Servo Controlled Robotic Hand with Flex Sensing

Abhi Agrahari and Eric Schilla

June 21

Abstract

The main goal was to create an animatronic controller for a robotic hand. The user is to wear a glove, that has flex sensors attached (similar to the Nintendo Power Glove). These sensors' values are processed by the board, and sent to the respective servos in the robotic hand. If the user bends their hand slightly, the robotic hand bends slightly. Thus, the robotic hand mimics the movement of the user's hand.

Table of Contents

Abstract

Table of Contents

Introduction

Hardware Design
- Prototype Versioning of a Finger
- Palm and Arm Design
- Flex Sensors

Software
- Servo Timing
- Github and MPLAB

Summary
- Future Improvements

References

Appendix
- GitHub Links:
- Graphite-based Flex Sensor
- Technical Drawings of the Hand
  - Basic Design
  - Fishing Line Paths
  - Screw Hole and Servo Dimensions
- Versions of the Index Finger
  - Original Design
  - Version 1 - Fundamental Design Flaw
  - Version 2 and 3 - Joints are too loose
  - Version 4 - Final Version, A Success

Introduction

Any intermediate programmer that is experienced with hardware, like ourselves, would be able to understand this report. The general technologies involved are servos, and flex sensors.

For our summative, we wanted to create something that combined mechanical/hardware aspects, and programming/software aspects. On YouTube, we had seen videos of creating drones, and others of robotic body parts. After making a drone, we would feel like we have achieved quite a lot as we would have made something that can fly. However, the cost of making a drone would be much greater than buying one, and the store bought one would be better calibrated and suitable for flying. So there really was no point to that. When deciding on a robotic body part, we narrowed down to a couple. We chose to go with a hand because we could print and test many parts to create the needed joints, but also because we could find a way to make it mimic our movements. So overall, design and programming a robotic hand seemed like a good challenge for us.

This report intends to cover the design process behind both the mechanical aspects of the robotic hand, the self-made flex sensors, and the code created for this project.

Hardware Design

For this project, we designed all hardware related components besides the servos, and the main circuit board. Essentially, we designed the flex sensors that are used, and the 3D printed hand. The 3D printed hand's joints are controlled by fishing line, and are attached to 5x 9 gram micro servos.

Prototype Versioning of a Finger

A long amount of time was spent in understanding and resolving the design to be able to have a finger that moves by pulling on two strings. Images of both prototype versions, and final versions can be seen in the appendix.

The versions progressed from not being able to move a finger at all due to a fundamental design flaw, to being able to fully bend all joints. A main difficulty was creating these joints to properly fit into one another. In the end, once a working version was made, we left the joints detachable instead of hooking onto one another as it is easier to create and assemble (see appendix for design).

Palm and Arm Design

Besides having 5 similar fingers, we needed to also create a palm and an arm that the fingers would attach to. The palm was designed such that each finger's fishing line had a unique path through the inside of the palm. This ensured that the lines did not cross over or tangle in each other. The arm's servo supports were then aligned with the holes in the base of the palm so that enough torque could be generated by the 9 gram servos.

Flex Sensors

Flex sensors are a special type of variable resistor. Their resistance changes as they bend. One side of the sensor is printed with a polymer ink that has conductive particles embedded in it. When the sensor is straight, the particles give the ink a low resistance, and when the sensor is bent away from the ink, the conductive particles move further apart, increasing this resistance. Thus, by combining the flex sensor with a static resistor we could create a voltage divider.

Or at least that was our idea. Originally, a plan was made to purchase 5-7 flex sensors from an online site, and use them. Unfortunately, flex sensors cost much too much (around $7) per sensor, so having just 5 sensors would have cost us around $35.

Various designs were found online to make these flex sensors. The designs had a general theme of using conductive foam, graphite covered paper, conductive thread and neoprene, and velostat. Purchasing conductive foam or velostat would have been very troublesome as no seller ships to Canada. So, many attempts were made to make flex sensors out of paper.

The design was sandwiching a piece of paper coloured with graphite, between two pieces of aluminum-taped cardboard. As expected, these did not work very well. Each sensor had large variations in the resistance it gave (~5kΩ), and each sensor by itself did not give consistent results.

Our next design was to sandwich conductive foam instead of paper. This worked much better, and all sensors give a reading of unbent → above 2.2kΩ and fully bent → below 300Ω. We then changed the sandwiching material to strips of yogurt boxes (plastic) instead of cardboard, so that the foam is bent more evenly, allowing greater sensitivity.

