

**T.C.
MİLLÎ EĞİTİM BAKANLIĞI**

BİLİŞİM TEKNOLOJİLERİ

MİKRODENETLEYİCİ

Ankara, 2014

- Bu modül, mesleki ve teknik eğitim okul/kurumlarında uygulanan Çerçeve Öğretim Programlarında yer alan yeterlikleri kazandırmaya yönelik olarak öğrencilere rehberlik etmek amacıyla hazırlanmış bireysel öğrenme materyalidir.
- Millî Eğitim Bakanlığınca ücretsiz olarak verilmiştir.
- PARA İLE SATILMAZ.

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. MİKRODENETLEYİCİ VE DONANIM SEÇİMİ	3
1.1. Mikrodenetleyici	3
1.1.1. Mikroişlemci	3
1.1.2. Mikrodenetleyici	3
1.1.3. Programlama İçin Gereken İhtiyaçlar	4
1.1.4. Mikrodenetleyici Çeşitleri	4
1.1.5. Mikrodenetleyici Yapısı	4
1.1.6. Mikrodenetleyici Bellek Çeşitleri	5
1.2. Mikrodenetleyicide Yazım Dili	6
1.2.1. Sayıların Tipi	6
1.2.2. Binary Dijit	6
1.2.3. Hexadesimal Dijit	7
1.2.4. Binary, Desimal ve Hexadesimal Sayıların Dönüşümü	7
1.2.5. Komutların Yazılış Biçimi	8
1.3. Mikrodenetleyici Yapısı	12
1.3.1. Besleme Gerilimi	12
1.3.2. Clock Düzeni ve Osilatör Çeşitleri	12
1.3.3. Reset Uçları ve Reset Devresi	13
1.3.4. Komut Akışı	13
1.3.5. Yazmaçlar	14
1.3.6. Program Belleği	16
1.3.7. Veri Belleği	17
1.3.8. I/O Portları	17
UYGULAMA FAALİYETİ	18
ÖLÇME VE DEĞERLENDİRME	19
ÖĞRENME FAALİYETİ-2	21
2. MİKRODENETLEYİCİ PROGRAMLAMA KARTI YAPIMI	21
2.1. Programlama Kartı	21
2.1.1. Programlama Devresinin Yapımı	21
2.1.2. Bağlantı Kablosunun Yapımı	24
UYGULAMA FAALİYETİ	25
ÖLÇME VE DEĞERLENDİRME	27
ÖĞRENME FAALİYETİ-3	28
3. MİKRODENETLEYİCİ PROGRAMLAMA EDITÖRÜNÜN KULLANIMI	28
3.1. Mikrodenetleyici Program Editörünün Kurulumu	28
3.1.1. Derleyici Editörün Yüklenmesi	28
3.1.2. Editörün Ayarlarının Yapılması	29
3.1.3. Editörün Özellikleri	30
3.1.4. Denetleyiciye Yüklenecek Programın Derlenmesi ve Denenmesi	32
3.1.5. Denetleyiciye Programın Yazdırılması	35

3.1.6. Denetleyicinin Denenmesi	37
UYGULAMA FAALİYETİ	38
ÖLÇME VE DEĞERLENDİRME	39
MODÜL DEĞERLENDİRME	40
CEVAP ANAHTARLARI	41
KAYNAKÇA	43

AÇIKLAMALAR

ALAN	Bilişim Teknolojileri
DAL/MESLEK	Bilgisayar Teknik Servis
MODÜLÜN ADI	Mikrodenetleyici
MODÜLÜN TANIMI	Bu modül, mikrodenetleyici teknolojisini öğreten, mikrodenetleyici programlama kartı yapma ve mikrodenetleyiciye yazılım yükleme becerilerinin kazandırıldığı bir öğrenme materyalidir.
SÜRE	40/24
ÖN KOŞUL	Bu modülün ön koşulu yoktur.
YETERLİK	Mikrodenetleyici programlama kartını yapmak
MODÜLÜN AMACI	Genel Amaç Bu modülle gerekli ortam sağlandığında mikrodenetleyici montajını yapabileceksiniz. Amaçlar <ol style="list-style-type: none">1. İhtiyaca uygun olarak mikrodenetleyici ve donanım seçimini yapabileceksiniz.2. Mikrodenetleyici programlama kartını hazırlayabileceksiniz.3. Mikrodenetleyici programlama editörünü kullanabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: Elektronik uygulamaları laboratuvarı Donanım: Mikrodenetleyici katalogları, elektronik devre elemanları, elektronik malzeme katalogları, ölçü aleti, baskı devre ve lehimleme araç gereçleri, mikrodenetleyici programlama editörü, bilgisayar
ÖLÇME VE DEĞERLENDİRME	Modül içinde yer alan her öğrenme faaliyetinden sonra verilen ölçme araçları ile kendinizi değerlendireceksiniz. Öğretmen, modül sonunda ölçme aracı (çoktan seçmeli test, doğru-yanlış testi, boşluk doldurma, eşleştirme vb.) kullanarak modül uygulamaları ile kazandığınız bilgi ve becerileri ölçerek sizi değerlendirecektir.

GİRİŞ

Sevgili Öğrenci,

Teknolojinin hızla ilerlediği bu zamanda, günümüz modern toplumları bilgisayar teknolojisini hayatın her alanında uygulamaya geçirmiş ve bunlardan azami fayda sağlamaya çalışmaktadır. Özellikle endüstriyel otomasyonda bilgisayarlar önemlidir.

İşte endüstrideki bu otomasyon işlemlerinin önemli bir bölümü mikrodenetleyiciler ile yapılmaktadır. Bu noktada güncelliğini koruyan mikrodenetleyiciler ile endüstride çok basit ama etkili devreler üretmek bizim için önemli hedeflerden biridir.

Modül sonunda edineceğiniz bilgi ve beceriler ile mikrodenetleyicileri tanıyacak onları programlamayı, programlarken kullanılacak editörleri ve yazılım tabanlı simülatörleri öğreneceksiniz. Yine aynı şekilde mikrodenetleyiciyi çalışır hâle getirmek için gerekli donanımları ve bu küçük devreler içine hazırlanan yazılımların yüklenmesi için gerekli olan bilgiye ulaşacaksınız.

ÖĞRENME FAALİYETİ-1

AMAÇ

İhtiyacınıza uygun mikrodenetleyici ve donanım seçimini yapabileceksiniz.

ARAŞTIRMA

- Günümüzde yaygın olarak kullanılan mikrodenetleyici çeşitlerini araştırınız.
- Mikrodenetleyicilerin tarihsel gelişim sürecini araştırınız ve elde ettiğiniz bilgileri sınıf içerisinde arkadaşlarınızla tartışınız.
- Mikrodenetleyicilerin fiziksel yapısını inceleyiniz ve mikrodenetleyici kullanılarak yapılan devrelerin diğer devrelere göre fiziksel yapısını karşılaştırınız

1. MİKRODENETLEYİCİ VE DONANIM SEÇİMİ

1.1. Mikrodenetleyici

1.1.1. Mikroişlemci

Mikroişlemciler, bilgisayar programlarının yapmak istediği tüm işlemleri yerine getirir. Başka bir ifadeyle CPU (Central Processing Unit-Merkezi İşlem Birimi) olarak adlandırılır.

Bir mikroişlemci işlevlerini yerine getirebilmek için yardımcı elemanlara ihtiyaç duyar. Bunlar:

- Input (Giriş) Ünitesi
- Output (Çıkış) Ünitesi
- Memory (Bellek) Ünitesi

Bu üniteler CPU çipinin dışında, sistemin farklı yerlerindedir. Aralarındaki iletişim için veri yolu ve adres yolları denilen iletim hatları kullanılır.

1.1.2. Mikrodenetleyici

Mikroişlemcilerden farklı olarak, giriş-çıkış-bellek ünitelerini tek bir çipte barındıran yapıya mikrodenetleyici (Microcontroller) denir. Mikrodenetleyiciler mikroişlemcilere göre daha basit ve ucuzdur.

Mikrodenetleyici kullanarak oluşturacağınız bir sistemde giriş-çıkış-bellek ünitelerine ve bu üniteler arasındaki iletişim için veri yollarına ihtiyaç duyulmayacağından daha basit ve daha ucuz bir yapı oluşturmak mümkündür.

1.1.3. Programlama İçin Gereken İhtiyaçlar

Bir mikrodenetleyiciyi programlamak için olması gerekenler:

- Kişisel bir bilgisayar
- Seçilen mikrodenetleyiciye uygun programlama kartı
- Komut satırlarının yazılacağı editör
- Mikrodenetleyici programlama yazılımı
- Yazılan programın makine diline çeviren derleyici
- Similasyon yazılımı

1.1.4. Mikrodenetleyici Çeşitleri

Temelde mikroişlemci mimarisi iki çeşittir. Bunlar RISC (ReducedInstruction Set Computer: Azaltılmış komut seti) tabanlı işlemciler ve CISC (ComplexInstruction Set Computer: Karmaşık komut seti) tabanlı işlemcilerdir. Mikrodenetleyicilerde aynı şekilde RISC ve CISC mimarisine göre türleri vardır. Yaygın olarak kullanılan mikrodenetleyiciler mimari açıdan bu iki işlemci sınıfından birine aittir.

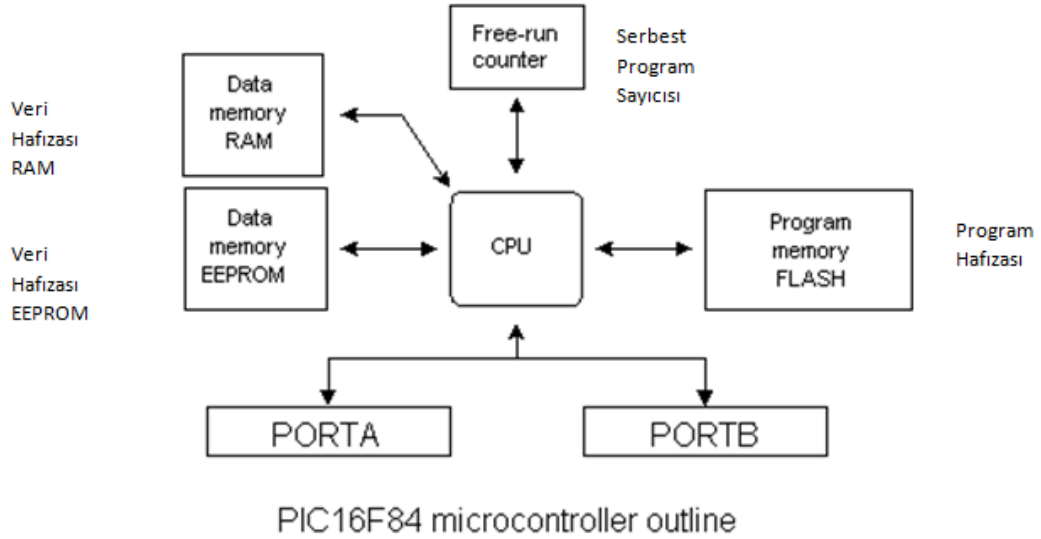
Üretici firma bakımından ise bilinen çok sayıda mikrodenetleyiciler vardır. Bunlar:

- Microchip firmasının PIC mikrodenetleyicileri (RISC),
- Intel firmasının MCS51 (8051) mikrodenetleyicileri (CISC),
- Atmel firmasının AVR mikrodenetleyicileri (RISC),
- Motorola FreeScalemikrodenetleyicileri

ve daha birçok firmanın burada sayamayacağımız kadar çok çeşitleri olan denetleyicileridir.

1.1.5. Mikrodenetleyici Yapısı

Mikrodenetleyicilerin yapıları birbirinden farklı olmakla birlikte temelde çok fazla benzerlikler içerir. Şekil 1.1'de PIC mikrodenetleyicilerinden PIC16F84 içyapısı verilmiştir.



