

İÇİNDEKİLER

ÖZET	ii
TEŞEKKÜR	iii
ŞEKİL LİSTESİ	v
1 GİRİŞ	1
2 PROJEDE KULLANILAN ARAÇLAR/PROGRAMLAR	2
2.1 Klon Arduino Uno R3 DIP	2
2.2 RC522 RFID Kart	4
2.3 HC-SR04 Ultrasonik Mesafe Sensörü	5
2.4 SG90 9G Servo Motor Mini	5
2.5 Arduino IDE	6
2.6 Kütüphaneler	7
3 PROJENİN TANIMLANMASI	9
3.1 Araç Girişi	9
3.2 Araç Çıkışı	10
3.3 Proje Şeması ve görüntüler	12
4 EKLER	14
4.1 Arduino IDE Kodları	14
KAYNAKLAR	16
ÖZGEÇMİŞ	17

ŞEKİL LİSTESİ

1	Araç otopark kapısının önünde beklemekte	9
2	RFID kart ID'si veritabanında var, kapı açıldı	10
3	Araç otopartan çıkmak için kapıda beklemekte	10
4	Ultrasonik mesafe sensörü arcın mesafesini algıladı ve kapı açıldı	11
5	Proje Şeması	12
6	Projenin Sağ Yandan İç tarafının Görünümü	12
7	Projenin Kod Bloğunun Bir Kısmı	13

1 GİRİŞ

Günümüzde insanlarınımızın öncelikli ihtiyaçlarından biri olan ulaşım için kullanılan toplu taşıma araçlarının yetmediği durumlarda veya ulaşım yaparken kendi konfumuzu sağlamak istediğimizden dolayı, birçoğumuzun şahsi araçları vardır.

Araç almanın artık temel ihtiyaca dönüştüğü bu dönemde bir yere gittiğimizde araçlarımızın güvenli bir yerde durmasını isteriz. Bunun için en uygun yerler olan otoparkların giriş çıkış kontrollerinin yapılmasını ve otoparkların daha güvenli hale gelmesini hepimiz isteriz.

Bu sebeplerden dolayı bende güvenli ve hızlı olması için kartlı giriş ve otomatik çıkış kapıları olan bir otopark turnikesi geliştirdim.

Projemde kullanmış olduğum araç/programlar:

Arduino IDE(Arduino mikroişlemci kartın kodlanması)

Klon Arduino Uno r3 DIP

RC522 RFID kart

HC-SR04 Ultrasonik sensör

SG90 9G Mini Servo Motor

2 PROJEDE KULLANILAN ARAÇLAR/PROGRAMLAR

2.1 Klon Arduino Uno R3 DIP

Arduino Uno R3 Arduino'nun Uno versiyonu için yaptığı 3. revizyonun sonucudur. Bu Arduino Uno R3 Klon ürününde ATmega328p DIP kılıf devreye monte edilmiştir. Ekstra bu kılıf sayesinde mikrodenetleyicimizde herhangi bir sorun olması durumda yenisini alıp devreye monte etmemizi ayrıca istediğimizde program yazıp kullandığımız ATmega328p'yi kendi yaptığımız DIY projelerimizde kullanma imkanı sunar.

Arduino Uno R3 CH340 SMD ile Arduino Uno Dip arasındaki fark mikrodenetleyinin SMD veya DIP olarak ayrılmاسının yanı sıra DIP türünün değiştirilebilir olmasıdır.

Klon Arduino Uno R3 DIP Özellikleri:

Mikrodenetçi: ATmega328P U

Çalışma voltajı: 5V

Giriş voltajı (önerilen): 7-12V

Giriş voltajı (limit değerler): 6-20V

Dijital I / O Pinleri 14 (bunlardan 6'sı PWM çıkışını sağlamaktadır)

