int echoPin1 = 2; //2 olarak güncellenecek denge sol

int trigPin1 = 3; //3 olarak güncellenecek

int echoPin2 = 4; //denge sağ

int trigPin2 = 5;

int echoPin3 = 6; //6 olark güncellenecek Mesafe ölcen sensor sensör3

int trigPin3 = 7; //7 olarak güncellenecek

int echoPin4 = 8; //yeni sensör sensör 4

int trigPin4 = 9; //yeni sensör

int echoPin5 = 10;

int trigPin5 = 11;

int buzzer=13;

int merdiveninSayac = 0;

//bool isMerdiven;

bool denge;

//bool normal; //sensör 3 ün kabloları değişstirildei doğru mu kontrol et, güncel kodlar yüklenmedi

bool everThingIsGood;

bool merdivenIniyorum;

long times1;

long times2;

long times3;

long times4;

long times5;

long distance1;

long distance2;

long distance3;

long distance4;

long distance5;

void setup()

{

pinMode(buzzer,OUTPUT); // Buzzer'in bağlı olduğu pini çıkış olarak ayarladık.

digitalWrite(buzzer,HIGH); //açılışta buzzer uzun öter ve iki defa kısa Öter, ürünün doğru bir şekide açıldığını anlarız.

delay(1500);

digitalWrite(buzzer,LOW);

delay(1500);

digitalWrite(buzzer,HIGH);

delay(500);

digitalWrite(buzzer,LOW);

delay(500);

digitalWrite(buzzer,HIGH);

delay(500);

digitalWrite(buzzer,LOW);

delay(500);

pinMode(trigPin1, OUTPUT);

pinMode(echoPin1,INPUT);

pinMode(trigPin2, OUTPUT);

pinMode(echoPin2,INPUT);

pinMode(trigPin3, OUTPUT);

pinMode(echoPin3,INPUT);

pinMode(trigPin4, OUTPUT);

pinMode(echoPin4,INPUT);

pinMode(trigPin5, OUTPUT);

pinMode(echoPin5,INPUT);

Serial.begin(9600);

}

void loop()

{

// sensor 1

digitalWrite(trigPin1, LOW);

delayMicroseconds(2);

digitalWrite(trigPin1, HIGH);

delayMicroseconds(2);

digitalWrite(trigPin1, LOW);

times1 = pulseIn(echoPin1, HIGH);

distance1 = 0.0343 \* (times1/2);

Serial.print("Sensor1: Distance: ");

Serial.print(distance1);

Serial.println(" cm");

delayMicroseconds(2);

//sensor 2

digitalWrite(trigPin2, LOW);

delayMicroseconds(2);

digitalWrite(trigPin2, HIGH);

delayMicroseconds(2);

digitalWrite(trigPin2, LOW);

times2 = pulseIn(echoPin2, HIGH);

distance2 = 0.0343 \* (times2/2);

Serial.print("Sensor2: Distance: ");

Serial.print(distance2);

Serial.println(" cm");

delayMicroseconds(2);

// sensor 3

digitalWrite(trigPin3, LOW);

delayMicroseconds(2);

digitalWrite(trigPin3, HIGH);

delayMicroseconds(2);

digitalWrite(trigPin3, LOW);

times3 = pulseIn(echoPin3, HIGH);

distance3 = 0.0343 \* (times3/2);

Serial.print("Sensor3: Distance: ");

Serial.print(distance3);

Serial.println(" cm");

delayMicroseconds(2);

// sensor 4

digitalWrite(trigPin4, LOW);

delayMicroseconds(2);

digitalWrite(trigPin4, HIGH);

delayMicroseconds(2);

digitalWrite(trigPin4, LOW);

times4 = pulseIn(echoPin4, HIGH);

distance4 = 0.0343 \* (times4/2);

Serial.print("Sensor4: Distance: ");

Serial.print(distance4);

Serial.println(" cm");

delayMicroseconds(2);

// sensor 5 (engel sensörü)

digitalWrite(trigPin5, LOW);

delayMicroseconds(2);

digitalWrite(trigPin5, HIGH);

delayMicroseconds(2);

digitalWrite(trigPin5, LOW);

times5 = pulseIn(echoPin5, HIGH);

distance5 = 0.0343 \* (times1/2);

