



Tillmann Albrecht
Bachelor Technische Informatik

ENTWICKLUNG EINES STIFT- PLOTTERS ZUM EINSATZ ALS MESSEDEMONSTRATOR

Bachelorarbeit an der Hochschule Augsburg

Im Auftrag von:

MIXED
MODE



**Hochschule
Augsburg** University of
Applied Sciences

Fakultät für
Informatik

Bachelorarbeit

Studienrichtung
Technische Informatik

Tillmann Albrecht **Entwicklung eines Stiftplotters zum** **Einsatz als Messedemonstrator**

Prüfer: Prof. Dr. Hubert Högl
Abgabe der Arbeit am: 23.04.2019

Hochschule für angewandte
Wissenschaften Augsburg
University of Applied Sciences

An der Hochschule 1
D-86161 Augsburg

Telefon +49 821 55 86-0
Fax +49 821 55 86-3222
www.hs-augsburg.de
info@hs-augsburg.de

Fakultät für Informatik
Telefon: +49 821 5586-3450
Fax: +49 821 5586-3499

Verfasser der Bachelorarbeit:
Tillmann Albrecht
Schloßstraße 52
86485 Markt
Telefon: +49 8271 421 7737
till@7i11.de

Abbildungsverzeichnis

Abb. 1: Bildverarbeitung in der Software von A. Kast (Kast, 2017).....	5
Abb. 2: Zeichnung der generierten Pfade (Kast, 2017)	5
Abb. 3: Graustufen-Portraitfoto.....	6
Abb. 4: Eine mithilfe der StippleGen2-Software erstellte Punktzeichnung (Oskay, 2017)	7
Abb. 5: Schraffur eines monochromen Portraits. Links das monochrome Bild, rechts die daraus resultierende Schraffur.....	8
Abb. 6: Durch kombinierte Schraffur entsteht die Illusion eines Bildes mit 3 Graustufen	9
Abb. 7: Anwendung eines visuellen Stils auf ein Bild zur Pfadgeneration (DeepArt UG, 2019)	10
Abb. 8: Hauptfenster der PlotCon-Software. Das Vorschaufenster zeigt den geladenen G-Code	11
Abb. 9: Dialog zur Auswahl der Druckparameter	12
Abb. 10: UML-Diagramm der Klassen "Path" und "Stroke". Beide haben ähnliche Attribute und Methoden, da bei beiden ähnliche Operationen angewandt werden.....	14
Abb. 11: Rendering aller an der Kinematik beteiligten Bauteile. Alle 3D-gedruckten Teile sind in orange dargestellt	17
Abb. 12: Rendering einer MGN-12 Profilschienenführung mit aufsitzendem MGN12-H Kugelumlaufwagen	18
Abb. 13: Vereinfachte Darstellung des Funktionsprinzips eines unipolaren Schrittmotors (Horina, 2004) .	19
Abb. 14: Schema der Geometrie von Rotor und Stator in Schrittmotoren (Stündle, 2010)	20
Abb. 15: Drehmomentverlauf eines vergleichbaren Motortyps im Verhältnis zur Drehzahl. Das Moment schwindet bei steigender Drehzahl. Dem kann mit einer Erhöhung der Spannung entgegengewirkt werden. (MOONS' Industries, 2019).....	21
Abb. 16: Blockdiagramm des Trinamic TMC2130 Schrittmotortreibers. Er unterscheidet sich u.A. durch die API-Kommunikation (links) von herkömmlichen Treibern. (TRINAMIC Motion Control GmbH & Co. KG, 2018)	22
Abb. 17: Aufbau der Elektronik auf Basis des RAMPS	24
Abb. 18: Oberseite der Platine des Mixed Mode Shields.....	25
Abb. 19: Typische Performance von DCDC-Wandlern auf Basis des TS30042. Bei 14 V Eingangsspannung liegt die Effizienz schon ab 300 mA deutlich über 80%. (Semtech Corporation, 2019)	26
Abb. 20: Für die Vertikalbewegung zuständiger Mechanismus mit eingespannter Kugelschreibermine	29
Abb. 21: Ein Schwertfisch (Marlin) ist Wappentier der Marlin-Firmware (Brázio, 2019)	30
Abb. 22: Angefertigte Zeichnung des schraffierten Portraits aus Abb. 5. Benötigte Zeit: 8:45 Minuten.	40
Abb. 23: Durch Kreise unterschiedlicher Radien gewichtete Punktzeichnung	41
Abb. 24: Zeichenpfad basierend auf "TSP-Art" mit 10 000 Punkten.....	42
Abb. 25: Zeichnung des Mixed Mode Logos. Zeichendauer: 24 s	43

Inhalt

Abbildungsverzeichnis.....	III
Danksagung	VI
Entstehungsgeschichte.....	1
Zielsetzung.....	1
Anforderungen an den Plotter	2
Funktionale Anforderungen.....	2
Nicht-funktionale Anforderungen.....	2
Schnittstelle zwischen Plotter und Bildverarbeitungssoftware	3
Generation der Zeichenpfade	5
Vektorbasierte Pfadgeneration.....	5
Rasterbasierte Pfadgeneration	6
Kombination von Dithering und dem Traveling Salesman Problem	6
Schraffieren von Monochrombildern	8
Kombination von schraffierten Portraits	9
Ausblick: Generation von Zeichenpfaden über Neuronale Netze.....	10
Anwendung zur Ansteuerung des Plotters.....	11
Benutzeroberfläche.....	11
Anbindung des Plotters.....	13
Verwaltung von G-Code	13
Programminterne Datenstruktur	13
Import von G-Code.....	14
Optimierungen.....	14
Vorschau von G-Code.....	15
Ausblick	15
Konstruktion des Plotters.....	16
Verwendete Technologien	16
Kinematische Plattform	16
Linearführungen.....	17

Schrittmotoren.....	18
Schrittmotortreiber.....	22
Treiberplatine	24
Teile aus Eigenproduktion.....	28
Minenhalter	28
Display und SD-Karte.....	29
Grundplatte.....	29
Marlin-Firmware	30
Evaluation der Geschwindigkeit des Plotters.....	31
Ausblick	31
Mixed Mode Shield v2.0	31
Eignung des Plotter-Systems für die Verwendung als Messedemonstrator	33
Literaturverzeichnis.....	35
Erklärung zur Abschlussarbeit	37
Anhang	38
Beispiel einer G-Code Datei	38
Detaillierte Zeichnungen	40
Dateistruktur auf dem beiliegenden Datenträger	44

Danksagung

Allen voran möchte ich meinen Dank an Paul Kissinger richten, der mir während meiner Arbeit stets beraten und unterstützend beiseite stand. Des Weiteren möchte ich mich bei Richard Kölbl bedanken, der mir viel über das Verfassen wissenschaftlicher Texte beibringen konnte. Paul Dolezal gebührt Dank, da er mir bei der Produktion der 3D-gedruckten Teile im Fablab der Hochschule Augsburg unterstützend zur Seite stand. Zu guter Letzt bleibt der Dank an Vanja Küter, für moralischen Support und die Dienste als Fotomodell für die Demonstration der Zeichnungen.

Entstehungsgeschichte

Wie bereits viele andere Studenten bzw. Mitarbeiter der Mixed Mode GmbH habe ich diese auf einer Messe kennen gelernt. Bei der Firmenkontaktmesse „Pyramid“ an der Hochschule Augsburg stellen sich Unternehmen vor, um Studenten für Praktika, Abschlussarbeiten oder auch Jobs zu begeistern. Mehr als jedes auch noch so informative Plakat am Stand von Mixed Mode zog mich eine große, blaue, sich unter großer Lärmabgabe bewegende Apparatur vom Flur, hin zu den Aufbauten der Firma.

Der Legosteinsortierer dient Mixed Mode schon lange als Messedemonstrator. So beeindruckend die Maschine auf Messen auch ist, birgt ihr Einsatz doch auch Nachteile. Besonders ihre Größe und das hohe Gewicht machen Einsatz und Transport ebenso umständlich wie anstrengend. Als Alternative einigte man sich auf einen Stiftplotter, der mit einem Stift das Portrait von Messebesuchern auf Papier bringen sollte.

Dazu soll von den Messebesuchern ein Foto aufgenommen werden, das der Plotter in den folgenden Minuten auf ein Blatt Mixed Mode Briefpapier zeichnet. Dadurch wird das Interesse der Besucher auf die Maschine und damit auch auf den Stand und das Angebot von Mixed Mode gelenkt. Außerdem dient dies als Einstiegspunkt für ein Gespräch. Verlässt der Besucher nun den Stand wieder, nimmt er sein Portrait mit und wird sich dank des Firmenlogos auf dem Papier auch an Mixed Mode erinnern.

Zielsetzung

Um das vorangehend beschriebene Szenario verwirklichen zu können, benötigt die Firma Mixed Mode zwei Dinge:

1. Einen Stiftplotter, der das Portrait auf Papier bringt.
2. Eine Software, die fotografierte Portraits so verarbeitet, dass der Stiftplotter sie zeichnen kann.

Im Folgenden wird sich zeigen, dass die Entwicklung beider Bestandteile des Systems weitgehend unabhängig voneinander betrachten lässt.

Da Mixed Mode bereits beides aus einer früheren Bachelorarbeit¹ zur Verfügung stand, gingen die Planungen in der Anfangsphase dieser Arbeit in die Richtung, das bereits bestehende System

¹Verfasst von Adrian M. Kast, veröffentlicht unter der Creative Commons Attribution License Germany (<http://creativecommons.org/licenses/by/3.0/de>). Eine Kopie der Arbeit wird auf dem Beigelegten Datenträger zur Verfügung gestellt.

aus Plotter und Software zu optimieren und so anzupassen, dass sie den Anforderungen der Messe genügen würden.

Nach kurzer Inspektion von Hard- und Software stand jedoch fest, dass sowohl die Maschine als auch die Software durch eine Neuentwicklung ersetzt werden. Die Verbesserung auf die gewünschten Anforderungen hätten einen größeren Entwicklungsaufwand bedeutet als eine Neuentwicklung.

Anforderungen an den Plotter

Funktionale Anforderungen

Die wichtigste Ursache dafür, dass das bereits vorhandene Selfie-Plotter-System nicht eingesetzt wurde, war dessen vergleichsweise lange Druckzeit von etwa 10 Minuten pro Bild. Daher war die wichtigste Anforderung an den Plotter eine hohe Zeichengeschwindigkeit. Während zu Beginn des Projektes geplant war, ein Portrait in unter 5 Minuten zeichnen zu können, wurde diese Anforderung schnell relativiert, da die Zeichendauer mit der Qualität des Bildes steigt². Stattdessen war das Ziel bei der Konstruktion der Maschine, die Einzelschritte eines Zeichenvorgangs größtmöglich zu beschleunigen. Dies konzentriert sich auf 3 wesentliche Zeichenoperationen:

1. Das Verfahren des Stifts
2. Das Anheben des Stifts nach einem Zeichenvorgang
3. Das Absenken des Stifts zu Beginn eines Zeichenvorgangs

Bei den Entscheidungen, die im Laufe der Entwicklung getroffen wurden, hat die Beschleunigung dieser drei Vorgänge stets die oberste Priorität.

Nicht-funktionale Anforderungen

Um beim Einsatz des Plotters auf der Messe nicht nur schnell ein Bild zu haben, sondern auch noch die Messebesucher für die Technik des Systems begeistern zu können, gab es neben der Geschwindigkeitsoptimierung noch andere Designanforderungen.

Eine der anspruchsvollsten nicht-funktionalen Anforderungen war, dass der Plotter möglichst laut sein sollte, um eventuelle Standbesucher schon aus großer Entfernung auf den Demonstrator aufmerksam zu machen. Da die verwendeten Komponenten grundsätzlich eine leise Operation begünstigen, wurde entschieden, die Geräusche direkt über Lautsprecher zu generieren.

Um auch vom Messeteam der Firma gerne benutzt zu werden, sollte die Bedienung des Plottersystems möglichst einfach gehalten werden. Ebenso sollten alle benötigten Komponenten leicht transportierbar sein.

² Mehr dazu im Kapitel „Generation der Zeichenpfade“

Schnittstelle zwischen Plotter und Bildverarbeitungssoftware

Wie bereits erwähnt, lässt sich der hauptsächliche Entwicklungsaufwand relativ gut auf die zwei Hauptbestandteile des Plotter-Systems aufteilen. Die entwickelte Bildverarbeitungssoftware lässt sich sowohl mit einem anderen Plotter nutzen als auch der Plotter mit einem anderen Generator für Zeichenpfade. Dies liegt primär an der in Teilen nach DIN 66025/ISO 6983 zertifizierten Schnittstelle, meist „G-Code“ genannt. G-Code spezifiziert einen Programmstandard, der alle Bewegung der Achsen von CNC³-Maschinen und die Schaltbefehle der Maschinen beinhaltet. Diese sind in der richtigen Reihenfolge aneinandergereiht (Hans B. Kief, 2015). Während Programmanweisungen in G-Code zunächst für zerspanende CNC-Fräsen konzipiert war, fand er sich schnell auch in anderen CNC-Maschinen wieder.

Auch der Plotter führt G-Code-Programmanweisungen aus. Die für den Zeichenbetrieb wichtigsten Anweisungen und ihre Bedeutung bei der Verwendung des Plotters sind nachfolgend erläutert.

Tabelle 1: Zum Zeichnen eines Portraits benötigte G-Code Anweisungen

<i>Befehl</i>	<i>Anwendungsbeispiel</i>	<i>Erläuterung</i>
G1	G1 X126.24 Y26.3 F9000	Lineare Bewegung zu den Koordinaten (126,24, 26.3) mit einer Geschwindigkeit von max. 9000mm/min (140mm/s)
G28	G28 X0 Y0	Finde die Nullreferenz der X- und Y-Achse („Homing“)
M106	M106	Anheben des Stifts
M107	M107	Absenken des Stifts
G4	G4 P30	Warte für 30ms

Dieser begrenzte Kommandosatz reicht aus, um Zeichnungen mit dem Plotter zu erstellen. Streng genommen sind die Befehle G28 und G4 nicht notwendig – jedoch müsste die Maschine ohne sie manuell referenziert (Der Werkzeugkopf beim Start auf die Nullposition gebracht) werden und die Zeichnung wäre verfälscht, da der Plotter nach dem Einleiten des Absenkvorgangs des Stifts nicht wartet, bis dieser auf dem Papier aufgetroffen ist.

Darüber hinaus können mit weiteren Befehlen zusätzliche Funktionen des Plotters gesteuert werden. Sie sind zwar für den Zeichenvorgang nicht erforderlich, werden aber für bestimmte Zusatzfunktionen verwendet. Zu diesen Befehlen gehören unter anderem die Folgenden:

³ Computerized Numerical Control

- G02⁴ X10 Y10 I0 J10
 - Kreisförmige Bewegung im Uhrzeigersinn von der aktuellen Position nach (10, 10) mit dem Mittelpunkt der Kreisbewegung 10 mm in positiver Richtung der Y-Achse von der aktuellen Position versetzt
- G03 X10 Y10 I0 J10
 - Kreisförmige Bewegung gegen den Uhrzeigersinn von der aktuellen Position nach (10, 10) mit dem Mittelpunkt der Kreisbewegung 10 mm in positiver Richtung der Y-Achse von der aktuellen Position versetzt
- M150 R255 U255 B255
 - Setzt die Farben der integrierten RGB-LED-Beleuchtung gemäß der Parameter R, U (Grün) und B. Bei dem hier gezeigten Beispiel wird die Beleuchtung weiß.
- M75
 - Startet einen Timer, der die bisher vergangene Druckzeit auf dem integrierten Display des Plotters anzeigt. Gerade um die Druckzeit zu optimieren ist es hilfreich, eine Zeitangabe direkt auf dem Gerät einsehen zu können.
- M77
 - Beendet den Timer.
- M201 X2000 Y2000
 - Bestimmt die Beschleunigung der Schrittmotoren in $\text{Schritten} \times \frac{1}{s^2}$
- M18
 - Deaktivierung der Schrittmotortreiber. Dies geschieht nach jeden Plottvorgang automatisch, um die Treiber abkühlen zu lassen.

