

T.C.
SELÇUK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
ELEKTRİK – ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

ELEKTRİK-ELEKTRONİK TASARIM

HAFIZALI ROBOT

Hazırlayan

Ahmed Mohamed ALNAZEER ALI

141202103

Danışman

DOKTOR ÖĞRETİM ÜYESİ AKİF DURDU

Konya – 2018

ÖZGEÇMİŞ

19.08.1996 tarihinde SUDAN 'da doğdu. İlk , orta ve lise öğrenimini SUDAN 'da tamamlayarak. Selçuk Üniversitesi Mühendislik Fakültesi Elektrik – Elektronik Mühendisliği Bölümü 'ne girdi. Halen bu bölümde öğrenimini sürdürmektedir.

TEŞEKKÜR

Bütün çalışmalarında değerli bilgi ve tecrübeleriyle bana yol gösteren, gerekli araştırma ve geliştirme çabalarında yardımlarını esirgemeyen danışmanım DOKTOR ÖĞRETİM ÜYESİ AKİF DURDU 'ya ve bölümümüzün değerli tüm öğretim elemanlarına, her türlü maddi manevi katkılarını esirgemeyen aileme ve öğrenci arkadaşlarıma teşekkür ederim.

ÖZET

Mikrodenetleyiciler endüstriyel uygulamalarda yaygın biçimde kullanılmaktadır. Daha önce kullanılmakta olan mikroişlemcilerin yerini programlama ve kullanım kolaylıklarından dolayı mikrodenetleyiciler almıştır.

Ayne zamanda robotlar hayatımızdan büyük bir parça oldu ve insanoğlu robotları her sektörde kullanmaya başladı çünkü robotlar yorulmuyor yemek , ara ve maaşa ihtiyacı yoktur yani daha ekonomiktir bunu için insanlar robotları daha fazla kullanmaya başladı bizim projemizde insanlığa yeni bir şekilde yardım eder , mesela şehir otobüsleri her gün ayne yoldan geçer ve ayne duraklarda durur O zaman niye her otobüsün başında bir şoför koyuruz ? yani otobüsü bir şoföre sürdürmektansa otubusu geçeceği yolları ve duracağı durakları izbarlatıp kandi kandise bu işi haledemez mi ?

Bu Projede biz bir robotu ilk başta yolu gösterip veya hata kandise engelden kaçarak yolu bulup ve bu yolu ezberleyip ve ayne yoldan geçer

İÇİNDEKİLER

Sayfa

ÖZGEÇMİŞ, TEŞEKKÜR.....	I
ÖZET.....	II
İÇİNDEKİLER.....	III
1.KULANILAN MAZLZEMELER.....	1
2.ÇALIŞMA ŞEKLİ.....	7
3.KULANILAN KOD.....	8
4. SONUÇLAR.....	12
5.KAYNAKLAR.....	12
6.EKLER.....	13

1) KULANILAN MAZLZEMELER :

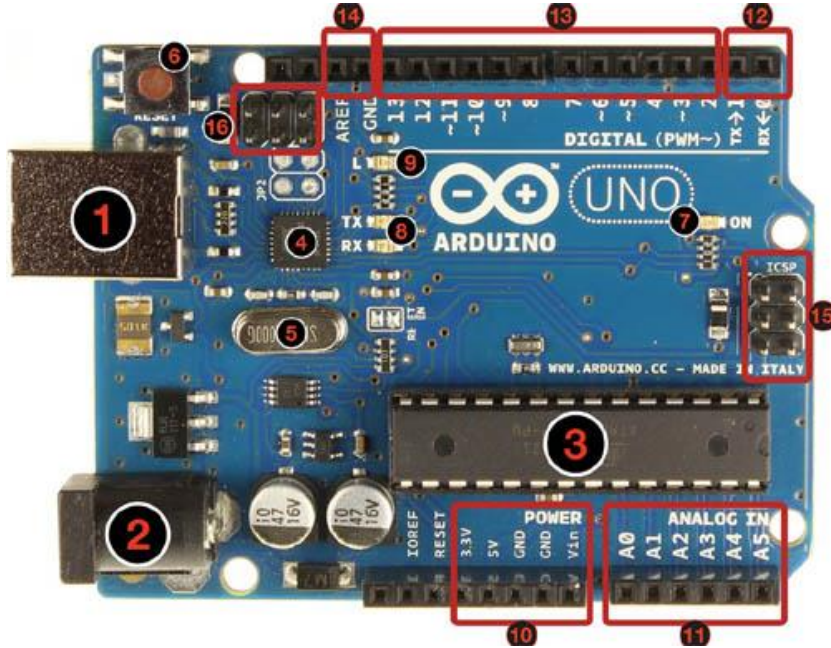
(1.1) Arduino UNO

(1.1.1) Genel bilgi :