To create a voltage divider in this scenario that would give the maximum range for an input signal, we needed a static resistor of the average of 300Ω and 2.2kΩ. This would be a 1.25kΩ resistor. Fortunately, we had access to a 1.37kΩ resistor, which is close enough for our purposes.

Since we used the CHRPMini board, we created the voltage dividers by connecting the 1.37kΩ resistor to each PORTA 5V and PORTA signal pins. The flex sensor was then connected to the signal and the ground pin.

Software

Servo Timing

One thing we learned, is that each type of servo has a different timing. For example, the specifications sheet states that a 1ms pulse is -90°, and a 2ms pulse is +90°. However, after some trial and error, it was found that a 540μs pulse is -90°, and a 2.2ms pulse is +90°.

It was pointed out that Servo code isn't necessarily exact, and can be fudged around a bit. However, we wanted to use the closest possible values. As such, in our code, we do some math to find the value to delay by so that the servos are pulsed every 20ms (see *Hand.C* in the *CHRPMini-Hand* repository on Github).

Github and MPLAB

Getting github to work with MPLAB was difficult at some stages. We were not aware of how to properly integrate the two. Links to our githubs are available in the appendix.

Summary

Overall, this project was very successful. The 9 gram micro servos were able to create enough torque and pull the fingers closed, and open them again, continuously. The conductive foam flex sensors also had enough variation in the resistance that the servos were able to move to different positions depending on the extremity of the bend. It is astounding that one is able to make flex sensors that cost at least $8, out of some conductive foam that is lying around from other products.

Some ideas this hand could be scaled to create are prosthetics, or a robotic extension for hazardous situations where having a human is not ideal.

Future Improvements

A recommendation for the future is to buy flex sensors instead of making them ourselves. This would allow us to get more reliable and consistent results. Although making our own sensors using conductive foam was undoubtedly cool, buying them would also bring a greater level of professionalism.

Another future improvement is one that is software related. Currently the thumb is known as "zero" and the pinkie as "four". We could write code to map finger numbers to more natural values (e.g., Thumb is "one" and Pinkie is "five"). Another code improvement would be to calibrate the flex sensors. This is currently a work in progress. The best way to do it would be to have a 10 second startup period where the user flexes the sensors repeatedly. The program would then determine the lowest resistance/value that each individual flex sensor has. That value would then be mapped to position 255 on the servo. This way, each flex sensor is able to fully bend it's respective robotic hand piece, regardless of the slight difference in sensors.

References

Below are websites and pdfs that we referenced while creating this project.

