# CAPACITIVE SENSOR PROGRAM

// Pin Definitions

#define REFLECTIVEIR\_PIN\_COLL A3

#define REFLECTIVEIR\_PIN\_COLL\_M A5

// define vars for testing menu

const int timeout = 10000; //define timeout of 10 sec

char menuOption = 0;

long time0;

#include <Servo.h>

#define redpin 5

#define bluepin 6

#define greenpin 7

#define redpin\_M 8

#define bluepin\_M 9

#define greenpin\_M 10

int plasticsensor=A0;

int anothersensor=A2;

int metalsensor=A4;

int metalsensor\_M=A6;

Servo myservo; // create servo object to control a servo

Servo myservo\_M; // create servo object to control a servo

// twelve servo objects can be created on most boards

int pos = 165; // variable to store the servo position

int pos\_M = 158; // variable to store the servo position

void setup() {

myservo.attach(11); // attaches the servo on pin 9 to the servo object

myservo\_M.attach(12); // attaches the servo on pin 9 to the servo object

pinMode(redpin,OUTPUT);

pinMode(bluepin,OUTPUT);

pinMode(greenpin,OUTPUT);

//metal sensor

pinMode(redpin\_M,OUTPUT);

pinMode(bluepin\_M,OUTPUT);

pinMode(greenpin\_M,OUTPUT);

pinMode(plasticsensor,INPUT\_PULLUP);

pinMode(metalsensor,INPUT\_PULLUP);

pinMode(anothersensor,INPUT\_PULLUP);

pinMode(metalsensor\_M,INPUT\_PULLUP);

Serial.begin(9600);

analogWrite(redpin,255);

analogWrite(bluepin,0255);

analogWrite(greenpin,255);

analogWrite(redpin\_M,255);

analogWrite(bluepin\_M,0255);

analogWrite(greenpin\_M,255);

}

void loop() {

int sensor\_read=digitalRead(plasticsensor);

int sensor\_read\_m=digitalRead(metalsensor);

int sensor\_read\_mm=digitalRead(metalsensor\_M);

Serial.println("plastic sensor");

Serial.println(sensor\_read);

Serial.println("metal sensor");

Serial.println(sensor\_read\_m);

Serial.println(sensor\_read\_mm);

//Plastic bin

if((sensor\_read==0)&&(sensor\_read\_m!=1)){

for (pos = 160; pos >= 90; pos -= 1) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

myservo.write(pos); // tell servo to go to position in variable 'pos'

delay(1); // waits 15ms for the servo to reach the position

}

analogWrite(redpin,255);

analogWrite(bluepin,0);

analogWrite(greenpin,255);

delay(2500);

for (pos = 90; pos <= 160; pos += 1) { // goes from 180 degrees to 0 degrees

myservo.write(pos); // tell servo to go to position in variable 'pos'

delay(1); // waits 15ms for the servo to reach the position

}

analogWrite(redpin,0);

analogWrite(bluepin,0);

analogWrite(greenpin,0);

}

else{

// goes from 180 degrees to 0 degrees

myservo.write(pos); // tell servo to go to position in variable 'pos'

// waits 15ms for the servo to reach the position

analogWrite(redpin,255);

analogWrite(bluepin,255);

analogWrite(greenpin,255);

}

//metal sensor

if(sensor\_read\_mm==1){

for (pos\_M = 160; pos\_M >= 90; pos\_M -= 1) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

myservo\_M.write(pos\_M); // tell servo to go to position in variable 'pos'

delay(1); // waits 15ms for the servo to reach the position

}

analogWrite(redpin\_M,255);

analogWrite(bluepin\_M,0);

analogWrite(greenpin\_M,255);

delay(2500);

for (pos\_M = 90; pos\_M <= 160; pos\_M += 1) { // goes from 180 degrees to 0 degrees

myservo\_M.write(pos\_M); // tell servo to go to position in variable 'pos'

delay(1); // waits 15ms for the servo to reach the position

}

analogWrite(redpin\_M,0);

analogWrite(bluepin\_M,0);

analogWrite(greenpin\_M,0);

}

else{

// goes from 180 degrees to 0 degrees

myservo\_M.write(pos\_M); // tell servo to go to position in variable 'pos'

// waits 15ms for the servo to reach the position

analogWrite(redpin\_M,255);

analogWrite(bluepin\_M,255);

analogWrite(greenpin\_M,255);

}

}

# IR sensor

#include <Servo.h>

Servo tap\_Servo;

int sensor\_pin=4;

int tap\_Servo\_pin=5;

int val;

void setup(){

pinMode(sensor\_pin,INPUT);

tap\_Servo.attach(tap\_Servo\_pin);

}

void loop(){

val = digitalRead(sensor\_pin);

if (val ==0){

tap\_Servo.write(0);

}

if(val==1)

{

tap\_Servo.write(180);

}

}

# ULTRA SONIC SENSOR

#include <Servo.0h> //INCLUDES SERVO LIBRARY

Servo servo;

int trig = 5;

int echo = 6;

int servoPin = 9;

long duration, distance, average;

long aver[3];

void setup() {

Serial.begin(9600);

servo.attach(servoPin);

pinMode(trig, OUTPUT);

pinMode(echo, INPUT);

servo.write(0); //CLOSES CAP ON STARTING

delay(00);

servo.detach();

}

void measure() {

digitalWrite(trig, LOW);

delayMicroseconds(5);

digitalWrite(trig, HIGH);

delayMicroseconds(15);

digitalWrite(trig, LOW);

pinMode(echo, INPUT);

duration = pulseIn(echo, HIGH);

distance = (duration / 2) / 29.1; //CALCULATES DISTANCE

}

void loop() {

Serial.println(distance); //CAN BE DISABLED

for (int i = 0; i <= 2; i++) { //CALCULATES AVERAGE DISTANCE

measure();

aver[i] = distance;

delay(10);

}

distance = (aver[0] + aver[1] + aver[2]) / 3;

if ( distance <= 10) { //CHANGE AS PER AS NEED

servo.attach(servoPin);

delay(1);

servo.write(0);

delay(3500); //CHANGE AS PER AS NEED

servo.write(90);

delay(1500); //CHANGE AS PER AS NEED

servo.detach();

}

}