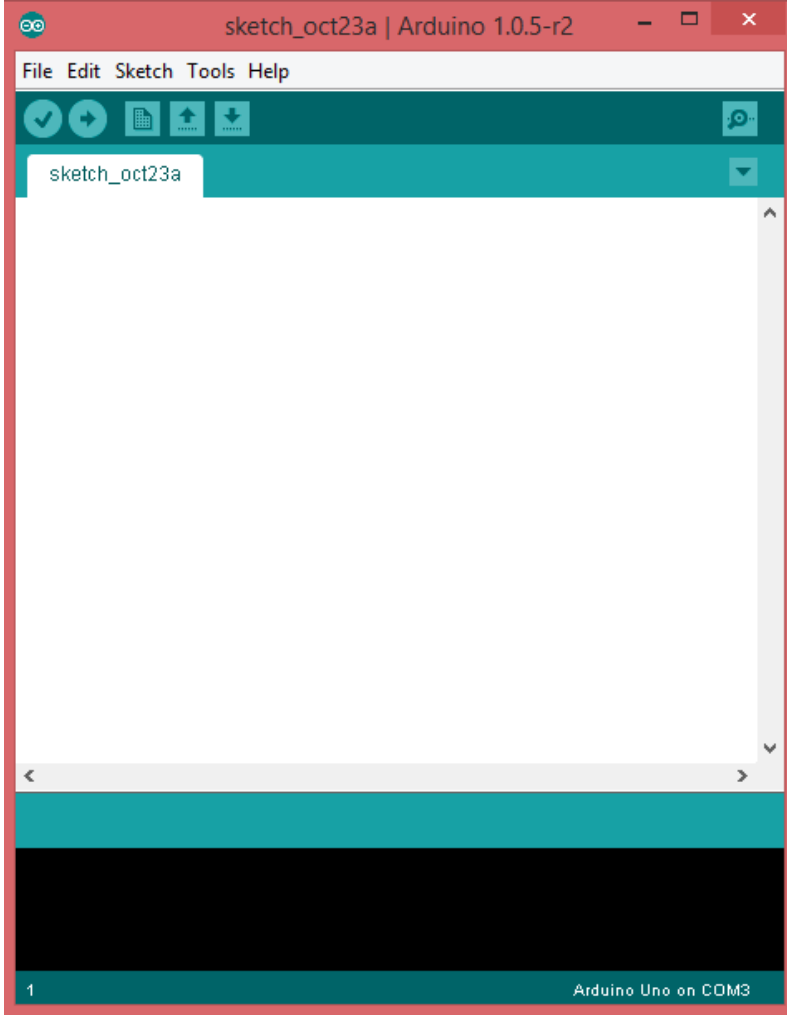


## 1. ARDUINO MİKRODENETLEYİCİ KARTININ BİLGİSAYARA TANITILMASI VE PROGRAM KURULUMU

Arduino kartını bilgisayara tanıtmak için öncelikle Arduino'nun resmi sitesinden Arduino IDE programını buradaki bağlantıdan indirip bilgisayarımıza kurmamız gerekiyor. (<https://www.arduino.cc/en/Main/Software>)

Programı, bilgisayara kurduktan sonra karşınıza aşağıdaki şekilde görülen bir arayüz çıkar.



Şekil 1. Arduino Uno arayüzü.

Artık Arduino IDE'yi kurduğumuza ve programlama arayüzüne eriştiğimize göre, şimdi sıra Arduino UNO kartını bilgisayara tanıtmaya gelmiştir. Bunun için;

1- Arduino kartını USB kablosu ile bilgisayara bağlayınız. Karşınıza ilk olarak "Windows Yeni Donanım Bulma Sihirbazı" çıkacaktır. Burada "Windows yazılım aramak için Windows Update sitesine bağlanabilir mi?" sorusuna "Hayır, bu kez değil" seçeneğini seçerek devam ediniz.

2- Yeni sayfada karşınıza çıkacak seçeneklerden bu kez "Listeden ya da belirli bir konumdan yükle (Gelişmiş)" seçeneğini seçerek ilerleyiniz.

3- Diğer sayfada "Bu konumlarda en iyi sürücüyü ara" seçeneğini seçip, "Arama şu konumu da içersin:" bölümünü işaretleyiniz ve ardından "Gözet" butonuna basarak daha önceden indirmiş olunan Arduino klasörünün içindeki "drivers" klasörünü seçiniz. (C:\arduino-1.04\drivers gibi).

- 4- Ardından karşınıza çıkan listeden Arduino UNO'ya ait sürücü bilgisini içeren "Arduino UNO"yu seçerek ilerleyiniz.
- 5- Bu aşamada karşınıza bir uyarı penceresi çıkabilir. Eğer bir uyarı penceresi çıkarsa buna "Devam et" deyiniz.
- 6- Dosyaların yüklenmesi bittikten sonra "Son" butonuna basıp yükleme tamamlanır. Şu anda Arduino UNO sürücüsü kullanıma hazırdır.

## 2. BREADBOARD

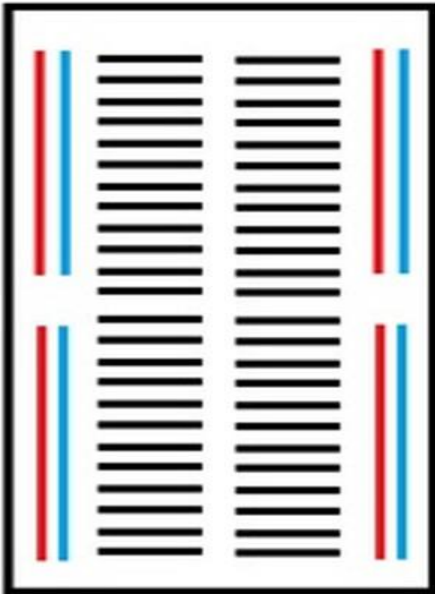
Breadboard; Türkçe'de, pratik kullanım adı ile deney bordu, deney tahtası, devre uygulama bordu gibi isimlerle anılır. Elektronik devre elemanlarının ayakları ve bağlantı kabloları, breadboard'un içindeki terminaller vasıtası ile birbirine bağlandığı için dışarıda hiçbir bağlantı gözükmez, lehimleme ihtiyaç duyulmaz. Board kullanacak olan kişiler, üzerindeki ayakların yapısını bilmeli ve devre elemanlarını ona göre yerleştirmelidir.

Elektronik Breadboard'ın üzerine harf ve rakamlardan oluşan bir koordinat sistemi yerleştirilmiştir. Hangi elemanın ayağının hangi delikte takılı olduğu koordinat sistemindeki harf ve rakamlara bakılarak anlaşılır.

Breadboard, kullanılacak elektronik elemanları bir arada tutar ve gerekli kablo bağlantılarını gerçekleştirir. Bunun üzerinde iki çeşit yol vardır: İlki **güç** yolları, yani beslememizin **artı** ve **eksi uçlarını** taktığımız yer olup resimde görülen kırmızı ve mavi şeritlerdir. Aşağıya doğru inen çizgilere karşılık gelen delikler kısa devre durumundadır. Sol üstteki kırmızından bağlanan bir kablo aynı çizgi üzerinden bağlanacak kablolar ile birleşiktir. Aynı durum mavi çizgiler için de geçerlidir.

Devrenizin çalışabilmesi için standartlara uygun olarak pilin artı ucu kırmızı çizgiye, eksi ucu ise mavi çizgiye takılmalıdır. Bazı breadboardlarda yanlarda bulunan besleme hatları ikiye bölünmüş olduğu gibi, bazılarında ise güç hatları tüm hat boyunca (yukarıdan aşağıya kadar) birbirine bağlıdır.

