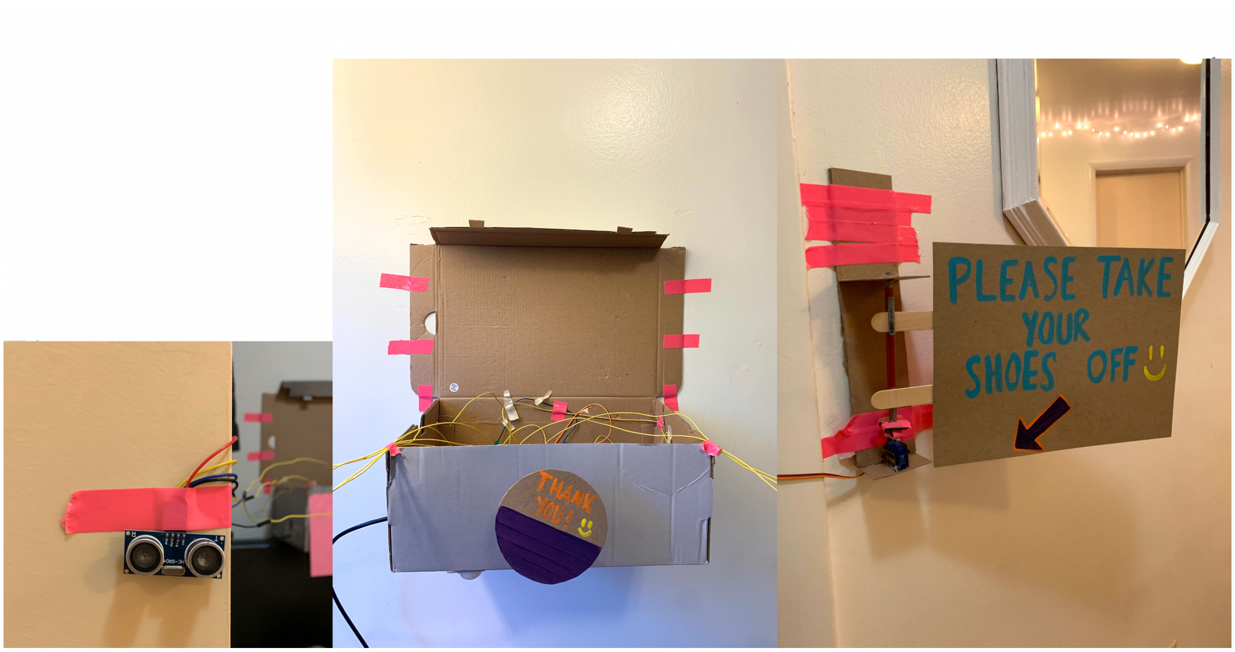
Overview:
An Interactive (and slightly passive aggressive) Take-Off-Your-Shoes Sign

This device detects the front door opening and swings a sign stating “please take your shoes off” away from the wall to face the door. When someone places their shoes on the shoe rack, this action triggers a “thank you” sign to spin. To accomplish this, an ultrasonic sonar distance sensor located behind the door detects the door swinging towards it. When the door is within a specified distance threshold of the sensor, this indicates to the Arduino to rotate a positional servo motor 90-degrees. This motion swings the “please take your shoes off” sign away from the wall, effectively displaying it to the person who has just entered. When the sign has swung off the wall a 10 second timer begins. When 10 seconds is up, the sign rotates negative 90-degrees back to its original position on the wall. A box attached to the wall above the shoe rack houses the Arduino and circuit. When someone places their shoes on the shoe rack, a PIR motion sensor located on the bottom of the box (above the shoe rack) detects this movement. This detected movement indicates to the Arduino to rotate a continuous servo motor to spin a “thank you” sign.



Design Considerations:

a. If I were to do this project again today, I would make a wall of cardboard around the motion sensor so that it only sensed motion when motion occurred directly under it (I used a sticky note, but I think walls surrounding it would be more effective). I would also spend more time using trial and error to adjust the sensitivity of the motion sensor, so it operated at the optimal amount of sensitivity for this application.

b. If I had more time and money, I would have 3D printed two servo motor connector pieces that attached the end of the servo motors. I also would have 3D printed rods that attach to those connector pieces. This would have been a superior alternative to the pencil, cardboard, and glue route I took. In addition, I would buy more wire so that I could make the connections between the circuit and Arduino in the shoe box to the ultrasonic sensor and positional servo motor more organized (I was running out of wire, so I had to make the connections as direct as possible which reduced overall organization).

Assembly

Circuit

1. Build circuit *(Appendix B).*
2. Determine exactly how the positional and continuous servos will incorporate into mechanical assembly.
    a. **Positional servo:** determine the orientation of the 180-degree range. Note how servo must be attached to the mechanical assembly so that the sign swings away from the wall as desired (Appendix C, Fig. 1).
    b. **Continuous servo:** determine the speed and time duration the servo needs to spin to make a 180-degree rotation and rotate the "thank you" sign.

Mechanical Assembly

Assembly #1: "Take off your shoes" sign & wall attachment

1. **"Take off your shoes sign":** Use paint or markers to design the "take off your shoes sign" on the piece of cardstock.
2. **Cardboard wall bracket:** Cut a 2" X 12" piece of cardboard and a 2" X 2" piece of cardboard. Use glue to attach the 2" X 2" piece to the bottom of the 2" X 12" piece at a 90-degree angle (*Appendix C, Fig. 1*).
3. **Positional servo to sign attachment assembly:**
    a. Cut 3, 0.5" X 1.25" pieces of cardboard. Use glue to attach them in a stack (*Appendix C, Fig. 2*).
    b. Cut 1, 1" X 1" piece of carboard and glue to the top of the stack made in (3. B) (*Appendix C, Fig. 2*).
    c. Create a small hole (~2mm diameter which is the width of a pencil) in the top of the 1" X 1" piece of cardboard and place a sizeable dot of superglue on the hole. Immediately put push the tip of a mechanical pencil about 0.25" into the hole so that the pencil is standing up vertically. Let glue dry. (*Appendix C, Fig. 2*).
    d. Glue the top face of a servo horn to the bottom face of the cardboard stack (*Appendix C, Fig. 2*).
    e. Tape two popsicle sticks to the back of the "take off your shoes sign" (*Appendix C, Fig. 2*).

