

Fachbericht

PROJEKT 4 - TEAM 1
8. Juni 2019

Auftraggeber:

Prof. Hans Gysin

Fachcoach:

Prof. Dr. Pascal Schleuniger
Matthias Meier
Albert Zihlmann
Dr. Anita Gertiser
Pascal Buchschacher
Dr. Marie-Thérèse Rudolf von Rohr

Projektleiter:

Fabian von Büren

Team:

Christoph Kuhn
Dennis Aeschbacher
Raffael Anklin
Raphael Nikles
Robin Aebi

Studiengang:

Elektro- und Informationstechnik

Semester:

Frühlingssemester 2019

Abstract

In this project, a pre-manufactured 3D printer was modified to demonstrate how the often low standards of printing quality, safety and user friendliness of cheap 3D printers may be enhanced. This includes but is not limited to the independent control of four stepper motors, zero-position recognition for all of the printer's axes and wireless transfer of GCode. Micro stepping capable motor drivers and a WiFi module were used for this purpose. A custom control board was designed with a strong emphasis on keeping it as small as possible. As the control board's firmware, the open source project Marlin was adapted and installed. An auto-leveling system for the heatbed and a filament run-out sensor were integrated into the printer. These measures increase the printer nozzle's movement resolution by a factor of 16 and allow for smoother acceleration, which results in a noticeable reduction of vibrations as well as noise, and a cleaner surface of printed objects. The Wifi-Module enables the wireless transfer of GCode files and the remote monitoring of printing progress as well as printer status information such as temperatures and movement. The results prove that the quality of a 3D printers control board has a significant impact on the printer's performance. The printing quality, safety and user-friendliness of low-end 3D printers may be highly improved by using a better control board.

Inhaltsverzeichnis

1 Einleitung	1
2 Gesamtübersicht	2
2.1 Anforderungen	2
2.2 Kriterien	2
2.3 Lösungskonzept	3
3 Technische Grundlagen	5
3.1 Druckverfahren	5
3.2 Ender 3 Pro	5
3.3 G-Code	8
3.4 Schrittmotoren	8
4 Hardware	9
4.1 Schema	9
4.2 Leiterplatte	15
4.3 Berechnungen	17
4.4 Modifikationen	25
5 Software	28
5.1 Anforderungen an die Software	28
5.2 Software Strukturierung	28
5.3 G-Code Interpreter	29
5.4 Webinterface	30
5.5 Lizenzen	30
6 Inbetriebnahme	31
6.1 Stabilität der 3.3V Speisung	31
6.2 Stabilität der 5.0V Speisung	31
6.3 Messung des Stromverbrauchs	31
6.4 Verifizierung der Temperaturmessung	31
6.5 MOSFET Schwingverhalten	32
6.6 Anhebung der 24V Speisung beim Bremsen der Schrittmotoren	32
6.7 Messung Übertragungsgeschwindigkeit Wifi zu SD-Karte	33
6.8 Messung problematischer Schrittverluste	33

7 Fazit	36
8 Schlusswort	37
9 Ehrlichkeitserklärung	39
Literatur	40
A Lastenheft	42
B Schema	43

1 Einleitung

Wie die meisten Technologien waren die ersten 3D-Drucker während einer langen Zeit für die Privatanwendung viel zu teuer. Seither machte die Technologie jedoch bedeutende Fortschritte, sodass die Preise kontinuierlich gesunken sind. Dies führte dazu, dass das Angebot an 3D-Druckern für Private während der letzten Jahre regelrecht explodiert ist. Mittlerweile befindet sich der durchschnittliche Preis für Einsteigermodelle bei wenigen Hundert Schweizer Franken. Trotz ihrer weiten Verbreitung existieren jedoch nach wie vor beträchtliche Unterschiede in Qualität und Sicherheit. Dies, da nicht zuletzt viele 3D-Drucker aus günstiger asiatischer Produktion stammen und mit unzureichender Hardware ausgerüstet sind, wie etwa primitive Motortreiber, fehlender Überstromschutz oder unzureichend dimensionierte Bauteile.



Abbildung 1.1: Ruine nach Brand eines 3D-Druckers [1]

Im Projekt 4 des Studiengangs Elektro- und Informationstechnik ist es deshalb das Ziel, zu untersuchen, inwiefern ein günstiger 3D-Drucker kosteneffizient modifiziert werden kann, damit er seine Funktion mit erhöhter Druckqualität und Benutzerfreundlichkeit wahrnehmen kann. Danach sollte es unter anderem möglich sein, zu druckende Dateien drahtlos zu übertragen und bessere Motortreiber zu verwenden. Die wichtigsten Ziele dieses Projektes waren das erfolgreiche Ansteuern eines Schrittmotors, die Druckdateien mittels SD-Karte oder Wifi auf den Drucker zu laden und einen Testdruck erfolgreich durchzuführen.

Für den gewählten 3D-Drucker *Ender 3 Pro* der Firma *Creality* wurde eine neue Leiterplatte zur Steuerung entwickelt. Diese soll im Vergleich mit dem Standardprodukt eine feinere und somit leisere Ansteuerung der Schrittmotoren und eine drahtlose Übertragung von zu druckenden Dateien erlauben. Weiterhin wurden mehrere Bauteile verbaut, welche beispielsweise mittels eines BLTouch Bed-Level-Sensors eine automatische Ausrichtung des Druckkopfes an der Heizplatte ermöglichen.

In den folgenden Kapiteln sind alle Schritte dokumentiert, welche für die Hardware- und Softwareentwicklung nötig sind. Auch sind die wichtigsten Messungen bei der Inbetriebnahme, sowie deren Ergebnisse aufgeführt. Abschliessend fasst ein Schlusswort die Projektarbeit zusammen.

2 Gesamtübersicht

Im diesem Kapitel folgt eine Übersicht über die Anforderungen, die daraus entstandenen Kriterien und das entworfene Grobkonzept. Vertiefte Beschreibungen der Hardware und der implementierten Software sind im Kapitel 4 und 5 zu finden.

2.1 Anforderungen

Die Basis dieses Projekts bildet, dass zu Beginn zusammen mit dem Auftraggeber verfasste Pflichtenheft. Die wichtigsten Punkte werden kurz erläutert, wobei das Lastenheft im Anhang A zu finden ist.

Der Auftraggeber verlangt eine verbesserte und funktionsfähige Steuerung für einen 3D-Drucker. Im Lastenheft handelt es sich um einen 3D-Drucker des Typs *K8200* der Firma *Velleman*. Nach Vereinbarung wurde ein Wechsel zu einem *Ender 3 Pro* vollzogen. Die Elektronik ist auf einer Leiterplatte mit allen Anschlüssen für die Aktoren und Sensoren des Druckers zu implementieren. Funktionen wie die Ansteuerung von Schrittmotoren für die Bewegung in allen drei Dimensionen, Regelung der Heizbetttemperatur oder der Extruderheizung sowie drahtlosen Datentransfer für den Druckauftrag sind massgebende Anforderungen. Die entworfene Leiterplattensteuerung soll fest im 3D-Drucker verbaut werden. Zur Verwaltung von Druckaufträgen und zur Beobachtung deren sauberen Abarbeitung soll ein PC-Zugriff möglich sein. Die Aufgabenstellung sieht ausserdem vor, dass die Endschalter durch eine einfachere Variante ersetzt werden. Als Wunschziel kann eine Nachlauf-Erkennung fürs Filament, ein Power-Save Modus oder eine Anzeige des aktuellen Druckfortschritts implementiert werden.

2.2 Kriterien

Aus dem Pflichtenheft sind die untenstehenden Ziele beziehungsweise Kriterien zusammenfassend aufgelistet. Im Pflichtenheft für den Auftraggeber sind diese Kriterien in Form von Pflicht- und Wunschzielen umformuliert. Dabei ist zu beachten, dass nicht alle Wunschziele erreicht wurden und diese hier nicht aufgelistet sind:

- Eine unabhängige Ansteuerung aller vier Schrittmotoren (x-, y-, z-Achse und Zuführung Filament) ist ohne Schrittverlust möglich.
- Die Nullpositions-Erkennung der x-, y- und z-Achse weist mittels Endschalter mindestens $\pm 1\text{mm}$ Genauigkeit auf.
- Die Temperaturen von Extruder und Heizbett werden auf eine Genauigkeit von $\pm 5^\circ\text{C}$ erfasst.
- Die Regelung für die Heizleistung von Extruder und Heizbett stellt den Istwert korrekt ein.
- Drei Ventilatoren dienen der Kühlung der frisch ausgedruckten Strukturen, des Extruders und der Leiterplatte selbst.
- G-Code wird via SD-Karte und Wifi geladen.
- Die Bedienelementen ermöglichen es, mit dem Drucker zu interagieren. Diese sind in Form von LEDs, Encoder und LCD implementiert.
- Alle Achsen und Sensoren können manuell angesteuert werden.
- Ein mechanischer Aufbau erkennt das Ende der Filamentrolle.
- Zur Verwaltung der Druckaufträge dient eine Webseite mit einer Benutzeroberfläche.
- Ein Sensor ermöglicht es, die z-Achse automatisch zu nivellieren.

2.3 Lösungskonzept

In Abbildung 2.1 ist das Blockschaltbild ersichtlich, welches alle Teilsysteme und Einheiten des Konzepts darstellt. Es ist modular gegliedert und bietet eine Übersicht der Schnittstellen zwischen den einzelnen Modulen. Das Lösungskonzept selbst gliedert sich in drei physikalisch getrennte Einheiten. Das Machine Controll System (*MCS*), das Human Maschine System (*HMS*) und den Drucker selbst, einen *Ender 3 Pro* der Firma *Creality*. Das *MCS* bildet den Hauptbestandteil des Lösungskonzepts. Es beinhaltet die Druckersteuerung und die Kommunikationsschnittstellen. Um diese zwei Aufgaben zu unterteilen befinden sich zwei Mikrocontroller in dieser Einheit. Zusätzlich werden die Aktoren und Sensoren des 3D-Druckers angesteuert und eingelesen. Eine Statusanzeige in Form eines LCD zur Zustandsvisualisierung ist ebenfalls angedacht. Das *HMS* bildet die Benutzerschnittstelle über ein internetfähiges Gerät. Es kommuniziert mittels Wifi mit dem *MCS*. Über dieses Gerät können Druckaufträge verwaltet und überwacht werden. Auf dem 3D-Drucker befinden sich alle Aktoren und Sensoren. Sie werden über Kabelverbindungen an das *MCS* angebunden.

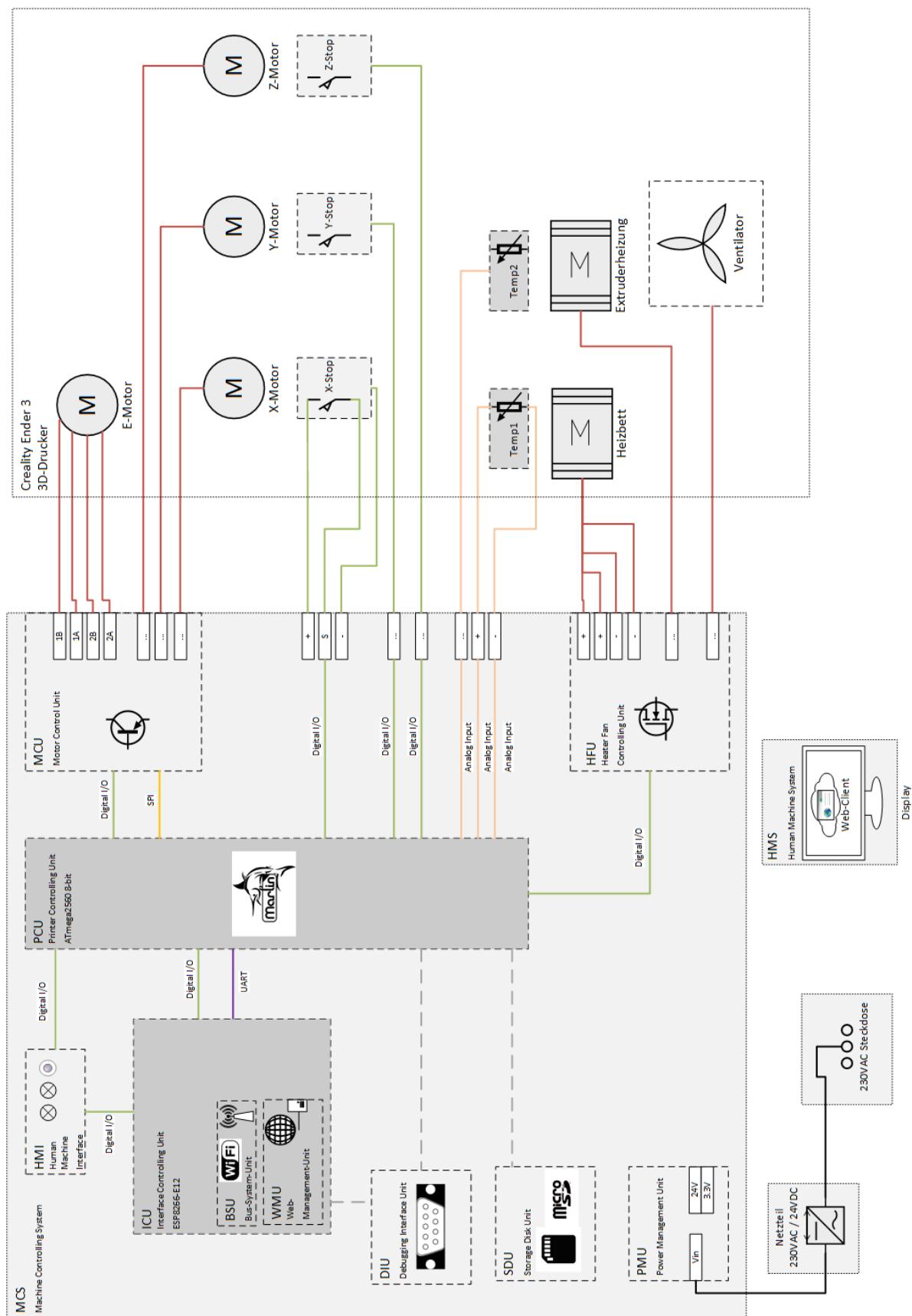


Abbildung 2.1: Blockschaltbild des Lösungskonzepts

3 Technische Grundlagen

In diesem Kapitel werden grundlegende Konzepte und Technologien, die zum Verständnis dieses Berichts beitragen, erklärt. Weiter wird der verwendete 3D-Drucker *Ender 3 Pro* vorgestellt und begründet, weshalb nicht der im Lastenheft angedachte *Velleman K8200* verwendet wird.

3.1 Druckverfahren

Beim Druckverfahren vom *Ender 3* handelt es sich um das sogenannte *Fused Deposition Modeling*. Bei diesem wird ein Kunststofffilament durch Erwärmung geschmolzen und durch eine feine Düse gepresst. Der extrudierte Kunststoff verschmilzt mit dem angrenzenden Kunststoff und bildet mit diesem einen soliden Körper (siehe Abbildung 3.1).

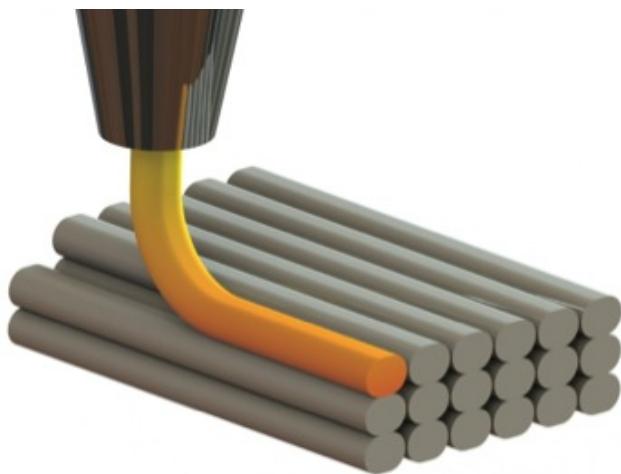


Abbildung 3.1: Fused Deposition Modeling [2]

Bevor der Druckvorgang gestartet werden kann, muss der 3D-Drucker mit Instruktionen versorgt werden, die ihm vorschreiben, wie das gewünschte Teil gefertigt werden soll. Zu diesem Zweck wird ein 3D-Modell des Werkstücks in eine Slicer-Software geladen. Diese „schneidet“ das Modell in horizontale Scheiben mit festlegbarer Dicke (Layer) und berechnet einen Pfad, welchen die Düse des 3D-Druckers befolgen muss, um einen solchen Layer zu generieren. Daraus wird eine G-Code-Datei (siehe Kapitel 3.3), eine Liste von Befehlen, erstellt. Diese beinhaltet wie sich die Motoren verhalten müssen, damit der errechnete Pfad eingehalten wird. Weitere Befehle, die das Verhalten von Ventilatoren und Heizungen definieren, werden ebenfalls durch die Slicer-Software erstellt. Diese Parameter sind in der Software einstellbar. Die gesamte G-Code-Datei ist durch den 3D-Drucker interpretierbar und kann somit an ihn übergeben werden, worauf dieser mit dem Druckvorgang beginnen kann.

