

# Wetterstation mit Solar Energie

Bachelor Diplomarbeit 2019

Brugg/Windisch, 15. August 2019

<b>Hochschule</b>	Hochschule für Technik - FHNW
<b>Studiengang</b>	Elektro- und Informationstechnik
<b>Autoren</b>	Mischa Knupfer, Andres Minder
<b>Auftraggeber</b>	Prof. Dr. Taoufik Nouri
<b>Experte</b>	Patrick Strittmatter
<b>Betreuer</b>	Prof. Dr. Taoufik Nouri
<b>Version</b>	2.1

## Abstract

Climate and weather data are the main sources to determine plots for specific plants or plant species for a specific climate. Because of that, farmers need to know the climate and weather data to optimize their work. Swiss farmers have the advantage of getting this information through a federal agency, which does not apply to farmers in subtropical areas. To provide these information for farmers in subtropical areas, a low-priced mobile weather station is required. This project should design a prototype of a weather station, which can record data for air temperature, Rainfall, wind strength and hours of sunshine. In addition to that, the DS3231 real time clock (RTC) should generate a timestamp to the data before it is stored in the data memory. The core of the weather station is the microcontroller ArduinoMega2560, which contains the program code and manages the data storage. The wind strength is being measured by the WD123 from Froggit, the rainfall by an ombrometer from Misol and the air temperature by the BME280 from Bosch that measures also humidity and pressure of the air. Moreover, a wind vane (p/n 80422) from Argent Data Systems allows the weather station to determine the wind direction. Those gained data points are stored on an internal microSD card.

The mobile weather station is able to provide data for temperature, humidity and pressure of the air as well as the Rainfall and the strength and direction of the wind. Furthermore, the mobile weather station is also able to store the data points with timestamp on the microSD card. In a further project, a battery will be implemented which will be supported by photovoltaic. Furthermore, GPS and data query over SMS (GSM) will also be implemented in that further project.

Key Words: mobile weather station, sensors, microSD card, GPS, SMS (GSM)

## Inhaltsverzeichnis

<b>1 Einleitung</b>	<b>1</b>
1.1 Einleitung . . . . .	2
1.2 Auftragsbeschreibung . . . . .	3
1.3 Ziele . . . . .	4
1.3.1 Projekt 5 . . . . .	4
1.3.2 Projekt 6 . . . . .	5
1.4 Konzept . . . . .	6
1.4.1 Sensoren . . . . .	7
1.4.2 RTC . . . . .	7
1.4.3 Datenspeicherung . . . . .	8
1.4.4 Kommunikationsmodul . . . . .	8
1.4.5 Energieversorgung . . . . .	9
<b>2 Firmware</b>	<b>11</b>
2.1 Development Environment . . . . .	12
2.1.1 In-System-Programmer . . . . .	12
2.2 UML-Diagramm . . . . .	14
2.2.1 Lizenzen . . . . .	15
2.3 Validierung der Firmware . . . . .	16
2.4 Manual . . . . .	17
2.4.1 USB-to-Computer . . . . .	17
2.4.2 SMS-Abfrage . . . . .	17
<b>3 Hardware</b>	<b>19</b>
3.1 Übersicht . . . . .	20
3.2 MCU . . . . .	24
3.2.1 MCU - Projekt 5 . . . . .	24
3.2.2 Ergänzungen aus der Bachelor-Thesis . . . . .	25
3.3 RTC . . . . .	26
3.3.1 RTC - Projekt 5 . . . . .	26
3.4 Sensoren . . . . .	28
3.4.1 Ombrometer - Projekt 5 . . . . .	28
3.4.2 Anemometer - Projekt 5 . . . . .	33

3.4.3	Windrichtungsgeber - Projekt 5 . . . . .	34
3.4.4	BME280 - Projekt 5 . . . . .	35
3.5	Ergänzungen aus der Bachelor-Thesis . . . . .	36
3.5.1	Lichtintensitätssensor . . . . .	37
3.5.2	BME280 . . . . .	38
3.6	Datenspeicherung . . . . .	39
3.6.1	Datenspeicherung - Projekt 5 . . . . .	39
3.6.2	Ergänzungen aus der Bachelor-Thesis . . . . .	40
3.7	Kommunikationsmodule . . . . .	41
3.7.1	SIM808 . . . . .	41
3.8	Energieversorgung . . . . .	43
3.8.1	Berechnung der Akkukapazität . . . . .	43
3.8.2	Einfluss der Photovoltaik . . . . .	44
3.8.3	ChargePump und Linearregler . . . . .	45
3.9	PCB . . . . .	47
3.9.1	Das Bestücken der Leiterplatte . . . . .	47
3.9.2	Mängel der Erstversion und dessen Verbesserungen . . . . .	48
3.10	Das Gehäuse . . . . .	50
3.10.1	Wichtige Merkmale . . . . .	50
3.10.2	Das Design . . . . .	51
<b>4</b>	<b>Abschluss</b>	<b>57</b>
4.1	Konzeptvalidierung . . . . .	58
4.1.1	Validierung der Hardware . . . . .	58
4.2	Schluss . . . . .	65
4.3	Authentizitätserklärung . . . . .	66
<b>Literatur</b>		<b>67</b>
<b>Tabellenverzeichnis</b>		<b>68</b>
<b>Abbildungsverzeichnis</b>		<b>68</b>
<b>Anhang</b>		<b>71</b>
<b>A Lastenheft</b>		<b>72</b>

<b>B Aufgabenstellung</b>	<b>73</b>
<b>C Lizenz-Texte</b>	<b>74</b>
C.1 Adafruit_BME280 . . . . .	74
C.2 RTClib . . . . .	74
<b>D Konzept Posterbild</b>	<b>76</b>
<b>E Schema</b>	<b>77</b>
<b>F PCB-Layout</b>	<b>82</b>

# **1 Einleitung**

## 1.1 Einleitung

Pflanzen benötigen eine ihnen entsprechende Umwelt. Durch meteorologische Messdaten kann diese ermittelt und durch Agronome optimal bewirtschaftet werden. Ausserdem können bei genügend Messdaten aus einem Erfassungsnetz Wetterprognosen erstellt werden. Die Erhebung solcher Messdaten trägt somit erheblich zum wirtschaftlichen Erfolg in der Agronomie bei und kann bei einem geeignet grossen Erfassungsnetz Bewohner vor Unwetter warnen. In den ärmeren Teilen Afrikas und Südamerikas sind solche Systeme jedoch nicht verbreitet. Aus diesem Grund wird eine kostengünstige, erweiterbare und mobile Wetterstation realisiert, welche die örtlichen Agronomen unterstützen kann. Diese Wetterstation kann die Niederschlagsmenge, die Windstärke & -richtung, die Lufttemperatur, -druck, -feuchtigkeit und die Sonnenstunden messen. Außerdem kann der Akkumulator der Wetterstation mittels Photovoltaik geladen werden. Die erhobenen Daten sind via SMS<sup>1</sup> abrufbar und mit einem GPS-Modul kann der Standort der mobilen Wetterstation erfasst werden. Zusätzlich kann mittels eines Terminals (z.B. PuTTY) eine Verbindung über eine serielle Schnittstelle über den USB 2.0 micro-B Anschluss der Wetterstation erstellt und mit ihr kommuniziert werden.

Mittels Ombrometer mit Kipplöffelprinzip wird die Niederschlagsmenge ermittelt. Die Windstärke wird über ein Schalenkreuzanemometer und die Windrichtung über einen Windrichtungsgeber eruiert. Über einen RJ-11 Stecker werden diese dann an die Wetterstation angeschlossen und die analogen Signale von einer Mikrocontrollereinheit ausgewertet. Die Lufttemperatur, -druck und -feuchtigkeit werden über den digitalen Kombinationssensor BME280 von *Bosch* und die Beleuchtungsstärke (Lux) zur Ermittlung der Sonnenstunden über den Light-To-Digital Converter TSL2561 von *TAOS<sup>2</sup>* gemessen. Um den Messdaten einen Zeitstempel beifügen zu können, wird die Extremely Accurate Real Time Clock DS3231 von *Maxim Integrated* verwendet. Diese drei digitalen Elemente sind über das I<sup>2</sup>C-Interface mit der Mikrocontrollereinheit verbunden. Zur Datenspeicherung wird eine  $\mu$ SDHC-Karte verwendet. Im Kommunikationsmodul SIM808 von *SIM Com* werden das GSM/GPRS- und GPS-Modul vereint.

Die (Teil-)Realisierung<sup>3</sup> dieser Wetterstation wurde in zwei Projekten an der Hochschule für Technik der FHNW im Studiengang Elektro- und Informationstechnik durchgeführt. Diese bestehen aus einem vorangehenden Projekt 5, sowie dem anschließenden Projekt 6 (Bachelor Diplomarbeit) und wurden vom gleichen Team bearbeitet. Dieser Bericht umfasst die Inhalte beider Projekte. Welche Aspekte in welchem Projekt behandelt wurden, können dem Kapitel 1.3 Ziele direkt entnommen werden.

### Berichtaufbau

---

<sup>1</sup>Mittels GSM/GPRS-Modul

<sup>2</sup>Texas Advanced Optoelectronic Solutions

<sup>3</sup>Es wird in weiterführenden Projekten weiter daran gearbeitet

## 1.2 Auftragsbeschreibung

Das Wetter spielt eine wichtige Rolle in der Agronomie. Regnet es nicht genug, müssen Pflanzen bewässert werden. Trifft auf ein Ort nur wenig Sonnenlicht, so sollten dort nicht die Pflanzen, welche viel Sonnenlicht brauchen, angebaut werden. Windet es zu stark, können Pflanzen beschädigt oder gar zerstört werden. Ist es tagsüber heiß, so benötigen die Pflanzen mehr Wasser. Hiesige Bauern besitzen den Luxus von guten Wettervorhersagen dank dem Bundesamt für Meteorologie und Klimatologie (MeteoSchweiz). Dieser Luxus ist in anderen Ländern noch nicht gegeben. Prof. Dr. Nouri Taoufik ist aufgefallen, dass in subtropischen Gegenden wie teile Südamerikas oder Afrikas dieser Luxus ebenso fehlt.

Aus diesem Grund soll eine kostengünstige, erweiterbare und mobile Wetterstation entwickelt werden, welche diese Bauern unterstützt. Diese Wetterstation soll die Regenmenge, die Windstärke, die Lufttemperatur und die Sonnenstunden messen können. Außerdem soll die Wetterstation mittels Photovoltaik unterstützt werden, und erhobene Daten via SMS abrufbar sein.

Die detaillierte Aufgabenbeschreibung ist im Anhang A Lastenheft und Anhang B Aufgabenstellung aufgeführt.

## 1.3 Ziele

Die Ziele sind strikt aufgeteilt in die zwei Projekte 5 und 6. Darin enthalten sind die jeweiligen zu erreichenden Muss- und Wunschziele mit ihren quantifizierten Spezifikationen.

### 1.3.1 Projekt 5

Tabelle 1.1: Ziele P5

	Ziel	Messbereiche	Genauigkeiten	Einheiten
<b>Mussziele</b>				
Sensoren	Lufttemperaturmessung	[-20;60]	$\pm 1$	°C
	Windgeschwindigkeitsmessung	[10;25]	$\pm 1$	m/s
	Niederschlagsmenge	Wasser	$\pm 100$	ml/m <sup>2</sup>
Datenspeicherung	Datenabfrage via PuTTY	$\geq 9600$		Bd/s
RTC	Implementation	Echtzeit	$\pm 1$	s/Jahr
<b>Wunschziele</b>				
Sensoren	Sonnenstunden Prototyp	Echtzeit		s

Tabelle 1.1 zeigt diverse Ziele im P5, unterteilt in Muss- und Wunschziele. Zu den Musszielen gehören die Lufttemperaturmessung, die Windgeschwindigkeitsmessung, die Niederschlagsmessung, die Implementation des RTC und die mögliche Datenabfrage via PuTTY vom Datenspeicher ( $\mu$ SD-Karte). Die Temperatur soll zwischen -20 bis 60 °C mit einer Genauigkeit von  $\pm 1$  °C ermittelbar sein. Die Windgeschwindigkeitsmessung soll vor allem stärkere Windgeschwindigkeiten erfassen, um vor Sturm warnen zu können. Deshalb können niedrigere Windgeschwindigkeiten vernachlässigt werden. Die Windgeschwindigkeit sollte zwischen 10 und 25 m/s auf  $\pm 1$  m/s genau gemessen werden. Zusätzlich wird die Stärke der Windgeschwindigkeit in der Beaufortskala zugeordnet. Die Niederschlagsmenge wird nur für Regenwasser bestimmt, mit einer Genauigkeit von  $\pm 100$  ml/m<sup>2</sup>. Andere Niederschlagstypen wie Hagel oder Schnee werden nicht berücksichtigt.

### 1.3.2 Projekt 6

Tabelle 1.2: Ziele P6

	Ziel	Spezifikation	Genauigkeit	Einheit
<b>Mussziele</b>				
Speisung	Akkulaufzeit	$\geq 100$		h
	Mittels DC-Ladekabel ladbar	5.5 / 2.1mm DC-Stecker		
	Photovoltaik	1		Akkuladungen/Tag
Kommunikationsmodule	GPS-Modul Standortupdate	< 5		Minuten
	GSM-Modul	SMS: Senden und Empfangen		
Sensoren	Sonnenstunden (Lichtstromdichte)	0.1 - 90'000		lx
PCB	Herstellung eines 4-Lagen Prints			
<b>Nichtziele</b>				
Sensoren aus Projekt 5	Keine Neukonzeption			
Wartungsfreiheit	Wartungsfenster	>7		Tag
<b>Wunschziele</b>				
Kommunikationsmodule	Einbindung in IoT	Bluetooth    WLAN		
Speisung	Akku leicht austauschbar			
	Mittels USB ladbar	USB 2.0 (Mini-B    Micro-B)		

Tabelle 1.2 zeigt die Muss-, Nicht- und Wunschziele des Projekts auf. Die Akkulaufzeit und das Laden mittels DC-Ladekabel sind spezifiziert. Ausserdem wird ein volles Aufladen des Akkus innerhalb eines Tages via Photovoltaik als Ziel gesetzt. Nicht Ziel des Projekts ist es, die in einem vorhergehenden Projekt benutzten Sensoren neu zu konzipieren, sowie die mobile Wetterstation komplett Wartungsfrei zu machen. Eine wöchentliche Wartung ( $\leq 7$  Tage) wird als vertretbar bis notwendig erachtet. Als Wunschziel folgt die Einbindung der mobilen Wetterstation in das *IoT*, ein leicht austauschbarer Akku, sowie die Möglichkeit, den Akku in der mobilen Wetterstation via USB-Kabel zu laden.

## 1.4 Konzept

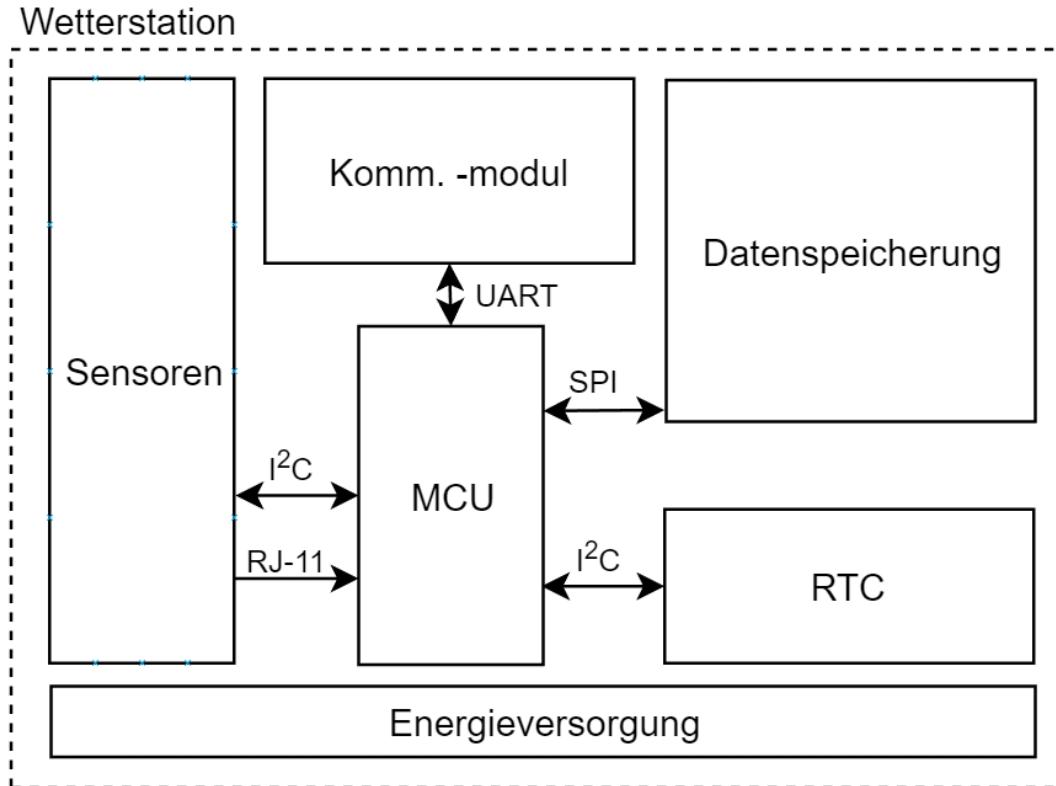


Abbildung 1.1: Konzeptionelle Übersicht

Das Gesamtkonzept ist, wie in der Abbildung 1.1 grafisch dargestellt, modular aufgebaut. Die gestrichelte Linie beschreibt die Systemgrenze der Wetterstation. Als Zentralrecheneinheit wird eine *Micro-Controller-Unit (MCU)* verwendet. Diese ist dafür verantwortlich, dass die Daten richtig verarbeitet und an die entsprechenden Module weitergeleitet werden. Die Messdaten werden in digitaler Form über das I<sup>2</sup>C-Interface vom Modul *Sensoren* an die *MCU* übertragen. Über die RJ-11 Anschlüsse werden analoge Signale übermittelt. Dieser fügt mit dem *Real-Time-Clock (RTC)* einen Timestamp hinzu, wobei anschließend die Daten in der *Datenspeicherung* nichtflüchtig gespeichert werden. Über das *Kommunikationsmodul* können dann die Daten von Nutznießern abgefragt werden.

### 1.4.1 Sensoren

In dem Block *Sensoren* (Abbildung 1.2) sind alle Messeinheiten untergebracht. Die Idee dieses Blockes besteht darin, dass dieser adaptiv ist und somit leicht erweitert werden kann.

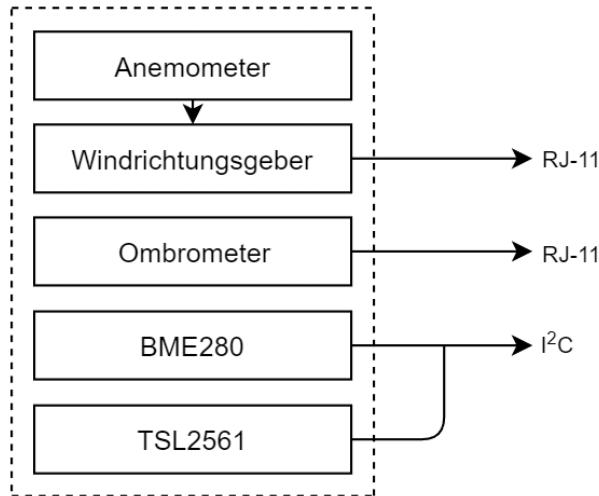


Abbildung 1.2: Sensoren

Der Lufttemperatur, -druck und -feuchtigkeits kombinierte Sensor BME280 und der Light-To-Digital Converter TSL2561 sind beide über das I<sup>2</sup>C-Interface mit der *MCU* verbunden. Das Anemometer kann direkt mit dem Windrichtungsgeber über die vorhandene RJ-11 Buchse angeschlossen werden, wobei dann der Ausgang des Windrichtungsgebers beide analogen Signale überträgt. Das Ombrometer wird separat angeschlossen.

### 1.4.2 RTC

Die Extremely Accurate Real Time Clock DS3231 (RTC) wird nur zur Zeitabfrage für einen Timestamp verwendet. An die *MCU* ist sie über das I<sup>2</sup>C-Interface angeschlossen. In der Abbildung 1.3 ist noch ein zusätzlicher Block CR2032 zu erkennen, der eine Knopfbatterie beschreibt. Sie hat den alleinigen Zweck, dass wenn mal keine Boardspeisung mehr vorhanden und somit der Akkumulator der Wetterstation leer ist, das RTC noch immer Speisung hat. Das RTC läuft dann weiter und muss bei einem kurzzeitigen Ausfall nicht gleich zurückgesetzt werden.

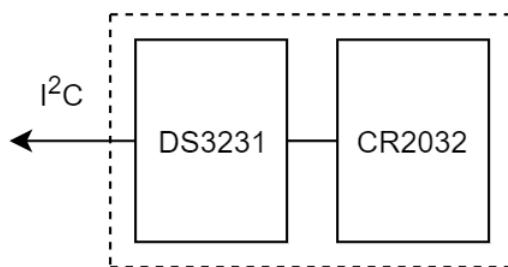


Abbildung 1.3: Real-Time-Clock (RTC)

### 1.4.3 Datenspeicherung

Die nichtflüchtige Datenspeicherung erfolgt auf einer  $\mu$ SD-Karte. Dabei werden die Wetterdaten pro Minute einmal in einem \*.txt File gespeichert. Zu den gespeicherten Daten zählen:

- Lufttemperatur, -druck, feuchtigkeit
- Windgeschwindigkeit, -stärke
- Regenmenge
- Bestrahlungsstärke
- GPS-Koordinaten
- Speicherzeitpunkt

Der Vorteil bei der Speicherung auf einer  $\mu$ SD-Karte ist, dass bei einem Kompletausfall der Wetterstation die Daten trotzdem problemlos mittels entfernen der  $\mu$ SD-Karte extrahiert werden können.

### 1.4.4 Kommunikationsmodul

Abbildung 1.4 zeigt die verschiedenen Schnittstellen, über welche Daten mit der Umgebung (User) und *MCU* ausgetauscht werden können.

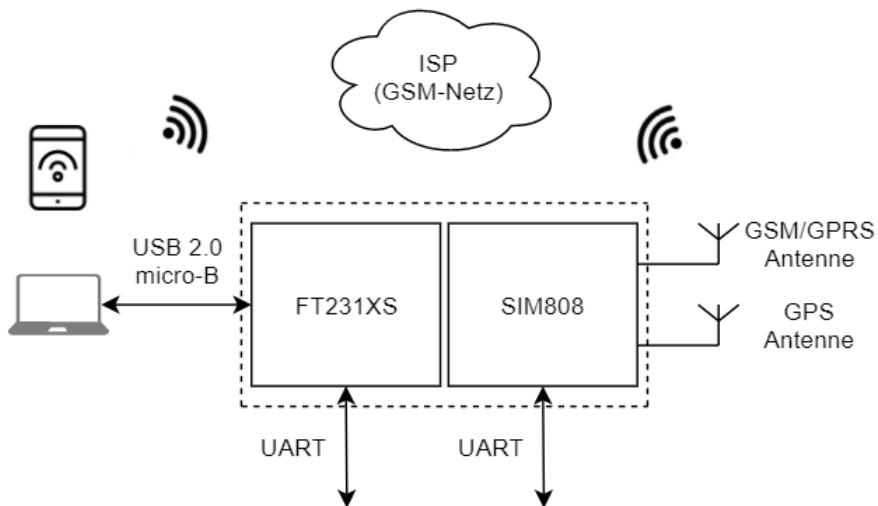


Abbildung 1.4: Kommunikationsmodul

Der User hat die Möglichkeit, per Computer die Daten der Wetterstation direkt abzufragen, oder aber auch über sein Mobilfunktelefon mittels SMS. Dafür benötigt der User eine zusätzliche SIM-Karte vom lokalen Internet Service Provider<sup>4</sup> (ISP) für die Wetterstation, welche er dann mittels Computer über die serielle Schnittstelle entsperren kann.

<sup>4</sup>Z.B. Swisscom, Salt, Sunrise, usw.

### 1.4.5 Energieversorgung

In der Abbildung 1.5 ist die Systemgrenze (gestrichelte Linie) der mobilen Wetterstation ersichtlich. Innerhalb dieser Systemgrenze befinden sich der *Akkumulator*, der *MCP73871* Ladechip, der *LM1117 3.3V* Linearregler und der *LMC7660* Spannungswandler. Ausserhalb der Systemgrenze sind die Photovoltaikanlage, rsp. das Solarpanel, das 230V AC-Stromnetz und ein Computer. Das Solarpanel, sowie auch das Stromnetz können über einen DC Jack Stecker mit den Dimensionen 5.5mm Aussen- und 2.1mm Innendurchmesser angeschlossen werden. Zusätzlich ist der Akkumulator ebenfalls über den USB 2.0 micro-B Anschluss mittels Computer ladbar. Somit kann die Wetterstation auch bei ungenügender Sonneneinstrahlung geladen werden. Es ist möglich, beide Anschlüsse gleichzeitig angeschlossen zu haben.

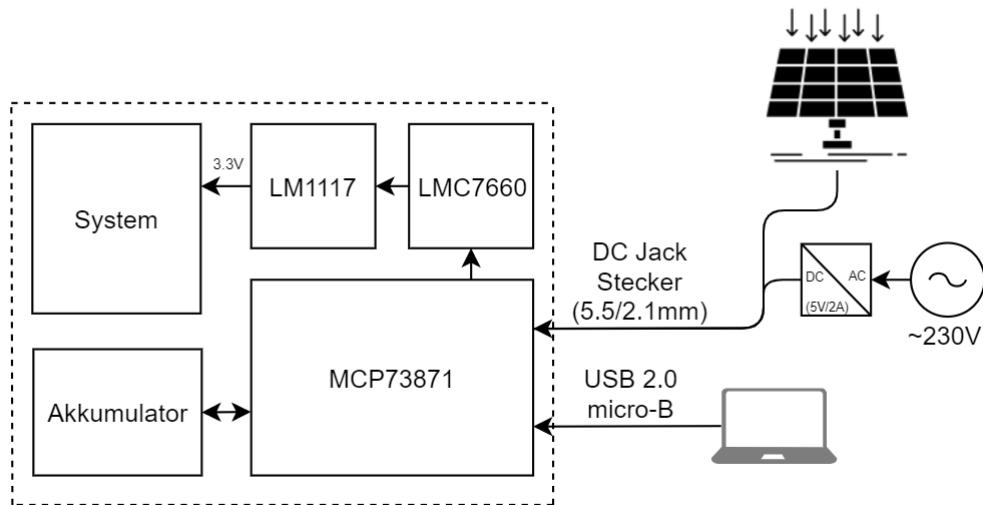


Abbildung 1.5: Energieversorgung

Der *Akkumulator* bildet das Kernstück der Energieversorgung, da dieser die Quelle für die mobile Wetterstation ist. Er dient als Stützung der Versorgungsspannung (3.3V), falls die Wetterstation nur über das Solarpanel betrieben wird, damit sie auch in der Nacht funktionstüchtig bleibt. Der *MCP73871* ist der Ladechip. Dieser reguliert den Ladestrom in Abhängigkeit des Laststromes vom System. Damit der *LM1117* konstant 3.3V als Systemspannung generieren kann, wird der *LMC7660* als positiver Spannungsmultiplizierer vorgeschaltet. Der Funktionsblock *System* beinhaltet alle Bauteile der Wetterstation, welche mit 3.3V Versorgungsspannung betrieben werden müssen. Da die Wetterstation auf Low Consumption ausgelegt wird, sind dies fast alle Bauelemente.



## **2 Firmware**

## 2.1 Development Environment

Die Firmware der Wetterstation wurde im AtmelStudio 7 der Atmel Corp. und der Arduino IDE geschrieben. Die Arduino IDE wurde hauptsächlich verwendet, um notwendige Librarys über dessen *Library Manager* hinzuzufügen. Dies war besonders für die Sensoren BME280 und TSL2561, sowie für die RTC DS3231 und das Schreiben/Lesen der µSD-Karte notwendig. Anschliessend konnte das gesamte Projekt ins AtmelStudio importiert werden. Grund für diese Importierung ist, dass über AtmelStudio ein ISP (In-System-Programmer) verwendet werden kann. Dies ist mit der Arduino IDE nicht möglich. Der Vorteil bei der Verwendung eines ISP's liegt darin, dass der Mikrocontroller auch nach dem Einbetten in die Hardware programmierbar bleibt. Das Programm wurde in der C/C++ Syntax objektorientiert geschrieben und aufgebaut, somit ist die Firmware der Wetterstation skalierbarer und übersichtlicher.

