



DEGREE PROJECT IN MECHANICAL ENGINEERING,  
FIRST CYCLE, 15 CREDITS  
*STOCKHOLM, SWEDEN 2021*

# Multicolor 3D printing

An analysis of the characteristics of multicolor 3D printing

**DAVRON PATKHULLAEV**

**YONAS MELAKE**



**KTH ROYAL INSTITUTE OF TECHNOLOGY  
SCHOOL OF INDUSTRIAL ENGINEERING AND MANAGEMENT**





# **Multicolor 3D printing**

*Thesis submitted in fulfillment of the  
requirements for the degree*

*of*

**The Bachelor of Science degree in Mechatronics Engineering**

*by*

**Davron Patkhullaev | Yonas Melake**

Under the supervision of

**Nihad Subasic**

**SCHOOL OF INDUSTRIAL ENGINEERING AND MANAGEMENT**

**KTH ROYAL INSTITUTE OF TECHNOLOGY**

**TRITA-ITM-EX 2021:23**

# ABSTRACT

3D printing has risen in popularity due to its capabilities which enable rapid prototyping and printing structures that were impossible to produce before and partially because of technological development in recent years. Most low-cost printers, based on the fused filament fabrication process, are capable of printing a single filament at a time that limits and complicates multicolor and multi-material 3D printing. Therefore, investigating and building a multicolor 3D printer have been chosen as a degree project in order to provide an understanding for multicolor 3D printing.

The project was started with an investigation of different possibilities and potential solutions. After that, a visual CAD model was created in Solid Edge, and a virtual model was constructed by using Acumen. Lastly, accumulated ideas were implemented into creating a prototype.

Construction of the prototype was accomplished successfully, and several experiments were executed to determine its capabilities and limitations. The results showed that the printer has high precision in XY-plane and is less precise in Z-axis. The printer was able to print 45 different colored objects, of which only 24 were chosen as distinct colors that the human eye could distinguish. One of the main challenges with a single nozzle multicolor 3D printing compared to a single-color printing is constant heatsink clogging. Another challenge is to create a well-integrated system where hardware and software comply perfectly in a manner that desired prints are obtained.

**Keywords :** Mechatronics, 3D printer, Arduino Mega, Diamond Hotend, Marlin.

# REFERAT

3D skrivare har ökat i popularitet på grund av dess kapacitet som möjliggör att skapa prototyper snabbt och att skriva ut strukturer som var omöjliga att producera tidigare samt delvis på grund av den tekniska utvecklingen under senare år. De flesta billiga skrivare, baserade på smält filamenttillverkningsprocess, kan skriva ut ett enda filament i taget som begränsar och komplicerar att skriva ut flerfärgat objekt. Därför har man valt att undersöka och bygga en flerfärg 3D skrivare som examensarbete för att ge en förståelse kring ämnet.

Projektet börjades med en undersökning av olika möjligheter och potentiella lösningar. Därefter skapades en visuell CAD-modell i Solid Edge och en virtuell modell konstruerades med hjälp av Acumen. Slutligen, implementerades ackumulerade idéer för att skapa en prototyp.

Konstruktionen av prototypen genomfördes framgångsrikt och flera experiment utfördes på den för att ta reda på dess kapacitet och begränsningar. Resultaten visade att skrivaren har hög precision i XY-plan och är mindre exakt i Z-led. Skrivaren kunde skriva ut 45 olika färgade föremål, varav endast 24 valdes som distinkta färger som kunde särskiljas av det mänskliga ögat. En av de största utmaningarna med ett flerfärg 3D skrivare med ett enda munstycke jämfört med enfärg skrivare är konstant propp i kylfläns. En annan utmaning är att skapa ett välintegrerat system där hårdvara och mjukvara går hand i hand så att önskade utskrifter erhålls.

**Nyckelord:** Mekanik, 3D printer, Arduino Mega, Diamond Hotend, Marlin.

## ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to our honorable teachers of the School of Industrial Engineering and Management at the KTH Royal Institute of Technology who taught us during our study. We are very thankful to have our parents for their tremendous support, unconditional love, and encouragement during tough periods.

We would like to say thanks to our friends who motivated us during the whole project by building their own robots and contributing to this project with their ideas and experience. We would like to thank our supervisor *Nihad Subasic* for providing guidance throughout this work. Finally, this project would not be possible without *Lasse Hässler* who provided us with several crucial components, and *Jan Stamer* who helped us with welding and creating the frame.

*Davron Patkhullaev* and *Yonas Melake*  
**KTH Royal Institute of Technology**  
**Date:** May 24, 2021

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background . . . . .	1
1.2	Scope . . . . .	2
1.3	Purpose . . . . .	2
1.4	Method . . . . .	2
<b>2</b>	<b>Theory</b>	<b>3</b>
2.1	The FFF Process . . . . .	3
2.1.1	Extruder . . . . .	4
2.1.2	Nozzle . . . . .	4
2.1.3	Hotends: Multi-Material Hotends . . . . .	6
2.2	Position Control . . . . .	7
2.2.1	Stepper Motor . . . . .	7
2.2.2	Stepper motor drivers . . . . .	7
2.3	Temperature Control . . . . .	7
<b>3</b>	<b>Implementation</b>	<b>8</b>
3.1	Electronics . . . . .	8
3.1.1	Arduino Mega and RAMPS 1.4 . . . . .	8
3.1.2	Power supply . . . . .	10
3.1.3	Thermistor: NTC Thermistor . . . . .	10
3.2	Firmware . . . . .	11
3.2.1	G-code for Color Mixing . . . . .	11
3.2.2	The Start of G-code . . . . .	11
3.3	Design . . . . .	12
3.3.1	Frame . . . . .	13
3.3.2	Components for Cartesian Movement . . . . .	13
<b>4</b>	<b>Results</b>	<b>14</b>
4.1	Prototype . . . . .	14
4.2	Experiments . . . . .	17
<b>5</b>	<b>Discussion</b>	<b>19</b>
5.1	Color mixing . . . . .	19
5.2	Precision . . . . .	19
5.3	Bed . . . . .	19
5.4	Diamond Hotend . . . . .	20
5.5	Motion . . . . .	20

<b>6 Conclusion</b>	<b>21</b>
<b>Bibliography</b>	<b>22</b>
<b>A Appendix: CPS Acuman</b>	<b>A1</b>
<b>B Appendix: CAD Frame</b>	<b>B1</b>
<b>C Appendix: Changes in Marlin</b>	<b>C1</b>
<b>D Appendix: Components of Z-axis</b>	<b>D1</b>
<b>E Appendix: CAD Components of XY- plane</b>	<b>E1</b>
<b>F Appendix: 3D photos of samples</b>	<b>F1</b>



# List of Figures

2.1	Illustration of the basic hardware setup for FFF 3D printing . . . . .	3
2.2	Direct- and Bowden extruder . . . . .	4
2.3	Comparison of time taken between different nozzle diameters to extrude square parts . . . . .	5
2.4	Decreasing pressure drop as the nozzle diameter becomes larger . . . . .	6
2.5	Diamond Hotend . . . . .	6
3.1	RAMPS - Circuit connection . . . . .	8
3.2	Expander - Circuit connection . . . . .	9
3.3	Voltage distribution of thermistor on RAMPS . . . . .	10
3.4	CPS model- Created in Acumen . . . . .	12
4.1	Front . . . . .	14
4.2	Side . . . . .	15
4.3	Back . . . . .	15
4.4	Diamond Hotend, fans and 3D printed parts that attach the hotend to linear rods . . . . .	16
4.5	X-axis Assembly . . . . .	16
4.6	Extruder . . . . .	16
4.7	Blue to red spectrum . . . . .	18
4.8	Red to green spectrum . . . . .	18
4.9	Green to blue spectrum . . . . .	18
B.1	Frame 1 - Created in Solid Edge 2020 . . . . .	B1
B.2	Frame 2 - Created in Solid Edge 2020 . . . . .	B2
D.1	Final bed supporter - Created in Solid Edge 2020 . . . . .	D1
D.2	Motor and smooth rods holder - Created in Solid Edge 2020 . . . . .	D2
D.3	First Z-axis Assembly . . . . .	D3
D.4	Final Z-axis Assembly . . . . .	D4
E.1	Bearing holder 1 - Created in Solid Edge 2020 . . . . .	E1
E.2	Bearing holder 2 - Created in Solid Edge 2020 . . . . .	E1
E.3	Bearing holder 3 - Created in Solid Edge 2020 . . . . .	E1
E.4	Bearing holder 4 - Created in Solid Edge 2020 . . . . .	E1
E.5	Nozzle mount - Created in Solid Edge 2020 . . . . .	E2
E.6	Motor Holder - Created in Solid Edge 2020 . . . . .	E2
F.1	3D photos of samples from side . . . . .	F1

F.2 3D photos of samples from top . . . . . F1

# List of Tables

2.1	Material properties for common nozzle materials . . . . .	5
3.1	The driver connection to AUX-2 of RAMPS 1.4 through a board . . . . .	9
4.1	Statistical information of measurements on rectangular prints . . . . .	17
C.1	Marlin - Changed configurations . . . . .	C1

# Nomenclature

## Abbreviations

CAD	Computer Aided Design
CPS	Cyber-Physical Systems
FFF	Fused Filament Fabrication
G-code	Geometric Code
LCD	Liquid Crystal Display
NTC	Negative Temperature Coefficient
PLA	Polylactic Acid
RAM	Random Access Memory
RAMPS	RepRap Arduino Mega Pololu Shield
STL	Standard Tessellation Language
TIG	Tungsten Inert Gas Welding
UART	Universal Asynchronous Receiver/Transmitter

# 1. Introduction

## 1.1. Background

3D printing is an additive manufacturing process that makes a three-dimensional object by printing the object layer by layer. 3D printing is a fast-growing and emerging technology. It is used widely in different industries, especially in high-quality automotive, locomotive, and aviation industries [1]. Moreover, it is a common tool that CAD designers and 3D printer hobbyists utilize to print prototypes and 3D scanned objects.

Most of these consumer-grade desktop printers are low-cost fused filament fabrication (FFF) 3D printers characterized by the extrusion of a thermoplastic tube called filament through a nozzle heating with a thermistor package [2]. As a product of this process, most FFF printers can print a single filament at a time that limits and complicates multicolor and multi-material 3D printing. Therefore, the challenging project of building a multicolor 3D printer has been chosen for the bachelor's degree project in mechatronics and mechanical engineering. The decision was inspired by those who built the chocolate 3D printer as their degree project in mechatronics the previous year [3].

To print different colored objects simultaneously, multiple filaments must be used. That can be achieved by modifying the printer head, including nozzle, extruder, and hotend. The technical theory of color printing and the color printing nozzle suggests a shape that is like a diamond of three hotends coming together in one end towards a conical nozzle [4]. This solution has been applied earlier by a group of students of Worcester Polytechnic Institute who have tried to build multicolor 3D printers [5]. They have got significant results. However, they suggest controlling the entire project by a smaller group can give even better results for color mixing since color mixing requires hardware to comply with software accurately.

## 1.2. Scope

The project was engaging and raised multiple research questions, which might require several months to obtain more complex and extensive answers. However, the time allocated for this project was 16 weeks part-time. That's why only the two most relevant questions were chosen. More solutions, particularly for nozzle and hotend, could have been considered, but only the Diamond Hotend solution became the focus of this project because of time and budget limitations.

## 1.3. Purpose

The primary goal of this project was to produce a fully functional FFF 3D printer. Additionally, as part of a bachelor's thesis in mechatronics at the Royal Institute of Technology in Stockholm, provide an understanding for multicolor 3D printing by investigating and examining the following questions:

- How can a 3D printer be designed to print multicolored objects simultaneously?
- In what ways is it still a challenge to make FFF multicolor 3D printing?

## 1.4. Method

The project was divided into three main parts. The first phase was an investigation of different possibilities and potential solutions that have already been tested. The second phase was about creating a visual CAD model using Solid Edge and simulating a mathematical model of cyber-physical system (CPS) of the 3D printer by using Acumen [6]. Lastly, accumulated ideas were implemented into creating a prototype.

Experiments were executed to test the prototype and answer the research questions. First, it was necessary to print something simple to test how many colors the printer can produce, which necessitated printing many samples of the same shape. Moreover, it should be easy to measure the dimensions of these prints. That all led to print rectangles.

