aek2_usb build guide

A very quick and dirty build guide, based on another quick and dirty build guide.

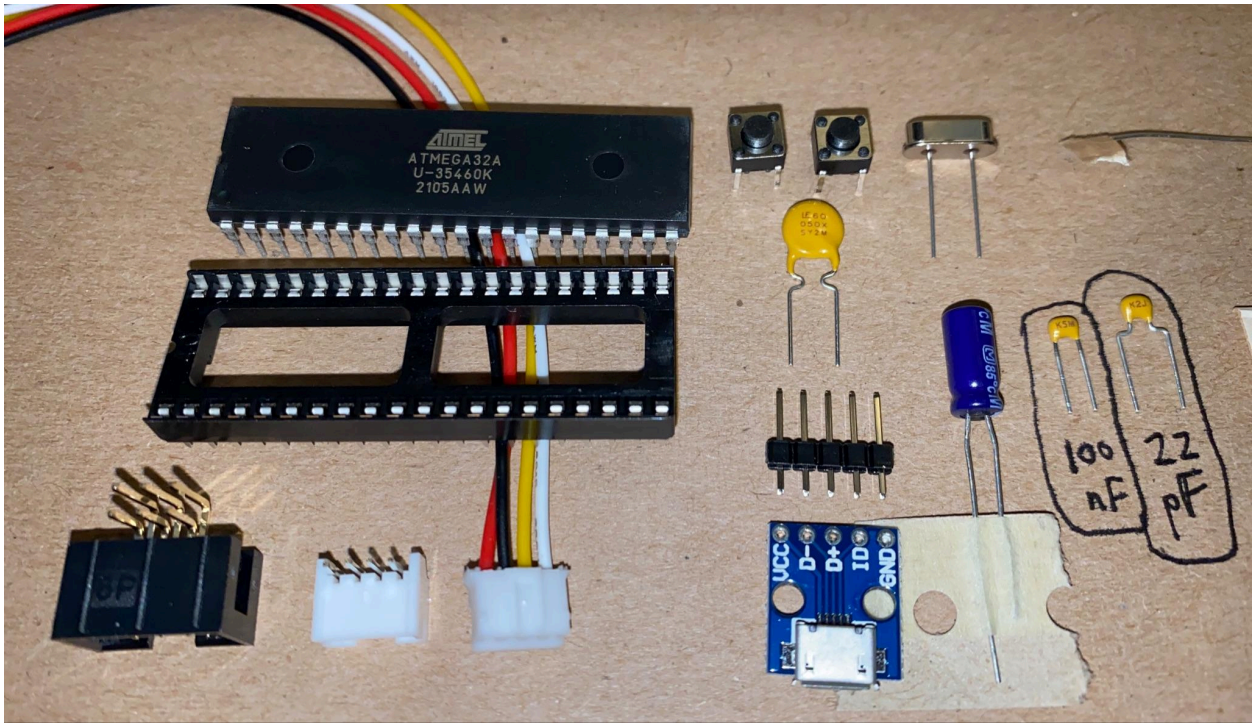
This guide was assembled by linuxleah from geekhack, based on Elmo's earlier assembly guide for the Sesame Ergo. The aek2_usb board itself was designed by kb-elmo (u/_vastrox_ on Reddit, elmo#0101 on Discord).

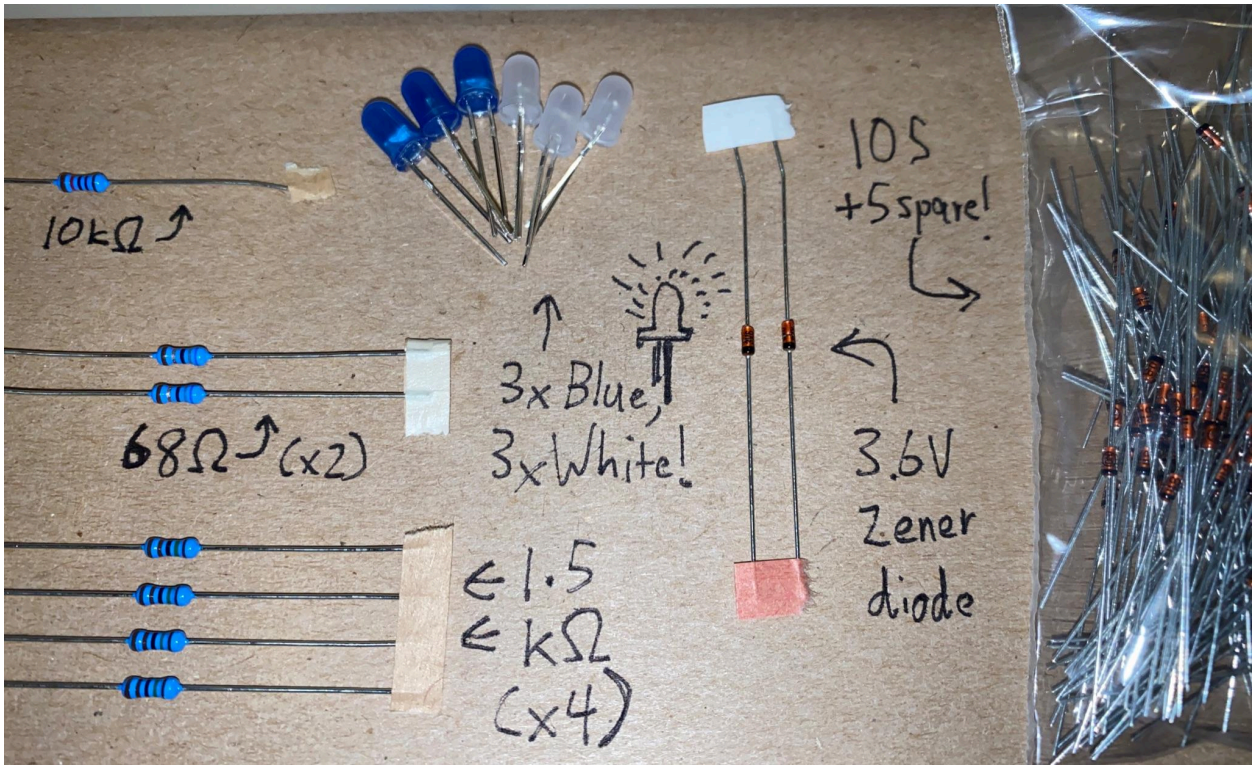
The geekhack thread on the second production run of these boards (June 2021, **blue** PCBs— Elmo's originals were **black**) is here.

