

April 2017

# Building a 3D Printer: Motors and Controls

Cameron A. Hastings  
*Worcester Polytechnic Institute*

Daniel Stephen Pfaff  
*Worcester Polytechnic Institute*

William Whidden Gorman  
*Worcester Polytechnic Institute*

Follow this and additional works at: <https://digitalcommons.wpi.edu/mqp-all>

---

## Repository Citation

Hastings, C. A., Pfaff, D. S., & Gorman, W. W. (2017). *Building a 3D Printer: Motors and Controls*. Retrieved from <https://digitalcommons.wpi.edu/mqp-all/9>

This Unrestricted is brought to you for free and open access by the Major Qualifying Projects at Digital WPI. It has been accepted for inclusion in Major Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact [digitalwpi@wpi.edu](mailto:digitalwpi@wpi.edu).



Project Number: JMS-1705

# Building a 3D Printer: Motors and Controls

A Major Qualifying Project Report  
Submitted to the Faculty  
of  
WORCESTER POLYTECHNIC INSTITUTE  
In Partial Fulfilment of the requirements for the  
Degree of Bachelor of Science  
In Mechanical Engineering  
By

---

William Gorman

---

Cameron Hastings

---

Daniel Pfaff

# Abstract

3D printers require a reliable and robust control system to provide the proper quality for printed parts. The goal of this project was to design and implement the electronic and software controls for the large scale 3D printer. This system was able to run all three print heads as well as color mixing in the central diamond print head. The resulting design for the control system utilized a RUMBA control board running with Repetier-Firmware as the printer firmware. A separate system was designed for the heat beds, which used an Arduino Uno to control a set of relays to maintain the set temperature to the heat beds.

# Authorship

**Cover Page:** Cameron Hastings

**Abstract:** Daniel Pfaff

**Authorship:** William Gorman, Cameron Hastings, Daniel Pfaff

**Table of Contents:** William Gorman, Cameron Hastings, Daniel Pfaff

**List of Figures:** William Gorman, Cameron Hastings, Daniel Pfaff

**List of Tables:** William Gorman, Cameron Hastings, Daniel Pfaff

**Introduction:** William Gorman

**Background:** William Gorman, Cameron Hastings, Daniel Pfaff

**Methodology:** William Gorman, Cameron Hastings, Daniel Pfaff

**Results:** William Gorman, Cameron Hastings, Daniel Pfaff

**Conclusion:** William Gorman, Cameron Hastings, Daniel Pfaff

**References:** William Gorman, Cameron Hastings, Daniel Pfaff

**Appendix A:** Daniel Pfaff

**Appendix B:** William Gorman, Cameron Hastings, Daniel Pfaff

# Table of Contents

Cover Page	i
Abstract	ii
Authorship	iii
Table of Contents	iv
List of Tables	vii
List of Figures	viii
1.0 Introduction	1
1.1 The Major Qualifying Project	2
1.2 Three Dimensional Printers and Additive Manufacturing	3
2.0 Background	4
2.1 Components for Basic Control	4
2.1.1 Stepper Motors	4
2.1.2 Power Supply	5
2.1.3 Control Board	7
2.1.4 Drivers	10
2.1.5 Endstops	11
2.1.6 Firmware	12
2.1.7 Thermistors	14
2.1.8 Printing Software	16
2.2 Arduinos and LCDs	16
2.3 Multiple Extruder Printing	17
3.0 Methodology	18
3.1 Motors and Controls Design	18
3.1.1 Design Considerations	19

3.1.2 Wire Selection	20
3.2 Components Justifications	22
3.2.1 Stepper Motors	22
3.2.2 Power Supplies	23
3.2.3 Control Board/CPU	25
3.2.4 Heat Bed and Temperature Controls	26
3.2.5 Non-Critical Elements	26
3.2.6 Initial Part Orders	26
3.3 Exploratory Assembly and Basic Testing	28
3.4 Final Assembly and Programming	30
3.4.1 Sideboard and Mounting	30
3.4.2 Wiring and Wire Management	32
3.4.3 Control Panel Assembly	33
3.4.4 Full 3D Printer Assimilation	34
3.5 Detailed Designs and Schematics	35
3.5.1 Heat Bed Wiring	35
3.5.2 Heat Bed Coding	37
3.5.3 Onboard LCD Wiring	41
4.0 Results and Discussion	42
4.1 Design Modifications	43
4.1.1 Firmware	43
4.1.2 Heat Beds	43
4.1.3 Control Panel	44
4.2 Manufactured Parts	45
5.0 Conclusion	49
5.1 Large Print Volume and Parallel Head Conclusions	50

5.2 Multicolor and Color Mixing Conclusions	50
6.0 References	51
7.0 Appendices	
Appendix A: Overall 3D Printer Electronics Design	53
Appendix B: Parts List and Budget	54

# List of Tables

Table 2.1: Control Boards	8
Table 2.2: Available 3D printing Firmware	13
Table 3.1: Wire Sizing and Coloring	21
Table 3.2: Initial Parts Order A	27
Table 3.3: Initial Parts Order B	27
Table B.1: Parts List and Budgeting	54



# List of Figures

Figure 2.1: Stepper Motor Cutaway	5
Figure 2.2: Ramps 1.4 Control Board	9
Figure 2.3: Rumba Control Board	10
Figure 2.4: Thermistor	15
Figure 3.1: Stepper Motor Dimensions	23
Figure 3.2: Mean Well Power Supply	24
Figure 3.3: EPBOWPT 360W Power Supply	24
Figure 3.4: RepRap Rumba Control Board	25
Figure 3.5: Exploratory Testing Software View	29
Figure 3.6: Stepper Motor Driver Testing	29
Figure 3.7: Exploratory Wiring	30
Figure 3.8: Sideboard Mounting	31
Figure 3.9: Front Panel and Front Panel Enclosure	34
Figure 3.10: Arduino Controlled Temperature Regulator Circuit	36
Figure 3.11: Thermistor Circuit for the Arduino	37
Figure 3.12: Initial Testing Circuit	38
Figure 3.13: LCD Code	39
Figure 3.14: Declaring and Initializing the Keypad	39
Figure 3.15: Keypad Function Code	40
Figure 3.16: Heat Bed Safety Turn Off Code	41
Figure 4.1: Final Side Panel Layout	42
Figure 4.2: Multiple Extruder Printed Parts	45
Figure 4.3: Gompei Statue Print	46
Figure 4.4: Heat Bed Leveling and Adhesion Test Print	46

Figure 4.5: Multi-Color Printing Tests	47
Figure 4.6: Color Mixing Tests	47
Figure 5.1: Printer Presentation Display	49
Figure A.1: Controls and Motors Wiring	53

# 1.0 Introduction

The additive manufacturing industries net worth is expected to exceed twenty billion dollars by 2020, tripling its current gross market value [4]. The projected cause of this sudden expansion is the upcoming assimilation of 3D printing technologies into our mass manufacturing methods. At present, additive manufacturing is primarily used in rapid prototyping and design refinement. However, as the technology becomes more prevalent, economical, and reliable, it will soon be possible to additively manufacture everything from housing and food needs to weapons and armor. In order to better understand how the 3D printer featured in this report caters to the upcoming needs of the industry, one must first be familiar with the current state of additive manufacturing and the technological progression used to reach this point.

According to Kyle Maxey of engineering.com, the first 3D printers were made in 1984 by Charles Hull [10]. More specifically, Mr. Hull invented stereo-lithography, without which we would be unable to produce components by additive manufacturing. Stereo-lithography is the process of representing an object in Cartesian vector space by layers, similar to how a topographical map is viewed. This graphical advancement allows us to print parts by laying filament onto a bed one layer at a time, producing the desired geographies in three dimensions. With the foundations of the additive manufacturing industries laid, 3D printing quickly found applications as a rapid prototyping technology. By 1995 parts were being consistently and accurately manufactured, and the biomedical industry was beginning to see potential applications for this technology in medicine. The first 3D printed, functioning organs and prosthetics were marketed around the turn of the millennia, and by 2010 the industry had gained enough momentum to have applications in the military, medical and manufacturing fields.

However, the largest and most advanced 3D printers are unique and expensive devices far outside the reach of most non-corporate entities. Knowing that the additive manufacturing industry is about to expand well beyond its current bounds, this yields a perfect opportunity to

prove that a large, versatile and modern 3D printer can be built on a civilian budget. If this is proven to be possible, it will open the rapidly expanding market to individuals and small companies, irreversibly changing the course of the additive manufacturing industries development.

## 1.1 The Major Qualifying Project

Worcester Polytechnic Institute's unique program affords multiple opportunities for students to work on long term, large scale, group oriented projects. The crowning achievement of a student's time at WPI is their Major Qualifying Project (MQP). During MQP, students work in small or medium teams to complete a more challenging project than anything that they have attempted before during their time at WPI. These projects cover a wide range of topics, from experimental research to design and manufacturing of modern devices.

As discussed in the introduction, the 3D printing market is on the verge of an exponential growth point. In recognition of this fact, Professor John Sullivan proposed an MQP designed to educate students in the field of additive manufacturing. This MQP aimed to design, build, test, calibrate and refine a state of the art 3D printer on an affordable budget. Because of the enormous scale of this undertaking, three separate MQP teams were formed, each tasked to complete one of the subsystems of the additive manufacturing device and to assimilate their system into the final design.

Cameron Hastings, Daniel Pfaff, and William Gorman volunteered to complete the Motors and Controls portion of the 3D printer. This team was tasked to research, design, purchase, and assemble the controllers, motors, sensors, and power supplies needed for the additive manufacturing device. During the initial meetings following the team organizations, Professor Sullivan was able to compile a list of design criteria for the final device. The 3D printer which we built has a much larger bed than any other device on campus, with multiple heads

capable of printing in parallel, and could potentially have the ability to automatically print in multiple colors. Later meetings and research revealed the potential for a red/blue/green color scale using mixed filaments, which then became an additional goal should the printer be assembled ahead of schedule.

## 1.2 Three Dimensional Printers and Additive Manufacturing

A 3D printer is a type of manufacturing tool which allows people to turn computer drawn models into a real world objects by adding multiple layers of material. This way of creating objects is known as additive manufacturing and is the primary reason that 3D printing is unique among manufacturing tools. The technology helps people create affordable, custom made, precision parts. These custom parts can be useful for prototyping, student projects, and sharing ideas.

There are many uses for 3D Printers. While plastic is currently the most common material, there are 3D printers that can print in metal, concrete, biomaterial, and food. This gives the technology a wide range of applications.

## 2.0 Background

The background contains our initial research and the prerequisite knowledge needed to understand the component justifications and initial parts orders.

### 2.1 Components for Basic Control

This section describes the electrical and mechanical components used to control the 3D printer. It is subdivided to allow for easier navigation.

#### 2.1.1 Stepper Motors

Stepper motors are DC motors that have multiple coils allowing them to move in small increments. These coils are “organized in groups called ‘phases’”. By energizing each phase in sequence, the motor will rotate, one step at a time.” [5]. By having a computer control these movements, they can be very precise. They are best used for positioning, speed control, and low speed torque.

While they can be very useful, they do have their limitations. To start, “stepper motor current consumption is independent of load.” [5] When a motor is not moving, it will still consume a large amount of current. This can make them run hot. Stepper motors also tend to lose torque the faster they spin [5]. Some stepper motors are designed for high speed torque, but the performance of these motors are also greatly dependent on the stepper drivers paired with them. Unlike servo motors, stepper motors do not provide feedback on its location. It is important to consider these limitations before incorporating them into a design.

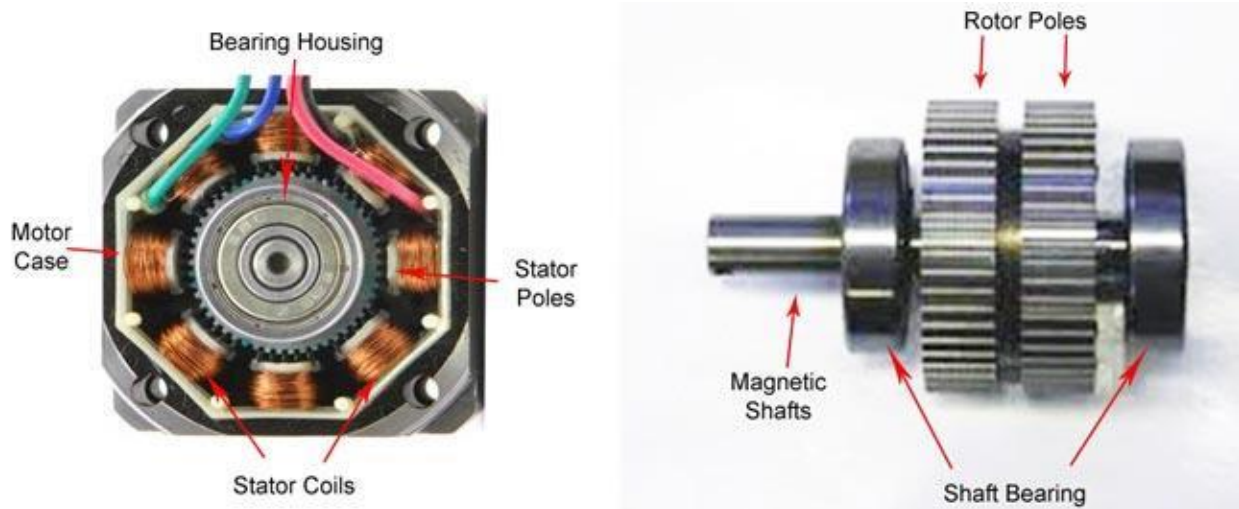


Figure 2.1: Stepper Motor Cutaway

Stepper motors come in many different shapes and sizes, and can differ in gearing ratios, wiring setup, step count, and shaft styles [5]. Choosing the correct stepper motor for a desired task is greatly dependent on those characteristics.

### 2.1.2 Power Supply

A 3D printer requires electricity in order to operate its motors, control boards, and heating elements. The most readily available source of electricity is a standard wall outlet. However, the 110-120V AC power cannot directly power the 3D printer. The components require lower-voltage DC power to operate, and would quickly burn out from the high voltage from the outlet. Therefore, a power supply unit (PSU) is necessary to convert the 110-120V AC to DC voltage, typically 12-24 V, which can be used to power the printer. There are two main designs for a PSU circuit: linear power supply or switch-mode power supply. Both designs share the same basic components: a transformer used to lower the AC voltage along with a rectifier diode, a filter capacitor, and a voltage regulator used to convert AC voltage to DC. The main difference between the two designs is the switch-mode power supply uses a high frequency AC input to the transformer, which reduces power loss and heat production in the transformer. This makes the switch mode power supply much more complicated than the linear power supply, but also allows it to deliver more power at a lower cost and weight since it has a lower power loss

and a smaller transformer [3]. Most power supplies that are applicable for use in a 3D printer are switch-mode power supplies.