## 2. Theory

### 2.1. The FFF Process

The FFF process is a 3D printing process where a filament of thermoplastic material is extruded through a nozzle by an extruder drive gear via thermistor package, hotend. This process creates molten plastic, which cools rapidly and solidifies while the nozzle moves in Cartesian axes: X, Y, Z. The relative distance between the build plate and the top of the nozzle becomes the height of each layer and changes with a fixed increment with each layer creating perimeters and infill as a result. It is illustrated in Figure 2.1.

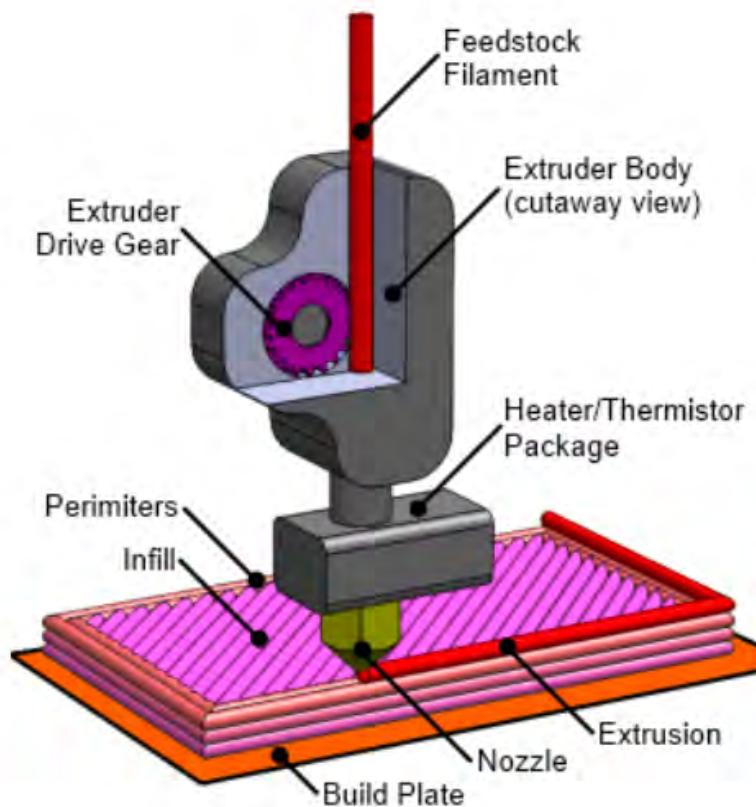


Figure 2.1: Illustration of the basic hardware setup for FFF 3D printing [2].

### 2.1.1. Extruder

The main function of the extruder is to feed the hotend, see section 2.1.3, with the right amount of filament, which is the build material. The two main types of extruders are direct and Bowden extruders [7], see Figure 2.2.

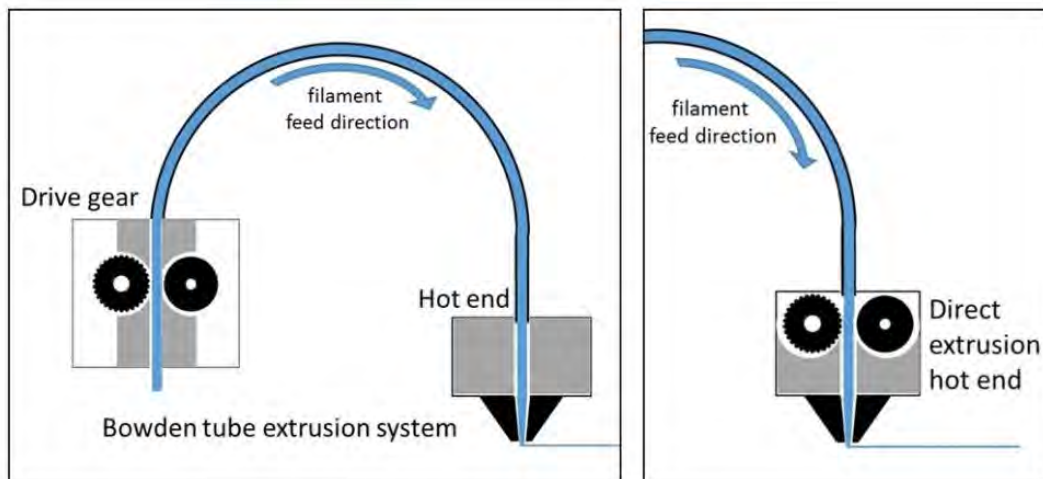


Figure 2.2: Direct- and Bowden extruder [8]

Direct extruders' name come from them being directly connected to the hotend. The short distance to the hotend leads to having better responsiveness to extrusion than Bowden extruders, which means cleaner prints - in other words, limited artifacts on the surface finish, such as stringing or oozing. In addition, direct extruders demand less torque in extrusion than Bowden extruders, so the size of the motor can be reduced or have a lower current setting. Generally, direct extruders can handle flexible materials better than Bowden extruders. However, the downside of direct extruders is that backlash, overshoot, and frame wobble can happen more often because they move along with the hotend, and it is more difficult to move more mass. In contrast, Bowden extruders are attached to the hotend with a Bowden tube that leads to faster movement [7].

### 2.1.2. Nozzle

The nozzle has a key role as it has the last contact with filament before it molds. It can be made of several different materials as long as the material has high thermal conductivity since the nozzle conducts thermal energy into the filament to melt it. Materials such as brass (copper-zinc alloy), stainless steel, and hardened steel are common mainly



because of their thermal properties but also cost.

Brass is inexpensive and has higher conductivity than stainless steel, and thus it is the most common one but is not as wear-resistant as steel, see Table 2.1.

Table 2.1: Material properties for common nozzle materials [9].

Material [SIS]	$\rho$ ( $kg/m^3$ )	$c_p$ ( $W/m \cdot ^\circ C$ )	Hardness ( $HB$ )
Brass [5122-02]	8540	120	<85
Stainless steel [2333-02]	7900	15	<180
Hardened steel [142534-5]	7810	42	<510

Another important parameter to consider is the hole size of the nozzle, i.e., inner diameter. Nozzle diameters generally range between  $0.1mm$  and  $1mm$ . The printer can achieve high precision with small nozzle diameters, but the print speed will decrease tremendously, see Figure 2.3. That's why the  $0.4mm$  nozzle is market standard because it gives a good balance between speed and precision. Moreover, a pressure drop is another physical aspect that affects the product quality. Consistent products can be produced by minimizing the pressure drop. A  $0.3mm$  nozzle diameter has been proposed to being the optimal diameter for extruding PLA material because of the decreasing pressure drop as the nozzle diameter becomes larger [10], see Figure 2.4.

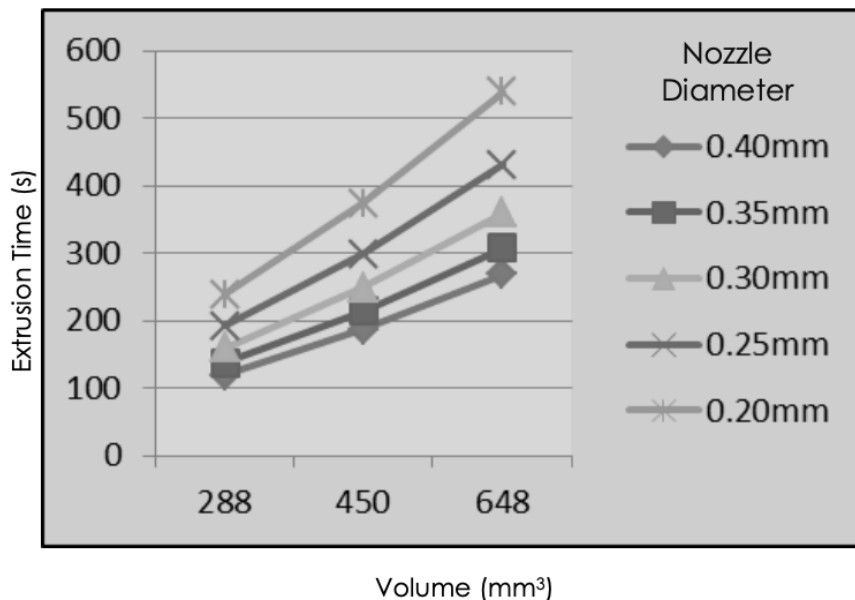


Figure 2.3: Comparison of time taken between different nozzle diameters to extrude square parts [10].

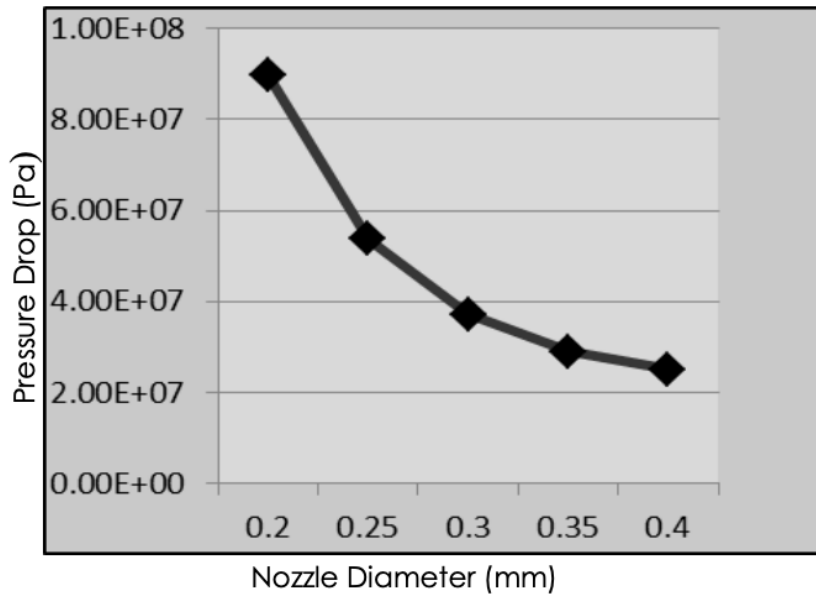


Figure 2.4: Decreasing pressure drop as the nozzle diameter becomes larger [10].

### 2.1.3. Hotends: Multi-Material Hotends

There are multiple types of multi-material hotends, but this report will focus on Diamond Hotends. Three different types of filament can be supplied to Diamond Hotends through their three feeds. Using this, it is possible to blend colors in almost unlimited ways. Because Diamond Hotends only use one nozzle, see Figure 2.5, the calibration is easy. One drawback is that it produces filament waste compared to Palette, another type of hotend, which produces almost none. But diamond hotends are much cheaper both to produce or to purchase [11].



Figure 2.5: Diamond Hotend [12]

## 2.2. Position Control

The Cartesian coordinate system can reasonably be implemented since the majority of desktop printers are Cartesian printers. The head of the 3D printer needs to move in the X- and Y-axis, while the print bed moves in the Z-axis. To get high precision print, the movement has to be smooth and linear in every direction.

### 2.2.1. Stepper Motor

A stepper motor rotates a certain interval for every pulse of power. A stepper driver gives the pulses of power, referred to as a step. Stepper motors do a full 360° rotation in a number of steps, usually in 200 steps [13].

### 2.2.2. Stepper motor drivers

Stepper motor drivers are used to give the motor higher power than the main microcontroller, such as Arduino, cannot. Additionally, the drivers' chips provide fractional steps, which makes the motion of the stepper motor smooth [14]. Up to three jumpers are used to determine the step size of a driver [15]. When all three are on, then the step size becomes 1/16 of a step for A4988 and DRV8825, drivers used in this project.

## 2.3. Temperature Control

The temperature of hotends and heated beds can be controlled by PID tuning. That's a proportional-integral-derivative control algorithm, see Equation 2.1.

$$u(t) = K_P * e(t) + K_I * \int_{t_0}^t e(\tau) d\tau + K_D * \frac{de(t)}{dt} \quad (2.1)$$

The error between the desired and current output is denoted  $e$ , the input to the system  $u$  and time  $t$  [16]. Open source firmware for 3D printers called Marlin became the choice of the project, see section 3.2. PID tuning is one of the features that is implemented in Marlin, and the parameters  $K_P$ ,  $K_I$ , and  $K_D$  or just P, I, and D can be given by G-code commands. For example, to auto-tune for a nozzle following can be used:

```
1 M303 E0 S200 C8
```

The code leads to the nozzle (E0) to heat up. Its temperature cycles around the target temperature 200°C (S200) 8 times (C8), and P, I, and D values will be returned.

