



Teknoloji Fakültesi Elektrik Elektronik Mühendisliği Bölümü

EE-302

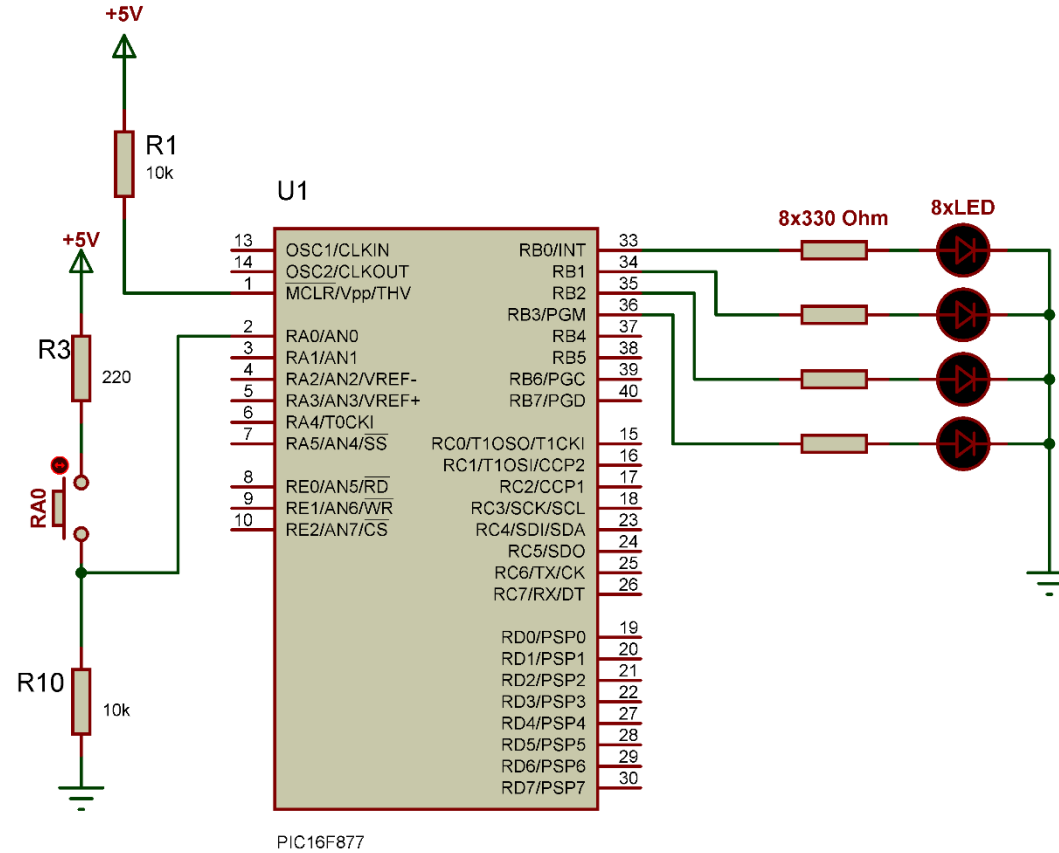
Mikroişlemciler

Karar Verme Döngüleri ve Display Kullanımı

4. Hafta

Prof. Dr. Mehmet DEMİRTAŞ

Karar verme döngüleri

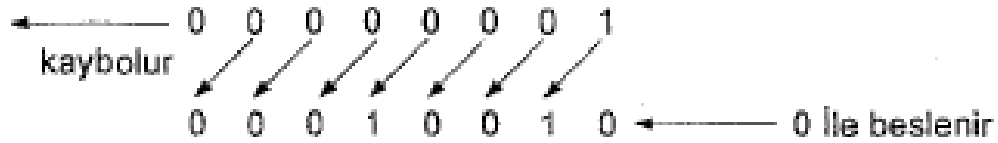


Port Giriş/Çıkış Komutları_örn

```
1  #include <16f877.h>
2  #fuses XT,NOWDT,NOPROTECT,
3  #use delay(clock=4000000)
4  #define buton pin_A0
5  int i=1;
6  void main ()
7  {
8      set_tris_a(0x01);    //
9      set_tris_b(0x00);    //
10     output_b(0x00);      //
11
12     while(1)              // Sonsuz döngü oluşturuluyor
13     {
14         if (input(buton))
15         {
16             delay_ms(15); // Butona basılınca meydana gelen arkın etkisini önlemek için verilen gecikme
17             output_b(i);  // b portuna i değeri yükleniyor
18             i=i+1;        // i değeri 1 arttırılıyor
19             while(input(buton)); // Buton bırakılana kadar bekle
20             if (i==10)    // i değeri test ediliyor.i değeri 10 olunca sıfırlanıyor.
21             {
22                 i=0;      // i değeri sıfırlanıyor
23             }
24         }
25     }
26 }
```

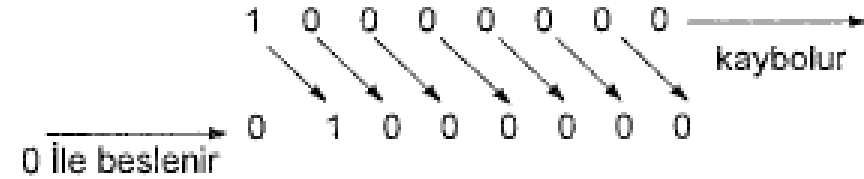
Port giriş/çıkış ve veri kaydırma komutları

led=led<<1;



