Prototype 2 and Customer Feedback

Shehryar Ali Memon

Oluwatamilore Ilupeju

Logan Jones

Saheel Ahmed

March 13, 2022

Abstract

This document is meant to portray our second prototype and analyze our inverse kinematics solutions for our end-effector. It also serves the purpose of defining any challenges we faced during prototyping and makes clear our plan for our next prototype.

Table of Contents

Introduction

This document follows up on the last document, in which we defined our prototype test plan and our stopping criteria. This document contains images of our initial 3D prints of our sander, the 3D render of our paint gun, our revised inverse kinematics solution, and our application module. It further defines our next steps for our future prototypes by virtue of another prototype test plan and makes known the software and hardware challenges we faced during prototyping.

Prototyping test plan

Test ID	**Test Objectives**	**Description of prototypes and basic test method**	**Description of results to be recorded and how these results will be used**	**Estimated test duration and planned start date**
01	Paint gun 3D print	Final product of paint gun	Efficiency checked once printed	03/17/2022 5 hour(s)
02	Minimum Viable (MVP) Product test	Efficiency of whole system checked	Results will be improved upon, or MVP will be revisited	03/14/2022 - 03/19/2022 5 day(s)
03	Design Day readiness	Aesthetics and safety checked	Cleaning up of final details	03/20/2022 5 hour(s)

04	Application module	Using bluetooth to control the arm	If instructions are properly followed and arm does what is required, will be reprogrammed if found not working	03/14/2022-03/17/2022 3 day(s)
06	Sander 3D print	Final product of sander	Efficiency checked once printed	03/17/2022 8 hour(s)
07	Vacuum Efficiency	Amount of residue picked up	Difference in mass in grams of vacuum bag per hour	03/19/2022 2 hour(s)
08	Sander Efficiency	Area cleared in a period of time along with accuracy and precision	Area in m^2 cleared per hour, difference checked between the desired area and cleared area	03/19/2022 2 hour(s)
09	Safety	Overall safety of system checked by obstruction testing and coding IR sensor	Safety and viability checked; Arduino solution will be revisited	03/20/2022 3 hour(s)

Prototype Analysis

Revised Inverse Kinematics Solver

```
#include <Servo.h>
Servo motor1;
Servo motor2;
Servo motor3;
// angles in degree and radian

// length of links of robot arm
volatile float L1;
volatile float L2;
// end effector

volatile float pi = 3.14159265359;

void setup() {
  motor1.attach(11);
  motor2.attach(3);
  motor3.attach(5);
  Serial.begin(9600);
  motor1.write(0);
  motor2.write(0);
  motor3.write(0);
 Serial.println("Enter the length of first arm ");
 while(Serial.available()==0){}
 L1=Serial.parseFloat();

 Serial.println("Enter the length of second arm ");
 while(Serial.available()==0){}
 L2=Serial.parseFloat();

}

void loop() {
   motor1.write(0);
   motor2.write(0);
   motor3.write(0);
   inverseKinematics();
}

void inverseKinematics(){
  float angle1;
  float angle2;
  float angle3;
  float rad_angle1;
```

```
  float rad_angle2;
  float rad_angle3;
  float x;
  float y;
  float z;
   Serial.println("Enter the value x ");
      while(Serial.available()==0){}
      x=Serial.parseFloat();


      Serial.println("Enter the value y ");
      while(Serial.available()==0){}
      y=Serial.parseFloat();

      Serial.println("Enter the value z ");
      while(Serial.available()==0){}
      z=Serial.parseFloat();

      rad_angle2 = acos((sq(z)+ sq(y) - sq(L1) - sq(L2)) / (2*L1*L2));
      rad_angle3 = acos((sq(L1) + sq(L2) - sq(x)- sq(z)) / (2*L1*L2));
      rad_angle1= atan(y / x) - atan((L2*sin(rad_angle2)) / (L1+ L2*cos(rad_angle2)));
      delay(1000);

      angle1= (rad_angle1*180)/pi;
      angle2= (rad_angle2*180)/pi;
      angle3= (rad_angle3*180)/pi;

  Serial.print("x is ");
  Serial.println(x);
  Serial.print("y is ");
  Serial.println(y);
  Serial.print("z is ");
  Serial.println(z);
  Serial.print("angle1 is  ");
  Serial.println(angle1);
  Serial.print("angle2 is ");
  Serial.println(angle2);
  Serial.print("angle3 is ");
  Serial.println(angle3);
  motor1.write(angle1);
  motor2.write(angle2);
  motor3.write(angle3);
  delay(2000);
}
```

Note: We are yet to test out the viability and effectiveness of this program, however we are confident in it's working.

