

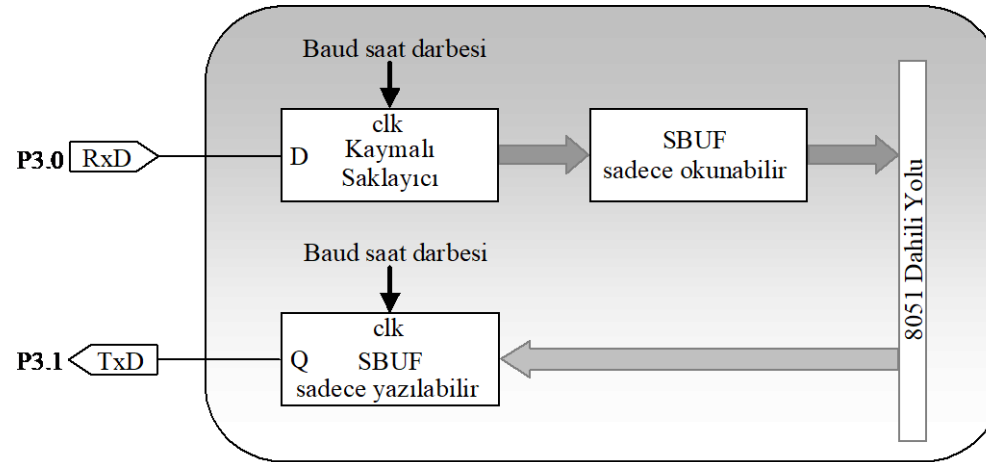


SAKARYA ÜNİVERSİTESİ
Bilgisayar ve Bilişim Bilimleri Fakültesi
Bilgisayar Mühendisliği Bölümü

Mikroişlemcili Sistemler ve Laboratuvarı

Seri Portun Genel Çalışması

- ❑ Veri gönderimi için paralel formdaki veriyi seriye ve dışardan alınan seri formdaki veriyi de paralel veriye dönüştürür.
- ❑ Veri gönderme **TxD** (P3.1) ve veri alma ise **RxD** (P3.0)'dir.
- ❑ En önemli özelliği tam çift yönlü (full-duplex, eşzamanlı veri gönderme/alma) veri iletişimi yapabilmidir.
- ❑ Seri haberleşme işlemlerini **SCON-Serial Port Control** kontrol etmekte ve tamponlama işlemi **SBUF-Serial Port Buffer** yardımıyla yapılmaktadır.



Seri Port Kontrol Saklayıcısı

SCON**Seri Port Kontrol Saklayıcısı**

	0	0	0	0	0	0	0	0	Reset Değeri
98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	Bit Adı
	Bit 7								Bit 0

Bit No	İsmi	Bit Adresi	Açıklama
0	RI	98h	Alma Kesme Bayrağı (Receive Interrupt Flag). Veri alma işlemi (tüm bayt) tamamlandığında '1''e kurulur, yazılımla sıfırlanır ('0').
1	TI	99h	Gönderme Kesme Bayrağı (Transmit Interrupt Flag). Veri gönderme işlemi (tüm bayt) tamamlandığında '1''e kurulur, yazılımla sıfırlanır ('0').
2	RB8	9Ah	Alınan bit-8 (Receive bit-8). Mod 2 ve Mod 3'de alınan 9. biti tutar. Mod 1'de dur (STOP) bitidir, Mod 0'da kullanılmaz.
3	TB8	9Bh	Gönderilecek bit-8 (Transmit bit-8). Mod 2 ve Mod 3'de gönderilecek 9.biti tutar.
4	REN	9Ch	Alımı Yetkilendirme (Receiver Enable). Veri alımı için yazılımla '1' yapılmalıdır. '0' olduğunda veri alımı yapılmaz.
5	SM2	9Dh	Seri port mod seçme biti (2). Çok işlemcili haberleşmeyi yetkilendirir. Mod 2 ve Mod 3'de çok işlemcili haberleşmede kullanılır.
6	SM1	9Eh	Seri port mod seçme biti (1).
7	SM0	9Fh	Seri port mod seçme biti (0).

Seri Port Çalışma Modları

- ☐ Seri port çalışma modları SCON saklayıcısındaki SM0 ve SM1 bitleri yardımıyla belirlenir.

