

MIKRODENETLEYICILER

Ege Üniversitesi Ege MYO
Mekatronik Programı

BÖLÜM 5

Zamanlayıcılar

Zamanlayıcı/Sayıcı

3

Zamanlayıcı/Sayıcı

- Zamanlayıcı: Zaman geciktirici olarak kullanıldığında verilen isim.
- Sayıcı: Dışarıda gerçekleşen bir olay sayıldığında verilen isim.

8051 T0 ve T1 olmak üzere iki adet 16 bit zamanlayıcıya sahiptir.

Çalışma kipleri TMOD yazacından seçilir

- Çalışmaları TCON yazacından denetlenir.
- Zamanlayıcının çalışma hızı = $f_{osc} / 12$ 'dir.

8052 T0 ve T1 ek olarak 16 bit T2 zamanlayıcısına sahiptir.

- T2 zamanlayıcısının çalışma kiplerinin seçimi ve denetimi T2CON yazacından yapılır.

Zamanlayıcı Yazaaçları

4

- 8051 zamanlayıcılar 8 bitlik yazaaçlardan oluşur. Bu yazaaçlara diğer SFR yazaaçları gibi erişilebilir.
- T0 yazaaçları TH0, TL0, T1 yazaaçları TH1, TL1.
- T2 zamanlayıcı yazaaçları da SFR bölgesinde yer alır, TH2, TL2, RCAP2L, RCAP2H

8052'de Zamanlayıcı Yazaaçları

5

SFR	Kullanım Amacı	Adres	Bit Adresleme
TCON	Denetim	88H	Var
TMOD	Çalışma kipi belirleme	89H	Yok
TL0	Zamanlayıcı 0 DDB	8AH	Yok
TL1	Zamanlayıcı 1 DDB	8BH	Yok
TH0	Zamanlayıcı 0 YDB	8CH	Yok
TH1	Zamanlayıcı 1 YDB	8DH	Yok
T2CON*	Zamanlayıcı 2 denetim	C8H	Var
RCAP2L*	Zamanlayıcı DDB yakalama	CAH	Yok
RCAP2H*	Zamanlayıcı YDB yakalama	CBH	Yok
TL2*	Zamanlayıcı 2 DDB	CCH	Yok
TH2*	Zamanlayıcı 2 YDB	CDH	Yok

* 8051, 8751, 8031' de yoktur.

(MSB)

(LSB)

GATE	C/ \bar{T}	M1	M0	GATE	C/ \bar{T}	M1	M0
Timer1				Timer0			

BİT	ADI	Z/S	Açıklama
7	GATE	1	Geçit biti kurulduğunda T1 sadece INT1 yüksek seviyede ise çalışır.
6	C/T	1	Sayıcı zamanlayıcı seçme biti, 1 ise sayıcı, 0 ise zamanlayıcı.
5	M1	1	Kip bit 1 (Bakınız Tablo-5.3'e)
4	M0	1	Kip bit 0 (Bakınız Tablo-5.3'e)
3	GATE	0	T0 geçit biti.
2	C/T	0	Sayıcı zamanlayıcı seçme biti, 1 ise sayıcı, 0 ise zamanlayıcı.
1	M1	0	Kip bit 1 (Bakınız Tablo-5.3'e)
0	M0	0	Kip bit 0 (Bakınız Tablo-5.3'e)

Çalışma Kipleri

7

M1	MO	KİP	AÇIKLAMA
0	0	0	13 bit zamanlayıcı kipi (8048 kipi olarak ta adlandırılır)
0	1	1	16 bit zamanlayıcı kipi.
1	0	2	8 bit yeniden yüklemeli zamanlayıcı kipi.
1	1	3	Ayrık zamanlayıcı kipi. Zamanlayıcı 0; TL0 8 bit zamanlayıcı çalışması zamanlayıcı 0 kip seçme bitleri tarafından denetlenir. TH0 aynı şekilde çalışır fakat zamanlayıcı 1 kip seçme bitleri tarafından denetlenir. Zamanlayıcı 1; çalışmaz.

