

# Sensor & Motor Control Lab

S. M. Bryan

Team A / The Avengers

Teammates: Tushar Agrawal, Pratik Chatrat, & Adam Yabroudi

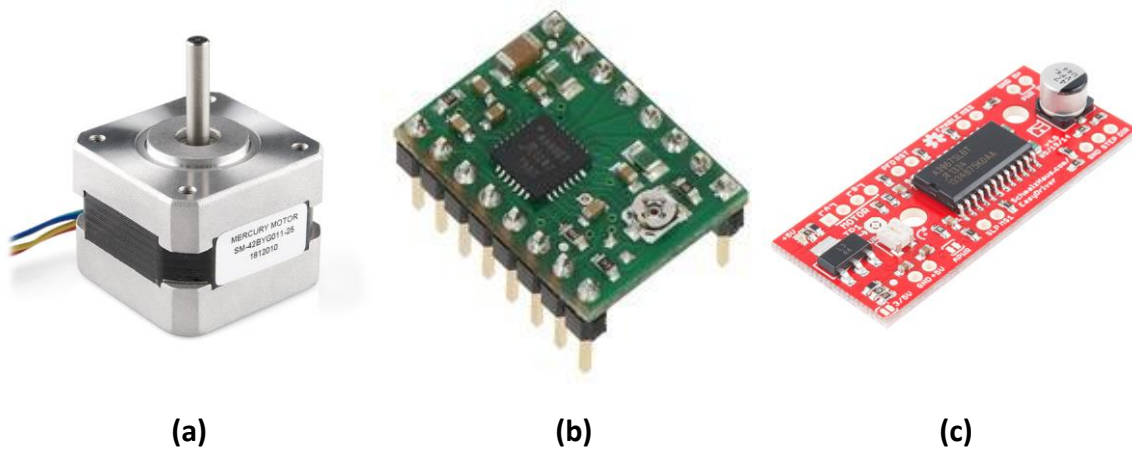
ILR01

October 16, 2015

## 1. Individual Progress

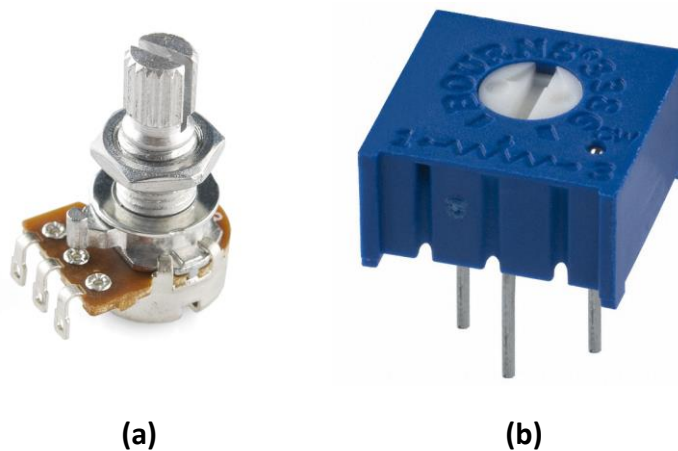
For the Sensor and Motors Lab, I selected to work on the stepper motor and potentiometer.

The stepper motor used was the Mercury Motor's SM-42BYG011-25 (figure 1.a) driven initially by the Pololu A4988 stepper motor driver (figure 1.b). However, due to issues I will discuss later, the Pololu was replaced with a SparkFun EasyDriver (figure 1.c). The wiring diagram is given below (figure 3).



*Figure 1: Stepper Motor (a) and drivers (b & c) used in Task 7*

As a sensor, I selected to use a rotary potentiometer. Initially, I used the COM-09939 (figure 2.a). This too had to be replaced, and the final component used was a Bourns Trimpot 3386 (figure 2.b). The wiring diagram for the potentiometer is given in figure 4.