Die G-Code-Befehle können dem Plotter sowohl über eine serielle Schnittstelle einzeln übermittelt werden als auch in einer Datei bereitgestellt, auf die die Maschine über eine Speicherkarte Zugriff erhält.

Im folgenden Kapitel sollen die notwendigen Arbeitsschritte beschrieben werden, um eine G-Code-Datei aus einem Foto zu erzeugen, die dann von der Maschine ausführbar ist.

Die Konstruktion der Maschine selbst wird anschließend im Kapitel „Konstruktion des Plotters“ erläutert.

⁴ Nicht alle G-Code Interpreter sind in der Lage, Radialbewegungen umzusetzen. Deshalb wird in der Software des Plotters darauf verzichtet.

Generation der Zeichenpfade

Als eine der größten Herausforderungen im Verlauf der Arbeit erwies sich die Nachforschung über die Generation der Zeichenpfade. Hierbei wurden mehrere Ansätze evaluiert, um eine geeignete Technologie zur Verwendung für den Messedemonstrator wählen zu können.

Vektorbasierte Pfadgeneration

Die Nachforschungen zur Pfadgeneration begannen bei der Software, die im Rahmen der vorhergehenden Bachelorarbeit der Pfadgeneration diente. Herr Kast hatte für seine Software einen Ansatz gewählt, der auf einer Kantendetektion basierte (Kast, 2017). Die Generation der Zeichenpfade basierte auf folgenden Schritten:

- Durchführung einer Kantendetektion auf den Zuschnitt
- Vektorisierung des Kantenbilds
- Konvertierung des Vektorbilds in G-Code-Kommandos



Abb. 1: Bildverarbeitung in der Software von A. Kast (Kast, 2017)

Leider lassen die aus der Kantenerkennung generierten Zeichenpfade eine gewisse Ästhetik missen (Abb. 2), da die Wiedergabe von Kanten allein nicht für einen hohen Wiedererkennungswert der gezeichneten Person ausreicht. Dies liegt unter anderem daran, dass diese Vorgehensweise flächenbezogene Helligkeitsinformationen außer Acht lässt.

Ein weiterer Nachteil der Kantenerkennung zeigt sich durch die übermäßige Betonung von Artefakten wie Hautunreinheiten oder Bildrauschen. Als Vorteil der rein kantenbasierten Generation sind die vergleichsweise kurzen Pfade.

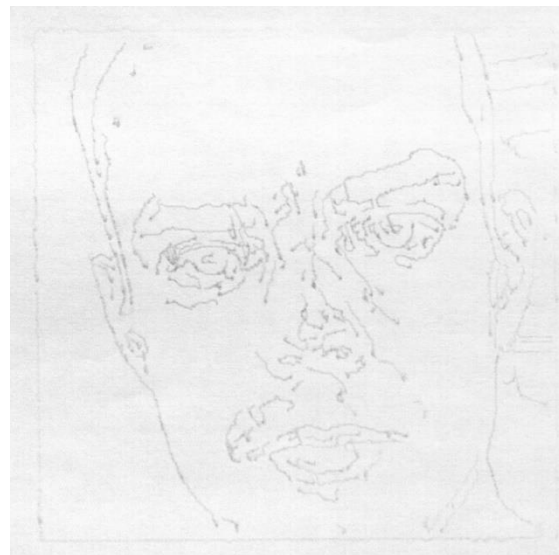


Abb. 2: Zeichnung der generierten Pfade (Kast, 2017)

Für die Implementation dieser Arbeit sollte ein anderer Ansatz dienen, der einen größeren Wiedererkennungswert der Zeichnungen verspricht.

Rasterbasierte Pfadgeneration

Um flächenbezogene Helligkeitsinformationen mit in die Zeichnung einfließen zu lassen, wurden Vorgehensweisen untersucht, die auf dem Pixelraster der digitalen Bilder basieren. Um die Pfadgeneration zu veranschaulichen, sind zu den jeweiligen Ansätzen Beispiele abgebildet, die auf dem Graustufenbild aus Abb. 3 basieren.



Abb. 3: Graustufen-Portraitfoto

Das Foto dient zugleich als Beispiel für ein ideales Ausgangsbild. Das Modell steht mit einem Abstand von ca. 50 cm vor einem weißen Hintergrund und wird unmittelbar von vorne beleuchtet. Eine Systemkamera mit möglichst großem Dynamikumfang bildet hierbei einen hilfreichen Kontrast zum Hintergrund und ein aufsteckbarer Blitz sorgt für eine optimale Ausrichtung. Ist der Blitz bei der Beleuchtung oberhalb des Objektivs, sorgt dies für eine symmetrische Ausleuchtung. Deshalb empfiehlt es sich nicht, die Kamera im Querformat zu orientieren.

Kombination von Dithering und dem Traveling Salesman Problem

Das Unternehmen „Evil Mad Science LLC“ stellt auf seiner Website das Programm „StippleGen2“ zur Verfügung, das aus Bilddateien Punktzeichnungen und „TSP-Art“ erstellt. Dabei werden über dem Ausgangsbild eine festgelegte Menge an Punkten verteilt. An dunkleren Regionen des Bildes werden mehr Punkte verteilt, an helleren weniger bis gar keine (vgl. Abb. 4).

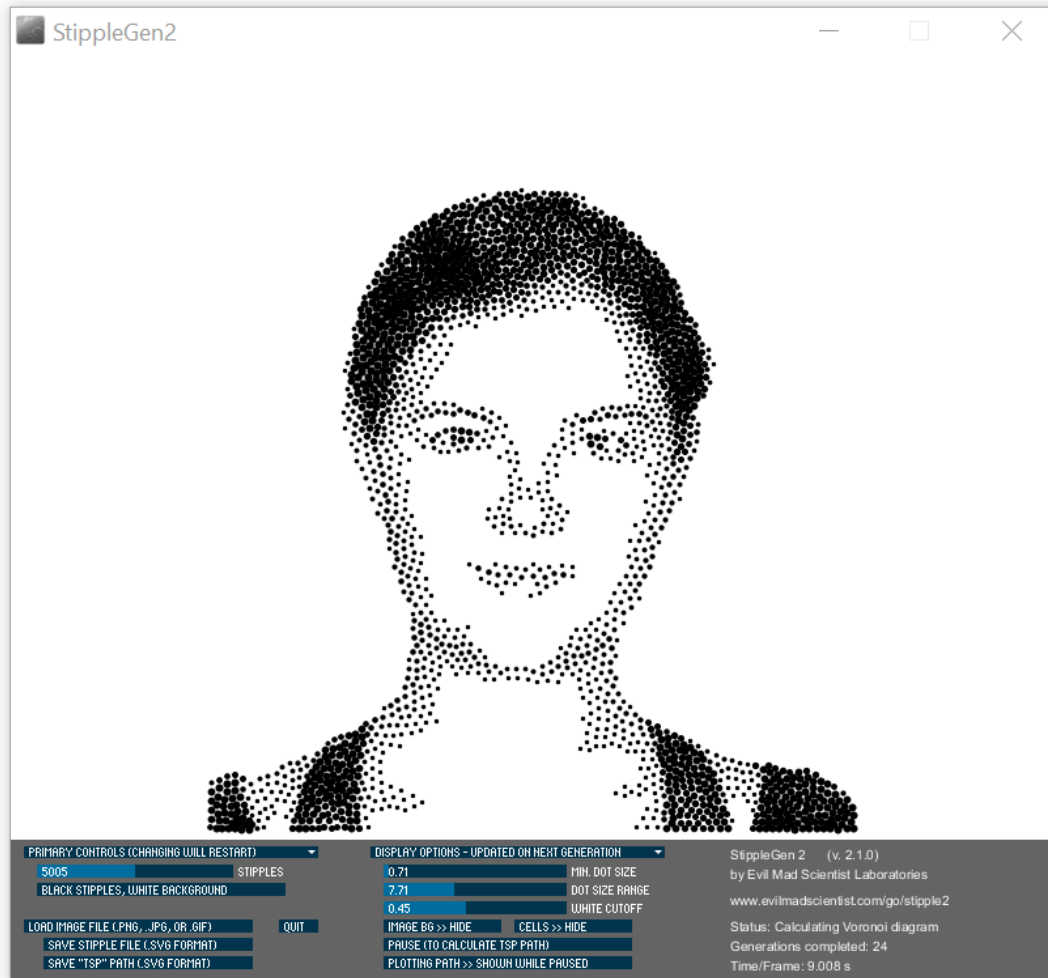


Abb. 4: Eine mithilfe der StippleGen2-Software erstellte Punktzeichnung (Oskay, 2017)

Der Algorithmus, der die Punkte nach Helligkeitswerten verteilt, basiert auf einem gewichteten Voronoi-Diagramm. Nachdem die Punktverteilung über das Ausgangsbild abgeschlossen ist, wird auf Basis der Punkte ein Zeichenpfad generiert. Hierzu existieren mehrere Ansätze.

Eine Möglichkeit ist, jeden Punkt als Kreis zu zeichnen. In stärker gewichteten Regionen kann der Radius der Kreise vergrößert werden, um die Region dunkler darzustellen. Eine beispielhafte Umsetzung ist im Anhang (Abb. 23: Durch Kreise unterschiedlicher Radien gewichtete Punktzeichnung) zu sehen.

In „StippleGen2“ wird der Zeichenpfad jedoch erstellt, indem ein kontinuierlicher Pfad durch sämtliche generierten Punkte gebildet wird, der jeden Punkt genau einmal schneidet. Der berechnete Pfad wird dahingehend optimiert, möglichst kurz zu sein. Dies ist eine bekannte Problematik und wird in der Informatik als „Travelling Salesman Problem“ bezeichnet (Gutin & Punnen, 2007). Deshalb bezeichnet der Hersteller der StippleGen2-Software die daraus entstandenen Kunstwerke als „TSP-Art“.

Mithilfe der Software lassen sich die generierten Zeichenpfade gut skalieren, beispielsweise durch Variation der Anzahl von Punkten („Stipples“). Eine beispielhafte Grafik ist in Abb. 24 zu sehen.

Die Berechnung der Punkteverteilung ist vergleichsweise komplex. Die Berechnung eines Bildes auf Basis von 10 000 Stipples benötigt selbst bei großer verfügbarer Rechenleistung⁵ ca. 25 Minuten. Zudem ist der Wiedererkennungswert verglichen mit der vektorbasierten Methode zwar subjektiv größer, jedoch ist das Ergebnis (besonders in Kombination mit dem hohen Berechnungsaufwand) nicht für die Verwendung im Plotter-System ausreichend.

Schraffieren von Monochrombildern

Der Plotter kann mit dem Stift lediglich zwei Zustände auf dem Papier abbilden:

1. Den Stift zum Zeichnen auf das Papier absetzen, sowie
2. Den Stift nicht auf das Papier zu setzen.

Aus diesem Grund ist es nicht möglich, verschiedene Graustufen mit derselben Linie abzubilden. Konvertiert man nun das Portraitfoto einer Person zu einem Monochrombild⁶, lassen sich die daraus entstandenen schwarzen Flächen mit einer Schraffur abbilden (vgl. Abb. 5). Ist der Linienabstand der Schraffur ausreichend klein, wird die Schraffur als Fläche wahrgenommen.



Abb. 5: Schraffur eines monochromen Portraits. Links das monochrome Bild, rechts die daraus resultierende Schraffur⁷.

Das Monochrombild, sowie das schraffierte Portrait in Abb. 4 wurde mit der im Rahmen dieser Bachelorarbeit erstellten Software generiert. Um das Monochrombild zu erstellen, wurde jeder Pixel des Ausgangsfotos, dessen Helligkeit unter einem Schwellwert liegt (in diesem Falle 127),

⁵ Verwendete Hardware: Intel Xeon 1280-V6, 64 GB RAM; Samsung 970-Evo

⁶ Jeder Pixel hat nur 1 Bit Farbinformation; das Bild ist also schwarz-weiß

als weiß dargestellt, während alle Pixel mit einem Helligkeitswert von 127 oder größer als schwarz dargestellt werden. Der Schwellwert wird als Threshold bezeichnet. Die Schraffur erfolgt mit einem Linienabstand von 8 Pixeln in horizontaler Richtung und einem Winkel von 45°.

Durch Variation des Linienabstands lässt sich die wahrgenommene Helligkeit der Fläche verändern. Rücken die Linien näher zusammen, erscheint die entstandene Fläche dunkler. Vergrößert sich der Abstand, scheint das Bild heller.

Während das schraffierte Portrait dem Ausgangsfoto zwar ähnelt, ist der subjektive Wiedererkennungswert für den Messeinsatz noch nicht ausreichend.

Kombination von schraffierten Portraits

Eine Möglichkeit, den Wiedererkennungswert zu steigern, ist die Kombination von mehreren schraffierten Portraits mit unterschiedlichen Schraffuren und Thresholds der Monochrombilder.

Die Überlagerung der einzelnen Schraffuren wirkt dabei ähnlich wie die Darstellung von verschiedenen Graustufen. In Bereichen, in denen sich mehrere Schraffuren überlagern, erscheint das Bild deutlich dunkler. Auf diese Art lassen sich Graustufenbilder zeichnen, bei denen die Anzahl der Graustufen der Menge von überlagerten schraffierten Bildern mit unterschiedlichem Threshold entspricht.



Abb. 6: Durch kombinierte Schraffur entsteht die Illusion eines Bildes mit 3 Graustufen

Das schraffierte Portrait in Abb. 6 basiert auf 3 Monochrombildern, bzw. 3 Schraffuren mit den folgenden Eigenschaften:

1. Threshold: 145, Winkel: 25°, horizontaler Linienabstand: 5 Pixel
2. Threshold: 40, Winkel: 75°, horizontaler Linienabstand: 8 Pixel
3. Threshold: 127, Winkel: 45°, horizontaler Linienabstand: 10 Pixel

Das gezeichnete Portrait des generierten Pfades befindet sich im Anhang (**Fehler! Verweisquelle konnte nicht gefunden werden.**).

Bereits mit 3 Monochrombildern und 4 Graustufen hat die Abbildung eine deutlich höhere Qualität. Je nach gewünschtem Ergebnis lassen sich beliebig viele Graustufen schraffieren. Durch niedrigere Linienabstände lässt sich die Qualität weiter erhöhen. Beides verlängert jedoch die für den Plottvorgang benötigte Zeit. Während vieler Tests hat sich die Schraffur mit 4 Graustufen und 3 Monochrom-Thresholds jedoch als bester Kompromiss zwischen Zeit und Darstellungsqualität bewährt.

Da das Kombinieren von schraffierten Portraits sehr gut in Abhängigkeit von Druckzeit und Anforderungen skalierbar ist und die Bilder einen guten Wiedererkennungswert besitzen, wurde beschlossen, dieses Verfahren für das Plotter-System zu verwenden.

Ausblick: Generation von Zeichenpfaden über Neuronale Netze

Eine weitere interessante Methode ist die Generation von Zeichenpfaden mithilfe von Neuronalen Netzen (NN). Ein Beispiel hierfür ist ein Neuronaler Algorithmus, der von Wissenschaftlern der Universität Tübingen entwickelt wurde. Das unter „deepart.io“ verfügbare System nutzt neuronale Netze, um stilistischen Inhalt von Bildern zu trennen und zu rekombinieren. Dies dient dann als Basis für die Kreation von neuen, künstlichen Bildern (Leon A. Gatys, 2015). Auf der Homepage „deepart.io“ kann mithilfe dieser Technologie der Visuelle Stil eines Bildes auf ein zweites angewandt werden.

