

**T.C.
MİLLÎ EĞİTİM BAKANLIĞI**



MEGEP

**(MESLEKİ EĞİTİM VE ÖĞRETİM SİSTEMİNİN
GÜÇLENDİRİLMESİ PROJESİ)**

BİLİŞİM TEKNOLOJİLERİ

MİKRODENETLEYİCİLER 1

ANKARA 2007

Milli Eğitim Bakanlığı tarafından geliştirilen modüller;

- Talim ve Terbiye Kurulu Başkanlığının 02.06.2006 tarih ve 269 sayılı Kararı ile onaylanan, Mesleki ve Teknik Eğitim Okul ve Kurumlarında kademeli olarak yaygınlaştırılan 42 alan ve 192 dala ait çerçeve öğretim programlarında amaçlanan mesleki yeterlikleri kazandırmaya yönelik geliştirilmiş öğretim materyalleridir (Ders Notlarıdır).
- Modüller, bireylere mesleki yeterlik kazandırmak ve bireysel öğrenmeye rehberlik etmek amacıyla öğrenme materyali olarak hazırlanmış, denenmek ve geliştirilmek üzere Mesleki ve Teknik Eğitim Okul ve Kurumlarında uygulanmaya başlanmıştır.
- Modüller teknolojik gelişmelere paralel olarak, amaçlanan yeterliği kazandırmak koşulu ile eğitim öğretim sırasında geliştirilebilir ve yapılması önerilen değişiklikler Bakanlıkta ilgili birime bildirilir.
- Örgün ve yaygın eğitim kurumları, işletmeler ve kendi kendine mesleki yeterlik kazanmak isteyen bireyler modüllere internet üzerinden ulaşılabilirler.
- Basılmış modüller, eğitim kurumlarında öğrencilere ücretsiz olarak dağıtılır.
- Modüller hiçbir şekilde ticari amaçla kullanılamaz ve ücret karşılığında satılamaz.

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. MİKRODENETLEYİCİ VE DONANIM SEÇİMİ YAPMAK	3
1.1. Mikrodenetleyici	3
1.1.1. Mikroişlemci Nedir.....	3
1.1.2. Mikrodenetleyici Nedir.....	4
1.1.3. Programlama İçin Nelere Gereksinim Vardır	5
1.1.4. Mikrodenetleyici Çeşitleri	5
1.1.5. Mikrodenetleyici Yapısı	6
1.1.6. Bellek Çeşitleri	7
1.2. Mikrodenetleyicide Yazım Dili	8
1.2.1. Sayıların Tipi	8
1.2.2. Binary Digit(Hane) Nedir	8
1.2.3. Hexadecimal Digit(Hane) Nedir?	8
1.2.4. Binary, Decimal ve Hexadecimal Sayıların Dönüşümü	9
1.2.5. Komutların Yazılış Biçimi?	10
1.3. Mikrodenetleyici Yapısı.....	14
1.3.1. Besleme Gerilimi	14
1.3.2. Clock Düzeni ve Osilatör Çeşitleri	14
1.3.3. Reset Uçları ve Reset Devresi	15
1.3.4. Komut Akışı	15
1.3.5. Yazmaçlar.....	16
1.3.6. Program Belleği.....	18
1.3.7. Veri Belleği.....	19
1.3.8. I/O Portları	19
UYGULAMA FAALİYETİ	20
ÖLÇME VE DEĞERLENDİRME	21
ÖĞRENME FAALİYETİ-2	22
2. MİKRODENETLEYİCİ PROGRAMLAMA KARTINI YAPMAK	22
2.1. Programlama Kartı.....	22
2.1.1. Programlama Devresinin Yapımı	22
2.1.2. Bağlantı Kablosunun Yapımı	25
UYGULAMA FAALİYETİ	26
ÖLÇME VE DEĞERLENDİRME	28
ÖĞRENME FAALİYETİ-3	29
3. MİKRODENETLEYİCİ PROGRAMLAMA EDITÖRÜNÜ KULLANMAK	29
3.1. Elektronik Devre Çizim Programı ile Denetleyici	29
3.1.1. Çizim Programının Kurulması.....	29
3.1.2. Denetleyici ile Devre Oluşturulması.....	32
3.1.3. Derlenmiş Programın Denetleyiciye Tanıtılması	35
3.1.4. Çizim Programında Programın Denenmesi	35
3.2. Mikrodenetleyici Program Editörünün Kurulumu	38
3.2.1. Derleyici Editörün Yüklenmesi	38
3.2.2. Editörün Ayarlarının Yapılması	43
3.2.3. Editörün Özellikleri	44