There are a variety of power supplies available to use for a 3D printer. Some power supplies are designed for use in 3D printers or CNC machines. These power supplies are very reliable and built to high quality standards, but are also very expensive and limited in availability [11]. Some alternative power supplies that are often used in 3D printers are LED light strip PSUs and ATX PSUs. LED light strip power supplies, as their name suggests, are power supplies used to power banks of LED light strips. Advanced Technology Extended (ATX) power supplies were developed and configured for use in personal computers (PCs). Both types of power supplies are inexpensive and widely available (about \$20-200, depending on the power output), although ATX power supplies have a longer history and stricter regulations than LED strip power supplies. In addition to these options, there are other switch-mode power supplies available that do not have a dedicated function. The selection of the power supply is mainly dependent on the power required by the printer. ATX and LED strip power supplies can provide from 200W to more than 1000W, depending on the power supply, while other power supplies can provide less power if needed. To calculate the power required by the printer, add up the amperage required by each component and multiply by the DC voltage [2]. The chosen power supply should at least be able to supply all the components of the printer with their required current draw.

Installation of the power supply is fairly simple and straightforward, although the exact steps depend on the type of power supply used. An ATX power supply uses a standard PC power cable as its input, and outputs DC voltage at 12V, 5V, and 3.3V. The 12V output can be wired to the control board of the printer, which then distributes power to the other components of the printer. Additionally, in many ATX power supplies, a 5V signal output from the PSU must be connected to ground through a resistor in order for the PSU to turn on. Other power supplies might just have a 12V output and require an input plug to be wired to ports on the power supply.



Independent of the power supply set up, proper safety precautions must be followed when working with the PSU. Mains power is dangerous, and can lead to injury if safety measures are not taken. Whenever someone is working on the PSU, ensure that the power cable is unplugged from the wall outlet. Additionally, wait 5-10 minutes after unplugging the PSU before working on the PSU to ensure the internal capacitors are fully discharged. Be sure to use components that are rated to handle the power being supplied to reduce the risk of fire. Finally, when working with electrical connections, make sure that all connections are properly insulated and grounded. Following these safety procedures is necessary to prevent the risk of injury.

### 2.1.3 Control Board

The Control board is made up of two components: a microcontroller and a circuit board. These components can be either combined into one board, or be separate units attached to each other. Both work simultaneously to control and distribute power to all other components listed below. It is referred to as the brains and the central nervous system of the printer. Without it, nothing else would work.

Even though there are many different control boards on the market, they all essentially perform the same task. The main differences are their features, capabilities, cost, and reliability. Table 2.1 summarizes the control boards that we considered.

Board Name	Features and Capabilities	Single Unit?	Cost
RAMPS 1.4	<ul style="list-style-type: none"> <li>- 5 Stepper motor max</li> <li>- Ability to be modified</li> <li>- Massive online support</li> <li>- Serviceable</li> </ul>	NO - Requires an Arduino Mega microcontroller	\$9.59
MKS v1.4	<ul style="list-style-type: none"> <li>- 5 Stepper motor max</li> <li>- Great Heat dissipation</li> <li>- 4 layer circuit board</li> <li>- 12v to 24v power input</li> </ul>	YES	\$75.00
Replicape	<ul style="list-style-type: none"> <li>- 5 Stepper motor max</li> <li>- High power stepper drivers (DRV8825)</li> <li>- MosFet Drivers for cooler operating temperatures</li> <li>- Powerful development board (BeagleBone Black)</li> <li>- Blown fuse indicator</li> <li>- 12v to 24v power input</li> </ul>	NO - Requires a BeagleBone Black microcontroller	\$99.00
Megatronics	<ul style="list-style-type: none"> <li>- 5 Stepper motor max</li> <li>- 4 Thermistor are supported</li> <li>- 4 layer circuit board</li> <li>- 6 stepper drivers are supported</li> </ul>	YES	€ 79.99
RUMBA	<ul style="list-style-type: none"> <li>- 6 Stepper motor driver max</li> <li>- High power stepper drivers (DRV8825)</li> </ul>		\$113.00

Table 2.1: Control Boards

All the control boards listed in Table 2.1 support up to four, five, or six stepper motor drivers, but are not designed to handle more than that limit. This puts a restraint on the number of extruders we can have. We view this as a setback to our goal of having more than two extruders for simultaneous printing.

Our top two competitors are the RAMPS 1.4 board and the RUMBA board. Both offer features that we believe would allow us to reach our goal.

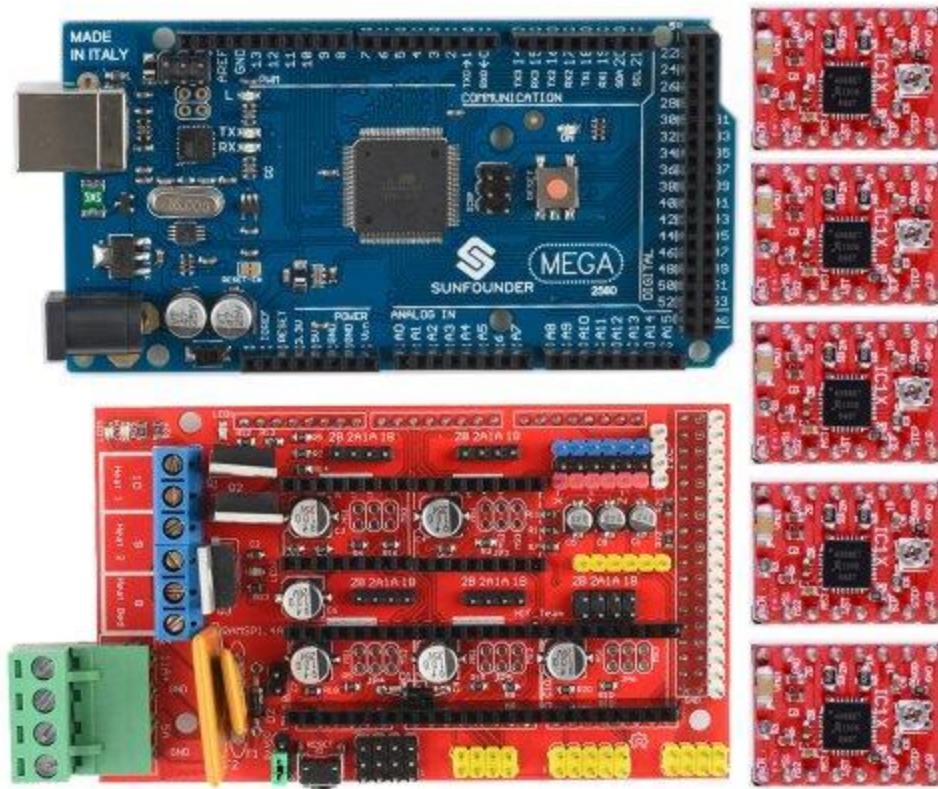


Figure 2.2: RAMPS 1.4 Control Board

The RAMPS 1.4 control board offered the best features for the price. It uses a microcontroller and shield setup, which makes it more serviceable than a two-in-one combination, such as the MKS v1.4. If something on the board were to fail, we would not have to replace the entire unit. We would be able to reuse the Arduino microcontroller and save money. The major setback of this board is the five stepper driver limit. It would limit us to only two colors per part.

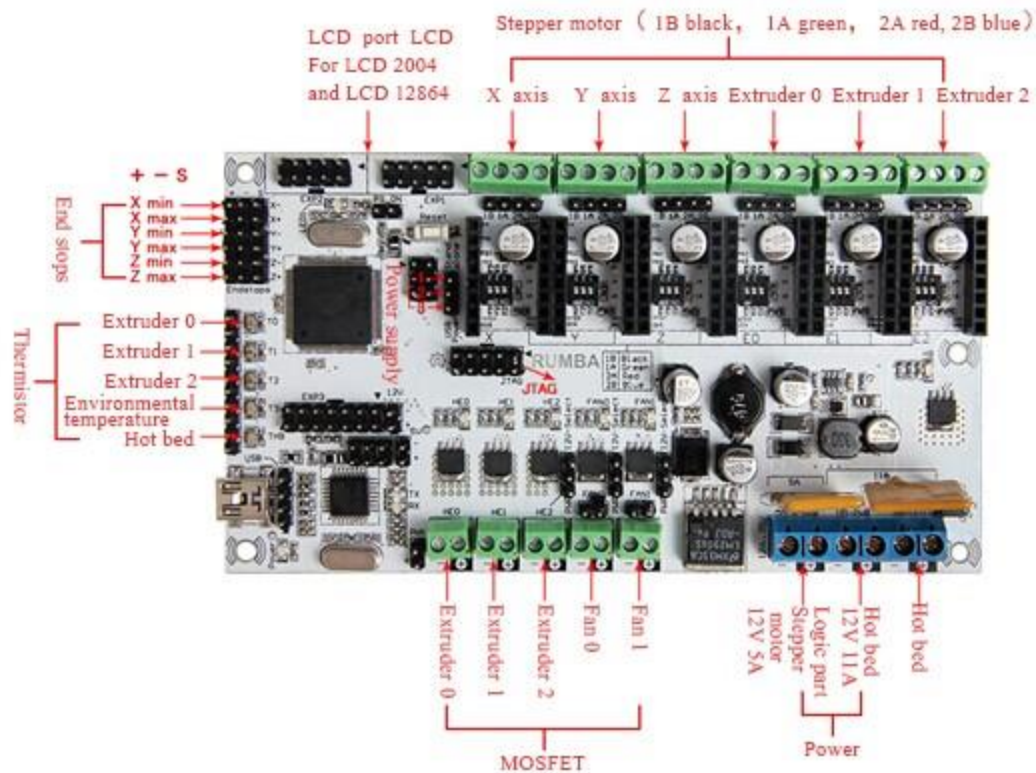


Figure 2.3: Rumba Control Board

The RUMBA board was decided to be our best option for this printer. It is the only board available that utilizes six stepper motor drivers. In addition, the stepper drivers this board uses are the powerful DRV8825s, which have a 2.5 amp limit. This would allow us to plug more than one stepper motor into each driver, which is useful if you want to print two parts at once.

## 2.1.4 Drivers

The stepper motors of a 3D printer need to be precisely controlled to produce a good quality print. One of the components responsible for this function is the stepper motor driver. A stepper motor driver is a chip that controls the power flow to the stepper motor so that the motor can be precisely positioned. The drivers receive a signal from the microcontroller that specifies how much the stepper motor needs to move. Typically, the microcontroller does not supply the power to the motor directly, since the microcontroller cannot provide enough power to drive the

stepper motor. Using a driver, the microcontroller can control the speed and position of the stepper motors while powering the motors directly from the power supply.

Motor drivers are used in many applications beyond 3D printing. Some applications include: relay and solenoid switching, LED and incandescent displays, automotive applications, audio-visual equipment, PC peripherals, car audio, and car navigation systems [7]. Motor drivers are typically available in the form of integrated circuits (ICs). Electronically, motor drivers act as current amplifiers since they take a small signal from the controller and output the same signal at a higher current. This system allows motors to operate at a high current that would otherwise damage the control board.

Stepper motor drivers rotate the stepper motors in steps, which are a portion of the rotation of the motor. The number of steps depends on the stepper motor, and the higher the number of steps, the more precisely the motor can be controlled. Stepper motor drivers have an additional benefit in that they can provide fractional steps [15]. The drivers commonly run with 16 microsteps, or 1/16 fractions of a step, for a 3D printer [8]. Running fractional steps results in smoother motion in the stepper motor and allows for much finer control of the position of the motor. The number of microsteps is one of the considerations to make when selecting the stepper motor drivers. Another important consideration is the current limit of the stepper motor driver. The current limit of the driver should be higher than the maximum current draw of the stepper motor, with a reasonable factor of safety. Exceeding the current limit of the stepper motor driver risks burning out the device.

### 2.1.5 Endstops

Endstops are the components responsible for ensuring that the print head stops moving in a given direction when it has reached the end of the 3D printers rail system. This sort of sensing is easily achieved either mechanically, magnetically, or optically depending on the

application. 3D printers like the one featured in this report typically use either optical or mechanical switches, since they are more accurate at lower prices than the magnetic options.

Optical endstops use both infrared or other forms of electromagnetic radiation and an optical sensor to determine the distance between themselves and the objects immediately in the path of their emitted light. These switches have the advantage of not using moving parts. This allows optical endstops to operate for much longer than their mechanical counterparts without needing maintenance. Unfortunately, they are often found less accurate and more difficult to configure since they rely on a very fast moving medium to measure small distances.

Mechanical endstops can achieve very accurate results with a high degree of repeatability [13]. This is because they rely on the physical collision of two objects in order to allow an electron flow, just as a light switch would, except with much smaller forces and movements needed in order signal the control board. Mechanical endstops can be physically worn out, and these switches must be maintained on occasion. However, mechanical switches can be purchased easily which will last over a hundred thousand cycles, making their replacement infrequent even when a printer is being heavily used.

## 2.1.6 Firmware

The firmware on a 3D printer is the programming within the control board on the printer. The primary function of the firmware is to interpret the G-code commands and control the movement of the stepper motors based on these commands. The G-code is a list of movement commands, created by a slicer program based on the CAD model that the printer needs to perform to create the final part. These commands are uploaded from a computer to the control board, where the firmware reads the code and sends signals to the stepper motor drivers, which then properly move the print head to create the final part. Firmware needs to be configured specifically for each 3D printer to operate properly. There are many types of firmware available for use in a 3D printer. Table 2.2 outlines some popular types of firmware used in 3D printers.

We selected the Marlin firmware given its popularity and extensive support [12] [9]. Sprinter and Sailfish are also popular firmware choices. Teacup is an easy-to-use firmware designed for beginners. Each firmware has different features available and is compatible with a selection of control boards. The firmware for the printer should be selected based on the required features, compatibility with the selected control board, and the amount of support available.

