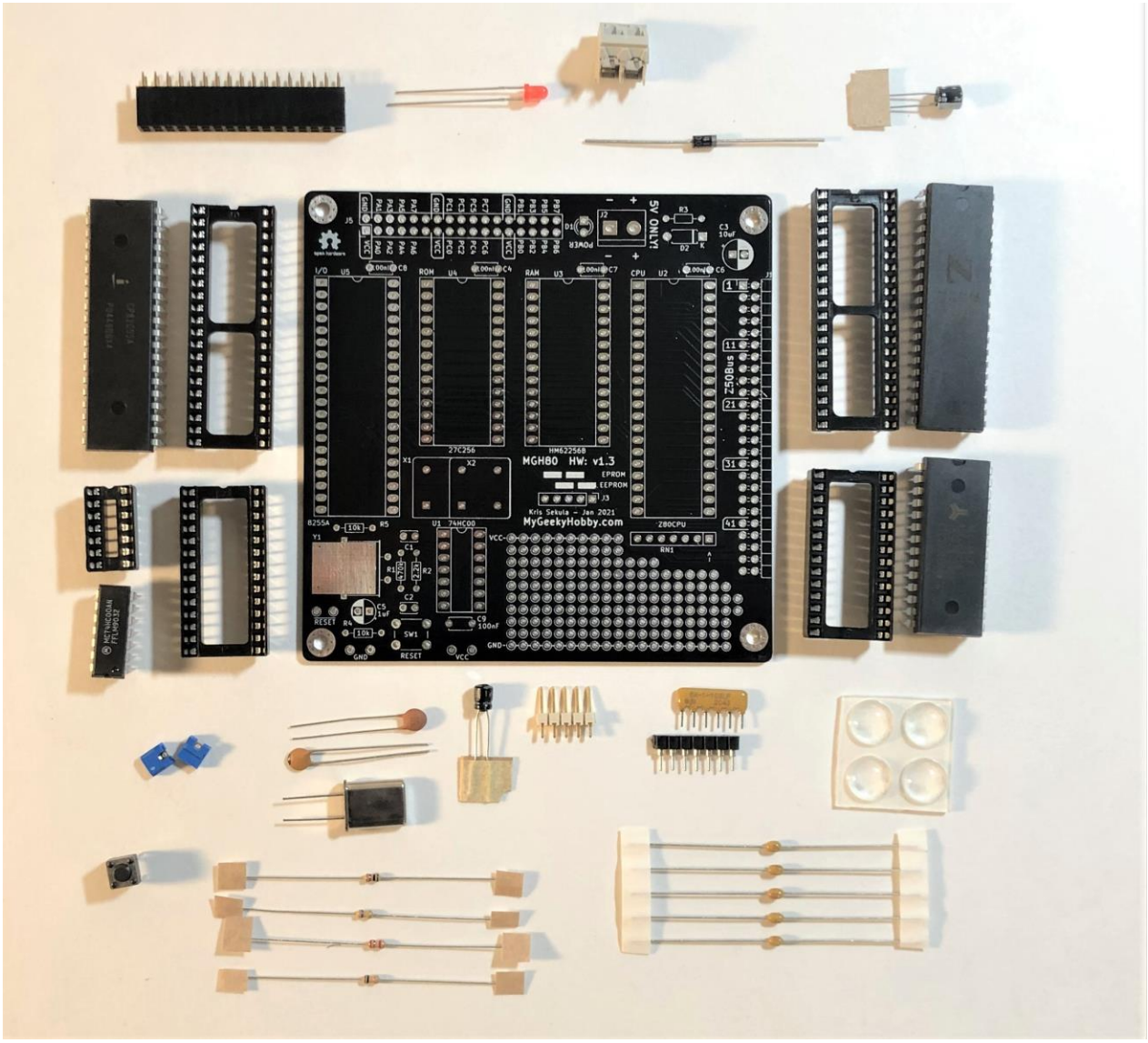
“MGH80” Z80 controller DIY kit – Assembly Instructions.




MyGeekyHobby.com ver 1.3.1 – March 2021

Thank you for purchasing the “MGH80” Z80 Controller DIY Kit, following is a brief assembly instruction.