3.2 Ender 3 Pro

Die Basis des Projekts bildet, wie schon in Kapitel 1 erwähnt, der *Ender 3 Pro* 3D-Drucker von *Creality* (siehe Abbildung 3.2). Der grosse Vorteil dieses Druckers ist das ausgewogene Preis-Leistungsverhältnis. Der Drucker ist von Werk aus vormontiert und somit innert Kürze aufgebaut und betriebsbereit. Der Laie kann ohne grosse Arbeit und grosses Vorwissen bereits nach kurzer Zeit erste Druckaufträge ausführen. Ein wichtiger Unterschied zwischen dem *Ender 3 Pro* und dem im Lastenheft erwähnten *Velleman K8200* ist der Aufbau des Druckers. Der gewählte

Drucker besteht aus sogenannten X-Profilen, welche aus Aluminium gefertigt sind. Diese sorgen für ein ausreichend stabiles Grundgerüst, welches für einen sauberen Druck entscheidend ist. Außerdem kann eine höhere Druckgeschwindigkeit verwendet werden, wenn der Rahmen genügend Stabilität bietet. Der Drucker verfügt über drei Achsen (x, y und z). Diese bestehen aus kugelgelagerten Linearführungen, welche eine saubere Führung des Druckbetts und des Druckkopfs garantieren. Der Extrudermotor ist beim *Ender 3 Pro* am Rahmen befestigt und nicht wie bei anderen Modellen am Druckkopf selbst. Dies bietet den Vorteil, dass der Druckkopf bei Bewegungen kein zusätzliches Gewicht beschleunigen oder abbremsen muss. Daher wird nebst einer höheren Druckgeschwindigkeit eine höhere Druckqualität ermöglicht. Um eine saubere Führung des Filaments vom Motor zur Düse gewährleisten zu können, wird dieses durch ein Teflonrohr geführt. Der Druckkopf verfügt über zwei Ventilatoren. Der erste sorgt dafür, dass die Filamentzufuhr, der sogenannte Throat, gekühlt wird. Der zweite Ventilator ist dafür zuständig, dass das bereits ausgedruckte Modell nach der Aufnahme des neuen Filaments ordnungsgemäß gekühlt wird, was besonders bei kleinen und schmalen Baustücken wichtig ist. Die maximale Druckhöhe beträgt 250mm, wobei das Druckbett einen Haftgrund von 220mm·220mm aufweist. Dieses Druckbett kann bis zu einer Temperatur von 110°C geheizt werden. Dadurch können nicht nur Kunststoffe wie *PLA*¹, sondern auch *PETG* ² oder *ABS* ³ für den 3D-Druck verwendet werden. Das Druckbett verfügt außerdem über Nivelierräder, welche eine manuelle Ausrichtung des Heizbettes vereinfachen. Ein seitlich angebrachtes Display sorgt mittels Encoder für eine leichte Bedienung. Für die notwendige Energie sorgt ein 360W Netzteil [6].

¹**Polylactide (kurz PLA)** sind synthetische Polymere, die zu den Polyester gehören. Aus ihnen wird Kunststoff gefertigt, der aus regenerativen Quellen gewonnen wird (wie beispielsweise Maisstärke). PLA ist ein sehr verbreiteter Kunststoff [3].

²**PETG** ist ein mit Glykol modifiziertes PET und gehört zu den thermoplastischen Kunststoff [4].

³**ABS** ist ein synthetisches Polymer, dass aus den einzelnen Monomeren Acrylnitril, 1,3 Butadien und Styrol hergestellt wird. ABS ist einer der meistverbreiteten Kunststoffe der Welt [5].

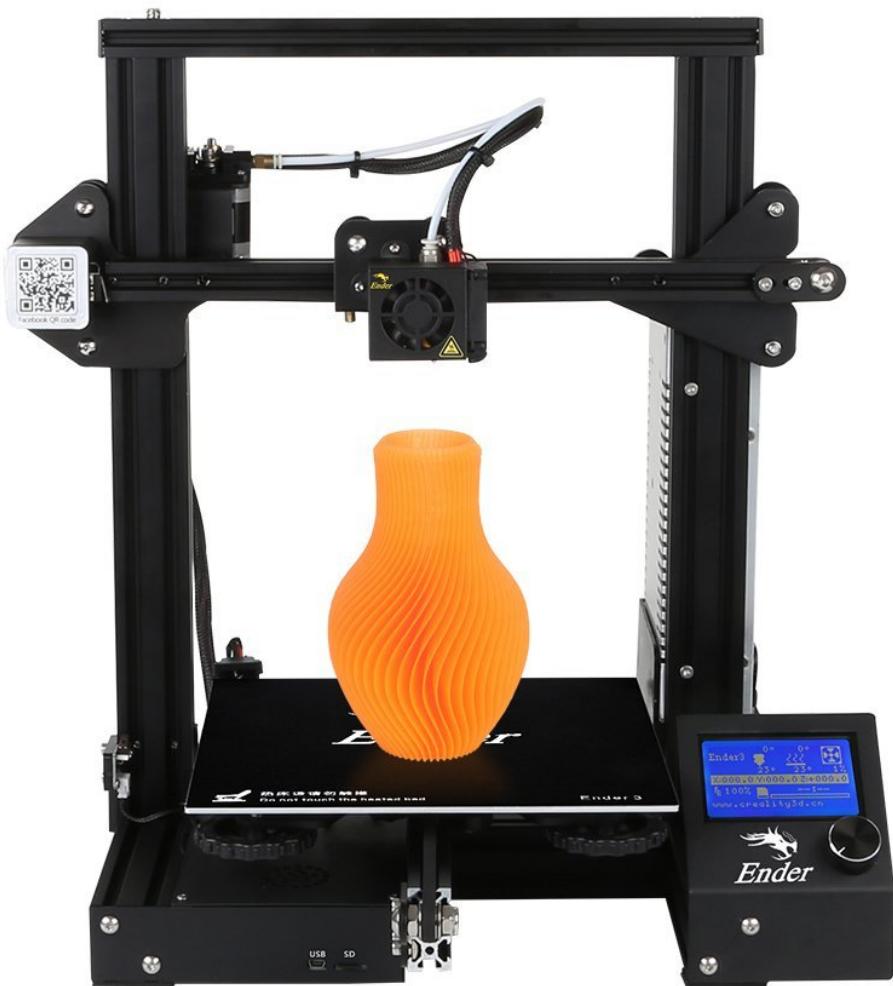


Abbildung 3.2: Verwendeter 3D-Drucker *Ender 3 Pro* [6]

3.3 G-Code

G-Code ist eine Maschinensprache welche in CNC-Maschinen eingesetzt wird. CNC steht dabei für "Computerized Numerical Control". Eine solche Maschine kann zum Beispiel eine Fräsmaschine oder wie in diesem Projekt ein 3D-Drucker sein. Der *G-Code* wird in Form einer Liste mit verschiedenen Befehlen der Maschine übergeben welche diese entsprechend umsetzt. Dieser *G-Code* wird bei einem 3D-Drucker, wie in Kapitel 3.1 beschrieben, automatisch durch das verwendete Slicer-Programm erzeugt. Diese Schichten werden dann durch die Interpretation der Befehle gedruckt. Bei 3D-Druckern sind die Befehle begrenzter als bei sonstigen CNC-Maschinen, da nur der Extruder als Werkzeug vorhanden ist. Die Abkürzung *G-Code* steht dabei für "Geometric Code" [7].

Nachfolgende Befehlszeile könnte in einer G-Code Datei für einen 3D-Druck stehen:

G1 X30 E10 F1800

Diese sagt dem Gerät es soll mit einer Geschwindigkeit von 1800mm/min zur x-Koordinate 30 fahren und 10mm Filament in den Extruder drücken [8].

3.4 Schrittmotoren

Schrittmotoren sind von der Arbeitsweise her anders aufgebaut wie normale Gleichstrommotoren, da diese mehr als nur eine konstante Spannung verlangen, um zu funktionieren. Schrittmotoren haben in ihrem Innern einen Rotor, welcher das drehbare Element des Motors darstellt, und Statorspulen. Werden diese Statorspulen nach einem bestimmten Muster angesteuert, wird der Rotor in Bewegung gesetzt. Dieser Aufbau hat den Vorteil, dass eine Umdrehung des Rotors in viele Schritte unterteilt werden kann. Die hohe Auflösung einer Umdrehung ermöglicht einem 3D-Drucker sehr genau zu arbeiten, eine äusserst wichtige Eigenschaft [9].

Um einen Schrittmotor mit einem Mikrocontroller steuern zu können, wird ein Treiber benötigt. Dieser ist in der Lage die notwendige Leistung zu liefern. Sie sind als fertige Bauteile erhältlich und haben oft noch zusätzliche Funktionen. Es gibt Motortreiber, welche Beispielsweise ein Blockieren des Motors detektieren und an den Mikrocontroller melden können. Andere können sogenanntes „micro-stepping“ betreiben, welches noch genauere Schritte erlaubt als der eigentliche Schrittmotor vorgibt [10].

4 Hardware

In diesem Kapitel wird vertieft auf die hardwarespezifischen Aspekte eingegangen. Es werden wichtige Bestandteile der Schaltung und der Leiterplatte dokumentiert und die zugrundeliegenden Berechnungen erläutert. Weiterhin sind die durchgeführten Modifikationen, die nicht Bestandteil der Leiterplatte oder des Bausatzes des 3D-Druckers sind, dokumentiert.

4.1 Schema

In diesem Kapitel wird nur auf ausgewählte Bestandteile des Schemas eingegangen. Das vollständige Schema ist im Anhang B zu finden.

Spannungsversorgung

Die 24V Spannungsversorgung, welche von einem externen Kabelnetzteil stammt, wird auf der Leiterplatte über 3x $1000\mu F$ Kondensatoren geführt. Dies wird gemacht um ein allfälliges anheben des Spannungspiegels beim Bremsen der Schrittmotoren zu verhindern (je nach Einstellung in der Firmware wird beim Bremsen Energie zurück ins 24V Netz gespeist). Benötigt werden die 24V um die Motoren, das Heizbett, die Extruderheizung sowie die Ventilatoren zu versorgen.

Die 24V werden mittels zwei *MCP16311* DC/DC-Wandler der Firma *Mircochip Technology* auf der Leiterplatte in je eine 5V und eine 3.3V Speisung gewandelt. Um die Verlustleistung in den Wendlern klein zu halten wurden bewusst DC/DC-Wandler und keine Linearregler verwendet, sowie darauf geachtet das der Drahtwiderstand der Spule $L1$ bzw. $L3$ $R_{L_{DCR}}$ mit $71m\Omega$ bzw. $120m\Omega$ klein ist. Außerdem wurde darauf geachtet das der Wirkungsgrad η beim zu erwartenden Ausgangsstrom I_{out} sicher über 80% liegt (siehe Abbildung 4.11) und der maximale Ausgangsstrom des DC/DC-Wandlers sicher genügend hoch ist (liefert max. $\sim 1.2A$, benötigt wird max. $300mA$, siehe Tabelle 4.1).

Aus Gründen der EMV wurden geschirmte Spulen verwendet, sowie am Ausgang ein LC-Tiefpassfilter implementiert (Dimensionierung siehe Abschnitt 4.3).

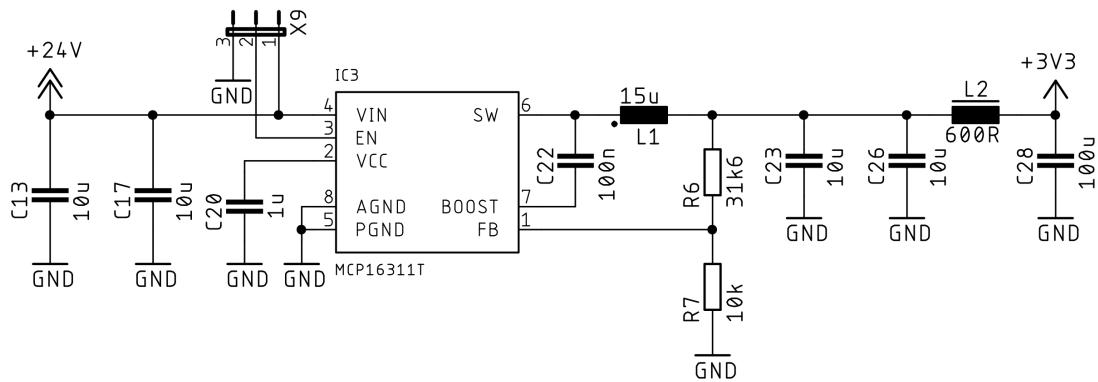


Abbildung 4.1: DC/DC-Wandler *MCP16311* sowie dessen Beschaltung

Mikrocontroller

Als Mikrocontroller wird ein *ATmega2560* der Firma *Mircochip Technology* verwendet. Dies aus dem Grund, dass die Firmware *Marlin* auf diesem Controller sicher funktioniert bzw. dafür programmiert wurde. Programmiert wird der Mikrocontroller über die herstellerseitig vorgesehene *ISP* Schnittstelle.

Um auch höhere Baudaten mit der *UART* Schnittstelle verwenden zu können, wurde ein externer $8MHz$ Quarz mit $\pm 50ppm$ Genauigkeit *ABM3B* verbaut (siehe Abbildung 4.2). Damit die Temperaturmessungen der NTC-Thermistoren über eine brauchbare Genauigkeit verfügen wurde die externe $2.048V$ Spannungsreferenz *IC1* mit $\pm 0.08\%$ Genauigkeit *MAX6070* verbaut (die interne Referenz des Mikrocontrollers hat $\pm 10\%$ Genauigkeit [11]). Dabei ist der *OUTF*-Ausgang des *IC1* möglichst nahe am *AREF*-Eingang des Mikrocontrollers mit dem *OUTS*-Ausgang verbunden, um den Spannungsverlust über den Zuführleitungen auszuregeln, bzw. zu minimieren. Für die Berechnung der Lastkapazität, siehe Abschnitt 4.3.

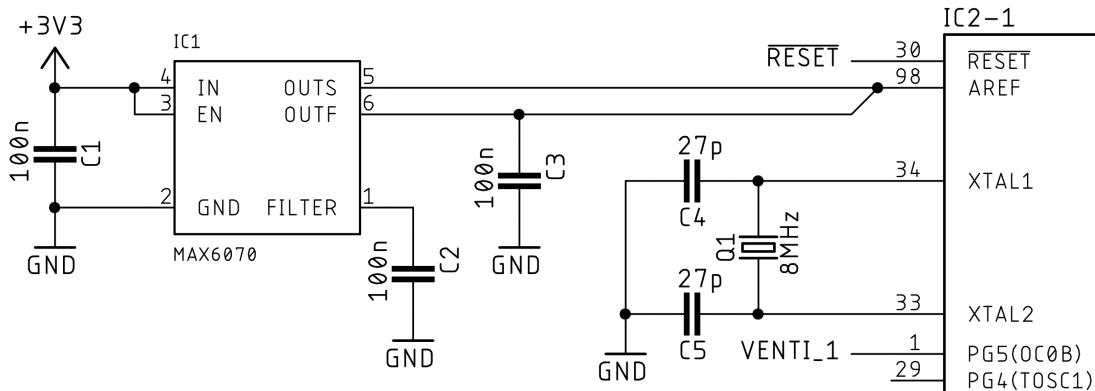


Abbildung 4.2: Mikrocontroller *IC2* mit externer Beschaltung (Quarz *Q1* sowie die externe Spannungsreferenz *IC1*)

Motorsteuerung

Für die Steuerung der Motoren wird ein *TMC2130* Motorcontroller der Firma *Trinamic* eingesetzt. Dieser hat den Vorteil, dass er in der Firmware *Marlin* bereits implementiert ist. Es werden vier dieser Treiber eingesetzt. Einen für den Extrudermotor und je einen für den x-, y- und z-Achsenmotor. Gespeist wird der Treiber mit $3.3V$ für die Kommunikation mit dem Mikrocontroller (*VCC_IO*), $24V$ um die Motoren zu steuern und $5V$ für die interne Logik. Um die Verlustleistung zu minimieren wird darauf verzichtet den internen Linearregler zu verwenden. Stattdessen werden externe $5V$ verwendet. Außerdem sind die Schottkydiode *D10 - D13* verbaut um die interne Verlustleistung bei höheren Strömen während der negativen Halbwelle der Ansteuerung zu verringern. Um den Treiber via Mikrocontroller zu steuern, gibt es die Anschlüsse *STEP* (Drehzahl des Motors), *DIR* (Drehrichtung des Motors), *!DRV_EN* (Aktivierung der Motorausgänge am Controller, active low), *DIAG0* (Alarm beim Blockieren des Motors, low active, open drain) sowie eine *SPI* Schnittstelle um Werte aus Registern zu laden, bzw. hineinzuschreiben. Zudem erlaubt der verwendete Motorcontroller micro-stepping, womit die Auflösung von 200 Schritten pro Umdrehung des Motors auf 3200 Schritte erhöht wird.