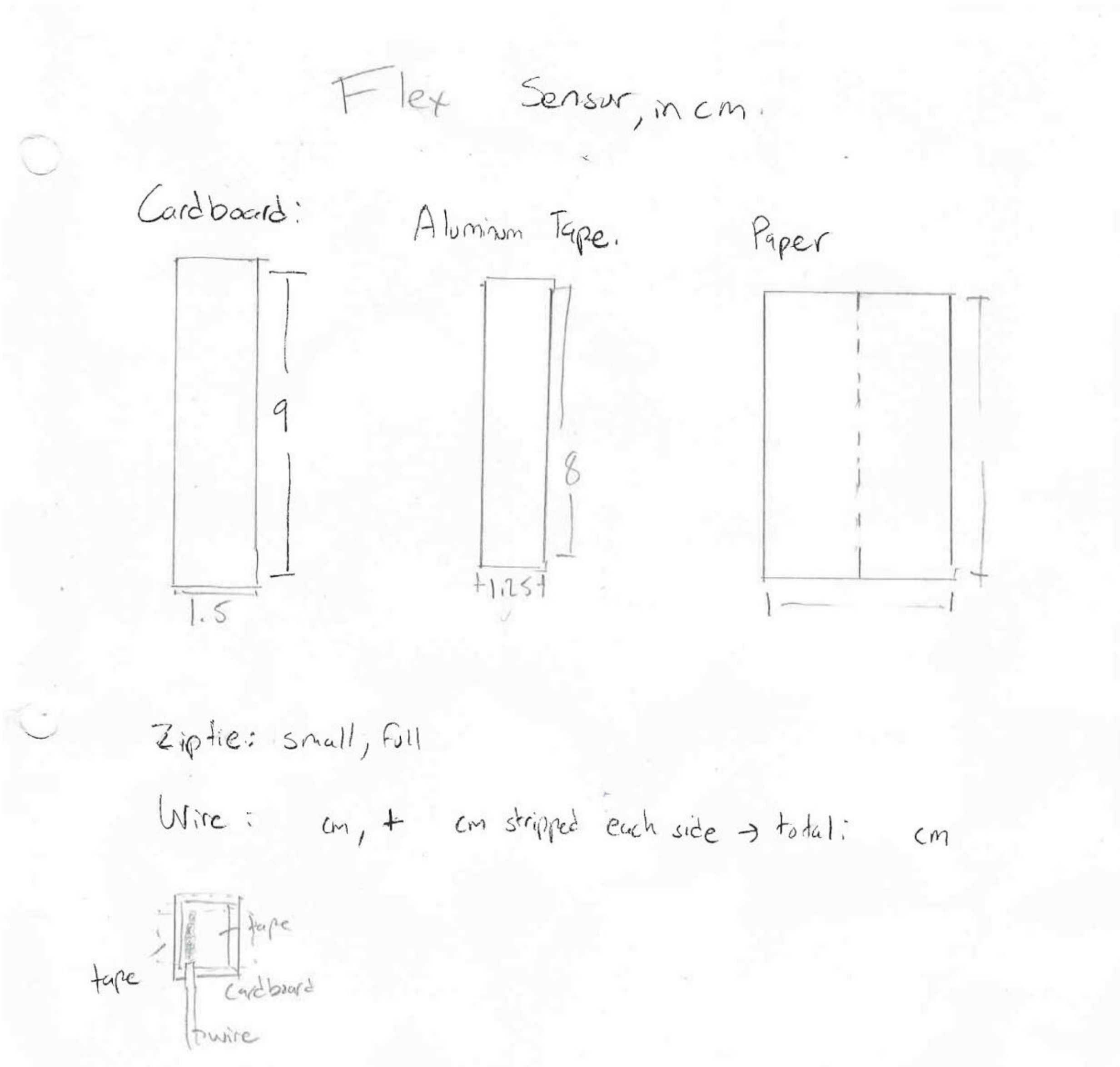
[PIC18F25K50 Datasheet]
[SG90 9gram MicroServo Datasheet]
[Learning Fusion 360]
[About Flex sensors]
[Paper Flex Sensors]