   f. Glue the popsicle sticks to the front face of the pencil (*Appendix C, Fig. 2*).
   g. Attach positional servo to the servo horn (*Appendix C, Fig. 2*).
4. Create a small hole (~2mm diameter which is the width of a pencil) on the 2" X 12" cardboard directly above where it is attached to the 2" X 2" piece (*Appendix C, Fig. 2*). Thread the positional servo wires through hole.
5. Attach positional servo to center 2" X 2" piece using glue (*Appendix C, Fig. 2*).
6. Cut a 2.5" X 2.5" piece of cardboard to serve as a brace for the top of the pencil. Measure and create a small hole (~2mm diameter which is the width of a pencil) to place the top of the pencil in. Glue the 2.5" X 2.5" piece to 2" X 12" piece at a 90-degree angle. Stick the top of the pencil through the bottom of the hole in the 2.5" X 2.5" piece (*Appendix C, Fig. 2*).
7. Use tape to attach *cardboard wall bracket* to the wall.

Assembly #2: "Thank you" sign, motion sensor, circuit, & shoe box

1. **"Thank you" sign:** Cut a circular piece of cardboard with a 3" diameter. Use paint/markers to design the "thank you" sign (*Appendix C, Fig. 3*).
2. **Continuous servo to sign attachment assembly:**
   a. Repeat steps 3. A-D. in Assembly #1 (*Appendix C, Fig. 2*).
   b. **Servo stand:** Cut 3, 1.5" X 1.5" pieces of cardboard. Use glue to attach them to create a stand. Glue the stand to the base of the shoe box. (*Appendix C, Fig. 4*).
   c. Attach continuous servo to the top of the servo stand to that the front face of the servo is facing the front of the box (*Appendix C, Fig. 4*).
   d. Create a small hole in the center of the front face of the shoe box (~2mm diameter which is the width of a pencil) and push the end of the pencil with the eraser from inside the shoe box through the hole so that the top of the pencil sits outside the box (*Appendix C, Fig. 5*).
   e. Attach the servo horn to the continuous servo motor (*Appendix C, Fig. 4).*
   f. Glue the back of the "thank you" sign to the eraser of the pencil (*Appendix C, Fig. 5).*
3. **Motion sensor attachment:**
   a. Cut out a 1.5" X 1.5" hole on the bottom front corner of the shoe box (*Appendix C, Fig. 5).*
   b. Tape the motion sensor just above the hole so that the bottom of the motion sensor is resting below the bottom face of the box (*Appendix C, Fig. 5).*
4. **Circuit:** place breadboard, Arduino, and 6V battery inside the shoe box.
5. Use tape to attach the shoebox to the wall so the bottom of the shoe box is located 2 feet above the shoe rack.

Assembly 3: Ultrasonic distance sensor

1. Attach ultrasonic distance sensor to the wall behind the front door using tape (*Appendix C, Fig. 6).*

Operation instructions

1. Turn on the 6V battery is on and attach Arduino to power (I attached my computer, but you can also use a 6V battery).
2. When the user enters the front door of the apartment, they just follow the sign's directions and remove their shoes.
3. The user must wait for the sign to return to the wall before placing their shoes on the shoe rack.
4. Push the reset button on the Arduino to reset the program for the next user (*Appendix C, Fig. 7).*
5. ***Note*** if the motion sensor is detecting movement in front of the shoe box place a piece of paper on the front face of the shoe box in front the motion sensor so that only motion under the box is detected by the sensor.

Appendix A: Bill of Materials (BOM)

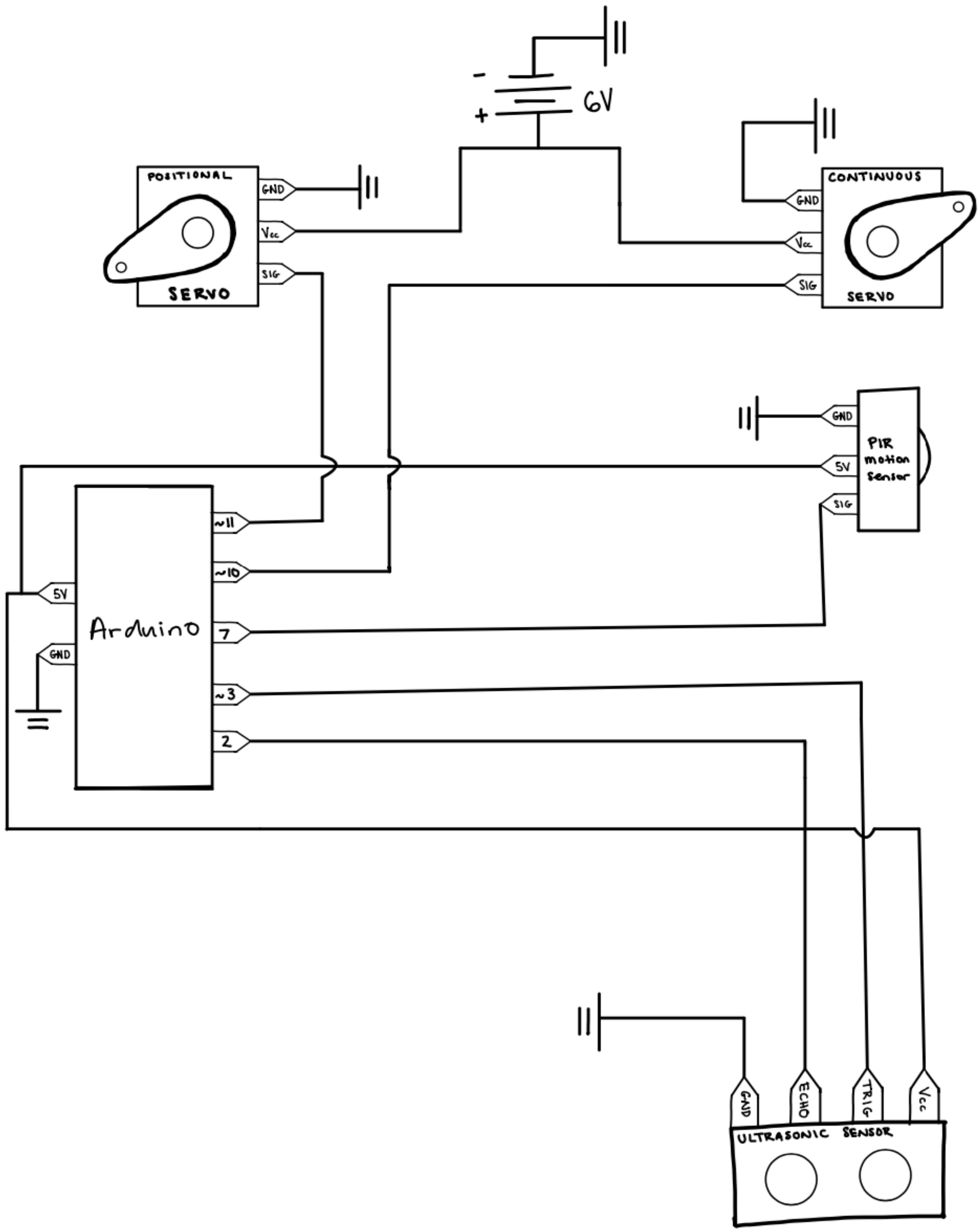
a. BOM for total cost of project if everything was bought from scratch

