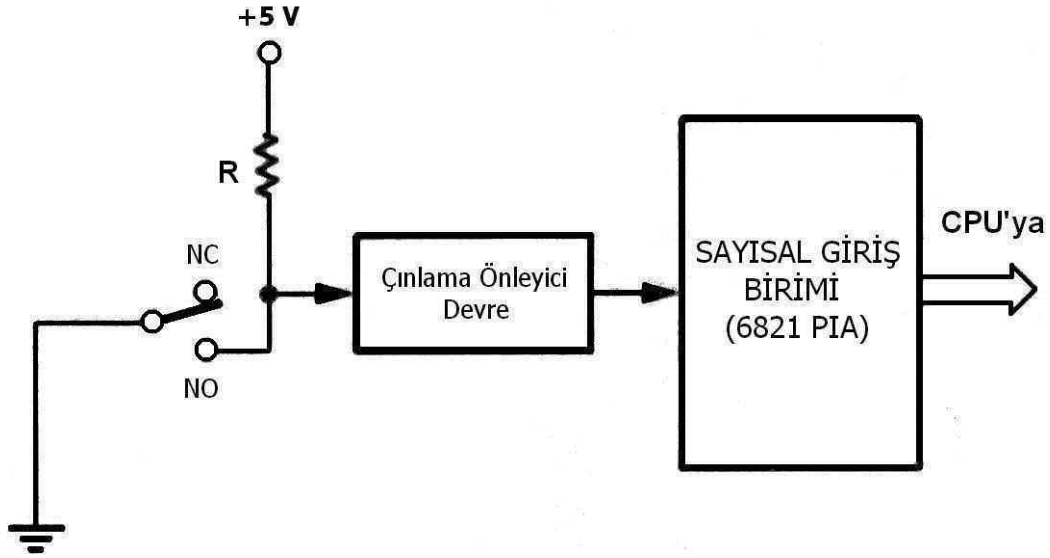


## 18. MİKROİŞLEMCİ TEMELLİ SİSTEM UYGULAMALARI

Mikroişlemci temelli sistemler eğitim, güvenlik, ticari, endüstriyel, askeri, sağlık, vs. alanlardaki uygulamalarda yaygın olarak kullanılmaktadır. Bunun sonucunda uygulamaya yönelik olarak mikroişlemcili sistemlerin donanım ve yazılımının tasarlanması ve gerçekleştirilmesi çeşitli alanlardaki mühendislik çalışmalarında önemli bir yer tutmaktadır. Endüstriyel uygulamalar, kendisi de mikroişlemci temelli bir sistem olan kişisel bilgisayarlar ile gerçekleştirilebilmektedir. İşaret işleme uygulamalarına özel mimariye sahip mikroişlemciler olarak tanımlanabilen Sayısal İşaret İşleyiciler (DSP), mikroişlemci temelli sistemler ile beraber kullanılmaktadır. Robot sistemleri gibi mekanik ile elektroniği birleştiren (Mekatronik) uygulamalarında mikrodenetleyiciler kullanılmaktadır.

### 18.1. Giriş Uygulamaları

Mikroişlemci temelli bir sisteme dışarıdan bilgi girişi yapmanın en basit yöntemi yapılan işe uygun bir anahtar kullanmaktır. ASCII tablosunda yer alan alfabetik veya sayısal bilgiler gibi karmaşık bilgileri girmek için ise bir anahtar takımı, klavye kullanılır. Günümüzde yaygın olarak kullanılan IBM uyumlu kişisel bilgisayar sistemleri gibi grafik işlem yeteneği varsa fare veya benzeri grafik işaretçiler kullanılarak, parmak izi veya insan sesi ile bilgi girişi yapılabilir.

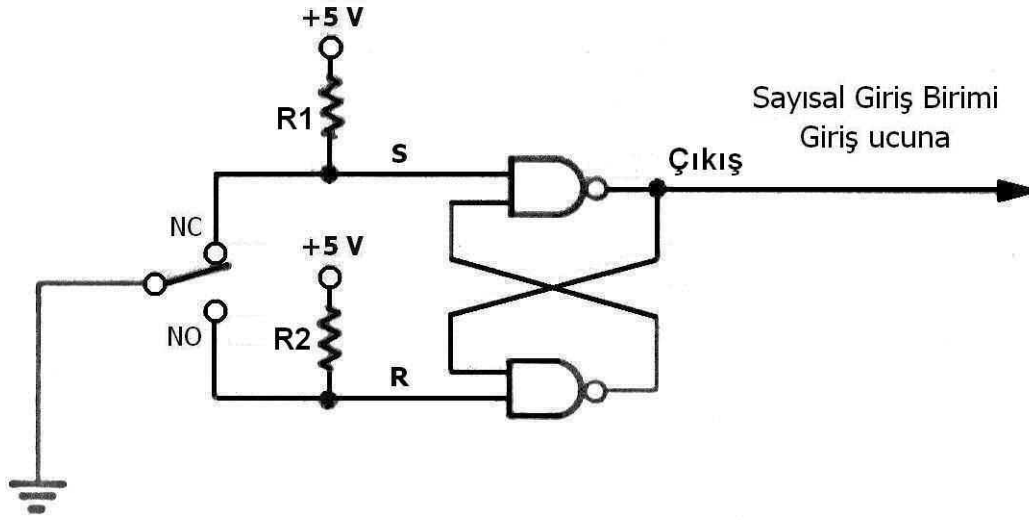


Şekil 18-1 Bir anahtar ile oluşturulmuş fiziksel giriş devresi

Aşağıda giriş işlemini gerçekleştirmek üzere tasarlanmış bir program parçası verilmiştir.

```
LDAA 8000H ; Sayısal giriş biriminin okunması
ANDA #80H ; Giriş biriminin 8. ucundan giriş yapılıyor.
BEQ KAP1 ; 8. uç "0" ise anahtar kapatılmıştır.
.... ; anahtar açık ise yapılacaklar burada yer alır.
```

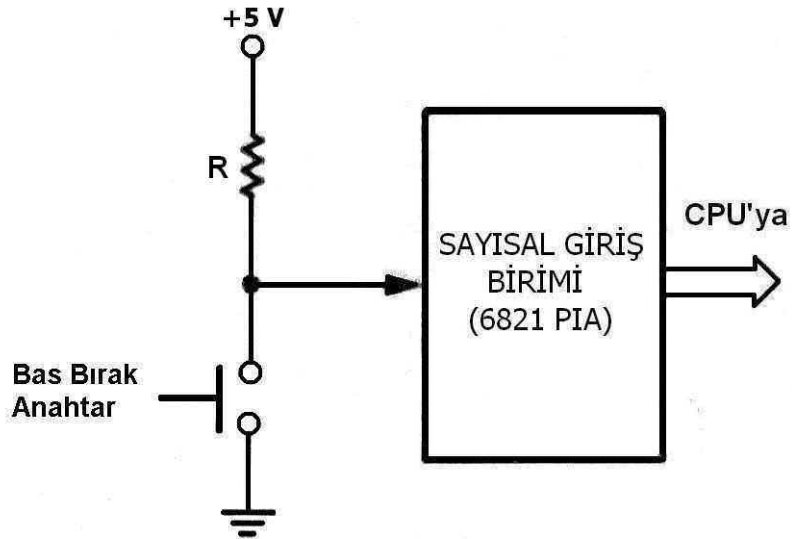
```
KAP1: .... ; anahtar kapalı ise yapılacaklar burada yer alır.
```



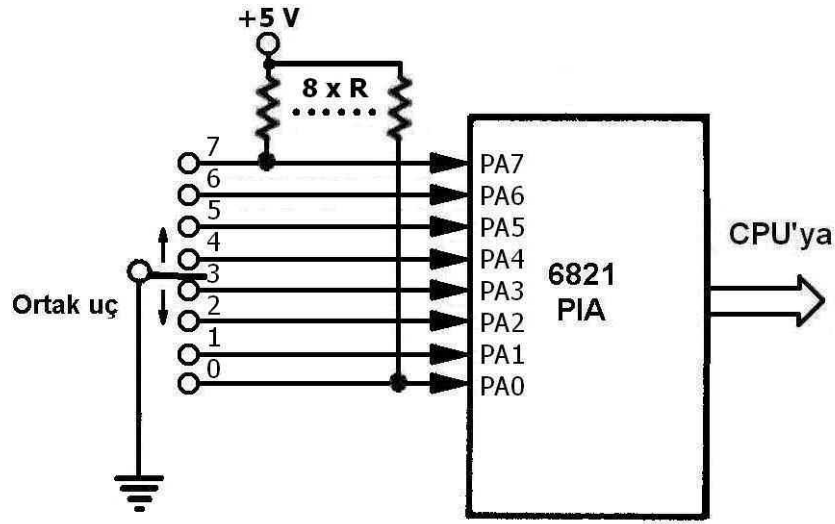
Şekil 18-2 Anahtarın çınlama etkisinin önlenmesi ile giriş devresi

ANA: ....  
 JSR BEK1 ;çınlamanın geçmesi için bekleme alt programı  
 LDAA 8000H ;Sayısal giriş biriminin okunması  
 ANDA #80H ;Giriş biriminin b7 ucu bayraklarda elde edilir.  
 BEQ KAP1 ;uç "0" ise sonuç sıfır Z=1, anahtar kapatılmıştır.  
 .... ;anahtar açık ise yapılacaklar burada yer alır.

KAP1: .... ;anahtar kapalı ise yapılacaklar burada yer alır.  
 JMP ANA ;ana program döngüsü



Şekil 18-3 Bir bas bırak anahtar ile oluşturulmuş fiziksel giriş devresi



Şekil 18-4 Çok konumlu anahtar ile oluşturulmuş fiziksel giriş devresi

```

ANA:  ....
      JSR   BEK1    ;çınlamanın geçmesi için bekleme alt programı
      LDAB  #8      ;Anahtar konumu sayacı (burada 8 anahtar var)
      LDAA  8000H   ;Sayısal giriş biriminin okunması
L1:   ASLA      ;Giriş biriminin ucunun durumu C bayrağına alınır.
      BCC   KAP7    ;uç "0" ise anahtar kapatılmıştır.
      DECB      ;anahtar konumu sayacını azalt
      BNE   L1      ; anahtarın 8 konumu bitene kadar devam.
      ....          ;anahtar açık ise yapılacaklar burada yer alır.

KAP7: CMPB  #7      ;anahtarın konumu sayacı "7" mi?
      BNE   KAP6
      ....          ; "7" ise yapılacak işlemler.
      JMP   ANA      ;ana program döngüsüne git
KAP6: CMPB  #6      ;anahtarın konumu sayacı "6" mı?
      BNE   KAP5
      ....          ; "6" ise yapılacak işlemler.
      JMP   ANA      ;ana program döngüsüne git
      ....
      ....
KAP1: CMPB  #1      ;anahtarın konumu sayacı "1" mi?
      BNE   KAP0
      ....          ; "1" ise yapılacak işlemler.
      JMP   ANA      ;ana program döngüsüne git
KAP0: ....          ; "0" ise yapılacak işlemler.
      JMP   ANA      ;ana program döngüsüne git
;çınlama etkisini önlemek için 10 ms gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 1249 x (4+4) + 5 = 10000 sistem saati
;sistem saati 1 MHz ise 10000 x 1us = 10 ms
BEK1: LDX   #1249   ;döngü sayacının ilk değeri
L2:   DEX      ;döngü sayacının
      BNE   L2      ;sıfır olana kadar azaltılması.
      RTS        ;ana programa geri dönülmesi.