Serial.print("Sensor5: Distance: ");

Serial.print(distance5);

Serial.println(" cm");

delayMicroseconds(2);

isDenge();

merdiven();

merdivenC();

meow3();

//cukur();

//meow();

//normal();

//meow2();

//test();

}

void isDenge()

{

if(distance1 - distance2 >=-10 && distance1 - distance2 <=10) // true

{

denge = true;

}

else

{

denge = false;

digitalWrite(buzzer,HIGH);

delay(50);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,HIGH);

delay(50);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,LOW);

}

}

/\*void normal() //yeni eklendi

{

if(denge==true)

{

if(distance3 > 30 && distance3 < 40)

{

if(distance4 > 20 && distance4 < 30)

{

everThingIsGood = true;

}

}

}

}

void meow2() //indimCıktım //yeni eklendi

{

if(merdivenIniyorum == true && everThingIsGood == true) // merdiven çıkıyorum için de aynısı yazılacak

{

merdivenIniyorum = false;

for (int i = 0; i <= 1; i++)

{

digitalWrite(buzzer,HIGH);

delay(50);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,HIGH);

delay(50);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,LOW);

}

}

}\*/

void meow3()

{

if(distance5 < 60)

{

digitalWrite(buzzer,HIGH);

delay(50);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,LOW);

}

}

void merdiven() //iniş

{

if (denge==true)

{

/\*if(distance3 >= 100 && distance4 <= 30) //test edilecek //yeni eklendi //100 && distance4 <= 30 normal değer olarak atandı

{

for (int i = 0; i <= 1; i++)

{

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,LOW);

}

}\*/

/\*else\*/ if(distance3 >= 100 && distance4 < 18) //distance 3 4 olarak güncellenecek //35 idi 40 yaptım //20, 17 oldu sensör norm 25 cm olması gerek, 40 => 100, 17 => 25

{

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,LOW);

merdivenIniyorum = true;

merdiveninSayac = 0;

}

else if(distance3 >= 100 && distance4 >21 && distance4 < 29)

{

if(merdiveninSayac < 3)

{

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,LOW);

merdiveninSayac++;

}

}

}

}

void merdivenC()

{

if (denge==true)

{

if(distance3 <= 65 && distance4 > 35) //distance 3 4 olarak güncellenecek //35 idi 40 yaptım//disance4 20 =>30 // distance3 e yukarıya doğru küçük bir eğim verildi, 40 => 60, 30 => 55

{

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,LOW);

merdiveninSayac = 0;

}

}

}

void cukur()

{

if(denge==true)

{

if(distance3 > 75 && distance4 < 30 && distance4 >20 ) //yeniden ayarlanacak 40 => 50 , 50 >= 75 //&& distance4 < 30 && distance4 >20 eklendi

{

digitalWrite(buzzer,HIGH);

delay(350);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,HIGH);

delay(150);

digitalWrite(buzzer,LOW);

delay(150);

digitalWrite(buzzer,LOW);

}

}

}

void meow() //cıkıntı fonksyon adı almıyor

{

if(denge==true)

{

if(distance3 < 45) //sensöre yukarıya doğru eğim verdik // so3 den baston ucuna mesafe 50 cm// 30 => 45 // eğimi arttırıp 40 yada 35 cm yapmalı

{

digitalWrite(buzzer,HIGH);

delay(1000);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,HIGH);

delay(100);

digitalWrite(buzzer,LOW);

delay(50);

digitalWrite(buzzer,LOW);

}

}

}

void engel()

{

// sadece diz üzeri ve göhüs arası engelleri algılamak için tasarlanır.

}

/\*

void test()

{

if(distance3 > 100)

{

if (son\_zaman-ilk\_zaman>=150)

{

digitalWrite(buzzer,HIGH);

ilk\_zaman=son\_zaman;

if(son\_zaman-ilk\_zaman>=150);

{

digitalWrite(buzzer,LOW);

ilk\_zaman=son\_zaman;

}

}

}

}

\*/

//şu anda 30-40 cm arası normal kabul ediliyor(distance3-sensör3). sensör4 sadece merdiven için kullanılıyor.

//çıkıntı ve çukurları algılayabiliyor. ayrıca cihazın dengede olup olmadığını da kontrol ediyor.