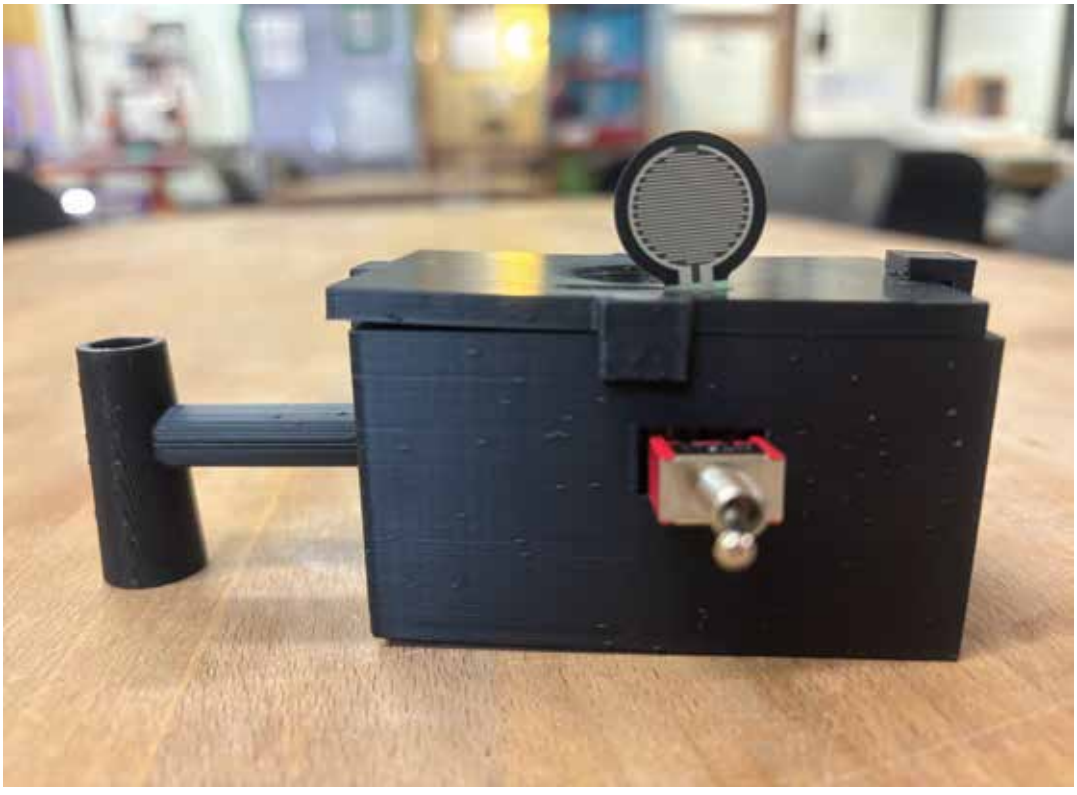
Aidan McLaughlin

ATLAS 3300 Object

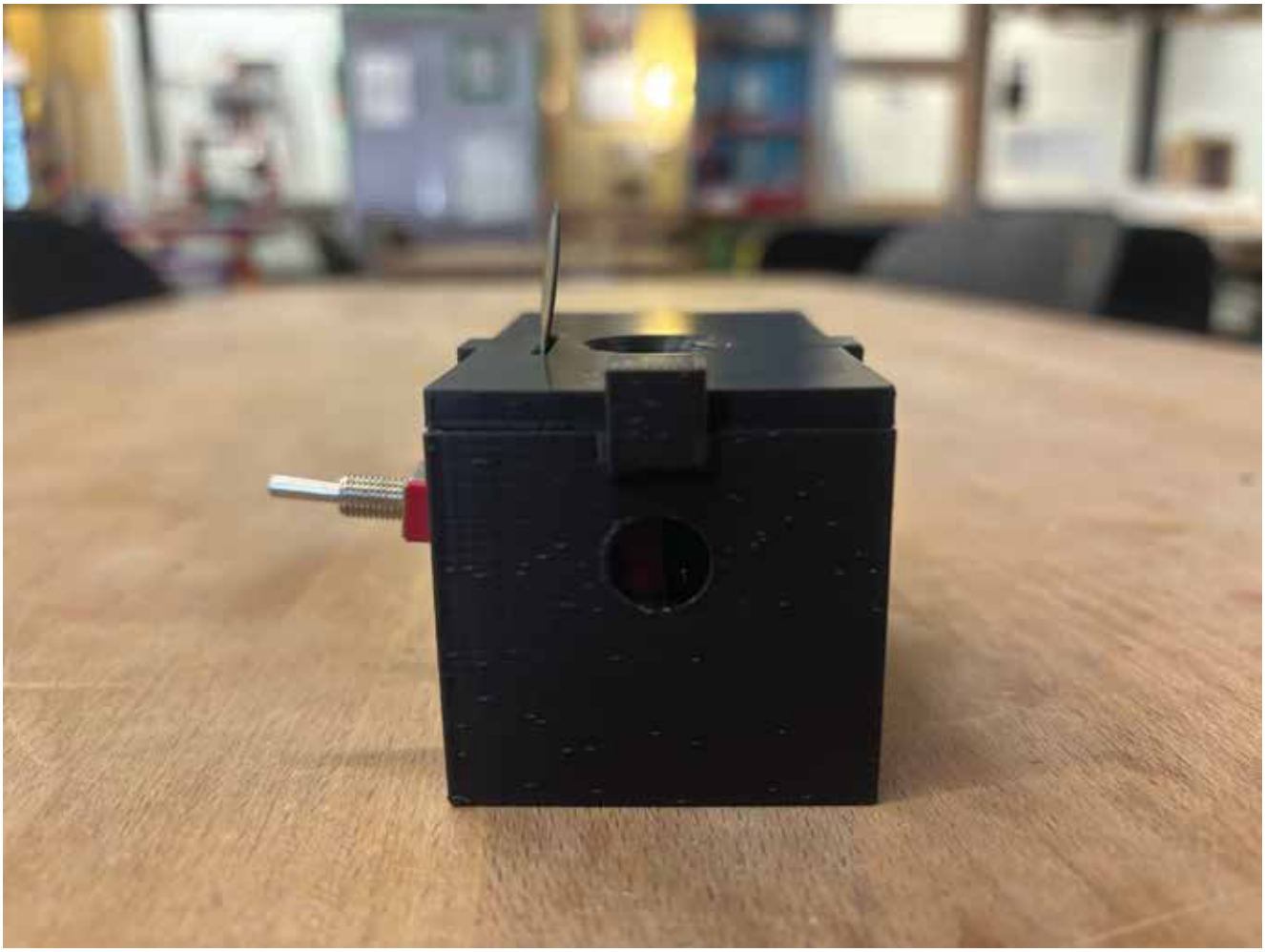
IDC #2

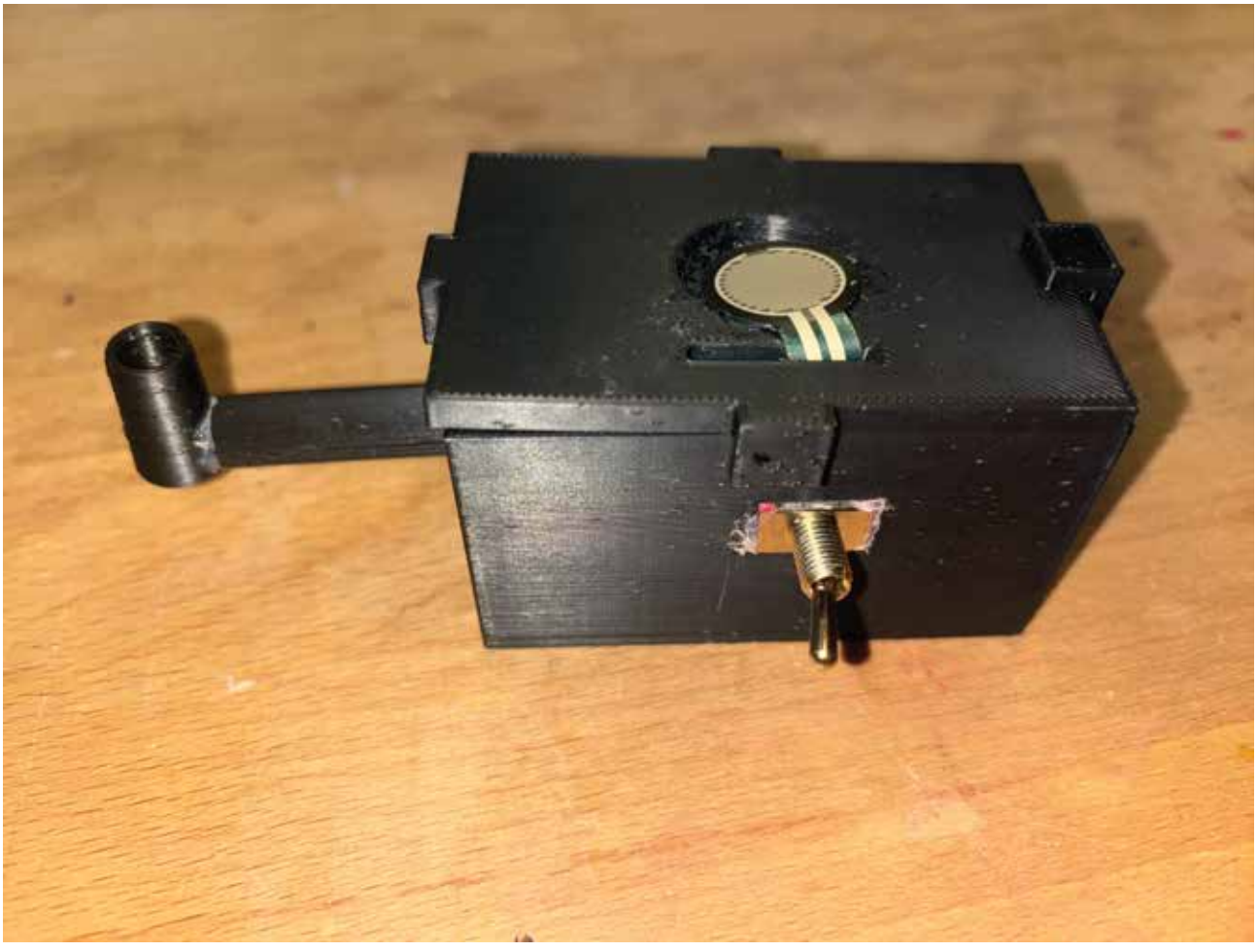
Monday, May 5th

The PASS THE J! device



(The “J” is made completely out of card board btw)





