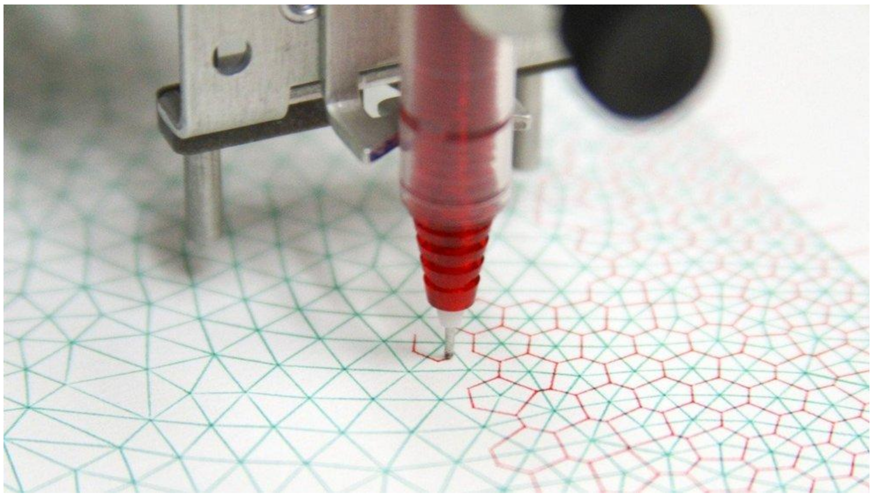
Senior Design 1

Automated Pen Plotter



Department of Electrical Engineering and Computer Science
University of Central Florida

Dr. Lei Wei
Dr. Samuel Richie

<u>Group 24</u>

Al Moatasem Al Abri, Electrical Engineering, mn5211@knights.ucf.edu
Patrick Caughey, Computer Engineering, patrick.caughey@knights.ucf.edu
Anthony DeMore, Computer Engineering, anthonyldemore@knights.ucf.edu
Peregrino Quansah, Computer Engineering, pquansah@knights.ucf.edu

Table of Contents

Total Pages: 121

1.0 Executive Summary

The act of applying ink to paper has been a concept that has been around since the 15th century with the invention of the printing press. This medium has advanced greatly since then but the end goal has always been relatively the same, to apply a drawing substance to a canvas without the need for a human to physically draw something. This allows for a much greater speed and precision when expressing a work on paper.

Throughout the history of printing, there have been two major forms of this medium which are in use to this day. The most common being laser printers, they are clearly the best for everyday use with their lower cost and speed of which they can perform their task. However, these types of printers are limited in the form of art which they can produce. There is a much less well known type of automated printing device that is still produced to this day and it is called a pen plotter. A pen plotters allows for fluid lines and beautiful artwork that tells a story, some would say to be created. Pen plotters are not as practical though, they typically take up more space, are slower and much more expensive than a typical laser printer. Also much of the software to use these printers can be seen as unintuitive for typical users.

The existing pen plotter market is relatively small in terms of new products. There is a sizable market for legacy plotters which were at one point sold as an alternative to traditional laser printers, but were eventually overtaken. There are new, modernized pen plotters being sold at fairly high prices. There are not many different competing companies, so we believe this is a good market to enter in order to bring about competition which would eventually lead to innovation of different aspects, especially in the software and usability categories.

Our design will consist of a mostly industry standard x-y axis automatically moving plotter. We choose to go with this industry standard route because it has been proven to be the most cost effective and versatile design. It also allows for an open viewing of the plotting process which is a decidedly very important aspect for our group. The software of our design is run by a user interface located on the computer attached to the pcb. The user interface will allow for the inputs of said users' desired plot.

We want to create a pen plotter that costs slightly less than one of comparable size, performance and technical specifications. Another goal of ours is to create software that makes our system easy to use and would allow for it to reach a wide range of users with varying technical skill levels. Our software will be a desktop application that will connect to the printer and control its output and allow for the user to make their inputs within it. Users will be able to enter coordinates or even input images to be outputted by our pen plotter.

2.0 Project Description

This section is used to detail our motivation on choosing this project as well as some of our goals categorized based on our group's thoughts on their implementation feasibility. There are also some high level specifications which we plan on implementing in our design. This section also illustrates how we will balance the marketing and engineering requirements in order to create the optimal design for our pen plotter.

2.1 Motivation

Printing as we know it has been around for quite some time now. The majority of printers we see on the market are inkjet and laser. A somewhat forgotten form of machines outputting images to paper format is known as pen plotting. A major reason for this is the lower cost and speed of inkjet and laser printers cannot be matched by pen plotters. However, the unique output style and artistic expression ability of the pen plotter cannot be matched by modern printers and that is what our group hopes to showcase with our project.

It is evident that pen plotters will by no means ever take over printers. There is however a significant number of enthusiasts who enjoy the art style and story-telling ability that pen plotters can generate. The more analog form of image and shape recreation is unique to pen plotters as they can draw continuous lines and write with the users' chosen writing utensil unlike modern printers.

We believe that the pen plotter market is somewhat of an undervalued and underutilized one. It is our belief that it is possible to make a more accessible product for the masses who will be able to take advantage of our design in order to produce interesting art pieces for the workplace or personal use. In order to do this we must make sure the parts we choose are optimized for our design in order to have no excess spending while still producing a quality product.

The commercialization of 3D printers in recent times brings to light the possibility for the greater popularization of pen plotters. They are similar in certain ways and it proves that there is a niche market for interesting cnc machines that can produce art right before the consumers eyes. The art of printing has been around for a very long time, ever since the time of the printing press the printing machine has been instrumental in the development of human civilization. A pen plotter can be used as another gateway of expression and to further the creation of art in different ways. A design such as our could possibly bring about further innovation in the pen plotting market which would allow for new ideas to be developed. Such as new types of writing utensils to be used for plotting, as well as the possibility for the writing canvas to be on something other than a 2D surface.

A pen plotter such as this truly rewards the experimenter. This is something we wish to allow for users to express when making different plots. A pen plotter is not a common instrument of expression for many people. With this new creative outlet, we hope that people will find a rewarding experience when they begin to utilize the plotter to its full potential. We do not expect this to be an easy thing to master right out of the box as it is more complex than a standard printer. Some designs may inherently work based on the nature of a pen plotter's movement and writing capabilities. There are so many aspects of it that are much better than a standard printer and that is another motivation for us during our group's brainstorming of this project.

Even though our pen plotter design will likely not reach an extremely large audience as it is not something that is considered a necessity by most, we hope that it will intrigue at least some people since it is such an interesting concept that we hope others will appreciate as well. This type of product is something that rewards the creative thinker and anyone willing to experiment. We look forward to the building and application phase of our design in the next semester as there are so many different pieces of art we can create. We hope that we are able to achieve as many goals as possible in the following senior design semester as it will allow us to create the best final product.

Another goal outside of the specific design that was a motivation was the challenge of working on such a project. This is a new field for our group and we aim to use it as a good learning experience that we can use to improve our futures by giving us valuable skills and lessons in engineering design and management along the way. It is important we learn to work as a group in this project since there are many moving elements and we all need a good understanding of our specific assigned elements, as well as the design as a whole. Every element we work on is important in its own way and we hope that we learn along throughout the course of this semester. We believe that this is a project we are all interested in and that is why we are motivated to work on this project based on the above paragraph.

Our reasons for choosing this as our design are many. Partly because we think this would be an interesting project, and one that we could easily display on a resume. It is something that can be shared to a wide range of people and for all of them to find something interesting with it. Another reason we wanted to make this our project was based on the fact that many pen plotters that are currently on the market are somewhat costly compared to the parts needed to create one yourself and some are not very user friendly. By creating easy to use software and hardware environments. We hope to accomplish both of those things with our pen plotter.

2.2 Goals

Our goals for the design of this product are to create an accessible and well-designed pen plotter. Accessibility means that we plan on making an easy-to-use interface which will allow the user to input their drawing without the need for programming knowledge. The interface will be accessible via software which is downloaded to the user's computer. The users will have different options for input. One implementation is an inputted image which will be converted to plottable lines by edge detection algorithms. We want to also allow for the user to determine how sensitive the edge detection should be by seeing a preview before the image is plotted. We also want the user to be able to input pseudocode or coordinates instead of an image in order to plot with a different method.

The choice of inputs is good for our ease of design and testing as well as important for the user. Different users may have different intentions in mind when creating a plot. A more professional space may need to input specific coordinates and take advantage of the use of pseudo code in order to speed up the process of inputting designs. A standard consumer may find the image inputting and edge detection method to be more to their liking as it is a more interesting and easier method from a graphic design standpoint.

If possible, we want to create both a desktop application as well as a mobile companion application or a standalone app as a stretch goal. The apps will allow the user to further control the machine as well as see a preview image of what the output will look like before it is drawn. There will also be a timer as well as a stop button included in the app just as in the hardware.

There are a number of ways in which we can implement the software for the project. There are many options available in terms of programming languages and they would most likely change if we needed equivalent mobile and desktop applications. The desktop app is certainly a higher priority for us than a mobile app as it seems like it may not be important for every use. Especially in the professional world, a mobile companion app would most likely be seen as a novelty and not something that would be used very much in a production sense.

The ability for the user to choose their desired writing instrument is also an important goal in order to allow for artistic expression or to meet a specific criterion of a drawing. A well-designed product would be one that is built sturdy enough to be moved from place to place as needed during the demonstration of our design as well as for any users' purposes. It also needs to be lightweight while remaining rigid. These types of machines can take up a large amount of space. Especially in a user's home, space can be an issue, therefore, we would like to make the machine as small as possible while still outputting a meaningful piece of art or design.

We would like to implement some quality of life features such as a kill button to stop the process if a change needs to be made or a malfunction occurs. Another possible quality of life feature would be a physical percentage done located somewhere on the plotter. The mobile companion app would likely need to utilize a local wifi network or bluetooth compatibility in order to function with the desktop app or as a standalone application especially. We put the wifi or bluetooth as a stretch goal since it would mean significant changes to our pcb design.

Core Features	Advanced Features	Stretch Features
● Plot user inputs on paper automatically simple coordinates and commands in desktop application ● Required desktop application which operator uses to set the output ● Plot user inputted images using edge detection algorithms to determine lines to draw ● Ability to use different pen/pencil writing utensils ● Kill button to stop drawing process immediately ● Plotter preview in software before drawing is started	● Companion mobile application which allows users to control certain features of plotter. Such as monitor percentage done and start/stop process ● LCD display showing percentage done ● Allowing for multiple writing utensils to be used on one drawing (manually changed) ● Ability for user to modify certain parameters of the output image preview within the software	● Automated changing of writing utensils ● The use of paint brushes/ability to retrieve paint automatically ● Mobile app with more features such as ability to draw idea with touchscreen to create output with plotter ● Wifi or bluetooth capabilities on the pcb for the mobile application.

Table 1: Goals and Features

2.3 Competitive products

There are various different pen plotters currently on the market with a fairly wide range of prices. Pen plotters are typically slightly more expensive than printers. We plan on designing a product to be marked towards the low to mid-price range. The differences in low to high priced pen plotters is typically their drawing area as well as additional features such as automated pen color/style switching.

There is also a large amount of pen plotter kits unassembled which can be purchased for a much lower price than a premade kit. It is clearly a hobby tool and that is our primary audience for this project. Seen on the next page is an example of a competitive product sold by GooGi Geek-Lab retailing for $275.

This is around the price range we are going for. Based on industry standards it seems like it would be fairly difficult to go under this price range without affecting the production to selling cost ratio. If automated production practices were used, it would definitely become a viable option. The visual representation seen below is the pen plotter that was described above:

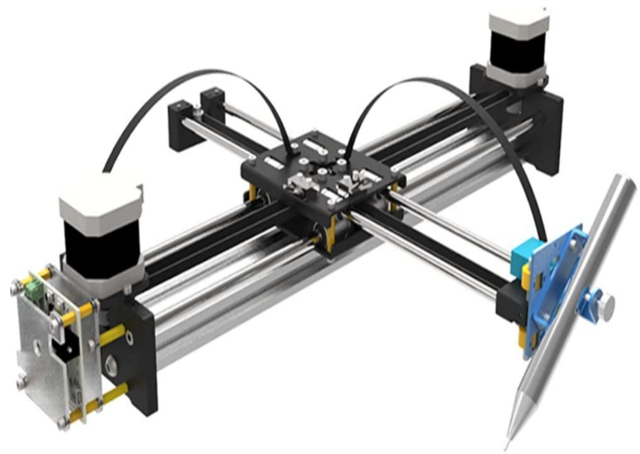


Figure 1: Existing Pre-Fabricated Pen Plotter

We will be basing our design on diy kits as we do not have the proper mechanical engineering experience to fully create our own unique design. Not only that, the blueprint for the optimal pen plotter design has pretty much been set by these diy kits and current pre-built plotters on the market. Premade kits are very expensive so that is why we will be assembling our kit ourselves and this will also give us a deeper understanding of how it is put together so that we can further manipulate it to create a better product.

There is also a market for legacy pen plotters which resemble more of a traditional closed printer-like design. These can be found on ebay, typically for a much lower price than a new-modernized plotter. Legacy plotters while they may make for a very interesting display piece. Do not match the precision and ease of use of modern pen plotters. The maintenance of legacy plotters should be enough to steer serious users away from such products alone. The parts and reliability of such devices come into question as they are from a much different time period and the market is small where there may be little to no replacement parts and documentation on how to replace broken parts if needed.

2.4 Specifications

The specifications shown in the table below were chosen by our group based on the capabilities of the parts on our parts list, and somewhat on industry standards of similar products. There were some major design decisions our group had to make based on the marketing and engineering requirement balance we had imposed on ourselves. We wanted a relatively cheap design, while still creating something that users would enjoy and at the same time would be an effective artistic device. Below there will be details highlighting why these specifications and their subsequent parts were chosen for our design.

The three stepper motors are used to move the rails which control the pen in the x and y directions in order to plot the desired output. The third motor is used to hold the pen and adjust it based on the size in order to get proper contact with the writing surface. The surface area is slightly larger than a standard 8'11'' page to account for the actual device being slightly outside of the writing area. We are choosing this writing size in order to make our machine somewhat portable and to not take up a massive space. Also for the convenience of using the most common sizing of paper in the US.

The base size of the plotter is the surface area plus the total area our plotter takes up. The plotting precision is there to give a possible margin of error for the output with respect to the original inputted drawing. Plotting speed is based on the power of the motors used and needs to be kept somewhat slow in order to create a smooth image given the writing utensils used. The clamp for holding the pen is powered by a servo motor in order to be adjusted based on writing utensil size and length. The pcb needs to be compact and light enough to fit on the moving x-y arms of our design.

We plan on implementing three different switches for different functions. Confirm, cancel and emergency stop. Confirm will start the plotting and cancel can be used before the plotting has commenced and emergency stop is mostly seen as a safety feature in order to stop the plotter while it is in motion. A power supply is needed in order to power the pcb and subsequently all the motors needed for our design. The voltage specified was based on the power standards of our microcontroller design as well as industry standards for outlets.

For our power supply, 12V is needed in order to power the arduino and the motors and other components attached. The arduino is also plugged into the computer via a usb and the output of it will be 5V. The firmware will be loaded onto the arduino in order to run the processes necessary to plot on the desired points. We believe we can acquire all of these parts for around $300 total.

Below is a table representing the design specification outline in the above paragraphs. There are three columns representing the specification itself, ther reasoning behind the specification and the technical value of each specification. It is slightly more technical than what was expressed in the preceding section. The described table can be seen below:

Specification	**Reason**	**Number**
Stepper motors	X Y and Z axes	3 motors
Working area	the larger dimensions of A4 and U.S. Letter Sizes with	9.5" x 12.75", 242 mm x 324 mm or greater

	additional margin of 0.5 Inches	
Base size	working area plus 100 mm on top, 30 mm on each side, and 30 mm on bottom	302 mm x 454 mm or greater
Plotting precision	Much less than the line weight of a standard pen	0.1 mm or less in the X and Y directions
Plotting speed	Reasonable speed for writing or drawing	~20 mm per second
Clamp fitting for pen	Fits most standard pen sizes	6 mm to 14 mm
PCB full size	Plenty of room for display and controls	120 mm x 100 mm or smaller
Switches	Confirm, cancel, and emergency stop	3 switches
Motor power supply voltage	Appropriate voltage for chosen motors	12V
Microprocessor power supply	Appropriate voltage for microprocessor	5V
Firmware size	Size of medium size C code when converted to 32 bit machine code	~10 kB or smaller

Table 2: Hardware Specifications

2.5 Marketing and Engineering Requirements

This section is used to represent the relation between our marketing and engineering requirements. Our group assembled a house of quality visual diagrams because it is an optimal way of displaying how certain marketing choices have an effect on the engineering aspects of the design and vice versa. This figure was carefully crafted in order to showcase the reasons we went certain design routes instead of others. We had to make sacrifices in some elements along the way, such as high cost, which can cause a slightly different engineering experience for the user. However, we do not believe that our choice will in any way provide a negative experience for the user.

The image seen below is known as a house of quality diagram. A house of quality diagram is used to represent the relationship between our engineering requirements and marketing requirements. The House of Qualities shows how each of the marketing and engineering requirements relate to each other and how strong or weak the relationship is by a symbol as it is shown in the legend. Lastly, at the bottom of the diagram we have our targets for each of the engineering requirements features. See the image below to see what has been referenced to in the above paragraphs:

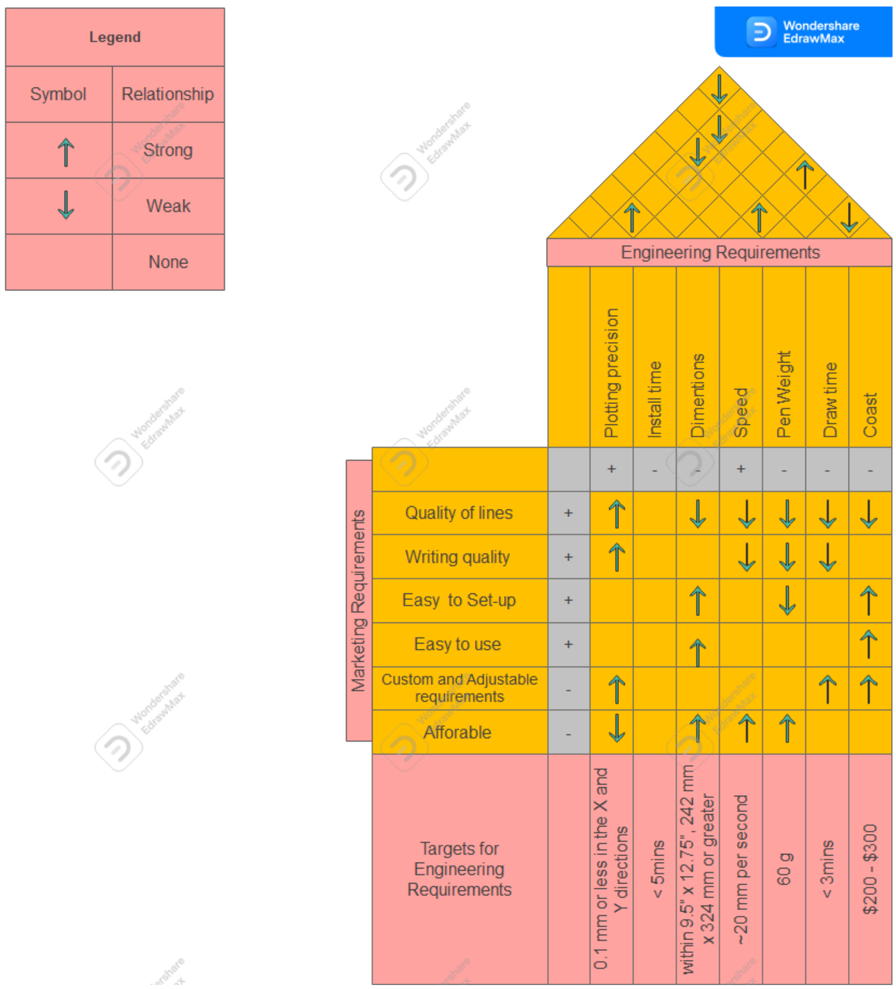


Figure 2: House of Quality

As shown above, there are several marking and engineering requirement tradeoffs we will need to take into account through the course of our design phase. The following considerations will need to be made in order to create the optimal design as determined by our group. Below are some of the thoughts and reasons we will make certain design decisions based on the above house of quality.

Firstly, if the speed of the printer is increased, the quality of the outputs will be greatly decreased. Draw time is correlated to speed in many ways, so it has the same negative impacts on marketing requirements as speed. Printing precision has positive impacts on marketing requirements except for affordability. One of our goals is to make our design somewhat affordable in comparison to some of the higher end offerings of pen plotters.

This is another thing we will need to take into account when it comes to engineering and marketing tradeoffs.

In terms of user ease of use and ease of assembly, there are several things our group will need to take into consideration. The device should be somewhat small in order to make for the assembly and the usage and storage of the device to be somewhat easy to manage. The parts need to be easy to work with and not require numerous tools and machining equipment which our group would not have access to.

2.6 Narrative Conclusion

In conclusion, our group plans on developing a well-rounded pen plotter to meet both the hardware and software requirements needed to create a viable and interesting product. It is important for us to create a design that is able to make unique works of art as well as be a tool which can be for entertainment. A pen plotter is a tool unlike the standard printers in most people's homes, so we hope to enable users to have a new outlet for creativity which they can use to expand their artistic talents.

As a group, we hope that this project is challenging and enjoyable in many different aspects. Not only do we hope that we are rewarded by choosing a somewhat challenging project in order to create a design that would be a good point of interest in our future careers as well as to expand our knowledge on many topics of hardware and software. This type of long term project allowed us to learn many things which are used within the industries that we will be entering post graduation.

There are many important elements of software and hardware that go outside the bounds of what we were taught in our core classes throughout our academic careers. One reason in particular that we feel this way is since this design is heavily influenced by mechanical engineering themes. Our group does not consist of a mechanical engineer so it was a challenge for us to understand this type of knowledge as it was not taught to us in our core courses. Throughout this design phase we hope to gather the necessary resources as well as knowledge in order to create a mechanically functional machine.

Throughout this paper, we will go into greater detail on all of the design specifics as well as how the pen plotter is used. There will also be many instructions on how this pen plotter is used, and the thought processes that went into the creation of these functionalities. We aim to fulfill as many of our goals as possible in order to ensure that our motivation for creating this project was one that was fulfilled in many different ways, since we believe this project will be a great learning experience that we can use to further our careers post graduation.

2.7 Project Illustrated Prototype

This section was created to showcase our project overall at a high level. It is not a representation of every element of our project as it would be difficult to depict such a thing in a concise and readable way. It was important for us to showcase this since it is a good image showing the key elements we hope to create throughout the course of this semester.

Below is a basic representation of the key aspects of our design. This outlines the flow of some of the ideas we may implement in our project but it is subject to change throughout the design process. We aim to make the necessary changes throughout the course of the project in order to have an effective balance between our marketing and engineering requirements that we had imposed upon ourselves. If all the elements in the figure below are successfully created by our group by the end of the following semester, we will be fully satisfied as our motivations and goals for this project will have been met. The prototype can be seen in the following visual:

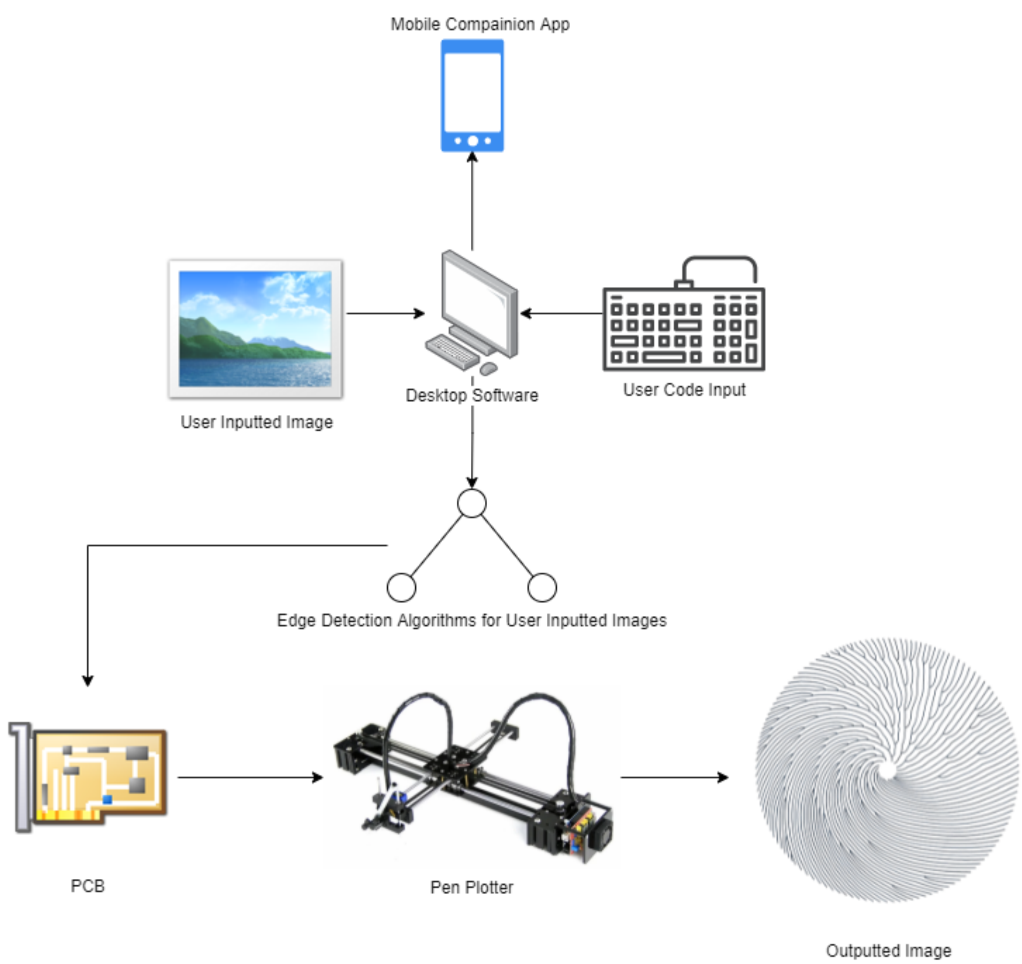


Figure 3: Illustrated Prototype

3.0 Related Research

3.1 Writing Implements

There are several varieties of writing implements available to the project. One available option is a pencil. Pencils are not ideal for several reasons. Firstly, they need to be sharpened occasionally. Sharpening the pencil could be done manually by stopping the machine, taking the pencil out, sharpening it, replacing it, and starting the machine again. It would, however, be difficult to put the pencil exactly where it was when we took it out, so it would have to be recalibrated every time it is sharpened. The machine could also be programmed to automatically sharpen the pencil after either some time has elapsed or by sensing something about the pencil. The pencil would also become shorter as it is used, so the machine would have to lower it as it is being used. The pencil would also require pressure on the paper to be used properly. Lowering the pencil as it is used and applying pressure to the paper could be accomplished with either something spring loaded, pressing it down into the paper, or a pressure sensor to handle it in software. A gravity based mechanism might work, but based on testing, the pencil alone doesn’t have enough weight to write under its own pressure, but weight could be added to the sliding mechanism.

Another option for writing implements is a pen. Pens have several advantages over pencils. Pens never need to be sharpened, unlike pencils. Pens also don’t change length as they are used, so they can sit at a single calibrated height the whole time. Pens also come in a variety of colors, which would allow for pictures to be drawn with a variety of colors in them. Different varieties of pen do have different constraints however.

One type of pen is a ballpoint pen. Ballpoint pens require pressure to be used, similar to pencils, though much less. Ballpoint pens do have a problem, which is noticeable in normal use, which is that the roller ball at the tip can sometimes dry up on one side making it write or draw inconsistently. Another problem is that sometimes the ball point stops rolling properly.

Another type of pen is the fountain pen. An ideal fountain pen writes with no pressure needed, which gives it an advantage over ballpoint pens. Fountain pens are, however, more expensive than other varieties of pen. Fountain pens also have a more limited selection of ink colors, as they are intended for calligraphy and professional works. Fountain pens also typically require a fairly specific angle to be held at to write properly. Similar to fountain pens are dip pens, though those also require dipping the pen into an ink well, which would add even more unneeded complexity.

The last type of pen to be discussed is the felt tip pen. Felt tip pens require slight pressure to write with, but they can work with a variety of pressures, and can even produce different line weights this way. Felt tip pens also come in the largest variety of colors and tip diameters, giving the most options for drawing a picture. Markers are a variety of felt tip pens.

Other options for writing in this machine are brush based writing or drawing implements, such as painting or brush based calligraphy. Brush based drawing or writing would require moving the brush through angles and using it to pick up more paint or ink. These would require adding much more complexity to both the hardware and software, but it would produce very impressive results.

Overall, Felt tip pens are the best choice for the simplicity to implement, the line color and width availability, and the lack of problems, which the other choices have. The device should still be usable with ball point pens, but where needed, it should be optimized with felt tip pens in mind.

3.2 Paper Dimensions and Working Area

There are several dimensions of paper available to us. The most common size of paper in America is US letter size, which is 8.5 x 11 inches. There is also US legal size, which is 8.5 x 14. The most common size of paper in other areas of the world is A4 size, which is 8.25 x 11.75. We are most concerned with US letter size, so our working area should be at least 8.5 x11 inches (216 x 279 mm). The machine will also need some amount of overscan in order to guarantee it can reach all four corners of the paper without hitting the limit switches. A 1 inch border on each side will be plenty of overscan. So the machine’s working area could be 10.5 x 13 inches.

If a pen switching mechanism is completed, the machine would need to be able to reach it, so it would need about 2 more inches clearance on one side, which would give a final working area of 12.5 x 13 inches.

To add space for limit switches and any other adjustments needed between planning and completion, the rails and carriages should have a sliding length between 15 and 20 inches long.

3.3 Materials

For this project we will need a lot of materials to build it in a perfect way to make it work perfectly. We have divided the materials used to build the project into two main parts first the assembling parts which they represent about eight parts such as, MGN15H Linear Rail, Linear Rod 6mm, Linear Bearing 6mm, GT2 Belt, Tooth Pulley, GT2 Idler Pulley, Spare Nuts and Bolts and Nuts set all these parts will be shaping the buddy of the project. Also we will be using a base to set up and that base can be made of anything that is cruel and steady so that the pen plotter moves smoothly and in a proper way.

The second part of the materials that are used to build the project are the electronic parts where the electronic parts will be taking the mission of moving and controlling the project parts and some of them will be ordered online from the common online markets and some of them will need to be designed by our group since this a requirement for accomplishing this project and learning from it.

The first assembling part that we need to mention and choose carefully is the Linear rails. Linear rails are parts that are commonly used in moving objects, so linear rails designed to support and help the movement of the objects in a linear way which means that the object will be moving either vertically or horizontally . Linear rails will be doing multiple tasks and will play a huge part in our project, so working on choosing what type of linear rails we should choose and what they are made of is very important. For our project we will need to use two linear rails one for the X-axis and one for the Y-axis.

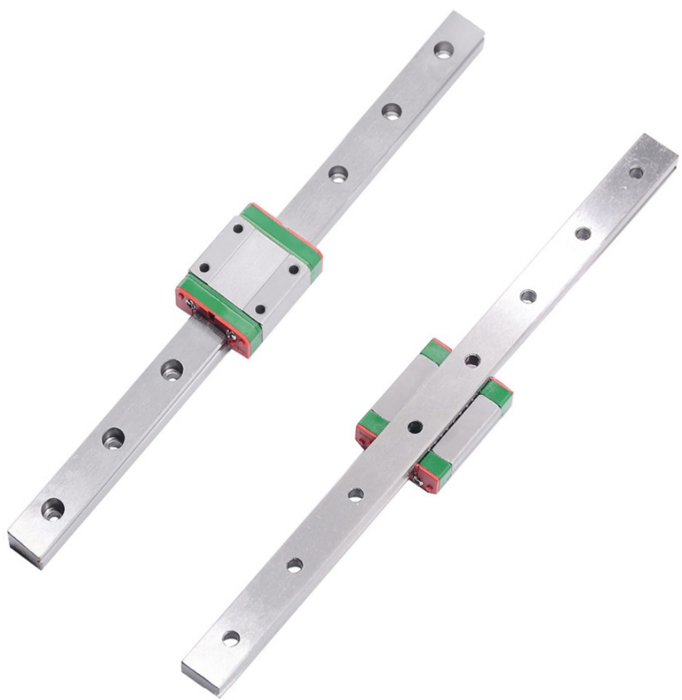


Figure 4: Linear Rails

There are different types of Linear rails and different shapes of them and each one of them has similarity and differences from the other one. For example there are round rails and square rails and each one of them is different from the other in some ways as shown in the picture below.

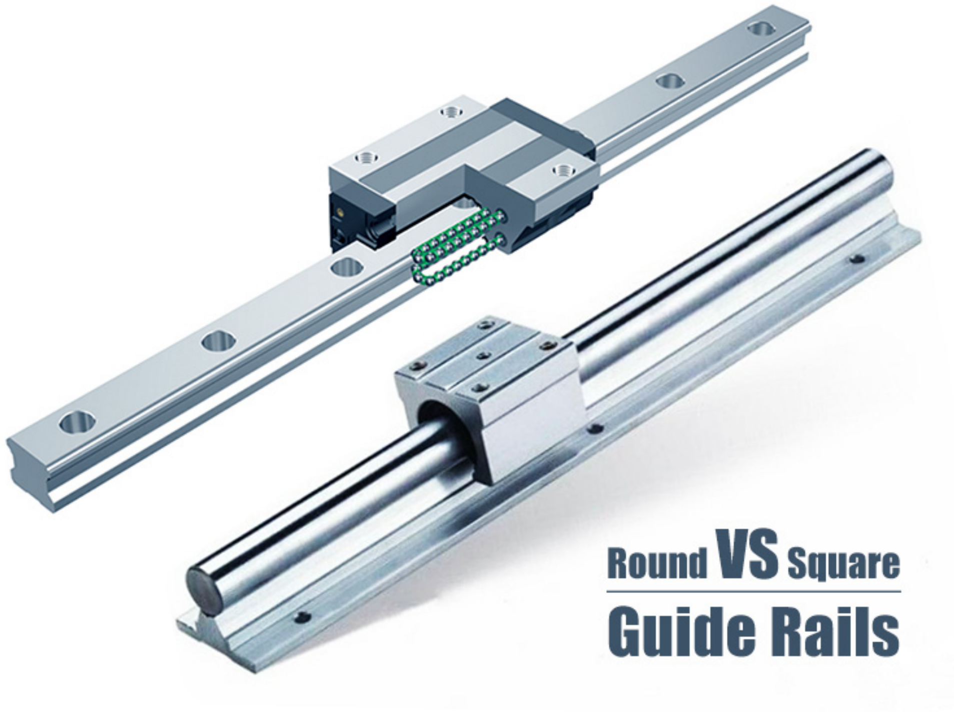


Figure 5: Round Vs Square Rails

To decide what rails we should use for our project we had to do research about what are the main differences between the two rails and which one of them will work perfectly in our project. We have collected a lot of information about the two rails and we have compared them in the table below and have chosen the squire rails since it is more suitable for our project and it will work perfectly.

Differences/ rail type	Round Rail	Square rail
Commonly used in	Automotion, printing, welding and inspection station.	Automotion and printing
Advantages	It can rotate around the axis and they can self aligning .	They have High accuracy rate and they can handle heavy loads.
Quietness	Round rails are the best.	Less than the round rails.
Smoothness	More smother.	Less smoothness.
Impact (shok) load	Do not have good surface contact.	They have the greatest surface contact.

High load	It is not the best choice for high load.	It is the best for high load.

Table 3: Square Vs Round Rails

Linear rails and the linear bearings that ride upon them need to be inflexible and smooth/low variance. Metal seems like the best option for this. It is also worth considering plastic, because it is lower weight and lower price. Plastic, however, is too flexible and can have static electric effects that may change resistance to motion in unpredictable ways. Wood is not smooth enough to be a good fit for this application. Metal parts are also much more commonly available than plastic for this application. Of available metals, steel is the most commonly made into these parts. For these reasons, the rails and linear bearings riding on the rails should be made of steel, to make that decision we had to do a comparison between several materials and what material should we use and from that copersin we figured that the rails that are made of metal are the best choices for our project.

After all the searches and copersion we have found the best rails for our project with the dimension that we will be using, all the specifications of the rails that are used in our project are represented in the table below.

Material	Alloy steel
Quantity	2
Part number	RZ008-500-2
Size	MGN15
Side travel distance	500 millimeters
Style	500 millimeters + MGN15H

Table 4: Rails Specifications

One of the most important parts that are used in our project which is a motion part are round rods. Round rods are parts that are used to help the moving objects in the project to move smoothly, for our project we will be using some of them to move the pen holder and they are made of metal. These rods will help the pen holder make better contact with the working surface. The round rods that are used in the project are represented in the figure below where it is already placed in the correct place.

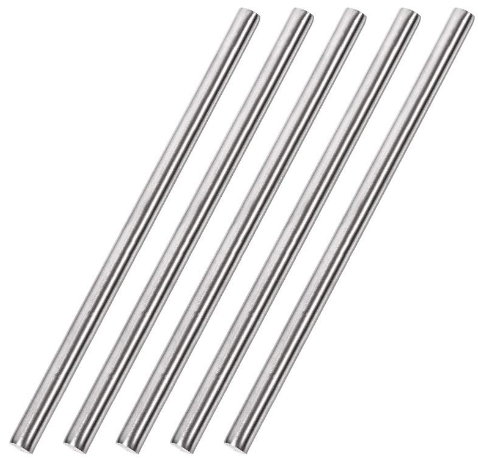


Figure 6: Round Rods

The used rods in this project must have specific features. It works perfectly with the pen holders and it gives best output otherwise there will be some issues faced during the process of the printing, and to prevent that from happening we have collected the best specifications of the round rods which are represented in the table below.

