

EK-A

MİKRO C'DE KOD YAZIMI

A-1 Küçük-Büyük Harf Duyarlılığı İlkesi

mikroC program editöründe programlama dili olarak C dilinin özellikleri kullanılır. Değişken türleri, diziler, döngüler, matematiksel operandlar, açıklama satırları, fonksiyonların yazımı, vb. yazılım sentaksıyla ilgili kurallar C dilinde olduğu gibidir.

C programlama dili bazı programlama dillerinden farklı olarak küçük büyük harf duyarlılığına sahiptir (case sensitive – harf duyarlı). Bu özellik mikroC'de kod yazımında da dikkat edilmesi gereken kurallardandır. Ancak mikrodenetleyiciye özgü kodların ya da kullanıcı tanımlı değişkenlerin yazımında küçük büyük harf duyarlılığı yoktur.

Örnekler:

```
int bekle=50;
```

Bu örnekte, 'int' kelimesi tamsayı (integer) sayı değişkeni tanımlamak için kullanılır ve tüm harfleri küçük harfle yazılmalıdır.

```
int bekle=50;
```

Bu örnekte, "bekle" kelimesi kullanıcı tanımlı bir değişkendir ve "BEKLE", "Bekle", "BeKle" gibi tüm farklı yazımları aynı anlama gelmektedir.

```
PORTB=0B00000000;
```

Bu örnekte, "PORTB" program tanımlı bir değişkendir ve mikrodenetleyicinin B portunu ifade eder. Küçük-büyük harf duyarlılığı yoktur.

```
do{  
PORTB = 0x00;  
Delay_ms(500);  
PORTB = 0xFF;  
Delay_ms(500);  
}while(1);
```

Bu örnekte, "do{ . . . }while(1)" mikroC'de kullanılan üç döngü yapısından biridir ve küçük-büyük harf duyarlıdır. Ancak mili saniye mertebesinde gecikme oluşturmak için kullanılan "delay_ms(500)" gecikme fonksiyonu mikroC editörünün mikrodenetleyici programlamaya özgü geliştirilmiş bir fonksiyonudur ve küçük-büyük harf duyarlı değildir.

```
void main(){
```

```
...
```

```
}
```

Bu örnekte, fonksiyonlarda kullanılan ve hiçbir değer döndürmeyen anlamına gelen 'void' argümanı harf duyarlıyken 'main()' gibi fonksiyon adları harf duyarlı değildir.

Görüldüğü üzere mikroC editöründe bazı kodlar harf duyarlıyken, bazıları değildir. Bu konuya alışkanlık kazanmanız çok sayıda kod yazarak gerçekleşecektir. Genel olarak, değişken ve fonksiyon tanımlamalarında küçük harf, program tanımlı “PORTB, TRISB, ANSEL, CMCON, vb.” değişkenlerde ve kaydedicilerin yazımında BÜYÜK harf kullanmanız ayırt edici ve anlaşılır bir kod yazımı açısından faydalı olacaktır.

A-2 Açıklama Satırı Ekleme

Daha önceden belirtildiği gibi mikroC programı kod yazım ilkeleri açısından pek çok özelliği C dilinden almıştır. Bunlardan biride program kodlarına etkisi olmayan açıklama satırlarıdır. Açıklama satırları uzun kodlamada son derece faydalı bir özelliktir. Açıklama satırı eklemenin iki yolu vardır.

“//” çift eğik çizgi (çift slaş) ile tek satırlı açıklama bilgisi yazılır.

Örnek:

```
int bekle=50; // Bekleme süresini ayarlayacak değişken
```

“/* ... */” ile birden fazla açıklama satırı eklenebilir. /* işareti konduktan sonra yazılan ve */ işaretinin başladığı yere kadar olan bütün satırlar açıklama satırı olarak yorumlanır. Bu nedenle açıldıktan sonra kapatılması gerekir.

Örnek:

```
TRISA = 0x00;
TRISB = 0x00;
PORTA = 0;
PORTB = 0;
/* Tüm portlar çıkış olarak ayarlanıyor.
Portlar başlangıç olarak sıfırlanıyor */
ANSEL = 0B00000000; /* Analog girişler kapatılıyor */
CMCON = 0x07; //Karşılaştırmacı devre dışı bırakılıyor
```

Profesyonel programcılar da programlarının kritik yerlerine açıklama satırları ekler. Bunu alışkanlık haline getirmeniz, çalışmaya ara verdiğiniz projelere döndüğünüzde ya da ihtiyaç duyduğunuzda size büyük fayda sağlayacaktır.

A-3 Programlama Sentaksı

mikroC program satırları “;” ile sonlandırılır. Fonksiyonlar ve döngüler gibi “{” küme paranteziyle başlanan satırlarda “;” kullanılmaz. Ayrıca ‘goto’ komutu kullanılarak yapılan dallanmalarda, dallanmanın yapıldığı etiketin kullanıldığı satırda “:” kullanılır. Aynı tür değişkenler aralarına virgül konularak tanımlanabilir ve tümüne ya da bir kısmına ilk değer atanabilir.

Örnek:

```
int a, i, bekle=50; //Aynı satırda birden fazla değişken tanımlanmıştır
void bekleme() { //bu değişkenlerden bazılarında değer atanmıştır
Vdelay_ms(bekle);
}
test: //goto komutuyla dallanma yapılan etiket
for(i=0;i<50;i++){
```

```

PORTB=0X02;
  bekleme();
  PORTB=0X00;
  bekleme();
}
a++;
if (a<100) goto test:

```

Örnek kod parçasında a değişkeni 1 artırılır ve 100'den küçük olduğu sürece 'for' döngüsünün üstünde yer alan "test:" etiketine dallanılır. Böylece 'for' döngüsü 100 defa işletilir. Program kodunda görüldüğü üzere her açılan küme parantezi için kapatma parantezi kullanılır. **Kural olarak açılan parantez kadar kapatma parantezi kullanılmalıdır.**

mikroC editöründe açılan küme parantezi için otomatik olarak kapatma parantezi açılmaz. Ayrıca otomatik olarak iç içe (nested) bir yapı da oluşturulmaz. Özellikle uzun kod yazımında iç içe geçmiş kodları seviyelerine göre girintili olarak yazmak program yazımının okunurluğunu artıracaktır.

Örnek:

```

void main() {
  ayarlar();
  uzunluk_msj01 = strlen(msj01);
  uzunluk_msj02 = strlen(msj02);
  konum = (16-uzunluk_msj02)/2;
  for(;;){ //sonsuz döngü
    pointer_msj01 = &msj01;
    pointer_msj02 = &msj02;
    for(i=0;i<16;i++){
      Lcd_Out(1,16-i,msj01);
      Lcd_Out(2,konum,pointer_msj02);
      delay_ms(100);
    }
    for(i=0;i<uzunluk_msj01;i++){
      Lcd_Out(1,1,pointer_msj01);
      Lcd_Out(2,konum,pointer_msj02);
      delay_ms(100);
      Lcd_Cmd(_LCD_CLEAR);
      pointer_msj01++;
    }
  }
}
//Görüldüğü üzere açılan her küme parantezi kadar
//kapatma parantezi kullanılmıştır.

```

Yapısal programcılık ilkeleri açısından 'goto' deyimiyle dallanma yapmamaya çalışılmalıdır. Özellikle döngülerin içinde 'goto' deyiminin kullanımı tavsiye edilmeyen bir uygulamadır. Döngünün kırılarak terk edilmesi iyi bir programlama uygulaması değildir. Dallanmalar olabildiğince "do...while" ya da "while" döngü yapıları kullanılarak yapılmalıdır. Ayrıca başlangıç koşulları tanımlanmış ve ona göre işleyen bir kod bloğunun içine 'goto' deyimiyle dallanma yapılamaz.

Örnek:

Bir fabrika otomasyon sisteminde girişlerden gelen bilgilerin belli bir döngü süresince taranarak mikrodenetleyicinin EEPROM veri hafızasına yazıldığını düşünelim. Bu verilerin zaman içinde EEPROM hafızasından silinip silinmediğini kontrol etmek istediğimizi varsayalım. Hafızadan okunan bilginin dizi değişikende tutulan bilgiyle aynı olmaması durumunda çıkışları hızlıca çakarlı şekilde yakarak uyarı verdikten sonra son okunan port bilgisini çıkışa aktaran bir programımız olsun ve bu durumda sistem tekrar başa dönsün. Programı ilk önce 'goto' deyimini kullanarak yapalım:

Tablo A.1 GOTO deyiminin örnek uygulaması

```
unsigned char i,k, oku;
unsigned char port[10];
void main() {
    TRISB = 0xFF;           // B portu tüm pinleriyle giriş yapılıyor
    TRISA = 0x00;           // A portu tüm pinleriyle çıkış yapılıyor
    PORTB = 0x00;
    PORTA = 0x00;
    CMCON = 0x07;
    INTCON.GIE=0;           //Evrensel kesme devre dışı bırakılıyor
    port_cikis:             //GOTO komutuyla dallanılacak etiket
    for(i=0;i<10;i++){
        port[i] = PORTB;     //EEPROM hafızasına 10 turda veri yazılıyor
        EEPROM_Write(0x00+i, port[i]);
        delay_ms(500);
    }
    for(i=0;i<10;i++){
        oku = EEPROM_Read(0x00+i); //EEPROM hafızasından okunan veri
        if(oku == port[i]){ //dizi değişkenin içeriğiyle karşılaştırılıyor
            delay_ms(250);
            PORTA = port[i];
        }
        else{
            for(k=0;k<5;k++){
                PORTA=0xFF; //Hafızadaki veri eşleşmiyorsa A portu çıkışlarına
                delay_ms(100); //çakarlı şekilde uyarı verilir.
                PORTA=0x00;
                delay_ms(100);
            }
            PORTA=oku; //Ardından eşleşmeyen veri A portuna gönderilir.
            goto port_cikis;
        }
    }
}
```

Bu uygulamayı 'goto' deyimi kullanmadan 'do...while' döngüsü kullanarak yapmak istersek takip eden düzenlemeyi yapabiliriz:

Tablo A.2 GOTO deyiminin yerine "do...while" döngüsünün kullanılması

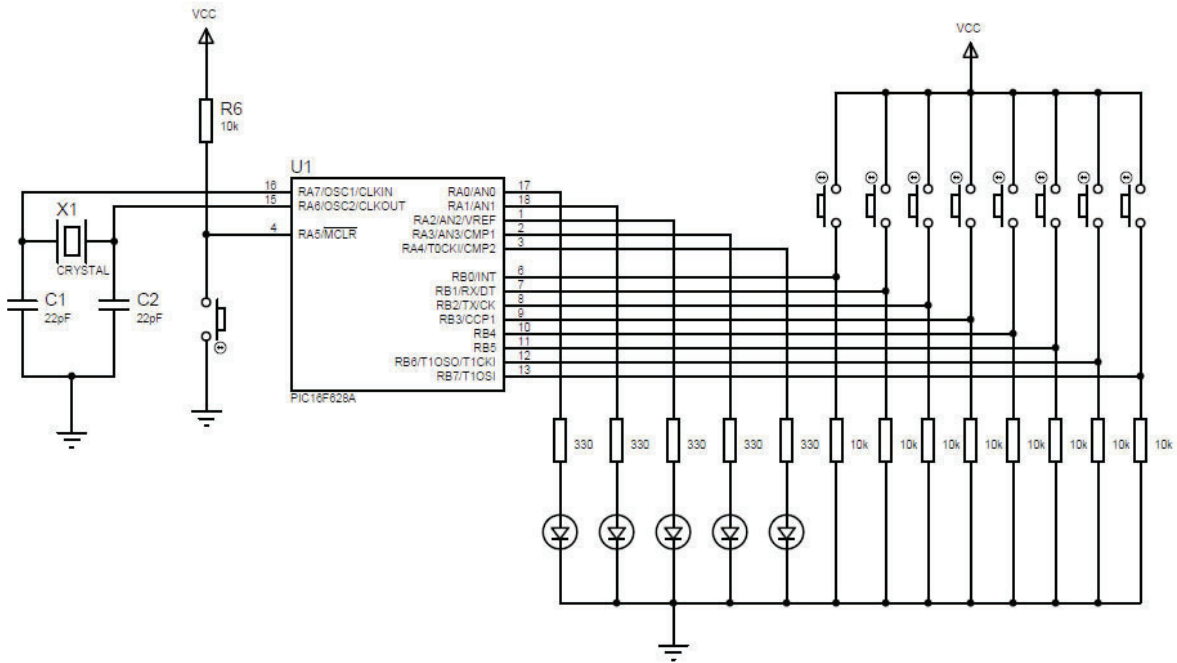
```
unsigned char i,k, oku, hata=0; //yeni bir değişken eklendi
unsigned char port[10];
void main() {
    TRISB = 0xFF;           // B portu tüm pinleriyle giriş yapılıyor
    TRISA = 0x00;           // A portu tüm pinleriyle çıkış yapılıyor
    PORTB = 0x00;
    PORTA = 0x00;
    CMCON = 0x07;
    INTCON.GIE=0;           //Evrensel kesme devre dışı bırakılıyor
    do{                       //GOTO etiketinin yerine konan do...while döngüsü
        for(i=0;i<10;i++){
            port[i] = PORTB; //EEPROM hafızasına 10 turda veri yazılıyor
            EEPROM_Write(0x00+i, port[i]);
            delay_ms(500);
        }
        i=0;
        hata=0;
        do{
            oku = EEPROM_Read(0x00+i+1); //EEPROM hafızasından okunan veri
            if(oku == port[i]){ //dizi değişkenin içeriğiyle karşılaştırılıyor
                delay_ms(250);
                PORTA = port[i];
            }
        }
    }
}
```

```

}
else{
for (k=0;k<5;k++){
PORTA=0xFF; //Hafızadaki veri eşleşmiyorsa A portu çıkışlarına
delay_ms(100); //çakarlı şekilde uyarı verdirilir.
PORTA=0x00;
delay_ms(100);
}
PORTA=oku;
hata=1; //hata değişkeni 1 olur
}
i++;
}while(!hata && i<10); //hata yoksa (0 ise) ve 10 olmamışsa
}while(hata); //hata varsa (1 ise) başa döner
}

```

Şekil A.1'de Tablo A.1 ve A.2'deki kodların örnek uygulama devresi gösterilmiştir. Devre canlandırmasında B portu girişlerine bağlı olduğu varsayılan sensörler butonlarla temsil edilmiştir.



Şekil A.1 GOTO deyimi uygulama örneği

A-4 Matematiksel İşlemler

mikroC dili matematiksel işlemlerin yapılabilmesi için bir dizi aritmetik operatör sunar. Bunlar, nümerik olarak işlem gören değişkenlerden ve işlem sonucunda işleme ve değişken türüne uygun sayı döndüren operatörlerdir. Tüm operatörler soldan sağa doğru işlem görür ve her birinin işlem önceliği vardır. İşlem önceliği yüksek değere sahip olan operatörler önce işlem görür. Uzun hesaplamalarda ilgili operatörün işlem gördüğü sayıları diğer işlemlerden ayırmak için parantez kullanılması tavsiye olunur.

Tablo A.3 Aritmetik operatörler

Operatör	İşlem	Öncelik
İki Sayılı İşlemler		
+	Toplama	12
-	Çıkartma	12
*	Çarpma	13
/	Bölme	13
%	Mod alma operatörüdür. Tam sayılarla kullanılır. Bölme işlemi sonucunda kalan kısmı döndürür. Kayan nokta sayılarda kullanılmaz (double ve float).	13
Çift Sayılı İşlemler		
+	Önüne geldiği sayının işaretini etkilemez	14
-	Önüne geldiği sayının işaretini değiştirir.	14
++	İşlem gördüğü sayının değerini bir artırır. Sayının önünde ya da sayıdan sonra kullanılabilir. Sayının önünde kullanıldığında hesaplamadan önce sayının değeri artırılır. Sonra kullanıldığında ise hesaplama işlemi yapıldıktan sonra sayının değeri artırılır.	14
--	İşlem gördüğü sayının değerini bir azaltır. Sayının önünde ya da sayıdan sonra kullanılabilir. Sayının önünde kullanıldığında hesaplamadan önce sayının değeri azaltılır. Sonra kullanıldığında ise hesaplama işlemi yapıldıktan sonra sayının değeri azaltılır.	14

NOT: '*' operatörü 'pointer' tipinde değişken tanımlamasında da kullanılır.

A-4.1 İki Sayılı Aritmetik İşlemler

İki tamsayının bölümünden bir tamsayı döndürülür. Eğer işlem sonucunda kalan varsa kalanlı kısım atılır:

```
/* örnek */
7/4; /* 1'e eşittir */
7*3/4 /* 5'e eşittir */
/* fakat */
7.*3./4.; /* 5.25'e eşittir. Çünkü float sayı işlemi yapılmıştır */ */
```

% Mod operatörü yalnızca tamsayı türünde sayı değişkenlerinde kullanılabilir:

```
/* örneğin: */
9 % 3; /* 0'a eşittir */
7 % 3; /* 1'e eşittir */
-7 % 3; /* -1'e eşittir */
```

Aritmetik operatörler karakter işlemlerinde de kullanılabilir:

```
'A' + 32; /* 'a' karakterine eşittir (yalnızca ASCII kodu için) */
'G' - 'A' + 'a'; /* 'g' karakterine eşittir (hem ASCII hem EBCDIC kodu için) */
```

Tablo A.4'te çift sayı operatörleriyle yapılan bazı aritmetik işlemlerin LCD ekranda nasıl gösterileceği üzerine uygulama kodu gösterilmiştir. Bu kodun PROTEUS-ISIS® ortamında hazırlayacağınız bir devrede izlenmesi için Şekil 3.7'deki devreyi kullanabilirsiniz. Tablo 3.11'deki programın LCD bağlantı ayarları ve mikrodenetleyici ayarlarıyla ilgili kodlarını bu koda eklemeyi unutmayınız.

Tablo A.4 Çift sayı operatörleriyle yapılan aritmetik işlemlerin LCD ekranda gösterilmesi

```
char i=0;
unsigned int j=0;
float k=0;
char str1[16], str2[16], str3[16], str4[16];
void main() {
    ayarlar(); /* Tablo3.11'deki LCD ve mikrodenetleyici ayarları kodunu
    programınıza ekleyin */
    i=9/7;
    BytetoStr(i, str1); //1 byte'lık (maks 255) tam sayıdan stringe dönüşüm
    Lcd_Out(1,1, strcat("1.ISLEM= ", str1));
    delay_ms(1000);
    Lcd_Cmd(_LCD_CLEAR);
    j=8*4/7;
    InttoStr(j, str2); //2 byte'lık (maks 65535) tam sayıdan stringe dönüşüm
    Lcd_Out(1,1, strcat("2.ISLEM= ", str2));
    delay_ms(1000);
    Lcd_Cmd(_LCD_CLEAR);
    k=9.5*4/0.7;
    FloattoStr(k, str3); //4 byte'lık kayan nokta sayıdan stringe dönüşüm
    Lcd_Out(1,1, strcat("3.ISLEM= ", str3));
    delay_ms(1000);
    Lcd_Cmd(_LCD_CLEAR);
    i=9%4; //9 sayısının 4'e göre modu alınıyor
    BytetoStr(i, str4); //1 byte'lık tam sayıdan stringe dönüşüm
    Lcd_Out(1,1, strcat("4.ISLEM= ", str4));
}
```

Tablo A.4'deki program kodunda çeşitli sayı değişkenlerini metin değişkenine (string) dönüşüm için mikroC'nin "Conversions - Dönüşümler" kütüphanesi kullanılmıştır. Ayrıca iki string ifadenin birbirine eklenmesini sağlayan "strcat" fonksiyonu kullanılmıştır. Bu işlemlerin yapılabilmesi için "View → Library Manager" penceresinden "Conversions" ve "C_String" kütüphanelerini seçiniz. Farklı aritmetik işlemleri deneyerek farklı sonuçlar elde etmeye çalışınız.

A-4.2 Tek Sayılı Aritmetik İşlemler

Tek sayı operatörleri ‘++’ ve ‘--’, C dilindeki hem önden hem sondan eklemeli tek operatör türüdür. Ön ek olarak (++k, --k) ya da son ek olarak (k++, k--) şeklinde kullanılır.

```
/* örneğin */
int j = 5;
k = ++j; /* k = k + 1, j = k, bize j = 6, k = 6 sonucunu verir */
/* fakat */
int j = 5;
k = j++; /* j = k, k = k + 1, bize j = 5, k = 6 sonucunu verir */
```

Bu operatörler daha çok yukarıdaki kullanımdan farklı olarak tek başına bir değişkenin değerini artırmada ya da eksiltmede kullanılır. Özellikle “for”, “while” ve “do..while” döngülerinde tam sayı değişkeni olarak döngü adedinin ayarlanmasında sayaç olarak kullanılırlar.

```
/* örneğin */
int i;
for(i=0;i<5;i++){
    PORTB = i; /* i değişkeni B portuna gönderilir */
    Delay_ms(100); /* ve 1 artırılır */
}

/* ya da */
int i;
for(i=4;i>=0;i--){
    PORTB = i; /* i değişkeni B portuna gönderilir */
    Delay_ms(100); // ve 1 azaltılır
}

/* örneğin */
int i = 0; /* i değişkeni için başlangıç koşulu */
while(i<5){
    PORTB = i; /* i değişkeni B portuna gönderilir */
    i++; // ve 1 artırılır
    Delay_ms(100);
}

/* ya da */
int i = 4; /* i değişkeni için başlangıç koşulu */
while(i>=0){
    PORTB = i; /* i değişkeni B portuna gönderilir */
    i--; // ve 1 azaltılır
    Delay_ms(100);
}
```

DİKKAT: Sayı eksiltme operatörünün (--) döngülerde kullanımına dikkat edilmelidir. Eğer değişken türünüz pozitif tanımlanmış bir tam sayı değişkeni ise (Ör: ‘char’ veya ‘unsigned int’) eksiltilen sayı 0’ın altına indiğinde ilgili değişkenin pozitif olarak en yüksek değerinden itibaren geriye gelmeye devam eder. Örneğin değişkeniniz ‘char’ türündeyseniz ve değeri 0’ın altına düştüyse değişkeninizin içeriği 255’e ayarlanır. Bu nedenle eksiltme işlemlerinde değişken türünüzün işaretli olarak tanımlanmış olması gerekir (Ör: ‘signed char’ veya ‘int’).

A-5 Bit Duyarlı (Bitwise) Operatörler

İşlem gören sayının bitlerinde bit ağırlıklı çalışma yapmayı sağlayan operatörlerdir. Özellikle mikrodenetleyicilerin portlarından okunan ya da portlara gönderilen bilgilerde bit ağırlıklı olarak işlem yapmada sıklıkla kullanılırlar.

Tablo A.5 Bit duyarlı operatörler

Operatör	İşlem	Öncelik
&	Bitler üzerinde Mantıksal VE (AND) işlemi uygular. Karşılaştırılan bitlerin hepsi '1' ise '1' döndürür. Aksi takdirde '0' döndürür.	8
	Bitler üzerinde Mantıksal VEYA (OR) işlemi uygular. Karşılaştırılan bitlerden en az birinin '1' olması durumunda '1' döndürür. Tüm bitler '0' ise '0' döndürür.	6
^	Bitler üzerinde Özel VEYA (XOR) işlemi uygular. Bit çiftlerini karşılaştırır ve eğer birbirlerinin tümleyeniyse '1' değilse '0' döndürür.	7
~	Bit duyarlı tümleç operatörüdür ve tek sayılı işlem görür. İşlem yaptığı sayının her bir bitinin tersini alır.	14
<<	Bit duyarlı sola kaydırma operatörüdür. İşlem gördüğü sayının bitlerini sola kaydırır. Sayının MSB biti atılır, LSB bitine '0' atanır.	11
>>	Bit duyarlı sağa kaydırma operatörüdür. İşlem gördüğü sayının bitlerini sağa kaydırır. Sayının LSB biti atılır, MSB bitine '0' atanır.	11

NOT: '&' operatörü bir 'pointer' değişkeninin adres referansı olarak da kullanılır.

Bit ağırlıklı operatörler soldan sağa doğru işlem görür. Yalnızca '~' operatörü sağdan sola işlem görür.

A-5.1 Bit Duyarlı Mantıksal İşlemler

Bit duyarlı operatörlerden '&', '|' ve '^' işlem gören sayıların karşılık gelen bit çiftleri arasında mantıksal işlemler gerçekleştirir. '~' operatörü işlem yaptığı sayının her bitinin tümleyenini alır.

Tablo A.6 Bit duyarlı mantıksal işlem operatörleri

&	0	1		0	1	^	0	1	~	0	1
0	0	0	0	0	1	0	0	1	1	1	0
1	0	1	1	1	1	1	1	0			

Örnekler

```
0x1234 & 0x5678 // 0x1230'a eşittir
```

```
/* çünkü ..
```

```
0x1234 : 0001 0010 0011 0100
```

```
0x5678 : 0101 0110 0111 1000
```

```
-----  
& : 0001 0010 0011 0000
```

```
.. buda 0x1230' denk gelir */
```

```
/* Benzer olarak: */
```

```
0x1234 | 0x5678; // 0x567C'a eşittir
```

```
0x1234 ^ 0x5678; // 0x444C'a eşittir
```

```
~ 0x1234; // 0xEDCB'a eşittir
```

A-5.2 Bit Duyarlı Kaydırma Operatörleri

'<<' ve '>>' ikili sayı sistemi (binary) operatörleri sol taraftaki sayının bitlerini sağ taraftaki sayının değeri kadar kaydırır. Sağ taraftaki sayının pozitif olması gerekir.

İşaretsiz bir sayının n değeri kadar sola kaydırılması, eğer atılan tüm bitler sifira eşitse o sayının 2^n ile çarpımına denktir. Bu durum atılan tüm bitlerin bir işaret bitine eşit olması durumunda işaretli sayılar için de geçerlidir.

Aşağıdaki örnekte onlu ve onaltılık iki sayının 4'er defa sola kaydırılması sonucu elde edilen değerler gösterilmiştir. 3 sayısı 4 defa sola kaydırıldığında sonuç $2^4 \times 3 = 48$ olmuştur.

```
3 << 4; /* 48'e eşittir */
0x3801 << 4; /* 0x8010'a eşittir ve taşma vardır! */
```

4 basamaklı heksadesimal sayının değişimi ise adım adım Tablo A.7'de gösterilmiştir.

Tablo A.7 Heksadesimal bir sayıda bitlerin kaydırılması

Bitler		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Heksadesimal
İlk durum		0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0x3801
1. adım	taşma yok	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0x7002
2. adım	taşma yok	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0xE004
3. adım	taşma var	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0xC008
4. adım	taşma var	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0x8010

Sağa kaydırma '>>' operatörü ile en sağdaki bitler atılır ve soldan boşalan bitler sıfırla (işaretsiz sayı olması durumunda) ya da işaret bitiyle doldurulur (işaretli sayı olması durumunda). Sayının n kadar sağa kaydırılması onun 2^n ile çarpımına denktir.

Aşağıdaki örnekte biri işaretli diğeri işaretsiz iki heksadesimal sayının 4'er adım sağa kaydırılması gösterilmiştir. İşaretli sayıda boşalan yerlere işaret biti '1' gelmekte ve bu işlem 4 adım yürütüldüğü için sayının başında 'F' değeri oluşmaktadır. İşaretsiz sayıda ise boşalan bitlerin yerleri '0' ile doldurulmaktadır.

```
0xFF56 >> 4; /* 0xFFF5'e eşit olur */
0xFF56u >> 4; /* 0x0FF5'e eşit olur */
```

Tablo A.8'de bit kaydırma operatörlerinin işaretli ve işaretsiz tamsayılarda kullanımı ve sonucun LCD ekrandan izlenmesi uygulaması gösterilmiştir. Bu program kodunun çalışması için Tablo 3.11'deki gerekli LCD ayarlamalarını yapmayı unutmayınız. Sonucu izlemek için Şekil 3.7'deki devreyi kullanabilirsiniz.

Tablo A.8 Bit kaydırma operatörlerinin kullanımı ve LCD ekrandan izlenmesi

```
char sayac;
signed char i=-1;
char ii=128;
char str1[16], str2[16];
void main() {
    ayarlar();
    for(sayac=0; sayac<8; sayac++){
        i=i<<1;
        InttoStr(i, str1);
```

```
Lcd_Out(1,1,str1);  
ii=ii>>1;  
InttoStr(ii,str2);  
Lcd_Out(2,1,str2);  
delay_ms(1000);  
}
```

Tablo A.8'deki örnekte -1 ve 128 sayıları iki farklı başlangıç değeri olarak ayarlanmıştır. 'for' Döngüsü 8 defa işletilmekte ve her bir adımda ilgili sayıların değeri 1 basamak kaydırılmaktadır. Bu durumda 8. döngü adımının sonunda i değişkeni $2^7 \times (-1) = -128$, ii değişkeni $128/2^7 = 1$ değerini almış olur.

A-6 Atama Operatörleri

Diğer pek çok programlama dillerinden farklı olarak C dili değer atamasını bir komuttan ziyade bir işlem olarak görür (bir operatör ile temsil edilen).

```
ifade1 = ifade2
```

mikroC'de değer atama işlemi eşittir ('=') simgesiyle gerçekleştirilir. Bu kullanımda mikrodenetleyicinin veri belleğinde "ifade1" olarak yer tutan değişkenin içeriğine "ifade2" olarak yer tutan değişkenin içeriği yüklenmektedir. Bu tür bir atamada veri türleri farklıysa, "ifade2"'nin veri türü "ifade1"'in türüne çevrilir.

Bu kullanımın yanında '=' operatörü, +, -, *, /, %, &, |, ^, <<, ya da >> operatörleriyle birlikte arada boşluk bırakılmadan kullanılır. Böylece +=, -=, *=, /=, %=, &=, |=, ^=, <<= ve >>= olmak üzere 10 farklı birleşik kullanım elde edilir.

Kullanımı;

```
ifade1 op= ifade2
```

Ör:

```
ifade1 += ifade2 veya
```

```
ifade1 <<= ifade2 gibi
```

şeklinde ve burada op ifadesinin yerine diğer operatörler yazılır. Bu kullanım,

```
ifade1 = ifade1 op ifade2
```

Ör:

```
ifade1 = ifade1 + ifade2 veya
```

```
ifade1 = ifade1 << ifade2
```

yazımına eş değerdir.

A-7 Mantıksal (Logical) Operatörler

Mantıksal işlemlerin işlem görenleri (operand) doğru (true) ya da yanlış (false) olarak düşünülür. Mantıksal operatörler her zaman lojik-1 ya da lojik-0 döndürür. Mantıksal bir ifadede işlem görenler sayılarla ifade edilebilir (skalar) olmalıdır. Mantıksal operatörler '&&' ve '||' soldan sağa doğrudur. Mantıksal değil '!' operatörü ise sağdan sola doğrudur. Mantıksal operatörler yaygın olarak 'if' ifadeleri ile yapılan karşılaştırma işlemlerinde ya da 'while' döngülerinde sayacın son bulması için gerekli koşulun sağlanıp sağlanmadığının testinde kullanılır.

Tablo A.9 Mantıksal operatörler

Operatör	İşlem	Öncelik
&&	Mantıksal VE	5
	Mantıksal VEYA	4
!	Mantıksal DEĞİL	14

Mantıksal, ilişkisel ve aritmetik operatörlerin önceliği karmaşık ifadelerin parantez kullanmaksızın gerçekleştirilmesini sağlayacak şekilde tasarlanmıştır. mikroC’de 15 dereceli öncelik durumu vardır.

```
c >= '0' && c <= '9'; /* (c >= '0') && (c <= '9') olarak okunur */
a + 1 == b || ! f(x); /* ((a + 1) == b) || (! (f(x))) olarak okunur */
```

Mantıksal VE '&&' yalnızca her iki ifade de sıfırdan farklıysa '1', aksi durumda '0' lojik değerini döndürür. Eğer ilk ifade 'false' üretirse, ikinci ifade hesaplanmaz. Örneğin:

```
a > b && c < d; /* (a > b) && (c < d) olarak okunur */
/* Eğer (a > b) yanlış-false (0) ise, (c < d) hesaplanmaz */
```

Mantıksal VEYA '||' her iki ifadeden biri de sıfırdan farklıysa 1 döndürür. Eğer ilk ifade 'true' üretirse, ikinci ifade hesaplanmaz. Örneğin:

```
a && b || c && d; /* (a && b) || (c && d) olarak okunur */
/* Eğer (a && b) is doğru-true (1) ise, (c && d) hesaplanmaz */
```

A-8 İlişkisel Operatörler

İlişkisel operatörler eşitlik ya da eşitsizlik durumlarının testi için kullanılır. Eğer bir ifade doğru (true) değeri üretiyorsa, '1', aksi takdirde '0' lojik değerini döndürür. Tüm ilişkisel operatörler soldan sağa doğrudur.

Tablo A.10 İlişkisel operatörler

Operatör	İşlem	Öncelik
==	Eşit	9
!=	Eşit değil	9
>	...’dan daha büyük	10
<	...’dan daha küçük	10
>=	...’dan daha büyük ya da eşit	10
<=	...’dan daha küçük ya da eşit	10

Aritmetik işlemlerle ilişkisel operatörlerin parantez kullanmadan kullanılmasını sağlayan öncelik katsayısı aşağıdaki gibi bir kullanım sunar:

```
a + 5 >= c - 1.0 / e /* → (a + 5) >= (c - (1.0 / e)) */
```

İlişkisel operatörler '1' ya da '0' değeri döndürür. Dolayısıyla aşağıdaki kullanımlarda:

```
/* doğruya karşılaştırma */
5 > 7 /* 0 döndürür */
10 <= 20 /* 1 döndürür */
/* bu yanıltıcı olabilir: */
8 == 13 > 5 /* 0 döndürür, 8 == (13 > 5) → 8 == 1 → 0 */
```

14 > 5 < 3 /* 1 döndürür, (14 > 5) < 3 → 1 < 3 → 1 */
a < b < 5 /* 1 döndürür, (a < b) < 5 → (0 ya da 1) < 5 → 1 */

A-8 İkili (Binary) ve Onaltılık (Hexadecimal) Sayı Sistemi

Mikrodenetleyicilerin portlarının kontrolünde ikili ve onaltılık sayı sisteminden çok yararlanır. Ayrıca bilindiği gibi mikrodenetleyicinin program belleğine ve veri belleğine bilgiler onaltılık sayı sistemiyle gönderilmektedir.

mikroC’de ikili bilgi ‘0B’ tanımlayıcısıyla belirtilir. İkili bilgiyle ilgili portun her bir bacağına açık şekilde kontrol etmek mümkündür. Örnek olarak,

```
TRISB = 0B00000000; // B portunun tüm pinleri çıkış yapıyor
PORTB = 0B00000000; // B portunun tüm pinleri temizleniyor
```

şeklinde yazılır.

mikroC’de onaltılı bilgi ‘0x’ tanımlayıcısıyla belirtilir. Örnek olarak,

```
TRISB = 0x01; // B portunun RBO-INT girişi hariç diğer tüm pinleri çıkış yapıyor
PORTB = 0x00; // B portunun tüm pinleri temizleniyor
```

Verilen örneklerde 8 basamak ikili ve 2 basamak onaltılı sayıların olması, kullanılan mikrodenetleyicinin 8 bitlik olmasından dolayıdır. 8 bitlik bir mikrodenetleyicide veriler 8 bitlik paketler halinde işlenir. Verilen örneklerde ve yapılan uygulamalarda ağırlıklı olarak 8 bitlik mikrodenetleyiciler kullanıldığı için sayı işlemleri de 8 bit üzerinden yapılmaktadır.

8 bitlik bir veri ilk dört bit ve ikinci dört bit olarak iki grupta değerlendirilir. Tablo A.11’de 5 adet sayının bit ağırlıkları, bunların ikili gösterimi, onaltılık ve onlu karşılıkları verilmiştir.

Tablo A.11 8 bitlik bir bilginin ikili, onaltılık ve onlu karşılıkları

Bitler	İkinci Dört Bit				İlk Dört Bit				Karşılıklar		
	7	6	5	4	3	2	1	0	İkili (Binary)	Onaltılık (Hexadecimal)	Onlu (Decimal)
İki tabanında kuvvetleri	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰			
Değerleri	128	64	32	16	8	4	2	1			
Örnekler	0	1	1	0	0	1	0	1	0B01100101	0x65	101
	0	0	0	0	0	0	1	0	0B00000010	0x02	2
	0	0	1	1	0	1	0	0	0B00110100	0x34	52
	1	0	0	0	1	0	0	0	0B10001000	0x88	136
	1	1	0	1	0	0	1	1	0B11010011	0xD3	211

İkili bir sayının onlu karşılığı her bite karşılık gelen bit ağırlığının toplanmasıyla bulunur. Bilgi 4’er bitlik paketler halinde bölündüğü zaman hızlı şekilde onaltılık karşılığı da elde edilir. Tablo A.12’de 0-15 arası sayıların ikili ve onaltılık karşılıkları verilmiştir.

Tablo A.12 0-15 arası sayıların ikili ve onaltılık karşılıkları

Onlu	İkili	Onaltılık
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Tablodan görüldüğü üzere 4 bitlik bir sayı dizisi için 16 farklı durum söz konusudur. 8 bitlik bir sayı sistemi için $16 \times 16 = 256$ (0 – 255) farklı durum ortaya çıkar. Bu mantıkla bit ağırlığına göre üretilebilecek en üst sayı değerini hesaplayabilirsiniz. Örneğin 'int' tamsayı değişken tipi hafızada 2 Byte'lık yer tutar. Bir Byte 256 sayı değeri tuttuğuna göre;

$256 \times 256 = 65536$ farklı sayı değerine karşılık gelir. Dolayısıyla işaretli bir tamsayı değişkeni için en yüksek sayı değerinin 65535 olacağı ortaya çıkar.

A-9 mikroC'de Aritmetik Veri Türleri

mikroC'de de diğer tüm programlama dillerinde olduğu üzere çeşitli aritmetik veri türleri vardır. `void`, `char`, `int`, `float` ve `double`, aritmetik veri türleridir. Bunların yanında `short`, `long`, `signed` ve `unsigned` öntakısı kullanılarak farklı değer aralığında değişkenler elde edilebilir.

Aritmetik veri türleri "tamsayı-integral" ve "kayan nokta – floating point" olmak üzere iki türedir. Tablo A.13'te tamsayı aritmetik veri türleri gösterilmiştir. Aritmetik veri türleri tuttukları sayı değerinin negatif ve pozitif değer aralığında ya da yalnızca pozitif olması durumuna göre `signed` ya da `unsigned` öntakısı alır. Tabloda veri türlerinin önlerinde yer alan parantez içindeki öntakıların kullanımı zorunlu değildir. Örneğin `char` veri türü zaten işaretli olduğundan ayrıca önüne `unsigned` öntakısını koymaya gerek yoktur. Ancak negatif değer de tutacak `char` veri türünde bir değişken oluşturmak istiyorsanız bu durumda `signed` öntakısını da yazmanız gerekir.

Tablo A.13 Tamsayı aritmetik veri türleri

Tür	Byte cinsinden boyutu	Sayı aralığı
<code>bit</code>	1-bit	0 or 1
<code>sbit</code>	1-bit	0 or 1
<code>(unsigned) char</code>	1	0 .. 255
<code>signed char</code>	1	- 128 .. 127

Tür	Byte cinsinden boyutu	Sayı aralığı
(signed) short (int)	1	- 128 .. 127
unsigned short (int)	1	0 .. 255
(signed) int	2	-32768 .. 32767
unsigned (int)	2	0 .. 65535
(signed) long (int)	4	-2147483648 .. 2147483647
unsigned long (int)	4	0 .. 4294967295

short ve long veri tanımlayıcıları yalnızca int veri türüne uygulanabilir. Bu durumda int veri türünün yazımı zorunlu değildir. short ile 1 byte uzunluğunda, long ile 4 byte uzunluğunda tamsayı değişkenler üretilir.

Küsüratlı sayı değerleriyle çalışılmak istendiğinde kayan nokta aritmetik veri türlerinin kullanımı gerekir. Bu durumda float, double ve long double olarak üç tür kayan nokta veri türü vardır. Ancak ANSI standartlarının “mikroC PRO for PIC” kullanımında bu üç veri türü de aynıdır. Tablo A.14’te de görüleceği üzere her üçünün de boyutları ve sayı aralıkları aynıdır.

Tablo A.14 Kayan nokta aritmetik veri türleri

Tür	Byte cinsinden boyutu	Sayı aralığı
float	4	$-1.5 * 10^{45} .. +3.4 * 10^{38}$
double	4	$-1.5 * 10^{45} .. +3.4 * 10^{38}$
long double	4	$-1.5 * 10^{45} .. +3.4 * 10^{38}$

A-10 mikroC’de Fonksiyonlar

mikroC’de fonksiyonlar bir dizi işlemin program içinden sık sık çağrılması istendiğinde ya da belli işlemlerin bir grup altında yazılması istendiğinde kullanılan program öbekleridir. Fonksiyonlar değer döndüren ya da döndürmeyen olarak iki ana gruba ayrılır. Değer döndürmeyen fonksiyonlar için “void” tanımlayıcısı kullanılır. mikroC’de “main()” ana fonksiyonu değer döndürmediği için başında “void” tanımlayıcısı kullanılır. Tablo A.15’te kitap içindeki çeşitli uygulamalarda sık sık kullandığımız “ayarlar()” fonksiyonunun PIC16F887 için yazımına örnek verilmiştir.

Tablo A.15 Değer döndürmeyen fonksiyon örneği

```
void ayarlar(){
    Keypad_Init(); // Tuş takımı başlatılıyor
    Lcd_Init();    // LCD başlatılıyor
    Lcd_Cmd(_LCD_CLEAR);
    Lcd_Cmd(_LCD_CURSOR_OFF);
    INTCON=0;
    OPTION_REG = 0X87;
    TRISA=0X00; // Portaların giriş-çıkış tanımlamaları yapılıyor
    PORTA=0; // Portlara başlangıç değerleri atanıyor
    TRISC = 0x00;
    PORTC = 0;
    TRISE=0X00;
    PORTE=0;
    ANSEL = 0; // AN pinlerini dijital I/O olarak ayarla
    ANSELH = 0;
```

```
CM1CON0.C1ON=0;    // Karşılaştırmaları kapat
CM2CON0.C2ON=0;
}
```

“ayarlar()” isimli bu fonksiyonun “main()” ana fonksiyonu içinden çağrılması için “main()” kod satırının üstünde yazılmış olması gerekir. Aksi durumda, derleşem işlemi yaptığınızda “**Undeclared identifier ‘ayarlar’ expression**” hata mesajı alırsınız.

Değer döndüren fonksiyonlara genellikle çeşitli değer ya da değerler de gönderilir. Ayrıca döndürülen değer türü neyse “void” yerine o değer türü yazılır. Tablo A.16’da kendisine gönderilen iki tamsayıyı alıp toplama işlemi yapan ve sonucu tamsayı olarak döndüren bir fonksiyon örneği gösterilmiştir. Bunun yanında aynı örnek içinde kendisine gönderilen iki sayının büyük olanını döndüren fonksiyon da vardır. Her iki fonksiyonun döndürdüğü değerler LCD ekrana yazdırılmaktadır.

Tablo A.16 Değer döndüren fonksiyon örneği

```
int toplam, maksimum;
char top[7], mak[7];
int toplama(int a, int b){
    int sonuc;
    sonuc = a+b;
    return sonuc;
}

int maks(int x, int y) {
    return (x>=y) ? x : y; /* karşılaştırma sonucunda x büyükse x'i,
değilse y'i döndüren ? ... : yapısı */
}

void main(){
    toplam=toplama(123,25);
    maksimum=maks(-21,15);
    IntToStr(toplam,top);
    IntToStr(maks,mak);
    Lcd_Out(1,1,top);
    Lcd_Out(2,1,mak);
}
```

Örnekleri inceleyecek olursak, ana program içinden iki fonksiyonun da çağrıldığı görülmektedir. Fonksiyonların içinde tanımlanmış değişkenler yerel (local) değişkendir. O değişken isimleri yalnızca kullanıldığı fonksiyonun kendi içinde çağrılabilir. Başka bir fonksiyon içinden çağrılmaları söz konusu değildir. Tanımlanan değişkenlerin her yerden çağrılabilmesi için evrensel tanımlama yapılmalıdır. Programın en başında “toplam” ve “maksimum” isimli iki evrensel değişken tanımlanmıştır. Bu değişkenlere, fonksiyonların döndürdüğü değerler atanmaktadır. Ardından tamsayı değerlerinin LCD ekranda yazdırılması için Integer-String dönüşümü yapılmaktadır. Tamsayı değişkenlerinin dönüşümü için kullanılan 7 karakterli metin değişkenlerine dönüştürülen değerler atanmaktadır. Böylece LCD ekrana yazdırma işlemi yapılabilir.

