Project Deliverable H: Prototype III and Customer Feedback

Wade Stanley, Jacob Thom, Samuel Cadotte, Ethan Zhang, Ramya Patel

March 23rd, 2025

University of Ottawa

Table of Contents

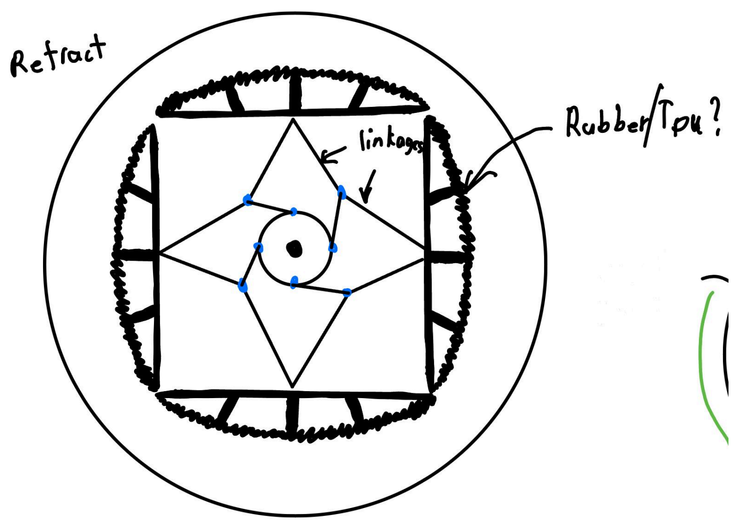
1. Feedback Received on Design

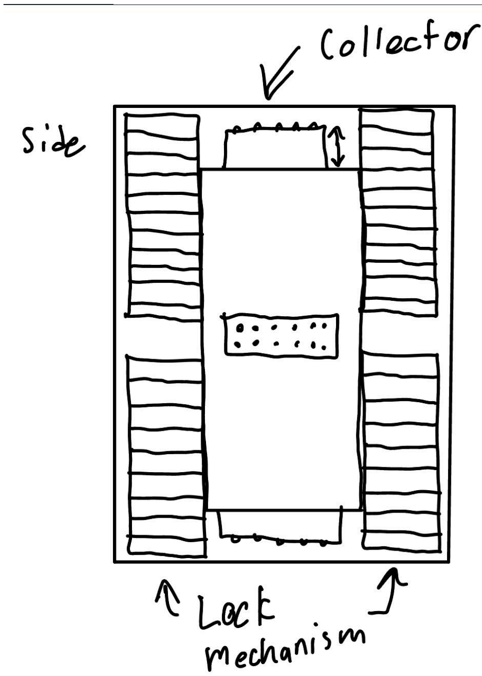
- Recommended to add the sand paper to the scraping mechanism so that we can better demonstrate the design since we had initially decided to not add the sand paper to the scraper to prevent damage to the pipe in testing.
- Recommended to add a rubber-like material to the lock mechanism so that the material of the tube will not affect how well the system locks in place. We were able to implement a quick version of this which you will read more about in the testing where we added some flex tape (thick rubber adhesive) to the pushing mechanisms to improve the tolerance and grip.