*Figure 2: Potentiometers used in Task 7*

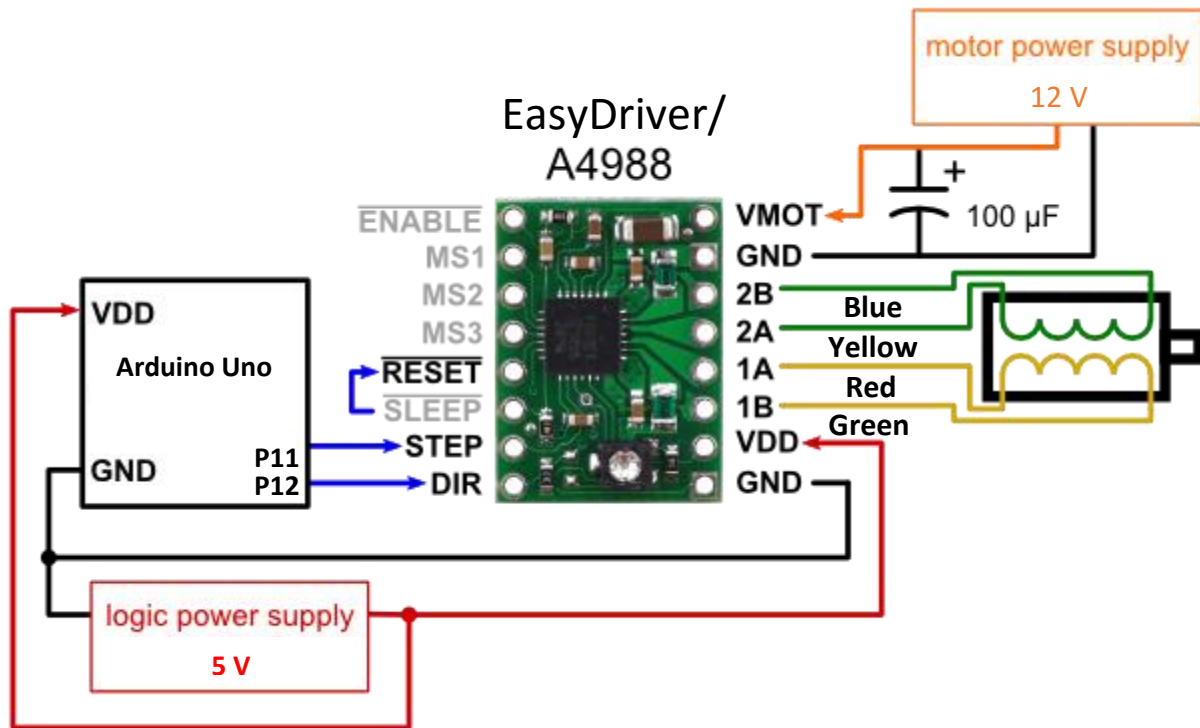


Figure 3: Stepper Motor Circuit (source: <https://www.pololu.com/product/1182>)

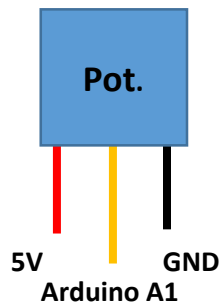


Figure 4: Potentiometer wiring diagram

Since I had been warned about the temperamental nature of the Pololu boards (including numerous warnings on the company website), I started with the potentiometer before wiring the stepper motor and driver.

I wired the potentiometer and programmed the Arduino to print the reading via serial. From that, I was able to determine that the range was from 0 to 1024. Next, I calculated the number of steps in a full rotation of the stepper. The Mercury SM-42BYG011-25 is a bipolar stepper motor capable of 1.8° steps.

$$Steps = \left( \frac{360 \frac{degrees}{rotation}}{1.8 \frac{degrees}{step}} \right) = 200 \frac{steps}{rotation} \quad (1)$$

Knowing that I wanted to use the potentiometer to control the position of the stepper, I then calculated the range of values for each step.

$$Range = \frac{1024}{200} = 5.12 \quad (2)$$

The range per step allowed me to simply map the potentiometer position to the range of values used in the stepper by dividing the analog reading by 5.12.

## II. Challenges

There were significant challenges in this project associated with poor components. Both the original potentiometer and stepper motor driver had to be replaced.

The first hurdle was with the potentiometer COM-09939. After wiring it to the Arduino and reading the outputs, I noticed significant noise in the signal which is unusual for potentiometers. Moreover, I noticed that the max range was inconsistent and not in-line with manufacturer specifications. The max reading was consistent with hysteresis; it would range from 60-80% of the correct value on different attempts.

After checking my wiring, I tested the potentiometer with the multimeter. Voltage across the input and ground was correct at 5v. However, the voltage across the signal to ground was erratic without rotating the potentiometer. The potentiometer had been soldered by a previous class and was probably done incorrectly. I elected to replace the potentiometer and move on.

The second challenge came from the stepper motor driver. The Pololu website contains numerous warnings about the hazards of modifying the wiring of the A4988 while the board is receiving power. Despite being careful, the board failed to move the stepper motor upon initial tests. I carefully reviewed my wiring and checked the voltage across both  $V_{MOT}$  and  $V_{DD}$ . As a test, I decided to replace the A4988 which allowed me to successfully move the stepper motor.

However, the trouble with the Pololu A4988 didn't end there. Upon consolidating the circuit as a team, this board too was damaged. Unable to recover the board, we switched to the SparkFun EasyDriver which required soldering. Fortunately, changes to wiring were minor and no changes were required to the code.

### **III. Teamwork**

Each team member took on different components of the task. This allowed the team to build their individual skillsets while expediting the work.

*Tushar Argawal*

Tushar took on the task of employing the DC motor and ultrasound sensor. Additionally, he was primarily responsible for the integrating the code from individual team members which made him ideal to present the project.

*Pratik Chatrat*

Pratik tackled the GUI in order to improve his programming abilities. He was able to design an interface to control all motors and gave the user the option to switch between direct and sensor control.

*Adam Yabroudi*

Adam was primarily responsible for the DC servo and IR sensor. He also integrated the separate circuits into one board for the team. As the resident electronics expert, I relied heavily on Adam's guidance in troubleshooting my own circuit.

### **IV. Plans**

Our project relies heavily on integrating ready-made subsystems. Thus, it's important to test individual subsystems and their interactions. I will be fabricating a layout board for the electronics that will allow the team to test different sensors, mounting configurations, and wiring options without requiring the actual drone.

Additionally, I will investigate acquiring the 'gripper' for our project. The Electro Permanent Magnet by NicaDrone is ideal for our project. However, they are currently out of stock. As an alternative, we are investigating building the system from provided schematics and sending off to have the board made. This will be a new experience for me.