Material	Stainless steel
Quantity	2
Size	6mm * 150mm
Shape	Round

Table 5: Rods Specifications

After choosing the Linear Rails and the round rods which are all these parts are motion parts now we need to find parts that are going to hold and install these moving parts. These parts need to be not cruel because we will need to install the rails on them and if the used parts are cruel parts they will cause problems during the installation. After searching and working on what type of material we should use we have decided that the material should not be heavy or light; it needs to be in between, because working on heavy material will cost a lot and working on light will cause problems with the moving parts.

The frame of the machine doesn’t need to be smooth, so price and availability are much more pressing concerns. Weight would also be a concern, but less pressing than availability or price. Metal appears to be a bad choice for this because it would be expensive and difficult to get in the exact shapes needed. Plastic would be a decent choice for the frame because it is cheap to 3D print any shape needed for the machine. Wood is also cheap compared to metal, and is also easy enough to get in whatever shapes we need. Of the available choices, 3D printed plastic is the easiest to make and remake, so it seems like the best choice for the frame.

We will need to use several plastic parts for holding and installing the rest of the project. We will be using two plastic parts as holders for the Linear Rails one of them will be long and wide and the other one will be smaller. Aslo, we will need one plastic part to hold the stepper motor and connect it with the linear rail, two plastic part will be used as the pen holder and to cover the round rods and one part will be used to hold the PCB and connected it to the plastic part that is connected to the linear rail.

We will use 3D printing to create and shape the plastic parts after we get the correct dimension for what our project will look like and try to connect all the parts together. The figure below will show how the plastic parts look and how they will be connected to the other parts.

Figure 7: Plastic Parts.

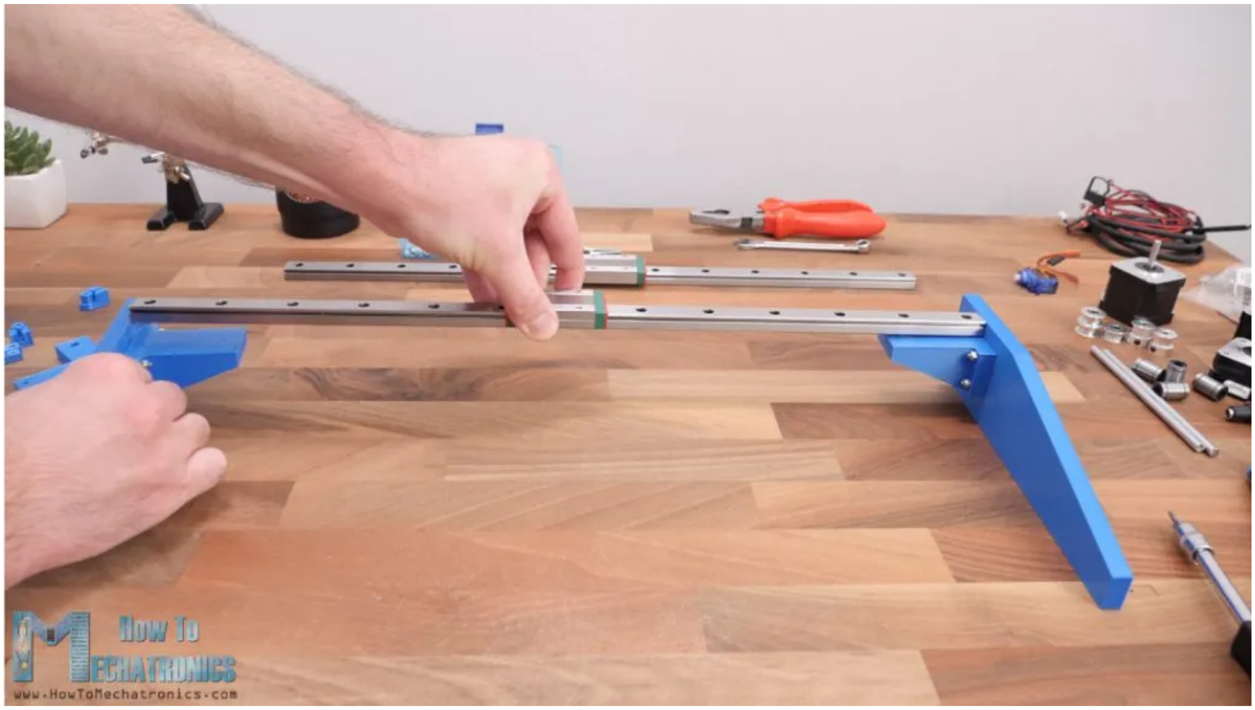


Figure 8: Plastic Parts Connected to the Linear Rails

After we finished with the plastic parts that will hold the moving parts we moved to the surface base that we will use to install the whole project on. The surface that we are going to use needs to have some features that are necessary to make the project work perfectly and for the pen plotter to move smoothly. On this surface will put the paper that will be used to get the output and to do that this surface must be bigger than the paper so that the pen plotter moves easily and smoothly. This surface can be made of materials that can hold the project well and not move while the project is working and this surface must be soft so there are no drawing issues. This surface can be made of wood or plastic and it will be designed using 3D printing. Also, this surface can be made of ceramic but it will cost a lot of money. The surface will look like the figure below.

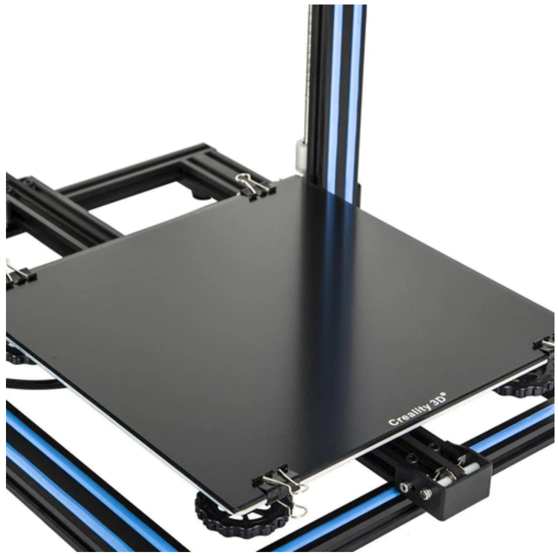


Figure 9: Working Surface

The surface the paper rests on must be smooth/low variance in order to not convey any pattern onto the drawing. Since the machine will most likely rest on top of a table, and the table will be smooth enough, the table will likely be a good enough surface for writing on. So the machine should have an open bottom, where the paper can rest directly on the table. This also reduces weight by not requiring an additional surface.

The paper that will be used on our project will be representing the working area and the pen plotter will be moving on that paper. This paper that we are going to use must have features and specifications before starting to work on the software, because these specifications need to be inserted into the software so that the output looks the same as the user wants. After some research and distinctions with the group members we have decided that the working area is larger dimensions of A4 and U.S. Letter Sizes with additional margin of 0.5 Inches. Which is 9.5” x 12.75”, 242 mm x 324 mm or greater. And the figure below shows how the working area will look like.

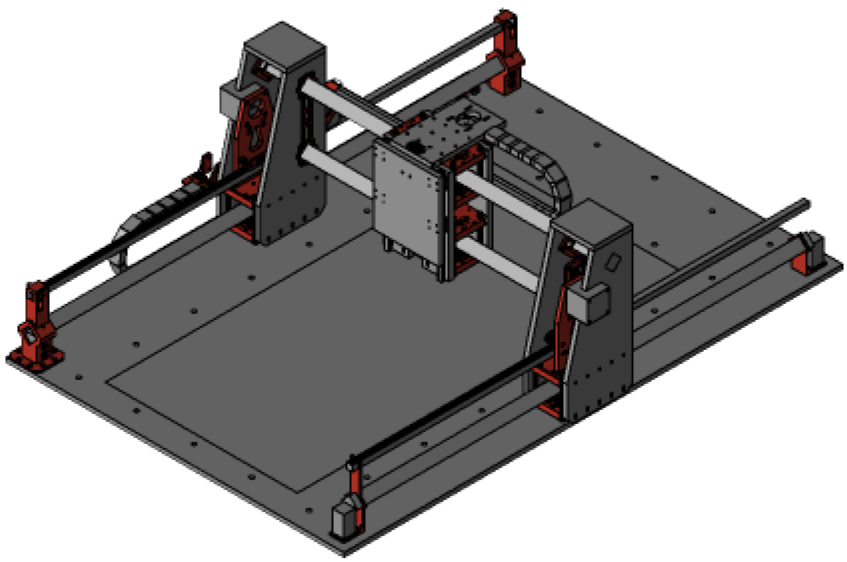


Figure 10: Working Area.

There are other materials that are needed while working on collecting the project part such as the belts that drive the slider.The belts to drive the sliders along the rails need to be cheap, available, and make affirmative contact with the sliders. Rubber belts are cheap and available, including varieties with teeth built into them for better contact. Leather belts are also available, though not usually at small scale like this project is. Leather belts would also be relatively difficult to make it have affirmative contact with the sliders. Plastic chains are also available, but may be more finicky to use. Belts can always be cut to size, but chains need to be made shorter or longer one link at a time, so if the desired length isn't a perfect multiple of links, a tension pulley would also be needed. Metal chains have all of the problems of plastic chains, except they are stronger, though we don't need that strength, and they are more expensive. Our belts should probably be toothed rubber belts, for their low price and availability. The belt that we are going to use is as shown in the figure below.

Figure 11: Timing Belt

With this timing Belt there are some metal material that will come with it which will be helpling the belt to move around while running the project the materials such as, pulleys that has 20 teeth with width of 16mm, Idler that has 20 teeth with width of 5mm, timing belt transistor spring that is made of stainless steel and inner diameter of 5mm. All these parts come with the belt which they have the same specifications to work together.

There are some other parts that need to be available while working on collecting the project parts, such as spare nets, screws, bolts and nut sets. All these parts must be provided while working on collecting the project parts so that we can get the best outcome of our project and not having any issues during running it. All these parts are shown in the figure below.

Figure 12: Needed Parts

3.4 Electronics

The second main part which is an important part in the project is the electronic parts. Every electrical project needs electronic parts to work otherwise it will not be called an electrical project. And to create a big project like our project we had to do so many research meetings and investigations of what electronic parts will our project need to work perfectly.

And we found that the relevant electronics for this project are 3 motors for controlled linear motion, 1 other motor for gripping the pen, and a microprocessor for controlling the motors. It will also need a display to provide information on ready, finished, and percent complete. It will also need input switches for start, continue, and emergency stop. It will also need passive components for power conversion.

Stepper motors are ideal for controlled linear motion, as they give a discrete precision of how much it has rotated, which will be converted to precise linear motion. The pen gripping motor doesn't need to be precise, so it can be any cheap motor, as long as it holds the pen in place. The microcontroller can be any microcontroller with a clock fast enough to drive the motors and that can run C code. The display can be a cheap LCD display. It can be either a segmented display or a pixel display. The input switches can just be simple press buttons, they signal to software, so normal closed or normal open doesn't matter.

All these ectronic parts that we have collected and researched about will need to be collected on a circuit board after testing each one of them and knowing the specifications of each component. It is important that we know all the features of the electrical component because for a project like our project we need specific electronic parts with specific features otherwise there will be some issues that we will face in the future while working on the project.

After we finish collecting all the perfect electronic parts and test them we will have to work and design a PCB which has to be our own design for the project and the PCB will collect all the electrical components in one board. To create this kind of board we have done our research and we found different softwares that will help us on getting a workable PCB with the features that we need for our project.

3.5 Relevant Technology

3.5.1 Printers

Standard printers perform a task that is in theory similar to the pen plotter. They use ink to make marks on a piece of paper. There are several differences between pen plotters and printers. There are also several similarities, which could provide insights into how to design a pen plotter.

Printers use various methods to apply ink to paper. Inkjet printers place individual drops of ink onto the paper in a resolution too high to be noticeable on the page. Laser printers use lasers to place charges on a roller which transfers dry ink to the paper which is then fused to the paper. The inkjet mechanism is more similar to the pen plotter mechanism than the laser system. This similarity is made better by the way some inkjet printers move the printing head along the X axis over the paper.

The pen plotter puts ink to page by moving a pen over it. The action of roller ball or felt tip pens are different, but both use capillary action to pull ink out of the pen's reservoir.

Printers can use different methods to navigate the page. All standard printers use a method called rasterization to place individual dots on a page. Rasterization converts an input image into a grid of dots which all have some proportion of colors in them. This allows printers to print theoretically any image, as with a high enough resolution, the individual dots are imperceivable to the naked eye. The difference between rasterization and control point drawing is shown below in figure 13.

Unlike printers, the pen plotter must plot using control points, which make tool paths. This system theoretically has no resolution, but pens have much worse control over ink flow than printers, so the image quality will always be worse than a high end printer when aiming for photorealism.

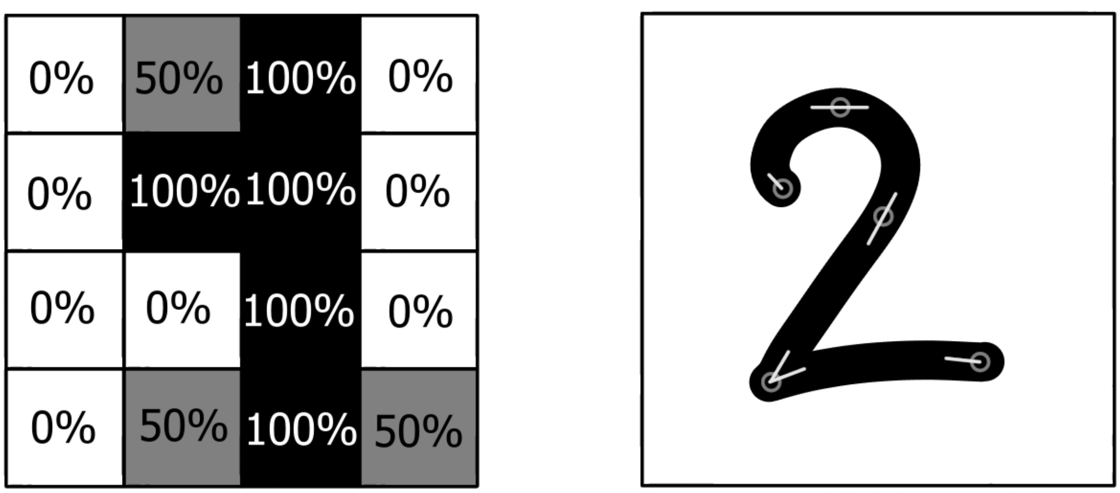


Figure 13: Rasterization Vs Control Points - Rasterization of some image (left). An image made with control points (right).

A secondary effect of rasterization is that it allows printers to print a page one row at a time. This means they never have to move up a page, which simplifies the mechanism. Some inkjet printers move the printing head across the page for each line, meaning they move in the X and Y directions, like a pen plotter, and others have several printing heads with enough horizontal resolution to get a good image.

The pen plotter will definitely need to move up and down the page, and left and right across the page. It will also need to lift the pen off the page between strokes. These actions are fundamentally different from how a printer prints.

The purpose of a printer is to create a physical copy of a page of text or an image, which it excels at. Pen plotters do not attempt to compete with printers for this use case, instead being used for artistic uses, such as creating complex or precise strokes with a pen that would be difficult to do by hand.

Pen plotters also have a theoretical advantage of pens being much cheaper to replace than printer ink, and are relatively much easier to construct from scratch, making them ideal for anyone who doesn't want to pay money to major printer sellers.

Printers are capable of printing nearly any color by mixing black, cyan, magenta, and yellow. This allows printers to make full color illustrations or photos without needing separate ink colors for every perceivable color. Pen plotters, however, can only put ink to page in colors that pens are available in. Colors can't be mixed in the same way as with printers, so the pen plotter must use a different pen for each color in the drawing. There are shading techniques that could allow colors to appear to be mixed to an observer, but these are not as effective because the line width of any colored pen will be too thick to produce a perfect illusion to any observers.

Overall, pen plotters appear to fulfill a similar purpose to printers, but in reality they serve two completely different purposes, and so use completely different methods to accomplish them. The purpose of a printer is to produce physical documents or images with the highest quality and reproducibility. This requires printers to use very high precision methods to control how ink is placed on the page. Since printer's use rasterization with a very high resolution, they can create nearly any image, even photorealistic images, even with multiple colors by mixing colors. Pen plotter, on the other hand, are used to create drawings in a way similar to how a human would do it. This makes it more of a novelty, and makes it entertaining to watch. Pen plotters can also produce images with more precision than human hands, and could produce the same strokes consistently. Pen plotters, however, cannot modulate how much ink they are putting on a page, so they cannot create images with the same fidelity of a printer. Pen plotters can also only use colors which pens exist in, and cannot mix colors very well. In these ways, pen plotters can't compete with printers for the niche of consistent fast documents and color photos, but they can have their own niche as artistic robots that produce drawings while being an entertaining mix of human-like and machine-like. Still, pen plotters could inspire some of their mechanisms from those of traditional printers.

3.5.2 3D Printers

3D printers share several similarities to pen plotters. Both move their working end through three axes, both mainly make artistic pieces, and both are mainly owned by hobbyists, though 3D printers have more practical uses outside of hobby projects.

3D printers work by heating up plastic filament and placing layer after layer of plastic onto each other to create a whole 3D object. Pen plotters work by dragging a pen across a piece of paper to draw lines on it to create a final 2D image.

While 3D printers need a full Z axis, Pen plotters can get away with a binary on or off the page, since it doesn't matter how high the pen is above the page, it'll stop writing anyway. More Z control does however give better control over line width when using felt tip pens.

A 3D printer's working end is permanently attached to the device, so there is no mechanism analogous to the mechanism a pen plotter needs to switch pens out. The pen plotter also has no need for the 3D printer's heating mechanism, since its pens work by capillary action.

While a 3D printer needs to create each layer on top of each other, even needing to make removable supports in some cases, the pen plotter can make its strokes in any order. This makes it harder for a pen plotter to completely fail a drawing compared to 3D printer, where many are known to fail a print if the model wasn't supported enough or they shook while printing.

3D printers need much more complexity than a typical pen plotter on account of them producing 3D objects out of melted and resolidified plastic. Pen plotters don't need all of that complexity, so it could be overkill to copy designs from a 3D printer. For example, most 3D printers use a gantry mechanism to hold the printing head. This is necessary in the case of 3D printers because the printing head is heavy enough that it would weigh down a single axis too much to be stable. The gantry mechanism allows it to keep the weight, but distribute it over two rails instead of one. This keeps the printing head stable at all times. This also leads to further complexity, where specific pulley mechanisms are required in order to keep the two rails synchronized. This creates even more complexity, because that requires extra math to be done in order to move both axes by the right amounts. All of this complexity is eliminated in the pen plotter by keeping the drawing mechanism light, which keeps it from causing the Y axis rail to sag too much. It also helps that the Z position of the pen is not very important. It only matters if the pen is on the page or off of it in most circumstances, so even if the Y axis rail does sag a small amount, it will still keep the pen off of the paper.

Overall, 3D printers appear at an outset to be much different devices from pen plotters, but their mechanisms are surprisingly similar. Both mechanisms make use of an XY mechanism to move either the printing head or the pen tip across an area. Both devices also use rails and pulleys to construct that mechanism. Both devices move their printing head or pen tip up and down on the Z axis. 3D printers need a precise Z axis, and in most cases a much taller Z axis than the pen plotter, which only needs enough precision to get the pen up off the page, or down on the page. Both devices also serve similar purposes in consumer use. They both produce pieces of art in novel ways. In the case of 3D printers, this art can also be useful in some circumstances, such as 3D printing a clip that fits on a camera to be able to mount it to a tripod. Most consumers of 3D printers instead produce small pieces of art, often with moving parts, which 3D printers excel at, and which no other devices can really be compared to. Pen plotters, however produce traditional pen and paper art in a similar way to a human. This makes the end result much less unique to a 3D printer, as pen plotters cannot do anything fundamentally different from a person holding a pen. They do use more precision than most amateur artists, so they would probably be most impressive when creating precise patterns such as lattices or spirals. Despite the similar, but different uses each device has, they use very similar mechanisms to one another, and so the 3D printer's mechanisms and protocols can be used to inspire the mechanisms and protocols that go into this project.

3.5.3 CNC Milling Machines

Computer numerical control (CNC) milling machines are similar to both 3D printers and pen plotters. CNC milling machines also use three axes, but most do not move the working end through all three of them. Most CNC milling machines move the work space around the working end in the X and Y axes and only move the working end in the Z axis. This is much easier to do than move the working end in all 3 axes because it is spinning at high speeds and much heavier than the working end of either a 3D printer or a pen plotter. The different ways the three devices move through the three axes is shown below in figure 14.

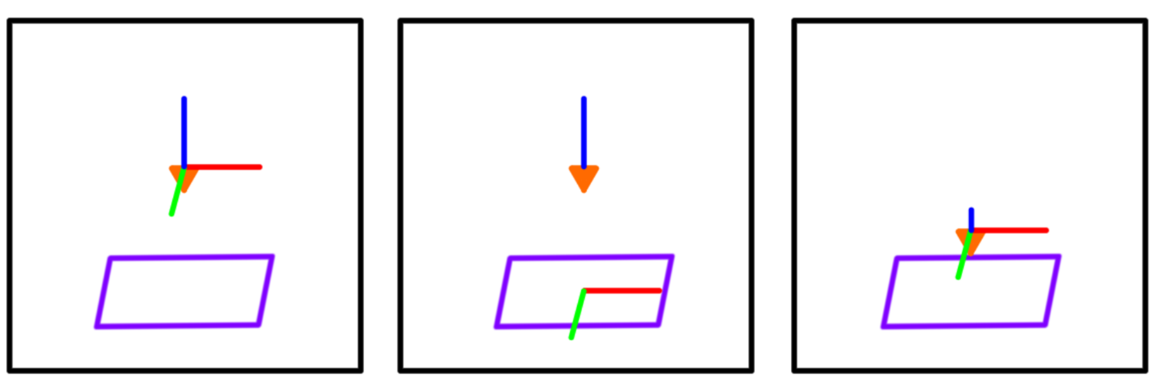


Figure 14: Motion of a 3D Printer, a CNC Milling Machine, and a Pen Plotter - Motion of a 3D printer (left). Motion of a CNC Milling Machine (center). Motion of a pen plotter (right). Purple is the working area, orange is the working end, red is the X axis, green is the Y axis, and blue is the Z axis.

CNC milling machines work by removing layers from a starting block of material, often metal. They accomplish this by spinning a tool at a high speed and moving the block across the tool in order to remove material where needed.

CNC milling machines often need to move the working end up between each stroke across the material because unlike a 3D printer, it has to move down the part, not up. This means it has to work around all of the material it is leaving in place in the higher parts. This is even more complicated to compute than a 3D printer's removable supports. Thankfully this task is precomputed in software, since it would be hard to handle in real time.

Many CNC milling machines use replaceable tool tips in the working end. This is similar to a pen plotter, which can also switch pens when desired. Some CNC milling machines also can switch parts automatically, as part of the programs they run.

All three devices move the working end through the work space (or the other way around) with an XY mechanism. All three devices also use microprocessors to move the working end along a predetermined tool path which was generated by software.

Both work by producing a file on a computer, which when transferred to the device will move the working end in such a way to create the desired results in real life.
Unlike pen plotters, CNC milling machines have real use in industries that need precision machined parts. This means that the technology needed to run them is matured, which gives great advantages to the project, since the pen plotter can make use of the existing technologies and standards behind CNC machines.

The technologies behind CNC milling machines also extend to other machines that could benefit from precision automation. They also extend to any number of dimensions or rotations, as long as the computer tells the machine how much to move in each dimension, the machine can create the desired part.

At a glance, CNC milling machines appear to be very different from pen plotters. While a pen plotter puts ink on a piece of paper, a CNC milling machine takes sections of material off a block of starting material. They are very similar, however, in the way they produce their final products. Both devices use X, Y, and Z axes to move their tool tip or pen tip around the block of material or piece of paper. Both devices use control points to follow tool paths in precisely the way they need to to produce the end produce. The use case for consumers of these two devices is completely different most of the time. While it is possible to make small pieces of art with a milling machine, it is uncommon as milling machines are very expensive, and so most consumers would only buy one if they stand to make a profit from owning it. They do this in most circumstances by producing parts they need for projects that are worth money. This means that most of the time, CNC milling machines are used to produce highly specific parts for use in important projects. Pen plotters, on the other hand, are almost always used to make frivolous pieces of art, which will never be needed again. Surprisingly, despite the different needs of each user, both devices use similar principals to create their final products. This means that CNC milling machines and their mechanisms, protocols, and standards could stand to be great sources of inspiration for this pen plotter project's own mechanisms, protocols, and standards.

3.5.4 Laser Engraving Machines

Laser engraving machines are similar to pen plotters in both purpose and function. Some of the mechanisms behind them could inspire mechanisms for the pen plotter.

Both devices work in two dimensions, creating a two dimensional final product. The pen plotter has to move the pen up and down, but the laser engraver can engrave the material from a distance, since it uses a laser. Both devices use an XY mechanism to move the working end above the working area. The pen plotter needs an extra Z dimension, which the laser engraver does not need.

Laser engravers can be dangerous. The laser diode can get hot, so it needs to be cooled. The laser's beam could damage anyone arounds eyesight if they look directly at it without proper eye protection. The laser could also start a fire if the moving belts slip, causing the laser to be focused on a single location for too long. The laser could also burn anything that gets under it, including whatever is under the thing being engraved if it goes off the edge.

The code running the laser engraver also has to account for modulating the laser's strength, since some applications of the laser engraver require more than two levels of engraving.

Many laser wood engravers can print photos onto the wood, using rasterization and multiple laser levels to get a photoreal image on the wood. This is very different from how a pen plotter works, since pen plotters cannot modulate how much ink the pen is putting on the paper. A pen plotter also cannot reach the same precision as a laser engraver because the line width of a pen is much greater than that of a laser beam.

Many commercially available laser engraving machines can also work to cut wood. This uses the same mechanisms as engraving, but it burns its way all the way through the wood, creating separate pieces that can be separated and constructed into something three dimensional.

Laser cutting must be done using control points, not rasterization, so it is much more like the pen plotter drawing lines than the photo mode. Unlike the pen plotter, it rarely ever creates crossing lines in this mode, and it mostly only makes complete cutouts.

Laser engravers can also work on a variety of materials. It is fairly common to have jewelery customized by having it laser engraved. This process is often done by hand, though laser engraving is also an option. Laser engraving for jewelery is sometimes done with mirrors rather than moving the laser diode around so the beam can have more precision and can engrave areas that require harder to reach angles. The use of mirrors is not analogous to the pen plotter, but there are some effects that could be achieved by angling the pen differently.

Despite laser engravers being more dangerous than pen plotters, there are several tutorials online teaching people how to make them at home. This may be due in part to the fact that people can make models out of laser cut pieces of wood. This is more novel to the average person because there is no direct substitution for it, unlike the pen plotter, which could be replaced by a printer or by hand drawing with a pen.

While laser printers can directly share an XY mechanism with a pen plotter, all of the other mechanics are different. The firmware can be similar, with the ability to make strokes on wood using control points, though most applications of laser engravers use rasterization instead, since it is more efficient at filling in areas and has a high enough precision that nobody will notice the individual dots.

3.5.5 Embroidery Machines

Embroidery machines are another example of a device that converts digital patterns into 2D images. They perform the task of stitching in a pattern automatically by using computer generated motion paths.

Similarly to a CNC milling machine, the embroidery machine moves the material in the X and Y dimensions. This is because the mechanisms needed to create stitches through the fabric need to be above and below the material, so it is much simpler to move the material instead.

The embroidery machine also does not really have a Z axis. The needle does move up and down through the material, but this is not precisely controlled by a computer, and is instead run similar to a standard sewing machine, where the entire motion is controlled by a series of gears and mechanisms that keep all of the stitching mechanisms in perfect alignment.

The embroidery machine does control how fast the stitching mechanism runs, which is not something the pen plotter needs to do, since the pens used have no mechanisms to control. This is theoretically similar to a CNC milling machine's control over its tool rotation speed. Tool rotation speed is less often tweaked mid pattern though, since most of the time the only concern is if the tool is running fast enough for the given material and tool path speed.

Fabric is similar to paper, because it is flexible, and it would be a problem if it wasn't held flat the entire time. Embroidery machines solve this problem by using an embroidery hoop, which holds the fabric taught the entire time. This allows the needle to move up and down without pushing or pulling on the fabric. This method could inspire a mechanism for the pen plotter, allowing it to hold the paper still while the pen moves on it, keeping it from being pushed around or folded. An embroidery hoop couldn't exactly be used, because it would cause sharp folds in the paper which would permanently harm it. A different but similar method could be used, however, which is to clamp down the paper at its edges. If a clipboard were used, this would cause the paper to still bunch up when moving up the page, so it would have to have clips at both ends. For this purpose, binder clips could be the best choice to clamp the paper down, because they are cheap and readily available, and they can clamp the paper without causing any permanent damage.

Embroidery machines can be made to support multiple colors, similar to pen plotters. Also similar to pen plotters there are multiple ways these colors can be supported. The first way is to do one color and then prompt the user to manually switch to the next color. This is similar to the simplest color switching method for pen plotters. Another way embroidery machines can sew patterns with multiple colors in it is to use multiple needles, one per color. The device can switch between needles at will, so it doesn't need to wait for the user to switch colors. These needles can't all be in the same place, so it needs to know the offset of each color needle compared to the first one. This could be implemented into a pen plotter, allowing multiple colors to be used without even needing to pick up or put down any of the pens. The method does, however, require a way to select which needle, or which pen, to be used at any given time. With the embroidery machine, it can move the head which holds the needles by the required amount, and the new needle to be used will line up with the mechanisms needed to run it. This allows the same mechanism to run several needles, but only one at a time. Theoretically, a similar system could be implemented on a pen plotter, but it would require a new slider for the pen chooser and would need a way for the pen lowerer to only effect the chosen pen, which could be designed, but would add further complexity and more failure points to the design. This design would also require all of the pens to be uncapped at all times, drying them out even while they aren't being used.

Overall, embroidery machines may be the most similar to pen plotters of any device examined here because they perform a task that is possible to accomplish by hand (embroidering a piece of fabric or using a pen on paper) but automate it to allow it to be completed faster and with more precision. Both are also used for artistic purposes. In the case of embroidery machines, that purpose might be more permanent, because it can be applied to clothing or blankets that are used every day, but pen plotters just write on paper, which at best can be framed and hung. The mechanisms are also similar, because both make use of an XY mechanism to move the needle or pen tip around the fabric or piece of paper.

3.6 Relevant Mechanisms

3.6.1 XY Mechanisms

The machine needs to move the pen across the paper in the XY plane. There are several ways this can be accomplished. 3D printers and CNC milling machines both perform similar tasks to pen plotters, so they could be good places to look for inspiration.

3.6.1.1 XY Table

A CNC milling machine moves its tool through the part's XY plane by moving the table the part is attached to. XY tables are usually driven by screws as long as the translation axes, which have no chance of slipping, but are expensive and relatively heavy. They move the part instead of the tool because they require the tool to be spinning at high speeds, which makes it more stable to move the part around the tool than the other way around. The machine needed to spin the tool at high speeds is also very heavy, so it would be more difficult to move in the XY axes than the part. These are not concerns shared by the pen plotter, because the pen won't spin, and won't be very heavy. This would also require the Paper to move through an area twice as large as itself, which would require the pen plotter to be much larger than other methods.

3.6.1.2 3D Printer Belt Mechanisms

A typical 3D printer moves its printing tip through space in the XY plane by using a gantry system. A gantry system moves the Y axis across two parallel X axes. This supports the Y axis on both sides, preventing the weight from pushing down on the axis, lowering the printing tip. This is a valid concern because the weight is fairly significant. They will usually use either H-bot or CoreXY mechanisms.

3.6.1.3 H-bot mechanism

H-bot mechanisms are gantry systems that work by connecting two motors to a single belt. The mechanism is represented in Figure 15. The belt is attached to the printing tip's carriage. The printing tip's carriage rides along the Y rail. The Y rail rides along the two X rails, one at each end. When one motor rotates, the printing tip's carriage moves diagonally. Both motors need to be moving to make the device move on the X or Y axis alone. This makes the motor programming slightly more complicated, but is solved with a simple equation. There is a problem with the H-bot mechanism, which is that it applies uneven torque to the system, causing it to skew slightly.

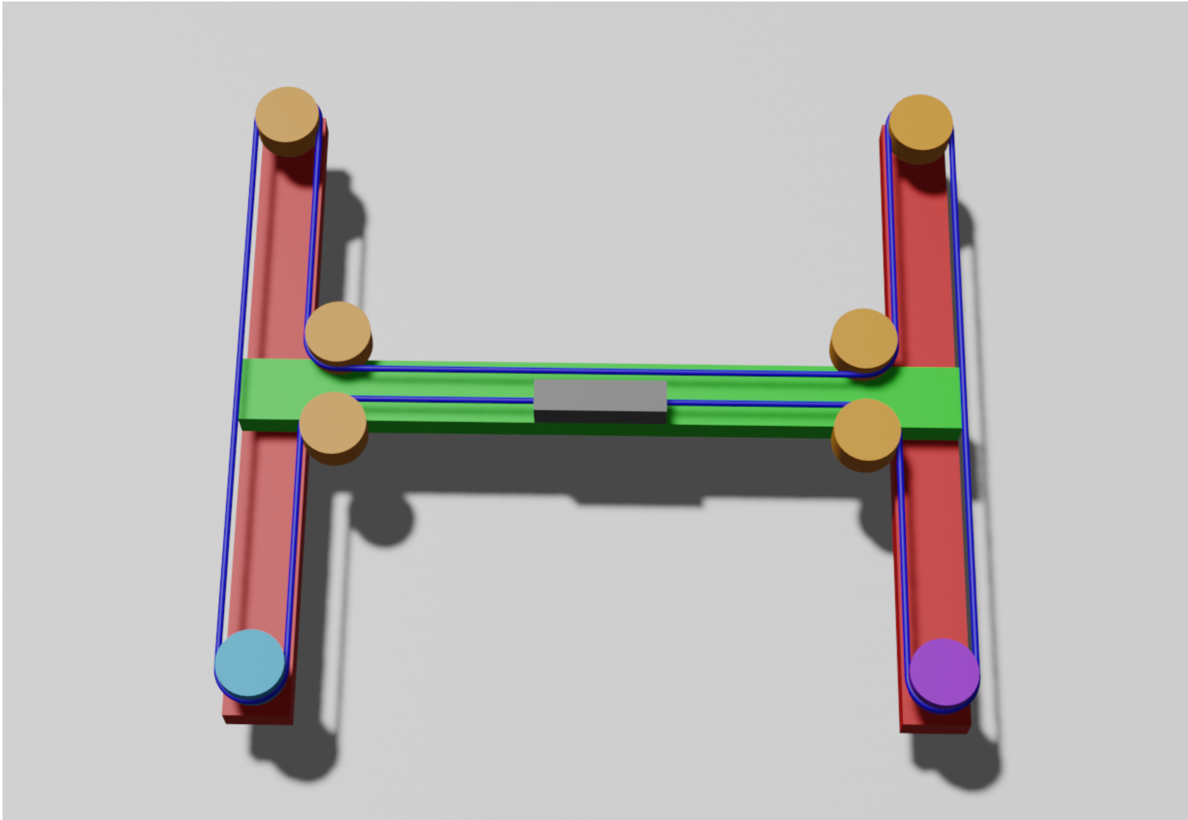


Figure 15: H-bot Mechanism - 3D representation of the H-bot mechanism. Red are the X rails, green is the Y rail, gray is the carriage, orange are the passive pulleys, blue is the belt, and aqua and purple are the motor driven pulleys.

3.6.1.4 CoreXY Mechanism

CoreXY mechanisms are also gantry systems. The mechanism is represented in Figure 16. CoreXY however uses two separate belts, driven by two motors. Both belts are attached to each side of the printing tip's carriage. The printing tip's carriage rides along the Y rail. The Y rail rides along the two X rails, one at each end. Similar to H-bot mechanisms, CoreXY mechanisms also need both motors to be running at the same time in order to move on the X or Y axis. It needs a similar equation to convert the X and Y motion into motor speeds. CoreXY solves the torque problem of H-bot, which prevents the frame from skewing.