```

Aynı programın tabloya bakma (lookup table) yöntemiyle tasarımı:

```

ANA:  ....
      JSR   BEK1   ;çınlamanın geçmesi için bekleme alt programı
      LDX   #8     ;Anahtar konumu sayacı (burada 8 anahtar var)
      LDAA  8000H   ;Sayısal giriş biriminin okunması
L1:   ASLA                ;Giriş biriminin ucunun durumu C bayrağına alınır.
      BCC   KAPX     ;uç "0" ise anahtar kapatılmıştır.
      DEX                ;anahtar konumu sayacını azalt
      BNE   L1        ; anahtarın 8 konumu bitene kadar devam.
      ....           ;anahtar açık ise yapılacaklar burada yer alır.
KAPX: DEX
      LDAA  X,TUS     ;anahtarın ASCII kodu ile
      JMP   ANA        ;ana program döngüsüne geri dön
;çınlama etkisini önlemek için 10 ms gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 1249 x (4+4) + 5 = 10000 sistem saati
;sistem saati 1 MHz ise 10000 x 1us = 10 ms
BEK1: LDX   #1249     ;döngü sayacının ilk değeri
L2:   DEX                ;döngü sayacının
      BNE   L2          ;sıfır olana kadar azaltılması.
      RTS                ;ana programa geri dönülmesi.
;TUS adlarının bulunduğu tablo
TUS:  DFB   31H,32H,33H,34H,35H,36H,37H,38H
      END

```

Çıkış Dosyası:

```

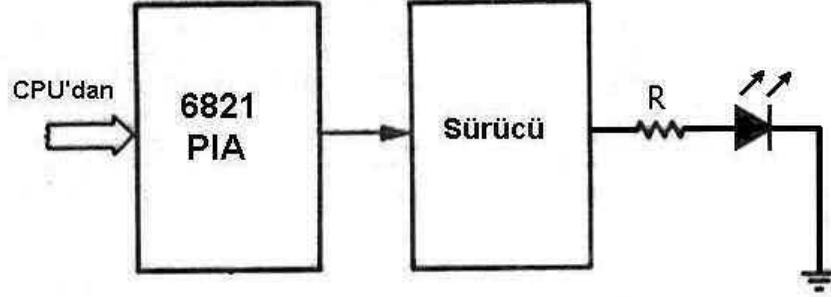
0100          ORG   100H
0100 8E007F BASLA:LDS #007FH ;Yığın işaretçi RAM'in tepe adresine
0103 01      ANA:  NOP        ;ana programda yapılacaklar burada yapılır.
0104 BD011A   JSR   BEK1     ;çınlamanın geçmesi için bekleme alt programı
0107 CE0008   LDX   #8       ;Anahtar konumu sayacı (burada 8 anahtar var)
010A B68000   LDAA  8000H    ;Sayısal giriş biriminin okunması
010D 48      L1:   ASLA      ;Giriş biriminin ucunun durumu C bayrağına alınır.
010E 2404     BCC   KAPX     ;uç "0" ise anahtar kapatılmıştır.
0110 09      DEX                ;anahtar konumu sayacını azalt
0111 26FA     BNE   L1        ; anahtarın 8 konumu bitene kadar devam.
0113 01      NOP            ;anahtar açık ise yapılacaklar burada yer alır.
0114 09      KAPX: DEX
0115 A660     LDAA  TUS,X     ;anahtarın ASCII kodu ile
0117 7E0103   JMP   ANA      ;ana program döngüsüne geri dön
;çınlama etkisini önlemek için 10 ms gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 1249 x (4+4) + 5 = 10000 sistem saati
;sistem saati 1 MHz ise 10000 x 1us = 10 ms
011A CE04E1 BEK1: LDX   #1249 ;döngü sayacının ilk değeri
011D 09      L2:   DEX        ;döngü sayacının
011E 26FD     BNE   L2        ;sıfır olana kadar azaltılması.
0120 39      RTS            ;ana programa geri dönülmesi.
;TUS adlarının bulunduğu tablo
0060          ORG   60H
0060 31323334 TUS: DFB   31H,32H,33H,34H
0064 35363738 DFB   35H,36H,37H,38H
0000          END

```

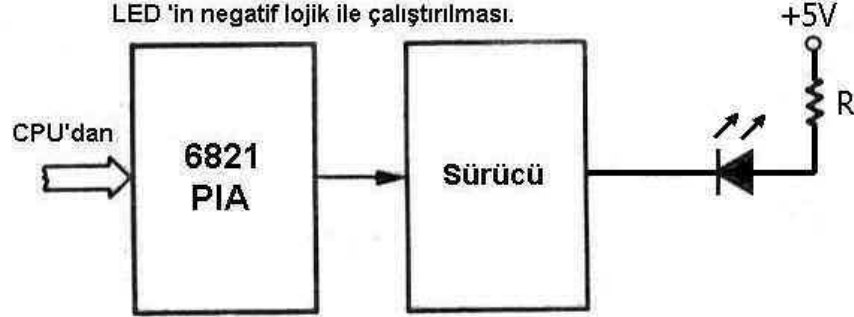
## 18.2. Çıkış Uygulamaları

Uygulamaların amacına bağlı olarak insanın duyu organlarının algılama kapasitesine uygun çıkış cihazları kullanılır. Bunlar için görme, duyma, temas vs. gibi algılama alanlarında çok geniş ölçekli mekanizmalar kullanılmaktadır. Ayrıca otomatik kontrol amacıyla giriş biriminden alınan bilgiler değerlendirilip çıkış birimine bağlı motor, ısıtıcı gibi birimler denetlenebilir. Burada yaygın olarak kullanılan çıkış birimlerinin mikroşlemcili donanımları ve yazılımları temel düzeyde ele alınacaktır.

LED 'in pozitif lojik ile çalıştırılması.



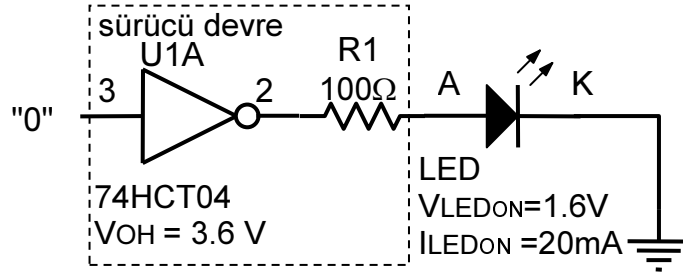
LED 'in negatif lojik ile çalıştırılması.



Şekil 18-5 LED kullanarak yapılan çıkış uygulamaları

Sürücü çıkışındaki akımı sınırlamak için LED'e seri R direnci kullanılır. Bu direncin hesabı için Şekil 18-6 ve Şekil 18-7'de verilen eşdeğer devreler kullanılabilir. Şekil 18-6'da verilen devrede LED'in pozitif lojik ile sürülmesi durumunda bir eşdeğer devre verilmiştir. Bu devrede kapı çıkışının lojik "1" gerilim seviyesinde olduğu  $V_{OH}$  gerilimi uygulandığında LED'in ışık yaydığı durum sağlanır. LED'in ışık yayması için gerekli olan gerilim ve akımın tipik değerleri Tablo 18-1'de verilmiştir. Bu tablodan elde edilen standart kırmızı LED için değerler kullanılarak R1 direncinin değeri, çevre denklemi yazılarak kolayca bulunabilir.

$$R1 = \frac{V_{OH} - V_{LEDON}}{I_{LEDON}} = \frac{3.6 - 1.6}{20 \cdot 10^{-3}} = 100\Omega$$



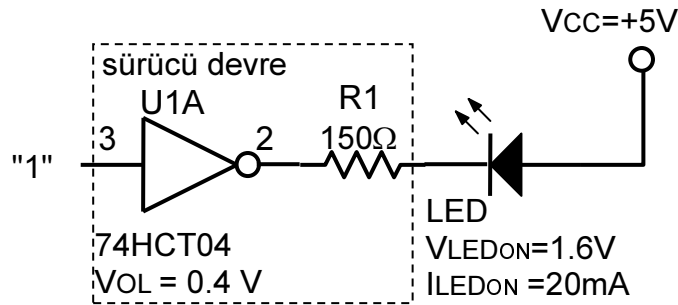
Şekil 18-6 LED'in pozitif lojik ile çalıştırılması için eşdeğer devre

Tablo 18-1 Değişik renkteki LED'lerin DC özellikleri

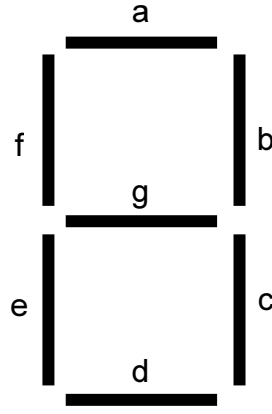
LED	$V_{LEDON}$	$I_{LEDON}$
Düşük Akımlı Kırmızı	1.8 V	2 mA
Standart Kırmızı	1.6 V	20 mA
Parlak Kırmızı	2.2 V	20 mA
Standart Sarı	2.2 V	10 mA
Standart Yeşil	2.3 V	10 mA

Şekil 18-7'de verilen devrede LED'in negatif lojik ile sürülmesi durumunda bir eşdeğer devre verilmiştir. Bu devrede kapı çıkışının lojik "0" gerilim seviyesinde olduğu  $V_{OL}$  gerilimi uygulandığında LED'in ışık yaydığı durum sağlanır. LED'in ışık yayması için gerekli olan gerilim ve akımın tipik değerleri Tablo 18-1'den elde edilen standart kırmızı LED için değerler kullanılarak R1 direncinin değeri, çevre denklemi yazılarak kolayca bulunabilir.

