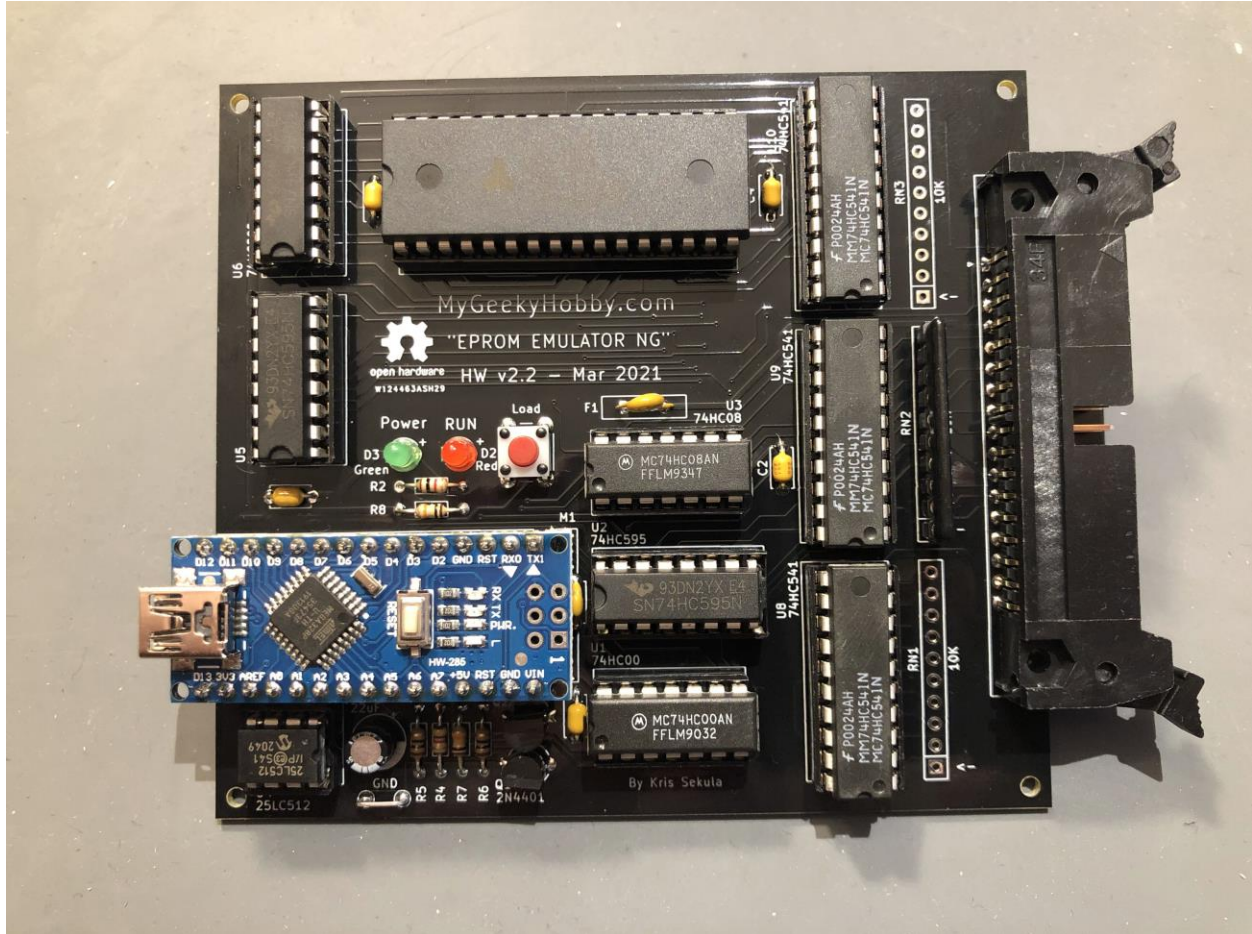


EPROM EMU NG – Assembly Instructions.

For PCB version 2.2



MyGeekyHobby.com – Apr 2021

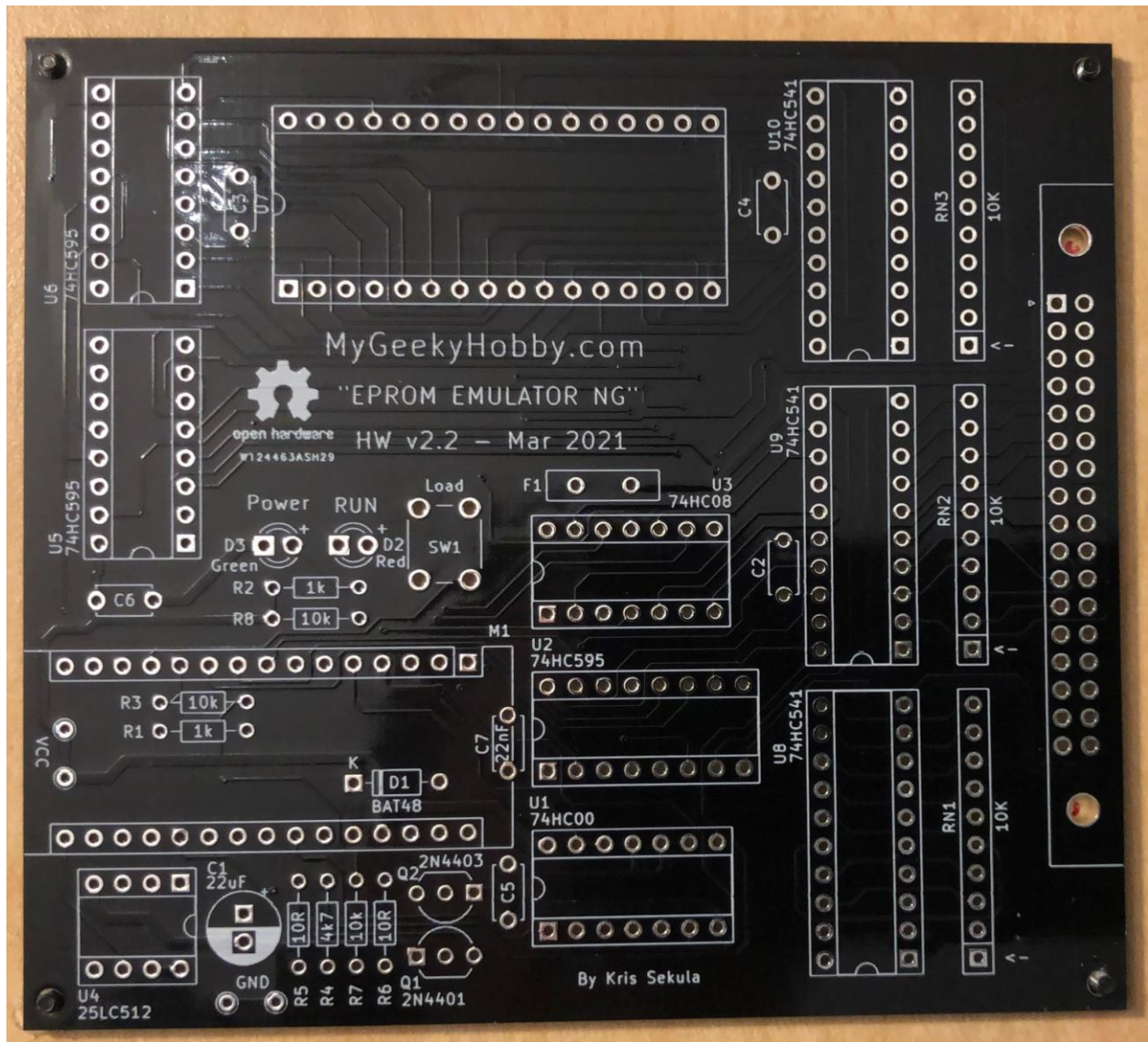
Document version: 2.2(A)

Thank you for purchasing the “EPROM EMU NG” DIY Kit, following is a brief assembly instruction.