Einzelne Versuche zur Nutzung dieser Services zur Generation von Zeichenpfaden führten leider zu keinem Erfolg (Abb. 7). Eine Adaption des bestehenden Netzwerks könnte jedoch unter Umständen Pfade erstellen, die eine noch höhere künstlerische Qualität besitzen.

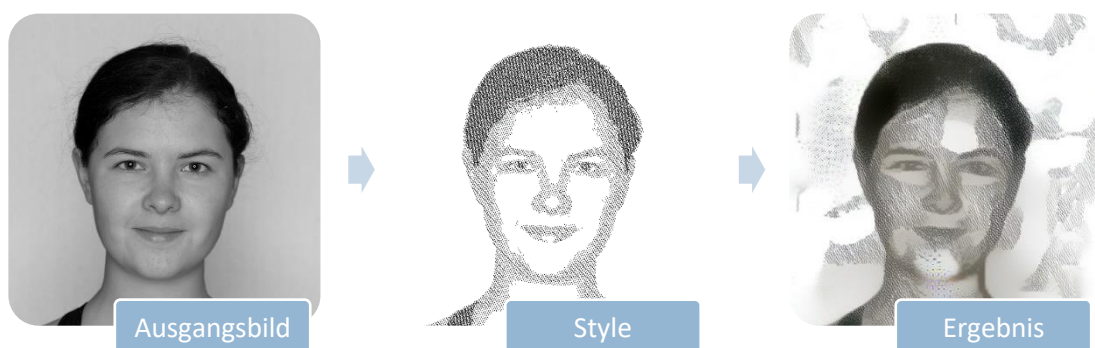


Abb. 7: Anwendung eines visuellen Stils auf ein Bild zur Pfadgeneration (DeepArt UG, 2019)

Leider ist die Entwicklung von neuronalen Netzen sehr komplex und zeitaufwändig und würde den Rahmen dieser Arbeit sprengen. Deshalb wurde der Ansatz hier nicht weiter untersucht.

Anwendung zur Ansteuerung des Plotters

Zum Einsatz des Plotters auf Messen wurde eine Software entwickelt, die den benötigten Funktionsumfang zur Zeichnung von Portraits komplett abbildet.

Die Software „PlotCon“ wurde komplett in Python entwickelt. Großer Wert wurde zudem auf Plattformunabhängigkeit gelegt. Getestet wurde die Software unter Windows und Linux. Der Quellcode ist auf dem beiliegenden Datenträger enthalten.

Benutzeroberfläche

Die Benutzeroberfläche der Applikation wurde mit TkInter erstellt. TkInter basiert auf Tcl/Tk und ist sowohl auf Linux als auch Windows verfügbar.

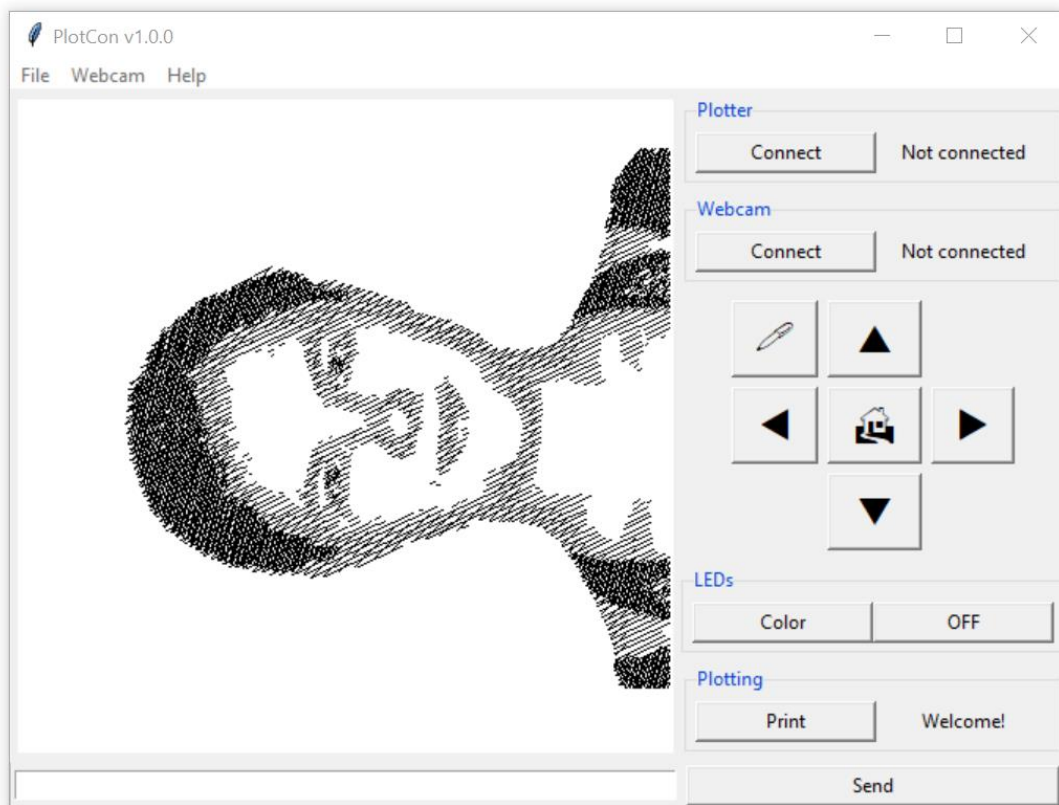


Abb. 8: Hauptfenster der PlotCon-Software. Das Vorschaufenster zeigt den geladenen G-Code

Das Hauptfenster der Anwendung besitzt ein Hauptmenü, ein Vorschaufenster, mehrere Steuerbuttons und eine Befehlszeile unterhalb des Vorschaufensters.

Ist der Plotter über USB mit dem Computer verbunden, auf dem die Anwendung ausgeführt wird, so verbindet sich die Software mit einem Klick auf den „Connect“-Button. Neben dem Button wird dies angezeigt.

Über die in Form von quadratischen Buttons umgesetzten Steuerelemente kann der Werkzeughalter des Plotters verfahren, der Stift abgesenkt bzw. angehoben und die Achsen des Plotters referenziert werden.

In der Gruppe „LEDs“ befinden sich Steuerelemente zur Auswahl von Farbe und Schaltzustand der in den Plotter eingelassenen LED-Streifen. Auch wenn die LEDs beim Drucken der Statusanzeige des Drucks dienen, lassen sie sich im Ruhezustand über die Software bedienen.

Über den Button „Print“ wird der im Vorschaufenster geladene G-Code über USB zur Ausführung an den Plotter gesendet.

Die Eingabezeile am unteren Rand des Fensters dient zur manuellen Eingabe von einzelnen G-Code Befehlen, die mit einem Klick auf den Button „Send“ an den Plotter gesendet werden. Ist das Textfeld aktiv, kann mit den Pfeiltasten „Hoch“ und „Runter“ zwischen den zuletzt abgesetzten Befehlen umschalten, ähnlich der Historie in einer Kommandozeile.

Die Schaltfläche „Webcam“ wird erst in einer zukünftigen Version aktiv sein und eine Webcam zur Aufnahme von Portraits verbinden. Die Implementation dieser Funktionalität war jedoch nicht vorgesehen.

Sowohl über das „File“-Menü als auch mit der Tastenkombination „Strg + O“ öffnet sich ein Dialog, der den Nutzer dazu auffordert, eine „G-Code“- bzw. Bilddatei zu öffnen.

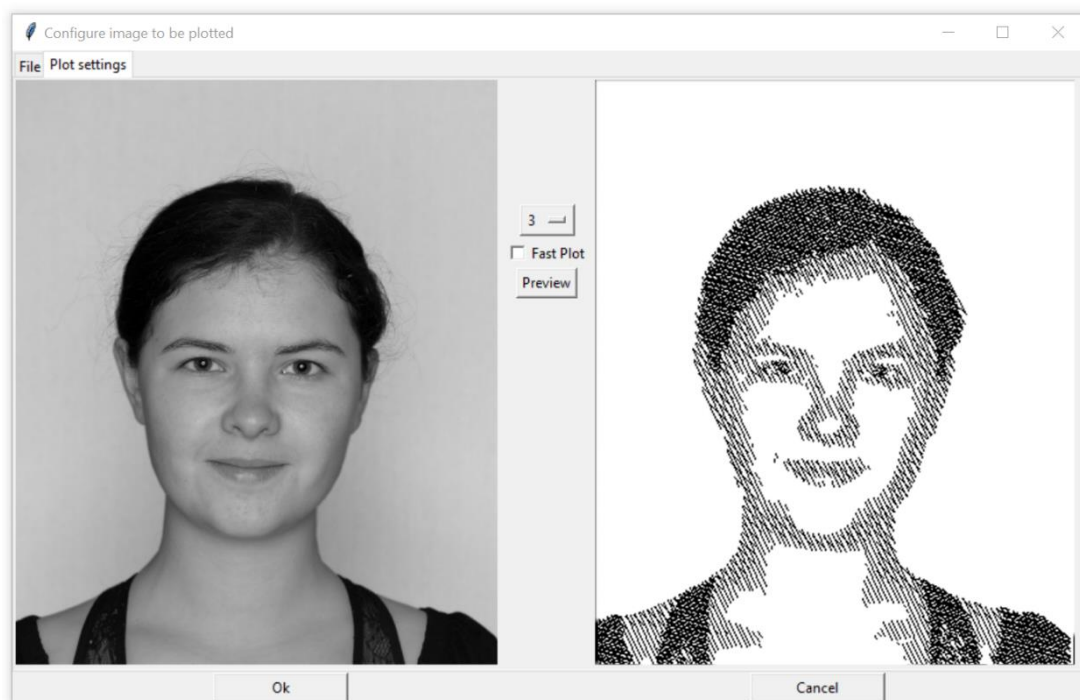


Abb. 9: Dialog zur Auswahl der Druckparameter

Ist die gewählte Datei eine Bilddatei, so öffnet sich der in Abb. 9 dargestellte Dialog. Hier kann der Nutzer auswählen, wie viele Graustufen verwendet werden sollen, ob eine verkleinerte Version des Bildes generiert wird („Fast Plot“) und ob er in diesem Fenster eine Vorschau des Zeichenpfades sehen möchte. Bestätigt er die Einstellungen mit einem Klick auf „Ok“, schließt sich der Dialog und das Hauptfenster zeigt die Vorschau des generierten Zeichenpfades.

Wählt der Benutzer eine G-Code-Datei, so wird diese importiert⁸ und in die Vorschau des Hauptfensters geladen. Die Darstellung des Vorschau Fensters ist um 90° gegen den Uhrzeigersinn gedreht, um die Orientierung des Plotters abzubilden, der das Papier in gleichem Maße gedreht einspannt.

Anbindung des Plotters

Die Verbindung des Hostcomputers zum Plotter wird über eine serielle Verbindung realisiert. In der Hostsoftware kommt hierfür die Bibliothek „pySerial“ zum Einsatz.

Wird eine G-Code Datei gedruckt, so wird diese Zeile für Zeile über die serielle Schnittstelle übertragen. Nach jeder übertragenen Zeile pausiert das Programm und wartet auf eine Antwort des Plotters. Dieser sendet ein „ok“ zurück und bestätigt damit die Ausführung des Kommandos. Erst nach dieser Bestätigung wird die nächste Zeile der Datei zum Plotter übertragen.

Verwaltung von G-Code

Die für die Verwaltung von G-Code zuständigen Programmstrukturen wurden speziell für die Verwendung mit dem Plotter entwickelt und sind daher auch an diese Anforderungen angepasst. Verglichen mit anderen Lösungen zur G-Code Verwaltung wie z.B. der Software „Repetier Host“ fällt zunächst auf, dass PlotCon nur zweidimensionalen G-Code unterstützt.

Programminterne Datenstruktur

Die Spezialisierung der Anwendung für den Plotter spiegelt sich auch bei der internen Datenstruktur wider (vgl. Abb. 10). Ein Zeichenpfad wird durch ein Objekt der Klasse „Path“ repräsentiert. Er besitzt das Attribut „strokes“, eine Liste aus einem oder mehreren Objekten der Klasse „Stroke“.

Jedes Objekt der Klasse „Stroke“ besitzt eine Liste („points“), in welcher beliebig viele Koordinaten gespeichert sind. Dieses Objekt repräsentiert eine vom Plotter durchgehend gezeichnete Linie.

⁸ Mehr zum Import im Kapitel „Import von G-Code“

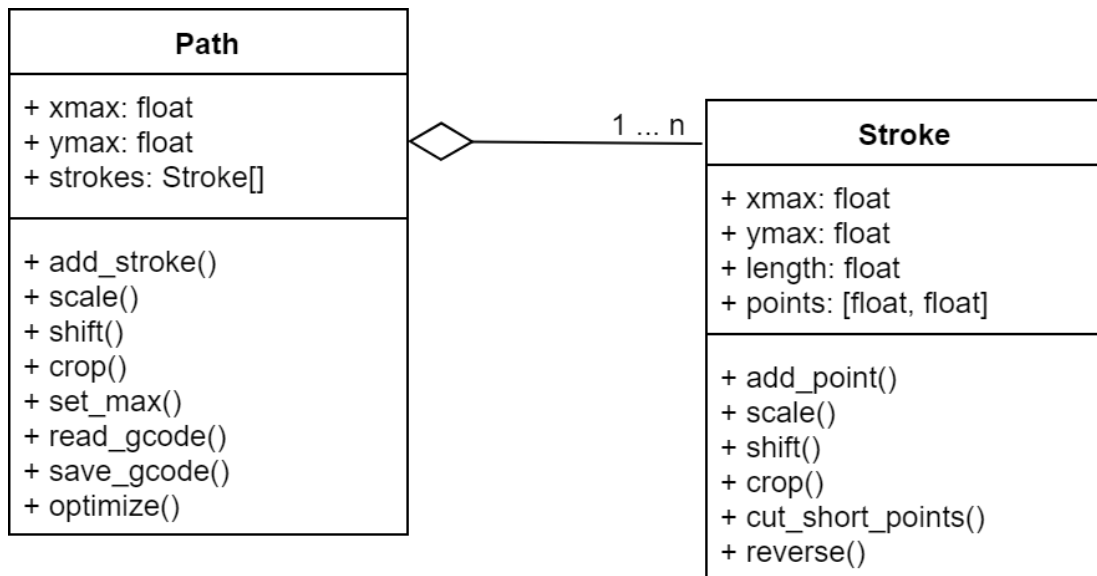


Abb. 10: UML-Diagramm der Klassen "Path" und "Stroke". Beide haben ähnliche Attribute und Methoden, da bei beiden ähnliche Operationen angewandt werden.

Die simple Struktur vereinfacht Methoden wie die Skalierung von Zeichenpfaden. Dafür wird die Funktion für die Klasse Stroke implementiert und vom Objekt der Klasse Path auf allen strokes angewandt.

In der Klasse Path sind Methoden zum Einlesen und Optimieren von G-Code implementiert, die zusammen mit der Vorschaufunktion nachfolgend erläutert werden.

Import von G-Code

Der Grund für die Implementation der Methode zum Einlesen von G-Code ist die Notwendigkeit einer Vorschau. Werden G-Code-Dateien in großen Mengen für einen späteren Druck gespeichert, bzw. von externen Tools erstellt, ist es oftmals hilfreich, eine Vorschau des Zeichenpfades angezeigt zu bekommen. Daher ist es notwendig, fertigen G-Code in die programminterne Datenstruktur importieren zu können.

