

# ARDUINO PROGRAMLAMA

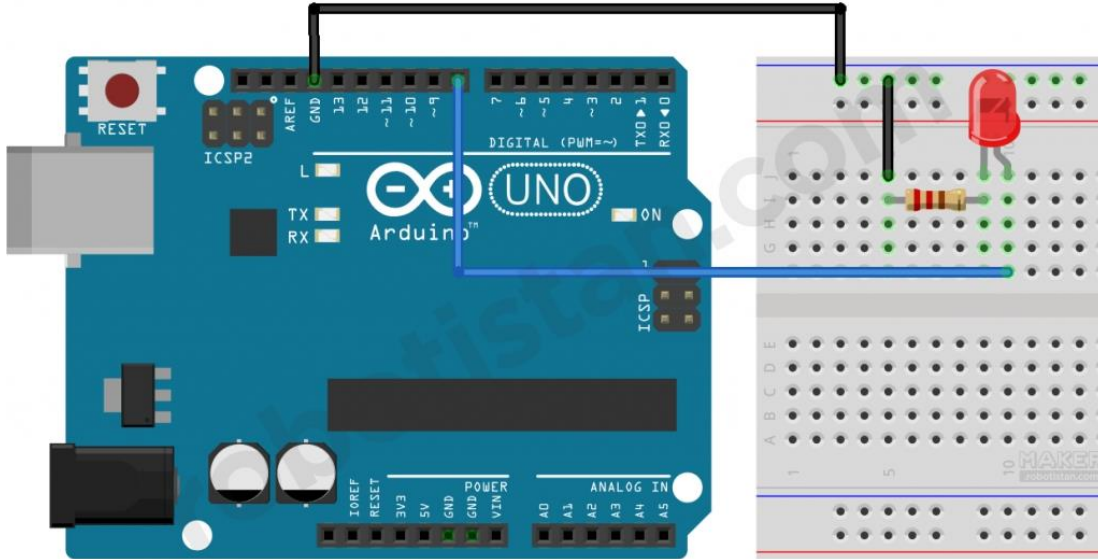
01.11.2018 Perşembe Etkinliđi

## Etkinlik 1: Arduino Ders 1: LED Yakıp Söndürme (Blink)

Gerekli Malzemeler:

- [Arduino Uno](#)
- [Breadboard](#)
- [LED](#)
- [220  \$\Omega\$  direnç](#)
- [jumper kablo](#)

Şema:



Kod:

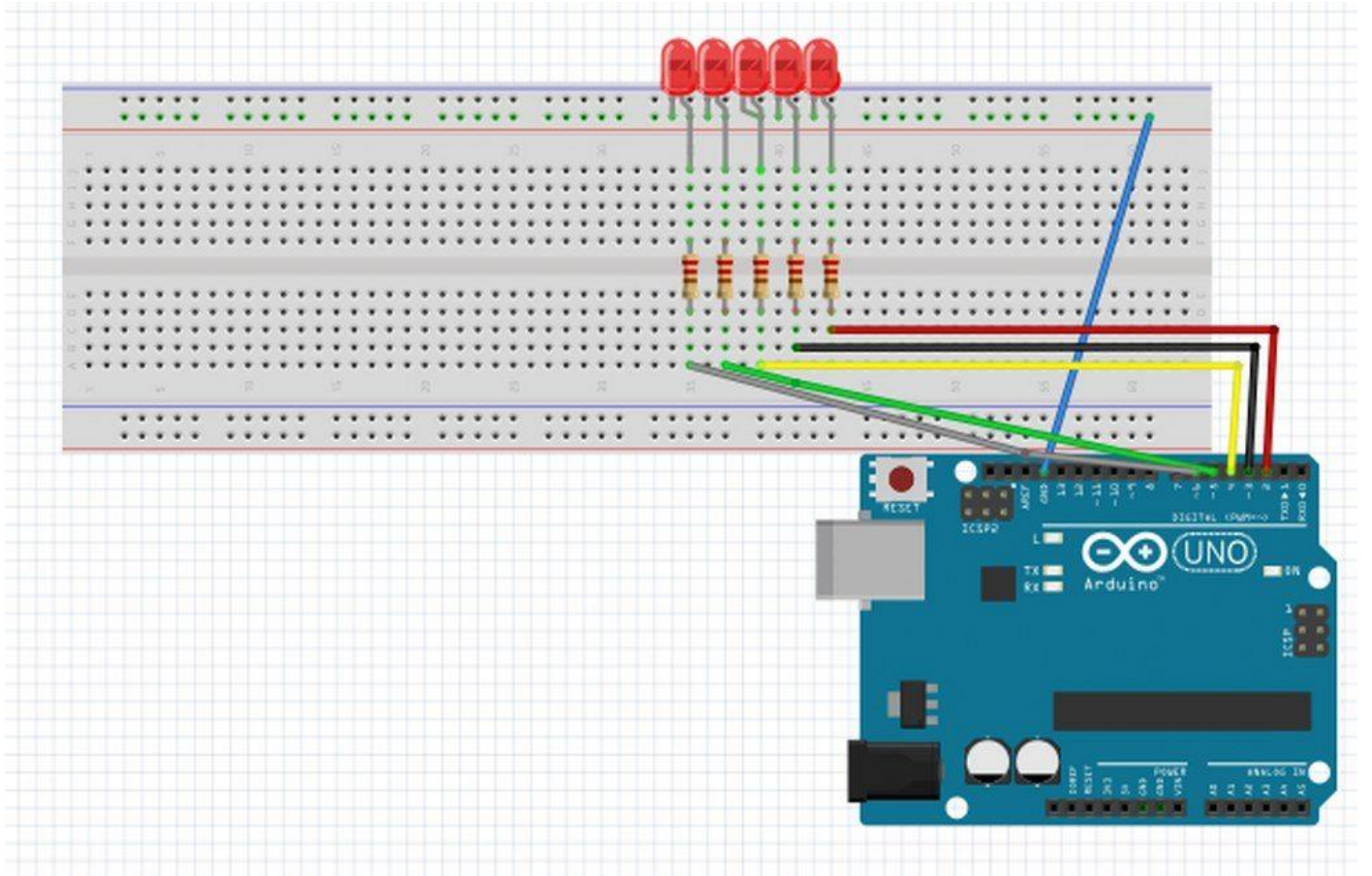
```
void setup() {  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH);  
  
  delay(1000);  
  
  digitalWrite(13, LOW);  
  
  delay(1000);  
}
```

## Etkinlik 2: Kara Şimşek Devresi İlk Uygulama

### GEREKLİ MALZEMELER

- ARDUINO
- BREAD BOARD
- 4 ADET LED
- 5 ADET 220 OHM DİRENÇ
- BİR MİKTAR JUMPER KABLO

### Şema:



### Kod:

```
voidsetup() {  
  pinMode (8, OUTPUT); //8. pin'i çıkış olarak tanımladık.  
  pinMode (9, OUTPUT); //9. pin'i çıkış olarak tanımladık.  
  pinMode (10, OUTPUT); //10. pin'i çıkış olarak tanımladık.  
  pinMode (11, OUTPUT); //11. pin'i çıkış olarak tanımladık.  
  pinMode (12, OUTPUT); //12. pin'i çıkış olarak tanımladık.  
}  
  
voidloop() {  
  digitalWrite (8, HIGH); //8'e enerji verdik.  
  delay (100); //0.1s boyunca yanmasını sağladık  
  digitalWrite (8, LOW); //8'in enerjisini kestik.  
  delay (100); //0.1s boyunca yanmasını sağladık  
  digitalWrite (9, HIGH); //9'e enerji verdik.  
  delay (100); //0.1s boyunca yanmasını sağladık  
  digitalWrite (9, LOW); //9'in enerjisini kestik.  
  delay (100); //0.1s boyunca yanmasını sağladık  
  digitalWrite (10, HIGH); // 10'e enerji verdik.  
  delay (100); //0.1s boyunca yanmasını sağladık  
  digitalWrite (10, LOW); //10'in enerjisini kestik.  
  delay (100); //0.1s boyunca yanmasını sağladık
```

```
digitalWrite (11, HIGH); // 11'e enerji verdik.  
delay (100); //0.1s boyunca yanmasını sağladık  
digitalWrite (11, LOW); //11'in enerjisini kestik.  
delay (100); //0.1s boyunca yanmasını sağladık  
digitalWrite (12, HIGH); //12'e enerji verdik.  
delay (100); //0.1s boyunca yanmasını sağladık  
digitalWrite (12, LOW); //12'in enerjisini kestik.  
}
```