Parts included in the full kit (in addition to the PCB itself):

1	ATmega32A 40 pin DIP (pre-flashed with bootloader and firmware)
1	40 pin DIP socket
1	2x3 pin header 2.54mm (for ISP flashing) - black socket
1	JST PH 2mm 4pin connector - white socket
1	... + JST cable for above
1	USB breakout PCB
1	500 mA polyfuse
2	6mm momentary push buttons
1	16 MHz quartz crystal
3	5mm LED (for the lock indicators) - **BLUE option**
3	**...WHITE option**
110	universal switching diode (DO-35 1N4148)
2	3.6 V zener diode (DO-35 BZX55C3V6)
1	4.7 uF electrolytic capacitor
2	100 nF ceramic disk capacitor
2	22 pF ceramic disk capacitor
1	10 kΩ resistor
2	68 Ω resistor
4	1.5 kΩ resistor

Photo references for the parts:





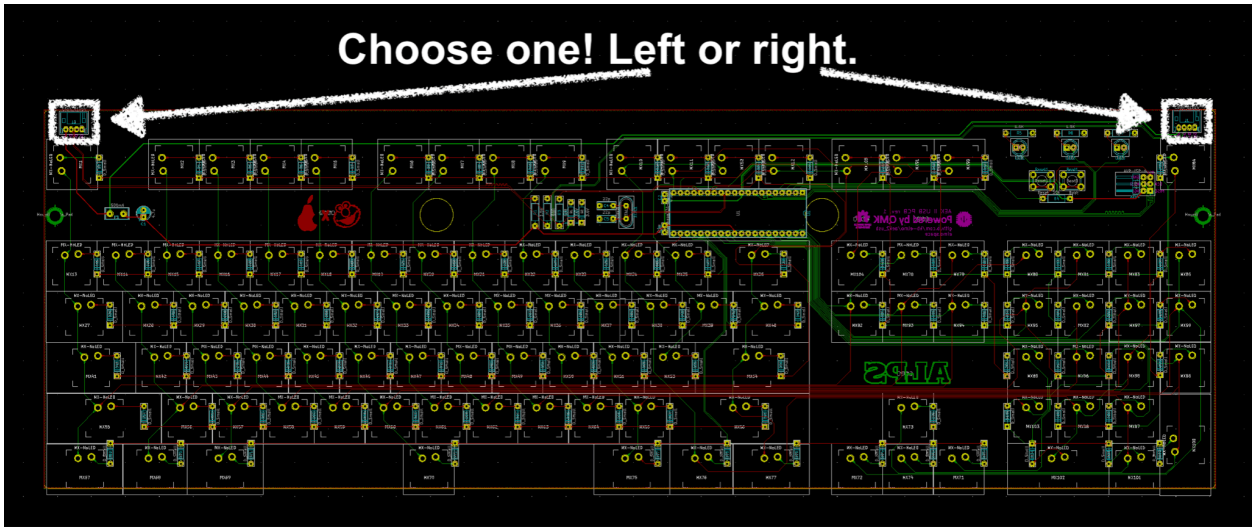
You only need 105 of the switching diodes for the board. The rest are replacements in case one gets damaged during soldering.

You also only need three LEDs in total. The kit includes three blue LEDs and three white LEDs. You can use whichever you wish, or mix and match as you choose.

Building the PCB:

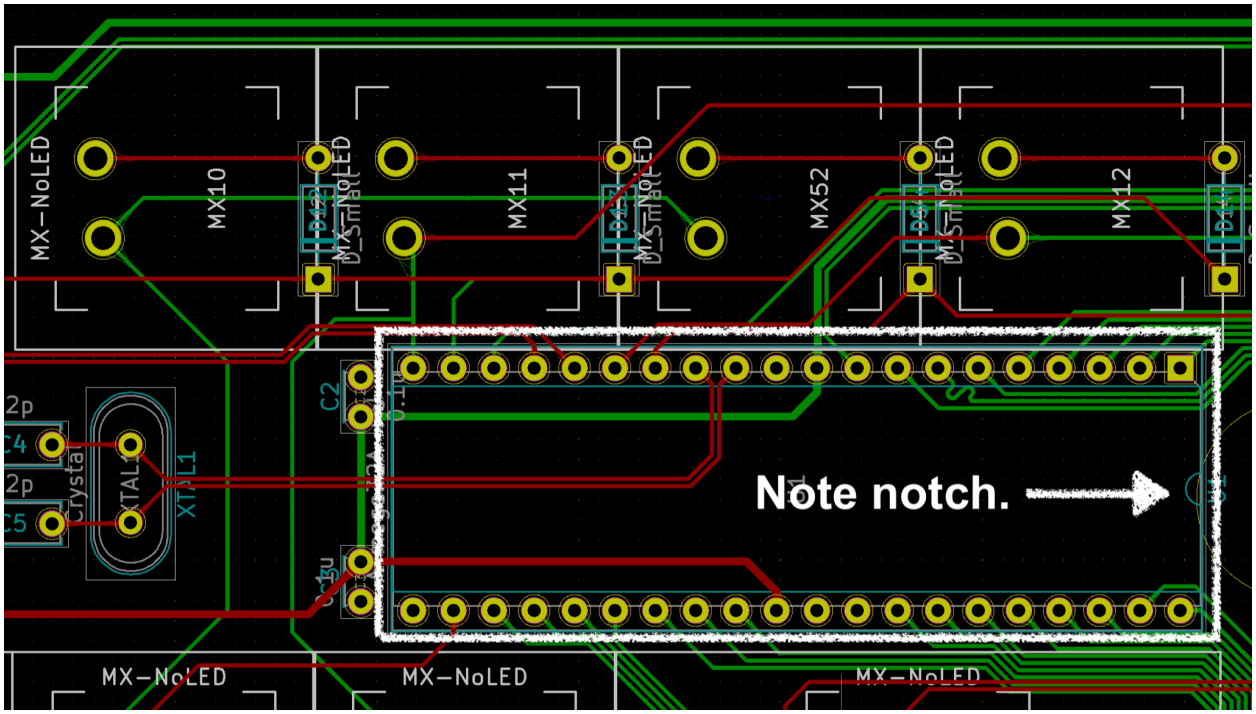
Note that the diagrams in the below are screen captures from KiCad, rendering the PCB source project from the aek2_usb github repository. This repository contains the actual source files used to fabricate the boards.

1. Start with the JST 4-pin connector (which is *white* as provided in linuxleah's kit). This goes at your choice of either the top *left* or top *right* of the board, depending upon where you wish to place your microUSB socket. Choose wisely!