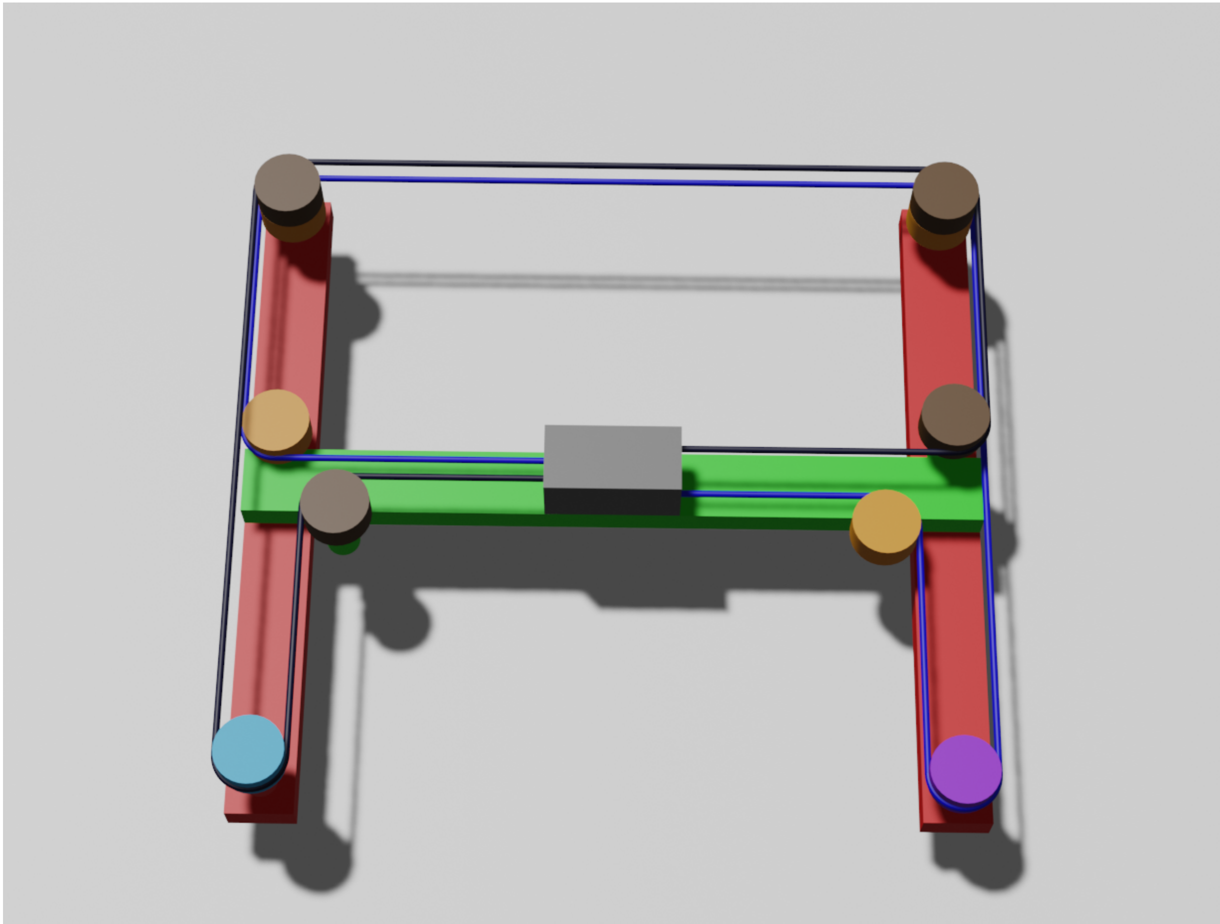


Figure 16: CoreXY Mechanism - 3D representation of the CoreXY mechanism. Red are the X rails, green is the Y rail, gray is the carriage, orange and brown are the passive pulleys, blue and dark blue are the belts, and aqua and purple are the motor driven pulleys.

3.6.1.5 Delta Mechanism

There are also more novel 3D printer mechanisms that could be taken for inspiration. One such design is the delta mechanism. The delta mechanism works by connecting the printing tip's carriage to three separate vertical axes. It uses much more complicated trigonometry to move the printing tip in 3D space. It is much more suited for vertical builds, as it can theoretically be extended vertically indefinitely without making the system more complicated.

3.6.1.6 Polar mechanism

Another novel 3D printer mechanism is the polar mechanism. The mechanism is represented in Figure 17. The polar mechanism works by rotating the platform and moving the printing tip along a linear rail from center to outer axis. This is better for situations where little working area is available, as it only needs extra clearance on one side.

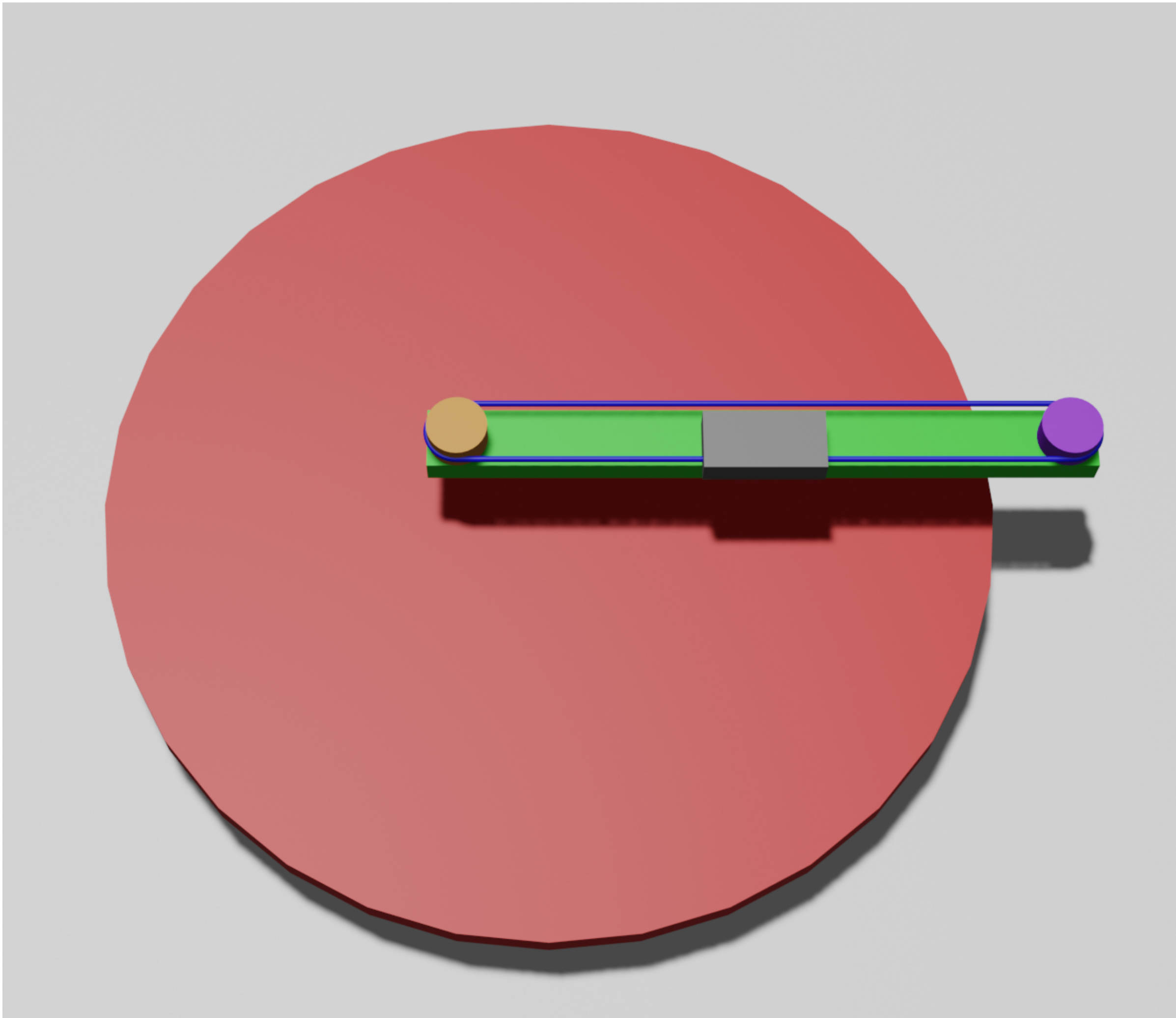


Figure 17: Polar Mechanism - 3D representation of the Polar mechanism. Red is the rotating platform, green is the radial rail, gray is the carriage, orange is the passive pulley, blue is the belt, and purple is the motor driven pulley.

3.6.1.7 Linear System

Other pen plotters can also serve as inspiration for XY mechanisms. The linear system is a simple system that only uses two linear rails. The mechanism is represented in Figure 18. The pen lifter rides on the Y axis rail. The Y axis rail rides along the X axis rail. The Y axis rail is only supported on one side, so to help balance the weight, the Y axis' driving motor is on the other side of the X axis. The machine is also supported by a wide base, preventing it from toppling over. Theoretically the movement along the Y axis will change the balance of the machine to lean the whole Y axis downward, but the exact height of the pen doesn't matter that much, since it just needs to be on the paper or above it, small height differences don't matter. The system is driven by two independent belts. One for the X axis and one for the Y axis. Having the Y axis unsupported on one end also decreases the needed clearance of the machine. There is also a nice aesthetic benefit to the appearance of the machine, reaching out over the paper, which makes it look subjectively better, more like a human, than other designs that just move around the inside of the system like a robot. The aesthetic benefits to the design are to be considered, because the novelty of a machine drawing like a human is the point of the project.

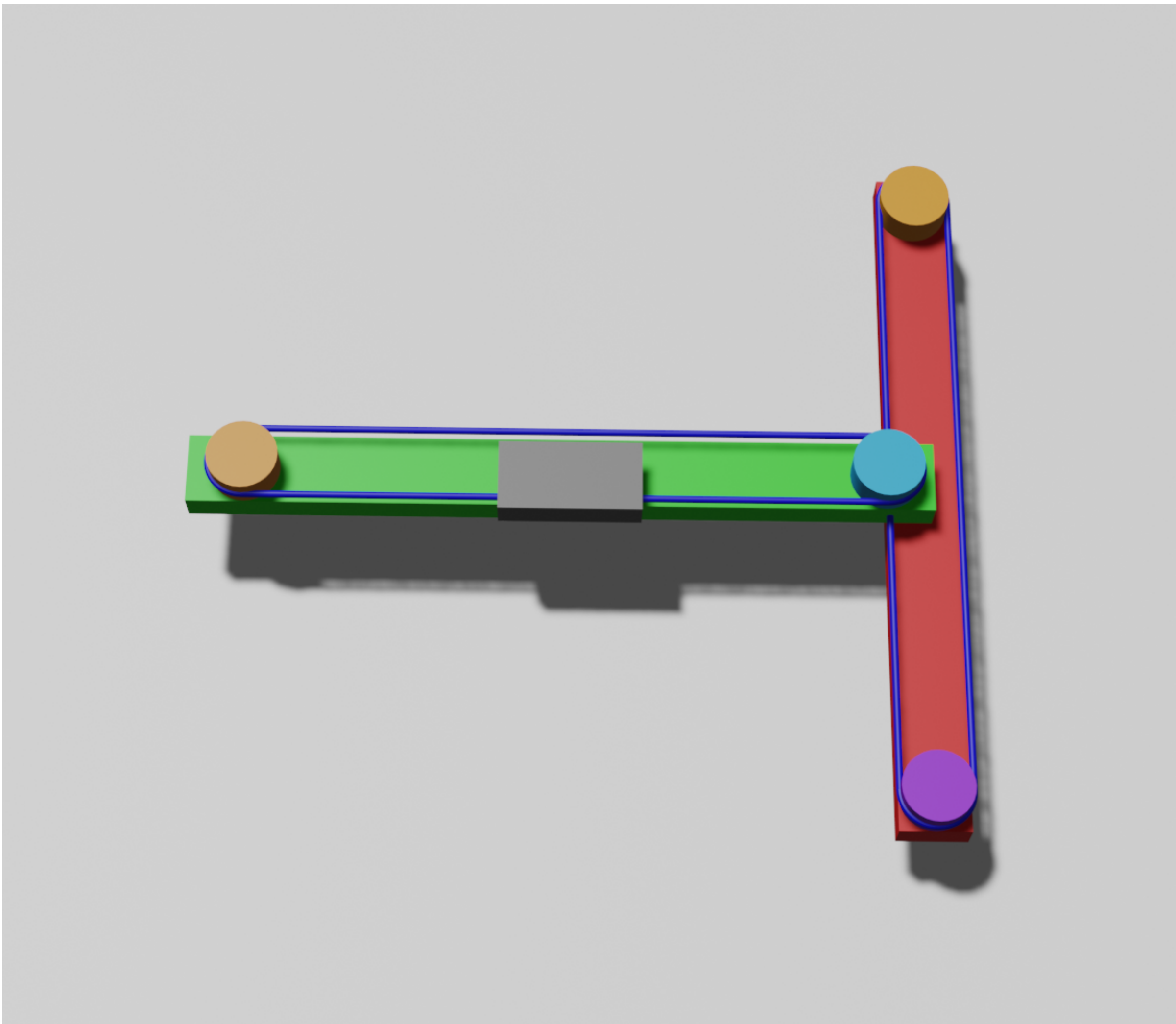


Figure 18: Linear Mechanism - 3D representation of the linear mechanism. Red is the X rail, green is the Y rail, gray is the carriage, orange are the passive pulleys, blue are the belts, and aqua and purple are the motor driven pulleys.

3.6.1.8 Conclusion

Overall, the best choice for XY mechanism is likely the linear mechanism. This mechanism both simplifies the construction and implementation of the XY mechanism by a lot, but it also simplifies the firmware of the pen plotter, as the linear mechanism doesn't require any extra math to figure out how the motors need to be driven in order to move the pen where it needs to go. This also massively simplifies testing of the device, since the axes can be tested independently, and any problems with a single motor or axis will be easily identifiable. This is also the most aesthetic choice, as it appears to be reaching over the paper, unlike any of the gantry mechanisms which need a rail on both ends of the paper.

3.6.2 Z Mechanisms

The machine needs to be able to lift the pen off the paper. There are several ways to do this. Inspiration can be taken from 3D printers and CNC milling machines. The most important qualities in a Z mechanism are low complexity, lower chance of failure, and appearance.

The first possibility is lifting and lowering the platform the paper sits on. This is a good idea when the XY mechanism is too complex to lift up and down stably. This is a problem for the pen plotter, however, because the paper will be resting on the table.

The second possibility is lifting and lowering the entire XY mechanism. This is a good idea when the machine is a 3D printer, where the thing being printed might be fragile, so they shouldn't be moved, and the printing head has to move a lot on the Z axis.

The third possibility is lifting and lowering only the pen. This is a good idea when the pen doesn't have to move up and down very much, which is the case in the pen plotter. It is also much simpler than lifting the whole XY mechanism, and needs to lift much less weight. This, however, requires a way to convert a motor's angular motion to linear motion. This also adds to the weight being put on the XY mechanism.

The fourth possibility is angling the pen up and down. This is much simpler to implement than lifting the pen up and down, because it can easily be done with a step motor. There is no need to convert from linear to radial motion. This, however, would likely leave streaks on the paper wherever the pen is put down or lifted up.

Out of these possibilities, lifting just the pen up and down seems like the correct choice for this project. It seems to add the least complexity without compromising the quality of the drawing. It also looks the most like a human, so it is the most aesthetic choice among them.

3.6.3 Pen Holding Mechanisms

3.6.3.1 The Grip

The machine needs to be able to hold the pen and move it up and down, left and right, and forward and back without the pen falling out or slipping. There are two main ways of doing this, vertical force and friction. Ease of manufacturing, ease of use, and price are main concerns for this mechanism.

Vertical force mechanisms work by holding the pen from above and below. There would have to be a hole in the bottom of the clamp, allowing the pen to fit through it. This would be able to hold a good assortment of pens, but the vertical clamp would have to be large enough to fit the pen into. The hole at the bottom of the clamp would not be able to fit every pen. The clamp would have to be very tall, and likely specific to the pens being used. The main benefit of vertical force mechanisms is that they don't depend on friction, meaning they work in any condition, regardless of moisture or temperature or being worn down over time.

Friction mechanisms work by holding the pen from the side, usually clamping it in place to get higher static friction forces. Clamping from the side can fit many different sizes of pen, with no need for holes in the clamp, and no need to extend to the length of the pen. A friction mechanism can hold just ~3 centimeters of the pen and have good enough control over it to not slip at all. Friction mechanisms, usually also make use of some higher friction material on the clamping surfaces to improve its grip. This can add some cost, but greatly improves the clamp's efficacy.

There is another possible mechanism, which is to modify the pens being used to make rigid connections with the device, such as drilling screw holes through the pen, and mounting it directly to the device. This makes the pen difficult to replace, and permanently alters the pen being used, so it is not an ideal choice.

Modified pens to fit the device could be used to make a standard for pens for a given pen plotter in order to force the consumer to buy pens exclusively for the pen plotter, but also to improve performance of the pen. This could theoretically improve the function of using the pen to draw and also the automation of switching pens, by cleverly designing a grip that can be enabled or disabled by just clicking it onto the pen and then clicking the pen back into its stand.

The friction based clamp design seems the best fit for the device, as it is easy to design and fits a wide variety of pens. Making pens that specifically fit into the device would most likely give the best possible performance, but it would be difficult to implement and design, and it would also limit how the device is used, by not allowing any normal pens to be used.

3.6.3.2 Controlling the Grip

If the device is to independently switch colors, then the grip must be controlled by a motor. This motor needs to somehow control a mechanism that applies a clamping force while closed and provides no force, or a clamping force in the opposite direction, into a stop, while open.

A large amount of clamping force is not required to hold a pen while writing, especially if a high friction surface were used to grip the pen. Because of this, the grip could be held closed using just the torque provided by the motor. This would require constant power, but it would easily hold the pen in its grip.

If a significantly smaller motor were used, it could maybe be geared down to increase its torque at the expense of speed of pen switching. At the extremes of small motors, this would either be way too slow for the device to reasonably switch pens automatically, or the motor wouldn't be able to even drive the gear train needed to gear it down far enough because the gear train itself would require too much torque.

If constantly requiring power is seen as a concern, a mechanism could be devised such that the grip could latch closed in such a way that the pen would not be able to slip out of the grip and the grip would be held closed, without power from the motor, until the motor unlatched the grip. This would be a problem in most cases, because it would be hard to design a latching mechanism that works for any size of pen and doesn't need constant power, as that would likely need either a ratcheting mechanism, which would be hard to undo with the same motor that drove the ratcheting mechanism, or a screw mechanism, which would make the grip very slow to open or close.

Overall, the power draw from the motor is not a big enough concern to design a novel clamping mechanism, and there's no reason to overcomplicate the mechanism with a gear train which could slip or break. For these reasons, the best approach is likely to just use a big enough motor to constantly close the grip, preventing the pen from slipping out, even if it does draw more power than the more complicated mechanism.

3.6.4 Paper Holding Mechanisms

The device needs to hold the paper down in order to write on it without moving it. There are several ways to accomplish this. Weight, clamping, fastening, and adhesion are all possibilities to keep the paper in place while the pen moves around on top of it. Ease of manufacturing, ease of use, usable work area and, reusability are all concerns for this mechanism.

Using weight to hold down the paper would be the simplest method to implement. A weight could be placed at each of the four corners of the paper. With enough weight, this would hold the paper in place as the pen writes on it. This would take up space on the paper, which the pen wouldn't be able to write on. The device could be made to write what it can on the available area and then wait for the user to move the weight inward so the device can write on the newly uncovered area, but that would make it harder to use and harder to design the firmware. Weights are also not failproof, as the user could place the weights in the wrong places, causing problems when the pen collides with them. Weights also don't provide an alignment system, so it could be difficult to align the paper to the device's coordinates.

Clamping the paper to something would be simple to use. It could be done with a binder clip at each corner of the paper. It would have to clamp on to something, so the device would need a board under the paper to clamp on to. This could be a clip board or a smooth piece of wood. This would also cover part of the paper, but less than the weights, since the clamps can produce more force than the weights while covering less surface area. Most artistic and professional uses of pen and paper leave a margin on the page, so it is unlikely the covered area will be detrimental to the usability of the device. Clamps like binder clips will theoretically stop being usable eventually due to metal fatigue, which makes them less reusable than weights, but their usability is so long, it will likely outlast the electronics in the device. Clamping the paper to a board attached to the device could provide easy alignment for the user, as the device will know where the corner of the board is. Clamps are also harder to misuse than weights, as they cannot be used anywhere except for the edge of the paper.

Fastening the paper to something is also a possibility. One example is using a push pin to fasten the paper to a cork board. This would effectively hold the paper in place, but it would also leave a hole in the paper, which would be undesirable in most cases. Theoretically, the paper could have additional parts off the edge of the used area which are made to be fastened down, similar to the pin feed paper used in old dot matrix printers. This would allow the parts of the paper with holes to be removed afterward, but it would also require specialized paper, which would make it more difficult to use. While corkboard is a bad example, since it would be difficult to write on, all forms of fastening share the problem of putting holes in the paper. Other fastening devices are brass paper fasteners, staples, ring fasteners like in a 3 ring binder, and one could also use nuts and bolts to clamp the paper through a hole in it. The example of pin feed paper, would theoretically provide great alignment for the user, and would have the highest reusability of any fastening method, but again would require special paper, which would place an unnecessary burden on the user.

Adhesion could also a possible way to hold the paper in place. Tape could be placed over the corners on top of the paper, or double sided tape could be placed under the paper. Both methods would hold the paper in place, but would have different side effects. Tape on top would cover the corners, making them unusable. If the tape used is too sticky, it would also harm the paper when it is taken off. Double sided tape under it would also harm the paper if it is too sticky, but that would be less noticeable because it is on the back. The tape would also cause a bump in the writing surface, which could cause the pen to warp as it goes over it. Tape would also be difficult to use perfectly, as it could cause the paper to not be flat if applied even slightly wrong. Tape is also disposable and would need to be replaced often, giving it the worst reusability of the methods.

Overall, the best method seems to be clamping the paper to a board attached to the device. This provides alignment, covers little of the margins of the paper, is easy to use, and is reusable for longer than the device itself.

3.6.5 Linear Rails and Carriages

The device will use linear rails and carriages to move the arm in the X dimension and to move the head in the Y dimension. There are several options for linear rail designs. These rails can have positive grooves, negative grooves, or no grooves at all. They can be square, or they can be round. They can be singular, or parallel. They can be supported, or free floating. There are also several options for carriages. They can use wheels, or low friction. They can ride on top of the rails, along the outside of the rails, along the inside of the rails, under the rails, or a combination of multiple of these. The type of carriage used will complement the type of rail being used. The main concerns for the linear rails and carriages is that the rail is strong enough to support the carriage and all connected parts' weights, the carriage must move smoothly along the rail, it must be available, and it must stay attached during the devices intended usage.

The first type of linear rail one might think of is a train track. Trains ride on train tracks by using wheels that ride on top of the tracks. There is also a clever design to train wheels and tracks. They are grooved such that even though the wheels can drift right or left, they are pushed by the inside of the track to align with it. This type of linear rail is not practical for our use because it can't withstand lateral forces, which our Y axis rail will need to do. A train style linear rail is shown below in figure 19.

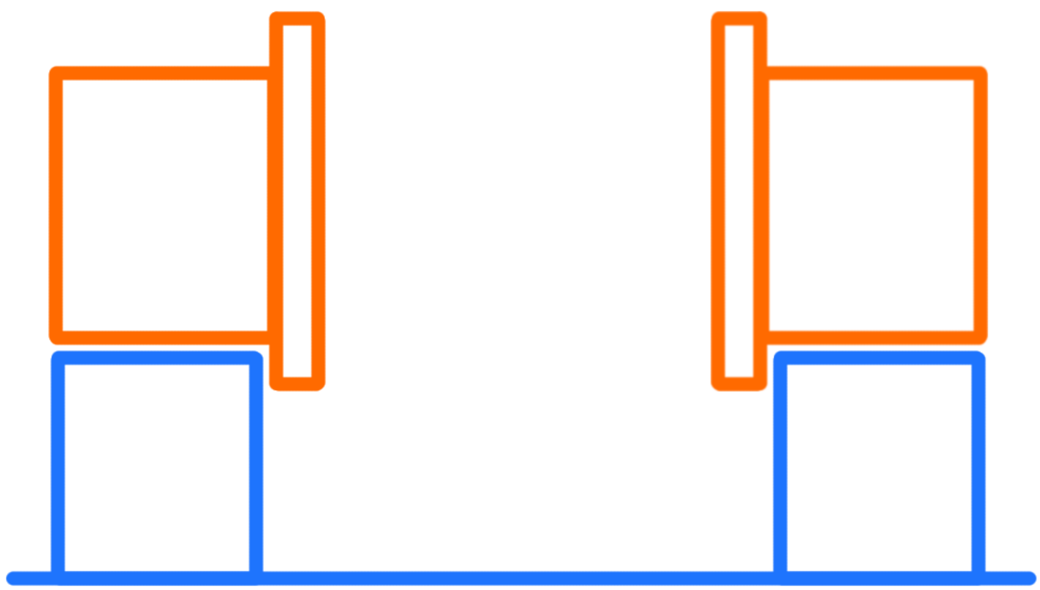


Figure 19: Train Style Linear Rail - Profile of a train style linear rail. Orange is the wheels of the carriage. Blue is the rail.

The next type of linear rail one might imagine is a roller coaster. There are several kinds of roller coasters, with the main mechanisms being rectangular mechanisms and circular mechanisms. Both mechanisms grip the rails in such a way that lateral forces and even inversions are not a problem. The carriages, however, are optimized for keeping stable on curved tracks, which means they need more space so they can fit around the bend, which makes them less stable on straight tracks, so they aren't the best pick for straight tracks. A roller coaster style linear rail is shown below in figure 20.

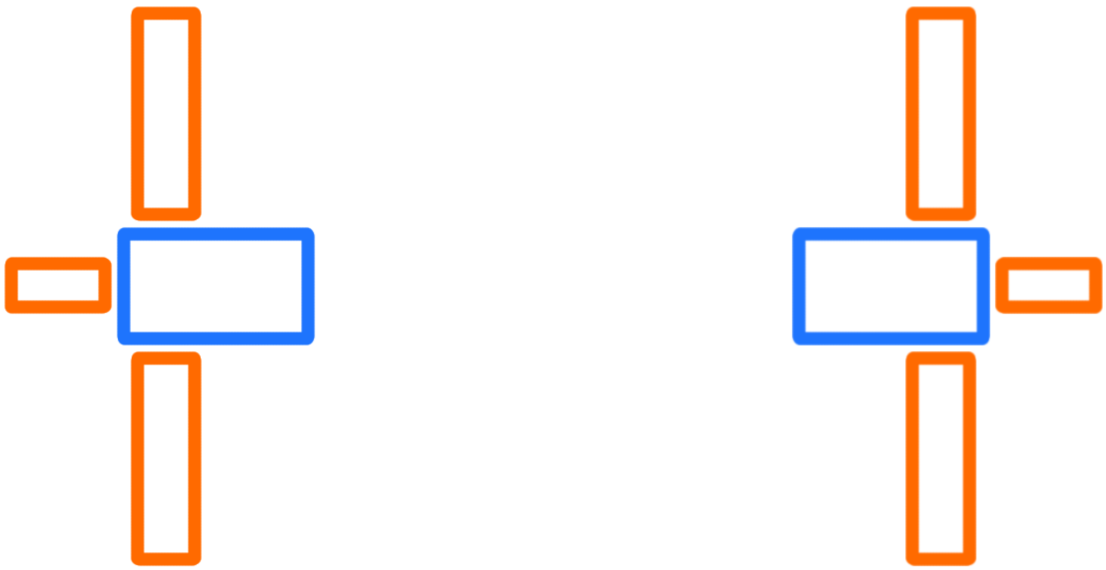


Figure 20: Roller Coaster Style Linear Rail - Profile of a roller coaster style linear rail. Orange is the wheels of the carriage. Blue is the rail.

Square rails are common and readily available linear rails. They come in several varieties, including ones with the carriage riding on top and riding on the outside. Riding on top has the same problems as train tracks, any lateral force is likely to dislodge the carriage from the track. Riding on the outside, the carriage must still rest on something, so it usually rides on the groove or outer rail of the square. It usually uses a special wheel that fits the rail from above and below, which prevents it from coming off when experiencing lateral forces. Some carriages also use ball bearings or low friction surfaces rather than wheels. These carriages are still fit specifically to the rail, allowing them to also stay on the track. An outside ridden square linear rail is shown bellow in figure 21, and an inside ridden one in figure 22.

Figure 21: Outside Ridden Square Linear Rail - Profile of an outside ridden square linear rail. Orange is the wheels of the carriage. Blue is the rail.

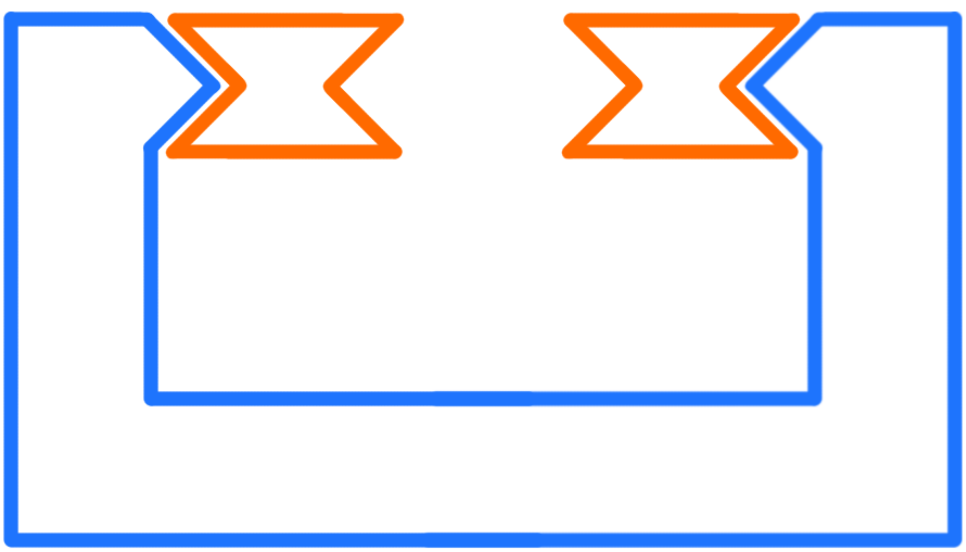


Figure 22: Inside Ridden Square Linear Rail - Profile of an inside ridden square linear rail. Orange is the wheels of the carriage. Blue is the rail.

Round rails are also available and may be simpler to construct from scratch, as they can be just featureless cylindrical rails. A carriage usually must sit on two rails, because it would freely rotate around a single rail. Round rails are harder to use because they need to be machined to be able to be attached to anything, since they are so smooth they can't be easily attached to anything. A round linear rail is shown below in figure 23.

Figure 23: Round Linear Rail - Profile of a round linear rail. Gray is the linear bearings of the carriage. Orange is the ball bearings inside the linear bearings. Blue is the rail.

Rails also need to be made in such a way that they can support themselves without collapsing under the carriage's weight. This is not a huge concern for the project because the carriages won't be carrying huge amounts of weight. This does, however, instruct manufacturers in what products they make available. This means that most available linear rails will be more than strong enough for the project's needs.

There is very little difference between riding on the inside or outside of the rail, because the carriage will be stable on two rails either way. Rails where the carriage rides on the outside are much more available than rails where the carriage rides on the inside, so riding on the outside is the better choice.

In most cases it is best to use linear bearings made with ball bearings, rather than wheels, especially since the carriages will be driven externally, by belt or chain drives, so they don't need any powered wheels. An optimal ball bearing ridden linear rail is shown below in figure 24.

Figure 24: Square Linear Rail With Ball Bearings - Profile of a square linear rail with ball bearings instead of wheels. Gray is the linear bearings of the carriage. Orange is the ball bearings inside the linear bearings. Blue is the rail.

It seems the best choice of linear rail is a square rail that is readily available, already has all of the necessary fastening points, and has a corresponding carriage that will not fall off when the rail shifts laterally. The best carriage is the one that best fits the chosen rail and moves smoothly along it.

3.6.6 Cable Management

The device will need to bus signals and power from the microcontroller to the motors driving the machine. This will require some form of cable management in order to keep the cables tidy and prevent them from getting caught in any mechanisms. There are several choices when it comes to cable management on moving parts.

When it comes to rotational mechanisms, such as a robot arm, it is actually much easier to implement cable management. This is because the distance between the axes of movement remain constant. This allows the designer to simply affix the cable directly to the mechanisms (only where it wouldn't interfere with clearance) and then they just need to add some extra slack to the joints. Unfortunately, this is not a possibility for the pen plotter, because it uses sliding mechanisms, which change the length between the axes, so the cable needed to reach the far end needs to go somewhere when the axis moves to the near end. This creates a much more difficult problem of cable management.

The first method of cable management for sliding mechanisms such as the axes on the pen plotter is to give enough slack for the cable to still reach when the mechanism slides to the far end of the rail and just let the cables hang loosely. This is illustrated in figure 25, below. The problem with this is that if the cables hang low, they could end up tangled or caught on something. If this happens, the device could easily move too quickly and either unplug or completely break the cables. If the cables are hanging over one or more rails they could also get under the carriages and cause those to get stuck. The bare minimum addition to this would be to bind the cables together, for example with tape. This would at least prevent the cables from getting tangled up with each other.

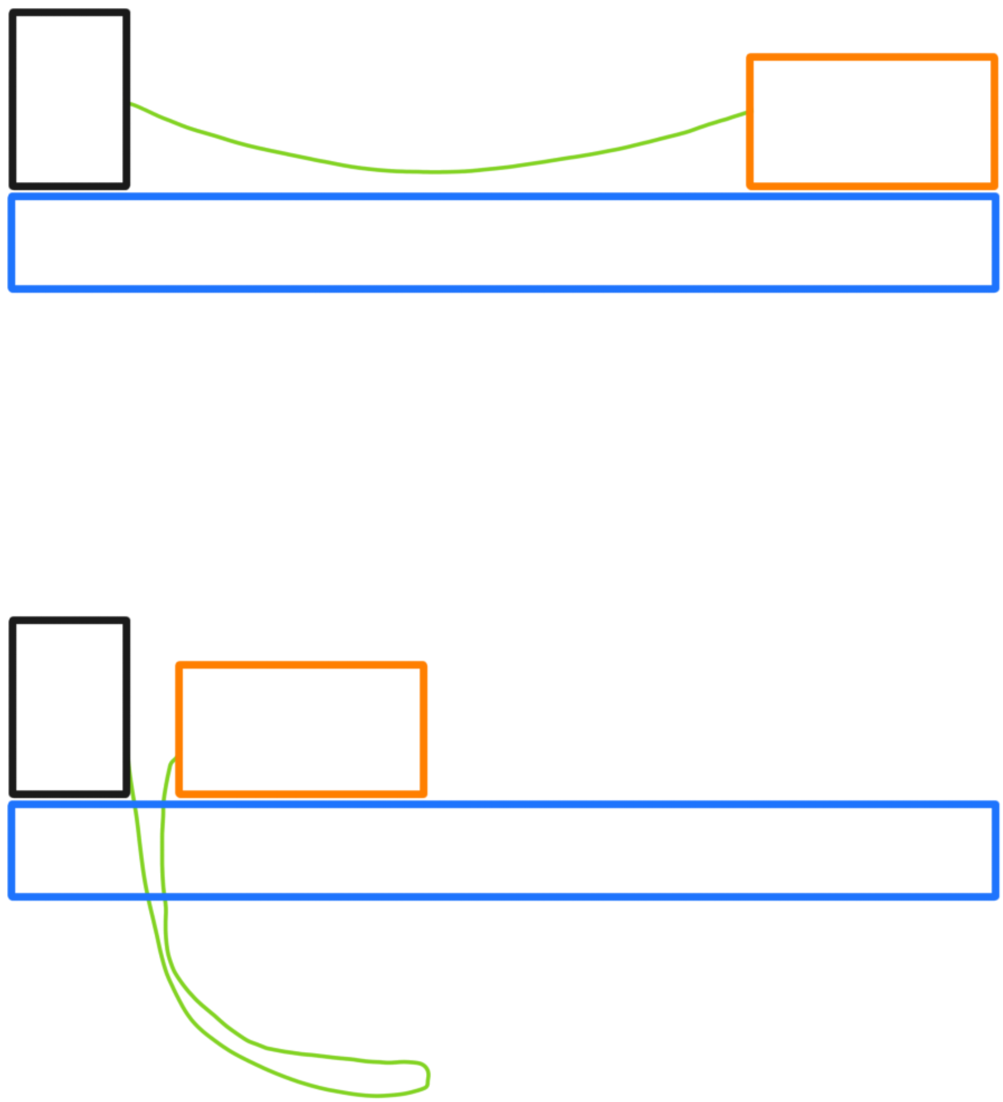


Figure 25: No Cable Management - Side view illustration of a carriage on a rail with no cable management. Blue is the rail, orange is the carriage, black is anything attached solidly to the rail, and green is the cable.

The next solution is one that is common with 3D printers. The solution is to suspend the cables from above. This prevents the cables from falling and getting caught in anything. It also allows the designer to add extra slack to the cables because it will add more space for the cable to hang. Adding more slack gives the design more room for error without the cable becoming taught and unplugging or breaking itself. Most 3D printers mange to suspend their cables by using a fairly rigid tube to transport the plastic filament and attaching the cables to that tube. Cables can also be suspended from a hook attached to the device. This would accomplish the task of suspending the cables, but would add a lot of extra clearance to the device, and would be aesthetically unpleasant. This is illustrated in figure 26, below.

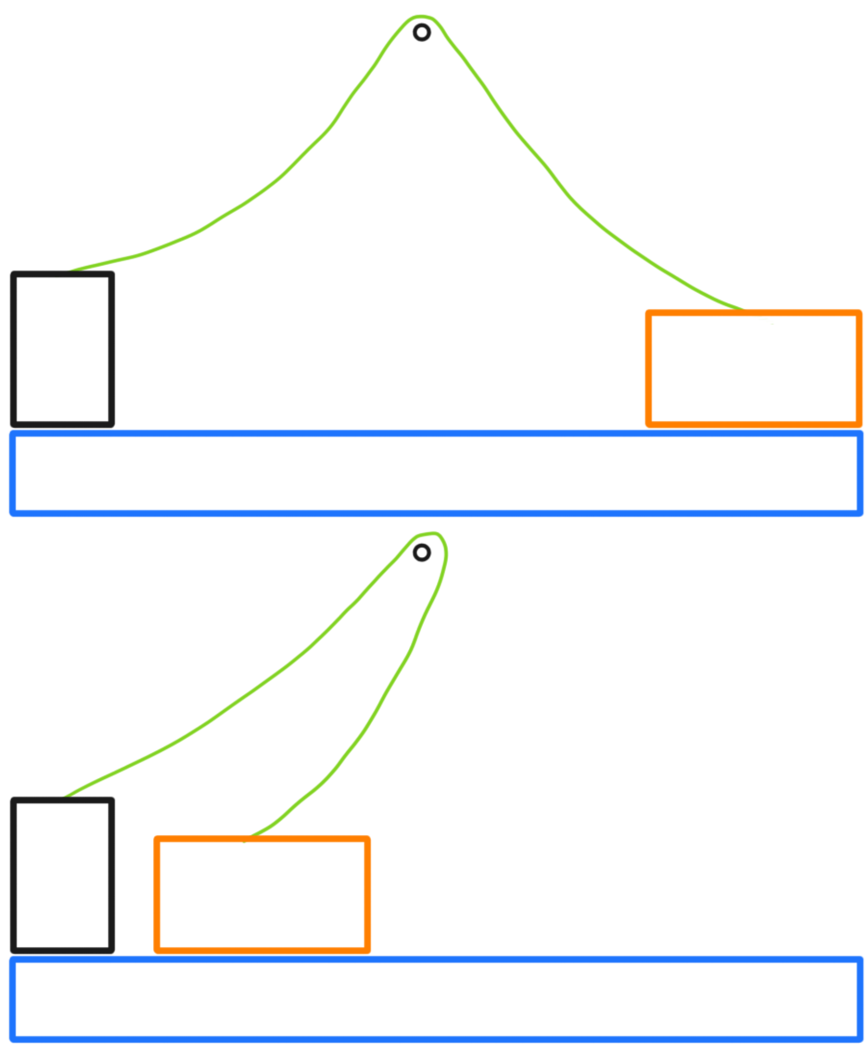


Figure 26: Suspended Cable Management - Side view illustration of a carriage on a rail with suspended cable management. Blue is the rail, orange is the carriage, black is anything attached solidly to the rail, including the hook suspending the cable, and green is the cable.