Beim Importvorgang einer G-Code-Datei Werden nun die Zielpositionen von G01-Befehlen als Punkte zu einem Stroke hinzugefügt. Die Befehle „M106“ und „M107“ dienen hier als Indikator für den Beginn eines neuen Strokes. Befehle, die zwischen dem Befehl zum Anheben und dem zum Absenken liegen, werden ignoriert.

Obwohl die PlotCon-Software aufgrund von Kompatibilitätsgründen auf G02/G03 Befehle verzichtet, kann das G-Code-Import-Tool mit diesen Befehlen umgehen. Beim Importvorgang werden die Radialbewegungen jedoch interpoliert und durch Geraden angenähert.

Optimierungen

Ist die Pfadgeneration abgeschlossen, werden auf den Pfad Optimierungen angewandt, die den Zeichenvorgang beschleunigen.

Zunächst werden sämtliche Geraden aussortiert, deren Länge unter 0,5 mm liegt. Da der verwendete Stift eine Breite von ca. 1 mm hat, sind diese ohnehin nicht sichtbar. Ist die Gerade innerhalb eines Strokes, so wird der Punkt am Ende der Geraden gelöscht und es bildet sich eine neue Gerade zum nächsten Punkt. Besitzt ein Stroke nur diese beiden Punkte, wird der gesamte Stroke gelöscht.

Die zweite Optimierung optimiert die Zeichenrichtung der Punkte innerhalb eines Strokes. Sobald der Endpunkt des nachfolgenden Strokes näher an der aktuellen Position liegt als sein Anfangspunkt, wird die Reihenfolge der Punkte dieses Strokes umgedreht. Besonders bei der Zeichnung von schraffierten Portraits spart dies Zeit, da die Zeichenrichtung hierbei für die einzelnen Striche alterniert.

Vorschau von G-Code

Die Vorschaufunktion basiert auf der Bibliothek „OpenCV“. Obwohl diese für computergestützte Bilderkennung konzipiert ist, findet sie aufgrund der einfachen Verwaltung von Bilddateien auch in PlotCon Verwendung.

Um eine Vorschau zu generieren wird zunächst ein OpenCV-Image-Objekt erzeugt, dessen Größe den X- / Y-Max-Werten des Path-Objektes entspricht. Anschließend iteriert die Vorschaufunktion sämtliche Strokes und zeichnet zwischen deren Punkten Geraden auf das OpenCV-Bild. Ist dieser Vorgang abgeschlossen, kann das Bild als Vorschau eingebettet werden.

Ausblick

Für eine zukünftige Version der Software kann eine Optimierung des Zeichenpfades einen weiteren Geschwindigkeitsvorteil bringen. Dies kann z.B. mit einer Anwendung eines Travelling-Salesman-Algorithmus umgesetzt werden, der die bestmögliche Reihenfolge bestimmt, in der die Strokes zu zeichnen sind.

Ebenso soll die Anbindung zu Webcams realisiert werden, sodass Bilder direkt innerhalb der PlotCon-Software aufgenommen werden können.

Konstruktion des Plotters

Verwendete Technologien

In diesem Kapitel werden das Designkonzept und die Konstruktion des Plotters detailliert erläutert.

Die im Plotter verwendeten Komponenten und Konstruktionsprinzipien, die den größten Einfluss auf die Funktionalität und Eignung als Messedemonstrator haben, werden in diesem Kapitel vorgestellt und erläutert.

Das Funktionsprinzip eines CNC-gesteuerten Stiftplotters ähnelt stark dem eines 3D-Druckers, der mit dem FFF-Verfahren⁹ arbeitet. Bei diesem Druckverfahren wird die Düse des Druckers (analog zum Stift des Plotters) über der Druckplattform bewegt (bzw. im Fall des Plotters über das Papier). Sowohl beim Plotter als auch beim 3D-Drucker ist eine hohe Verfahrensgeschwindigkeit erforderlich.

Da sich die Anforderungen an die Hardware beider Geräte in vielen Bereichen überschneiden, habe ich beschlossen, auch beim Mixed Mode Plotter Technologien zu verwenden, die sich bereits im 3D-Druck etabliert haben. Dies reduziert die Entwicklungszeit maßgeblich und ermöglicht es mir deshalb Features zu realisieren, deren Entwicklungsaufwand allein genug Umfang für eine eigene Bachelorarbeit bietet.

Kinematische Plattform

Um sowohl Beschleunigung als auch Endgeschwindigkeit des Werkzeugkopfes zu maximieren, ist es essenziell, die bewegte Masse so gering wie möglich zu halten. Auch im 3D-Druck haben sich daher verschiedene Kinematiksysteme etabliert, die auf unterschiedliche Art die bewegte Masse so gering wie möglich halten (Carey, 2019). Für den Mixed Mode Plotter habe ich die CoreXY-Kinematik gewählt.

Die CoreXY-Kinematik zeichnet sich dadurch aus, dass die Motoren stationär angebracht sind und den Werkzeugkopf über ein System an Riemen und Umlenkrollen entlang der zwei Achsen bewegen. Da nicht, wie bei vielen anderen kinematischen Systemen, ein Motor mit einer Achse mit bewegt werden muss, reduziert sich die bewegte Masse deutlich und erlaubt daher höhere Beschleunigungen und eine größere Endgeschwindigkeit.

⁹ Fused Filament Fabrication

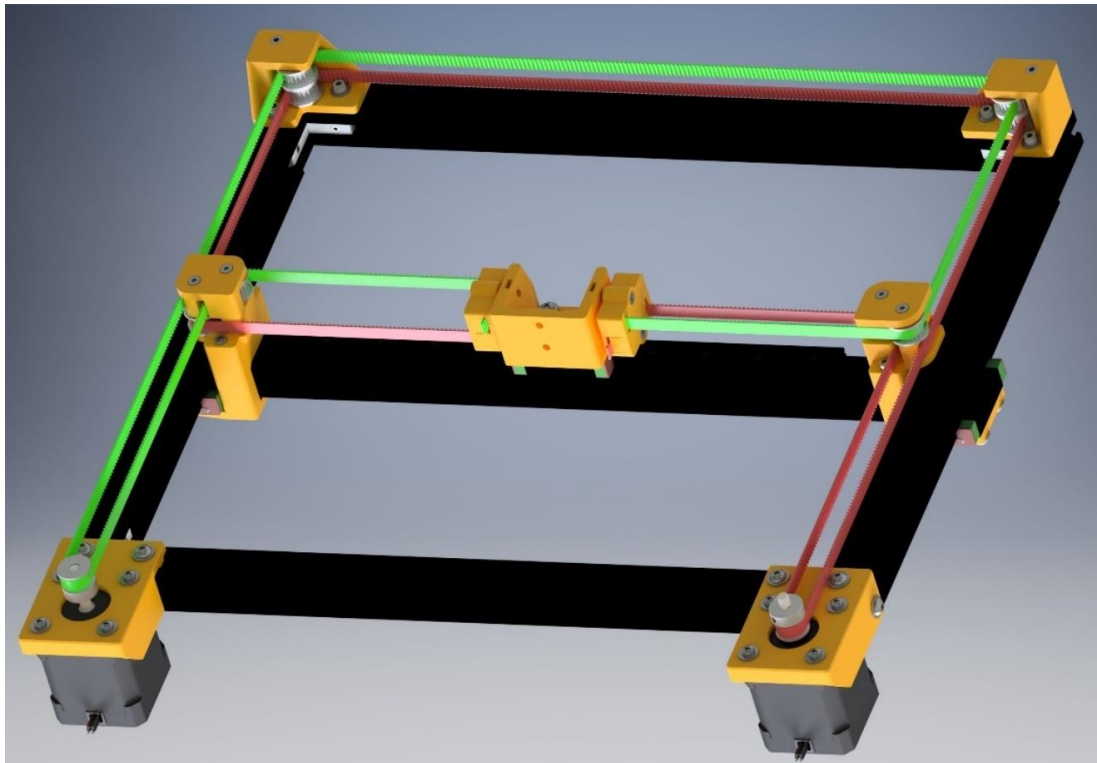


Abb. 11: Rendering aller an der Kinematik beteiligten Bauteile. Alle 3D-gedruckten Teile sind in orange dargestellt

Die Kraftübertragung von den Motoren auf den Werkzeughalter findet über zwei GT2-Riemen mit 6 mm Breite statt. Diese sind in dem Rendering auf Abb. 1 in roter, bzw. grüner Farbe dargestellt. Dreht sich einer der Motoren, so bewegt sich der Werkzeughalter entlang der Diagonalen im Rahmen, an deren Ende der Schrittmotor befestigt ist.

Soll der Stift nun beispielsweise entlang der Y-Achse¹⁰ verfahren werden, müssen sich beide Motoren gleichermaßen in entgegengesetzte Richtung drehen. Soll der Verfahrweg entlang der positiven Richtung der Y-Achse sein, muss sich der linke Motor gegen den Uhrzeigersinn drehen und der Rechte gleichermaßen im Uhrzeigersinn.

Da die Bewegungsanweisungen in kartesischen Koordinaten gegeben sind, werden diese von der Firmware des Plotters umgerechnet, so dass die Bewegungen der Motoren auf die Kinematik angepasst sind.

Linearführungen

Beide Achsen des Plotters werden von Profilschienenführungen des Typs MGN-12 (Siehe Abb. 12: Rendering einer MGN-12 Profilschienenführung) geführt. Auf der Profilschiene aus rostfreiem Stahl laufen Kugelumlaufwägen des Typs MGN-12H. Wie der Name bereits impliziert laufen im Inneren des Wagens Kugeln, die den Wagen auf der Profilschiene lagern. Dies erfordert eine hohe

¹⁰ Bei CNC-Geräten jeglicher Art verläuft die Y-Achse „nach hinten“; d.h. vom davorstehenden Benutzer weg, parallel zum Boden

Oberflächenhärte¹¹, da sonst die Kugeln die Lagerflächen deformieren. Die Passungen der Lager, die sich maßgeblich auf den Laufwiderstand, die Maximallast und Haltbarkeit des Linearsystems auswirken, sind ausschließlich durch die Abmessungen der Teile bestimmt und damit nach Fertigung von Schiene und Wagen nicht veränderbar. Daher ist bei der Herstellung von Schiene und Wagen besonders hohe Präzision erforderlich.

All dies sorgt dafür, dass Profilschienenführungen sowohl hohe Präzision als auch große Belastbarkeit haben. Durch die aufwändige Fertigung sind sie jedoch auch vergleichsweise teuer.

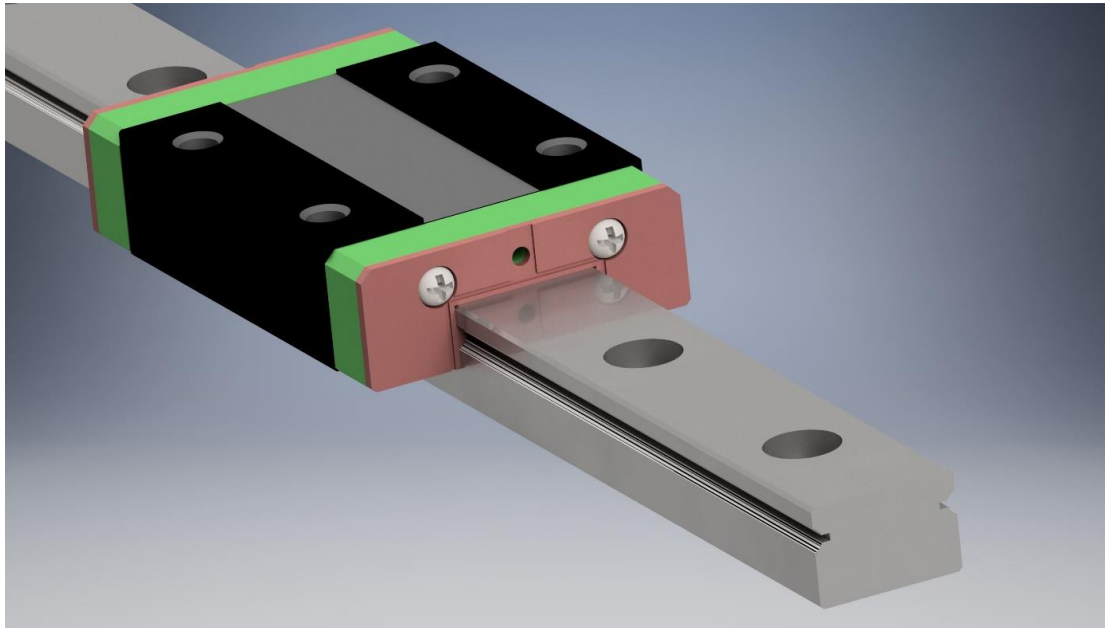


Abb. 12: Rendering einer MGN-12 Profilschienenführung mit aufsitzendem MGN12-H Kugelumlaufwagen

Die Profilschienen des Plotters sind zusätzlich schwarz brüniert, um sich optisch besser den restlichen Teilen des Plotters einzupassen. Die Brünierschicht weist einen ebenfalls hohen Härtegrad auf und ist sehr dünn (i.d.R. kleiner als 1 μm), weshalb sie sich kaum auf die Laufeigenschaften auswirkt.

Schrittmotoren

Um die Bewegung des Werkzeughalters präzise steuern zu können, wird dieser von bipolaren Schrittmotoren angetrieben. Schrittmotoren ohne Encoder-Feedback werden bei 3D-Drucker häufig eingesetzt und bieten eine günstige Lösung für einen präzisen Antrieb.

Das grundlegende Funktionsprinzip wird vereinfacht in Abb. 13 illustriert. Über die Statorspulen, die um den permanent magnetisierten Rotor angeordnet sind, wird ein Magnetfeld aufgebaut. Ist der Rotor nicht in entgegengesetzter Richtung des Spulenmagnetfelds ausgerichtet, wirkt auf ihn ein Moment.

¹¹ Bei den meisten Systemen hat sich eine Oberflächenhärte von ca. 58 HRC etabliert

Ist das Magnetfeld (und damit einhergehend das Moment) groß genug, dreht sich der Rotor und richtet sich damit aus. Um den Rotor nun weiter zu drehen, wird durch eine andere Beschaltung der Spulen das Statormagnetfeld verändert. Im Falle von Abb. 13 geschieht dies durch Deaktivierung von Schalter „S3“ und anschließend (je nach Drehrichtung) der Aktivierung von „S2“ (im Uhrzeigersinn) oder „S3“ (gegen den Uhrzeigersinn).