Firmware Name	Features	Compatible Controllers
Marlin	High step rate, interrupt-based temperature protection and movement, full endstop support, SD card support, LCD support, Arc support, temperature oversampling, Dynamic Temperature setpointing (AutoTemp), heat power reporting	<ul style="list-style-type: none"> <li>• RAMPS</li> <li>• RAMBo</li> <li>• RUMBA</li> <li>• Sanguinololu</li> <li>• Ultimaker's v 1.0-1.5</li> <li>• Gen 6 and Gen 7</li> <li>• Duemilanove 328P</li> <li>• Melzi</li> <li>• Azteeg X1, X3</li> <li>• Megatronics</li> </ul>
Sprinter	SD card reader, stepper extruder, extruder speed control, movement speed control, constant or exponential acceleration, heated build platform	<ul style="list-style-type: none"> <li>• RAMPS</li> <li>• Sanguinololu</li> <li>• Teensylu</li> <li>• Ultimaker's v 1.0-1.5</li> <li>• Gen 6</li> </ul>
Teacup	Moves steppers smoothly, start-stop ramping, lookahead, printing from SD card, unlimited number of heaters, devices, and temperature sensors	<ul style="list-style-type: none"> <li>• 8-, 32-, and 64-bit controllers</li> <li>• Most ATmega boards</li> <li>• Teensy3, HBox, and NXP LPC1114 based controllers</li> </ul>
sjfw	High speed G-code pipeline, LCD/keypad control panel, acceleration, SD card, full runtime config	<ul style="list-style-type: none"> <li>• RAMPS 1.2/1.3</li> <li>• Gen 4</li> <li>• Sanguinololu</li> </ul>
Sailfish	Acceleration support, dual extrusion (including "ditto printing"), high step rate, heated build platform support, safety cutoff support, interrupt based movement and temperature protection, PID based temperature control, full endstop support, SD card support, LCD interface	<ul style="list-style-type: none"> <li>• Gen 3</li> <li>• Gen 4</li> <li>• MightyBoard</li> </ul>

Grbl	Simple controller for CNC milling, written in C, does not require parallel port, high step rate, acceleration management	<ul style="list-style-type: none"> <li>• Arduino Uno</li> </ul>
Repetier - Firmware	RAMP acceleration support, path planning, fast ooze prevent, trajectory smoothing, nozzle pressure control, multiple extruder support, continuous temperature monitoring, PID control for extruder temperature, SD card support LCD support	<ul style="list-style-type: none"> <li>• RAMPS</li> <li>• RADDs</li> <li>• Sanguinolou</li> <li>• Gen 6 and Gen 7</li> <li>• SMART RAMPS</li> <li>• Melzi</li> <li>• Printboard</li> </ul>

Table 2.2: Available 3D Printing Firmware [12]

### 2.1.7 Thermistors

An inherent part of a 3D printing operation is the controlled heating of extruders to melt the print medium. This necessitates temperature feedback from the print head to the control board. This is done using a temperature variable resistor, or “thermistor”. Merriam Webster Online defines a thermistor as: “an electrical resistor making use of a semiconductor whose resistance varies sharply in a known manner with the temperature”. Because this change in resistance is electronically detectable, thermistors properties allow the control board to calculate the temperature of the extruders.

A thermistors key component is its high quality semiconductor. Typically comprised of ceramics containing metal oxides, these semiconductors exhibit a relatively large range of conductivities based on their temperature. They achieve this effect because highly oxidized metals can change greatly in the energy needed for them to transmit a flow of electrons. The higher the temperature of the semiconductor, the less energy is required move electrons, since the entire material is closer to the activation energy needed to move electrons in a polar direction. Understanding this property allows the manufacture of specific behaviors in thermistors by managing the chemical makeup of the material to include more or less oxides. According to Encyclopedia Britannica, this leads to more or less replacement ions in the crystal



structure, allowing us to manufacture thermistors with a wide range of resistances and temperatures.

3D printers such as the one featured in this report typically use thermistors whose resistances and temperatures are based on the melting points of the desired print medium and the accepted inputs of the control board respectively. This allows the thermistor to be calibrated by exposing the thermistor (after installation) to ranges of temperatures and recording the resistances reported by the control board. Once sufficient tests have been performed and analyzed, the thermistor is ready to be used.

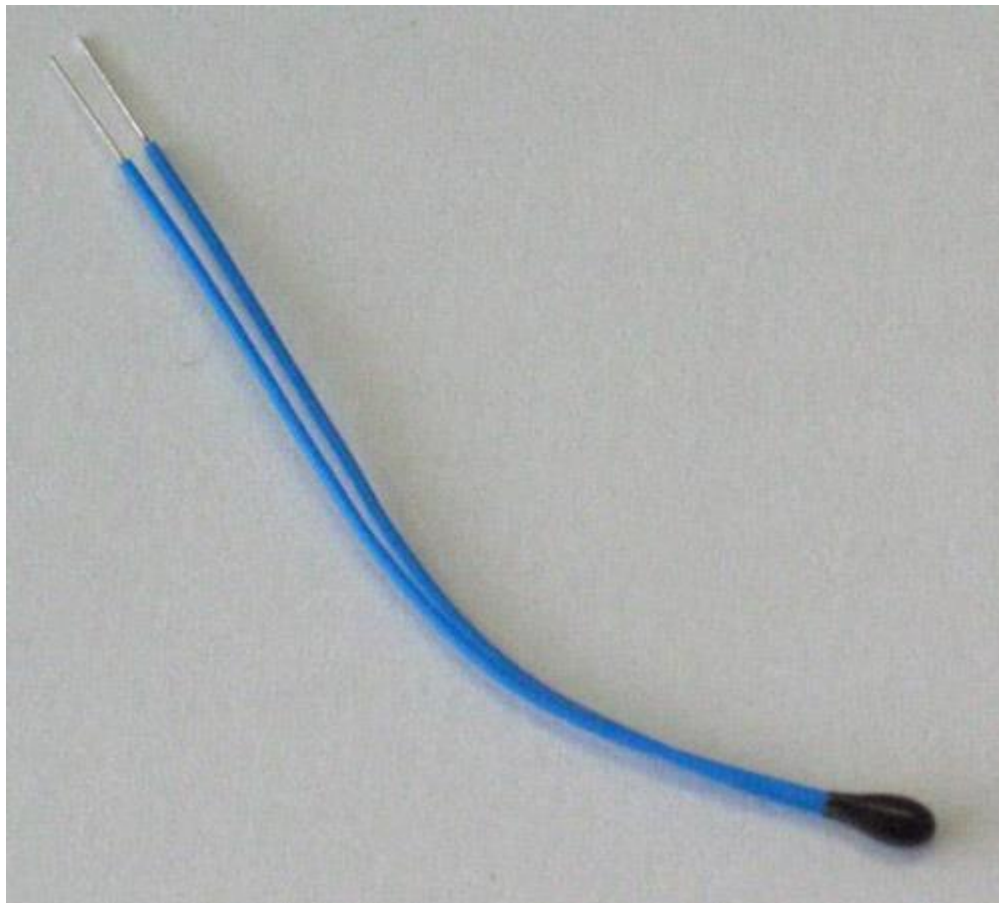


Figure 2.4: Thermistor [6]

Figure 2.4 shows a generic thermistor before installation. The two blue leads allow the control board to monitor the resistance, while the actual semiconductor is housed in the black

rubber head. Installation requires a specialized conductive glue or epoxy to allow heat to transfer from the extruder tip to the thermistor.

### 2.1.8 Printing Software

There are many computer programs that assist people with 3D printing. They can be broken down into five types:

1. Programs that create virtual models.
2. Programs that manipulate and prep models for printing.
3. Programs that create printer instructions (G-Code) to create the models.
4. Programs that control the 3D printer.
5. Programs that interpret the G-Code and send instructions to the stepper motors.

Each one has its purpose to help bring an idea into the real world.

Some programs may try to combine two or three of these types into one to make it easier on the user. A user might find themselves using multiple programs of the same type. This is due to the limited feature of each program. For example, the program “Pronterface” is a control program that works from a computer, but is unable operate wirelessly. Someone might load “OctoPi” onto a Raspberry Pi in order to free their computer from the printer and operate the printer wirelessly through a local area network. This gives the user the option to print wirelessly from home, but also to print wired if they are disconnected from their network.

## 2.2 Arduinos

An Arduino is an open-source electronics board designed for simple prototyping [1]. Using the free Arduino IDE, programs are written in C/C++ and then uploaded onto an on-board microprocessor. This chip allows the use of multiple input output pins located on the outside of the board.

## 2.3 Multiple Extruder Printing

One important consideration when designing a 3D printer is how many extruders the printer will have. Adding multiple extruders to a printer offers several benefits. Multiple extruders is one way to provide multi-color prints. While there are other methods for printing in color, there is no color bleed when transitioning between different colors since each color has its own nozzle. Additionally, multiple extruders allow for multiple materials to be used in the same print. For example, one extruder can print support material for a complex part while the other extruder prints the part itself. Each material uses its own extruder, which can be independently temperature control for the specific material. Finally, having multiple extruders allows for continuous printing using multiple filaments. The typical benefit for this is that the print job does not need to stop to switch between colors or switching between the component material and the support material. An additional benefit is that, in a more spread-out configuration, multiple extruders can be used for “ditto printing”, or printing multiple copies of the same part at the same time.

## 3.0 Methodology

This portion of the report is dedicated to the actual methods used to assemble, refine and run the 3D printer. It is organized to present the information in the most functional order, and is not representative of the chronological order of construction.

### 3.1 Motors and Controls Design

Initially, we spent our time searching for the components that would work with the requested features. We had multiple changes in the design as extra features were added. In the end, we found components that allowed us to print multiple large parts at once, utilize multiple extruders, with one extruder capable of printing in RGB color.

Our biggest challenge in the beginning was finding parts that would allow us to utilize more than two extruders. This limit was due to the amount of stepper drivers available control boards were able to handle. We had to ensure that there was a driver for all three axes, and enough for at least two extruder motors, which is what would give the printer the ability to print in color. For multiple weeks, we were limited to five stepper driver ports, which limited us two colors. After getting more than halfway into the term, we discovered the RUMBA control board, which allows the use of six stepper drivers. The board allows us to print more than two colors and permits us to print RGB color.

By choosing this board, we did not have many options when selecting stepper drivers. The recommended stepper driver was the DRV8825 which is made by Texas Instruments. The DRV8825 stepper driver has a 2.5 amp limit according to its specifications. This limit happens to be fairly high when compared with other stepper drivers.

Stepper motors were selected based on the current requirements. Since a few of our stepper motors were going to be wired together and share the same stepper driver, we had to

make sure they did not exceed the 2.5amp limit. We ended up purchasing a combination of 2 amp, 1.2 amp, and 0.4 amp stepper motors.

### 3.1.1 Design Considerations

Designing any complex device requires a considered discussion between its inventor and those responsible for hands on construction. Even innovating and re-iterating an existing device for new applications requires a back and forth between the design team and their management to determine the features and functions which are both desirable and manufacturable. This 3D printer is no exception, and the following is a description of the final goals for this project and the discussion which lead us to their finalization.

In broad terms, our intention was to create a larger and more advanced 3D printer than those currently found on campus, for use by faculty and students. The initial design served this purpose by having as many as six simultaneous print heads on a large bed to allow for rapid mass production of plastic parts. However, searching for the relevant components quickly showed that the control boards available would not support the quantity of stepper motors needed for so many simultaneous printing operations. With the inclusion of the axis motors, we were forced (temporarily) to settle for only one print head. To offset this disadvantage, we determined to include a multiple color functionality and a heated print bed for higher quality parts.

The team was able to generate and report several methods of multiple color printing through a single extruder. The most common method revealed was to halt the machine and manually thread a new color of filament into the extruder whenever a color change was necessary. A more complex iteration of the same method is also sometimes used, wherein the print head is coded to depress a physical switch on the inside surface of the printer, making the operator's job less intensive and saving a small amount of time for each color change. These methods are time consuming but would definitely work. Unfortunately, for a display printer these

method would be both slow and not terribly imposing. Ultimately, our decision for that time was to research a product called the “Pallet”, which is a recent innovation in 3D printing. The Pallet works by splicing different colored filaments into a single strand which is then extruded all at once, producing good quality prints.

Our next major advancement was the discovery of a more specialized control board known as the RUMBA. More specifically, it allowed us to integrated more than five stepper motor drivers. This grants us to control additional extruders and two separate print heads, allowing us to print in two different materials or to print two parts simultaneously. Detailed technical specifications for the board may be found under section 2.1.3 of the background. The increased number of motors allowed us to use a specialized print head which achieves a variety of colors by mixing different tones of plastic through three combined extruders. Since we no longer required the Pallet, we were able to save room in our budget while still meeting the design specifications required by our advisor.

Once the team had a plan to accomplish our overarching goals, we were left with the questions of component placement, cable and wire management, and modularity. To allow for ease of updating or part replacement, we determined that snap together and screw together cables would be preferred wherever possible. Initially, the intention was to create a slide out drawer in the bottom of the printer to house as many of the electronics as possible, further increasing ease of access and modularity. However, on further consideration we determined that by mounting the electronics on the side of the printer we would be able to improve heat flow by separating the sensitive components from the heated print bed. It also displayed our components better and allowed for easy maintenance.

### 3.1.2 Wire Selection

The numerous components of the printer each have different power requirements and, therefore, require different sized wire. Selecting appropriately sized wire is important to the

operation of the printer. Wire that is too thin can cause a voltage drop from the power supply to the components, which can interfere with the proper operation of the printer. It can also become too hot, which can damage the wire or other components. We have chosen to use three different sized wires for the electronics. The largest size, 12 AWG, is used to supply power to the heat beds, as they draw the most power. The next largest wire size is 14 AWG and is used to supply DC power to the RUMBA board and supply AC power to the power supplies. Finally, 22 AWG wire is used to supply the components from the RUMBA board. In addition to selecting the wire size, the wires are color coded to more easily tell what each wire connects to. Table 3.1 below outlines wire size and color for each component.