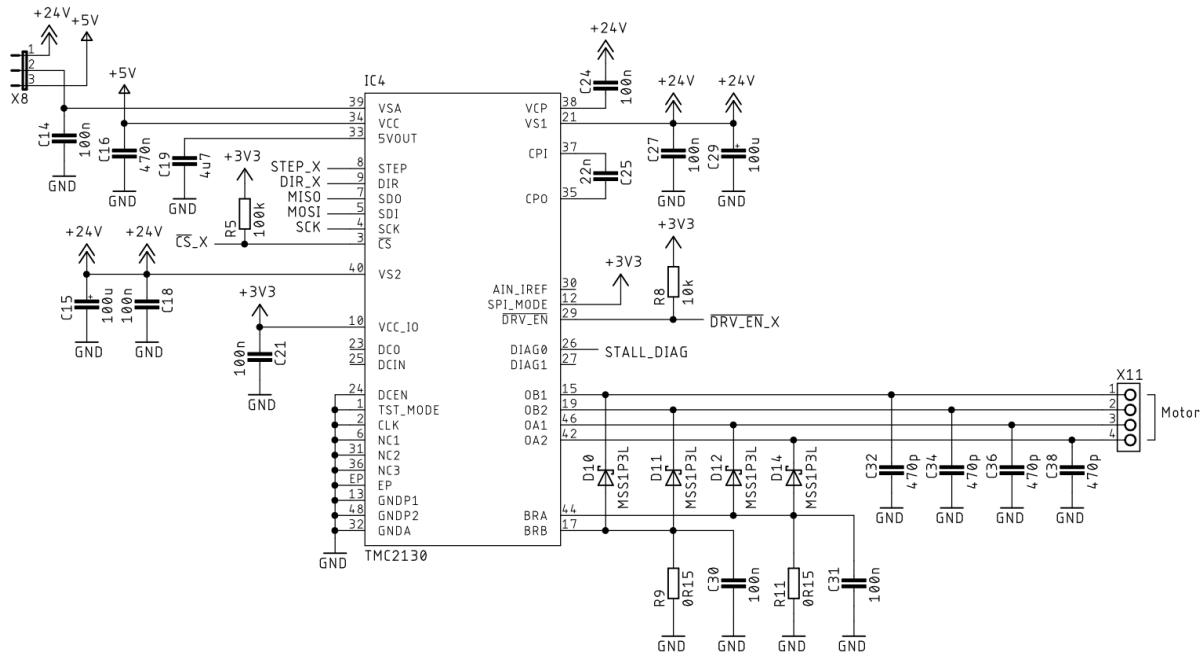


Abbildung 4.3: Motorcontroller $IC4$ welcher die x-Achse steuert mit externer Beschaltung

Ventilatorsteuerung

Die drei zur Kühlung verwendeten Ventilatoren werden durch MOSFETs des Typs *IRLR8726* angesteuert (siehe Abbildung 4.4). Bei der Selektion des MOSFET-Typs wurde darauf geachtet das die Spannung $V_{gs(th)}$ beim zu erwartenden Drain-Strom $I_D = 1A$ über 3.3V hoch ist, da ansonst der MOSFET nicht durchgesteuert werden könnte (siehe Abbildung 4.5). Der Wert für I_D war während der Dimensionierung geschätzt anhand von ähnlichen Ventilator Modellen, da zu diesem Zeitpunkt der 3D-Drucker noch nicht für Strommessungen zur Verfügung stand. Ausserdem ist die Drain-Source-Spannung V_{dss} relevant ($> 24V$) sowie der $R_{DS(on)}$ ($\sim 8m\Omega$) soll möglichst tief sein um die Eigenerwärmung klein zu halten (für Berechnung siehe Abschnitt 4.3).

Die MOSFETs selbst werden durch *PWM* Ausgänge am Mikrocontroller angesteuert, wobei eine grüne Status-LED leuchtet wenn der Ventilator in Betrieb ist (50% Duty Cycle entsprechen derselben Helligkeit wie die Status-LEDs z.B. der Endschalter, sodass Änderungen sichtbar sind).

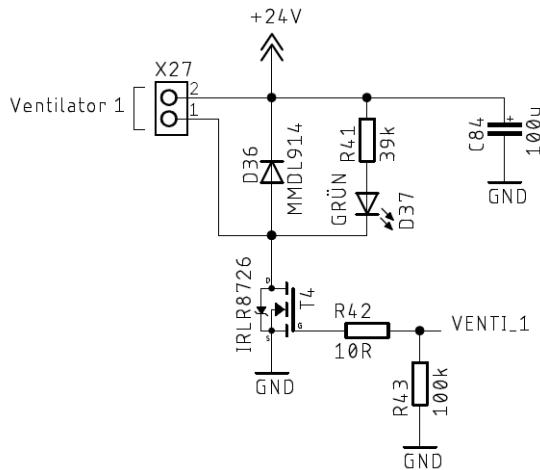


Abbildung 4.4: Ventilatorsteuerung mit grüner Status-LED, Freilaufdiode um die Spannung der Induktiven Last zu begrenzen und Gatewiderstand um Schwingvorgängen vorzubeugen

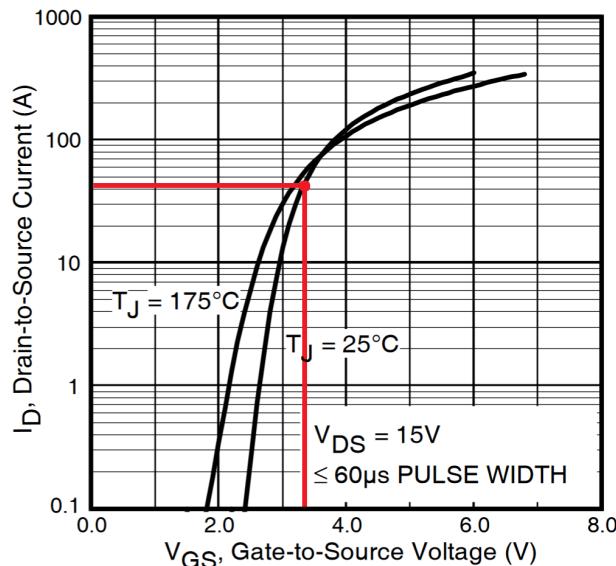


Abbildung 4.5: $V_{gs(th)}$ vs. Drain-Strom der MOSFETs T3, T4 und T5 ??

Heizbett- und -Extrudersteuerung

Die Dimensionierung der Heizbett- und Extrudersteuerung verhält sich analog zur Ventilatorsteuerung, mit Ausnahme dass höhere Ströme geschaltet werden (Heizbett: $I_D \approx 9.2A$, Ventilatoren: $I_D \approx 1.0A$ (eher weniger, Wert war zur Zeit der Dimensionierung nicht genau bekannt)). Daher wurde zusätzlich ein kleinerer $R_{DS(on)}$ ($\sim 4m\Omega$) sowie ein kleiner Temperaturübergangswiderstand T_j zu T_A ($62^{\circ}C/W$) geachtet. Da sich in der Vergangenheit gezeigt hat, dass beim Ablösen des fertigen Modells mittels eines Spachtels vom Heizbett öfters Kurzschlüsse durch Beschädigungen des Heizbettes entstehen, ist aus Sicherheitsgründen eine 15A Flink Sicherung verbaut.

Abbildung 4.5 was sollen ?? im Abbildungsbeschrieb?

Da sich in der Vergangenheit?
Ich würde gerne mal sehen wie man

Temperaturmessung

Um alle drei Ventilatoren regeln zu können, muss die Temperatur am entsprechenden Ort gemessen werden (Kühlung Extruder, Kühlung des frisch gedruckten Filaments und Kühlung Leiterplatte, direkt an den Motorcontroller, wo es am heissten wird). Dazu werden $100k\Omega$ NTC-Thermistoren verwendet, welche via internem 10-Bit-AD-Wandler des Mikrocontrollers gemessen werden (für die Dimensionierung und Berechnung siehe Abschnitt 4.3).

Display, Summer und Encoder

Um das vom 3D-Drucker *Ender 3 Pro* standardmäßig verbaute Bedienkonzept (Display, Summer und Encoder zur Bedienung) verwenden zu können, wird die mitgelieferte Leiterplatte angesteuert. Dazu sind die *SPI*-Schnittstelle, sowie Leitungen für den Summer und den Encoder auf Schraubklemmen X1 und X2 herausgeführt. Zum Schutz sind TVS-Dioden an allen Leitungen abgebracht auf welche von aussen zugegriffen werden können (siehe Abbildung 4.6).

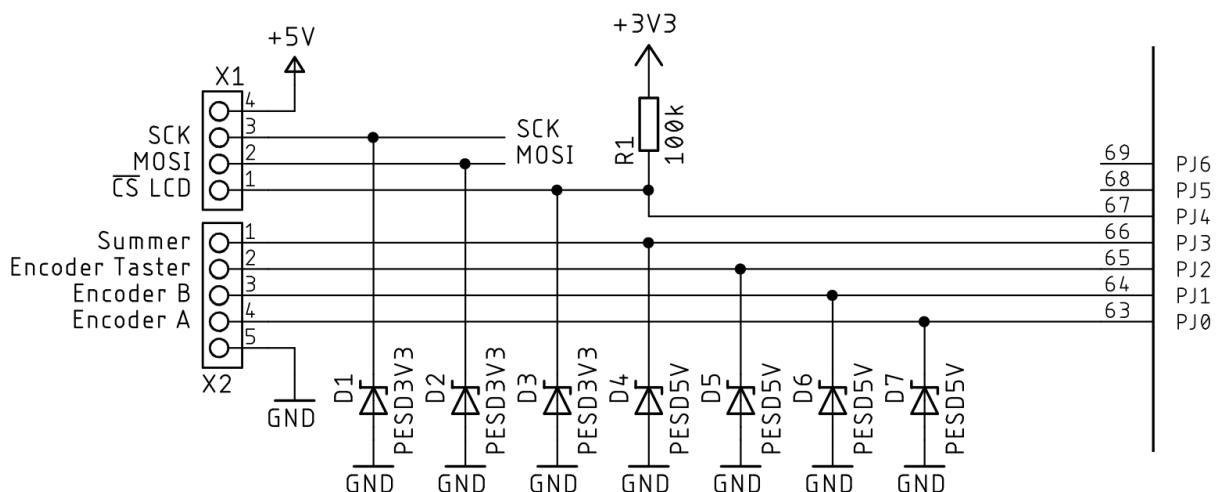


Abbildung 4.6: Anbindung der *SPI*-Schnittstelle sowie der Leitungen für den Summer und den Encoder am Mikrocontroller

SD-Karte

Die SD-Karte dient grundsätzlich zur Speicherung eines Druckauftrags. Vom Mikrocontroller wird auf die gespeicherten Daten zugegriffen, welche dann ausgeführt werden. Der Druckauftrag, welcher über Wi-Fi dem Drucker zugesendet wird, soll zudem auf der SD-Karte abgespeichert werden können.

Schraubklemmen wurden alle (ausser Heizbett und extruder) mit steckern ersetzt!

USB-Schnittstelle

Für die *USB*-Kommunikation wird eine *FT232R USB/UART*-Bridge der Firma *FTDI* verwendet, wobei der *UART*-Teil am Mikrocontroller angehängt ist. Die *USB*-Schnittstelle dient neben Wi-Fi als Backup-Variante, um Zugriff auf den 3D-Drucker zu erhalten, sowie zusätzlich um zu Debuggen während des Entwicklungsprozesses. Dabei erfolgt die Spannungsversorgung direkt über die externen +5V der *USB*-Schnittstelle, welche über einen (Ferrit)-Tiefpassfilter gefiltert wird (siehe Abbildung 4.7).

Was ist ein Eisen Tiefpassfilter?

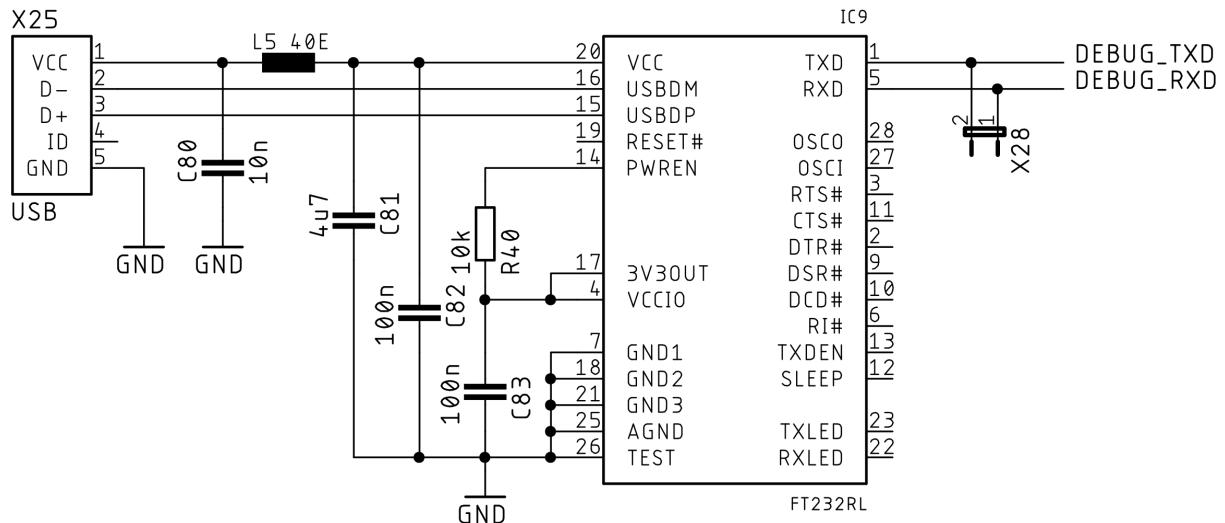


Abbildung 4.7: USB-Schnittstelle mit *USB/UART*-Bridge und (Ferrit)-Tiefpassfilter in der externen +5V Speisungsleitung (mit +5V Speisung wird kein externer Quarz benötigt)

Wi-Fi-Modul

Das WiFi-Modul repräsentiert die primäre Schnittstelle zwischen dem 3D-Drucker und dem Computer des Benutzers. Die Wahl fiel auf das *ESP826612E* Modul der Firma *Espressif Systems*, da es ohne grossen Aufwand die für 3D-Drucker ausgelegte Firmware *ESP3D* unterstützt, günstig und von kleiner Baugrösse ist. Da die Programmierung des Moduls ausschliesslich über die *UART0* Schnittstelle erfolgen kann, diese jedoch auch verwendet werden muss um die zu transferierenden Daten dem Mikrocontroller zu senden, werden *TXD0* und *RXD0* mittels der Jumper *X18* und *X19* getrennt (siehe Abbildung 4.8). Da damit mit dem Programmiergerät von aussen auf die Pins zugegriffen wird, sind an diesen beiden Pins die TVS-Dioden *D26* und *D27* als ESD-Schutz verbaut. Zusätzlich muss der Pin *IO_0* am *ESP826612E* Modul während des Programmievorgangs von *GND* auf *+3.3V* umgehängt werden können (Jumper *X22*).

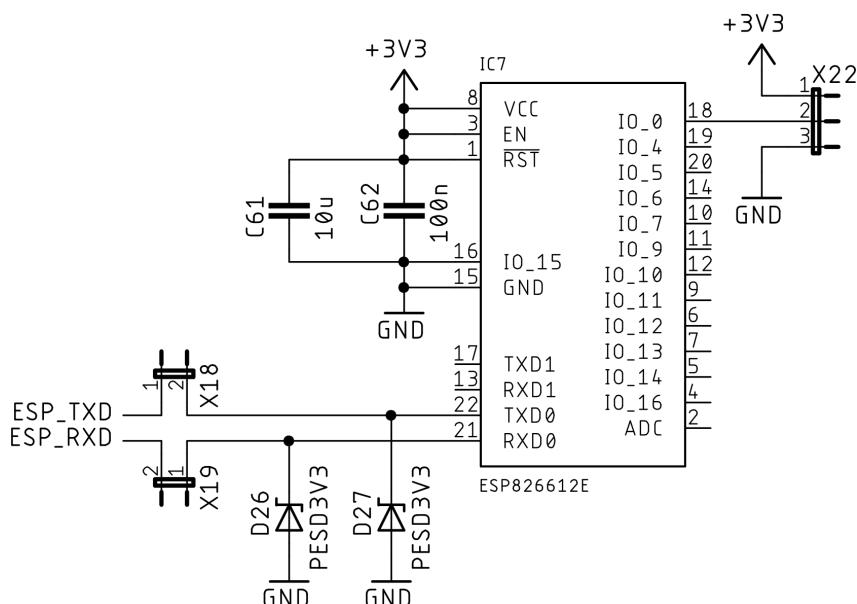


Abbildung 4.8: WiFi-Modul *ESP826612E* mit externer Beschaltung

4.2 Leiterplatte

Das Leiterplattenlayout wurde im ECAD Programm *Eagle* der Firma Autodesk erstellt. Die einzelnen Schaltungsteile wurden nach zweckmässigen Gesichtspunkten wie folgt angeordnet und gelayoutet (siehe Abbildung 4.9):

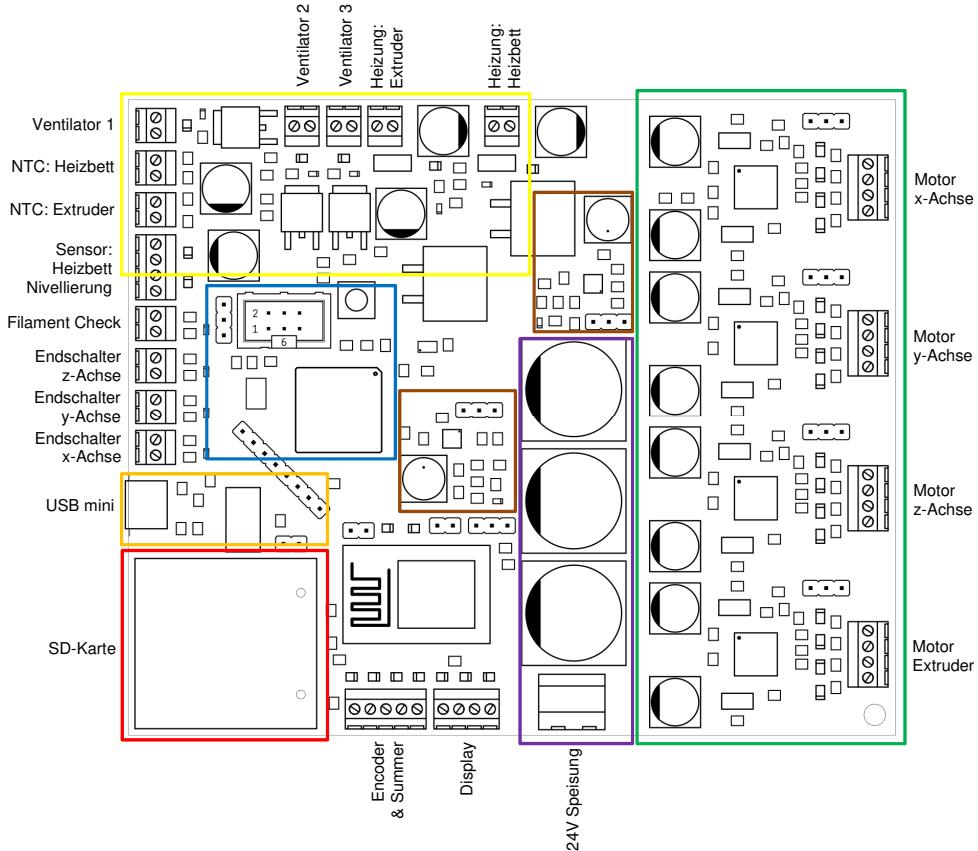


Abbildung 4.9: Anordnung der einzelnen Schaltungsteile

- *3.3V und 5.0V Spannungsversorgung (braun)*: Beide Spannungsversorgungen sind möglichst wie im *recommended design* aus dem Datenblatt gelayoutet, um Störungen und Abstrahlung zu minimieren.
- *Mikrocontroller (blau)*: Platzierung mittig, da der Mikrocontroller alle anderen Schaltungsteile verbindet.
- *Motoransteuerung (grün)*: Platzierung ganz rechts, für eine bessere Übersichtlichkeit.
- *Leistungsteil für die Heizungen sowie Temperaturmessung (gelb)*: Da grosse Ströme fliessen, sind die Kühlflächen sowie die Leiterbahndicken entstprechend gross dimensioniert (siehe Abschnitt 4.3). Außerdem ist ein NTC-Thermistor zwischen der x- und der y-Motorachse verbaut, da dort voraussichtlich die grösste Erwärmung auftritt. Anhand dieser Temperatur wird danach der Lüfter, welcher die Leiterplatte kühlt, geregelt.
- *24V Spannungsversorgung (violett)*: Um ein allfälliges Anheben des 24V Pegels beim gleichzeitigen Bremsen mehrerer Achsen zu verhindern, sind über dem Eingang grosse Kondensatoren verbaut.
- *USB Kommunikation (orange)*: Da der Stecker aus dem Gehäuse herausgeführt werden muss, ist die Position durch die Frontplatte vorgegeben.
- *SD-Karte (rot)*: Da der Slot aus dem Gehäuse herausgeführt werden muss, ist die Position durch die Frontplatte vorgegeben.

