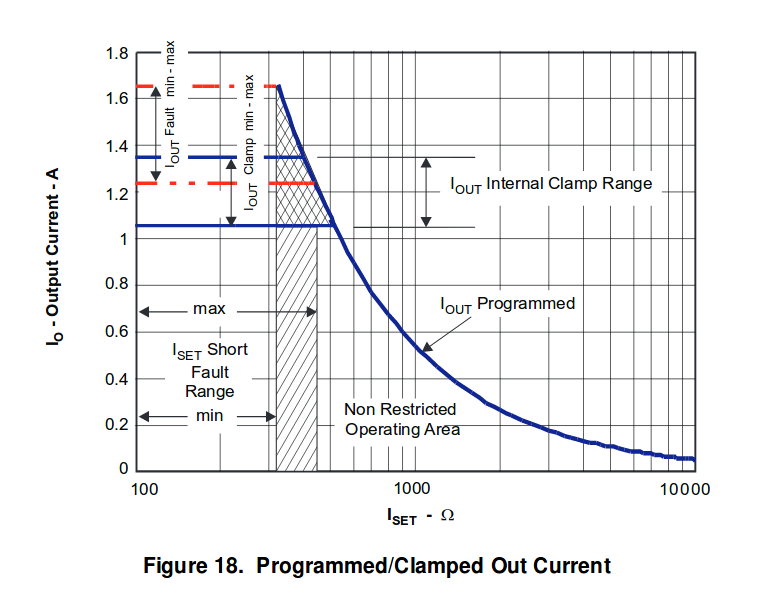
**3 Hardware**

The following chapter deals with the hardware used for the cyberglove. It will explain why how the new developed circuit board works and explains why certain decisions were made in this way.

**3.1 Hardware selection and board design**

The old version of the cyberglove used an Arduino Uno board with a breakout chip for wireless data transmission. It also has an external battery in form of a typical smartphone power bank. Wiring is realized by a breadboard which is equipped with resistors and connects the glove with its components to the GPIO pins of the Arduino Uno. This setup needs a lot of space and is also heavy and unwieldy. In order to get a smaller and more light processing unit for the cyberglove, we need to develop a new hardware basis. The basic functions of the glove should be preserved, so we orient ourselves to the development of the original version.

The Arduino Uno in the previous cyberglove uses a ATmega328 microprocessor which provides 16 MHz and a total amout of 20 GPIO pins, six of them are analog pins and the rest are for digital input and output. We decide to use the ATmega32U4 since it’s similar to the ATmega328 and can be programmed in the same way with the Arduino IDE. Just like the ATmega328, the ATmega32U4 has a clock frequency of 16 MHz and provides up to 32 I/O pins.