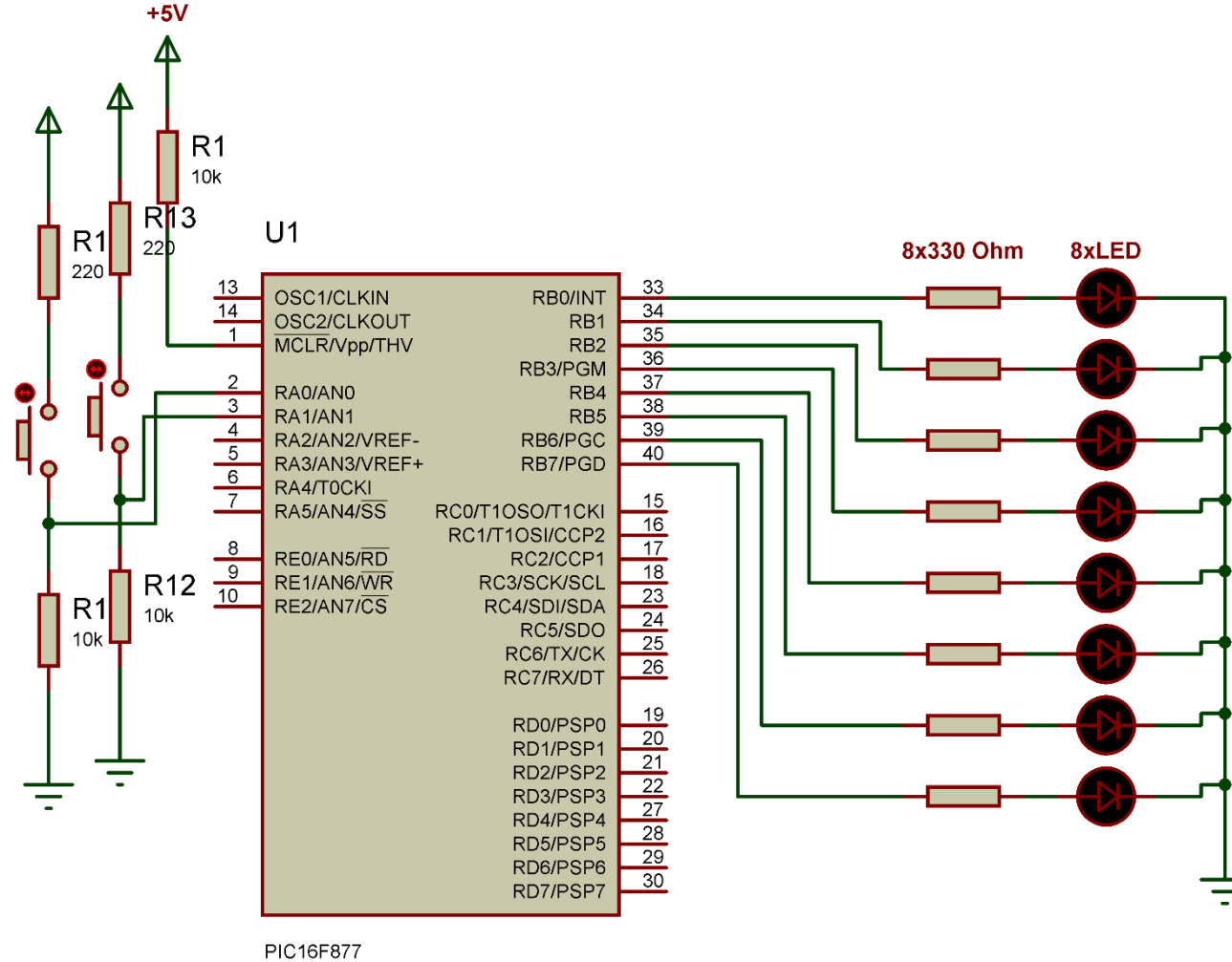
Şekil-7.3. Sola kaydırma işlemi.

led=led>>1;



Şekil-7.4. Sağa kaydırma işlemi.

Yürüyen Işık Uygulaması



Yürüyen Işık Uyg.

```

1  #include <16f877.h>
2  #fuses XT,NOVDT,NOPROTECT
3  #use delay(clock=4000000)
4  #define start pin_A0
5  #define stop pin_A1
6  int i, led;
7
8  void main ()
9  {
10     set_tris_a(0xff);
11     set_tris_b(0x00);
12
13     output_b(0x00);
14

```

```

15     while(1)
16     {
17         if(input(start))
18         {
19             led=1;
20
21             for(i=0;i<=7;i++)
22             {
23                 output_b(led);
24                 led=led<<1;
25                 delay_ms(100);
26             }
27
28             led=0x80;
29             output_b(led);
30             delay_ms(100);
31

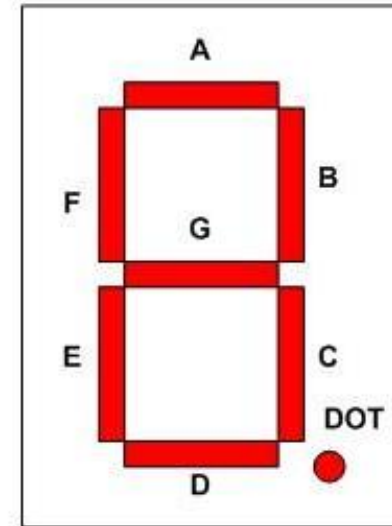
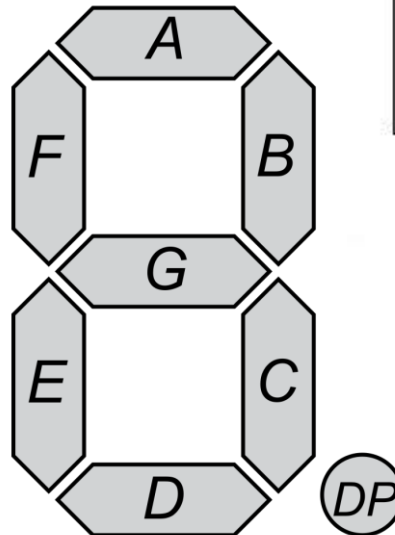
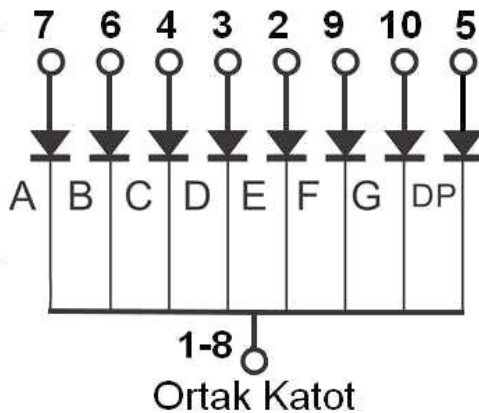
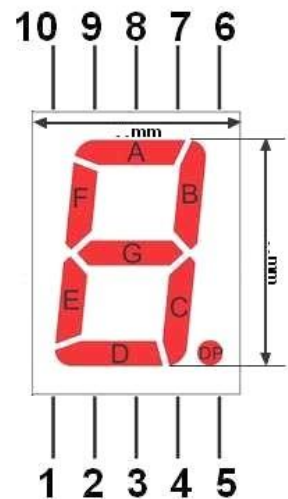
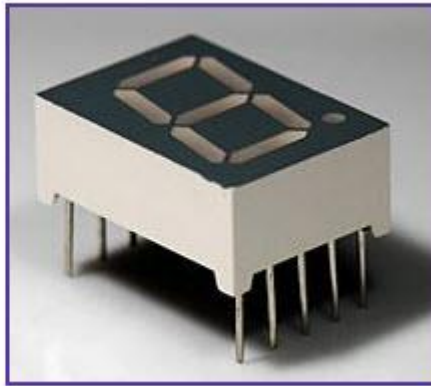
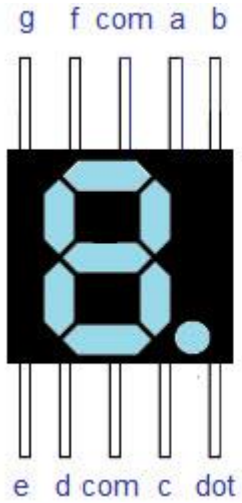
```

```

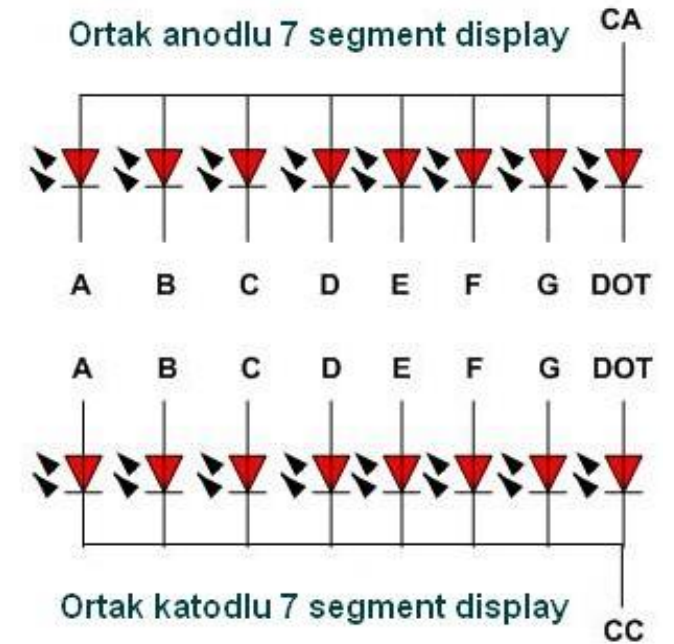
22         {
23             output_b(led);
24             led=led<<1;
25             delay_ms(100);
26         }
27
28         led=0x80;
29         output_b(led);
30         delay_ms(100);
31
32         for(i=0;i<=6;i++)
33         {
34             led=led>>1;
35             output_b(led);
36             delay_ms(100);
37         }
38     }
39 }
40

