

# **Final Report**

## **I am already Tracer**

### **Introduction**

#### *Statement of purpose*

Our project was a vertical plotter that could be controlled using a Resistive Touch Overlay in real time. Essentially, the primary goal of this project was to create a drawing robot to trace an image drawn by a user such that a user can see a printed version of their drawing simultaneously.

### **Design details**

Using a resistive touch overlay, a user will be able to draw an image that will be traced on a separate surface by the plotter in real time. The plotter will be controlled by two stepper motors, moving along a fixed x-y plane. The plotter was designed with a redboard at its core and was programmed using the arduino IDE.

### **Parts**

- Redboard
- 2 Stepper Motors
- Motor Drivers for stepper motors
- Resistive touch Overlay
- Weights
- Whiteboard
- Gondola and stepper mounts (3D printed)
- Marker
- Power Adapter
- Pulley
- Chain

## Circuit schematic

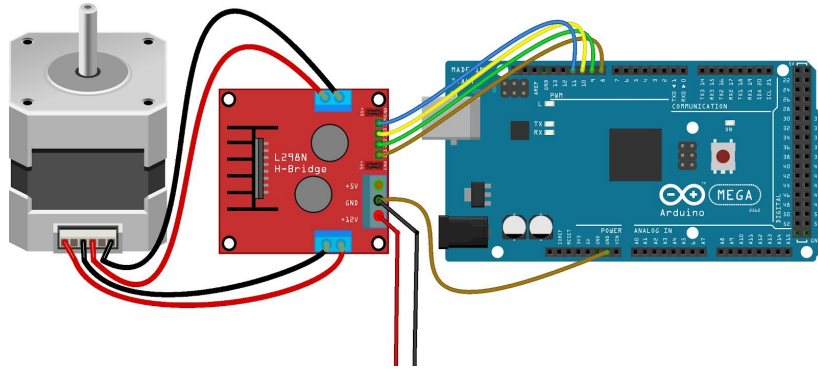


Figure 1: Stepper motor connections to the arduino <sup>1</sup>

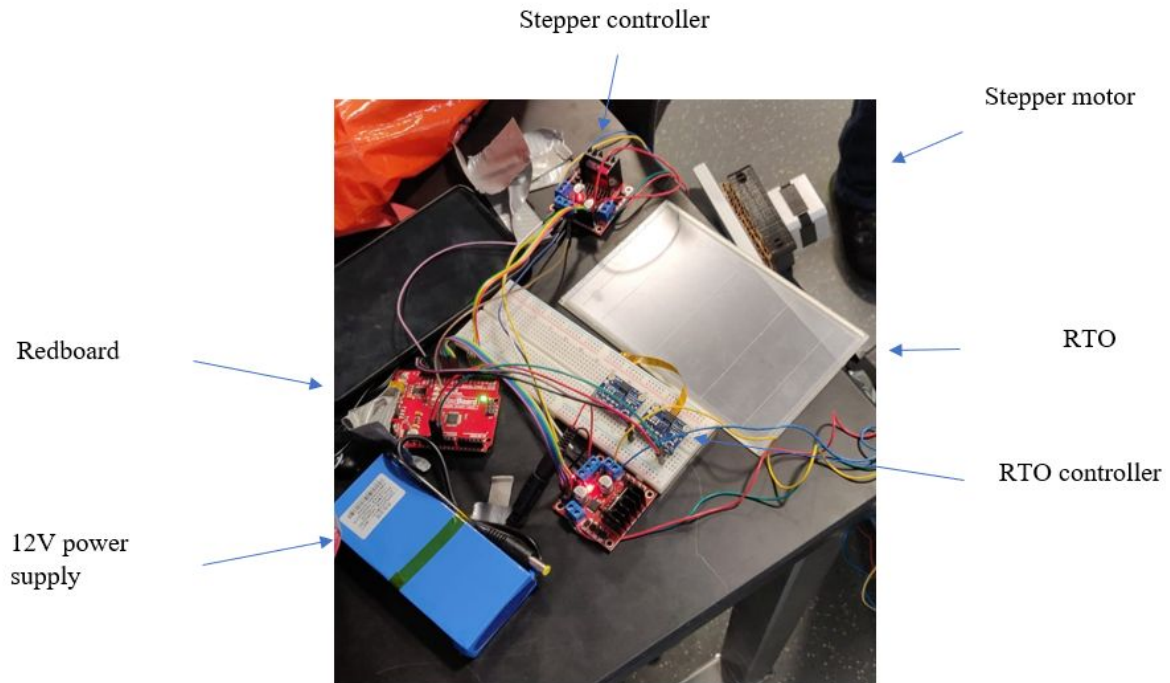


Figure 2: all circuit connections

<sup>1</sup> "How to Run Stepper Motor with Arduino using L298N ... - Circuit Magic." 29 Mar. 2018, <https://www.circuitmagic.com/arduino/how-to-run-stepper-motor-with-arduino-using-l298n-driver-module/>. Accessed 2 May. 2019.