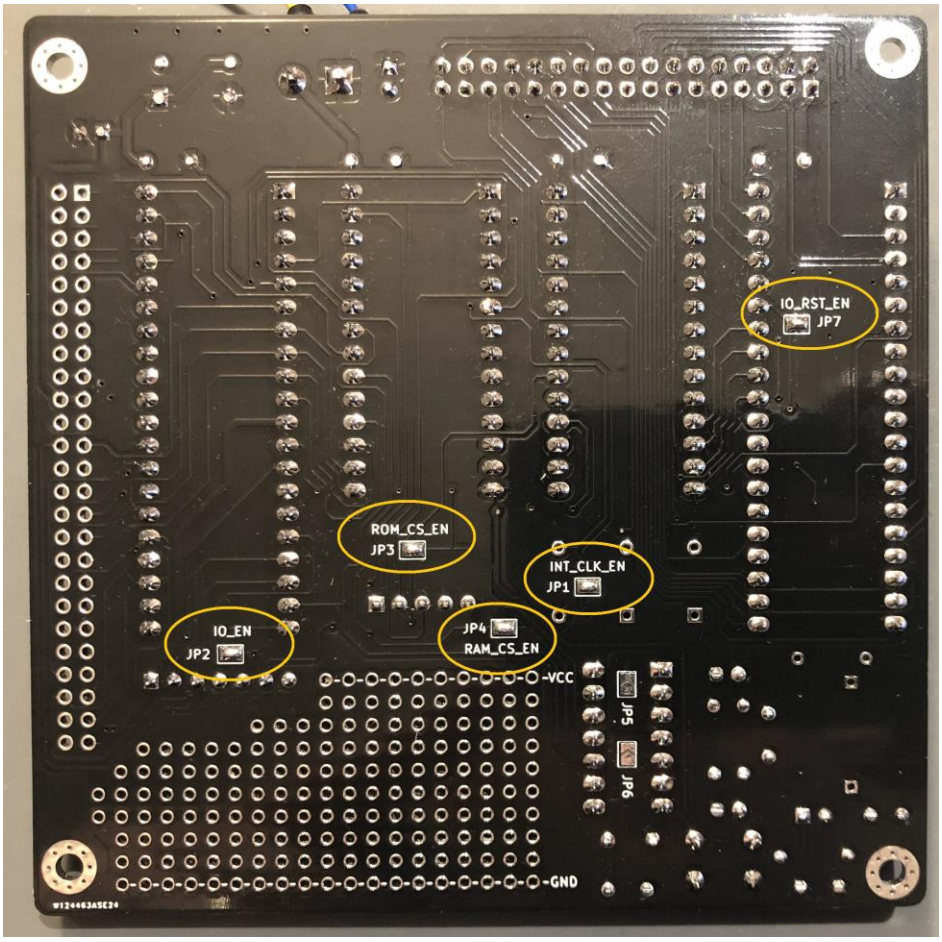
This document is being updated regularly, check my GitHub page for updates:

https://github.com/Kris-Sekula/mgh80

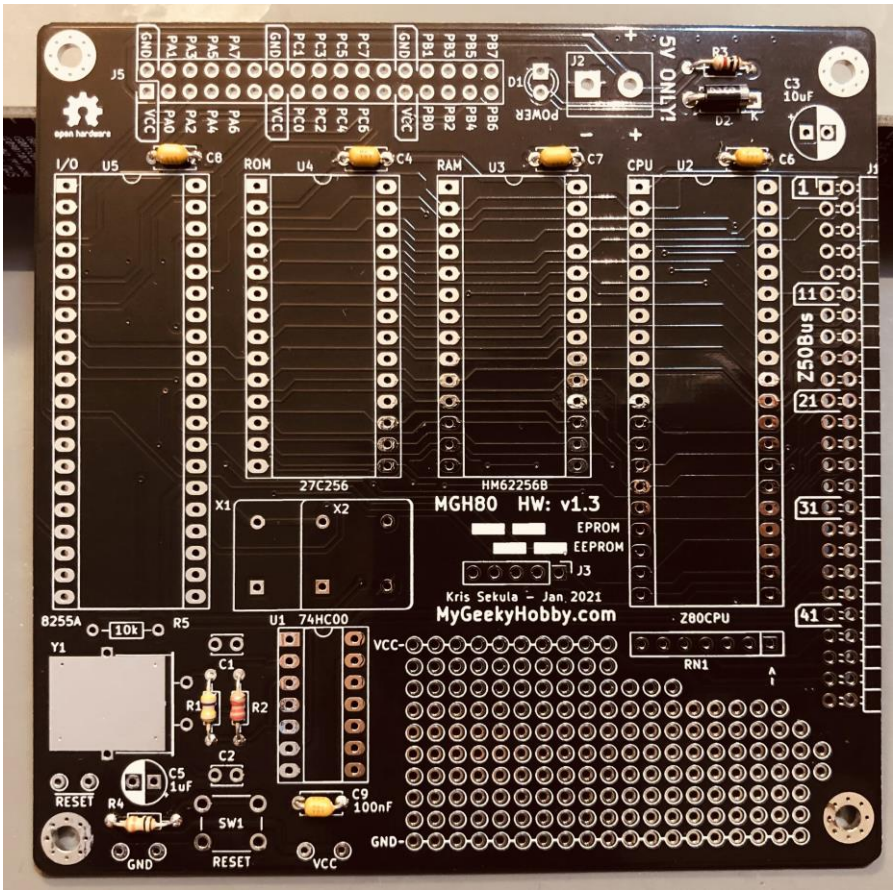
BOM mgh80 v1.3

| Qty | Reference(s) | Value | Included |
|---|---|---|---|
| 3 | reset,gnd,vcc | wire jumpers (optional) | N/A |
| 1 | PCB | HW ver 1.3 | ✓ |
| 5 | C4, C6, C7, C8, C9 | 100nF | ✓ |
| 1 | R1 | 470k | ✓ |
| 1 | R2 | 2.2k | ✓ |
| 1 | R3 | 1k | ✓ |
| 1 | D2 | 1N4002 | ✓ |
| 1 | R4 | 10k | ✓ |
| 1 | R5 | 10k (optional) | N/A |
| 1 | RN1 | SIP7 Socket | ✓ |
| 2 | U2,U3 | DIP40 Socket | ✓ |
| 2 | U3,U4 | DIP28 Socket | ✓ |
| 1 | U1 | DIP14 Socket | ✓ |
| 1 | D1 | LED | ✓ |
| 1 | Y1 | 4MHz | ✓ |
| 2 | C1, C2 | 22pF | ✓ |
| 1 | C3 | 22uF | ✓ |
| 1 | C5 | 1uF | ✓ |
| 1 | SW1 | RESET | ✓ |
| 1 | J3 | 2x 2pin jumper | ✓ |
| 1 | J5 | 34 pin connector female | ✓ |
| 1 | J2 | Screw_Terminal_01x02 | ✓ |
| 1 | J1 | 50 pin connector (optional) | N/A |
| 4 | H1, H2, H3, H4 | Rubber Feet | ✓ |
| 1 | RN1 | 5x10k Resistor Network | ✓ |
| 1 | U1 | 74HC00 | ✓ |
| 1 | U2 | Z80CPU (4MHz or faster) | ✓ |
| 1 | U3 | HM62256B | ✓ |
| 1 | U4 | 27C256 EPROM or EEPROM | N/A |
| 1 | U5 | 82C55A | ✓ |

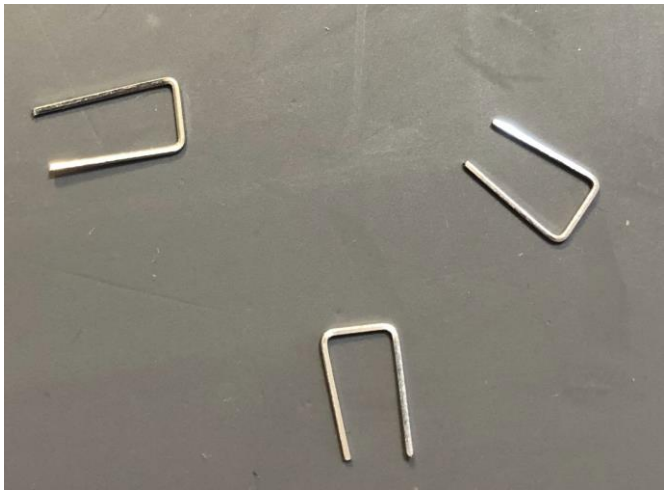
1. Configure the board option jumpers at the bottom of the PCB. Using a solder “blob” connect **JP1, JP2, JP3, JP4** and **JP7** as highlighted in yellow. **JP5 and JP6 are should be left opened.**



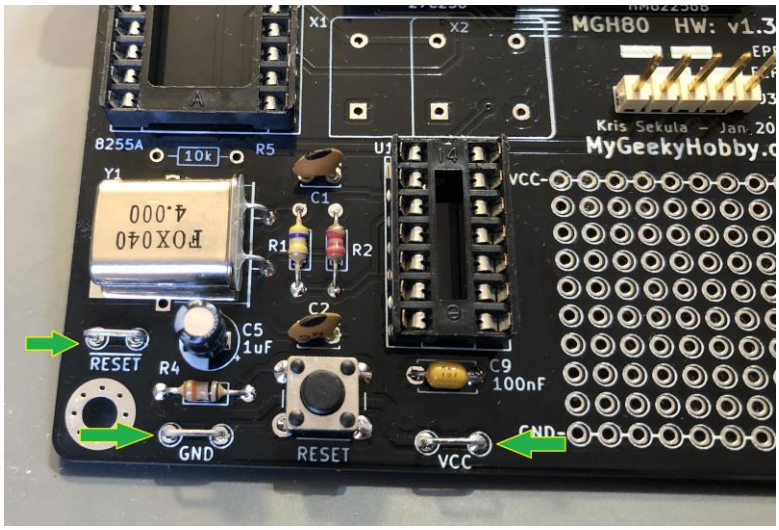
2. Start populating components from the shortest to the tallest: R3(1k), R1(470k), R2(2k2), R4(10k)… additionally if the supplied 100nF capacitors are the axial type, insert those into the PCB (C8, C4, C7, C6, C9) solder and trim:




3. Using leftover wire from trimming of the resistors, create the RESET, GND and VCC test points:

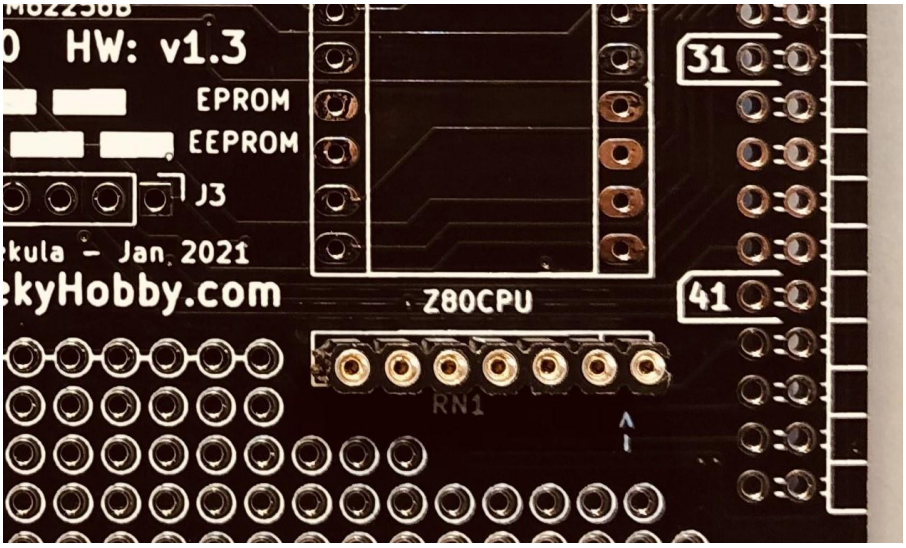


4. Solder the test points:

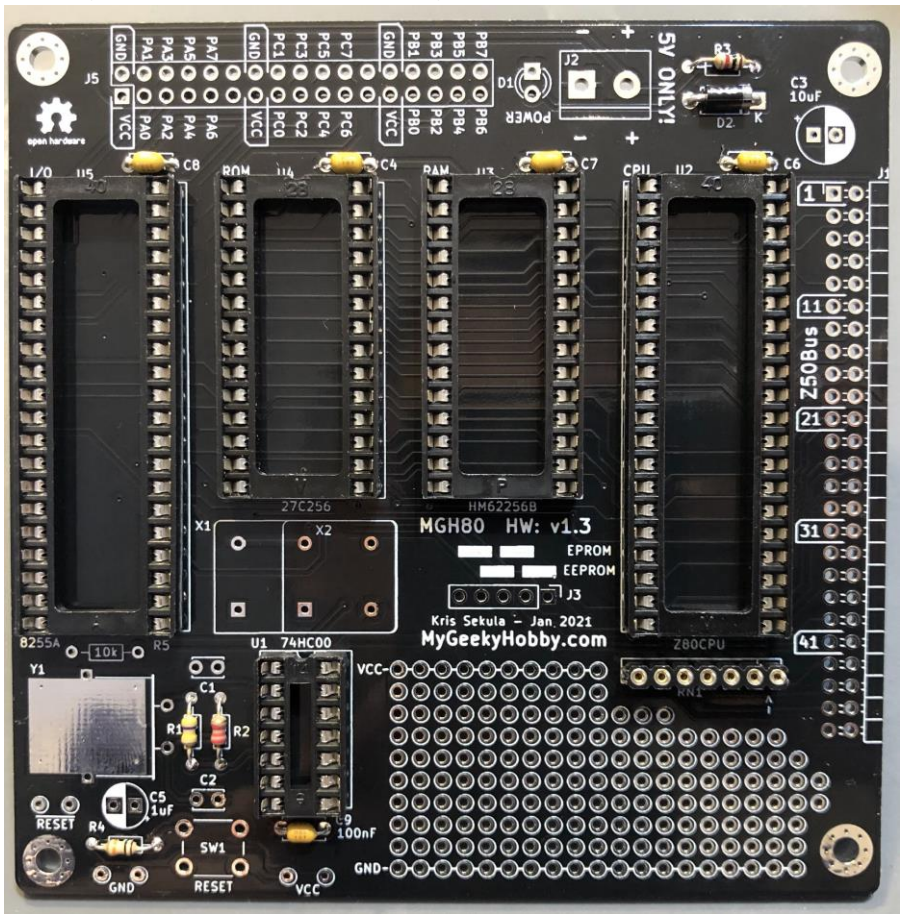



(note: other components are shown already soldered, but I changed the order of operations)

