



Ardunio İle Üzerine Atılan Cisimden Kaçan Robot

Escaping block robot with Arduino

İdris IŞIK*, İbrahim ÇAYIROĞLU**

*Karabük Üniversitesi, Mühendislik Fakültesi, Mekatronik Mühendisliği, 78050, Karabük, iisik3706@gmail.com

**Karabük Üniversitesi, Mühendislik Fakültesi, Mekatronik Mühendisliği, 78050, Karabük, icayiroglu@yahoo.com

Anahtar Kelimeler:

Engel Algılama, Engelden Kaçma

Özet: Bu makalede Teknolojinin ilerlemesiyle robotik sistemler günlük yaşamımızın bir parçası haline gelmiştir. Bununla birlikte ortaya çıkan fonksiyonellik sorunlarından biride engel algılama, engelden kaçma veya engel aşma özellikleridir.

Engelden kaçan robot sistemi günümüzde birçok alanda kullanılmaktadır. Bunu motorlu taşıtlarda örneğin seyir halinde engel algılama, günlük yaşamımızda örneğin mobilyalara çarpmadan gezinen elektrik süpürgesi veya askeri alanda arazi şartlarında gezinen robotun engel(mayın, taş blok vb.) algılamasında kullanılmaktadır.

Ardunio ile üzerine atılan cisimden kaçan robot tasarım projesi iki aşamadan oluşmuştur. Donanım kısmı; servo motorlar, sensör ve mikrodenetleyici ve yazılım kısmı, robotun karşılaştığı engellerle belirlenen algoritmalarla karşılık vermesi için arduunionun yazılımının yazılması olarak ayrılmıştır.

Keywords:

Robotic systems, Detecting block

Abstract: Robotic systems is a part of our casual life with improving technology. According to this which has some of functionality problems are detecting block, escape block and overcoming block.

Nowadays, escape block robotic systems is using wide area. In motor vehicles, for example, cruising detect block. For example in daily life, moving vacuum cleaners without collision with furniture. Or moving robot is detecting block or any danger (rock, mine etc.) in military area conditions.

Escaping block robot with Arduino design project has two steps. Hardware part; servo motors, sensor and microcontroller and software part, the blocks which is detecting block algorithms written in arduino software.

©2015 ibrahimcayiroglu.com, All rights reserved. Bu makale hakem kontrolünden geçmeden bilgi paylaşımı amacıyla yayınlanan bir dökümandır. Olusabilecek hata ve yanlışlıklardan dolayı sorumluluk kabul edilmez. Makaledeki bilgiler referans gösterilip yayınlanabilir. {These articles are published documents for the purpose of information sharing without checked by the referee. Not accepted responsibility for errors or inaccuracies that may occur. The information in the article can be published by referred. }

1. Giriş

1.1 Arduino

Ardunio bir G/Ç kartı ve Processing/Wiring dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur. Ardunio tek başına çalışan interaktif nesneler geliştirmek için kullanılabileceği gibi bilgisayar üzerinde çalışan yazılımlara da (Macromedia Flash, Processing, Max/MSP, Pure Data, SuperCollider gibi) bağlanabilir.

Hazır üretilmiş kartlar satın alınabilir veya kendileri üretmek isteyenler için donanım tasarımı ile ilgili bilgiler mevcuttur.

Donanımsal açıdan arduino kartları bir Atmel AVR mikrodenetleyici (Eski kartlarda ATmega8 veya ATmega168, yenilerinde ATmega328) ve programlama ve diğer devrelere bağlantı için gerekli yan elemanlardan oluşur. Her kartta en azından bir 5 voltluk regüle entegresi ve bir 16 MHZ kristal osilator (bazılarında seramik rezonatör) bulunur.

Mikrodenetleyiciye önceden bir bootloader programı yazılı olduğundan programlama için harici bir programlayıcıya ihtiyaç duyulmaz.

Yazılım açısından Arduino IDE kod editörü ve derleyici olarak görev yapan, aynı zamanda derlenen programı karta yükleme işlemini de yapabilen, her platformda çalışabilen Java programlama dilinde yazılmış bir uygulamadır.

Geliştirme ortamı, sanatçıları programlamayla tanıştırmak için geliştirilmiş Processing yazılımından yola çıkılarak geliştirilmiştir.

Arduino donanım referans tasarımları Creative Commons Attribution Share-Alike 2.5 lisansı ile dağıtılmaktadır ve Arduino web sitesinden indirilebilir.