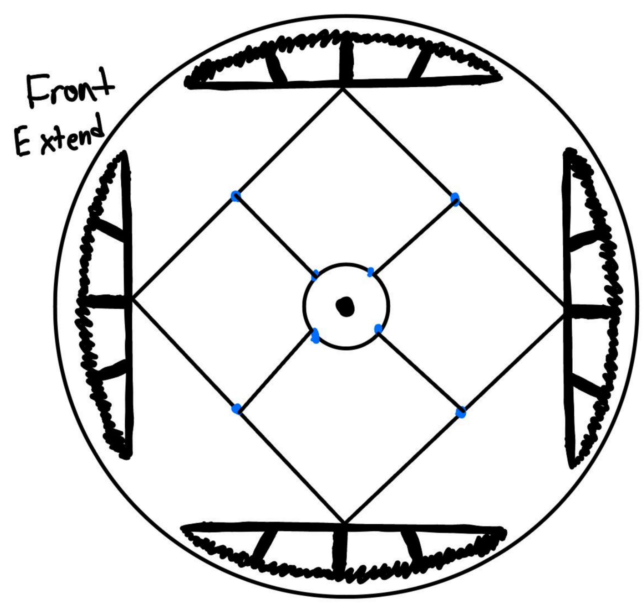
2. New Prototype

a. Updated “Rough” Design Drawings

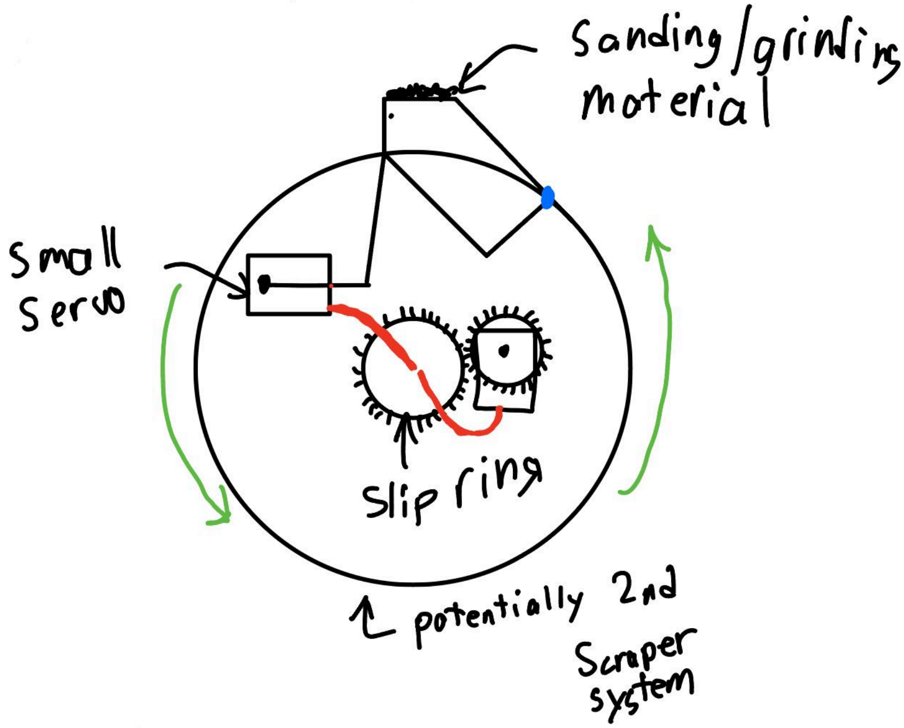
Locking Mechanism:



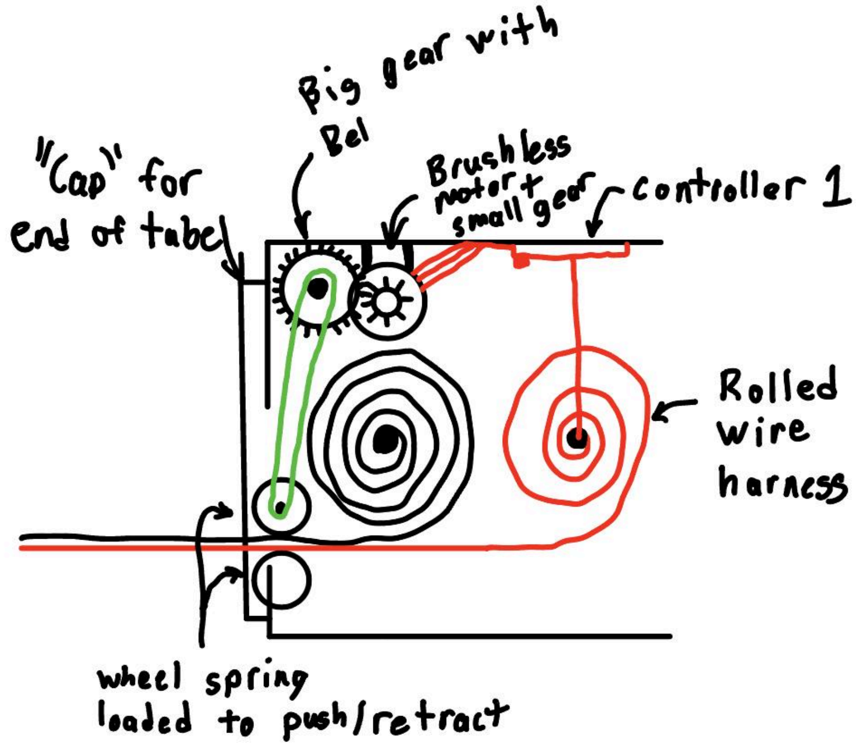




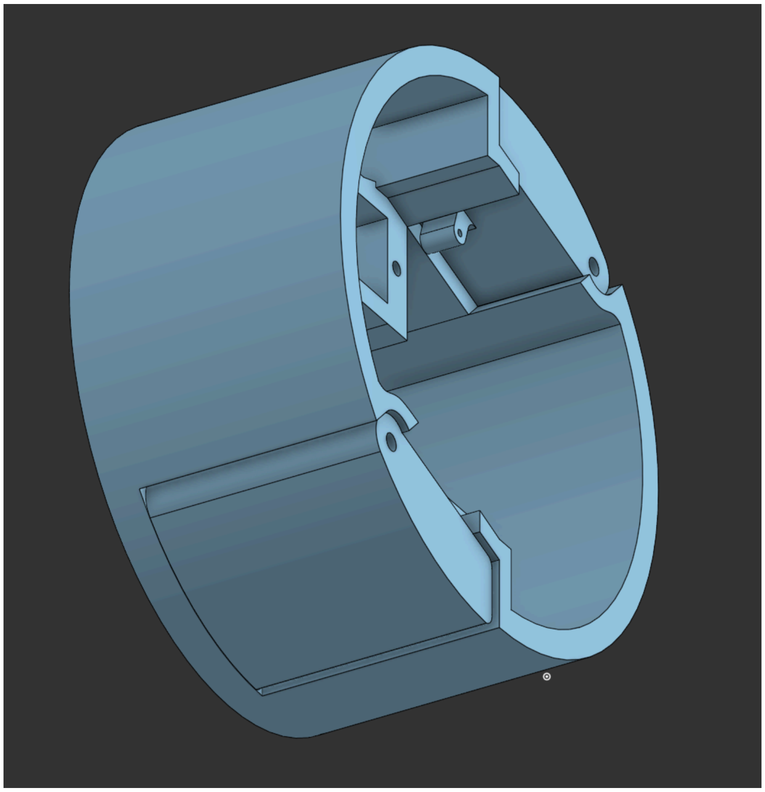
Scraping Mechanism:

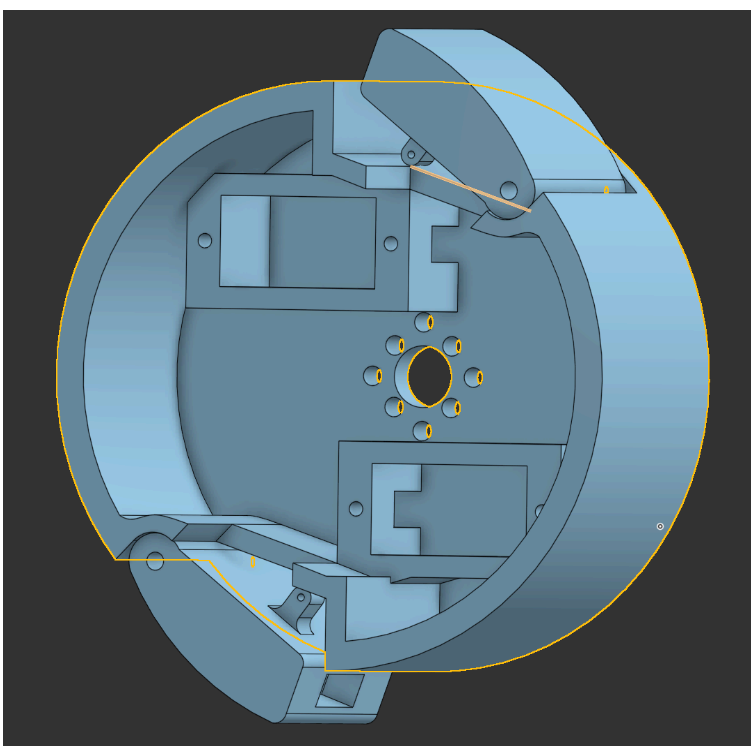


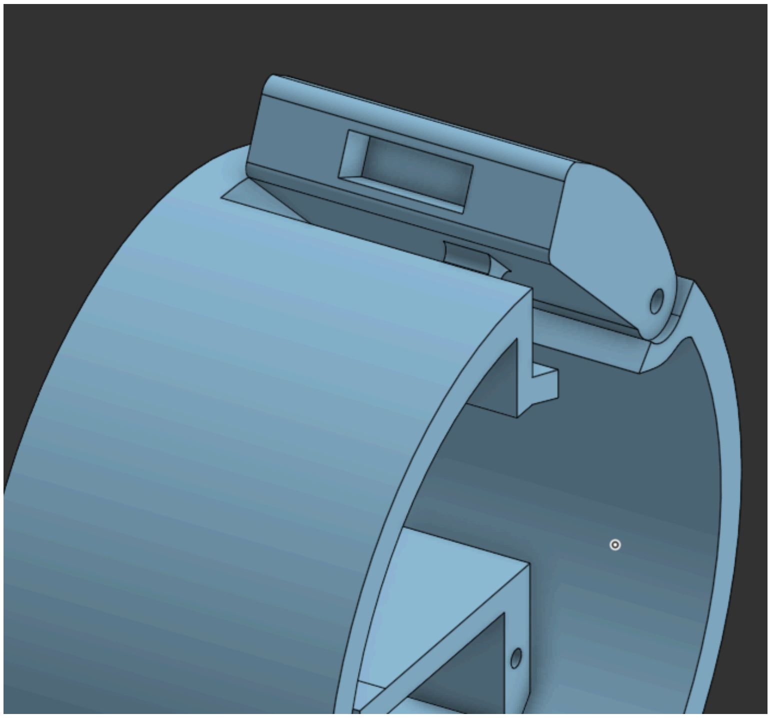
Movement Mechanism:

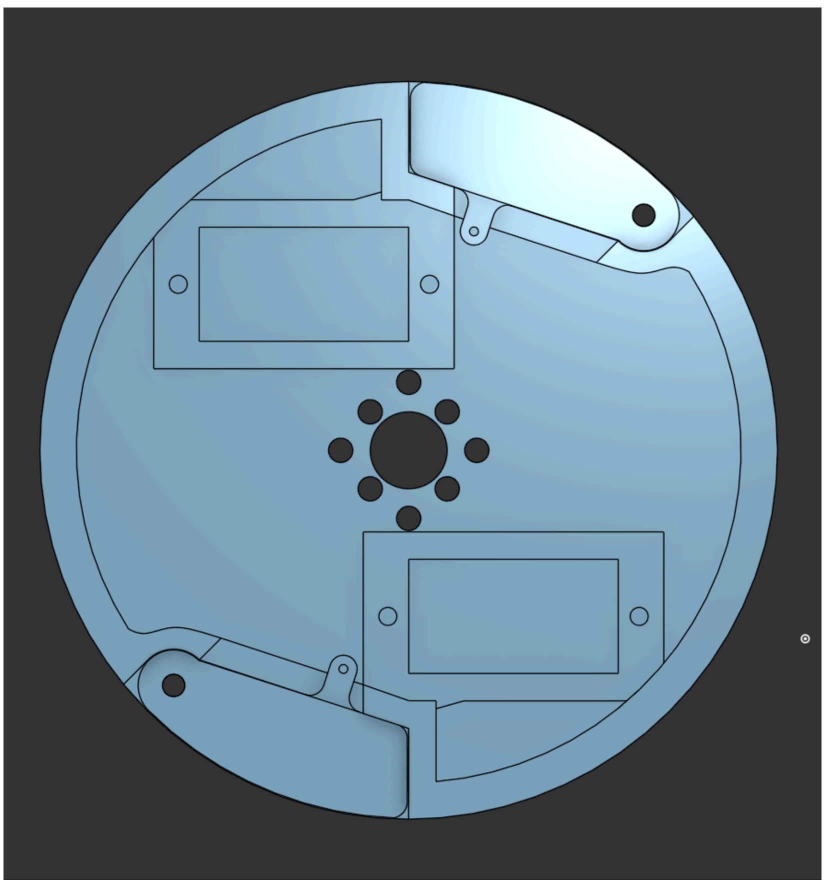


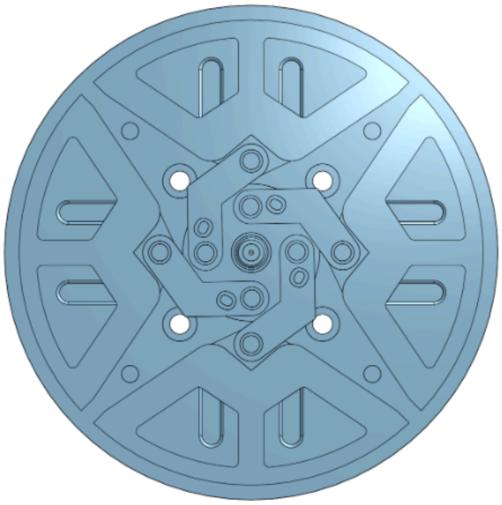
3. CAD Models

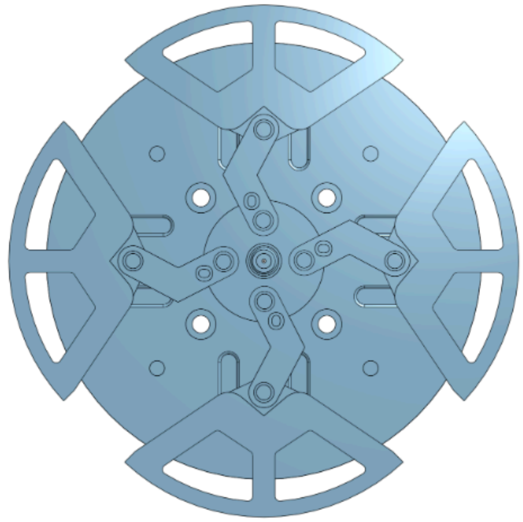




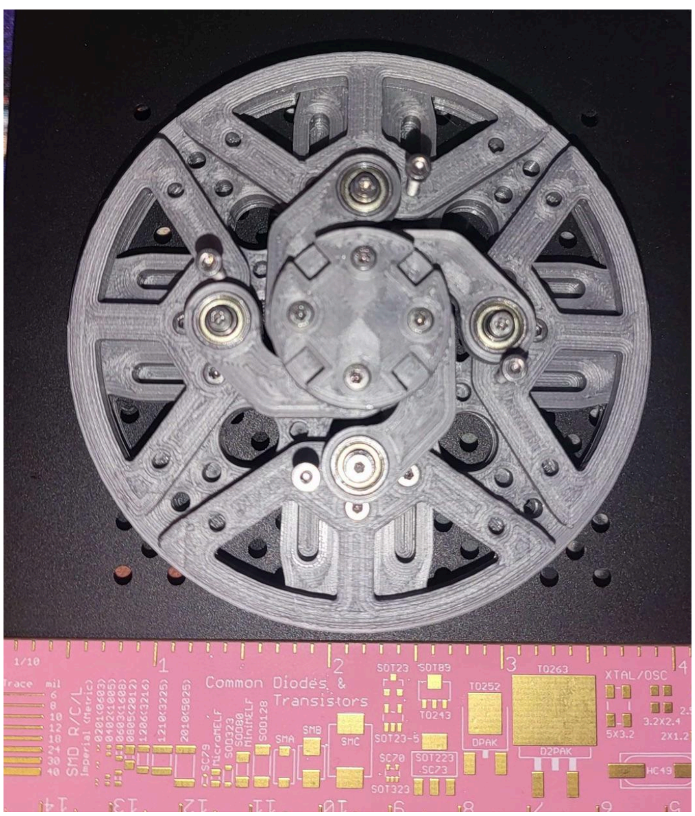


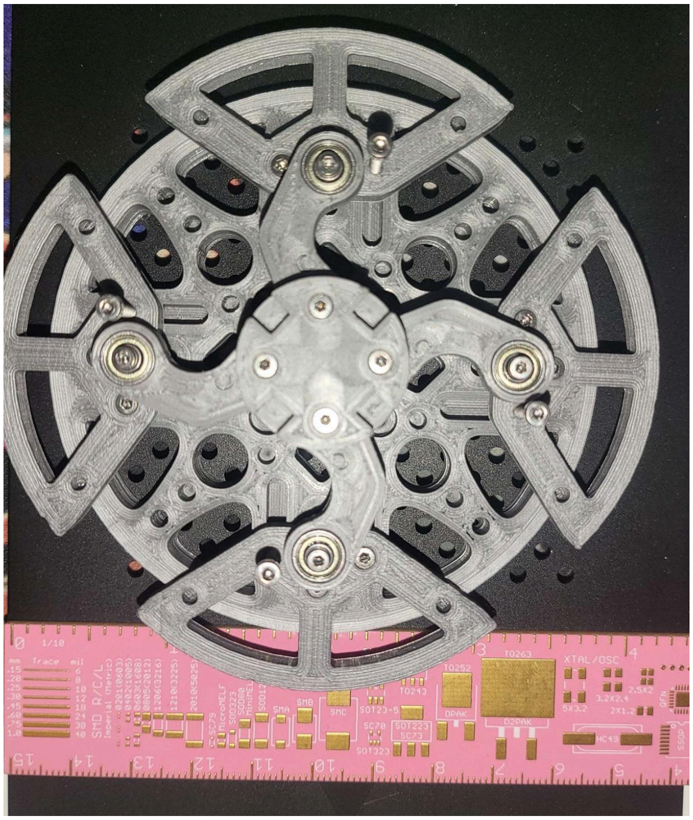


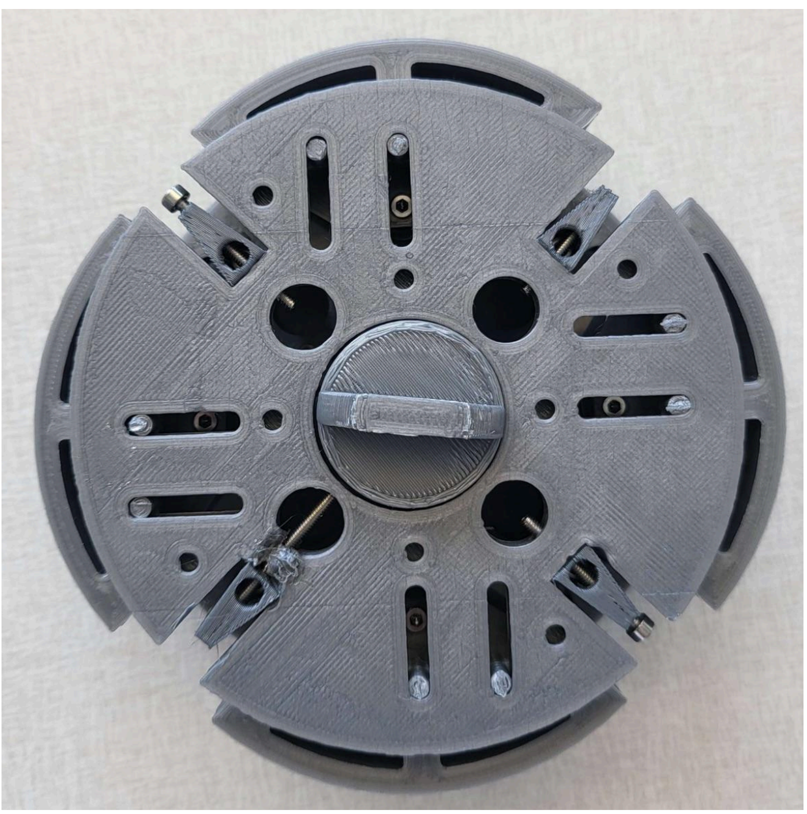


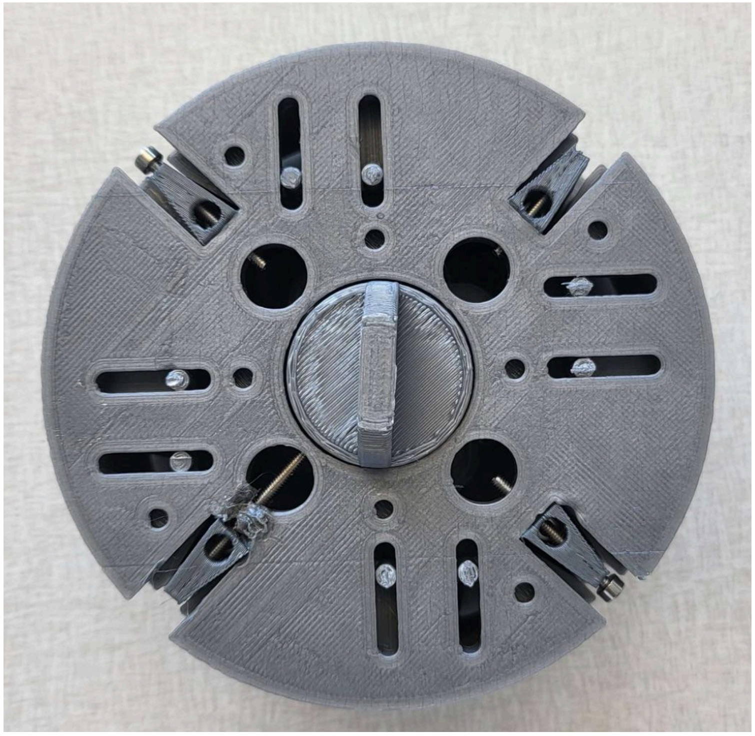


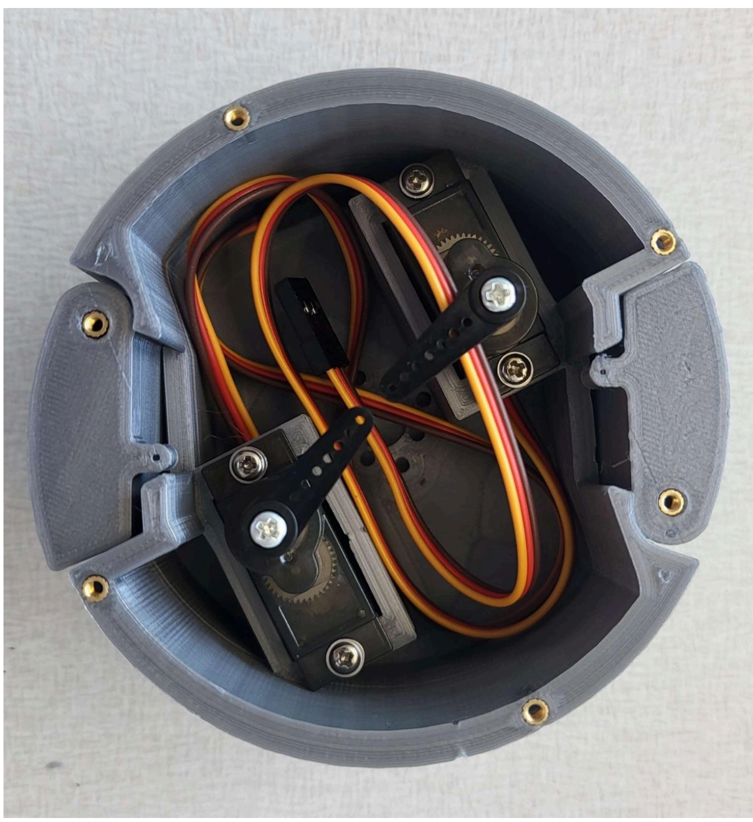