Yataydaki siyah çizgiler ise orta boşluğa kadar birbirine bağlıdır. Breadboard üzerinde nesnelere (jumper kablo, direnç, potansiyometre vb.) bağlantılarını gerçekleştirmek için bu yatay çizgiler kullanılabilir.

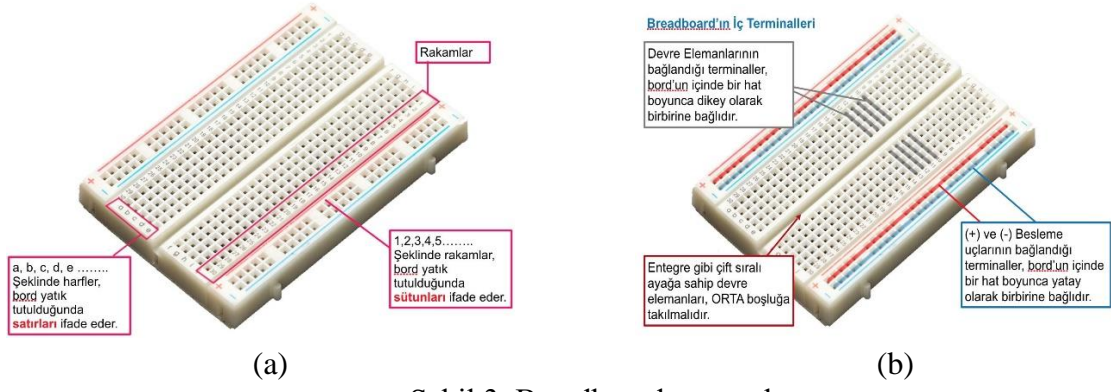


Yandaki görselde bir Breadboard'un iç yapısı görülmektedir. Bu görselden, Breadboard'daki deliklerin hangilerinin birbirine bağlı olduğunu anlayabilirsiniz.

Breadboard'un içindeki devre elemanlarının takıldığı terminaller aşağıdaki gibi bir yapıya sahiptir. 5'li grup halinde bağlıdır. Terminal uçları, devre elemanlarını sıkıştırarak şekilde yay gibi imal edilmiştir. Eleman takıldığı zaman bu uçlar aralanır. Eleman söküldüğü zaman, tekrar eski haline döner.

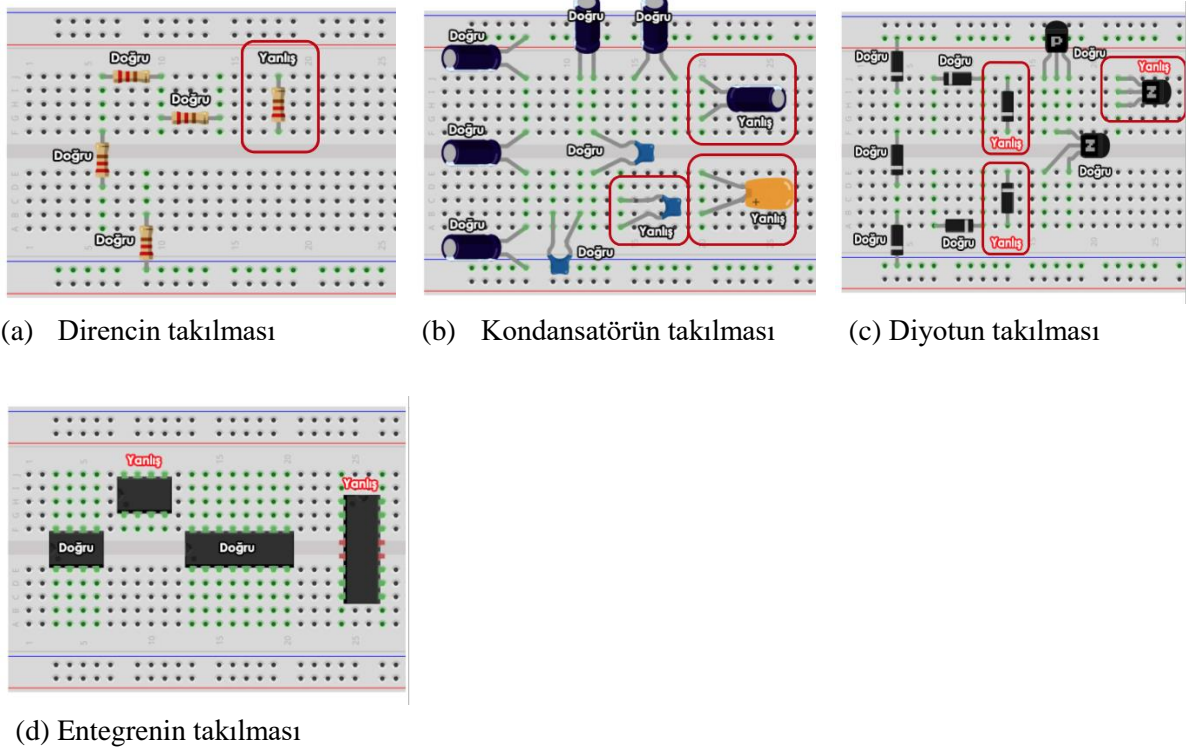
Bu terminallere çok kalın ayaklı (röle, anahtar, kalın kablo, güçlü diyot v.b) devre elemanları takılmaz. Bu elemanlar, terminalleri genişletir ve esnekliğinin kaybolmasına yol açar. Bunun sonucunda, takılan devre elemanları arasında temassızlıklar meydana gelir.

Şekil 2. Bir breadboard yapısı.



Şekil 3. Breadboard tanıtımları.

### Pasif Elemanların Breadboard'a Takılması



Şekil 4. Breadboard üzerine değişik elektronik elemanların yerleştirilmesi

## DENEY NO: 1

### DENEY KONUSU: Kayarak Yanan LED'ler

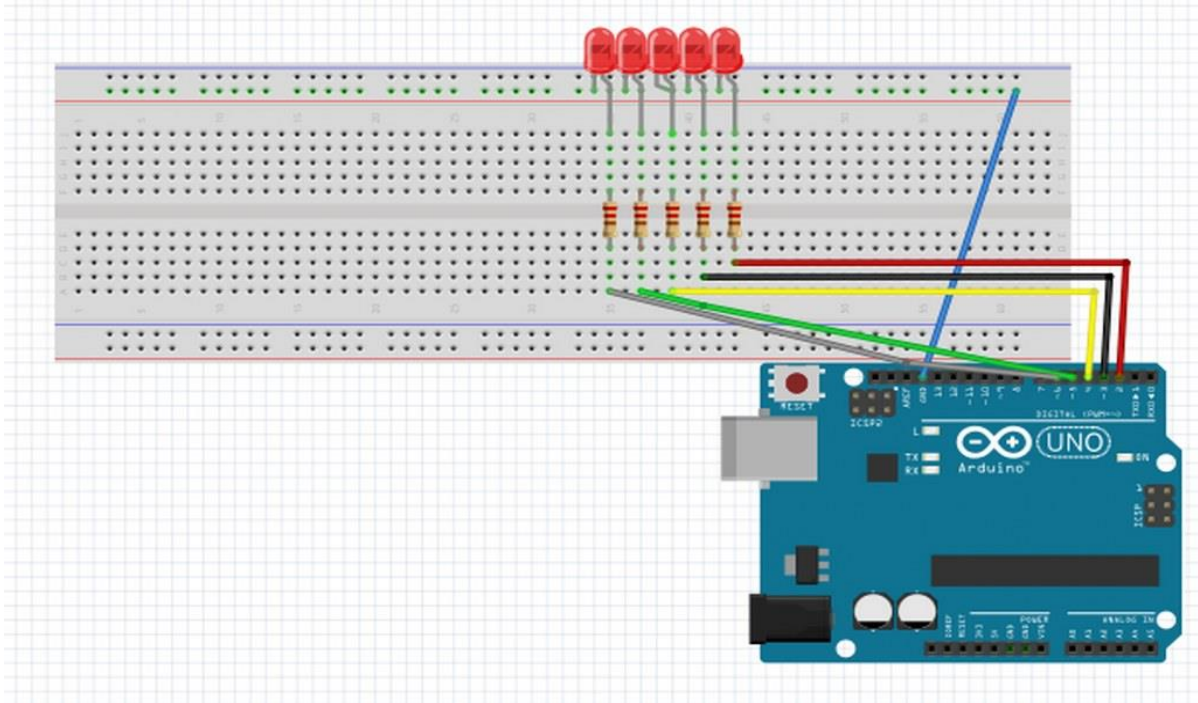
**ÖZET:** Bu deneysel uygulama, Arduino mikrodenetleyicili kartını kullanarak breadboard üzerinde bulunan LED'lerin sırayla yakıp söndürme ile ilgilidir.