PWM Dijital I/O Pinleri: 6

Analog Giriş Pinleri: 6

I/O Pin Başına DC akım: 20 mA

3.3V Pin DC akımı: 50 mA

Flash Bellek 32 KB (ATmega328P) 0.5 KB bootloader tarafından kullanılır

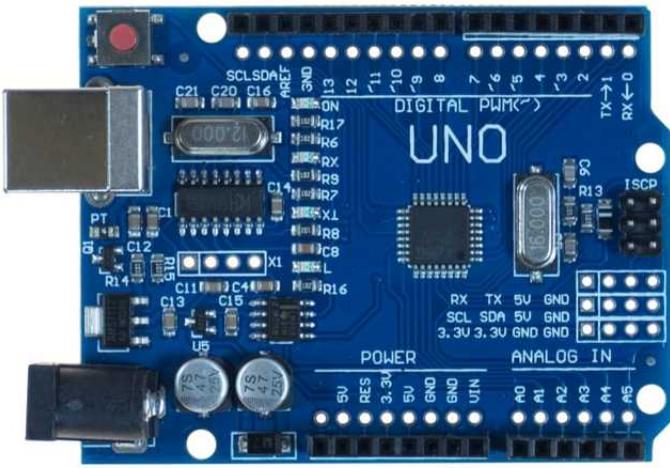
SRAM: 2 KB (ATmega328P)

EEPROM: 1 KB (ATmega328P)

Saat Hızı: 16 MHz

Uzunluk: 68.6 mm

Genişlik: 53.4 mm



Arduino Nedir?

Arduino kullanımı kolay yazılım ve donanım sunan açık kaynak kodlu elektronik bir platformdur. Herkesin kolayca prototip geliştirmesi ve interaktif projeler yapması için tasarlanmıştır. Temelde ATMEL mikrodenetleyicilerini kullanan Arduino platformu yıllar içinde gelişen kütüphaneleri ve bu kütüphaneleri yazan topluluğuyla elektronik ve programlama alanında kendini geliştirmek isteyen herkes için kolay ve ulaşılabilir çözümler sunuyor.

Arduino açık kaynak kodlu bir platformdur yani Arduino'nun tüm ürünleri, ilgili yazımları, tasarımları açıkça herkes ile paylaşılmıştır. Bu da isteyen herkesin Arduino üretmesine imkan sağlar ki dünya üzerinde bir çok elektronik şirketi Arduino klonlarını üretmektedir. Bu durum kullanıcılara bir çok çeşit ve fiyat aralığı sunması açısından faydalı sağlar.

Arduino'nun donanımları ne kadar popüler ve kullanışlı ise yazılımı da bir o kadar kullanışlı ve popülerdir. Arduino yazılım geliştirme platformu (IDE) sadece Arduino için değil elektronik piyasasında bulunan farklı geliştirme/prototip kartları için dahi kullanılabilir. Sayısız kütüphaneleri ile istediğiniz her türlü genişletme kartı için aradığınız kütüphaneleri bulabilirsiniz. Kütüphaneler Arduino yazılım geliştirme platformu üzerinden sorunsuzca yüklenebilir.

2.2 RC522 RFID Kart

RC522 RFID kartı, NFC frekansı olan 13,56 MHz frekansında çalışan tagler üzerinde okuma ve yazma işlemini yapabilen, düşük güç tüketimli, ufak boyutlu bir karttır. Arduino başta olmak üzere bir çok mikrodenetleyici platformu ile beraber rahatlıkla kullanılabilir. 424 kbit/s haberleşme hızına sahiptir. RFID üzerinde farklı şifreleme türlerini desteklemektedir. Desteklediği kart türleri mifare1 S50, mifare1 S70 mifare ultralight, mifare pro ve mifare desfire'dir.

Özellikleri:

Çalışma Frekansı: 13,56 MHz

Çalışma Akımı: 13-26mA

Uyku Akımı: 80uA

Haberleşme Protokolü: SPI

Desteklenen Kartlar: mifare1 S50, mifare1 S70 mifare ultralight, mifare pro ve mifare desfire

Kart Boyutları: 40x60mm



2.3 HC-SR04 Ultrasonik Mesafe Sensörü

2cm'den 400cm'ye kadar 3mm hassasiyetle ölçüm yapabilen bu ultrasonik sensör çeşididir. Uzaklık okuma, radar ve robot uygulamalarında kullanılabilir.