Şekil 1.1: Basit bir mikrodnetleyici blok şeması

Görüldüğü gibi mikrodnetleyicilerde hemen hemen tüm çevre birimler denetleyici içerisinde yer almaktadır. Yapılacak işin niteliğine göre hazırlanacak cihazlarda ek devreler kullanılabilir ancak PIC mikrodnetleyicisi çalışmak için ek hiçbir devreye ihtiyaç duymayacaktır.

1.1.6. Mikrodnetleyici Bellek Çeşitleri

Mikrodnetleyicilerin farklı özellikte program bellekleri bulunur. Bunlar:

- **EPROM** (ErasablePROgrammable Memory – Silinebilir ve programlanabilir bellek)
- **EEPROM** (ElectricallyErasablePROgrammable Memory – Elektriksel olarak silinebilir ve programlanabilir bellek)
- **ROM** (Read Only Memory – Sadece okunabilir bellek)

Her bellek tipinin; hız, fiyat, defalarca kullanmaya yatkınlık gibi faktörler açısından kullanılacağı uygulamaya yönelik avantajları ve dezavantajları vardır.

1.1.6.1. EPROM

EPROM bellek hücrelerine mikrodnetleyici programlayıcı vasıtasıyla elektriksel sinyal uygulayarak kayıt yapılır ve üzerindeki enerji kesilse bile bellek içindeki veri kalıcıdır.

EPROM belleği, üzerindeki program silinip başka bir program yazılacağı zaman EPROM silici denilen özel aygıtlar vasıtasıyla ultraviyole ışını altında bir süre tutmak gerekir.

1.1.6.2. EEPROM

EEPROM bellek hücrelerine mikrodenetleyici programlayıcı vasıtasıyla elektriksel sinyal uygulayarak kayıt yapılır ve üzerindeki enerji kesilse bile bellek içindeki veri kalıcıdır.

EEPROM belleğe, üzerindeki program silinip başka bir program yazılacağı zaman mikrodenetleyici programlayıcısı tarafından elektriksel sinyal gönderilerek silme veya programlama işlemi gerçekleştirilir.

Bu tip belleğe sahip mikrodenetleyiciler EPROM belleğe sahip olanlara göre daha pahalıdır.

1.1.6.3. ROM

ROM belleğe sahip mikrodenetleyicilerin programları üretildikleri fabrikada bir defaya mahsus olarak yazılır. Bu yüzden program geliştirmeye uygun bellek tipi değildir.

Fiyatları diğer bellek tiplerine göre çok düşüktür.

1.2. Mikrodenetleyicide Yazım Dili

1.2.1. Sayıların Tipi

Mikrodenetleyicilerde program geliştirirken üç tür veri tipi kullanılır:

Binary (ikili): 0 ve 1 değeri dışında değer alamaz. En küçük veri tipidir.

Hexadecimal (on altılı): 0-9 arası rakamlar ve A-F arası harflerden oluşan toplam 16 farklı değer alabilmektedir. PIC mikrodenetleyicisinde her bir bellek adresi 8 bit uzunluğunda 2 haneli onaltılı sayıları saklayabilmektedir.

Decimal (onlu): 0-9 arası rakamlardan oluşan 10 farklı değer alabilen veri türüdür.

1.2.2. Binary Dijit

İkili sayı sisteminde ifade edilen sayının her bir basamağı bir binarydigit olarak değerlendirilir. Binarydigitler 0 ya da 1 değeri dışında değer alamazlar. Binarydigitlere kısaca bit adı verilir.

Örneğin PIC mikrodenetleyici için hazırlanmış bir programda şu tür bit komutlarına rastlamak mümkündür:

BCF 0x03, 5 ; 03 Hex adresindeki verinin 5. bitini 0 yap;

1.2.3. Hexadesimal Dijit

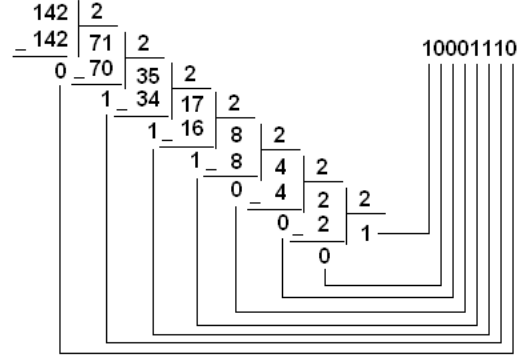
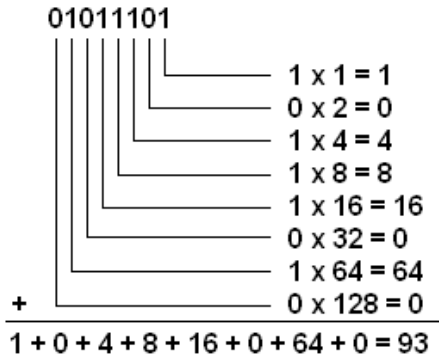
Onaltılı sayı sisteminde ifade edilen verilerin her bir basamağı hexadecimaldigit olarak adlandırılır.

Örneğin; PIC microdenetleyicisinde onaltılı bir sayının kullanımı aşağıdaki şekilde olabilir.

MOVLW 0xA0 ; Akümülatöre (W Yazmacı) A0 onaltılı sayısını yükle.

1.2.4. Binary, Desimal ve Hexadesimal Sayılarının Dönüşümü

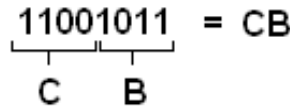
- **Binary – Decimal dönüşümü:** İkili sayı sisteminden onlu sayı sistemine dönüşüm için her bit, bit değeri ile çarpılarak bu çarpımlar toplanmaktadır. (Şekil 1.2). Aynı şekilde onlu sayı sisteminden ikili sayı sistemine geçiş için ise sayı sürekli ikiye bölünerek kalanlar yan yana yazılır. (Şekil 1.3)



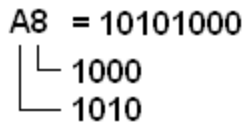
Şekil 1.2: Binary' den Decimal' e dönüşüm

Şekil 1.3: Decimal' den Binary' ye dönüşüm

- **Binary – Hexadecimal dönüşümü:** İkili sayı sisteminden onaltılı sayı sistemine geçişte her dört bitin hex değeri hesaplanarak yan yana yazılır (Şekil 1.4). Aynı şekilde onaltılı sayı sisteminden ikili sayı sistemine geçişte her basamağın Binary değeri hesaplanarak yan yana yazılır (Şekil 1.5).



Şekil 1.4: Binary' den Hexadecimal' e dönüşüm



Şekil 1.5: Hexadecimal' den Binary' ye dönüşüm

- **Decimal – Hexadecimal dönüşümü:** Onlu sayı sisteminden onaltılı sayı sistemine geçişte sayı sürekli onaltıya bölünerek kalanlar yan yana yazılır (Şekil 1.6). Aynı şekilde onaltılı sayıdan onlu sayıya geçiş için her bir basamak kendi değeriyle çarpılarak çarpımlar toplanır. Böylece dönüşüm gerçekleşir (Şekil 1.7).

$$\begin{array}{r|l}
 155 & 16 \\
 \hline
 -144 & 9 \\
 \hline
 11 &
 \end{array}
 \quad
 \begin{array}{l}
 9B \\
 \hline
 \end{array}$$

Şekil 1.6: Decimal'den Hexadecimal'e dönüşüm

$$\begin{array}{r}
 7F2 \\
 \hline
 + \\
 \hline
 2 + 240 + 1792 = 2034
 \end{array}$$

Şekil 1.7: Hexadecimal'den Decimal'e dönüşüm

1.2.5. Komutların Yazılış Biçimi

Her mikrodenetleyici için kendine has derleyiciler bulunmaktadır. Derleyicilerin de kendine has özellik ve yazım kuralları vardır.

PIC mikro denetleyici için en yaygın derleyici olan MPASM derleyicisi aşağıdaki yazım kurallarını getirmiştir:

- Satır başına ; (noktalı virgül) konulduğunda o satır yorum satırı olarak değerlendirilir ve programa herhangi bir etkisi yoktur.
- Genellikle derleyiciler program komutlarının belli bir düzen içerisinde TAB denilen sütunlar hâlinde yazılım koşulunu getirirse de MPASM derleyicisi bu şartı aramamaktadır.
- MPASM derleyicisi için sabit verilerin atandığı bazı sembolik isimlere etiket denir. Etiketlerde şu kurallara uyulmalıdır:
 - Etiketler 1. kolondan itibaren yazılır.
 - Etiketler bir harf ya da _ (altçizgi) ile başlamalıdır.
 - Etiketlerde Türkçe karakter kullanılamaz
 - Bir komut ismi olmamalıdır
 - En fazla 31 karakter uzunluğunda olabilir
 - Büyük/küçük harf duyarlılığı vardır.

- Atama deyimi (EQU): Bu deyim bazı adresleri bazı etiketlere atmak için kullanılır.

PORTB EQU 0x06 ; Bu komutla 06hex adresi PORTB etiketine atanmıştır.

- Sabitler: Bazı komutlarda doğrudan rakamlar kullanılabilir. Burada kullanılan rakamlar birer sabittir.

MOVLW 0x07 ; Bu komutla W yazmacına 7 hex sabit değeri aktarılır.

- ORG deyimi: iki amaç için kullanılabilir. Birincisi program başlangıç adresini belirtmek için ikincisi ise interrupt alt programı başlangıç adresini belirtmek içindir.
- Sonlandırma deyimi: END komutu programı bitirmek içindir. PIC mikrodenetleyicilerindehalt komutu yoktur. Bunun yerine sonsuz döngüler kullanılır.
- Sayı ve karakter yazılış biçimleri:

- Hexadecimal(Onaltılı) Sayılar:
 - 0x03
 - 03
 - 03h
 - h'03'
- Binary (İkili) Sayılar:
 - b'00010101'
- Decimal (Onlu) Sayılar:
 - d'122'
- ASCII Karakterler:
 - 's'

Bu kurallar haricinde PIC mikrodenetleyicilerinde 4 tipte komut yazılışı vardır:

- Byte yönlendirmeli komutlar.
- Bit yönlendirmeli komutlar.
- Sabit işleyen komutlar.
- Kontrol ve akış komutları.

➤ **Byte Yönlendirmeli Komutlar:**

KOMUT YAZMAÇ, HEDEF

MOVF 0x03,0 : Bu komut 03 adresindeki yazmaç içeriğini akümülatöre (W yazmacı) kopyalar.

MOVF 0x03,1 : Bu komut 03 adresindeki yazmaç içeriğini yine 03 adresine kopyalar.

Komut **yazmaç (register)** ile yaptığı işlem sonucunu **hedef(destination)** ile belirlenen hedefe yazar. Hedef iki değer alabilir 0 ya da 1. 0 değeri W yazmacını temsil eder. 1 değeri ise komutta belirtilen yazmacı temsil eder.

➤ **Bit Yönlendirmeli Komutlar:**

KOMUT YAZMAÇ, BIT

BCF STATUS, 5 ;Bu komut STATUS yazmacının 5. bitini 0 yapar. Komut **yazmaç(register)** ile belirtilen yazmacın **bit (hane)** ile belirtilen basamağı ile işlem yapar.

➤ **Sabit İşleyen Komutlar:**

KOMUT SABIT

ADDLW b'01011010' ;Bu komut verilen binary (ikili) sabit sayısını W yazmacı ile toplar. Komut verilen sabit ile işlem yapmaktadır.

➤ **Kontrol Komutları:**

KOMUT ETIKET

GOTO DONGU :Bu komut programın akışını DONGU ile belirtilen etikete yönlendirir. Komut verilen etikete doğru program akışını yönlendirmektedir.