```

7 segment display

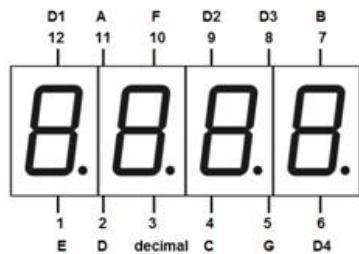
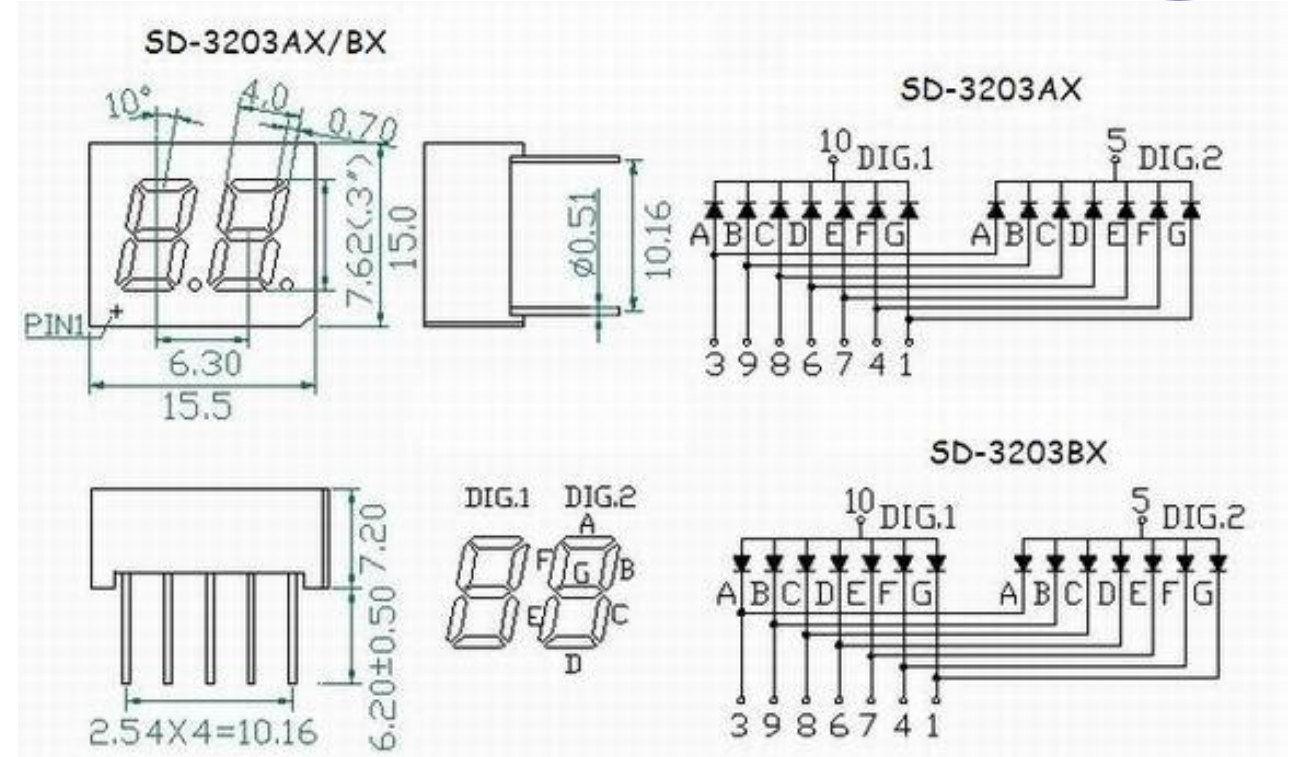
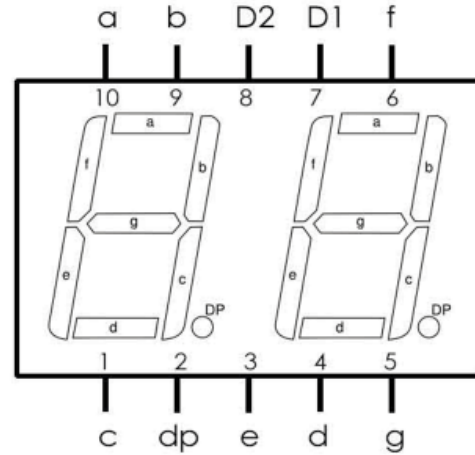
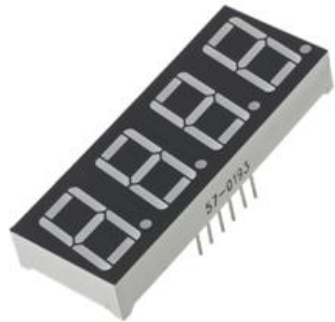
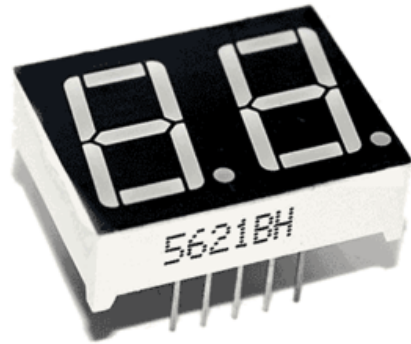