**Block diagram**

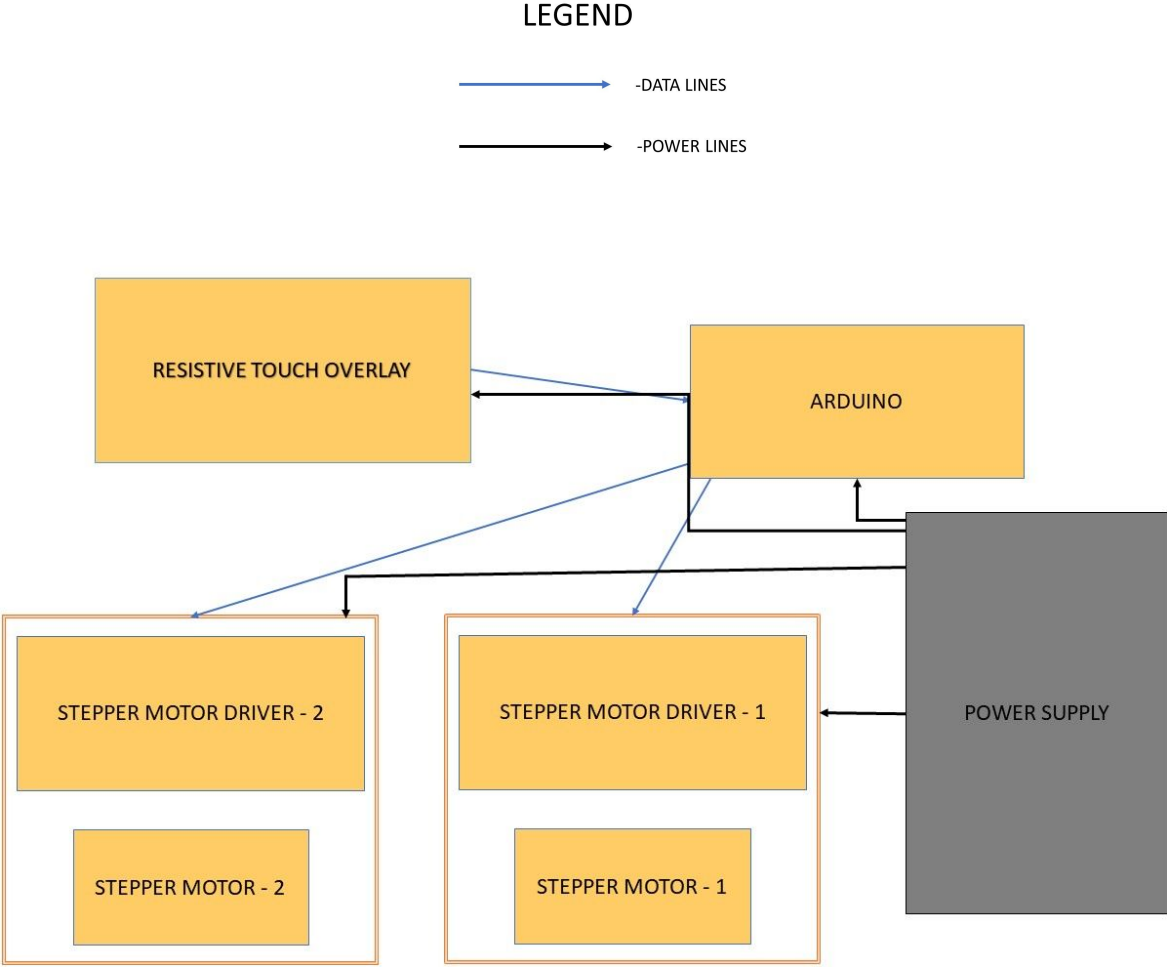


Figure 3: Block diagram

**Sensor characterization**

For this project we used a 165mm x 105mm (7" diagonal) resistive touch overlay. This sensor is designed such that there are two potentiometers set up inside it. When any point on it is pressed, we get two separate voltage values, one for x and one for y, which can be read in using the arduino to determine the position of where we touched it.