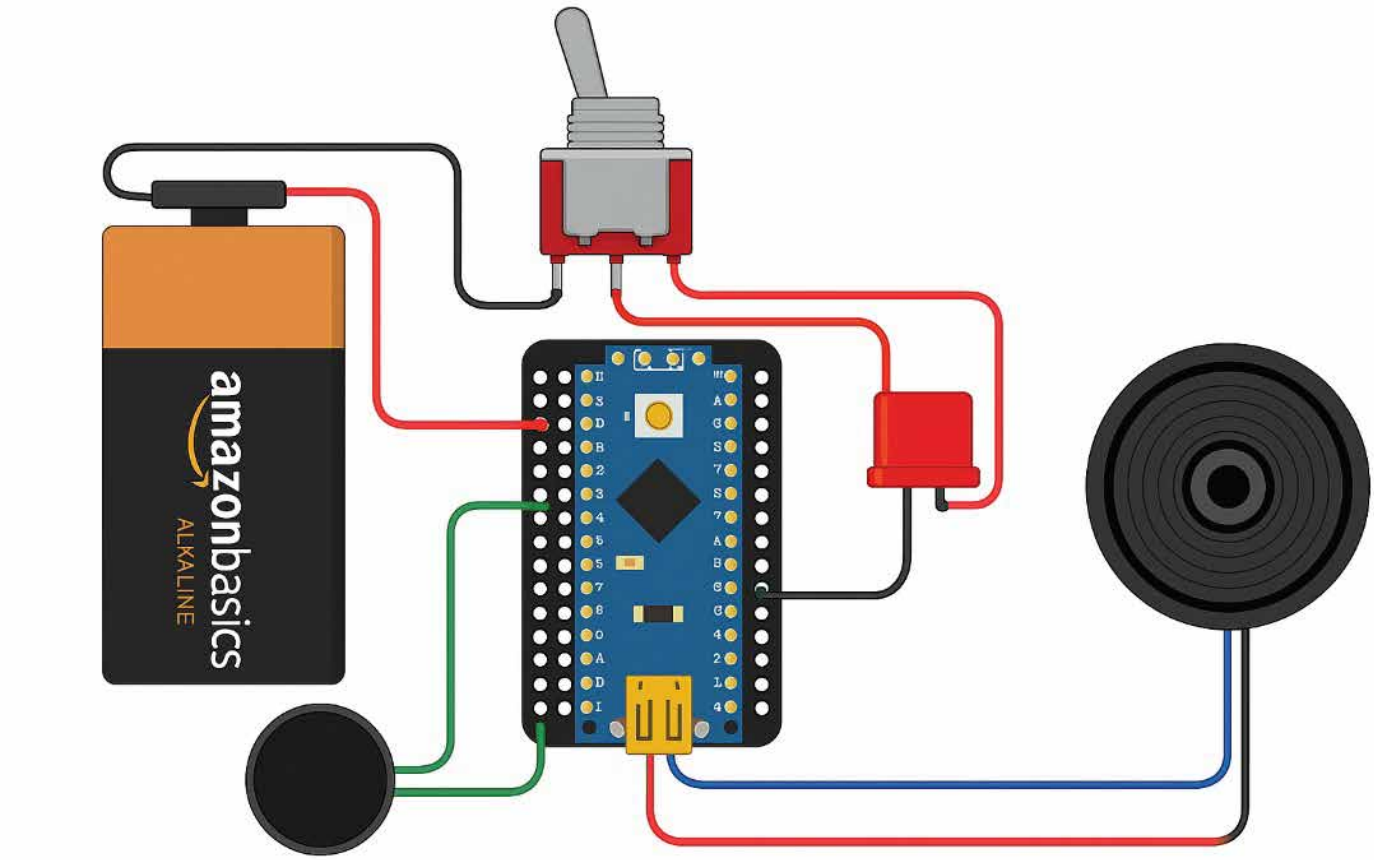
Problem:/Inspiration

Have you ever been enjoying a nice J with your friends and someone has dove deep into a story? You enjoy the company and the story but after a while you realize, this person has forgotten to pass the J.
How do you tell the person to pass the J without seeming rude?

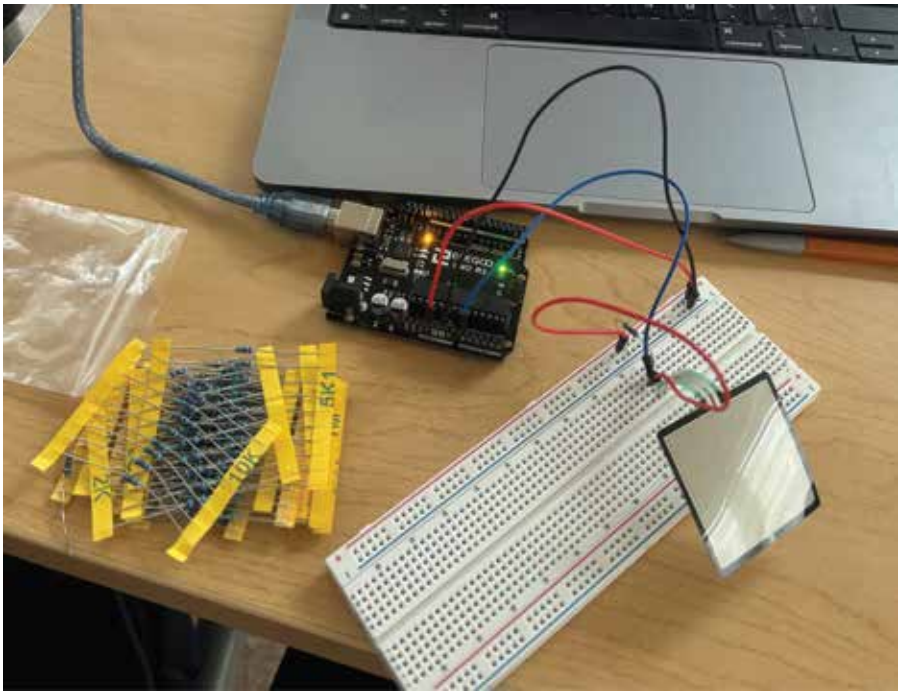
Solution:

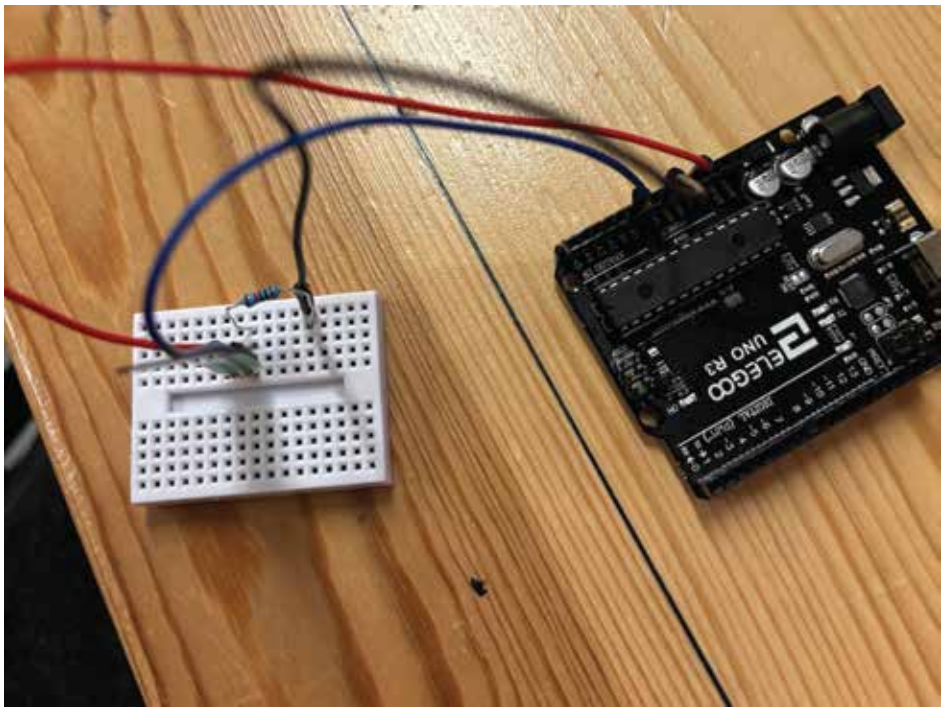
The solution? The PASS THE J device!
The PASS THE J device uses a pressure sensor to sense when someone has been holding onto a J for too long. No longer will the J stand stagnant and waste away!

