

İçerik 1

- **Arduino Nedir ?**
- **Arduino'da Kullanılan Genel Komutlar**
- **Arduino'da LED Uygulaması**
- **Arduino'da Karaşimşek Uygulaması**
- **Arduino'da Trafik Lambası Uygulaması**
- **Bazı Programlama Komutları**
- **Arduino'da Buton Uygulaması**
- **Arduino'da Serial Monitör Kullanımı**
- **Arduino'da Analog Veri Okuma**
- **Arduino'da Potansiyometre Kullanımı**
- **Arduino'da LED Parlaklığı Uygulaması**
- **Arduino'da LDR Kullanımı**
- **Arduino'da Servo Motor Kullanımı**

İçerik 2

- **Potansiyometre ile Servo Motor Kullanımı**
- **Arduino'da Sıcaklık Sensörü Kullanımı**
- **Arduino'da Mesafe Sensörü Kullanımı**
- **Arduino'da Hareket Sensörü Kullanımı**
- **Arduino'da L298N Motor Sürücü Kullanımı**
- **Arduino'da LCD Ekran Kullanımı**
- **Arduino'da RGB LED Kullanımı**
- **Arduino'da Buzzer Kullanımı**
- **Arduino'da Joystick Kullanımı**
- **Arduino'da Yağmur – Su Sensörü Kullanımı**
- **Arduino'da CN-70 Sensörü Kullanımı**
- **Arduino'da Bluetooth Modülü Kullanımı**

Arduino Nedir?

- Arduino bir G/Ç kartı ve Processing/Wiring dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur.
- Arduino tek başına çalışan interaktif nesnelere 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 bilgileri mevcuttur.

Arduino ile Neler Yapılabilir?

Arduino ile Hayatta Bulunan Çoğu Şeyleri Yapabilirsiniz.

Örnek Olarak;

- Robotik kol
- Uzaktan Kumanda
- Drone(Çok Kanatlı Helikopter)
- RC Tank, RC Araba
- Alarm
- Şifreli Kapı Kilidi
- Boy Ölçüm Cihazı
- Uzaktan kontrollü Araçlar vb. şeyleri yapmak mümkün.

Arduino'da Kullanılan Genel Komutlar

- **Void setup ()**
 - Pinlerin Giriş mi Çıkış mı olacağını bu kısımda karar veririz ayrıca Serial monitör ekranına ekleyebiliriz.
 - İlk değerler ve bir kere çalışacak kodlar yazılır.
- **Void loop ()**
 - Sürekli tekrarlanarak sırasıyla çalışacak kodların yazıldığı alandır.

Arduino'da Kullanılan Genel Komutlar

- **DigitalWrite**

- Bu komut ile pinlerimize 0 V ya da 5 V güç veririz. Örnek olarak;

- Kullanımı → `digitalWrite(değişken, HIGH)`

- Kullanımı → `digitalWrite(değişken, LOW)`

- `DigitalWrite(LED, HIGH)` → Burada LED isimli değişkene güç verdik.

- `DigitalWrite(motor, LOW)` → Burada Motor isimli değişkeni durdurduk.

Arduino'da Kullanılan Genel Komutlar

- **AnalogWrite**

- Bu komutla 0 ve 5 V arası değerleri vermemizi sağlıyor. Gelen ve giden elektrik akımı her zaman 0-255 arasındadır. Bu komutun yararı ise bazı bileşenlerin hızıyla oynamamızı sağlıyor ya da konumlanmasını sağlıyor.
- Örneğin motorun yavaş, orta hızlı veya çok hızlı olarak sürülmesini sağlıyor. Ya da bir servo motorun alacağı konumu ayarlama işlemini gerçekleştirebiliyoruz.

- Kullanımı → `AnalogWrite(değişken, LOW)`

- `AnalogWrite(motor,255)` → Motora 5V verdik.
- `AnalogWrite(motor,204)` → Motora 4V verdik.

Arduino'da Kullanılan Genel Komutlar

- **DigitalRead**

- Gelen 0-5 V değerlerini okumamızı sağlar.
- Digital olarak 5V=1'e 0V ise = 0'a eşittir.

- Kullanımı → `digitalRead(Değişken)`

- `DigitalRead(blueetooth);` → Bluetooth'dan gelen verileri okur.
- `DigitalRead(potansiyometre);` → Potansiyometreden gelen verileri okur.

Arduino'da Kullanılan Genel Komutlar

- **AnalogRead**
 - Bu komut ise gelen 0-5V arasındaki değerleri okumamızı sağlar.
- **Kullanımı → analogRead(değişken);**
 - analogRead(blueetooth); → Bluetooth'dan gelen verileri okur.
 - analogRead(potansiyometre); → Potansiyometreden gelen verileri okur.

Arduino'da Kullanılan Genel Komutlar

- **Delay();**
 - Komut ile bekleme süresini ayarlıyoruz.
- **Kullanımı → delay(beklenecek zaman)**
 - `delayMicroseconds(100);` → 100 milisaniye bekliyor
 - `delaySeconds(100);` → 100 saniye bekliyor
 - `delay(100);` → eğer “seconds ya da microseconds vb.” yazılmaz ise Arduino Microseconds olarak algılar.

Arduino'da Kullanılan Genel Komutlar

- **Serial.begin(haberleşme_hızı);**
 - Bu komut ile seri haberleşmeyi açıyoruz. Bu serial haberleşmeden Metin, Bilgi, Sayı Gönderilebilir ya da Alabiliriz. Genellikle haberleşme hızı 9600'dur.
- **Kullanımı → Serial.begin(haberleşme_hızı);**
 - Serial.begin(9600); → 9600 Baund hızında seri haberleşmeyi başlattık.

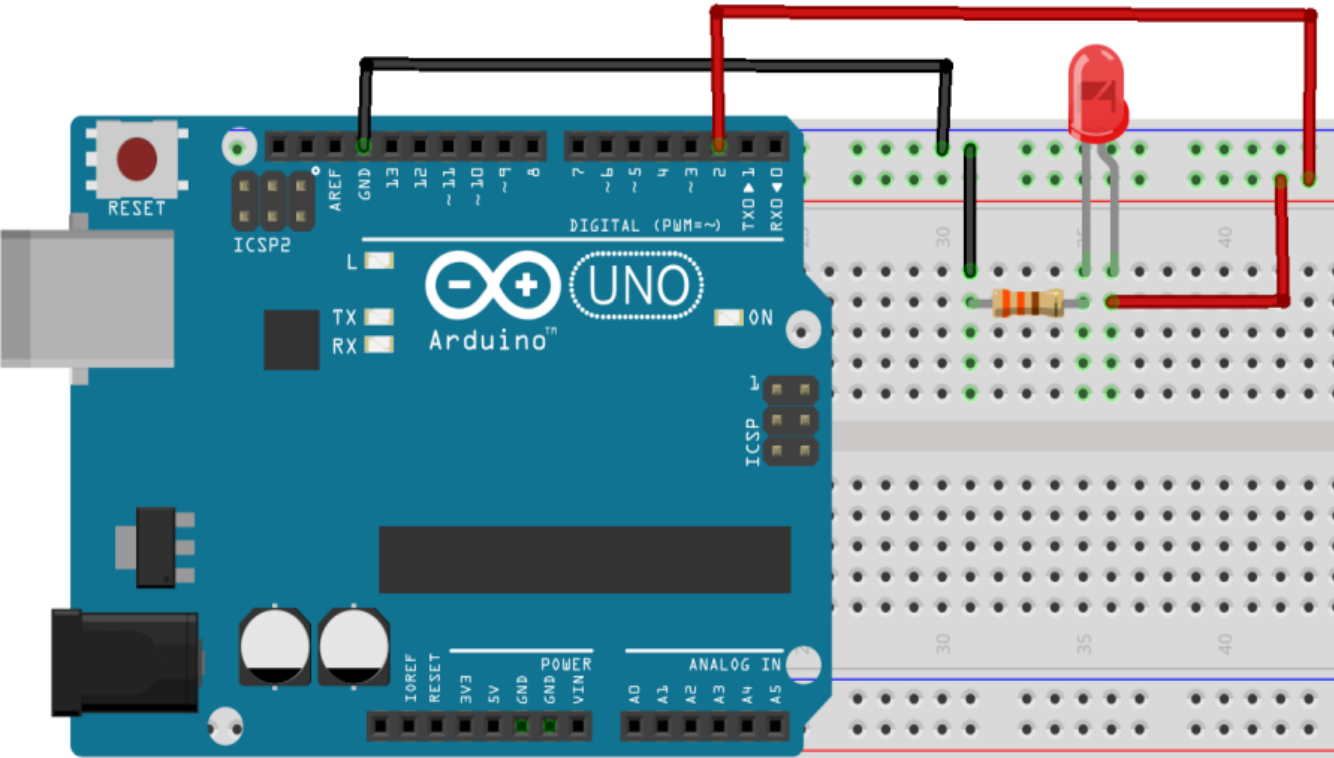
Arduino'da Kullanılan Genel Komutlar

- **pinMode ();**
 - Pinlerin Giriş mi Çıkış mı olacağına karar veriyoruz.
 - Çıkış(OUTPUT) olarak kullanılanlar → LED, Motor, LCD Ekran
 - Giriş(INPUT) olarak kullanılanlar → Potansiyometre, Buton
- **Kullanımı →**
 - pinMode(Pin numarası ya da değişken adı, OUTPUT ya da INPUT);
 - pinMode(motor, OUTPUT);
 - pinMode(11, INPUT);

Arduino'da Kullanılan Genel Komutlar

- **Serial.print();**
 - Bu komut ile sensörlerden gelen verileri ekrana yazdırabiliriz.
- **Kullanımı** → `Serial.print(değişken);`
 - `Serial.print("Merhaba Dünya");`
 - `Serial.print(potansiyometre);` → Potansiyometre verilerini ekrana yanyana yazdırıyor.
 - `Serial.println(potansiyometre);` → Potansiyometre verilerini ekrana altalta yazdırıyor.

Arduino'da LED uygulaması



```
int led=2;
```

```
void setup()
```

```
{  
  pinMode(led,OUTPUT);  
}
```

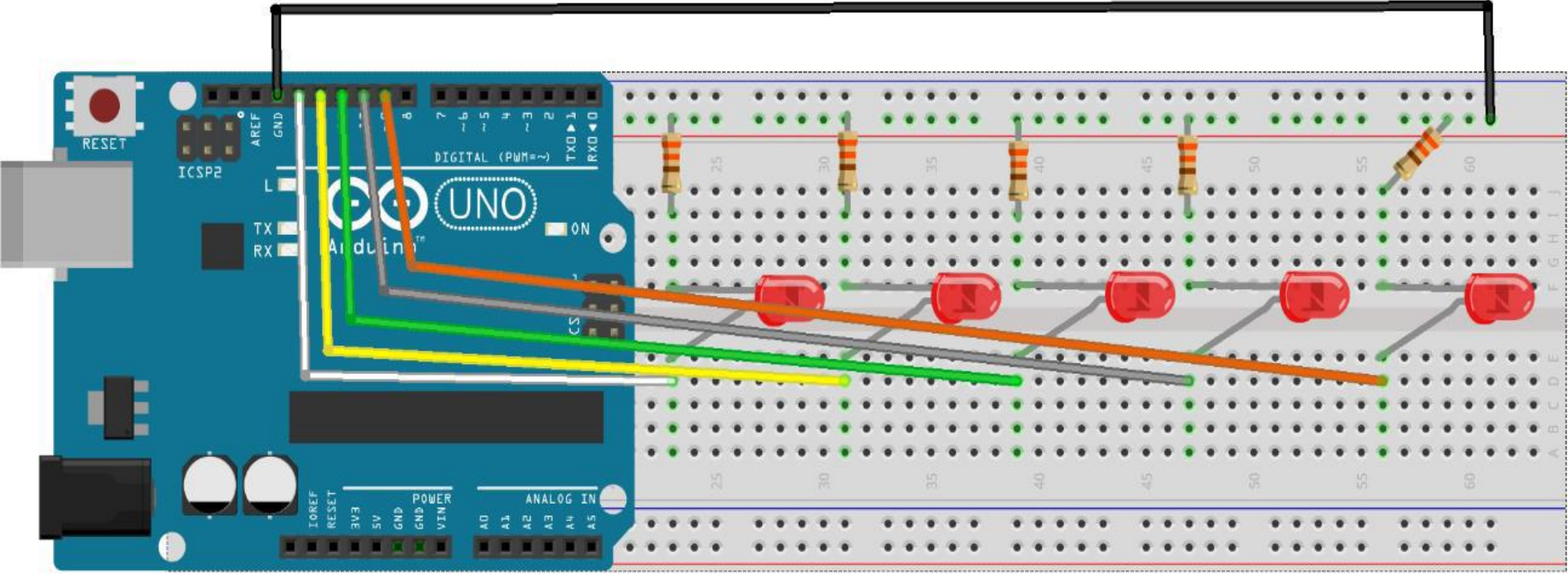
```
void loop()
```

```
{  
  digitalWrite(led, HIGH);  
  delay(1000);  
  digitalWrite(led, LOW);  
  delay(1000);  
}
```

