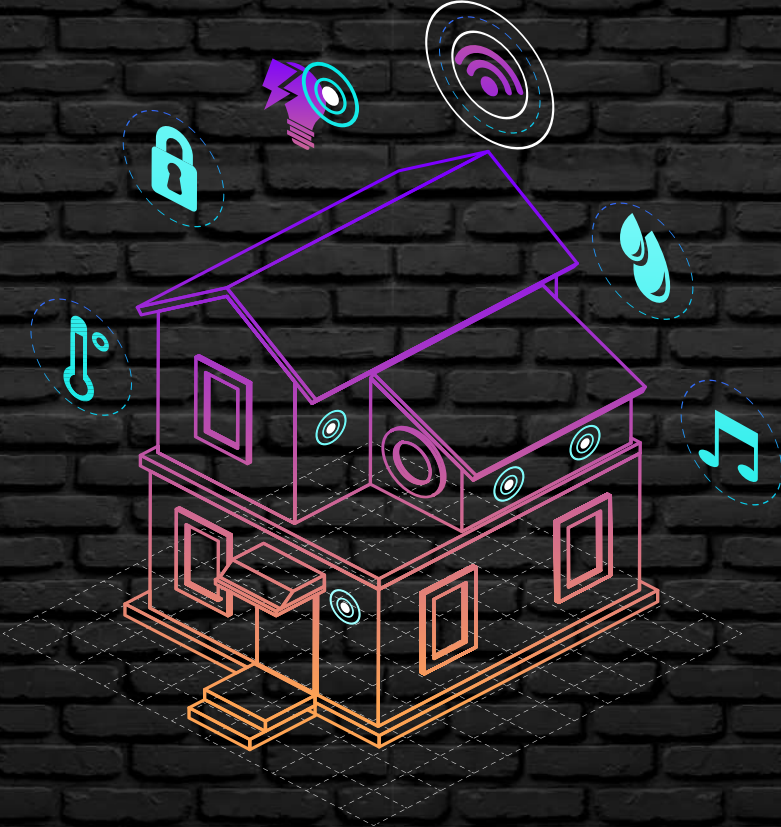


robotistan



Akıllı Ev Proje Kitabı





Arduino Uno Klon

Atmega328 temelli bir mikrodeneleyici kartıdır. Üzerinde 14 adet dijital giriş/ıkış pini, 6 analog giriş, 16Mhz kristal, usb soketi, ICSP konektörü ve reset tuşu bulundurmaktadır. Kart üzerinde mikrodeneleyicinin alışması için gerekli olan her şey bulunmaktadır. Kolayca usb kablosu üzerinden bilgisayara Bađlanabilir, adaptör veya pil ile alıştırılabilir.



MQ-2 Gaz Sensörü

Ortama bulunan ve konsantrasyonu 300 ile 10,000 ppm arasında deđiřen yanıcı ve patlayıcı gaz ve/veya dumanı algılayan bu yarıiletken gaz sensörü, -20 ile 50°C arasında alışabilir ve 5V'ta sadece 150mA akım eker. Analog ıkışı sayesinde algılanan gaz konsantrasyonu kolayca okunabilir.



Role Kontrol Kartı

5V ile kontakların kontrol edilebildiđi, Arduino veya diđer bařka mikrodeneleyeciler ile kullanılabilen bir rölle kartıdır. Mikrodeneleyeciden tetik sinyali sırasında 20mA'lık bir akım ekmektedir. eřitli hobi, endüstriyel ve robotik projelerde sıklıkla kullanılır. 30VDC veya 220VAC gerilimde 10A'e kadar akımı anahtarlayabilmektedir. Her bir role için kontrol ledleri bulunmaktadır.



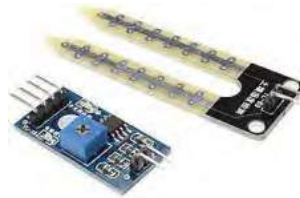
Dht 11 Sıcaklık Ve Nem Sensörü

DHT11 sıcaklık ve nem algılayıcı kalibre edilmiř dijital sinyal ıkışı veren geliřmiř bir algılayıcı birimidir. Yüksek güvenilirliktedir ve uzun dönem alışmalarda dengelidir. 0 ile 50°C arasında 2°C hata payı ile sıcaklık ölçer.



Bluetooth Modülü

Bluetooth SSP(Serial Port Standart) kullanımı ve kablosuz seri haberleşme uygulamaları için tasarlanmıştır. Hızlı prototiplemeye imkan sağlaması, breadboard, arduino ve eřitli devrelerde rahata kullanılabilmesi için gerekli pinler devre kartı sayesinde dıřarıya alınmıştır.



Toprak Nem Sensörü

Toprađın ierisindeki nem miktarını veya ufak ölçekte bir sıvının seviyesini ölçmek için kullanılabileceđiniz bir sensördür. Nem ölçer prob lar ölçüm yapılacak ortama batırılarak kullanılır. Toprađın veya iine batırılan sıvının meydana getirdiđi direnten dolayı, prob uçları arasında bir gerilim farkı oluşur.



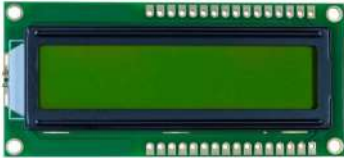
Diyot LED

Işık yayan diyot (LED), adından da anlaşılacağı gibi enerji verildiği zaman görülebilir bir ışık yayan diyottur. Genel olarak kırmızı, sarı ve yeşil olmak üzere üç değişik renkte yapılırlar. Çalışma akımları 5 mA ile 50 mA arasındadır.



Pir Sensörü

PIR sensörleri, bir ortamda oluşan canlı hareketini algılamak için kullanılan sensörlerdir. Bu minik boyutlu sensör, çeşitli elektronik, robotik ve hobi uygulamalarında rahatça kullanılabileceğiniz, Arduino başta olmak üzere bir çok mikrodenetleyeci platformu ile beraber kullanılabilir modüldür.



2x16 LCD Ekran

Arduino ve diğer mikrokontrolcü projelerinizde kullanılabileceğiniz kaliteli bir 16x2 (16 sütun, 2 satır) LCD ekrandır. Yeşil zemin üzerine siyah yazı rengine sahiptir. Arka LED aydınlatması mevcuttur. 5V gerilim ile çalışır.



LDR

LDR (Light Dependent Resistor), Türkçede "Işığa Bağımlı Direnç" anlamına gelmektedir. Bir diğer adı da foto dirençtir. LDR her ne kadar bir direnç çeşidi olsa da aynı zamanda pasif bir sensördür. LDR'ler bulundukları devrelerde değişen direnç değerleri ile bir çıkış sağlarlar fakat bu işlemi dış ortamdan aldıkları fiziksel bir değişim ile gerçekleştirdiklerinden dolayı bir Sensör görevi görmüş olurlar.



Yağmur Sensörü

Birbirine paralel olarak çekilmiş iletken hatların su ile teması sonucu sensör çıkış pininde analog bir değer okunabilmektedir. Arduino başta olmak üzere bir çok Mikrodenetleyeci platformu ile beraber kullanılabilir.



Buzzer

Buzzer; mekanik, elektromekanik ya da piezoelektrik Prensiplerine bağlı olarak çalışan işitsel ikaz cihazı çeşididir. Kullanım alanları oldukça fazla olan buzzerlar, genel itibarıyla piezoelektrik prensibiyle çalışmaktadırlar. Buzzerlar, kullanım alanlarına da bağlı olarak alarm, zamanlayıcı, onaylama cevap ikazı gibi işlevlerde kullanılabilmektedirler.



RC522 RFID NFC Modülü, Kart ve Anahtarlık Kiti

Rc522 RFID kartı, NFC frekansı olan 13,56 MHz frekansında çalışan tagler üzerinde okuma ve yazma işlemi yapabilen, düşük güç tüketimli, ufak boyutlu bir karttır.Arduino başta olmak üzere bir çok mikrodenetleyeci platformu ile beraber rahatlıkla kullanılabilir.



NeoPixel 12'li Halka

NeoPixel, sadece tek mikrokontrolcü pini kullanarak sürebileceđiniz, peş peşe bağlayarak çoklu kullanım imkanı sağlayan oldukça pratik bir RGB LED'dir. Bu dairesel kartta 12 adet NeoPixel, birbirine peş peşe bağlı şekilde bulunmaktadır. Ayrıca kartları da peşpeşe bağlayarak çoklu LED halkaları kullanabilirsiniz.



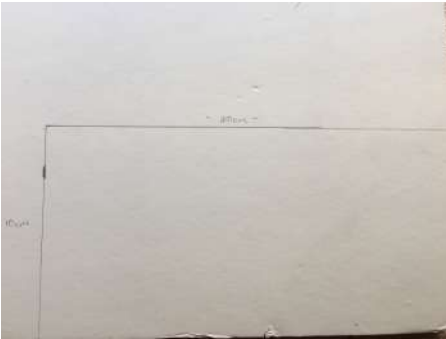
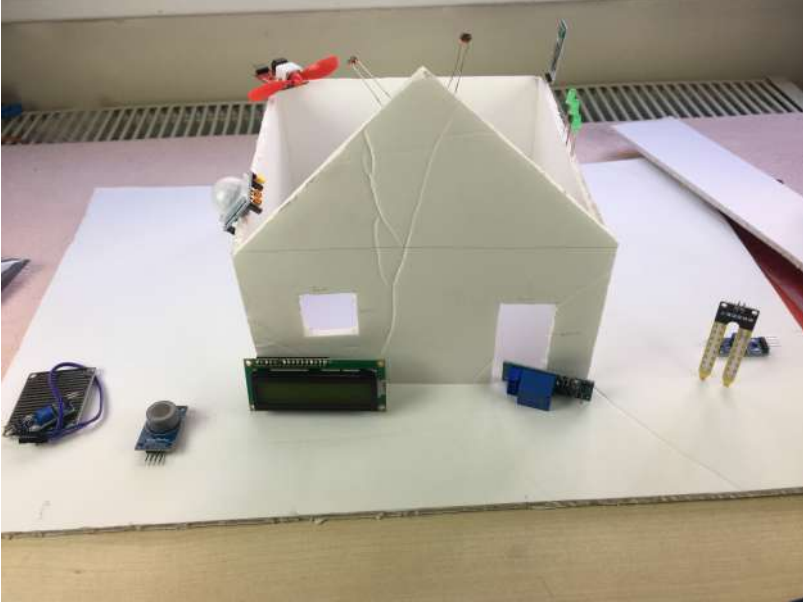
Pır Sensörü

PIR sensörleri, bir ortamda oluşan canlı hareketini algılamak için kullanılan sensörlerdir. Bu minik boyutlu sensör, çeşitli elektronik, robotik ve hobi uygulamalarında rahatça kullanabileceđiniz, Arduino başta olmak üzere bir çok mikrodenetleyeci platformu ile beraber kullanılabilir modüldür.

Karanlık Olunca Yanan Ledler

Evinizin tasarımı tamamen size aittir. Setimizle beraber gelen dekotamızı istediğimiz ölçülerde bir maket bıçağıyla kolayca kesip evimizi oluşturalım. Bizim tasarımıımız görseldeki Gibidir.

Sensörlerimizi dilediğimiz yerlere çift taraflı bant veya sıcak silikonla yapıştıralım. Tasarımıımıza ufak bir göz gezdirelim.

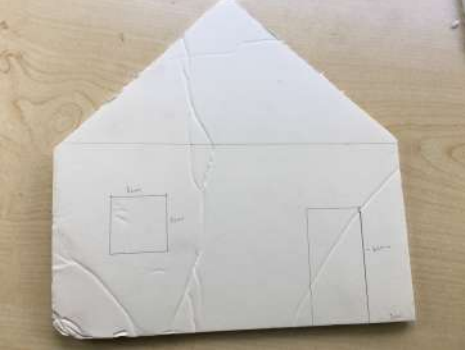


Evimizin duvarlarını oluşturacağımız 20* 10 cm parçalarımızı kesiyoruz.

Karanlık Olunca Yanan Ledler



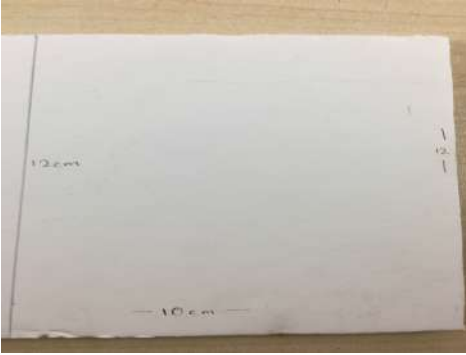
Duvarlar için parçaları hazırladığımıza göre kapı pencere ve çatının bulunacağı parçamızı kesiyoruz.