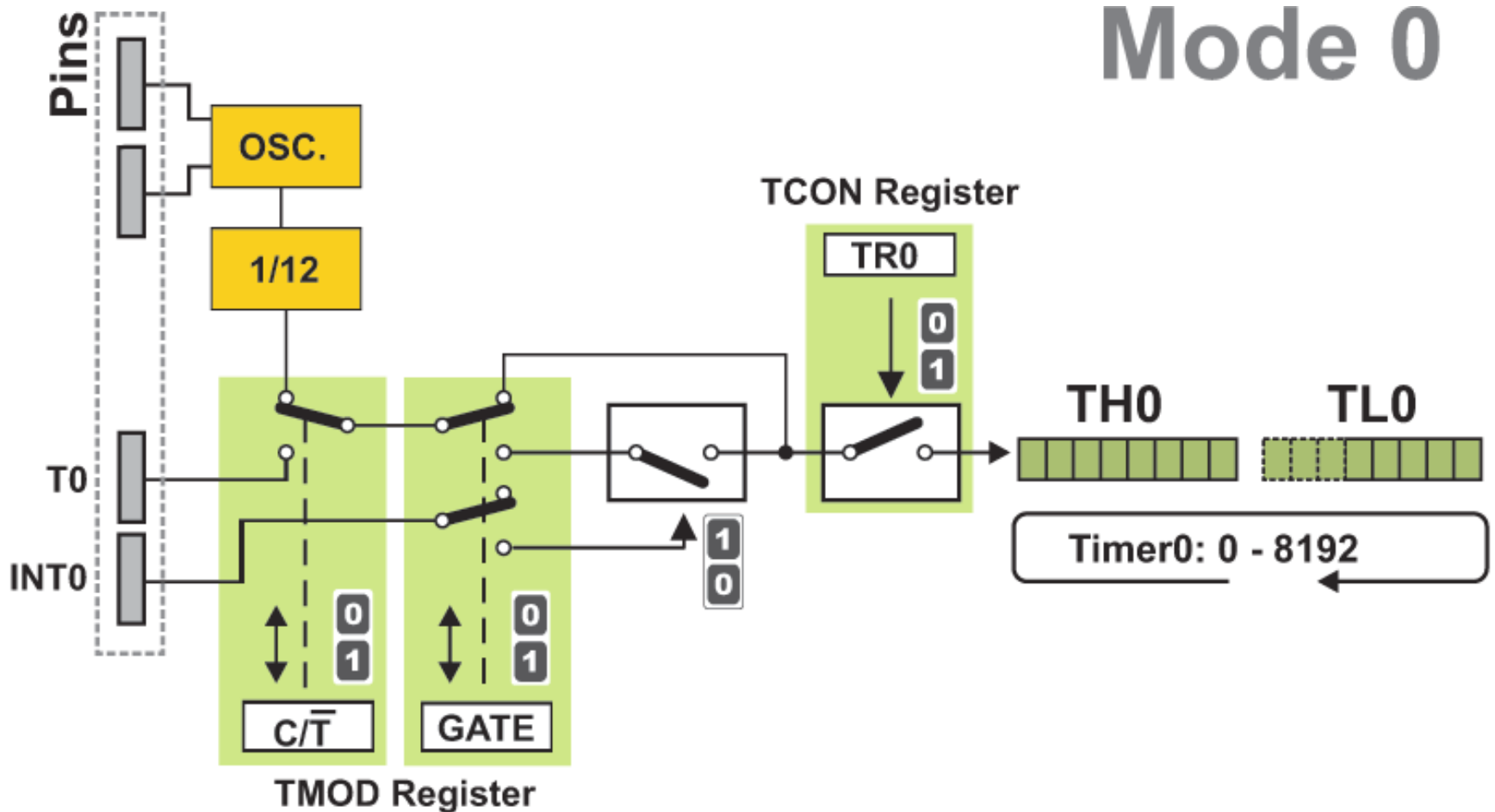
Zamanlayıcı Denetim Yazacı

8

BİT	ADI	ADRES	AÇIKLAMA
TCON.7	TF1	8FH	T1 taşma bayrağı, taşma olduğunda kurulur, yazılımla veya donanım tarafından kesme vektör adresine gidildiğinde temizlenir.
TCON.6	TR1	8EH	T1 çalıştırma/durdurma biti yazılım ile içeriği değişir.
TCON.5	TF0	8DH	T0 taşma bayrağı.
TCON.4	TR0	8CH	T0 çalıştırma/durdurma biti.
TCON.3	IE1	8BH	Dış kesme 1 kenar bayrağı. INT1 girişinde düşen kenar geldiğinde kurulur; yazılım veya CPU kesme vektörüne bağlandığında donanım ile temizlenir.
TCON.2	IT1	8AH	Dış kesme 1 tip seçme bayrağı. Yazılım ile kurulup temizlenir. kurulduğunda olduğunda INT1'de düşen kenarda kesme bayrağı kurulur; temizlendiğinde düşük seviyede kesme algılanır ve kesme bayrağı kurulur.
TCON.1	IE0	89H	Dış kesme 0 kenar bayrağı.
TCON.0	IT0	88H	Dış kesme 0 tip seçme bayrağı.

