

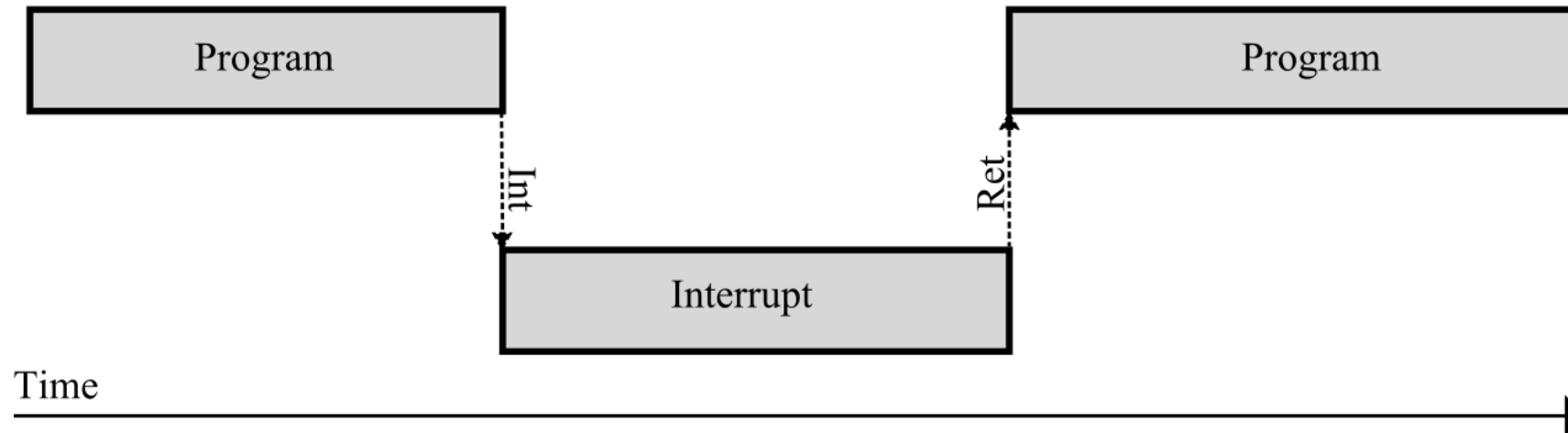


Donanimsal Kesmeler (Interrupts)



Kesme (Interrupt) Nedir?

Kesmeler, herhangi bir kontrolü sürekli ve gecikme olmaksızın takip edebilmek, ve söz konusu durum olduğu anda reaksiyon gösterebilmek için kullanılan bir donanımsal takip fonksiyonudur.





Kesme (Interrupt) Nedir?

Kod içerisinde bir pinin LOW veya HIGH olma durumunu (*bir butona basılması veya bir sensörden çıkış alınması gibi*) veya belirli bir değişkenin değerinin değişimini takip etmek için LOOP fonksiyonu içinde çeşitli mantıksal kontroller yapıyoruz.

Fakat takip ettiğimiz durum oluşsa bile eğer kod koştururken işaretçi daha kontrolün yapıldığı satıra ulaşmadıysa kontrolümüz gerçekleşmez, dahası belki de kontrol satırına gelindiğinde durum ortadan kalkmış olur ve yakalayamayız.



Fakat buraya her döngüde uğraması belki de 2-3 sn'de bir gerçekleşecek ancak bu zaman zarfında aranan durum ortadan kalkabilecektir.

.....



