



TECHNIKERARBEIT

# **ENTWICKLUNG EINER SMART HOME ZENTRALE AUF BASIS EINES RASPBERRYPI**

Felix KUSCHEL & Manuel STARZ

Betreut durch  
Matthias KOHLER

25. März 2021

# Inhaltsverzeichnis

<b>1 Vorwort</b>	<b>4</b>
1.1 Einleitung . . . . .	4
1.2 Projektrahmen . . . . .	4
1.3 Aufgabenstellung . . . . .	4
1.4 Zusatzinformationen . . . . .	4
1.5 Definition Smart Home Zentrale . . . . .	5
1.6 Datenschutzhinweis . . . . .	5
<b>2 Zielsetzung</b>	<b>6</b>
2.1 Konzeption . . . . .	6
2.2 Anforderungen und gewünschte Features . . . . .	6
<b>3 Herangehensweise</b>	<b>8</b>
3.1 Hardware . . . . .	8
3.2 Software . . . . .	8
<b>4 Zeitplan</b>	<b>9</b>
<b>5 Komplettübersicht</b>	<b>10</b>
5.1 Das „fertige“ Produkt . . . . .	10
5.2 Kostenaufstellung . . . . .	11
5.2.1 Beschaffungskosten . . . . .	11
5.2.2 Kostenberechnung für die 3D-Druckteile . . . . .	12
<b>6 Hardware</b>	<b>13</b>
6.1 Raspberry Pi 4 . . . . .	13
6.2 Aufbau des Prototypen . . . . .	14
6.3 Gehäuse . . . . .	15
6.3.1 Erster Test des Gehäuses . . . . .	15
6.3.2 Modellentwicklung am Objekt . . . . .	16
6.3.3 Herstellung des Gehäuses . . . . .	22
6.4 Erstellung des RPi-HATs . . . . .	24
<b>7 Software</b>	<b>27</b>
7.1 Überblick . . . . .	27
7.2 MQTT-Broker . . . . .	27
7.3 Zigbee . . . . .	27
7.4 HomeAssistant . . . . .	27
7.5 MyCroft . . . . .	27
<b>8 Epilog &amp; Fazit</b>	<b>28</b>
8.0.1 L <sup>A</sup> T <sub>E</sub> Xvs Word . . . . .	28
<b>9 Quellen</b>	<b>29</b>
9.1 Dokumentationsquellen . . . . .	29
9.2 Verwendete Software . . . . .	29
9.3 Verwendete Hardware . . . . .	30

---

<b>10 Anhang</b>	<b>31</b>
10.1 Lastenheft . . . . .	31
10.2 Installationsskript . . . . .	33
10.3 Gehäuse-Zeichnungen . . . . .	33
10.4 Raspberry Pi HAT E-Schema . . . . .	39
10.5 Hardware-Dokumentationen . . . . .	40

## Erklärung

Wir versichern, dass die vorliegende Abschlussarbeit von uns selbstständig angefertigt und nur die angegebenen Hilfsmittel benutzt wurden. An Stellen, die dem Wortlaut oder dem Sinne nach anderen Werken entnommen sind, haben wir dies durch die Angabe der Quellen kenntlich gemacht.

---

Datum, Ort

Felix Kuschel

---

Datum, Ort

Manuel Starz

# 1 Vorwort

## 1.1 Einleitung

Bei dieser Ausarbeitung handelt es sich um die Abschlussarbeit zur Weiterbildung zum staatlich geprüften Techniker in der Fachrichtung Informationstechnik. Diese Arbeit basiert auf dem in den zwei Jahren erlernten Stoffs sowie selbst erarbeiteten Kenntnissen und dient zur Feststellung des Erreichen des Fortbildungszieles.

## 1.2 Projektrahmen

Das Projekt zur Erstellung einer Smart Home Zentrale wurde von uns, Felix Kuschel und Manuel Starz, durchgeführt. Der Projektzeitraum war vom 1. September 2020 bis zum 30.03.2021 angesetzt. Es stand zur Umsetzung des Projekts ein untermittelfreier Tag pro Woche zur Verfügung.

Des weiteren wurden die in dem Zeitraum zur Verfügung stehenden Ferien zur Umsetzung des Projekts genutzt. Die Projektbetreuung erfolgten seitens der Schule durch Herr Matthias Kohler. Da das Projekt nicht in Zusammenarbeit mit einem Unternehmen durchgeführt wurde, gibt es keine weiteren Betreuer. Die Materialkosten für die im Projekt genutzte Hardware wurde von den uns selbst getragen.

## 1.3 Aufgabenstellung

Das Projekt ist aus dem Zusammenschluss der Abschlussarbeitsideen von uns, Felix Kuschel und Manuel Starz, entstanden und wurde mit Rücksprache mit dem betreuenden Lehrer entwickelt.

Durch die große Verfügbarkeit von Smart Home Geräten und den zahlreichen Standards der Anbieter entschlossen wir uns, eine einfache und leicht zu replizierende Lösung zu entwickeln, die Smart Home Geräte mehrerer Hersteller miteinander verknüpft und so die Notwendigkeit mehrerer verschiedener sogenannter Hubs zu eliminieren.

Des weiteren soll das Gerät noch über einen Touchscreen steuerbar sein und die Werte der verbundenen Smart Home Geräte anzeigen. Dies umfasst unter anderem den Status von Leuchtmitteln, die Werte von Thermostaten sowie den Zustand von Tür- und Fensterkontakten. Die genaue Aufgabenstellung kann dem abgegebenen Lastenheft im Anhang entnommen werden.

## 1.4 Zusatzinformationen

Die Rohdaten des Projekts wurden der Einfachheit in einem GIT-Projekt zusammengefasst. Dadurch konnten die Durchführenden unabhängig voneinander an dem Projekt und der Dokumentation arbeiten. Der Link zu dem Projekt lautet:

<https://github.com/Pharias/TAR>

Ursprünglich wurde für die Verwendung des schulinternen GIT verwendet. Dies stand zum Zeitpunkt der Erstellung der Dokumentation allerdings nicht zur Verfügung<sup>1</sup>, weshalb eine Alternative genutzt wurde.

<sup>1</sup> War lt. Herr Kohler nicht verfügbar aufgrund Sicherheitsrisiken

## 1.5 Definition Smart Home Zentrale



(a) Amazon Alexa Echo Show 8

(b) Google Nest Hub

(c) Glancr Smart Mirror

Abbildung 1: Smart Home Zentralen

**Smart Home Zentralen**, auch **Smart Hubs** oder **Smart Mirrors** genannt, sind Geräte, die als zentraler Knotenpunkt in einem Smart Home Netzwerk sitzen und dort Informationen verarbeiten, weiterleiten und darstellen können.

Diese Geräte werden von den meisten Herstellern mit und ohne Bildschirm geliefert, um entweder ein neues Smart Home aufzubauen oder ein bestehendes Smart Home zu erweitern.

Bei einem Smart-Home-Hub handelt es sich um eine schlaue Zentrale, durch die all deine intelligenten Geräte miteinander vernetzt werden – und dadurch erst wirklich ihren gesamten Leistungsumfang ausschöpfen.<sup>2</sup>

Als Smart Home Zentrale können auch Software-Lösungen gezählt werden, die mit den im Netz befindlichen Smart Hubs kommunizieren und die Informationen mit Hilfe eines Web-Interfaces oder einer Smartphone-Anbindung darstellen und steuern können. Beispiele hierfür sind homeassistant.io, openHAB und Google Home.



(a) Home Assistant



(b) Google Home



(c) openHAB

Abbildung 2: Smart Home Software

## 1.6 Datenschutzhinweis

Aus Datenschutzgründen sind lokale IP-Adressen und lokale Domänennamen innerhalb der Dokumentation unkenntlich gemacht.

<sup>2</sup>Li (2017): Was ist ein Smart-Home-Hub? Alles über die intelligente Zentrale

## 2 Zielsetzung

Als Ziel für das Projekt war ein funktionsfähiges Smart Home Hub mit Zigbee-Anbindung auf Basis eines Raspberry Pi 4 geplant. Im Laufe des Projekts kam dann noch eine Erweiterungskarte für den Raspberry Pi, ein sog. Raspberry Pi HAT, hinzu, welcher aber aufgrund der vorliegenden Lage mit der COVID-19-Pandemie und den damit verbundenen Liefer- und Zollschwierigkeiten verworfen wurde. Die Planung des HAT ist daher nur rudimentär und nicht vollständig, kann aber bei Bedarf nach Beenden des Projekts durchgeführt werden. Das Smart Home Hub in seiner Grundfunktion soll in der Lage sein, die mit ihm verbundenen Geräte über den Zigbee-Standard anzusteuern. Darüber sollte die Möglichkeit einer Erweiterung mit einem Sprachassistenten gegeben sein.

### 2.1 Konzeption

Zu Beginn des Projekts haben wir Nachforschung angestellt und uns bereits vorhandene Open-Source-Lösungen im Bereich Smart Home angeschaut. Dabei sind wir neben dem Smart Mirror GLANCR auch auf die Gesamtlösung homeassistant.io sowie openHAB gestoßen. Darauf hin haben wir ein Grundkonzept in unserem Lastenheft zusammengefasst und dieses in Absprache mit unserem Betreuungslehrer, Herr Kohler, ausgearbeitet.

Nach Abgabe des Lastenhefts haben wir uns dann an die Beschaffung der unserer Meinung nach nötigen Komponenten für das Projekt gemacht.

### 2.2 Anforderungen und gewünschte Features

Die Anforderungen an das Projekt lauteten demnach wie folgt:

- Anbindung von ZigBee-fähigen Endgeräten
- Steuerung der angebundenen Endgeräte
- Übermittlung der Zustände der angebundenen Geräte an z.B. ein Smartphone

Diese Anforderungen lassen sich mit einem Raspberry Pi und einem Zigbee-USB-Stick realisieren. Darüber hinaus waren von unserer Seite noch folgende Features gewünscht:

- Ein- und Ausgabe über einen Touch-Bildschirm
- Einbindung eines Sprachassistenten zur Steuerung der eingebundenen Endgeräte

Nach Fortschritt des Projekts kam bei einer Rücksprache mit unserem Projektbetreuer die Idee auf, einen Raspberry Pi Hat speziell für das Projekt zu entwickeln. Dieser sollte die Hardware des Projekts falls möglich auf einer Platine vereinen, die dann auf den Raspberry Pi aufgesteckt werden konnte.

Diese Erweiterungsplatine sollte folgende Eigenschaften besitzen:

- ZigBee-Controller und Antenne

- NFC-Controller und Antenne
- RGB-LED zur Statusanzeige
- Anschluss für Lüfter
- Sensoren für:
  - Luftfeuchtigkeit
  - Temperatur
  - Luftdruck
- Pins zum Anschluss an Versuchsaufbau für Laborgebrauch

Die Erweiterungsplatine wurde aber wie zuvor aufgrund der aktuellen Pandemie-Situation und den damit verbundenen Beschaffungsschwierigkeiten verworfen. Darauf wurde dann klar, dass eine Erstellung eines Installationsskripts für den Raspberry Pi eine sinnvolle Ergänzung der Projektarbeit wäre. Zusätzlich haben wir ein Gehäuse für die Hardware geplant, um das Endprodukt so wertiger gestalten zu können.

## 3 Herangehensweise

Nach Festlegung der Anforderungen haben wir uns dann an die Beschaffung und der benötigten Materialien und die Einrichtung der Hard- und Software gemacht. Hierfür haben wir zum Teil bereits vorhandene Hardware, z.B. den Raspberry Pi 4 mit weiteren Komponenten wie dem Touch-Bildschirm und den Lautsprechern sowie dem Mikrofon ergänzt. Damit ist die Umsetzung des Projekts in zwei Teile geteilt:

### 3.1 Hardware

Bei der Entwicklung der Hardware haben wir für unser Projekt intern einige Rahmenbedingungen festgelegt:

- Das Projekt sollte nach Möglichkeit von durchschnittlichen Bastlern durchgeführt werden
- Zur Fertigstellung des Projekts sollte keine weiteren Werkzeuge als die bei uns persönlich vorhandenen Werkzeuge vorhanden sein
- Die Grundplattform sollte der 2020 herausgebrachte Raspberry Pi 4 8GB sein<sup>3</sup>
- Das Gerät sollte möglichst von vielen Herstellern vertriebenen Zigbee-Endgeräte verwalten können.

Die Beschaffung der Teile lief dank Online-Versandhandel relativ problemlos. Beim Zusammenbau haben wir den Prototyp lediglich mit dem Bildschirm, dem Pi und dem Zigbee-USB-Stick aufgebaut (vgl. 6.2). Die beiden Standfüße stammen vom Hersteller des Bildschirms Sunfounder <sup>4</sup> und wurden PETG auf dem Ender 3 von Manuel Starz gedruckt. Danach wurde die Softwareseite von Felix Kuschel bearbeitet, der die Software auf dem Raspberry OS verwendet hatte (vgl. 3.2 & 7).

Nachdem der Prototyp soweit funktionsfähig war, haben wir uns daran gemacht, das Präsentationsmodell zu entwerfen. Dabei ging es uns vornehmlich um die Unterbringung der geplanten Komponenten in einem simplen Gehäuse. Hierfür waren einige Verlängerungskabel nötig, um Stromzufuhr, Netzwerk und die Antenne des Zigbee-Sticks nach außen zu leiten (vgl. 6.3.2 & 5.1). Das Gehäuse wurde dann nach dem Druck mit dem Bildschirm verklebt und die einzelnen Komponenten verbaut (vgl. 6.3.3). Nachdem das Gerät soweit fertig war, hat Felix Kuschel einen Versuchsaufbau zur Präsentation aufgebaut, der die Funktionsweise des Geräts in einem „typischen“ Heimnetz darstellen soll.