➤ **PIC Mikrodenetleyicilerin Komut Seti:**

Aşağıdaki tabloda PIC mikrodenetleyicilerinde kullanılan komutlar verilmiştir. Tabloda şunlara dikkat ediniz.

ADDWF f, d

Burada ADDWF komutun kendisidir. Verilen parametrelerden f herhangi bir file register'ı temsil etmektedir. Yani işlemin yapılacağı yazmacı belirtmiş oluruz. d simgesi ise işlem yapıldıktan sonra sonucun nerede saklanacağını belirler. 0 ise sonuç W yazmacında 1 ise f ile belirtilen yazmaçta saklanır. Komutun kaç saat çevriminde işlendiği ve hangi bayrakları etkilediği zaten Tablo 1.1'de açıkça belirtilmiştir.

Komutlar		Açıklamalar	Çevrim	Bayraklar	Notlar
ADDWF	f, d	W ve f'i topla	1	C,DC,Z	1,2
ANDWF	f, d	W ile f'i mantıksal VE işlemine tabi tut	1	Z	1,2
CLRF	f	f'i temizle	1	Z	2
CLRWF	-	W'yi temizle	1	Z	
COMF	f, d	f'in tersini al	1	Z	1,2
DECF	f, d	f'i azalt	1	Z	1,2
DECFSZ	f, d	f'i azalt, sonuç 0 ise zıpla, sonraki komutu geç	1(2)	Hiçbiri	1,2,3
INCF	f, d	f'i arttır	1	Z	1,2
INCFSZ	f, d	f'i arttır, sonuç 0 ise zıpla, sonraki komutu geç	1(2)	Hiçbiri	1,2,3
IORWF	f, d	W ile f'i mantıksal VEYA işlemine tabi tut	1	Z	1,2
MOVF	f, d	f'i taşı	1	Z	1,2
MOVWF	f	W'yif'e taşı	1	Hiçbiri	
NOP	-	Hiç bir işlem yapma	1	Hiçbiri	
RLF	f, d	Elde ile sola döndür	1	C	1,2
RRF	f, d	Elde ile sağa döndür	1	C	1,2
SUBWF	f, d	f'ten W'yi çıkar	1	C,DC,Z	1,2
SWAPF	f, d	f'tekinibılların (4 bit) yerini değiştir	1	Hiçbiri	1,2
XORWF	f, d	W ile f'i mantıksal Özel-VEYA işlemine tabi tut	1	Z	1,2
BCF	f, b	f'in bitini temizle	1	Hiçbiri	1,2
BSF	f, b	f'in bitini set et 1 yap	1	Hiçbiri	1,2
BTFSC	f, b	f'in bitine bak, 0 ise zıpla, sonraki komutu geç	1(2)	Hiçbiri	3
BTFSS	f, b	f'in bitine bak, 1 ise zıpla, sonraki komutu geç	1(2)	Hiçbiri	3
ADDLW	k	W ve sabiti topla	1	C,DC,Z	
ANDLW	k	W ile sabiti mantıksal VE işlemine tabi tut	1	Z	
CALL	k	Alt program çağır	2	Hiçbiri	
CLRWDT	-	izleyen-köpek Saatini temizle	1	TO,PD	
GOTO	k	Adrese git	2	Hiçbiri	
IORLW	k	W ile sabiti mantıksal VEYA işlemine tabi tut	1	Z	
MOVLW	k	W ye sabit yükle	1	Hiçbiri	
RETFIE	-	Kesme programından dön	2	Hiçbiri	
RETLW	k	W de bir sabit ile dön	2	Hiçbiri	
RETURN	-	Alt programdan dön	2	Hiçbiri	
SLEEP	-	Uyku moduna gir	1	TO,PD	
SUBLW	k	Sabitten W'yi çıkar	1	C,DC,Z	
XORLW	k	W ile sabiti mantıksal Özel-VE işlemine tabi tut	1	Z	

NOTLAR:

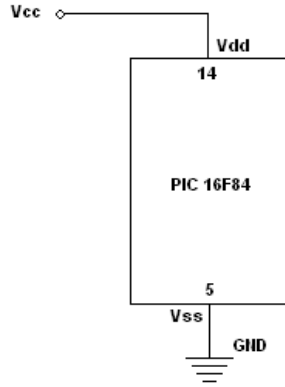
- (1) Bir G/Ç kütüğü kendisinin bir fonksiyonu olarak değiştiğinde (MOVF PORTB, 1), kullanılan değer, uçların kendisinde bulunan değer olacaktır. Örneğin, eğer giriş olarak tanımlanan bir uça tutulan değer '1' ise ve harici bir aygıttan mantık 0 olarak sürülüyor ise, veri '0' olarak geri yazılacaktır.
- (2) Eğer bu komut TMR0 kütüğünde kullanılırsa (ve, mümkünse, d = 1 ise), önbölücü TMR0'a adanmışsa temizlenecektir.
- (3) Eğer Program Sayacı (PC) değişmiş ise veya bir şarta bağlı test doğru ise, bu komut iki çevrimde işlenir. İkinci çevrim bir NOP komutu olarak icra edilir.

Tablo 1.1: PIC 16F84 komut seti

1.3. Mikrodenetleyici Yapısı

1.3.1. Besleme Gerilimi

PIC mikrodenetleyicisi besleme devresi için özel bir devreye gereksinim duymamaktadır. Besleme gerilimi 2 ila 6 V DC arasında deęişmekle birlikte dięer dijital devrelerle birlikte daha rahat kullanılabilmesi için 5V DC uygun bir deęerdir.



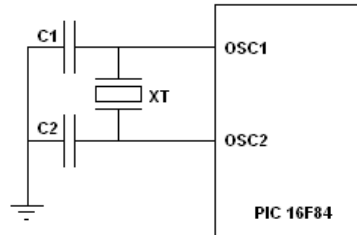
Şekil 1.8: PIC 16F84 Besleme gerilimi

1.3.2. Clock Düzeni ve Osilatör Çeşitleri

PIC16F84 Mikrodenetleyicisi farklı osilatör tipleri ile çalışabilmektedir. Bu osilatör tipleri şunlardır:

- LP LowPowerCrystal (Düşük Güçlü Kristal Osilatör)
- XT Crystal/Resonator (Kristal, Seramik Rezonatör)
- HS High SpeedCrystal/Resonator (Yüksek Hızlı Kristal, Rezonatör)
- RC Resistor/Capacitor (Direnç, Kondansatör)

LP / XT / HS Kristal ya da seramik rezonatör elemanları kullanıldığında en basit devre şekil 1.9'daki gibi gerçekleştirilebilir. Eğer seramik rezonatör kullanılıyor ise C1 ve C2 kondansatörlerine gerek yoktur.

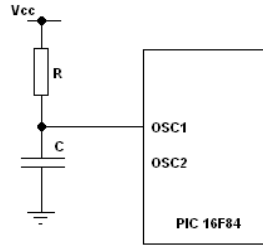


Şekil 1.9: PIC için LP/XT/HS osilatör bağlantısı

Mod	Frekans	OSC1/C1	OSC2/C2
LP	32 kHz	68 - 100 pF	68 - 100 pF
	200 kHz	15 - 33 pF	15 - 33 pF
XT	100 kHz	100 - 150 pF	100 - 150 pF
	2 MHz	15 - 33 pF	15 - 33 pF
	4 MHz	15 - 33 pF	15 - 33 pF
HS	4 MHz	15 - 33 pF	15 - 33 pF
	10 MHz	15 - 33 pF	15 - 33 pF

Tablo 1.2: PIC 16F84 için osilatör frekansları

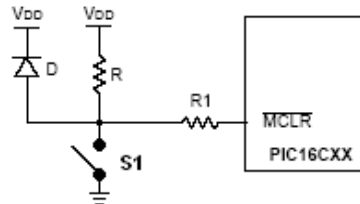
RC osilatör zamanlamanın hassas olmadığı durumlarda maliyeti düşürmek için tercih edilebilir. Ayrıca ortam ısısı nem vs. gibi çevresel etkilere duyarlıdır. Yüksek değerli direnç kullanılması önerilmez. Bu durumda dış etkilere duyarlılığı artacaktır. Direnç değerinin 3 KΩ ile 100 KΩ arasında bir değerde olması önerilir. Aynı şekilde kondansatörde 20 pF değerinin üzerinde olmalıdır. (Şekil 1.10)



Şekil 1.10: PIC 16F84 için RC osilatörü bağlantısı

1.3.3. Reset Uçları ve Reset Devresi

PIC mikrodenetleyicisi besleme gerilimi aldığı anda programın başlangıç adresinden itibaren çalışması için dâhili Power on Reset devresi bulunmaktadır. Ancak bazen kasıtlı olarak programın çalışmasını başlangıç durumuna almak gerekebilir. Bu durumda Reset uçları kullanılmaktadır.

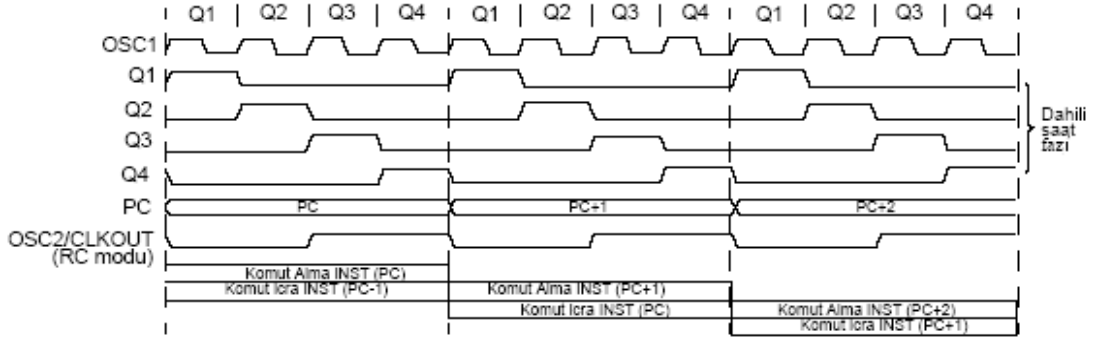


Şekil 1.11: PIC 16F84 için reset devresi

1.3.4. Komut Akışı

Bir komut çevrimi, dört Q çevrimini (Q1, Q2, Q3 ve Q4) içerir. Komut alınması bir komut çevrimi ve komut kodu çözümü ve komutun icra edilmesi de diğer bir komut çevrimi olacak şekilde, komut alınması ve icra edilmesi pipeline (boruhattına) yerleştirilir. Boruhattı sayesinde, her komut tek bir çevrimde etkili bir şekilde icra edilmektedir. Bir komut, program sayıcının(PC) değişmesine sebep olursa(GOTO), bu komut için iki çevrim gerekecektir. Komut alınma çevrimi, program sayıcının (PC) Q1'de artırılması ile başlar.

Komut icra çevriminde, alınan komut, komut kütüğünde Q1 esnasında tutulur. Q2, Q3 ve Q4 fazları sürecinde ise, bu komutun kodu çözülür ve icra edilir. Veri hafızası Q2 fazında okunur ve Q4 fazında yazılır.(Şekil 1.12)



Şekil 1.12: PIC 16F84 komut akış diyagramı (Saat çevrimi)

1.3.5. Yazmaçlar

PIC 16F84 mikrodenetleyicisinde yazmaçlar RAM bellek içerisinde yer almaktadır. Şekilde hafızanın yerleşim düzeni verilmiştir. Burada 0x0C adresinden itibaren 0x4F adresine kadar olan kısım genel amaçlı hafızadır ve kullanıcıya ayrılmıştır. Aynı şekilde 0x8C adresinden 0xCF adresine kadar olan kısımda kullanıcı içindir. Ancak bu iki hafıza bölümü farklı banklarda yer almasına rağmen aslında birbirinin aynı verileri taşımaktadır. Bu yazmaç haricinde PIC 16F84 veri hafızası içinde yer almayan W yazmacı ve PC yazmaçları bulunmaktadır. W yazmacı akümülatör görevi üstlenir.