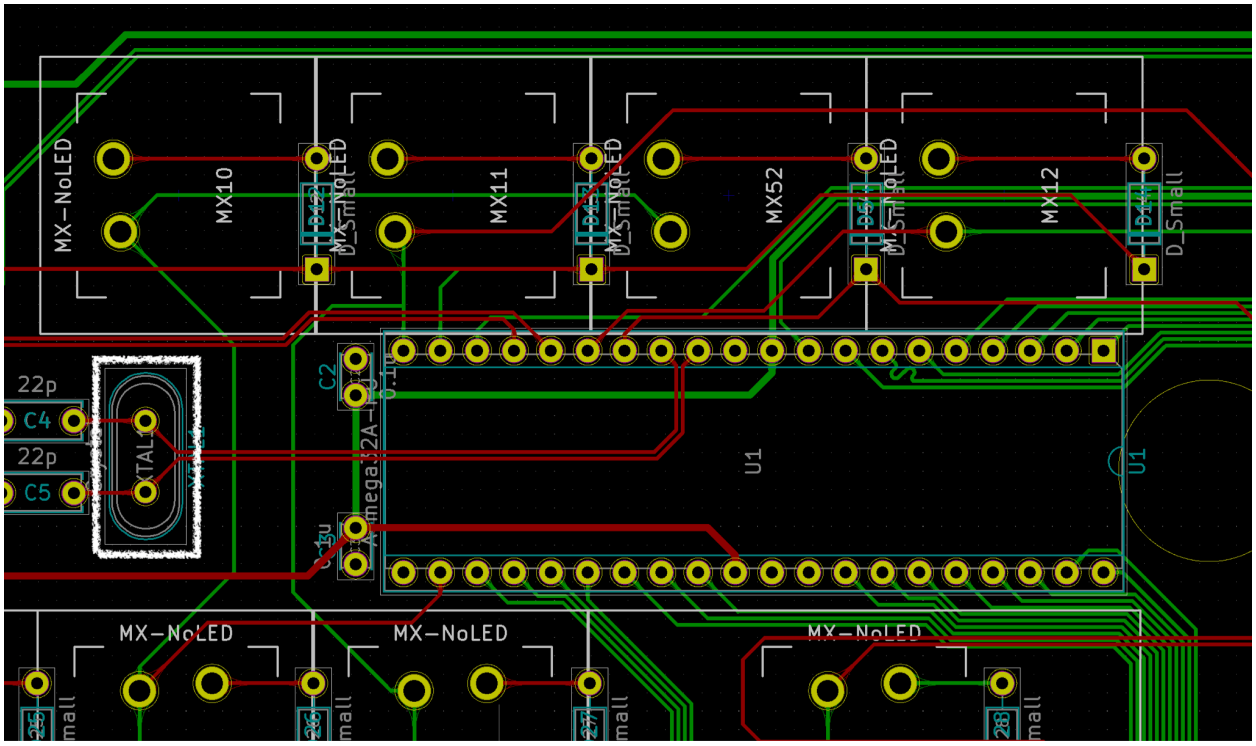


Put in the connector and secure it in place with some electrical tape. Then solder it on the bottom of the board. You can then remove the tape.

2. Next should be the controller socket. Put it in with the cutout notch facing like it's shown on the PCB silkscreen. Hold it in place with some electrical tape on top of the board. Solder every second pin on the board to prevent overheating the plastic. Then do the rest of the pins.

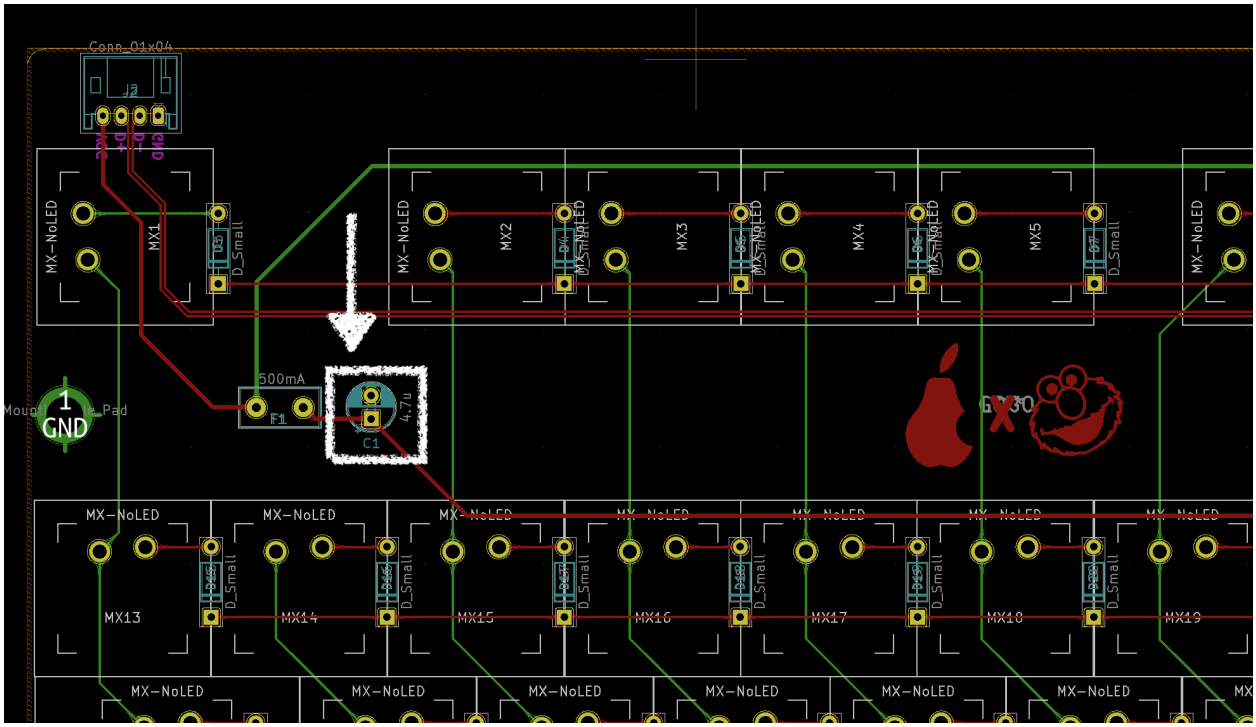


3. Solder the quartz crystal into its spot next to the controller (XTAL1). Polarity doesn‘t matter here. Do not pull hard on the part’s legs, as this can damage the part.