Malzemeler

- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Karaşimşek Uygulaması



Malzemeler

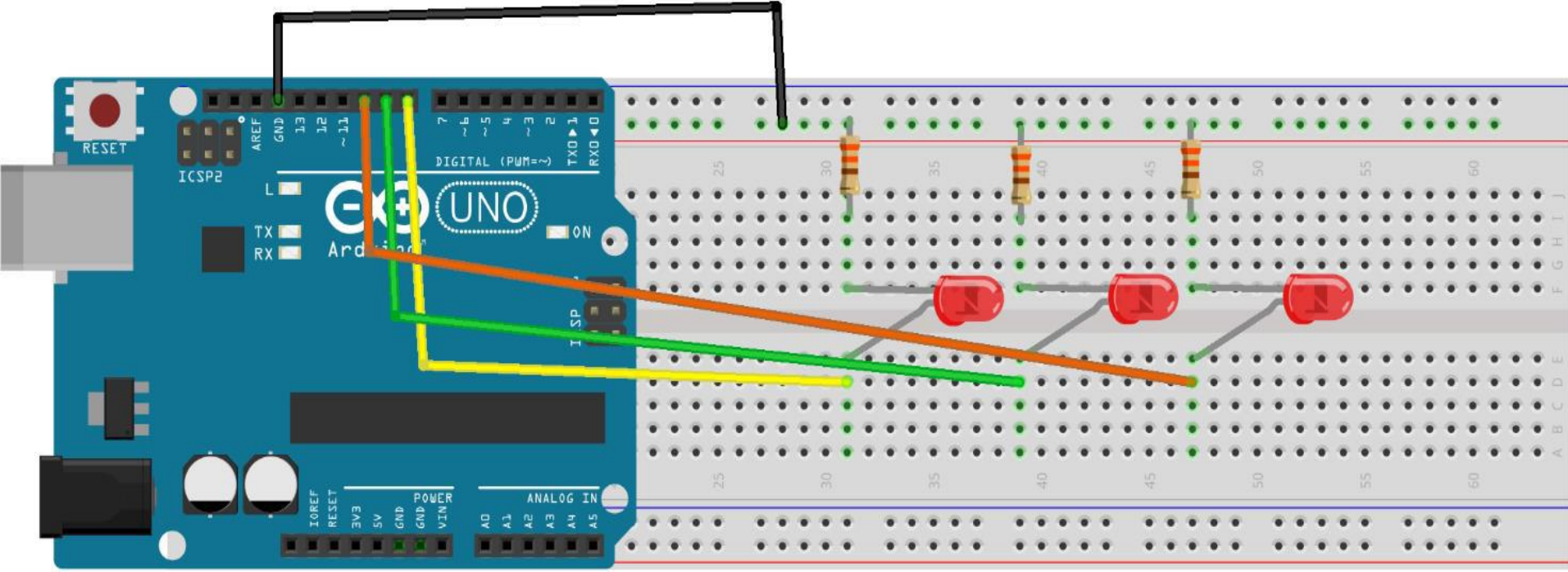
- 5 Led
- 5 Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Karaşimşek Uygulaması

```
• void setup()  
• {  
• pinMode(9,OUTPUT);  
• pinMode(10,OUTPUT);  
• pinMode(11,OUTPUT);  
• pinMode(12,OUTPUT);  
• pinMode(13,OUTPUT);  
• }
```

```
void loop()  
{  
digitalWrite(9, HIGH);  
delay(100);  
digitalWrite(9, LOW);  
digitalWrite(10, HIGH);  
delay(100);  
digitalWrite(10, LOW);  
digitalWrite(11, HIGH);  
delay(100);  
digitalWrite(11, LOW);  
digitalWrite(12, HIGH);  
delay(100);  
digitalWrite(12, LOW);  
digitalWrite(13, HIGH);  
delay(100);  
digitalWrite(13, LOW);  
}
```


Arduino'da Trafik Lambası Uygulaması



Malzemeler

- Sarı-Kırmızı-Yeşil Led
- 3 Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Trafik Lambası Uygulaması

```
int kirmizi=13;
int sari=12;
int yesil=11;

void setup()
{
pinMode(kirmizi,OUTPUT);
pinMode(sari,OUTPUT);
pinMode(yesil,OUTPUT);
}

void loop()
{
digitalWrite(kirmizi, HIGH);
delay(3000);

digitalWrite(yesil, LOW);
delay(200);

digitalWrite(yesil, HIGH);
delay(500);

digitalWrite(yesil, LOW);
delay(100);

digitalWrite(sari, HIGH);
delay(2000);
digitalWrite(sari, LOW);
delay(50);
}
```


Programlama Komutları

For Komutu

for (değişken tanımlama ve eşitleme; koşul;döngü)

Kullanımı →

```
for (x=0;x<100;x=x+2)
{
Serial.println(x); //100'den küçük çift sayılar ekrana yazılacak
}
```

For'un sağladığı kolaylıklar

```
digitalWrite(yesil, LOW);  
delay(200);
```

```
digitalWrite(yesil, HIGH);  
delay(500);
```

```
digitalWrite(yesil, LOW);  
delay(200);
```

```
digitalWrite(yesil, HIGH);  
delay(500);
```

```
digitalWrite(yesil, LOW);  
delay(200);
```

```
digitalWrite(yesil, HIGH);  
delay(500);
```

```
for(i=0; i<=3; i++)  
{  
  digitalWrite(yesil, LOW);  
  delay(200);  
  
  digitalWrite(yesil, HIGH);  
  delay(500);  
}
```

Arduino'da Trafik Lambası Uygulaması (For ile)

```
int kirmizi=13;
int sari=12;
int yesil=11;
int i;

void setup()
{
  pinMode(kirmizi,OUTPUT);
  pinMode(sari,OUTPUT);
  pinMode(yesil,OUTPUT);
}

void loop()
{
  digitalWrite(kirmizi, HIGH);
  delay(3000);

  digitalWrite(sari, HIGH);
  delay(2000);

  digitalWrite(kirmizi, LOW);
  digitalWrite(sari, LOW);
  delay(50);

  digitalWrite(yesil, HIGH);
  delay(3000);

  for(i=0; i<=3; i++)
  {
    digitalWrite(yesil, LOW);
    delay(200);

    digitalWrite(yesil, HIGH);
    delay(500);
  }

  digitalWrite(yesil, LOW);
  delay(100);

  digitalWrite(sari, HIGH);
  delay(2000);
  digitalWrite(sari, LOW);
  delay(50);
}
```


Programlama Komutları

- if- else komutu

– if → Eğer else → Değilse

```
if(değişken == deger,)
```

```
{
```

```
}
```

```
Else
```

```
{
```

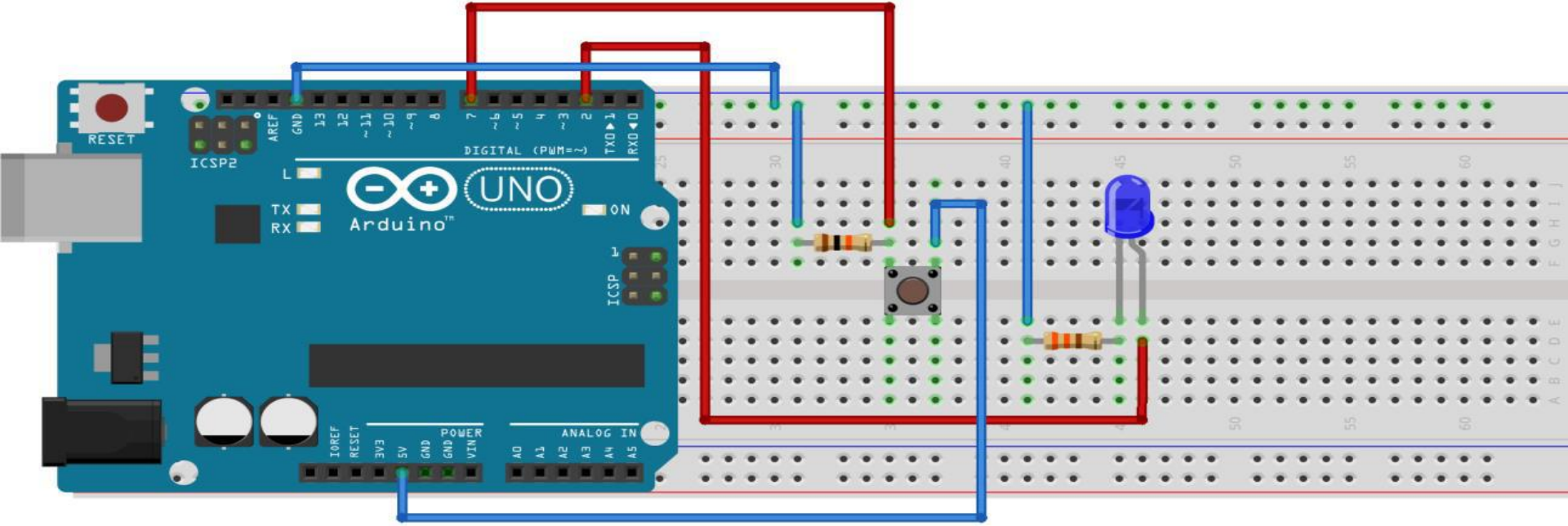
```
}
```

Programlama Komutları

Kullanımı →

```
if(buton == 1)
{
Serial.print("Butona Basıldı");
}
else
{
Serial.print("Butona Basılmadı !");
}
```

Arduino'da Buton Uygulaması



Malzemeler

- Buton
- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Buton Uygulaması

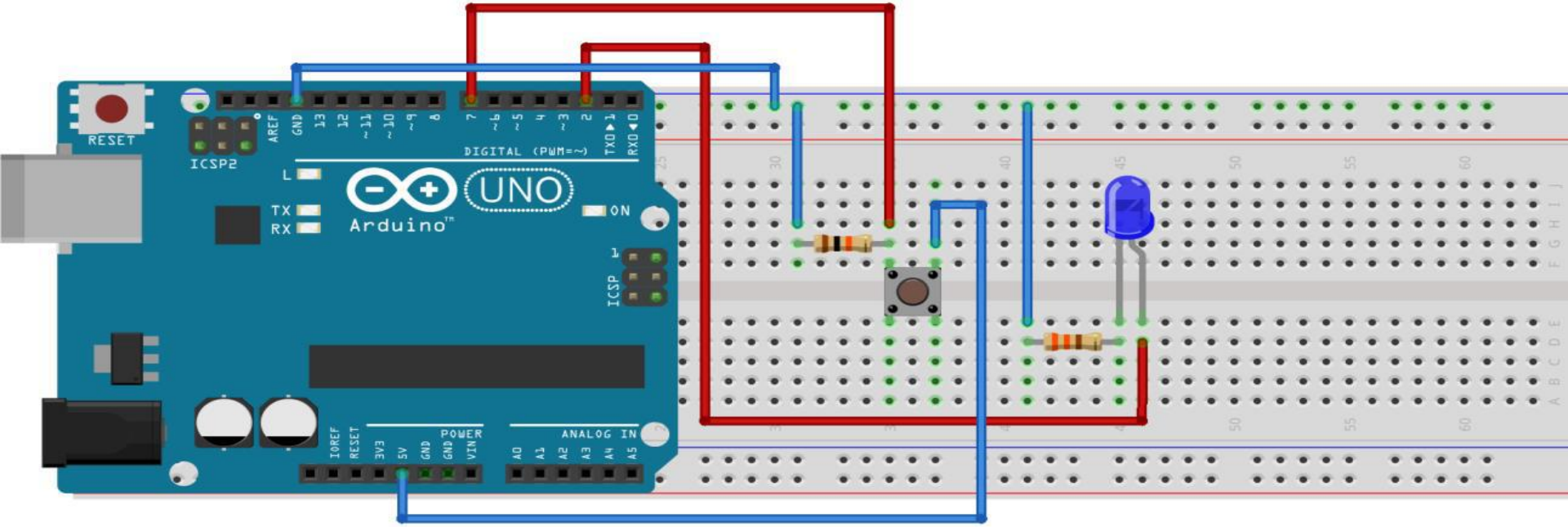
```
int buton=7;
int led=2;
int butondurum=0;

void setup()
{
  pinMode(buton, INPUT);
  pinMode(led, OUTPUT);
}

void loop()
{
  butondurum= digitalRead(buton);

  if(butondurum ==1)
  {
    digitalWrite(led, HIGH);
  }
  else
  {
    digitalWrite(led, LOW);
  }
}
```


