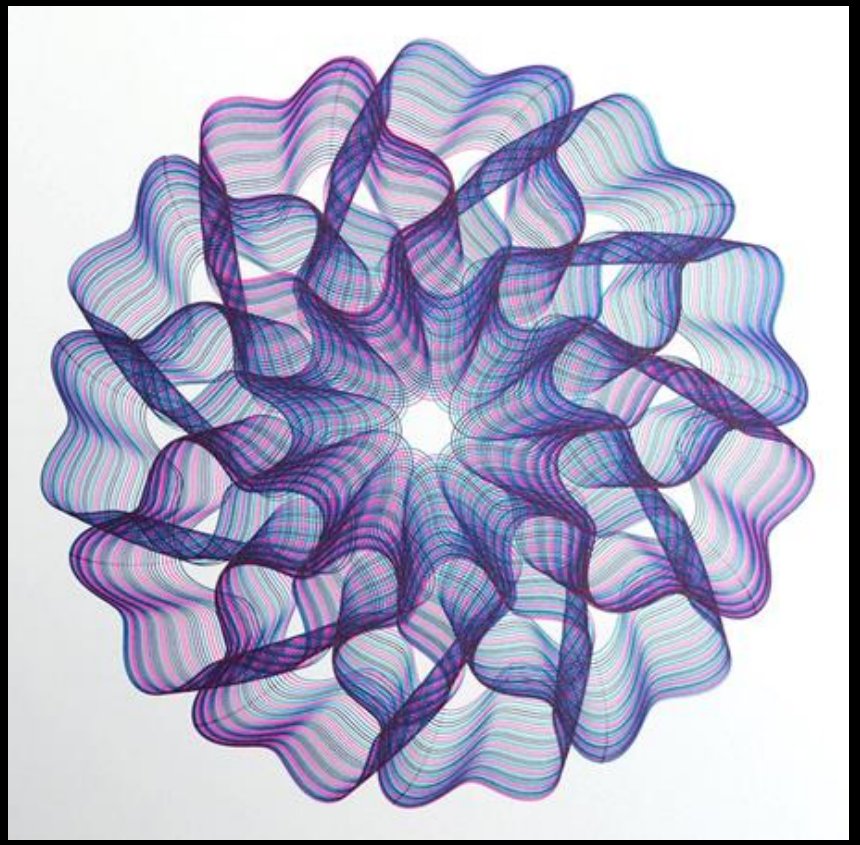
Automated Pen Plotter

- Group 24

- Al Moatasem Al Abri - EE
- Patrick Caughey - CPE
- Anthony DeMore - CPE
- Peregrino Quansah - CPE

INTRODUCTION

- Unique way to generate images with speed and precision.
  - Unlike laser printers are able to print continuous lines.
- Mainly used by architects and engineers.
  - Used for blueprints, cad drawings and architectural renderings.
- Used by hobbyists for unique art piece and designs.
  - Typically smaller scale than professional designs.
  - Allow for the use of writing utensil of choice.



MOTIVATION

- Pen plotters can be very costly.
  - Typically priced at $300-$1000.
- Our design aims to be an easy to use and cheaper product that will create more interest in pen plotting.
- Pen plotting can create unique works of art we want to allow more hobbyists to explore.

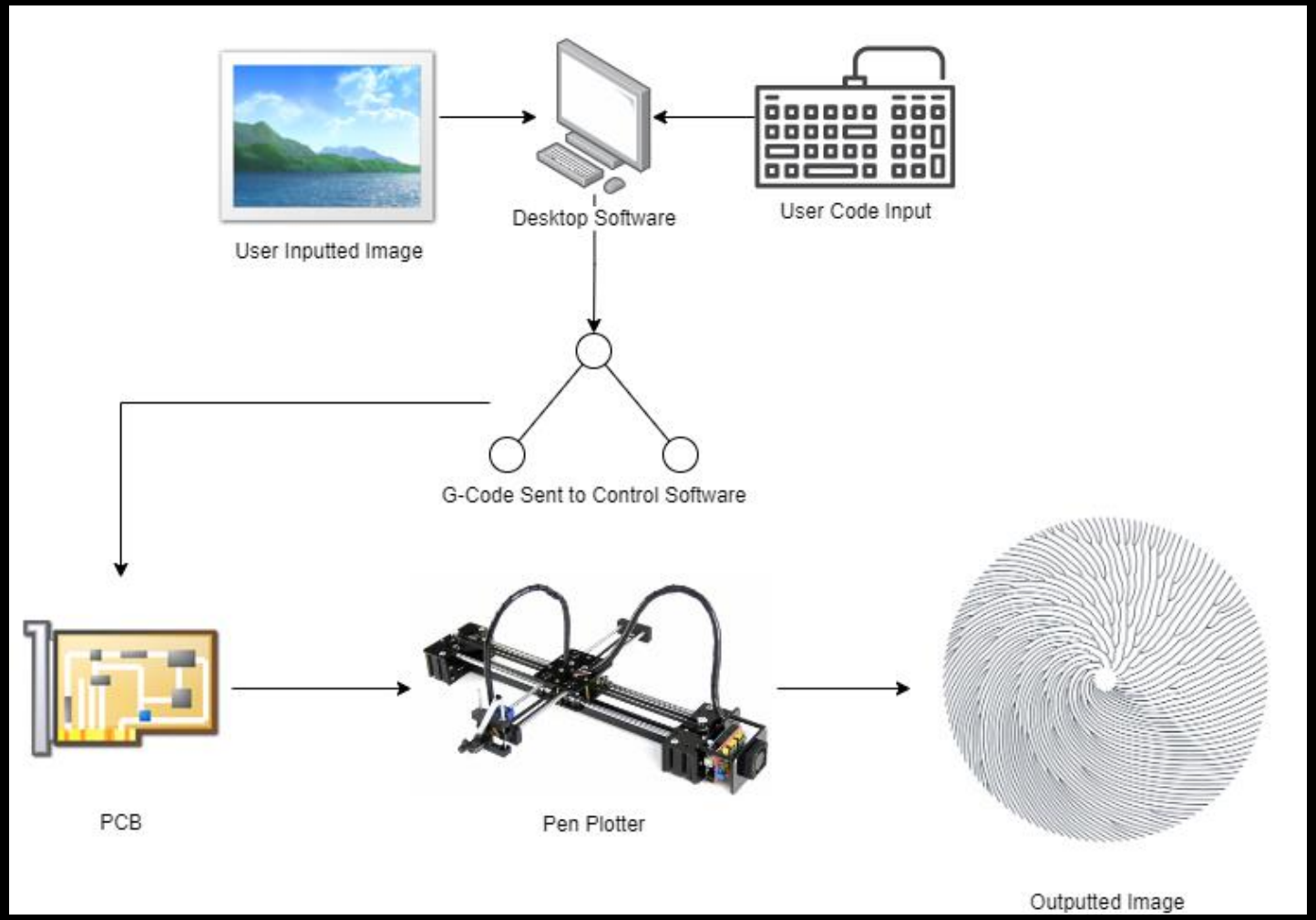
GOALS & OBJECTIVES

- Create a precise and mechanically sound design.
  - Plotter must be durable.
- Stability is important with this type of CNC adaptation.
- Allow the user to control the plotter from their PC.
- Ability for user to use the writing utensil and surface of their choice.
- Design an easy-to-use GUI which will control the entire process.

