 3.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitroing", subsection "RAM test on MC for processor monitoring". All variables with Mode "O" and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

FUNCTION DESCRIPTION:

General information:

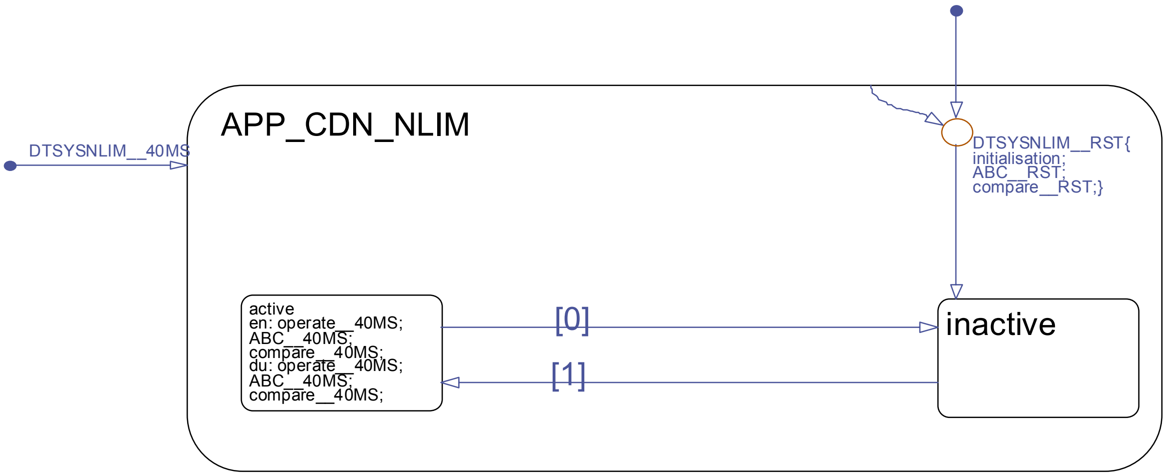
The objective of this module is to verify that the related level 1 function 'Engine speed limitation' works correctly. In case that the engine speed exceeds a certain speed level, the level 1 function activates ETC limp home mode (safety fuel cut-off) to protect the engine.

The level 2 function 'Monitoring of engine speed limitation' considers the status of the injection of all cylinders and the engine speed value and the ETC request flag (LV_N_LIM_ETC_MON). This flag is set to 1 if any function requests limp home activation.

In case that the engine speed exceeds the defined level 2 value C_N_MAX_MTC_LIH_THD_MON and limphome mode is requested (LV_N_LIM_ETC_MON = 1) and injection is active in at least one cylinder (PREV_STATE_IV ≠ 0; fuel cut-off is not active!), the related error bit LV_ERR_TQI_N_MAX_MON is set after having waited a certain debounce time defined by C_ABC_MAX_TQI_N_MAX_MON and C_ABC_INC_TQI_N_MAX_MON.
The injection status has to be checked to avoid that level 2 detects a failure in case that the engine speed exceeds the threshold although fuel cut-off has been put into action. This could happen if the vehicle is running down a hill and the engine speed increases because of the engine directly linked to the transmission.

E.27.1 FUNCTION PART: NLIM

Application Condition



Note: DTSYSNLIM__RST includes the function calls as defined in application incidences.

Function Description

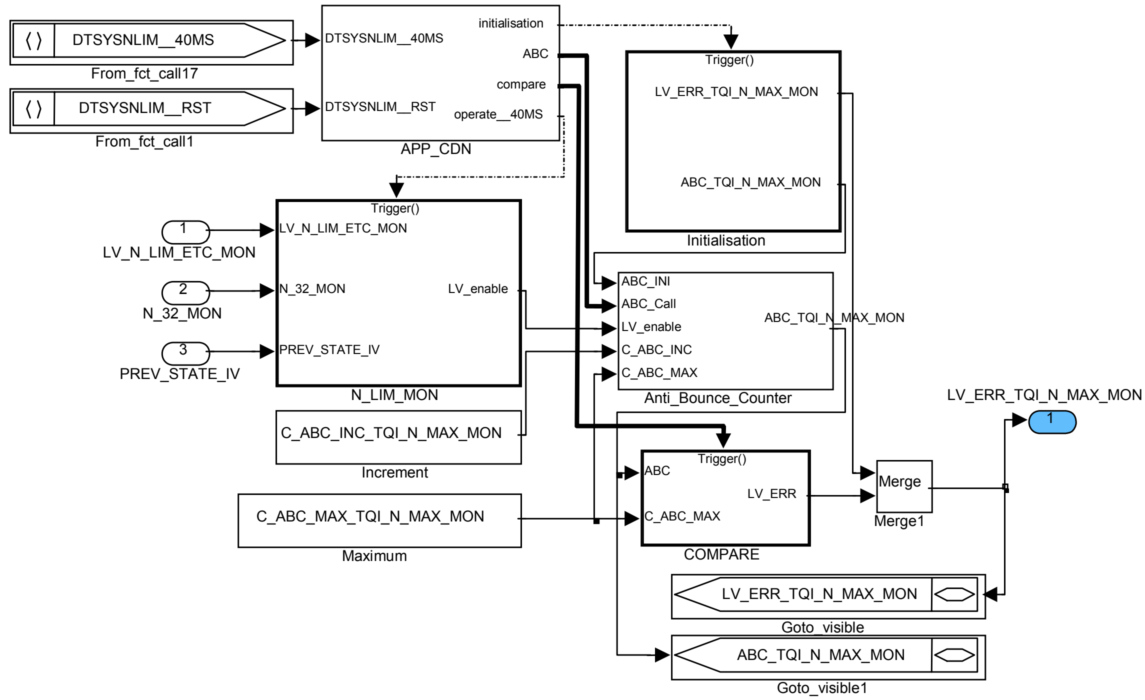


Figure 72 NLIM

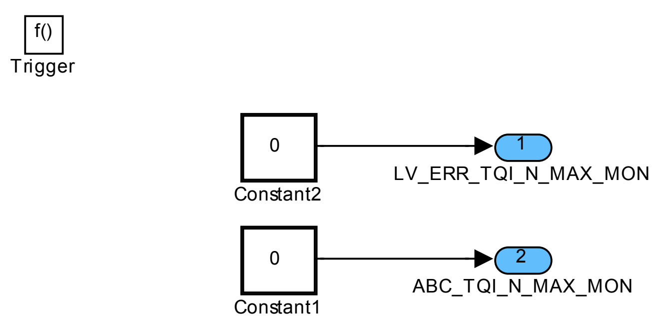


Figure 2 NLIM/ INITIALISATION

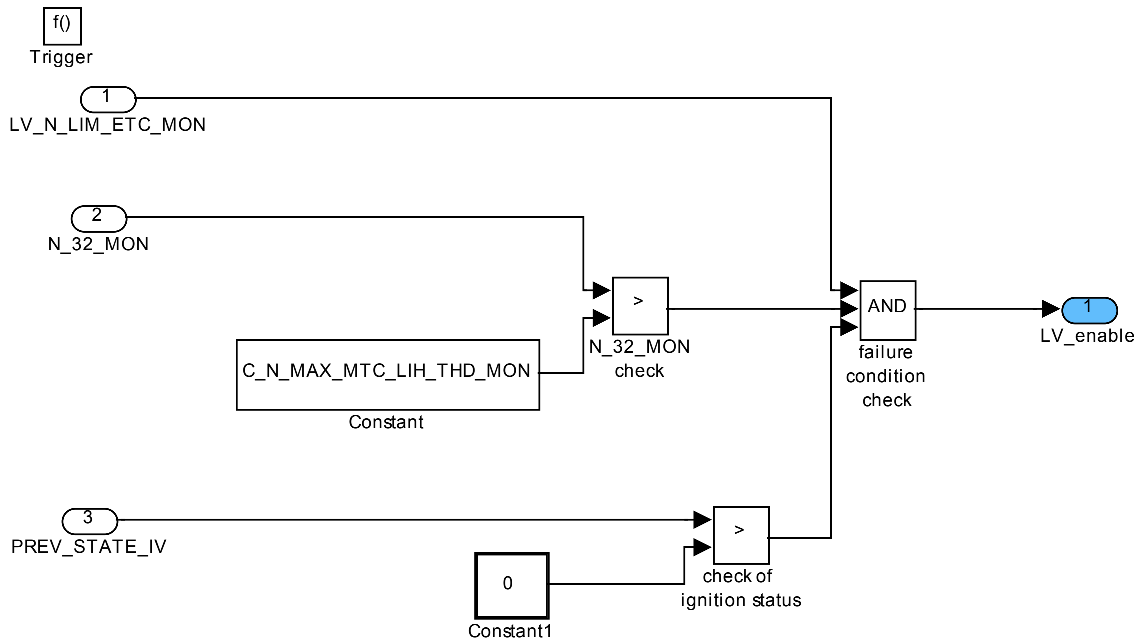


Figure 3 NLIM/ N_LIM_MON

E.28 Processor monitoring by copy of process monitoring (level 2')

General information:

A copy of the main part of the process monitoring (level 2) is built (called level 2') and used as a set of test calculations. The target is to verify the correct behaviour of the process monitoring at any time using predefined test data and comparing the resulting answer to the expected answer stored in a predefined table.

This processor monitoring function includes the monitoring functions described in the table below. However there is no real fault reaction like disable of throttle etc. released.

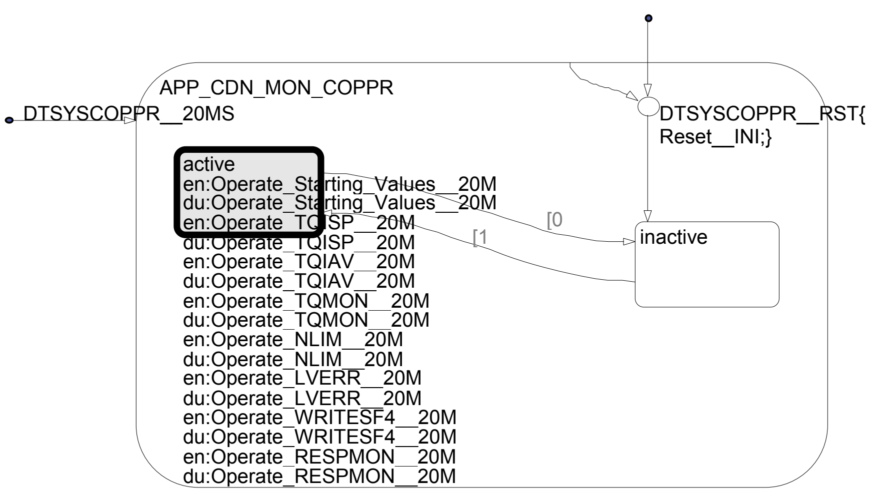
Since the answers of this test caclucations will be undecoded and compared to the predefined table ID_RESP_MON2 (see processor monitoring), it is important to generate a different answer for every index of starting values (NC_TEST_REC_IDX_MAX_MON2 different answers overall). The list of data points for the starting values must contain the indices 0...NC_TEST_REC_IDX_MAX_MON2-1 in ascending order. For detaills see also "Processor Monitoring" and "Processor Monitoring (Appl.Inc.)".

Table of software-code of level 2 modules that is included in level 2':

Module:	Specification:
Desired indicated engine torque	30E00K01.00E
Actual indicated engine torque	30E00L02.00D
Monitoring of actual indicated engine torque	30E00M02.00D
Monitoring of engine speed limitation	30E00I01.00D
Fault reaction of process monitoring	30E00G01.00E

The software-code of the level 2' sub-modules is an exact copy of software-code of the level 2 modules listed in the table above, with exceptions described in the following sub chapters.

Application conditions:



Function Description Overview:

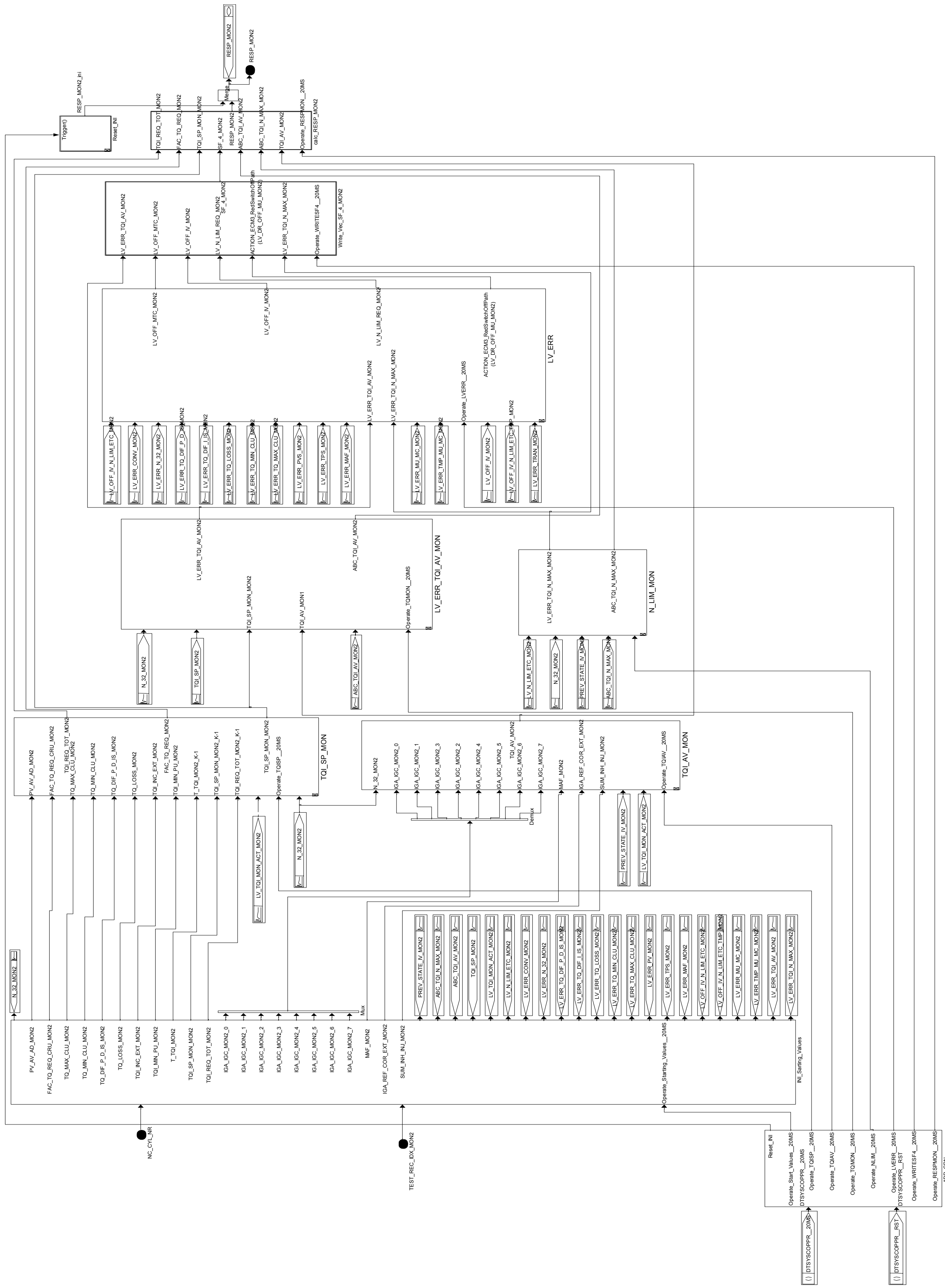


Figure 1 ECM2_DTSYSCOPPR

E.28.1 Initialisation of variables from the lists of starting values

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_INC_EXT_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
External torque request					
PV_AV_AD_MON2	V	0 ... FF H	0 ... 99.6	0.39	%
Pedal value adaptation					
FAC_TQ_REQ_CRU_MON2	V	0 ... FF H	0 ... 1.992	7.8125e-3	-
Torque demand from cruise control					
TQ_MIN_CLU_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
Monitored minimum torque at clutch					
TQ_MAX_CLU_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
Monitored maximum torque at clutch					
TQ_DIF_P_D_IS_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
P- and D- controller output for idle speed control					
TQ_LOSS_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
Torque losses					
N_32_MON2	V	0 ... FF H	0 ... 8160	32	rpm
Engine speed					
TQI_MIN_PU_MON2	V	0 ... 7FFF H	0 ... 1023,97	0,03125	Nm
Torque for trailing throttle					
IGA_REF_COR_EXT_MON2	V	0 ... FF H	-35.625 ... 60	0.375	°CRK
Correction of reference ignition angle (for external functions e.g. VVT, EGR...)					
IGA_IGC_MON2_x	V	0 ... FF H	-35.625 ... 60	0.375	°CRK
Actual ignition angle					
IGA_REF_MON2	V	0 ... FF H	-35.625 ... 60	0.375	°CRK
Reference ignition angle					
SUM_INH_INJ_MON2	V	0 .... 8 H	0 ... 8	1	-
SUM of inhibit injection (all cylinders)					
MAF_MON2	V	0..FFH	0..1.389E3	5.447	mg/stk
Mass air flow signal used monitoring level					
PREV_STATE_IV_MON2	V	0 ... FF H	0 ... 255	1	-
State of injection valves at previous cycle					
TQI_SP_MON2	V	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
Setpoint indicated engine torque (function level)					
SF_1_MON2	-	0 ... FF H	0 ... 255	1	-
Byte of logical starting values 1					
LV_TQI_MON_ACT_MON2	V	0 ... 1 H	0 ... 1	1	-
Logical variable for torque monitoring active					
LV_N_LIM_ETC_MON2	V	0 ... 1 H	0 ... 1	1	-
Engine speed limitation request					
LV_ERR_CONV_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in A/D-conversion present					
LV_ERR_N_32_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in engine speed present					
LV_ERR_TQ_DIF_P_D_IS_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault currently present in torque generation, symptom ‘idle speed controller-PD-part'					
LV_ERR_TQ_DIF_I_IS_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault currently present in torque generation, symptom ‘idle speed controller – I-part’					
LV_ERR_TQ_LOSS_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in torque losses present					
LV_ERR_TQ_MIN_CLU_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault currently present in torque generation, symptom ‘minimum torque at clutch’					
SF_2_MON2	-	0 ... FF H	0 ... 255	1	-
Byte of logical starting values 2					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQ_MAX_CLU_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault currently present in torque generation, symptom 'maximum torque at clutch'					
LV_ERR_PVS_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in pedal value signal present					
LV_ERR_TPS_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in TPS present					
LV_ERR_MAF_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault in mass air flow signal					
LV_OFF_IV_N_LIM_ETC_MON2	V	0 ... 1 H	0 ... 1	1	1
Reversible fuel Cut Off Switch for all cylinders					
SF_3_MON2	-	0 ... FF H	0 ... 255	1	-
Byte of logical starting values 3					
LV_ERR_MU_MC_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault detected by processor monitoring (continuous)					
LV_ERR_TMP_MU_MC_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault detected by processor monitoring (temporary)					
LV_ERR_TRAN_MON2	V	0 ... 1 H	0 ... 1	1	-
Only temporary error flag active in previous function execution					
SF_4_MON2	V	0 ... FF H	0 ... 255	1	-
Byte of logical starting values 4					
LV_N_LIM_REQ_MON2	V	0 ... 1 H	0 ... 1	1	1
Request for engine speed limitation					
LV_DR_OFF_MU_MON2	V	0 ... 1 H	0 ... 1	1	1
Request for wrong answers of the level 2' test calcualtions to the monitoring unit in order to disable MTC and IV					
LV_OFF_IV_MON2	V	0 ... 1 H	0 ... 1	1	1
Request for disable of IV power stage by main controller					
LV_OFF_MTC_MON2	V	0 ... 1 H	0 ... 1	1	1
Request for disable of MTC power stage by main controller					
LV_ERR_TQI_AV_MON2	V	0 ... 1 H	0 ... 1	1	1
Fault currently present in torque generation					
LV_ERR_TQI_N_MAX_MON2	V	0 ... 1 H	0 ... 1	1	1
Fault currently present in engine speed limitation					

Input data:

TEST_REC_IDX_MON2	NC_CYL_NR	NC_TEST_REC_IDX_MAX_MON2

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Res.	Unit
ID_TQI_INC_EXT_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
External torque request					
ID_TQI_MIN_PU_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... 7FFF H	0..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Torque for trailing throttle					
ID_PV_AV_AD_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 99.6	0.39	%

Name	Dim	Hex. limits	Phys. limits	Res.	Unit
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Pedal value adaptation					
ID_FAC_TQ_REQ_CRU_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 1.992	7.8125e-3	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Torque demand from cruise control					
ID_TQ_MIN_CLU_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Monitored minimum torque at clutch					
ID_TQ_MAX_CLU_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Monitored maximum torque at clutch					
ID_TQ_DIF_P_D_IS_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
P- and D- controller output for idle speed control					
ID_TQ_LOSS_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Torque losses					
ID_N_32_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 8160	32	rpm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Engine speed (monitoring level)					
ID_IGA_REF_COR_EXT_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	-35.625 ... 60	0.375	°CRK
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Reference ignition angle					
ID_IGA_IGC_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	-35.625 ... 60	0.375	°CRK
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Actual ignition angle					
ID_SUM_INH_INJ_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... 8 H	0 ... 8	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
SUM of inhibit injection (all cylinders)					
ID_MAF_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 1389	5.447	mg/stk

Name	Dim	Hex. limits	Phys. limits	Res.	Unit
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Actual fuel mass flow (torque monitoring)					
ID_PREV_STATE_IV_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
State of injection valves at previous cycle					
ID_TQI_SP_MON2	**NC_TEST_REC_IDX_MAX_MON2**	8000 ... 7FFF H	-1024..1023.97	0.03125	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Setpoint indicated engine torque					
ID_SF_1_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Table of logical starting values 1					
ID_SF_2_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Table of logical starting values 2					
ID_SF_3_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Table of logical starting values 3					
ID_SF_4_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Table of logical starting values 4					
ID_ABC_TQI_AV _MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Anti-bounce counter for fault in torque generation in AFS mode					
ID_ABC_TQI_N_MAX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 255	1	-
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Anti-bounce counter for monitoring of engine speed limitation					
ID_TQI_SP_MON_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 510	2	Nm
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Resulting desired indicated engine torque, old value					
ID_TQI_REQ_TOT_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 510	2	Nm

Name	Dim	Hex. limits	Phys. limits	Res.	Unit
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Desired non-delayed indicated engine torque, old value					
ID_T_TQI_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... FF H	0 ... 10.2	0.04	s
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Dead time for delay of desired indicated engine torque, old value					

General information:

Based on the number of the question asked by the monitoring unit a corresponding set of starting values for the test caclucation is selected from the lists of starting values.

The values of LDPM_TEST_REC_IDX_MON2 are fixed. They depend on the maximum NC_TEST_REC_IDX_MAX_MON2 of the generated index TEST_REC_IDX_MON2. This leads to the phys. limits of 0 ... NC_TEST_REC_IDX_MAX_MON2-1 which corresponds to the hex. limits 0 ... NC_TEST_REC_IDX_MAX_MON2-1. The single values of LDPM_TEST_REC_IDX_MON2 are given by:

Cell	0	1	2	....	NC_TEST_REC_IDX_MAX_MON2-1
value	0	1	2	....	NC_TEST_REC_IDX_MAX_MON2-1

FUNCTION DESCRIPTION:

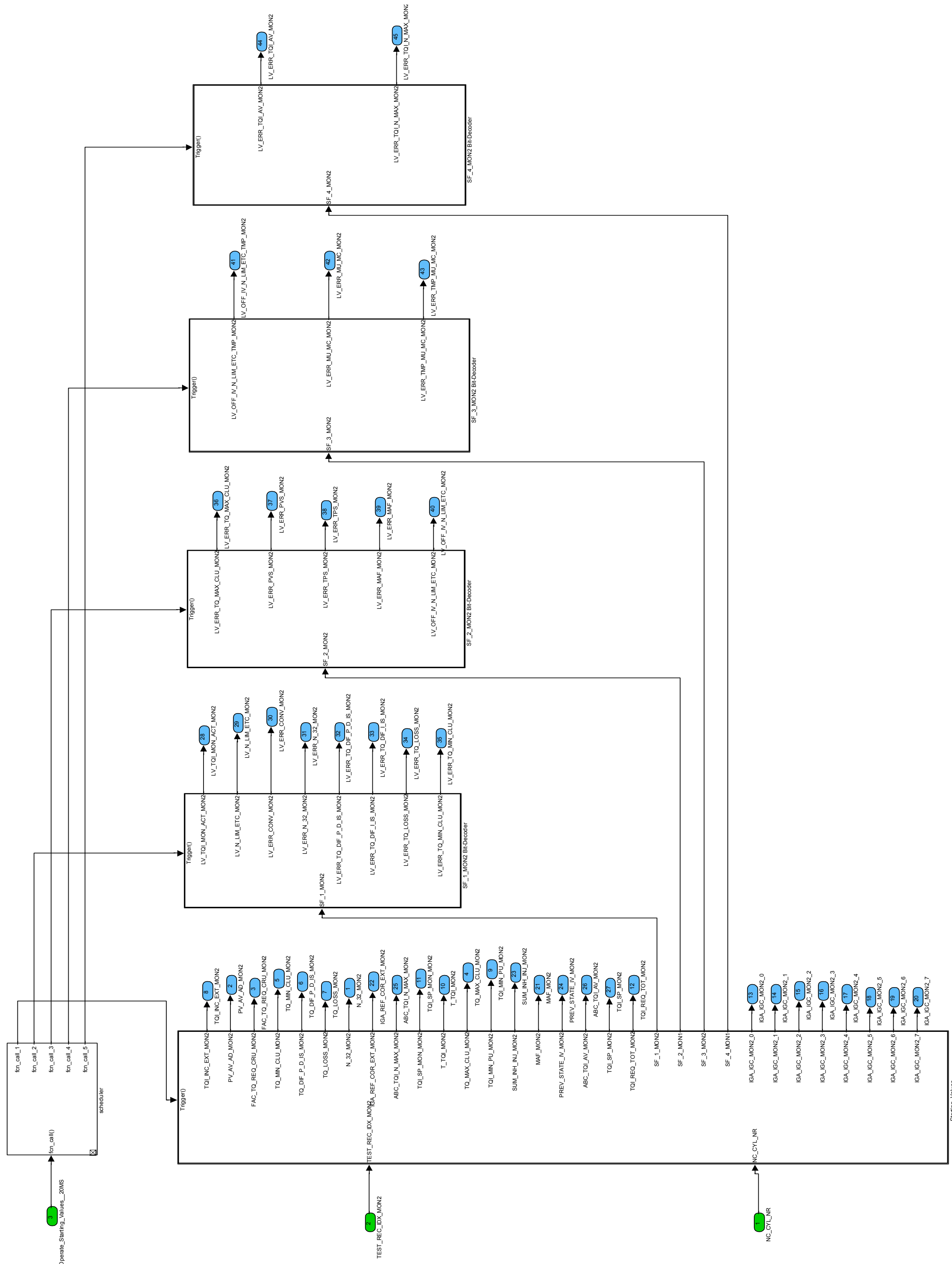


Figure 2 ECM2_DTSYSCOPPR/INI_Starting_Values

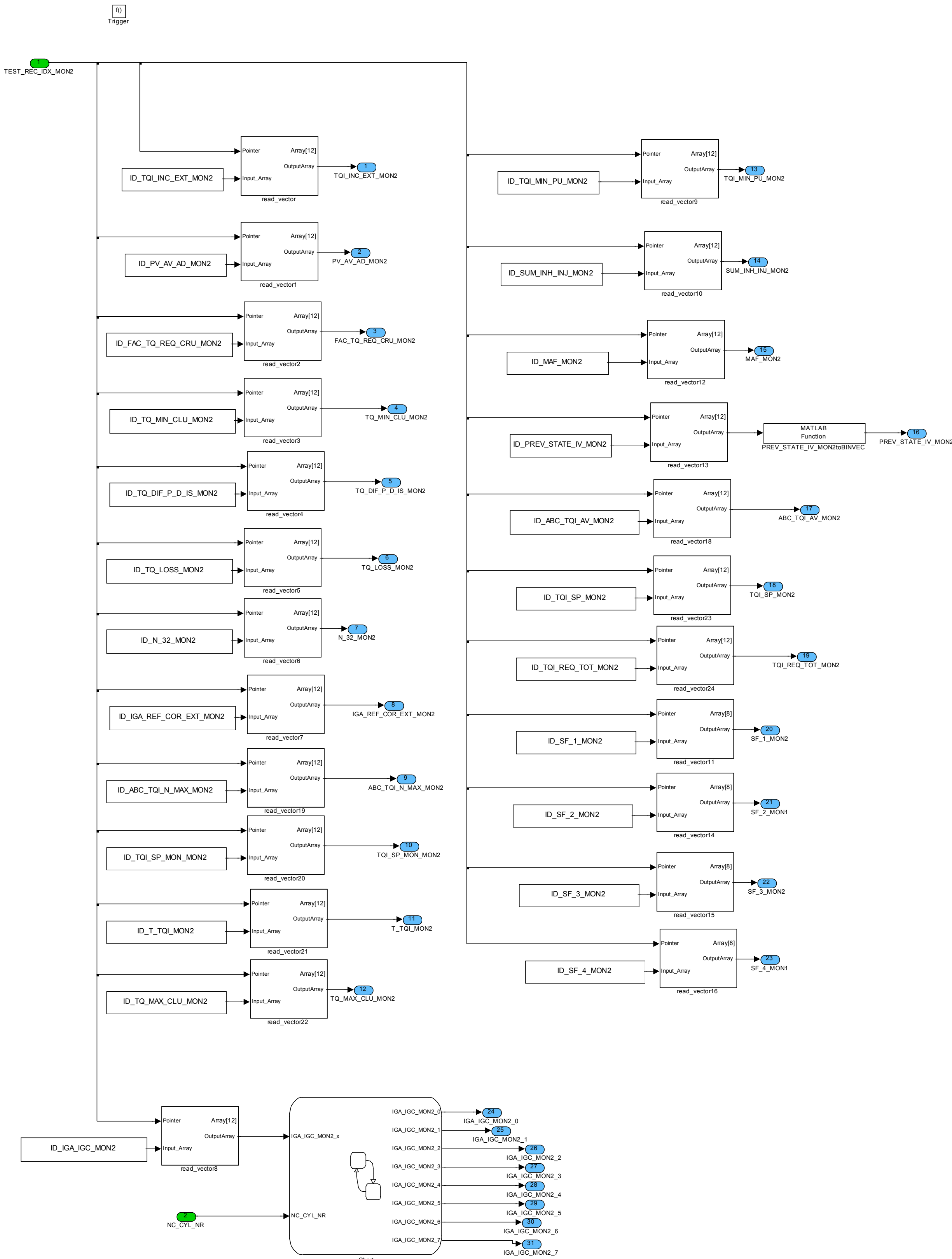


Figure 3 ECM"_DTSYSCOPPR/INI_Starting_Values/Starting_Values

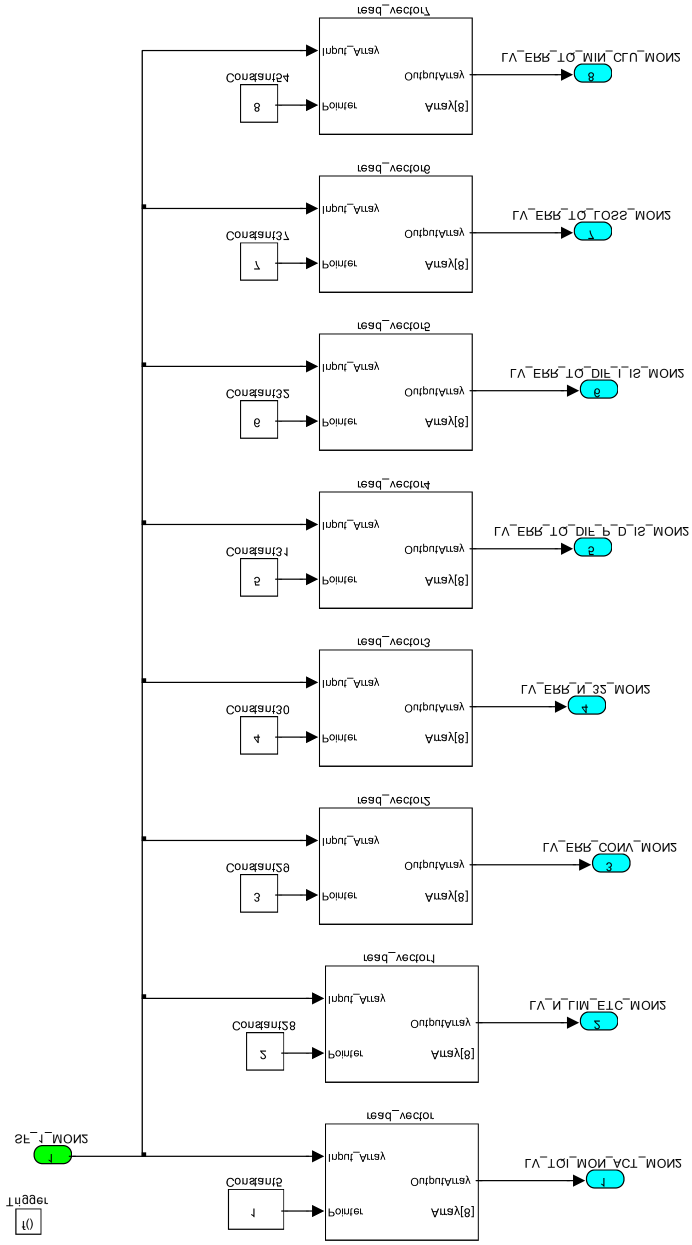


Figure 4 ECM2_DTSYSCOPPR/INI_Starting_Values/Bit_Selection_SF1_MON2

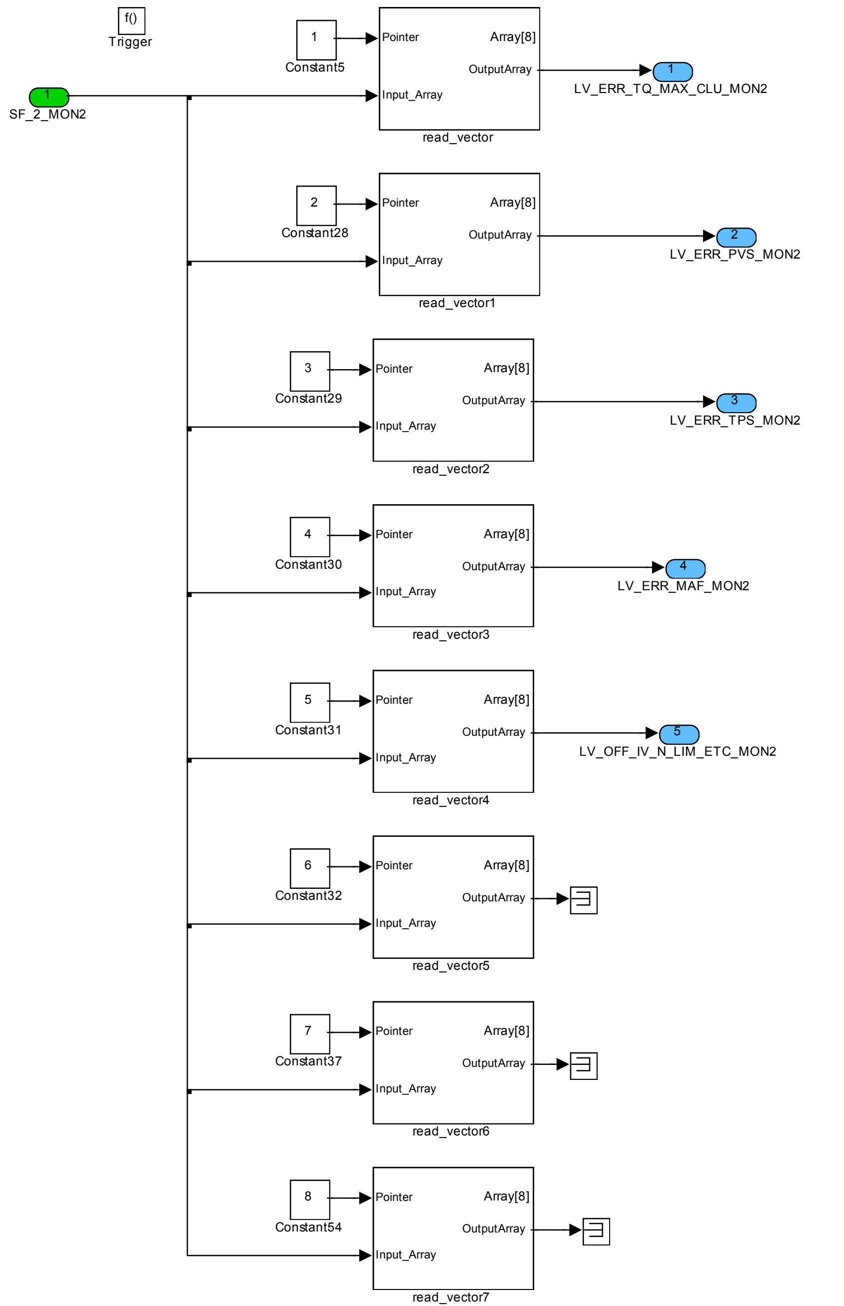


Figure 5 ECM2_DTSYSCOPPR/INI_Starting_Values/Bit_Selection_SF2_MON2

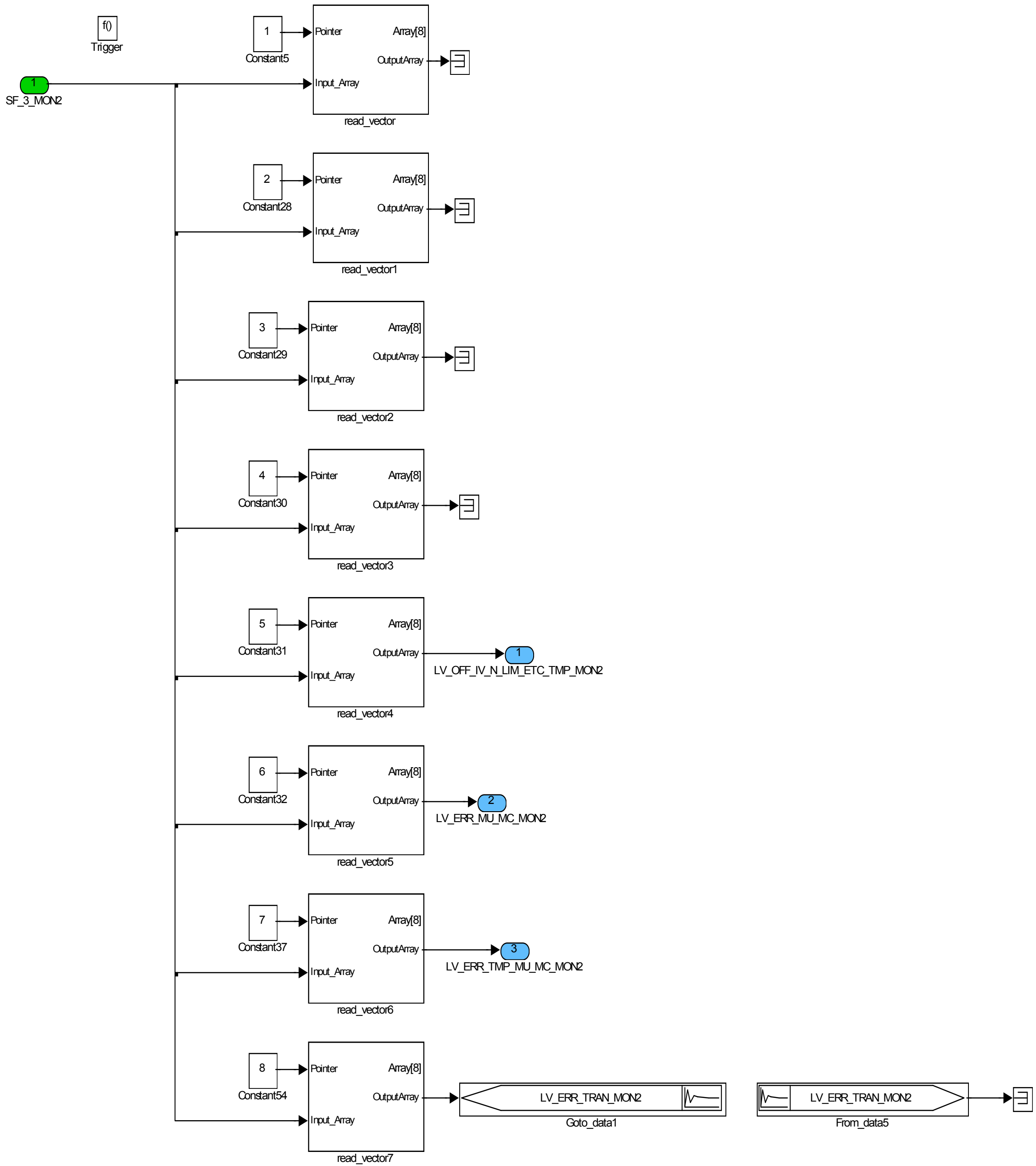


Figure 6 ECM2_DTSYSCOPPR/INI_Starting_Values/Bit_Selection_SF3_MON2

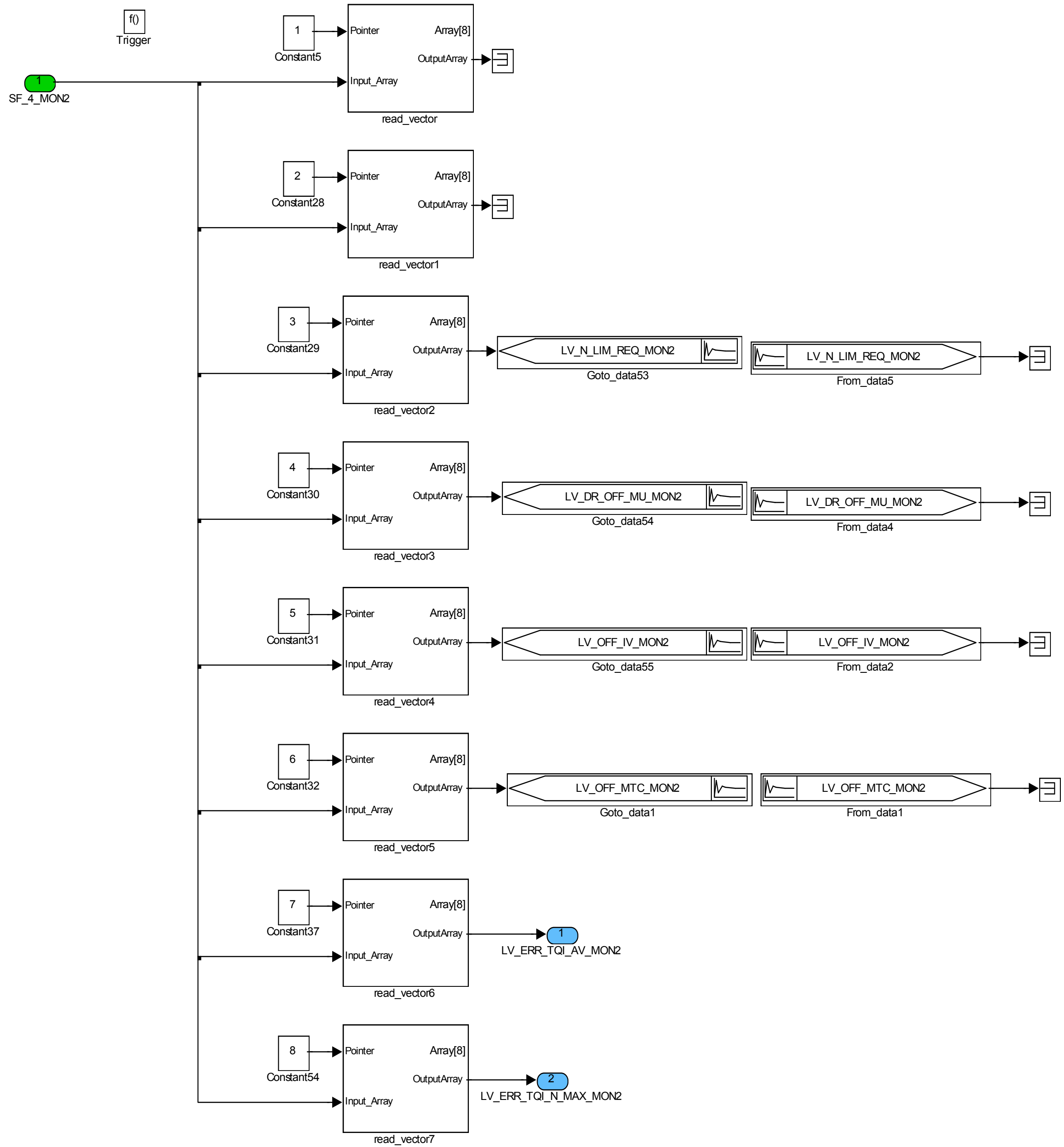


Figure 7 ECM2_DTSYSCOPPR/INI_Starting_Values/Bit_Selection_SF4_MON2

Initialization of the following values are needed to ensure an unambiguous final result for RESP_MON2 in case of (LV_TQI_MON_ACT_MON2 = 0):

TQI_AV_MON2 = 0
FAC_TQ_REQ_MON2 = 0

E.28.2 Desired indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TQI_SP_MON_MON2	V	0 ... FF H	0 ... 510	2	Nm
Desired indicated engine torque					
TQI_REQ_TOT_MON2	V	0 ... FF H	0 ... 510	2	Nm
Desired non-delayed indicated engine torque					
T_TQI_MON2	V	0 ... FF H	0 ... 10.2	0.04	S
Dead time for delay of desired indicated engine torque					
FAC_TQ_REQ_DRIV_MON2	V	0 ... FF H	0 ... 1.992	7.8125e-3	-
Scaling factor for requested torque at clutch from driver					
FAC_TQ_REQ_MON2	V	0 ... FF H	0 ... 1.992	7.8125e-3	-
Scaling factor for requested torque at clutch from driver and cruide control					

