**Designing a keyboard**

|  |  |
| --- | --- |
| Author | Stijn Verhoeven |

Content

[1 Introduction 1](#_Toc68369923)

[2 Material and methods 2](#_Toc68369924)

[3 Results 3](#_Toc68369925)

[3.1 Subtitle 1 3](#_Toc68369926)

[3.1.1 Subtitle 3](#_Toc68369927)

[3.1.2 Subtitle 3](#_Toc68369928)

[3.2 Subtitle 2 3](#_Toc68369929)

[3.2.1 Subtitle 3](#_Toc68369930)

[3.2.2 Subtitle 3](#_Toc68369931)

[4 Discussion 3](#_Toc68369932)

[5 Reference list 3](#_Toc68369933)

# Introduction

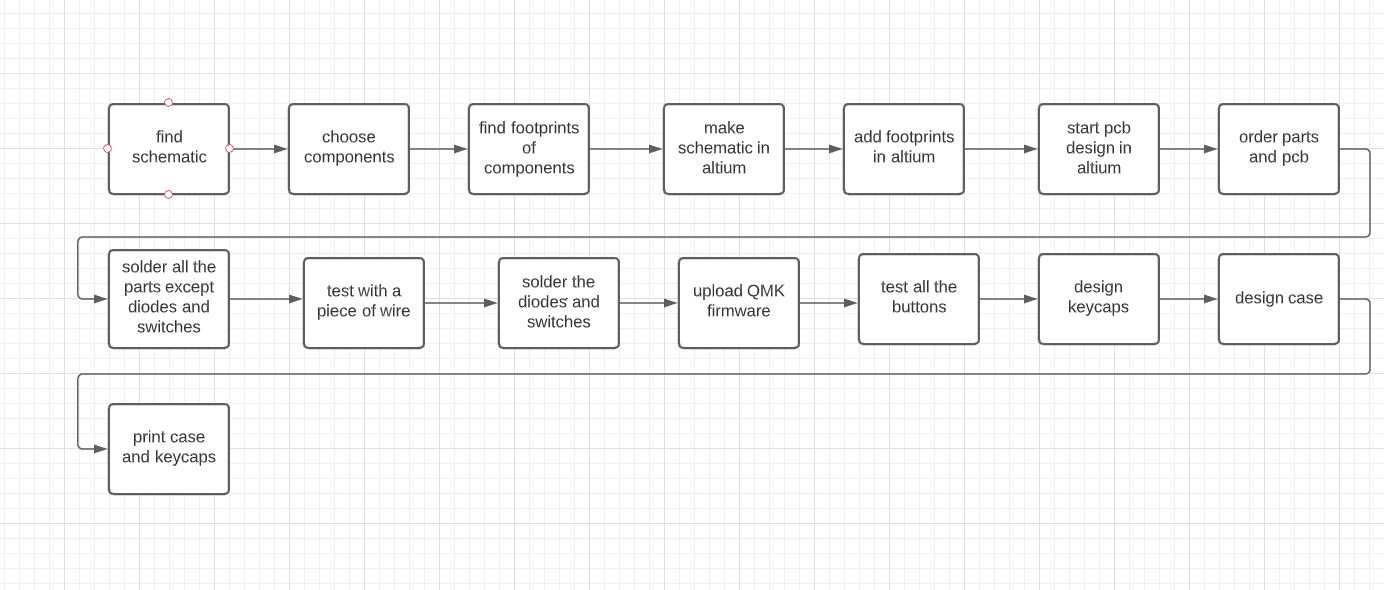
Making a keyboard from scratch. There are inexpensive keyboards, but this is a higher-end one. On every keyboard, there are keys from a to z and special characters. There also are some special keys (backspace, page up, page down, space bar, etc…) and sometimes a numeric pad. This one is not going to have a numeric pad. Starting this project with a schematic from a website. In the paragraphs below, it is going to be explained how to make the PCB design, how to design the case and the keycaps.

# Material and methods

For firmware, I use QMK to map the buttons on the keyboard. Originally, I was planning to use TMK firmware, but I found that QMK was better because it is more regularly updated, has more options on how to map the buttons, has a smaller data package and is much easier to use. For switches, I wanted to use tactile switches, so I compared Gatheron, Razer, Logitech, Cherry. I opted for the cherry MX blues because they are popular and have good recommendations. I use a 16MHz crystal, in the schematic they use one from IQD frequency, but these are not available anymore, so I needed a different one. I found a couple with 16MHz, so I compared the data sheets and the Abracon one was the most similar. In the schematic they use a reset switch from C&K, but there was no footprint available. There are many of these switches available, but I needed one that is not too tall because it is going to be located on the back of the PCB. I also compared some stabilizers for the bigger keys, I am going to use cherry style stabilizers that are plate mounted because I am going to use Cherry switches and plate mounted because they are really easy to install.