Part Name	Vendor/Source	Part Number	Quantity	Price	Subtotal
Ultrasonic Sensor	Digi-key	1528-2711-ND	1	$3.95	$3.95
Motion Sensor	Digi-key	1528-1991-ND	1	$9.95	$9.95
5" x 7" x 1/16" Cardboard	in stock crafts	-	4	$0.11	$0.44
22AWG Solid-core Hookup Wire	in stock crafts	-	20 ft	$0.10	$2.00
Shoe Box	scavenged	-	1	$0.74	$0.74
Cardstock	in stock crafts	-	1	$0.15	$0.15
Tongue Depressor	in stock crafts	-	2	$0.03	$0.06
Pencil	scavenged	-	2	$0.27	$0.54
Micro Servo positional	DFRobot	SER0006	1	$3.30	$3.30
Micro Servo continuous	DFRobot	SER-0043	1	$3.90	$3.90
Arduino Board	Digi-key	1050-1024-ND	1	$20.90	$20.90
Bread Board	Newark	79X3922	1	$2.71	$2.71
4-AA Battery Holder	Jameco	216187	1	$1.75	$1.75
AA Batteries	McMaster-Carr	71455K58	4	$0.40	$1.60
Wire Kit	Amazon: Austor	ASIN : B07PQKNQ22	1	$2.17	$2.17
Wire Stripper	Newark	85X3066	1	$4.53	$4.53
3-Wire Extension	Digi-key	1568-1930-ND	2	$1.35	$2.70
Alligator Clip Lead	Sparkfun	PRT-12978	2	$0.33	$0.66
USB Cable A to B	Monoprice	39918	1	$1.09	$1.09
Glue	in stock crafts	-	1	free	-
Tape	in stock crafts	-	1	free	-
				Total:	$63.14

b. BOM for total cost of purchased/scavenged parts

Part Name	Vendor/Source	Part Number	Quantity	Price	Subtotal
Ultrasonic Sensor	Digikey	1528-2711-ND	1	$3.95	$3.95
Motion Sensor	Digikey	1528-1991-ND	1	$9.95	$9.95
5" x 7" x 1/16" Cardboard	in stock crafts	-	4	$0.11	$0.44
22AWG Solid-Core Hookup Wire	in stock crafts	-	20 ft	$0.10	$2.00
Shoe Box	scavenged	-	1	$0.74	$0.74
Cardstock	in stock crafts	-	1	$0.15	$0.15
Tongque Depressor	in stock crafts	-	2	$0.03	$0.06
Pencil	scavenged	-	2	$0.27	$0.54
				Total:	$17.83

