



**SAKARYA ÜNİVERSİTESİ**  
**Bilgisayar ve Bilişim Bilimleri Fakültesi**  
**Bilgisayar Mühendisliği Bölümü**

# **Mikroişlemcili Sistemler ve Laboratuvarı**

- ❑ **Yoklama:** Kontrol edilmek istenen giriş/çıkış ucu ya da bayrak, yazılım yardımıyla belli zaman aralıklarında sürekli denetlenir.
- ❑ **Kesme:** Mikrodenetleyicinin donanımsal olarak denetimde bulunmasıdır.

## ❑ Sürekli denetim (**yoklama**) işleminde

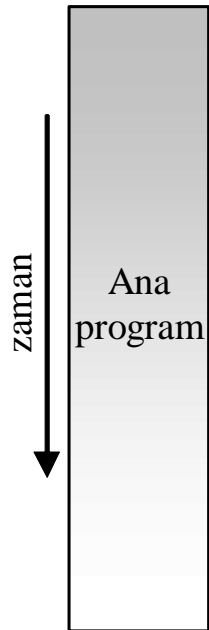
- Mikrodenetleyici, yalnızca bir bayrak ya da giriş/çıkış ucunu kontrol edebilmektedir.
- Dolayısıyla başka bir işlem gerçekleştirememektedir.

## ❑ Kesme yönteminde

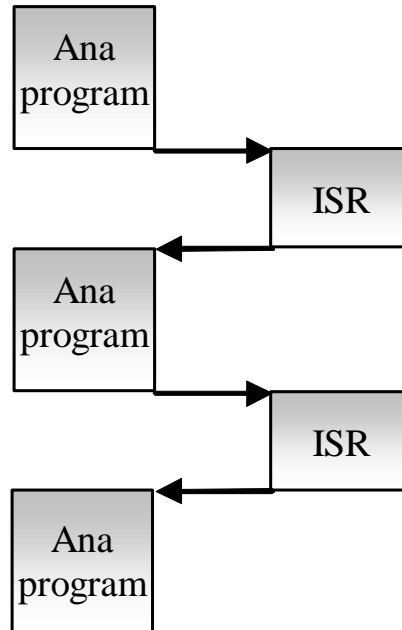
- Mikrodenetleyici normal çalışmasına devam etmektedir.
- Herhangi bir kesme sinyali oluştuğunda o kesmeye ait alt programı icra edip, ana programa (kaldığı yere) geri dönülmektedir.
- Bu sayede mikrodenetleyici aynı zamanda birden fazla işlemi gerçekleştirebilmektedir.

# Kesmelerin Çalışması

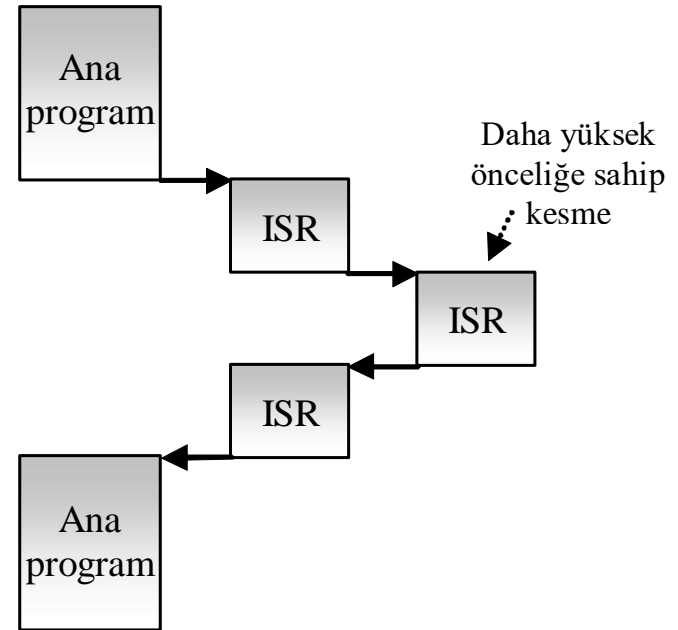
- ❑ Kesme geldiğinde çalıştırılan alt programa “**kesme hizmet programı**” (**ISR-Interrupt Service Routine**) denir.
- ❑ ISR’ler icra ettirildikten sonra ana programda kalınan yere dönülür.



