

## Simple & Cost-Effective Two-Axis Robotic Drawing Arm 003-4 ECE 342-W23

The purpose of this project is to have a fast, accurate, and easily interfaceable drawing robot. The usefulness of such a system would allow the customer to quickly draw different designs and make any adjustments as needed.

The first consideration of the project was how to meet customer requirements. The robot had to follow SCARA topology, involving two rotating joints. To accomplish this, FUSION 360 was used to create the models of the arm. An Ender 3 V2 was used to print the models using PLA plastic. An Arduino Uno 3 shield was used for our PCB to wire the cables to the correct locations.

To handle user input from the Python command line, pyfirmata was used to give the G-code command to the arduino using the standard pyfirmata library.

Inside the Arduino code, the first thing was to add the libraries or headers we need for the calculations and math, then assign the PINs on the board for two motors. The 'kinematic' function was the main part of calculating the angle by passing in the destination coordinates, that would create a triangle with the X and y axis. By applying the cosine and sine rules, we can get the angle for both motors. Since we decided to implement our second motor upside down and both motors are rotating clockwise from 0 to 180 degrees, only the first motor angle needs to be 180 minus the calculated result to create a normal X-Y plane to draw on paper. With the concept of angle calculation, the next thing was to make the path to a straight line instead of an arc. In the function called 'linear', if we have two points, we can get a line equation, then chop the line into many small pieces and pass all the coordinates of those points to the kinematic function to drive the motors. Last things in the code were 5 different drawing functions, depending on the amount of sides the shape has, there will be the same number of times calling the linear function.

Some key challenges include changing our block diagram late into the project, integrating our blocks late, and PCB shipment issues. When we were starting on the project, we had difficulty evenly dividing the project into 6 blocks, resulting in later revisions after our block 1 check-offs. This resulted in some miscommunication in interfaces and made integrating everything difficult. There were also many issues integrating translation with the microcontroller, mainly due to lack of experience with Python, pyfirmata, and Arduino.

Some ways to improve the project would be to revolve the project around higher power and more accurate motors. The two S3003 Servo motors are cheap and extremely accessible, but for better performance, more research and testing should have gone into choosing the motors. The project has also been eye opening for the general design process. More care should have been taken in the timeline.



# Project Timeline