SPECIFICATIONS AND REQUIREMENTS

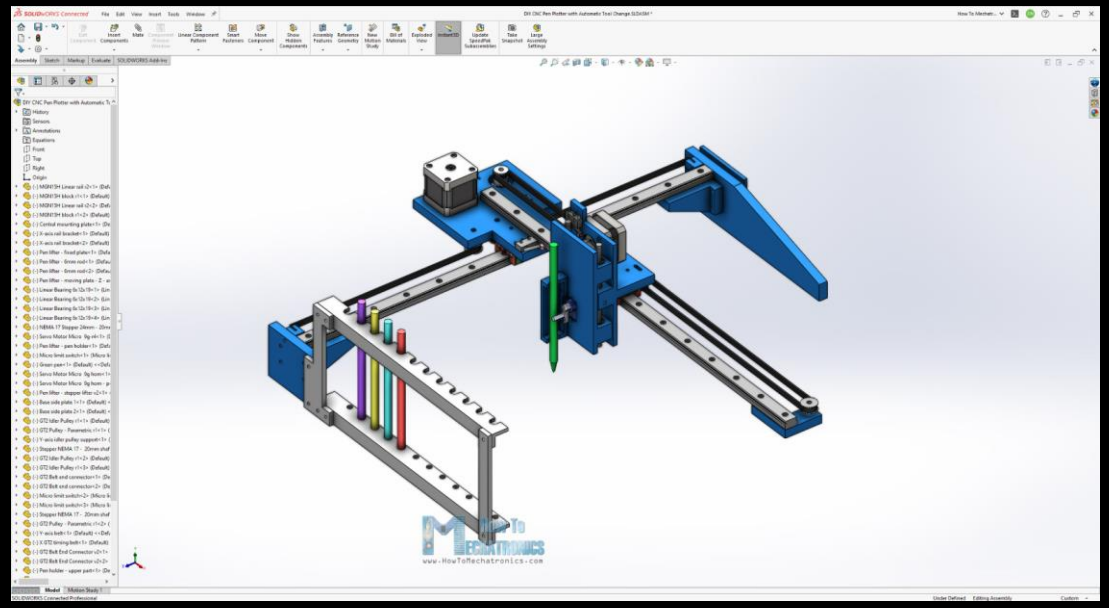
Specification	Requirement	Value
Stepper motors	X Y and Z axes	3 motors
Working area	the larger dimensions of A4 and U.S. Letter Sizes with additional margin of 0.5 Inches	9.5” x 12.75”, 242 mm x 324 mm or greater
Base size	working area plus 100 mm on top, 30 mm on each side, and 30 mm on bottom	302 mm x 454 mm or greater
Plotting accuracy	Output accuracy is key for useability.	90% accuracy in comparison to GUI representation
Plotting speed	Reasonable speed for writing or drawing	~10 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