ATmega328 mikrodnetleyici ieren bir Arduino kartıdır. Arduino 'nun en yaygın kullanılan kartı olduėu ylenebilir. Arduino Uno 'nun ilk modelinden sonra Arduino Uno R2, Arduino Uno SMD ve son olarak Arduino Uno R3 ıkmıřtır. Arduino 'nun kardeř markası olan **Genuino** markasını taşıyan **Genuino Uno** kartı ile tamamen aynı zelliklere sahiptir.

Arduino Uno 'nun 14 tane dijital giriř / ıkıř pini vardır. Bunlardan 6 tanesi PWM ıkıřı olarak kullanılabilir. Ayrıca 6 adet analog giriři, bir adet 16 MHz kristal osilatr, USB baėlantısı, power jakı (2.1mm), ICSP bařlıėı ve reset butonu bulunmaktadır. Arduino Uno bir mikrodnetleyiciyi desteklemek iin gerekli bileřenlerin hepsini ierir. Arduino Uno 'yu bir bilgisayara baėlayarak, bir adaptr ile ya da pil ile alıřtırabilirsiniz. Ařaėıdaki resimde Arduino Uno R3 'n kısımları gsterilmektedir.

(1.1.2) Arduino paraları :



(řekil (1.1))

- 1 : USB jakı
- 2 : Power jakı (7-12 V DC)
- 3 : Mikrodenetleyici ATmega328
- 4 : Haberleşme çipi
- 5 : 16 MHz kristal
- 6 : Reset butonu
- 7 : Power ledi
- 8 : TX / NX ledleri
- 9 : Led
- 10 : Power pinleri
- 11 : Analog girişler
- 12 : TX / RX pinleri
- 13 : Dijital giriş / çıkış pinleri (yanında ~ işareti olan pinler PWM çıkışı olarak kullanılabilir.)
- 14 : Ground ve AREF pinleri
- 15 : ATmega328 için ICSP
- 16 : USB arayüzü için ICSP

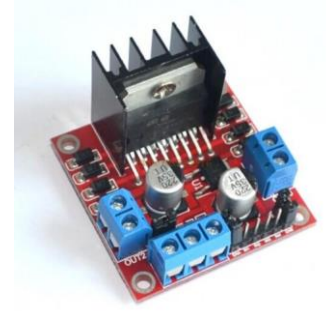
(1.1.3) Arduino Uno Teknik Özellikleri

- Mikrodenetleyici : ATmega328
- Çalışma gerilimi : +5 V DC
- Tavsiye edilen besleme gerilimi : 7 - 12 V DC
- Besleme gerilimi limitleri : 6 - 20 V
- Dijital giriş / çıkış pinleri : 14 tane (6 tanesi PWM çıkışını destekler)
- Analog giriş pinleri : 6 tane
- Giriş / çıkış pini başına düşen DC akım : 40 mA
- 3,3 V pini için akım : 50 mA
- Flash hafıza : 32 KB (0.5 KB bootloader için kullanılır)

- SRAM : 2 KB
- EEPROM : 1 KB
- Saat frekansı : 16 MHz stal osilatörü, USB bağlantısı, power jakı (2.1mm), ICSP başlığı ve reset butonu bulunmaktadır. Arduino Uno bir mikrodenetleyiciyi desteklemek için gerekli bileşenlerin hepsini içerir. Arduino Uno 'yu bir bilgisayara bağlayarak, bir adaptör ile ya da pil ile çalıştırabilirsiniz. Aşağıdaki resimde Arduino Uno R3 'ün kısımları gösterilmektedir.

(1.2) L298n

24V'a kadar olan motorları sürmek için hazırlanmış olan bu motor sürücü kartı, iki kanallı olup, kanal başına 2A akım vermektedir. Kart üzerinde L298N motor sürücü entegresi kullanılmıştır. Sumo, mini sumo, çizgi izleyen robotlarda ve çok çeşitli motor kontrol uygulamalarında kullanılabilir.



Şekil (1.2)

DC motorlardan ayrı olarak step motor kontrolüne de imkan sağlamaktadır. Daha ayrıntılı bilgi için aşağıdaki örnek dokümanı inceleyebilirsiniz.