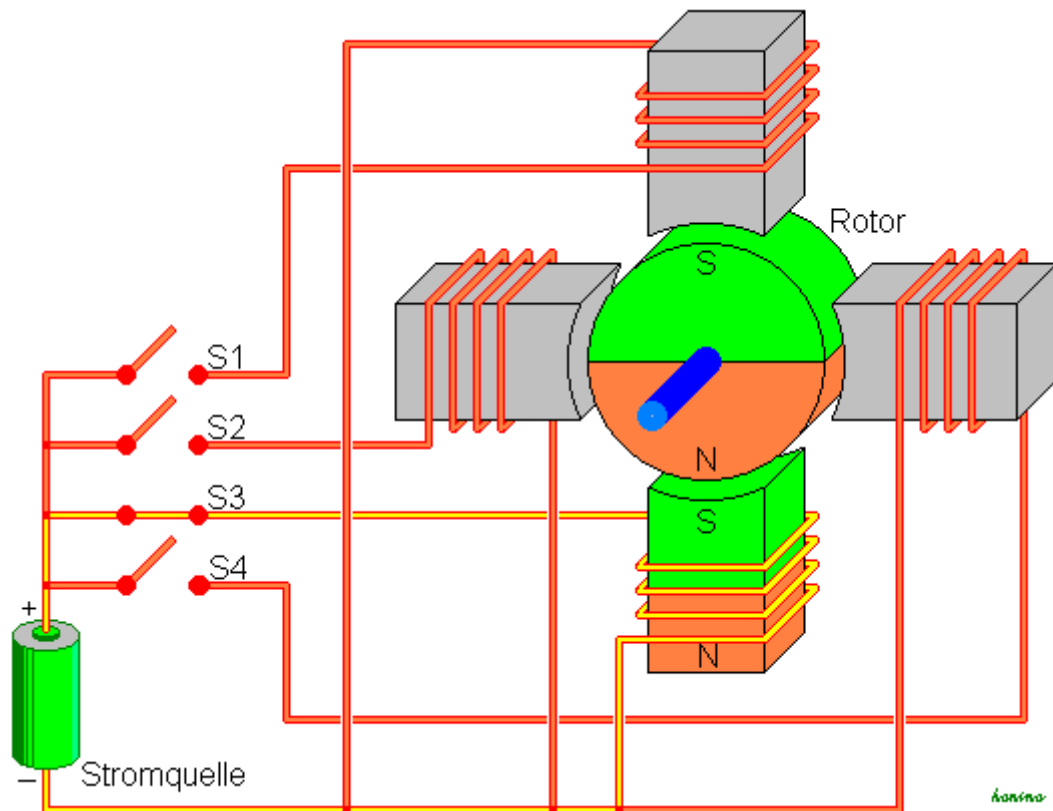


Abb. 13: Vereinfachte Darstellung des Funktionsprinzips eines unipolaren Schrittmotors (Horina, 2004)

Das Drehmoment auf den Rotor ist direkt proportional zur magnetischen Flussdichte. Für die durch die Spulen induzierte magnetische Flussdichte gilt allgemein folgende Formel (Zastrow, 2000):

$$B = \mu_0 * \frac{I * N}{l} = \mu_0 * I * n$$

Dabei ist I die Stromstärke, N die Windungszahl, l die Spulenlänge, $n = \frac{N}{l}$ die Windungsdichte und μ_0 die magnetische Feldkonstante. Um das maximale Drehmoment an einem Schrittmotor zu maximieren, müssen Strom und Windungszahl hoch, sowie die Spulenlänge gering sein. Fertigungs- sowie geometriebedingt steigt die Spulenlänge mit der Windungszahl; der Strom wird durch die thermische Verlustleistung und die Fähigkeit des Stators bestimmt, die entstandene Wärme abzuleiten.

Die im Plotter verbauten Schrittmotoren basieren zwar auf demselben Funktionsprinzip, haben jedoch eine andere Beschaltung der Spulen, da es sich um bipolare Schrittmotoren handelt. Bei

diesem Typ haben die Spulen keinen gemeinsamen Pol, sondern verändern das Magnetfeld durch eine Umkehr der Stromflussrichtung.

Darüber hinaus unterscheiden sich die verbauten Motoren durch die Rotor- sowie Statorgeometrie von der vereinfachten Darstellung in Abb. 13. Während bei einem Motor nach der vereinfachten Darstellung bereits ein viermaliges Umpolen eine vollständige Umdrehung des Rotors zu Folge hätte, ist dies bei kommerziell erhältlichen Schrittmotoren i.d.R. nicht der Fall.

Stattdessen werden die Polpaare im Rotor sowie die Spulenpaare im Stator vervielfacht und versetzt zueinander angeordnet, um

eine höhere Anzahl von Schritten pro Umdrehung zu erreichen und damit die Präzision zu erhöhen (vgl. Abb. 14). Übliche Polzahlen bei kommerziellen Motoren sind 96, 200 oder 400 Schritte pro Umdrehung.

Auf der Achse der Motoren befinden sich beim Plotter Zahnriemenscheiben mit 16 Zähnen für einen GT2-Riemen bei einem Zahnabstand von 2 mm. Bei einer vollständigen Umdrehung bewegt der Motor den Riemen daher $2 \text{ mm} * 16 = 32 \text{ mm}$ weit. Da eine vollständige Umdrehung aus 200 Einzelschritten besteht, bewegt sich der Werkzeugkopf $\frac{32 \text{ mm}}{200} = 0,16 \text{ mm}$ pro Schritt des Motors. Für den Anwendungsfall ist das bereits ausreichend, obwohl die Auflösung beim Plotter durch Interpolation von Zwischenschritten erhöht wird (Microstepping).

Die Aufteilung einer Umdrehung in kleinere Einzelschritte bringt zwar den Vorteil der Präzision, jedoch führt sie auch zu einer, sich im Projektverlauf besonders komplex gestaltenden, Herausforderung.

Die hohe Anzahl an Schritten (und damit auch an Umpolvorgängen der Spulen) pro Umdrehung führt dazu, dass der Schrittmotor bei hohen Drehzahlen zunehmend an Drehmoment verliert. Dies ist der Fall, da sich der Strom an den Spulen träge verhält und dem schnellen Umpolen entgegenwirkt. Abb. 15 zeigt dieses Verhalten bei einem Motor gleicher Bauart.

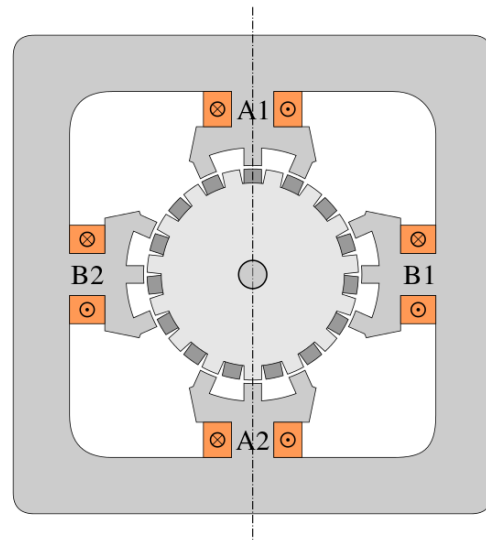


Abb. 14: Schema der Geometrie von Rotor und Stator in Schrittmotoren (Stündle, 2010)

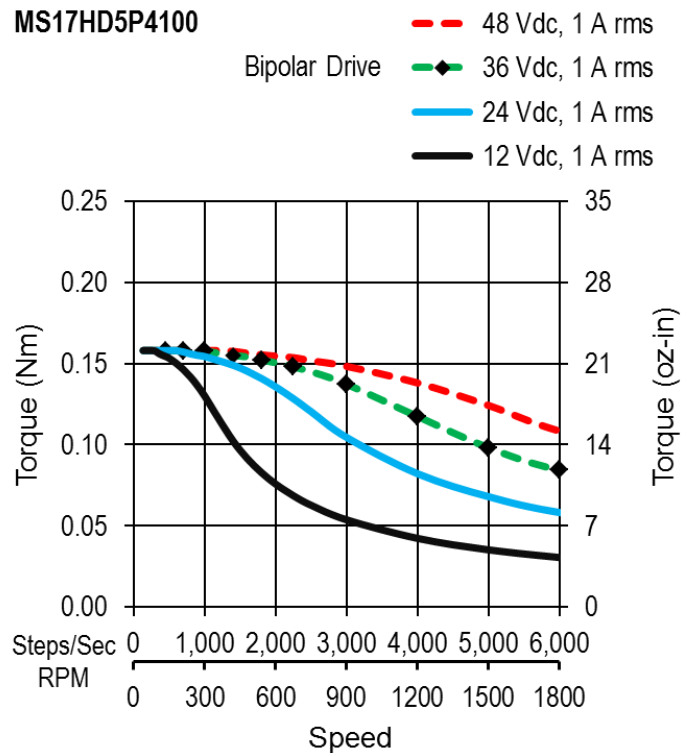


Abb. 15: Drehmomentverlauf eines vergleichbaren Motortyps im Verhältnis zur Drehzahl. Das Moment schwindet bei steigender Drehzahl. Dem kann mit einer Erhöhung der Spannung entgegengewirkt werden. (MOONS' Industries, 2019)

Abb. 15 illustriert zudem eine von zwei Möglichkeiten, diesem Problem entgegenzuwirken: Das Erhöhen der Betriebsspannung. Der Strom in einer Spule steigt nach einer e-Funktion an und erreicht nach einer Zeit von 5τ annähernd seinen Endwert I (Zastrow, 2000):

$$i_L = \frac{U}{R} (1 - e^{-\frac{t}{\tau}})$$

Demnach ist die Zeit bis zum Erreichen des Endwertes des Spulenstroms direkt proportional von der angelegten Spannung abhängig. Da der Plotter auf Messen zum Einsatz kommt, bei denen nicht ausgeschlossen werden kann, dass ein Besucher die Maschine berührt, wurde beschlossen, die Spannung auf max. 24 V zu begrenzen.

Durch die festgelegte Spannungsgrenze und die daraus resultierenden Drehmomentkurve begrenzt sich die Geschwindigkeit, die der Plotter bei der verwendeten Kombination aus Schrittmotoren und Treibern erreichen kann.

Wird diese Geschwindigkeit überschritten, kann der Rotor der Schrittmotoren sich nicht mehr schnell genug nach dem veränderten Magnetfeld ausrichten. Dies bezeichnet man als Schrittwverlust. Da der Steuerungssoftware kein Feedback bezüglich der tatsächlichen Position der Motoren gegeben wird, werden die Motoren weiter wie zuvor betrieben, selbst wenn dies bedeutet, dass eine Achse mit anderen Maschinenteilen kollidiert. Sollten bei einem Druckvorgang Schrittwverluste auftreten, ist das Ergebnis in der Regel unbrauchbar.

Die zweite Möglichkeit, auch bei zunehmender Drehzahl ein höheres Moment aufrecht zu halten, ist die Verwendung von Motoren mit niedriger Induktivität und damit einhergehend höherem Spulenstrom. Im Fall des Plotters ließen sich größere Ströme jedoch nicht mit den gewählten Treibern realisieren, da diese die Verlustleistung sonst nicht hätten abführen können.

Die gewählten Motoren sind auf einen Spulenstrom von 1,6 A ausgelegt. Diese Angabe bezeichnet den maximalen Stromfluss, bei dem die Motortemperatur 80° C nicht überschreitet. Kurzzeitig kann ein Motor auch mit größeren Strömen betrieben werden, solange die Temperatur niedrig genug bleibt, die Isolation des Spulendrahtes nicht zu beschädigen.

Schrittmotortreiber

Obwohl nach einem Schrittverlust in der Regel keine zuverlässige Positionsreferenz vorhanden ist, kann beim Plotter eine Gegenmaßnahme eingeleitet werden: Die verwendeten Schrittmotortreiber bieten die Möglichkeit, Schrittverluste zu erkennen. Dazu überwacht der Treiber den Spannungsverlauf nach einem Schritt. Je nach Lastwinkel verändert sich das Back-EMF des Motors, d.h. die elektromagnetische Rückkopplung der Spulen. Ist der Lastwinkel zu groß, erkennt der Treiber dies als Schrittverlust und signalisiert dies dem Mikrocontroller über eine steigende Flanke an einem Interrupt-fähigen Pin. Der Hersteller bezeichnet diese Funktion als „stallGuard2“.

Die integrierten Schaltkreise der Schrittmotortreiber werden von der Firma Trinamic Motion Control entwickelt und sind unter der Bezeichnung TMC2130 erhältlich. Da diese gerne im 3D-Druck-Bereich verwendet werden, sind fertige Treiberplatinen auf Basis des Chips und im „StepStick“ genannten Formfaktor verfügbar (vgl. Abb. 17).

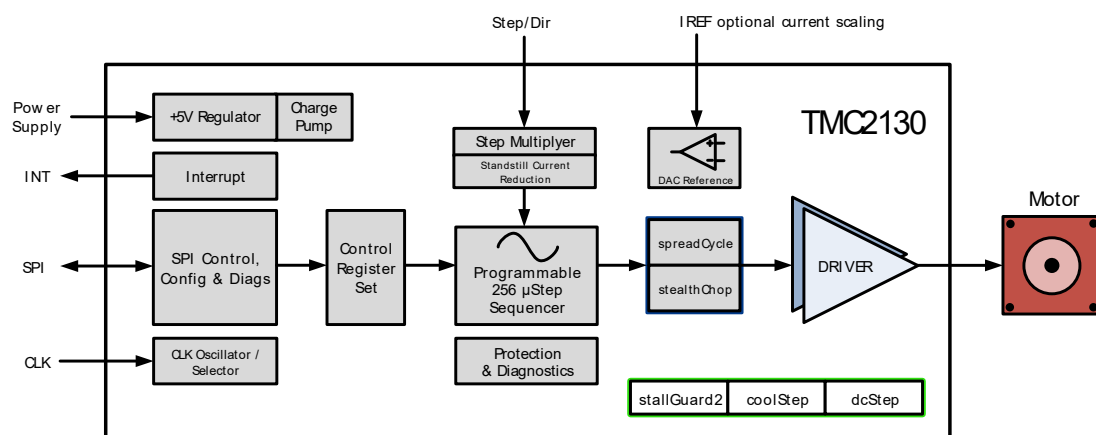


Abb. 16: Blockdiagramm des Trinamic TMC2130 Schrittmotortreibers. Er unterscheidet sich u.A. durch die API-Kommunikation (links) von herkömmlichen Treibern. (TRINAMIC Motion Control GmbH & Co. KG, 2018)

Bei günstigeren Schrittmotortreibern ist es üblich, die Kommunikation zum Mikrocontroller über lediglich 3 Signalleitungen abzuwickeln:

1. „Enable“:

Ist das Enable-Signal high, versorgt der Treiber den Motor mit Strom. Normalerweise ist

dieses Signal nur kurz nach dem Einschaltvorgang deaktiviert, da die Motoren sonst ihre Position nicht halten können.

2. „Step“:

Erkennt der Treiber an dieser Signalleitung eine steigende Taktflanke, so bewegt er den Schrittmotor einen Schritt weiter.

3. „Dir“:

Je nachdem, ob der Mikrocontroller diese Leitung gegen V_{CC} oder GND zieht, dreht der Treiber den Schrittmotor im- bzw. gegen den Uhrzeigersinn.

Die Treiber auf Basis des TMC2130 haben jedoch zusätzlich weitere Kommunikationsmöglichkeiten (vgl. Abb. 16). Diese beinhalten den bereits erwähnten Indikator für Schrittvverlust. Der dafür zuständige Pin wird mit „DIAG0“ bezeichnet.

Der größte Unterschied zu weit verbreiteten Controllern, etwa vom Typ „Allegro A4988“ und „Texas Instruments DRV8825“, liegt in der SPI-Schnittstelle des TMC2130. Über diese Schnittstelle lassen sich Einstellungen vornehmen, Informationen abrufen und auch Schrittbefehle erteilen.

Die zusätzlichen Funktionen der Treiber ermöglichen es, bei dem Plotter bestimmte Funktionen umzusetzen. So ist es durch die Erkennung von Schrittvverlusten möglich, den Plotter ohne den Einsatz von zusätzlichen Sensoren oder Endschaltern zu referenzieren („sensorless homing“), in dem die Achsen langsam so lange in eine Richtung bewegt werden, bis sie kollidieren. Die Motoren werden dann angehalten und die Positionsreferenz wird gespeichert. Es ist möglich, Schrittvverluste während des Drucks zu erkennen. Die Firmware ist leider aktuell nicht in der Lage, bei erkannten Schrittvverlusten erneut einen Homing-Vorgang durchzuführen.