Arduino'da Serial Monitör Kullanımı



Malzemeler

- Buton
- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Buton Uygulaması

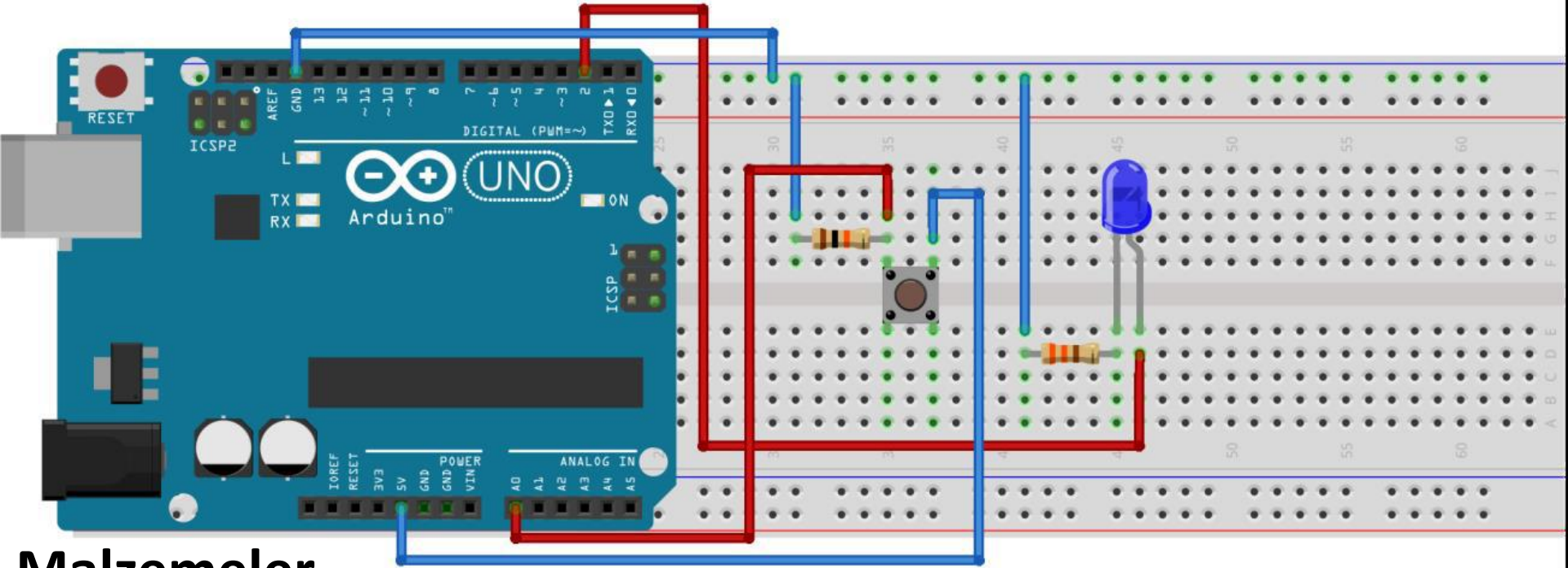
```
int buton=7;
int led=2;
int butondurum=0;

void setup()
{
  Serial.begin(9600);
  pinMode(buton, INPUT);
  pinMode(led, OUTPUT);
}

void loop()
{
  butondurum= digitalRead(buton);

  if(butondurum ==1)
  {
    digitalWrite(led, HIGH);
    Serial.println("Led Yanıyor");
  }
  else
  {
    digitalWrite(led, LOW);
    Serial.println("Led Yanmıyor");
  }
}
```

Arduino'da Analog Veri Okuma



Malzemeler

- Buton
- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino'da Analog Veri Okuma

```
int buton=A1;
int led=2;
int butondurum=0;

void setup()
{
  Serial.begin(9600);
  pinMode(buton, INPUT);
  pinMode(led, OUTPUT);
}

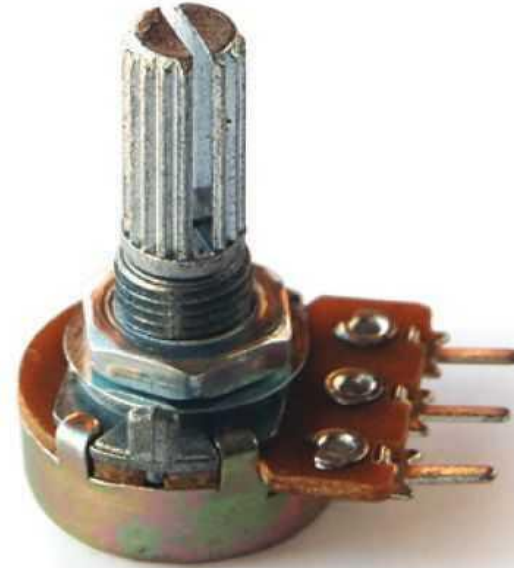
void loop()
{
  butondurum= analogRead(buton);

  if(butondurum ==1023)
  {
    digitalWrite(led, HIGH);
    Serial.println(butondurum);
  }
  else
  {
    digitalWrite(led, LOW);
    Serial.println(butondurum);
  }
}
```

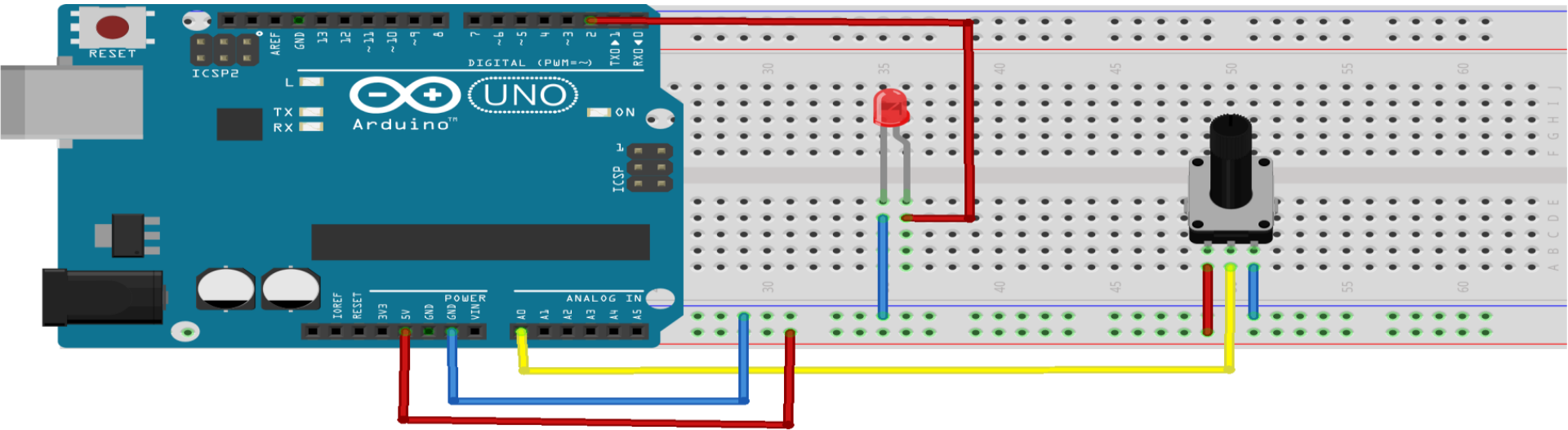
Arduino'da Potansiyometre Kullanımı

- Potansiyometre bir direnç türüdür. Direnç gruplarından ayıran en büyük özelliği ayarlanabilir olmasıdır. Ocaklarda, ses sistemlerinde gibi ayar çubuğu olan yerlerde kullanımı kolaylaştırmaktadır. Bacak Dizilimi → Sol bacak Artı, orta bacak Sinyal, sağ bacak GND.

Analog girişe ve çıkışa dikkat



Arduino'da Potansiyometre Kullanımı



Malzemeler

- Potansiyometre
- Led
- Jumper Kablolar
- Breadboard

Arduino'da Potansiyometre Kullanımı

```
int pot=A0;
int led=3;

void setup()
{
  Serial.begin(9600);
  pinMode(pot, INPUT);
  pinMode(led, OUTPUT);
}

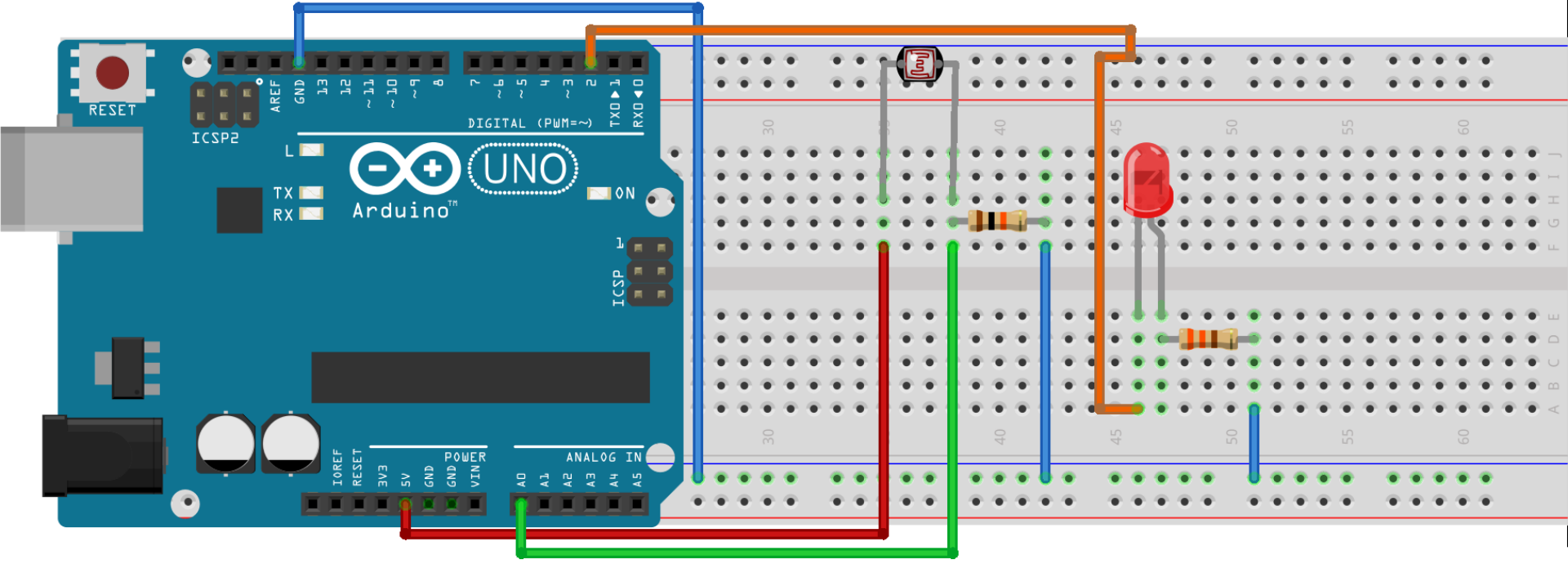
void loop()
{
  int potdeger= analogRead(pot);
  //potdeger=map(potdeger,0,1023,
  0,255);
  analogWrite(led, potdeger);
  delay(10);
  Serial.println(potdeger);
}
```


Arduino'da LDR Kullanımı

- Ortamdaki ışığın şiddetine göre üzerindeki direnç değerini ışık ile ters orantılı olarak ayarlayabilen sensördür.