Input data:

N_32_MON2	PV_AV_AD_MON2	FAC_TQ_REQ_CRU_MON2	TQ_MAX_CLU_MON2
TQ_MIN_CLU_MON2	TQ_DIF_P_D_IS_MON2	TQ_LOSS_MON2	TQI_INC_EXT_MON2
LV_TQI_MON_ACT_MON2	TQI_MIN_PU_MON2		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_REQ_DRIV_MON2	8x8	0 ... FF H	0 ... 1.992	7.8125e-3	-
LDP_N_32_MON2_IP_FAC_TQ_MON2	8	0 ... FF H	0 ... 8160	32	Rpm
LDP_PV_AV_AD_MON2_IP_FAC_MON2	8	0 ... FF H	0 ... 99.6	0.39	%
Scaling factor for requested torque at clutch from driver					
C_FIL_TQI_SP_INP_0_MON2	1	8000 ... 7FFF H	-1.0 ... 0.999939	3.05e-5	-
Filter constant for realization of first order filer					
C_FIL_TQI_SP_INP_1_MON2	1	8000 ... 7FFF H	-1.0 ... 0.999939	3.05e-5	-
Filter constant for realization of first order filer					
C_FIL_TQI_SP_OUT_1_MON2	1	8000 ... 7FFF H	-1.0 ... 0.999939	3.05e-5	-
Filter constant for realization of first order filer					
C_T_TQI_MON2	1	0 ... FF H	0 ... 10.2	0.04	S
Dead time for torque decrease					

General information:

The module has exactly the same functionality than the corresponding module 'Desired indicated engine torque' of level 2 (`30E00K01.00E`) except:

- No RAM test of output data performed
- No initialisation section included
- TQI_SP_MON (output) renamed to TQI_SP_MON_MON2
- All input data, output data and calibration data renamed to level 2' naming
  Example: xyz_MON renamed to xyz_MON2

E.28.3 Actual indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
TQI_AV_MON2	V	0 ... FF H	0 ... 510	2	Nm
actual indicated engine torque (process monitoring)					
EFF_SCC_MON2	V	0 ... FF H	0 ... 1.992156	7.843e-3	-
actual efficiency of cylinder cut-off for torque monitoring					
TQI_REF_MON2	V	0 ... FF H	0 ... 510	2	Nm
reference indicated engine torque (process monitoring)					
EFF_IGA_MON2	V	0 ... FF H	0 ... 1.992156	7.843e-3	-
actual ignition efficiency for torque monitoring					
IGA_DIF_AV_MON2	V	0 ... FF H	0 ... –95.625	0.375	°CRK
difference of IGA_AV_MV_MON and IGA_REF_COR_MON					

Input data:

N_32_MON2	MAF_MON2	IGA_IGC_MON2_x	IGA_REF_ COR_EXT_MON2
PREV_STATE_IV_MON2	SUM_INH_INJ_MON2	LV_TQI_MON_ACT_MON2	

Calibration data:

Name	Dim	Hex. Limits	Phys. Limits	Resol.	Unit
IP_TQI_REF_MON2	6x8	0 ... FF H	0 ... 510	2	Nm
LDP_N_32_MON2_IP_TQI_REF_MON2	6	0 ... FF H	0 ... 8160	32	rpm
LDP_MAF_MON2_IP_TQI_REF_MON2	8	0 ... FF H	0 ... 1389	5.43	mg/stk
Indicated engine torque at reference conditions (process monitoring)					
IP_IGA_REF_MON2	8x8	0 ... FF H	-35.625 ...60	0.375	°CRK
LDP_N_32_MON2_IP_IGA_REF_MON2	8	0 ... FF H	0 ... 8160	32	rpm
LDP_MAF_MON2_IP_IGA_REF_MON2	8	0 ... FF H	0 ... 1389	5.43	mg/stk
Reference ignition angle (process monitoring)					
IP_EFF_IGA_MON2	8	0 ... FF H	0 ... 1.992156	7.843e-3	-
LDP_IGA_DIF_AV_MON2_IP_EFF_MON2	8	0 ... FF H	0 ... –95.625	0.375	°CRK
Basic ignition efficiency (process monitoring)					

General information:

The module has exactly the same functionality than the corresponding module 'Actual indicated engine torque' of level 2 (`30E00L02.00D`) except:

- No RAM test of output data performed
- No initialisation section included
- All input data, output data and calibration data renamed to level 2' naming
  Example: xyz_MON renamed to xyz_MON2

E.28.4 Monitoring of actual indicated engine torque

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_TQI_AV_MON2	V	0 ... 1 H	0 ... 1	1	-
fault currently present in torque generation					
TQI_DIF_AV_MON2	V	0 ... FF H	0 ... 510	2	Nm
deviation between actual and desired indicated engine torque					
TQI_DIF_SP_MON2	V	0 ... FF H	0 ... 510	2	Nm
deviation between torque setpoint and desired indicated engine torque					
TQI_DIF_MAX_MON2	V	0 ... FF H	0 ... 510	2	Nm
Permitted deviation between actual and desired indicated engine torque					
ABC_TQI_AV_ MON2	V	0 ... FF H	0 ... 255	1	-
Anti bounce counter for fault in torque generation					

Input data:

TQI_SP_MON_MON2	TQI_AV_MON2	TQI_SP_MON2	N_32_MON2
	LV_TQI_MON_ACT_MON2		

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_TQI_DIF_MAX_MON2	4x4	0 ... FF H	0 ... 510	2	Nm
LDP_N_32_MON2_IP_TQI_DIF_MON2	4	0 ... FF H	0 ... 8160	32	Rpm
LDP_TQI_SP_MON_MON2_IP_MON2	4	0 ... FF H	0 ... 510	2	Nm
Threshold for diagnosis of engine torque overflow					
C_GAIN_TQI_DIF_MAX_MON2	1	0 ... FF H	0 ... 63.75	0.25	-
Factor for threshold diagnosis with TQI_SP					
C_ABC_INC_TQI_AV_MON2	1	0 ... FF H	0 ... 255	1	-
Anti bounce counter increment for torque monitoring					
C_ABC_MAX_TQI_AV_MON2	1	1 ... FF H	1 ... 255	1	-
Maximum value of anti-bounce counter for torque monitoring					

General information:

The module has exactly the same functionality than the corresponding module 'Monitoring of actual indicated engine torque' of level 2 (`30E00M02.00D`) except:

- No RAM test of output data performed
- No initialisation section included
- TQI_SP_MON (input) renamed to TQI_SP_MON_MON2
- All input data, output data and calibration data renamed to level 2' naming
  Example: xyz_MON renamed to xyz_MON2

E.28.5 Monitoring of engine speed limitation

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_TQI_N_MAX_MON2	V	0 ... 1 H	0 ... 1	1	-
Fault currently present in engine speed limitation					
ABC_TQI_N_MAX_MON2	V	0 ... FF H	0 ... 255	1	-
Anti-bounce-counter for fault in engine speed limitation					

Input data:

N_32_MON2	LV_N_LIM_ETC_MON2	PREV_STATE_IV_MON2	

Calibration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
C_N_MAX_MTC_LIH_THD_MON2	1	0 ... FF H	0 ... 8160	32	Rpm
Speed threshold for the detection of a fault in the engine speed limitation					
C_ABC_INC_TQI_N_MAX_MON2	1	0 ... FF H	0 ... 255	1	-
Anti bounce counter increment					
C_ABC_MAX_TQI_N_MAX_MON2	1	1 ... FF H	1 ... 255	1	-
Maximum value of anti-bounce counter					

General information:

The module has exactly the same functionality than the corresponding module 'Monitoring of engine speed limitation' of level 2 (`30E00I01.00D`) except:

- No RAM test of output data performed
- No initialisation section included
- All input data, output data and calibration data renamed to level 2' naming
  Example: xyz_MON renamed to xyz_MON2

E.28.6 Fault reaction of process monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_N_LIM_REQ_MON2	V	0 ... 1 H	0 ... 1	1	-
Request for engine speed limitation					
LV_OFF_MTC_MON2	V	0 ... 1 H	0 ... 1	1	-
Request for disable of MTC power stage by main controller					
LV_OFF_IV_MON2	V	0 ... 1 H	0 ... 1	1	-
Request for disable of IV power stage by main controller					
LV_ERR_TRAN_MON2	V	0 ... 1 H	0 ... 1	1	-
Flag for temporary fault during last recurrency					
LV_OFF_IV_N_LIM_ETC_TMP_MON2	V	0 ... 1 H	0 ... 1	1	-
Flag for temporary injection cut off during last recurrency					

Input data:

LV_ERR_CONV_MON2	LV_ERR_N_32_MON2	LV_ERR_TQ_DIF_P_D_IS_MON2	LV_ERR_TQ_DIF_I_IS_MON2
LV_ERR_TQ_LOSS_MON2	LV_ERR_TQ_MIN_CLU_MON2	LV_ERR_TQ_MAX_CLU_MON2	LV_ERR_PVS_MON2
LV_ERR_TPS_MON2	LV_ERR_MAF_MON2	LV_ERR_TQI_AV_MON2	LV_ERR_TQI_N_MAX_MON2
LV_ERR_MU_MC_MON2	LV_ERR_TMP_MU_MC_MON2	LV_OFF_IV_N_LIM_ETC_MON2	

General information:

The module has exactly the same functionality than the corresponding module ‘Fault reaction of process monitoring’ of level 2 (`30E00G01.00E`) except:

- No RAM test of output data performed
- No initialisation section included
- All input data, output data and calibration data renamed to level 2’ naming
  Example: xyz_MON renamed to xyz_MON2

The ACTION_ECM3_RedSwitchOffPath() is replaced, instead of the original action call LV_DR_OFF_MU_MON2 is set to 1.

The byte SF_4_MON2 is updated with the output flags after execution.

E.28.7 Calculation of final result for level 2' test calcucation

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
RESP_MON2	V/O	0 ... FFFF H	0 ... 65535	1	-
Answer of copy of process monitoring (Level 2’)					

Input data:

TQI_SP_MON_MON2	TQI_AV_MON2	FAC_TQ_REQ_MON2	ABC_TQI_AV_MON2
ABC_TQI_N_MAX_MON2	SF_4_MON2	TQI_REQ_TOT_MON2	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Res.	Unit
ID_RESP_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0000h ... FFFFh	0 ... 65535	1	1
LDPM_TEST_REC_IDX_MON2	**NC_TEST_REC_IDX_MAX_MON2**	0 ... NC_TEST_REC_IDX_MAX_MON2-1	0 ... NC_TEST_REC_IDX_MAX_MON2-1	1	1
Assignment table for answer decoder of copy of process monitoring					

General information:

There is a possibility for faults, which lead to wrong interim results but to a correct final result of the test calculation. In order to ensure that such a kind of fault is detectable, the final result is composed of several interim results of the whole test calculation.

FUNCTION DESCRIPTION:

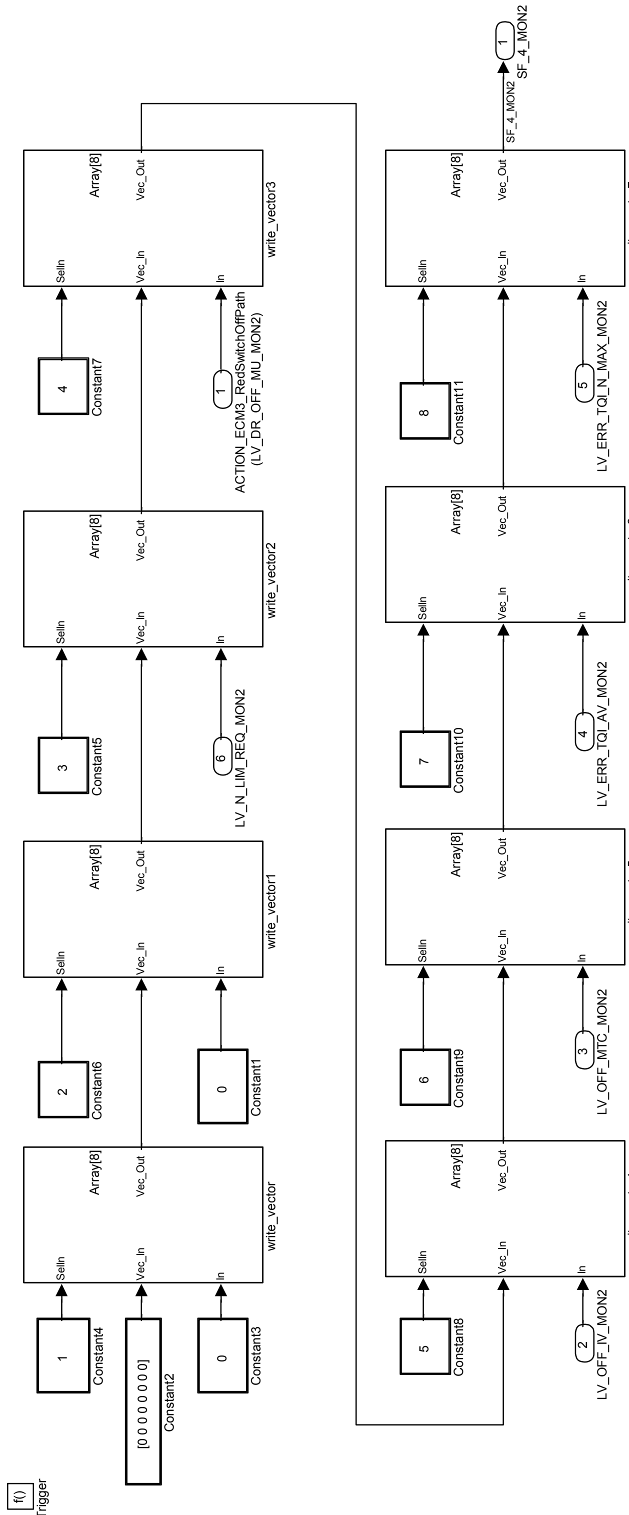


Figure 8 ECM2_DTSYSCOPPR/Rewrite_SF_4_MON2

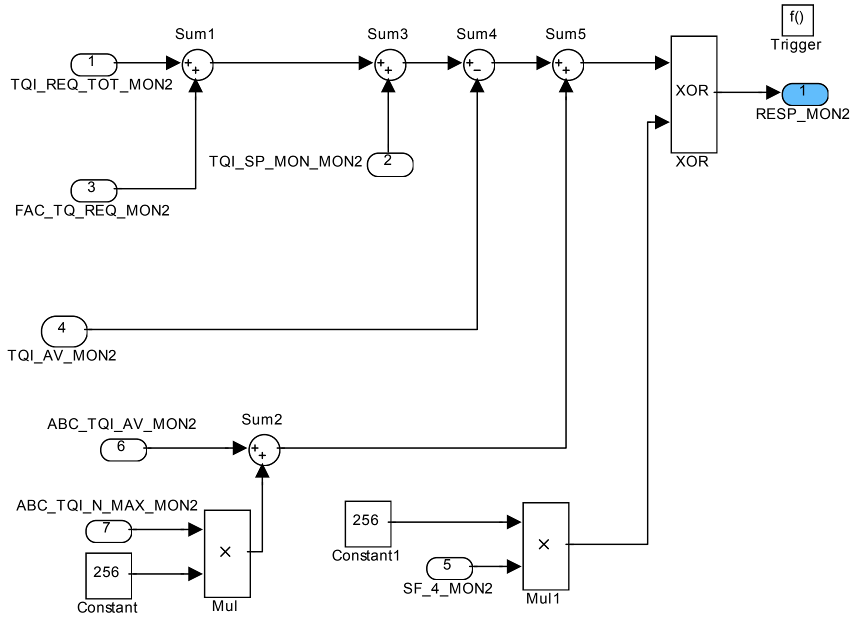


Figure 9 ECM2_DTSYSCOPPR/Calc_RESP_MON2

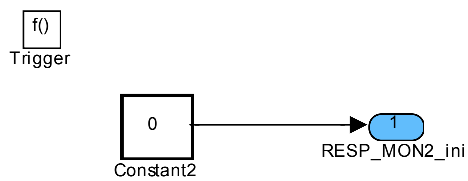


Figure 10 ECM2_DTSYSCOPPR/Reset_INI

All operations are performed with hexadecimal values only. The different units are not taken into account. The result of the operations and all intermediate results are 32 Bit word size, overflow and underflow is permitted.

Because all intermediate results have the size 32bit, only the low word (the least significant 16 bits) are taken as result RESP_MON2. In case of an underflow or overflow while computing the result there is still a significant change in the 16bit result RESP_MON2.

ID_RESP_MON2(TEST_REC_IDX_MON2) must fit to RESP_MON2 in case of a correct processing of test calulation with index TEST_REC_IDX_MON2.

E.29 Processor Monitoring

During operation of the electronic control unit (ECU), the Main Controller MC and the Monitoring Unit MU permanently exchange data to enable the MU the monitoring of the MC, especially to ensure the correct work of the Level 2 (process monitoring). This monitoring is the *Processor Monitoring* of the Monitoring Concept (Level 3). It is based on a set of test calculations (based on a Copy of Process Monitoring) as well as several other tests concerning communication and detection of memory faults. "Communication" used in the context of *Processor Monitoring* is a synonym for "Monitoring Communication". There are two independent interfaces to different power stages, MC as well as MU is able to disable the power stages separately.

According the figure “Sketch of the *Processor Monitoring* with two power stages” there exists two separate microcontrollers, MC and MU. Each microcontroller has its own memory and is running its own software. Therefore variables implemented on MU are not accessible by MC and vice versa, exept the values of the variables are transferred via the internal communication to the other mircocontroller. However, a consequence of this item is that one must carefully distinguish the scope of a variable according to the context.

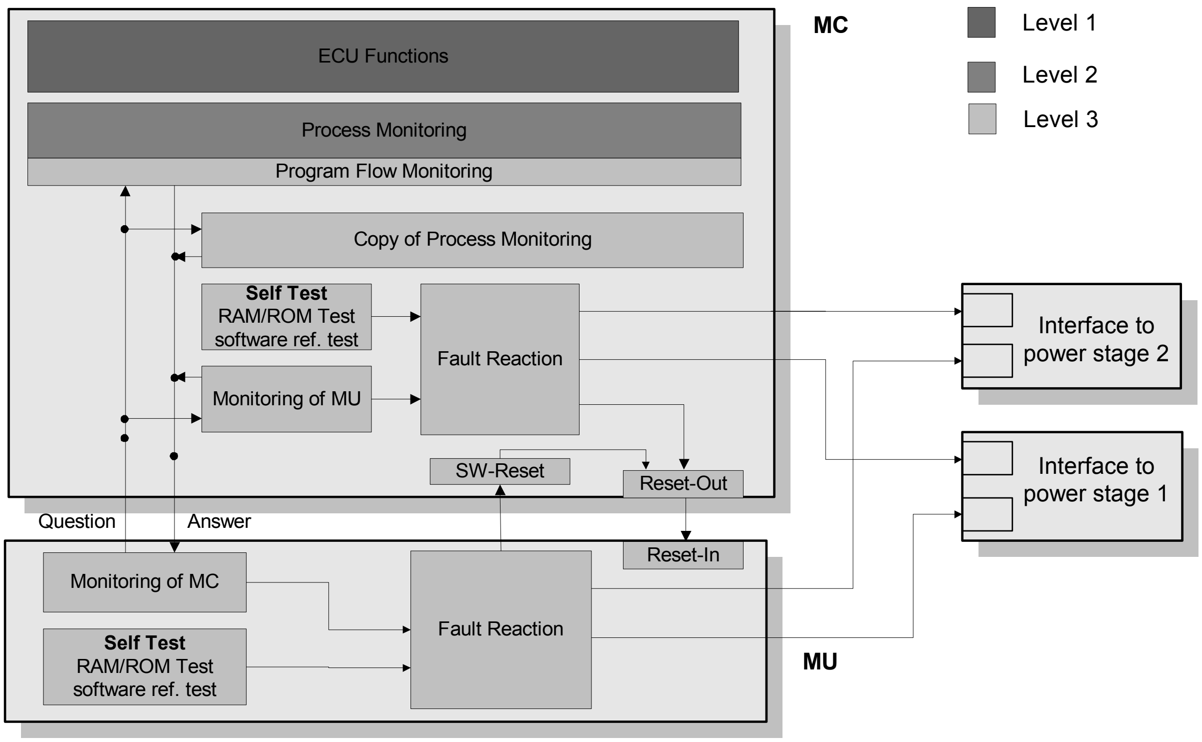


Figure 73: Sketch of the *Processor Monitoring* with two power stages

E.29.1 Delay of activation of Processor Monitoring

Input data:

LV_DIAG_END_RLY_MAIN_DLY	LV_RLY_MAIN_DLY_ERR		

FUNCTION DESCRIPTION:

Shortly after start-up, the Predrive Check (see section "Initialisation of the ECU") is performed in order to determine whether the MU can disable the safety-relevant power stages.

At the point when the Predrive Check is performed, it has to be ensured that the "environment" inside the ECU is in a suitable state: the check should only fail due to a real problem with the switch-off paths but not due to "extraneous" effects, like certain supply voltages not (yet) being available – such topics are not in the focus of the Predrive Check and consequently should not interfere with it.

Consequently, the Predrive Check has to be delayed until the environment is ready. While Processor Monitoring is not active, it has to be ensured that the power stages remain disabled (this has to be done in other SW).

As a matter of fact, in order to avoid harmful side effects, the **complete Processor Monitoring** function shall be delayed; in particular, no communications with the MU shall be triggered as long as the environment is not yet ready.

The readyness of the environment is provided via a dedicated diagnosis which is executed until a positive result is obtained or the power latch phase is entered; the delay shall be active as long as this diagnosis has not yet been completed (i.e., LV_DIAG_END_RLY_MAIN_DLY = 0) or has detected an error (i.e., LV_RLY_MAIN_DLY_ERR = 1). Consequently, Processor Monitoring shall only be activated when the diagnosis is completed (i.e., LV_DIAG_END_RLY_MAIN_DLY = 1) AND has not detected an error (i.e., LV_RLY_MAIN_DLY_ERR = 0).

Formula section:

IF ((LV_DIAG_END_RLY_MAIN_DLY = 1) **AND** (LV_RLY_MAIN_DLY_ERR = 0))

THEN

/ diagnosis has been completed with positive result */*

calculate Processor Monitoring function

ELSE

/ diagnosis not yet completed or negative result */*

skip the complete Processor Monitoring function package

ENDIF

E.29.2 Initialisation of the ECU

Output data:

Name	Mode	Hex. Limits	Phys. Limits	Resol.	Unit
LV_PRDR_ACT	O	0...1H	0...1	1	[-]
Active flag of PREDRIVE Check (0=inactive, 1=active)					

Input data:

RST_CTR_MC	RST_CTR_MU	LV_ERR_TMP_MU	LV_ERR_TMP_MC

FUNCTION DESCRIPTION:

After a reset or power up the ECU is initialized and the communication between MC and MU for the Monitoring Concept runs through an INIT-State, a PREDRIVE-State, into a NORMAL-State (Figure 74). The transitions of the states are controlled by the communication itself. In case of some faults a transition into DISABLE-State is performed. For details see chapter “Communication between MC and MU” and “Fault reaction of processor monitoring”. During the inizialisation also the digital output ports have to be configurated. For details see “Communication between MC and MU”.

In the different states different actions are carried out. In this chapter the main focus is the activation of the power stages.

Application conditions:

Initialisation: LV_PRDR_ACT = 1;

Recurrence: before calling the PREDRIVE check

Activation: always after each reset or power up

E.29.2.1 Reset or Power up

Both, MC and MU, run a self test, including a RAM and a ROM test. The MU disables the safety relevant power stages; the bit LV_PRDR_ACT is set before calling the PREDRIVE check. The MC enables all safety relevant power stages. The Predive check is done to make sure that the MU is able to disable the saftey relevant power stages, even if MC enables the power stages.

The Predrive Check itself is performed in the following sequence for all safety relevant power stages:

1. the partial result of the dedicated power stage is set to not ok.
2. kicks off the diagnosis of the dedicated power stage
3. the partial result is ok only, if
   - the diagnois is non-ambiguous
   - the diagnosis could be performed within a specific time periode
   - the power stage is disabled

After having collected all partial result the final result is set to ok only, if all partial results are ok. The final result of the Predrive Check is not stored in an intermediate variable to avoid unallowed changes in the time periode between storage and evaluation. If a storage of the result is needed, it has to be done with value and complement (like in "RAM test on MC for process monitoring") to provide a consitency check when evaluating the result.

In case of an error detected (i.e. RAM error) during active Predrive Check, the Predrive Check must be cancelled. In this case LV_PRDR_ACT has to be reset in order to be able to disable the safety relevant power stages. The result of the Predrive Check is not evaluated in this case.

In error free case after having received the result of the Predrive Check the INIT communication is performed.

E.29.2.2 INIT state communication

After the INIT communication the reset counters RST_CTR_MU and RST_CTR_MC are checked. If RST_CTR_MC or RST_CTR_MU (number of resets requested on MC or MU) is bigger than one, both controllers (MC and MU) disable the power stages. Therefore the reset counters have to be transferred in the INIT-communication.

After the INIT communication the error code and reset counter of the main controller as well as the error code and reset counter of the monitoring unit are avaliable. This values have to be copied (see "Fault reaction of processor monitoring") to maintain an optional error memory management.

A software reference check and a pin configuration check is done on both, MC and MU.

The result of the PREDRIVE-check is tested now. If the result of the Predive check indicates a fault then a fault reaction is carried out (see chapter “Fault reaction of processor monitoring”).

E.29.2.3 PREDRIVE state communication

The second communication is the PREDRIVE communication. MU sends LV_ERR_TMP_MU to MC (see “Fault reaction of processor monitoring”). If LV_ERR_TMP_MC is set, the MC has to disable the power stages, while a set LV_ERR_TMP_MU indicates the MC that the MU disables. So, if LV_ERR_TMP_MC or LV_ERR_TMP_MU is set the power stages are disabled temporary.

After the PREDRIVE communication the bit LV_PRDR_ACT is reset.

E.29.2.4 First NORMAL state communication

The third communication is a NORMAL state communication. If LV_ERR_TMP_MU and LV_ERR_TMP_MC are reset, the power stages are enabled after this communication. Otherwise the MC keeps the power stages disabled (LV_ERR_TMP_MC is set) or MU keeps the the power stages disabled (LV_ERR_TMP_MU is set). All other following monitoring communications will be NORMAL state or DISABLE state.

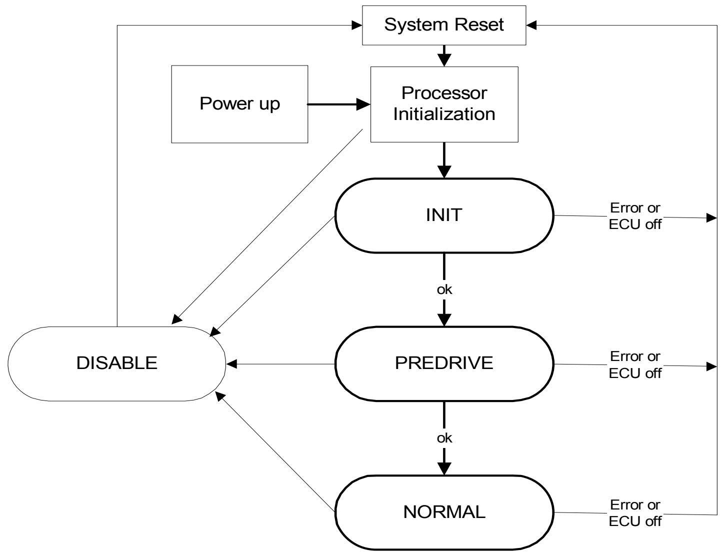


Figure 74: State of the monitoring communication of the ECU

E.29.3 ROM tests

E.29.3.1 Standard ROM Test on MC

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ROM_CRC_READY	V/O	0...1H	0...1	1	[-]
Indicates that the ROM test has been finished since last power up if set					

Note: LV_ROM_CRC_READY and its complement are located in the reset safe RAM area, as indicated by the mode “S”.

Input data:

ERR_COD_MC			

FUNCTION DESCRIPTION:

The standard ROM test checks the whole ROM (code, constant and calibration data) only after reset or power up. The CRC-32-Checksum is calculated in the background of the operating system. The Processor Monitoring checks cyclical whether the ROM test is finished or not. If the ROM test is finished and the Processor Monitoring detects an error, a fault reaction is performed directly (refer to chapter: “Fault reaction of processor monitoring”).
The standard ROM test has to be finished 3 minutes after power up or reset.

The variable LV_ROM_CRC_READY is mandatory to do a correct decrementation of the reset counter, see "Fault reaction of processor monitoring". If a reset is done because of a standard ROM test error or a cyclical ROM test error and after power up LV_ROM_CRC_READY has to be initialized with zero. Otherwise (in case of a reset not caused by any ROM error) LV_ROM_CRC_READY stays unchanged.

Application conditions:

Initialisation: LV_ROM_CRC_READY = 0; /* after power up */

LV_ROM_CRC_READY = 0; /* after reset caused by any ROM error */

No Initialization: after reset caused by other errors (no ROM error)

E.29.3.2 Cyclical ROM Test of Process Monitoring on MC

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ROM_CKS_ECU_CLC	-	0... FFFFFFFFFFFFFFFFH	0...1.84467E+19	1	[-]
Actual value of ECU code ROM test of process monitoring					
ROM_CKS_CAL_CLC_L	V	0...FFFFFFFFH	0...4294967295	1	[-]
Actual value of CAL data ROM test of process monitoring (low dword)					
ROM_CKS_CAL_CLC_H	V	0...FFFFFFFFH	0...4294967295	1	[-]
Actual value of CAL data ROM test of process monitoring (high dword)					

FUNCTION DESCRIPTION:

General information:

The ROM of the process monitoring has to be tested cyclically. After one procedure is finished it starts again. The tested areas consist of the algorithm code, the mathematics library functions and the calibration data. The several parts are located in different ranges of the whole ROM of the MC. The test procedure jumps from one part to the next.

The checksum is calculated as an 64bit addition checksum by adding up double words (32bit) of the ROM areas. To be able to process the 64bit values, the values are splitted into low double word and high double word. In this subsection symbolic 64bit identifiers are used in the formula section.

Before every start of the procedure the actual values have to be initialized with the init values NC_ROM_CKS_ECU_INI and NC_ROM_CKS_CAL_INI.

In case of a cyclical ROM error the redundant switch off path and a fault reaction is carried out. The redundant switch off path is activated using the action defined in "The redundant switch off path". The redundant switch off path performs a reset via MU, if the fault reaction on MC fails due to existing faults. If the fault reaction on MC is performed correct, the redundant switch off path has no effect.

Application conditions:

Initialisation:

ROM_CKS_CAL_CLC = NC_ROM_CKS_CAL_INI

ROM_CKS_ECU_CLC = NC_ROM_CKS_ECU_INI

Recurrence: 40 ms (10ms in case of a ROM error and after power up)

Activation: always

After every cycle the actual values of ROM_CKS_ECU_CLC and ROM_CKS_CAL_CLC are compared with the setpoint values NC_ROM_CKS_ECU and C_ROM_CKS_CAL.

for production software:

IF ((ROM_CKS_ECU_CLC != NC_ROM_CKS_ECU) **OR**
(ROM_CKS_CAL_CLC!= C_ROM_CKS_CAL))

THEN

ACTION_ECM3_RedSwitchOffPath()

central failure reaction carried out

(refer to chapter: “Fault reaction of processor monitoring”)

ENDIF

Note: *Application software may only be used during the development process of a specific product. Under all circumstances it is strictly forbidden to use in any procuction ECU.*

for application software only:

IF ((ROM_CKS_ECU_CLC != NC_ROM_CKS_ECU) **OR**
(ROM_CKS_CAL_CLC!= C_ROM_CKS_CAL))
AND
(LC_SWI_ROM_CKS_CAL!= NC_SWI_ROM_CKS_CAL_DEAC)

THEN

ACTION_ECM3_RedSwitchOffPath()

central failure reaction carried out

(refer to chapter: “Fault reaction of processor monitoring”)

ENDIF

The calculation of the checksum at runtime must last shorter than the recover time NC_T_RST_CTR_MC of the reset counter RST_CTR_MC on MC. This is mandatory to enter the DISABLE state in case of resets caused by ROM failures.

Normally the time grid of the ROM test is 40 ms (NC_T_ROM_CKS_MON). If a reset is done, because the cyclic ROM test of process monitoring (level 2) or the standard ROM test of level

1 detected a failure, the time grid is switched to 10 ms. After power up the standard ROM test runs also in the fast mode. After the second reset in order the power stages are disabled for the time periode NC_T_DR_DI_RST_MC or NC_T_DR_DI_RST_MU (see "Fault reaction of processor monitoring"). Therefore it must be ensured, that the cyclical ROM test of Process Monitoring on MC has to be finished within this time periode.

The checksum for the calibration data has to be applied to C_ROM_CKS_CAL by checking the result of ROM_CKS_CAL_CLC. The memory area of C_ROM_CKS_CAL and LC_SWI_ROM_CKS_CAL is not a part of the test area.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_ROM_CKS_CAL_L	1	0...FFFFFFFFH	0...4294967295	1	[-]
Setpoint of ROM test process monitoring (low dword)					
C_ROM_CKS_CAL_H	1	0...FFFFFFFFH	0...4294967295	1	[-]
Setpoint of ROM test process monitoring (high dword)					
LC_SWI_ROM_CKS_CAL	1	0...FFH	0...255	1	[-]
Switch to set the fault of the CAL data ROM test inactive					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ROM_CKS_CAL_INI_L	1	0...FFFFFFFFH	0...4294967295	1	[-]
Init value of ROM_CKS_CAL_CLC (low dword)					
NC_ROM_CKS_CAL_INI_H	1	0...FFFFFFFFH	0...4294967295	1	[-]
Init value of ROM_CKS_CAL_CLC (high dword)					
NC_ROM_CKS_ECU_INI_L	1	0...FFFFFFFFH	0...4294967295	1	[-]
Init value of ROM_CKS_ECU_CLC (low dword)					
NC_ROM_CKS_ECU_INI_H	1	0...FFFFFFFFH	0...4294967295	1	[-]
Init value of ROM_CKS_ECU_CLC (high dword)					
NC_ROM_CKS_ECU_L	1	0...FFFFFFFFH	0...4294967295	1	[-]
Setpoint of ROM test process monitoring (low dword)					
NC_ROM_CKS_ECU_H	1	0...FFFFFFFFH	0...4294967295	1	[-]
Setpoint of ROM test process monitoring (high dword)					
NC_SWI_ROM_CKS_CAL_DEAC	1	0...FFH	0...255	1	[-]
Value of the switch to set the fault of the CAL data ROM test inactive (A5h means inactive)					
NC_T_ROM_CKS_MON	1	0...FFH	10...2560	10	[ms]
Time grid for how often the checksum calculation is called (3 means every 40ms)					

E.29.3.3 Cyclical ROM Test on the MU

Output data: (on MU)

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
CRC_CKS_CLC_MU	-	0...FFFFH	0...65535	1	[-]
Actual calculated CRC16-Checksum					

FUNCTION DESCRIPTION:

The ROM of the MU is used to store data tables and program. The CRC-16 Checksum is calculated in the background function of the operating system (no other functions need processor power). If the calculation is finished the actual calculated value

CRC_CKS_CLC_MU is compared with the setpoint value NC_CRC_CKS_MU. If no faults are indicated the checksum is initialized again and the calculation starts again.

IF (CRC_CKS_CLC_MU != NC_CRC_CKS_MU)

THEN

central fault reaction is carried out

(refer to chapter: “Fault reaction of processor monitoring”)

ELSE

CRC_CKS_CLC_MU = NC_CRC_CKS_INI_MU;

/* start new calculation */

ENDIF

Configuration data: (on MU)

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_ROM_START_ADR	1	0...FFFFH	0...65535	1	[-]
Start address of ROM-Test					
NC_ROM_STOP_ADR	1	0...FFFFH	0...65535	1	[-]
Stop address of ROM-Test					
NC_CRC_CKS_INI_MU	1	0...FFFFH	0...65535	1	[-]
Init value of CRC-Checksum					
NC_CRC_CKS_MU	1	0...FFFFH	0...65535	1	[-]
Setpoint CRC-Checksum					

An additional 8-bit-additions-checksum is calculated about the ROM area of the above-mentioned ROM-Test. This additional test is run cyclical too and in case of fault the central fault reaction is carried out, refer to chapter “Fault reaction of processor monitoring”.

Note: The calculation of the checksum at runtime must last shorter than the recover time NC_T_RST_CTR_MU of the reset counter RST_CTR_MU on MU. This is mandatory to enter the DISABLE state in case of resets caused by ROM failures.

E.29.4 RAM tests

E.29.4.1 Standard RAM Test on MC

FUNCTION DESCRIPTION:

The standard RAM test checks the whole RAM only during the initialisation after power up or reset. If an error is detected the central failure reaction is carried out (refer to chapter “Fault reaction of processor monitoring”). The fault reaction is either carried out directly, or the result of RAM test is saved as value and complement into different physical RAM areas and evaluated later.

The standard RAM test includes also the area of reset safe stored variables.

However, as the reset safe area of the RAM may not be changed after the test the original values have to be recovered.

Note: The standard RAM test at runtime must last shorter than the recover time NC_T_RST_CTR_MC of the reset counter RST_CTR_MC on MC. This is mandatory to enter the DISABLE state in case of resets caused by RAM failures. After the second reset in order the power stages are disable for the time periode NC_T_DR_DI_RST_MC or NC_T_DR_DI_RST_MU (see "Fault reaction of processor monitoring"). Therefore it must be ensured, that the standard RAM test on MC has to be finished within this time periode.

E.29.4.2 RAM test on MC for process monitoring

Export actions:

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
Writes the register value into RAM and cross checks the written value. The complement is generated and stored.
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
Compares the value of xyz_MON with its complement xyz_MON_CPL
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)
Read variable an its complement, checks value agains complement, register reg contains xyz_MON after action.

Note: All actions have to be implemented as macros to avoid function calls in the code of the *Process Monitoring*.

Descripiton for ACTION_ECM3_WriteChkCpl(...):

Description:

Syntax: ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>,
OUT <xyz_MON_CPL>, IN <reg>)

Parameter (IN):	type	data type of xyz_MON, xyz_MON_CPL and reg
	reg	register containing the value to be stored
Parameter (OUT):	xyz_MON_CPL	complement value of register/xyz_MON
Parameter (INOUT):	xyz_MON	variable to be written to RAM and checked

Formula section:

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>,
IN <reg>, SYNCHRONISATION CALL)

- Store register value *reg* into RAM, location *xyz_MON*
- Generate complement *xyz_MON_CPL* using the value of *reg* and store complement in RAM, location *xyz_MON_CPL*.
- Read variable *xyz_MON* from RAM and compare read value with register value *reg*:

IF (xyz_MON != reg)

THEN

central fault reaction is carried out

ENDIF

Descripiton for ACTION_ECM3_ChkCpl(...):

Description:

Syntax: ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Parameter (IN):	type	data type of *xyz_MON* and *xyz_MON_CPL*
	xyz_MON	variable to be checked, stored in RAM
	xyz_MON_CPL	complement value of *xyz_MON*, stored in RAM

Formula section:

ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>, SYNCHRONISATION CALL)

- Read variables *xyz_MON* and *xyz_MON_CPL* from RAM and compare read values:

  IF (*xyz_MON !*= complement(*xyz_MON_CPL*)) **THEN**

  central fault reaction is carried out

  ENDIF

Descripiton for ACTION_ECM3_ReadChkCpl(...):

Description:

Syntax: ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Parameter (IN) :	type	data type of *xyz_MON*, *xyz_MON_CPL* and *reg*
	xyz_MON	variable to be checked, stored in RAM
	xyz_MON_CPL	complement value of *xyz_MON*, stored in RAM
Parameter (OUT) :	reg	register containing value of *xyz_MON*

Formula section:

ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg> IN <xyz_MON>,
IN <xyz_MON_CPL>, SYNCHRONISATION CALL)

- Loads register *reg* with value stored in RAM, location *xyz_MON*
- Read variables *xyz_MON* and *xyz_MON_CPL* from RAM and compare read values:

  IF (*xyz_MON !*= complement(*xyz_MON_CPL*)) **THEN**

  central fault reaction is carried out

  ENDIF

FUNCTION DESCRIPTION:

General information:

The functionality of the process monitoring (level 2) is calculated cyclically. The calculation and the storage of intermediate results of the process monitoring functions is performed using the registers of the processor as far as possible. The registers are tested by the copy of process monitoring (level 3 and level 2' respectively, see "The test calculation Copy of process monitoring").

All variables located in the RAM of the process monitoring used in the next recurrence of the task (e.g. anti-bounce-counter) or in a successive task are stored in original and complementary form in the normal RAM area. After execution of a *Process Monitoring* module the results have to be copied from registers into variables located in the RAM. After the storage of a variable from an internal register to the RAM, the variable is loaded from RAM again and compared with the original value of the register. This procedure is done with the action ACTION_ECM3_WriteChkCpl(...) given above.
At the beginning of the following recurrence of the task, the two complementary variables are checked for plausibility. Also intermediate results, which can not be completely handled by the internal registers, are stored complementary and checked for plausibility. This is done at the begining of a *Process Monitoring* module using the action ACTION_ChkCpl(...) or ACTION_ECM3_ReadChkCpl(...). Hereby faults of the RAM cells and an undesired overwriting of the data by a faulty function will be detected and a fault reaction will be carried out without debouncing, see chapter “Fault reaction of processor monitoring”.

Application conditions:

Activation and Deactivation: According to the respective function modules.

Initialization: Before the first test of the complementary variables, the variables must be properly initialized using the action ACTION_ECM3_WriteChkCpl( ).

E.29.4.3 Standard RAM Test on MU

Output data: (on MU)

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
RAM_TEST_MU	-	0...FFH	0...255	1	[-]
Result of RAM test					
RAM_TEST_MU_CPL	-	0...FFH	0...255	1	[-]
Complement result of RAM test					

FUNCTION DESCRIPTION:

The RAM of the MU is used as a data storage area and a stack area for subroutine call and interrupt occurrences. The RAM is tested after a power-up-reset. After a software reset specific data are stored in a reset safe RAM area. These data are not allowed to overwrite through the RAM-Test. Therefore only the other area is tested after a software reset. After a power up also the reset safe RAM area is tested. The result of the RAM test is stored in RAM_TEST_MU and its complement RAM_TEST_MU_CPL. If RAM_TEST_MU or RAM_TEST_MU_CPL indicates a fault a fault reaction is carried out, refer to chapter “Fault reaction of processor monitoring”.

Note: The standard RAM test at runtime must last shorter than the recover time NC_T_RST_CTR_MU of the reset counter RST_CTR_MU on MU. This is mandatory to enter the DISABLE state in case of resets caused by RAM failures. After the second reset in order the power stages are disable for the time periode NC_T_DR_DI_RST_MC or NC_T_DR_DI_RST_MU (see "Fault reaction of processor monitoring"). Therefore it must be ensured, that the standard RAM test on MU has to be finished within this time periode.

Configuration data: (on MU)

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_RAM_ST_ADR	1	0...FFFFH	0...65535	1	[-]
Start address of RAM-Test					
NC_RAM_STOP_ADR	1	0...FFFFH	0...65535	1	[-]
Stop address of RAM-Test					

E.29.5 The test calculation Copy of process monitoring

Output data: (on MC to MU)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DR_OFF_MU_MON	V/O	0...1H	0...1	1	[-]
Bit for redundant switch off path (1=active)					
RESP_MON2_MC	-	0...FFH	0...255	1	[-]
Decoded answer of the test calculation, transferred to MU					

Input data: (on MC)

ABC_MON2_MU	TEST_REC_IDX_MON2	LV_DR_OFF_MU_MON
LDPM_TEST_REC_IDX_MON2	ID_RESP_MON2	

Output data: (on MU)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ABC_MON2_MU	V/O	0...FFH	0...255	1	[-]
Anti-bounce counter of level 2' test calculation, evaluated on MU, transmitted to MC					
TEST_REC_IDX_MON2	V/O	0...3FH	0...63	1	[-]
Start value for the test record of Copy of Process Monitoring					

Input data: (on MU)

LV_DR_OFF_MU_MON	RESP_MON2_MC		

FUNCTION DESCRIPTION:

Note: The interface variables between level3 and level2' are described below in "Interface to Copy of Process Monitoring" detailed.

Application conditions:

Initialisation: ABC_MON2_MU=0 after reset or power up

General information:

In order to ensure the correct computation and processing on MC, a **test of the process monitoring** is performed in the NORMAL state, see Figure 74 or section "Communication between MC and MU". To validate the results it is neccesarry to compare the calculated result with a known, fixed result. Therefore the given start values are fixed at runtime. The calculations themselves are done with a copy of process monitoring (level 2'). The interface to this level is a start value (index number TEST_REC_IDX_MON2) and the 16-bit answer. This result RESP_MON2 of level 2' affects the final answer RESP_MON2_MC which is sent to MU. On MU this answer is compared with the expected result.

The mapping of the test index to the expected values must be one to one. Note, that the range of the index value generated on MU does not overlap the range of the expected result transferred to MU at all.

The start value TEST_REC_IDX_MON2 is transferred from MU to MC, see “Communication between MC and MU”.

The task level 2’ is executed every second communication cycle, so the result is evaluated on MU also every second communication cycle.

If the monitoring level of the process monitoring (see “Fault Reaction of Process Monitoring”) or the ROM test (see “Cyclical ROM Test of Process Monitoring on MC”) detects a critical error the copy of process monitoring is forced to give wrong answers. This mechanism, called the redundant switch off path, leads to a disable of the safety relevant power stages by the monitoring unit and is activated with set LV_DR_OFF_MU_MON. (refer to subsection "The redundant switch off path"). The variable LV_DR_OFF_MU_MON is monitored here with a recurrence of 2*NC_T_SPI_COM using the cyclical RAM test functions ("RAM test on MC for process monitoring").