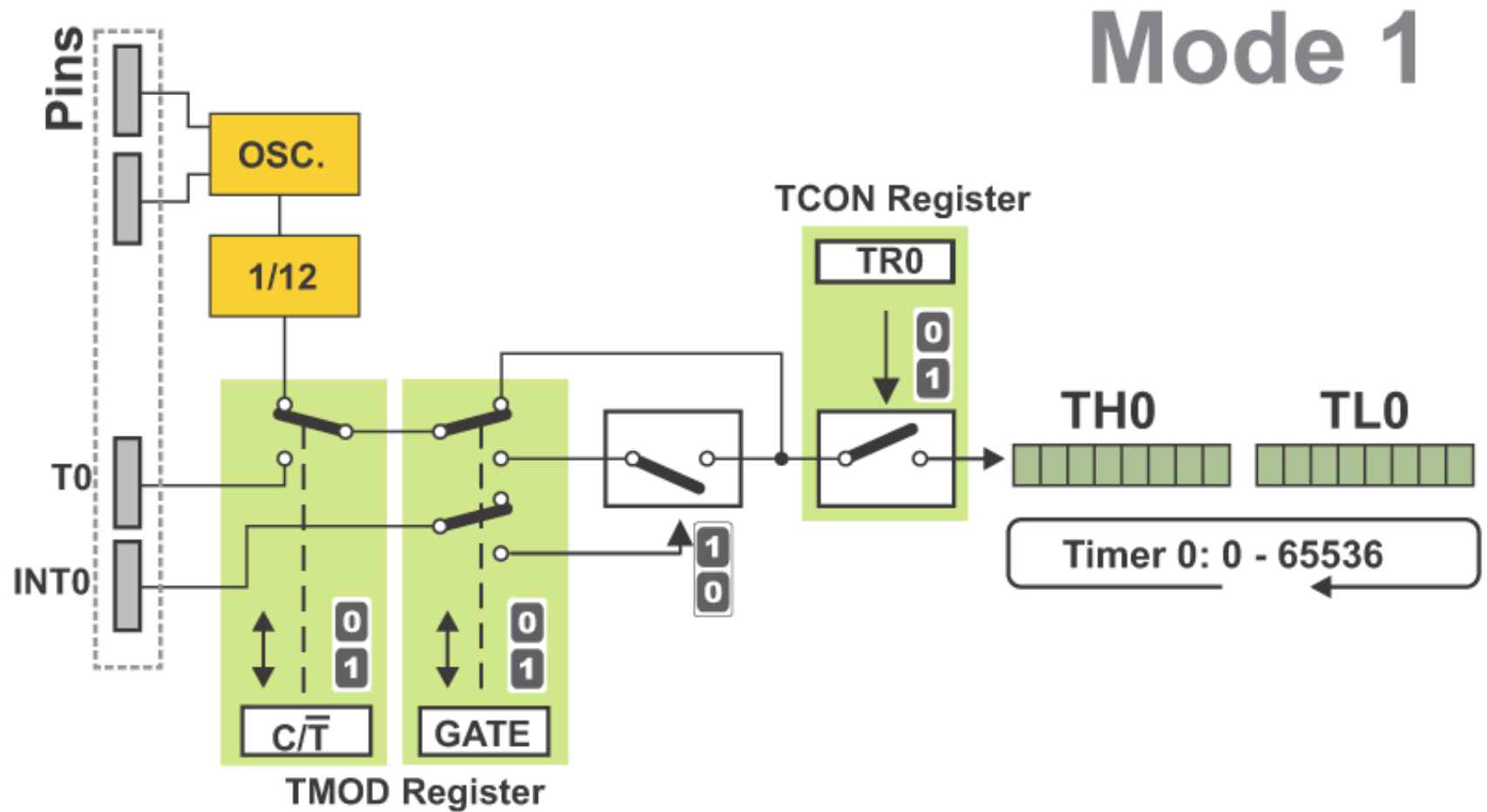
Kip 0, 13 Bit Zamanlayıcı

9



Kip 1, 16 Bit zamanlayıcı

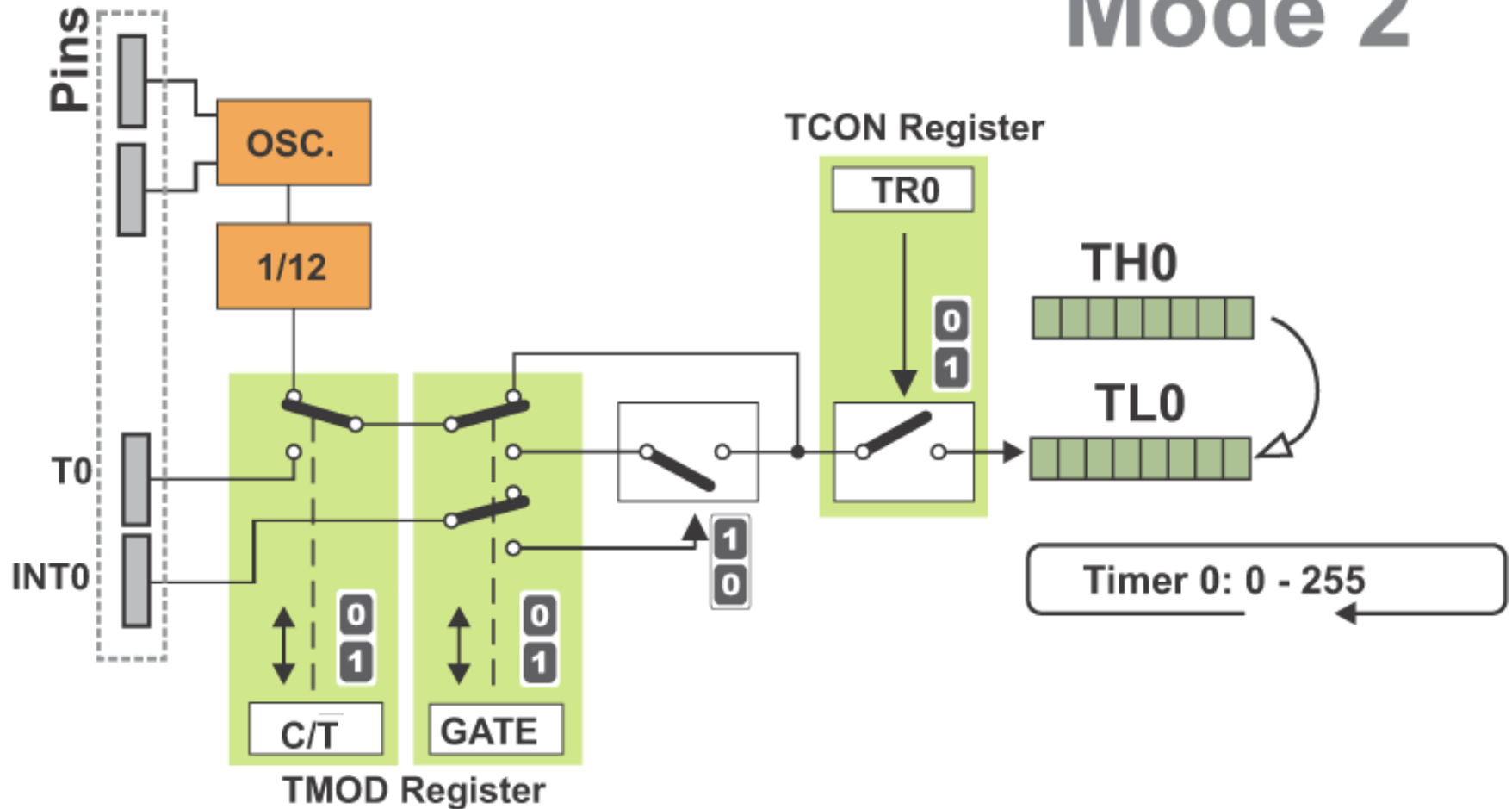
10



Kip 2, Yeniden Yüklmeli

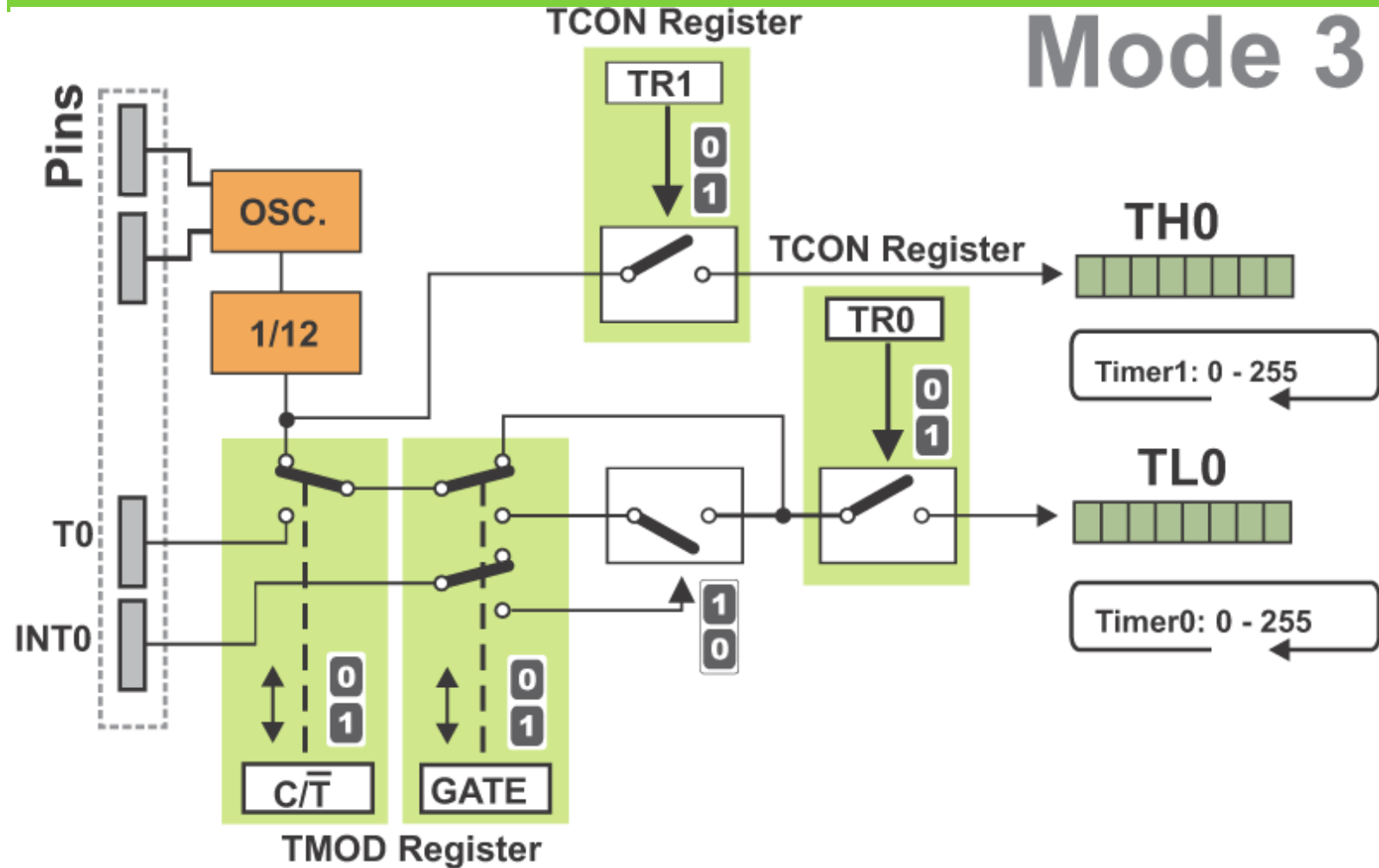
11

Mode 2



Kip 3 Ayrik Zamanlayıcı

