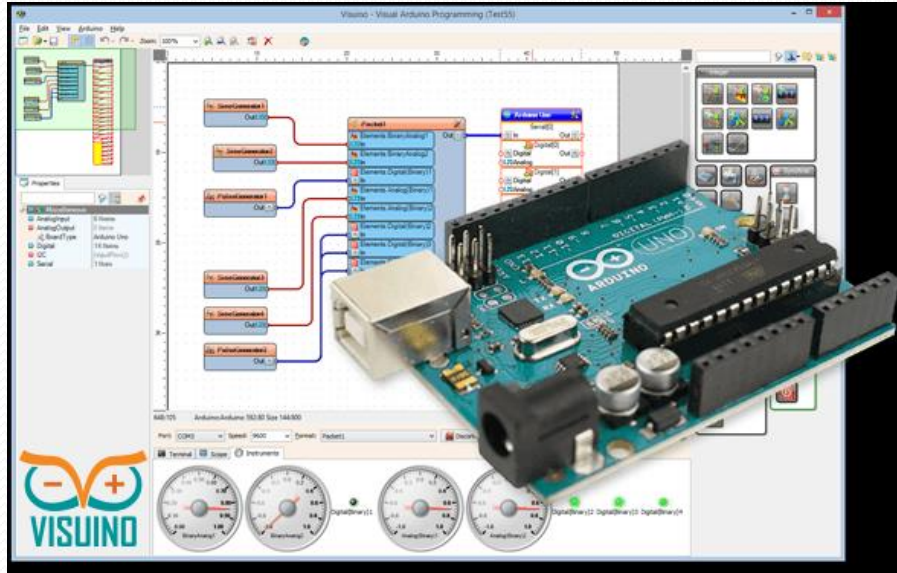




**ARDUINO**  
**OPEN-SOURCE**  
**COMMUNITY**

# Arduino nedir?



Arduino donanım ve yazılımın kolayca kullanılmasına dayalı bir açık kaynak elektronik platformdur.

# Açık kaynak nedir?

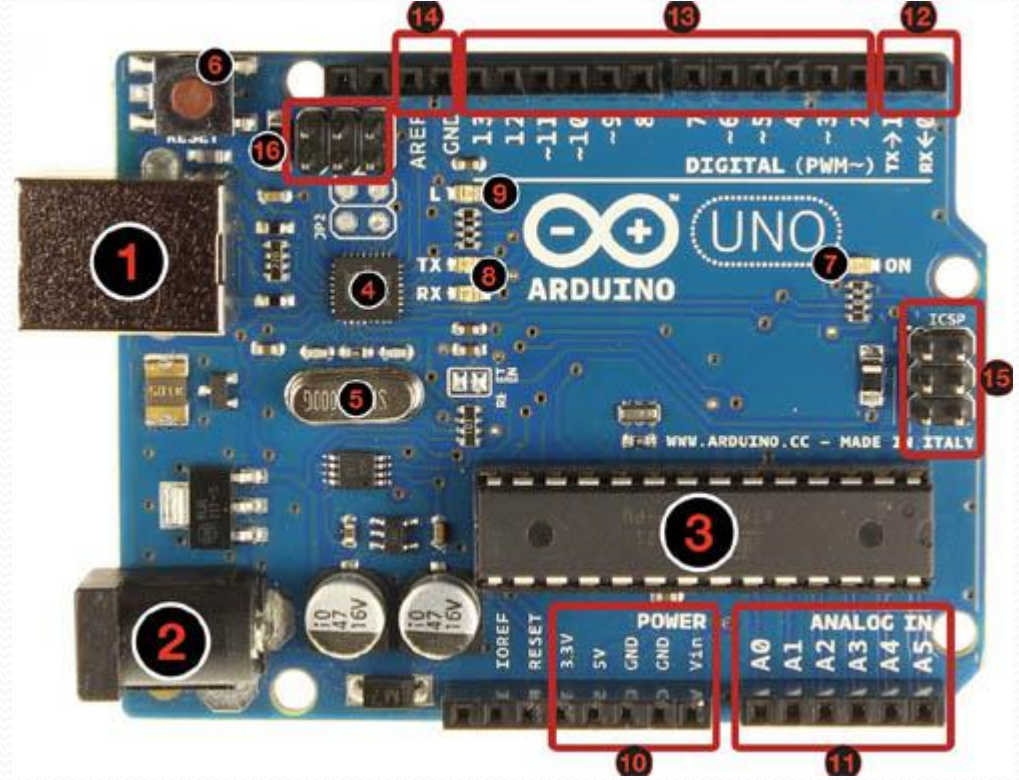
**Açık kaynak**, bir bilgisayar yazılımının makina diline dönüştürülüp kullanımından önceki, programcılar tarafından okunur, anlaşılır, yeni amaçlara uygun değiştirilebilir hâlinin gizli tutulmayıp açık, yani okunabilir hâlde kamuya paylaşılıyor olmasına verilen isimdir.



Arduino aynı zamanda İtalyancada dost anlamına gelmektedir.