3.2.4. Denetleyiciye Yklenecek Programın Derlenmesi ve Denenmesi	46
3.2.5. Denetleyiciye Yklenecek Programın Derlenmesi ve Denenmesi	50
3.2.6. Denetleyiciye Yklenecek Programın Derlenmesi ve Denenmesi	51
UYGULAMA FAALİYETİ	52
LÇME VE DEĞERLENDİRME	54
MODL DEĞERLENDİRME	55
CEVAP ANAHTARLARI	56
NERİLEN KAYNAKLAR	57
KAYNAKÇA	58

AÇIKLAMALAR

KOD	523EO0191
ALAN	Bilişim Teknolojileri
DAL/MESLEK	Bilgisayar Teknik Servisi
MODÜLÜN ADI	Mikrodenetleyiciler 1
MODÜLÜN TANIMI	Mikrodenetleyicilerin tanınması ve kullanımına ilişkin öğrenme materyalidir.
SÜRE	40/32
ÖN KOŞUL	
YETERLİK	Mikro denetleyici programlama kartını yapmak
MODÜLÜN AMACI	<p>Genel Amaç Mikrodenetleyici devre elemanlarını tanıyacak ve denetleyiciye uygun yazılımlar geliştirebileceksiniz.</p> <p>Amaçlar</p> <ul style="list-style-type: none">➤ Mikrodenetleyicileri tanıyacak ve yapacağınız işe uygun denetleyiciyi seçebileceksiniz.➤ Mikrodenetleyici programlama kartı hazırlayabileceksiniz.➤ Mikrodenetleyici programlama editörünü kullanabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	PIC mikrodenetleyici deney setleri, çalışır vaziyette bir bilgisayar. Uygulamalı olarak laboratuvar ortamında işlenmelidir.
ÖLÇME VE DEĞERLENDİRME	Her faaliyet sonrasında o faaliyetle ilgili değerlendirme soruları ile kendi kendinizi değerlendireceksiniz. Modül içinde ve sonunda verilen öğretici sorularla edindiğiniz bilgileri pekiştirecek, uygulama örneklerini ve testleri gerekli süre içinde tamamlayarak etkili öğrenmeyi gerçekleştireceksiniz. Sırasıyla araştırma yaparak, grup çalışmalarına katılarak ve en son aşamada alan öğretmenlerine danışarak ölçme ve değerlendirme uygulamalarını gerçekleştireceksiniz.

GİRİŞ

Sevgili Öğrenci,

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

İşte endüstrideki bu otomasyon işlemlerinin önemli bir bölümü mikrodenetleyiciler ile yapılmaktadır. Mikrodenetleyiciler ile kontrol edilmeyen cihaz yok denecek kadar azdır. Öyle ki bilgisayar teknolojisi mikrodenetleyiciler ile birlikte çamaşır makinemizin içine kadar girmiştir. 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. Gene aynı şekilde mikrodenetleyiciyi çalışır hale getirmek için gerekli donanımları ve bu küçük devreler içine hazırlanan yazılımların yüklenmesi için gerekli olan bilgiye ulaşacaksınız. Bunun yanı sıra mikrodenetleyiciler ile basit sayısal devreler hazırlayıp temellerini öğreneceksiniz.