### DENEY MALZEMELERİ :

- 1 Adet Arduino UNO
- 1 Adet Breadbord
- 4 Adet LED
- 5 Adet 220  $\Omega$ 'luk Direnç
- Yeteri kadar jumper kablolar

### DEVRENİN KURULMASI VE ÇALIŞTIRMALAR

1. Aşağıda şeması verilen devreyi inceleyerek kuruluma adımlarına geçiniz.



Şekil 5. Arduino UNO, breadboard, LED, direnç ve jumper'lar ile kurulacak devre şeması.

2. LED'leri breadboard üzerine yerleştiriniz.
3. LED'lerin bir ucuna dirençleri, dirençlerin diğer ucuna ise jumper kabloları bağlayınız ki böylece Arduino ile breadboard arasındaki bağlantıyı gerçekleştirilmiş olur.
4. LED'lerin diğer ucuna, breadboard üzerinden GND (topraklama) ucu bağlantısı yapılır.
5. Şimdi, bilgisayarınızdaki Arduino programını çalıştırınız.
  - a) Arduino Uno arayüzünde; Tools menüsü altındaki Serial Port seçeneğine geliniz ve Arduino'nun bağlı olduğu portu seçiniz.
  - b) Sonra yine Tools menüsündeki Board seçeneğinden kullanılacak olan Arduino modelini seçiniz.
6. Ardından aşağıda verilen kodları eksiksiz bir şekilde Arduino programına atınız.

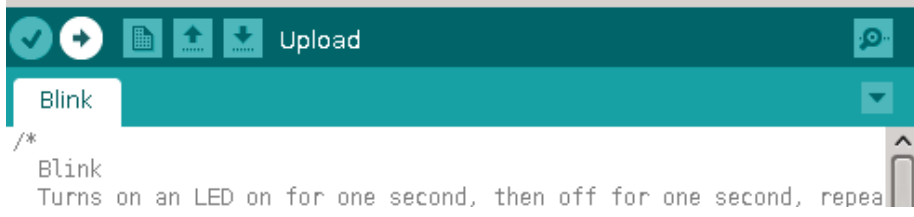
**Kodlar:**

```

//hso sırayla yanıp sönen ledler(kara simsek)
int aled = 3; //3. pin'i aled olarak tanımladık.
int bled = 4; //4. pin'i bled olarak tanımladık.
int cled = 5; //5. pin'i cled olarak tanımladık.
int dled = 6; //6. pin'i dled olarak tanımladık.
int eled = 7; //7. pin'i eled olarak tanımladık.
void setup()
{
    pinMode (aled, OUTPUT); //8. pin'i çıkış olarak tanımladık.
    pinMode (bled, OUTPUT); //9. pin'i çıkış olarak tanımladık.
    pinMode (cled, OUTPUT); //10. pin'i çıkış olarak tanımladık.
    pinMode (dled, OUTPUT); //11. pin'i çıkış olarak tanımladık.
    pinMode (eled, OUTPUT); //12. pin'i çıkış olarak tanımladık.
}
void loop()
{
    digitalWrite (aled, HIGH); //aled'e enerji verdik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (aled, LOW); //aled'in enerjisini kestik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (bled, HIGH); //bled'e enerji verdik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (bled, LOW); //bled'in enerjisini kestik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (cled, HIGH); // cled'e enerji verdik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (cled, LOW); //cled'in enerjisini kestik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (dled, HIGH); // dled'e enerji verdik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (dled, LOW); //dled'in enerjisini kestik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (eled, HIGH); //eled'e enerji verdik.
    delay (100); //0.1s boyunca yanmasını sağladık
    digitalWrite (eled, LOW); //eled'in enerjisini kestik.
}

```

7. Yukarıda verilen program kodlarını yazdıktan sonra, bunları Arduino programına yüklemek için, aşağıdaki şekilde görünen üst sekmedeki **Sağ Ok** simgesine tıklayınız.



Şekil 6. İlk uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi işlemi.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---

---

## ÇALIŞMA SORULARI

- 5 LED'in aynı anda 5 saniye yanıp sönmelerini sağlayan Arduino program kodlarını yazınız.
- Önce ilk 3 LED yansın; sonra, diğer 2 LED yansın ve böylece LED'lerin yanış sıralarını 3, 2, 3, 2, ... şeklinde ilerleten Arduino program kodlarını yazınız.



## DENEY NO: 2

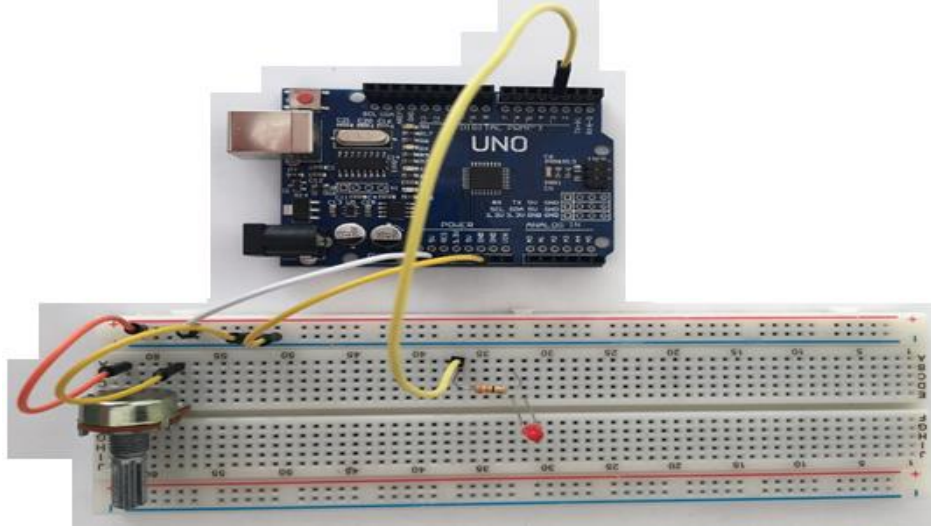
### DENEY KONUSU: Potansiyometre İle LED Parlaklığı Ayarlama

**ÖZET:** Bu deneysel uygulama, Arduino mikrodenetleyicisini kullanarak yakılan LED'lerin parlaklıklarını ayarlamakla ilgilidir.

#### DENEY MALZEMELERİ:

- 1 Adet Arduino UNO
- 1 Adet 10K $\Omega$  Potansiyometre
- 1 Adet 330K $\Omega$  Direnç
- 1 Adet LED
- 1 Adet Breadboard
- 5 Adet Jumper kabloları

#### DEVRE ŞEMASI :



Şekil 7.