HC-SR04 Özellikleri:

Çalışma Voltajı: DC 5V

Çektiği Akım: 15 mA

Çalışma Frekansı: 40 Hz

Maksimum Görme Menzili: 4m

Minimum Görme Menzili: 2cm

Tetik Bacağı Giriş Sinyali: 10 us TTL Darbesi

Echo Çıkış Sinyali: Giriş TTL sinyali ve Mesafe Oranı

Boyu: 45mm x 20mm x 15mm



2.4 SG90 9G Servo Motor Mini

Tower Pro SG90 9G Servo Motor Mini, küçük boyutlu projeler geliştirebilmeniz için son derece uygun bir servodur. RC Kumandalar ile uyumlu olarak çalışmaktadır.



SG90 9G Mini Servo Motor Özellikleri:

Voltaj Aralığı: 4,8V ve 7V aralığında çalışmaktadır.

Yüklü Tork: 4,8V 1,3 Kg.cm ve 6V 1,6 Kg.cm

Hız: Yüksüz 4,8V 0,12/60 Derece

2.5 Arduino IDE

Arduino için Entegre Geliştirme Ortamı(IDE), C ve C ++ dilleri ile yazılmış bir platformlar arası uygulamadır (Linux, macOS, Windows için,). Arduino uyumlu kartlara program yazmak ve yüklemek için kullanılır, aynı zamanda 3. taraf çekirdekler ve satıcıların geliştirme kartları içinde kullanılabilir.

Arduino IDE'nin kaynak kodu GNU Genel Kamu Lisansı sürüm 2 ile yayınlanmıştır. Arduino IDE'si, özel kod yapılandırması kuralları kullanarak C ve C ++ dillerini destekler. Arduino IDE, Wiring projesinde bulunan birçok yaygın giriş ve çıkış prosedürünü bir yazılım kütüphanesi ile sağlar. Kullanıcı tarafından yazılan kod, iki temel fonksiyon gerektirmektedir, başlama noktası olarak adlandırılabileniz bölüm ve ana döngünün gerçekleşeceği kısmı. Bunlar GNU araç zinciri sayesinde bağlanır ve derlenir. Arduino

IDE temelde, çalıştırılabilir kodu hexadecimal format ile metin dosyasına işler. Ardından kullandığımız Arduino IDE'si bu metin dosyasını bağlı olan arduino kartına firmware'ına yükleyici program ile aktarmayı gerçekleştirir.



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1.8.5". The main window displays the "Blink" sketch code. The code is as follows:

```
This example code is in the public domain.  
http://www.arduino.cc/en/Tutorial/Blink  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```

The status bar at the bottom right shows "Arduino/Genuino Uno on COM1".

2.6 Kütüphaneler

Arduino Spi Kütüphanesi

"include <SPI.h>" komutu ile kullanılır. Spi haberleşme protokolü, spi.h kütüphanesi ile ana cihaz olarak Arduino ve SPI cihazlarıyla iletişim kurmanızı sağlar.

Arduino MFRC522 Kütüphanesi

"include <MFRC522.h>" komutu ile kullanılır. MFRC522 ve diğer RFID RC522 tabanlı modüller için Arduino kütüphanesidir. Seri Çevre Arabirim (SPI) arabirimini aracılığıyla bağlanan RC522 tabanlı bir okuyucunu kullanarak Arduino'nuzda farklı Radyo Frekansı Tanımlama (RFID) kart türlerini okuyumaya ve yazmaya yarar.

Arduino Servo Kütüphanesi

Arduino ile Servo Motor kullanımı için Arduino'nun Servo.h kütüphanesinden yararlanılabilmektedir. Servo kütüphanesini kullanmak bizler için kolaylıklar sağlamaktadır ancak

bu bir gereklilik değildir. Servo Motorların kontrolü için ihtiyaç duyulan sinyali Arduino üzerinden bulunan PWM destekli pinler kullanılarakta üretilebilir ve Servo Motor kontrolünü sağlanabilir.

3 PROJENİN TANIMLANMASI

3.1 Araç Girişİ

Şekil 1'de gösterildiği gibi toparka araçlarını parketmek isteyen araç sahipleri araçlarını kapının dışına getirerek RFID kartlarını okuturlar.



Şekil 1: Araç otopark kapısının önünde beklemekte

Burada şekil 2'de gösterildiği gibi okutulan kartın ID numarası veritabanına kayıtlı ise (burada veritabanı kullanmadım, kod içinde tanımladım) kapı açılır ve kapı açıldıktan 5 saniye sonra otomatik olarak kapanır.