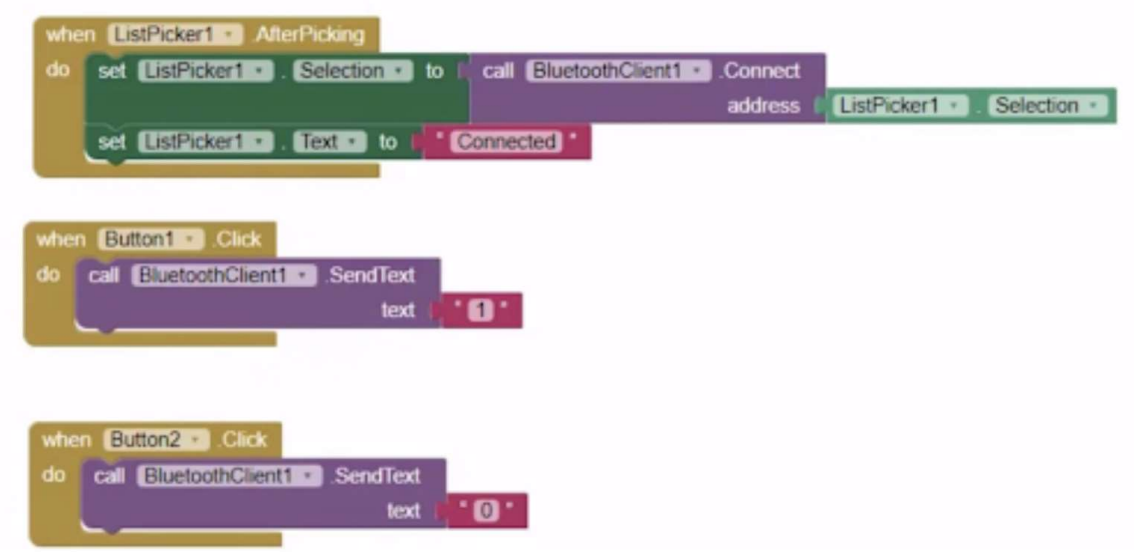
Graphical User Interface

```
sketch_mar06a §
char Incoming_value = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(13, OUTPUT);
}

void loop()
{
  if(Serial.available() > 0)
  {
    Incoming_value = Serial.read();
    Serial.print(Incoming_value);
    Serial.print("\n");
    if(Incoming_value == '1')
      digitalWrite(13, HIGH);
    else if(Incoming_value == '0')
      digitalWrite(13, LOW);
  }
}
```

Arduino code that allows the MIT app to communicate to our arduino through the bluetooth module.



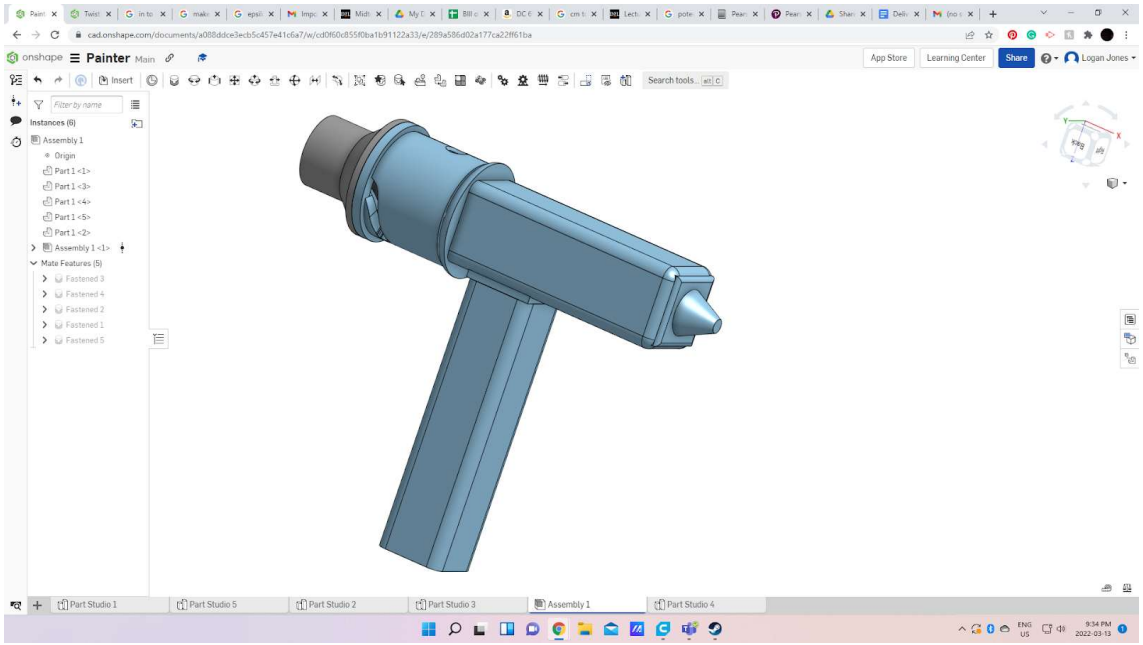
This is the MIT App inventor setup to control the application.

Sander Prototype 3D print

We have begun the process of 3D printing the sander and have the first few parts completed and will continue printing the next parts.

Paint Gun 3D render

The new 3D model is a redesigned version of the paint gun from the first prototype. This new design will make 3D printing and assembly faster and easier.



Challenges

Revised Inverse Kinematics Solver

- Code does not compile properly when Stepper library is used, however works properly with servo motors.
- Will need to be revisited after discussion with TA
- Once problem is figured out, the inverse kinematics solution will be ready to be deployed

Sander Prototype

- Time constraints (long printing times)
- Installation of DC motor
- Assembly of sander

Paint Gun Prototype

- Time constraints (long printing times)
- Feed lines for paint
- Assembly of paint gun

Graphical User Interface

- Understanding how to use bluetooth to control the arduino uno
- Understanding the use of MIT App inventor to show video output
- Making the app easy to use
- Making the app easy to update / change
- Understanding how to connect the arduino to the phone

Wrike Updates: Wrike Snapshot