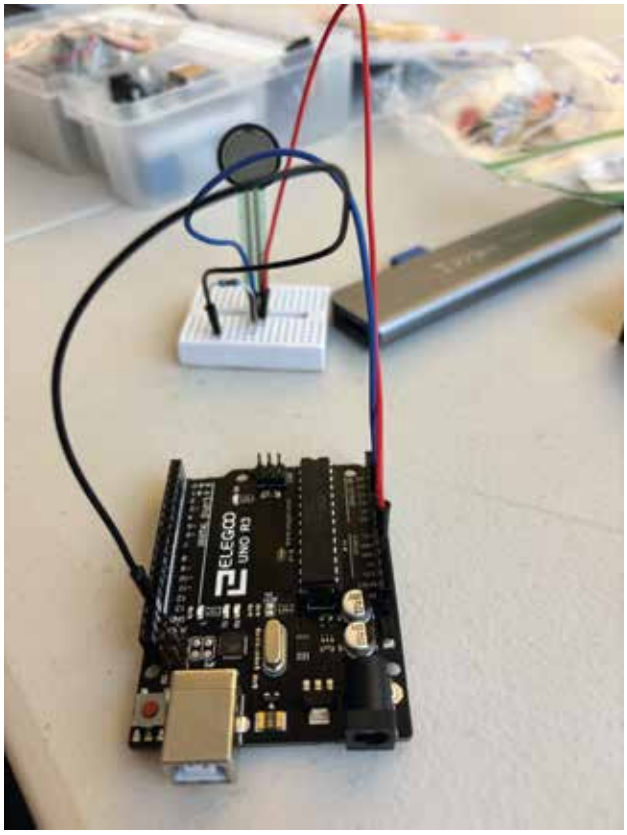
Lets Talk About the Circuit!



The circuit went through many iterations.

