# Şehit Uzman Çavuş Harun Şenözüar

# Mesleki ve Teknik Anadolu Lisesi

# Muhammed Can Karabulut 2348

# Sadettin Ünsal 166

# 10/A

# Arduino İle Engelden Kaçan Robot Yapımı

# İçindekiler

Proje amacı +tanımı…………………………………………………………………………………………….2

Malzeme Listesi…………………………………………………………………………………………………2

Projede kullanılan malzemelerin kullanış amaçları………………………………………………………3

Projenin Devre şeması…………………………………………………………………………………………4

Projenin adım adım anlatımı resimler…………………………………………………………………….

Projenin Kodları ve açıklaması………………………………………………………………………………

Kaynakça……………………………………………………………………………………………

Projenin Linkleri…………………………………………………………………………………………….

1

# amacı + tanımı

Amacı;Engelden kaçan robot sitemi günümüzde birçok alanda kullanılmaktadır.

Bunu motorlu taşıtlarda örneğin seyir halinde engel algılama,günlük yaşantımızda

Örneğin mobilyalara çarpmadan gezinen elektrik süpürgesi veya askeri alanda

arazi şartlarında gezinen robotun engel(mayın,taş blok vb.)algılamasında kullanılmaktadır

Engelden kaçan robot, diğer adıyla engel algılayan robot, otonom olarak çevre kontrolü yapabilen ve hareketini önleyebilecek cisimleri atlatabilen robot tipidir.

Çevre kontrolünü sağlayabilmesi için ultrasonik, kızılötesi vb. gibi çeşitli sensörlere ihtiyaç duyar.

Tanımı; Engelden Kaçan Robot Nedir? Engelden kaçan robot, üzerindeki sensör yada sensörlerle çevresini kontrol ederek, hareket halindeki aracın bu engellere çarpmasını engeller. Yazılımınıza göre kendi yolunu belirleyerek engellere çarpmadan yoluna devam eder. Hc-sr04 yada Mz80 gibi sensörler bu proje için idealdir.

Malzeme Listesi

* **1-Arduino Uno**
* **2-Çok amaçlı robot Platformu**
* **3-l298N voltaj regulatörlü çift motor sürücü kartı**
* **hc-sr04 ultrasonik mesafe sensörü**
* **Pil**
* **6 lı aa pil yuvası**
* **Jumper**

2

# Projede kullanılan malzemlerin kullanış amaçları

1-Arduino uno

Arduino / Uno, bir bilgisayar, başka bir Arduino/Genuino kartı veya diğer mikrodenetleyicilerle iletişim kurmak için bir dizi özelliğe sahiptir. Atmega328, 0 (RX) ve 1 (TX) dijital pinlerinde bulunan UART TTL (5V) seri iletişimi sağlamaktadır.

2-çok amaçlı robot platformu

2WD Çok Amaçlı Mobil Robot Platformu iki adet dc motor ile bir adet sarhoş tekerlek bağlamak kaydı ile hareket sağlayabilen bir mobil robot platformudur.

**3-l298N voltaj regulatörlü çift motor sürücü kartı**

L298N Voltaj Regulatörlü Çift Motor Sürücü Kartı, 24V'a kadar olan motorları sürmek için hazırlanmış olan bu dc ve step motor sürücü kartı, iki kanallı olup, kanal başına 2A akım vermektedir.

**4-**hc-sr04 ultrasonik mesafe

 sonar iletişim kullanıp karşısında yer alan nesneye olan mesafeyi hesaplamakta olan bir kaynak olmaktadır.