<b>Component</b>	<b>Wire Size</b>	<b>Wire Color</b>
AC Power Hot	14 AWG	Black
AC Power Neutral	14 AWG	White
AC Power Ground	14 AWG	Green
RUMBA DC Power (V+)	14 AWG	Yellow
RUMBA DC Ground (COM)	14 AWG	Black
Heat Beds Power (V+)	12 AWG	Red
Heat Beds Ground (COM)	12 AWG	Black
Stepper Motors	22 AWG	Black, Green, Red, Blue
Endstops	22 AWG	Red, Black, Green
Thermistors	22 AWG	Green
Extruder Hotends	22 AWG	White
Extruder Cooling Fans Power	22 AWG	Red
Extruder Cooling Fans Ground	22 AWG	Black

Table 3.1: Wire Sizing and Coloring

## 3.2 Component Justifications

This section contains the exact components which we ordered, along with the attributes by which we selected them. Certain parts were excluded from this section as being trivial, but the elements which majorly affect printer performance and budgeting have all been detailed. The team had several overarching needs to consider, each having to be balanced against the size of the budget. Good reliability and high functionality were perhaps the most critical characteristics, since all of the parts must perform to our standards without any failure to endure. Of secondary importance was availability. In the event of a hardware failure, the team wanted to be able to replace any part with relative ease to ensure that progress is never halted for more than a few days at most. It stands to reason that we chose not to use any parts which could not be ordered and shipped on short notice, or any parts which were likely to become unavailable in the near future. Any unfamiliarity with the parts mentioned in this section can be assuaged by reading the corresponding portions of the background.

### 3.2.1 Stepper Motors

For head movement and medium extrusion, the team chose to use Nema 17 stepper motors. These motors were designed specifically for use in 3D printers, and have received good reviews both by academic peers and online. For these reasons, it was easy to assume good reliability and consistent availability. There are slightly higher quality competitors available, but the improvement in performance appears to be marginal and the prices are also much higher.



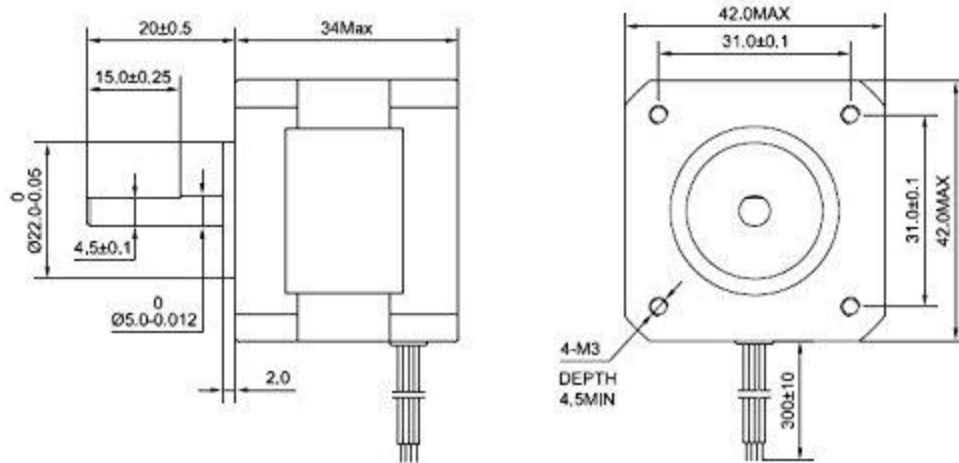


Figure 3.1: Stepper Motor Dimensions

The included image (Figure 3.1) made it possible to check the exact dimensions of the motors to ensure compatibility with the rest of the 3D printer design. Having determined that these motors are durable, compatible, and economical the team added them to the initial parts order.

### 3.2.2 Power Supplies

As designed, this 3D printer requires well over 1000W of power in order to function. In light of this, the team decided to use a combination of three AC to DC power converters working in concert. This makes sense for several reasons. Firstly, power supplies are often much more expensive once they exceed commonly demanded watt values. For example, a 1500W power supply is likely to be more than double a 750W converter, despite being only twice as powerful. Secondly, having three devices allows for easier power distribution and mechanical redundancy. The selected components are shown here in Figures 3.2.2a and 3.2.2b.



Figure 3.2: 600 Watt Mean Well Power Supply



Figure 3.3: 360 Watt Power Supply

Figure 3.2 shows a Mean Well 600W, 12V DC output power supply, while Figure 3.3 shows the EPBOWPT 360W, 12V DC output power supply. Due to the team's lack of prior experience in this area, these parts were ordered based on maximizing the apparent reliability

and minimizing the total cost using reviews and guarantees from the suppliers and unbiased users.

### 3.2.3 Control Board/CPU

The selection of the control board was perhaps the most critical, since so much of the 3D printer's functionality relies upon this single component. In order to control the additional stepper motors needed for the double print head feature, the team required a board which could support an above average number of drivers, in addition to meeting the reliability and availability factors. Budgeting played a relatively minor role in our selection process, both because of the eclectic specifications and its show-case nature. The chosen control board is the Rumba Control Board from RepRap. Figure 3.4 clearly shows the additional spaces for drivers, as well as other highly desirable features such as the boards compact nature and aesthetic appeal. All research indicates that the Rumba will easily live up to the team's expectations and needs.

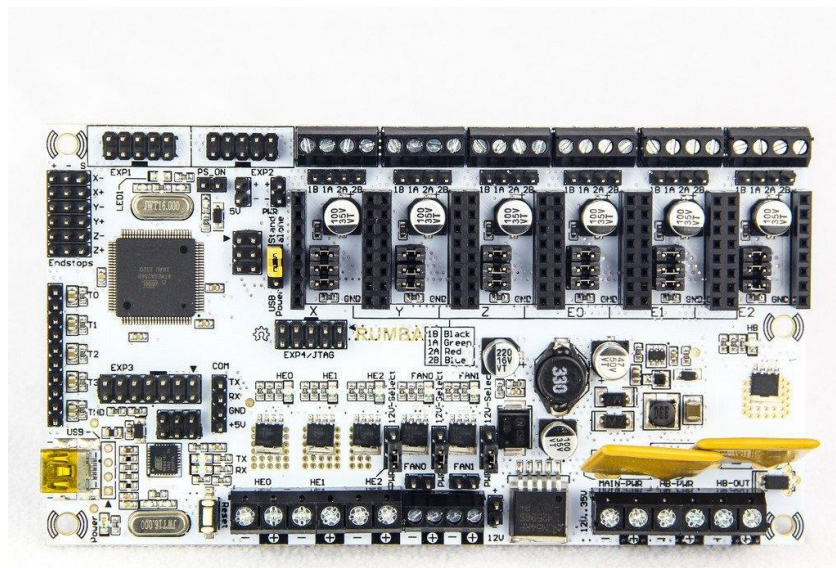


Figure 3.4: RepRap Rumba Control Board

### 3.2.4 Heat Bed and Temperature Controls

Due to the power requirements of the heat beds, each one requires its own 600 Watt power supply. In order to control them, we used an Arduino Uno loaded with our own temperature regulating firmware to control and maintain the temperature set point and display the real-time temperature of the heat beds. The Arduino powers four 30 A relays, two for each bed, to control the power to the heat beds. Additional information about the Arduino control circuit can be found in section 3.5.1. The system is controlled through an LCD screen and keypad, which allows the user to set and monitor the temperature of the heat beds. Additionally, each heat bed has a power switch, so that one or both of the heaters can be turned off if it is not needed.

### 3.2.5 Non-Critical Elements

In addition to the main attractions featured in sections 3.2.1 through 3.2.4, there were a host of smaller parts needed. Thermistors, end stops, relays, drivers, and countless other non-critical elements have been included in our initial orders. As described in the introduction of the methodology, these components needed to be economical, readily available, and reliable. Fortunately, most of these smaller components have been manufactured for decades for use in other applications, making them inexpensive and easy to find. Rather than providing a detailed report on such trivial components, this document allows section 3.2.5 and 3.2.6 to speak for themselves, since the sources in the orders listed can be followed for further information if needed.

### 3.2.6 Initial Part Orders

This section contains tables detailing our first two parts orders. See Tables 3.2 and 3.3.

Part	Location	Quantity	Total Price
Power Supply 600 W	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	2	160
Power Supply 360 W	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	19
Controller Board	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	113
Drivers	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	2	24
Endstops	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	9
2A Stepper Motors	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	4	52
.4A Stepper Motors	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	5	65
1.2A Stepper Motors	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	2	26
Power Cord	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	\$8
Power Switch	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	6
PC Power Connectors	<a href="https://www.amazon.com/">https://www.amazon.com/</a>	1	5
		Total	\$487

[illegible]

As evident from the figures, the first order placed was sourced entirely from Amazon.com, while the second order included parts from McMaster-Carr and Professor Sullivan as well. Any additional components were purchased locally by team members.

### 3.3 Exploratory Assembly and Basic Testing

Any project of meaningful size requires a high degree of organization in order to run smoothly. The assembly of this 3D printer is no exception. In order to maximize the efficiency of our motors and controls assembly and basic testing, the team decided to order the minimum necessities first. This strategy makes sense for several reasons. Firstly, it allows us to fulfill the basic design specification as soon as possible, relieving us of anxiety about the gross functionality of the project. Secondly, this starting point offers significantly reduced financial risk in the event that our first estimation of the components needed is in some way faulty.

Having decided this, the team ordered and received the control board, the stepper motors, and the drivers and sensors needed to connect and test the basic capabilities. Once the parts were in hand, the group was able to finalize the physical layout of the controls. After arranging for wiring and installing the software, the floor was cleared for assembly to commence.

For ease of transportation, storage, and cable management the decision was made to mount all stationary parts onto two sheets of heat resistance PVC plastic. Once these had been purchased, it was possible to drill mounting holes based on our schematic. It was decided that initially we need only mount the power supplies, since the other components have varying length of wires and may need to be accessed from the underside. Once the power supplies were mounted, all other components could be temporarily wired for testing. Figures 3.5, 3.6, and 3.7 show the groups working together on these tests.