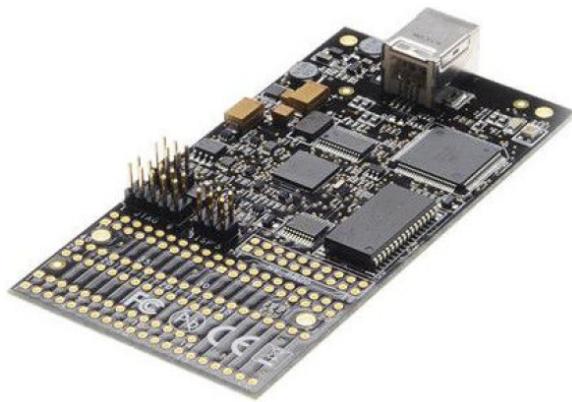
Zur Übersicht der verwendeten Toolchain dient die Tabelle 2.1. Zusätzlich sind auch die Downloadlinks der aktuellen Versionen darin enthalten. Die gesamte Toolchain ist als Freeware erhältlich.

**Tabelle 2.1:** Toolchain

Programm	Version	Link
AtmelStudio 7	7.0.1645	<a href="https://www.microchip.com/mplab/atv-support/atmel-studio-7">https://www.microchip.com/mplab/atv-support/atmel-studio-7</a>
Arduino IDE	1.8.5	<a href="https://www.arduino.cc/en/main/software">https://www.arduino.cc/en/main/software</a>
PuTTY	0.71	<a href="https://www.putty.org/">https://www.putty.org/</a>

### 2.1.1 In-System-Programmer

Als ISP wird der AVR Dragon von Atmel verwendet. Dieser ist ein kostengünstiges Entwicklungstool, welches alle Programmiermodi für die Atmel AVR-Device Familien (z.B. der verwendete atmega2560 Mikrocontroller, siehe **Referenz**) unterstützt. Er enthält auch eine vollständige Debugging-Unterstützung für die meisten AVR-Devices. Diese wurde aber in diesem Projekt nicht genutzt. Der AVR Dragon wird über das USB 2.0 Kabel (Typ B) gespiesen, mit welchem er an den Computer angeschlossen wird. [1]



**Abbildung 2.1:** AVR Dragon [1]

### Programmierinterface

Der AVR Dragon kann direkt mit dem Target Board, also der Wetterstation verbunden werden. Dafür werden vom AVR Dragon verschiedene Programmierinterfaces unterstützt: [1]

- SPI Programming
- High Voltage Serial Programming
- Parallel Programming
- JTAG Programming (*D*)
- PDI Programming (*D*)
- aWire Programming (*D*)

Jene, welche mit einem (*D*) gekennzeichnet sind, dienen auch als Debugging Interface. In Bezug auf das Target Board wird der atmega2560 Mikrocontroller der Wetterstation über das SPI Interface programmiert. Der dafür vorgesehene SPI Header (6-Pin Header) ist auf dem AVR Dragon bereits gemounted, wodurch keine Lötarbeiten erforderlich sind (Abb. 2.2, rotes Rechteck).

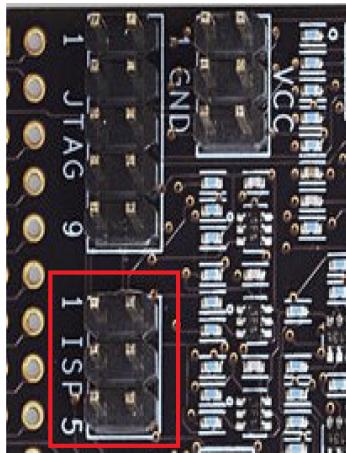


Abbildung 2.2: SPI Header [1]

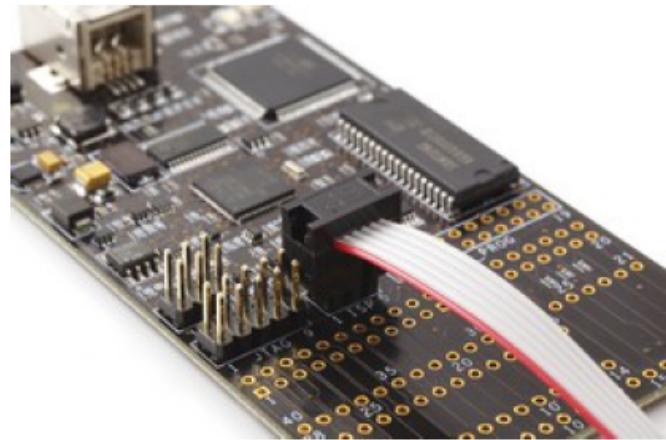


Abbildung 2.3: Anschluss [1]

Beim Anschließen des Target Boards sollte darauf geachtet werden, dass die Speisespannung des Targets (3.3V) an dem Pin 2 des Headers angeschlossen ist. Somit kann der AVR Dragon mittels internem Level Converter alle Signale zwischen dem Target Board und sich selbst dem Target anpassen. Das Flachbandkabel sollte also wie in der Abbildung 2.3 angeschlossen werden.

**Zu Beachten:** Da die µSD-Karte auf dem Target Board auch über das SPI Interface an den Mikrocontroller angeschlossen ist, so sollten die Jumper beim Anschließen entfernt werden. Es könnte zu Komplikationen führen, wenn Peripherie beim Flash-Vorgang noch an das SPI Interface angeschlossen ist.

## Device Programming

## 2.2 UML-Diagramm

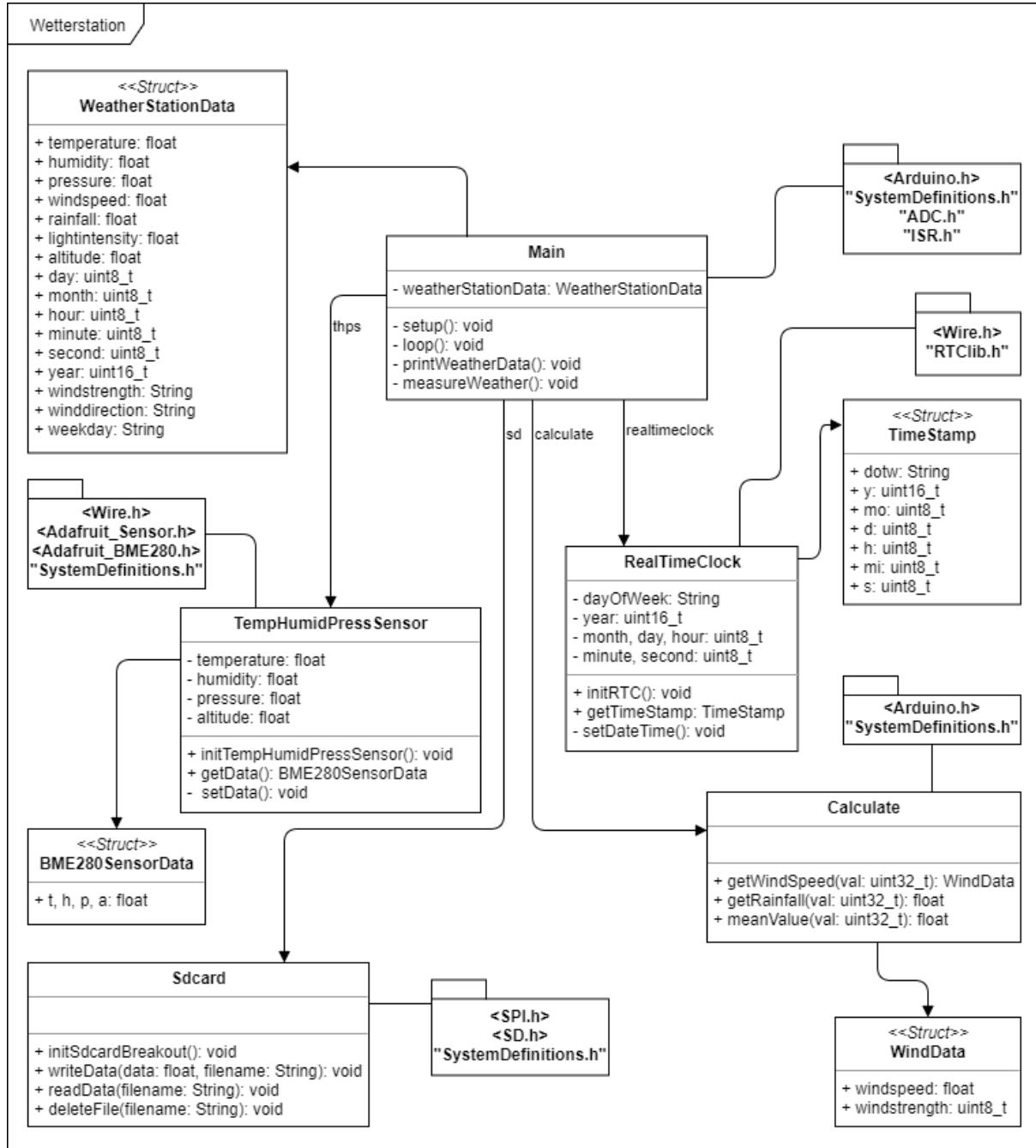


Abbildung 2.4: UML-Diagramm

Das in der Abbildung 2.4 dargestellte UML-Diagramm soll den Aufbau und die logischen Zusammenhänge der Firmware visualisieren. Darin wird gezeigt, welche Headerfiles bei den Klassen benötigt werden. Zudem sind die Attribute, wie auch die Funktionen mit ihren Access Specifier, Argumenten und Rückgabewerten aufgelistet.

Durch diese Struktur ist es möglich, adaptiv mehrere Komponenten hinzuzufügen und anzupassen. Zudem könnten somit auch mehrere Sensoren vom gleichen Typ ohne grossen weiteren

Aufwand implementiert werden.

### 2.2.1 Lizenzen

Die verwendeten Librarys sind grundsätzlich aus der Arduino IDE von Arduino selbst ist eine aus Soft- und Hardware bestehende Physical-Computing-Plattform, bei welcher beide Komponenten auf Open-Source-Basis quelloffen sind [2]. Nach Aussage von Arduino selbst, stehen alle C/C++ Mikrocontroller Librarys unter der **LGPL** [3]. Einige Lizenz-Texte stehen im Anhang C.

**Tabelle 2.2:** Lizenzen

Library	Author	Lizenz
Arduino	•	•
SPI	•	•
Wire	•	•
SoftwareSerial	•	•
SD	•	•
RTClib	•	•
Adafruit_Sensor	•	•
Adafruit_BME280	•	•
Adafruit_FONA	•	•
Adafruit_TSL2561_U	•	•

### 2.3 Validierung der Firmware

## **2.4 Manual**

### **2.4.1 USB-to-Computer**

### **2.4.2 SMS-Abfrage**



### **3    Hardware**

### 3.1 Übersicht

Um eine bessere Vorstellung über die Wetterstation zu erhalten, wird hier das noch unbestückte PCB grafisch dargestellt und die verschiedenen Komponenten erläutert. Um die Übersichtlichkeit wahren zu können, werden in verschiedenen Bildern die Hauptbereiche mit einem roten Rahmen markiert und mit einer Zahl, jeweils neu beginnend bei 1, versehen. Wichtige Teilbereiche werden ebenfalls markiert und mit einem Buchstaben versehen, jeweils neu beginnend bei A.

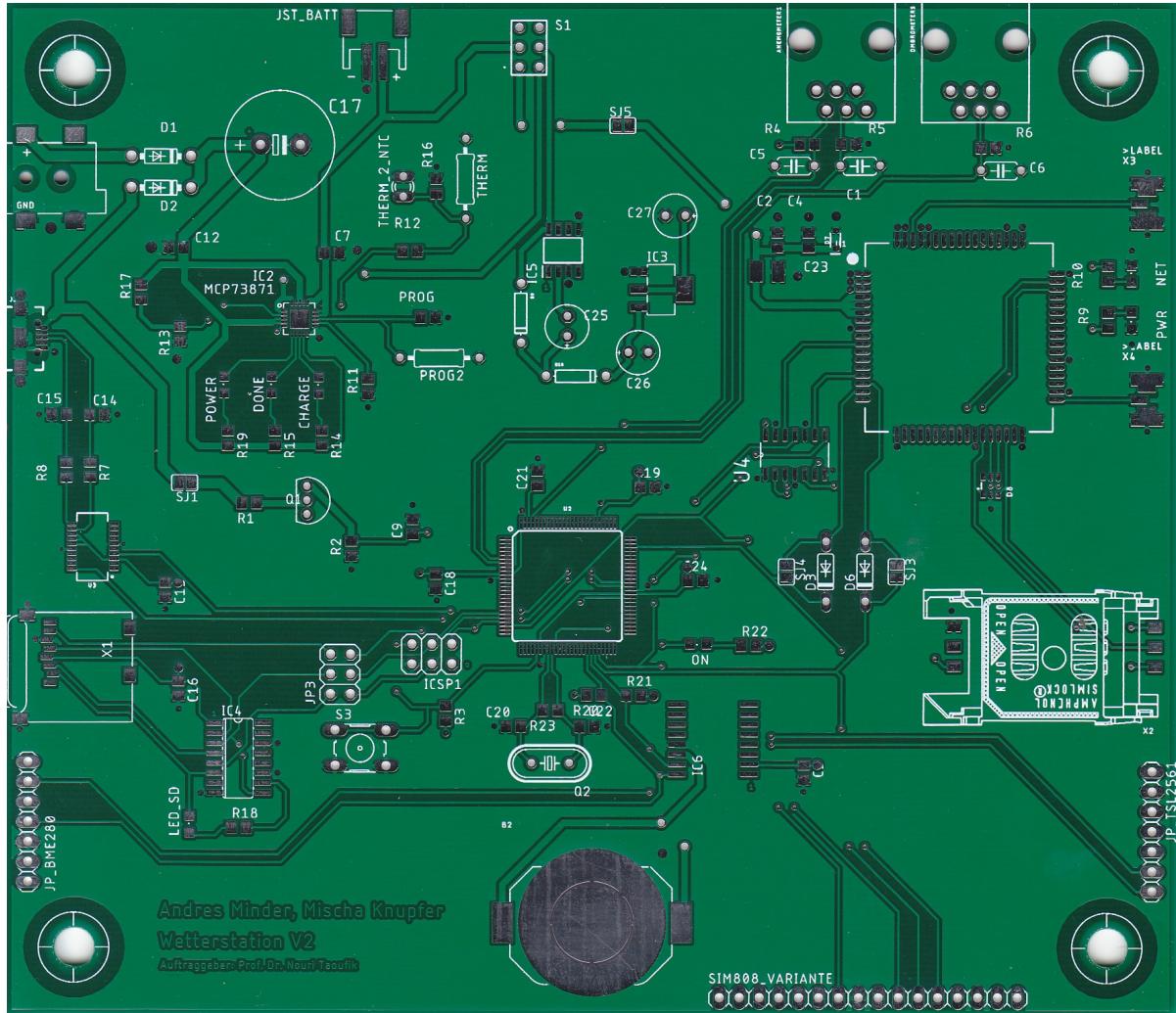
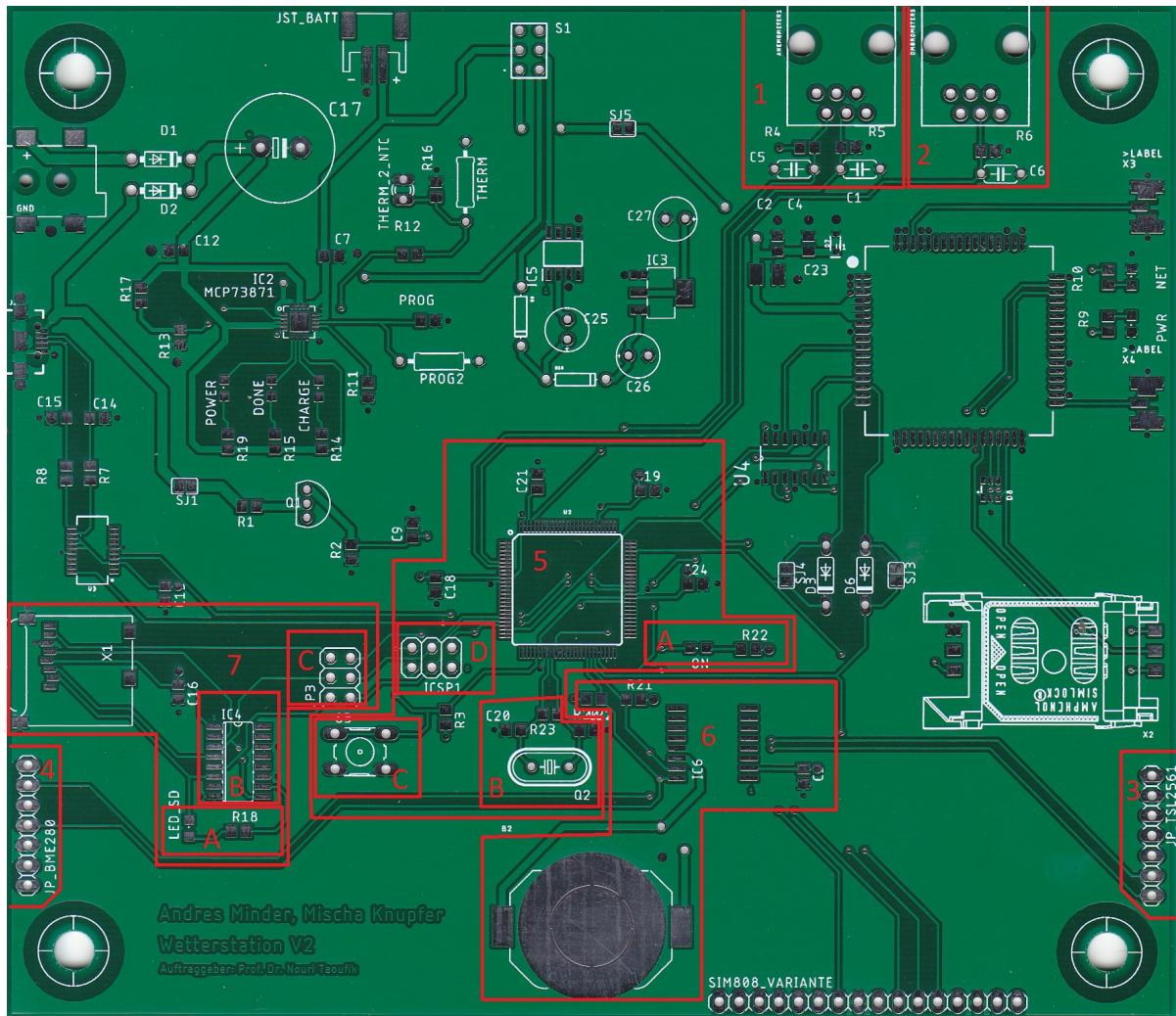


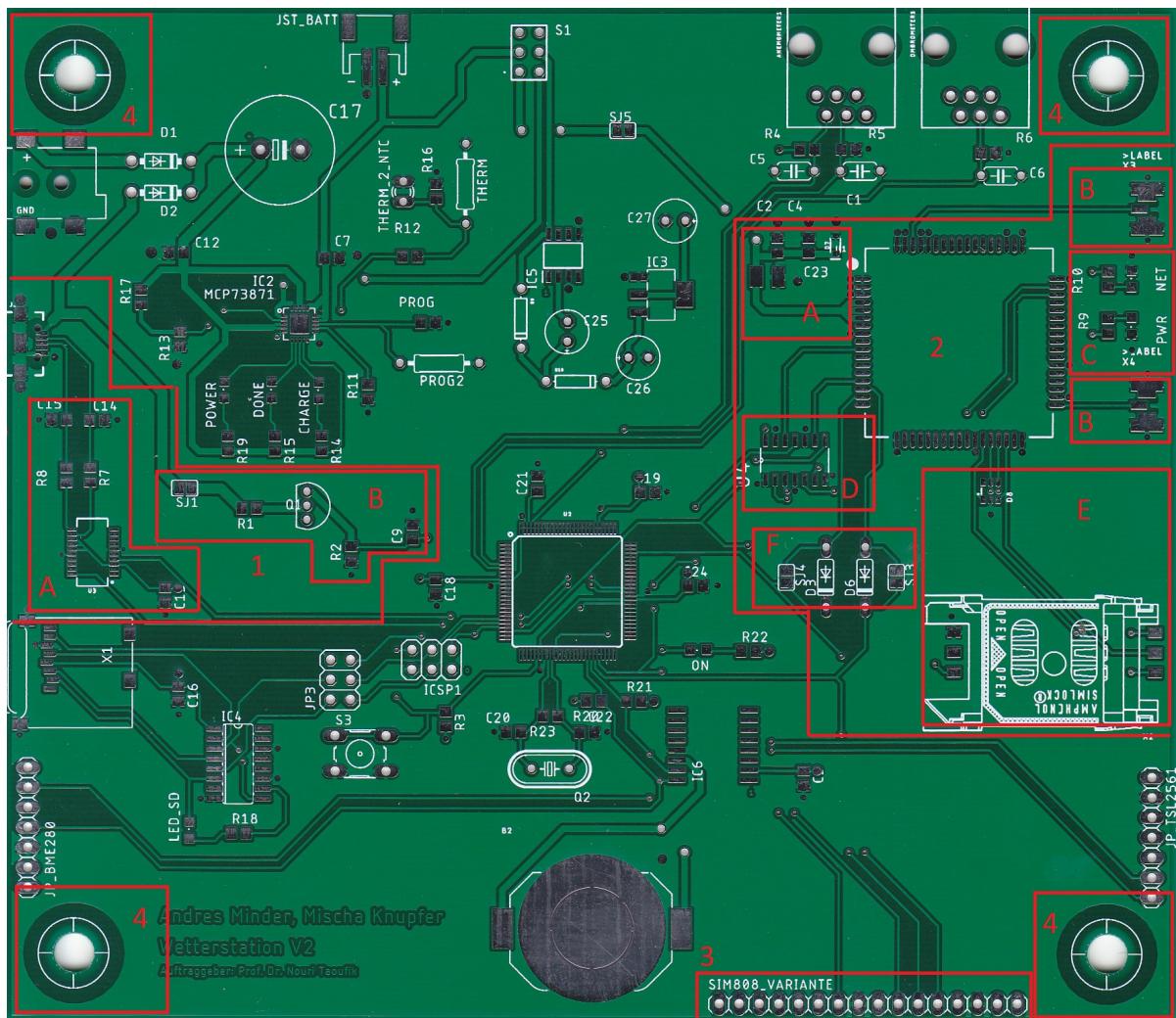
Abbildung 3.1: Das noch unbestückte PCB.

Abbildung 3.1 zeigt das noch unbestückte PCB ohne markierte Bereiche. Man erkennt deutlich die Leiterbahnen sowie die Lötpads für die verschiedenen Bauteile. Welche Bauteile nun zur gleichen Bauteilgruppe gehören wird mit den nächsten Abbildungen erkenntlich.



**Abbildung 3.2:** Das unbestückte PCB mit den markierten Bereichen der MCU, der Datenspeicherung, der RTC und der Sensoren.

Sieben Bauteilgruppen wurden in der Abbildung 3.2 markiert. Markierung 1 zeigt den RJ11-Stecker für das Anemometer mit Windrichtungsgeber, wobei an den Ausgängen jeweils ein RC-Filter angefügt wurde um Störungen im höheren Frequenzbereich herauszufiltern, um so ein nutzbares Signal zu erhalten. Das Ombrometer trägt die Markierung 2 und besitzt ebenfalls ein RC-Filter am Ausgang. Die 3 markiert die Pinleiste für den Anschluss des TSL2561 (Lichtintensitätssensor), ein nach aussen geführtes Breakout Board. Gleich Verhält es sich mit der 4, dem BME280. Die 5 markiert die Bauteilgruppe der MCU (ATMega2560), welche 4 wichtige Untergruppen beherbergt. 5-A zeigt eine Status-LED, welche leuchtet, sobald die MCU in betrieb ist. 5-B ist ein externer Clock, welcher der MCU den Takt vorgibt. Ein Reset der MCU und somit des Systems ist über den Reset-Button (5-C) möglich. Über 5-D, ein ICSP-Header (In Circuit System Programming), kann die MCU mit der Firmware geladen werden. Die 6 markiert die Bauteilgruppe des RTC (Real Time Clock) mit der zugehörigen Stützbatterie (CR2032 Knopfbatterie). Die 7-te Bauteilgruppe ist die Datenspeicherung. 7-A zeigt ein Status-LED, welche bei Aktionen mit der  $\mu$ SD-Karte aufleuchtet. 7-B beherbergt ein IC mit integrierten Operationsverstärkern, welche die Leitungen der  $\mu$ SD-Karte treiben. Über 5-C kann die Datenspeicherung über Jumper abgekoppelt werden, dies ist notwendig, da es während des Ladens der Firmware auf die MCU es sonst zu unkontrollierte Zugriffe auf die  $\mu$ SD-Karte kommen kann.



**Abbildung 3.3:** Das unbestückte PCB mit den markierten Bereichen des  $\mu$ USB-Interfaces, des SIM808 und der Bohrlöcher für die Montage im Gehäuse.

Abbildung 3.3 zeigt die markierten Bereiche für die SIM808 (2), des  $\mu$ USB-Interfaces (1) und die Bohrlöcher für die Montage des PCBs im Gehäuse (4). Ausserdem gibt es eine Bestückungsvariante (3), bei der ein Breakout Board des SIM808 angeschlossen werden könnte, falls die On-Board Variante nicht funktionieren sollte. Im Bereich des  $\mu$ USB-Interfaces (1) sieht man deutlich zwei Teilbereiche, A und B. Teilbereich 1-A enthält den FT231XS, ein USB-to-UART IC, und dessen Beschaltung gemäss Datenblatt [4]. Für die Anschlusserkennung wird die Schaltung im Teilbereich 1-B verwendet. Für die SIM808 wird eine SIM-Karte (2-E) verwendet, sowie Antennen (2-C) und eine Schaltung zur Stabilisierung und Entstörung der Speisung (2-A) benötigt. 2-C beinhaltet zwei Status-LED, um den Betrieb und die Konnektivität anzuzeigen. Für die Anpassung des Spannungspegels vom 4V gespiesenen SIM808 auf die mit 3.3V gespiesene MCU dient der 74VHC125, zu sehen im Teilbereich 2-D. Zwei Dioden schützen den RST- und den RX-Pin des SIM808 vor ungewollten Spannungsspitzen und deren Folgen im Teilbereich 2-F.

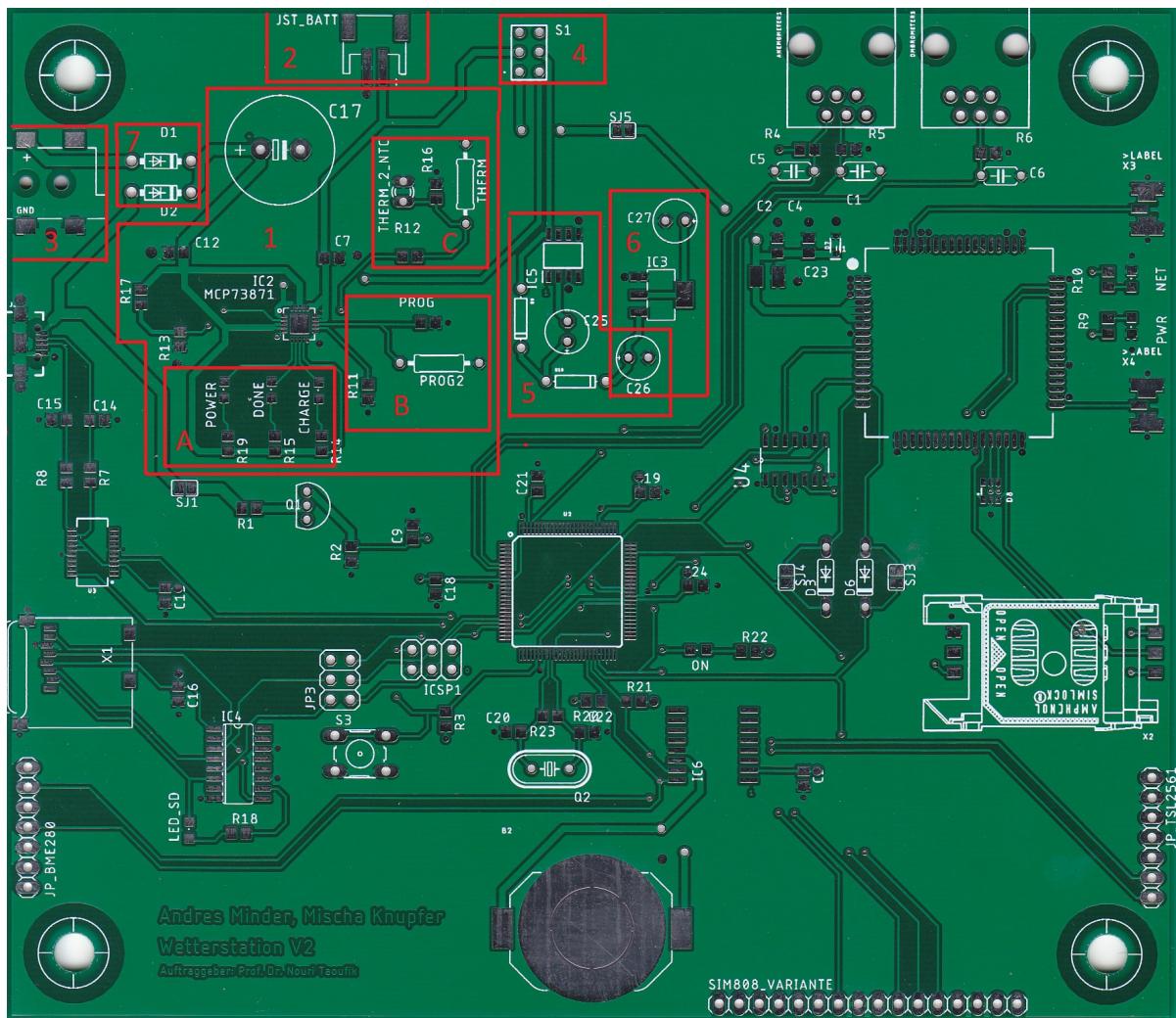


Abbildung 3.4: Das unbestückte PCB mit den markierten Bereichen für die Energieversorgung.