ÖĞRENME FAALİYETİ-1

AMAÇ

Mikrodenetleyicileri tanıyacak ve ihtiyaca uygun olarak mikrodenetleyici seçimini yapabileceksiniz.

ARAŞTIRMA

Bu faaliyet öncesinde yapmanız gereken öncelikli araştırmalar şunlardır:

- Günümüzde yaygın olarak kullanılan mikrodenetleyici çeşitlerini araştırınız.
- Mikrodenetleyicilerin tarihsel gelişim sürecini araştırınız ve elde ettiğiniz bilgileri sınıf içerisinde arkadaşlarınızla tartışınız.
- RISC (Reduced Instruction Set Computer) işlemcileri ile CISC (Complex Instruction Set Computer) işlemcileri arasındaki avantaj ve dezavantajları araştırarak sınıf içerisinde tartışınız.
- Mikrodenetleyicilerin fiziksel yapısını inceleyiniz ve mikrodenetleyici kullanılarak yapılan devrelerin diğer devrelere göre fiziksel yapısını karşılaştırınız

Araştırma işlemleri için bilgisayar laboratuvarında veya ev ortamında bulunan hazır bir bilgisayar ve internet ortamını kullanınız.

1. MİKRODENETLEYİCİ VE DONANIM SEÇİMİ YAPMAK

1.1. Mikrodenetleyici

1.1.1. Mikroişlemci Nedir

Mikroişlemci, temelde bilgisayarın tüm işlemleri yapmasını sağlayan, halk tabiri ile bilgisayarın beyni olarak nitelendirilebilecek hesaplama, karar verme ve yönetim mekanizmasıdır.

Bir mikro işlemcinin üzerinde mantık kapılarının bileşiminden oluşan ve çeşitli işlevleri yürütmeye yarayan birimler bulunur. Bu birimler arasında aritmetik-mantık birimi, kontrol birimi, girdi-çıkı ve küçük bir miktar bellek vardır. Bu temel birimlerin dışında mikro işlemcinin kullanım alanına bağlı olarak farklı görevlerde özelleşmiş birimler de

bulunabilir. Ancak bir mikro işlemcinin ana işlevlerini yerine getirebilmesi için temel birimler yeterlidir.

Bir mikroişlemci çoğunlukla kendisine bağlanacak bazı entegre devrelerle birlikte çalışmak için tasarlanır. Örneğin mikroişlemcinin çalıştıracağı programın ve kullanacağı verinin yüklü olduğu, yazılabilir ve okunabilir, hızlı bir bellek birimi (genellikle RAM) ve sisteme güç verilmezken programı saklayabilecek bir bellek birimi (genellikle bir çeşit ROM) mikroişlemcilerin olmazsa olmaz çevre birimleridir. Bunun dışında, mikroişlemciler genellikle elektronik devrelerde kontrol mekanizması görevinde bulundukları için kontrol edecekleri cihazlarla da bağlantı kurlmaları gerekecektir.



Resim 1.1: Bir Mikroişlemci



Resim 1.2 Bir Mikrodenetleyici

1.1.2. Mikrodenetleyici Nedir

Mikrokontrolörler (mikrodenetleyiciler) tek bir silikon yonga üstünde birleştirilmiş bir mikroişlemci, veri ve program belleği, sayısal (lojik) giriş ve çıkışlar (I/O), analog girişler ve daha fazla güç veren ve işlev katan öteki çevre birimleri (zamanlayıcılar, sayaçlar, kesiciler, analogtan sayısala çeviriciler, vb.) barındıran mikrobilgisayarlardır.

En basit mikrokontrolör mimarisi bir mikroişlemci, bir bellek ve giriş ve çıkıştan (I/O) oluşur. Mikroişlemci merkezi işlemci ünitesi (CPU - Central Processing Unit) ve bir kontrol ünitesinden (CU - Control Unit) oluşur. CPU mikroişlemcinin beynini oluşturur, aritmetik ve mantıksal işlemlerin gerçekleştirildiği yerdir. CU kontrol ünitesi mikroişlemcinin dahili işlemlerini kontrol eder ve istenen komutları yerine getirmek için kontrol sinyallerini diğer bölümlere gönderir.