For correct error memory management, it is mandatory to distinguish between the use of the redundant switch off path and wrong answers caused by wrong results of level 2’, see “Fault reaction of processor monitoring”.

Specifications of the test calculation evaluation on MU:

MU determines the test index TEST_REC_IDX_MON2. This index runs in default case (no faults, no exeptions) from the initial value 0 to NC_TEST_REC_IDX_MAX_MON2-1 with an increment of one. If the maximum value is reached, the index starts again with zero.

Exception: If the system recognizes that a test calculation provides an incorrect result, the same test calculation is repeated with the same input value, either until the error has been recognized reliably, or until it is ensured that there is no error (ABC_MON2_MU= 0).

As the result is evaluated on the MU, the repetition of the incorrect test calculation can only be initiated one step later. Thus, the MU requests only one time another calculation between the incorrect calculation and its repetition.

Example: Result of test calculation 2 is permanetly incorrect:

Sequence: 0,1,2,3,2,2,2,2.....

Evaluation of the answer:

The MU compares the result RESP_MON2_MC obtained with the corresponding value from the ROM. If the answer is correct, TEST_REC_IDX_MON2 is incremented and the anti bounce counter ABC_MON2_MU is decremented until zero. If not, the anti-bounce counter ABC_MON2_MU is incremented by NC_ABC_INC_MON2_MU. The maximum value of ABC_MON2_MU is limited to NC_ABC_MAX_MON2_MU.

Exeption: If there are two wrong answers in series, the second wrong answer is not evaluated, therefore the value of the anti bounce counter stays unchanged. This is mandatory to do a correct debouncing.

If ABC_MON2_MU>= NC_ABC_MAX_MON2_MU, the central fault reaction is carried out (refer to chapter: “Fault reaction of processor monitoring”) and ABC_MON2_MU is no longer incremented.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ABC_MAX_MON2_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of level 2' test calculations on MU					
NC_ABC_INC_MON2_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of level 2' test calculations on MU					
NC_TEST_REC_IDX_MAX_MON2	1	0...FH	0...15	1	[-]
(NC_TEST_REC_IDX_MAX_MON2-1) is the maximum value for the index of test calculation Therfore NC_TEST_REC_IDX_MAX_MON2 answers are calculated cyclically					

E.29.5.1 Inspection of MU by the MC according to test calculation

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
ABC_MON2_TEST_MC	-	0...FFH	0...255	1	[-]
Anti-bounce counter to test anti-bounce counter of level2' test calculation on MU					
ABC_MON2_TEST_2_MC	-	0...FFH	0...255	1	[-]
Anti-bounce counter to test the incrementation of index for level2' test calculation on MU					

FUNCTION DESCRIPTION:

Application conditions:

Initialisation: ABC_MON2_TEST_MU =ABC_MON2_TEST_2_MC=0 after reset or power up

The anti bounce counter ABC_MON2_MU of the copy of process monitoring is transferred from MU to MC. The MC tests cyclically in time periodes of NC_T_MON2_TEST_MC whether the anti-bounce counter of the MU is incremented by returning an incorrect result RESP_MON2_MC.

The system checks at time periodes of NC_T_MON2_TEST_MC whether the MU's anti-bounce counter ABC_MON2_MU is zero. If so, an incorrect result RESP_MON2_MC is sent from the MC to the MU as the result of the next test calculation. Then the MC checks whether the anti-bounce counter evaluated on MU and sent from MU to MC has changed. If the anti-bounce counter remains zero, incorrect results of the corresponding test calculations are sent from the MC to the MU cyclically, until the anti-bounce counter of the MU responds.

If the anti-bounce counter ABC_MON2_MU of the MU responds, the anti-bounce counter ABC_MON2_TEST_MC on MC is decremented until zero, otherwise the anti-bounce counter ABC_MON2_TEST_MC is incremented by the increment value NC_ABC_INC_MON2_TEST_MC. If the anti-bounce counter reaches its maximum value NC_ABC_MAX_MON2_TEST_MC, a central failure reaction carried out (refer to chapter: “Fault reaction of processor monitoring”). The value of ABC_MON2_TEST_MC is limited to NC_ABC_MAX_MON2_TEST_MC.

Furthermore the MC tests if the question index TEST_REC_IDX_MON2 determined by MU is changed as long as the anti bouncer ABC_MON2_MU is zero. In case of a detected error according to the question index the anti bounce counter ABC_MON2_TEST_2_MC is incremented by NC_ABC_INC_MON2_TEST2_MU, otherwise the anti bounce counter is decremented until zero. If the anti bounce counter in the MC reaches the maximum value NC_ABC_MAX_MON2_TEST_2_MU, a central failure reaction carried out (refer to chapter:

"Fault reaction of processor monitoring"). The value of ABC_MON2_TEST_2_MC is limited to NC_ABC_MAX_MON2_TEST_2_MC.

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_ABC_INC_MON2_TEST_MC	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter to test the anti-bounce counter of level 2' test calculations on MU					
NC_ABC_MAX_MON2_TEST_MC	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter to test the anti-bounce counter of level 2' test calculations on MU					
NC_ABC_INC_MON2_TEST_2_MC	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter to test the incrementation of level 2' test calculation index on MU					
NC_ABC_MAX_MON2_TEST_2_MC	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter to test the incrementation of level 2' test calculation index on MU					
NC_T_MON2_TEST_MC	1	0...FFH	0...510	2	[-]
Time base for inspection the evaluation of level2' test calculations, measured as multiple of NC_T_SPI_COM					

E.29.5.2 Interface to Copy of Process Monitoring

Output data: (from level 3 to level2')

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
TEST_REC_IDX_MON2	V/O	0...3FH	0...63	1	[-]
Start value for the test record of Copy of Process Monitoring					
CONV_RESP_MON2	-	0...FH	0...15	1	[-]
Intermediate result of answer decoder					
RESP_MON2_MC	-	0...FFH	0...255	1	[-]
Decoded answer of the test calculation, transferred to MU					

Input data: (from level2' to level3)

RESP_MON2	ID_RESP_MON2	LDPM_TEST_REC_IDX_MON2

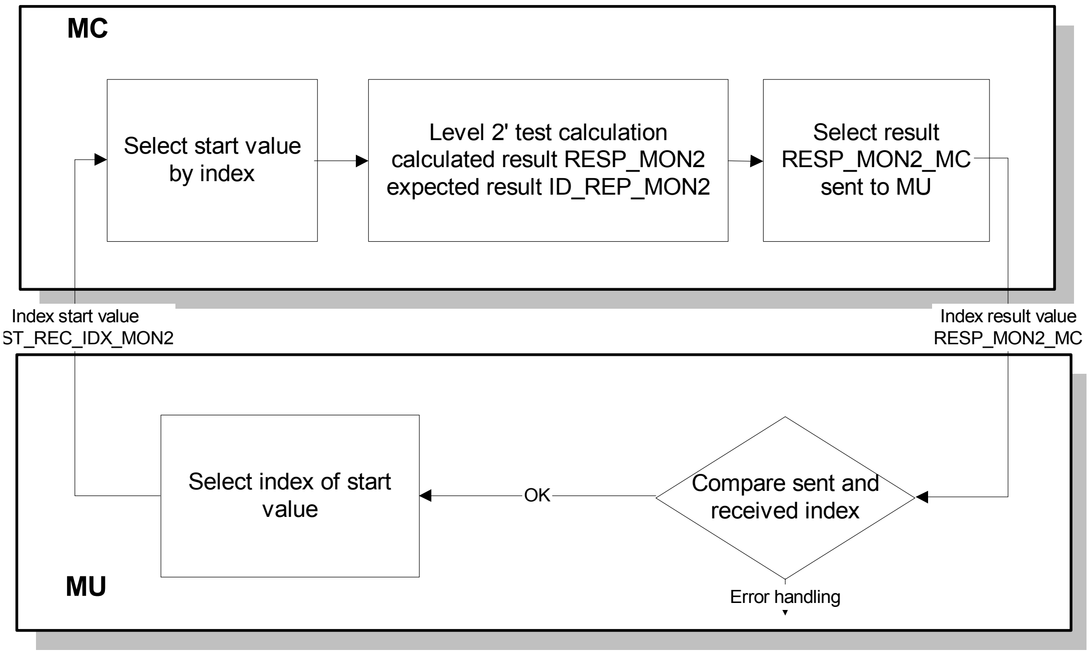


Figure 75: Interface to copy of process monitoring

FUNCTION DESCRIPTION:

As specified above the task copy of process monitoring (level 2') is called cyclically. The interface between level 3 and level 2' is performed in the following steps:

1. The input for level 2' is the start value TEST_REC_IDX_MON2.
2. The result of level 2' is the 16-bit answer RESP_MON2.
3. The value of RESP_MON2 has to be found in the array ID_RESP_MON2. This is done in the answer decoder.

TEST_REC_IDX_MON2 —level 2'→ RESP_MON2 —answer decoder→ CONV_RESP_MON2

4. The intermediate result CONV_RESP_MON2 and the scattering of wrong answers (see "Inspection of MU by the MC according to test calculation") forms the final result RESP_MON2_MC which is transferred to MU.

If the answer RESP_MON2 calculated by level 2' is incorrect, the value of RESP_MON2_MC indicates an error independent of the inspection of MU by the MC. Note also that the initial value of RESP_MON2_MC when entering the answer decoder indicates an error.

Note, that the calibration data ID_RESP_MON2 and LDPM_TEST_REC_IDX_MON2 are imported from "Copy of Process Monitoring". This data have to be calibrated in "Copy of Process Monitoring", they are needed here to evaluate the intermediate result CONV_RESP_MON2.

E.29.5.3 The redundant switch off path

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_DR_OFF_MU_MON	V/O	0...1H	0...1	1	[-]
Bit for redundant switch off path (1=active)					
LV_DR_OFF_MU_MON_CPL	O	0...1H	0...1	1	[-]
Complement bit of redundant switch off path					

Export actions:

ACTION_ECM3_RedSwitchOffPath()
Activates the redundant switch off path

Note: This action has to be implemented as macros to avoid function calls in the code of the *Process Monitoring*.

FUNCTION DESCRIPTION:

This function activates the redundant switch off path. Therefore LV_DR_OFF_MU_MON is set and the copy of process monitoring (see “The test calculation Copy of process monitoring”) is forced to give wrong answers and transferred to MU next and the following communication. In “The test calculation Copy of process monitoring” also the cyclical RAM test of LV_DR_OFF_MU_MON is performed using the functions defined in "RAM test on MC for process monitoring"; the value LV_DR_OFF_MU_MON and LV_DR_OFF_MU_MON_CPL are checked for consitency. The recurrence is NC_T_SPI_COM.

LV_DR_OFF_MU_MON is transferred to MU only to ensure a correct error memory management (see “Fault reaction of processor monitoring”).

Descripiton for ACTION_ECM3_RedSwitchOffPath():

Description:

Syntax: ACTION_ECM3_RedSwitchOffPath()

Parameter: none

Formula section:

ACTION_ECM3_RedSwitchOffPath(SYNCHRONISATION CALL)

- Sets the variable LV_DR_OFF_MU_MON using ACTION_ECM3_WriteChkCpl()

Application conditions:

Initialisation: LV_DR_OFF_MU_MON = 0 /* using ACTION_ECM3_WriteChkCpl() */

Recurrence: only after power up or SW reset

E.29.6 Watchdog on the monitoring unit

FUNCTION DESCRIPTION:

General information:

A timer is used to generate a watchdog functionality on the monitoring unit. A software watchdog is used to ensure that endless program loops on the MU are detected. The operating system of the MU resets the watchdog timer cyclically. If the watchdog timer hasn't been reset for 100ms, the safety critical power stages are disabled and a software reset on MC is triggered via the NMI line. This causes a restart of the ECU.

E.29.7 Program Flow Monitoring (PFM)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
RESP_PFM_MC	-	0...FFH	0...255	1	[-]
Answer of the program flow monitoring					

Input data:

SYN_PFM_MU			

FUNCTION DESCRIPTION:

It is intended to achieve a program flow monitoring and to implement this in terms of an „intelligent" watchdog. This is to ensure that safety-related-program sections have been executed demonstrably in the expected time grid and also in the given sequence. Therefore modules of level2 (process monitoring) are monitored in the NORMAL state, see Figure 74 or section "Communication between MC and MU". The identifier byte RESP_PFM_MC is transferred to MU:

RESP_PFM_MC and SYN_PFM_MU:

7 (MSB)	6	5	4	3	2	1	0 (LSB)
Task 8	Task 7	Task 6	Task 5	Task 4	Task 3	Task 2	Task 1

Task *n* (*n* = 1 ... 8) is running in the nominal time gird NC_T_PFM_*n*_MU. The identifier RESP_PFM_MC is only generated in the MC, but not checked. After transfer to the MU each communication it is tested on MU. Here the word task is used for a given, fixed sequence of modules running in the same nominal time grid.

The following algorithm is used:

The MU generates a synchronization byte (SYN_PFM_MU) which is transferred to the MC. SYN_PFM_MU has the same binary structure as RESP_PFM_MC. SYN_PFM_MU is evaluated by MC to ensure, that the MU is still alive. Each bit in SYN_PFM_MU is inverted and used to initialize the dedicated bit in RESP_PFM_MC. Calling the modules to be program flow monitored influences finally the bits in RESP_PFM_MC. The final RESP_PFM_MC is transferred unchanged to MU in the following communication cycle and is evaluated there.

The n^{th} bit in RESP_PFM_MC is initialized within ACTION_ECM3_ServicePfm(*n*). Furthermore this action is used to initialize a specific bit pattern assigned to the task *n* (see stucture of RESP_PFM_MC and SYN_PFM_MU). This internal bit pattern is manipulated by ACTION_ECM3_Service*k*TaskPfm(*n*) at the entrance and the exit of each module to be program flow monitored within a given sequence of several mudules. After the execution of all modules this internal bit pattern is evaluated in ACTION_ECM3_ServicePfm() to form the n^{th} bit in RESP_PFM_MU. The available actions are desctibed later.

E.29.7.1 Inspection of MU by the MC on PFM

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ABC_PFM_TEST_MC	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring					

Input data:

SYN_PFM_MU			

FUNCTION DESCRIPTION:

General information:

Because this inspection is a test of the algorithm itself, it is needful to check the bits belonging to the task running in the shortest time gird only.

Application conditions:

Initialisation: ABC_PFM_TEST_MC=0 after power up or reset

The MC checks if the correct bits of SYN_PFM_MU are toggled by MU. If they are not toggled, the anti-bounce counter ABC_PFM_TEST_MC is incremented with NC_ABC_INC_PFM_TEST_MC. If ABC_PFM_TEST_MC gets bigger than or equal to ABC_MAX_PFM_TEST_MC, the central failure reaction carried out (refer to chapter: "Fault reaction of processor monitoring"). If no error is detected, ABC_PFM_TEST_MC is decremented until zero. ABC_PFM_TEST_MC is limited to ABC_MAX_PFM_TEST_MC.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ABC_INC_PFM_TEST_MC	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU					
NC_ABC_MAX_PFM_TEST_MC	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU					

E.29.7.2 Inspection of MC by the MU on PFM

Output data: (on MU)

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SYN_PFM_MU	-	0...FFH	0...255	1	[-]
Synchronization byte of the program flow monitoring					
ABC_PFM_1_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 1					
ABC_PFM_2_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 2					
ABC_PFM_3_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 3					
ABC_PFM_4_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 4					
ABC_PFM_5_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 5					
ABC_PFM_6_MU	-	0...FFH	0...255	1	[-]

Anti-bounce counter of the program flow monitoring for task 6					
ABC_PFM_7_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 7					
ABC_PFM_8_MU	-	0...FFH	0...255	1	[-]
Anti-bounce counter of the program flow monitoring for task 8					

Note: The value of the variable SYN_PFM_MU is set on MU and has to be transferred to MC via the internal communication.

Input data: (on MU)

RESP_PFM_MC			

Note: The value of this variable is set on MC and has to be transferred to MU via the internal communication.

FUNCTION DESCRIPTION:

Application conditions:

Initialisation: ABC_PFM_n_MU=NC_ABC_INI_PFM_n_MU (n=1 ... 8) after power up or reset

RESP_PFM_MC is sent to the MU each communication. The bits of the according time levels are toggled at dedicated time intervals. SYN_PFM_MU sent to the MC must be saved for the next run. The currently determined RESP_PFM_MC and SYN_PFM_MU can be used to verify the program flow monitoring received from the MC.

In case of error, the appropriate anti-bounce counters ABC_PFM_n_MU (n=1 ... 8) are incremented with NC_ABC_INC_PFM_n_MU. If the anti-bounce counter in the MU gets bigger than or equal to the maximum value NC_ABC_MAX_PFM_n_MU, the central failure reaction is carried out (refer to chapter: “Fault reaction of processor monitoring”). If no error is detected, ABC_PFM_n_MU is decremented until zero. ABC_PFM_n_MU is limited to NC_ABC_MAX_PFM_n_MU.

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_ABC_INC_PFM_1_MU	-	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 1					
NC_ABC_MAX_PFM_1_MU	-	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 1					
NC_ABC_INI_PFM_1_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 1					
NC_T_PFM_1_MU	1	0...FFH	0...255	1	[-]
Time grid of task 1, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_1_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 1					
NC_ABC_INC_PFM_2_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 2					
NC_ABC_MAX_PFM_2_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 2					
NC_ABC_INI_PFM_2_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 2					
NC_T_PFM_2_MU	1	0...FFH	0...255	1	[-]
Time grid of task 2, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_2_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 2, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_3_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 3					
NC_ABC_MAX_PFM_3_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 3					
NC_ABC_INI_PFM_3_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 3					
NC_T_PFM_3_MU	1	0...FFH	0...255	1	[-]
Time grid of task 3, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_3_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 3, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_4_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 4					
NC_ABC_MAX_PFM_4_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 4					
NC_ABC_INI_PFM_4_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 4					
NC_T_PFM_4_MU	1	0...FFH	0...255	1	[-]
Time grid of task 4, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_4_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 4, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_5_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 5					
NC_ABC_MAX_PFM_5_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 5					
NC_ABC_INI_PFM_5_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 5					
NC_T_PFM_5_MU	1	0...FFH	0...255	1	[-]
Time grid of task 5, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_5_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 5, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_6_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 6					
NC_ABC_MAX_PFM_6_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 6					
NC_ABC_INI_PFM_6_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 6					

NC_T_PFM_6_MU	1	0...FFH	0...255	1	[-]
Time grid of task 6, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_6_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 6, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_7_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 7					
NC_ABC_MAX_PFM_7_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 7					
NC_ABC_INI_PFM_7_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 7					
NC_T_PFM_7_MU	1	0...FFH	0...255	1	[-]
Time grid of task 7, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_7_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 7, measured as multiple of NC_T_SPI_COM					
NC_ABC_INC_PFM_8_MU	1	0...FFH	0...255	1	[-]
Increment value of anti-bounce counter of program flow monitoring on MU for task 8					
NC_ABC_MAX_PFM_8_MU	1	0...FFH	0...255	1	[-]
Maximum value of anti-bounce counter of program flow monitoring on MU for task 8					
NC_ABC_INI_PFM_8_MU	1	0...FFH	0...255	1	[-]
Init value of anti-bounce counter of program flow monitoring on MU for task 8					
NC_T_PFM_8_MU	1	0...FFH	0...255	1	[-]
Time grid of task 8, measured as multiple of NC_T_SPI_COM					
NC_T_CHK_PFM_8_MU	1	0...FFH	0...255	1	[-]
Time to check PFM bits of task 8, measured as multiple of NC_T_SPI_COM					

E.29.7.3 Program Flow Monitoring services for Process Monitoring

Export actions:

ACTION_ECM3_Service0TaskPfm(IN <n>)
Action call to be done right before execution of 1st module to be program flow monitored
ACTION_ECM3_Service1TaskPfm(IN <n>)
Action call to be done right after execution of 1st module to be program flow monitored
ACTION_ECM3_Service2TaskPfm(IN <n>)
Action call to be done right before execution of 2nd module to be program flow monitored
ACTION_ECM3_Service3TaskPfm(IN <n>)
Action call to be done right after execution of 2nd module to be program flow monitored
ACTION_ECM3_Service4TaskPfm(IN <n>)
Action call to be done right before execution of 3rd module to be program flow monitored
ACTION_ECM3_Service5TaskPfm(IN <n>)
Action call to be done right after execution of 3rd module to be program flow monitored
ACTION_ECM3_Service6TaskPfm(IN <n>)
Action call to be done right before execution of 4th module to be program flow monitored
ACTION_ECM3_Service7TaskPfm(IN <n>)
Action call to be done right after execution of 4th module to be program flow monitored
ACTION_ECM3_Service8TaskPfm(IN <n>)
Action call to be done right before execution of 5th module to be program flow monitored
ACTION_ECM3_Service9TaskPfm(IN <n>)
Action call to be done right after execution of 5th module to be program flow monitored
ACTION_ECM3_Service10TaskPfm(IN <n>)
Action call to be done right before execution of 6th module to be program flow monitored
ACTION_ECM3_Service11TaskPfm(IN <n>)
Action call to be done right after execution of 6th module to be program flow monitored
ACTION_ECM3_Service12TaskPfm(IN <n>)
Action call to be done right before execution of 7th module to be program flow monitored
ACTION_ECM3_Service13TaskPfm(IN <n>)
Action call to be done right after execution of 7th module to be program flow monitored
ACTION_ECM3_Service14TaskPfm(IN <n>)
Action call to be done right before execution of 8th module to be program flow monitored
ACTION_ECM3_Service15TaskPfm(IN <n>)
Action call to be done right after execution of 8th module to be program flow monitored
ACTION_ECM3_Service16TaskPfm(IN <n>)
Action call to be done right before execution of 9th module to be program flow monitored
ACTION_ECM3_Service17TaskPfm(IN <n>)
Action call to be done right after execution of 9th module to be program flow monitored
ACTION_ECM3_Service18TaskPfm(IN <n>)
Action call to be done right before execution of 10th module to be program flow monitored
ACTION_ECM3_Service19TaskPfm(IN <n>)
Action call to be done right after execution of 10th module to be program flow monitored
ACTION_ECM3_ServicePfm(IN <n>)
Action call to determine result of PFM which will be sent to the MU

Description of ACTION_ECM3_Service*k*TaskPfm(...), *k=0 ... 19*:

Description:

Syntax: ACTION_ECM3_ServicekTaskPfm(IN <*n*>)

Parameter (IN): *n* variable name of the bitpattern to be toggled

Parameter (OUT): -

Formula section:

ACTION_ECM3_Service*k*TaskPfm(IN <*n*>, SYNCHRONISATION CALL)

Changes bits in the bit pattern given by name *n* belonging to task *n*.

Description of ACTION_ECM3_ServicePfm(...):

Description:

Syntax:	ACTION_ECM3_ServicePfm(IN <*n*>)	
Parameter (IN):	*n*	variable name of the bitpattern to be toggled
Parameter (OUT):	-	

Formula section:

ACTION_ECM3_ServicePfm(IN <*n*>, SYNCHRONISATION CALL)

Calculates the final result of PFM which will be sent to the monitoring unit

General information:

Up to ten level 2 modules can be program flow monitored by the *Processor Monitoring* in one time grid. Therefore it is mandatory to sandwich the modules between services defined here. One service action call has to be done *before* the execution and another one right *after* the execution of a level 2 module instruction. Therefore the service actions ACTION_ECM3_Service*k*TaskPfm(*n*) are implemented. For the level 2 module *j* (with j=0 ... 9) the service action call ACTION_ECM3_Service*2j*TaskPfm(*n*) has to be done before the execution of the first instruction of the module. After the executing of the last instruction of the module the service action call ACTION_ECM3_Service*(2j+1)*TaskPfm(*n*) has to be done.

Finally at the end of all level 2 modules the answer bits in RESP_PFM_MC has to be determined using the service ACTION_ECM3_ServicePfm(*n*).

The agrument *n* belongs to the bits in RESP_PFM_MC and SYN_PFM_MU.

E.29.8 Interface to monitoring of A/D conversion

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ACQ_MC_0_MON	V/O	0...3FFH	0...4.99511	4.8828e-3	[V]
AD-value of the MC, monitored with RAM test function					
ACQ_MC_0_MON_CPL	O	0...3FFH	0...4.99511	4.8828e-3	[V]
Complement of AD-value of MC					
ACQ_MU_0_MON	V/O	0...3FFH	0...4.99511	4.8828e-3	[V]
AD-value of MU, monitored with RAM test function					
ACQ_MU_0_MON_CPL	O	0...3FFH	0...4.99511	4.8828e-3	[V]
Complement AD-value of MU					

Input data:

AN_BUF_0			

Note: This value is a direct input from the intenal communication receive buffer.

FUNCTION DESCRIPTION:

The ECU itself provides the possibility to monitor *all safety critical* A/D conversions. The A/D converters on both controllers, MC and MU, as well as the communication between MC and MU are involved. The digitalisation of *all safety critical* analog inputs on one controller is done with the *same* converter. This is realized with an internal mulitplexed input on both controllers, MC and MU. To do the monitoring of A/D conversion, it is mandatory that *one* analog input on MC is the same as the dedicated analog input on MU. This digitalized value has to be transmitted via the internal communication (for details refer to chapter “Communication between MC and MU”). By comparing one value (digitalized on MC) with the dedicated value (digitalized on MU) a monitoring of the A/D conversion of all saftey critical A/D values is possible. Conversion time as well as communication time has to be taken into account when comparing non constant values. Therefore one AD-channel of the MU (see “Analog and pseudo digital inputs”) is converted directly after the communication and transferred to MC.

On MC the received value is loaded in case of fault free communication from the receive buffer directly into a register. With the action ACTION_ECM3_WriteChkCpl( ) defined in "RAM test on MC for process monitoring" the variables ACQ_MU_0_MON and ACQ_MU_0_MON_CPL are stored and checked.

The ADC value of the dedicated channel on the MC is directly copied from the ADC readout into some intermediate variables. The storage of the variables has to be done with value and complement (like in "RAM test on MC for process monitoring") to provide a consitency check when evaluating the result. Like MU the MC converts the dedicated analog signal after the communication. The elapsed time between the measurements on MC and MU has to be less than 1ms.

After a fault free communication a register is loaded with the intermediate stored value. This register is used to store the value into ACQ_MC_0_MON and ACQ_MC_0_MON_CPL using the action ACTION_ECM3_WriteChkCpl( ).

For details on signal processing (based on ACQ_MU_0_MON and ACQ_MC_0_MON) see chapter “Monitoring of A/D conversion”. When passing the output signal ACQ_MC_0_MON and ACQ_MU_0_MON to the signal processing subroutine ("Monitoring of A/D conversion") they have to be checks with ACTION_ECM3_ReadChkCpl() against their complements. This is mandatory to do a correct RAM test.

Faults of the internal multiplexers are not monitored directly. As the redundancy of ACQ_MC_0_MON and ACQ_MU_0_MON is very important, the basic software for reading out the ADCs has to be coded by two different persons working independent of each other to provide diversy signal paths.

If there are more than one ADC available on MC or MU, the saftey relevant analog signals have to be converted on monitoried the ADCs.

Application conditions:

Initialisation: ACQ_MU_0_MON = ACQ_MC_0_MON = 0 using the ACTION_ECM3_WriteChkCpl()

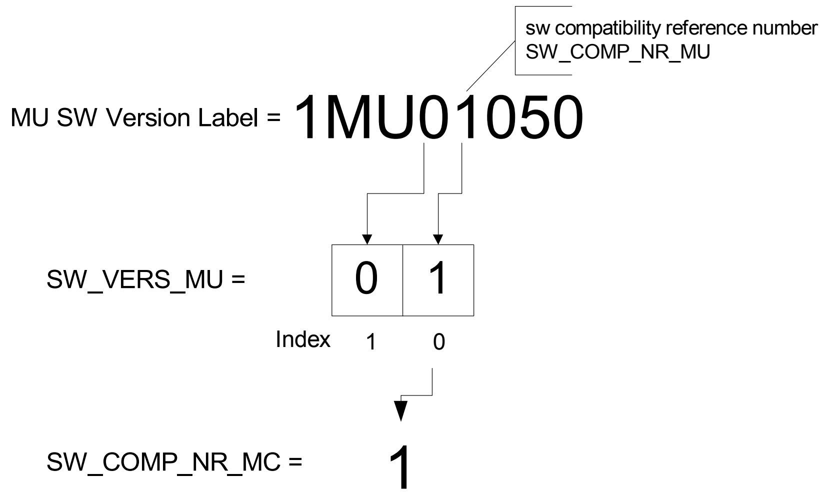
Recurrence: after each power up or reset

Activation: always

E.29.9 Software Reference Check

FUNCTION DESCRIPTION:

Each controller, MU and MC, has its own software version label which consists of characters. The software version reference SW_VERS_MU of the MU and the software compatibility number SW_COMP_NR_MC of MC are transmitted in the first communication, see chapter “Communication between MC and MU”. A software reference check is done on MC and MU.



Note: The given value of the Version Label is an example only.

The four characters of a software version referencene of MU has the following meaning:

Index	
1	Mask number
0	Software compatibility reference number

E.29.9.1 Software Reference Check on MC

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
SW_COMP_NR_MC	V	0...3FH	0...63	1	[-]
Software compatibility reference number of MC, sent to MU in the first communication					

Note: The value of the variable SW_COMP_NR_MC is set on MC and has to be transferred to MU via the intenal communication.

Input data:

SW_VERS_MU[NC_SW_VERS_MU_NR]			

Note: The value of this variable is set on MU and has to be transferred to MC via the internal communication.

The software compatibility reference number SW_COMP_NR_MC (one character) is initialized on MC with NC_SW_COMP_NR_MC. The MC compares SW_COMP_NR_MC with the software compatibility number of the MU after the first communication.

Formula section:

IF (SW_COMP_NR_MC <> SW_VERS_MU[0])

THEN

central failure reaction carried out

(refer to chapter: "Fault reaction of processor monitoring")

ENDIF

Configuration data:

Name	Dim	Hex. limits	Phys. Limits	Resol.	Unit
NC_SW_COMP_NR_MC	1	0...3FH	0...63	1	[-]
Software version label of MC					

E.29.9.2 Software Reference Check on MU

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
SW_VERS_MU[NC_SW_VERS_MU_NR]	V	0...3FH	0...63	1	[-]
software version label of MU					

Note: The value of this variable is set on MU and has to be transferred to MC via the internal communication. After the communication it is available on MC.

Input data: (on MU)

SW_COMP_NR_MC			

Note: The value of this variable is set on MC and has to be transferred to MU via the internal communication.

FUNCTION DESCRIPTION:

The characters of the data SW_VERS_MU on MU are initialised with the MU internal software version label NC_SW_VERS_MU. The two characters are transferred from MU to to MC in the INIT communication and stored in SW_VERS_MU on MC.

The internal software version reference number NC_SW_VERS_MU[0] of the MU is compared with the software compatibility number SW_COMP_NR_MC of MC after the INIT communication on MU:

IF (SW_COMP_NR_MC <> NC_SW_VERS_MU[0])

THEN

central fault reaction is carried out

(refer to chapter: "Fault reaction of processor monitoring")

ENDIF

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_SW_VERS_MU[NC_SW_VERS_MU_NR]	1	0...3FH	0...63	1	[-]
Software version reference of the MU, two bits are configurated via pins					
NC_SW_VERS_MU_NR	1	0...3FH	0...63	1	[-]
Dimension of the array SW_VERS_MU					

E.29.9.3 Pin configuration test

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
MU_PIN_CONF	V	0...3H	0...3	1	[-]
Pin configuration of the variant coding of the MU					

General information:

During the initialization the MU reads two of its input pins, which are used for variant coding. Depending on this pin configuration, the MU runs different software. The read pins are transferred together with SW_VERS_MU[0] to the main controller. A pin configuration test is done on both controllers, like a Software Reference Check.

Each controller carries out its fault reaction, if the pin configuration indicates a fault, see also "Fault reaction of processor monitoring".

E.29.10 Switch off path activation due to endless resets on MC

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
ERR_COD_MC	V/O/S	0...FFH	0...255	1	[-]
Error code , which represents a failure detected by MC					

Note: ERR_COD_MC has to be located in the software reset safe RAM area; the exact meaning of the error codes can be found in “Fault reaction of processor monitoring.”

Input data:

RST_CTR_MC	ERR_COD_MU		

Additionally, information from the reset status register of MC is used.

FUNCTION DESCRIPTION:

General information:

In the presence of endless resets at ‘low frequency,’ the process monitoring may not be able to perform a fault reaction but the engine may produce high torque; in this context, ‘low frequency’ means that the resets occur during the fault reaction time of process monitoring.

In order to reduce the torque, the powerstages shall be disabled for as long as process monitoring is unable to react to any application software fault. This shall be achieved by triggering the redundant switch off path when endless resets are detected, refer to section “The redundant switch off path”.

Additionally, in order to distinguish this behaviour from a fault reaction by the process monitoring, a dedicated error code on the MC is set, see "Fault reaction of processor monitoring.”

Application conditions:

Initialisation: none

Recurrence: once

Activation: after each INIT communication (i.e., when ERR_COD_MU has become available)

Formula section:

IF (RST_CTR_MC = 0) **AND** (ERR_COD_MU = 0) **AND**

(reset status register on MC indicates a reset that has been triggered by the MC)

THEN

trigger redundant switch off path (refer to “The redundant switch off path”);

assign corresponding error code to ERR_COD_MC (see “Fault reaction of processor monitoring”)

ENDIF

E.30 Processor Monitoring (Appl. Inc.)

E.30.1 Communication between MC and MU

These data affects the communication itself as well as the monotoring of the communication on MC and MU.

Configuration data: (value assignment)

Name	Hex. value	Phys. value	Resol.	Unit
NC_T_SPI_COM	0Ah	10	1	ms
time for monitoring communication				
NC_T_SPI_MAX	0Dh	13	1	ms
Maximum time for timeout monitoring on MU				
NC_T_SPI_MIN	07h	7	1	ms
Minimum time for timeout monitoring on MU				
NC_ABC_INC_HD_MC	03h	3	1	1
increment value of anti bounce counter for maximum of header/command on MC				
NC_ABC_MAX_HD_MC	0Ch	12	1	1
maximum value of anti bounce counter for maximum of header/command on MC				
NC_ABC_INC_CKS_MC	03h	3	1	1
Increment value of anti bounce counter for monitoring of checksum on MC				
NC_ABC_MAX_CKS_MC	0Ch	12	1	1
Maximum value of anti bounce counter for monitoring of checksum on MC				
NC_ABC_INC_TOUT_MAX_MU	03h	3	1	1
Increment value of anti bounce counter for maximum timeout monitoring on MU				
NC_ABC_MAX_TOUT_MAX_MU	0Fh	15	1	1
Maximum value of anti bounce counter for maximum timeout monitoring on MU				
NC_ABC_INC_TOUT_MIN_MU	03h	3	1	1
Increment value of anti bounce counter for minimum timeout monitoring on MU				
NC_ABC_MAX_TOUT_MIN_MU	0Fh	15	1	1
Maximum value of anti bounce counter for minimum timeout monitoring on MU				
NC_ABC_INC_HD_MU	03h	3	1	1
Increment value of anti bounce counter for monitoring of header/command on MU				
NC_ABC_MAX_HD_MU	0Ch	12	1	1
Maximum value of anti bounce counter for monitoring of header/command on MU				
NC_ABC_INC_CKS_MU	03h	3	1	1
Increment value of anti bounce counter for monitoring of checksum on MU				
NC_ABC_MAX_CKS_MU	0Ch	12	1	1
Maximum value of anti bounce counter for monitoring of checksum on MU				

E.30.1.1 Initialisation of the Port Extensions

During initialisation (see “Processor Monitoring”) the digital output port extensions (see “Digital output port extensions”) are initialized (see “Communication between MC and MU”). NC_ENA_PORT_MU has the same structure like OUT_PORT_MU, which is described in chapter “Digital output port extensions”.

Configuration data: (value assignment)

Name	Hex. value	Binary value	Resol.	Unit
NC_ENA_PORT_MU	F9h	1111 1001	1	1
Bit pattern on MC, set to MU for initializing the digital output port extension				

E.30.2 Analog and pseudo digital inputs

E.30.2.1 Analog inputs on MU

Configuration data: (value assignment)

Name	Hex. limits	Phys. limits	Resol.	Unit
NC_ACQ_MU_NR	08h	8	1	1
Dimension of the array ACQ_MU				

E.30.3 Digital output port extension

E.30.3.1 Generation of Diagnosis Pulses on the Digital Output Port Extension

Configuration data: (value assignment)

Name	Hex. limits	Phys. limits	Resol.	Unit
NC_T_DGO_OFS	02h	2	1	ms
Time to wait after communication to send diagnosis pulse				

E.30.4 Processor monitoring

E.30.4.1 The test calcualtion Copy of process monitoring

Configuration data: (value assignment)

Name	Hex. value	Phys. value	Resol.	Unit
NC_ABC_INC_MON2_MU	03h	3	1	1
Increment value of anti bounce counter of level 2’ test calculations on MU				
NC_ABC_MAX_MON2_MU	0Ch	12	1	1
Maximum value of anti bounce counter of level 2’ test calculations on MU				
NC_TEST_REC_IDX_MAX_MON2	0Ch	12	1	1
(NC_TEST_REC_IDX_MAX_MON2-1) is the maximum value for the index of test calculation Therfore NC_TEST_REC_IDX_MAX_MON2 answers are calculated cyclically				

E.30.4.1.1 Inspection of MU by the MC according to test calculation level 2’

Configuration data: (value assignment)

Name	Hex. value	Phys. value	Resol.	Unit
NC_ABC_INC_MON2_TEST_MC	03h	3	1	1
Increment value of anti-bounce counter to test the anti-bounce counter of level 2’ test calculations on MU				
NC_ABC_MAX_MON2_TEST_MC	0Ch	12	1	1
Maximum value of anti-bounce counter to test the anti-bounce counter of level 2' test calculations on MU				
NC_ABC_INC_MON2_TEST_2_MC	03h	3	1	1
Increment value of anti-bounce counter to test the incrementation of level 2’ test calculation index on MU				
NC_ABC_MAX_MON2_TEST_2_MC	0Ch	12	1	1
Maximum value of anti-bounce counter to test the incrementation of level 2' test calculation index on MU				
NC_T_MON2_TEST_MC	0Fh	320	2*NC_T_SPI_COM	1
Time base for inspection the evaluation of level 2' test calculation on MU				

E.30.4.2 Program flow monitoring

The bits of RESP_PFM_MC and SYN_PFM_MU as well as the anti bounce counters of the program flow monitoring has to be assigned to specific tasks running on different time levels:

Configuration data: (value assignment)

Name	Hex. value	Phys. Value	Resol.	Unit
NC_ABC_INC_PFM_TEST_MC	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MC				
NC_ABC_MAX_PFM_TEST_MC	0Ch	12	1	1
Increment value of anti-bounce counter of program flow monitoring on MC				
NC_ABC_INC_PFM_1_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 1				
NC_ABC_MAX_PFM_1_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 1				
NC_ABC_INI_PFM_1_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 1				
NC_T_PFM_1_MU	01h	10	NC_T_SPI_COM	1
Time grid of task 1				
NC_T_CHK_PFM_1_MU	02h	20	NC_T_SPI_COM	1
Time to check PFM bits of task 1				
NC_ABC_INC_PFM_2_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 2				
NC_ABC_MAX_PFM_2_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 2				
NC_ABC_INI_PFM_2_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 2				
NC_T_PFM_2_MU	04h	40	NC_T_SPI_COM	1
Time grid of task 2				
NC_T_CHK_PFM_2_MU	08h	80	NC_T_SPI_COM	1
Time to check PFM bits of task 2				
NC_ABC_INC_PFM_3_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 3				
NC_ABC_MAX_PFM_3_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 3				
NC_ABC_INI_PFM_3_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 3				
NC_T_PFM_3_MU	04h	40	NC_T_SPI_COM	1
Time grid of task 3				
NC_T_CHK_PFM_3_MU	08h	80	NC_T_SPI_COM	1
Time to check PFM bits of task 3				
NC_ABC_INC_PFM_4_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 4				
NC_ABC_MAX_PFM_4_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 4				
NC_ABC_INI_PFM_4_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 4				
NC_T_PFM_4_MU	28h	40	1	ms
Time grid of task 4				
NC_T_CHK_PFM_4_MU	08h	80	NC_T_SPI_COM	1
Time to check PFM bits of task 4				
NC_ABC_INC_PFM_5_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 5				
NC_ABC_MAX_PFM_5_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 5				
NC_ABC_INI_PFM_5_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 5				

Name	Hex. value	Phys. Value	Resol.	Unit
NC_T_PFM_5_MU	04h	40	NC_T_SPI_COM	1
Time grid of task 5				
NC_T_CHK_PFM_5_MU	08h	80	NC_T_SPI_COM	1
Time to check PFM bits of task 5				
NC_ABC_INC_PFM_6_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 6				
NC_ABC_MAX_PFM_6_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 6				
NC_ABC_INI_PFM_6_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 6				
NC_T_PFM_6_MU	04h	40	NC_T_SPI_COM	1
Time grid of task 6				
NC_T_CHK_PFM_6_MU	08h	80	NC_T_SPI_COM	1
Time to check PFM bits of task 6				
NC_ABC_INC_PFM_7_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 7				
NC_ABC_MAX_PFM_7_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 7				
NC_ABC_INI_PFM_7_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 7				
NC_T_PFM_7_MU	0Ah	100	NC_T_SPI_COM	1
Time grid of task 7				
NC_T_CHK_PFM_7_MU	14h	200	NC_T_SPI_COM	1
Time to check PFM bits of task 7				
NC_ABC_INC_PFM_8_MU	03h	3	1	1
Increment value of anti-bounce counter of program flow monitoring on MU for task 8				
NC_ABC_MAX_PFM_8_MU	0Ch	12	1	1
Maximum value of anti-bounce counter of program flow monitoring on MU for task 8				
NC_ABC_INI_PFM_8_MU	0Ch	12	1	1
Init value of anti-bounce counter of program flow monitoring on MU for task 8				
NC_T_PFM_8_MU	64h	1000	NC_T_SPI_COM	1
Time grid of task 8				
NC_T_CHK_PFM_8_MU	C8h	2000	NC_T_SPI_COM	1
Time to check PFM bits of task 8				

Note: If the task *n* is not program flow monitored, the configuration data NC_ABC_INC_PFM_*n*_MU and NC_ABC_INI_PFM_*n*_MU must be set to zero.

E.30.5 Fault Reaction of Processor Monitoring

Configuration data: (value assignment)

Name	Hex. value	Phys. value	Resol.	Unit
NC_RST_CTR_MAX_MC	07h	7	1	1
Maximum value of software resets on MC				
NC_RST_CTR_MAX_MU	07h	7	1	1
Maximum value of software resets on MU				
NC_T_RST_CTR_MC	FFh	255	NC_T_SPI_COM	1
Time to decrement the reset counter on MC				
NC_T_RST_CTR_MU	18h	6144	256	ms
Time to decrement the reset counter on MU				
NC_T_DR_DI_RST_MC	34h	520	NC_T_SPI_COM	1
Time delay for enabling power stages after error reset on MC				
NC_T_DR_DI_RST_MU	34h	520	NC_T_SPI_COM	1
time delay for enabling power stages after error reset on MU				

E.30.6 Classification of Processor Faults

The fault reaction time of all safety critical faults must be shorter than the required fault reaction time of Process Monitoring, which is 500ms (see "General"; chapter E).

Safety critical Processor Monitoring faults are:

- Standard RAM test on MC
- Cyclical ROM test of Process Monitoring
- Program flow monitoring of Process Monitoring
- The test calculation Copy of process monitoring
  All test calculations and the debouncing of a fault must be performed within the fault reaction time

Additional the following tests are performed within 500ms:

- RAM test on MC for process monitoring.
  This test is also safety critical. Note, that the fault reaction time of this test is implicitly given by the scheduler of level2. The complete RAM test of Process Monitoring is performed within the recurrence time of the Process Monitoring modules.
- Watchdog on MU
  This test is a redundant test to the PFM.
- Inspection of MU by the MC according to test calculation (level 2')
  The aim of this test is the detection of dormant faults. One has to avoid, that a level 2 module get defect *and* the dedicated level 2' module is no longer tested because of a defect in the generation of the level 2' test index.
- Inspection of MU by the MC on PFM
  The aim of this test is the detection of a dormant fault. With this test it is ensured, that the evaluation of PFM data in level 3 are able to detect a PFM error.
- Communication monitoring (communication between MC and MU)
  This tests ensures a correct communication and data transfer between MC and MU. This tests are performed permanently, if the communication is active.

The following additional test are performed during the initialization only:

- Predrive Check
  The aim of this test is the detection of dormant fault
- Software reference check
- Pin configuration test

The following tests are not seen as safety critical:

- Standard ROM test on MC
  The aim of this test is the detection of a dormant fault. Note, that the ROM area of the Process Monitoring is teseted within the required fault reaction time. Therefore it is ensured, that the performance of the Process Monitoring is not affected by any ROM error.