[Determining Size of array in C]

Appendix

GitHub Links:

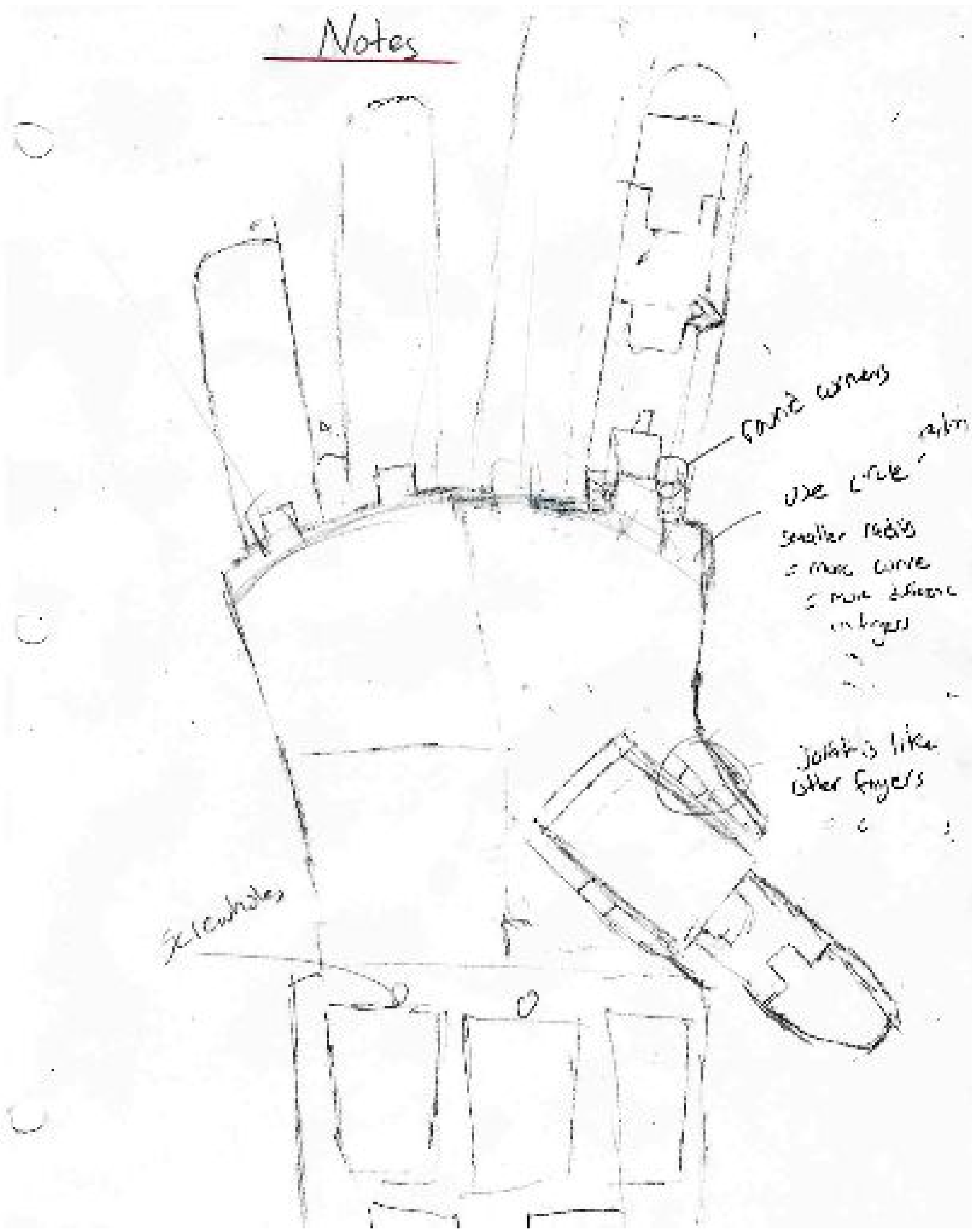
https://github.com/AbhinavA10
https://github.com/EricSchilha