### 3.2 Software

---

<sup>3</sup>Wikipedia: Raspberry Pi - Generations

<sup>4</sup>Sunfounder: 10.1 Inch Touch Screen for Raspberry Pi(NEW) - 3D-printed Touch Screen Support

## 4 Zeitplan

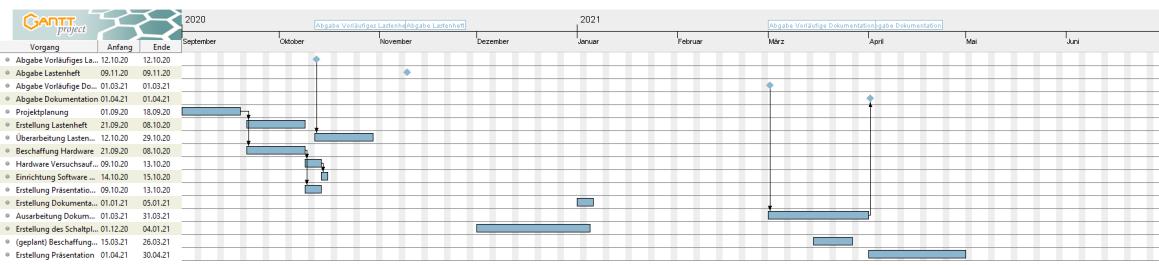


Abbildung 3: gantt-Diagramm des Projektablaufs

## 5 Komplettübersicht

### 5.1 Das „fertige“ Produkt



Abbildung 4: Smart Home Zentrale

Dies ist das Ergebnis der Technikerarbeit von Felix Kuschel und Manuel Starz. Eine Smart Home Zentrale auf Basis von Homeassistant mit einer ZigBee-Antenne, einem 10.1" großen Touchbildschirm und der Möglichkeit, per LAN und WLAN mit dem Heimnetz in Verbindung zu treten.

Das Gehäuse ist für die Wand-Montage konzipiert, die Nutzung eines Gehäuse um die Smart Home Zentrale auf einer Kommode oder in einem Regal zu platzieren (mit einem leichten Winkel) ist ebenfalls möglich. (vgl. 5)

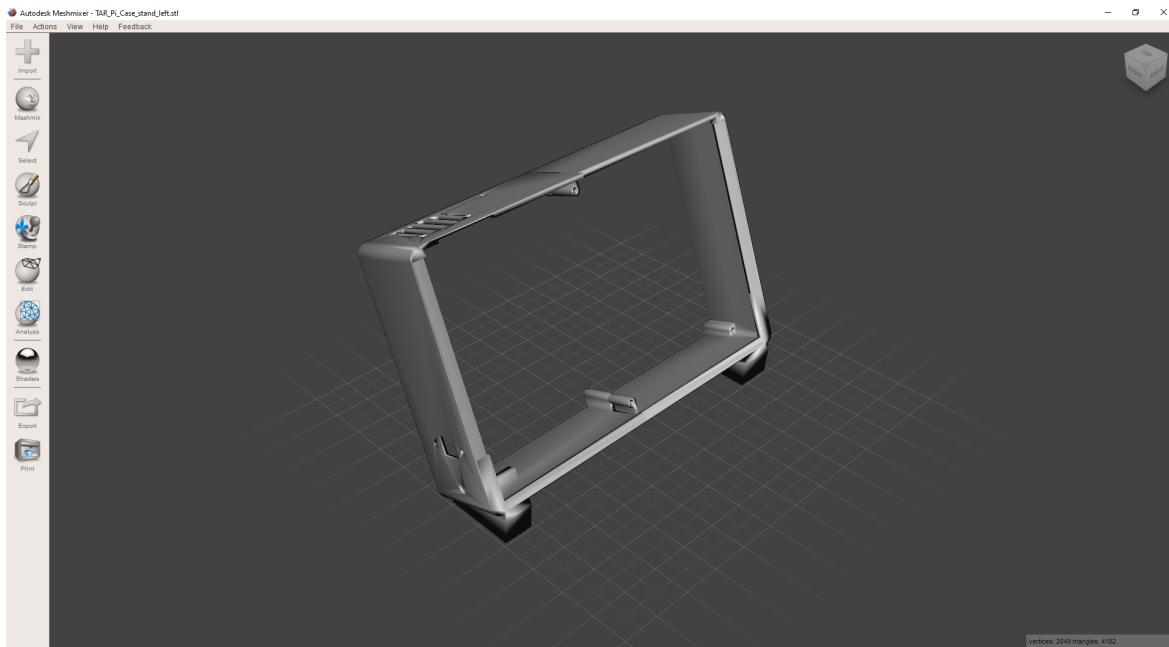


Abbildung 5: Gehäuse mit Füßen (Render in Meshmixer)

## 5.2 Kostenaufstellung

Nachfolgen haben wir die Kosten des Projekts aufgelistet.

### 5.2.1 Beschaffungskosten

Produkt	Menge	Kosten/Stk	Kosten/Gesamt
Raspberry Pi 4 Modell B 8GB	1	87,22€	87,22€
SanDisk Extreme microSD 128 GB	1	19,99€	19,99€
SUNFOUNDER RPi 10.1" Touch Display	1	129,99€	129,99€
Noctua NF-A4x10 5v	1	12,90€	12,90€
ITSTUFF CC2531 Zigbee USB-Stick	1	14,90€	14,90€
ITSTUFF Alu-Kühlkörper Set RPi 4	1	4,91€	4,91€
Eightwood SMA Verlängerung	1	7,99€	7,99€
VCE 5,5mm Stecker & Buchse	1	8,29€	8,29€
dasFilament PTGE schwarz 1,75mm	1	24,95€ <sup>5</sup>	11,61€ <sup>6</sup>
USB-Verlängerung	1	9,99€	9,99€
<b>Gesamtkosten</b>			<b>XXX€</b>

<sup>5</sup>Kosten der Spule (800g).

<sup>6</sup>Kosten verwendetes Material für Gehäusedruck

### 5.2.2 Kostenberechnung für die 3D-Druckteile

Die Kosten lassen mit folgender Formel darstellen:

$$K_{gesamt} = (K_{Material/g} \cdot M_{Material}) + (T_{Druck} \cdot K_{kWh} \cdot V_{\emptyset/h})$$

Die Kosten der 3D-Druckteile lassen sich in zwei Einzelpositionen, Material und Energie. Die Materialkosten lassen sich leicht berechnen, da wir die Kosten der Spule Filament mit 24,95€ und einem Gewicht von 800g Filament auf der Spule. Dies bedeutet einen Preis von 0,02(74...)€ ( $K_{Material/g}$ ) pro Gramm verwendetem Filament. Der Druck der Gehäuseteile verbraucht 423g Material ( $M_{Material}$ ), dass heißt, dass die Materialkosten sich auf 11,61€ belaufen.

Die Energiekosten zu berechnen ist etwas komplexer. Der 3D-Drucker verbraucht circa 120W pro Stunde( $V_{\emptyset/h}$ )<sup>7</sup>, was bei einem Strompreis von 22,3€ct / kWh ( $K_{kWh}$ ) bedeuten würde, dass wir pro Stunde etwa 2,676€ct Stromkosten haben. Bei einer Gesamtdruckzeit von 36 Stunden und 36 Minuten( $T_{Druck}$ ) ergibt sich also Stromkosten von 97,9416€ct.

Zählt man nun die Kosten für Material und Energie zusammen, ergibt sich ein Fertigungspreis von 12,59€ ( $K_{gesamt}$ ). Dabei ist die Veredelung (Schleifen & Lackieren) noch nicht mit einkalkuliert.

---

<sup>7</sup>Robert (2019): Antwort auf Creality Ender 3 printer power consumption? - 3dprinting Stack Exchange

## 6 Hardware

### 6.1 Raspberry Pi 4

Als Basis für die Smart Home Zentrale haben wir einen Raspberry Pi in der Version 4 mit 8 GB gewählt, um als von anderen Servern unabhängige Plattform zu agieren. Der Raspberry Pi 4 ist mit einem ARM Cortex-A72 Prozessor ausgestattet, der über 4 Kerne verfügt. Darüber hinaus verfügt der Raspberry Pi 4 über einen Gigabit-Netzanschluss. Darüber hinaus ist der Raspberry Pi in der Bastlerszene weit verbreitet und dient bei anspruchsvoller Projekten als Kern

Der Raspberry Pi hat sich seit der ersten Veröffentlichung Anfang 2012 weltweit schon millionenfach verkauft und erfreut sich immer noch größter Beliebtheit. Denn viele Raspberry Pi User haben nicht nur einen Einplatinen-Computer zu Hause, sondern teils 4-5 Stück. Der eine fungiert als HD-Mediacenter mit externer Festplatte für das heimische Kino oder als Internetradio mit Display, der nächste als Webcam-Server für die Kameraüberwachung mit Livestream auf das Handy, dann noch einer für die Hausautomatisierung wie bspw. die Heizungs- oder Lichtsteuerung und noch einer als einfacher WLAN-Druckerserver oder als Mini-Computer zum allgemeinen Surfen im Internet, um ein paar wenige Anwendungsszenarien zu nennen. Sie merken, der kleine „Tausendsassa“ kann nicht nur viel, sondern ist zudem auch noch extrem günstig und eben das macht den Reiz aus. Lediglich den Hang zum Programmieren sollten Sie mitbringen und selbst nicht mal das, denn Sie können auch einfach nach Anleitung aus Foren oder Büchern nachprogrammieren, oder ganz bequem ein fertiges Image auf den Raspberry Pi installieren - trauen Sie sich!<sup>8</sup>

---

<sup>8</sup>reichelt.de: Produktbeschreibung des Raspberry Pi 4

## 6.2 Aufbau des Prototypen

Der Prototyp war eine Kombination aus dem verwendeten Touchscreen sowie dem Raspberry Pi 4 und dem Zigbee-Stick. Dieser Prototyp diente als Entwicklungsplattform für die Software.

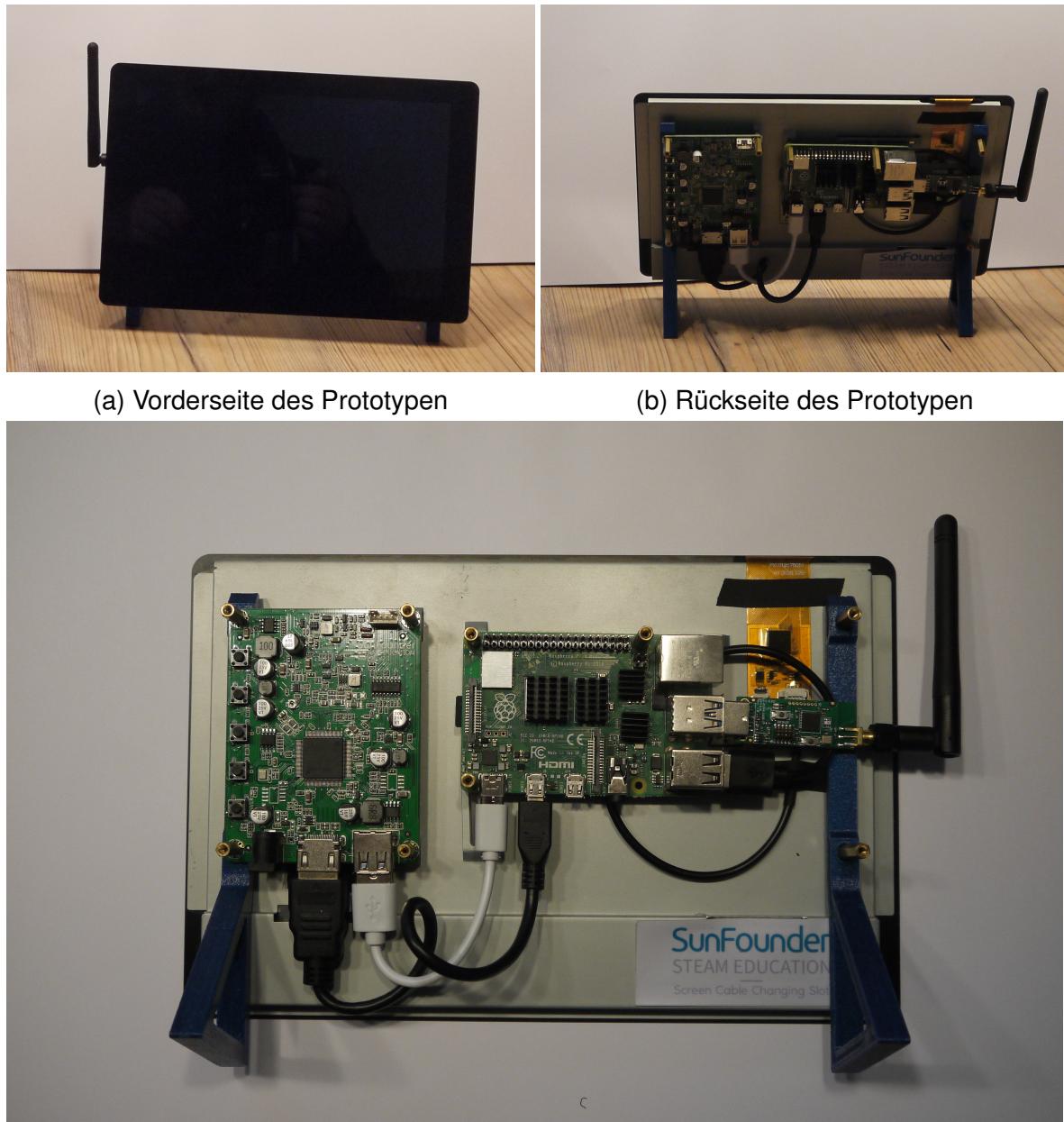


Abbildung 6: Prototyp der Smart Home Zentrale

## 6.3 Gehäuse

### 6.3.1 Erster Test des Gehäuses

Zur Erstellung des Gehäuses haben wir die Maße des Bildschirms als Anhaltspunkt genommen. Das Gehäuse befand sich zu diesem Zeitpunkt bei Felix Kuschel, der die Messungen vornahm. Der Bildschirm hatte an der Hinterseite eine Erhebung, weshalb er diese ebenfalls ausgemessen hatte. Die Maße beliefen sich dann auf:

- 255,5 mm Breite
- 167 mm Höhe
- Kantenradius 10 mm

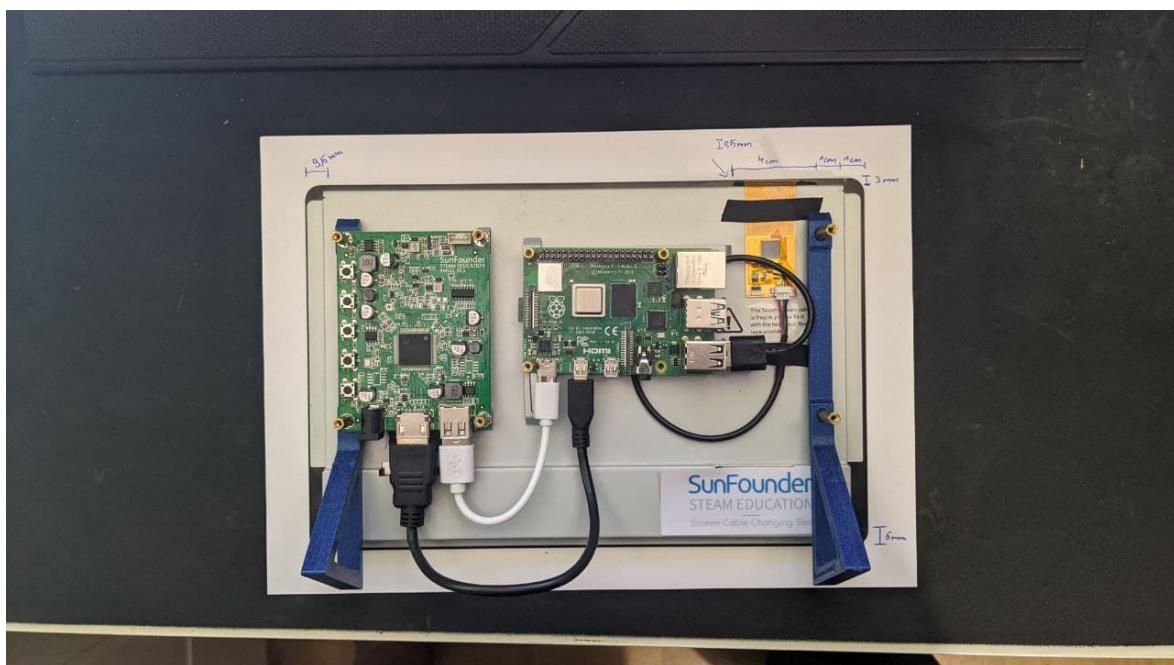


Abbildung 7: Abmessungen Bildschirm Rückseite

Die Maße an der Rückseite übermittelte er Manuel Starz als Bild. Anhand dieser Maße hat Manuel Starz dann in Fusion 360 eine Grundplanzeichnung erstellt und ein 3D-Modell gefertigt.

Nach Überprüfung der Maße mussten wir dann allerdings feststellen, dass der zur Verfügung stehende 3D-Drucker, ein Ender 3 Pro der Firma Creality3D, ein maximales Druckvolumen von 235x235x220mm besitzt und somit das Gehäuse nicht wie ursprünglich geplant aus einem Stück sondern in mehreren Teilen gedruckt werden musste. Hierfür war eine Änderung der Konstruktion von Nöten. Das Gehäuse besteht nun aus vier Teilen, zwei bilden jeweils die Seitenwände während zwei den Deckel des Gehäuses bilden. Die Teile werden mit langen M3 Senkkopfschrauben verbunden, die zusätzlich als Verschluss des Gehäuses dient.

Um die Genauigkeit der Konstruktion zu testen, hat Manuel Starz eine 5 Millimeter hohe Testschablone ausgedruckt, die an den Bildschirm angelegt werden kann. Diese entstand mit Hilfe des Slicers CURA (vgl. Abbildung 8), in dem die Modelle der beiden Seitenteile so angeordnet wurden, dass lediglich 5 Millimeter des Teils im druckbaren Bereich des Druckers verblieben (vgl. Abbildung 9).

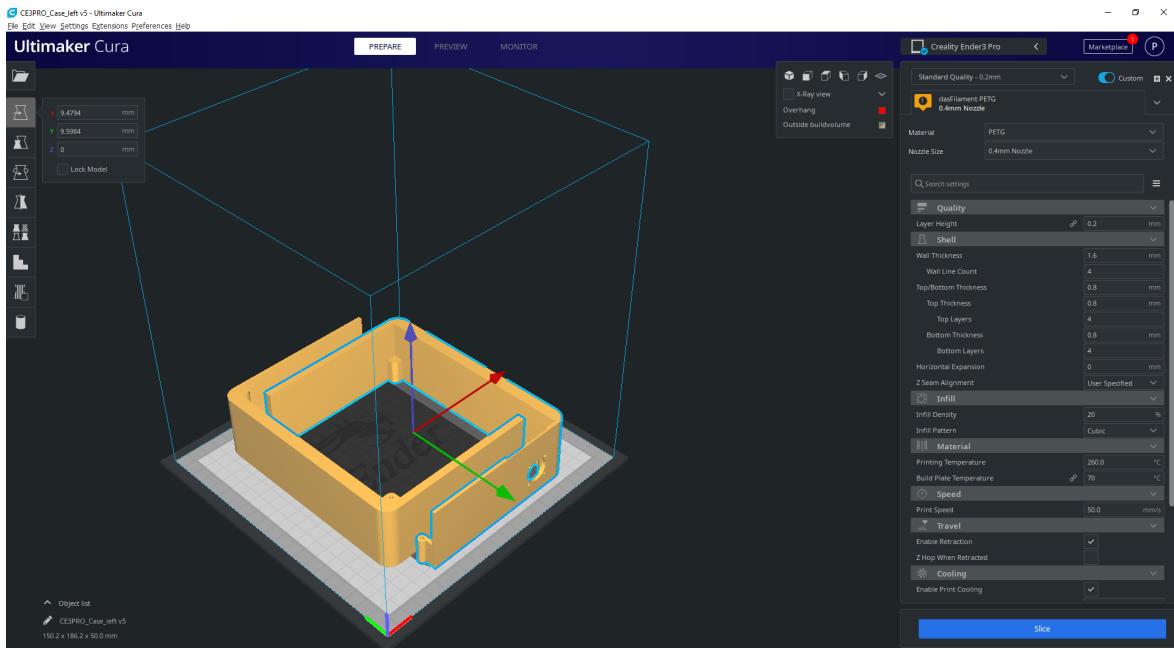


Abbildung 8: Platzierung der beiden Gehäusewände in CURA

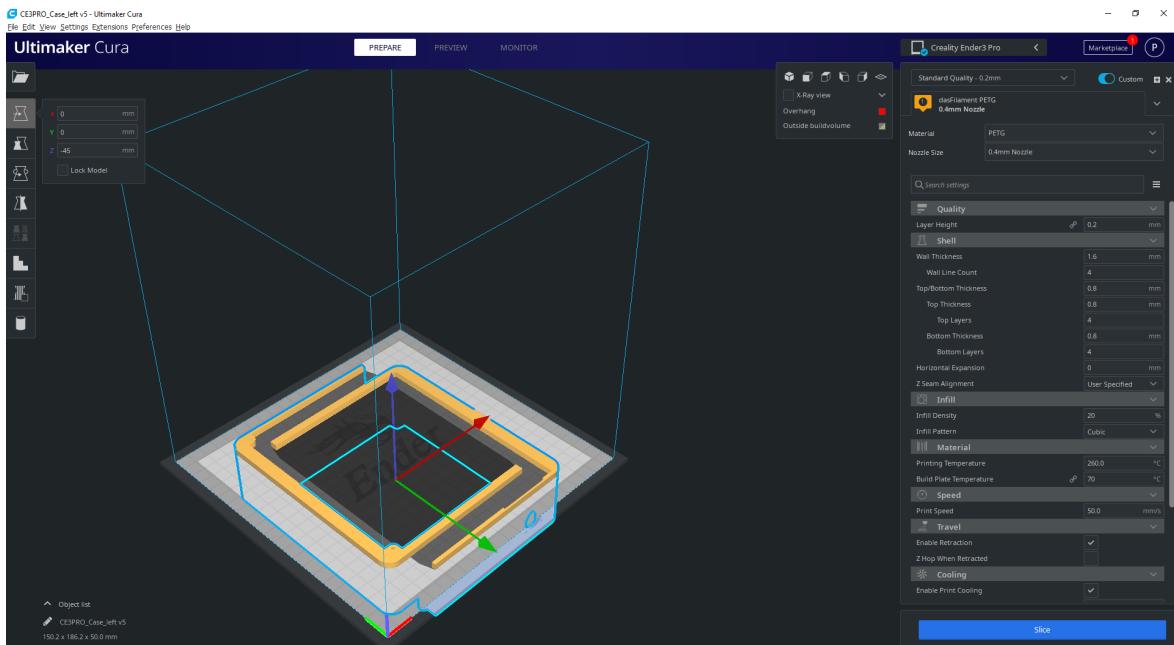


Abbildung 9: Verschiebung der Modelle entlang der Z-Achse um 45mm

### 6.3.2 Modellentwicklung am Objekt

**Grundskizze** Nachdem das Testmodell (vgl. Abbildung 11) nicht zu 100 % gepasst hat, hat Manuel Starz die Abmessungen neu geklärt und diese in Fusion 360 übertragen (vgl. ??). Um Material für den 3D-Druck zu sparen, wurde die Zeichnung dann im Maßstab 1:1 auf Papier gedruckt, ausgeschnitten und angelegt. Da hier einige Maße noch nicht gestimmt haben, hat Manuel Starz den Plan überarbeitet (vgl. ??). Diese neuen Bemaßungen waren dann korrekt. Daraufhin wurde dann die Zeichnung in zwei eigenständige Dateien gesplittet, um die linke und die rechte Seite des Gehäuses zu konstruieren.