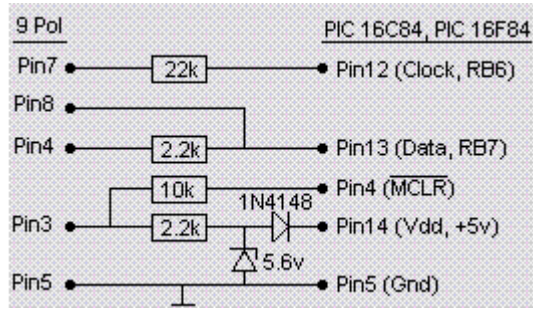
Görüldüğü gibi mikroişlemci ile mikrodenetleyici birbirine yakın teknolojiler olmakla birlikte en önemli farkları mikrodenetleyicilerin kendi üzerinde belleklerini ve çevre birimlerini bulundurmasıdır. Bu sebeple çok yüksek hafıza, hız ve işlem yeteneğinin gerekmediği durumlarda mikrodenetleyiciler tercih edilir.

1.1.3. Programlama İçin Nelere Gereksinim Vardır

Mikrodenetleyicileri programlamak için iki önemli şeye ihtiyaç vardır. Birincisi mikrodenetleyicinin yapısına uygun bir programlayıcı devre, ikincisi de bu devreye hazırladığınız derlenmiş programı aktaracak bir yazılım.

Programlayıcı Donanımı:

Şekil 1.1’de PIC16F84 mikrodenetleyicisi için hazırlanmış seri porttan haberleşen basit bir devre göreceksiniz.



Şekil 1.1: Basit bir programlayıcı devre şeması

Programlayıcı Yazılımı:

Mikrodenetleyicilere hazırladığınız programı yükleyebilmek için mikrodenetleyicinin yapısına uygun bir elektronik devre ve bu devreye göre hazırlanmış bir programa ihtiyaç vardır. Birçok PIC mikrodenetleyicisi için hazırlanmış programlayıcı devreler için MPLAB program paketinin içerisinde programlayıcı mevcuttur ancak yine de bunun dışında kendine has yazılımları mevcuttur.

Şekil 1.1’deki programlayıcı devre için aşağıda linkte verilen yazılımı kullanabilirsiniz.

<http://people.freenet.de/dl4yhf/winpic/winpicpr.zip>

1.1.4. Mikrodenetleyici Çeşitleri

Temelde mikroişlemci mimarisi iki çeşittir. Bunlar RISC (Reduced Instruction Set Computer: Azaltılmış komut seti) tabanlı işlemciler ve CISC (Complex Instruction Set Computer: Karmaşık komut seti) tabanlı işlemcilerdir. Mikrodenetleyicilerin de aynı şekilde RISC ve CISC mimarisine göre türleri vardır. Yaygın olarak kullanılan mikrodenetleyiciler mimari açısından bu iki işlemci sınıfından birine aittir.

Üretici firma bakımından ise bilinen çok sayıda mikrodnetleyiciler vardır.