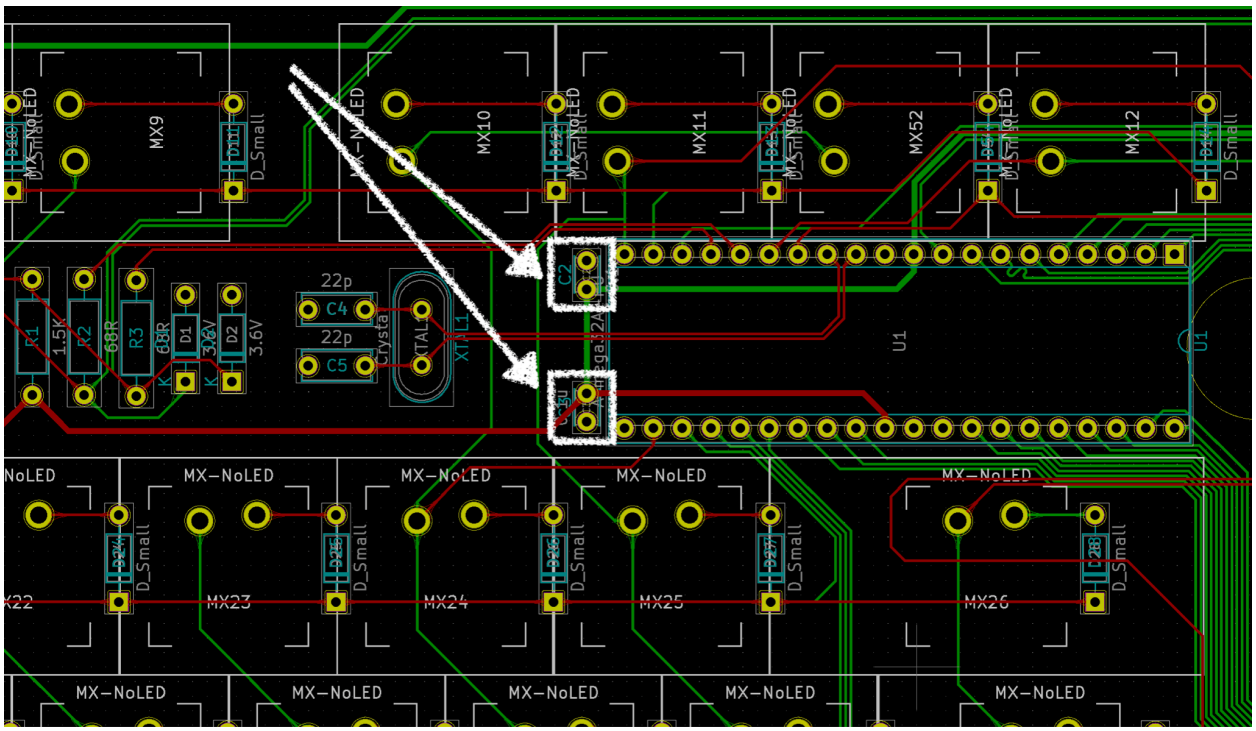


4. Solder in the capacitors. The values are as follows:

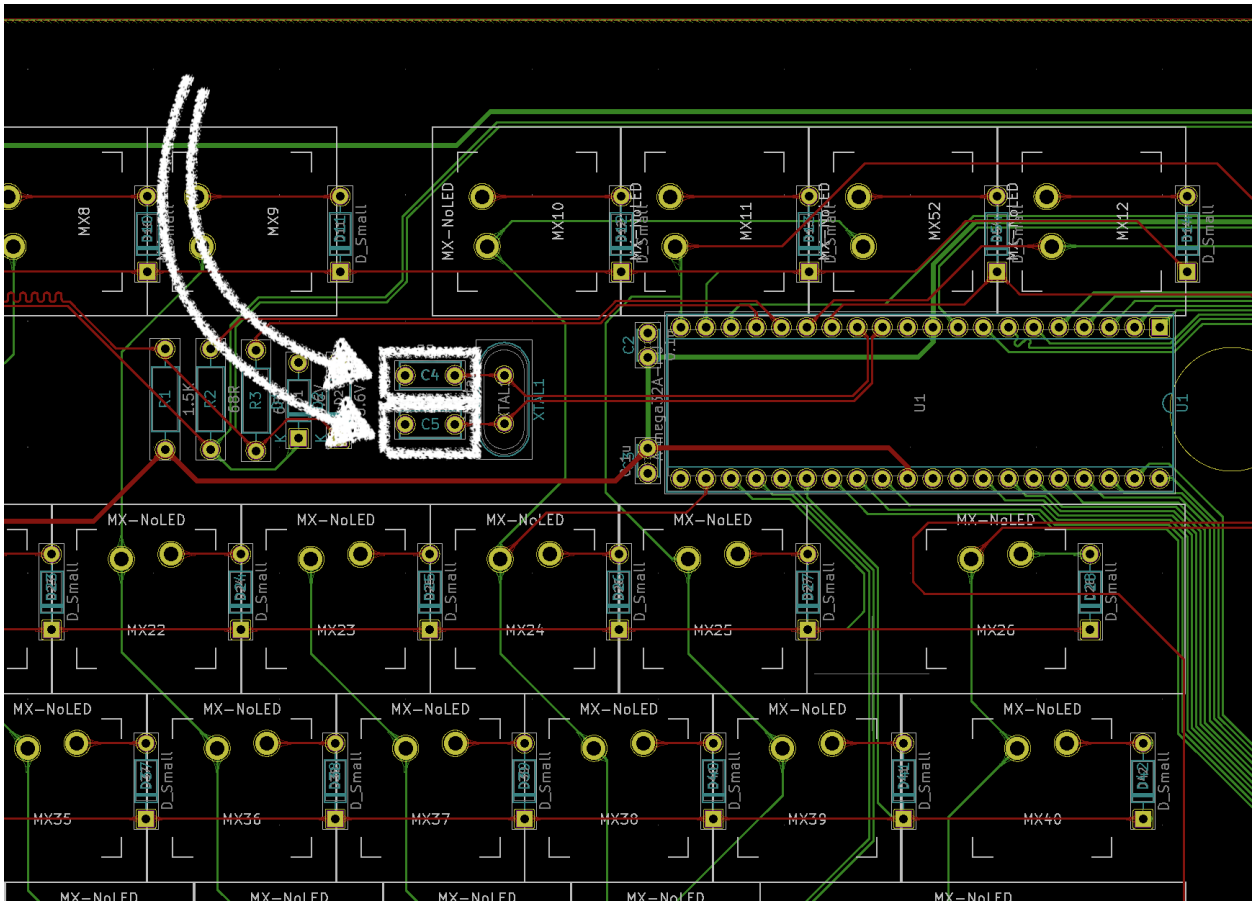
- C1 (ELKO! Watch the polarity! white side = negative) = 4.7uF; this one lives at the top-left of the board when viewed from the top, not far from one of the possible spots for the controller socket.



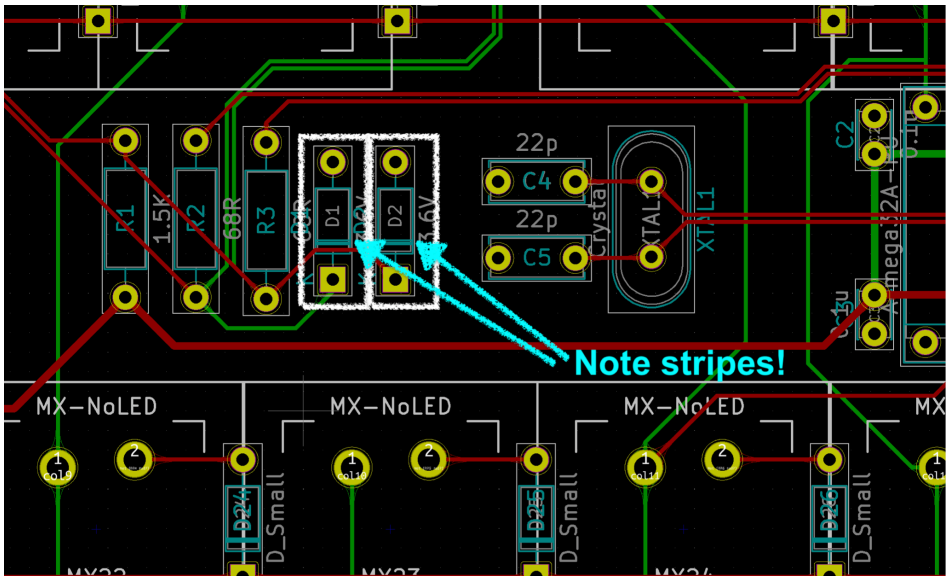
- C2 & C3 (next to the controller socket) = 100nF. Little yellow capacitors; polarity does not matter.