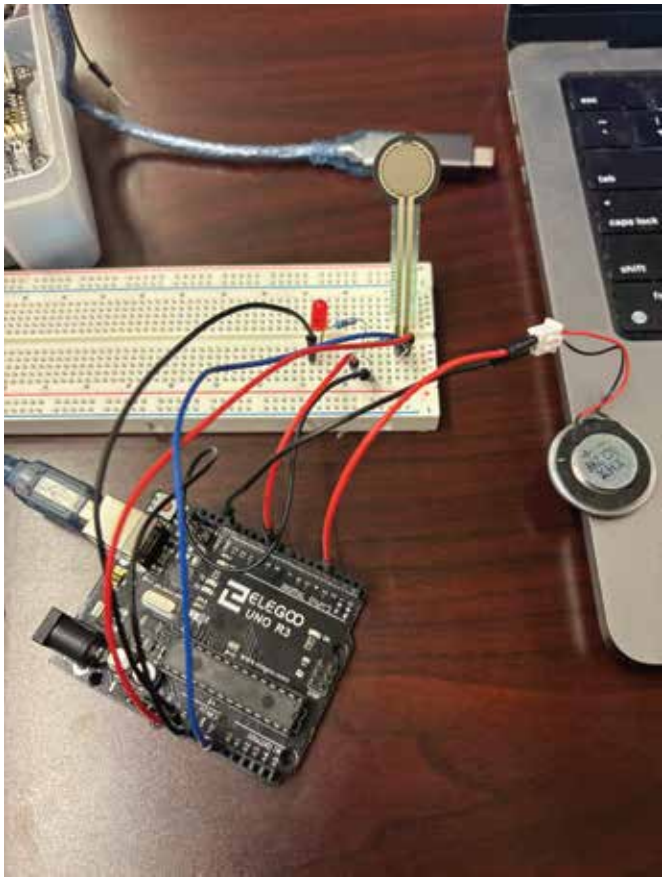


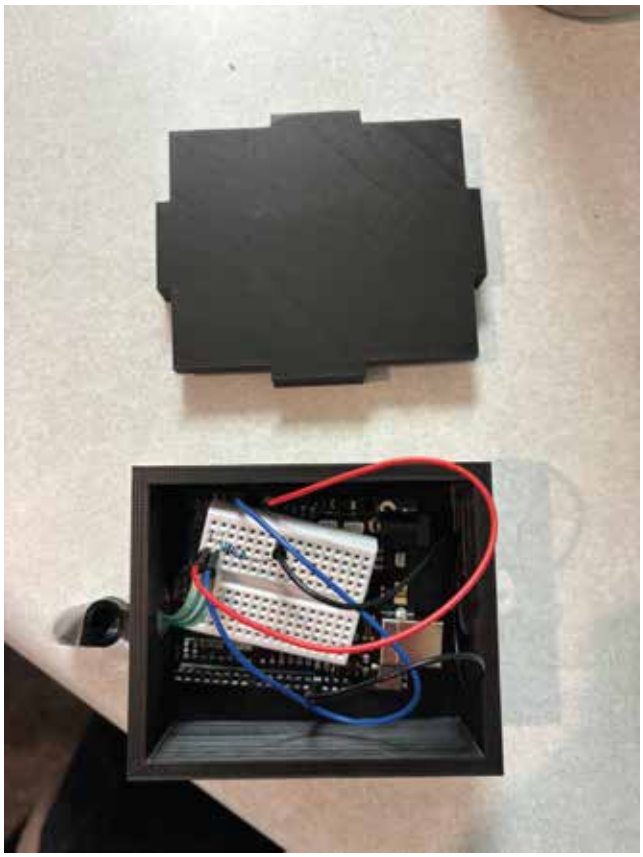




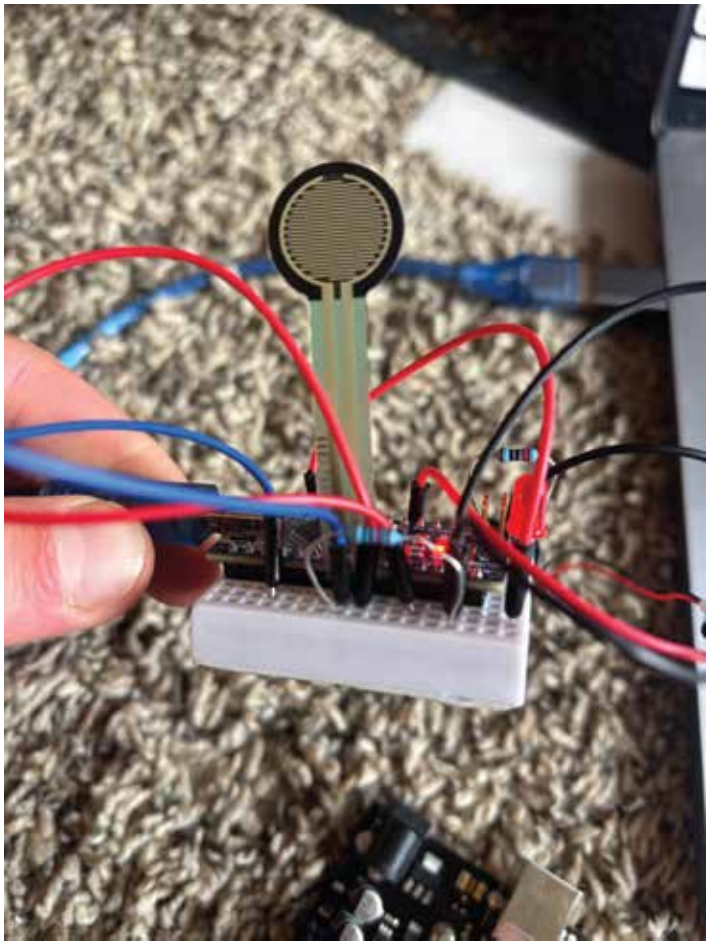
Like... lots of iterations

And the circuit started to grow

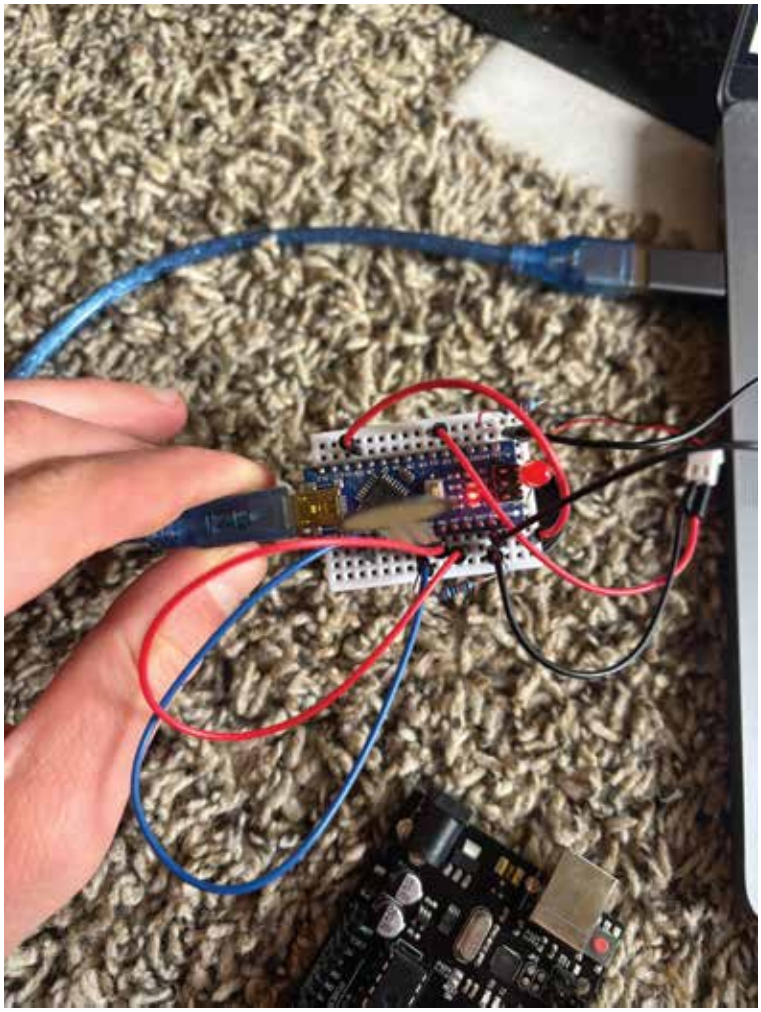


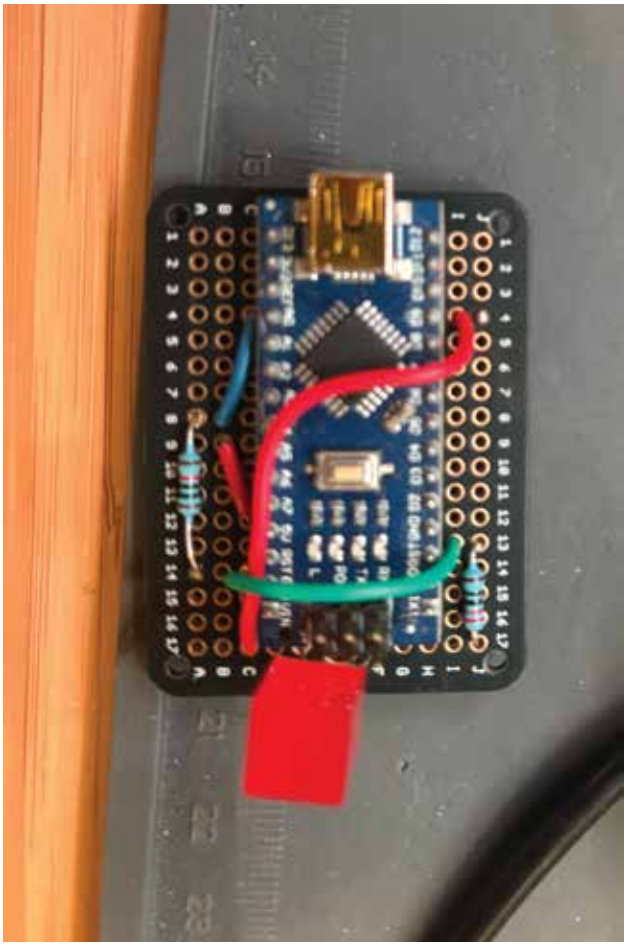


And grow

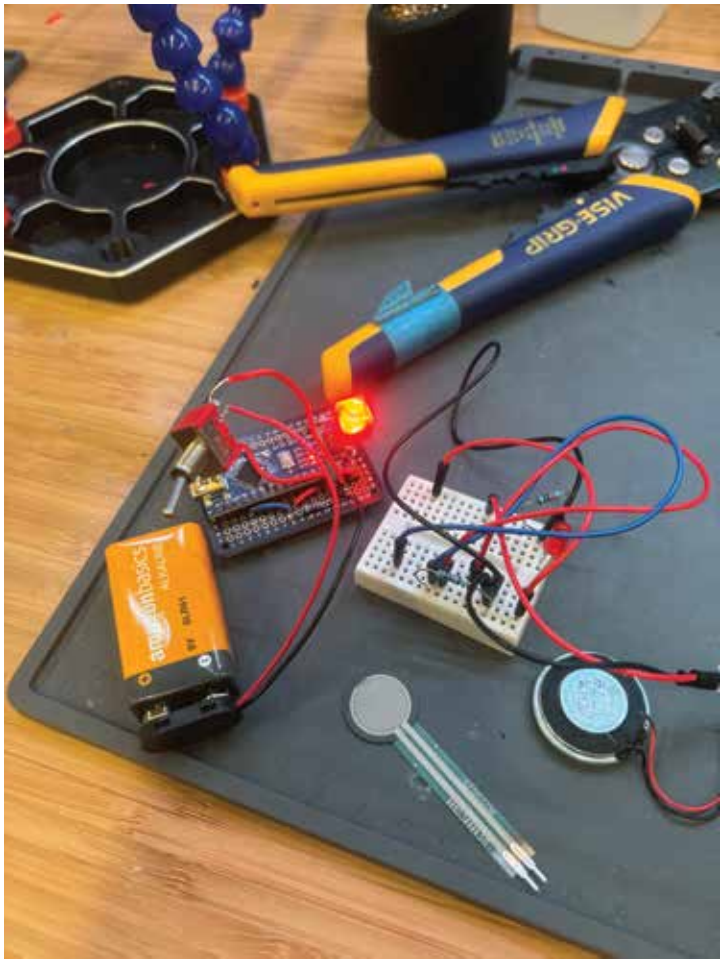


And grow

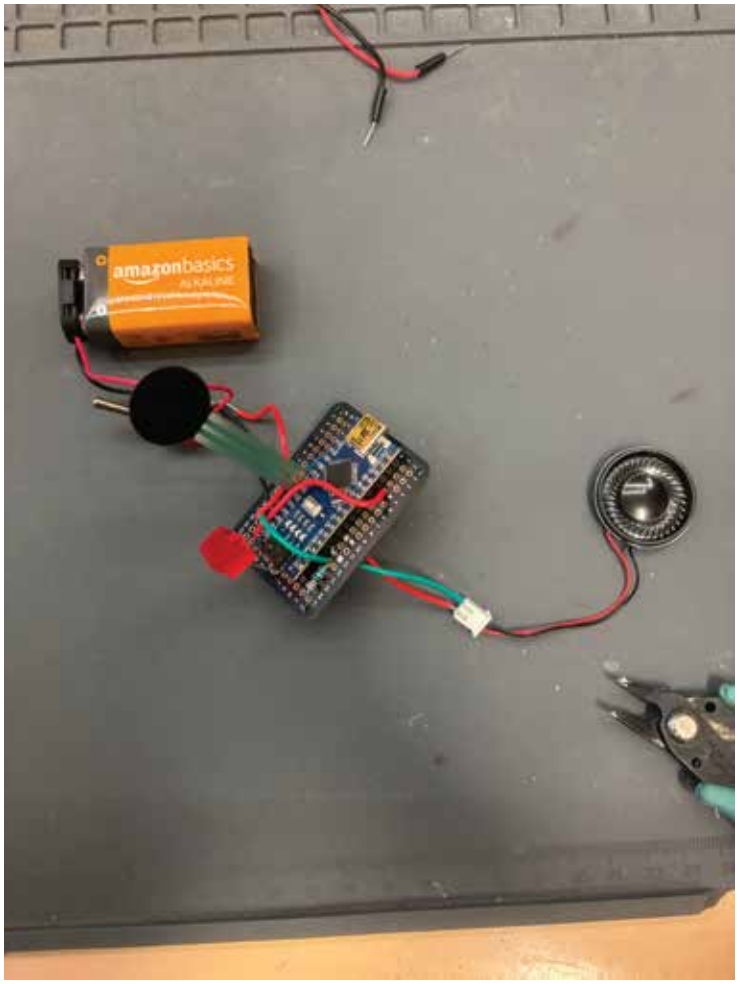




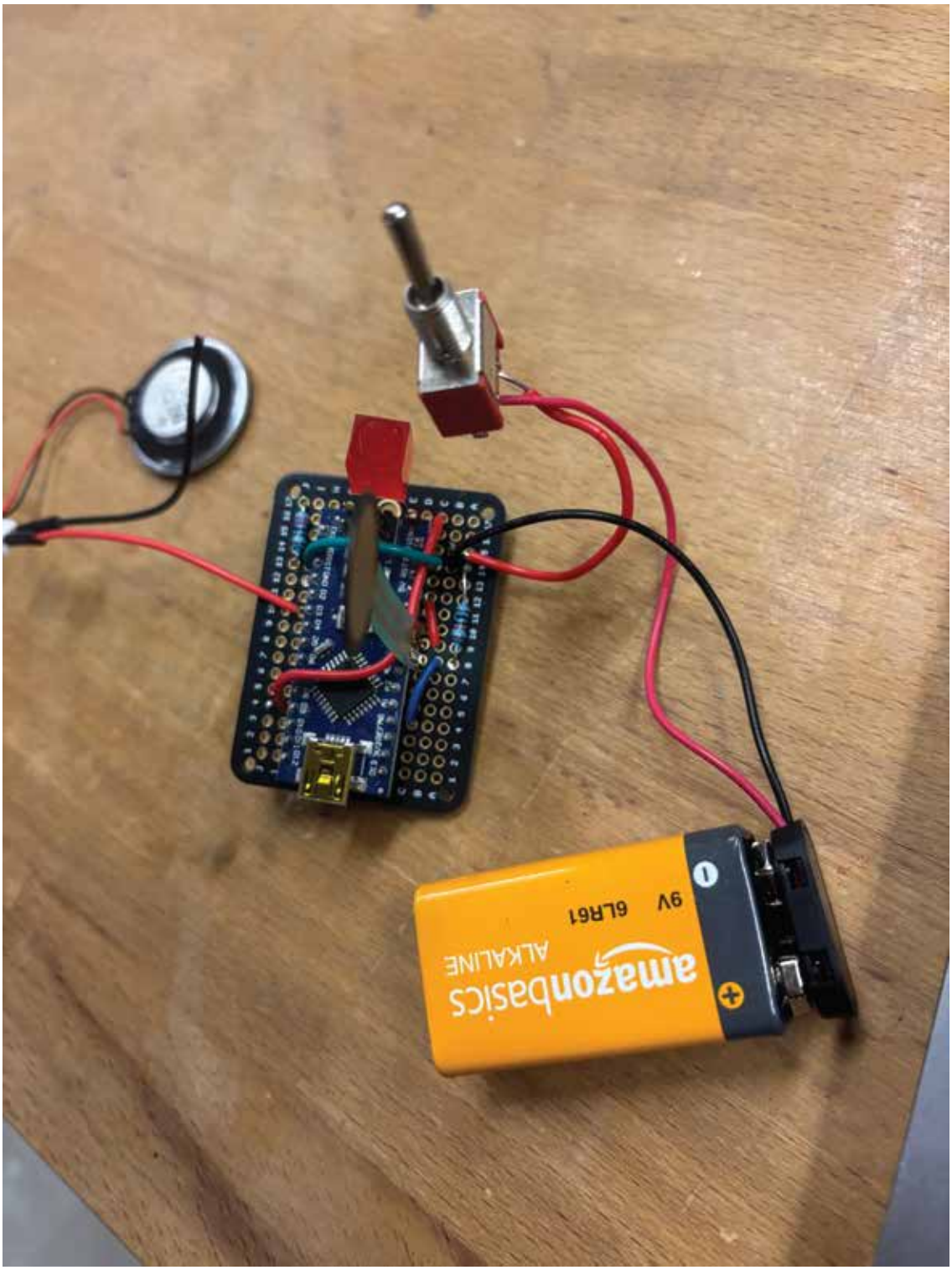
Then I tried to shrink it down as much as possible



Then it grew again...