Arduino'da LDR Kullanımı



Malzemeler

- LDR
- Led
- Jumper Kablolar
- Breadboard

Arduino'da LDR Kullanımı

```
int LDR = A0;
```

```
int deger = 0;
```

```
int led=2;
```

```
void setup()
```

```
{
```

```
  pinMode(led, OUTPUT);
```

```
  Serial.begin(9600);
```

```
}
```

```
//pinMode(LDR, INPUT);  
  Kullanmıyoruz çünkü A  
  pinleri giriş için  
  tasarlanmıştır.
```

```
void loop()
```

```
{
```

```
  deger = analogRead(LDR);
```

```
  Serial.println(deger);
```

```
  if(deger < 150)
```

```
  {
```

```
    digitalWrite(led,HIGH);
```

```
    delay(100);
```

```
  }
```

```
  else
```

```
  {
```

```
    digitalWrite(led,LOW);
```

```
  }
```

```
  delay(100);
```

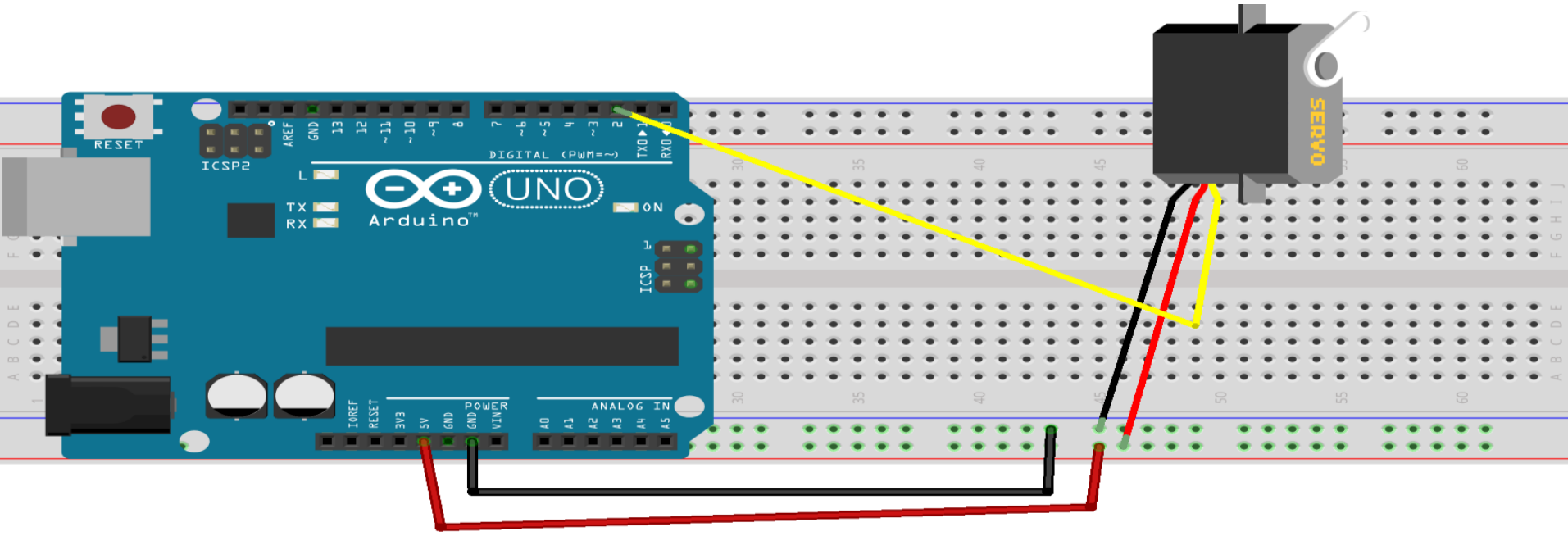
```
}
```


Arduino'da Servo Motor Kullanımı

- 0 ile 180 derece açılarında çalışabilen. Yönünü ve hızını değiştirebileceğimiz elektronik devre ürünüdür. Robotlarda, mekanik kollar vb. kullanımı oldukça yaygındır.



Arduino'da Servo Motor Kullanımı



Malzemeler

- Servo Motor
- Jumper Kablolar
- Breadboard

Arduino'da Servo Motor Kullanımı

```
#include <Servo.h>
```

```
Servo servom;
```

```
int pos = 0;
```

```
void setup()
```

```
{  
  servom.attach(9);
```

```
}
```

```
void loop()
```

```
{
```

```
  for (pos = 0; pos <= 180; pos += 1)
```

```
  {
```

```
    servom.write(pos);
```

```
    delay(15);
```

```
  }
```

```
  for (pos = 180; pos >= 0; pos -= 1)
```

```
  {
```

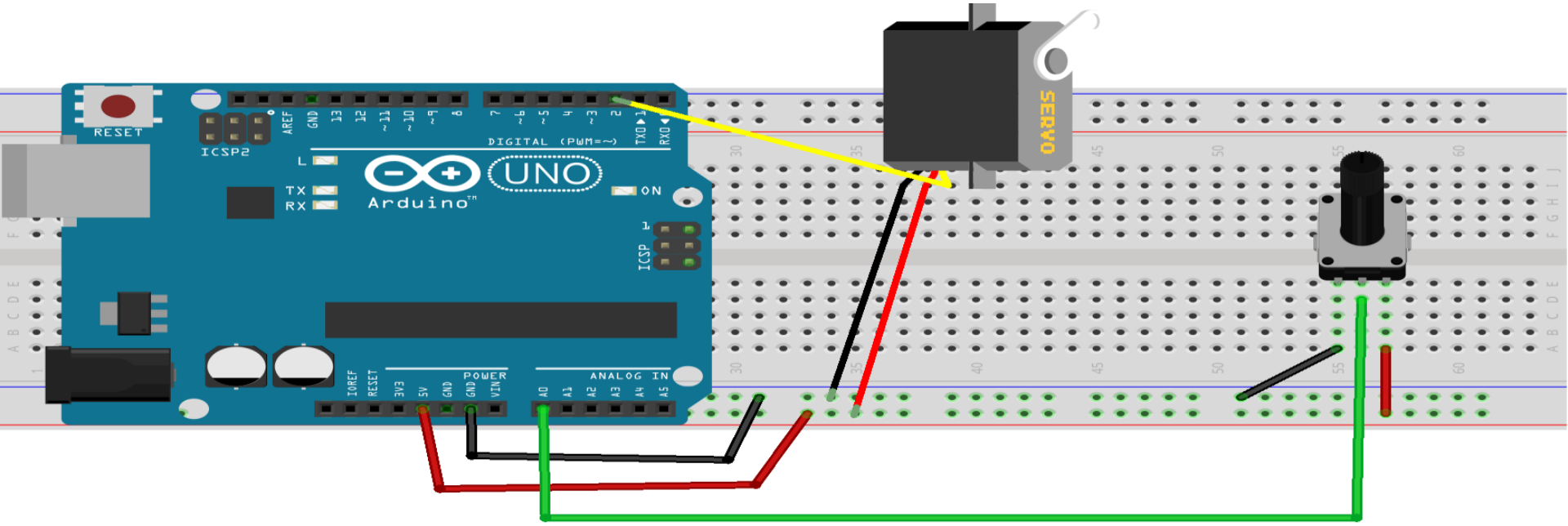
```
    servom.write(pos);
```

```
    delay(15);
```

```
  }
```

```
}
```


Potansiyometre ile Servo Motor Kontrolü



Malzemeler

- Potansiyometre
- Servo Motor
- Jumper Kablolar
- Breadboard

Potansiyometre ile Servo Motor Kontrolü

```
#include <Servo.h>

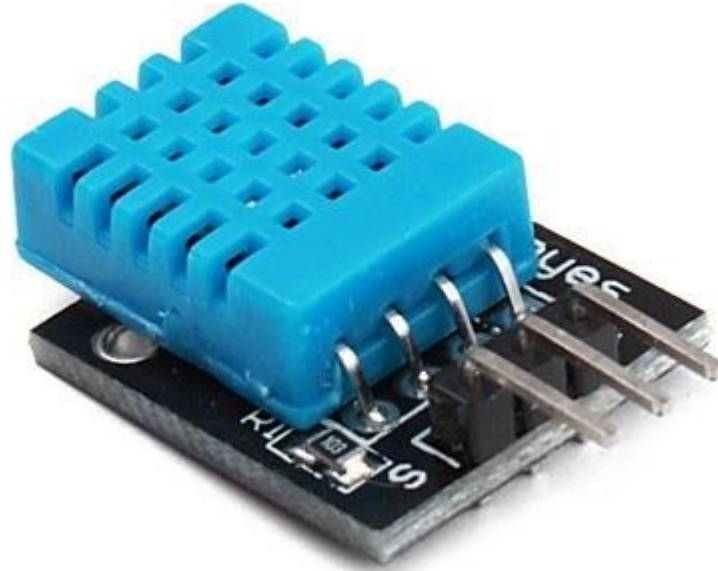
Servo servom;
const byte pot=A0;
int pot_deger;
int ydeger;

void loop()
{
  servom.write(0);
  pot_deger=analogRead(pot);
  ydeger=map(pot_deger,0,1023,0,180);
  servom.write(ydeger);
  delay(50);
}

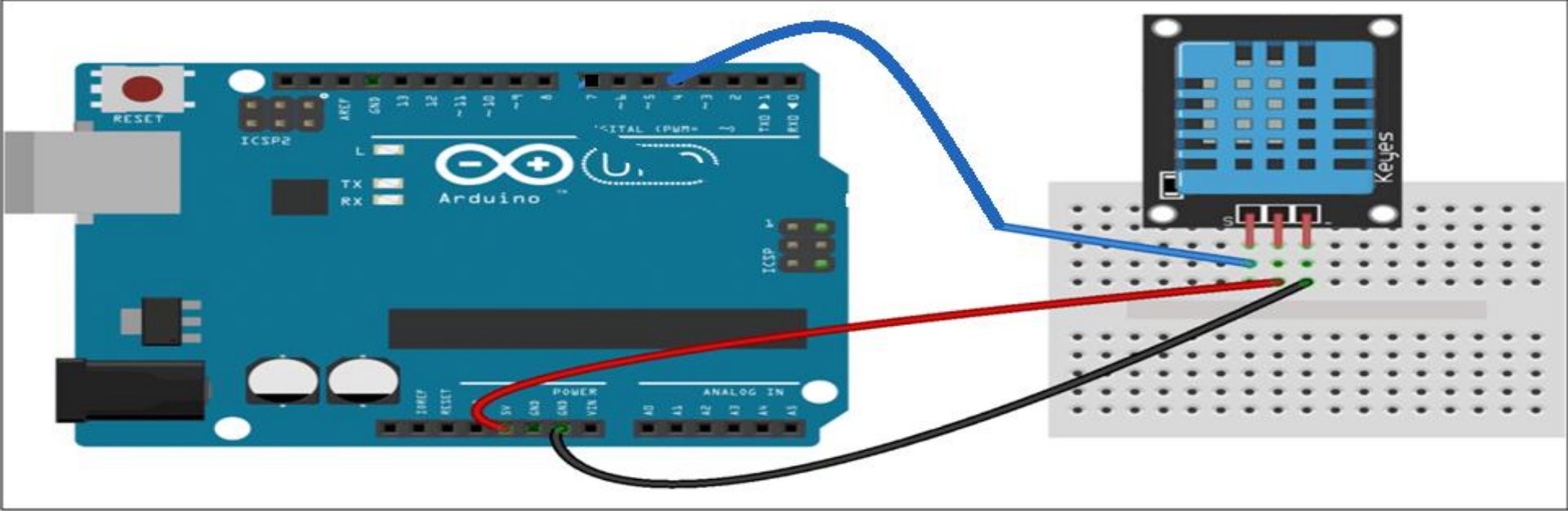
void setup()
{
  servom.attach(9);
}
```


Arduino ile Sıcaklık Ölçümü

- DHT-11 Sıcaklık ve Nem ölçmeye yarayan sensördür.



Arduino ile Sıcaklık Ölçümü



Malzemeler

- DHT-11
- 10 K ohm direnç
- Jumper Kablolar
- Breadboard

Kütüphane yok ise eklenmelidir.