Realized Prototype:

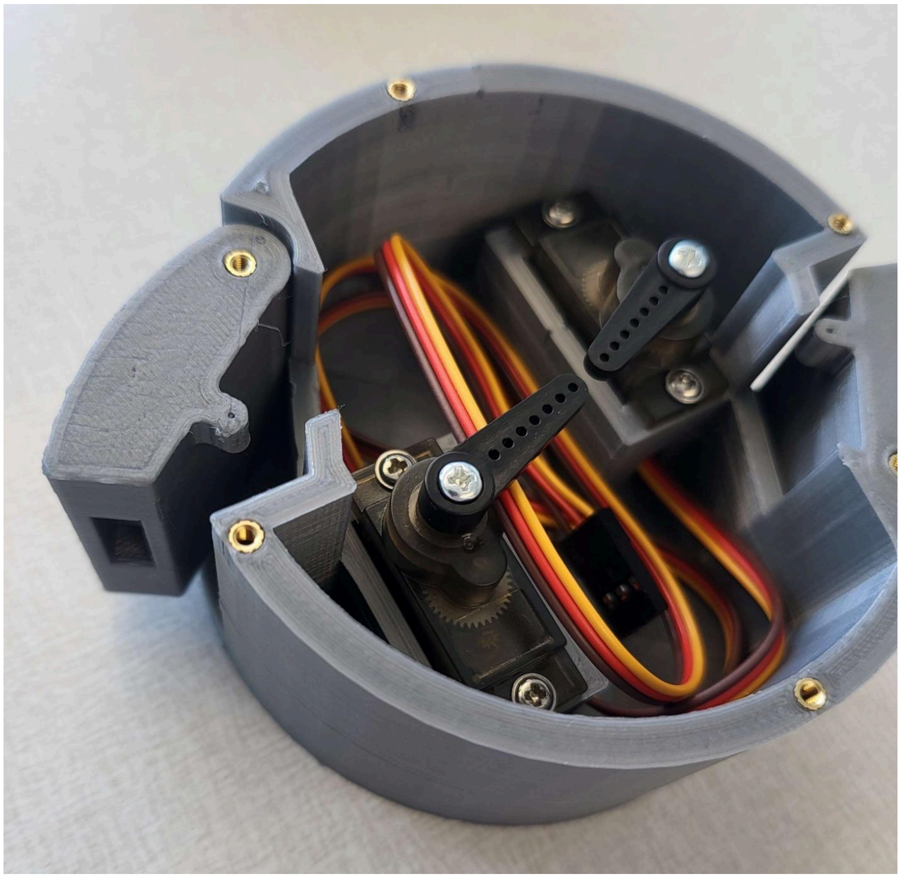



SMD R/C/L
TO263
XTAL/OSC
TO252
DPAK
D2PAK









In the prototypes you can see two of the 3 subsystems designed and printed, one is the locking mechanism and one is the scraping mechanism. Our current plan is to finalize these two subsystems as they are the most critical in making sure our design can function in the environment. Past this deliverable, we want to finish the pushing mechanism that brings our robot into the correct position inside of the tube. We plan on spending some more time on the presentation aspects of the project such as the poster board and the other elements that we were going to incorporate.