Kapı ve pencere görselliğini oluşturacağımız parçaları çizip kesiyoruz.



Evimizi zemine yapıştırmadan önce masa üzerinde uygun yapıştırıcı ile yapıştırıyoruz. Yapışıp kuruduktan sonra arta kalan parça üzerine koyup garajımızın tasarımına geçiyoruz.



Garaj için 12* 10 cm lik parçalar ayarladım. İlk parçamızı tamamen kesiyoruz diğer 3 parçayı sadece bir birinden biraz ayrılacak şekilde kesiyoruz (yaralama şeklinde). Daha sonra bunları birleştiriyoruz.



Gerekli Malzemeler

- Arduino Uno
- 2 adet LED
- Ldr
- 10k 330R direnç
- Yeteri kadar jumper kablo

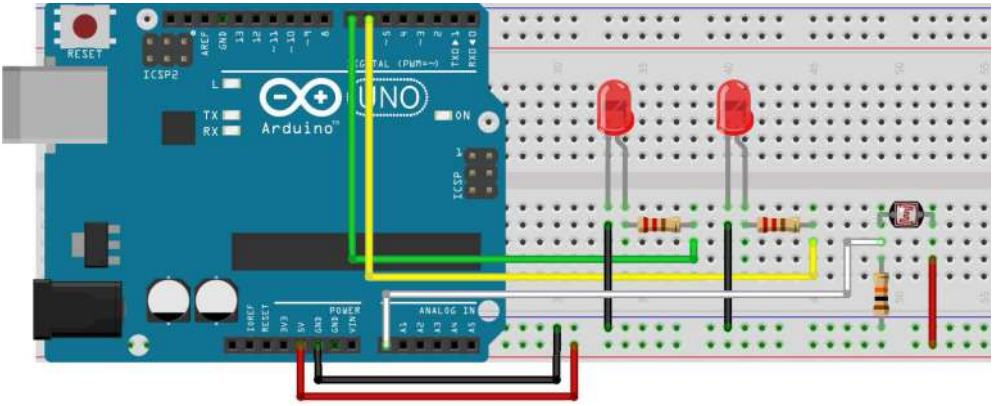
Evimizin bahçesinde akşam saatlerinde otomatik olarak yanan ledler tasarlamak istiyoruz. Bu sayede bahçeden eve doğru geçerken yolumuz aydınlık olacaktır.

Karanlık Olunca Yanan Ledler

Önce biraz ldr'nin çalışma prensibinden bahsedelim. Üzerine düşen ışık şiddeti ile ters orantılı bir çalışma prensibine sahiptir. Yani üzerine düşen ışık şiddeti arttıkça sahip olduğu direnç değeri azalır, ışık şiddeti azaldıkça sahip olduğu direnç değeri artar. LDR'ler sahip oldukları direnç değerlerinin değişmesi ile bir anahtarlama görevi görürler. Başka bir açıdan bakacak olursak bir optik sensör görevi de görmüş olurlar. Aşırı ısı altında bozulmaktadırlar (Maksimum: 60 °C).

Devremizde ldr 'den okuduğumuz değeri bir koşul içerisinde sokarak karanlık olduğunda ledlerimizi yakacağız. Devremizi aşağıdaki gibi kuralım.

Devre Kurulumu



Ledlerimizin uzun ömürlü olması için 220 ohm'luk dirençler kullanıyoruz. Ldr için 10K direnç kullanıyoruz.

Devremizin kurulumunu yaptığımıza göre kodumuza geçebiliriz.

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



Karanlık Olunca Yanan Ledler

```
int led1 = 7;
int led2 = 6;
int LDR = A0;

void setup()
{
  Serial.begin(9600);
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop()
{
  int ldr_deger = analogRead(LDR);
  Serial.println(ldr_deger);
  if(ldr_deger < 500)
  {
    digitalWrite(7, HIGH);
    digitalWrite(6, HIGH);
  }
  else
  {
    digitalWrite(7, LOW);
    digitalWrite(6, LOW);
  }
}
```

Ortam ışık şiddetini görmemiz için kodu yükledikten sonra seri ekranda ışık şiddetinin değerini görebilirsiniz bu değere göre aşağıdaki kod bloğundaki değeri değiştirebilirsiniz.

```
if(ldr_deger < 500)
```

Mblock Kodu

```
genel Arduino Yükle Arduino BGOyla düzenle

1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9   pinMode(A0, INPUT);
10  pinMode(6, OUTPUT);
11  pinMode(7, OUTPUT);
12 }
13
14 void loop() {
15   if((analogRead(A0) < 500)){
16     digitalWrite(6,1);
17     digitalWrite(7,1);
18   }else{
19     digitalWrite(6,0);
20     digitalWrite(7,0);
21   }
22   _delay(1);
23 }
24
25 void _delay(float seconds){
26   long startTime = millis(); while( ( millis() - startTime ) < ( seconds * 1000 ) ) ;
27 }
```

send encode mode
☒ binary hex ☐ harf harf

recv encode mode
☐ binary hex ☒ harf harf

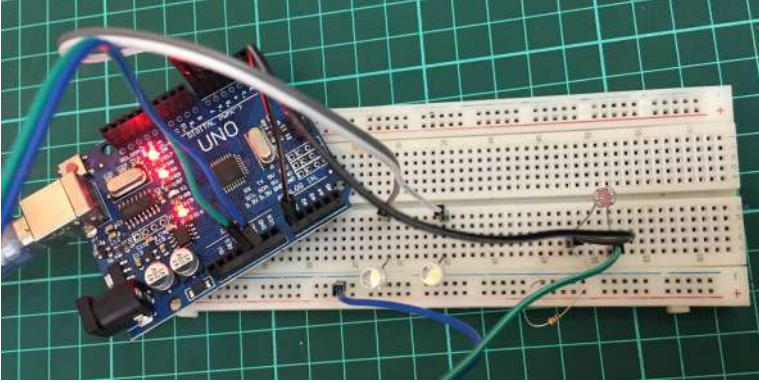
Yolla

Karanlık Olunca Yanan Ledler

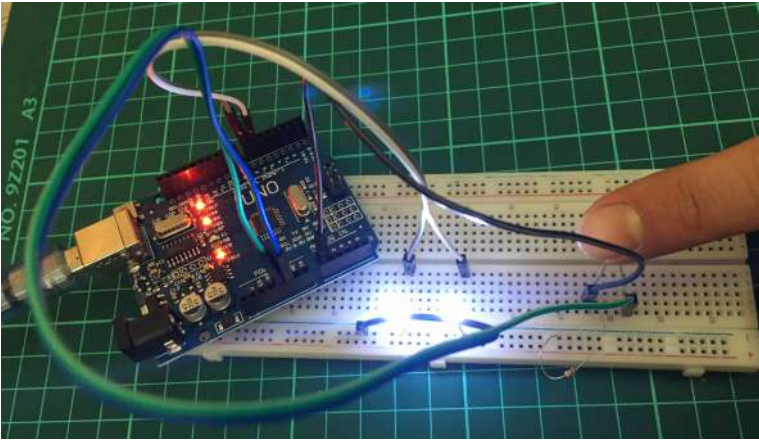
Ortam ışık şiddetini görmemiz için kodu yükledikten sonra seri ekranda ışık şiddetinin değerini görebilirsiniz bu değere göre aşağıdaki kod bloğundaki değeri değiştirebilirsiniz.

```
eğer (A) 0 analog pini oku < 500 ise
```

Komponentleri evimize montajlamadan önce tüm devreleri önce test edip breadboard üzerinde kurulumlarını gerçekleştiriyoruz.



Devremizi kurduk hava aydınlık iken ledler kapalı konumda olucaktır.



Ldr üzerine ışık düşmeyince yani hava kararınca ledler yanacaktır.

Hırsız Alarmı

Gerekli Malzemeler

- Arduino Uno
- 2 adet LED
- Pır Sensörü
- 330R direnç
- Buzzer
- Yeteri kadar jumper kablo

Evimizin bahçesinde akşam saatlerinde otomatik olarak yanan ledler tasarlamak istiyoruz. Bu sayede bahçeden eve doğru geçerken yolumuz aydınlık olacaktır.

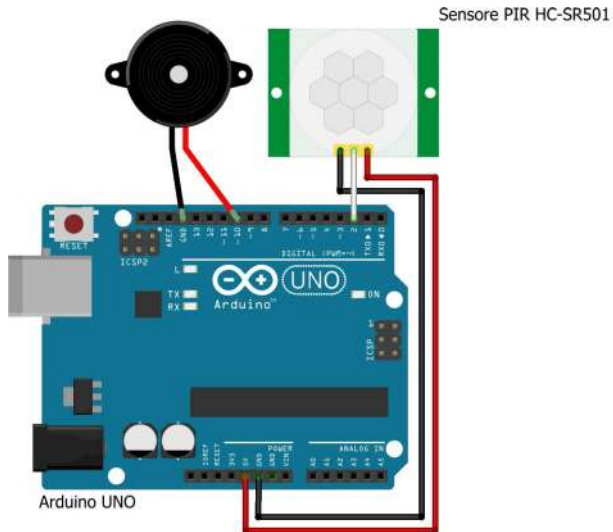
Bu projemizde bir hırsız alarmı tasarlayacağız. Dilerseniz evimiz için bir garaj oluşturup oraya sensörünüzü yerleştirebilirsiniz dilersenizde kasa çekmece gibi yerleri tercih edebilirsiniz.

Amacımız bizim olmadığımız zamanlarda sensörü koyduğumuz yerde bir hareketlilik olursa bir bildirim(alarm) verip bizi uyarmasıdır.

Sensör olarak pır (hareket sensörünü) kullanacağız hareket olursa bize bildirim sağlayıp ordaki hırsız tedirgin etmesi içinde bir buzzer kullanacağız.

Önce pır sensörünü inceleyelim. PIR (pasif kızılötesi) sensörler, tüm nesnelerden yayılan ısıyı ölçmek için kızılötesi algılamayı kullanan, genellikle hırsız alarm sistemlerinde ve kameralarda yer alan kızılötesi ışınımı kullanan devrelerdir. Sensör tarafından herhangi bir ısı veya enerji yayılmadığı için bu sensörler pasif kızılötesi olarak adlandırılırlar.

Devre kurulumu



Hırsız Alarmı

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
const int buzzer= 10; // ledin hangi bacağına bağlandığını tanımladık
const int pirPin = 2; //Pir sensörünün hangi bacağına bağlandığını tanımladık

void setup() {
  pinMode(pirPin, INPUT); //pir sensörünü giriş olarak atadık
  pinMode(10, OUTPUT); // ledimizi çıkış olarak atadık
}

void loop() {
  int pirDeger = digitalRead(pirPin);
  if (pirDeger == HIGH) // hareket algılandığında
  {
    digitalWrite(10, HIGH);
  }
  if (pirDeger == LOW) // hareket algılanmadığında
  {
    digitalWrite(10, LOW);
  }
}
```

Mblock Kodu

The screenshot shows the Mblock software interface. On the left, there is a block-based code editor with the following structure:

- Arduino Programı** (blue block)
- sürekli tekrarla** (yellow loop block)
- seri porta 2 sayısal pini oku yaz** (blue block)
- eğer 2 sayısal pini oku = 1 ise** (green if block)
- 10 sayısal pini YÜKSEK yap** (blue block)
- değilse** (yellow else block)
- 10 sayısal pini DÜŞÜK yap** (blue block)

On the right, the C++ code is displayed:

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9   Serial.begin(115200);
10  pinMode(2, INPUT);
11  pinMode(10, OUTPUT);
12 }
13
14 void loop() {
15   Serial.println(digitalRead(2));
16   if ((digitalRead(2)) == (1)) {
17     digitalWrite(10, 1);
18   } else {
19     digitalWrite(10, 0);
20   }
21   _loop();
22 }
23
24 void _delay(float seconds) {
25   long endTime = millis() + seconds * 1000;
26   while (millis() < endTime) _loop();
27 }
```

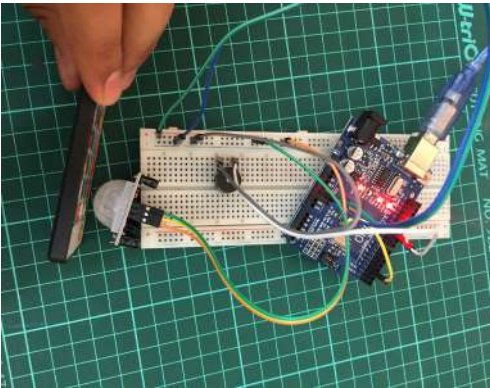
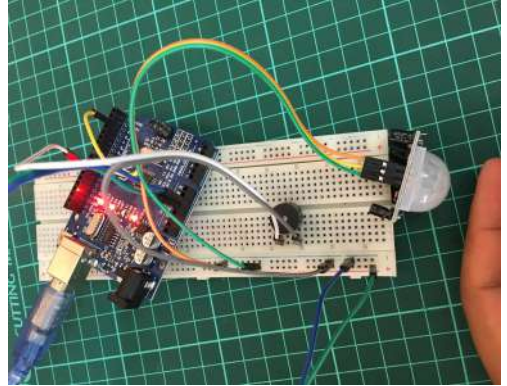
At the bottom, there are settings for the serial communication:

- send encode mode**: ☒ binary hal ☐ harf hal
- recv encode mode**: ☐ binary hal ☒ harf hal
- Yolla** button



Pir sensörü üzerindeki dış kaplamayı çıkartıp bağlantıları daha net öğrenebilirsiniz.