- C4 & C5 (left of the crystal) = 22pF. Little yellow capacitors; polarity does not matter.



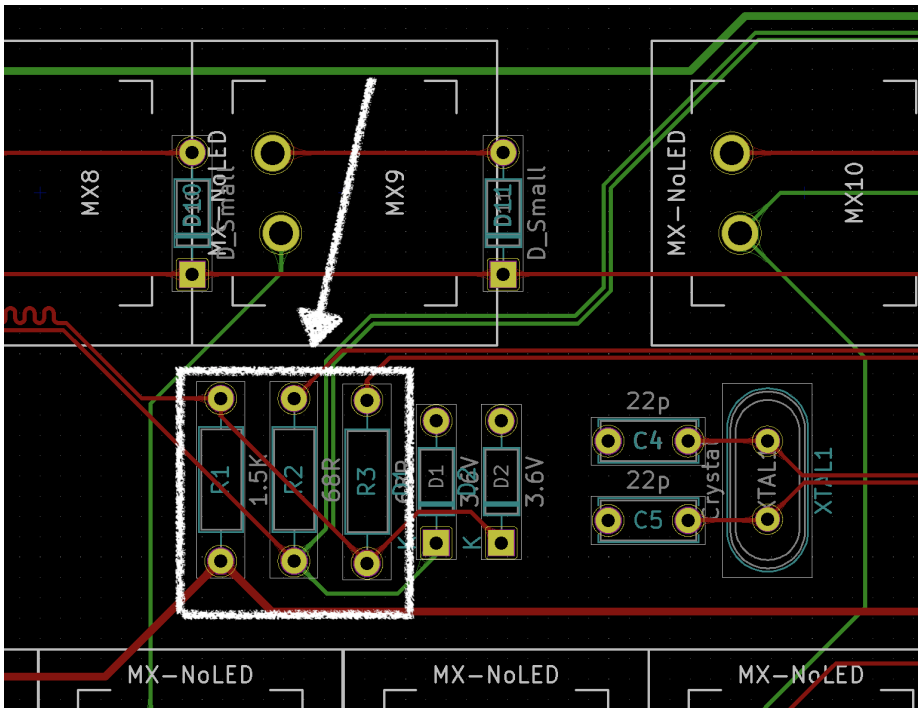
5. Next up are the two 3.6V Zener diodes. They go into the spots D1 and D2. Watch the polarity! The white line on the silkscreen has to match with the black line on the diode.



These are special diodes! They can not be exchanged with the standard switching diodes! Make sure that you are using the correct ones.

In the kits from linuxleah, the zener diodes are attached together with paper tape on the legs. The switching diodes are provided loose in a bag.

6. Now come the resistors. The first three live to the left of the diodes we just added:

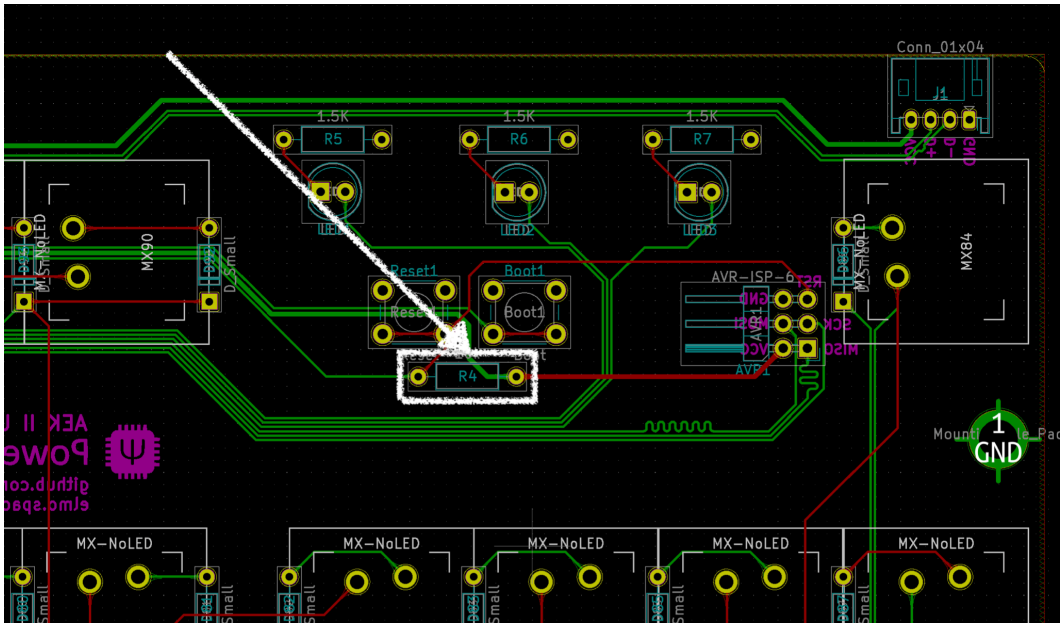


Their values are as follows:

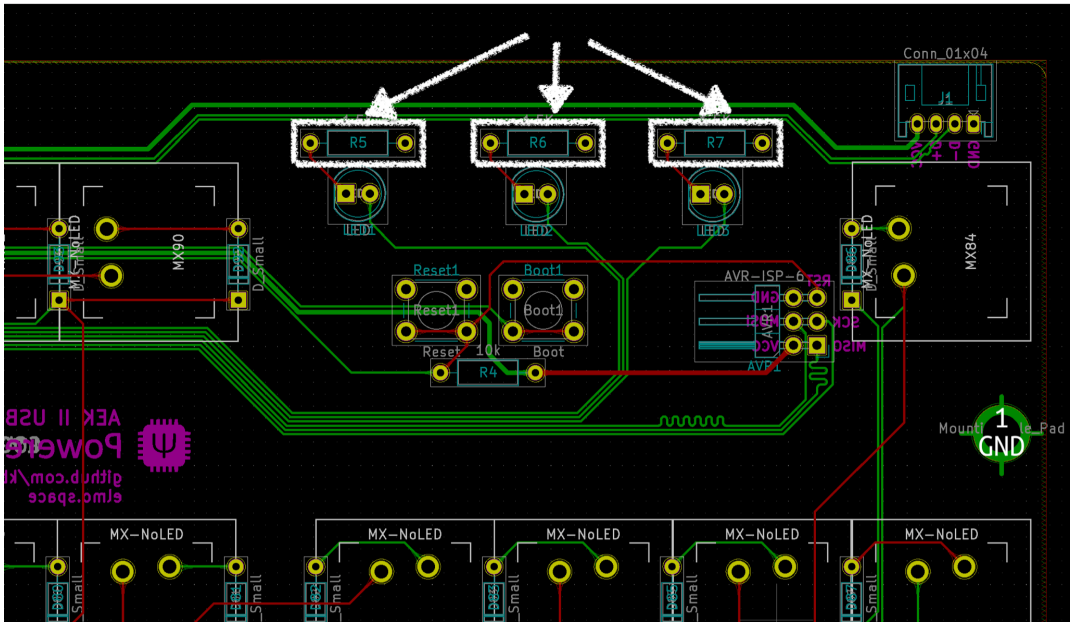
- R1 (left of the crystal, leftmost resistor) = 1.5kΩ