Lagenaufbau

Für die Leiterplatte wurde der Lagenaufbau aus Abbildung 4.10 verwendet. Als Oberfläche wurde ENIG gewählt, da diese vollkommen plan ist und so das Reflow-Löten bessere Ergebnisse liefert. Die Leiterplatte wurde mit vier Lagen entworfen, da mit diesen eine kleinere Leiterplattengröße, sowie ein besseres EMV-Verhalten realisierbar ist. Die dadurch verursachte Erhöhung der Kosten ist vernachlässigbar klein.

	Dicke [µm]
Lötstopp	
ENIG	6
Kupfer	35
Prepreg 1080	70
Prepreg 1080	70
GND	
Kupfer	35
Kern	1200
VCC	
Kupfer	35
Prepreg 1080	70
Prepreg 1080	70
Signal	
Kupfer	35
ENIG	6
Lötstopp	
Gesamt	
1632	

Abbildung 4.10: Verwendeter Lagenaufbau

4.3 Berechnungen

Im folgenden Abschnitt werden die wichtigsten Berechnungen, Abschätzungen und Dimensionierungen dokumentiert, welche gemacht wurden um Schema und Layout zu erstellen.

Stromverbrauch der gesamten Schaltung

Für die Dimensionierung der 3.3V sowie der 5.0V Speisung wurde der jeweilige Stromverbrauch anhand der Datenblätter berechnet (siehe Tabelle 4.1 und 4.2). In Rot sind die aufgerundeten Werte angegeben um mögliche Toleranzen zu kompensieren.

Bauteil	Stromverbrauch [mA]
IC1	0.3
IC2	14
X7	75
IC4, IC8, IC10, IC11	0.12
D13, D19, D20, D25, D28, D29	1.95
IC7	170
Total:	262 → 300

Tabelle 4.1: Stromverbrauch im normalen Betrieb der 3.3V Versorgung

Bauteil	Stromverbrauch [mA]
IC4, IC8, IC10, IC11	88
Display	15
Summer	50
Total:	153 → 170

Tabelle 4.2: Stromverbrauch im normalen Betrieb der 5.0V Versorgung

Berechnung des Widerstandsverhältnisses der Speisung

Die Ausgangsspannung U_{out} wird durch das Verhältnis zweier Widerstände eingestellt (R_6 und R_7 für 3.3V bzw. R_{12} und R_{13} für 5.0V). Dabei wird Formel 4.2 verwendet.

$$R_{6/12} = R_{7/13} \cdot \left(\frac{U_{out}}{U_{FB}} - 1 \right) \quad (4.1)$$

U_{FB}	Feedback-Spannung = typ.: 0.800V, min.: 0.784V, max.: 0.816V [12]
U_{out}	Ausgangsspannung = 3.3V bzw. 5.0V

Wird die Toleranz von U_{FB} , sowie die 1% Toleranz der beiden Widerstände mit eingerechnet, ergeben sich als Extremwerte für die Ausgangsspannung die Werte in Tabelle 4.3.

Sättigungsstrom der DC/DC-Wandler Spule L1/L3

Um eine geeignete Spule zu dimensionieren muss der maximale Spitzenstrom I_{LPK} berechnet werden (siehe Formel 4.2). Der Sättigungsstrom der Spule I_{sat} ist, gemäss Hersteller, der Strom

U_{soll} [V]	U_{min} [V]	U_{max} [V]
3.3	3.21	3.45
5.0	4.90	5.28

Tabelle 4.3: Maximale bzw. minimale Toleranz der 3.3V und 5.0V Ausgangsspannungen mit Widerstandswerten der R48 Reihe und 1% Toleranz

bei dem die Induktivität um 30% gesunken ist [13]. Der Sättigungsstrom sollte daher höher gewählt werden.

$$I_{LPK} = \frac{\frac{U_{in}-U_{out}}{L} \cdot t_{on}}{2} + I_{out} \quad (4.2)$$

I_{LPK}	Maximaler Spitzenstrom durch die Spule
U_{in}	Eingangsspannung = 24V
U_{out}	Ausgangsspannung = 3.3V bzw. 5.0V
I_{out}	Ausgangsstrom = 300mA bzw. 170mA
L	Induktivität der Spule L1/L3 = 15 μ H bzw. 22 μ H
t_{on}	Anstiegszeit = 550ns

U_{out} [V]	Berechneter I_{LPK} [A]	Gewählter I_{sat} [A]
3.3	0.68	2.40
5.0	0.48	2.00

Tabelle 4.4: Maximaler Spitzenstrom I_{LPK} vs. gewählter Sättigungsstrom I_{sat} der Spulen L1 und L3 bei beiden DC/DC-Wandler Spannungsversorgungen

Erwärmung der DC/DC-Wandler IC3/IC5

Anhand des in den Tabellen 4.1 und 4.2 abgeschätzten Stromverbrauchs, kann mittels Wirkungsgrad (siehe Abbildung 4.11) die Erwärmung der DC/DC-Wandler *IC3* und *IC5* abgeschätzt werden.

$$P_v = \frac{U_{out} \cdot I_{out}}{\eta} - (U_{out} \cdot I_{out}) - (R_{L_{DCR}} \cdot I_{out}^2) \quad (4.3)$$

P_v	Verlustleistung [W]
U_{out}	Ausgangsspannung = 3.3V bzw. 5.0V
I_{out}	Ausgangsstrom = 300mA bzw. 170mA
η	Wirkungsgrad = 88% bzw. 85%
$R_{L_{DCR}}$	Drahtwiderstand der Spule L1/L3 = 71m Ω bzw. 120m Ω
$R_{thJ/A}$	Wärmeübergangswiderstand: Junction to Air = 211°C/W
ΔT	Temperaturerhöhung [°C]

Mittels der Formeln 4.3 und 4.7 kann somit die zu erwartende maximale Erwärmung des DC/DC-Wandlers abgeschätzt werden (siehe Tabelle 4.5). Da der Summer welcher an der 5.0V Speisung hängt nur kurzzeitig betätigt wird, liegt die Erwärmung bei einem realistischen Nutzungsszenario im Bereich von $\sim 42^\circ C$.

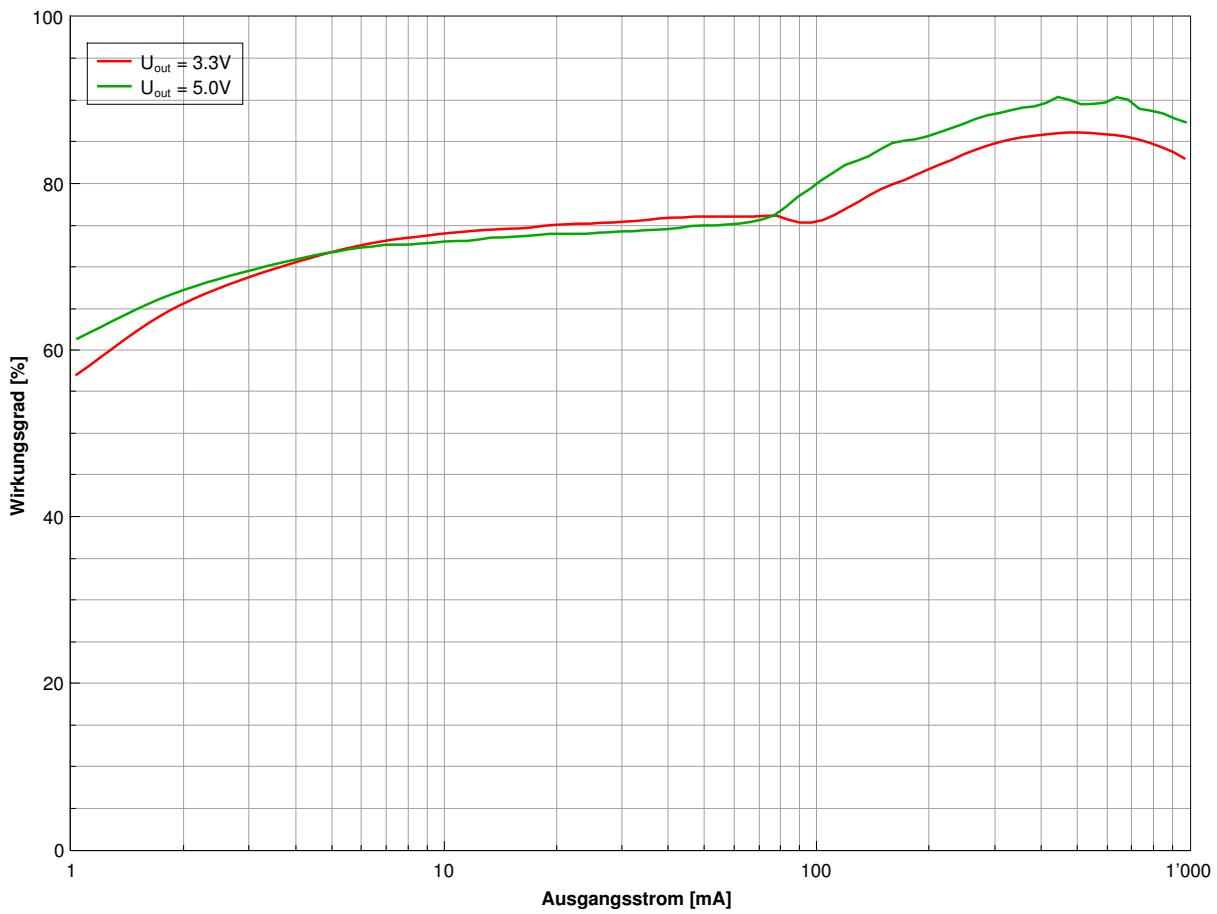


Abbildung 4.11: Wirkungsgrad des DC/DC-Wandlers bei 24V Eingangsspannung. Rot: Wirkungsgrad bei 3.3V Ausgangsspannung. Grün: Wirkungsgrad bei 5.0V Ausgangsspannung. Beim zu erwartenden Laststrom (3.3V) von $\sim 270mA$ ist ein Wirkungsgrad von $\sim 88\%$ zu erwarten. Beim zu erwartenden Laststrom (5.0V) von $\sim 170mA$ ist ein Wirkungsgrad von $\sim 85\%$ zu erwarten.

	$U_{out} = 3.3V$	$U_{out} = 5.0V$
$\Delta T [^{\circ}C]$	~ 36	~ 59

Tabelle 4.5: Die maximale Erwärmung der DC/DC-Wandler *IC3* und *IC5*

Tiefpassfilter Speisungstrennung

Um hochfrequente Störungen zu unterdrücken (insbesondere die Schaltfrequenz des DC/DC-Wandlers) wurde mittels C28 bzw. C44 sowie des Ferrits L2 bzw. L4 ein Tiefpassfilter dimensioniert. Dabei bildet der induktive Anteil des Ferrits mit dem Kondensator einen LC-Tiefpassfilter, während der resistive Anteil des Ferrits störende Signale eliminiert (siehe Abbildung 4.12). Der Amplitudengang des Filters ist in Abbildung 4.13 dargestellt.

Leiterbahndicke im Leistungsteil

Gemäss Herstellerangaben beträgt der maximale Leistungsverbrauch des Heizbettes 220W. Mittels Formel 4.4 kann der fliessende Strom in den Leiterbahnen des Leistungsteils abgeschätzt

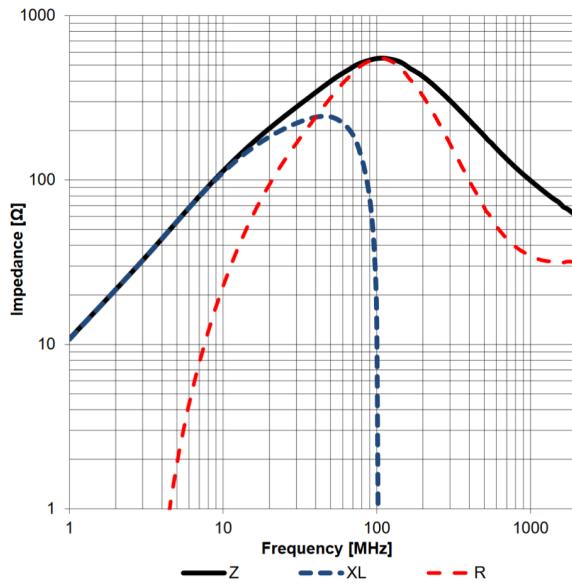


Abbildung 4.12: R, XL und Z Kennlinie des verwendeten Ferrits

werden. Die Leiterbahnen erwärmen sich demzufolge gemäss Formel 4.5. Analog dazu wurde die Erwärmung der Leiterbahnen des Leistungsteils des Extruders abgeschätzt [14].

$$I_{Last} = \frac{220W}{24V} \approx 9.2A \quad (4.4)$$

$$\Delta T \approx \frac{I_{Last}^2}{b^{1.28} \cdot h \cdot k^2} \approx 77^\circ C \quad (4.5)$$

- I_{Last} Laststrom = 9.2 [A]
- k Lagen-Faktor (2 Lagen: K = 3.3, 4 Lagen: K = 3.6)
- b Breite der Leiterbahn = 2 [mm]
- h Höhe der Leiterbahn = 0.035 [mm]
- ΔT Temperaturerhöhung [$^\circ C$]

Da jedoch die meiste Zeit via PWM nur 50% der Leistung auftritt, ist lediglich mit einem ΔT von $\sim 38^\circ C$ zu rechnen.