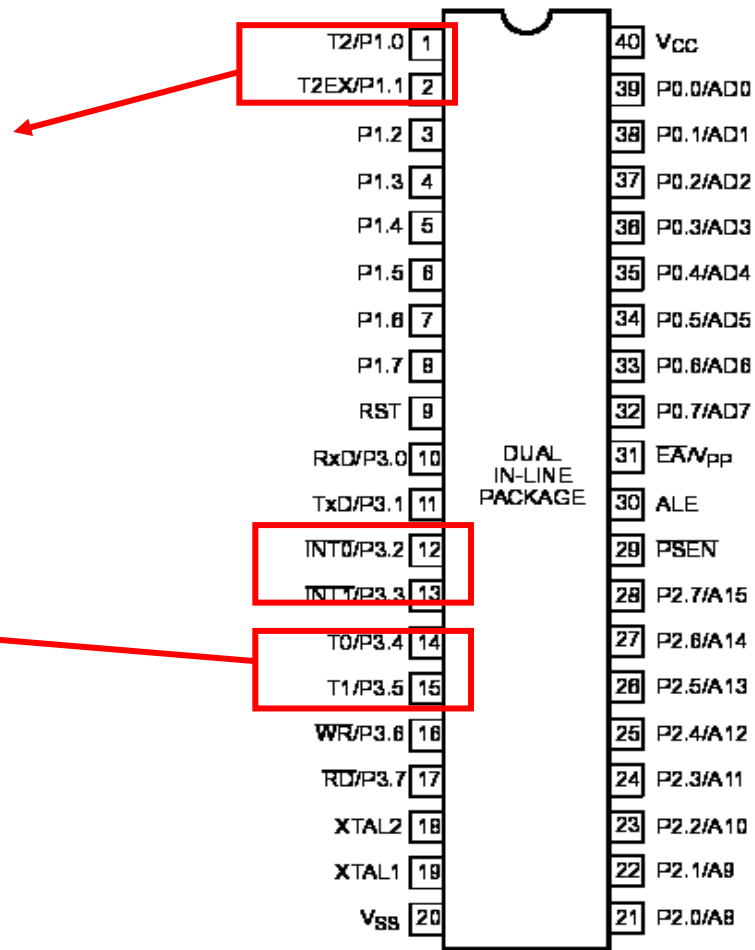
12



Zamanlayıcı Giriş ve Çıkış Bacakları

13

- 8052, P1'in iki bacağı, zamanlayıcı 2 tetikleme girişi ve vuru çıkışı olarak görevlendirilmiştir.
- Zamanlayıcı 0 ve 1'in dış tetikleme girişleri



Zamanlayıcıları Kullanma Algoritması

14

1. Çalışma kipini belirle (TMOD)
2. Başlangıç değerini belirle (THX, TLX)
3. Çalıştır (SETB TRX)
4. Zaman aşım bayrağını test et (JNB TFX,\$)
 - ▣ Evet adım 5'e git
 - ▣ Hayır adım 4'e git
5. TFX=0 yap, iş yap.
6. Adım 4'e git (sürekli yapılacaksa), son.