# Arduino Uno Kartını Tanıyalım

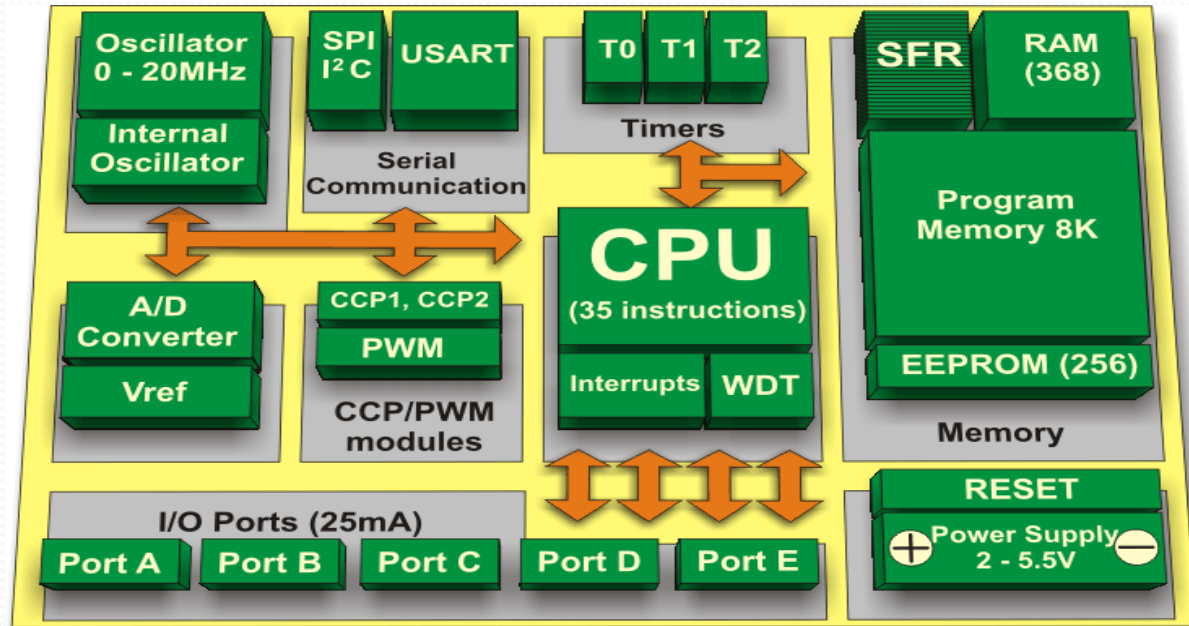
- 1 : USB jakı
- 2 : Power jakı (7-12 V DC)
- 3 : Mikrodenetleyici ATmega328
- 4 : Haberleşme çipi
- 5 : 16 MHz kristal
- 6 : Reset butonu
- 7 : Power ledi
- 8 : TX / RX ledleri
- 9 : Led
- 10 : Power pinleri
- 11 : Analog girişler
- 12 : TX / RX pinleri
- 13 : Dijital giriş / çıkış pinleri (yanında ~ işareti olan pinler PWM çıkışı olarak kullanılabilir.)
- 14 : Ground ve AREF pinleri
- 15 : ATmega328 için ICSP
- 16 : USB arayüzü için ICSP





# Mikrodenetleyici nedir?

- Yazılım ve donanımın kesiştiği noktadır arduino kartlarımızda.
- Arduino kartları genellikle atmel avr mikrodenetleyicilerini kullanmaktadır.



# Arduino Uno Kartının Özellikleri

- ◆ Mikrodenetleyici: ATmega328
- ◆ Besleme Voltajı: 7-12V
- ◆ Çalışma Voltajı: 5V (Arduino UNO kaç volta çalışır?)
- ◆ Giriş-Çıkış Pinleri Sayısı: 14 adet dijital pini mevcuttur. Bunlardan 6 tanesi PWM olarak kullanılabilir. 6 pinden 8 bitlik analog sinyal çıkışı elde edilebilmektedir.
- ◆ Pinlerdeki Akım: 40mA(5V), 50mA(3.3V)
- ◆ FLASH: 32 KB (2 KB Bootloader tarafından kullanılıyor)  
1 KB EEPROM  
2 KB SRAM
- ◆ Çalışma Saat Hızı: 16MHz
- ◆ İletişimi de USB üzerinden sağlayabilmektedir.





# Arduino Nano Kartının Özellikleri

Mikrodenetleyici : Arduino Nano V3 te ATmega328  
(önceki versiyonlarda ATmega168)

Çalışma gerilimi : +5 V DC

Tavsiye edilen besleme gerilimi : 7 - 12 V DC

Besleme gerilimi limitleri : 6 - 20 V

Dijital giriş / çıkış pinleri : 14 tane (6 tanesi PWM çıkışını destekler)

Analog giriş pinleri : 8 tane

Giriş / çıkış pini başına düşen DC akım : 40 mA

Flash hafıza : ATmega328 için 32 KB, ATmega168 için 16 KB (2 KB bootloader için kullanılır)