Segment isimleri



Ortak katodlu 7 segment display

7 segment display

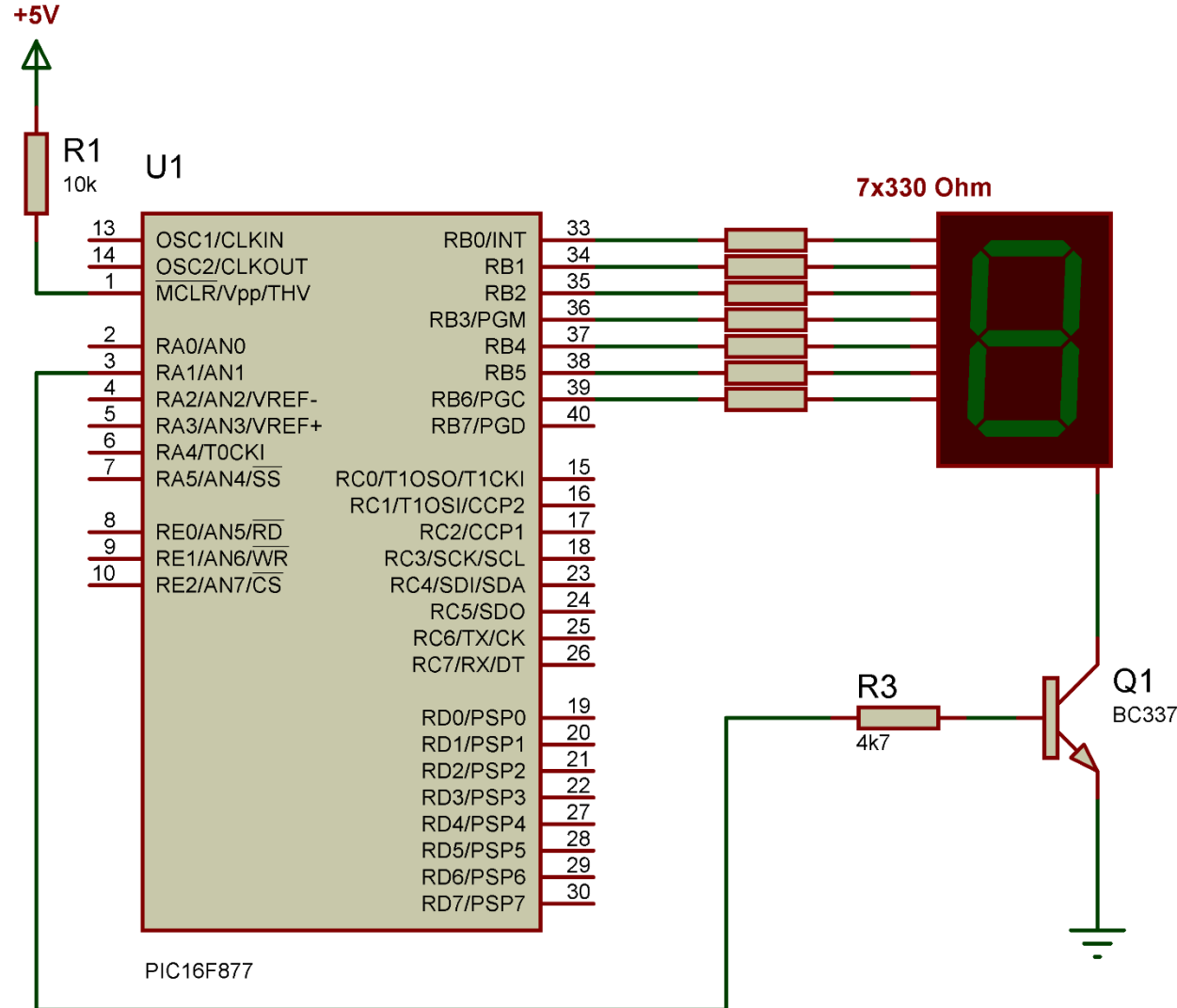


7 segment display

Sayı	Ortak Katot Display									Ortak Anot Display								
	h	g	f	e	d	c	b	a	HEX	h	g	f	e	d	c	b	a	HEX
0	x	0	1	1	1	1	1	1	0x3F	x	1	0	0	0	0	0	0	0x40
1	x	0	0	0	0	1	1	0	0x06	x	1	1	1	1	0	0	1	0x79
2	x	1	0	1	1	0	1	1	0x5B	x	0	1	0	0	1	0	0	0x24
3	x	1	0	0	1	1	1	1	0x4F	x	0	1	1	0	0	0	0	0x30
4	x	1	1	0	0	1	1	0	0x66	x	0	0	1	1	0	0	1	0x19
5	x	1	1	0	1	1	0	1	0x6D	x	0	0	1	0	0	1	0	0x12
6	x	1	1	1	1	1	0	0	0x7C	x	0	0	0	0	0	1	1	0x02
7	x	0	0	0	0	1	1	1	0x07	x	1	1	1	1	0	0	0	0x78
8	x	1	1	1	1	1	1	1	0x7F	x	0	0	0	0	0	0	0	0x00
9	x	1	1	0	1	1	1	1	0x6F	x	0	0	1	0	0	0	0	0x10
A	x	1	1	1	0	1	1	1	0x77	x	0	0	0	1	0	0	0	0x08
B	x	1	1	1	1	1	0	0	0x7C	x	0	0	0	0	0	1	1	0x03
C	x	0	1	1	1	0	0	1	0x39	x	1	0	0	0	1	1	0	0x46
D	x	1	0	1	1	1	1	0	0x5E	x	0	1	0	0	0	0	1	0x21
E	x	1	1	1	1	0	0	1	0x79	x	0	0	0	0	1	1	0	0x06
F	x	1	1	1	0	0	0	1	0x71	x	0	0	0	1	1	1	0	0x0E

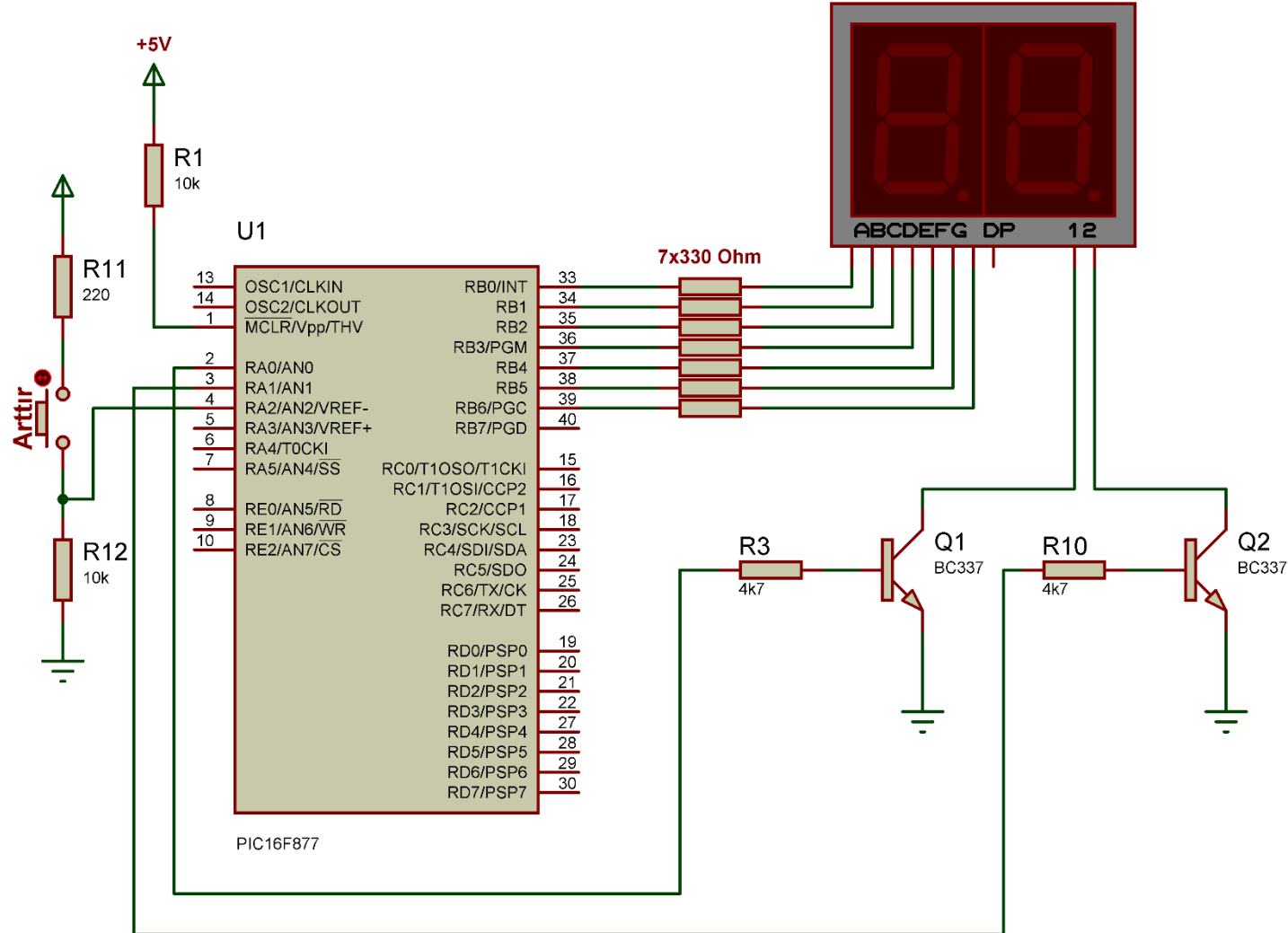
Tablo-7.1. Ortak katot display için değerler.

7 segment display örnek_1 (0-9 sayıcı)



```
1  #include <16f877.h>
2  #fuses XT,NOWDT,NOPROTECT
3  #use delay(clock=4000000)
4  int i; // i isminde 8 bitlik bir değişken tanımlanıyor.
5  // Ortak katot display için veri değerleri
6  const int digit[16]={0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7C, 0x07,
7                        0x7F, 0x6F, 0x77, 0x7C, 0x39, 0x5E, 0x79, 0x71};
8  void main ()
9  {
10     set_tris_a(0x05); //
11     set_tris_b(0x00); // B portu tümüyle çıkış olarak yönlendiriliyor
12     output_b(0x00); // B portu çıkışları sıfırlanıyor
13     output_a(0x02); // RA1 pini lojik-1 yapılıyor, diğer pinler lojik-0
14
15     while(1) // Sonsuz döngü oluşturuluyor
16     {
17         for(i=0;i<=15;i++)
18         {
19             output_b(digit[i]); // digit[i] değerini B portuna gönder
20             delay_ms(500); // 500 msn bekle
21         }
22     }
23 }
```