Devremizin kurulumunu gerçekleştiriyoruz.



Pir sensörünün etrafında hareketlenme yapınca veya bir nesne yaklaştınca buzzerin ses verdiğini gözlemleyeceksiniz.

Toprak Nem Sensörü ile Röle Kontrolü

Kitimizde kullandığımız komponentleri maket evimizin haricinde dilersek gerçek hayatta bahçemizde ve evimizdede kullanabiliriz.

Bu projemizde bunu öğrenmiş olacağız. Toprak nem sensörümüzden toprağın içerisindeki nem değerini öğrenip buna bağlı olarak rölemizi çalıştıracağız.

Röleyi burda kullanmamızın amacı bahçe ,tarla veya sera gibi yerlerde 220V'luk bir su motoru ile sulama sistemini nasıl aktif edebileceğimizi göstermektir.

Gerekli Malzemeler

- Arduino Uno
- Toprak nem sensörü
- Röle
- Yeteri kadar jumper kablo

Devremizi kurmadan önce elemanlarımızı biraz tanıyalım.

Toprak nem sensörü, toprağın içerisindeki nem miktarını veya ufak ölçekte bir sıvının seviyesini ölçmek için kullanabileceğiniz bir sensördür.

Nem ölçer problar ölçüm yapılacak ortama batırılarak kullanılır. Toprağın veya içine batırılan sıvının meydana getirdiği dirençten dolayı, prob uçları arasında bir gerilim farkı oluşur. Bu gerilim farkının büyüklüğüne göre de nem miktarı ölçülebilir. Topraktaki nem oranı arttıkça iletkenliği de artmaktadır. Kart üzerinde yer alan trimpot sayesinde hassasiyet ayarı yapılabilmektedir.

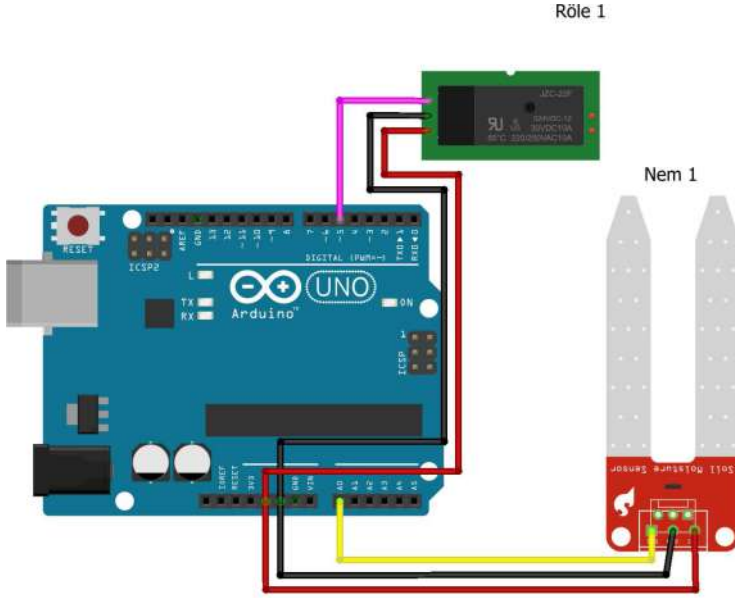
Röle, düşük akımlar kullanarak yüksek akım çeken cihazları anahtarlama görevinde kullanılan devre elemanıdır. Kısaca çalışma prensipleri: rölenin bobinine enerji verildiğinde mıknatıslanan bobin bir armatürü hareket ettirerek kontakların birbirine temasını sağlar ve devrede iletim sağlanmış olur.

Şuan biz sadece rölenin çalışması için gerekli enerjiyi sağlayacağız. Yani 5V ile çalışacağız.

Eğer siz rölenin çıkışına su motoru lamba gibi 220V ile çalışan bir ürün yerleştirecekseniz öncelikle röleyi izole ettiğinizden emin olun.

Toprak Nem Sensörü ile Röle Kontrolü

Devre kurulumu



Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
const int toprak = A0;
int role = 5;

void setup() {
  Serial.begin(9600);
  pinMode(role, HIGH);
}

void loop() {
  int olcum_sonucu = analogRead(toprak);
  Serial.print("toprak ıslaklığı = ");
  Serial.println(olcum_sonucu);
  delay(500);

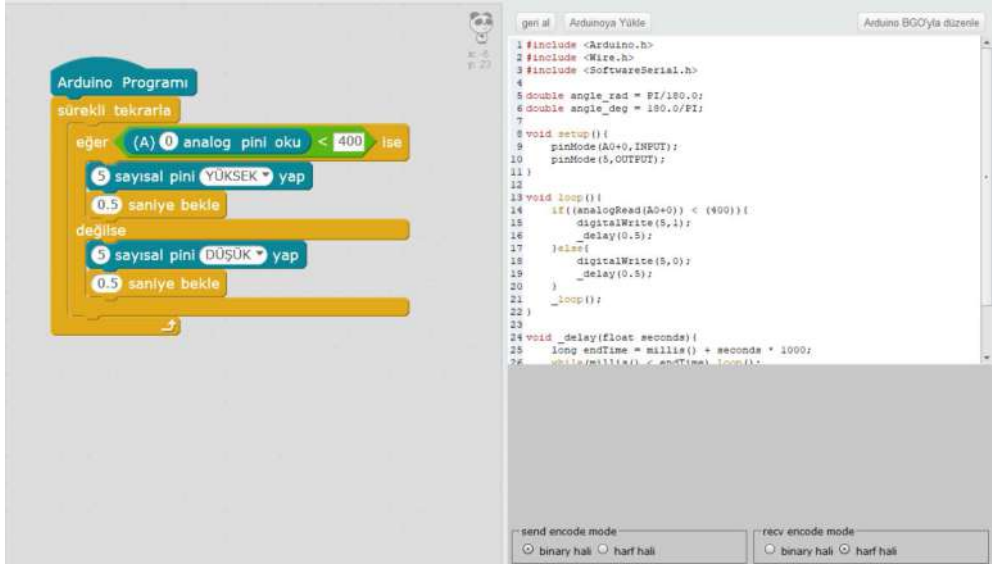
  if (olcum_sonucu <= 400) {
    digitalWrite(5, HIGH);
  } else {
    digitalWrite(5, LOW);
  }
}
```

Toprak Nem Sensörü ile Röle Kontrolü

Toprağın nem değerini görmemiz için kodu yükledikten sonra seri ekranda toprağın nem değerini görebilirsiniz bu değere göre aşağıdaki kod bloğundaki değeri değiştirebilirsiniz.

```
if (olcun_sonucu <=400) {
```

Mblock Kodu



The screenshot shows the Arduino IDE interface. On the left, the Mblock code is displayed, featuring a 'sürekli tekrarla' (loop forever) block. Inside the loop, there is an 'eğer' (if) block with the condition '(A) 0 analog pini oku < 400 ise'. If the condition is true, it executes '5 sayısal pini YÜKSEK yap' (set digital pin 5 to HIGH) followed by a '0.5 saniye bekle' (wait 0.5 seconds) block. If the condition is false, it executes '5 sayısal pini DÜŞÜK yap' (set digital pin 5 to LOW) followed by a '0.5 saniye bekle' (wait 0.5 seconds) block. On the right, the C++ code is shown, which includes the necessary libraries, pin definitions, and the logic implemented in the setup and loop functions. The code sets pin 5 as an output and checks the analog value of pin A0. If the value is less than 400, it sets pin 5 to HIGH; otherwise, it sets pin 5 to LOW. A delay of 500 milliseconds is used between checks.

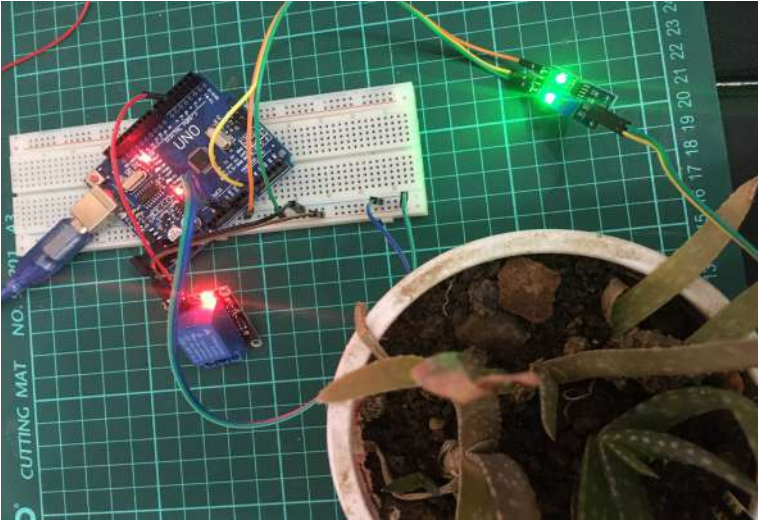
Toprağın nem değerini görmemiz için kodu yükledikten sonra seri ekranda toprağın nem değerini görebilirsiniz bu değere göre aşağıdaki kod bloğundaki değeri değiştirebilirsiniz.

eğer (A) 0 analog pini oku < 400 ise

Toprak Nem Sensörü ile Röle Kontrolü



Toprak nem sensörümüzü bir toprağın içersine sabitliyoruz ben bir saksı kullandım.



Daha sonra devremizi kurup toprak nem sensörünün değeriine göre rölemizin aktifleştigini gözlemliyoruz.

Sıcaklık veya nem değerine göre fan kontrolü yapmak. Bu projeyi çok fazla alanda kullanabiliriz. Bir odanın ısısını kontrol edebiliriz bir bilgisayarın sıcaklığı arttığında otomatik olarak fanları devreye sokup performansını arttırabiliriz bir serada havalandırma yapabiliriz gibi onlarca proje aklınıza gelebilir.

Günümüz oyuncu bilgisayarlarında güçlü performanslar için güçlü ekran kartları işlemciler ramler tercih edilir. Bu kompenantlar güçlendikçe dışarıya doğru yaydıkları sıcaklıkta artar. Hem kendilerinin zarar görmemesi hemde kullanıcı konforu için fanların doğru zamanlarda devreye girmesi gerekmektedir. Projemizle beraber hem bu sistemlerin nasıl çalıştığını hemde oda sıcaklığı ve neme göre fan çalıştırmayı öğreneceğiz.

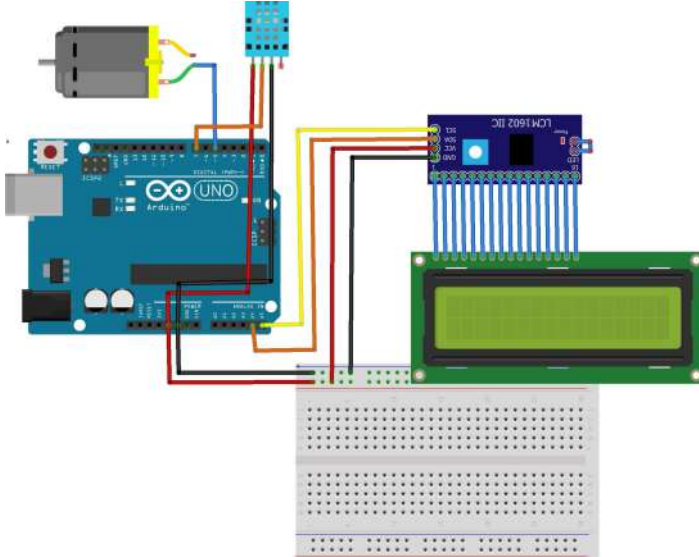
Anlık olarak sıcaklık ve nem değerini lcd ekranda görüp sistemimizi daha cazip bir hale getireceğiz.

Gerekli Malzemeler

- Arduino Uno
- Dht11 Sıcaklık Nem Sensörü
- Fan
- Yeteri kadar jumper kablo

Kitimizle beraber gelen fanın üzerinde kendi motor sürücüsü bulunduğu için ek bir transistör veya motor sürücü kullanmayacağız. Devremizin kurulumuna geçebiliriz.

Devre Kurulumu



Bizim kullandığımız fan motorunun kendi motor sürücüsü olduğu için ekstra bir sürücü kullanmıyoruz. Görsel sadece temsildir bağlantıları ekstra olarak aşağıya yazıyorum.

VCC=>VCC

GND=>GND

INA=>5

Motor pinlerinin hangisine enerji verirse o yöne döner biz bu yüzden motorun sadece 1 pinini kullandık.

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
#include <dht11.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x3f, 16, 2);
DHT11 DHT11;
int motor = 5;

void setup()
{
  lcd.begin(); pinMode(5, OUTPUT);
  DHT11.attach(7); //dht nim = bacağının bağlantı pini
}

void loop()
{
  lcd.setCursor(0, 0);
  int chk = DHT11.read();

  lcd.print("  Nem (%)" ); //Nem yüzdesini ekrana yazdırıyoruz
  lcd.print((float)DHT11.humidity, 2);
  lcd.setCursor(0, 1);
  lcd.print(" Sıcaklık (C)"); //Sıcaklığı santigrad olarak ekrana yazdırıyoruz
  lcd.print((float)DHT11.temperature, 2);

  if (DHT11.temperature>35){
    digitalWrite(5,1);
  }
  else {
    digitalWrite(5,0);
  }
}
```