EK-B

MIKROC PROGRAM EDİTÖRÜNÜN ARAÇLARI VE KULLANIMI

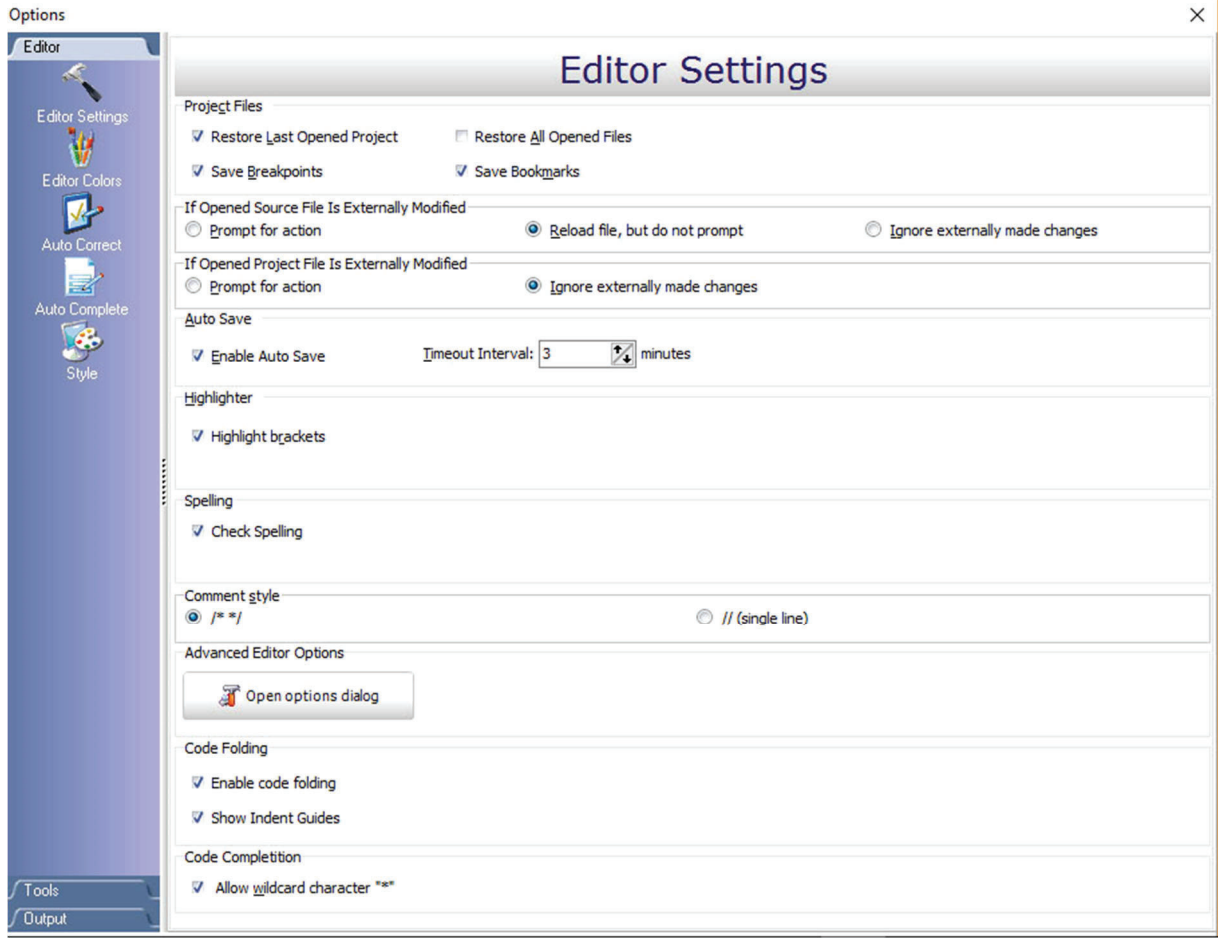
mikroC program editöründe çeşitli işlemleri kolay ve pratik şekilde gerçekleştirmemizi sağlayan bir dizi kısayol tuşu vardır. Program araçlarından bahsetmeden önce bu kısayol tuşlarını bilmeniz mikroC program editörünü daha verimli kullanmanızı sağlayacaktır.

Tablo B.1 mikroC Program Editörü Kısayolları

IDE Kısayolları		İleri Düzey Editör Kısayolları	
F1	Yardım	Ctrl+Space	Kos Asistanı
Ctrl+N	Yeni Kod Sayfası Açar	Ctrl+Shift+Space	Parametre Asistanı
Ctrl+O	Aç	Ctrl+D	Bildirim Bul
Ctrl+Shift+O	Proje Aç	Ctrl+E	Artımlı Arama
Ctrl+Shift+N	Yeni Proje	Ctrl+L	Rutin (Fonksiyon) Listesi
Ctrl+K	Projeyi Kapat	Ctrl+G	Satıra Git
Ctrl+F4	Kod Sayfasını Kapatır	Ctrl+J	Kod Sihirbazı Ekle
Ctrl+Shift+E	Proje Düzenleme Penceresi	Ctrl+Shift+.	Yorum Kodu
Ctrl+F9	Derleme Yapar	Ctrl+Shift+,	Yorum Kodunu Kaldır
Shift+F9	Tümünü Derler	Ctrl+number	Yer İmine Git
Ctrl+F11	Derler ve Programlar	Ctrl+Shift+number	Yer İmi Ayarla
Shift+F4	Kesme Noktalarını İzler	Ctrl+Shift+I	Satırbaşı Sekmesi
Ctrl+Shift+F5	Kesme Noktalarını Temizler	Ctrl+Shift+U	Sekme Kaldırma
F11	Programlayıcıyı Başlatır	TAB	Satırbaşı Sekmesi
Ctrl+Shift+F1	Proje Yöneticisi	Shift+TAB	Sekme Kaldırma
F12	Seçenekler	Alt+Select	Sütun Olarak Seçim
Alt+X	Programı Kapatır	Ctrl+Alt+Select	Sütun Olarak Seçim
		Ctrl+Alt+L	Seçileni Küçük Harf Yap

Temel Editör Kısayolları		Hata Ayıklayıcı ve Yazılım Simülatörü Kısayolları	
F3	Bul, Sonrakini Bul	F2	Kesmeye Atlar
Shift+F3	Öncekini Bul	F4	Kursörü Çalıştırır
Alt+F3	Arama Penceresini Açar	F5	Kesme Noktasını Değiştirir
Ctrl+A	Tümünü Seç	F6	Hata Ayıklayıcıyı Çalıştır/Durdur
Ctrl+C	Kopyala	F7	İçine Adım
Ctrl+F	Bul	F8	Üstüne Adım
Ctrl+R	Yerdeğiştir	F9	Hata Ayıklayıcıyı Başlat
Ctrl+P	Yazdır	Ctrl+F2	Hata Ayıklayıcıyı Durdur
Ctrl+S	Kod Sayfasını Kaydet	Ctrl+F5	İzleme Listesine Ekle
Ctrl+Shift+S	Tümünü Kaydet	Ctrl+F8	Dışarı Adım
Ctrl+V	Yapıştır	Alt+D	Assembly Kod İzleme
Ctrl+X	Kes	Shift+F5	İzleme Penceresini Aç
Ctrl+Y	Tüm Satırı Sil	Ctrl+Shift+A	İleri Düzey Kesme Noktalarını Göster
Ctrl+Z	Geri Al		
Ctrl+Shift+Z	Yinele		
Ctrl+Alt+U	Seçileni Büyük Harf Yap		
Ctrl+Alt+T	İlk Harfleri Büyütür		

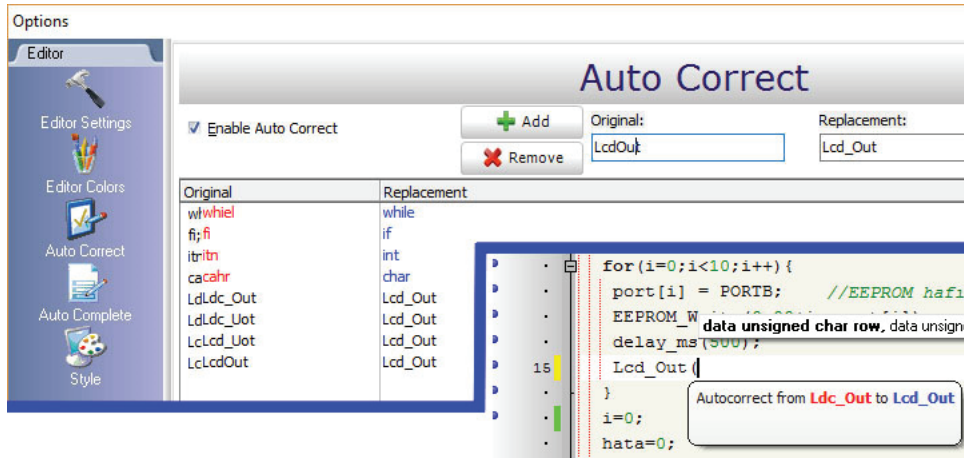
mikroC programının en önemli özelliği kullanıcıya işini kolaylaştırıcı çok sayıda araç sunmasıdır. mikroC’de program yazımı bu araçlar sayesinde daha keyifli ve hızlı gerçekleştirilebilir. mikroC’de kullanılan araçlara erişim menü çubuğundaki “Tools” menüsünden gerçekleştirilir. Şekil B.1’de “Tools→Options (Tercihler)” ya da F12 ile açılan “Editor Settings – Editör Ayarları” penceresi görülmektedir.



Şekil B.1 Editör ayarları penceresi

Bu pencereden program editörünüzün açılıştaki varsayılan ayarlarını değiştirebilirsiniz. Kodlamada kullanılan renklerden, otomatik yorum satırına çevirmede kullanılacak yorum satırı tipi, otomatik kaydetme, kesme noktaları, sekmeler ve yer imleriyle ilgili ayarlar gibi pek çok ayarlamayı bu pencereden yapabilirsiniz.

Editör ayarları penceresi üzerinden yapabileceğiniz en önemli ayar, kod yazımınızı hızlandıracak olan “Auto Correct – Otomatik Düzelt” ve “Auto Complete – Otomatik Tamamla” seçenekleridir. Yoğun kod yazımlarında bazen çok temel ve çok sık kullanılan kodları yazarken harf hataları yapabilirsiniz. Bildiğiniz üzere programlamada tek bir karakter hatası bile derleme hatasına neden olabilir ya da programınızın öngördüğünüz şekilde çalışmasını engelleyebilir. Programlamada çok sık kullandığınız ‘while’, ‘if’, ‘for’, ‘void’, ‘double’, ‘float’, ‘int’, ‘signed’, ‘unsigned’, ‘PORTA’, ‘PORTB’, ‘PORTC’, ‘PORTD’, ‘Lcd_Out’, ‘Lcd_Cmd’ gibi daha pek çok kod ögesinin yanlış yazılma durumunda otomatik olarak düzeltilmesi için “Auto Correct” menüsünü kullanabilirsiniz. Şekil B.2’de otomatik düzeltme penceresinden bir kesit ve uygulama sonucunun bir kod parçasından kesiti gösterilmiştir.



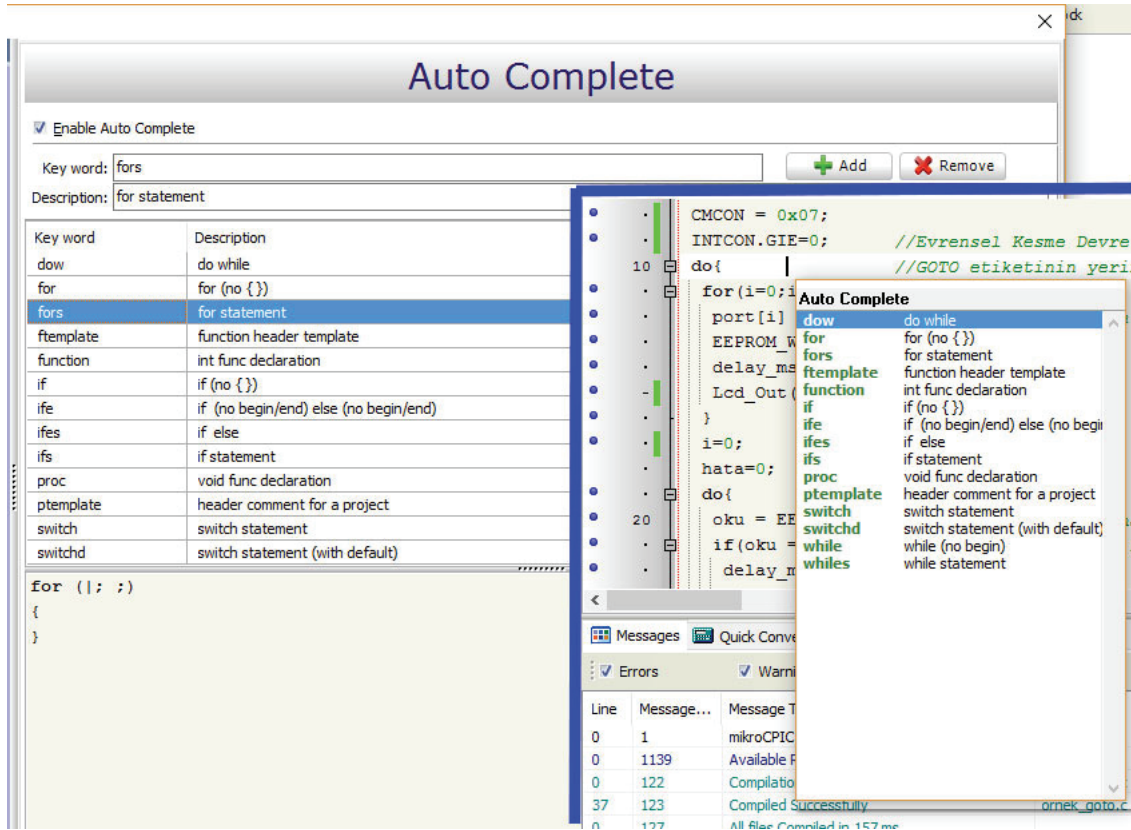
Şekil B.2 Otomatik Düzeltme penceresi ve örnek düzeltme uygulaması

Şekildeki örnekte 'Lcd_Out' komutunun 'Ldc_Out', 'Lcd_Uot', 'LcdOut' gibi yanlış yazımlarda düzeltilmesini sağlamak için "Original" ve "Replacement" alanlarına veri girilmiştir. "Original" yazan metin kutusuna düzeltilen hatalı sözcük girilir. "Replacement" yazan metin kutusuna ise doğru olan sözcük girilir. Her bir düzeltme alternatifi için ayrı giriştir ve "Add" butonuna tıklanır. Şekilde görüldüğü üzere Lcd_out komutu kasıtlı olarak Ldc_Out olarak yazılmış ve parantez açıldığı anda otomatik düzeltme uyarısı gelmiştir. Bu özelliği kullanarak kendi otomatik düzeltme kütüphanenizi oluşturabilirsiniz.

Kod yazımınızı hızlandıracak bir diğer uygulama "Options" penceresinden erişilebilen otomatik tamamlama özelliğidir. Program yazımında çok kullandığınız kodları "Ctrl+J" tuş ikilisiyle hızlıca kod yazım alanına ekleyebilirsiniz. Şekil B.3'te böyle bir komut sonrası açılan pencerede seçebileceğiniz otomatik tamamlama kodları listelenmiştir.

"Auto Complete" penceresini kullanarak listeye yeni kodlar ekleyebilirsiniz. Ya da yazdığınız kodların başında her zaman olmasını istediğiniz "header-başlık" denilen ve genellikle programı yazanın künyesinin olduğu bir sabit metin eklemek istiyorsanız bu özelliği kullanabilirsiniz. Şekilde örnek olarak "for statement" yapısının nasıl eklendiği görülmektedir. Eklemek istediğiniz ifadenin içeriğini pencerenin alt bölümünde yer alan metin yazım alanına yazıp "Add" butonuna tıklayarak eklersiniz.

Bu pencereler açıkken yapmış olduğunuz değişikliklerin kalıcı olabilmesi için, pencereleri kapattığınızda karşınıza çıkan "Apply Changes? – Değişiklikler Uygulansın mı?" penceresinde "Do you want to apply changes? – Değişiklikleri uygulamak istiyor musunuz?" sorusuna Evet cevabını tıklamanız gerekmektedir.



Şekil B.3 Otomatik tamamlama özelliği örnek uygulaması

mikroC'nin araçlar menüsünden ulaşılabilen bazı araçları şu şekilde listeleyebiliriz:

- **mE Programmer (F11):** mikroC programının kurulumu sırasında yüklemek isteyip istemediğinizin sorulduğu "mikroProg Suite For PIC" programının başlatıcısıdır. Microchip® firmasının yüzlerce farklı PIC mikrodenetleyicisine HEX kodunu yazmak için kullanılan programlayıcıyı kontrol eder. mikroElektronika® firmasınınca hangi model cihaz için hangi mikrodenetleyicilerin programlanabileceği belirtilmiştir. Ayrıntılı ve güncel bilgi için <http://www.mikroe.com/mikroprog/pic-dspic-pic32/> adresini ziyaret edebilirsiniz. Bu sayfada PIC®, dsPIC®, PIC24® ve PIC32® ailesi mikrodenetleyiciler için tek donanımda programlama olanağı sunan cihazla ilgili ayrıntılı bilgi bulunmaktadır.
- **MikroProg Power Options (Ctrl+Alt+P):** Mikrodenetleyicinin programlanması sırasında kullanılacak voltaj değerini ayarladığınız araçtır.
- **Active Comment Editor:** Programın ilgili yerine dikkat çekici bir yazı formatında yorum yazmanızı ve yorumu çeşitli şekillerde düzenleme fırsatı sunan araçtır. Yorumun eklendiği yerin başına otomatik olarak açıklama satırı "//" karakteri eklenir.
- **Ascii Chart:** ASCII karakter tablosu aracıdır. Bilindiği üzere ASCII karakter tablosu 8 bitlik ve dolayısıyla 0-255 arası toplam 256 karakterin desimal ve heksadesimal olarak kodlandığı bir karakter tablosudur. Programlarınızda desimal ya da heksadesimal olarak yazdırmak istediğiniz karakterlerin kodlarını bu tablodan yararlanarak öğrenebilirsiniz.
- **EEPROM Editor:** Bu araç yardımıyla üzerinde çalıştığınız mikrodenetleyicinin EEPROM veri belleğinde düzenleme yapabilirsiniz. ASCII karakter tablosundan yararlanarak veri belleğinin istediğiniz bölgesine 8 bitlik bir karakter yazabilirsiniz. Yaptığınız değişikliklerin mikrodenetleyicinin veri belleğine yazılabilmesi için mikroElektronika®'nın programlayıcı

cihazının çalışır durumda olması gerekmektedir. EEPROM düzenleyicisinde istediğiniz düzenlemeyi yaptıktan sonra "Save" butonuyla kaydetme işlemi yaptığınızda intel hex dosyası (.ihex) oluşturulur. Programlama cihazı tarafından bu dosyanın manuel olarak yüklenmesi gerekmektedir.

- **Export Code to HTML:** Bu araç yardımıyla kodunuzu web sayfalarınız için yayına hazır hale getirebilirsiniz. Program kodunuzun yer aldığı dosya klasöründe .html uzantılı bir dosya oluşur.
- **GLCD Bitmap Editor:** KS0108, T6963 ve Nokia® 3310 Grafik LCD'leri için Bitmap formatında monokrom (tekrenk) desen düzenlemesi yapmanızı sağlayan araçtır. Dilerseniz standartlara uygun hazır bir dosya ya da bir Bitmap programında sizin hazırlayacağınız bir dosyayı seçip yükleyebileceğiniz gibi, çizmek istediğiniz deseni program arayüzünü kullanarak da hazırlayabilirsiniz. Pencerenin "Generate Code" alanında kodunuzda kullanabileceğiniz kodun üretildiğini görürsünüz. Bu kodu kopyalamak suretiyle üzerinde çalıştığınız .c dosyasına yapıştırabilirsiniz.
- **HID Terminal:** USB haberleşmesi için bir haberleşme terminalidir. USB arabirimi ile bilgisayarınıza bağlanan PIC mikrodenetleyicili bir cihaza veri yollamak için kullanılır. Doğrudan USB desteği sunan 18F4550 gibi bir mikrodenetleyici ile yapacağınız HID donanımına bilgisayarınızdan mesaj yollayabilirsiniz.
- **Interrupt Assistant:** Program içinde kendi kesme rutinlerinizi düzenlemek için kullanacağınız bir yardımcı araçtır. mikroC ile kesmelerle çalışırken normalde "interrupt()" rutini kullanılır. Ancak yeni rutin isimleriyle kesmelerinizi farklı rutinlerde işletmeniz mümkündür.
- **LCD Custom Character:** LCD ekran için 5x7 ya da 5x10 karakter boyutlarında nokta bazında istediğiniz karakteri oluşturmak için kullanabileceğiniz bir araçtır. Özellikle Türkçe karakterleri yazdırma olanağımızın olmadığı metin tabanlı LCD ekranlarda, dilediğimiz karakteri yazdırabilmek için çok iyi bir araçtır. Ancak 2x16 formatında karakter LCD ekranda 5x7 formatını kullanmamız gerekir. Deseni fare ile tıklamak suretiyle oluşturduktan sonra "Generate Code" butonu ile karakter kodu üretilir. "Copy to Clipboard" ile üretilen karakter kodu panoya kopyalanır. Kopyalama işleminden sonra karakter kodunu kendi programınızda kullanabilirsiniz.

Örnek:

Tablo B.2 2x16 Satır LCD için özel karakter üretilmesi

```

: sbit LCD_RS at RB2_bit; //LCD bağlantıları ayarlanıyor
: sbit LCD_EN at RB3_bit;
: sbit LCD_D4 at RB4_bit;
: sbit LCD_D5 at RB5_bit;
: sbit LCD_D6 at RB6_bit;
: sbit LCD_D7 at RB7_bit;
:
: sbit LCD_RS_Direction at TRISB2_bit;
: sbit LCD_EN_Direction at TRISB3_bit;
: sbit LCD_D4_Direction at TRISB4_bit;
: sbit LCD_D5_Direction at TRISB5_bit;
: sbit LCD_D6_Direction at TRISB6_bit;
: sbit LCD_D7_Direction at TRISB7_bit;
:
: const char character[] = {10,0,14,17,17,17,14,0};
: void CustomChar(char pos_row, char pos_char) {
:     char i;
```

```

    Lcd_Cmd(64);
    for (i = 0; i<=7; i++) Lcd_Chrcp(character[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
    Lcd_Chrcp(pos_row, pos_char, 0);
}
void main() {
    Lcd_Init(); // Lcd_Init PORTB LCD için hazırlandı
    Lcd_Cmd(_LCD_CURSOR_OFF); // LCD kursör kapatıldı
    Lcd_Cmd(_LCD_CLEAR); /* LCD'de rastgele karakter oluşmaması
    için silindi */
    CustomChar(1,1);
}

```

Tablo B.2'de LCD karakter üretme aracı kullanılarak elde edilen 'ö' harfinin kullanımını gösteren bir program örneği görülmektedir. CustomChar(1,1) rutini ana rutin içinde kullanılarak 1.satır 1.sütuna tasarlanan karakter yazdırılmaktadır. Burada pos_row ve pos_char değişkenlerine yolladığınız sayı değerlerini değiştirerek LCD ekranın farklı noktalarına yazabilirsiniz.

- **mikroBootloader:** PIC16F87X ailesi mikrodenetleyiciler kendi program hafızalarına yazabilme kabiliyetine sahiptir. Bu özellik hafızaya yeni bir firmware yazabilen küçük bir önyükleyiciye (bootloader) imkân sağlar. En basit formunda, yeni bir firmware indirilmesi gerektiğini görmüyorsa önyükleyici kullanıcı kodunu çalıştırmaya başlar. Eğer indirilecek yeni bir firmware varsa veriyi alır ve onu program hafızasına yazar. Bir önyükleyicinin daha yaygın özellikleri şu şekilde sıralanabilir:

- Mikrodenetleyicinin Reset konumuna kod yazabilme (Böylece kesme rutinlerini kullanmak zorunda kalmadan, Resetleme işlemiyle birlikte mikrodenetleyicinin Reset vektör adresini kullanan bir program yükleyebilirsiniz).
- Küçük bir bellek alanında başka yerlere kod yazabilme.
- Kullanıcının yeni kullanıcı kodu yüklemek isteyip istemediğini kontrol eder.
- Yeni kullanıcı kodu yüklenmeyecekse kullanıcı kodunun yürütülmesini başlatır.
- Kod yüklenecekse yeni bir iletişim kanalıyla yeni kullanıcı kodunu alır.
- Yeni kullanıcı kodunu belleğe programlar.

Önyükleme kodu çoğunlukla Reset konumunu ve bir miktarda ek program belleğini kullanır. Kesmeleri kullanmaya ihtiyaç duymayan basit bir kod parçasıdır. Bu nedenle, kullanıcı kodu normal kesme vektörü 0x0004'ü kullanabilir. Önyükleme kodunun kesme vektörünü kullanmaması gerekir. Bu nedenle 0x0000 ile 0x0003 adresleri arasında bir program dallanmasına sahip olmalıdır. Önyükleme programı klasik programlama teknikleriyle belleğe programlanmalıdır ve konfigürasyon bitleri bu sırada programlanmalıdır. Konfigürasyon bitleri program belleği alanında adreslenmediğinden önyükleme kodu konfigürasyon bitlerine erişemez.

- **Seven Segment Editor:** 7 Segmentli göstergede görmek istediğiniz karakterleri desimal ve hegzadesimal değerlerde elde etmenizi sağlar.
- **UDP Terminal:** Verileri bağlantı kurmadan yollamak için kullanılan ve IP protokolü üzerinden çalışan bir veri aktarım katmanı olan UDP'yi açmayı ve kullanmayı sağlayan araçtır.
- **USART Terminal:** Senkron ya da Asenkron olarak seri iletişim protokolü üzerinden iletişim kurmayı sağlayan seri iletişim portlarını kullanan araçtır.
- **USB HID Bootloader:** Önyükleme programının yaptığı işi USB arabirimi üzerinden gerçekleştiren araçtır.

EK-C

PIC MİKRODENETLEYİCİLERİNDE PORTLARA YÜK BAĞLANTISI VE PORT KONTROLÜ

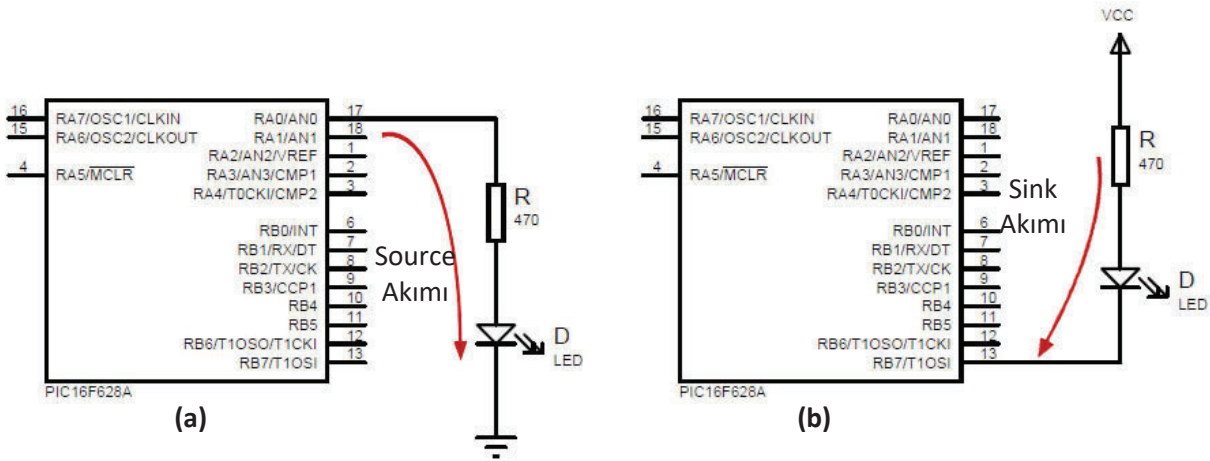
Mikrodenetleyicilerin ilgili portlarına yük bağlantısı gerçek devre tasarımlarında çok dikkat edilmesi gereken bir konudur. Mikrodenetleyicinin her bir portunun pinlerinden geçebilecek en yüksek akım değeri üretici tarafından belirtilmiştir. PIC mikrodenetleyici portlarının sürebileceği akım değeri LED gibi doğrudan yük bağlantılarında hesaba katılmalıdır.

C-1 Mikrodenetleyicilerde Sink ve Source Akımları

Mikrodenetleyici portlarına yük bağlantısında kullanılacak iki yöntem vardır. Bu yöntemlerden biri çıkış portu ile GND (devre şasesi) arasına yük bağlanmasıdır (Bkz. Şekil C.1-a). Bu şekilde porttan çekilen akıma “Kaynak (Source) Akımı” denir. Kaynak akımı MCU modeline bağlı olarak 20mA ya da 25mA kadardır.

Diğer yöntemde ise yük, çıkış portu ile V_{CC} (genellikle +5V) arasına bağlanır (Bkz. Şekil C.1-b). Bu durumda kaynaktan porta ve oradan devre şasesine akım geçişi olur. Besleme kaynağından çekilen bu akıma “Batarya (Sink) Akımı” denir ve 25mA kadardır.

Her bir pin için çekilebilecek en yüksek akım değeri 25mA kadarken, o pinin ait olduğu porttan toplu halde aynı anda çekilebilecek akım genellikle 200mA kadardır.



Şekil C.1 Mikrodenetleyici portlarında sink ve source akımları

Yaygın olarak Şekil C.1-a'da gösterilen bağlantı yöntemi kullanılmaktadır. Bu durumda yükün aktif edilmesi için ilgili portun çıkışı lojik-1 yapılmalıdır. Diğer durumda ise lojik-0 ile çıkış aktif yapılır.

Şekil C.1-a'da gösterilen uygulama devresinde çıkışı aktif yapmak için aşağıdaki yöntemlerden istenilen kullanılabilir.

Tablo C.1 Kaynak akımı yöntemiyle port kontrolü

```
void main() {  
    CMCON = 0X07; //Karşılaştırıcılar kapatılıyor  
    TRISA = 0;
```



```

PORTA.B0 = 1; //Doğrudan RA0 pininin yüklenmesi
//PORTA = 1; //PORTA'nın Desimal olarak yüklenmesi
//PORTA = 0B00000001; //PORTA'nın Binary olarak yüklenmesi
//PORTA = 0X01; //PORTA'nın Hekzadesimal olarak yüklenmesi
//PORTA.F0 = 1;
}

```

Şekil C.1-b'de gösterilen uygulama devresinde çıkışı aktif yapmak için aşağıdaki yöntemlerden istenilen kullanılabilir.

Tablo C.2 Batarya akımı yöntemiyle port kontrolü

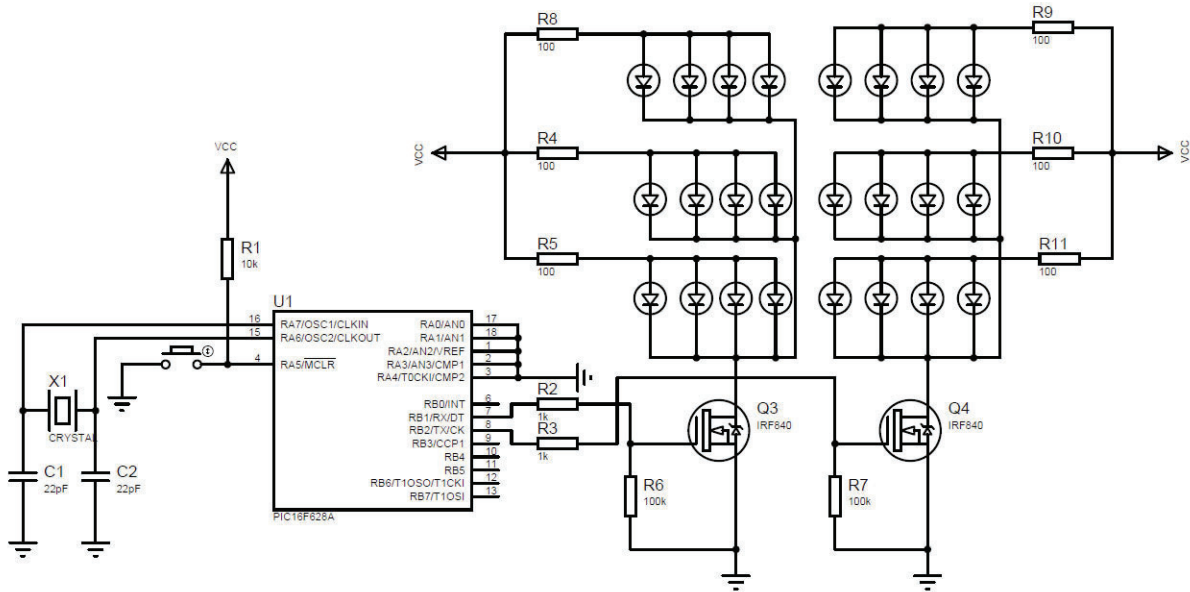
```

void main() {
CMCON = 0X07; //Karşılaştırıcılar kapatılıyor
TRISB = 0;
PORTB.B7 = 0; //Doğrudan RB7 pininin yüklenmesi
//PORTB = 0; //PORTB'nin Desimal olarak yüklenmesi
//PORTB = 0B00000000; //PORTB'nin Binary olarak yüklenmesi
//PORTB = 0X00; //PORTB'nin Hekzadesimal olarak yüklenmesi
}

```

C-2 Mikrodenetleyicilerde Anahtarlama Elemanlarının Kullanımı

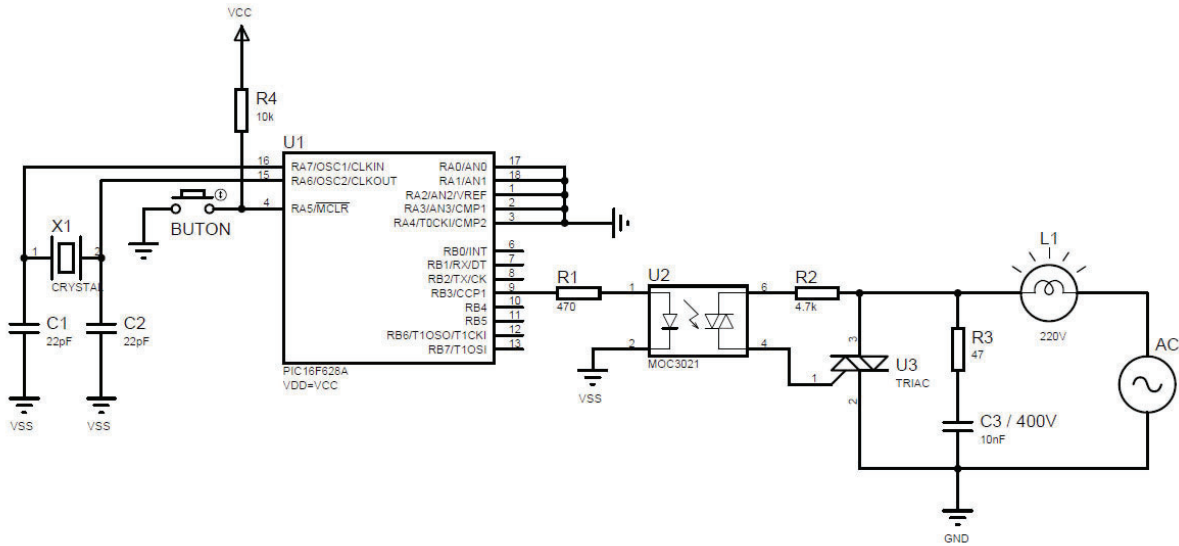
Mikrodenetleyiciler profesyonel ve endüstriyel tip uygulamalarda elektronik tetikleme elemanlarıyla kullanılırlar. Portların daha yüksek akım çeken yükleri kontrol etmesinin istendiği yerlerde transistör ya da mosfet kullanılır. Tristör ve triyak gibi AC akım kontrolü gerçekleştirebilen elemanların sürülmesi için de optokuplör kullanılarak voltaj yalıtımı sağlanmalıdır. Şekil 3.5'te uygulaması gösterilen flaşör devresinin mosfet ile gerçekleştirilmiş hali Şekil C.2'de gösterilmiştir.



Şekil C.2 PIC ile MOSFET üzerinden yük sürülmesi

Şekildeki devrede IRF840 MOSFET'i kullanılmıştır ve MOSFET 1kΩ'luk direnç üzerinden tetiklenmiştir. MOSFET'in giriş ucunun boşa bırakılması durumunda kararsız çalışmasını önlemek için Gate ile şase arasına 100kΩ'luk direnç bağlanmıştır. Bu devrede kullanılan MOSFET'in kanal (drain) akımı 8A kadardır ve çok sayıda LED'den oluşan bir aydınlatma grubunu rahatlıkla kontrol edebilir. Power LEDler kullanarak yapılacak bir uygulama için çok uygundur.

Şekil C.3'te Mikrodenetleyici ile triyak üzerinden bir yükün sürülmesi gösterilmiştir. Örnek uygulamada MOC3021 optokuplör elemanı kullanılmıştır. Tristör ve triyak gibi anahtarlama elemanlarının kontrolünde optokuplör kullanılarak mikrodenetleyicinin kontrol edilecek yüksek akım devresinden yalıtımı yapılmış olur. PIC16F628A'nın RB3 portunda yer alan CCP1 modülü kullanılarak PWM sinyali elde edilmiştir. Bunun için mikroC'nin PWM kütüphanesi kullanılmıştır.



Şekil C.3 Mikrodenetleyici ile triyak üzerinden bir yükün sürülmesi

Tablo C.3'te triyak ile AC akımda lambanın kontrolü için kullanılacak kod gösterilmiştir. Gerçekte AC yükün bu şekilde kontrolü kıpraşma (flickering) dediğimiz bir sorunun ortaya çıkmasına neden olur. Kıpraşma etkisinin en aza indirgenmesi amacıyla Şekil 4.10'da uygulaması gösterilmiş devre örneği kullanılmalıdır. Söz konusu uygulamada ikinci bir optokuplör yardımıyla sıfır nokta tespiti (zero-crossing) tekniği gerçekleştirilmiştir. Böylece, AC şebeke geriliminin eksenine sıfıra yakın noktalarda kesmesi anında PWM sinyalinin kesilerek triyakin kesime gitmesi sağlanmıştır.

Tablo C.3 Mikrodenetleyicide PWM tekniği

```

unsigned short current_duty; /*PWM darbe sinyalinin aktif ve pasif
                             olma süresini tutacak olan değişken */
void main() {
    CMCON = 0X07; //Karşılaştırıcılar kapatılıyor.
    TRISA = 0; // A portu çıkış
    TRISB = 0; // B portu çıkış
    PORTA=0; // A portu temizleniyor
    PWM1_Init(1000); //1KHz'lik PWM sinyali
    current_duty = 16; //Darbe sinyalinin aktiflik oranı (16/255)*100
    PWM1_Start(); //PWM sinyali B portunun B3 pininden uygulanıyor
    PWM1_Set_Duty(current_duty);/*PWM sinyalinin aktiflik oranı
    kuruluyor */
    while(1){ //parlaklık kontrolü sürekli olarak yapılıyor
        do{ // İlk önce aktiflik oranı yükseltilerek kontrol edilen
            current_duty++; //lambanın parlaklığı yükseltiliyor
            delay_ms(100);
        }while(current_duty<256);
        do{ // ardından ikinci döngüyle kontrol edilen
            current_duty--; // lambanın parlaklığı azaltılıyor
            delay_ms(100);
        }while(current_duty>0);
    }
}

```

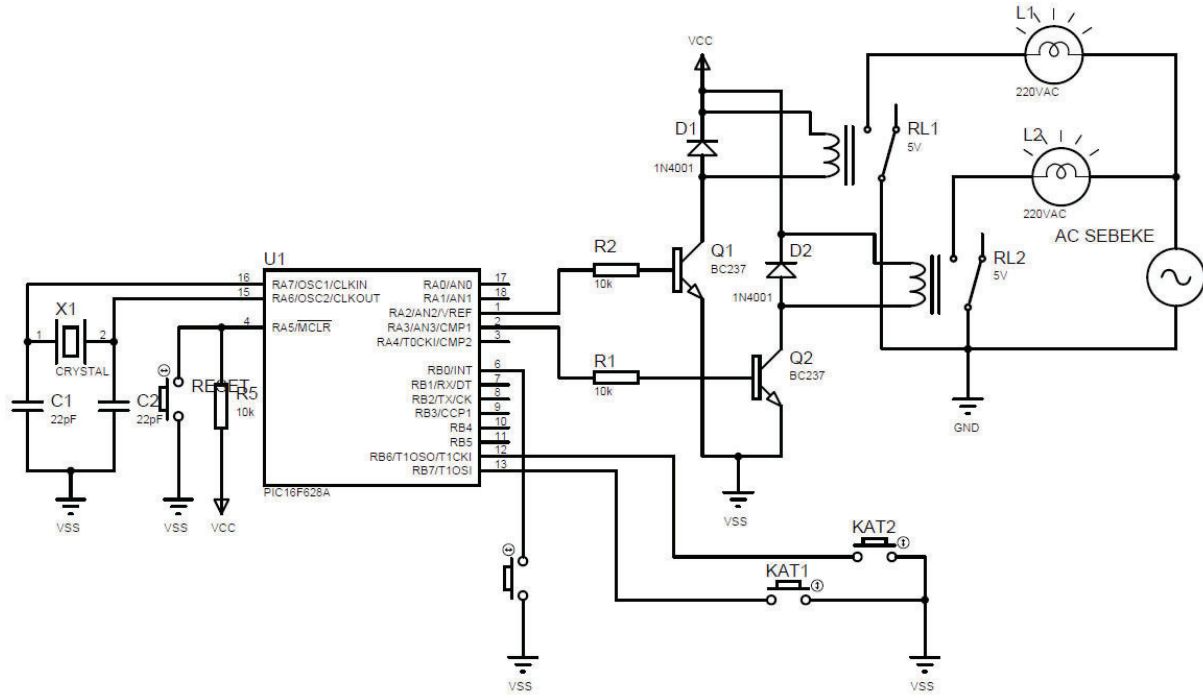
DİKKAT: Yüksek akım kontrolü yapılan uygulamalarda son derece dikkatli olunmalıdır. Elektrik-elektronik konusunda tecrübe eksikliği olanların bu uygulamaları yapmadan önce güvenilir kaynaklardan yeterli bilgiye sahip olmaları gerekmektedir.

ÇOK ÖNEMLİ: PROTEUS-ISIS® ortamında yapılan simülasyonlarda varsayılan olarak tek şase (ground) kullanılır. AC hattın ve DC besleme hattının şasesi ayrılmalıdır. Bilindiği üzere AC beslemeli sistemlerde giriş yönü farkı yoktur ve DC hattınızın şasesine Faz (L) hattının denk gelecek olması ciddi yaralanmalara ve hasarlara yol açabilir. Baskı devresi çıkartılması düşünülen uygulamalarda hatların birbirinden ayrılarak yol bağlantıları oluşturulmalıdır. Bu işlem için:

- Öncelikle her hattın şasesi farklı isimlendirilir. Şekil C.3'te DC hattın şasesi 'VSS', AC hattın şasesi 'GND' olarak isimlendirilmiştir.
- Bu yapıldıktan sonra PROTEUS-ISIS® 8 sürümünde Design→Configure Power Rails menüsünden VSS adında yeni bir POWER hattı eklenmelidir.
- PROTEUS'un varsayılan güç bağlantıları (Use default power rail connections?) seçeneği seçiliyse kaldırılmalıdır.
- "Power Supplies" seçeneğinin "Names" alanında GND seçiliyken, sağ tarafta "Nets connected to GND" yazan alanda yalnızca GND olmalıdır.
- Eğer bu alanda "VSS" hattı da gözükyorsa üzerine tıklanmalı ve "<-Remove" butonuyla listeden kaldırılmalıdır.
- Ardından ikinci adımda oluşturduğunuz VSS hattı aynı pencerenin "Power Supplies" alanından seçilmeli ve "Unconnected power nets:" alanında yer alan VSS değerine tıklanarak "Nets connected to GND" yazan alana "Add->" butonuyla eklenmelidir.

Mikrodenetleyicilerin endüstriyel ya da pratik uygulamalarından biride röle kumandasıdır. Yaygın olarak mikrodenetleyicilerle röle kumandasında elektronik anahtarlama elemanı olan transistör tercih edilmektedir.

Şekil 4.4'te uygulaması yapılan devrede LED elemanlarının yerine daha yüksek akım çeken ve 220VAC ile çalışan aydınlatma elemanları takılabilir. Böylece devre günlük yaşamda kullanılabilecek bir yapıya kavuşturulmuş olur. Bunun için DC 5V tetiklemeyle çalışan ve 7-8 Amper akıma kadar yük kontrol edebilen rölelerden faydalanabiliriz. Şekil C.4'te böyle bir uygulama devresi gösterilmiştir.

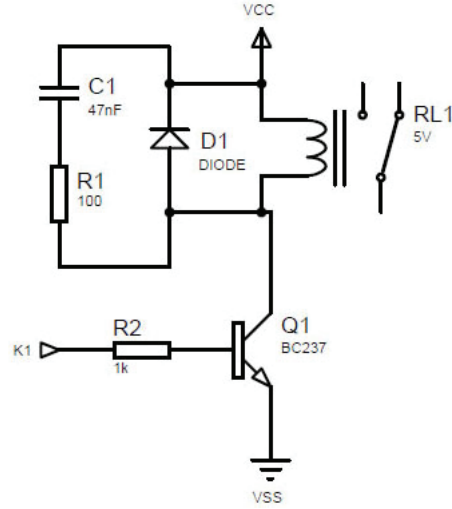


Şekil C.4 Mikrodenetleyici ile röle üzerinden AC şebekeye bağlı yükün kontrol edilmesi

Transistörler 1 mA'in altında akımlarla tetiklenebilen devre elemanlarıdır. Baz ayağı ile mikrodenetleyicinin portu arasına bağlanacak 10K'lık bir direnç ile tetikleme anında 0,4-0,5 mA dolaylarında bir akım elde edilir (VCC'nin 5V olması durumunda). Yeterli tetikleme akımı oluştuğunda transistorün kolektör-emetör arası kısa devre gibi davranır. VCC ve kolektör arasına 5V'luk bir rölenin besleme sargısı bağlandığında röle kontakları NC (Normal Close – Normalde Kapalı) ucundan NA (Normal Open – Normalde Açık) ucuna geçer. Bobin sargısının bağlandığı hatta direnç bağlanmasına gerek yoktur. Bobin iç direnci 200-300 ohm kadardır ve tetikleme sırasında transistor kolektöründen 20 mA dolaylarında bir akım geçer. Bu değer transistorün en yüksek yük akımı sınırının çok altındadır.