Ebenfalls wird die SPI-Anbindung genutzt, um Steuerungsparameter wie den Motorstrom einzustellen. Laut Datenblatt ist der Treiber in der Lage, einen Strom von 1,2 A RMS bereitzustellen, jedoch war es mir nicht möglich, die Treiber bei dieser Last ausreichend zu kühlen. Die Ableitung der Wärme erfolgt durch die Platine, auf deren Unterseite der IC gelötet ist. Die unter dem IC platzierten Vias leiten die Wärme deutlich schneller ab, als es durch das Plastikgehäuse des Chips auf der Oberseite möglich wäre. So konnten die Treiber bis zu einem Dauerstrom von 800 mA RMS noch ausreichend gekühlt werden.

Ein ebenfalls hilfreiches Feature des TMC2130 ist die integrierte Temperaturüberwachung. Wird eine Temperatur von über 100° C gemessen, reduziert sich der Motorstrom automatisch um 50 mA und die Firmware wird über SPI davon in Kenntnis gesetzt.

Treiberplatine

Der erste, funktionsfähige Prototyp des Plotters verwendete eine kommerziell erhältliche Platine aus dem 3D-Druck. Das RAMPS¹² bietet eine Vielzahl an Steckplätzen für Bausteine wie Schrittmotortreiber, Heißeiter, Endschalter und Heizelemente. Die Quelldateien der Platine sind quelloffen und stehen unter der GNU-Lizenz¹³. Da die für die TMC2130 benötigten SPI-Busse nicht im Board vorgesehen sind, mussten sie oberhalb der Platine verkabelt werden. Dies, sowie die hohe Zahl an nicht benötigten Komponenten, führte zu einem unschönen Aussehen, was dem Einsatz als Messedemonstrator entgegensteht.

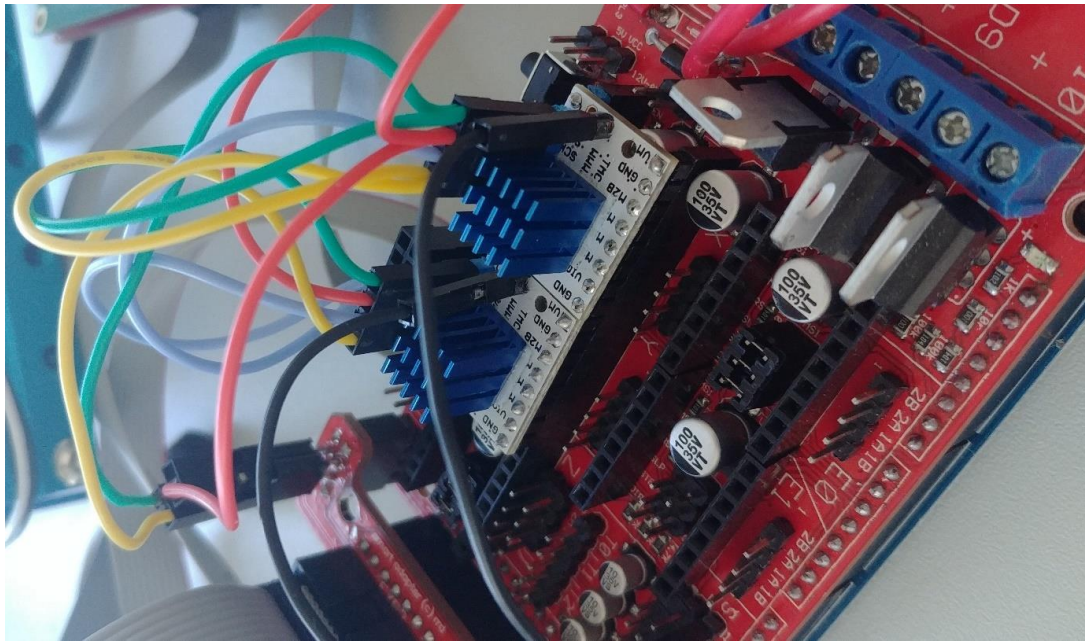


Abb. 17: Aufbau der Elektronik auf Basis des RAMPS

Zusammen mit meinem Betreuer habe ich mich daher entschlossen, eine neue Platine auf Basis des quelloffenen RAMPS zu entwickeln, die sich auf die benötigte Funktionalität beschränkt und deren Design ansprechender ist. Hierbei entstand das Mixed Mode Shield¹⁴.

¹² Reprap Arduino Mega Polulu Shield (Users, 2019)

¹³ Geplant ist, die Quelldateien des Mixed Mode Shields ebenfalls zu veröffentlichen

¹⁴ Der Schaltplan befindet sich im Anhang

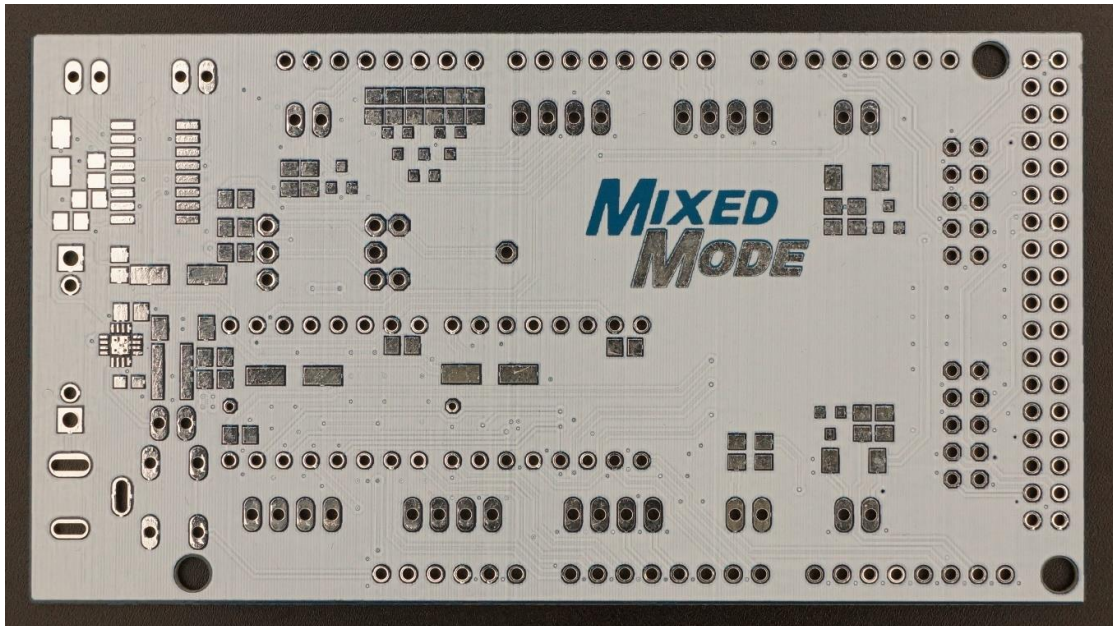


Abb. 18: Oberseite der Platine des Mixed Mode Shields

Während das RAMPS versucht, möglichst viele Anwendungsfälle bei unterschiedlichsten 3D-Druckern zu erfüllen, ist die Funktionalität des Mixed Mode Shields isoliert auf den Betrieb des Mes-
sedemonstrators.

Mikrocontroller

Wie auch das RAMPS basiert das Mixed Mode Shield auf einem Arduino Mega. Der Mikrocontroller dieser Entwicklerplatine ist der 8-Bit-Prozessor ATmega2560 (Atmel). Er findet sich mit Abstand in den meisten 3D-Drucker-Controllern und ist der Referenzcontroller der Marlin-Firmware, die auch den Plotter steuert¹⁵.

Spannungsversorgung

Das originale RAMPS v1.4 ist auf eine Versorgungsspannung von 12 V ausgelegt. Für das Mixed Mode Shield ist eine Spannung von 24 V vorgesehen. Die höhere Spannung bietet zwei wesentliche Vorteile:

1. Die Schrittmotoren entwickeln ein größeres Moment, da die Treiber den eingestellten Spulenstrom schneller aufbauen können. Dies resultiert sowohl in einer höheren maximalen Beschleunigung als auch einer größeren Maximalgeschwindigkeit. (Atwell, 2017)
2. Der zum Anheben der Kugelschreibermine verwendete Elektromagnet entwickelt eine größere Kraft und kann den Stift schneller anheben. Damit einhergehend erlaubt dies den Einsatz einer steiferen Rückstellfeder. Beides in Kombination reduziert die Stell-/Rückstellzeit der Mine und damit auch die Druckzeit.

¹⁵ Mehr über Marlin im Kapitel „Marlin-Firmware“

Die höhere Versorgungsspannung setzt jedoch mehrere Modifikationen voraus: Zunächst wird die Platine nun über einen Hohlstecker mit dem Netzteil verbunden. Im Gegensatz zu der sonst üblichen Klemmlösung lässt sich das verwendete Netzteil so schnell lösen und eine Verpolung ausschließen.

Während das RAMPS 1.4 die Versorgungsspannung direkt mit den V_{in} -Pin des Arduino verbindet, wurde beschlossen, die vom Arduino benötigten 5 V mit einem Schaltregler auf dem Mixed Mode

Shield zu generieren. Die verwendeten Komponenten auf 5 V-Basis benötigen einen Strom von bis zu 400 mA. Der üblicherweise verwendete Linearregler müsste dabei eine Verlustleistung von $P_{Verlust} = (24V - 5V) * 400mA = 7,6W$ abführen. Selbst bei einem Regler mit großem Formfaktor und hoher Wärmeablei-

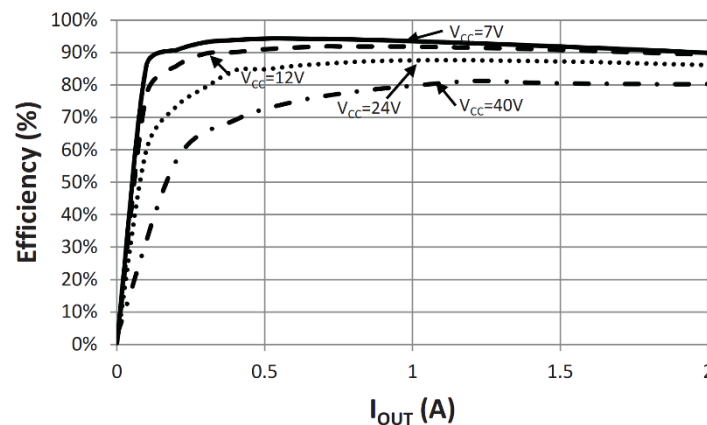


Abb. 19: Typische Performance von DCDC-Wandlern auf Basis des TS30042. Bei 14 V Eingangsspannung liegt die Effizienz schon ab 300 mA deutlich über 80%. (Semtech Corporation, 2019)

tung resultiert dies in einer hohen Betriebstemperatur. Der verwendete Tiefsetzsteller auf Basis eines Semtech TS30042 bietet in der verwendeten Beschaltung eine Effizienz von bis zu 85% (Semtech Corporation, 2019).

SPI Bus

Die von den Motortreibern benötigten Busleitungen sind als Leiterbahn in das Mixed Mode Shield integriert. Dies verbessert das Erscheinungsbild maßgeblich, verglichen mit der externen Verkabelung auf dem RAMPS. Neben den SPI-Leitungen sind zudem die DIAG-Pins über die Platine angeschlossen. (Atwell, 2017)

Die SPI-Anbindung der Motortreiber erlaubt der Firmware in Echtzeit Parameter der Treiber, wie den Spulenstrom, zu ändern. Über den DIAG-Pin signalisiert der Treiber erkannte Schrittverluste.

Display-Verbinder

Das verwendete LC-Display wird, zusammen mit dem Kartenleser, über zwei 10-Pin-Wannenstecker angeschlossen. Über die Wannenstecker wird das pixelbasierte Display und der in das Display integrierte SD-Kartenleser über den SPI-Bus verbunden. Um das Display auf dem RAMPS nutzen zu können ist ein spezieller Displayadapter erforderlich, der die Pins der AUX-1- und AUX-4-Verbinder auf die Pins der Wannenstecker verbindet.

Leistungstransistoren

Auf dem RAMPS v1.4 sind 3 MOSFET-Transistoren der Bauform TO—220 verlötet. Auf dem Mixed Mode Shield wurden diese gegen die wesentlich kleineren FDN 337N mit der Bauform SOT-23 getauscht. Trotz der kleinen Bauform ist der zulässige I_D mit bis zu 2,2 A ausreichend, um die Ströme von LEDs und Solenoid zu schalten, die sich auf bis zu 300mA beschränken.

Audioverstärker

Um die Lautstärke während des Betriebs weiter anheben zu können, werden ein Audioverstärker und zwei Lautsprecher verbaut, die sich unterhalb der Treiberplatine befinden. Hierbei wird das Signal auf der STEP-Leitung beider Motortreiber über die beiden Kanäle eines Audioverstärker vom Typ PAM8403 verstärkt und auf Lautsprechern wiedergegeben. Der Verstärker liefert bei Treibern mit einer Impedanz von $2\ \Omega$ eine Leistung von 3W. Die daraus resultierende maximale Lautstärke ist so hoch, dass der Aufenthalt neben der Maschine unangenehm ist. Die Lautstärke lässt sich mit einem Potentiometer pro Kanal stufenlos regeln, weshalb sie auf ein angenehmes Maß reduziert werden kann, so dass die Geräuschkulisse dennoch die Aufmerksamkeit der Messebesucher auf sich zieht.

Teile aus Eigenproduktion

Auch wenn es sich bei einem Teil der Komponenten um standardisierte Artikel handelt, mussten einige Teile speziell für den Plotter gefertigt werden. Um diese Teile deutlich erkennbar zu machen, sind sie auf allen Abbildungen in orangener Farbe dargestellt.

Ausnahmslos wurden diese Teile mit dem FFF-Verfahren¹⁶ auf einem 3D-Drucker des Typs „Ulti-maker 2“ hergestellt. Selbst für Teile mit vergleichsweise hohen Toleranzanforderungen ist die Präzision der Drucker ausreichend und das optische Erscheinungsbild ansprechend.

Als Material kam PETG¹⁷-Filament zum Einsatz. Verglichen mit dem weit verbreiteten Material PLA besitzt PETG eine deutlich höhere Zugfestigkeit von bis zu 50 MPa, leidet jedoch nicht unter den starken thermischen Deformationen, die den Druckvorgang bei ABS erschweren.

Minenhalter

Der in der Vorgängerarbeit verwendete Stiftplotter der Firma Markerblock nutzt – wie viele andere Stiftplotter im Hobby-Bereich – einen Servomotor aus dem RC-Modellbau, um den Stift vom Papier anzuheben bzw. abzusenken. Um jedoch die Geschwindigkeit der Vertikalbewegung zu steigern, ist bei der neu entwickelten Maschine ein Elektromagnet verbaut, der den Werkzeughalter vom Papier anhebt, sobald er von einem ausreichend hohen Strom durchflossen wird. Der Magnet ist in einer Zylinderspule (auch Solenoid genannt) verbaut und bereits mit einem Aktuator für lineare Bewegungen versehen (vgl. Abb. 20). Um den Stift abzusenken schaltet der Mikrocontroller den Magneten ab und er fällt zurück auf das Papier. Unterstützt wird die Schwerkraft dabei durch eine Druckfeder, die den Anforderungen entsprechend gegen eine andere mit passender Federkonstante ausgetauscht werden kann. Zu einer Beschädigung der eingespannten Kugelschreibermine durch die Kollision bei der Abwärtsbewegung ist es während der Entwicklung nicht gekommen. Das Gewicht der fallenden Teile ist vergleichsweise gering und der Abstand zum Papier beträgt lediglich 5 mm. Vor dem Absenken des Werkzeughalters pausiert die Firmware für 40 ms, damit die Mine auf dem Papier aufsitzt, bevor die Zeichenbewegung durchgeführt wird. Ebenso wird nach dem Anhebevorgang pausiert, jedoch reicht hier eine Pause von 10 ms. Um die

¹⁶ Fused Filament Fabrication

¹⁷ Polyethylenterephthalat Glycolmodifiziert

Wartezeiten weiter zu verkürzen ist es denkbar, eine Feder mit stärkerer Federkonstante zu verwenden. Das wurde allerdings im Rahmen dieser Arbeit nicht versucht.