Kesmesiz  
program



Kesmeli program



❑ Standart 8051 mikrodenetleyicisinde 5 adet kesme kaynağı bulunmaktadır.

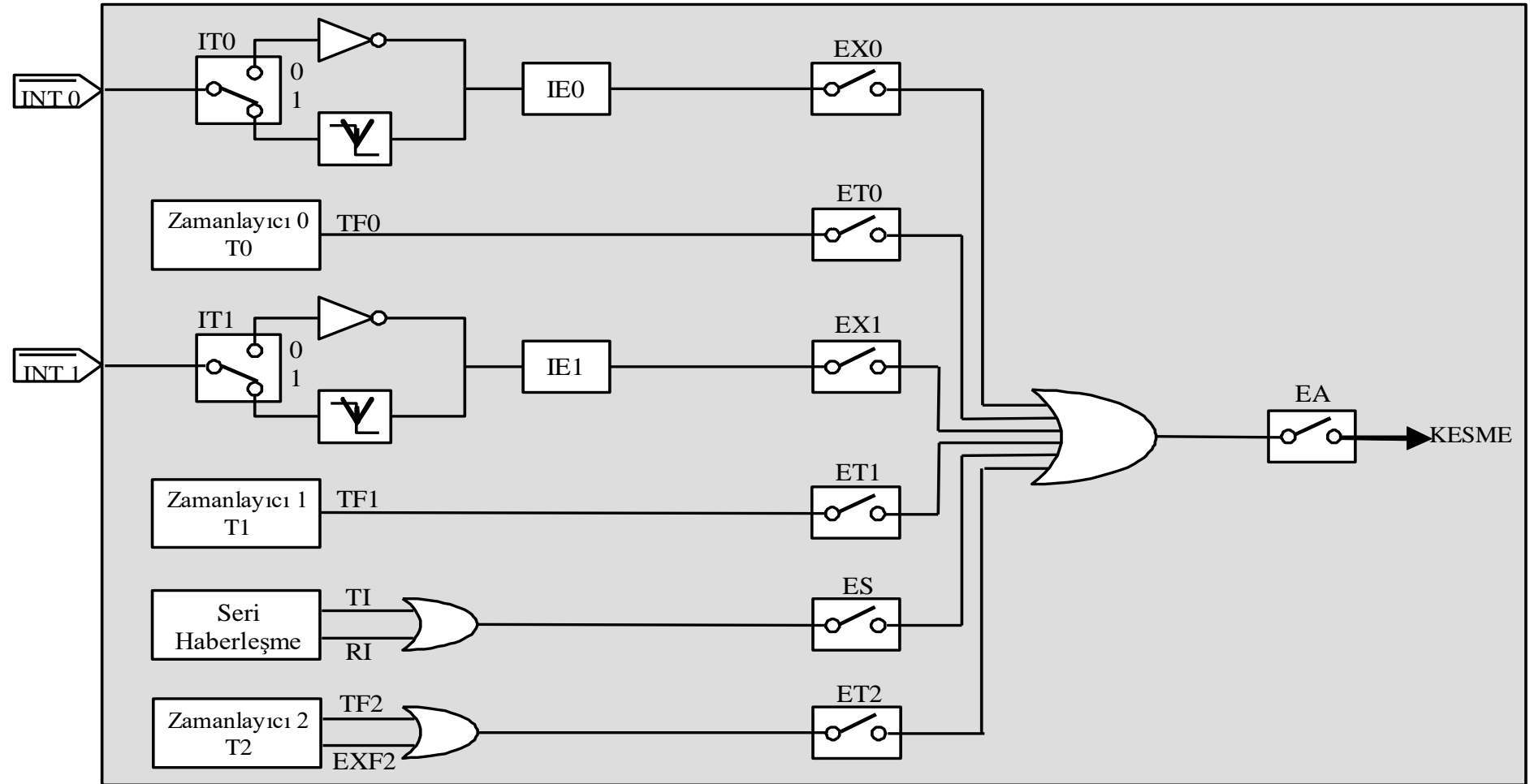
## ❑ Dahili Kesmeler

- Zamanlayıcı/sayıcı 0 (TF0)
- Zamanlayıcı/sayıcı 1 (TF1)
- Seri haberleşme (TI, RI)