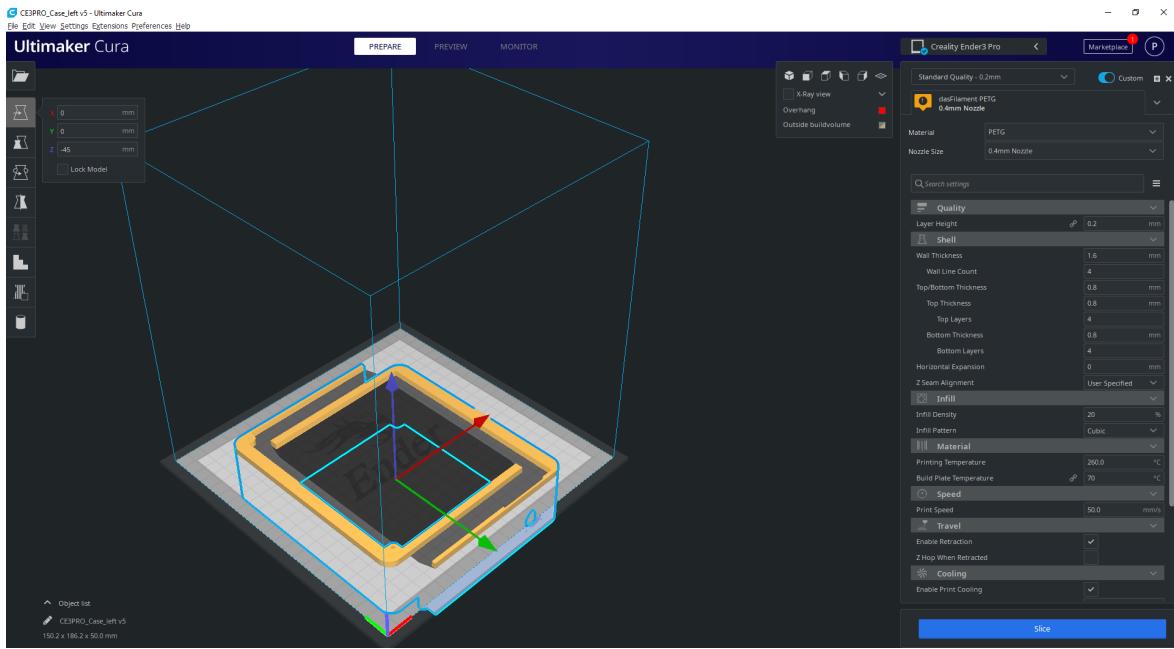


Abbildung 10: Slicen der Modelle

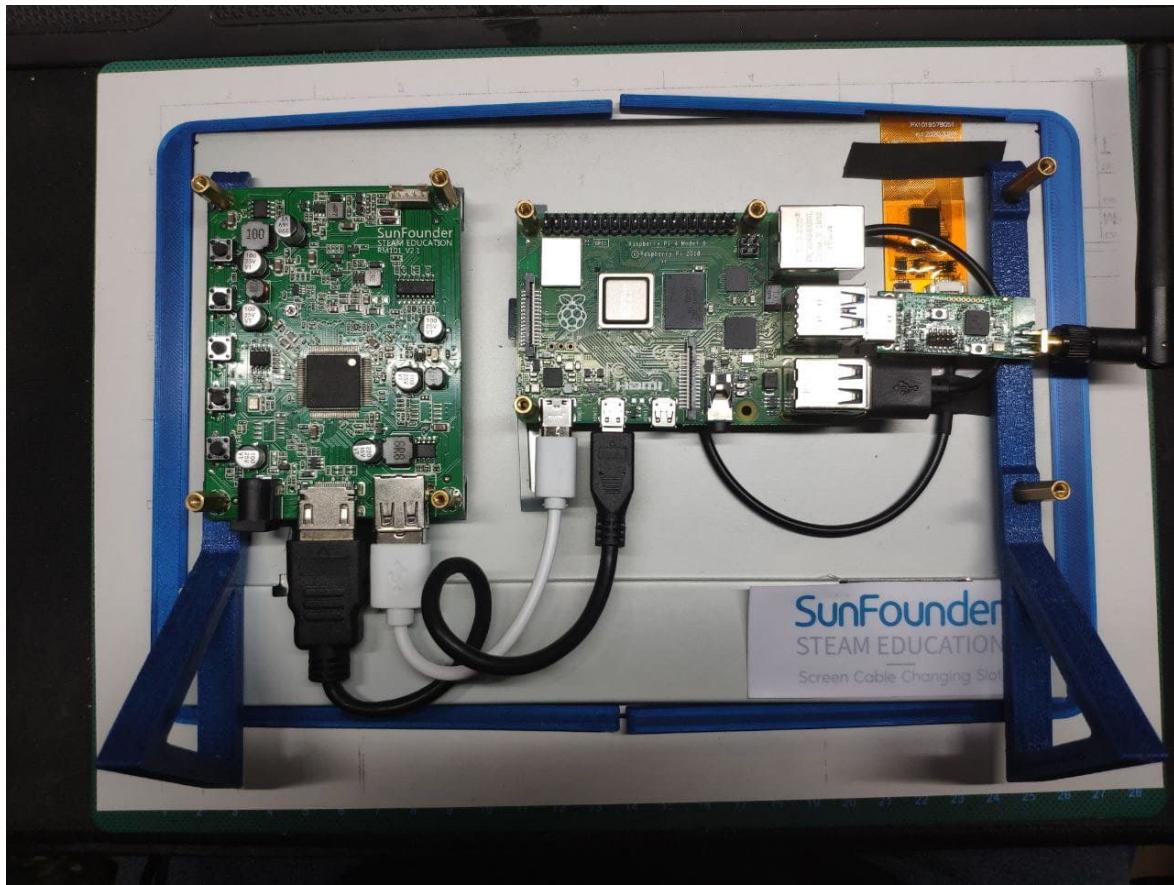


Abbildung 11: Passung des Testdrucks

Die Grundabmessung des Bildschirms werden in Fusion 360 als neue Zeichnung angelegt. Hierzu wurde ein einfaches Rechteck mit den entsprechenden Außenmaßen angelegt (vlg. 12a). Anschließend wurden die Ecken mit einem Radius von 7mm

abgerundet (vgl. 12b). Abschließend wurde die Zeichnung in der Mitte geteilt, um die beiden Hälften des Gehäuses unabhängig von einander zu konstruieren.

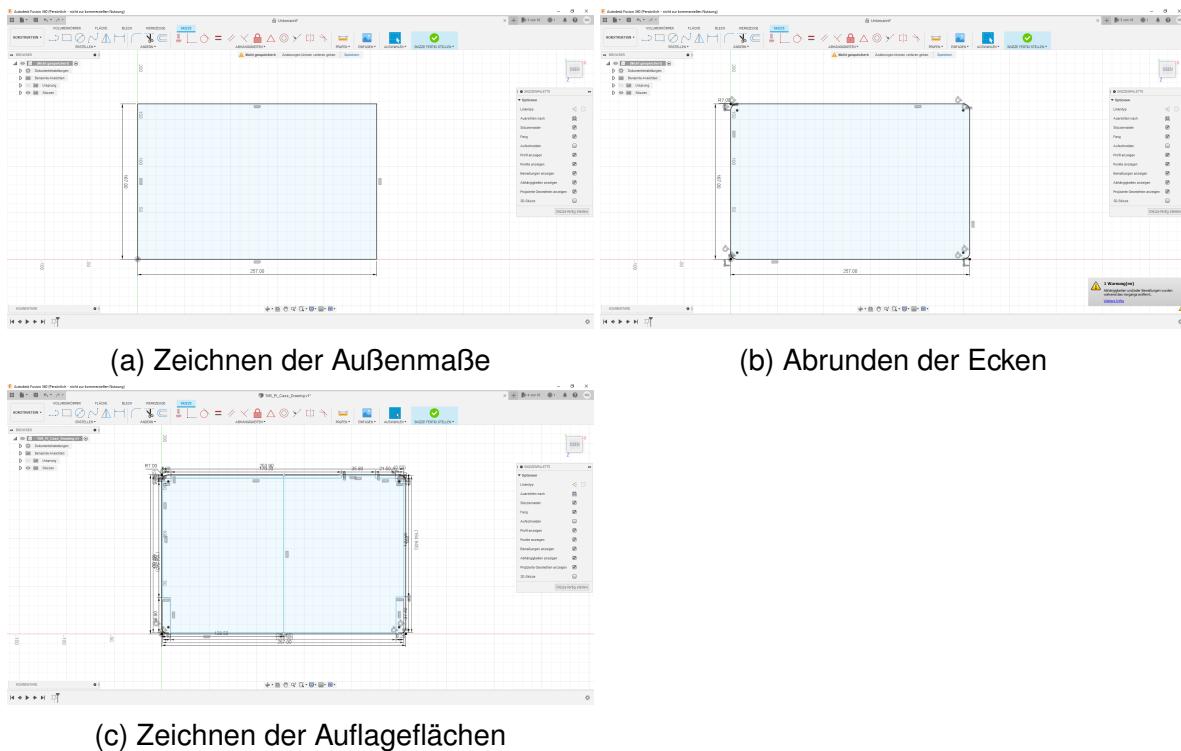


Abbildung 12: Grundzeichnung als Basis des Modells

Die so entstandene Zeichnung (vgl. 12c) wurde dann in eine weitere Datei kopiert um als Grundlage für die beiden geteilten Seitenteile zu dienen. Von hier aus wurden die beiden Wand-Teile mehr oder weniger unabhängig voneinander entworfen.

**Linkes Wandteil** Beim linken Wandteil wurde der entsprechende Teil der Zeichnung um 2 mm entlang der Z-Achse extrudiert (vgl. 13a). Auf der erhöhten Seite wurde darauf hin eine weitere Zeichnung gelegt, die die Bleche an der Rückseite des Bildschirms überdecken sollte (vgl. 13b), die dann wie zuvor um 3 mm entlang der Z-Achse extrudiert wurde (vgl. 13c). Diese sollte dem Gehäuse genug Auflagefläche an dem Bildschirm bieten, um die Verklebung so stark wie möglich zu machen. Auf die nun entstandene erhöhte Seite wurde eine Zeichnung der „tatsächlichen“ Wandstärke von 3 mm erstellt (vgl. 13d). Diese wurde dann um 6 mm entlang der Z-Achse extrudiert, um für die Verbinder genügend Platz vor dem klobigen Blechbereich im unteren Teil des Bildschirms zu bieten (vgl. 13e). Auf die nun obenliegende Seite wurde die Zeichnung der Verbindungsstücke gelegt (vgl. 13f) und um 19 mm entlang der Z-Achse extrudiert (vgl. 13g) und im Kreismittelpunkt der Zeichnungen eine Bohrung für M3-Gewinde gesetzt (vgl. 13h). Auf die nun entstandene Oberfläche wurde die Zeichnung für die Deckelverbindung gesetzt (vgl. 13i), um 25 mm entlang der Z-Achse extrudiert (vgl. 13j) und ebenfalls mit Bohrungen für M3-Gewinde versehen (vgl. 13k).

**Rechtes Wandteil** Beim rechten Teil des Gehäuses war die Vorgehensweise weitestgehend die selbe wie beim linken Gehäuseteil (vgl. 14a - 14h mit 13). Der

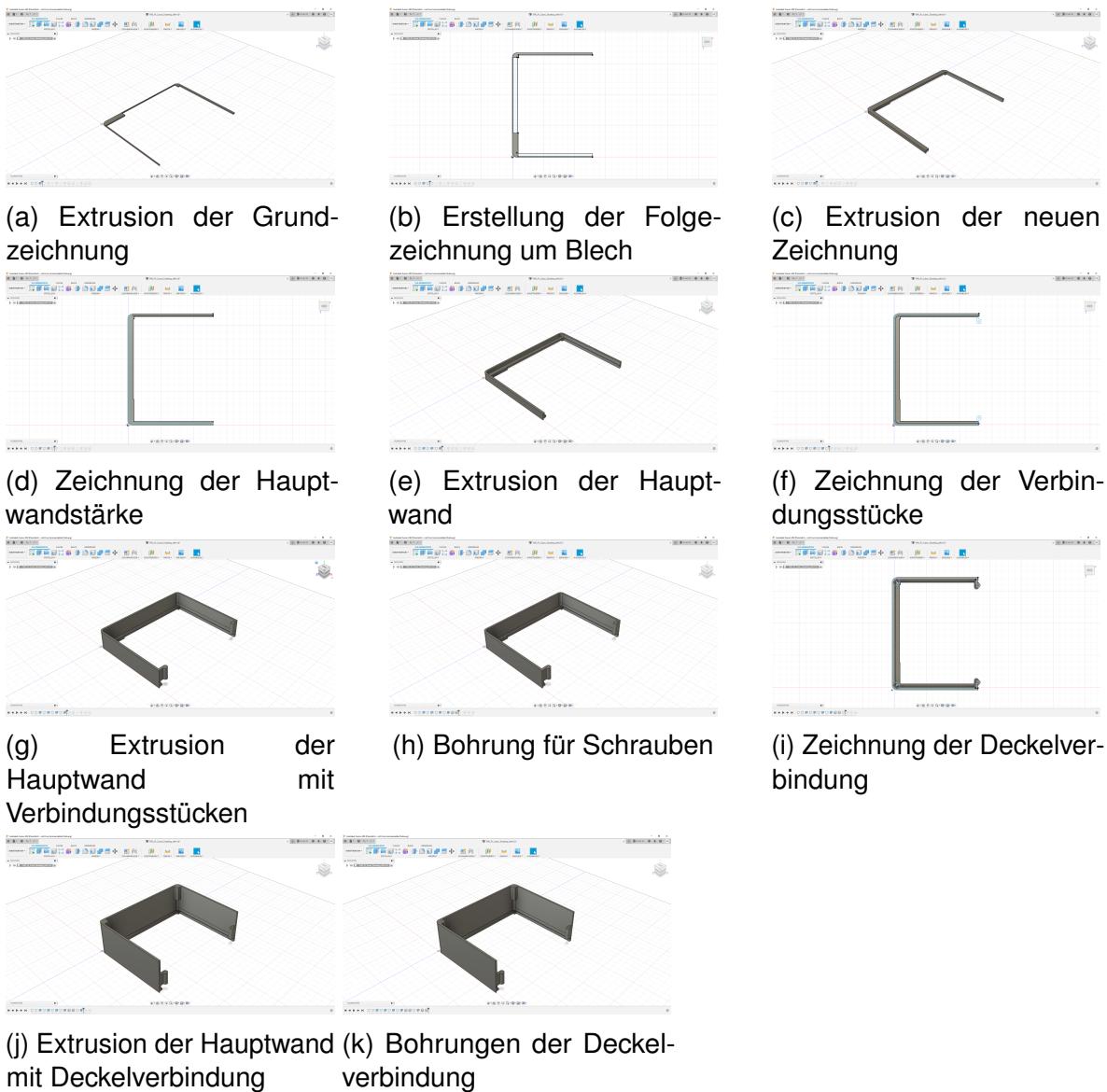


Abbildung 13: Entwurf des linken Wandteils

Unterschied zwischen beiden Teilen, abgesehen von den Verbindungsstücken und der Aussparung für das Flachkabel des Bildschirms, waren die Aussparungen für Lüfter, Antenne, Netzwerkanschluss und Strombuchse. Für den Lüfter und die Antenne wurde auf der „Oberseite“ des Gehäuses eine Zeichnung aufgelegt (vgl. 14i), die dann ins Negative extruiert wurde, was die gezeichnete Fläche aus dem Körper löscht (vgl. 14j). Auf ähnliche Weise wurde die Aussparung für eine RJ45-Verlängerung und eine 5,5 mm DC-Buchse gesetzt. Zuerst wurde die Zeichnung für die Aussparung auf die Seite gelegt (vgl. 14k), die dann ebenfalls ins Negative extruiert wurde (vgl. 14l). Um für die Befestigung der 5,5 mm DC-Buchse eine Unterlage im Gehäuse zu haben, wurde der Grundriss eines Rechtecks auf die Innenseite des Gehäusebodens gelegt (vgl. 14m) und extruiert (vgl. 14n), um die Möglichkeit zu bieten, die Buchse im Gehäuse zu verkleben. Damit die Buchse nicht zu tief im Gehäuse steckt, wurde eine Zeichnung eines konzentrischen Kreises auf die bereits vorhandene Aussparung auf die Innenseite gelegt (vgl. 14o) und dann um einige Millimeter ins Negative extruiert, um die Aussparung zu generieren (vgl. 14).