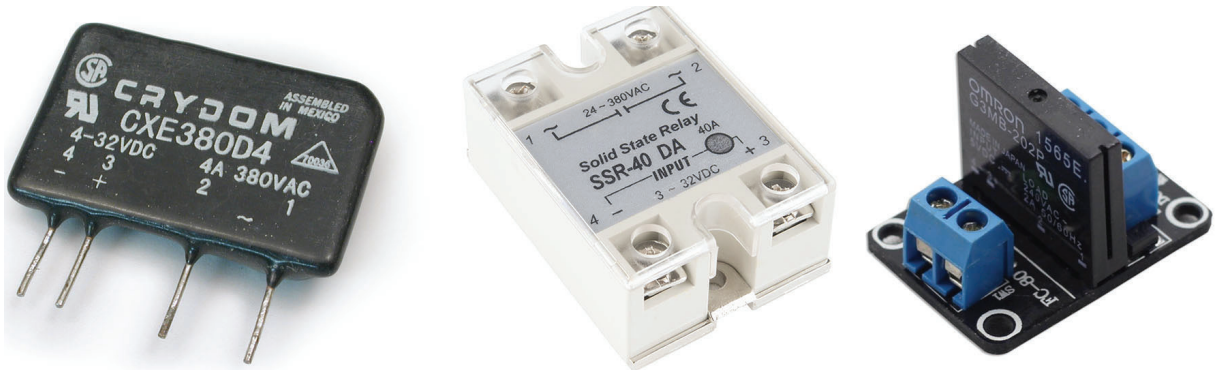
Transistorün röleyi tetiklemesinde VCC beslemesi ile transistorün kolektörü arasında (rölenin sargılarına paralel) ters diyot bağlanır. DC akımda rölenin bobin görevi gören sargısı hemen akım geçirmez. Bobin sargılarında zıt EMK'dan kaynaklı ters polarizeli voltaj yüklemesi olur ve bu ilk anda doğrudan transistor kolektöründe belirir. Bu durum transistorün aşırı akım geçirmesine ve arızalanmasına neden olur. Bu nedenle yalnızca tek yönde akım ileten diyot devre elemanının kullanılması şiddetle tavsiye edilir. Rölenin NA ucuna yükün bir ayağı bağlanır ve röle tetiklendiğinde enerjilenmiş olur. Rölelerin C (Common-Ortak) ayakları AC beslemenin diğer girişine ortak bağlanmıştır. Bu devrede de DC ve AC hattın şaseleri birbirinden ayrılmıştır.

AC yüklerin röle üzerinden mikrodenetleyici ile sürülmesi sırasında hatta oluşan manyetik parazit MCU'nun resetlenmesine neden olabilir. Bu nedenle rölenin bobin sargısı uçları arasında "stubber-frenleme" denilen ve kondansatör-direnç ikilisinden oluşan bir ek yapıdır. Frenleme devresinin uygulama örneği Şekil C.5'te gösterilmiştir.



Şekil C.5 47nF ve 100Ω kullanılarak yapılan bir frenleme devresi uygulaması

Resimde K1 girişi MCU'nun dijital I/O pininden gelen kontrol bağlantısıdır. Röle gibi selenoid yüklerin sağlıklı şekilde tetiklenmesinde kullanılan ve kondansatör-direnç ikilisi kullanılarak yapılan frenleme devrelerinin tasarımı kitap kapsamı dışındadır. Ancak, piyasada yaygın olarak kullanılan rölelerin selenoid sargıları için kullanılacak uygun frenleme devresi değerleri Şekil C.5'te gösterildiği gibi yeterli olacaktır. Frenleme devresiyle birlikte daha dengeli çalışan bu rölelerden farklı olarak "solidstate-katıhal" rölesi olarak adlandırılan röleler bulunmaktadır. Şekil C.6'da gösterilen benzeri solidstate rölelerden yararlanarak daha kararlı çalışan anahtarlama uygulamaları yapabilirsiniz.

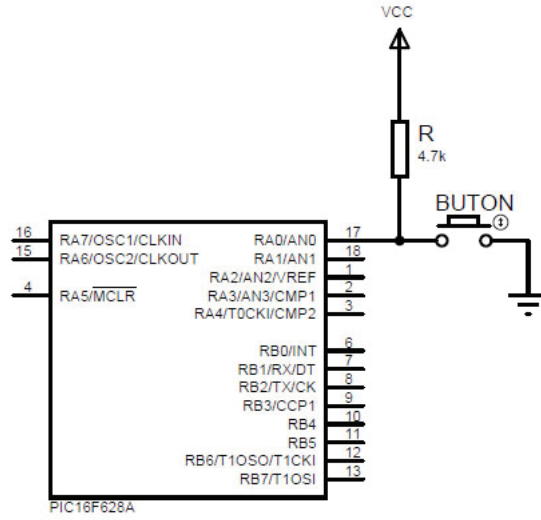


Şekil C.6 Çeşitli tipte solidstate röleler

C-3 Mikrodenetleyicilerde Butonların Kullanımı

Mikrodenetleyicilerde buton bağlantısı iki yöntemle gerçekleştirilir:

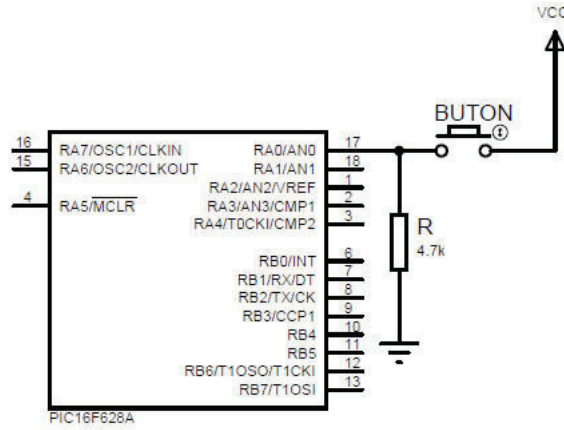
Birinci yöntemde butona basıldığında giriş Lojik-0 (Low-Düşük) olur. Bu uygulamada butonun bir ucu genelde direnç üzerinden VCC'ye (kaynak), diğer ucu ise şaseye bağlıdır (Bkz. Şekil C.7). Direnç değeri genellikle 4.7KΩ-10KΩ arasında seçilir. Bu dirence pull-up direnci denir.



Şekil C.7 Pull-up buton bağlantısı

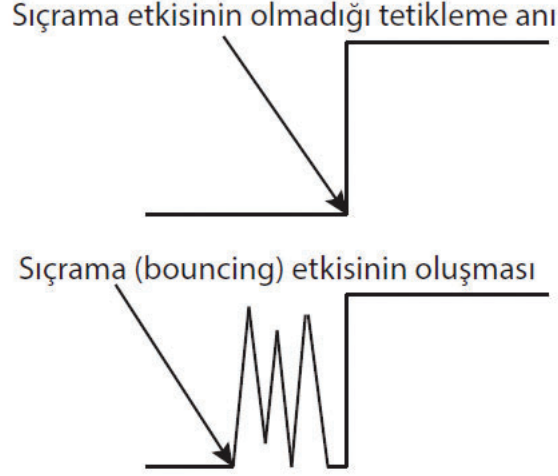
Pull-up bağlantılı buton ile devre kontrolünde girişin Lojik-0 olma durumu izlenir.

Diğer yöntemde ise butona basıldığında giriş Lojik-1 (High-Yüksek) olur. Bu uygulamada butonun bir ucu direnç üzerinden şaseye, diğer ucu ise VCC'ye bağlanır (Bkz. Şekil C.8). Bu bağlantı yöntemine pull-down denir. Pull-down bağlantılı buton ile devre kontrolünde girişin Lojik-1 olma durumu izlenir.



Şekil C.8 Pull-down buton bağlantısı

Tüm mikrodeneleyicilerde buton kullanımı sırasında ilgili port girişinde sıçrama etkisi (bouncing) dediğimiz istenmeyen bir olay meydana gelir. Bu olay buton, anahtar (switch) ve sınır anahtarı gibi mekanik bileşenlerin fiziki özelliğinden kaynaklanan bir durumdur. Şekilde C.9'da sıçrama etkisinin olmadığı ideal durum ve sıçrama etkisinin yaşandığı tetikleme anı resmedilmiştir.



Şekil C.9 Mekanik tetikleme bileşenlerinde sıçrama etkisi

Bu etkiyi ortadan kaldırmak için çeşitli yöntemler vardır. MikroC programınca kullanılan Buton ve Keypad kütüphanelerinde bu etki ortadan kaldırılacak şekilde kodlama yapılmıştır. Ancak kütüphane kullanmadan bu etkiyi pas geçmek istiyorsanız Tablo C.4'te gösterildiği gibi sık kullanılan ufak bir yazılım hilesi yapabilirsiniz.

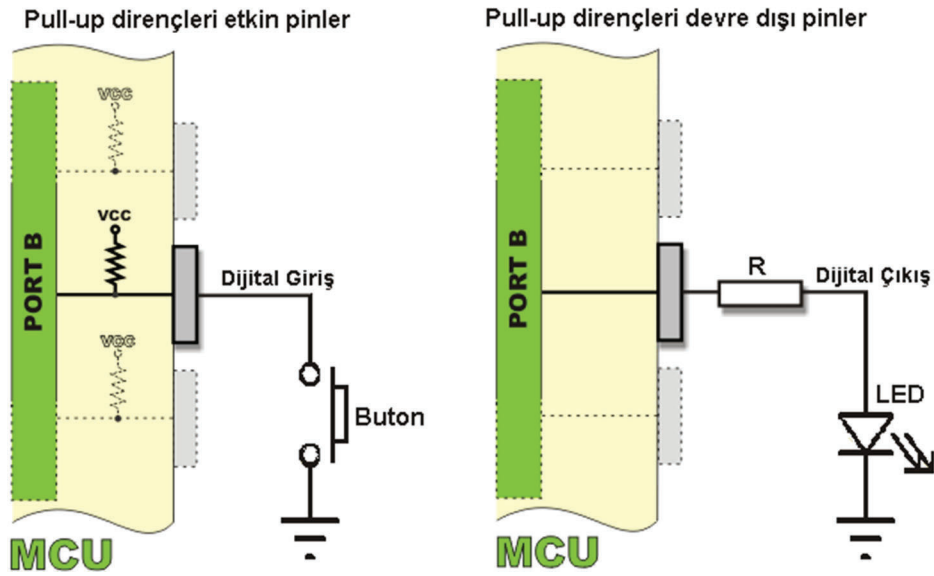
Tablo C.4 Sıçrama (bouncing) etkisini ortadan kaldıran kodlama örneği

```
if(PORTB.F0 == 0) //PORTB.0'a bağlı butona basıldı mı kontrol ediliyor
{
    Delay_ms(100); //100ms kadar bekle
    if(PORTB.F0 == 0) //butona hala basılı mı kontrol et
    {
        /* Butona gerçekten basılmış olduğuna karar verilir. Butona
           basıldıktan sonraki işlemler bu blokta gerçekleştirilir. */
    }
}
```

Bir butona ideal basma ya da mekanik bir anahtar ideal olarak tam kapalı hale getirme süresinin minimum 100 ms kadar olduğu ölçülmüştür. Dolayısıyla hazırlanan kodlamada bu ilkeden yola çıkılarak geçici bir bekleme süresi oluşturulmuş ve bu sürenin sonunda hala mevcut durumun korunup korunmadığına bakılmıştır. Yapay gecikmenin kullanılmasının istenmediği durumlarda Timer kesmesi kullanılarak da benzer işlem gerçekleştirilebilir.

C-4 Mikrodenetleyicilerde Dâhili Pull-Up Dirençleri

PIC mikrodenetleyicilerin genellikle B portları dâhili pull-up direnci (Weak Pull-ups) içerir. Bu özellik harici bir direnç kullanmayı gerektirmeden girişlerin doğrudan bir buton ya da algılayıcı kontrolü için kullanılacak bir Lojik kapı devresine bağlanmasını sağlar. Bu dâhili pull-up dirençleri gerçek bir direnç yerine MOSFET kullanılarak tasarlanmış ve direnç değeri 10'larca K Ω (Ör: 50 K Ω) olabilen devre elemanlarıdır. Giriş tetiklendiğinde mikroamperler seviyesinde (yaklaşık 200 μ A) akım çekerler. Genellikle SRAM'de yer alan kütüklerden birinin yazılımsal olarak programlanması suretiyle aktif ya da pasif kılınabilirler. Pull-up dirençlerinin bağlı olduğu port çıkış yapıldığında diğer portlar gibi çalışır ve pull-up dirençleri devre dışı bırakılır. Şekil C.10'da dâhili pull-up dirençlerinin portun giriş ya da çıkış yapılması durumundaki etkisi gösterilmiştir.



Şekil C.10 Dâhili pull-up dirençlerinin etkin ve pasif kılınması

Şekilden görüldüğü üzere dâhili pull-up dirençleri port giriş yapıldığında anlam kazanmaktadır. Bu özellik sayesinde ilgili port (PORTB) hızlı şekilde giriş değişimlerine tepki gösterebilir. Aşağıda bazı PIC modelleri için dâhili pull-up dirençlerini etkin kılmayı sağlayan ilgili kütük ve bitleri gösterilmiştir:

PIC16F628A için Tercih Kütüğünün (OPTION REGISTER – 80h, 81h) 7. biti (\overline{RPBU}) kullanılır. Bit değeri 1 yapılırsa PORTB dâhili pull-up dirençleri devre dışı bırakılır. B portu çıkış yapıldığında ya da Power-on Reset durumunda otomatik olarak bu dirençler devre dışı bırakılır. Bit değeri 0 yapılırsa, pull-up dirençleri tek tek port 'latch' değerleriyle etkinleştirilir.

PIC18F2450/4450 için Kesme Kontrol Kütüğü-2 (Interrupt Control Register 2 – FF1h) 7. biti (\overline{RPBU}) kullanılır. Bit değeri 1 yapılırsa PORTB dâhili pull-up dirençleri devre dışı bırakılır. B portu çıkış yapıldığında ya da Power-on Reset durumunda otomatik olarak bu dirençler devre dışı bırakılır. Bit değeri 0 yapılırsa, pull-up dirençleri tek tek port 'latch' değerleriyle etkinleştirilir.

PIC16F882/883/884/886/887 için Tercih Kütüğünün (OPTION REGISTER – 81h, 181h) 7. biti (\overline{RPBU}) kullanılır. Bit değeri 1 yapılırsa PORTB dâhili pull-up dirençleri devre dışı bırakılır. B portu çıkış yapıldığında ya da Power-on Reset durumunda otomatik olarak bu dirençler devre dışı bırakılır. Bit değeri 0 yapılırsa, pull-up dirençleri tek tek port 'latch' değerleriyle etkinleştirilir. Bu mikrodenetleyicide farklı olarak B portu pinlerini tek tek kontrol etmeyi sağlayan bir kütük daha vardır. Zayıf pull-up PORTB kütüğü olan WPUB ile B portunun her bir pull-up girişi birbirlerinden bağımsız olarak programlanabilir.

WPUB (WEAK PULL-UP PORTB REGISTER – 95h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB5	WPUB4	WPUB3
bit 7				bit 0			

WPUB<7:0> : Weak Pull-up Kütüğü bitleri

1 = Pull-up etkin

0 = Pull-up devre dışı

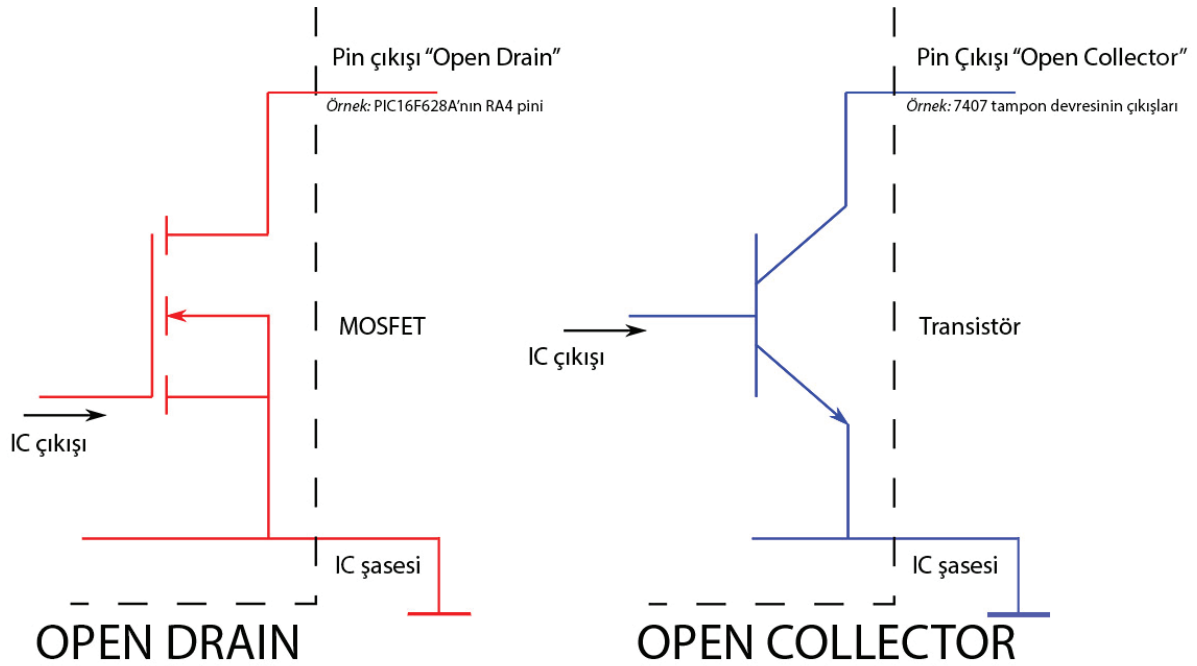
NOT 1: Bu kütüğün kullanılabilmesi için Tercih Kütüğü (OPTION) Evrensel $\overline{RPB\bar{U}}$ biti temizlenmelidir.

2: İlgili port pini çıkış yapılırsa weak pull-up devre dışı kalır.

C-5 Mikrodenetleyicilerde Open Drain ve Open Collector Çıkışlar

Özellikle sayısal entegre devrelerde (mikrodenetleyiciler, kapı entegreleri, dijital sinyal entegreleri, vb.) sıkça karşımıza çıkan bir deyimdir. Açık akar (Open Drain) özellikli IC pinleri OD, açık kollektör (Open Collector) özellikli IC pinleri OC olarak gösterilirler. "Drain", FET transistörlerde, "Collector" ise BJT transistörlerde olan bir bacak ismidir.

Mikrodenetleyicilerin I/O hatlarında yaygın olarak bir FET türevi olan MOSFET kullanıldığından, "OD" pinleri mikrodenetleyicilerde sıklıkla karşımıza çıkar. 74 serisi sayısal entegre devre bileşenlerinin I/O hatlarında ise yaygın olarak BJT kullanıldığından, "OC" özellikli pinler bu IC bileşenlerinde karşımıza çıkar. Şekil C.11'de örnek bir mikrodenetleyicinin OD özellikli pini, ve 74 serisi bir entegrenin de OC özellikli pini gösterilmiştir.



Şekil C.11 Sayısal entegre devre bileşenlerinde Open Drain ve Open Collector çıkış pinleri

OD ve OC özellikli pinler için karşımıza çıkacak bir yabancı terimde "current sink" deyimidir. Bu deyim akımın içeri doğru aktığını ifade eder (Bkz. Şekil C.1-b). Kısacası OD ve OC özellikli pinlerde akım içeri doğru akmaktadır. Şekilden de görüleceği üzere MOSFET'in drain ayağı ve BJT'nin kolektör ayağı açıktadır ve doğrudan entegre paketinin dışına bağlantı pini olarak çıkmaktadır. Dolayısıyla OD ve OC özellikli pinlerin sayısal I/O olarak kullanılması istendiğinde pinlere harici pull-up direnci bağlanmalıdır. Direnç bağlantısının haricinde Şekil 5.7'de gösterilen uygulamada olduğu gibi NPN ve PNP transistör ikilisi kullanılarak yapılan push-pull bağlantı tekniği de kullanılabilir. Mikrodenetleyicilerin OD pinlerini

ıkıř olarak kullanmak istediđinizde push-pull bađlantılı devre dzenlemesi uygun lojik seviyenin sađlanması noktasında daha dođru bir yaklařım olacaktır.

EK-D

D1-16F628A MİKRODENETLEYİCİSİNİN TEKNİK ÖZELLİKLERİ

PIC16F628A, 18-pinli orta ölçekli (Mid-Range) 8 bitlik Flash ROM özellikli bir Microchip® mikrodenetleyicisidir. 627A ve 648A serisiyle aynı sınıftandır. Teknik özellikleri genel olarak kısaca aşağıdaki gibidir:

Yüksek Performans RISC mimarili CMOS CPU:

- Çalışma frekansı 0 Hz (DC) – 20 MHz arasındadır.
- Kesme kabiliyetlidir.
- 8-seviye derinliğinde donanım yığınınına sahiptir.
- Doğrudan, dolaylı ve bağıl adresleme modları mevcuttur.
- 35 adet tek kelimelik komut kümesine sahiptir.
 - Dallonma komutları hariç tüm komutlar tek çevrim süresinde işletilir.

Özel Mikrodenetleyici Özellikleri:

- Dâhili ve harici osilatör seçenekleri vardır.
- Dâhili 4 MHz osilatör frekansı hassasiyeti $\pm 1\%$ fabrika ayarlıdır.
- 37 kHz'de dâhili düşük güç osilatörüne sahiptir.
- Harici osilatör için kristal ve rezonatör seçenekleri mevcuttur.
- SLEEP (Uyku) modunda güç tasarrufu sağlar.
- PORTB için programlanabilir zayıf pull-up girişleri vardır.
- Çoktan seçmeli \overline{MCLR} (Master Clear) girişi: Ana donanım resetlemesi pini aynı zamanda dijital I/O olarak kullanılabilir.
- Güvenilir çalışma için bağımsız osilatörlü bekçi zamanlayıcısı (WDT – WatchDog Timer)
- Düşük güç programlama
- Devre üzerinde seri programlama (ICSP™) ile programlanabilir.
- Programlanabilir kod koruma özelliği ile yazılan programlar kopyalamaya karşı korunabilir.
- Voltaj düşmesinde resetlenme özelliği (BOR-Brown-out Reset): Besleme geriliminde azalma ya da güç kaybı yaşandığında MCU yanlış ve kararsız çalışmaya karşı kendini resetler.
- İlk besleme ile resetlenme özelliği (POR-Power-on Reset): MCU'ya ilk enerji verildiğinde başlangıç adresine gidilir ve program kodu yeni baştan işletilir.
- İlk besleme gecikmesi zamanlayıcısı (PWRT-Power-up Timer): MCU'ya ilk enerji verildiğinde besleme dengesi sağlanana kadar 72 msn'lik bir gecikme sağlayan zamanlayıcıdır.
- Osilatör başlangıç zamanlayıcısı (OST-Oscillator Start-up Timer): MCU'ya ilk enerji verildiğinde kristal osilatörü dengesi sağlanana kadar MCU'yu reset modunda tutan zamanlayıcıdır.
- Geniş çalışma voltajı aralığı (2.0 – 5.5V) vardır.
- Endüstriyel ve geniş çalışma sıcaklığı aralığına (-40 – 125 °C) sahiptir.
- Yüksek dayanıklı FLASH/EEPROM hücresi vardır.

- FLASH için 100.000 yazma dayanıklılığına
- EEPROM için 1.000.000 yazma dayanıklılığına ve
- 100 yıl veri saklama süresine sahiptir.

Düşük Güç Özellikleri:

- Bekleme (Standby) akımı: 2.0V'ta **100nA**
- Çalışma akımı: 32 kHz 2.0V'ta **12uA**, 1 MHz 2.0V'ta **120uA**
- Bekçi zamanlayıcısı (WDT) akımı: 2.0V'ta **1uA**
- Timer1 osilatör akımı: 32 kHz 2.0V'ta **1,2uA**
- Çift hızlı dâhili osilatör:
 - 4 MHz ve 37 kHz çalışma frekansları arasında tercih
 - 3.0V'ta 4 us içinde uyku (SLEEP) modundan çıkabilme

Çevresel Aygıt Özellikleri:

- Bağımsız yön kontrollü 16 I/O pini
- Doğrudan LED sürmek için yüksek sink/source akımı desteği vardır.
- Aşağıdaki özelliklere sahip analog karşılaştırıcı modülü:
 - İki analog karşılaştırıcı
 - Programlanabilir on-chip voltaj referansı modülü (V_{REF})
 - Seçilebilir dâhili ya da harici referans
 - Karşılaştırıcı çıkışları harici olarak erişilebilir
- Timer0: 8-bit önölçekleyicili 8-bitlik zamanlayıcı/sayıcı
- Timer1: Harici kristal/saat darbesi özellikli 16-bitlik zamanlayıcı/sayıcı
- Timer2: 8-bit periyot kaydedicili, önölçekleyicili ve son ölçekleyicili 8-bitlik zamanlayıcı/sayıcı
- Yakalama, Karşılaştırma, PWM modülü
 - 16-bit Yakalama/Karşılaştırma
 - 10-bit PWM
- Adreslenebilir USART/SPI

Gerçek zamanlı uygulamaları gerçekleştirmeyi sağlayan özel amaçlı devreler mikrodenetleyicileri diğer işlemcilerden ayıran en önemli özelliklerdendir. PIC16F627A/628A/648A ailesi sistem güvenilirliğini artırmayı ve harici devre bileşeni kullanımını ortadan kaldırarak maliyetleri düşürmeyi amaçlayan böylesi bir dizi özelliklerle donatılmıştır. Bu özellikleri şu şekilde sıralayabiliriz:

- 1- OSC seçimi
- 2- RESET
- 3- İlk güç uygulandığında MCU'nun resetlenmesi (Power-on Reset – POR)
- 4- İlk besleme gecikmesi zamanlayıcısı (Power-up Timer – PWRT)
- 5- Osilatör açılış zamanlayıcısı (Oscillator Start-up Timer – OST)
- 6- Voltaj düşmesi resetlemesi (Brown-out Reset – BOR)
- 7- Kesmeler (Interrupts)
- 8- Vardiya zamanlayıcısı (Watchdog Timer – WDT)
- 9- Uyku modu (SLEEP)
- 10- Kod koruma (Code protection)

11- ID lokasyonları

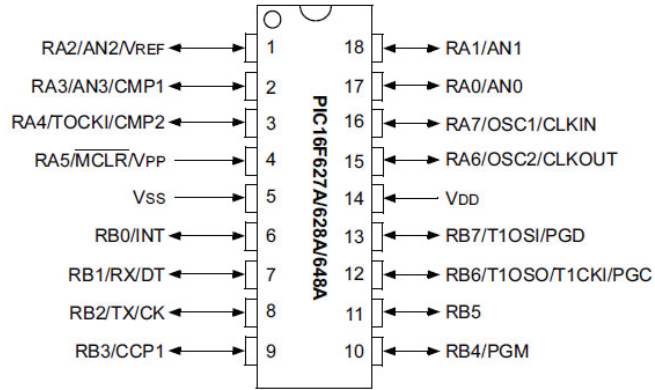
12- Devre üzerindeki programlanabilme (In-Circuit Programming – ICSP)

PIC16F627A/628A/648A konfigürasyon bitlerince kontrol edilebilen vardiya zamanlayıcısına sahiptir. İlave güvenlik için kendi RC osilatörünü kapatır. İlk besleme verilmesi anında gerekli gecikmeyi sağlayan iki zamanlayıcısı daha vardır. Biri, kristal osilatör dengeli çalışmaya başlayana kadar çipi RESET modunda tutan OST zamanlayıcısı, diğeri ise besleme gerilimi dengeye oturana kadar çipi RESET modunda tutan ve yalnızca ilk açılış sırasında sabit 72msn'lik gecikme sağlayan PWRT zamanlayıcısıdır. Ayrıca voltaj düşmesi meydana geldiğinde cihazı resetleyen bir güvenlik devresi de vardır. Çipin sahip olduğu bu üç özellik sayesinde, çoğu uygulamada harici bir RESET devresine ihtiyaç yoktur.

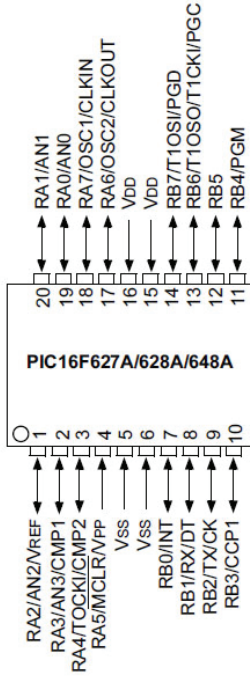
Uyku modu (SLEEP) çok düşük akım sunan bir düşük güç modudur. Kullanıcı mikrodenetleyiciyi, harici reset, vardiya zamanlayıcısı ya da bir kesme yoluyla uyku modundan çıkartabilir. Mikrodenetleyici için uygulamaya göre çeşitli osilatör seçenekleri de mevcuttur. RC osilatör seçeneği sistem maliyetini düşürürken LP kristal seçeneği güç tasarrufu sağlar. Konfigürasyon bitleri ayarlanarak farklı tercihler yapılabilir.

Şekil D.1'de PIC16F628A'nın farklı kılıf türleri gösterilmiştir. SSOP ve QFN paketleri yüzey temaslı bir paket türü olup, PDIP ise DIP soket türündedir.

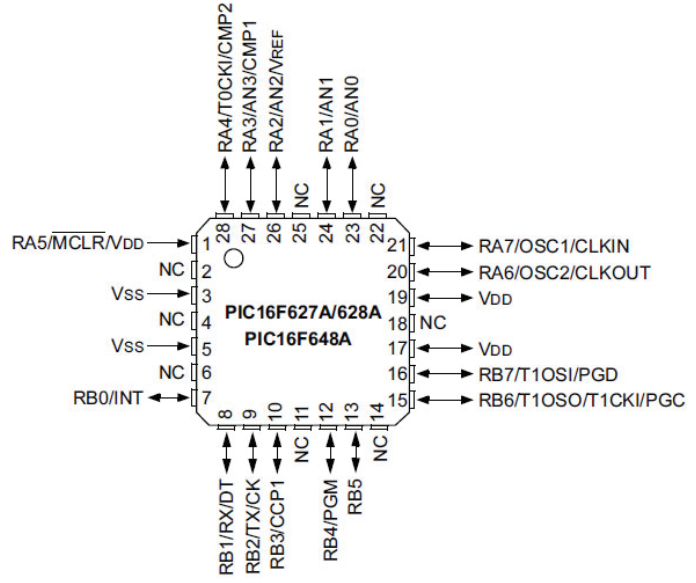
PDIP, SOIC



SSOP



28-Pin QFN



Şekil D.1 16F628A'nın farklı kılıf türleri

D2-PIC16F628A'NIN BACAK (PİN) AÇIKLAMALARI

Tablo D.1 PIC16F628A'nın bacak açıklamaları

İsim	İşlev	Giriş Türü	Çıkış Türü	Açıklama
RA0/AN0	RA0	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	AN0	AN	-	Analog karşılaştırıcı girişi
RA1/AN1	RA1	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	AN1	AN	-	Analog karşılaştırıcı girişi
RA2/AN2/V _{REF}	RA2	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	AN2	AN	-	Analog karşılaştırıcı girişi
	V _{REF}	-	AN	V _{REF} çıkışı
RA3/AN3/CMP1	RA3	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	AN3	AN	-	Analog karşılaştırıcı girişi
	CMP1	-	CMOS	Karşılaştırıcı 1 çıkışı

RA4/T0CKI/CMP2	RA4	ST	OD	Genel amaçlı çift yönlü I/O portu
	T0CKI	ST	-	Timer0 saat darbesi girişi
	CMP2	-	OD	Karşılaştırıcı 2 çıkışı
RA5/ $\overline{\text{MCLR}}$ /V _{PP}	RA5	ST	-	Giriş portu
	$\overline{\text{MCLR}}$	ST	-	Ana reset girişi. MCLR olarak ayarlandığında denetleyicinin RESET edilmesi için aktif 0 yapılır. Normal çalışma anında MCLR/VPP üzerindeki voltaj değeri VDD'yi aşmamalıdır.
	V _{PP}	-	-	Programlama voltajı girişi
RA6/OSC2/CLKOUT	RA6	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	OSC2	-	XTAL	Osilatör kristal çıkışı. Kristal osilatör modunda rezonatör ya da kristale bağlanır.
	CLKOUT	-	CMOS	RC/INTOSC modunda, OSC2 pini OSC1 frekansının 4'te 1'i frekansında çıkış olabilir.
RA7/OSC1/CLKIN	RA7	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	OSC1	XTAL	-	Osilatör kristal girişi
	CLKIN	ST	-	Harici saat darbesi girişi. RC kutuplama pini.
RB0/INT	RB0	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	INT	ST	-	Harici kesme
RB1/RX/DT	RB1	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	RX	ST	-	USART alma pini
	DT	ST	CMOS	Eşzamanlı I/O veri portu

RB2/TX/CK	RB2	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	TX	-	CMOS	USART gönderme pini
	CK	ST	CMOS	Eşzamanlı I/O saat darbesi portu
RB3/CP1	RB3	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	CP1	ST	CMOS	Yakalama/Karşılaştırma/PWM I/O portu
RB4/PGM	RB4	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	PGM	ST	-	Düşük voltaj programlama pini. Düşük voltaj programlaması aktif kılındığında pin değişimi kesmesi ve dâhili pull-up devre dışı bırakılır.
RB5	RB5	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
RB6/T1OSO/T1CKI/PGC	RB6	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	T1OSO	-	XTAL	Timer1 osilatör çıkışı
	T1CKI	ST	-	Timer1 saat darbesi girişi
	PGC	ST	-	ICSP programlama girişi
RB7/T1OSI/PGD	RB7	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	T1OSI	XTAL	-	Timer1 osilatör girişi

RB7/T1OSI/PGD	PGD	ST	CMOS	ICSP I/O verisi
VSS	VSS	Besleme	-	Lojik ve I/O pinleri için şase referansı
VDD	VDD	Besleme	-	Lojik ve I/O pinleri için pozitif besleme

Açıklama: O = Çıkış CMOS = CMOS çıkış P = Besleme
- = Kullanılmaz I = Giriş ST = Schmitt Triggerli giriş
TTL = TTL girişi OD = Open Drain Çıkış AN = Analog

D3-PIC16F628A MİKRODENETLEYİCİSİ İÇİN “mikroC PRO for PIC” PROJE EDITÖRÜ

PIC16F628A mikrodnetleyicisinin yapılandırma bitleri program belleğinin 2007h adresinde yer alır. Bu bölge kullanıcının kodlama yaptığı adres alanının dışındadır. Yapılandırma bitleri özel ayar belleği bölgesi (Special Configuration Memory Space – 2000h-3FFFh) içinde yer alır. Yapılandırma bitleri ilk fabrika ayarlarında kurulu olmadan gelir. Diğer bir ifadeyle mikrodnetleyicilerde yapılandırma bitleri kurulu değilse lojik-1’dir. Programlayıcı çalıştırdığınızda 14 bitlik bellek alanının tüm adreslerinde 3FFF görülmesinin nedeni bundandır. Bu bitleri kurmak için lojik-0 yapılması gerekir.

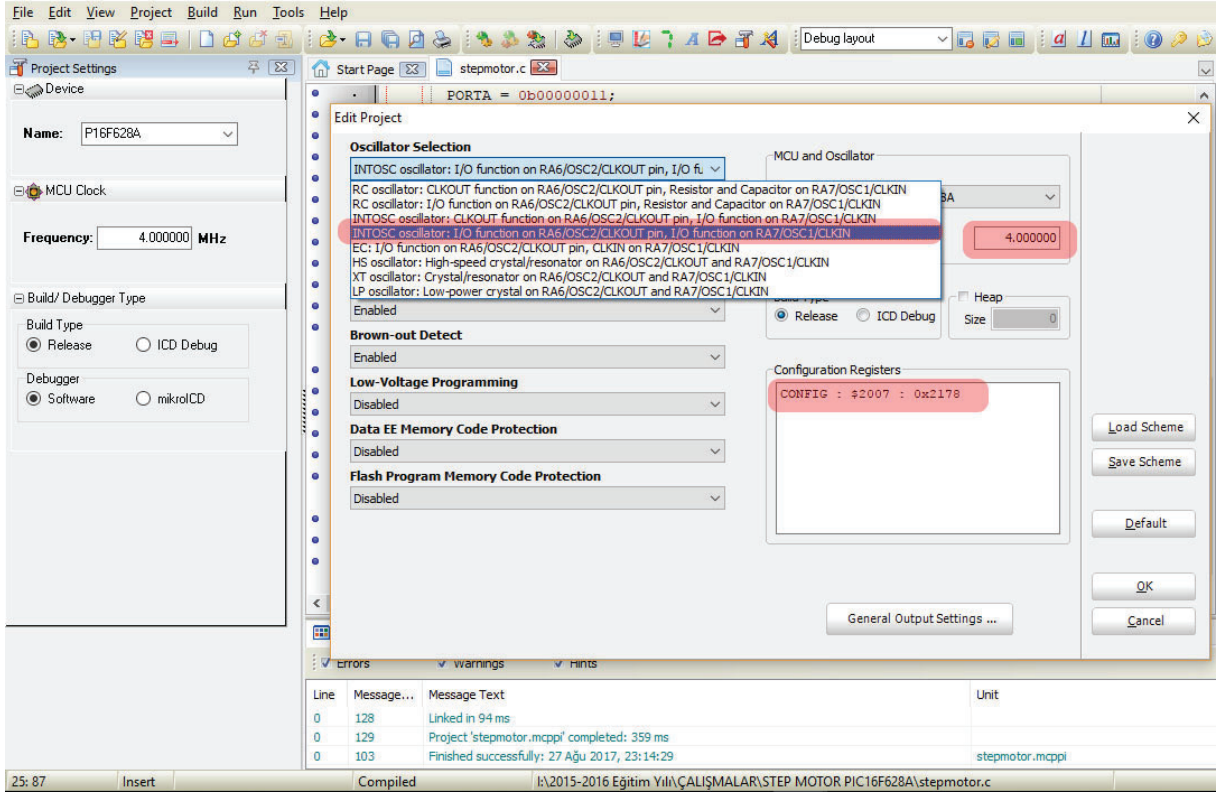
mikroC programı kullanıcıya yapılandırma bitlerini kullanıcı arayüzü üzerinden ayarlama olanağı sağlar. mikroC programında menü çubuğundan Project→Edit Project komutunu verdiğinizde çıkan pencerede mikrodnetleyicinin modeline bağlı olarak farklı başlıklar altından ayarlamalar yapabileceğiniz bir pencere karşınıza gelir. Şekil D.2’de 16F628A için osilatör ayarlarının yapılması gösterilmiştir.

Örnek pencere ekranında osilatör tercihi (Oscillator Selection) başlıklı kısımdan, mikrodnetleyicinin RA6 ve RA7 pinlerinin I/O portu olarak kullanılabilmesini sağlayan “INTOSC oscillator” tercihi seçilmiştir. Böylece kristal ve kondansatörler kullanılarak yapılan harici osilatör ihtiyacı ortadan kalkmış ve bu portlar boşa çıkmıştır.

Bu tercihin seçilmesi sonrası “Configuration Registers” başlıklı yapılandırma kütüğü alanında,

CONFIG: \$2007: 0x2178 hex değeri oluşur.

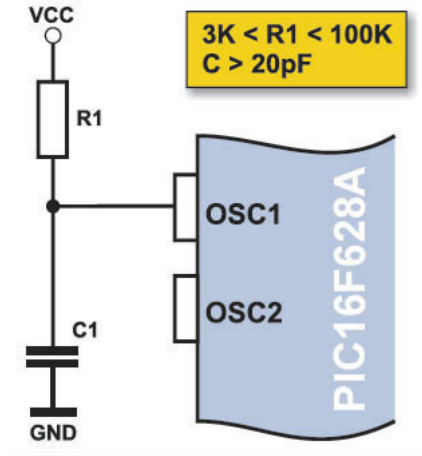
Farklı bir değer seçilmesi durumunda yapılandırma sözcüğünün (CONFIGURATION WORD) onaltılı değerinin değiştiği görülecektir. Devre tasarımınıza uygun ayarlamayı yaptıktan sonra OK butonuyla pencereyi kapatın. Ardından değişikliklerin derlenmiş hex dosyasında yer alması için, Ctrl+F9 tuşlarıyla ya da menü başlığından Build komutuyla derleme işlemini yeniden yapmanız gerekir. Dâhili osilatör tercihinin seçimi durumunda yalnızca 4 MHz ve 37kHz’lik iki frekans seçimi yapılabileceği unutulmamalıdır. Bu durumda “MCU Clock Frequency [MHz]” alanına uygun değer yazılmalıdır. Ancak mikroC proje editöründe osilatör frekansı olarak 1 MHz’in altında bir değer girişi yapılamamaktadır. MCU’yu 37kHz frekansında çalıştırmak istiyorsanız program kodunuzda “PCON.OSCF = 0;” kodunu yazmalısınız. Burada PCON kaydedicisinin dâhili osilatör frekansı tercihi OSCF pinini ayarlamış olursunuz.



Şekil D.2 16F628A'da Yapılandırma bitlerinin ayarlanması

16F628A için karşınıza gelecek pencerede yer alan ayar başlıkları şunlardır:

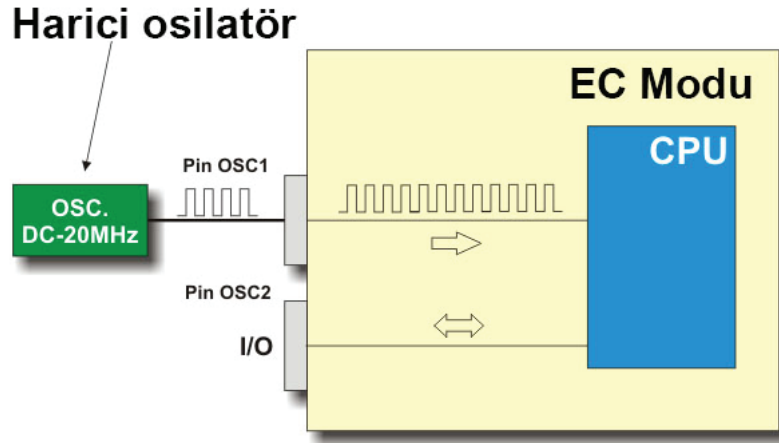
- **Osilatör tercihi (Oscillator Selection)**= 8 farklı osilatör tercihi yapılabilir.
 - **RC oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, Resistor and Capacitor on RA7/OSC1/CLKIN** = Osilatör frekansının kararlı çalışması önemli değilse, RA7 pinine 3K – 100K arasında bir direnç ve 20pF’den büyük bir kondansatör şeklindeki gibi bağlanır.



Şekil D.3 RC osilatör bağlantısı

OSC2 pininden üretilen frekansın 4'te 1'i çıkış olarak alınabilir.

- **RC oscillator: I/O function on RA6/OSC2/CLKOUT pin, Resistor and Capacitor on RA7/OSC1/CLKIN** = İlk osilatör tercihinde olduğu gibi RC bağlantısı yapılır. Ancak bu durumda RA6 pini I/O portu olarak kullanılabilir. Frekans çıkışı alınmaz.
- **INTOSC oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN** = İki frekanslı dâhili osilatör tercihidir. RA7 pini I/O portu olarak kullanılabilir. RA6 pininden çalışma frekansının 4'te 1'i çıkış olarak alınır.
- **INTOSC oscillator: I/O function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN** = İki frekanslı dâhili osilatör tercihidir. RA6 ve RA7 pinlerinin her ikisi de I/O portu olarak kullanılabilir. Frekans çıkışı alınmaz.
- **EC: I/O function on RA6/OSC2/CLKOUT pin, CLKIN on RA7/OSC1/CLKIN** = Harici bir saat darbesi üreticinin osilatör kaynağı olarak kullanılması istendiğinde bu tercih edilir. RA6 pini I/O portu olarak kullanılabilir. Giriş RA7 pini üzerinden yapılır. Harici osilatör kaynağının frekans aralığının MCU'nun kılavuzunda belirtilen çalışma aralığında olması gerekir. Girişe uygulanabilecek en yüksek sinyal frekansı 20 MHz'dir. EC modunda MCU'ya güç verildiği anda gecikme olmaksızın çalışır. EC modu, MCU'nun diğer harici devre bileşenleriyle senkronize çalışması istenen durumlarda kullanılır.