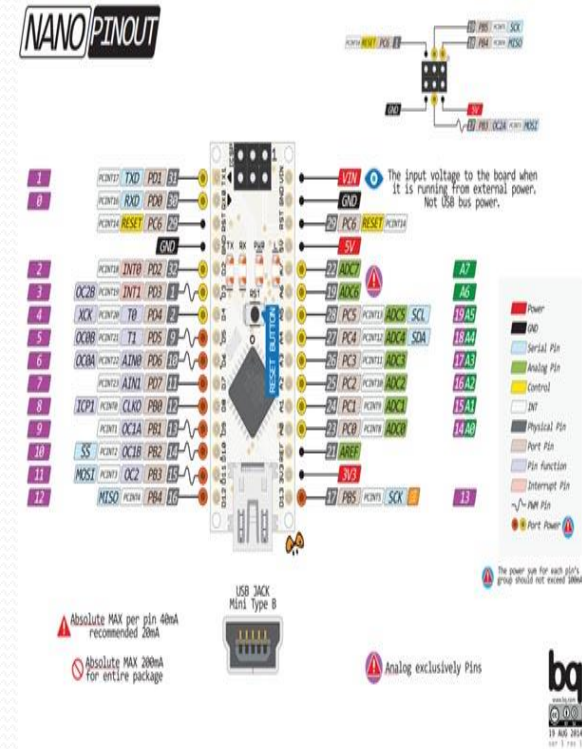
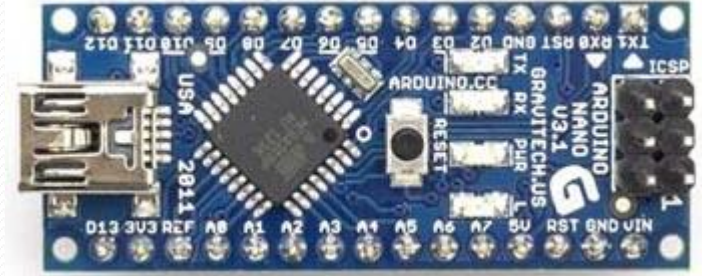
SRAM : ATmega328 için 2 KB, ATmega168 için 1 K

EEPROM : ATmega328 için 1 KB, ATmega168 için 512 byte

Saat frekansı : 16 MHz

Boyutları : 18 mm x 45 mm

Ağırlık : 5 g



# Peki Projem İin Hangi Kart Uygun ?

1-Projenin amacı belirlenmeli, bu amalar dođrultusunda kullanılacak evre birimlerin belirlenmeli.

2-Proje iin büte durumu belirlenmeli.

Yanıtlar dođrultusunda uygun geliřtirme kartı seilmelidir.

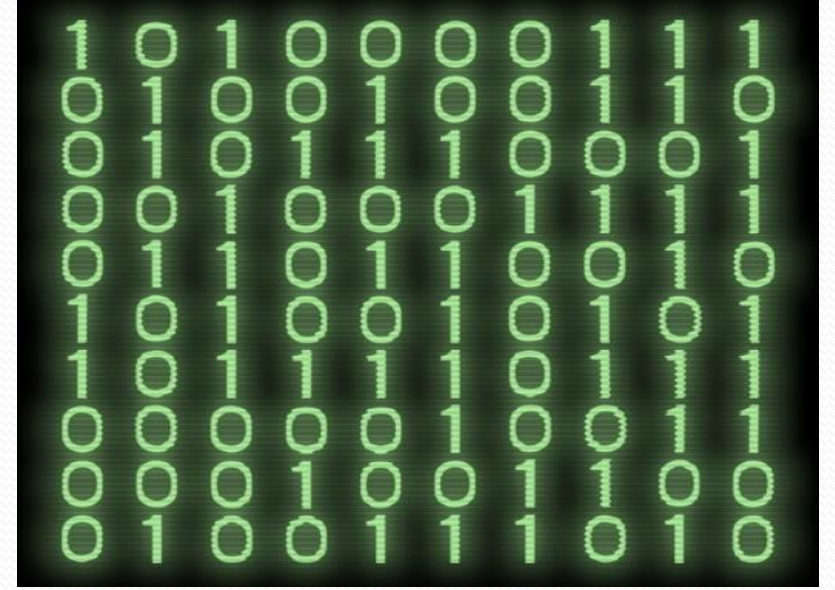
# Arduino IDE sini nasıl yüklerim ?