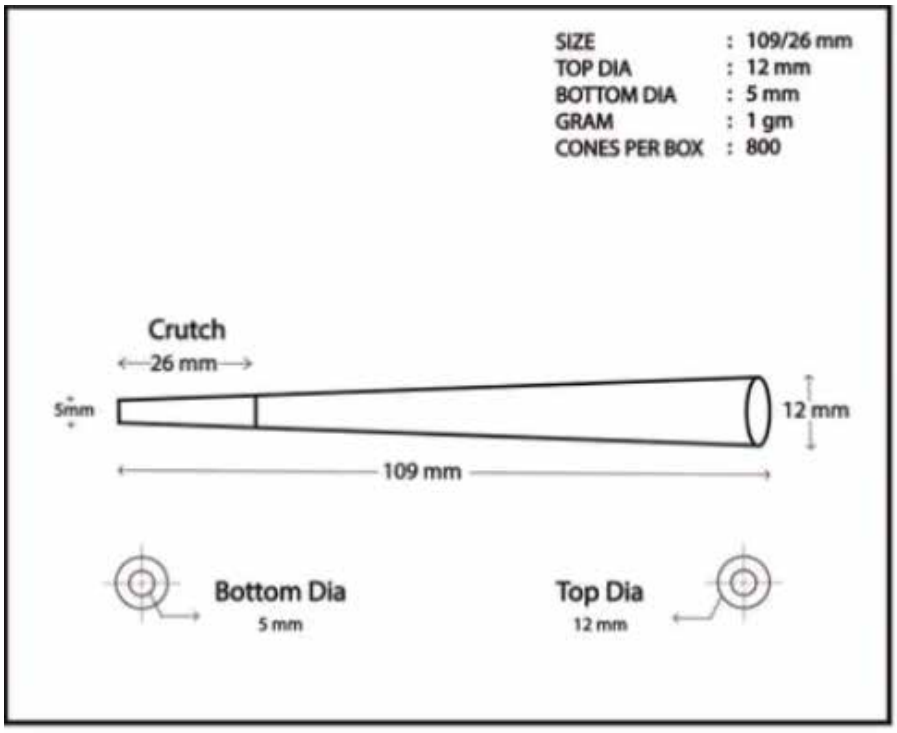


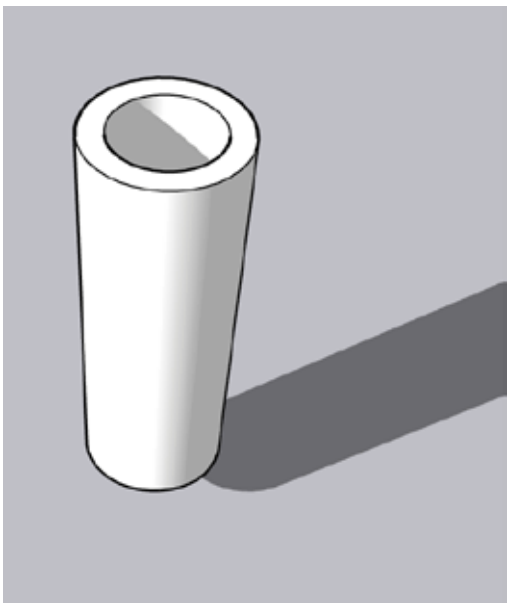
BAM! FInal Circuit Complete

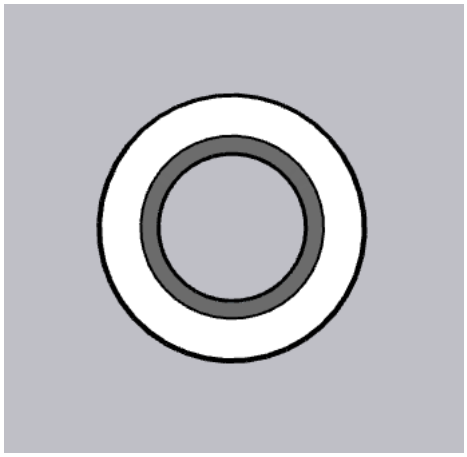


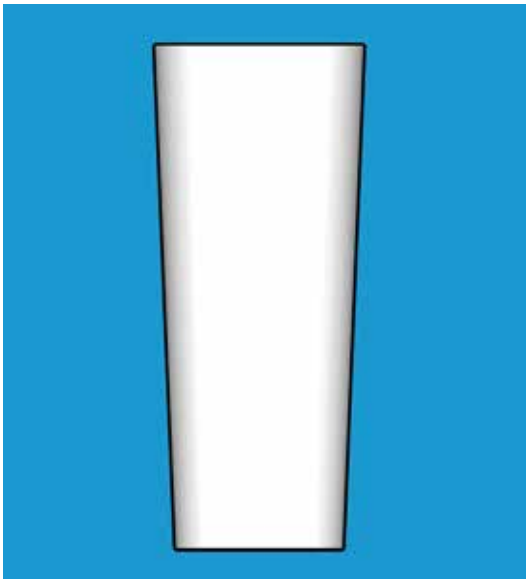
Now lets build the enclosure!