(The delivery date is going to be added later because I did not order yet.)



First you need to find or design a schematic, I found my schematic online. Then you need to choose your components. If you found, your schematic online or somewhere else the components you need are already named on the schematic. You need to check the availability of your components, if a component is not available any more you can check the datasheet of that component to search a similar component. Then you are going to start your schematic in Altium or another program. Some components are already available on Altium with footprint some footprints you need to add yourself. If you search on Octoparts or snap eda you get the symbol with footprint, or you search the footprint in the datasheet of the component. Then you need to connect the wires in Altium. Then you open a PCB design in Altium and add your schematic to that PCB design. Then you can put the components on the PCB and switch them around as you wish. Then you can order the parts you need for the design and order the PCB. If you have received the parts and the PCB you can start soldering. Solder all the components except the diodes and switches. Then you can plug the keyboard in and check all the connections if they have power. Then you can solder the diodes and switches and upload the QMK firmware. Then you can test all the buttons if they do what they are supposed to do. After that the PCB is done. Then you can design the case and key caps on AutoCAD or any cad software. If you are done with that you can print the key caps and case with a 3d printer, or you can have it milled. If you are done with that put all together, and you now have a functioning keyboard.

# Results

[Describe the end result you accomplished.

* Describe every aspect of your device. How does it function?
* Add an image of the electrical schematic, PCB design, finalized mechanical design, and finalized product

Write a well-structured text using subtitles and paragraphs.

**+/-500**]

## MCU and buttons

A keyboard exists of key caps, switches, PCB and a case. The PCB is the electrical board that translates keystrokes into commands that the computer can interpret. A microcontroller and an array of switches is located on the PCB. The MCU compromises a CPU, some RAM, flash memory, and several I/O ports. To mirror the arrangement of real keys, the switches are placed in columns and rows. The MCU will scan the matrix at a high speed by applying a voltage to one column, then the next, and so on reading the line. When the MCU feeds the tension on a key’s column, the voltage will appear on the pushed key’s line because current will travel from the active line to the row through the switch. By looking at the rows and columns it is presently applying and measuring the voltage, the MCU deduces which key was hit. But if one presses more than one key at a time, it is possible for the MCU to register a fake key press. We use diodes between the switch and the row it is attached to prevent this. Diodes are electrical components that block the passage of electricity in the opposite direction.

## Power decoupling

Every +5V pin of the MCU should contain decoupling capacitors, according to the data sheet. For an active IC, the decoupling capacitors are critical. The voltage of the power source will drop if the component begins to draw current while performing its function, which might be detrimental not only for the component but also for all other components powered by the same source. Decoupling capacitors are added to each power pin of the IC to prevent this. Local energy storage will be provided by the decoupling capacitors. When the IC starts to need energy, the capacitors will be able to provide that demand without causing too much damage to the power supply. The decoupling capacitors progressively replenish when the component does not use energy, making them ready for the next serve.