Die gesamte Schaltung für die Energieversorgung wurde in Abbildung 3.4 in Bereiche unterteilt. Bereich 1 zeigt den MCP73871, den Power-Management-Chip, mit seinen Status-LED (1-A), sowie dessen Beschaltungen zur Einstellung des maximalen Ladestroms (1-B) und der Ausschalttemperatur (1-C). Der Anschluss für die Batterie erfolgt über den JST-Stecker (2). Über ein DC-Power Jack (3) wird die Photovoltaikanlage an die Wetterstation angeschlossen. Durch einen Schalter (4) können alle anderen Systeme der Wetterstation von der Energieversorgung getrennt werden. Die Charge-Pump (5) erhöht die erhaltene Batteriespannung, damit der Linearregler (6) die 3.3V-Ausgangsspannung mit möglichst kleinem Rippel erzeugen kann. Zwei Dioden (7) schützen die Eingänge vor rückfliessenden Strömen, indem diese nur in die gewünschte Richtung durchlassen und in die andere Richtung sperren.

In diesem Kapitel wurden die Bereiche des noch unbestückten PCBs erläutert und dadurch eine Übersicht über die Komponenten gegeben. In den nächsten Kapiteln soll auf diese Komponenten eingegangen werden, wobei mit der MCU begonnen wird.

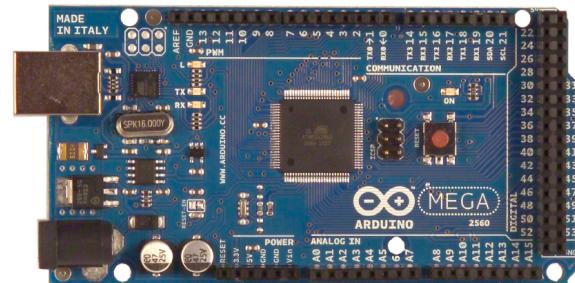
## 3.2 MCU

Im Projekt 5 wurde bereits eine MCU ausgewählt, mit der das Prototyping erfolgte. Aufgrund der Verfügbarkeit (bei der FHNW bezugsbereit) wurde ein Arduino Mega2560 Board verwendet. Auf den kommenden Abschnitt aus dem Fachbericht des Projekt 5 folgt ein Abschnitt mit ergänzenden Informationen aus der Bachelor-Thesis.

### 3.2.1 MCU - Projekt 5

Die Micro Controller Unit (MCU) ist der zentrale Bestandteil für die Kommunikation, resp. für den Datenaustausch zwischen den unterschiedlichen Modulen. Sie interpretiert die Signale der Sensoren und rechnet sie in die interessierenden Messwerte um. Dann weist die MCU jedem Messwert einen Zeitstempel über das RTC zu und übergibt diesen der Datenspeicherung. Wenn die Daten vom Kommunikationsmodul angefordert werden, liest die MCU die Datenspeicherung aus und übergibt sie dem Kommunikationsmodul.

Für die Entwicklung der MCU wird ein Arduino Mega Board verwendet. Der Vorteil besteht darin, dass elementare Bauteile (Hardware) bereits implementiert sind, wie z.B. Oszillator, der USB-Anschluss und die PCB-Connectors für ein schnelles Prototyping. Die wichtigsten technischen Daten sind in der Tabelle 3.1 aufgelistet.



**Abbildung 3.5:** Arduino Mega Board [5, S.1]

**Tabelle 3.1:** Technische Daten [5, S.3]

Microcontroller	ATmega2560
Operating Voltage	5V
Digital I/O Pins	54
Analog Input Pins	16
Flash Memory	256 KB, 8 KB werden vom bootloader benötigt
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

### 3.2.2 Ergänzungen aus der Bachelor-Thesis

Während der Bachelor-Thesis kam die Idee auf, eine kleinere MCU zu verwenden. Eine kleinere MCU hat die Vorteile, dass diese günstiger ist in der Anschaffung, einen geringeren Stromverbrauch aufweist und weniger Platz auf dem PCB benötigt. Aufgrund der Firmware wird jedoch eine erhöhte Menge Speicher alloziert, weshalb die *Data Memory Usage* des verwendeten ATMega328 zu klein war und wieder auf den bereits für das Prototyping verwendeten ATMega2560 zurückgegriffen werden musste. Für die Wetterstation wird schlussendlich kein Board mehr verwendet, sondern der Mikroprozessor (ATMega2560, siehe Abbildung 3.6) selbst auf dem PCB platziert. Damit dieser wie gewünscht funktioniert, werden weitere Bauteile verwendet. Im nächsten Abschnitt wird mehr auf die zusätzlichen Bauteile eingegangen.

Wie in der Übersicht in Abbildung 3.2 zu sehen ist, wird für die MCU ein ICSP-Header (In Circuit System Programming) benötigt. Durch diesen ICSP-Header kann die Firmware über ein AVR-Dragon auf die MCU geladen werden, wobei spätere Anpassungen der Firmware möglich bleiben, was wiederum zur Skalierbarkeit (Erweiterbarkeit) des Systems beiträgt. Die MCU selbst (ATMega2560) hat einen internen Clock, welcher über ein RC-Oszillator generiert wird und eine wesentlich höhere Temperaturabhängigkeit besitzt als ein externer Quarz. Aus diesem Grund wird ein externer Quarz verwendet, wie in Abbildung 3.2 ebenfalls ersichtlich ist. Außerdem werden Kondensatoren bei den 3.3V-Speisepins verwendet, um Induktivitäten durch Leiterbahnängen auszugleichen. Um das System rebooten zu können wird ein Reset-Button hinzugefügt, wobei über eine Status-LED der Betrieb der MCU ersichtlich ist. Über ein CLI (Command Line Interface) soll mit der MCU interagiert werden können, wodurch eine Verbindung mit einem  $\mu$ USB-Interface erforderlich wird. Die Sensorik, die RTC (Real Time Clock) und die SIM808 werden ebenfalls mit der MCU verbunden, was die MCU zum Herzstück der Wetterstation macht, da hier alle Daten verarbeitet werden. Die verarbeiteten Daten werden in der angeschlossenen Datenspeicherung wiederum gespeichert.

In diesem Kapitel wurde die Thematik der MCU eingehend thematisiert. RC-Oszillatoren und Quarze wurden ebenfalls erwähnt. Aus diesem Grund wird im nächsten Kapitel die RTC näher erläutert.



**Abbildung 3.6:** Der Mikroprozessor ATMega2560, ohne Board.

### 3.3 RTC

Im vergangenen Kapitel (MCU) wurden bereits RC-Oszillatoren und Quarze erwähnt. Die RTC (Real Time Clock) beinhaltet ebenfalls einen Quarz, weshalb die RTC in diesem Kapitel näher erläutert wird. Die RTC wurde bereits während des Projekt 5 integriert und in dessen Fachbericht dokumentiert. Nachfolgender Text wurde übernommen, damit die Vollständigkeit gewahrt wird.

#### 3.3.1 RTC - Projekt 5

Um die gewonnenen Messdaten mit einem Zeitstempel zu versehen, wird eine **RTC** (*real time clock*) benötigt. Um eine möglichst kleine Abweichung zu haben, wird eine hohe Präzision vorausgesetzt. Außerdem soll die **RTC** über das in Kapitel ?? erwähnte **I<sup>2</sup>C**-Bus angesteuert werden. Aus diesen Gründen wurde der **DS3231** implementiert, da dieser als Präzisions-I<sup>2</sup>C-RTC die Anforderungen erfüllt. Die hohe Präzision des **DS3231** wird mit einem internen Temperatursensor erreicht, welcher Temperaturbedingte Abweichungen des Oszillators kompensiert. Zu sehen ist der **DS3231** mit seinen Anschlüssen in Abbildung 3.7.

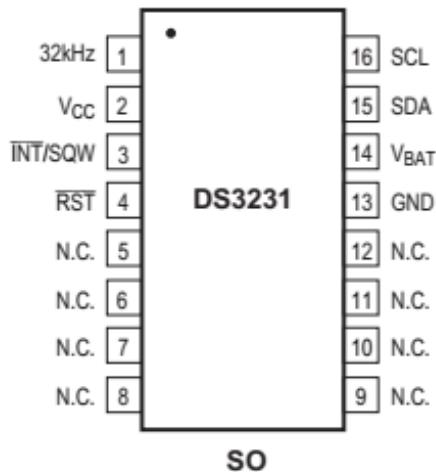


Abbildung 3.7: DS3231 mit seinen Anschlüssen [6].

Die Anschlüsse des **DS3231** sind **VCC**, **GND**, **SCL**, **SDA**, **BAT**, **32K**, **SQW** und **RST**. **VCC** und **GND** werden für die Speisung benötigt. **SCL** und **SDA** sind Anschlüsse für den I<sup>2</sup>C-Bus. **BAT** ist der positive Anschluss der Knopfbatterie, worüber deren Zustand kontrolliert werden, etwas anderes gespiesen oder eine andere Battery als Backup angeschlossen werden kann. **32K** ist ein Anschlusspin um den Output des 32kHz-Oszillators des **RTC** abzugreifen, was jedoch nicht verwendet wird. **SQW** ist ein zusätzlicher Output- oder Interrupt-Pin, welcher jedoch auch nicht verwendet wird. **RST** wird verwendet, um ein externes Element zu reseten oder als Indikator wenn die Hauptspeisung unterbrochen wird. [6]

Die relevanten Spezifikationen des Chips sind in Tabelle 3.2 aufgelistet.

Parameter	Min.	Typ.	Max.	Einheit	Condition
VCC	2.3	3.3	5.5	V	
VBAT	2.3	3	5.5	V	
Active Supply Current			200	$\mu$ A	3.63 V
			300	$\mu$ A	5.5 V
Standby Supply Current			110	$\mu$ A	3.63 V
			170	$\mu$ A	5.5 V
Crystal Aging	$\pm$ 1			ppm	First Year
	$\pm$ 5			ppm	0-10 Years
Active Battery Current			70	$\mu$ A	3.63 V
			150	$\mu$ A	5.5 V
Timekeeping Battery Current	0.84	3		$\mu$ A	3.63 V
	1	3.5		$\mu$ A	5.5 V

**Tabelle 3.2:** Spezifikationen des DS3231 [6].

Tabelle 3.2 zeigt die für das Projekt relevanten Spezifikationen des **DS3231**. Wichtig ist, dass die Alterung des Quarzes zu einem Fehler führt, dieser jedoch im ppm-Bereich liegt und somit erst über viele Jahre hinweg bemerkbar wird.

Wie erwähnt wurde, wird die RTC über den I<sup>2</sup>C-Bus von der MCU angesteuert. Da dies ebenfalls für gewisse Sensoren gilt, werden im nächsten Kapitel die Sensoren thematisiert.

### 3.4 Sensoren

Der grossteil der Sensorik wurde bereits während des Projekt 5 implementiert und in dessen Fachbericht dokumentiert. Die Dokumentation der bereits implementierten Sensoren wird in nachfolgendem Abschnitt inkludiert und darauf folgend die während der Bachelor-Thesis hinzugekommene Sensorik bzw. Ergänzungen in einem separaten Abschnitt erläutert.

#### 3.4.1 Ombrometer - Projekt 5

##### Ermittlung der Niederschlagsmenge

Dieses Unterkapitel befasst sich mit der Realisierung der Niederschlagsmessung. Diese soll nach einem Kipplöffelprinzip funktionieren und gemäss definierten Zielen eine Genauigkeit von  $\pm 100 \text{ ml/m}^2$  aufweisen. Ausserdem soll als alternative zusätzlich ein Messbecher an der Wetterstation installiert werden, damit der Bauer die Niederschlagsmenge anhand einer Skala ablesen kann. In einem ersten Schritt soll das Kipplöffelprinzip näher erläutert und mit einem Selbstbau die Funktionsweise getestet werden. Anschliessend soll ein gekaufter Sensor die Wetterstation erweitern und die Implementation in der Firmware thematisiert werden. Zu guter Letzt soll die Validierung des Teilsystems folgen.

##### Das Kipplöffelprinzip

Das Prinzip des Kipplöffels wird in Abbildung 3.8 graphisch dargestellt.

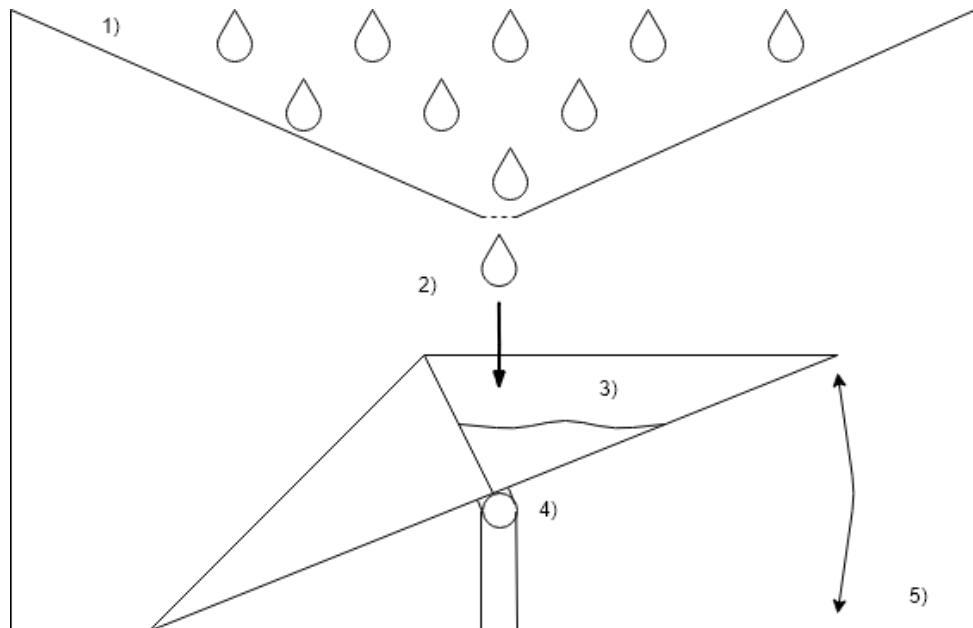


Abbildung 3.8: Darstellung des Kipplöffelprinzips

Abbildung 3.8 zeigt das Kipplöffelprinzip. Der Kipplöffel („3“) besteht im Grunde aus zwei Löffeln und ist in der Mitte drehbar mit dem Gehäuse befestigt („4“). Regenwasser wird über eine Öffnung im Gehäusedeckel (Trichter, „1“) zum Kipplöffel befördert („2“). Ist der Löffel mit Regenwasser gefüllt, so kippt dieser aufgrund des Gewichts und leert das Wasser über eine Öffnung im Gehäuseboden („5“) aus. Durch die Kippung wird der andere Löffel in die Ausgangsposition bewegt und kann nun mit Wasser füllen. Mit der Hilfe von Reedkontakte und Magneten wird die Anzahl der Kippbewegungen gezählt. Die Niederschlagsmenge ergibt sich aus der Anzahl Kippbewegungen, multipliziert mit dem Volumen des Kipplöffels.

### Die Realisierung des Niederschlagsmengensensors

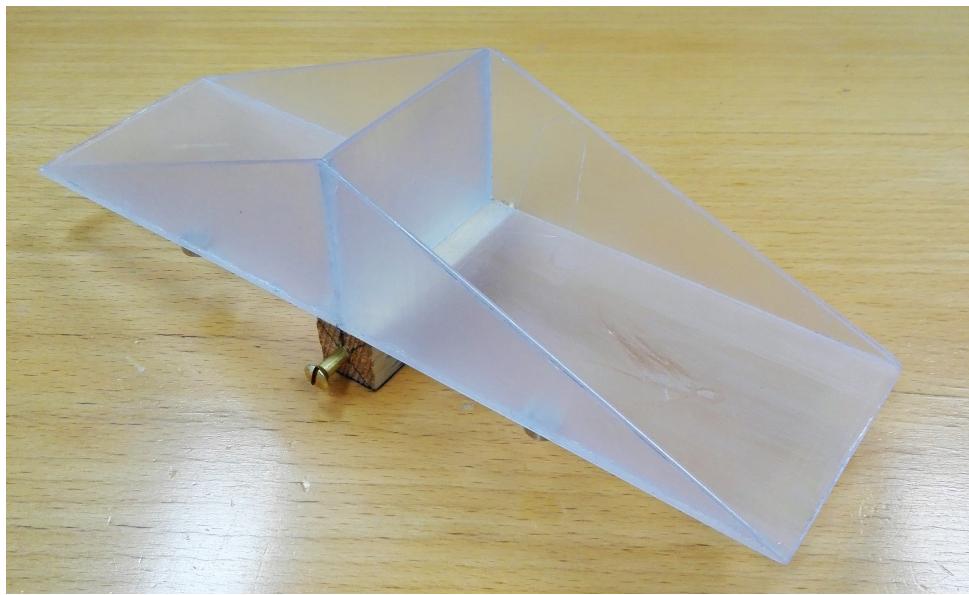
Um die Funktionsweise des Niederschlagsmengensensors zu testen, wird, wie im Pflichtenheft festgehalten, dieser in einem ersten Schritt selbst erstellt. Die Erstellung kann in vier Etappen unterteilt werden. Die erste Etappe ist die Erstellung des Kipplöffels. Die zweite Etappe folgt mit der Erstellung der drehbaren Lagerung. Als dritte Etappe folgt der Trichter und die vierte und letzte Etappe widmet sich dem Gehäuse, wobei der Trichter ein Teil des Gehäuses darstellt. Das Gehäuse wird, bei Verwendung der Eigenproduktion, erst im Projekt 6 mit dem Gehäuse der gesamten Wetterstation erstellt.

#### Etappe 1: Realisierung des Kipplöffels

Wichtig für die Erstellung des Kipplöffels sind die Dimensionierung und die Materialwahl.

Das Material soll wetterbeständig, einfach bearbeitbar und günstig sein und eine möglichst glatte Oberfläche haben. Die möglichst glatte Oberfläche ist notwendig, damit das Wasser im Kipplöffel sich nicht an der Oberfläche festhält und somit gut abfließt. Acrylglas erfüllt diese Bedingungen und ist in jedem Baumarkt erhältlich, weshalb es als Material gewählt wird.

Die Dimensionierung ist Abhängig von der gewählten Genauigkeit im Pflichtenheft. Damit eine Genauigkeit von  $\pm 100 \text{ ml}/\text{m}^2$  erreicht werden kann, müssen beide Löffel des Kipplöffels bei genau 100 ml Fassungsvermögen kippen. Damit dies erreicht wird, kann man physikalisch die statische Gleichgewichtsbedingung aufstellen und daraus die Dimensionierung folgern. Dies ist jedoch ein sehr aufwändiger, komplizierter und zeitintensiver Weg. Einfacher ist es, wenn der Kipplöffel extra zu gross dimensioniert und die Füllmenge im Nachhinein justiert wird. Die Justierung erfolgt mittels einer in der Höhe verstellbaren Lagerung, sowie mit in der Höhe verstellbaren Schrauben im Gehäuseboden, welche die Neigung der Endposition des Kipplöffels beeinflusst. Ein weiterer Vorteil dieser Nachjustierung ist, dass auch eine andere Füllmenge einstellbar wäre.



**Abbildung 3.9:** Selbsterstellter Kipplöffel.

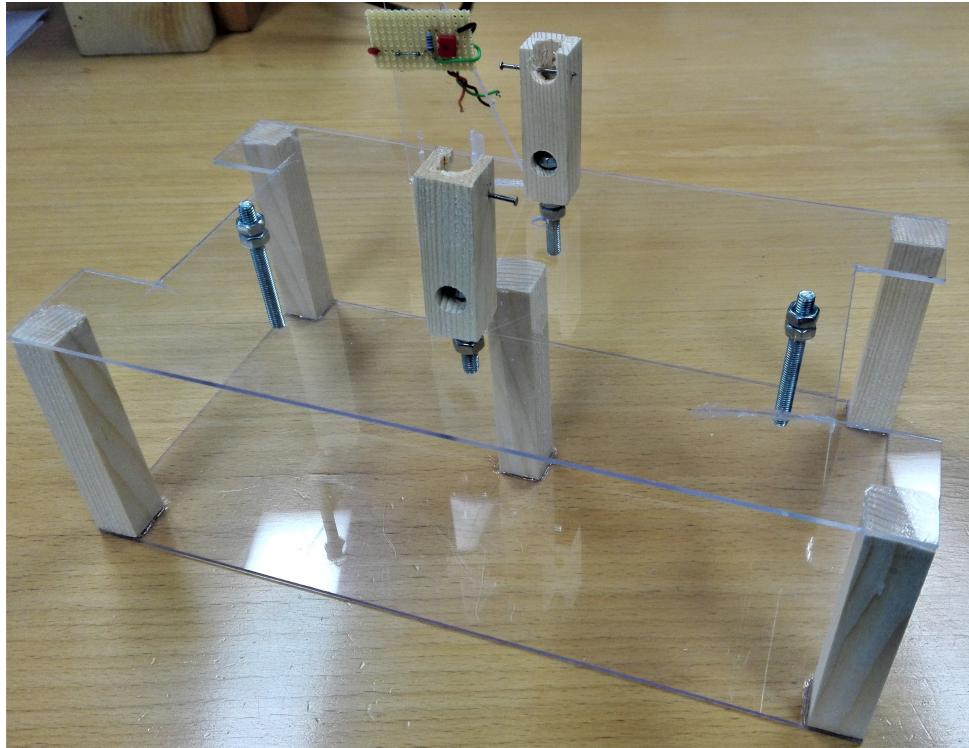
Abbildung 3.9 zeigt den selbsterstellten Kipplöffel aus Acrylglas. Die Drehachse ist mittig unter dem Kipplöffel befestigt und besteht aus einem Holzklotz mit je einer Schraube pro Seite.

#### Etappe 2: Realisierung der drehbaren Lagerung

Die drehbare Lagerung des Kipplöffels ist wichtig, damit der Kipplöffel auf beide Seiten kippen kann. Die Drehachse soll direkt unterhalb der Mitte des Kipplöffels befestigt sein um ein gleich-

mässiges Kippen zu ermöglichen. Die Höhe des Kipplöffels wird definiert durch die einstellbare Höhe der Drehachsenlagerung.

Die Drehachse wird aus einem Stück Holz und zwei Schrauben gefertigt, wobei das Holz direkt am Kipplöffel befestigt wird. Die zwei Schrauben werden auf einem höhenverstellbarem Gerüst gelagert, so dass ein drehen möglich ist. Dieses Gerüst wird auch aus Holz gefertigt und enthält eine Metallische Fläche an der Kontaktstelle der zuvor erwähnten Schrauben, um aufkommende Reibkräfte zu verringern. Ausserdem ist dieses Gerüst höhenverstellbar über zwei mit Muttern feststellbaren Gewinden (für jede Seite eine).



**Abbildung 3.10:** Selbsterstelltes Gerüst.

Abbildung 3.10 zeigt das selbsterstellte Gerüst für die Höhenverstellbare, drehbare Lagerung des Kipplöffels. Es kann sowohl die Höhe des Kipplöffels, sowie dessen Neigung in den Endpositionen über Gewinden mit Muttern eingestellt werden. Die Schrauben des Kipplöffels kommen auf einen Stahlnagel zu liegen, womit Reibungsverluste gering gehalten werden.

#### **Etappe 3: Realisierung des Trichters**

Der Trichter sorgt dafür, dass der Regen, welcher auf die Trichterfläche fällt, über der Mitte des Kipplöffels in den Löffel fliesst. Die Trichterfläche stellt gleichzeitig die Referenzfläche dar, da die gesamte Regenmenge dieser Fläche über den Kipplöffel erfasst wird. Ist diese Fläche von  $1 m^2$  abweichend, so muss in der Firmware ein Skalierungsfaktor implementiert werden, damit die Regenmenge wie gewünscht gemäss Pflichtenheft ermittelt werden kann. Der Trichter wird aus demselben Material gefertigt wie der Kipplöffel, da hier die gleichen Anforderungen gelten. Es sei angemerkt, dass der Trichter nur bei weiteren Verwendung des selbst erstellten Kipplöffels, zusammen mit dem Gehäuse der gesamten Wetterstation, erstellt wird.

#### **Etappe 4: Realisierung des Gehäuses**

Das Gehäuse soll den Sensor vor ungewollten äusseren Einflüssen schützen, sowie umgebende Elektronik vor eventuellen Regenwasserspritzern. Ausserdem soll ein Schaltkreis mit Reedrelais implementiert werden, damit die Kippbewegungen von der Elektronik erfasst werden können.

Es sei erwähnt, dass das Gehäuse nur bei weiteren Verwendung des selbst erstellten Sensors, zusammen mit dem Gehäuse der Wetterstation, konstruiert wird.

### Implementierung des Schaltkreises

Der Schaltkreis, welcher die Kippbewegungen feststellen soll, besteht im wesentlichen aus einem Reedrelais und einem Permanentmagneten. Das Reedrelais ist NO (Normally Open) und wirkt als stromkreisschliessender Schalter, sobald ein magnetisches Feld (z.B. das eines Permanentmagneten) sich in unmittelbarer Nähe befindet. Der Permanentmagnet wird auf dem Kipplöffel befestigt und das Reedrelais als Gegenstück an einem Fixpunkt in der Nähe. Wichtig dabei ist, dass das Reedrelais bei den Endpositionen des Kipplöffels nicht geschlossen ist, damit der Stromkreis geöffnet ist und Strom gespart werden kann. Das Reedrelais benötigt einen seriellen Widerstand, damit bei einem schliessen des Stromkreises kein Kurzschluss auftritt. Außerdem soll ein Kondensator parallel zum Widerstand sein, um die Speisespannung zu glätten und so ein nutzbares Signal zu erhalten. Die Speisespannung stellt den Pegel für ein schliessen des Reedrelais, und somit auch für eine Kippbewegung dar. Um die Kippbewegungen zu zählen, kann somit entweder jede steigende oder jede fallende Flanke des Signals gezählt werden.

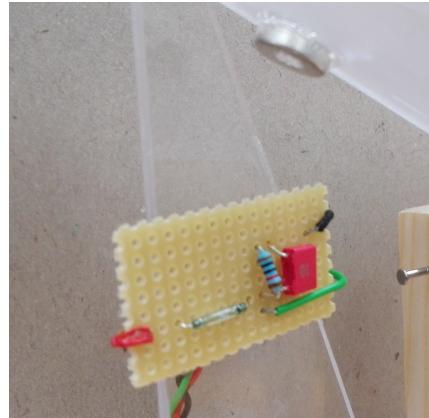


Abbildung 3.11: Schaltkreis zur detektion der Kippbewegung.