Here is the piece to hold the J

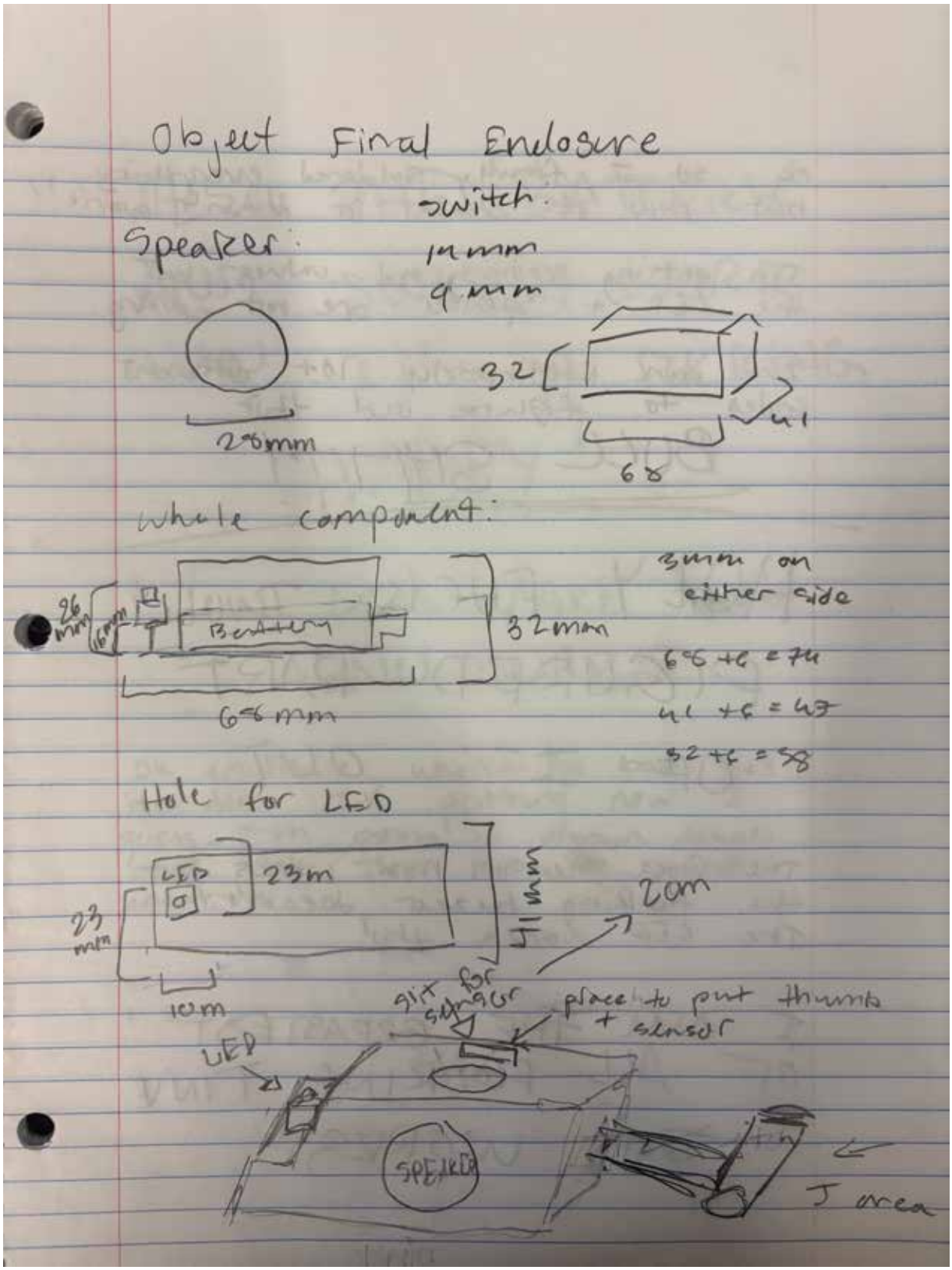




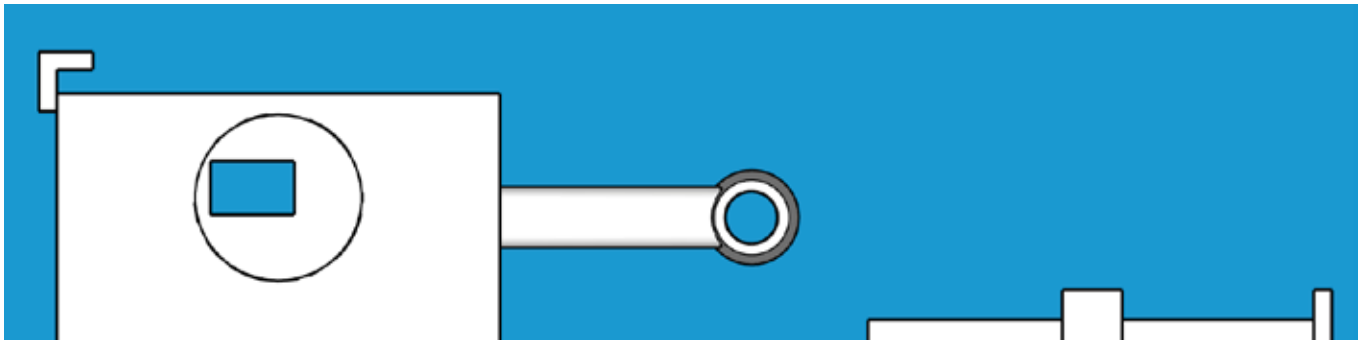


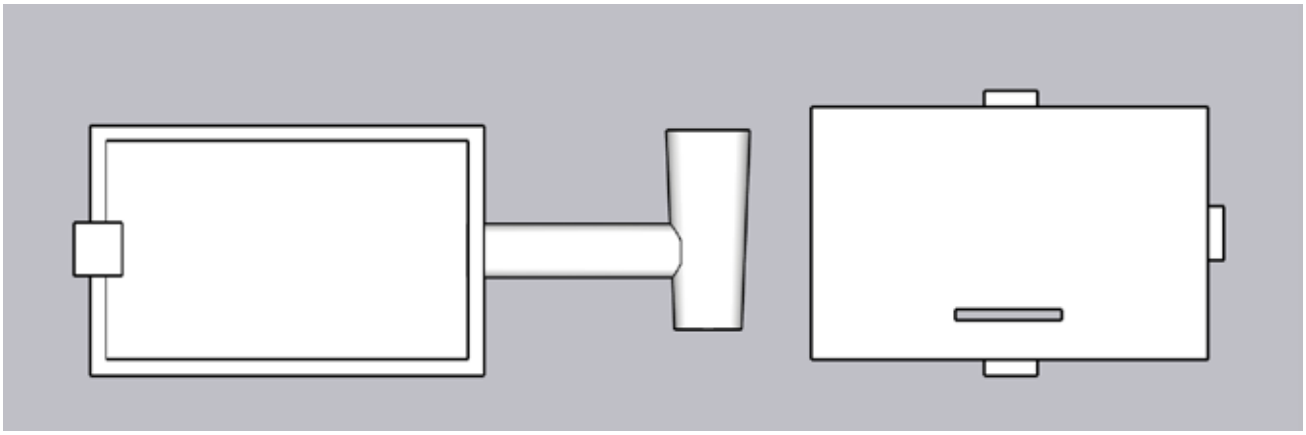


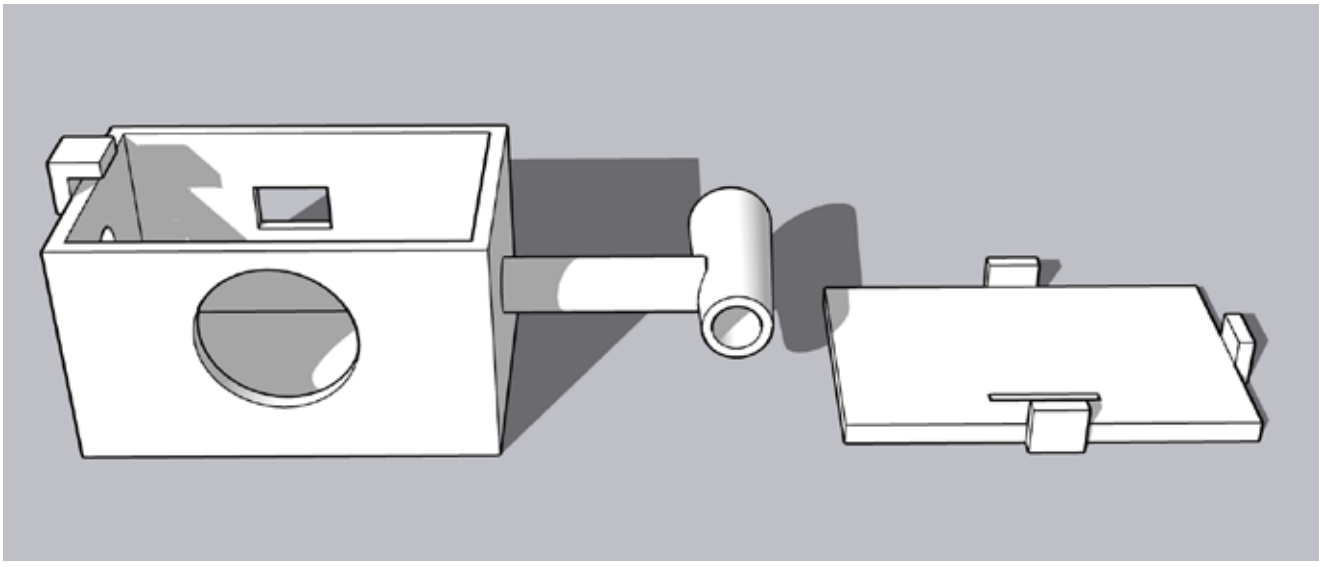
Here is the design for the rest of it.



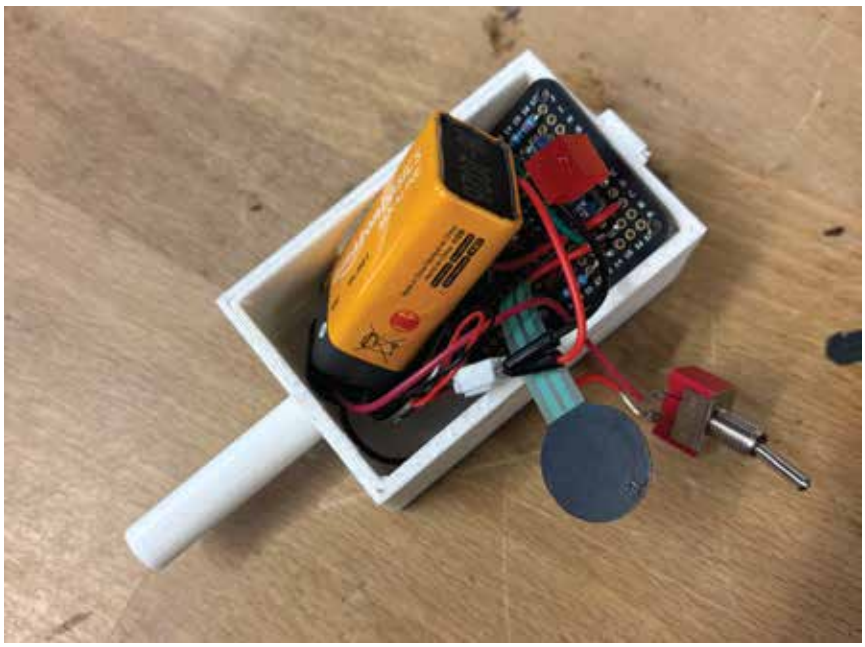
New 3D printed enclosure

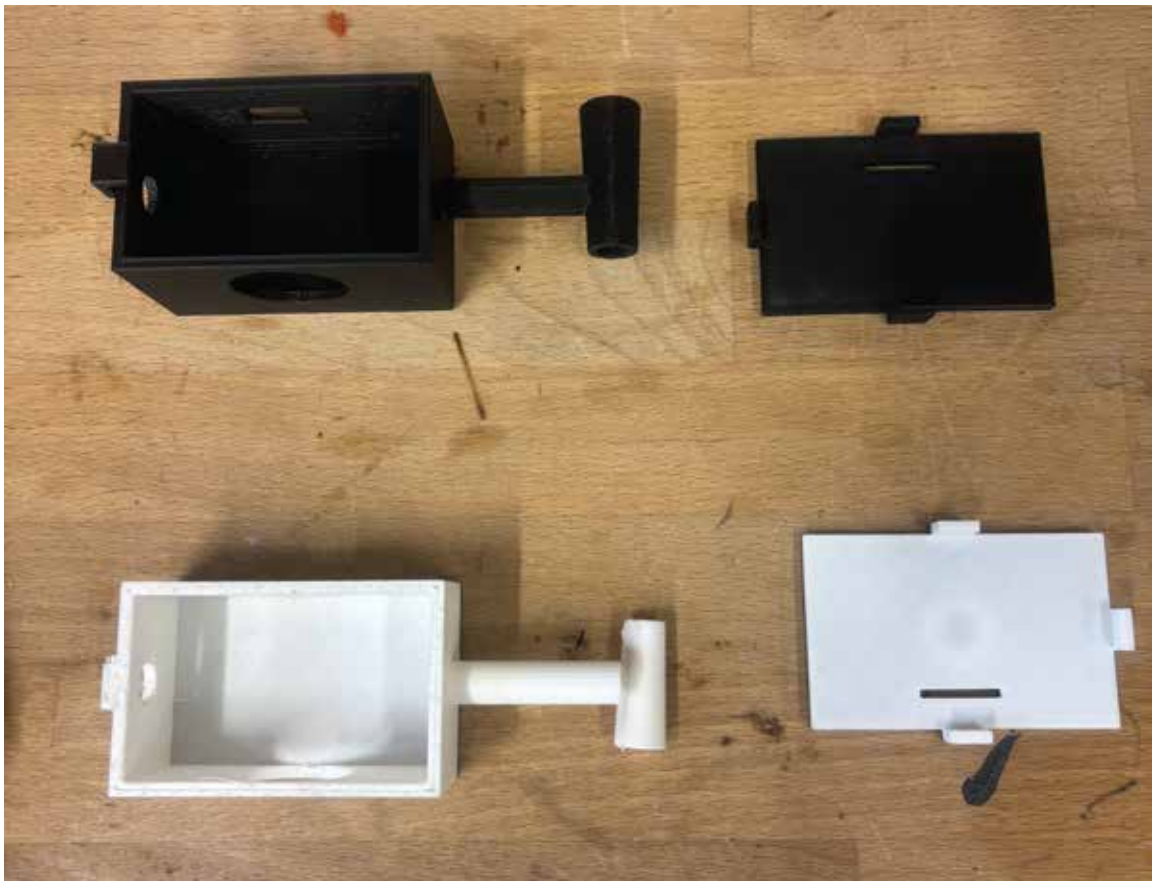






The first print ran out of fillament about half way through. The enclosure was a bit small anyways so I reprinted the enclosure just a few milimeters bigger.