This document is being updated regularly, check my GitHub page for updates:

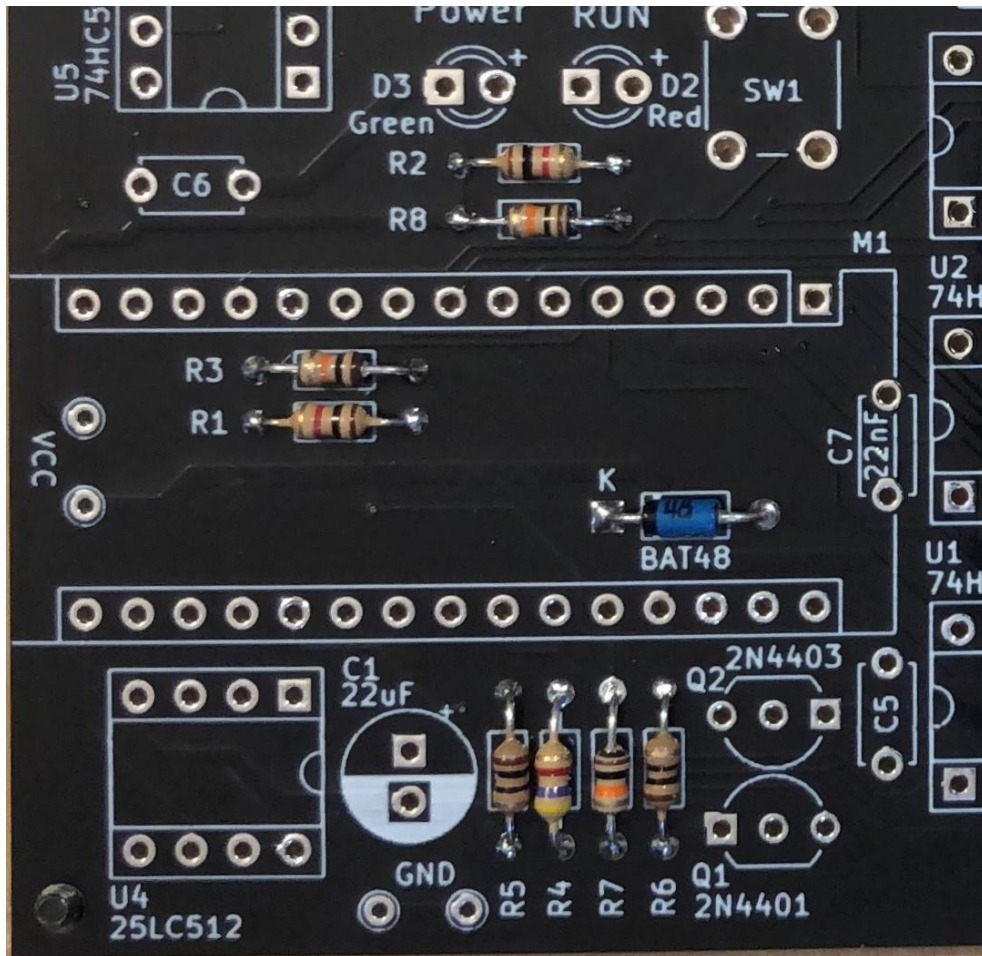
<https://github.com/Kris-Sekula/EPROM-EMU-NG>

Before you begin, familiarize yourself with locations of components on the blank PCB

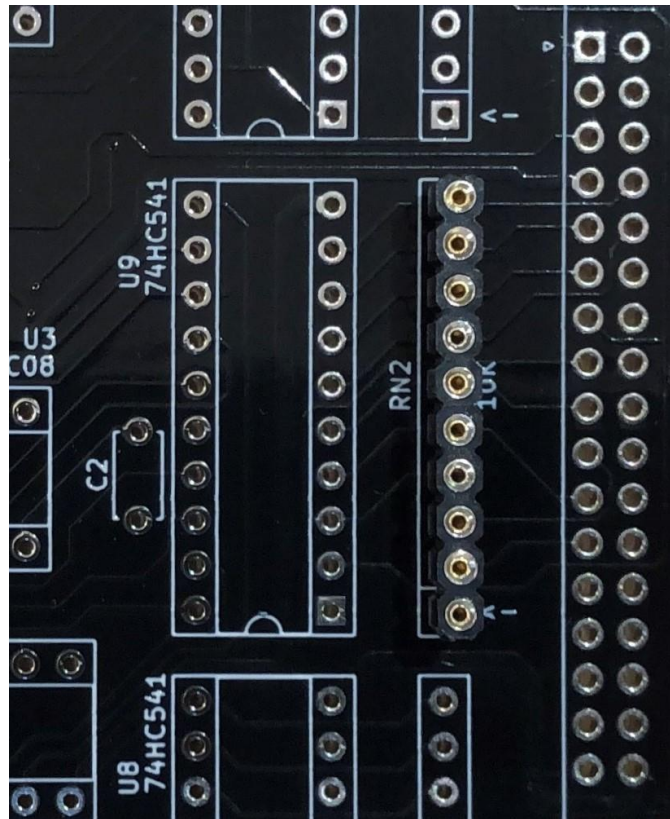


Solder components starting with the shortest.

1. Populate resistors R1, R2 – 1k, resistors R3, R7, R8 – 10k, resistor R4 – 4.7k, resistors R5,R6 – 10R and diode D4 (BAT48):



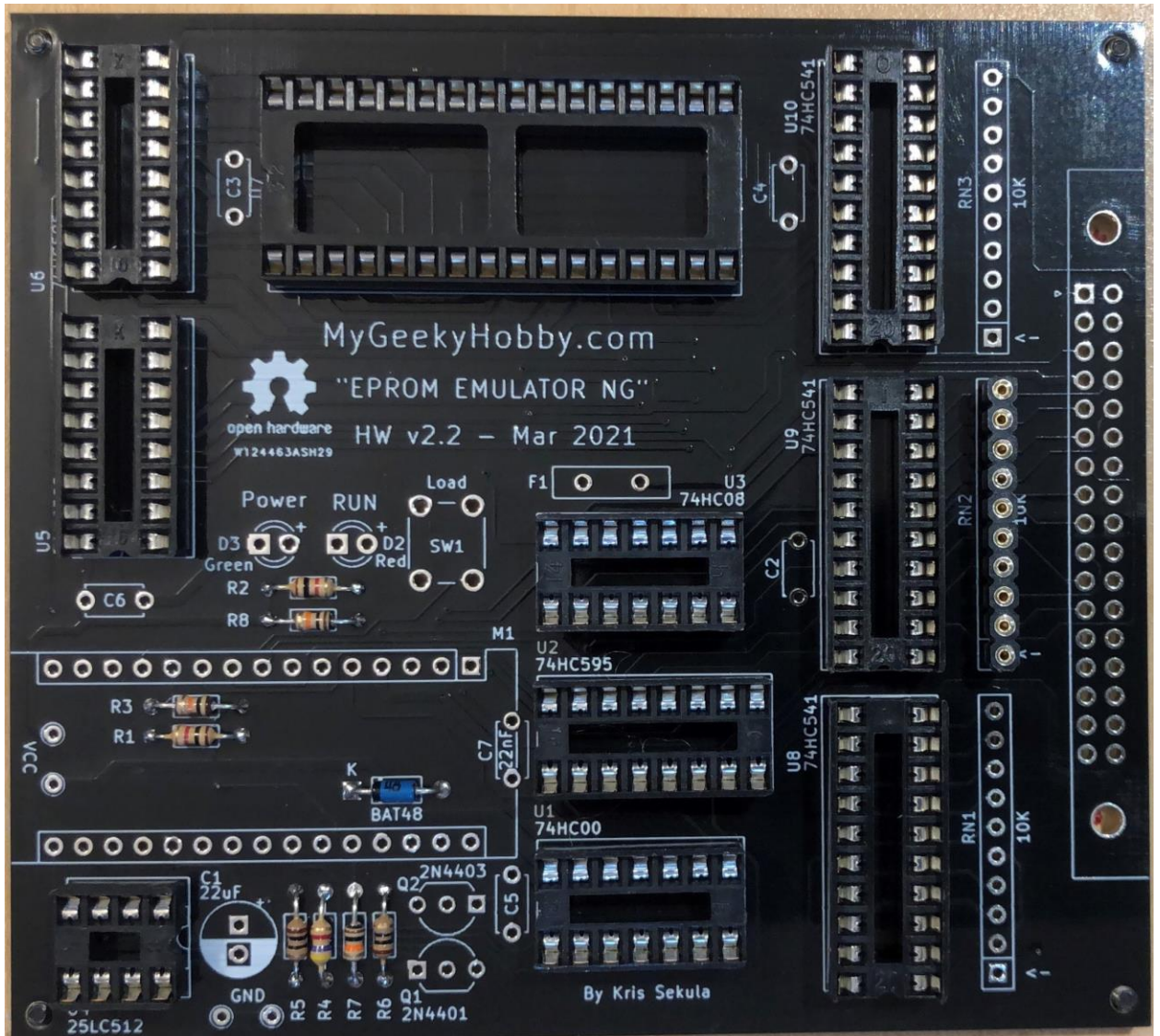
2. Populate the precision socket for resistor network RN2