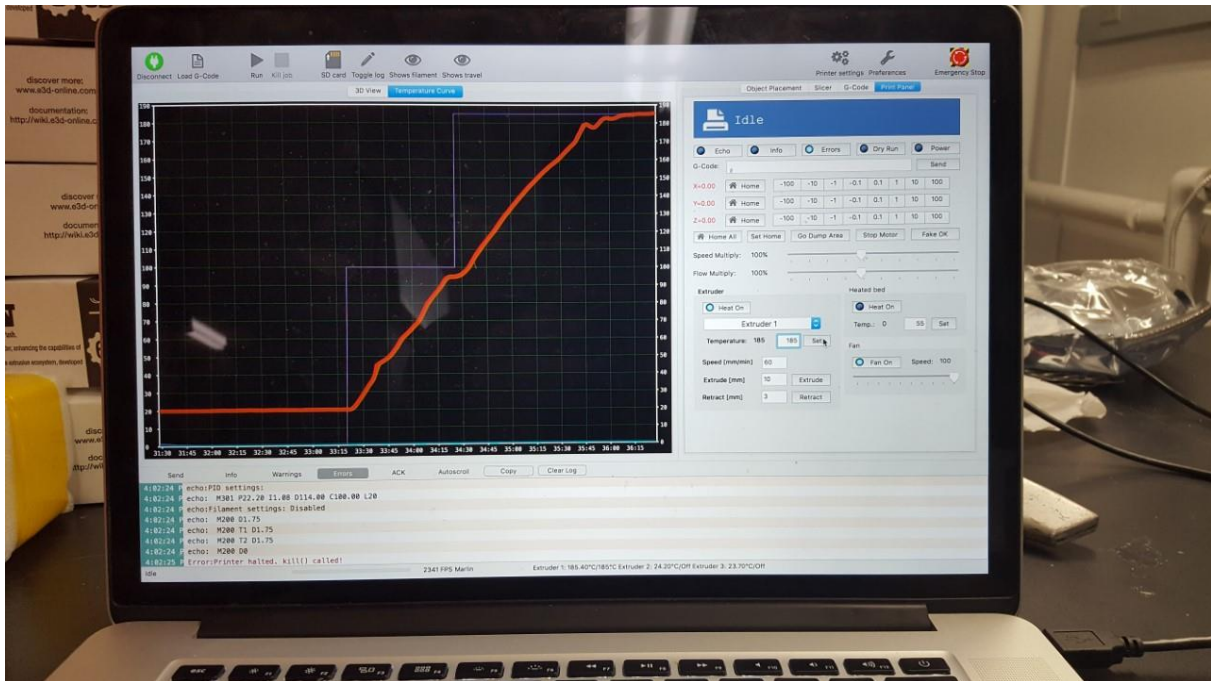


Figure 3.5: Exploratory Testing Software View

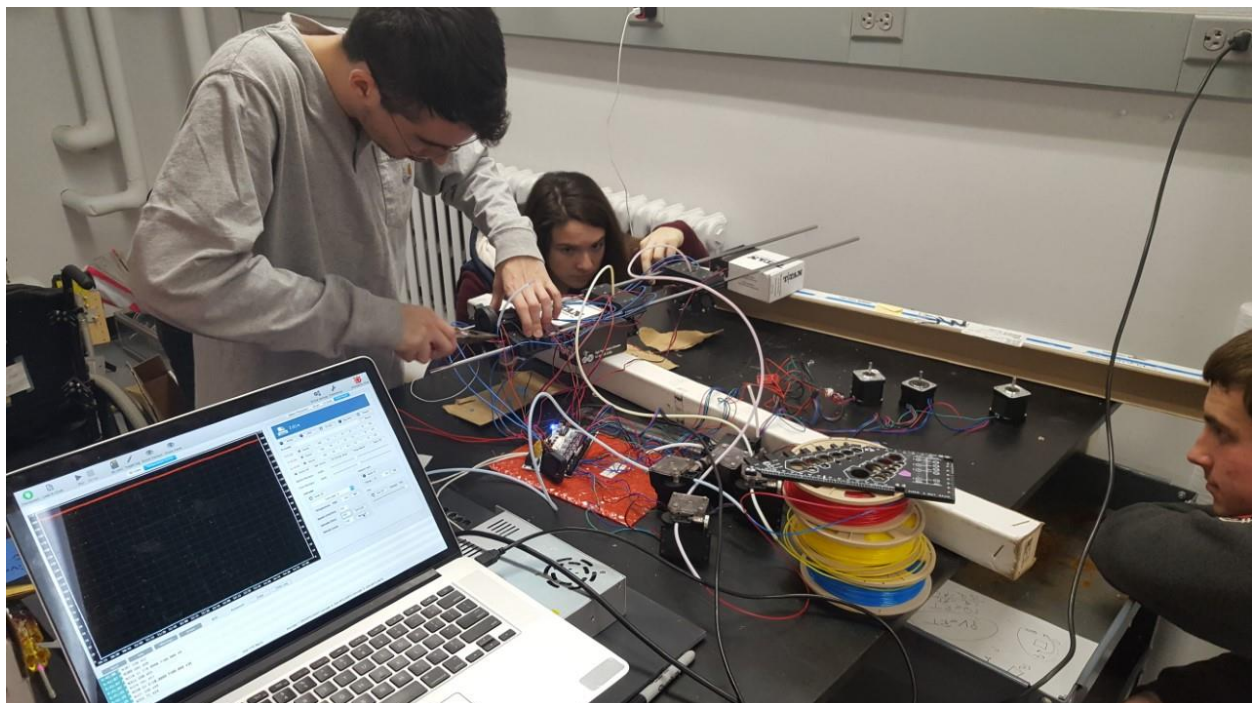


Figure 3.6: Stepper Motor Driver Testing

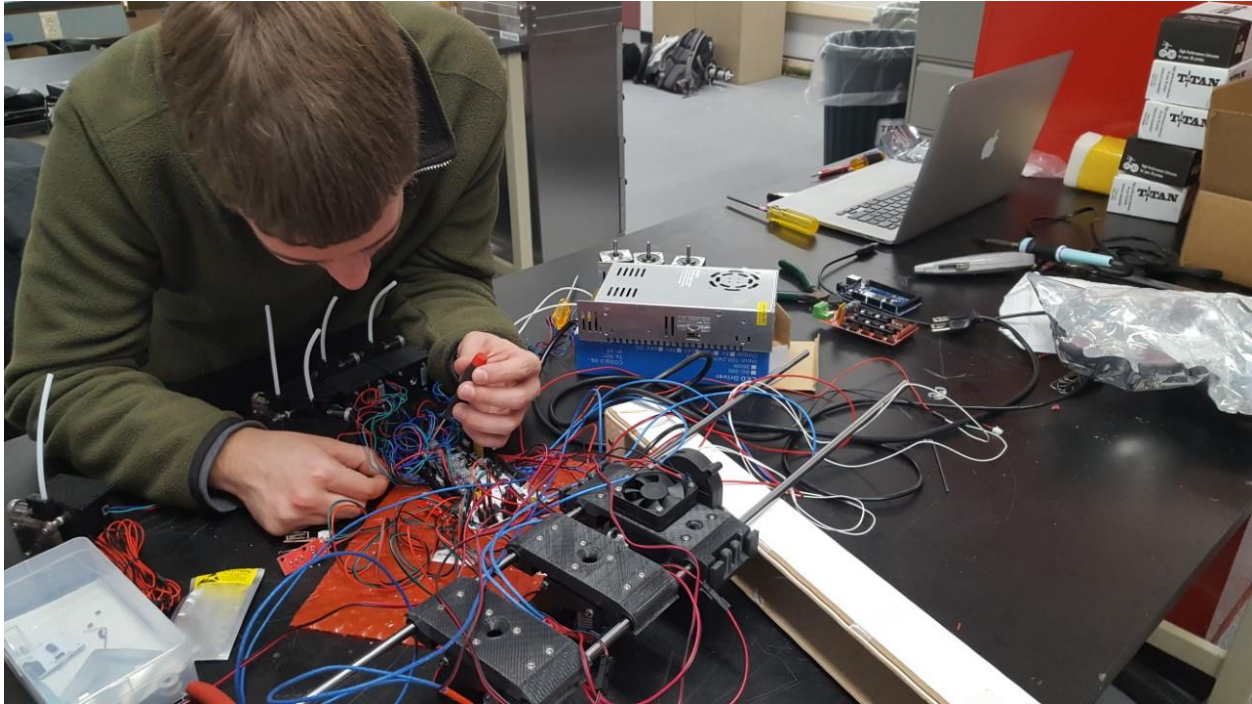


Figure 3.7: Exploratory Wiring

Working with the Print Head and Platform groups, the team was able to test the extruders and the hot ends, in addition to the motors, control boards and sensors needed for the 3D printer Motors and Controls MQP. At this point in construction, the team was confident that all necessary elements were working smoothly and as anticipated. Once all three groups were prepared to proceed, the Motors and Controls team was able to integrate its now functioning system into the full 3D printer.

## 3.4 Final Assembly and Programming

This section details the assembly of our components, their assimilation with the complete device, and the construction of the software logic that controls the printer.

### 3.4.1 Sideboard and Mounting

Since the 3D printer motors and controls group was working in parallel with the 3D printer platform group, the bulk of our components need to be mounted to a surface extraneous to the main platform. This method allows us to assemble, wire, and test the majority of the



functionality at our leisure. The controls must also be able to survive the ranges of heat produced by the heated beds mounted in the floor of the 3D printer. The front of the printer must open, the roof of the printer is needed for print head movement. The sides are the only remaining face which can be easily manipulated from the front.

The team decided to use heat resistant polyvinyl chloride (PVC) sheeting as the sideboard material. This decision is driven by the knowledge that PVC is not as easily shattered as glass or Acrylic, and is meaningfully more economical than either alternative. The final selling point for PVC is our ability to order it through the schools approved supplier list.

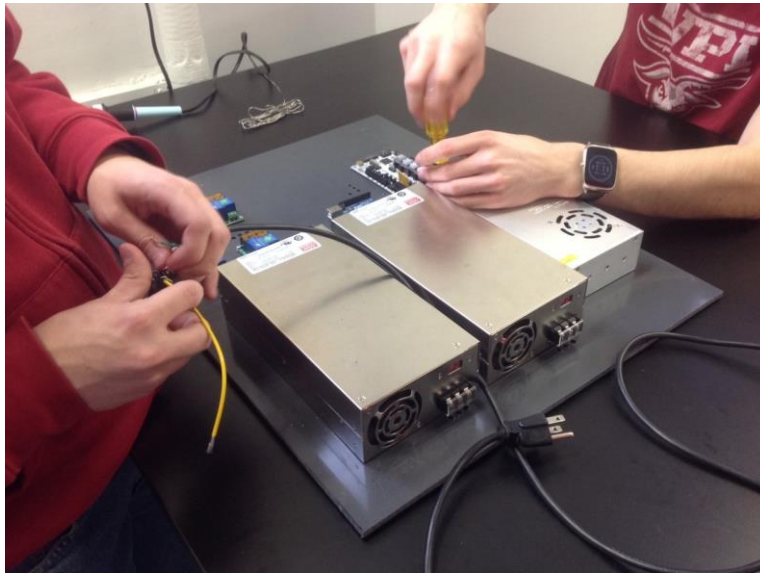


Figure 3.8: Sideboard Mounting

Having received the sideboard and the components needed for wiring, we were able to map our intended layout by setting the components on top of the PVC and arranging them to our needs. The final layout places the power supplies at one end and the controllers and drivers across the long sides and the center. The exact placement and orientation is a function of circuit proximity and component similarity to facilitate wiring and maintenance. This execution allows the printer to be efficiently wired wherever possible while maintaining visual organization.

### 3.4.2 Wiring and Wire Management

The many electronic components on the 3D printer require many lengths of wire in order to supply both power and communication between components. If not properly managed, the amount of wire required could quickly become a mess and make it difficult to track which wire runs to which component. Wire management is also a safety concern, as connections must be properly protected to protect from shocks, and tangled wire could overheat. For our design, most of the wires run behind the panel to which the electronics are mounted. This design makes it easier to manage the wires and makes the electronics panel more aesthetic. The wires run up through holes in the panel to connect with their respective components. On the reverse side of the panel, the wires are tied together and to the board using zip ties. The zip ties are attached to wire guides stuck on the back side of the panel.

Another important part of wire management is ensuring good connections when connecting wires to each other or components. A good connection has both a solid contact between conductors and an insulator acting as protection. Good contact is important to achieve proper power flow, since poor contact creates a point of high resistance. The connection also needs to be covered by an insulator for protection. This is important both to protect the component from possible short circuit and to protect a person from injury due to shock. The most important connections to protect are the connections to the power supply, as these wires carry high current and can injure someone who touches an exposed connection. To solve this issue, terminal connectors are crimped and soldered to the ends of the wires that connect to the power supplies. The power supplies, as well as the relays, are also equipped with a protective plastic shield over their terminals. The other critical connections are the connections with the power switch and the PC power cable connector. These connections were made using insulated terminal quick connectors, which are designed to provide a safe and reliable connection.

### 3.4.3 Control Panel Assembly

The control panel is the interface through which the user is able to control our system. It was mounted to a box attached to the front of the printer on the upper left corner of the frame. The most prominent features on the control panel are the two LCD screens. The screen on the left controls the firmware on the control board. It features a knob that can be used to run different features of the firmware, such as hotends or cooling fans, as well as an emergency stop button. The LCD on the right displays information for the heat beds and is controlled using the keypad mounted below the screen. There are also several switches mounted to the control panel. The large, red switch controls main power to the printer. Shutting off this switch will power down all systems on the printer. Two switches labeled “EX1” and “EX2” control power to the side extruders and can be used to shut off the extruders if they are not being used. The switches labeled “HB1” and “HB2” control power to the heat beds and can be used to shut off one or both of the heat beds. Finally, the button labeled “Reset” resets the Arduino program for the heat beds. This button is used if the set temperature of the heat beds needs to be changed, if there is an issue with the heat beds or LCD display, or if one wants to quickly shut off the heat beds. A picture of the final control panel is shown below in Figure 3.9.

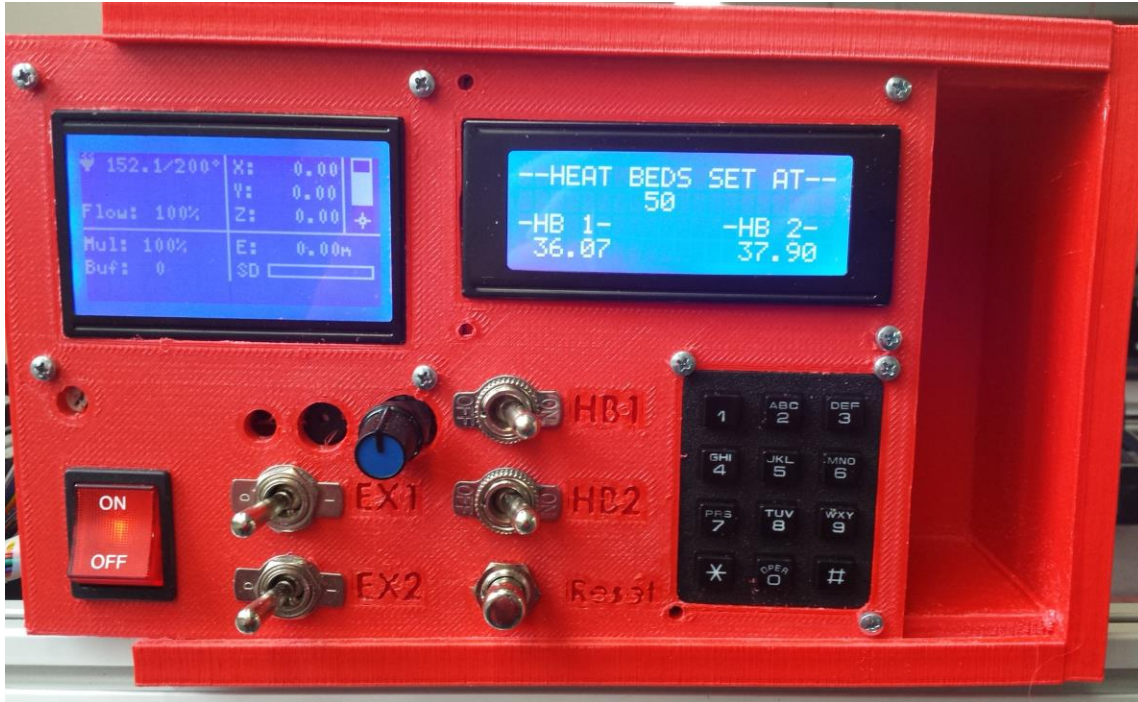


Figure 3.9: Front Panel and Front Panel Enclosure

### 3.4.4 Full 3D Printer Assimilation

One of the most important aspects of our design is its integration with the other subsections of the printer. This process has two main aspects: attaching the side panel with the electronics and connecting the wiring to the onboard components. The simpler part of this process is attaching the inner and outer panels for the electronics. The panels simply bolt onto the side of the frame to form the left wall of the printer enclosure. Since the frame of the printer is made from 8020 extruded aluminum, it is a straightforward process of inserting nuts into the channels of the extrusion and attaching the panels using bolts.

The second part of integrating our system with the printer is connecting the onboard components to our electronics board. Through collaboration with the print head group, we decided to use 20-pin and 24-pin ATX connectors to connect the wires from the stepper motors, hotends, cooling fans, and thermistors to the control board on our electronics panel. These connectors help to organize the wiring running to the various components on the printer. The power wires for the heat beds are too large to run through the ATX connectors. Therefore, each

wire for the heat beds and the heat bed thermistors are connected with quick connectors so that they can be quickly connected or disconnected if needed. The components on the front control panel are also connected to the control board and Arduino using quick connectors. Due to the use of the ATX and quick connectors, all of the electronic components on the printer can be quickly and easily disconnected if the board ever needs to be removed from the printer.

## 3.5 Detailed Designs and Schematics

This section provides detailed designs and schematics of parts of the design that are not covered well by the overall design schematic in Appendix A. One such part is the heat bed control circuit. The overall diagram shows the connections between the Arduino controller and the relays and thermistors, but excludes the additional circuit elements to simplify the diagram. The following section provides these additional details for the heat bed control design element.

### 3.5.1 Heat Bed Wiring

The heated beds of the 3D printer must be regulated to provide the desired temperature. However, the heat beds require too much current to be powered through the control board. They must be powered in a manner that isolates the controller from the current drawn by the heat beds. To solve this issue, a relay is used as the isolation device. Figure 3.10 shows the circuit used to regulate the power supplied to the heat beds. Each of the heat beds on the printer are regulated by two relays. The relays of the circuit are controlled by an Arduino, which monitors the temperature and supplies power to the heat beds when necessary. When the temperature is too low, the Arduino supplies power to the relay, which activates the internal magnet and closes the switch, supplying power to the heat bed from the power supply. Since the digital output pin of the Arduino cannot supply enough current to trigger the relay on its own, it instead powers a transistor on the Arduino 5V power rail which then powers the relay. Once the temperature reaches the desired level, the Arduino turns off the signal to the relay, which

opens the switch and cuts the power to the heat bed. The relay is wired in the “normally-open” (NO) condition, which means the heat bed only receives power when the relay is being powered by the Arduino. The diode across the magnet side of the relay provides a path for the current induced by the electromagnet when the Arduino shuts off. Without the diode, a high voltage spike would occur across the Arduino, which could damage it after prolonged use.