# 3. Implementation

## 3.1. Electronics

There are several ways to control temperature and motion. Therefore, in the following sections, only electronics chosen for this project will be introduced briefly.

### 3.1.1. Arduino Mega and RAMPS 1.4

The Arduino MEGA 2560 is designed for projects that require many digital inputs and outputs, a lot of sketch memory, and RAM. It has 54 digital input/output pins, 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, [17] and can easily be connected to RAMPS. See the top of the RAMPS in Figure 3.1.

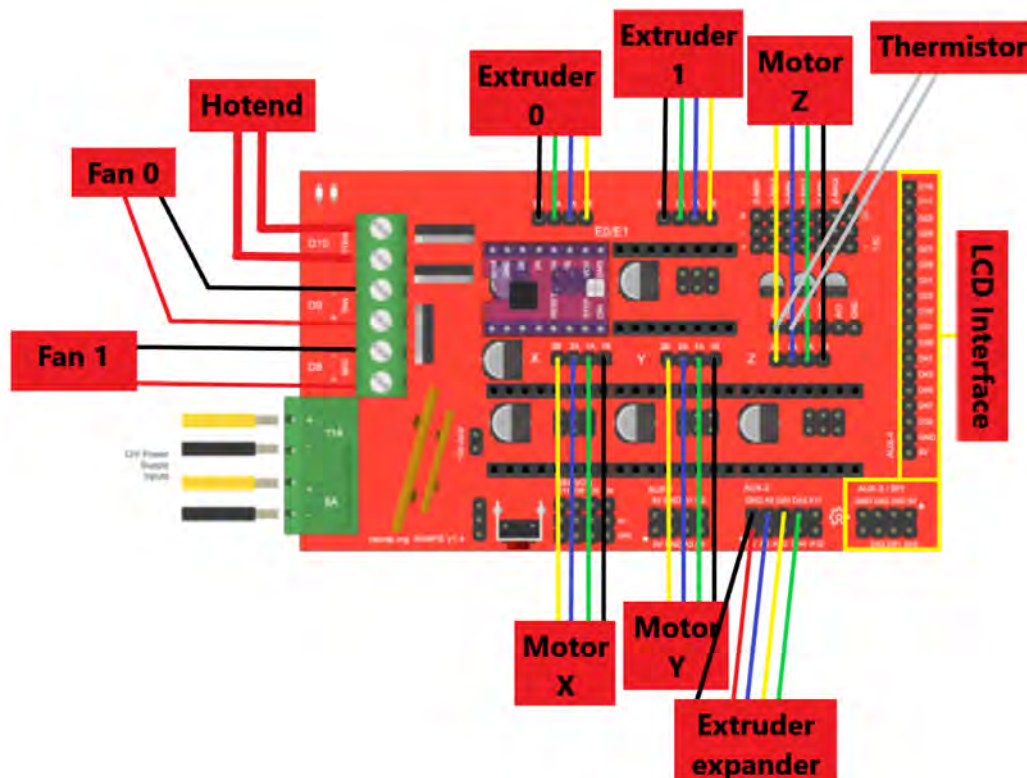


Figure 3.1: RAMPS - Circuit connection [Modified] [18].

Arduino Mega is chosen as a micro-controller because of its simplicity, effectiveness, and capabilities stated above. RepRap Arduino Mega Pololu Shield, RAMPS is a low-cost product but powerful electronics extension for Arduino Mega and includes plugins for several stepper drivers and extruder control electronics, see Figure 3.1 for all plugins and the circuit connection for this project. Only one stepper driver is shown in the figure to illustrate the point, but five more were used, including on the extruder expander. Moreover, version 1.4 has surface mount capacitors and resistors. RAMPS is the perfect choice for this project because of these features. For this project, D8 and D9 are connected to fans, and also AUX-2 is connected to extruder 2 by creating a circuit called extruder expander, see Figure 3.2 and Table 3.1.

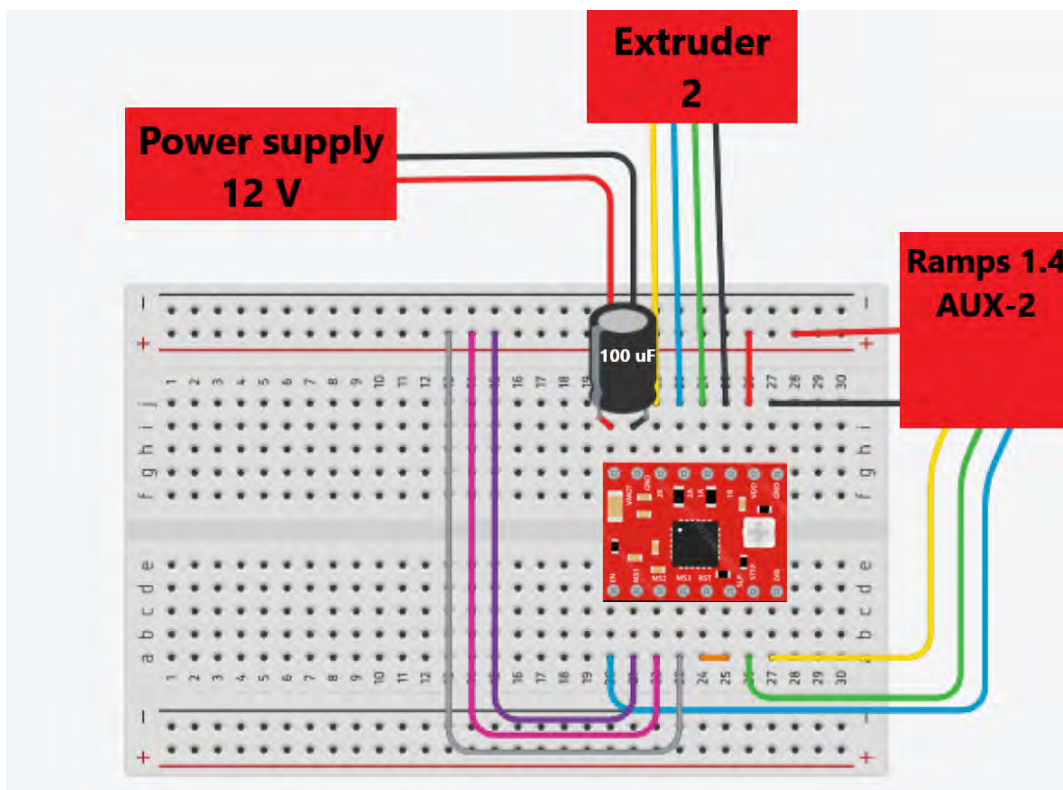


Figure 3.2: Expander - Circuit connection [Modified] - Created in Tinkercad

Table 3.1: The driver connection to AUX-2 of RAMPS 1.4 through a board

Driver Pin	5V	GND	ENABLE	DIR	Step	Ms1, Ms2, Ms1
AUX-2 Pin	5V	GND	A9[63 in Marlin]	D40[40]	D42[42]	5V

### 3.1.2. Power supply

To run this 3D printer, one of the power suppliers needed to give current up to 11A and 12V. To achieve that, chargeable motorcycle and car batteries were used. The other 5A power input to RAMPS and the power input for the extruder expander were connected to a separate wall socket which could only provide up to 4A and 12V. These inputs could have been connected to a battery, but to get the most life out of the battery, these inputs were connected to a wall socket with an unlimited supply.

### 3.1.3. Thermistor: NTC Thermistor

Thermistors are used to measure the temperature. This functionality comes from the fact that thermistors are built of sintered semiconductor material so that small changes of temperature lead to proportional large changes in resistance. For example, a negative temperature coefficient (NTC) thermistor is a thermistor whose resistance declines as the temperature rises [19]. Up to three thermistors can be utilized on RAMPS, see Figure 3.3.

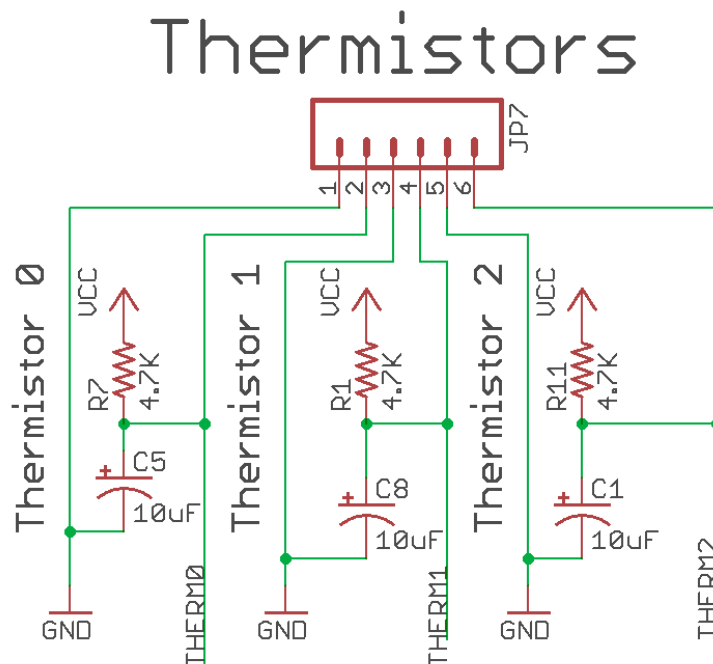


Figure 3.3: Voltage distribution of thermistor on RAMPS [20]

## 3.2. Firmware

Marlin is an open-source firmware for 3D printers and well-operating firmware for an Arduino Mega2560 with RAMPS 1.4. Marlin Firmware runs on the 3D printer's mainboard, managing all the real-time activities of the machine. It coordinates the heaters, steppers, sensors, lights, LCD, buttons, and everything else involved in the 3D printing process [21].

Marlin implements FFF. The control language for Marlin is a derivative of G-code. To print a CAD model with Marlin, the model should be saved as STL format, which must be later converted into G-code in any slicer program. Movement commands will be put into the movement queue to be executed in the received order. Linear movements get converted into precisely timed electronic pulses to the stepper motors. After downloading Marlin-2.0.X [22] and setting up the motherboard to BOARD\_RAMPS\_14\_EFF, changes given in the Table C.1 were applied.

### 3.2.1. G-code for Color Mixing

After the changes have been done on the Marlin code and the diamond nozzle was set up with three filaments of different colors. Then the amount of each filament extruded can be controlled by the code in Listing 3.1.

```
1 M163 S0 P0.5 ; Extruder 0 print at 50 percent
2 M163 S1 P0.2 ; Extruder 1 print at 20 percent
3 M163 S2 P0.3 ; Extruder 2 print at 30 percent
4 M164 S0 ; Set for default extrusion
```

Listing 3.1: Code for color mixing

### 3.2.2. The Start of G-code

The first action is to print from the constructed 3D printer to produce G-code from the Cura engine using the model's STL file as an input. In the Cura engine, the unit for Z-axis is in millimeters, while in the uploaded Marlin firmware, it is in centimeters. This must be in consideration when dimensioning and placing the model. The model should be placed between 0 – 25mm on the X-axis, between 0 – 120mm on the Y-axis, and between 0 – 5mm (becomes 0 – 5cm in reality) on the Z-axis. The profile of each layer must be set to 0.04mm (becomes 0.04cm in reality). This value was found with experimenting. The start and end of the G-code must be changed to the code in Listing 3.2 and 3.3.

```
1 M163 S0 P0.5 ; Extruder 0 print at 50 percent
2 M163 S1 P0.2 ; Extruder 1 print at 20 percent
3 M163 S2 P0.3 ; Extruder 2 print at 30 percent
4 M164 S0 ; Set for default extrusion
5
6 G92 X0 Y0 Z0 E0 ; Set current position for X, Y, Z and E(extrusion) to 0
```

Listing 3.2: Start G-code

```
1 G1 X0 Y0 ; Go to position X = 0 and Y = 0
```

Listing 3.3: End G-code

### 3.3. Design

The printer's design was influenced by the virtual model of the 3D printer created by using Acumen [6], see the whole code in Appendix A. It gives an overview picture of the 3D printer design. The model has a blue bed, two different colored rods, and a green head that all moved together and printed the green triangle, see Figure 3.4.