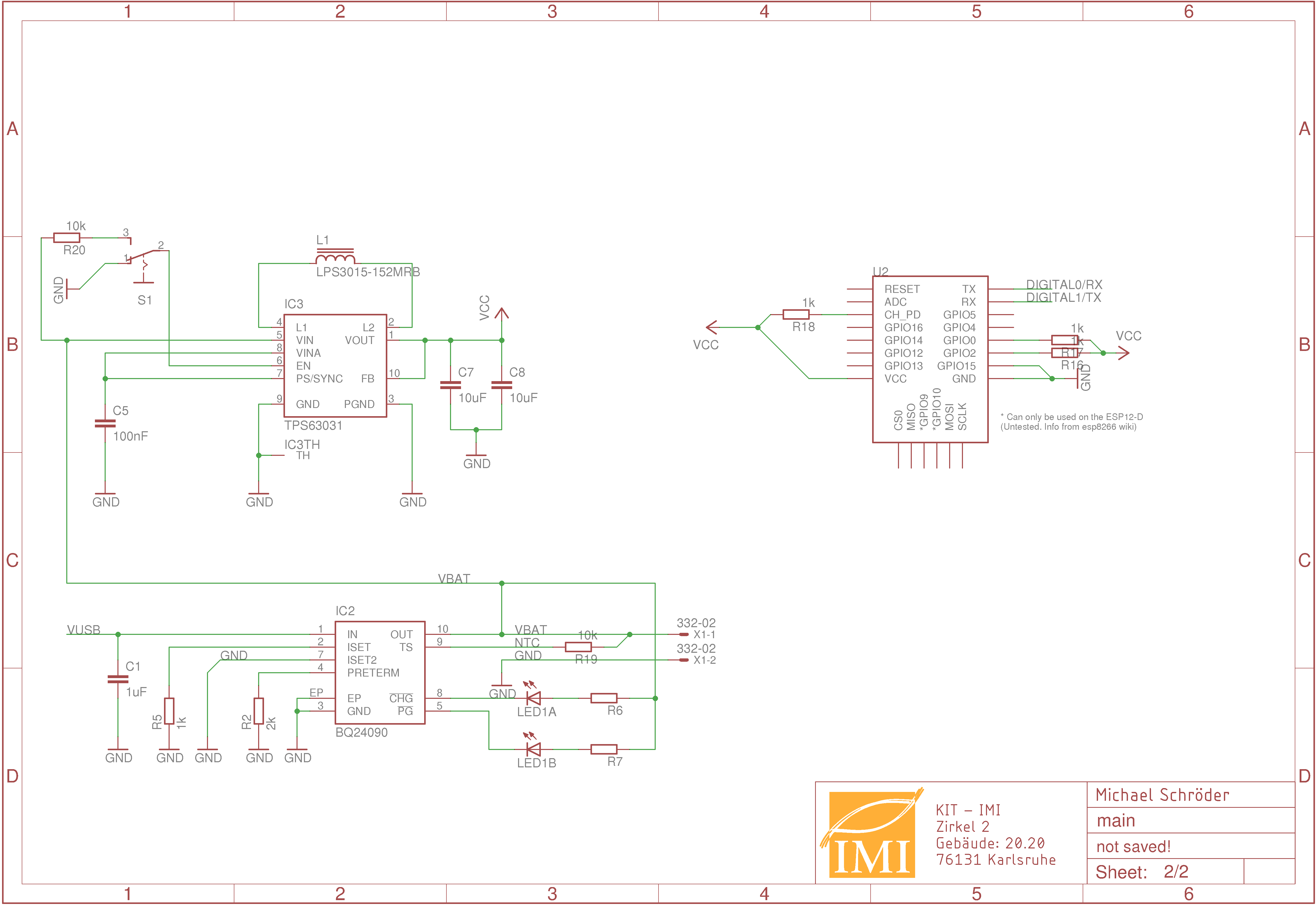
We use an Li-Io battery cell with xxx mAh to power the circuit board. The battery provides 3.7V, so we need a TPS63031 converter which converts the voltage to a constant 3.3V supply voltage for the ATmega32U4. Because we also want to charge the battery over our USB connector, we use the BQ24090 microchip to adjust the charging current. According to the following image, we set the maximum charging current to 500mA by installing a 1k Ohm resistor.

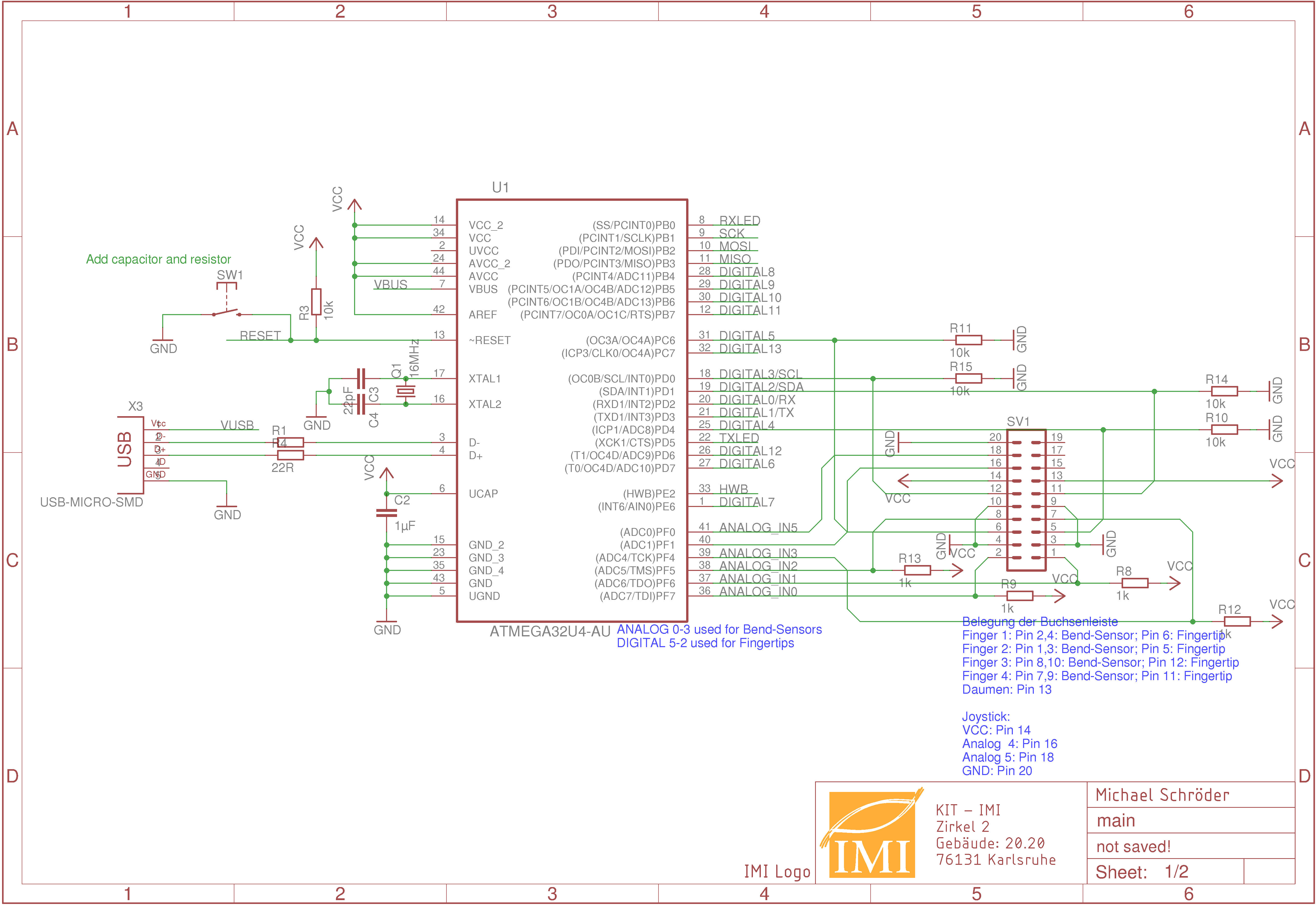
For the Wi-Fi connection we use the same ESP-12E module as the previous cyberglove used.