Bazı Arduino donanımları için yerleşim ve üretim dosyaları da mevcuttur. Geliştirme ortamının kaynak kodu ve Arduino kütüphane kodları GPLv2 lisansı ile lisanslanmıştır.

1.2. Servo Motor

Servo, herhangi bir mekanizmanın işleyişini hatayı algılayarak yan bir geri besleme düzeneğinin yardımıyla denetleyen ve hatayı gideren otomatik aygıttır. Robot teknolojisinde en çok kullanılan motor çeşididir. Bu sistemler mekanik olabileceği gibi elektronik, hidrolik-pnömatik veya başka alanlarda da kullanılabilir. Servo motorlar da çıkış; mekaniksel konum, hız veya ivme gibi parametrelerin kontrol edildiği bir düzendir. Servo motor içerisinde herhangi bir motor AC, DC veya Step motor bulunmaktadır. Ayrıca sürücü ve kontrol devresini de içerisinde barındırmaktadır.

D.C servo motorları, genel olarak bir D.C. motoru olup, motora gerekli D.C. aşağıdaki metotlardan elde edilir.

1. Bir elektrik yükselteçten.
2. A.C. akımın doyumlu reaktörden geçirilmesinden.
3. A.C. akımın tristörden geçirilmesinden.
4. Amplidin, retotrol, regüleks gibi dönel yükselteçlerden elde edilir.

D.C. servo motorlar çok küçük güçlerden çok büyük güçlere kadar imal edilirler(0,05 Hp'den 1000 Hp'ye kadar). Bu motorlar klasik D.C. motorlar gibi imal edilirler. Bu motorlar küçük yapıdadır ve endüvileri (yükseklik, uzunluk / Çap oranıyla) kutup atalet momentini minimum yapacak şekilde tasarlanırlar. Küçük çaplı ve genellikle içerisinde kompanzasyon sargısı olan, kuvvetli manyetik alanı boyu uzun doğru akım motorlarına da servo motor denir. D.C. servisi motor çalışma prensibi açısından aslında, Statoru Daimi Mıknatıs bir D.C. motordur.

Manyetik alan ile içinden akım geçirilen iletkenler arasındaki etkileşim nedeniyle bir döndürme momenti meydana gelir. Bu döndürme momenti manyetik alan vektörü ile sargı akım vektörü

arasındaki açı 90° olduğunda maksimum değerin alınır.

Bir D.C. servo motorda fırçaların konumları, her iki dönüş yönü için de döndürme momenti açısının 90° olmasını sağlayacak şekilde belirlenmiştir. Kolektör segmentlerinin fazla olması neticesinde momentin sıfır bir noktada rotorun hareketsiz kalması engellenmiş olur.

Sanayide kullanılan çeşitli doğru akım motorları vardır. Servo sistemlerde kullanılan doğru akım motorlarına ise D.C. servo motorlar adı verilir. D.C. servo motorlarda rotor eylemsizlik momenti çok küçüktür. Bu sebepten piyasada çıkış momentinin eylemsizlik momentine oranı çok büyük olan motorlar bulunur.

Bazı D.C. servo motorların çok küçük zaman sabitleri vardır. Düşük güçlü D.C. servo motorlar piyasada genellikle bilgisayar kontrollü cihazlarda (disket sürücüler, teyp sürücüler, yazıcılar, kelime işlemciler, tarayıcılar vs.) kullanılırlar. Orta ve büyük güçlü servo motorlar ise sanayide genellikle robot sistemleri ile sayısal denetimli hassas diş açma tezgâhlarında kullanılır. D.C. motorlarda alan sargıları rotor sargılarına seri veya paralel bağlanır. Endüvi sargılarından bağımsız olarak uyarılan alan sargılarının akısı endüvi sargılarından geçen akımın fonksiyonu değildir.

Bazı D.C. motorlarda manyetik akı sabittir. Uyarma sargıları endüviden bağımsız olan veya sabit mıknatısla uyarılan motorlarda hız kontrolü endüvi gerilimi ile yapılabilir. Bu tip kontrol yöntemine endüvi kontrol yöntemi denir.