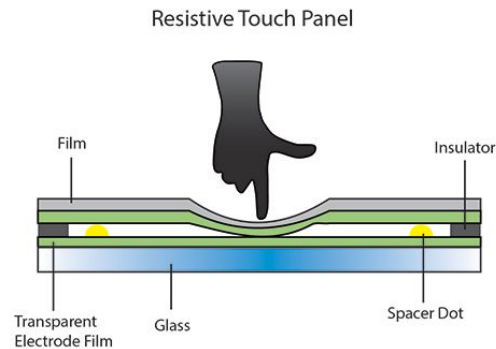


Figure 4: Resistive touch Overlay design<sup>2</sup>

### Code outline and logic

To interface with this touch screen, we used a Resistive Touch Screen Controller (Adafruit's STMPE610). This breakout board took the raw values from the touch screen and using I2C, it sends the data to the Arduino. It uses interrupt based I/O to send data which reduces the complexity of having to manually poll the data. Adafruit also provided us with an arduino library for this breakout board which gave us the ability to conveniently get the coordinates, status of the screen along with the ability to use a buffer to sequentially process inputs and avoid race conditions. We used a map function to scale the input to the size of our output board. When used these values to calculate the length of the pulley that needs to be shifted for each stepper and we converted that length into steps and called the step() function defined in the stepper library to control the stepper. We moved each stepper one step at a time to simulate simultaneous control.

The code is available on our Github: <https://github.com/zatch10/I-am-already-Tracer>

---

<sup>2</sup> "Technology within Touch Panels - How Do Touch Screens Actually ...."  
<https://www.epectec.com/articles/the-technology-within-touch-panels.html>. Accessed 2 May. 2019.

## **Problems & Challenges**

We ended up running into quite a few problems along the way due to shipping and our lack of experience. For example, we originally had a plan to use a servo to lift up the pen so that we can have two or more distinct shapes which are not connected to each other, but we never got the time to test things out and this feature had to be removed. Due to our lack of experience with Arduino and stepper motors, it took us way longer to set up the stepper motors than it should have. Finally, due to shipping problems, we had some of the most important parts like the stepper motors and belt come in way too late, which made testing and refining of the system very difficult.

## **Future Plans**

Having created the basic structure for the vertical plotter, we hope to try and use openCV to directly draw images from a computer. On a more ambitious note, we also want to try and create a drawing of a person from a photo.

## **Conclusion**

To conclude, we can say that while we did not necessarily get a perfect drawing of shapes drawn on the RTO, this project was pretty much a success since we could get a rough jaggedy outline of the shape drawn. This was a big learning experience for each one of us, as we got to learn a lot about Arduino and using external moving components with it, and we also got a chance to use a lot of the concepts we learned in class.

## Final Design

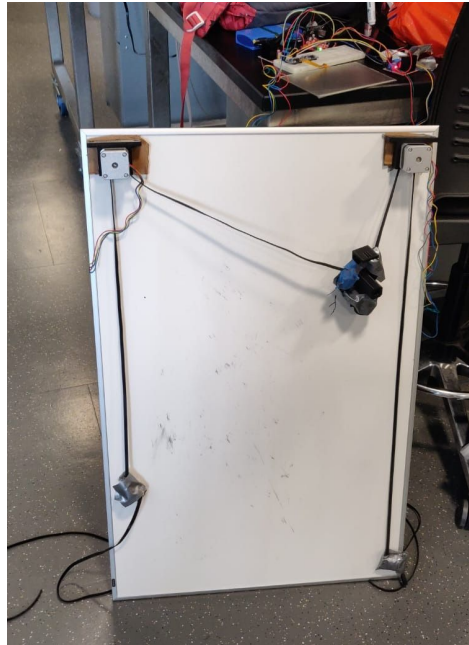


Figure 5: The complete vertical plotter

## Bibliography

- "How to Run Stepper Motor with Arduino using L298N ... - Circuit Magic." 29 Mar. 2018, <https://www.circuitmagic.com/arduino/how-to-run-stepper-motor-with-arduino-using-l298n-driver-module/>. Accessed 2 May. 2019.
- Instructables. "XY Plotter Drawing Robot | Arduino | Polargraph." Instructables. June 06, 2018. Accessed May 02, 2019. <https://www.instructables.com/id/XY-Plotter-Drawing-Robot-Arduino-Polargraph/>.
- Adafruit Industries. "Resistive Touchscreen Overlay - 7" Diag. 165mm X 105mm - 4 Wire." Adafruit Industries Blog RSS. Accessed May 02, 2019. <https://www.adafruit.com/product/1676>.
- "Technology within Touch Panels - How Do Touch Screens Actually ...." Accessed 2 May. 2019. <https://www.epectec.com/articles/the-technology-within-touch-panels.html>.

By: Chirag Nanda (110), Vatsin Shah (110), Prannav Gupta (110)