Neden İhtiyaç Duyarız?

```
void loop(){
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
if (a ==1){
```

```
  kontrol
```

```
  }
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```



- Buton kullanırken zaman zaman butona basıldığının kod tarafından anlaşılmaması,
- Sensörlerden tetikleme sinyallerini takip ederken durum gerçekleşmesine rağmen tetiklemenin alınamaması,
- Zamana karşı hassas ve belli aralıklarla yapılması gereken işlemlerde gecikme ve sarkma yaşanması gibi durumlarda kullanılabilir.



Kesme Türleri

- Kesmeler,

- 1) Donanımsal Kesmeler

- 2) Zaman Kesmeleri

olarak ikiye ayrılır.

- 1) **Donanımsal Kesmeler** pinlerin,

- ✓ RISING (LOW'dan HIGH'a çekilme)

- ✓ FALLING (HIGH'dan LOW'a çekilme)

- ✓ CHANGE (Herhangi bir durumdan diğerine geçme)

durumlarında tetiklenir. (ESP32 için ilave 2 durum daha mevcuttur. LOW veya HIGH olduğu sürece de çalışsın (değişim olmadığında) şeklinde kullanılabilir.)

- 2) **Zaman Kesmeleri** ise,

işlemcinin saat frekansından faydalanılarak belirli periyotta kontrolün gerçekleştirilmesiyle çalıştırılır. (Zaman kesmelerini başka bir çalışmada ele alacağız.)



Kesmenin Çalışacağı Pinler

ESP-32

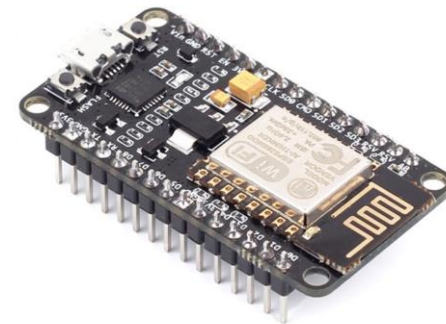
- GPIO-6
- GPIO-7
- GPIO-8
- GPIO-9
- GPIO-10
- GPIO-11

Pinleri haricinde tüm pinler kesme olarak kullanılabilir.



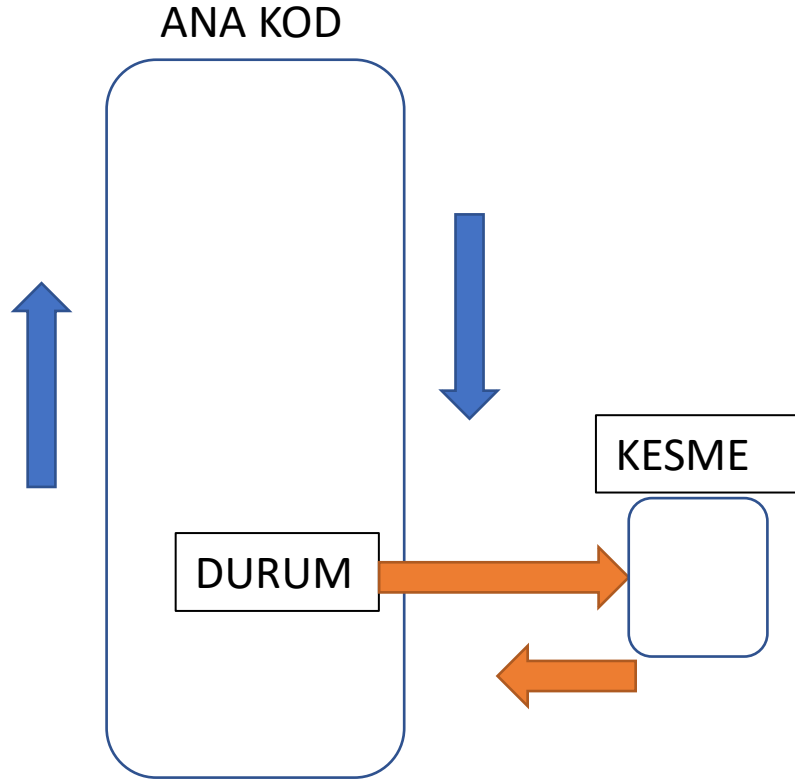
ESP-8266

- GPIO-16
pini haricinde tüm pinler kesme olarak kullanılabilir.





Çalışma Mantığı



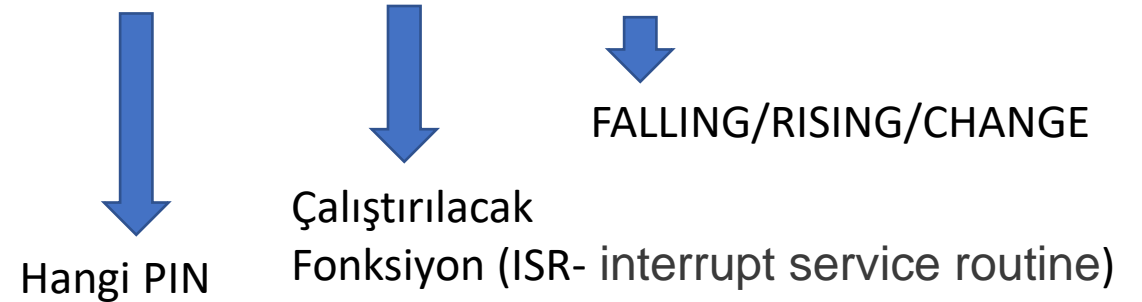
- Sistem, donanımsal olarak, tanımlanan Kesme fonksiyonunu sürekli takip eder ve tetikleme olduğunda Ana Kodu durdurarak kesme fonksiyonuna geçer, gereken işlemleri yapar ve tekrar ana koda döner.
- Bu nedenle kesme kodu içeriği basit ve küçük tutulmalıdır.
- Doğrudan burada işlem yapmak yerine değişkenler BAYRAK olarak kullanılarak ana koda mantıksal işaretler gönderilebilir.



Nasıl Kullanılır?

1. Kodda SETUP bloğu içerisinde aşağıdaki tanımlama yapılır:

```
attachInterrupt(digitalPinToInterrupt(GPIO), function, mode);
```



2. Kesme gerçekleştiğinde yapılacak işler fonksiyonu tanımlanır:

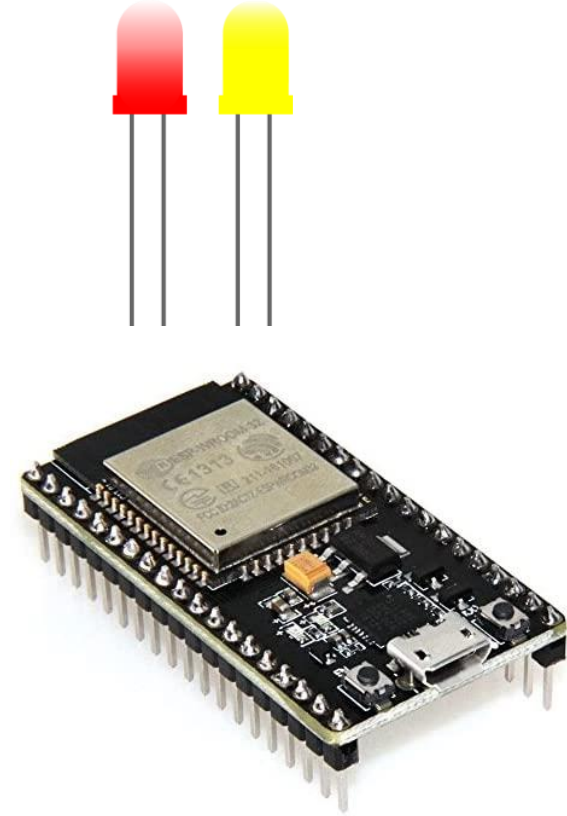
```
void IRAM_ATTR hareketSENSOR() {  
    Serial.println("Hareket Algılandı");  
    digitalWrite(led, HIGH);  
}
```

DİKKAT: fonksiyonu tanımlarken void ibaresinden sonra «IRAM_ATTR» ibaresi kullanılmalıdır.



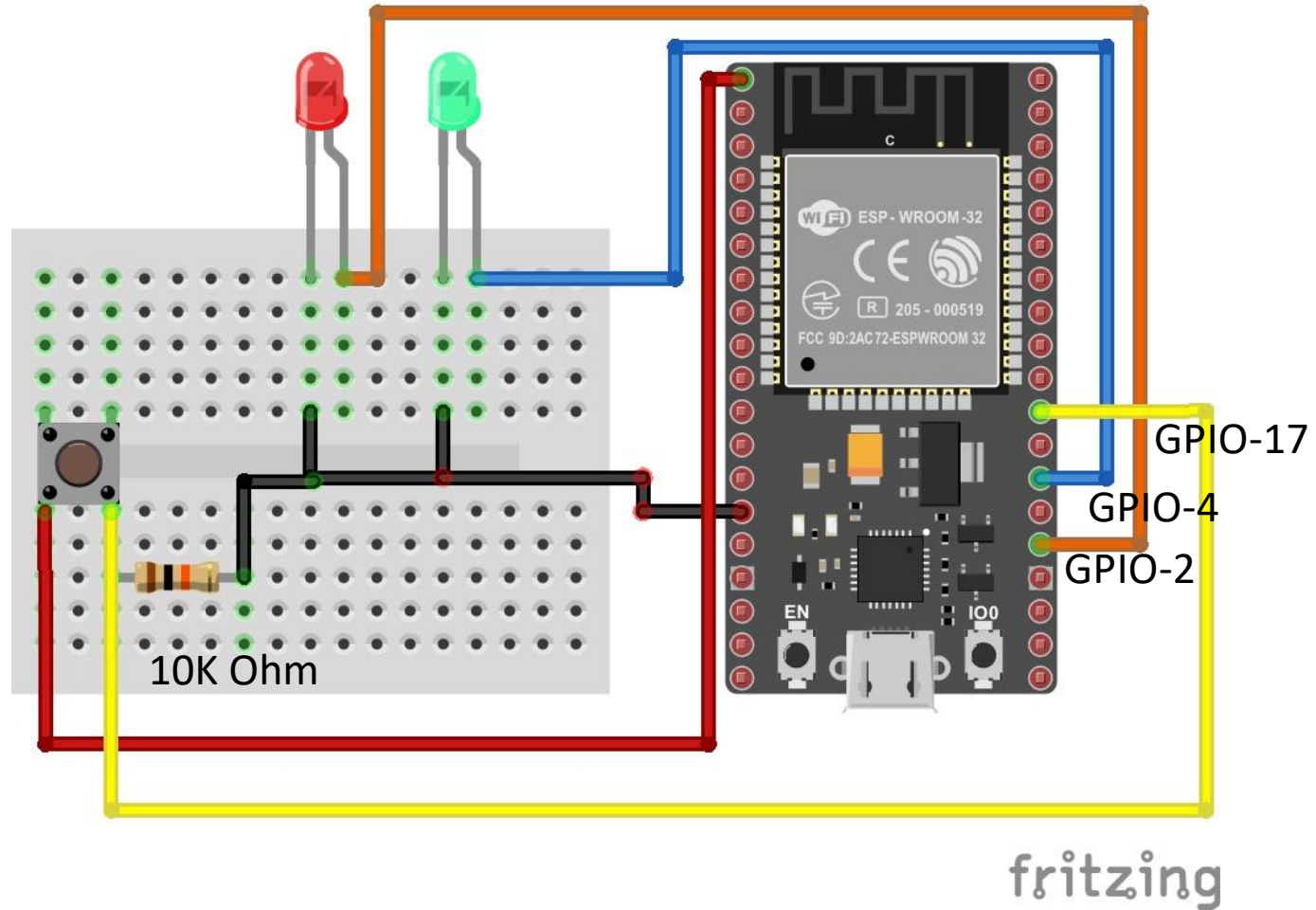
Örnek Uygulama

- Bu örnekte sabit aralıkla yanıp sönen bir LED devresinde bir butona basarak kesmeyi devreye sokacağız ve söz konusu LED'i söndürüp başka bir LED'in yanmasını sağlayacağız.
- Hareket sensörü kullanarak çalışan ve hareket algıladığı durumda bir işlem yapan başka bir örneği de siz yapabilirsiniz.





Devre Şeması





Kaynaklar

- Google 😊
- <https://www.espressif.com/en/products/software/>
- randomnerdtutorials.com
- Lastminuteengineers.com