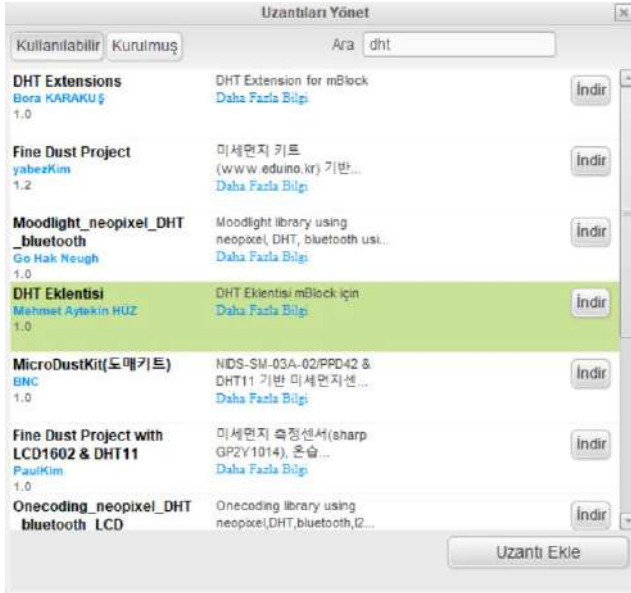
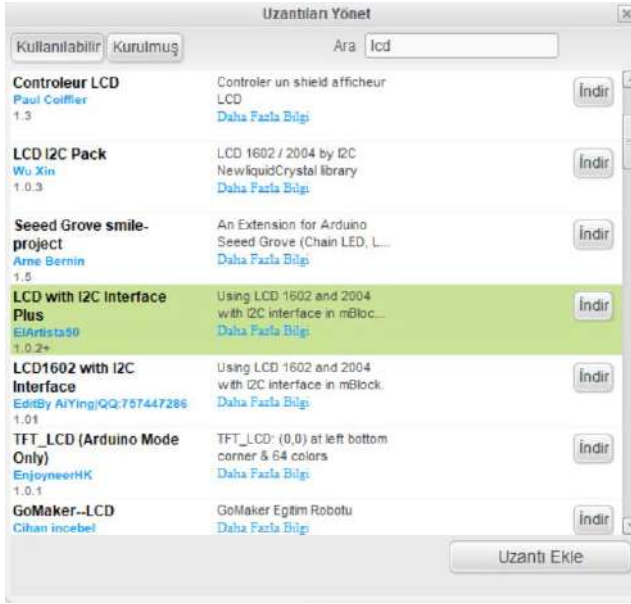
I2C modüllerimizin farklılık göstermesiyle beraber eğer kodu yüklediğinizde lcd ekranınıza yazılar gelmiyor ise aşağıdaki kod satırında 0x3f yerine 0x27 yazınız.

```
LiquidCrystal_I2C lcd(0x3f, 16, 2);
```

Ben sıcaklık değerine göre sıcaklık 35 derecenin üzerine çıkınca fanın çalışmasını istedim siz derseniz nem değerine göre veya hem sıcaklık hem nem değerine göre fanınızı kontrol edebilirsiniz.

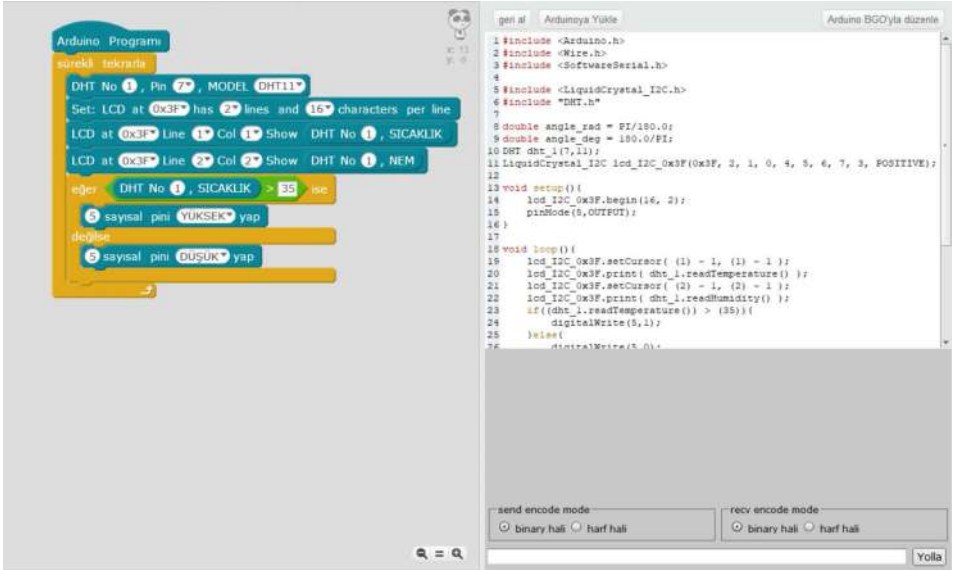
Dht11 / LCD / Fan Kontrolü

Mblock Kodları



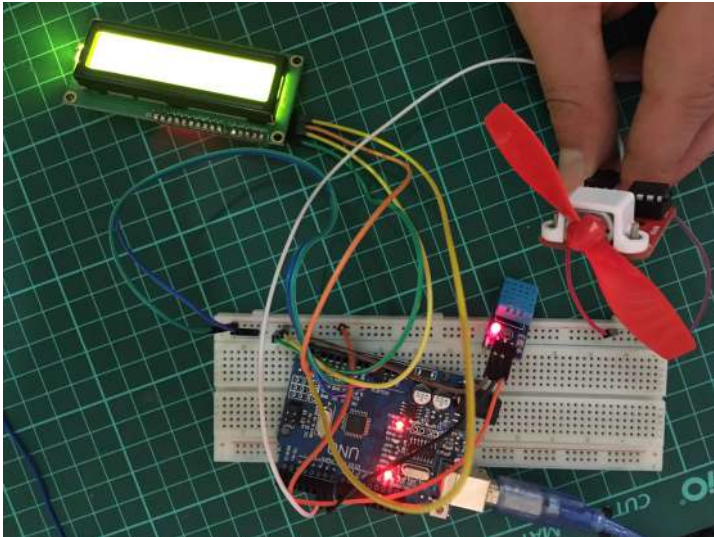
Kodlarımıza geçmeden önce Uzantıları Yönet kısmından dht ve lcd eklentilerini ekleyelim.

Dht11 / LCD / Fan Kontrolü



Dht11 den aldığımız sıcaklık değerini değiştirip motorumuzu ona göre değiştirebiliriz. Bunun için alttaki kod satırındaki değeri değiştirebilirsiniz.

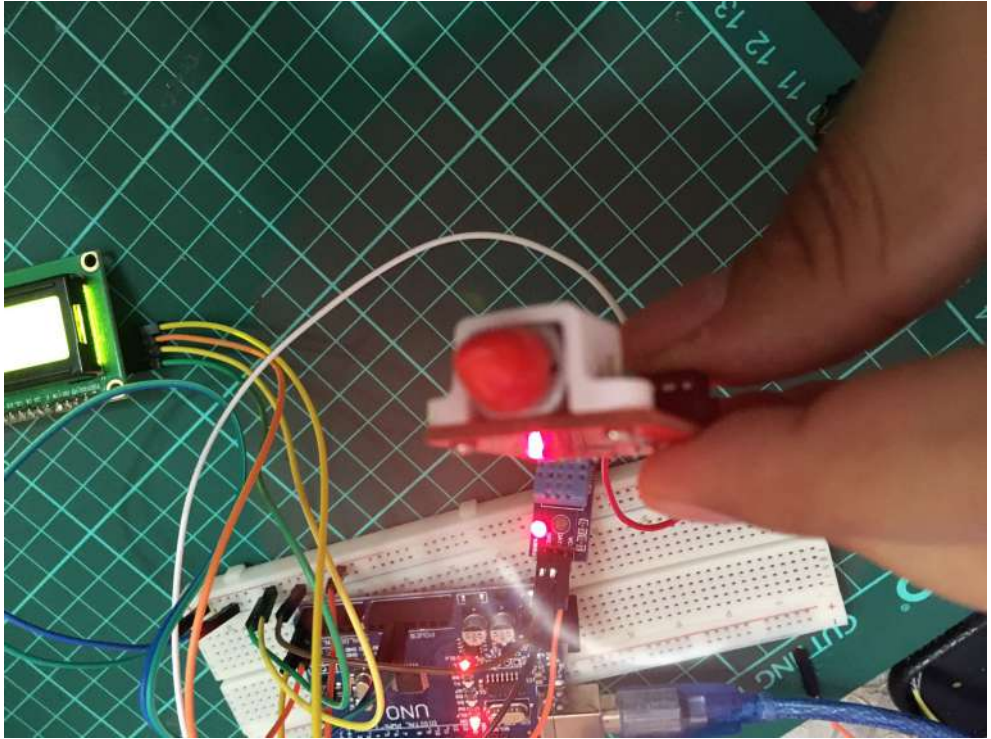
eğer DHT No 1 , SICAKLIK > 35 ise



Dht11 / LCD / Fan Kontrolü

Devremizi kuralım kurduktan sonra lcd ekran üzerindeki deęerleri gözlemleyelim.

Ortamın şuan ki sıcaklık deęeri 31 C olduęu için fan çalışmıyor.



Fanın çalıştığını gözlemleyeceksiniz.

Odamızın içerisinde bulunan elektrikli cihazları ve aydınlatmaları cep telefonumuzla kontrol edebilmek akıllı evlerin olmazsa olmazlarındandır.

Oturduğumuz yerden istediğimiz şeyleri kontrol etmek her zaman konforludur. Günümüz dünyasında bu yöne doğru hızlı bir şekilde ilerlemektedir.

Bluetooth modülü çekim mesafesinden dolayı oda içerisinde ve ev koşullarında idealdir.