Uyarma sargılarının oluşturduğu akı ile yapılan denetlemede ise endüvi akımı sabit tutulur. Statorda bulunan uyarım sargılarının oluşturduğu akımın kontrolü ile hız ayarlanır. Bu tip motorlara alan kontrollü motorlar denir. Fakat rotor sargılarından geçen akımın sabit tutulabilmesi ciddi bir problemdir. Zira rotor akımı yükün ve kaynağın birer fonksiyonudur. Endüvi kontrollü motorlara göre alan kontrollü motorların alan sabitleri daha büyüktür. Büyük aralıklarda değişen hız ayarlarında rotor geriliminin değiştirilmesi; buna karşılık küçük aralıklarda hassas hız ayarı gereken yerlerde ise alan sargılarının yaratmış olduğu manyetik akı hız kontrolü yöntemi tercih edilir.

D.C. servo motorlar genellikle “elektronik hareketli denetleyiciler ” adı verilen servo sürücüler ile kontrol edilirler. Servo sürücüler servo motorun hareketini kontrol ederler. Kontrol edilen büyüklükler çoğu zaman noktadan noktaya konum kontrolü, hız kontrolü ve ivme programlamasıdır. PWM tekniği adı verilen darbe genişlik modülasyonu genellikle robot kontrol sistemlerinde, sayısal kontrol sistemlerinde ve diğer konum denetleyicilerinde kullanılırlar.

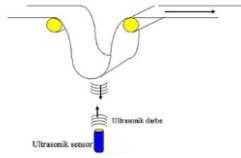


Şekil 1.9. Dc ve Servo Motor

1.3. Ultrasonic Sensor

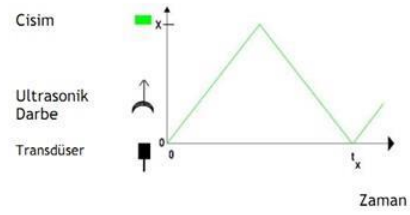
Ses dalgaları sınıflandırılmasında 20Khz-1Ghz aralığındaki ses sinyalleri ultrasonic ses olarak tanımlanmıştır. Birçok ultrasonic sensör 40Khz frekansında ultrasonic ses üretmektedir. Burada önemli olan sesin yüksekliğinde belirleyici olan etken frekanstır. Ses yüksekse frekansta yüksektir. Ultrasonic ses sinyallerini insan kulağı algılayamaz.

Sensörün çalışmasını incelersek;



Şekil 1.10. Sensörlerin çalışma prensibi.

Transdüser ultrasonik darbeyi iletir. Darbe sehimden yansır ve transdüser tarafından alınır. Darbenin gidiş geliş zamanı sensörle sehimin mesafesine göre orantılıdır.



Şekil 1.11. Darbe zaman grafiği.

Ultrasonik darbe $t=0$ zamanında transdüser tarafından iletiliyor. X pozisyonundaki hedef tarafından yansıtıldıktan sonra $t= t_x$ zamanında darbe alınıyor. t_x ; X mesafesi ile orantılıdır.

$T=0$ zamanında darbe iletilir (ultrasonic ses sinyali), cisimden yansır, transdüser tarafından algılanır ve tekrar gönderilir. Sonraki darbe ilk darbenin ultrasonik enerjisinin hepsi absorbe edildiğinde iletilmelidir. Bu yüzden sensöre bir pals gönderilir sensör okunur ve sensörün datasheetinde yazan süre kadar sensöre tekrar pals gönderilmez. Eğer bekleme yapmaksak sensör saçma değerler döndürür. Çünkü ilk yolladığımız sinyal bir yerden yansıyarak sensöre geri dönmeye devam eder.

Tüm katı ve sıvı cisimler ultrasonik dalgayı çok iyi oranda yansıtırlar. Hem katı hem de sıvı cisimlerden ultrasonik enerjinin %99u yansıtılır. Çok ufak oranlardaki enerji miktarı cisim tarafından emilir. Bundan dolayı sensörü çok çeşitli uygulamalarla da sorunsuz kullanabilmemiz mümkündür. Ayrıca robotlarda da sıkça kullanılmaktadır. Aşağıdaki resim bu tarz uygulamalara güzel bir örnektir.