PIC mikrodenetleyicisinde özel amaçlar için kullanılan bazı yazmaçlar vardır. Bu yazmaçlar şunlardır:

➤ Durum Yazmaçları:

IRP	RP1	RP0	TO	PD	Z	DC	C
-----	-----	-----	----	----	---	----	---

- **bit 0 – C (CarryFlag):** Toplama çıkarma ve kaydırma işlemleri esnasında bir taşma meydana gelirse bu bit etkilenir. Taşma gerçekleştiği durumda Lojik 1 olur. Taşma yoksa Lojik 0 olur.
- **bit 1 – DC (DigitCarryFlag):** Toplama çıkarma ve kaydırma işlemleri esnasında üçüncü bitten dördüncü bite bir taşma meydana gelirse bu bit etkilenir. Taşma gerçekleştiği durumda Lojik 1 olur. Taşma yoksa Lojik 0 olur.
- **bit 2 – Z (Zero Flag):** Aritmetik veya lojik bir operasyonun sonucu 0 oluyorsa bu bit etkilenir ve Lojik 1 olur.
- **bit 3 – PD (PowerDownFlag):** Enerji kesinti bitidir. Bu bit mikrodenetleyici ilk çalıştığında ve CLRWDT komutu ile Lojik 1 olur. Sleep komutu çalışınca Lojik 0 olur.

- **bit 4 – TO (Time OutFlag):** Watchdog taşma bitidir. PIC'e enerji verilince ve CLRWDT, SLEEP komutları ile Lojik 1 olur. WatchdogTimer saymayı tamamlayınca Lojik 0 olur.
- **bit 6:5 – RP1:RP0 (Register Bank Select Bits):**Yazmaç Bank Seçim Biti: Bu bitler değiştirilerek hafıza alanındaki banklara erişim mümkün olmaktadır. RP1 PIC 16F84'te kullanılmamaktadır. Daha yüksek hafızaya sahip üst modellerde kullanılabilir. Bank seçim için aşağıdaki gibi ayarlanır.

0:0 Bank 0 (00H – FFH)
0:1 Bank 1 (80H – FFH)

- **bit 7 – IRP (Register Bank Select Bits):**Dahili hafızada dolaylı adresleme için kullanılmaktadır. Lojik 0 olursa bank 0 ve 1 kullanılır. Lojik 1 olursa bank 2 ve 3 kullanılır. PIC16F84 denetleyicisinde 0 olmalıdır.

➤ **Seçenek Yazmaçları:**

RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
------	--------	------	------	-----	-----	-----	-----

- **bit 2:0 – PS2,PS1,PS0 (Prescalar Rate Select Bit):**Zamanlayıcı için kullanılan sinyal kaynağının frekansını bölmek içindir. Frekansın kaç bölüneceği bu bitler ile ayarlanır. Frekans bölme değerleri aşağıda verilmiştir.

Frekans Bölme Sayısı	TMR0 Oranı	WDT Oranı
000	1/2	1/1
001	1/4	1/2
010	1/8	1/4
011	1/16	1/8
100	1/32	1/16
101	1/64	1/32
110	1/128	1/64
111	1/256	1/128

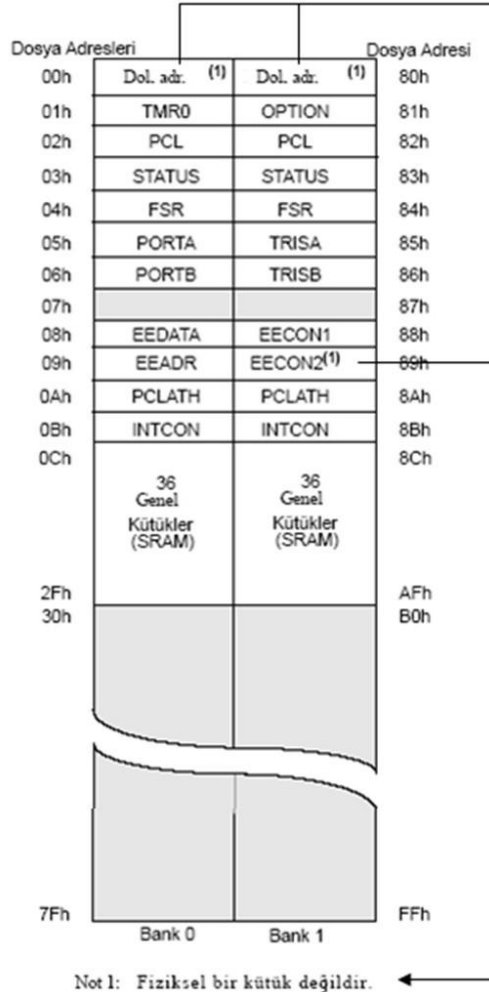
Tablo 1.3: Frekans bölme değerleri

- **bit 3 – PSA (PrescalarAssignment Bit):**Frekans bölücü ayırma bitidir. Bu bit Lojik 0 ise frekans bölme işlemi TMR0 için geçerlidir Lojik 1 ise WDT için geçerlidir.
- **bit 4 – TOSE (TMR0 Source Edge Select Bit):**TMR0 için sinyal kaynağı kenar seçme bitidir. Eğer Lojik 0 yapılırsa RA4/TOCKI ucundan düşen kenar tetiklemeli olur. Lojik 1 ise yükselen kenar tetiklemeli olur.

- **bit 5 – TOCS (TMR0 Clock Source Select Bit):**TMR0 için sinyal kaynağı seçme bitidir. Lojik 0 ise ¼ osilatör frekansı seçilir. Lojik 1 ise harici dijital sinyal (RA4/TOCKI) seçilir.
- **bit 6 – INTEDG (InterruptEdge Select Bit):**Harici kesme aktif ise (RB0/INT) sinyalin hangi kenarında tetikleneceği belirlenir. Lojik 0 düşen kenarda Lojik 1 yükselen kenarda tetiklemek içindir.
- **bit 7 – RBPU (PORTB Pull-UpEnable Bit):**PORTB dahili Pull-Up dirençlerini kontrol eder. Lojik 0 ise devre dışıdır. Lojik 1 ise devrededir.

1.3.6. Program Belleği

Program belleği EEPROM tabanlı ve 1K x 14 bir hafıza organizasyonuna sahiptir. Yani 14 bit uzunluğunda 1024 adet hücre içerir. Programın çalışması esnasında bu bölümdeki verilere müdahale edilemez.



Şekil 1.13: PIC 16F84 Mikrodenetleyicisinde yer alan genel ve özel amaçlı yazmaçların yer aldığı veri hafızası.

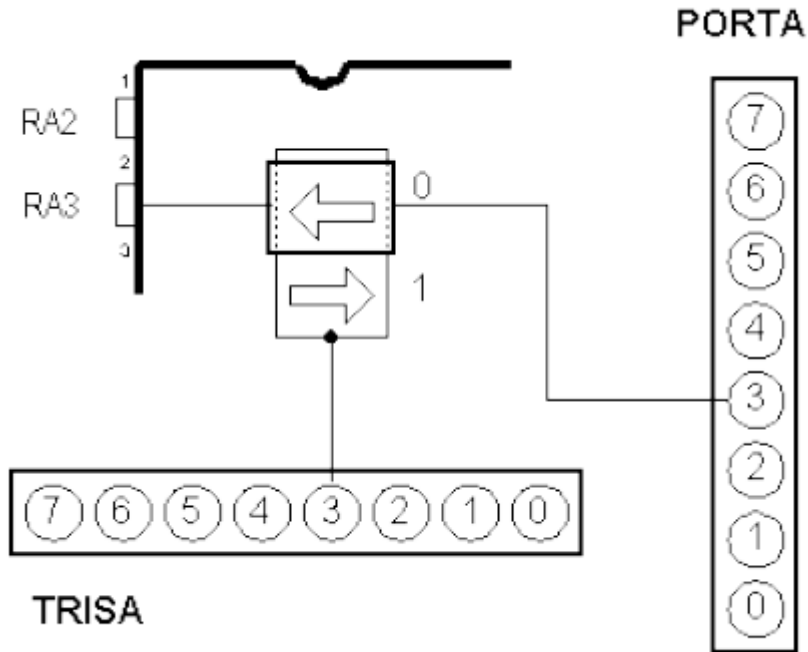
1.3.7. Veri Belleği

Veri belleği 80 adet bank 1’de ve 80 adet bank 2’de olmak üzere 160 adet gibi görünse de aslında bazı özel amaçlı yazmaçlar dışında bank 1 ve bank 2’deki bilgiler birbirinin kopyasıdır. Genellikle 0x0C ile 0x04 adresleri arasındaki bölüm programlar tarafından genel amaçlı veri hafızası olarak kullanılır.

1.3.8. I/O Portları

PIC 16F84 denetleyicisi PORTA ve PORTB olmak üzere iki porta sahiptir. Ancak bu portların bazı uçları denetleyicinin diğer özellikleri için kullanılabilir.

- **PORTA ve TRISA:** PORTA 5 bit uzunluğundadır. TRISA yazmacının herhangi bir biti 1 yapılırsa PORTA’nın aynı değerli ucu giriş ucu olarak kullanılır. Aynı şekilde TRISA yazmacının herhangi bir biti 0 yapılırsa bu sefer PORTA’nın aynı değerli ucu çıkış olacaktır. RA4 portu aynı zamanda TIMER(Zamanlayıcı) için kullanılmaktadır.
- **PORTB ve TRISB:** PORTB 8 bit uzunluğunda bir porttur. Giriş ya da çıkış olarak ayarlanması TRISB ile yapılmaktadır. PORTB’nin 4 hattında (RB4:RB7 arası) değişim olduğunda interrupt oluşturma yeteneği vardır. Aynı zamanda RB0/INT ucu externalinterrupt için kullanılabilir.



Şekil 1.14: PIC 16F84 Mikrodenetleyicisinde PORTA ve TRISA arasındaki ilişki.

UYGULAMA FAALİYETİ

Aşağıda verilen işlem basamaklarını takip ederek mikrodenetleyici katalog bilgilerinden ve internet sayfalarından araştırma yapınız ve hangi mikrodenetleyiciyi ve hangi donanımları kullanacağınızı seçiniz.

İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Çeşitli mikrodenetleyicilere ait veri kitapçıklarını (Data Sheet) internetten temin ederek denetleyici karakteristiklerini besleme gerilimini ve osilatör devrelerini defterinize not alınız.	<ul style="list-style-type: none">➤ Mikrodenetleyici üretici firmaların internet adreslerini ziyaret ediniz.
<ul style="list-style-type: none">➤ Veri kitapçıklarından mikrodenetleyicilerin birbirine göre avantaj ve dezavantajlarını defterinize not ediniz.	<ul style="list-style-type: none">➤ Mikrodenetleyici üretici firmaların internet adreslerini ziyaret ediniz.
<ul style="list-style-type: none">➤ Farklı mikrodenetleyicileri maliyet açısından kıyaslayınız.	<ul style="list-style-type: none">➤ Mikrodenetleyici üretici firmaların internet adreslerini ziyaret ediniz.
<ul style="list-style-type: none">➤ Çeşitli mikrodenetleyicileri elinize alarak boyutlarını tespit ediniz.	<ul style="list-style-type: none">➤ Mikrodenetleyici kontrollü devrelerin boyutlarını göz önüne alarak devrelerin üretim safhalarını ve süresini sınıfta arkadaşlarınızla birlikte tartışınız.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () Mikrodenetleyici tek bir yonga üzerinde bir mikroişlemci, hafıza, giriş/çıkış portları vb. yardımcı devreler içeren devre elemanıdır.
2. () RISC işlemcilerinde, CISC işlemcilere göre komut sayısı daha azdır.
3. () PIC16F84 mikrodenetleyicisi çalışmak için harici bir RAM belleğe ve program yüklemek için bir EEPROM'a ihtiyaç duymaktadır.
4. () EEPROM bellekler içerisindeki bilgiler kalıcıdır ve içerisindeki bilgileri silmek yada değiştirmek mümkün değildir.
5. () RAM bellekler hızlıdır fakat devrenin enerjisi kesildiğinde içindeki veriler silinir. Bu nedenle geçici bellek olarak kullanılır.
6. () İkili sayı sisteminde ifade edilen sayının her bir basamağı bir Binarydigit olarak değerlendirilir.