Abbildung 3.11 zeigt den Schaltkreis zur detektion der Kippbewegung. Benutzt wurde ein  $10\text{k}\Omega$  Widerstand mit einem parallel angeschlossenen  $100\text{nF}$  Kondensator. Der Reedkontakt reagiert auf den ebenso sichtbaren Permanentmagneten, welcher an der Unterseite des Kipplöffels befestigt ist.

### Nachteile des Selbstgebauten Niederschlagsmengensensor

Der selbstgebaute Niederschlagssensor beweist, dass das Prinzip des Kipplöffels funktioniert. Dennoch weist der Selbstbau Mängel auf. Der verwendete Permanentmagnet muss geklebt werden, weshalb dessen Magnetfeld massiv an Stärke verliert und die Schaltung deshalb äußerst nahe angebracht werden muss. Außerdem kam es, dadurch dass keine Werkstatt zugänglich war, zu Improvisation bei nahezu allen Fertigungsschritten, was zu unkalkulierbaren Abweichungen führt. Als Beispiel sei das Spiel der drehbaren Lagerung des Kipplöffels auf dem Gerüst angeführt, was jegliche Justierungsversuche der Niederschlagsmenge beeinflusst. Aus den genannten Gründen wird vorgefertigter Sensor mit Kipplöffelprinzip verwendet.



**Abbildung 3.12:** Das Ombrometer

Das verwendete Ombrometer von MISOL hat eine Auflösung von ca. 2ml pro Kippbewegung auf die Dimensionen  $5.5E - 2m * 11.5E - 2m$  des Trichters. Zudem enthält der Trichter oben noch eine kleine Wasserwaage, damit das Ombrometer auch gerade steht und die gemessene Regenmenge nicht verfälscht wird.

### 3.4.2 Anemometer - Projekt 5

Für die Windgeschwindigkeitsmessung wurde ein Ersatz Anemometer von Froggit genommen (Abb. 3.13). Das Anschlusskabel hat einen vier poligen RJ-11 Stecker, dessen Signal über eine Buchse zum MCU geführt wird. Das Anemometer selbst hat allerdings nur zwei Anschlüsse, die Speisung (rot) und das durch einen mit einem Dauermagneten schließbaren Reedkontakt modulierte pulsförmige Ausgangssignal (grün, Abb. 3.14). In der Abb. 3.15 ist ersichtlich, dass das Ausgangssignal über R1 abfällt und C1 als Spannungsstabilisierung dient. Das daraus resultierende Signal ist in der Abb. 3.16 aufgezeigt. Die Windgeschwindigkeit ist nun aus der Anzahl Rechteckpulsen direkt interpretierbar:

Wenn über einen Zeitraum  $T$  die Anzahl Pulse  $A$  gemessen werden, dann kann auf die Winkelgeschwindigkeit  $\omega$  nach

$$\omega = \frac{A}{T} \quad [s^{-1}] \quad (3.1)$$

geschlossen werden. Da allerdings verschiedene Faktoren wie das Trägheitsmoment des Schalenkreuzes, Reibungsverluste bei der Drehbewegung, Verfälschung bei wechselnder Windrichtung usw. zusätzlich auf das Anemometer wirken, wird es sehr komplex die Windgeschwindigkeit exakt zu berechnen. Deshalb wird nur ein Näherungswert ermittelt und mit einem Skalierungsfaktor  $SF$  korrigiert. Somit ergibt sich für die Windgeschwindigkeit  $v_{Wind}$  mit Radius  $r$  des Schalenkreuzes

$$v_{Wind} = \frac{A \cdot r \cdot SF}{T} \quad [m/s]. \quad (3.2)$$

Der Skalierungsfaktor  $SF$  wird mittels Referenzmessungen der Windgeschwindigkeit eines digitalen Anemometers eruiert.

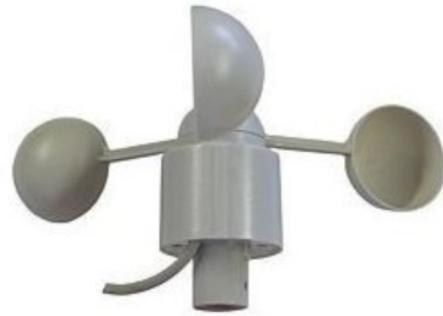


Abbildung 3.13: Anemometer [7]

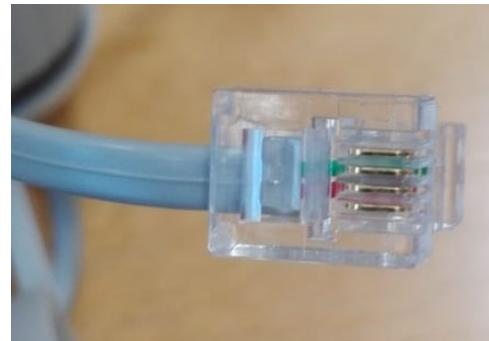


Abbildung 3.14: RJ-11 Stecker

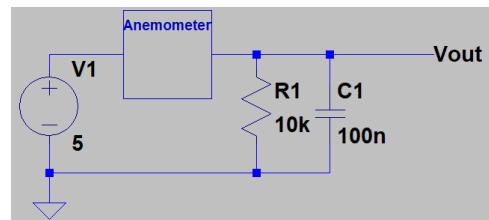


Abbildung 3.15: Beschaltung des Ausgangs des Anemometers.

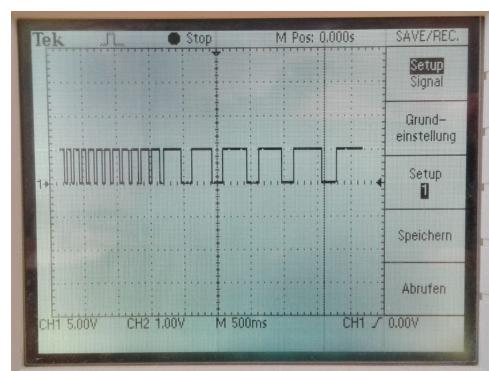


Abbildung 3.16: Ausgangssignal  $V_{out}$

### 3.4.3 Windrichtungsgeber - Projekt 5

Um die Windrichtung angeben zu können, wurde ein Windrichtungsgeber, wie in Abb. 3.19 gezeigt von MISOL verwendet. Er ist genau wie das Ombrometer und das Anemometer mit Reedkontakte realisiert worden (siehe Abb. 3.17). Dafür sind acht Reedkontakte im Kreis angeordnet und jeder hat einen in Serie geschalteten Widerstand von unterschiedlichen Dimensionen. Der Dauermagnet kann, je nach Drehwinkel bis zu zwei Reedkontakte gleichzeitig schließen. Dies erlaubt sechzehn verschiedene Winkelpositionen und somit eine Auflösung von  $22.5^\circ$ . Mit einem externen Widerstand  $R = 10k\Omega$  (siehe Abb. 3.18) wird eine Spannung generiert, welche dann mit dem ADC des Microcontrollers gelesen werden kann. Der Windrichtungsgeber wird mit einer Speisespannung von  $V_+ = 5V$  betrieben. Der Windrichtungsgeber hat einen vierpoligen RJ-11 Anschluss. Zudem hat er auf der unteren Seite noch eine RJ-11 Buchse, bei der das Anemometer direkt angeschlossen werden kann. Wie diese Anschlüsse gemapped sind, wird in der Abb. 3.17 gezeigt.

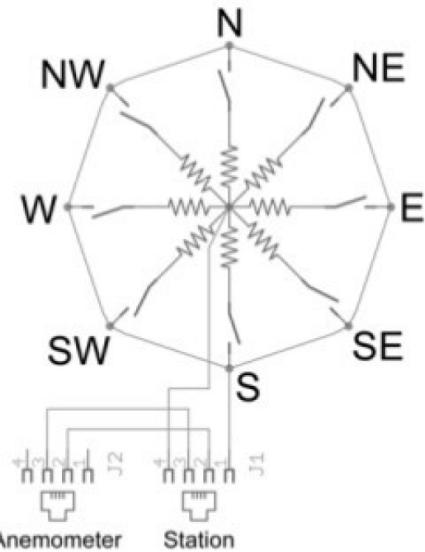
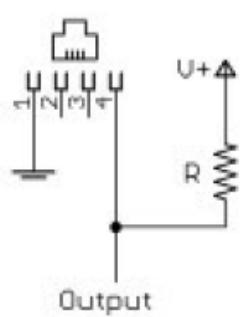


Abbildung 3.17: Interne Schaltung [8]

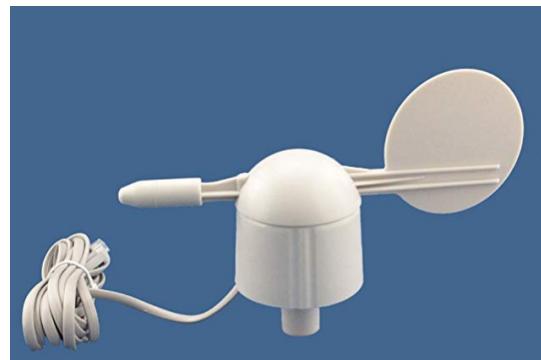
**Tabelle 3.3:** Technische Werte [8]

Richtung [ $^\circ$ ]	Himmelsrichtung	Widerstand [ $\Omega$ ]	Ausgangsspannung [V]
0	N	33k	3.84
22.5		6.57k	1.98
45	NE	8.2k	2.25
67.5		891	0.41
90	E	1k	0.45
112.5		688	0.32
135	SE	2.2k	0.90
157.5		1.41k	0.62
180	S	3.9k	1.40
202.5		3.14k	1.19
225	SW	16k	3.08
247.5		14.12k	2.93
270	W	120k	4.62
292.5		42.12k	4.04
315	NW	64.9k	4.33
337.5		21.88k	3.43

In der Tabelle 3.3 sind die Widerstandswerte der in Abb. 3.17 gezeigten Widerständen und die Werte der Ausgangsspannung bei variabler Winkelposition aufgelistet. Da nun abhängig von der Winkelposition unterschiedliche Widerstände parallel geschalten werden, resultiert am Ausgang eine vom Winkel abhängige Ausgangsspannung.



**Abbildung 3.18:** Spannungsteiler mit  $R = 10k\Omega$  [8]



**Abbildung 3.19:** Windrichtungsgeber von MISOL [9]

#### 3.4.4 BME280 - Projekt 5

Der *BME280* ist ein low powered digitaler Feuchtigkeits-, Luftdruck- und Temperatursensor von Bosch. Er ist in einem 2.5mm x 2.5mm x 0.93mm metal lid LGA Gehäuse verpackt und kann über die Interfaces I<sup>2</sup>C und SPI kommunizieren. Durch seinen niedrigen Stromverbrauch, große operating range der drei Messgrößen und schnellen Ansprechzeit von etwa 1s eignet er sich für die solarbetriebene mobile Wetterstation besonders. [10]



**Abbildung 3.20:** BME280 [10]

**Tabelle 3.4:** Elektrische Spezifikationen [10]

Parameter	Min.	Typ.	Max.	Einheit
Versorgungsspannung	1.71	1.8	3.6	V
Stromverbrauch (sleep mode)	0.1	0.3		$\mu$ A
Stromverbrauch inaktiv (normal mode)	0.2	0.5		$\mu$ A
Stromverbrauch Feuchtigkeitsmessung	340			$\mu$ A
Stromverbrauch Luftdruckmessung	714			$\mu$ A
Stromverbrauch Temperaturmessung	350			$\mu$ A

Bei einer Messfrequenz von 1Hz für die drei Messgrößen verbraucht der BME280 somit laut Datenblatt nur **3.6 $\mu$ A**. [10, S. 2]

#### Feuchtigkeitsmessung

In der Tabelle 3.5 sind die wichtigsten Parameter zur Feuchtigkeitsmessung aufgelistet. Zu vermerken ist, dass die digitalen Werte des BME280 zur Feuchtigkeitsmessung relativ sind und deshalb prozentual angegeben werden.

**Tabelle 3.5:** Sezifikationen der Feuchtigkeitsmessung [10]

Parameter	Min.	Typ.	Max.	Einheit
Operating range	-40 0	25	85 100	°C %
Absolute Genauigkeitstoleranz		±3		%
Hysterese		±1		%
Auflösung		0.008		%
Langzeitstabilität		0.5		% pro Jahr

### Luftdruckmessung

Die Genauigkeit der Luftdruckmessung ist an einen Temperaturbereich gebunden. Bei niedrigeren Temperaturen ( $<0^{\circ}\text{C}$ ) weist der Sensor eine höhere Unsicherheit auf als bei Temperaturen von 0 bis  $65^{\circ}\text{C}$  (siehe Tabelle 3.6).

**Tabelle 3.6:** Sezifikationen der Luftdruckmessung [10]

Parameter	Temperaturbereich	Min.	Typ.	Max.	Einheit
Operating range		-40 300	25	85 1100	°C hPa
Absolute Genauigkeit	-20 bis $0^{\circ}\text{C}$ $0$ bis $65^{\circ}\text{C}$		±1.7 ±1		hPa hPa
Auflösung			0.18		hPa
Langzeitstabilität			±1		hPa pro Jahr

### Temperaturmessung

Die Wetterstation wird hauptsächlich in einem Temperaturbereich von 0 bis  $65^{\circ}\text{C}$  betrieben, wodurch vom Sensor eine Unsicherheit von max.  $\pm 1^{\circ}\text{C}$  erreicht werden kann.

**Tabelle 3.7:** Sezifikationen der Temperaturmessung [10]

Parameter	Temperaturbereich	Min.	Typ.	Max.	Einheit
Operating range		-40	25	85	°C
Absolute Genauigkeit	$25^{\circ}\text{C}$ $0$ bis $65^{\circ}\text{C}$ -20 bis $0^{\circ}\text{C}$ -40 bis -20 C		±0.5 ±1 ±1.25 ±1.5		°C °C °C °C
Auflösung			0.01		°C

### 3.5 Ergänzungen aus der Bachelor-Thesis

Während der Bachelor-Thesis soll ein Sensor zur Ermittlung der Sonnenstunden implementiert werden. Um dies zu erreichen wurde die Idee, die Sonnenstunden direkt über den Ladestrom

der Photovoltaikanlage zu eruiieren, verworfen, da auf diese Weise mit höheren Energieverlusten zu rechnen wäre durch das Abzweigen des Ladestroms und die damit verbundene zusätzliche Speisung der zusätzlichen Schaltung. Stattdessen wird über ein Lichtintensitätssensor die Bestrahlungsstärke gemessen, womit die Sonnenstunden über einen Schwellwert berechnet werden können. Im folgenden Abschnitt wird der verwendete Lichtintensitätssensor näher erläutert.

### 3.5.1 Lichtintensitätssensor

Die Lichtintensität wird über den TSL2561, ein digitaler low power Sensor, eruiert. Dieser misst über eine Kombination aus einer Breitband-Photodiode und einer Infrarot-Photodiode die Lichtstärke mit einer 16-Bit Auflösung. Über zwei integrierte AD-Wandler wird der Photodiodenstrom gewandelt und auf einen digitalen Ausgang gegeben. Über eine I<sup>2</sup>C Schnittstelle kann dieser Ausgang auf die MCU gegeben und so ausgewertet werden. Ab einem bestimmten Schwellenwert kann die Einstrahlung als direkte Sonnenstrahlung definiert, und so auch gezählt werden. Der erwähnte Schwellwert ist vom Ort abhängig an dem die Wetterstation sich befindet und muss eruiert und in der Firmware eingestellt werden, was im entsprechenden Teil (Literaturpart:Firmware ??)) erläutert wird. Da dieser Sensor nicht direkt auf der Platine sondern am oberen Teil des Gehäuses montiert werden muss, wird der TSL2561 als Breakout Board (Abbildung 3.21) verwendet und über Pinheader mit dem PCB verbunden. Die wichtigsten elektrischen Spezifikationen sind in der Tabelle 3.8 zusammengefasst. [11]

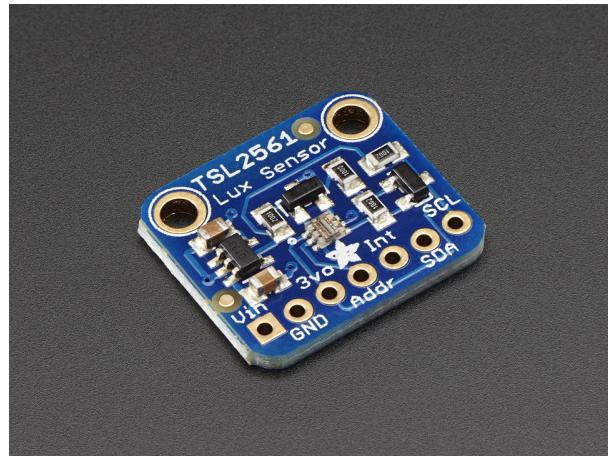


Abbildung 3.21: TSL2561 Breakout Board von Adafruit [12]

Tabelle 3.8: Elektrische Spezifikationen des TSL2561 [11]

Parameter	Min.	Typ.	Max.	Einheit
Versorgungsspannung	2.7	3	3.6	V
Stromverbrauch (Aktiv)		0.24	0.6	mA
Stromverbrauch inaktiv (Power down)	3.2	15		µA

### 3.5.2 BME280

Im Verlauf der Bachelor-Thesis wurde über das Design des Gehäuses der Wetterstation diskutiert, dabei wurde ebenfalls die Notwendigkeit eines Zugangs zu frischer Umgebungsluft für den BME280 erwähnt. Würde der BME280 auf dem PCB angebracht werden, so würde dieser durch die Abwärme der Elektronik stärker beeinflusst und Fehlmessungen generieren. Aus diesem Grund soll der BME280 an der unteren Seite des Gehäuses, zugänglich zu frischer Außenluft, montiert werden. Damit Fehlmessungen möglichst vermieden werden können, wird wie beim TSL2561 für den BME280 ein Breakout Board verwendet (Abbildung 3.22) und über Pinheader mit dem PCB verbunden.

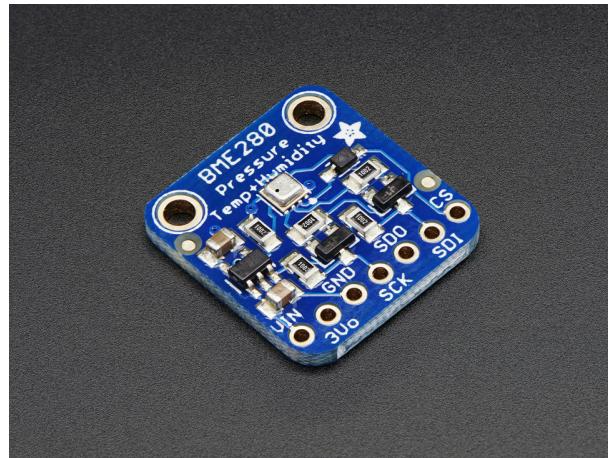


Abbildung 3.22: BME280 Breakout Board von Adafruit [13]

Die gesammelten Daten der Sensorik werden an die MCU geliefert. Die MCU kann jedoch nicht alle Daten speichern, weshalb ein grösserer, externer Speicher notwendig ist. Aus diesem Grund wird im nächsten Kapitel auf die Datenspeicherung eingegangen.

## 3.6 Datenspeicherung

Während des Projekt 5 wurde die Datenspeicherung bereits mittels Breakout Board im Prototyp implementiert. Nachfolgend wird die Dokumentation aus dem Fachbericht des Projekt 5 übernommen und in einem weiteren Abschnitt Ergänzungen dazu aus der Bachelor-Thesis erläutert.

### 3.6.1 Datenspeicherung - Projekt 5

Die Datenspeicherung beinhaltet die gespeicherten Messwerte der Sensoren. Die Daten werden dann in einem \*.txt-File nicht flüchtig gespeichert. Bei Beschädigung der Hardware können dann die zuletzt erfassten Daten immer noch mittels eines SD-Karten-Adapters von einem Computer ausgelesen werden. Als Kommunikationsprotokoll für das Schreiben und Auslesen der Karte wird SPI verwendet.

#### $\mu$ SD-Karte



Abbildung 3.23: 16 GB  $\mu$ SD-Karte [14]

Bei der  $\mu$ SD-Karte muss auf die Kompatibilität mit dem Breakoutboard geachtet werden. Dafür sind folgende Kriterien zu beachten:

- Die  $\mu$ SD-Karte muss FAT16 oder FAT 32 formatiert sein.
- Es sind nur die SD und SD High Capacity (SDHC) kompatibel.

Für die Umsetzung dieses Projektes wurde eine  $\mu$ SD-Karte der SD-Familie SDHC I verwendet (siehe Abb. 3.23). SDHC sind Kapazitäten bis zu 32GB möglich und FAT32 formatiert. [15] Die Grösse eines Strings lässt sich nach der Gleichung 3.3 berechnen, wenn angenommen wird, dass jeder Buchstaben (char) 8 Bit hat und zum Schluss noch ein Terminator für das Stringende angehängt wird:

$$\text{Anzahl Zeichen} * 1 \text{ Byte} + 1 = \text{String Grösse in Byte} \quad (3.3)$$

Bei einer Speicherung der Daten nach der Struktur in Abb. ?? würden somit leicht aufgerundet ca. 216 Byte pro Speichersatz benötigt werden.

Da momentan eine 16GB grosse  $\mu$ SD-Karte verwendet wird, ergeben sich daraus

$$\frac{16E9}{216} = 74.1E6 \approx \underline{\underline{74E6}} \quad (3.4)$$

Speichersätze. Würden in einem zehn Minuten Zeitintervall ein Speichersatz abgespeichert werden, dann könnten 1411 Jahre 40 Wochen 4 Tage 21 Stunden 20 Minuten lang Werte von der Wetterstation abgespeichert werden. Diese Anzahl könnte noch verdoppelt werden, da wie oben bereits erwähnt  $\mu$ SD-Karten bis zu 32GB kompatibel sind.

Kontrolle  
ref  
fig:datenausga

### 3.6.2 Ergänzungen aus der Bachelor-Thesis

Im Gegensatz zum Prototyping wird für die endgültige Wetterstation kein Breakout Board mehr verwendet. Die  $\mu$ SD-Karte wird in ein  $\mu$ SD-Karten-Adapter (Speicherkartenverbinder) gesteckt, welcher über ein Buffer (74HC4050) mit der MCU verbunden ist (siehe Abbildung 3.2). Der 74HC4050 kann als Levelshifter verwendet werden und als Treiber für kapazitive Ladungen. In unserem Fall werden MCU und Speicherkartenverbinder mit 3.3V betrieben, weshalb kein Levelshifter notwendig ist. Der 74HC4050 wird demnach bei der Wetterstation als Treiber für kapazitive Ladungen verwendet.

Die Daten werden mit der Sensorik ermittelt, über die MCU verarbeitet und in der  $\mu$ SD-Karte gespeichert. Nun sollen die Daten jedoch per SMS (GSM) abrufbar sein und die Wetterstation ebenfalls über ein GPS-Modul verfügen. Aus diesem Grund wird im nächsten Kapitel auf die Kommunikationsmodule (GSM, GPS) eingegangen.

### 3.7 Kommunikationsmodule

Gemäss Auftraggeber sollen Daten der Wetterstation per SMS abrufbar und der Standort der Wetterstation ermittelbar sein. Um SMS empfangen und auch senden zu können, wird ein GSM-Modul benötigt. Weiter wird ein GPS-Modul benötigt, um eine Standortermittlung durchführen zu können. Der SIM808 ist ein Kombinationsmodul aus GSM und GPS, was nicht nur Platz auf dem PCB, sondern auch Energietechnisch und Preislich Vorteile bietet, da nur ein Modul betrieben werden muss. Aus diesen Gründen wurde der SIM808 implementiert.

#### 3.7.1 SIM808

Der SIM808 (Abbildung 3.24) verfügt über GSM/GPRS (Global System for Mobile Communications / General Packet Radio Service), GPS (Global Positioning System) und BT (Bluetooth). Für die Wetterstation wird das GSM und das GPS verwendet. Tabelle 3.9 zeigt die elektrischen Spezifikationen des SIM808. [16]



Abbildung 3.24: SIM808 von Vorne [16]

Tabelle 3.9: Elektrische Spezifikationen des SIM808 [16]

Parameter	Min	Typ	Max	Einheit
Power Supply	3.4		4.4	V
Power saving		1		mA
Transmitting power (GSM)		2		W
SIM interface support		1.8/3		V
Power consumption (Idle GSM)		22.1		mA
Power consumption (Data GSM)		445.82		mA
Power consumption (TX Burst GSM)			2	A
Power consumption (GPS Acquisition)		42		mA
Power consumption (GPS Continuous tracking)		24		mA

Damit das GSM über den SIM808 genutzt werden kann, wird eine SIM-Karte benötigt. Dafür wurde eine Prepaid M-Budget SIM-Karte von Andres Minder erworben, welche jedoch Vertragsbedingt persönlich ist und nicht Teil der Wetterstation bleibt. Diese SIM-Karte wird lediglich zu Testzwecken genutzt und muss später durch eine SIM-Karte des Endbenutzers ersetzt werden. Die SIM-Karte wird mit einem SIM-Karten-Adapter (Speicherkartensteckverbinder) auf dem PCB integriert und mit dem SIM808 verbunden.

Wie in der Übersicht zu sehen ist (Abbildung 3.3), benötigt die SIM808 zwei verschiedene Antennen. Eine Antenne für das GSM, die andere für das GPS. Diese Antennen sind standardisiert und können direkt online erworben werden. Die Energieversorgung des SIM808 benötigt eine Schaltung aus Kondensatoren und einer Diode, welche die Spannung des Akkus von möglichen Störungen befreit und den SIM808 vor Überspannung schützt. Der verwendete 74VHC125 wird als Pegelwandler verwendet, da die SIM808 mit einer höheren Spannung betrieben wird als die MCU. Außerdem schützen Dioden den RST- und RX-Pin des SIM808 vor ungewollten Spannungsspitzen und deren Folgen (z.B. das Auslösen eines Reset). Die SIM-Karte wird in einem entsprechenden Adapter über ein Schutzdioden-Array mit dem SIM808 verbunden. Das Schutzdioden-Array schützt ebenfalls die SIM-Karte vor Überspannungen, indem diese über die Ground-Plane abgeleitet werden. Zwei Status-LED zeigen den Betrieb der SIM808 und dessen Konnektivität.

Die verschiedenen Systeme wurden erläutert, wobei die Energieversorgung bereits öfter angesprochen und nie thematisiert wurde. Im nächsten Kapitel soll dies nachgeholt und die Energieversorgung detailliert erläutert werden.

## 3.8 Energieversorgung

Die verschiedenen Bauteile auf dem PCB müssen mit Energie versorgt werden. Gemäss Auftragsbeschreibung soll die Wetterstation desweiteren über Photovoltaik gespiesen werden. Da die Sonne nur tagsüber scheint und um eine konstante Versorgung gewährleisten zu können, wird ein Akku benötigt. Um die Speisung via Akku und gleichzeitig ein Speisen des Akkus über die Photovoltaikanlage zu gewährleisten, muss der Ladestrom geregelt werden, was mit einem Power-Management-IC, dem MCP73871, erfolgt. Der MCP73871 führt die Speisung des Akkus weiter auf einen Linearregler (LM1117), welcher das 3.3V-Netz generiert. Damit der LM1117 eine saubere Ausgangsspannung (3.3V mit möglichst kleinem Rippel) generieren kann, muss dieser einen erhöhten Spannungspegel am Eingang haben, weshalb eine ChargePump (Ladungspumpe) benötigt wird. Das Konzept der Energieversorgung wird in Kapitel ?? vorgestellt, weshalb nun die Berechnung der Akkukapazität, der Einfluss der Photovoltaik, sowie die Kombination von ChargePump und Linearregler näher erläutert werden.

### 3.8.1 Berechnung der Akkukapazität

Um die Kapazität des Akkus zu berechnen, muss eine Schätzung des Energieverbrauchs gemacht werden. In Tabelle 3.10 werden deshalb die verwendeten Bauteile mit ihrem Stromverbrauch aufgelistet.