SM0	SM1	MOD	Açıklama	Baud Hızı
0	0	0	Kaymalı Saklayıcı (Shift Register)	Sabit (Osilatör frekansı / 12)
0	1	1	8-Bit UART	Değişkendir, T1 tarafından ayarlanabilir
1	0	2	9-Bit UART	Sabit (Osilatör frekansı / 32 veya 64)
1	1	3	9-Bit UART	Değişkendir, T1 tarafından ayarlanabilir

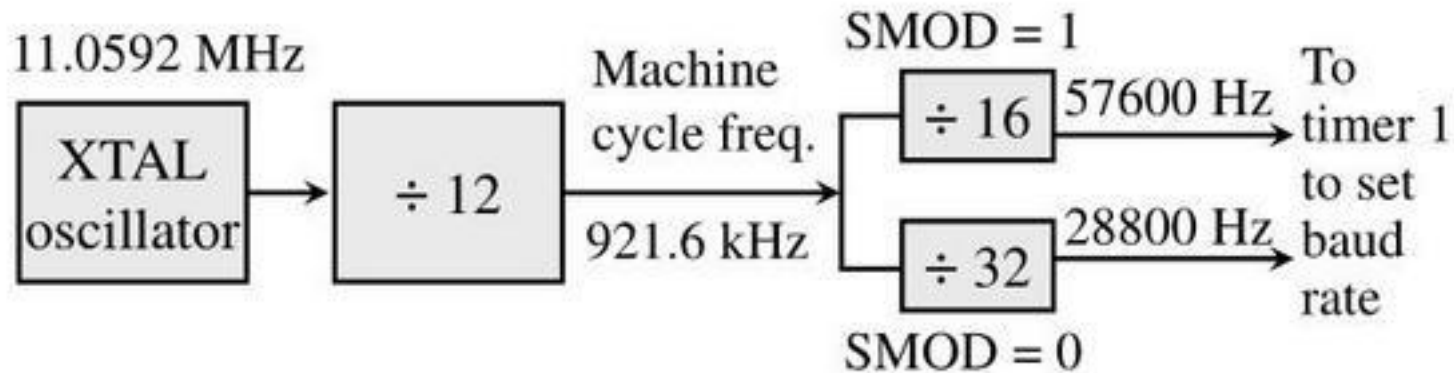
Not: Mode 2-3 çoklu işlemci haberleşmede ve parity göndermede kullanılır.

❑ Seri haberleşmeye başlamadan önce yazılım yoluyla şu işlemler yapılmalıdır:

- Çalışma modu
- Baud hızı
- Eşlik bitinin eklenmesi
- Alımın yetkilendirilmesi
- Eğer Mod 2 ya da Mod 3 kullanılıyorsa 9. bitin kullanımı
- Kesme bayraklarının kullanımı

Seri Portun Kullanımı

- Seri Port, Timer1'in ayarlanması:



Seri Portun Kullanımı

- XTAL = 11.0592 MHz, TH1 değerleri neler olmalıdır
- (a) 9600 (b) 2400 (c) 1200
- **Çözüm:**
- XTAL = 11.0592 MHz:
- 8051'in makine çevrim frekansı = $11.0592 \text{ MHz} / 12 = 921.6 \text{ kHz}$,
- ve $921.6 \text{ kHz} / 32 = 28,800 \text{ Hz}$, UART tarafından Timer 1'in baud hızını ayarlamak için sağladığı frekanstır.
- $28,800/3 = 9,600$ TH1=-3
- $28,800/12 = 2,400$ TH1=-12
- $28,800/24 = 1,200$ TH1=-24

❑ İstenilen baud hızı için zamanlayıcıların kurulması gereken değerler:

Baud Hızı (bit/s)	Kristal Frekansı (MHz)	SMOD	TH1	Gerçek Baud Hızı (bit/s)	Hata Oranı
9600	12,000	1	-7 (F9h)	8923	%7
2400	12,000	0	-13 (F3h)	2404	%0,16
1200	12,000	0	-26 (E6h)	1202	%0,16
19200	11,0592	1	-3 (FDh)	19200	0
9600	11,0592	0	-3 (FDh)	9600	0
2400	11,0592	0	-12 (F4h)	2400	0
1200	11,0592	0	-24 (E8h)	1200	0

Seri Port Örneği - 1

❑ A'dan Z'ye kadar harfleri seri porttan gönderen uygulama programı:

Komut	Açıklama
ORG 0h	<i>;Kod belleğin başlangıç adresi</i>
SJMP ANA	<i>;ANA etiketli programa dallan</i>
ORG 30h	<i>;ANA programının başlangıç adresi</i>
ANA: MOV SCON,#00010010B	<i>;Mod 0, REN=1, ilk gönderim için TI=1</i>
MOV A,#41h	<i>;A harfinin ASCII karşılığını aküye ;yükle</i>
Bekle: JNB TI,Bekle	<i>;TI=1 olana kadar bekle</i>
CLR TI	<i>;TI'yı sıfırla</i>
MOV SBUF,A	<i>;Aküyü SBUF'a yükle</i>
INC A	<i>;Aküdeki değeri 1 arttır ;(bir sonraki ;harf)</i>
CJNE A,#5Ch,Bekle	<i>;Z (5Ch) harfine kadar gönderme ;kontrolü</i>
END	<i>;Programı bitir</i>

Seri Port Örneği - 2

❑ D harfini seri porttan gönderen uygulama programı:

ORG 0H

SJMP BASLA

ORG 30H

BASLA: MOV TMOD,#20H ; timer 1,mode 2(auto reload)