Another solution is to use something that can only flex in one direction, and that direction would be up, such as a hinged device, like a plastic chain with a chanel inside for cables to pass through. These devices exist for cable management, and theoretically could be used for this application. Unfortunately, most available ones are made for a larger scale, and would be too large and heavy for this application. They would also get in the way of the device's driving belts. There is also another option, which is inspired by the 3D printer's rigid tube. The option is to use a flexible piece of plastic, which is wide enough that it can only flex upwards or downwards, and then mount it on top of the device, so that it can only flex upwards. Then the cables can be connected to the plastic piece, so when the carriage is near, the plastic flexes upwards and the cables go with it. This keeps the cables in the air, and not falling on anything or getting caught. This is illustrated in figure 27, below.

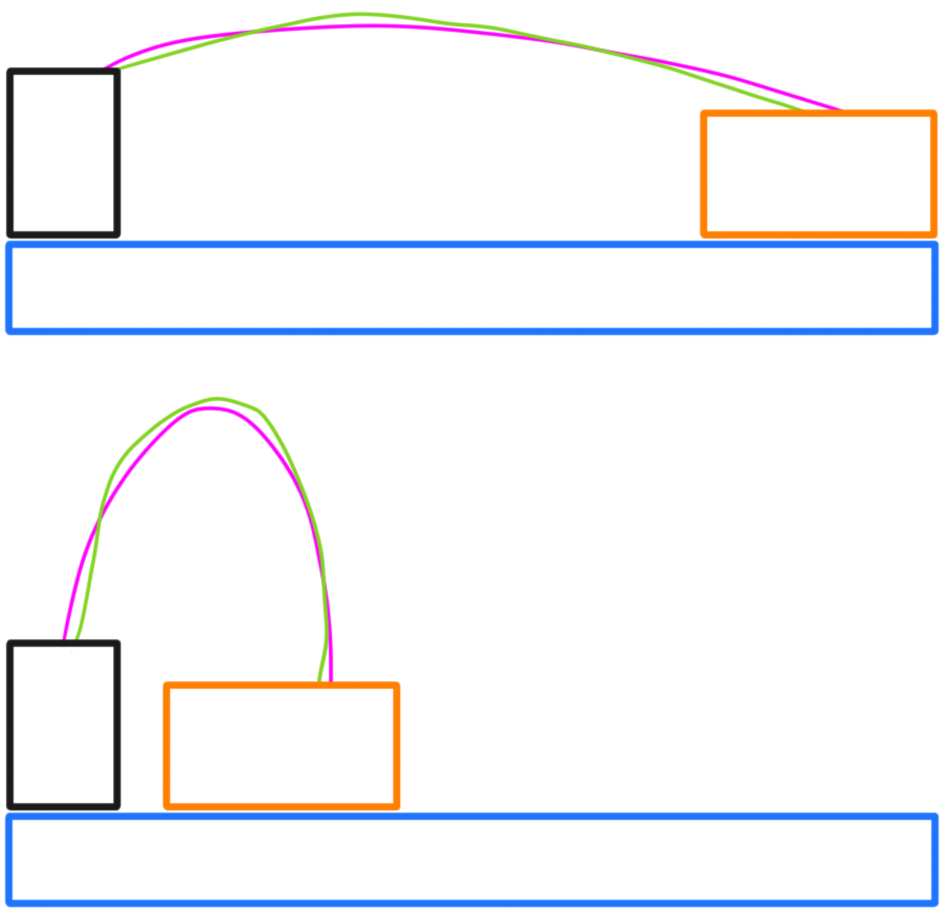


Figure 27: One Directional Flexing Cable Management - Side view illustration of a carriage on a rail with a component that only flexes in one direction used for cable management. Blue is the rail, orange is the carriage, black is anything attached solidly to the rail, pink is the piece of plastic that only flexes upwards, and green is the cable.

There is another possible solution involving sliding connections. The device could be designed with conductive rails along the slider's rails. This design is similar to the third rail, common among subway designs. Data and power could be transported through these conductive rails. The microcontroller would connect signaling and power pins to the conductive rails with fixed connections, as these parts wouldn't need to move. The carriage riding the rail could then use conductive pads to make contact with the conductive rail and make a complete circuit with the devices on the carriage. This is illustrated in figure 28, below. This method would eliminate any extra moving parts from the system, which is theoretically desirable as it reduces failure points, but unfortunately the design would likely be too inconsistent in practice. It would be difficult to keep the contact pads in perfect contact with the rails at all times, and if the contact pads drifted at all, the signal to the downstream mechanisms could be affected. Friction between the parts could also cause wear over time, as they would be constantly sliding across each other. Moving the contact pads would also effectively change the length of the lead running to the parts, and if the resistance of the conductive rails were too significant, this would change the impedance of the leads while the device is moving, which could cause further problems with the electric motors. For these reasons, this is likely a poor choice for this project.

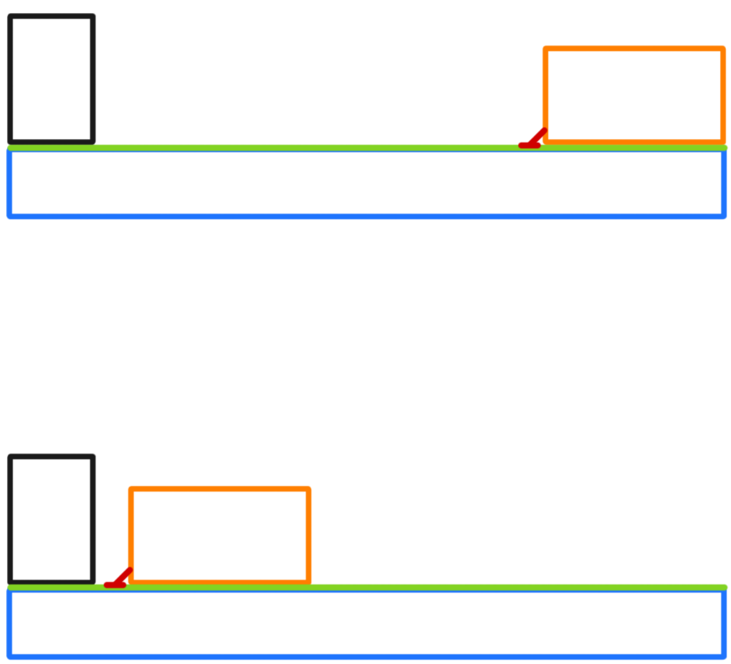


Figure 28: Conductive Rail Cable Management - Side view illustration of a carriage on a rail with a conductive rail on it in place of cables. Blue is the rail, orange is the carriage, black is anything attached solidly to the rail, green is the conductive rail, and red is the carriages conductive pad.

Overall, the best choice for this project is likely the flexible plastic, as it pulls the cables up as the slack is not needed, rather than letting it fall down, and it is cheap and easy to implement.

4.0 Standards and Design Constraints

This section provides an overview of the relevant standards for our project, the impact each has on our design and why this makes a difference. This part also goes over many different types of constraints we will face throughout the course of this project and what we can do to overcome or mitigate these in order to produce the best product.

4.1 Relevant Standards

Standards are a crucial way to establish consistency and safety for designs. If relevant standards are adopted by a wide range of creators, it will allow for a better made and safer product to reach their customers. The standards referred to below are all put into place by a Standards Development Organization also known as an SDO. We searched through several SDO databases to find relevant standards.

For a pen plotter, precision is very important in a number of aspects. When it comes to the output, a precise image is wanted in order to best replicate the intended output. To create this level of detail in output, the plotter must be assembled with parts that were machined with consistency, standardization and safety in mind.

Our process for finding relevant standards was to firstly determine areas where standardization would be seen as important. By importance we mean the standards or regulations that would allow our project to be safe and meet normalized design specifications so that we know our design will work similar to existing ideas on the market.

There are a number of different types of standards we need to take into account. The following standards are common types seen in many engineering designs:

- Safety
- Testing
- Reliability
- Communications

- Data Formats
- Documentation
- Design Methods
- Programming Languages
- Connector Standards
- Meta Standards

We came to the conclusion that it would be necessary to take into account some of these standards and disregard others as they do not seem to apply to our pen plotter design. In the following sections these general standard types will be discussed and the relevance will be determined. This will allow us to narrow down the type of standards we should be looking for and to assist us in finding new standards in the future.

4.1.1 Safety

Safety standards are something every design should take into account as it can make or break a design as well as the market implementation of that product. If safety standards are not met, although many are considered voluntary, they may not meet safety regulations established by a government because of that. A common regulation we must take into account is RoHS. RoHs is the restriction of hazardous substances directive. They regulate several substances deemed to be hazardous and some of those substances can be found in our design. Specifically in the pcb so we must make sure to take care if a disposal of materials containing such substances is necessary. These kinds of regulations are important for protecting the environment from substantial harm.

Other types of safety standards that may be seen as relevant are those dealing with moving or openly housed machinery. Such as that of many cnc machines. Our project being an open design allows for the possibility of bodily injury if one is not careful or if the design is not made as safe as possible by following standards put into place. Not only is bodily injury affected by this but the machine could also be damaged if the open design aspect is not addressed.

4.1.2 Testing

Through the course of these semesters we will be forming many tests in order to get the final product we are looking for. There are many testing standards put into place which would be useful for many different reasons. The major reason would be that it could streamline our testing process. We may attempt to try numerous irrelevant tests that may not affect our final product if standards are not followed. Testing on similar products has been done many times now so it could be useful to see how the industry typically performs said tests.

We could use testing standards in the implementation of our x-y axis moving arms as well as a pcb. We will be implementing a somewhat standard cnc style design and testing it using standardized methods will allow us to have more effective and efficient methods.

4.1.3 Reliability

Reliability standards are seen as an important aspect to our group since we are building a somewhat complicated and precise machine. If it is determined to be unreliable, it may no longer function as intended while only being used as intended. There are numerous reliability standards out there, many have to do with things such as stress screening of different components. It could be important to determine if the parts we selected for our mechanical parts were stress tested in order to determine that they are built to last and can withstand the engineering specifications we will be imposing upon them.

There is also a decent amount of programming and software development related reliability standards. It is common for a customer to impose a certain percentage on a piece of software that should say that said software function without any errors. If these reliability standards imposed are not met, the company should be made to fix the bugs causing these issues. It is something that our group is looking into as well. Software reliability is an important factor for us since a large part of our design is software dependent such as the gui as well as the processor's firmware.

4.1.4 Data Formats

Standardized data formats are another area we will be focusing on. Our design will have multiple communicating parts written in different programming languages. It is important to determine that they can all communicate properly and all of the commands given can be understood by each part of the system.

The user interface we will be designing and firmware w software which controls the pen plotters pcb will have different programming languages they are written in, but they must be able to communicate. It is the API’s job to make sure that data formatting standards are followed in order to make this communication seamless. There are standard formats which have been proven to be effective at these processes and they are something which are important for a project such as a pen plotter that relies on user input from a separate computer such as ours.

4.1.5 Documentation

Proper documentation is crucial in any engineering design. Documentation allows for all of the processes which went into the research, design and implementation to be expressed in a way that people outside of the project can understand the theory and uses of the design. It is important to use standardized design practices in order to create user friendly and readable documentation.

With our project, we will need to implement software and hardware related documentation. It is important to keep all necessary information within the documentation in order to allow outside use of our project. If something like our design would even make it to the market it would be essentially useless without the necessary documentation since most people would find it unnecessarily difficult or even impossible to use.

Hardware, software and user manual documentation are separate entities which all serve their own respective purposes. There are many related standards in existence which allow for the creation of these types of documentation to be consistent throughout many different kinds of products and therefore readable to people used to these kinds of standard documentations. Software documentation for our project would give detailed steps on how to use the user interface, as well as possibly high level details on the firmware and some of the api within our project. This would allow for more context to be given when explaining how the software is used and why it was made this way. Hardware documentation would be used to document specifications of the hardware we used. This would be useful for users to understand how components work with each other. And also if the user wants to make modifications or their own version of our design, they have some components and related research and development to take inspiration from. User manual documentation can have general info regarding the proper use and possibly common issues and debugging methods. All different standardized documentation for quality of life measures for the user.

4.1.6 Design Methods

We also need to take into consideration standardized design methods. This can also tie into documentation because standardized design methods are used for software design standardization as well as hardware descriptions of a product. It is important to detail technical specifications of our design in standard ways so that they are easy to follow and read. Hardware description standardization is especially important for our project since hardware specifications are very standardized in similar applications as well as hardware applications universally.

4.1.7 Programming Languages

Since our design is heavily software based, it is very important for us to be following proper programming language standards. If programming language standards are not followed it could be very difficult to create proper software and could create many production reliability and portability issues. Our software is meant to be used on different computers, if it is not standardized, machines using different operating systems for example could have difficulty running it.

Programming language standardization is very important for us for a number of reasons. One is that it would subsequently make documentation easier if it follows the proper standards. If the software you are documenting is not standardized it could have a negative impact on the creation of standardized documentation for it.

Non-standardized code could also prove to be extremely difficult to follow for the person who has not written it. It is also very possible that the developer themselves may have trouble understanding what was written or how a function works down the line. This makes debugging and future additions to the software much more difficult than necessary and this is a problem we hope to avoid by doing proper research and implementation of standardized programming practices and development.

As stated before, since we will be using multiple languages and libraries, it is important that they are all standardized. Large code libraries can be difficult to follow if they are not properly standardized and also if industry standard coding practices such as the use of git repositories are not followed. Multiple people will be working on the software during the course of our designing and implementation phases and this makes standardization all the more important. We wish for our code to be seamlessly written by multiple developers and that is done through the standardization written by the creators of the programming languages themselves as well as other developers with much more experience in the languages than that of ourselves.

4.1.8 Meta Standards

Meta standards typically describe how a system should function in unison. This can also be described as a standard which is describing the specifications of a part created with the use of standardization. An example from the course textbook on engineering specifications and requirements use the example of an electrical standard describing the voltages of a design.

Meta standards are something our group can use to determine that we are using standards properly. If we are making use of proper standardization, they should be able to describe certain elements of our design.

The standards we determined were important to highlight are expressed in the following sections:

4.1.9 IEEE/ISO/IEC 29148-2011 Systems and Software engineering Life Cycle Processes

This standard is described by the IEEE as, “the provisions for the processes and products related to the engineering of requirements for systems and software products and services throughout the life cycle.” Another way of saying this is that this is the standard related to making sure a design's software and software systems are standardized throughout the entire design, build and implementation cycles as well as other elements in a products life cycle. This is important for any design that wants to make a long term impact from a software standpoint.

We want our product to be continuously developing and be able to receive new updates and improvements in the future or as needed. It is important to have standardized software development practices in each stage of a product's development. This increases the credibility of a product within its market.

This is an important standard to show that our software system is a viable product to possible markers and other investors. If they see that we are standardizing the processes of development throughout its lifecycle, they will see that there is a good amount of longevity in our design. With our need to develop a user interface and other software for our design. It is very important that this standard is considered.

4.1.10 ISO/IEC/IEEE 29119 Software testing

A standard relating to software testing is very important for any design with a software aspect. This standard is described by the ISO as a, “series of software testing standards is to define an internationally-agreed set of standards for software testing that can be used by any organization when performing any form of software testing.” The testing of our software during the course of this project is a very important standing which we should follow. The standardized testing of software leads to the assurance of proper coding practices as well as the implementation of standardized software development.

If we use standardized testing it provides users and customers with greater assurance that the product was assembled correctly from a software standpoint. Standardized testing allows for many aspects of development to be accounted for. If non standardized testing methods are used, it is possible not every major issue would be found. There will always be edge cases that standard testing will not be able to detect, but to use a standard testing practice will lessen the amount of common issues that will arise.

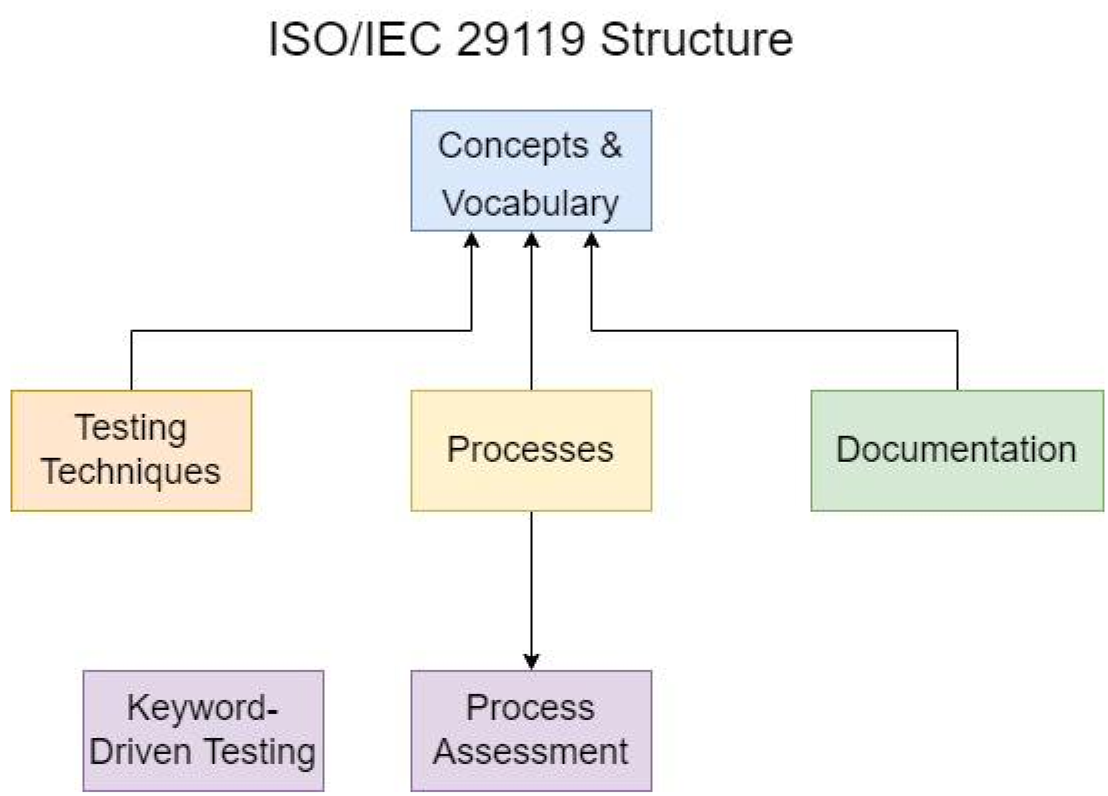


Figure 29: The Structure of ISO/IEC/IEEE 29119

4.1.11 IPC-2221 Generic Standard on Printed Board Design

Being that our design implements a pcb, it is very important that we use standardized methods in the development and implementation of that circuit board and the components it is composed of. This standard IPC-2221 is described by its creators as, "a generic standard for circuit board design. It lays down the requirements for PCB design and different forms of component mounting/interconnection structures." This standard provides a baseline for all circuit boards to follow in order to establish consistency among pcbs and the products they are used to control.

It is called a generic standard since it covers many aspects of circuit board design. It details the standard specifications of many things such as the clearance required between components on the board, materials used to create the board, sizes of components and other engineering specifications that we will see implemented on our own PCB design. It is important to use these standard design principles to ensure that our board works as expected and is reliable and consistent throughout testing and implementation. If we did not follow the standard format of a pcb, we may encounter unwanted results. These circuit board specifications have already been thoroughly tested and are proven to be appropriate for pcb design which is why they were deemed an industry standard.

4.2 Design Constraints

This section details the realistic constraints we may face throughout the course of our design and implementation of our project, as well as some constraints we may face if this project was sold to customers and used by the public. We will be outlining constraints related to many different aspects of a product and its development. These are important issues for our group to consider as we may or may not be affected by these issues and learning about them before they occur will better assist us in overcoming a problem more quickly.

There are certain types of constraints which were emphasized by our course leaders which will be covered in the following sections. The relevant constraints were grouped together by similar relevance and the topics are as follows:

- Economic and time
- Environment, social and political
- Ethical, health and safety
- Manufacturability and sustainability

4.2.1 Economic & Time

The initial budget we determined to be acceptable was $200 for each group member. In a group with 4 members this was equivalent to a total of $800. We have no form of outside funding or sponsorship, so these costs are out of pocket. Fortunately, this initial budget exceeded our parts needed for our initial testing greatly, as we initially spent around $300 on the parts. The rest of our budget will be allocated to further research and development costs if necessary.

At the time of the design and testing of our project, many relevant parts are suffering from a supply shortage. We had to make certain that all the parts necessary were in stock and shippable within a reasonable timeframe. We needed to order parts as soon as possible in order to make sure we had everything for testing in the allotted time for this project. We did not suffer many setbacks as anticipated in this section as we were able to find most of our parts available from a single seller.

We will also need to have several 3D printed parts fabricated. In order to make sure this is done in a timely manner, we will need to make sure these are created in time for the assembly of our design.

In terms of time constraints, we have the semester containing Senior Design 1 and following that Senior Design 2. These two courses take place throughout the Spring and Summer semesters. This ranges from the months of January until the beginning of August. There are certain aspects of the project expected to be completed by the end of May, mostly relating to the design and research of the project. In the summer semester it is expected to be complete with the design and initial testing phase and be working on the implementation of the design.

We decided on milestones which will best help us complete everything necessary in the time given. We have a somewhat shortened Senior Design 2 semester given that it takes place in a shorter summer semester rather than a fall or spring one. This however, gives us a continuous allotted range of dates which we will be focused on all the necessary aspects of this project. If the timeline goals are following, the time constraints should not be too taxing on our project.

4.2.2 Environmental, Social and Political

Our design will likely not face many design constraints relating to social or political issues. Printers and 3d printers are not much of a concern with these matters as they are universally accepted to be a non-threat to society. Our project could be faced with certain environmental constraints if not handled properly.

With our pen plotter, we plan to use a variety of mainly ink-based tools for plotting. Ink found in pens and other writing utensils can be considered hazardous if consumed or discarded into the environment. Since the user is able to use many different writing utensils at their disposal, it is up to us to make sure that they use discretion when disposing or handling their equipment. This is not a major concern but is still something our group thought about when designing our project.

Our project is also powered electronically with batteries and there are certain standards detailing the importance of transportation, use and disposal of batteries as they can be hazardous to the environment. If this product was taken to the market, it would be necessary to make sure our design met all of the required and voluntary standards relevant to the electronic elements of our pen plotter.

4.2.3 Ethical, Health and Safety

This section will be covering the impacts our design could have ethically, as well as on the health and safety of operators of our pen plotter. Our design in general is relatively safe as we are not dealing with many hazardous materials or making use of a design which could harm the user in any way. Ethically, we did not see many issues with our project as it is very non-invasive and uncontroversial from our perspective. It is however, very important to take into account all possible health and safety constraints we could face on our design. Without taking into account these issues, it would not make for a viable project and it would not pass many government regulations and other standards.

Our design has metal mechanical parts moving in the open air. It is important to maintain awareness of these parts when they are in use as it could be a safety concern if a body part or other object came in contact with our device while it is in the process of plotting. The motors used are not very powerful but it is another concern our group is aware of when implementing our design. We are looking for any way to mitigate the possibility of injury to ourselves as well as anyone who may use our pen plotter. Since it is not possible to completely mitigate all risks entirely, it is important for us to facilitate awareness of these potential hazards.

4.2.4 Manufacturability and Sustainability

The manufacturing of our design is not something that seems like it would be easy to quickly be made on a large scale. There are similar products on the market and even the more popular 3D printer, which shares many of the same design characteristics of a pen plotter, but if we were to start manufacturing at a large scale from scratch it would take some planning as this is a somewhat large project with multiple moving parts. All of the parts to assemble are fairly common and can all be sourced from multiple retailers. The motors and the majority of the electronics could be bought and assembled to our design as over the counter products to make manufacturing more simple.

Our design is largely mechanical in terms of functionality. It could take a decent amount of time to assemble based on the proficiency and experience of the assembler. We tried to make our design as simplistic as possible so that hobbyists would be able to assemble something such as this and not only trained professionals.

Our design should remain sustainable as long as there is a market for this type of product. It does not appear that there will ever be a large market for pen plotters as there is not a huge market for them in the professional field like there is for 3D printers. There is a significant amount of people who enjoy the artwork and the machinery behind a pen plotter so there is currently still sustainability for our design.

The parts used to assemble our design will remain sustainable for the foreseeable future as it takes inspiration from existing machines that are used widely across many areas of the professional world. Our design could be prone to mechanical faults and operator related failures as it is an open design and many of the important components have no form of protection from its surroundings. However, if our design is kept in a safe location, it should have none of these related sustainability issues.

5.0 Design

The design of the Automated Pen Plotter will be divided into two main parts: the software and the hardware. Since this project has a variety of components and parts, we decided to split it and elaborate each part and component and see what is needed and what is messing up to meet the goals of the project. For example, what will the project need to move the pine while drawing, what system will it use, what power will be needed, what system is used etc. After several meetings and discussion with the group members, we carefully selected the suitable parts and the materials needed for the project to work perfectly.

While designing the project we will have to do some tests for the parts that we have and see if they work perfectly and if not, what needs to be changed to make them work properly and in order to do that we have collected all the parts for the project and started to work on them.

After we collect all the parts and components for the project, we will have to design our PCB (printed circuit board) to gather all the electrical parts in one board and organize them in a proper way, and to that we need to find the best software to cork on to design the PCB.

Next, we need to test all the electrical parts that we have bought for the project and see if there is something missing or any other issues with the part so we can figure out what to do and not to be late, for this part we will test most of the electrical parts one by one.

5.1 Hardware Design

For the Automated Pen Plotter, there are multiple parts that represent it such as, the microcontroller that is used to send and receive orders, the servos that will move the pen holder across the X-Y plane, Stepper Motor which will be connected to the driver motor, CNC Drive Expansion Board hold and divide the connections between the parts, Micro Switches, and power supply. All these parts that are mentioned will be powered by AC/DC power supply, also this power will go through the microcontroller voltage regulator which will provide the needed small value of voltage. All these parts will work together to move the pen holder and draw the input picture.

For our project to work perfectly we had to buy some of the parts that are ready to use and work on, for example a CNC shield and UNO REV3. We will be doing our demo project using these two parts and see how every part is connected and how our project will look. These parts helped us to get more ideas and thoughts about how the 3D printers work and how our project is going to work. After work on the demo project and collecting as much as ideas we need to improve and run our project we will need to work on making our own parts and how to connect them together to work perfectly.

5.1.1 Stepper Motor

Stepper motor is an electrical device that is used to control and move parts by converting the digital pulses to mechanical shaft rotation. Also, the stepper motor is used to control the angular position of the rotor. Stepper motors are called by that name because each pulse of electricity turns the motor by one step and stepper motors are usually controlled by a part that is called stepper driver. The diagram below explains how the stepper motors work.

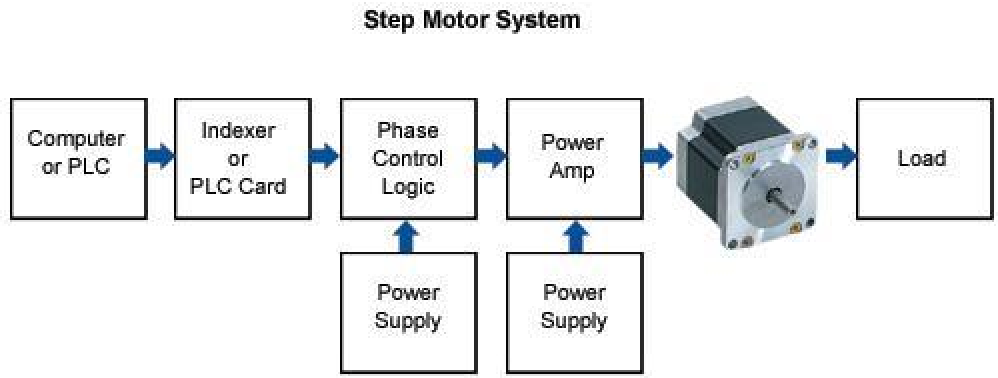


Figure 30: Stepper Motor Diagram.

Stepper motors used in many different manufacturers. Such as, toys, hard disk drivers, telescopes, and robotics. Also, stepper motors are commonly used in 3D printers because of the features that they have, so stepper motors are used because they do not work at high speed, however they have a high holding torque, and the picture below demonstrates what the stepper motor consists of from the inside.

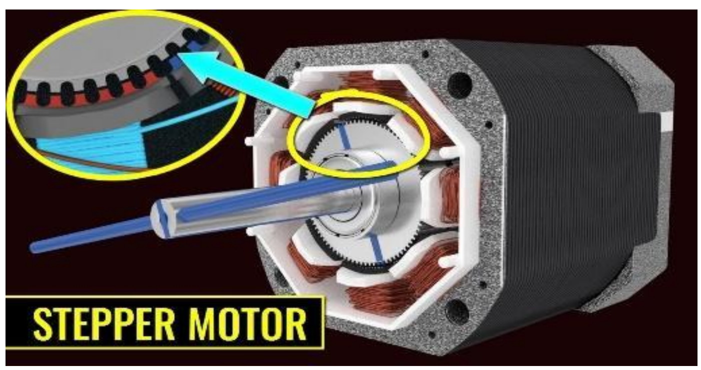


Figure 31: Stepper Motor Components.

We will use the stepper motors for our project since they are so important and can do the work needed. There are many different types of stepper motors which can work differently and each one of them have different features than the others. For example, permanent magnet step motor (PM), variable reluctance stepper motor(VR) and Hybrid stepper motor (HY), these are the most common types of stepper motor.

For this project we chose to use two different types of steppers motors: the hybrid stepper motor and the variable reluctance stepper motor. To distinguish between the two steppers, the hybrid stepper is having high torque and low resistance, however it is more accurate and the table below explains more about the difference between the two of the motors. To create a stepper system, we will need to provide a stepper motor, power source, and stepper drive. There are different types of motors that can be used for different tasks but for this task we choose a stepper motor because it can give a high accuracy by working in a high amount of torque at low speed.

characteristics	hybrid stepper motor	variable reluctance stepper motor
Cost	Expensive	Moderate
Design	Complex	Simple
Speed	Low	High
torque	High	Low
Stepping	Full, half and micro	Full step only
Resolution	1.8 and smaller	1.8 and smaller
Noise	Quite	Noisy

Table 6: Hybrid Stepper Motor Vs Variable Reluctance Stepper Motor

5.1.2 NEMA 17

There are several types of stepper motors that can be used in printing and drawing and 3D printers. NEMA 17 stepper motor is one of the most famous steppers and it is commonly used. There are some features that the stepper motors have, and they can be different from each other. The most distinguishing one is the torque where it can be different from stepper motor to other and by increasing the torque rating the size of the stepper motor and cost increase. The torque of the stepper motor is measured in newtons centimeters.

The most standard Nema 17 stepper motor is 45Ncm. For our project we decided to use the 17HS4023(13Ncm) . This stepper motor has dimensions of 42x42x23mm/1.65""x1.65""x0.9"" ,1.8 deg. This stepper motor works under 1.0A current and resistance of 3.5 ohms. And to know the rated voltage that the stepper motor can work under we use ohm’s law, and we can find that the voltage needed is 3.5 volt.

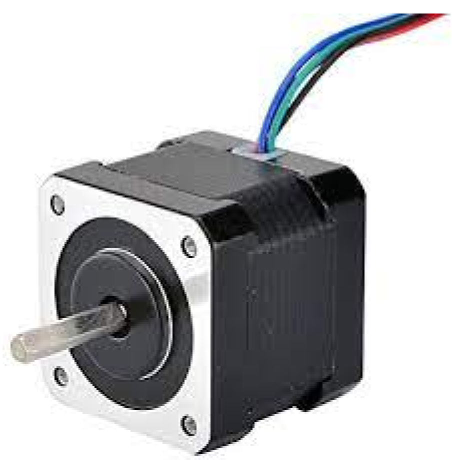


Figure 32: STEPPERONLINE Stepper Motor Nema 17 Bipolar 40mm 64oz.in(45Ncm) 2A 4 Lead 3D Printer Hobby CNC

5.1.3 Stepper Motor Drivers

Stepper motor drivers are an electrical component that are used to run and control the stepper motors. To run the stepper motor, we will have to use a stepper motor driver on our project which will be connected to the microcontroller and the stepper motors. The stepper motor drivers used to provide and adjust the current that is coming and going to the stepper motor, also it provides individual voltage signals to run the stepper motor, so stepper motor drivers are a very important part to run the stepper drivers.

To control the stepper motor the stepper motor driver uses four main pins Clock which is used for provide the orders and command signals, the direction pin which is used to set the direction of the rotation of the motor, the mode pin which is used to determine the size of the step and the rest pin that is used to rest the driver settings to the initial settings.

As with any other electrical component, there are different types of stepper motor drivers which have different features than the others. For example, there is a Unipolar stepper motor driver, Bipolar series stepper motor driver and Bipolar parallel stepper motor driver and all these three stepper motor drivers are different from each other and the table below shows the difference between them.

Driver	Number of wires	Features
Unipolar	5,6,8	Simple operate, low cost and it used for low speed
Bipolar	4,6,8	Low speed at high torque and low torque at high speed
Bipolar parallel	4,8	High torque at high speed

Table 7: Stepper Motor Drivers

Each one of the stepper motor drivers that are mentioned above has different wire connections and the diagram below explains how each one is wired and how they look like.

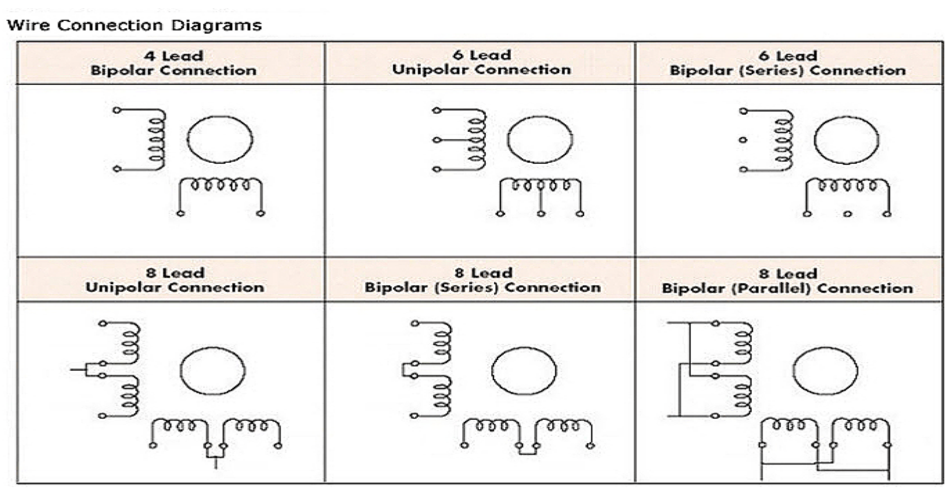


Figure 33: Wire Connection Diagrams

To choose the best stepper motor drive for our project we must consider some features so that we do not face any future issues that can cause damage to the stepper motor or the stepper motor driver itself. BIQU A4988 is the stepper motor drive that we chose to work on since it can interface with 3.3v and 5v systems. This driver has different step resolutions which are full, half, 1/4, 1/8 and 1/16. And the maximum current rate is 1.0125A.

Figure 34: BIQU A4988 Compatible StepStick Stepper Motor Driver Module with Heat Sink for 3D Printer Controller Ramps 1.4(Pack of 5pcs)

5.1.4 Carriage Block

For carrying the pen holder, we use two aluminum linear rails that are connected to each other by wheels which allows them to move smoothly in the X-axis. The pen holder is located on one of the aluminum rails and is also connected to wheels which allows it to move vertically and horizontally. The aluminum linear rails that we chose to use have 500 millimeters of slide travel distance with size of MGN15 as shown in the picture.

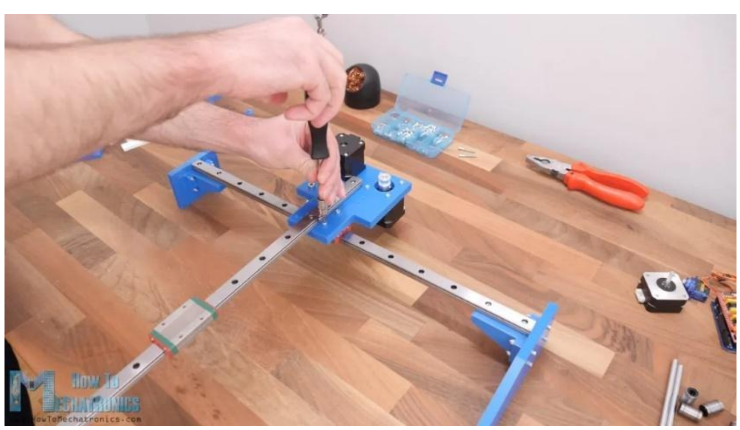


Figure 35: Carriage Block

These aluminum rails will hold the machine that is going to hold the pen and help it move smoothly and make the drawing easy. There are different types of rails that can be used to hold the pen holder, but we decided that we will use the aluminum once because they are very light in weight which makes it move faster and they do not cause any noise or over friction. The aluminum will have wheels that are separating between it the pen holds so that the pen can easily and there will be a belt used for moving the wheels. The two rails will be positioned in base of wood and in the top of that the paper will placed and on this paper the work will be done, the wood base that we are going to use will help on holding the aluminum rails while the pen holder is moving so that when it moves it does not cause any issues.