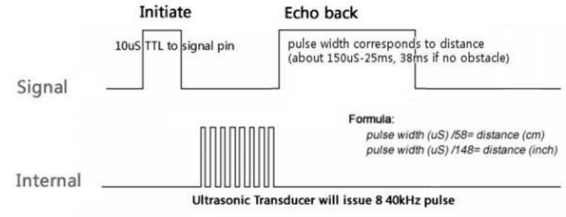
1.4. HC-SR04 Sensörü



Şekil 1.13. Ön taraf görünüm

HC-SR04 sensör üzerinde 4 adet pin mevcut. Bunlar; vcc, gnd, trig, echo pinleri. Sensörü kullanmak için trig pininden yaklaşık 10µs'lik bir puls gönderiyoruz. Sensör kendi içerisinde 40kHz frekansında bir sinyal üretilip 8 pils verici transdüserine gönderiyor. Bu ses dalgası havada, deniz seviyesinde ve 15 °C sıcaklıkta 340 m/s bir hızla yol alır. Bir cisme çarpar ve geri sensöre yansır.

Cismin sensörden uzaklığı ile doğru orantılı olarak echo pini bir süre lojik 1 seviyesinde kalır ve tekrar lojik 0 olur. Mesafeyi ölçmek için tek yapmamız gereken echo pininin ne kadar lojik1 olduğunun süresini bulmaktır. Bu yapı aşağıdaki Şekil 1.14. 'te verilmiştir.



Şekil 1.14. Lojik bir gelme süreleri

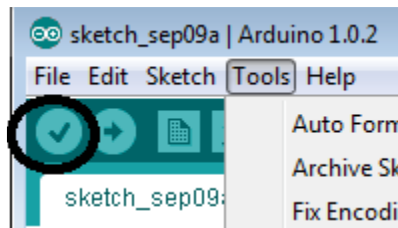
2. Denklemlerin Tanıtılması

Arduino programlama dilinde 2 temel fonksiyon bulunur.

1-) setup () : Bu fonksiyon Arduino çalışmaya başladıktan sonra ya da reset butonuna basıldıktan sonra 1 kere çalıştırılır. Bu fonksiyonda tek seferlik fonksiyonlar çalıştırılır. Örneğin pin ayarlaması, seri haberleşme başlatılması gibi.

2-) loop() : Bu fonksiyon sonsuz döngü fonksiyonlarıdır. setup() fonksiyonunun hemen ardından çalıştırılır ve arduino çalıştığı sürece devam eder.

- Birçok programlama dilinde olduğu gibi arduino programlama dili de case-sensitive(büyük küçük harf duyarlı) bir dildir.
- Arduino da yazdığımız programları Verify butonu ile derleriz.
- Verify butonu yukarıda gösterilmiştir.
- Yanında ki buton ise upload butonudur. Bu buton ile yazdığımız programı Arduino'ya yükleriz.



Şekil 1.2. Verify butonu.

Dijital Giriş Çıkış Fonksiyonları ise,

- Dijital giriş/çıkış, 1 veya 0 bilgisini okumak ya da göndermek için kullanılır.

- pinMode(pin no,giris yada çıkış modu); Bu fonksiyon pinlerin nasıl kullanılacağını ayarlar.eğer çıkış olarak kullanılacaksa OUTPUT giriş olarak kullanılacaksa INPUT yazılır.

Örnek,

```
pinMode(13,INPUT);
```

```
pinMode(13,OUTPUT);
```

- digitalWrite(pin no, HIGH or LOW); Dijital olarak çıkış ayarlanmış pinlere 1 ya da 0 verilmesini sağlayan fonksiyondur. HIGH ise 5v LOW ise 0 volt çıkış verir.

- digitalRead(pin no); Dijital olarak giriş olarak ayarlanmış pinlerdeki değerin 1 ya da 0 olduğu değerini gösterir.

NOT: Çıkış olarak ayarladığımız pinler 5v verse de maksimum verebileceği akım değeri 40mA dir. Yüksek akım isteyen elemanlarla çalışırken yükselteç kullanılmalıdır. (opamp, transistor)

- Arduino üzerinde bulunan mikrodenetleyicide 10 bitlik ADC bulunuyor. Bunun anlamı şudur ki 0-5v arası gerilimler 0 ile 1023 arasındaki sayılar olarak okunur. (1024 farklı değer)

- Eğer okuduğumuz analog değerin kaç volt olduğunu öğrenmek istersek deger*5/1023 işlemini yapmamız gerekir.

- Arduino'nun çeşidine göre üzerindeki analog giriş sayıları farklılık gösterir.

- Analog giriş den değer okumak için kullanacağımız fonksiyon analogRead(pin no); fonksiyonudur. Pin numarası olarak A0, A1, ... yazılır.