```
Mesafe:430cm  
Mesafe:428cm  
Mesafe:160cm  
Mesafe:430cm  
Kapi acildi  
ID Numarasi: 58 27 162 127
```



Şekil 2: RFID kart ID'si veritabanında var, kapı açıldı

3.2 Araç Çıkışı

Şekil 3'de gösterildiği gibi otoparktan çıkmak isteyen araç sahipleri, araçlarını ultrasonik sensörün aracı alglayacağı mesafeye getiri.(9 cm ayarladım)



Şekil 3: Araç otopartan çıkmak için kapıda beklemekte

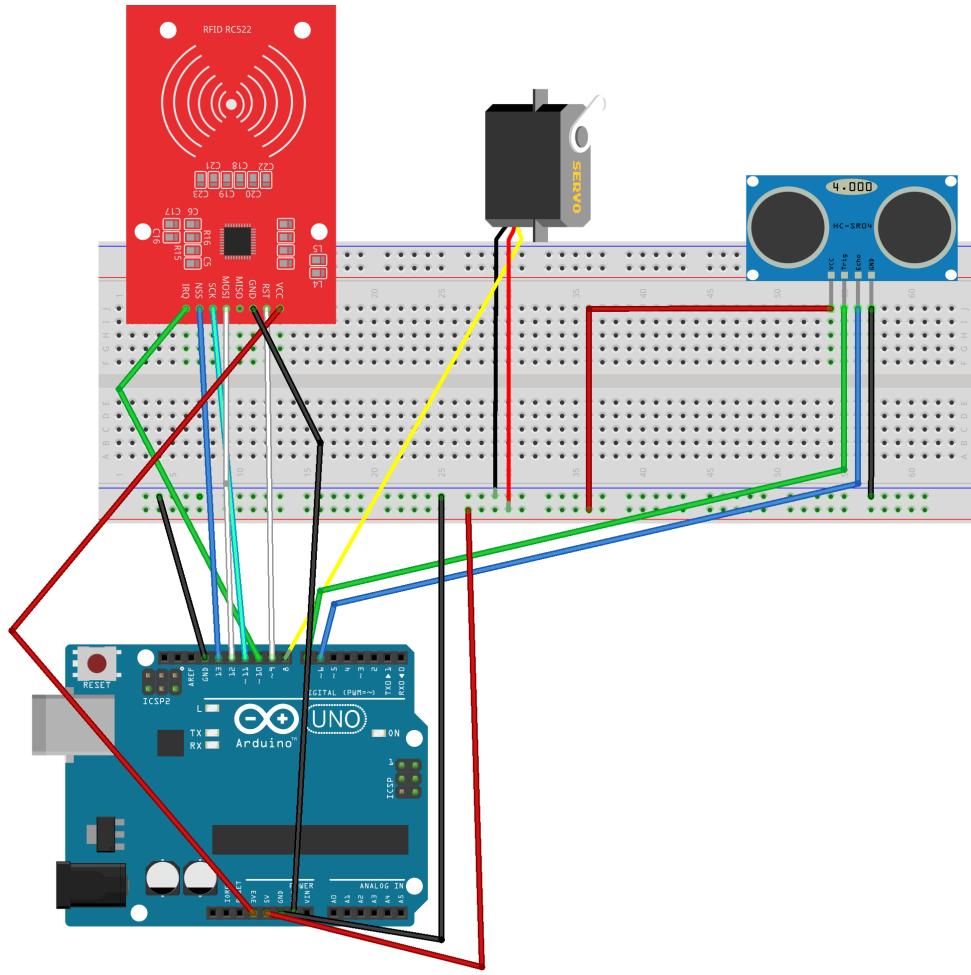
Ultrasonik mesafe sensörü sistem çalıştığı andan itibaren her 1 saniyede 1 kez mesafe algılaması yapmaktadır. Şekil 4 'de gösterildiği gibi sensörün önüne yaklaşan cismin mesafesi 9 cm'den küçükse, kapı açılmakta ve 5 saniye bekleyip otomatik olarak kapanmaktadır.

Mesafe: 431cm
Mesafe: 430cm
Mesafe: 428cm
Mesafe: 2cm
Kapi acildi

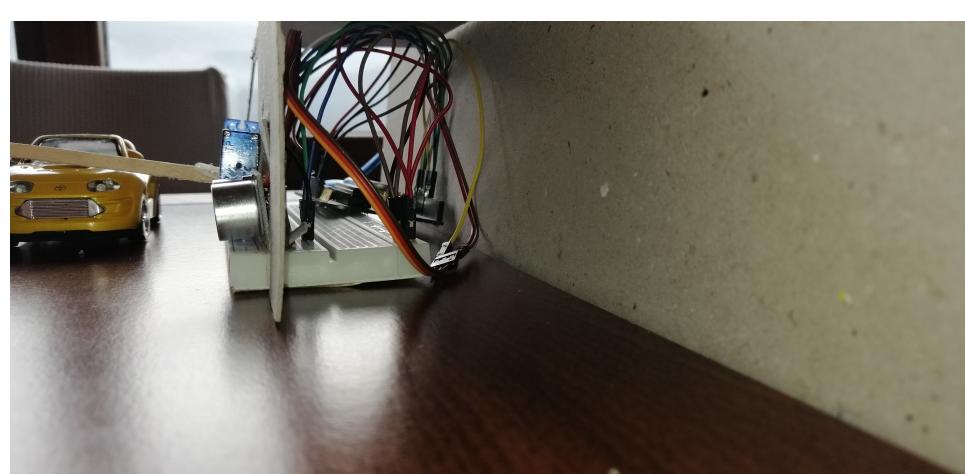


Şekil 4: Ultrasonik mesafe sensörü arcın mesafesini algıladı ve kapı açıldı

3.3 Proje Şeması ve görüntüler



Şekil 5: Proje Şeması



Şekil 6: Projenin Sağ Yandan İç taraflarının Görünümü