## ❑ Harici Kesmeler

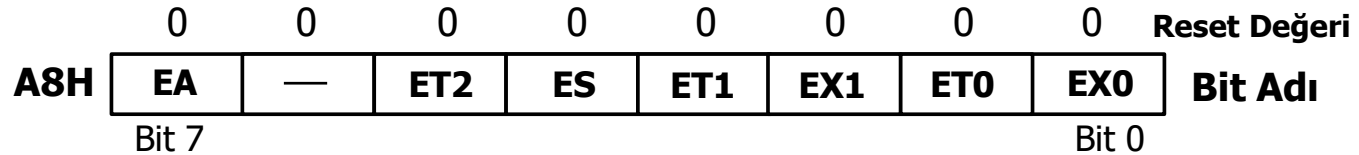
- Harici kesme 0 (INT0)
- Harici kesme 1 (INT1).

# 8051 Kesme Organizasyonu



**IE**

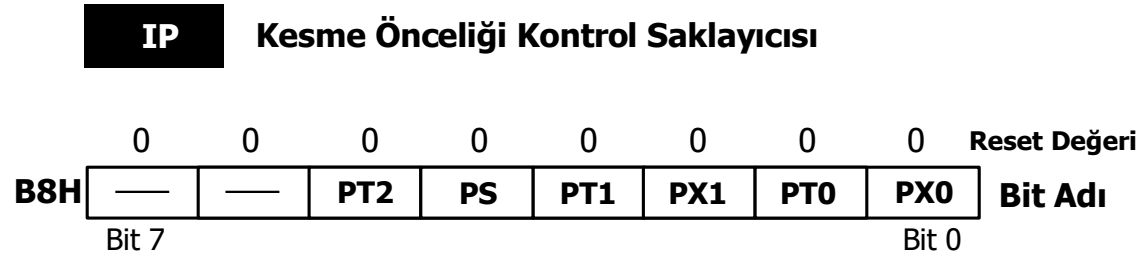
**Kesme Yetkilendirme Saklayıcısı**



Bit No	İsmi	Bit Adresi	Açıklama
IE.0	EX0	A8h	Harici kesme 0 ( $\overline{INT0}$ ) yetkilendirme biti
IE.1	ET0	A9h	Zamanlayıcı/sayıcı 0 (T0) kesme yetkilendirme biti
IE.2	EX1	AAh	Harici kesme 1 ( $\overline{INT1}$ ) yetkilendirme biti
IE.3	ET1	ABh	Zamanlayıcı/sayıcı 1 (T1) kesme yetkilendirme biti
IE.4	ES	ACH	Seri port kesme yetkilendirme biti
IE.5	ET2	ADh	Zamanlayıcı/sayıcı 2 (T2) kesme yetkilendirme biti
IE.6	—	A Eh	Kullanılmıyor
IE.7	EA	AFh	Genel kesme yetkilendirme biti

# Kesmelerin Öncelik Sırasının Belirlenmesi

- ❑ Eş zamanlı olarak birden fazla kesme meydana gelebilir. Bu durumda kesmelere bir öncelik verilmelidir.
- ❑ Kesmelerin öncelikleri **IP** saklayıcısındaki ilgili bitler ile belirlenir.



Bit No	İsmi	Bit Adresi	Açıklama
IP.0	PX0	B8h	Harici kesme 0 ( $\overline{INT0}$ ) öncelik biti
IP.1	PT0	B9h	Zamanlayıcı/sayıcı 0 (T0) kesme öncelik biti
IP.2	PX1	BAh	Harici kesme 1 ( $\overline{INT1}$ ) öncelik biti
IP.3	PT1	BBh	Zamanlayıcı/sayıcı 1 (T1) kesme öncelik biti
IP.4	PS	BCh	Seri port kesme öncelik biti
IP.5	PT2	BDh	Zamanlayıcı/sayıcı 2 (T2) kesme öncelik biti
IP.6	----	BEh	Kullanılmıyor
IP.7	----	BFh	Kullanılmıyor



# Kesme Vektör Adresleri

- ❑ Program belleğinde, her bir kesme kaynağı için kesme hizmet program (**ISR**) alanı tahsis edilmiştir.
- ❑ Bir kesme geldiğinde, program doğrudan kesmeye tahsis edilmiş olan program belleğindeki alanın başlangıç adresine gider ve bu alandaki komutlar işletilir.
- ❑ Kesme kaynakları için ayrılan alanın program belleğindeki başlangıç adresi “**Kesme Vektörü**” olarak adlandırılır.