7 segment display örnek_2 (0-99 sayıcı)



```

1  #include <16f877.h>
2  #fuses XT,NOWDT,NOPROTECT
3  #use delay(clock=4000000)
4  #use fast_io(a)
5  #use fast_io(b)
6  #define display_1 pin_a0
7  #define display_2 pin_a1
8  #define buton pin_a2
9  char birler=0, onlar=0, sayi=0;
10 // Ortak katot display için veri değerleri
11 const int digit[10]={0x3F, 0x06, 0x5B, 0x4F, 0x66,
12                      0x6D, 0x7D, 0x07, 0x7F, 0x6F};
13 void main ()
14 {
15     set_tris_a(0xFC);
16     set_tris_b(0x00);
17     output_low(display_1); // 1. display pasif durumda
18     output_low(display_2); // 2. display pasif durumda
19
20     while(1)
21     {
22         birler=sayi%10; // birler hanesi hesaplanıyor
23         onlar=sayi/10; // onlar hanesi hesaplanıyor
24
25         output_high(display_2); // 2. display aktif durumda
26         output_b(digit[birler]); // displaye bilgi gönderiliyor
27         delay_ms(5); // gecikme veriliyor
28         output_low(display_2); // 2. display pasif durumda
29         output_high(display_1); // 1. display aktif durumda
30         output_b(digit[onlar]); // displaye bilgi gönderiliyor
31         delay_ms(5); // gecikme veriliyor
32         output_low(display_1); // 1. display pasif durumda
33

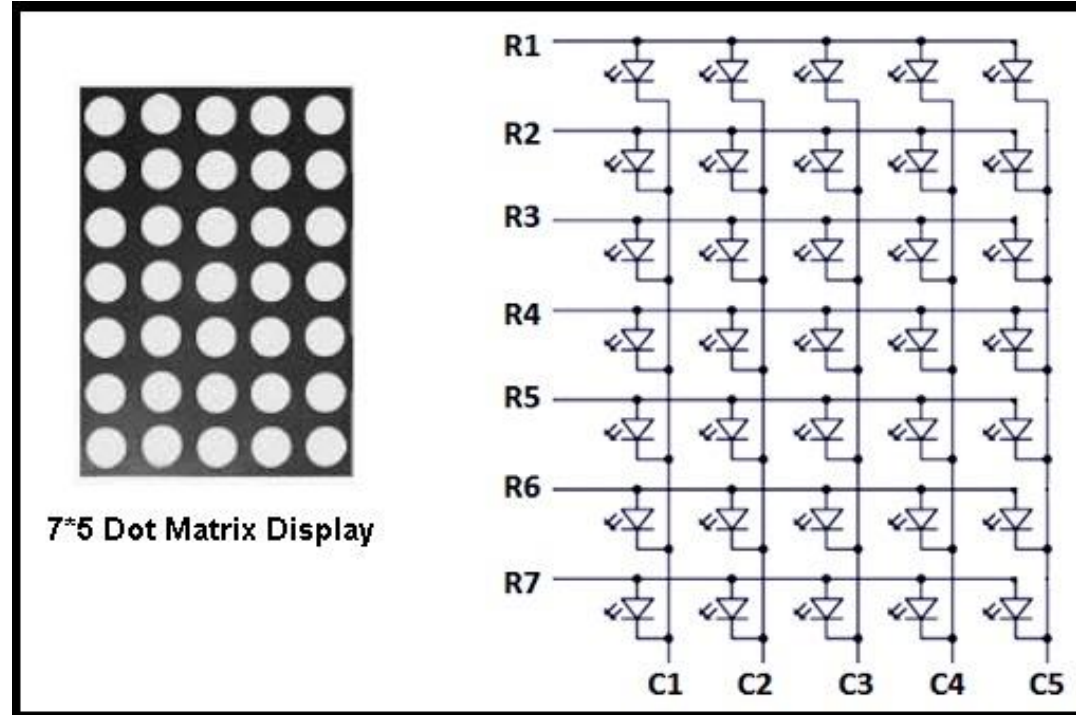
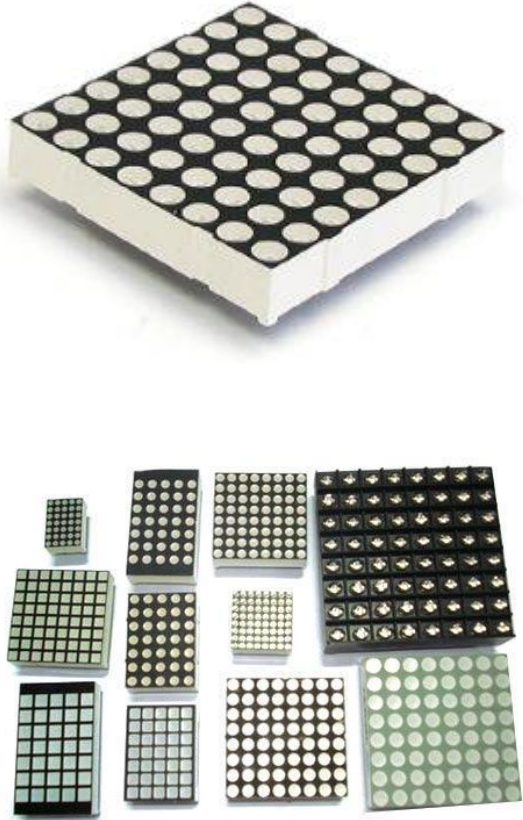
```

```

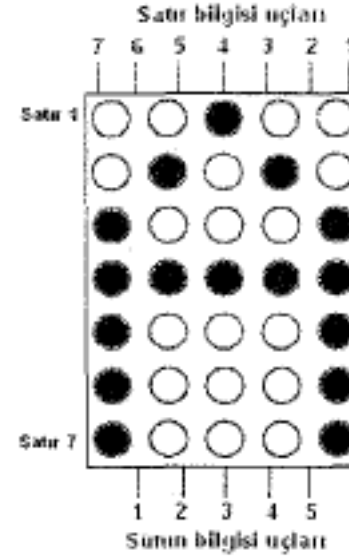
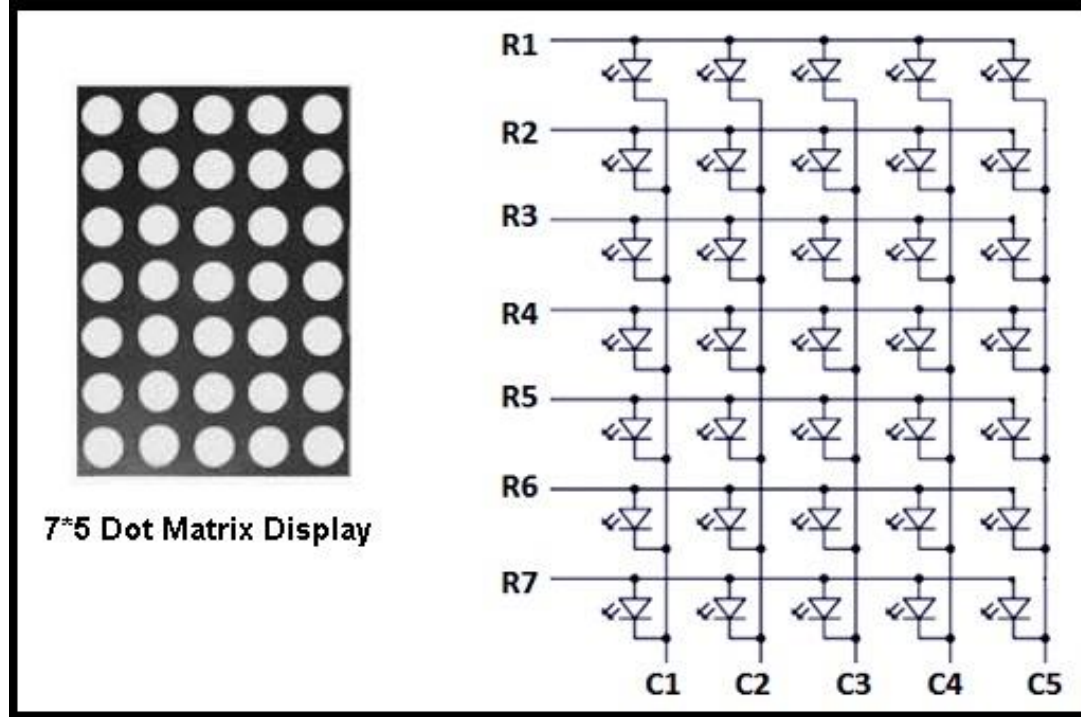
34     if (input(buton)) // Arttır buton
35     {
36         delay_ms(10); // Buton aralığı
37
38         sayi++; // sayi değeri
39
40         if (sayi==100) // Eğer sayı 100'a ulaştıysa
41             sayi=0;
42
43         birler=sayi%10; // Birler hanesi
44         onlar=sayi/10; // Onlar hanesi
45
46         while(input(buton)) // Buton basılıyken
47         {
48             output_high(display_2);
49             output_b(digit[birler]);
50             delay_ms(5);
51             output_low(display_2);
52             output_high(display_1);
53             output_b(digit[onlar]);
54             delay_ms(5);
55             output_low(display_1);
56         }