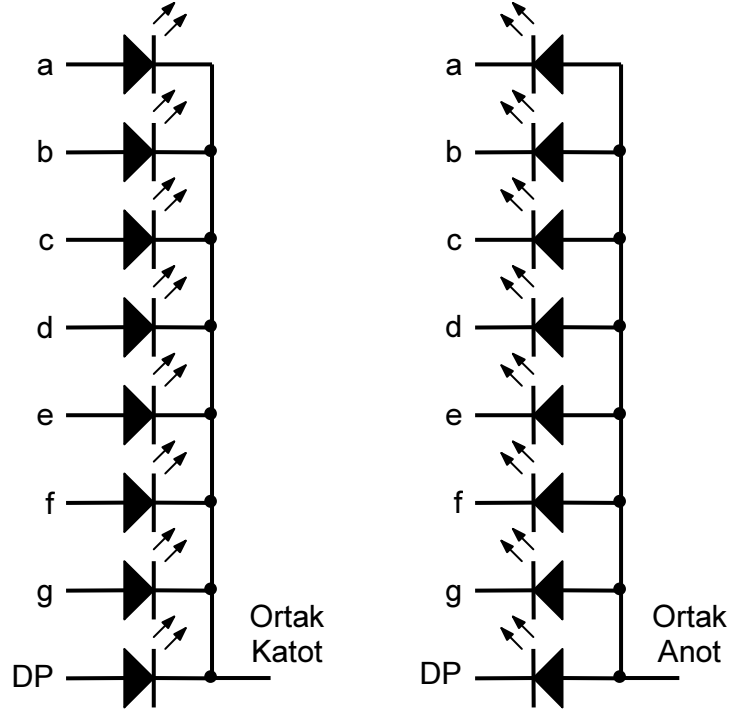
$$R1 = \frac{V_{CC} - V_{OL} - V_{LEDON}}{I_{LEDON}} = \frac{5 - 0.4 - 1.6}{20 \cdot 10^{-3}} = 150\Omega$$



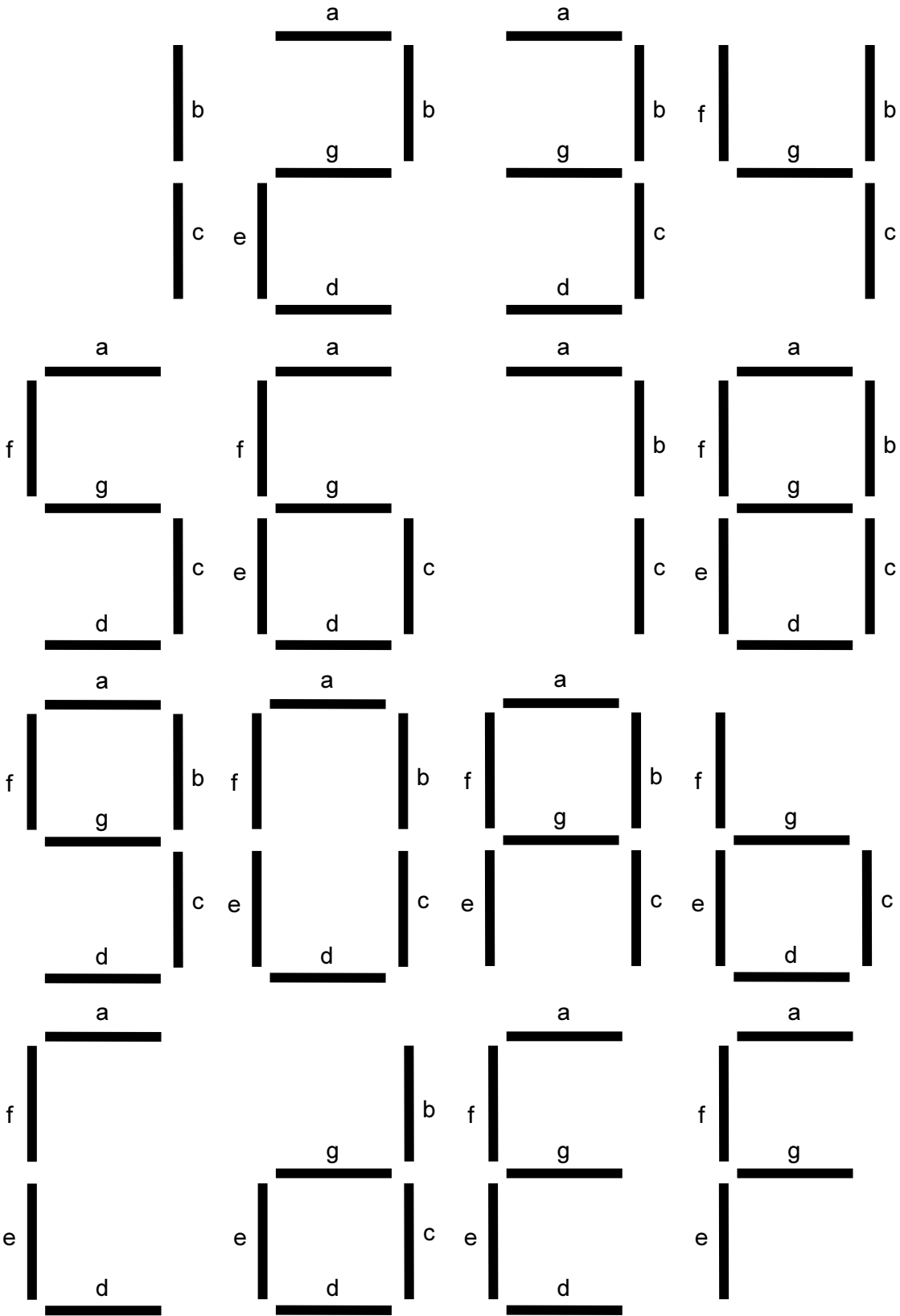
Şekil 18-7 LED'in negatif lojik ile çalıştırılması için eşdeğer devre



Şekil 18-8 7-parça LED gösterge parça tanımları



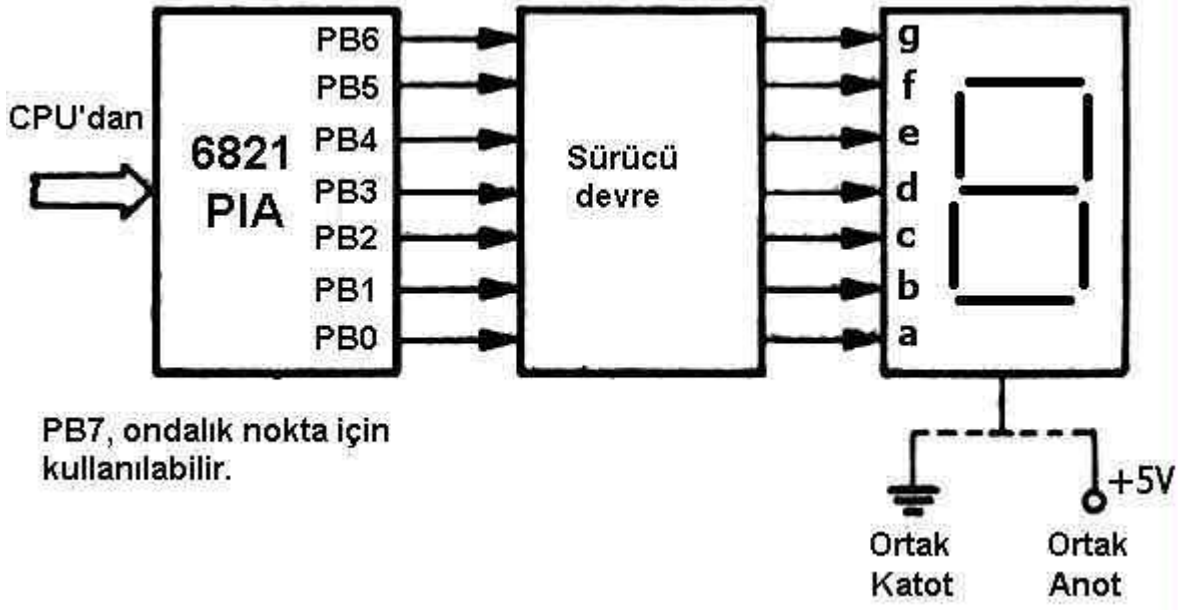
Şekil 18-9 7-parça LED gösterge için Ortak Katot ve Ortak Anot bağlantı şekli



Şekil 18-10 7-parça LED göstergede sayıların şekil ve tanımları



7-parça göstergiyi mikroişlemci temelli bir sisteme bağlamak için bir blok diyagramı Şekil 18-11'de verilmiştir. Blok diyagramda paralel G/Ç biriminin B portu çıkış olarak tanımlanır. Şekil 18-8'de görülen şekillerin göstergede oluşturulması için B portuna yazılan 8-bit değer sürücü devre kullanılarak 7-parça göstergeye Tablo 18-2'de verilen dönüşüm tablosunda belirtilen veriler uygulanır.