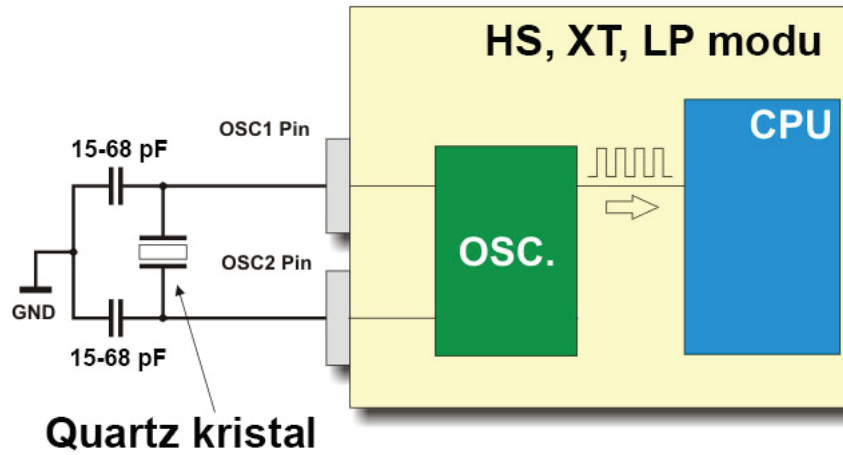


Şekil D.4 EC osilatör modu

- **HS oscillator: High speed crystal/resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN** = Yüksek frekanslı kristal/ rezonatör osilatör seçeneğidir. 4 MHz ve üzeri kristaller bu modda kullanılır. Akım tüketimi HS-XT-LP modları içinde en yüksek olanıdır.
- **XT oscillator: Crystal/resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN** = Orta ölçekli frekansta kristal/rezonatör osilatör seçeneğidir. 4 MHz'e kadar olan kristaller kullanılabilir.
- **LP oscillator: Low-power crystal on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN** = Düşük güçlü quartz kristal seçeneğidir. Bu modda yalnızca 32.768 kHz frekansındaki quartz kristaller kullanılabilir. Bu tür kristaller quartz saatlerin içinde bulunabilir. Akım tüketimi HS-XT-LP modları içinde en düşük olanıdır.

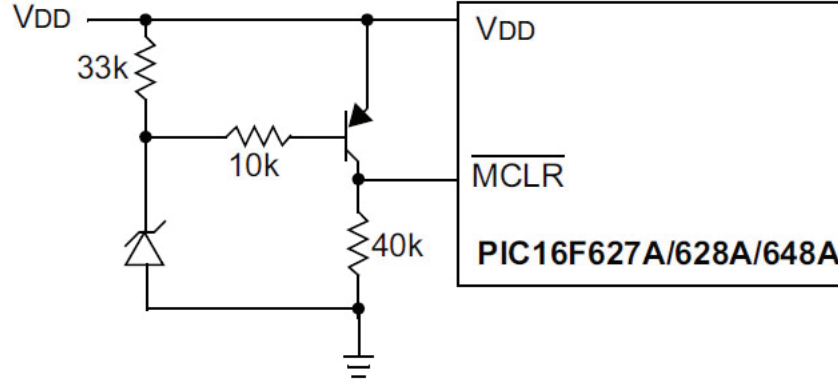
Tablo D.2 Kristal/rezonatör frekanslarına göre kondansatör değerleri

Mod	Frekans	C1, C2
LP	32 kHz	33 pF
	200 kHz	15 pF
XT	200 kHz	47-68 pF
	1 MHz	15 pF
	4 MHz	15 pF
HS	4 MHz	15 pF
	8 MHz	15-33 pF
	20 MHz	15-33 pF



Şekil D.5 HS, XT, LP osilatör modları

- **Watchdog Timer**= Vardiya zamanlayıcısıdır. Etkinleştirildiği (Enable) taktirde WDT reset aktif olur. Yazılımsal olarak sıfırlanabilir. MCU'nun kendi içindeki RC osilatörüyle çalışır. Çalışması harici osilatörden etkilenmez.
- **Power-up Timer**: İlk besleme gecikmesi zamanlayıcısıdır. 'Enable – lojik0' yapıldığında etkinleştirilir. 'Disable – lojik1' yapıldığında pasifleştirilir. Güç ilk verildiğinde nominal olarak 72 msn'lik bir gecikme sağlayarak MCU'yu voltaj dengesi sağlanması sırasında RESET modunda tutmayı amaçlar.
- **RA5/MCLR/VPP Pin Function**: Etkinleştirildiğinde normal donanımsal RESET girişi olarak kullanılır. 'Disable' ile pasifleştirildiğinde RA5 pini dijital giriş olarak kullanılabilir. \overline{MCLR} dâhili olarak V_{DD} 'ye bağlanır.
- **Brown-out Detect**: Bu özellik 'Enable' yapılırsa, MCU besleme voltajında düşme meydana geldiğinde MCU resetlenir. Özellikle mikrodenetleyicilerin yüksek akım çeken elektromekanik bileşenleri kontrol ettiği durumlarda yeterli akım sağlanmadığında MCU voltajında salınımlar oluşur. Bu durum MCU'nun resetlenmesine neden olur. Besleme hattının yalıtılmadığı yerlerde ya da elektromekanik elemanların DC beslemesinin aynı kaynak üzerinden yapıldığı yerlerde bu sorunla karşılaşabilirsiniz. Şekil D.6'da harici bir Brown-out devresinin \overline{MCLR} girişine bağlantısı gösterilmiştir. Dâhili BOR kesmesi 'Disable' yapıldığında böyle bir devre yardımıyla kendi voltaj kesmenizi yapabilirsiniz. MCU besleme gerilimi V_{DD} , zener gerilimi $V_Z + 0.7V$ 'un altına düştüğünde reset işlemi gerçekleşir.



Şekil D.6 Harici Brown-out devresi

- **Low-Voltage Programming**= Normal şartlarda 'Disable –lojik0' durumundadır. 'Enable – lojik1' yapıldığında RB4/PGM girişi PGM (ProGraMming) olarak kullanılır ve bu girişten düşük voltajla programlama yapılabilir. Etkin değilken RB4 pini normal I/O portu olarak kullanılır. Bu durumda $\overline{\text{MCLR}}$ girişinden HV programlama yapılmalıdır.
- **Data EE Memory Code Protection**= Veri belleği kod koruma bitidir.
- **Flash Program Memory Code Protection**= Program belleği koruma bitidir.

Şekil D.7'de PIC16F628A mikrodenetleyicisinin yapılandırma sözcüğü (Configuration Word) gösterilmiştir.

CP	—	—	—	—	CPD	LVP	BOREN	MCLRE	FOSC2	$\overline{\text{PWRTE}}$	WDTE	F0SC1	F0SC0
bit 13													bit 0

Şekil D.7 YAPILANDIRMA SÖZCÜĞÜ (CONFIGURATION WORD)

Yapılandırma sözcüğünü mikroC program editöründe yazılımsal olarak programlamak söz konusu değildir. Bu nedenle 14 bitlik bu kütüğün her bir bitinin ne olduğunu merak edenler ilgili MCU'nun veri kılavuzunu inceleyebilir. Kılavuzun 94. sayfasında yapılandırma sözcüğü açıklanmıştır.

EK-E

E1-16F887 MİKRODENETLEYİCİSİNİN TEKNİK ÖZELLİKLERİ

PIC16F887, 40-pinli 8 bitlik orta ölçekli (Mid-Range) Flash ROM özellikli ve nanoWatt teknolojili bir Microchip® mikrodenetleyicisidir. 16 serisi mikrodenetleyicilerin mimarisine sahip olmakla birlikte bu serinin en gelişmiş ve donanımlı modellerinden biridir. PIC16F882/883/884/886 serisiyle aynı sınıftandır. Teknik özellikleri genel olarak kısaca aşağıdaki gibidir:

Yüksek Performanslı RISC mimarili CMOS CPU:

- Çalışma frekansı 0 Hz (DC) – 20 MHz arasındadır.
- 200ns'ye kadar komut çevrimi vardır.
- Kesme kabiliyetlidir.
- 8-seviye derinliğinde donanım yığınınına sahiptir.
- Doğrudan, dolaylı ve bağıl adresleme modları mevcuttur.
- 35 adet tek kelimelik komut kümesine sahiptir.
 - Dalların komutları hariç tüm komutlar tek çevrim süresinde işlenir.

Özel Mikrodenetleyici Özellikleri:

- Hassas Dâhili Osilatör:
 - $\pm 1\%$ fabrika ayarlıdır.
 - Yazılımsal olarak seçilebilir frekans aralığı 8 MHz'den 31 kHz'e kadardır.
 - Yazılımsal olarak ayarlanabilir.
 - İki hızlı açılış modu vardır.
 - Kritik uygulamalar için kristal hata algılaması vardır.
 - Güç tasarrufu için çalışma sırasında saat modu anahtarlaması vardır.
- Güç Tasarrufu Uyku modu vardır.
- Geniş çalışma voltajı aralığına sahiptir (2,0V – 5,5V).
- Endüstriyel ve geniş çalışma sıcaklığı aralığına (-40 – 125 °C) sahiptir.
- İlk besleme ile resetlenme özelliği (POR) vardır.
- İlk besleme gecikmesi zamanlayıcısı (PWRT) ve Osilatör başlangıç zamanlayıcısı (OST) vardır.
- Yazılımsal kontrol seçeneği voltaj düşmesinde resetlenme özelliği (BOR) vardır.
- Gelişmiş düşük akımlı Bekçi Zamanlayıcısı (WDT): Çipte yer alan 31 kHz'lik osilatör ile çalışan ve zaman aşımı 1ms'den 268 saniyeye kadar yazılımsal olarak tam önölçekleyici ile seçilebilir bir zamanlayıcıdır.
- Çoktan seçmeli \overline{MCLR} (Master Clear) girişi: Ana donanım resetlemesi pini aynı zamanda dijital I/O olarak kullanılabilir.
- Programlanabilir kod koruması vardır.
- Yüksek dayanıklı FLASH/EEPROM hücresi vardır.
 - FLASH için 100.000 yazma dayanıklılığı
 - EEPROM için 1.000.000 yazma dayanıklılığı
 - Flash/EEPROM veri saklama: >40 yıl
- Çalışma sırasında program belleği Okuma/Yazma yapabilir.

- Devre içi hata ayıklayıcısı (debugger) vardır.

Düşük Güç Özellikleri:

- Bekleme (Standby) akımı: 2,0V'ta **50nA**
- Çalışma akımı: 32 kHz 2,0V'ta **11uA**, 4 MHz 2,0V'ta **220uA**
- Bekçi zamanlayıcısı (WDT) akımı: 2,0V'ta **1uA**

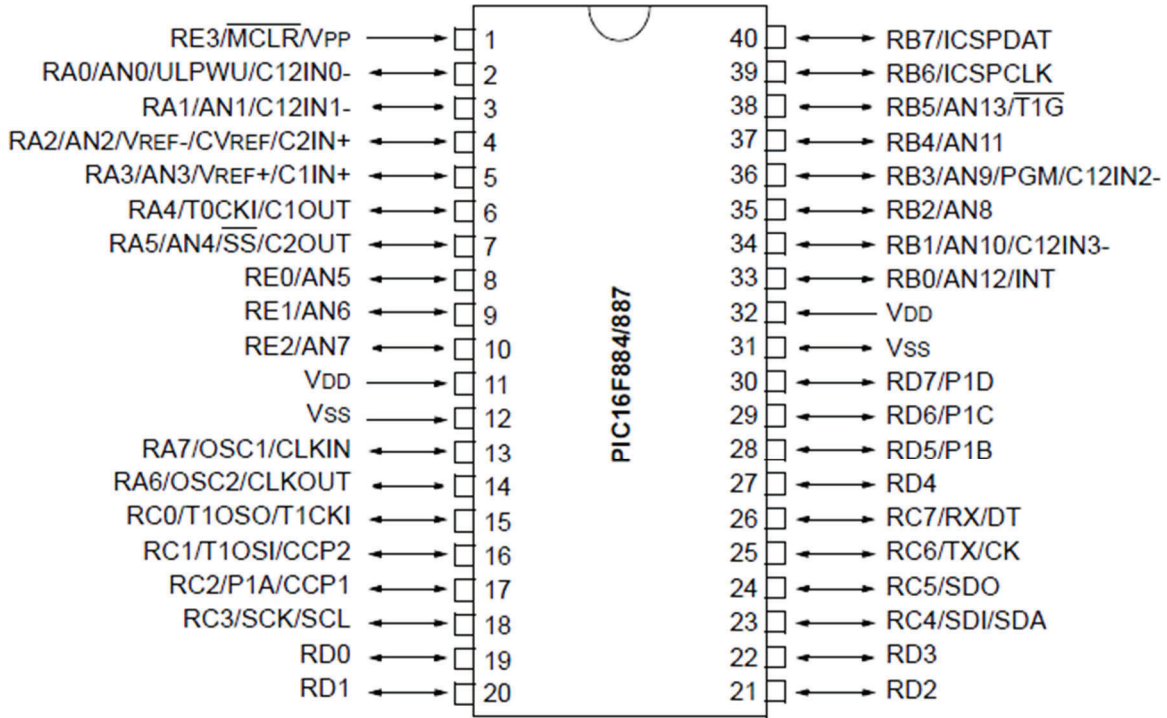
Çevresel Aygıt Özellikleri:

- Bağımsız yön kontrollü 35 I/O pini
 - Doğrudan LED sürmek için yüksek sink/source akımı desteği vardır.
 - Pinlerde değişim algılaması kesmesi vardır.
 - Bağımsız olarak programlanabilir zayıf pull-uplar vardır.
 - Ultra düşük güçlü uyandırma kesmesi (Ultra Low-Power Wake-up – ULPWU): Mikrodenetleyicinin RA0 pinine bağlanacak bir kapasitörün şarj ve deşarj edilmesi tekniğiyle, RA0 girişindeki voltaj seviyesinin VIL değerinin altına düşmesi durumu izlenerek düşük güçlü uyandırma kesmesi aktif kılınır ve bir sonraki komut işletilir.
- Analog Karşılaştırıcı modülü:
 - İki analog karşılaştırıcıdır.
 - Programlanabilir dâhili voltaj referansı (CV_{REF}) modülü (V_{DD} 'nin %'si).
 - Sabit voltaj referansı (0,6V)
 - Karşılaştırıcı girişleri ve çıkışları harici olarak erişilebilirdir.
 - SR (Set-Reset) Tutma (Latch) modu
 - Harici Timer1 kapısı (sayma etkinleştirmeli)
- A/D Dönüştürücü
 - 10-bit çözünürlüklü ve 14 kanallı
- Timer0: 8-bitlik programlanabilir önölçekleyicili 8-bit zamanlayıcı/sayıcı
- Geliştirilmiş Timer1:
 - Önölçekleyicili 16-bit zamanlayıcı/sayıcı
 - Harici kapı giriş modu
 - Kendisine tahsis edilmiş düşük güçlü 32 kHz osilatör
- Timer2: 8-bit periyot kaydedicili, önölçekleyicili ve sonölçekleyicili 8-bitlik zamanlayıcı/sayıcı.
- Geliştirilmiş Yakalama, Karşılaştırma, PWM+ modülü:
 - 16-bitlik Yakalama modülünün maksimum çözünürlüğü 12,5 ns'dir.
 - Karşılaştırma modülünün maksimum çözünürlüğü 200 ns'dir.
 - 1, 2 ya da 4 çıkış kanallı 10-bit PWM'i, programlanabilir "dead time" a sahiptir ve maksimum frekansı 20 kHz'dir.
 - PWM çıkışı sevk ve idaresi kontrolü vardır.
- Yakalama, Karşılaştırma, PWM modülü:
 - 16-bitlik Yakalama modülünün maksimum çözünürlüğü 12,5 ns'dir.
 - 16-bitlik Karşılaştırma modülünün maksimum çözünürlüğü 200 ns'dir.
 - 10-bitlik PWM'in maksimum frekansı 20 kHz'dir.
- Geliştirilmiş USART modülü:
 - RS-485, RS-232 ve LIN 2,0 seri iletişim protokollerini,

- Otomatik hız (Baud) algılamayı,
 - Seri iletişimin başlama bitiyle birlikte otomatik uyanmayı destekler.
- İki pin üzerinden ICSP™ desteği vardır.
 - 3-hatlı SPI (tüm dört modda) ve I²C™ protokollerini destekleyen MSSP (Master Synchronous Serial Port) modülü vardır.

Şekil E.1’de PIC16F887’nin 40 pinli DIP kılıf türü gösterilmiştir. 44 pinli QFN ve TQFP kılıf türleri de mevcuttur.

40-pin PDIP



Şekil E.1 PIC16F887'nin kılıf yapısı

E2-PIC16F887’NİN BACAK (PİN) AÇIKLAMALARI

Tablo E.1 PIC16F887'nin bacak açıklamaları

İsim	İşlev	Giriş Türü	Çıkış Türü	Açıklama
RA0/AN0/ULPWU/ C12IN0-	RA0	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN0	AN	-	A/D Kanal 0 girişi
	ULPWU	AN	-	Ultra düşük güçlü uyanma girişi
	C12IN0-	AN	-	C1 ya da C2 Karşılaştırıcıları 0 numaralı negatif girişi

RA1/AN1/C12IN1-	RA1	TTL	CMOS	Genel amaçlı çift yönlü I/O portu.
	AN1	AN	-	A/D Kanal 1 girişi
	C12IN1-	AN	-	C1 ya da C2 Karşılaştırıcıları 1 numaralı negatif girişi
RA2/AN2/V _{REF-} /CV _{REF} /C2IN+	RA2	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN2	AN	-	A/D Kanal 2 girişi
	V _{REF-}	AN	-	A/D Negatif voltaj referans girişi
	CV _{REF}	-	AN	Karşılaştırıcı voltaj referansı çıkışı
	C2IN+	AN	-	C2 Karşılaştırıcısı pozitif girişi
RA3/AN3/V _{REF+} /C1IN+	RA3	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN3	AN	-	A/D Kanal 3 girişi
	V _{REF+}	AN	-	Programlama voltajı
	C1IN+	AN	-	C1 Karşılaştırıcısı pozitif girişi
RA4/T0CKI/C1OUT	RA4	TTL	CMOS	Genel amaçlı çift yönlü I/O portu.
	T0CKI	ST	-	Timer0 saat darbesi girişi
	C1OUT	-	CMOS	C1 Karşılaştırıcısı çıkışı
RA5/AN4/ \overline{SS} /C2OUT	RA5	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN4	AN	-	4 numaralı A/D Kanal girişi
	\overline{SS}	ST	-	Slave seçme girişi
	C2OUT	-	CMOS	C2 Karşılaştırıcısı çıkışı

RA6/OSC2/CLKOUT	RA6	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	OSC2	-	XTAL	Osilatör kristal çıkışı. Kristal osilatör modunda rezonatör ya da kristale bağlanır.
	CLKOUT	-	CMOS	Osilatör frekansının 4'te 1'i çıkış olarak elde edilir.
RA7/OSC1/CLKIN	RA7	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	OSC1	XTAL	-	Osilatör kristal/rezonatör girişi
	CLKIN	ST	-	Harici saat darbesi girişi. RC osilatör bağlantısı.
RB0/AN12/INT	RB0	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	AN12	AN	-	12 numaralı A/D Kanal girişi
	INT	ST	-	Harici kesme
RB1/AN10/C12IN3-	RB1	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	AN10	AN	-	10 numaralı A/D Kanal girişi
	C12IN3-	AN	-	C1 ya da C2 Karşılaştırıcıları 3 numaralı negatif girişi

RB2/AN8	RB2	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	AN8	AN	-	8 numaralı A/D Kanal girişi
RB3/AN9/PGM/ C12IN2-	RB3	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	AN9	AN	-	9 numaralı A/D Kanal girişi
	PGM	ST	-	Düşük voltaj ICSP™ etkinleştirme girişi
	C12IN2-	AN	-	C1 ya da C2 Karşılaştırıcıları 2 numaralı negatif girişi
RB4/AN11	RB4	TTL	CMOS	Çift yönlü I/O portu. Dâhili pull-up'lar için yazılımsal olarak programlanabilir.
	AN11	AN	-	11 numaralı A/D Kanal girişi
RB5/AN13/ $\overline{T1G}$	RB5	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	AN13	AN	-	13 numaralı A/D Kanal girişi
	$\overline{T1G}$	ST	-	Timer1 Kapı (Gate) girişi
RB6/ICSPCLK	RB6	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.

RB6/ICSPCLK	ICSPCLK	ST	-	Seri programlama için saat darbesi
RB7/ICSPDAT	RB7	TTL	CMOS	Genel amaçlı çift yönlü I/O portu. Bağımsız olarak kontrol edilebilen port durum değişim kesmesi. Dâhili pull-up'lar bağımsız olarak etkinleştirilebilir.
	ICSPDAT	ST	TTL	ICSP™ Veri I/O portu
RC0/T1OSO/T1CKI	RC0	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	T1OSO	-	XTAL	Timer1 osilatör çıkışı
	T1CKI	ST	-	Timer1 saat darbesi girişi
RC1/T1OSI/CCP2	RC1	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	T1OSI	XTAL	-	Timer1 osilatör girişi
	CCP2	ST	CMOS	2 numaralı Yakalama/Karşılaştırma/PWM modülü
RC2/P1A/CCP1	RC2	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	P1A	-	CMOS	A numaralı PWM çıkışı
	CCP1	ST	CMOS	1 numaralı Yakalama/Karşılaştırma/PWM modülü
RC3/SCK/SCL	RC3	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	SCK	ST	CMOS	SPI saat darbesi
	SCL	ST	OD	I ² C™ saat darbesi
RC4/SDI/SDA	RC4	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	SDI	ST	-	SPI veri girişi
	SDA	ST	OD	I ² C™ veri giriş çıkışı

RC5/SDO	RC5	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	SDO	-	CMOS	SPI veri çıkışı
RC6/TX/CK	RC6	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	TX	-	CMOS	EUSART asenkron gönderme
	CK	ST	CMOS	EUSART senkronize saat darbesi
RC7/RX/DT	RC7	ST	CMOS	Genel amaçlı çift yönlü I/O portu
	RX	ST	-	EUSART asenkron alma
	DT	ST	CMOS	EUSART senkron veri
RD0	RD0	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
RD1	RD1	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
RD2	RD2	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
RD3	RD3	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
RD4	RD4	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
RD5/P1B	RD5	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	P1B	-	CMOS	B numaralı PWM çıkışı
RD6/P1C	RD6	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	P1C	-	CMOS	C numaralı PWM çıkışı
RD7/P1D	RD7	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	P1D	-	CMOS	D numaralı PWM çıkışı

RE0/AN5	RE0	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN5	AN	-	5 numaralı A/D Kanal girişi
RE1/AN6	RE1	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN6	AN	-	6 numaralı A/D Kanal girişi
RE2/AN7	RE2	TTL	CMOS	Genel amaçlı çift yönlü I/O portu
	AN7	AN	-	7 numaralı A/D Kanal girişi
RE3/ \overline{MCLR} /V _{PP}	RE3	TTL	-	Genel amaçlı giriş portu
	\overline{MCLR}	ST	-	Dâhili pull-up'lı ana RESET girişi
	V _{PP}	HV	-	Programlama voltajı
VSS	VSS	Besleme	-	Şase referansı
VDD	VDD	Besleme	-	Pozitif besleme

Açıklama:

O = Çıkış

CMOS = CMOS uyumlu giriş ya da çıkış

HV = Yüksek Voltaj

XTAL = Kristal

I = Giriş

ST = CMOS seviyeli Schmitt Trigger giriş

TTL = TTL uyumlu giriş

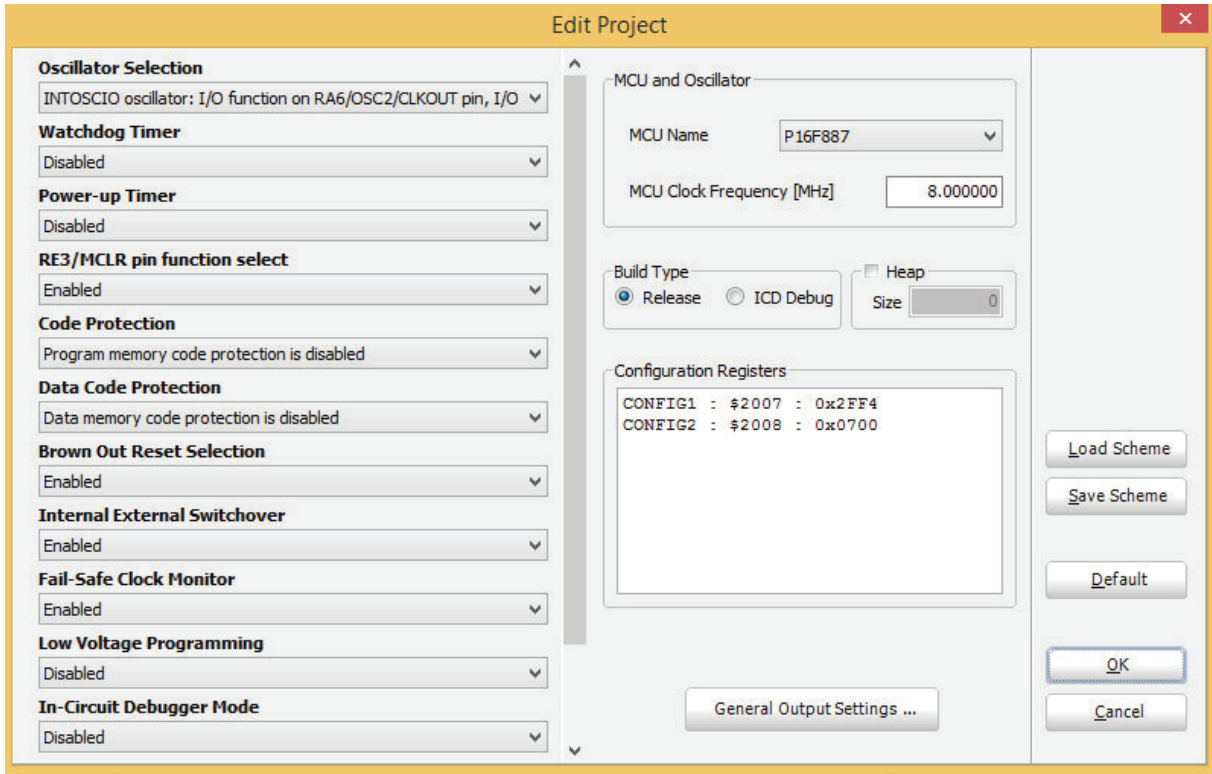
OD = Open Drain Çıkış

AN = Analog giriş ya da çıkış

E3-PIC16F887 MİKRODENETLEYİCİSİ İÇİN “mikroC PRO for PIC” PROJE EDITÖRÜ

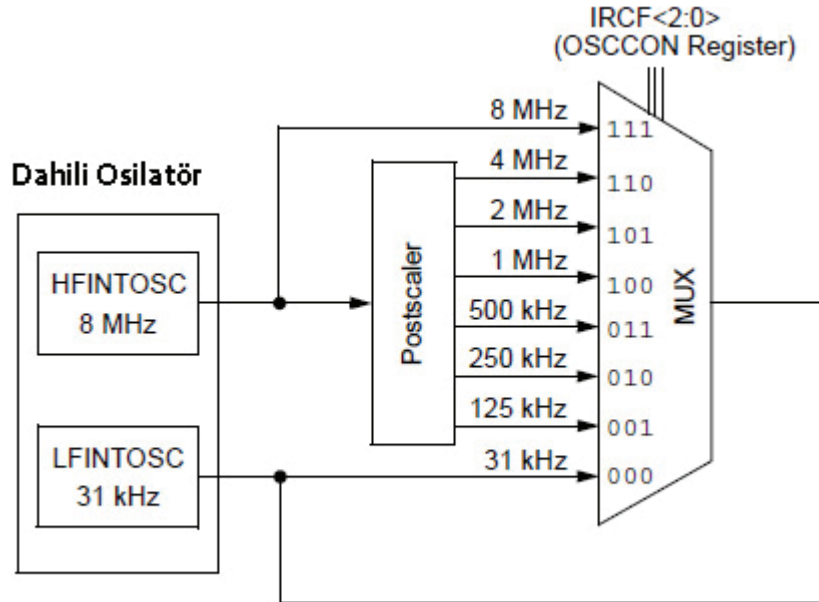
PIC16F887 mikrodenetleyicisinin iki adet yapılandırma sözcüğü vardır. Bu mikrodenetleyicinin de yapılandırma bitleri özel ayar belleği bölgesi (Special Configuration Memory Space – 2000h-3FFFh) alanında yer alır. CONFIG1 sözcüğü 2007h adresinde, CONFIG2 sözcüğü 2008h adresinde yer alır.

Şekil E.2’de PIC16F887 için proje editörü penceresi ve yapılandırma sözcüklerinin değeri gösterilmiştir. Çok sayıya ayar biti olduğundan ekranda “Brown-out Reset Selection” ve “Flash Program Memory Self Write” sözcükleri gözükmemiştir. Kaydırma çubuğunu kaydırarak o bitlerin güncel değerini görebilirsiniz.



Şekil E.2 PIC16F887'de Yapılandırma bitlerinin ayarlanması

PIC16F887 ile kitap içinde yapılan uygulamaların önemli bir kısmında 8 farklı osilatör tercihinden genellikle "INTOSCIO" değeri seçilmiştir. RA6 ve RA7 pinlerinin her ikisinin de I/O pini olarak kullanıldığı bu modda ya da RA6 pininden saat frekansının $\frac{1}{4}$ 'ünün alınmasını sağlayan "INTOSC" modunda, OSCCON kaydedicisi yardımıyla Şekil E.3'te gösterildiği gibi 8 farklı dâhili osilatörden istenilen seçilir.



Şekil E.3 OSCCON kaydedicisiyle tercihleri yapılan dâhili osilatörler

Proje editörü penceresinden seçilen osilatör türlerine göre CONFIG1 sözcüğünün ilk 3 biti değer değiştirmektedir. Örneğin diğer yapılandırma bitleri sabit kalmak koşuluyla osilatör tercihleri için CONFIG1 değeri;

CONFIG1: Ş2007: 0x2FF0-0x2FF8 arasında değişmektedir.

Dâhili osilatör tercihi yapıldıktan sonra yazılım içinde OSCCON kaydedicisinin ayarlanması gerekmektedir. Dikkat edileceği üzere aslında yüksek frekans osilatörü HFINTOSC (8 MHz) ve alçak frekans osilatörü LFINTOSC (31 kHz) olmak üzere iki dâhili osilatör vardır. Ancak HFINTOSC yüksek frekans osilatörü son ölçekleyici ile 6 alt frekansa bölünebilmektedir. Böylece dâhili olarak seçilebilen frekans değeri 8'e çıkar. Eğer kaydedici ayarı yapılmazsa varsayılan dâhili çalışma frekansı 4 MHz olur. Aşağıda PIC16F887'nin osilatör kontrol kaydedicisinin ayarları gösterilmiştir.

OSCCON: OSİLATÖR KONTROL KAYDEDİCİSİ (ADRES: 8Fh – PIC16F887)

U-0	R/W-1	R/W-1	R/W-0	R-1	R-0	R-0	R/W-0
—	IRCF2	IRCF1	IRCF0	OSTS ⁽¹⁾	HTS	LTS	SCS
bit 7							bit 0

Osilatör kontrol kaydedicisinin bitleri işlevsel olarak 3 gruba ayrılır:

- Frekans tercih bitleri (IRCF)
- Frekans durum bitleri (HTS, LTS)
- Sistem saat kontrol bitleri (OSTS, SCS)

bit 7 **Kullanılmıyor:** '0' olarak okunur.

bit 6-4 **IRCF<2:0>:** Dâhili osilatör tercih bitleridir.

111 = 8 MHz
110 = 4 MHz (varsayılan)
101 = 2 MHz
100 = 1 MHz
011 = 500 kHz
010 = 250 kHz
001 = 125 kHz
000 = 31 kHz (LFINTOSC)

bit 3 **OSTS:** Osilatör açılışı zaman aşımı durum bitidir⁽¹⁾.

1 = Cihaz CONFIG1 kaydedicisinin FOSC<2:0> bitleriyle tanımlanan harici saat frekansı kaynağından çalışıyor.

0 = Cihaz (HFINTOSC ya da LFINTOSC) dâhili osilatörlerinden biriyle çalışıyor.

bit 2 **HTS:** HFINTOSC durum bitidir (Yüksek frekans – 8 MHz'den 125 kHz'e).

1 = HFINTOSC dengeli

0 = HFINTOSC dengeli değil

bit 1 **LTS:** LFINTOSC durum bitidir (Alçak frekans – 31 kHz).

1 = LFINTOSC dengeli

0 = LFINTOSC dengeli değil

bit 0 **SCS:** Sistem saat frekansı seçim bitidir.

1 = Sistem saat frekansı olarak dâhili osilatör kullanılır.

0 = Sistem saat frekansı CONFIG1 kaydedicisinin FOSC<2:0> bitleriyle seçilir.

Not 1: Bit iki hızlı açılış (Two-Speed Start-up) moduyla lojik-0'a resetlenir ve osilatör modu olarak LP, XT ya da HS seçilir ya da güvenli mod (Fail-Safe) etkinleştirilir.

CONFIG1: YAPILANDIRMA SÖZCÜĞÜ 1

—	—	$\overline{\text{DEBUG}}$	LVP	FCMEN	IESO	BOREN1	BOREN0	
bit 15								bit 8

$\overline{\text{CPD}}$	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0	
bit 7								bit 0

bit 15-14 **Kullanılmıyor:** '1' olarak okunur.

bit 13 **$\overline{\text{DEBUG}}$:** Devre içi hata ayıklayıcısı (In-Circuit Debugger) modudur.

1 = Devre içi hata ayıklayıcısı kapalı, RB6/ICSPCLK ve RB7/ICSPDAT genel amaçlı I/O pinleridir.

0 = Devre içi hata ayıklayıcısı açık, RB6/ICSPCLK ve RB7/ICSPDAT pinleri hata ayıklayıcısı için tahsis edilir.

NOT: Devre içi hata ayıklayıcısı etkinleştirildiğinde dâhili seri programlama (In-Circuit Serial Programming) bitleri üzerinden hata ayıklama işlemi gerçekleştirilebilir. Bunun için Microchip firmasının geliştirmiş olduğu MPLAB ICD2 ve ICD3 donanımı ya da "PICFlash with mikroICD" donanımı kullanılabilir. Bu donanımlar mikrodenetleyici programlamada kullanıldıkları gibi aynı zamanda çevrimiçi ve eşzamanlı hata ayıklama özelliğine de sahiptir. Bunun için ilgili donanım bilgisayara bağlı ve ayarları yapılmışken, mikroC PRO for PIC editöründen Şekil 2.10'da gösterildiği gibi Hata Ayıklayıcısı (Debugger) çalıştırılır.

Yazmış olduğunuz program derlenip mikrodenetleyiciye yüklendiğinde artık MCU içindeki program adım adım çalıştırılabilir. Hata ayıklayıcının en önemli avantajlarından biri program satırlarını teker teker ve yavaşlatarak çalıştırabilmenizdir. Yazılımınız içinde kullanılan değişkenlerin ve kaydedicilerin değerleri program editöründen izlenebilirken aynı zamanda donanımınızın çalışmasını da eşzamanlı görebilirsiniz.



Şekil E.4 Devre içi hata ayıklamada kullanılan örnek bir donanım

bit 12 LVP: Düşük voltaj programlama etkinleştirme bitidir.

1 = RB3/PGM pini PGM işlevindedir. Düşük voltaj programlama etkinleştirilir.

0 = RB3 pini dijital I/O'dur. $\overline{\text{MCLR}}$ pini üzerinden yüksek voltaj (HV) programlama yapılır.

bit 11 FCMEN: Güvenli saat frekansı izlemesi etkinleştirme bitidir.

1 = Güvenli saat frekansı izlemesi açık

0 = Güvenli saat frekansı izlemesi kapalı

NOT: Güvenli saat frekansı izleme (Fail-Safe Clock Monitor – FSCM) modu, harici saat kaynaklarından (LP, XT, HS, EC ya da RC modları) birinde hata algılanması durumunda saat frekansı kaynağını otomatik olarak dâhili osilatöre anahtarlayan bir özelliktir.

bit 10 IESO: Dâhili/Harici değiştirme bitidir.

1 = Dâhili/Harici Switchover modu etkin

0 = Dâhili/Harici Switchover modu kapalı

NOT: İki hızlı açılış (Two-Speed Start-up) modunu yapılandırmada kullanılır. İki hızlı açılış modu aşağıdaki şekilde yapılandırılır:

- CONFIG1 yapılandırma sözcüğünün IESO biti lojik-1 yapılır. Böylece iki hızlı açılış modu etkinleştirilir.
- OSCCON kaydedicisinin SCS biti lojik-0'a ayarlanır.
- CONFIG1 yapılandırma sözcüğünün FOSC<2:0> bitleri LP, XT ya da HS moduna ayarlanır.

OSİLATÖR AÇILIŞ ZAMANLAYICISI (Oscillator Start-up Timer - OST):

Osilatör frekansı kaynağı olarak LP, XT ya da HS modlarından biri seçildiğinde (osilatörünüzün bunlardan biri olması gerekir), osilatör açılış zamanlayıcısı (Oscillator Start-up Timer - OST) OSC1 kaynağından 1024 salınım kadar sayma işlemi gerçekleştirir. Bu olay, ilk açılış resetlemesi (POR) olduğunda ve ilk besleme gecikmesi zamanlayıcısı (PWRT) zaman aşımı gerçekleştiğinde (CONFIG1 - $\overline{\text{PWRTÉ}}$ biti '0' ise) ya da uykudan uyanıldığında olur. Bu sırada program sayıcı (PC) artmaz ve program çalışması askıya alınır.

OST, quartz kristal rezonatör ya da seramik rezonatör kullanan osilatör devresinin osilatör modülüne dengeli bir saat darbesi sunar. Saat kaynakları arasında anahtarlama gerçekleştiğinde yeni saat kaynağının dengeye gelmesi için bir gecikme gerekir. Bu gecikme değerleri ve PIC16F887'nin osilatör devresinin çalışma ilkesiyle ilgili ayrıntılar için ilgili veri kılavuzunu incelemeniz tavsiye edilir.

İKİ HIZLI AÇILIŞ (Two-Speed Start-up): İki hızlı açılış modu ek bir güç tasarrufu sağlar. Bu işlem, harici osilatör kaynağının hazır hale gelmesiyle kodu çalıştırma arasındaki gecikme en aza indirilerek gerçekleştirilir. Özellikle uyku modunun yoğun kullanıldığı uygulamalarda kullanışlıdır. Mikrodenetleyicinin uyku modundan uyanmasını sağlayan bir olay meydana geldiğinde, iki hızlı açılış modu harici osilatörün açılış zamanlayıcısını uyanma sırasında geçen süreden çıkartır. Böylece MCU'nun toplam güç tüketimi azalır. Bu mod uyku modundan uyanan MCU'nun saat frekans kaynağı olarak bir süreliğine INTOSC'yi kullanmasını sağlar. Ardından ana osilatörün dengeye gelmesi beklenmeden tekrar uyku moduna geri dönlür.

İki hızlı açılış moduna; POR durumunda, aktif kılındıysa PWRT zaman aşımı gerçekleştiğinde ve uyku modundan uyanıldığında girilir. İki hızlı açılış sırası şu şekilde gerçekleşir:

1. Uykudan ya da ilk güç uygulandığında MCU'nun resetlenmesinden sonra (POR) uyanılır.
2. Komutlar OSCCON kaydedicisinin IRCF<2:0> bitleriyle ayarlanmış dâhili osilatör frekansı hızıyla çalıştırılır.
3. OST 1024 saat çevrimi sayacak kadar etkinleştirilir.
4. OST zaman aşımı olur, dâhili osilatörün düşen kenarı beklenir.
5. OSTS kurulur.
6. Sistem saati yeni saatin (LP, XT ya da HS) bir sonraki düşen kenarına kadar lojik düşük seviyede tutulur.
7. Sistem saati tekrar harici saat kaynağına anahtarlanır.

bit 9-8 BOREN<1:0>: Voltaj düşmesi reset tercihi biti ⁽¹⁾

11 = BOR etkin

10 = BOR çalışma sırasında açık SLEEP modunda kapalı

01 = BOR PCON kaydedicisinin SBOREN bitiyle kontrol edilir

00 = BOR kapalı

bit 7 CPD: Veri kodu koruma bitidir⁽²⁾.

1 = Veri belleği kod koruması kapalı

0 = Veri belleği kod koruması açık

bit 6 CP: Kod koruma bitidir⁽³⁾.

1 = Program belleği kod koruması kapalı

0 = Program belleği kod koruması açık

bit 5 **MCLRE:** RE3/ $\overline{\text{MCLR}}$ pin işlevi seçim bitidir.

1 = RE3/ $\overline{\text{MCLR}}$ pini $\overline{\text{MCLR}}$ olarak işlev görür.

0 = RE3/ $\overline{\text{MCLR}}$ pini dijital giriş olarak işlev görür. Reset hattı dâhili olarak V_{DD} 'ye bağlanır.

bit 4 **$\overline{\text{PWRTÉ}}$:** Güç dengesi zamanlayıcısı (Power-up Timer - PWRT) etkinleştirme bitidir.

1 = PWRT kapalı

0 = PWRT açık

bit 3 **WDTE:** Vardiya zamanlayıcısı (Watchdog Timer) etkinleştirme bitidir.

1 = WDT etkin

0 = WDT kapatılır ve WDTCN kaydedicisinin SWDTEN bitiyle yazılım içinden etkinleştirilir.

bit 2-0 **FOSC<2:0>:** Osilatör seçme bitleridir.

111 = RC osilatörü: RA6/OSC2/CLKOUT pininden saat frekansının $\frac{1}{4}$ 'ü alınır, RC osilatör devresi RA7/OSC1/CLKIN pinine bağlanır.

110 = RCIO osilatörü: RA6/OSC2/CLKOUT pini dijital I/O, RC osilatör devresi RA7/OSC1/CLKIN pinine bağlanır.

101 = INTOSC osilatörü: RA6/OSC2/CLKOUT pininden saat frekansının $\frac{1}{4}$ 'ü alınır, RA7/OSC1/CLKIN pini dijital I/O olarak kullanılır.

100 = INTOSCIO osilatörü: RA6/OSC2/CLKOUT ve RA7/OSC1/CLKIN pinlerinin her ikisi de dijital I/O olarak kullanılır.

011 = EC: RA6/OSC2/CLKOUT pini dijital I/O, RA7/OSC1/CLKIN pini ise harici saat darbesi üretici girişi olarak kullanılır.

010 = HS osilatörü: RA6/OSC2/CLKOUT ve RA7/OSC1/CLKIN pinlerinde yüksek hızlı kristal/rezonatör bağlıdır.

001 = XT osilatörü: RA6/OSC2/CLKOUT ve RA7/OSC1/CLKIN pinlerinde kristal/rezonatör bağlıdır.

000 = LP osilatörü: RA6/OSC2/CLKOUT ve RA7/OSC1/CLKIN pinlerinde düşük güç kristal bağlıdır.

Not 1: Voltaj düşmesi resetlenmesini etkinleştirmek ilk besleme gecikmesi zamanlayıcısını (güç dengeleyicisi) (Power-up Timer) otomatik olarak etkinleştirmez.

2: Kod koruması kapatıldığında EEPROM üzerindeki tüm veri silinir.

3: Kod koruması kapatıldığında tüm program belleği temizlenir.

CONFIG2: YAPILANDIRMA SÖZCÜĞÜ 2

—	—	—	—	—	WRT1	WRT0	BOR4V
bit 15					bit 8		
—	—	—	—	—	—	—	—
bit 7					bit 0		

bit 15-11 Kullanılmıyor: '1' olarak okunur

bit 10-9 **WRT<1:0>**: Flash Program belleği öz yazma etkinleştirme bitleri

PIC16F883/PIC16F884

00 = 0000h - 07FFh arası yazma korumalı, 0800h - 0FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

01 = 0000h - 03FFh arası yazma korumalı, 0400h - 0FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

10 = 0000h - 00FFh arası yazma korumalı, 0100h - 0FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

11 = Yazma koruması kapalı

PIC16F886/PIC16F887

00 = 0000h - 0FFFh arası yazma korumalı, 1000h - 1FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

01 = 0000h - 07FFh arası yazma korumalı, 0800h - 1FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

10 = 0000h - 00FFh write arası yazma korumalı, 0100h - 1FFFh arası EECON kaydedicisi tarafından değiştirilebilir.

11 = Yazma koruması kapalı

PIC16F882

00 = 0000h - 03FFh arası yazma korumalı, 0400h - 07FFh arası EECON kaydedicisi tarafından değiştirilebilir.

01 = 0000h - 00FFh arası yazma korumalı, 0100h - 07FFh arası EECON kaydedicisi tarafından değiştirilebilir.

11 = Yazma koruması kapalı

bit 8 **BOR4V**: Voltaj düşmesi reset tercihi biti

0 = Voltaj düşmesi resetlemesi 2.1V'a kurulur.

1 = Voltaj düşmesi resetlemesi 4.0V'a kurulur.

bit 7-0 **Kullanılmıyor:** '1' olarak okunur.

Dikkat: PIC16F887'nin osilatör kaynaklarıyla ve diğer yapılandırma bitleriyle ilgili bilgilendirmenin yapıldığı bu kısımda, PIC16F628A'nın ilgili başlığı altında anlatılmış olan konular tekrar olmaması için ayrıntılı olarak açıklanmamıştır. Daha ayrıntılı bilgi için PIC16F628A'nın proje editörü penceresi ile ilgili başlığının incelenmesi faydalı olacaktır.

PIC16F887 çok gelişmiş osilatör tercihlerine sahiptir. Bunun yanında vardiya zamanlayıcısının da daha ayrıntılı olarak programlanmasını sağlayan WDTCON kaydedicisine sahiptir. Bu kaydediciyle vardiya zamanlayıcısının yazılımsal olarak kontrol edilmesi ve uyku modunda daha uzun uyku sürelerinin elde edilmesi mümkündür.

WDTCON: VARDİYA ZAMANLAYICISI KONTROL KAYDEDİCİSİ (ADRES: 105h – PIC16F887)

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN ⁽¹⁾
bit 7							bit 0

bit 7-5 **Kullanılmıyor:** '0' olarak okunur

bit 4-1 **WDTPS<3:0>:** Vardiya zamanlayıcısı periyot seçim bitleridir.