E.31 Interface to power stages

See the following figure for the HW-schematic:

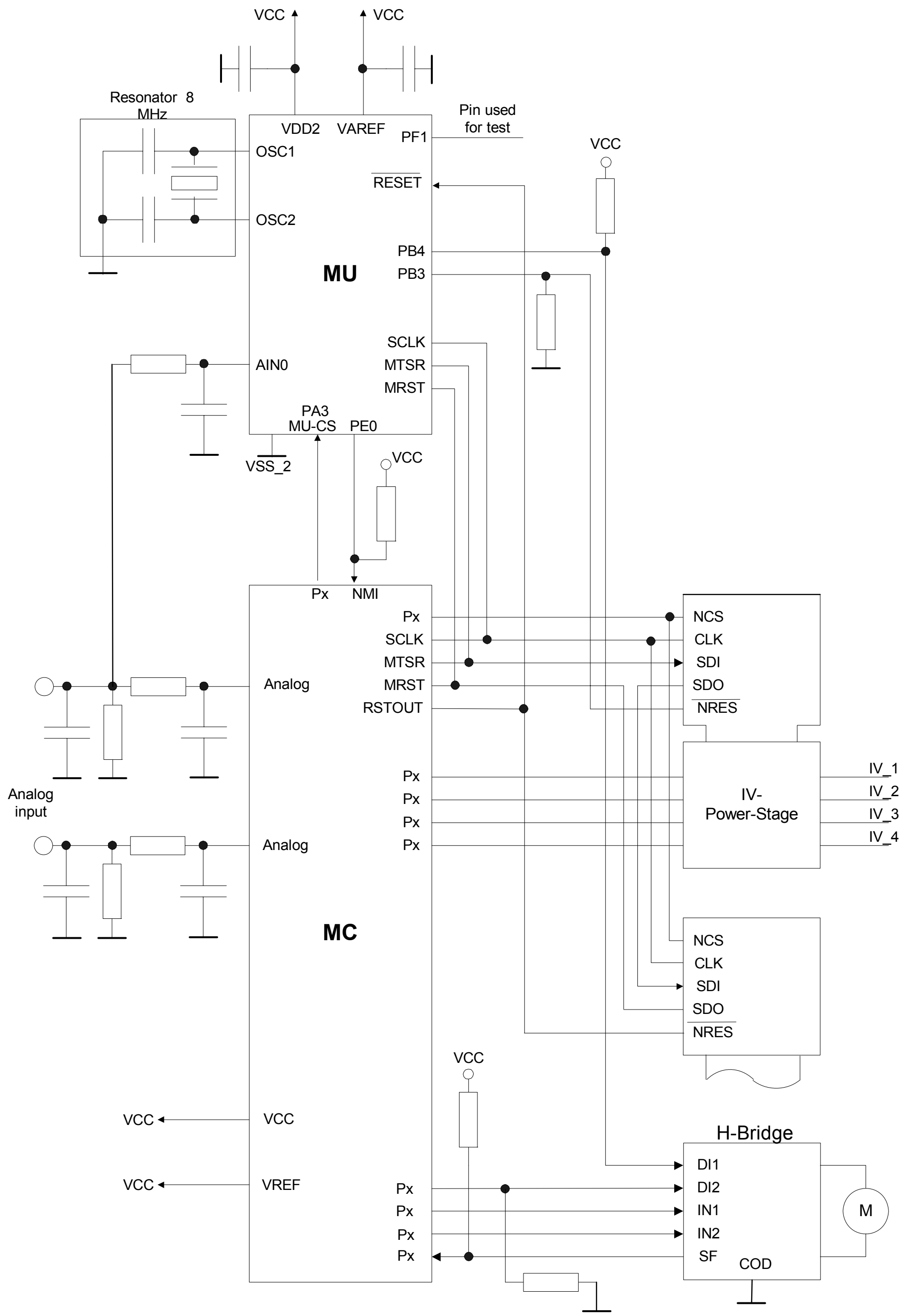


Figure 76: Hardware Safety Module

The main-processor (MC) and the monitoring processor (MU, incl. port extensions) are connected via SPI-communication. MU has its own oscillator (ceramic resonator).

When resetting, MC gives a reset signal to MU so it will also reset. This enables synchronisation between them.

In case of processor monitoring faults detected on MU, the throttle is disabled via PB4 and injectors are disabled via PB3. Via PB0, MU resets MC to increase availability.

The status when the powerstages are disabled via MU are:

PE0 = HIGH

PE1 = LOW (only for 32bit main controller applications)

PB3 = LOW

PB4 = HIGH

LV_ERR_TMP_MU = 1;

The analog input of first AD-channel on MU is connected to the dedicated analog input on the MC; the value is transmitted to MC only to check the ADCs. Remaining AD-channels and digital outputs can be used as port extension (no safety function, but it has to be assured, that no damaging voltage is fed to the ports in case of any fault).

E.32 Fault reaction of process monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_N_LIM_REQ_MON	O/V	0...1H	0...1	1	-
Request for engine speed limitation					
LV_OFF_IV_MON	O/V	0...1H	0...1	1	-
Request for disable of IV power stage by main controller					
LV_OFF_MTC_MON	O/V	0...1H	0...1	1	-
Request for disable of MTC power stage by main controller					
LV_ERR_TRAN_MON	V	0...1H	0...1	1	-
Flag for temporary fault during last recurrency					
LV_OFF_IV_N_LIM_ETC_TMP_MON	V	0...1H	0...1	1	-
Flag for temporary injection cut off during last recurrency					

Input data:

LV_OFF_IV_N_LIM_ETC_MON	LV_ERR_CONV_MON	LV_ERR_N_32_MON	LV_ERR_TQ_DIF_P_D_IS_MON
LV_ERR_TQ_DIF_I_IS_MON	LV_ERR_TQ_LOSS_MON	LV_ERR_TQ_MIN_CLU_MON	LV_ERR_TQ_MAX_CLU_MON
LV_ERR_PVS_MON	LV_ERR_TPS_MON	LV_ERR_MAF_MON	LV_ERR_TQI_AV_MON
LV_ERR_TQI_N_MAX_MON	LV_ERR_MU_MC	LV_ERR_TMP_MU_MC	

Import actions:

ACTION_ECM3_Service8TaskPfm(4)
ACTION_ECM3_Service9TaskPfm(4)

Note: This actions are defined in chapter "Processor Monitoring", subsection "Program Flow Monitoring services for Process Monitoring". The first action has to be executed as first instruction in this module, the second action has to be executed as last instruction of this module. The argument has the value 4.

ACTION_ECM3_WriteChkCpl(IN <type>, INOUT <xyz_MON>, OUT <xyz_MON_CPL>, IN <reg>)
ACTION_ECM3_ChkCpl(IN <type>, IN <xyz_MON>, IN <xyz_MON_CPL>)
ACTION_ECM3_ReadChkCpl(IN <type>, OUT <reg>, IN <xyz_MON>, IN <xyz_MON_CPL>)

Note: This actions are defined in chapter "Processor Monitoring", subsection "RAM test on MC for processor monitoring". All variables with Mode “O” and history variables in modules of the *Process Monitoring* have to be checked by the cyclical RAM test.

ACTION_ECM3_RedSwitchOffPath()

Note: This action is defined in chapter "Processor Monitoring", subsection "The redundant switch off path".

Export Actions:

ACTION_ECM2_LockPws()
Disable power stages by process monitoring (used when no ECM2 tasks executed)
ACTION_ECM2_UnlockPws()
Enable power stages by process monitoring for exactly one time (used when ECM2 tasks executed)

Description for ACTION_ECM2_LockPws():

Description:

Syntax: ACTION_ECM2_LockPws()

Formula section:

LV_OFF_IV_MON = 1;

LV_OFF_MTC_MON = 1;

/* Disable injection and H-bridge */

Description for ACTION_ECM2_UnlockPws():

Description:

Syntax: ACTION_ECM2_UnlockPws()

Formula section:

LV_OFF_IV_MON = 0;

LV_OFF_MTC_MON = 0;

/* Enable injection and H-bridge for exactly one time */

General description

The objective of this module is to react on faults detected by the process and processor monitoring of the etc-monitoring-concept. The following actions are introduced depending on the error level and the degree of degradation.

disable of throttle power stage (LV_OFF_MTC_MON)

disable of injection power stage (LV_OFF_IV_MON)

transmission of wrong answers from level 2' to the monitoring processor in order to force the monitoring unit to reset the main controller and switch of the throttle and the injection power

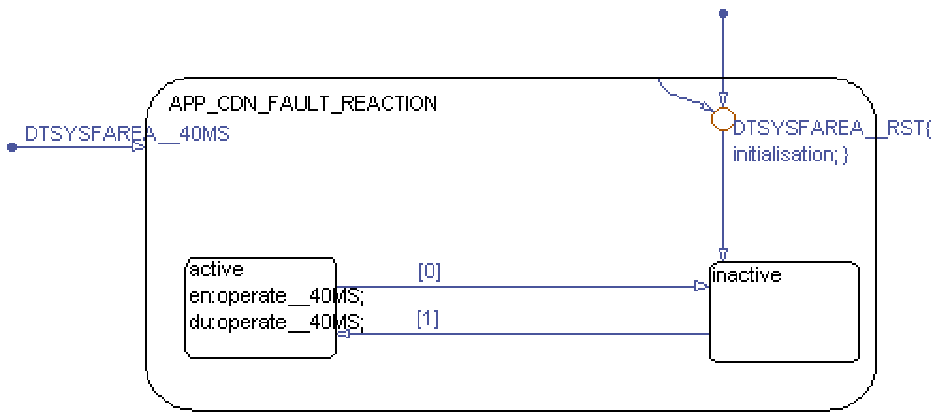
The following table shows the fault reactions corresponding to the underlying faults. No fault reaction is reset until next ignition key on (transition from LV_IGK = 0 to 1). There are only two exceptions, 1.) for a temporary fault detected by the processor monitoring on the main controller (LV_ERR_TMP_MU_MC) and 2.) for injection cut off (LV_OFF_IV_MON) during engine speed limitation.

Faults \ Fault reaction	Disable of MTC via main controller (LV_OFF_MTC_MON = 1)	Request engine speed limitation (LV_N_LIM_REQ_MON = 1)	Disable of IV via main controller (LV_OFF_IV_MON = 1)	Disable of MTC and IV via monitoring unit (ACTION_ECM3_RedSwitchOffPath())
LV_ERR_CONV_MON	X	X	Z	Z
LV_ERR_N_32_MON	X	X	Z	Z
LV_ERR_TQ_DIF_P_D_IS_MON	X	X	Z	Z
LV_ERR_TQ_DIF_I_IS_MON	X	X	Z	Z
LV_ERR_TQ_LOSS_MON	X	X	Z	Z
LV_ERR_TQ_MIN_CLU_MON	X	X	Z	Z
LV_ERR_TQ_MAX_CLU_MON	X	X	Z	Z
LV_ERR_PVS_MON	X	X	Z	Z
LV_ERR_TPS_MON	X	X	Z	Z
LV_ERR_MAF_MON	X	X	Z	Z
LV_ERR_TQI_AV_MON	X	X	Z	Z
LV_OFF_IV_N_LIM_ETC_MON*	Z	Z	X / 0	Z
LV_ERR_TQI_N_MAX_MON	X	Z	X	X
LV_ERR_MU_MC	X	Z	X	Z
LV_ERR_TMP_MU_MC*	X	Z	X	Z

* reversible during a driving cycle,

The map shows resulting states of output bits. The proceeding is top-down, e.g. the undermost set fault bit of a column defines the resulting state. Symbol “x” means the related bit is set, “z” the former result of present recurrency will not be changed and “0 “ resets the related bit, if the fault is not set.

LV_OFF_IV_MON must be calculated first by LV_OFF_IV_N_LIM_ETC_MON and than by non reversible fault reactions to be sure that a wrong write operation with LV_OFF_IV_MON = 0 is less prior than fault reactions with permanent fuel cut-off.



Application Condition

Note: DTSYSFAREA__RST includes the function calls as defined in application incidences.

Function Description

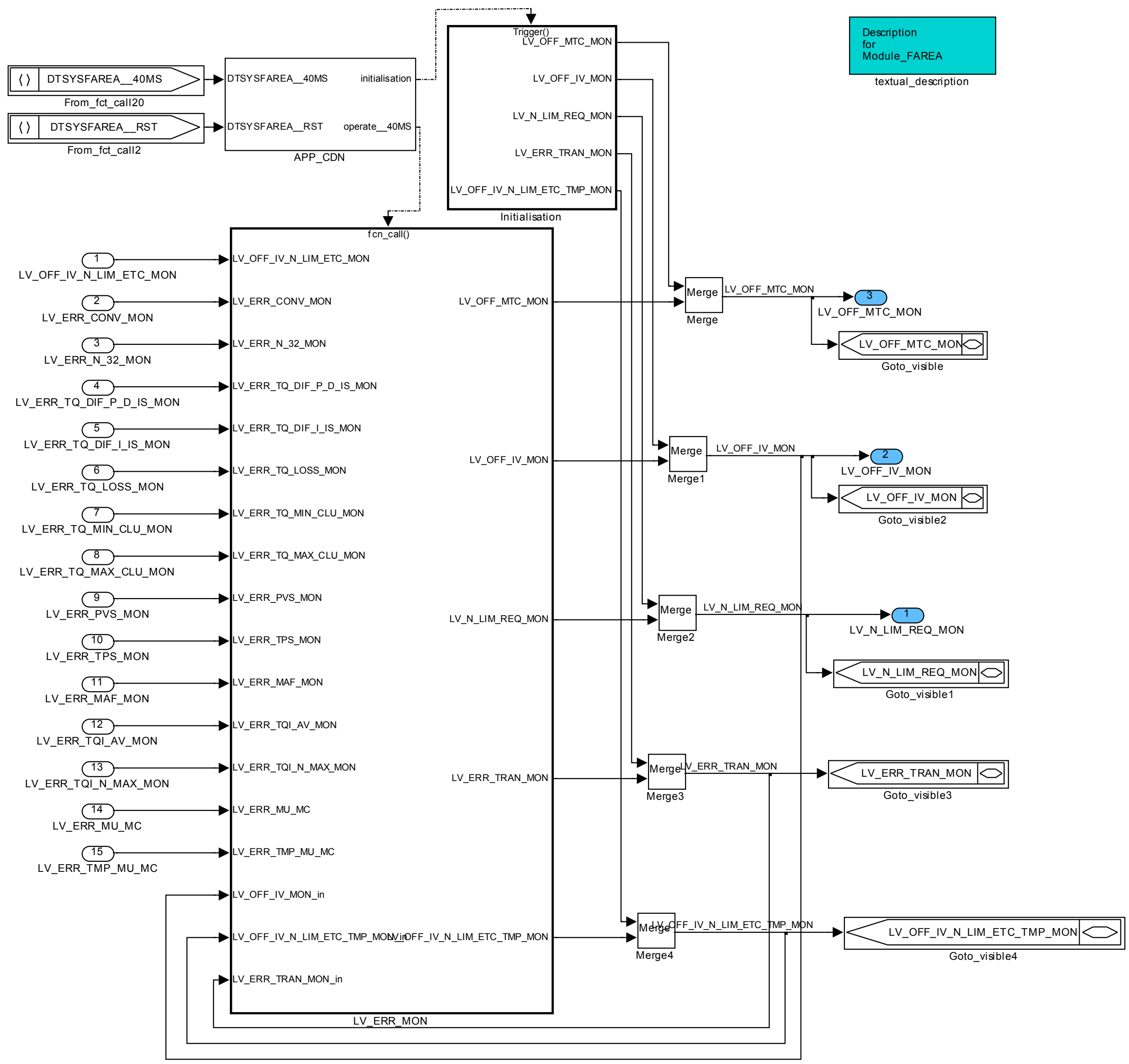


Figure 77 FAREA

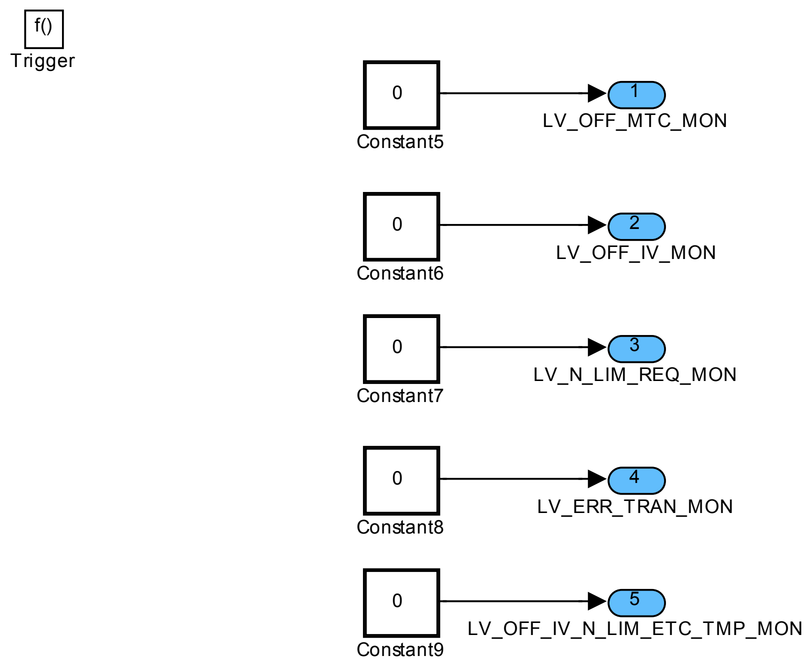


Figure 78 FAREA/ Initialisation

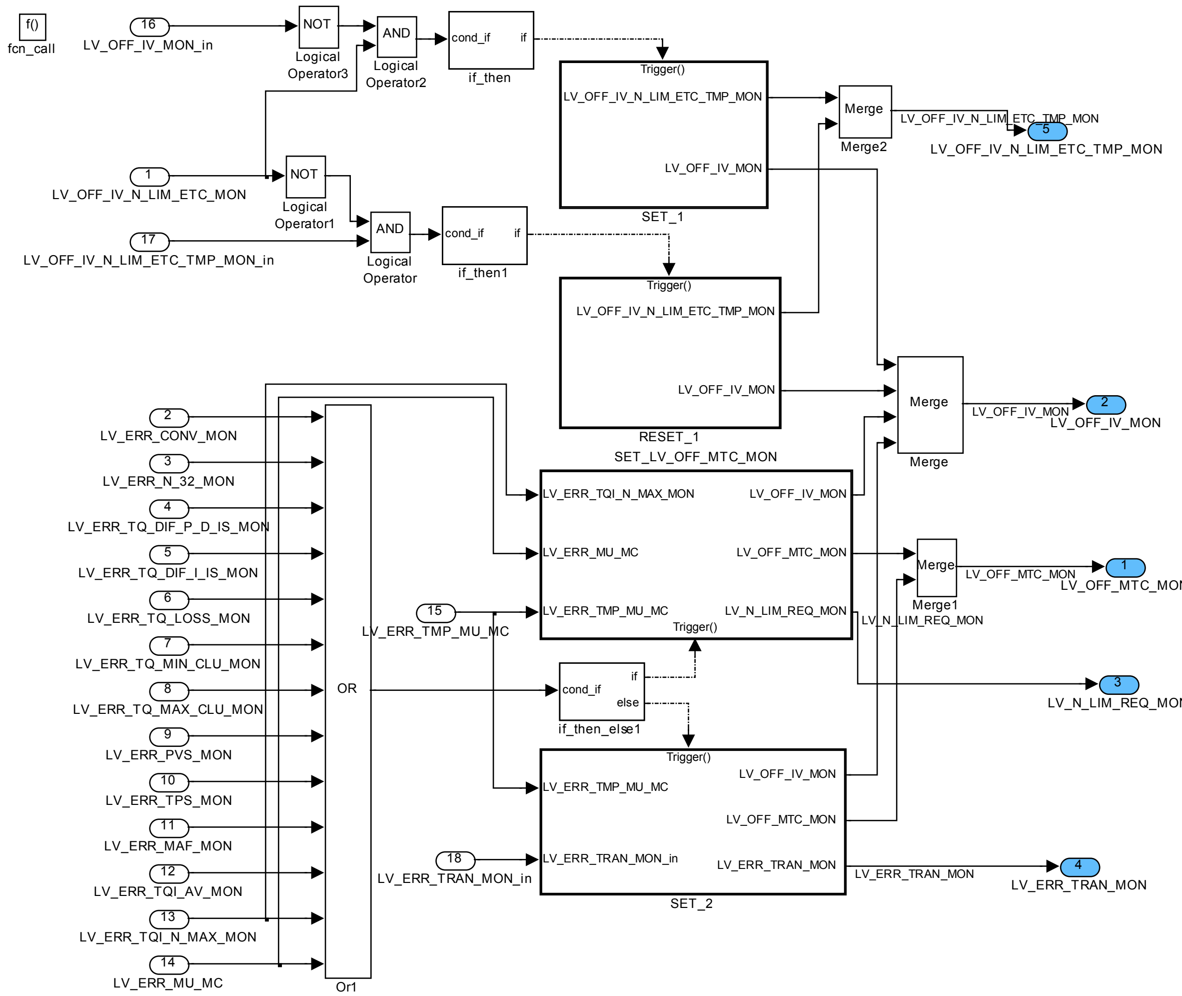


Figure 79 FAREA/ LV_ERR_MON

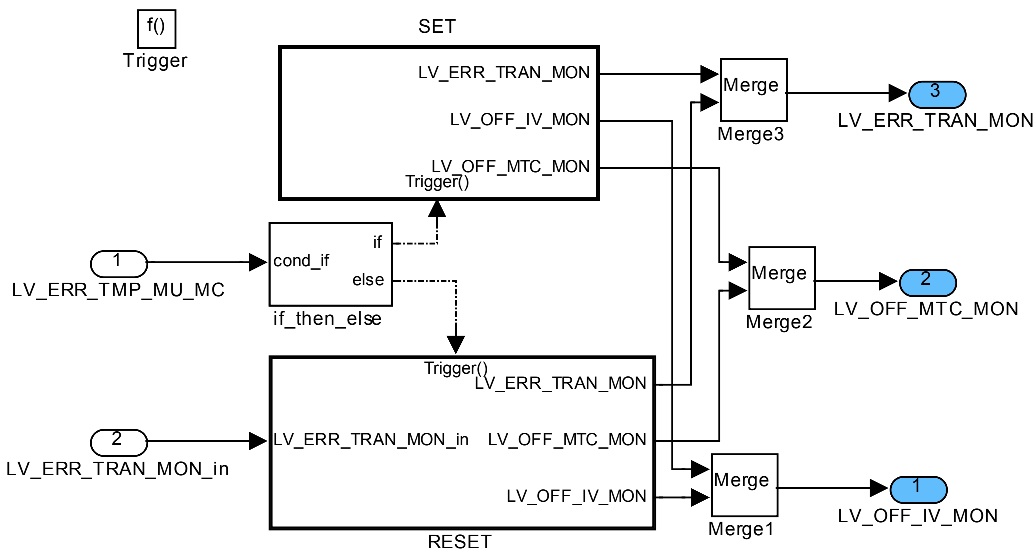


Figure 4 FAREA/ LV_ERR_MON/ SET_2

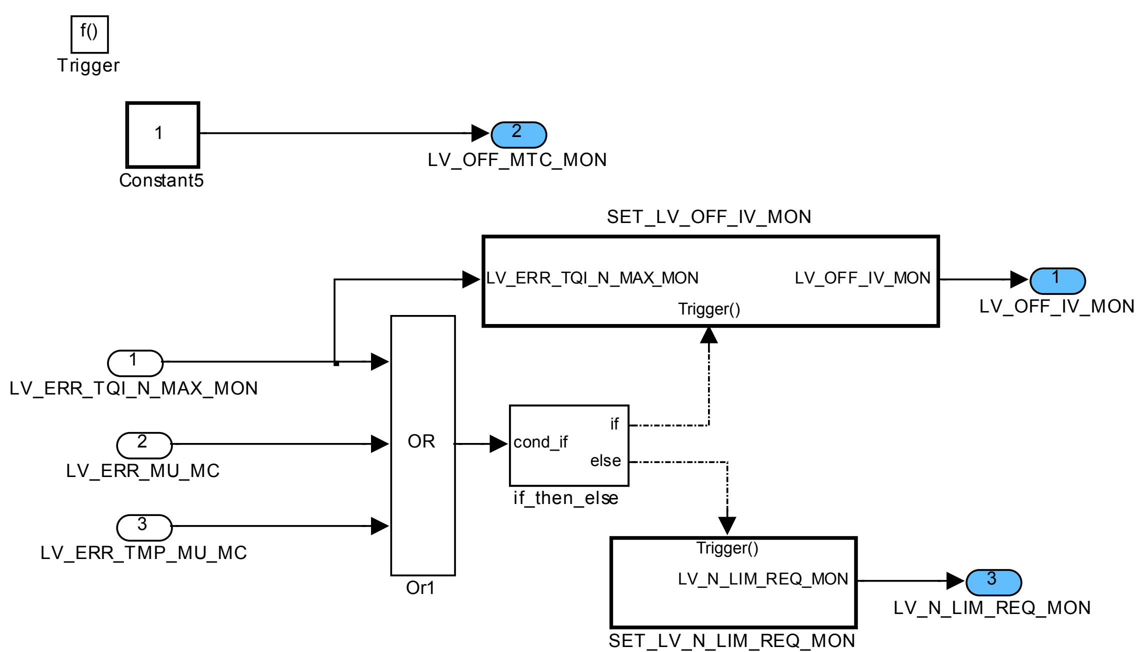


Figure 5 FAREA/ LV_ERR_MON/ SET_LV_OFF_MTC_MON

E.33 Fault Reaction of Processor Monitoring

If the MC or the MU detect a processor fault, the safety critical power stages are disabled and a SW-reset is done. The disabling of the power stages is handled in different ways: The MU sets the level of the disable pins (see "Interface to power stages") and therefore directly disables the power stages, while the MC only sets some boolean variables, which are evaluated elsewhere, i.e. *Fault Reaction of Process Monitoring*.

The release of a SW-reset has to be implementend in the level 3 (*Processor Monitoring*) itself in a very robust way; it may not occure at all that the SW-reset is not carried out because of any dormant fault.

The subroutine ‘central fault reaction’ is called with parmeters which enables to know exactly where what kind of fault has been detected. The number of resets is limited on the MC and the MU separately. After this limited number of resets MC and MU are permanently disabled with disabled power stages for this drive cycle.

During the initialisation (see “Processor Monitoring”) both, MU and MC, keeps the power stages disabled if one of the reset counters (RST_CTR_MC, RST_CTR_MU) is bigger than one. Therefore the reset counters are transferred during the INIT communication, see “Communication between MC and MU”.

Both, MU and MC are able to request a reset. If the MU requests a reset, this is transferred to MC via NMI. Reseting the MC will automatically entail a reset on the MU.

It must be differentiated therefore whether a fault reaction is initiated on the MC or on the MU.

To enforce the ECU to enter a safe state, the safety critical power stage are disabled. There are two separate and independed paths to do this via MC and/or via MU.

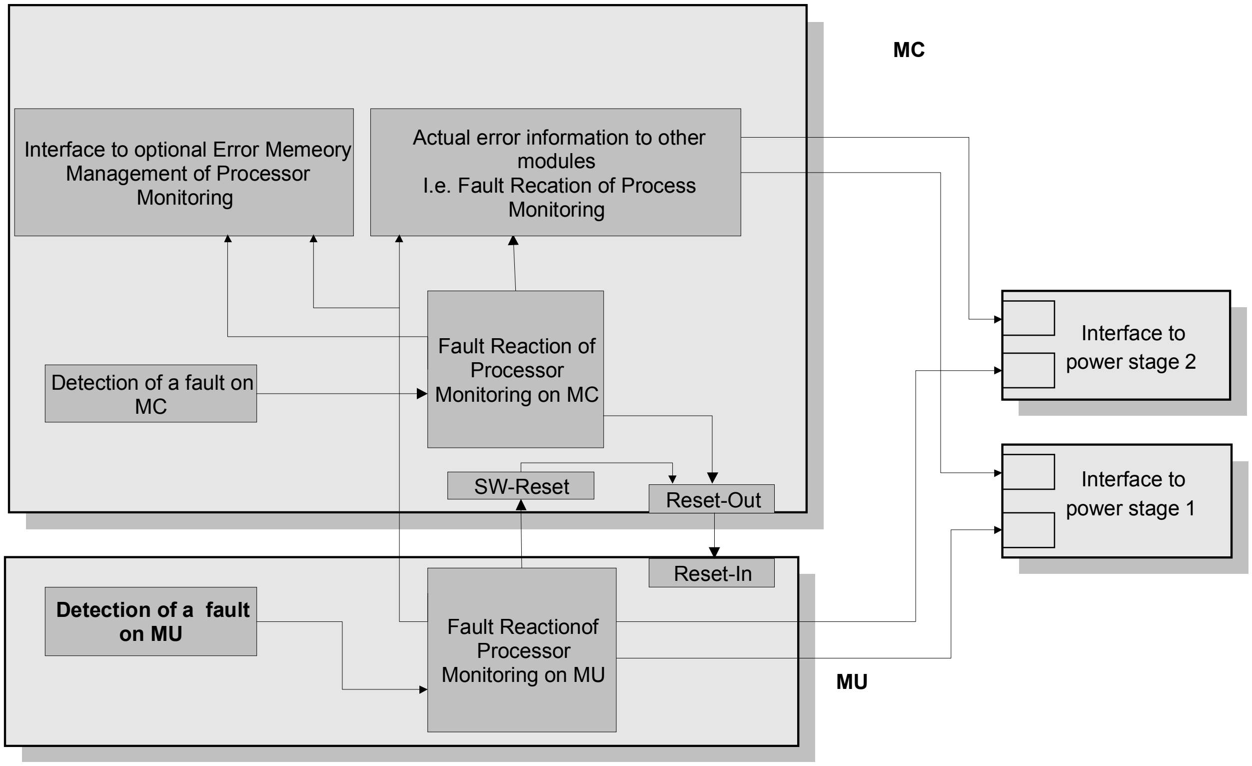


Figure 80: Fault Reaction paths of the *Processor Monitoring* with two power stages

E.33.1 Link between fault variables and initialization of MU fault variables on MC

FUNCTION DESCRIPTION:

General information:

In the first communication the values of the MU fault variables RST_CTR_MU and ERR_COD_MU are transferred from MU to MC, see "Communication between MC and MU". LV_ERR_TMP_MU on MC can also be updated only *after* a dedicated communication. Therefore it is mandatory to initialize this variables on MC *before* this communications, because the values set by MU are only avaliable *after* a fault free communication. The first communication could be delayed i.e. by the Predrive Check. Because other software moduls could evaluate the variables inbetween, a useful initialization is needed.

Application conditions:

Initialisation: LV_ERR_TMP_MU = 1; /* MU disables the power stages temporay */

RST_CTR_MU = 0;

ERR_COD_MU = 0h;

E.33.2 Fault Reaction of MC

Output data:

Name	Mode	Hex. limits	Phys. Limits	Resol.	Unit
LV_ERR_MC	V/O	0h ... 1h	0...1	1	1
Final request for disabled power stages by processor monitoring on MC					
LV_ERR_TMP_MC	V/O	0h ... 1h	0...1	1	1
Temporary request for disabled power stages by processor monitoring on MC					
ERR_COD_MC	V/O/S	00h ... FFh	0...255	1	1
Error code , which represents a failure detected by MC					
RST_CTR_MC	V/O/S	00h ... FFh	0...255	1	1
Reset counter on MC					

Note: ERR_COD_MC and RST_CTR_MC have to be located in the software reset safe RAM area, as indicated by the mode “S”.

Input data:

SW_COMP_NR_MC	SW_VERS_MU	ABC_MON2_MU
ABC_PFM_TEST_MC	ABC_MON2_TEST_2_MC	ABC_CKS_MC
ABC_HD_MC	LV_ROM_CRC_READY	RST_CTR_MU

FUNCTION DESCRIPTION:

General information:

The MC has different fault locations:

ERR_COD_MC

Value	Meaning	Reason
0h	No error	
1h	Communication, checksum error	ABC_CKS_MC
2h	Communication, MU sends wrong command/header	ABC_HD_MC
3h	Predrive check error	MU is unable to disable the safety relevant power stages
4h	Software version, MU sends wrong SW version	SW_COMP_NR_MC, SW_VERS_MU
5h	Software version, MU sends wrong variant coding	Pin-Configuration
6h	Level 2’, MU does not increment ABC_MON2_MU	ABC_MON2_TEST_MU
7h	Level 2’, MU does not increment TEST_REC_IDX_MON2	ABC_MON2_TEST_2_MC
8h	Program flow monitoring, MU does not toggle the control bit(s)	ABC_PFM_TEST_MC
9h	ROM test of process monitoring (Level 2)	
Ah	RAM test of process monitoring (Level 2)	
Bh	Standard total ROM test	
Ch	Standard total RAM test	
Dh	Switch off path activation due to endless resets on MC	

After a power up ERR_COD_MC is initialized with zero.

Application conditions:

Initialisation: ERR_COD_MC = 0
after power up only

RST_CTR_MC = 0
after power up only

LV_ERR_TMP_MC = 1
after reset or power up

LV_ERR_MC = 0 after power up
LV_ERR_MC = 0 after reset if RST_CTR_MC <= NC_RST_CTR_MAX_MC
LV_ERR_MC = 1 after reset if RST_CTR_MC > NC_RST_CTR_MAX_MC

E.33.2.1 Setting of error flag on MC

If a fault is debounced, a central fault reaction carried out and the number of reset requests is counted. Before the first communication the reset safe stored reset counter RST_CTR_MC is evalauted to set the variable LV_ERR_TMP_MC:

Formula section:

```
/* In case of an error detected on MC: */
IF (RST_CTR_MC < NC_RST_CTR_MAX_MC) THEN
    RST_CTR_MC++;
    RESET on MC;
ELSE
    LV_ERR_MC = 1;
    RST_CTR_MC = NC_RST_CTR_MAX_MC + 1;
    /* The Predrive check has to be cancelled */
    enter DISABLE State

END


/* during initialization (after reset caused by MC or triggered by MU via NMI)   */
/* this block is executed every recurrence of a level 3 call in the             */
/* time periode of NC_T_DR_DI_RST_MC after reset or power up                    */
/* After this time periode LV_ERR_TMP_MC is reset                               */
IF (RST_CTR_MC > 1) OR (RST_CTR_MU > 1) THEN
    LV_ERR_TMP_MC = 1;
ELSE
    LV_ERR_TMP_MC = 0;
END

IF (RST_CTR_MC > NC_RST_CTR_MAX_MC) THEN
    LV_ERR_MC = 1;
    /* The Predrive check has to be cancelled */
    enter DISABLE State

END
```

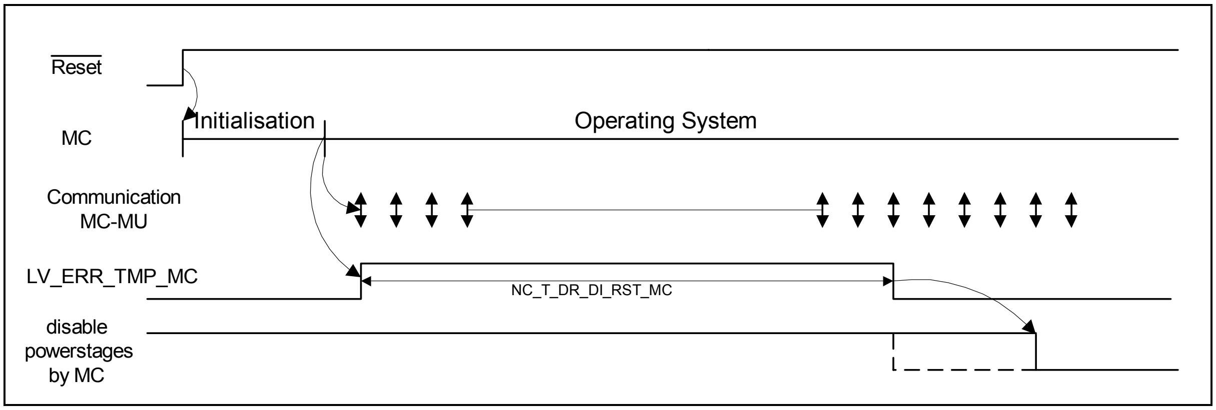
RST_CTR_MC is decremented if the time NC_T_RST_CTR_MC has passed *and* LV_ROM_CRC_READY is set (Standard ROM test finished, see "Processor Monitoring", subsection "Standard ROM Test on MC").

Note: All tests on MC at runtime must not last longer than NC_T_RST_CTR_MC. An exeption is the standard ROM test on MC.

E.33.2.2 Fault Reaction after second to (NC_RST_CTR_MAX_MC)th fault reset

(fault does not longer exist)

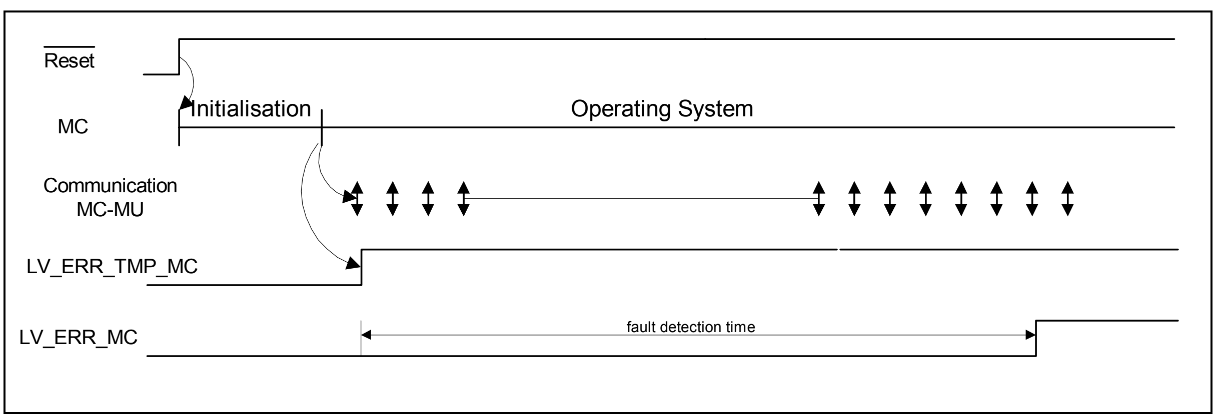
After the second error reset (RST_CTR_MC>1 or RST_CTR_MU>1) LV_ERR_TMP_MC is set and the MC waits NC_T_DR_DI_RST_MC before it enables the power stages (LV_ERR_TMP_MC = 0). Within the required fault reaction time all Processor Monitoring related tests have to be performed. During this time periode the safety relevant power stages must be disabled. The safety critical faults are listed in the subsection "Classification of Processor Faults". Processor Monitoring Faults during this time periode would lead to another reset with disabled power stages.



E.33.2.3 Fault Reaction after (NC_RST_CTR_MAX_MC)th fault reset

(fault still exists)

After the (NC_RST_CTR_MAX_MC)th fault reset on MC a complete test of MU and MC is performed. Faults during that test would lead to permanently disabled power stages for this driving cycle. The Predrive check must be cancelled, if active (see "Processor Monitoring", subsection "Reset or power up").



Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NC_T_RST_CTR_MC	1	00h ... FFFFh	0...65535*NC_T_SPI_COM	NC_T_SPI_COM	1
time to decrement the reset counter					
NC_T_DR_DI_RST_MC	1	00h ... FFh	0...255*NC_T_SPI_COM	NC_T_SPI_COM	1
Time delay for enabling power stages after error					
NC_RST_CTR_MAX_MC	1	00h ... FFh	0...255	1	1
max value of software resets on MC					

E.33.3 Fault Reaction of MU

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ERR_COD_MU	V/O	00h ... FFh	0...255	1	1
Error code which represents an error on MC detected by MU					
RST_CTR_MU	V/O	00h ... FFh	0...255	1	1
reset counter on MU, transmitted to MC after reset					
LV_ERR_TMP_MU	V/O	0 ... 1h	0 ... 1	1	1
Request of MU to disable the power stages temporary, sent to MC					

Note: The values of these variables are available on MC after the dedicated internal communication. The MU internal variables ERR_COD_MU and RST_CTR_MU have to be located in the software reset safe RAM area.

Input data:

RAM_TEST_MU	RAM_TEST_MU_CPL	CRC_CKS_CLC_MU	NC_CRC_CKS_MU
ABC_MON2_MU	SW_COMP_NR_MC	NC_SW_VERS_MU	RST_CTR_MC
ABC_TOUT_MAX_MU	ABC_TOUT_MIN_MU	ABC_HD_MU	ABC_CKS_MU
ABC_PFM_1_MU	ABC_PFM_2_MU	ABC_PFM_3_MU	ABC_PFM_4_MU
ABC_PFM_5_MU	ABC_PFM_6_MU	ABC_PFM_7_MU	ABC_PFM_8_MU

FUNCTION DESCRIPTION:

Note:

The MU internal variables are not visible at all. Their values are available on MC after the dedicated internal communication only if the values are transferred.

General information:

The MU has different failure locations:

ERR_COD_MU

Value	Meaning	Reason
00h	no error	
01h	Wrong header/command in communication	ABC_HD_MU
02h	Checksum error in communication	ABC_CKS_MU
03h	Wrong pin configuration	
04h	MU-SW does not match MC-SW	SW_COMP_NR
05h	Maximum timeout communication error	ABC_TOUT_MAX_MU
06h	Minimum timeout communication error	ABC_TOUT_MIN_MU
07h	RAM error	RAM_TEST_MU, RAM_TEST_MU_CPL
08h	ROM error	(CRC_CKS_CLC<>NC_CRC_CKS)
09h	Wrong result of testcalculation level2' (redundant switch off path not activated)	ABC_MON2_MU
0Ah	Redundant switch off path was triggered by MC	LV_OFF_DR_MU_MON
0Bh	Program flow monitoring task 1	ABC_PFM_1_MU
0Ch	Program flow monitoring task 2	ABC_PFM_2_MU
0Dh	Program flow monitoring task 3	ABC_PFM_3_MU
0Eh	Program flow monitoring task 4	ABC_PFM_4_MU
0Fh	Program flow monitoring task 5	ABC_PFM_5_MU
10h	Program flow monitoring task 6	ABC_PFM_6_MU
11h	Program flow monitoring task 7	ABC_PFM_7_MU
12h	Program flow monitoring task 8	ABC_PFM_8_MU
13h	Watchdog error	
14h	Communication error, wrong enable byte	ENA_PORT_MU

After a power up ERR_COD_MU is initialized with zero.

Application conditions on MU:

Initialisation: ERR_COD_MU = 0
only after power up

LV_ERR_TMP_MU = 1
value copied into sent buffer accoring to reset counter on MU

E.33.3.1 Actual error response

If the failure is debounced, a central fault reaction is carried out and the number of reset requests is counted:

Formula section:

- disabling of power stages (see “Interface to power stages”)

IF (RST_CTR_MU < NC_RST_CTR_MAX_MU)

THEN

RST_CTR_MU++;

RESET on MC via NMI;

ELSE

RST_CTR_MU = NC_RST_CTR_MAX_MU + 1;

enter DISABLE state

ENDIF

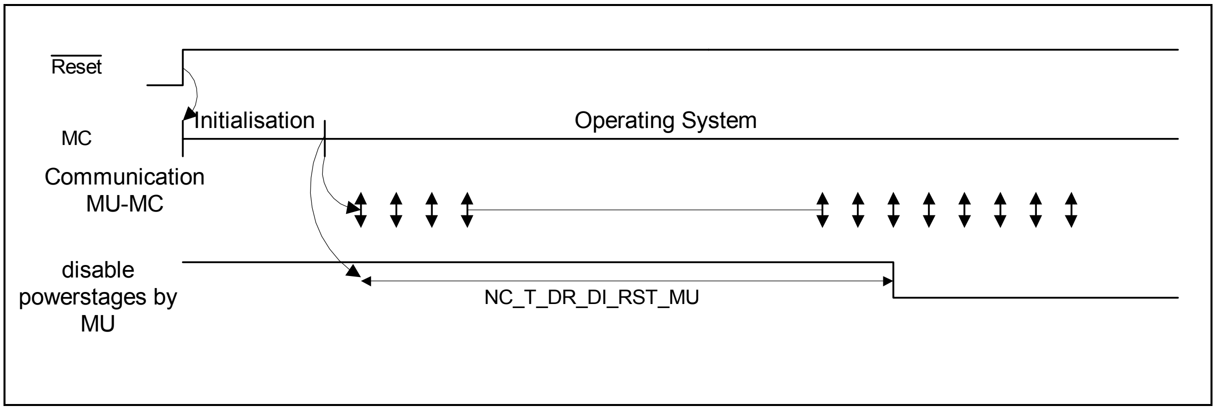
RST_CTR_MU is decremented every NC_T_RST_CTR_MU.

Note: All tests on MU at runtime must not last longer than NC_T_RST_CTR_MU.

E.33.3.2 Fault Reaction after second to (NC_RST_CTR_MAX_MU)th fault reset

(fault does not longer exist)

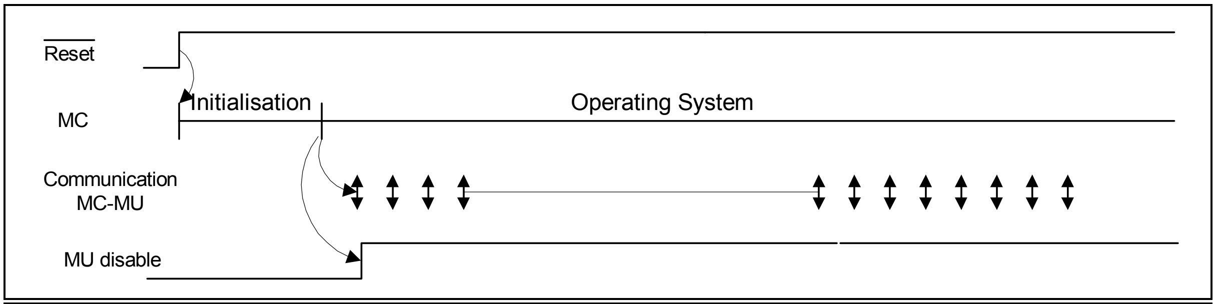
After the second error reset (RST_CTR_MU>1 or RST_CTR_MC>1) MU waits NC_T_DR_DI_RST_MU before it enables the power stages by resetting LV_ERR_TMP_MU. Within the required fault reaction time all Processor Monitoring related tests have to be performed. During this time periode the safety relevant power stages must be disabled. The safety critical fault are listed in the subsection "Classification of Processor Faults". Processor Monitoring Faults during this time periode would lead to another reset with disabled power stages.