Aşağıdaki cümlelerde boş bırakılan yerlere doğru sözcükleri yazınız.

7. $(1100101)_2$ sayısını Decimal ve Hexadecimal sistemde gösteriniz.
.....,
8. $(3F8)_{16}$ sayısını Binary ve Decimal sistemde gösteriniz.....,
9. MPASM derleyicisinde satır başında ; (noktalı virgül)kullanıldığında o satırolarak değerlendirilir.
10. Bazı komutlarda doğrudan rakamlar kullanılabilir. Bunlaradenilir.
11. deyimi iki amaç için kullanılabilir: Birincisi programın başlangıç adresini belirlemek için, ikincisi isealt programı başlangıç adresini belirlemektir.
12.yönlendirmeli komular d ile belirtilen hedefe yapılan işlem sonucunu yazarlar. d 1 ise kütük yazmacına, 0 ise W yazmacına yazılır.

13.yönlendirmeli komutlar belirtilen yazmacın yine komutta belirtilen biti ile işlem yapmaktadırlar.
14. EEPROM tabanlı ve 1K x 14 bir hafıza organizasyonuna sahiptir.
15.zamanlamanın hassas olmadığı durumlarda maliyeti düşürmek için tercih edilebilir.
16.sayesinde, her komut tek bir çevrimde etkili bir şekilde icra edilmektedir.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

Mikrodenetleyici programlama kartını hazırlayabileceksiniz.

ARAŞTIRMA

- Çeşitli mikrodenetleyiciler için programlama kartları ve yazılımları hakkında araştırma yapınız.
- Birden fazla mikrodenetleyiciye uygun programlama kartlarını araştırınız ve bu kartların avantaj ve dezavantajlarını belirleyiniz.

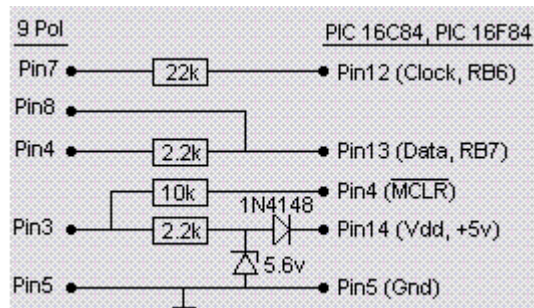
2. MİKRODENETLEYİCİ PROGRAMLAMA KARTI YAPIMI

2.1. Programlama Kartı

2.1.1. Programlama Devresinin Yapımı

Mikrodenetleyicilerle yapılan devrelerin dizaynı yapıp uygulama aşamasına gelindiğinde denetleyici içerisine yazılım yüklemek için çeşitli programlama cihazlarına gereksinim vardır. Ancak PIC mikrodenetleyicileri için bu cihazlar uygulaması basit ve etkili devrelerdir.

Eğitim amaçlı ve deneysel çalışmalarda PIC16F84 için yapılmış çok basit devreler kullanılabilir. Şekil 2.1’ de örnek olarak kullanılacak bir seri programlama devresi verilmiştir. Devre, güç kaynağı gerektirmeden seri port aracılığı ile haberleşen deney amaçlı çalışmalar için çok basit ve yeterli bir devredir.



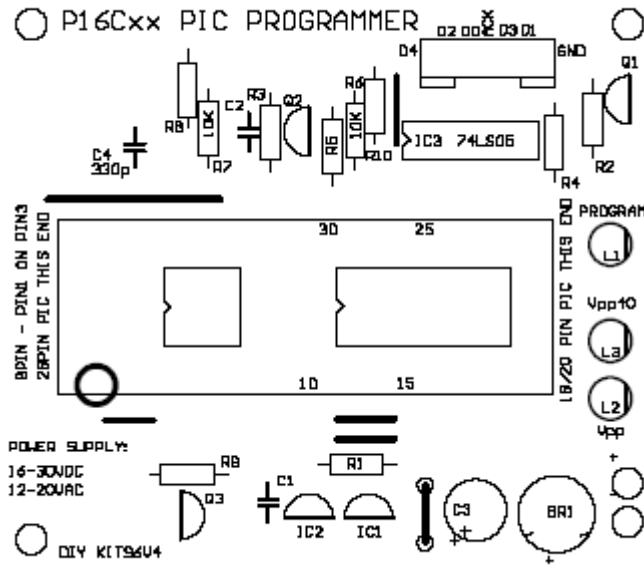
Şekil 2.1: PIC16F84 için deneysel çalışmalarda kullanılacak basit bir devre şeması

Tablo 2.1’ de P16PRO devresi malzeme listesi verilmiştir. Şekil2.3’ te P16PRO devresinin malzeme yerleşim planı, Şekil2.4’ teP16PRO devresinin baskılı devre üst yüzeyi, Şekil 2.5’ te P16PRO devresinin baskılı devre alt yüzeyi gösterilmiştir.

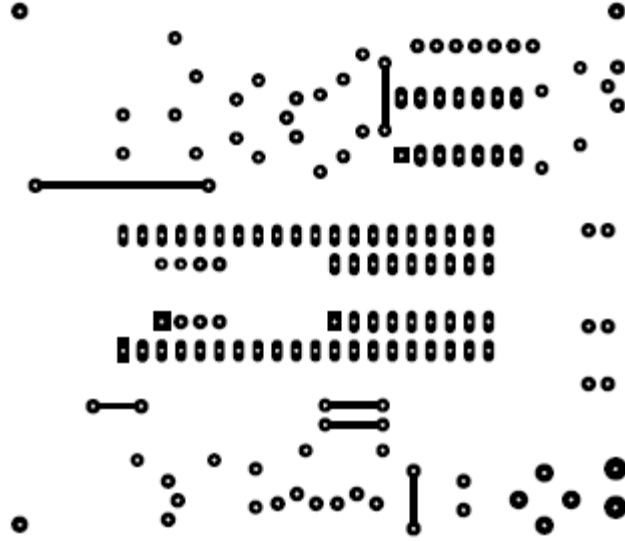
Malzeme Listesi:

No	Adet	Malzeme Adı	Tip
1	1	TEXT TOOL 40 ZIF	SOCKET
2	1	1k	R1
3	6	4k7	R2 R3 R4 R5 R9 R10
4	3	10k	R6 R7 R8 *
5	1	74LS05	IC3
6	1	78L05	IC1
7	1	78L08	IC2
8	2	100nF	C1 C2
9	1	220uF/40V	C3
10	1	330pF	C4 *
11	1	680R	R11
12	1	B80C800	G1
13	3	BC557	T1 T2 T3
14	1	Green LED	D2
15	2	Red LED	D1 D3

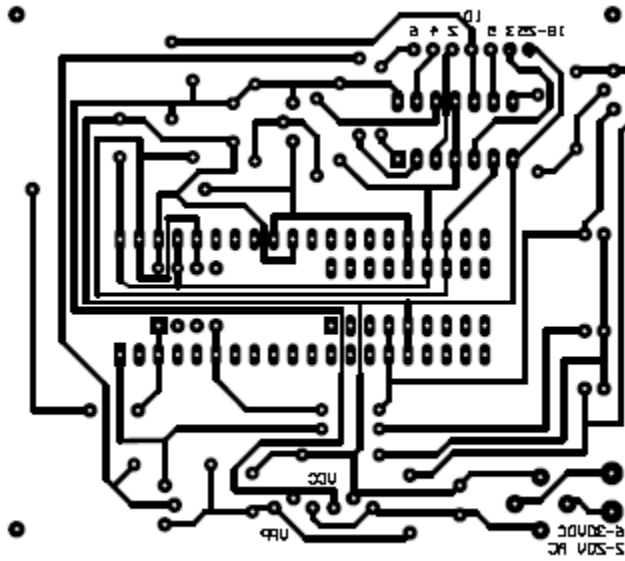
Tablo 2.1:Malzeme listesi



Şekil 2.3: P16PRO malzeme yerleşim planı



Şekil 2.4: P16PRO baskılı devre üst yüzeyi







Şekil 2.5: P16PRO baskılı devre alt yüzeyi

2.1.2. Bağlantı Kablosunun Yapımı

Standart paralel yazıcı bağlantı kablosunun ilgili pinlerini baskılı devre üzerinde belirtilen bağlantı noktalarına lehimleyerek devreyi gerçekleştirebilirsiniz. Ayrıca bir kabloya gereksinim duyulmamaktadır.

UYGULAMA FAALİYETİ

Aşağıda verilen işlem basamaklarını takip ederek mikrodenetleyici programlama kartı yapınız.

İşlem Basamakları	Öneriler
<p>➤ P16Pro programlayıcı devrenin baskılı devre tasarımını baskı devre transfer kağıdı üzerine lazer yazıcı ile çıkarınız.</p> 	<p>➤ Baskı devreyi çıkarırken kâğıda ters görüntü çıkarmayı ve maksimum kalitede baskı yapmayı ihmal etmeyiniz. Böylece toner miktarı fazla olacaktır ve hata oranı az olur.</p>
<p>➤ Baskı yapılan transfer kâğıdını bakırlı plaketin bakır yüzeyine yerleştirerek yüksek ısıya ayarlı ütü ile ütöleyiniz. Böylece toner bakıra yapışacaktır.</p> 	<p>➤ Kâğıdın ve plaketin yeterli miktarda ısınıp tonerin bakır üzerine yapıştığından emin olunuz.</p>
<p>➤ Tonerin plakete yapıştığından emin olduktan sonra transfer kağıdını dikkatlice ayırınız.</p> 	<p>➤ Çizimin eksik olan kısımlarını asetat kalem ile düzeltiniz.</p>
<p>➤ Hazırladığınız plaketi asit içerisine atarak (3:1 oranında Tuz Ruhü ve Perhidrol karışımı) boyanmamış bölümlerin çözülerek kaybolmasını bekleyiniz.</p> 	<p>➤ Asitin çözülmesi esnasında açığa çıkan gazı solumayınız ve açık havada işlemi gerçekleştirin. Aksi takdirde sağlık sorunlarına yol açabilir.</p>

<p>➤ Hazırlanan plaketi ince zımpara ile zımparalayarak plaketin temizlenmesini sağlayınız.</p> 	<p>➤ Zımparanın mümkün olduğunca ince seçilmesine dikkat ediniz. Aksi takdirde ince çizilen yollar zarar görebilir.</p>
<p>➤ Devrenin gerekli deliklerini bir matkap yardımı ile deliniz.</p> 	<p>➤ Delik için 1mm matkap ucu ve basit bir el matkabı kullanınız.</p>
<p>➤ Lehimleme işlemini yaparak devreyi hazırlayınız.</p> 	<p>➤ Lehimleme yaptıktan sonra kalan lehim atıklarını temizleyiniz.</p>

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise **D**, yanlış ise **Y** yazınız.

