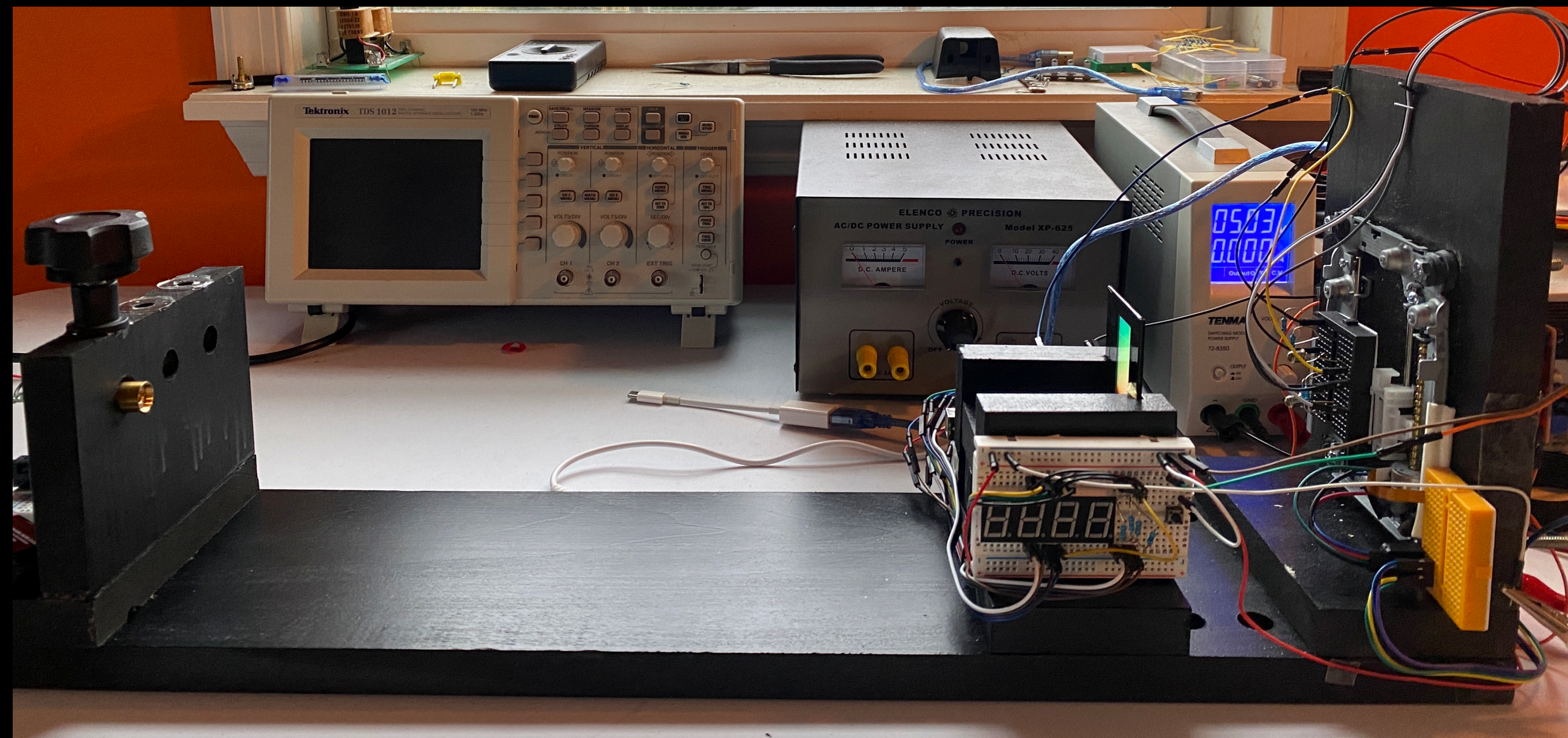


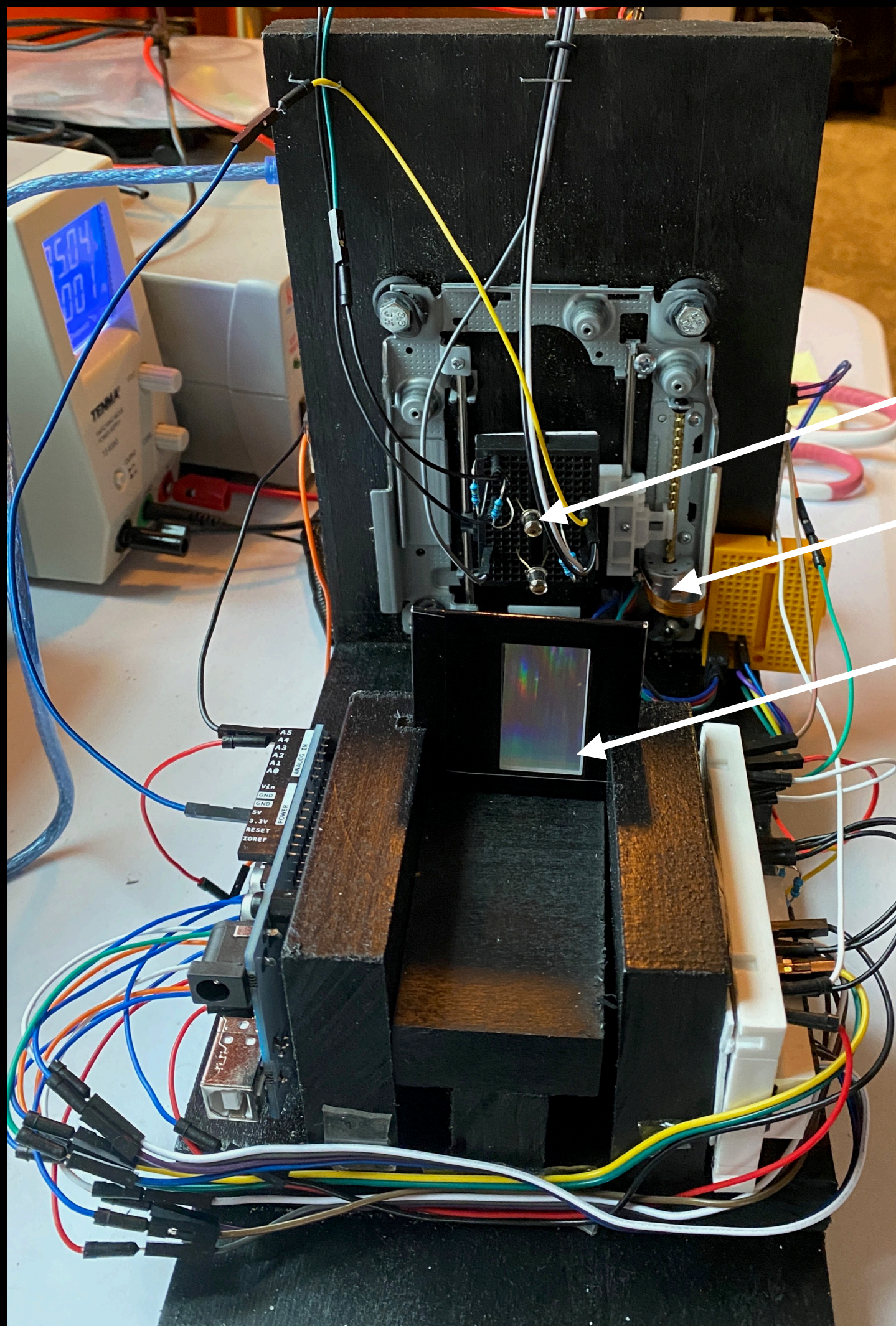
# Inexpensive Diffraction Spectrometer for Monochromatic Light

William Suber III  
Boston University  
Electronics & Advanced Lab  
December 7, 2020