Useful tip:

Solder **one** of the middle pins first, this will give you opportunity to check if the socket is straight, and adjust if needed before you solder all other pins.

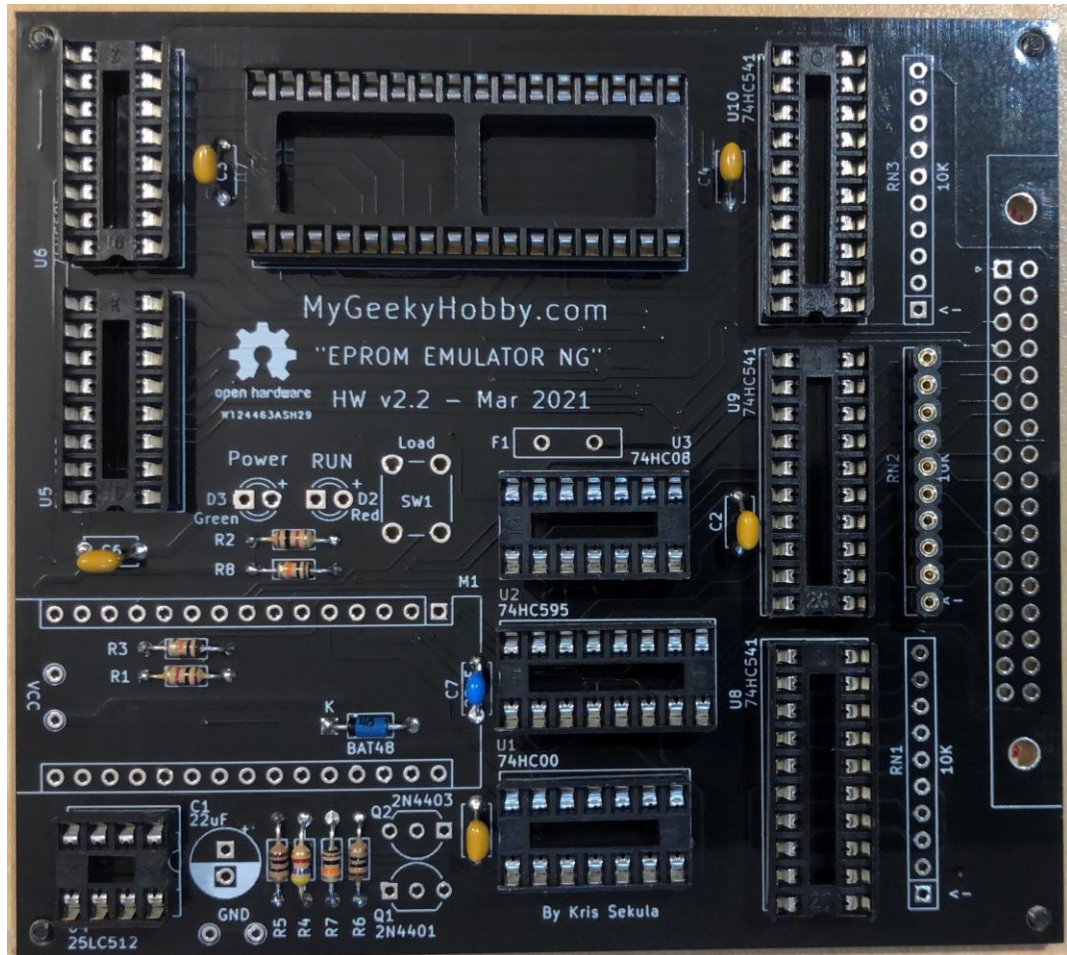
3. Populate all integrated circuit sockets:



Useful Tip:

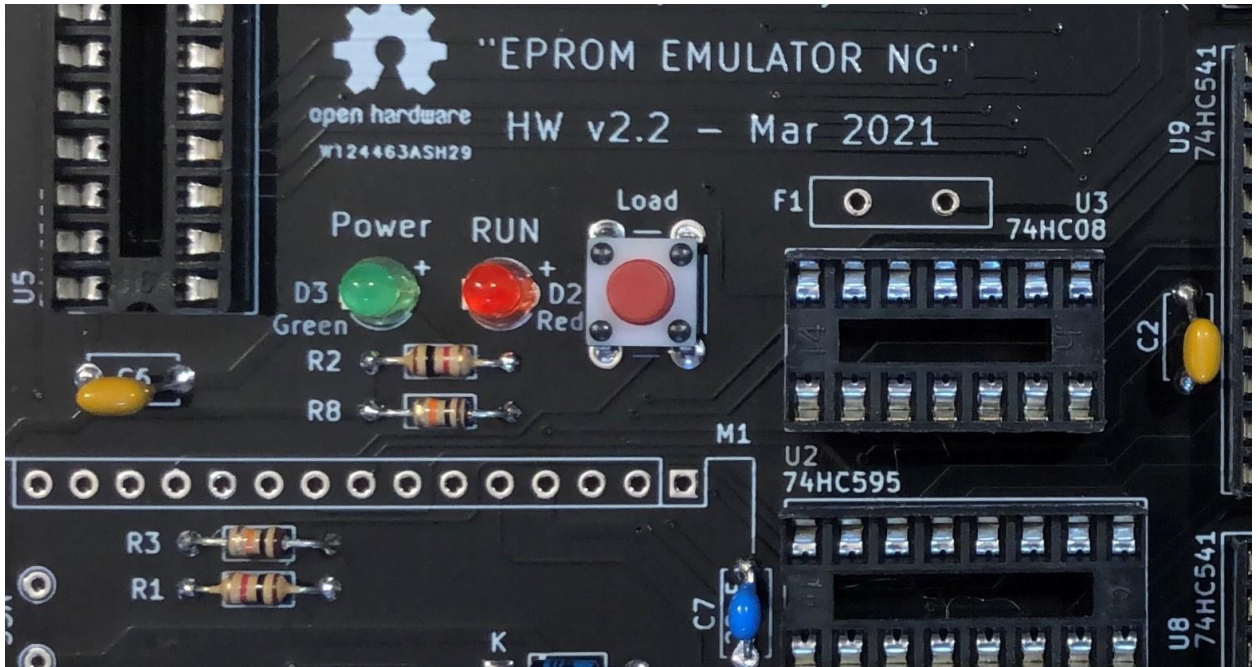
You can insert all sockets into the PCB, using the 3d printed base plate hold all sockets in place while you “flip” the board. Start by soldering **two** pins of each socket (opposite pins for example 7 and 14). Turn the PCB back and double check if all sockets are oriented correctly, if yes – solder the remaining pins. If not, it will be much easier to fix it at this point.