**Tabelle 3.10:** Übersicht über die verwendeten Bauteile mit ihrem Stromverbrauch.

Bauteilname	Modus	Min	Typ	Max	Einheit
MCP73871 [17]	Charge Complete	120	200	$\mu\text{A}$	
LM1117 [18]		5	10	mA	
SIM808 [16]	PWR Down	38	50	$\mu\text{A}$	
	Idle	22.1		mA	
	Data GSM	445.82		mA	
	GPS acquisition	42		mA	
	GPS continuous tracking	24		mA	
	TX Burst (peak)	2		A	
Anemometer					
Ombrometer					
Windrichtungsgeber [8]		0.028		4.8	mA
74VHC125 [19]		20		40	$\mu\text{A}$
ATMega2560 [5]	active	7		mA	
PN2222A	active	600		mA	
FT231XS [4]	normal	8	8	8.4	mA
	USB suspend	125		$\mu\text{A}$	
74HC4050 [20]			40	$\mu\text{A}$	
BME280 [10]	standby	0.2	0.5	$\mu\text{A}$	
	humidity measuring	340		$\mu\text{A}$	
	pressure measuring	714		$\mu\text{A}$	
	temperature measuring	350		$\mu\text{A}$	
TSL2561 [11]	active	0.24	0.6	mA	
	power down	3.2	15	$\mu\text{A}$	
DS3231 [6]	active		200	$\mu\text{A}$	
	standby		110	$\mu\text{A}$	

Tabelle 3.10 zeigt den Stromverbrauch der verwendeten Bauteile gemäss ihren Datenblättern. Das Anemometer und das Ombrometer besitzen keine Werte, da nur beim Schliessen eines der integrierten Reedkontakte ein Strom fließt, weshalb diese Ströme vernachlässigt werden. Die Ströme des Windrichtungsgebers wurden berechnet anhand der 3.3V Versorgungsspannung und den Widerstandswerten gemäss Datenblatt. Vom MCP73871 wurde lediglich der Strom bei geladenem Akku genommen, da das Laden des Akkus mit dem Ladestrom der Photovoltaik erfolgt und somit nicht direkt ein Verbrauch ist. Der SIM808 besitzt mit seinen 2A Spitzenstrom bei einem TX Burst (senden einer SMS) den höchsten Konsum, jedoch ist dieser Burst lediglich  $577\mu\text{s}$  lang und wird somit auch nicht berücksichtigt, zumal nicht gesagt werden kann wie oft dieser Burst tatsächlich vorkommt. Der SIM808 wird im Normalbetrieb das GPS im continuous tracking und das GSM im Idle Modus haben, weshalb hier insgesamt ein Strom von Typ 46,1 mA verbraucht wird. Der ATMega2560 wird im Normalbetrieb stets im active Modus sein, weshalb nur dieser Wert aus dem Datenblatt ausgelesen wurde. Die serielle Schnittstelle wird im Normalbetrieb nicht benutzt (kein Notebook angehängt), weshalb beim FT231XS der USB suspend Modus zur Berechnung benutzt und der PN2222A vernachlässigt wird. Der BME280 sollte stets Messungen durchführen, weshalb hier mit dem höchsten Wert (pressure measuring Modus) gerechnet wird. Ähnlich verhält es sich beim TSL2561 und beim DS3231, hier werden die active Modi in die Rechnung mit einbezogen.

Um den SIM808 mit einer optimalen Spannung versorgen zu können, wird ein Akku mit einer Spannung von 4V benötigt. Der Gesamtstrombedarf im Normalbetrieb beläuft sich gemäss Tabelle 3.10 und verwendeten Maximalwerten auf rund 70mA. Multipliziert man die Spannung mit dem Gesamtstrombedarf, so ergibt dies eine Leistung von 0.28 Watt. Um daraus die benötigte Akkukapazität zu ermitteln, muss festgelegt werden, wie lange dieser die Wetterstation mindestens speisen können soll. Gemäss Pflichtenheft soll der Akku die Wetterstation für mindestens 100 Stunden aufrecht erhalten können. Die berechnete Leistung wird mit den Anzahl Stunden multipliziert, was 28 Wh ergibt. Bei einer Akkuspannung von 4V ergibt das (durch dividieren der 28 Wh mit den 4V) eine Mindest-Akkukapazität von 7Ah oder auch 7000mAh.

### 3.8.2 Einfluss der Photovoltaik

Die Photovoltaik (das Solarpanel) erzeugt Strom durch Sonnenenergie. Der erzeugte Strom wird dazu verwendet den Akku zu laden, während dieser die Wetterstation speist. Im vorhergehenden Kapitel wurde die benötigte Leistung der Wetterstation von 0.28 Watt berechnet und die daraus resultierende Mindest-Akkukapazität von 7000mAh. Gemäss Pflichtenheft soll das Solarpanel den Akku innerhalb einem Tag laden können.

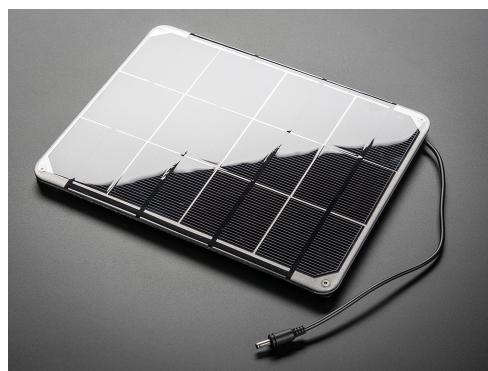


Abbildung 3.25: 6V 6W Solarpanel von Adafruit [21]

Da das Solarpanel nur bei Sonneneinstrahlung Strom erzeugen kann und die Sonnenstunden von Land zu Land abweichen, werden bei dieser Rechnung 6 Sonnenstunden pro Tag angenommen. Um 7000mAh innert 6h laden zu können, müssen konstant 1166.66mA bzw. rund 1.2A Strom erzeugt werden. Hier beschränkt jedoch der MCP73871, da dieser bei dementsprechender Be- schaltung ein Fast Charge Current von Typ 1A und Max 1.1A zulässt. Rechnet man zurück so wird das Solarpanel typischerweise 7h benötigen um bei einem erzeugten Ladestrom von 1A den Akku voll zu laden. Während den Sommermonaten sollte dies in den meisten Gegenden kein Problem darstellen, doch könnte dies über die Wintermonate nicht reichen. Um dies zu verifizieren rechnen wir den Verbrauch der Schaltung pro Tag aus und schauen, wie viele Sonnenstunden das Solarpanel bräuchte um bei 1A Ladestrom den Verbrauch auszugleichen. Bei 0.28W Verbrauch und 24h pro Tag ergibt das 6.72Wh. Da der Akku 4V liefert folgen daraus 1.68Ah Verbrauch für den Akku. Geht man wieder von einem Ladestrom von 1A aus, so werden 1.68 Sonnenstunden benötigt bei denen das Solarpanel 1A Ladestrom erzeugt um den Verbrauch zu decken. Aus diesem Grund kann gesagt werden, dass jedes Solarpanel wegen des MCP73871 nicht ausreichen wird, um die Mindest-Akkukapazität von 7000mAh innert 6 Sonnenstunden laden zu können. Wird jedoch ein Solarpanel verwendet das 1A Ladestrom erzeugen kann, so werden lediglich 1.68 Sonnenstunden benötigt um den Verbrauch zu decken. Für die Wetterstation wurde ein 6V 6W Solarpanel von Adafruit implementiert (Abbildung 3.25), was somit 1A Ladestrom erzeugen kann und ausreicht um den Verbrauch der Wetterstation innert 1.68 Sonnenstunden zu decken [21].

### 3.8.3 ChargePump und Linearregler

Die Wetterstation besteht aus einigen Bauteilen, welche eine 3.3V-Versorgungsspannung benötigen. Da der Akku eine höhere Spannung liefert, muss ein Linearregler die Spannung auf 3.3V regeln. Dafür wird der LM1117 verwendet, welcher eine Eingangsspannung von bis zu Max 15V aushält. Damit der LM1117 die Eingangsspannung sauber (möglichst ohne Rauschen) auf die 3.3V regeln kann, sollte die Eingangsspannung mindestens 5V betragen. Um von den 4V Batteriespannung auf mindestens 5V zu gelangen, wird eine ChargePump verwendet. Eine ChargePump ist eine Schaltung, welche aus einem Schalter, 2 Kondensatoren und 2 Dioden besteht, wie in Abbildung 3.26 schematisch dargestellt.

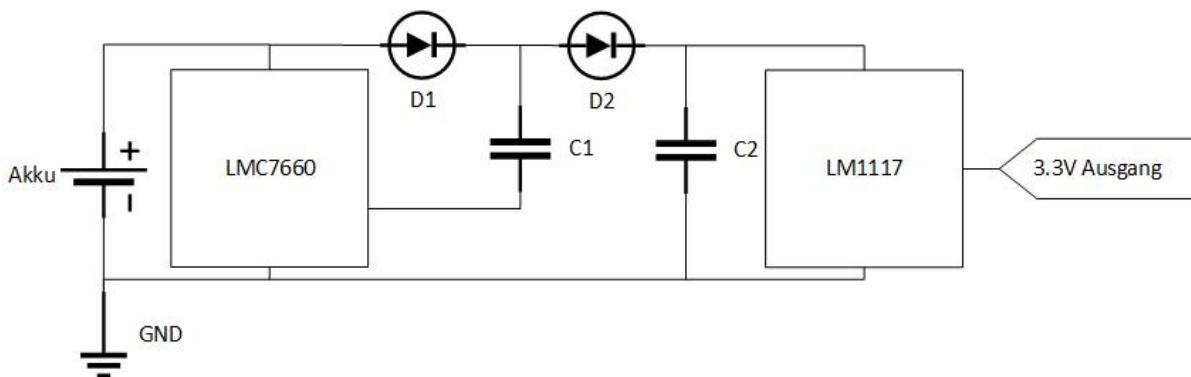


Abbildung 3.26: Schematische Darstellung der ChargePump.

Abbildung 3.26 zeigt die Schematische Darstellung der verwendeten ChargePump. Als Schalter wird der LMC7660 verwendet, welcher zwischen 2 Stromkreisen wechselt. Das wechseln der Stromkreise bewirkt das Aufladen des jeweiligen Kondensators. Im ersten Schalterzustand bilden D1, C1 und der Akku einen Stromkreis, wobei sich C1 auf die Akkuspannung lädt (abzüglich der Diodenspannung von D1). Im zweiten Zustand ist der Akku in Serie zu C1, welcher nun als Spannungsquelle fungiert, und der Kondensator wird auf die doppelte Akkuspannung (abzüglich der Diodenspannungen) geladen. In beiden Zuständen sorgen die Dioden dafür, dass sich der Kondensator kontrolliert auf die gewünschte Seite hin entlädt. Durch diese ChargePump soll im Idealfall eine Spannung von etwa 6.6V (Doppelte Akkuspannung von 8V abzüglich der zwei Diodenspannungen von je ca. 0.7V) generiert werden können. Durch diese erhöhte Spannung ist es dem LM1117 möglich die Spannung auf 3.3V zu regeln, wobei durch Toleranzen ein nur relativ kleiner Rippel entsteht. Die Spannung am Ausgang des Linearreglers ist Ausgangspunkt für die Speisespannungen der Bauteile, welche eine 3.3V-Speisespannung benötigen.

Nun wurde die Energieversorgung ebenfalls detailliert erläutert. Auf die verschiedenen Bauteilgruppen wurden nun in den verschiedenen Kapiteln jeweils näher eingegangen. Im nächsten Kapitel wird deshalb die Realisierung der Wetterstation auf einem PCB (Printed Circuit Board) thematisiert.

## 3.9 PCB

Ein PCB (Printed Circuit Board, dt. Leiterplatte) beherbergt die verwendeten Bauteile und verbindet diese elektronisch miteinander. Die verwendeten Bauteile werden auf das PCB gelötet, wobei es SMD (Surface Mount Device) und THT (Trough Hole Technology) zu unterscheiden gilt. Wie der Name schon sagt, werden THT-Bauteile durch das PCB gesteckt und SMD-Bauteile lediglich auf die Oberfläche gelegt, wobei beide vorgesehene Lötstellen haben auf diese sie gelötet werden um elektrischen Kontakt über Leiterbahnen zu erstellen. Das PCB und das dazugehörige Schema werden in einem EDA-Programm (Electronic Design Automation) erstellt. Für die Wetterstation wurde das Programm EAGLE (Einfach Anzuwendender Grafischer Layout-Editor) der Firma Autodesk verwendet, wegen der im Namen schon angedeuteten einfachen Anwendung. Für die Wetterstation wurde ein 4-Lagen-PCB ausgewählt, da dies im Vergleich zu einem 2-Lagen-PCB zu einem einfacheren Entwurf verhilft, hauptsächlich in Bezug zu Stromrückflüssen und EMV. Das Schema und das dazugehörige PCB-Layout sind im Anhang zu finden (Schema in Kapitel E, PCB-Layout in Kapitel F). Dem Anhang ist zu entnehmen, dass der erste und der vierte Layer für Signale bestimmt sind, der dritte Layer für die Versorgung (+3.3V) und der zweite Layer für das Ground (0V). In den nachfolgenden Abschnitten wird auf das Bestücken der Leiterplatte eingegangen und auf Mängel der ersten Version hingewiesen, sowie die Art und Weise wie diese Mängel behoben wurden.

### 3.9.1 Das Bestücken der Leiterplatte

THT Bauteile werden durch das PCB durchgesteckt und von unten her angelötet. Nach dem anlöten werden die Beine, welche standardmäßig relativ lang sind, möglichst nahe am PCB mit einem kleinen Seitenschneider gekürzt. In Abbildung 3.27 das Löten eines THT-Bauteils illustriert.

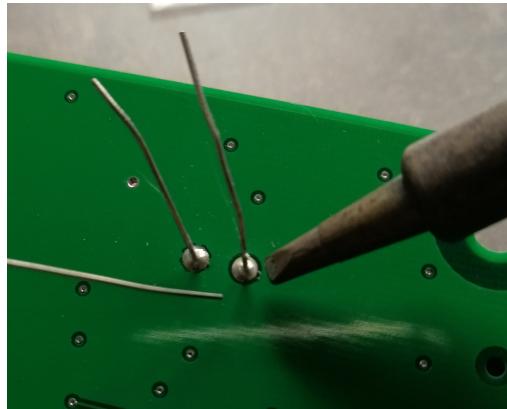
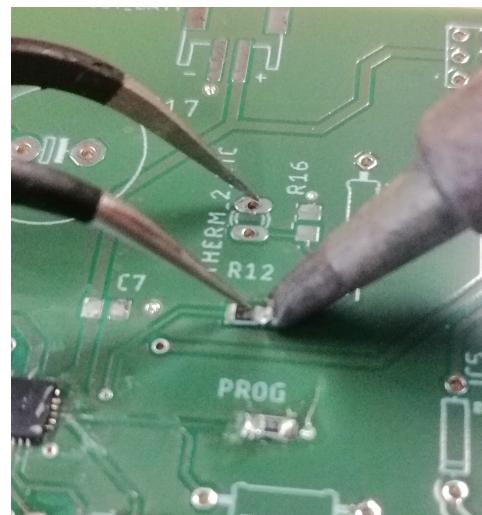


Abbildung 3.27: Das Löten eines THT-Bauteils.

SMD Bauteile werden auf die Oberfläche des PCBs gelegt und dort angelötet. Dabei wird zuerst auf dem ersten Lötauge etwas Lötzinn aufgetragen. Mit einer Pinzette wird nun das SMD Bauteil korrekt platziert, während mit dem Lötstab der vorher aufgetragene Lötzinn flüssig gehalten wird. Ist das Bauteil korrekt platziert, wird es mit der Pinzette angedrückt, so dass es sich nicht verschiebt wenn der Lötstab entfernt wird und dieser dann entfernt, so dass der Lötzinn sich erhärtet und das Bauteil festhält. Nun kann von der anderen Seite her mit dem Lötstab das Lötauge erhitzt und mit Lötzinn das Bauteil angelötet werden. Dieser Vorgang wird in Abbildung 3.28 illustriert.



**Abbildung 3.28:** Das Löten eines SMD-Bauteils.

Die MCU hat ein TQFP Gehäuse mit insgesamt 100 Pins (25 je Seite). Diese Pins sind äußerst nahe beieinander und sehr dünn, weshalb ein normales anlöten wie bei oben genannten SMD Bauteilen nicht gut funktioniert. Hier wird das oft *Ziehlöten* genannte Verfahren angewendet. Dabei wird die MCU erst äußerst sorgfältig korrekt platziert. Mit einer Pinzette wird die MCU an Ort und Stelle gehalten. Die Pins auf einer Seite werden nun großzügig mit Flussmittel eingestrichen. Direkt nach dem Einstreichen muss der Lötstab mit dem breiten Lötaufsatz und ganz wenig Lötzinn über die Pins fahren, um den Lötzinn auf die Pins zu verteilen. Nun kann wieder mit dem Flussmittel gearbeitet werden, wobei der Lötstab frontal von den Enden der Pins (MCU-seitig) nach aussen gezogen wird. Lötzinn muss im zweiten Schritt meistens keiner mehr verwendet werden und wenn doch, dann nur äußerst wenig. Die Kunst besteht darin, nicht zu viel Lötzinn zu verwenden, da sonst Lötbrücken entstehen können, welche nicht ganz einfach zu entfernen sind. Dieses Verfahren wird in Abbildung

ref Bild einfügen MCU löten

dargestellt.

### 3.9.2 Mängel der Erstversion und dessen Verbesserungen

Die Mängel der Erstversion werden hier aufgelistet und deren Verbesserung für die Zweitversion direkt erläutert.

1. Beim ersten Testen des Boards wurde erkannt, dass Probleme mit dem Versorgungslayer bestehen, da fast alle Bauteile, welche mit diesem über ein Via (ein Verbinder zwischen Layer) verbunden sein sollten, nicht mit Strom versorgt wurden. Dieses Manöver wurde provisorisch mit einer Leiterplatte und Drähten gelöst, so dass weitere Tests gemacht werden konnten.  
Für die Zweitversion wurde die Füllung des Versorgungslayers mit direkten Leiterbahnen zu den entsprechenden Pins ersetzt.
2. In einem weiteren Schritt wurde erkannt, dass die ChargePump nicht die gewünschte Spannung ausgibt wie auf einem Steckbrett zuvor getestet, wobei sich dieser Mangel erst bei geringerem Akkustand bemerkbar macht, weshalb weitere Tests gemacht werden konnten. Um dieses Problem zu beheben, sollen andere Dioden verwendbar sein. Aus diesem Grund werden für den zweiten Print THT-Bauteile verwendet, da diese einfacher auszulöten und zu ersetzen sind.

3. Gemäss den Tests sperren die Schutzdiode in der Nähe des DCIN Jacks nicht wie gewünscht in Sperrrichtung.  
Auch hier werden für den zweiten Print THT-Bauteile verwendet, um ein austauschen der Dioden vornehmen zu können.
4. Pins für die serielle Schnittstelle (TXD und RXD) wurden vertauscht, weshalb keine Kommunikation über PuTTY erfolgen konnte.  
Dieses Problem wurde im Redesign behoben, indem die Anschlüsse vertauscht wurden.
5. Beim SIM808 muss der Pin PWRKEY auf Ground sein um diesen starten zu können.  
Dieser Pin war jedoch nicht auf Ground gezogen sondern floating.  
Der entsprechende Pin wurde im Redesign mit Ground verbunden.
6. Desweiteren konnte beim SIM808 die 1.8V Ausgangsspannung für die SIM-Karte nicht gemessen werden. Es wurde vermutet, dass es wegen des TVS-Arrays nicht möglich war. Für das TVS-Array wurde ein zu kleiner Footprint verwendet, weshalb die Lötaugen nicht zur Grösse des Gehäuses passte, was schliesslich zu defekten auf dem PCB führte.  
Der Footprint wurde von Hand angepasst, so dass das Bauteil nun richtig angelötet werden kann. Ausserdem wurde eine Bestückungsvariante implementiert, mit der der SIM808 mit Breakout Board direkt angeschlossen werden kann über Pinheader.
7. Die Signale des Anemometers und des Ombrometers waren verrauscht, was auf einen ineffektiven Tiefpass hindeutet. Die Abstände zwischen Widerstand und Kondensator waren wirklich etwas weit entfernt.  
Die Abstände wurden im Redesign massiv verringert.
8. Der verwendete Kippschalter trennt die Wetterstation von der Versorgung, jedoch ist der SIM808 direkt vom Akku gespiesen und läuft deshalb weiter.  
Durch das Ersetzen des Kippschalters mit einem anderen, welcher 2 Stromkreise gleichzeitig schaltet, konnte dieses Problem im Redesign behoben werden.

Die Mängel wurden gemäss den Erläuterungen behoben und ein zweites PCB bestellt. Das eingetroffene PCB wies weiterhin Problem 1 auf, was erst zu Ratlosigkeit führte. Schnell konnte jedoch herausgefunden werden, dass der Hersteller des PCBs (JLCPCB) keine Blind Vias unterstützt, welche jedoch für die Verbindung zwischen Bauteil und Versorgungslayer im Design verwendet wurden. Es musste deshalb ein drittes PCB bei einem Hersteller bestellt werden, welches Blind Vias unterstützt. Auf der Webseite [www.multi-circuit-boards.eu](http://www.multi-circuit-boards.eu) konnten Blind Vias als Option angekreuzt werden, weshalb bei diesem Hersteller das dritte PCB bestellt wurde. Bestellen musste dies jedoch die FHNW, da dieser Anbieter nur an Institutionen und Unternehmen liefert und die Projektteilnehmer die Kosten eines PCBs von rund 280.- nicht auf sich nehmen konnten.

Das erfolgreich erhaltene dritte PCB wurde wie erläutert bestückt und getestet gemäss den dokumentierten Tests in Kapitel 4.1.1. Die Firmware konnte ebenso auf die MCU geflasht werden und funktioniert gemäss.

Referenz zum Kapitel Firmwareverifikation

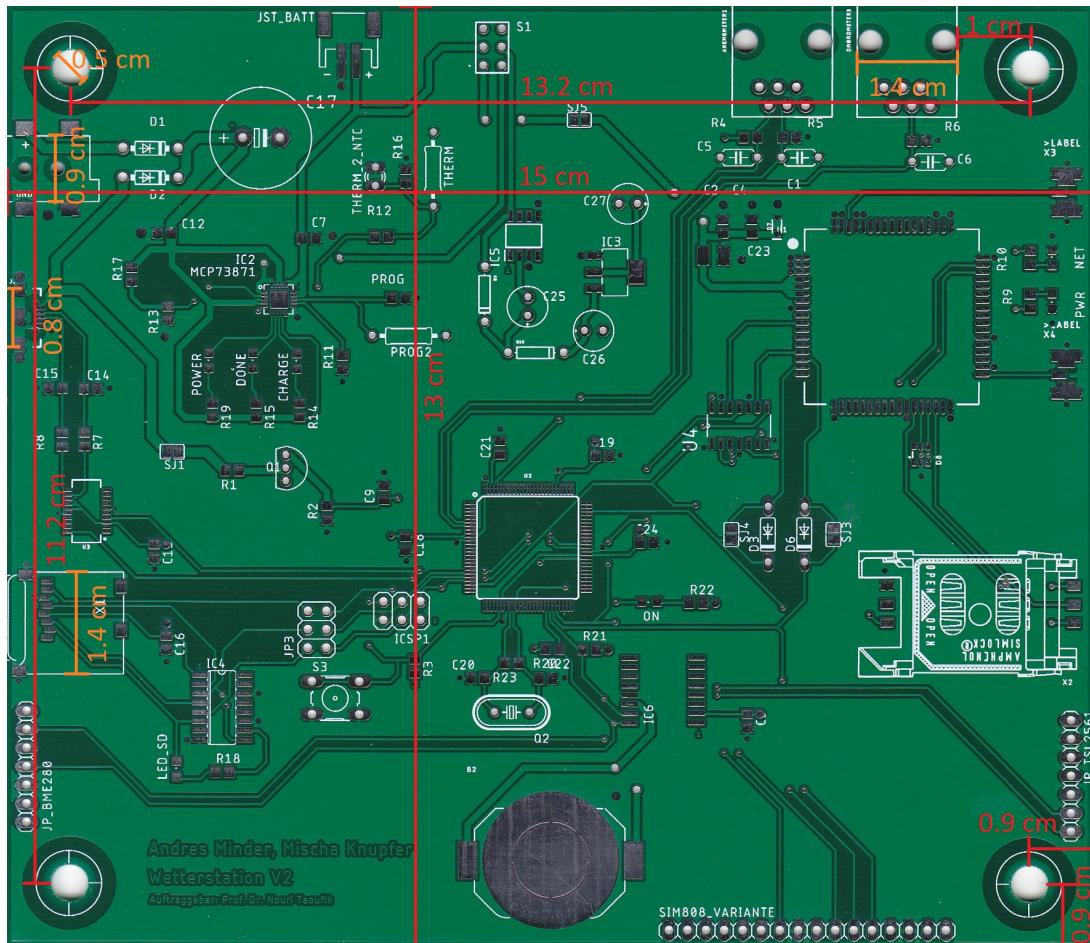
Die Wetterstation funktioniert soweit und benötigt lediglich ein Gehäuse, um präsentabel zu sein und um ein Betrieb im Freien zu ermöglichen. Auf diese Thematik wird im nächsten Kapitel eingegangen.

### 3.10 Das Gehäuse

Damit die Wetterstation gebraucht werden kann, muss das PCB vor den Wettereinflüssen geschützt werden. In Absprache mit dem Auftraggeber soll als Wunschziel ein provisorisches Design des Gehäuses erfolgen und dieses mit einem 3D-Drucker gedruckt werden um bei der Bachelor-Thesis Ausstellung am 16. August 2019 ein anschauliches Modell zu haben. Im folgenden Unterkapitel wird aufgezeigt, auf was beim Design des Gehäuses acht gegeben werden muss und in einem folgenden Unterkapitel wird das Design für das provisorische Gehäuse vorgestellt.

#### 3.10.1 Wichtige Merkmale

Damit beim Design des Gehäuses nichts vergessen geht, werden hier die wichtigsten Punkte, auf die acht gegeben werden muss, erläutert. Die grobe Dimensionierung wird durch das PCB vorgegeben und ist in Abbildung 3.29 zu sehen.



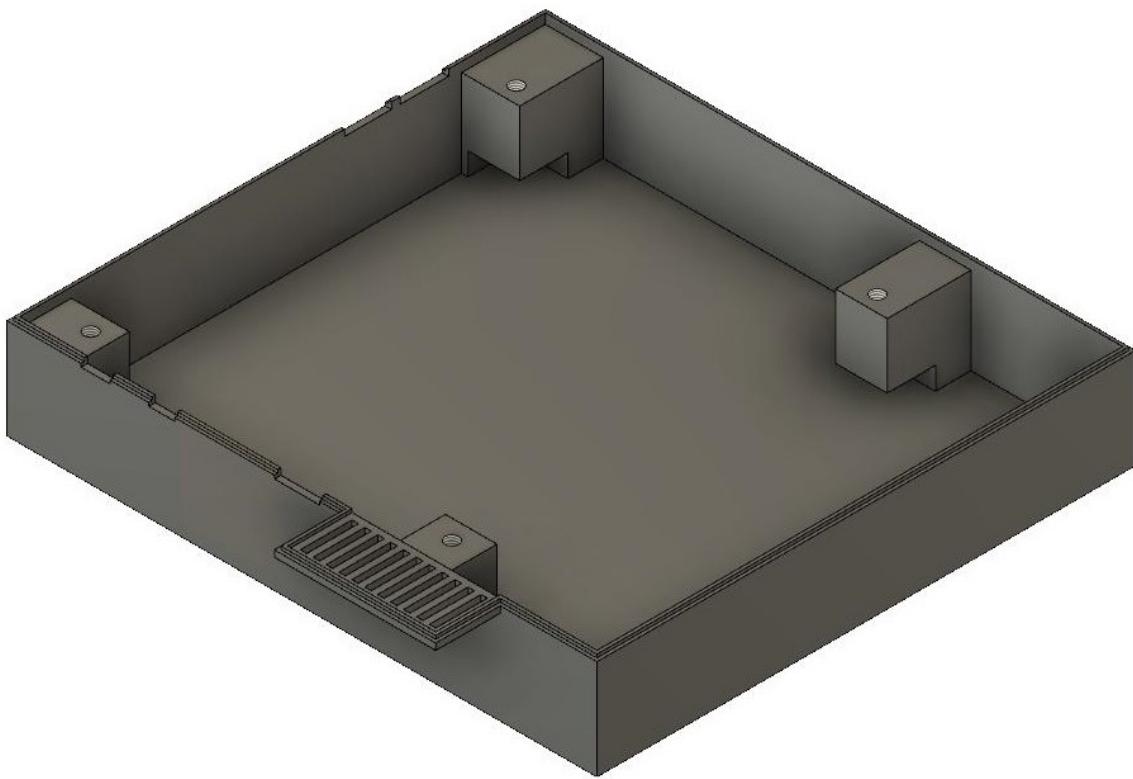
**Abbildung 3.29:** Das noch unbestückte PCB mit den wichtigsten Dimensionen.