```

#include <SPI.h>
#include <MFRC522.h>
#include <Servo.h>

int RST_PIN = 9; //SPI kütüphanemizi tanımlıyoruz.
int SS_PIN = 10; //MFRC522 kütüphanemizi tanımlıyoruz.
int servoPin = 8; //Servo kütüphanemizi tanımlıyoruz.

int aure;
int mesafe;
const int trig = 7;
const int echo = 6;

Servo motor; //Servo motor için değişken oluşturuyoruz.
MFRC522 rfid(SS_PIN, RST_PIN); //MFRC522 modülün ayarlarını yapıyoruz.
byte ID[4] = {38, 27, 162, 127}; //Yetkili kart ID'sini tanımlıyoruz.

void setup() {
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);

    motor.attach(servoPin); //Servo motor pinini motor değişkeni ile ilişkilendiriyoruz.
    Serial.begin(9600); //Seri haberleşmeyi başlatıyoruz.
    SPI.begin(); //SPI iletişimini başlatıyoruz.
    rfid.PCD_Tx(); //MFRC522 modülünü başlatıyoruz.
}

void loop() {
    //ultrasonik sensör işlemlerini başlangıcı

    digitalWrite(trig, LOW);
    delayMicroseconds(2);
    digitalWrite(trig, HIGH);
    delayMicroseconds(2);
    digitalWrite(trig, LOW);

    aure = pulseIn(echo, HIGH);
    mesafe = aure/58.2;

    Serial.print("Mesafe:");
    Serial.print(mesafe);
    Serial.print("cm");
    Serial.print("\n");
    delay(1000);

    if(mesafe<9){
        Serial.println("Kapı açıldı");
        motor.write(70);
        delay(5000);
        motor.write(0);
        delay(10);
    }
    else
        exit();