Şekil 18-11 Paralel G/Ç tümleşik devresi ile 7-parça LED gösterge parça bağlantısı

Tablo 18-2 7-parça gösterge için veri dönüşüm tablosu

Sayı	Ortak Anot			Ortak Katot		
	g	f	e	d	c	b
0	1	0	0	0	0	0
1	1	1	1	1	0	0
2	0	1	0	0	1	0
3	0	1	1	0	0	0
4	0	0	1	1	0	0
5	0	0	1	0	0	1
6	0	0	0	0	0	1
7	1	1	1	1	0	0
8	0	0	0	0	0	0
9	0	0	1	1	0	0

Aşağıda Şekil 18-11'deki blok diyagram kullanarak ortak katot 7-parça LED göstergeye çıkış işlemini gerçekleştirmek üzere tasarlanmış iki örnek program parçası verilmiştir.

```
LDAA #3FH ;göstergeye "0" bilgisi için kod
STAA 8002H ;Sayısal çıkış biriminin B portuna yazılması
LDAA #06H ;göstergeye "1" bilgisi için kod
STAA 8002H ;Sayısal çıkış biriminin B portuna yazılması
```

Aşağıda verilen örnek program parçasında göstergede sırayla 0'dan 9'a kadar olan sayılar gösterilmektedir.

```

ANA:  LDAA  #3FH    ;göstergeye "0" bilgisi için kod
      STAA  8002H    ;Sayısal çıkış biriminin B portuna yazılması
      JSR   BEK1     ;yazılanı gözün algılaması için bekleme alt programı
      LDAA  #06H    ;göstergeye "1" bilgisi için kod
      STAA  8002H    ;Sayısal çıkış biriminin B portuna yazılması
      JSR   BEK1     ;yazılanı gözün algılaması için bekleme alt programı
      ....
      LDAA  #67H    ;göstergeye "9" bilgisi için kod
      STAA  8002H    ;Sayısal çıkış biriminin B portuna yazılması
      JSR   BEK1     ;yazılanı gözün algılaması için bekleme alt programı
      JMP   ANA      ;ana program döngüsüne git
;Göstergenin okunabilmesi için 1 sn gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 4095 x (4+4) + 5 = 32768 sistem saati
;sistem saati 32.768 kHz ise 32768 x 30.518us = 1 sn
BEK1: LDX   #4095    ;döngü sayacının ilk değeri
L2:   DEX                ;döngü sayacının
      BNE   L2         ;sıfır olana kadar azaltılması.
      RTS                ;ana programa geri dönülmesi.
      END

```

Aynı programın tabloya bakma (lookup table) yöntemiyle tasarımı:

```

BASLA: LDS  #007FH   ;Yığın işaretçi RAM'in tepe adresine
ANA:  LDAB  #10      ;göstergeye yazılacak sayı adedi
      LDX   #GTBL    ;gösterge değerleri için tablonun başlangıç adresi
L1:   LDAA  0,X      ;gösterge bilgisi için kodun tablodan alınması
      STAA  8002H    ;Sayısal çıkış biriminin B portuna yazılması
      STX   60H      ;X dizin yazmacının bellekte saklanması
      JSR   BEK1     ;yazılanı gözün algılaması için bekleme alt programı
      LDX   60H      ;X dizin yazmacının bellekten geri alınması
      INX                ;bir sonraki gösterge bilgisi adresi
      DECB                ;sayı adedi sayacını azalt
      BNE   L1       ;sayıların gösterilmesi bitene kadar devam.
      JMP   ANA      ;ana program döngüsüne geri dön
;Göstergenin okunabilmesi için 1 sn gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 1023 x (4+4) + 5 = 32768 sistem saati
;fXTAL= 32.768kHz ise sistem saati 8.192kHz olur. Gecikme 8192 x 122.07us = 1 sn
BEK1: LDX   #1023    ;döngü sayacının ilk değeri
L2:   DEX                ;döngü sayacının
      BNE   L2         ;sıfır olana kadar azaltılması.
      RTS                ;ana programa geri dönülmesi.
;Ortak katot gösterge kodlarının bulunduğu tablo
GTBL: DFB   3FH,06H,5BH,4FH,66H,"0","1","2","3","4"
      DFB   6CH,7CH,07H,7FH,67H,"5","6","7","8","9"
      END

```

Algoritma kullanmak programın insan düşüncesine yakın, kısa, daha anlaşılır ve esnek olmasını sağlamıştır.

Çıkış Dosyası:

```

0100          ORG    100H
0100 8E007F BASLA: LDS    #007FH    ;Yığın işaretçi RAM'in tepe adresine
0103 C60A  ANA:  LDAB   #10      ;göstergeye yazılacak sayı adedi
0105 CEF100          LDX    #GTBL    ;gösterge değerleri için tablonun başlangıç adresi
0108 A600  L1:   LDAA   0,X      ;gösterge bilgisi için kodun tablodan alınması
010A B78002          STAA  8002H    ;Sayısal çıkış biriminin B portuna yazılması
010D DF60          STX    60H      ;X dizin yazmacının bellekte saklanması
010F BD011B          JSR    BEK1    ;yazılanı gözün algılaması için bekleme alt programı
0112 DE60          LDX    60H      ;X dizin yazmacının bellekten geri alınması
0114 08          INX          ;bir sonraki gösterge bilgisi adresi
0115 5A          DECB          ;sayı adedi sayacını azalt
0116 26F0          BNE    L1      ;sayıların gösterilmesi bitene kadar devam.
0118 7E0103          JMP    ANA    ;ana program döngüsüne geri dön
;Göstergenin okunabilmesi için 1 sn gecikme sağlayan alt program.
;6802 için toplam gecikme = 3 + 1023 x (4+4) + 5 = 32768 sistem saati
;fXTAL= 32.768kHz ise sistem saati 8.192kHz olur. Gecikme 8192 x 122.07us = 1 sn
011B CE03FF BEK1: LDX    #1023    ;döngü sayacının ilk değeri
011E 09  L2:   DEX          ;döngü sayacının
011F 26FD          BNE    L2      ;sıfır olana kadar azaltılması.
0121 39          RTS          ;ana programa geri dönülmesi.
;Gösterge kodlarının bulunduğu tablo
F100          ORG    0F100H
F100 3F065B4F66 GTBL: DFB    3FH,06H,5BH,4FH,66H ;"0","1","2","3","4"
F105 6C7C077F67          DFB    6CH,7CH,07H,7FH,67H ;"5","6","7","8","9"
0000          END

```

### 18.3. Zamanlama Uygulamaları

;10 ms gecikme sağlayan alt program.

;f<sub>XTAL</sub> = 4 MHz ise sistem saati 1 MHz dir ve süre 10000 x 1us = 10 ms

;6802 için toplam gecikme = 3 + TE (toplam sistem saati) x (4+4) + 5 ≅ 10000

; TE = (10000 – 8 )/8 = 1249

```

G10ms:      LDX    #1249    ;döngü sayacının ilk değeri
L2:         DEX          ;döngü sayacının
           BNE    L2      ;sıfır olana kadar azaltılması.
           RTS          ;ana programa geri dönülmesi.

```

;6802 için gerçek toplam gecikme = 3 + 1249 x (4+4) + 5 = 10000 sistem saati  
= 10000 x (1) us = 10 ms

;1 sn gecikme sağlayan alt program.

;f<sub>XTAL</sub> = 32.768kHz ise sistem saati 8.192 kHz dir ve gecikme 8192 x 122 us = 1 sn

;6802 için toplam gecikme = 3 + TE (toplam sistem saati) x (4+4) + 5 = 8192

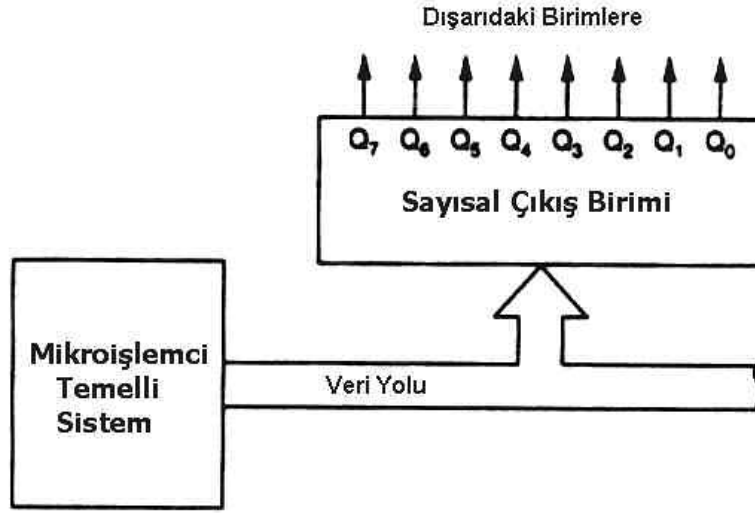
; TE = (8192 – 8 )/8 = 1023

```

G10ms:      LDX    #1023    ;döngü sayacının ilk değeri
L2:         DEX          ;döngü sayacının
           BNE    L2      ;sıfır olana kadar azaltılması.
           RTS          ;ana programa geri dönülmesi.

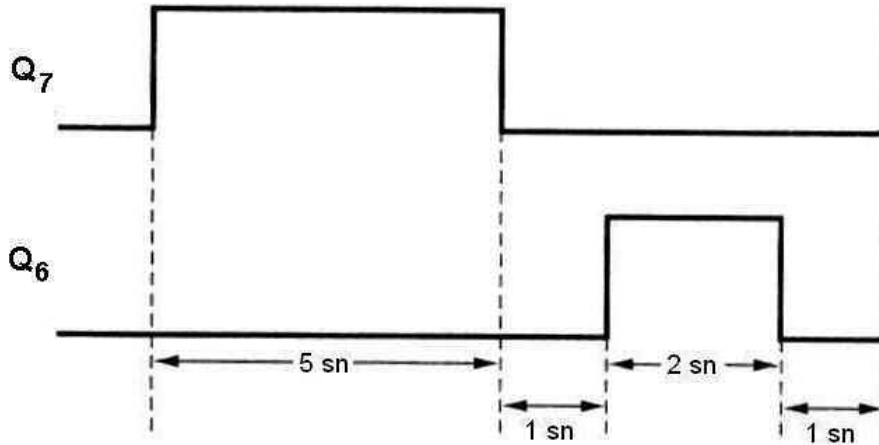
```

;6802 için gerçek toplam gecikme = 3 + 1023 x (4+4) + 5 = 8192 sistem saati  
= 8192 x (1/8192) us = 1sn



Şekil 18-12 Mikroişlemci yazılımı kullanılarak zamanlama işaretlerinin üretilmesi

**Örnek Pr. 18-1** Aşağıda verilen darbe diyagramını oluşturan zamanlama yazılımını tasarlayınız:



**Çözüm:**

;sayısal çıkış biriminin port adresi 8000H olsun.

ANA: LDAB #\$80  
 STAB \$8000 ; Q7="1"  
 LDX #5117 ; 5 sn gecikme için 5117  
 JSR G1 ; gecikme altprogramı

LDAB #\$00  
 STAB \$8000 ; Q0-Q7="0"  
 LDX #1021 ; 1 sn gecikme için 1021  
 JSR G1 ; gecikme altprogramı

LDAB #\$40  
 STAB \$8000 ; Q6="1"  
 LDX #2045 ; 2 sn gecikme için 2045  
 JSR G1 ; gecikme altprogramı

LDAB #\$00  
 STAB \$8000 ; Q0-Q7="0"  
 LDX #1021 ; 1 sn gecikme için 1021  
 JSR G1 ; gecikme altprogramı

JMP ANA ;darbe zamanlamasını tekrarlamak için başa dön

;1,2,5 sn gecikme sağlayan alt program.