Bit Değeri = Önölçekleme Oranı

0000 = 1:32

0001 = 1:64

0010 = 1:128

0011 = 1:256

0100 = 1:512 (Reset değeri)

0101 = 1:1024

0110 = 1:2048

0111 = 1:4096

1000 = 1:8192

1001 = 1:16384

1010 = 1:32768

1011 = 1:65536

1100 = ayrılmıştır

1101 = ayrılmıştır

1110 = ayrılmıştır

1111 = ayrılmıştır

bit 0 **SWDTEN:** Yazılımsal olarak vardiya zamanlayıcısını etkinleştirme ya da kapatma bitidir⁽¹⁾.

1 = WDT açılır

0 = WDT kapatılır (Reset değeri)

Not 1: Eğer WDT yapılandırma biti = 1 ise WDT her zaman etkindir. Eğer WDT yapılandırma biti = 0 ise bu kontrol biti yardımıyla yazılımsal olarak kontrol etmek mümkün olur.

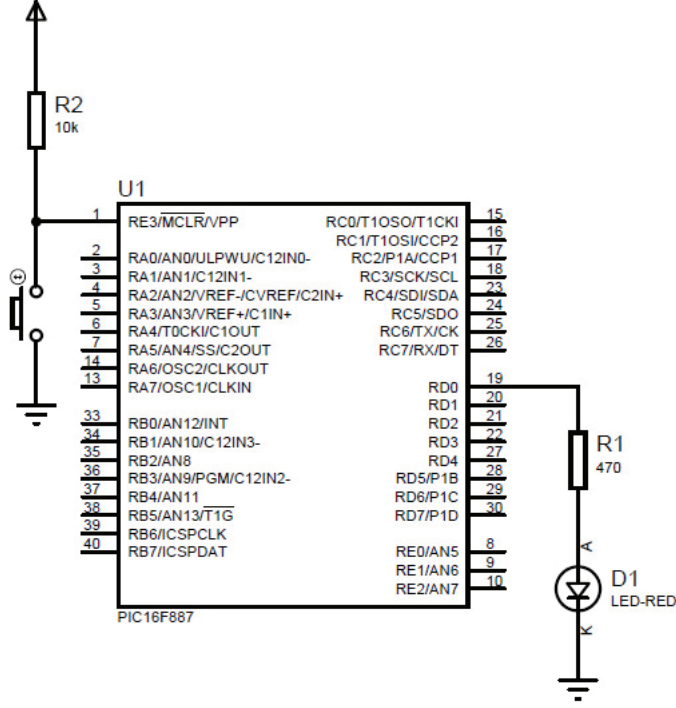
PIC16F887’de yazılımsal olarak ayarlanan WDT değeri nominal 268 sn’dir. Ancak OPTION_REG kaydedicisinin önölçekleme değeri de devreye alındığında bu süre çarpan faktörü nispetince uzar. OPTION_REG önölçekleyicisi WDT için ayarlanmadığında ya da 1:1 olduğunda ve WDTCON önölçekleme oranı da 1:32’de olduğunda minimum resetlenme süresi 0,576 sn kadardır. Şekil E.5’te PIC16F887’nin WDTCON kaydedicisiyle yazılımsal olarak vardiya zamanlayıcısının kullanımı gösterilmiştir.

Uygulamada, vardiya zamanlayıcısı yazılımsal olarak ayarlanmış ve aynı anda OPTION_REG kaydedicisinin PSA biti lojik-0 yapılarak WDT önölçekleme değeri devre dışı bırakılmıştır. Uygulamada WDTCON kaydedicisinin önölçekleme değeri ‘0001’ olarak yüklenmiş böylece $0,576 \times 2 = 1,15$ sn resetlenme süresi elde edilmiştir. Timer1 modülü devreye alınmak suretiyle yaklaşık 0,5 sn’lik kesme süresi elde edilmiştir. Elde edilen kesme süresi kullanılarak fasıllı olarak sinyal veren bir devre uygulaması gerçekleştirilmiştir. Timer1 kesme süresi şu şekilde hesaplanmıştır:

$$\text{Süre} = \frac{4 \times (65536 - TMR1) \times \text{prescaler}}{F_{osc}} = \frac{4 \times (65536 - 61440) \times 1}{31000} = 0,528 \text{ sn}$$

61440 değeri TMR1L = 0 ve TMR1H = F0h değerleri yüklenerek elde edilmiştir. OSCCON kaydedicisiyle mikrodenetleyicinin F_{osc} osilatör frekansı 31 kHz olarak ayarlanmıştır. Böylece minimum akımla çalışan ve uyarılan bir uygulama gerçekleştirilmiştir.

Eğer yazılımınızda uyku modunu devreye alırsanız fasıllı olarak uyanan ve ikaz verdikten sonra Timer1 kesmesi süresince uyanık kaldıktan sonra tekrar uykuya giren bir uygulama yapabilirsiniz. Ancak uyku komutunun WDT’yi resetlemesinden dolayı bekleme süresi bir müddet daha artacaktır. Uyku moduyla yapacağınız böyle bir uygulamada güç tüketimini çok daha düşük seviyelere indirmiş olursunuz. Tablo E.2’de bu şekilde güncellenmiş uygulama kodu gösterilmiştir. Derleme öncesi, proje düzenleyicisi penceresinde “Watchdog Timer” ın kapalı (disabled) olmasına ve osilatör kaynağı olarak INTOSC ya da INTOSCIO’nun seçili olmasına dikkat ediniz. Program içinde $F_{osc} = 31$ kHz LFINTOSC olarak ayarlandığı için, proje düzenleyicisinde ve PROTEUS-ISIS® uygulamasında frekans değeri olarak ne ayarladığınızı önemli değildir.



Şekil E.5 PIC16F887 ile yazılımsal vardiya zamanlayıcısı uygulaması

Tablo E.2 PIC16F887 ile yazılımsal vardiya zamanlayıcısı uygulaması mikroC kodu

```
#define KONTROL PORTD.RD0 // D portunun 0. pinine (RD0) KONTROL ismi veriliyor
unsigned int sayac, saniye;
void main() {
    OPTION_REG = 0x00; // Ön ölçekleyici WDT için ayarlanmadı
    TRISD = 0;
    PORTD = 0;
    ANSEL = 0;
    ANSELH = 0;
    CM1CON0 = 0;
    CM2CON0 = 0;
    OSCCON=0X00;
    INTCON.GIE=1;
    INTCON.PEIE=1;
    PIE1.TMR1IE=1;
    T1CON=0X01;
    TMR1L=0X00;
    TMR1H=0XF0; //61440 değeri yükleniyor
    WDTCON=0x03;
    KONTROL = 1; // RD0 lojik-1 yapılıyor
    while(!PIR1.TMR1IF); //Timer1 kesmesi oluşana kadar D0 çıkışı aktif kalıyor
    saniye=0;
    KONTROL = 0;
    asm{ // Assembly kod bloğu açılıyor
        SLEEP; // Mikrodenetleyici uyku moduna alınıyor
        NOP;
    }
}
void interrupt(){
    if(PIR1.TMR1IF){
        PIR1.TMR1IF=0;
        TMR1L=0X00;
    }
}
```



```

TMR1H=0XF0;
asm{// Timer1 tekrar açılmadan önce geçici bekleme yapılıyor
nop;nop;nop;nop;nop;
}
T1CON.TMR1ON=1;
}
}

```

E4-PIC16F887 İÇİN TEMEL ELEKTRİKSEL ÖZELLİKLER

PIC16F887'den en yüksek verimi almak için çalışma koşullarına göre elektriksel verileri göz önünde bulundurmanız gerekir. Aşağıda mikrodenetleyicinin ön görülen üst sınır değerleri listelenmiştir. Mikrodenetleyicinizin uzunca bir süre bu değerlerin üzerinde çalıştırılması cihazınıza ve sisteminize kalıcı zararlar verebilir.

Besleme altındaki ortam sıcaklığı	-40° – +125°C
Saklanma koşulları altındaki sıcaklık.....	-65°C – +150°C
V _{SS} 'ye göre V _{DD} üzerindeki voltaj	-0,3V – +6,5V
V _{SS} 'ye göre \overline{MCLR} üzerindeki voltaj	-0,3V – +13,5V
V _{SS} 'ye göre tüm diğer pinlerdeki voltaj	-0,3V – (V _{DD} + 0,3V)
Toplam güç tüketimi ⁽¹⁾	800 mW
V _{SS} pininden geçen en yüksek akım.....	300 mA
V _{DD} pinine giren en yüksek akım.....	250 mA
Giriş kenetleme akımı, I _{IK} (V _I < 0 ya da V _I > V _{DD}).....	± 20 mA
Çıkış kenetleme akımı, I _{OK} (V _O < 0 ya da V _O > V _{DD})	± 20 mA
I/O pini tarafından çekilen en yüksek çıkış akımı	25 mA
I/O pini tarafından sunulan en yüksek çıkış akımı	25 mA
PORTA, PORTB ve PORTE (üçü toplam) tarafından çekilen en yüksek akım	200 mA
PORTA, PORTB ve PORTE (üçü toplam) tarafından sunulan en yüksek akım	200 mA
PORTC ve PORTD (ikisi toplam) tarafından çekilen en yüksek akım	200 mA
PORTC ve PORTD (ikisi toplam) tarafından sunulan en yüksek akım	200 mA

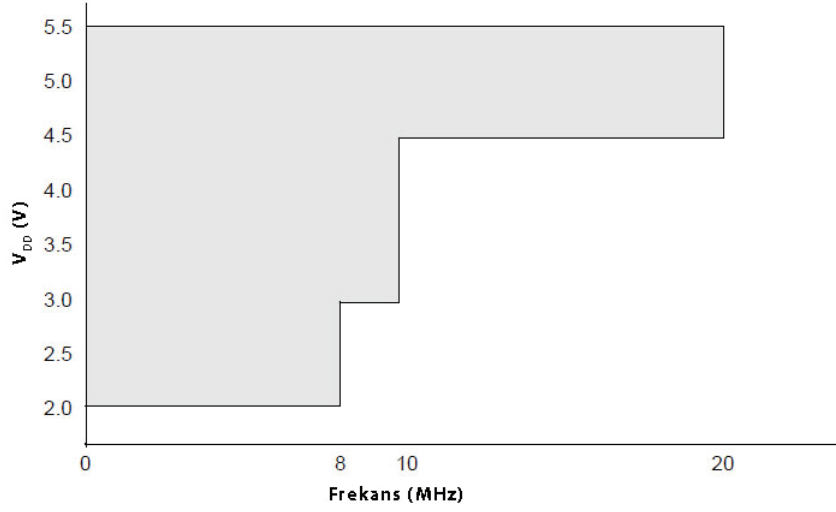
Not 1: Güç tüketimi şu bağıntıyla hesaplanır: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$.

Burada; I_{DD}: Ana besleme akımı

I_{OH}: I/O portlarının (OSC2 CLKOUT modunu da içerir) V_{OH} (en az V_{DD}-0,7 V) olması durumunda geçen akım = 8,5 mA (V_{DD} = 4,5V)

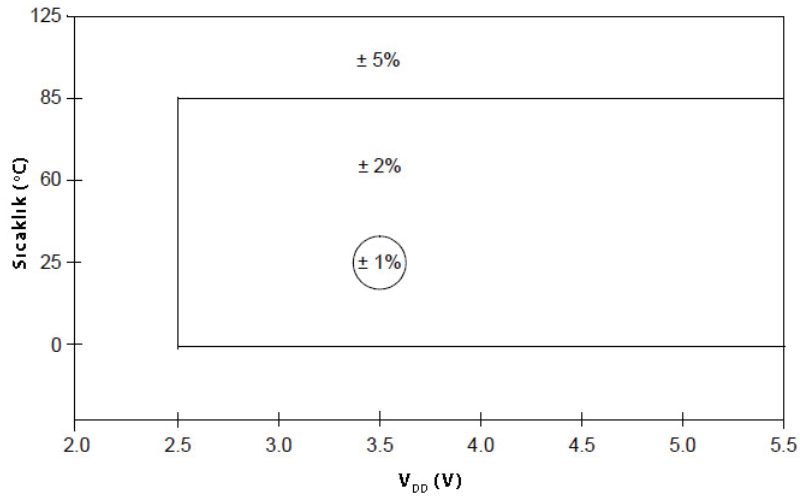
I_{OL} : I/O portlarının (OSC2 CLKOUT modunu da içerir) V_{OL} (en çok 0,6 V) olması durumunda geçen akım = -3 mA ($V_{DD} = 4,5V$)

Tüm PIC mikrodenetleyicilerinin ideal çalışma frekansları besleme voltajıyla birlikte değişir. Osilatör frekansı yükseldikçe ihtiyaç duyulan besleme voltajı da artar. Bu nedenle düşük güç tüketiminin istendiği (uyku modu, vb.) ve mikrodenetleyicinin batarya ve pil gibi sabit bir DC kaynaktan beslendiği uygulamalarda bu duruma dikkat edilmelidir. Besleme gerilimini düşürmeniz durumunda çalışma frekansını da düşürmeniz uygun olacaktır. Uygulama kodunuzu da bu koşullara göre revize ederek yazmanız yerinde olur. Şekil E.6'da frekans ve çalışma voltajı ilişkisi gösterilmiştir. Şekilde görüldüğü üzere çalışma frekansı yükseldikçe ihtiyaç duyulan besleme gerilimi de (V_{DD}) artmaktadır. Taralı alan izin verilen bölgeyi ifade etmektedir.



Şekil E.6 Frekans ve çalışma voltajı ilişkisi

Sıcaklık sistem kararlılığına etki eden en önemli etkenlerdendir. Şekil E.7'de gösterildiği gibi HFINTOSC dâhili yüksek frekanslı osilatörün kararlılığı sıcaklık arttıkça azalmaktadır. Bu durum zaman temelli uygulamalarınızda hatalara neden olur.



Şekil E.7 Sıcaklık ve çalışma voltajı ilişkisi

Not: Zamanlayıcılar, karşılaştırıcılar, analog modüller, osilatörler, DC bileşenler ve AC bileşenler gibi çeşitli modüllerin ve katların elektriksel verileriyle ilgili daha ayrıntılı bilgi için veri kılavuzunu incelemeniz tavsiye edilir.

EK-F

F1-18F2550 MİKRODENETLEYİCİSİNİN TEKNİK ÖZELLİKLERİ

PIC18F2550, 28-pinli 18F serisi yüksek ölçekli (High-End) bir mikrodenetleyicidir. Saniye başına 12 milyon komut işleme özellikli (12 MIPS), 8-bit çekirdek mimarisinde, komut uzunluğu 16-bit olan yüksek performans ve düşük güç tüketim (nanoWatt) özelliğine sahiptir. PIC18F4550 ile aynı sınıftandır.

Evrensel Seri Yolu Özellikleri:

- USB V2.0 uyumludur.
- Düşük hız (1.5Mb/s) ve yüksek hız (12Mb/s) veri transferi özelliklidir.
- Kontrol, kesme, eşfrekans (isochronous) ve yığın transferlerini destekler.
- 32 adet (16 çift yönlü) USB bitiş noktası (endpoint) vardır.
- USB için 1 Kbyte çift erişimli RAM belleği vardır.
- Yongası içinde voltaj regülatörlü dâhili (on-chip) USB alıcı-vericisi vardır.
- USB bağlantısı için arayüzü vardır.

Güç Yönetimi Modları:

- Run (İşliyor): CPU açık, donanımlar açık
- Idle (Boş): CPU kapalı, donanımlar açık
- Sleep (Uykuda): CPU kapalı, donanımlar kapalı
- Boştaki akımı 5,8uA'e kadar iner.
- Uyku modundaki akım 0,1uA'e kadar iner.
- Timer1 osilatörü: 32 kHz 2V'ta 1,1uA
- Bekçi zamanlayıcısı: 2,1uA
- İki-hızlı osilatör açılış seçeneği

Esnek Osilatör Yapısı:

- USB için Yüksek Hassasiyetli PLL (Phase Locked Loop) devresini de içeren dört kristal modu
- 48 MHz'e kadar İki harici saat modu
- Kullanıcı tarafından ince frekans ayarına izin veren, 31 kHz – 8 MHz arasında 8 farklı kullanıcı seçimli dâhili osilatör bloğu
- 32 kHz'de Timer1'i kullanan ikincil osilatör
- Mikrodenetleyici ve USB modülünün farklı saat hızlarında çalışması için çift osilatör seçeneği
- Herhangi bir saat darbesi durursa güvenli kapanmaya imkân sunan emniyetli saat monitörü

Önemli Çevresel Aygıt Özellikleri:

- Yüksek giriş/kaynak (sink/source) akımı: 25 mA/25 mA
- 3 adet harici kesme
- 4 adet Timer modülü (Timer0 – Timer3)
- 2 adete kadar Yakalama/Karşılaştırma/PWM (CCP) modülü:
 - 16-bitlik Yakalama modülünün maksimum çözünürlüğü 5,2 ns'dir.
 - 16-bitlik Karşılaştırma modülünün maksimum çözünürlüğü 83,3 ns'dir.

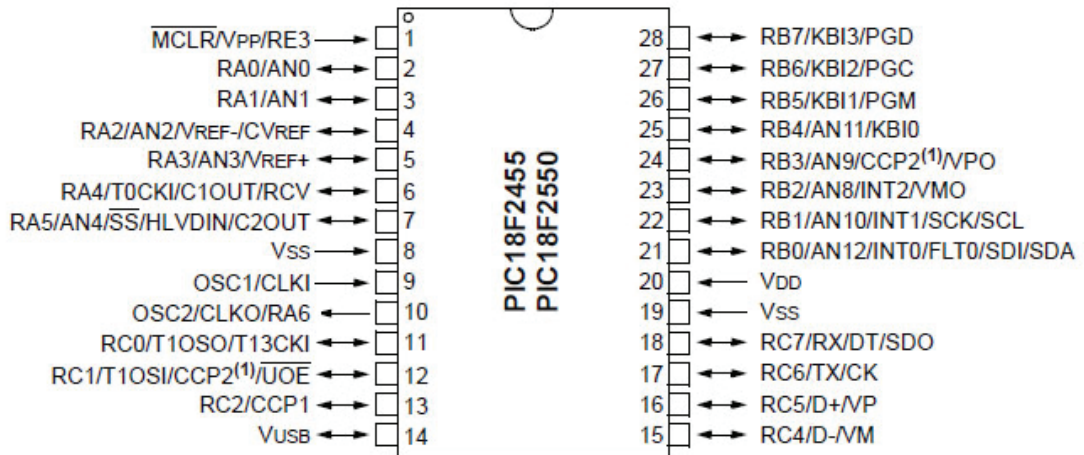
- PWM çıkışı: PWM çözünürlüğü 1-10bit arasındır.
- Geliştirilmiş CCP modülü (ECCP)
- Geliştirilmiş USART modülü (EUSART):
 - LIN veriyolu desteği
- MSSP modülü
 - 4 modu da destekleyen 3-Wire SPI
 - Master ve Slave modlarını destekleyen I²C™
- Programlanabilir yakalama zamanına sahip, 13 kanallı 10-bitlik Analog Dijital Dönüştürücü
- Giriş çoklamalı çift analog karşılaştırıcı

Özel Mikrodenetleyici Özellikleri:

- Opsiyonel genişletilmiş komut setiyle optimize edilmiş C derleyici mimarisi
- 100.000 Yazma/Silme çevrimine sahip Geliştirilmiş Flash Program Belleği
- 1.000.000 Yazma/Silme çevrimine sahip Veri EEPROM Belleği
- Flash/Veri EEPROM bilgi saklama ömrü: > 40 yıldır
- Yazılım kontrolü altında kendi başına programlanabilme
- Kesmeler için öncelik seviyeleri
- 8x8 tek çevrim donanım çoklayıcısı
- Genişletilmiş bekçi zamanlayıcısı (WDT):
 - 41 ms'den 131 s'ye kadar programlanabilir periyotlu
- Programlanabilir kod koruması
- İki pin yardımıyla tek kaynaklı 5V ICSP™ programlama desteği
- İki pin üzerinden devre içi hata ayıklayıcısı (ICD – In-Circuit Debugger)
- Opsiyonel olarak tahsis edilmiş ICD/ICSP portu (yalnızca 44-pinli TQFP kılıf türü modelinde)
- Geniş çalışma voltajı aralığına (2,0V – 5,0V)

Şekil F.1'de PIC18F2550'nin 28 pinli SDIP ve SOIC kılıf türü için bacak numaraları gösterilmiştir.

28-Pin PDIP, SOIC



Şekil F.1 PIC18F2550'nin kılıf yapısı

F2-PIC18F2550'NİN BACAK (PİN) AÇIKLAMALARI

Tablo F.1 PIC18F2550'nin bacak açıklamaları

İsim	Pin Numarası	İşlev	Pin Türü	Tampon Türü	Açıklama
RA0/AN0	2	RA0	I/O	TTL	Dijital I/O
		AN0	I	Analog	0 numaralı analog girişi
RA1/AN1	3	RA1	I/O	TTL	Dijital I/O
		AN1	I	Analog	1 numaralı analog girişi
RA2/AN2/V _{REF-} /CV _{REF}	4	RA2	I/O	TTL	Dijital I/O
		AN2	I	Analog	2 numaralı analog girişi
		V _{REF-}	I	Analog	A/D referans voltaj (low) girişi
		CV _{REF}	O	Analog	Analog karşılaştırıcı referans çıkışı
RA3/AN3/V _{REF+}	5	RA3	I/O	TTL	Dijital I/O
		AN3	I	Analog	3 numaralı analog girişi
		V _{REF+}	I	Analog	A/D referans voltaj (high) girişi
RA4/T0CKI/C1OUT/ RCV	6	RA4	I/O	ST	Dijital I/O
		T0CKI	I	ST	Timer0 harici saat darbesi girişi
		C1OUT	O	-	1 numaralı karşılaştırıcı çıkışı
RA4/T0CKI/C1OUT/ RCV		RCV	I	TTL	Harici USB alıcı/vericisi RCV girişi
RA5/AN4/ \overline{SS} / HLVDIN/C2OUT	7	RA5	I/O	TTL	Dijital I/O
		AN4	I	Analog	4 numaralı analog girişi
		\overline{SS}	I	TTL	SPI slave seçme girişi
		HLVDIN	I	Analog	Yüksek/Alçak (High/Low) voltaj algılama girişi
		C2OUT	O	-	2 numaralı karşılaştırıcı çıkışı
RA6/OSC2/CLKO	10	RA6	I/O	TTL	Genel amaçlı çift yönlü I/O
		OSC2	O	-	Osilatör kristal çıkışı. Kristal osilatör modunda rezonatör ya da kristale bağlanır.
		CLKO	O	-	Seçme modlarında, OSC2 pini OSC1 frekansının 4'te 1'i

					frekansında çıkış üretir ve bir komut çevrimi hızını sağlar.
OSC1/CLKI	9	OSC1	I	Analog	Osilatör kristal girişi ya da harici saat darbesi kaynağı giriştir.
		CLKI	I	Analog	Harici saat darbesi kaynağı giriştir. Her zaman pinin OSC1 işleviyle ilişkilidir.
RB0/AN12/INT0/ FLT0/SDI/SDA	21	RB0	I/O	TTL	Dijital I/O
		AN12	I	Analog	12 numaralı analog girişi
		INT0	I	ST	0 numaralı harici kesme girişi
		FLT0	I	ST	PWM hata girişi (CCP 1)
		SDI	I	ST	SPI veri girişi
		SDA	I/O	ST	I ² C™ veri I/O
RB1/AN10/INT1/ SCK/SCL	22	RB1	I/O	TTL	Dijital I/O
		AN10	I	Analog	10 numaralı analog girişi
		INT1	I	ST	1 numaralı harici kesme girişi
		SCK	I/O	ST	SPI modu için eşzamanlı seri saat darbesi giriş/çıkışı
		SCL	I/O	ST	I ² C™ modu için eşzamanlı seri saat darbesi giriş/çıkışı
RB2/AN8/INT2/ VMO	23	RB2	I/O	TTL	Dijital I/O
		AN8	I	Analog	8 numaralı analog girişi
		INT2	I	ST	2 numaralı harici kesme girişi
		VMO	O	-	Harici USB alıcı/vericisi VMO çıkışı
RB3/AN9/CCP2/VPO	24	RB3	I/O	TTL	Dijital I/O
		AN9	I	Analog	9 numaralı analog girişi
		CCP2 ⁽¹⁾	I/O	ST	2 numaralı Capture girişi/2 numaralı Compare çıkışı/PWM2 çıkışı
		VPO	O	-	Harici USB alıcı/vericisi VPO çıkışı
RB4/AN11/KBIO	25	RB4	I/O	TTL	Dijital I/O

		AN11	I	Analog	11 numaralı analog girişi
		KBIO	I	TTL	0 numaralı durum değişimi algılama kesmesi pini
RB5/KBI1/PGM	26	RB5	I/O	TTL	Dijital I/O
		KBI1	I	TTL	1 numaralı durum değişimi algılama kesmesi pini
		PGM	I/O	ST	Düşük voltaj ICSP™ programlama etkinleştirme pini
RB6/KBI2/PGC	27	RB6	I/O	TTL	Dijital I/O
		KBI2	I	TTL	2 numaralı durum değişimi algılama kesmesi pini
		PGC	I/O	ST	Devre üzerinde hata ayıklama ve ICSP™ programlama saat darbesi pini
RB7/KBI3/PGD	28	RB7	I/O	TTL	Dijital I/O
		KBI3	I	TTL	3 numaralı durum değişimi algılama kesmesi pini
		PGD	I/O	ST	Devre üzerinde hata ayıklama ve ICSP™ programlama veri pini
RC0/T1OSO/T13CKI	11	RC0	I/O	ST	Dijital I/O
		T1OSO	O	-	Timer1 osilatör çıkışı
		T13CKI	I	ST	Timer1/Timer3 harici CKI
RC1/T1OSI/CCP2/ \overline{UOE}	12	RC1	I/O	ST	Dijital I/O
		T1OSI	I	CMOS	Timer1 osilatör girişi
		CCP2 ⁽²⁾	I/O	ST	2 numaralı Capture girişi/2 numaralı Compare çıkışı/PWM2 çıkışı
		\overline{UOE}	O	-	Harici USB alıcı/vericisi \overline{OE} çıkışı
RC2/CCP1	13	RC2	I/O	ST	Dijital I/O
		CCP1	I/O	ST	1 numaralı yakalama (Capture) girişi / 1 numaralı karşılaştırma (Compare) çıkışı /PWM1 çıkışı
RC4/D-/VM	15	RC4	I	TTL	Dijital giriş

		D-	I/O	-	USB diferansiyel eksi hattı (giriş/çıkış)
		VM	I	TTL	Harici USB alıcı/vericisi VM girişi
RC5/D+/VP	16	RC5	I	TTL	Dijital giriş
		D+	I/O		USB diferansiyel artı hattı (giriş/çıkış)
		VP	O	TTL	Harici USB alıcı/vericisi VP girişi
RC6/TX/CK	17	RC6	I/O	ST	Dijital I/O
		TX	O	-	EUSART asenkron gönderme
		CK	I/O	ST	EUSART senkronize saat darbesi (RX/DT'ye bakınız)
RC7/RX/DT/SDO	18	RC7	I/O	ST	Dijital I/O
		RX	I	ST	EUSART asenkron alma
		DT	I/O	ST	EUSART senkronize veri (TX/CK'ya bakınız)
		SDO	O	-	SPI veri çıkışı
RE3/ $\overline{\text{MCLR}}$ /V _{PP}	1	RE3	I	ST	Dijital giriş
		$\overline{\text{MCLR}}$	I	ST	Ana RESET girişi. Bu giriş cihazın resetlenmesi için aktif-0 yapılmalıdır.
		V _{PP}	P	-	Programlama voltajı girişi
V _{USB}	14	V _{USB}	P	-	Dâhili USB 3.3V voltaj regülatörü, dâhili USB alıcı/vericisi için pozitif besleme
V _{SS}	8, 9	V _{SS}	P	-	Lojik ve I/O pinleri için şase referansı
V _{DD}	20	V _{DD}	P	-	Lojik ve I/O pinleri için pozitif besleme

Açıklama:

O = Çıkış

CMOS = CMOS uyumlu giriş ya da çıkış

I = Giriş

TTL = TTL uyumlu giriş

P = Besleme

ST = CMOS seviyeli Schmitt Triggerli giriş

Not

1: CCP2MX biti temizlendiğinde CCP2 için alternatif atama

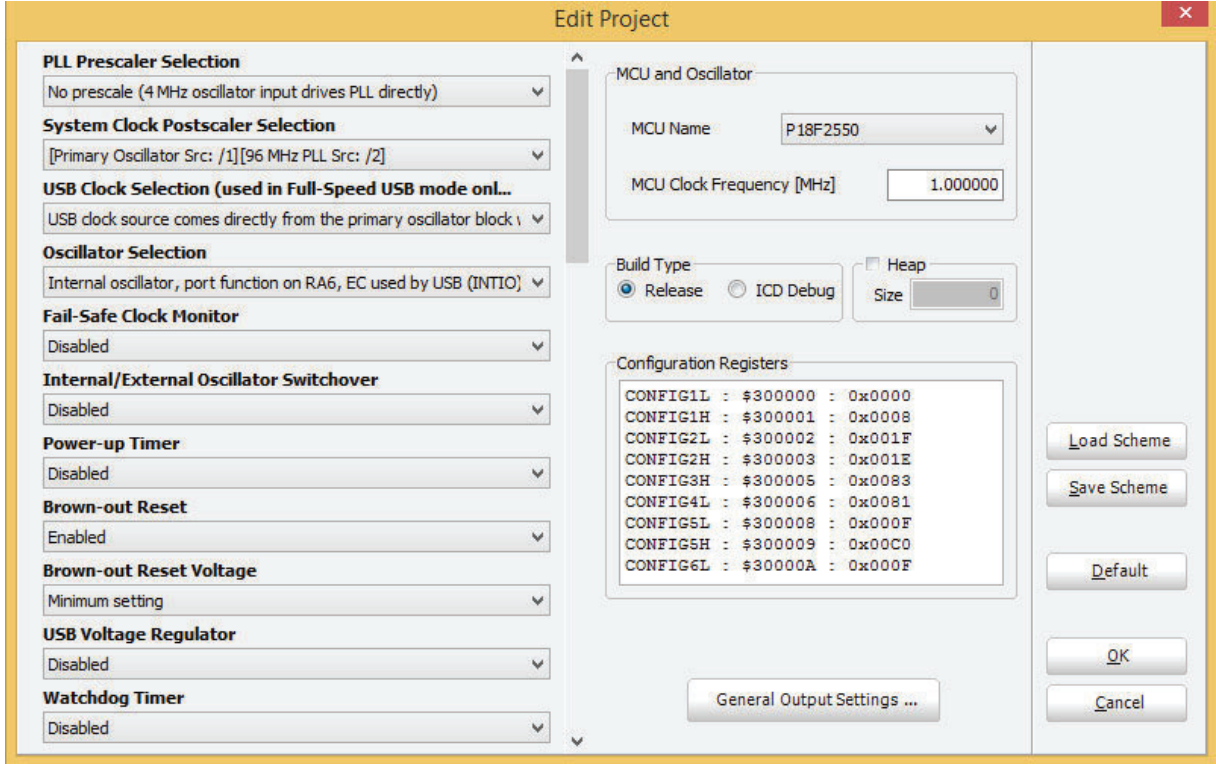
2: CCP2MX biti kurulduğunda CCP2 için varsayılan atama

F3-PIC18F2550 MİKRODENETLEYİCİSİ İÇİN “mikroC PRO for PIC” PROJE EDİTÖRÜ

PIC18F2455/2550/4455/4550 serisi mikrodenetleyicilerin yedi adet 16-bitlik yapılandırma sözcüğü vardır. Bu nedenle proje editörü penceresinde çok sayıda bit ayar hanesi bulunur. Pencerenin tek seferde görüntüleyebildiği alan yeterli gelmediğinden, yapılandırma sözcüklerinin bitlerine ait alan üç ayrı pencere şeklinde gösterilmiştir.

Yapılandırma bitleri program belleğinin 300000h adresinden başlar. Dikkat edileceği üzere normal koşullarda bu bellek alanı kullanıcı program belleği aralığının dışındadır. Gerçekte bu alan yapılandırma bellek alanı 300000h – 3FFFFFFh adresine aittir. Bu bölgeye ancak tablo okuma (TBLRD) ve tablo yazma (TBLWT) assembly komutlarıyla erişilir. Yapılandırma sözcüklerinin programlanması Flash Belleğin programlanmasına benzer bir yolla yapılır. MikroC proje düzenleyicisinde pencere üzerinden ilgili bitlere ait hücrelerin değiştirilmesi şeklinde yapılan ayarlamayı, program içinden kod yazarak yapmak istediğinizde assembly dili komutlarını ve EECON1 kaydedicisinin WR bitini kullanmanız gerekir.

Normal çalışma modunda, bir TBLWT komutu, yapılandırma kaydedicisini işaret eden TBLPTR komutuyla yapılandırma kaydedicisine yazılacak veriyi ve verinin yazılacağı adresi ayarlar. WR bitinin ayarlanmasıyla birlikte yapılandırma sözcüğüne uzun bir yazma işlemi gerçekleşir. Bir yapılandırma hücresinin silinmesi ya da o hücreye değer yazılması, ilgili hücreye TBLWT komutuyla '1' ya da '0' yazılmasıyla olur. Bilindiği üzere yapılandırma sözcüklerinin programlanması proje düzenleyicisi penceresi üzerinden yapılmaktadır. İlgili assembly komutlarını ve kaydedicilerini kullanarak, yazdığınız program içinden bu işi gerçekleştirmeyi sağlayan bir uygulama konunun sonunda verilmiştir.



Şekil F.2 PIC18F2550 proje editörü penceresi-I

CONFIG1L: YAPILANDIRMA SÖZCÜĞÜ 1 LSB KISMI

U-0	U-0	R/P-0	R/P-0	R/P-0	R/P-0	R/P-0	R/P-0
—	—	USBDIV	CPUDIV1	CPUDIV0	PLLDIV2	PLLDIV1	PLLDIV0
bit 7							bit 0

bit 7-6 **Kullanılmıyor:** '0' olarak okunur.

bit 5 **USBDIV:** USB saat darbesi seçim bitidir (yalnızca Full-Speed USB modunda kullanılır; UCFG:FSEN = 1).

1 = 96 MHz PLL'den gelen USB saat darbesi 2'ye bölünür.

0 = USB saat darbesi sonölçekleyici olmadan doğrudan birincil osilatörden gelir.

bit 4-3 **CPUDIV1:CPUDIV0:** Sistem saat darbesi sonölçekleyici seçim bitleridir.

XT, HS, EC ve ECIO osilatör modları için:

11 = Birincil osilatör sistem saat darbesinin üretilmesi için 4'e bölünür.

10 = Birincil osilatör sistem saat darbesinin üretilmesi için 3'e bölünür.

01 = Birincil osilatör sistem saat darbesinin üretilmesi için 2'ye bölünür.

00 = Birincil osilatör doğrudan sistem saat darbesi için kullanılır (sonölçekleyici yok).

XTPLL, HSPLL, ECPLL ve ECPIO osilatör modları için:

11 = Sistem saat darbesinin üretilmesi için 96 MHz PLL 6'ya bölünür.

10 = Sistem saat darbesinin üretilmesi için 96 MHz PLL 4'e bölünür.

01 = Sistem saat darbesinin üretilmesi için 96 MHz PLL 3'e bölünür.

00 = Sistem saat darbesinin üretilmesi için 96 MHz PLL 2'ye bölünür.

bit 2-0 **PLLDIV2:PLLDIV0:** PLL önölçekleyici seçim bitleridir.

111 = 12'ye bölünür (48 MHz osilatör girişi)

110 = 10'a bölünür (40 MHz osilatör girişi)

101 = 6'ya bölünür (24 MHz osilatör girişi)

100 = 5'e bölünür (20 MHz osilatör girişi)

011 = 4'e bölünür (16 MHz osilatör girişi)

010 = 3'e bölünür (12 MHz osilatör girişi)

001 = 2'ye bölünür (8 MHz osilatör girişi)

000 = Önölçekleyici yok (4 MHz osilatör girişi doğrudan PLL sinyalini üretir)

Not: PLL (Phase Locked Loop – Faz Kitlemeli Döngü) bir frekans sentezleme devresidir. Daha düşük frekanstan yüksek frekans üretmeyi sağlar. USB gibi yüksek veri iletimi gerektiren haberleşmelerde ihtiyaç duyulan yüksek frekans bu devre katı yardımıyla üretilir. Giriş osilatörünün frekans değerine göre yapılan ölçeklemeyle PLL devresinin girişi için gerekli 4 MHz’lik saat darbesi elde edilir. Ardından bu sinyalin 24 katı alınır ve 96 MHz’lik PLL saat darbesi elde edilir.

CONFIG1H: YAPILANDIRMA SÖZCÜĞÜ 1 MSB KISMI

R/P-0	R/P-0	U-0	U-0	R/P-0	R/P-1	R/P-0	R/P-1
IESO	FCMEN	—	—	FOSC3 ⁽¹⁾	FOSC2 ⁽¹⁾	FOSC1 ⁽¹⁾	FOSC0 ⁽¹⁾
bit 7							bit 0

bit 7 **IESO:** Dâhili/Harici osilatör anahtarlama bitidir.

1 = Osilatör anahtarlama modu etkinleştirildi.

0 = Osilatör anahtarlama modu kapalı.

bit 6 **FCMEN:** Güvenli saat darbesi izlemeyi etkinleştirme bitidir.

1 = Güvenli saat darbesi izlemesi etkinleştirildi.

0 = Güvenli-saat darbesi izlemesi kapatıldı.

bit 5-4 **Kullanılmıyor:** ‘0’ olarak okunur.

bit 3-0 **FOSC3:FOSC0:** Osilatör seçim bitleridir⁽¹⁾.

111x = HS osilatör, PLL etkin (HSPLL)

110x = HS osilatör (HS)

1011 = Dâhili osilatör, HS osilatörü USB tarafından kullanılır (INTHS)

1010 = Dâhili osilatör, XT USB tarafından kullanılır (INTXT)

1001 = Dâhili osilatör, RA6 pini CLKO işlevine sahiptir, EC USB tarafından kullanılır (INTCKO)

1000 = Dâhili osilatör, RA6 pini port işlevine sahiptir, EC USB tarafından kullanılır (INTIO)

0111 = EC osilatörü, PLL etkin, RA6 pini CLKO işlevine sahiptir (ECPLL)

0110 = EC osilatörü, PLL etkin, RA6 pini port işlevine sahiptir (ECPIO)

0101 = EC osilatörü, RA6 pini CLKO işlevine sahiptir (EC)

0100 = EC osilatörü, RA6 pini port işlevine sahiptir (ECIO)

001x = XT osilatörü, PLL etkin (XTPLL)

000x = XT osilatörü (XT)

Not 1: Mikrodenetleyici ve USB modülünün her ikisi de, XT, HS ve EC modlarında seçili osilatörü kendi saat kaynağı olarak kullanır. Mikrodenetleyici dâhili osilatörü kullandığında USB modülü işaret edilen XT, HS ya da EC osilatörünü kendi saat kaynağı olarak kullanır.

CONFIG2L: YAPILANDIRMA SÖZCÜĞÜ 2 LSB KISMI

U-0	U-0	R/P-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	VREGEN	BORV1 ⁽¹⁾	BORV0 ⁽¹⁾	BOREN1 ⁽²⁾	BOREN0 ⁽²⁾	PWRTEN ⁽²⁾
bit 7							bit 0

bit 7-6 **Kullanılmıyor:** '0' olarak okunur.

bit 5 **VREGEN:** USB dâhili voltaj regülatörü etkinleştirme bitidir.

1 = USB voltaj regülatörü etkin

0 = USB voltaj regülatörü kapalı

bit 4-3 **BORV1: BORV0:** Voltaj düşmesi reseti (Brown-out reset - BOR) için voltaj seviyesi bitleridir.

Min Typ. Maks.

11 = Minimum ayar 2,00 2,05 2,16 V

10 = Bir üst seviye 2,65 2,79 2,93 V

01 = Daha üst seviye 4,11 4,33 4,55 V

00 = Maksimum ayar 4,36 4,59 4,82 V

bit 2-1 **BOREN1: BOREN0:** Voltaj düşmesi etkinleştirme bitleridir.

11 = Voltaj düşmesi reseti yalnızca donanımsal olarak etkindir (SBOREN devre dışı).

10 = Voltaj düşmesi reseti yalnızca donanımsal olarak etkindir ve uyku modunda kapatılır (SBOREN devre dışı).

01 = Voltaj düşmesi reseti etkindir ve yazılım içinden kontrol edilir (SBOREN biti etkindir)

00 = Voltaj düşmesi reseti hem yazılımsal hem de donanımsal olarak kapatılır.

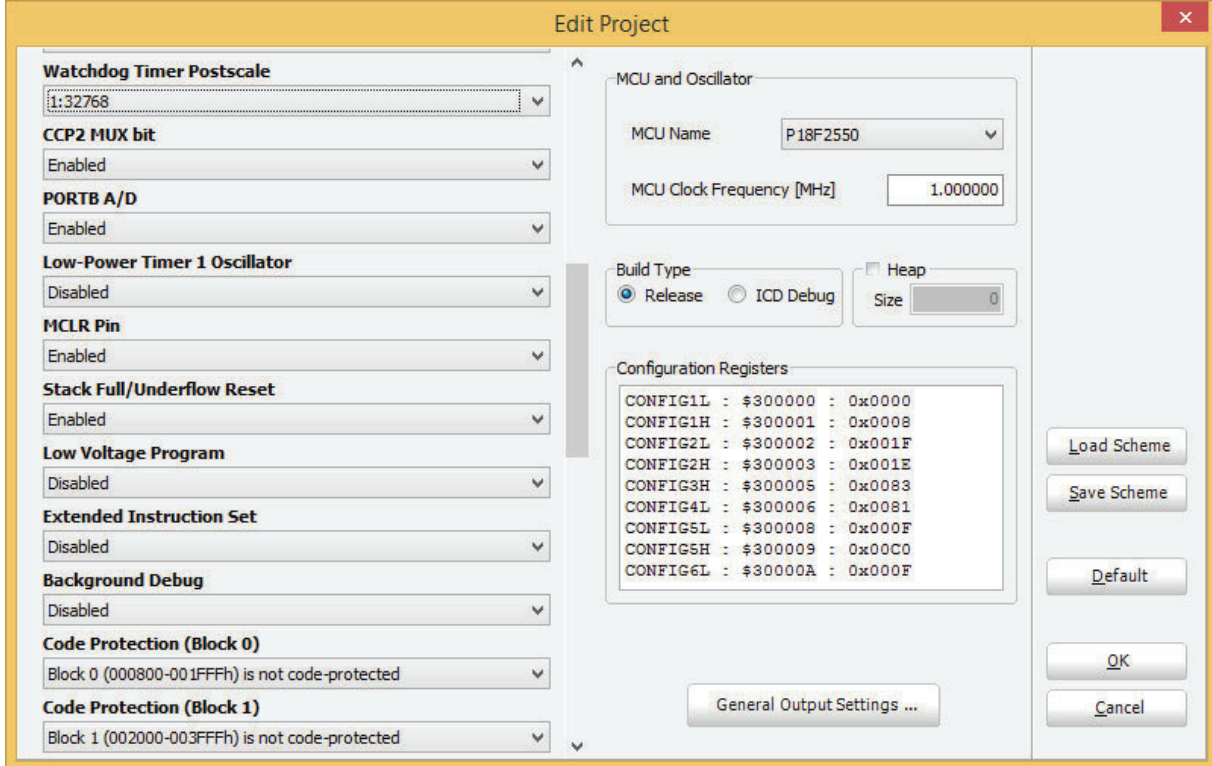
Not: PIC18F2550 voltaj kesmesi konusunda daha ayrıntılı bir seçenek sunar. RCON kaydedicisinin (Reset Control Register) SBOREN biti ile voltaj düşmesi kesmesinin yazılımsal olarak kontrolü sağlanır. Ancak SBOREN biti yalnızca CONFIG2L yapılandırma sözcüğünün BOREN1:BOREN0 bitleri '01' olduğunda etkili olur.

Eğer ilk besleme gecikmesi zamanlayıcısı (Power-up Timer - PWRT) etkinse, V_{DD} gerilimi VBOR seviyesinin üstüne çıktığında voltaj düşmesi reseti uyarılır. Ardından ek bir süre (TPWRT) kadar çipi reset konumunda tutmaya devam eder. PWRT çalışırken V_{DD} VBOR'un altına düşerse, çip bir voltaj düşmesi reseti durumuna geri döner ve PWRT başlatılır. V_{DD} 'nin VBOR'un üstüne çıkmasıyla birlikte PWRT ek bir zaman gecikmesi işletir. BOR ve PWRT, her ikisi de bağımsız olarak yapılandırılır. BOR resetini etkinleştirmek PWRT'yi otomatik olarak etkinleştirmez.

bit 0 **PWRTEN**: İlk besleme gecikmesi zamanlayıcısı etkinleştirme bitidir⁽²⁾.

1 = PWRT kapalı

0 = PWRT etkin



Şekil F.3 PIC18F2550 proje editörü penceresi-II

CONFIG2H: YAPILANDIRMA SÖZCÜĞÜ 2 MSB KISMI

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTEN
bit 7							bit 0

bit 7-5 **Kullanılmıyor**: '0' olarak okunur.

bit 4-1 **WDTPS3:WDTPS0**: Vardiya zamanlayıcısı sonölçekleyici seçim bitleridir.