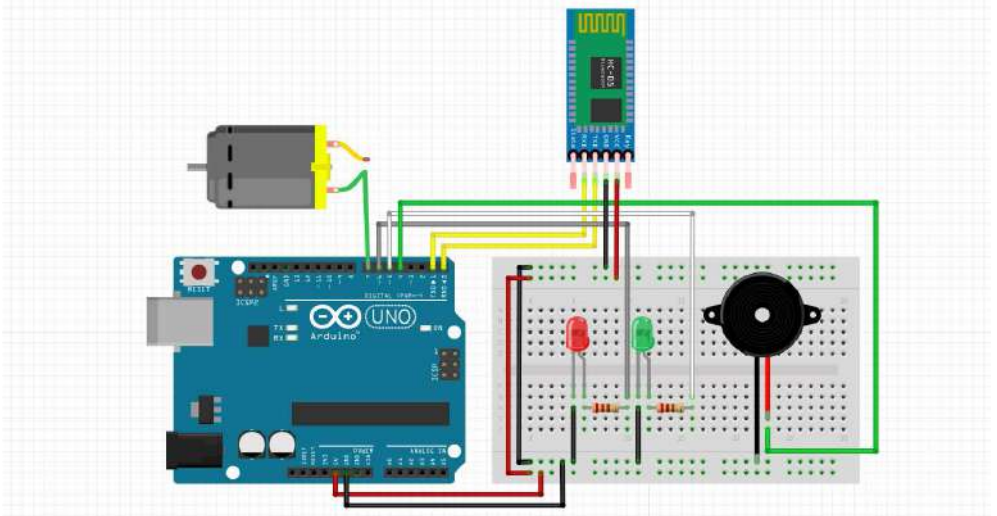
Ama daha sonra dışarıda aynı komponentleri kontrol etmek istersek bluetooth çekim mesafesinden dolayı işimizi görmeyecektir bu kez devreye bir wi-fi modülü ekleyip kontrol edebiliriz.

Gerekli Malzemeler:

- Arduino Uno
- 2 Adet Led
- Fan
- Buzzer
- Yeteri kadar jumper kablo

Kendi mobil uygulamamızı yazarak kendimize göre özgün bir tasarım yapabiliriz ve istediğimiz kadar komponent kontrolünü sağlayabiliriz. Önce devremizi kuralım daha sonra mobil uygulamamızı yazalım.

Devre Kurulumu



APK Yazılımı

Bu projede mitappinventor2 ile kendi mobil uygulamamızı yaparak ayrı ayrı 2 ledi kontrol edeceğiz alarm olarak kullandığımız buzzer'ı yönetebileceğiz ve klima görevini görmesi için fanımızı açıp kapatacağız.

Hazır uygulamaların aksine kendi uygulamamız tamamen bize özel olup dilediğimiz gibi geliştirebileceğiz.

Daha sonra yeni modüller ve komponentler eklemek istersek devremize uygulamamıza yeni eklemeler yapmak oldukça basit olacaktır. Bu sayede herhangi bir bağımlılığımız söz konusu olmayacaktır tamamen özgün işler yapabileceğiz.

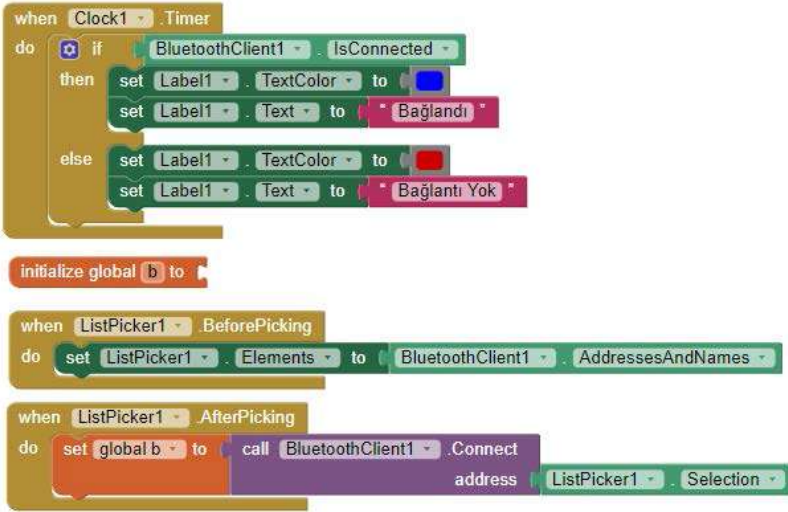
Hadi Başlayalım! Devre şemasına bakıp kurulumumuzu yapmayı unutmayalım.

App Inventor Nedir?

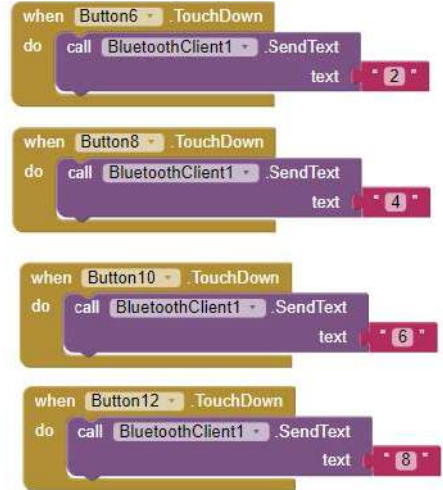
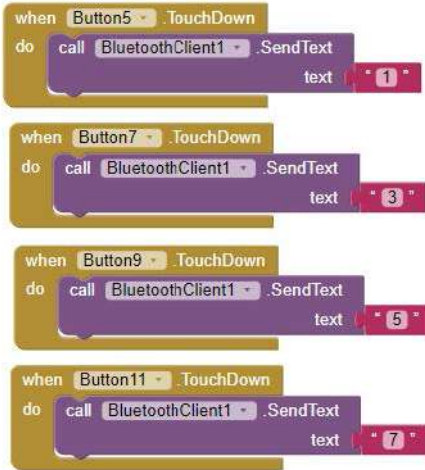
MIT tarafından geliştirilen ve bloklar ile “sürükle-bırak” mantığında programlanabilen Android uygulama geliştirme programıdır. Kullanmanız için Android programlamaya dair hiçbir bilginiz olmasına ve bilgisayarımıza herhangi bir program indirmemize gerek yoktur.Projeden kısaca bahsetmek gerekirse önce MIT App Inventor 2 isimli bir uygulama geliştirme programında arayüz oluşturuyoruz. Oluşturduğumuz uygulama yoluyla Android cihazımızı Arduino'ya bağladığımız Bluetooth modülü ile eşleştiriyoruz. Bu eşleşme sayesinde Robotumuzu Android cihaz üzerinden kontrol edebiliyoruz.



Ben tasarımı bu şekilde yaptım siz dilediğiniz gibi değiştirebilirsiniz.



Uygulamamızın bluetooth bağlantı kısmını ayarlıyoruz. Burda bluetooth'a bağlandığında ve bağlanmadığındaki durumlarını görmemizi sağlayan kod blokları mevcut.



On off durumlarını belirlediğimiz butonların sayısal karşılıklarını burada belirliyoruz. Kodlarımızda burdaki rakamları kullanacağız.

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>

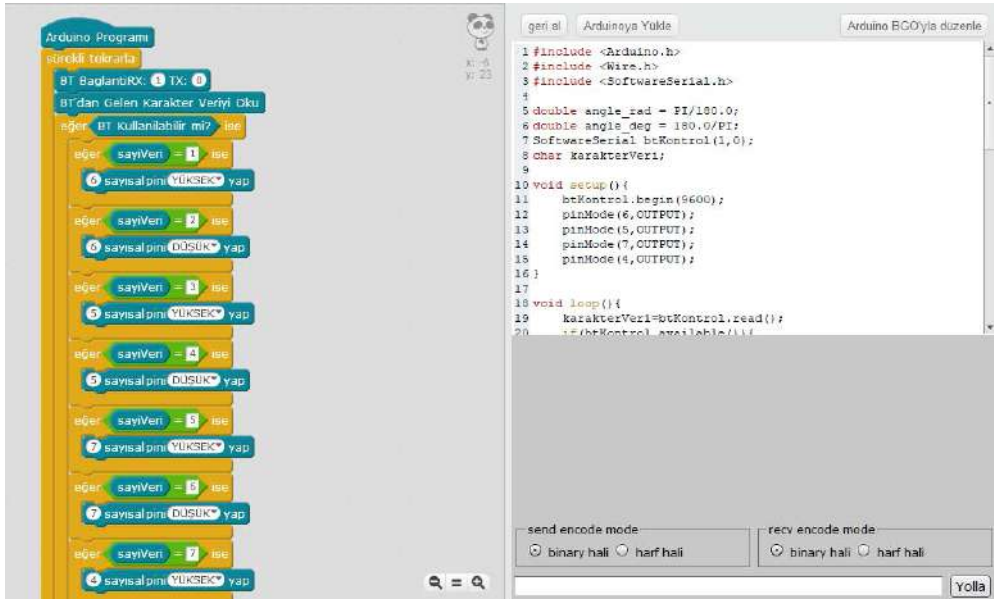


```
int motor = 7;
int led1 = 6;
int led2 = 5;
int buzzer = 4;
void setup() {
  Serial.begin(9600);
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(4, OUTPUT);
}
void loop() {
  if(Serial.available())
  {
    int data=Serial.read();
    delay(100);

    if(data=='1') {
      digitalWrite(6,HIGH);
    }
    if(data=='2') {
      digitalWrite(6,LOW);
    }
    if(data=='3') {
      digitalWrite(5,HIGH);
    }
  }
}
```

```
if(data=='4') {
  digitalWrite(5,LOW);
}
if(data=='5') {
  digitalWrite(7,HIGH);
}
if(data=='6') {
  digitalWrite(7,LOW);
}
if(data=='7') {
  digitalWrite(4,HIGH);
}
if(data=='8') {
  digitalWrite(4,LOW);
}
}
```


Mblock Kodu



Arduino Programı

steckle tablasına

BT BağlantıRX: 4 TX: 0

BT'dan Gelen Karakter Veriyi Oku

eğer BT kullanılabiliyor mı? ise

eğer sayıVeri = 1 ise

6 sayısal pini YÜKSEK yap

eğer sayıVeri = 2 ise

6 sayısal pini DÜŞÜK yap

eğer sayıVeri = 3 ise

5 sayısal pini YÜKSEK yap

eğer sayıVeri = 4 ise

5 sayısal pini DÜŞÜK yap

eğer sayıVeri = 5 ise

7 sayısal pini YÜKSEK yap

eğer sayıVeri = 6 ise

7 sayısal pini DÜŞÜK yap

eğer sayıVeri = 7 ise

4 sayısal pini YÜKSEK yap

geni al Arduino'ya Yükle

Arduino BCG'yle düzenle

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7 SoftwareSerial btKontrol(1,0);
8 char KarakterVeri;
9
10 void setup() {
11   btKontrol.begin(9600);
12   pinMode(6,OUTPUT);
13   pinMode(5,OUTPUT);
14   pinMode(7,OUTPUT);
15   pinMode(4,OUTPUT);
16 }
17
18 void loop() {
19   KarakterVeri=btKontrol.read();
20   if(btKontrol.available()) {
```

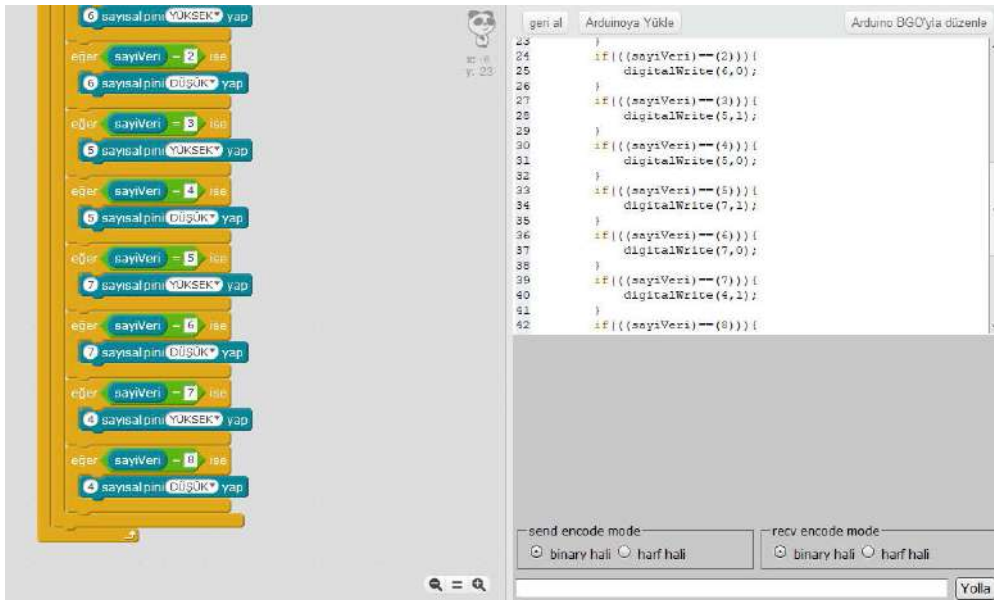
send encode mode

binary hali harf hali

recv encode mode

binary hali harf hali

Yolla



6 sayısal pini YÜKSEK yap

eğer sayıVeri = 2 ise

6 sayısal pini DÜŞÜK yap

eğer sayıVeri = 3 ise

5 sayısal pini YÜKSEK yap

eğer sayıVeri = 4 ise

5 sayısal pini DÜŞÜK yap

eğer sayıVeri = 5 ise

7 sayısal pini YÜKSEK yap

eğer sayıVeri = 6 ise

7 sayısal pini DÜŞÜK yap

eğer sayıVeri = 7 ise

4 sayısal pini YÜKSEK yap

eğer sayıVeri = 8 ise

4 sayısal pini DÜŞÜK yap

geni al Arduino'ya Yükle

Arduino BCG'yle düzenle