1. () Mikrodenetleyiciler için geliştirilen yazılımları denetleyicilere aktarmak için basit bir programlama yazılımı yeterlidir.
2. () Programlayıcı devreler bilgisayara seri, paralel veya usb bağlantı noktası ile bağlanarak iletişim kurarlar.
3. () Her programlayıcı cihaz için cihaza özel yazılım gerekmez. Derleyici bunların hepsini programlayabilecek yapıdadır.
4. () Baskılı devre üretimi esnasında asit içinde çözme işlemi kapalı bir ortamda gerçekleştirilmelidir.
5. () Baskı devre transfer kağıdına en yüksek kalite ile baskı almak gerekir. Böylece toner miktarı fazla olur ve bakır yüzeye aktarım daha sağlıklı olur.
6. () P16PRO programlayıcı devre seri port ile bağlanan bir JDM programlayıcı devresidir.
7. () P16PRO programlayıcısı PICALLW yazılımı ile rahatlıkla kullanılabilir.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarlarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-3

AMAÇ

Mikrodenetleyici programlama editörünü kullanabileceksiniz.

ARAŞTIRMA

- Mikrodenetleyici için yazılan programları derleyen ve mikrodenetleyici aktaran programları araştırınız.
- Araştırma sonucunda belirlediğiniz bazı programların birbirlerine göre avantaj ve dezavantajlarını tartışınız.
- MPASM programının yapısı ve çalışması hakkında bir ön araştırma yapınız.

3. MİKRODENETLEYİCİ PROGRAMLAMA EDİTÖRÜNÜN KULLANIMI

3.1. Mikrodenetleyici Program Editörünün Kurulumu

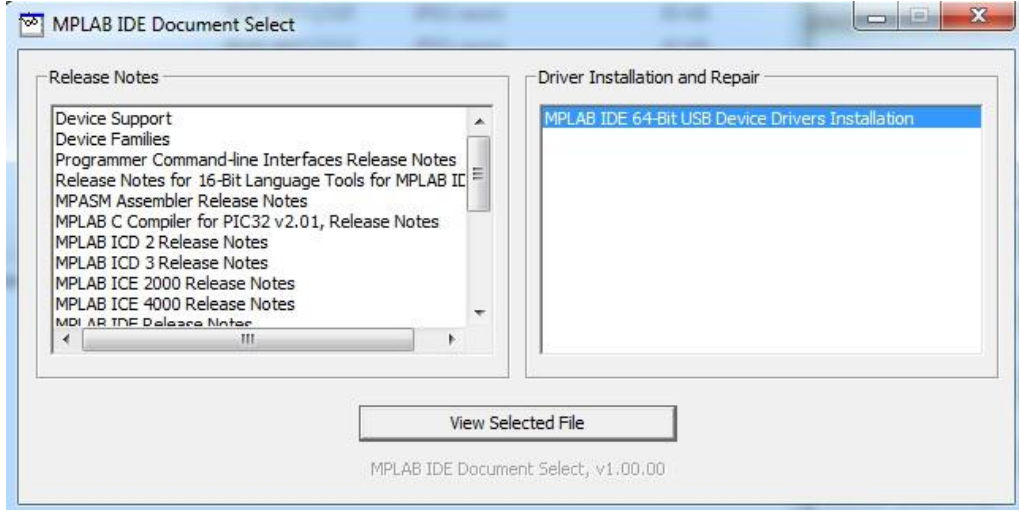
Microchip PIC mikrodenetleyicisi için en yaygın kullanılan derleyici programı, kendisine ait olan MPASM derleyicisidir. MPASM derleyicisi MPLAB paket programının bir parçasıdır. MPLAB paket programı kapsamlı bir uygulamadır. Editör, Derleyici, Simulatörün bir arada olduğu bir entegre geliştirme ortamıdır (IDE: Integrated Development Environment).

3.1.1. Derleyici Editörün Yüklenmesi

MPLAB yazılımını kurmak için üreticinin kendi web sitesinden ücretsiz kurulum paketini kişisel bilgisayarına indirmek mümkündür.

Firmanın internet sitesinden mplab paket yazılımının son sürümünü kurmak için indirilen sıkıştırılmış dosyayı bilgisayarınızda bir klasöre açarak setup uygulama dosyasını çalıştırın. Kurulum programı hazırlıklarını yapacaktır. Adımları takip ederek yazılımı kurabilirsiniz.

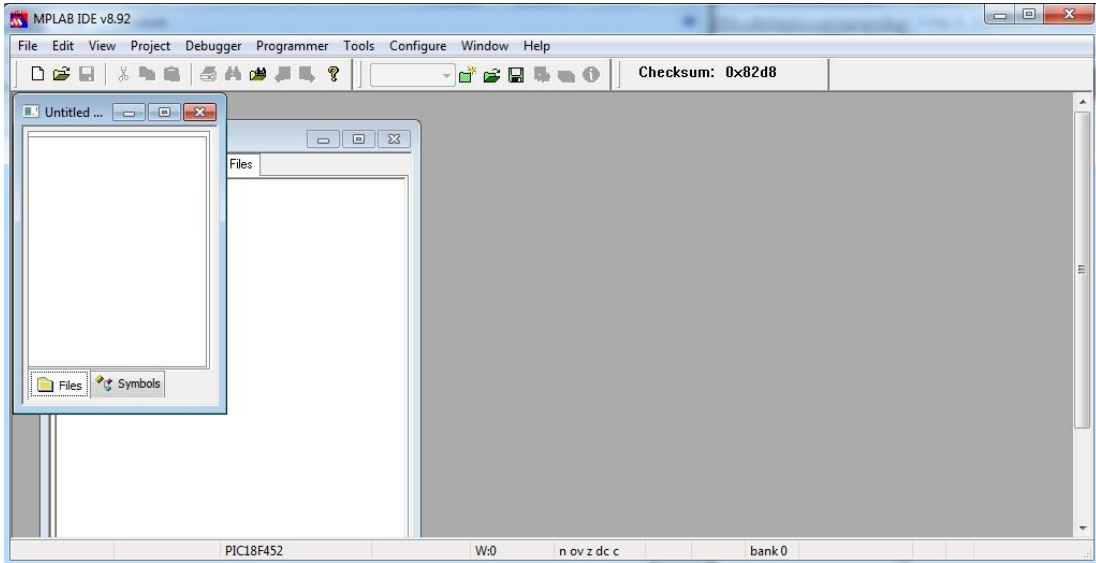
Eğer USB arabirim ile haberleşen bir programlama kartı, debug kartı veya deney kartı gibi bir cihazda kullanılacaksa, (Firmanın kendi sitesinde bu setler tanıtılmaktadır ve ücrete tabidir.) USB sürücüler yüklenmelidir, aksi takdirde yüklenmesine gerek yoktur. Yükleme için ilk açılışta ekrana gelen ve şekil 3.1’ de gösterilen ekrandaki seçim yapılır. Program firmanın internet sayfasına yönelir ve ilgili sürücü dosyaları siteden yüklenir.



Şekil 3.1: MPLAB USB arayüz sürücü kurulumu. (Adım 9)

3.1.2. Editörün Ayarlarının Yapılması

MPLAB programı başlatıldığında şekildeki gibi bir ekranla karşılaşırız. Yazılımın genel görünümü bu şekildedir.

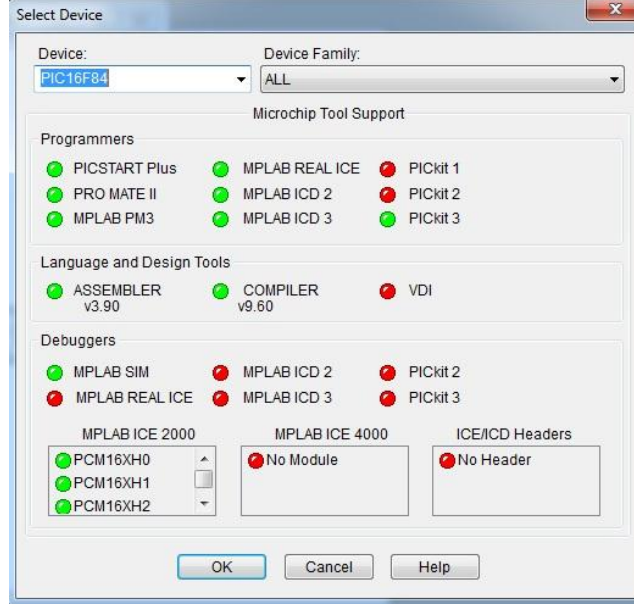


Şekil 3.2: MPLAB ana ekranı

MPLAB programını kullanmadan önce yazılım geliştirilecek denetleyiciye göre programda bazı ayarlamalar yapılması gerekir. Bunlar:

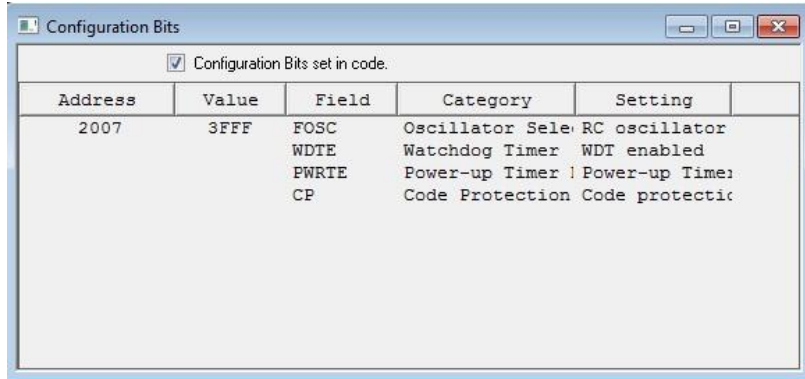
- *Debugger* menüsünden *Select Tool* komutu ile *MPLAB SIM* seçilir. Bu sayede simülasyon kullanılabilir.

- Daha sonra *Configure* menüsünden *Select Device* komutu ile kullanmak istediğimiz denetleyici seçilmelidir.



Şekil 3.3: MPLAB denetleyici ve simülâtör seçimi

- Bu aşamadan sonra denetleyicinin yapılandırma bitlerini ayarlamak için *Configure* menüsünden *ConfigurationBits* komutu verilerek denetleyicinin yapılandırma bitleri ayarlanır.

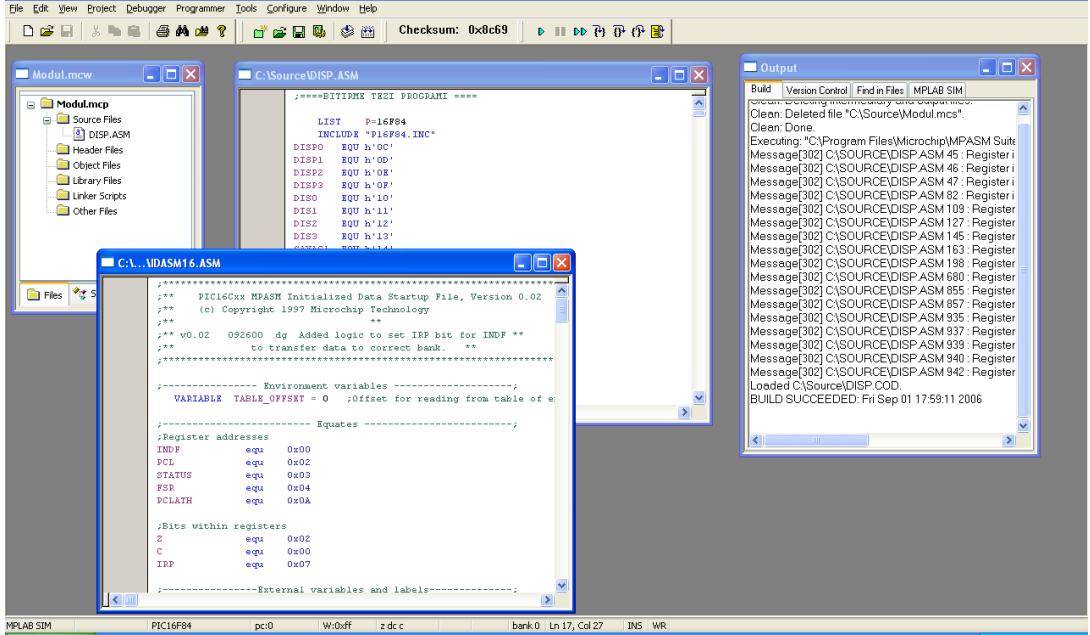


Şekil 3.4: MPLAB denetleyicinin yapılandırılan bitleri

3.1.3. Editörün Özellikleri

MPLAB başlatıldığında doğrudan iki adet pencere açılmaktadır. Workspace ve Output pencereleri. Workspace penceresinde hazırlanan projenin bileşenleri listelenecektir. Bir çeşit görev bölmesi gibidir. Output penceresi programın verdiği mesajları kullanıcıya iletmekle görevlidir. Çeşitli bileşenleri izleyerek raporlar verir.