# Key Components:



Arduino (1) Microprocessor

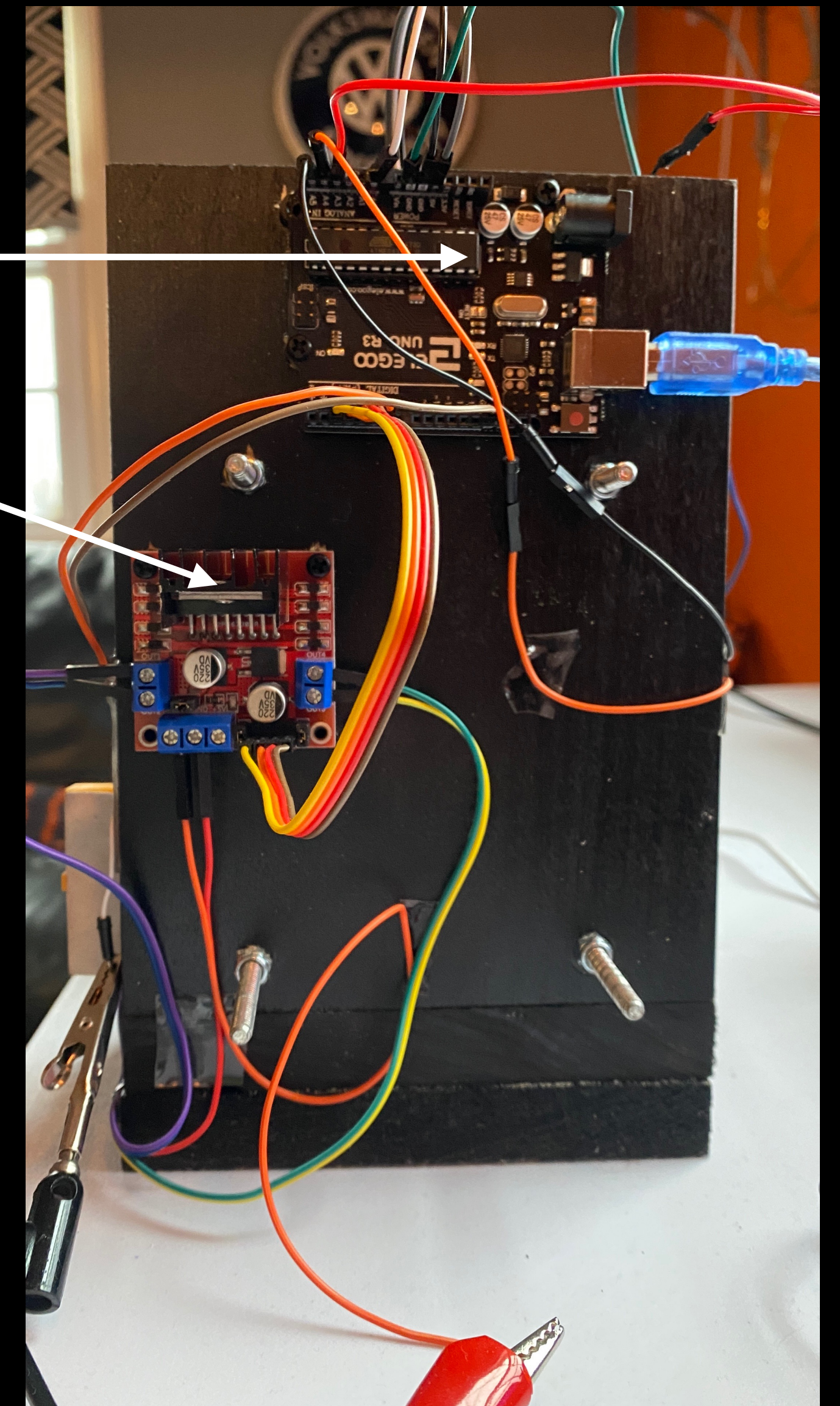
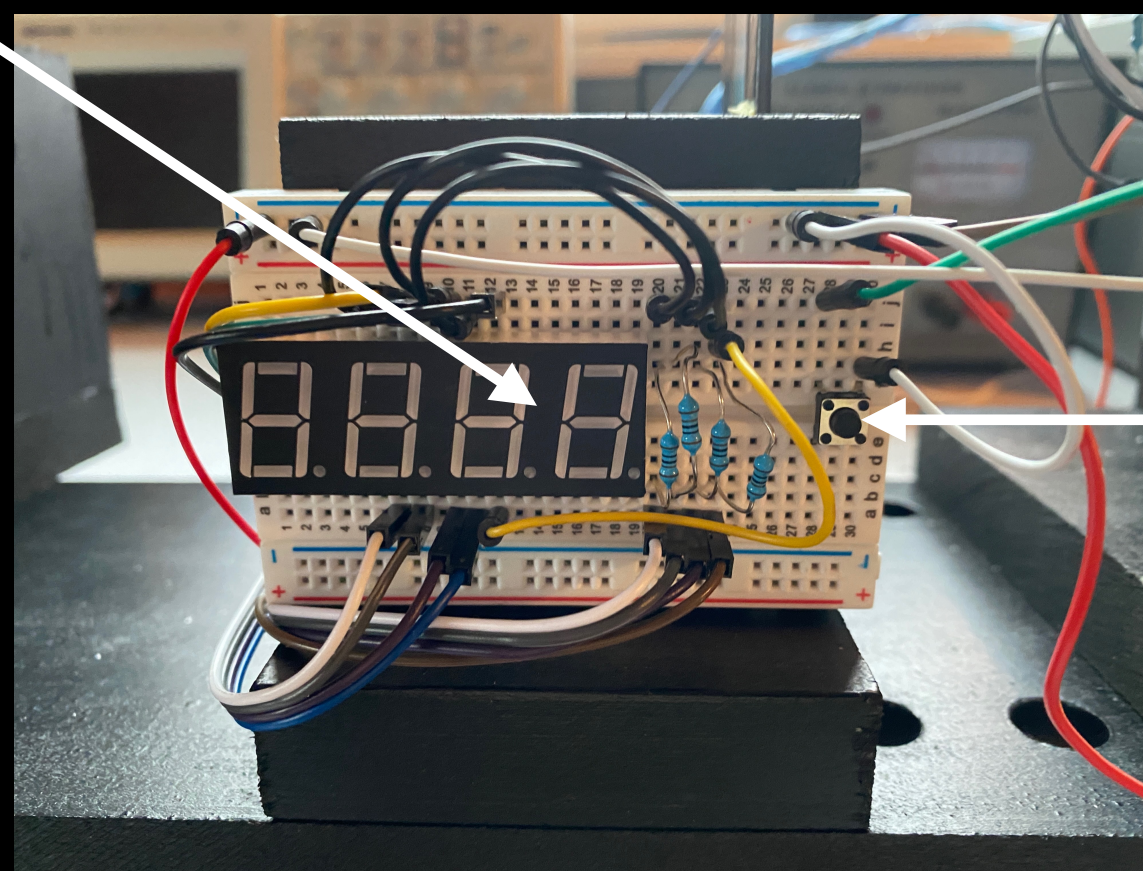
H-Bridge Motor Driver