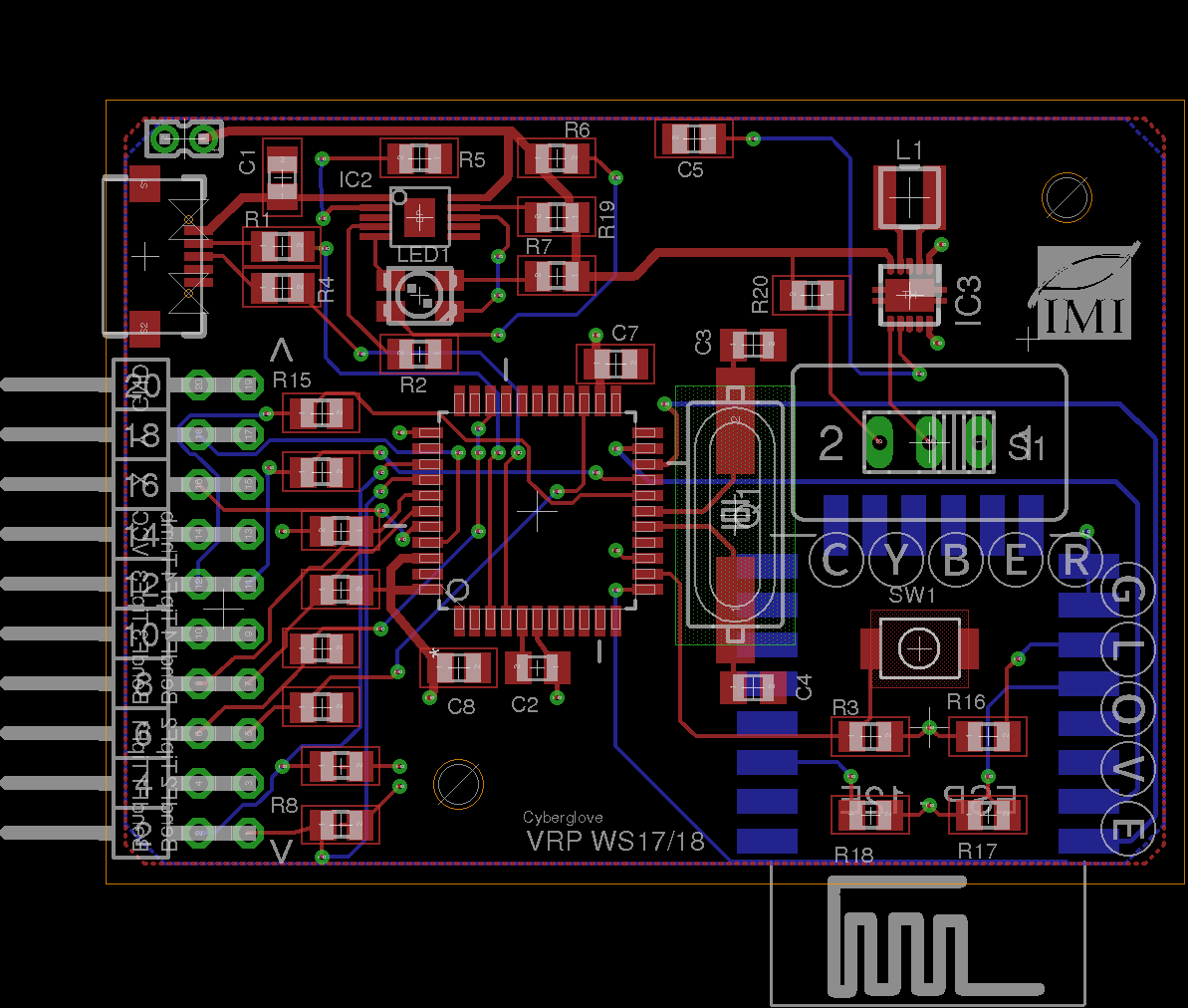
This way we could built up on the previous groups work and had the chance to improve it, instead of reinventing the data transfer, wasting resources.