VIDEO VAR

# Arduino için kaynaklar

- Arduino - Coşkun TAŞDEMİR
- YOUTUBE === >>>>>LEZZETLİ ROBOT TARİFLERİ

İNTERNET



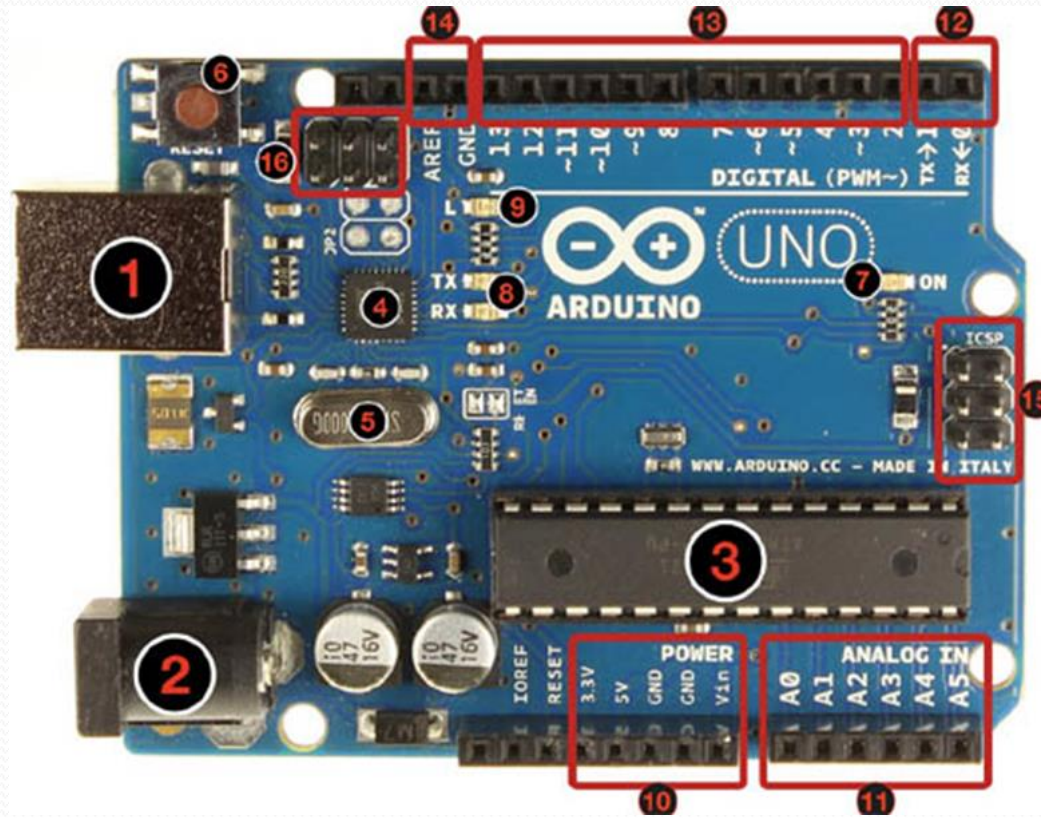
# ARDUINO İLE DİJİTAL GİRİŞ - ÇIKIŞ İŞLEMLERİ

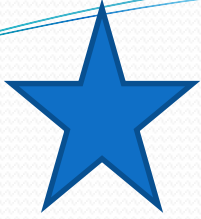


# Arduino uno ile GPIO

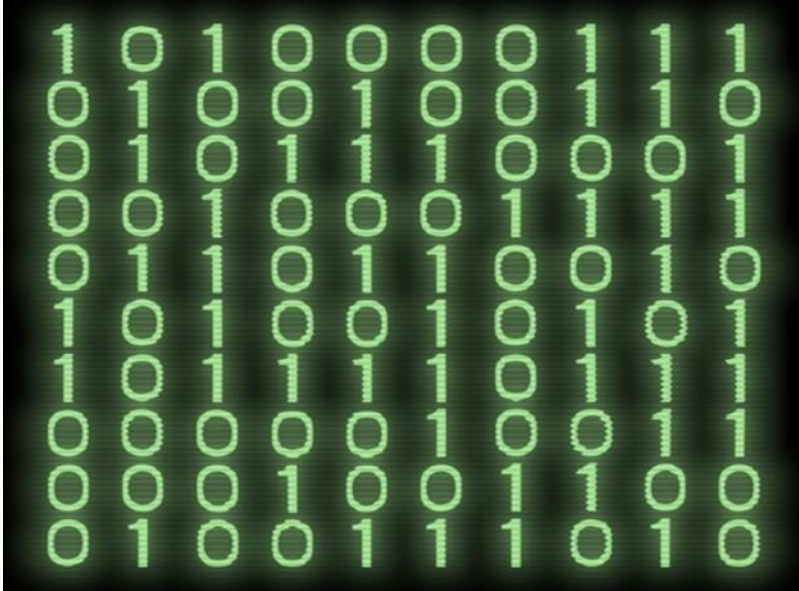
- Dijital girş-i-çıkıř fonksiyonları Arduino ile harici bir dijital birime 1 veya 0 deęerini göndermemizi veya dışarıdan gelen deęerleri okuyabilmemizi sağlar.
- Örnek bir butona basılıp basılmadığını, ledi yakıp söndürmek vb işlemler gibi

# Arduino uno kartında nerede bu pinler?





Arduino'ya hangi pinlerinin ne amaçla kullanılacağı EN BAŞTA BELİRTİLMELİDİR. AKSİ TAKTİRDE KOD ÇALIŞMAZ YADA İSTENİLEN PERFORMANS ALINAMAZ