```
23 }
24 if(((sayıVeri) == (2))) {
25   digitalWrite(6,0);
26 }
27 if(((sayıVeri) == (3))) {
28   digitalWrite(5,1);
29 }
30 if(((sayıVeri) == (4))) {
31   digitalWrite(5,0);
32 }
33 if(((sayıVeri) == (5))) {
34   digitalWrite(7,1);
35 }
36 if(((sayıVeri) == (6))) {
37   digitalWrite(7,0);
38 }
39 if(((sayıVeri) == (7))) {
40   digitalWrite(4,1);
41 }
42 if(((sayıVeri) == (8))) {
```

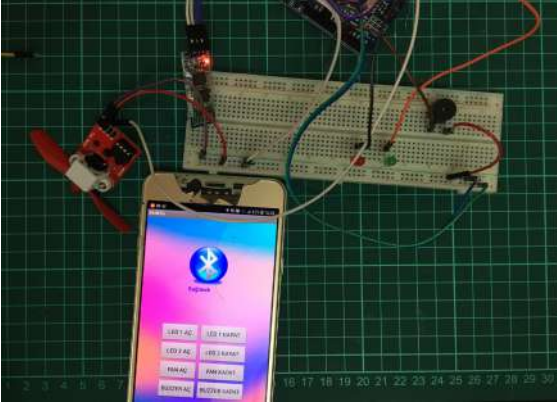
send encode mode

binary hali harf hali

recv encode mode

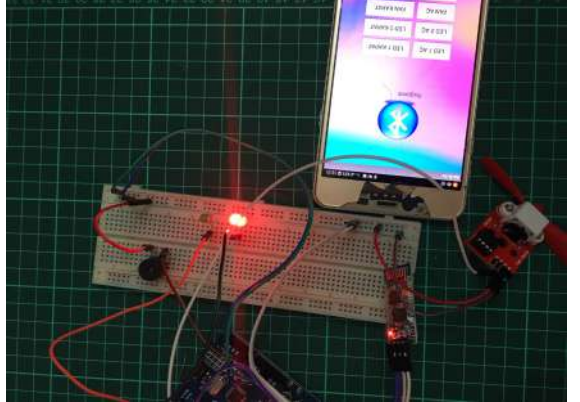
binary hali harf hali

Yolla



Mobil uygulamayı telefonumuza kurup devremizin kurulumunu gerçekleştireyoruz.

Led 1 aç tuşuna basınca kırmızı ledimiz yandı.



Led 2 aç tuşuna basıncada kırmızı ledimizin yandığını gözlemledik bu şekilde fan ve buzzerda çalıştırıp Kapatabiliriz.

Duman - Gaz Dedektörü

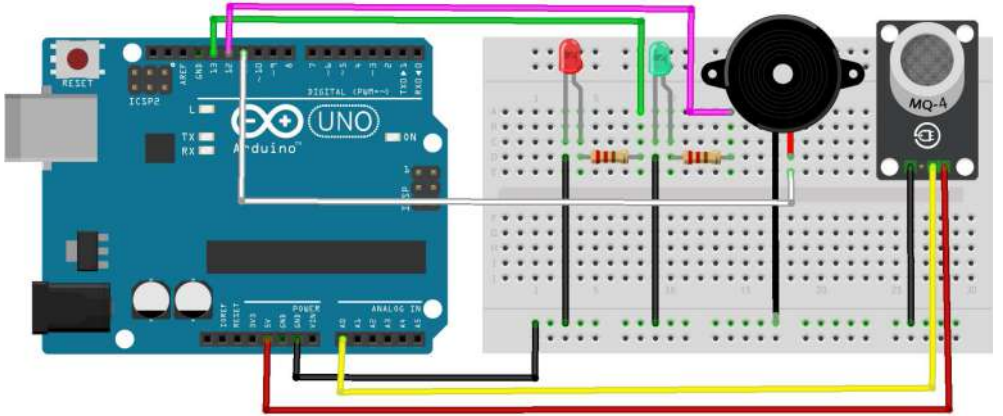
Bu projede Gaz sensöründen gelen veriler sayesinde ortamın gaz ve duman seviyesini ölçebileceksiniz, belli seviye üzerinde buzzer çalışacak. Bu sayede tehlike durumun önceden anlayı müdahale edebileceğiz.

Gerekli Malzemeler:

- Arduino Uno
- 2 adet Led
- 2 adet 220 ohm Direnç
- MQ2 Gaz sensörü
- Buzzer
- Jumper Kablolar

Kitimizle beraber gelen kırmızı ve yeşil ledlerimizde olaya dahil edip daha efektif bir proje yapalım. Eğer gaz seviyesi yüksek ise hem buzzer aktif olsun hemde kırmızı ledi yakalım. Eğer gaz seviyesi düşük(normal) ise buzzer pasif durumda kalıp yeşil ledimizi yakalım bir problemin olmadığını göstermiş olacağız bu sayede. Hadi devremizi kuralım.

Devre Şeması



Gaz Sensörümüzü istersek analog istersek digital olarak bağlayıp kodlayabiliriz. Ben analog kısmı tercih ettim.

Devremizi kurduysak kodlarımıza geçebiliriz.

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
int led1 = 13;
int led2 = 12;
int buzzer = 11;
int sensor = A0;

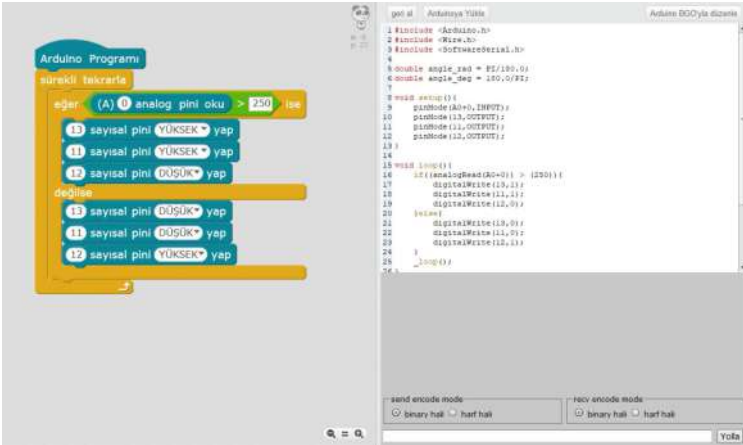
void setup() {
  Serial.begin(9600);
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
}

void loop() {
  int deger=analogRead(A0);
  Serial.println(deger);
  if (deger>250) {
    digitalWrite(13,1);
    digitalWrite(11,1);
    digitalWrite(12,0);
  }
  else {
    digitalWrite(13,0);
    digitalWrite(11,0);
    digitalWrite(12,1);
  }
}
```

Gaz sensörünün değerini görmeniz için kodu yükledikten sonra seri ekranda gaz değerini görebilirsiniz bu değere göre aşağıdaki kod blogundaki değeri değiştirebilirsiniz.

```
if (deger>250) {
```

Mblock Kodu



Gaz sensörünün değerini görmeniz için kodu yükledikten sonra seri ekranda gaz değerini görebilirsiniz bu değere göre aşağıdaki kod blogundaki değeri değiştirebilirsiniz.

eğer (A) 0 analog pini oku > 250 ise

Sensörleri Birleştirelim

7 proje boyunca bir çok sensörün kullanımını nasıl değer okuduğumuzu okuduğumuz bu değerlere göre nasıl işlemler yapabileceğimizi ayrı ayrı öğrendik. Şimdi öğrendiğimiz bu projeleri aynı sistem üzerinde nasıl kullanabileceğimizi öğrenelim.

İlk önce sensörlerin çalışma mantıklarını öğrendik bu sayede istediğimiz gibi sensörden değer okuyarak yeni projeler gerçekleştirebiliriz. Mevcut projelere eklenti yapmak projeyi sıfırdan yapmaktan her zaman daha zordur. Çünkü çalışan bir sistem üzerindeki algoritmayı anlayıp yeni oluşturacağınız algoritma ile sisteme eklenti yapmış olacaksınız.

Şimdi birkaç projeyi birleştirelim ve yeni bir sistem ortaya çıkaralım.

Bir yanda hırsız alarmı devredeyken bir yandan gaz sensörümüz devrede olsun. Ek olarakta sıcaklık nem değerine göre odamızdaki fan çalışsın.ve karanlık olunca bahçe aydınlatmalarımız devreye girsin. Tabi bahçemizdeki bitkileri düşünmemiz gerekiyor bundan dolayıda toprak nem sensörümüzüde sisteme eklemeliyiz.

İlk başta projeyi okurken bile kafamız biraz karışmış olabilir ama daha önce oluşturduğumuz üç kodu doğru şekilde birleştirip düzenlemeler yaparak bu işin üstesinden gelebiliriz.

Hadi kolları sıvayıp önce devremizi kuralım.

Devre Kurulumu

Çok fazla komponent olduğu için bağlantı şeması aşırı derece karmaşık gelebilir düşüncesi ile bu kez bir şema eklemek yerine bağlantıları yazmayı tercih ettim.

DHT11

VCC => VCC

GND => GND

S => 7

Pır Sensörü

VCC => VCC

GND => GND

S => 10

Toprak Nem Sensörü

VCC => VCC

GND => GND

S => A0

Fan => 2

Bahçe Led1 => 3

Bahçe Led2 => 4

Oda Led1 => 5

Oda Led2 => 6

Buzzer => 8

Role => 8

Gaz Sensörü

VCC => VCC

GND => GND

S => A2

LDR

VCC => VCC

GND => GND

S => A3

Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
#include <dht11.h>
DHT11 DHT11;

const int toprak = A0;
const int pir    = 10;
const int gaz    = A2;
const int ldr    = A3;

int fan        =2;
int bahce_led1 =3;
int bahce_led2 =4;
int oda_led1   =5;
int oda_led2   =6;
int buzzer     =9;
int role       =9;

void setup()
{
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(9,OUTPUT);
  DHT11.attach(7);
}

void loop()
{
  int pir_deger    = digitalRead(pir);
  int toprak_deger = analogRead(toprak);
  int gaz_deger    = analogRead(gaz);
  int ldr_deger    = analogRead(ldr);

  int chk = DHT11.read();
  if (DHT11.temperature>35){
    digitalWrite(fan,1);
  }
  else {
    digitalWrite(fan,0);
  }

  if (toprak_deger <=400){
    digitalWrite(role, 1);}

  else {
    digitalWrite(9,0);
  }
  if(ldr_deger < 500)
  {
    digitalWrite(bahce_led1, HIGH);
    digitalWrite(bahce_led2, HIGH);
  }
}
```


Mblock Kodları

Arduino Program

Herkesi bilgilendir

DHT No 1, Pin 2, MODEL DHT11

eğer DHT No 1, SICAKLIK > 35 ise

sayısal pini YUKSEK yap

değelse

sayısal pini DUSUK yap

eğer (A) 0 analog pini oku < 400 ise

sayısal pini YUKSEK yap

değelse

sayısal pini DUSUK yap

eğer (A) 2 analog pini oku < 500 ise

sayısal pini YUKSEK yap

değelse

sayısal pini DUSUK yap

geri al

Arduinoya Yükle

Arduino BGO'ya düzenle

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 #include "DHT.h"
6
7 double angle_rad = PI/180.0;
8 double angle_deg = 180.0/PI;
9 DHT dht_1(7,11);
10
11 void setup() {
12   pinMode(2,OUTPUT);
13   pinMode(A0+0,INPUT);
14   pinMode(5,OUTPUT);
15   pinMode(A0+3,INPUT);
16   pinMode(3,OUTPUT);
17   pinMode(4,OUTPUT);
18   pinMode(10,INPUT);
19   pinMode(8,OUTPUT);
20   pinMode(A0+2,INPUT);
21   pinMode(6,OUTPUT);
22 }
23 }
```