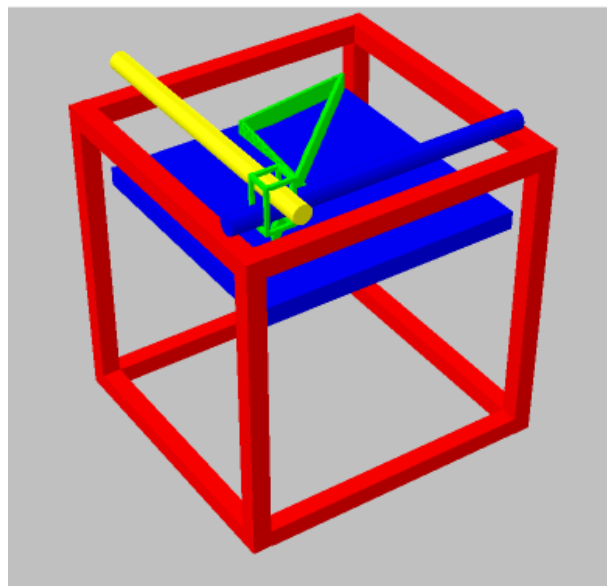


Figure 3.4: CPS model created in Acumen [6]



### **3.3.1. Frame**

The frame was planned to be 3D printed, and two CAD models were created shown in Appendix B. The frames are very similar because the frame in Figure B.1 might have been too big to print, but the other frame would have been printed in components.

However, the frame was built of scrap metal found in the KTH production facility because of the tremendous material and time consumption that is necessary for printing. The scrap metal was cut to desired sizes and then welded together to form the shape shown in Figure 4.1 and 4.2. The welding process that was used is tungsten inert gas welding (TIG).

### **3.3.2. Components for Cartesian Movement**

Two smooth rods, a motor with an attached threaded rod, and a brass nut were found in KTH from previous projects. The first solution involved an aluminum plate that was screwed to the brass nut and stayed fixed in the XY-plane with the two rods. The friction between the aluminum plate and the rods was too high, so 3D printed components were used as a plain bearing between them. The metal bed was joined to the aluminum plate with screws and springs. The springs were added for calibration - making the bed horizontally parallel to the XY-plane. The bed was made of metal so that it could be heated with the nozzle, and the heated bed is better for printing. See the first solution in Figure D.3.

The first solution failed to move smoothly, partly because the weight was too much for the motor and also the aluminum plate tilted. For the final solution, the aluminum plate was replaced by 3D printed plate to make it less heavy, and a U-formed weight was added to prevent tilting. To hold the smooth rods and the motor fixed to the frame, a CAD model was developed. See the final solution in Figure D.4.

Most of the components for the XY-plane of the prototype were found in KTH from older projects. The developed components were four bearing holders, a component to fix the motor for X-axis movement to the frame, and a component to hold the nozzle. These components were 3D printed, see the developed components in Appendix E.

## 4. Results

### 4.1. Prototype

The prototype frame is made from welded rectangular steel profiles. It has four holes in which two metal rods go through, and these rods serve multiple purposes. They are part of the frame and hold perpendicular rods that are needed for Y-axis. They even hold the filament in place without locking rotation, see Figure 4.1.

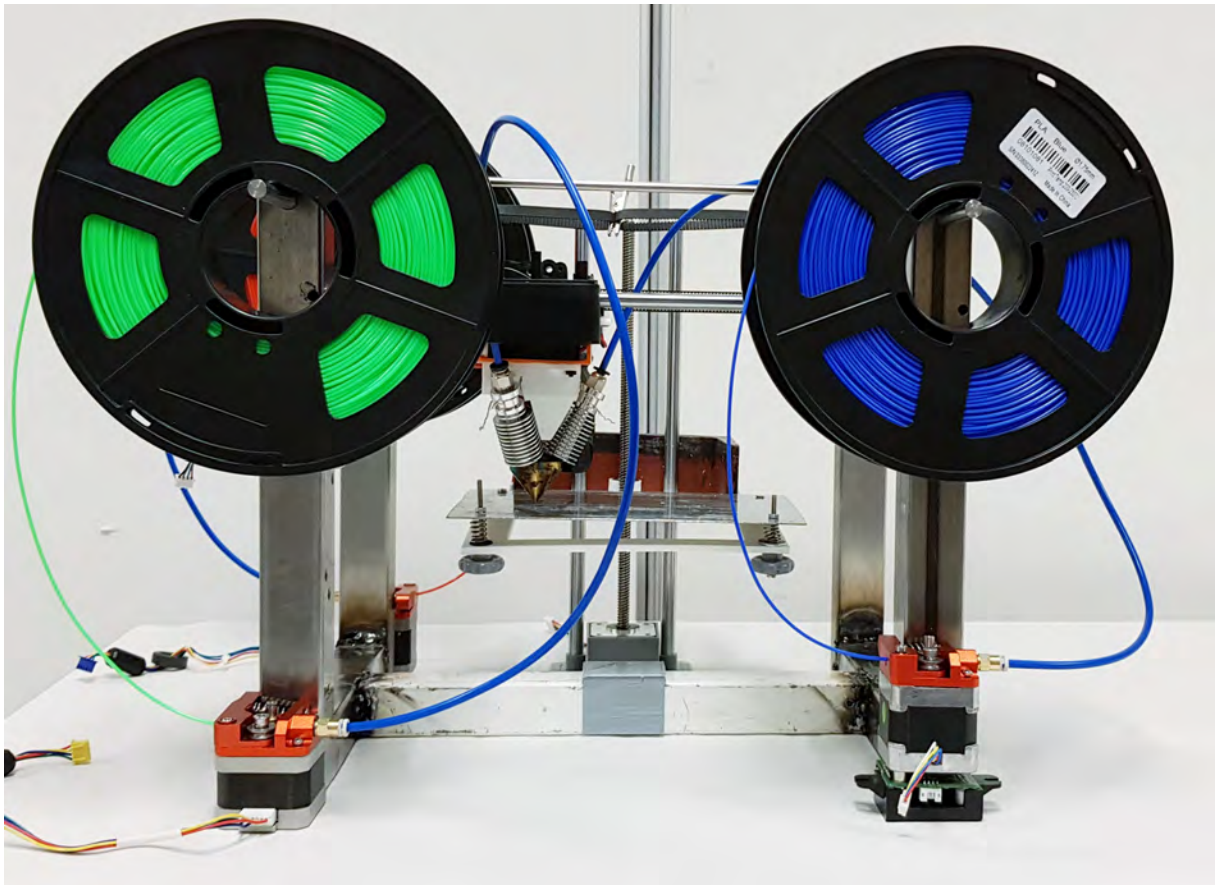


Figure 4.1: Front

Y-axis parts, black plastic that attaches four rods with bearings were given, and the prototype was built with respect to the size of bearings. A timing belt was used to turn the rotational motion of step motors into linear, and four 3D printed parts were used to keep timing belts apart, see Figure 4.2 and Figure 4.3.

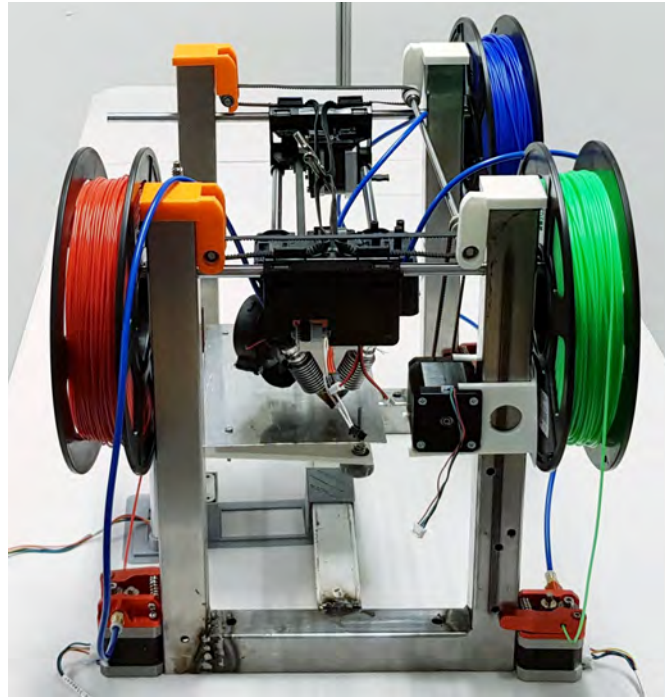


Figure 4.2: Side

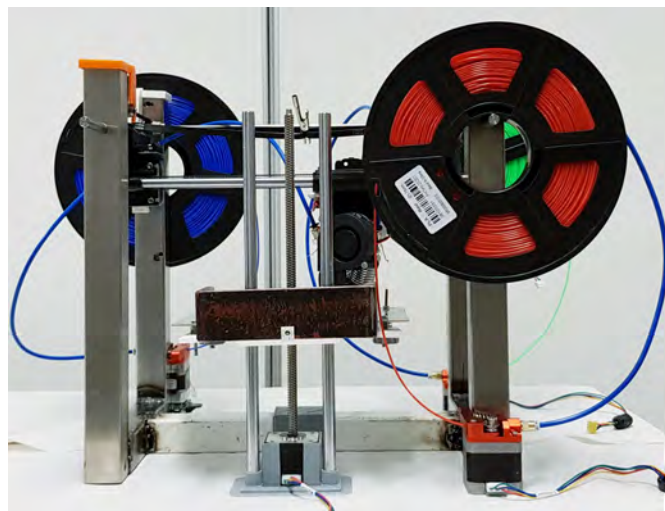


Figure 4.3: Back

Several 3D parts were printed to be able to assemble nozzle, heatsinks, and fans together and onto the Y-axis rods, see Figure 4.4. The X-axis step motor is fixed in place by tension force and friction between the 3D printed solution and frame, see Figure 4.5. The Bowden tube and extruders were also bought to save time and get high-quality components, see Figure 4.6.

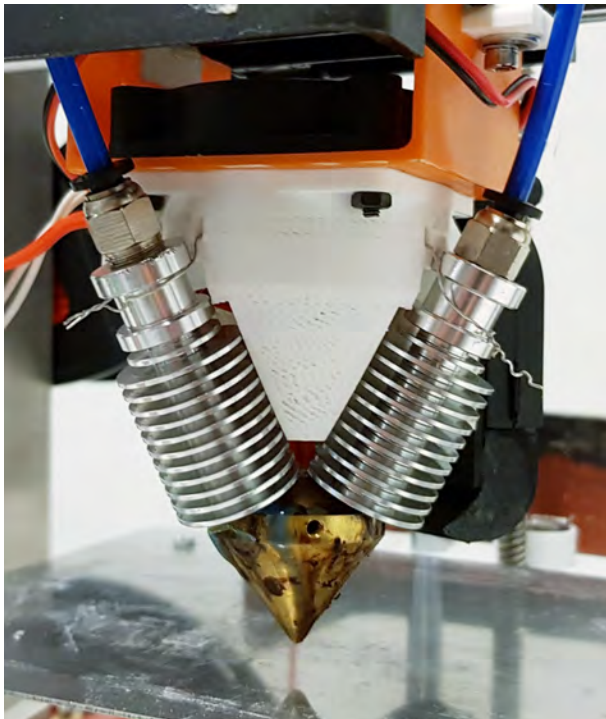


Figure 4.4: Diamond Hotend, fans and 3D printed parts that attach the hotend to linear rods

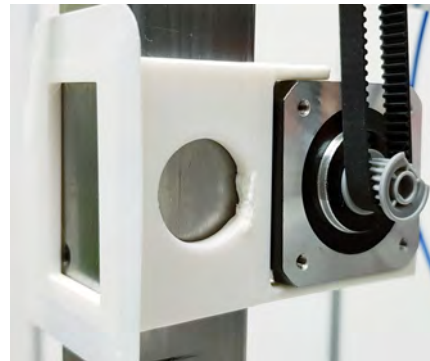


Figure 4.5: X-axis Assembly

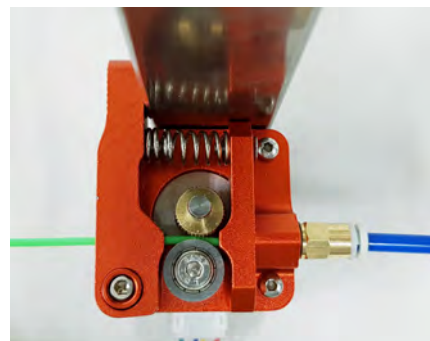


Figure 4.6: Extruder

## 4.2. Experiments

To test the prototype and answer the research questions, some experiments were done. There was a need to print something simple so that it was possible to examine how many colors the printer could generate, which required printing many samples of the same shape. The second criteria for the prints should be easy to measure dimensions which all considered led to print rectangles.