- R2 & R3 (the other two in that group, immediately right of R1) = 68Ω

Next comes R4, a 10 kΩ resistor. This lives at the top right of the board, below where the Reset and Boot pushbuttons will go:

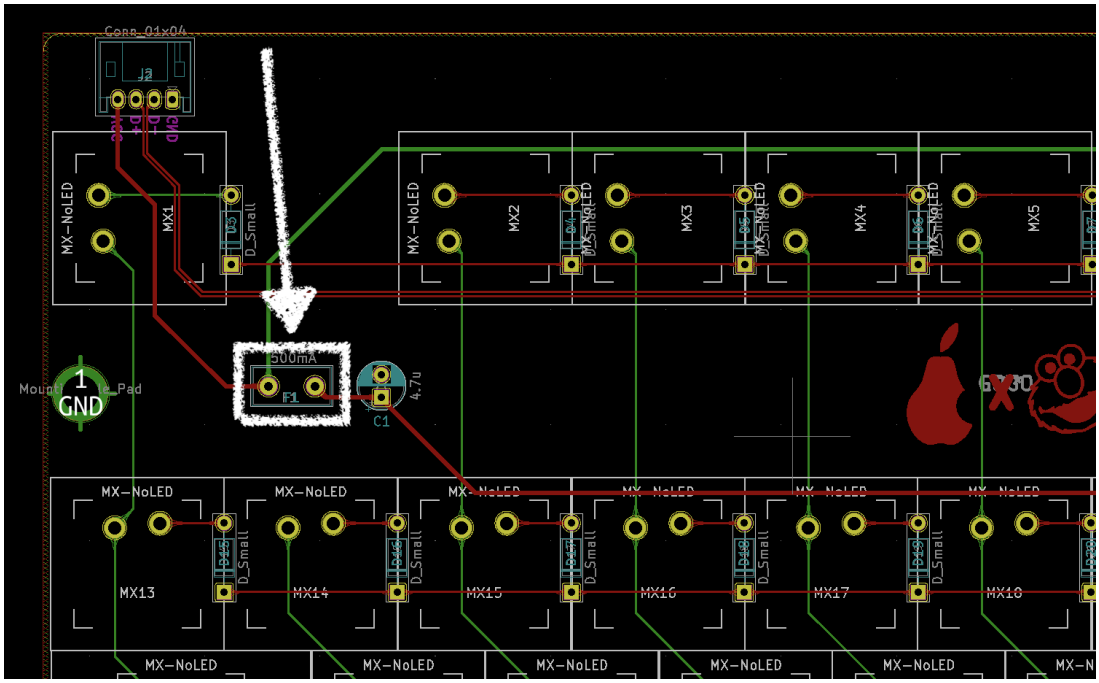


Finally, the remaining three 1.5kΩ resistors (R5, R6, and R7) live immediately above where the LEDs will go:

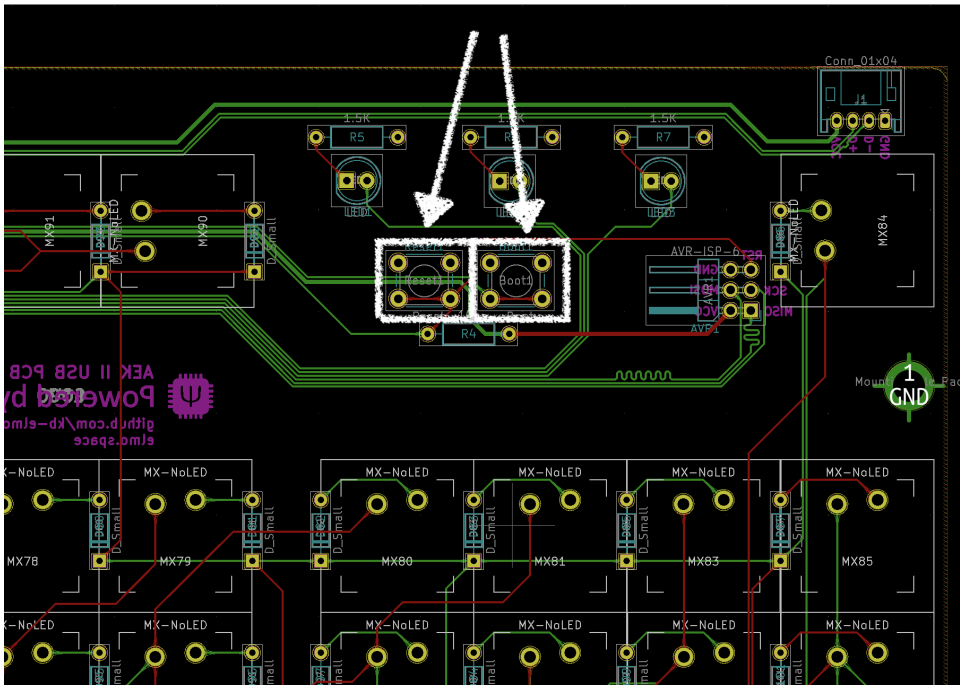


7. Solder in the rest of the parts:

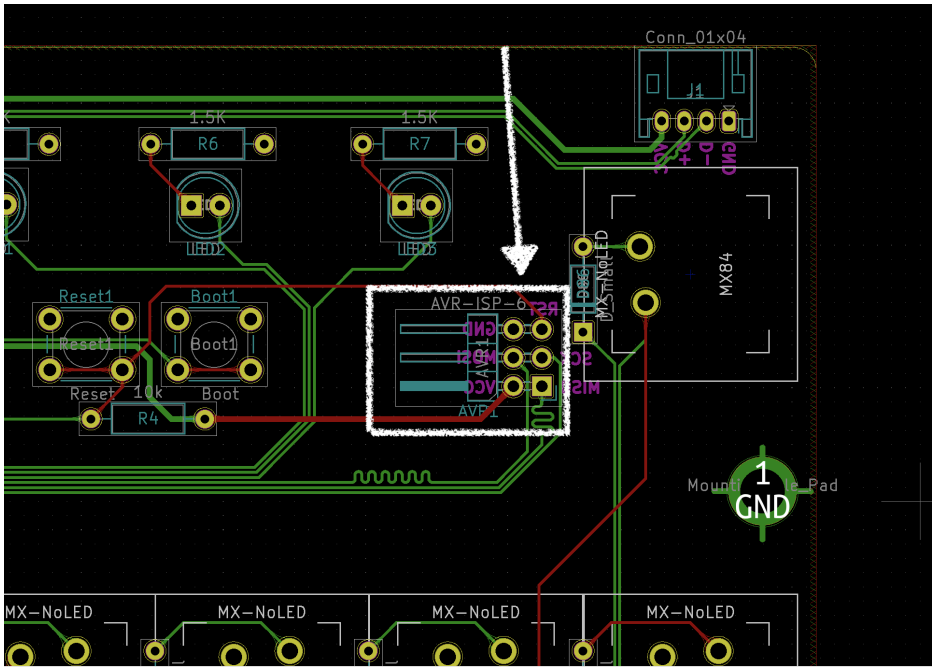
- the 500mA polyfuse (spot F1, at top-left adjacent to the large capacitor C1):