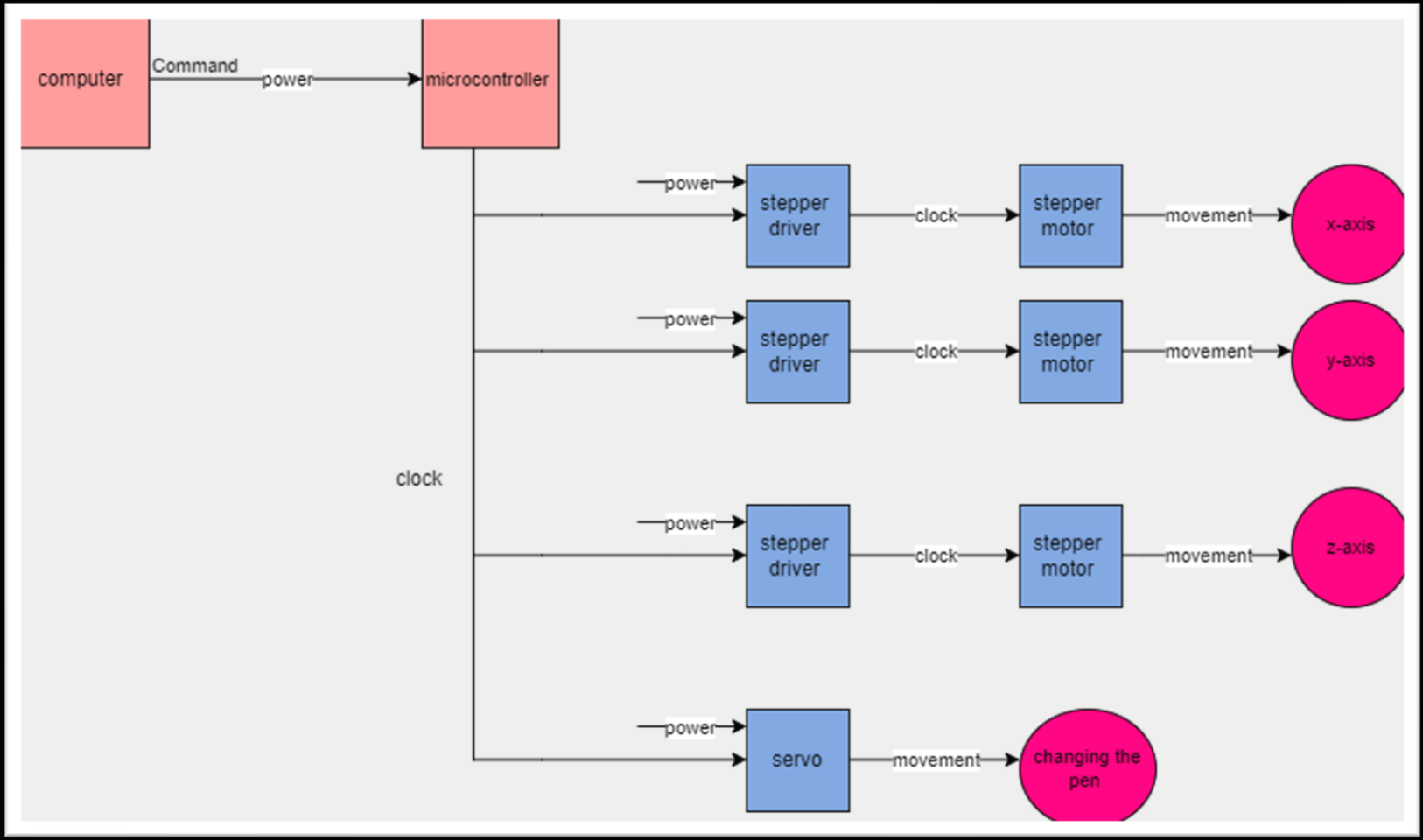
OVERALL PROJECT DIAGRAM



Mechanical Design

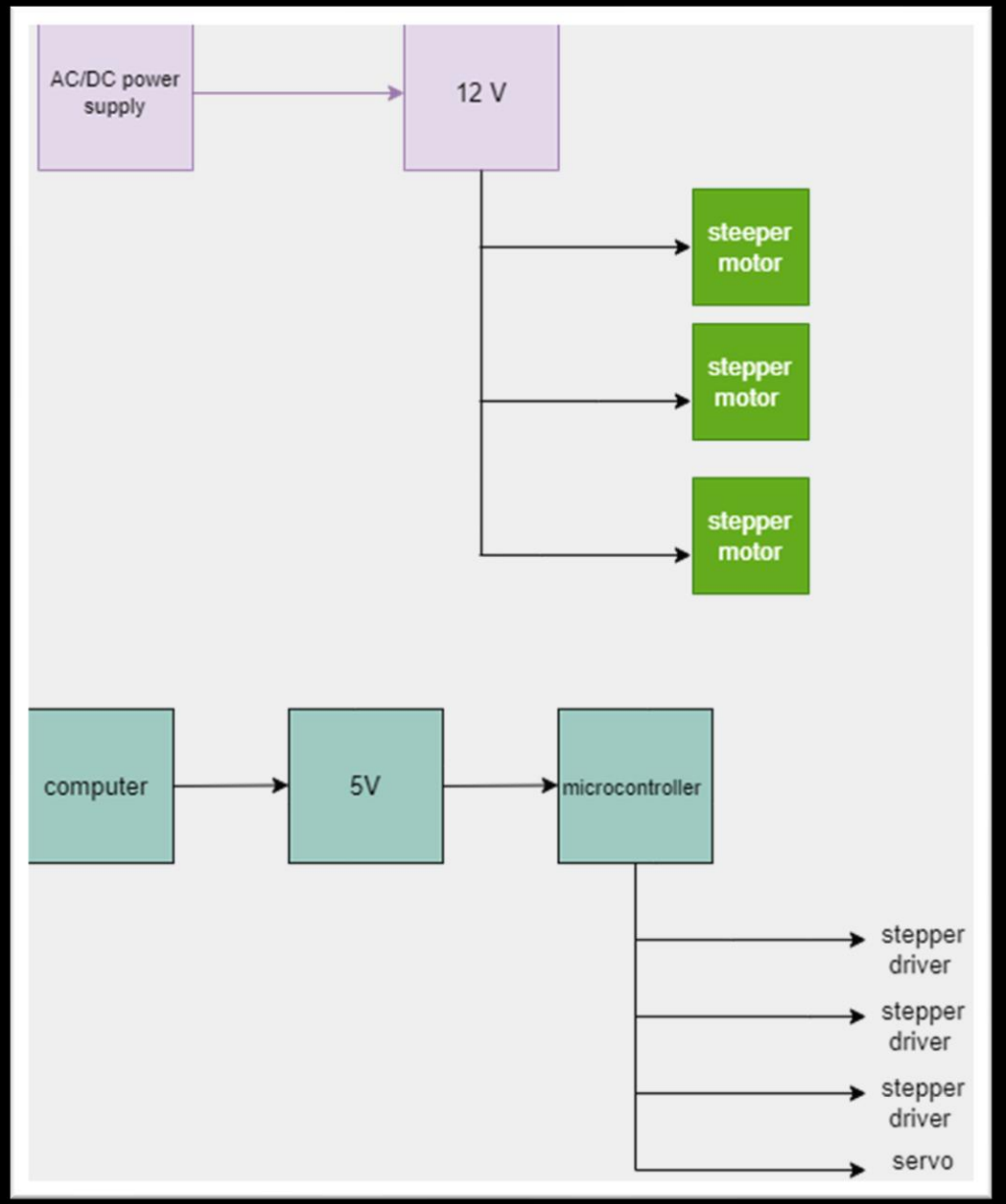
- We used an existing design
- It uses belts and stepper motors to drive the X and Y axes, and uses a sliding mechanism to move the pen in the Z axis
- Some difficulty with 3D printing
- Some changes were made to the design



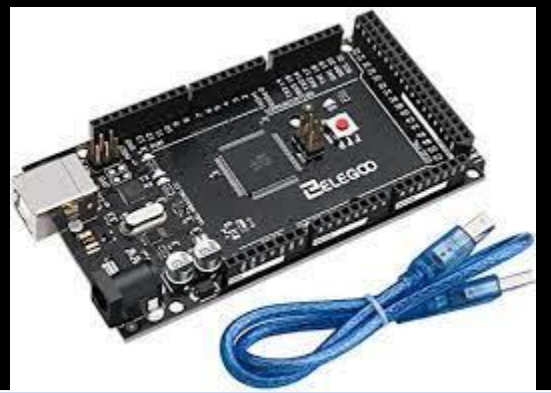


System block diagram:

Power distribution diagram:



Microcontroller:



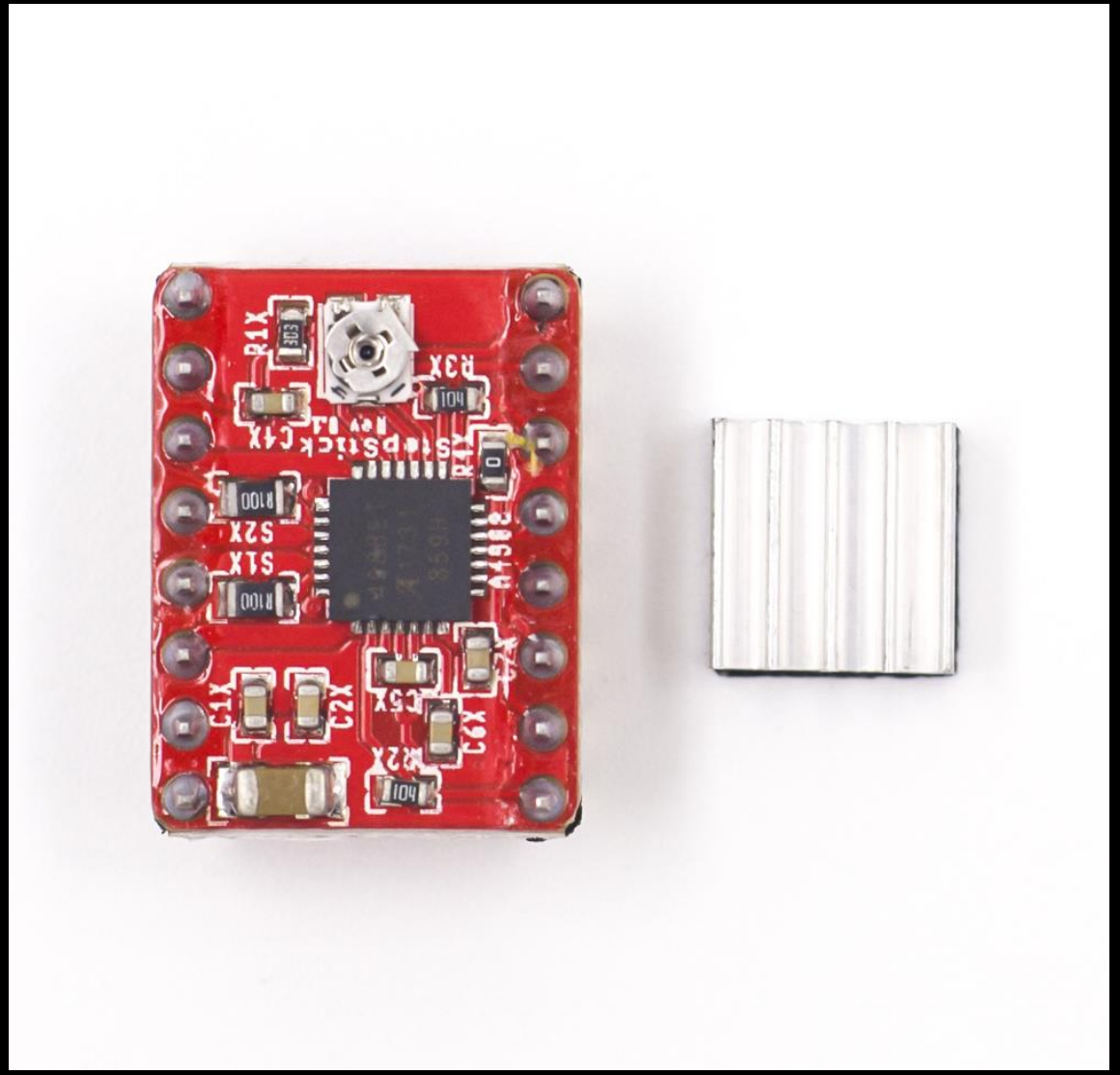
Specifications	Arduino Nano	Elegoo Mega R3
Microcontroller	ATmega328	ATmega2560
Voltage	5V	5V
Flash memory	32KB	256KB
SRAM	2KB	8KB
PCB size	18*45mm	101.52*53.3mm
Price	$19.80	$20