4. Populate decoupling capacitors, C7 – 22nF, C2, C3, C4, C5, C6 – 100nF



Note: only C7 has its value (22nF) written in the silkscreen, the rest of the decoupling capacitors don't, and are simply 100nF.

5. Populate the LEDs and the push button:



Useful Tip:

The longer lead of the LED should go to the hole marked with “+”.

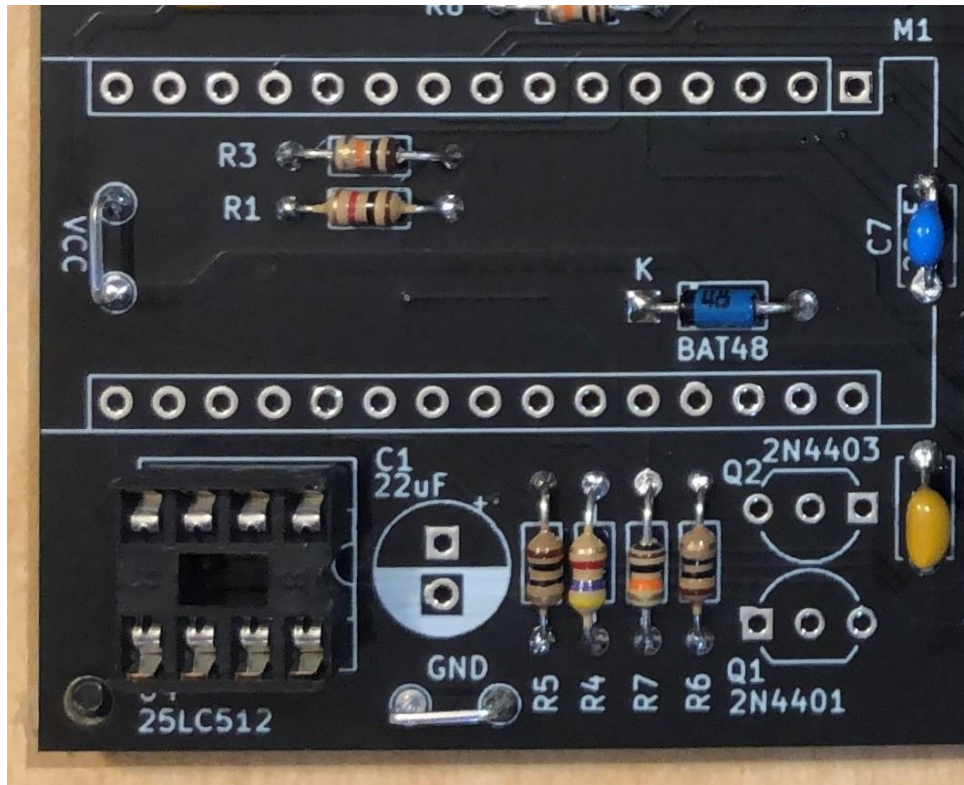
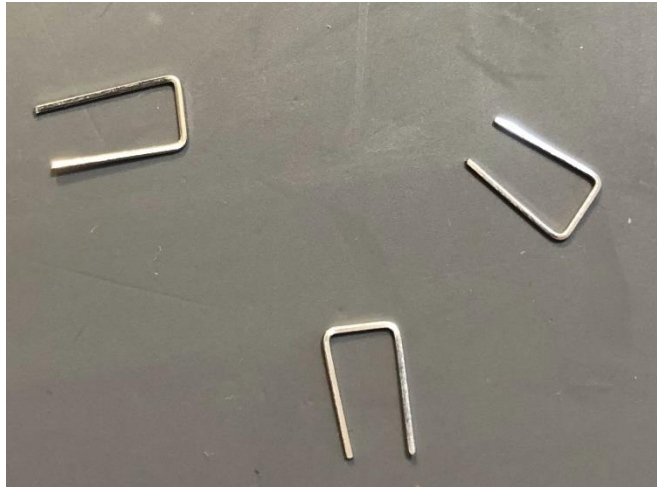
Keep the cut-offs from the LED as you will need them in the next step

Note:

The current firmware of the emulator uses the push button in the following way:

- a) Short press – loads the content of the SPI EEPROM into the SRAM
- b) Long press – resets the target device (if the reset hooks are attached)

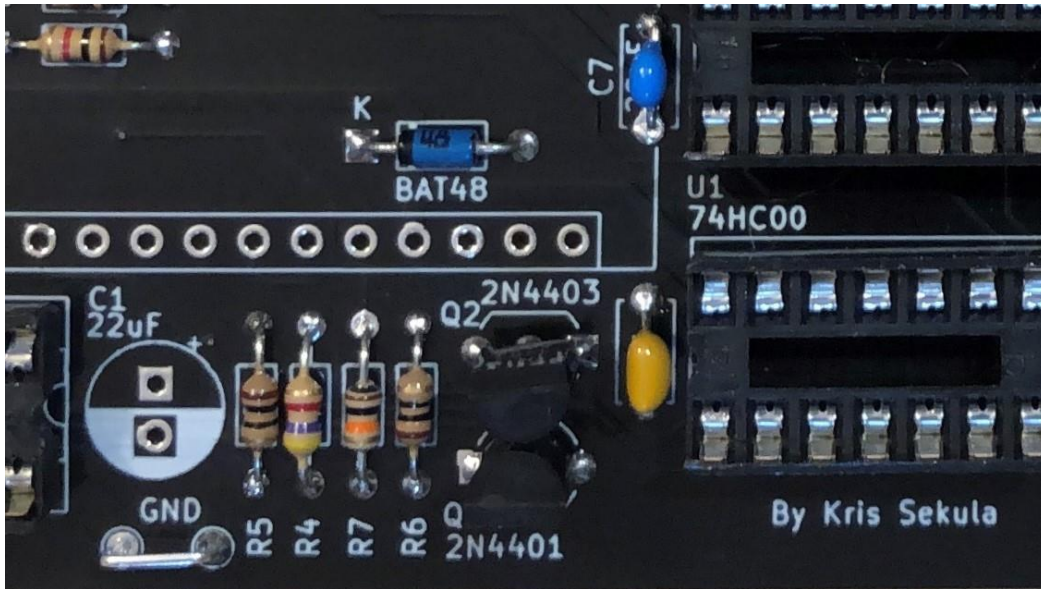
6. Use the LED cut-offs to populate the GND and VCC test points:



Useful Tip:

Those test points are “handy” place to attach ground from a logic probe or multimeter during firmware development.

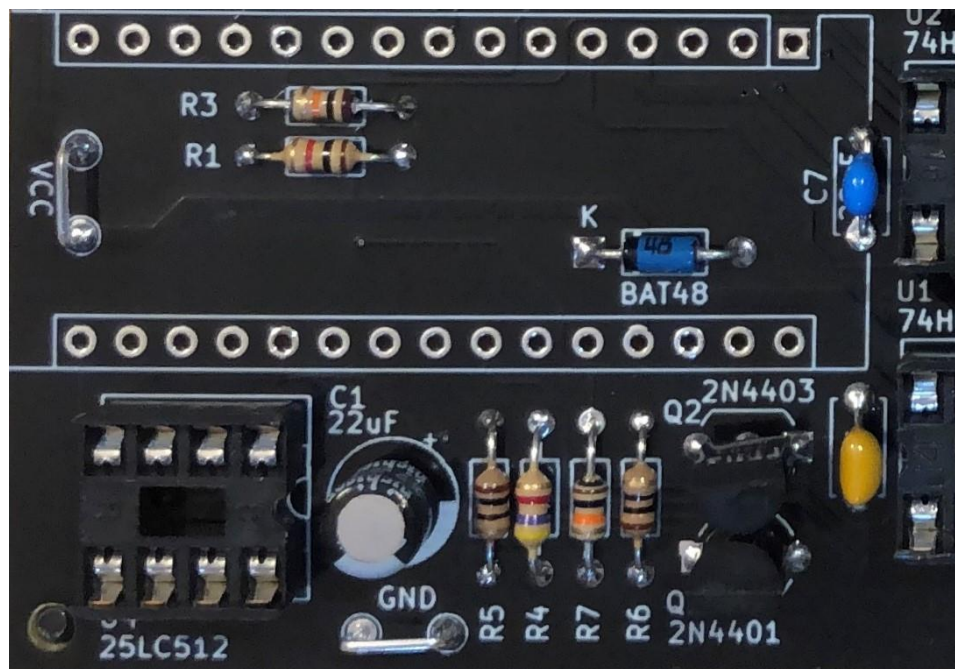
7. Populate transistors Q1 – 2N4401 and Q2 – 2N4403:



Useful tip:

Solder just the middle pin of the transistors, check if the transistor “stands” straight and adjust if needed before you solder the other two pins.