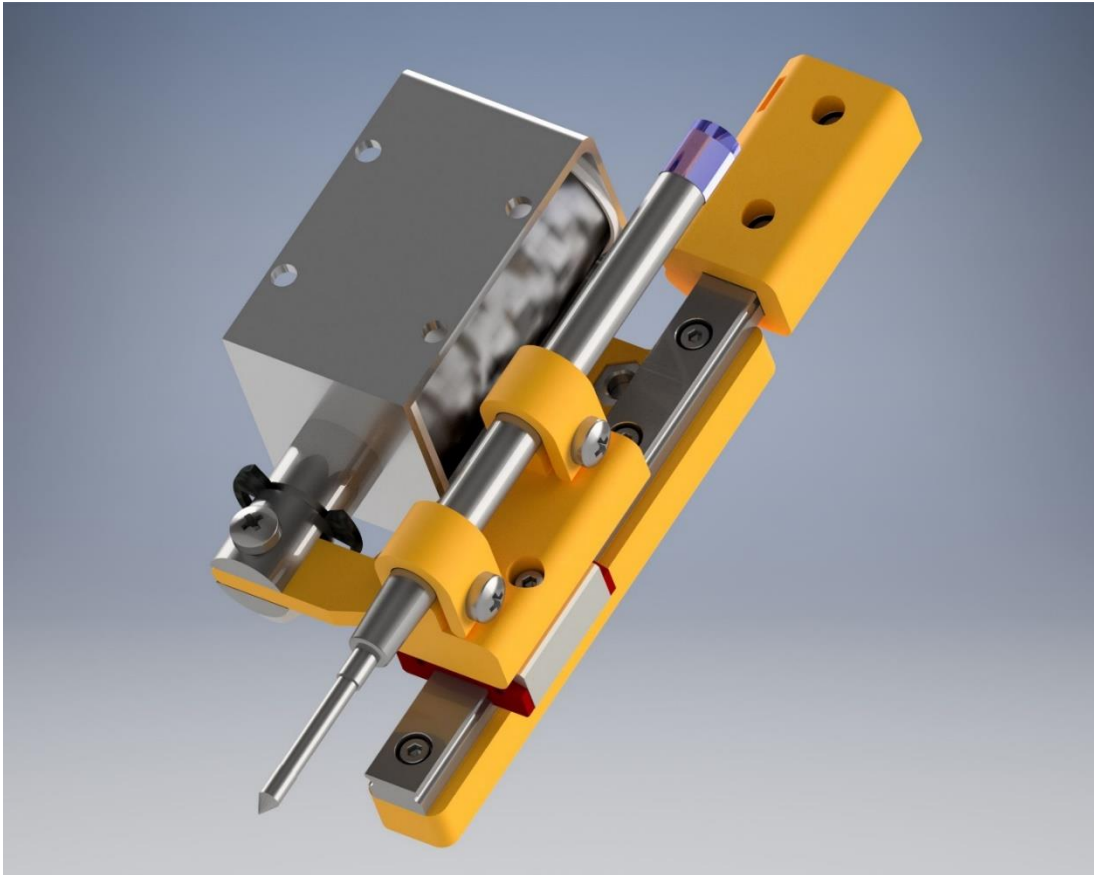


Abb. 20: Für die Vertikalbewegung zuständiger Mechanismus mit eingespannter Kugelschreibermine

Um die Vertikalbewegung möglichst ohne Abweichung von der Z-Achse durchzuführen, ist der Werkzeughalter an einer Profilschienenführung gelagert. Die Profilschiene der Z-Achse ist mit 7 mm Breite deutlich kleiner und leichter als die der X- und Y-Achse.

Display und SD-Karte

Mit der Hauptplatine des Stiftplotters ist ein Display mit integriertem Drehimpulsgeber und einem Kartenleser integriert. Dabei handelt es sich um einen kommerziell erhältlichen Controller für 3D-Drucker, der quelloffene RepRap Full Graphic Smart Controller (dancer, 2013). Von der SD-Karte können über das Display, unabhängig von einem Hostcomputer, bereits vorbereitete G-Code-Dateien ausgeführt werden. Dies können vorbereitete Portraits sein, ebenso wie mit anderen Tools generierte, generische Demo-Zeichnungen wie etwa das Mixed Mode Logo (Abb. 25).

Grundplatte

Die Grundplatte des Plotters besteht aus Aluminium-Dibond. Dies ist ein Verbundmaterial, bei dem zwei 0,3 mm dicke Aluminiumschichten mit einem Polyethylenkern thermisch verbunden werden. Die Platte ist leicht, stabil und optisch ansprechend.

Der Rahmen des Plotters registriert mit einer Aussparung an jeder Ecke in Gumminippel, die auf der Grundplatte verschraubt sind. Diese stammen ebenfalls aus dem 3D-Drucker und dämpfen die Schwingungen, durch die Bewegung etwas ab.

Auf der Dibondplatte befindet sich ein Klemmrahmen im Din A4-Format, in den das Zeichenpapier eingespannt wird.

Marlin-Firmware



Abb. 21: Ein Schwertfisch (Marlin) ist Wappentier der Marlin-Firmware (Brázio, 2019)

Die weltweit meist verbreitete Firmware auf 3D-Druckern ist die Marlin Firmware. Basierend auf den Firmwares „Sprinter“ und „Grbl“ wurde Marlin am 12. August 2011 unter der GP-Lizenz auf Github veröffentlicht. (Brázio, 2019)

Marlin bietet eine Fülle von Features für 3D-Drucker, die teilweise auch im Plotter Verwendung finden. Neben den essenziellen Funktionen – dem G-Code Interpreter und der Ansteuerung der Motortreiber – nutzt der Mixed Mode Plotter auch andere funktionale sowie kosmetische Features, wie beispielsweise die grafische Benutzeroberfläche oder die Wiedergabe von G-Code-Dateien von einer SD-Karte.

Einige Funktionen (etwa die für die Extrusion von Plastik verantwortlichen Module) sind für den Einsatz im Plotter irrelevant, bzw. wurden deaktiviert. Besonders Schutzfunktionen, wie der „Thermal Runaway“¹⁸ müssen deaktiviert werden, da keine Heizelemente vorhanden sind.

Im Plotter wird Marlin in der Version 2.0 verwendet. Diese Version trägt zwar das Präfix „bugfix“, läuft jedoch stabil und unterstützt Features, die in anderen Versionen nicht vorhanden sind (Kap. „Umgebung“).

¹⁸ Die automatische Abschaltung aller Heizelemente, sollte der gemessene Temperaturverlauf nicht mit dem zu erwartenden übereinstimmen

Umgebung

Die Referenzplattform von Marlin ist ein Mikroprozessor der Firma Atmel, der ATmega2560. Dieser Chip wird auch auf der Platine des Arduino Mega verwendet, der im Plotter zum Einsatz kommt. Während frühere Marlin Versionen in der Arduino-IDE entwickelt wurden, lässt sich Marlin 2.0 auch mit anderen Tools kompilieren. Bei der Firmware für den Plotter kam dafür die freie, quelloffene Software PlatformIO zum Einsatz. Sie beinhaltet Unterstützung für die Plattformen verschiedener Mikrocontroller, Software-Bibliotheken, Compiler-, sowie Debugwerkzeuge.

Im Vergleich zu der sehr beliebten Arduino IDE bietet PlatformIO diverse Vorteile:

- Hohe Compilegeschwindigkeiten dank multithreaded Compiling. Innerhalb der Arduino IDE dauerte das Kompilieren bis zu 30 Minuten, in PlatformIO oft weniger als 30 Sekunden.
- Volle Integration in die ebenfalls quelloffene Entwicklungsumgebung „Visual Studio Code“. Da die Software der Pfadgeneration ebenfalls in Visual Studio Code entwickelt wurde, konnte für beides dieselbe Umgebung verwendet werden.
- Unterstützung von vielen weiteren Mikrocontroller-Architekturen. Sollte zukünftig eine CPU auf 32-Bit Architektur verwendet werden, ist der Aufwand der Neukonfiguration der Entwicklungsumgebung minimal.

Evaluation der Geschwindigkeit des Plotters

Um die Zeichengeschwindigkeit vergleichend bewerten zu können, sind zwei Angaben erforderlich: die maximale Verfahrgeschwindigkeit [mm/s], sowie die maximale Beschleunigung [mm/s²].

Der Plotter erreicht bei einem Spulenstrom von max. 680 mA eine Höchstgeschwindigkeit von 300 mm/s bei einer Beschleunigung von 4000 mm/s².

Vergleichbar sind diese Größen mit denen eines 3D-Druckers, der ähnliche Technologie verwendet. Die Drucker vom Typ „Ultimaker 2“ (Ultimaker) sind bekannt für ihre hohen Geschwindigkeiten und erreichen eine Geschwindigkeit von ebenfalls 300 mm/s, jedoch bei einer etwas geringeren Beschleunigung von 3000 mm/s² (Ultimaker BV, 2018).

Ausblick

Mixed Mode Shield v2.0

Obwohl das im Plotter verbaute Mixed Mode Shield die benötigte Funktionalität besitzt, habe ich zusammen mit meinem Betreuer beschlossen, eine zweite Version der Platine zu entwickeln und fertigen zu lassen, die jedoch nicht vor Ende dieser Arbeit fertiggestellt werden konnte.

Die zweite Version enthält folgende Verbesserungen:

- Einen Tiefpass-Filter vor den Eingängen der Audioverstärker, der die Amplitude unangenehm hoher Frequenzen reduziert und somit die ausgegebenen Geräusche angenehmer klingen lässt.
- Eine modulare Montage des Tiefsetzstellers. Im Falle eines Fehlers kann dieser so einfacher ausgetauscht werden.
- LED-Schaltindikatoren neben den MOSFET-gesteuerten Ausgängen. Diese lassen mit bloßem Auge erkennen, ob ein Ausgang mit Spannung versorgt wird.
- Reduktion der Ausgänge für ein schlankeres Layout.
- Neupositionierung von Baugruppen. Einige der Elektrolytkondensatoren sind so leichter positionierbar.

Eignung des Plotter-Systems für die Verwendung als Messedemonstrator

Zum aktuellen Zeitpunkt hat der Stiftplotter bereits zwei Abnahmetests in der Praxis hinter sich und sich auf zwei Messen als Demonstrator bewährt. Der Erfolg der Entwicklung wurde hier durch das Interesse der Messebesucher bestärkt und durch die Erfüllung der Ziele bestätigt.

Der Plotter ist in der Lage, in unter fünf Minuten ein Portrait von einem Messebesucher zu zeichnen. Im Praxiseinsatz hat sich gezeigt, dass die Möglichkeit zur Variation zwischen Qualität und Zeichendauer von großem Nutzen ist. Auch kurze Demo-Zeichnungen wie das Mixed Mode-Logo ziehen große Aufmerksamkeit auf die Maschine und sind besonders schnell abgeschlossen. Auch wenn die maximale Geschwindigkeit mit 300 mm/s unter dem erwarteten Wert von 400 mm/s liegt, ist dies für den Alltagsgebrauch ausreichend. Größere Geschwindigkeiten setzten einen größeren Motortreiber voraus, was im Widerspruch zu den geringen Abmessungen des Plotters steht.

Der Wiedererkennungswert der gezeichneten Portraits ist hervorragend. Besonders zeigte sich dies auf der firmeninternen Messe „InHouse“. Hier erkannten die Besucher ihre Kollegen ausnahmslos auf den angefertigten Zeichnungen, die zur Demonstration neben der Maschine auslagen.

Der eingebaute Audioverstärker war auf den Messen aufgrund einer plötzlich notwendigen Reparatur nicht im Einsatz, konnte jedoch seine Leistung unter Beweis stellen. Selbst ohne aktive Geräuschquelle waren die Motoren in der Lage, einen ausreichenden Geräuschpegel zu erzeugen, um noch außerhalb der Sichtweite des Standes hörbar zu sein.

Besonderes Lob bekam die Benutzerfreundlichkeit durch die Mitarbeiter der Personalabteilung, die den Plotter eingesetzt haben. Das mitgelieferte Tool funktioniert ohne Probleme und auch die Bedienung ohne verbundenen PC über das integrierte Display verlief reibungslos. Erwähnenswert ist zudem die Wartungsfreundlichkeit. Sie zeigt sich zum einen darin, dass die verwendete Kugelschreibermine seit der Montage

nicht ausgewechselt wurde, zum anderen in der Zuverlässigkeit der Profilschienenführungen, deren Laufruhe und Präzision nach wie vor unverändert hoch ist.

Durch das Sensorlose Homing reduziert sich die Anzahl von Bauteilen, die für den Aufbau der Maschine benötigt werden, bzw. mechanischer Abnutzung ausgesetzt sind. Allgemein ist der Stiftplotter sehr einfach nachzubauen und mit Gesamtkosten von unter 250 € zudem sehr günstig.

Literaturverzeichnis

- Atmel. (31. 03 2019). *Microchip.com*. Von Datasheet atmega2560: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf abgerufen
- Atwell, C. (17. Februar 2017). *MachineDesign.com*. Von Misconceptions about Stepper Motors Explained: <https://www.machinedesign.com/motorsdrives/misconceptions-about-stepper-motors-explained> abgerufen
- Brázio, J. (19. Februar 2019). *MarlinFW*. Von About: <http://marlinfw.org/docs/basics/introduction.html> abgerufen
- Carey, D. (14. 4 2019). *Drive Trains*. Von RepRap.org: <https://reprap.org/wiki/Category:DriveTrains> abgerufen
- dancer, R. (13. 4 2013). *RepRapDiscount Full Graphic Smart Controller*. Von RapRap.org: https://reprap.org/wiki/RepRapDiscount_Full_Graphic_Smart_Controller abgerufen
- DeepArt UG. (13. 4 2019). *Create Your Own*. Von DeepArt.io: <https://deepart.io/hire/> abgerufen
- Gutin, G., & Punnen, A. P. (2007). *The Traveling Salesman Problem and Its Variations*. 5: 18.
- Hans B. Kief, H. A. (2015). *CNC-Handbuch*. München: HANSER.
- Horina. (12. 1 2004). *File:Schrittmotor.PNG: Wikimedia Commons*. Von Wikimedia Commons: <https://commons.wikimedia.org/wiki/File:Schrittmotor.PNG> abgerufen
- Kast, A. M. (6. Juli 2017). Umsetzung und Optimierung eines als Messedemonstrator konzipierten Selfie-Plotter-Systems. München, Bayern, Deutschland.
- Leon A. Gatys, A. S. (26 . 8 2015). A Neural Algorithm of Artistic Style. Tübingen, Baden-Württemberg, Deutschland.
- Libert, B. (14. Februar 2019). *Dev#007 CoreXY Motion System*. Von Laser Beaver: <https://www.laserbeaver.com/dev007-corexy-motion-system/2/> abgerufen
- Makeblock Co., Ltd. (16. 12 2018). *XY Plotter Robot Kit*. Von makeblock.com: <http://download.makeblock.com/XY/pdf/XY%20Plotter%20V2.0%20Assembly%20instructions%EF%BC%882016.6%EF%BC%89.pdf> abgerufen
- MOONS' Industries. (6. 4 2019). *NEMA 17 Standard Hybrid Stepper Motors*. Von MOONS' Industries: <https://www.moonsindustries.com/series/nema-17-standard-hybrid-stepper-motors-b020105> abgerufen

Oskay, W. (22. 8 2017). *StippleGen*. Von Evil Mad Scientis Wiki:
<https://wiki.evilmadscientist.com/StippleGen> abgerufen

Semtech Corporation. (30. Januar 2019). *Semtech*. Von TS30042 1A/2A Current-Mode Synchronous Buck Converter: <https://www.semtech.com/products/power-management/femtobuck-dc-dc-regulators-controllers/ts30042> abgerufen

Stündle. (20. 9 2010). *File:Hybrid-schrittmotor-v2.svg*: *Wikimedia Commons*. Von Wikimedia Commons: <https://commons.wikimedia.org/wiki/File:Hybrid-schrittmotor-v2.svg> abgerufen

TRINAMIC Motion Control GmbH & Co. KG. (14. 12 2018). *TMC2130: Integrated Circuits*. Von trinamic.com:
https://www.trinamic.com/fileadmin/assets/Products/ICs_Documents/TMC2130_datasheet.pdf abgerufen

Ultimaker BV. (20. 6 2018). *Ultimaker/Ultimaker2Marlin*. Von GitHub:
<https://github.com/Ultimaker/Ultimaker2Marlin> abgerufen

Users, F. (28. 1 2019). *RAMPS 1.4*. Von RepRap.org: https://reprap.org/wiki/RAMPS_1.4 abgerufen

Zastrow, D. (2000). *Elektrotechnik - Ein Grundlagenbuch*. Braunschweig / Wiesbaden: Vieweg.