A CAD model with dimensions  $12mm$  in width,  $42mm$  in length, and  $2mm$  in height was created and turned into several different G-code for each print as explained in Section 3.2. After that, all dimensions of each sample were measured, and the following sample statistics were calculated using Python's pandas library, see Table 4.1. All values are given in millimeters (mm).

Table 4.1: Statistical information of measurements on rectangular prints

	Width ( $12mm$ )	Length ( $42mm$ )	Height ( $2mm$ )
count	45,000000	45,000000	45,000000
mean	11,928887	42,104676	1,562033
std	0,394388	0,525066	0,323058
min	11,100000	41,000000	0,900000
max	12,300000	42,800000	2,100000

The table illustrates the number of prints was 45, each having a different composition of colors. However, many mixtures were removed from the color spectrum until only distinct colors were left. This meant only half of them. Each spectrum consists of nine colors, varying from blue to red, shown in Figure 4.7, red to green, see Figure 4.8 and finally green to blue, see Figure 4.9. To conclude, colors red, blue, and green are given twice in those three spectra, which means the total number of distinct colors created by the prototype is 24.



Figure 4.7: Blue to red spectrum



Figure 4.8: Red to green spectrum



Figure 4.9: Green to blue spectrum

## **5. Discussion**

### **5.1. Color mixing**

One unsolved problem was one of the heatsinks getting clogged constantly. The other heatsinks were getting clogged, but after clearing them, they would work for a while. The leading assumption is the construction of the nozzle is inaccurate. Hence, only two color mixing.

The main problems besides heatsinks getting clogged were the different strengths in the extruder motors and different color strength of the filaments. Those factors led to the expected color from the code did not match with the printed color, so printed colors were generated by experimenting with the code.

### **5.2. Precision**

The printer has high precision in width and length of prints, but it is less precise in the height of prints due to non-consistent bed leveling causing pressure in between bed and nozzle, leading to shorter prints. However, without that, pressure prints do not stick to the bed well. A possible solution to that is to consider it while slicing and generating G-code. Moreover, instead of manual bed leveling in between each print, a sensor should be used.

### **5.3. Bed**

The U-formed weight on the bed had a crucial role in providing force balance. The rods only prevent rotation, thus opposing the torque from the motor and making the bed go up and down smoothly. One of the mistakes was while drilling the hole on the bed plate, it deformed. By flattening it afterward, the holes were moved from desired positions. So the bed plate should be flattened before each drilling of a hole so that the holes are in the desired position. At the beginning of printing, filament should stick to the bed, which was made possible by putting glue on the bed and heating the bed using the nozzle before printing.

## 5.4. Diamond Hotend

The project was centered around the Diamond Hotend [11]. It was very crucial that it worked properly, which it did apart from consistent clogging in one of the heatsinks caused by a melted filament from other extruders coming up and molding in the heatsink instead of going down through the  $0.4mm$  nozzle hole. The manufacturing of the nozzle might explain it. One of the assumptions was that nozzle was not made of brass and causing conductivity issues. But raising the temperature from  $185^{\circ}C$  to  $230^{\circ}C$  did not solve the problem.

Another assumption was that the nozzle holes were not made precisely. It was even leaking filament through heatsink threads, even if heatsinks were screwed as tight as possible. This caused issues with the thermal insulator between heatsinks and nozzle, which at last consist of more filament than thermal insulator material. Therefore, it was removed, but that caused a heat leak around the nozzle. So, the nozzle was heated up to a higher temperature of  $210^{\circ}C$ . Attempts were made to build even the Diamond Hotend [11] in KTH, but there were no tools in the KTH production facility for drilling a  $0.4mm$  hole as specified in CAD drawings [23]. Therefore, a cheap alternative was bought in order not to exceed budget limitations.

## 5.5. Motion

Extruders 0 and 1 had the same type of step motor, but it was impossible to find a motor of the same type for extruder 2, so three different step motors were tested. The first step motor could not extrude at all due to lack of power. The second one worked when it was only running at 100% but failed to extrude when combined with extruders 0 and 1. Finally, the third step motor was promising and worked very fine before the heatsink was clogged repeatedly, as described above.

High speed caused a shift in the X-axis between the print layers, which was solved by lowering the speed on the G-code. Stepper drivers had a crucial role both in controlling the position and extruding. Having different stepper drivers made it harder to make the software comply with the hardware accurately even after adjusting the reference voltage and specifying which kind of drivers each motor had in Marlin.



## 6. Conclusion

The primary goal of this project was to produce a fully functional FFF 3D printer which was achieved by applying skills and knowledge in mechanics, electronics, and programming. A prototype was built to illustrate a possible design for a multicolor 3D printer. Additionally, the printer was built so that it can print multiple colors, to be precise, 24 distinct colors. Therefore, a 3D printer can be designed as presented in this rapport to print multicolored objects simultaneously.

Various challenges arise when there is a single nozzle to combine and print several filaments by using a fused filament fabrication process. One of the challenges that emerged is the heatsink clogging problem which is non-substantial in single-color printing. Another challenge is that there is a need for more components such as stepper motors, drivers, and extruder expander, which increase the complicity and cost of hardware. Stepper motor and stepper driver become prominent when electronic pulses for extruding must comply perfectly with percentages of different colors given in G-code to obtain printed objects of the desired color.

Another important aspect is the difference in color strength of the filaments, meaning that, for example, the used blue filament was much darker blue meanwhile green filament was much lighter green. Proper bed and Z-axis construction are very crucial to have precise prints. The constructed Z-axis solution can be worked on by doing force and mechanical stress analysis in order to get more precise prints and smoother movement. After addressing that issue, more work can be done to improve the design aesthetically in general because our main focus was functionality.

A quick transition between colors is valuable for consumers so that they can print designs of complex nature. Therefore, more manufacturing methods on a similar system should be examined so that color mixing and color transition during printing can be done easily by any 3D hobbyist or user. All of our testings were conducted using the most common filament, PLA. More tests can be done by experimenting with other filament types. Even combining filament of different materials to get new materials could also be examined.

# Bibliography

- [1] N. Shahrubudin, T.C. Lee, and R. Ramlan. An Overview on 3D Printing Technology: Technological, Materials, and Applications. Available: <https://www.sciencedirect.com/science/article/pii/S2351978919308169?via%3Dihub>, 2019. Accessed: 12/02/2021.
- [2] John Steuben, Douglas L. Van Bossuyt, and Cameron Turner. Design for Fused Filament Fabrication Additive Manufacturing. *Volume 4: 20th Design for Manufacturing and the Life Cycle Conference; 9th International Conference on Micro- and Nanosystems*, 2015.
- [3] Anton Hylander and Malinda Blomqvist. *Choklad-3D-skrivare : En undersökning av chokladutskrifters egenskaper*. Degree project, first cycle, 15 credits, KTH, School of Industrial Engineering and Management (ITM) Stockholm, Sweden, 2020. Available: <http://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A1462017&dswid=7479>. Accessed: 12/02/2021.
- [4] Jinxian Qi, Chong Tan, and Cuiqiao Li. Technical Analysis of FDM Color 3D Printing Nozzle. In *Proceedings of the 2017 7th International Conference on Mechatronics, Computer and Education Informationization (MCEI 2017)*, pages 299–303. Atlantis Press, 2017.
- [5] Jessica Susan DiPersio, Robert Daniels Boulanger, and Alex Joseph Beaudoin. *Multihead 3D Printer*. Degree of Bachelor of Science In Mechanical Engineering, Worcester Polytechnic Institute, 2016. Available: [https://digital.wpi.edu/concern/student\\_works/db78td511](https://digital.wpi.edu/concern/student_works/db78td511). Accessed: 12/02/2021.
- [6] Acumen. An intuitive tool for simulating mathematical models of cyber-physical systems and for visualizing them as plots or in 3D. Available: <http://www.acumen-language.org/>, 2016. Accessed: 12/02/2021.
- [7] Taylor Landry. Extruders 101: A crash course on an essential component of your 3D printer. Available: <https://www.matterhackers.com/articles/extruders-101-a-crash-course-on-an-essential-component-of-your-3d-printer>, 2016. Accessed: 14/02/2021.
- [8] Tobias Hullette. Direct Drive vs Bowden Extruder Setup: The Differences. Available: <https://all3dp.com/2/direct-vs-bowden-extruder-technology-shootout/>, Mar 2021. Accessed: 25/04/2021.
- [9] Bo Alfredsson. *Handbok och förmelsamling i Hållfasthetslära*. Institutionen för hållfasthetslära KTH, 2014.

- [10] Sukindar, Nor Aiman and Ariffin, M. K. and Baharudin, B. T. and Jaafar, Che Nor and Ismail, Mohd Idris. Analyzing the effect of nozzle diameter in fused deposition modeling for extruding polylactic acid using open source 3D printing. *Jurnal Teknologi*, 78(10), 2016.
- [11] Klarmann, Alexander. Kickstarter: Two great hotends for 3D printers. Available: <https://all3dp.com/solution-for-multi-color-multi-material-3d-printers/>, 2015. Accessed: 14/02/2021.
- [12] RepRap. Diamond Hotend - Multi Color & Material 3D Printing. Available: <https://www.reprap.me/extruder/diamond-hotend>. Accessed: 25/04/2021.
- [13] ElProCus. Stepper Motor:Construction, Working, Types and Its Applications. Available: <https://www.elprocus.com/stepper-motor-types-advantages-applications/>. Accessed: 03/05/2021.
- [14] ElProCus. What is a Stepper Motor Driver? Available: <https://www.elprocus.com/what-is-a-stepper-motor-driver-types-and-its-applications/>. Accessed: 03/05/2021.
- [15] RepRap. RAMPS 1.4. Available: [https://reprap.org/wiki/RAMPS\\_1.4](https://reprap.org/wiki/RAMPS_1.4). Accessed: 03/05/2021.
- [16] T. Glad and L. Ljung. *Reglerteknik Grundläggande Teori*. Studentlitteratur, 2006.
- [17] Getting Started with Arduino MEGA2560. Available: <https://www.arduino.cc/en/Guide/ArduinoMega2560>. Accessed: 14/02/2021.
- [18] DomoticX Knowledge Center. Arduino MEGA Shield – RAMPS. Available: <http://domoticx.com/arduino-mega-shield-ramps/>. Accessed: 14/02/2021.
- [19] Ametherm. What Is An NTC Thermistor. Available: <https://www.ametherm.com/thermistor/what-is-an-ntc-thermistor>. Accessed: 14/02/2021.
- [20] Russell Johnny on RepRap . File:Arduinomegapololushieldschematic.png. Available: <https://reprap.org/wiki/File:Arduinomegapololushieldschematic.png>, 2011. Accessed: 14/02/2021.
- [21] What is Marlin? Available: <https://marlinfw.org/docs/basics/introduction.html>. Accessed: 14/02/2021.
- [22] GitHub. Marlin-2.0.X. Available: <https://github.com/MarlinFirmware/Marlin/archive/2.0.x.zip>. Accessed: 10/03/2021.
- [23] RepRap. File:Diamond Nozzle.pdf. Available: [https://reprap.org/wiki/File:Diamond\\_Nozzle.pdf](https://reprap.org/wiki/File:Diamond_Nozzle.pdf). Accessed: 03/05/2021.