- Analog çıkış olarak PWM tekniği kullanılır. Bu teknikle dijital yöntemle analog çıkış değerleri üretebiliyoruz.

- analogWrite(pin no,duty cycle); Bu fonksiyonla analog çıkış verebiliyoruz. 0 ile 255 arasında bir değer verilebilir. 255 değeri 5 volta denk gelir.

- Burada dikkat edilmesi gereken nokta bütün dijital çıkış pinleri analogtur.

Elektronik birimler bazı projelerde birbirleriyle iletişim kurmaları gerekebilir. Dijital haberleşmede 2 yöntem vardır. Bunlar seri ve paralel olmak üzere iki tanedir.

- Seri haberleşmede veriler tek bir hat üzerinden sıra ile gönderilir.

- Seri haberleşmede daha az veri hattı gerekmektedir. Bu yüzden sıkça kullanılır. Günümüzde en çok kullanılan USB buna en iyi örnektir. Derlediğimiz programları arduino kartına yükleme işlemi de aslında seri haberleşme ile olur (USB ile).

- Seri haberleşme 2 ayrı hat üzerinden (RX ve TX) gerçekleşir.

- Arduino üzerinde bulunan seri haberleşme ünitesine UART (Universal asynchronous receiver/transmitter: Evrensel asenkron alıcı/verici) adı verilir. Arduino modeline göre 1 ya da daha fazla haberleşme ünitesi bulunabilir

- TX ve RX in bağlı olduğu pinler seri haberleşme esnasında dijital olarak giriş ya da çıkış olarak kullanılamaz.

- available()...Kaç tane okunmayı bekleyen veri (bayt) var?

- begin().....Seri İletişimi başlatma

- end().....Seri iletişimi sonlandırma

- print().....Seri iletişim üzerinden veri gönderme (text)

- println().....Veri gönderme (satır sonu karakteri eklenir)

- read().....Gelen veriden okuma

- readBytes()...Gelen verileri topluca okuma

- write().....Veri gönderme (binary)

2.1. Dijital Giriş Çıkış Fonksiyonları

- Dijital giriş/çıkış 1 veya 0 bilgisini okumak ya da göndermek için kullanılır.

- pinMode(pin no,giris yada çıkış modu); Bu fonksiyon pinlerin nasıl kullanılacağını ayarlar. Eğer çıkış olarak kullanılacaksa OUTPUT giriş olarak kullanılacaksa INPUT yazılır. Örnek,

```
pinMode(13,INPUT);
```

```
pinMode(13,OUTPUT);
```

- digitalWrite(pin no, HIGH or LOW); Dijital olarak çıkış ayarlanmış pinlere 1 ya da 0 verilmesini sağlayan fonksiyondur. HIGH ise 5v LOW ise 0 volt çıkış verir.

- digitalRead(pin no); Dijital olarak giriş olarak ayarlanmış pinlerdeki değerin 1 ya da 0 olduğu değerini gösterir.

- NOT: Çıkış olarak ayarladığımız pinler 5v verse de maksimum verebileceği akım değeri 40mA dir.

Yüksek akım isteyen isteyen elemanlarla çalışırken yükselteç kullanılmalıdır. (opamp, transistor, vb.)

2.2. Gecikme Fonksiyonları

- delay(); Bu fonksiyonun içine yazdığımız kadar fonksiyonumuz o noktada o kadar milisaniye cinsinden durur.
- delayMicroseconds(); Bu fonksiyon ise microsanaye cinsinden fonksiyonu durdurur.

2.3. Üzerine Atılan Cisimden Kaçan Robot Çalışma prensibi

İlk olarak robotun montajını gerçekleştirdik(Robot gövdesi DC motorlar, Servo motor, HC-SR04 Ultrasonik Sensör, Ardunio, Sarhoş teker, bağlantı kabloları ve Piller). Robotun üst kısmı ve robotun tam orta bölgesine Micro Servo motoru sabitledik. Projemiz; üzerine atılan herhangi bir cisimden kaçan robot olduğu için, Micro Servo motorun üzerine birbirine sabitlenmiş iki adet Ultrasonik Sensör yerleştirdik. Micro Servo motor sayesinde bu iki sensörün 180 derece dönmesini sağlıyoruz ve Sensörlerin bir tanesi 15 derecelik açı taradığı için iki Sensör ile bu taranan açıyı arttırıp 180 derece Mikro Servo motor ile döndürerek üzerine atılan cismi tam olarak kısa bir zamanda algılamasını sağlıyoruz. Ultrasonik sensörler herhangi bir cisim algıladığı zaman ve cisim 30 cm den az mesafede ise servo motorlara hareket veriyor ve robot hızlı bir şekilde ileri yönde atılan bu cisimden kaçma işlemini gerçekleştiriyor.