1111 = 1:32,768

1110 = 1:16,384

1101 = 1:8,192

1100 = 1:4,096

1011 = 1:2,048

1010 = 1:1,024

1001 = 1:512

1000 = 1:256

0111 = 1:128

0110 = 1:64

0101 = 1:32

0100 = 1:16
 0011 = 1:8
 0010 = 1:4
 0001 = 1:2
 0000 = 1:1

bit 0 **WDTEN:** Vardiya zamanlayıcısı etkinleştirme bitidir.

1 = WDT etkin

0 = WDT kapalı (kontrol, WDTCN kaydedicisinin SWDTEN bitiyle sağlanır)

CONFIG3H: YAPILANDIRMA SÖZCÜĞÜ 3 MSB KISMI

R/P-1	U-0	U-0	U-0	U-0	R/P-0	R/P-1	R/P-1
MCLRE	—	—	—	—	LPT1OSC	PBADEN	CCP2MX
bit 7							bit 0

bit 7 **MCLRE:** MCLR pini etkinleştirme bitidir.

1 = MCLR pini etkin, RE3 giriş pini kapalı

0 = RE3 giriş pini etkin, MCLR pini kapalı

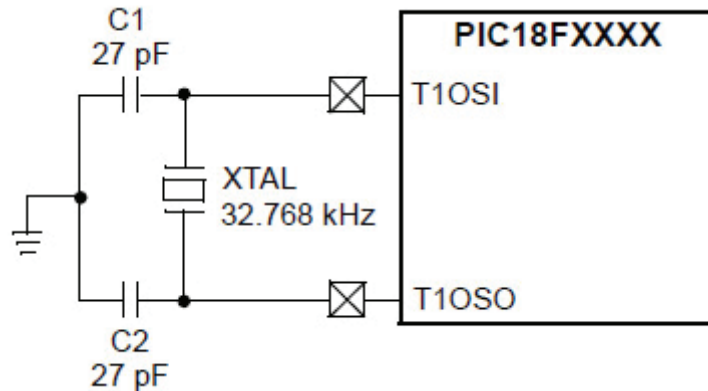
bit 6-3 **Kullanılmıyor:** '0' olarak okunur.

bit 2 **LPT1OSC:** Düşük-güç Timer1 osilatörü etkinleştirme bitidir.

1 = Timer1 düşük-güç ile çalışacak şekilde yapılandırılır.

0 = Timer1 daha yüksek güç ile çalışacak şekilde yapılandırılır.

Not: Timer1 zamanlayıcısı için varsayılan çalışma modu yüksek güç çalışmasıdır. Düşük güç tüketiminin istendiği durumlarda (uyku modunda alarm devresi gibi) Şekil F.4'teki gibi LP osilatörü bağlantısı yapılabilir. Burda dikkat edilmesi gereken mesele, düşük güç modunda osilatörün gürültülere ve girişime karşı daha duyarlı olmasıdır. Bu tür uygulamalar, düşük gürültülü ve güç tasarrufunun önemli olduğu uygulamalarda en iyi sonucu verir.



Şekil F.4 Düşük güç (Low Power - LP) osilatörü

bit 1 **PBADEN:** PORTB A/D etkinleştirme bitidir.

(ADCON1 reset durumunu etkiler. ADCON1 PORTB<4:0> pin yapılandırmasını kontrol eder.)

1 = PORTB<4:0> pinleri reset durumunda analog giriş olarak yapılandırılır.

0 = PORTB<4:0> pinleri reset durumunda dijital I/O olarak yapılandırılır.

bit 0 **CCP2MX:** CCP2 MUX bitidir.

1 = CCP2 giriş/çıkış RC1'e çoklanır.

0 = CCP2 giriş/çıkış RB3'e çoklanır.

CONFIG4L: YAPILANDIRMA SÖZCÜĞÜ 4 LSB KISMI

R/P-1	R/P-0	R/P-0	U-0	U-0	R/P-1	U-0	R/P-1
DEBUG	XINST	ICPRT ⁽¹⁾	—	—	LVP	—	STVREN
bit 7							bit 0

bit 7 **DEBUG:** Arkazemin hata ayıklayıcısı etkinleştirme bitidir.

1 = Arkazemin hata ayıklayıcısı kapalıdır, RB6 ve RB7 genel amaçlı I/O pini olarak yapılandırılır.

0 = Arkazemin hata ayıklayıcısı etkinleştirilir, RB6 ve RB7 devre içi hata ayıklayıcısı için tahsis edilir.

bit 6 **XINST:** Genişletilmiş komut seti etkinleştirme bitidir.

1 = Genişletilmiş komut seti ve indisli adresleme modu etkinleştirilir.

0 = Genişletilmiş komut seti ve indisli adresleme modu kapatılır (eski mod)

bit 5 **ICPRT:** Devre içi hata ayıklama/programlama portu (ICPORT) etkinleştirme bitidir⁽¹⁾.

1 = ICPORT etkin

0 = ICPORT kapalı

bit 4-3 **Kullanılmıyor:** '0' olarak okunur.

bit 2 **LVP:** Tek-kaynak (Single-Supply) ICSP™ etkinleştirme bitidir.

1 = Tek kaynak ICSP etkin

0 = Tek-kaynak ICSP kapalı

Not: LVP yapılandırma biti eski adıyla düşük voltaj ICSP programlamayı etkinleştirir. Tek-kaynak programlama etkinleştirildiğinde, mikrodenetleyici MCLR/VPP/RE3 pinine yüksek voltaj uygulanmasına ihtiyaç duymadan programlanabilir. Fakat RB5/KBI1/PGM pini program modu girişini kontrol etmeye tahsis edilir ve genel amaçlı I/O pini özelliği devre dışı kalır. Tek-

kaynak programlamayı kullanarak programlama sırasında, MCLR/VPP/RE3 pinine V_{DD} normal çalışma modunda uygulanır. Programlama moduna girmek için, PGM pinine de V_{DD} uygulanır.

bit 1 Kullanılmıyor: '0' olarak okunur

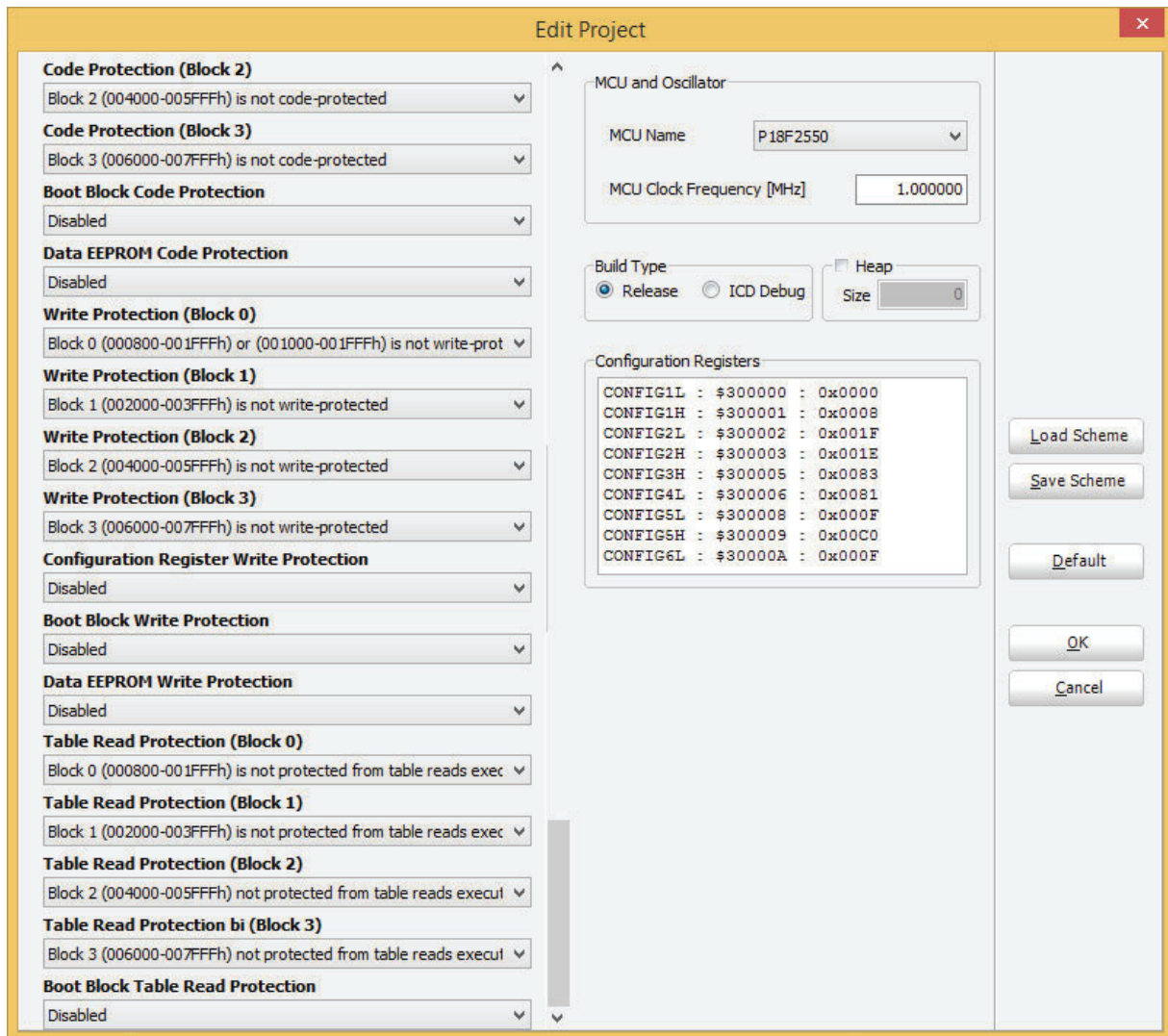
bit 0 **STVREN:** Stack Full/Underflow Reset etkinleştirme bitidir.

1 = Yiğın işaretçisi taşma/yetersiz olma durumu resete neden olacaktır.

0 = Yiğın işaretçisi taşma/yetersiz olma durumu resete neden olmaz.

Not: Yiğın (Stack) alanı, işlemci tarafından verilerin geçici olarak saklandığı veya uygulamanın kullandığı değişkenlerin tutulduğu ve büyüklüğü işletim sistemine göre değişen bellek bölgesidir. Zaman içinde bu geçici hafıza alanı program işleyişine bağlı olarak dolar ya da yetersiz gelir. Bu durumda cihaza reset işlemi yaptırılarak yiğın bölgesinin temizlenmesi sağlanabilir.

Not 1: Bu özellik yalnızca 44-pin TQFP paketlerde vardır. Diğer tüm aygıtlar için '0' olarak tutulmalıdır.



Şekil F.5 PIC18F2550 proje editörü penceresi-III

CONFIG5L: YAPILANDIRMA SÖZCÜĞÜ 5 LSB KISMI

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	CP3 ⁽¹⁾	CP2	CP1	CP0
bit 7							bit 0

bit 7-4 **Kullanılmıyor:** '0' olarak okunur.

bit 3 **CP3:** Kod koruma bitidir⁽¹⁾.

1 = Blok 3 (006000-007FFFh) kod korumalı değildir.

0 = Blok 3 (006000-007FFFh) kod korumalıdır.

bit 2 **CP2:** Kod koruma bitidir.

1 = Blok 2 (004000-005FFFh) kod korumalı değildir.

0 = Blok 2 (004000-005FFFh) kod korumalıdır.

bit 1 **CP1:** Kod koruma bitidir.

1 = Blok 1 (002000-003FFFh) kod korumalı değildir.

0 = Blok 1 (002000-003FFFh) kod korumalıdır.

bit 0 **CP0:** Kod koruma bitidir.

1 = Blok 0 (000800-001FFFh) kod korumalı değildir.

0 = Blok 0 (000800-001FFFh) kod korumalıdır.

Not 1: PIC18FX455 mikrodenetleyicilerinde kullanılmaz; bu biti kurulu olarak bırakın.

CONFIG5H: YAPILANDIRMA SÖZCÜĞÜ 5 MSB KISMI

R/C-1	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
CPD	CPB	—	—	—	—	—	—
bit 7							bit 0

bit 7 **CPD:** Veri EEPROM'u kod koruma bitidir.

1 = Veri EEPROM'u kod korumalı değildir.

0 = Veri EEPROM'u kod korumalıdır.

bit 6 **CPB:** Açılış bloğu kod koruma bitidir.

1 = Açılış bloğu (000000-0007FFh) kod korumalı değildir.

0 = Açılış bloğu (000000-0007FFh) kod korumalıdır.

bit 5-0 **Kullanılmıyor:** '0' olarak okunur.

CONFIG6L: YAPILANDIRMA SÖZCÜĞÜ 6 LSB KISMI

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	WRT3 ⁽¹⁾	WRT2	WRT1	WRT0
bit 7							bit 0

bit 7-4 **Kullanılmıyor:** '0' olarak okunur.

bit 3 **WRT3:** Yazma koruma bitidir⁽¹⁾.

1 = Blok 3 (006000-007FFFh) yazma korumalı değildir.

0 = Blok 3 (006000-007FFFh) yazma korumalıdır.

bit 2 **WRT2:** Yazma koruma bitidir.

1 = Blok 2 (004000-005FFFh) yazma korumalı değildir.

0 = Blok 2 (004000-005FFFh) yazma korumalıdır.

bit 1 **WRT1:** Yazma koruma bitidir.

1 = Blok 1 (002000-003FFFh) yazma korumalı değildir.

0 = Blok 1 (002000-003FFFh) yazma korumalıdır.

bit 0 **WRT0:** Yazma koruma bitidir.

1 = Blok 0 (000800-001FFFh) ya da (001000-001FFFh) yazma korumalı değildir.

0 = Blok 0 (000800-001FFFh) ya da (001000-001FFFh) yazma korumalıdır.

Not 1: PIC18FX455 mikrodenetleyicilerinde kullanılmaz; bu biti kurulu olarak bırakın.

CONFIG6H: YAPILANDIRMA SÖZCÜĞÜ 6 MSB KISMI

R/C-1	R/C-1	R-1	U-0	U-0	U-0	U-0	U-0
WRTD	WRTB	WRTC ⁽¹⁾	—	—	—	—	—
bit 7							bit 0

bit 7 **WRTD:** Veri EEPROM'u yazma koruma bitidir.

1 = Veri EEPROM'u yazma korumalı değildir.

0 = Veri EEPROM'u yazma korumalıdır.

bit 6 **WRTB:** Açılış bloğu yazma koruma bitidir.

1 = Açılış bloğu (000000-0007FFh) yazma korumalı değildir.

0 = Açılış bloğu (000000-0007FFh) yazma korumalıdır.

bit 5 **WRTC:** Yapılandırma kaydedicisi yazma koruma bitidir⁽¹⁾.

1 = Yapılandırma kaydedicileri (300000-3000FFh) yazma korumalı değildir.

0 = Yapılandırma kaydedicileri (300000-3000FFh) yazma korumalıdır.

bit 4-0 **Kullanılmıyor:** '0' olarak okunur.

Not 1: Bu bit normal çalışma modunda salt okunurdur; Yalnızca programlama modundayken yazılabilir.

CONFIG7L: YAPILANDIRMA SÖZCÜĞÜ 7 LSB KISMI

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	EBTR3 ⁽¹⁾	EBTR2	EBTR1	EBTR0
bit 7							bit 0

bit 7-4 **Kullanılmıyor:** '0' olarak okunur.

bit 3 **EBTR3:** Tablo okuma koruma bitidir⁽¹⁾.

1 = Blok 3 (006000-007FFFh) bölgesi diğer bloklarda işletilen tablo okumalarından korunmaz.

0 = Blok 3 (006000-007FFFh) bölgesi diğer bloklarda işletilen tablo okumalarından korunur.

bit 2 **EBTR2:** Tablo okuma koruma bitidir.

1 = Blok 2 (004000-005FFFh) diğer bloklarda işletilen tablo okumalarından korunmaz.

0 = Blok 2 (004000-005FFFh) diğer bloklarda işletilen tablo okumalarından korunur.

bit 1 **EBTR1:** Tablo okuma koruma bitidir.

1 = Blok 1 (002000-003FFFh) diğer bloklarda işletilen tablo okumalarından korunmaz.

0 = Blok 1 (002000-003FFFh) diğer bloklarda işletilen tablo okumalarından korunur.

bit 0 **EBTR0:** Tablo okuma koruma bitidir.

1 = Blok 0 (000800-001FFFh) diğer bloklarda işletilen tablo okumalarından korunmaz.

0 = Blok 0 (000800-001FFFh) diğer bloklarda işletilen tablo okumalarından korunur.

Not 1: PIC18FX455 mikrodenetleyicilerinde kullanılmaz; bu biti kurulu olarak bırakın.

CONFIG7H: YAPILANDIRMA SÖZCÜĞÜ 7 MSB KISMI

U-0	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
—	EBTRB	—	—	—	—	—	—
bit 7							bit 0

bit 7 **Kullanılmıyor:** '0' olarak okunur.

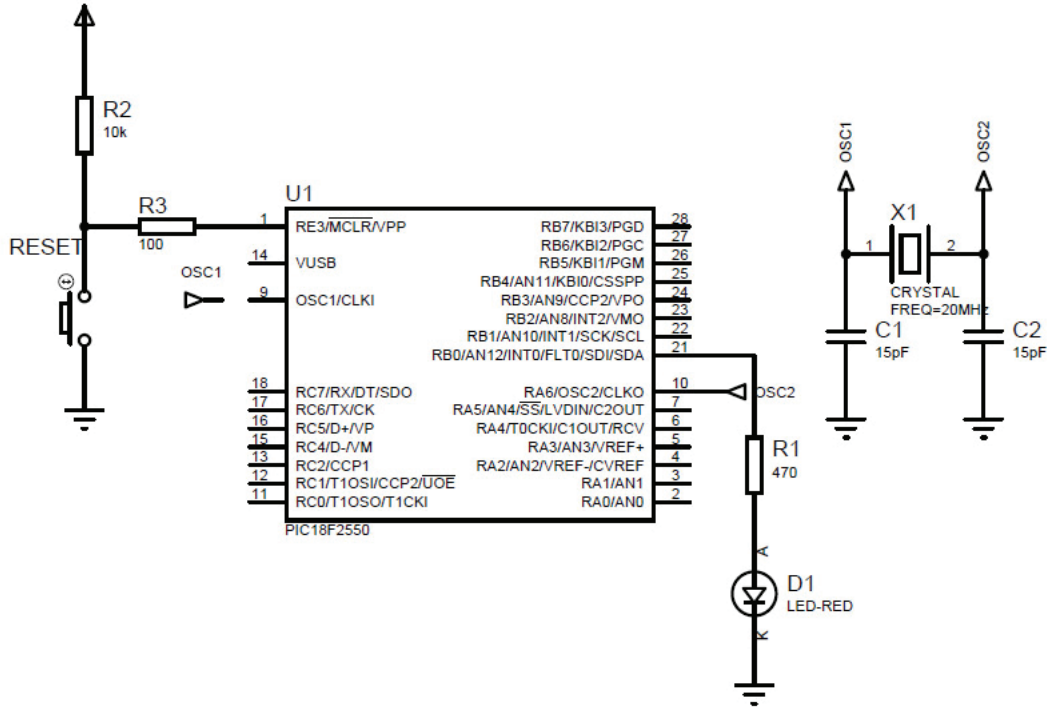
bit 6 **EBTRB:** Açılış bloğu tablo okuma koruma bitidir.

1 = Açılış bloğu (000000-0007FFFh) diğer bloklarda işletilen tablo okumalarından korumalı değildir.

0 = Açılış bloğu (000000-0007FFh) diğer bloklarda işletilen tablo okumalarından korumalıdır.

bit 5-0 Kullanılmıyor: '0' olarak okunur.

Önceki bölümlerde de söylendiği gibi, yapılandırma sözcükleri mikroC program kullanıcı arayüzünde proje düzenleyicisi penceresinden ayarlanmaktadır. Ancak duruma göre bu sözcüklerin yazılım içinden ayarlanması istenebilir. Bunun için EEPROM ve Flash program belleği yazma kaydedicilerinin kullanımı gerekmektedir. Şekil F.6'da yazılımsal olarak yapılandırma sözcüklerinin programlanması uygulaması gösterilmiştir. Uygulama devresinde yapılmak istenen işlem sonucunun hızlı ve pratik şekilde izlenmesi için LED kullanılmıştır. Uygulamada vardiya zamanlayıcısının reset süresi değiştirilmektedir. Böylece mikrodenetleyicinin uyku modundan çıkıp ikaz LED'inin farklı sürelerde fasıllı olarak yanması sağlanmaktadır. Yapılan ölçümlerde uyku modu sırasında mikrodenetleyicinin 30uA kadar akım tükettiği gözlemlenmiştir. Yapılandırma sözcüklerinin adres alanı Flash program belleği bölgesinde bulunmaktadır ve mikroC programının bellek yazma kütüphanesi EEPROM veri belleğine yazma işlemi yapabilir. Bu nedenle yapılandırma sözcüklerinin yazılım içinden programlanabilmesi için kaydedicilerin kullanılması gerekmektedir.



Şekil F.6 PIC18F2550'nin yazılımsal olarak yapılandırma sözcüklerinin ayarlanması

Devrenin gerçek uygulamasında 20 MHz'lik kristal kullanılmış ve osilatör HS moduna ayarlanmıştır. CONFIG1H yapılandırma sözcüğü '0x000C' değerine kurulmuştur. Devrenin gerçek pratik uygulamasından elde edilen sonuçla PROTEUS-ISIS® uygulamasından elde edilen sonuç aynı değildir. PROTEUS-ISIS® programı vardiya zamanlayıcısıyla ilgili uygulamaları ve yapılandırma sözcüklerinin yazılımsal olarak değiştirilmesi uygulamalarını tam olarak canlandıramamaktadır. Ayrıca söz konusu uygulamada sürekli olarak vardiya zamanlayıcısının süresi değiştirildiğinden ve mikrodenetleyicinin

Flash program belleği bölgesine yazma işlemi yapıldığından donanım ömrü göz önünde bulundurulmalıdır.

Flash program belleği bilindiği gibi en fazla 100.000 yazma/silme ömrüne sahiptir. Dolayısıyla böyle bir uygulamanın özelliği kullanıcıya yazılım yoluyla yapılandırma sözcüklerine ve Flash program belleğine müdahale etme yöntemini göstermektir. Gerçek bir uygulamada WDT süresi mümkün olduğunca yazılım içinden değiştirilmemeli ya da ihtiyaca göre reset süresi uzun tutulmalıdır.

Tablo F.2 PIC18F2550'nin yazılımsal olarak yapılandırma sözcüklerinin ayarlanması mikroC kodu

```
short i, WDT_degeri;
void write_WDT(){
    TABLAT=WDT_degeri; //EEPROMUN yapılandırma alanına yazılacak veriyi alan
    kaydedici
    asm{
        CLRWDT //Vardiya zamanlayıcısı resetleniyor
    }
    WDTCON.SWDTEN=0;
    TBLPTRU = 0x30; //0x300003 CONFIG2H adresinin 30 kısmı
    TBLPTRH = 0X00; //0x300003 CONFIG2H adresinin 00 kısmı
    TBLPTRL = 0X03; //0x300003 CONFIG2H adresinin 03 kısmı
    asm{
        TBLWT* /* 1'er byte artarak giden adres değerlerine işaretleme yapıp
        tablo değerini yazdıran assembly komutu */
    }
    EECON1 = 0b11000100;
    STATUS.C = 0; // Taşma olmadı
    if(INTCON.GIE){
        STATUS.C = 1;
    }
    INTCON.GIE=0;
    EECON2 = 0X55;
    EECON2 = 0XAA;
    EECON1.WR = 1;
    if(STATUS.C){
        INTCON.GIE=1;
    }
    EECON1.WREN=0;
    WDTCON.SWDTEN=1;
}

void blink_LED(){
    for(i=0;i<10;i++){
        asm{
            sleep
        }
        LATB = ~LATB; //Toggle RB0
    }
}

void main() {
    ADCON0=0;
    ADCON1=0;
    CMCON=0X07;
    LATB = 0;
    TRISB = 0;
    WDTCON.SWDTEN=1;
    while(1){
        WDT_degeri = 12; // 0b00001100, WDT sonölçekleyicisi 1:64, yaklaşık 1/4
        saniye
        write_WDT();
        blink_LED();
    }
}
```

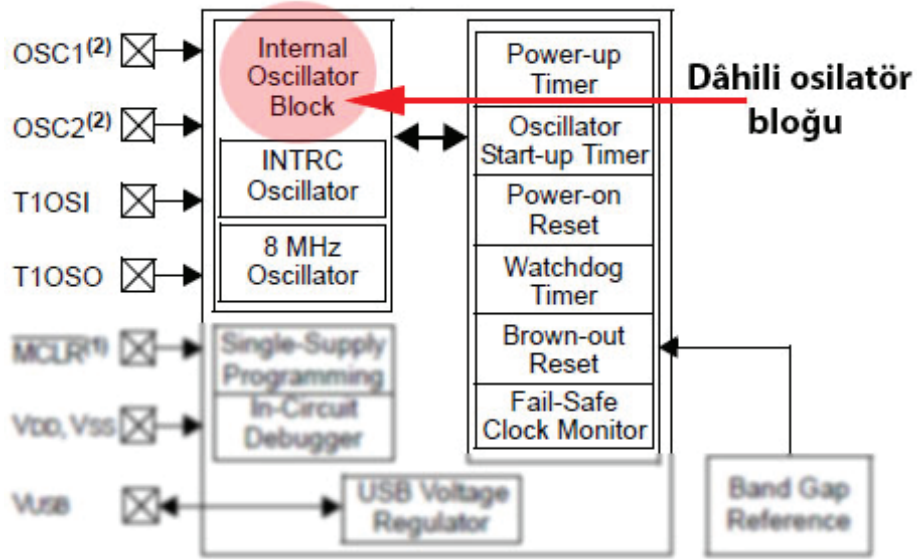
```

WDT_degeri = 16; // 0b00010000, WDT sonölçekleyicisi 1:256, yaklaşık 1
saniye
write_WDT();
blink_LED();
}}

```

F4-PIC18F2550/4550 MİKRODENETLEYİCİSİ İÇİN OSCCON OSİLATÖR KONTROL KAYDEDİCİSİNİN KONTROLÜ VE TEMEL GÜÇ YÖNETİMİ

PIC18F2455/2550/4455/4550 ailesi gelişmiş bir osilatör yönetimine sahiptir. Bu tür gelişmiş mikrodenetleyiciler cihaz saat darbesi kaynağının ana osilatör kaynağından bir başka alternatif anahtarlanmasını sağlayacak özelliktedir. Bu cihazlar iki alternatif saat darbesi kaynağı sunar. Alternatif saat darbesi kaynağı etkinleştirildiğinde farklı güç yönetim modları devreye alınır. Şekil F.7’de bu mikrodenetleyiciler için ortak olan osilatör blokları basitleştirilmiş diyagramı gösterilmiştir. Dikkat edileceği üzere dâhili osilatör bloğu (Internal Oscillator Block), INTRC (dâhili RC) osilatörü ve 8 MHz osilatöründen oluşmaktadır.



Şekil F.7 PIC18F2455/2550/4455/4550 ailesinde osilatör blokları

Esasında, bu mikrodenetleyiciler için üç ayrı saat darbesi kaynağı vardır:

- 1- Birincil osilatörler
- 2- İkincil osilatörler
- 3- Dâhili osilatör bloğu

Birincil osilatörler, harici kristal ve rezonatör modlarını, harici saat kaynağı modlarını ve dâhili osilatör bloğunu içerir. Belirli modlar bir önceki konu başlığında anlatılmış olan CONFIG1H yapılandırma sözcüğünün **FOSC3:FOSC0** osilatör bitleri ile seçilmektedir.

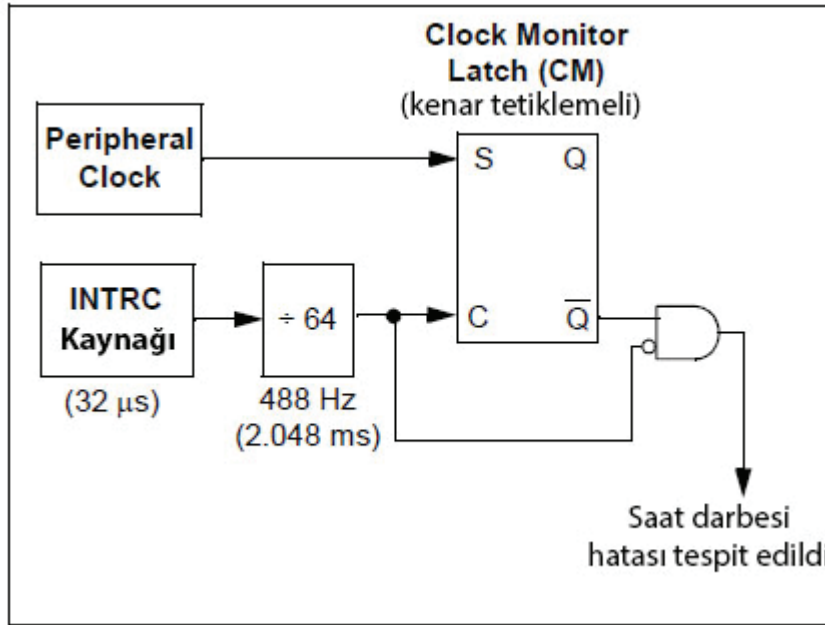
İkincil osilatörler, OSC1 ya da OSC2 osilatör kaynağı girişlerine bağlanmayan osilatörleri ifade eder. Bu osilatörler mikrodenetleyici bir güç yönetimi moduna alınsa bile çalışmaya devam eder. PIC18F2455/2550/4455/4550 cihazları Timer1 osilatörünü ikincil bir osilatör olarak değerlendirir. Bu osilatör, tüm güç yönetimi modlarında, çoğunlukla da RTC (RealTime Clock – Gerçek Zaman Saati) gibi zaman temelli uygulamalarda temel zaman kaynağı olur. Çoğu zaman, RC0/T1OSO/T13CKI ve

RC1/T1OSI/UOE pinlerine bir 32.768 kHz saat kristali bağlanır. XT ve HS osilatör modu devrelerinde olduğu gibi şase ve her iki pin arasına yükleme kapasitörleri de bağlanır.

Birincil saat darbesi kaynağı olmasının yanında, “dâhili osilatör bloğu” bir güç yönetim modu saat darbesi kaynağı olarak da görev yapar. INTRC osilatörü, WDT ve FSCM (Fail-Safe Clock Monitor) gibi bazı spesifik özellikler için de saat darbesi kaynağı olarak kullanılır.

18F serisi gelişmiş mikrodenetleyicilerde FSCM gibi güvenli osilatör kaynağı izleme teknolojisi bulunur. Bu özellik CONFIG1H yapılandırma sözcüğünün 6.biti FCMEN etkinleştirilerek devreye alınır. Bir hata tespit edildiğinde:

- FSCM modülü PIR2 kaydedicisinin 7.biti olan OSCFIF’i etkinleştirir,
- Cihaz saat darbesi kaynağı dâhili osilatör bloğuna anahtarlanır (ancak OSCCON kaydedicisi güncellenmez),
- WDT resetlenir.



Şekil F.8 FSCM bloğunun çalışma ilkesi

Osilatör kontrol kaydedicisi olan OSCCON ile cihaz saat darbesi kaynağı birkaç yönden kontrol edilir.

OSCCON: OSİLATÖR KONTROL KAYDEDİCİSİ (ADRES: FD3h – PIC18F2455/2550/4455/4550)

R/W-0	R/W-1	R/W-0	R/W-0	R ⁽¹⁾	R-0	R/W-0	R/W-0
IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0
bit 7							bit 0

bit 7 IDLEN: Boşta durma modunu etkinleştirme bitidir.

1 = SLEEP komutuyla birlikte cihaz boşta durma (idle) moduna girer.

0 = SLEEP komutuyla birlikte cihaz uyku moduna girer.

bit 6-4 IRCF<2:0>: Dâhili osilatör tercih bitleridir.

111 = 8 MHz (INTOSC saat darbesini doğrudan sürer. Diğer bir ifadeyle frekans bölme/çarpma gibi bir işlem yapılmaz. Çünkü INTOSC osilatörü temel frekansı 8 MHz'dir.)

110 = 4 MHz

101 = 2 MHz

100 = 1 MHz (Cihaz resetlendiğinde varsayılan INTOSC frekansı 1 MHz'dir.)

011 = 500 kHz

010 = 250 kHz

001 = 125 kHz

000 = 31 kHz⁽²⁾

bit 3 **OSTS:** Osilatör açılış zaman aşımı durum bitidir⁽¹⁾.

1 = Osilatör açılış zamanlayıcısı zaman aşımı gerçekleşti ve bitti; birincil osilatör koşturuluyor.

0 = Osilatör açılış zamanlayıcısı zaman aşımı devam ediyor; birincil osilatör hazır değil.

bit 2 **IOFS:** INTOSC frekansı kararlılık bitidir.

1 = INTOSC kararlı/dengeli

0 = INTOSC kararlı/dengeli değil

bit 1-0 **SCS1:SCS0:** Sistem saat darbesi seçme bitidir.

1x = Dahili osilatör

01 = Timer1 osilatörü

00 = Birincil osilatör

Not 1: CONFIG1H 7.biti olan IESO'nun durumuna bağlıdır. Eğer bu bit etkinleştirilmemişse zaman aşımı dikkate alınmaz. Osilatör frekansının dengeye oturması için geçen o ilk süre hassas iletişim yapılan durumlarda önemlidir. Bu durumda IESO biti etkinleştirilir ve OSTS biti 1 olana kadar iletişim başlatılmaz. Böyle bir işlem programın ana fonksiyonu içinde aşağıdaki kodla sağlanabilir.

```
while(!OSCCON.OSTS) {  
    Delay_us(1);  
}
```

2: Bu değer iki şekilde elde edilebilir; ya INTOSC temel frekansı 256 ile bölünerek ya da doğrudan dahili RC – INTRC osilatörü ile. Bu işlem OSC_TUNE kaydedicisinin 7.biti olan INTSRC ile yapılır. Ancak bu bitin durumundan bağımsız olarak INTRC osilatörü her zaman WDT ve FSCM modülü için ana frekans kaynağıdır.

Sistem saat darbesi kaynağı SCS1:SCS0 bitleri ile seçilir. Mevcut saat darbesi kaynakları, birincil kaynak (FOSC3:FOSC0 yapılandırma bitleriyle tercih edilen), ikincil kaynak (Timer1 osilatörü) ve dahili osilatör bloğudur. Saat darbesi kaynağı, bu bitlerde bir değişim olduğu gibi hemen değişir. Bu geçiş sırasında kısa süreli bir geçiş aralığı olur. Cihaz resetlendiğinde SCS bitleri temizlenir.

OSTS, IOFS ve T1RUN bitleri, mikrodenetleyicinin saat darbesini hangi saat kaynağının sağladığına işaret eder. OSTS biti, OST (Oscillator Start-up Timer) devresinin zaman aşımını geçtiğini ve birincil saat darbesi modlarında birincil saat darbesinin cihazı sürdürdüğünü işaret eder. IOFS biti dahili osilatör bloğunun kararlı olduğunu ve dahili RC modlarından biriyle cihazı sürdürdüğünü işaret eder. T1CON kaydedicisinin T1RUN biti ise Timer1 osilatörünün ikincil saat modunda cihazı sürdürdüğünü gösterir. Güç

yönetimi modlarında, herhangi bir zamanda bu üç bittten yalnızca biri kurulur. Eğer bu üç bittten hiçbiri kurulu değilse, cihaz için gerekli saat darbesini INTRC sürüyor ya da dahili osilatör bloğu henüz işlemeye başlamış ve dengeye oturmamış demektir.

Bu mikrodenetleyici ailesi, INTOSC dahili osilatörü 8 MHz'e kalibre edilmiş olarak satılır. Ancak ortam sıcaklığı ve çalışma voltajı (V_{DD}) üzerindeki dalgalanmalar ya da seviye değişikliği bu frekans değerini değiştirebilir. Dolayısıyla fabrika kalibrasyonu ile gelen INTOSC modülünün merkez frekansında yukarı-aşağı yönlü ufak adımlarla ayar yapmak için OSCTUNE kaydedicisi sunulmuştur. Her kademe için ayar hassasiyeti eşit değerdedir. Yapılan ayar yüksek hassasiyetli osiloskop cihazlarıyla OSC2/CLKO pini üzerinden alınan $F_{OSC}/4$ frekansının izlenmesiyle görülebilir. Ayar yapıldığında INTOSC saat darbesi 1 msn içinde kararlı hale gelecektir. Bu sırada program kodu işlemeye devam eder.

OSCTUNE: OSİLATÖR İNCE AYAR KAYDEDİCİSİ (ADRES: F9Bh – PIC18F2455/2550/4455/4550)

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INTSRC	—	—	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

bit 7 **INTSRC:** Dahili osilatör düşük frekans kaynağını seçme bitidir.

1 = 31.25 kHz'lik cihaz frekansı 8 MHz'lik INTOSC kaynağının 256'ya bölünmesiyle elde edilir.

0 = 31 kHz'lik cihaz frekansı doğrudan INTRC dahili osilatöründen elde edilir.

bit 6-5 **Kullanılmıyor:** 0 olarak okunur.

bit 4-0 **TUN4:TUN0:** Frekans ince ayar bitleridir.

01111 = Maksimum frekans

• •

00001

00000 = Merkez frekansıdır. Osilatör modülü kalibre edilmiş değerde koşturulmaktadır.

• •

• •

10000 = Minimum frekans

Tablo F.3'te tüm osilatör çalışma anlarının ve güç durumlarının birbirleriyle ilişkileri özetlenmiştir.

Tablo F.3 GÜÇ YÖNETİMİ MODLARI

Mod	OSCCON<7, 1:0>		Modüle Saat Darbesi Uygulanması Durumu		Mevcut Saat Darbesi ve Osilatör Kaynağı
	IDLEN	SCS1:SCS0	CPU	Çevresel Donanımlar	
Sleep	0	N/A	Kapalı	Kapalı	Hiçbiri – tüm osilatörler kapalı
PRI_RUN	N/A	00	Saat darbeleri	Saat darbeleri	Birincil – tüm osilatör modları Bu normal tam güçte çalışma modudur.
SEC_RUN	N/A	01	Saat darbeleri	Saat darbeleri	İkincil – Timer1 osilatörü
RC_RUN	N/A	1X	Saat darbeleri	Saat darbeleri	Dahili osilatör bloğu (INTOSC ve INTRC)
PRI_IDLE	1	00	Kapalı	Saat darbeleri	Birincil – tüm osilatör modları
SEC_IDLE	1	01	Kapalı	Saat darbeleri	İkincil – Timer1 osilatörü

RC_IDLE	1	1X	Kapalı	Saat darbeleri	Dahili osilatör bloęu (INTOSC ve INTRC)
---------	---	----	--------	----------------	---

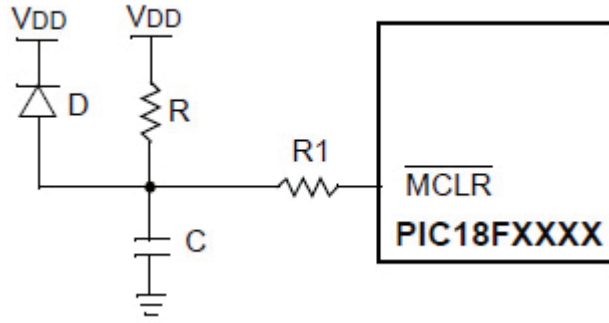
F5-PIC18F2550/4550 İÇİN TEMEL ELEKTRİKSEL ÖZELLİKLER

PIC18F2550'den en yüksek verimi almak için çalışma koşullarına göre elektriksel verileri göz önünde bulundurmanız gerekir. Aşağıda mikrodenetleyicinin ön görülen üst sınır değerleri listelenmiştir. Mikrodenetleyicinizin uzunca bir süre bu değerlerin üzerinde çalıştırılması cihazınıza ve sisteminize kalıcı zararlar verebilir.

Besleme altındaki ortam sıcaklığı	-40° – +85°C
Saklanma koşulları altındaki sıcaklık.....	-65°C – +150°C
Toplam güç tüketimi ⁽¹⁾	1 W
V _{SS} 'ye göre V _{DD} üzerindeki voltaj	-0.3V – +7,5V
V _{SS} 'ye göre $\overline{\text{MCLR}}$ üzerindeki voltaj ⁽²⁾	0V – +13,25V
V _{SS} 'ye göre tüm diğer pinlerdeki voltaj (V _{DD} ve $\overline{\text{MCLR}}$ hariç) ⁽³⁾	-0,3V – (V _{DD} + 0,3V)
V _{SS} pininden geçen en yüksek akım.....	300 mA
V _{DD} pinine giren en yüksek akım.....	250 mA
Giriş kenetleme akımı, I _{IK} (V _I < 0 ya da V _I > V _{DD}).....	± 20 mA
Çıkış kenetleme akımı, I _{OK} (V _O < 0 ya da V _O > V _{DD})	± 20 mA
I/O pini tarafından çekilen en yüksek çıkış akımı	25 mA
I/O pini tarafından sunulan en yüksek çıkış akımı	25 mA
Tüm portlar tarafından çekilen en yüksek akım.....	200 mA
Tüm portlar tarafından sunulan en yüksek akım	200 mA

Not 1: Güç tüketimi şu bağıntıyla hesaplanır: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$.

2: $\overline{\text{MCLR}}/\text{VPP}/\text{RE3}$ pininde gerçekleşen V_{SS} seviyesinin altındaki voltaj arkları, 80mA'den büyük akımların indüklenmesine ve giriş hattında aşırı akım yüklemesinden kaynaklı kısa devre oluşumuna (latch-up) neden olur. Bu nedenle, $\overline{\text{MCLR}}/\text{VPP}/\text{RE3}$ pininin "alçak" seviyeye çekilmesi girişe seri bağlı 50-100Ω'luk bir direnç üzerinden gerçekleştirilmelidir. Bu girişe bağlayacağınız reset butonu doğrudan şase seviyesine (V_{SS}) çekilmemelidir. PIC18F2550 serisi mikrodenetleyicilerde kullanılacak uygun bir reset devresi örneği Şekil F.9'da gösterilmiştir.



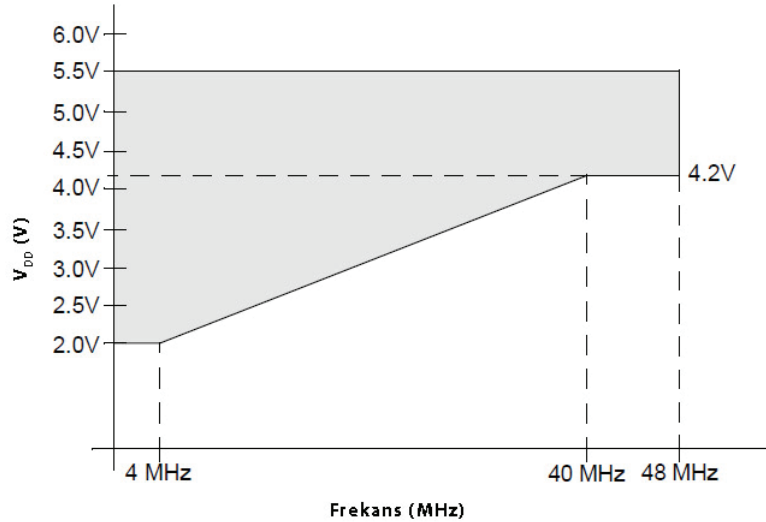
Şekil F.9 PIC18F2550 için RESET devresi

Bu devrede;

- Harici güç beslemesi reseti (Power-on Reset - POR) devresi yalnızca V_{DD} 'nin voltaj yükselmesi çok yavaşsa gereklidir. D diyotu V_{DD} beslemesi kesildiğinde kondansatörün hızlı şekilde deşarj olmasını sağlar.
- $R < 40 \text{ k}\Omega$ olması tavsiye edilir.
- $R1 \geq 1 \text{ k}\Omega$, harici C kondansatörü üzerinden $\overline{\text{MCLR}}$ pinine, $\overline{\text{MCLR}}/\text{VPP}$ pininin ESD (Electrostatic Discharge) ve EOS (Electrical Overstress) durumunda arızalanması sonucu oluşacak herhangi bir akım akışını sınırlamayı sağlar.

3: Dâhili USB regülatörü etkinleştirildiğinde ya da VUSB harici olarak beslendiğinde, RC4 ve RC5 V_{SS} 'ye göre $-0,3 - (V_{USB} + 0,3\text{V})$ arasına sınırlandırılır.

Diğer mikrodenetleyicilerde de olduğu gibi çalışma voltajına göre çıkılabilecek en yüksek frekans değişmektedir. Şekil F.10'da frekans ve voltaj ilişkisi gösterilmiştir.



Şekil F.10 Frekans ve voltaj ilişkisi

EC, HS, XT ve dâhili osilatör modlarında kaynak gerilimi minimum 2,0 V ve maksimum 5,5 V olarak tavsiye edilirken; HSPLL, XTPLL, ECPIO ve ECPLL osilatör modlarında minimum 3,0 V ve maksimum 5,5 V olarak tavsiye edilir. Dijital ve analog modüllerin elektriksel özellikleriyle ilgili daha ayrıntılı bilgi için veri kılavuzunu incelemeniz önerilir.