# A. Appendix: CPS Acuman

```
1 // Bachelor's Thesis at ITM, KTH
2 // University: Royal Institute of Technology, KTH
3 // Date: 2021-04-20
4 // Authors: Davron Patkhullaev, Yonas Melake
5 // Examiner: Nihad Subasic
6 // TRITA number: TRITA ITM-EX 2021:23
7 // Course: MF133X, Degree Project in Mechatronics
8 // Description of the program:
9 // This is a basic simulation of the 3D printer's movement when printing.
10
11 // Creates a frame thickness - thickness of boxes
12 // len - default length of boxes
13 // cen - center of ram in xy-plane
14 // RamColor - color of ram
15 // exLen - Adds height of ram
16 // exZ - moves the ram i z-axis
17 // D - moves the ram in xyz-axis
18
19 model Ram(thickness, len, cen, RamColor, exLen, exZ, D) =
20   initially
21     _3D = ()
22   always
23     _3D = (
24       // Base
25       Box center=(cen-(len/2), cen, thickness/2+exZ)+D
26       size=(thickness, len, thickness) color=RamColor,
27       Box center=(cen+(len/2), cen, thickness/2+exZ)+D
28       size=(thickness, len, thickness) color=RamColor,
29       Box center=(cen, cen-(len/2-thickness/2), thickness/2+exZ)+D
30       size=(len-thickness, thickness, thickness) color=RamColor,
31       Box center=(cen, cen+(len/2-thickness/2), thickness/2+exZ)+D
32       size=(len-thickness, thickness, thickness) color=RamColor,
33
34       // Support
35       Box center=(cen+(len/2), cen-(len/2-thickness/2), thickness/2+len/2+
36       exLen/2+exZ)+D size=(thickness, thickness, len-thickness+exLen) color=
37       RamColor,
38       Box center=(cen-(len/2), cen+(len/2-thickness/2), thickness/2+len/2+
39       exLen/2+exZ)+D size=(thickness, thickness, len-thickness+exLen) color=
40       RamColor,
41       Box center=(cen+(len/2), cen+(len/2-thickness/2), thickness/2+len/2+
42       exLen/2+exZ)+D size=(thickness, thickness, len-thickness+exLen) color=
43       RamColor,
44       Box center=(cen-(len/2), cen-(len/2-thickness/2), thickness/2+len/2+
45       exLen/2+exZ)+D size=(thickness, thickness, len-thickness+exLen) color=
46       RamColor,
```

```

35
36 // Top
37 Box center=(cen-(len/2),cen,thickness/2+len+exLen+exZ)+D
    size=(thickness, len, thickness) color=RamColor ,
38 Box center=(cen+(len/2),cen,thickness/2+len+exLen+exZ)+D
    size=(thickness, len, thickness) color=RamColor ,
39 Box center=(cen,cen-(len/2-thickness/2),thickness/2+len+exLen+exZ)
+D size=(len-thickness,thickness,thickness) color=RamColor,
40 Box center=(cen,cen+(len/2-thickness/2),thickness/2+len+exLen+exZ)
+D size=(len-thickness,thickness,thickness) color=RamColor)
41
42
43
44
45 // Creates two cylinder that are across each other
46 // cen - center of ram in xy-plane
47 // len - length of cylinder
48 // r - radius of cylinder
49 // D1 - moves one of the cylinder in xyz-axis
50 // D2 - moves one of the cylinder in xyz-axis
51 model cross_cyl (cen, len, r, D1, D2) =
52   initially
53     _3D = ()
54   always
55     _3D = (
56       Cylinder center=(cen,cen,len+3*r)+D1
57               radius=r length=len color=blue
58               rotation=(-1*pi/2,1*pi/2,0)
59       Cylinder center=(cen,cen,len+5*r)+D2
60               radius=r length=len color=yellow
61               rotation=(1*pi/2,-1*pi/2,pi/2))
62
63 // Just used for holding three variables(x,y,z)
64 model statePos(x,y,z) =
65   initially
66     x = 0,
67     y = 0,
68     z = 0
69
70 // Creates filament that drops with a constant speed
71 // x,y,z - filament position
72 // speed - a constant speed
73 // c - color of filament
74 model filament(x,y,z,speed, c) =
75   initially
76     z' = speed,
77     _3D = ()
78   always
79     z' = speed,
80     _3D = (
81       Box center=(x,y,z) size=(0.01,0.01,0.05) color=c)
82

```

```

83 model Main (simulator) =
84   initially
85     mode = "Init",
86     x    = 0, // Used to move the cylinders and hotend
87     x'   = 0.4,
88     y    = 0, // Used to move the bed
89     y'   = -0.05, // The speed of filament and bed
90
91     len= 0.8,
92     SPos = create statePos(0,0,1),
93     state = 0,
94
95     R = create Ram(0.05,0.8,0,red,0,0, (0,0,0)), // The ram of the 3D-
96     printer
97     h = create Ram(0.01,0.07,0,green,0.025+0.01*2,0.8+0.025*2-0.01*3/2,
98     (0,0,0)), // The "hotend" of the 3D-printer
99     C = create cross_cyl (0,0.8, 0.025,(0,0,0), (0,0,0)), // Cylinders that
100     "moves" the hotend
101
102     _3D = (),
103     _3DView = ((-17,-20,15)/5,(0,0,0.5)) // Changes the 3Dview
104 always
105     x' = 0.4,
106     y' = -0.05,
107     // In state 0, the hotend moves in positiv y direction
108     // In state 1, the hotend moves in positiv x direction
109     // In state 2, the hotend moves in negativ x,y direction
110     // In state 3, the hotend moves in positiv x,y direction
111     if state == 0 && C.D1(1)>(len/2-0.1) then state = 1
112     elseif not(state == 0) && C.D2(0)>(len/2-0.1) then state = 2
113     elseif state == 2 && C.D2(0)<-(len/2-0.1) then state = 3 noelse,
114
115     if state == 0 then C.D1 = (0,x, 0), h.D =(0,x, 0), SPos.x = x
116     elseif state == 1 then C.D2 = (x-SPos.x,0, 0), h.D =(x-SPos.x,SPos.x,
117     0), SPos.y = 2*x-SPos.x
118     elseif state == 3 then C.D2 = (x-SPos.x,0, 0),C.D1 = (0,x-SPos.x, 0), h
119     .D = (x-SPos.x,(x-SPos.x), 0), SPos.y = 2*x-SPos.x
120     elseif state == 2 then C.D2 = -(x-SPos.y,0, 0),C.D1 = -(0,x-SPos.y, 0)
121     , h.D = -(x-SPos.y,(x-SPos.y), 0), SPos.x = 2*x-SPos.y noelse,
122
123     _3D = (
124     // The bed of the 3D-printer that moves down from the top of the ram
125     Box      center=(0,0,len+y) size=(0.7,0.7,0.05) color=blue,
126     Text     center=(-1,1, 1.2) size=0.5 color=red
127     rotation=(1.5 - pi/2,0,-0.3) content="3D-printer")
128     ,create filament(h.D(0),h.D(1),len,y',green)

```

## B. Appendix: CAD Frame

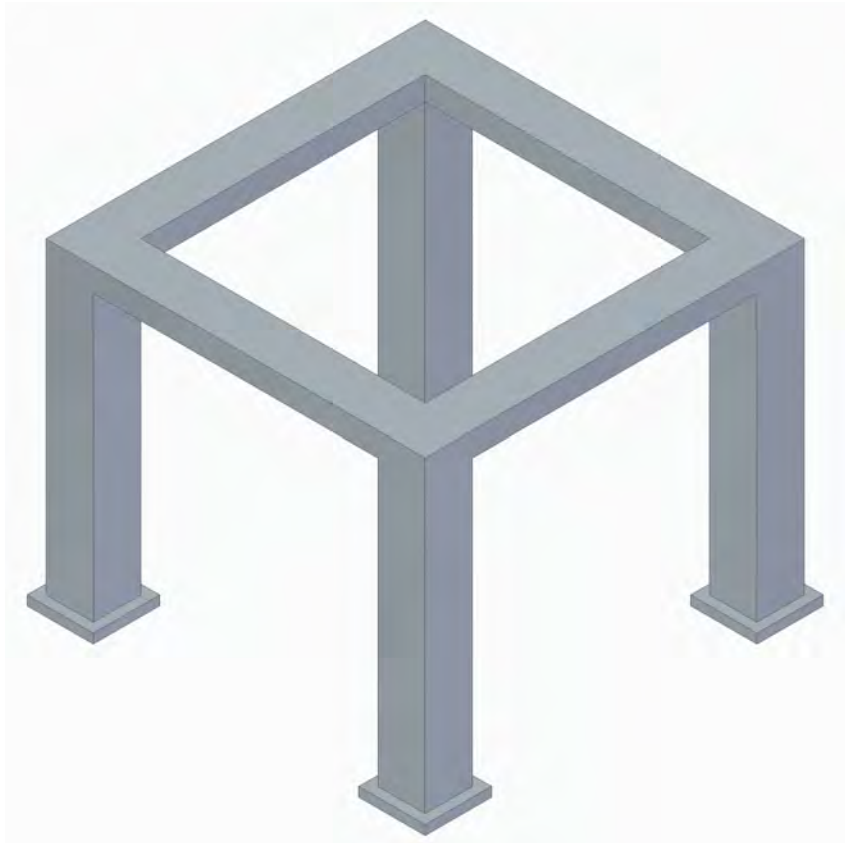


Figure B.1: Frame 1 - Created in Solid Edge 2020

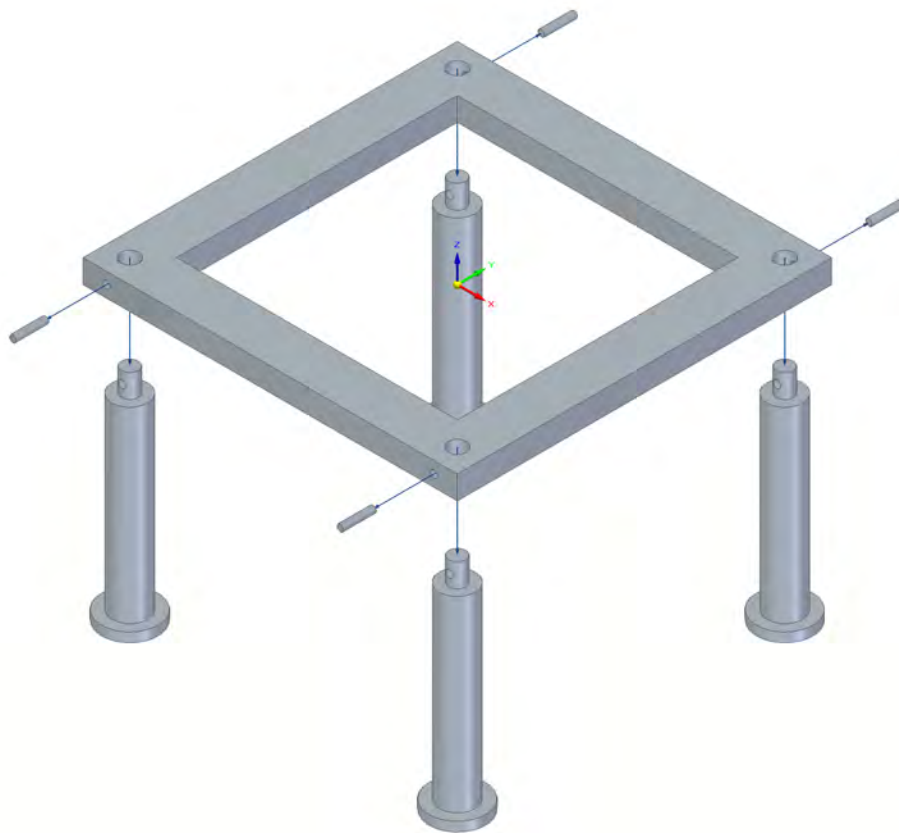


Figure B.2: Frame 2 - Created in Solid Edge 2020



# C. Appendix: Changes in Marlin

Table C.1: Marlin - Changed configurations

Before	After
In Configuration_adv.h	
<code>//#define SDCARD_CONNECTION LCD</code>	<code>#define SDCARD_CONNECTION LCD</code>
In Configuration.h	
<pre>//#define MIXING_EXTRUDER #define MIXING_STEPPERS 2 //#define X_DRIVER_TYPE A4988 //#define Y_DRIVER_TYPE A4988 //#define Z_DRIVER_TYPE A4988 //#define E0_DRIVER_TYPE A4988 //#define E1_DRIVER_TYPE A4988 //#define E2_DRIVER_TYPE A4988 #define X_BED_SIZE 200 #define Y_BED_SIZE 200 #define Z_MAX_POS 200 //#define SDSUPPORT</pre>	<pre>#define MIXING_EXTRUDER #define MIXING_STEPPERS 3 #define X_DRIVER_TYPE A4988 #define Y_DRIVER_TYPE A4988 #define Z_DRIVER_TYPE A4988 #define E0_DRIVER_TYPE A4988 #define E1_DRIVER_TYPE A4988 #define E2_DRIVER_TYPE DRV8825 #define X_BED_SIZE 25 #define Y_BED_SIZE 120 #define Z_MAX_POS 50 #define SDSUPPORT</pre>
In pins_RAMPS.h	
<pre>#ifndef X_CS_PIN #define X_CS_PIN 49 #endif #ifndef Y_CS_PIN #define Y_CS_PIN 53 #endif #ifndef Z_CS_PIN #define Z_CS_PIN 40 #endif #ifndef E0_CS_PIN #define E0_CS_PIN 42 #endif #ifndef E1_CS_PIN #define E1_CS_PIN 44 #endif Empty Empty Empty</pre>	<pre>//ifndef X_CS_PIN // #define X_CS_PIN 49 //endif //ifndef Y_CS_PIN // #define Y_CS_PIN 53 //endif //ifndef Z_CS_PIN // #define Z_CS_PIN 40 //endif //ifndef E0_CS_PIN // #define E0_CS_PIN 42 //endif //ifndef E1_CS_PIN // #define E1_CS_PIN 44 //endif #define E2_STEP_PIN 42 #define E2_DIR_PIN 40 #define E2_ENABLE_PIN 63</pre>