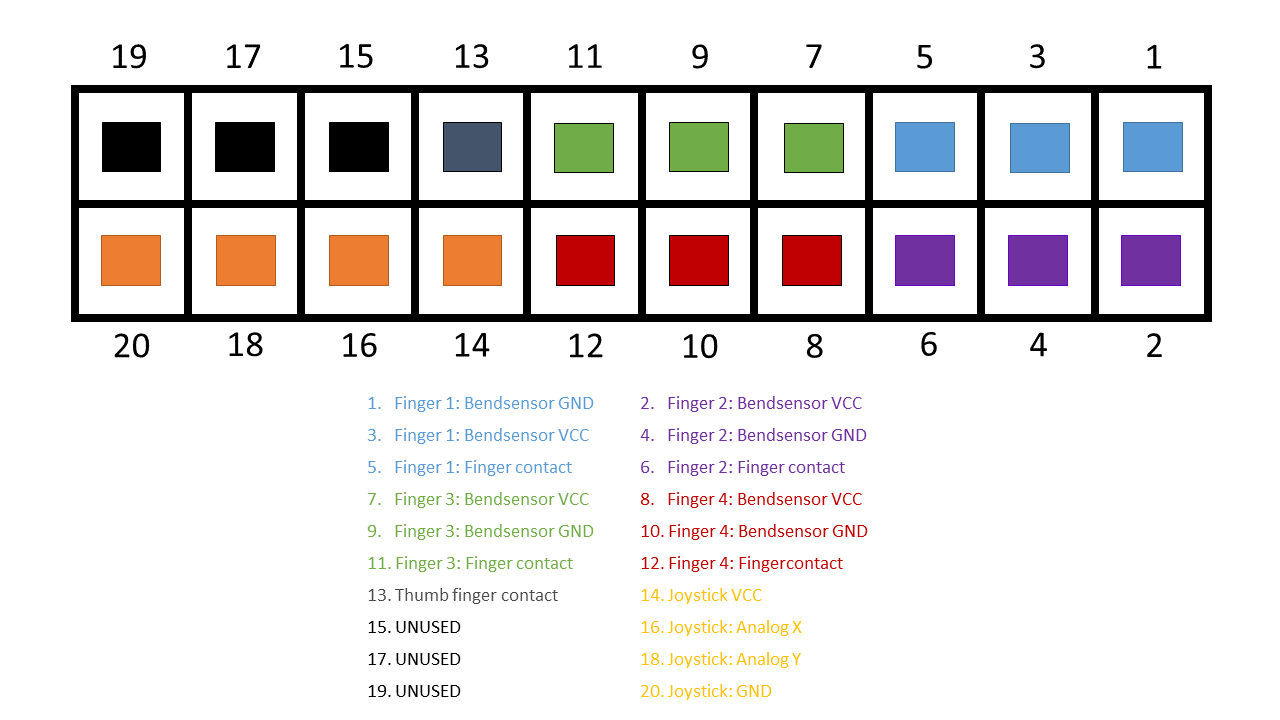
The Wi-Fi module connects the hardware with the CAVE on startup and sends the evaluated output signal of the glove constantly to the CAVE server application.

After selecting the main hardware elements, we created a sketch in Eagle and wired the corresponding connections. Unfortunately, we made a spelling mistake with the USB connection of the microcontroller. That’s why there needs to be an additional connection between pin 7 “VBUS” and the USB connector “VUSB”. The final wiring can be seen in the following images.







We decided to use a bar of connector pins to connect the bendsensors and the fingercontacts with the circuit board. This allows the user to connect the glove with the board easily and disconnect both parts from each other for e.g. reprogramming the board without taking the glove itself with you. Image X.X shows the pin assignment of the pin bar.

3.2 The cyberglove

The glove itself contains the following elements:

* Fingercontacts on all fingers
* Three bendsensors
* A two-axis joystick
* Steel sheets for mounting the hardware case

Putting the fingercontacts together with thumb closes the electric circuit and sends a “high” signal to the microcontroller. These signals can be used by the software to function as four buttons, one for each finger. If one of the bendsensors is bent this increases its resistance, which can be evaluated by the software. This makes it possible to recognize different gestures made by the glove.