EK-G

G1-12F675 MİKRODENETLEYİCİSİNİN TEKNİK ÖZELLİKLERİ

PIC12F675, 8-pinli 8 bitlik giriş seviyesi (Base-Line) Flash ROM özellikli bir mikrodenetleyicidir. 14-bitlik komut setine sahiptir. Giriş seviyesi olmakla birlikte piyasada çok geniş bir kullanım alanı vardır. Parayla çalışan otomatlardan, ışığa duyarlı olarak çalışan sistemlere kadar, fazla port ihtiyacının duyulmadığı yerlerde pek çok uygulama alanı vardır. Teknik özellikleri genel olarak kısaca aşağıdaki gibidir:

Yüksek Performanslı RISC mimarili CMOS CPU:

- Çalışma frekansı 0 Hz (DC) – 20 MHz arasında
- 200ns'ye kadar komut çevrimi
- Kesme kabiliyetli
- 8-seviye derinliğinde donanım yığını
- Doğrudan, dolaylı ve bağıl adresleme modları
- 35 adet tek kelimelik komut kümesi
 - Dalların komutları hariç tüm komutlar tek çevrim süresinde

Özel Mikrodenetleyici Özellikleri:

- Dâhili ve Harici Osilatör Seçeneği:
 - $\pm 1\%$ fabrika kalibreli kararlı 4 MHz osilatör
 - Kristaller ve rezonatörler için harici osilatör desteği
 - 3,0V'ta uyku modundan 5 μ s'de uyanabilme özelliği
- Güç Tasarrufu Uyku modu
- Geniş çalışma voltajı aralığı (2,0V – 5,5V)
- Düşük güç POR özelliği
- İlk besleme gecikmesi zamanlayıcısı (PWRT) ve Osilatör başlangıç zamanlayıcısı (OST)
- Voltaj düşmesi algılama (BOD) özelliği
- Güvenilir çalışma için bağımsız osilatörlü Bekçi Zamanlayıcısı (WDT)
- Çoktan seçmeli \overline{MCLR} (Master Clear) girişi: Ana donanım resetlemesi pini aynı zamanda dijital I/O olarak kullanılabilir.
- Programlanabilir kod koruması
- Yüksek dayanıklı FLASH/EEPROM hücresi
 - FLASH için 100.000 yazma dayanıklılığı
 - EEPROM için 1.000.000 yazma dayanıklılığı
 - Flash/EEPROM veri saklama: >40 yıl

Düşük Güç Özellikleri:

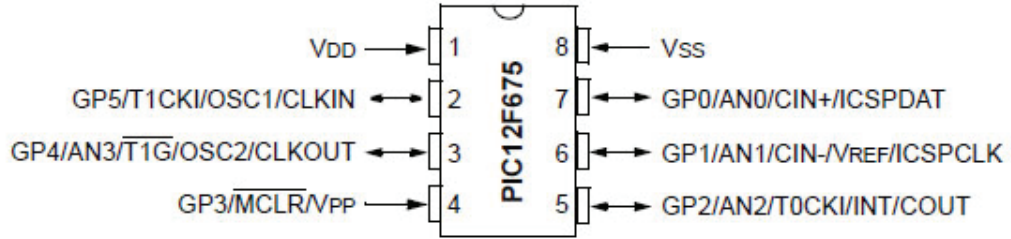
- Bekleme (Standby) akımı: 2,0V'ta **1nA**
- Çalışma akımı: 32 kHz 2,0V'ta **8,5uA**, 1 MHz 2,0V'ta **100uA**
- Bekçi zamanlayıcısı (WDT) akımı: 2,0V'ta **300nA**
- Timer1 osilatör akımı: 32 kHz 2,0V'ta **4uA**

Çevresel Aygıt Özellikleri:

- Bağımsız yön kontrollü 6 giriş, 5 çıkış (GP3 yalnızca giriş) pini
- Doğrudan LED sürmek için yüksek giriş/kaynak akımı
- Analog Karşılaştırıcı modülü:
 - Bir analog karşılaştırıcı
 - Programlanabilir dâhili voltaj referansı (CV_{REF}) modülü
 - Programlanabilir olarak çoktan seçmeli giriş
 - Harici olarak erişilebilir karşılaştırıcı çıkışı
- A/D Dönüştürücü modülü
 - 10-bit çözünürlüklü ve 4 kanallı
 - Voltaj referans girişi
- Timer0: 8-bitlik programlanabilir önölçekleyicili 8-bit zamanlayıcı/sayıcı
- Geliştirilmiş Timer1:
 - Önölçekleyicili 16-bit zamanlayıcı/sayıcı
 - Harici kapı giriş modu
 - Eğer INTOSC modu seçiliyse LP modunda OSC1 ve OSC2'yi Timer1 olarak kullanma seçeneği
- İki pin üzerinden ICSP™ desteği vardır.

Şekil G.1'de PIC12F675'in 8 pinli DIP, SOIC, DFN-S ve DFN kılıf türleri gösterilmiştir.

8-pin PDIP, SOIC, DFN-S, DFN



Şekil G.1 PIC12F675'in kılıf yapısı

G2-PIC12F675'İN BACAĞI (PİN) AÇIKLAMALARI

Tablo G.1 PIC12F675'in bacak açıklamaları

İsim	İşlev	Giriş Türü	Çıkış Türü	Açıklama
GP0/AN0/ CIN+/ICSPDAT	GP0	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN0	AN	-	0 numaralı A/D kanal girişi
	CIN+	AN	-	Karşılaştırmacı girişi
	ICSPDAT	TTL	CMOS	Seri programlama I/O
GP1/AN1/CIN-/ V _{REF} /ICSPCLK	GP1	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN1	AN	-	1 numaralı A/D kanal girişi
	CIN-	AN	-	Karşılaştırmacı girişi
	V _{REF}	AN	-	Harici voltaj referansı
	ICSPCLK	ST	-	Seri programlama saat darbesi girişi
GP2/AN2/TOCKI/ INT/COU	GP2	ST	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN2	AN	-	2 numaralı A/D kanal girişi
	TOCKI	ST	-	TMRO saat darbesi girişi
	INT	ST	-	Harici kesme girişi
	COU	-	CMOS	Karşılaştırmacı çıkışı
GP3/ $\overline{\text{MCLR}}$ /V _{PP}	GP3	TTL	-	Giriş portu ve durum değişim algılaması
	$\overline{\text{MCLR}}$	ST	-	Ana Reset
	V _{PP}	HV	-	Programlama voltajı
GP4/AN3/ $\overline{\text{T1G}}$ / OSC2/CLKOUT	GP4	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN3	AN	-	3 numaralı A/D kanal girişi

GP4/AN3/ $\overline{T1G}$ / OSC2/CLKOUT	$\overline{T1G}$	ST	-	TMR1 kapısı
	OSC2	-	XTAL	Kristal/rezonatör
	CLKOUT	-	CMOS	F _{osc} /4 çıkışı
GP5/T1CKI/OSC1/ CLKIN	GP5	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	T1CKI	ST	-	TMR1 saat darbesi girişi
	OSC1	-	-	Kristal/rezonatör
	CLKIN			Harici saat darbesi girişi/RC osilatör bağlantısı
VSS	VSS	Besleme	-	Lojik ve I/O pinleri için şase referansı
VDD	VDD	Besleme	-	Lojik ve I/O pinleri için pozitif besleme

Açıklama: ST = Schmitt tetikleyicili giriş tamponu

TTL = TTL'li giriş tamponu

EK-H

H1-12F1840 MİKRODENETLEYİCİSİNİN TEKNİK ÖZELLİKLERİ

PIC12F1840, 8-pinli 8 bitlik giriş seviyesi (Base-Line) Flash ROM özellikli bir mikrodenetleyici olmakla birlikte son derece gelişmiş özelliklere sahip çok güçlü bir mikrodenetleyicidir. Dâhili kapasitif algılayıcı modülü sayesinde ek bir donanıma ihtiyaç duymadan dokunmatik panel uygulamaları geliştirilebilir. Kitapta anlatılan mikrodenetleyiciler içinde dâhili DAC modülü olan tek mikrodenetleyicidir. Harici bir DAC entegresine ihtiyaç duymadan geniş bir tercih aralığında dijital-analog dönüşüm işlemi yapabilirsiniz. Yazılımsal olarak sistem hızını geniş bir aralıkta değiştirmenize imkan veren çok güçlü ve esnek osilatör donanımına sahiptir. Ayrıca dijital haberleşme uygulamalarında kullanılabilecek modülatör modülü de bulunmaktadır.

Tek bir I/O hattına çok sayıda görev tanımlanmış olmasından dolayı bazı özellikler birden fazla pinde de yer almaktadır. Böylece o özelliğin de aynı anda kullanılmasını gerektiren durumlarda kullanıcıya alternatif pin kullanma imkanı sağlanmıştır.

14-bitlik komut setine sahip bu mikrodenetleyicinin teknik özellikleri genel olarak kısaca aşağıdaki gibidir:

Yüksek Performanslı RISC mimarili CMOS CPU:

- 49 komuda sahiptir:
 - Dalların komutları hariç tümü tek saat çevrimlidir.
- Çalışma Hızı:
 - DC – 32 MHz osilatör/saat darbesi girişi
 - DC – 125 ns komut çevrim süresi
- Otomatik içerik tasarruflu kesme kabiliyeti
- Taşma/yetersizlik reset işlevli 16-Seviyeli derin donanım yığın adreslemesi
- Doğrudan, dolaylı ve bağıl adresleme modları:
 - İki tam 16-bitli Dosya Seçim Kaydedicileri (FSRs)
 - FSR'ler program ve veri belleğini okuyabilir

Esnek Osilatör Yapısı:

- Hassas 32 MHz dâhili osilatör bloğu:
 - $\pm 1\%$ Fabrika kalibreli
 - Yazılımsal olarak 31 kHz – 32 MHz arası seçilebilir frekans aralığı
- 31 kHz düşük-güç dâhili osilatör bloğu
- 32 MHz'e kadar dört kristal modu
- 32 MHz'e kadar üç harici saat darbesi modu
- 4x çarpan faktörlü Faz Kilitlemeli Döngü (PLL) devresi
- Güvenli (Fail-Safe) saat sarbesi monitörü:
 - Saat darbesi durursa güvenli kapanma
- İki hızlı osilatör açılış seçeneği
- Referans saat darbesi modülü:
 - Programlanabilir saat darbesi çıkış frekansı ve görev-çevrimi

Özel Mikrodenetleyici Özellikleri:

- Tam 5,5V çalışma – PIC12F1840
- 1,8V – 3,6V arası çalışma – PIC12LF1840 (düşük güç versiyonu)
- Yazılım kontrolü altında öz-programlanabilir
- Power-on Reset (POR), Power-up Timer (PWRT) ve Oscillator Start-up Timer (OST)
- Programlanabilir Brown-out Reset (BOR)
- Genişletilmiş vardiya zamanlayıcısı (Watchdog Timer - WDT)
- İki pin üzerinden In-Circuit Serial Programming™ (ICSP™) donanımıyla programlanabilme
- İki pin üzerinden dâhili hata izleme (In-Circuit Debug - ICD)
- Gelişmiş düşük güç programlama (LVP)
- Çalışma gerilimi aralığı:
 - 1,8V-5,5V (PIC12F1840)
 - 1,8V-3,6V (PIC12LF1840)
- Programlanabilir kod koruma
- Güç-tasarrufu uyku modu

Düşük Güç Özellikleri:

- Bekleme (Standby) akımı (PIC12LF1840): 1,8V'ta **20nA**
- Çalışma akımı (PIC12F1840): 1 MHz 1,8V'ta **75uA**
- Düşük güçlü vardiya zamanlayıcısı (WDT) akımı: 1,8V'ta **500nA**

Analog Özellikleri:

- Analog-Dijital dönüştürücü (DAC) modülü:
 - 10-bit çözünürlük, 4 kanal
 - Analog dönüştürme uyku sırasında da yapılabilir
- Analog karşılaştırıcı modülü:
 - Bir rail-to-rail analog karşılaştırıcı
 - Güç modu kontrolü
 - Yazılımsal olarak kontrol edilebilir histeresiz
- Voltaj Referans Modülü:
 - 1,024V, 2,048V ve 4,096V çıkış seviyeli sabit voltaj referansı (Fixed Voltage Reference - FVR)
 - 5-bit rail-to-rail pozitif ve negatif referans tercihli rezistif DAC

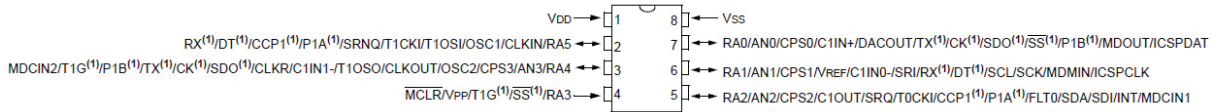
Çevresel Aygıt Özellikleri:

- 5 adet I/O pini ve 1 adet yalnızca giriş pini:
 - 25 mA/25 mA yüksek giriş/kaynak akımı
 - Programlanabilir dâhili zayıf çekme dirençleri
 - Programlanabilir durum değişim kesmesi pinleri
- Timer0: 8-bit önölçekleyici 8-bit zamanlayıcı/sayıcı
- Güçlendirilmiş Timer1:
 - Önölçekleyicili 16-bit zamanlayıcı/sayıcı
 - Harici kapı giriş modu

- Düşük-güçlü 32 kHz osilatör sürücüsü
- Timer2: 8-bit periyotlu kaydedici, önölçekleyici ve son ölçekleyicili 8-bit zamanlayıcı/sayıcı
- Güçlendirilmiş CCP (ECCP) modülü:
 - Yazılımsal seçilebilir zaman tabanı
 - Otomatik kapanma ve yeniden başlama
 - PWM yönlendirme
- SPI ve I²C modlarına sahip Master Synchronous Serial Port (MSSP):
 - 7-bit adres maskeleyme
 - SMBus/PMBusTM uyumluluğu
- Güçlendirilmiş Evrensel Senkron/Asenkron Verici-Alıcı (EUSART) modülü:
 - RS-232, RS-485 ve LIN uyumlu
 - Otomatik baud algılama
- Kapasitif algılama (CPS)
- Veri sinyal modülatörü modülü:
 - Seçilebilir modülatör ve taşıyıcı kaynakları
- SR tutma:
 - Çoklu Set/Reset giriş tercihi
 - 555 zamanlayıcı uygulamalarını taklit edebilme (555 Emulator)

Şekil H.1’de PIC12F1840’ın 8 pinli PDIP, SOIC ve DFN kılıf türleri gösterilmiştir.

PDIP, SOIC, DFN



Şekil H.1 PIC12F1840’ın kılıf yapısı

H2-PIC12F1840’IN BACAK (PİN) AÇIKLAMALARI

Tablo H.1 PIC12F1840’ın bacak açıklamaları

İsim	İşlev	Giriş Türü	Çıkış Türü	Açıklama
RA0/AN0/CPS0/C1IN+/ DACOUT/TX ⁽¹⁾ /CK ⁽¹⁾ / SDO ⁽¹⁾ /SS ⁽¹⁾ /P1B ⁽¹⁾ / MDOUT/ICSPDAT/ ICDDAT	RA0	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN0	AN	-	0 numaralı A/D kanal girişi
	CPS0	AN	-	0 numaralı kapasitif algılama girişi
	C1IN+	AN	-	Karşılaştırmacı C1 evirmeyen girişi
	DACOUT	-	AN	Dijital-Analog dönüştürücü çıkışı
	TX	-	CMOS	USART asenkron iletim hattı
	CK	ST	CMOS	USART senkron saat darbesi hattı

RA0/AN0/CPS0/C1IN+/ DACOUT/TX ⁽¹⁾ /CK ⁽¹⁾ / SDO ⁽¹⁾ / \overline{SS} ⁽¹⁾ /P1B ⁽¹⁾ / MDOOUT/ICSPDAT/ ICDDAT	SDO	-	CMOS	SPI veri çıkışı
	\overline{SS}	ST	-	Bağımlı aygıt seçme (Slave Select) girişi
	P1B	-	CMOS	PWM çıkışı
	MDOOUT	-	CMOS	Modülatör çıkışı
	ICSPDAT	ST	CMOS	ICSP™ veri I/O hattı
RA1/AN1/CPS1/VREF/ C1IN0-/SRI/RX ⁽¹⁾ /DT ⁽¹⁾ / SCL/SCK/MDMIN/ ICSPCLK/ICDCLK	RA1	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN1	AN	-	1 numaralı A/D kanal girişi
	CPS1	AN	-	1 numaralı kapasitif algılama girişi
	V _{REF}	AN	-	Harici voltaj referansı
	C1IN0-	AN	-	Karşılaştırmacı C1 0 numaralı eviren giriş kanalı
	SRI	ST	-	SR tutma girişi
	RX	ST	-	USART asenkron girişi
	DT	ST	CMOS	USART senkron veri hattı
	SCL	I ² C™	OD	I ² C™ saat darbesi hattı
	SCK	ST	CMOS	SPI saat darbesi hattı
	MDMIN	ST	-	Modülatör kaynağı girişi
ICSPCLK	ST	-	Seri programlama saat darbesi girişi	
RA2/AN2/CPS2/ C1OUT/SRQ/T0CKI/ CCP1 ⁽¹⁾ /P1A ⁽¹⁾ /FLT0/ SDA/SDI/INT/MDCIN1	RA2	ST	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN2	AN	-	2 numaralı A/D kanal girişi
	CPS2	AN	-	2 numaralı kapasitif algılama girişi
	C1OUT	-	CMOS	Karşılaştırmacı C1 çıkışı
	SRQ	-	CMOS	SR tutma evirmeyen çıkışı (Q)
	T0CKI	ST	-	Timer0 saat darbesi kaynağı girişi
	CCP1	ST	CMOS	Capture/Compare/PWM 1

RA2/AN2/CPS2/ C1OUT/SRQ/T0CKI/ CCP1 ⁽¹⁾ /P1A ⁽¹⁾ /FLT0/ SDA/SDI/INT/MDCIN1	P1A	-	CMOS	PWM çıkışı
	FLT0	ST	-	ECCP otomatik kapanma hata girişi
	SDA	I ² C™	OD	I ² C™ veri girişi/çıkışı
	SDI	CMOS	-	SPI veri girişi
	INT	ST	-	Harici kesme
	MDCIN1	ST	-	Modülatör taşıyıcısı 1 numaralı girişi
RA3/ $\overline{SS}^{(1)}$ /T1G ⁽¹⁾ / VPP/ \overline{MCLR}	RA3	TTL	-	Giriş portu ve durum değişim algılaması
	\overline{SS}	ST	-	Bağımlı seçme girişi
	T1G	ST	-	Timer 1 kapı girişi
	V _{PP}	HV	-	Programlama voltajı
	\overline{MCLR}	ST	-	Dâhili pull-up'lı ana Reset
RA4/AN3/CPS3/OSC2/ CLKOUT/T1OSO/ C1IN1-/CLKR/SDO ⁽¹⁾ / CK ⁽¹⁾ /TX(1)/P1B ⁽¹⁾ / T1G ⁽¹⁾ /MDCIN2	RA4	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	AN3	AN	-	3 numaralı A/D kanal girişi
	CPS3	AN	-	3 numaralı kapasitif algılama girişi
	OSC2	-	XTAL	Kristal/Rezonatör (LP, XT ve HS modları)
	CLKOUT	-	CMOS	F _{osc} /4
	T1OSO	XTAL	XTAL	Timer1 osilatör bağlantısı
	C1IN1-	AN	-	Karşılaştırıcı C1 1 numaralı eviren giriş kanalı
	CLKR	-	CMOS	Saat darbesi referans çıkışı
	SDO	-	CMOS	SPI veri çıkışı
	CK	ST	CMOS	USART senkron saat darbesi hattı
	TX	-	CMOS	USART asenkron iletim hattı
	P1B	-	CMOS	PWM çıkışı
	T1G	ST	-	TMR1 kapısı girişi
MDCIN2	ST	-	Modülatör taşıyıcısı 2 numaralı girişi	

RA5/CLKIN/OSC1/ T1OSI/T1CKI/SRNQ/ P1A ⁽¹⁾ /CCP1 ⁽¹⁾ / DT ⁽¹⁾ /RX ⁽¹⁾	RA5	TTL	CMOS	Çift yönlü I/O, programlanabilir pull-up ve durum değişim algılaması kesmesi
	CLKIN	CMOS	-	Harici saat darbesi girişi (EC)
	OSC1	XTAL	-	Kristal/Rezonatör (LP, XT ve HS modları)
	T1OSI	XTAL	XTAL	Timer1 osilatör bağlantısı
	T1CKI	ST	-	Timer1 saat darbesi kaynağı girişi
	SRNQ	-	CMOS	SR tutma eviren çıkışı (Not Q)
	P1A	-	CMOS	PWM çıkışı
	CCP1	ST	CMOS	Capture/Compare/PWM 1
	DT	ST	CMOS	USART senkron veri
	RX	ST	-	USART asenkron giriş
VSS	VSS	Besleme	-	Lojik ve I/O pinleri için şase referansı
VDD	VDD	Besleme	-	Lojik ve I/O pinleri için pozitif besleme

Açıklama: ST = CMOS seviyeli Schmitt tetikleyicili giriş

OD: Open Drain

TTL = TTL uyumlu giriş

AN: Analog giriş ya da çıkış

CMOS = CMOS uyumlu giriş ya da çıkış

XTAL: Kristal

I²C™ = I²C™ seviyelerinde Schmitt tetikleyicili giriş

HV: Yüksek voltaj

Not 1: Alternatif Pin İşlevi Kontrol (Alternate Pin Function Control - APFCON) kaydedicisi ile bu özellikler iki pinden biri arasında değiştirilebilir.

H3- APFCON (ALTERNATE PIN FUNCTION CONTROL REGISTER) KAYDEDİCİSİ

Alternatif pin işlevi kontrol kaydedicisi belirli çevresel aygıt giriş ve çıkışlarını pinler arasında geçiş yaptırmak için kullanılır. Yeni nesil mikrodenetleyicilerde bu işlem PPS (Peripheral Pin Select) adıyla ve daha kapsamlı olarak sunulmuştur. APFCON kaydedicisi ile aşağıdaki görevler farklı pinler arasında hareket ettirilebilir.

- RX/DT
- TX/CK
- SDO
- \overline{SS} (Slave Select)
- T1G
- P1B
- CCP1/P1A

Bu bitlerin herhangi TRIS kaydedicisi değeri üzerinde etkisi yoktur. PORT ve TRIS yönlendirmeleri de otomatik olarak doğru pine yapılır. Seçilmeyen pin etkilenmez.

APFCON - ALTERNATİF PİN İŞLEVİ KONTROL KAYDEDİCİSİ

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
RXDTSEL	SDOSEL	SSSEL	—	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL
bit 7							bit 0

- bit 7** **RXDTSEL:** RX asenkron seri alma ve DT senkron veri pinini yönlendirme bitidir.
0 = RX/DT işlevi RA1 üzerindedir.
1 = RX/DT işlevi RA5 üzerindedir.
- bit 6** **SDOSEL:** SPI veri çıkış pinini yönlendirme bitidir.
0 = SDO işlevi RA0 üzerindedir.
1 = SDO işlevi RA4 üzerindedir.
- bit 5** **SSSEL:** Bağımlı aygıt seçme pinini yönlendirme bitidir.
0 = \overline{SS} işlevi RA3 üzerindedir.
1 = \overline{SS} işlevi RA0 üzerindedir.
- bit 4** **Kullanılmaz:** '0' olarak okunur.
- bit 3** **T1GSEL:** Timer1 kapı girişini yönlendirme bitidir.
0 = T1G işlevi RA4 üzerindedir.
1 = T1G işlevi RA3 üzerindedir.
- bit 2** **TXCKSEL:** TX asenkron iletim ve CK senkron saat darbesi pinini yönlendirme bitidir.
0 = TX/CK işlevi RA0 üzerindedir.
1 = TX/CK işlevi işlevi RA4 üzerindedir.
- bit 1** **P1BSEL:** PWM B çıkışını yönlendirme bitidir.
0 = P1B işlevi RA0 üzerindedir.
1 = P1B işlevi RA4 üzerindedir.
- bit 0** **CCP1SEL:** CCP girişini ve PWM A çıkışını yönlendirme bitidir.
0 = CCP1/P1A işlevi RA2 üzerindedir.
1 = CCP1/P1A işlevi RA5 üzerindedir.

EK-I

KİTAPTA KULLANILAN DİĞER YARI İLETKEN ELEKTRONİK MALZEMELER HAKKINDA TEMEL KATALOG BİLGİSİ

11 - ULN2003A YÜKSEK AKIMLI DARLİNGTON DİZİSİ

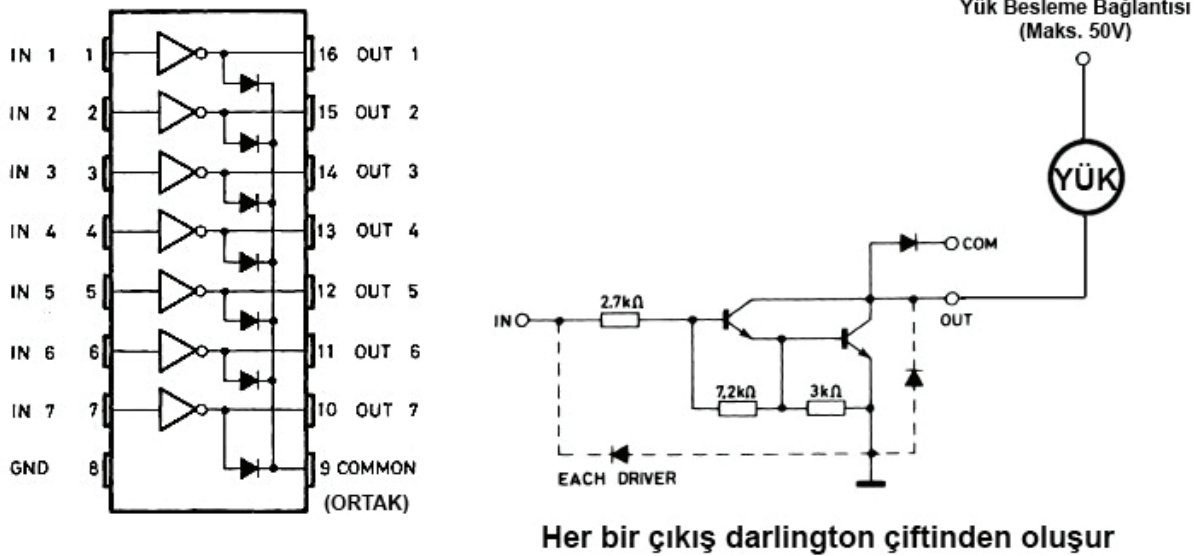
ULN2003A yüksek voltaja sürülebilir, yüksek akımlı darlington dizisi çıkışlardan oluşan ortak emetörlü açık kollektörlü bir sürücü entegresidir. Her kanalı ortalama 500mA'lık olup, 600mA'lık pik akımlarına dayanabilir. Endüktif yük sürme uygulamaları için dâhili bastırma diyotları eklenmiştir. Ayrıca giriş ve çıkışlar karşılıklıdır. Böylece devre tasarımlarında ek bağlantı ve malzeme ihtiyacı ortadan kalkar.

50V'a kadar yük beslemesi yapılabilir. Bu durum röleleri, adım motorlarını, selenoidleri, aydınlatma elemanlarını vb. yükleri kontrol etmenizi kolaylaştırır. Daha yüksek güç gerektiren uygulamalar için ULN2023 A/L ya da ULN2024 A/L modelleri kullanılabilir. Çıkışlar paralel bağlanarak daha yüksek akımlı yükler kontrol edilebilir.

Not: Her ne kadar giriş gerilimi 5V olarak tanımlansa da, maksimum çıkış akımı 300mA'e düşse bile 3.3V'luk giriş voltajıyla da iyi çalışır.

Özellikler

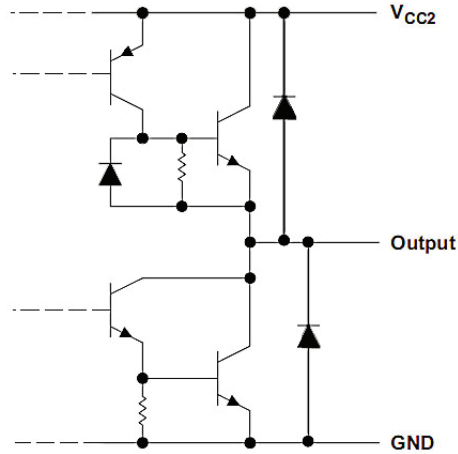
- Paket: 16 pin DIP
- Darlington çifti sayısı: 7
- Çıkış akımı: Darlington çifti başına 500mA
- Çıkış gerilimi (yük gerilimi): Maks 50V
- Giriş gerilimi: 5V (3.3V'ta da çalışır)
- Daha yüksek yük akımları için çıkışlar paralel bağlanabilir



Şekil I.1 ULN2003A yüksek akımlı darlington dizisi

I2 - L293D DÖRT KANALLI YARIM-H SÜRÜCÜSÜ

L293D, dörtlü yüksek akım yarım-H sürücü entegresidir. 4.5V ile 36V arasında 600mA'e kadar çift yönlü akım çekilebilir. ULN2003A entegresi gibi bu entegre de endüktif yükleri kontrol etmede kullanılır. En önemli özelliği fiziki olarak bağlantıyı değiştirmeden voltaj kutuplarını anahtarlayarak tersleyebilmesidir. Böylece DC motor yönünün değiştirilmesi istenen durumlarda ya da bipolar adım motorlarının komütasyonunda önemli bir kullanım kolaylığı sağlar. Tüm girişleri TTL seviyelidir ve MCU'larla uyumludur. Her bir çıkış darlington bağlı transistör çiftlerinin push-pull bağlantısı şeklinde tasarlanmıştır (Bkz. Şekil I.2). Çıkışlarda kenetleme diyotları bulunur.



Şekil I.2 L293D'nin çıkışlarının şematik gösterimi

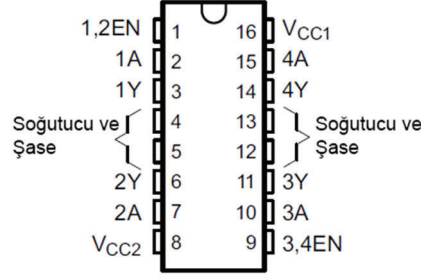
Özellikler

- Paket: 16 pin DIP
- Geniş besleme aralığı: 4.5V – 36V
- Ayrı giriş-lojik beslemesi
- Dâhili elektrostatik deşarj (ESD) koruması vardır
- Girişler gürültüye karşı yüksek bağışıklıdır
- Çıkış akımı kanal başına 600mA'dir
- Çıkış pik akımı kanal başına 1.2A'dir
- Endüktif yükleme etkisini bastırmak için çıkışlarda kenetleme diyotları vardır

Uygulama Alanları

- Adım motor sürücüleri
- DC motor sürücüleri
- Kilitleme rölesi sürücüleri
- Solenoid yükler

Entegrenin içindeki 4 yarım-H sürücüsü için 4 adet çıkış vardır. Her sürücü çiftinin EN yetkilendirme girişi vardır. EN girişlerine lojik-1 (5V) uygulandığında ilgili sürücü çifti etkin olur. Entegre bacak bağlantıları Şekil I.3'te gösterilmiştir. Yetkilendirme girişinin lojik-0 olması durumunda giriş ne olursa olsun ilgili çıkış lojik-0 olur.

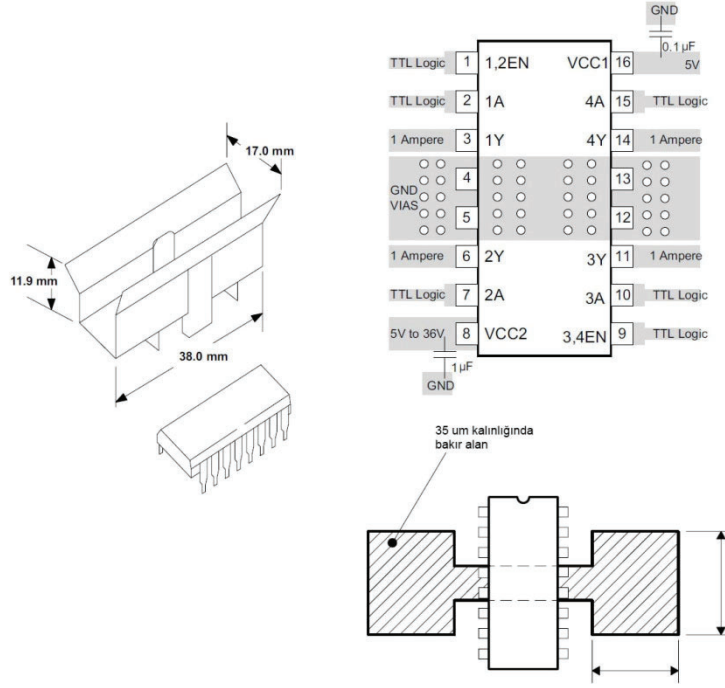


Şekil I.3 L293D Entegresi

Tablo I.1 L293D bacak açıklamaları

PİN		TÜR	AÇIKLAMA
İSİM	NO		
1,2EN	1	I	1 ve 2 sürücü kanallarını etkinleştirme (aktif yüksek giriş)
<1:4>A	2, 7, 10, 15	I	Sürücü girişi, terslemeyen
<1:4>Y	3, 6, 11, 14	O	Sürücü çıkışları
3,4EN	9	I	3 ve 4 sürücü kanallarını etkinleştirme (aktif yüksek giriş)
ŞASE	4, 5, 12, 13	-	Cihaz şase ve soğutucu pini. Devre PCB'sinin şasesine çoklu yekpare yol ile bağlayın
V _{CC1}	16	-	Dâhili lojik için 5V besleme
V _{CC2}	8	-	Sürücüler için 4.5-36V arasında değişen güç beslemesi

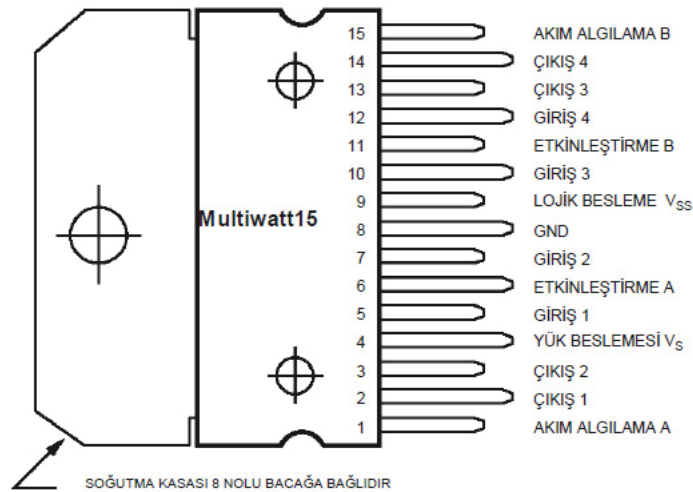
Yüksek akımlı uygulamalarda entegrenin soğutulması gerekir. Şekil I.4'te gösterildiği şekilde entegre gövdesine uygun bir soğutucu kullanılabilir. Ayrıca devre kartının PCB'si çıkartılırken entegrenin şase pinlerinin lehmleneceği bakır hat şeklinde gösterildiği gibi yekpare şekilde çizilmek suretiyle soğutma alanı genişletilebilir. Her iki durumda da hem harici soğutucu hem de baskı devre bakırı elektriksel olarak şaseselenmelidir.



Şekil I.4 Solda harici soğutma uygulaması, sağda baskı devre bakır hattıyla soğutma uygulaması

13 - L298N ÇİFT KANALLI TAM KÖPRÜ SÜRÜCÜSÜ

Bu sürücü de L293 sürücüsüne benzer. Dört yarım-H sürücüsüne sahiptir ve bunlar 2 tam köprü sürücü sistemini oluşturur. A ve B köprüleri olarak isimlendirilirler. L293 ve L293D sürücülerine göre daha yüksek yük akımlarını kontrol edebilir. L298N 15 pinli Multiwatt kasa formatında tasarlanmıştır. Standart TTL lojik seviyesinde tetiklenerek çalışır. Dolayısıyla bu sürücü de mikrodenetleyicilerle kullanıma uygundur.



Şekil I.5 L298N Çift Kanallı Tam Köprü Sürücüsü

Özellikler

- Paket: 15 pin Multiwatt
- Geniş besleme aralığı: Maksimum 46V
- Ayrı giriş-lojik beslemesi: Maksimum 7V

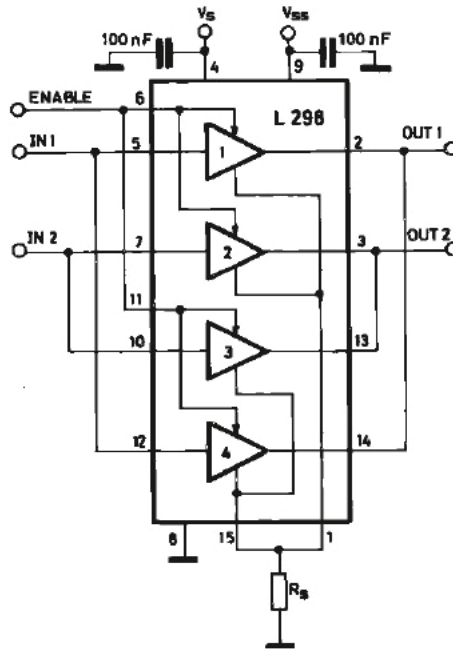
- Çıkış akımı kanal başına tam DC sürmede 2A'dır
- Akım algılama voltaj hassasiyeti -1V – 2,3V arasındadır

Uygulama Alanları

- Adım motor sürücüleri
- DC motor sürücüleri
- Röleler
- Solenoid yükler

Bu sürücüde de 2 etkinleştirme girişi bulunur. Böylece giriş sinyallerinden bağımsız olarak her iki kanal da aktif ya da pasif kılınabilir. Bu entegre ile motor gibi endüktif ya da röle gibi solenoid sargılı yükler kontrol edilirken çıkışlara kenetleme diyodu bağlanmalıdır (Bkz. Şekil 8.8). İhtiyacınıza uygun devre bağlantı modelleri için veri kılavuzunu inceleyiniz. Kenetleme diyodu olarak Schottky diyot kullanılabileceği gibi 1N4148 ya da daha yüksek ters kırılma voltajına sahip 1N4007 gibi normal silisyum diyotlar da kullanılabilir.

DİKKAT: L298 serisi sürücü entegrelerinin V_S ve V_{SS} besleme girişlerinin hemen yakınına kutupsuz ve endüktif olmayan 100nF kondansatörlerden bağlanması tavsiye edilir. Akım algılama girişlerine (Sense A, Sense B) tel sargılı olmayan Wattlı dirençler, V_S kaynağının negatif kutbunun hemen yakınında şaseleme şeklide bağlanmalıdır. Bunun için 1 Ω /1W karbon film direnç kullanılabilir. L298N ile daha yüksek akım kontrol etmek istediğinizde çıkışlar paralel bağlanabilir. Bu durumda Çıkış 1 (2), Çıkış 4 (14) ile Çıkış 2 (3) ise Çıkış 3 (13) ile paralel olarak bağlanır. Böylece toplamda 3,5A'e kadar yük kontrolü sağlanabilir. Bu ilkeler ışığında gerçekleştirilecek bir bağlantı Şekil 1.6'daki devre örneğindeki gibi olmalıdır.

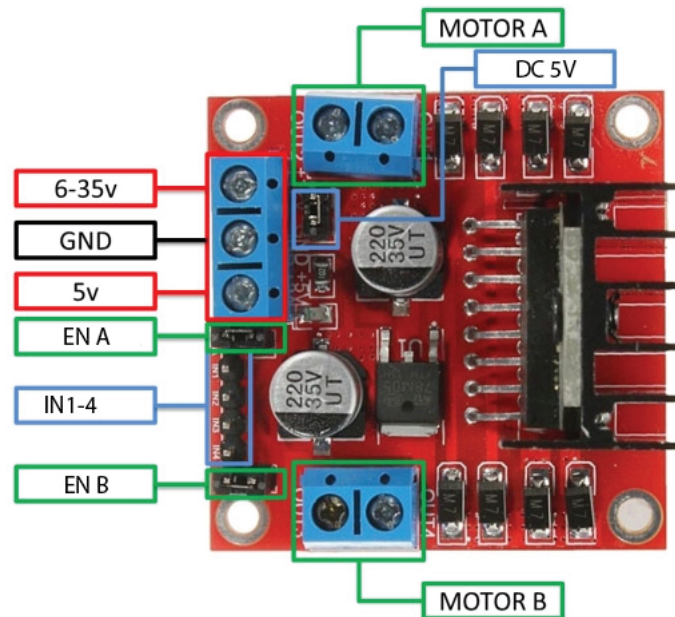


Şekil 1.6 L298N'de çıkışların paralel bağlanması

Tablo I.2 L298N bacak açıklamaları

İSİM	NO	AÇIKLAMA
Sense A, Sense B	1, 15	Bu pin ve şase arasına yük akımını kontrol etmek için algılama direnci (R_s) bağlanır.
Out 1, Out 2	2, 3	A köprüsünün çıkışlarıdır. Bu iki pin arasında bağlı yükten geçen akım 1 numaralı bacadan izlenebilir.
Vs	4	Yük besleme bağlantısıdır. Bu pin ve şase arasında 100nF'lık endüktif olmayan bir kondansatör bağlanmalıdır.
Input 1, Input 2	5, 7	A köprüsünün TTL uyumlu girişleridir.
Enable A, Enable B	6, 11	TTL uyumlu köprü etkinleştirme girişleridir. Lojik-0 durumunda ilgili köprü devre dışı kalır. İlgili köprü girişlerinin kullanılabilmesi için lojik-1 yapılmalıdır.
GND	8	Şase
Vss	9	Lojik devre besleme bağlantısıdır. Bu pin ve şase arasında 100nF'lık endüktif olmayan bir kondansatör bağlanmalıdır.
Input 3, Input 4	10, 12	B köprüsünün TTL uyumlu girişleridir.
Out 3, Out 4	13, 14	B köprüsünün çıkışlarıdır. Bu iki pin arasında bağlı yükten geçen akım 15 numaralı bacadan izlenebilir.

Piyasada L298N sürücüsü kullanılarak yapılmış paket motor sürücüsü kartı vardır (Bkz. Şekil I.7). Bu kartı kullanarak çizgi izleyen robot, sumo robotu, vb. araç kontrollerini gerçekleştirebilirsiniz. Kartın üzerinde 5V sabit voltaj çıkışı bulunmaktadır. Bu çıkışı kullanarak mikrodenetleyiciniz için gerekli beslemeyi elde edebilirsiniz. Böylece 12V akü gruplarıyla ya da yüksek akımlı LİPO (Lithium-Polymer) bataryalarla hazırladığınız tek bir DC beslemeyle tüm güç ihtiyacınızı ikinci bir güç kaynağı kullanmadan sağlamanız mümkündür. Ayrıca kart üzerinde çıkış kenetleme diyotları da olduğundan haricen bir devre elemanı bağlantısı yapmanıza gerek yoktur. MotorA ya da MotorB yazan klemenslere kumanda etmek istediğiniz yükleri bağlamanız yeterli olacaktır.



Şekil I.7 Hazır L298N sürücüsü paket devresi

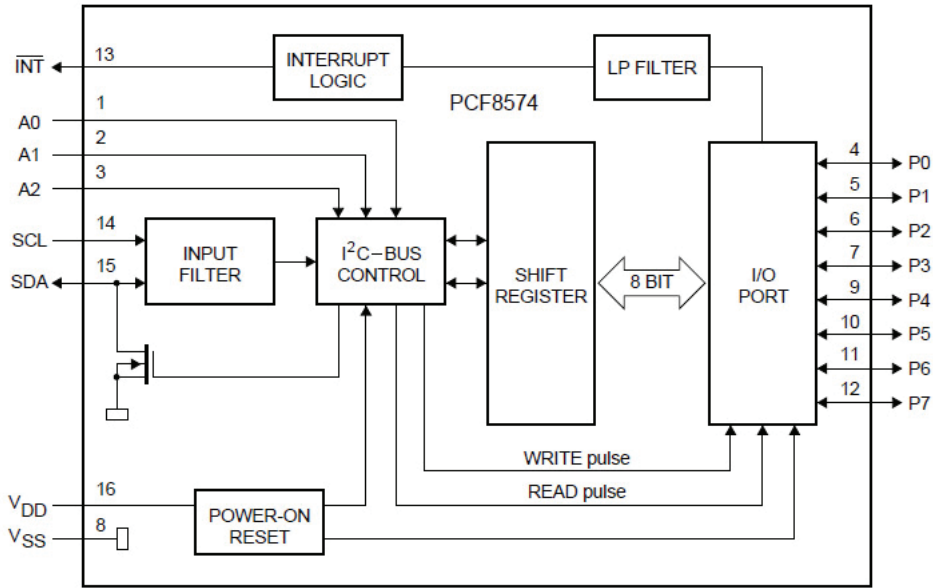
14 - PCF8574/A PORT GENİŞLETİCİSİ

PCF8574 ve PCF8574A Texas Instruments firmasının geliştirmiş olduğu ve MSSP I²C iletişim modunu kullanan bir port genişleticisidir. 2 hatlı I²C yolunu 8-bit paralel yola genişletir. İki arasındaki tek fark adres aralıklarının farklı olmasıdır.

Bazı Teknik Özellikleri:

- Adresleri donanımsal olarak ayarlanır
- Her bir pininden 25mA akım geçebilir.
- Mikrodenetleyicilerin I/O pinleriyle yapılan pek çok kontrol ve izleme uygulaması bu aygıt pinleri üzerinden de yapılabilir.
- Pinlerden birinde meydana gelen durum değişiminde açık-akar (open drain) kesme çıkışına sahiptir.
- 2,5V – 5,5V arası besleme gerilimine sahiptir.

PCF8574 farklı paket türlerinde üretilmekte olup pin sıralamaları değişmektedir. Şekil 1.8’de 16 pinli PDIP paketinin bacak bağlantıları ve Tablo 1.3’te de 16 pinli PDIP paketine göre bacak açıklamaları gösterilmiştir.