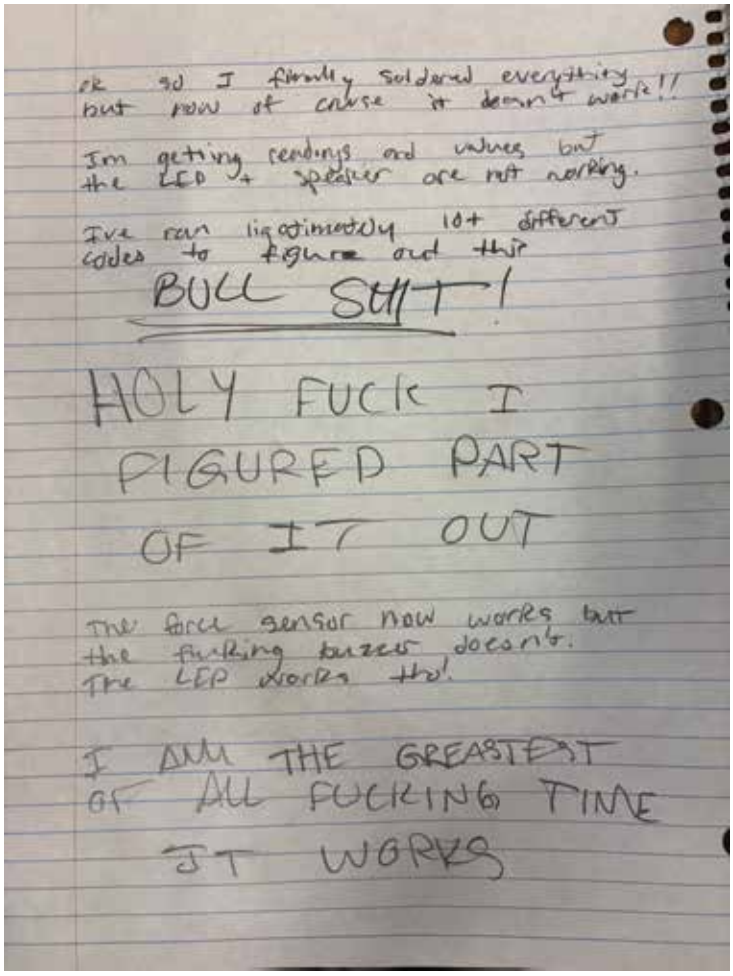
The top even snaps on and off so you can access the circuit much more easily.

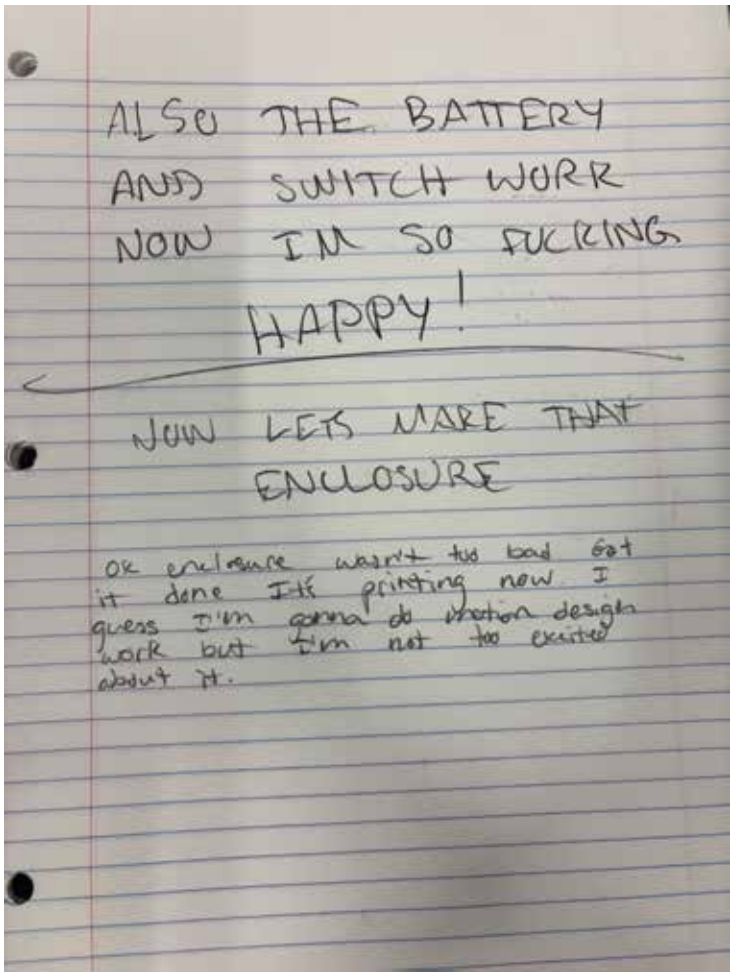
Lastly, here is my code!

In short, the code checks to see if the pressure sensor is feeling any pressure. In theory, if someone is talking and not hitting the J, there should be no pressure being applied. If no pressure is applied for 15 seconds, the code sends a signal to the LED and speaker to flash the light and play the beeps.

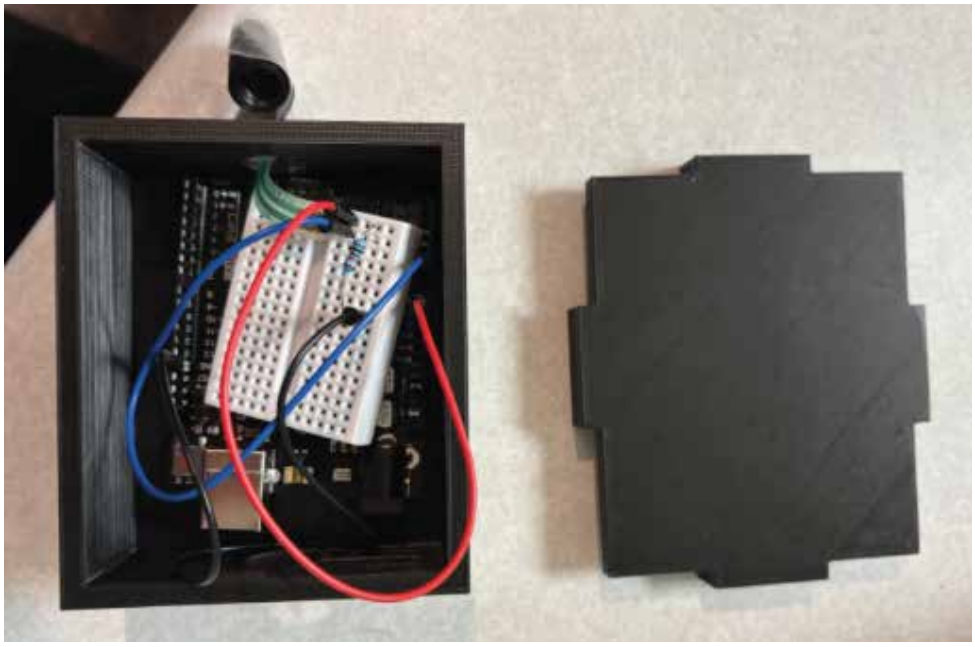
```
#define fsrpin A0
#define FORCE_THRESHOLD 0.0005  // Minimum force required to register (adjust as needed)
#define CYCLE_THRESHOLD 10      // Threshold for force (10 N)
#define MAX_CYCLES 5            // Number of cycles required before message

int cycleCount = 0;  // To keep track of cycles where force is below the threshold

const int speakerPin = 3; // Pin connected to the speaker
const int ledPin = 9;    // Pin connected to the LED (built-in LED)

unsigned long lastForceReadTime = 0; // Time when force was last read
unsigned long forceReadInterval = 3000; // 3 seconds interval for reading force

void setup() {
  Serial.begin(9600); // Start Serial Monitor
  Serial.println("FSR Sensor Reading - Force Estimation");
  Serial.println("-------------------------------------");

  pinMode(speakerPin, OUTPUT);  // Set speaker pin as output
  pinMode(ledPin, OUTPUT);      // Set LED pin as output
}

void loop() {
  // Check if 3 seconds have passed since the last force reading
  if (millis() - lastForceReadTime >= forceReadInterval) {
    lastForceReadTime = millis(); // Update the time when the force was last read

    int fsrreading = analogRead(fsrpin); // Read FSR value

    // Convert to voltage (assuming 5V system)
    float voltage = fsrreading * (5.0 / 1023.0);

    // Approximate force (for an Interlink FSR, values may vary)
    float force;
    if (voltage == 0) {
      force = 0; // No force applied
    } else {
      float resistance = (5.0 - voltage) * 10000 / voltage; // Using 10kΩ pull-down resistor
      force = pow(10, (log10(resistance) - 6) / -1.4); // Approximate force in Newtons
    }

    // Check if force is BELOW the threshold
    if (force < CYCLE_THRESHOLD) {
      cycleCount++;  // Increment cycle count if force is too low
      Serial.print("Low Force detected! Cycle: ");
      Serial.println(cycleCount);

      // If max cycles threshold is met, activate the speaker and LED
      if (cycleCount >= MAX_CYCLES) {
        Serial.println("PASS THE J!");
        playBeepAndFlash(); // Start beeping and flashing once max cycles is met
      }
    } else {
      cycleCount = 0;  // Reset cycle count if force goes above threshold
      // Stop the beeping and flashing when the force goes back above the threshold
      noTone(speakerPin);  // Stop tone
      digitalWrite(ledPin, LOW); // Turn LED off
    }

    // Print force data for debugging
    Serial.println("\n-----------------------------");
    Serial.print("Analog Reading: ");
    Serial.println(fsrreading);
    Serial.print("Estimated Force: ");
    Serial.print(force);
    Serial.println(" N");
    Serial.println("-----------------------------");
  }

  // Short delay to make the loop responsive (you can adjust this)
  delay(100);
}

// Function to play a sequence of 3 loud beeps and flash the LED 3 times
void playBeepAndFlash() {
  for (int i = 0; i < 10; i++) {
    // Play the beep
    tone(speakerPin, 1000); // Play tone at 1000 Hz
    digitalWrite(ledPin, HIGH); // Turn LED on
    delay(100);               // Delay for 0.1 seconds
    noTone(speakerPin);      // Stop the tone
    digitalWrite(ledPin, LOW); // Turn LED off
    delay(100);               // Delay for 0.1 seconds between beeps and flashes
  }
}
```