Kesme Kaynağı	Bayrak	Kesme Vektör Adresi
Reset	RST	0000h
Harici Kesme 0	IE0	0003h
Zamanlayıcı/sayıcı 0	TF0	000Bh
Harici Kesme 1	IE1	0013h
Zamanlayıcı/sayıcı 1	TF1	001Bh
Seri Port	RI, TI	0023h
Zamanlayıcı/sayıcı 2	TF2, EXF2	002Bh

# Kesme Vektör Adreslerinin Kullanımı

```
ORG 0000H
SJMP ANA
ORG 0003H
KESME_INT0: ..... ; 8 Baytlık Harici Kesme 0 (INT0) Servis Rutini
.....
RETI
ORG 000BH
KESME_T0: ..... ; 8 Baytlık Zamanlayıcı/Sayıcı 0 (T0) Kesme Servis Rutini
.....
RETI
ORG 0013H
KESME_INT1: ..... ; 8 Baytlık Harici Kesme 1 (INT1) Servis Rutini
.....
RETI
ORG 001BH
KESME_T1: ..... ; 8 Baytlık Zamanlayıcı/Sayıcı 1 (T1) Kesme Servis Rutini
.....
RETI
ORG 0023H
KESME_SERI: ..... ; 8 Baytlık Seri Port Kesme Servis Rutini
.....
RETI
ORG 0030H
ANA: .....
.....
END
```

# Kesme Vektör Adreslerinin Kullanımı

- ❑ 8 Baytlık Vektör Adres alanına sığmayan kesme hizmet programları ayrı bir bellek alanında yazılır.
- ❑ Vektör Adres, alanına bu kesme hizmet programına yönlendirecek dallanma komut satırı yazılır.

```
ORG 0000H  
SJMP ANA
```

```
ORG 000BH  
LJMP KESME_TO           ; 8 Bayttan Büyük Zamanlayıcı/Sayıcı 0 (T0) Kesme Hizmet Programına Git
```

```
ANA:  ORG 0030H  
      .....  
      .....  
      END
```

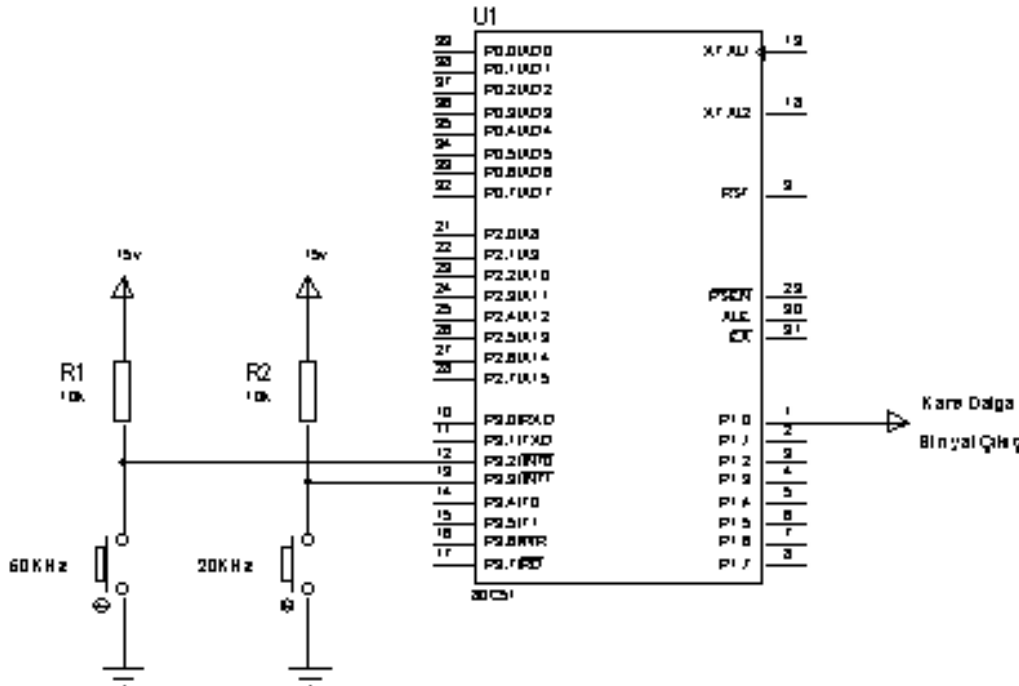
```
KESME_TO: .....           ; 8 Bayttan Büyük Zamanlayıcı/Sayıcı 0 (T0) Kesme Hizmet Programı  
          .....  
          RETI
```