Graphite-based Flex Sensor

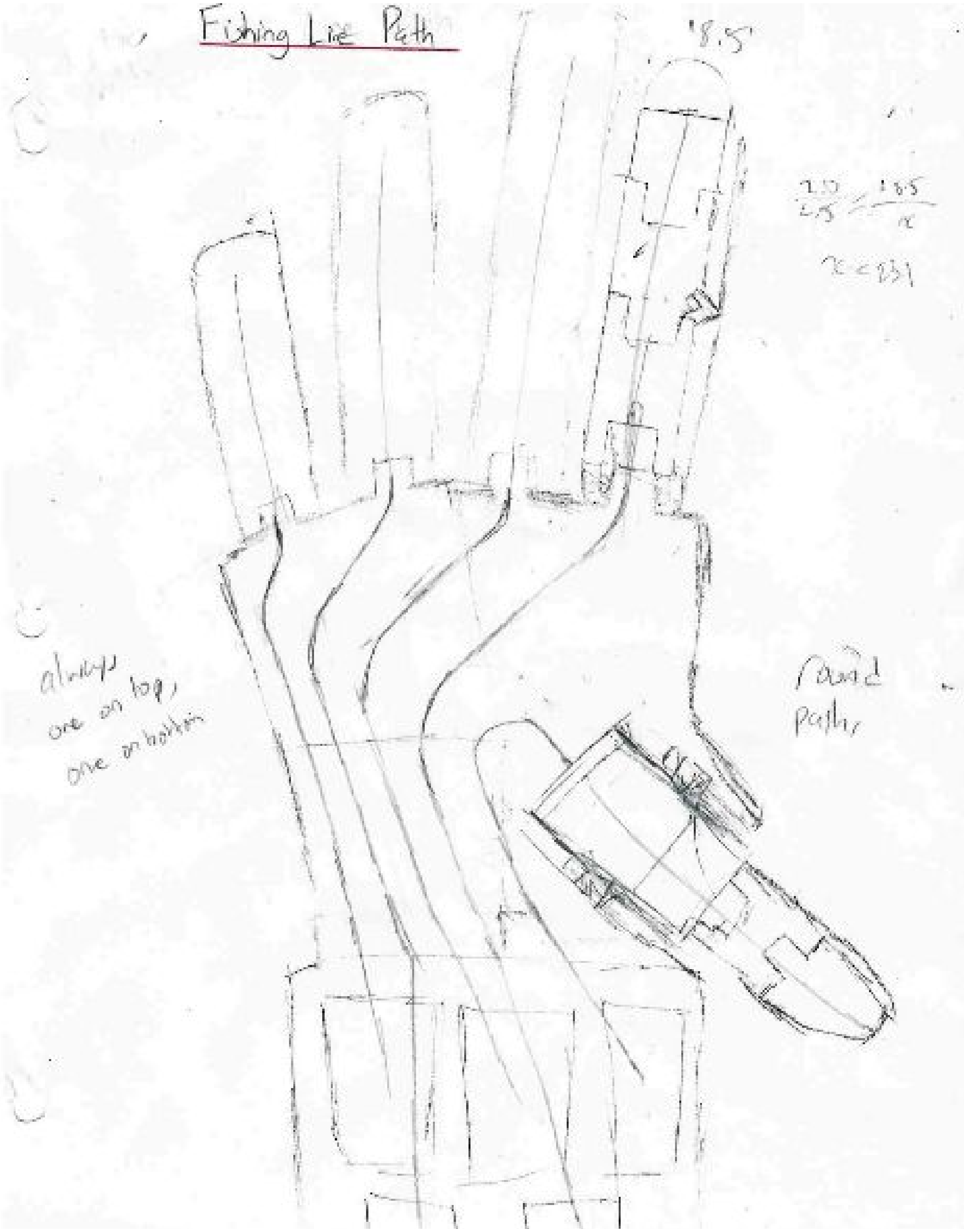


Technical Drawings of the Hand

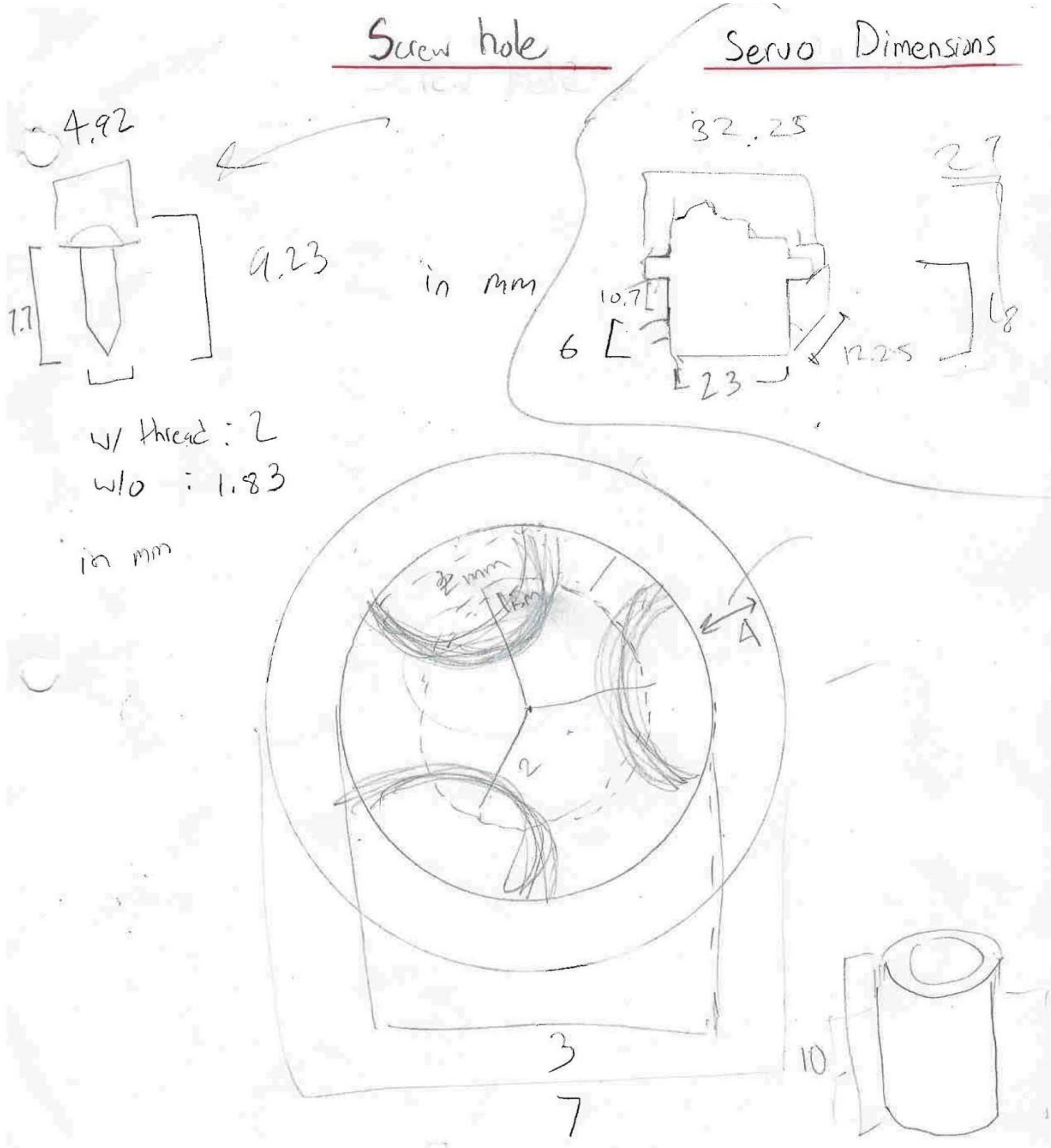
Basic Design



Fishing Line Paths