Stepper Motors:

- Used for the movement of the carriage.
- There are several types of stepper motors.(hybrid stepper motor, variable reluctance stepper motor).
- Name 17 stepper:
- Commonly used in 3D printing.
- Higher torque at low speed.
- 1.8-degree step size .
- Torque 45Ncm.

Stepper derivers:

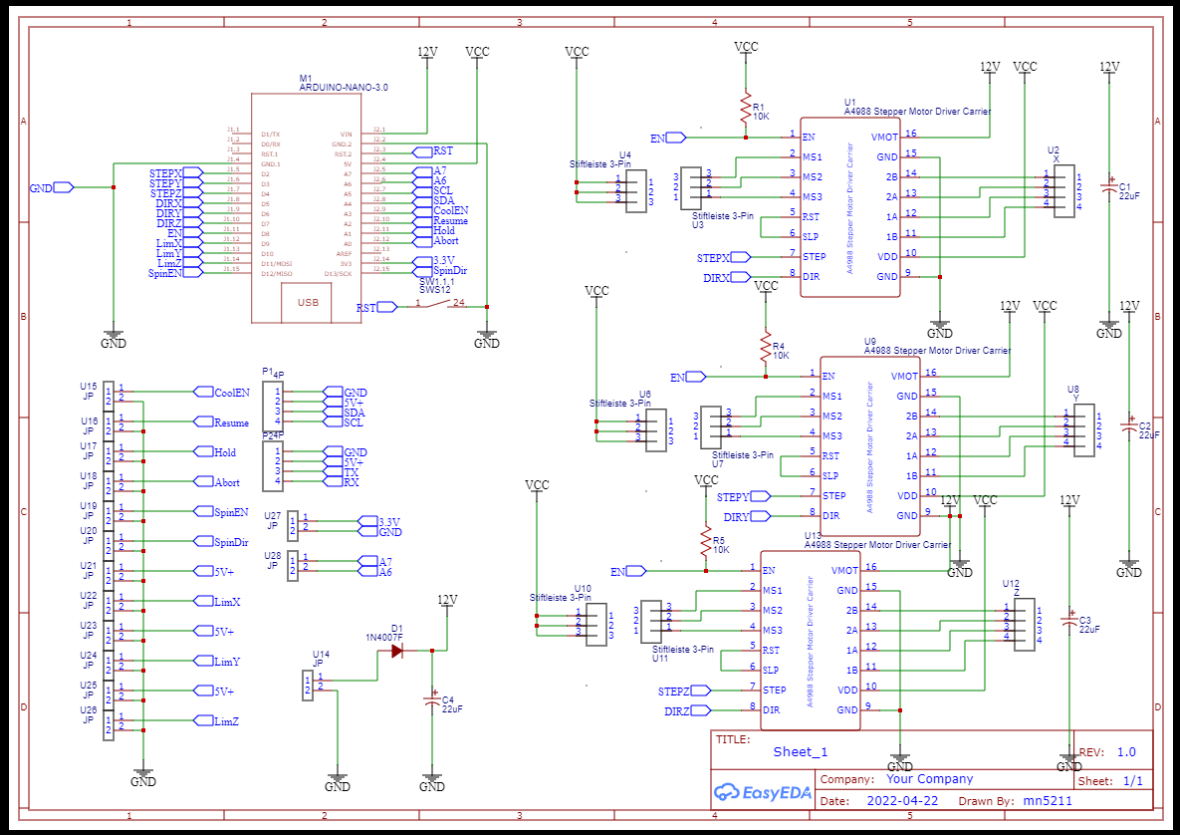
- Provides current required.
- Changing polarity.
- Voltage supply.
- Changing the step size
- Logic supply of 3.3-5V
- It can handle 36V and minimum voltage of 8V.
- stepper drivers. (A4988, STSPIN820…)



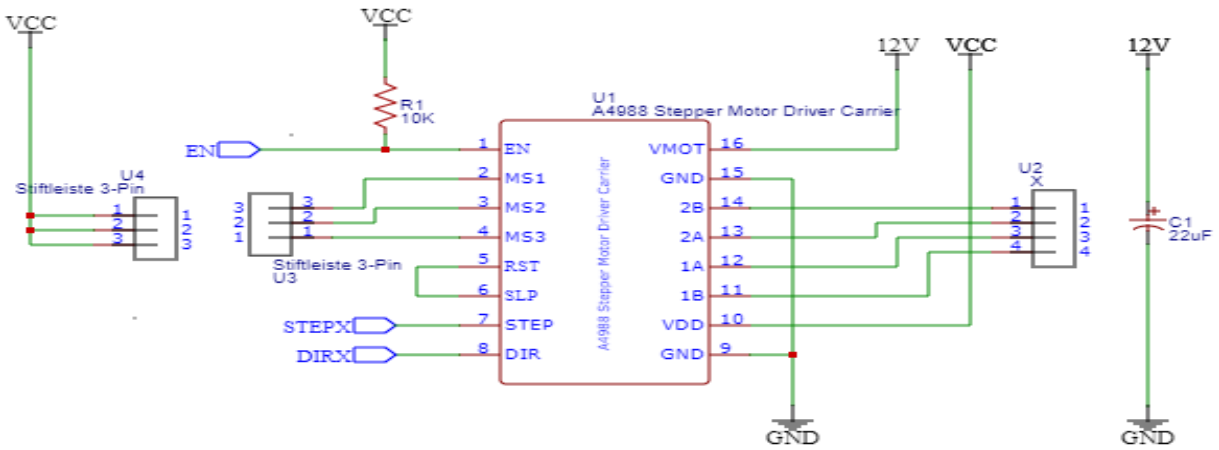
Limit switches:

- Commonly used in moving projects.
- Controls the movement and dimensions.
- prevent any issues that might cause by the moving parts.
- Used in (x-axis, y-axis, z-axis).
- Switches will be connected to the microcontroller.

