//Created by : Eng. Hossam Arafa

//www.youtube.com/arafamicrosystems

//www.fb.com/arafa.microsys

#define in1 2 //Right motor pole 1

#define in2 3 //Right motor pole 2

#define in3 4 //left motor pole 1

#define in4 7 //left motor pole 1

#define an A0

#define led 13

#define ab A1

#define led 12

float value=0;

unsigned long res=10;

const int pingPin = 8;//Trigger Pin of Ultrasonic

const int echoPin = 9;//Echo Pin of Ultrasonic

long duration;//used by ultrasonic Function

int distance,Cdistance,Rdistance,Ldistance;//Cdistance=center Distance Rdistance=Right Distance Ldistance=Left Distance

void setup()

{

Serial.begin(9600); // Starting Serial Terminal

pinMode(in1,OUTPUT);

pinMode(in2,OUTPUT);

pinMode(in3,OUTPUT);

pinMode(in4,OUTPUT);

Serial.begin(9600);

pinMode(led,OUTPUT);

pinMode(pingPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

}

void loop()

{

mark:

Cdistance=ultrasonic();//take ultrasonic distance value

if( Cdistance == 0 ){

goto mark;

}

Serial.println(Cdistance);

if(Cdistance>30)//if greater than 30 Cm

{

forward();//go to forward

}else

{

if(Cdistance<=10&&Cdistance>2)reverse();// if close to object reverse to get more space to change orientation

off(); //stop robot

delay(2500);

Rdistance=ultrasonic(); //get ultrasonic value at right

delay(2500);

Ldistance=ultrasonic(); //get ultrasonic value at left

delay(2500);

comparison(Rdistance, Ldistance); //go to Comparasion function

off(); // stop after execute Comparison function action

delay(150);

}

// reset all variables

Rdistance=0;

Ldistance=0;

Cdistance=0;

photoresistor\_an();

photoresistor\_ab();

}

int ultrasonic(void)//get distance captured by ultrasonic sensor

{

long duration, inches, cm;

pinMode(pingPin, OUTPUT);

digitalWrite(pingPin, LOW);

delayMicroseconds(2);

digitalWrite(pingPin, HIGH);

delayMicroseconds(10);

digitalWrite(pingPin, LOW);

pinMode(echoPin, INPUT);

duration = pulseIn(echoPin, HIGH);

cm = microsecondsToCentimeters(duration);

return cm;

}

long microsecondsToCentimeters(long microseconds) {

return microseconds / 29 / 2;

}

void comparison(int r, int l)//compare values of right and left Servo Pos

{

if(r>25||l>25)

{

if(r>l||r==l)

{

Tright90();

}else if(l>r)

{

Tleft90();

}

}else if(r<25&&l<25)

{

Tleft180();

}

}

void forward()

{

digitalWrite(in1,LOW);

digitalWrite(in2,HIGH);

digitalWrite(in3,LOW);

digitalWrite(in4,HIGH);

}

void off()

{

digitalWrite(in1,LOW);

digitalWrite(in2,LOW);

digitalWrite(in3,LOW);

digitalWrite(in4,LOW);

}

void Tleft90()

{

digitalWrite(in1,LOW);

digitalWrite(in2,HIGH);

digitalWrite(in3,HIGH);

digitalWrite(in4,LOW);

delay(650);

}

void Tright90()

{

digitalWrite(in1,HIGH);

digitalWrite(in2,LOW);

digitalWrite(in3,LOW);

digitalWrite(in4,HIGH);

delay(750);

}

void Tleft180()

{

digitalWrite(in1,LOW);

digitalWrite(in2,HIGH);

digitalWrite(in3,HIGH);

digitalWrite(in4,LOW);

delay(1500);

}

void reverse()

{

digitalWrite(in1,HIGH);

digitalWrite(in2,LOW);

digitalWrite(in3,HIGH);

digitalWrite(in4,LOW);

delay(450);

}

void photoresistor\_an(void)

{

value=analogRead(an);

value=(value/1023)\*5;

res=(1000\*value)/(5-value);

Serial.println(res);

if(res>=3000){

forward();

}

else{

off();

}

delay(100);

}

void photoresistor\_ab(void)

{

value=analogRead(ab);

value=(value/1023)\*5;

res=(1000\*value)/(5-value);

Serial.println(res);

if(res>=3000){

reverse();

}

else{

off();

}

delay(200);

}