değelse

sayısal pini DUSUK yap

sayısal pini DUSUK yap

eğer 10 sayısal pini oku > 1 ise

sayısal pini YUKSEK yap

değelse

sayısal pini DUSUK yap

eğer (A) 0 analog pini oku > 250 ise

sayısal pini YUKSEK yap

sayısal pini DUSUK yap

değelse

sayısal pini YUKSEK yap

sayısal pini DUSUK yap

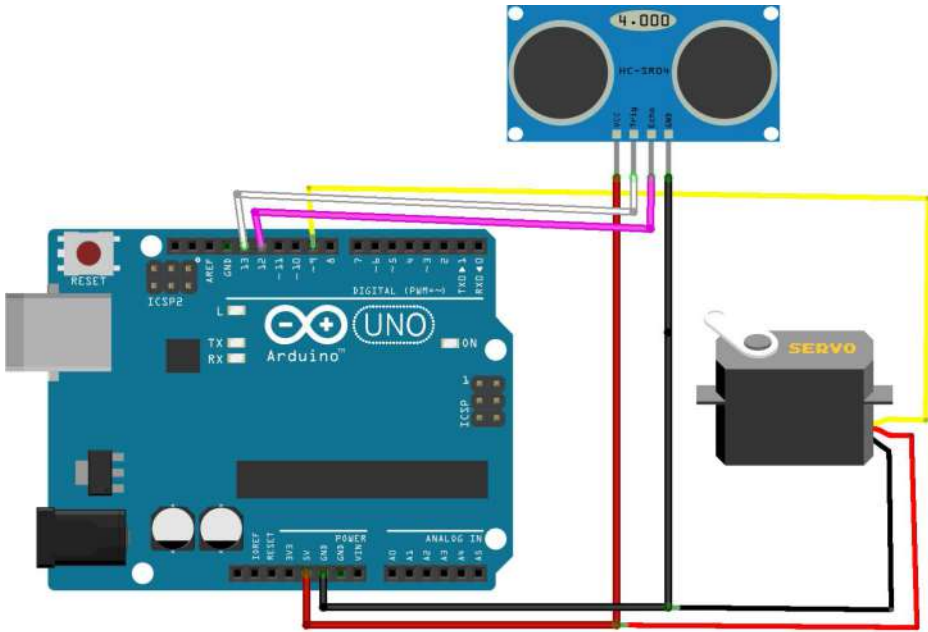
```
35 }
36 if((analogRead(A0+3)) < (500)) {
37   digitalWrite(3,1);
38   digitalWrite(4,1);
39 }else{
40   digitalWrite(3,0);
41   digitalWrite(4,0);
42 }
43 if(((digitalRead(10))==(1))) {
44   digitalWrite(8,1);
45 }else{
46   digitalWrite(8,0);
47 }
48 if((analogRead(A0+2)) > (250)) {
```

Otomatik Açılan Garaj Kapısı

Aracımız garaj kapısına yaklaştığında otomatik olarak kapıyı açan sistemi yapacağız. Arabayla garaj kapısına kadar gelip araçtan inip kapıyı açmak günümüz teknolojisine pekte yakışmayan bir eylem olur veya bir kumanda ile açmak için kumandayı aramak algılaması için bazen şekilden şekile girmek çokta istenen bir durum değildir.

Biz bu devremizde bir ultrasonik mesafe sensörü kullanacağız mesafemiz 30 cm in altına gelince servo motorumuzu 90 konumuna getirip aracımızı park edebileceğiz. Sonuç olarak bir protatip yaptığımız için servo motor daha efektif olacaktır. Ama dilerseniz tente şeklinde bir garaj tasarlayıp step motor ilede yapabilirsiniz. Kolları sıvayıp devremizi kuralım.

Devre Kurulumu



Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



Otomatik Açılan Garaj Kapısı

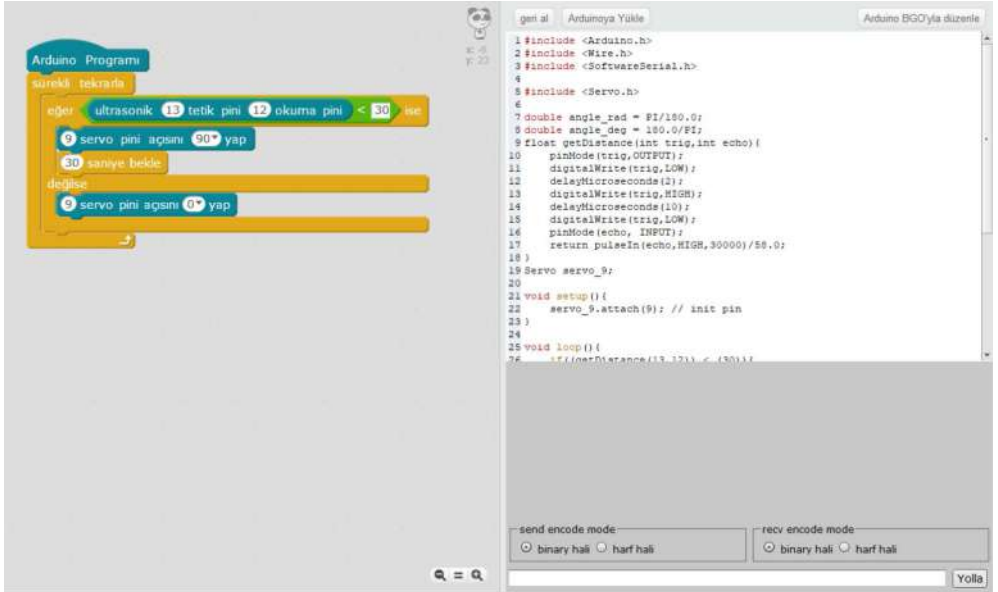
```
#include<Servo.h>
Servo myservo;
const int trig = 13;
const int echo = 12;
int sure;
int mesafe;

void setup() {
  Serial.begin(9600);
  pinMode (trig , OUTPUT);
  pinMode (echo , INPUT);
  myservo.attach(9);
}

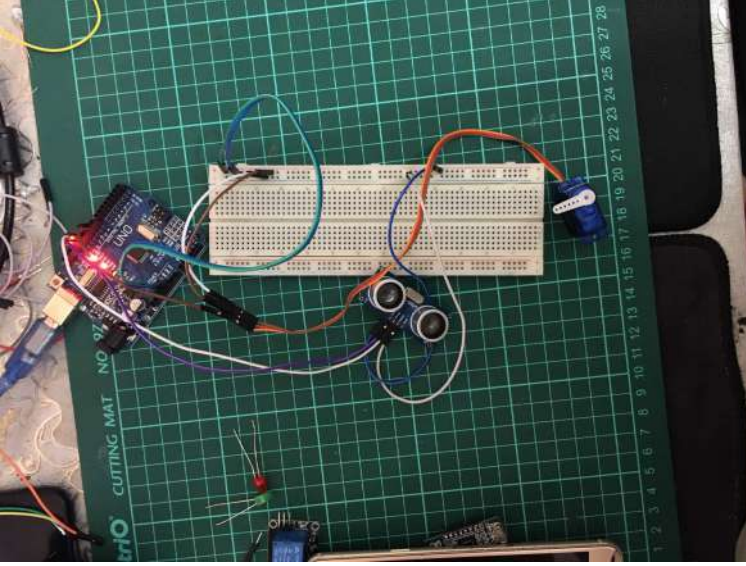
void loop() {
  digitalWrite(trig , HIGH);
  delayMicroseconds(1000);
  digitalWrite(trig, LOW);

  sure = pulseIn(echo ,HIGH);
  mesafe = (sure/2) / 28.5;
  Serial.println(mesafe);
  if(mesafe<30){
    myservo.write(90);
    delay(30000);
  }
  else{
    myservo.write(0);
  }
}
```

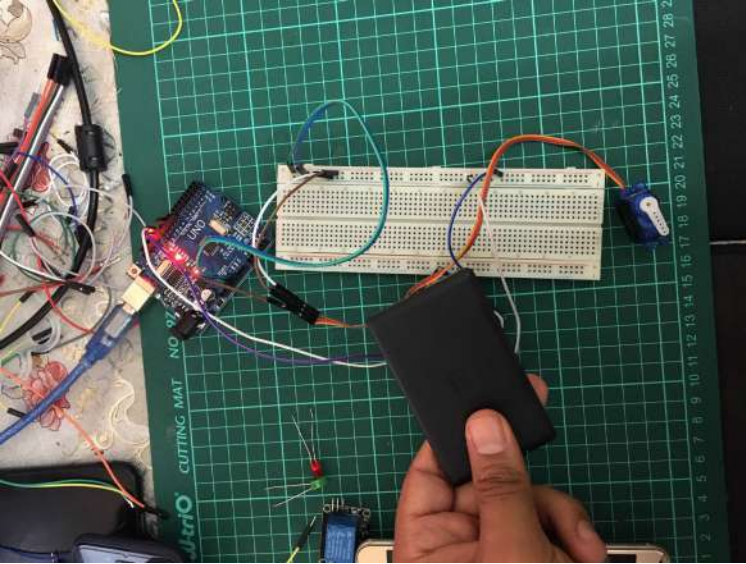
Mblock Kodu



The image shows the Mblock software interface. On the left, the 'Arduino Programı' (Arduino Program) window displays a block-based code: a 'sürekli tekrarda' (while loop) block containing an 'eğer' (if) block. The 'if' block checks 'ultrasonik 13 tetik pini 12 okuma pini < 30 ise' (if ultrasonic pin 13 trigger pin 12 read pin < 30). If true, it executes 'servo pini açısını 90° yap' (set servo pin angle to 90°), followed by a '30 saniye bekle' (wait 30 seconds) block, and then 'servo pini açısını 0° yap' (set servo pin angle to 0°). On the right, the 'Arduinoya Yükle' (Upload to Arduino) window shows the compiled C++ code, which matches the block-based code shown in the first image. At the bottom, there are settings for 'send encode mode' and 'recv encode mode', both set to 'binary hal'.



Devremizi kuralım ve mesafe sensörümüzün değışkeni 30 cm nin altındayken servonun açısını gözlemleyelim.



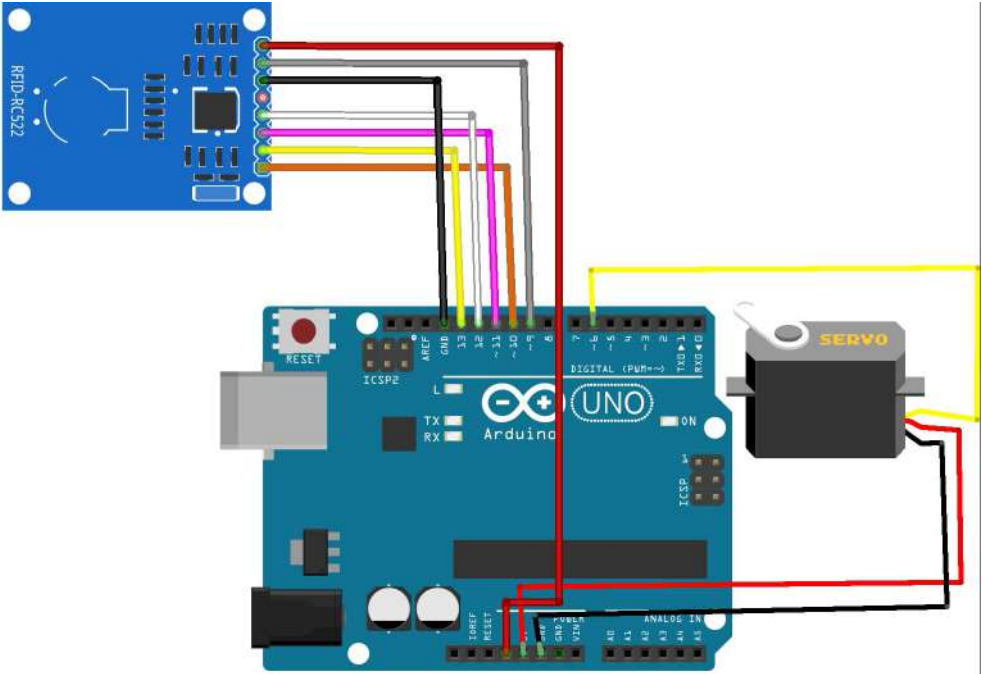
Mesafe sensörünün önüne herhangi bir engel koyduğumuzda yani mesafe değışkenimizi 30 cm nin altına çektiğimizde servo motorumuzun açısının değışimini gözlemleyelim.