Arduino ile Sıcaklık Ölçümü

```
#include <dht.h>
```

```
dht DHT;
```

```
#define DHT11_PIN 4
```

```
void setup()
```

```
{  
  Serial.begin(9600);  
  Serial.println("DHT TEST PROGRAMI ");  
  Serial.println();  
  Serial.println("Sensor, \Durum, \Nem (%),  
  \Sicaklik (C)");  
}
```

```
void loop()
```

```
{
```

```
  Serial.print("DHT11, \t");  
  int sinyal = DHT.read11(DHT11_PIN);
```

```
  switch (sinyal)
```

```
  {
```

```
    case DHTLIB_OK:  
      Serial.print(" Calisiyor,\t");  
      break;
```

```
    case DHTLIB_ERROR_CONNECT:  
      Serial.print("Baglanti Hatasi,\t");  
      break;
```

```
  }
```

```
  Serial.print(DHT.humidity,1);  
  Serial.print(", ");  
  Serial.println(DHT.temperature,1);
```

```
  delay(1000);
```

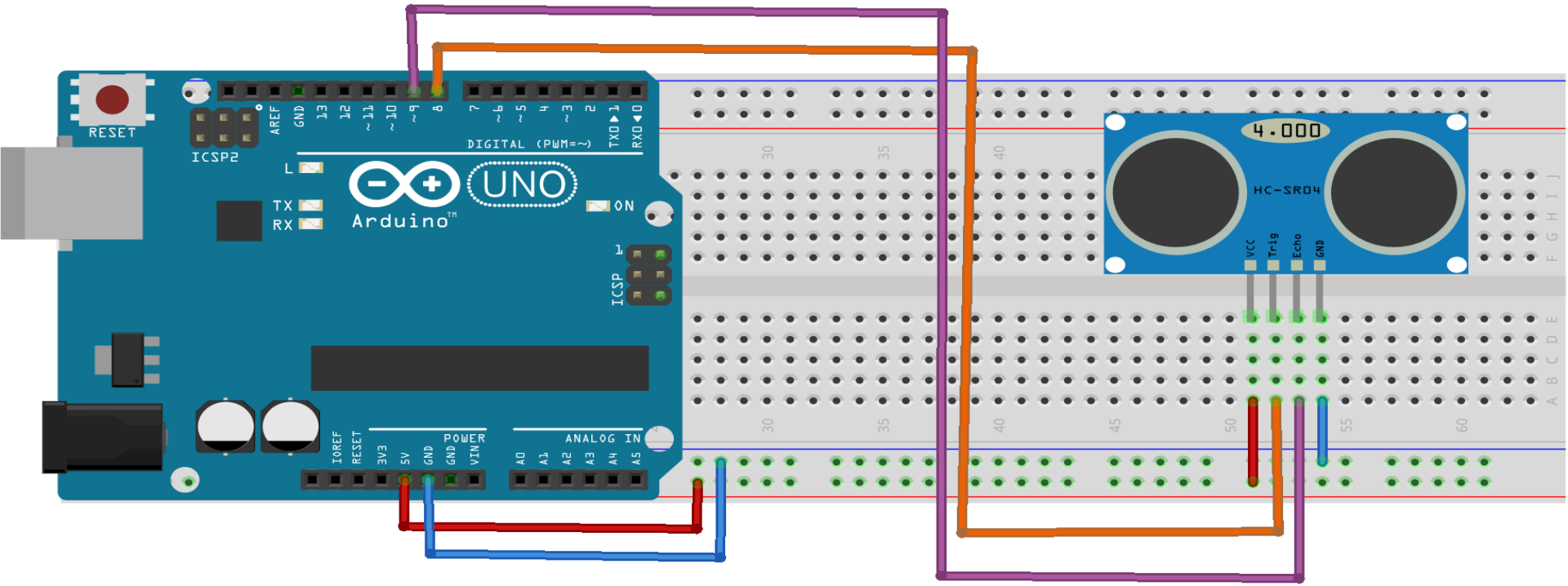
```
}
```


Arduino ile Mesafe Sensörü Kullanımı

Bu modülde Vcc, Gnd, Echo ve Trigger pinleri vardır. Trigger pini dalga gönderir, Echo pini ise dalga okur. Bu şekilde mesafe bilgisini görürüz.



Arduino ile Mesafe Sensörü Kullanımı



Malzemeler

- HC-SR04 Mesafe Sensörü
- Jumper Kablolar
- Breadboard

Arduino ile Mesafe Sensörü Kullanımı

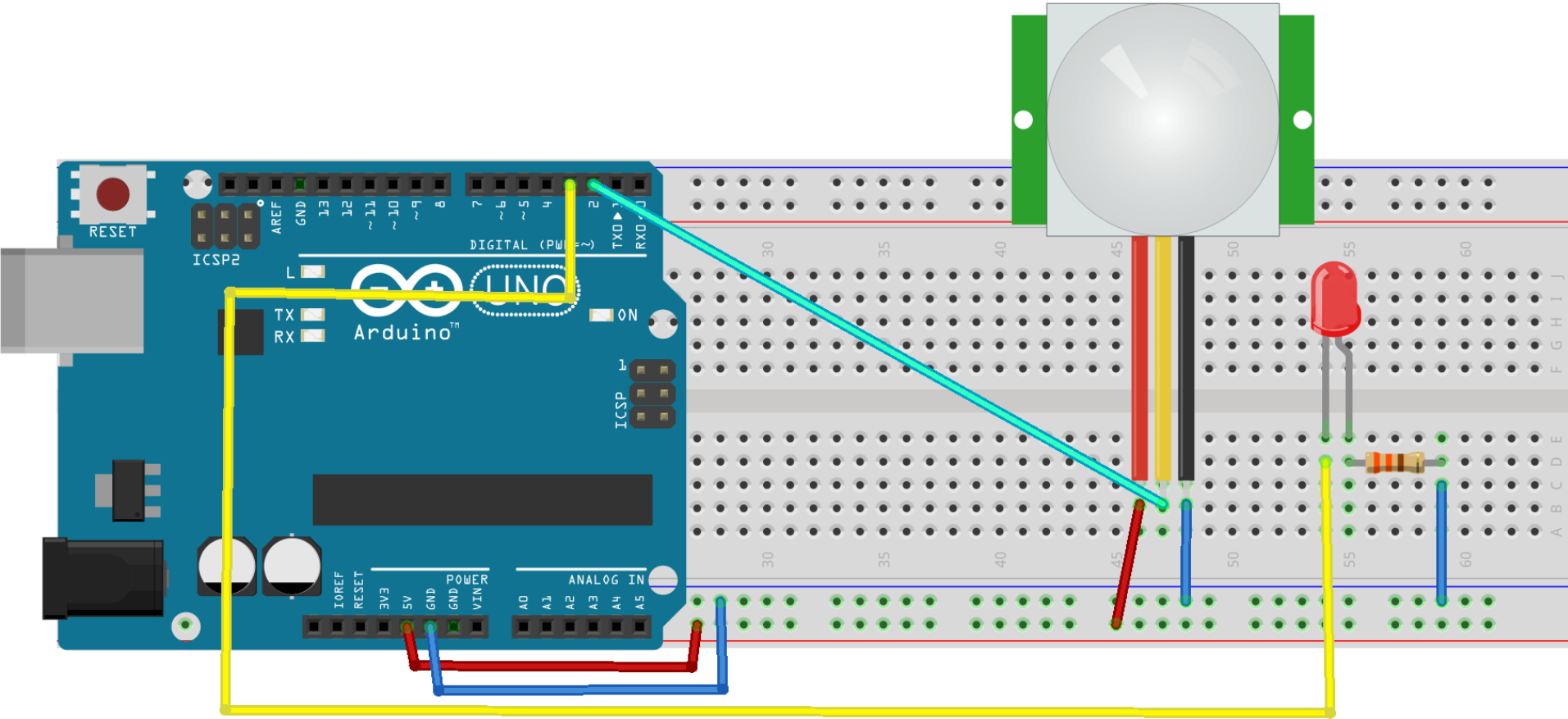
```
const int t_pin=8;
const int e_pin=9;

float sure;
float mesafe;

void setup()
{
pinMode(t_pin, OUTPUT);
pinMode(e_pin, INPUT);
Serial.begin(9600);
}

void loop()
{
digitalWrite(t_pin,HIGH);
delay(1000);
digitalWrite(t_pin,LOW);
sure = pulseIn(e_pin,HIGH);
mesafe=(sure/2)/29.1;
Serial.print("Cisme olan uzaklık ==> ");
Serial.println(mesafe,2);
Serial.println("-----");
delay(100);
}
```


Arduino ile Hareket Sensörü Kullanımı



Malzemeler

- HC-SR501 PIR Sensörü
- Led
- 220 Ohm Direnç
- Jumper Kablolar
- Breadboard

MAKERSOFT

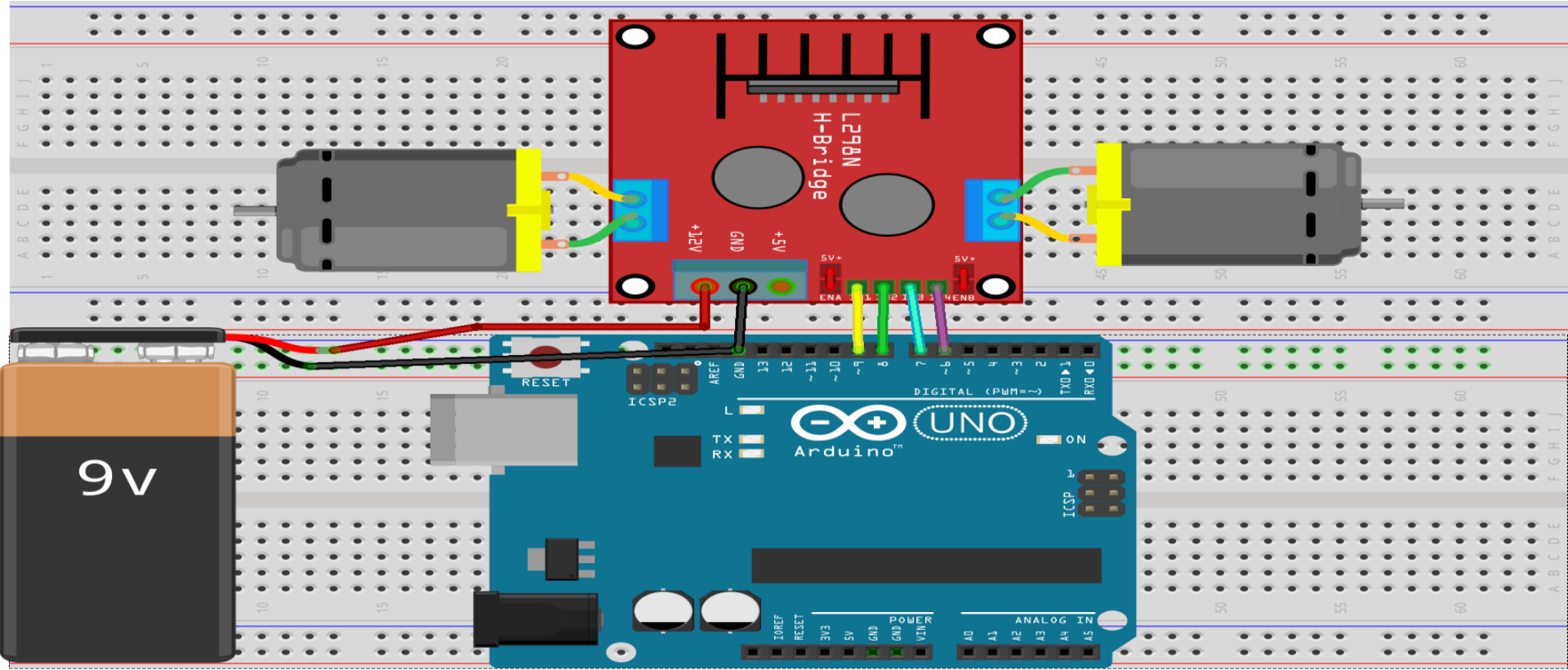
Arduino ile Hareket Sensörü Kullanımı

```
int pirPin=2;  
int ledPin=3;  
int deger=0;
```

```
void setup()  
{  
  pinMode(pirPin,INPUT);  
  pinMode(ledPin, OUTPUT);  
  Serial.begin(9600);  
}
```

```
void loop()  
{  
  deger=digitalRead(pirPin);  
  Serial.println(deger);  
  
  if(deger==HIGH)  
  {  
    digitalWrite(ledPin,HIGH);  
  }  
  else  
  {  
    digitalWrite(ledPin,LOW);  
  }  
}
```