```

Matris Karakter Display

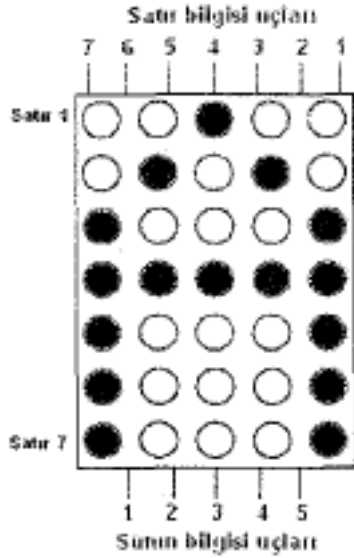


Matris Karakter Display



Matris LED display'leri sürmek için çeşitli entegrelerde mevcuttur. Örneğin MAX6952 ve MAX6953, maxim-dallas firmasının matris LED display'ler için sürücü entegre ürünleridir. **Katot sütun ve Anot sütun olmak üzere iki çeşit matris LED display vardır.** Katot sütun matris LED display'lerde sütunları aktif etmek için ilgili sütuna lojik-0, satırları aktif etmek için ilgili satıra lojik- 1 uygulanmalıdır. Anot sütun matris LED display'lerde ise sütunları aktif etmek için ilgili sütuna lojik- 1, satırları aktif etmek için ilgili satıra lojik-0 uygulanmalıdır.

Matris Karakter Display



1.sütun aktif (sütun1=1) diğer sütunlar pasif (sütun2=0, sütun3=0, sütun4=0, sütun5=0),
satir bilgisi olarak da 0b00000011(0x03)

Gecikme

2.sütun aktif (sütun2=1) diğer sütunlar pasif (sütun1=0, sütun3=0, sütun4=0, sütun5=0),
satir bilgisi olarak da 0b01110101(0x75)

Gecikme

3.sütun aktif (sütun3=1) diğer sütunlar pasif (sütun1=0, sütun2=0, sütun4=0, sütun5=0),
satir bilgisi olarak da 0b01110110 (0x76)