Erklärung zur Abschlussarbeit

Hiermit versichere ich, die eingereichte Abschlussarbeit selbständig verfasst und keine andere als die von mir angegebenen Quellen und Hilfsmittel benutzt zu haben. Wörtlich oder inhaltlich verwendete Quellen wurden entsprechend den anerkannten Regeln wissenschaftlichen Arbeitens zitiert. Ich erkläre weiterhin, dass die vorliegende Arbeit noch nicht anderweitig als Abschlussarbeit eingereicht wurde.

Das Merkblatt zum Täuschungsverbot im Prüfungsverfahren der Hochschule Augsburg habe ich gelesen und zur Kenntnis genommen. Ich versichere, dass die von mir abgegebene Arbeit keinerlei Plagiate, Texte oder Bilder umfasst, die durch von mir beauftragte Dritte erstellt wurden.

A handwritten signature in black ink, appearing to read 'Albrecht', with a stylized, cursive script.

Tillmann Albrecht,

Markt, den 18.04.2019

Anhang

Beispiel einer G-Code Datei

Der Inhalt folgender G-Code Datei zeichnet das Mixed Mode Logo. Das Ergebnis ist in Abb. 25 dargestellt. Zeichendauer: 24 Sekunden. Gekürzte Darstellung. Originallänge: 316 Zeilen (8 KB)

```
M150 R255 U255 B255
M106
G4 P50
G28 X0 Y0
G00 X 68.6962 Y 91.0252 F 12000
```

```
M107
G4 P50
G01 X 56.0257 Y 43.4417
G01 X 64.6001 Y 43.4417
G01 X 72.8738 Y 74.7350
G01 X 73.0230 Y 43.4417
G01 X 81.8441 Y 43.4417
G01 X 98.4730 Y 74.6527
G01 X 90.2802 Y 43.4417
G01 X 98.8534 Y 43.4417
G01 X 98.8857 Y 43.5616
G01 X 111.5269 Y 91.0252
G01 X 97.8256 Y 91.0252
G01 X 81.6842 Y 59.3031
G01 X 82.3448 Y 91.0252
G01 X 68.6962 Y 91.0252
G4 P50
```

[...]

```
M106
G00 X 33.5051 Y 133.1197
```

```
M107
G4 P50
G01 X 46.6092 Y 133.1197
G01 X 45.9378 Y 101.0594
G01 X 46.2259 Y 100.9759
G01 X 62.5853 Y 133.1197
G01 X 75.6978 Y 133.1197
G01 X 63.2040 Y 86.1989
G01 X 55.2639 Y 86.1989
G01 X 63.6843 Y 118.2748
G01 X 63.4040 Y 118.4024
G01 X 46.2451 Y 86.1989
G01 X 37.9061 Y 86.1989
G01 X 37.7533 Y 118.3208
G01 X 37.4527 Y 118.3655
G01 X 28.9472 Y 86.1989
G01 X 21.0112 Y 86.1989
G01 X 33.5051 Y 133.1197
M106
```

G1 X207 Y187
M150 R0 U255 B0

Detaillierte Zeichnungen



Abb. 22: Angefertigte Zeichnung des schraffierten Portraits aus Abb. 5. Benötigte Zeit: 8:45 Minuten.

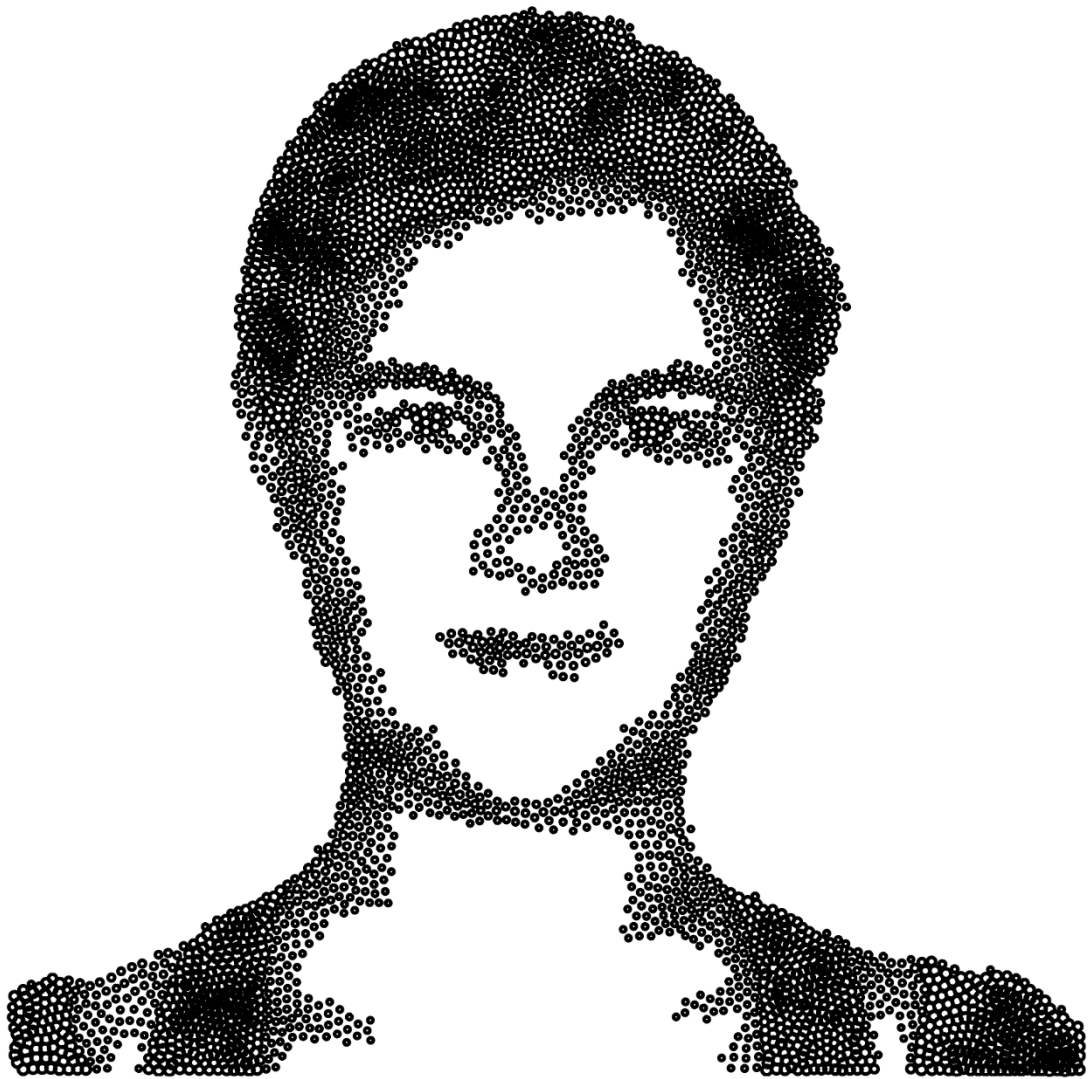


Abb. 23: Durch Kreise unterschiedlicher Radien gewichtete Punktzeichnung

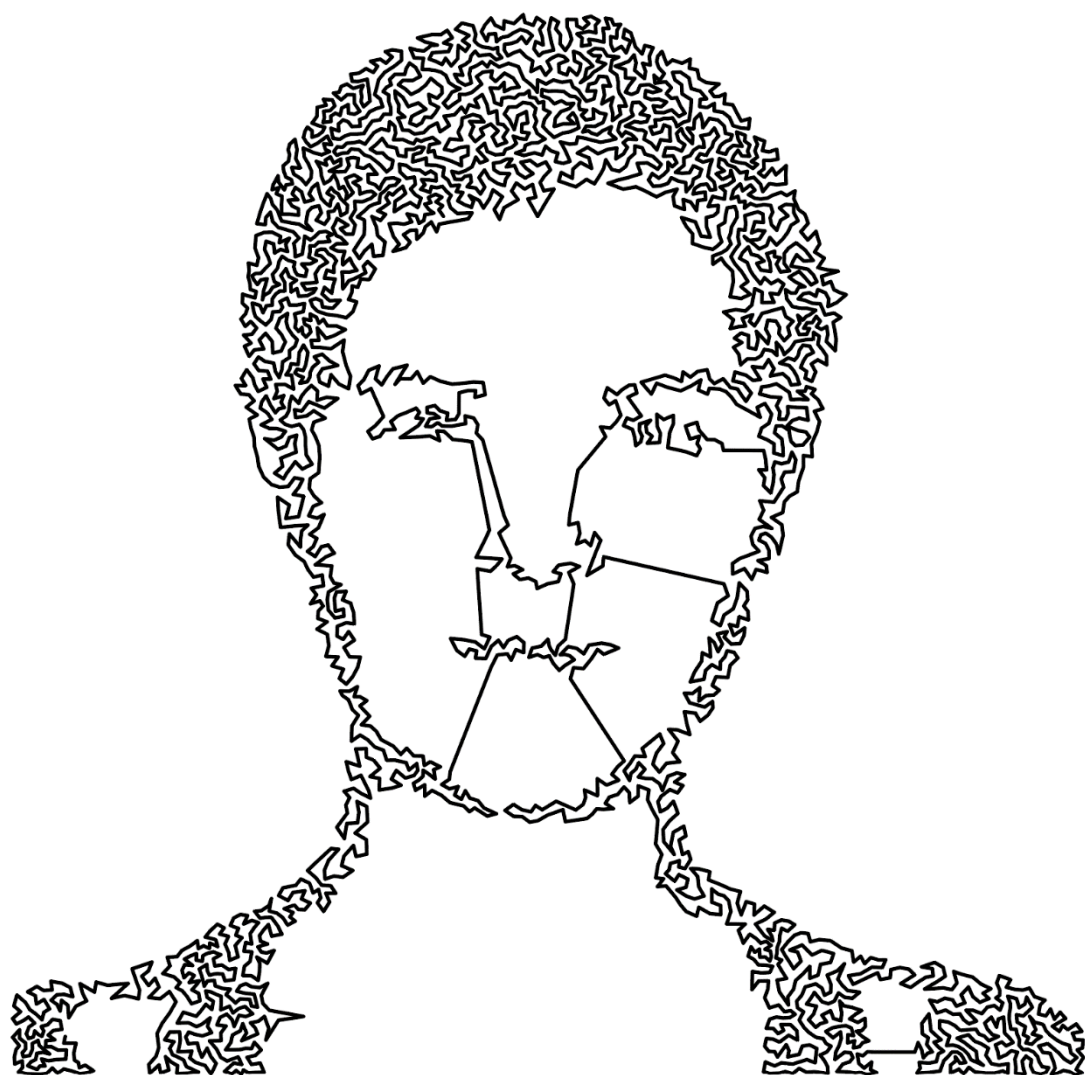


Abb. 24: Zeichenpfad basierend auf "TSP-Art" mit 10 000 Punkten

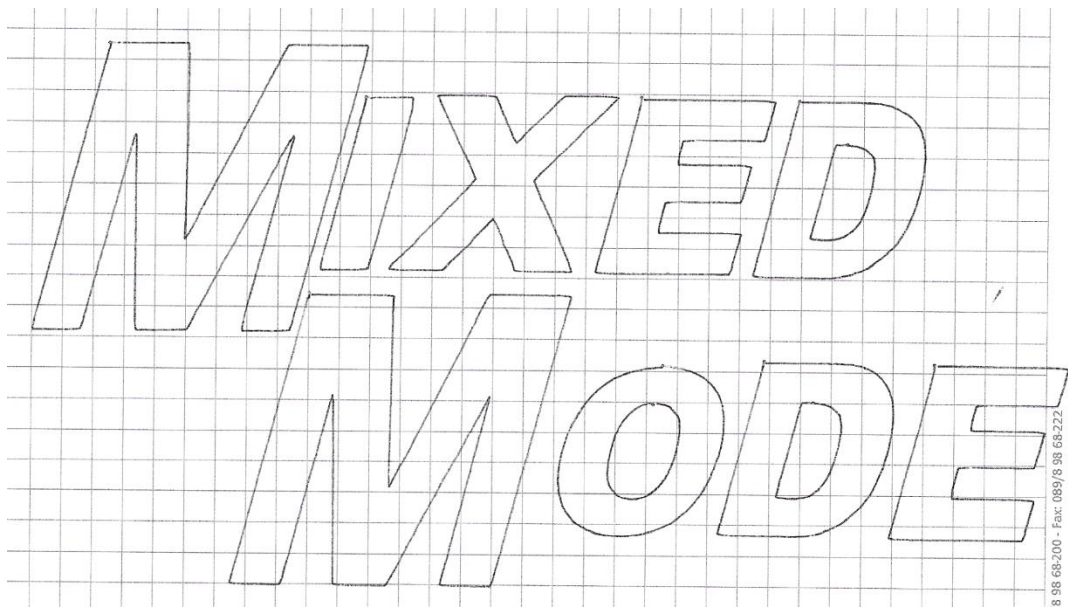


Abb. 25: Zeichnung des Mixed Mode Logos. Zeichendauer: 24 s

Dateistruktur auf dem beiliegenden Datenträger

- /
 - **py**

Dieses Verzeichnis enthält alle Quelldateien der PlotCon-Software. Gestartet wird diese durch Ausführung von „plot_con.py“

 - **requirements**

Enthält „pip_requirements.txt“. Hier sind sämtliche benötigten Python-Module angegeben
 - **demo**

Enthält Beispiel-G-Code-Dateien

 - **portraits**

Die hier enthaltenen Beispiele zeichnen Portraits
 - **svg**

Diese Beispiele sind aus Vektorgrafiken generiert
 - **cad**

Enthält 3D-Objektdateien

 - **stl**

Sämtliche benötigten Druckteile
 - **source**

Dateien der CAD-Software „Autodesk Inventor“
 - **img**

Hier befinden sich einige Bilddateien, welche im Laufe dieser Arbeit aufgenommen wurden
 - **doc**

In diesem Verzeichnis befinden sich (soweit vorhanden) Datenblätter der verwendeten Komponenten.
 - **marlin**

Dieses Verzeichnis enthält den Quellcode der Firmware des Stiftplotters