5.1.5 Limit Switches

Limit switches are electrical parts that are commonly used in the projects that have moving parts. To know more about the limit switches we will explain it in an example, so let us imagine that we have a moving part that does a certain job and this moving part has a specific dimensions that have to follow and cannot do anything that is not on the dimension , so if the moving part exceeded the dimensions given an issue will accrue or something wrong will happen to prevent this issue from happening we will need to use the limit switch which by it part gives the moving part a limits that can move on it which are based on the given specification and if the moving part went beyond the limit the switch will stop it. The picture below shows how the limit switches work.

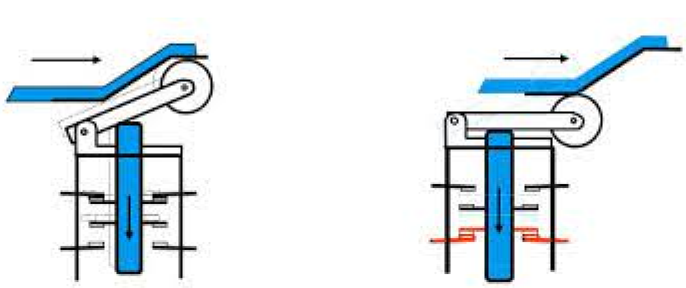


Figure 36: Limit Switch Diagram.

Since our project has moving rails and moving pen holders we see that one of the main and the most important parts for this project to work perfectly are the limit switches. To protect our project and to make sure that the project will work perfectly with no issues we saw that we need to put switches in specific places, so the limit switches need to be used in places that have moving parts where these places has a limit space to move on, let's say that the input that is given to the project is a picture and this picture will have specific dimensions where the pen holder and the rails needs to stop on, so here the job of the witches come where they prevents the rails and wheels to go beyond the required dimensions as these switches are electromechanical devices which can send signals and response whenever there is force applied to them. As many other electrical parts there are many types of limit switches and each one of them does a specific job and has different features than the others but for our project the switches that we chose can work under rated current of 5A, and operating voltage is 250V.

Figure 37:URBEST AC 250V 5A SPDT 1NO 1NC Momentary Hinge Roller Lever Micro Switches 3 Pins 10 Pcs

5.1.6 Power Supply

Every electrical project requires many different parts and sources to work perfectly with no issues, one of the most important requirements is the power source. So for our project we will have to provide a suitable power source that can sun our project with no issues and in order to do that we have to know what power source type we are going to use and how much our project needs to work properly.

For the Automated Pen Plotter to work we need to provide a power source and we need the type of the power source that we are going to use. There are several types of power sources such as AC and DC power that can be used to run different projects and it depends on the size of the project that we want to run. For the size of our project, we will use AC adapter which will provide us with the power needed to run the microcontroller and stepper motors and to know more about the AC and DC and what is the differences between them we provided a table that demonstrate the difference and from this table we can see why we need an AC power in our project.

Alternating current	Direct current
It is safer for long distance transfers.	It cannot transfer for long distances.
The direction keeps changing.	Has a single direction, so direction changes.
Frequency range is 50-60Hz	It has no frequency.
The current changes by the time.	The current does not change by the time.
The source of it is the alternating current generators.	The source of it are cells and batteries.

Table 8: AC Vs DC

Also, to explain more about the difference between them we have a diagram that shows the flow of AC and DC.

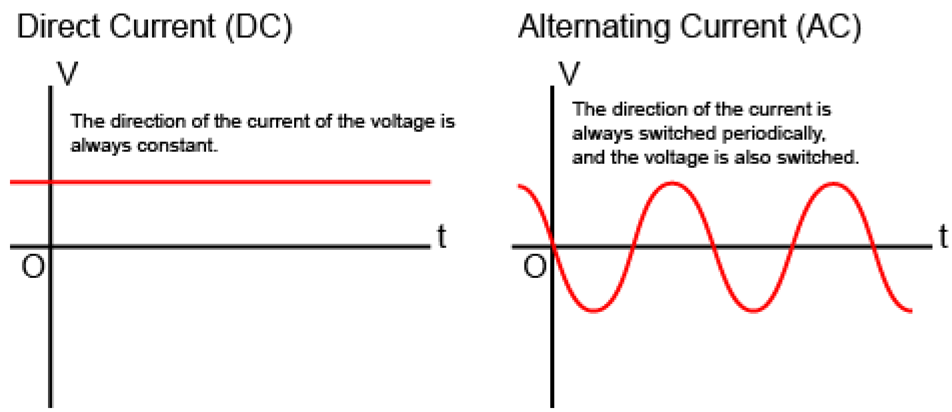


Figure 38: AC and DC Diagram.

In this project there are parts that cannot work under high voltage nether high current working under these circumstances will cause a damage in the project parts, also we do not want our project to work under low current because it prevent it from working in a proper way, and to solve this problem will need to use a voltage regulator to prevent all these mistakes from happening.

5.1.6 AC Power Supply Adapter

To make all the electrical that are used in the project work in a good condition we will need to use an AC adapter like the one that is shown in the picture below.

Figure 39: AC Adapter.

The AC adapter allows transferring energy from high energy source to low energy receiver, so the AC adapter will provide the needed current, so basically it works as a protector for the project. The AC adapter does this by converting AC to DC. To know how the AC adapter works we need to know the parts that are used to build the adapter, so the adapter consists of a transformer, bridge rectifier and capacitor as it is shown in the picture.

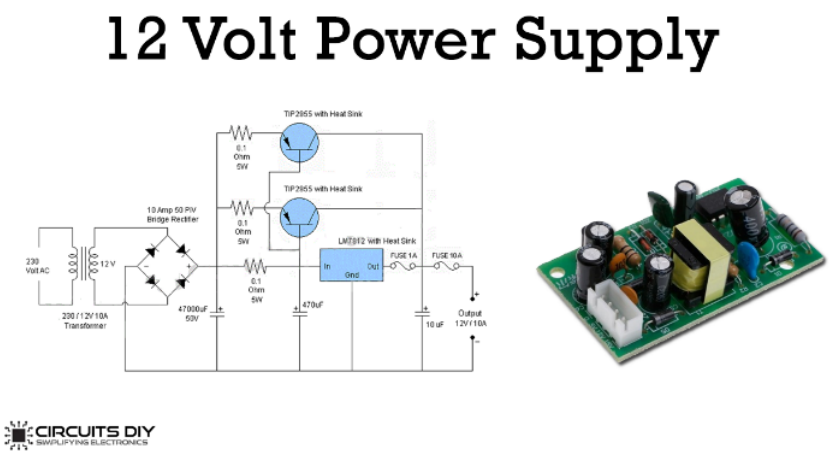


Figure 40: AC Adapter Circuit.

So the transformer works on changing the high voltage that is coming from the input into low voltage, and the capacitor works on storing the energy that is sent to it and releasing it when there is no voltage released.

5.1.7 Voltage Regulator

Since we are using an AC adapter in our project and the adapter that we are using produces 12V, we need to reduce the voltage because some of the part that we are using in the project works under low voltage otherwise we will face issues on our project, and we will not get a constant voltage. The best way to control the voltage is using a voltage regulator.

A voltage regulator is an electrical or electronic device that maintains the voltage that is produced by the power source within specified voltage, so a voltage regulator is used to produce a required output voltage for the electrical component that cannot work under high voltage. The picture below shows how the voltage regulation works.

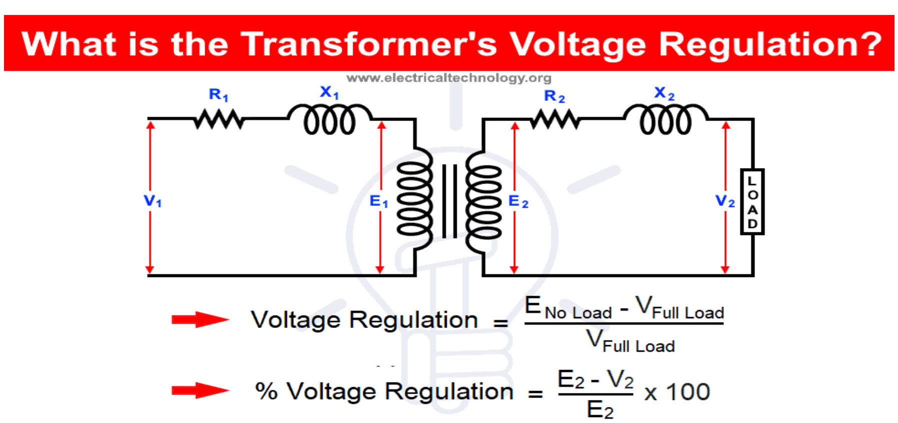


Figure 41: Voltage Regulation Diagram.

Since we are using a microcontroller and most of the microcontrollers need a minimum of 5V, the voltage regulator that we need to use must produce 5V. L7805CV is a good voltage regulator for our project.

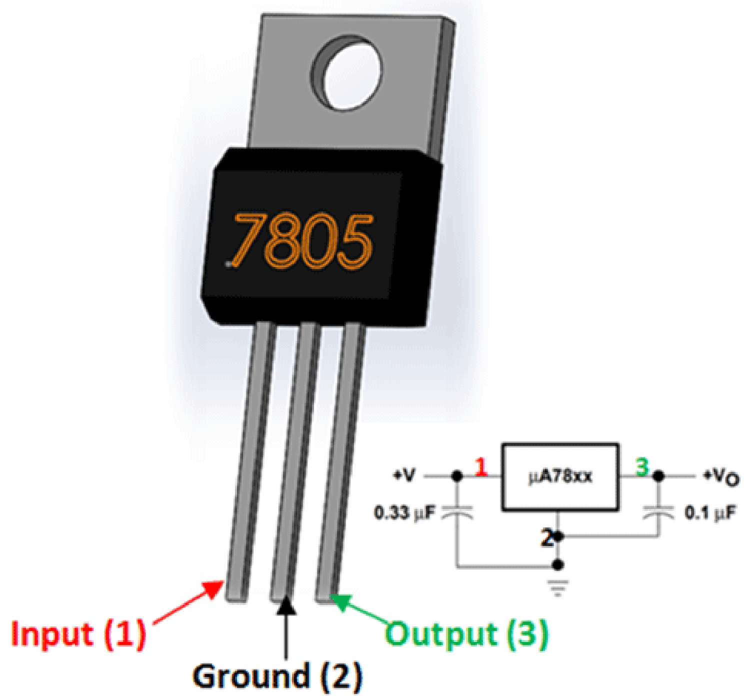


Figure 42: L7805CV Voltage Regulator.

5.1.8 Arduino components and the Microcontroller

Most of the advanced electrical projects needs brain to work because electrical project depends on sending and receiving orders, also to work the need a memory to remember and memorize the orders given and here where the job of the microcontroller come, so microcontroller is one of the most important components in the hole project and needs to be chosen wisely.

For our project, we know that we need something to work as a microcomputer to receive and give orders and to connect parts together. There are many options that we can use for our project and make them work as a microcomputer. Some of the microcontrollers we have learned and done different experiences on can be very useful on our project for example MSP430, Arduino, and many other microcontrollers. For our project we choose to work on the Arduino Uno REV3. The Arduino Uno that we are going to work on has 14 digital input and output pins. It has a USB connection, power jack and many other features that will be useful for our project and help it work well with no issues. The Arduino Uno REV3 that we are going to use is shown in the picture below.

We have chosen the Arduino UNO REV3 to work on and test our project in it because it has all the necessary parts that will help in operating our project. For example, it has a FT232RLSSOP and this chip is used for the wired connection between the USB that we will be using to send the input and the information for what we need to appear in our output, and the ATMEGA328P-PU which is the microcontroller in the Arduino, also the Arduino has a 5 volt voltage regulator, and finally the NFR24_module_SMD which is a wireless transceiver module and it works with the help of SPI communications.

Figure 43: Arduino.

On of the most important things that the Arduino Uno we are going to use will work as a storage for the firmware, so the Arduino will receive information as a cod for the user and will store them then it will give orders to the motors on how to move and start the drawing or the writing.

For the Arduino to give orders to the motors we need to connect it to something called the Arduino CNC shield and to connect it to the Arduino we need to use the stepper driver, the two parts needs to be connected to the Arduino so that the stepper motors can receive orders and work perfectly, these two parts are shown in the pictures below.
We will be working on the Arduino to get to know what the best microcontroller for our project and what parts and components are required for our project to work perfectly, so we will do some experiences and tests on the Arduino and from there we will decide what is the best for our project to work in the best shape.

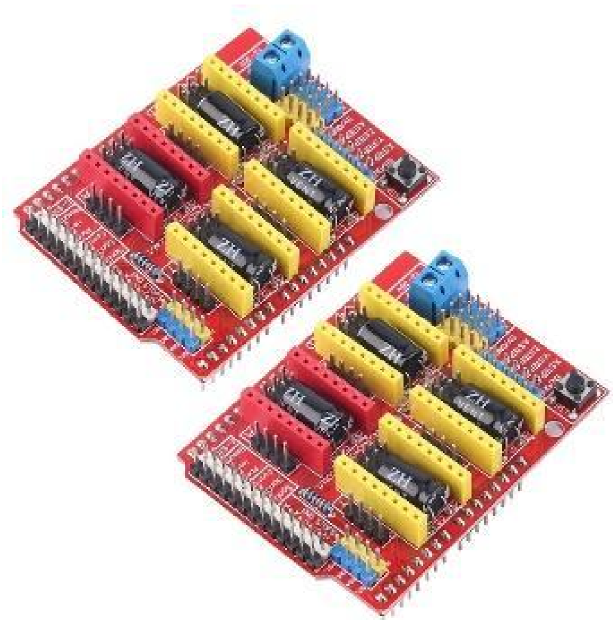


Figure 44: HiLetgo 2pcs A4988 V3 Engraver Drive Shield
3D Printer CNC Drive Expansion Board for Arduino 3D Printer CNC

5.1.9 Automated Pen plotter Circuit Diagram

For our project we need to work on a good circuit to follow and to see what we will need to get the project working perfectly. Here is a circuit diagram that shows how each part of the project will be connected to each other. The diagram in the picture below shows how the components will be connected to the CNC that is connected to the Arduino.

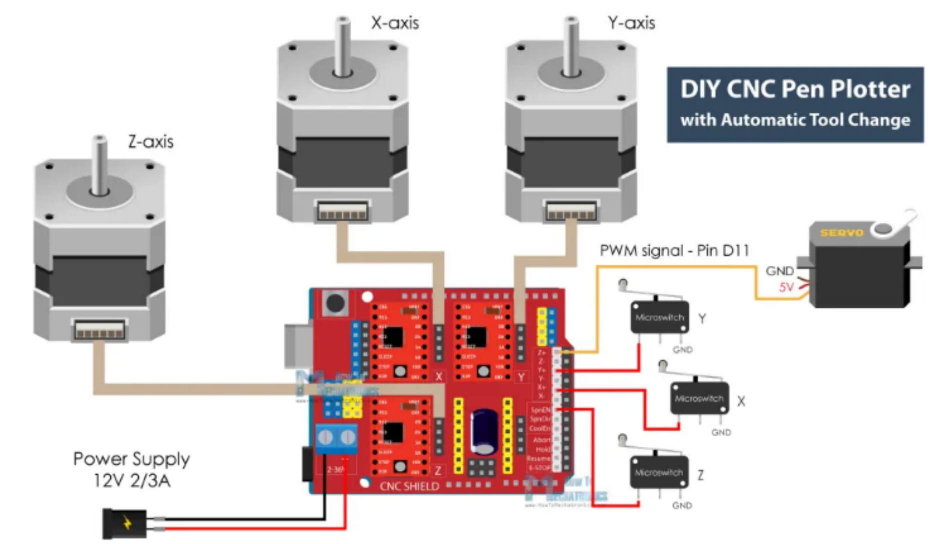


Figure 45: CNC Pen Plotter Circuit Diagram

This diagram demonstrates how each part of the electrical components is connected to the CNC shield which is going to be connected in the Arduino UNO, so as demo we have learned how the parts are connected and know we can decide how the PCB of our project needs to be created and how the parts will be connect. Also, from this diagram we can take the pins that the parts will connect to in the CNC shield and what pins are going to be connected in the microcontroller.

This diagram gives an obvious and clear picture of how the stepper motors are connected and how the servo, stepper drivers, witches and the power source are connected which makes it easy for us to work on the PCB.

5.1.10 Testing the electronic parts of the project

We have ordered all the components that we need for the project, as we mentioned before the project will consist of electronic parts such as the stepper motors, Arduino, CNC shield machine, servo motor switches and the DC power supply, and the other parts that built the project such as the rails nets and the aluminum rods. For testing the project, we will only need to see if the electrical parts work perfectly and if not what we will need to fix so they can work, since our project needs a specific Firmware and Control Software to work and move motors, we could not operate it since we still working on the cod that we will be using it as soon as it is finished.

Even though we did not have the full code for the project we were able to collect the electronic parts and get them connected to each other in a proper way where they are ready to use. We have connected the CNC shield to the Arduino, and we connected the stepper motors into the CNC shield, and we attached the drive motors to it as shown on the picture below. All the parts are connected and ready to use and for now we are not missing anything that can cause a huge problem, or will cause a delay on finishing our project.

Also, we have tested and did some research to know what it the range of input voltage of the electronic parts that we have, so that we do not apply less or more than the required voltage to the electronic parts and the coming table shows some components and parts and the input voltage that is needs to use on them:

The component	Input Voltage
Nema 17 Stepper Motor	Up to 36V
CNC shield	12V
Arduino	5V
Servo Motor	6V
Nema 17 23mm Stepper Motor	4.1V
A4988 Stepper Driver	8V-35V
Limit Switch	250V

Table 9: Voltage Range of the Electronic Parts

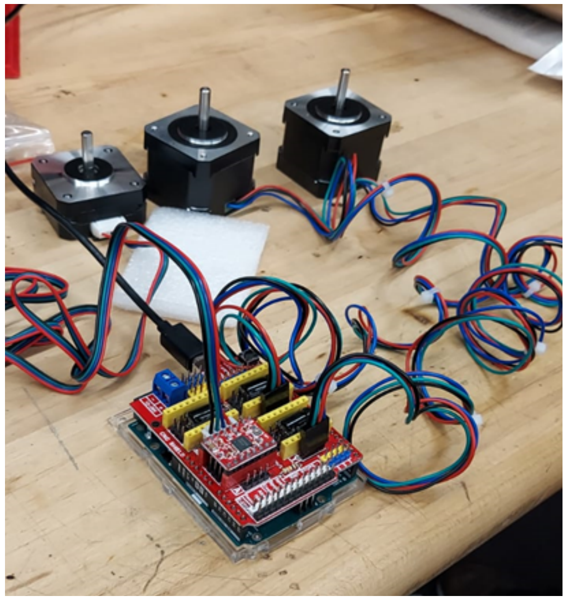


Figure 46: Demo Testing.

After we get the software done, we will start our experiments and testing the components to see what needs to be fixed and what needs to be corrected.

5.2 PCB

One of the most important steps of designing a project is working on designing the PCB for the project, basically after finish on working, planning and collecting ideas of a specific project it is important that you collect all the component and electrical part on an organized board that hold them and connect them properly and here where the job of PCB coms.

One of the most common components that is used in the electric field is the PCB (printed circuit board). PCB is a component that is designed at the beginning by a specific program to connect the electronic components electrically. Rather than having the parts connected to each other using wires. The PCB solved the problem by connecting each part using a copper foil which comes together in one green board that is ready to get the parts and solder the components on it. The PCB gives the project a perfect look and it shows it in an organized shape.

For the PCB that we are going to work on it has to be the board that holds all the electrical parts that are used for our project, so our PCB will consist of the power input, the microcontroller, three stepper motor drivers, and pins for connecting components.

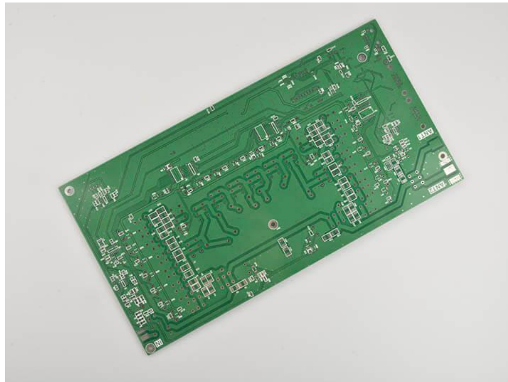


Figure 47: PCB Sample.

5.2.1 PCB Software Design

There are several different software that are used for designing PCBs. This software is used for drawing a certain schematic which will be used to create a perfect PCB by collecting the electrical components and organizing them in a way that follows the user specifications. Working on software is an important step in most electrical projects and it is necessary to make a perfect project. Most of the designing software provide all the needed components and parts for creating a PCB and each one of the design software has it is own features and advantages which makes it different that the others.

EasyEda, Egal and many other programs are used to create and draw schematics that are used to design the PCB. For our project we decided to use EasyEda. This software is an easy and powerful online PCB design tool which allows student engineers and educators to perfectly design their PCB. EasyEda has several features that make it different and unique from the other designing programs, it is easy, fast, online, and has powerful drawing capabilities. After reading, searching, and knowing about the software features we decided to use it for our project so we can come up with a perfect PCB design.

For our project PCB we will need to create three different schematics and connect them together for the project to work in a proper way. The first component that will need to be inserted in the PCB is the microcontroller and make the connections between it and the three stepper drivers which there will be three stepper drivers used in our project, and our PCB will have a MOSFET control for a power full servo.

5.2.2 Microcontroller schematic

After we did some tests on our demo project and collected some ideas, we decided to use the Arduino-nano-3.0 which consists of USB, ATmega328 and 32K flash memory. We have chosen this one because it suits the requirements of our project, and it has all the features that we need. The picture below shows the Arduino that we are going to use.

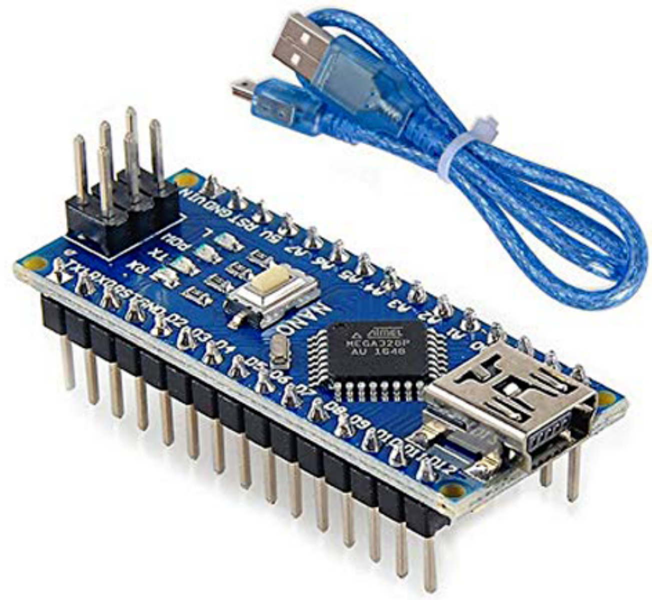


Figure 48: Arduino Nano.

As we stated before for our project, we will be using the Arduino to control can give orders to Stepper Drivers to move. The Arduino will have three stepper Drivers connected to it and the Arduino will work as the Master. In the schematic below we will show the Audio that we will be using and the pins that need to be used and how it is connected to the other components.

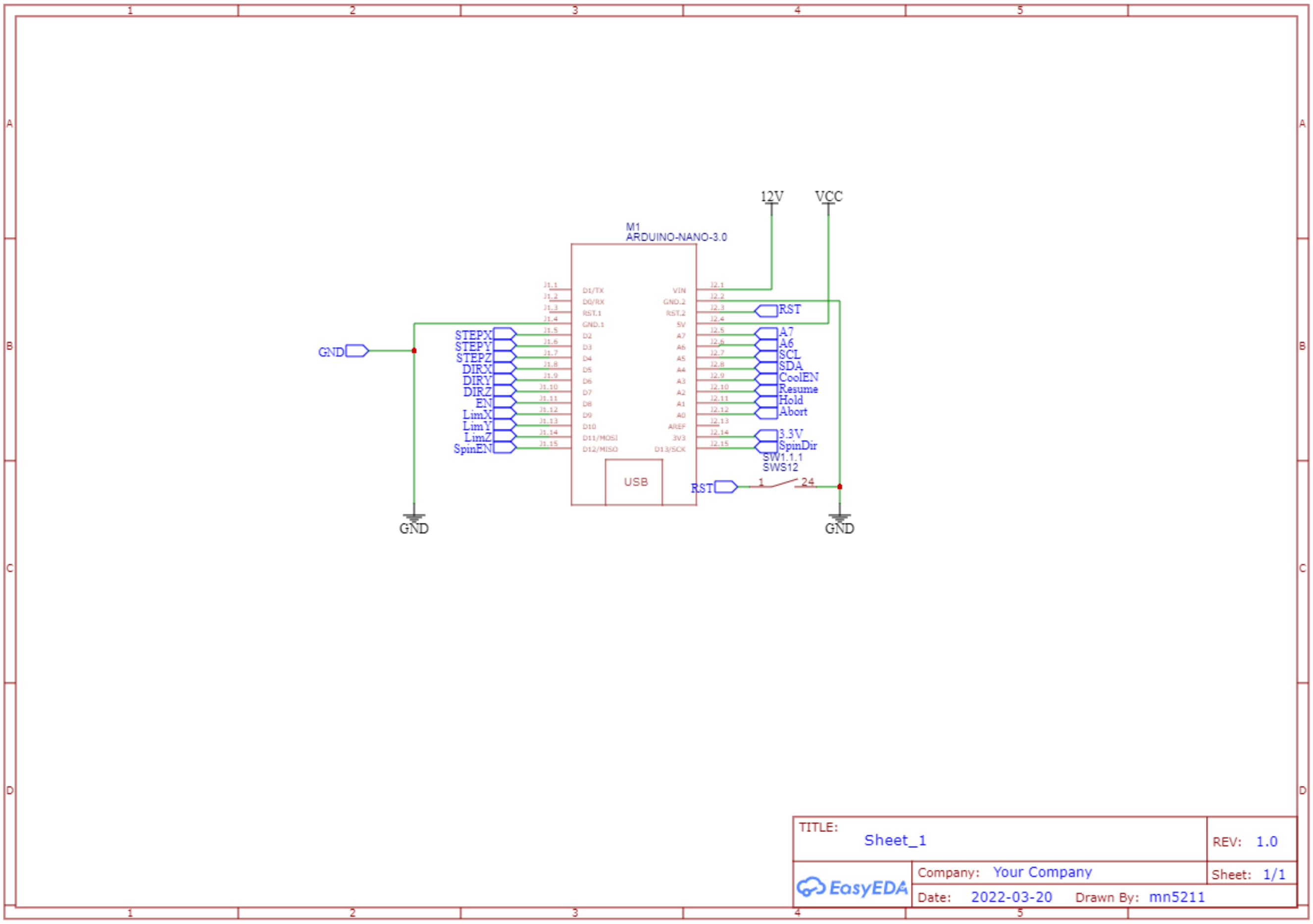


Figure 49: Microcontroller Schematic.

5.2.3 Stepper Driver schematic

There will be three stepper Drivers connected to the Arduino and each one of them control a different motor and they are connected to the Arduino as shown in the tabulable. Each one of the three stepper drivers will be connected to a ten kilo ohm resistor and each one of them will have pins which will be connected to the stepper motors and get signals from them.

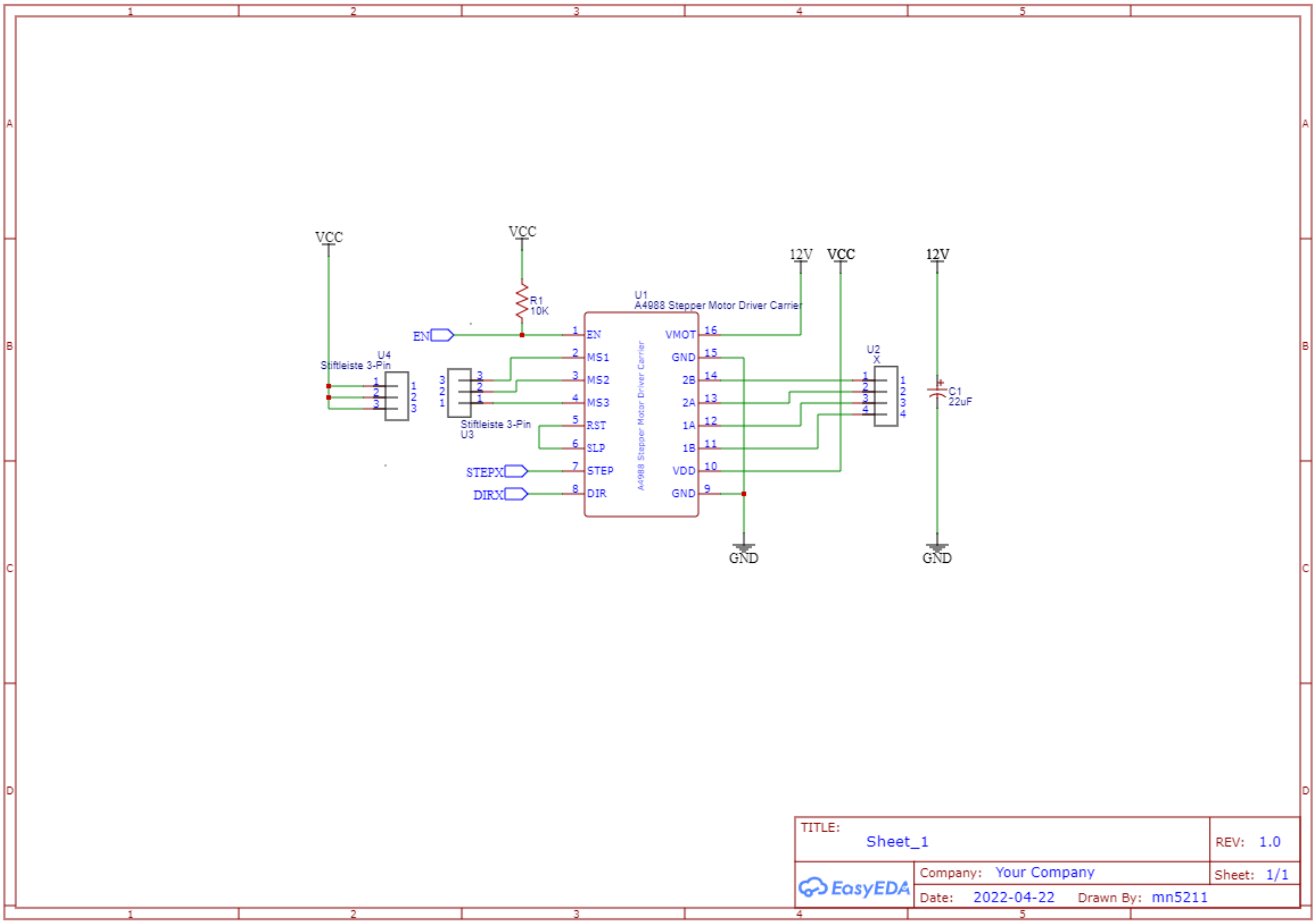


Figure 50: Stepper Driver X Schematic.

On our project we will be using three stepper drivers and all of them have the same schematic, so we will create three schematics for them since they are the same we will show one of them in a picture however they have different pins to connect with the Arduino which will be shown in the table.

Arduino pin	Stepper X pin	Stepper Y pin	Stepper Z pin
D2	STEP	-	-
D3	DIR	-	-
D4	-	STEP	-
D5	-	DIR	-
D6	-	-	STEP
D7	-	-	DIR

Table 10: Arduino Motor Control Pin Layout

So each one of the stepper drivers are connected separately to the microcontroller through using two different outputs and inputs and they are called STEP which means the Steph input and DIR which means direction input. As mentioned before, stepper drivers will be connected to pins that are going to send signals to stepper motors and the table will demonstrate how these inputs and outputs are connected.

Stepper Driver pin	Stepper motor pin
2B	PIN 1
2A	PIN 2
1A	PIN 3
1B	PIN 4

Table 11: Stepper Driver to Stepper Motor Pins

The two schematics below show the other two stepper drivers and how they are connected to the microcontroller and each one of them will be connected to different stepper motor.

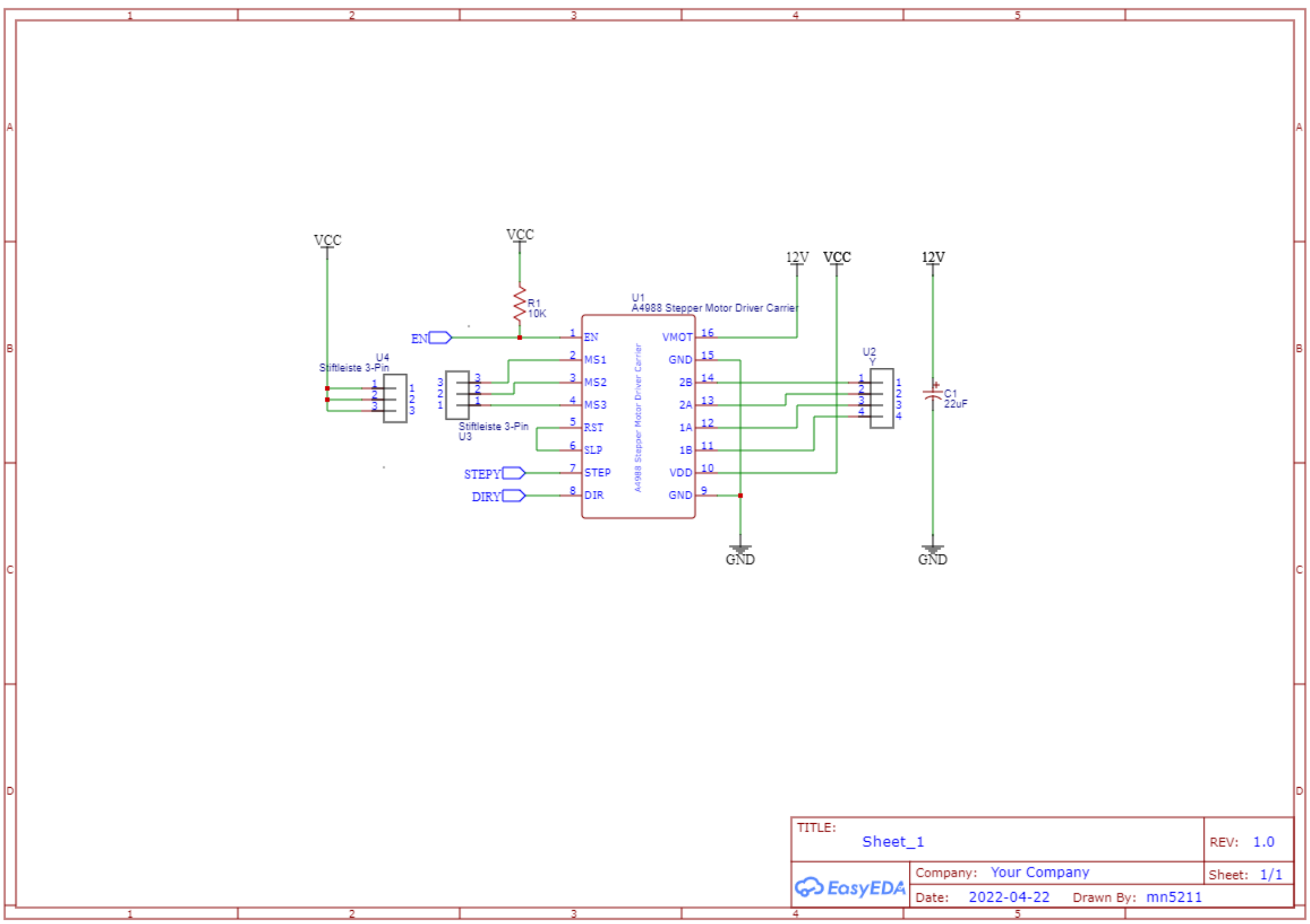


Figure 51: Stepper Driver Y Schematic.