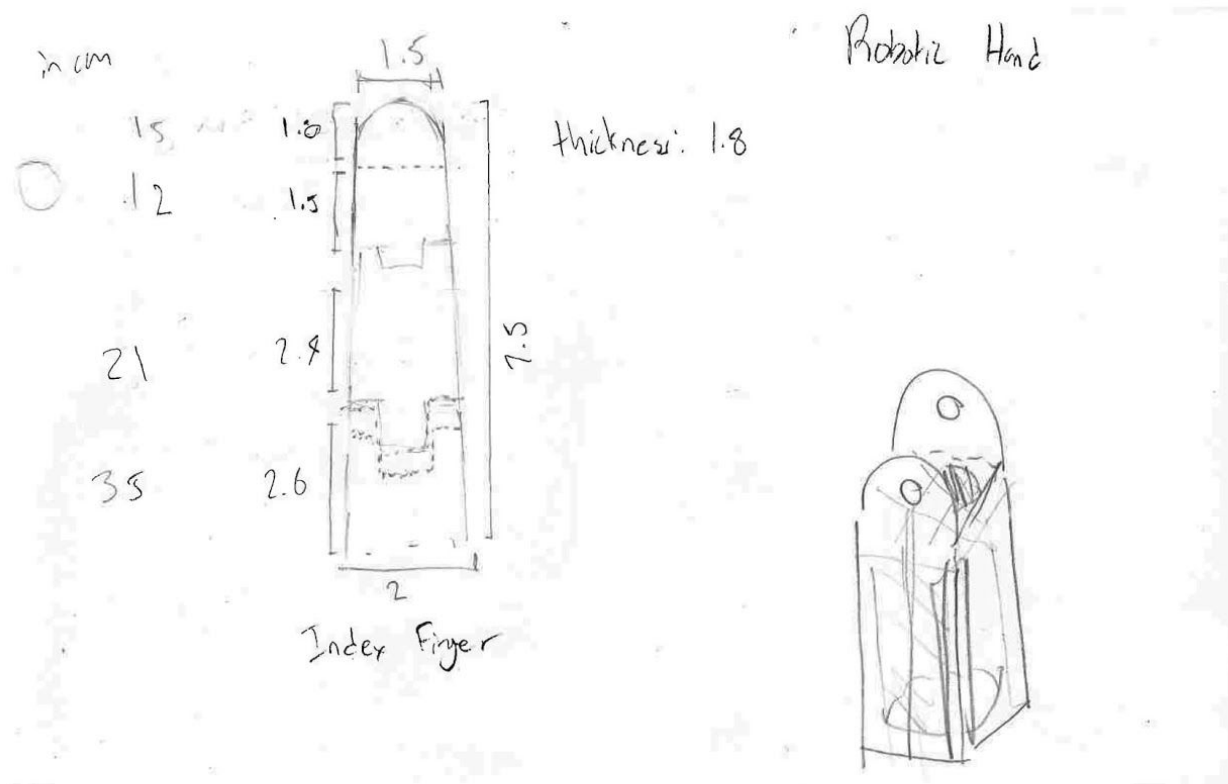
Screw Hole and Servo Dimensions



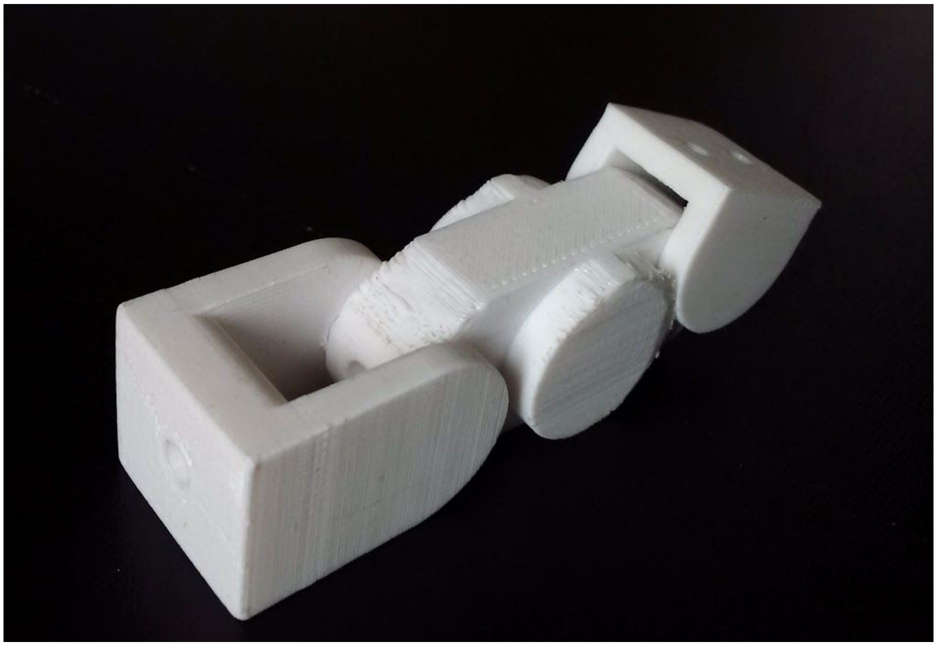
Note: need to make bigger and a hole down so screw sits in straight.