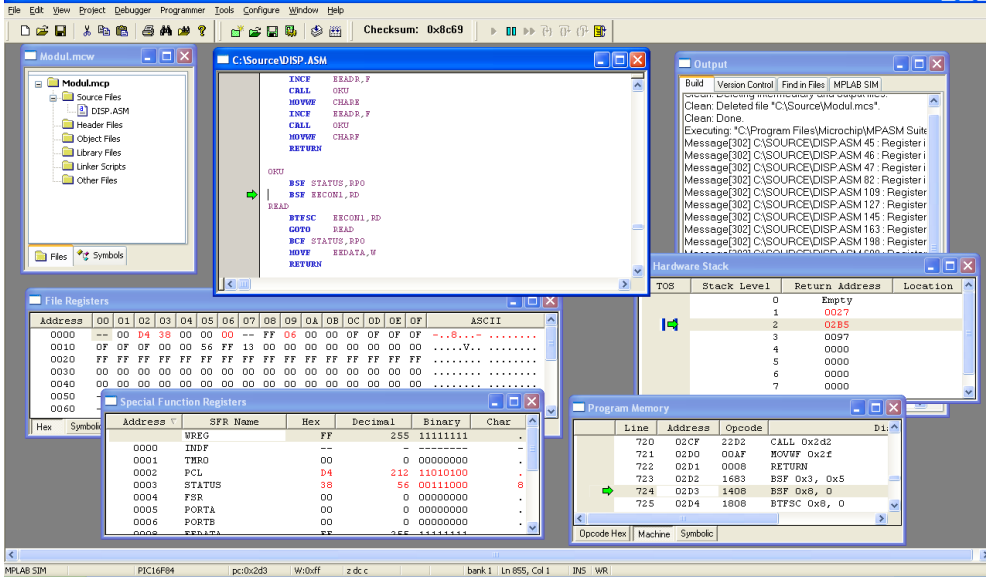
Editing yani düzenleme ve program kodlarını yazma safhasında olan bir projenin görünümü şekil3.5.'deki gibidir.



Şekil 3.5: MPLAB editör özellikleri

Kodlar renklendirilerek hata oranı azaltılmıştır.

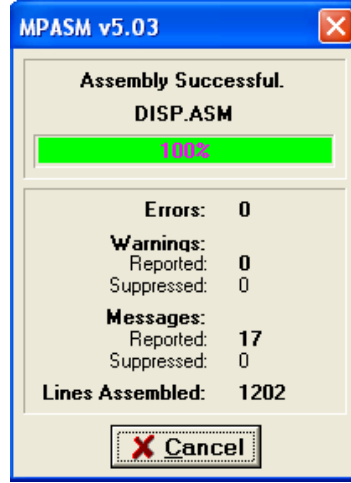
Kodla işlemi bittikten sonra derlenmiş bir projenin simülasyonu sırasında MPLAB'ın görünümü şekil3.6.'dakine benzer biçimdedir.



Şekil 3.6: MPLAB Simülasyon durumundaki görünüm

3.1.4. Denetleyiciye Yüklenecek Programın Derlenmesi ve Denenmesi

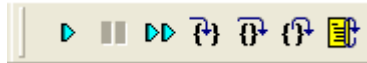
MPLAB yazılımında hazırlanmış bir projeyi denetleyiciye yükleyebilmek için derlenmiş Binary dosyalara ihtiyaç vardır. Bu MPASM aracılığı ile dışarıda haricen yapabileceği gibi, aynı zamanda MPLAB içerisinde *Project* menüsünden *Quickbuild*, *Make*, *BuildAll* gibi komutlarla derleme işlemi gerçekleştirilebilir. Derleme işlemi tamamlanınca şekildeki gibi bir mesajla karşılaşılır.



Şekil 3.7: MPLAB derleyici sonuç ekranı

Eğer yüzde çubuğu yeşil renkte ise derleme işleminin hatasız bir şekilde gerçekleştiği anlaşılır. Derleme işlemi bittikten sonra simülatör başlatılarak program analiz edilebilir.

Şekil 3.8’de *Run*, *Pause*, *Animate*, *Step Into*, *Step Over*, *Step Out*, *Reset* komutları ile programın simülasyonuna başlanabilir. Programın akışı ağır çekimde izlenir gibi incelenecek ise *Animate* komutu verilmelidir. Doğrudan sonuca bakılacak ise *Run* komutu seçilir.



Şekil 3.8: MPLAB simülasyon komutları

Simülasyon esnasında programı daha iyi analiz etmek için kullanılan pencereler aşağıdaki gibi sıralanabilir:

- **Program kodları:** Burada kullanıcının yazdığı kodların çalışma zamanındaki akışı izlenebilir. Yeşil işaretin olduğu satır icra edilmektedir.

```

C:\Source\DISP.ASM
CRLL    WAIT
GOTO   DONGUO
IPTAL
MOVLW  h'FF'
MOVWF  SAYAC3
MOVLW  h'13'
MOVWF  SAYAC4
BCF    BAYRAK,0
BCF    BAYRAK,1
BCF    BAYRAK,2
BCF    BAYRAK,3
BCF    BAYRAK,4
MOVLW  h'0F'
MOVWF  DISP0
MOVWF  DISP1
MOVWF  DISP2
MOVWF  DISP3
MOVWF  DISP4

```

Şekil 3.9: MPLAB program kodları

- **File registers:** Bununla veri hafızasının durumu çalışma zamanında izlenebilir. Kırmızı renkli hücreler en son işlem gören hücrelerdir.

Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	ASCII
0000	--	00	D4	38	00	00	00	--	FF	06	00	00	0F	0F	0F	0F	-.8...-
0010	0F	0F	0F	00	00	6B	FF	13	00	00	00	00	00	00	00	00k.....
0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0050	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
0060	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Şekil 3.10: MPLAB file registers

- **Program memory:** Bununla da program hafızasının durumu çalışma zamanında izlenmektedir. Yeşil işaret hangi komutun icra edildiğini göstermektedir.

Line	Address	Opcode	Di:
158	009D	30FF	MOVLW Oxff
159	009E	0095	MOVWF OX15
160	009F	0B95	DECFSZ OX15, F
161	00A0	289F	GOTO OX9f
162	00A1	0008	RETURN
163	00A2	0098	MOVWF OX18
164	00A3	3A0A	XORLW Oxa
165	00A4	1903	BTFSZ OX3, OX2
166	00A5	290D	GOTO OX10d
167	00A6	0818	MOVF OX18, W
168	00A7	3A0B	XORLW Oxb
169	00A8	1903	BTFSZ OX3, OX2
170	00A9	289F	GOTO OX9f

Şekil 3.11: MPLAB program hafızası

- **Special function registers:** Özel amaçlı kaydedicileri izlemek içindir. Kırmızı renkte olanlar son işlem gören kaydedicilerdir.

Address	SFR Name	Hex	Decimal	Binary	Char
	WREG	FF	255	11111111	.
0000	INDF	--	--	-----	-
0001	TMRO	00	0	00000000	.
0002	PCL	9F	159	10011111	.
0003	STATUS	18	24	00011000	.
0004	FSR	00	0	00000000	.
0005	PORTA	00	0	00000000	.
0006	PORTB	30	48	00110000	0
0008	FEDATA	FF	255	11111111	.

Şekil 3.12: MPLAB özel fonksiyon registerleri

- **Hardware stack:** Stack hafızasını gösterir.

TOS	Stack Level	Return Address	Location
	0	Empty	
	1	0028	
	2	0038	
→	3	0089	
	4	0000	
	5	0000	
	6	0000	
	7	0000	
	8	0000	

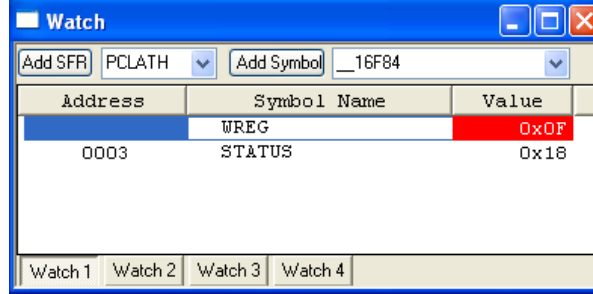
Şekil 3.13: MPLAB yığın yazmaçları

- **Disassembly listing:** Derlenmiş dosyadan tekrar elde edilen komut dosyasını gösterir. Aktif komut yeşil gösterge ile işaretlenmiştir.

Address	Mnemonic	Comment
08F	CALL	0x9d
090	MOVLW	0x10
091	BTFS	0x19, 0x7
092	IORWF	0x11, W
093	BTFS	0x19, 0x7
094	IORWF	0xd, W
095	MOVWF	0x6
096	CALL	0x9d
097	MOVLW	0
098	BTFS	0x19, 0x7
099	IORWF	0x10, W
09A	BTFS	0x19, 0x7
09B	IORWF	0xc, W
09C	MOVWF	0x6
09D	MOVLW	0xff
09E	MOVWF	0x15
09F	DECFSZ	0x15, F
0A0	GOTO	0x9f
0A1	RETURN	

Şekil 3.14: MPLAB disassembly listeleme

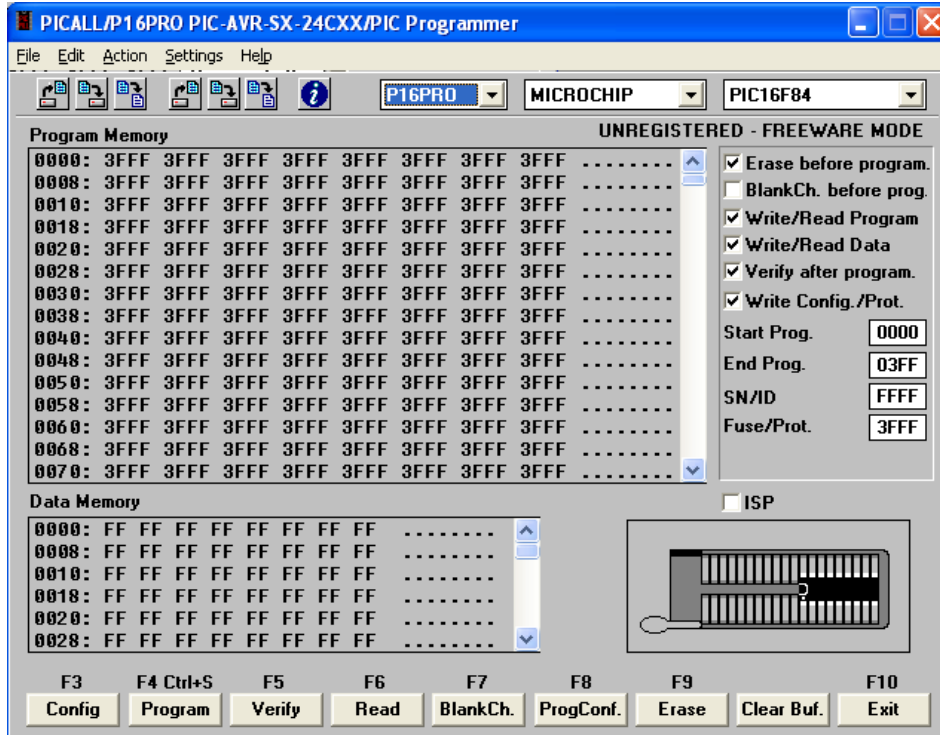
- **Watch:** Bu pencereden istediğimiz herhangi bir hafıza adresi, kaydedici ya da portu izleyebiliriz. İzlenmek istenen hafıza vs. adresi elle yazılmalıdır.