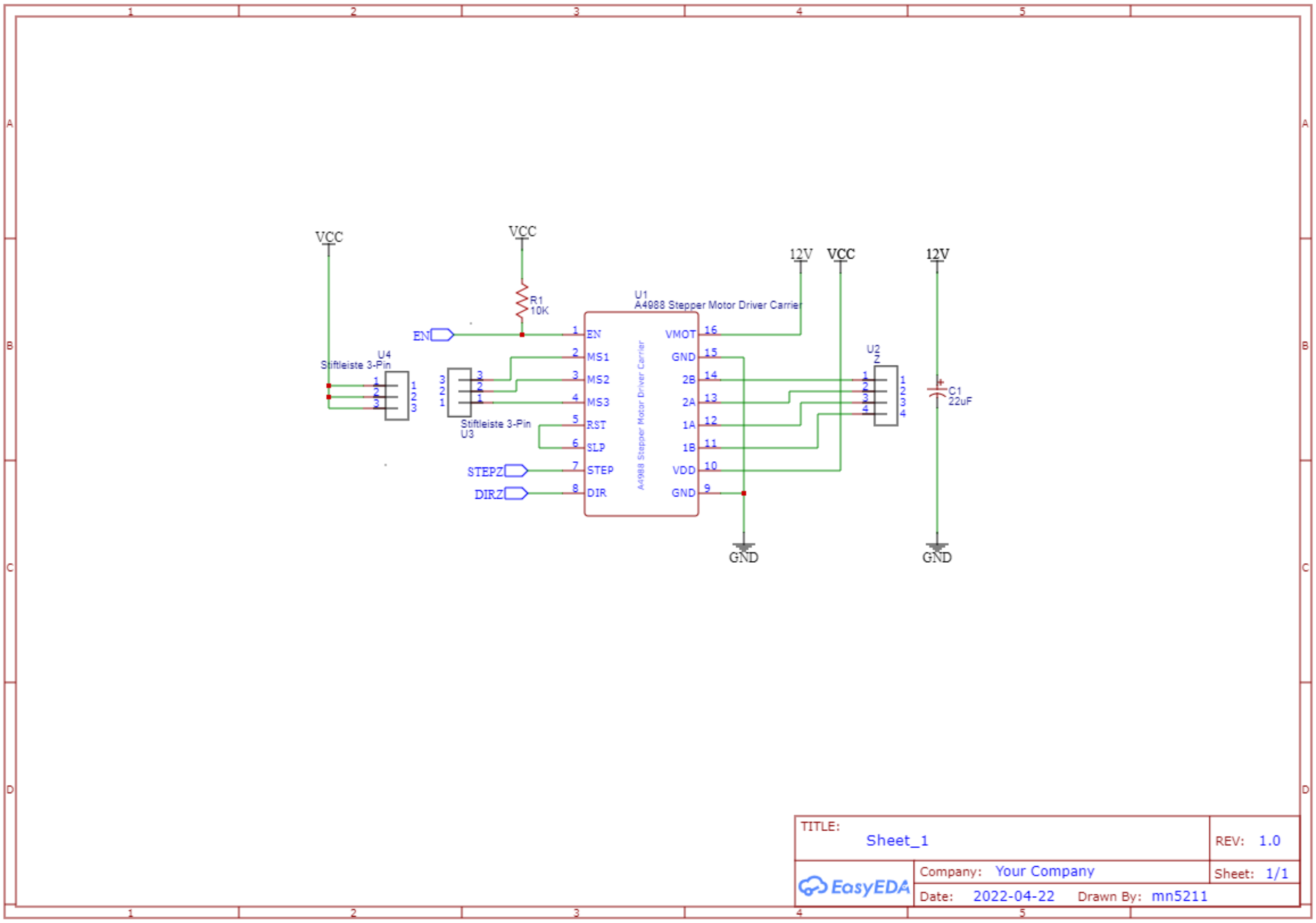


Figure 52: Stepper Driver Z Schematic.

These are all the stepper drivers schematics needed to build the PCB and each one of them has a specific output to connect to the stepper motor. For example, for the stepper that is going to move in the x-axis has output pins named X, for the stepper motor that is going to move on the y-axis has output named Y and for the last stepper motor that is moving on the z-axis has output named Z.

5.2.4 Full Schematic

After we did a separate schematic for all the components that we will be using for the project we have collected them together and they are ready to be ordered and soldered as shown below.

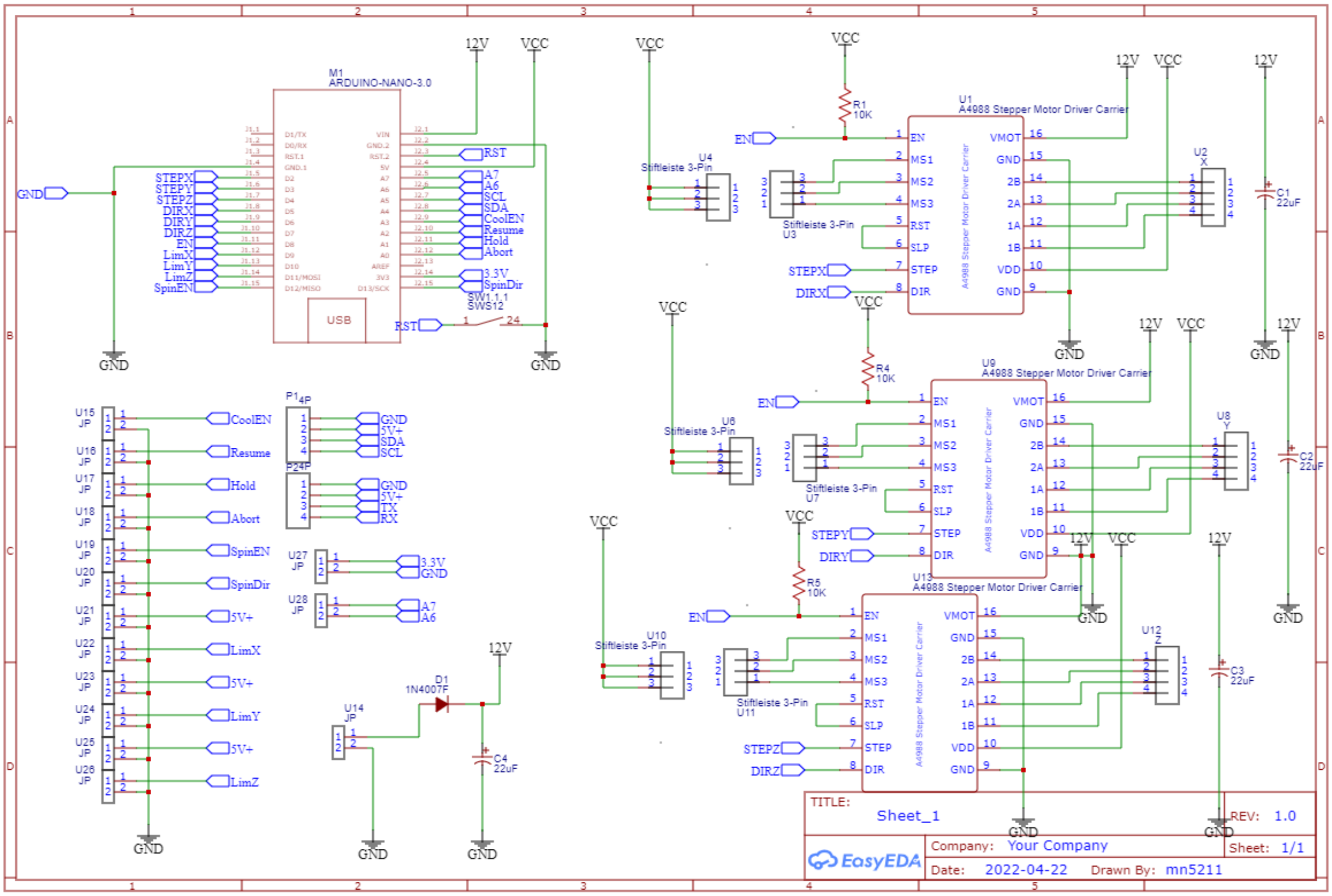


Figure 53: Full Schematic.

5.3 Hardware Block Diagram

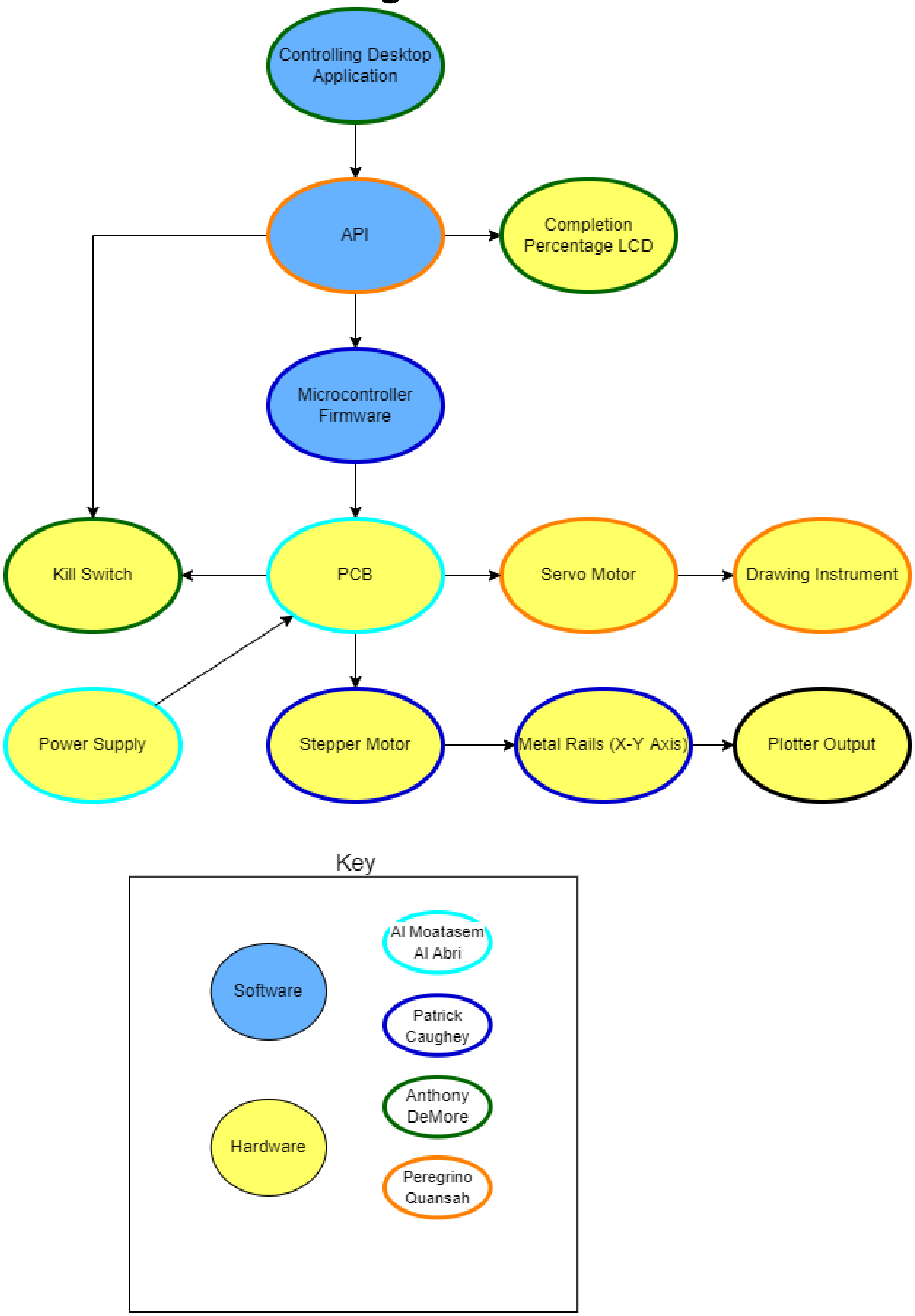


Figure 54: Block Diagram

6.0 Software Design

In this project, we will use two microcontrollers which will be designed and printed by our team, one for controlling the motors (the Computer Numerical Control ‘CNC’), and one for communicating with other peripherals. For the purpose of illustration and example,I am going to use the arduino CNC shield as an example for our microcontroller board that will be responsible for controlling and moving the motors. Arduino CNC is a microcontroller that provides the necessary electrical signals for controlling stepper motors. There are various options that are available when it comes to controlling stepper motors, some people opt for a dedicated controller but then the arduino CNC shield provides a wider range of possibilities because it is open source and there is a lot of documentation about it. It allows the user to control at least three different stepper motors which is very important for our design. In order to get this done successfully, we will need softwares to run on these microcontrollers. Apart from these softwares, we will also need programs that will let the user input commands, and send these commands to these microcontrollers for interpretation. Finally, if time permits, we will have a mobile companion application that will enable the user to send commands from their mobile phones. This software will allow users to input their design, convert the design into G-code, and the microprocessor board will interpret those G-codes to move the motors. Below is a use case diagram of our software.

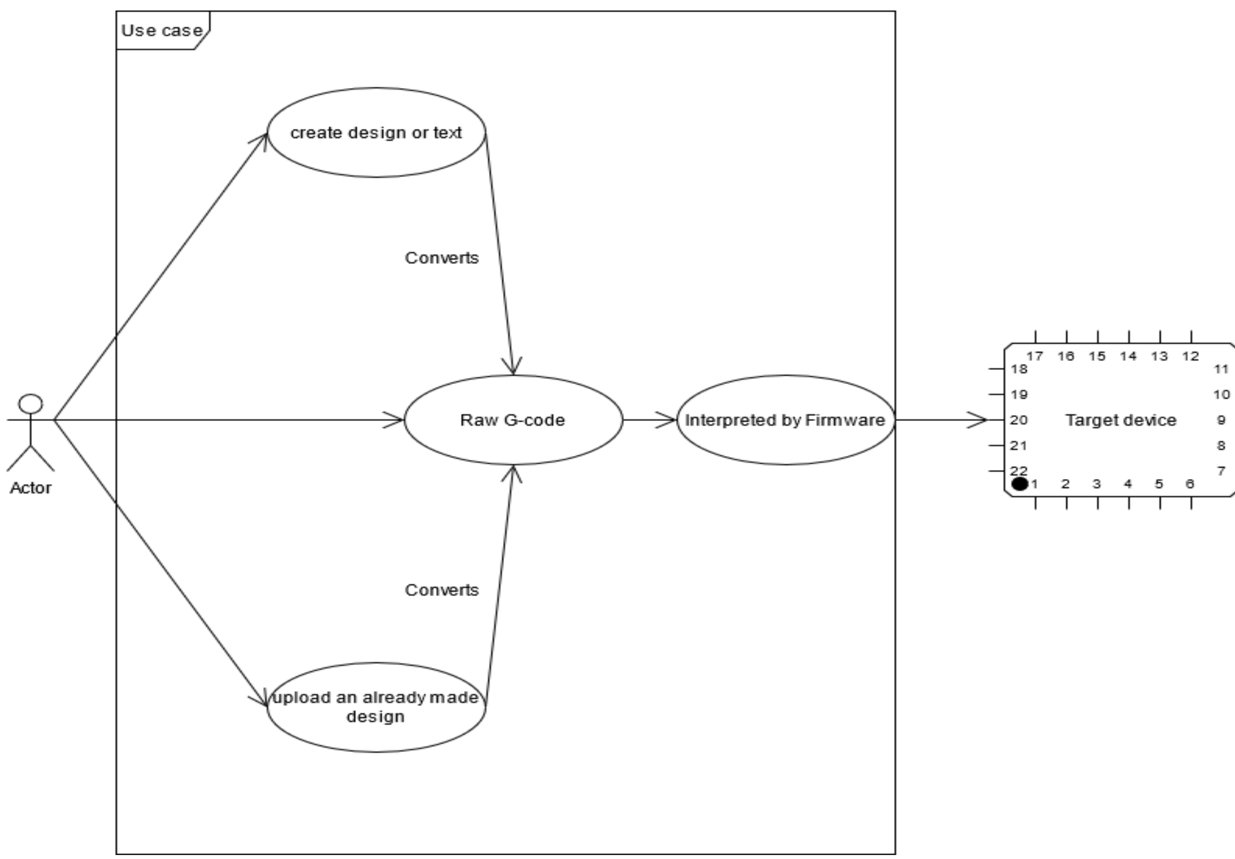


Figure 55: Software Use Case Diagram

6.1 Software development tools

A software development tool is a software program that gives the software developer the ability to design, create, maintain and debug software programs that are needed to solve a particular problem. For the purpose of our project, we will need at least a software editor and compiler, or an integrated Development environment (IDE). Also we will need a graphical user Interface generator because we will like to create our own user interface that will be used to take in inputs.

For a successful software design, the design team will have to use various tools that enable the team to successfully design software that works with the various microcontrollers and hardwares involved. We begin by investigating which programming languages that will best suit the various microcontrollers, like which firmware works for the arduino uno and the arduino CNC shield microcontrollers. Also we will need to research on which integrated development interface best works for the development of these softwares. Arduino has its own programming language which is similar to C++, but it can also work with high level languages like C, Java, and python. For the purpose of this project, we will design our software for the CNC shield microcontroller in C and the other softwares that will communicate with the microcontrollers will be developed in Java. We will use the arduino IDE for the development of these codes since it works with the various microcontrollers.

The design team also brainstormed on what language to develop the user interface and user program in. There are various options to choose from like Java and C, not leaving out python since it tends to work well with arduino. These aforementioned tools will be further discussed in subsequent sections.

6.1.1 Programming Languages

A programming language is a tool used for software development. They are a set of rules that convert given strings or characters, or even graphic elements into different kinds of instructions that a machine can understand. They are used to convey the thought process of the software developer to the machine or whatever they wish to control by the implementation of algorithms. There are different types of programming languages and each of them have their strengths and weaknesses depending on what they are used for. For the purpose of this project, we are going to focus on programming languages that are used for the development of user interfaces and also programming languages that help us communicate with hardware like microcontrollers. We will also compare and contrast various languages depending on the ability of the program helping us achieve our design objectives.

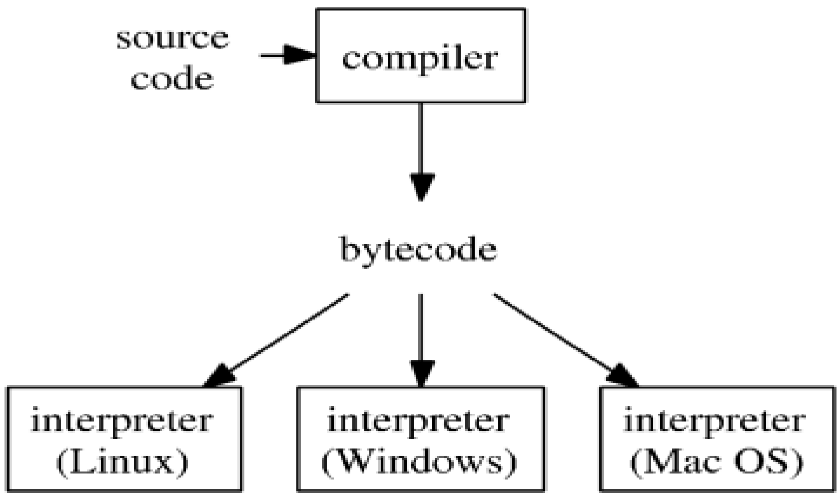


Figure 56: The Process of a Programming Language

6.1.1.1 C Programming language

The first programming language we are going to discuss in this section is the C programming language for the purpose of embedded systems. C is one of the most powerful and versatile programming languages to ever emerge. It is one of the go to languages for embedded systems programming. This is so because it has language is a low-level language, i.e. it is able to communicated almost directly with the hardware or microprocessor without a lot of interpretations compared to other programs like python. C also has a low memory usage during the compilation. C provides constructs that make it possible for it to map to typical machine learning instructions. Apart from embedded systems programming, C is also used in the programming of applications like operating systems and application softwares for various computer architectures. C was originally developed by Bell Labs for the purpose of constructing utilities running on Unix. To use C for embedded systems programming, one will require non standard extensions that support excotic features like fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations. Some of the major highlights of this language includes but not limited to the fact that C gives access to the most low-level system components through its built-in pointer. Software developers can create a compiler for their embedded systems projects very quickly and efficiently because of its popularity. C programming amalgamates the low-level functionality of the Assembly language harmoniously with the modern day programming ways. C also provides appropriate data types that makes it very suitable for embedded systems programming. C makes it easier to port an embedded systems program from one device to another compared to programs written in other languages.

6.1.1.2 C++ Programming Language

C++ is a general purpose programming language that was developed by Bjane that serves as an extension of the C programming language.The language has evolved over time and the current C++ has object oriented features, together with generic and functional features. Just like C, C++ facilitates low-level memory manipulation and it is always implemented as a compiled language. Compilers for the program are very readily available courtesy vendors like Free Software Foundation, LLVM, Microsoft, intel, Oracle and IBM. C++ was designed mainly for systems programming and embedded systems programming with performance, flexibility and efficiency as its hallmark.C++ has also gained popularity in other fields like desktop applications, video game programming, servers and performance critical applications. Some highlights of C++ includes the ability to let you use inline functions as a replacement for macro definitions, a feature that is not readily found in C. C++ also boasts of the implementation of templates and verbose casting that makes it faster than C. Another reason why C++ is the go to choice for embedded systems programming is the ability to use overloaded functions and constructors, not forgetting the object oriented feature that allows developers to come up with the most complex embedded system software without overflowing memory. C++ has the native constructor and destructor feature that introduces a lot of versatility as compared to C.

6.1.1.3 Java Programming Language

Java is a class-based, high-level, object oriented programming language that is designed to possess very small implementation dependencies as possible. One of the main purposes of Java is the fact that it provides programmers the luxury of writing once and running everywhere. This means that once a java code is compiled, it can run on any platform that supports java without the need to recompile. This is done by compiling java code into bytecodes that run on any Java virtual machine without the underlying computer architecture being a hindrance. The syntax of java is just like C or C++ but it has a smaller number of low-level facilities compared to either of them. Also java runtime provides very dynamic capabilities that are normally not found in traditional programming languages. Java was originally developed at Suns Microsystems as a core component of Suns Microsystems Java platform. Java provides automatic memory management through the implementation of the automatic garbage collector to manage memory in the object lifecycle. When a programmer creates an object, java runtime is responsible for recovering the memory once the object is no longer in use. This feature saves time and also prevents memory leaks that would have occurred while using languages like C.

Java also provides a very lucrative platform for the building of user interfaces to provide very good user experience. We will expand more on this topic in the user interface section later.

6.1.1.4 Python Programming Language

Python is a general-purpose, high-level programming language which has design emphasis on code readability with the use of significant indentation. It aims to help programmers write clear and logical code on all scales and levels through its language constructs and object-oriented approach. Python also implements garbage collection, a feature very prominent in Java and it is also dynamically typed.Python supports different programming paradigms such as object oriented, functional programming and structured programming. It contains a very standard library and it is often described as a ‘battery included’ language. Python’s standard library, which is very much well known as one of pythons strength, provides a wide range of tools for different tasks. It includes modules for creating graphical user interfaces, unit testing, automation, computer networking and embedded programming. Python requires a runtime when python programs run. It also makes it easier to write automated tests for your embedded systems that require real time analysis. Python excels when it comes to the field of image processing and real time data processing. It also provides a wide range of cutting edge networking libraries for writing the network portion of your embedded systems program.

6.1.1.5 G-code Programming Language

G-code is one of the programming languages used for CNC (Computer Numerical Control) machines. G-code is the shortened form of “Geometric Code”. We utilize this programming language to tell a motor-controlled machine what to do or how to do something. The G-code commands direct the machine where to move, how quick to move, and what way to follow. In the case of a machine instrument such as an automated cutter, the cutting instrument is driven by these commands to take after a specific toolpath, cutting away fabric in order to induce the specified shape. Similarly, within the case of added substance fabricating or 3D printers, the G-code commands taught the machine to store fabric, layer upon layer, shaping an exact geometric shape.

6.1.2 Integrated Development Environment

An integrated development environment is a software development tool that provides all the necessary facilities for software developers to develop a complete program. IDE’s mainly consists of a code editor used for writing code, build automation tools and a debugger for debugging the code. This tool is very important for this project because it will provide a holistic environment for our software design team to write and test the software for the various microcontroller pcbs we have in this project. We will be able to follow along the execution process of the code using the dubbeger so as to identify any bugs that may occur during the development of the software.

When a software engineer is developing code for various reasons, being it for microcontrollers, mobile devices or desktop platforms requires a number of steps that are cut across. These steps include a text editor that enables you to write code in the desired language, these texts are then converted into machine readable code which is suitable for the target device. This is done by the compiler or interpreter depending on which language being used and finally the machine-readable code is then uploaded to the target device. Below is a visual representation of this process.

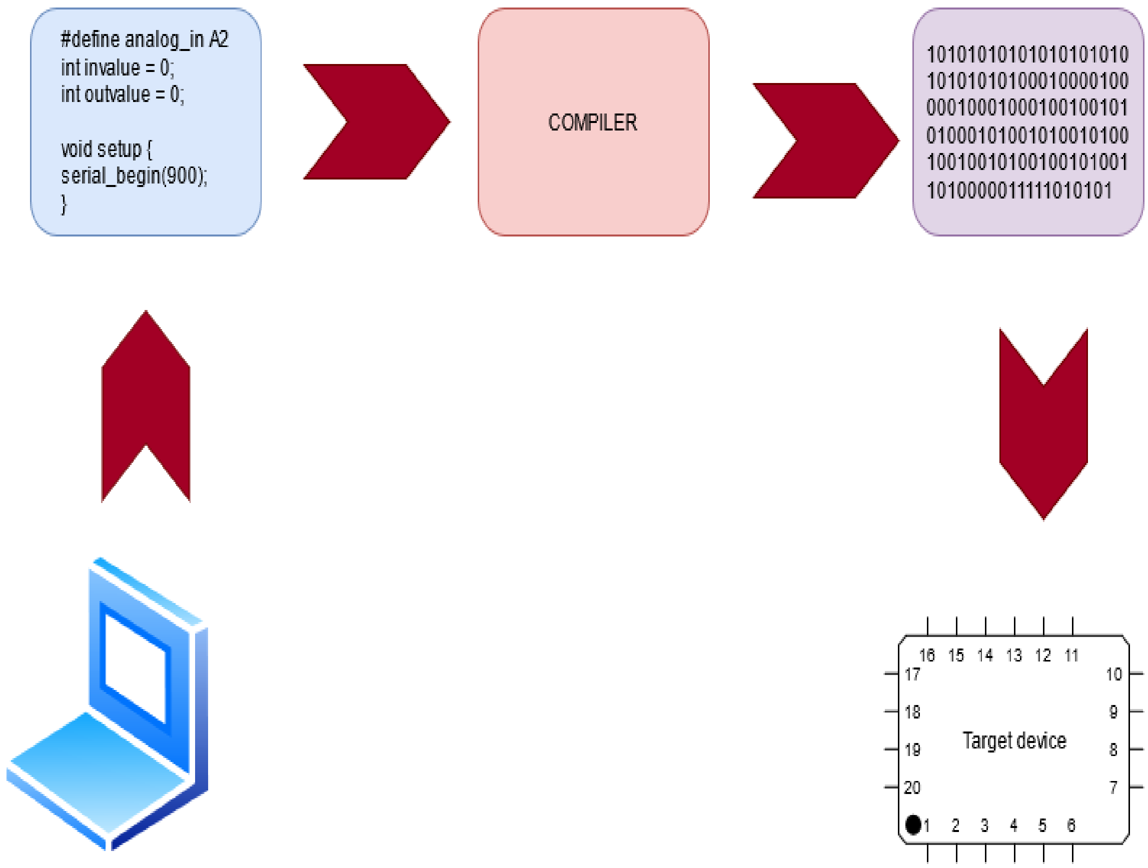


Figure 57: Software Cycle

We will compare and contrast these IDEs and then decide on one that actually works best for the purpose of this project.

During our hunt for the perfect IDE, we came across several IDEs that can be used to create code for our arduino testing microcontroller. We will compare and contrast them and then select which one will be more suitable for our project.

PlatformIO is a very well known arduino IDE. It is mainly used as an integrated development environment for Internet Of Things(IoT) projects. This IDE supports not only arduino boards but other boards such as Raspberry, ESP8266, ESP32 and the likes. The IDE has the normal features that one will expect in a regular IDE such as code completion and a debugger. It also supports multi projects, has different themes and a built-in command line. One of the outstanding features of this IDE is that it requires python to run, and it can run on several operating systems. It is used by several well known companies like Texas Instruments, Energia, FreeScale, just to name a few. It is installed as an extension for visual studio code and has various libraries that help you with whatever project you are working on. It also boasts of a built-in terminal with PlatformIO Core (CLI) and powerful Serial Port Monitor. Based on its popularity and available documentation, our team decided to go with this for the development of software for our microcontroller board.

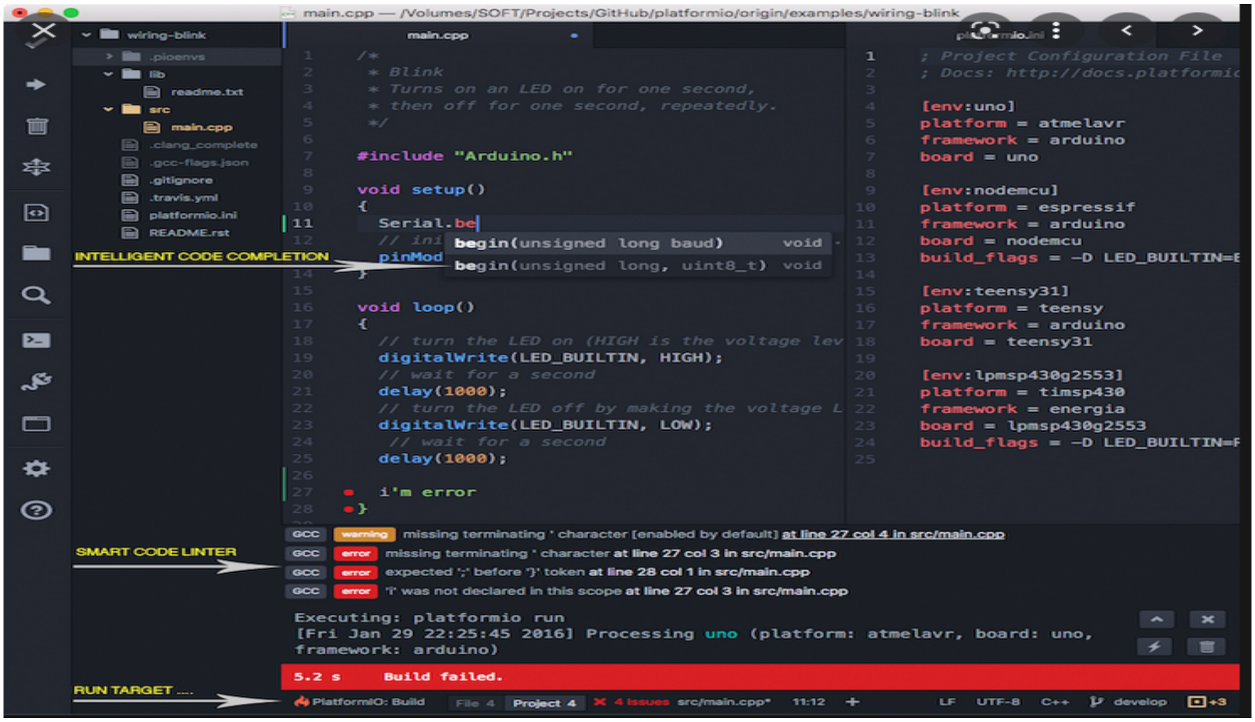


Figure 58: Showing Various Features of PlatformIO

Another IDE worth talking about is the Eclipse Arduino IDE. It is not a stand alone IDE but a plugin for the Eclipse IDE. It has the same features as the standard Eclipse IDE which is best known to support Java coding styles. It comes with a standard arduino serial monitor which helps in debugging arduino codes.

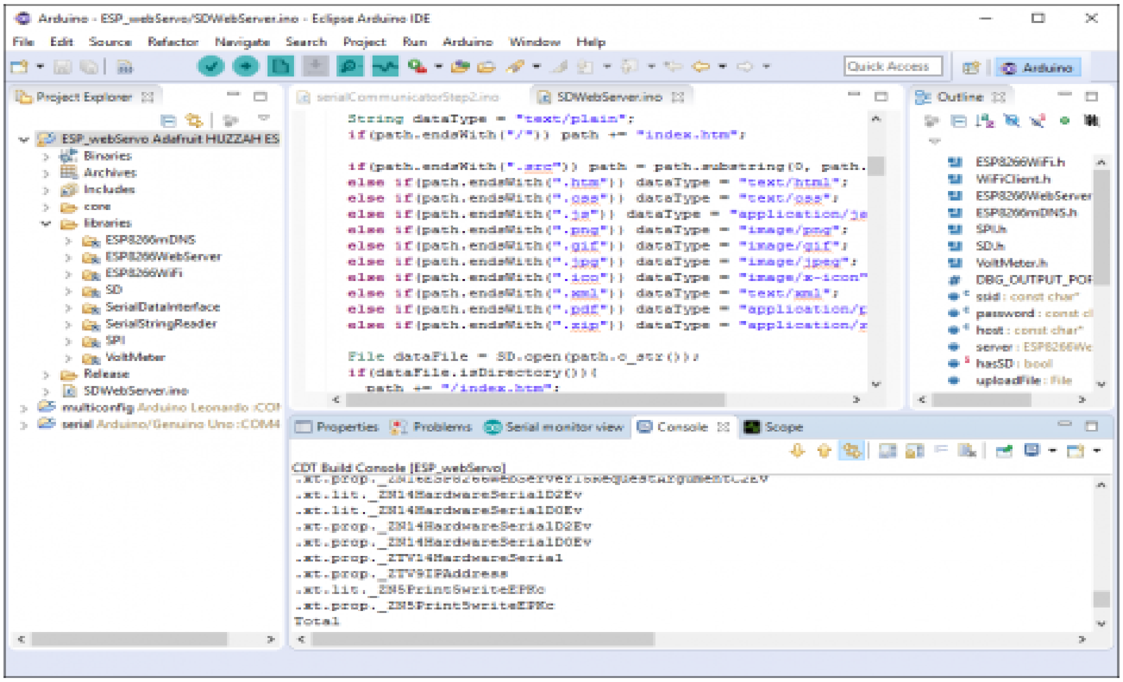


Figure 59: Eclipse Arduino IDE

The final IDE we came across is the standard Arduino Integrated Development Environment. It was developed mainly for arduino boards but supports other microcontroller boards too. It is very simple to use even for beginners. Based on its popularity and available documentation, our team decided to go with this for the development of software for our microcontroller board.

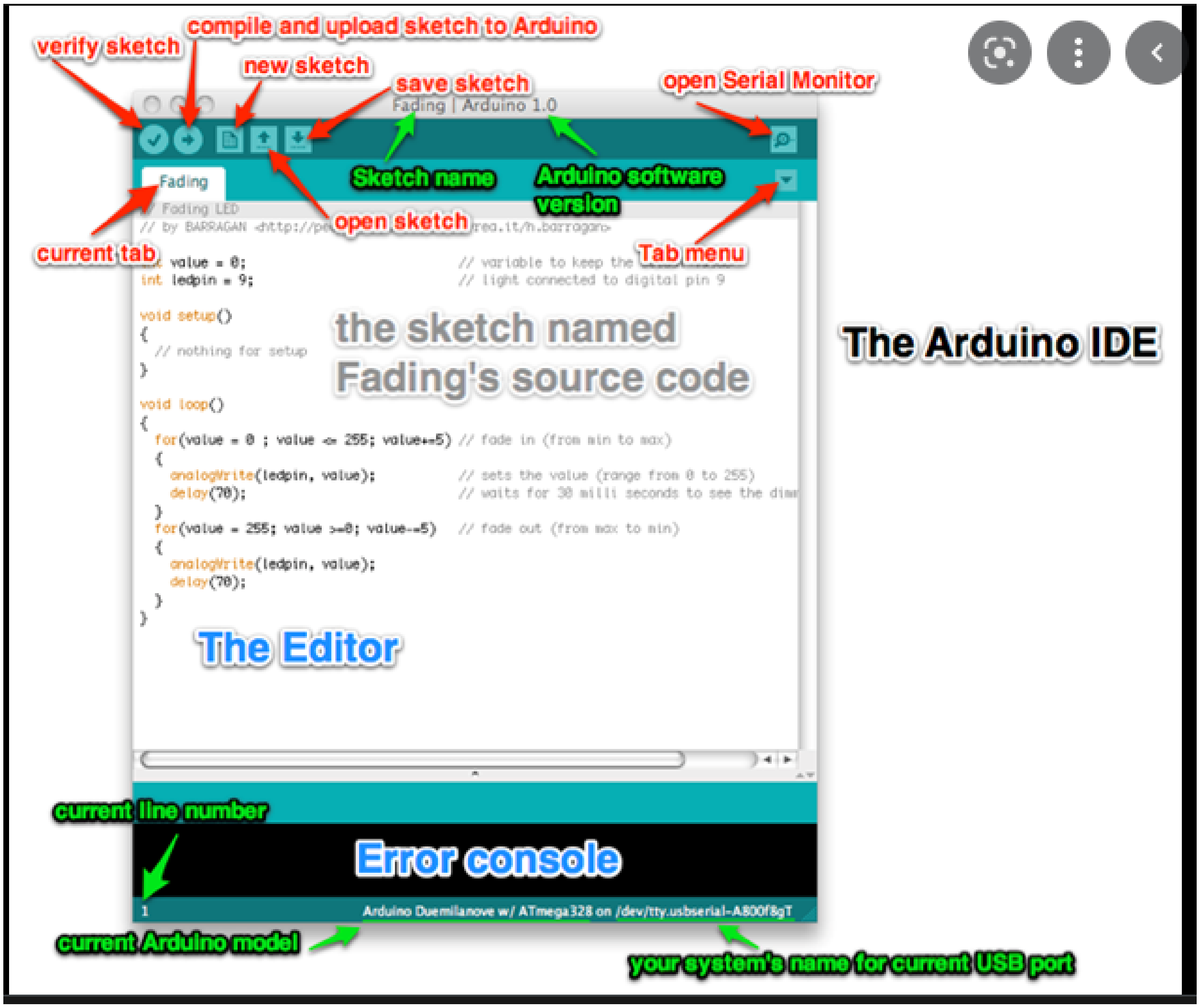


Figure 60: Arduino IDE

	Arduino IDE	PlatformIO
Text Editor	YES	YES
Compiler	YES	YES
Communication with target device	YES	YES
Supports various boards	YES	YES
Supports various languages	NO	YES
Auto complete	NO	YES

Inline error checking	NO	YES
Onboard reference help	NO	YES

Table 12: Arduino IDE Vs PlatformIO

6.2 Overview of Software Modules

The software needed for the project can be divided into three parts. First, we will need a software that takes in whatever figure we wish to reproduce with our machine. This will also serve as the software the user gets to interact with. The functions of this software will be converting the input image into G-code.

G-code is one of the most popular and most widely used computer numerical control (CNC) programming languages. This kind of programming language was first developed at MIT servo mechanics, a department responsible for creating machines that moved with the help of motors.G-codes are basically instructions that tell the machines where to move to, what path to follow and how fast to move. The reason they are called G-codes is that they usually start with G.

After our input has been converted to G-code, we will need another software to send these G-codes to the firmware of our microcontroller. These kinds of softwares are normally called G-code senders. They serve as a communication tool between the microcontroller firmware and the input data. When these G-codes are received by the microcontroller, they are then translated into signals that tell the motors where to move, what sequence to follow and the speed needed. All this movement will be done by the microcontroller firmware. Below is a high level diagram showing the various software that will be needed for this project.