#### İŞLEM ADIMLARI:

1. LED'i ve potansiyometreyi breadboard üzerine yerleştiriniz.
2. LED'in bir ucuna dirençleri, dirençlerin diğer ucuna Arduino ile breadboard arasındaki bağlantıyı jumper kabloları ile sağlayınız.
3. LED'lerin diğer ucuna breadboard üstünden GND pini bağlantısını yapınız.
4. Şimdi, Arduino programına giriniz. Tools menüsü altındaki Serial Port seçeneğinden Arduino'nun bağlı olduğu port'u seçiniz.
5. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak Arduino modelini seçiniz.
6. Ardından aşağıdaki kodları Arduino ortamında eksiksiz bir şekilde yazınız.

#### Kodlar:

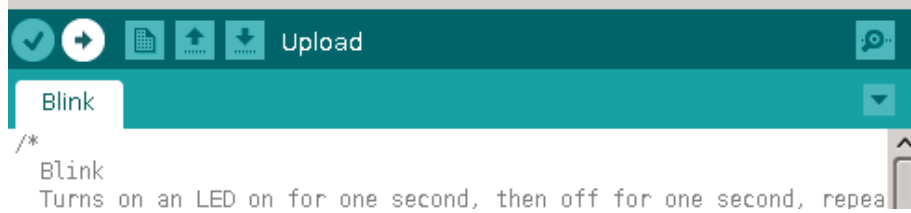
```
const int led=3;  
const int potansiyometre=0;  
int parlaklik;  
void setup()
```

```

{
    pinMode(led,OUTPUT);
}
void loop()
{
    parlaklik=analogRead(potansiyometre) / 4;
    analogWrite(led,parlaklik);
}

```

7. Kodları yazdıktan sonra, bunları Arduino programına yüklemek için aşağıdaki şekilde görülen **Sağ Ok** sekmesine tıklayınız.



Şekil 8. 2. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---



---

## ÇALIŞMA SORULARI

- 3'er saniye aralıklarla 1 LED'in yanıp sönmesini sağlayan Arduino program kodlarını yazınız.

## DENEY NO: 3

### DENEY KONUSU: Arduino Yardımı İle Buzzer Kullanarak Ses Elde Etme Deneyi

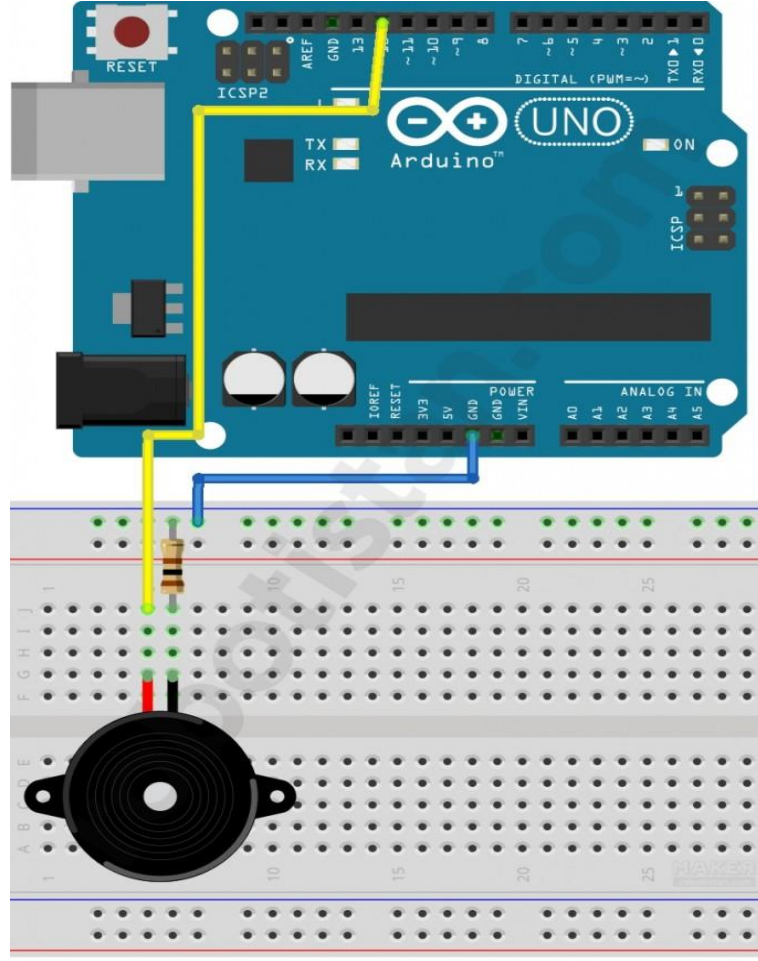
**ÖZET:** Bu deneysel uygulama, Arduino mikrodenetleyicili kartını kullanarak bir buzzer yardımı ses çıkışını sağlamak.

### DENEY MALZEMELERİ:

- 1 adet Buzzer
- 1 adet Arduino
- 1 adet Direnç

Yeteri kadar jumper kablo



**DEVRE ŞEMASI:**

Şekil 9. Arduino yardımı ile buzzer kullanarak ses üretme bağlantı şeması.

**İŞLEM ADIMLARI:**

1. Breadboard üzerine bir buzzer ve direnci, yukarıda verilen şemada görüldüğü gibi bağlayınız.
2. Arduino pinlerini doğru şekilde bağladıktan sonra Arduino programını çalıştırınız.
3. Tools menüsü altında Serial Port seçeneğine gelerek Arduino'nun bağlı olduğu portu seçiniz.
4. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak Arduino modelini seçiniz.
5. Ardından aşağıdaki kodları Arduino ortamında eksiksiz bir şekilde yazınız.

**Kodlar:**

```
int buzzerPin=12; //Buzzer'in + bacağına bağlı olduğu pin
void setup(){
  pinMode(buzzerPin,OUTPUT); // Buzzer'in bağlı olduğu pini çıkış olarak ayarladık.
}
```

```
void loopp(){  
    digitalWrite(buzzerPin.HIGH); //Buzzer'a güç göndererek ses çıkarmasını sağladık.  
    delay(1000); //1 saniye boyunca çalışmasını söyledik.  
    digitalWrite(buzzerPin.LOW); //Buzzer'ın gücünü keserek susmasını sağladık.  
    delay(1000); //1 saniye boyunca susmasını söyledik.  
}
```