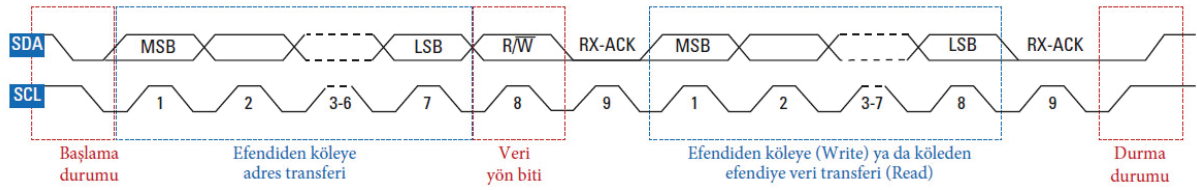
Şekil 1.8 PCF8574/A bacak bağlantıları

Tablo 1.3 PCF8574 bacak açıklamaları

İSİM	NO	TÜR	AÇIKLAMA
A0, A1, A2	1, 2, 3	I	Adres bağlantı bacaklarıdır. Doğrudan V _{DD} ya da GND'ye bağlanır. Direnç kullanmaya gerek yoktur.
GND	8	-	Şase
INT̄	13	O	Kesme çıkışıdır. Bir pull-up direnci ile V _{DD} 'ye bağlanır.
P0, P1, P2, P3, P4, P5, P6, P7	4, 5, 6, 7, 9, 10, 11, 12	I/O	P giriş/çıkış portudur.

SCL	14	I	Seri saat darbesi hattıdır. Bir pull-up direnci ile V _{DD} 'ye bağlanır.
SDA	15	I/O	Seri veri hattıdır. Bir pull-up direnci ile V _{DD} 'ye bağlanır.
V _{CC}	16	-	Besleme

Şekil I.9'da iletişim sinyalizasyonunun nasıl gerçekleştiği gösterilmiştir. Sinyalizasyona dikkat edileceği üzere 3 bitlik adres sinyali ardından 1 bitlik R/ \bar{W} (okuma/yazma) sinyali gönderilir. Bu sinyal lojik-1 ise köle aygıtın P portundan MCU'ya okuma, eğer lojik-0 ise MCU'dan köle aygıtın P portuna yazma işlemi yapılacağı anlaşılır. 8-bitlik adres sinyali bittikten sonra onay (ACK – Acknowledge) biti gönderilir. SCL hattı lojik-1 iken, SDA hattının onay bitinden sonra lojik-1 seviyesine geçmesi haberleşmeyi bitirir. Onay sinyalinin iletildiği sırada oluşacak bir kesme kaybolabilir. Çünkü onay sinyali sırasında kesme resetlenir. Reset işleminden sonra, saat darbesinin bir sonraki yükselen kenarında I/O hatlarındaki herhangi bir değişim kesme sinyalini tekrar üretir.



Şekil I.9 I²C seri okuma ve yazma sinyalizasyonu

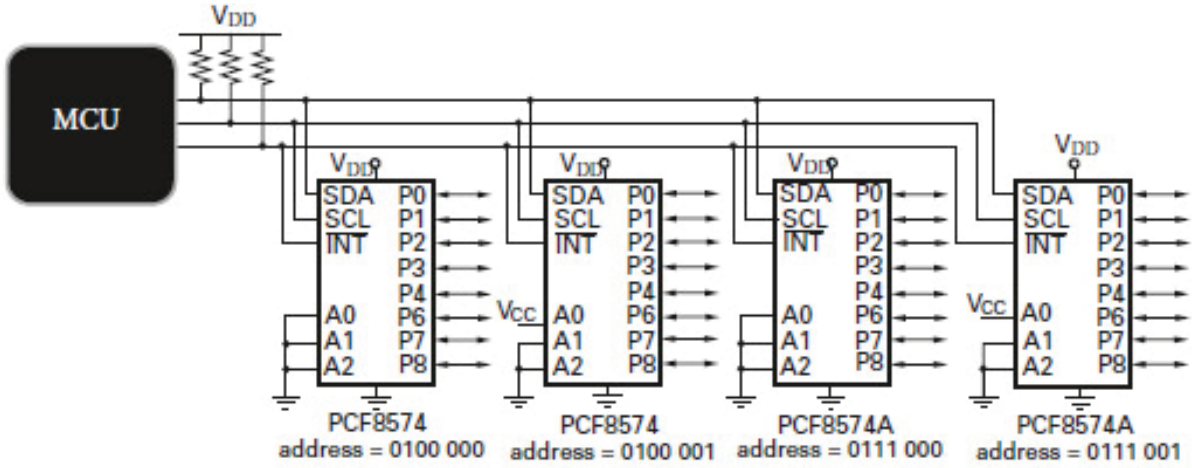
Okuma ve yazma modlarında cihazların adresleri değişir. Pinlerin donanımsal bağlantılarına göre adres tablosu, her iki aygıt modeli için de Tablo I.4'te gösterilmiştir.

Tablo I.4 PCF8574/A için adres tablosu

GİRİŞLER			I ² C köle aygıttan 8-bit okuma adresi		I ² C köle aygıt 8-bit yazma adresi	
A2	A1	A0	PCF8574	PCF8574A	PCF8574	PCF8574A
L	L	L	41	71	40	70
L	L	H	43	73	42	72
L	H	L	45	75	44	74
L	H	H	47	77	46	76
H	L	L	49	79	48	78
H	L	H	4B	7B	4A	7A
H	H	L	4D	7D	4C	7C
H	H	H	4F	7F	4E	7E

Tablodan görüldüğü üzere A0-A1-A2 pinlerinin lojik seviyesine göre $2^3 = 8$ farklı adres elde edilir. Böylece tek bir I²C yoluna 8 adet PCF8574 ya da PCF8574A bağlanabilir. İki modelin adres aralıkları farklı olduğu için, 8'er adet PCF8574 ve PCF8574A olmak üzere toplamda 16 aygıt aynı anda tek bir hat üzerine bağlanabilir. Şekil I.10'da böyle bir bağlantının nasıl yapılacağı gösterilmiştir.

Kesme çıkışı mikrodenetleyicinin kesme girişine bağlanarak MCU'yu uyarmada kullanılabilir. Aygıtın pinleri giriş moduna alındığında, girişin lojik-0'dan lojik-1'e ya da lojik-1'den lojik-0'a geçme durumlarında kesme aktif olur.



Sekil I.10 Çok sayıda PCF8574/A'nın aynı I2C hattına bağlantısı (Kaynak: Texas Instruments)

Not: PCF8574'ün donanımsal özellikleri hakkında daha ayrıntılı bilgi için veri kılavuzunu incelemeniz tavsiye edilir.

15 - HD44780U ALFANÜMERİK LCD EKLAN SÜRÜCÜSÜ

HD44780U Hitachi firmasının düşük çözünürlüklü sıvı kristal göstergeler için geliştirmiş olduğu, dot-matris formatta alfanümerik karakter üretici ve sürücü çipsetidir. Sahip olduğu 8 veri hattı üzerinden (DB7 ... DBO) gönderilen veri paketlerini işleyerek karakter üretir. Daha az pin sayısı ile karakter yazdırılmasını sağlayan yapısı sayesinde 4-bitlik paketler halinde de veri işleyip karakter üretebilir. Görüntü RAM'i, karakter üretici ve sıvı kristal sürücüsü tek bir çip üzerinde yer alır. Karakter üretici ROM belleğinde standart ASCII karakterlerini, Japon Kana karakterlerini ve bazı sembolleri saklamaktadır. Japon ve Avrupa dil paketi için farklı sürümleri mevcuttur. Karakter üretici ROM belleği 208 adet 5x8 formatında, 32 adet 5x10 formatında olmak üzere 240 karakter fontunu saklayabilir.

Bu sürücü çipseti kullanılarak tasarlanmış 16x1, 16x2, 16x4, 20x2, 20x4, 32x2, 40x2 gibi farklı sütun ve satır formatında LCD ekranlar vardır. Tek bir HD44780U çipseti tek 8-karakter satırına ya da iki 8-karakter satırına kadar görüntüleme yapabilmektedir. Genişleme sürücüsü kullanılarak HD44780U çipsetiyle aynı anda 80 karakter görüntülemesi yapılabilir. Bu çipset yardımıyla yalnızca LCD ekrana karakter gönderimi yapılmakla kalmaz, aynı zamanda o an için LCD ekran veri belleğinde yazılı bilgi de mikrodenetleyiciye okunabilir.

Teknik Özellikleri:

- 5 x 8 ve 5 x 10 nokta maktris görüntüleme mümkündür.
- 2,7V'tan 5,5V'a kadar düşük güç desteği vardır.
- 3V'tan 11V'a kadar geniş bir voltaj aralığında sıvı kristal ekran ışığı beslemesi seçeneği vardır.
- Sıvı kristal aydınlatması AC gerilimle çalışabilir.
- İşlemciyle yüksek hızda iletişim kurabilir (5V'ta 2MHz).
- 4-bit ya da 8-bit işlemci bağlantı arayüzü mümkündür.
- 80 x 8 bit görüntü RAM belleği vardır (en fazla 80 karakter)
- 9920 bit karakter üretici ROM (CGROM-Character Generator ROM) belleği vardır (toplamda 240 karakter fontu).
- 64 x 8 bit karakter üretici RAM (CGRAM) belleği vardır (isteğe özel geçici karakter fontu, sembol, vs. tasarlamak mümkündür).
 - 8 karakter fontu (5 x 8)
 - 4 karakter fontu (5 x 10)
- 16 ortak olmak üzere x 40 segment sıvı kristal görüntü sürücüsü bacağı vardır. Ekranda karakterlerin oluşturulması bu 56 iletişim hattı ile sağlanır.
- Programlanabilir görev çevrim oranları vardır. Böylece aynı anda kaç satır ve kaç nokta matris kontrolünün yapılacağı belirlenir.
 - Karakter altında kursör alanı dahil 5 x 8 nokta matris ve 1 satır için 1/8
 - Karakter altında kursör alanı dahil 5 x 10 nokta matris ve 1 satır için 1/11
 - Karakter altında kursör alanı dahil 5 x 8 nokta matris ve 2 satır için 1/16
- Geniş bir aralıkta farklı komut işlevine sahiptir.
 - Ekran temizleme, kursörü başa alma, görüntüyü açma/kapatma, kursörü açma/kapatma, karakter yakıp söndürme, kursörü kaydırma, görüntüyü kaydırma
- Enerjilendiğinde kontrolcü ve sürücüyü başlatan otomatik reset devresi vardır.
- Harici dirençli dâhili osilatörü vardır.

- Batarya temelli uygulamalar için düşük güç tüketimine sahiptir.

HD44780U çipseti iki adet 8-bitlik kaydediciye sahiptir:

- Komut kaydedicisi (IR-Instruction Register)
- Veri kaydedicisi (DR-Data Register)

IR kaydedicisi; kursör kaydırma, ekran temizleme ve görüntü veri belleği (DDRAM-Display Data RAM) için adres bilgisi gibi komut kodlarını tutar.

DR ise DDRAM'e ve CGRAM'e geçici olarak yazılan ya da onlardan geçici olarak okunacak bilgiyi tutar. Adres bilgisi komut kaydedicisine yazıldığında, veri dâhili işlemle DDRAM'den ya da CGRAM'den veri kaydedicisine okunur ve ardından kaydedilir.

Çipsetin RS piniyle bu iki kaydedici arasında geçiş yapılırken, R/ \bar{W} piniyle okuma ya da yazma işlemi gerçekleştirilir. DB7-DB0 pinleri karakter bilgisinin iletilmesi ve çeşitli komutların gerçekleştirilmesi için gerekli veri yolu olarak kullanılır. Tablo I.5'te RS, R/ \bar{W} ve DB7-DB0 pinlerinden oluşan HD44780U temelli komut kümesi gösterilmiştir. Bu komut seti uygun şekilde kullanıldığında mikroC ya da benzer başka bir geliştirme platformunun LCD kütüphanesini kullanmadan kendi LCD kontrolünüzü gerçekleştirebilirsiniz.

Tablo I.5 HD44780U komut seti kümesi

Komut	Kod										Açıklama	İşlem süresi (maks.) ($f_{cp} = 270kHz$ olduğunda)	
	RS	R/ \bar{W}	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0			
Clear display	0	0	0	0	0	0	0	0	0	1	Ekranı temizler ve kursörü başlangıç (adres 0) konumuna alır.	1,52 ms	
Cursor home	0	0	0	0	0	0	0	0	0	1	Kursörü başlangıç konumuna alır. DDRAM içeriği değişmez.	1,52 ms	
Entry Mode Set	0	0	0	0	0	0	0	0	1	I/D	S	Kursörün hareket yönünü (I/D) ve görüntü kaydırmayı (S) ayarlar. Bu işlemler veri okuma ve yazma sırasında gerçekleştirilir.	37 μs
Display on/off control	0	0	0	0	0	0	0	1	D	C	B	Tüm ekranın (D) ve kursörün (C) açma/kapama işlemini ve karakter konum kursörünün yanıp/sönme işlemini gerçekleştirir.	37 μs
Cursor/display shift	0	0	0	0	0	0	1	S/C	R/L	*	*	Kursör konumunu ya da tüm ekran hareketi (S/C) ve hareketin yönü (R/L) tanımlanır.	37 μs
Function set	0	0	0	0	1	DL	N	F	*	*	Arayüz veri uzunluğunu (DL), görüntü satır sayısını (N) ve karakter fontunu (F) ayarlar.	37 μs	
Set CGRAM address	0	0	0	1	CGRAM adresi					CGRAM adresini ayarlar. CGRAM verisi bu ayarlardan sonra gönderilir.	37 μs		
Set DDRAM address	0	0	1	DDRAM adresi					DDRAM adresini ayarlar. DDRAM verisi bu ayarlardan sonra gönderilir.	37 μs			
Read busy flag & address counter	0	1	BF	CGRAM/DDRAM adresi					Dâhili işlemin gerçekleştirildiğini ifade eden meşgul bayrağını (BF-Busy Flag) okur ve CGRAM ya da DDRAM adres sayıcısı içeriğini okur.	0 μs			

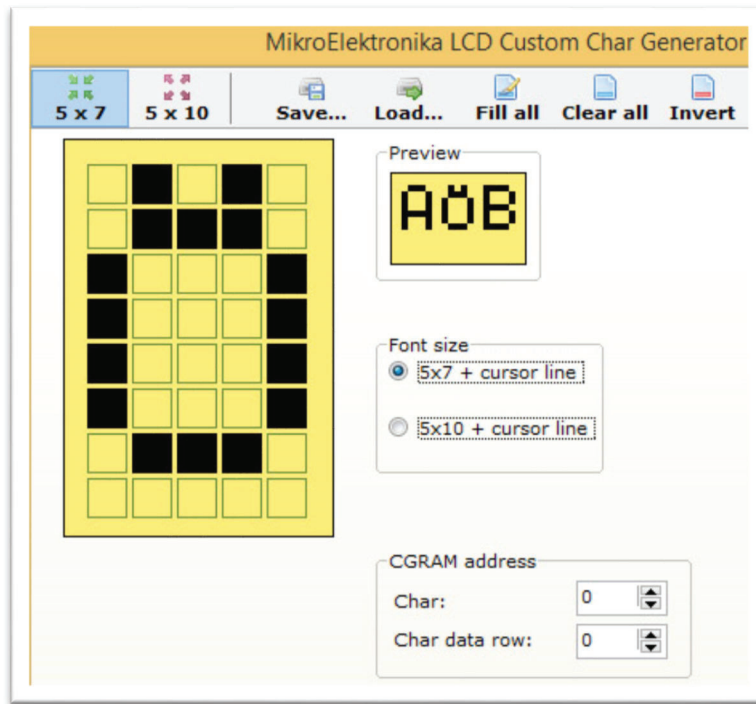
Write CGRAM or DDRAM	1	0	Veriyi yaz	Veriyi CGRAM ya da DDRAM belleklerine yazar.	37 µs
Read from CG/DDRAM	1	1	Veriyi oku	Veriyi CGRAM ya da DDRAM belleklerinden okur.	37 µs

Komut bit açıklamaları:

I/D – 0 = kursör konumunu düşürür, 1 = kursör konumunu yükseltir; **S** – 0 = görüntü kaydırma yok, 1 = görüntü kaydırma; **D** – 0 = görüntü kapalı, 1 = görüntü açık; **C** – 0 = kursör kapalı, 1 = kursör açık; **B** – 0 = kursör yanıp/sönme kapalı, 1 = kursör yanıp/sönme açık; **S/C** – 0 = kursörü hareket ettir, 1 = görüntüyü kaydır; **R/L** – 0 = sola kaydır, 1 = sağa kaydır; **DL** – 0 = 4-bit arayüz, 1 = 8-bit arayüz; **N** – 0 = 1/8 ya da 1/11 görev çevrimi (1 satır), 1 = 1/16 görev çevrimi (2 satır); **F** – 0 = 5x8 nokta matris, 1 = 5x10 nokta matris; **BF** – 0 = komut kabul edilebilir, 1 = dâhili işlem var, bu sırada veri okuma ya da yazma yapılamaz.

Not: HD44780U çipsetinin donanımsal özellikleri, karakter üretimi ve yazdırma/okuma işlemleri hakkında daha ayrıntılı bilgi için veri kılavuzunu incelemeniz tavsiye edilir.

HD44780 ile 5x8 çözünürlüğünde özel karakter üretmek ve ekranda yazdırmak için Şekil I.11'deki karakter üretici matrisinden yararlanılabilir. Mikroelektronika programının “Tools→LCD Custom Character” menüsünden özel karakter elde etmenize yardımcı olacak aracı çağırabilirsiniz. 5x8 matris hücresine sahip ekranların 8. satırı kursör imleci olarak kullanılır. Bu nedenle karakterlerin ilk 7 satırı kapsayacak şekilde oluşturulmasında fayda vardır.



Şekil I.11 LCD karakter üretici

Şekildeki örnekte “Ö” harfi oluşturulmuştur. Her sütundaki siyah hücreler lojik-1 değerine karşılık gelir. Dolayısıyla verilen örnek için {0b01010, 0b01110, 0b10001, 0b10001, 0b10001, 0b10001, 0b01110, 0b00000} ikilik kod dizisi elde edilir. Program düzenleyicisinde bu karakterin HD44780 sürücüsüne uygun olarak hazırlanmasını sağlayacak mikroC kod bloğu Tablo I.6’da verilmiştir.

Tablo I.6 LCD ekran için özel karakter üretilmesini sağlayan kod bloğu

```
char buyuk_o[] = {10,0,14,17,17,17,14,0};
void CustomChar(char pos_row, char pos_char) {
    char i;
    Lcd_Cmd(64);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(butuk_o[i]);
    Lcd_Cmd(LCD_RETURN_HOME);
}
```

```
Lcd_Chr(pos_row, pos_char, 0);  
}
```

HD44780U sürücüsü için 8 adet isteğe bağlı 5x8 matris karakter tasarlanabileceği söylenmişti. Bu karakterlerin yer alacağı CGROM adresi desimal 64 değerinden başlar. Her karakter için 8 baytlık alan kullanıldığından dolayı bir sonraki karakter adresi 72'den devam eder ve 120 adres değerinde son bulur. Bu adreslerden karakterlerin çağrılıp yazdırılması için sırasıyla 0,1,2,3,4,5,6,7 değerleri kullanılır. İlgili adreslere karşılık gelen yazdırma numaraları aşağıdaki gibidir:

<u>Adres</u>	<u>Sıra</u>
64	0
72	1
80	2
88	3
96	4
104	5
112	6
120	7

Lcd_Cmd(Adres) komutuyla ilgili adres alanına gidilir. Ardından 8'lik döngü ile her bir satır için 8 farklı ikilik kod Lcd_Chr_Cp(karakter[i]) komutuyla adres alanına yazdırılır. Adres alanına kaydedilen karakterin LCD ekranına yazdırılması için Lcd_Chr(satir, sutun, sıra) komutu kullanılır.

16 - USB Donanımı ve Protokolü

Tak ve çalıştır aygıtlarına bağlantı standardı getirmek amacıyla çok sayıda firmanın bir araya gelerek ortaya koydukları bir seri haberleşme protokolüdür. Evrensel Seri Yolu (Universal Serial Bus) olarak ismi duyurulan bu teknolojinin günümüze kadar çeşitli versiyonları çıkmıştır. Gelişim kronolojisini kısaca aşağıdaki gibi verebiliriz:

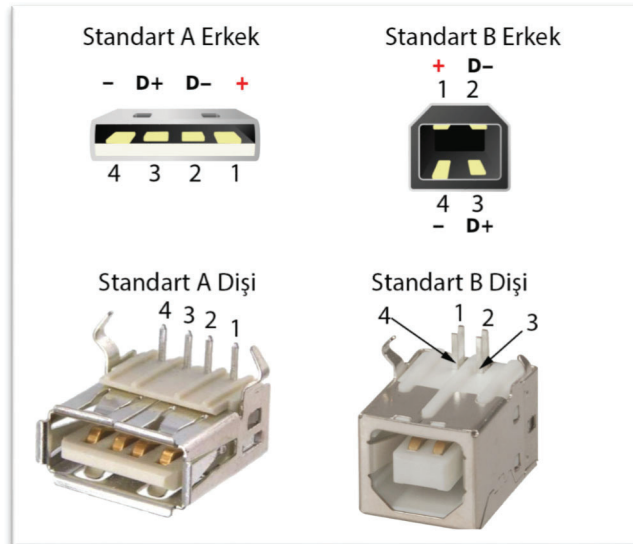
- USB 1.x: Düşük bant genişliğinde 1,5 Mbit/s, yüksek bant genişliğinde 12 Mbit/s hızlarında iletişim gerçekleştiren ilk versiyonu USB 1.0 olarak sürülmüştür. 1.1 sürümüyle birlikte USB aygıtları yaygınlaşmıştır.
- USB 2.0: İletişim hızının 480 Mbit/s seviyesine çıkartılmış olan sürümdür. Tümleşik USB donanımına sahip pek çok 8-bit mikrodenetleyicinin de kullandığı standarttır.
- USB 3.x: SuperSpeed (5 Gbit/s), SuperSpeed+ (10 Gbit/s) ve SuperSpeed+ (20 Gbit/s) hızlarını destekleyen sürümdür.

Bu sürümleri birbirinden ayırmak için Şekil I.12’de gösterildiği gibi farklı logolar kullanılır.



Şekil I.12 USB sürümlerine göre logolar

Piyasada farklı USB priz (soket) ve konnektör türüne sahip aygıtlar vardır. Şekil I.13’te mikrodenetleyici temelli uygulamalarda da sıklıkla tercih ettiğimiz ve yaygın olarak kullanılan standart A ve Standart B erkek/dişi konnektör türleri gösterilmiştir.







Şekil I.13 Standart A ve Standart B USB erkek konnektörleri

Mini A, Mini B, Mikro A, Mikro B, Mikro AB ve Full-duplex C türü gibi çeşitli taşınabilir cihazlar ve video aygıtları için farklı konektör türleri mevcuttur.

Elektriksel Özellikleri:

USB sinyalleri, bir çift burgulu veri kablosuyla diferansiyel fark sinyali tekniğine göre işlev gören, $\pm 15\%$ toleranslı 90Ω 'luk karakteristik empedansı olan kablo üzerinden taşınır.

Tablo I.7 USB besleme ve veri bacakları

USB konektörlerde besleme ve veri bacakları	
Pin 1	 V_{BUS} (+5 V)
Pin 2	 Data-
Pin 3	 Data+
Pin 4	 Şase

USB donanımı USB alıcı aygıtlarını $\pm 5\%$ toleransla 5V'ta besler. Düşük güç aygıtları (USB fare ve klavye, vb.) en fazla 1 birim yük çekebilir (USB 2.0'da 1 birim yük 100 mA, USB 3.0'da ise 150 mA'dir). Tüm aygıtlar, yapılandırılmamış olarak başlatılırken düşük güçlü cihazlar olarak davranmalıdır.

Yüksek güçlü aygıtlar (2.5 inç USB HDD, vb.), USB 2.0'da en az 1 birim yük ve en çok 5 birim yük akım (500mA) çekerler. SuperSpeed aygıtlar (USB 3.0) içinse en fazla 6 birim yük (900 mA) akım çekilebilir.

USB 1.2 standardını kullanan batarya şarj sistemlerinden 1,5A akım çekilebilir.

Düşük hız (Low Speed-LS) ve tam hız (Full Speed-FS) modları, D+ ve D- etiketli tek bir veri üzerinden yarım dubleks olarak haberleşir. İletilen sinyaller için; 0.0 V – 0.3 V aralığı lojik-0, 2.8 V – 3.6V aralığı lojik-1 seviyesindedir. Sinyal hatları sonlandırılmaz.

Yüksek hız (High Speed - HS) mod da aynı kablo çiftini kullanır, fakat farklı bir elektriksel yöntem kullanır. Düşük sinyal voltajı tekniği denen bu yöntemde; -10 mV – 10 mV aralığı lojik-0, 360 mV – 440 mV aralığı lojik-1 seviyesidir.

Süper hız (SuperSpeed-SS) modunda ilave iki çift kalkanlı burgulu (twisted-pair) kablo vardır. Bunlar full-dubleks süper hız çalışma için tahsis edilmişlerdir. Eski yarım dubleks hatlar korunmuş ve uyumluluk devam ettirilmiştir. Şekil I.14'te 3.0 nesli standart A ve standart B tipi konektörler gösterilmiştir.

USB 3.0 B Tipi Bacak Bağlantıları ve Donanımı

Pin	Signal Name	Description
1	VBUS	Kırmızı
2	D-	Beyaz
3	D+	Yeşil
4	GND	Siyah
5	StdA_SSTX-	Mavi
6	StdA_SSTX+	Sarı
7	GND_DRAIN	ŞASE
8	StdA_SSRX-	Mor
9	StdA_SSRX+	Turuncu
Kasa	Kalkan	Konnektör kasası

USB 3.0 B tipi konnektör



USB 3.0 İLETİŞİM KABİLİYETİ

USB 3.0 hızı tam dubleks iletişim kabiliyeti, iki hattın veri gönderimi ve iki hattın da veri alımı için tahsis edilmesi sonucu mümkün olmaktadır.

USB 3.0 B

USB 3.0 A Tipi Bacak Bağlantıları ve Donanımı

Pin	Signal Name	Description
1	VBUS	Kırmızı
2	D-	Beyaz
3	D+	Yeşil
4	GND	Siyah
5	StdA_SSTX-	Mavi
6	StdA_SSTX+	Sarı
7	GND_DRAIN	ŞASE
8	StdA_SSRX-	Mor
9	StdA_SSRX+	Turuncu
Kasa	Kalkan	Konnektör kasası

USB 3.0 A tipi konnektör



USB 3.0 İLETİŞİM KABİLİYETİ

USB 3.0 hızı tam dubleks iletişim kabiliyeti, iki hattın veri gönderimi ve iki hattın da veri alımı için tahsis edilmesi sonucu mümkün olmaktadır.

USB 3.0 A

Şekil I.14 USB 3.0 standart A ve standart B tipi konnektörler

SuperSpeed+ (SS+) artırılmış veri hızını (Gen 2x1 modu) ve/veya C-tipi konektörde (Gen 1x2 ve Gen 2x2 modları) ek bir hat kullanır.

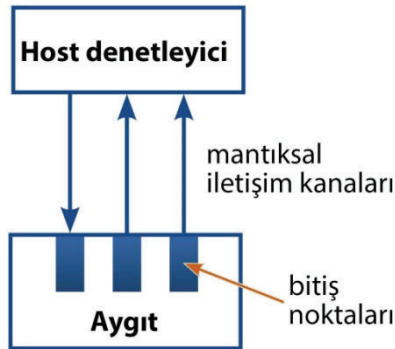
USB Topolojisi ve İletişim Protokolü

Bir USB sistem, katmanlı yıldız topolojisine sahip bir ya da daha fazla alıcı portlu ve çok sayıda çevresel aygıtlı bir yöneticiden (host) oluşmaktadır. Beş katmana kadar izin verilen ek USB hub'ları sisteme dahil olabilir. Bir USB yönetici çok sayıda denetleyiciye sahiptir. Her biri bir ya da daha fazla porta sahiptir. USB denetleyicileri bilgisayarınızın aygıt yöneticisinden görmeniz mümkündür. PIC mikrodenetleyicili bir USB donanımını da bilgisayarınızın uygun bir USB portuna bağladığınızda HID aygıtı olarak aygıt yöneticisinde görülecektir.

Tek bir ana yöneticiye 127 kadar aygıt bağlanabilir. USB aygıtları hub'lar üzerinden seri olarak bağlıdır. Yönetici denetleyicisi olarak tasarlanmış bir hub, kök hub (root hub) olarak adlandırılır.

Bir USB aygıtı çok sayıda mantıksal alt aygıtlardan oluşabilir. Örneğin, webcam gibi tümleşik bir aygıt video ve ses donanımları gibi birden fazla özelliğe sahip olabilir. Bu her mantıksal alt aygıt için yönetici (host) tarafından farklı bir adres tahsis edilmektedir ve tüm mantıksal aygıtlar fiziksel USB kablosuna bağlı yerleşik bir hub'a bağlıdır.

USB ağındaki aygıtlar doğrudan birbirleriyle haberleşemez. Haberleşme yönetici (host) üzerinden gerçekleşir.



Şekil I.15 USB aygıtında bitiş noktaları (endpoints)

USB aygıtı mantıksal iletişim kanalları üzerine kurulmuştur. Bir kanal, ana yöneticiden mantıksal bir elemana bağlantı anlamına gelir. Bu mantıksal elemana bitiş noktası (endpoint) denir. Çünkü kanallar bitiş noktalarına karşılık gelir. Bu terimler karşılıklı olarak birbirlerinin yerine kullanılabilir. Bir USB aygıtı her ne kadar 16 giriş, 16 çıkış olmak üzere 32 bitiş noktasına sahip olabilese de, pratikte bu kadarını kullanan bir aygıt yoktur. Bir bitiş noktası, ilk kullanıma hazırlanma sırasında (fiziksel bağlantı sonrası periyot “enumeration-numaralandırma” olarak adlandırılır) aygıt tarafından tanımlanır ve numaralandırılır. Böylece nispeten kalıcı bir kanal açılabilir ve kapatılabilir .

İki tür kanal vardır: Akış ve mesaj kanalı. Mesaj kanalı, veri transferlerini kontrol etmek için kullanılan iki-yönlü bir kanaldır. Mesaj kanalları genellikle aygıtta kısa ve basit komutlarla durum kontrolü gerçekleştirmek için kullanılır. Akış kanalı ise, verileri *eşzamanlı*, *kesme* ya da *toplu transfer* yöntemleriyle ileten tek yönlü bir bitiş noktasına bağlanan tek yönlü bir kanaldır.

USB TANIMLAYICILARI

Tüm USB aygıtları aygıtın farklı özelliklerini açıklayan bir tanımlayıcı hiyerarşisine sahiptir. Üretici kimliği, aygıtın versiyonu, desteklenen USB versiyonu, aygıtın ne olduğu, güç gereksinimleri, uçnokta sayısı ve bu gibi bilgiler cihaz içindeki EEPROM belleğinde ya da mikrodenetleyicinin ilgili belleğinde yer alır.

En yaygın USB tanımlayıcıları şunlardır:

- Aygıt tanımlayıcıları
- Yapılandırma tanımlayıcıları
- Arayüz tanımlayıcıları
- HID (Human Interface Device – İnsan Arayüz Aygıtı) tanımlayıcıları
- Bitiş noktası tanımlayıcıları

Tanımlayıcılar hiyerarşik bir dizilime sahiptir ve aygıt tanımlayıcısı en üstte yer alır. Aygıt tanımlayıcısında, USB organizasyonu tarafından belirlenen satıcı kimliği (Vendor IDentification - VID) ve üretici tarafından tanımlanan ürün kimliği (Product IDentification - PID) bilgileri de yer alır.

Her tanımlayıcının hiyerarşik sırada birtakım görevleri vardır. Bilgisayara bağlayacağınız uygulama kartı hakkında işletim sistemine bilgilendirme bu tanımlayıcılar ile yapılır. “mikroC PRO for PIC” programının Tools → HID Terminal komutuyla erişilen bir HID tanımlayıcı oluşturma programı vardır. Bu araç yardımıyla mikrodenetleyicili kartınıza masaüstü yazılımına ihtiyaç duymadan veri göndermeniz de mümkündür. Şekil I.16’da HID aracı üzerinden örnek bir tanımlayıcının oluşturulması gösterilmiştir.

HID aracının “Descriptor-Tanımlayıcı” sekmesine gelindiğinde satıcı ve üretici isimleriyle birlikte veriyolunun çekeceği maksimum akım değeri, bitiş noktası havuzu için kesme akımı, giriş ve çıkış rapor uzunlukları bilgileri de değiştirilebilmektedir. Genel olarak donanımsal niteliklerle ilgili bilgileri varsayılan değerlerinde tutun. “Save descriptor” butonuna tıkladığınızda, varsayılan olarak “USBdsc.c” ismiyle yukarıda isimleri verilmiş olan tüm tanımlayıcıları kapsayacak şekilde bir dosya oluşturulur. Programınızın derlenmesi sırasında bu tanımlayıcı dosyasının çalışmaya “#include” komutuyla dahil edilmesi gerekir.

17 - 7447 ve 7448 BCD 7-Segment Gösterge Sürücüleri

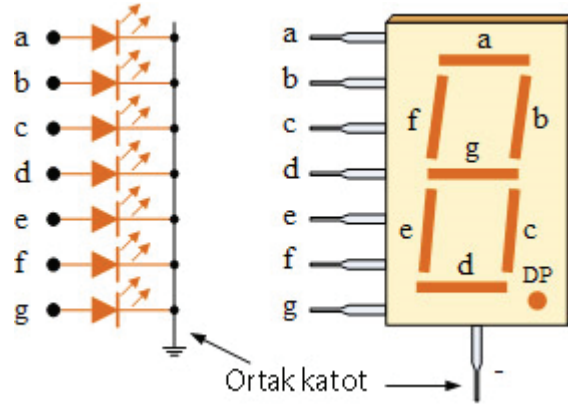
Sayısal bir kod çözücü entegresi (IC), sayısal bir değeri başka bir sayısal değere dönüştüren aygıttır. İkilik Kodlu Onlu sayı sistemini (Binary Coded Decimal) 7-segment gösterge sürücüsüne dönüştürme işlemi sayısal kod çözmede yaygın olarak karşımıza çıkan bir uygulamadır.

7-segment LED ya da LCD tip ekranlar; sayılar, harfler, ya da diğer alfanümerik karakterlerden oluşan bilginin gösterilmesinde yaygın olarak tercih edilirler. 7-segment göstergeler 7 adet bağımsız renkli LED'din tek bir paket altına alınmış halidir. LED'lerin uygun şekilde sürülüp, 0-9 arası sayı ve A-F arası HEX karakterlerinin gösterilmesi için BCD – 7-segment gösterge sürücüleri kullanılır.

7-segment göstergeler içlerindeki LED'lerin ortak anotlu ya da ortak katotlu olmasına göre isimlendirilirler. Ortak anotlu göstergeleri sürmek için çıkışları terslemeli 7447 entegresi kullanılırken, ortak katotlu göstergeleri sürmek için 7448 entegresi kullanılır.

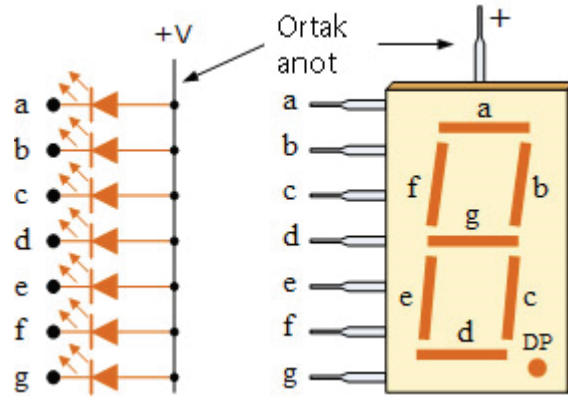
Standart 7-segment LED göstergesi genellikle 8 giriş bağlantıya sahiptir. Her bir LED segmenti için bir pin ve ortak terminal için de bir pin bulunur. Ayrıca nokta sinyali için de ek bir pini bulunan göstergeler vardır. Piyasada yaygın olarak 5 + 5 şeklinde her iki tarafında 5'er adet pin bulunan 7-segment göstergeler bulunur. Her iki tarafın orta kısmında yer alan pinler ortak bağlantı pinidir. Piyasada farklı boyutlarda ve farklı karakter tiplerine uygun geniş bir yelpazede 7-segment gösterge bulunsun da, çalışma ilkesi bakımından iki tip 7-segment gösterge kullanılır.

1. Ortak katot gösterge (Common Cathode Display - CCD): Bu göstergede LED'lerin tüm katot bağlantıları ortak bir terminale bağlanmıştır. Bu terminal devre bağlantılarında şase hattına bağlanmalıdır. Boşta kalan diğer pinler LED'lerin anot bağlantılarıdır. Hangi segment yakılmak istenirse ilgili segmentin bacağına lojik-1 (High) uygulanmalıdır.



Şekil I.17 7-segment ortak katot göstergenin yapısı

2. Ortak anot gösterge (Common Anode Display - CAD): Bu göstergede LED'lerin tüm anot bağlantıları ortak bir terminale bağlanmıştır. Bu terminal devre bağlantılarında V_{CC}/V_{DD} hattına bağlanmalıdır. Boşta kalan diğer pinler LED'lerin katot bağlantılarıdır. Hangi segment yakılmak istenirse ilgili segmentin bacağına lojik-0 (Low) uygulanmalıdır.



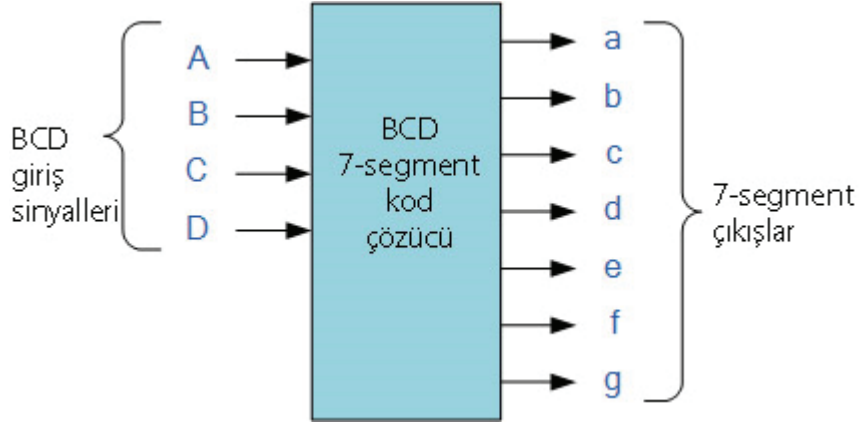
Şekil I.18 7-segment ortak anot göstergenin yapısı

7447 ve 7448 kod çözücü ve 7-segment gösterge sürücü entegreleri 4 BCD giriş ve 7 çıkışa sahiptir. Bu entegreler onlu sayı tabanında işlem yaptıkları için yalnızca 0-9 arası rakamların bilgisini sürebilir. Bu nedenle bu entegrelerle onaltılı sayı tabanının A-F karakterlerini sürdürmezsiniz. BCD – 7-segment gösterge sürücüsünün doğruluk tablosu Tablo 3.1’de gösterilmiş olup Tablo I.9’da BCD kod tablosu verilmiştir.

Tablo I.9 BCD kod tablosu

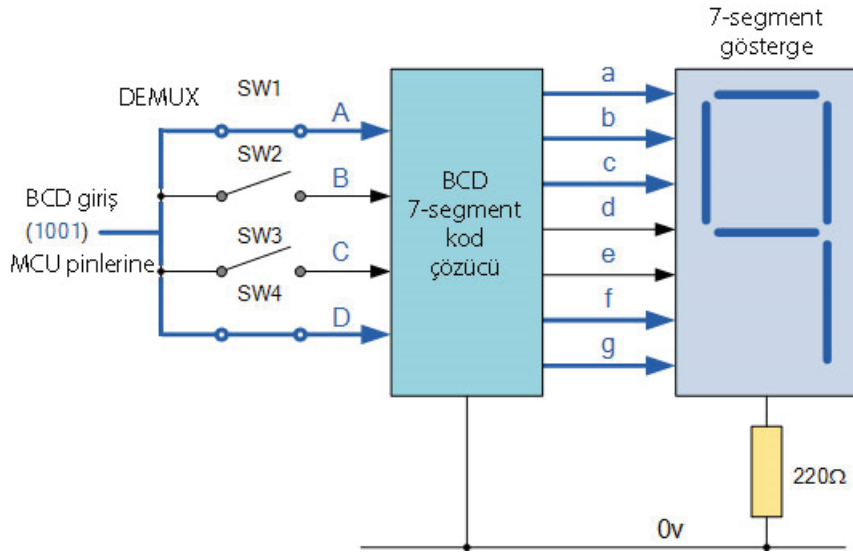
Onlu	İkili Değer Karşılığı				BCD
	8	4	2	1	
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	Geçersiz
11	1	0	1	1	Geçersiz
12	1	1	0	0	Geçersiz
13	1	1	0	1	Geçersiz
14	1	1	1	0	Geçersiz
15	1	1	1	1	Geçersiz

Şekil I.19’da 7447 ve 7448 entegrelerinin BCD giriş ve 7-segment çıkış pinleri gösterilmiştir. 7-segment gösterge girişlerine ya da ortak terminale direnç bağlantısı unutulmamalıdır.



Şekil I.19 BCD'den 7-segment'e kod çözücü

Şekil I.20'de BCD – 7-segment sürücü bağlantısı ve örnek olarak '9' sayısının elde edilmesi gösterilmiştir.



Şekil I.20 BCD - 7-segment kod çözücü örnek uygulaması

18 – LM1117 LOW-DROPOUT VOLTAJ REGÜLATÖRÜ

Mikrodenetleyici uygulamalarındaki en önemli sorun kararlı bir beslemenin sağlanması sırasında yaşanan sıkıntılardır. Mikrodenetleyicilerin çekirdek işlemcisinin, dış dünya ile bağlantı sağlayan çevre birimlerinin ve o çevre birimlerine bağlı algılayıcıların kararlı çalışabilmesi için gerekli besleme hatasız şekilde sağlanmalıdır.

F serisi çoğu PIC mikrodenetleyicisi düşük çalışma aralığına da sahiptir. Her ne kadar 2 V – 5.5 V aralığında çalışabilseler de sabit kaynaktan beslenmeleri durumunda ideal olarak 5V ile çalışmaları tavsiye edilir. LF serisi XLP PIC mikrodenetleyicilerinde ise çalışma aralığı 1.8 V – 3.6 V seviyesine iner. Mikrodenetleyicili kartların üzerinde çalışması gereken ve 3.3 V ya da 5 V DC gerilime ihtiyaç duyan algılayıcılar olabilir. İşte tüm bu devre bileşenleri için bazı durumlarda 3.3 V, 5 V ve hatta değişebilir aralıkta DC beslemeye ihtiyaç duyulur. LDO özellikli gerilim regülatörleri maksimum çıkış akımlarını düşük çıkış voltajında bile yüke sunabilir.

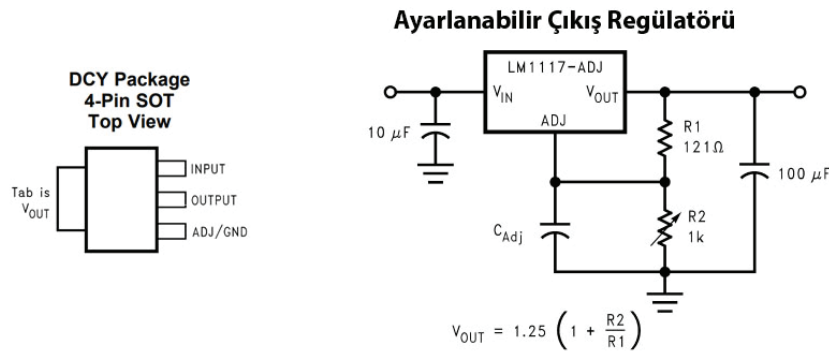
Teknik özellikleri şu şekildedir:

- 1.8 V, 2.5 V, 3.3 V, 5 V ve ayarlanabilir versiyonları mevcuttur. Ayarlanabilir olanda çıkış gerilimi 1.25 V – 13.8 V arasında değiştirilebilir.
- Maksimum giriş gerilimi 20V'tur. Giriş ve çıkış arasındaki farkın çok yüksek olmaması istenir. Bu fark LM1117 üzerinde fazladan ısı yayınımına yol açar.
- Akım sınırlama ve termal koruma özelliklidir.
- Çıkış akımı maksimum 800 mA'dir. Bu akımı çıkış 1.2 V'a düşse bile yüke verebilir.
- Sıcaklık aralığı 0 °C – 125 °C, I modelinde ise -40 °C dereceye kadar çalışabilir

Kullanım alanları:

- DC – DC dönüştürmede son regülatör katı olarak kullanılır.
- Batarya şarj donanımı olarak kullanılabilir.
- Portatif ve batarya ile çalışan aygıtlar için kullanılabilir.

Şekilde LM1117'nin en sık kullanılan SOT-223 kılıf yapısı ve ayarlanabilir regülatör olarak kullanılması istendiğinde kurulacak devre düzeni gösterilmiştir. C_{ADJ} kondansatörü isteğe bağlı olarak bağlanır. Kullanılması durumunda çıkış dalgalanması zayıflatmasına katkı sağlar. 10 uF'lık bir kondansatör bağlayabilirsiniz.



Şekil 1.21 LM1117 SOT-223 kılıfı ve ayarlanabilir voltaj devresi uygulaması