E.33.3.3 Fault Reaction after the (NC_RST_CTR_MAX_MU)th fault reset

(fault still exists)

After the (NC_RST_CTR_MAX_MU)th fault reset on the MU a complete test of MU and MC is performed. Faults during that test would lead to permanently disabled power stages for this driving cycle.



Configuration data:

Name	Dim	Hex. Limits	Phys. limits	Resol.	Unit
NC_RST_CTR_MAX_MU	1	00h ... FFh	0...255	1	1
Max value of SW-reset on MU					
NC_T_RST_CTR_MU	1	00h ... FFh	0...255	256	ms
time to decrement the reset counter					
NC_T_DR_DI_RST_MU	1	00h ... FFh	0...255*NC_T_SPI_COM	NC_T_SPI_COM	1
Time delay for enabling power stages after error reset					

E.33.4 Evaluation of MU error information on MC

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
LV_ERR_MU	V/O	0 ... 1h	0 ... 1	1	1
Request of MU to diasble the power stages permanently, sent to MC in DISABLE communication					

Input data:

RST_CTR_MU			

FUNCTION DESCRIPTION:

General information:

The variable LV_ERR_MU indicates on MC that the MU disables the power stages permanently for this driving cycle. LV_ERR_MU is set on MC, if the reset counter of MU reached its maximum. Therefore the reset counter of MU has to be transferred in the INIT and DISABLE communication, see "Communication between MC and MU".

Application conditions:

Initialisation: LV_ERR_MU = 0;

After reset or power up

Formula section:

IF (RST_CTR_MU > NC_RST_CTR_MAX_MU)

THEN

LV_ERR_MU = 1;

ENDIF

E.33.5 Interface to Error memory management of processor monitoring

Output data:

Name	Mode	Hex. Limits	Phys. limits	Resol.	Unit
ERR_COD_MC_SAVE	V/O	0 ... FFh	0 ... 255	1	1
Saved error code of main controller, value of ERR_COD_MC after first communication					
RST_CTR_MC_SAVE	V/O	0 ... FFh	0 ... 255	1	1
Saved reset counter of main controller, value of RST_CTR_MC after first communication					
ERR_COD_MU_SAVE	V/O	0 ... FFh	0 ... 255	1	1
Saved error code of monitoring unit, value of ERR_COD_MU after first communication on main controller					
RST_CTR_MU_SAVE	V/O	0 ... FFh	0 ... 255	1	1
Saved reset counter of monitoring unit, value of RST_CTR_MU after first communication on main controller					

Input data:

ERR_COD_MC	ERR_COD_MU	RST_CTR_MC	RST_CTR_MU

FUNCTION DESCRIPTION:

General information:

If the MC does a software reset due to an error, the MC saves the reason of that error in ERR_COD_MC prior to the reset. ERR_COD_MC is not deleted after a software reset.

The MU has no possibility to keep error information after power down. For this reason the error information is transmitted to MC. If MU releases a software reset due to an error, MU saves the cause of the error in ERR_COD_MU before. ERR_COD_MU is not deleted after a software reset. After the reset the error code is transmitted to MC.

The error code zero, indicates no detected faults. After a normal power up the error codes ERR_COD_MC and ERR_COD_MU are initialized with zero.

After the first communication past a fault reaction (INIT or DISABLE) the complete error information, the values of the main controller as well as the values of the monitoring unit, are available on the main contoller. To provide a storage of this error information the values of ERR_COD_MC, ERR_COD_MU, RST_CTR_MC and RST_CTR_MU has to be saved in other variables. After having updated ERR_COD_MC_SAVE, ERR_COD_MU_SAVE, RST_CTR_MC_SAVE and RST_CTR_MU_SAVE, the module given in "Error memory management of processor monitoring" is called.

To aviod uninitalizes variables ERR_COD_MC_SAVE, ERR_COD_MU_SAVE, RST_CTR_MC_SAVE and RST_CTR_MU_SAVE on MC before the first communcation, a initialization with zero in done.

Application conditions:

Initialisation: *ERR_COD_MC_SAVE = ERR_COD_MU_SAVE = RST_CTR_MC_SAVE = RST_CTR_MU_SAVE = 0*

Recurrence: after each power up or reset

Activation: always

Formula section:

After the first communication (INIT or DISABLE past a fault reaction):

ERR_COD_MC_SAVE = ERR_COD_MC;

ERR_COD_MU_SAVE = ERR_COD_MU;

RST_CTR_MC_SAVE = RST_CTR_MC;

RST_CTR_MU_SAVE = RST_CTR_MU;

CALL "Error memory management of processor monitoring"

E.33.6 Status of power stages by Processor Monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_DR_MU_MC	V/O	1h 2h 4h 8h 10h E0h	"PREDRIVE" "TMP_DIS" "DISABLE_REQ" "DISABLE_MC" "DISABLE_MU" "ENABLE"	1	1
State of Processor Monitoring (MU and MC) according to the safety relevant power stages					
LV_ERR_TMP_MU_MC	V/O	0 ... 1h	0 ... 1	1	1
Reversible disabled power stages by processor monitoring (MC and/or MU)					
LV_ERR_MU_MC	V/O	0 ... 1h	0 ... 1	1	1
Irreversible disabled power stages by processor monitoring (MC and/or MU)					

Input data:

LV_ERR_MC	LV_ERR_MU	LV_ERR_TMP_MU	LV_ERR_TMP_MC
LV_PRDR_ACT	MU_DI_STATE		

FUNCTION DESCRIPTION:

General information:

- LV_ERR_TMP_MU_MC and LV_ERR_MU_MC provide the interface to *Process Monitoring* for disable request of the safety relevant power stages by MC to provide a redundant disabling
- STATE_DR_MU_MC provides an interface to the ASW

The overall state of the *Processor Monitoring* according to the disabling of the safety relevant power stages is stored in the variable STATE_DR_MU_MC, which depends on several logical variables and the state of disable request (refer to "Communication between MC and MU"). This state includes the disabling of the saftey relevant power stages by the MU regarding the predrive check.

- STATE_DR_MU_MC="ENABLE" belongs to a normal operation (no request for disabling the safety relevant power stages, no disable request, no predrive check in progress) of the *Processor Monitoring*. In this state the power stages are enabled with respect to the *Processor Monitoring*. The other way round STATE_DR_MU_MC<>"ENABLE" means disables power stages.
- STATE_DR_MU_MC="PREDRIVE" belongs to a state where the MU disables the power stages, while the MC performs the predrive check.
- ((STATE_DR_MU_MC!="PREDRIVE") AND (STATE_DR_MU_MC!="ENABLE")) belongs to states where the *Processor Monitoring* requests a disabling of the saftey relevant power stages by MC or by MU. Note, that on MC the disabling of the safety relevant power stages is **not** performed by the *Processor Monitoring* itself. It has to be performed elsewhere to perform a redundant disabling.

The logical output variables LV_ERR_TMP_MU_MC and LV_ERR_MU_MC indicate a disable request in case of a set variable:

- LV_ERR_TMP_MU_MC indicates a temporary, reversible disable request of the safety relevant power stages. It is set in case of a set LV_ERR_TMP_MU, a set LV_ERR_TMP_MC. In addition it is set in caser of a set LV_ERR_MU or a set LV_ERR_MC.
- LV_ERR_MU_MC indicates a final, irreversible disable request for this duty cycle of the safety relevant power stages. It is set in case of a set LV_ERR_MU or a set LV_ERR_MC.

Application conditions:

Initialisation: *STATE_DR_MU_MC = "PREDRIVE"*
LV_ERR_TMP_MU_MC = 1
LV_ERR_MU_MC = 0

Recurrence: *STATE_DR_MU_MC, LV_ERR_TMP_MU_MC and LV_ERR_MU_MC are updated each call of the Processor Monitoring as last task.*

Activation: *at each ECU state*

Deactivation: *never*

Formula section:

IF (LV_ERR_TMP_MU==1) **OR** (LV_ERR_TMP_MC==1) **OR**
(LV_ERR_MU==1) **OR** (LV_ERR_MC==1) **THEN** LV_ERR_TMP_MU_MC=1;
ELSE LV_ERR_TMP_MU_MC = 0;

IF (LV_ERR_MU==1) **OR** (LV_ERR_MC==1) **THEN** LV_ERR_MU_MC=1;

IF (LV_PRDR_ACT == 1) **THEN** STATE_DR_MU_MC = "PREDRIVE";
ELSE STATE_DR_MU_MC = "ENABLE";

IF (LV_ERR_TMP_MU_MC == 1) **THEN** STATE_DR_MU_MC = "TMP_DIS";
IF (MU_DI_STATE!=MU_DI_NOT_REQ) **THEN** STATE_DR_MU_MC = "DISABLE_REQ";
IF (LV_ERR_MU == 1) **THEN** STATE_DR_MU_MC = "DISABLE_MU";

IF (LV_ERR_MC == 1) **THEN** STATE_DR_MU_MC = "DISABLE_MC";

E.34 Error memory management of process monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MON	V/O	0 ... 1 H	0 ... 1	1	-
General monitoring error debounced					
LV_ERR_TQI_MON	V/O	0 ... 1 H	0 ... 1	1	-
Torque monitoring error debounced					
LV_ERR_N_MAX_MON	V/O	0 ... 1 H	0 ... 1	1	-
N_LIM monitoring error debounced					
LV_ERR_N_MAX_MON_SAVE	S	0 ... 1 H	0 ... 1	1	-
N_LIM monitoring error saved reset resistant					
ERR_SYM_MON	V	0 H 8 H	NO_SYM SYM_3	1	-
Symtom for general monitoring error					
ERR_SYM_TQI_MON	V	0 H 4 H	NO_SYM SYM_2	1	-
Symtom for torque monitoring error					
ERR_SYM_N_MAX_MON	V	0 H 2 H	NO_SYM SYM_1	1	-
Symptom for N_LIM monitoring error					
LV_END_DIAG_MON	V/O	0 ... 1 H	0 ... 1	1	-
Indicates end of diagnosis of General monitoring error, if set					
LV_END_DIAG_TQI_MON	V/O	0 ... 1 H	0 ... 1	1	-
Indicates end of diagnosis of Torque monitoring error, if set					
LV_END_DIAG_N_MAX_MON	V/O	0 ... 1 H	0 ... 1	1	-
Indicates end of diagnosis of N_LIM monitoring error, if set					
LV_CDN_DIAG_MON	V/O	0 ... 1 H	0 ... 1	1	-
Diagnosis condition for general MON error					
LV_CDN_DIAG_TQI_MON	V/O	0 ... 1 H	0 ... 1	1	-
Diagnosis condition for torque monitoring error					
LV_CDN_DIAG_N_MAX_MON	V/O	0 ... 1 H	0 ... 1	1	-
Diagnosis condition for N_LIM monitoring error					

Input data:

LV_TQI_MON_ACT_MON	LV_ERR_TQI_N_MAX_MON	LV_ERR_N_32_MON
LV_ERR_PVS_MON	LV_ERR_TPS_MON	LV_ERR_MAF_MON
LV_ERR_TQI_AV_MON	LV_ERR_TQ_DIF_P_D_IS_MON	LV_ERR_TQ_DIF_I_IS_MON
LV_ERR_TQ_LOSS_MON	LV_ERR_TQ_MIN_CLU_MON	LV_ERR_TQ_MAX_CLU_MON
LV_ERR_CONV_MON	ERR_SYM_ECU	

FUNCTION DESCRIPTION:

General information:

When a fault is detected by the level 2 of the ETC-Monitoring concept and the reaction is released in the module ‘fault reaction of process monitoring’, the fault is transferred to the error memory management.

The memory area of all output data variables of this module is not permanently RAM tested, the memory area of this module is not cyclic ROM tested.

Description:

Since the different faults can occur in a successive way, they are stored in different locations of the error memory.

The first location contains the logical variables for a fault detected by the diagnosis of the input variables and are stored in LV_ERR_MON.

The faults LV_ERR_TQI_AV_MON and LV_ERR_TQI_N_MAX_MON are stored in separate locations (LV_ERR_TQI_MON and LV_ERR_N_MAX_MON) in order to be able to analyze a scenario of successive faults.

The information whether a warning lamp (ETC MIL) is switched on in case of an error can be found in the general OBD error code table.

Application conditions:

Activation: at every engine state
Deactivation: -
Initialization: for condition see 'Application Incidences of Process Monitoring'
All diagnosis flags (LV_END_DIAG*, LV_CDN_DIAG*, LV_ERR*) and ERR_SYM* has to be reset.
For LV_ERR_N_MAX_MON_SAVE the variables has to be initialized with zero, if a power up is recognized. A power up is detected, if the variables in the reset safe RAM are not consitend.
Update Rate: 40 ms

E.34.1 Error memory management of LV_ERR_MON

Formula section:

LV_CDN_DIAG_MON = 1

If (LV_ERR_CONV_MON = 1
Or LV_ERR_N_32_MON = 1
Or LV_ERR_TQ_DIF_P_D_IS_MON = 1
Or LV_ERR_TQ_DIF_I_IS_MON = 1
Or LV_ERR_TQ_LOSS_MON = 1
Or LV_ERR_TQ_MIN_CLU_MON = 1
Or LV_ERR_TQ_MAX_CLU_MON = 1
Or LV_ERR_PVS_MON = 1
Or LV_ERR_TPS_MON = 1
Or LV_ERR_MAF_MON = 1)
Then
ERR_SYM_MON = SYM_3;
If (LV_ERR_MON = 0) **then**
Set environmental data for LV_ERR_MON according to
"Error Memory Management of Process Monitoring (Appl.Inc.)"
endif

```
        LV_ERR_MON = 1;
        LV_END_DIAG_MON = 1;
    Else
        ERR_SYM_MON = NO_SYM;
        LV_ERR_MON = 0;
        If { delay 520ms } then
        /* After the level2 fault reaction time of 500ms the errors are updated */
            LV_END_DIAG_MON = 1;
        Endif
    Endif
```

E.34.2 Error memory management of LV_ERR_TQI_MON

Formula section:

```
If (LV_TQI_MON_ACT_MON = 1)
Then
    LV_CDN_DIAG_TQI_MON = 1
    If      LV_ERR_TQI_AV_MON = 1
    Then
        ERR_SYM_TQI_MON = SYM_2;
        If (LV_ERR_TQI_MON = 0) then
            Set environmental data for LV_ERR_TQI_MON according to
                "Error Memory Management of Process Monitoring (Appl.Inc.)"
        endif
        LV_ERR_TQI_MON = 1;
        LV_END_DIAG_TQI_MON = 1;
    Else
        ERR_SYM_TQI_MON = NO_SYM;
        LV_ERR_TQI_MON = 0;
        If { delay 520ms } then
        /* After the level2 fault reaction time of 500ms the errors are updated */
            LV_END_DIAG_TQI_MON = 1;
        Endif
    Endif
Else
    LV_CDN_DIAG_TQI_MON = 0
Endif
```

E.34.3 Error memory management of LV_ERR_N_MAX_MON

The error LV_ERR_N_MAX_MON triggers the redundant switch off path (see "Fault Reaction of Process Monitoring") during the actual duty cycle. This finally leads to a reset after debouncing (see "Processor Monitoring"). The entry into the error memory has to be performed past reset. Therefore it is needful to store the error information reset safe; LV_ERR_N_MAX_MON_SAVE is located in the reset safe RAM area. The entry into the error memory is done past reset by evaluation of the information stored before reset.

Application conditions:

Initialisation: LV_ERR_N_MAX_MON_SAVE = 0
only after power up, no initialization after reset

Formula section:

```
LV_CDN_DIAG_N_MAX_MON = 1;
If    (LV_ERR_N_MAX_MON_SAVE = 1)          /* error detected last duty cycle */
Then
   /* error not jet passed to ERRM =>  transfer error information to ERRM*/
   ERR_SYM_N_MAX_MON = SYM_1;
   /* environment data are already reset safe stored before reset */
   LV_ERR_N_MAX_MON = 1;
   LV_END_DIAG_N_MAX_MON = 1;
   /* avoid more than one entry into error memory */
   if {delay 120ms} then
     /* this delay time periode depends on Processor Monitoring */
     /* After this time periode the reset has to be carried out */
     LV_ERR_N_MAX_MON_SAVE = 0;
   end
Else         /* error not detected last duty cycle */
     If (LV_ERR_N_MAX_MON = 1) then          /* error was stored last recurrence */
          ERR_SYM_N_MAX_MON = SYM_1;
          LV_END_DIAG_N_MAX_MON = 1;
     Else
        If { delay 520ms } then
          /* After the level2 fault reaction time of 500ms the errors are updated */
          LV_END_DIAG_N_MAX_MON = 1;
        End
     Endif
Endif
```

```
If (LV_ERR_TQI_N_MAX_MON = 1) then        /* error detected in actual duty cycle */
    If (LV_ERR_N_MAX_MON_SAVE = 0) then

            Set environmental data for LV_ERR_N_MAX_MON reset safe according to
                 "Error Memory Management of Process Monitoring (Appl.Inc.)"

    endif

    /* store information reset save and avoid saving environment data next recurrence */
    LV_ERR_N_MAX_MON_SAVE =1;
end
```

E.34.4 Environmental data management for LV_ERR_ECU

Formula section:

IF ERR_SYM_ECU = 1H

THEN Set environmental data for LV_ERR_ECU according to

"Error memory management of process monitoring (appl. inc.)

ENDIF

E.34.5 Error memory management of LV_ERR_N_MAX_MON

E.35 Error memory management of process monitoring (Appl.Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ENVD_0_MON	O/V	0...FFH	0...255	1	[-]
Environmental data 0 for general monitoring error					
ENVD_1_MON	O/V	0...FFH	0...255	1	[-]
Environmental data 1 for general monitoring error					
ENVD_2_MON	O/V	0...FFH	0...255	1	[-]
Environmental data 2 for general monitoring error					
ENVD_3_MON	O/V	0...FFH	0...255	1	[-]
Environmental data 3 for general monitoring error					
DIAG_INST_MON	V/S	0H 1H 2H 3H 4H 5H 6H 7H 8H 9H 10H	NO_ERROR N_32_MON TQ_DIF_P_D_IS _MON TQ_DIF_I_IS_M ON TQ_LOSS_MON TQ_MIN_CLU_M ON TQ_MAX_CLU_ MON PVS_MON TPS_MON MAF_MON CONV_MON	1	-
Diagnosis instance for ETC monitoring					
ERR_INTM_DIAG_INST_MON	V/S	0H 1H 2H 3H 4H 5H 6H 7H 8H 9H 10H	NO_ERROR N_32_MON TQ_DIF_P_D_IS _MON TQ_DIF_I_IS_M ON TQ_LOSS_MON TQ_MIN_CLU_M ON TQ_MAX_CLU_ MON PVS_MON TPS_MON MAF_MON CONV_MON	1	-
Diagnosis instance for intermittent monitoring error					
STATE_ERR_INTM_MON	V/S	0H 1H 2H 3H 4H	NO_ERROR ERROR_PATH1 ERROR_PATH2 ERROR_PATH3 ERROR_PATH4	1	[-]
State variable for intermittent monitoring error					

Input data:

LV_ERR_CONV_MON	LV_TQI_MON_ACT_MON	TPS_AV
MAF_MON	ECU_STATE	LV_ERR_N_32_MON
LV_PAS_RAMP_ACT_P_D_CHG_MON	LV_REQ_ISC	LV_ERR_TQ_DIF_P_D_IS_MON
N_32_MON	TQ_DIF_P_D_IS_MON	LV_IS

LV_PAS_RAMP_ACT_I_CHG_MON	TQ_DIF_I_IS_MON	LV_ERR_TQ_DIF_I_IS_MON
TQ_LOSS_MON	TCO_MON	LV_ERR_TQ_LOSS_MON
LV_TQ_MIN_CLU	TQ_MIN_CLU_MON	LV_ERR_TQ_MIN_CLU_MON
LV_ERR_TQ_MAX_CLU_MON	LV_ERR_PVS_MON	TQ_MAX_CLU_MON
V_PVS_1_BAS	V_PVS_2_BAS	PV_AV
LV_ERR_TPS_1	LV_ERR_TPS_2	LV_ERR_TPS_MON
LV_ERR_TQI_AV_MON	TQI_SP_MON	LV_ERR_MAF_MON
MAF_SUB_COR_MMV_MON	V_TPS_2_BAS	TQI_AV_MON
FUP	V_TPS_1_BAS	MFF_SP_MV
PREV_STATE_IV	N_DIF_SP_IS_MON	LV_ERR_TQI_N_MAX_MON
LV_ERR_TPS	LV_ERR_PVS	LV_IGK_MON
PV_AV_MON	LV_N_LIM_ETC_LIH	LV_MTC_CUR_OFF
STATE_ERR_TQ_DIF_P_D_IS_MON	ACQ_MU_0_MON	ACQ_MC_0_MON
STATE_ERR_TQ_MIN_CLU_MON	STATE_ERR_DET_TQ_MIN_MON	STATE_ERR_TQ_DIF_I_IS_MON
LV_BRAKE_MON	ECU_DIAG_SF_HB	TQ_LOSS_REQ_CLU_MON
ERR_COD_MC_SAVE	RST_CTR_MC_SAVE	ECU_DIAG_SF_LB
ABC_N_32_MON	ABC_TQ_DIF_P_D_IS_MON	RST_CTR_MU_SAVE
ABC_TQ_DIF_I_IS_MON	ABC_TQ_LOSS_MON	ABC_TQ_MIN_CLU_MON
ABC_TQ_MAX_CLU_MON	ABC_PVS_RATIO_MON	ABC_TPS_RATIO_MON
ABC_MAF_MON	ABC_CONV_MON	STATE_ERR_TQ_MIN_CLU_MON
STATE_ERR_TQ_DIF_P_D_IS_MON	STATE_ERR_TQ_DIF_I_IS_MON	LV_ERR_PVS_1
LV_ERR_PVS_2	LV_ERR_PVS_RATIO	ERR_SYM_PVS_RATIO

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
LC_ERR_DET_UPD_MON	1	0...1H	0...1	1	[-]
Logical constant to enable detection of intermittent monitoring error					
LC_ERR_INTM_MON_CLR	1	0...1H	0...1	1	[-]
Logical constant to erase intermittent monitoring error memory					

FUNCTION DESCRIPTION:

General information:

For every fault, the environmental conditions are stored in the environmental data variables. The first byte is used for encoding the type of the fault if more than one level 2 error could lead to the error. The variables are chosen such that the fault scenario can be analyzed.

Error	First byte	Second Byte	Third byte	Fourth byte
LV_ERR_MON	ENVD_0_MON	ENVD_1_MON	ENVD_2_MON	ENVD_3_MON
LV_ERR_TQI_MON	ENVD_0_MON	ENVD_1_MON	ENVD_2_MON	ENVD_3_MON
LV_ERR_N_MAX_MON	ENVD_0_MON	ENVD_1_MON	ENVD_2_MON	ENVD_3_MON
LV_ERR_ECU	ENVD_0_MON	ENVD_1_MON	ENVD_2_MON	ENVD_3_MON
LV_ERR_MON_3	ENVD_0_MON	ENVD_1_MON	ENVD_2_MON	ENVD_3_MON

The memory area of all output data variables of this module is not cyclical RAM tested,
the memory area of this module is not cyclic ROM tested.

Application conditions:

Initialization: -

Activation: *according to "Error Memory management of Process Monitoring"*

Update Rate: *according to "Error Memory management of Process Monitoring"*

Formula section:

Since the different faults can occur in a successive way, they are stored in different locations of the failure memory.
The first location contains the logical variables for a fault detected by the diagnosis of the input variables and are stored in LV_ERR_MON.

The faults LV_ERR_TQI_MON and LV_ERR_N_MAX_MON are stored in separate locations in order to be able to analyze a scenario of successive faults.

All environmental data configuration is listed in the following table:

Error	Error Cause	Environmental data			
		First byte	Second Byte	Third byte	Fourth byte
LV_ERR_MON	LV_ERR_CONV_MON	BIT 0...4: 1*dec.* BIT 5: LV_TQI_MON_ACT_MON BIT 6/7: 0	ACQ_MC_0_MON (low byte)	ACQ_MU_0_MON (low byte)	0
	LV_ERR_N_32_MON	BIT 0...4: 2*dec.* BIT 5: LV_TQI_MON_ACT_MON BIT 6/7: 0	N_32_MON	N_32_SUB_MON	ECU_STATE
	LV_ERR_TQ_DIF_P_D_IS_MON	BIT 0...4: 3*dec.* BIT 5: LV_PAS_RAMP_ACT_P_D_CHG_MON BIT 6: LV_REQ_ISC BIT 7: LV_IS	N_32_MON	TQ_DIF_P_D_IS_MON (high byte)	STATE_ERR_TQ_DIF_P_D_IS_MON
	LV_ERR_TQ_DIF_I_IS_MON	BIT 0...4: 4*dec.* BIT 5: LV_PAS_RAMP_ACT_I_CHG_MON BIT 6: LV_REQ_ISC BIT 7: LV_IS	N_32_MON	TQ_DIF_I_IS_MON (high byte)	STATE_ERR_TQ_DIF_I_IS_MON
	LV_ERR_TQ_LOSS_MON	BIT 0...4: 5*dec.* BIT 5: 0 BIT 6: 0 BIT 7: 0	N_32_MON	TQ_LOSS_MON (high byte)	TCO_MON
	LV_ERR_TQ_MIN_CLU_MON	BIT 0...4: 6*dec.* BIT 5: LV_TQ_MIN_CLU BIT 6: STATE_ERR_TQ_MIN_CLU_MON (BIT 0) Bit 7: STATE_ERR_TQ_MIN_CLU_MON (Bit 1)	N_DIF_SP_IS_MON	TQ_MIN_CLU_MON (high byte)	STATE_ERR_DET_TQ_MIN_MON
	LV_ERR_TQ_MAX_CLU_MON	BIT 0...4: 7*dec.* BIT 5: LV_TQI_MON_ACT_MON Bit 6/7: 0	N_32_MON	TQ_MAX_CLU_MON (high byte)	TQ_LOSS_REQ_CLU_MON (high byte)
	LV_ERR_PVS_MON	BIT 0...4: 8*dec.* BIT 5:LV_TQI_MON_ACT_MON BIT 6: LV_BRAKE_MON 7: 0	PV_AV	V_PVS_1_BAS (bit 2..9)	V_PVS_2_BAS (bit 2..9)
	LV_ERR_TPS_MON	BIT 0...4: 9*dec.* BIT 5/6/7: 0	TPS_AV (bit 6..13)	V_TPS_1_BAS (bit 2..9)	V_TPS_2_BAS (bit 2..9)
	LV_ERR_MAF_MON	BIT 0...4: 10*dec.* BIT 5: LV_ERR_TPS_1 BIT 6: LV_ERR_TPS_2 BIT 7: 0	N_32_MON	MAF_MON	MAF_SUB_COR_MMV_MON
LV_ERR_TQI_MON	LV_ERR_TQI_AV_MON	BIT 0...4: 11*dec.* BIT 5/6/7: 0	TQI_SP_MON	TQI_AV_MON	PV_AV_MON
LV_ERR_N_MAX_MON	LV_ERR_TQI_N_MAX_MON	BIT 0...4: 12*dec.* BIT 5/6/7: 0	PREV_STATE_IV	Status byte 2	Status byte 3
LV_ERR_ECU	LV_ERR_ECU	BIT 0...4: 13*dec.* BIT 5/6/7: 0	0	ECU_DIAG_SF_HB	ECU_DIAG_SF_LB

Error	Error Cause	Environmental data			
LV_ERR_MON_3	LV_ERR_MON_3	BIT 0...4: 14*dec.* BIT 5/6/7: 0	ERR_COD_MC_SAVE	RST_CTR_MC_SAVE	RST_CTR_MU_SAVE

First Byte: The following rule is used:
The lower 5 BITs (BIT 0...4) are used for encoding the type of the fault (0dec ... 31dec can be used).
The higher 3 BITs (BIT 5...7) can be used to store additional logical information according the table above.

Bit definition of first byte:

BIT7 [| | | | | | |] BIT0

The information whether a warning lamp (ETC MIL) is switched on in case of an error can be found in the general OBD error code table.

In case of more than one error cause occurs, always the first error (according to the table above; see BIT 0...4 of first Byte for the order) is taken into account.

E.35.1 Error memory management of LV_ERR_TQI_N_MAX_MON

Bit definition status byte 2:

BIT7 [| | | | | | |] BIT0

- BIT0 -> LV_ERR_TQI_AV_MON
- BIT1 -> LV_IGK_MON
- BIT2 -> LV_ERR_TPS
- BIT3 -> LV_ERR_PVS
- BIT4 -> LV_MTC_CUR_OFF
- BIT5 -> (PV_AV_MON = 0)
- BIT6 -> LV_N_LIM_ETC_LIH
- BIT7 -> (LV_ERR_TQ_LOSS_MON=1 **OR** LV_ERR_TQ_MIN_CLU_MON=1 **OR** LV_ERR_TQ_MAX_CLU_MON=1)

Bit definition status byte 3:

BIT7 [| | | | | | |] BIT0

- BIT0 -> LV_ERR_CONV_MON
- BIT1 -> LV_MSR_INH_MON
- BIT2 -> LV_ERR_N_32_MON
- BIT3 -> (LV_ERR_TQ_DIF_P_D_IS_MON=1 **OR** LV_ERR_TQ_DIF_I_IS_MON=1)
- BIT4 -> LV_ERR_PVS_MON
- BIT5 -> LV_ERR_TPS_MON
- BIT6 -> LV_ERR_MAF_MON
- BIT7 -> 0

List of error codes and symptoms

The list of error codes and the list of error symptoms can be found in the chapter "Table of failures".

Configuration for diagnostic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
General monitoring error		SYM_0	*NO*
		SYM_1	
		SYM_2	
MON	General monitoring error	SYM_3	

List of Environmental Data to store in Failure Memory: ENVD_0_MON
ENVD_1_MON
ENVD_2_MON
ENVD_3_MON

Torque monitoring error		SYM_0	*NO*
		SYM_1	
	Torque monitoring	SYM_2	
TQI_MON		SYM_3	

List of Environmental Data to store in Failure Memory: ENVD_0_MON
ENVD_1_MON
ENVD_2_MON
ENVD_3_MON

N_LIM monitoring error		SYM_0	*NO*
	N_LIM monitoring error	SYM_1	
		SYM_2	
N_MAX_MON		SYM_3	

List of Environmental Data to store in Failure Memory: ENVD_0_MON
ENVD_1_MON
ENVD_2_MON
ENVD_3_MON

E.35.2 Diagnosis instance for ETC monitoring

DIAG_INST_MON was defined to be able to see different monitoring errors in an easy way.

Application conditions:

Initialization: *AT FMY reset DIAG_INST_MON = 0H*

Activation: *LV_ERR_MON = 1*

Update Rate: *according to "Error Memory management of Process Monitoring"*

Formula section:

IF	LV_ERR_N_32_MON = 1	**THEN**	DIAG_INST_MON = 1H
ELSE IF	LV_ERR_TQ_DIF_P_D_IS_MON = 1	**THEN**	DIAG_INST_MON = 2H
ELSE IF	LV_ERR_TQ_DIF_I_IS_MON = 1	**THEN**	DIAG_INST_MON = 3H
ELSE IF	LV_ERR_TQ_LOSS_MON = 1	**THEN**	DIAG_INST_MON = 4H
ELSE IF	LV_ERR_TQ_MIN_CLU_MON = 1	**THEN**	DIAG_INST_MON = 5H
ELSE IF	LV_ERR_TQ_MAX_CLU_MON = 1	**THEN**	DIAG_INST_MON = 6H
ELSE IF	LV_ERR_PVS_MON = 1	**THEN**	DIAG_INST_MON = 7H
ELSE IF	LV_ERR_TPS_MON = 1	**THEN**	DIAG_INST_MON = 8H
ELSE IF	LV_ERR_MAF_MON = 1	**THEN**	DIAG_INST_MON = 9H
ELSE IF	LV_ERR_CONV_MON = 1	**THEN**	DIAG_INST_MON = 10H

If more than one of above mentioned error occurs, just write the first one in the DIAG_INST_MON and do not overwrite this one by a second or third ... error.

E.35.3 Storage of last intermittent monitoring error

FUNCTION DESCRIPTION:

General information:

The purpose of this function is the visualization of the diagnosis instances that show intermittent monitoring error (antibounce counter >= 2) but do not enter into the dynamic error management structure, because the antibounce counter does not reach its maximum value.
If the antibounce counter of a monitoring error using the anti-bounce algorithm is incremented for the 2^{nd} time, it is stored in the array ERR_INTM_DIAG_INST_MON.

Application conditions:

Initialization:	At FMY reset or LC_ERR_INTM_MON_CLR 0 ->1
	ERR_INTM_DIAG_INST_MON = 0
	STATE_ERR_INTM_MON = 0
Recurrence:	*according to "Error Memory management of Process Monitoring"*
Activation:	LC_ERR_DET_UPD_MON = 1

Formula section:

```
If      ABC_N_32_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  1H                        {N_32_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_TQ_DIF_P_D_IS_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  2H                        {TQ_DIF_P_D_IS_MON}
        If STATE_ERR_TQ_DIF_P_D_IS_MON <> 0H
        Then
        STATE_ERR_INTM_MON =  STATE_ERR_TQ_DIF_P_D_IS_MON
        EndIf

ElseIf  ABC_TQ_DIF_I_IS_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  3H                        {TQ_DIF_I_IS_MON}
        If STATE_ERR_TQ_DIF_I_IS_MON <> 0H
        Then
        STATE_ERR_INTM_MON =  STATE_ERR_TQ_DIF_I_IS_MON
        EndIf

ElseIf  ABC_TQ_LOSS_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  4H                        {TQ_LOSS_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_TQ_MIN_CLU_MON >= 2
```

```
Then
        ERR_INTM_DIAG_INST_MON =  5H                                {TQ_MIN_CLU_MON}
        If STATE_ERR_TQ_MIN_CLU_MON <> 0H
        Then
        STATE_ERR_INTM_MON =  STATE_ERR_TQ_MIN_CLU_MON
        EndIf

ElseIf  ABC_TQ_MAX_CLU_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  6H                                {TQ_MAX_CLU_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_PVS_RATIO_MON >= 2  AND
        LV_ERR_PVS_1 = 0        AND
        LV_ERR_PVS_2 = 0        AND
        LV_ERR_PVS_RATIO = 0    AND
        ERR_SYM_PVS_RATIO = 0
Then
        ERR_INTM_DIAG_INST_MON =  7H                                {PVS_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_TPS_RATIO_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  8H                                {TPS_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_MAF_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  9H                                {MAF_MON}
        STATE_ERR_INTM_MON = 1H

ElseIf  ABC_CONV_MON >= 2
Then
        ERR_INTM_DIAG_INST_MON =  10H                               {CONV_MON}
        STATE_ERR_INTM_MON = 1H

EndIf
```

E.36 Error memory management of processor monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_ERR_MON_3	V/O	0...1H	0...1	1	[-]
Processor Monitoring error detected					
LV_CDN_DIAG_MON_3	V	0...1H	0...1	1	[-]
Diagnosis condition for general Processor Monitoring error					
LV_END_DIAG_MON_3	V	0...1H	0...1	1	[-]
Indicates that diagnosis is completely finished					
ERR_SYM_MON_3	V	0H 1H 2H 4H 8H	NO_SYM SYM_0 SYM_1 SYM_2 SYM_3	1	[-]
Error symptom extracted from ERR_COD_MC and ERR_COD_MU					

Input data:

RST_CTR_MC_SAVE	RST_CTR_MU_SAVE	ERR_COD_MC_SAVE	ERR_COD_MU_SAVE

FUNCTION DESCRIPTION:

General information:

This module here provides a link to the customer error memory managment.

Detection of a point in time to store the error information non volatile

When a fault is detected by the Processor Monitoring (see "fault reaction of processor monitoring"), a reset is done or a transition to DISABLE is preformed. The complete error information is available on the main controller after the first communication past the reset or after the first DISABLE communication (cf. activation conditions of the module).

Performing the storage of the error information

If an error information is avaliable, LV_ERR_MON_3 performs a transition from 0 to 1. A set logical variable LV_CDN_DIAG_MON_3 indicates that the conditions for storing an error information is true; this is the case, if either ERR_COD_MC_SAVE or ERR_COD_MU_SAVE is nonzero.

A rising edge of LV_ERR_MON_3 triggers the storage of the error information ERR_SYM_MON_3 into the customer error memory. ERR_SYM_MON_3 consists of four bits:

Bit3	Bit2	Bit1	Bit0 (LSB)
8	4	2	1
Level 2' error	Error in selftest/ Predrive check error	not used	ROM/RAM error on MC

Regardless of whether an error is present, LV_END_DIAG_MON_3 must be set because this indicates the end of the "diagnosis", whether an error was found or not; a similar remark applies to LV_CDN_DIAG_MON_3.

Application conditions:

Initialisation: */* Performed by the standard RAM check */*

LV_ERR_MON_3 = 0;
LV_CDN_DIAG_MON_3 = 0;
LV_END_DIAG_MON_3 = 0;
ERR_SYM_MON_3 = NO_SYM;

Activation: This module is called by Processor Monitoring as a subroutine on demand in two places:

- after the INIT communication;
- after the first DISABLE communication when either RST_CTR_MC or RST_CTR_MU are greater than NC_RST_CTR_MC_MAX or NC_RST_CTR_MU_MAX, respectively.

In both cases the complete error information (MC and MU) is available on the MC.

Note: In case of a Processor Monitoring error one cannot ensure a correct error memory management due to a possible damaged ECU.

Formula section:

/ Initialisation: it is* **extremely important** *to reset LV_ERR_MON_3; otherwise, the first MON_3 error can never be updated */*

LV_ERR_MON_3 = 0;
LV_CDN_DIAG_MON_3 = 0;
ERR_SYM_MON_3 = NO_SYM;

IF ((ERR_COD_MC_SAVE>0) AND (RST_CTR_MC_SAVE>1)) **OR** */* MC error */*

((ERR_COD_MU_SAVE>0) AND (RST_CTR_MU_SAVE>1)) **THEN** */* MU error */*

/ When entering this block, an error code is set and diagnosis is finished */*

Set environmental data for LV_ERR_MON_3 according to "Error Memory Management of Process Monitoring (Appl.Inc.)"

/ Now set the filtered error information ERR_SYM_MON_3 */*

/ according to customer specification by evaluation of */*

/ ERR_COD_MC_SAVE, RST_CTR_MC_SAVE and */*

/ ERR_COD_MU_SAVE, RST_CTR_MU_SAVE */*

IF (RST_CTR_MC_SAVE>0) **THEN**

IF (ERR_COD_MC_SAVE == 0Bh) **OR** (ERR_COD_MC_SAVE == 09h) **THEN**
ERR_SYM_MON_3 = SYM_0; */* ROM/RAM error on MC */*
END

IF (ERR_COD_MC_SAVE == 0Ch) **OR** (ERR_COD_MC_SAVE == 0Ah))**THEN**
ERR_SYM_MON_3 = SYM_0; */* RAM/RAM error on MC */*
END

```
        IF (ERR_COD_MC_SAVE == 3h) THEN                 /* Predrive check error */
                ERR_SYM_MON_3 = SYM_2;
        END
    END
    IF (RST_CTR_MU_SAVE >0) THEN
        IF (ERR_COD_MU_SAVE == 09h) THEN                /* level 2' error */
                ERR_SYM_MON_3 = SYM_3;
        END
    END
    /* now some specific error codes are evaluated and ERR_SYM_MON_3 is set */
    /* all unspecific errors are threated as "Error in selftest" */
    IF (ERR_SYM_MON_3 == NO_SYM) THEN
            ERR_SYM_MON_3 = SYM_2;
    END
    LV_ERR_MON_3 = 1;
END

/* LV_END_DIAG_MON_3 has to be set regardless of whether an error was found or not; otherwise, the diagnosis will never be ready in the good case (no error present) */
LV_END_DIAG_MON_3 = 1;
/* LV_CDN_DIAG_MON_3 is activated: whenever this module is called, the processor monitoring diagnosis condition is given */
LV_CDN_DIAG_MON_3 = 1;
```

E.37 Error Memory Management of Processor Monitoring (Appl.Inc.)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
ENVD_0_MON_3	O	0...FFH	0...255	1	[-]
Environmental data 0 for general monitoring error					
ENVD_1_MON_3	O	0...FFH	0...255	1	[-]
Environmental data 1 for general monitoring error					
ENVD_2_MON_3	O	0...FFH	0...255	1	[-]
Environmental data 2 for general monitoring error					
ENVD_3_MON_3	O	0...FFH	0...255	1	[-]
Environmental data 3 for general monitoring error					

Input data:

ERR_COD_MC_SAVE	RST_CTR_MC_SAVE	ERR_COD_MU_SAVE	RST_CTR_MU_SAVE

FUNCTION DESCRIPTION:

General information:

For every fault, the environmental conditions are stored in the environmental data variables. The variables are chosen such that the fault scenario can be analyzed.

Formula section:

ENVD_0_MON_3 = ERR_COD_MC_SAVE;
ENVD_1_MON_3 = RST_CTR_MC_SAVE;
ENVD_2_MON_3 = ERR_COD_MU_SAVE;
ENVD_3_MON_3 = RST_CTR_MC_SAVE;

E.37.1 List of error codes and symptoms

The list of error codes and the list of error symptoms can be found in the chapter "Table of failures".

Configuration diagnostic symptoms:

Diagnosis XX	Symptom	Nr	ABC type
Diagnosis description Processor Monitoting Error	*ROM error on MC*	*0*	NO
	RAM error on MC	*1*	
	Error in selftest/ Predrive check error	*2*	
	Level 2' error	*3*	
MON_3	*General ECU error*		

F Tuning functions

Table of Contents

Index

F.1 Misfire tuning functions

F.1.1 Misfire software generator

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
INH_IV_MIS_GEN	V/O	0...FFH	0...255	1	[-]
Identification of cylinders shut off by injection misfire generator					
INH_IGC_MIS_GEN	V/O	0...FFH	0...255	1	[-]
Identification of cylinders shut off by ignition misfire generator					
LV_MIS_GEN	V	0...1H	0...1	1	[-]
Flag to indicate the missing ignition and/or injection (current segment)					
LV_MIS_GEN_DET	V	0...1H	0...1	1	[-]
Flag to indicate the missing ignition and/or injection, delayed to be synchronised with misfire detection flags					
INH_IV_MIS_GEN_16	V/O	0...FFFH	0...4095	1	[-]
Identification of cylinders shut off by injection misfire generator					
INH_IGC_MIS_GEN_16	V/O	0...FFFH	0...4095	1	[-]
Identification of cylinders shut off by ignition misfire generator					

Input data:

SEG_NR	N	TD_IGC[NC_CYL_NR]	IGA_AV[NC_CYL_NR]
EOI_LIM[NC_CYL_NR]	SOI_MAX		

FUNCTION DESCRIPTION:

According the project SI integration plan and/or customer requirements/wishes, a software misfire generator can be included on EMS softwares and used during integration validation and calibration stages. It is strongly recommended to remove this functionally on serial product software: integration choice via NLC_USE_MIS_GEN compilation switch.

Application conditions:

Initialisation: at reset & engine stalling

LV_MIS_GEN = 0

LV_MIS_GEN_DET = 0

INH_IV_MIS_GEN_16 = 0

INH_IGC_MIS_GEN_16 = 0

INH_IV_MIS_GEN = 0

INH_IGC_MIS_GEN = 0

No function calls to INJR & IGRE

Recurrence: Segment

Activation:

The regular or random misfiring generation is applied on ignition and/or injection sequences.

If LC_IGN_OFF_MIS_GEN = 1

Then Ignition cut-off mode

(using INH_IGC_MIS_GEN_16 carrier, see § Ignition misfire realisation)

LV_MIS_GEN is set to 1 according misfiring generation mode selected

(pseudo random or regular mode, see dedicated chapters below)

EndIf

If LC_INJ_OFF_MIS_GEN = 1

Then Injection cut-off mode

(using INH_IV_MIS_GEN_16 carrier, see § Injection misfire realisation)

LV_MIS_GEN is set to 1 according misfiring generation mode selected

(pseudo random or regular mode, see dedicated chapters below)

EndIf

If LC_IGN_OFF_MIS_GEN = 1

And LC_INJ_OFF_MIS_GEN = 1

Then Ignition and injection off modes (for the same TDC)

(using INH_IGC_MIS_GEN_16 & INH_IV_MIS_GEN_16 carrier)

LV_MIS_GEN is set to 1 according misfiring generation mode selected

(pseudo random or regular mode, see dedicated chapters below)

EndIf

If LC_IGN_OFF_MIS_GEN = 0

And LC_INJ_OFF_MIS_GEN = 0

Then No misfire generation allowed

LV_MIS_GEN = 0

INH_IV_MIS_GEN = 0

INH_IGC_MIS_GEN = 0

INH_IV_MIS_GEN_16 = 0

INH_IGC_MIS_GEN_16 = 0

No function calls to INJR & IGRE

EndIf

LV_MIS_GEN_DET = LV_MIS_GEN delayed with C_NR_TDC_DLY_MIS_GEN tdc's

INH_IV_MIS_GEN = u8(INH_IV_MIS_GEN_16) // management only for the 8 cylinders

INH_IGC_MIS_GEN = u8(INH_IGC_MIS_GEN_16)

F.1.1.1 Pseudo random misfiring generation mode

<u>Application conditions:</u>

The misfiring generation mode is authorised when the misfiring tuning function and the pseudo random misfiring tuning function are active.

This misfiring generation mode is global to all cylinders.

Recurrence: Segment

Activation:

If LC_REQ_MIS_GEN = 1

And C_MOD_MIS_GEN = 0

And SEG_NR = C_SEG_ST_MIS_GEN

Then Pseudo random misfiring generation is triggered

EndIf

Deactivation:

If LC_REQ_MIS_GEN = 0

Or C_MOD_MIS_GEN ≠ 0

Then Pseudo random misfiring generation mode is disabled

EndIf

Formula section:

A random ignition and/or injection is missing following the bitfield sequence ID_RND_PAT_MIS_GEN, where each bit corresponds to a TDC.