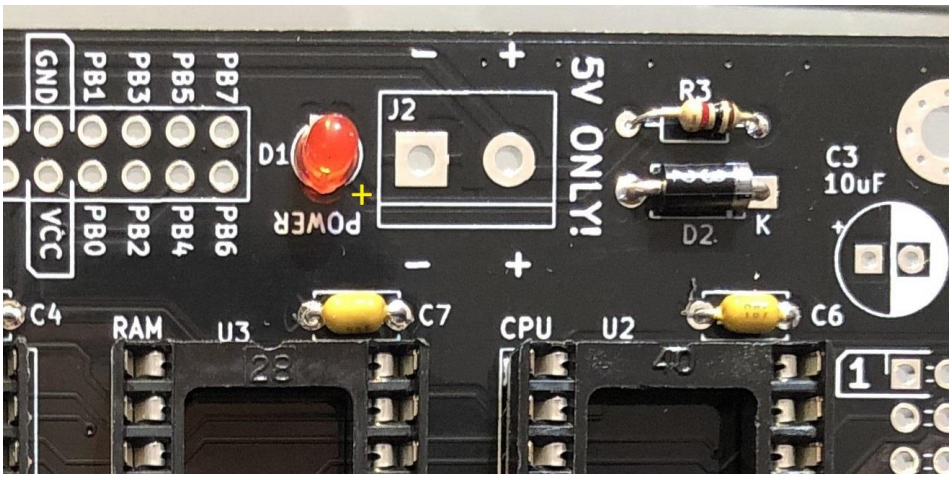
5. Solder the precision socket for resistor network RN1:



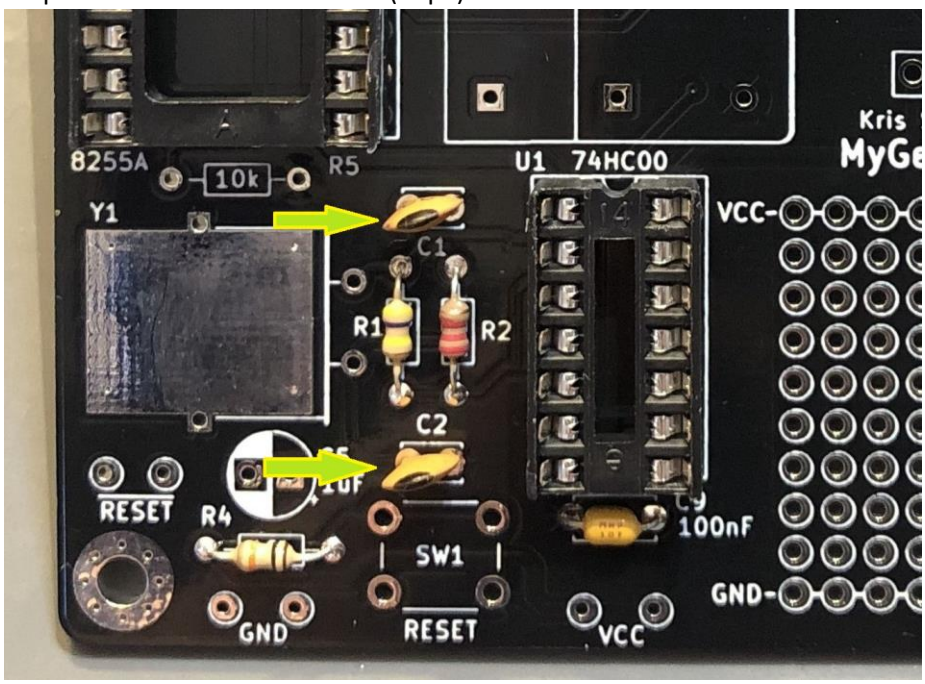
6. Solder IC Sockets (2x DIP28, 2x DIP40, 1x DIP14):