Phototransistor in sensor bay

Bipolar Stepper Motor

600 line/mm Diffraction Grating

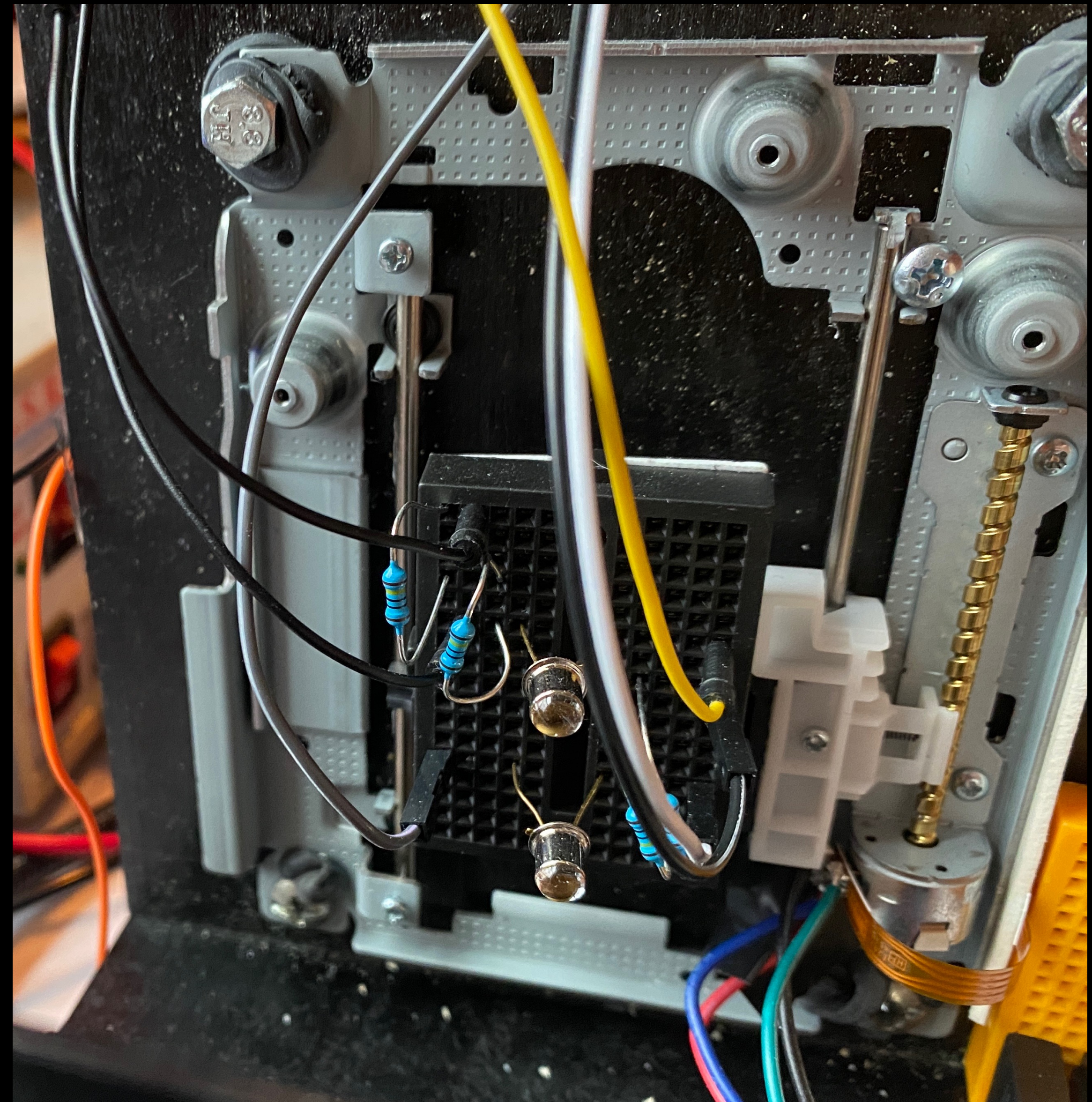
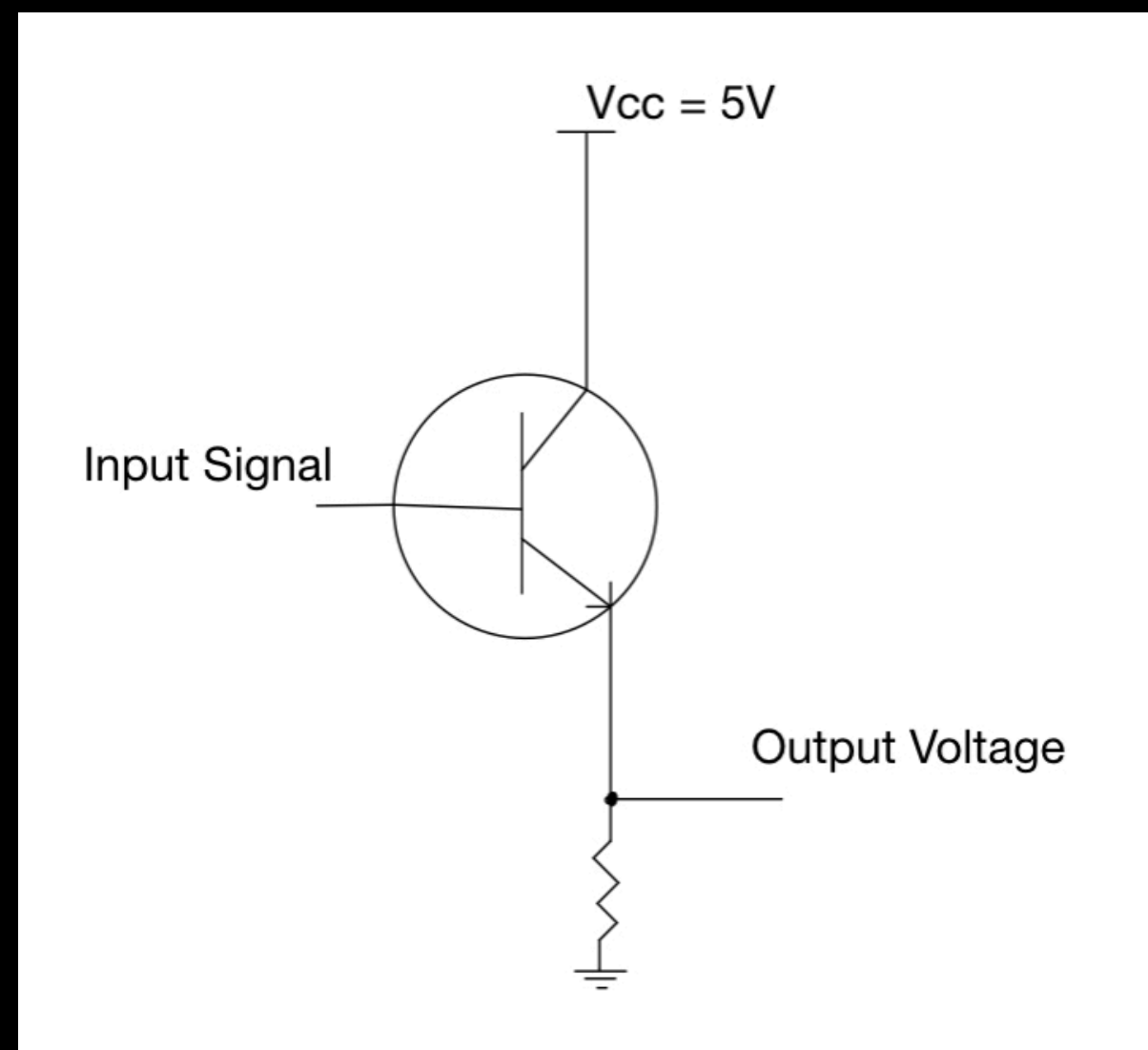
Seven-segment display