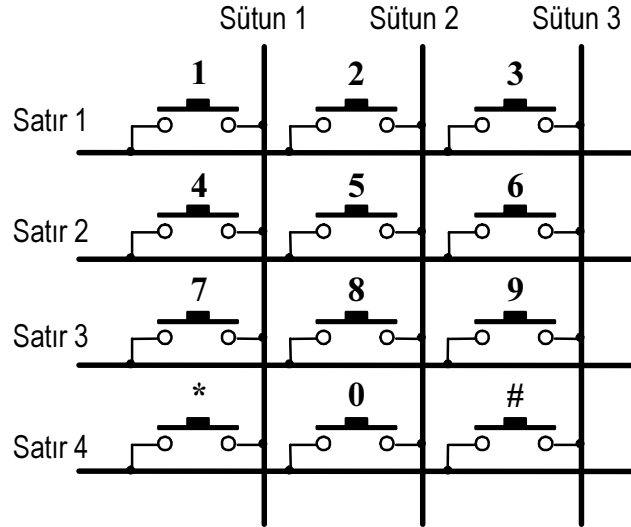
;6802 için toplam gecikme = 5 + 3 + 9 + 1023 x (4+4) + 5 + 2 = 32768 sistem saati

;f<sub>XTAL</sub> = 32.768kHz ise sistem saati 8.192 kHz ise 8192 x 122.07us = 1sn

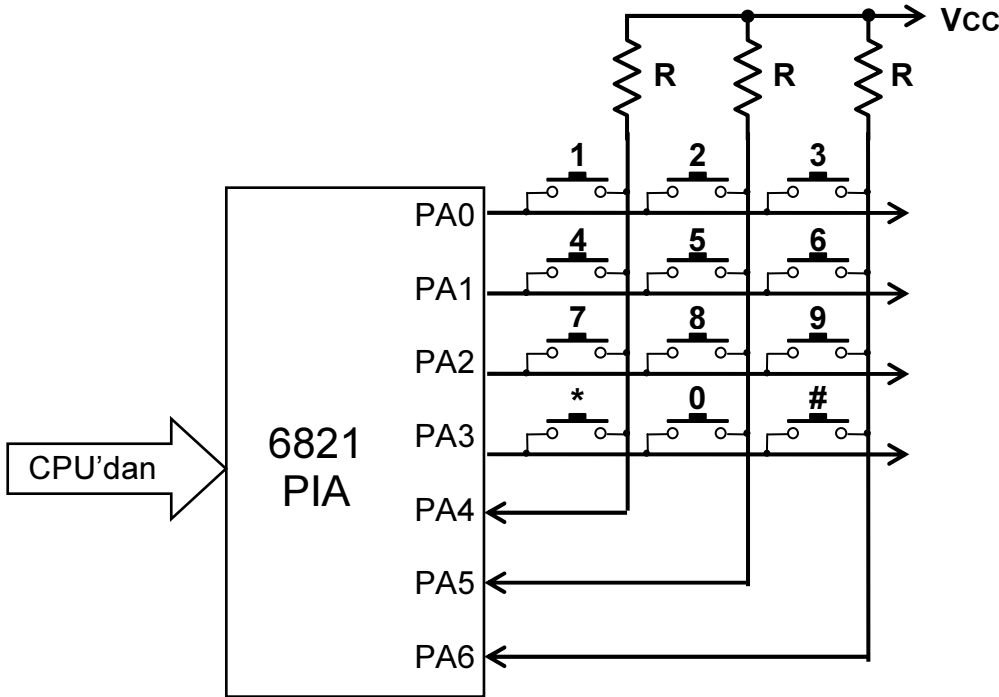
;döngü sayacının ilk değeri 1sn=1021, 2sn=2045, 5sn=5117

G1: DEX ;döngü sayacının  
BNE G1 ;sıfır olana kadar azaltılması.  
RTS ;ana programa geri dönülmesi.

## 18.4. Giriş/Çıkış Uygulamaları



Şekil 18-13 Matris tipi tuş takımı bağlantı diyagramı



Şekil 18-14 Paralel Giriş/Çıkış tümleşik devresine matris tipi tuş takımı bağlanması.

ANA: ....  
JSR BEK1 ;çinlamanın geçmesi için 10 ms gecikme alt programı  
CLR 8000H ;Sayısal çıkış birimine "0000" yazılması  
LDAA 8000H ;Sayısal giriş biriminin okunması  
ANDA #01110000B ;PA4-PA6 uçlarından okunan verinin elde edilmesi  
CMPA #01110000B ;Herhangi bir tuşa basıldı mı?  
BEQ ANA ;Tuşa basılmadıysa ana program döngüsüne git

SAT1:

LDA #00001110B ;Basılan tuş 1.satırda mı?  
 STAA 8000H ;Sayısal çıkış birimine "1110" yazılması  
 LDA 8000H ;Sayısal giriş biriminin okunması  
 ANDA #01110000B ;PA4-PA6 uçlarından okunan verinin elde edilmesi

CPMA #01100000B ;Basılan tuş "1" mi?  
 BNE S1K2  
 .... ;Basılan tuş "1" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S1K2: CPMA #01010000B ;Basılan tuş "2" mi?  
 BNE S1K3  
 .... ;Basılan tuş "2" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S1K3: CPMA #00110000B ;Basılan tuş "3" mü?  
 BNE SAT2  
 .... ;Basılan tuş "3" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

SAT2: LDA #00001101B ;Basılan tuş 2.satırda mı?  
 STAA 8000H ;Sayısal çıkış birimine "1101" yazılması  
 LDA 8000H ;Sayısal giriş biriminin okunması  
 AND #01110000B ;PA4-PA6 uçlarından okunan verinin elde edilmesi

CPMA #01100000B ; Basılan tuş "4" mü?  
 BNE S2K2  
 .... ;Basılan tuş "4" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S2K2: CPMA #01010000B ;Basılan tuş "5" mi?  
 BNE S2K3  
 .... ;Basılan tuş "5" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S2K3: CPMA #00110000B ;Basılan tuş "6" mü?  
 BNE SAT3  
 .... ;Basılan tuş "6" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

SAT3:

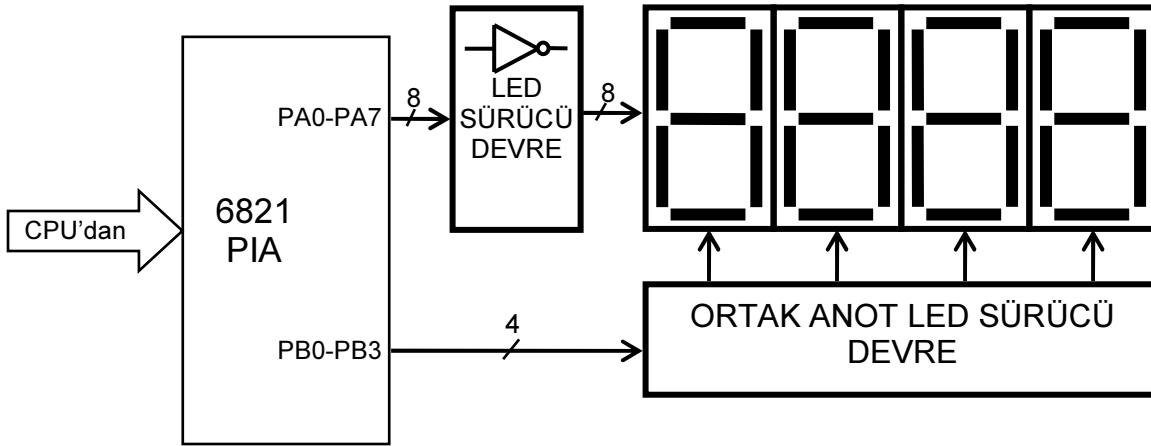
....  
 ....

SAT4: LDA #00000111B ;Basılan tuş 4.satırda mı?  
 STAA 8000H ;Sayısal çıkış birimine "0111" yazılması  
 LDA 8000H ;Sayısal giriş biriminin okunması  
 AND #01110000B ;PA4-PA6 uçlarından okunan verinin elde edilmesi

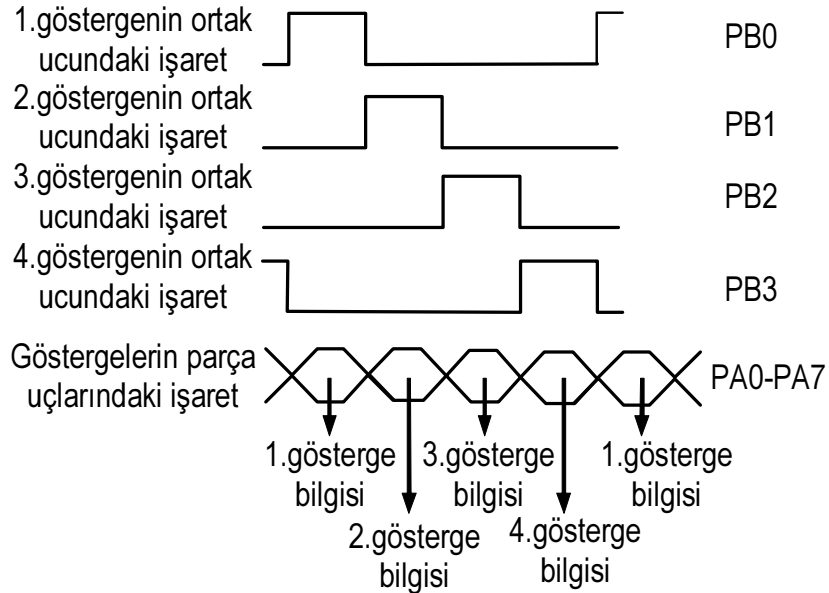
CPMA #01100000B ; Basılan tuş " \* " mı?  
 BNE S4K2  
 .... ;Basılan tuş " \* " ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S4K2: CPMA #01010000B ;Basılan tuş "0" mı?  
 BNE S4K3  
 .... ;Basılan tuş "0" ise yapılacak işlemler.  
 JMP ANA ;ana program döngüsüne git

S4K3: CMPA #00110000B ;Basılan tuş “ # ” mi?  
 BNE SATX  
 .... ;Basılan tuş ” # ” ise yapılacak işlemler.  
 SATX: JMP ANA ;ana program döngüsüne git  
 ;çınlama etkisini önlemek için 10 ms gecikme sağlayan alt program.  
 ;6802 için toplam gecikme = 3 + 1249 x (4+4) + 5 = 10000 sistem saati  
 ;sistem saati 1 MHz ise 10000 x 1us = 10 ms  
 BEK1: LDX #1249 ;döngü sayacının ilk değeri  
 L2: DEX ;döngü sayacının  
 BNE L2 ;sıfır olana kadar azaltılması.  
 RTS ;ana programa geri dönülmesi.  
 END