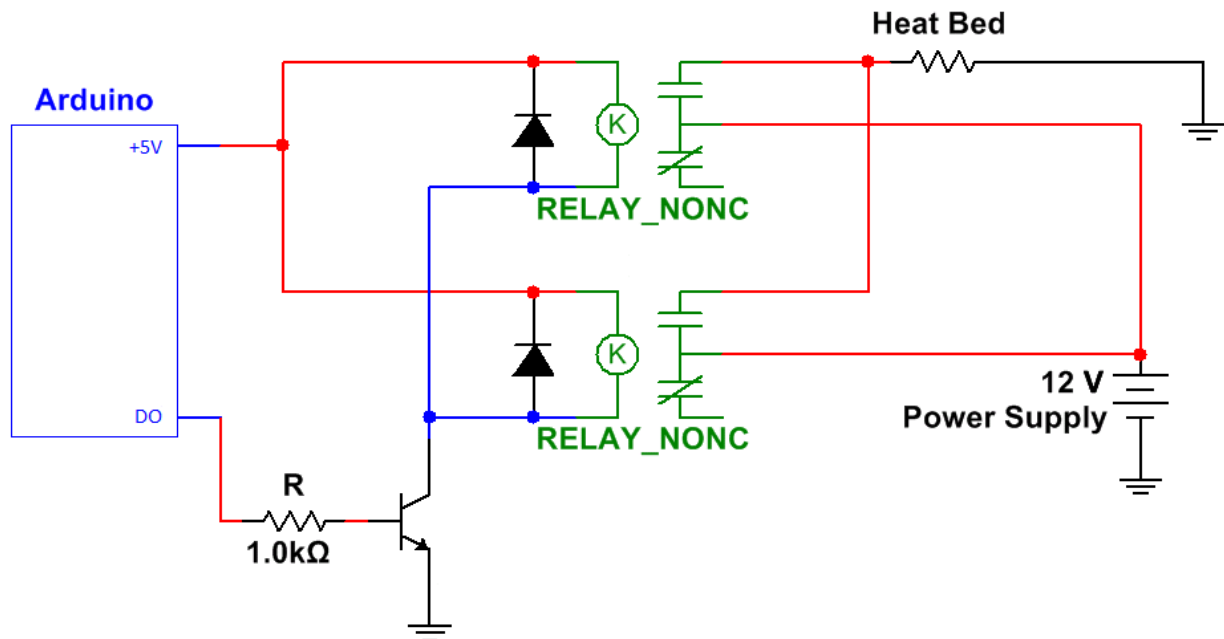


Figure 3.10: Arduino controlled temperature regulator circuit

The temperature of the heat beds is measured using thermistors, which are devices that vary in resistance depending on the surrounding temperature. However, the Arduino cannot directly measure the resistance of the thermistor. To measure the resistance of the thermistor using the Arduino, the circuit in Figure 3.11 is used. This circuit is a voltage divider, which divides the source voltage, 5V from the Arduino, between two resistors, in this case the thermistor  $T$  and a fixed resistance  $R$ , depending on their relative resistance values. The point between the thermistor and resistor  $R$  is connected to an analog in (AI) port on the Arduino, which measures the voltage at this point. This voltage is equal to the voltage across  $R$ , which has a resistance of about 51 kΩ. Since the resistance  $R$ , the voltage across  $R$ , and the source

voltage, 5V from the Arduino, are known, the resistance of the thermistor can be calculated using Ohm's Law. The resistance of the thermistor can then be used to determine the temperature of the heat bed. This calculation is done automatically through the Arduino program.

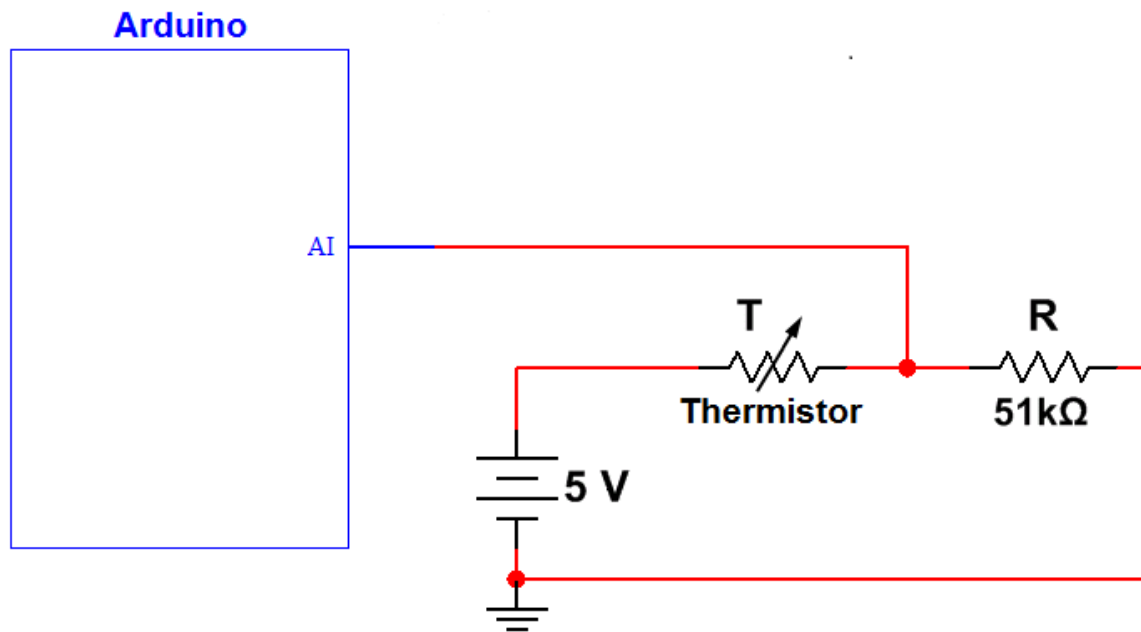


Figure 3.11: Thermistor circuit for the Arduino

### 3.5.2 Heat Bed Coding

We decided to use an Arduino board to regulate the two heat beds and allow for user interaction. An open source program was not available, so we made a program from the ground up. Features were added step by step until the full program was complete.

Our first program was tested using a basic breadboard testing circuit. A potentiometer simulated the thermistors and an LED simulated the heat beds. When the potentiometer reached a certain resistance, a small relay was triggered and the led would light up. We later added to the program to allow the use of an LCD screen.

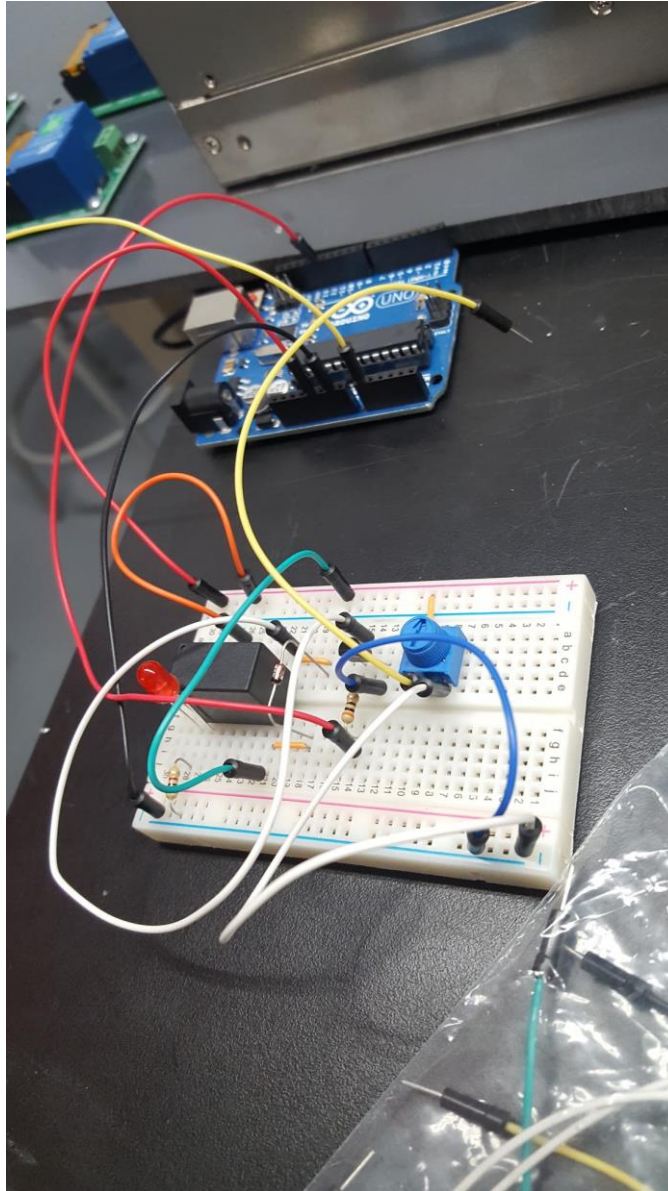


Figure 3.12: Initial Testing Circuit

The LCD screen was set up so that it would display a welcome screen before asking the user to input a desired temperature. We originally did not have a keypad to allow this, so we manually set a desired temperature within the code.



```

Serial.begin(9600); // Used to type in characters

lcd.init(); // initialize the lcd for 20 chars 4 lines

lcd.backlight(); // turns backlight on

// NOTE: Cursor Position: (CHAR, LINE) start at 0
lcd.setCursor(5, 0); //Start at character 5 on line 0
lcd.print("WORCESTER");
delay(500);
lcd.setCursor(4, 1); // Displays this once switched on:
lcd.print("POLYTECHNIC"); //
delay(500); //| WORCESTER | <-- instantly
lcd.setCursor(5, 2); //| POLYTECHNIC | <-- after half a second (500milSec)
lcd.print("INSTITUTE"); //| INSTITUTE | <-- after one second
lcd.setCursor(3, 3); //| MQP: 2016-2017 | <-- after 1.7 seconds
delay(700); //----- -- Screen stays on like this for two seconds (total time: 3.7 seconds)
lcd.print("MQP: 2016-2017"); //
delay(3000);

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("-Input Target Temp:-");
lcd.setCursor(0, 3);
lcd.print("---Then Press '#'---");
lcd.setCursor(8, 1);
lcd.print("___");
lcd.setCursor(8, 1);
int v1 = GetNumber();
TargetTemp = v1;

```

Figure 3.13: LCD Code

A keypad was later added to allow user input. In order to write the code, we observed sample code online designed for password locks. We used a similar method to get our keypad function working.

```

/*-----( Declare objects )-----*/
// set the LCD address to 0x27 for a 20 chars 4 line display (*was originally 0x20)
// Set the pins on the I2C chip used for LCD connections:
//          addr, en,rw,rs,d4,d5,d6,d7,b1,blpol
LiquidCrystal_I2C lcd(0x3F, 20, 4); // Set the LCD I2C address

//Keypad Object:
const byte ROWS = 4; //four rows
const byte COLS = 3; //four columns
//define the cymbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
byte rowPins[ROWS] = {5, 6, 7, 8}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {2, 3, 4}; //connect to the column pinouts of the keypad

// command for library for keypad
//initializes an instance of the Keypad class
Keypad keypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

```

Figure 3.14: Declaring and Initializing the Keypad

```

/*-----( Declare Functions )-----*/
int GetNumber()
{
    int num = 0;
    char key = keypad.getKey();
    while (key != '#')
    {
        switch (key)
        {
            case NO_KEY:
                break;

            case '0': case '1': case '2': case '3': case '4':
            case '5': case '6': case '7': case '8': case '9':
                lcd.print(key);
                num = num * 10 + (key - '0');
                break;

            case '*':
                num = 0;
                lcd.clear();
                break;
        }

        key = keypad.getKey();
    }

    return num;
}

```

Figure 3.15: Keypad Function Code

In the event a thermistor malfunctioned or became loose, the heat beds would heat up until destruction due to an incorrect temperature reading. In order to prevent this from happening, code was written to shut down the heat beds and throw an error message if a temperature reading went below -7 degrees Celsius.

```

//Safety turn off
if (Therm1Temp < MinTemp || Therm2Temp < MinTemp) {

    digitalWrite(Relay1Pin, LOW);
    digitalWrite(Relay2Pin, LOW);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("-----ERROR!-----");
    lcd.setCursor(0, 1);
    lcd.print("::MinTemp Reached::");
    lcd.setCursor(0, 2);
    lcd.print("-----Check your-----");
    lcd.setCursor(0, 3);
    lcd.print("----thermistors.----");

    for (int i = 0; i <= 459000; i++) { //Stops loop, flashes and holds error message for 24 hours.
        lcd.backlight();
        delay(500);
        lcd.noBacklight();
        delay(200);
    }
}

```

Figure 3.16: Heat Bed Safety Turn Off Code

### 3.5.3 Onboard LCD Wiring

The printer uses two LCD screens to interface with the controls on the RUMBA board and Arduino. For the RUMBA board, the LCD screen is connected through two six-wire ribbon cables. These ribbon wires connect to the designated LCD ports next to the stepper motor drivers. The firmware on the control board can be controlled through the LCD screen. For the Arduino Uno, four wires connect it to its respective LCD screen. A module on the back of the LCD screen then splits the four wires into the necessary pins for the screen. Additionally, a seven-wire ribbon cable connects the Arduino to the number pad. Through the LCD screen and the number pad, the target temperature for the heat beds can be set, and the real-time temperature of each heat bed can be monitored.

## 4.0 Results and Discussion

This section outlines the performance of our controls system through testing of the operation of the printer. Figure 4.1 shows a picture of the completed side panel. Overall, the controls system performed in accordance with our expectations. The electronics of the system experienced no issues. Due to our careful selection of wire and well made electrical connections, there was no excess heating in the wires, even after prolonged use of the printer in full operation. There were a few bugs in the software of the system, but they were minor issues and easily corrected. The following section describes issues we faced and how we resolved them, along with improvements made to the system. Section 4.2 displays some of the pieces made during tests on the printer.