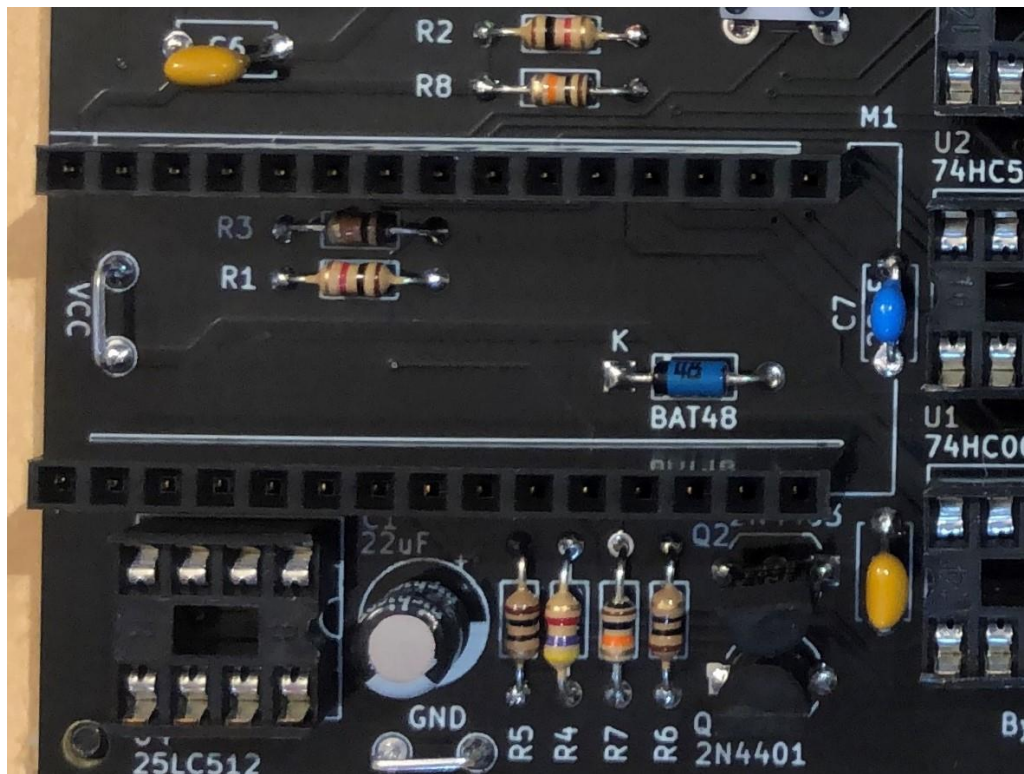
8. Populate C1 – 22uF/16V



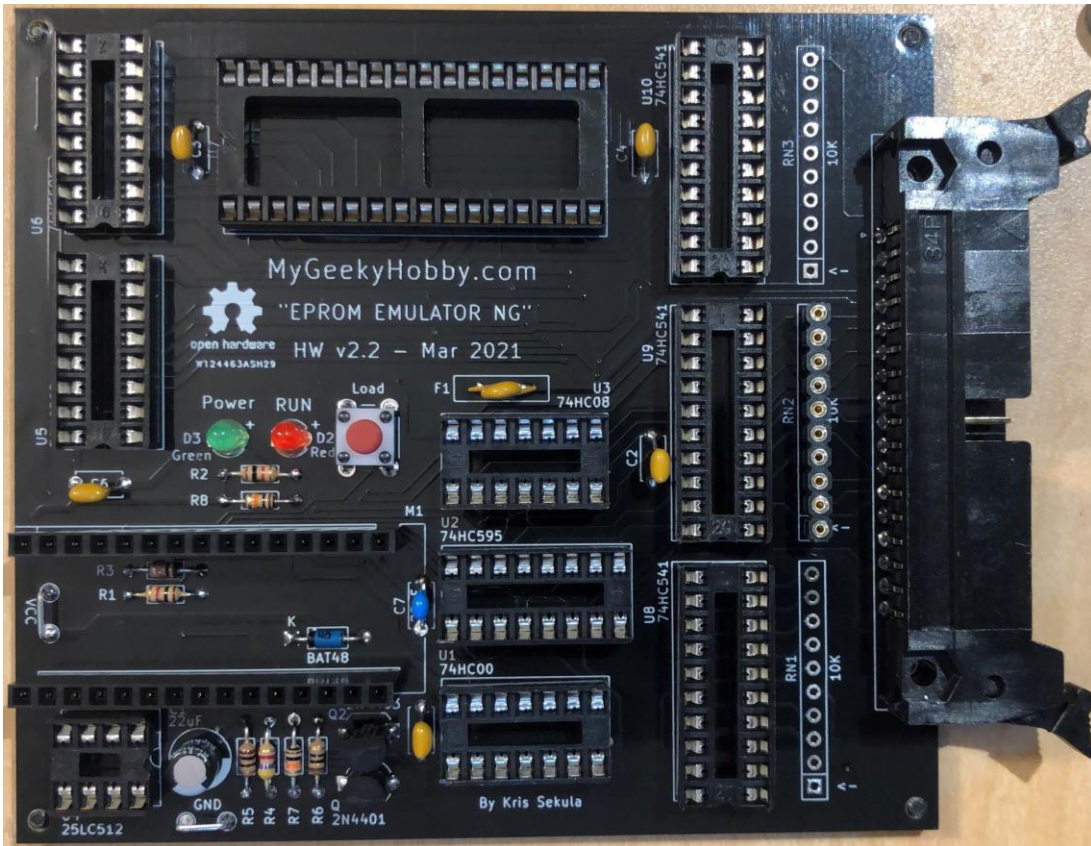
9. Populate the polyfuse F1 – 50mA:



10. Populate the Arduino Nano sockets M1a/M1b:



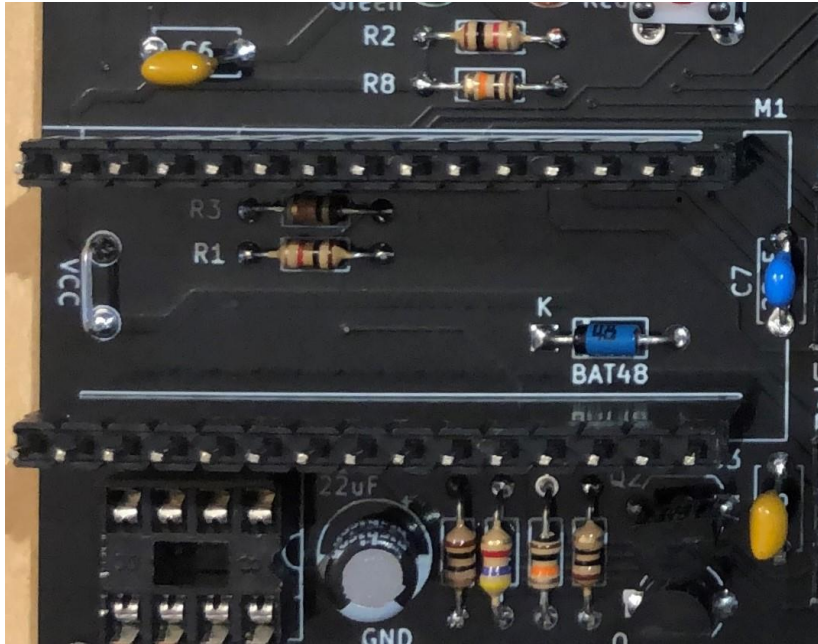
11. Populate the emulator probe connector:



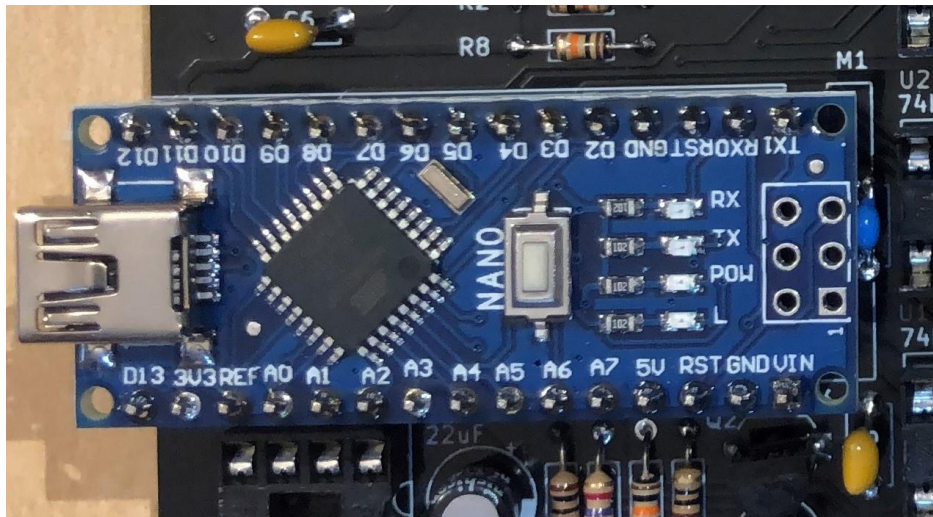
Useful Tip:

When soldering the 34 pin main connector, start with just two pins on opposite side of the connector, turn the PCB and check if the connector is flat with the PCB, adjust if needed and solder the rest of the pins.

12. Solder the Arduino pins, you can first insert the pins into sockets in the PCB before soldering the Arduino module:

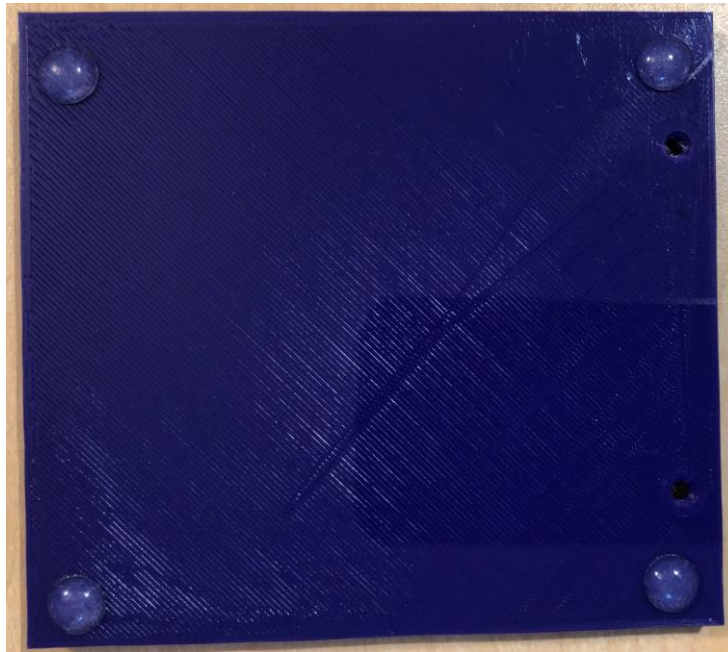


Be careful not to misalign the module, make sure all pins are in correct position before you start soldering:

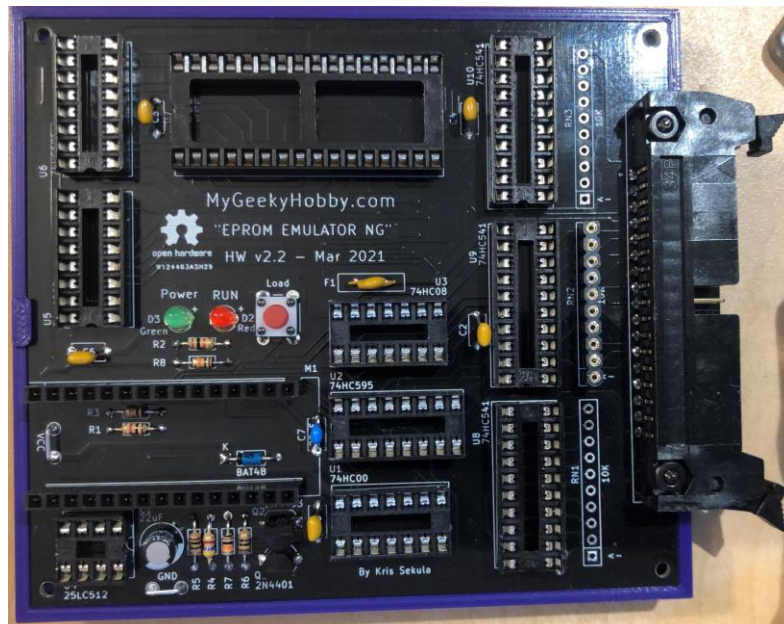


At this point all soldering is finished, you should clean the PCB and the Arduino module with 99% Isopropyl alcohol or similar PCB cleaner.

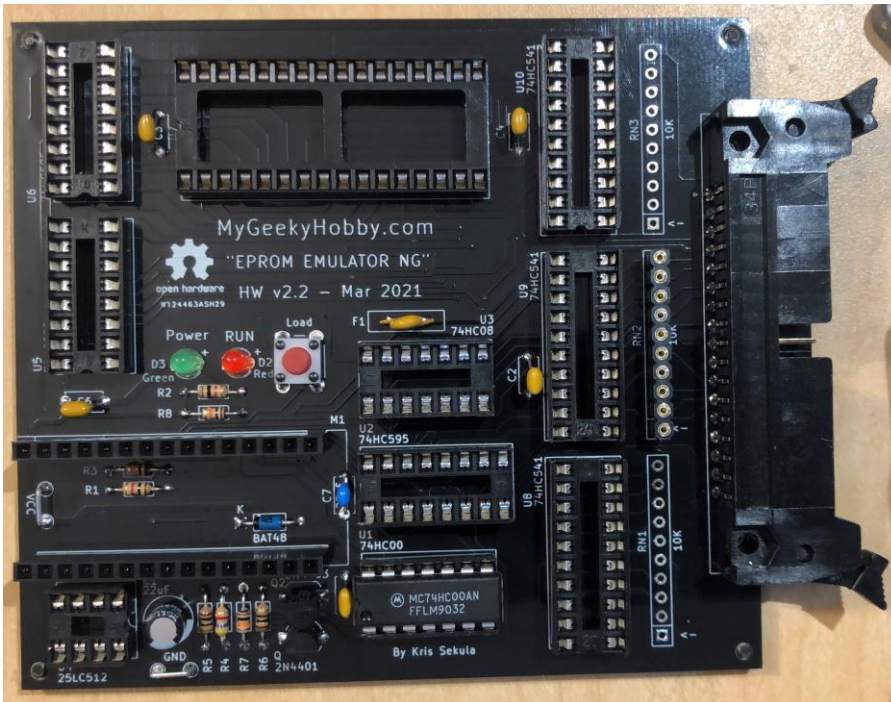
13. Prepare the base plate by adding the rubber feet:



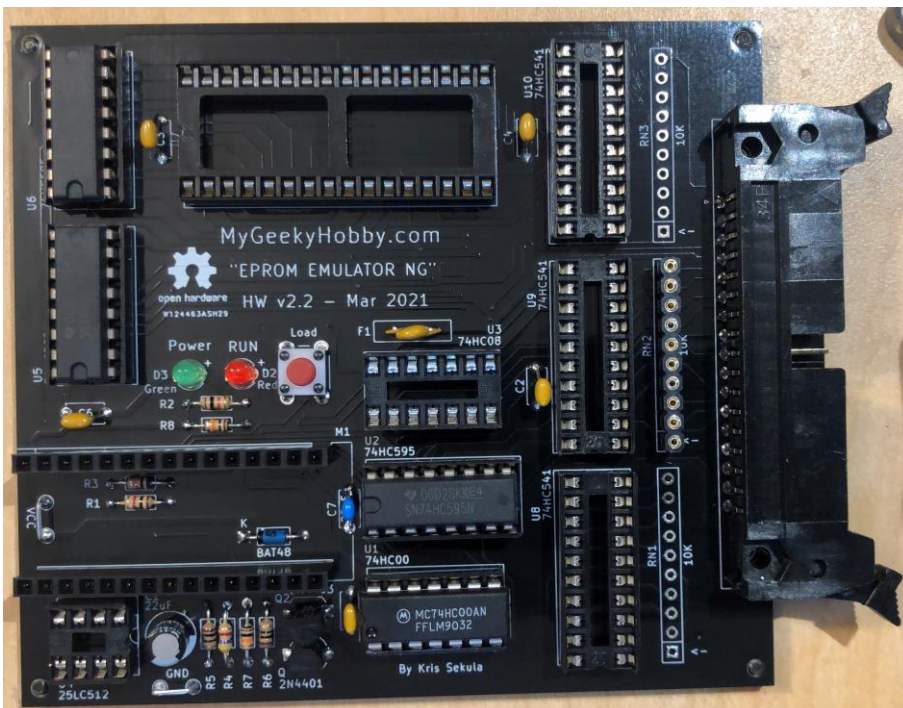
14. Secure the PCB into the base plate using two sets of screws and nuts, the PCB is now ready to populate the components:



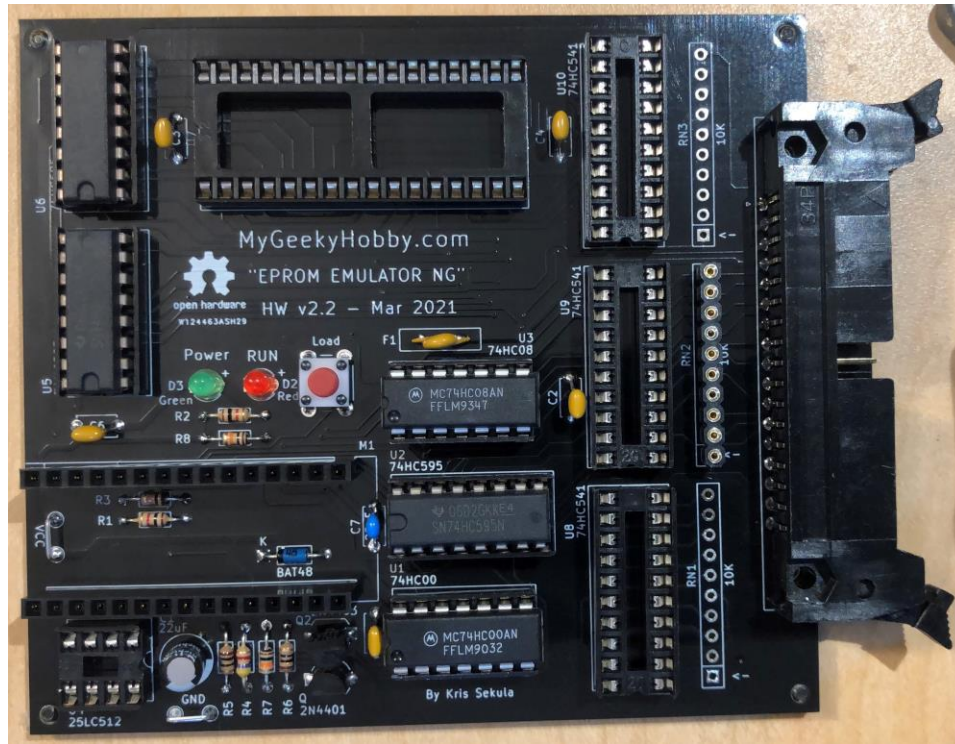
15. Insert U1 – 74HC00 (NAND gate):



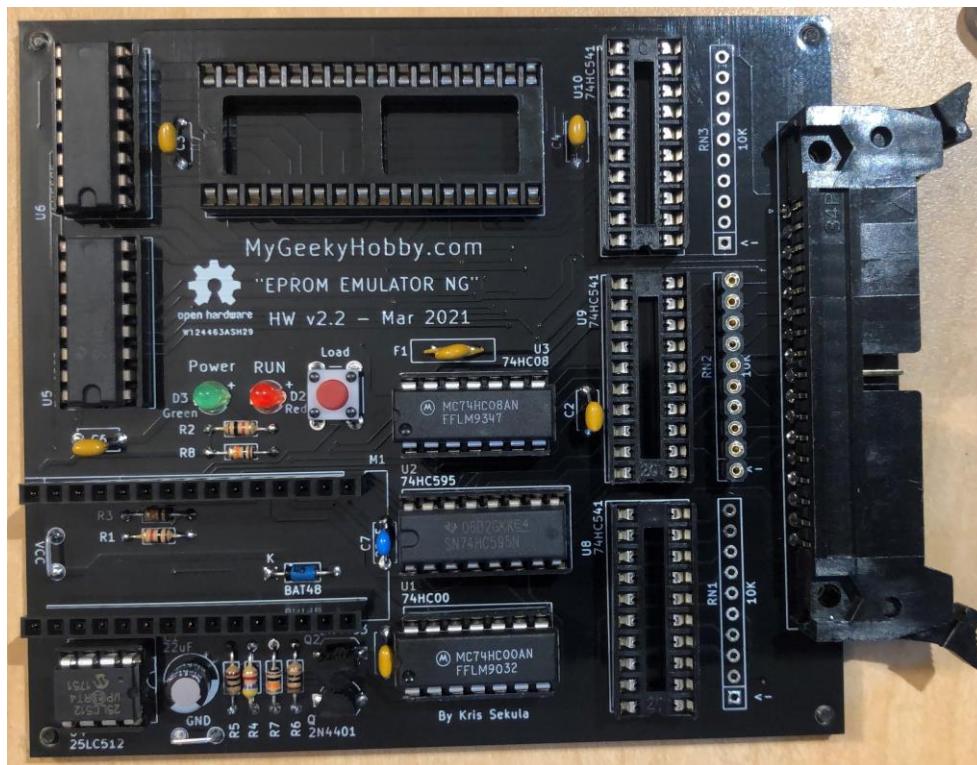
16. Insert U3, U5, U6 – 74HC595 (Shift register):