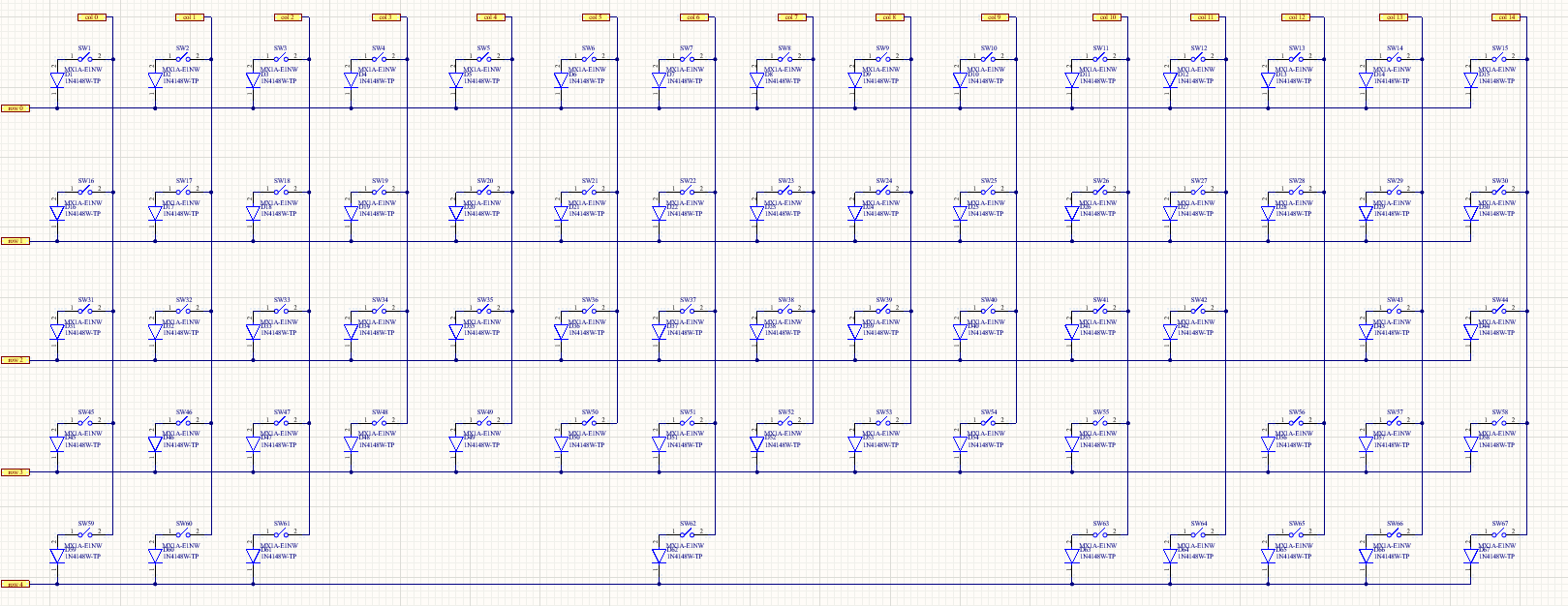
## ISP header

If the MCU fails catastrophically, it may be essential to reprogram it. We can't utilize the USB port in this circumstance (for example, if the DFU boot-loader is missing). We must utilize the ISP programming mode to access the Serial Peripheral Interface (SPI) programming interface.

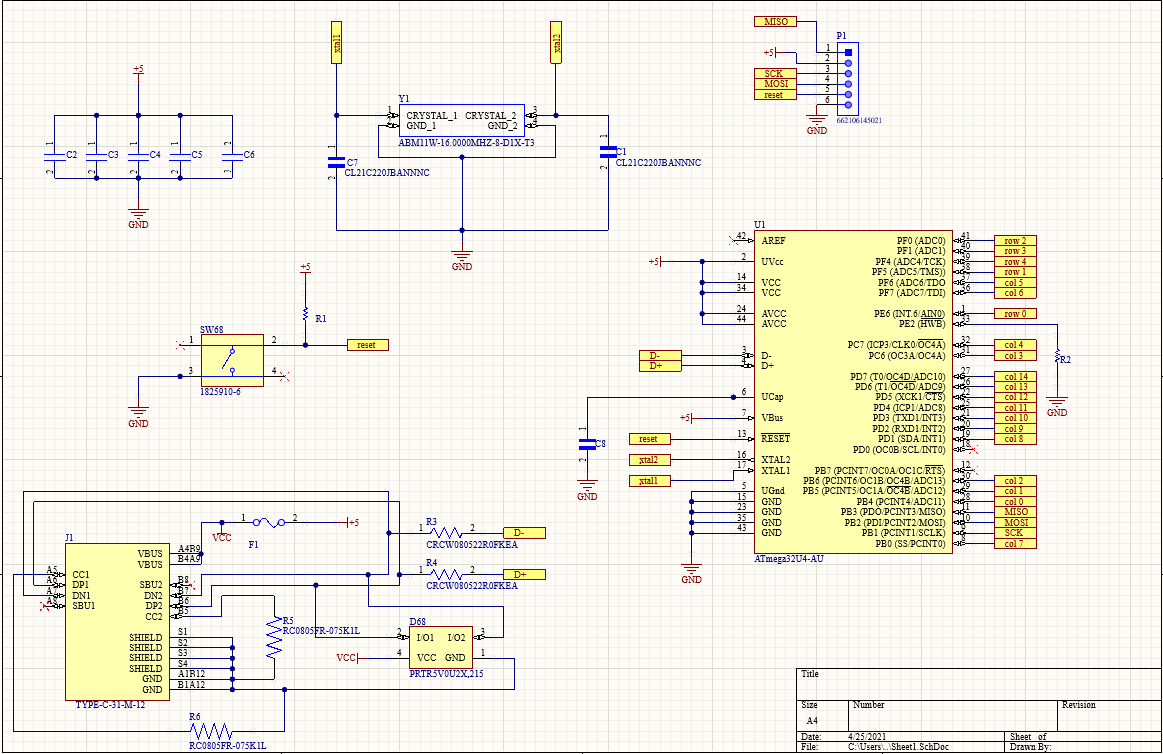
## USB

I don't want the electronic components of the keyboard to be harmed by an electrostatic discharge, thus the USB circuit needs to be as safe as possible for the rest of the keyboard electronics (ESD). This implies the data lines will be ESD protected, and the VCC will be protected from power surges.