Appendix B: Circuit Diagram



Appendix C: Schematics and Drawings for Mechanical Assembly

Figure 1

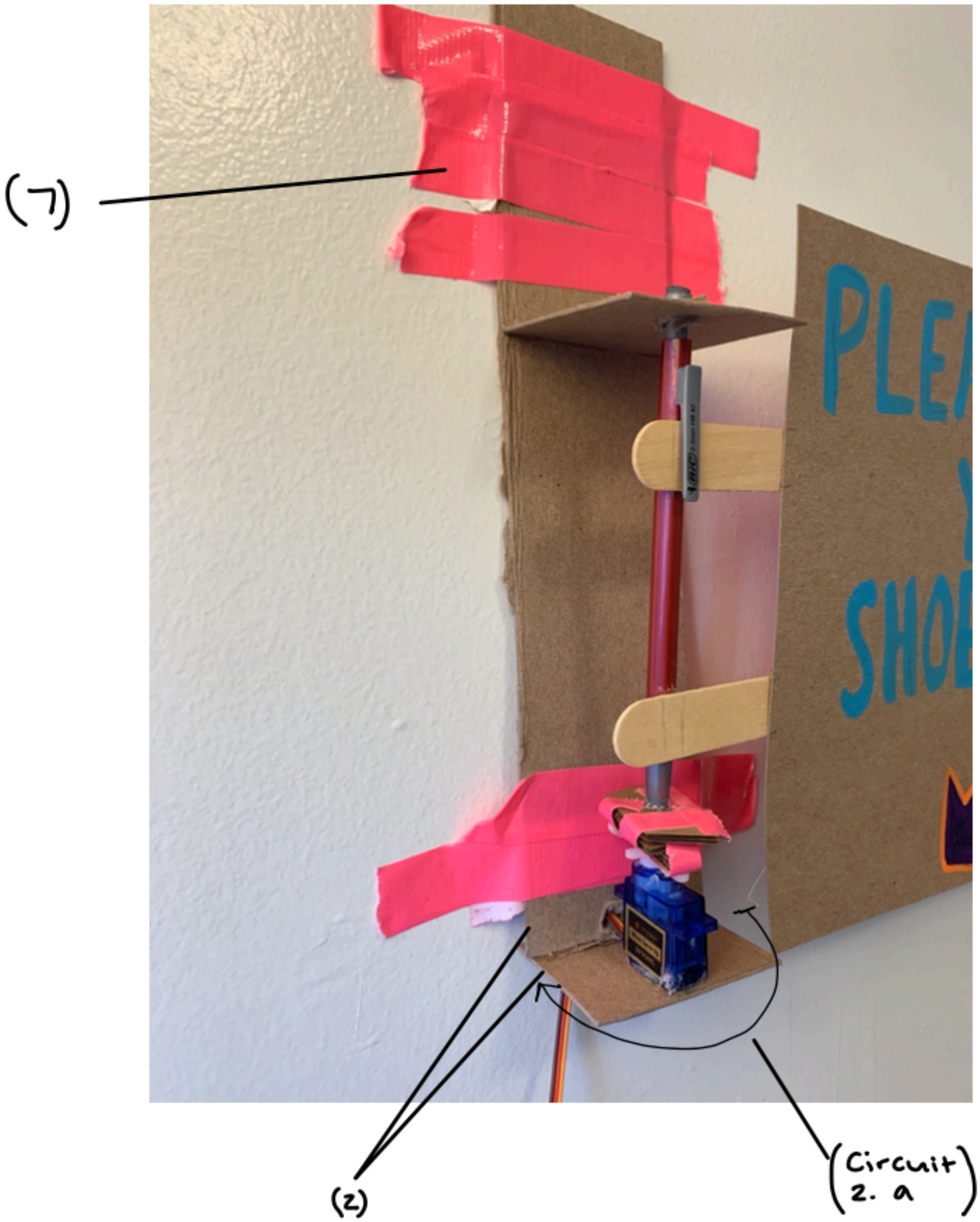


Figure 2

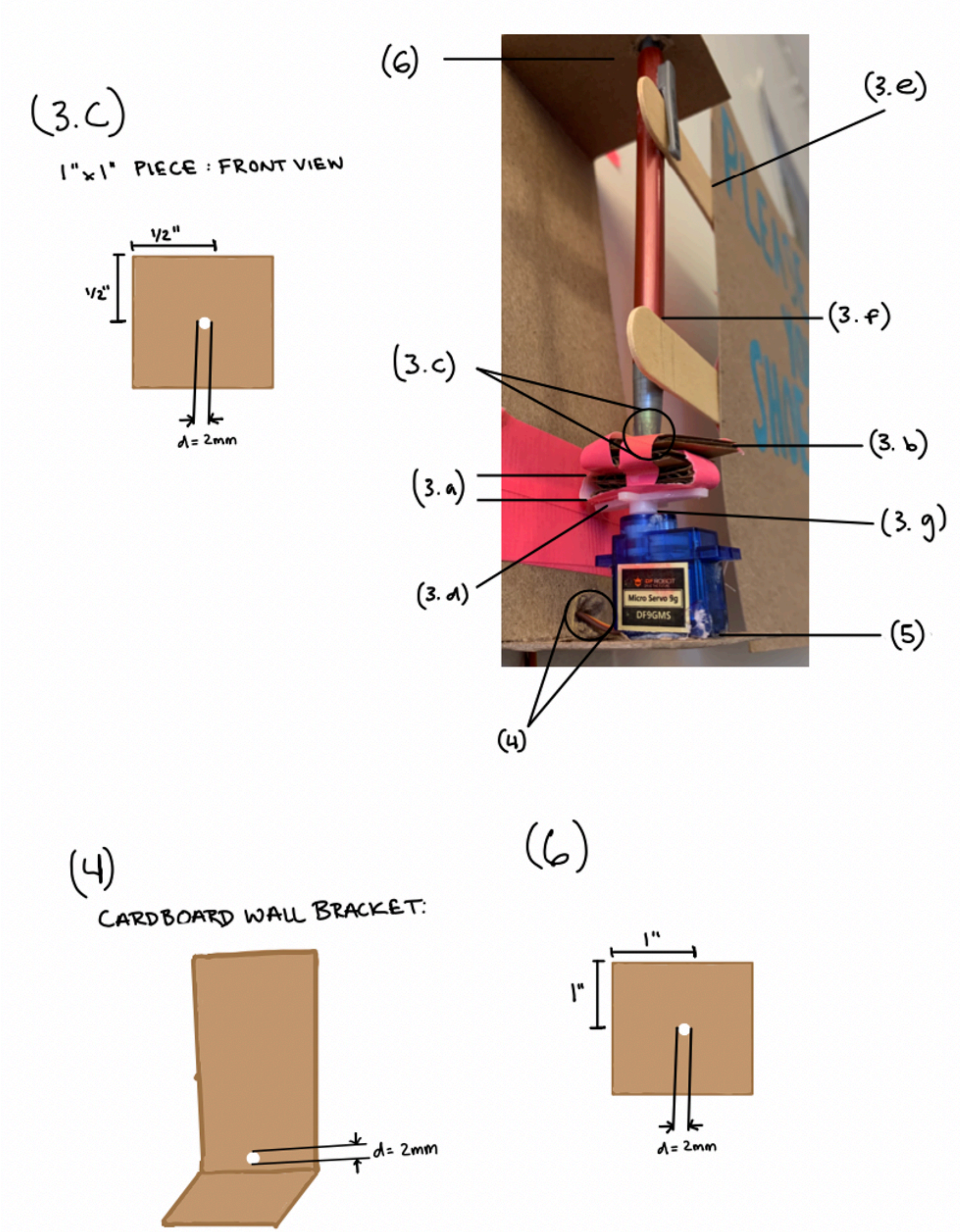

(3.C)
1"x1" PIECE : FRONT VIEW
1/2"
1/2"
d= 2mm
(6)
(3.e)
(3.f)
(3.c)
(3.b)
(3.a)
(3.g)
(3.d)
(5)
(4)
Micro Servo 9g
DF9GMS
(4)
CARDBOARD WALL BRACKET:
d= 2mm
(6)
1"
1"
d= 2mm