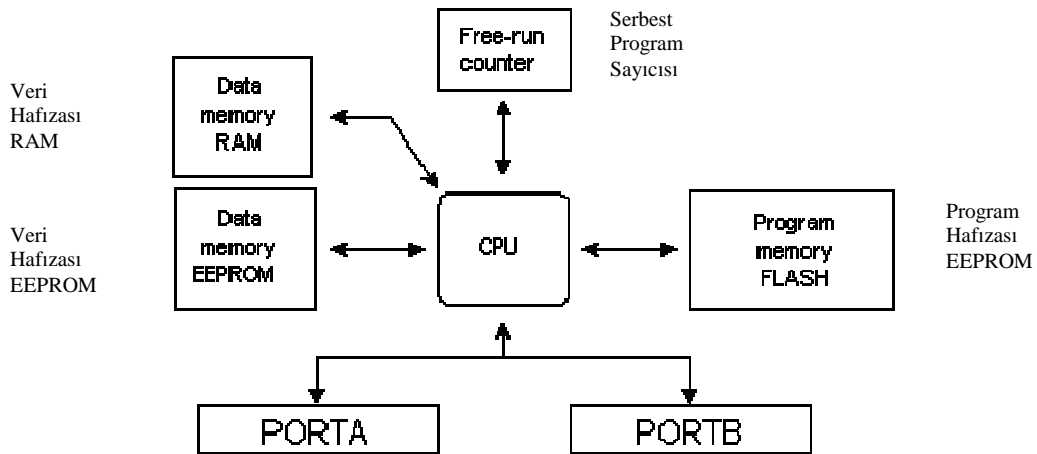
Bunlar;

- Microchip firmasının PIC mikrodnetleyicileri (RISC),
- Intel firmasının MCS51 (8051) mikrodnetleyicileri (CISC)
- Atmel firmasını AVR mikrodnetleyicileri (RISC)
- Motorola FreeScale mikrodnetleyicileri ve daha birçok firmanın burada sayamayacağımız kadar çok çeşitleri olan denetleyicileridir.

<http://www.cpubpages.com/store/index.php> internet adresinden çeşitli mikrodnetleyiciler ile ilgili bazı temel bilgilere ulaşabilirsiniz.

1.1.5. Mikrodnetleyici Yapısı

Mikrodnetleyiciler yapıları birbirinden farklı olmakla birlikte temelde çok fazla benzerlikler içerir. Şekil 1.2’de PIC mikrodnetleyicilerinden PIC16F84 iç yapısı verilmiştir.



PIC16F84 microcontroller outline

Şekil 1.2: 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 mikrodnetleyici çalışmak için ek hiçbir devreye ihtiyaç duymayacaktır.

1.1.6. Bellek Çeşitleri

Bellekler çeşitli özelliklere göre sınıflandırılabilir.

➤ **Fiziksel Yapılarına Göre Bellekler:**

- Elektronik bellekler (RAM, ROM, Flash vs.)
- Elektromanyetik bellekler (disket, sabit disk vs.)
- Mekanik bellekler (delikli kart vs.)
- Optik bellekler (CD-ROM, DVD-ROM vs.)

➤ **Veri Saklama Şekline Göre Elektronik Bellekler.**

- RAM (Random Access Memory) : Geçici belleklerdir. Devrenin enerjisi kesilirse içindeki bilgiler kaybolur. Veriler istenildiği gibi değiştirilebilir ve silinebilir.
- ROM (Read Only Memory) : Kalıcı belleklerdir. Enerji kesildiğinde bilgileri saklamaya devam eder. İçindeki veriler hiçbir suretle silinemez ve değiştirilemez.
- EEPROM (Electrically Erasable Programmable ROM) : Kalıcı belleklerdir. Enerji kesintisinde bilgileri saklamaya devam ederler. Ancak istenildiği zaman belli yöntemler ve devreler sayesinde içindeki bilgiler silinip değiştirilebilmektedir.
- Flash bellek: Bu tür bellekler tıpkı EEPROM bellekler gibidir ancak çok daha hızlı ve yüksek kapasiteli olabilmektedir. Günümüzde memorstick ve pendrive gibi çeşitli isimlerle anılan taşınabilir bellek olarak karşımıza çokça çıkmaktadırlar.