## 08.11.2018 Perşembe Etkinliği

### Etkinlik 3: Kara Şimşek Devresi For Döngülü 2. Uygulama

Şema: Yukarıdaki Etkinlik 2 ile aynı

**Kod:**

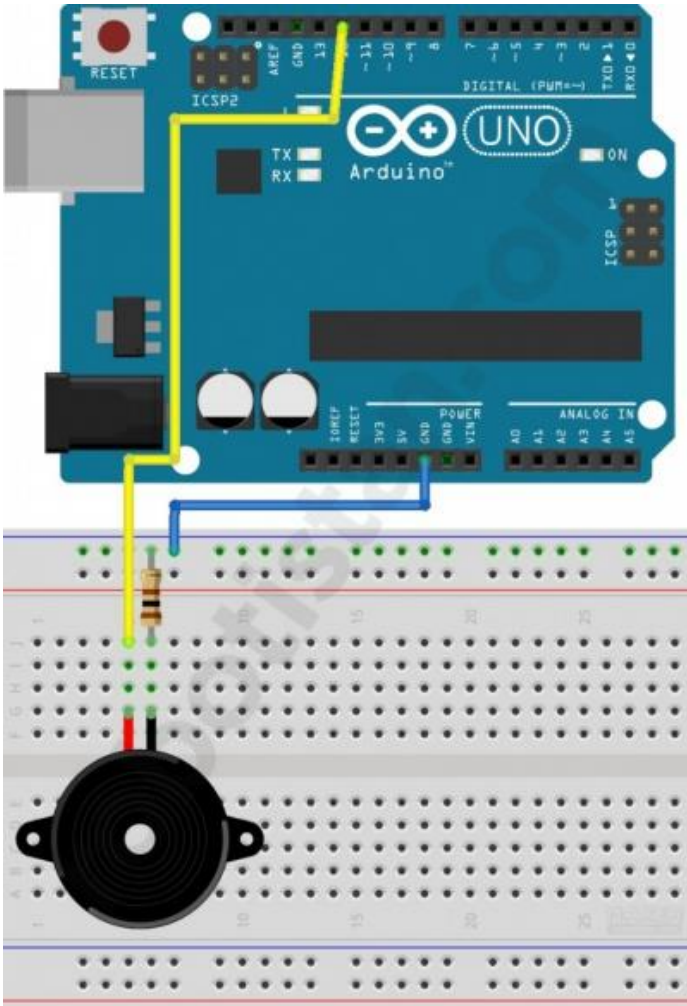
```
const int LedDizi[] = {8,9,10,11};  
  
void setup () {  
    for(int i=0; i<=3; i++)  
    {  
        pinMode(LedDizi[i], OUTPUT); // LED pinleri cikis olarak ayarlandi  
    }  
}  
  
void loop() {  
    for(int i=0; i<=3;i++)  
    {  
        digitalWrite(LedDizi[i],HIGH);  
        delay(50);  
        digitalWrite(LedDizi[i],LOW);  
    }  
  
    for(int j=3;j>-1; j--) // LEDleri geri yonde 50 milisaniye yakip sonduruyoruz  
    {  
        digitalWrite(LedDizi[j],HIGH);  
        delay(50);  
        digitalWrite(LedDizi[j], LOW);  
    }  
}
```

## Etkinlik 4: Buzzer ile nota ve mzk uygulaması

Gerekli malzemeler:

- Arduino UNO
- Breadboard
- 100  $\Omega$  direnç
- Buzzer
- İki adet jumper kablo

Şema:



**Kod:**

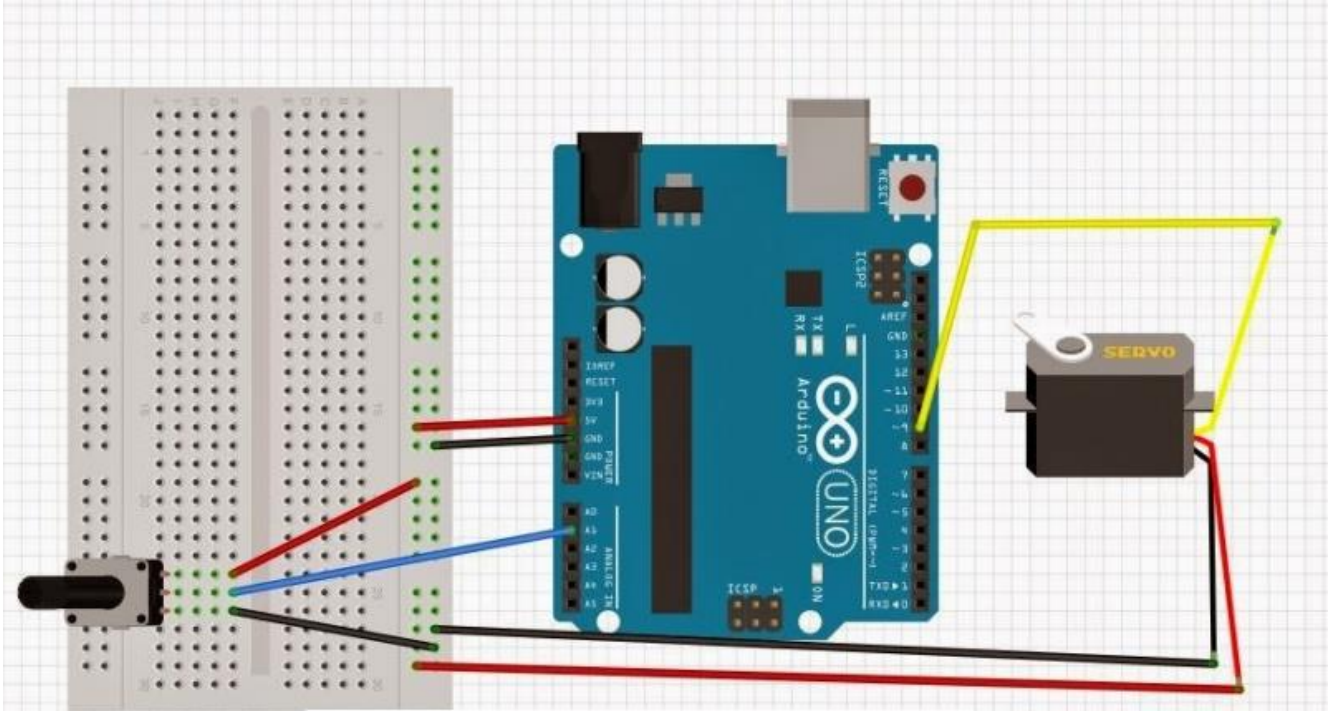
```
int buzzerPin = 12;  
int notaSayisi = 8;  
int C = 262;  
int D = 294;  
int E = 330;  
int F = 349;  
int G = 392;  
int A = 440;
```

```
int B = 494;
int C_ = 523;
int notalar[] = {C, D, E, F, G, A, B, C_};
void setup()
{
  for (int i = 0; i < notaSayisi; i++)
  {
    tone(buzzerPin, notalar[i]);
    delay(500);
    noTone(buzzerPin);
    delay(20);
  }
  noTone(buzzerPin);
}
void loop()
{
}
```

## Etkinlik 5: Potansiyometre İle servo motor kontrolü

### Gerekli Malzemeler:

- Arduino
  - 10K Potansiyometre
  - Servo
  - Bağlantı Kabloları
- Bağlantı Şeması:



Kod:

```
//Potansiyometre ile servo kontrolü
```

```
//Potansiyometre Kısaltması:Pot
```

```
-----  
#include<Servo.h>// Servo kütüphanesini ekliyoruz.
```

```
const int potPin=A1; // Servo kontrol potunu bağlayacağımız analog giriş.
```

```
Servo servomuz; // Bu komutla servo kontrol nesnemizi oluşturduk.
```

```
int potDeger=0;
```

```
int pozisyon = 0;
```

```
void setup(){
```

```
    servomuz.attach(9); // Servomuzu bağlayacağımız pini belirledik.
```

```
}
```

```
void loop(){
```

```
    potDeger=analogRead(potPin);
```

```
    pozisyon= map(potDeger,0,1023,0,180);//Pottan okunan değeri 180e oranlıyoruz.
```

```
    {
```

```
        servomuz.write(pozisyon); // Yukarıdaki oranladığımız değeri servoya yazdırıyoruz.
```

```
    }
```

```
}
```

```
-----
```