Sample Code

Arduino

```
#include <Servo.h>
#include <Wire.h>
#include <MT6701.h>

#define MOTOR1_PIN  12
#define MOTOR2_PIN  13
#define MOTOR3_PIN  14
#define SERVO1_PIN  22
#define SERVO2_PIN  23
#define ENCODER_PIN 21

Servo motor1;
Servo motor2;
Servo motor3;
Servo servo1;
Servo servo2;
MT6701 encoder;

const int lockRotationAngle = 45;
const int unlockRotationAngle = 0;
volatile int pulseCount = 0;
unsigned long lastTime = 0;
float velocity = 0.0;
float angle = 0.0;
String currentStatus = "Idle";

void countPulse() {
  pulseCount++;
}

void setup() {
    motor1.attach(MOTOR1_PIN);
    motor2.attach(MOTOR2_PIN);
    motor3.attach(MOTOR3_PIN);
    servo1.attach(SERVO1_PIN);
```

```
    servo2.attach(SERVO2_PIN);
    Serial.begin(9600);
    Wire.begin();
    encoder.begin();

    pinMode(ENCODER_PIN, INPUT);
    attachInterrupt(digitalPinToInterrupt(ENCODER_PIN), countPulse,
RISING);

    unlockMechanism();
    delay(1000);
    unravelToDepth(1600);
    delay(2000);
    stopMotors();
    delay(3000);
    lockMechanism();
    delay(3000);
    rotateCollectionSystem(1600);
    delay(3000);
    stopMotors();
    delay(5000);
    unlockMechanism();
    delay(1000);
}

void loop() {
    angle = encoder.getAngle();
    velocity = encoder.getVelocity();
    float angleDegrees = angle * (180.0 / PI);

    // Print formatted data
    Serial.print("STATUS: ");
    Serial.print(getStatus());
    Serial.print(", ANGLE: ");
    Serial.print(angleDegrees);
    Serial.print(", VELOCITY: ");
    Serial.print(velocity);
    Serial.println(" rad/s");

    delay(1000);
}

String getStatus() {
```

```
    return currentStatus;
}

void unravelToDepth(int speed) {
    currentStatus = "Unraveling";
    motor1.writeMicroseconds(speed);
    motor2.writeMicroseconds(speed);
    delay(5000);
    stopMotors();
    currentStatus = "Idle";
}

void rotateCollectionSystem(int speed) {
    currentStatus = "Collecting";
    motor3.writeMicroseconds(speed);
}

void stopMotors() {
    currentStatus = "Idle";
    motor1.writeMicroseconds(1000);
    motor2.writeMicroseconds(1000);
    motor3.writeMicroseconds(1000);
}

void lockMechanism() {
    currentStatus = "Locking";
    servo2.write(lockRotationAngle);
    delay(500);
}

void unlockMechanism() {
    currentStatus = "Unlocking";
    servo2.write(unlockRotationAngle);
    delay(500);
}
```

Processing

```
import processing.serial.*;
```

```
Serial myPort;
String status = "Idle";
float angle = 0;
float velocity = 0;
boolean started = false;

void setup() {
  size(400, 300);
  printArray(Serial.list());
  if (Serial.list().length > 0) {
    myPort = new Serial(this, Serial.list()[0], 9600);
    myPort.bufferUntil('\n');
  }
}

void draw() {
  background(30);

  fill(255);
  textSize(24);
  textAlign(CENTER);
  text("LockTech Tool Monitor", width / 2, 40);

  drawLabel("Status", 50, 80, status);
  drawLabel("Scraper Angle", 50, 140, nf(angle, 0, 2) + "°");
  drawLabel("Scraper Velocity", 50, 200, nf(velocity, 0, 2) + " rad/s");

  drawButton();
}

void drawLabel(String title, int x, int y, String value) {
  fill(200);
  textSize(16);
  textAlign(LEFT);
  text(title, x, y);

  fill(getColorForStatus(title, value));
  rect(x + 100, y - 20, 200, 30, 10);

  fill(0);
  textAlign(CENTER, CENTER);
  text(value, x + 200, y - 5);
}
```

```
void drawButton() {
  fill(started ? color(100, 200, 100) : color(200, 100, 100));
  rect(150, 250, 100, 40, 10);

  fill(255);
  textSize(16);
  textAlign(CENTER, CENTER);
  text(started ? "Running" : "Start", 200, 270);
}

color getColorForStatus(String title, String value) {
  if (title.equals("Status")) {
    if (value.equals("Unraveling")) return color(255, 165, 0);     if
(value.equals("Locking")) return color(255, 0, 0);
    if (value.equals("Collecting")) return color(0, 0, 255);
    if (value.equals("Unlocking")) return color(0, 255, 0);
    if (value.equals("Idle")) return color(200, 200, 200);
  }
  return color(255)}

void mousePressed() {
  // Check if the Start button is clicked
  if (mouseX > 150 && mouseX < 250 && mouseY > 250 && mouseY < 290) {
    if (!started) {
      started = true;
      if (myPort != null) {
        myPort.write("START\n");
        println("START command sent");
      }
    }
  }
}

void serialEvent(Serial myPort) {
  String data = myPort.readStringUntil('\n');
  if (data != null) {
    data = trim(data);
    println("Received: " + data);

    if (data.equals("Collection Started!")) {
      started = true;
      return;
```

```
    }


    String[] parts = split(data, ", ");
    if (parts.length >= 3) {
      String[] statusParts = split(parts[0], ": ");
      if (statusParts.length >= 2) {
        status = statusParts[1];
      }

      String[] angleParts = split(parts[1], ": ");
      if (angleParts.length >= 2) {
        angle = float(angleParts[1]);
      }

      String[] velocityParts = split(parts[2], ": ");
      if (velocityParts.length >= 2) {
        velocity = float(velocityParts[1]);
      }
    }
  }
}
```

***AI used

Extension Component

The extension component will have a rigid measuring tape type design that is able to push the device into the tube and also retract it back if or when needed. The rolling and unrolling of the tape will be done by a sort of mechanism that will have enough force and strength to both be able to hold the mechanism up vertically and also push it 15 feet deep into the tube.

Scraping/Collection Mechanism

The scraping mechanism will go hand in hand with the locking mechanism as the inner part of our design will revolve around the center of itself. While the machine is turning, an extendable arm will come out with both the scraper and the collection bin where it will scrape the pipe on each turn and collect it at the same time.