7. Solder the LED, the positive lead of the LED (Anode) should be facing the “POWER” writing on the PCB:

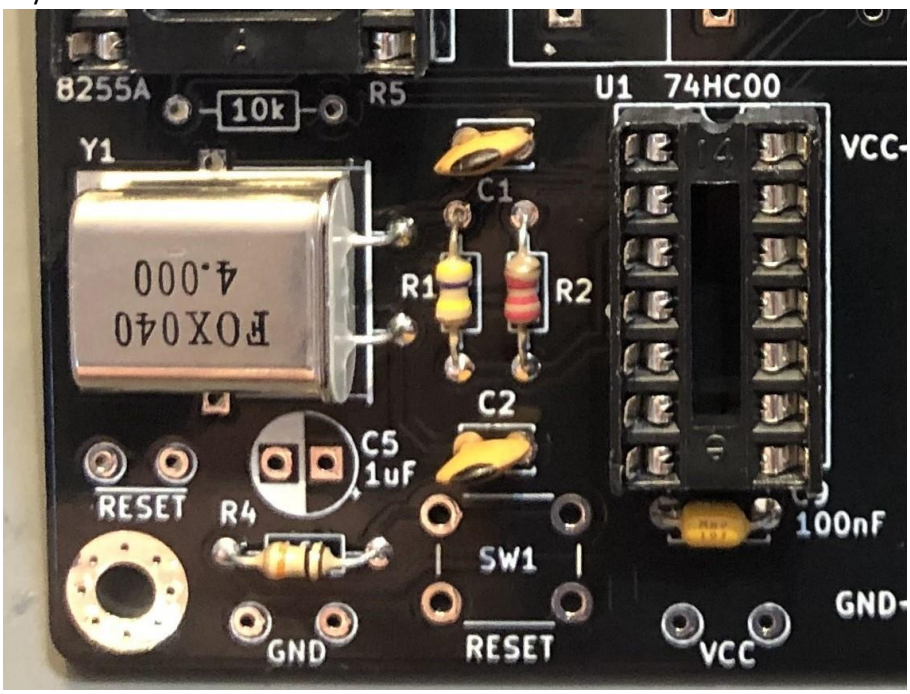


8. Solder the capacitor C1 and C2 – both are (22pF):

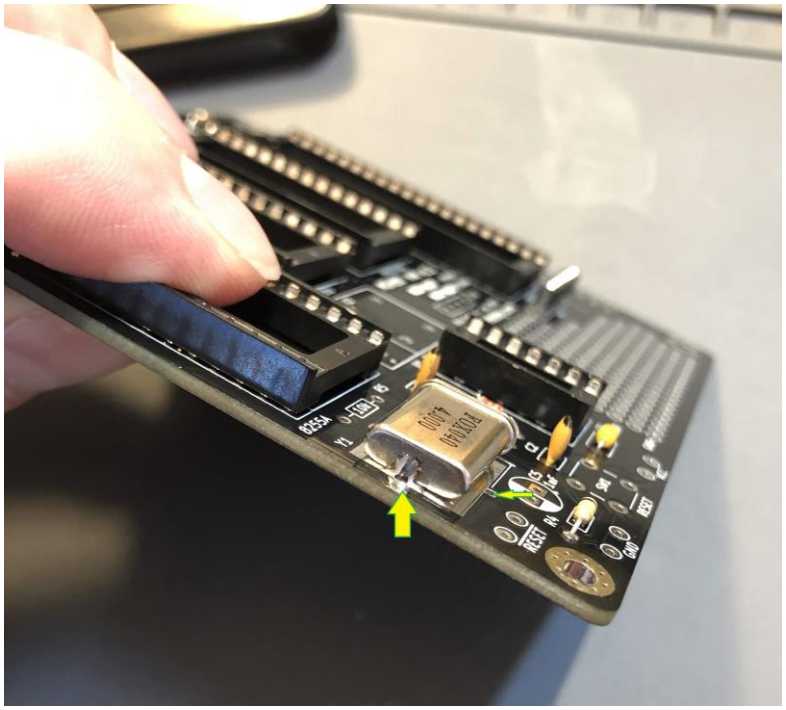



9. Prepare crystal oscillator Y1 by bending it's legs 90deg, I use the edge of the PCB to help:

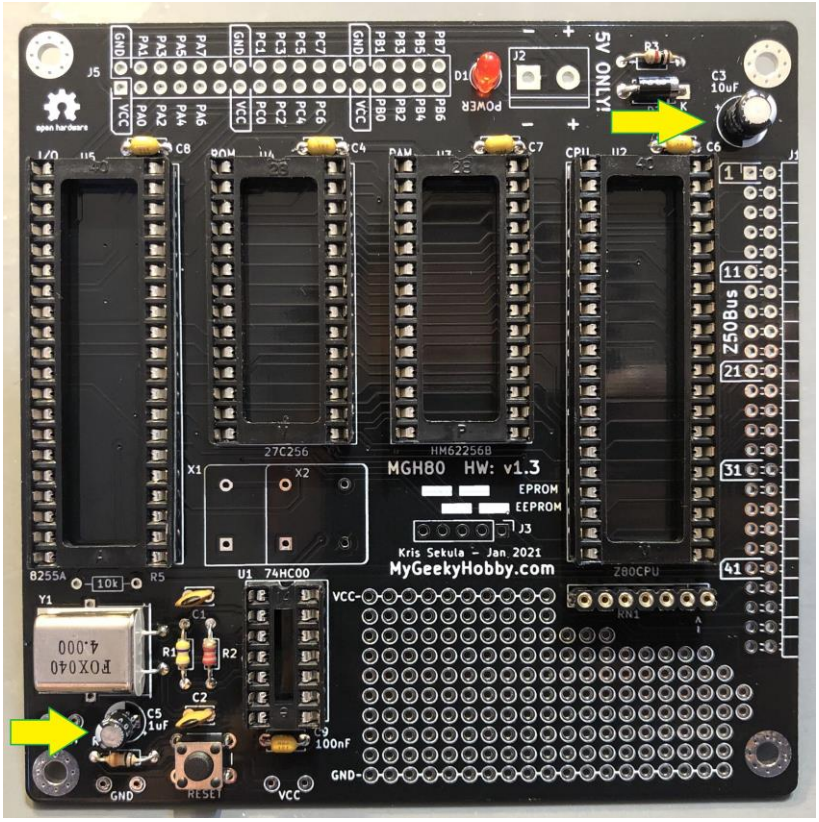
10. Solder the crystal oscillator Y1:




11. Secure the oscillator Y1 to the PCB with either a short link of wire or by soldering the oscillator directly to the PCB:

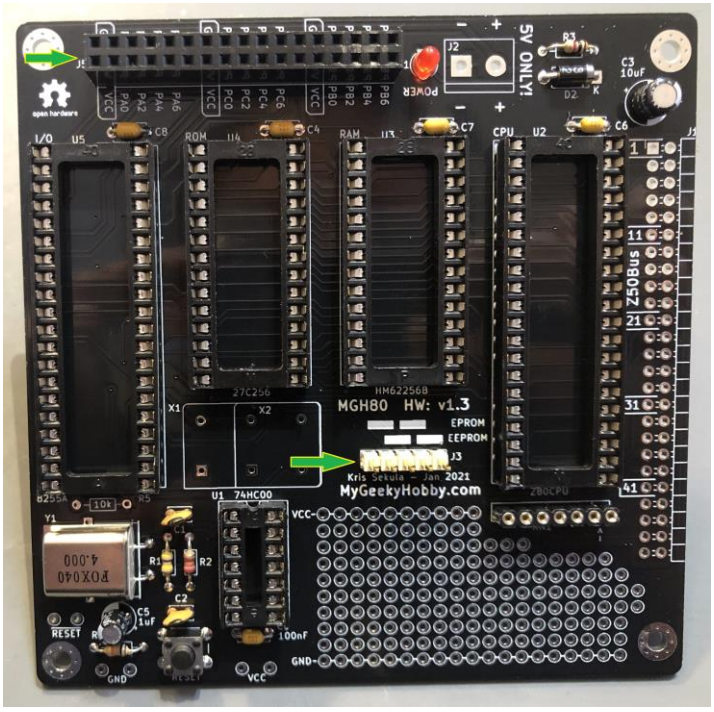


12. Install the electrolytic capacitors:

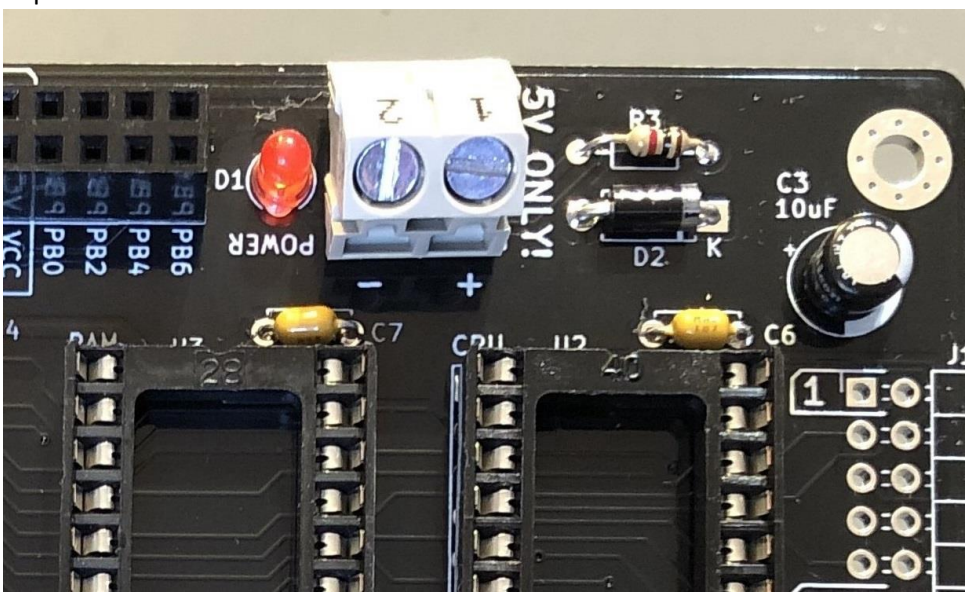



C5 is 1uF but C3 can be either 10uF or 22uF

13. Install the 34pin female connector J5 and 5 pin male header J3

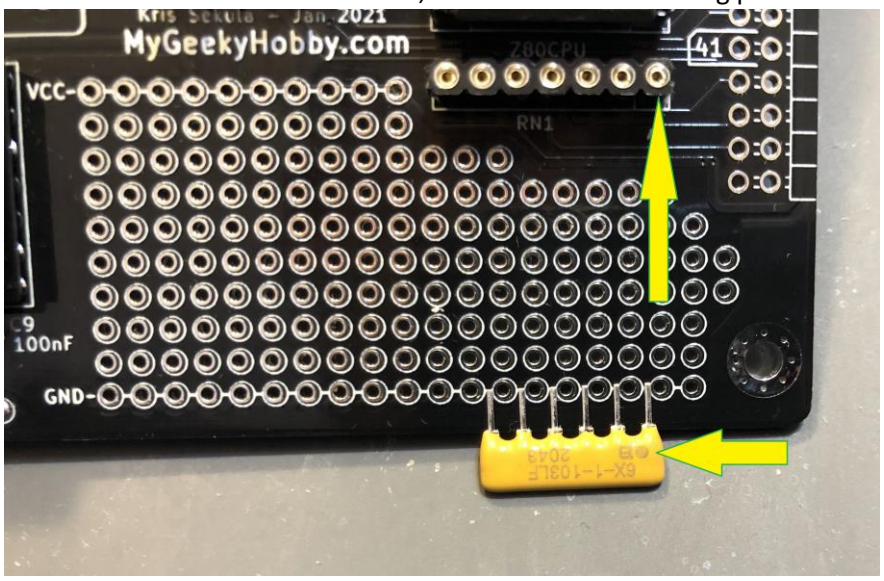


14. Install the power connector J2:

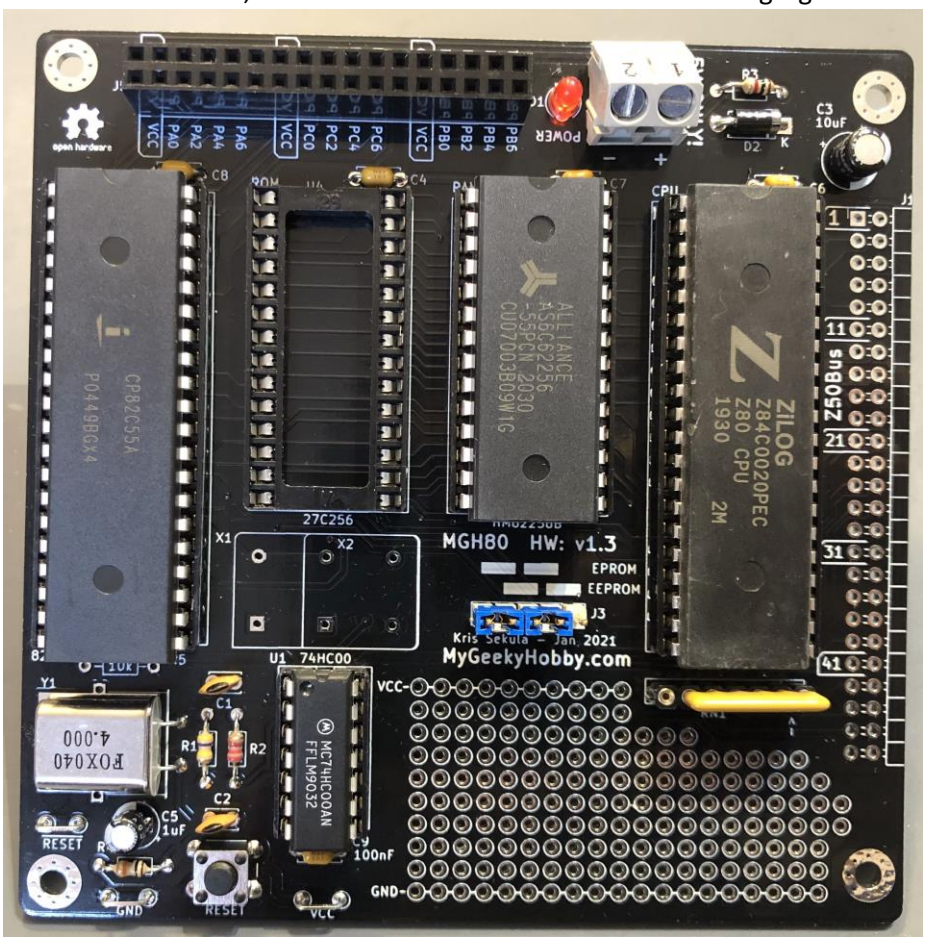



At this point all components are soldered, you should clean the solder residue with an appropriate cleaner (Isopropyl alcohol or similar). You can now insert components into sockets:

15. Insert the 10k resistor network into the socket, note the “dot” indicating pin one:



16. Insert the CPU in U2, the RAM in U3 and the IO in U5 and the logic gate into U1




17. Attach the provided rubber feet to the bottom of the PCB.

18. Depending on the type of ROM used, configure J3 jumpers:
    a. 27C256 EPROM memory (blue jumpers on the left of J3) like this:


    b. AT28C256 EEPROM (jumpers should be placed to the right of J3).


    c. W27C512 EEPROM from Winbond: those as pretty cheap on eBay and are pin compatible with EPROM but can be erased electricaly (I'm using the cheap TL866 programmer). Jumpers should be configured as per example a. (EPROM) and since this memory is 64k, you should load the program into the programmer starting at address 0x8000h (as A15 is held high for the 27C256).

You "MGH80" system is ready to use, you can find a simple "blink" program in the GitHub repository.

**Legal Notice:**