Arduino ile L298N Motor Sürücü Kullanımı



Malzemeler

- L298N motor sürücü
- DC motor
- Jumper Kablolar
- Breadboard

Arduino ile L298N Motor Sürücü Kullanımı

```
int sol_ileri=9;
int sol_geri=8;
int sag_ileri=7;
int sag_geri=6;

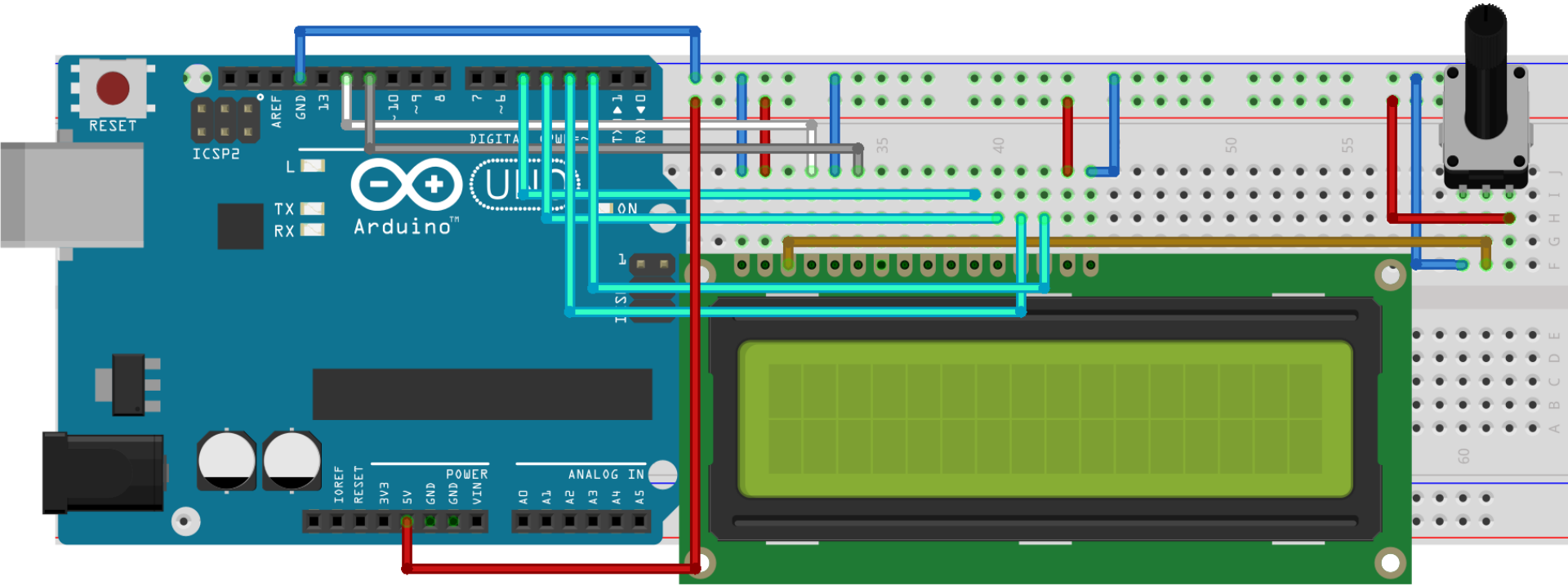
void setup()
{
pinMode(sol_ileri,OUTPUT);
pinMode(sol_geri,OUTPUT);
pinMode(sag_ileri,OUTPUT);
pinMode(sag_geri,OUTPUT);
}
```

```
void loop()
{
digitalWrite(sol_ileri, HIGH);
digitalWrite(sol_geri, LOW);
digitalWrite(sag_ileri, HIGH);
digitalWrite(sag_geri, LOW);
delay(1000);
digitalWrite(sol_ileri, LOW);
digitalWrite(sol_geri, HIGH);
digitalWrite(sag_ileri, LOW);
digitalWrite(sag_geri, HIGH);
delay(1000);
}
```


Arduino ile LCD Kullanımı

- `lcd.begin(sütunsayısı,satır sayısı); // LCD ekranın boyutunu belirlememizi sağlar.`
- `lcd.print("CANER YILDIZ"); // Ekrana yazdıracağımız verileri gösterir.`
- `lcd.setCursor(sütünsayısı,satır sayısı);// Yazının İlk yerini Ayarlamak İçin Kullanılır.`
- `lcd.clear(); // Ekranı Temizler.`
- `millis();//Arduino'nun Çalışmaya Başladığı Süreyi Ele Alır.`

Arduino ile LCD Kullanımı



Malzemeler

- 2x16 LCD Ekran
- Potansiyometre
- Jumper Kablolar
- Breadboard

Kütüphane yok ise eklenmelidir.

Arduino ile LCD Kullanımı

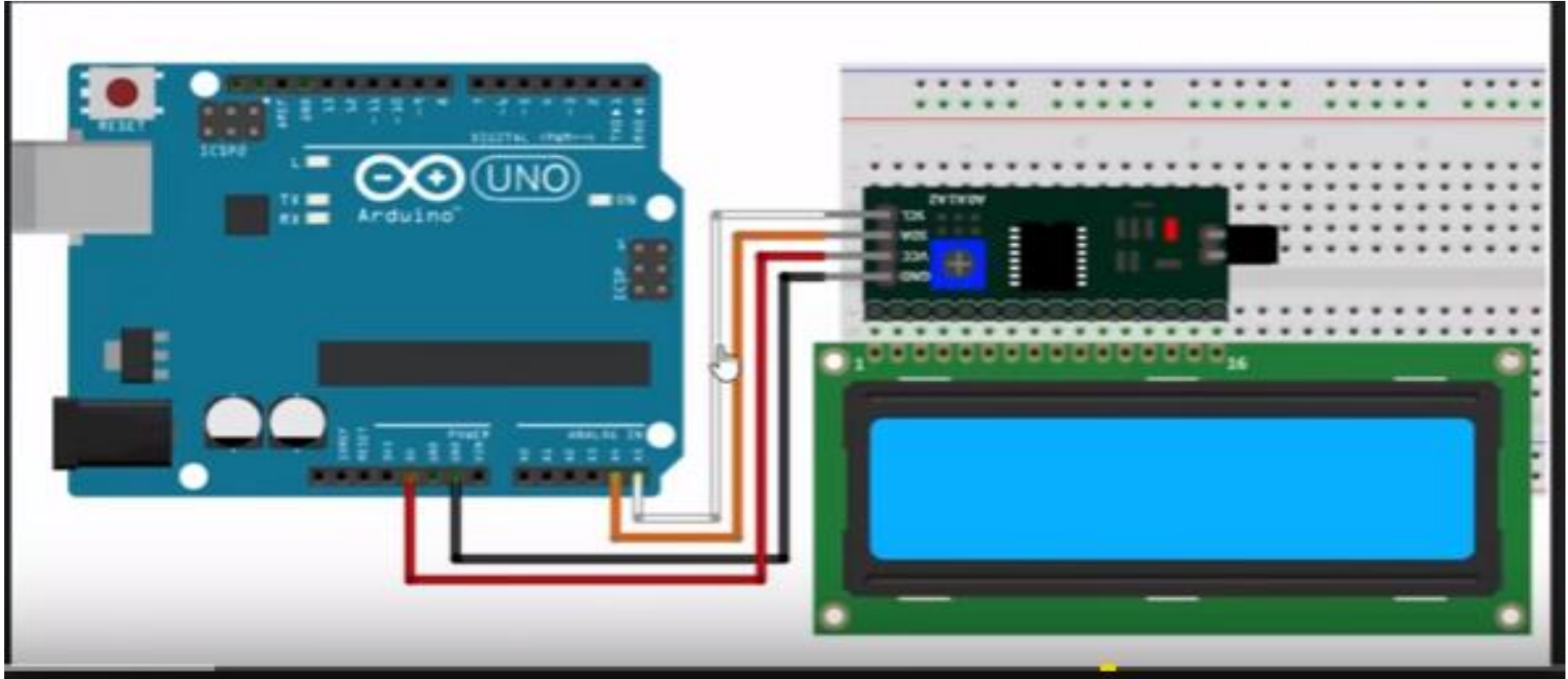
```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12,11,5,4,3,2);

void setup()
{
  lcd.begin(16,2);
  lcd.print("Caner   YILDIZ");
}

void loop()
{
  lcd.setCursor(0,1);
  lcd.print(millis()/1000);
}
```


Arduino ile LCD Kullanımı I2C' li



Malzemeler

- 2x16 LCD Ekran
- I2C Modülü
- Jumper Kablolar
- Breadboard

Kütüphane yok ise eklenmelidir.

Arduino ile LCD Kullanımı I2C' li

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

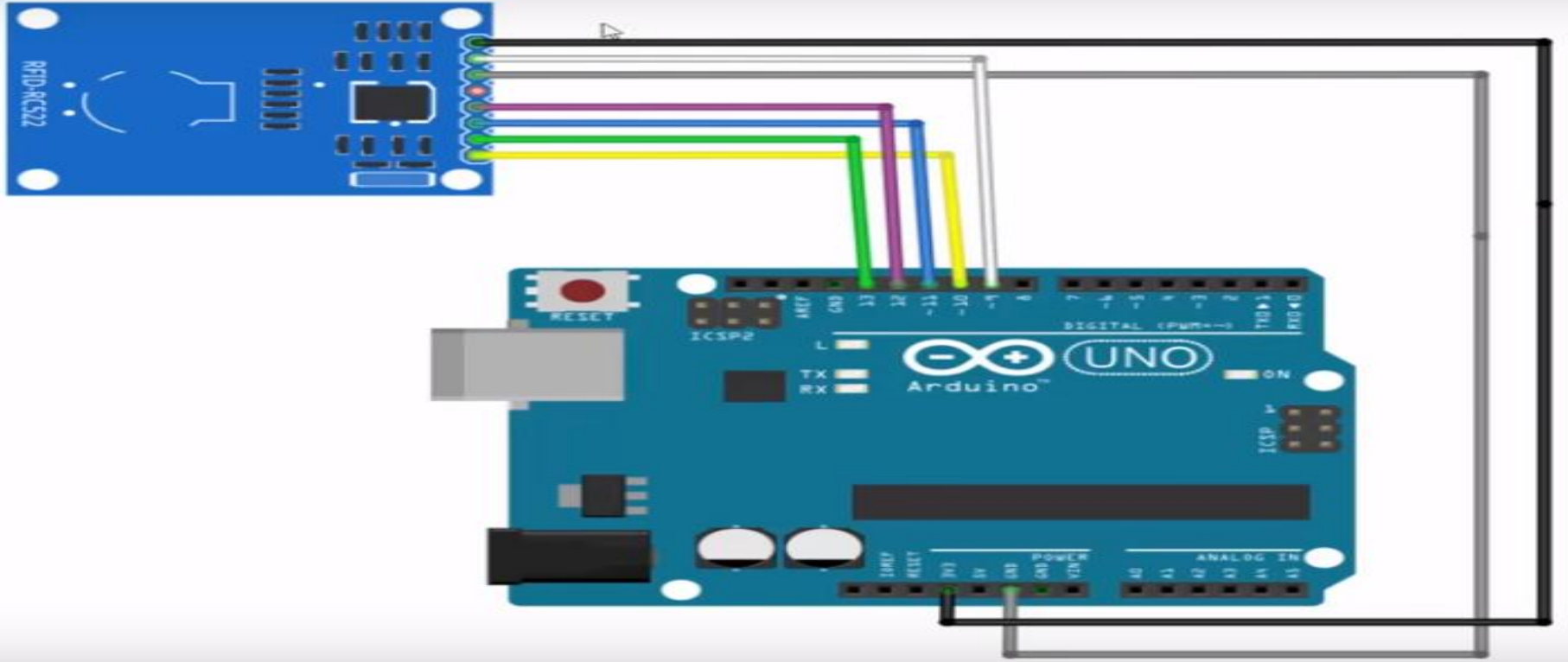
void setup()
{
  lcd.begin();
  lcd.clear();
}

void loop()
{
  //lcd.home();
  lcd.setCursor(0,0);
  //lcd.print(155, BIN); //155 deęerinin 0 1
  // Binary deęeri

  lcd.print("Tusun");
  lcd.setCursor(0,1);
  lcd.print("Akademi");