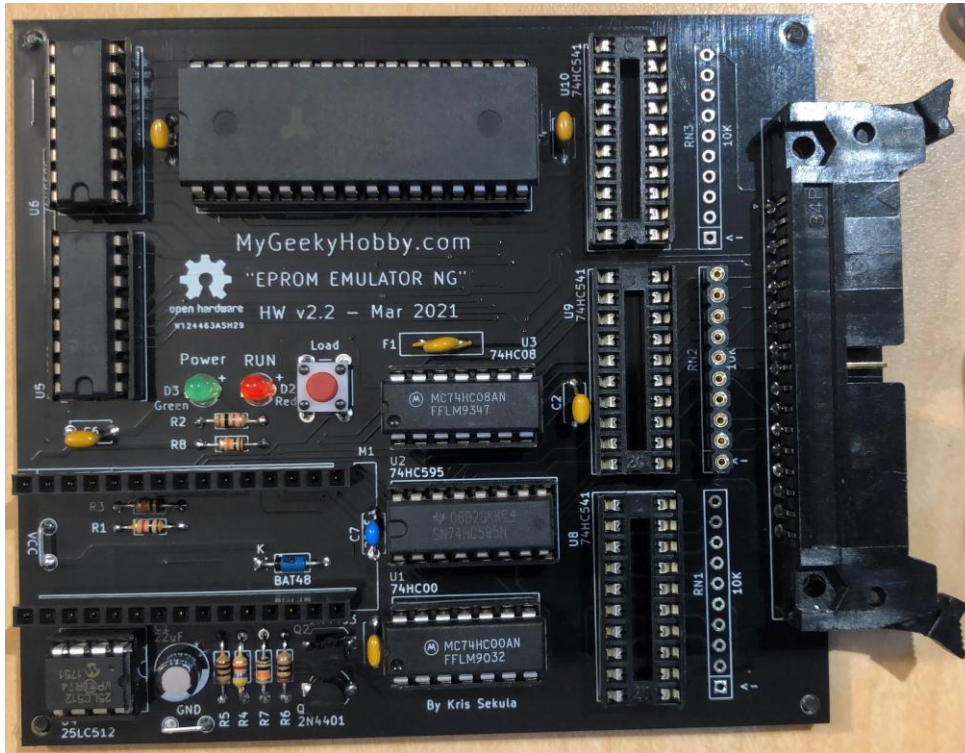
17. Insert U3 – 74HC08 (AND GATE):



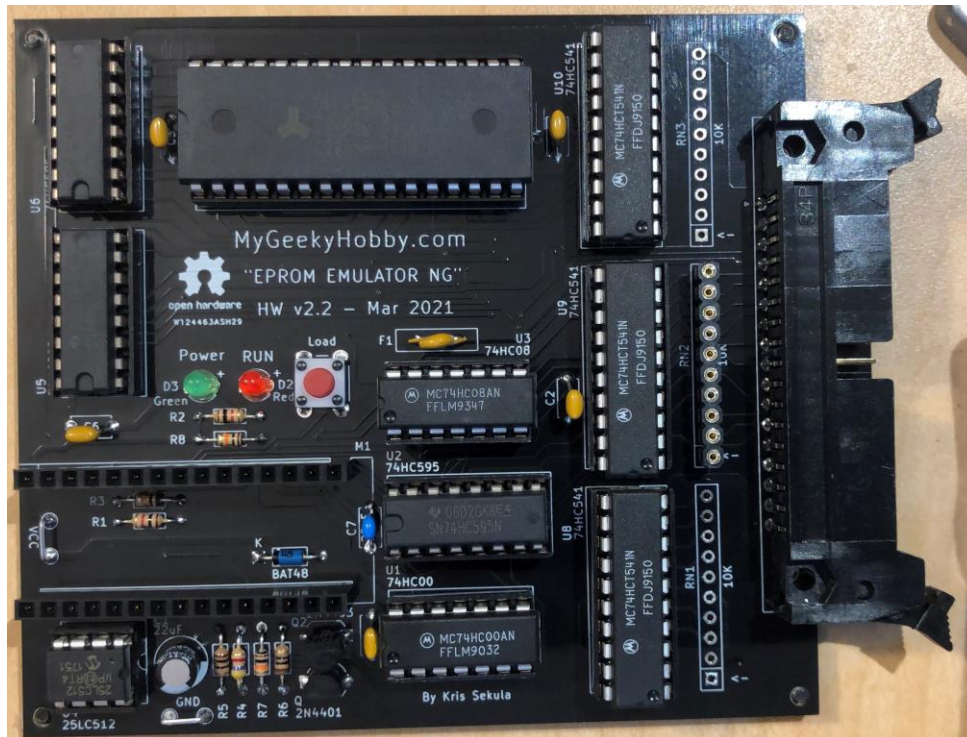
18. Inset U4 – 25LC512 (SPI EEPROM):



19. Insert U7 – AS6C1008 (SRAM):

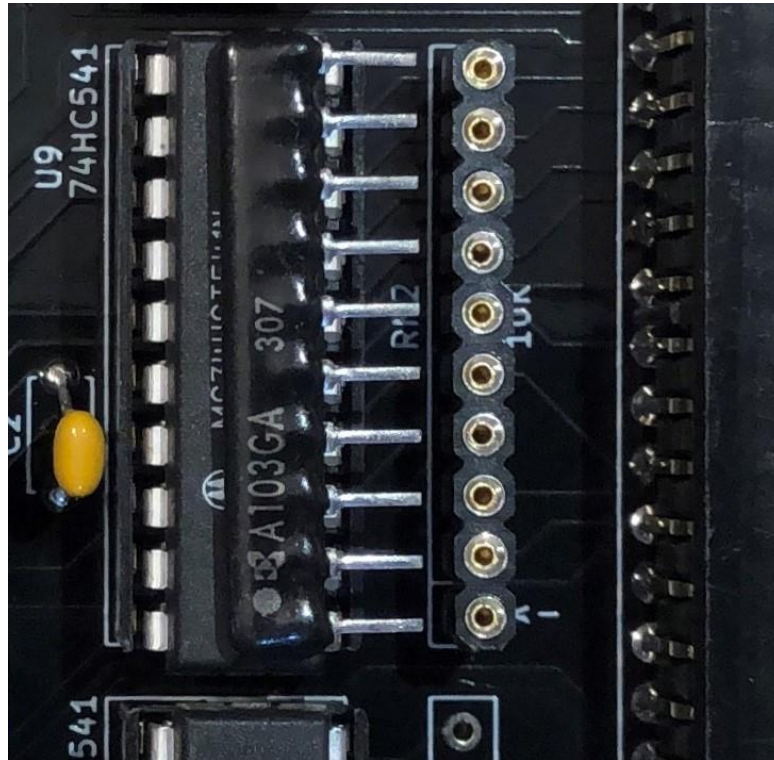


20. Insert U8, U9, U10 – 74HC541 (3 state buffer):

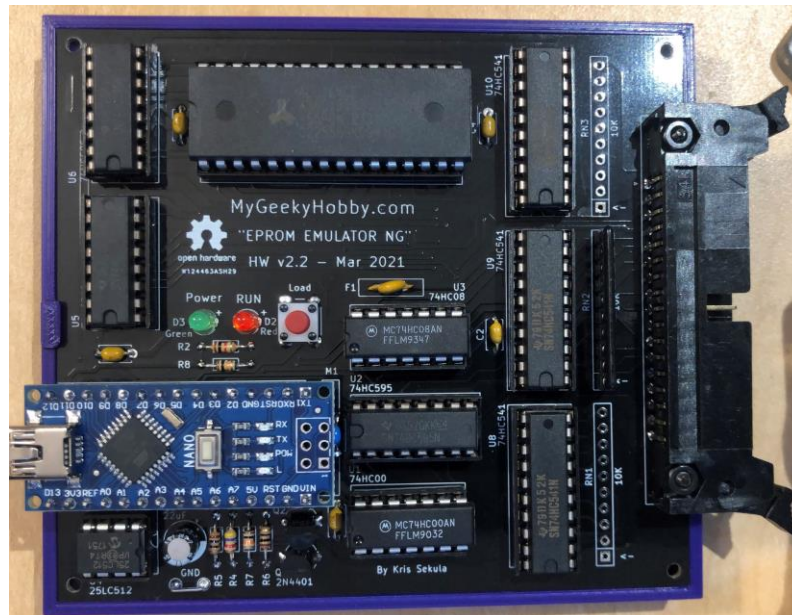


21. Insert the resistor network RN2 (10k):

Pay attention to the orientation of the resistor network, pin 1 is marked with a “dot” and on the PCB, pin one has an arrow pointing at it “<-“:



22. Insert the Arduino Nano module:



At this point your “EPROM EMULATOR NG” is assembled, you should program the Arduino Nano with the recommended version of firmware (Arduino sketch) and use it with compatible version of software (python script).

Legal Notice:

THE SOFTWARE, DESIGN AND/OR HARDWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.