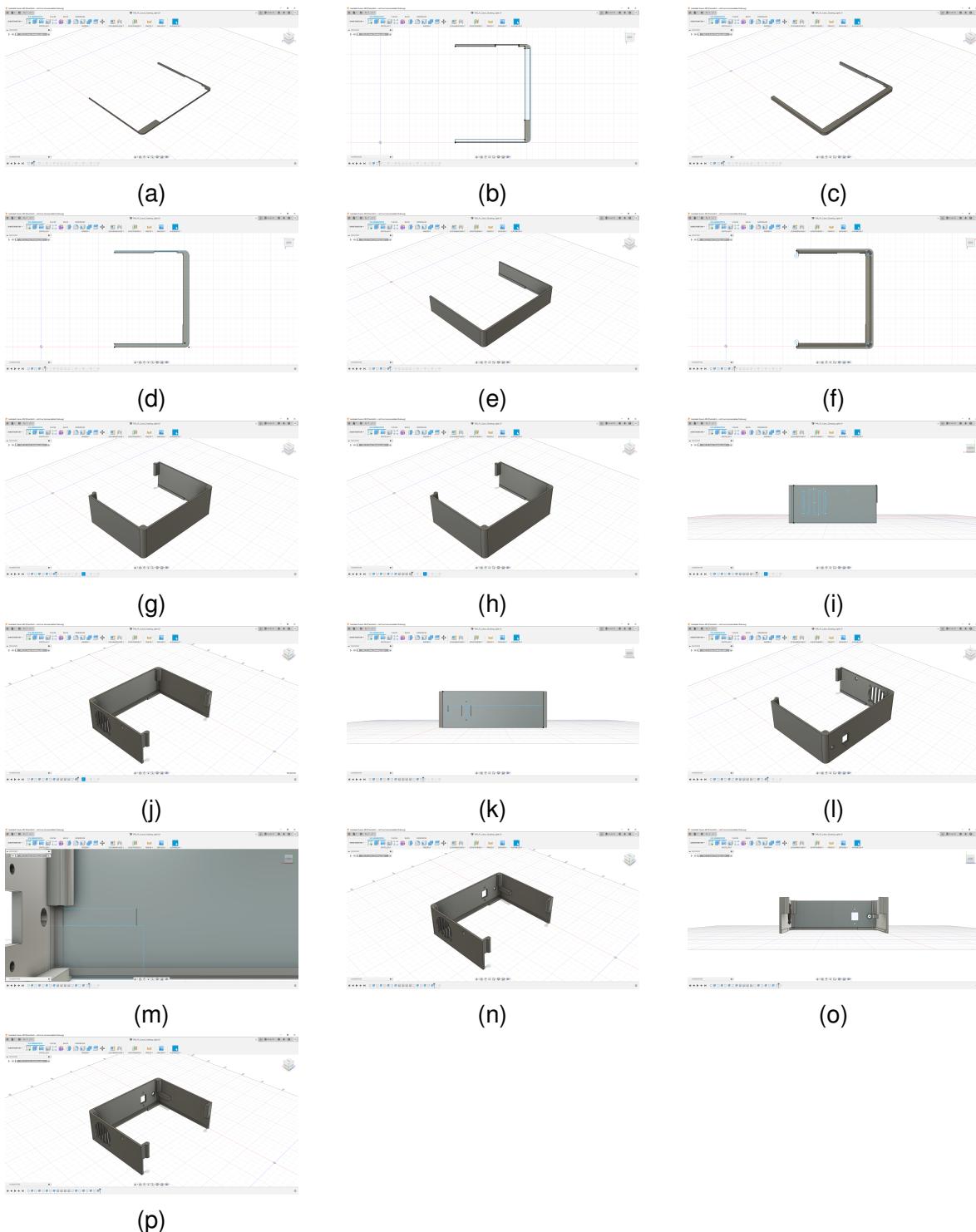


Abbildung 14: Entwurf des rechten Wandteils

**Gehäuserückwand** Das Design der Gehäuserückwand basiert nur rudimentär auf der in 12 erstellten Zeichnung. Die Außenmaße stimmen zwar überein, die Zeichnung für die Auflage auf den Seitenwänden musste aber neu erstellt werden. Zusätzlich wurde eine Diagonale zur punktsymmetrischen Trennung der Teile eingefügt (vgl. 15a), welche die Zeit für die Konstruktion von zwei unterschiedlichen Teilen für die Rückseite reduzieren soll. Die Zeichnung wurde dann entsprechend extrudiert (vgl. 15b). Um die Teile nach dem Druck besser verbinden zu können, wurde ein

Vorsprung auf der kurzen Seite des Bauteils extruiert (vgl. 15c). Um beim Druck Material zu sparen, wurde ein weiter Teil der Innenzeichnung ins Negative extruiert, um den Bereich freizustellen (vgl. 15d). Für eventuelle Toleranzen zwischen den beiden Teilen wurde auf dem Vorsprung die Aussparung für die unterliegende Schraubendurchführung erhöht (vgl. 15e). Zusätzlich wurde der Vorsprung von unten mit einer Phase versehen, um die Materialnutzung für das Teil weiter zu reduzieren (vgl. 15f). Des Weiteren wurde an die obere Kante (vgl. 15g) und an die Innenkante (vgl. 15h) mit einer Phase versehen. Eine ähnliche Phase wurde auch an dem Vorsprung angebracht (vgl. 15i). Um die Bohrung am die richtige Stelle zu setzen, wurde eine Hilfzeichnung auf die Außenfläche gesetzt (vgl. 15j). Anschließend wurden drei M3-Bohrungen für Senkkopfschrauben gesetzt (vgl. 15k). Um das Gehäuse an der Wand anbringen zu können, wurden eine Zeichnung auf die Oberseite des Gehäuses gelegt (vgl. 15l) und dann ins Negativ extruiert (vgl. 15m). Um bei der Anbringung nicht an einen bestimmten Typ Schrauben gebunden zu sein, wurde eine kleine Phase an den engen Teil der Aufhängungslöcher gelegt (vgl. 15n). Damit bei der Verbindung der beiden Einzelteile die Vorsprünge nicht aneinander stoßen, wurde an den Vorsprung auch eine Phase angelegt (vgl. 15o).

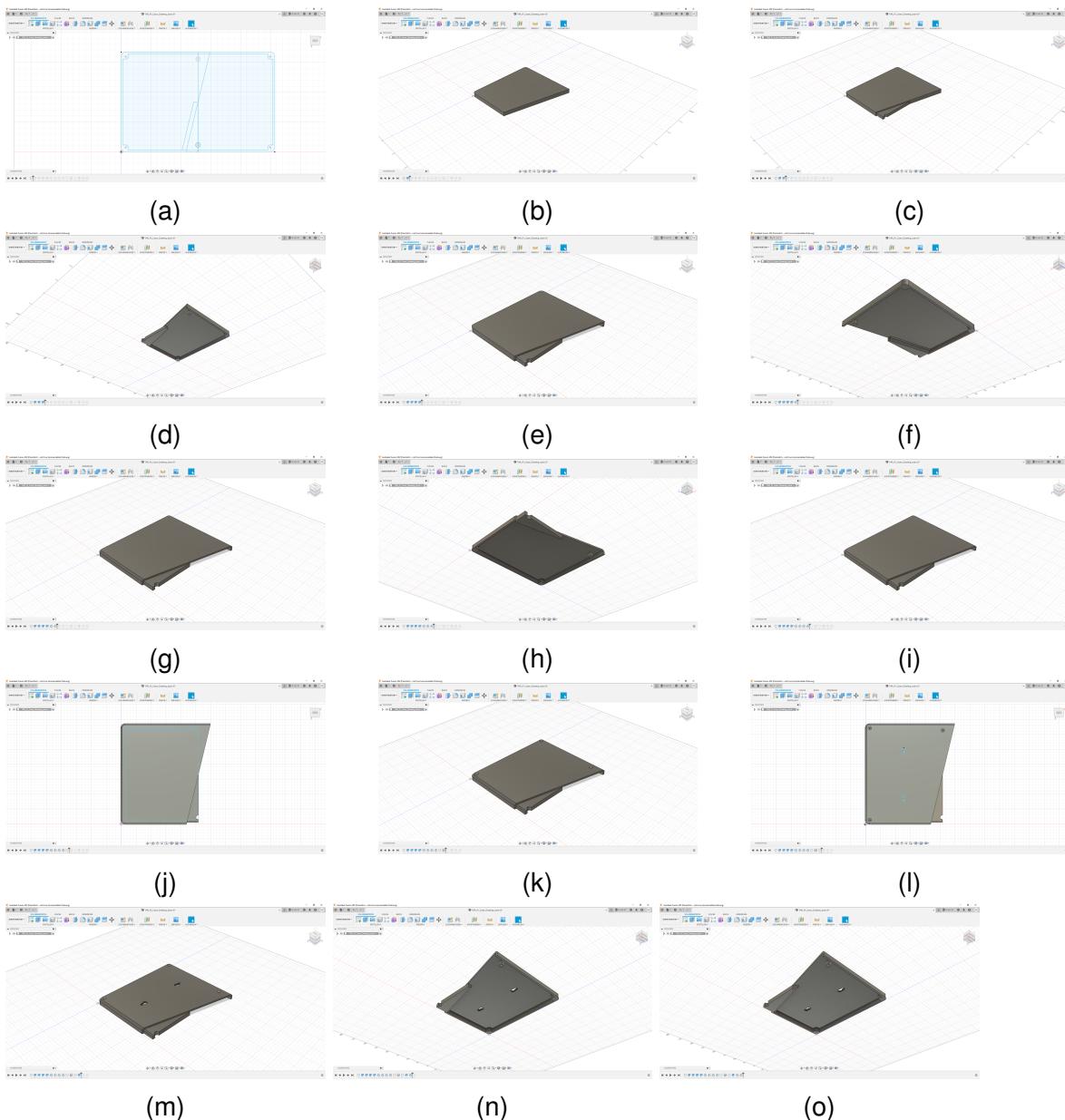


Abbildung 15: Entwurf der Gehäuserückwand

### 6.3.3 Herstellung des Gehäuses

Zur Herstellung des Gehäuses kommt ein 3D-Drucker der Marke Creality 3D zum Einsatz. Der stark modifizierte Ender 3 Pro von Manuel Starz (vgl. 16) druckt die zuvor erstellten STL-Dateien mit PETG-Filament der Firma dasfilament. Der für den 3D-Druck nötige G-Code wird mit Ultimaker CURA generiert und mit Hilfe von OctoPrint (vgl. 17) an den Drucker übertragen. Das Gehäuse besteht aus vier Einzelteilen, je zwei Teile für die Seitenwände und zwei Teile für den Deckel. Die gedruckten Teile (vgl. 19) werden dann zum Teil mit Zwei-Komponenten-Epoxidkleber verbunden (vgl. 20a) und mit Hilfe von Schleifpapier und einigen Schichten Klarlack zu einem klavierlackähnlichen Finish veredelt (vgl. ??). Die Seitenwände werden dann mit dem Bildschirm mit Hilfe des Zwei-Komponenten-Epoxidklebers und Heißkleber permanent verklebt (vgl. 20c). Danach wurden die Platinen und Kabel im



Abbildung 16: 3D-Drucker Ender 3 Pro (mod)

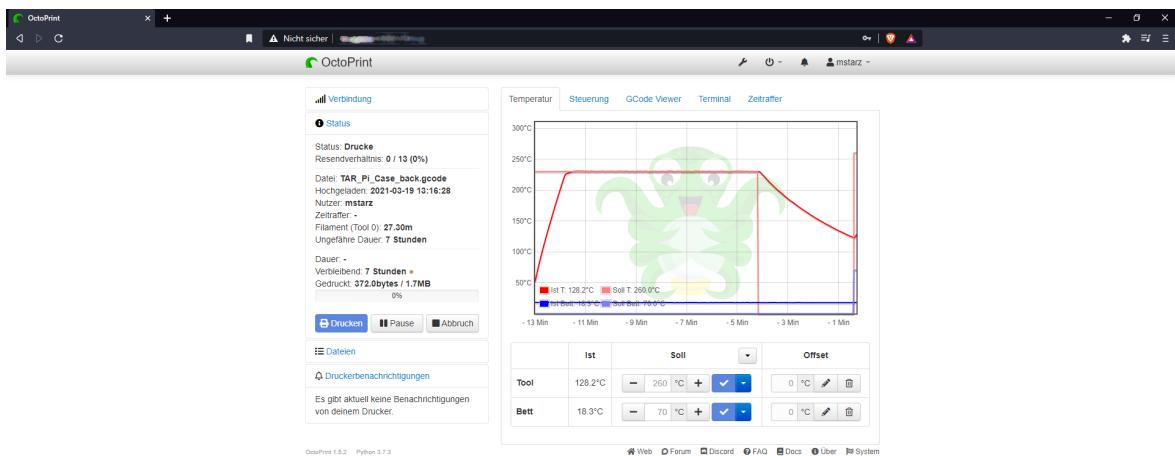
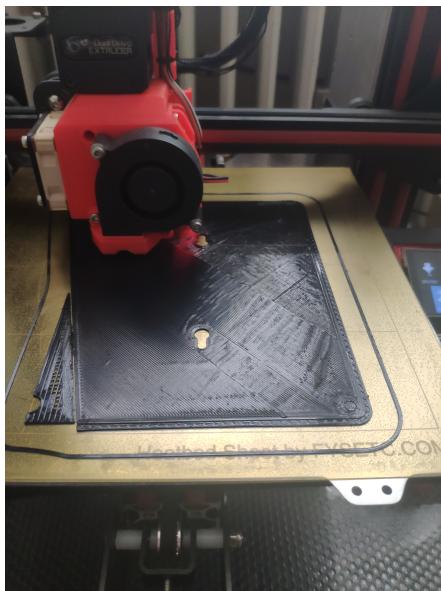


Abbildung 17: OctoPrint-Weboberfläche

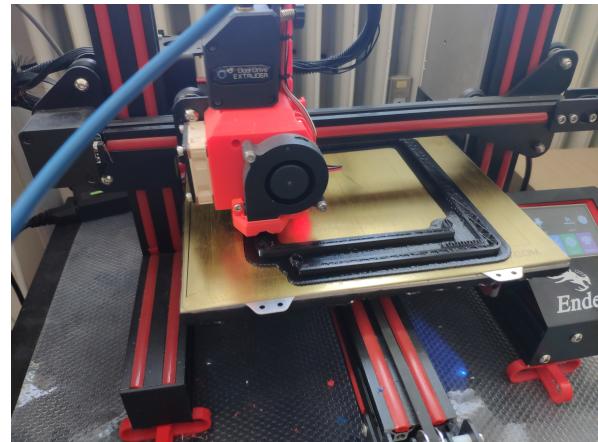
Gehäuse angebracht (vgl. 20c). Die Rückseite besteht der Einfachheit halber aus zwei identischen, punktsymmetrischen Teilen, die ebenfalls miteinander verklebt wurden.