Örnek 1: T0'ı kullanarak P1.5'den sürekli D=%50 olan kare dalga üreten programı kip 2'de yazın (Frekans yüklenen sayı ile değişir)

15

T0D50_K2:		T0 Sayma Sırası
Mov TMOD,#00000010b	;T0, kip 2	F2 TF0=0
Mov TH0,#0F2H	;Başlangıç=F2H	F3 TF0=0
Mov TL0,#0F2h		F4 TF0=0
Setb TR0	;T0'ı çalıştır.
tekrar:	
Jnb TF0,\$;TF0'ı zaman aşımı için tara.	FE TF0=0
Clr TF0	;TF0 bayrağını temizle	FF TF0=1
Cpl P1.5		
Sjmp tekrar		

Örnek 2: P3.5'teki vuruların sayısını P2'de görüntüle.

16

Say_8:

Mov TMOD, #01100000B ;kip 2, C/T=1

Mov TH1, #0 ;0x00 - 0xFF say.

Setb P3.5 ;P3.5 giriş

Setb TR1 ;Çalıştır

oku:

Mov P2, TL1 ; TL1'i oku, P2'de göster

Jnb TF1,oku ;TF1 kuruldu mu? Tara.

Clr TF1 ;TF1 bayrağını temizle.

Inc B ;Taşma oldu ise B'yi arttır

Sjmp oku ;Sürekli yap.

- Zamanlayıcıların yetmediği durumlarda yazılım zamanlayıcısı olarak ta adlandırılan Zaman Geciktirme Döngüleri yazılır.
- Zamanlayıcılar kullanıldığında aynı anda CPU başka komut yürütebilir.
- ZGD ile zamanlamada CPU meşgul olduğu için başka komut yürütemez.
- Zamanlayıcıların yetersiz kaldığı durumlarda ZGD ile birlikte kullanılabilir.

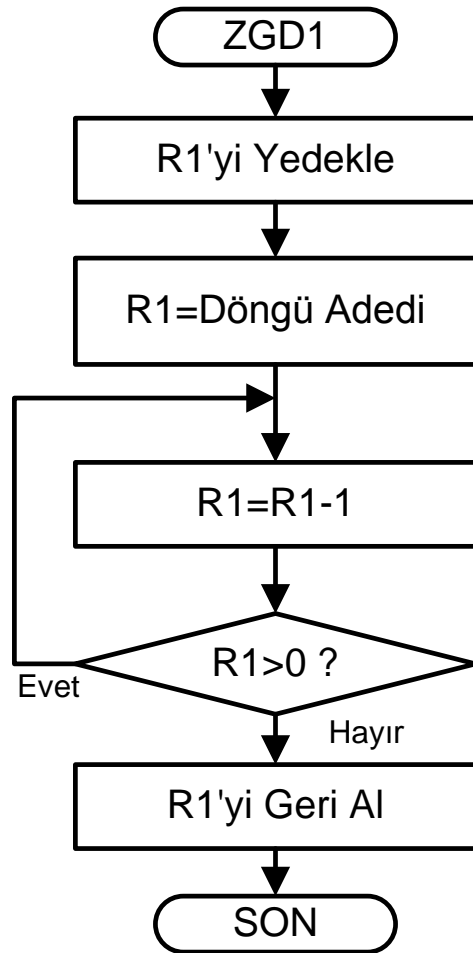
ZGD Algoritması

18

1. Yazaca değer yüklenir
2. Yazacı bir azaltılır
3. Sıfır değilse adım 2'ye git.
4. Sıfır ise son

Akış Diyagramı

19



Tek Döngülü ZGD

	Komut	Saykıl	Tekrarlanma	Gecikme
Zgd_1:	PUSH 01h	2	1	2 saykıl
	MOV R1,#X	1	1	1 saykıl
	DJNZ R1,\$	2	X	2X saykıl
	POP 01h	2	1	2 saykıl
	RET	2	1	2 saykıl

$$T_{MS} = (2X + 7) \text{ Saykıl}$$

En Küçük ve En Büyük Gecikme

21

- En düşük gecikme $X=1$;
- $T_{MS} = 2.1 + 7 = 9$ makine saykılı.
- En yüksek gecikme $X=255$;
- $T_{MS} = 2.255 + 7 = 517$ makine saykılı.