    // ultrasonik sensör işlemlerinin sonu

    if ( ! rfid.PICC_IsNewCardPresent()) //Yeni kartın okunmasını bekliyoruz.
        return;

    if ( ! rfid.PICC_ReadCardSerial()) //Kart okunmadığı zaman bekliyoruz.
        return;
}

```

Şekil 7: Projenin Kod Bloğunun Bir Kısımı

Projenin kodlarının tam hali "EKLER" bölümünde verilmiştir.

4 EKLER

4.1 Arduino IDE Kodları

```
include <SPI.h> //SPI kütüphanemizi tanımlıyoruz. include <MFRC522.h> //MFRC522 kütüphanemizi tanımlıyoruz. include <Servo.h> //Servo kütüphanemizi tanımlıyoruz.

int RSTPIN = 9; //RC522 modül reset pinini tanımlıyoruz. int SSPIN = 10; //RC522 modül chips 8; //Servomotor pinini tanımlıyoruz.

int sure; int mesafe; const int trig = 7 ; const int echo = 6 ;

Servo motor; //Servo motor için değişken oluşturuyoruz. MFRC522 rfid(SSPIN, RSTPIN); //RC522 58, 27, 162, 127; //Yetkilik kart ID'sini tanımlıyoruz.

void setup() pinMode(trig,OUTPUT); pinMode(echo,INPUT);

motor.attach(servoPin); //Servo motor pinini motor değişkeni ile ilişkilendiriyoruz. Serial.begin(9600); //Seri haberleşmeyi başlatıyoruz. SPI.begin(); //SPI iletişimini başlatıyoruz. rfid.PCDInit(); //RC522 modül balatlığını.

void loop() //ultrasonik sensör işlemleri başlangıcı

digitalWrite(trig,LOW); delayMicroseconds(2); digitalWrite(trig,HIGH); delayMicroseconds(2); digitalWrite(trig, LOW);

sure = pulseIn(echo,HIGH); mesafe = sure/58.2;

Serial.print("Mesafe:"); Serial.print(mesafe); Serial.print("cm"); Serial.print(""); delay(1000);

if(mesafe<9) Serial.println("Kapi acildi"); motor.write(70); delay(5000); motor.write(0); delay(10);

else exit;

// ultrasonik sensör işlemlerin sonu
```

```

if ( ! rfid.PICC_IsNewCardPresent())//Yeni kartın yokunması bekliyoruz.return;

if ( ! rfid.PICC_ReadCardSerial())//Kart okunmadığı zaman bekliyoruz.return;

if (rfid.uid.uidByte[0] == ID[0] //Okunan kart ID'si ile ID değişkenini karşılaştırıyoruz.
rfid.uid.uidByte[1] == ID[1] rfid.uid.uidByte[2] == ID[2] rfid.uid.uidByte[3] == ID[3]
) Serial.println("Kapı açıldı"); ekranaYazdir(); motor.write(70); //Servo motoru 180 dereceye getiriyoruz. delay(5000); motor.write(0); //Servo motoru 0 dereceye getiriyoruz.
delay(1000); else //Yetkisiz girişte içerisindeki komutlar çalıştırılır. Serial.println("Yetkisiz
Kart"); ekranaYazdir(); rfid.PICC_HaltA();

void ekranaYazdir() Serial.print("ID Numarası: "); for(int sayac = 0; sayac < 4; sayac++)
Serial.print(rfid.uid.uidByte[sayac]); Serial.print(" "); Serial.println("");

```

KAYNAKLAR

- [1] [https://github.com/Robotistan/YouTube-DIY-Projeler/
blob/master/Arduino%20Dersleri/Ders-21%20RFID%20Kart%
20Okuyucu/RFID_Kapi.ino](https://github.com/Robotistan/YouTube-DIY-Projeler/blob/master/Arduino%20Dersleri/Ders-21%20RFID%20Kart%20Okuyucu/RFID_Kapi.ino)
- [2] <https://www.youtube.com/watch?v=SCo0K9zsnV4>
- [3] https://www.youtube.com/watch?v=FxY_-MaJV-k
- [4] <https://www.youtube.com/watch?v=ewTyfZYIsZU>
- [5] <https://github.com/arduino/Arduino>