Laut Ultimaker CURA beträgt die Gesamtdruckdauer des Gehäuses 36 Stunden 36 Minuten und verbraucht insgesamt 423 Gramm des verwendeten Filaments (dasFilament PETG Schwarz). Damit belaufen sich die Materialkosten des Gehäuses auf 11,61€.

Bei der Herstellung kam es aber zu einem Problem, dass die Stützstrukturen den Druck am Lüftereinlass an den Seitenwänden gestört haben was dazu führte, dass das Lüftergitter an der linken Seitenwand (vgl. 19b) nicht richtig gedruckt wurde.



(a) Druck eines der Deckel-Teile



(b) Druck der Wand-Teile

Abbildung 18: Das Gehäuse entsteht



(a) Rückseite



(b) Linke Seite



(c) Rechte Seite

Abbildung 19: Ausgedruckte Teile des Gehäuses

## 6.4 Erstellung des RPi-HATs

Zur Schaltplanzeichnung haben wir das OpenSource-Programm KiCAD genutzt. Dieses ist für zahlreiche Betriebssysteme verfügbar.<sup>9</sup> Teilebibliotheken sind zahlreich im Internet verfügbar, wir haben uns auf den Anbieter SnapEDA beschränkt, da dieser die meisten gängigen Bauteile für Elektronik-CAD-Systeme anbietet. Eine weitere Bezugsquelle ist der Verkäufer Digi-Key Electronics, der einen Großteil seines Sortiments an SMD-Bauteilen als Bibliothek anbietet<sup>10</sup>. Als Grundlage für den HAT sollte die Projektvorlage „Raspberry Pi - 40-Pin HAT“ von Jon Buford dienen. Diese Vorlage richtete sich nach den offiziellen HAT-Spezifikationen für den Raspberry Pi<sup>11</sup>. Zu Beginn des Projekts haben wir in KiCAD die GPIO-Ports anhand der offiziellen Dokumentation beschriftet, um das Nachschlagen der Belegung in Zukunft zu vermeiden.

<sup>9</sup>Siehe <https://www.kicad.org/download/>

<sup>10</sup>Siehe <https://www.digikey.de/de/resources/design-tools/kicad>

<sup>11</sup>Siehe <https://github.com/raspberrypi/hats>



Abbildung 20: Fertigung des Gehäuses

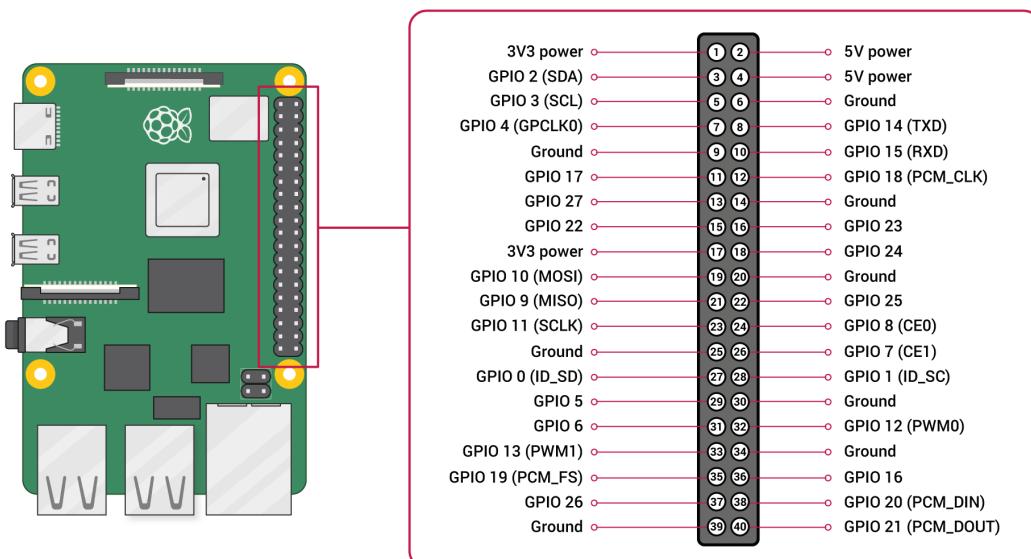


Abbildung 21: Raspberry Pi 4 GPIO-Pins

Anschließend haben wir die (unserer Meinung nach) benötigten Komponenten auf

den Schaltplan gezogen beziehungsweise vorher importiert und dann versucht, diese sauber zu verdrahten.

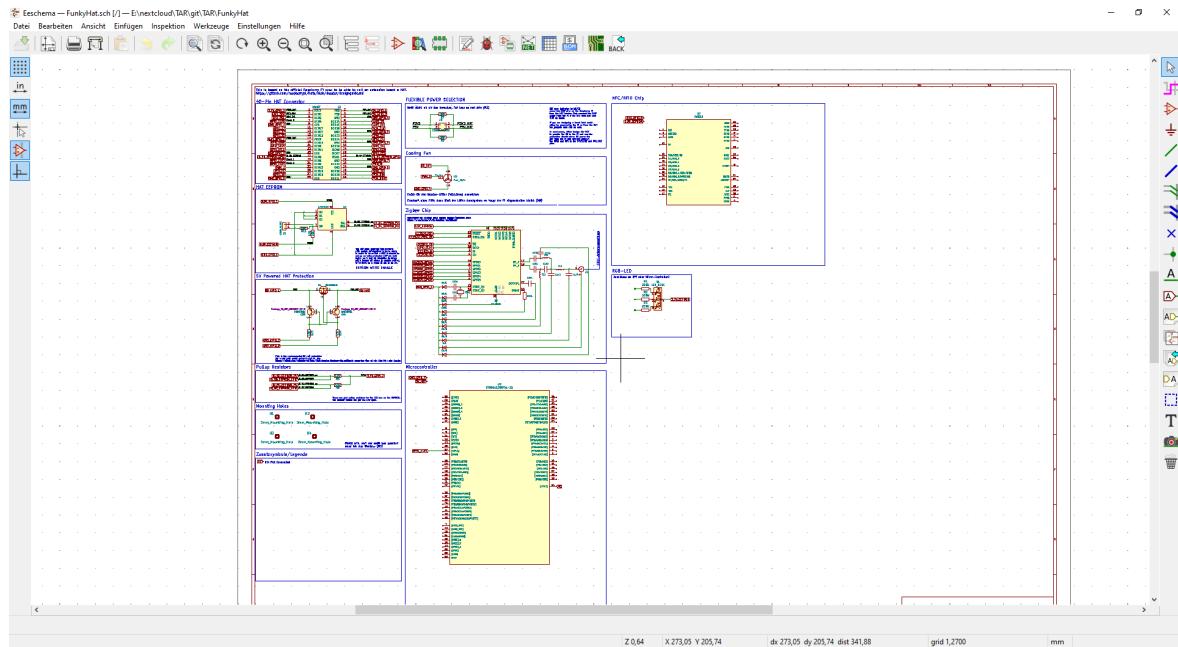


Abbildung 22: Aktueller Stand HAT-Design

Aufgrund der bereits weit vorangeschrittenen Zeit im Projektplan, der aktuellen COVID-19-Pandemie und unserer Unwissenheit im Bezug auf Schaltplanentwicklung sowie der von uns verwendeten Hardware mussten wir aber einsehen, dass die Entwicklung und Fertigung eines HATs den Zeitrahmen sprengen würde und so haben wir zu Ende Februar 2021 beschlossen, dieses Projekt vorerst zurück zu stellen und uns auf die Fertstellung unserer bisherigen Testerfolge und Prototypen zu fokussieren. Die KiCAD-Projekt-Dateien stehen, wie alle anderen Dateien des Projekts nach Abgabe frei über das entsprechende GITHUB zur Verfügung.

## 7 Software

### 7.1 Überblick

Auf der Softwareseite dient Raspberry Pi OS<sup>12</sup> als Basis für die Smart Home Zentrale. Darauf laufen die gewählten Softwarelösungen

- Home Assistant Supervised (Web Oberfläche und Verwaltung)
- Mosquitto Broker (MQTT Broker)
- Zigbee2mqtt (Zigbee-Schnittstelle)

Zur Installation der einzelnen Komponenten und für die Vorkonfiguration des Systems haben wir ein bash-Skript (vgl. 10.2) erstellt, dass das System entsprechend vorbereitet und die einzelnen Komponenten installiert.

### 7.2 MQTT-Broker

### 7.3 Zigbee

### 7.4 HomeAssistant

### 7.5 MyCroft

---

<sup>12</sup>Raspberry Pi OS with Desktop, Kernel Version 5.4

## 8 Epilog & Fazit

### 8.0.1 $\text{\LaTeX}$ vs Word

Danke Herr Kohler. Ihre fast 2 Jahre langes Gejammer über WYSIWYG-Dokumenteneditoren haben dazu geführt, dass wir uns in den letzten Zügen der Dokumentation dazu entschlossen haben, OHNE JEGLICHE VORERFAHRUNG diese in  $\text{\LaTeX}$  zu schreiben. Ich hoffe, Sie sind stolz auf das Monster, dass Sie geschaffen haben...

## 9 Quellen

### 9.1 Dokumentationsquellen

Quellen für Verweise, die in Fußnoten innerhalb dieser Dokumentation erwähnt wurden:

- Li (2017): Was ist ein Smart-Home-Hub? Alles über die intelligente Zentrale  
<https://www.otto.de/updated/ratgeber/erklaert-was-ist-ein-smart-home-hub-80634/>
- Robert (2019): Antwort auf Creality Ender 3 printer power consumption? - 3dprinting Stack Exchange  
<https://3dprinting.stackexchange.com/questions/8616/creality-ender-3-printer-power-consumption#8623/>
- reichelt.de: Produktbeschreibung des Raspberry Pi 4  
<https://www.reichelt.de/raspberry-pi-4-b-4x-1-5-ghz-8-gb-ram-wlan-bt-rasp-pi-4-b-8gb-p276923.html>
- Wikipedia: Raspberry Pi - Generations  
[https://en.wikipedia.org/wiki/Raspberry\\_Pi#Generations](https://en.wikipedia.org/wiki/Raspberry_Pi#Generations)
- Sunfounder: 10.1 Inch Touch Screen for Raspberry Pi(NEW)  
[http://wiki.sunfounder.cc/index.php?title=10.1\\_Inch\\_Touch\\_Screen\\_for\\_Raspberry\\_Pi\(NEW\)#3D-printed\\_Touch\\_Screen\\_Support](http://wiki.sunfounder.cc/index.php?title=10.1_Inch_Touch_Screen_for_Raspberry_Pi(NEW)#3D-printed_Touch_Screen_Support)

Quellen, die für die Herstellung dieser Dokumentation genutzt wurden:

- Menmiloud Mohammed: L<sup>A</sup>T<sub>E</sub>X-Tutorial.com  
<https://latex-tutorial.com/>
- Overleaf: L<sup>A</sup>T<sub>E</sub>X-Guides  
<https://www.overleaf.com/learn>

### 9.2 Verwendete Software

Software	Verwendung	Version
Raspberry Pi OS	Betriebssystem und Oberfläche für Hardware	5.4
Home Assistant	Betriebssystem und Oberfläche für Smart Home	5.12
KiCad EDA	Erstellung von Schaltplan und Gerber-Datei des Hats	5.1.8
TexMaker	Erstellung der Dokumentation	5.0.4
Fusion360	Erstellung von Gehäusemodell	2.0.9849
CURA	Erstellung von G-Code für 3D-Drucker	4.7
GanttProject	Erstellung von Gantt-Diagrammen	3.0.3
Autodesk Meshmixer	Vorschau-Render der STL-Dateien	3.5.474

### 9.3 Verwendete Hardware

Hardware	Verwendung	Version
Raspberry Pi	Hauptplatine für die Smart Home Zentrale	Version 4B (8GB)
CC2531 Zigbee USB Stick mit Firmware	USB-Stick mit ZigBee-Chip	Rev 2.4
Sunfounder 10.1 Touch Screen	Bildschirm und Input für die Smart Home Zentrale	Unbekannt

## 10 Anhang

### 10.1 Lastenheft

	TAR	Erstellt am: 04.06.2020 13:35:00
SmartHome e.V. An der Ecke 12 70707 Kleinstadt	Lastenheft	

## Lastenheft

### I. VERANLASSUNG

Der Smart-Home Sektor wird zu einem immer größer werdenden Markt und immer mehr Hersteller liefern neue Produkte um das eigene Haus zu steuern. Um einen zentralen Informationspunkt oder eine zentrale Steuereinheit einzurichten, soll ein Produkt entwickelt werden, welches sich mit Smart-Home-Anwendungen verschiedener Hersteller verbinden kann, um deren Produkte dann per Touchscreen oder wahlweise per Sprachassistent steuern kann.

### II. PROJEKTUMFELD

Für das Projekt stehen diverse Smart-Home-Komponenten zur Verfügung, darunter unter anderem die Smart-Lampen von IKEA, Phillips Hue und diverse Zigbee Komponenten.