Abbildung 3.29 zeigt das noch unbestückte PCB, wie es schon in der Hardware Übersicht verwendet wurde, mit den wichtigsten Dimensionen. Das PCB hat eine Breite von 15cm, eine Länge von 13cm. Die Mittelpunkte der 5mm breiten Bohrlöcher sind 13.2cm weit entfernt in der Breite und 11.2cm in der Länge, wobei diese jeweils 9mm vom Rand entfernt sind. Die Breite der Bauteile die aus dem Gehäuse hinausragen sind 1.4cm (RJ11,  $\mu$ SD), 0.9cm (DCIN Jack) und 0.8cm ( $\mu$ USB). Damit alle Bauteile von der Höhe her Platz haben, muss das Gehäuse nach oben hin mindestens über 3cm Freiraum verfügen wegen dem zum MCP73871 gehörenden Elektrolyt-Kondensator (C17). Der Akkumulator soll ebenfalls im Gehäuse Platz finden, weshalb genügend Platz (min. 2cm) in der Tiefe für diesen zur Verfügung stehen muss. Da es eine Bestückungsvariante gibt für den SIM808, soll das Gehäuse an der unteren Seite um mindestens 4cm erweitert werden, da das Breakout Board des SIM808 ansonsten nicht angeschlossen werden kann. Auf der rechten Seite werden Antennen für den SIM808 angeschlossen, weshalb das Gehäuse auf dieser Seite um mindestens 1cm erweitert werden soll, damit diese nicht knicken. Ebenfalls sollen auch die Kabel des Akkumulators vom knicken geschützt werden, hier benötigt man eine Erweiterung des Gehäuses von mindestens 3mm, da die Kabel selbst eine höhere Flexibilität als die Antennen aufweisen. Auf der linken Seite befindet sich die  $\mu$ USB-Buchse, sowie die Öffnung für die  $\mu$ SD-Karte. Diese Öffnungen ragen ein wenig hinaus und sollten gut geschützt werden, weshalb eine Erweiterung der Dimensionierung auf der Linken Seite von 1mm empfohlen ist. Äußerst wichtig ist ebenfalls, dass der BME280 auf der linken Seite des Gehäuses nach aussen gebracht wird und in einem gut mit Aussenluft durchströmten eigenen Raum seinen Platz findet. Der TSL2561 wird ebenfalls auf der rechten Seite nach aussen geführt, damit dieser dort unter einem mit durchsichtigem Plexiglas versehenen eigenen Raum die Lichtintensität messen kann.

Die wichtigsten Punkte wurden erläutert, weshalb es nun zum Design des Gehäuses kommt, welches im nächsten Abschnitt vorgestellt wird.

### 3.10.2 Das Design

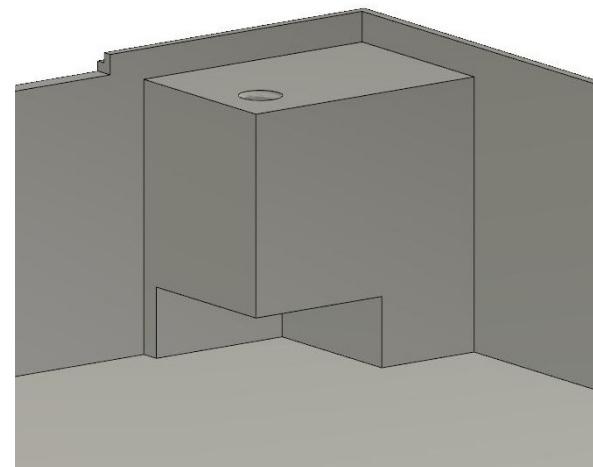
Im vorhergehenden Unterkapitel wurden die wichtigsten Merkmale und Dimensionen des Gehäuses erläutert. In diesem Unterkapitel wird nun das provisorische Design vorgestellt. Da für den Zusammenbau das PCB auf einen Teil des Gehäuses gelegt wird und mit einem zweiten Teil überdeckt, wird in diesem Abschnitt der Begriff *Boden* für den unteren Teil und *Deckel* für den oberen Teil verwendet. Das Design wurde mit dem CAD-Programm Autodesk Fusion 360 erstellt, welches den Studenten von Bildungseinrichtungen wie der FHNW frei zur Verfügung steht.



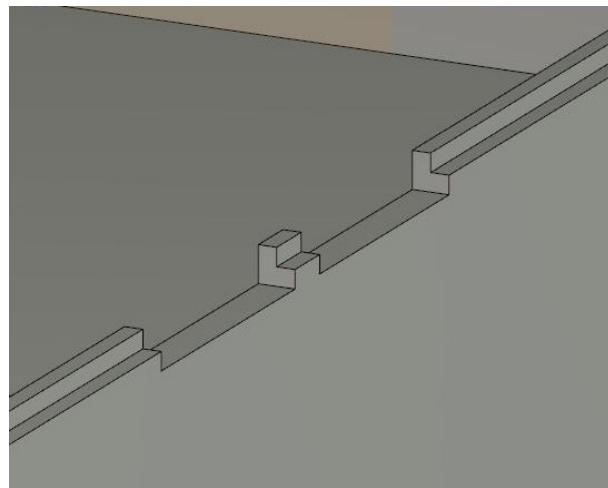
**Abbildung 3.30:** Gesamtansicht des Designs des Bodens.

Die Gesamtansicht des Designs des Bodens ist in Abbildung 3.30 zu sehen. Diverse Blöcke und Aussparungen sind deutlich zu sehen, worauf mit der Hilfe von den nachfolgenden Abbildungen näher eingegangen wird.

Das PCB muss erhöht sein, damit der Akku darunter Platz findet. Außerdem soll eine Schraube das Gehäuse mit dem PCB verbinden. Aus diesen Gründen wurden im Boden Blöcke implementiert, welche ein Schraubengewinde besitzen, wie in Abbildung 3.31 ersichtlich ist. Am unteren Ende des Schraubenblocks befindet sich eine Aussparung. Diese Aussparung bietet Platz für eine Mutter, falls das implementierte Gewinde fehlerhaft gedruckt wird.

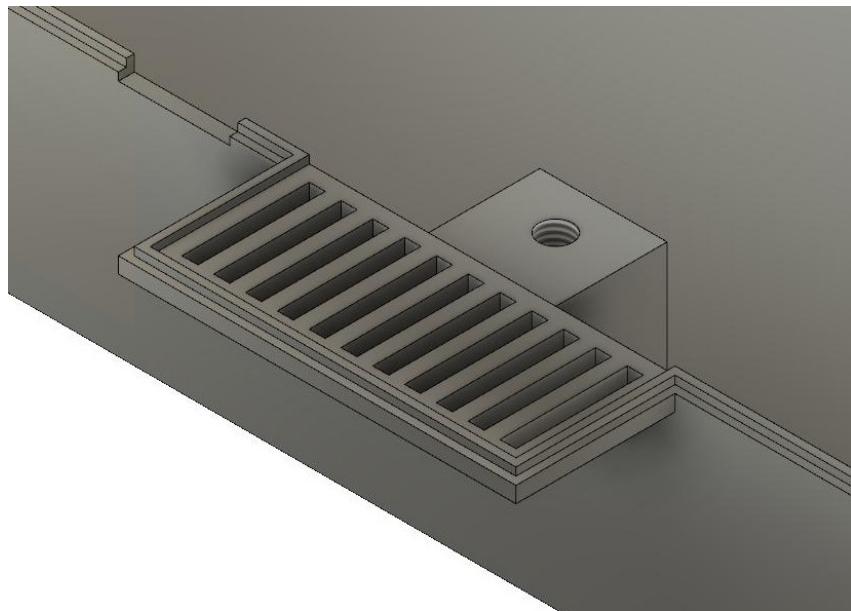


**Abbildung 3.31:** Design des Schraubenblocks des Bodens.



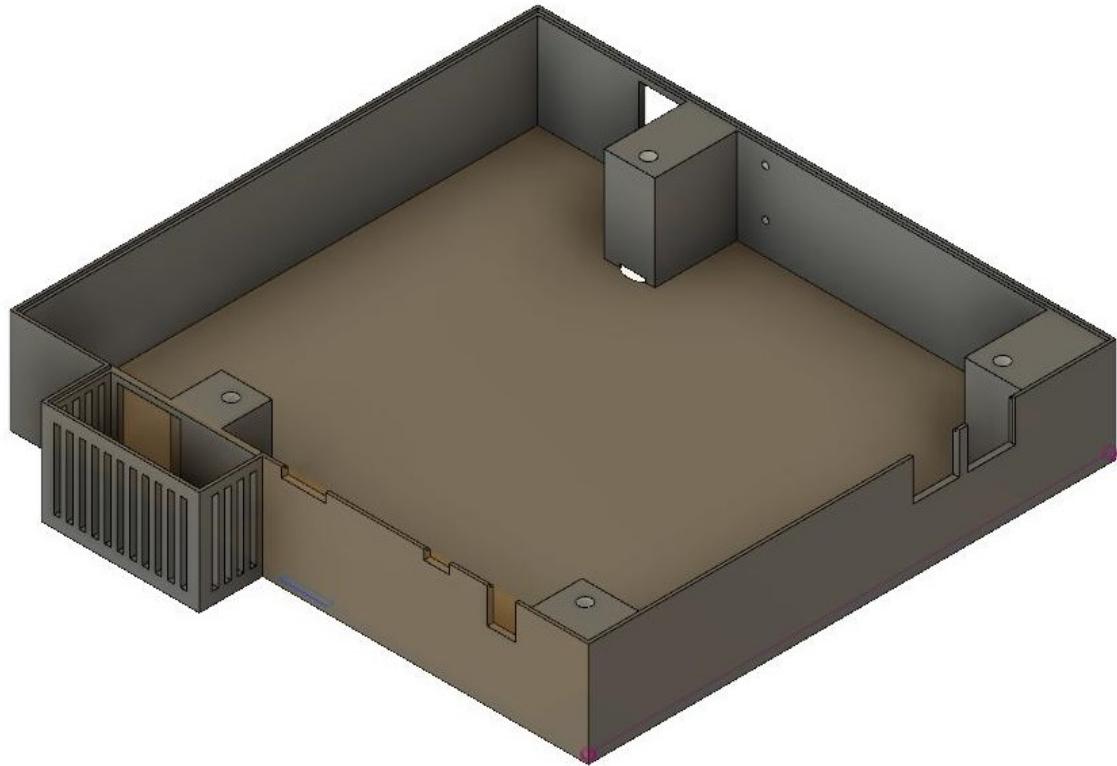
**Abbildung 3.32:** Aussparungen für Bauteile, welche aus dem Gehäuse hinausragen.

Die in Abbildung 3.32 ersichtlichen Aussparungen sind notwendig um Öffnungen im Gehäuse zu generieren, welche Platz für die hinausragenden Bauteile bieten. Beim Boden sind diese Aussparungen alle gleich tief, da die Bauteile alle auf dem planen PCB angelötet sind und dieses somit als Referenz gilt.



**Abbildung 3.33:** Seitenwand für den BME280-Extraraum mit Luftschlitten.

Der BME280 muss nach aussen geführt werden in einen kleinen Extraraum, welcher gut belüftet ist. Nur so ist der BME280 in der Lage, die exakten Messwerte liefern zu können. Aus diesem Grund wird ein kleiner Extraraum am Rand des Gehäuses gefertigt, welcher mit Luftschlitten versehen ist. In Abbildung 3.33 sieht man den Boden für den im Deckel implementierten Extraraum. Der Boden ist, wie erwähnt, mit 2mm breiten Luftschlitten versehen.



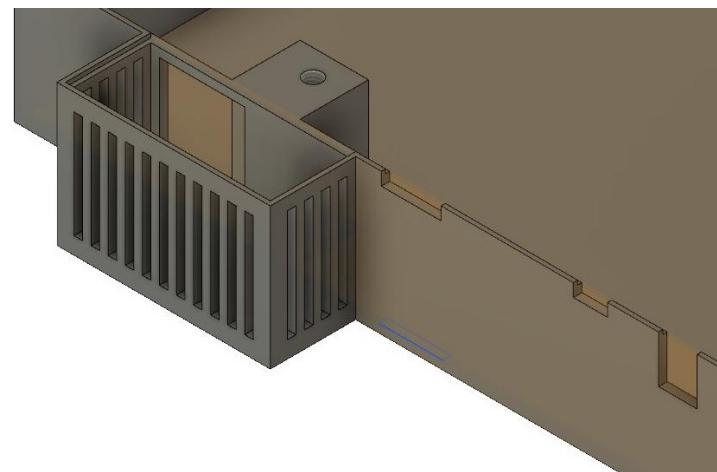
**Abbildung 3.34:** Gesamtansicht des Designs des Deckels.

Die Gesamtansicht des Designs des Deckels ist in Abbildung 3.34 zu sehen. Wie schon beim Boden sind auch hier diverse Blöcke und Aussparungen zu sehen, auf welche mit der Hilfe von den nachfolgenden Abbildungen näher eingegangen wird.

Die Schraubenblöcke des Deckels sind weniger interessant, da diese durchgehende Blöcke sind mit einem entsprechenden Gewinde. Aus diesem Grund wurde in Abbildung 3.35 die Außenseite des Deckels abgebildet, welche die Schraubenversenkung zeigt. In dieser Schraubenversenkung wird der Kopf der verwendeten Schraube seinen Platz finden, weshalb es zu keinem herausragen einer Schraube kommt.



**Abbildung 3.35:** Schraubenversenkung im Deckel.



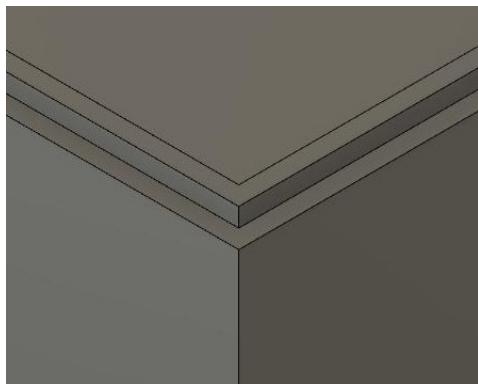
**Abbildung 3.36:** Aussparungen für Bauteile, welche aus dem Gehäuse hinausragen und der designte Extraraum für den BME280 mit Luftschlitzten.

Abbildung 3.36 zeigt den bereits erwähnten Extraraum des BME280. Die Luftschlitzte sind deutlich zu erkennen und an jeder Seite des Extraraumes angebracht. Außerdem ist auf dieser Abbildung ebenso ersichtlich, dass die Aussparungen verschiedene Höhen aufweisen. Die Höhe einer Aussparung ist Bauteilabhängig, genauso wie deren Breite.

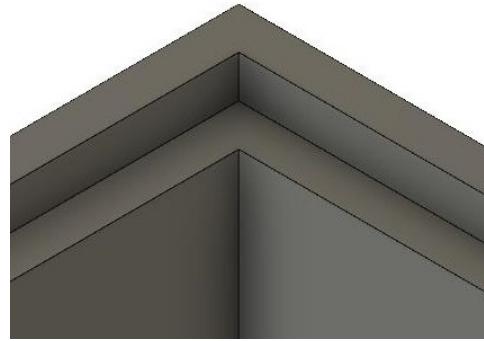


**Abbildung 3.37:** Öffnung und Bohrungen für den TSL2561 und den BME280.

Um den BME280 und den TSL2561 hinausführen zu können, wurden im Gehäuse entsprechende Öffnungen und auch Schraubenlöcher eingelassen. Die eben genannten Öffnungen und Schraubenlöcher sind in Abbildung 3.37 äusserst gut ersichtlich.



**Abbildung 3.38:** Design des Randes vom Boden zum Deckel hin.



**Abbildung 3.39:** Design des Randes vom Deckel zum Boden hin.

Damit das Befestigen des Bodens mit dem Deckel einfacher zu handhaben ist, wurde beim Boden die äussere Hälfte des Randes um 1mm gesenkt, wobei beim Deckel die innere Hälfte des Randes um 1mm gesenkt wurde. Dank diesem Feature ist es möglich den Deckel wie ein Puzzlestück am Boden anzubringen, bevor die Verschraubung von statten geht.

Das Design des Gehäuses für das PCB wurde in diesem Unterkapitel näher erläutert. Es sei jedoch erwähnt, dass dieses Gehäuse nicht geeignet ist um die Wetterstation im freien betreiben zu können. Das hier vorgestellte Design zeigt lediglich auf, auf was beim Gehäuse, in Bezug auf das PCB, geachtet werden muss. Damit das Gehäuse für den Gebrauch im freien taugt, muss es zwingend gegen Regen und Staub dicht sein, so dass die verwendete Elektronik keine Schäden erleidet. Dennoch muss der BME280 mit Aussenluft belüftet werden, was bei einem möglichst Regen- und Staubdichten Gehäuse eine zusätzliche, kontrollierte Belüftung notwendig macht. Ausserdem muss auf einen möglichen Hitzestau durch Sonneneinstrahlung getestet und gegebenfalls eine Kühlung integriert werden.

## **4 Abschluss**

## 4.1 Konzeptvalidierung

### 4.1.1 Validierung der Hardware

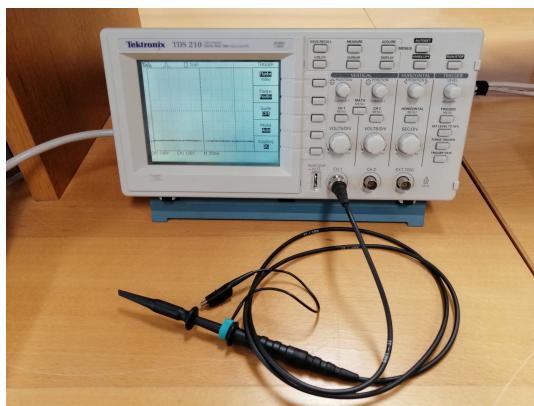
Die Validierung der Hardware ist ein wichtiger Schritt um Denk- und Designfehler zu entdecken bzw. den geplanten Betrieb der Hardware sicherzustellen. Aus diesem Grund wird in diesem Unterkapitel die Validierung der Hardware thematisiert und mit Abbildungen anschaulich dargestellt. Bevor jedoch die Validierung von statthen geht, werden die verwendeten Gerätschaften im nächsten Abschnitt kurz erläutert.

#### Verwendete Gerätschaften

Damit die Hardware validiert werden kann, braucht es spezielle Geräte. Zum einen wird ein FLUKE 117 (Abbildung 4.1), ein handliches RMS Multimeter, benutzt. Zwei Hauptfunktionen des FLUKEs werden hauptsächlich verwendet, diese sind das Messen von Gleichspannung, sowie die Messgerätelfunktion Kontinuität, um zu testen ob elektrische Verbindungen bestehen **FLUKE**. Zum anderen wird der TDS 210, ein Digitales Real-Time Oszilloskop von Tektronix, verwendet, um Spannungsverläufe zu analysieren. In Abbildung 4.2 ist dieser zu sehen. Um genaue Eingangsspannungen generieren zu können, wird ein Speisegerät verwendet, welches in der Abbildung 4.3 zu sehen ist.



**Abbildung 4.1:** Das handliche RMS Multimeter FLUKE 117 mit Sonden.



**Abbildung 4.2:** TDS 210 von Tektronix.



**Abbildung 4.3:** Das verwendete Speisegerät.

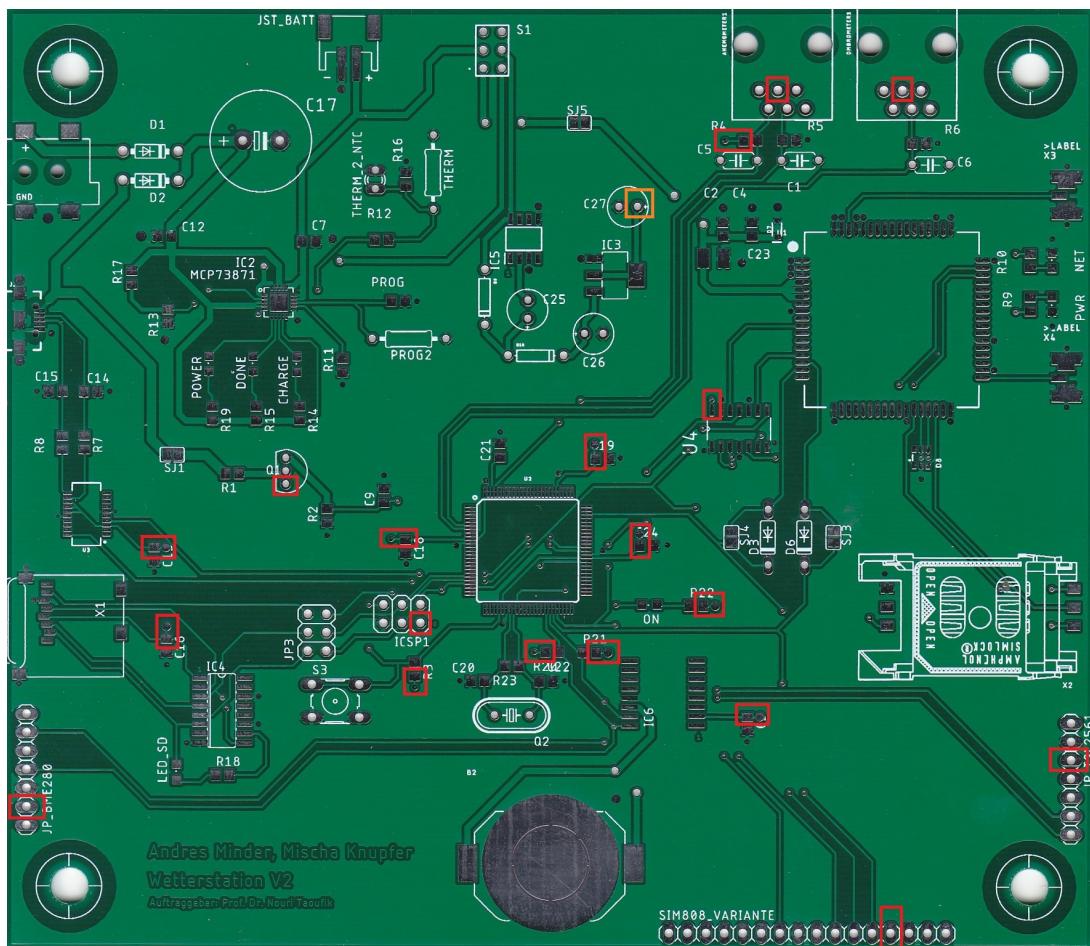
Die Geräte wurden nun vorgestellt, weshalb im nächsten Abschnitt bereits mit der Validierung der Energieversorgung begonnen wird.

### Validierung der Energieversorgung

Die Energieversorgung ist äusserst wichtig, da ohne eine funktionierende Energieversorgung kein Bauteil seiner Funktion nachgehen kann. Zuerst wird kontrolliert, ob die Verbindungen der Leiterbahnen für das 3.3V-Netz bestehen. Dies wird mit der Kontinuitätsfunktion (Abbildung 4.4) des FLUKES gemacht, bei welcher ein Ton erklingt sobald eine elektrische Verbindung besteht (kleiner Widerstand). Es werden vom Ausgang des Linearreglers aus alle Verbindungen getestet, welche in Abbildung 4.5 eingezeichnet wurden.



**Abbildung 4.4:** Das Zeichen für die Kontinuitätsfunktion des FLUKES.



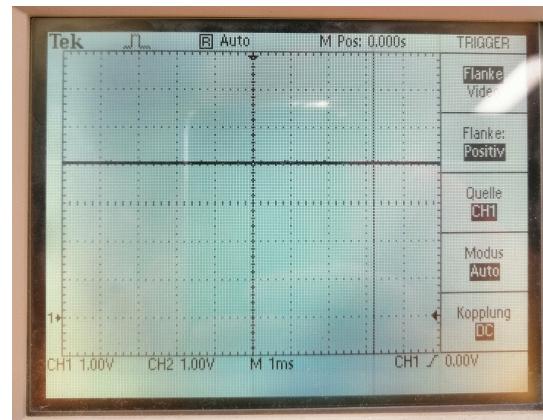
**Abbildung 4.5:** Die Rot markierten Stellen zeigen die Endpunkte des Verbindungstests an. Die Orange markierte Stelle zeigt den Startpunkt an.

Die Verbindungen konnten erfolgreich verifiziert werden. Bei jedem Punkt gemäss Abbildung 4.5 ist ein Signalton aus dem FLUKE gekommen, was eine bestehende elektrische Verbindung signalisiert.

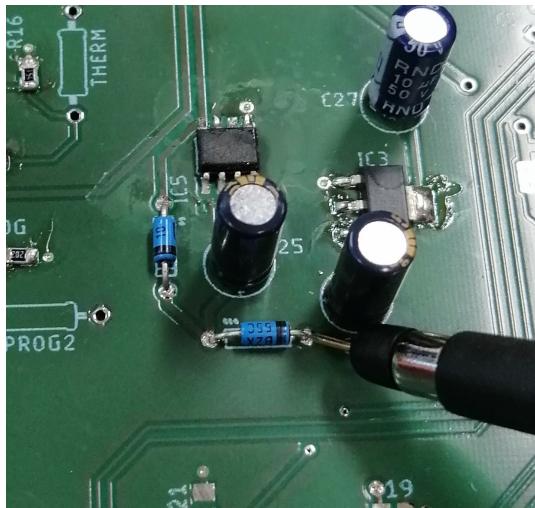
Als zweites wird nach dem Bestücken der Energieversorgung diese ausgetestet. Es wird überprüft, ob der Linearregler am Ausgang auf 3.3V regelt. Außerdem muss überprüft werden, ob die ChargePump die Akkuspannung auf das gewünschte Potenzial bringt. Dariüber hinaus muss noch getestet werden, ob die Dioden in der Nähe des DCIN Jacks wie gewünscht in Sperrrichtung sperren. Für diese Tests müssen Spannungsverläufe auf dem TDS 210 betrachtet werden, sowie mit dem Speisegerät eine Eingangsspannung erzeugt werden. Die nachfolgende Bilder zeigen jeweils die Testpunkte der Messung, sowie die Visualisierung der gemessenen Spannung auf dem TDS 210.



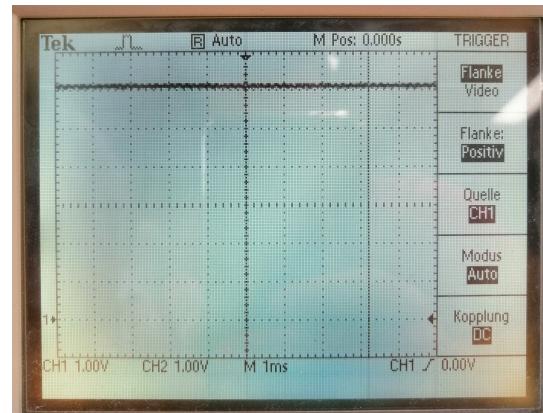
**Abbildung 4.6:** Testpunkt zur Überprüfung der Eingangsspannung.



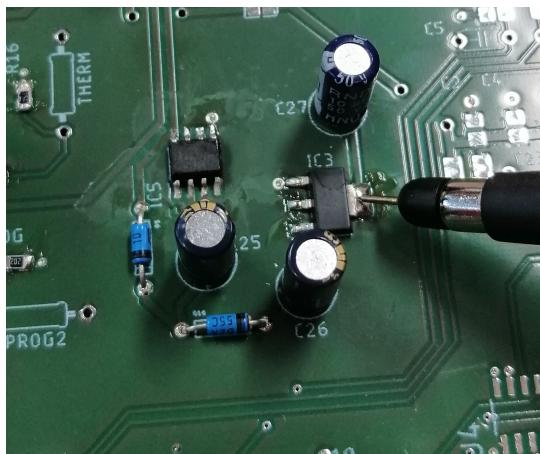
**Abbildung 4.7:** Die Eingangsspannung auf dem TDS 210 visualisiert. Es werden 4V angezeigt.



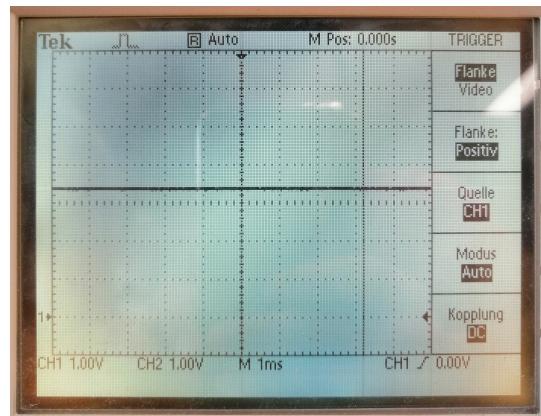
**Abbildung 4.8:** Testpunkt um die Spannungserhöhung der ChargePump zu messen.