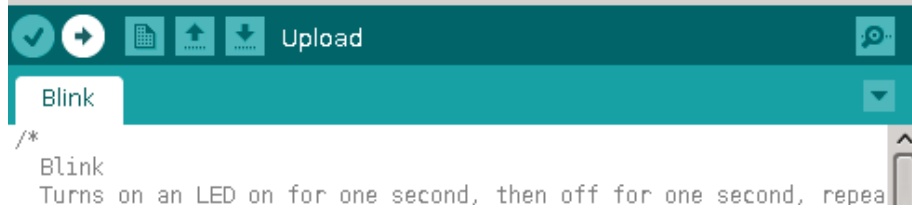








6. Kodları yazdıktan sonra Arduino programına yüklemek için resimde görünen **Sağ Ok** sekmesine tıklayınız.



Şekil 10. 3. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---

---

## ÇALIŞMA SORULARI

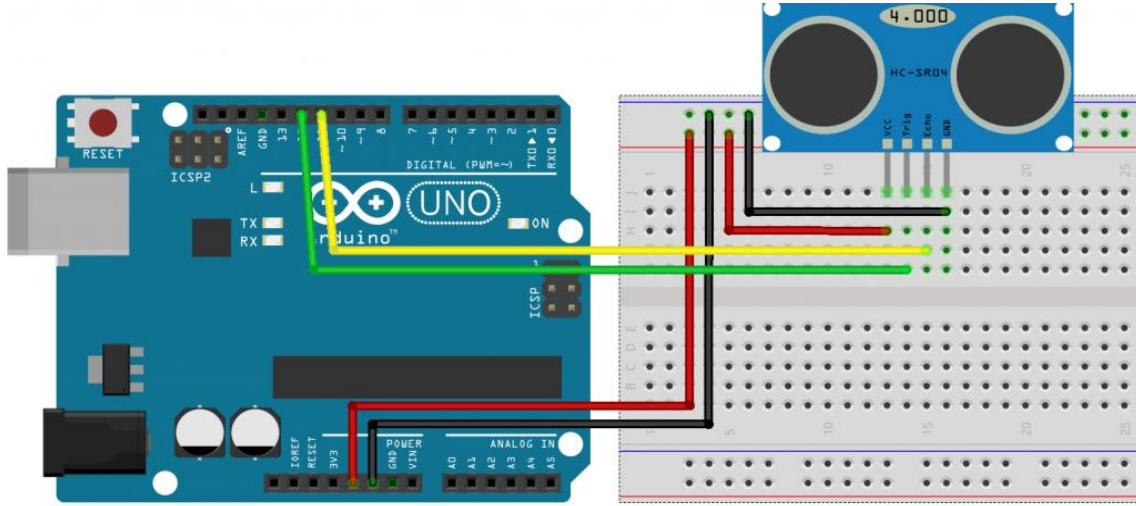
- İstiklal marşı için Arduino kodlarını internetten bularak Arduino kartındaki mikro denetleyiciye yükleyiniz. Devrenin çalışmasını sağlayacak deneyi yapınız.

**DENEY NO: 4****DENEY KONUSU: Arduino Kartına Mesafe Sensörü Bağlayarak Ekranı Uzaklık Miktarını Yazdırmak**

**ÖZET:** Bu deneysel uygulama, Arduino mikrodenetleyicili kartını kullanarak bir “mesafe sensörü” yardımı ile oluşacak uzaklık değerlerini Seri Port ekranına yazdırma uygulamasıdır.

**DENEY MALZEMELERİ:**

- 1 adet Arduino
- 1 adet Mesafe sensörü
- Yeteri kadar Jumper kabloları

**DEVRE ŞEMASI:**

Şekil 11. Arduino ve mesafe sensörü kullanarak ekrana uzaklığı yazdıran deney bağlantı şeması.

**İŞLEM ADIMLARI:**

1. Arduino ile mesafe sensörü arasındaki bağlantıyı yukarıda verilen şemada gösterildiği gibi yapınız.
2. Arduino programını çalıştırınız. Tools menüsü altındaki Serial Port seçeneğine gelerek Arduino'nun bağlı olduğu portu seçiniz.
3. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak olan Arduino'nun modelini seçiniz.
4. Ardından aşağıda verilen kodları, Arduino ortamında eksiksiz bir şekilde yazınız.
5. Ardından Araçlar menüsünden Seri Port Ekranı seçeneğinden ekranı açarak ölçümü izleyiniz.

**Kodlar:**

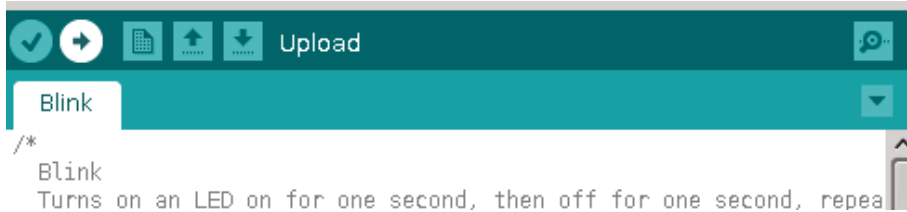
```
int trigPin = 12; /* Sensorun trig pini Arduinonun 12 numaralı ayağına bağlandı */
int echoPin = 11; /* Sensorun echo pini Arduinonun 11 numaralı ayağına bağlandı */
```

```

long sure;
long uzaklik;
void setup(){
  pinMode(trigPin, OUTPUT); /* trig pini çıkış olarak ayarlandı */
  pinMode(echoPin,INPUT); /* echo pini giriş olarak ayarlandı */
  Serial.begin(9600); /* Seri haberleşme başlatıldı */
}
void loop()
{
  digitalWrite(trigPin, LOW); /* sensör pasif hale getirildi */
  delayMicroseconds(5);
  digitalWrite(trigPin, HIGH); /* Sensöre ses dalgasının üretmesi için emir verildi */
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW); /* Yeni dalgaların üretilmemesi için trig pini LOW konumuna getirildi */
  /*
  sure = pulseIn(echoPin, HIGH); /* ses dalgasının geri dönmesi için geçen süre ölçülüyor */
  uzaklik= sure /29.1/2; /* ölçülen süre uzaklığa çevriliyor */
  if(uzaklik > 200)
    uzaklik = 200;
  Serial.print("Uzaklik ");
  Serial.print(uzaklik); /* hesaplanan uzaklık bilgisayara aktarılıyor */
  Serial.println(" CM olarak ölçülmüştür.");
  delay(500);
  */
}

```

6. Kodları yazdıktan sonra Arduino programına yüklemek için resimde görünen **Sağ Ok** sekmesine tıklayınız.



Şekil 12. 4. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---



---

## ÇALIŞMA SORULARI

- Mesafe sensörü ile yapılan ölçüme bağlı olarak aşağıdaki senaryo durumlarını karşılayacak olan Arduino kodlarını yazarak istenenleri gerçekleştiriniz.
  - i> Mesafe 0-10 cm arasında ise buzzer'dan yüksek limitte 50 ms aralıklarla, düşük limitte 50 ms aralıkla ses çıkışı,

- ii> Mesafe 10-30 cm arasında ise buzzer'dan yüksek limitte 30 ms, düşük limitte 250 ms aralıklarla ses çıkışı,
- iii> Mesafe 30-50 cm arasında ise buzzer'dan yüksek limitte 50 ms, düşük limitte 500 ms aralıklarla ses çıkışı,
- iv> Mesafe 50 cm'den daha uzak ise buzzer hiçbir işlem yapmasın,
- v> Mesafe 0 cm ve altında ise buzzer'dan düşük limitte sonsuz ses çıkışı sağlansın.