### III. ZIELBEDINGUNG

Das Produkt soll eine Lösung zur Verknüpfung von Smart-Home-Geräten via Zigbee-Bridge darstellen und einen Sprachassistenten integrieren, in diesem Fall MyCroft. Darüber soll das Produkt als zentrale Steuereinheit für die angebundenen Sensoren und Geräte dienen und den Status dieser in einem kompakten Display darstellen.

### IV. PRODUKTEINSATZ

Das Produkt dient zur Steuerung eigener Zigbee fähiger Smart-Home-Komponenten. Als solches kann es vom Kunden im eigenen Netz integriert werden und dient dort als Steuerung oder Übersicht der eigenen Smart-Home-Geräten

### V. PRODUKTMERKMALE

- V.1. Hardware in einem Gehäuse (Touch-Bildschirm mit Raspberry Pi 4B als „Gehirn“)
- V.2. Dashboard-Funktion für Smart-Home-Geräte
- V.3. Integration von Zigbee, MQTT
- V.4. GUI erstellt über Java oder als WebUI (eventuell als Smartphone-App)
- V.5. Mögliche Integration von Sprachassistenten MyCroft

### VI. PRODUKTDATEN

Das Gerät soll neben Daten der Sensoren (Luftdruck, Temperatur, Feuchtigkeit, etc.) auch den Zustand von anderen Smart-Home-kompatiblen Geräten anzeigen können (ein- oder ausgeschaltet, Fenster offen, etc.)

### VII. ABLAUFORGANISATION

Die Abläufe für die in Punkt V genannten Vorgänge entsprechen zum Großteil dem geplanten Umfang des Projekts. Zusätzlich soll noch ein Gehäuse entwickelt werden, dass die Hardware (Bildschirm, SBC, Mikrofon und eventuell Kamera und Netzteil) in einem Gerät vereint. Weitere Produktfähigkeiten werden bei Bedarf nachgereicht. Der Projektlauf soll als SCRUM stattfinden, ein Sprint hat die geplante Dauer von einem Monat.

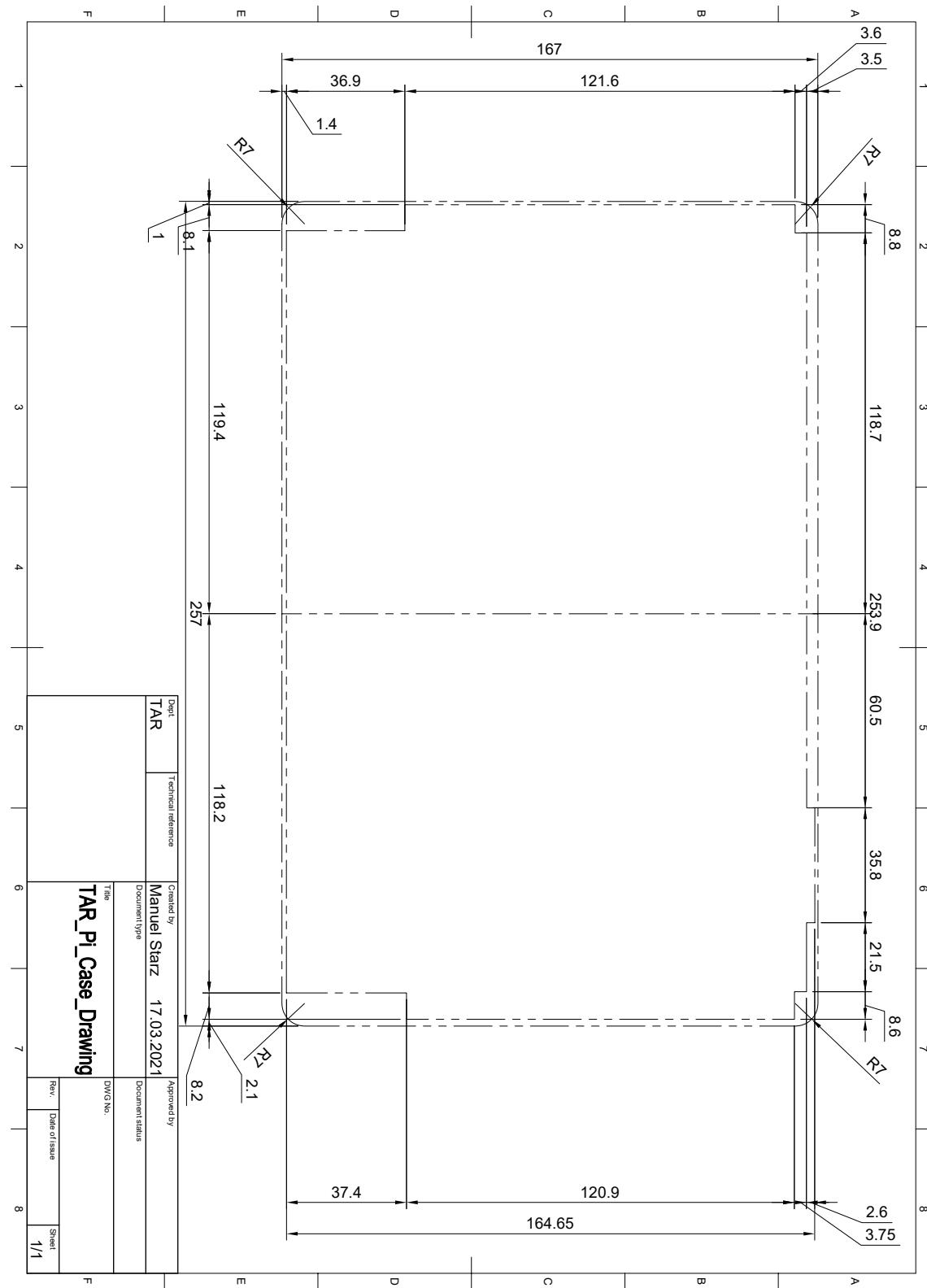
 it.schule stuttgart	TAR	Erstellt am: 04.06.2020 13:35:00
	Lastenheft	

### VIII. PRODUKTLEISTUNG

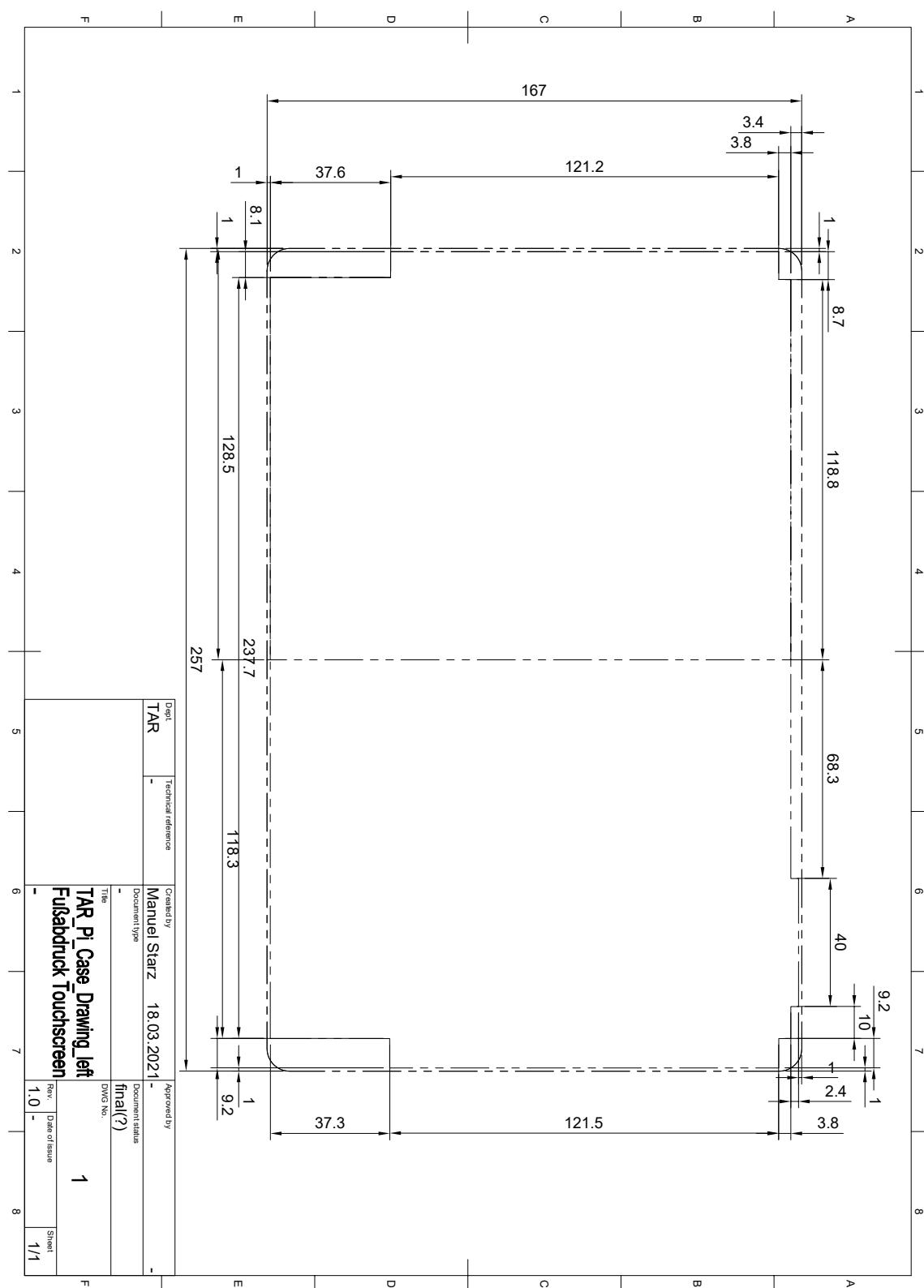
	Sehr gut	Gut	Normal	Nicht Relevant
Funktionalität		x		
Bedienerfreundlichkeit	x			
Änderbarkeit		x		
Erweiterbarkeit			x	