Here is some of my favorite process journaling



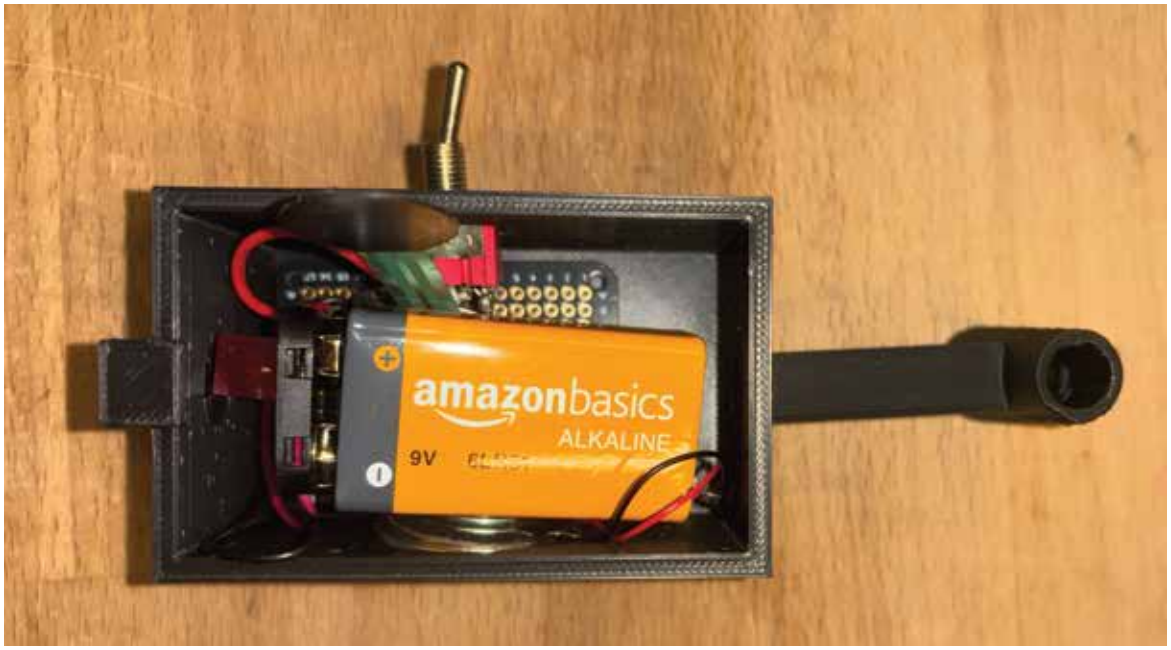


IDC 1



VS

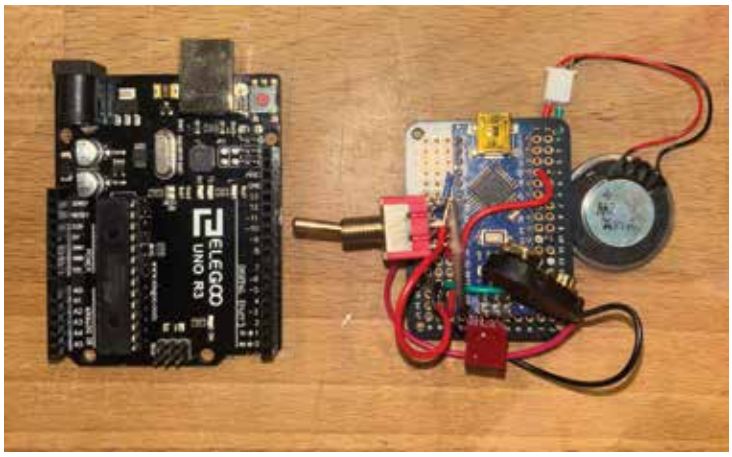
IDC 2



IDC 1 vs IDC 2

Arduino

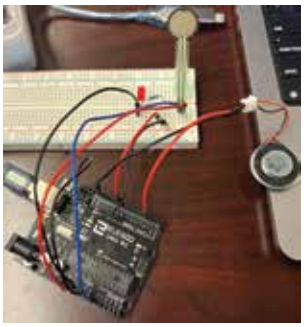
One of the biggest changes I made going from IDC 1 to IDC 2 was the computer I used. I wanted to try and make my circuit as small as possible so that it can easily fit in your hand. Therefore, I opted to switch the Arduino Uno for the Arduino Nano.



The arduino uno is on the left and the arduino nano is the small blue piece in the middle on the right. Clearly this helped to size my circuit down drasticly.

Durability

The durability of my project from IDC 1 to IDC 2 had greatly improved. In IDC 1 my project ran off of a bread board and all the wires were very loosely held into place. Also, the top of my enclosure would fall off if turned upside down. In IDC 2 the entire circuit is soddered in nicely and the circuit doesnt move. Also, the enclosure is built so precisely that the pieces inside of it don't move around.



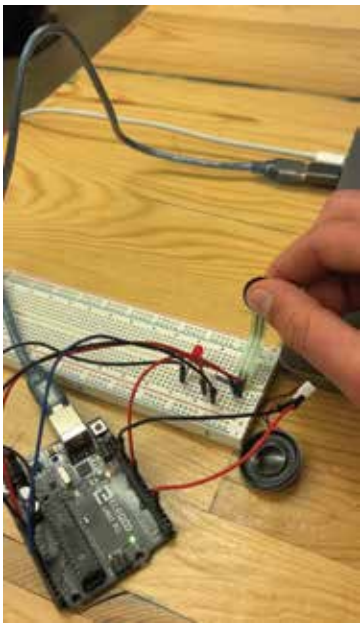
Functionality

The most noateable change I made between IDC 1 and IDC 2 was the functionality of my device.

I had two major design flaws in my IDC 1. The first was my J holder and pressure sensor. The second was the power source.

The J holder in IDC 1 was only connect to the pressure sensor by tape and the J holder was a loose 3D printed piece.

In IDC 2 the pressure sensor sits nicely in an indent designed to fit around the thumb very comfortably.



The power source in IDC 1 was unreliable as well because you needed to thread the USB cable from your computer into the enclosure and into the arduino for the circuit to run.

In IDC 2 the circuit has a switch and a 9V battery that sits snuggly inside the new enclosure.

Many changes happened between IDC 1 and IDC 2 for example but not limited to:

Change in wiring
Change in code
Change in LED light
Change in enclosure
Arduino nano instead of Arduino uno
Added Circuit Board
Added On/Off swtich
Added 9V battery
Soddered Wires
Shortened wires

I tested a plethora of other ideas and iterations for the Pass the J Device. For instance, I wanted the speaker to say "Yo! Pass the J" but to do this I would have needed to build an amp which would have made my curcuit too big for what I want.

The biggest challenge for me during the entire process of going from IDC 1 to IDC 2 was the soddering. My wands were so shaky and sweaty the whole time because I didn't want to mess up. However, by far the worst part of soddering was when you sodder something and it doesn't work. Or worse even than that, when you sodder something and the whole circuit just stops working. I spent hours banging my head in my hands and drowning in missery getting the circuit to function properly after soddering. Many late nights and early mornings spent alone in the BTU getting this curcuit done. I am so glad it works.

I am so incredibly proud and excited to say that my machine works perfectly! It took so many hours, fails, changes, and frustrations but it finally works.

I learned so much especially from the changes I made from IDC 1 to 2. I totally forgot about needing a power source and last second as I was starting to sodder my circuit I rembered that I needed one. This threw me through such a loop and I had to completely redesign my circuit on the fly. I am so glad that I was able to do it. I am so glad that my machine no longer needs to be connected to a computer and that it is completely soddered in and transportable.

Also, I had the J holder part of my IDC 2 enclosure printed 90 degrees to the right on accident so I had to use an x-acto blade and a glue gun to orient it correctly.

I have lived at the BTU trying to get this done and it feels like a million pounds have been lifted off my shoulders.

Here is the link to watch the video of it working!

https://drive.google.com/file/d/16s5kYNIWxu43XlWj
Py2j94ezsXZ9Khcq/view?usp=sharing