NOP Ekleyerek Süreyi Uzatmak

	Komut	Saykıl	Tekrarlanma	Gecikme
Zgd_1:	PUSH 01h	2	1	2 saykıl
	MOV R1,#X	1	1	1 saykıl
Z1:	NOP	1	X	X
	NOP	1	X	X
	DJNZ R1, Z1	2	X	2X saykıl
	POP 01h	2	1	2 saykıl
	RET	2	1	2 saykıl

$$T_{MS} = 4X+7 \text{ Saykıl}$$

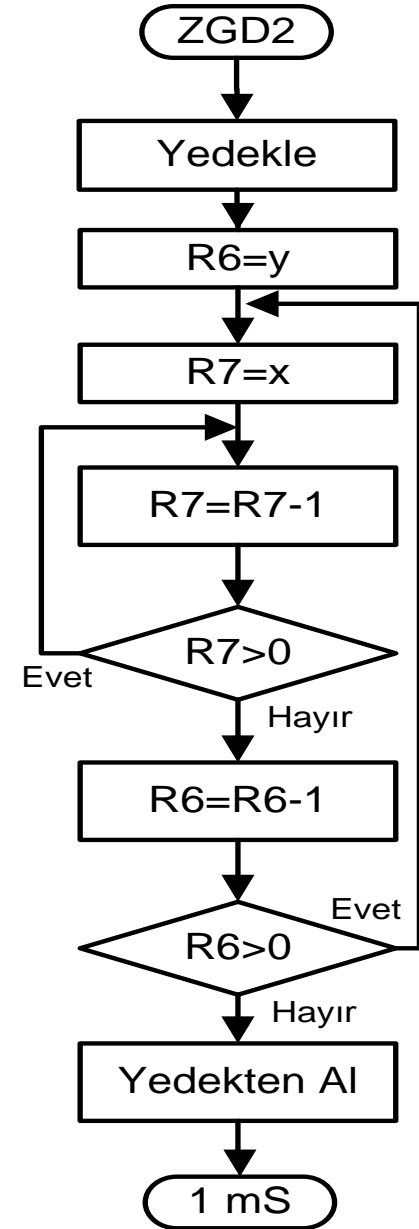
1 ms İçin Yazaç Değeri

23

- Elde edilmek istenen süre $t=1$ mS olduğunda ve $FOSC=12$ MHz ise;
- $T_{\text{makine}} \text{ saykılı} = 12 / 12 \cdot 10^6 = 1 \mu\text{S}$
- $T_{\text{MS}} = t / T = 1000 \mu\text{S} / 1 \mu\text{S} = 1000$ makine saykılı
- $1000 = 4 \cdot X + 7$
- $X = (1000 - 7) / 4 = 248.25$ olarak hesaplanır.

Çift Döngülü ZG Altprogramı Algoritması

1. İkinci Yazaca değer yüklenir
2. Birinci Yazaca değer yüklenir
3. Birinci Yazacı bir azaltılır
4. Birinci yazacı Sıfır değilse adım 3'e git.
5. Sıfır ise, İkinci yazacı bir azalt
6. İkinci yazacı Sıfır değilse adım 2'ye git.
7. Sıfır ise son



Toplam Makine Saykılının Hesaplanması

	Komut	Saykıl	Tekrarlanma	Gecikme
Zgd_2:	PUSH 07h	2	1	2 saykıl
	PUSH 06h	2	1	2 saykıl
	MOV R6,#y	1	1	1 saykıl
Z2:	MOV R7,#x	1	y	y saykıl
	DJNZ R7,\$	2	x.y	2xy saykıl
	DJNZ R6,Z2	2	y	2y saykıl
	POP 06h	2	1	2 saykıl
	POP 07h	2	1	2 saykıl
	RET	2	1	2 saykıl

$$T_{MS} = 2xy + 3y + 11$$

Örnek

26

$f_{OSC}=8$ MHz olan 8051 için 100 mS'lik zaman geciktirme altprogramını yazın.

- $T = 12/8 \cdot 10^6 = 1.5 \mu S$
- $T_{MS} = t / T = 100000 \mu S / 1.5 \mu S = 66666.66$ saykıl
- $T_{MS} = 66667$ saykıl alınabilir.
- $66667 = 2xy + 3y + 11$
- $X=255$ ile çözüme başlanır
- $66667 = 2 \cdot 255 \cdot y + 3y + 11$
- $66667 - 11 = 513y$
- $y = 129.93$
- Üste tamamlayarak 130 kabul edilebilir

Devamı

27

- Kabul edilen ve bulunan değerler denklemde yerine konur ve hata hesaplanır.
- $66667 = 2.255.130 + 3.130 + 11$
- $66667 = 66701$
- Hata 34 makine saykılı
- 1 makine saykılı $1.5 \mu\text{S}$ olduğuna göre $51 \mu\text{S}$ 'lik bir sapma gerçekleşir.
- Bu değer kullanıldığı uygulamaya bağlı olarak kabul edilebilir.