Erwärmung der MOSFETs von Heizbett, Extruder und Ventilatoren

Gemäss Herstellerangaben beträgt der maximale Strom welcher vom MOSFET des Heizbettes geschaltet wird 9.2A (siehe Formel 4.4). Die Formel 4.6 schätzt die daraus zu erwartende Erwärmung des MOSFETs während der Aufwärmphase ab (MOSFET mehrere Minuten andauernd geöffnet, daher nur P_{stat} und kein P_{dyn}). P_{dyn} im Schalt- bzw. Regelbetrieb ist mit $\sim 4mW$ zu vernachlässigen.

$$\Delta T \approx I_{Last}^2 \cdot R_{DS(on)} \cdot R_{thJ/A} \approx 21^\circ C \quad (4.6)$$

- I_{Last} Laststrom = 9.2 [A]
- $R_{DS(on)}$ Durchgangswiderstand MOSFET = $4m\Omega$

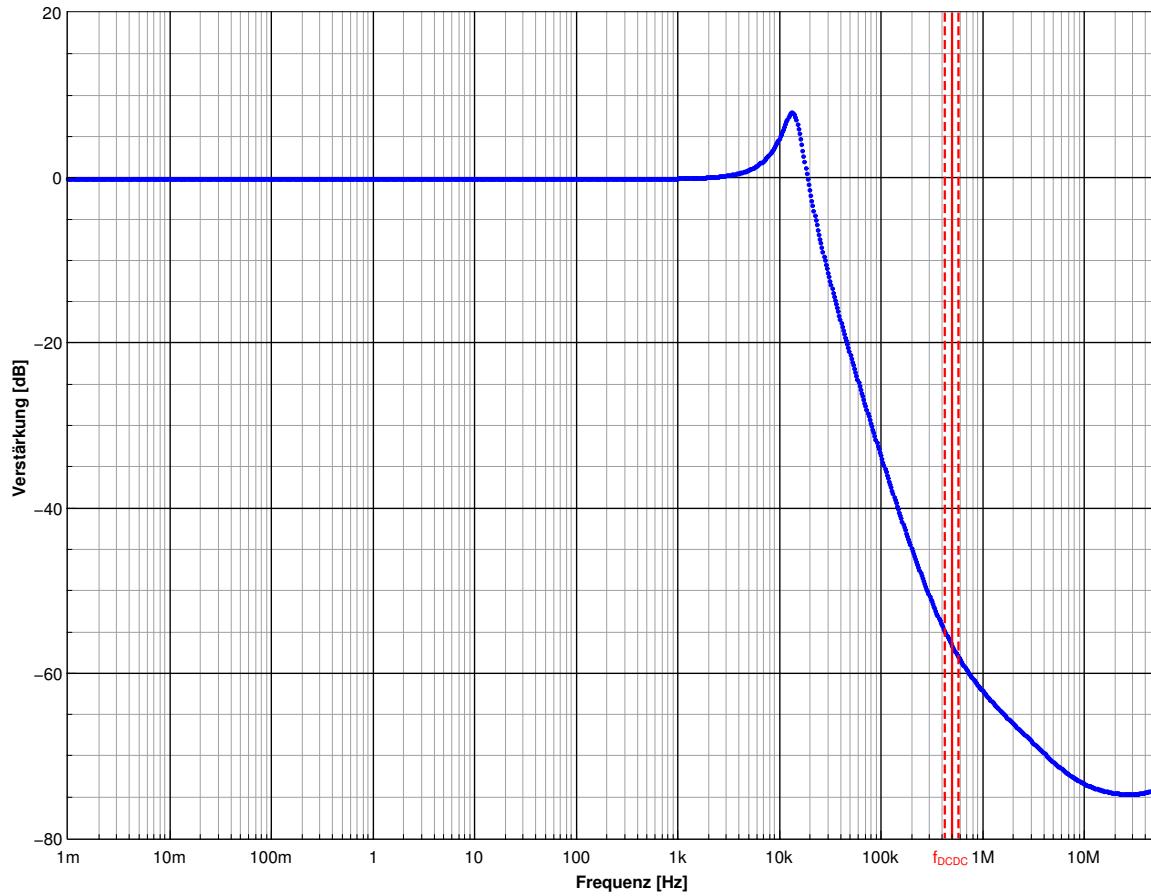


Abbildung 4.13: Blau: Amplitudengang des LC-Tiefpassfilters bei $500mA$ Laststrom, simuliert mit realen Bauteilmodellen des Herstellers. Rot: Schwingfrequenz des DC/DC-Wandlers (min., typ., max.). Bei der typischen Frequenz wird eine Dämpfung von $\sim 57dB$ erreicht.

$$\begin{array}{ll} R_{thJ/A} & \text{Wärmeübergangswiderstand: Junction to Air} = 62^\circ C/W \\ \Delta T & \text{Temperaturerhöhung } [^\circ C] \end{array}$$

Analog dazu wurde die Temperaturerhöhung ΔT der MOSFETs der Ventilatoren und des Extruders auf $\sim 1^\circ C$ abgeschätzt und ist damit zu vernachlässigen.

Erwärmung Motor Controller

Um die aufwendige Berechnung der zu erwartenden Verlustleitung zu vereinfachen, wurde das Excel-Sheet des Herstellers verwendet [15]. Dies errechnet für die verwendeten Parameter eine Verlustleistung von $2.1W$. Daraus errechnet sich die zu erwartende Temperaturerhöhung unter Volllast (siehe Formel 4.7).

$$\Delta T \approx P_v \cdot R_{thJ/A} \approx 44^\circ C \quad (4.7)$$

$$\begin{array}{ll} P_v & \text{Verlustleistung} = 2.1W \\ R_{thJ/A} & \text{Wärmeübergangswiderstand: Junction to Air} = 21^\circ C/W \\ \Delta T & \text{Temperaturerhöhung } [^\circ C] \end{array}$$

Um die Temperatur zusätzlich zu verringern, wurde eine grössere Kühlfläche verwendet als in der Messung des Datenblattes des Wärmeübergangswiderstandes vorgesehen ($\sim 123\text{cm}^2$ anstatt $\sim 93\text{cm}^2$). Zusätzlich wurde das Gehäuse mit einem Kühlkörper versehen auf den ein Ventilator je nach Temperatur Luft bläst.

Kalibrierung der NTC-Temperatursensoren

Da die genauen Charakteristiken der im 3D-Drucker verbauten NTC-Thermistoren unbekannt sind, wurden diese ausgemessen und mit dem entsprechenden Temperaturkoeffizient β kalibriert (siehe Abbildung 4.14). Dabei wurde die Formel 4.8 benutzt.

$$R_T = R_{25} \cdot e^{\beta \cdot \left(\frac{1}{T} - \frac{1}{298.15K} \right)} \quad (4.8)$$

R_T	NTC Widerstandswert bei Temperatur T [Ω]
R_{25}	NTC Widerstandswert bei $298.15K$ ($25^\circ C$) [K]
β	Temperaturkoeffizient []
T	Temperatur des NTC [K]

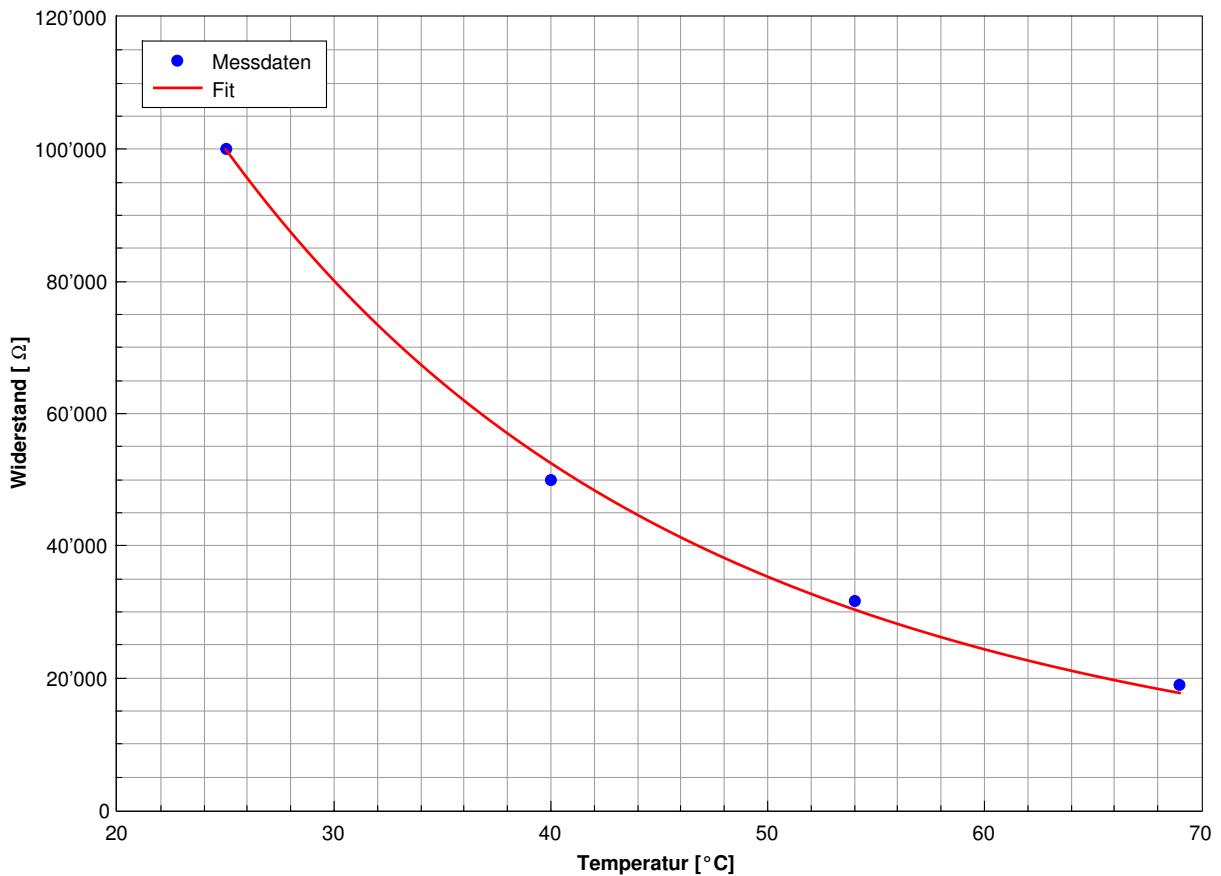


Abbildung 4.14: Fit des Temperaturkoeffizienten der vom Hersteller verbauten NTC-Thermistoren.

Der daraus entnommene Temperaturkoeffizient beträgt: $\beta \approx 4006 \rightarrow 4000$

Vorwiderstand Temperaturmessung

Da die internen 10 Bit A/D-Wandler des ATmega2560 mit einer externen analogen Spannungsre-

ferenz betrieben werden, darf die Eingangsspannung der drei A/D-Wandler nicht über 2.048V betragen. Um dies zu erreichen wurde mittels eines $240\text{k}\Omega$ Widerstandes und des NTC-Thermistors ein Spannungsteiler realisiert, bei dem (bei den vorab festgelegten minimalen bzw. maximalen Temperaturen) die Spannungen aus Tabelle 4.6 anliegen.

	$\text{U}_{\text{ADC}} @ 0^\circ\text{C} [\text{V}]$	$\text{U}_{\text{ADC}} @ 150^\circ\text{C} [\text{V}]$	$\text{U}_{\text{ADC}} @ 280^\circ\text{C} [\text{V}]$
Heizbett	1.9393	0.0258	-
Extruder	1.9393	-	0.0028

Tabelle 4.6: Am A/D-Wandler bei min. und max. Temperatur der jeweiligen Heizung anliegende Spannung U_{ADC} (Nennwerte ohne Toleranz der Widerstände).

Die Dimensionierung des NTC Spannungsteilers für die Messung der Temperatur der Motor Controller (R50 platziert im Bereich der x- und y-Achsen auf der Leiterplatte, da dort die grösste Erwärmung zu erwarten ist) erfolgt analog dazu.

Lastkapazität Quarz

Die Lastkapazität C_L berechnet sich aus den beiden angehängten Kondensatoren C_1 und C_2 sowie der C_{Stray} , welche sich aus den Pinkapazitäten der XTAL Pins des Mikrocontrollers, der Kapazität des Quarzgehäuses und seiner Anschlüsse sowie dem Layout ergibt. Diese kann daher lediglich abgeschätzt werden. Die Lastkondensatoren C_1 und C_2 sind wie in Abbildung 4.15 am Mikrocontroller angehängt.

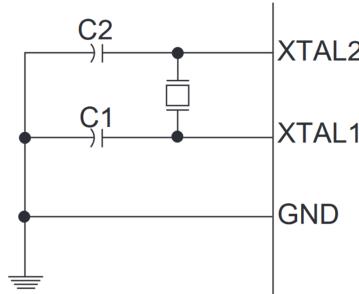


Abbildung 4.15: Beschaltung des Quarzes am Mikrocontroller [11]

Mit den folgenden gegebenen Werten berechnen sich C_1 und C_2 gemäss Formel 4.9 [16].

$$C_{1/2} = 2 \cdot C_L - 2 \cdot C_{\text{Stray}} = 27\text{pF} \quad (4.9)$$

$$\begin{aligned} C_L & \quad \text{Lastkapazität} = 18\text{pF} \\ C_{\text{Stray}} & \quad \text{Streukapazität} \approx 5\text{pF} \quad [17] \end{aligned}$$

Vorwiderstände der LEDs

Um Strom zu sparen, wurden lowpower LEDs verwendet, welche bereits bei Stromstärken unter 1mA gut sichtbar leuchten. Dazu wurden mit der Formel 4.10 alle Vorwiderstände berechnet.

$$R_{\text{vor}} = \frac{U_q - U_{\text{LED}}}{I_{\text{LED}}} \quad (4.10)$$

$$\begin{aligned}
 U_q & \quad \text{Speisespannung} = 3.3V \\
 U_{LED} & \quad \text{Durchflussspannung LED Grün} = 2.72V \\
 I_{LED} & \quad \text{Durchflusstrom LED Grün} = 390\mu A
 \end{aligned}$$

Die Vorwiderstände der LEDs, welche an den Ventilatoren angeschlossen sind, wurden so dimensioniert, dass diese mit 50% DutyCycle des PWMs dieselbe Helligkeit aufweisen wie die statisch betriebenen LEDs. So kann optisch erkannt werden wenn der DutyCycle vergrössert und verringert wird.

Übertragungsgeschwindigkeit UART

Bei UART wird die Übertragungsrate in *Baud* angegeben. Dabei entspricht 1 *Baud* einem Symbol pro Sekunde. Die Datenmenge eines Symbols hängt von der Kodierung ab [18].

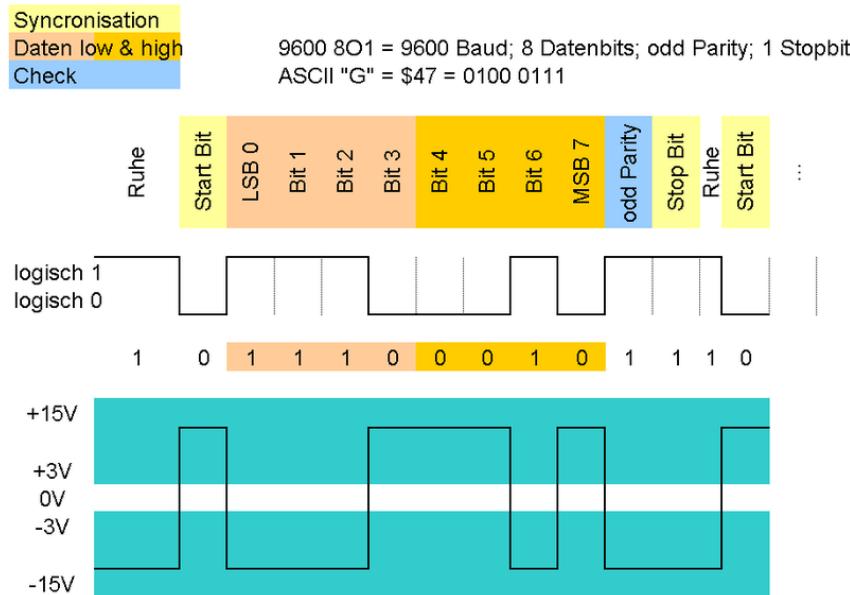


Abbildung 4.16: UART Übertragung im Zeitverlauf [19]

Der in Abbildung 4.16 dargestellte Signalverlauf zeigt die Übertragung beispielhaft. In Marlin ist die Kodierung der seriellen Schnittstelle standardmäßig auf 8 Datenbits, 1 Stopbit, 1 Startbit sowie 1 Parität-Bit eingestellt. Dementsprechend ergeben sich pro 8 übertragene Datenbits drei zusätzliche Bits welche zusammen 11 Signalbits bilden.

Um die theoretische maximale Datenrate in *bit/s* zu berechnen kann die folgende Formel angewendet werden:

$$\frac{\text{Baud}}{S_{bits}} \cdot D_{bits} = C \quad (4.11)$$

$$\begin{aligned}
 \text{Baud} & \quad \text{Symbolrate} = 250000baud \\
 S_{bits} & \quad \text{Signalbits} = 11bit \\
 D_{bits} & \quad \text{Datenbits} = 8bit \\
 C & \quad \text{Datenrate} = 181.81kbits/s = 22.72kbyte/s
 \end{aligned}$$

Die in Formel 4.11 verwendeten Werte entsprechen den angewendeten Parametern. Daher beschreibt das Resultat bereits die maximal theoretische Datenrate von unserem System. Es handelt sich hierbei nur um theoretische Werte, da die Verluste der Wartezeiten und die des Softwareprotokolls nicht eingerechnet wurden. Die effektiven Werte werden mit einer Messung am System im Abschnitt 6.7 beschrieben.

4.4 Modifikationen

Da der *Ender 3 Pro* nicht alle Features aufweist, welche in der Zielsetzung formuliert sind, ist er mit einigen Modifikationen versehen. Zusätzliche Bauteile sind mit dem Programm *3D-Builder* von Windows kreiert, welches ein Einsteigerprogramm zum dreidimensionalen Zeichen ist. Andere Bauteile wiederum stammen von der Website *Thingiverse*. Thingiverse ist eine Online-Gemeinschaft, welche sich dem 3D-Druck widmet. Auf dieser Plattform können die Nutzer dreidimensionale Modelle teilen und anderen Nutzern zur Verfügung stellen. Speziell dabei ist, dass diese Modelle gratis und lizenzzfrei zur Verfügung gestellt werden. Außerdem können Diskussionen gestartet, und bei Problemen um Hilfe gebeten werden. Thingiverse erfreut sich an einer immer grösseren Beliebtheit und ist auch dementsprechend gewachsen. Die am Computer gezeichneten Teile wurden schlussendlich mittels 3D-Druck erstellt und anschliessend montiert [20][21].

Gehäuse

Die wohl grösste Veränderung stellt das neue Gehäuse des Mainboards dar. Durch die zahlreichen Modifikationen wie dem ESP8266 WiFi-Modul, welches in Kapitel 4.1 Hardwaremäßig und im Kapitel ?? Softwaremäßig erläutert wird, dem Filament-Runout-Sensor und dem BLTouch Bed-Level-Sensor musste dieses vergrössert werden. Eine detaillierte Beschreibung des BLTouch kann dem Kapitel 4.1 entnommen werden. Durch diese Dimensionsänderungen kann das Mainboard nicht am selben Ort wie das mitgelieferte platziert werden. Um ein sauberes Kabelmanagement und eine ausreichende Kühlung gewährleisten zu können, ist das Mainboard unter die Mittelstrebe gemäss den Abbildungen 4.17 und 4.18 versetzt [20].



Abbildung 4.17: Gehäuse für die Elektronik von links.