❑ Bir kesme isteği alındığında mikrodenetleyicideki işlemler

- O an çalıştırılmakta olan komutun çalışması tamamlanır.
- Program sayacının değeri (**PC**) yığına (**Stack**) kaydedilir.
- Gelen kesme durumu dahili olarak kaydedilir.
- Diğer kesmeler (**düşük öncelikli**) engellenir.
- Program sayacı, kesme hizmet programının vektör adresi ile yüklenir.
- Kesme hizmet programı çalışmaya başlar.

# Kesme Örneği - 1

- ❑ Şekilde görüldüğü gibi 8051'in harici kesme girişlerine iki buton bağlanmıştır. INTO girişine bağlı butona basıldığında P1.0 ucunda 50KHz'lik, INT1 girişine bağlı butona basıldığında P1.0 ucunda 20KHz'lik kare dalga sinyal çıkışı oluşmaktadır. Gerekli çalışmayı sağlayacak programı kesme yapısını kullanarak assembly dilinde yazınız. (INT1 kesmesi, INTO kesmesine göre daha yüksek öncelikli olacaktır)



BASLA:

```
ORG 0H
SJMP BASLA
ORG 3H
CLR TR1
SETB TR0
RETI
ORG 0BH
CPL P1.0
RETI
ORG 13H
CLR TR0
SETB TR1
RETI
ORG 30H
MOV IE,#8FH
SETB IT0 ;düşen kenar
SETB IT1
SETB PX1
MOV TMOD,#22H
MOV TH0,#-10 ;50KHz için
MOV TL0,#-10
MOV TH1,#-25 ;20 KHz için
MOV TL1,#-25
SJMP $
END
```

# Kesme Örneği - 2: Kesme ile Seri Port Kullanımı

- ❑ Bir 8051 mikrodenetleyicisi ile bir PC, RS-232 seri ara yüzünden haberleşmektedir. 8051 mikrodenetleyicisi PC tarafından gönderilen karakterleri almaktadır. 8051'in aldığı karakterler **büyük harf ise PC'ye büyük harfi küçük harfe dönüştürerek** göndermekte, **değilse FFh** bilgisi göndermektedir. Gerekli programı 8051 **kesme organizasyonunu kullanarak** assembly dilinde yazınız?
- ❑ (8051 seri arabirimi 2400 baud hızında, 8 bit Timer1 ile kullanılacaktır, OSC=12 MHz için TH1= -13)
- ❑ (Not: ASCII kodlar A-Z = 65 – 90, a - z = 97 – 122)

```
ORG 0H
SJMP BASLA
```

```
ORG 23H
SJMP SERI
```

```
ORG 30H
```

```
BASLA: MOV IE,#90H      ; seri kesme aktif ES=1, EA=1
        MOV SCON,#50H  ; mod 1 8 bit degisken hiz
        MOV TMOD,#20H  ; mod 2 otomatik yukleme
        MOV TH1,#-13   ; 2400 baud
        MOV TL1,#-13
        SETB TR1
        SJMP $
```

; programın devamı

```
SERI:   MOV A,SBUF      ; seri porttan karakteri al
        MOV R1,A        ; yedekle
```

```
CLR RI
```

```
SUBB A,#65      ; A'dan öncemi
```

```
JNC HARFZ      ; değilse Z'den öncemi
```

```
SJMP YUKLEFF   ; A'dan önce ise FF yukle
```

```
HARFZ: MOV A,R1
```

```
SUBB A,#90
```

```
JNC YUKLEFF
```

```
KUCUK: MOV A,R1      ; buyuk harfi kucuk harfe döndür
```

```
ADD A,#32
```

```
SJMP GONDER
```

```
YUKLEFF: MOV A,#0FFH
```

```
GONDER: CLR TI
```

```
MOV SBUF,A
```

```
RET
```

END