If the current bit value is 0 then the ignition and/or injection are missing; else ignition and injection are realised.

During the TDC where the ignition and/or injection are missing **And** if at least one of the cut-off mode is allowed (LC_IGN_OFF_MIS_GEN = 1 **Or** LC_INJ_OFF_MIS_GEN = 1), the flag LV_MIS_GEN is set to 1 else the flag LV_MIS_GEN is set to 0.

F.1.1.2 Regular misfiring generation mode

Application conditions:

This generation mode is authorised when the misfiring tuning function and the regular misfiring tuning function are active.

This misfiring generation mode is global to all cylinders.

Recurrence: Segment

Activation:

If LC_REQ_MIS_GEN = 1

And C_MOD_MIS_GEN = 1

And SEG_NR = C_SEG_ST_MIS_GEN

Then Regular misfiring generation mode is triggered

EndIf

Deactivation:

If LC_REQ_MIS_GEN = 0

Or C_MOD_MIS_GEN ≠ 1

Then Regular misfiring generation mode is disabled

EndIf

Formula section:

Regular ignition and/or injection are missing every C_MIS_RATE tdc.

During the TDC where the ignition and/or injection are missing **And** if at least one of the cut-off mode is allowed (LC_IGN_OFF_MIS_GEN = 1 **Or** LC_INJ_OFF_MIS_GEN = 1), the flag LV_MIS_GEN is set to 1 else the flag LV_MIS_GEN is set to 0.

F.1.1.3 Cylinder regular misfiring generation mode

Application conditions:

This generation mode is authorised when the misfiring tuning function and the regular misfiring tuning function are active.

This misfiring mode is dedicated to specific cylinder generation.

Recurrence: Segment

Activation:

If LC_REQ_MIS_GEN = 1

And C_MOD_MIS_GEN = 2

And SEG_NR = C_SEG_ST_MIS_GEN

Then Cylilnder regular misfiring generation mode is triggered

EndIf

Deactivation:

If LC_REQ_MIS_GEN = 0

Or C_MOD_MIS_GEN ≠ 2

Then Cylinder regular misfiring generation mode is disabled

EndIf

Formula section:

If(1) LC_MIS_RATE_ENA_CYL[SEG_NR] = 1

Then(1) Ignition and/or injection are missing on cylinder SEG_NR every C_RATE_MIS_CYL[SEG_NR] tdc's.

If(2) During the TDC where the ignition **And/Or** injection are missing

And at least one of the cut-off mode is allowed (LC_IGN_OFF_MIS_GEN = 1 **Or** LC_INJ_OFF_MIS_GEN = 1),

Then(2) LV_MIS_GEN = 1

Else(2) LV_MIS_GEN = 0

EndIf(2)

Else(1) No Ignition and/or injection are missing on cylinder SEG_NR

LV_MIS_GEN is set to 0

EndIf(1)

F.1.1.4 Injection misfire realisation

Recurrence: Every engine cycle, phased with required cylinder injection

Formula section:

The injection misfire realisation is realised via the INH_IV_MIS_GEN_16 carrier. This is an input for the Injection Cylinder Shut-Off Application Incidences file.

Cylinder switched off by injection software misfire generator is set to 1 within INH_IV_MIS_GEN_16 carrier.

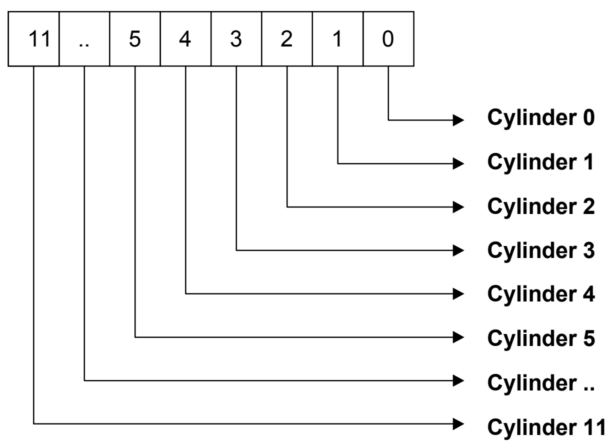
All others cylinder bits are reset to 0 only if not set by the misfire pattern defined for the current engine cycle

Important note on injection shut-off triggering:

To be effective, the injection shut-off on Cylinder x must be triggered before the injection time starts on this cylinder. To evaluate the correct phasing for the injection shut-off, the maximum SOI *(start of injection angle)* is evaluated.

INH_IV_MIS_GEN_16 carrier must be updated at the segment just before the maximum start of injection (SOI_MAX + EOI_LIM[0]) to be strictly taken in account on this injection phase.

INH_IV_MIS_GEN_16



F.1.1.5 Ignition misfire realisation

Recurrence: Every engine cycle, phased with required cylinder /coil ignition

Formula section:

The ignition misfire realisation is realised via the INH_IGC_MIS_GEN_16 carrier. This is an input for the Ignition Cylinder Shut-Off Application Incidences file.

Cylinder switched off by ignition software misfire generator is set to 1 within INH_IGC_MIS_GEN_16 carrier.

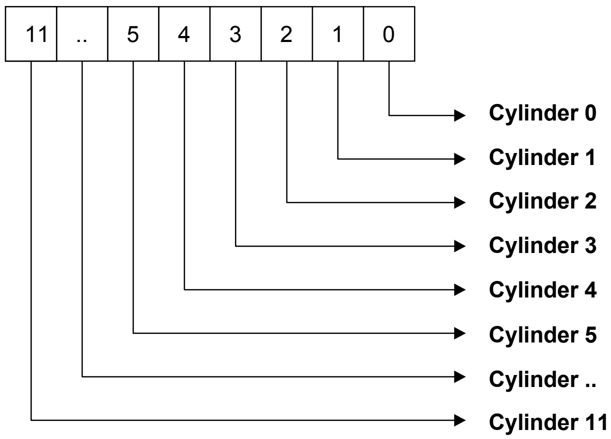
All others cylinder bits are reset to 0 only if not set by the misfire pattern defined for the current engine cycle

Important note on ignition shut-off triggering:

To be effective, the ignition shut-off on cylinder x must be triggered on the segment before the dwell time on the coil y starts for this cylinder x.

To evaluate the correct phasing for the ignition shut-off, the period TD_IGC[x] must be converted in angle and added to the IGA_AV[x] angle to defined the correct segment for triggering.

INH_IGC_MIS_GEN_16



Calibration data:

Name	Dim.	Hex. limits	Phys. limits	Resol.	Unit
LC_REQ_MIS_GEN	1	0...1H	0...1	1	[-]
Misfire software generator request					
C_MOD_MIS_GEN	1	0...2H	0...2	1	[-]
Misfire software generator function mode					
C_MIS_RATE	1	0...FFH	0...255	1	[-]
Misfire rate for regular misfire generator mode					
C_MIS_RATE_CYL[NC_CYL_NR]	1	0...FFH	0...255	1	[-]
Misfire rate for cylinder regular misfire generator mode					
LC_IGN_OFF_MIS_GEN	1	0...1H	0...1	1	[-]
Boolean for misfire generation with ignition cut-off (=1)					
LC_INJ_OFF_MIS_GEN	1	0...1H	0...1	1	[-]
Boolean for misfire generation with injection cut-off (=1)					
C_SEG_ST_MIS_GEN	1	0...7H	0...7	1	[-]
Start segment for misfire software generator triggering					
C_NR_TDC_DLY_MIS_GEN	1	0...FH	0...15	1	[-]
Delay to synchronise LV_MIS_GEN_DET with misfire detection flags					
LC_MIS_RATE_ENA_CYL[NC_CYL_NR]	1	0...1H	0...1	1	[-]
Booleans to activate cylinder regular misfire generator mode per cylinder					
ID_RND_PAT_MIS_GEN	16*16	0...1H	0...1	1	[-]
LDP_X_ID_RND_PAT_MIS_GEN	16	0...FFH	0...255	1	[-]
LDP_Y_ID_RND_PAT_MIS_GEN	16	0...FH	0...15	1	[-]
Table of Boolean to define the pseudo random misfire sequence					

Configuration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
NLC_USE_MIS_GEN	1	0...1H	0...1	1	[-]
Misfire software generator integration					

F.2 IVVT Faulty Actuator Emulation

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
CAM_SP_IND_DFCT_IVVT_IN_i	O/V	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint for individual camshaft emulating faulty actuator; inlet					
CAM_SP_IND_DFCT_IVVT_EX_i	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint for individual camshaft emulating faulty actuator; exhaust					
LV_CAM_SP_DFCT_IVVT_IN_i	O/V	0...1H	0...1	1	-
1 = faulty actuator emulation active, 0 = inactive; inlet					
LV_CAM_SP_DFCT_IVVT_EX_i	O/V	0...1H	0...1	1	-
1 = faulty actuator emulation active, 0 = inactive; exhaust					
LV_CAM_JAM_ACT_IVVT_IN_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation in active adjustment direction active, 0 = inactive; inlet					
LV_CAM_JAM_ACT_IVVT_EX_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation in active adjustment direction active, 0 = inactive; exhaust					
LV_CAM_JAM_PAS_IVVT_IN_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation in passive adjustment direction active, 0 = inactive; inlet					
LV_CAM_JAM_PAS_IVVT_EX_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation in passive adjustment direction active, 0 = inactive; exhaust					
LV_CAM_JAM_IVVT_IN_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation at current position active, 0 = inactive; inlet					
LV_CAM_JAM_IVVT_EX_i	V	0...1H	0...1	1	-
1 = jammed actuator emulation at current position active, 0 = inactive; exhaust					
CAM_SP_MMV_DFCT_IVVT_IN_i	-	0...FFFFH	60...155.9985	1.465e-3	°CRK
PT1 filtering represents slow actuator response; inlet					
CAM_SP_MMV_DFCT_IVVT_EX_i	-	0...FFFFH	-40.125... -136.1235	1.465e-3	°CRK
PT1 filtering represents slow actuator response; exhaust					

Input data:

LV_ACT_IND_IVVT_IN_i	LV_ACT_IND_IVVT_EX_i	NC_NR_CBK_IVVT	
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i	N_32	TOIL
CAM_SP_IND_DIAG_IVVT_IN_i	CAM_SP_IND_DIAG_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX

FUNCTION DESCRIPTION:

General information:

It is possible to emulate a faulty actuator.

Description:

The following possibilities of a faulty actuator emulation are possible:

C_IDX_DFCT_IVVT_IN(_EX)_i	Description
0	No failure
1	Actuator jams at active direction adjustment (IN - ADC, EX - RTD)
2	Actuator jams at passive direction adjustment (IN - RTD, EX - ADC)
3	Actuator jams at current position
4	Actuator responses slowly

Setting C_IDX_DFCT_IVVT_IN(_EX)_i = 0 starts healing the "failure". Transitions among 1, 2 and 3 have actually less meaning. The healing or slow response is to insert between these failure emulations.

Application conditions:

Initialization:

At reset, at transition "Engine run" → "Engine stop", at deactivation:

CAM_SP_IND_DFCT_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

At transition "Engine run" → "Engine stop", at deactivation:

LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_IVVT_IN(_EX)_i = 0

Recurrence:

 # NC_NR_CBK_IVVT = 1:

360 °CRK linked task, see "IVVT Scheduler"

 # NC_NR_CBK_IVVT = 2:

At each used camshaft edge, see "IVVT Scheduler"

Activation:

LV_ACT_IND_IVVT_IN(_EX)_i = 1

Deactivation:

Not activation

Formula section:

if C_IDX_DFCT_IVVT_IN(_EX)_i = 0

then *No failure*

LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 0

CAM_SP_IND_DFCT_IVVT_IN(_EX)_i = CAM_SP_IND_DIAG_IVVT_IN(_EX)_i

LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_IVVT_IN(_EX)_i = 0

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i

elseif C_IDX_DFCT_IVVT_IN(_EX)_i = 1

then *Jammed actuator at active direction adjustment*

if LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0

then

if Inlet:

CAM_AV_IVVT_IN_i(n-1) >= C_CAM_JAM_IVVT_IN_i

and CAM_AV_IVVT_IN_i(n) < C_CAM_JAM_IVVT_IN_i

Exhaust:

```
                        CAM_AV_IVVT_EX_i(n-1) <= C_CAM_JAM_IVVT_EX_i
                   and  CAM_AV_IVVT_EX_i(n) > C_CAM_JAM_IVVT_EX_i
              then
                   LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 1
                   CAM_SP_IND_DFCT_IVVT_IN(_EX)_i =
                   C_CAM_JAM_IVVT_IN(_EX)_i
                   LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1
              endif
         endif
         LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0
         LV_CAM_JAM_IVVT_IN(_EX)_i = 0
         CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i
elseif   C_IDX_DFCT_IVVT_IN(_EX)_i = 2
then     Jammed actuator at passive direction adjustment
         if   LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0
         then
              if   Inlet:
                        CAM_AV_IVVT_IN_i(n-1) <= C_CAM_JAM_IVVT_IN_i
                   and  CAM_AV_IVVT_IN_i(n) > C_CAM_JAM_IVVT_IN_i
                   Exhaust:
                        CAM_AV_IVVT_EX_i(n-1) => C_CAM_JAM_IVVT_EX_i
                   and  CAM_AV_IVVT_EX_i(n) < C_CAM_JAM_IVVT_EX_i
              then
                   LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 1
                   CAM_SP_IND_DFCT_IVVT_IN(_EX)_i =
                   C_CAM_JAM_IVVT_IN(_EX)_i
                   LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1
              endif
         endif
         LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0
         LV_CAM_JAM_IVVT_IN(_EX)_i = 0
         CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i
elseif   C_IDX_DFCT_IVVT_IN(_EX)_i = 3
then     Jammed actuator at current position
         if   LV_CAM_JAM_IVVT_IN(_EX)_i = 0
         then
```

LV_CAM_JAM_IVVT_IN(_EX)_i = 1

CAM_SP_IND_DFCT_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i

LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1

endif

LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i = CAM_AV_IVVT_IN(_EX)_i

else *Actuator responses slowly (C_IDX_DFCT_IVVT_IN(_EX)_i = 4)*

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i(n) =

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i(n-1) +

IP_CRLC_DFCT_IVVT_IN(_EX)_i *

(CAM_SP_IND_DIAG_IVVT_IN(_EX)_i -

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i(n-1))

CAM_SP_IND_DFCT_IVVT_IN(_EX)_i =

CAM_SP_MMV_DFCT_IVVT_IN(_EX)_i(n)

LV_CAM_SP_DFCT_IVVT_IN(_EX)_i = 1

LV_CAM_JAM_ACT_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_PAS_IVVT_IN(_EX)_i = 0

LV_CAM_JAM_IVVT_IN(_EX)_i = 0

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_IDX_DFCT_IVVT_IN_i	1	0...4H	0...4	1	-
Index of IVVT actuator emulated failure; inlet					
C_IDX_DFCT_IVVT_EX_i	1	0...4H	0...4	1	-
Index of IVVT actuator emulated failure; exhaust					
C_CAM_JAM_IVVT_IN_i	1	0...FFH	60...155.625	0.375	°CRK
Jammed actuator position; inlet					
C_CAM_JAM_IVVT_EX_i	1	0...FFH	-40.125...-135.75	0.375	°CRK
Jammed actuator position; exhaust					
IP_CRLC_DFCT_IVVT_IN_i	4x4	0...FFH	0...0.99609	0.0039	-
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_2_VVTI	4	0...C8H	-40...160	1	°C
Correlation constants for slow responsing actuator emulation; inlet					
IP_CRLC_DFCT_IVVT_EX_i	4x4	0...FFH	0...0.99609	0.0039	-
LDPM_N_32_5_VVTI	4	0...FFH	0...8160	32	rpm
LDPM_TOIL_2_VVTI	4	0...C8H	-40...160	1	°C
Correlation constants for slow responsing actuator emulation; exhaust					

G Vehicle speed control

Table of Contents

Index

G.1 Cruise control (CRU)

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CRU _MAIN_SWI	V/O	0...1H	0...1	1	-
indicate CRU readiness					
CRU_SWI_DRIV_STATE	V/O	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
Switch state raw input (after NR_CRU_BAS acquisition)					
CRU_SWI_STATE_TEMP	V	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
Switch state of the cruise button					
NR_CRU_BAS	V	0...FFH	0...255	1	-
Number of raw acquisitions in the same range					
T_CRU_SWI_STATE	V	0...FFH	0...2.55	0.01	s
time since switch state is stable					
CRU_SWI_STATE	V	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
Filtered Driver Request					
REQ_MSW	V/O	0H 1H 2H 3H 4H 5H 6H 7H	Hard off Soft off TIP up TIP Down Set / Accelerate Resume Decelerate No demand	1	-
Driver demand					
STATE_CRU_CTL	V/O	0H 1H 2H 3H 5H 7H 9H	MAIN_SWITCH_OFF CONST DRIVING PASSIVE RESUME SET_ACC DEC TIP	1	-
Cruise control state					
LV_STATE_CRU_ON_OFF_SET_ENA	V	0...1H	0...1	1	-
Bit enabling to turn the CRU main switch on and off.					
T_READY_CRU	V	1...FFH	1..2550	10	ms
Timer starting with LV_IGK = 1					
LV_CRU_PAS_RES_TO_SET	V/O	0...1H	0...1	1	-
Logical variable indicating that the SET button was set out of passive or resume.					

Input data:

LV_IGK	LV_CRU_ACT	LV_CRU_OFF_IRR	LV_ST_END
VB	CRU_BAS	STATE_CRU	
LV_CRU_OVER_ACT	C_VS_MIN_CRU	VS_SP_DRIV_CRU	

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
ID_CRU_SWI_STATE	1*12	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
LDP_CRU_BAS	12	0...3FFH	0...5	4.8876 e-3	V
Switch assignment					
ID_FIL_CRU_SWI	6*6	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
LDP_CRU_SWI_STATE_TEMP	6	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
LDP_CRU_SWI_STATE	6	0H 1H 2H 3H 4H 5H	RESTING STATE ON/OFF CANCEL RES/ACC SET/COAST OUT OF RANGE	1	-
Cruise control switch filter.					
C_T_CRU_SWI_OFF	1	0...FFH	0...2.55	0.01	s
Time since switch state stable for OFF detection					
C_T_CRU_SWI_ON	1	0...FFH	0...2.55	0.01	s
Time since switch state stable for ON detection					
C_T_MIN_MSW	1	AH...FFH	0.1...2.5	0.01	s
Minimum time for ACCand DEC detection.					
C_T_READY_CRU	1	1...FFH	1..2550	10	ms
Delay time after control unit initialization					
C_NR_CRU_BAS	1	0...FFH	0...255	1	-
Number of identical measurement to validate the CRU_SWI _STATE					

FUNCTION DESCRIPTION

General information:

G.1.1 Summary

The cruise control (CRU) controls the vehicle speed by a P/I controller to the required target speed. The CRU button in the multifunction steering wheel activates the control functions, if no cut-off conditions or switch-on prohibitions are present.

The driver then can set the desired speed using the following functions:

- Set/Coast
- Resume/Acceleration
- Cancel
- OFF

G.1.1.1 Constant drive

After reaching the desired speed, the vehicle speed is kept constant independent of driving resistance changes which act on the vehicle, e.g. rising and falling gradient or headwind, if there are no abnormal termination conditions and the engine power or the braking effect is sufficient.

G.1.1.2 Setting/deceleration

By activating the SET/COAST function, the driver can store the current vehicle speed as desired speed and thus switch on the CRU system. Also the speed can be changed with a defined deceleration from the controlled constant drive. After reaching the desired speed, the constant drive function of the control is active.

G.1.1.3 Resume/acceleration

When the driver activates the RES function, the vehicle is accelerated from the current vehicle speed with a defined acceleration to the stored desired speed. The stored desired speed is the vehicle speed of the last control situation.
If the controlled constant drive function is active, the driver activates the acceleration mode by pressing the RES/ACCEL button. If the driver ends the function by releasing the button, the current vehicle speed is stored as new desired speed and the controlled constant drive function is activated.

G.1.1.4 Tip-up

Starting at controlled constant drive, the driver can increase his desired speed in steps of 1 k.p.h. by activating the TIP_UP function (press RES/ACCEL not longer than C_T_MIN_MSW). This function can be selected several times consecutively, thus the vehicle can be accelerated to a higher desired speed.

G.1.1.5 Tip-down

Starting at controlled constant drive, the driver can reduce his desired speed in steps of 1 k.p.h. by activating the TIP_DOWN function (press SET/COAST not longer than C_T_MIN_MSW). This function can be selected several times consecutively, thus the vehicle can be decelerated to a lower desired speed.

G.1.1.6 CANCEL

To switch the system off, the driver can select the CANCEL button of the multifunction steering wheel. In this way the cruise control system is switched off soft, and the desired speed remains stored for later activation by the RES function.

G.1.1.7 CRUISE ON/OFF

To switch the system on and off, the driver can select the CRUISE ON/OFF button of the multifunction steering wheel. In this way the cruise control system is switched on and off hard, and the desired speed doesn't remain stored for later activation by the RES function.
If the CRUISE ON/OFF button is detected during CONF_CRU = 0 then Cruise Control is learnt (CONF_CRU = 1), see Chapter "Variant Coding".

G.1.2 Ready state and display

A precondition for activating the CRU functions is the CRU ready state. After ECU initialization, the CRU state is initially passive. C_T_READY_CRU after ignition key on (LV_IGK = 1) the CRU state can be switched to the ready state by pressing the ON/OFF button once. To indicate CRU readiness to the driver, the signal LV_CRU_MAIN_SWI is set from 0 to 1. In this way the CRU ready lamp is activated at the instrument panel.

The transition to the passive state occurs by pressing the ON/OFF button once from the CRU ready state. The CRU ready lamp is then switched off.

G.1.3 Cruise control demand

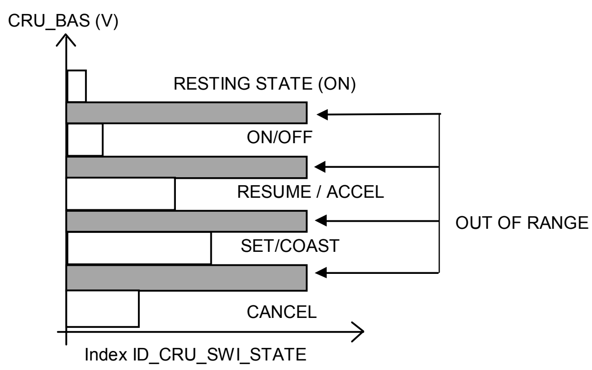
The cruise control button is measured via resistors from ignition power. The raw acquisition voltage CRU_BAS is perfomed every 10 ms.

This file is able to detect different voltage levels corresponding to the different positions of the cruise button.

If 2 buttons are depressed at the same time or if a ground problem occurs so that the ECU is not able to assign the right request versus the voltage input, then the CRU_SWI_STATE_TEMP = "OUT OF RANGE", and the cruise control is turned off.

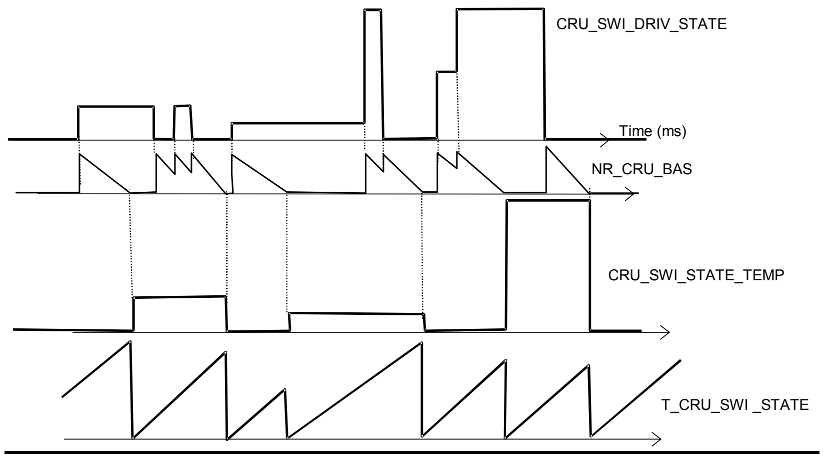
The voltage level must stay C_NR_CRU_BAS time in the same range to be valid. CRU_SWI_DRIV_STATE must be constant for the last C_NR_CRU_BAS consecutive calculations to be valid and provided as an output through CRU_SWI_STATE_TEMP.

Voltage and index relation (EXAMPLE):

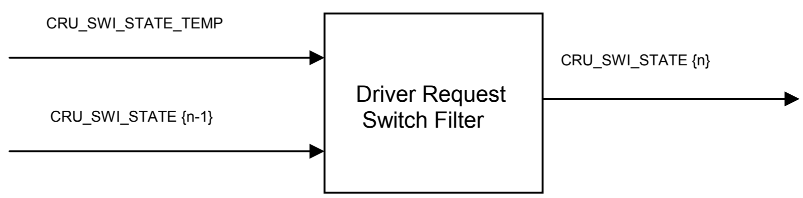


Validation of the switch level:

NR_CRU_BAS is set to C_NR_CRU_BAS at every transition of CRU_SWI_DRIV_STATE.
CRU_SWI_STATE_TEMP = CRU_SWI_DRIV_STATE once NR_CRU_BAS reach 0.
T_CRU_SWI_STATE is initialised to 0 at every transition of CRU_SWI_STATE_TEMP, and incremented while the state is stable.



G.1.4 Cruise request from driver



Simultaneously pressing two buttons (i.e. RES/ACC and SET/COAST) may not be detectable as an “Out Of Range” condition since the voltage tolerance of one button overlaps the tolerance of both pressed together.

The purpose of this module is to provide a means to overcome this issue by forcing a specific switch state (CANCEL) until an “all buttons released” condition may be verified.

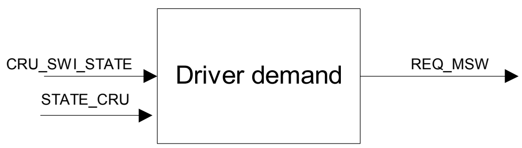
The applied switch output, CRU_SWI_STATE {n}, from table ID_FIL_CRU_SWI, depends on the previous switch output, CRU_SWI_STATE {n-1}, and the switch request from the driver, CRU_SWI_STATE_TEMP.

Example Operation:
If the Cancel button is detected, then this MUST be followed by a Rest State input (no buttons pressed) before any other input may be acknowledged (stuck button feature). If both the Cancel and RES/ACC are pressed simultaneously with the Cancel button released first; then the Cruise *Switch* State will remain in the Cancel State until a Switch Rest State is detected.

Example: The output of this table ID_FIL_CRU_SWI is **CRU_SWI_STATE {n}**

CRU_SWI_STATE {n} (Output)		**CRU_SWI_STATE_TEMP {n}**					
		REST STATE	*ON/OFF*	*CANCEL*	*RES/ACC*	*SET/ COAST*	*OUT OF RANGE*
Previous **CRU_SWI _STATE {n-1}**	*REST STATE*	Rest State	ON/OFF	Cancel	RES / ACC	SET/ COAST	OUT OF RANGE
	ON/OFF	Rest State	ON/OFF	Cancel	RES / ACC	SET/ COAST	OUT OF RANGE
	CANCEL	Rest State	ON/OFF	Cancel	Cancel	Cancel	Cancel
	ACC/ RES	Rest State	ON/OFF	Cancel	RES / ACC	SET/ COAST	OUT OF RANGE
	SET/ COAST	Rest State	ON/OFF	Cancel	RES / ACC	SET/ COAST	OUT OF RANGE
	OUT OF RANGE	Rest State	ON/OFF	Cancel	Cancel	Cancel	Cancel

G.1.5 Cruise request state



Some cruise buttons have multiple functions per position, such as RES/ACC or ON/OFF. The goal of this file is to "normalise" the different cruise button designs in order to keep a single cruise algorithm and state diagram afterward.

Example: When the driver activates the cruise switch with "RES/ACC" and the cruise control state is "PASSIVE", this means that the driver requests the "RESUME" function. If the cruise control state is "Constant driving" and the driver presses "RES/ACC", then the acceleration mode is activated.

The driver knows in which state the cruise control is currently in (ON, OFF, PASSIVE, …), therefore his demand must be calculated versus the button AND versus the actual state of the cruise control function.

Application conditions:

Activation: LV_IGK = 1

Deactivation: LV_IGK = 0

Initialisation:

At reset: CRU_SWI_STATE = `RESTING STATE'

CRU_SWI_DRIV_STATE = `RESTING STATE'

NR_CRU_BAS = 0

T_CRU_SWI_STATE = 0

REQ_MSW = 'HARD_OFF'

LV_STATE_CRU_ON_OFF_SET_ENA =1

At transition LV_IGK = 0 → 1:

LV_CRU_MAIN_SWI = 0

T_CRU_SWI_STATE = 0

T_READY_CRU = 0

In case of Error or Initialisation of NVMY:

LV_CRU_MAIN_SWI = 0

T_READY_CRU = 0

REQ_MSW = 'HARD_OFF'

Recurrence: 10ms

Formula section:

```
if          LV_IGK = 1
then        T_READY_CRU (n) = T_READY_CRU (n-1) + 10ms
    if              CRU_SWI_DRIV_STATE <> ID_CRU_SWI_STATE
    then            NR_CRU_BAS = C_NR_CRU_BAS
                    CRU_SWI_DRIV_STATE = ID_CRU_SWI_STATE
    else    if              NR_CRU_BAS > 0
            then            NR_CRU_BAS (n) = NR_CRU_BAS (n-1) -1
            endif
    endif
    if              NR_CRU_BAS = 0
    then            CRU_SWI_STATE_TEMP = CRU_SWI_DRIV_STATE
                    CRU_SWI_STATE = ID_FIL_CRU_SWI
                    At transition of CRU_SWI_STATE_TEMP
                    T_CRU_SWI_STATE = 0
    endif
    T_CRU_SWI_STATE (n) = T_CRU_SWI_STATE (n-1) + 10ms
    if              T_READY_CRU >= C_T_READY_CRU
    and             LV_CRU_OFF_IRR = 0
    then    if              LV_CRU_MAIN_SWI = 0
            then    if              CRU_SWI_STATE = 'ON/OFF'
                    and             T_CRU_SWI_STATE >= C_T_CRU_SWI_ON
                    and             LV_STATE_CRU_ON_OFF_SET_ENA = 1
                    then            LV_CRU_MAIN_SWI = 1
                                    LV_STATE_CRU_ON_OFF_SET_ENA = 0
                    endif
```

```
          else   if            CRU_SWI_STATE = 'ON/OFF'
                 and          T_CRU_SWI_STATE >= C_T_CRU_SWI_OFF
                 and          LV_STATE_CRU_ON_OFF_SET_ENA = 1
                 then         LV_CRU_MAIN_SWI = 0
                              LV_STATE_CRU_ON_OFF_SET_ENA = 0
                 endif
          endif
   else          LV_MAIN_SWI = 0
   endif
   if            LV_ST_END = 1
   then   if            LV_CRU_MAIN_SWI = 0
          then          STATE_CRU_CTL = 'MAIN_SWITCH_OFF'
          else          STATE_CRU_CTL = STATE_CRU
           endif
          If  Cruise switch aquisition is in process
                 if           STATE_CRU_CTL = 'PASSIVE'
                 or           STATE_CRU_CTL = 'RESUME'
                 and          CRU_SWI_STATE = 'SET/COAST'
                 then         LV_CRU_PAS_RES_TO_SET = 1
                 endif
                 if           CRU_SWI_STATE <> 'SET/COAST'
                 then         LV_CRU_PAS_RES_TO_SET = 1
                 endif
                 if           REQ_MSW = 'RESUME'
                 and          CRU_SWI_STATE= 'RES/ACC'
                 then         REQ_MSW = 'RESUME'
                 else         set REQ_MSW according to table below
                 endif
          endif
          if            STATE_CRU = 'PASSIVE'
          and           CRU_SWI_STATE = 'REC/ACC'
          and           VS_SP_DRIV_CRU = C_VS_MIN_CRU
          then          REQ_MSW = 'NONE'
          endif
          if            CRU_SWI_STATE = 'ON/OFF'
          and           T_CRU_SWI_STATE >= C_T_CRU_SWI_STATE
```

```
            then
                    if          LV_CRU_MAIN_SWI = 1
                    then        REQ_MSW = 'SOFT_OFF'
                    else        REQ_MSW = 'HARD_OFF'
                    endif
            endif
      else          REQ_MSW = 'HARD_OFF'
      endif
      if            CRU_SWI_STATE_TEMP = 'RESTING_STATE'
      then          LV_STATE_CRU_ON_OFF_SET_ENA = 1
      endif
else        Initialisation at Reset
endif
```

REQ_MSW (Output)		**CRU_SWI_STATE**				
		RESTING STATE	***CANCEL***	***RES/ACC***	***SET COAST***	***OUT OF RANGE***
	MAIN_SWITCH_OFF	**No Demand**	**Hard off**	**Hard off**	**Hard off**	**Hard off**
	CONST_DRIVE LV_CRU_OVER_ACT = 0	**No Demand**	**Soft off**	**Tip up*** SET/ **ACC****	**Tip down* DEC****	**Soft off**
	CONST_DRIVE LV_CRU_OVER_ACT = 1	**No Demand**	**Soft off**	**Tip up*** SET/ **ACC****	**SET/** ACC	**Soft off**
	PASSIVE	**No Demand**	**Soft off**	**Resume**	**SET/** ACC	**Soft off**
(Current)	***RESUME***	**No Demand**	**Soft off**	**Tip up**	**Tip down**	**Soft off**
STATE_ CRU_CTL	***SET_ACC***	**No Demand**	**Soft off**	**Tip up*** SET/ **ACC****	**Tip down* DEC****	**Soft off**
	DEC	**No Demand**	**Soft off**	**Tip up*** SET/ **ACC****	**Tip down* DEC****	**Soft off**
	TIP	**No Demand**	**Soft off**	**Tip up*** SET/ **ACC****	**Tip down* DEC****	**Soft off**

Note: In some cases two states are possible.

***)** if T_CRU_SWI_STATE < C_T_MIN_MSW then 'TIP UP' or 'TIP DOWN' are activated

****)** if T_CRU_SWI_STATE >= C_T_MIN_MSW then 'SET/ACC' or 'DEC' are activated

G.1.6 Functionality of the ETC Cruise Control

G.1.6.1 Terminology

CRU manipulated variable FAC_TQ_CRU

Reference input variable VS_SP_CRU

Target variable VS_SP_DRIV_CRU

Control deviation VS_DIF_CRU ~ VS_SP_CRU - VS

Reference input/target variable difference

VS_SP_DIF_CRU = VS_SP_CRU - VS_SP_DRIV_CRU

G.1.6.2 Conditions for cruise control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
REQ_CRU	V/O	0H 1H 2H 3H 4H 5H 6H 7H	Hard_off Soft_off TIP up TIP Down Set / Accelerate Resume Decelerate None	-	-
Request of cruise control functionality					
LV_CRU_OFF_IRR	V/O	0...1H	0....1	1	-
Irreversible cut off conditions for cruise control					
LV_CRU_ACT	V/O	0...1H	0...1	1	-
Intervention of cruise control					
STATE_CRU_OFF_REV	V/O/S	0H 1H 2H 4H 8H 10H 20H 40H 80H 100H 200H 400H 800H 10000H 20000H 40000H 80000H 100000H	0 Main_SW_OFF VS_CRU_LOW LV_ES Brake_ACT LV_N_MAX Runup_lock ACCEL_MAX CLUTCH_ACT Gear_INT REQ_MSW_H_off LV_VS_MAX REQ_TCU REQ_MSW_S_off High VS_DIF_CRU Long overt or tar speed VS_SP_MAX_exc EXT_TQ_INT	-	-
Status switch off condition reversible					
T_DLY_CRU	V	0...FFFFH	0...655.35	0.1	s
Time counter overtaking					
LV_CRU_OVER_ACT	V	0...1H	0...1	1	-
Overtaking function active					
T_DLY_DOWN_SHIFT_REQ_RELEASE	V	0...FFH	0...255	1	s
Time counter to release down shift request to TCU					
LV_REQ_DOWN_SHIFT	V/O	0...1H	0...1	-	-
Down shift requested to TCU					

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
STATE_CRU_OFF_REV_SAVE	V	0H 1H 2H 4H 8H 10H 20H 40H 80H 100H 200H 400H 800H 10000H 20000H 40000H 80000H 100000H	0 Main_SW_OFF VS_CRU_LOW LV_ES Brake_ACT LV_N_MAX Runup_lock ACCEL_MAX CLUTCH_ACT Gear_INT REQ_MSW_H_off LV_VS_MAX REQ_TCU REQ_MSW_S_off High VS_DIF_CRU Long overt or tar speed VS_SP_MAX_exc EXT_TQ_INT	-	-
Status switch off condition reversible saved					
STATE_CRU_OFF_IRR	V/O/S	0H 1H 2H 4H 8H 10H 20H 40H 80H	0 ERR_MTC ERR_TPS ERR_BLS_BTS_or_CS ERR_VS ERR_L2_MON ERR_CAN_BUS_OFF ERR_PVS ERR_CRU_SWI	-	-
Status switch off condition irreversible					
REQ_DOWN_SHIFT	V	0H 1H 2H 3H 4H	RELE ACCEL RESUME UPHILL DOWNHILL	-	-
Down shift request status					
LV_MSW_IDLE	V/O	0...1H	0...1	-	-
“None” request from MSW					
LV_AC_RELE_TO_ECU	V	0...1H	0...1	-	-
autocruise release request					

Input data:

LV_ERR_VS	LV_ERR_BLS_BTS	LV_N_LIM_REQ_MON	LV_ERR_TQI_N_MAX_MON
LV_VS_MAX	LV_IM_CS_PN	LV_CRU_INH_MON	LV_ERR_CAN_BUS_OFF
REQ_MSW	LV_CRU__MAIN_SWI	LV_CRU_OVER_ACT_ACK	IP_FAC_TQ_REQ_CLU_IS
VS_CRU	STATE_CRU_CTL	VS_SP_DRIV_CRU	LV_CRU_OFF_BY_ASR_MSR_CTL
LV_N_MAX	C_VS_SP_CRU_MAX	FAC_TQ_REQ_DRIV	FAC_TQ_REQ_CRU
GR_AT	VS_DIF_CRU	LV_AT	AC_CRU
LV_ES	LV_ERR_PVS	VS_SP_CRU	G_SEL_DISP
GR_MT	LV_ERR_CS	LV_CS_ACT_DC	LV_BLS_BTS_ACT_DC

FUNCTION DESCRIPTION:

The State of cruise control is the evaluation of the request and the cruise conditions.
If one of the following conditions is detected, no controller can be activated or an activated controller function is switched off. In this case a distinction is made between hard (FAC_TQ_CRU = 0) and soft cut-off (FAC_TQ_CRU through ramp function to zero).

Application conditions:

Recurrence: 10 ms

Initialisation: At Reset, NVMY_CLR and LV_IGK from 0 to 1:
set LV_CRU_OFF_IRR = 0 and STATE_CRU_OFF_REV_SAVE = 0.

Formula section:

If req_msw = NONE
Then LV_MSW_IDLE = 1
If hard_off or soft_off is requested
Then LV_MSW_IDLE = 0
Endif

if LV_AT = 1
and GR_AT = 0 " *(gear in position N,P)"*
or GR_AT = 7 *"(gear in position R)"*
or [G_SEL_DISP = 08h *"(Sports mode)"*
and GR_AT <= C_TAR_GC_AC_RELE]
then LV_AC_RELE_TO_ECU = 1
else LV_AC_RELE_TO_ECU = 0
endif

G.1.6.2.1 Conditions for hard cut-off

The control functions of the cruise control can be activated only if:

- CRU ready state present LV_CRU_MAIN_SWI = 1

Irreversible cut-off due to diagnosis failures

With the following diagnosis errors the CRU is irreversible hard cut off for this engine run. The CRU-readiness and the CRU readiness light are reseted. The CRU function can be activated by LV_IGK=0/1 change, as long as no running diagnosis errors are present.

Formula section:

IF

1.	Error in MTC:	LV_ERR_MTC_CTL or LV_ERR_MTC_DR = 1	**or** *(1H)*
2.	Error TPS Limp Home:	LV_ERR_TPS = 1	**or** *(2H)*
3.	Error in brake light/test or clutch switch system	LV_ERR_BLS_BTS = 1 LV_ERR_CS = 1	**or** *(4H)* **or**
4.	Error in vehicle speed signal	LV_ERR_VS = 1, (sensor error)	**or** *(8H)*
5.	Error Level 2 Monitoring	LV_ERR_TQI_N_MAX_MON =1 LV_CRU_INH_MON =1 LV_N_LIM_REQ_MON =1	**or** *(10H)* **or** **or**

6. CAN Bus off: LV_ERR_CAN_BUS_OFF = 1 **or** *(20H)*
7. PVS-Error: LV_ERR_PVS = 1 **or** *(40H)*
8. Cruise switch error LV_ERR_CRU_SWI_1 = 1 **or** *(80H)*
   LV_ERR_CRU_SWI_2 = 1 **or**
   LV_ERR_CRU_SWI_3 = 1

Then LV_CRU_OFF_IRR = 1
REQ_CRU = Hard off (The cru readiness lamp is switched off.)

Else LV_CRU_OFF_IRR = 0
STATE_CRU_OFF_IRR = 00H
STATE_CRU_OFF_REV = 00H

Byte							
7	6	5	4	3	2	1	0
8.	7.	6.	5.	4.	3.	2.	1.
Error CRU SW	Error PVS	CAN Buss off	Monitoring level 2	Error VS	Brake SW **or** Clutch SW error	Error TPS	Error MTC

(Table: STATE_CRU_OFF_IRR)

Reversible cut-off:

If one of the following conditions is current in active CRU function, CRU is transferred by hard cut-off (FAC_TQ_CRU = 0) into the ready state without controlled drive:

IF

1. Main Switch off: LV_CRU_MAIN_SWI = 0 **or** (1H)
2. VS_CRU low: VS_CRU < C_VS_MIN_PLAUS_CRU for longer than C_T_DLY_CRU_OFF_REV_VS_FAULT **or** (2H)
3. Engine stopped: LV_ES = 1 **or** (4H)
4. Brake switch: LV_BLS = 1 **or** (8H)
   or LV_BTS = 1
   or if LV_AT = 1
   and LV_BLS_BTS_ACT_DC = 0
   or if LV_AT = 0
   and LC_INH_BLS_BTS_ACT_MT = 0
   and LV_BLS_BTS_ACT_DC = 0

   For MT-vehicles it is not mandatory to push brake pedal during vehicle start or gear select , so BLS/BTS switch acting might still not be detected at Cruise request Therefore this condition can be disabled via Application switch

5. Engine speed limitation be active: LV_N_MAX = 1 **or** (10H)

6. Run up lock: (| (N_VS_CRU$_n$ - N_VS_CRU$_{n-1}$) / N_VS_CRU$_{n-1}$ | * 100 > C_N_VS_MAX_CRU and LV_AT = 0) **or** (20H)

7. Exceeding a maximum acceleration: | AC_CRU | > C_AC_MAX_CRU for longer than C_T_DLY_CRU_OFF_REV_VS_FAULT **or** (40H) (not during active overtaking function)

8. Detection of clutch actuation with manual transmissions:
   if LV_AT = 0
   and (LV_IM_CS_PN =0 **or** (80H)
   or LV_CS_ACT_DC = 0)

9. Neutral Gear (A/T and M/T): (LV_AT = 0 and (GR_MT = 1 or GR_MT = 0))
   or (LV_AT = 1 and GR_AT = 0)
   for longer than C_T_DLY_CRU_OFF_REV_VS_FAULT
   or (100H)

10. REQ_MSW = HARD_OFF **or** (200H)

11. Overtaking function is active and vehicle speed limitation is activated **or** (400H)
    LV_VS_MAX = 1

12. Cruise release/disable request by TCU:
    LV_AC_RELE_TO_ECU = 1 **or** (800H)

Then REQ_CRU = Hard off

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_MIN_CRU	1	0...FFFFH	0...511.9922	0.0078	km.p.h.
Vehicle speed condition for VS_CRU					
C_VS_MIN_PLAUS_CRU	1	0...FFFFH	0...511.9922	0.0078	km.p.h.
Vehicle speed condition for VS_CRU					
C_N_VS_MAX_CRU	1	0...FFH	0...99.84	0.39	%
Threshold value for hard cut-off					
C_AC_MAX_CRU	1	0...7FFFH	0...15,72	0.00048	m/s^2
Max. acceleration					
C_TAR_GC_AC_RELE	1	0...FFH	0...255	1	-
Gear position to disable cruise control					
LC_INH_BLS_BTS_ACT_MT	1	0...1H	0.1	1	-
Switch to inhibit BLS_BTS acting detection for MT vehicles					

G.1.6.2.2 Conditions for soft cut-off

General information:

If one of the following conditions is detected, the controller function is switched off. In this FAC_TQ_CRU is set through ramp function to zero.