### DENEY NO: 5

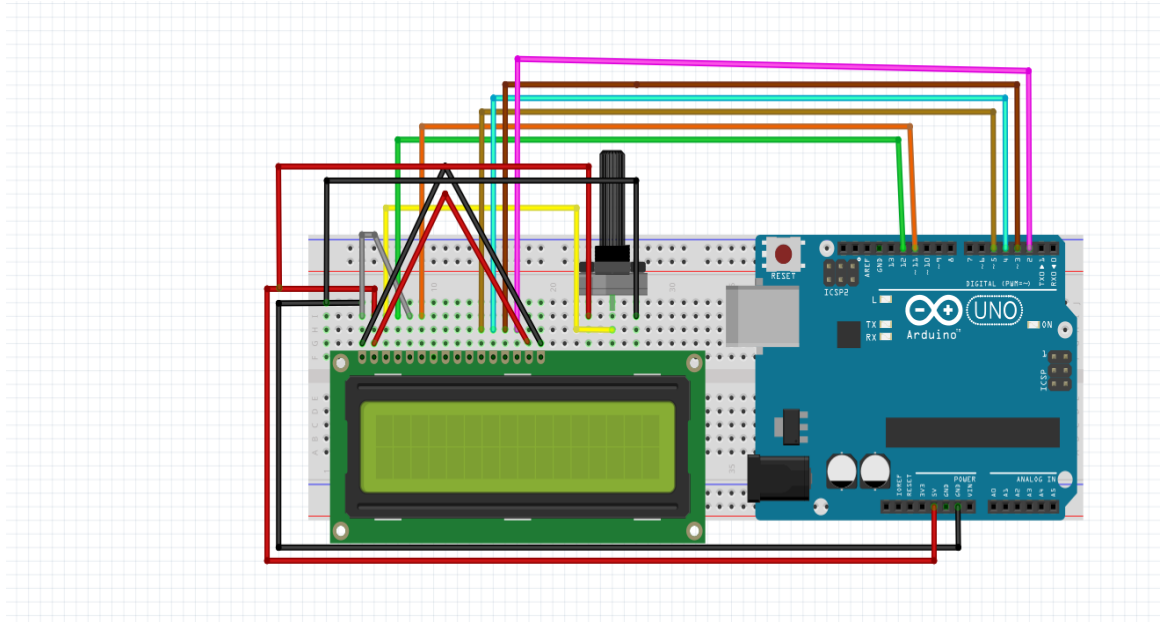
### DENEY KONUSU: Arduino Yardımı ile LCD Ekranda Yazı Yazdırma

**ÖZET:** Bu deney, Arduino mikrodenetleyicili kartını kullanarak bir LCD ekranda yazı yazdıran bir uygulamadır.

### DENEY MALZEMELERİ:

- 1 adet LCD ekran
- 1 adet Arduino
- Yeteri kadar jumper kablo
- 1 Adet potansiyometre

### DEVRE ŞEMASI:



Şekil 13. Arduino ve LCD ekran kullanarak LCD ekranda yazı yazdırma deneyi bağlantı şeması.

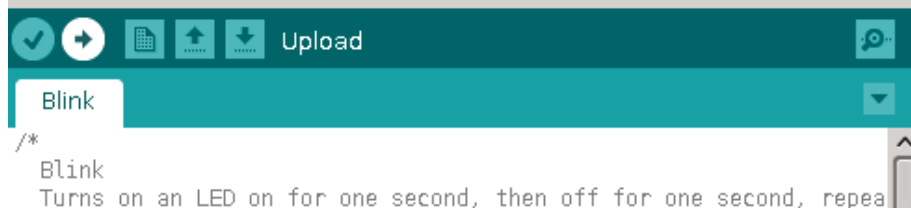
### İŞLEM ADIMLARI:

1. Breadboard üzerine bir LCD ekran ve potansiyometreyi, yukarıda verilen şemada görüldüğü gibi bağlayınız.
2. Arduino pinlerini doğru şekilde bağladıktan sonra Arduino programını çalıştırınız.
3. Tools menüsü altında Serial Port seçeneğine gelerek Arduino'nun bağlı olduğu portu seçiniz.
4. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak Arduino modelini seçiniz.
5. Ardından aşağıdaki kodları Arduino ortamında eksiksiz bir şekilde yazınız.

### **Kodlar:**

```
#include <LiquidCrystal.h>
lcd(12,11,5,4,3,2);
void setup()
{
  lcd.begin(16,2);
  lcd.clear();
  lcd.print("MSKU BSM");
  lcd.setCursor(0,1);
  lcd.print("Merhaba Dunya");
}
void loop()
{
}
```

6. Kodları yazdıktan sonra Arduino programına yüklemek için resimde görünen **Sağ Ok** sekmesine tıklayınız.



Şekil 14. 5. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

### **UYGULAMA GÖZLEM VE SONUÇLARI**

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---



---

### **ÇALIŞMA SORULARI**

- 1'den 5'e kadar sayıları, 10'ar saniye aralıklarla bir LCD ekranda yazdıracak olan yazılım kodlarını yazınız.

- 5'er saniye aralıklarla ekranda, "**Merhaba Dünya**" metnini gösteren yazılım kodlarını yazınız.

### DENEY NO: 6

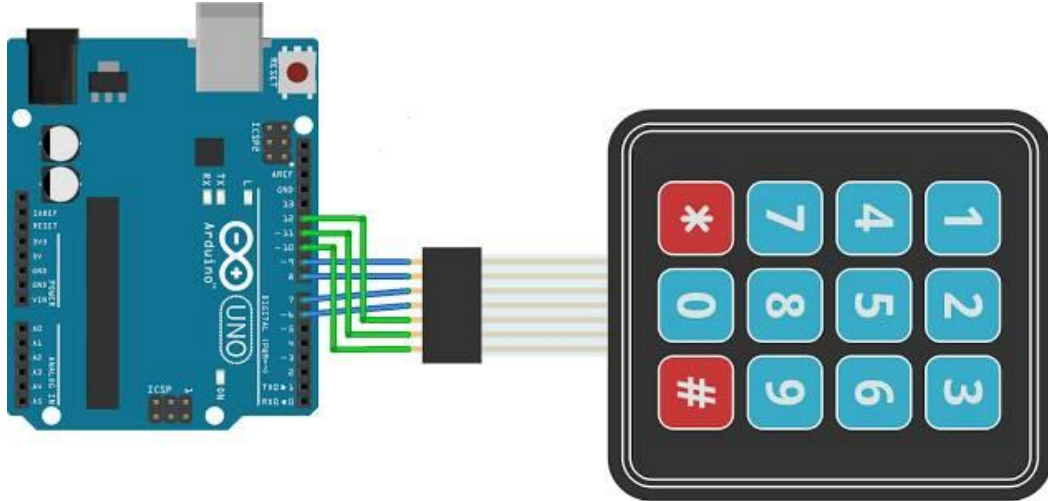
### DENEY KONUSU: Arduino'da Keypad Kullanımı

**ÖZET:** Bu deneysel uygulama, Arduino mikrodenetleyicili kartını kullanarak bir "keypad" tuşları ile basılan karakterleri bir ekranda yazdırma uygulamasıdır.

### DENEY MALZEMELERİ:

- 1 adet Arduino
- 1 adet Keypad (tuş takımı)
- Yeteri kadar Jumper kabloları

### DEVRE ŞEMASI:



Şekil 15.

### İŞLEM ADIMLARI:

1. Arduino ile Keypad arasındaki bağlantıyı yukarıda verilen şemada gösterildiği gibi yapınız.
2. Arduino programını çalıştırınız. Tools menüsü altındaki Serial Port seçeneğine gelerek Arduino'nun bağlı olduğu portu seçiniz.

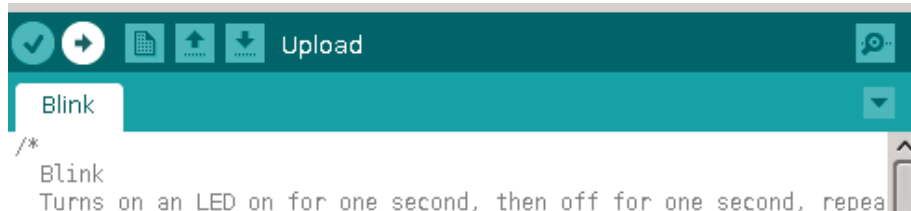


3. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak olan Arduino'nun modelini seçiniz.
4. Ardından aşağıda verilen kodları, Arduino ortamında eksiksiz bir şekilde yazınız.