  lcd.backlight(); // Arkaplan ışıklarını açar.
  delay(400);
  lcd.noBacklight(); // Arkaplan ışıklarını kapatır.
  delay(400);
}
```

RFID Kart Kullanımı



Malzemeler

- RFID Kart Okuyucu
- RFIF Kart ya da Anahtarlık
- Jumper Kablolar

RFID Kart Kullanımı

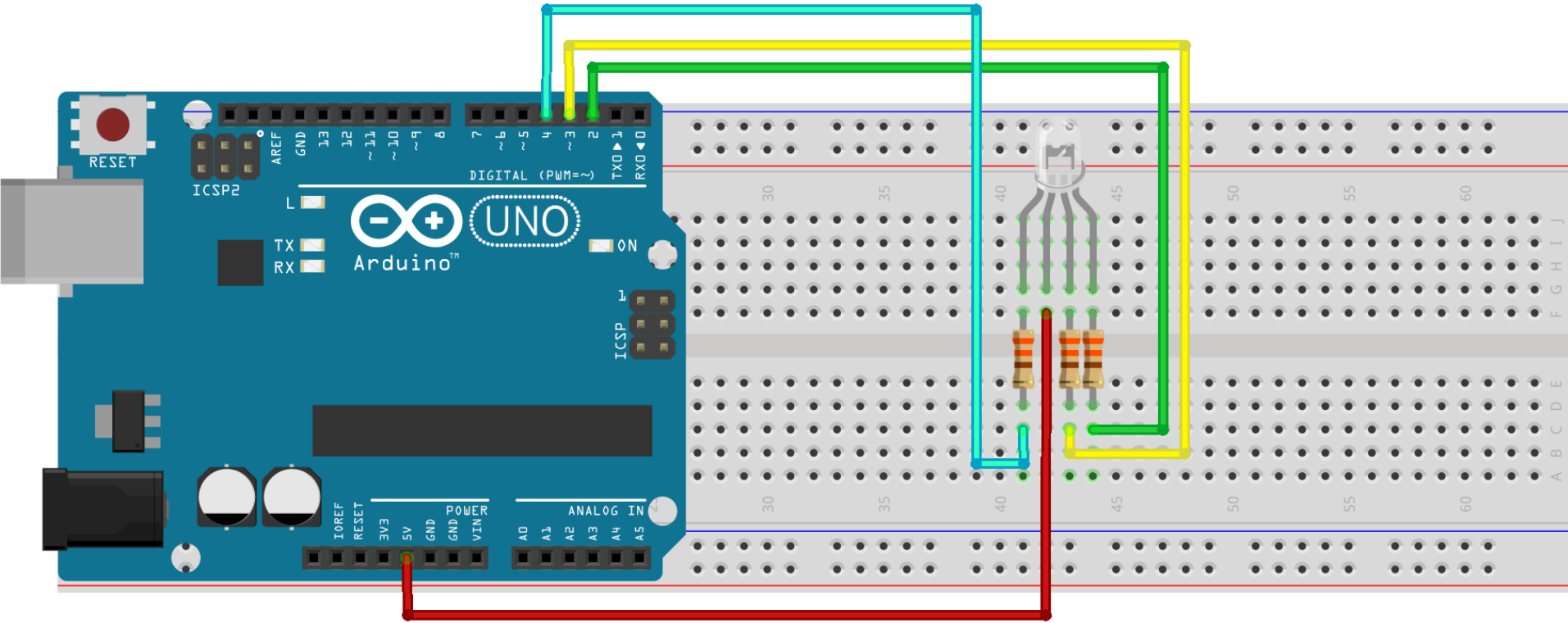
RFID Kart Kullanımı

Arduino ile RGB Led Kullanımı

- RGB LED Kırmızı(R), Yeşil(G), Mavi(B) renkleri temsil ediyor. Girilen değerlere göre bu LED tüm renklere ulaşmamızı mümkün kılmaktadır.

Ortak ANOT – Ortak KATOT

Arduino ile RGB Led Kullanımı



Malzemeler

- RGB Led
- Jumper Kablolar
- Breadboard

Arduino ile RGB Led Kullanımı

```
int m=5;
int k=4;
int y=3;

void setup()
{
pinMode(m,OUTPUT);
pinMode(k,OUTPUT);
pinMode(y,OUTPUT);
}

void loop()
{
digitalWrite(m,LOW);
digitalWrite(k,HIGH);
digitalWrite(y,HIGH);
}
```

```
int rpin = 11; // kırmızı
int gpin = 10; // yeşil
int bpin = 9; // mavi
float h = 0; //hue (ton) değeri
int r=0, g=0, b=0; // renk değerleri
```

1

```
void setup()
{
}

void loop()
{
// Her seferinde hue değeri 0.001 artırılıyor ve 1 olduğunda başa dönülüyor
h = h + 0.001;
if(h >= 1.0)
{
h = 0;
}
// hue değeri r,g,b bileşenlerine çevriliyor
h2rgb(h,r,g,b);
```

```
// Ortak anot RGB LED
analogWrite(rpin, 255 - r);
analogWrite(gpin, 255 - g);
analogWrite(bpin, 255 - b);
```

2

```
/* Ortak katot için aşağıdaki kodları kullanabilirsiniz
analogWrite(rpin, r);
analogWrite(gpin, g);
analogWrite(bpin, b);
*/

delay(20);
}

/* h2rgb
0 ile 1 arasında verilen renk tonu (hue) değerine göre
Kırmızı (R), yeşil(G) ve mavi (B) değerlerini verir.
Giriş: H => Hue değeri (0-1 arasında)
Çıkış: R, G, B (0-255 arasında)
*/
```

```
void h2rgb(float H, int& R, int& G, int& B) {
int var_i;
float S=1, V=1, var_1, var_2, var_3, var_h, var_r, var_g, var_b;
```

3

```
if ( S == 0 )
{
R = V * 255;
G = V * 255;
B = V * 255;
}
else
{
var_h = H * 6;
if ( var_h == 6 ) var_h = 0;
var_i = int( var_h );
var_1 = V * ( 1 - S );
var_2 = V * ( 1 - S * ( var_h - var_i ) );
var_3 = V * ( 1 - S * ( 1 - ( var_h - var_i ) ) );
```

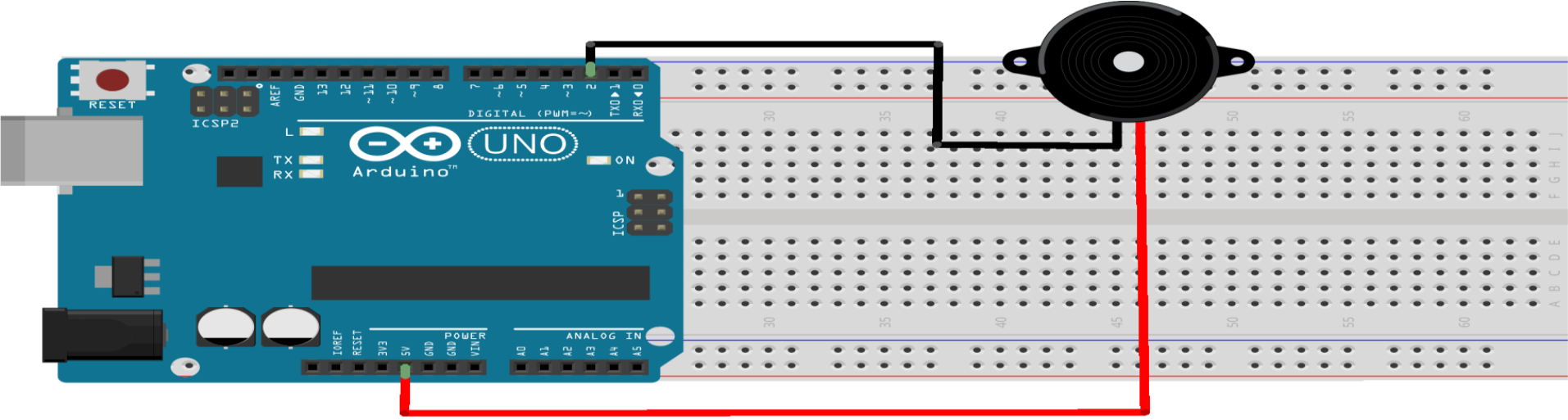
```
if ( var_i == 0 ) {
var_r = V ;
var_g = var_3 ;
var_b = var_1 ;
}
else if ( var_i == 1 ) {
var_r = var_2 ;
var_g = V;
var_b = var_1 ;
}
else if ( var_i == 2 ) {
var_r = var_1 ;
var_g = V;
var_b = var_3 ;
}
```

4

```
else if ( var_i == 3 ) {
var_r = var_1 ;
var_g = var_2 ;
var_b = V;
}
else if ( var_i == 4 ) {
var_r = var_3 ;
var_g = var_1 ;
var_b = V;
}
else {
var_r = V;
var_g = var_1 ;
var_b = var_2 ;
}
R = (1-var_r) * 255;
G = (1-var_g) * 255;
B = (1-var_b) * 255;
}
}
```

5

Arduino ile Buzzer Kullanımı



Malzemeler

- Buzzer
- Jumper Kablolar
- Breadboard

Arduino ile Buzzer Kullanımı

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

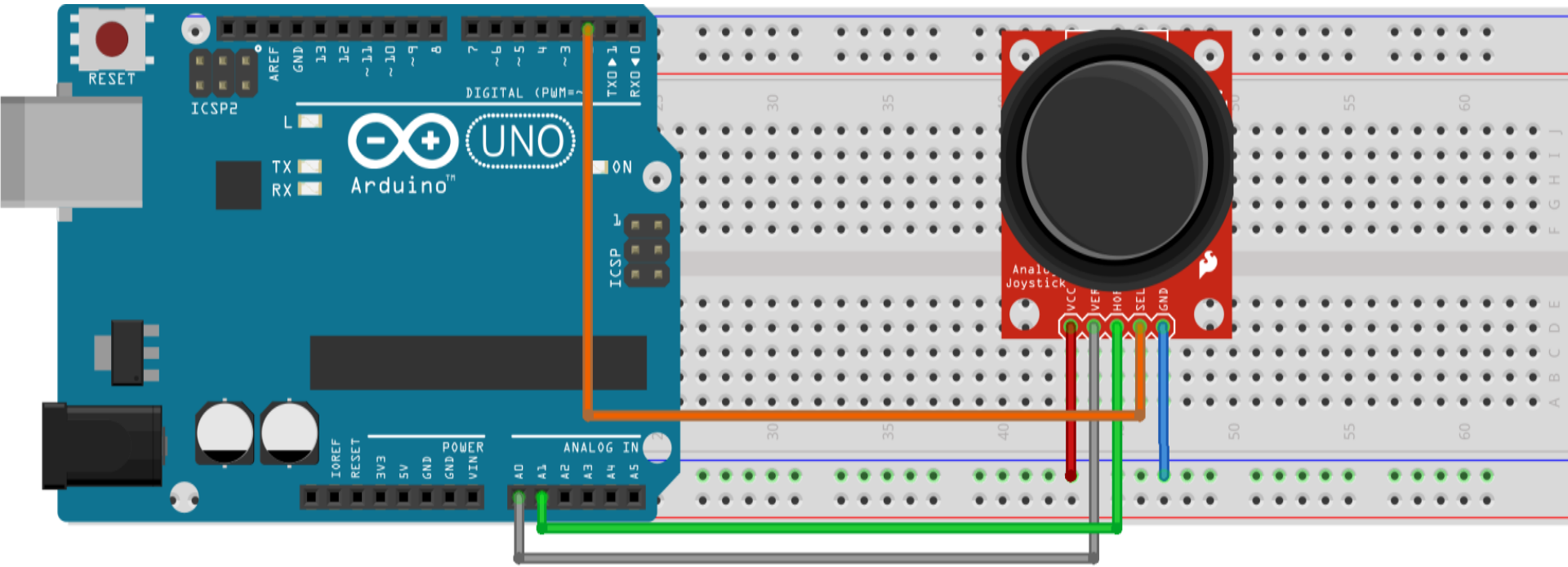
Arduino ile Buzzer Kullanımı

```
//int ledPin = 12;
int speakerOut = 2;
byte names[] = {'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C'};
int tones[] = {1915, 1700, 1519, 1432, 1275, 1136, 1014, 956};
byte melody[] = "2d2a1f2c2d2a2d2c2f2d2a2c2d2a1f2c2d2a2a2g2p8p8p";
// count length: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
//                10      20      30
int count = 0;
int count2 = 0;
int count3 = 0;
int MAX_COUNT = 24;
int statePin = LOW;

void setup() {
  pinMode(ledPin, OUTPUT);
}
```

```
void loop() {
  analogWrite(speakerOut, 0);
  for (count = 0; count < MAX_COUNT; count++) {
    statePin = !statePin;
    digitalWrite(ledPin, statePin);
    for (count3 = 0; count3 <= (melody[count*2] - 48) * 30; count3++) {
      for (count2=0;count2<8;count2++) {
        if (names[count2] == melody[count*2 + 1]) {
          analogWrite(speakerOut,500);
          delayMicroseconds(tones[count2]);
          analogWrite(speakerOut, 0);
          delayMicroseconds(tones[count2]);
        }
        if (melody[count*2 + 1] == 'p') {
          // make a pause of a certain size
          analogWrite(speakerOut, 0);
          delayMicroseconds(500);
        }
      }
    }
  }
}
```

Arduino ile Joistik Kullanımı



Malzemeler

- Joistik
- Jumper Kablolar
- Breadboard

Arduino ile Joistik Kullanımı