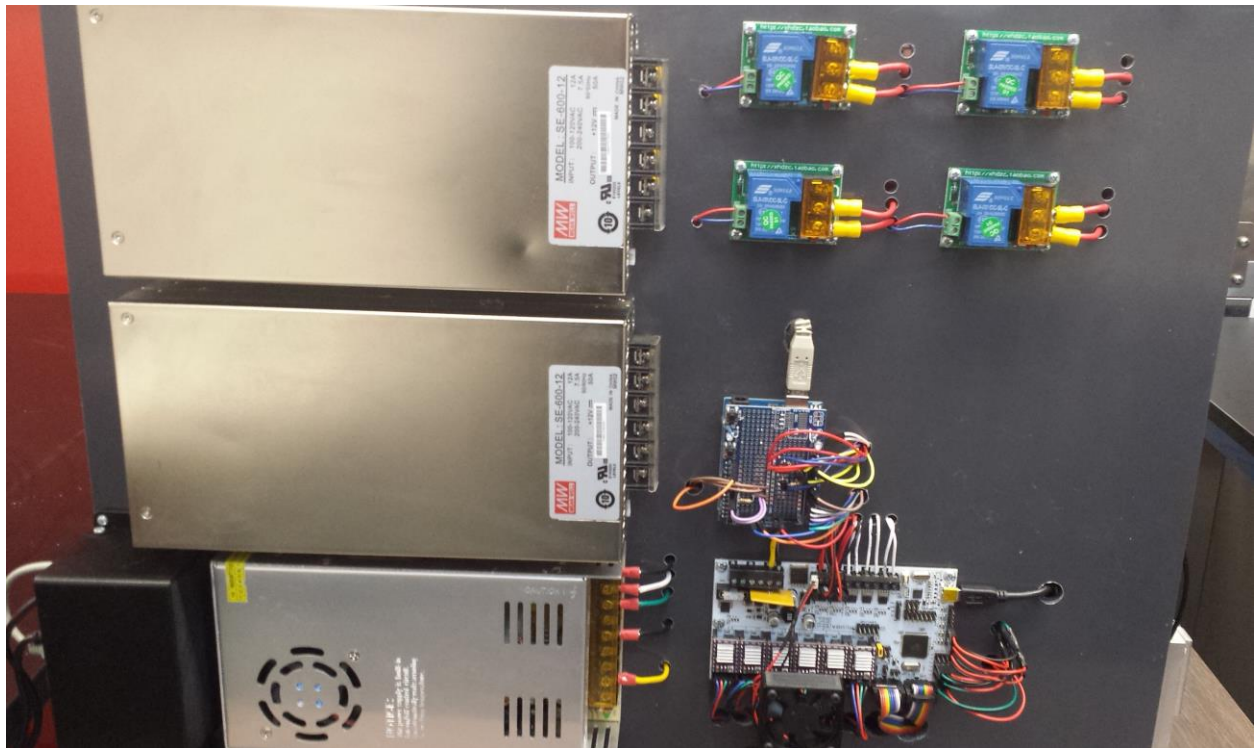


Figure 4.1: Final side panel layout

## 4.1 Design Modifications

As the project progressed, it became necessary to modify certain aspects of our design to meet the needs of the printer. These changes were implemented after testing of the printer as a whole had begun. Fortunately, only minor changes to the overall design were required. This section summarizes the design changes that were implemented into the printer.

### 4.1.1 Firmware

The largest change to the original design was the switch from Marlin firmware to Repetier-Firmware. Marlin was originally selected as the firmware for the control board due to the large amount of open source data available as well as the firmware's compatibility with both multiple extruder printing and color mixing. For multiple extruder printing, Marlin performed well in our tests. It was able to run all three hotends and print multiple parts with no issue. However, we encountered difficulty with using color mixing with Marlin. We found that Repetier-Firmware performed better when using the printer in the color mixing setup. Due to this, we decided to switch the color mixing firmware to Repetier-Firmware. We also switched the multiple extruder firmware to Repetier-Firmware to maintain consistency for the printer.

### 4.1.2 Heat Beds

During testing of the operation of the printer, we observed that the temperature of the heat beds displayed on the LCD screen would fluctuate greatly. The fluctuations would often exceed 10°C from the actual temperature of the heat beds, whether that temperature was the set point or temperature while the heat beds were unpowered. These fluctuations would worsen at higher temperatures; at the maximum temperature of about 90°C, the temperature would occasionally jump to over 120°C, which was well above the temperature limit with the current setup. A voltage test of the system determined that the thermistor was functioning properly, and the voltage divider was providing the proper voltage to the Arduino input port. We decided that the issue was most likely due to the reading from the program. To resolve this issue, we

modified the Arduino code to display the average of five measurement samples every five seconds, rather than a single measurement. This change eliminates outlier measurements through the averaging process, and should reduce the temperature fluctuations in the LCD display.

Another issue that surfaced with the heat bed system was the emergency shutoff of the heat beds would trigger occasionally since it could not read the thermistor. We first checked all the connections between the thermistors and the Arduino to make sure the shutdown was not caused by a loose connection creating a break in the circuit. Once we determined that all the connections were good and the issue persisted, we figured that there was a short circuit within the system. There is a short length of exposed wire where the wires meet the thermistor, which we thought was the most likely location for the short. We added an insulator between the lengths of exposed wire to prevent accidental contact. We also added thermal paste to the thermistors to create a better thermal contact with the heat beds to help correct the temperature fluctuation problem. Between the modifications to the Arduino code and the thermistors, the temperature fluctuation was substantially reduced, and the emergency shutoff issue was eliminated entirely.

### 4.1.3 Control Panel

The control panel on the front of the printer underwent several modifications as its requirements changed. The original design for the control panel used two separate panels, with one designated for the main control board and the other designated for the heat bed control system. This design took up too much space on the front of the printer, though, and required a redesign. The new design for the control panel condensed all the controls onto a single, smaller panel. Another change on the control panel was made to the side extruder switches. Since stepper motors use two separate coils, both coils must be unpowered to completely shut off the motor. Therefore, two of the four wires running to the stepper motor needed to have a switch

connected to control the power to the motor. To solve this problem, a dual-phase single-throw (DPST) switch was used to control power to the extruder motors. A DPST switch allows one to control two independent circuits with a single switch. Using this switch, we were able to fully shut off power to the side extruders.

## 4.2 Printed Parts

In order to demonstrate the functionality of our 3D printer, parts were developed with the intent to exercise the device to its full potential. Figures 4.3 through 4.7 and their descriptions display the extent of this printer's abilities.



Figure 4.2: Multiple Extruder Printed Parts

Figure 4.2 shows three “Hairy Lions” which were printed simultaneously using all three extruders. The parallel print functionality allowed our printer to manufacture these large, detailed components at three times the speed of a single extruder printer, without sacrificing precision or accuracy. An additional benefit of multiple extruder printing is that any idiosyncrasies or properties unique to this specific print will be recreated faithfully in all three parts, which would not normally be possible using only one extruder.



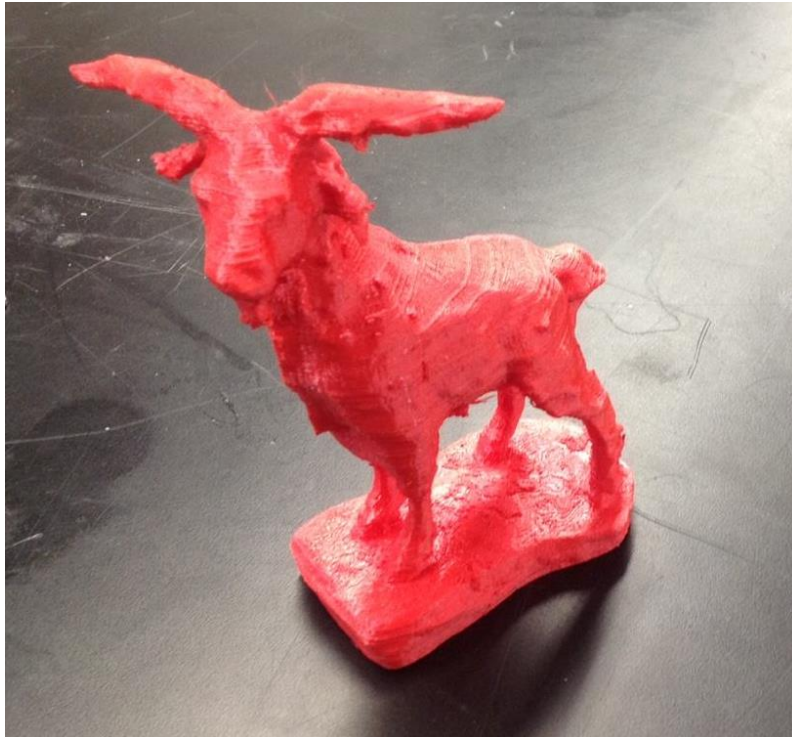


Figure 4.3: Gompei Statue Print

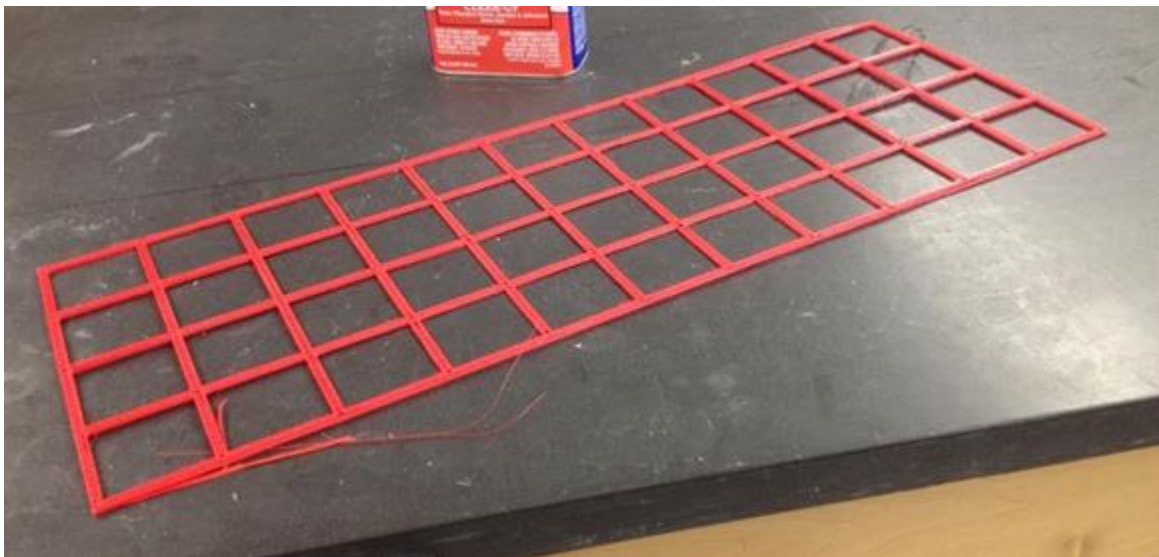


Figure 4.4: Heat Bed Leveling and Adhesion Test Print

The two parts displayed in Figures 4.3 and 4.4 are highly revealing about the massive scale on which this printer is capable of manufacturing parts. The Gompei statue stands over ten inches tall, and is constructed using several meters of filament. The Heat bed adhesion and



bed leveling grid has a maximum cross sectional area of several square feet, a scale which can be easily confirmed by comparison to the quart of acetone standing behind the part.

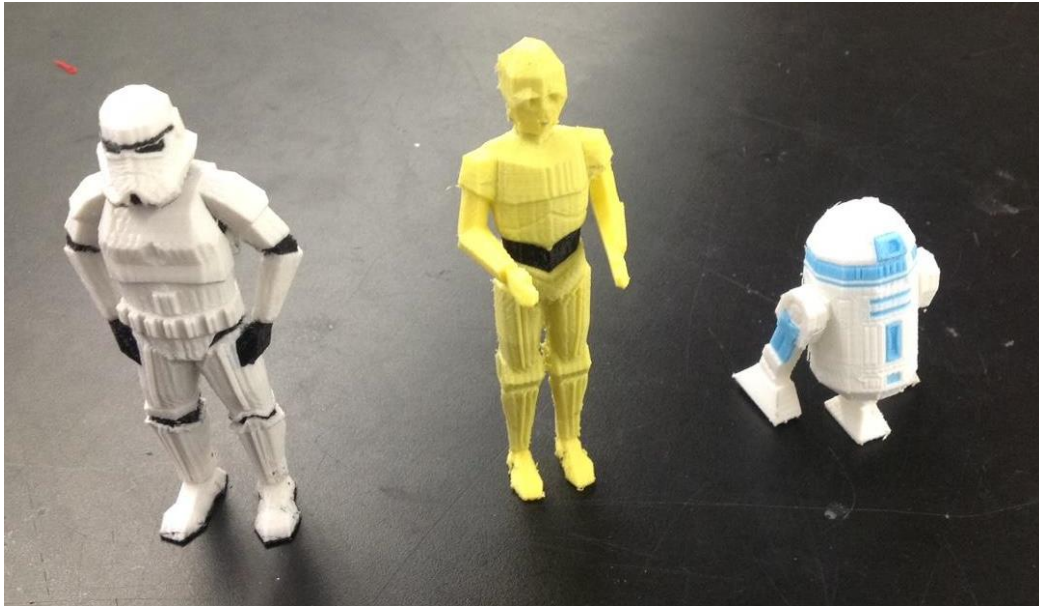


Figure 4.5: Multi-Color Printing Tests

The Star Wars figurines featured in Figure 4.5 are perhaps the finest demonstration of the multiple color function that this 3D printer has produced to date. It is especially noteworthy that the changes in filament color occur during multiple layers of each of these models, and in some cases, such as the lower torso of the yellow C3PO, the colors are varied at an angle relative to the print layers.