Figure 3

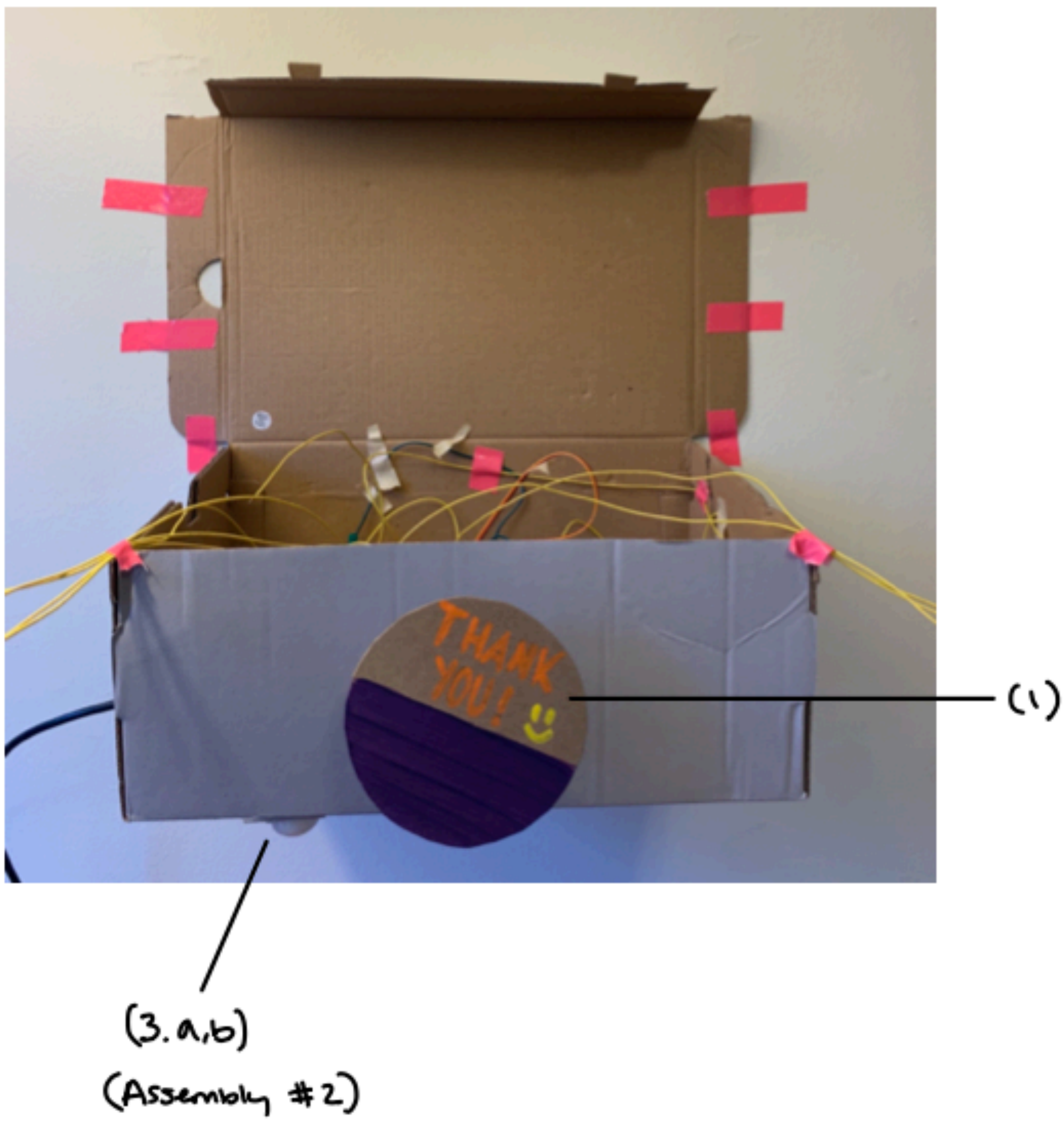


Figure 4

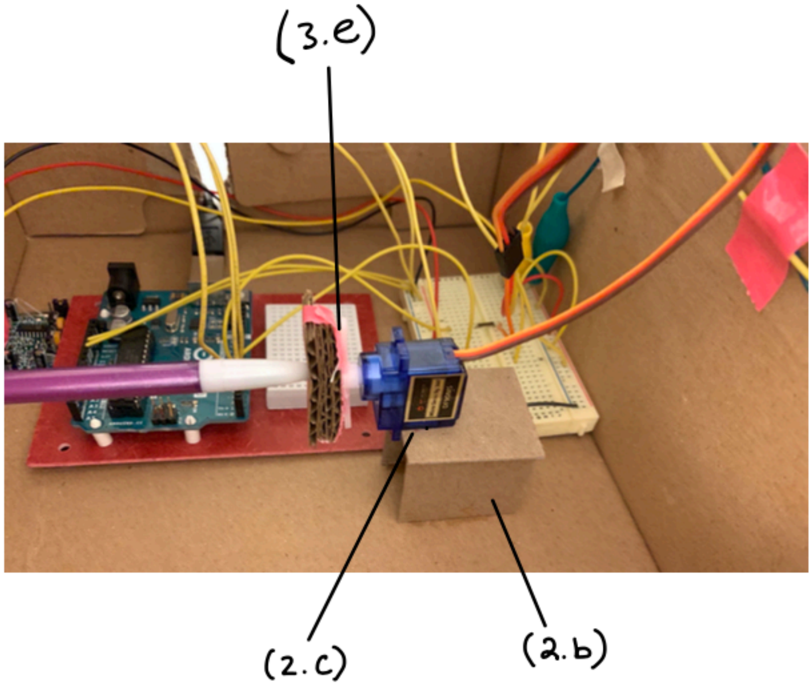
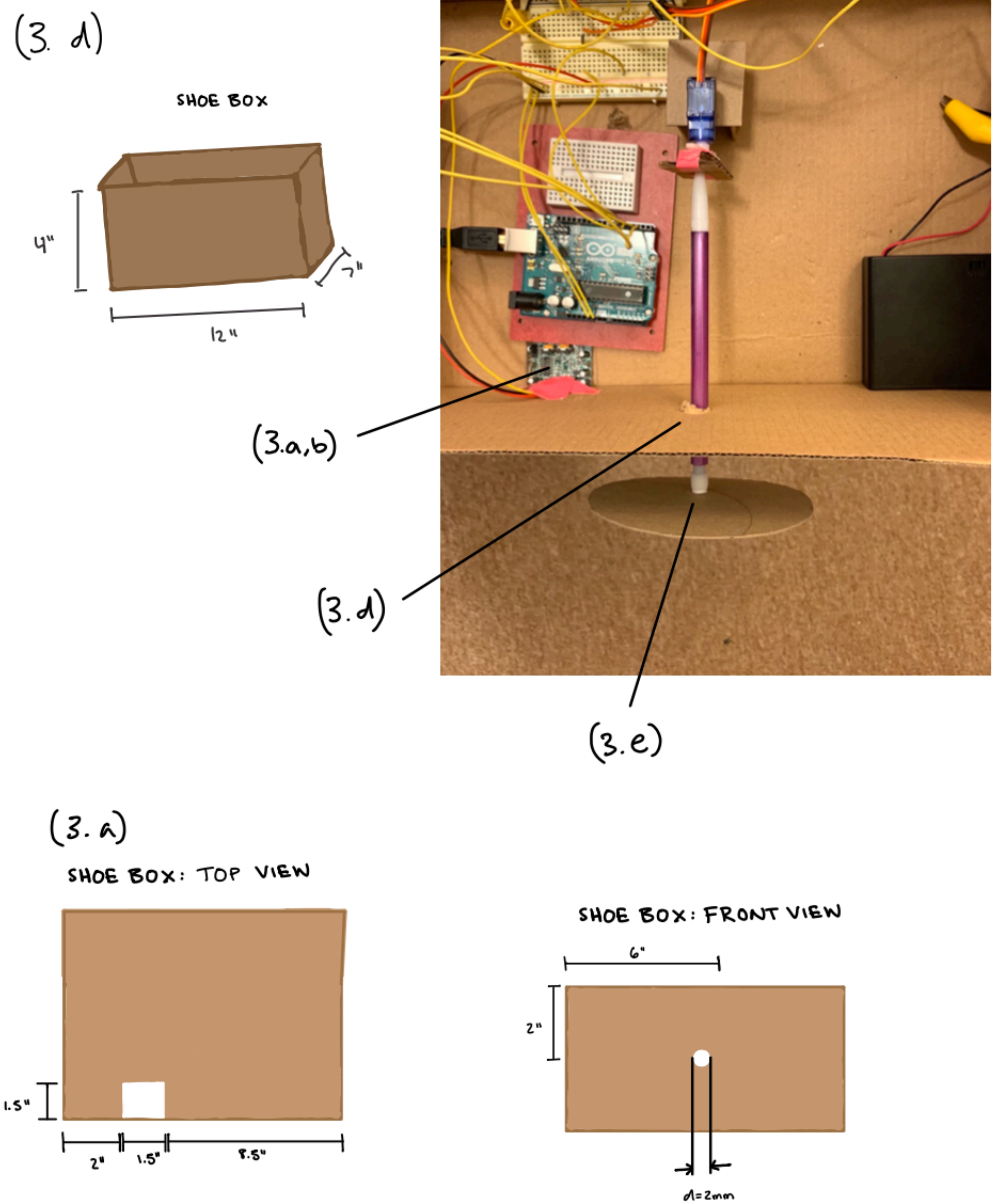


Figure 5



(3. d)
SHOE BOX
4"
7"
12"
(3.a,b)
(3.d)
(3.e)
(3. a)
SHOE BOX: TOP VIEW
1.5"
2"
1.5"
8.5"
SHOE BOX: FRONT VIEW
6"
2"
d=2mm