Devamı

28

- Bu hata yaklaşıma devam edilerek küçültülebilir.
- $x = 254$ edilirse.
- $66667 = 2.254.y + 3.y + 11$
- $y = \underline{130.44}$ alta tamamlanır 130 alınır
- Sapma -220 üste tamamlanırsa +285 olur.
- Sapma artmıştır, çözüme devam edilirse hata küçülecektir.
- Adımları birer birer azaltmak yerine daha yüksek değerle yaklaşım yapılabilir.

ÖRNEK 5.3 P1. 6'ya bir buton, P1.7'ye bir buzzer bağlanmıştır. Butona her basıldığında buzzerı 1 saniye öttüren programı yazın.

29

```
TUS_BUZ:
    SETB P1.7      ;başlangıçta buzzerı kapat
    MOV TMOD, #01H ;T0 16 bit zamanlayıcı
    SETB P1.6      ;Tuşun bağlandığı hattı giriş yap
TUS1:   JB P1.6,$   ;tuşa basıldı mı?
        JNB P1.6,$
        CLR P1.7   ;buzzerı aç.
        CALL ZGD1  ;1 saniye buzzer ötsün.
        SETB P1.7  ;buzzerı kapat.
        SJMP TUS1  ;sürekli yap

ZGD1:   MOV R7,#20 ;50000 X 20=1 saniye
ZGD3:   MOV TH0,#3CH ;YDB'ı yükle.
        MOV TL0,#0B0H ;DDB'ı yükle.
        SETB TR0    ;zamanlayıcıyı çalıştı
ZGD2:   JNB TF0,ZGD2 ;zaman aşıldı mı?
        CLR TF0    ;evet, bayrağı temizle
        CLR TR0    ;T0'ı durdur.
        DJNZ R7,ZGD3 ;1 saniye oldu mu?
        RET        ;evet son
```

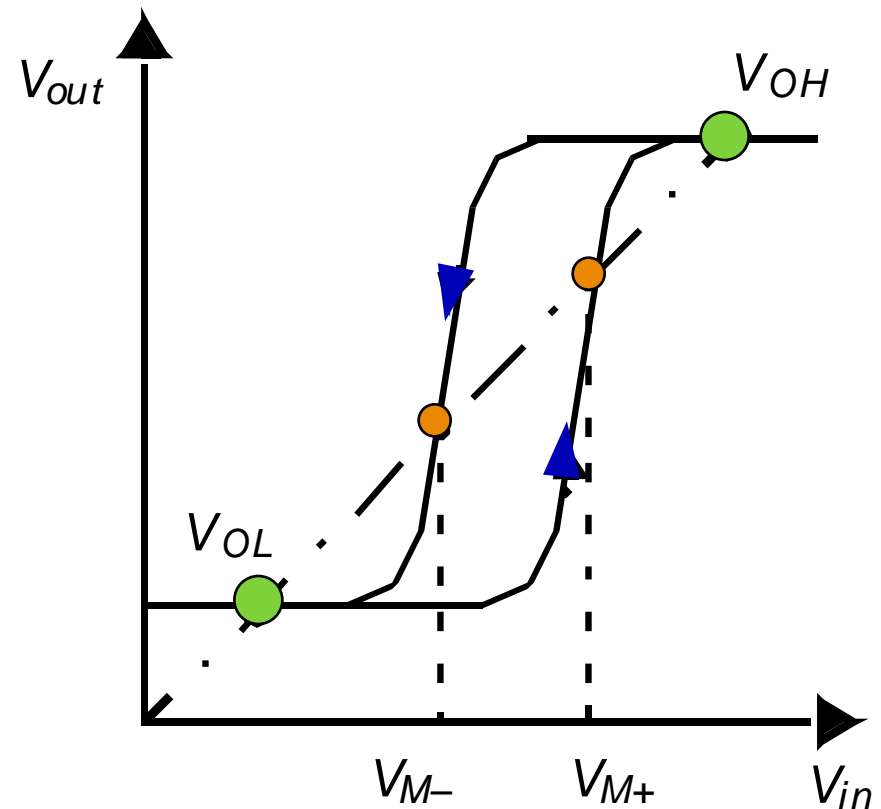
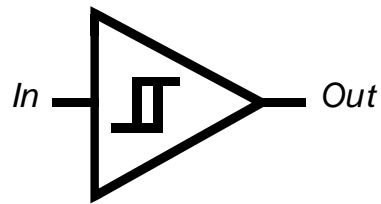
Ödev 5: ZGD yazmak

30

- 100 μS
- 1 mS
- 100 mS
- 1 S
- 1 dakikalık zaman geciktirme döngülerini yazarak simulatörde çalıştırın.

Non-Bistable Sequential Circuits Schmitt Trigger

31



Schmitt Trigger Gürültü Yoketme

32