## 10.2 Installationsskript

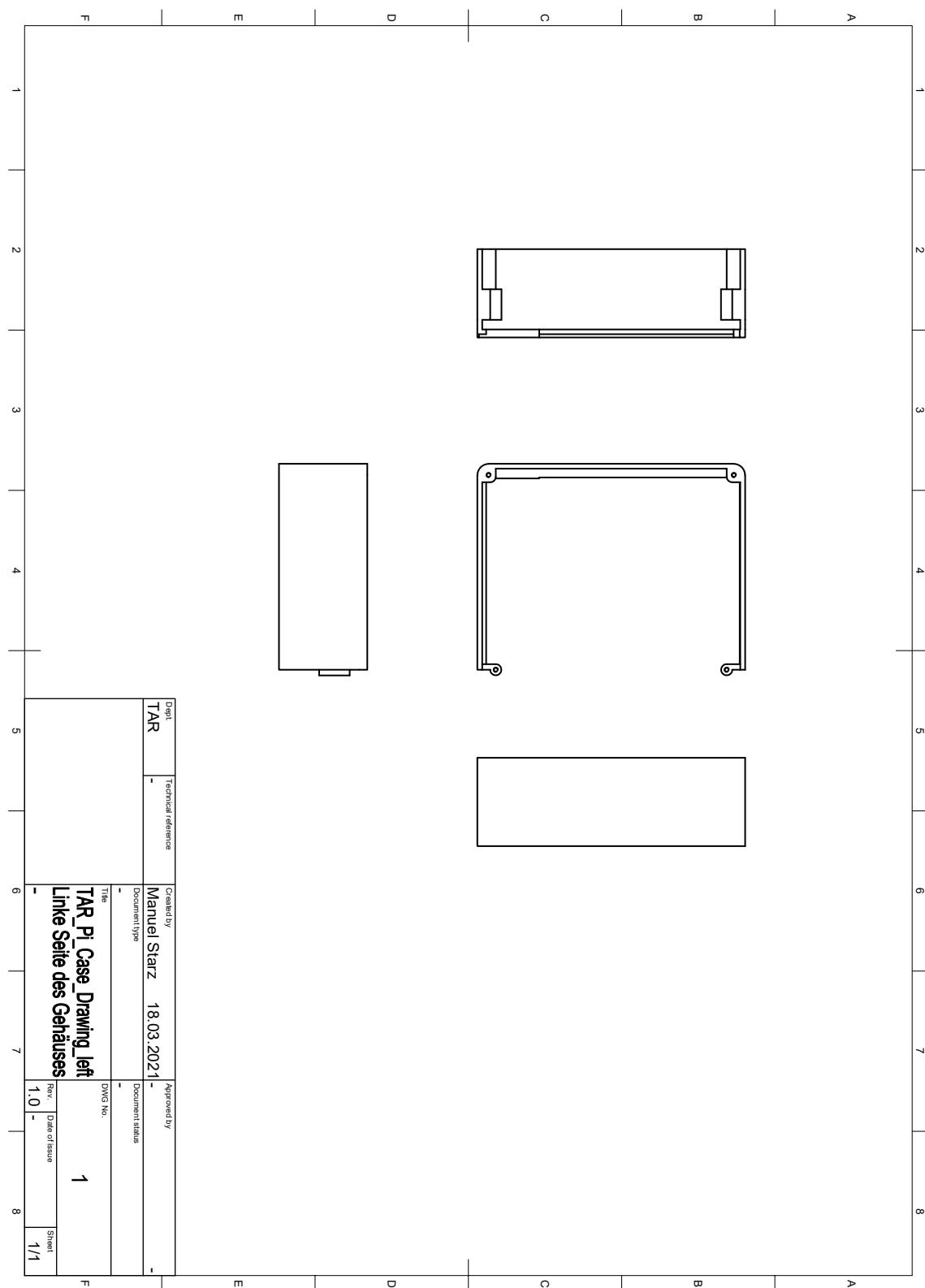
## 10.3 Gehäuse-Zeichnungen



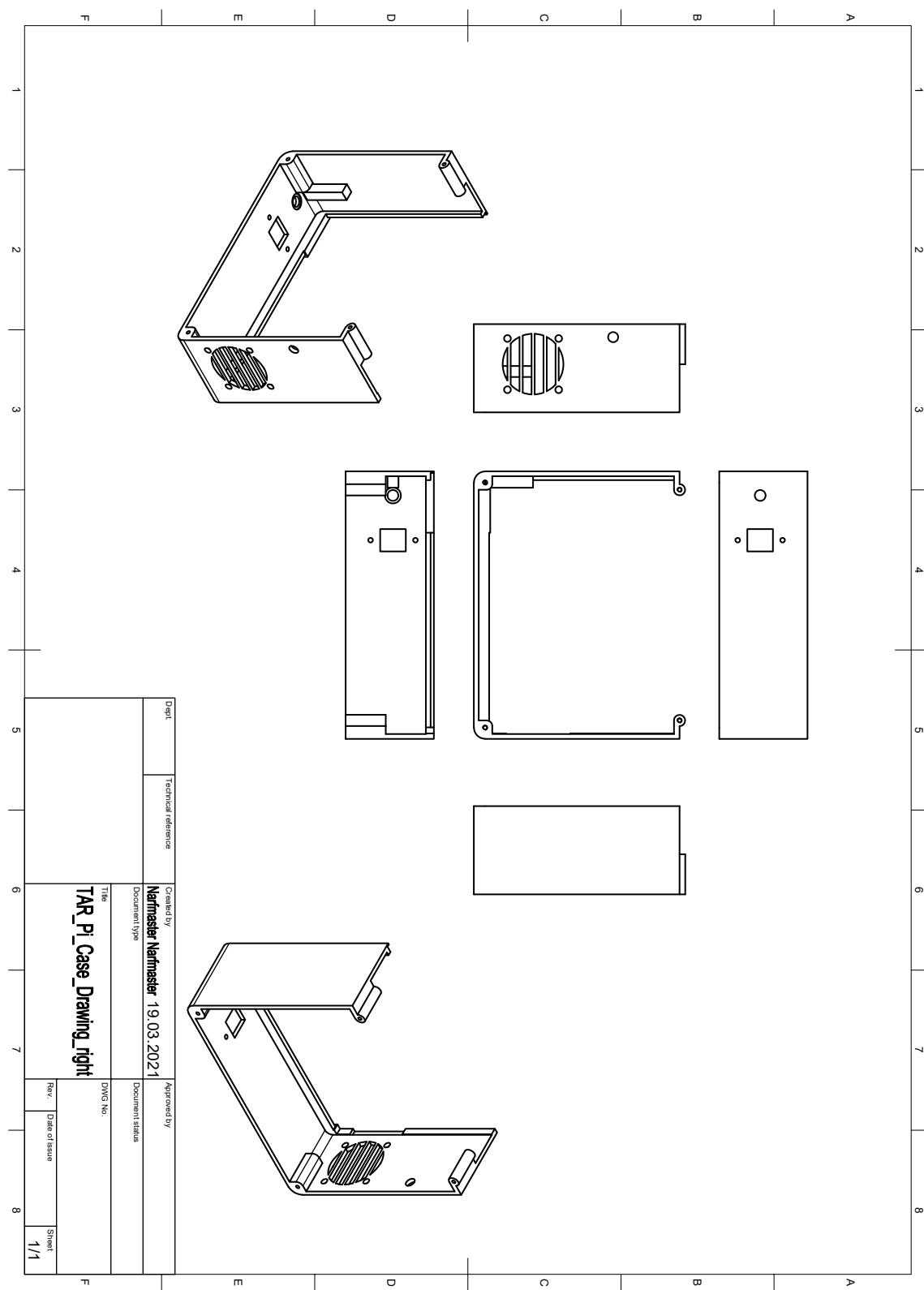
Erster Versuch der Zeichnung des Fußabdrucks des Gehäuses



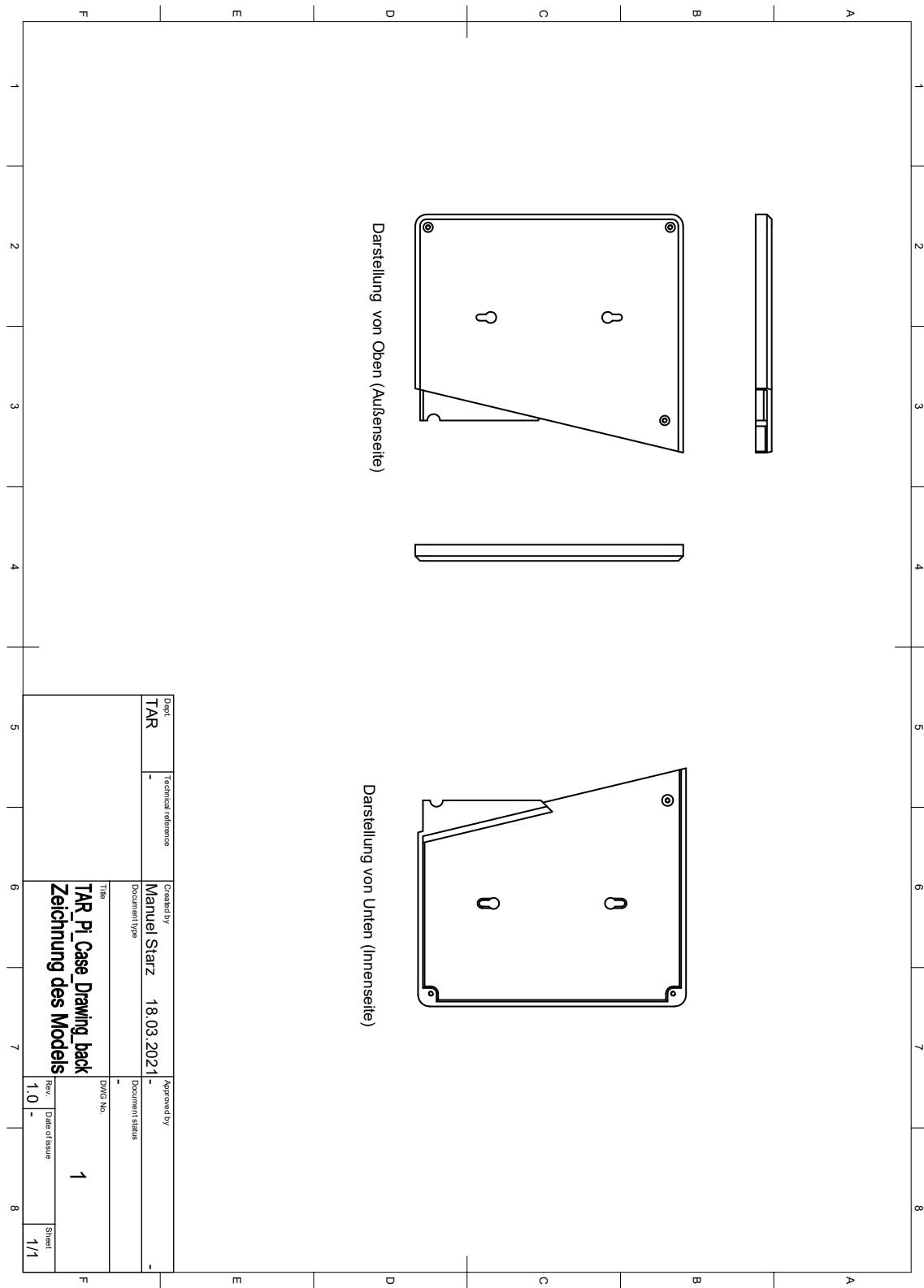
Finale Version der Zeichnung des Fußabdrucks des Gehäuses



Linker Teil des Gehäuses

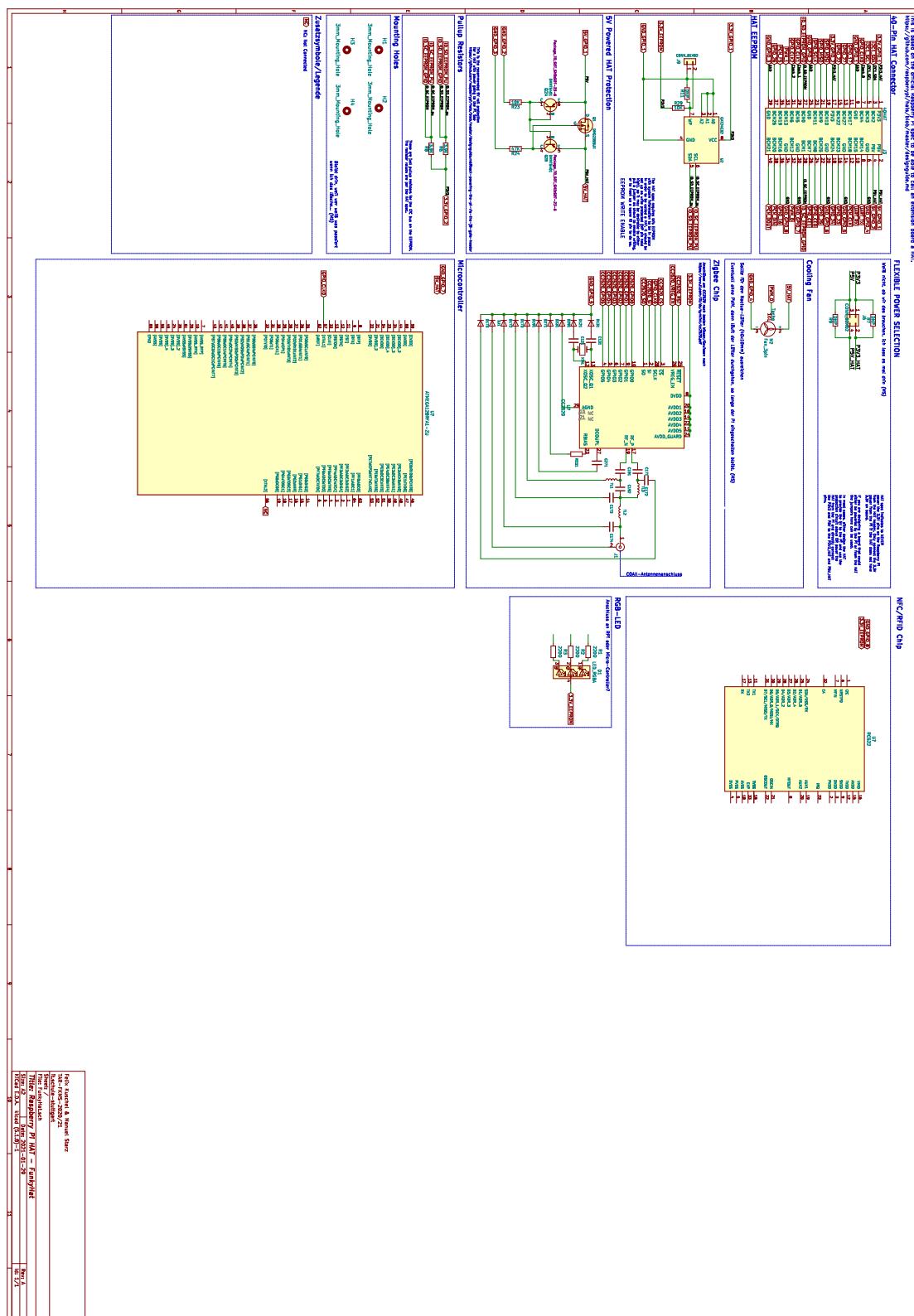


Rechter Teil des Gehäuses



Ein Teil des Gehäusesdeckels

## 10.4 Raspberry Pi HAT E-Schema



## 10.5 Hardware-Dokumentationen

Durch den Umfang der einzelnen Dokumentationen hier nur eine Auflistung der Dokumentationen mit dem Link zu den PDF im GIT-Projekt bzw. den Herstellerseiten.

- Raspberry Pi:  
[https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2711/rpi\\_DATA\\_2711\\_1p0.pdf](https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2711/rpi_DATA_2711_1p0.pdf)
- MiFare MFRC522:  
<https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>
- CC2531 ZigBee SoC:  
<https://www.ti.com/lit/ds/symlink/cc2531.pdf>
- ATmega128RFA1-ZU:  
[https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8266-MCU\\_Wireless-ATmega128RFA1\\_Datasheet.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8266-MCU_Wireless-ATmega128RFA1_Datasheet.pdf)

## Abbildungsverzeichnis

1	Beispiele für Smart Home Zentralen . . . . .	5
2	Beispiele für Smart Home Software . . . . .	5
3	gantt-Diagramm des Projektablaufs . . . . .	9
4	Smart Home Zentrale . . . . .	10
5	Gehäuse mit Füßen (Render in Meshmixer) . . . . .	11
6	Prototyp der Smart Home Zentrale . . . . .	14
7	Abmessungen Bildschirm Rückseite . . . . .	15
8	Platzierung der beiden Gehäusewände in CURA . . . . .	16
9	Verschiebung der Modelle entlang der Z-Achse um 45mm . . . . .	16
10	Slicen der Modelle . . . . .	17
11	Passung des Testdrucks . . . . .	17
12	Grundzeichnung als Basis des Modells . . . . .	18
13	Entwurf des linken Wandteils . . . . .	19
14	Entwurf des rechten Wandteils . . . . .	20
15	Entwurf der Gehäuserückwand . . . . .	22
16	3D-Drucker Ender 3 Pro (mod) . . . . .	23
17	OctoPrint-Weboberfläche . . . . .	23
18	Das Gehäuse entsteht . . . . .	24
19	Ausgedruckte Teile des Gehäuses . . . . .	24
20	Fertigung des Gehäuses . . . . .	25
21	Raspberry Pi 4 GPIO-Pins . . . . .	25
22	Aktueller Stand HAT-Design . . . . .	26