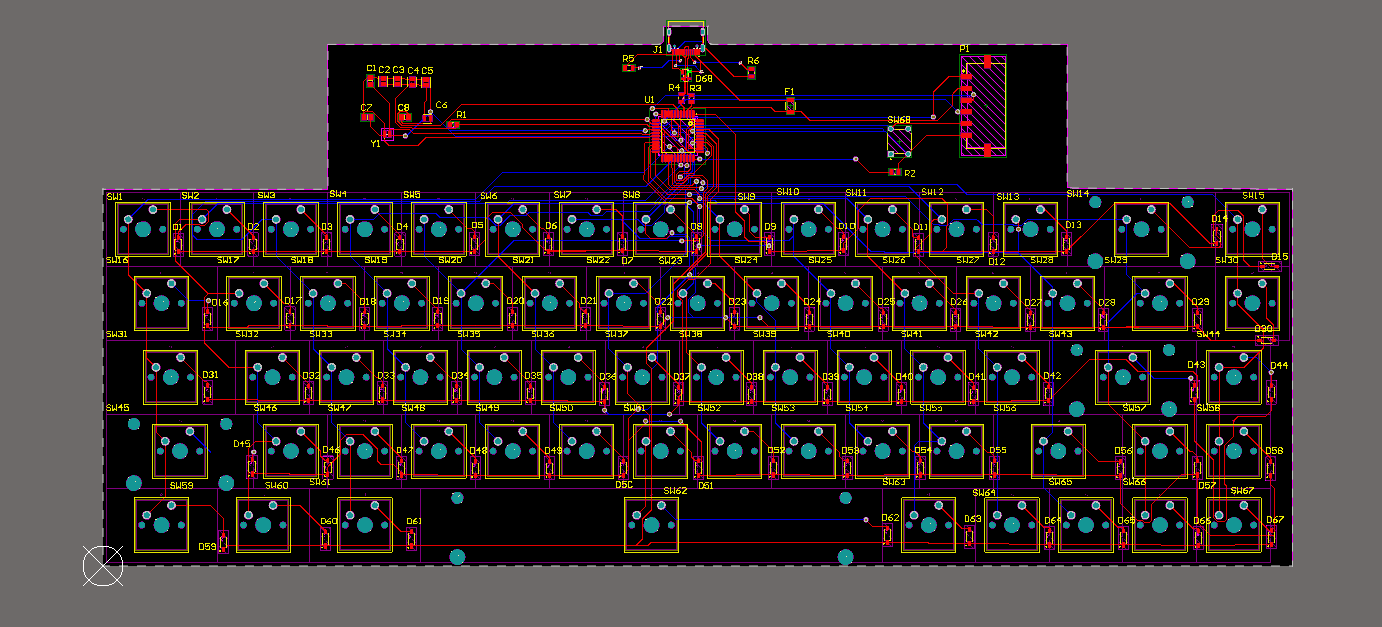
Switch matrix



Schematic



Pcb design



Finalized product



# Discussion

At the start of the project when choosing the electrical schematic, it was hard finding one with the specs I wanted that had all the components available. When putting the design in altium there were some components without footprints or symbols. When routing the PCB the autoroute wasn’t able to do the routing, so it needed to be done by hand. While designing the PCB the board shape needed to change because of how the keyboard is build. Most mechanical keyboards have a plate to separate the PCB from the keys. This one doesn’t have that, so there is no space to put some components on a standard size keyboard PCB. There now is an extra bit at the top of the PCB for the header and chip. When starting the project all the components were available on one site. When ordering the components this wasn’t the case anymore. Ordering the chip and some pushbuttons from rs-online. This order eventually was canceled because for ordering the chip you needed a VAT number. When receiving that message the chip sold out at every website. Then ordered an Arduino Leonardo, on the data sheet they said it was the same chip. When receiving the Arduino it was a newer version then they advertised, so it didn’t have the same chip. Eventually ordering two used kits from Amazon with that chip on it. While designing the case for the PCB the size of the PCB changed, so the design was useless. When soldering the smd components in the oven some diodes flew of, so they needed to be soldered on by hand. The chip arrived too late, so that needed to be soldered on by hand as well.

# Reference list

<https://switchandclick.com/how-to-build-a-keyboard/>

https://www.reddit.com/r/MechanicalKeyboards/