**Abbildung 4.9:** Die durch die ChargePump erhöhte Spannung, welche auf dem TDS 210 visualisiert wurde. Es werden etwas mehr als 6V angezeigt.



**Abbildung 4.10:** Testpunkt für die Ausgangsspannung des Linearreglers.



**Abbildung 4.11:** Die durch den Linearregler geregelte Ausgangsspannung. Es werden rund 3.3V angezeigt.

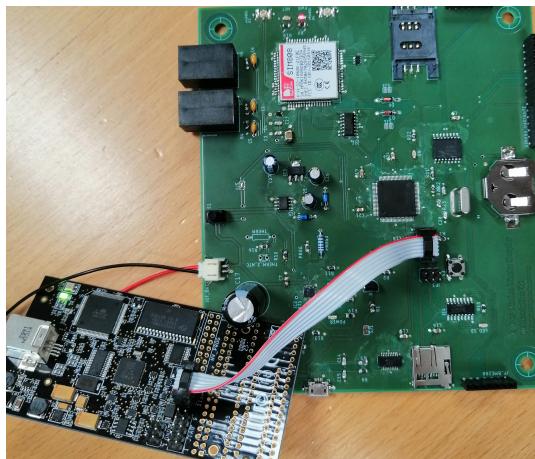
Die Abbildungen 4.6, 4.8 und 4.10 zeigen die Testpunkte, an denen gemessen wurde. Die Ergebnisse davon sind in den Abbildungen 4.7, 4.9 und 4.11 zu sehen. Das Ergebnis der Eingangsspannung zeigt, dass kein Kurzschluss vorhanden ist und die Spannung vom Speisegerät her auf dem PCB ankommt. Gemäss dem Ergebnis der Messung der ChargePump stellt man fest, dass diese ebenfalls funktioniert und die Spannung deutlich erhöht. Der Linearregler erzeugt aus der Spannung der ChargePump die benötigten 3.3V, was dem Ergebnis dieser Messung zu entnehmen ist. Es kann gesagt werden, dass die Energieversorgung soweit funktioniert.

Wie niedrige Spannung noch funktionstauglich, messung mit solarpanel ebenfalls

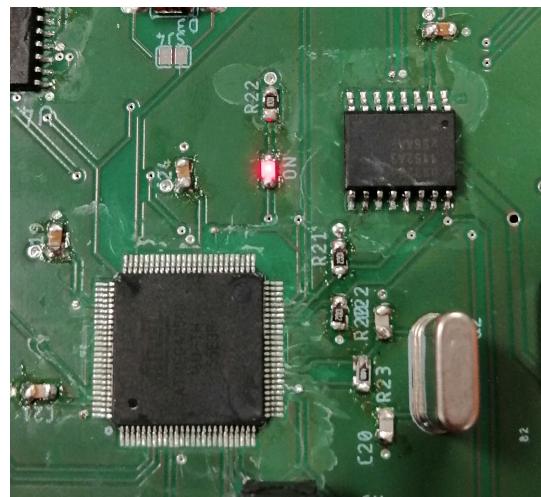
Versorgungstechnisch hat das PCB die Tests bestanden, weshalb nun mit dem Herzstück der Wetterstation, der MCU-Bauteilgruppe, fortgefahren wird.

### Validierung der MCU-Bauteilgruppe

In einem weiteren Schritt wird die MCU mit ihrer Peripherie auf dem PCB angelötet. Um zu testen, ob die MCU funktioniert, wird diese über den ICSP-Header geflasht (über ein AVR-Dragon) und dabei auf einen externen 8MHz-Clock eingestellt. Die LED soll nach dem flashen zu Testzwecken blinken, da die MCU in betrieb ist. Ausserdem soll die LED nicht mehr leuchten, solange der Reset-Button gedrückt wird.

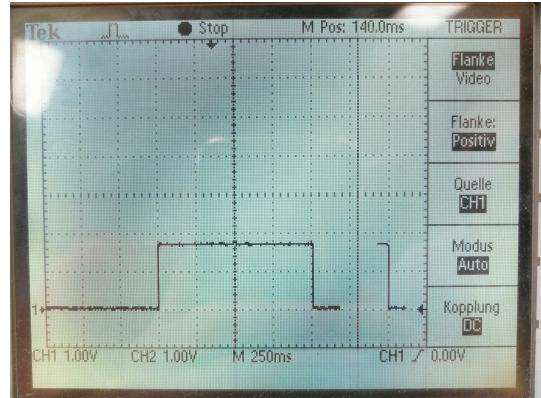


**Abbildung 4.12:** Verbindung zwischen AVR-Dragon und MCU.

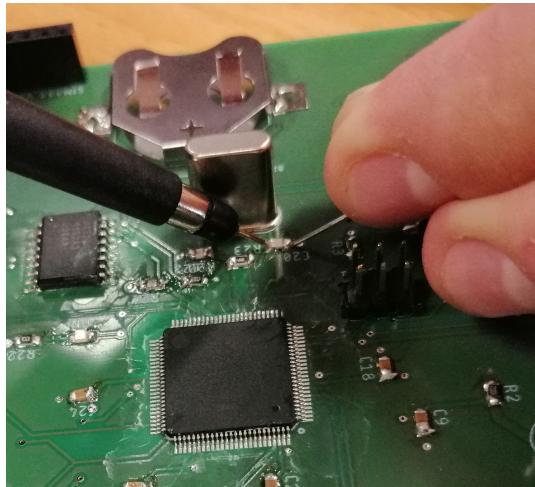


**Abbildung 4.13:** Die leuchtende LED der MCU.

Abbildung 4.12 zeigt, wie die MCU mit einem Programm geflasht wird, wobei ein AVR-Dragon verwendet wird, welcher am ICSP Header angeschlossen ist. Das Blinken der LED kann leider nicht mit Bildern festgehalten werden, jedoch kann mit Hilfe des TDS 210 auch dieser Signalverlauf angezeigt werden, wie in Abbildung 4.14 ersichtlich.

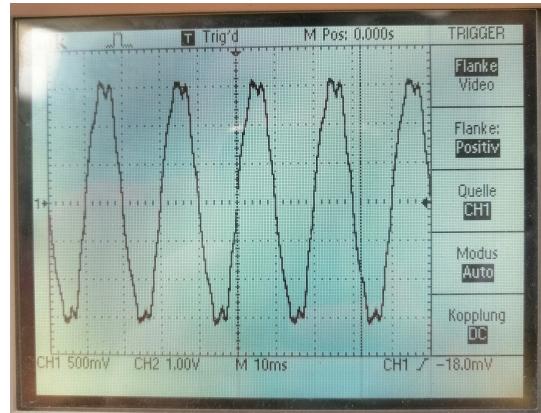


**Abbildung 4.14:** Signalverlauf der blinkenden LED.



**Abbildung 4.15:** Messung am Quarz.

An welcher Stelle der Quarz getestet wurde, ist in Abbildung 4.15 ersichtlich. Das Ergebnis kann in Abbildung 4.16 betrachtet werden. Es ist zu sehen, dass der Quarz mit den gewünschten 8MHz schwingt.

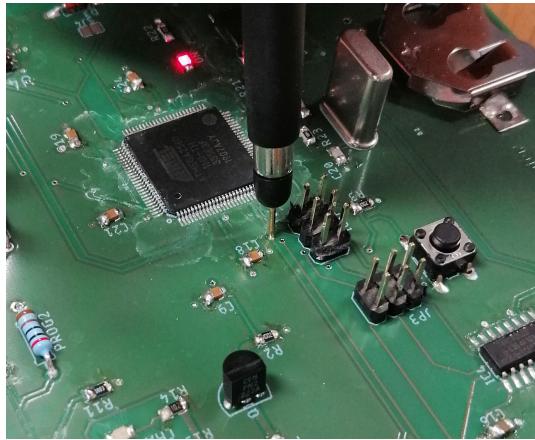


**Abbildung 4.16:** Das oszillierende Signal des Quarzes auf dem TDS 210.

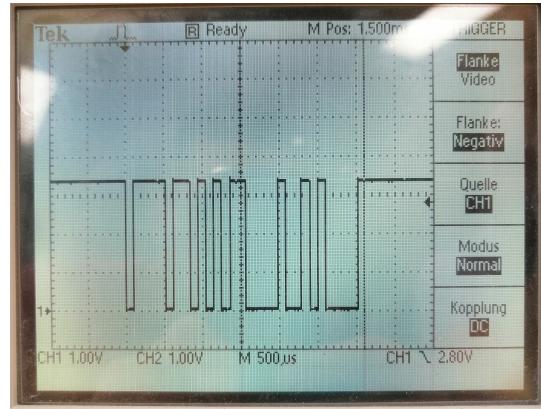
Da die MCU-Bauteilgruppe nun getestet wurde, wird als nächstes die serielle Schnittstelle ( $\mu$ USB-Anschluss) validiert, um weitere Bauteilgruppen mit dessen Hilfe validieren zu können.

### Validierung der seriellen Schnittstelle

Die Bauteilgruppe der seriellen Schnittstelle wird nun auf dem PCB angelötet. Deren Funktion wird mit Hilfe eines Programms auf der MCU und PuTTY (ein Emulator für serielle Schnittstellen auf dem Computer) getestet.



**Abbildung 4.17:** Messung an einem Via der TRX-Signalbahn der MCU.



**Abbildung 4.18:** Signalverlauf der von der MCU gesendeten Zeichen.

Die Abbildungen 4.17 und 4.18 zeigen den Ort des gemessenen Signals und dessen Verlauf. Es ist ersichtlich, dass die gesendeten Zeichen von der MCU aus auf den FT231XS gelangen. Der Ausgang des FT231XS wird über die  $\mu$ USB-Schnittstelle an das Notebook gesendet und dort mit Hilfe von PuTTY auf dem Bildschirm sichtbar.

Daten können über die serielle Schnittstelle übertragen werden, womit im folgenden Abschnitt auf die Validierung der Datenspeicherung eingegangen werden kann.

### Validierung der Datenspeicherung

Bei der Bauteilgruppe der Datenspeicherung wird mit Hilfe der seriellen Schnittstelle die Funktionsweise getestet. Dies geschieht, indem ein File auf der  $\mu$ SD-Karte gespeichert und über die serielle Schnittstelle ausgelesen wird.

Die Datenspeicherung wurde erfolgreich validiert, weshalb im nächsten Abschnitt auf die etwas grössere Bauteilgruppe des SIM808 eingegangen wird.

### Validierung des SIM808

Als nächster grosser Schritt wird die SIM808 mit deren Peripherie implementiert. Es wird getestet, ob die SIM-Karte eine 1.8V Speisung erhält (einfache Spannungsmessung). Nachfolgend wird dann das Senden und Empfangen einer SMS, sowie das ermitteln des Standorts über ein kleines Testprogramm getestet. Die LEDs werden dabei auf ihr korrektes Verhalten hin beobachtet.

#### Fazit SIM808test

Es wurden bereits einige Bauteilgruppen validiert, jedoch bleiben unter anderem die RTC, das Ombrometer, das Anemometer und der Windrichtungsgeber, weshalb im nächsten Abschnitt darauf eingegangen wird.

### Validierung der Peripherie und der RTC

Nun werden die RTC, das Ombrometer und das Anemometer mit Windrichtungsgeber angeschlossen. Die Signale des Ombrometers, des Anemometers und des Windrichtungsgebers werden auf ein möglichst sauberes (unverrauscht) Signal hin getestet und die Flanken des Ombrometers und des Anemometers auf Störungen untersucht. Die Flanke sollte gemäss Abbildung ??, stetig steigend, aussehen, sodass keine Mehrfachdetektionen wegen Störungen auftreten.

#### Fazit RTC Anemo Ombro test

Nun bleiben nur noch der BME280 und der TSL2561 offen für die Validierung, weshalb im nächsten Abschnitt darauf eingegangen wird.

#### Validierung des BME280 und des TSL2561

Zu guter letzt können der BME280 und der TSL2561 angeschlossen werden und über ein Testprogramm auf ihre korrekte Funktionsweise hin überprüft werden.

#### Fazit BME-TSL-test

Die Hardware wurde validiert, es kann gesagt werden, dass alle Tests bestanden wurden. In einem weiteren Schritt muss nun die Firmware auf die MCU geladen und getestet werden, was im nächsten Kapitel erfolgt.

**4.2 Schluss**

### **4.3 Authentizitätserklärung**

Wir, Mischa Knupfer und Andres Minder, versichern, dass dieses Projekt und Fachbericht selbstständig erarbeitet wurden. Alle Quellen und Hilfsmittel aus anderen Werken, die dem Wortlaut oder dem Sinne nach entnommen wurden und zu dieser Arbeit beigetragen haben, sind jeweils kenntlich referenziert.

Aufgrund dessen, dass der Fachbericht als PDF per E-Mail abgegeben wurde, wie vom Auftraggeber/Betreuer gefordert, wird keine Unterschrift gesetzt.

**Ort, Datum:**

**Mitwirkende:**

Brugg/Windisch,  
15. August 2019

## Literatur

- [1] Atmel Corporation. (Juni 2016). AVR Dragon User Guide, Adresse: [http://ww1.microchip.com/downloads/en/devicedoc/atmel-42723-avr-dragon\\_userguide.pdf](http://ww1.microchip.com/downloads/en/devicedoc/atmel-42723-avr-dragon_userguide.pdf) (besucht am 2. Aug. 2019).
- [2] Wikipedia. (19. Mai 2019). Arduino (Plattform), Adresse: [https://de.wikipedia.org/wiki/Arduino\\_\(Plattform\)](https://de.wikipedia.org/wiki/Arduino_(Plattform)) (besucht am 13. Aug. 2019).
- [3] Arduino. (13. Aug. 2019). Frequently Asked Questions, Adresse: [file:///C:/Program%20Files%20\(x86\)/Arduino/reference/www.arduino.cc/en/Main/FAQ.html](file:///C:/Program%20Files%20(x86)/Arduino/reference/www.arduino.cc/en/Main/FAQ.html) (besucht am 13. Aug. 2019).
- [4] FTDI Chip, *Future Technology Devices International Ltd.* *FT231X (USB to FULL HANDSHAKE UART IC)*, Techn. Ber., 2013. Adresse: [https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS\\_FT231X.pdf](https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT231X.pdf) (besucht am 24. Juli 2019).
- [5] Arduino, *Arduino Mega 2560 Datasheet*, Techn. Ber., keine Angabe. Adresse: <https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf> (besucht am 30. Nov. 2018).
- [6] Maxim Integrated, „DS3231 Extremely Accurate I<sup>2</sup>C-Integrated RTC/TCXO/Crystal“, Maxim Integrated Products Inc., Datasheet, 25. Sep. 2018. Adresse: <https://web.wpi.edu/Pubs/E-project/Available/E-project-010908-124414/unrestricted/DS3231-DS3231S.pdf> (besucht am 16. Jan. 2019).
- [7] Amazon. (keine Angabe). Ersatz Sensor Windgeschwindigkeit für Froggit WH1080 WH3080 WH1090, Adresse: [https://www.amazon.de/Ersatz-Sensor-Windgeschwindigkeit-Froggit-WH1080/dp/B00GGM5HEA/ref=pd\\_rhf\\_eetyp\\_p\\_img\\_1?\\_encoding=UTF8&psc=1&refRID=4R5P7RRF8H7RC3A4KAHQ](https://www.amazon.de/Ersatz-Sensor-Windgeschwindigkeit-Froggit-WH1080/dp/B00GGM5HEA/ref=pd_rhf_eetyp_p_img_1?_encoding=UTF8&psc=1&refRID=4R5P7RRF8H7RC3A4KAHQ) (besucht am 30. Nov. 2018).
- [8] Argent Data Systems, „Weather Sensor Assembly p/n 80422“, Argent Data Systems, Bericht, keine Angabe. Adresse: <https://www.sparkfun.com/datasheets/Sensors/Weather/Weather%20Sensor%20Assembly..pdf> (besucht am 17. Jan. 2019).
- [9] Amazon, *MISOL 1 PCS of Spare part for weather station to test the wind direction*, keine Angabe. Adresse: [https://www.amazon.de/gp/product/B00QDMBU80/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o00\\_\\_o00\\_s00?ie=UTF8&psc=1](https://www.amazon.de/gp/product/B00QDMBU80/ref=ppx_yo_dt_b_asin_title_o00__o00_s00?ie=UTF8&psc=1) (besucht am 17. Jan. 2019).
- [10] Bosch Sensortec, „BME280 Combined humidity and pressure sensor“, Bosch, Datasheet, Sep. 2018. Adresse: [https://ae-bst.resource.bosch.com/media/\\_tech/media/datasheets/BST-BME280-DS002.pdf](https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BME280-DS002.pdf) (besucht am 16. Jan. 2019).
- [11] TAOS, *TSL2560, TSL2561 Light-to-digital Converter*, Techn. Ber., 2009. Adresse: <https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf> (besucht am 24. Juli 2019).
- [12] Adafruit, *Adafruit TSL2561 Digital Luminosity/Lux/Light Sensor Breakout*. Adresse: <https://www.adafruit.com/product/439> (besucht am 24. Juli 2019).
- [13] —, *Adafruit BME280 I<sup>2</sup>C or SPI Temperature Humidity Pressure Sensor*. Adresse: <https://www.adafruit.com/product/2652> (besucht am 24. Juli 2019).
- [14] SanDisk. (keine Angabe). SanDisk microSDHC Speicherkarte, Adresse: <https://www.sandisk.de/home/memory-cards/microsd-cards/sandisk-microsd> (besucht am 27. Nov. 2018).
- [15] —, (keine Angabe). SD/SDHC/SDXC Spezifikationen und Kompatibilitäten, Adresse: [https://kb-de.sandisk.com/app/answers/detail/a\\_id/8317/~/sd%2Fsdhc%2Fsdxc-spezifikationen-und-kompatibilit%C3%A4ten](https://kb-de.sandisk.com/app/answers/detail/a_id/8317/~/sd%2Fsdhc%2Fsdxc-spezifikationen-und-kompatibilit%C3%A4ten) (besucht am 27. Nov. 2018).
- [16] SIMCom, *SIM808 Hardware Design V1.00*, Techn. Ber., 2014. Adresse: [https://cdn-shop.adafruit.com/datasheets/SIM808\\_Hardware+Design\\_V1.00.pdf](https://cdn-shop.adafruit.com/datasheets/SIM808_Hardware+Design_V1.00.pdf) (besucht am 24. Juli 2019).

- [17] Microchip, *Stand-Alone System Load Sharing and Li-Ion/Li-Polymer battery Charge Management Controller*, Techn. Ber., 2013. Adresse: <http://www.microchip.com/downloads/en/DeviceDoc/20002090C.pdf> (besucht am 24. Juli 2019).
- [18] Texas Instruments, *LM1117 800-mA Low-Dropout Linear Regulator*, Techn. Ber., 2016. Adresse: <http://www.ti.com/lit/ds/symlink/lm1117.pdf> (besucht am 24. Juli 2019).
- [19] DIODES Incorporated, *74HC125*, Techn. Ber., 2013. Adresse: <https://www.diodes.com/assets/Datasheets/74HC125.pdf> (besucht am 24. Juli 2019).
- [20] Nexpria, *74HC4050 Hex non-inverting HIGH-to-LOW level shifter*, Techn. Ber., 2016. Adresse: <https://www.mouser.ch/datasheet/2/916/74HC4050-1598370.pdf> (besucht am 24. Juli 2019).
- [21] Adafruit. (). Huge 6V 6W Solar panel - 6.0 Watt, Adresse: <https://www.adafruit.com/product/1525> (besucht am 26. Juli 2019).
- [22] ——, (). LICENSE, Adresse: [https://github.com/adafruit/Adafruit\\_BME280\\_Library/blob/master/LICENSE](https://github.com/adafruit/Adafruit_BME280_Library/blob/master/LICENSE) (besucht am 13. Aug. 2019).
- [23] ——, (). LICENSE, Adresse: <https://github.com/adafruit/RTClib/blob/master/LICENSE> (besucht am 13. Aug. 2019).

## Tabellenverzeichnis

1.1	Ziele P5 . . . . .	4
1.2	Ziele P6 . . . . .	5
2.1	Toolchain . . . . .	12
2.2	Lizenzen . . . . .	15
3.1	Technische Daten [5, S.3] . . . . .	24
3.2	Spezifikationen des DS3231 [6]. . . . .	27
3.3	Technische Werte [8] . . . . .	34
3.4	Elektrische Spezifikationen [10] . . . . .	35
3.5	Sezifikationen der Feuchtigkeitsmessung [10] . . . . .	36
3.6	Sezifikationen der Luftdruckmessung [10] . . . . .	36
3.7	Sezifikationen der Temperaturmessung [10] . . . . .	36
3.8	Elektrische Spezifikationen des TSL2561 [11] . . . . .	37
3.9	Elektrische Spezifikationen des SIM808 [16] . . . . .	41
3.10	Übersicht über die verwendeten Bauteile mit ihrem Stromverbrauch. . . . .	43

## Abbildungsverzeichnis

1.1	Konzeptionelle Übersicht . . . . .	6
1.2	Sensoren . . . . .	7
1.3	Real-Time-Clock (RTC) . . . . .	7
1.4	Kommunikationsmodul . . . . .	8
1.5	Energieversorgung . . . . .	9

2.1	AVR Dragon [1] . . . . .	12
2.2	SPI Header [1] . . . . .	13
2.3	Anschluss [1] . . . . .	13
2.4	UML-Diagramm . . . . .	14
3.1	Das noch unbestückte PCB. . . . .	20
3.2	Das unbestückte PCB mit den markierten Bereichen der MCU, der Datenspeicherung, der RTC und der Sensoren. . . . .	21
3.3	Das unbestückte PCB mit den markierten Bereichen des $\mu$ USB-Interfaces, des SIM808 und der Bohrlöcher für die Montage im Gehäuse. . . . .	22
3.4	Das unbestückte PCB mit den markierten Bereichen für die Energieversorgung. .	23
3.5	Arduino Mega Board [5, S.1] . . . . .	24
3.6	Der Mikroprozessor ATMega2560, ohne Board. . . . .	25
3.7	DS3231 mit seinen Anschlüssen [6]. . . . .	26
3.8	Darstellung des Kipplöffelprinzips . . . . .	28
3.9	Selbsterstellter Kipplöffel. . . . .	29
3.10	Selbsterstelltes Gerüst. . . . .	30
3.11	Schaltkreis zur detektion der Kippbewegung. . . . .	31
3.12	Das Ombrometer . . . . .	32
3.13	Anemometer [7] . . . . .	33
3.14	RJ-11 Stecker . . . . .	33
3.15	Beschaltung des Ausgangs des Anemometers. . . . .	33
3.16	Ausgangssignal $V_{out}$ . . . . .	33
3.17	Interne Schaltung [8] . . . . .	34
3.18	Spannungsteiler mit $R = 10k\Omega$ [8] . . . . .	35
3.19	Windrichtungsgeber von MISOL [9] . . . . .	35
3.20	BME280 [10] . . . . .	35
3.21	TSL2561 Breakout Board von Adafruit [12] . . . . .	37
3.22	BME280 Breakout Board von Adafruit [13] . . . . .	38
3.23	16 GB $\mu$ SD-Karte [14] . . . . .	39
3.24	SIM808 von Vorne [16] . . . . .	41
3.25	6V 6W Solarpanel von Adafruit [21] . . . . .	44
3.26	Schematische Darstellung der ChargePump. . . . .	45
3.27	Das Löten eines THT-Bauteils. . . . .	47
3.28	Das Löten eines SMD-Bauteils. . . . .	48
3.29	Das noch unbestückte PCB mit den wichtigsten Dimensionen. . . . .	50

3.30	Gesamtansicht des Designs des Bodens.	52
3.31	Design des Schraubenblocks des Bodens.	52
3.32	Aussparungen für Bauteile, welche aus dem Gehäuse hinausragen.	53
3.33	Seitenwand für den BME280-Extraraum mit Luftschlitzten.	53
3.34	Gesamtansicht des Designs des Deckels.	54
3.35	Schraubenversenkung im Deckel.	54
3.36	Aussparungen für Bauteile, welche aus dem Gehäuse hinausragen und der designte Extraraum für den BME280 mit Luftschlitzten.	55
3.37	Öffnung und Bohrungen für den TSL2561 und den BME280.	55
3.38	Design des Randes vom Boden zum Deckel hin.	56
3.39	Design des Randes vom Deckel zum Boden hin.	56
4.1	Das handliche RMS Multimeter FLUKE 117 mit Sonden.	58
4.2	TDS 210 von Tektronix.	58
4.3	Das verwendete Speisegerät.	58
4.4	Das Zeichen für die Kontinuitätsfunktion des FLUKEs.	59
4.5	Die Rot markierten Stellen zeigen die Endpunkte des Verbindungstests an. Die Orange markierte Stelle zeigt den Startpunkt an.	59
4.6	Testpunkt zur Überprüfung der Eingangsspannung.	60
4.7	Die Eingangsspannung auf dem TDS 210 visualisiert. Es werden 4V angezeigt.	60
4.8	Testpunkt um die Spannungserhöhung der ChargePump zu messen.	60
4.9	Die durch die ChargePump erhöhte Spannung, welche auf dem TDS 210 visualisiert wurde. Es werden etwas mehr als 6V angezeigt.	60
4.10	Testpunkt für die Ausgangsspannung des Linearreglers.	61
4.11	Die durch den Linearregler geregelte Ausgangsspannung. Es werden rund 3.3V angezeigt.	61
4.12	Verbindung zwischen AVR-Dragon und MCU.	62
4.13	Die leuchtende LED der MCU.	62
4.14	Signalverlauf der blinkenden LED.	62
4.15	Messung am Quarz.	62
4.16	Das oszillierende Signal des Quarzes auf dem TDS 210.	62
4.17	Messung an einem Via der TRX-Signalbahn der MCU.	63
4.18	Signalverlauf der von der MCU gesendeten Zeichen.	63
D.1	Komplette Darstellung des Konzepts	76
E.1	Schema der Energieversorgung.	77
E.2	Schema der MCU-Bauteilgruppe.	78

E.3	Schema der $\mu$ SD-Bauteilgruppe. . . . .	79
E.4	Schema des $\mu$ USB-Anschlusses und der Sensorik. . . . .	80
E.5	Schema des SIM808. . . . .	81
F.1	Das PCB-Layout der Wetterstation. . . . .	82



## **Anhang**

## A Lastenheft



Fachhochschule Nordwestschweiz  
Hochschule für Technik

### **Ausschreibung Studierendenprojekt P5/P6 Studiengang Elektro- und Informationstechnik**

<b>Titel:</b>
Wetterstation mit Solar Energie
<b>Betreuer:</b>
Prof. Dr. Taoufik Nouri (Institut für Mobile und Verteilte Systeme)
<b>Auftraggeber:</b>
Prof. Dr. Taoufik Nouri (Institut für Mobile und Verteilte Systeme)
<b>Aufgabenbeschreibung:</b>
<p>Ausgangslage:          Wetterstation sind viele verlangt besonders im Gebiete ohne Strom. Wir schlagen solche Möglichkeit zu realisieren.</p> <p>Zielsetzung:          1. Diese Wetterstation misst Regen, Wind- Geschwindigkeit, -Richtung, Temperatur, Sonnenlicht, Feuchtigkeit, Zeit usw.          2. Sie ist dotiert mit verschiedener Kommunikation Module wie GPS, SIM Karte.          3. Sie ist fern abfragbar durch Handy          4. Sie speichert regelmässig die verschiedenen Parameter (Journal).          5. Sie ist komplett automatisiert z.B. Regenwasser wird automatisch ausgeleert.</p> <p>Schlüsselwörter: Energie, Mikrokontroller, Programmierung, Elektronik</p>

## B Aufgabenstellung



### EIT XX: Wetterstation mit Solar Energie

**Studentin:** Knupfer Mischa (s) [mischa.knupfer@students.fhnw.ch](mailto:mischa.knupfer@students.fhnw.ch)  
 Minder Andres (s) [andres.minder@students.fhnw.ch](mailto:andres.minder@students.fhnw.ch)

**Betreuer:** Prof. Dr. Taoufik Nouri [Taoufik.Nouri@Nouri.ch](mailto:Taoufik.Nouri@Nouri.ch)

**Auftraggeber:** Prof. Dr. Taoufik Nouri [Taoufik.Nouri@Nouri.ch](mailto:Taoufik.Nouri@Nouri.ch)

**Experte:** Patrick Strittmatter [patrick.strittmatter@actemium.ch](mailto:patrick.strittmatter@actemium.ch)

**Dauer:** 18.2.19-16.8.19

**Ausgangslage:**  
 Im Projekt 5 wurde ein Teil dieser Arbeit durchgeführt.  
 S

#### Ziel der Arbeit:

Die Zeit wird nicht von der Wetterstation gemessen. Sie hat ein internes RTC um Zeitstempel zu generieren für die erhobenen Messdaten.