Artık neredeyse tüm apartman kapılarında gördüğümüz ve oldukça pratik olan rfid kart sistemlerini bizde projemize dahil edelim. Rfid kart sistemlerini kimlik olarak düşünebiliriz. Nasıl bir kimliğimiz var ve biz hayatımızdaki tüm işlemleri oradaki bilgilerimizle yapıyosak rfid kartlardada bir ID var ve bu ID göre belli işlemler gerçekleştirebiliyoruz. Sık kullanım alanlarından bir hastahanelerdir doktor hemşire veya hastahane personeli pratik olarak odalara girmek için kullanır ve onların dışında yetkisi olmayanların girmesini istemedikleri için bu sistem hastahaneler için biçilmiş kaftandır.

Özellikle büyük şehirlerde otobüs kartı , kent kart gibi kart isimleriyle muhattap olmuşuzdur. Burda yine bizim kendi ID miz üzerine bilgiler girilir ve Rfid sisteme okutunca işlemler gerçekleştirilmiş olur. Bunun gibi onlarca örnek sayabiliriz.

Biz devremizde evimize girmek bu sistemi tercih ettik. Siz bunu bir kasa olarak düşünebilirsiniz veya herhangi bir kilit sistemine entegre edebilirsiniz. Hadi başlayalım.

Devre Şeması



Kodun tamamına kare kod ya da kısa link üzerinden ulaşabilirsiniz.

<https://www.kisa.link/OU0X>



```
#include <DHT11.h>
DHT11 DHT11;

const int toprak = A0;
const int pir    = 10;
const int gaz    = A2;
const int ldr    = A3;

int fan          =2;
int bahce_led1  =3;
int bahce_led2  =4;
int oda_led1    =5;
int oda_led2    =6;
int buzzer      =8;
int role        =9;

void setup()
{
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(9,OUTPUT);
  DHT11.attach(7);
}

void loop()
{
  int pir_deger    = digitalRead(pir);
  int toprak_deger = analogRead(toprak);
  int gaz_deger    = analogRead(gaz);
  int ldr_deger    = analogRead(ldr);

  int chk = DHT11.read();
  if (DHT11.temperature>35){
    digitalWrite(fan,1);
  }
  else {
    digitalWrite(fan,0);
  }

  if (toprak_deger <=400){
    digitalWrite(role, 1);
  }
  else {
    digitalWrite(5,0);
  }
  if(ldr_deger < 500)
  {
    digitalWrite(bahce_led1, HIGH);
    digitalWrite(bahce_led2, HIGH);
  }
}
```


Rfid Kart Sistemleri

```
else
{
  digitalWrite(bahce_led1, LOW);
  digitalWrite(bahce_led2, LOW);
}

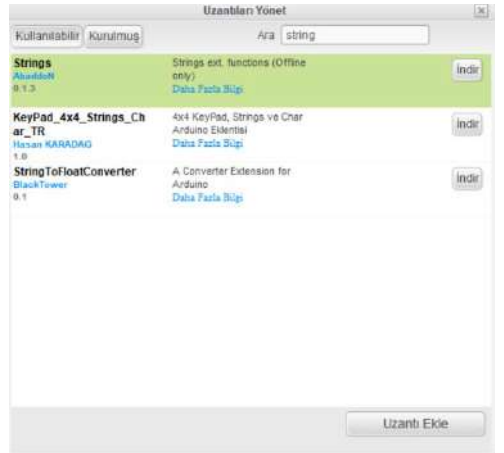
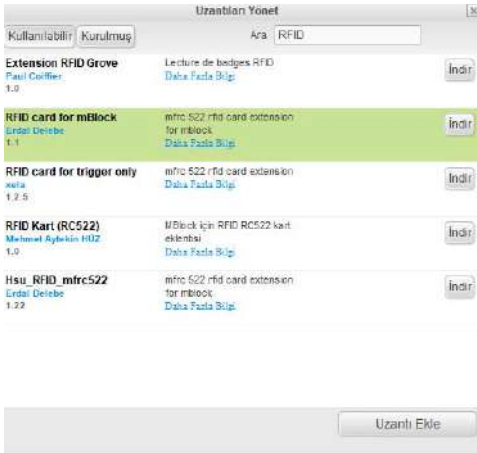
if (pir_deger == HIGH)
{
  digitalWrite(buzzer, HIGH);
}
if (pir_deger == LOW)
{
  digitalWrite(buzzer, LOW);
}

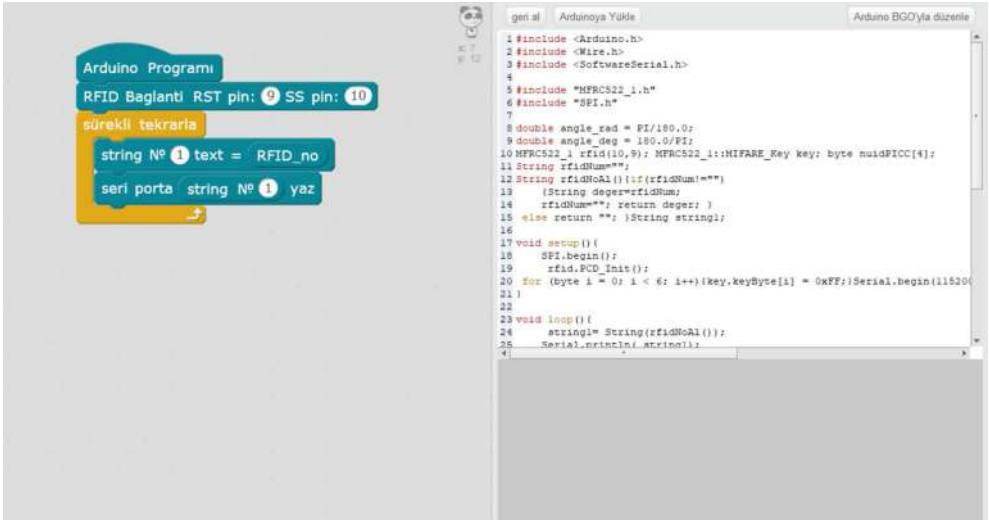
if(gaz_deger>250){
  digitalWrite(oda_led1,1);
  digitalWrite(oda_led2,0);
}

else {
  digitalWrite(oda_led1,0);
  digitalWrite(oda_led2,1);
}
}
```

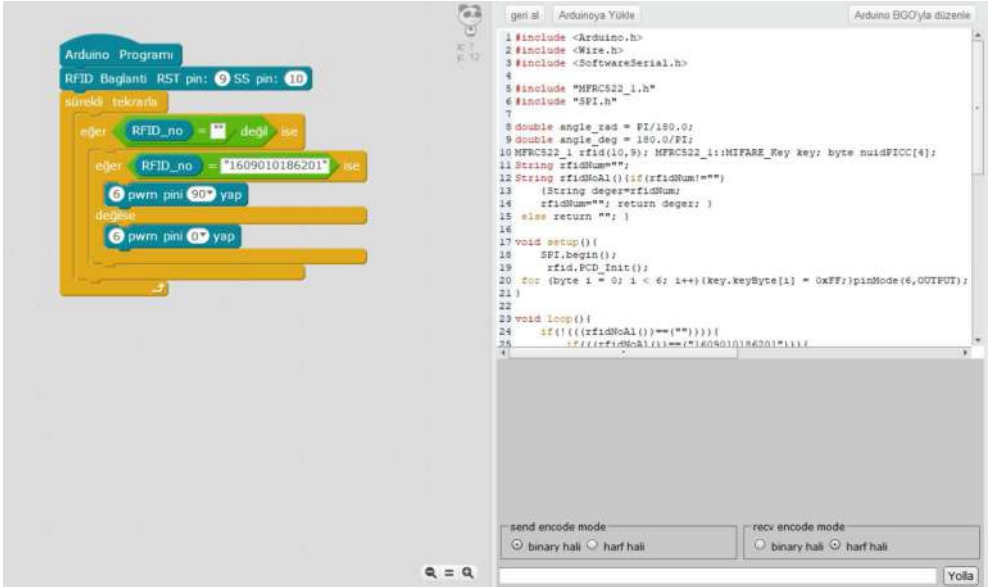
Mblock Kodları

Kodlarımıza geçmeden önce Uzantıları Yönet kısmından Rfid card for mBlock ve Strings eklentilerini ekleyelim.

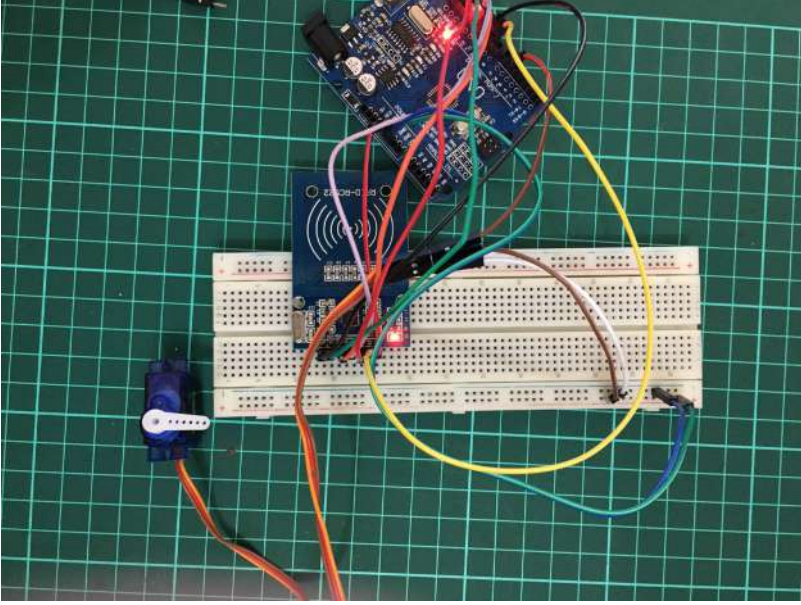




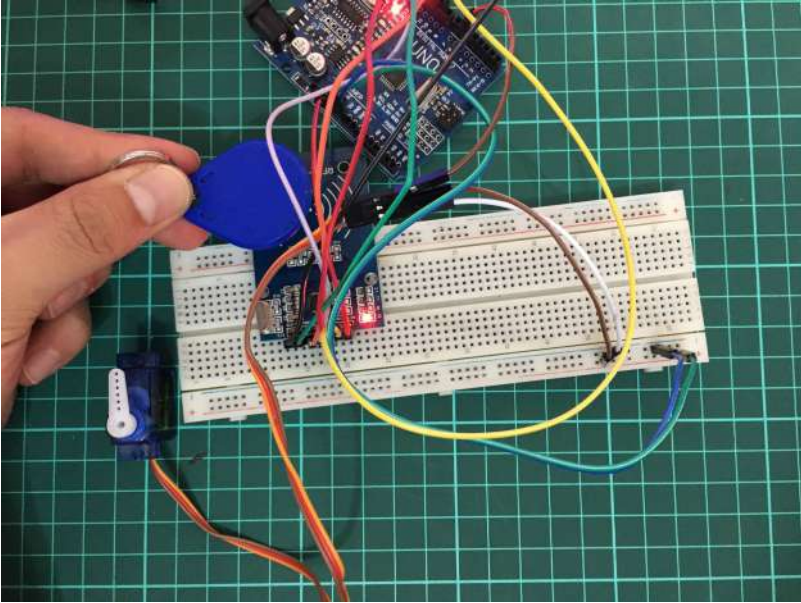
Önce RFID kart üzerindeki değerlerimizi öğrenmek için örnekteki kodu arduinomuza yüklüyoruz.



Rfid kimliğini girdikten sonra kodumuzu tekrar yüklüyoruz. Siz kendi rfid kimliğinizi yüklemelisiniz.



Devremizi kuralım Rfid kart okutmadan önce servomuzun açısı 0 derecede olacaktır.



Doğru Rfid kartımızı okutunca giriş izni verileceğinden servo motorumuz 90 dereceye gelecektir.



youtube.com/robotistan

FORUM

robotistan



forum.robotistan.com

BLOG

robotistan



maker.robotistan.com

Robotistan Elektronik Ticaret AŞ

Muhammed Ali ÖZEN (İçerik) - Mehmet AKÇALI (Editör) - (Mehmet Nasır KARAER (Grafik)
info@robotistan.com - www.robotistan.com
Phone: 0850 766 0 425