Arduino Uno Kartında

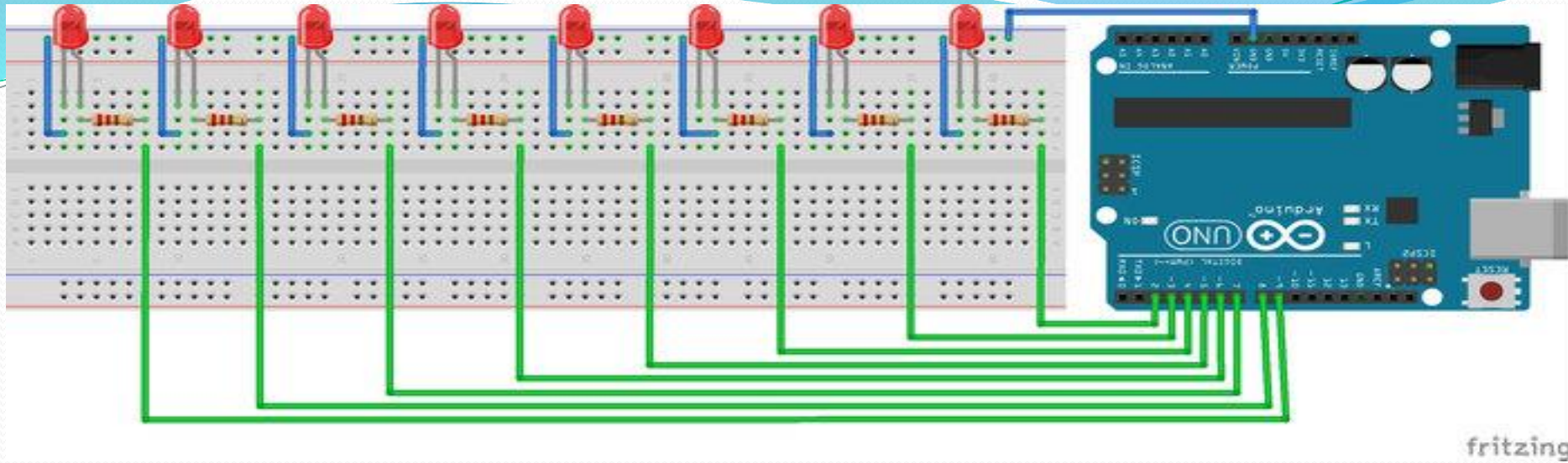
Dijital 1 = + 5 VOLT

Dijital 0 = 0 VOLT

Pinlerdeki Akım: 40mA(5V),  
50mA(3.3V)

# KARA ŐİMŐEK PROJESİ



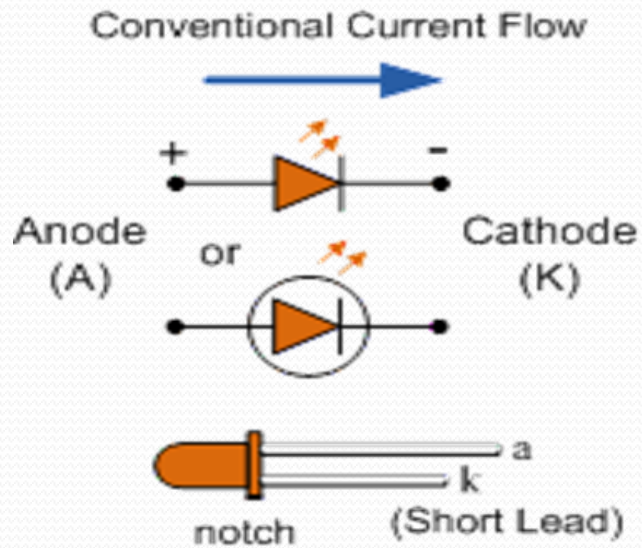


Arduino Uno Kartında  
Dijital 1 = + 5 VOLT  
Dijital 0 = 0 VOLT

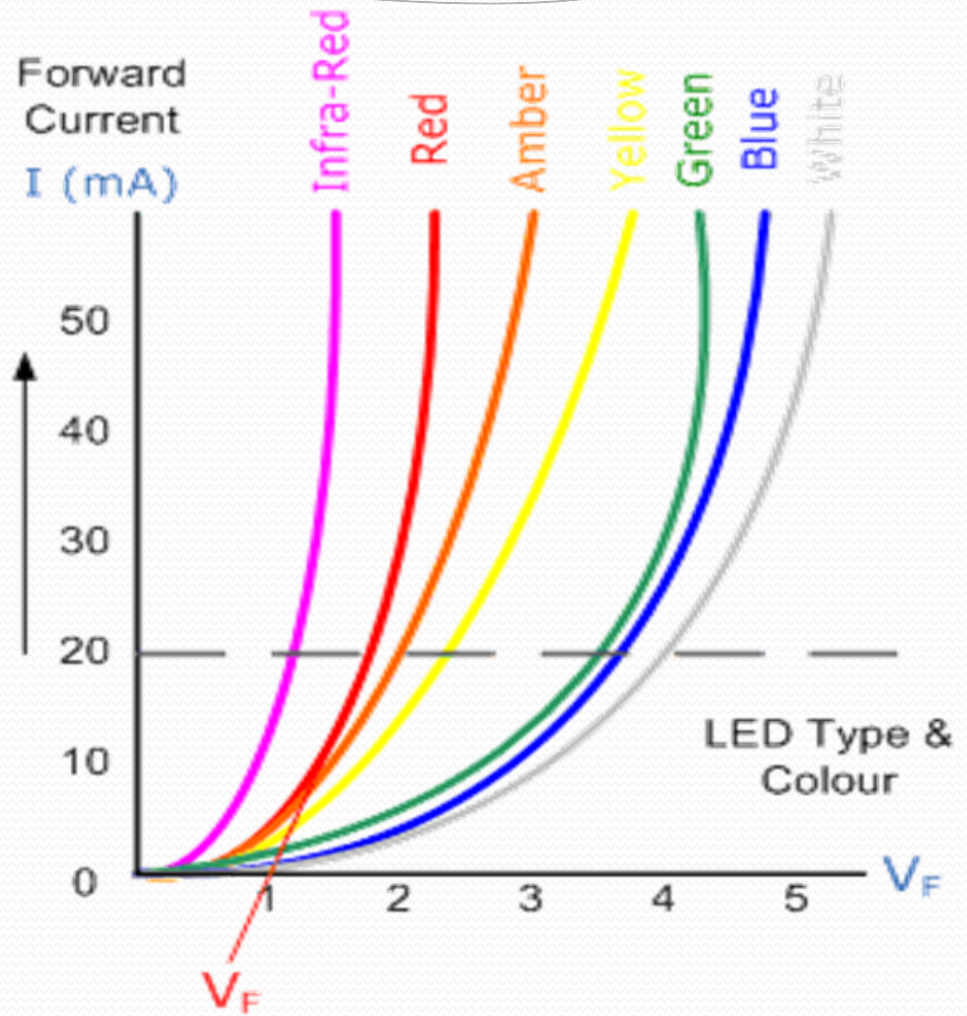
Pinlerdeki Akım: 40mA(5V),  
50mA(3.3V)