```
int x=A0;
int y=A1;
int buton = 5;

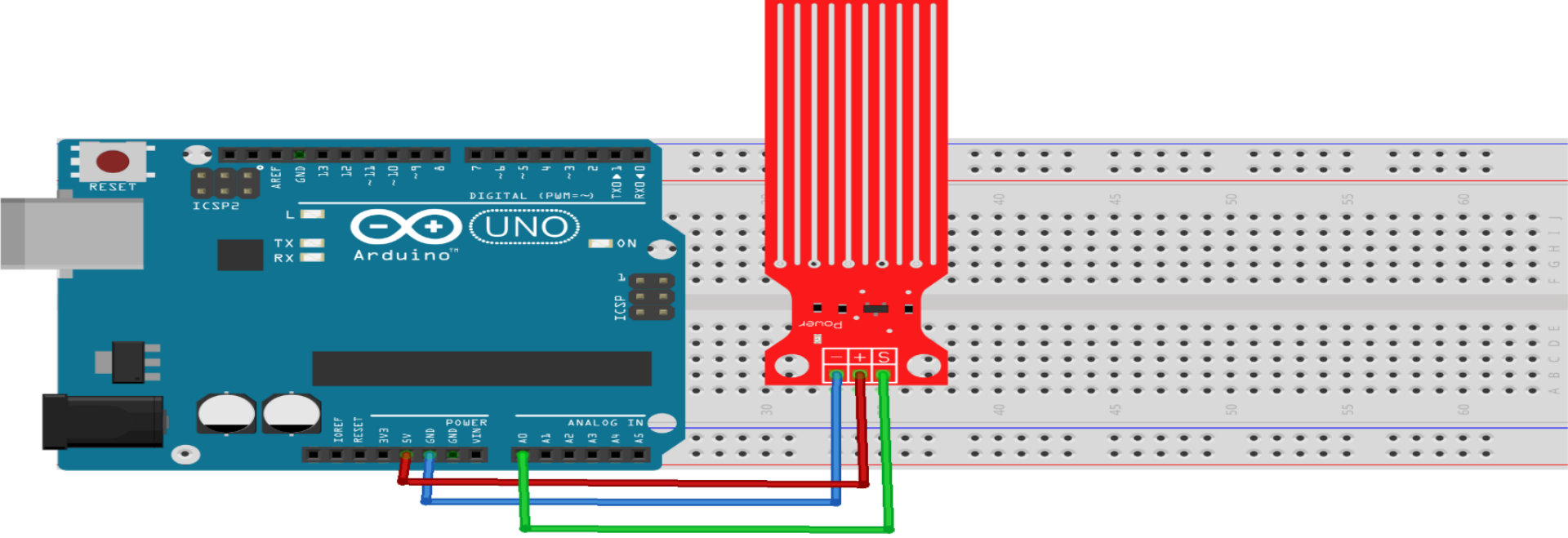
int x_durum=0;
int y_durum=0;
int buton_durum=0;

void setup()
{
  Serial.begin(9600);
  pinMode(x,INPUT);
  pinMode(y,INPUT);
  pinMode(buton,INPUT);
}

void loop()
{
  x_durum=analogRead(x);
  y_durum=analogRead(y);
  buton_durum=digitalRead(buton);

  Serial.print("X: ");
  Serial.println(x_durum);
  Serial.print("Y: ");
  Serial.println(y_durum);
  Serial.print("Buton: ");
  Serial.println(buton_durum);
  delay(1000);
}
```

Arduino ile Yağmur-Su Sensörü Kullanımı



Malzemeler

- Yağmur-Su sensörü
- Jumper Kablolar
- Breadboard

Arduino ile Yağmur-Su Sensörü Kullanımı

```
int sensor=0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  sensor=analogRead(A0);
  Serial.println(sensor);
}
```

Arduino ile CNY-70 Sensörü Kullanımı

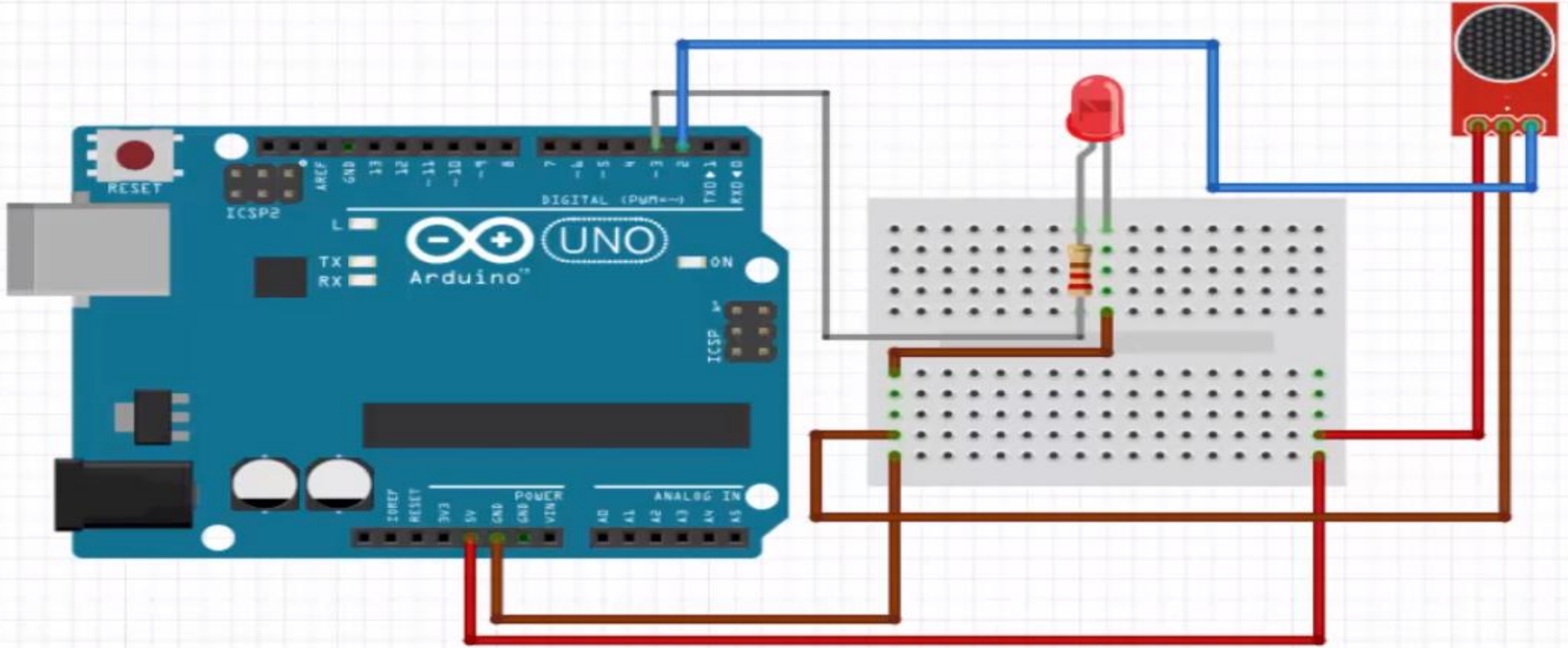
```
int referansDegeri=900;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int sensorDegeri=analogRead(A0);
  Serial.print("Sensorden okunan deger=");
  Serial.print(sensorDegeri);
  Serial.print("\t renk= ");

  if(sensorDegeri>referansDegeri)
  {
    Serial.println("beyaz");
  }
  else
  {
    Serial.println("siyah");
  }
  delay(1000);
}
```

Arduino ile Ses Sensörü Kullanımı



Malzemeler

- Ses Sensörü
- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino ile Ses Sensörü Kullanımı

```
int mikrofonsesi=A0;
int LED=3;
int sesdegeri=0;

void setup()
{
pinMode(3,OUTPUT);

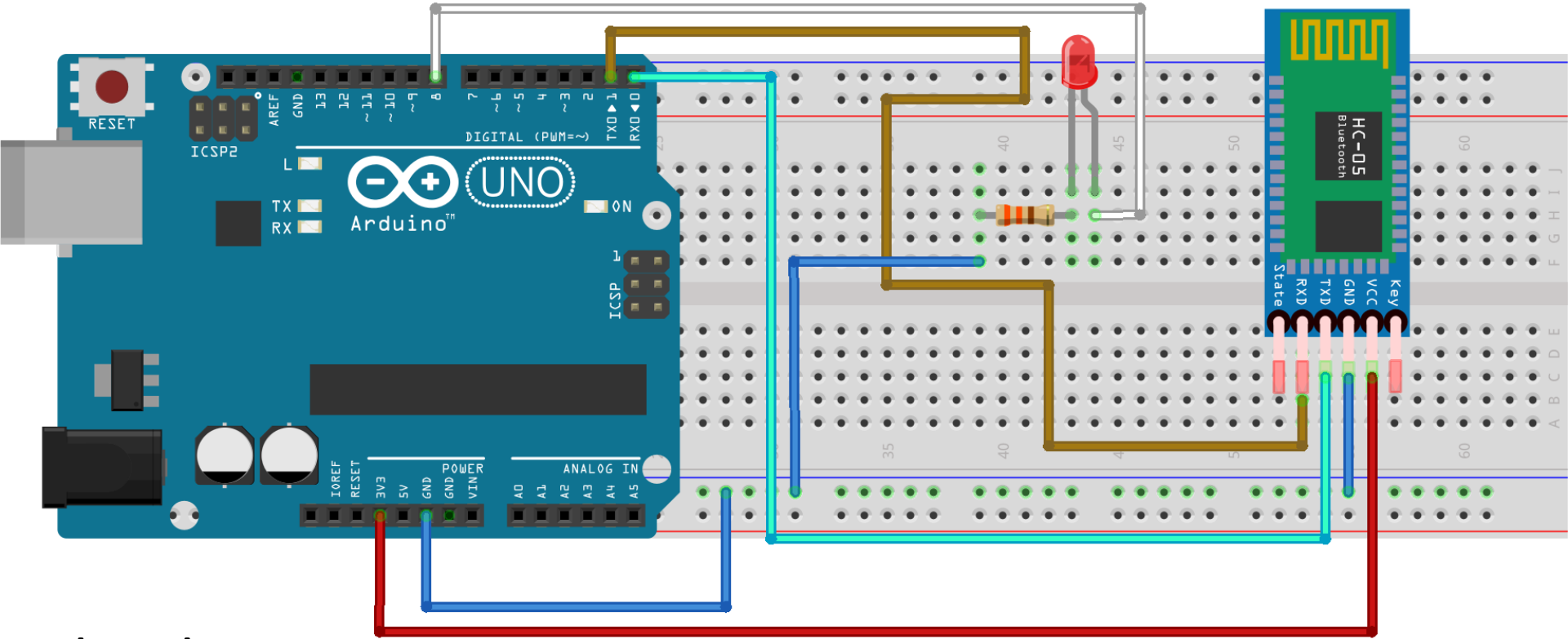
Serial.begin(9600);
}

void loop()
{
sesdegeri=analogRead(mikrofonsesi);

if (sesdegeri<30)
{
analogWrite(LED, 10);
}
else if (sesdegeri<100)
{
analogWrite(LED, 100);
}
else if(sesdegeri<400)
analogWrite(LED, 200);

Serial.println(sesdegeri);
delay(100);
}
```

Arduino ile Bluetooth Modülü Kullanımı



Malzemeler

- HC-05 Bluetooth modülü
- Led
- Direnç
- Jumper Kablolar
- Breadboard

Arduino ile Bluetooth Modülü Kullanımı

```
int data;
int led1=11;
int led2=12;
int led3=13;

void setup() {

pinMode(led1,OUTPUT);
pinMode(led2,OUTPUT);
pinMode(led3,OUTPUT);
Serial.begin(9600);
}

void loop() {
  if (Serial.available())
  {
    data = Serial.read();
    delay(100);

    if(data=='1')
    digitalWrite(led1,1);
    if(data=='2')
    digitalWrite(led1,0);
    delay(100);
  }
}
```