MOV TH1, #-3 ; 9600 baud rate

MOV SCON, #50H ; 8-bit, 1 stop, REN enabled

SETB TR1 ; start timer 1

AGAIN: MOV SBUF, #"D" ; letter "D" to transfer

HERE: JNB TI, HERE ; wait for the last bit

CLR TI ;clear TI for next char

SJMP AGAIN ; keep sending A

END

Seri porttan sürekli olarak “YES” mesajını gönderen prog. yazın. 9600 baud, 8 bit veri, 1 stop biti

Solution:

```
MOV    TMOD,#20H ;Timer 1, mode 2
MOV    TH1,#-3   ;9600 baud
MOV    SCON,#50H ;8-bit, 1 stop bit, REN enabled
SETB   TR1       ;start Timer 1
AGAIN:  MOV    A, #"Y" ;transfer "Y"
        ACALL  TRANS
        MOV    A, #"E" ;transfer "E"
        ACALL  TRANS
        MOV    A, #"S" ;transfer "S"
        ACALL  TRANS
        SJMP   AGAIN ;keep doing it
;-----serial data transfer subroutine
TRANS:  MOV    SBUF,A ;load SBUF
HERE:   JNB    TI,HERE ;wait for last bit to transfer
        CLR    TI ;get ready for next byte
        RET
```

```

        SW1    EQU P1.7
        ORG    0H                ;starting position
MAIN:    MOV    TMOD,#20H
        MOV    TH1,#-3          ;9600 baud rate
        MOV    SCON,#50H
        SETB   TR1              ;start timer
        SETB   SW1              ;make SW an input
S1:      JB     P2.1,NEXT        ;check SW status
        MOV    DPTR,#MESS1      ;if SW=0 display "NO"
FN:      CLR    A
        MOVC   A,@A+DPTR        ;read the value
        JZ     S1               ;check for end of line
        ACALL  SENDCOM          ;send value to serial port
        INC    DPTR             ;move to next value
        SJMP   FN               ;repeat
NEXT:    MOV    DPTR,#MESS2      ;if SW=1 display "YES"
LN:      CLR    A
        MOVC   A,@A+DPTR        ;read the value
        JZ     S1               ;check for end of line
        ACALL  SENDCOM          ;send value to serial port
        INC    DPTR             ;move to next value
        SJMP   LN               ;repeat
;-----
SENDCOM: MOV    SBUF,A           ;place value in buffer
HERE:    JNB    TI,HERE          ;wait until transmitted
        CLR    TI               ;clear
        RET                    ;return
;-----
MESS1:   DB     "NO",0
MESS2:   DB     "YES",0
        END

```

Bir anahtarın P1.7 pimine bağlı olduğunu varsayın.

Durumunu izlemek ve aşağıdaki gibi sürekli olarak seri bağlantı noktasına iki mesaj gönderen bir program yazın:

SW = 0 "NO" gönderir

SW = 1 "EVET" gönderir

XTAL = 11.0592 MHz, 9600 baud, 8 bit veri ve 1 stop bitini varsayalım.

Seri Port Örneği - 3

- ❑ 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 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 30H

BASLA:

MOV SCON,#50H ; mod 1 8 bit değişken hız

MOV TMOD,#20H ; mod 2 otomatik yük.

MOV TH1,#-13 ; 2400 baud

MOV TL1,#-13

SETB TR1

SERI:

JNB RI, SERI

CLR RI

MOV A,SBUF ; seri porttan karakteri al

MOV R1,A ; yedekle

SUBB A,#65 ; A'dan öncemi

JNC HARFZ ; değilse Z'den öncemi

SJMP YUKLEFF ; A'dan önce ise FF yukle

; programın devamı

HARFZ: MOV A,R1

SUBB A,#90

JNC YUKLEFF

KUCUK: MOV A,R1 ;buyuk harfi kucuk h

ADD A,#32

SJMP GONDER

YUKLEFF: MOV A,#0FFH

GONDER: CLR TI

MOV SBUF,A

SJMP SERI

END

Çok İşlemcili Haberleşme

- ❑ Mod 2 ve Mod 3 seri haberleşme modları kullanılır.
- ❑ Ana (Master) 8051, bir uydu (Slave) 8051'e veri göndermeden önce, iletişim kuracağı uyduyu belirten adres bilgisini gönderir.
- ❑ Ana 8051'den gönderilen bilginin adres ya da veri olduğunu 9. bitin (RB8) durumu belirler. Bu bit '1' ise gönderilen bilgi adres, '0' ise veridir.
- ❑ Bütün uydu'larda kesme oluşur.
- ❑ Tüm uydu'lar adreslenip adreslenmediklerini kontrol ederler.

Çok İşlemcili Haberleşme

- ❑ Adreslenen uydu 8051, kendi SM2 bitini temizler ('0') ve verileri almaya hazır hale gelir.
- ❑ Adreslenmeyen uydu'lar SM2 bitlerini '1' yaparak, gelen verileri dikkate almazlar ve kendi işlerine devam ederler.
- ❑ Bir sonraki adres bilgisine kadar olaylar devam eder.