PEKİ NEDEN DİRENÇ KULLANDIK ???





LED and its I-V Characteristics



# Kodun incelenmesi

```
const int LEDdizisi[] = {2,3,4,5,6,7,8,9};
```

```
void setup () { /* void setup kısmı arduino kartımızın temel kurulum işlemlerini yaptığımız yerdir yani ilk başta arduino kartımız başlarken hangi ayarları yapması gerekiyorsa o ayarları yaptığımız yerdir !!! void setup her çalışmada sadece 1 kez çalışır ve biter.*/
```

```
  for(int i=0; i<8 ;i++)
  { /* For dongusuyle LEDdizisi elemanlarına ulasiyoruz */
    pinMode(LEDdizisi[i], OUTPUT); /* LED pinleri cikis olarak ayarlandi */
  }
  Serial.begin(9600);/*PC ve Arduino arası SERİ HABERLEŞMEYİ AYARLAR*/
}
```

```
void loop() {/*
  void loop bölümü arduino kartımızın çalışıp void setup ayarlarını yaptıktan sonra SÜREKLİ OLARAK ÇALIŞTIĞI YERDİR WHILE(1) komutu ile AYNI MANTIKLA ÇALIŞIR.
```

```
  */
  for(int i=0; i<8; i++){ /* Tum LEDleri sirayla 50 milisaniye yakip sonduruyoruz */
    digitalWrite(LEDdizisi[i],HIGH);
    delay(50);
    digitalWrite(LEDdizisi[i],LOW);
  }
```

```
  for(int j=7;j>-1; j--)
  { /* LEDleri geri yonde 50 milisaniye yakip sonduruyoruz */
    digitalWrite(LEDdizisi[j],HIGH);
    delay(50);
    digitalWrite(LEDdizisi[j], LOW);
  }
}
```

# Arduino giriş çıkış fonksiyonlarının incelenmesi

- pinMode(PİN NUMARASI,HANGİ DURUMDA KULLANACAĞIZ)

pinMode(3,OUTPUT) yani; 3 numaralı pin 0/1 olarak kullanılacak

pinMode(5,INPUT) yani 5 numaralı pin giriş olarak kullanılacak

- `digitalWrite(pin NUMARASI, PINİN DURUMU )`

`HIGH === DİJİTAL 1 === 5V`

`LOW=== DİJİTAL 0 === 0 V`

`digitalWrite(3,HIGH)`

3 numaralı pin 5 v olsun

`digitalWrite(4,LOW)`

4 numralı pin 0 volt olsun

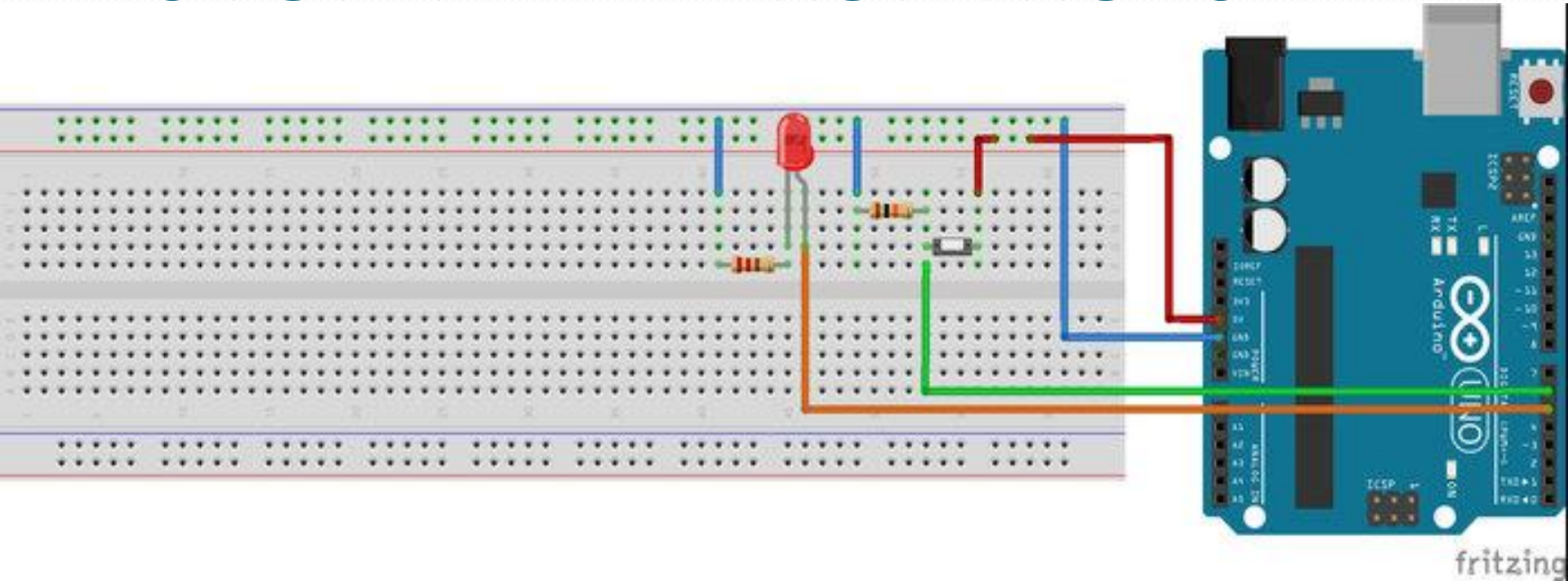
Arduino üzerinde bekleme süresi vermek  
kodu