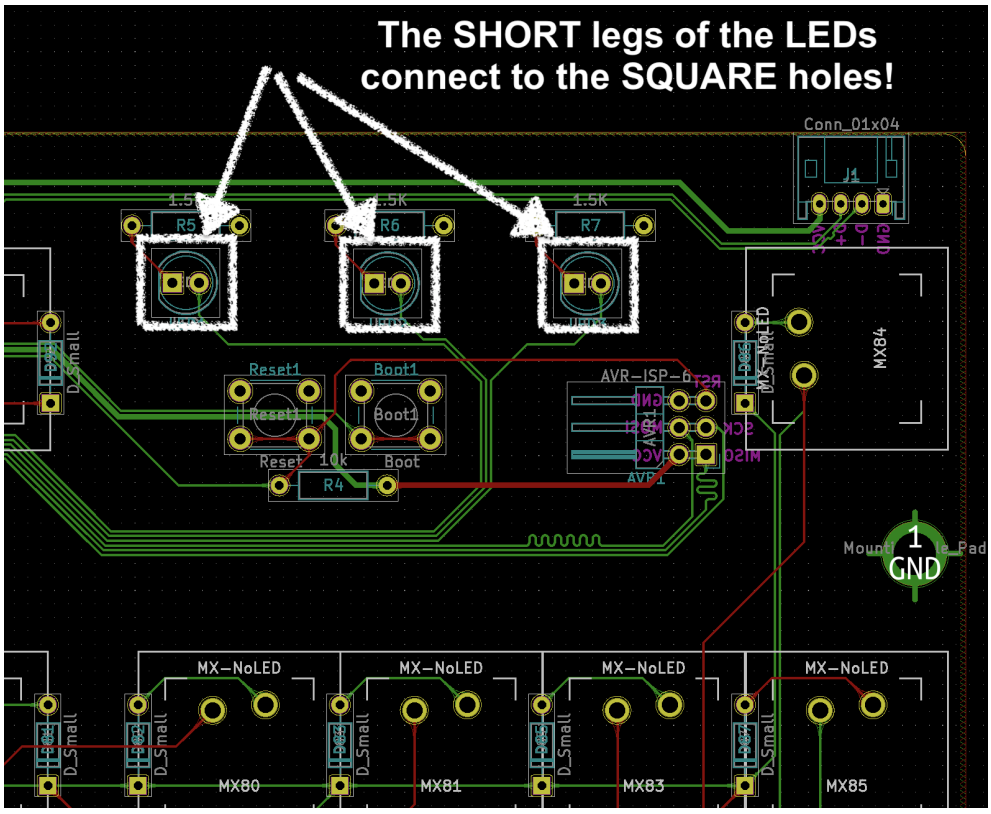
- the boot and reset pushbuttons (at top-right under the LEDs):



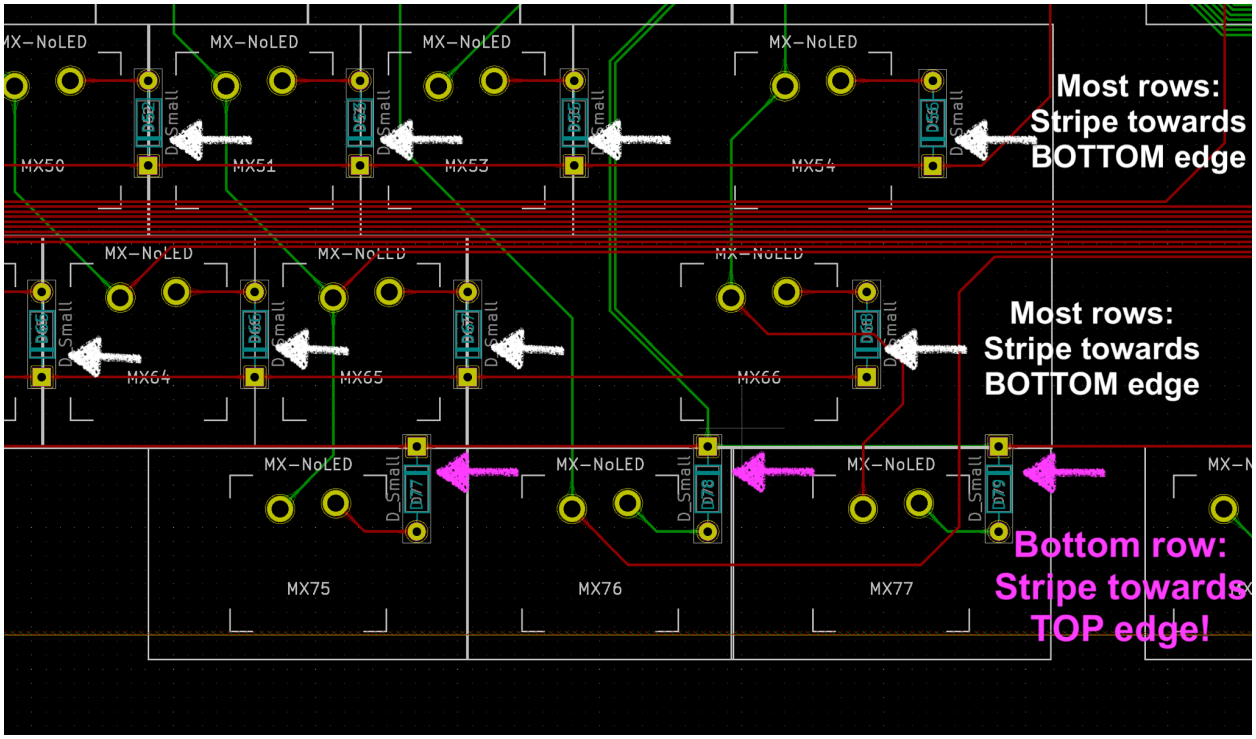
- the ISP flashing header (which is *black* as provided in linuxleah's kit), at top right under where one of the LEDs goes:



- The three LEDs (white and/or blue; your choice!) at top-right. The **SHORT** legs (negative, a.k.a. cathode) of the LEDs connect to the **SQUARE** holes!