## Sensor Bay:

- Two Phototransistors for input
- Independent emitter-followers
  - Current amplifiers with output voltage proportional to signal amplitude at base
  - Signal is light intensity for phototransistors

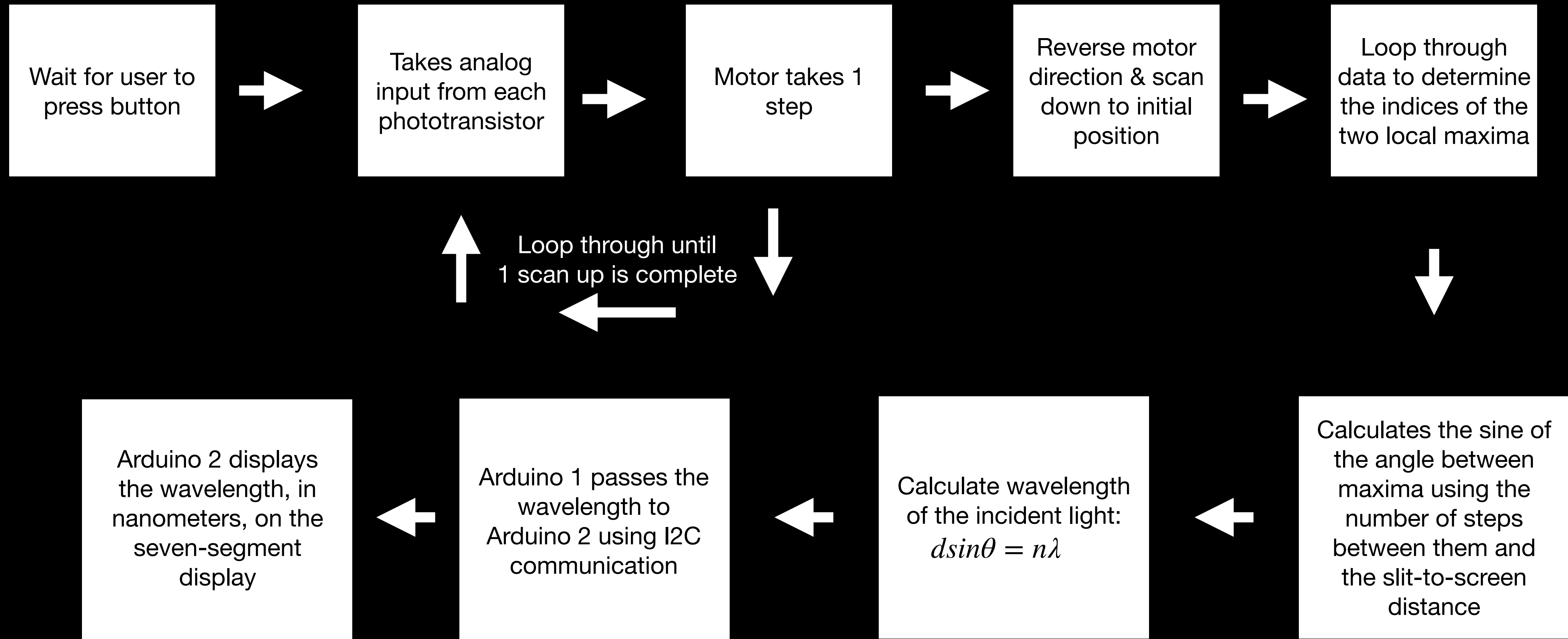




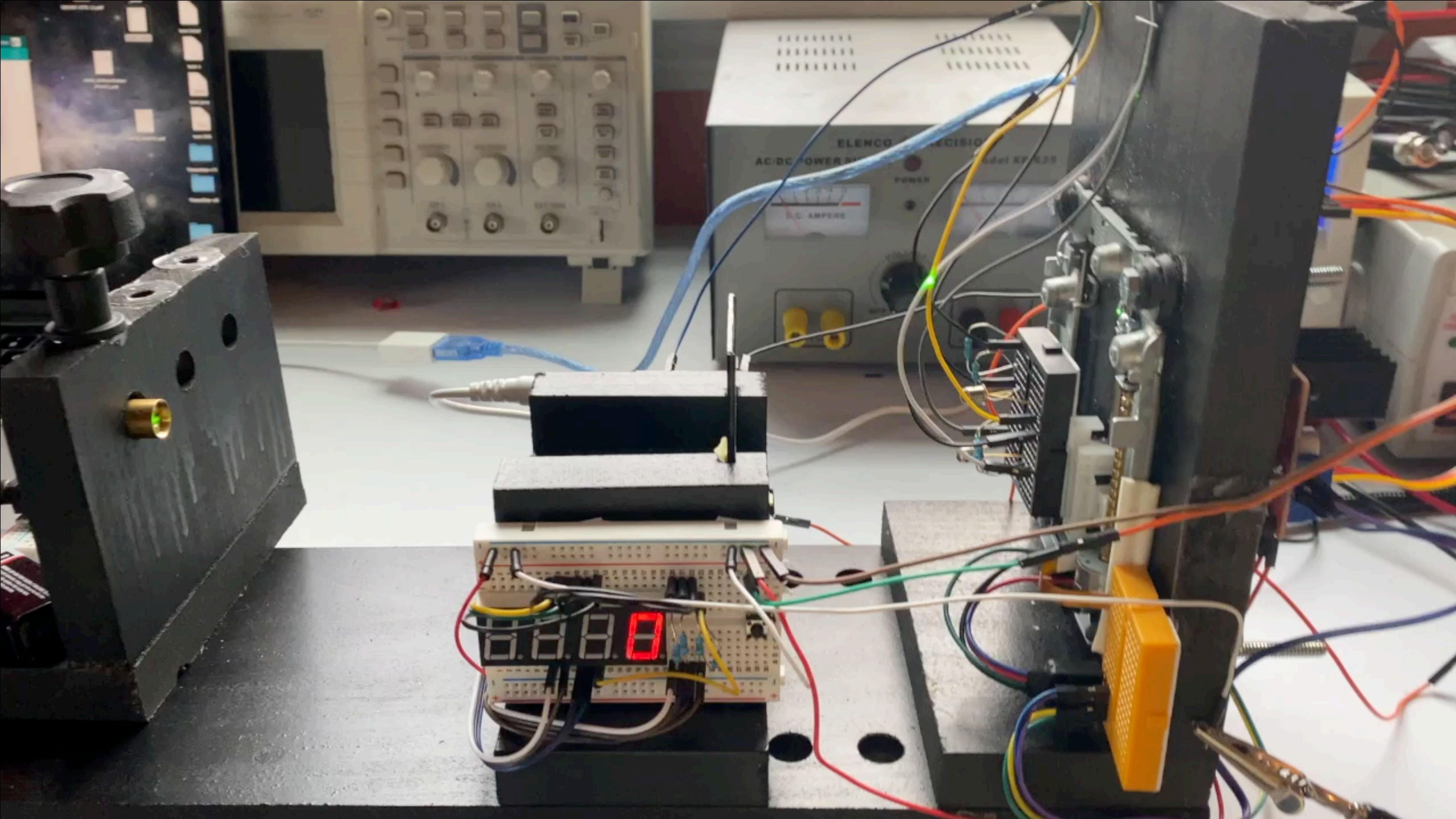




# Block Diagram of Software







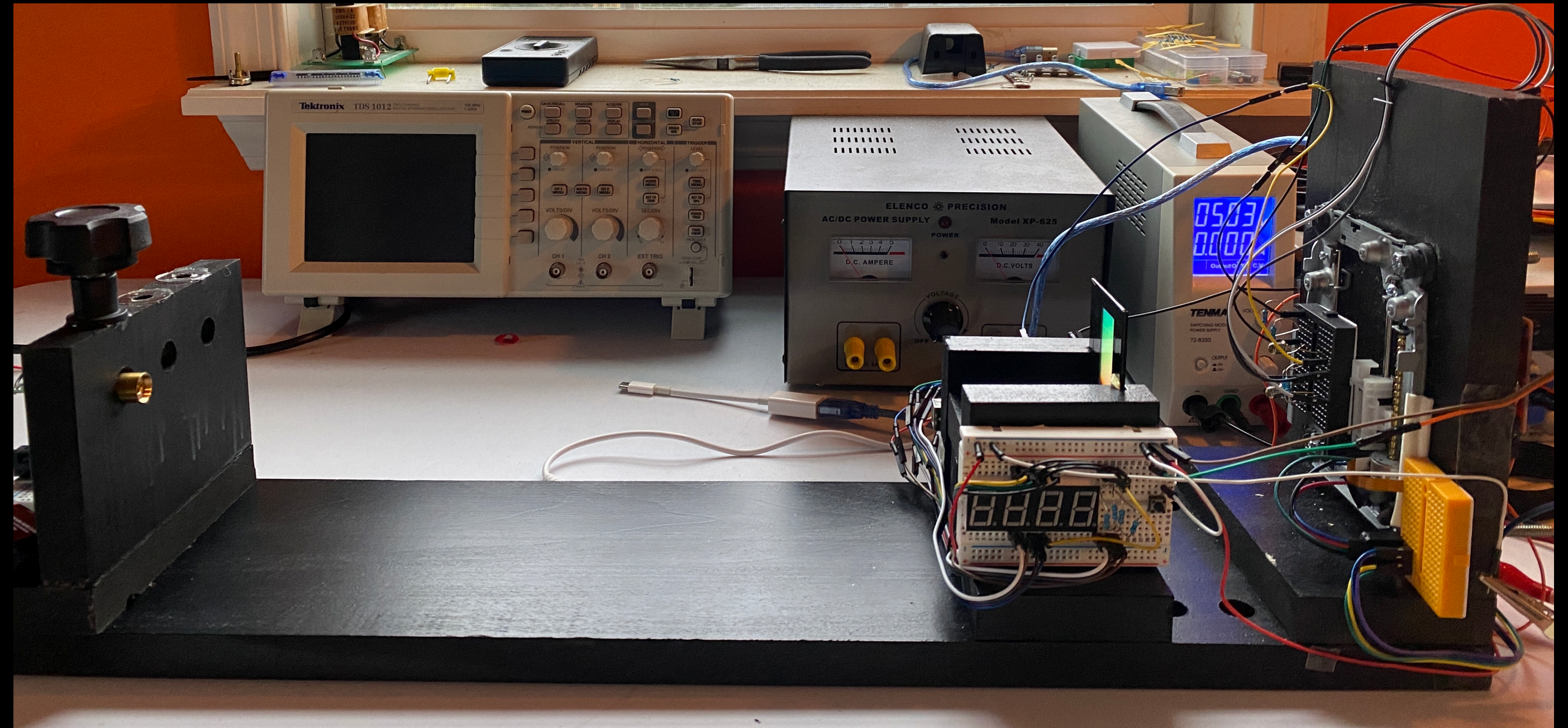






## Summary:

- Arduino 1 & L298N to actuate a stepper motor while taking analog readings from two independent phototransistor emitter-followers
- Produces an array of relative intensity values, from which Arduino 1 determines the distance between maxima
- Using:  $d \sin \theta = n \lambda$  Arduino 1 calculates  $\lambda$ , passing it to Arduino 2 using I2C communication to display on seven segment display