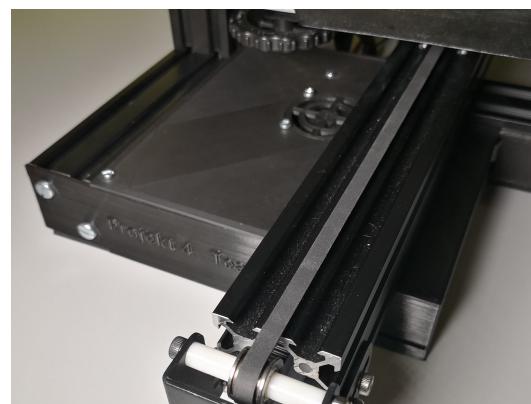


Abbildung 4.18: Gehäuse für die Elektronik von rechts.

LCD

Um die Elektronik des Display's zu schützen, ist dieses gemäss Abbildung 4.19 in ein Gehäuse

verbaut, welches von Thingiverse stammt. Somit sieht auch das Display aufgeräumter und sauberer aus. Damit das Flachbandkabel, welches das Display versorgt, sich nirgends verfängt, wird eine ausgedruckte Kabelführung gemäss Abbildung 4.20 verwendet [22][23].

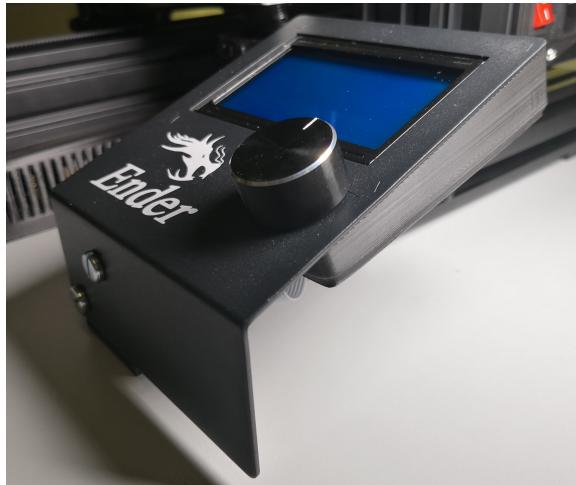


Abbildung 4.19: LCD mit Gehäuse.

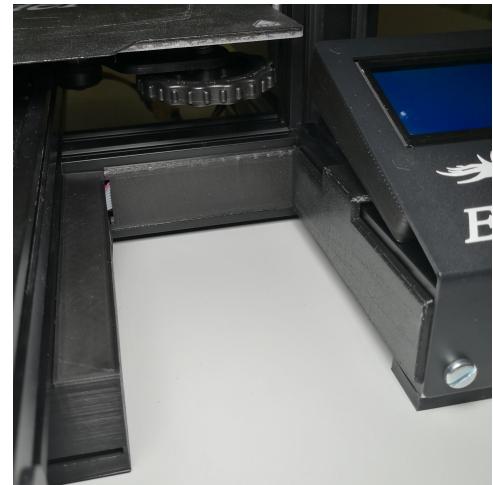


Abbildung 4.20: Kabelführung zum LCD.

BLTouch

Um den Bed-Level-Sensor befestigen zu können, wird eine gedruckte Halterung eingesetzt, welche die Befestigung des BLTouch neben dem Druckkopf ermöglicht (siehe 4.21). Diese kann ebenfalls auf Thingiverse heruntergeladen werden [24].

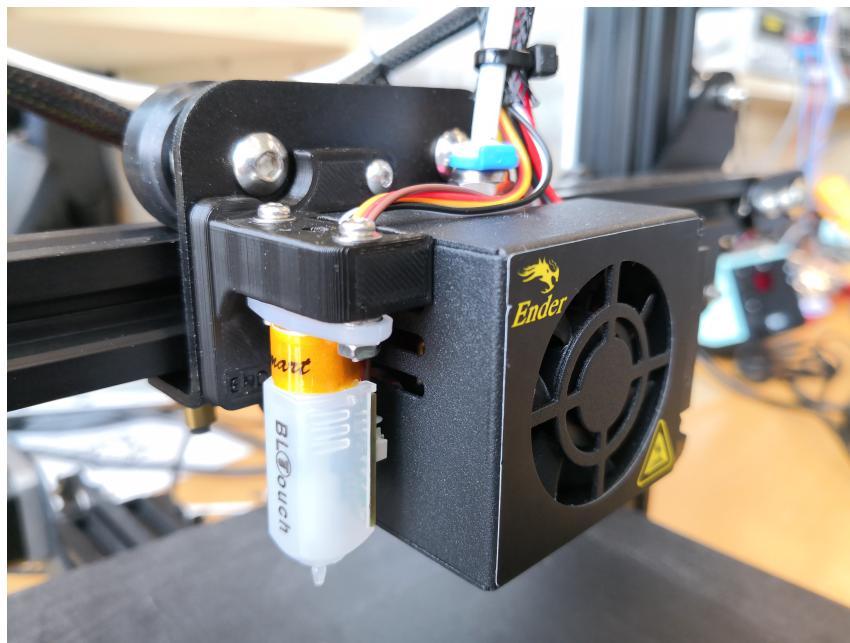


Abbildung 4.21: BLTouch Montage

Filament-Runout-Sensor

Der Filament-Runout-Sensor gemäss Abbildung 4.22 stammt von Thingiverse. Da dieser jedoch für eine physisch andere Position entworfen wurde, wird eine angepasste Version verwendet. Dadurch wird gewährleistet, dass die Bauhöhe nicht beinträchtigt wird und dass der Extruder

das Filament jederzeit ohne überhöhten Kraftaufwand fördern kann. Ausserdem wird verhindert, dass das Filament nach dem Sensor nicht zu stark gebogen wird, um die Gefahr von Brüchen im Filament zu vermindern. Auch bei dieser Modifikation ist die Kabelführung sauber gestaltet gemäss Abbildung 4.23. Dazu wird das gleiche Bauteil wie beim LCD verwendet, wenn auch in einer leicht angepassten Version [23].



Abbildung 4.22: Filament-Runout-Sensor.



Abbildung 4.23: Implementierte Kabelführung des Filament-Runout-Sensors

5 Software

In diesem Abschnitt wird auf die Software von unserem System eingegangen. Dazu werden zuerst die Anforderungen geschildert um anschliessend auf die Strukturierung, sowie die Entscheidung und Realisierung der Software einzugehen.

5.1 Anforderungen an die Software

Wie bereits in Abschnitt 2.1 angedeutet, soll durch die Software ein autonomer Betrieb des 3D-Druckers gewährleistet sein. Dies bedeutet, dass der Drucker ohne eine andauernde Verbindung zu einem Host einen Druckauftrag abarbeiten kann. Ebenfalls soll eine drahtlose Verbindung mit dem Drucker hergestellt werden können, um Druckaufträge zu verwalten, Statusinformationen auszulesen oder den Drucker manuell zu steuern. Zusätzlich soll die Benutzerfreundlichkeit mithilfe einer ansprechenden grafischen Oberfläche gewährleistet sein, welche auf internetfähigen Geräten dargestellt werden kann.

Um diese Anforderungen zu erfüllen musste die Strukturierung erfolgen welche im nachfolgenden Abschnitt 5.2 erfolgt.

5.2 Software Strukturierung

Das Datenflusskonzept dient als Grundlage für die Strukturierung der Software. Es zeigt auf wie ein Druckauftrag über die einzelnen Zwischenstationen auf den Drucker gelangt.

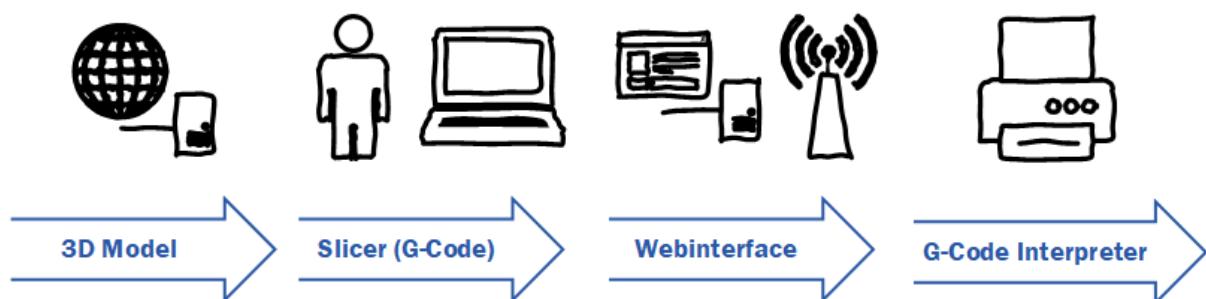


Abbildung 5.1: Datenflusskonzept

Wie in Abbildung 5.1 zu sehen ist, wird ein 3D-Model aus dem Internet heruntergeladen oder selber erstellt. Anschließend wird aus dem 3D-Model ein G-Code generiert. Dies geschieht mithilfe eines Programms welches „Slicer“ genannt wird. Über ein Webinterface kann dieser G-Code auf den 3D-Drucker geladen werden. Dazu muss auf dem Drucker ein G-Code Interpreter installiert sein, welcher den G-Code lesen und interpretieren kann.

Die Software lässt sich also in zwei Hauptbestandteile aufteilen, den **G-Code Interpreter**, welcher in Abschnitt 5.3 beschrieben wird und das **Webinterface**, welches im Abschnitt 5.4 erläutert wird.

5.3 G-Code Interpreter

In den nachfolgenden Abschnitten wird auf die Entscheidung, sowie Realisation des G-Code Interpreters eingegangen.

Entscheidungsfindung

Der G-Code Interpreter soll auf dem gewählten Mikrocontroller ATmega2560, wie in Abschnitt 4.1 beschrieben laufen und den gesamten Druckprozess koordinieren. Einerseits soll diese Firmware direkt am Drucker über die Bedienelemente (Encoder, Display usw.) und andererseits von dem geplanten Webinterface aus gesteuert werden können. Da das Programmieren einer solchen Firmware den Rahmen des Projektes sprengen würde, soll eine bereits bestehende Firmware eingesetzt werden. Ebenfalls ist es von Vorteil einen verbreiteten Interpreter zu verwenden, da diese meist eine bessere Dokumentation haben und auch mehr Komponenten bereits unterstützen.

Die am meisten verwendeten Interpreter sind RepRap Firmware, Repetier, Smoothieware und Marlin.

Entscheidung

Es wurde für die Firmware Marlin entschieden. Diese weist diverse Vorteile auf. Zum einen ist sie weitverbreitet und gut dokumentiert. Weiter hat sie im Vergleich zu anderen Firmwares viele Zusatzfunktionen welche auf dem Drucker implementiert werden können. Zudem ist sie mit vielen Komponenten (Bsp. Motortreiber) kompatibel, was eine grösse Auswahl an Peripherie erlaubt. Man kann die Firmware direkt durch Bedienelemente Steuern oder extern beispielsweise über ein Webinterface.

Realisation

Um Marlin auf eine Hardware implementieren zu können sind vor allem die beiden Konfigurationsdateien *Configuration.h* und *Configuration_adv.h* zu bearbeiten. Darin können diverse `#define`-Statements verändert werden. Gearbeitet wurde dabei mit *Visual Studio Code*.

CPU Frequenz

Marlin ist standartmässig für eine Frequenz von 16MHz ausgelegt. Der gewählte Mikrocontroller kann bei einer Speisespannung von 3.3V jedoch nur mit 8MHz getaktet werden. Es ist nicht ausreichend lediglich die Variable *CPU_FREQUENCY* anzupassen. Die Firmware greift für diverse Operationen auf sogenannte *Lookuptables* zurück. Mittels eines Python-Skript kann eine solche Tabelle für jede Frequenz automatisch generiert werden, also auch für 8MHz [25]. Diese ist dafür verantwortlich, dass *Marlin* die Motoren mit der richtigen Geschwindigkeit ansteuert. Ansonsten würde nur halb so schnell gedruckt werden.

Pin Adressierung

Da *Marlin* auf *Arduino* basiert, erfolgt die Adressierung der Pins über die von *Arduino* definierten Standard Pinbelegung [26]. Jedoch wurden design- und vor allem layouttechnisch gewisse Pins belegt, welche von *Arduino* nicht unterstützt werden. Um dies zu korrigieren musste das Erweiterungspaket *Megacore* implementiert werden, welches es erlaubt alle Pins anzusteuern [27]. Um die Erweiterung korrekt zu implementieren, musste auf den G-Code Befehl *M43* zurückgegriffen werden. Dieser gibt zur Laufzeit die aktuelle Pinbelegung und deren Zustände zurück. Um das Testen der Pins durchzuführen wurde der G-Code Befehl *M42*⁴ verwendet, welcher

⁴Dies ist ein experimentelles Feature

erlaubt die Pins einzeln anzusteuern. Dies erleichterte die Inbetriebnahme der Ventilator- und Heizungssteuerungen erheblich.

5.4 Webinterface

Das Webinterface umfasst die Benutzerschnittstelle über WLAN. In den nachfolgenden Abschnitten wird auf die Entscheidung und Realisation eingegangen.

Entscheidungsfindung

Um den Drucker direkt steuern zu können soll das Webinterface mit dem verwendeten G-Code Interpreter kommunizieren und somit kompatibel sein. Es soll möglich sein die Daten für den Druckauftrag direkt über Wifi auf den Drucker zu laden.

Entscheidung

Für das Webinterface wurde für die Firmware ESP3D entschieden. Diese ist kompatibel mit Marlin und hat viele Funktionen, mit denen der Drucker bedient werden kann. Das Webinterface läuft dabei direkt auf dem Wifi-Modul ESP8266 und erstellt einen Accesspoint, mit welchem darauf zugegriffen werden kann. Die Temperatur aller Komponenten kann in der Software direkt entnommen werden und alle Motoren manuell angesteuert werden.

Realisation

Die Firmware ESP3D kann sehr benutzerfreundlich direkt über die Weboberfläche konfiguriert werden. Damit die serielle Schnittstelle zum G-Code Interpreter funktioniert, musst die Baudrate entsprechend angepasst werden. Zusätzlich musste eine Funktion um die Dateien von der SD-Karte aufzulisten mit einem weiteren G-Code Befehl erweitert werden. Der dafür zuständige Befehl lautet *M21* um die SD-Karte zu initialisieren.

5.5 Lizenzen

In der untenstehenden Tabelle können die Lizenzen der verwendeten Bibliotheken und Firmwares entnommen werden.

Bibliothek/Firmware	Lizenz
Arduino IDE	GNU GPL v2.0
Marlin	GNU GPL v3.0
ESP3D	GNU GPL v3.0
ESP3D-WEBUI	GNU GPL v3.0
Megacore	GNU GPL v2.0

6 Inbetriebnahme

Im Rahmen der Inbetriebnahme wurden diverse Messungen durchgeführt. Die wichtigsten sind nachfolgend dokumentiert.

6.1 Stabilität der 3.3V Speisung

Der Ripple auf der $+3.3V$ Spannungsversorgung ist mit $\sim 40mV$ im erwarteten und tolerierbaren Bereich (siehe Abbildung 6.1).

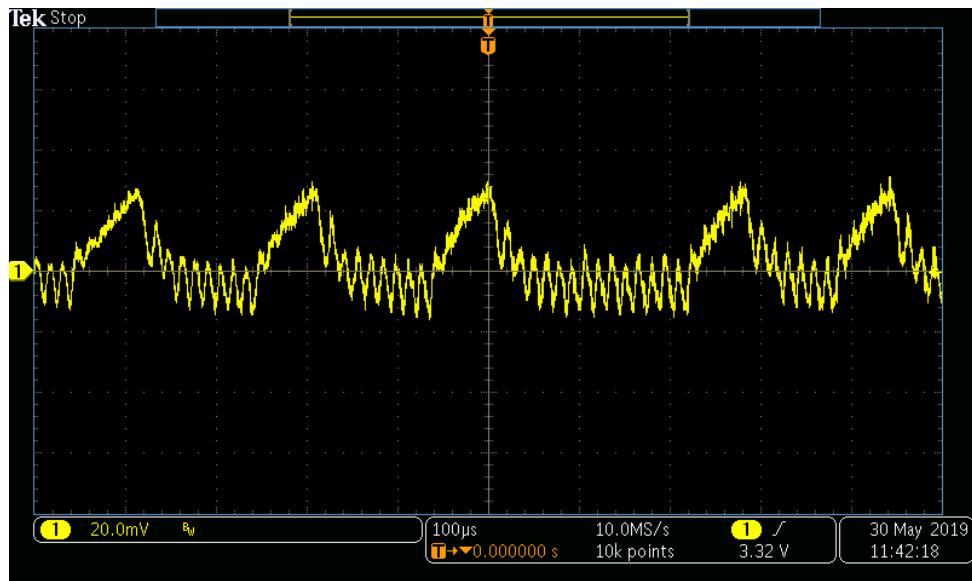


Abbildung 6.1: $\sim 40mV$ Spannungsripple auf der $+3.3V$ Speisung des DC/DC-Wandlers IC3 (mit Spannungsoffset $+3.3V$)

6.2 Stabilität der 5.0V Speisung

Der Ripple auf der $+5.0V$ Spannungsversorgung ist mit $\sim 60mV$ im erwarteten und tolerierbaren Bereich (siehe Abbildung 6.2). Die Speisung weist einen DC-Offset von $0.1V$ auf was ebenfalls im zu erwartenden Bereich der Dimensionierung ist (Dimensionierter Bereich ist $4.90 - 5.28V$, siehe Abschnitt 4.3).

6.3 Messung des Stromverbrauchs

Der Stromverbrauch der gesamten Leiterplatte, gemessen in der $+24V$ Speisung, beträgt $\sim 125mA$ und bewegt sich im zu erwartenden Bereich (siehe Abbildung 6.3).