Formula section:

If

13. Detection of the REQ_MSW = SOFT_OFF function of the cruise control (10000H)

or

14. [if LV_CRU_OVER_ACT = 0 (20000H)

and {(Exceeding a permissible VS_DIF_CRU > speed deviation C_VS_SP_DIF_MAX_CRU

or Dropping below a permissible VS_DIF_CRU < speed deviation C_VS_SP_DIF_MIN_CRU)]

for longer than C_T_DLY_CRU_OFF_REV_VS_FAULT}

or

15. [if LV_CRU_OVER_ACT = 1 (40000H)

and (the overtaking time T_DLY_CRU >= C_T_DLY_MAX_CRU

or the target speed is exceeded:
(VS_SP_DRIV_CRU - VS_CRU >= C_VS_DIF_DLY_CRU_POS)

or (VS_SP_DRIV_CRU - VS_CRU < C_VS_DIF_DLY_CRU_NEG)]

or

16. VS_CRU >= C_VS_SP_CRU_MAX for longer than C_T_DLY_VS_SP_CRU_MAX (80000H)
If this condition is interupted then the time-counter is reset to 0

or

17. ASR- or MSR-control for minimum time active (100000H)
(LV_CRU_OFF_BY_ASR_MSR_CTL = 1)

Then REQ_CRU = Soft off

Low Word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				12.	11.	10.	9.	8.	7.	6.	5.	4.	3.	2.	1.
				Disable/release request from TCU	Overtaking and VS_MAX active	MSW request HARD_OFF	Gear intervention	Clutch activ	Acceleration monitoring	Runup lock	Engine speed limitation	Brake activ	Engine stop	VS_CRU to low	Cruise Main Switch off

(Table: STATE_CRU_OFF_REV, low word)

High Word															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
											17.	16.	15.	14.	13.
											External TQ intervention	VS_SP_MAX to long	Overtaking Funct. To long or to high VS_DIF	Too high VS deviation	MSW request soft-off

(Table: STATE_CRU_OFF_REV, high word)

STATE_CRU_OFF_REV_SAVE:

if LV_CRU_ACT n-1 = 1
and REQ_CRU = 'hard off' or 'soft' off
then STATE_CRU_OFF_REV_SAVE = STATE_CRU_OFF_REV

G.1.6.2.3 Overtaking function

Description:

If the driver torque request FAC_TQ_REQ_DRIV > FAC_TQ_REQ_CRU and ≥ IP_FAC_TQ_REQ_CLU_IS__N_32 , then the controller detects overtaking and the value VS_DIF_I_CRU remains stored as threshold value. The time for the overtaking function T_DLY_CRU is started. The adjustable time C_T_DLY_MAX_CRU states whether start controlled constant drive is possible after ending the overtaking function.
During the overtaking function, 'Overdriving' FAC_TQ_REQ_DRIV by acceleration (ACCEL) and Tip-up functions is possible.
If the time C_T_DLY_MAX_CRU or the max. deviation from the stored target speed is exceeded, no automatic start of controlled constant drive is possible any more. The CRU mode is ended. The functions of the cruise control can be activated again exclusively by the multifunction steering wheel. At resume, after overtaking function is finished, the resume function is activated with the initialisation calculation of VS_DIF_I_CRU.
During the overtaking function the calculation of FAC_TQ_CRU is calculated normal with freezed I - share and calculated P-Share.

Application conditions:

Initialisation: LV_CRU_OVER_ACT = 0

Formula section:

If transition FAC_TQ_REQ_DRIV > FAC_TQ_REQ_CRU
and FAC_TQ_REQ_DRIV > IP_FAC_TQ_REQ_CLU_IS__N_32
and STATE_CRU = "constant drive" (vehicle in cruise control)
Then $T\_DLY\_CRU_{n} = T\_DLY\_CRU_{n-1} + 10$ ms
LV_CRU_OVER_ACT = 1

"VS_DIF_I_CRU stored see calculation of I-Share"

if LV_CRU_OVER_ACT = 1
and T_DLY_CRU < C_T_DLY_MAX_CRU
and VS_SP_DRIV_CRU - VS_CRU < C_VS_DIF_DLY_CRU_NEG
and VS_SP_DRIV_CRU - VS_CRU > C_VS_DIF_DLY_CRU_POS
then **if** (transition FAC_TQ_REQ_DRIV <
FAC_TQ_REQ_CRU -C_FAC_TQ_REQ_HYS_CRU
or FAC_TQ_REQ_DRIV <
IP_FAC_TQ_REQ_CLU_IS - C_FAC_TQ_REQ_CLU_IS_HYS)

and VS_DIF_CRU > C_VS_DIF_MIN_CRU_OVER

then "RESUME" with initialisation of VS_DIF_I_CRU

LV_CRU_OVER_ACT = 0

else LV_CRU_OVER_ACT = 1

else Cruise control passive soft cut off, LV_CRU_OVER_ACT = 0

IF LV_CRU_OVER_ACT =1

and (REQ_MSW = =TIP_UP or ACC/SET)

Then **IF** LV_CRU_OVER_ACT_ACK = 0

Then REQ_CRU = "TIP_UP"

else LV_CRU_OVER_ACT = 0

REQ_CRU = REQ_MSW

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_TQ_REQ_HYS_CRU	1	0...FFFFH	0...1.999969	3.052E-5	-
FAC_TQ_REQ hysteresis for start of the restart function					
C_T_DLY_MAX_CRU	1	0...FFFFH	0...655.35	0.01	s
Delay threshold for start of the restart function					
C_T_DLY_CRU_OFF_REV_VS_FAULT	1	0...FFFFH	0...655.35	0.01	s
Delay threshold for reversible cut off at short VS signal fault					
C_VS_DIF_DLY_CRU_NEG	1	0...7FFFH	0...255.9922	0.0078	km.p.h.
Max. difference from VS_SP_DRIV_CRU and C_VS_DIF_DLY_CRU					
C_VS_DIF_DLY_CRU_POS	1	8000...0H	-255.9922...0	0.0078	km.p.h.
Max. difference from VS_SP_DRIV_CRU and C_VS_DIF_DLY_CRU					
C_VS_DIF_MIN_CRU_OVER	1	8000H...7FFFH	-256...255.9922	0.0078	Km.p.h
VS_DIF_CRU min value for finish overtaking function					

G.1.6.2.4 Condition for cruise control intervention

Description:

The cruise control intervention is active if the vehicle is in operation depending of FAC_TQ_CRU. The Status is needed in the Cruise control monitoring of ETC Safety Unit.

Formula section:

IF - STATE_CRU = "Passive" **and**
- VS_CRU >= C_VS_MIN_CRU **and**
- REQ_CRU = not "Soft off" **and**
- REQ_CRU = not "Hard off" **and**
- **{**REQ_MSW = TIP_UP **or** SET/ACC **or**
[REQ_MSW = RES **and**
Triggering the RES function for plausible VS_SP_DRIV_CRU
(C_VS_MIN_CRU <= VS_SP_DRIV_CRU <= C_VS_SP_CRU_MAX)] **}**

Then LV_CRU_ACT = 1

LV_CRU_ACT is reseted to 0 if an off condition is present.

IF LV_CRU_ACT = 1 **and** REQ_CRU = "Soft off" or "Hard off"

Then LV_CRU_ACT = 0 after FAC_TQ_REQ_CRU = 0 (ramp down is finshed)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_SP_DIF_MAX_CRU	1	8000...7FFFH	-256...255.9922	0.0078	km.p.h.
Max. vehicle speed deviation					
C_VS_SP_DIF_MIN_CRU	1	8000...7FFFH	-256...255.9922	0.0078	km.p.h.
Min. vehicle speed deviation					
C_T_DLY_VS_SP_CRU_MAX	1	0...FFH	0...255	1	s
Max .time for VS more than VS max at CRU					

G.1.6.2.5 Downshift request to TCU

Application conditions:

Activation: LV_AT = 1 and Gear = 5 or 4 (for 5 and 4 speed gearbox), LV_CRU_ACT=1, LV_CRU_OVER_ACT=0

Deactivation: LV_CRU_ACT=0 ==> LV_REQ_DOWN_SHIFT=0, REQ_DOWN_SHIFT= RELE

No automatic deactivation if GEAR is not equal to 5 or 4!!

Initialisation: LV_IGK 0 –> 1, LV_REQ_DOWN_SHIFT=0

Recurrence: 10 ms

Formula section:

Downshift request: **IF** REQ_DOWN_SHIFT = RELE
THEN LV_REQ_DOWN_SHIFT = 0
ELSE LV_REQ_DOWN_SHIFT = 1

General downshift request release:

Downshift request release at not cruise controlled driving or if any cruise control button has been pressed:

IF LV_CRU_ACT = 0 **or**
CRU_SWI_STATE from RESTING_STATE to not RESTING_STATE
THEN REQ_DOWN_SHIFT = RELE
ELSE -

For ACCEL mode

Downshift request:

IF {REQ_DOWN_SHIFT = RELE **and** (GR_AT = 5 or GR_AT = 4) **and**
STATE_CRU_CTL = SET_ACC **and**
CRU_SWI_STATE = RES/ACC **and**
AC_CRU <= C_AC_CRU_TAR_ACCEL more than C_T_AC_CRU_TAR_ACCEL
THEN REQ_DOWN_SHIFT = ACCEL
ELSE -

Downshift request release:

IF REQ_DOWN_SHIFT = ACCEL **and** STATE_CRU_CTL ≠ SET_ACC **and**
THEN (REQ_DOWN_SHIFT = RELE after C_T_DLY_DOWN_SHIFT_RELE_ACCEL)
ELSE -

For RESUME mode

Downshift request:

IF {REQ_DOWN_SHIFT = RELE **and** (GR_AT = 5 or GR_AT = 4) **and**
STATE_CRU_CTL = RESUME **and**
VS_SP_DRIV_CRU – VS_CRU >= C_VS_DIF_DOWN_SHIFT_RES **and**
AC_CRU <= C_AC_CRU_TAR_RES} for more than C_T_AC_CRU_TAR_RES
THEN REQ_DOWN_SHIFT = RESUME
ELSE -

Downshift request release:

IF REQ_DOWN_SHIFT = RESUME **and**
VS_SP_DRIV_CRU – VS_CRU <= C_VS_DIF_DOWN_SHIFT_RELE_RES
THEN REQ_DOWN_SHIFT = RELE
Else -

For Up-Hill case

Downshift request:

IF REQ_DOWN_SHIFT = RELE **and** (GR_AT = 5 or GR_AT = 4) **and**
STATE_CRU_CTL = CONST_DRIVE **and**
VS_CRU <= C_VS_MAX_DOWN SHIFT_UPDOWNHILL **and**
VS_SP_DRIV_CRU – VS_CRU >= C_VS_DIF_DOWN_SHIFT_UPHILL
THEN REQ_DOWN_SHIFT = UPHILL

Downshift request release:

IF {REQ_DOWN_SHIFT = UPHILL **and** VS_SP_DRIV_CRU – VS_CRU <= C_VS_DIF_DOWN_SHIFT_RELE_UPHILL}
for more than C_T_DLY_DOWN_SHIFT_RELE_UPDOWN
THEN REQ_DOWN_SHIFT = RELE
ELSE -

For Down-Hill case

Downshift request:

IF REQ_DOWN_SHIFT = RELE **and** (GR_AT = 5 or GR_AT = 4) **and**
STATE_CRU_CTL = CONST_DRIVE **and**
VS_CRU <= C_VS_MAX_DOWN SHIFT_UPDOWNHILL **and**
VS_CRU – VS_SP_DRIV_CRU >= C_VS_DIF_DOWN_SHIFT_DOWNHILL
THEN REQ_DOWN_SHIFT = DOWNHILL
ELSE -

Downshift request release:

IF {REQ_DOWN_SHIFT = DOWNHILL **and** VS_CRU - VS_SP_DRIV_CRU <= C_VS_DIF_DOWN_SHIFT_RELE_DOWNHI}
for more than C_T_DLY_DOWN_SHIFT_RELE_UPDOWN
THEN REQ_DOWN_SHIFT = RELE
ELSE -

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_DIF_DOWN_SHIFT_RES	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request in Cruise Res_Mode					
C_VS_DIF_DOWN_SHIFT_RELE_RES	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request release in Cruise Res_Mode					
C_VS_DIF_DOWN_SHIFT_UPHILL	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request in Cruise UP-Hill case					
C_VS_DIF_DOWN_SHIFT_RELE_UPHILL	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request release in Cruise UP-Hill case					
C_T_DLY_DOWN_SHIFT_RELE_UPDOWN	1	0...FFH	0...255	1	s
Delay time threshold to release down shift request to TCU in Cruise UP- and DOWN-Hill case					
C_VS_DIF_DOWN_SHIFT_DOWNHILL	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request in Cruise DOWN-Hill case					
C_VS_DIF_DOWN_SHIFT_RELE_DOWNHI	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed deviation for down shift request release in Cruise DOWN-Hill case					
C_VS_MAX_DOWN SHIFT_UPDOWNHILL	1	0...7FFFH	0...255.9922	0.0078	Km.p.h.
Vehicle speed threshold to inhibit down shift request to TCU during Up-Down hill driving					
C_T_AC_CRU_TAR_RES	1	0 ... FFH	0 ... 2.55	0.01	s
Acceleration monitoring time during RESUME					
C_AC_CRU_TAR_RES	1	8000 ... 7FFFH	-15.72 ... 15.72	0.00048	m/s²
Target acceleration during RESUME					
C_T_DLY_DOWN_SHIFT_RELE_ACCEL	1	0 ... FFH	0 ... 255	1	s
Delay time threshold to release down shift request to TCU after ACCEL mode					
C_T_AC_CRU_TAR_ACCEL	1	0 ... FFH	0 ... 2.55	0.01	s
Acceleration monitoring time during ACCEL					
C_AC_CRU_TAR_ACCEL	1	8000 ... 7FFFH	-15.72 ... 15.72	0.00048	m/s²
Target acceleration during ACCEL					

G.1.7 Control basic function

G.1.7.1 General calculation of cruise control and constant drive

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_DIF_CRU	O/V	FFFF8000...7FFFH	-256...255.9922	0.0078	km.p.h.
Control deviation					
VS_DIF_I_CRU	V	FFFF8000...7FFFH	-256...255.9922	0.0078	km.p.h.
Sum of the control deviations					
FAC_TQ_CRU	O/V	0...FFFFH	0...99,998	0.0015	%
CRU control variable					
FAC_TQ_CRU_INI	V	0...FFFFH	0...99,998	0.0015	%
CRU control variable initialisation value					
STATE_CRU	V	 1H 2H 3H 5H 7H 9H	 CONST_DRIVE PASSIVE RESUME SET_ACC RETARD TIP	-	-
Status of current cruise control functionality					
STATE_I_CRU	V	0H 1H 2H 3H	NO LIM FREEZE NORM	1	-
Condition of I Share VS_DIF_I_CRU					

Input data:

VS_CRU	VS_SP_CRU	AC_CRU	FAC_TQ_REQ_DRIV_MMV
N_32	REQ_CRU	IP_FAC_TQ_REQ_CLU_IS	C_FAC_TQ_REQ_CLU_IS_HYS
LV_AT	LV_RNG_L_REQ	VS_SP_DRIV_CRU	LV_CRU_OVER_ACT

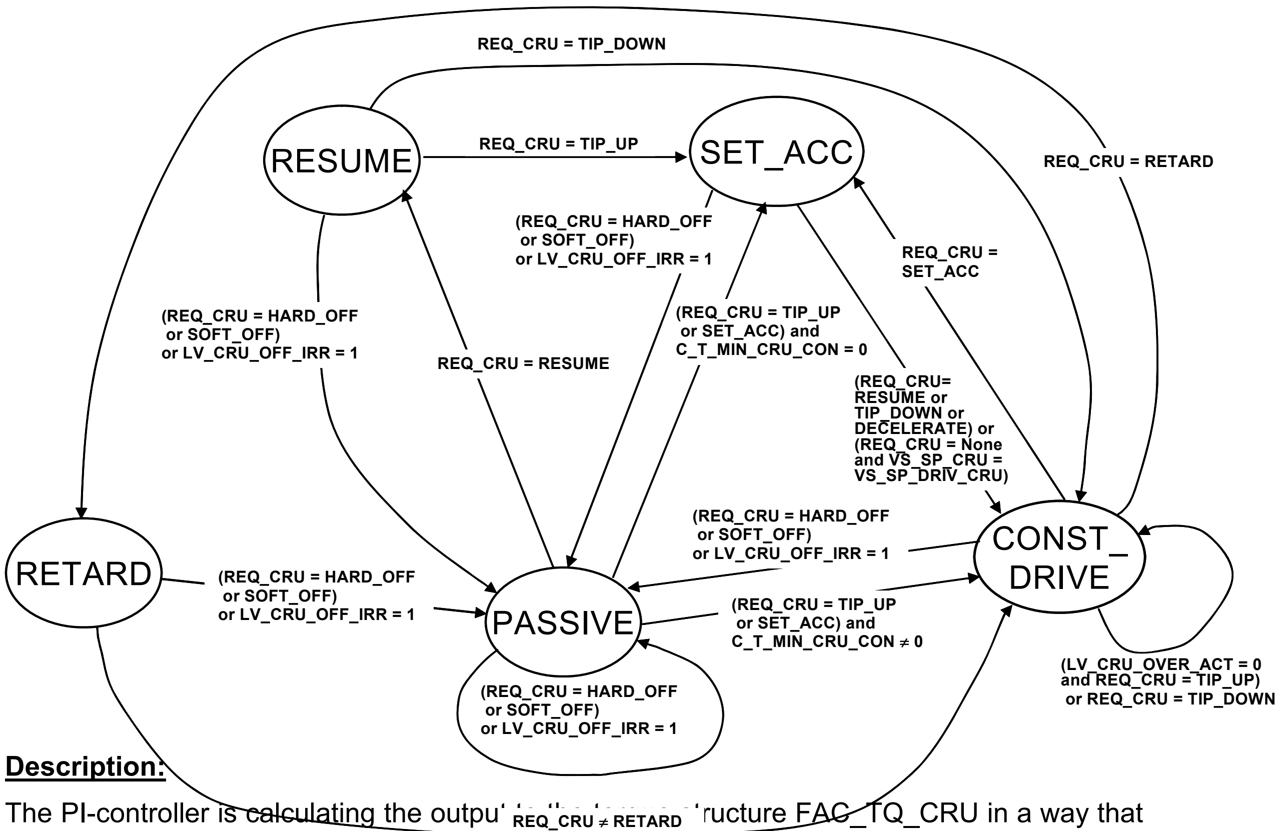
Application conditions:

Recurrence: 100 ms

Initialisation: FAC_TQ_CRU = FAC_TQ_CRU_INI based on setting conditions
STATE_CRU = PASSIVE

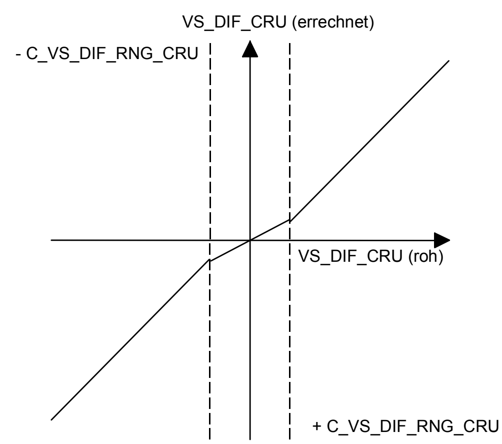
FUNCTION DESCRIPTION:Definition of the STATE_CRU:

The diagram below shows the state machine of the STATE_CRU.



Description:

The PI-controller is calculating the output to the torque structure FAC_TQ_CRU in a way that the current vehicle speed VS_CRU is controlled to reached and keep the vehicle speed setpoint VS_SP_CRU.



Formula section:

Basically calculation for all cruise states:

Total formula PI-controller:

FAC_TQ_CRU = (VS_DIF_CRU + VS_DIF_I_CRU) * IP_FAC_PROP_CRU

Conditions for change limitations of FAC_TQ_CRU:

$|FAC\_TQ\_CRU_{n-1} - FAC\_TQ\_CRU_n| \leq C\_ FAC\_TQ\_CRU\_LGRD$

At driving with an active cruise control the torque request FAC_TQ_CRU is limited to the maximum value C_FAC_TQ_CRU_MAX.

FAC_TQ_CRU <= C_ FAC_TQ_CRU_MAX

The raw value of the control deviation from the reference input variable VS_SP_CRU and the control variable VS_CRU is calculated as

VS_DIF_CRU (raw)= VS_SP_CRU - VS_CRU

To achieve a smooth control in the area of the target vehicle speed, the internal controller deviation VS_DIF_CRU can be reduced in the adjustable bandwidth | C_VS_DIF_RNG_CRU | by an applicable factor C_FAC_VS_DIF_RNG_CRU.

IF(1) |VS_SP_CRU - VS_CRU | ≤ C_VS_DIF_RNG_CRU

Then(1) VS_DIF_CRU = (VS_SP_CRU - VS_CRU) * C_FAC_VS_DIF_RNG_CRU

Else(1) If(2) (VS_SP_CRU - VS_CRU) > C_VS_DIF_RNG_CRU

Then(2) VS_DIF_CRU = (VS_SP_CRU - VS_CRU)
- C_VS_DIF_RNG_CRU
+ C_VS_DIF_RNG_CRU * C_FAC_VS_DIF_RNG_CRU.

Else(2) VS_DIF_CRU = (VS_SP_CRU - VS_CRU)
+ C_VS_DIF_RNG_CRU
- C_VS_DIF_RNG_CRU * C_FAC_VS_DIF_RNG_CRU.

Endif

I share calculation:

General calculation ***STATE_I_CRU = NORM***:

$$VS\_DIF\_I\_CRU_n = VS\_DIF\_I\_CRU_{n-1} + VS\_DIF\_CRU * IP\_FAC\_VS\_DIF\_I\_CRU$$

If the FAC_TQ_CRU on limitation (max, min value or freeze at PUC) then the **I** share is freezed **STATE_I_CRU = LIM** until a normal calculation of FAC_TQ_CRU is possible again. At or LV_CRU_OVER_ACT = 1 the **I** share is freezed **STATE_I_CRU = LIM,** but at the calculation of FAC_TQ_CRU is no interupt.

Setting at the transition “passive” to "setting" or "resume":

VS_DIF_I_CRU = FAC_TQ_CRU_INI / IP_FAC_PROP_CRU

*Calculation during freeze of I- share at SET/ACC, RES (**STATE_I_CRU= FREEZE**):*

$VS\_SP\_CRU\_STAT = VS\_SP\_CRU_n - VS\_SP\_CRU_{START}$

$VS\_DIF\_I\_CRU_{END} = VS\_DIF\_I\_CRU_{START} + IP\_FAC\_VS\_SP\_DIF\_I\_CRU * VS\_SP\_CRU\_STAT$

$VS\_DIF\_I\_CRU_{End}$ is calulated during the freeze of **I** share

FAC_TQ_CRU calculation during freeze of I- share at SET/ACC, RES, min/max limitation:

$FAC\_TQ\_CRU = (VS\_DIF\_CRU + VS\_DIF\_I\_CRU_{END}) * IP\_FAC\_PROP\_CRU$

On setting in PU/PUC or with very small FAC_TQ_REQ_DRIV (driver gets off the pedal before setting Cruise Control), a very small **I**-part initialisation would be the results which leads to an undershooting below the required target speed. To prevent this, the **I**-part initialisation value is calculated depending on AC_CRU, to get a better initialisation value.

FAC_TQ_CRU initialisation value:

IF LV_AT = 0

Then FAC_TQ_CRU_INI = FAC_TQ_REQ_DRIV_MMV * 99,998 % +
IP_FAC_TQ_CRU_INI_ADD_MT * AC_CRU

Else FAC_TQ_CRU_INI = FAC_TQ_REQ_DRIV_MMV * 99,998 % +
IP_FAC_TQ_CRU_INI_ADD_AT * AC_CRU

endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_PROP_CRU	5x4	01...FFFFH	0.00076... 49,999	0.00076	%/km.p.h.
LDPM_VS_SP_CRU__1	5	0...FFFFH	0...511.9922	0.0078	km.p.h.
LDP_VS_DIF_CRU__FAC_PROP_CRU	4	0...FFFFH	-256...255.9922	0.0078	km.p.h.
Proportional coefficient of the PI controller					
IP_FAC_VS_DIF_I_CRU	5x4	0...FFFFH	0...0.9999	0.15e-6	-
LDPM_VS_SP_CRU__1	5	0...FFFFH	0...511.9922	0.0078	km.p.h.
LDP_VS_DIF_CRU__FAC_PROP_CRU	4	0...FFFFH	-256...255.9922	0.0078	km.p.h.
Proportional coefficient of the PI controller					
IP_FAC_TQ_CRU_INI_ADD_AT/MT	6x12	0...FFFFH	-200...199,993	0,0061	%/(m/s²)
LDP_AC_CRU__FAC_TQ_CRU	6	0...FFFFH	-15,72...15,72	0,00048	%
LDP_GEAR__FAC_TQ_CRU	12	0...FFH	0...255	1	-
Additive torque based on gear and acceleration					
C_FAC_TQ_CRU_LGRD	1	0...FFFFH	0...99,998	0.0015	%/100ms
Control variable limitation gradient					
C_FAC_VS_DIF_RNG_CRU	1	0...FFH	0...0.996	0.0039	-
Reduction factor for the control deviation VS_DIF_CRU					
C_VS_DIF_RNG_CRU	1	0...FFFFH	0...511.9922	0.0078	km.p.h.
Bandwidth for reducing the control deviation					
IP_FAC_VS_SP_DIF_I_CRU	1x5	0...FFFFH	0...2	30e-6	-
LDPM_VS_SP_CRU__1	5	0...FFFFH	0...511.9922	0.0078	km.p.h.
Delta VS_SP_CRU correction					
C_ FAC_TQ_CRU_MAX	1	0...FFFFH	0...99,998	0.0015	%/10ms
Maximum request of Cruise Control					

G.1.7.1.1 Controlled constant drive (CONST_DRIVE)

For controlled constant drive, the reference input variable VS_SP_CRU corresponds to the last target speed reached, i.e. VS_SP_DRIV_CRU. No control of the reference input variable VS_SP_CRU is performed in this range.

The PI-controller is calculating the output to the torque structure FAC_TQ_CRU in a way that the current vehicle speed VS_CRU is controlled to keep the vehicle speed setpoint VS_SP_CRU. This is done until a cut-off condition arises or a condition for cruise control is infringed.

To be able to trigger trailing throttle fuel cut-off appropriately or not to trigger trailing throttle fuel cut-off, the value FAC_TQ_CRU is initially limited downwards to the value ((IP_FAC_TQ_REQ_CLU_IS__N_32 - C_FAC_TQ_REQ_CLU_CRU_IS_HYS)*99,998%). Thus the STATE_ENG “IS” is prevented. Only if the control deviation VS_DIF_CRU drops below C_VS_DIF_MIN_IS_CRU, the FAC_TQ_CRU set to zero. However, this zero setting will not take place in the first C_T_MIN_PUC_CRU seconds after entry into controlled constant drive. The trailing throttle fuel cut-off is deactivated when the control deviation exceeds the value C_VS_DIF_MAX_IS_CRU.

The calculation of the I-part is interrupted, if the torque request FAC_TQ_CRU is limited upwards or FAC_TQ_CRU ≤ ((IP_FAC_TQ_REQ_CLU_IS__N_32-C_FAC_TQ_REQ_CLU_IS_HYS)*99,998%).

Formula section:

FAC_TQ_CRU conditions:

IF T_CONST_DRIVE < C_T_MIN_PUC_CRU

Then FAC_TQ_CRU > ((IP_FAC_TQ_REQ_CLU_IS__N_32-
C_FAC_TQ_REQ_CLU_IS_CRU_HYS) * 99,998%).

ElseIF VS_DIF_CRU < C_VS_DIF_MIN_IS_CRU

Then FAC_TQ_CRU = 0

Else **IF** VS_DIF_CRU > C_VS_DIF_MAX_IS_CRU

Then FAC_TQ_CRU > ((IP_FAC_TQ_REQ_CLU_IS__N_32-
C_FAC_TQ_REQ_CLU_IS_CRU_HYS) * 99,998%).

Else calculation FAC_TQ_CRU $_{n}$= calculation FAC_TQ_CRU $_{n-1}$

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_DIF_MIN_IS_CRU	1	FFFF8000 ... 7FFFH	-256...255.9922	0.0078	km.p.h.
Threshold of the control deviation for zero setting of FAC_TQ_CRU					
C_VS_DIF_MAX_IS_CRU	1	FFFF8000 ... 7FFFH	-256...255.9922	0.0078	km.p.h.
Threshold of the control deviation for PUC end					
C_T_MIN_PUC_CRU	1	0...FFH	0...25.5	0.1	s
Time after entry into controlled constant drive					
C_FAC_TQ_REQ_CLU_IS_CRU_HYS	1	0...FFFFH	0...1.999969	3.052E-5	-
Hysteresis on FAC_TQ_REQ_CLU for IS detection					

G.1.7.1.2 Cut-off of the CRU function

General information:

Soft cut-off

In the soft cut-off of the cruise control, the torque request from cruise control FAC_TQ_REQ_CRU is reduced to the driver request FAC_TQ_REQ_DRIV with the adjustable decrement IP_FAC_TQ_CRU_DEC__N_32 until the driver request is reached. FAC_TQ_CRU is then set to zero.

Hard cut-off

In the hard cut-off mode of the cruise control, the torque request FAC_TQ_CRU is set equal to zero. There is no limitation gradient. The CRU State is 'PASSIVE'.

Application conditions:

Activation: see Capter:Cut -off Conditions and switch- on prohibitions

Formula section:

Soft cut-off

For FAC_TQ_REQ_DRIV < FAC_TQ_REQ_CRU there results:

FAC_TQ_CRU = FAC_TQ_CRU $_{n-1}$ - IP_FAC_TQ_CRU_DEC __N_32 until

FAC_TQ_REQ_DRIV = FAC_TQ_REQ_CRU

Hard cut-off

FAC_TQ_CRU = 0

Then, the CRU ready status is reached. Only now can the CRU mode be activated again.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_TQ_CRU_DEC __N_32	1x5	0...FFFFH	0...99,998	0.0015	%
LDP_N_32__ FAC_TQ_CRU	5	0...FFH	0...8160	32	rpm
Decrement of the FAC_TQ_CRU					

G.1.7.2 Controlled acceleration

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_CTL_CRU	V	0...FFH	0...25.5	0.1	s
Time of the controlled acceleration					
VS_SP_CRU	V	0...FFFFH	0...511.9922	0.0078	km.p.h.
Vehicle speed setpoint for the cruise control.					
VS_SP_DIF_CRU	V	FFFF8000 ... 7FFFH	-256...255.9922	0.0078	km.p.h.
Difference between current vehicle speed setoint and requested setpoint of the driver.					

Input data:

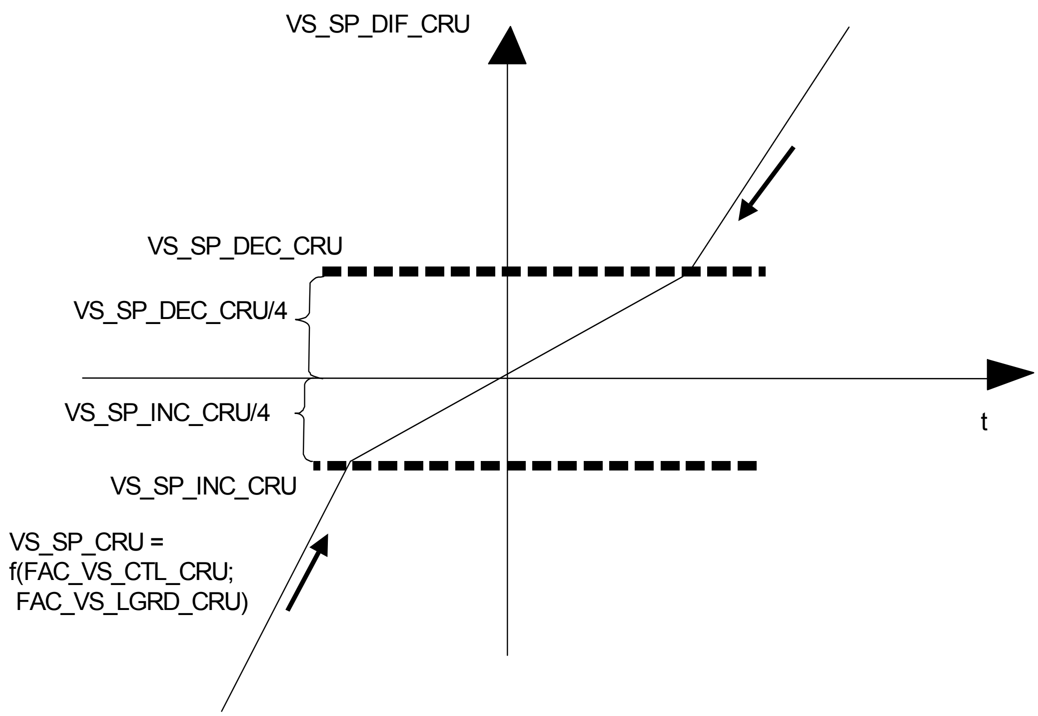
VS_SP_DRIV_CRU	VS_SP_DEC_CRU	VS_SP_INC_CRU	FAC_VS_CTL_CRU
FAC_VS_LGRD_CRU	VS_CRU		

FUNCTION DESCRIPTION:

Application conditions:

Initialisation: VS_SP_CRU = VS_CRU

Description:



The vehicle speed setpoint VS_SP_CRU is set to the current speed of the vehicle VS_CRU at the start of the function.

general specification

The controlled acceleration of the vehicle is performed by controlled in/decrementation of the vehicle speed setpoint. At the start of the function, the time T_CTL_CRU is started and incremented in each computing cycle.
When VS_SP_DIF_CRU is less than VS_SP_DEC/INC_CRU the step size of the decrement/increment is reduced (to VS_SP_DEC_CRU/4 or VS_SP_INC_CRU/4).
A distinction is made between positive and negative acceleration.
For the definition of FAC_VS_CTL_CRU and FAC_VS_LGRD_CRU see next subchapters.

Formula section:

T_CTL_CRUn = T_CTL_CRUn-1 + 100 ms

VS_SP_DIF_CRU = VS_SP_CRU - VS_SP_DRIV_CRU

IF 0 < VS_SP_DIF_CRU ≤ VS_SP_DEC_CRU

Then $VS\_SP\_CRU_n$ = $VS\_SP\_CRU_{n-1}$ - VS_SP_DEC_CRU /4

Else **IF** VS_SP_DIF_CRU > VS_SP_DEC_CRU > 0

Then $VS\_SP\_CRU_n$ = $VS\_SP\_CRU_{n-1}$

- VS_SP_DEC_CRU Decrement
* FAC_VS_CTL_CRU Factor for controlling
* FAC_VS_LGRD_CRU Factor for intercepting

Else If (-1 * VS_SP_INC_CRU) ≤ VS_SP_DIF_CRU < 0

Then $VS\_SP\_CRU_n$ = $VS\_SP\_CRU_{n-1}$ + VS_SP_INC_CRU/4

Else **If** VS_SP_DIF_CRU < (-1 * VS_SP_INC_CRU)

Then $VS\_SP\_CRU_n$ = $VS\_SP\_CRU_{n-1}$

+ VS_SP_INC_CRU Increment
* FAC_VS_CTL_CRU Factor for controlling
* FAC_VS_LGRD_CRU Factor for intercepting

Dec / Incrementation is interrupted if
VS_DIF_CRU > C_VS_DIF_MAX_AC_CRU **or**
VS_DIF_CRU < C_VS_DIF_MIN_AC_CRU

Dec / Incrementation is continued if
VS_DIF_CRU <= C_VS_DIF_MAX_AC_CRU **or**
VS_DIF_CRU >= C_VS_DIF_MIN_AC_CRU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_DIF_MAX_AC_CRU	1	FFFF8000 ... 7FFFH	-256...255.9922	0.0078	km.p.h.
Condition for interrupting the incrementation or decrementation					
C_VS_DIF_MIN_AC_CRU	1	FFFF8000 ... 7FFFH	-256...255.9922	0.0078	km.p.h.
Condition for interrupting the incrementation or decrementation					

G.1.7.3 Actuating the reference input variable

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_VS_CTL_CRU	V	0...FFH	0...1.9922	0.0078	-
Correction factor for actuating the reference input variable					

Input data:

T_CTL_CRU		LV_AT	

General information:

The multiplicative correction factor FAC_VS_CTL_CRU for in/decrementation of the vehicle speed setpoint in the controlled acceleration is calculated calculated separately for each case of the cruise control. The correction factor depends upon the time T_CTL_CRU. While the actuation of this function is active, the calculation of the I-controller part is interrupted and kept constant.

Formula section:

FAC_VS_CTL_CRU = IP_FAC_CRU_yy__T_CTL_CRU

A distinction is made between the following CRU functions:

Resume: IP_FAC_CRU_REST__T_CTL_CRU

Acceleration: IP_FAC_ CRU_AC__T_CTL_CRU

Deceleration: IP_FAC_ CRU_DECE__T_CTL_CRU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
IP_FAC_CRU_REST__T_CTL_CRU	1x5	0...FFH	0...1.9922	0.0078	-
LDPM_T_CTL_CRU_1	5	0...FFH	0...25.5	0.1	s
Correction factor for actuating the vehicle speed setpoint, resume					
IP_FAC_CRU_AC__T_CTL_CRU	1x5	0...FFH	0...1.9922	0.0078	-
LDPM_T_CTL_CRU_1	5	0...FFH	0...25.5	0.1	s
Correction factor for actuating the vehicle speed setpoint, acceleration					
IP_FAC_CRU_DECE__T_CTL_CRU	1x5	0...FFH	0...1.9922	0.0078	-
LDPM_T_CTL_CRU_1	5	0...FFH	0...25.5	0.1	s
Correction factor for actuating the vehicle speed setpoint, deceleration					

G.1.7.4 Intercepting the reference input variable target acceleration

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_VS_LGRD_CRU	V	0...FFH	0...0.9984	0.0039	-
Factor for intercepting the reference input variable					

Input data:

VS_SP_CRU	VS_SP_DRIV_CRU	REQ_MSW	LV_RNG_L_REQ

General information:

To take the vehicle dynamics into account, the positive accelerations at high vehicle speeds must be reduced in the acceleration and resume functions. For this purpose, the accelerations must be valued with a factor FAC_VS_LGRD_CRU for in/decrementing the reference input variable. The acceleration is valued with 1 below the threshold C_VS_AC_MIN_CRU. If VS_SP_CRU exceeds this threshold, then the factor FAC_VS_LGRD_CRU is reduced linearly by a ramp function up to the upper threshold C_VS_AC_MAX_CRU. The maximum reduction can be set through the value C_FAC_AC_POS_CRU.
To achieve a comfortable interception of the vehicle speed setpoint, the multiplicative correction factor, which is limited dependent on speed, must be calculated when in/decrementing the vehicle speed setpoint in each case of the cruise control. While the interception of the reference input variable function is active, the calculation of the I-part is interrupted and kept constant. The correction factor depends upon the difference between vehicle speed setpointand the requested vehicle speed of the driver.

Formula section:

IF VS_SP_CRU < C_VS_AC_MIN_CRU
Then FAC_VS_LGRD_CRU = 1
Else If VS_SP_CRU > C_VS_AC_MAX_CRU
Then **IF LV_RNG_L_REQ = 1**
Then FAC_VS_LGRD_CRU = C_FAC_AC_POS_CRU_RNG_L
Else FAC_VS_LGRD_CRU = C_FAC_AC_POS_CRU
Else FAC_VS_LGRD_CRU = FAC_VS_LGRD_CRU$_{\text{vehicle-speed-dependent}}$
* IP_FAC_LGRD_CRU_xx__VS_SP_DIF

- During the acceleration IP_FAC_LGRD_CRU_xx__VS_SP_DIF = 1
(FAC_VS_LGRD_CRU = FAC_VS_LGRD_CRU$_{\text{vehicle-speed-dependent}}$)
- Resume (RES): IP_FAC_LGRD_CRU_REST__VS_SP_DIF
- Tip Up and Tip down: IP_FAC_LGRD_CRU_TIP__VS_SP_DIF
(TP_UP, TP_DOWN)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_VS_AC_MIN_CRU	1	0...FFH	0...255	1	km.p.h.
Vehicle speed threshold for acceleration valuation					
C_VS_AC_MAX_CRU	1	0...FFH	0...255	1	km.p.h.
Vehicle speed threshold for acceleration valuation					
C_FAC_AC_POS_CRU	1	0...FFH	0...0.996	0.0039	-
Maximum reduction of FAC_VS_LGRD_CRU					
C_FAC_AC_POS_CRU_RNG_L	1	0...FFH	0...0.996	0.0039	-
Maximum reduction of FAC_VS_LGRD_CRU at RNG_L					
IP_FAC_LGRD_CRU_xx__VS_SP_DIF	1x5	0...FFH	0...0.996	0.0039	-
LDPM_VS_SP_DIF_CRU_1	5	0...FFFFH	-256...255.9922	0.0078	km.p.h.
Interception factor of the in/decrementing (xx = REST, TIP)					

G.1.7.5 Interventions

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_SP_INC_CRU	V	0...FFFFH	0...511.9922	0.0078	km.p.h.
Increment for VS_SP_CRU					
VS_SP_DEC_CRU	V	0...FFFFH	0...511.9922	0.0078	km.p.h.
Decrement for VS_SP_CRU					
VS_SP_DRIV_CRU	V/O	0...FFFFH	0...511.9922	0.0078	km.p.h.
Requested vehicle speed of the driver					
VS_SP_DIF_DRIV_CRU	V/O	0...FFFFH	0...511.9922	0.0078	km p.h.
VS difference between driver request and VS_CRU					
LV_CRU_OVER_ACT_ACK	V/O	0...1H	0...1	1	-
hand shake overtaking function					

Input data:

T_CTL_CRU	VS_CRU	VS_SP_CRU	AC_CRU
C_VS_MIN_CRU	VS_DIF_CRU	REQ_MSW	FAC_TQ_CRU
LV_RNG_L_REQ	N	VS_SP_DRIV_STEP	LV_IGK
REQ_CRU	LV_CRU_OVER_ACT		

Application conditions:

Initialisation: VS_SP_DRIV_CRU = C_VS_MIN_CRU at LV_CRU_MAIN_SWI 0 to1

G.1.7.5.1 Resume (RES)

Description:

When the resume function is activated, the vehicle is brought to the last target speed with the adjustable positive or negative acceleration. In this case the control acceleration function is activated and the vehicle speed setpoint is controlled. When controlling the vehicle speed setpoint VS_SP_CRU, a distinction is made between an acceleration and a deceleration.
When the resume function is finished by pressing the SET/COAST button, the VS_SP_DRIV_CRU = VS_SP_CRU = VS_CRU. The SET/COAST button will be interpreted as SET. If a deceleration is required the SET/COAST button has to be pressed a second time.
In resume mode the vehicle speed setpoint is in/decremented every 100 ms.

To achieve a comfortable transition from the STATE_CRU = “PASSIVE” to “RESUME” at the start of the resume function, the I-part of the P/I controller is initialised depending on the acceleration of the vehicle.
At activation the resume mode after overtaking function is finished, the resume function is activated with the initialisation of FAC_TQ_CRU.

Formula section:

IF VS_CRU < VS_SP_DRIV_CRU

Then IF LV_RNG_L_REQ = 1

Then VS_SP_INC_CRU = 3.6 * C_AC_POS_CRU_RESU_RNG_L * 100 ms

Else VS_SP_INC_CRU = 3.6 * ID_AC_POS_CRU_RESU_MT/AT * 100 ms

VS_SP_CRU = capter "calculation of controlled acceleration"

Else

If VS_CRU > VS_SP_DRIV_CRU

Then VS_SP_DEC_CRU = 3.6 * ID_AC_NEG_CRU_RESU_MT/AT * 100 ms

VS_SP_CRU = capter "calculation of controlled acceleration"

Endif

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AC_POS_CRU_RESU_RNG_L	1	0...5C28H	0...511.9789	0.022	m/s^2
Positive acceleration at CRU resume at RNG_L					
ID_AC_POS_CRU_RESU_MT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Positive acceleration in resume for MT-vehicle					
ID_AC_POS_CRU_RESU_AT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Positive acceleration in resume for AT-vehicle					
ID_AC_NEG_CRU_RESU_MT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Negative acceleration in resume for MT-vehicle					
ID_AC_NEG_CRU_RESU_AT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Negative acceleration in resume for AT-vehicle					

G.1.7.5.2 Setting/acceleration (SET/ACC)

Description:

At the activation of the setting/acceleration function from the STATE_CRU = “PASSIVE” , the vehicle speed setpoint VS_SP_CRU and requested vehicle speed of the driver VS_SP_DRIV_CRU are set to the current vehicle speed VS_CRU.

On activating the setting/acceleration function from the STATE_CRU = “PASSIVE”, a timer T_CTL_CRU is started. After this timer reaches the value C_T_MIN_CRU_CON, the STATE_CRU changes from ‘controlled constant drive’ to ‘controlled acceleration’. The vehicle is brought to the required target speed with the adjustable positive ID_AC_POS_CRU_AC_MT/AT or negative acceleration IP_AC_NEG_CRU_AC__N__FAC_TQ_CRU. In this case the controlled acceleration function is activated and the vehicle speed setpoint VS_SP_CRU is controlled.
As long as the setting/acceleration function is active, the incrementation of the vehicle speed setpoint takes place per computing cycle (100 ms) with the controlled acceleration function.
On ending the setting/acceleration function by releasing the SET button, the target speed VS_SP_DRIV_CRU is set equal to the achieved vehicle speed VS_CRU. If the target speed VS_SP_DRIV_CRU stored in this way is greater than the current vehicle speed setpoint VS_SP_CRU, then the value VS_SP_CRU is incremented further by the value VS_SP_INC_CRU through the controlled acceleration function. If the current vehicle speed setpoint VS_SP_CRU is larger than the stored target speed VS_SP_DRIV_CRU, then the vehicle speed setpoint is decremented with the controlled acceleration function.