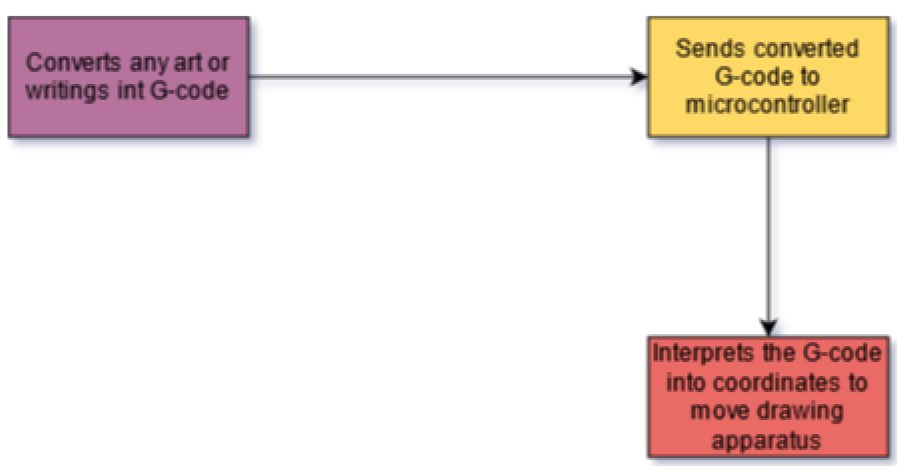


Figure 61: High Level Software Flow

6.2.1 Computer Aided Machining Software Module

Computer Aided Machining (CAM) Softwares are responsible for converting designs to instructions that the controller software module can understand. For this project we will not write our own CAM softwares because it is time intensive, we will choose from a variety of free and open-source software that are available. In this section we will go through all the available options and compare them so decide which software will be suitable for our project.

In our quest to design a software that can convert our scalable vector graphics input into a G code that can be interpreted by the microcontroller for the motor to to use, we can across a classical well known open source software that is normally used for CNC projects like this called Inkscape. Inkscape is a free and open-source vector graphics editor that is used for making vector images in scalable vector graphics format. It allows for the importation of other formats and can render very primitive vector shapes and text. Inkscape can boast of very important features like object creation. This feature allows for free hand drawing with pen tool, pencil tool and other calligraphy tools. It also has a shape tool that allows users to draw basic shapes like rectangles, ellipses, polygons and spiral objects. Still on the object creation feature, inkscape allows for a text tool that allows users to create texts selecting from a wide range of fonts and nice designs. It also has embedded bitmaps that use commands to create and embed bitmaps of selected objects. Another useful feature of this software is that it allows users to manipulate the object they create. Users can perform transformations, thus moving, scaling, rotating and skewing a created object both interactively and by specifying the exact numerical values. It also allows the user to work around the z-axis(vertical). Thus the user can raise and lower objects. All these features, and the ones not mentioned are very useful but one feature that will be very useful in our case is the ability to add extensions that convert these vector graphics into G-code. Inkscape provides the ability to add extensions that can convert the created vector graphics into G-code. An example of such an extension is called the g code tool, which creates the G-code and has the ability to add other features like the orientation type, change the layer types and many others. Using Inkscape as an inspiration and template, our design team will implement certain features of this software since it is open source and also add personalized features to it to meet the needs of this project in particular

Another open source software readily available that coverts the various types of input graphics into G-code and sends them to the microcontroller firmware is the GRBL-plotter. The GRBL-plotter is mainly focused on the post-processing of vector graphics into various forms. We can use this software to convert any given vector graphics input into G-code based on the properties of the imported file including but not limited to resizing and sorting by shortest path that the microprocessor can use with the help of the firmware.

Openbuilds is also a CAM software that is open source and has a lot of supporters in the online community. It also comes with it control software that allows your to connect directly to the microcontroller board and communicate with it. Openbuilds lets you import various file and converts them into g-code of which the controller software can understand and interpret. The only downside of openbuilds is that you will have to use their controller software together with their CAM software for it to work. You cannot combine it with any other controller software. It also needs active internet to work since it is internet based.

Universal Gcode sender (UGS) for short is a gcode platform that enables the user interface with controllers that are responsible for controlling the motors by reading the G-code sent to it. UGS is a java application that has several external dependencies which can run on almost all operating systems including Windows, Linux and MacOSX. Some of the features that comes with this software includes a 3D gcode visualizer that will enable us have real time tool positioning and also feedback. It also provides us with duration estimates. Thus how long it will take the tool to map out our design on the provided paper. It also has a web pendant interface that allows us to use out phone via a common port and ip address to send gcode to our machine for it to work. This also come with over 3000 lines of unit test codes that allows us to test various features and makes sure they work. With this kind of software, we can be sure to achieve our set goal for a design that can help us view progress and have a preview of our tool path before we begin sending the gcode to the controller software for action. For this project, we intend to go with the UGS software because it allows us to customize and take out what we do not need and also add to it by writing certain features.

6.2.2 Control Software Module

For there to be communication between the computer and the motor that moves the various parts, we will need a controller software. A controller software is responsible for reading the G-code and then runs the stepper motor that drives other moving parts of the machine. All these cannot be done without a firmware so therefore it is very essential to flash the microcontroller board with a firmware that will make it possible for the board to communicate with the controller software.

GRBL which is pronounced gerbl is a very popular firmware that makes it possible for Computer Numerical Control (CNC) configurations. GRBL is open-source and is mostly used to flash microcontroller boards to enable it to support CNC configurations. This kind of firmware provides the means to control the 3-axis motion of a CNC machine using a computer with the help of a Universal Serial Bus(USB) interfacing. The reason why this is very popular is that it is fairly easy to set up and has very simple configurations that make it very understandable, also users can tailor it to meet specific needs. The two types of control softwares available are the G-code senders, one which simply passes on the G-code to a microcontroller board that has been flashed with a CNC firmware and another that can actually interpret the G-code and provide instructions for the onboard motors to use. In our case, we will go for the simple G-code senders because we will want to have total control at each stage of the software execution process and be able to intervene or rectify any mishaps that may occur.

TinyG CNC controller software is a universal serial bus based 6-axis software that enables microcontroller boards work with 6-axis controllers. It also supports XYZ linear axis and ABC rotary axes with a 4-motor output. It was mainly design for small CNC applications with the aim of satisfying projects with highly controllable motion controls.

6.3 Graphical User Interface

User interface is very important in the development of the software for our project. UI is a place were the user of the machine can communicate with the machine in order to put their ideas across. The importance of the UI is to allow a very efficient and effective control and interaction of the machine by the human end. At the same time, feed back from the machine is also displayed on the UI for the user to understand what is going on and make informed decision. For our project, we intend to design our own user interface from already existing open source softwares that have several added on features that we will not need. Our design will be usable, thus it will be simple to use and only the serve the purpose of this project without any extra unsable features. Also our design of the user interface will be very useful. It will fulfill the need of the project making sure the goals we set out to achieve are met. We will also have accessibility in mind when designing the user interface. It will provide feedbacks like the progress of the drawing of the object onto the paper, and a simulation of the tool path. This will help us make informed decisions.

In order to build a good graphical user interface, we need to consider what language will best suite our needs and also what language our CAM software was written in to ensure compactibility and accessibility. Since we will be using the the Universal Gcode Sender CAM software, we will intend use Java to build our graphical user interface because our CAM software is written in java. Java provides several tools for building graphical user interfaces like JavaServer Faces (JSF) which is based on the MVC architecture. This framework from Java is a component based user interface framework. This means it allows users to drag and drop components around to build a GUI while doing all the work behind the scenes allowing the user not to worry about knowing the main technology behind it such as Javascript. It also do away with induction of a new framework because it provides the needed and available backend java code the ability to entend a web interface. This is a very convenient choice if one does not really want to spend time on building a GUI for a project.

Another framework for building a graphical user interface in Java is by using the AWT or Abstract Window Toolkit framework. This is one of the oldest java graphical user interface frameworks available by Java so therefore it is one of the most thoroughly tested and well documented user interface module. The only problem with this framework is that it is outdate and lacks certain modern and advance components needed to build a graphical user interface. It is mostly used for smaller projects that has very small and simple features. The AWT frame work hierarchy is composed of an object. Every object can contain a components and these components can comprises of several things such as buttons, labels, checkboxes, lists, choice selection and containers. The containers also can hold a window or a panel and a panel can hold an applet. A window on its own can hold a frame our a dialog which is mainly used to give feedback to the user about their actions on the gui.This framework will be very useful for our project if we were building a simple user interface but for the fact that our project will have various features and complex actions, this will not really be suitable but then it provides us with a basis for other java frameworks. Below is a figure showing the hierarchy of the AWT

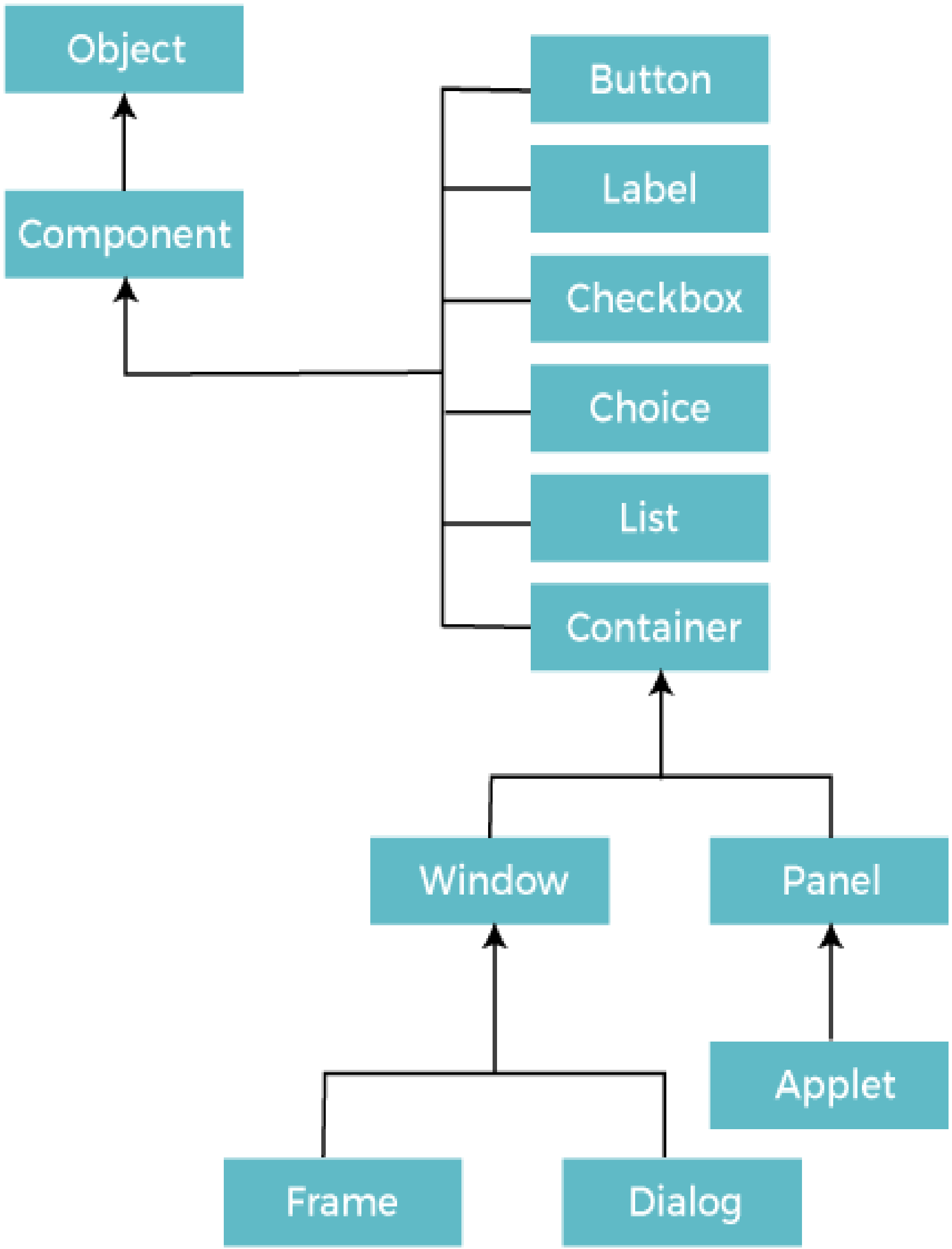


Figure 62: Hierarchy of AWT

Swing is also a Java framework that was built on AWT to fix and replace some of the shortcomings of AWT.It also add extra features to it which makes it more modern.It also has the same hierarchy like the AWT and can be used for more complex project as compared to AWT. Even though they Swing was built on AWT, they have very significant difference which will influence the choice of one over the other. Below is a table showing some of the notable differences between Swing and AWT.

Java AWT	Java Swing
Components are platform-dependent	Components are platform-independent
Components are heavyweight	Components are lightweight
Does not support pluggable look and feel	Supports pluggable look and feel
Less components	More components
Does not follow Model View Controller(MVC)	Follows Model View Controller(MVC)

Table 13: Difference Between AWT and Swing

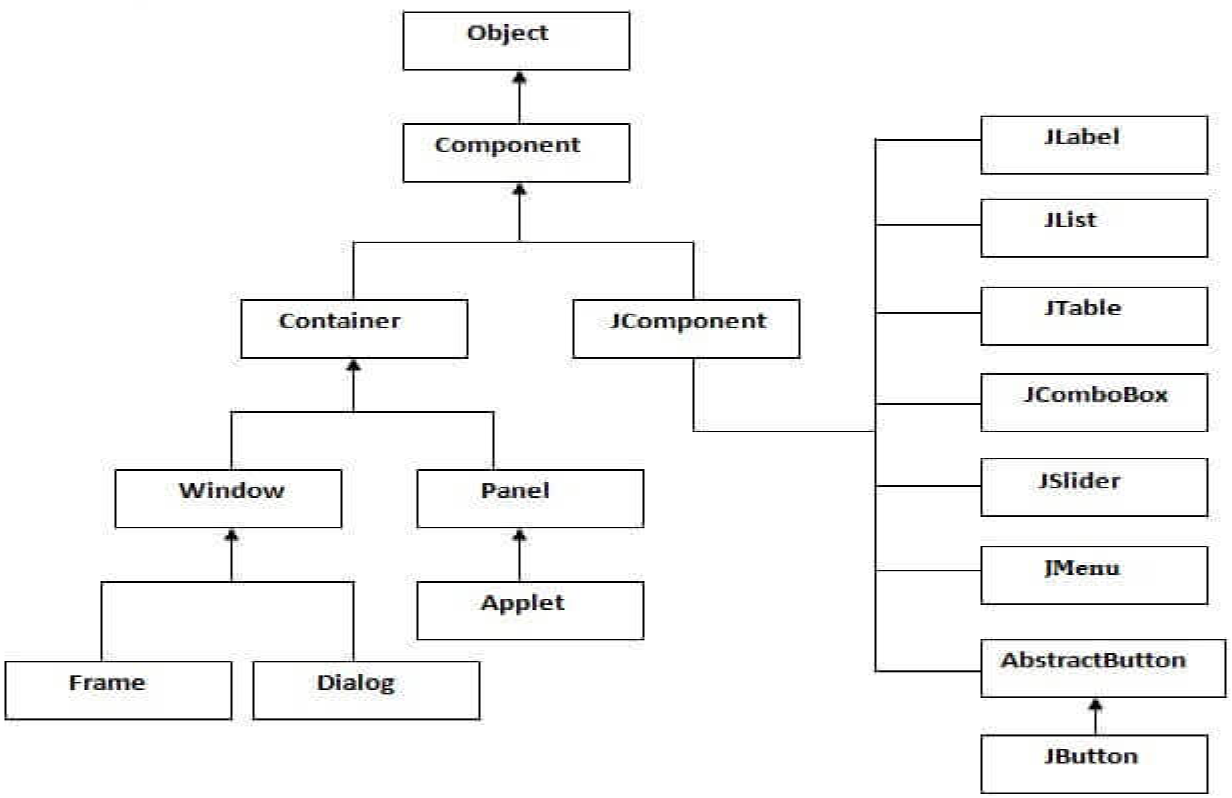


Figure 63: Hierarchy of Swing

JavaFx is one of the latest java frameworks used to make graphical user interfaces. It comes to replace swing with a variety of rich desktop components and rich internet applications. Applications that are developed using JavaFX has the ability to run on mobile, desktop and the internet. One of the interesting features of this framework is that it provides components that do not rely on the operating system. This means it supports every operating systems, very lightweight and hardware accelerated. It comes with a design tool called Scene Builder which reduces the work the developer has to do by providing the ability to drag and drop components onto a window to create a graphical user interface. Some of the features of scene builder includes a drag-and-drop interface. It also has a live editing and preview feature that allows you to edit and preview as you build your user interface. Scene builder also gives you a complete access to JavaFX gui controls library making it very convenient so as to prevent any unnecessary import and installations. You can also add custom made graphical user interface components to the scene builder and use them in your development. It supports 3D gui development by loading FXML documents containing 3D objects. Properties of these 3D objects can be edited using the inspector panel. It is cross-platform and it supports CSS to enhance looks and feels of the user interface. It starts with a stage where the entire user interface is based on. Then we have a scene which has a scene graph that holds other components in a form of nodes that can have parents and child nodes. For example, a region node can hold a pane as a child node and another controls node as another child node. Because JavaFX is more advanced and has the ability to run on phones, we will use this for our project in building an accessible, usable and useful graphical user interface for our project. Below is a figure showing the hierarchy of JavaFX.

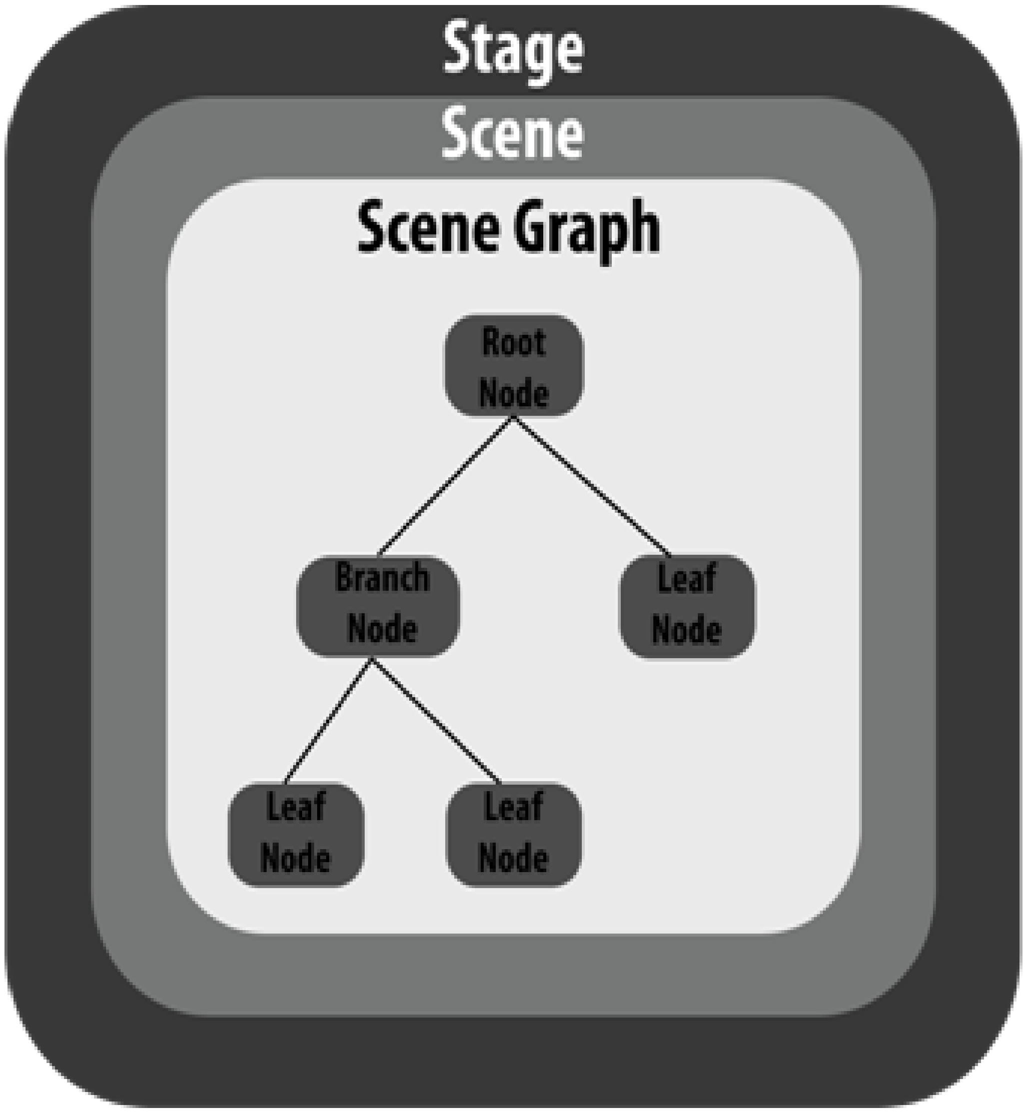


Figure 64: JavaFX Hierarchy

The initial design of our Graphical user interface is one that will contain basic functionalities that will be needed for our project. The initial idea is to build a user interface that can run on various operating systems for the most part. It also should have a visualizer feature that will help use visualize the image we intend to plot and also show the dimensions and other details. We also intend to add three different panes, one for the control actions, another for the state controls and last one for console and overrides. The action control pane will house buttons that allows us to control the X, Y and X axis of our drawing tool while showing use the step size and measure in millimeters. It will also house common action buttons like start and stop. The state controller pane will also house the locations of our drawing tool, thus the X, Y and Z locations and also some functions like active state and idle state. The console pane will show the debug of the Gcode running along with when the machine is operating and will

show the type of command running. All these will be done using the netbeans ide which is ideal for making java programs and graphical user interfaces
Below is a flowgraph showing a prototype of our proposed graphical user interface.

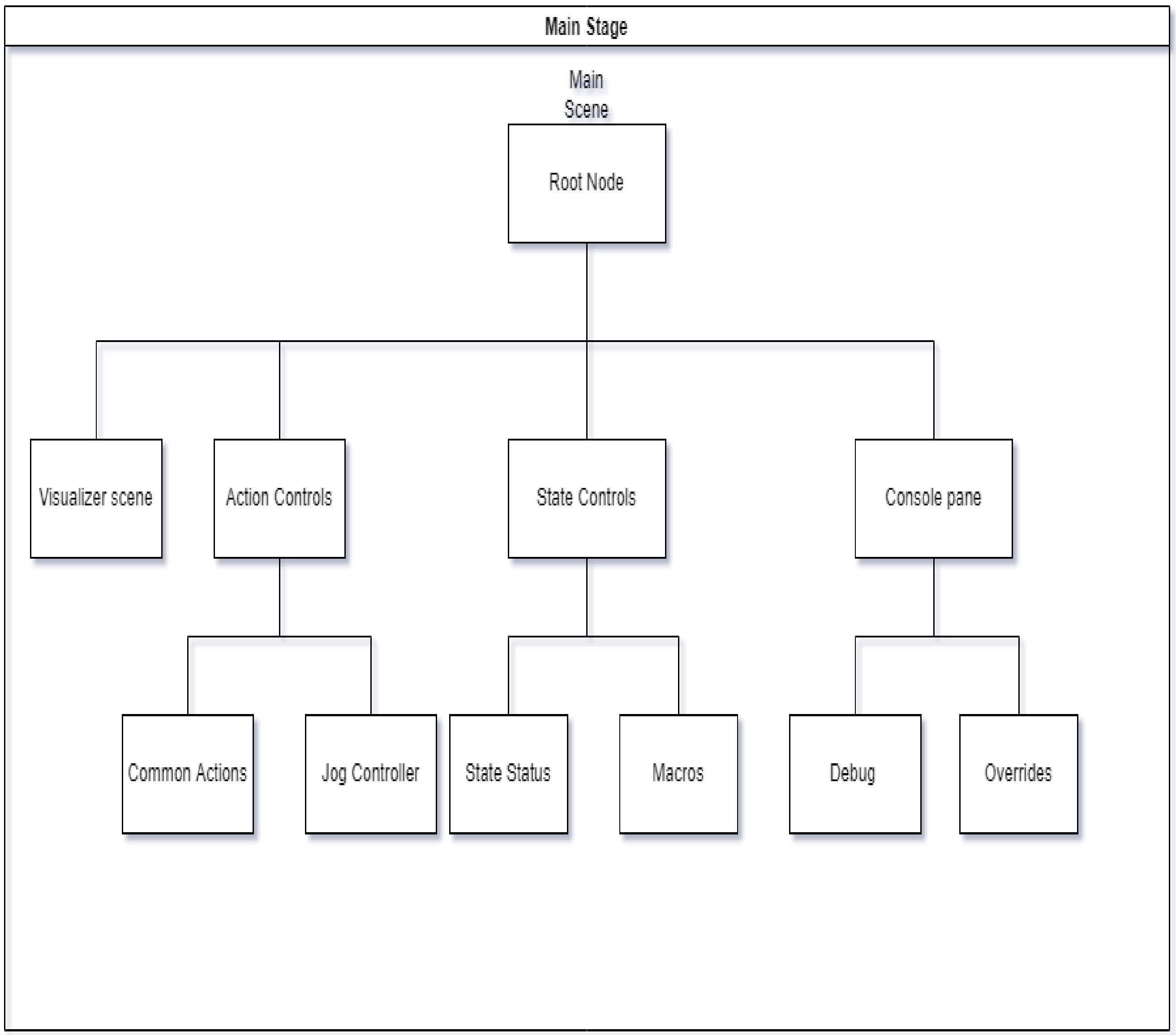


Figure 65: Prototype of GUI

6.4 Mobile Application (Stretch goal)

We also plan to use a mobile application that will enable us send data from out phone to the machine. This mobile application will use a bluetooth model to communicate with the plotter. There are already mobile Computer Aided Machining softwares available for free. So if time permits, we will integrate a mobile application and a bluetooth model to our project. This will require that we connect an HC-05 or HC-06 bluetooth module to our plotter. Some examples of already existing mobile CAM softwares includes the CNC bluetooth commander for android, March 3 and the likes. These softwares are mostly for devices running on android and not IOS. This will be a very advance feature we would like to pursue even after the completion of this project.

7.0 Mechanical Design

This section will detail the mechanical aspects of our design and how all of the moving parts are combined to form a working pen plotter. The mechanical parts are anything controlled by the hardware that do not contain any electronic parts themselves. Our design is heavily influenced by mechanical aspects which needed to be heavily researched by our group, since we did not have a dedicated mechanical engineer. and this section will detail how the hardware will create a functioning product.

The design mainly consists of two metal rails with a sliding piece on each allowing for movement of the pen in both directions on the plotting area. There are motors controlling separate pulley systems which allow accurate and quick movement. The motors are controlled by microcontrollers and they are mounted on 3d printed mounting platforms. The platforms are specifically designed to fit our pen plotter. There are many different shapes of 3d printed mounts all serving their own purpose. The following sections go into greater detail on some of the main mechanical design areas of our project.

7.1 X-Y-Z Axis Table

The X-Y-Z axis table corresponds to the movement directions of our pen plotter. With x and y axes representing the movement of left, right, up and down. With the Z axis representing the upwards and downwards movement of the pen with relation to the drawing table. This section details how this works mechanically and why certain parts were selected in order to create mechanical efficiency in our design and to allow for our expected outputs of art.

On top of our work area sits the x-y direction moving plotter. The main parts used for the mechanical design of this are the MGN15H linear rails. Which are two large metal pieces which hold our design together and allow for the movement of the plotter to take place. The linear rails we chose for the x-y design must be rigid enough to ensure stability throughout the process of the plotting. These rails have a sliding metal piece on top which allows for the smooth movement of the pen along the rails.

The metal piece slides with small metal balls inside of it to allow for this smooth and seamless travel, a very important aspect of our mechanical design. Without this smooth and stable travel, our plotters output would not be nearly as clear as intended. In order to ensure the prolonged smoothness of travel, we will apply oil lubricant to the rollers before assembly. The majority of the linear reals are bolted together using m3 sized machine bolts along with a few different sized m bolts.

The metal rails need to be robust enough to support the weight of itself, the pulley system and some of the electronic components such as the motor. The metal rails are one of the most expensive components of our design since it has been machined to be able to support its weight while being able to support fluid and effortless movement.

Our design also consists of Z-axis motion which is used for the actual movement up and down of the motors and mounting points of the pen. This movement is done by the use of two metal rods which rotate on bearings which allow for the up and down movement as the commands are sent by the microcontroller. An example of the movement you will see in our pen plotter can be represented by the 3-Axis plotter basic diagram that you can be seen below:

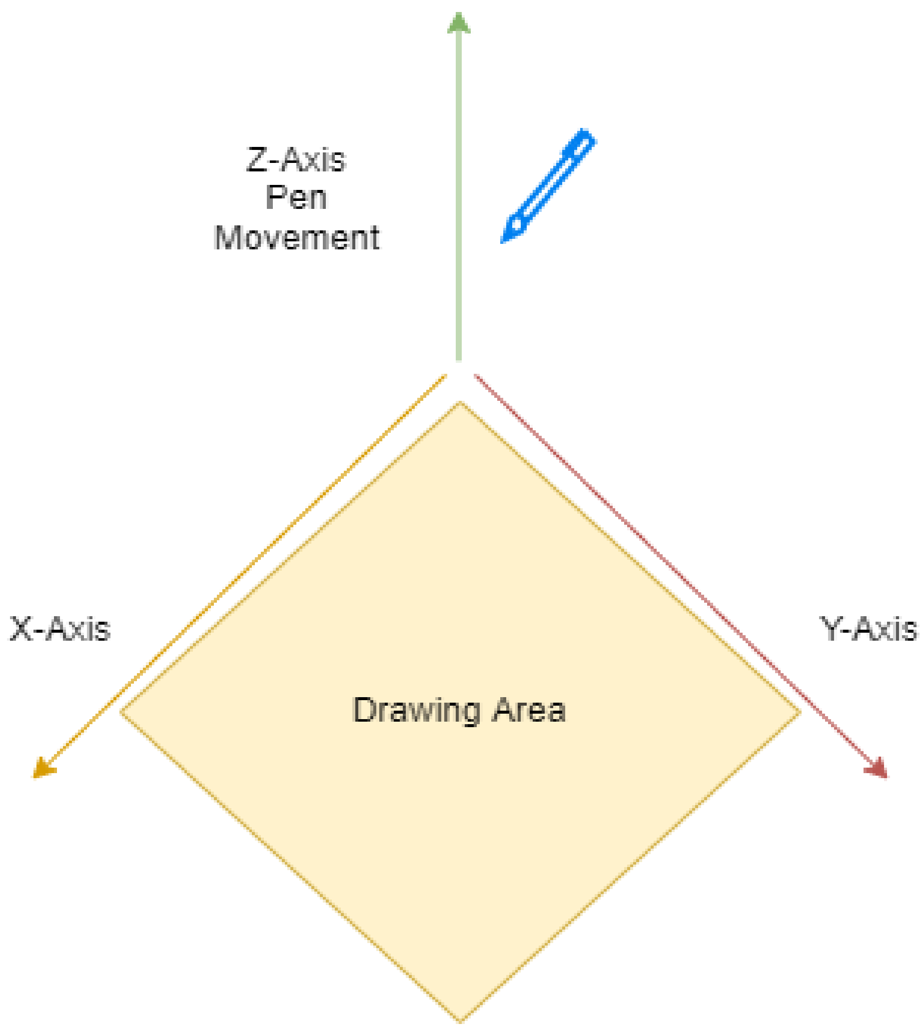


Figure 66: Plotter Movement Directions

Mounting plates will be applied to the linear rails in order to mount the electronic parts for the pen plotter as well as the various servo motors needed to provide the movement for our design. These mounting plates are 3d printed based on the specifications set by the x-y plotter template we will be using. They will be used for many features from lifting the plotter off the ground to running some of the pulley system. The sliding of the rails is done via a pulley system. These pulleys will be detailed in the following section.

7.2 Pulley System

The pulley system is what will be the mechanical part of the design that allows for the servo motors to move the linear rail to allow for the plotting process. It is very important for the pulley system to be strong and reliable. It is also essential for the pulley to be able to handle quick and consistent movements. The pulley is crucial for our design to work. There are pulley bearings or wheels, which spin and allow for movement of the pulley's wheel. The belt is attached to the pulley wheel at each end. An example pulley attached to its wheel which is attached to a mount for movement along a metal rail can be seen below:

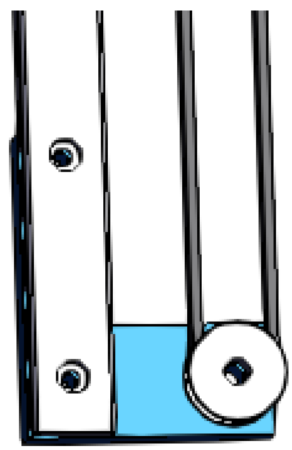


Figure 67: Pulley System
Image via Fangs.com

The belt is also attached to the mounting piece which holds the pen as well as the controlling motor for it. The belt moves back and forth based on the servos inputs and this allows for the movement along each axis. This pulley design is used on both x and y axis rails of the plotter.

The belts we purchased were quite large so they could be cut to the needed size and reconnected so they continue to work. There are 3d printed parts which connect belts to the sliding rail. This is what really allows for the device to move back and forth on both axes. The tension of the belt will need to be adjusted as well as seen fit by the group in order to have a proper tension so there is no delay in the movement of the ray when a command is read by the processor.

7.3 Mounting Modules

The 3d printed mounting modules which are an integral part of our design also heavily influence the mechanical aspects of our design. Some of the parts work with the other machinery such as the pulley system as well as the sliding rails. The mounting modules as we will call them are also used to secure the electronic parts such as the servo motors as well as the microcontrollers during the plotting process. This section will cover the mechanical aspects of these parts and why their structure is important to the functioning of our design.

There are many intricate 3D printed parts so we believed outsourcing the creation of the is our best option. One important piece of the mechanical design are the pulley holders which can be seen below:

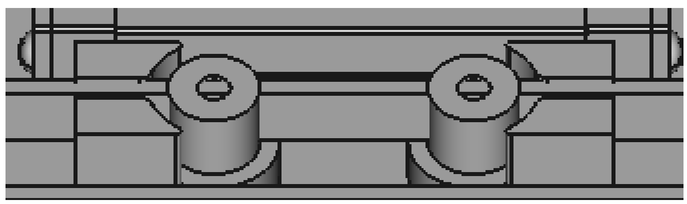


Figure 68: Mounted Pulley Connectors

These pulley connectors are able to hold the saw shaped pulleys in place and they have enough friction to be able to pull the rails along as the parts move.

It is important for the mounting modules to be robust and rigid enough to handle repeated use as well as the drilled holes need to be positioned in a way that they do not crack when screws are fastened in place.

8.0 Testing and Integration

After finishing all requirements of the project, we have to do some testing for our project, and we need to predict the challenges and the obstacles that might face us during the project testing. We will be testing our project in different stages such as, hardware, software, PCB and paper. The testing will take place in the engineering lab in the university and we will be using the lab tools and equipment to test our project and to see what needs to be changed and to figure out solutions for the problems that we are facing.

This section will also contain testing on hardware based software elements as well as purley software aspects. The hardware elements will be the microcontrollers firmware and the software elements will be the user interface, API connecting interface and firmware as well as the mobile application. The mobile application is a stretch goal but it will still be included in testing as it will also need to be thoroughly tested as it would be a key element of our design if completed.

8.1 Hardware Testing and Integration

For this part we need to make sure that all the parts that we ordered for our project are correct and complete because there might be some mistakes during delivering or manufacturing the components, so we need to be ready for such problems and always have another plan. Also, we need to make sure that the voltage input and the current input for all the components that we have ordered are as required in our project to work perfectly. What we should do here is check every part and make sure it has the correct features if not try to contact the seller or try to get a replacement as soon as possible.

In order to make the project work as required we need to make sure that the rails we are using has the same dimensions that we need otherwise we will face problems during the work on the output of the project, o the rails are important and they must be positioned well on the top of the wood base that are going to make and this wood needs to be thick and has good length and width so the rails does not face any issues during the movement of the ben holder.

One of the most important things that we need to make sure that is well placed are the limit switches. It is necessary that we place limit switches in a way that matches the dimensions will be giving to the motors to move we do not want that the rails and the pen holder moves in a random direction which will cause problems on the output of the project, so we need to calculate the dimensions and sync it with the given code to the motors. By mentioning the switches and the demotions it is important that we do not forget about the paper used to put the output on the top of it where this paper has to have a specific dimension that sync with the motion of the rails and the pen holder. Also, we have to know the speed of our project and how much we need to increase it or decrease it, so this way we can get a perfect output.

While testing the parts we need to know the quantity of the parts that we will need for our project and dimensions for the parts that we are going to use. For example, we will be using three stepper motors one for X, one for Y and one for Z and we will need three switches, one for confirm, one for cancel and one for emergency stop. Also, there is a specific dimension that we need to follow and test if the components are within these dimensions or not. For example, the working area needs to bae a larger dimensions of A4 and U.S. Letter Sizes with additional margin of 0.5 Inches, the base size needs to be plus 100 mm on top, 30 mm on each side, and 30 mm on bottom, the plotting precision 0.1 mm or less in the X and Y directions, the plotting speed expected to be 20 mm per second which is a reasonable speed for writing and drawing, also we need to make sure that the clamp fitting for pen fits most standard pen which is 6mm to 14mm. We will be expecting to have all these requirements on our project and we will test them using the lab tools.

Making sure that every part used to build the project is the perfect one is very important to end up with a well-prepared project, and to accomplish that we need to make sure and check if that is the correct part used or not. Also, we need to make sure that we measure and test all the parts that we are going to use and the quantity that we are going to use in our project.