Figure 6

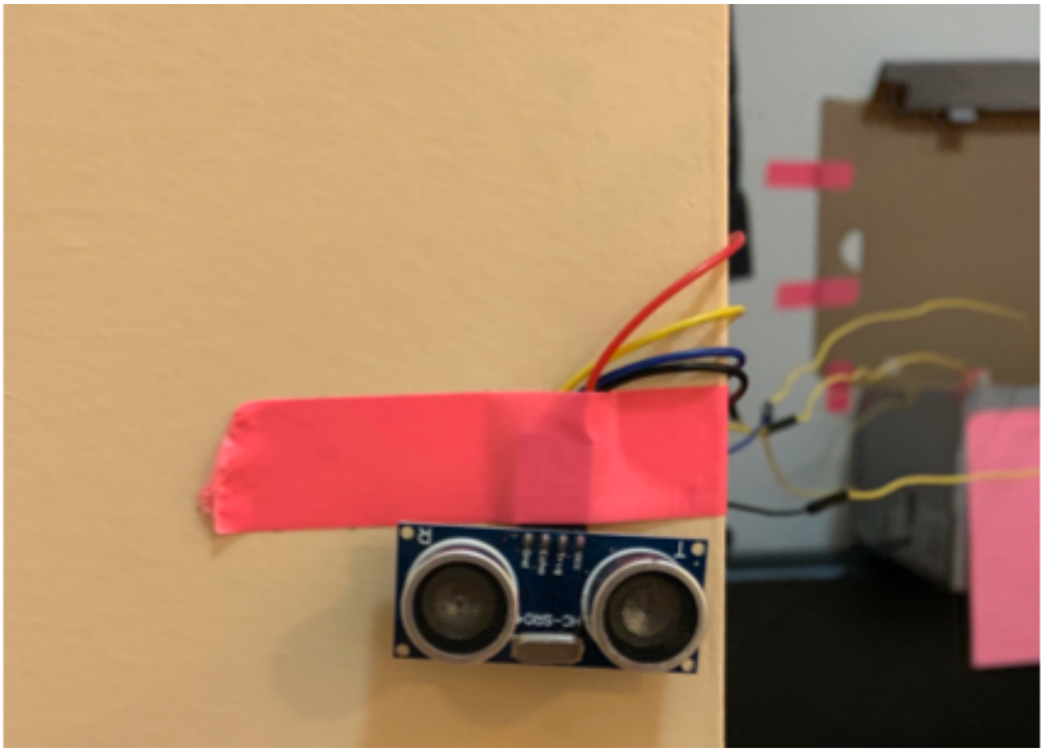
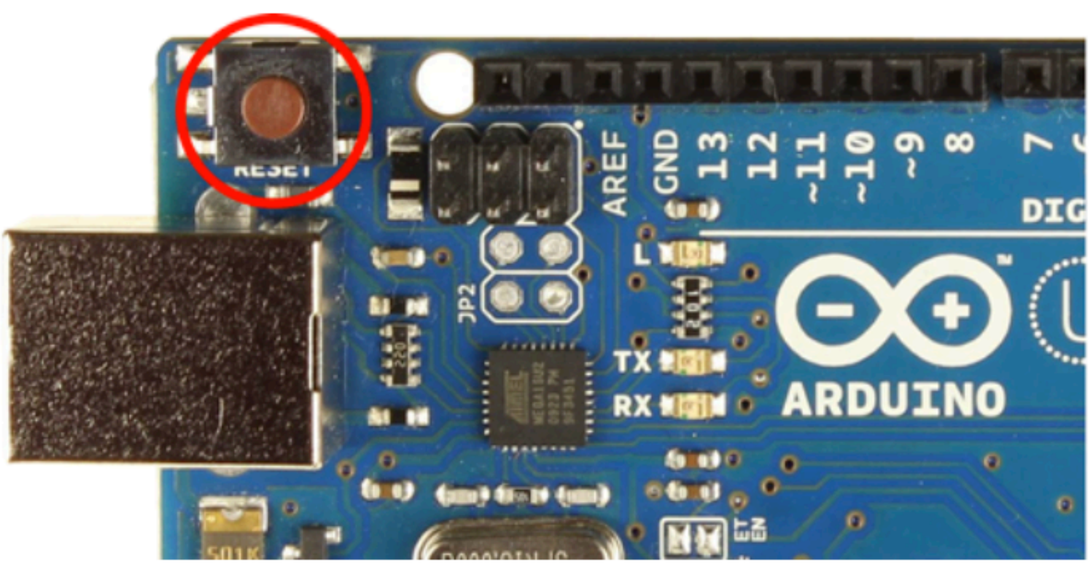