Şekil 18-15 Çok sayıda göstergenin bir PIA ile sürülmesi



Şekil 18-16 Çok basamaklı 7-parça göstergelerin dinamik tarama ile sürülmesi

Aşağıda Şekil 18-15'deki blok diyagram kullanarak ortak katot 7-parça LED göstergeye çıkış işlemini gerçekleştirmek üzere tasarlanmış iki örnek program parçası verilmiştir.

```

ANA:  LDAA  #00000001B ;göstergenin 1.basamağının seçilmesi
      STAA  8002H      ;Sayısal çıkış biriminin B portuna yazılması
      LDAA  #66H       ;göstergeye "4" bilgisi için kod
      STAA  8000H      ;Sayısal çıkış biriminin A portuna yazılması
      JSR   BEK1       ;yazılanı gözün algılaması için bekleme alt programı
      LDAA  #00000010B ;göstergenin 2.basamağının seçilmesi
      STAA  8002H      ;Sayısal çıkış biriminin B portuna yazılması
      LDAA  #4FH       ;göstergeye "3" bilgisi için kod
      STAA  8000H      ;Sayısal çıkış biriminin A portuna yazılması
      JSR   BEK1       ;yazılanı gözün algılaması için bekleme alt programı
      LDAA  #00000100B ;göstergenin 3.basamağının seçilmesi
      STAA  8002H      ;Sayısal çıkış biriminin B portuna yazılması
      LDAA  #5BH       ;göstergeye "2" bilgisi için kod
      STAA  8000H      ;Sayısal çıkış biriminin A portuna yazılması
      JSR   BEK1       ;yazılanı gözün algılaması için bekleme alt programı
      LDAA  #00001000B ;göstergenin 4.basamağının seçilmesi
      STAA  8002H      ;Sayısal çıkış biriminin B portuna yazılması
      LDAA  #06H       ;göstergeye "1" bilgisi için kod
      STAA  8000H      ;Sayısal çıkış biriminin A portuna yazılması
      JSR   BEK1       ;yazılanı gözün algılaması için bekleme alt programı
      JMP   ANA        ;ana program döngüsüne git

```

Aynı programın tabloya bakma (lookup table) yöntemiyle tasarımı:

```

ANA:  LDAB  #00001000B ;göstergenin 4.basamağının seçilmesi
      LDX   #GTBL      ;gösterge değerleri için tablonun başlangıç adresi
L1:   LDAA  0,X         ;gösterge bilgisi için kodun tablodan alınması
      STAA  8000H      ;Sayısal çıkış biriminin A portuna yazılması
      STAB  8002H      ;basamak seçimi için PIA'nın B portuna yazılması
      STX   60H        ;X dizin yazmacının bellekte saklanması
      JSR   BEK1       ;yazılanı gözün algılaması için bekleme alt programı
      LDX   60H        ;X dizin yazmacının bellekten geri alınması
      INX                ;bir sonraki gösterge bilgisi adresi
      RORB                ;bir sonraki basamağın seçimi
      BNE   L1         ;sayıların gösterilmesi bitene kadar devam.
      JMP   ANA        ;ana program döngüsüne geri dön

```

;Gösterge kodlarının bulunduğu tablo

GTBL: DFB 06H,5BH,4FH,66H ;"1","2","3","4"

;Göstergenin okunabilmesi için 20 ms gecikme sağlayan alt program.

;6802 için toplam gecikme = 5+9+3+ 2497 x (4+4) +5+4 = 20002 sistem saati

;fXTAL= 4 MHz ise sistem saati 1 MHz ise 20002 x 1us  $\cong$  20 ms

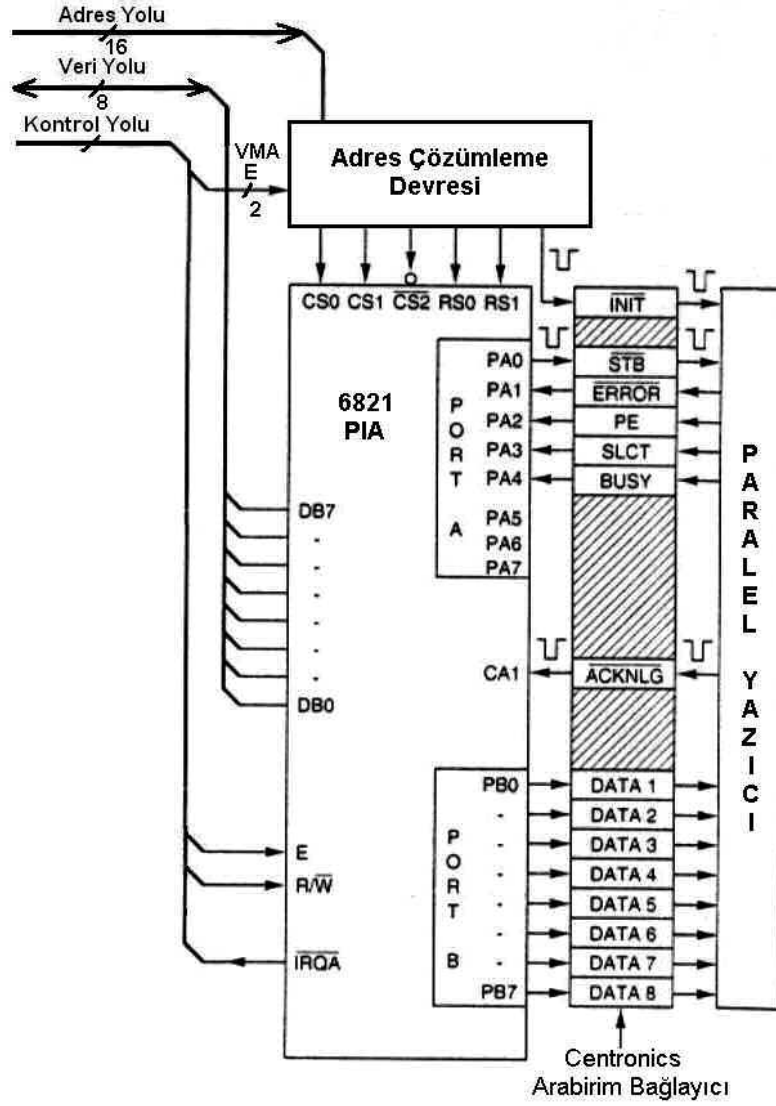
```

BEK1: LDX   #1249      ;döngü sayacının ilk değeri
L2:   DEX                ;döngü sayacının
      BNE   L2         ;sıfır olana kadar azaltılması.
      RTS                ;ana programa geri dönülmesi.
      END

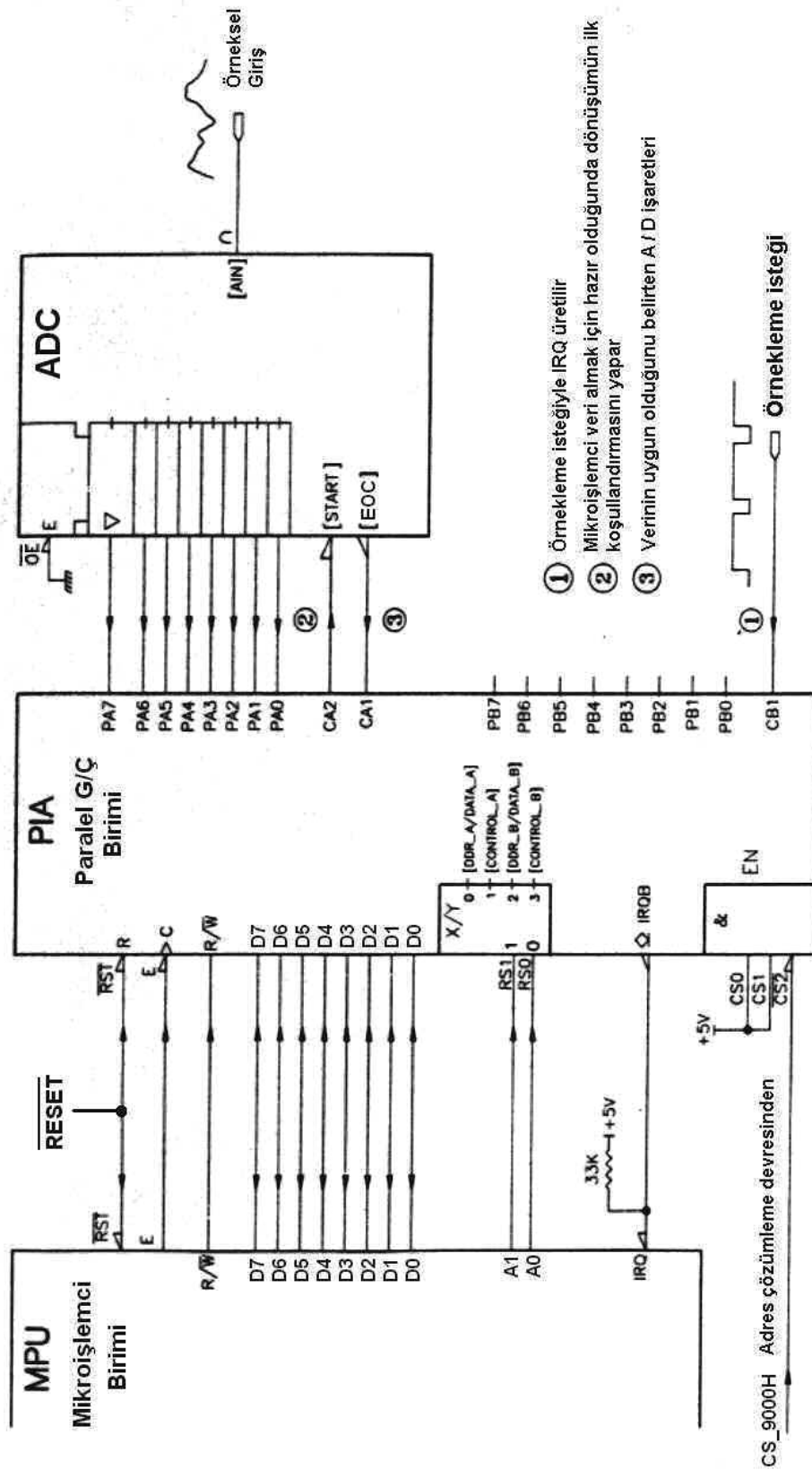
```



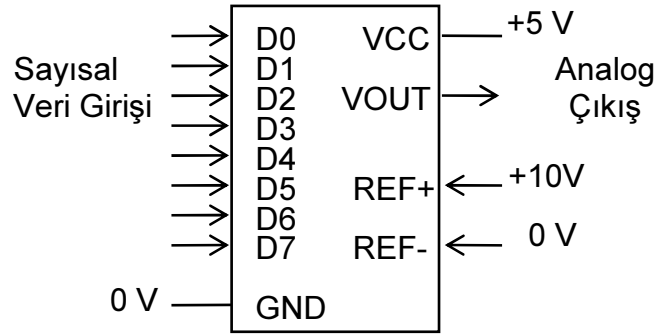
Şekil 18-17’de 6802 mikroişlemci temelli bir sisteme 6821 PIA ile centronics arabirimli paralel yazıcı bağlamak için düşünülmüş bir ek donanım tasarım örneği verilmiştir.