Versions of the Index Finger

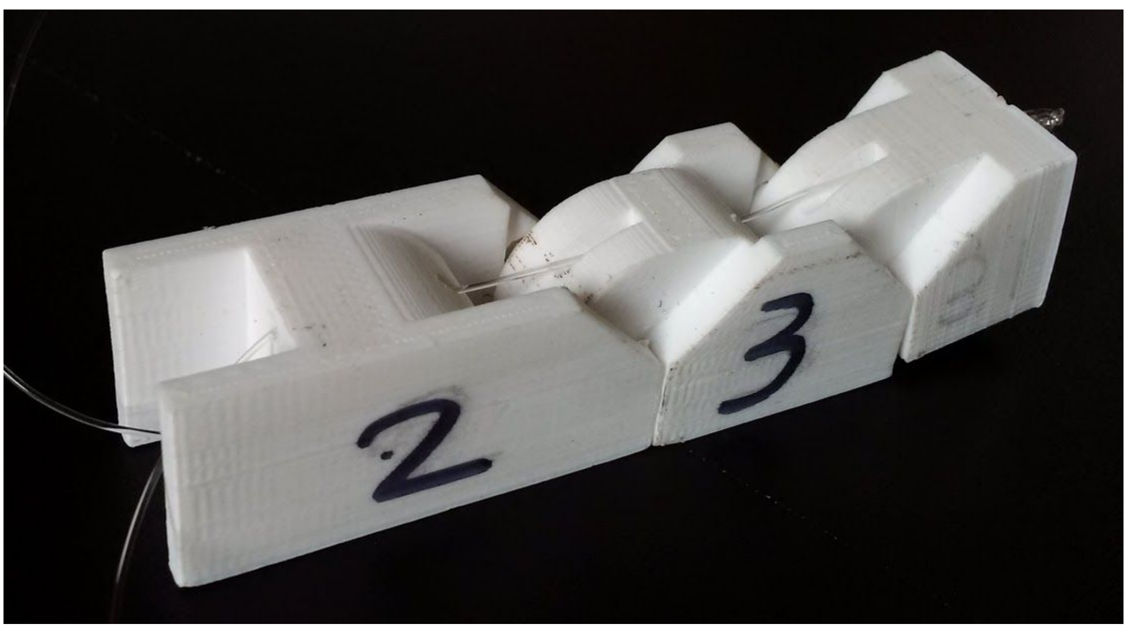
Original Design



Version 1 - Fundamental Design Flaw



Version 2 and 3 - Joints are too loose



Version 4 - Final Version, A Success