PCB schematic:



Stepper Driver Schematic:



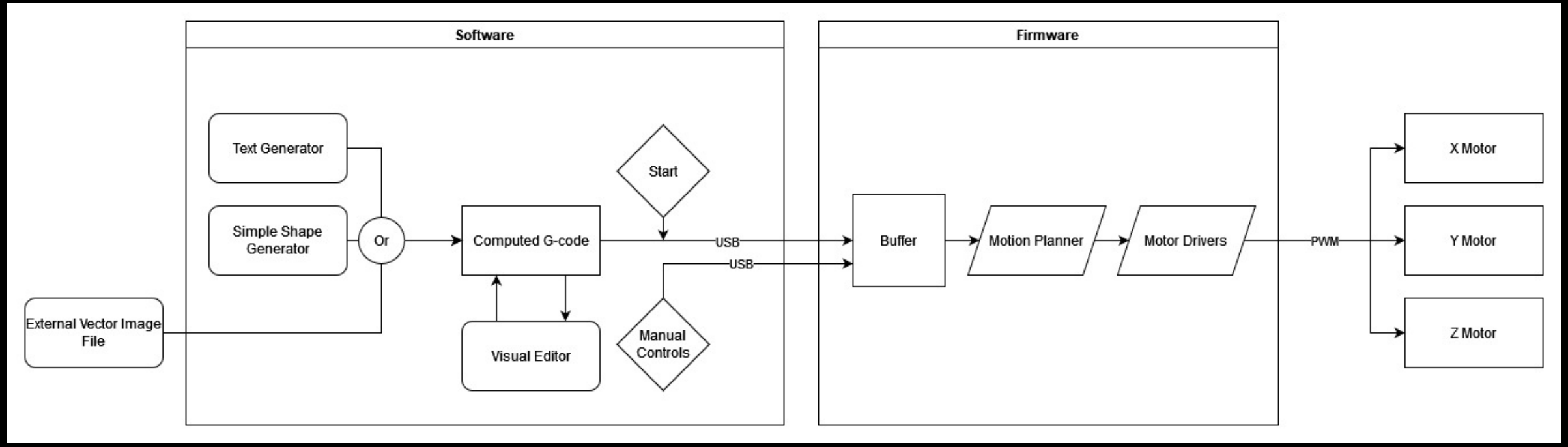
TITLE: Sheet_1		REV: 1.0
EasyEDA	Company: Your Company	Sheet: 1/1
	Date: 2022-04-22 Drawn By: mn5211	

Software

The project makes use of two different software components to operate:

- The firmware on the microcontroller:
  - Stores incoming G-code in a buffer
  - Plans motion paths and motor speeds based on buffered G-code
  - Controls the motors based on the planned motor speeds using pulse-width modulation (PWM)
- The control software on the computer:
  - Converts vector graphics into G-code
  - Allows the user to move, rotate, and resize the components of the G-code
  - Allows the user to manually control the pen plotter
  - Sends the calculated G-code to the microcontroller over USB

Software Block Diagram



Software Selection

Grbl-plotter is an open source CNC control software

- This software is developed specifically for plotting devices
- It makes room to incorporate a graphics converter
- The graphic converter will be responsible for creating G-code out of inputted graphics

Software Selection

The firmware that would be suitable for this project should be able to control multiple servo motors, be very user-friendly, run on almost all OS version.

CNC options for firmware includes:

- Marlin
- Smoothieware
- LinuxCNC
- Grbl-servo

Firmware comparison

Grbl-servo	Marlin	Smoothieware	LinuxCNC
Runs on all major OS platforms	Runs on all major OS platforms	Runs on all major OS platforms	Runs on only linux OS
Very user-friendly	Relatively less user-friendly	Requires extensive knowledge of CNC to be able to use	Requires extensive knowledge of linux to be able to use
Servo support	NO servo support	NO servo support	Servo support

GRBL-Servo

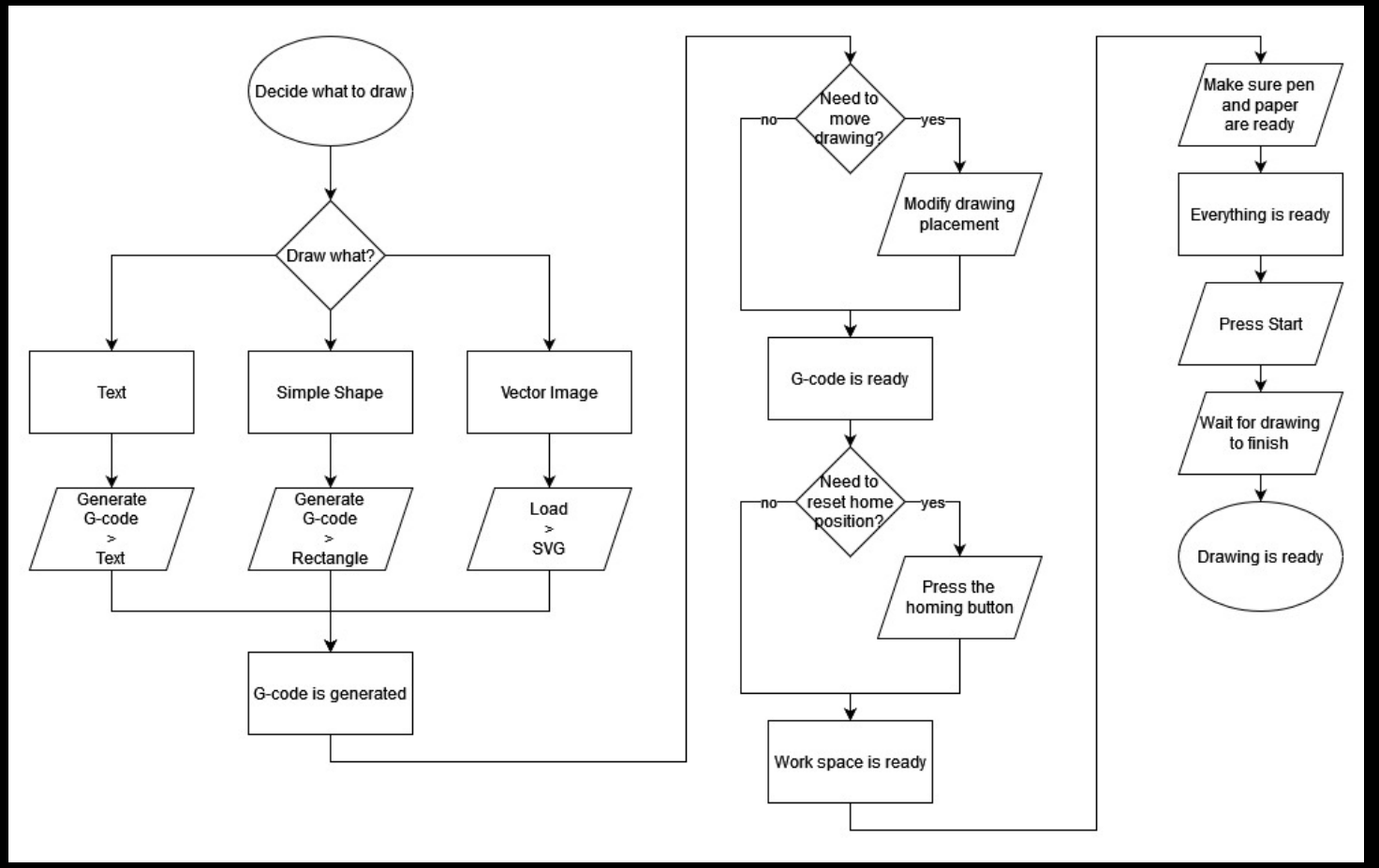
The grbl-servo firmware can be seen to have three main modules:

- The state machine: The Real Time control and execution of the state machine
- The interpreter: The interpreter and streaming of the G-code
- The peripherals: Report, Settings, Tool, Limits, Steppers etc.

Use case diagram

From the UI of the control software, users can also:

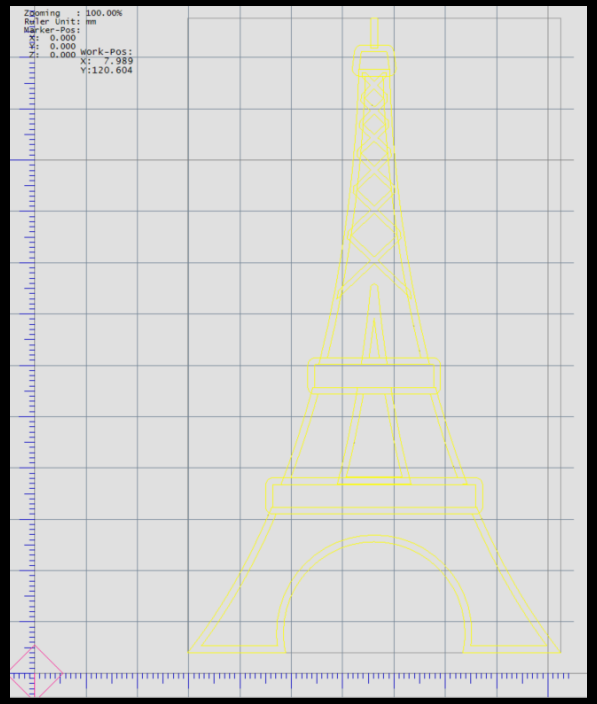
- Manually control the device
- Input and edit G-code
- Control machine variables