Özellikleri:

- Birbirinden bağımsız olarak iki ayrı motoru kontrol edebilir.
- Kanal başına 2A akım verebilmektedir.
- Üzerinde dahili regülatörü vardır.
- Yüksek sıcaklık ve kısa devre koruması vardır.
- Motor dönüş yönüne göre yanan ledler vardır.
- Kart üzerinde dahili soğutucu vardır.
- Akım okuma (current sense) pinleri dışa verilmiş haldedir.
- Kartın 4 yanında istenilen yüzeye sabitleyebileceğiniz 4 adet vida deliği bulunmaktadır.

(1.3) 2 adet DC motor

Robotikte en sık tercih edilen motor tipi DC motorlardır. DC motorlar ucuz, küçük ve etkilidir. Ayrıca boyut, şekil ve güç bakımından çok çeşitli olmaları da DC motorların sık kullanılmalarının bir diğer sebebidir.

DC motorlar robotlarda veya herhangi bir sistemde direkt ya da dişli kutularıyla (redüktörlü ya da redüktörsüz olarak) birlikte kullanılabilirler. DC motorların robotlarda kullanımına dair temel özellikler aşağıda açıklanmıştır.



Şekil (1.3)

(1.4) HC-SR04

HC-SR04, robotik projelerde Arduino ile kullanılan en popüler sensörlerden birisidir. Kullanımı oldukça kolaydır ve program kısmı düzgün olduğu sürece 2cm – 400cm arası uzaklıkları düzgün bir şekilde ölçebilmektedir. Çalışma prensibi ise şu şekildedir: Sensörün Trig pininden uygulanan sinyal 40 kHz frekansında ultrasonik bir ses yayılmasını sağlar. Bu ses dalgası herhangi bir cisme çarpıp sensöre geri döndüğünde, Echo pini aktif hale gelir. Biz ise bu iki sinyal arasındaki süreyi ölçerek -yani sesin yankısını algılayarak- cismin sensörden uzaklığını tespit edebiliriz.



Şekil (1.4)

(1.5) Servo motor

FS90R standart servo pozisyonlarının sinyallerini sürekli dönüş hızına dönüştürür. Bu motorun 1.5 ms olarak belirlenmiş bir duruş pozisyonu vardır. Bu süre motor üzerinde bulunan potansiyometreden arttırılıp azaltılabilir. Motora duruş pozisyon



Şekil (1.5)

süresinin üzerindeki bir genişliğe sahip sinyal yollandığında saat yönünün tersine, bu sürenin altında genişliğe sahip bir sinyal yollandığında ise saat yönünde dönüş gerçekleştirilecektir. Sinyal genişliğinin artmasıyla dönüş hızı da yükselecektir.

(1.6) Araba şasesi

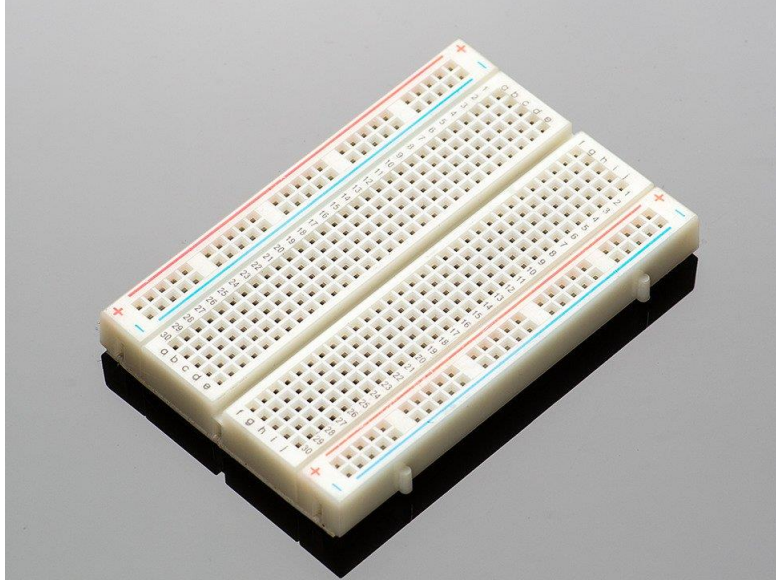
Şekil (1.6) daki araba şasesi Ali express'ten alındı



Şekil (1.6)

(1.7) Breadbord

8.5CM x 5.5CM 400 Hücreli Bread Board kullanıldı



Şekil (1.7)

(1.8) Güç kaynağı

8 adet 1.5 V luk (Toplam 12 v motor için) piller ve 9v luk (Arduino için) kullanıldı

2) ÇALIŞMA ŞEKLİ :

Robot ilk başta ileriye gider ve ultrasonic'le mesafe ölçer yakın bir engel gördüğü zaman durup servo motor sağ ve sola hareket eder , sağ ve sola baktığı zaman her taraftaki mesafeyi ölçer , uzak engel olan tarafı belirler ve O yöne hareket eder .