## D. Appendix: Components of Z-axis

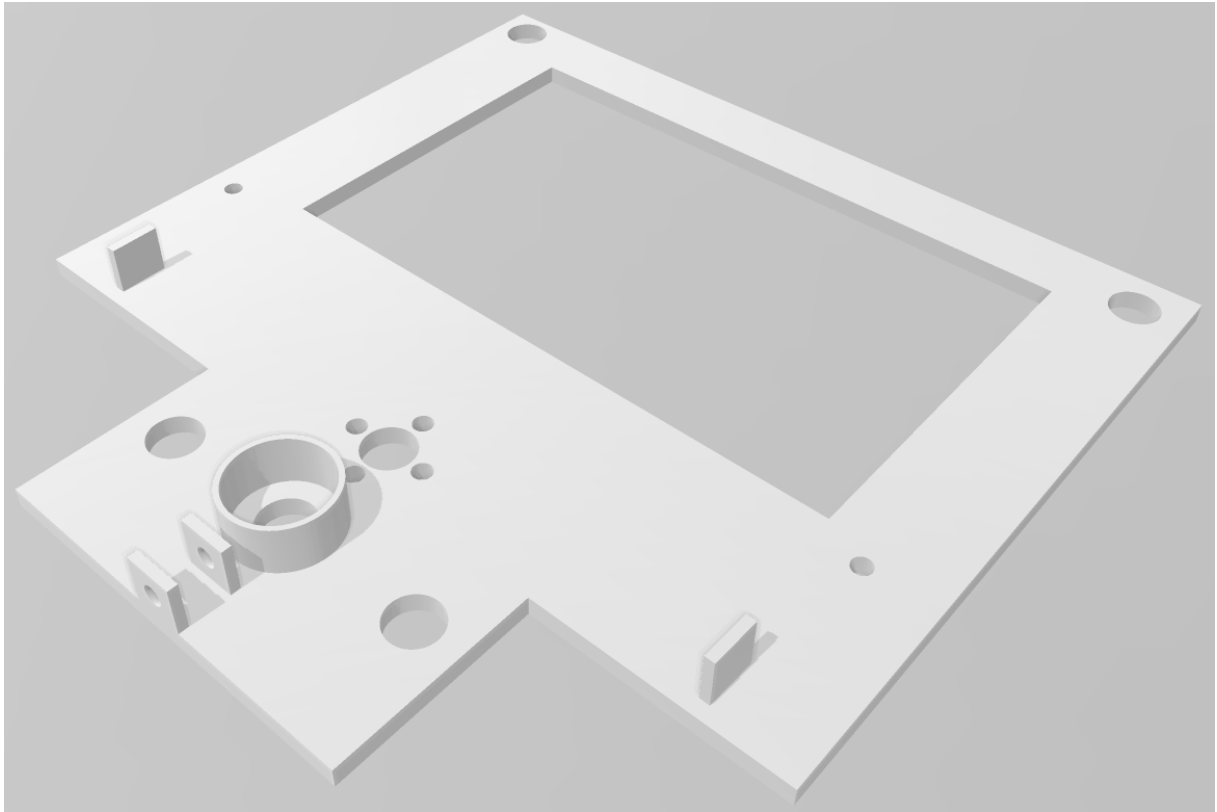


Figure D.1: Final bed supporter - Created in Solid Edge 2020

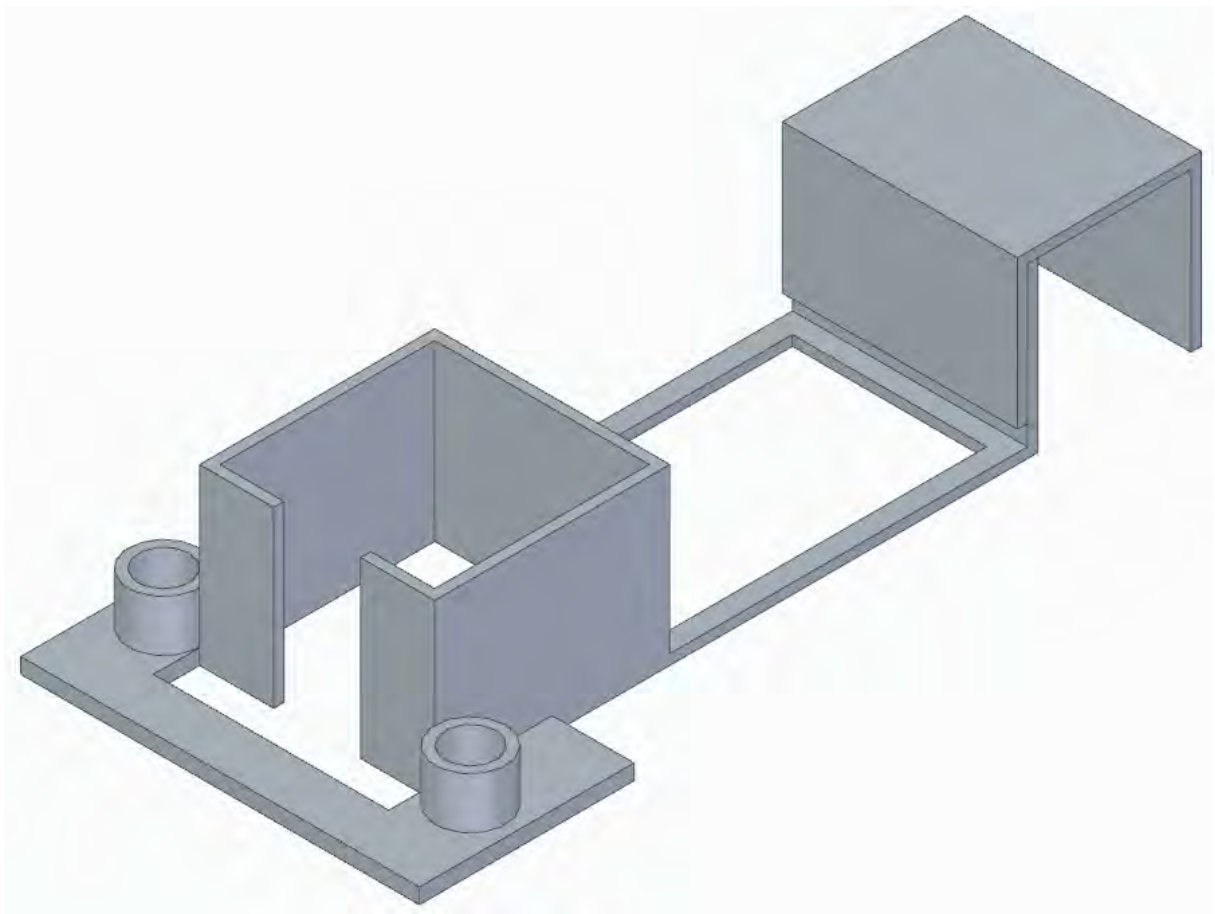


Figure D.2: Motor and smooth rods holder - Created in Solid Edge 2020

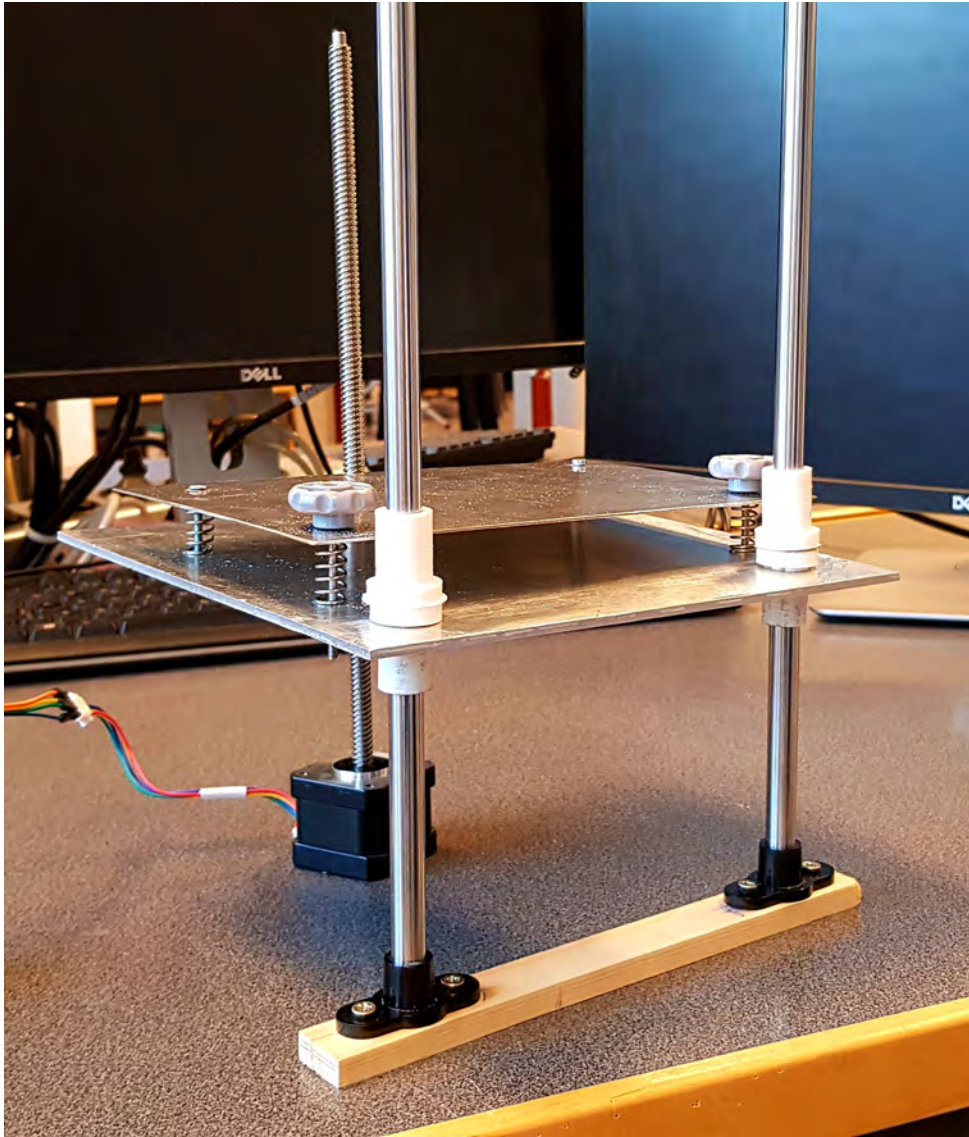


Figure D.3: First Z-axis Assembly

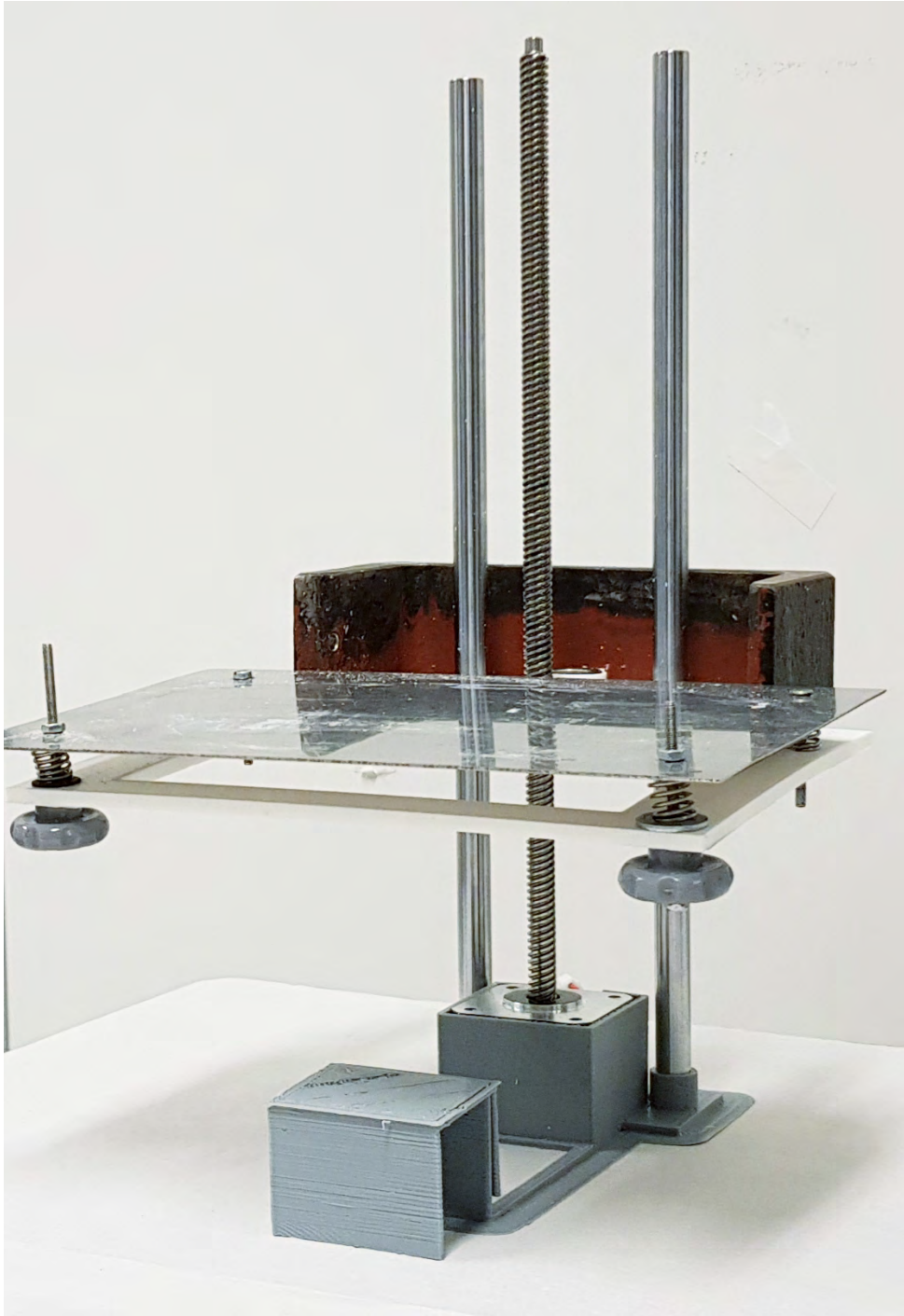


Figure D.4: Final Z-axis Assembly

## E. Appendix: CAD Components of XY-plane

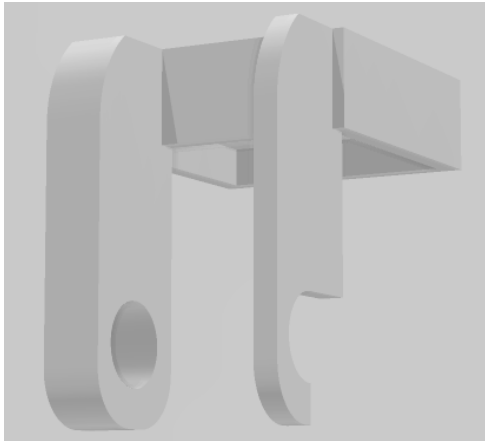


Figure E.1: Bearing holder 1 - Created in Solid Edge 2020

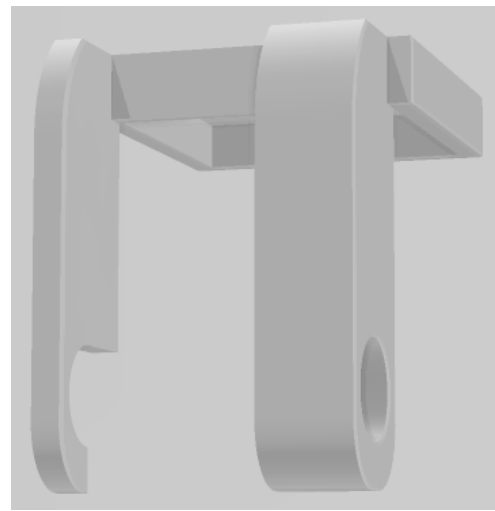