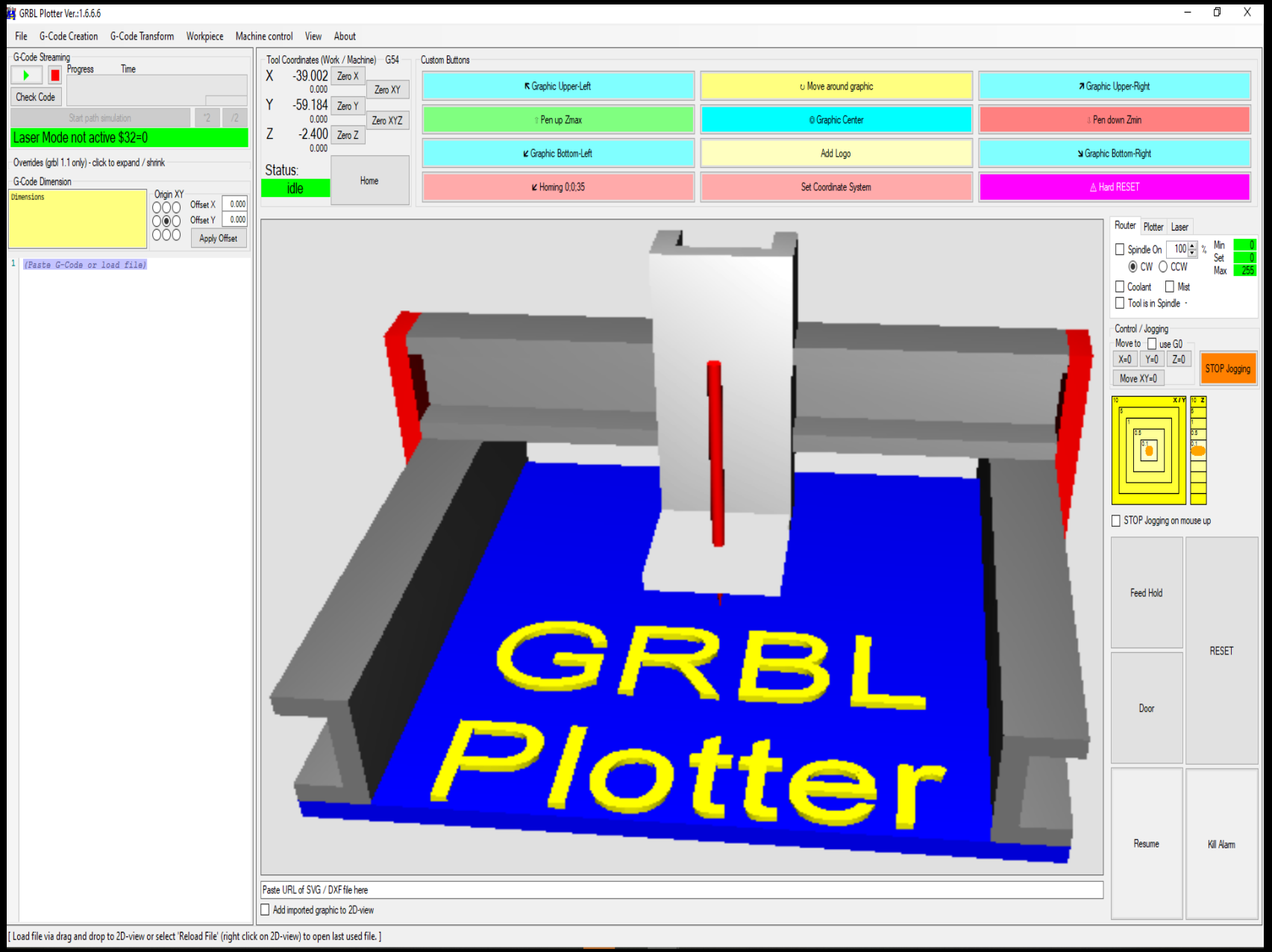


GUI

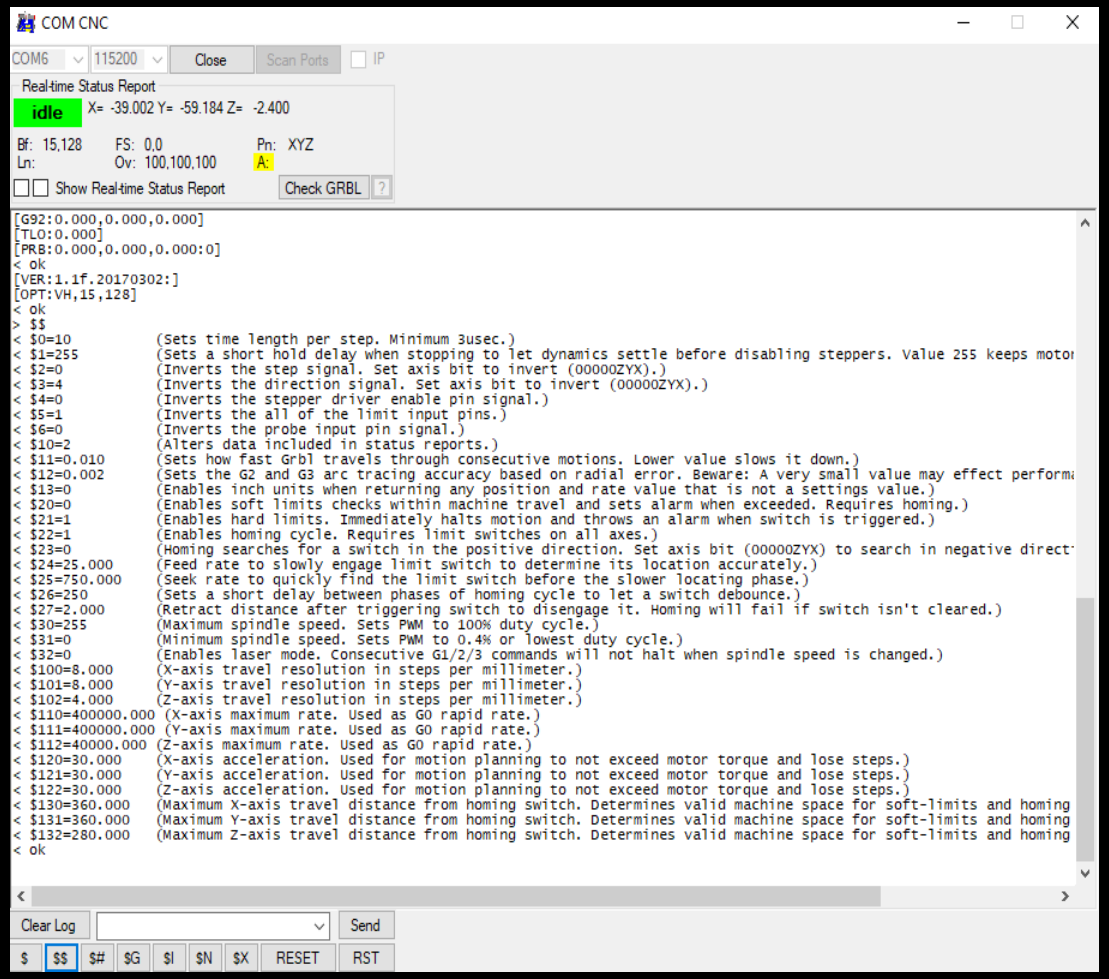
- Based user interface on open source project: GRBL-Plotter
  - Worked well with our firmware
  - Many built in features
  - Built in image conversion
  - Robust plotting image editor
- Many similar G-Code senders
  - Fairly complex programs
  - Universal G-Code Sender (UGS)
  - Not quite as optimal for our purposes
    - Similar features throughout
    - Some features missing

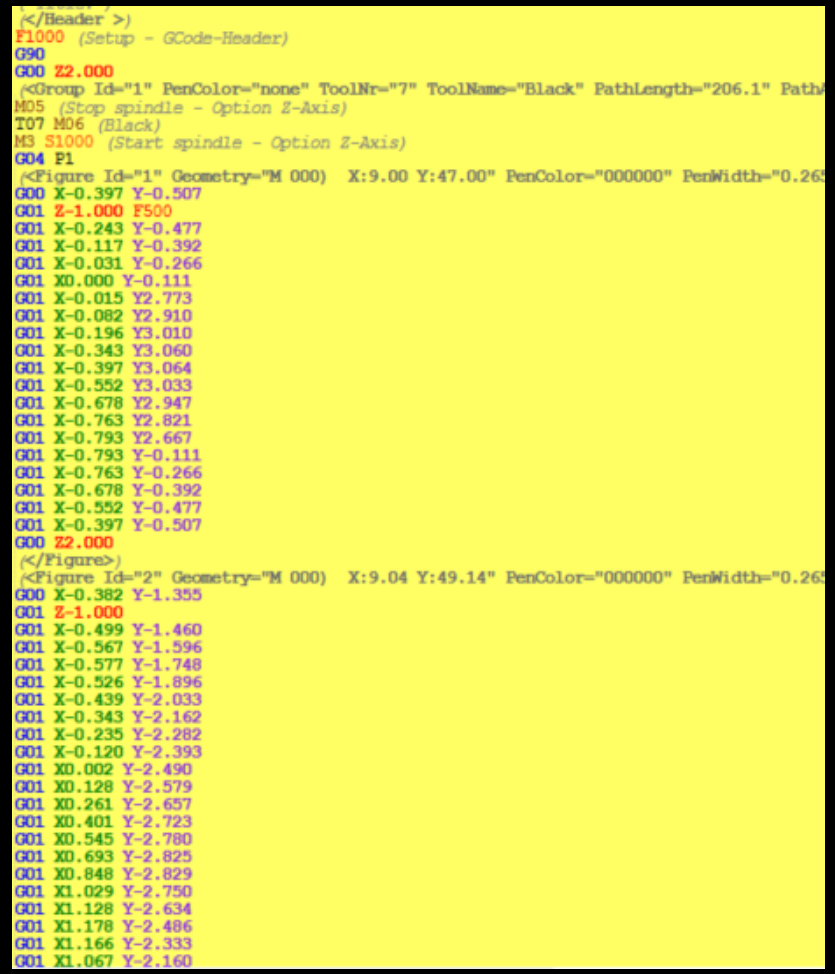
GUI




GRBL Plotter Ver.:1.6.6.6
File
G-Code Creation
G-Code Transform
Workpiece
Machine control
View
About
G-Code Streaming
Progress
Time
Check Code
Start path simulation
Laser Mode not active $32=0
Overrides (grbl 1.1 only) - click to expand / shrink
G-Code Dimension
Dimensions
Origin XY
Offset X
Offset Y
Apply Offset
1 (Paste G-Code or load file)
Tool Coordinates (Work / Machine)
G54
X -39.002 0.000
Y -59.184 0.000
Z -2.400 0.000
Zero X
Zero Y
Zero Z
Zero XY
Zero XYZ
Status:
idle
Home
Custom Buttons
Graphic Upper-Left
Move around graphic
Graphic Upper-Right
Pen up Zmax
Graphic Center
Pen down Zmin
Graphic Bottom-Left
Add Logo
Graphic Bottom-Right
Homing 0:0:35
Set Coordinate System
Hard RESET
GRBL
Plotter
Router
Plotter
Laser
Spindle On
CW
CCW
Min
Set
Max
Coolant
Mist
Tool is in Spindle
Control / Jogging
Move to
use G0
X=0
Y=0
Z=0
Move XY=0
STOP Jogging
STOP Jogging on mouse up
Feed Hold
RESET
Door
Resume
Kill Alarm
Paste URL of SVG / DXF file here
Add imported graphic to 2D-view
[Load file via drag and drop to 2D-view or select 'Reload File' (right click on 2D-view) to open last used file.]