Die folgende Aufgaben sollten u realisiert werden:

1. Diese Wetterstation misst die Regenmenge, Windgeschwindigkeit, Windrichtung, Lufttemperatur, Bestrahlungsstärke der Sonne, Luftfeuchtigkeit usw.
2. Sie ist dotiert mit verschiedener Kommunikation Module wie GPS, GSM.
3. Sie ist fern abfragbar durch Handy via SMS
4. Sie speichert regelmässig die verschiedenen Parameter (Journal).
5. Sie misst selbstständig, ist jedoch nicht wartungsfrei (z.B. Reinigung bei Verschmutzung).

#### Technologien/Fachliche Schwerpunkte/Referenzen

Energie, Mikrokontroller, Programmierung, Elektronik

**Voraussetzungen:** P5, Wetterstation mit Solar Energie (erfüllt)

#### Bemerkungen:

Hardware wie Photovoltaik Zellen, Batteriespeicher usw, Data Acquisition Module werden zuerst in FHNW gesucht, falls keine gefunden wird dann durch die Kandidatin mit Absprache mit der Auftraggeber bestellt/geliefert.

**Aufwand:** P6-2X360 h

**Teamgrösse:** 2 Studenten

## C Lizenz-Texte

### C.1 Adafruit\_BME280

Copyright (c) 2015, Limor Fried & Kevin Townsend for Adafruit Industries All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- \* Neither the name of Adafruit Industries nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS „AS IS“ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. [22]

### C.2 RTClib

MIT License

Copyright (c) 2019 Adafruit Industries

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the „Software“), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE 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. [23]

## D Konzept Posterbild

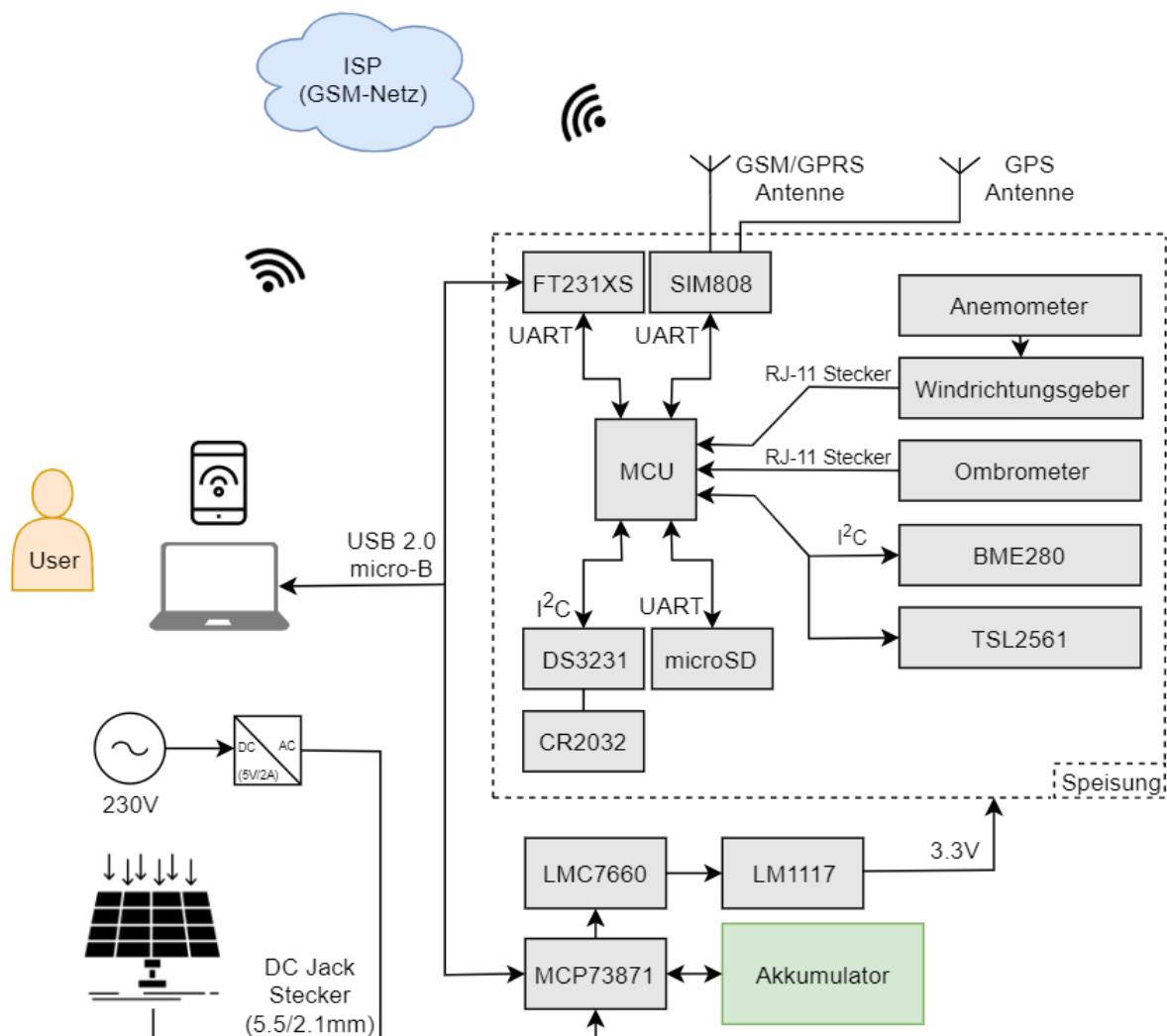
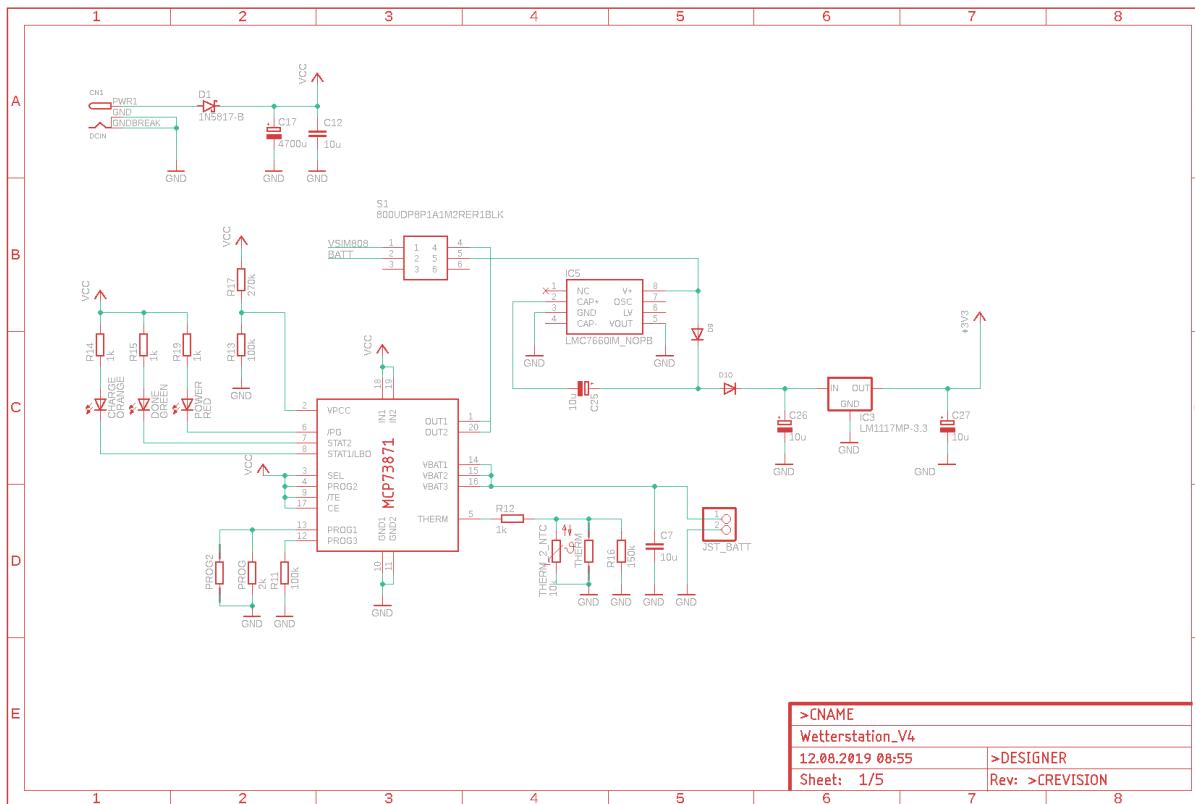


Abbildung D.1: Komplette Darstellung des Konzepts

## E Schema



**Abbildung E.1:** Schema der Energieversorgung.

Abbildung E.1 zeigt das Schema der Energieversorgungs-Bauteilgruppe.

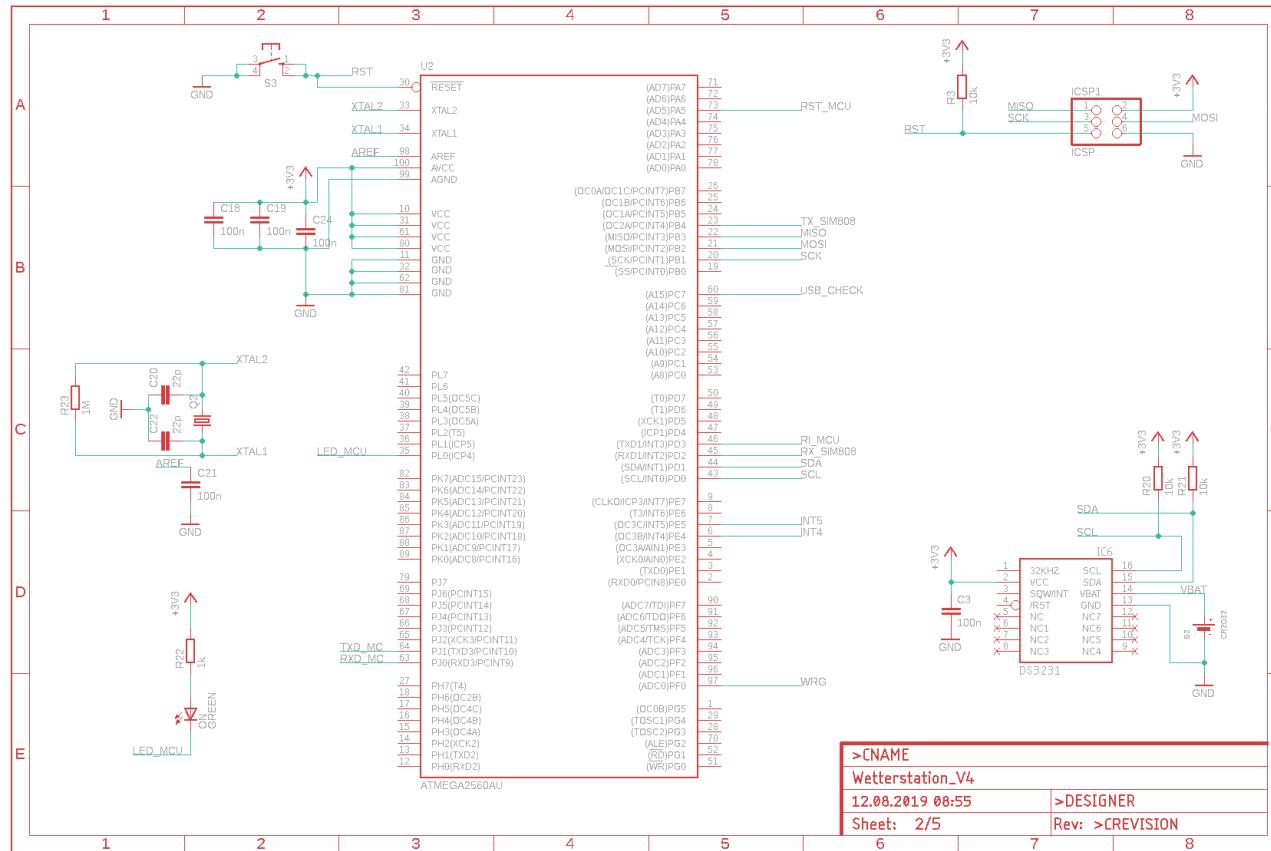
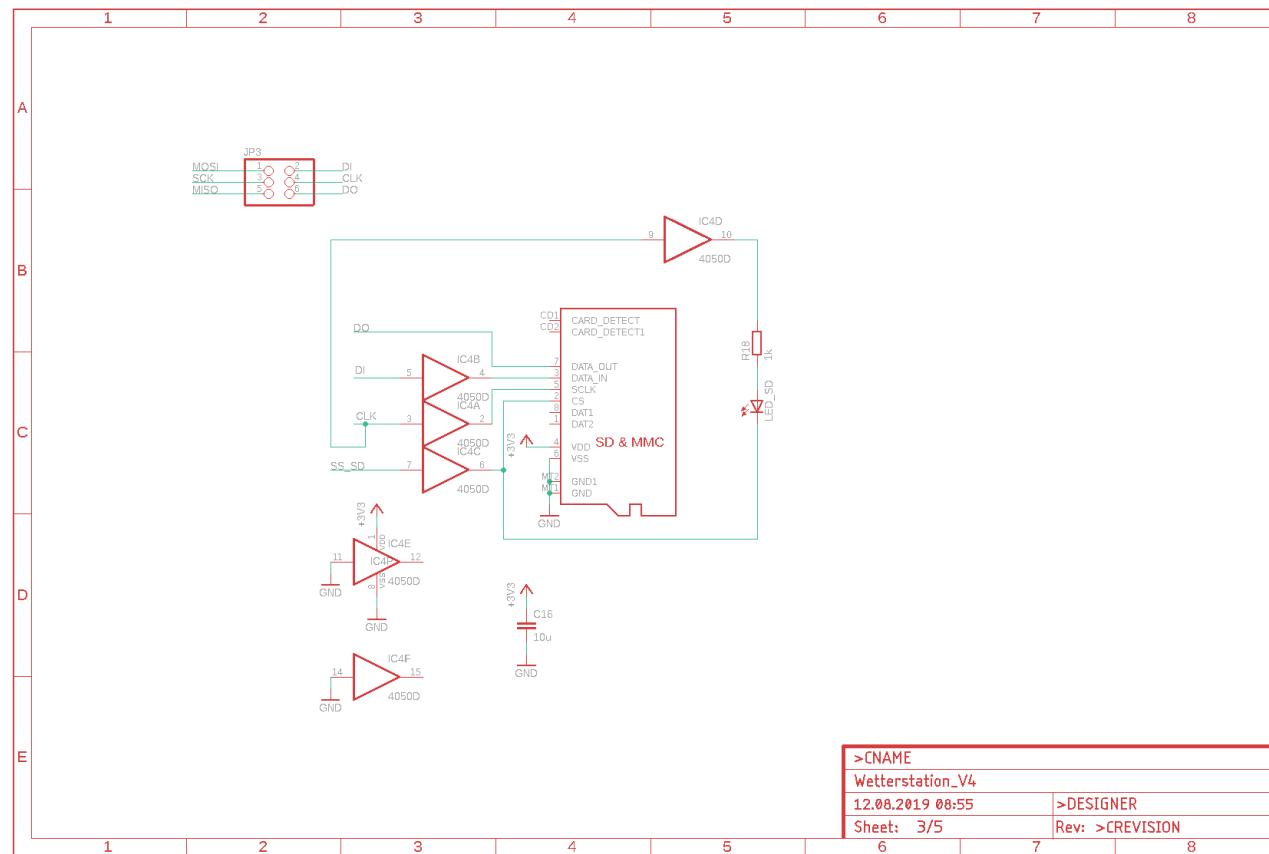


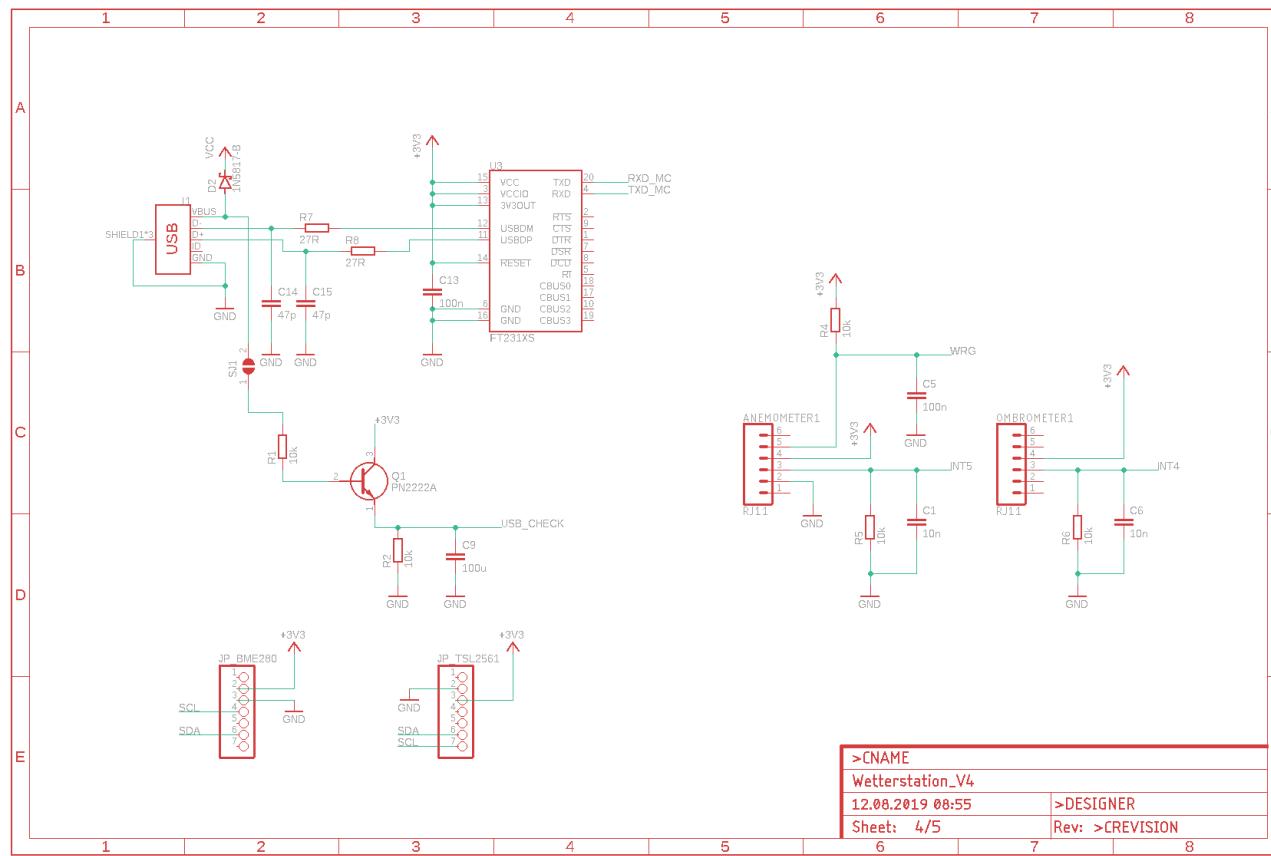
Abbildung E.2: Schema der MCU-Bauteilgruppe.

Abbildung E.2 zeigt das Schema der MCU-Bauteilgruppe.



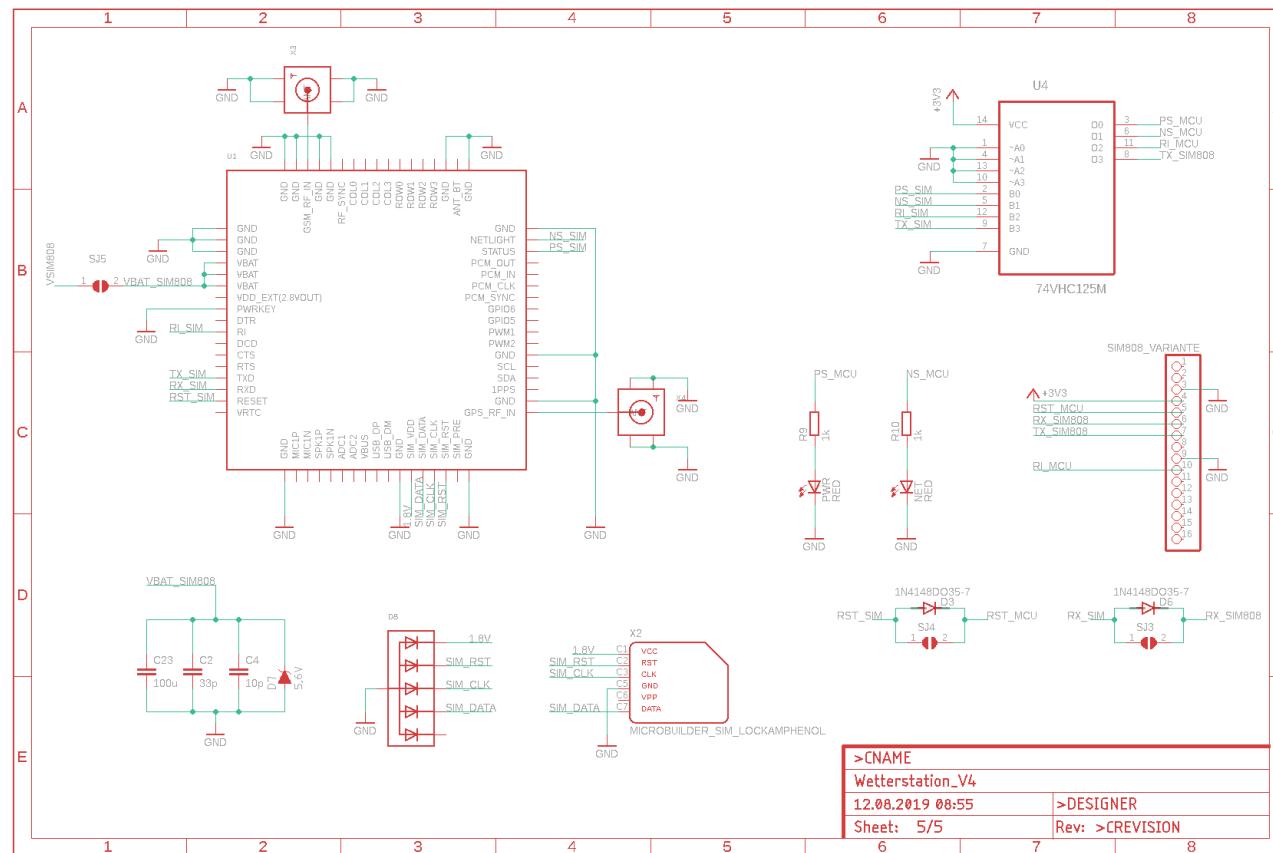
**Abbildung E.3:** Schema der  $\mu$ SD-Bauteilgruppe.

Abbildung E.3 zeigt das Schema der Datenspeicherungs-Bauteilgruppe.



**Abbildung E.4:** Schema des  $\mu$ USB-Anschlusses und der Sensorik.

Abbildung E.4 zeigt das Schema der seriellen Schnittstelle und der Sensorik.



**Abbildung E.5:** Schema des SIM808.

E.5 zeigt das Schema der SIM808-Bauteilgruppe.

## F PCB-Layout

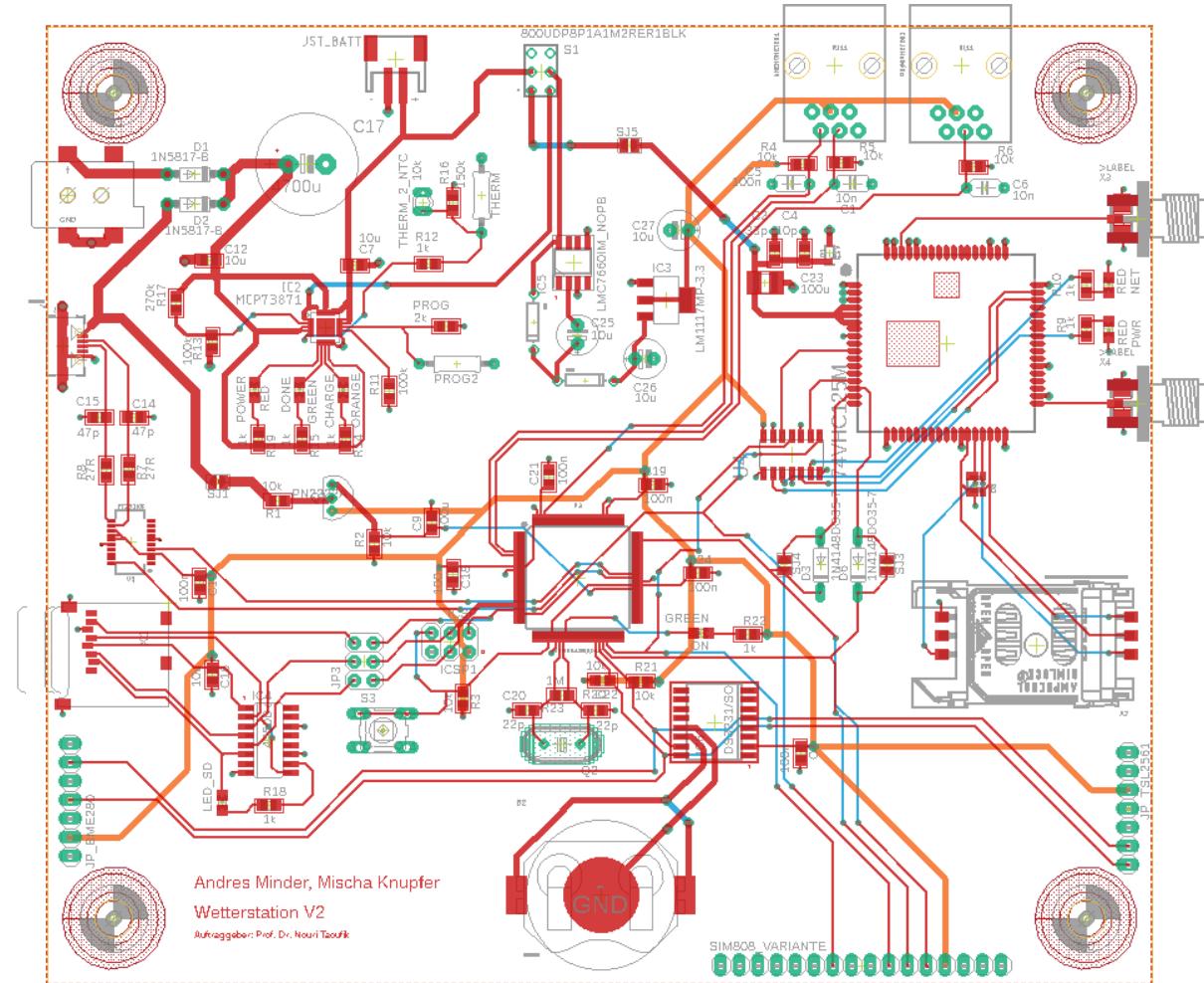


Abbildung F.1: Das PCB-Layout der Wetterstation.

Abbildung F.1 zeigt das PCB-Layout der Wetterstation. Es sind deutlich die verwendeten Bauteile zu erkennen, die Bohrlöcher, Vias und verschiedenfarbige Linien. Rote Linien sind auf dem obersten Layer (Layer 1), welcher hauptsächlich für Signale verwendet wird. Der zweitoberste Layer (Layer 2) ist mit einer Groundfläche (0V) ausgefüllt und wird über Vias angesteuert. Der zweitunterste Layer (Layer 3) dient dem 3.3V-Netz um die Bauteilgruppen zu speisen (Orange). Der unterste Layer (Layer 4) ist ebenfalls für Signale reserviert (Blau), um anderen Leiterbahnen besser Ausweichen zu können. Etwaige Wechsel zwischen Layer 1 und Layer 4 werden mit durchgehenden Vias gemacht. Wechsel zwischen Layer 1 und Layer 2 bzw. Layer 3 werden mit blind Vias gemacht. Die verschiedenen Layer wurden alle mit Groundflächen ausgefüllt, damit die Herstellung des PCBs dem Hersteller vereinfacht wird (weniger ätzen).