Formula section:

- REQ_MSW = ACC active:

  IF LV_RNG_L_REQ = 1

  Then VS_SP_INC_CRU = 3.6 * C_AC_POS_CRU_AC_RNG_L * 100 ms

  Else VS_SP_INC_CRU = 3.6 * ID_AC_POS_CRU_AC_MT/AT * 100 ms

  $VS\_SP\_CRU_n$ = $VS\_SP\_CRU_{n-1}$

  + VS_SP_INC_CRU :Increment

  * FAC_VS_CTL_CRU :Factor for controlling

  * FAC_VS_LGRD_CRU :Factor for intercepting

- REQ_MSW = ACC passive:

  VS_SP_DRIV_CRU = VS_CRU

IF T_CONST_DRIVE >= C_T_MIN_CRU_CON

Then IF VS_SP_CRU < VS_SP_DRIV_CRU

Then IF LV_RNG_L_REQ = 1

Then VS_SP_INC_CRU = 3.6 * C_AC_POS_CRU_AC_RNG_L * 100 ms

Else VS_SP_INC_CRU = 3.6 * ID_AC_POS_CRU_AC_MT/AT * 100 ms

VS_SP_CRU = capter "calculation of controlled acceleration"

Else

If VS_SP_CRU > VS_SP_DRIV_CRU

Then VS_SP_DEC_CRU = 3.6 * IP_AC_NEG_CRU_AC__N__FAC_TQ_CRU* 100 ms

VS_SP_CRU = capter "calculation of controlled acceleration"

Endif

VS_SP_CRU <= C_VS_SP_CRU_MAX
VS_SP_DRIV_CRU <= C_VS_SP_CRU_MAX

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AC_POS_CRU_AC_RNG_L	1	0...5C28H	0...511.9789	0.022	m/s^2
Positive acceleration in setting/acceleration at RNG_L					
C_T_MIN_CRU_CON	1	0...FFH	0...25.5	0.1	s
Time condition for controlled acceleration					
IP_AC_NEG_CRU_AC__N__FAC_TQ_CRU	3x3	0...5C28H	0...511.9789	0.021701	m/s^2
LDPM_N_21	3	0...FFH	0...8160	32	rpm
LDPM_FAC_TQ_CRU_1	3	0...FFFFH	0...99,998	0.0015	%
Negative acceleration in setting/acceleration					
ID_AC_POS_CRU_AC_MT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Positive acceleration in setting/acceleration for MT-vehicle					
ID_AC_POS_CRU_AC_AT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Positive acceleration in setting/acceleration for AT-vehicle					

G.1.7.5.3 Tip-up

Description:

On activating the TIP_UP function, the vehicle is accelerated to the last target speed reached VS_SP_DRIV_CRU plus C_VS_CRU_TIP_STEP. with the adjustable positive acceleration C_AC_POS_CRU_TIP. In this case the controlled acceleration function is activated and the vehicle speed setpoint VS_SP_CRU is controlled. At each activation of the TIP_UP function, the target speed VS_SP_DRIV_CRU is increased by C_VS_CRU_TIP_STEP.
If the absolute amount of the difference VS_DIF_CRU is greater than or equal to C_VS_DIF_MAX_CRU, the vehicle speed setpoint VS_SP_CRU is incremented per computing cycle (100 ms) with the controlled acceleration function as defined below:

Formula section:

```
IF      LV_CRU_OVER_ACT = 1   and   REQ_CRU = "TIP_UP"
Then    VS_SP_CRU =VS_SP_DRIV_CRU = VS_CRU
        LV_CRU_OVER_ACT_ACK = 1
else    LV_CRU_OVER_ACT_ACK = 0
endif
```

VS_SP_DIF_DRIV_CRU = VS_SP_DRIV_CRU – VS_CRU

IF VS_SP_DRIV_CRU >= C_VS_SP_CRU_MAX

THEN VS_SP_DRIV_CRU = C_VS_SP_CRU_MAX

ELSE *(Limitation of VS_SP_DRIV_CRU increase with tip-up in case of VS_CRU cannot reach VS_SP_DRIV_CRU:)*

IF VS_SP_DRIV_CRU >= VS_CRU + C_VS_DIF_DRIV_CRU

THEN **IF** VS_SP_DIF_DRIV_CRU ≤ C_VS_SP_DIF_DRIV_CRU

THEN VS_SP_DRIV_CRU = VS_SP_DRIV_CRU + C_VS_CRU_TIP_STEP

ELSE VS_SP_DRIV_CRU (n) = VS_SP_DRIV_CRU (n-1)

ELSE VS_SP_DRIV_CRU = VS_SP_DRIV_CRU + C_VS_CRU_TIP_STEP

IF VS_DIF_CRU >= C_VS_DIF_MAX_CRU

Then VS_SP_INC_CRU = 3.6 * C_AC_POS_CRU_TIP * 100 ms

$VS\_SP\_CRU_{n} = VS\_SP\_CRU_{n-1}$

+ VS_SP_INC_CRU :Increment

* FAC_VS_CTL_CRU :Factor for controlling

* FAC_VS_LGRD_CRU :Factor for intercepting

Else VS_SP_CRU = VS_SP_CRU + C_VS_CRU_TIP_STEP.

VS_SP_CRU <= C_VS_SP_CRU_MAX
VS_SP_DRIV_CRU <= C_VS_SP_CRU_MAX

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AC_POS_CRU_TIP	1	0...5C28H	0...511.9789	0.0217	m/s^2
Positive acceleration for Tip-up					
C_VS_DIF_MAX_CRU	1	0...FFH	0...31.875	0.125	km.p.h.
Control difference threshold for incrementing VS_SP_CRU					
C_VS_SP_CRU_MAX	1	0...FFFFH	0...511.9922	0.0078	km.p.h.
maximum limitation of cruise control					
C_VS_CRU_TIP_STEP	1	0...FFFFH	0...8.045	1.23e-4	Km.p.h.
Step size for a tip up/tip down.					
C_VS_DIF_DRIV_CRU	1	0...FFFFH	0...511.9922	0.0078	km p.h.
VS difference threshold for VS_SP_DRIV_CRU limitation by using tip up					
C_VS_SP_DIF_DRIV_CRU	1	0...FFFFH	0...511.9922	0.0078	km p.h.
Max. allowed VS diff. without SP limitation with tip up					

G.1.7.5.4 Tip-down

Description:

On activating the TIP_DOWN function, the vehicle is decelerated to the last target speed reached VS_SP_DRIV_CRU minus C_VS_CRU_TIP_STEP by the adjustable negative acceleration C_AC_NEG_CRU_TIP. In this case the controlled acceleration function is activated and the vehicle speed setpoint VS_SP_CRU is controlled. At each activation of the TIP_DOWN function, the target speed VS_SP_DRIV_CRU is reduced by C_VS_CRU_TIP_STEP VS_SP_CRU and VS_SP_DRIV_CRU may not be less than C_VS_MIN_CRU.
If the absolute amount of the difference VS_DIF_CRU is greater than or equal to C_VS_DIF_MAX_CRU, the vehicle speed setpoint VS_SP_CRU is decrements per computing cycle (100 ms) with the controlled acceleration function. The torque request of the cruise control FAC_TQ_CRU is limited to the value (IP_FAC_TQ_REQ_CLU_IS__N_32*99,998%), so that no trailing throttle cut-off is triggered (except for: PUC was already active before TIP_DOWN).

Formula section:

IF VS_SP_DRIV_CRU < C_VS_MIN_CRU

Then VS_SP_DRIV_CRU = C_VS_SP_CRU_MIN

Else VS_SP_DRIV_CRU = VS_SP_DRIV_CRU - C_VS_CRU_TIP_STEP

IF |VS_DIF_CRU | >= C_VS_DIF_MAX_CRU

Then VS_SP_DEC_CRU = 3.6 * C_AC_NEG_CRU_TIP * 100 ms

$VS\_SP\_CRU_n = VS\_SP\_CRU_{n-1}$

- VS_SP_DEC_CRU :Decrement

* FAC_VS_CTL_CRU :Factor for controlling

* FAC_VS_LGRD_CRU :Factor for intercepting

Else VS_SP_CRU = VS_SP_CRU - C_VS_CRU_TIP_STEP

Endif

FAC_TQ_CRU >= ((IP_FAC_TQ_REQ_CLU_IS__N_32 - C_FAC_TQ_REQ_CLU_IS_HYS) * 99,998%)

VS_SP_CRU >= C_VS_MIN_CRU

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AC_NEG_CRU_TIP	1	0...5C28H	0...511.9789	0.0217	m/s^2
Positive acceleration for Tip-up					
C_VS_DIF_MAX_CRU	1	0...FFH	0...31.875	0.125	km.p.h.
Control difference threshold for incrementing VS_SP_CRU					

G.1.7.5.5 Deceleration

Description:

On activating the deceleration function, the vehicle is decelerated to the required target speed with the adjustable negative ID_AC_NEG_CRU_DEACC_MT/AT or positive acceleration C_AC_POS_CRU_DEACC (for the steady state). In this case the controlled acceleration function is activated and the vehicle speed setpoint VS_SP_CRU is controlled.
As long as the deceleration function is activated, the decrementation of the vehicle speed setpoint takes place per computing cycle (100 ms) with the controlled acceleration function.

On ending the deceleration function, the target speed VS_SP_DRIV_CRU is set equal with the reached vehicle speed VS_CRU. If the target speed VS_SP_DRIV_CRU thus stored is smaller than the current vehicle speed setpoint VS_SP_CRU, then the value VS_SP_CRU is decremented by the value VS_SP_DEC_CRU further via the controlled acceleration function. VS_SP_CRU and VS_SP_DRIV_CRU may not be smaller than C_VS_MIN_CRU. If the current vehicle speed setpoint VS_SP_CRU is smaller than the stored target speed VS_SP_DRIV_CRU, then the vehicle speed setpoint is incremented with the controlled acceleration function.

Formula section:

- REQ_MSW = DEC active:

  VS_SP_DEC_CRU = 3.6 * ID_AC_NEG_CRU_DEACC_MT/AT * 100 ms

  $VS\_SP\_CRU_{n}$ = $VS\_SP\_CRU_{n-1}$

  - VS_SP_DEC_CRU :Decrement

  * FAC_VS_CTL_CRU :Factor for controlling

  * FAC_VS_LGRD_CRU :Factor for intercepting

- REQ_MSW = DEC passive:

  VS_SP_DRIV_CRU = VS_CRU

  IF VS_SP_DRIV_CRU < VS_SP_CRU

  Then VS_SP_DEC_CRU = 3.6 * ID_AC_NEG_CRU_DEACC_MT/AT * 100 ms

  VS_SP_CRU = capter "calculation of controlled acceleration"

  Else VS_SP_INC_CRU = 3.6 * C_AC_POS_CRU_DEACC * 100 ms

  VS_SP_CRU = capter "calculation of controlled acceleration"

  Endif

As long as the decrementation of the vehicle speed setpoint VS_SP_CRU is active, the calculation of the I-part is released.

Calculation of VS_DIF_I_CRU until

VS_DIF_I_CRU + C_VS_DIF_DELTA_I < |VS_DIF_CRU|

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_AC_POS_CRU_DEACC	1	0...5C28H	0...511.9789	0.022	m/s^2
Positive acceleration					
C_VS_DIF_DELTA_I	1	0...FFH	0...63,75	0.25	km.p.h
Positive acceleration					
ID_AC_NEG_CRU_DEACC_MT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Negative acceleration in deceleration for MT-vehicle					
ID_AC_NEG_CRU_DEACC_AT	1x9	0...5C28H	0...511.9789	0.022	m/s^2
LDPM_GEAR	9	0...FFH	0...255	1	-
Negative acceleration in deceleration for AT-vehicle					

G.1.7.5.6 OFF

In the OFF function, the last setpoint VS_SP_DRIV_CRU remains stored and soft cut-off is activated. If the engine running is stopped (LV_IGK, LV_ES) the CRU - readiness and the last speed-setpoint are to reset to their initial values.

G.1.8 Additional functions

G.1.8.1 Runup lock

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
N_VS_CRU	O/V	0...1FE0H	0...16320	2	min^{-1} / km.p.h.
Ratio between engine speed and vehicle speed					

Input data:

N	VS_WORD	LV_AT	

Application conditions:

Initialisation: 0 at reset

Activation: LV_AT = 0

Deactivation: LV_AT = 1

Recurrence: 100 ms

Description:

In vehicles with manual transmission, the ratio between engine speed and vehicle speed is monitored on activating a function of the cruise control. At the start of a function, the ratio

N_VS_CRU = N / VS_WORD is stored.

The instantaneous ratio between engine speed and vehicle speed is formed in each cycle

$N\_VS\_CRU_n = N_n / VS\_WORD_n$

Remark:

If the percentage ratio exceeds the threshold value C_N_VS_MAX_CRU, hard cut-off has to ocurrs.

G.1.8.2 Filter of Vehicle Speed for Cruise control

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
VS_CRU	O/V	0...FFFFH	0...511.9922	0.0078	km.p.h.
Vehicle speed for cruise control					
VS_CRU_FIL	O/V	0...FFFFH	0...511.9922	0.0078	km.p.h.
Filtered vehicle speed for cruise control					

Input data:

VS_WORD			

Application conditions:

Initialisation: 0 at reset

Activation: at every engine state

Recurrence: 10ms

General information:

Formula section:

$VS\_CRU\_FIL = VS\_CRU\_FIL_{n-1} + C\_GAIN\_VS\_CRU\_FIL * (VS\_WORD_n – VS\_CRU\_FIL_{n-1})$

$VS\_CRU = VS\_CRU\_FIL$

Calibration data:

C_GAIN_VS_CRU_FIL	1	0...FFFFH	0...1	1.526 E-5	-
Filter gain of the vehicle speed for cruise control					

G.1.9 Checking the conditions of the CRU mode

G.1.9.1 Acceleration monitoring

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
AC_CRU	O/V	8000...7FFFH	-15,72...15,72	0.00048	m/s^2
Vehicle acceleration in CRU					

Input data:

VS_CRU_FIL			

Application conditions:

Initialisation: 0 m/s² at reset

Recurrence: 10 ms

Description:

To monitor the vehicle dynamics in cruise control mode, the instantaneous acceleration of the vehicle is calculated in each computing cycle. The acceleration calculation is performed independently of the CRU mode with a 10 ms time pattern.

Formula section:

$$AC\_CRU = AC\_CRU + (((VS\_CRU\_FIL_n - VS\_CRU\_FIL_{n-1}) / 10ms) - AC\_CRU) * C\_FAC\_AC\_CRU\_MMV$$

Remark:
If the instantaneous acceleration exceeds the adjustable value C_AC_MAX_CRU, hard cut-off is performed, see chapter hard off conditions.

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_AC_CRU_MMV	1	0...FFFFH	0...1	1.5E-05	-
Filtering AC_CRU					

G.1.9.2 Recognition ASR/ESP-control and valid minimum time

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
LV_CRU_OFF_BY_ASR_MSR_CTL	V/O	0...1H	0....1	1	-
Soft off condition depending on ASR, MSR interventions					

Input data:

LV_MSR_REQ	LV_ASR_REQ		

Application conditions:

Initialisation: 0 at reset

Recurrence: 100 ms

Description:

Recognition ASR/MSR-control and valid minimum time :

LV_CRU_OFF_BY_ASR_MSR_CTL = active:

Recognition due to the combination of the signals LV_MSR_REQ, LV_ASR_REQ .

LV_MSR_REQ	LV_TQ_ASR_REQ		function	minimum time before cut-off CRU
0	1		ASR-control	C_T_CRU_ASR_OFF
1	0		MSR-control	C_T_CRU_MSR_OFF

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_T_CRU_MSR_OFF	1	0...FFH	0...25.5	0.1	s
minimum time before cut-off CRU in case of ESP intervention					
C_T_CRU_ASR_OFF	1	0:::FFH	0...25.5	0.1	s
minimum time before cut-off CRU in case of ASR intervention					

G.1.9.3 Mean moving value calculation of Driver demand:

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
FAC_TQ_REQ_DRIV_MMV	V/O	0...FFFFH	0...1,999969	30,5e-6	-
Mean moving value of driver torque demand					

Input data:

FAC_TQ_REQ_DRIV			

FUNCTION DESCRIPTION:

For initialisation of I share of the cruise control function it is nessessary to calulate a very slow driver demand

Application conditions:

Initialisation: 0 at reset

Recurrence: 10 ms

Formula section:

FAC_TQ_REQ_DRIV_MMV = $\text{FAC\_TQ\_REQ\_DRIV\_MMV}_{n-1}$ +

(FAC_TQ_REQ_DRIV - FAC_TQ_REQ_DRIV_MMV$_{n-1}$) *

C_FAC_TQ_REQ_DRIV_MMV

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_FAC_TQ_REQ_DRIV_MMV	1	0...FFFFH	0...0.9999	15e-6	-
Filtering of driver demand					

J EOL

Table of Contents

Index

J.1 EVAP leakage detection

Output data:

Name	Mode	Hex. limits	Phys. limits	Resol.	Unit
T_DLY_EOL_DIAGCP	O	0...FFH	0...255	1	s
Status of evaporative system monitoring during end of line test					
EOL_STATE_DIAGCP	V	0...FFH	0...255	1	-
Status of evaporative system monitoring during end of line test					

Input data:

LV_ERR_VS	LV_ERR_MWSS	LV_ERR_TIA	LV_ERR_ISC
LV_N_LIM_ETC_LIH	LV_MTC_CUR_OFF	LV_ERR_MAP	LV_ERR_TCO
LV_ERR_TCO_PLAUS	LV_ERR_TCO_STUCK	LV_ERR_TCO_GRD	LV_ERR_CPS
LV_ERR_FSD_LAM_LIM_i	LV_ERR_LS_UP_i	FAC_DIAM_DIAGCP	STATE_INTR_DIAGCP
LV_ERR_TPS	LV_ERR_TPS_PLAUS	LV_ERR_DTP	LV_ERR_ISA_2
LV_ERR_SOV	LV_ERR_MEC_SOV	LV_ERR_MAF	LV_ERR_ISA_1
LV_ERR_IV_x	LV_ERR_IGC	CL_MMV	FAC_DIAGCP_VOL
LV_ERR_FSD_i	LV_MIS_STATE_A	VS	C_VS_MAX_DIAGCP
LV_CP_SET_CLOSE	LV_CDN_VB_OBD2	DTP	C_DTP_MIN_DIAG
C_DTP_MAX_DIAG	CP_STATE	LV_TRIP_ACT_IVVT_IN	LV_TRIP_ACT_IVVT_EX
LV_LSCL_i	FAC_DIAGCP_MDL_BAS	FAC_LAM_MV_MMV	FAC_LAM_DIAGCP
DTP_DIF_ACT	C_DTP_DIF_MIN_DIAGCP	N_DIF	

J.1.1 General information

The end of line test is used in the manufacturing plant or at the service for a functional check of the evaporative system and leakage check.

The diagnosis is started via the serial interface and works as in case of the “normal” evaporative diagnosis activation.

J.1.2 Conditions to start the diagnosis

Application conditions:

Activation:

To start the algorithm, the following conditions must exist:

- No electrical diagnostic error related to:
  - vehicle speed (LV_ERR_VS = 0)
  - magnetic wheel speed-sensor (LV_ERR_MWSS = 0)
  - intake air temperature (LV_ERR_TIA = 0)
  - coolant temperature (LV_ERR_TCO = 0 , LV_ERR_TCO_PLAUS = 0, LV_ERR_TCO_STUCK = 0, LV_ERR_TCO_GRD = 0)
  - evaporative emission control valve (LV_ERR_CPS = 0)
  - lambda controller (LV_ERR_FSD_LAM_LIM_i = 0)
  - O2 sensors (LV_ERR_LS_UP_i = 0)
  - ETC (LV_ERR_TPS = 0, LV_ERR_TPS_PLAUS = 0, LV_N_LIM_ETC_LIH = 0, LV_MTC_CUR_OFF = 0)
  - ISA (LV_ERR_ISC=0, LV_ERR_ISA_1=0, LV_ERR_ISA_2=0)
  - tank pressure sensor (LV_ERR_DTP =0)
  - shutoff valve (LV_ERR_SOV = 0)
  - shutoff valve stuck (LV_ERR_MEC_SOV = 0)
  - Load-sensor (LV_ERR_MAF = 0, LV_ERR_MAP = 0)
  - injector valves (LV_ERR_IV_x = 0)
  - ignition coils (LV_ERR_IGC = 0)
  - fuel system diagnosis (LV_ERR_FSD_i = 0)
  - catalyst damage misfire (LV_MIS_STATE_A = 0)
- VS < C_VS_MAX_DIAGCP
- Canister purge not forced to NO_PURGE mode by “lambda control limit diagnosis” (LV_CP_SET_CLOSE = 0)
- To avoid problems with low battery voltage, LV_CDN_VB_OBD2 = 1. If this condition is not fulfilled the system switches to ‘wait period’.
- Tank pressure has to be within a certain range :

  C_DTP_MIN_DIAG < DTP < C_DTP_MAX_DIAG
- Canister purge must be active: CP_STATE <> CP_NOT_ACTIVE
- No request of IVVT EOL to increase N_SP_IS. Need to avoid duplicate rpm increment

  LV_TRIP_ACT_IVVT_IN = 0 and LV_TRIP_ACT_IVVT_EX = 0

Deactivation:

The evaporative system monitoring is stopped for this engine run if an electrical or mechanical error is present or the leakage detection algoritm is finished:

- Diagnostic error related to:
  - vehicle speed sensor (LV_ERR_VS = 1)
  - magnetic wheel speed-sensor (LV_ERR_MWSS = 1)
  - coolant temperature sensor (LV_ERR_TCO = 1, LV_ERR_TCO_PLAUS = 1, LV_ERR_TCO_STUCK = 1, LV_ERR_TCO_GRD = 1)
  - intake air temperature sensor (LV_ERR_TIA = 1)
  - evaporative emission control valve (LV_ERR_CPS = 1)
  - lambda controller (LV_ERR_FSD_LAM_LIM_i = 1)
  - O2 sensor (LV_ERR_LS_UP_i = 1)
  - ETC (LV_ERR_TPS = 1, LV_ERR_TPS_PLAUS = 1, LV_N_LIM_ETC_LIH = 1, LV_MTC_CUR_OFF = 1)
  - ISA (LV_ERR_ISC=1, LV_ERR_ISA_1=1, LV_ERR_ISA_2=1)
  - tank pressure sensor (LV_ERR_DTP =1)
  - shutoff valve (LV_ERR_SOV = 1)
  - shutoff valve stuck (LV_ERR_MEC_SOV = 1)
  - Load-sensor (LV_ERR_MAF = 1, LV_ERR_MAP = 1)
  - injector valves (LV_ERR_IV_x = 1)
  - ignition coils (LV_ERR_IGC = 1)
  - fuel system diagnosis (LV_ERR_FSD_i = 1)
  - catalyst damage misfire (LV_MIS_STATE_A = 1)

or

- VS >= C_VS_MAX_DIAGCP
- Conditions of limited dynamics for pressure curve no longer fulfilled
- LV_CP_SET_CLOSE = 1
- LV_CDN_VB_OBD2 = 0
- Lambda Control loop opened (LV_LSCL_i = 0)

or

- Too high pressure increase during ´test for vapour generation´:
  FAC_DIAGCP_MDL_BAS > C_FAC_DIAGCP_MDL_BAS_MAX_EOL

or

- deviation of lambda-integrator above the threshold during ´evacuation´:

```
{  (    FAC_LAM_DIAGCP -  FAC_LAM_MV_MMV <= C_FAC_LAM_MIN_DIAGCP_EOL
   or   FAC_LAM_DIAGCP -  FAC_LAM_MV_MMV >= C_FAC_LAM_MAX_DIAGCP_EOL )
and
   (    DTP_DIF_ACT <= C_DTP_DIF_MIN_DIAGCP        //  missing tank cap not suspected
   or   N_DIF > C_N_DIF_MAX_DIAGCP_EOL        )    //  engine stability check
}
```

or

- error detection by the algorithm (CPS stuck, tank cap missing, large leak or small leak)

or

- algorithm finished without error detection

or

- Degree of canister saturation is too high: CL_MMV > C_CL_MMV_MAX_EOL

Remark: The EOL-test can be restarted by diagnosis tool without waiting period.

Delay time T_DLY_EOL_DIAGCP is set to C_T_DLY_EOL_DIAGCP after EOL-test(EOL_STATE from _ACT to _RDY) and decremented.

J.1.3 Outputs to the diagnostic tester

To evaluate the diagnosis result or to understand the cause of diagnosis end, it is neccessary to send some information via KW 2000 to the diagnosis tester:

- DTP
- FAC_DIAGCP_VOL
- FAC_DIAM_DIAGCP
- STATE_INTR_DIAGCP
- EOL_STATE_DIAGCP

In case of an error this will be entered in the normal failure memory (i.e. the DTCs are normally managed).

With :

EOL_STATE_DIAGCP =	0 - EOL_PAS	(EOL-test passive)
	1 - EOL_ACT	(EOL-test active, no result yet)
	2 - EOL_RDY	(EOL-test passive, result is present)

Calibration data:

Name	Dim	Hex. limits	Phys. limits	Resol.	Unit
C_CL_MMV_MAX_EOL	1	0...FFFFH	0...8	0.00012	-
Maximum Canister Load for End-Of-Line Test start					
C_T_DLY_EOL_DIAGCP	1	0...FFH	0...255	1	s
Time delay to avtivate normal Evap.diagnosis after EOL test.					
C_FAC_LAM_MIN_DIAGCP_EOL	1	8000...7FFFH	-50...50	0.0015	%
Minimum allowed lambda integrator deviation during evacuation at EOL test.					
C_FAC_LAM_MAX_DIAGCP_EOL	1	8000...7FFFH	-50...50	0.0015	%
Maxium allowed lambda integrator deviation during evacuation at EOL test.					
C_FAC_DIAGCP_MDL_BAS_MAX_EOL	1	0...FFFFH	0.467	7.126 e-6	-
Maximum allowed leakage parameter during vapor generation period at EOL test					
C_N_DIF_MAX_DIAGCP_EOL	1	8000...7FFFH	-32768...32767	1	rpm
RPM deviation threshold for engine stability during evacuation at EOL test					

J.2 IVVT Short Trip (Forced Diagnosis)

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
STATE_TRIP_IVVT_IN_i	V	0H 1H 2H 3H 4H 5H	POS_ST POS_0 POS_1 POS_2 POS_ERR POS_END	1	-
Short trip state; inlet					
STATE_TRIP_IVVT_EX_i	V	0H 1H 2H 3H 4H 5H	POS_ST POS_0 POS_1 POS_2 POS_ERR POS_END	1	-
Short trip state; exhaust					
LV_TRIP_ACT_IVVT_IN	O/V	0...1H	0...1	1	-
1 = short trip active (in progress), 0 = not active; inlet					
LV_TRIP_ACT_IVVT_EX	O/V	0...1H	0...1	1	-
1 = short trip active (in progress), 0 = not active; exhaust					
LV_TRIP_END_IVVT_IN	O/V	0...1H	0...1	1	-
1 = short trip finished, 0 = not finished; inlet					
LV_TRIP_END_IVVT_EX	O/V	0...1H	0...1	1	-
1 = short trip finished, 0 = not finished; exhaust					
LV_ERR_TRIP_IVVT_IN_i	O/V	0...1H	0...1	1	-
Present failure short trip; inlet					
LV_ERR_TRIP_IVVT_EX_i	O/V	0...1H	0...1	1	-
Present failure short trip; exhaust					
LV_END_DIAG_TRIP_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; inlet					
LV_END_DIAG_TRIP_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnosis done completely at least one time; exhaust					
LV_CDN_DIAG_TRIP_IVVT_IN_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); inlet					
LV_CDN_DIAG_TRIP_IVVT_EX_i	V	0...1H	0...1	1	-
Diagnostic condition to start symptom detection (set to 1 when condition is fulfilled); exhaust					
ERR_SYM_TRIP_IVVT_IN_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); inlet					
ERR_SYM_TRIP_IVVT_EX_i	V	0H 1H 2H 4H 8H	NO SYM_0 SYM_1 SYM_2 SYM_3	1	-
For each symptom : status of failure (set to 1 when failure symptom detected); exhaust					
T_TRIP_ST_IVVT_IN_i	-	0...FFFFH	0...655350	10	ms
Point in time at adjustment start of short trip phase; inlet					
T_TRIP_ST_IVVT_EX_i	-	0...FFFFH	0...655350	10	ms
Point in time at adjustment start of short trip phase; exhaust					
CAM_SP_TRIP_IVVT_IN_i	O/V	0...FFH	60...155.625	0.375	°CRK
Camshaft position setpoint during short trip; inlet					
CAM_SP_TRIP_IVVT_EX_i	O/V	0...FFH	-40.125...-135.75	0.375	°CRK
Camshaft position setpoint during short trip; exhaust					

Input data:

LV_NOT_ADJ_CAM_IVVT_IN_i	LV_NOT_ADJ_CAM_IVVT_EX_i	NC_NR_CBK_IVVT	LV_ACT_IVVT
LV_DIAG_TRIP_AVL_IVVT_IN	LV_DIAG_TRIP_AVL_IVVT_EX	N_SP_IS_EOL	N
CAM_AV_IVVT_IN_i	CAM_AV_IVVT_EX_i	C_CAM_INI_IN	C_CAM_INI_EX
LC_CAM_SP_MAN_IVVT_IN	LC_CAM_SP_MAN_IVVT_EX	T_DLY_MES_IVVT	LV_ERR_TPS
LV_ERR_LOAD_TPS_PLAUS	LV_MTC_CUR_OFF	LV_ERR_ISA_i	EOL_STATE_DIAG CP

FUNCTION DESCRIPTION:

General information:

The IVVT actuator test is realized by a short trip. The short trip is a forced diagnosis. It means that certain setpoints are prescribed. They must be reached in specified time.

Description:

The short trip is enabled only under certain conditions, see "IVVT Short Trip (Forced Diagnosis) - Application Incidences". The short trip is a direct error detection.

The prescribed camshaft positions of the short trip are: reference position, C_CAM_SP_TRIP_IVVT_IN(_EX), reference position. Each transition must be reached within the time C_T_ADJ_TRIP_IVVT_IN(_EX). If not the error flag LV_ERR_TRIP_IVVT_IN(_EX)_i is set 1. In this case the short trip must be inactivated by cancelling of the short trip request, see "IVVT Short Trip (Forced Diagnosis) - Application Incidences". A successful short trip is indicated by LV_TRIP_END_IVVT_IN(_EX) = 1.

Error-symptoms are defined to this diagnostic function as follows:

```
[ | | | ]
 │  │  │  └─> TRIP_IVVT_IN(_EX)_i          (= SYM_0)
 │  │  └────> (not used)                  (= SYM_1)
 │  └───────> (not used)                  (= SYM_2)
 └──────────> (not used)                  (= SYM_3)
```

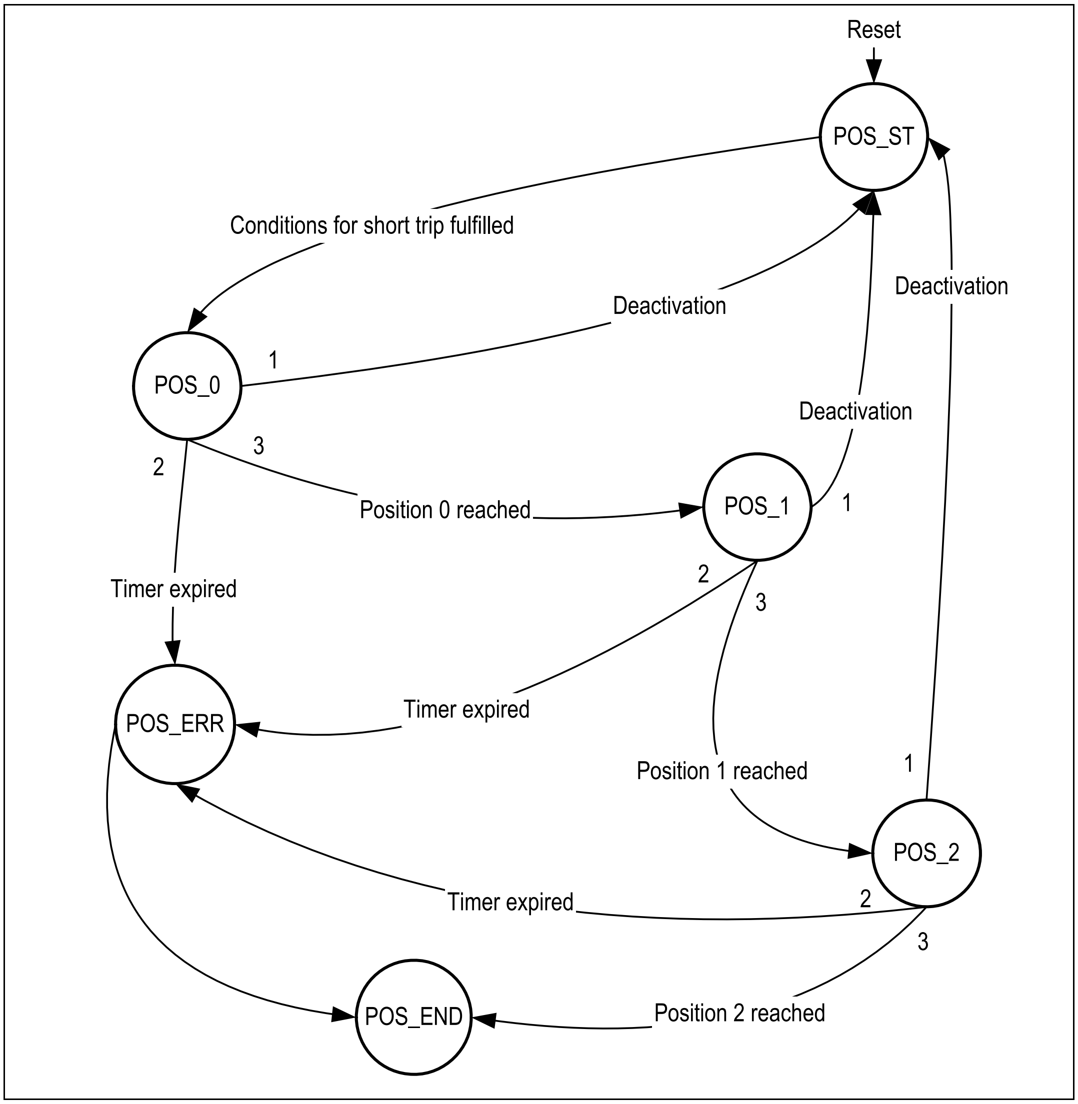


Figure: "Short trip state diagram."

<u>Application conditions:</u>

Initialization:

At reset and at failure memory reset:

STATE_TRIP_IVVT_IN(_EX)_i = "POS_ST"

LV_TRIP_END_IVVT_IN(_EX) = 0

At reset:

LV_CDN_DIAG_TRIP_IVVT_IN(_EX)_i = 0

LV_END_DIAG_TRIP_IVVT_IN(_EX)_i = 0

LV_ERR_TRIP_IVVT_IN(_EX)_i = 0

ERR_SYM_TRIP_IVVT_IN(_EX)_i = 0

At deactivation:

if LV_TRIP_ACT_IVVT_IN(_EX) = 1

then

STATE_TRIP_IVVT_IN(_EX)_i = "POS_ST"

LV_TRIP_ACT_IVVT_IN(_EX) = 0

endif

LV_CDN_DIAG_TRIP_IVVT_IN(_EX)_i = 0

Recurrence:

100 ms

Activation:

(LV_DIAG_TRIP_AVL_IVVT_IN(_EX) = 1

or LC_DIAG_TRIP_MAN_IVVT_IN(_EX) = 1)

and LV_ACT_IVVT = 1

and LC_CAM_SP_MAN_IVVT_IN(_EX) = 0

and LV_ERR_LOAD_TPS_PLAUS = 0

and LV_ERR_TPS = 0

and LV_ERR_ISA_i = 0

and LV_MTC_CUR_OFF = 0

and EOL_STATE_DIAGCP <> 1h (EOL_ACT)

Formula section:

Short trip algorithm:

State "POS_ST":

T_TRIP_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT

CAM_SP_TRIP_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

LV_TRIP_ACT_IVVT_IN(_EX) = 1

If LV_TRIP_ACT_IVVT_IN(_EX) = 1

and | N_SP_IS_EOL – N | < C_N_SP_IS_DIF_EOL_IVVT

then STATE_TRIP_IVVT_IN(_EX)_i = "POS_0"

endif

State "POS_0":

if T_DLY_MES_IVVT - T_TRIP_ST_IVVT_IN(_EX)_i <
C_T_ADJ_TRIP_IVVT_IN(_EX)

then

if Inlet:

CAM_AV_IVVT_IN_1 >= C_CAM_INI_IN - C_CAM_HYS_TRIP_IVVT

Exhaust:

CAM_AV_IVVT_EX_i <= C_CAM_INI_EX + C_CAM_HYS_TRIP_IVVT

then *Position 0 reached*

STATE_TRIP_IVVT_IN(_EX)_i = "POS_1"

T_TRIP_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT

CAM_SP_TRIP_IVVT_IN(_EX)_i = C_CAM_SP_TRIP_IVVT_IN(_EX)

endif

else

STATE_TRIP_IVVT_IN(_EX)_i = "POS_ERR"

endif

State "POS_1":

if T_DLY_MES_IVVT - T_TRIP_ST_IVVT_IN(_EX)_i <

C_T_ADJ_TRIP_IVVT_IN(_EX)

then

if Inlet:

CAM_AV_IVVT_IN_i <=

C_CAM_SP_TRIP_IVVT_IN + C_CAM_HYS_TRIP_IVVT

Exhaust:

CAM_AV_IVVT_EX_i >=

C_CAM_SP_TRIP_IVVT_EX - C_CAM_HYS_TRIP_IVVT

then *Position 1 reached*

STATE_TRIP_IVVT_IN(_EX)_i = "POS_2"

T_TRIP_ST_IVVT_IN(_EX)_i = T_DLY_MES_IVVT

CAM_SP_TRIP_IVVT_IN(_EX)_i = C_CAM_INI_IN(_EX)

endif

else

STATE_TRIP_IVVT_IN(_EX)_i = "POS_ERR"

endif

State "POS_2":

if T_DLY_MES_IVVT - T_TRIP_ST_IVVT_IN(_EX)_i <

C_T_ADJ_TRIP_IVVT_IN(_EX)

then

if Inlet:

CAM_AV_IVVT_IN_i >= C_CAM_INI_IN - C_CAM_HYS_TRIP_IVVT

Exhaust:

CAM_AV_IVVT_EX_i <= C_CAM_INI_EX + C_CAM_HYS_TRIP_IVVT

then *Position 2 reached*

STATE_TRIP_IVVT_IN(_EX)_i = "POS_END"

LV_CDN_DIAG_TRIP_IVVT_IN(_EX)_i = 1

ERR_SYM_TRIP_IVVT_IN(_EX)_i = SYM_0 "inactive"

LV_ERR_TRIP_IVVT_IN(_EX)_i = 0

endif

else

STATE_TRIP_IVVT_IN(_EX)_i = "POS_ERR"

endif

State "POS_ERR":

STATE_TRIP_IVVT_IN(_EX)_i = "POS_END"

LV_CDN_DIAG_TRIP_IVVT_IN(_EX)_i = 1

ERR_SYM_TRIP_IVVT_IN(_EX)_i = SYM_0 "active"

LV_ERR_TRIP_IVVT_IN(_EX)_i = 1

State "POS_END":

LV_END_DIAG_TRIP_IVVT_IN(_EX)_i = 1

LV_CDN_DIAG_TRIP_IVVT_IN(_EX)_i = 0

<u>Short trip end:</u>

NC_NR_CBK_IVVT = 1:

if STATE_TRIP_IVVT_IN(_EX)_1 = "POS_END"

then

LV_TRIP_ACT_IVVT_IN(_EX) = 0

LV_TRIP_END_IVVT_IN(_EX) = 1

endif

NC_NR_CBK_IVVT = 2:

if STATE_TRIP_IVVT_IN(_EX)_1 = "POS_END"

and (STATE_TRIP_IVVT_IN(_EX)_2 = "POS_END"

or LV_NOT_ADJ_CAM_IVVT_IN(_EX)_2 = 1)

then

LV_TRIP_ACT_IVVT_IN(_EX) = 0

LV_TRIP_END_IVVT_IN(_EX) = 1

endif

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_DIAG_TRIP_MAN_IVVT_IN	1	0...1H	0...1	1	-
1 = manual short trip start by application, 0 = no manual short trip; inlet					
LC_DIAG_TRIP_MAN_IVVT_EX	1	0...1H	0...1	1	-
1 = manual short trip start by application, 0 = no manual short trip; exhaust					
C_T_ADJ_TRIP_IVVT_IN	1	0...FFFFH	0...655350	10	ms
Maximum allowed time for adjustment between two short trip positions; inlet					
C_T_ADJ_TRIP_IVVT_EX	1	0...FFFFH	0...655350	10	ms
Maximum allowed time for adjustment between two short trip positions; exhaust					
C_CAM_SP_TRIP_IVVT_IN	1	0...FFH	60...155.625	0.375	°CRK
Short trip camshaft position 1, away from reference position; inlet					
C_CAM_SP_TRIP_IVVT_EX	1	0...FFH	-40.125...-135.75	0.375	°CRK
Short trip camshaft position 1, away from reference position; exhaust					
C_CAM_HYS_TRIP_IVVT	1	0...A0H	0...60	0.375	°CRK
Hystersis for reaching short trip position					
C_N_SP_IS_DIF_EOL_IVVT	1	0000...1FE0H	0...8160	1	rpm
Idle deviation from idle setpoint to start IVVT_EOL check					

J.3 IVVT Short Trip (Forced Diagnosis) - Application Incidences

Output data:

Name	Mode	Hex. Limit	Phys. Limit	Resol.	Unit
LV_INH_DIAG_TRIP_IVVT	V	0...1H	0...1	1	-
1 = short trip inhibited due to non-fulfillment of condition for it, 0 = enabled					
LV_DIAG_TRIP_AVL_IVVT_IN	O/V	0...1H	0...1	1	-
1 = short trip is available, 0 = not available					
LV_DIAG_TRIP_AVL_IVVT_EX	O/V	0...1H	0...1	1	-
1 = short trip is available, 0 = not available					

Input data:

N_32	LV_ERR_TOIL_STUCK	VB	LV_ERR_IV
LV_MTC_CUR_OFF	LV_OFF_MTC_MON	LV_ACT_IVVT	LV_ERR_TOIL
LC_CAM_SP_MAN_IVVT_IN	LC_CAM_SP_MAN_IVVT_EX	NLC_IVVT_IN	NLC_IVVT_EX
LV_ERR_TOIL_PLAUS_H	LV_ERR_TOIL_PLAUS_L	LV_ERR_LIH_IVVT	TOIL

FUNCTION DESCRIPTION:

Description:

The short trip is a special kind of IVVT forced diagnosis. There is no error management for it.

Application conditions:

Initialization: *At reset:* LV_INH_DIAG_TRIP_IVVT = 1
LV_DIAG_TRIP_AVL_IVVT_IN(_EX) = 0

Recurrence: *100 ms*

Activation: *At every engine state*

Formula section:

Conditions for short trip:

if LC_DIAG_TRIP_EXT_TEST_ACT_IVVT = 1
and LV_ACT_IVVT = 1
and C_N_32_MIN_TRIP_IVVT <= N_32 <= C_N_32_MAX_TRIP_IVVT
and C_TOIL_MIN_TRIP_IVVT <= TOIL <= C_TOIL_MAX_TRIP_IVVT
and VB >= C_VB_DIAG_TRIP_IVVT
and LV_ERR_TOIL = 0
and LV_ERR_TOIL_PLAUS_H = 0
and LV_ERR_TOIL_PLAUS_L = 0
and LV_ERR_TOIL_STUCK = 0
and LV_ERR_IV = 0
and LV_MTC_CUR_OFF = 0
and LV_OFF_MTC_MON = 0

and LV_ERR_LIH_IVVT = 0
and LC_CAM_SP_MAN_IVVT_IN = 0 *(NLC_IVVT_IN = 1)*
and LC_CAM_SP_MAN_IVVT_EX = 0 *(NLC_IVVT_EX = 1)*
then LV_INH_DIAG_TRIP_IVVT = 0
else LV_INH_DIAG_TRIP_IVVT = 1

Short trip activation:

if LV_INH_DIAG_TRIP_IVVT = 0
and LC_DIAG_TRIP_EXT_REQ_ST_IVVT = 1 **or** request by external tester
then LV_DIAG_TRIP_AVL_IVVT_IN(_EX) = 1
else LV_DIAG_TRIP_AVL_IVVT_IN(_EX) = 0

IVVT end-of-line test error detection:

Two symptoms can be distinguished: Target position not reached in specified time intake/exhaust. Both symptoms are handled by direct error entry.

Configuration for diagnositic symptoms:

Diagnostic	Symptom Description	Symptom	Filter type
TRIP_IVVT_IN_1	Trip IVVT	SYM_0	*STD*
		SYM_1	
		SYM_2	
		SYM_3	

List of Environmental Data to store in failure memory: DIAG_INST_ENVD
CAM_SP_IVVT_IN_1
CAM_IN_1
CAM_DIF_INT_DIAG_IVVT_IN_1_ENVD

Calibration data:

Name	Dim	Hex. Limit	Phys. Limit	Resol.	Unit
LC_DIAG_TRIP_EXT_TEST_ACT_IVVT	1	0...1H	0...1	1	-
1 = requirement for short trip activation by application, 0 = no requirement					
C_N_32_MIN_TRIP_IVVT	1	0...FFH	0...8160	32	rpm
Minimum engine speed for short trip					
C_N_32_MAX_TRIP_IVVT	1	0...FFH	0...8160	32	rpm
Maximum engine speed for short trip					
C_TOIL_MIN_TRIP_IVVT	1	0...C8H	-40...160	1	°C
Minimum oil temperature for short trip					
C_TOIL_MAX_TRIP_IVVT	1	0...C8H	-40...160	1	°C
Maximum oil temperature for short trip					
C_VB_DIAG_TRIP_IVVT	1	0...FFH	0...26	0.1	V
Minimum battery voltage for short trip					
LC_DIAG_TRIP_EXT_REQ_ST_IVVT	1	0...1H	0...1	1	-
1 = requirement for short trip start by application, 0 = no requirement					

Index of All Chapters

A

D

E

G

M

N

S

W

X