Figure 7



Appendix D: Commented Arduino Code

```
// Maia Raynor, msr266
//MAE 3780 Individual Project
// 11/3/21

//This code uses an ultrasonic sensor and motion detector to move a positional
//servo motor and a continuous servo motor for the "Interactive (and slightly
// passive aggressive" Take-Off-Your-Shoes Sign" project.

//PART 1:
//When the ultrasonic sensor detects the front door within a defined  distance
//threshold the positional servo motor rotates 90 degrees (this rotates the
// "take-off-your-shoes" sign 90 degrees off of the wall to face the user.

//PART 2:
//When the motion sensor detects the movement of someone placing their shoes on
//the shoe rack, a continuous servo motor to rotates at a defined speed for a
//short period of time, effectively rotating approximately 180 degrees. This rotation
//spins and displays a "thank you" sign.


#include <Servo.h>


/////inputs for ultrasonic sensor & servoMotor1 (a positional servo motor) //////

int triggerPin=3; //output signal pin for ultrasonic sensor
int echoPin=2; //input signal pin for ultrasonic sensor
int servoPin1=11; //setting servoMotor1 to pin 11
int distance_threshold; //target distance threshold
bool signOpen =false; //sign is initially closed
bool motionDetected =false; //no motion is initially detected
Servo servoMotor1; //setting up servoMotor1 which will rotate 90 degrees due to
ultrasonic sensor


/////inputs for motion sensor & servoMotor2 (a continuous servo motor) //////

int motionSensor_state=0; //creating variable to store current state of motion sensor
int servoPin2=10; //setting servoMotor2 to pin 12
Servo servoMotor2; //setting up servoMotor2 which will rotate continuously for a set
amount of time
```

```
void setup()
{
 Serial.begin(9600); //defining rate of communication from Arduino to Serial Monitor

 //set up for ultrasonic sensor & servoMotor1
 pinMode(servoPin1, OUTPUT);// setting servoMotor1 pin as an output
 servoMotor1.attach(servoPin1); //attaching servoPin1 to servoMotor1
 servoMotor1.write(180); //initial position for servoMotor1

 //set up for motion sensor & servoMotor2
  pinMode(7, INPUT); //setting input signal for motion sensor to be pin 7
  pinMode(servoPin2, OUTPUT); // setting Servo_2 pin as an output
  servoMotor2.attach(servoPin2); //attaching servoPin2 to servoMotor2
  servoMotor2.write(90); //initial speed of servoMotor2 is 0

}

void loop()
{

//loop for ultrasonic sensor & servoMotor1
distance_threshold= 45; //[cm] setting threshold distance to activate servo motor

//function that measures distance reported by ultrasonic sensor in cm
long duration, inches, cm;
  {
pinMode(triggerPin, OUTPUT); // clearing triggerPin
digitalWrite(triggerPin, LOW);
delayMicroseconds(2); // wait for 2 microseconds
digitalWrite(triggerPin, HIGH); //sets triggerPin to HIGH for 5 microseconds
delayMicroseconds(5); // wait for 5 microseconds
digitalWrite(triggerPin, LOW);

//reads the echoPin and returns sound wave travel time
pinMode(echoPin, INPUT);  //sets echoPin as input
duration = pulseIn(echoPin, HIGH); //returns length of pulse in microseconds

//convert the time into a distance
cm = microsecondsToCentimeters(duration);


//prints distance detected by ultrasonic sensor in [cm]
Serial.print(cm);
```

```
Serial.print("cm");
Serial.println();

delay(100); //delay 100 microseconds


}


 //ultrasonic sensor & servoMotor1
 //


  //if distance detected by ultrasonic sensor is less than threshold distance,
  //servoMotor1 rotates 90 degrees at 5 degree increments
  if (cm<distance_threshold && signOpen==false){
    servoMotor1.write(180);
    delay(100); //delays 100 microseconds
    servoMotor1.write(175); //moves servoMotor1 to position 175

    delay(100); //delays 100 microseconds
    servoMotor1.write(170); //moves servoMotor1 to position 170

    delay(100); //delays 100 microseconds
    servoMotor1.write(165); //moves servoMotor1 to position 165

    delay(100); //delays 100 microseconds
    servoMotor1.write(160); //moves servoMotor1 to position 160

    delay(100); //delays 100 microseconds
    servoMotor1.write(155); //moves servoMotor1 to position 155

    delay(100); //delays 100 microseconds
    servoMotor1.write(150); //moves servoMotor1 to position 150

    delay(100); //delays 100 microseconds
    servoMotor1.write(145); //moves servoMotor1 to position 145

    delay(100); //delays 100 microseconds
    servoMotor1.write(140); //moves servoMotor1 to position 140

    delay(100); //delays 100 microseconds
    servoMotor1.write(135); //moves servoMotor1 to position 135

    delay(100); //delays 100 microseconds
    servoMotor1.write(130); //moves servoMotor1 to position 130
```

```
    delay(100); //delays 100 microseconds
    servoMotor1.write(125); //moves servoMotor1 to position 125

    delay(100); //delays 100 microseconds
    servoMotor1.write(120); //moves servoMotor1 to position 120

    delay(100); //delays 100 microseconds
    servoMotor1.write(115); //moves servoMotor1 to position 115

    delay(100); //delays 100 microseconds
    servoMotor1.write(110); //moves servoMotor1 to position 110

    delay(100); //delays 100 microseconds
    servoMotor1.write(105); //moves servoMotor1 to position 105

    delay(100); //delays 100 microseconds
    servoMotor1.write(100); //moves servoMotor1 to position 100

    delay(100); //delays 100 microseconds
    servoMotor1.write(95); //moves servoMotor1 to position 95


    delay(100); //delays 100 microseconds
    servoMotor1.write(90); //moves servoMotor1 to position 90



    delay(10000); //starts timer for 10 seconds


//servoMotor1 rotates back -90 degrees to its original position on the wall

    servoMotor1.write(90); //moves servoMotor1 to position 90
    delay(100); //delays 100 microseconds

    servoMotor1.write(95); //moves servoMotor1 to position 95
    delay(100); //delays 100 microseconds

    servoMotor1.write(100); //moves servoMotor1 to position 100
    delay(100); //delays 100 microseconds

    servoMotor1.write(105); //moves servoMotor1 to position 105
    delay(100); //delays 100 microseconds
```

```
servoMotor1.write(110); //moves servoMotor1 to position 110
delay(100); //delays 100 microseconds

servoMotor1.write(115); //moves servoMotor1 to position 115
delay(100); //delays 100 microseconds

servoMotor1.write(120); //moves servoMotor1 to position 120
delay(100); //delays 100 microseconds

servoMotor1.write(125); //moves servoMotor1 to position 125
delay(100); //delays 100 microseconds

servoMotor1.write(130); //moves servoMotor1 to position 130
delay(100); //delays 100 microseconds

servoMotor1.write(135); //moves servoMotor1 to position 135
delay(100); //delays 100 microseconds

servoMotor1.write(140); //moves servoMotor1 to position 140
delay(100); //delays 100 microseconds

servoMotor1.write(145); //moves servoMotor1 to position 145
delay(100); //delays 100 microseconds

servoMotor1.write(150); //moves servoMotor1 to position 150
delay(100); //delays 100 microseconds

servoMotor1.write(155); //moves servoMotor1 to position 155
delay(100); //delays 100 microseconds

servoMotor1.write(160); //moves servoMotor1 to position 160
delay(100); //delays 100 microseconds

servoMotor1.write(165); //moves servoMotor1 to position 165
delay(100); //delays 100 microseconds

servoMotor1.write(170); //moves servoMotor1 to position 170
delay(100); //delays 100 microseconds

servoMotor1.write(175); //moves servoMotor1 to position 175
delay(100); //delays 100 microseconds

servoMotor1.write(180); //moves servoMotor1 to position 180
delay(100); //delays 100 microseconds
```

```
    signOpen=true; //sign has been opened, exit if statement

    //displays that sign is open in Serial Monitor
    Serial.print(signOpen);
    Serial.print("signOpen");
    Serial.println();

    delay(100); //wait for 100 milliseconds

  }



 //motion sensor & servoMotor2

  motionSensor_state= digitalRead(7); //reads the state of the sensor digital input

  //if the motion sensor detects movement of someone placing shoes on shoe rack,
  //indicates to servoMotor2 to rotate 180 degrees to display "thank you" sign

  if (motionSensor_state==HIGH && motionDetected==false) {
    servoMotor2.write(80); //rotates servo motor at speed 80
    delay(1800); //starts timer for 1.800 seconds (the time it takes to rotate ~180 degrees
at speed 80 (determined by testing)
    servoMotor2.write(90); //after 1.8 second delay, servoMotor2 stops rotating
    motionDetected=true; //motion has been deteced, exit if loop

    //displays that motion has been detected in Serial Monitor
    Serial.print(motionDetected);
    Serial.print("motionDetected");
    Serial.println();
    delay(100);
  }


  delay(100); //wait for 100 milliseconds



  }

//converting microseconds to centimeters for ultrasonic sensor
```

```
long microsecondsToCentimeters(long microseconds) {

  //speed of sound is 340 m/s or 29 microseconds per cm.
  //the ping travels out and back, so to the find distance of the
  //object we take half of the distance traveled
  return microseconds / 29 / 2;
}
```