8.2 PCB Testing

For this section we will do several steps before we get the PCB. First of all we need to make sure the PCB size is120 mm x 100 mm or smaller, then we will need to design the PCB in a perfect way where every part needed or required is in the PCB, we do not want to make any mistake on the PCB because it is a part the we order it form a manufacturer and the manufacture design what we have submitted and gave them, so we need to make sure that we are using the correct and inserted part in the PCB. Also, there is something we need to know and be prepared for that some parts are not available, or they are out of stock. In this case we need to have an alternative part for it.

After ordering the PCB we will have to check if our order is correct and as planned then we can start on placing the components on the PCB and before placing any component we need to make sure it is the correct one in the correct place with correct requirement, we have a specific way to test each component before using it. For example, the three stepper motors we will use using the breadboard, Arduino and stepper driver connected to it, as in the diagram below.

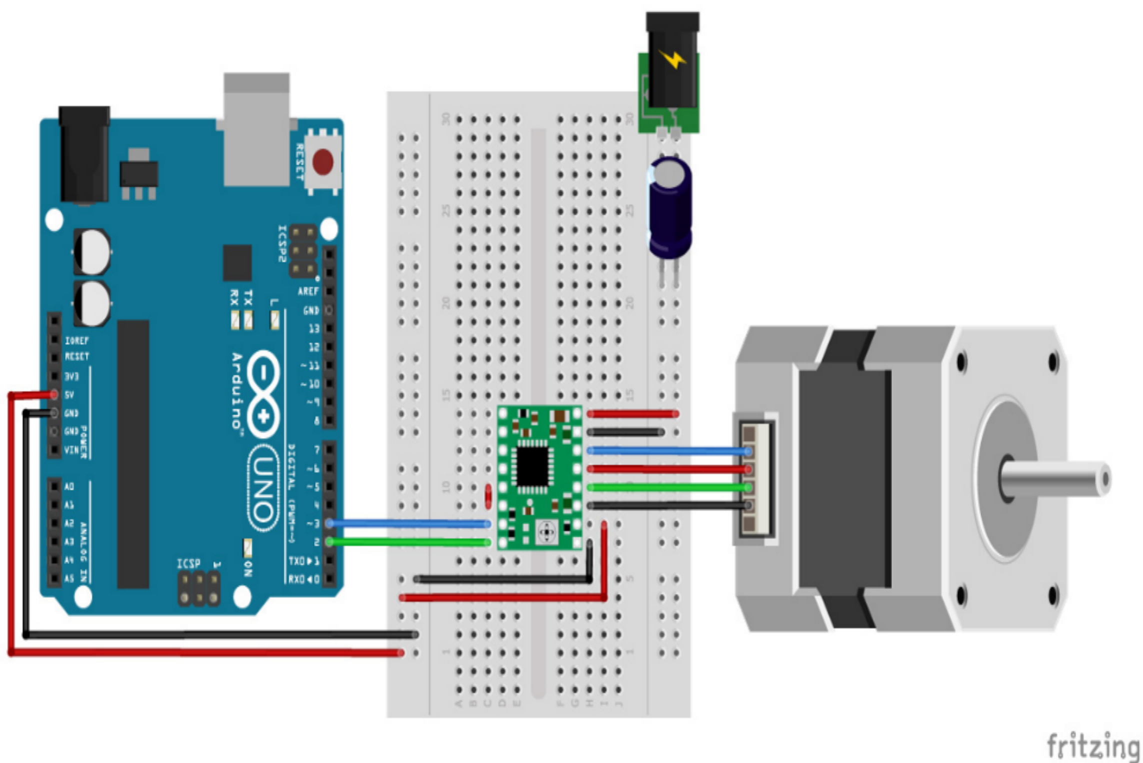


Figure 69: Testing the Stepper Motor.

This diagram shows how the testing of stepper motors will be, there are three stepper motors used in our project and we will need to test all of them to make sure that they have the required specifications. Also, while testing the stepper motor we will test the stepper drivers to see if they have the required features or not.

Also we need to test the Arduino and to see if it is the proper one for our project and if it has all the outputs to connect the components with and we will test it by trying to connect all the components that need to be connected to the Arduino. Also, we need to make sure that the microprocessor power supply is 5V, and the motor power supply is 12V. This table below shows all the PCB components and the voltage and the current needed in each one of them.

The Part	The voltage range	The current range
Stepper Motor Bipolar	4.1V	1A
Stepper motor driver	8V to 35V	1A
Servo Motor	4V to 6V	1A
Arduino Nano	Up to 21V	950mA
Limit switches	250V	5A
Stepper Motor	36V	2A

Table 14: Appropriate Voltage and Current for Each Part

After we make sure that all parts are correct we will start on soldering the parts and while doing that we need to work on it perfectly because there some parts that are small and need to be focused will solder it.

In this section, we have to make sure that all the components that we have placed in the PCB are correct and they match the ones that we need for our project because any mistake in that will cause issues and will affect our project. Also, after ordering the PCB and collecting all the required parts all parts must be placed and soldered properly on the PCB and we have to know that we cannot make mistakes during pacing and putting parts on the PCB. To avoid any mistakes and problems during soldering the parts in the PCB we are planning on getting a PCB map, so we know where to place every part.

One of the most important things that we need to make sure of is that the power source we are using is the correct one because if we put a lower or higher voltage into the PCB that will cause several damages on the parts and the PCB and by that the project will not work properly. So to make sure that we do not get into this problem we will make sure to use the required power source that makes the components work in a good shape.

After finishing the PCB we need to test all the outputs and inputs that are on the PCB and see if they are working properly or if there are any mistakes happening during the placement of the parts.

8.3 Software Testing and Integration

Our pen plotter is controlled by a user interface that is loaded on the users desktop. With that said, software is a very important aspect for our design for many different reasons. The interface as well as the controlling firmware of our processor must be tested thoroughly in order to assure that it is functioning as intended.

This section will contain testing on hardware based software elements as well as purley software aspects. The hardware elements will be the microcontrollers firmware and the software elements will be the user interface, API connecting interface and firmware as well as the mobile application. The mobile application is a stretch goal but it will still be included in testing as it will also need to be thoroughly tested as it would be a key element of our design if completed.

Below is a diagram which depicts the different software elements and how they are related. We will use these relations to establish our testing protocols.

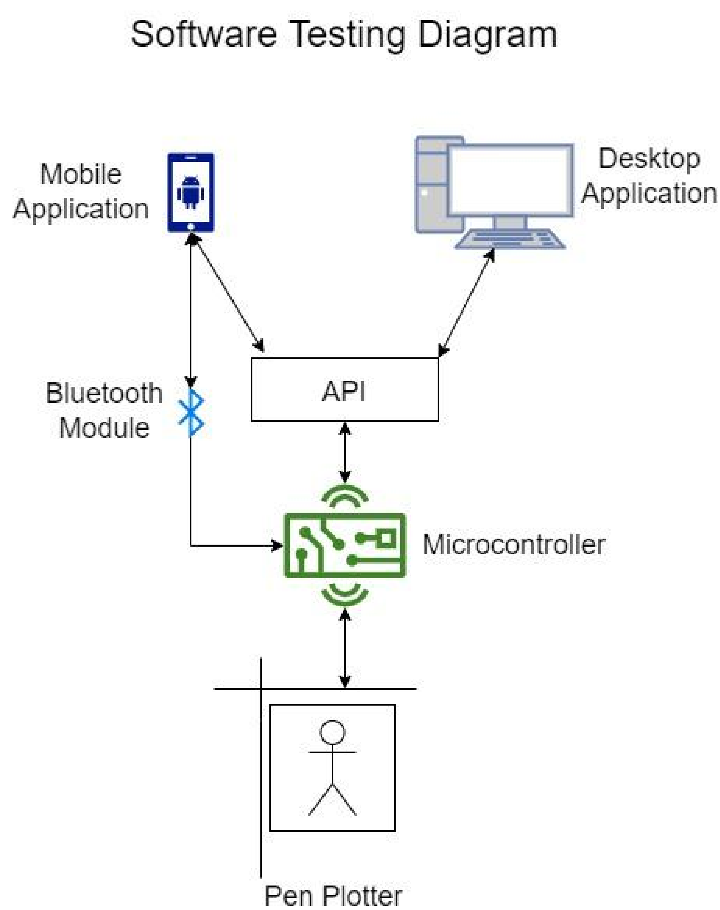


Figure 70: Software Testing Diagram

8.3.1 Firmware Testing

Firstly when new additions are made to our firmware code, we will run it through software such as sonarqube. Sonarqube is free to use software which you can input your code and it will return possible bugs, vulnerabilities, stylistic problems and other general imperfections within the code. This can be a good way to automate a lot of our testing and the use of software would allow for more precision in our testing of the firmware.

We will also be required to run unit testing of certain aspects of the firmware, whether manually or automated, this is something important that will test the functionality of the firmware in the implementation phase. There are many frameworks in existence for embedded systems in the C language which we can make use of for our design. Automated unit testing allows for faster deployment and the ability to make repeatable tests which allow for the easier detection of bugs or other errors in the code. Automated unit tests also allow for the easier detection of harder to detect errors such as memory leaks and deadlocks.

Check is one such framework we may utilize for unit testing of our firmware. It allows for the catching of memory leaks and fast and optimized testing. Check is implemented using the check.h header file in the c code as well as the library functions contained within it. We will assemble a separate unit test c file which will test our firmware code.

Some unit testing we can run are the implementation of the G-Code and its conversion to the C language which our processor can understand. G-Code values will be passed and if the correct output for the plotter is detected, the unit test will be considered successful. These tests will be run with many different variables, some will be designed to cause problems which our detection of edge cases in the code should be able to detect and manage if coded properly.

8.3.1.1 G-Code to C conversion

Tests will need to be done to determine that this conversion is functional. This can be done manually or with automated unit testing as described above. We will be inputting G-Code and we will check the output initially via console logs, to determine if the output is correct. We will then be able to fully test the G-Code to C conversion in the implementation phase when the design is assembled.

This conversion is very important for our design as a whole. The G-code is what the images are first translated into, this is what the CNC machinery can interpret as movements for the plotter. The microcontroller can only read the C language and that is why the testing on this step is important.

8.3.1.2 GUI to Hardware Communication

We need to determine the API which communicates the GUI's inputs to the firmware which controls the plotter. This will be done by determining that the text output from the gui is getting properly parsed, read and inputted by the C code firmware. Inputs will be made into the gui, which create the text output. This will be tested with multiple lengths of output, as well as with empty output to determine if there are any outliers which could make the code present any error.

Below is a table which will be the basis of this section of testing. It has not been tested as of yet, but this table will be updated and further expanded upon in the following weeks. This table is a representation of some of the key elements we will be testing, and the service used for testing.

Test	SonarQube	Unit Test (Check)	Manual Test
G-Code to C conversion	TBD	TBD	TBD
GUI to Hardware API	TBD	TBD	TBD

Table 15: Firmware Testing Results

8.4.2 GUI Testing

The GUI used by our design will need to be tested as well in order to determine it is ready for deployment and fully functional. As with the firmware, sonarqube can be utilized in order to check to see if different parameters are met. It will determine if there are bugs in the code or other vulnerabilities in addition to other common errors. Unit testing can be automated for GUIs as well, which we will make use of for our design.

The javaFX gui library can make use of the Junit testing framework. Junit testing is a common regression testing framework. Our gui is fairly simple so testing does not need to be done at great lengths if the backend code was successfully tested.

We will need to test that the buttons and input areas all function as intended. Gcode script can be entered as well as the user file directory can be browsed for images to be inputted. These will need to be tested to determine if they are functioning properly. Image preview will also be tested within these unit tests as it is another important aspect of the GUI.

8.4.2.1 Script Input

The input of scripts of G-code will need to be tested to determine if the input is being read by the gui and passed onto the firmware. We will input multiple instances of G-code scripts, if there is a correctly formatted text output made by the GUI, it can be determined that the script input gui element is working as intended. There can be parsing and code length errors and these are the types of things we will need to look for when testing this gui element.

8.4.2.2 Image Input Button

Image input is another gui operation which will need testing. This element opens the users file directory and loads their requested image and translates into G-code, this is then sent to the firmware through the api. The selected image is also shown in the preview window for the user to see and determine if it is what they want to be parsed.

This will be tested by using the input image button and determining if there is a correct output on the text file which is read by the firmware. There also needs to be a proper image representation on the output preview window of the gui. This will be tested with multiple images as well as non image files to determine if there are any errors with this part of the gui.

8.4.2.3 Output Preview Window

The output preview is a key element of our gui. It presents the user with an image that was, if outputted correctly, already parsed by the code and in the proper format to be sent and parsed by the firmware. To test this part of the code, since there are no specific gui elements which will change this initially, we will need to provide multiple inputs to determine that it is functional. Image inputs as well as script inputs can be applied to receive a change in the output preview window. Different things will be tested such as script length, image file type, image type, etc to determine if there are any errors.

Below is a table which will be the basis of the testing for different elements of our GUI design. It has not been tested as of yet, but this table will be updated and further expanded upon in the following weeks. This table is a representation of some of the key elements we will be testing, and the service used for testing.

Test	SonarQube	Unit Testing	Manual Test
Script Input	TBD	TBD	TBD
Image Input Button	TBD	TBD	TBD
Output Preview	TBD	TBD	TBD

Table 16: GUI Testing Results

8.4.3 Mobile Application Testing

A companion mobile application alongside of our controlling gui desktop software is a stretch goal we set. The application will only be made for android as it is the most convenient operating system to create a free app for mainly testing purposes and short term application use. We do not plan on developing an IOS application as the process for becoming an Apple developer and deploying the application for Android is deemed to be not reasonable for our design purposes. The testing aspects of the mobile application and its functionality will be detailed in the section for testing if we are able to complete this stretch goal. The mobile application has some of the features of the main gui and should be able to control some of the aspects of the plotter.

8.4.3.1 Connectivity Testing

To connect the microcontroller to the mobile application we will be using bluetooth. The bluetooth module is embedded within the microcontroller and it will allow for our device to connect and send signals back and forth from the mobile device and the pen plotter. These tests will be conducted manually as it is the most efficient method for this type of test as it is not very complex.

The bluetooth module will have a led light and on the mobile device there is a bluetooth logo. If both of these indicate that the connection is successful the first part of testing is successfully completed. It is important to test this multiple times, and with different devices. We will also test multiple distances of connectivity to determine how far a user can be from the microcontroller and still have connectivity as well as full functionality from the bluetooth module.

8.4.3.2 Interface Testing

With interface testing, we need to ensure that all of the gui elements work properly. Every gui element ID will need to be sending the correct information when an input is applied. These types of tests are known as screen UI tests. They are performed by a user manually clicking on crucial parts of the GUI. Actions such as clicking on buttons, such as a stop/start button. Making inputs in text boxes as well as controlling scrolling elements.

We will also test the interface flow. This is done by checking how elements interact with each other and it is used to see if all elements of the gui work seamlessly with each other. These types of tests are simple and can be done manually. If there are any connectivity issues or application crashes they will need to be reported and further looked into to fix such bugs or errors.

8.4.3.3 Functionality Testing

Functionality testing determines that all user actions have an effect on other systems. Other systems such as the api and firmware are actually controlling the actions of the pen plotter. These can be tested manually or with unit testing frameworks such as Robolectric. Robolectric is a popular Android development unit testing library that is used with the Gradle toolkit which manages android dependencies and allows for the deployment of logic. In this case we will be using this logic for unit testing.

Once this is completed, the unit testing code for the can be implemented using Java classes. Robolectric uses built in java methods which provide automated button clicks and other activities which the user would perform. These types of automated tests are good practice to easily change the inputs and to make sure the same types of tests are being performed. This allows for further edge case testing and bug finding within the mobile application.

The following code snippet shows some of the capabilities of Robolectric. The MyActivityTest class is the encompassing class which performs the tests. Inside of this class, there are various activities such as button clicks and message outputs. The syntax of activity.button.performClick(); is standard for this library.

As you can see, different elements are tested. This is similar to the types of testing we will be performing on our mobile applications interface. With the specific activities tested tailored to our actual mobile application based on our expected user actions.

9.0 Project Operation

This section covers necessary user information about the use, troubleshooting as well as safety information related to this pen plotter. With this pen plotter being a device that is controlled by user input, it is possible that the expected output could cause issues relating to safety or could cause other issues on the device. It is important that the steps listed below are followed in order to maintain a properly working device and to avoid any unnecessary safety risks when using, or being in the general proximity of the pen plotter in operation.

It is also important to understand that there are many exposed mechanical parts based on the nature of our design. That being said, it is of utmost importance that the user is aware of the following information in order to preserve the integrity and working order of the device. An incorrectly assembled device could also cause safety hazards or other environmental risks if the device is not properly cared for and used in a proper environment. The following sections will also detail how a user can troubleshoot this pen plotter given common errors.

9.1 Safety Precautions

Our design has been engineered with safety in design. While there are no major safety hazards while dealing with the pen plotter and its associated machinery, there are always risks involved when dealing with electronics with large moving parts. In the following section these risks will be highlighted and we will present ideas on how to mitigate these risks to allow for the continued safe use of our pen plotter.

The area the user chooses to place the pen plotter is important. Not only for the users safety but also to maintain the structural integrity of the machine. If the machine is damaged because of its poor placement environment, it could become a greater hazard than necessary. It is important that the pen plotter is used on an elevated surface with an area of placement larger than the total working area at least 2x2 feet of space is recommended.

A smaller working area could create a safety hazard if the plotter is somewhat off of a table and causes imbalance, causing the plotter to be displaced from its elevated position. An elevated location is also important for visibility purposes and so there is no tripping hazard associated with the machine as this could cause bodily injury for the user as well as destruction to the plotter.

There are several wires carrying electricity to and from the machine. It is important to be aware of such wires in order to prevent injury. The wires in question are the usb connecting the controlling desktop to the microcontroller as well as the power supply connection running from a wall outlet to the microcontroller. It is important to confirm that all connections are secure and in safe and functioning electrical input points in order to prevent any harmful electric shocks associated with insecure wiring.

It is important to ensure that no liquids are in the direct vicinity of the pen plotter or on the working surface if possible. The moving machinery could cause spillage if open containers are nearby. Liquid could cause damage to electronic parts and could present a safety hazard in the event of an electronic shortage.

Safety standards are something every design should take into account as it can make or break a design as well as the market implementation of that product. If safety standards are not met, although many are considered voluntary, they may not meet safety regulations established by a government because of that. A common regulation we must take into account is RoHS. RoHs is the restriction of hazardous substances directive. They regulate several substances deemed to be hazardous and some of those substances can be found in our design. Specifically in the pcb so we must make sure to take care if a disposal of materials containing such substances is necessary. These kinds of regulations are important for protecting the environment from substantial harm.

Other types of safety standards that may be seen as relevant are those dealing with moving or openly housed machinery. Such as that of many cnc machines. Our project being an open design allows for the possibility of bodily injury if one is not careful or if the design is not made as safe as possible by following standards put into place. Not only is bodily injury affected by this but the machine could also be damaged if the open design aspect is not addressed.

9.2 General Information

This section covers the basic functionality of the pen plotter. The bullet points listed are the capabilities we hope to provide with our design:

- A Desktop GUI used for user input.
- Output of images written strictly in G-Code.
- Image to G-Code conversion algorithms.
- Transfer of data from desktop gui to pen plotter microcontroller.
- Display of progress of output on desktop application.
- Ability to stop the plotting process at any time.
- Ability to display output on GUI before output is begun.

9.3 Using the Pen Plotter

The following bulleted list describes the startup and use procedures of the pen plotter. Use this as a guide for the initial startup and shut off procedures.

- Find a suitable elevated working surface to place a pen plotter.
- Surface must have suitable space and be considered a safe working environment.
- Plug power supply into microcontroller and connect to suitable wall outlet.
- Plug Mini-Usb cable into the computer and ensure it is connected to the microcontroller.
- Load GUI software on the desktop by running ‘PenPlotterGui.jar’.
- Ensure the device is powered on by indicator lights on the microcontroller.
- Use interface to input either G-Code or Input image.
- View Output preview.
- If output preview is satisfactory, select run on gui and the plotting process will commence.
- There will be an indication of completion percentage.
- Wait for the plotting process to end completely before moving anything on device in order to ensure proper completion of plot.
- If at any time the user needs to stop the plotting process, hit the stop button and the machine will be stopped and reset.
- Store Plotter in a safe location not in use to avoid damage to any mechanical or electronic parts as it is somewhat delicate due to its open design.

9.4 Troubleshooting Tips

Below are common issues a user may face when using our design. Not every possible bug will be listed as there are many errors that can occur. The following table will serve as a guide to solve common issues.

Problem	Solution
Device is not indicating that it has been powered on.	Ensure that wall outlets have power and the cable is correctly connected to the microcontroller.
No Connection to Microcontroller.	Check if the usb cable is properly connected from the usb port on the microcontroller to the desktop.
Gui will not open.	Confirm that the correct jre version is installed on the desktop.
Plotting process will not start.	Ensure device is connected and it is indicated as so on gui and microcontroller.
Input image is not processed.	Ensure that a valid file type was inputted. An error message can be seen if this is not the case.
Output image is not as the preview showed.	Test desired writing utensil for drawing ability manually then try again.
The Plotter's mechanical elements are not moving.	Ensure the pulley system is on correct rails and all parts are assembled correctly.

Table 17: Troubleshooting Tips

10.0 Administrative Content

In this section, we give a detailed analysis of the various milestones set for the project, the budget and the cost involved for purchasing of the parts and division of labor indicating the various parts of the projects and the team member responsible for it.

10.1 Initial Project Milestones

During the beginning of senior design I, we got help in the formulation of our group from Dr. Wei. After the group was formed we proceeded to brainstorm to look for ideas and possible projects ideas that the group can work on during the semester. The group met with Dr. Richie to discuss all the ideas the group had come up with and then settled on this project. The group then started with the documentation of which we divided the project into various parts and tackled them as time went on. As of now we are working on the initial documentation alongside other things like high level software design, high level hardware design and the design for the chassis and moving parts. Below is a tentative table for the initial project milestone which was agreed on by the group members.

Milestone	Planned Completion Date
Research programming language and development platform	2/20
Decide on software specs and development platform	2/28
Research moving parts	3/1
Research chassis	3/7
High level plan of hardware design	3/14
High level plan of software design	3/28
Design chassis and moving parts	4/6
Research protocols and electrical parts needed for the chosen moving parts	4/20
Order hardware, development platform, and electrical parts	4/28
Prototype code	5/20
Get parts moving with prototype code	5/30
Drawing code	6/5
Tool Switching code	6/15
Refine hardware	6/18
Refine software	6/18
Finish code	6/21
Order PCB	6/22
Implement PCB	7/22
Finish development	7/30

Table 18: Initial Deadlines

10.2 Budget

During the early days of brainstorming for the project, the team decided that each team member will allocate an amount of not less than $150 for the completion of the project. This money was to serve as funds for the procurement of the various parts and tools needed for the project. Also this was to cover any external services that may be needed to get the word done like printing of parts or the making of the PCB. We divided how the money was going to be spent into two different sections, thus the purchase of the various hardware parts and also the purchase of the more heavily electronic parts. This is an estimated list of the parts that the project will need, this list might change in the future and there will be some added parts and there might be some deleted parts. These are most of parts that will be needed in order to build the machine:

Part name	Seller	Quantity	Estimated unit price
MGN15H Linear Rail	Amazon / AliExpress	2	$43.99
Linear Rod 6mm	Amazon / AliExpress	2	$9.99
Linear Bearing 6mm	Amazon / AliExpress	1	$9.49
Stepper Motor – NEMA 17	Amazon / AliExpress	2	$12.99
GT2 Belt + Tooth Pulley	Amazon / AliExpress	2	$15.99
GT2 Idler Pulley	Amazon / AliExpress	2	$8.59
Spacer Nuts	Amazon / AliExpress	1	$9.99

Bolts and Nuts set	Amazon / AliExpress	1	$26.99
3D printed mounting part			$100.00

Table 19: Hardware Budget

Table 4 also shows the estimated cost for acquiring the electronic components needed to complete the project. This budget is just a rough estimate as prices of these components tend to increase in relation to the market demands. This list will be updated as to when it is necessary to reflect the actual money utilized to get this project done.

Part name	Seller	Estimated price
Stepper Motor – NEMA 17	Amazon / AliExpress	$12.99
Stepper Motor – NEMA 17 23mm	Amazon / AliExpress	$9.99
A4988 Stepper Driver	Amazon / AliExpress	$10.99
Servo Motor	Amazon / AliExpress	$10.69
Arduino CNC Shield	Amazon / AliExpress	$7.39
Arduino Uno	Amazon / AliExpress	$23
Limit Switch	Amazon / AliExpress	$6.99
DC Power Supply	Amazon / AliExpress	$12.99

Table 20: Electronic Component Budget

10.3 Division of labor

This team is made up of team members from different engineering disciplines. To make sure that we have a successful project, we decided to allocate various parts of the projects to team members according to their strengths and engineering fields taking into consideration their experiences too.

The team unanimously settled on Anthony DeMore who is a computer engineering major, to be the team lead because he is very outspoken and also good with time management. He is also responsible for the LCD that will display the completion percentage of the work being done by the device, the kill switch that will stop the device when pressed and also make sure the output produced by the device matches the input design. He has a strong background in software and this will come in handy for the writing of the software for the communication of the LCD with the devices microcontroller. Al Moatasem Al Abri, who is an Electrical engineering major is responsible for mainly the hardware of the device. Also he led the design of the various circuits for the PCBs that will be used for the device. He is also responsible for the design of the power supply that will provide the device with the necessary power to enable it to run as desired.

Peregrino Quansah, also a computer engineering major is responsible for the software that will translate the instructions into G code for the microcontroller to use and also the servo motor that will move the various parts of the device. Patrick Caughey, a computer engineering major is responsible for the stepper motor, the microcontroller firmware and the metal rails. The figure below shows the division of labor among group members.

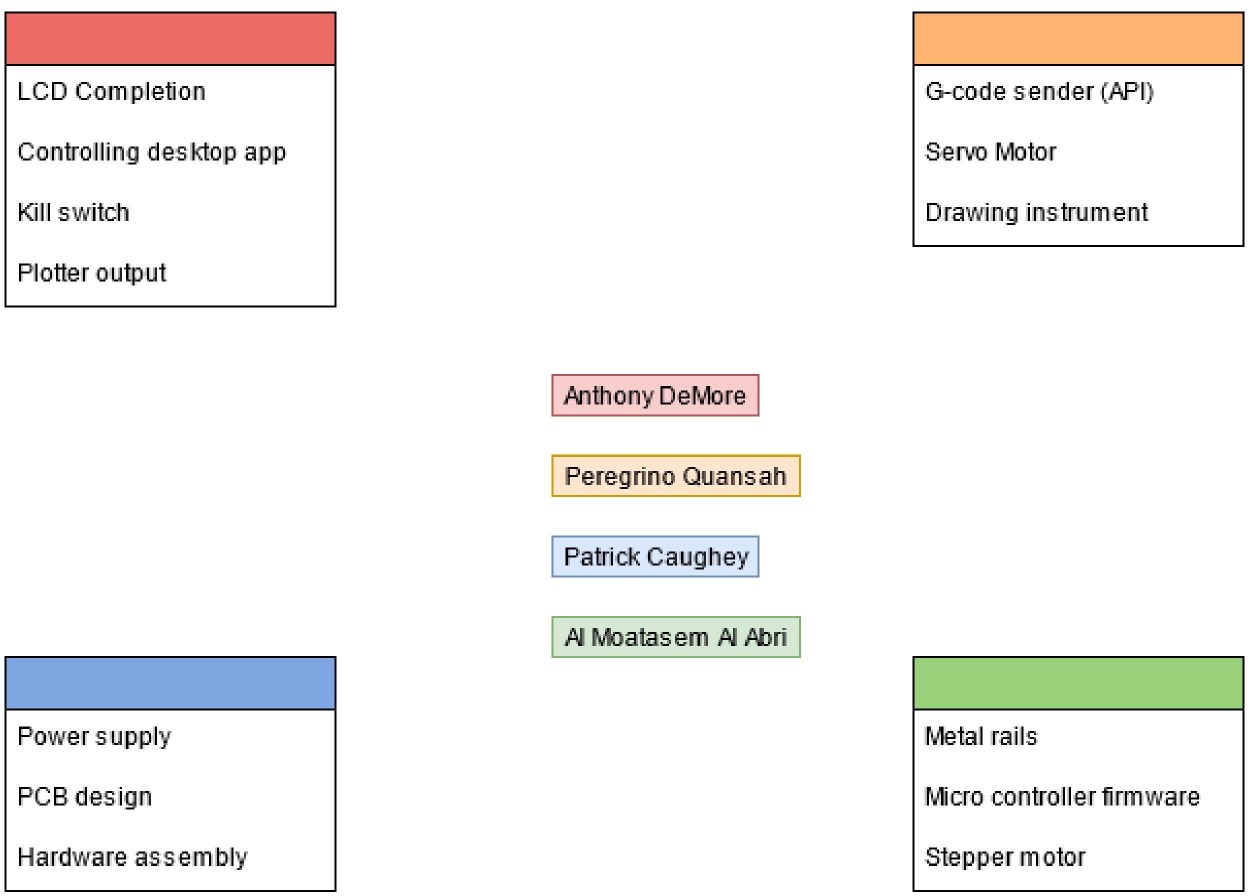


Figure 71: Division of Labor

11.0 Conclusion

Our goal for this project is to create a user-friendly pen plotter that is controlled by computer software which will be easily accessible by anyone. The software will allow for the user to input their design and they will be able to watch their artwork be created in front of their eyes. We will use parts like mentioned in the budget to create the robotic arm needed to draw on paper. The design of this pen plotter is similar to many dual axis robotic machines making for many readily available parts such as motors and metal rails making for lower costs.

Our pen plotter will feature a two axis design which will allow it to bring the pen to any point on a piece of paper. It will also use a third axis to move the pen up and down, allowing it to start and stop strokes anywhere. The user will create a pattern for the pen plotter to follow in a piece of software developed for desktop computers. This software will allow the user to create patterns for the pen plotter from numerical positions, curves, or an input image. The API will convert this pattern to a file that is readable by the microcontroller's firmware. This firmware will then control the motors of the pen plotter in such a way to move the pen to make strokes on the paper matching the pattern created by the user in the desktop software.

As a first stretch goal, if everything goes well, the pen plotter may also be made to make drawings with multiple colors. It would switch pens without any input from the user. This would allow it to create even more impressive pieces of art. This would also require changes to the software, API, and firmware to support multiple colors.

As a second stretch goal, the user may be able to create patterns and control the pen plotter from a mobile app. This would simplify the interface between the user and the pen plotter, while allowing for the same control over the device's final illustration.

This project presented us with many challenges along the way. It was our first experience with assembling multiple pcbs as well as assembling a complicated mechanical engineering machine which would become our plotter. Despite this, our group managed to create a very interesting design that we hope will be a great example of an artistic instrument.

We chose a somewhat challenging project in order to create a design that would be a good point of interest in our future careers as well as to expand our knowledge on many topics of hardware and software. This type of long term project allowed us to learn many things which are used within the industries that we will be entering post graduation.

This project so far has been a great learning experience. Not only in the design aspects of software and hardware, but also in terms of working as a team, establishing goals, creating a managing a budget and other important aspects which are not taught in typical class curriculums. These experiences are something we can all take into account when managing our future careers and there are many valuable lessons we are learning throughout this process.

Even though our pen plotter design will likely not reach an extremely large audience as it is not something that is considered a necessity by most, we hope that it will intrigue at least some people since it is such an interesting concept that we hope others will appreciate as well. This type of product is something that rewards the creative thinker and anyone willing to experiment. We look forward to the building and application phase of our design in the next semester as there are so many different pieces of art we can create. We hope that we are able to achieve as many goals as possible in the following senior design semester as it will allow us to create the best final product.

12.0 References

This section contains the resources used as references throughout the paper. They are separated by sections based on their source information.

Information on pen plotters:

- “Pen Plotters: The Best XY Plotters of 2021.” *All3DP*, 3 Apr. 2022, https://all3dp.com/2/pen-plotters-best-xy-plotters/.
- Geere, Duncan. “Pen Plotters Are the Perfect Tool for Data Storytelling.” *Medium*, Nightingale, 23 May 2021, https://medium.com/nightingale/pen-plotters-are-the-perfect-tool-for-data-storytelling-b05c71ceadd5.
- Toft, Tobias. “An Intro to Pen Plotters.” *Medium*, Quarterstudio, 18 Sept. 2019, https://medium.com/quarterstudio/an-intro-to-pen-plotters-29b6bd4327ba.

Title image:

- Laboratories, Evil Mad Scientist. “AxiDraw Writing and Drawing Machines.” *AxiDraw Writing and Drawing Machines*, https://www.axidraw.com/.

Other products:

- Kuklovskai͡a Elizaveta. “DP.” *Amazon*, "Books by Mail" Pub. Co., https://www.amazon.com/dp/B06ZXRV6GQ?tag=all3dp0c-20.
- “What Is 3D Printing?” *3D Printing*, 17 Sept. 2021, https://3dprinting.com/what-is-3d-printing/.
- “Behind the Scenes: How Ultimaker 3D Printers Are Manufactured.” *Ultimaker.com*, https://ultimaker.com/learn/behind-the-scenes-how-ultimaker-3d-printers-are-manufactured.
- Winder, Will. “Screenshots.” *UGS*, https://winder.github.io/ugs_website/.
- Cass, Stephen. “The Axidraw MiniKit Is the Modern XY Plotter You Didn't Know You Wanted.” *IEEE Spectrum*, IEEE Spectrum, 28 July 2021, https://spectrum.ieee.org/the-axidraw-minikit-is-the-modern-xy-plotter-you-didnt-know-you-wanted.

Standards and Specifications:

- “IEEE SA - ISO/IEC/IEEE International Standard - Systems and Software Engineering -- Life Cycle Processes --Requirements Engineering.” *SA Main Site*, https://standards.ieee.org/ieee/29148/5289/#:~:text=ISO%2FIEC%2FIEEE%2029148%3A,services%20throughout%20the%20life%20cycle.
- “ISO/IEC/IEEE 29119-1:2013.” *ISO*, 4 Feb. 2022, https://www.iso.org/standard/45142.html.
- “Applying IPC-2221 Standards in Circuit Board Design.” *Sierra Circuits*, 14 Apr. 2022, https://www.protoexpress.com/blog/ipc-2221-circuit-board-design/.
- “IPC Standards.” *IPC International, Inc.*, 5 Oct. 2021, https://www.ipc.org/ipc-standards.
- “How Are Standards Made?” *SA Main Site*, 18 Jan. 2022, https://standards.ieee.org/develop/develop-standards/process/.

- Tim LudwigTim is a former Copywriter for Heretto. “What Is the Best Standard for Technical Documentation?” *Heretto*, 15 June 2021, https://heretto.com/best-standard-for-documentation/.
- “ISO/IEC/IEEE 29119: Creating a Standard Approach to Test Software.” *ReQtest*, 14 June 2019, https://reqtest.com/testing-blog/isoiecieee-29119-creating-a-standard-approach-to-test-software/.
- Round vs Square Rail. LEE Linear Round Rail vs Square Rail. (n.d.). Retrieved April 25, 2022, from https://www.leelinear.com/Blog/2019/March/Round-vs-Square-Rail
- Claire. (2022, March 3). Round Guide rails vs square guide rails for CNC Routers. STYLECNC. Retrieved April 25, 2022, from https://www.stylecnc.com/user-manual/round-square-guide-rails-cnc-routers.html

Programming languages and IDEs

- “Top 15 Best Embedded Systems Programming Languages.” *UbuntuPIT*, 29 Apr. 2021, https://www.ubuntupit.com/best-embedded-systems-programming-languages/.
- “Platformio vs Arduino Eclipse Plugin.” *Compare Differences & Reviews?*, https://www.saashub.com/compare-platformio-vs-arduino-eclipse-plugin.

Software

- “SonarQube Documentation.” *SonarQube Documentation | SonarQube Docs*, https://docs.sonarqube.org/latest/.
- “3 Tutorial: Basic Unit Testing.” *Check 0.15.2: 3 Tutorial: Basic Unit Testing*, https://libcheck.github.io/check/doc/check_html/check_3.html.
- “About.” *JUnit*, https://junit.org/junit4/.
- written by John, John. “Best CNC Software [2022] for Hobbyists and Pros [Free and Paid].” *MellowPine*, 20 Feb. 2022, https://mellowpine.com/cnc/best-cnc-software/#firmware.
- “Fundamentals of Testing Android Apps : Android Developers.” *Android Developers*, https://developer.android.com/training/testing/fundamentals.

Hardware

- Engineering, Omega. “Stepper Motors.” *Https://Www.omega.com/En-Us/*, Omega Engineering Inc, 4 Nov. 2021, https://www.omega.com/en-us/resources/stepper-motors.
- “Stepper Motor Basics.” *Oriental Motor U.S.A. Corp.*, https://www.orientalmotor.com/stepper-motors/technology/stepper-motor-basics.html.
- “Motor Control Applications - Stepper Motor.” *Motor Control Applications - Stepper Motor | Mouser Electronics*, https://www.mouser.com/applications/motor-control-stepper/.
- “Stepper Motor Drives Information.” *Stepper Motor Drives Selection Guide: Types, Features, Applications | Engineering360*, https://www.globalspec.com/learnmore/motion_controls/controls_drives/stepper_motor_drives.
- “Voltage Regulator.” *Encyclopædia Britannica*, Encyclopædia Britannica, Inc., https://www.britannica.com/technology/voltage-regulator.

13.0 Appendix

13.1 List of Figures

13.2 List of Tables

10. Arduino Motor Control Pin Layout
11. Stepper Driver to Stepper Motor Pins
12. Arduino IDE Vs PlatformIO
13. Difference between AWT and Swing
14. Appropriate Voltage and Current for Each Part
15. Firmware Testing Results
16. GUI Testing Results
17. Troubleshooting Tips
18. Initial Deadlines
19. Hardware Budget
20. Electronic Component Budget