Şekil 18-17 Paralel Giriş/Çıkış tümleşik devresinin yazıcı arabirim için kullanılması



Şekil 18-18 Analog giriş uygulaması



Şekil 18-19 Gerilim çıkışlı ikili girişli sayısal analog dönüştürücü birimi

VOUT çıkış geriliminin sayısal giriş ve REF+ ucuna uygulanan gerileme bağlı ifadesi aşağıda verilen şekildedir.

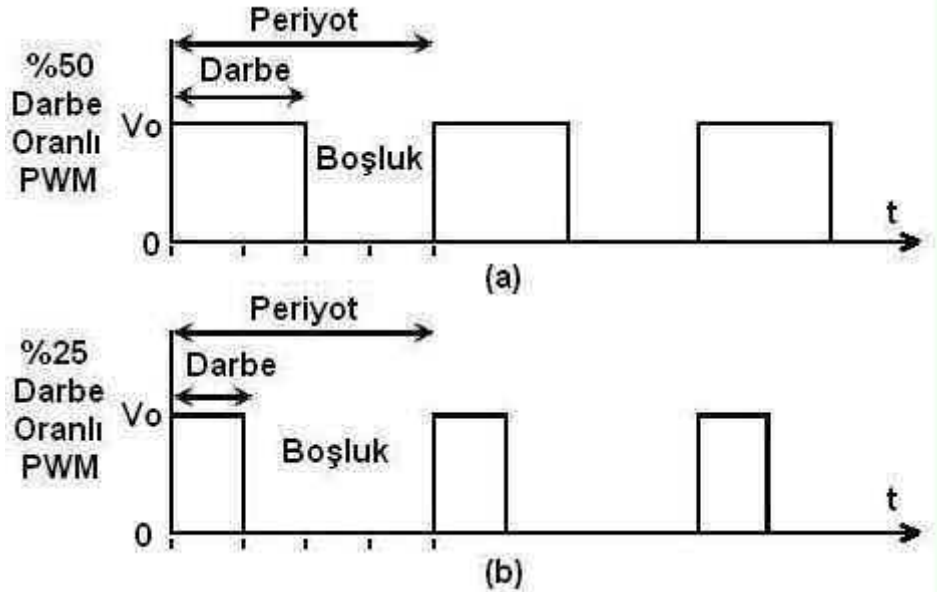
$$V_{OUT} = 10V \left( \frac{D7}{2} + \frac{D6}{4} + \dots + \frac{D0}{256} \right)$$

En küçük analog çıkış gerilimi için D0-D7="0" dır ve çıkış=0V olur.

En büyük analog çıkış gerilimi için D0-D7="1" dir ve çıkış $\approx$ 10V olur.

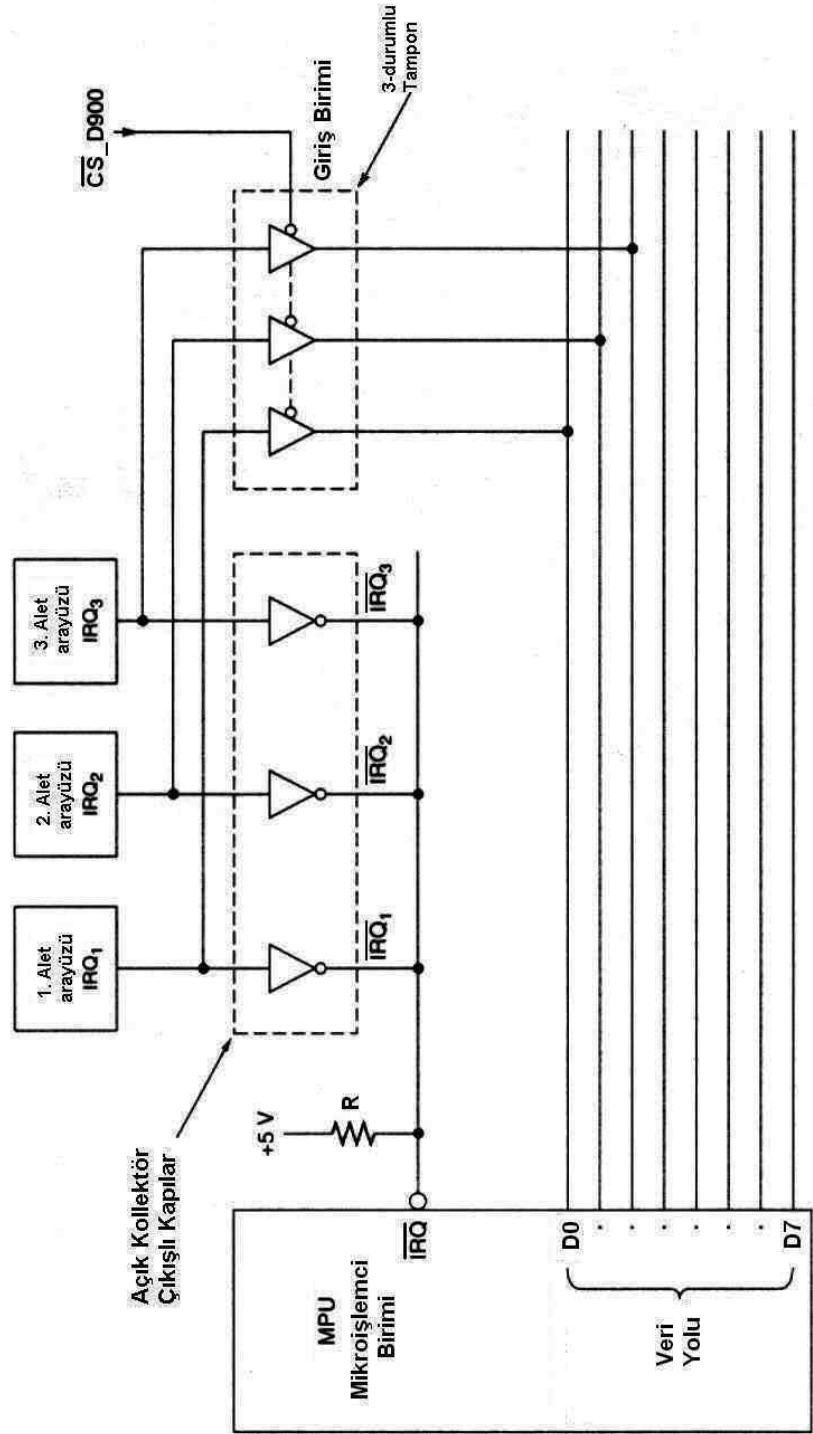
$$V_{OUT} = 10V \left( \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{256} \right) = 10 \frac{255}{256} = 9,96V$$

Mikroişlemci temelli sistemlerde yaygın olarak kullanılan diğer bir sayısal işareti analog işarete dönüştürme yöntemi ise darbe genişlik modülasyonu (PWM) yöntemidir.



Şekil 18-20 Darbe genişliği modülasyonu

## 18.5. Birden Fazla Kesme İşaretinin Mikroişlemciye Uygulanması



Şekil 18-21 Birden fazla kesme işaretinin mikroişlemciye bağlanması

Aşağıda Şekil 18-21'deki yöntemin 6802 mikroişlemcisi için tasarlanmış bir kesme servis programı ve ek yazılımı birlikte verilmiştir.