Aynı zamanda 2 adet dizi (yön ve zaman) bulunuyor , O dizinin boyu ezberlenecek adımların sayısıdır Robot ilk başta ileriye giderken zaman süresini sayıp ve 'zaman' dizisine ekler , Robot bir tarafa döndüğü zaman 'yön' dizisinde yazar , Adım sayısı bitince durup ve kaydettiği belgilere göre baştan hareket etmeye başlar .

3) KULANILAN KOD :

```
#include <Servo.h>

int start = 3 ;
int trigPin = 13;
int echoPin = 12;
int MotorR = 5;
int MotorL = 3;
int sag=0 ;
int sol=0 ;
int pos = 0;
int i = 0 ;
String yon[3] ;
int a = 0 ;
unsigned long zaman[3];
int b = 0 ;
double time1 ;
double time2 ;
Servo myservo;
void setup() {
  myservo.attach(11);
  Serial.begin(9600);
  pinMode(echoPin, INPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(MotorL, OUTPUT);
  pinMode(MotorR, OUTPUT);}
void loop() {
  myservo.write(90);
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
```

```

digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
if (i>=start)
{ dur () ;
  delay(5000);
  for (int say=0 ; say < start ;say++){
    ileri();
    delay(zaman[say]);
    dur () ;
    delay(1000);
    if (yon [say]== "sol" ) { SOL();
      delay(900);
      dur();
      delay (500); }
    else { SAG();
      delay(900);
      dur();
      delay (500);} } }
    else if (distance > 20 ) {
      ileri();}
    else {
      dur();
      myservo.write(0);
      delay (500);
      duration, distance;
      digitalWrite(trigPin, LOW);
      delayMicroseconds(2);
      digitalWrite(trigPin, HIGH);

```

```

delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
sag=distance ;
delay (100);
myservo.write(180);
delay (500);
duration, distance;
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
sol=distance ;
delay (100);
myservo.write(90);
if (sol > sag){
    SOL();
    delay(900);
    dur();
    delay (500);
    yon[a]="sol";}
else {
    SAG();
    delay(900);
    dur();
    delay (500);
    yon[a]="sag"; }

```

```

        i= i+1 ;
        time1 = millis () ;
        Serial.println (yon[a]);
        time2 = ((time1/(b+1))-3000) ;
        zaman[b]=time2;
        Serial.println (zaman[b]);
        a++ ;          b++ ;}
    delay(500); }
void ileri(){
    analogWrite(MotorR, 125);
    analogWrite(MotorL, 100); }
void dur(){
    analogWrite(MotorR, 0);
    analogWrite(MotorL, 0); }
void SAG(){
    analogWrite(MotorR, 0);
    analogWrite(MotorL, 110);}
void SOL(){
    analogWrite(MotorR, 100);
    analogWrite(MotorL, 0); }

```

kodu github'ta aşağıdaki linkten ulaşabilirsiniz :

<https://github.com/raclab/RACLAB/blob/master/Lisans%20Projeleri/hafizali%20robot>

4) SONUÇLAR:

Robotu 2 Dc motordan oluştuğu için robotun dengeli bir şekilde sürmesi zor oldu bunun için robotu ileriye gitiğinde bir terafta daha hafif bir yük varsa O tarafa harket eder .