5-**6 lı aa pil yuvası**

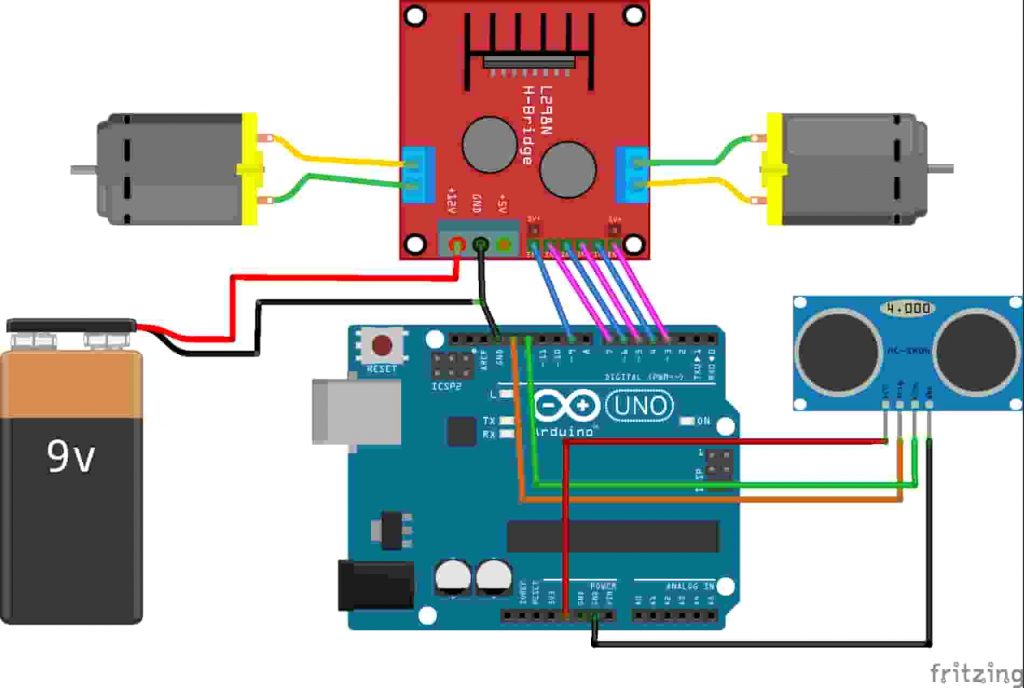
6 Adet pil alabilen pil yuvasıdır

**6-Jumper**

Bağlantı Kablosu

3

# Proje devre şeması

****

4

# PROJE KODLARI

#define echoPin 12 //Ultrasonik sensörün echo pini

#define trigPin 13 //Ultrasonik sensörün trig pini

#define MotorR1 7

#define MotorR2 6

#define MotorRE 9 // Motor pinlerini tanımlıyoruz.

#define MotorL1 5

#define MotorL2 4

#define MotorLE 3

long sure, uzaklik; //Süre ve uzaklık diye 2 değişken tanımladık

void setup(){

// ultrasoniksensör Trig pininnden ses dalgaları gönderdiği için OUTPUT (Çıkış),

// bu dalgaları Echo pini ile geri aldığı için UNPUT (giriş) olarak tanımlanır.

pinMode(echoPin,INPUT);

pinMode(trigPin,OUTPUT);

pinMode(MotorL1,OUTPUT);

pinMode(MotorL2,OUTPUT);

pinMode(MotorLE,OUTPUT);//Motorlarımızı Çıkış Olarak tanımlıyoruz

pinMode(MotorR1,OUTPUT);

pinMode(MotorR2,OUTPUT);

pinMode(MotorRE, OUTPUT);

Serial.begin(9600);

}

void loop() {

digitalWrite(trigPin, LOW); //sensör pasif hale getirildi

delayMicroseconds(5);

digitalWrite(trigPin, HIGH); //Sensore ses dalgasının üretmesi için emir verildi

delayMicroseconds(10);

digitalWrite(trigPin, LOW); //Yeni dalgaların üretilmemesi için trig pini LOW konumuna getirildi

sure = pulseIn(echoPin, HIGH); //ses dalgasının geri dönmesi için geçen sure ölçülüyor

uzaklik = sure / 29.1 / 2; //ölçülen süre uzaklığa çevriliyor

Serial.println(uzaklik);

if (uzaklik < 15)// Uzaklık 15'den küçük ise,

{

geri();// 150 ms geri git

delay(150);

sag();// 250 ms sağa dön

delay(250);

}

else {// değil ise,

ileri();// ileri git

}

}

void ileri(){ // Robotun ileri yönde hareketi için fonksiyon tanımlıyoruz.

digitalWrite(MotorR1, HIGH);// Sağ motorun ileri hareketi aktif

digitalWrite(MotorR2, LOW);// Sağ motorun geri hareketi pasif

analogWrite(MotorRE, 150);// Sağ motorun hızı 150

digitalWrite(MotorL1, HIGH);// Sol motorun ileri hareketi aktif

digitalWrite(MotorL2, LOW);// Sol motorun geri hareketi pasif

analogWrite(MotorLE, 150);// Sol motorun hızı 150

}

void sag(){ // Robotun sağa dönme hareketi için fonksiyon tanımlıyoruz.

digitalWrite(MotorR1, HIGH);// Sağ motorun ileri hareketi aktif

digitalWrite(MotorR2, LOW);// Sağ motorun geri hareketi pasif

analogWrite(MotorRE, 0);// Sağ motorun hızı 0 (Motor duruyor)

digitalWrite(MotorL1, HIGH);// Sol motorun ileri hareketi aktif

digitalWrite(MotorL2, LOW);// Sol motorun geri hareketi pasif

analogWrite(MotorLE, 150);// Sol motorun hızı 150

}

void geri(){ // Robotun geri yönde hareketi için fonksiyon tanımlıyoruz.

digitalWrite(MotorR1, LOW);// Sağ motorun ileri hareketi pasif

digitalWrite(MotorR2, HIGH);// Sağ motorun geri hareketi aktif

analogWrite(MotorRE, 150);// Sağ motorun hızı 150

digitalWrite(MotorL1, LOW);// Sol motorun ileri hareketi pasif

digitalWrite(MotorL2, HIGH);// Sol motorun geri hareketi aktif

analogWrite(MotorLE, 150);

}

Link https://github.com/Can35710/Proje-Kodlar.git