Şekil 3.15: MPLAB isteğe bağlı register izleme

3.1.5. Denetleyiciye Programın Yazdırılması

Hazırlanan kodların derleme işi bittikten sonra bunları PIC mikrodenetleyicisinin program hafızasına aktarmak gerekmektedir. Bunun için programlayıcı devrelere ihtiyaç vardır. Her ne kadar MPLAB içerisinde bazı programlayıcı devrelere destek olsa dahi bunların çoğu ücretli ve temini zor olduğu için kullanıcının kullanmak istediği programlayıcıya ait özgün programları kullanması daha akıllıca olacaktır. Önceki bölümlerde anlatılan P16PRO programlayıcısı için hazırlanmış PICALL yazılımının program görünümü Şekil 3.16’da gösterilmiştir.



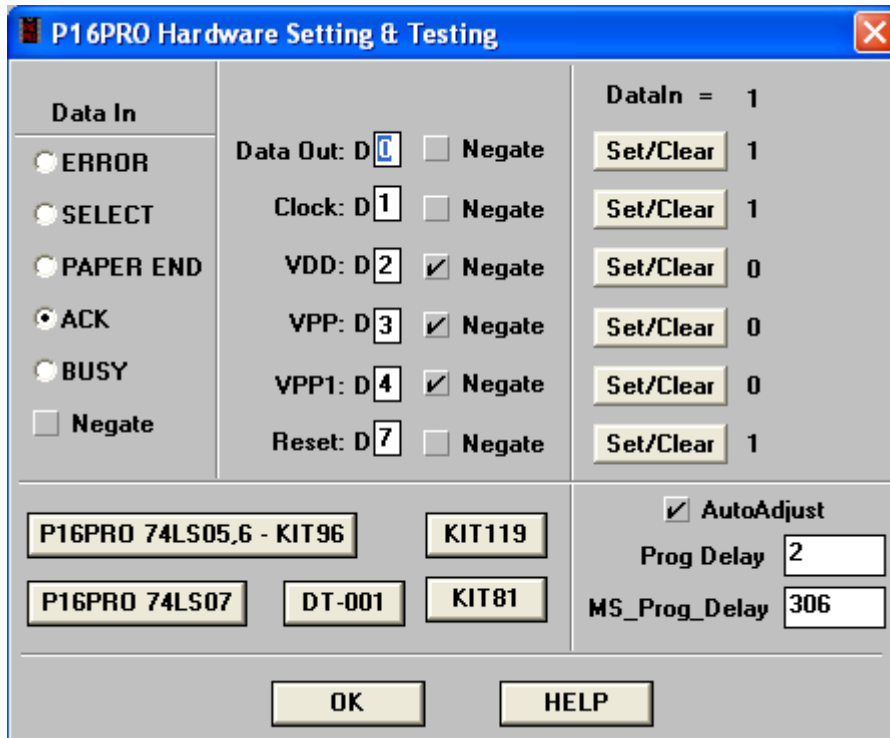
Şekil 3.16: PICALLW genel görünüm

Araç çubuğundan programlayıcı devre, yonga üreticisi ve yonga çeşidi seçilir.

File menüsünden *Open Program* ve *Open Data* komutları ile mikrodenetleyicinin program ve eeprom hafızalarına yüklenecek veriler okunur. *Settings* menüsünden *Hardware Setup/Test* komutu ile kullanılacak olan programlayıcının özellikleri belirtilir. P16PRO için KIT96 seçilmelidir. Ayrıca kod koruma osilatör türü vb. ayarlar için *Edit* menüsünden *Configuration* komutu çalıştırılır. Şekil 3.17 ve Şekil 3.18'deki gibi ekranlar gelecektir burada istenilen özellikler ayarlanarak programlama işlemine hazır hâle gelinir.



Şekil 3.17: PICALLW sigorta, kod koruma ve osilatör seçimi



Şekil 3.18: PICALLW programlayıcı donanımı özelliklerinin seçimi

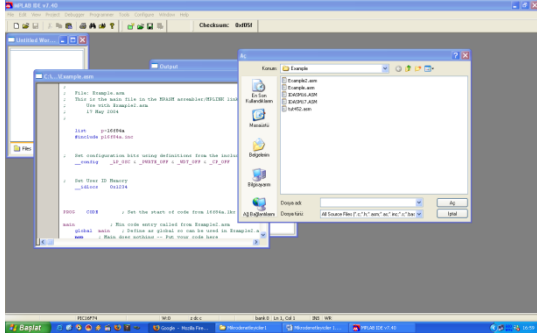
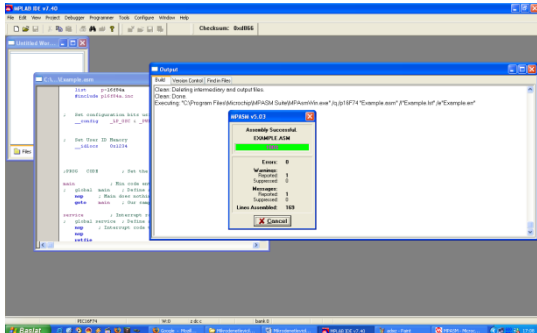
F4 tuşuna basarak ya da program komutunu çalıştırarak Binary kodlarının denetleyiciye aktarımı sağlanmış olur.

3.1.6. Denetleyicinin Denenmesi

Mikrodenetleyicinin programlanması ve simülasyonu tamamlanıp programlandıktan sonra artık denetleyicinin fiziksel ortamda testlerinin yapılmasına başlanmalıdır. Bunun için devre elemanları bir araya getirilerek baskılı devre kartı üzerinde ya da breadboard üzerinde bağlantıları gerçekleştirilebilir.

UYGULAMA FAALİYETİ

Aşağıda verilen işlem basamaklarını takip ederek mikrodenetleyici program editörünüzü açarak bulunan bir projeyi açınız ve derleyiniz.

İşlem Basamakları	Öneriler
<p>➤ MPASM yazılımını çalıştırarak örnek dosyalardan herhangi birini açınız.</p> 	<p>➤ Açarken Başlat – Programlar – MPLAB – MPLAB IDE yolunu izleyiniz.</p>
<p>➤ Project menüsünden Quickbuild seçeneği ile programı derleyiniz.</p> 	<p>➤ Derlenen programın .hex ve .lst dosyalarını açarak inceleyiniz. ➤ Derleme sonundaki hata ve uyarı mesajlarını izleyerek hatalar bulunduğu takdirde düzeltmeye çalışınız.</p>

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () MPASM programı MPLAB paketinin bir parçasıdır ve programın derlenmesi için kullanılır.
2. () MPLAB programı çalıştırma zamanında oluşabilecek hataları önceden görüp giderebilmek için bir simülatör içermektedir.
3. () MPLAB ile her türlü devrenin simülasyonu mümkün olmaktadır.

Aşağıdaki cümlelerde boş bırakılan yerlere doğru sözcükleri yazınız.

4. PICALLW uygulamasında Osilatör ve kod koruma seçenekleri F3 tuşuna basılarak komutu çalıştırılır ve ayarlanır.
5. PIC denetleyicisi ile hazırlanmış bir program kodu? programı ile simüle edilebilir.
6. programı MPLAB paketinin içerisinde derleyici görevi yapan bölümüdür.
7. programı MPLAB paketinin içinde simülatör görevi yapan bölümüdür.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru “Modül Değerlendirme” ye geçiniz.

MODÜL DEĞERLENDİRME

Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise D, yanlış ise Y yazınız.

1. () Giriş-çıkış-bellek ünitelerini tek bir chip' te barındıran yapıya mikrodenetleyici (Microcontroller) denir.
2. () Mikrodenetleyicinin içinde çevre birimlerinden hiçbiri bulunmaz.
3. () EEPROM belleğe; üzerindeki program silinip başka bir program yazılacağı zaman mikrodenetleyici programlayıcısı tarafından elektriksel sinyal gönderilerek silme veya programlama işlemi gerçekleştirilir.
4. () ADDWF f,d komutu W ile f'nin ortalamasını alır.
5. () RC, mikrodenetleyicinin çalışabileceği bir osilatör tipidir.
6. () StatusRegister 'ın 0. Biti CarryFlag (taşma biti) dir.
7. () MPLAB programı ücretli ve pahalı bir programdır.
8. () Hazırlanmış bir projeyi MPLAB programında derlemek mümkündür.
9. () PICALL programı ile binary kodlar mikrodenetleyiciye aktarılabilir.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki modüle geçmek için öğretmeninize başvurunuz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ 1'İN CEVAP ANAHTARI

1.	Doğru
2.	Doğru
3.	Yanlış
4.	Yanlış
5.	Doğru
6.	Doğru
7.	$(101)_{10}, (65)_{16}$
8.	$(111111000)_2,$ $(1016)_{10}$
9.	Yorum satırı
10.	Sabit değer
11.	ORG, interrupt
12.	Byte
13.	Bit
14.	Program Hafızası
15.	RC Osilatör
16.	Pipeline (boru hattı)

ÖĞRENME FAALİYETİ 2'NİN CEVAP ANAHTARI

1.	Yanlış
2.	Doğru
3.	Yanlış
4.	Yanlış
5.	Doğru
6.	Yanlış
7.	Doğru

ÖĞRENME FAALİYETİ 3'ÜN CEVAP ANAHTARI

1	Doğru
2	Doğru
3	Yanlış
4	Config
5	MPLAB
6	MPASM
7	MPSIM

MODÜL DEĞERLENDİRME' NİN CEVAP ANAHTARI

1	Doğru
2	Yanlış
3	Doğru
4	Yanlış
5	Doğru
6	Doğru
7	Yanlış
8	Doğru
9	Doğru

KAYNAKÇA

- ALTINBAŞAK Orhan, **Mikrodenetleyiciler ve PIC Programlama**, Şubat 2001
- http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002 (30.07.2013/ 11.00)
- http://elektroteknoloji.com/Elektrik_Elektronik/Temel_Eletronik/MPASM_Assembler_Programi_Derleyici_Kullanimi_Hakkinda_Programi_ve_Anlatimi.html (29.07.2013/ 14.00)
- <http://www.elektromania.net/default.asp?tid=405> (30.07.2013/ 10.42)
- <http://www.elektromania.net/default.asp?tid=410> (30.07.2013/ 10.29)
- <http://www.microchip.com>(08.07.2013/ 11.00)
- <http://www.mikroe.com> (09.07.2013/ 14.00)
- <http://www.antrag.org.tr>(09.07.2013/16.00)
- <http://www.hitachi.com> (10.07.2013/ 11.00)
- <http://www.elektronikhobi.net>(10.07.2013/ 14.30)
- <http://www.picallw.com> (11.07.2013/ 10.00)
- <http://www.ekenrooi.net/lcd/lcd.shtml> (11.07.2013/ 14.00)
- <http://www.phanderson.com/PIC> (12.07.2013/ 09.30)
- http://www.winpicprog.co.uk/pic_tutorial3.htm (12.07.2013/ 15.00)