GUI


COM CNC
COM6
115200
Close
Scan Ports
IP
Real-time Status Report
idle
X= -39.002 Y= -59.184 Z= -2.400
Bf: 15,128
Ln:
FS: 0,0
Ov: 100,100,100
Pn: XYZ
A:
Show Real-time Status Report
Check GRBL
[G92:0.000,0.000,0.000]
[TLO:0.000]
[PRB:0.000,0.000,0.000:0]
< ok
[VER:1.1f.20170302:]
[OPT:VH,15,128]
< ok
> $$
< $0=10 (Sets time length per step. Minimum 3usec.)
< $1=255 (Sets a short hold delay when stopping to let dynamics settle before disabling steppers. Value 255 keeps motor
< $2=0 (Inverts the step signal. Set axis bit to invert (00000ZYX).)
< $3=4 (Inverts the direction signal. Set axis bit to invert (00000ZYX).)
< $4=0 (Inverts the stepper driver enable pin signal.)
< $5=1 (Inverts the all of the limit input pins.)
< $6=0 (Inverts the probe input pin signal.)
< $10=2 (Alters data included in status reports.)
< $11=0.010 (Sets how fast Grbl travels through consecutive motions. Lower value slows it down.)
< $12=0.002 (Sets the G2 and G3 arc tracing accuracy based on radial error. Beware: A very small value may effect performa
< $13=0 (Enables inch units when returning any position and rate value that is not a settings value.)
< $20=0 (Enables soft limits checks within machine travel and sets alarm when exceeded. Requires homing.)
< $21=1 (Enables hard limits. Immediately halts motion and throws an alarm when switch is triggered.)
< $22=1 (Enables homing cycle. Requires limit switches on all axes.)
< $23=0 (Homing searches for a switch in the positive direction. Set axis bit (00000ZYX) to search in negative direct
< $24=25.000 (Feed rate to slowly engage limit switch to determine its location accurately.)
< $25=750.000 (Seek rate to quickly find the limit switch before the slower locating phase.)
< $26=250 (Sets a short delay between phases of homing cycle to let a switch debounce.)
< $27=2.000 (Retract distance after triggering switch to disengage it. Homing will fail if switch isn't cleared.)
< $30=255 (Maximum spindle speed. Sets PWM to 100% duty cycle.)
< $31=0 (Minimum spindle speed. Sets PWM to 0.4% or lowest duty cycle.)
< $32=0 (Enables laser mode. Consecutive G1/2/3 commands will not halt when spindle speed is changed.)
< $100=8.000 (X-axis travel resolution in steps per millimeter.)
< $101=8.000 (Y-axis travel resolution in steps per millimeter.)
< $102=4.000 (Z-axis travel resolution in steps per millimeter.)
< $110=400000.000 (X-axis maximum rate. Used as G0 rapid rate.)
< $111=400000.000 (Y-axis maximum rate. Used as G0 rapid rate.)
< $112=40000.000 (Z-axis maximum rate. Used as G0 rapid rate.)
< $120=30.000 (X-axis acceleration. Used for motion planning to not exceed motor torque and lose steps.)
< $121=30.000 (Y-axis acceleration. Used for motion planning to not exceed motor torque and lose steps.)
< $122=30.000 (Z-axis acceleration. Used for motion planning to not exceed motor torque and lose steps.)
< $130=360.000 (Maximum X-axis travel distance from homing switch. Determines valid machine space for soft-limits and homing
< $131=360.000 (Maximum Y-axis travel distance from homing switch. Determines valid machine space for soft-limits and homing
< $132=280.000 (Maximum Z-axis travel distance from homing switch. Determines valid machine space for soft-limits and homing
< ok
Clear Log
Send
$
$$
$#
$G
$I
$N
$X
RESET
RST


(</Header >)
F1000 (Setup - GCode-Header)
G90
G00 Z2.000
(<Group Id="1" PenColor="none" ToolNr="7" ToolName="Black" PathLength="206.1" Path
M05 (Stop spindle - Option Z-Axis)
T07 M06 (Black)
M3 S1000 (Start spindle - Option Z-Axis)
G04 P1
(<Figure Id="1" Geometry="M 000) X:9.00 Y:47.00" PenColor="000000" PenWidth="0.26
G00 X-0.397 Y-0.507
G01 Z-1.000 F500
G01 X-0.243 Y-0.477
G01 X-0.117 Y-0.392
G01 X-0.031 Y-0.266
G01 X0.000 Y-0.111
G01 X-0.015 Y2.773
G01 X-0.082 Y2.910
G01 X-0.196 Y3.010
G01 X-0.343 Y3.060
G01 X-0.397 Y3.064
G01 X-0.552 Y3.033
G01 X-0.678 Y2.947
G01 X-0.763 Y2.821
G01 X-0.793 Y2.667
G01 X-0.793 Y-0.111
G01 X-0.763 Y-0.266
G01 X-0.678 Y-0.392
G01 X-0.552 Y-0.477
G01 X-0.397 Y-0.507
G00 Z2.000
(</Figure>)
(<Figure Id="2" Geometry="M 000) X:9.04 Y:49.14" PenColor="000000" PenWidth="0.26
G00 X-0.382 Y-1.355
G01 Z-1.000
G01 X-0.499 Y-1.460
G01 X-0.567 Y-1.596
G01 X-0.577 Y-1.748
G01 X-0.526 Y-1.896
G01 X-0.439 Y-2.033
G01 X-0.343 Y-2.162
G01 X-0.235 Y-2.282
G01 X-0.120 Y-2.393
G01 X0.002 Y-2.490
G01 X0.128 Y-2.579
G01 X0.261 Y-2.657
G01 X0.401 Y-2.723
G01 X0.545 Y-2.780
G01 X0.693 Y-2.825
G01 X0.848 Y-2.829
G01 X1.029 Y-2.750
G01 X1.128 Y-2.634
G01 X1.178 Y-2.486
G01 X1.166 Y-2.333
G01 X1.067 Y-2.160

Budget

Part Name	Quantity	Cost
MGN15H Linear Rails	2	$88
Nema 17 Stepper Motors	3	$36
GT2 Belt & Pulley Kits	2	$50
Various bolts and nuts		$40
Linear Rod & Bearings	2 (each)	$20
Various 3D Printed Design Parts	20	$165
A4988 Stepper Drivers	1	$11
Servo Motor	1	$11
Limit Switches	1	$7
DC Power Supply	1	$13
Arduino Uno	1	$23
Arduino CNC Shield	1	$8
PCB	6	$165
Total	$624	

Issues

- Certain 3D printed parts not printed correctly.
- Mechanical design needing modifications when assembled.
- Z-axis movement hindered by mechanical design.

Future Goals

- Bluetooth module
  - Mobile application development
- Add more features to GUI
  - Better image conversion
- Improve mechanical design visually


Questions?