**Kodlar:**

```
#include <Keypad.h>
const byte SATIR = 4;
const byte SUTUN= 3;
char keys[SATIR][SUTUN] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'#','0','*'}
};
byte rowPins[SATIR] = { 9, 8, 7, 6 };
byte colPins[SUTUN] = { 12, 11, 10 };
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, SATIR, SUTUN );
void setup()
{
  Serial.begin(9600);
  pinMode(A0, OUTPUT);
}
void loop()
{
  char key = kpd.getKey();
  if(key)
  {
    Serial.println(key);
    delay(15);
  }
}
```

5. Kodları yazdıktan sonra Arduino programına yüklemek için resimde görünen **Sağ Ok** sekmesine tıklayınız.



Şekil 16. 6. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---

---

### ÇALIŞMA SORULARI

- Önceden belirlenmiş bir sayıyı şifre olarak tanımlayınız. Sonra, bir keypad'den girilen rakam ile bu sayıyı karşılaştırıp doğru ise "Tebrikler", yanlış girildiyse "Üzgünüm. Yanlış Şifre!" mesajını ekrana yazdıracak olan Arduino kodlarını yazınız.
- 0 ile 10 arasında rastgele bir sayı üretiniz. Sonra bir keypad'den girilen rakam ile üretilen bu sayıyı karşılaştırıp doğru ise "Tebrikler", yanlış girildiyse "Üzgünüm. Yanlış Sayı!" mesajını ekrana yazdıracak olan Arduino kodlarını yazınız.

### DENEY NO: 7

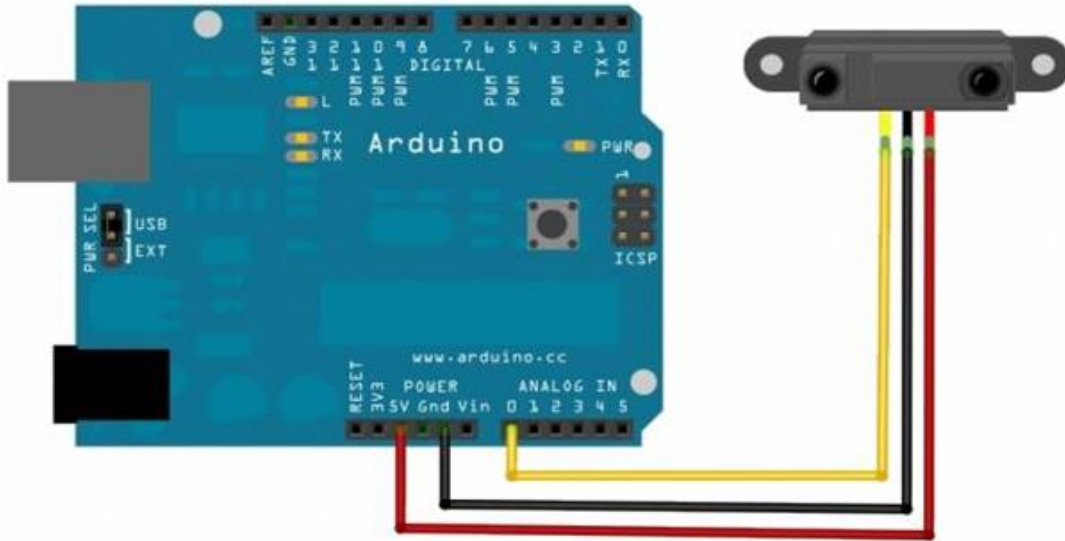
#### DENEY KONUSU: Arduino İle Bir Kızılötesi Sensörü Kullanmak

**ÖZET:** Bu deney, Arduino mikrodenetleyicili kartını kullanarak bir "kızılötesi sensörü" ile bir mesafeyi ölçerek ölçülen değeri ekrana yazdırmaya ilişkin bir uygulamadır.

#### DENEY MALZEMELERİ:

- 1 Adet Arduino UNO,
- 1 Adet kızılötesi sensörü
- 3 Adet Jumper kablo

#### DEVRE ŞEMASI:



Şekil 17.

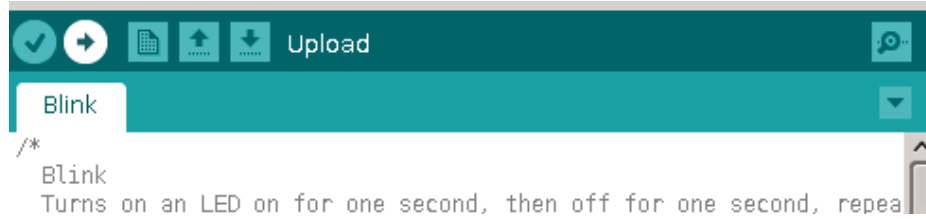
#### İŞLEM ADIMLARI:

1. Arduino ile bir kızılötesi sensörü, yukarıdaki şemada gösterildiği gibi birbirine bağlayınız.
2. Arduino programını çalıştırınız. Tools menüsü altındaki Serial Port seçeneğine gelerek Arduino'nun bağlı olduğu portu seçiniz.
3. Sonra, yine Tools menüsündeki Board seçeneğinden kullanılacak olan Arduino'nun modelini seçiniz.
4. Ardından aşağıda verilen kodları, Arduino ortamında eksiksiz bir şekilde yazınız.

### **Kodlar:**

```
int sensorpin = 0;
int val = 0;
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  val = analogRead(sensorpin);
  Serial.println(val);
  delay(100);
}
```

5. Kodları yazdıktan sonra Arduino programına yüklemek için resimde görünen **Sağ Ok** sekmesine tıklayınız.



Şekil 18. 7. Uygulama için yazılan Arduino kodlarının, arayüzden upload edilmesi.

## **UYGULAMA GÖZLEM VE SONUÇLARI**

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---

---

## **ÇALIŞMA SORULARI**

- ✓ Ölçülen mesafe eğer 0-10 arasında ise ekranda “**yakın**”, 11-20 arasında ise ekranda “**uzak**” mesajını yazdıracak olan Arduino kodlarını yazınız.

- ✓ Devreye 1 adet LED'i bağlayarak mesafeyi ölçünüz. Eğer ölçülen mesafe, 20 değerinden daha büyük değerde ise LED'i yakan, 20 değerinden küçük ise LED'i söndüren program kodlarını yazınız.

## **DENEY NO: 8**

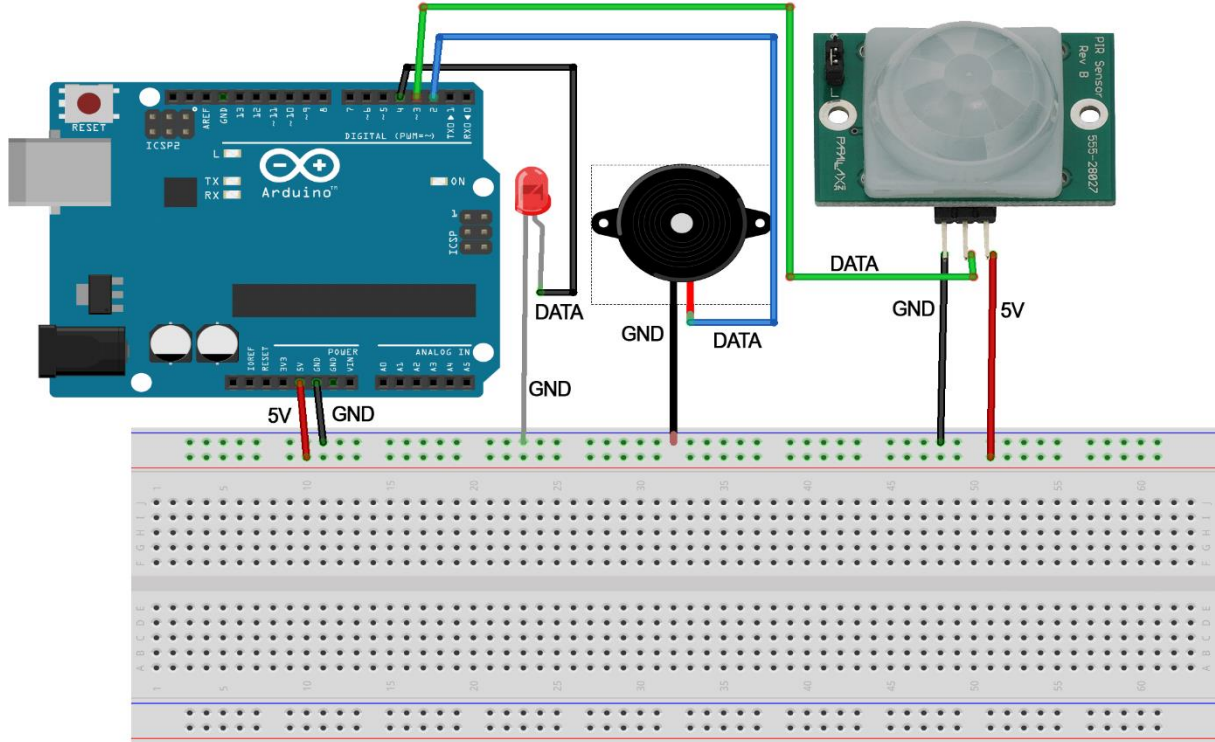
### **DENEYİN ADI: Arduino Mega ile PIR Sensörü ve Buzzer Kullanımı**