6.4 Verifizierung der Temperaturmessung

Die via Marlin am NTC-Thermistor des Heizbettes gemessene Temperatur wurde in einer Temperaturkammer mit den Messungen eines externen PT100 Thermistors verglichen. Dabei zeigte sich eine maximale Abweichung von $\sim 10^\circ C$ (siehe Abbildung 6.4). Diese Abweichung ist in Anbetracht der wechselhaften Umweltbedingungen in welchen der 3D-Drucker betrieben wird

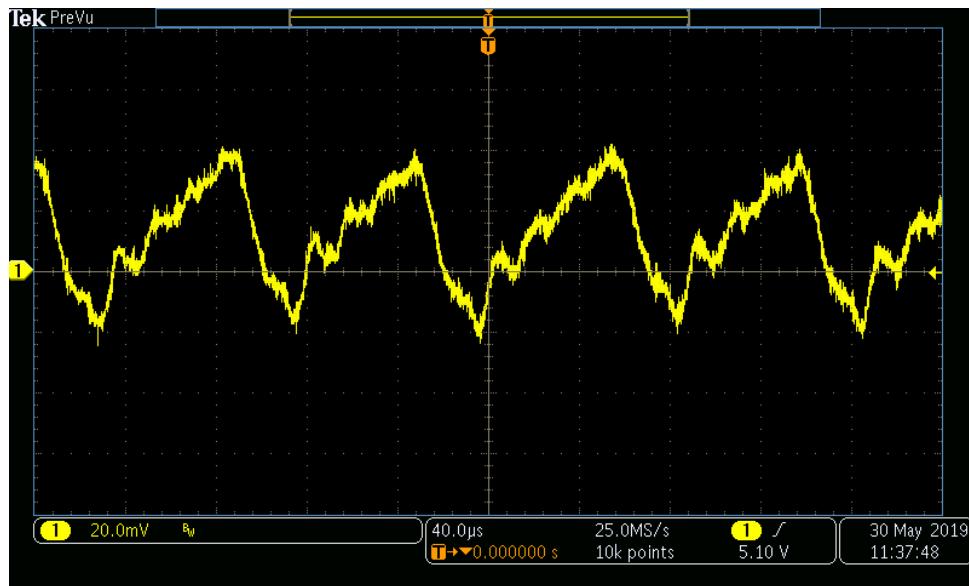


Abbildung 6.2: $\sim 60\text{mV}$ Spannungsripple auf der $+5.0\text{V}$ Speisung des DC/DC-Wandlers $IC5$ (mit Spannungsoffset $+5.1\text{V}$)

zu vernachlässigen. Zu erklären ist die Abweichung mit den Fehlern welche beim Linearisieren der Lookup-Tabelle in der Firmware erfolgen, durch die Verwendung von $\pm 1\%$ Widerständen im Spannungsteiler der NTC-Thermistoren, durch Unkenntnis der Toleranz des NTC-Widerstandes (verwendeter NTC-Thermistor wird vom Hersteller des 3D-Druckers nicht genau spezifiziert) sowie durch die Verwendung der $+3.3\text{V}$ Speisung als Analogreferenz anstatt des externen $+2.048\text{V}$ Referenzbausteins $IC1$. Dies ist nötig da die Kombination aus Spannungsteiler, $+2.048\text{V}$ Referenz, exponentiell abfallender Thermistor-Kennlinie (siehe Abbildung 6.5) und niedriger 10-Bit ADC-Auflösung unglücklich gewählt ist (wird der Vorwiderstand klein gewählt, ist bei tiefen Temperaturen ein Unterscheiden der Temperaturen als ADC-Wert nicht mehr möglich, wird er hingegen gross gewählt, ist dies bei hohen Temperaturen nicht mehr möglich). Lösung dafür wäre einen zeitgemässen AD-Wandler mit grösserer Bitzahl zu verwenden, sowie mit der genauen Funktion und nicht mit einer Lookup-Tabelle der Widerstandswerte in der Firmware zu rechnen.

6.5 MOSFET Schwingverhalten

Es wurden alle MOSFETs für Extruder-, Heizbett- sowie Ventiltorsteuerung im laufenden Betrieb ausgemessen und auf ihr Schwingverhalten untersucht. Dabei stellte sich heraus, dass keiner der MOSFETs ein problematisches Schwingverhalten zeigt (siehe Abbildung 6.6)

6.6 Anhebung der 24V Speisung beim Bremsen der Schrittmotoren

Es wurde gemessen, ob bei gleichzeitigem starken Bremsen aller Achsen, die $+24\text{V}$ Speisung angehoben werden könnte (aus maximalem Vorschub zu Stillstand). Dabei zeigte sich das dies nicht der Fall ist.



Abbildung 6.3: Stromverbrauch der gesamten Leiterplatte inklusive der drei laufenden Ventilatoren (keine fahrenden Achsen, keine Heizungen aktiv)

6.7 Messung Übertragungsgeschwindigkeit Wifi zu SD-Karte

Die Übertragungsgeschwindigkeit für den Upload via Wifi kann mittels Upload eines Files festgestellt werden. Dazu muss lediglich die Dateigrösse bekannt sein und die Zeit während des Uploads gemessen werden.

$$\frac{Fs}{t} = C \quad (6.1)$$

Fs	Dateigrösse = 156kbyte
t	Übertragungszeit = 268s
C	Datenrate = 0.582kbyte/s

Die Werte sind sehr gering im Vergleich zur maximal möglichen Übertragungsrate. Dies ist mit dem Softwareprotokoll zu Begründen welches zur Übertragung genutzt wird.

6.8 Messung problematischer Schrittverluste

Um festzustellen ob während dem Druck Schrittverluste entstehen wurde ein Testdruck durchgeführt. Dabei sind an dem fertigen Druck keine Verschiebung der Schichten aufgefallen, womit Schrittverluste ausgeschlossen werden können.

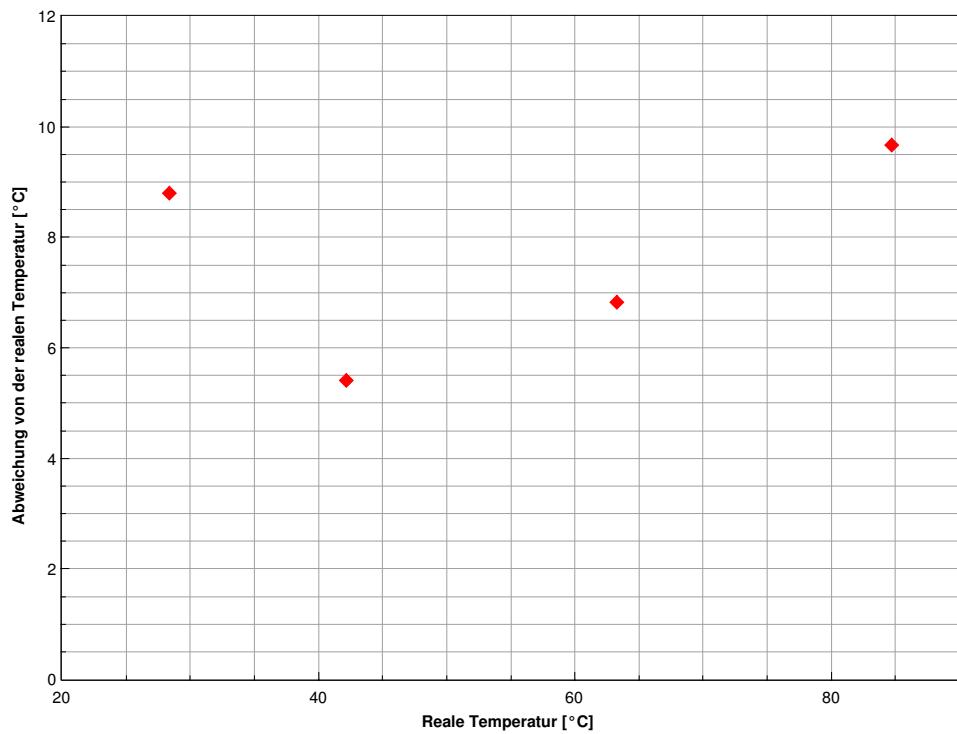


Abbildung 6.4: Temperaturabweichung der mittels Marlin gemessenen von der realen Temperatur

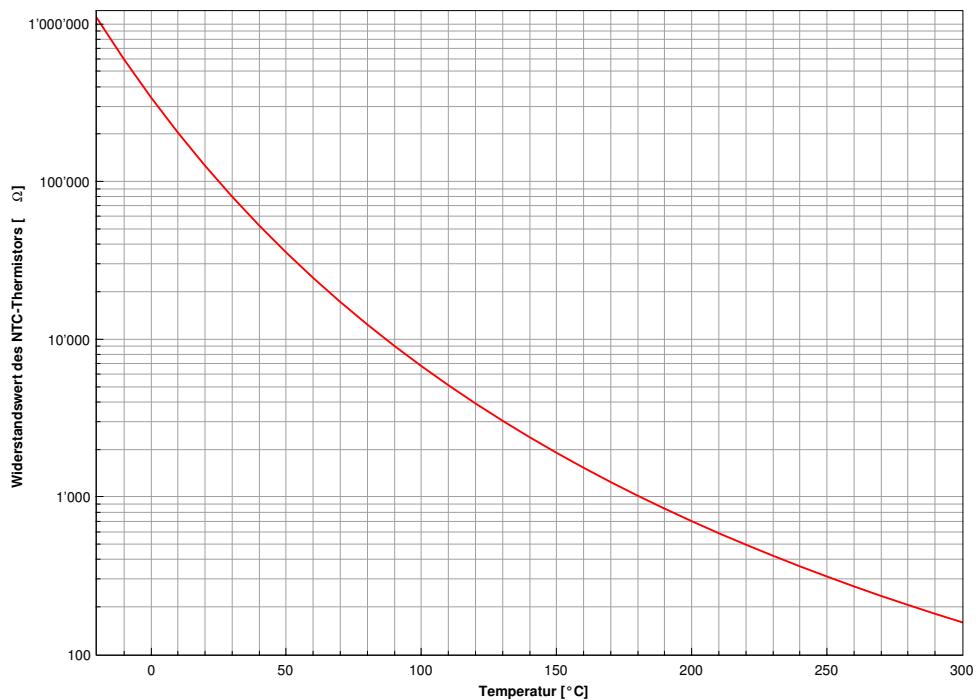


Abbildung 6.5: Kennlinie des verwendeten NTC-Thermistors über den gesamten relevanten Temperaturbereich. Sichtbar ist das starke Abfallen des Widerstandswertes bei hohen Temperaturen.

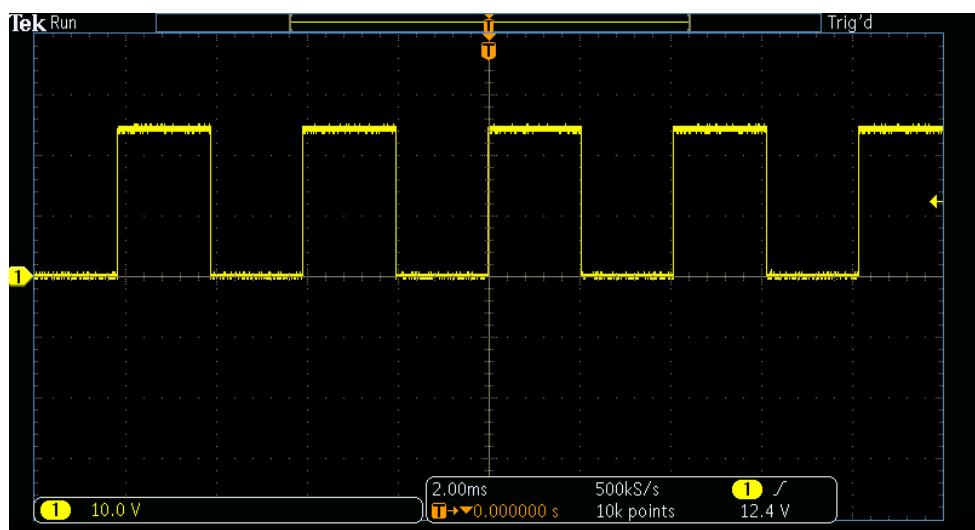


Abbildung 6.6: U_{DS} gemessen am MOSFET T1 mit angehänger Last und 50% DutyCycle. Es ist kein Schwingverhalten messbar

7 Fazit

Im Projekt 4 des Studiengangs Elektro- und Informationstechnik war es das Ziel, zu untersuchen, inwiefern ein günstiger 3D-Drucker modifiziert werden kann, damit er seine Funktion mit erhöhter Druckqualität und Benutzerfreundlichkeit wahrnehmen kann. Dazu wurde eine neue Steuerplatine entwickelt. Diese unterstützt eine Ansteuerung der Schrittmotoren mittels Microstepping und verfügt über ein WiFi-Modul, welche die drahtlose Übertragung von G-Code Dateien an den Drucker erlaubte. Zusätzlich wurde der verwendete *Ender 3 Pro* mit einem System zur automatischen Nivellierung des Heizbettes und einem Sensor zur Erkennung einer Filamenterschöpfung ausgerüstet.

Durch die verbesserte Motorkontrolle konnten während des Druckvorgangs entstehende Vibratiorionen stark verringert werden, was sich in einer gesteigerten Druckqualität und einer spürbar geringeren Lautstärke bemerkbar machte. Am meisten konnte die Benutzerfreundlichkeit verbessert werden. Das automatische Nivelliersystem eliminierte die Notwendigkeit, das Heizbett vor jedem Druckauftrag manuell auszurichten. Weiterhin erlaubte das WiFi-Modul einen Verzicht auf Kabelverbindungen zwischen dem 3D-Drucker und einem Computer und das Übertragen von G-Code Dateien mittels SD-Karte.

8 Schlusswort

Nach Abschluss der Entwicklung wurde das Produkt validiert. Die Resultate sind in Kapitel 6 dokumentiert. In diesem Kapitel soll aufgelistet werden, welche Teilsysteme funktionieren und welche nicht. Im letzten Abschnitt wird das Verbesserungspotenzial für eine mögliche Weiterentwicklung erfasst.

Der entwickelte Print erfüllt einen Grossteil der in Kapitel 2.2 formulierten Kriterien. Das Produkt umfasst den ersten Prototyp des Steuerprints und die Mechanik des Druckers mit diversen Modifikationen. Die Software besteht aus den beiden Firmwares Marlin und ESP3D. Bei der Steuerung der Motoren funktionieren folgende Punkte einwandfrei.

- Alle vier Motoren (x-, y-, z-Achse und Filamentzuführung) können unabhängig und ohne Schrittverlust angesteuert werden.
- Die Nullpositions-Erkennung der x-, y- und z-Achse funktioniert und wird jedes mal automatisch vor jedem Druck ausgeführt.

Die Bedienung des Druckers direkt am Drucker wie auch auf dem Webinterface funktioniert.

- Die Steuerung des Druckers erfolgt über Bedienelemente direkt am Drucker (Encoder, LCD und LEDs).
- Alle Achsen und Motoren können manuell durch Encoder und Display oder das Webinterface gesteuert werden.
- Eine Verwaltung der Druckaufträge kann über das Webinterface durchgeführt werden. Weiter wird auf diesem der Status (Temperatur, Position usw.) angezeigt.

Auch die zusätzlich eingebaute Sensorik funktioniert ohne Probleme.

- Ein Sensor erkennt automatisch eine Erschöpfung des Filaments und meldet dies an das Programm.
- Durch den BLTouch Sensor kann die z-Achse automatisch nivelliert werden. Der BLTouch ist dabei auch für die Nullpositions-Erkennung zuständig.

Die Temperatursteuerung der verschiedenen Komponenten im Drucker funktioniert ebenfalls.

- Die Temperaturen von Extruder und Heizbett werden erfasst und können über das Webinterface ausgelesen werden.
- Die Leiterplatte, das Filament und der Druck werden durch Ventilatoren gekühlt.

Leider gibt es auch noch Dinge die noch nicht funktionieren.

- Die Temperaturmessung ist nur auf etwa 10° genau. Somit konnte das Kriterium der genauen Regelung des Heizbettes und des Extruders nicht erreicht werden.
- Das senden von G-Code Dateien über das Webinterface auf den Drucker funktioniert nicht richtig. Überschreitet die Datei eine bestimmte Grösse scheitert das laden.

Wie die Validierung zeigt, funktioniert der erste Prototyp. Jedoch gibt es einige Kriterien die zu diesem Zeitpunkt noch nicht erfüllt sind. Diese könnten in einem weiterführenden Projekt noch realisiert werden.

Wie in Kapitel 6.4 bereits erwähnt wurde, braucht die Messung der Temperaturen eine Überarbeitung, um eine höhere Genauigkeit als 10° zu erreichen. Dazu müssten genauere Komponenten gewählt werden und in der Software eine Anpassung getätigter werden.

Weiter könnte noch eine Nullpositionserkennung durch Überstromdetektion der Motoren implementiert werden. Der verwendete Motortreiber unterstützt diese Funktion. Dazu müsste lediglich die Software angepasst werden, da die Hardware die Funktion schon unterstützt. Der Vorteil daran wäre, dass keine mechanische Abnutzung an den Schaltern entstehen würde.

Da der verwendete 8MHz Mikrocontroller viele Probleme mit sich bringt müsste man bei einer Weiterentwicklung ein neues Bauteil verwenden mit 16MHz Taktfrequenz. Damit könnte die Übertragungsrate des Webinterface erhöht werden. Dies würde womöglich das Problem des Ladens der G-Code Dateien über Wifi beheben, welches ebenfalls in einer weiterführenden Entwicklung behoben werden müsste.