-----

`delay(MİLİSANİYE MİKTARI)`

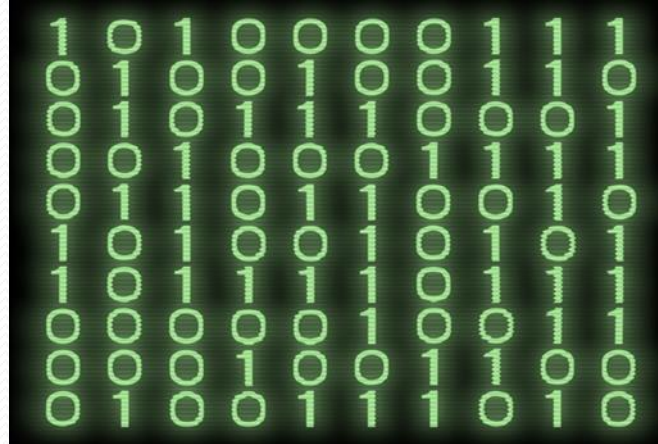


# BUTON İLE LED KONTROLÜ

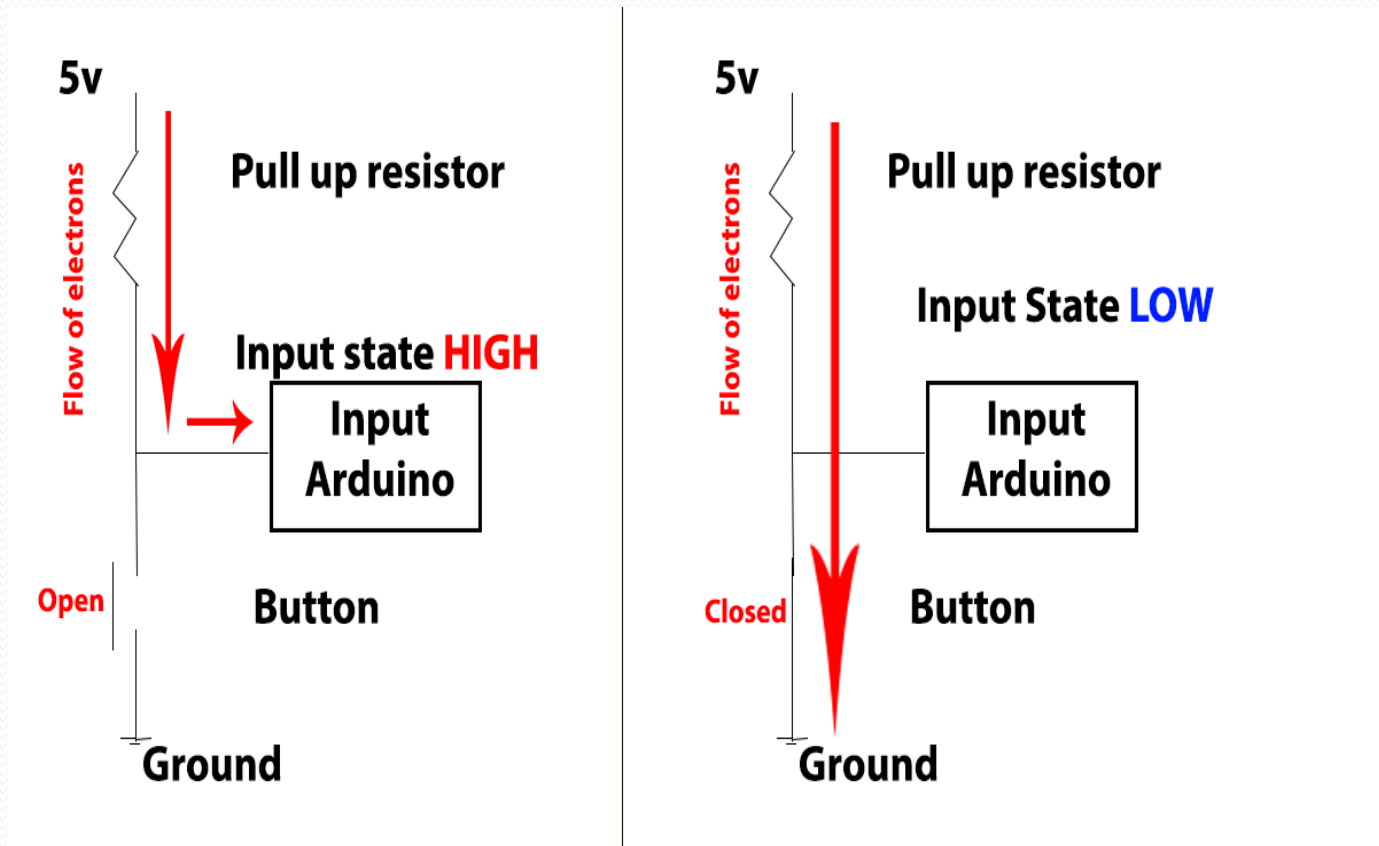


Kullanıcıyla etkileşim halinde olan birçok projede düğme kullanılmaktadır. Düğme, arasında az bir boşluk bulunan iki tel gibi düşünülebilir. Kullanıcı düğmeye bastığında bu boşluk kapanır ve düğme iletken duruma geçer, üzerinden akım akar. Kullanıcı düğmeden elini çektiğinde devrenin eski konumuna dönmesi için, pull up ve pull down denilen direnç bağlantıları kullanılır.

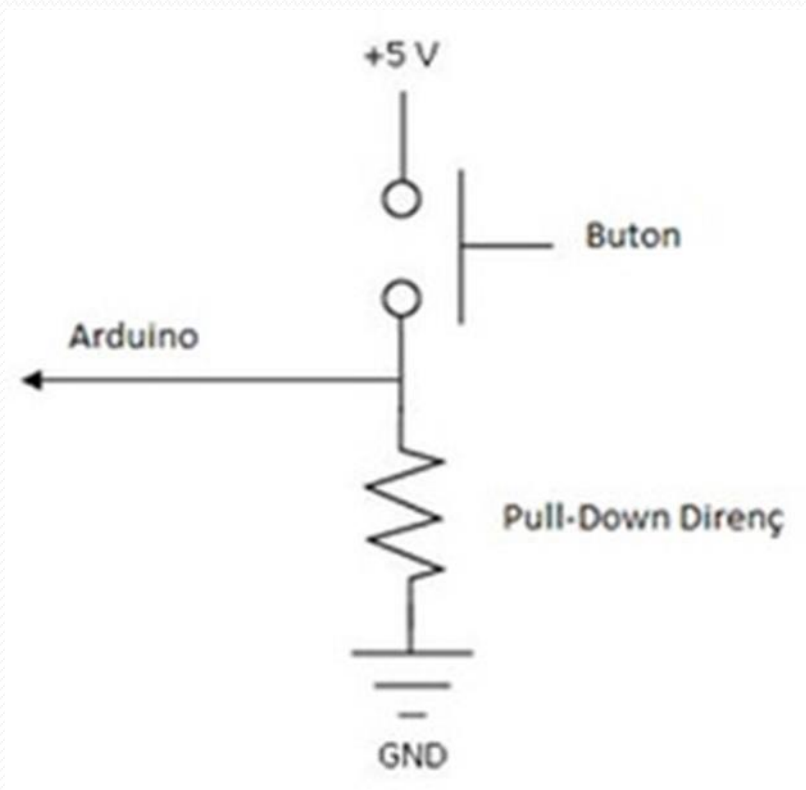
Pull up ve pull down direnç ismi değil, dirençlerin bağlanma şeklidir. Genellikle 10K ohm direnç kullanılır.



# PULL UP DİRENCİ



# PULL DOWN DİRENCİ



- `const int Dugme = 6; /* düğmenin bağlı olduğu pin */`
- `const int LED = 5; /* LEDin bağlı olduğu pin */`
- `int dugmeDurumu = 0; /* düğmenin durumu */`
- `int LEDDurumu = 0; /* birinci yöntem için LED durumu */`