2.4. Yazılım

Robotun tasarlanan algoritmasını Ardunio IDE kullanarak robota aktarma kısmında kullanılan fonksiyonlar arduinonun sunduğu eğitim setlerinden öğrenilerek örnek fonksiyonlar üzerinden yazılmıştır.

\\Tanımlama işlemi gerçekleşiyor

```
const int trig=10;
const int trig1=11;
const int echo=8;
const int echo1=9;
const int sol_i=2;
const int sol_g=3;
const int sag_i=4;
const int sag_g=5;
\\Süre ve mesafe tanımlanıyor
int sure=0,sure1=0;
int mesafe=0,mesafe1=0;
#include <Servo.h>
Servo myservo;
```

```
int pos = 0;
void setup()
//Pinler atanıyor
{
pinMode(trig1,OUTPUT);
pinMode(trig,OUTPUT);
pinMode(echo,INPUT);
pinMode(echo1,INPUT);
pinMode(sol_i,OUTPUT);
pinMode(sol_g,OUTPUT);
pinMode(sag_i,OUTPUT);
pinMode(sag_g,OUTPUT);
// Seri iletişim bağlantısı
myservo.attach(6);
Serial.begin(9600);
}
//Döngü oluşturuluyor mesafe hesaplanıyor
void loop() {
digitalWrite(trig,HIGH);
delayMicroseconds(1000);

digitalWrite(trig,LOW);
sure=pulseIn(echo,HIGH);
mesafe=(sure/2)/28.5;
digitalWrite(trig1,HIGH);
delayMicroseconds(1000);
digitalWrite(trig1,LOW);
sure1=pulseIn(echo1,HIGH);
mesafe1=(sure1/2)/28.5;
//Servo motoru 90 derecelik açılarla döndürme
for(pos = 0; pos <= 91; pos += 7)
{
myservo.write(pos);
delay(15);
//Mesafeler 30 cm den az ise ileri hareket
if (mesafe<30||mesafe1<30)
{
digitalWrite(sag_i,HIGH);
digitalWrite(sol_i,HIGH);
digitalWrite(sag_g,LOW);
```



```
digitalWrite(sol_g,LOW);
Serial.println("araba ileri gidiyor");
Serial.println(mesafe);
Serial.println(mesafe1);
}
//Diger durumda motorlar dursun
else
{
digitalWrite(sag_i,LOW);
digitalWrite(sol_i,LOW);
digitalWrite(sag_g,LOW);
digitalWrite(sol_g,LOW);
}
}
for(pos = 91; pos>=0; pos-=7)
{
myservo.write(pos);
delay(15);
```

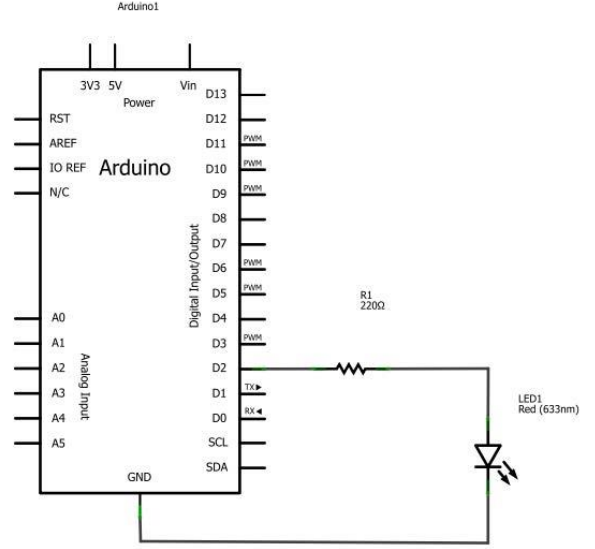
```
// 30 cm den az ise mesafe cismi algılıyor motora
hareket veriyor
if (mesafe<30||mesafe1<30)
{
digitalWrite(sag_i,HIGH);
digitalWrite(sol_i,HIGH);
digitalWrite(sag_g,LOW);
digitalWrite(sol_g,LOW);
Serial.println("arabaor");
}
else
{
//Diger durumlarda motorların hepsi duruyor

digitalWrite(sag_i,LOW);
digitalWrite(sol_i,LOW);
digitalWrite(sag_g,LOW);
digitalWrite(sol_g,LOW);
}
```

```
}
}
```