**DENEYİN ÖZET VE AMACI:** Bu deney, Arduino Mega ile, PIR sensörü kullanılarak hareket eden nesne(ler)i algılayarak, ardından buzzer aktivasyonunu sağlayacak bir uygulamadır.

### **DENEY MALZEMELERİ (ARAÇ GEREÇLER):**


- 1 Adet Arduino Mega
- 1 Adet PIR sensörü
- 1 Adet Buzzer
- 7 Adet Erkek-Dişi jumper kablo
- 1 Adet Breadboard
- 1 Adet LED
- 1 Adet Arduino PC bağlantı kablosu

### **DEVRE ŞEMASI:**



Şekil-1: Arduino ve sensörlerin bağlantı şeması

### İŞLEM ADIMLARI:

1. Programlamaya başlamak için Arduino, Buzzer ve PIR sensörü bağlantıları Şekil 1' de verilen şemadaki gibi Breadboard üzerinden yapılır. Güç pinleri, Breadboard'ın ilgili yerlerine (5V-GND pin dizilerine) bağlayınız.
2. Sensörlerin giriş çıkış pinlerini ve sinyal pinlerini ise Arduino üzerindeki ilgili yerlere (2-3-4. pinlere) bağlayınız.
3. Harici LED bağlantısına gerek yoktur. Arduino Mega içerisinde LED bulunmaktadır. Ekstra LED bağlantısı yapılmak istenirse kullanılabilir. Bunun için 1 adet LED'i, Arduino'nun 5V'luk güç ve GND pinlerine bağlayınız.
4. Şekil 1'de görülen şematik durumu elde ediniz.
5. Arduino Mega mikrodenetleyici kartını, bilgisayara bağlantı kablosu ile bağlayınız. PC ile bağlantısı yapıldıktan sonra, Arduino Masaüstü Uygulaması  simgesine tıklayarak Arduino programını çalıştırınız.
6. Açılmış olan Arduino arayüz penceresinde, aşağıda verilen kodları giriniz.

### Kodlar:

```
int pirPin = 3; // PIR pin
int ledPin = 13; // LED pin
int buzPin = 4; // Buzzer pin
int deger = 0; // PIR sensörden okunacak dijital deger

void setup() {
  pinMode(pirPin, INPUT); // PIR Pin'i giriş olarak ayarlandı
  pinMode(buzPin, OUTPUT); // BUZZER Pin'i çıkış olarak ayarlandı
  Serial.begin(9600); //Serial Port ekranı baslatıldı
}

void loop(){
```

```

deger = digitalRead(pirPin); // PIR sensöründen gelen dijital deger okunuyor
Serial.println(deger); // Okunan deger seri porttan okunuyor.
if (deger == HIGH) {
digitalWrite(ledPin, HIGH); // Eger okunan deger 1 ise LED yakiliyor.
digitalWrite(buzPin, HIGH); // Eger okunan deger 1 ise BUZZER ses veriyor.
}
else{
digitalWrite(ledPin,LOW); // Eger okunan deger 0 ise LED söndürülüyor.
digitalWrite(buzPin, LOW); // Eger okunan deger 0 ise BUZZER susuyor.
}
}
}

```

## 7. Arduino arayüz penceresindeki görünüm, Şekil 2'deki gibidir.



```

sketch_may02c | Arduino 1.8.5
Dosya Düzenle Taslak Araçlar Yardım

sketch_may02c $

int pirPin = 3; // PIR pin
int ledPin = 13; // LED pin
int buzPin = 4; // Buzzer pin
int deger = 0; // PIR sensörden okunacak dijital deger

void setup() {
pinMode(pirPin, INPUT); // PIR Pin'i giriş olarak ayarlandı
pinMode(ledPin, OUTPUT); // LED Pin'i çıkış olarak ayarlandı
pinMode(buzPin, OUTPUT); // BUZZER Pin'i çıkış olarak ayarlandı
Serial.begin(9600); //Serial Port ekranı baslatıldı
}

void loop(){
deger = digitalRead(pirPin); // PIR sensöründen gelen dijital deger okunuyor
Serial.println(deger); // Okunan deger seri porttan okunuyor.
if (deger == HIGH) {
digitalWrite(ledPin, HIGH); // Eger okunan deger 1 ise LED yakiliyor.
digitalWrite(buzPin, HIGH); // Eger okunan deger 1 ise BUZZER ses veriyor.
}
else{
digitalWrite(ledPin,LOW); // Eger okunan deger 0 ise LED söndürülüyor.
digitalWrite(buzPin, LOW); // Eger okunan deger 0 ise BUZZER susuyor.
}
}
}

```

Şekil-2: Arduino arayüzündeki kodlar

Şekil 2'deki kodların yaptırdığı işlevler şu şekildedir;

İlk olarak, sensörün ve Buzzer'ın bağlı olduğu pinler tanımlanır. Arduino Mega 13. pinine bağlı olan bir LED taşımaktadır. Bu LED'in de tanımı yapılır.

2. Adımda (Setup fonksiyonu) tanımlanan pinlerin giriş ve çıkış pinleri olarak ayarlamaları yapılır.

3. Adımda döngü içerisinde (Loop fonksiyonu) PIR sensöründen gelen dijital değer okunur. Dijital değer seri port içerisinde okunması sağlanır.

Eğer, okunan değer 1 ise, BUZZER ses verir ve LED yanar.

Eğer okunan değer 0 ise BUZZER susar ve LED söner.

8. Kodları derlemek için, Arduino arayüz penceresindeki yükle düğmesine tıklayınız.

9. Kodların derlenmesi sonucunda, hareketi algılayan bir PIR sensörünün çalışması gözlemlenmelidir. Bu gözlem ile birlikte PIR sensörünün algı alanında bir hareketlilik söz konusu ise, bağlanmış olduğumuz Buzzer ve LED aktifleşir. Aktifleşen Buzzer'dan ses çıkışı, LED'den ise ışık alınmış olur.

## UYGULAMA GÖZLEM VE SONUÇLARI

Bu çalışmadaki gözlemlerinizi aşağıda boş bırakılan satırlara yazınız.

---

---

## ÇALIŞMA SORULARI