Locking Mechanism

The locking mechanism includes 4 curved wheel type designs that will press against the tube walls when the entire machine is in the correct place, allowing for easy collection of samples. These wheels will be retracted and will only extend when 15 feet inside the tube and scraping is required.

4. Prototyping Test Plan

Test Number	Reason for Prototype	Evaluation Criteria/Determine Measurables	Level of Prototype	Kind of Prototype	Metrics	Test Description	Analysis Method
1	Performance Measurement	Depth test	HiFi Comprehensive	Physical	Distance: feet	Depth test: Testing the maximum depth attainable by the tool	Measure using measuring tape or through other means of analysis
2	Performance Measurement	Portability test	HiFi Comprehensive	Physical	Portability: dimensions of shape (in) and weight (kg)	Portability test: Testing the portability of the device	Weigh using scale and measure using measuring tape
3	Performance Measurement	Reusability test	HiFi Comprehensive	Physical	Reusability: number of samples	Reusability test: Testing the number of samples that can be taken in a certain length of time	Let device run in repeated trials and determine the maximum number of samples possible within x amount of minutes
4	Performance Measurement	Durability test	HiFi Comprehensive	Physical	Durability: number of uses before failure or errors	Durability test: Testing the "life" of the device	Continuously run device and determine how many uses before failure
5	Performance Measurement	Failsafe test	HiFi Comprehensive	Physical	Failsafe test	Failsafe test: Determining if the device can	Run device and then attempt to retrieve the device

						be retrieved in an emergency scenario	while it is conducting the trial
6	Performance Measurement	Movement Mechanism Reliability Test	HiFi Comprehensive	Physical	% of attempts that reach the desired depth	Movement Reliability test: Testing the reliability of the movement system of the device	Conduct a large number of trials and note down how many failed to reach desired depth and calculate % of reliability of mechanism
7	Performance Measurement	Storage test	HiFi Comprehensive	Physical	Volume (mg) and % of attempts that succeed in sample storage	Storage test: Testing to see if the device can successfully storage appropriate amounts of sample consistently	Load the container and measure volume, then run trials and note down number of trials that failed to store desired amounts of sample
8	Performance Measurement	Scraping Mechanism Reliability Test	HiFi Comprehensive	Physical	Amount of sample removed per trial	Scraping Reliability Test: Testing to determine the consistency of the sample collection device	Run device multiple times and determine the average sample size, as well as how often the device collects a sample around the average size
9	Performance Measurement	Locking test	HiFi Comprehensive	Physical	Seal Integrity: maximum weight supported by the locking mechanism	Locking test: Testing the maximum amount of weight that can be supported by the locking mechanism and ensuring that the mechanism can support the rest of the device	Gradually add weight to the device when in vertical orientation until locking system fails
10	Usability Measurement	UX test	HiFi Comprehensive	Analytical	UX Satisfaction:	UX test: Observing users	Conduct trials with different users and

					Time taken for a user to operate device (minutes) and user satisfaction levels	interacting with the device and determining the amount of time needed for the user to learn to operate the device and how satisfied they are with how it is controlled	recording both task completion time and user satisfaction levels
11	Usability Measurement	UI test	HiFi Comprehensive	Analytical	UI Satisfaction: User opinions on UI design	UI test: Observing users interacting with device and getting their opinions on how satisfied they are with the overall appearance of the device	Asking different users about how they feel about the design of the device, and noting down their opinions

5. Testing Results

Reusability test:

- Device is able to collect two samples in one trial

Scraping test:

- Adjustments will need to made to ensure more consistent sample sizes

Locking test:

- Device was able to support some force to the locking mechanism
- Rubber strips were added to the locking mechanism to increase the amount of force that the mechanism could withstand

6. Updated BOM

Item Number	Material/ Component	Unit Cost	Quantity	Total Cost
1	Servo motors: used to roll up and unroll measuring tape, and used for the clamping mechanism	$5	3	$15
2	Tape Measure	$17	1	$17
3	30-Pack 3/8" x 13" Zirconia Sanding Belts, 30 Grit Nylon Material, Metal Grinding SandPaper for Polishing and Grinding: this will be used to grind and sand metal shavings off of the inner pipe to be collected	$15	1	$15
4	Screws	$9	1	$9
5	Magnetic Encoder MT6701 Magnetic Induction Angle Measurement Sensor Module 14bit High Precision Can Perfectly Replace AS5600	$2.47	1	$2.47

6	ABS	$0.13 per gram	1 spool	$22

7. Testing and Procedures for Final Device

a. Design specifications

Design specification	Units	Value	Verification method
Movement mechanism reliability	Percentage	95%	95% reliability of system without fail
Movement distance	Feet	50	see if the prototype can travel 50 feet
Container storage	Mg	30-80	Test scrape in pipe
Scraping mechanism reliability	Percentage	95%	95% reliability of scraping the appropriate amount

b. Validation plan

Design specifications	Test method and materials needed	Methods of recording results	Estimated time duration
Movement mechanism reliability	Shorter length of 4 inch diameter tube. Testing by running the prototype through the tube at least 20 times	How many successful attempts are achieved	20/20
Movement distance	Test the prototype through the tube up to 15 feet 20 times	How many successful attempts are achieved	20/20
Container storage	Prototype container and prototype pipe	If the container can successfully gather an appropriate sample of size from the prototype pipe	10/20

Scraping mechanism	Prototype scraper and prototype pipe	If the scraper is reliable and scrapes the valid amount of pipe.	15/20
Locking Mechanism	Force	If pipe can easily scrape	15/20