3. Uygulamalı Basit Bir Örnek

Led Yakma

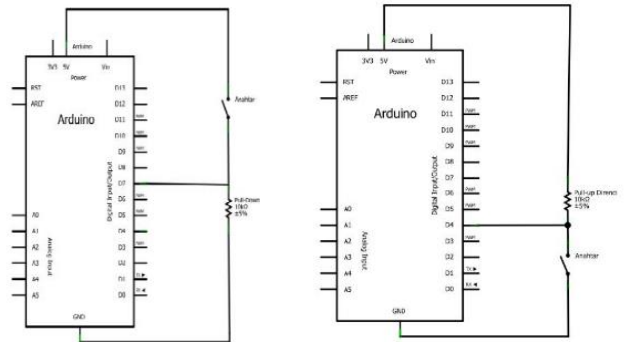


Şekil 1.5. Devre tasarımı.

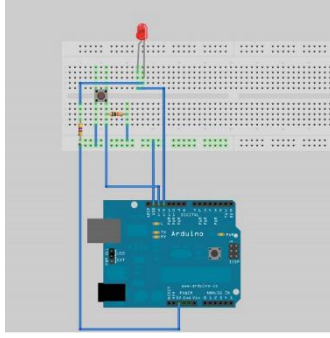
Program:

```
void setup() {
pinMode(2, OUTPUT);
}
void loop() {
digitalWrite(2, HIGH);
delay(1000);
digitalWrite(2, LOW);
delay(1000);
}
```

Buton Girişi Okuma



Şekil 1.6. Devre Tasarımları.



Şekil 1.7. Program ile Çizim.

```
Int ledPin=12;
Int butonPin=13;
void setup()
{
  pinMode(ledPin,OUTPUT);
  pinMode(butonPin,LOW);
}
void loop() {
  // Buton durumunu oku
  buttonDurumu =
  digitalRead(butonPin);

  /* Butona basıldığında butonun
  durumu HIGH olacaktır.
  Bu durumda LED çıkışını HIGH
  yapıyoruz. Ters durumda
  ise LOW yapıyoruz */
  if (buttonDurumu == HIGH) {

    digitalWrite(ledPin, HIGH);

  }
  else {
    digitalWrite(ledPin, LOW);
  }
}
```

4. Kaynaklar

1. İnternet:”Arduino Özellikleri”
http://www.robotiksistem.com/arduino_ne_dir_arduino_ozellikleri.html
2. İnternet:”HC-SR04 ultrasonik sensör”,
<http://robotiktak.com/arduino-4-hc-sr04-ultrasonik-mesafe-sensoru-ile-uygulama/>,(2016).
3. İnternet:”Servo motor”,
<http://www.robotistan.com/tower-pro-sg90-rc-mini-servo-motor> ,(2016).
4. İnternet:”Dc motor”,
http://www.robotiksistem.com/dc_motor_ozellikleri.html
5. İnternet:”Engelden kaçan özellikler”,
<http://arduinom.org/engelden-kacan-robot/>
6. İnternet: “9V Pil Göresli”,
http://www.direnc.net/Pil--Aku--SolarPanel,LA_360-2.html#labels=360-2,(2016).

Özgeçmiş



İdris IŞIK, 1991'de Tosya'da doğdu; İlk ve Orta öğrenimini aynı şehirde tamamladı; Ankara'da Ahmet YESEVİ Lisesi'nden mezun olduktan sonra 2011 yılında Karabük Üniversitesi, Mühendislik Fakültesi, Mekatronik

Mühendisliği Bölümü'ne girdi. Halen bu bölümde eğitimini sürdürmektedir.



Ibrahim Cayiroglu is an insructor in Mechatronic Engineering at Karabuk University, Turkey. He received his B.Sc. in Mechanical Engineering from Istanbul Technical University in 1991. He received his M.Sc. and Ph.D. in

Computer Aided Design and Manufacturing from Kirikkale University, in 1996 and 2002, respectively. His research interests include CAD-CAM, Software and Mechatronic Systems.