9 Ehrlichkeitserklärung

Mit der Unterschrift bestätigt der Unterzeichnende (Projektleiter), dass das Dokument selbst geschrieben worden ist und alle Quellen sauber und korrekt deklariert worden sind.

Fabian von Büren

Datum, Ort

Literatur

- [1] Tiffany Orr. (2014). Fire Aftermath. (Abrufdatum: 28.05.2019), Adresse: <https://3dprint.com/18064/smoke-signal-3d-print-fire/>.
- [2] additive3d. (2016). Fused Deposition Modeling. (Abrufdatum: 06.05.2019), Adresse: <https://additive3d.com/wp-content/uploads/2016/08/Fused-Deposition-Modeling.jpg>.
- [3] Filamentworld. (2019). PLA. (Abrufdatum 07.06.2019), Adresse: <https://www.filamentworld.de/3d-druck-wissen/was-ist-pla>.
- [4] —, (2019). PETG. (Abrufdatum 07.06.2019), Adresse: <https://www.filamentworld.de/3d-druck-wissen/was-ist-petg>.
- [5] Filamentworld. (2019). ABS. (Abrufdatum 07.06.2019), Adresse: <https://www.filamentworld.de/3d-druck-wissen/was-ist-abs>.
- [6] Creality 3D. (2019). Creality3D Ender-3 3D-Drucker. (Abrufdatum 10.05.2019), Adresse: <https://www.creality3d.shop/products/creality-ender-3-3d-printer-economic-ender-diy-kits-with-resume-printing-function-v-slot-prusa-i3-220x220x250mm?lang=de>.
- [7] Autodesk. (2019). Getting Started with G-Code. (Abrufdatum 06.05.2019), Adresse: <https://www.autodesk.com/industry/manufacturing/resources/manufacturing-engineer/g-code>.
- [8] 3D-Druckerzentrum Schweiz. (2019). Die wichtigsten G-Codes. (Abrufdatum 06.05.2019), Adresse: http://www.3d-druckerzentrum.ch/shop/shop_content.php?coID=73.
- [9] RN Wissen. (2017). Schrittmotoren. (Abrufdatum 06.05.2019), Adresse: <https://rn-wissen.de/wiki/index.php/Schrittmotoren>.
- [10] Robert Keim. (2018). How to Choose the Right Driver IC for Stepper Motors. (Abrufdatum 06.05.2019), Adresse: <https://www.allaboutcircuits.com/technical-articles/how-to-select-the-right-driver-ic-for-stepper-motors/>.
- [11] Microchip Technology. (2014). Datasheet ATmega2560. (Abrufdatum: 06.06.2019), Adresse: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf.
- [12] Microchip. (2014). Datasheet MCP16311. (Abrufdatum: 06.05.2019), Adresse: <http://ww1.microchip.com/downloads/en/DeviceDoc/20005255B.pdf>.
- [13] Murata Seisakusho K.K. (2010). Datasheet: DS75LC Ferrite Inductors. (Abrufdatum: 10.05.2019), Adresse: https://www.murata.com/~/media/webrenewal/products/inductor/chip/tokoprodukte/wirewoundferrittypeforpl/m_ds75lc.ashx.
- [14] Multi Circuit Boards Ltd. (2009). Application Note: Formeln zur Berechnung der maximalen Stromstärke. (Abrufdatum: 10.05.2019), Adresse: <https://www.multi-circuit-boards.eu/leiterplatten-design-hilfe/oberflaeche/leiterbahn-strombelastbarkeit.html>.
- [15] Trinamic. (2016). Calculation sheet TMC2130. (Abrufdatum: 09.05.2019), Adresse: http://www.trinamic.com/_scripts/download.php?file=_articles%2Fproducts%2Fintegrated-circuits%2Ftmc5130%2F_appnotes%2FTMC5130_TMC2130_TMC2100_Calculations.xlsx.
- [16] B. Neubig und W. Briese, *Das grosse Quarzkochbuch*, 4. Auflage. Feldkirchen, Franzis-Verlag, 1997.
- [17] Abracon Inc. (2015). Datasheet ABM3B-8.000MHZ-B2-T. (Abrufdatum: 06.05.2019), Adresse: <https://abracon.com/Resonators/abm3b.pdf>.
- [18] Microcontroller.net. (2009). Baud. Abrufdatum: 07.05.2019, Adresse: <https://www.mikrocontroller.net/articles/Baud>.

- [19] Markus Ulsaß. (2009). Picture UART Kodierung. Abrufdatum: 07.05.2019, Adresse: <http://docplayer.org/docs-images/34/13096909/images/8-0.png>.
- [20] Microsoft. (2013). 3D Builder. (Abrufdatum 20.04.2019), Adresse: <https://www.microsoft.com/de-ch/p/3d-builder/9wzdncrfj3t6?activetab=pivot:overviewtab>.
- [21] MakerBot Industries. (2019). Thingiverse. (Abrufdatum 20.04.2019), Adresse: <http://www.Thingiverse.com>.
- [22] pruckelshaus. (2018). Ender 3 LCD Case flip-it remix. (Abrufdatum 20.04.2019), Adresse: <https://www.thingiverse.com/thing:3035562>.
- [23] JaycenS. (2019). Ender 3 Pro LCD Ribbon Cable Cover. (Abrufdatum 20.04.2019), Adresse: <https://www.thingiverse.com/thing:3419155/files>.
- [24] registeredthing. (2018). BlTouch Mount For Ender 3 (remix) - no supports needed. (Abrufdatum 20.04.2019), Adresse: <https://www.thingiverse.com/thing:3003725/files>.
- [25] B. Gamari. (2012). Create speed lookuptable. (Abrufdatum 14.05.2019), Adresse: https://github.com/reprapro/Marlin/blob/master/Marlin/create_speed_lookuptable.py.
- [26] Marlin. (2019). FastIO and Pin Mapping. (Abrufdatum 14.05.2019), Adresse: <http://marlinfw.org/docs/hardware/boards.html>.
- [27] MCUdude. (2019). MegaCore. (Abrufdatum 14.05.2019), Adresse: <https://github.com/MCUdude/MegaCore>.

A Lastenheft

Auftrag / Lastenheft

Steuerung für einen 3D-Drucker

Anlass: Dreidimensionale Drucker, oder 3D-Drucker, werden die technische Produktion grundlegend verändern. Es gibt nichts, was dereinst nicht 3D gedruckt werden kann. Von der Geburtstagstorte bis zum Fertighaus wird über alles spekuliert. Wir bleiben am Boden und versuchen in diesem P4 nach dem gängigsten Verfahren einen Kunststofffaden zu schmelzen und damit ein dreidimensionales Objekt aufzubauen.

Unter der Bezeichnung K8200 verkauft die Firma Velleman den Bausatz eines einfachen 3D-Druckers. Wichtig für uns ist darin die Mechanik. Gestellrahmen, X-Y-Motoren, Z-Antrieb mit Spindel, Extruderkopf mit Heizung und Vorschub, ja selbst die Rollenhalterung für das Filament ist in diesem Bausatz inbegriffen. Auch ein Steuerprint gehört dazu. Vom PC aus (USB) kann der Drucker über diese Steuerung direkt betrieben werden.

In diesem P4 sollen Sie eine neue Steuerung zu diesem Drucker K8200 entwickeln und realisieren, um damit den Drucker etwas einfacher, robuster und leistungsfähiger zu machen. Mit einfacher und robuster meinen wir die Reduktion und Verbesserung von Sensorik und Verkabelung und mit leistungsfähiger ist vor allem ein PC unabhängiger Offlinebetrieb und die wireless-Kommunikation gemeint.

Aufgabe: Entwerfen Sie eine Ein-Platinen-Steuerung für den K8200. Realisieren und testen Sie diesen Steuerprint und entwickeln Sie dazu die Firmware. Bauen sie Ihr Werk in die Mechanik des 3D-Druckers ein und demonstrieren Sie seine Funktionstüchtigkeit. Die Steuerung muss in der Lage sein, die Schrittmotoren der X-, Y- und Z-Achse anzusteuern. Weitere Steuerfunktionen sind für den Extruder mit Heizung und Vorschub notwendig. Die Signale der Sensoren sind, soweit nötig, in die Steuerung einzulesen. Vereinfachungen, wie z.B. das Weglassen der Endschalter, sind erwünscht, wenn eine einfachere Lösung möglich ist. Zu Gunsten der Reduktion der Anzahl Litzen im Z-Achsen Hängekabel darf ein kleines "Printchen" beim Extruderkopf montiert werden. Statusanzeigen, wie z.B. einzelne LED's, und notwendige Bedienelemente, wie z.B. eine Taste, sollen direkt auf dem Print realisiert werden. Als Kommunikationsschnittstelle ist eine wireless-Verbindung zum PC zu begründen. Auch der Einsatz einer SD-Karte als Speicher und zum Datentransfer macht durchaus Sinn.

Im Projekt ist ein **Go/No-go-Punkt** vorgesehen. Der Drucker-Bausatz wird erst dann beschafft, wenn das jeweilige Team zeigen kann, dass die Steuerung einen Schrittmotor sauber ansteuern kann. Konkret muss das jeweilige Team zeigen, dass G-Code-Move-Befehle an einer Achse dauerhaft ohne Schrittverluste ausgeführt werden können.

Anforderungen Hardware:

- Standard-Elektronik für den Betrieb im Innenraum, einbaubar in den Drucker
- Printversorgung über Steckernetzteil oder Netzmodul auf Print
- Umsetzung des jeweiligen Bedienungskonzeptes
- Kommunikation zum PC / Datenspeicher: wireless / SD-Karte

Anforderungen Firmware:

- Firmware zur Verarbeitung von G-Code
- Speichermöglichkeit für einen gesamten Druck (Job)
- Ansteuern und Einlesen aller Aktoren und Sensoren
- Kommunikationssoftware
- Verwendung eines Firmware-Konfigurators erwünscht

Wunschziele:

- Nachlauf-Erkennung Filament, beim Extruder (Alarmfunktion)
- Benachrichtigung des Benutzers über den aktuellen Druckfortschritt
- Power-Save Modus

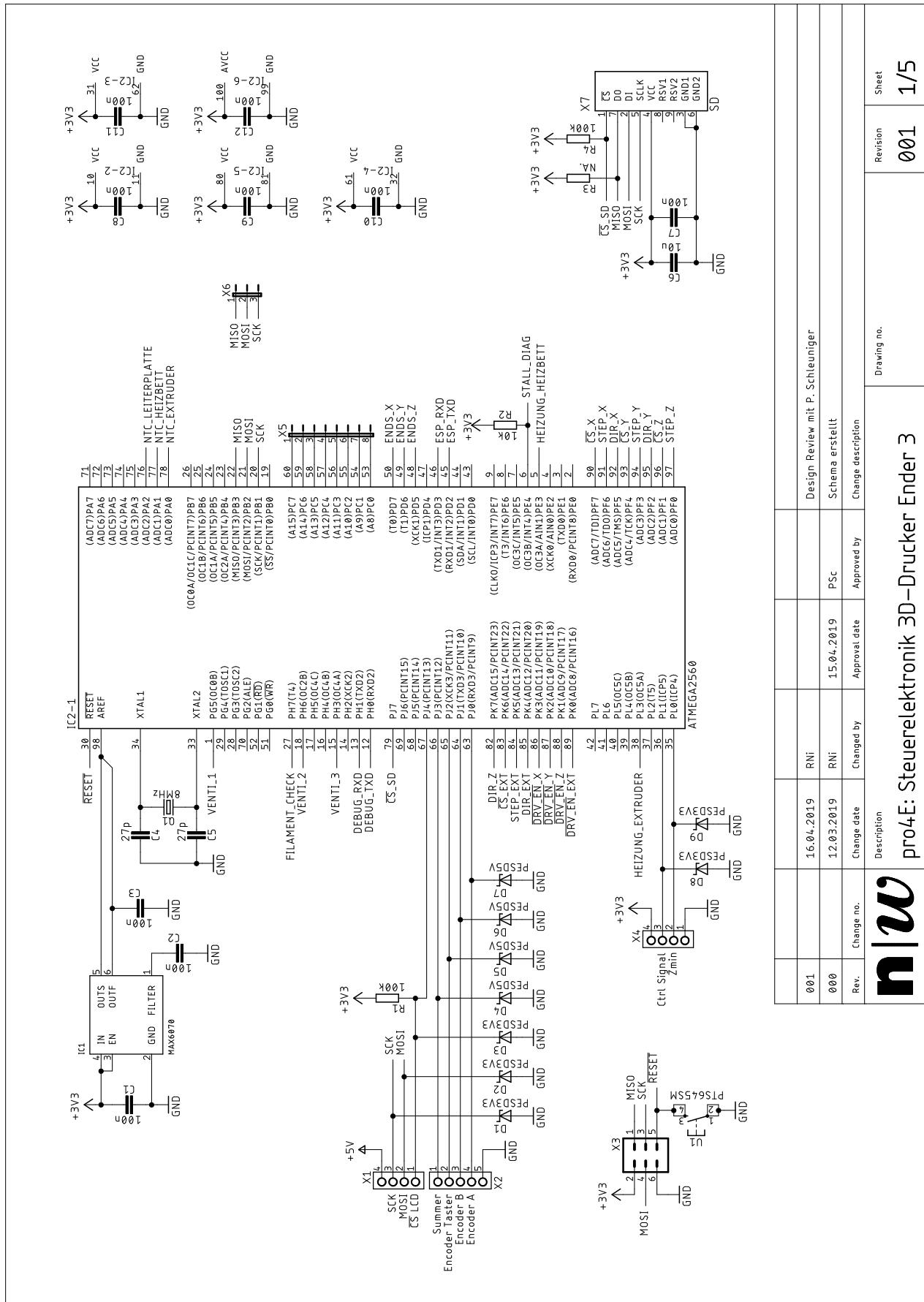
Viel Erfolg !

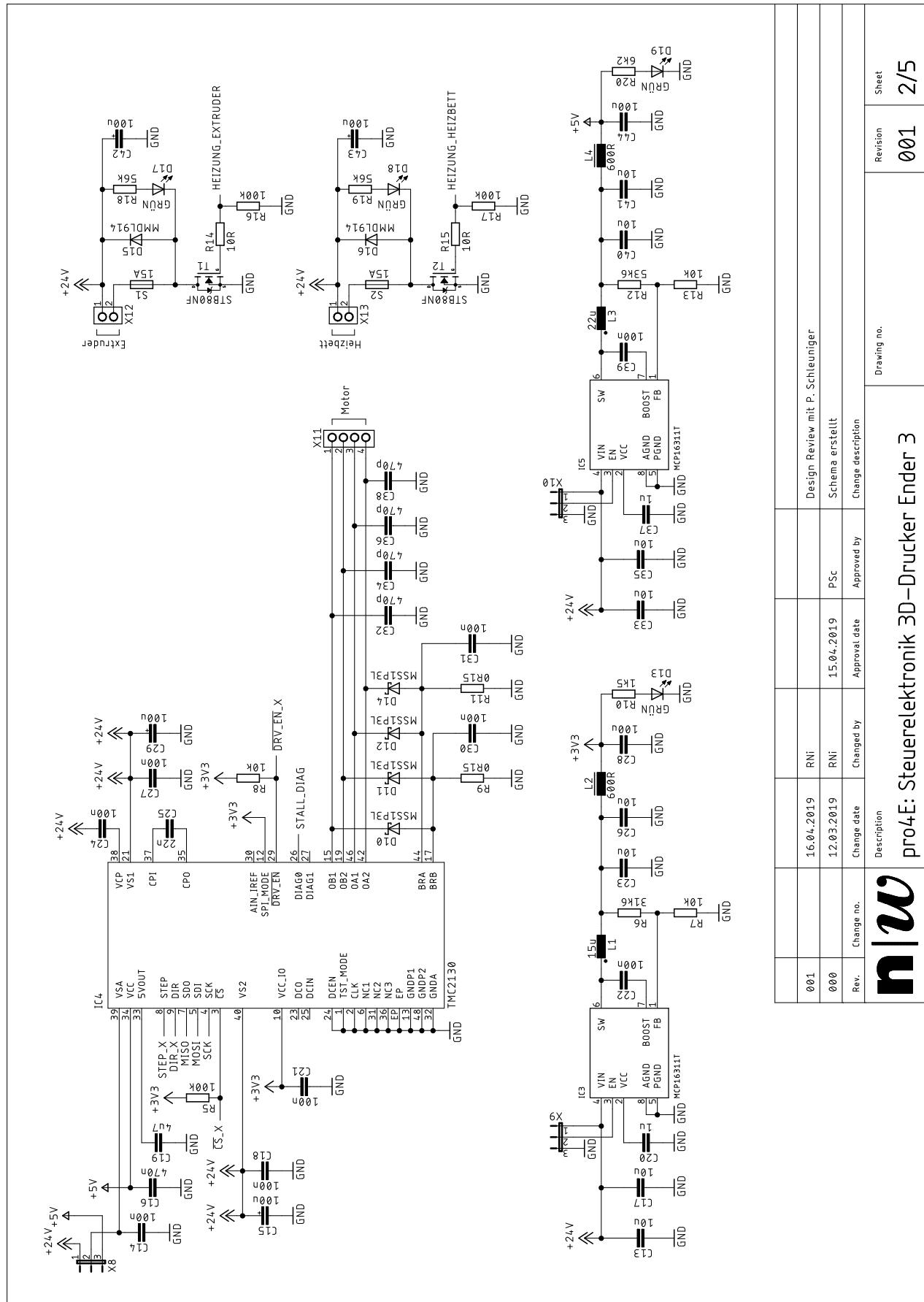
Auftraggeber / Fachcoaches:

Hans Gysin, Pascal Schleuniger, Albert Zihlmann

18.02.2019

B Schema





n|w

001
Sheet
2/5

001

000

Rev.

Change no.

001

000

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

000

001

</