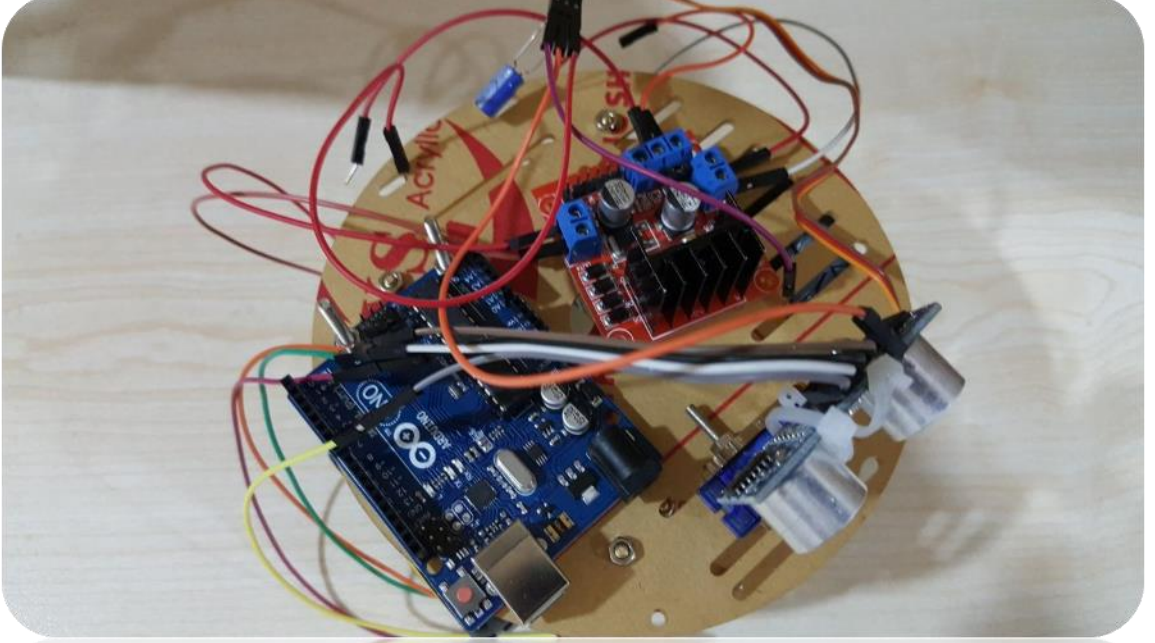
Sonuçta robotu istediğimiz şekilde çalıştı ve 3 tane adım ezberler ondan sonra 10 saniye bekleyip aynen yoldan geçer ve kod'ta (Start)'ın değeri değiştirerek kaç adım ezeberleteceğimizi ayarlarız .

5) KAYNAKLAR :

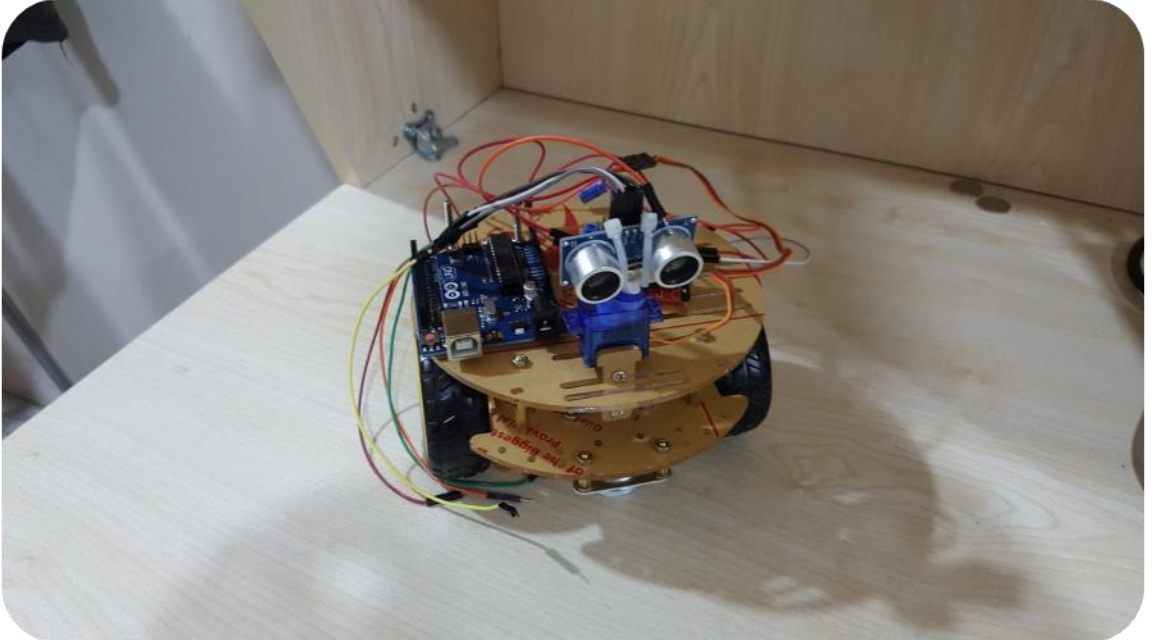
- ✓ Derinlemesine Arduino
- ✓ Udeemy Tech Explorations Arduino step by step kursu
- ✓ www.robotistan.com
- ✓ <http://www.robotiksistem.com>

6) EKLER :

Ropotun fotğrafları aşğıdaki şekillerde bulunur



Şekil (6.1)



Şekil (6.2)