8. Finally, solder on the last components: All 105 diodes. (In linuxleah's kit, these are provided in a large bag. You're provided with 5 spares, just in case.) Watch the polarity: The black stripe on the diode has to match the white stripe on the board. **The bottom-most row of keys has their stripes facing the "opposite way" from most other rows; watch the stripes carefully!**



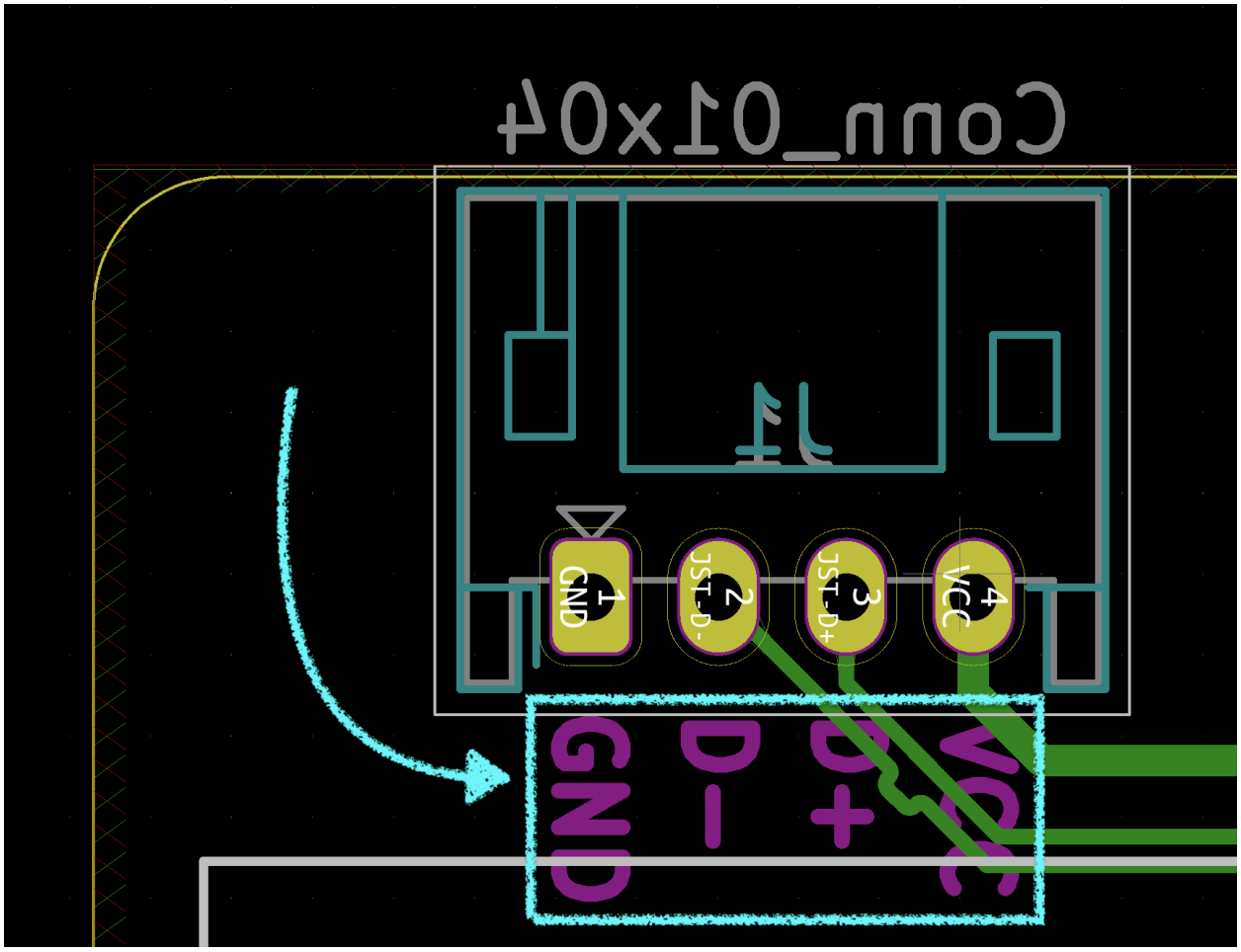
9. Lastly, put the controller into the socket. Make sure that the indentation on the controller matches with the indentation in the socket! If you put it in the wrong way around, it can destroy the controller.

Press it down firmly so that it sits flush with the socket. Take care to not bend any legs.

Keyboard integration guide:

1. Take the JST cable and solder its four wires to the Ground, VCC (+5V), Data+, and Data- lines on the USB breakout board. **Reference the order of the pins as indicated on the BACK OF THE BOARD:** GND, D-, D+, VCC. Pay attention to the order that the wires will be in when the JST cable is seated in its socket, and ***note that the colours and order of the wires on the provided JST cable won't match the usual order of***

USB wires!



2. Plug the USB breakout into the white JST header/socket at the top-left or top-right of the board (wherever you chose).

3. Either simply thread the header out of the hole where the ADB connector went, or devise a more clever way of wiring up the USB connection to your computer. (Here's some inspiration: USB signals run over the original ADB header, with a custom cable!)

Building and flashing the bootloader:

(If you bought the kit from linuxleah, the controller is already flashed with the bootloader. You can skip this step!)

Download the bootloader firmware from here:

https://github.com/kb-elmo/aek2_usb/tree/master/firmware/bootloader

You need either a Linux system or MingW on Windows for this, as this requires compiling some stuff.

The firmware is already set up and you don‘t need to change anything in the configuration files for this board.

1. Build the bootloader firmware by running “make” on your console

2. Make sure that you can reach the controller by executing a chip erase with

“avrdude -c usbasp -p m32 -e” (you have to run these commands as root in Linux)

3. Flash the firmware on the controller with “make flash”

4. Set the controller fuses to the correct values with “make fuse”

5. Finally lock the controller bootloader in place with “make lock”

The avrdude tool should show a “success” after each operation. If the flashing process was successful you can unplug the ASP flasher and plug in the USB now.

Building and flashing QMK:

Note: The firmware preloaded on the controller provided with linuxleah’s kit uses the “default” keymap for the aek2_usb keyboard. This keymap has two notable features:

1. ***The Command keys are mapped to Alt (a.k.a. Option), which is mostly useful for Windows computers. This can be changed with a re-flash. (Alternatively, you can use System Preferences to swap the Modifier Keys.)***
2. ***The right Option key is the layer change key and <u>does not function as a regular key.</u>***

The keyboard can be found on https://config.qmk.fm if you want to configure a static keymap or you can use VIA. I recommend the latter.

For VIA download the firmware for the board and the client from:

https://caniusevia.com

Flash the firmware on the board using the QMK Toolbox which can be found here:

https://github.com/qmk/qmk_toolbox/releases

To put the board's controller into bootloader mode, press and hold down the Boot button and tap the Reset button once. Then let go of the boot button.

The board should show up in your OS as a "USBasp" device (or similar).

After the flashing process was successfull hit the reset button once to get into normal keyboard mode again.

You can of course also build the firmware yourself if you prefer that.

The code for this can be found in the QMK repository in keyboards/kb_elmo/aek2_usb. Details of this are beyond the scope of this document.