Figure E.3: Bearing holder 3 - Created in Solid Edge 2020

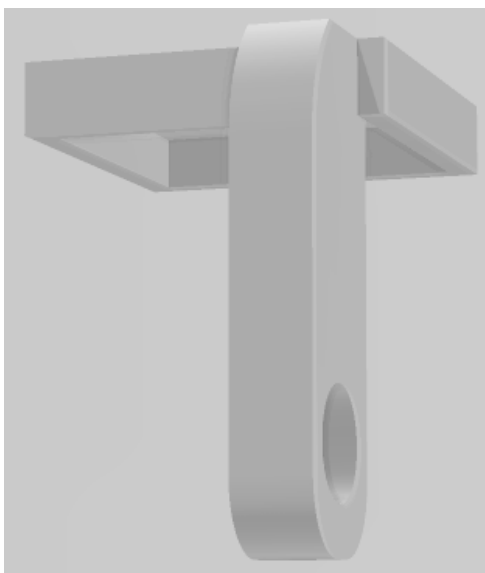


Figure E.2: Bearing holder 2 - Created in Solid Edge 2020

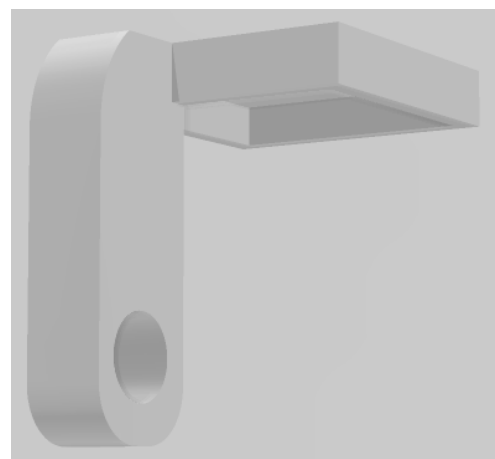


Figure E.4: Bearing holder 4 - Created in Solid Edge 2020

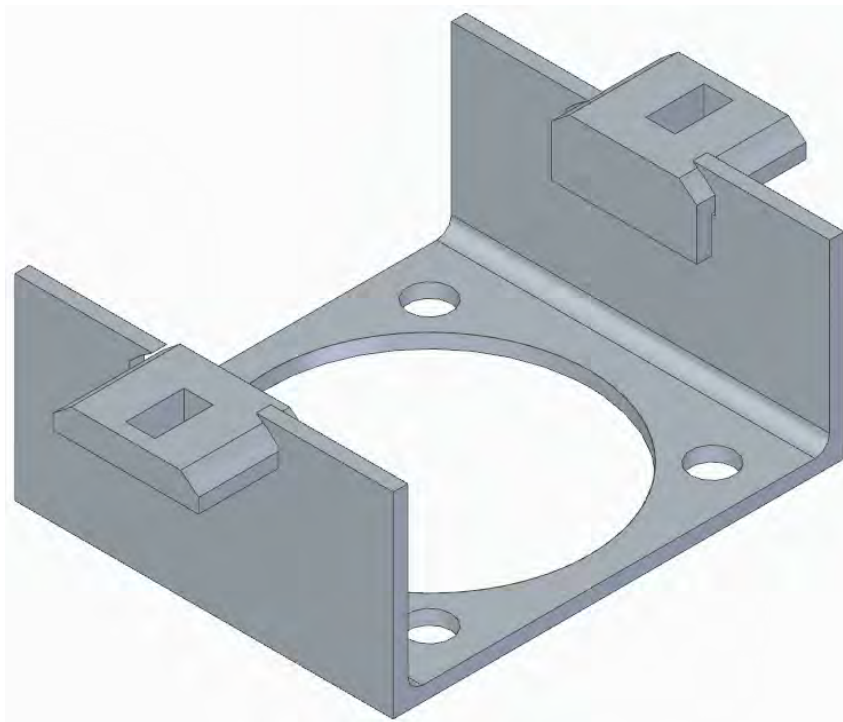


Figure E.5: Nozzle mount - Created in Solid Edge 2020

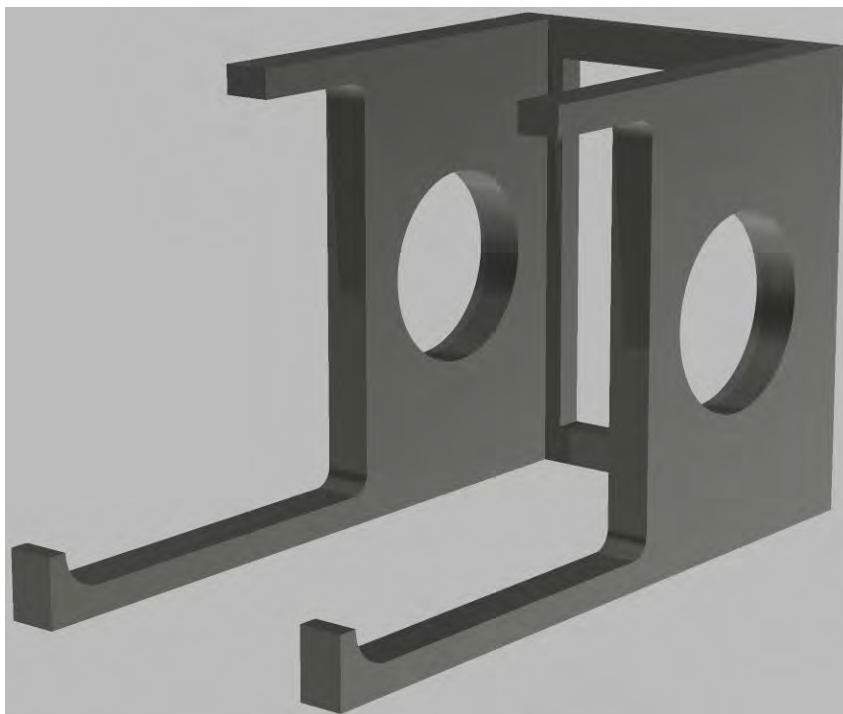


Figure E.6: Motor Holder - Created in Solid Edge 2020

## F. Appendix: 3D photos of samples



Figure F.1: 3D photos of samples from side

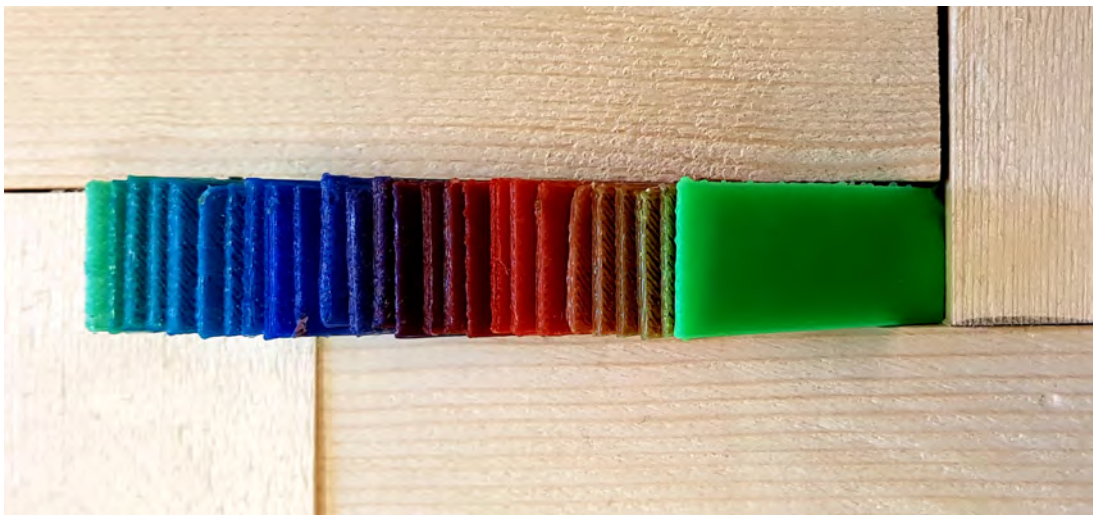


Figure F.2: 3D photos of samples from top





TRITA-ITM-EX 2021:23