Gecikme

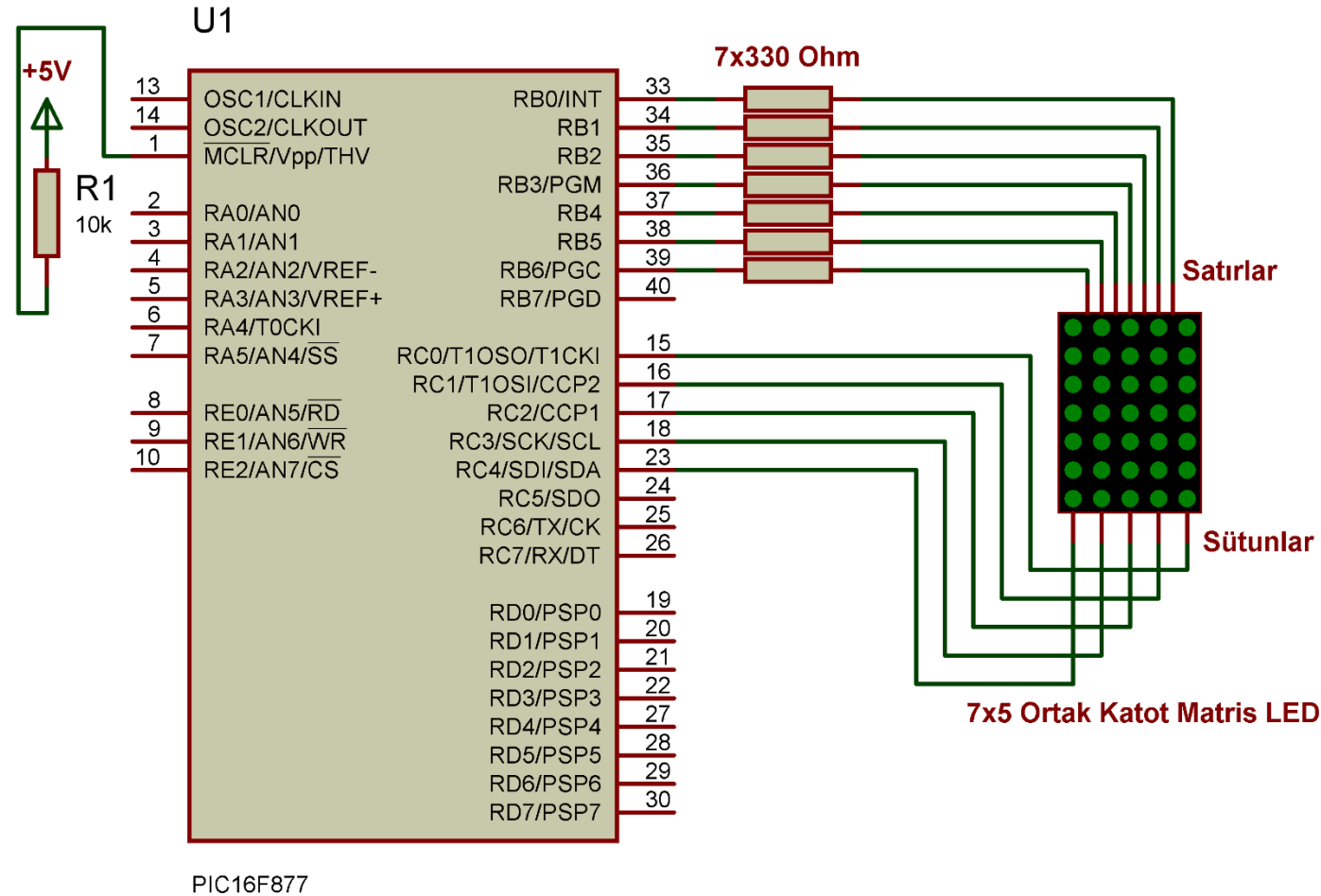
4.sütun aktif (sütun4=1) diğer sütunlar pasif (sütun1=0, sütun2=0, sütun3=0, sütun5=0),
satir bilgisi olarak da 0b01110101(0x75)

Gecikme

5.sütun aktif (sütun5=1) diğer sütunlar pasif (sütun1=0, sütun2=0, sütun3=0, sütun4=0),
satir bilgisi olarak da 0b00000011(0x03)

Gecikme

Matris Karakter Display



```
1 #include <16f877.h>
2 #fuses XT,NOWDT,NOPROTECT
3 #use delay(clock=4000000)
4 #use fast_io(b)
5 #use fast_io(c)
6 //***** Matrise Gönderilecek Karakter Bilgileri ****
7 int A[]={0x03,0x75,0x76,0x75,0x03};
8 int B[]={0x00,0x36,0x36,0x36,0x49};
9 int C[]={0x41,0x3E,0x3E,0x3E,0x5D};
10 int D[]={0x00,0x3E,0x3E,0x3E,0x41};
11 int E[]={0x00,0xB6,0xB6,0xB6,0xBE};
12 int arti[]={0x77,0x77,0x41,0x77,0x77};
13 int cift_ok[]={0x6B,0x5D,0x00,0x5D,0x6B};
14 //***** Sütun Seçme Fonksiyonu ****
15 int sutunsec (int s)
16 { switch (s)
17 { case 0: return(0x10);break;//1.Sütun aktif,diğer sütunlar pasif
18   case 1: return(0x08);break;//2.Sütun aktif,diğer sütunlar pasif
19   case 2: return(0x04);break;//3.Sütun aktif,diğer sütunlar pasif
20   case 3: return(0x02);break;//4.Sütun aktif,diğer sütunlar pasif
21   case 4: return(0x01);break;//5.Sütun aktif,diğer sütunlar pasif
22 }
23 }
```

```
25 //***** Matrise Veri Gönderme Fonksiyonu *****
26 void matris_karakter_gonder(char harf[], int tekrar)
27 {
28     int i,j; // Değişkenler tanımlanıyor
29
30     for (i=1;i<tekrar;i++) // Karakterin Matris LED'de görünmesi için tekrar döngüsü
31     {
32         for(j=0;j<5;j++) // // Bu döngüde tarama işlemi yapılıyor
33         {
34             output_c(sutunsec(j)); // Sütun seç ve C portuna sütun değerini gönder
35             output_b(harf[j]);     // Gösterilecek harfin satır bilgisini B portuna gönder
36             delay_ms(10);          // Gecikme veriliyor
37         }
38     }
39 }
40
```

```
41  /***** ANA PROGRAM FONKSİYONU*****/
42
43  void main ()
44  {
45      set_tris_b(0x00); // B portu tümüyle çıkış olarak yönlendiriliyor
46      set_tris_c(0x00); // C portu tümüyle çıkış olarak yönlendiriliyor
47
48      while(1)
49      {
50          matris_karakter_gonder(A,25);// A karakterini matriste göster
51          matris_karakter_gonder(B,25);// B karakterini matriste göster
52          matris_karakter_gonder(C,25);// C karakterini matriste göster
53          matris_karakter_gonder(D,25);// D karakterini matriste göster
54          matris_karakter_gonder(E,25);// E karakterini matriste göster
55          matris_karakter_gonder(arti,25);//+ karakterini matriste göster
56          matris_karakter_gonder(cift_ok,25);// Çift yönlü ok karakterini matriste göster
57
58      }
59  }
60
```

Kaynaklar

- CCS C Programlama Kitabı, Serdar Çiçek, Altaş Yayıncılık
- Mikroelektronika C programlama e-kitabı «<https://www.mikroe.com/ebooks/pic-microcontrollers-programming-in-c>»
- Mikroelektronika C programlama e-kitabı «<https://www.mikroe.com/ebooks/pic-microcontrollers-programming-in-c/inputoutput-ports>»