- `void setup() {`
- `pinMode(LED, OUTPUT); /* LED pini çıkış olarak ayarlandı */`
- `pinMode(Dugme, INPUT); /* düğme pini giriş olarak ayarlandı */`
- `}`
- `void loop(){`
- `dugmeDurumu = digitalRead(Dugme); /* düğmenin durumu okundu ve değişkene aktarıldı */`
- `if(dugmeDurumu == HIGH) { /* düğmeye basılmış ise */`
- `delay(10); /* dalgalanmalar için biraz bekleyelim */`
- `if(LEDDurumu == 0){ /* LED yanmıyorsa */`
- `digitalWrite(LED, HIGH); /* LEDi yak */`
- `LEDDurumu = 1;`
- `}else { /* LED yanıyorsa */`
- `digitalWrite(LED, LOW); /* LEDi sondur */`
- `LEDDurumu = 0;`
- `}`
- `}`
- `while(dugmeDurumu == HIGH){ /* düğmeye basılı olduğu sürece bekle */`
- `dugmeDurumu = digitalRead(Dugme); /* düğmenin durumunu kontrol et */`
- `}`
- `delay(10); /* dalgalanmalar için biraz bekleyelim */`
- `}`
- `}`



digitalRead(DİJİTAL GİRİŞİN YAPILMASI İSTENEN  
PİN)

```
if(digitalRead(3)==HIGH)
```



**TEŐEKKÜRLER**