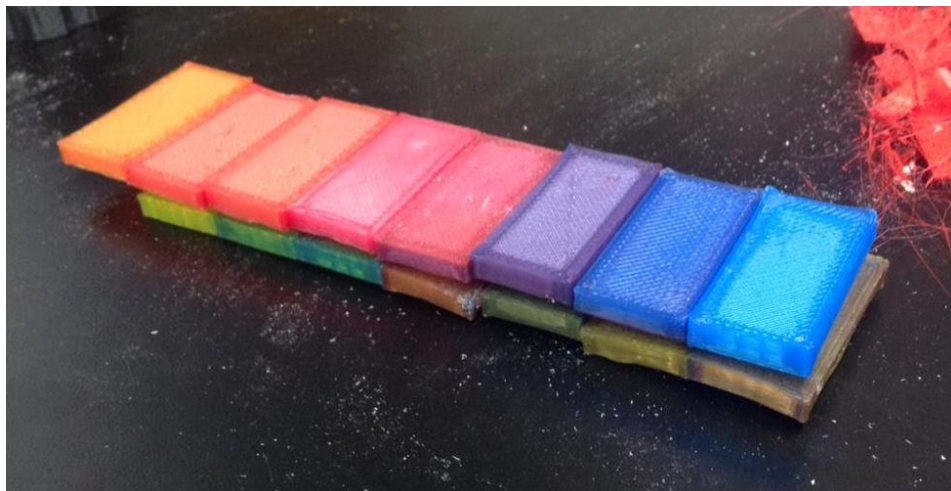


Figure 4.6: Color Mixing Tests

As a final demonstration of the this printer's ability to meet the design criteria specified at the beginning of the MQP, the color pallet from Figure 4.6 shows the result of our mixed filament color production. Using only three colors of filament, this additive manufacturing device was able to take advantage of open source algorithms through the Repetier software package to produce a full range of colors by mixing filaments in different ratios.

## 5.0 Conclusion

This section of the report documents the outcome of the 3D Printer Motors and Controls MQP. It is divided based on the design specifications laid out in the introduction, background, and methodology sections of this report. We first approach the criteria for a large print volume and multiple parallel heads, since these were the most heavily stressed features during our initial meetings. Having reported the increased scale and multi-print functions, we further detail the outcome of our multicolor and color mixing experiments. Each of these portions focuses initially on the conclusions of the combined efforts of all three groups, followed by an analysis of how the Motors and Controls group's input influenced the 3D printer. Finally, a section regarding budgeting is used to summarize the outcome of this project with respect to the design limitations implied by our limited funding.

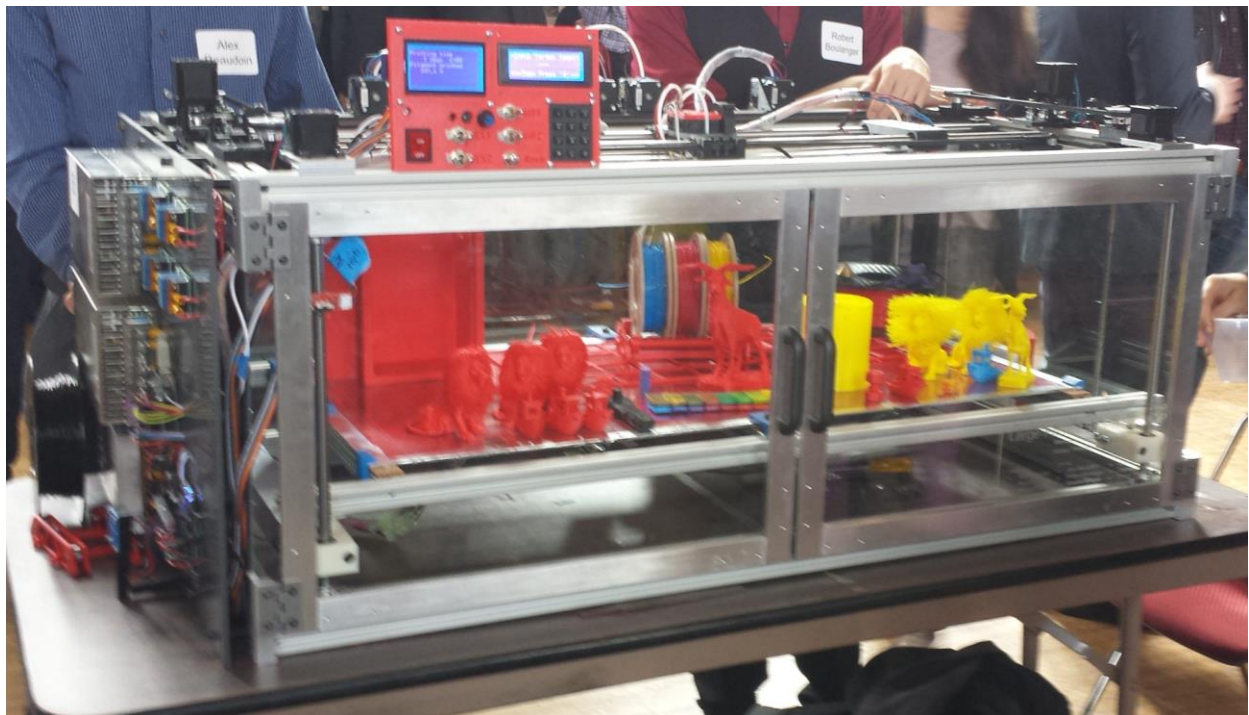


Figure 5.1: Printer presentation display

## 5.1 Large Print Volume and Parallel Head Conclusions

As evidenced by Figures 4.2, 4.3 and 4.4, both large volume prints and simultaneous print functions performed more than adequately. This success can be attributed largely to our fellow MQP teams. Their exemplary efforts in constructing the platform and fine tuning the print heads in conjunction with our motors and controls allowed our additive manufacturing device to meet the basic design criteria defined during the early stages of this MQP.

The advancements needed in the motors and controls in order to facilitate these design specifications were primarily affected by the additional motors and software updates needed to run the additional extruder heads across the oversized print volume. Our choice of the Rumba Board in place of the standard Arduino allowed us to manage the additional stepper motors, and our open source software selections made it possible to configure the printer for extra long axis motion. Additionally, the extra power needed to support the two large heated beds and the heavy duty stepper motors necessitated the incorporation of additional power supplies into our schematics and designs.

## 5.2 Multicolor and Color Mixing Conclusions

We were able to successfully print multiple colors thanks to the hardware provided by the other two MQP groups and the firmware we selected and configured. The Marlin open source firmware allowed us to print three separate colors from the diamond head, but was not easily configurable to allow color blending. For that, we ended up switching to Repetier-Firmware. We configured this firmware to use its built in algorithms to produce a full range of colors by mixing filaments in different ratios.

## 6.0 References

Arduino. "What is Arduino?" *Arduino - Home*. N.p., n.d. Web. 24 Apr. 2017.

<<https://www.arduino.cc> > [1]

C, Josh. *How to Build a 3D Printer: Getting Started Guide and Tips*. 26 February 2016. 22 April 2017. <<http://www.3dprinterprices.net/how-to-build-a-3d-printer-getting-started-guide-tips/>>. [2]

Coates, Eric. *Switched Mode Power Supply*. 17 February 2017. 22 April 2017.

<<http://www.learnabout-electronics.org/PSU/psu30.php>>. [3]

Columbus, Louis. "2015 Roundup Of 3D Printing Market Forecasts And Estimates." *Forbes*. Forbes Magazine, 28 Apr. 2015. Web. 25 Apr. 2017. [4]

Earl, Bill. *What is a Stepper Motor?* 23 November 2015. 22 April 2017.

<<https://learn.adafruit.com/all-about-stepper-motors/what-is-a-stepper-motor>>. [5]

Encyclopedia Britannica. *Thermistor*. 8 September 2009. 23 April 2017.

<<https://www.britannica.com/technology/thermistor>>. [6]

Future Electronics. *What is a Motor Driver?* n.d. 22 April 2017.

<<http://www.futureelectronics.com/en/drivers/motor-driver.aspx>>. [7]

Hausman, Kalani Kirk and Richard Horne. *How to Use Motor-Driver Modules on your RepRap 3D Printer*. n.d. 22 April 2017. <<http://www.dummies.com/computers/pcs/printers/how-to-use-motor-driver-modules-on-your-reprap-3d-printer/>>. [8]

Marlin. *Open Source 3D Printer Firmware*. n.d. 23 April 2017. <<http://marlinfw.org/>>. [9]

Maxey, Kyle. "Infographic: The History of 3D Printing." *Infographic: The History of 3D Printing* *ENGINEERING.com*. Engineering.com, 03 Sept. 2013. Web. 24 Apr. 2017. [10]

RepRap. *Choosing a Power Supply for Your RepRap*. 26 December 2015. 22 April 2017.

<[http://reprap.org/wiki/Choosing\\_a\\_Power\\_Supply\\_for\\_your\\_RepRap](http://reprap.org/wiki/Choosing_a_Power_Supply_for_your_RepRap)>. [11]

RepRap. *List of Firmware*. 14 March 2017. 23 April 2017.

<[http://www.reprap.org/wiki/List\\_of\\_Firmware](http://www.reprap.org/wiki/List_of_Firmware)>. [12]

RepRap. *Mechanical Endstop*. 27 December 2014. 23 April 2017.

<[http://reprap.org/wiki/Mechanical\\_Endstop](http://reprap.org/wiki/Mechanical_Endstop)>. [13]

RepRap. *Rumba Wiring*. 12 February 2013. 24 April 2017.

<[http://reprap.org/wiki/File:Rumba\\_wiring.svg](http://reprap.org/wiki/File:Rumba_wiring.svg)>. [14]

RepRap. *Stepper motor driver*. 14 March 2017. 22 April 2017.

<[http://reprap.org/wiki/Stepper\\_motor\\_driver](http://reprap.org/wiki/Stepper_motor_driver)>. [15]

## Appendix A: Overall 3D Printer Electronics Design

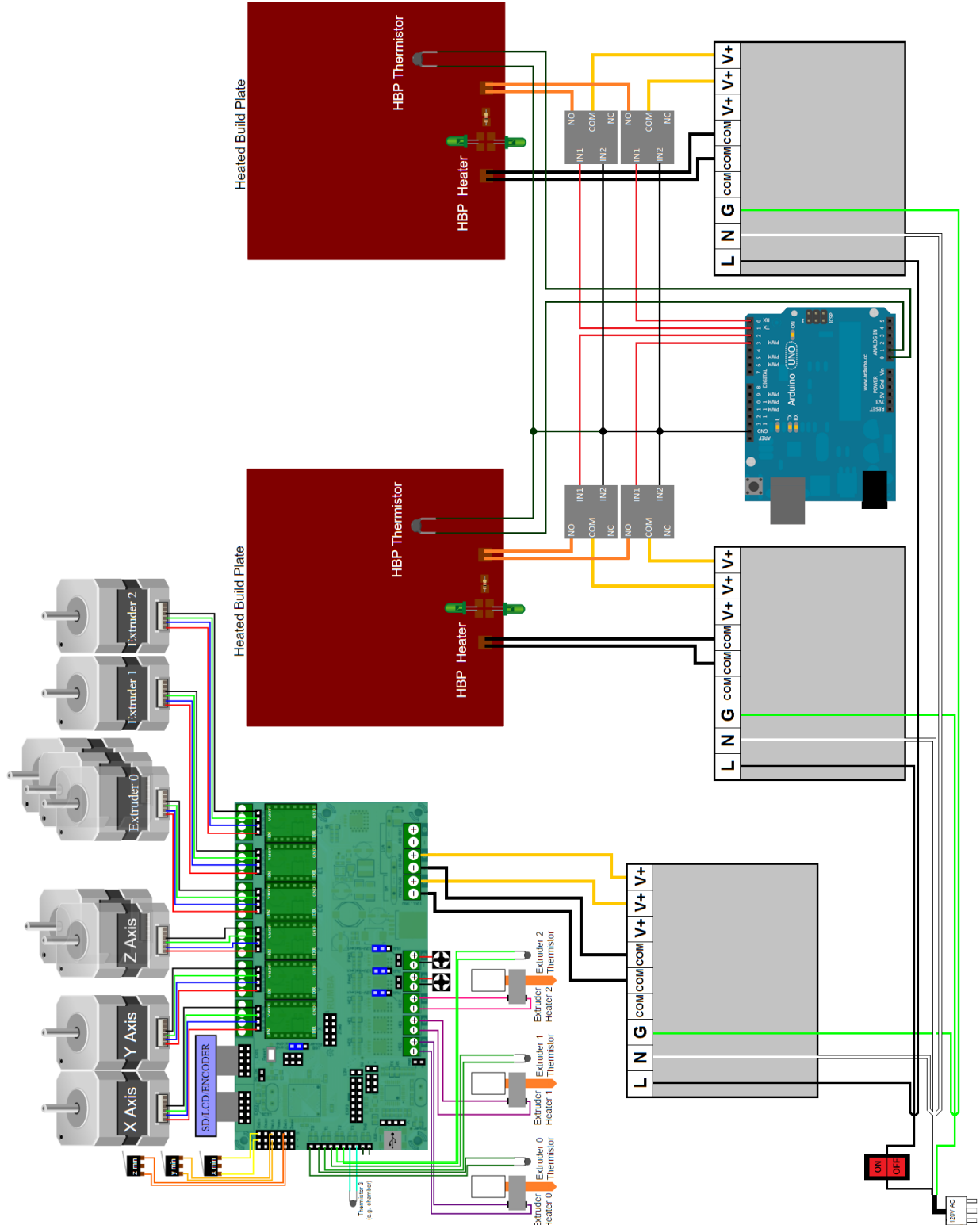


Figure A.1: Controls and Motors Wiring [14]

## Appendix B: Part List and Budget

Part	Quantity	Unit Price	Notes	Total
<b>RUMBA Control Board</b>	1	\$113		\$113.00
<b>Stepper Drivers (DRV8825)</b>	2(set of 5)	\$15.59	- Texas Instruments - 2.5A limit	\$31.18
<b>Stepper Motors</b>	5	\$12	- .4A per phase - 12v per phase - 26 Ncm - 200 steps/rev (1.8 degrees/step) - NEMA 17 - Bipolar Stepper - Comes with (short) leads	\$60.00
<b>End Stops</b>	1 (set of 3)	\$8.55	- Mechanical - Includes cables	\$8.55
<b>LCD Display + SD Card Reader</b>	1	\$16.80		\$16.80
<b>LED Strip</b>	1	\$8.00	- For lighting up the print area - 5 meters worth	\$8.00
<b>Power Supply</b>	1		- Must supply over 850 watts (12v)* *Need estimate for total power draw before power supply can be selected	
<b>Arduino (Heatbed Temp Regulator)</b>	1		UNO, Mega(depends on # of pins needed )	
<b>Relay switch (Temp Regulator)</b>	2		Need current draw from head beds	
<b>Total:</b>				<b>\$237.07</b>

Table B.1: Parts List and Budgeting