## Future Enhancements:

- Build larger rail mounted sensor bay with new motor for larger swing to pick up additional maxima
- Use LCD multiplexed display over seven-segment display, no longer need Arduino 2

**Thank You For Your Attention!**



# I2C Communication

## Using <wire.h> header file:

```
#include<Wire.h> //This library is used for I2C communication
float lambda;
int intWL;
int send_data[2];

void setup() {
  Wire.begin(); //used for communication between Arduino 1 and 2
  Serial.begin(9600); // Begin serial communication at 9600 Baud
}

void loop() {
  lambda = slit_spacing*sinTheta; //float lambda
  intWL = lambda / 1 ; //Truncated INT lambda
  send_data[0] = intWL / 100; //hundreds place of lambda to pass to Arduino 2
  send_data[1] = intWL - send_data[0]*100; //tens, ones places of lambda to pass to Arduino 2
  send_data[0] *= 10; //Would rather send 60 than 6, just in case the value is changed by 1 during transfer

  Wire.beginTransmission(9); //9 is simply an address for Arduino 2
  Wire.write(send_data[0]); // Write the hundreds place
  Wire.write(send_data[1]); //write tens and ones place
  Wire.endTransmission(); //close buffer that held the two above bytes while they waited to be sent.
}
```

```
#include<Wire.h>
int i = 0;
int x1;
int x2;
int x[2];

void setup(){
  Wire.begin(9); //9 = address
  Wire.onReceive(getLambda); //point to data handling function
  Serial.begin(9600);
}

void getLambda(int data) {
  for(i = 0; i < bytes; i++){
    x[i] = Wire.read(); //read both bytes of data, one per loop iteration
  }
  lambda = x[0]*10 + x[1] ; // 'build' the proper lambda from the data passed
}

void loop(){
}
```