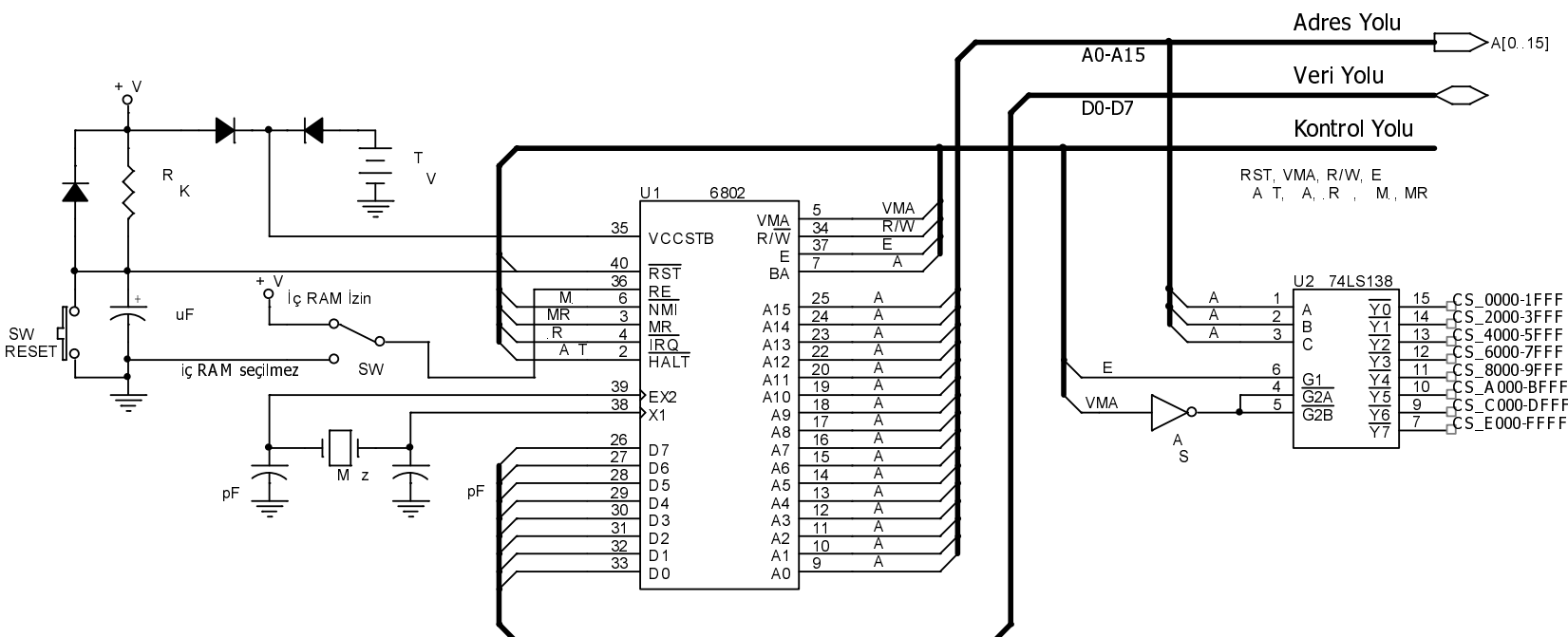
```

BASLA: .... ; başlangıçta yapılacak işlemler.
        .... ;
        CLI ;örtülebilir kesme isteğine izin
ANA: .... ; ana programda yapılacak işlemler.
        .... ;
        JMP ANA ;ana program döngüsüne git

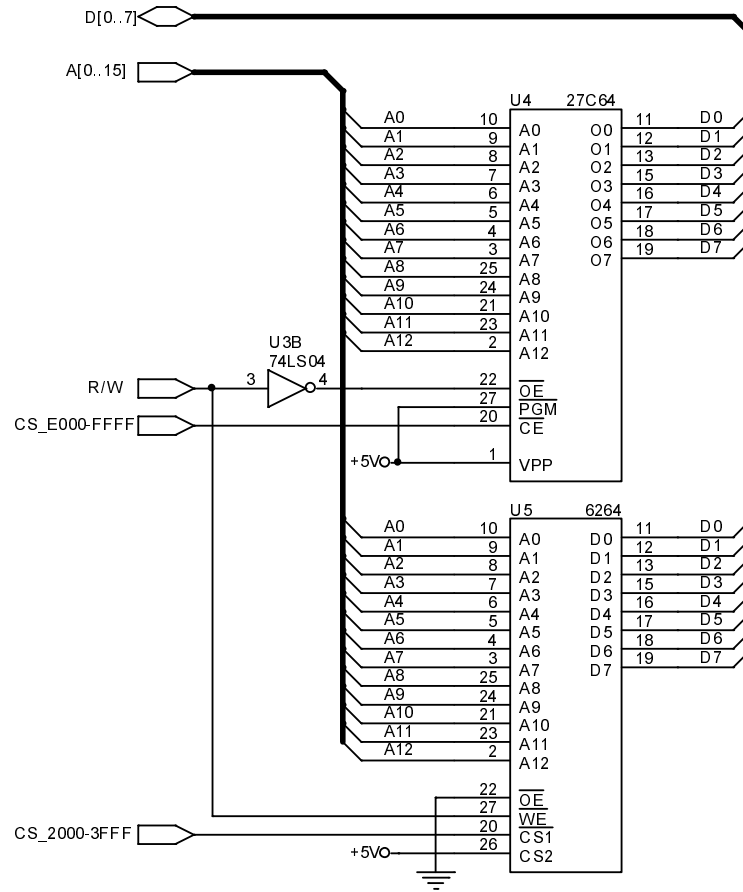
;IRQ örtülebilir kesme isteği için servis programı
KESME: LDAA 0D900H ;kesme kaynağı için giriş biriminin okunması
        AND #00000111B ;D0-D3 uçlarından okunan verinin elde edilmesi
IRQ1: CMPA #00000001B ;kaynak IRQ1 mi?
        BNE IRQ2
        .... ;IRQ1 ise yapılacak işlemler.
        RTI ;ana program döngüsüne geri dön
IRQ2: CMPA #00000010B ;kaynak IRQ2 mi?
        BNE IRQ3
        .... ;IRQ2 ise yapılacak işlemler.
        RTI ;ana program döngüsüne geri dön
IRQ3: CMPA #00000100B ;kaynak IRQ3 mü?
        BNE IRQX
        .... ;IRQ3 ise yapılacak işlemler.
IRQX: RTI ;ana program döngüsüne geri dön
;Vektör Adresleri
ORG 0FFF8H
DWM KESME ;IRQ Örtülebilir Kesme Servis Program Adresi
DWM BASLA ;SWI Yazılım İle Kesme Servis Program Adresi
DWM BASLA ;NMI Örtülemez Kesme Servis Program Adresi
DWM BASLA ;RES Reset, Mikroişlemciyi Yeniden Başlatma Adresi
END

```

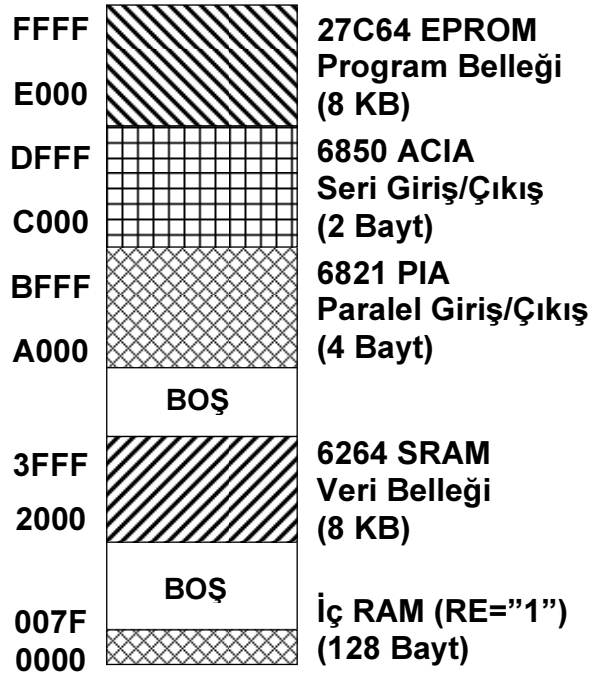
## 18.6. Genel Amaçlı Bir 6802 Mikroişlemcili Sistem Uygulaması



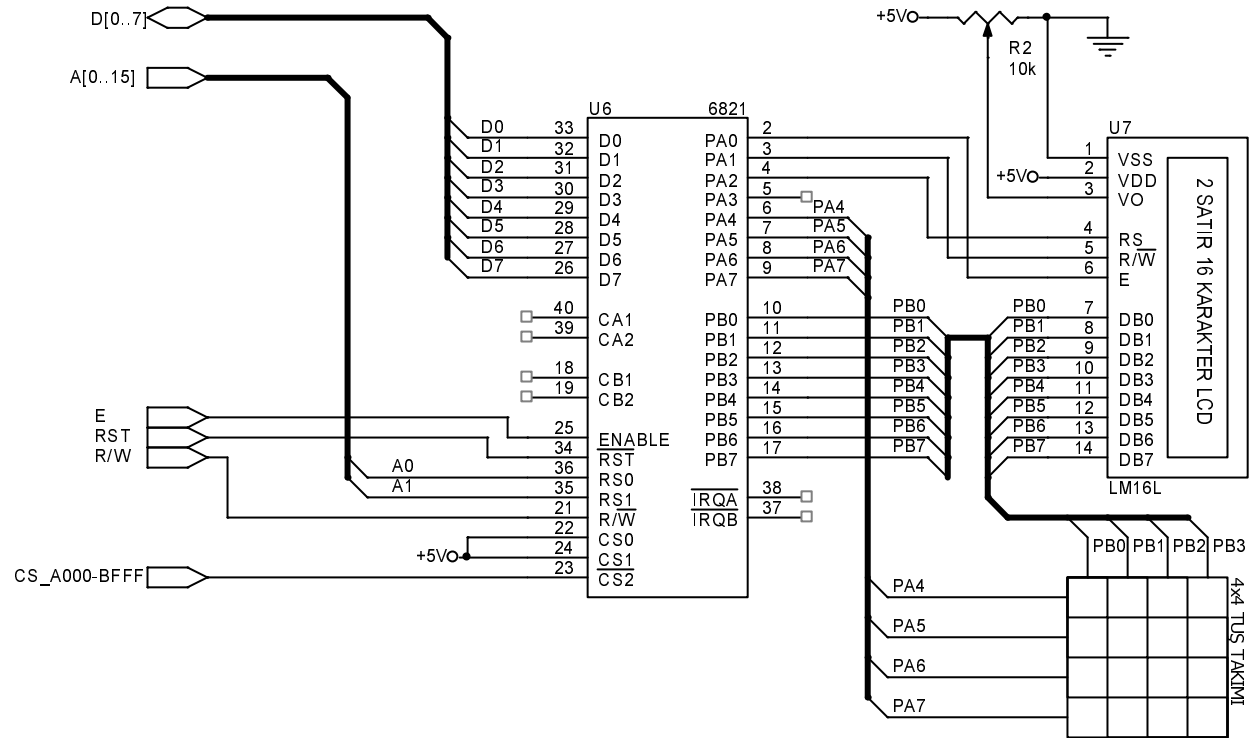
Şekil 18-22 MC6802 Mikroişlemci ve Adres Çözümleme Birimi



Şekil 18-23 Program ve Veri Belleği Birimi

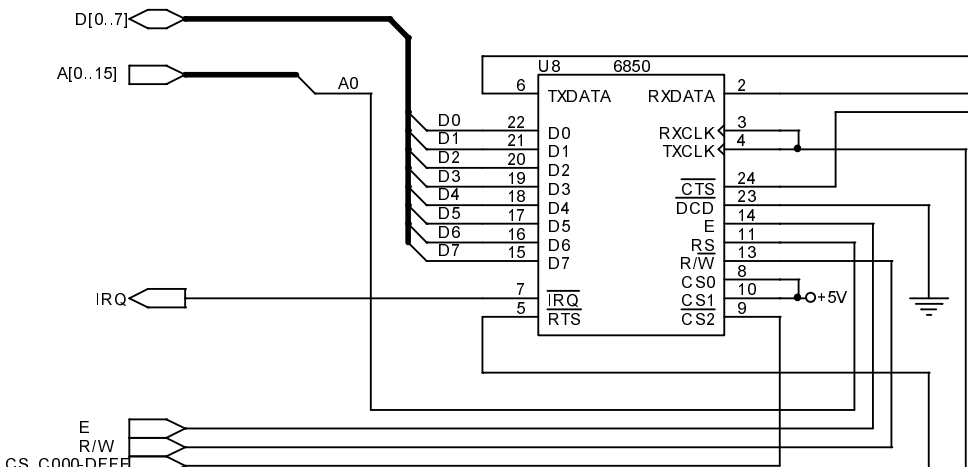


Şekil 18-24 Genel Amaçlı Sistemin Bellek Haritası



Şekil 18-25 Paralel G/Ç ve Gösterge Birimi





**Şekil 18-26 Seri G/Ç Birimi, Baud Oranı Üretici ve RS-232 Sürücü Devreleri**