➤ **Veri saklama Şekline Göre Optik Bellekler:**

- CD-ROM (Compact Disk Read Only Memory): Bilgi kaydedilemeyen, sadece okunabilen CD (Compact Disk) çeşididir. İçindeki bilgiler değiştirilemez ve silinemez.
- CD-R (Compact Disk Recordable): İçeriğine bilgi yazılabilen ancak yazılan bilgiler silinemeyen ve değiştirilemeyen CD çeşididir.
- CD-RW (Compact Disk Rewritable): Tekrar tekrar bilgi yazılabilen CD türüdür. İçindeki bilgiler çeşitli yöntemlerle silinebilmektedir. Tekrar yazılabilmesi için diskin tamamen silinmesi gerekir.
- DVD-ROM (Digital Versatile Disk ROM): CD-ROM yerine kullanılacak çok daha yüksek kapasiteli optik medya formatıdır.
- DVD±R (DVD-Recordable): CD-R yerine kullanılacak yüksek kapasiteli optik medya formatı.
- DVD±RW (DVD-Rewritable): CD-RW yerine kullanılacak yüksek kapasiteli optik medya formatı.
- DVD+RAM (DVD Random Access Memory): DVD±RW gibidir ancak, içindeki bilgileri değiştirmek için diskin tamamının silinmesi gerekmez. Bilgiler istenildiği gibi silinir ve değiştirilir.

➤ Elektromanyetik Bellekler

Bu tür bellekler bilgileri çeşitli tekniklerle manyetik ortamda saklar ve içerisindeki bilgiler istenildiği gibi değiştirilebilmekte ve silinebilmektedir. Enerji kesintisinde bilgileri saklamaya devam eder yani kalıcı bir bellek türüdür.

➤ Mekanik Bellekler

(Artık günümüzde kullanılmamaktadır). Mikrodenetleyiciler içerisinde RAM bellek ve ROM bellek yerinde daha geliştirilmiş bir bellek türü olan EEPROM ya da flash bellek bulunmaktadır. Mikrodenetleyici iç yapısına bakıldığında üç önemli hafıza göze çarpar. Program Memory (Program kodlarının bulunduğu bellek) EEPROM bellek türündedir. Data memory ise iki kısımdır. Geçici verilerin tutulduğu RAM ve sürekli verilerin ve sabit değerlerin tutulduğu EEPROM bellek.

Mikrodenetleyiciler içerisinde bu üç bellek dışında mikrodenetleyici çekirdeğinde yer alan Registers (yazmaçlar) vardır. Bunlar denetleyici türüne göre farklı olmakla birlikte PIC denetleyicilerinde RAM bellek içinde yer alırlar.

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 Digit(Hane) Nedir

İkili sayı sisteminde ifade edilen sayının her bir basamağı bir binary digit olarak değerlendirilir. Binary digitler 0 ya da 1 değeri dışında değer alamazlar. Binary digitlere kısaca bit denilir. 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. Hexadecimal Digit(Hane) Nedir?

On altılı sayı sisteminde ifade edilen verilerin her bir basamağı hexadecimal digit olarak adlandırılır. PIC mikrodenetleyicisinde on altılı bir sayının kullanımı şu şekilde olabilir.

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

1.2.4. Binary, Decimal ve Hexadecimal Sayıların Dönüşümü

1. Binary – Decimal dönüşümü: İkili sayı sisteminden onlu sayı sistemine dönüşüm için her bit, bit değeri ile çarpılarak bu çarpımlar toplanmaktadır. Bu durum Şekil 1.3'te izah edilmiştir. Aynı şekilde onlu sayı sisteminden ikili sayı sistemine geçiş için ise sayı sürekli ikiye bölünerek kalanlar yan yana yazılır. Bu Şekil 1.4'te izah edilmiştir.

$$\begin{array}{r} 01011101 \\ \begin{array}{l} 1 \times 1 = 1 \\ 0 \times 2 = 0 \\ 1 \times 4 = 4 \\ 1 \times 8 = 8 \\ 1 \times 16 = 16 \\ 0 \times 32 = 0 \\ 1 \times 64 = 64 \\ 0 \times 128 = 0 \end{array} \\ + \\ \hline 1 + 0 + 4 + 8 + 16 + 0 + 64 + 0 = 93 \end{array}$$

Şekil 1.3: Binaryden Decimale geçiş

$$\begin{array}{r} 142 \div 2 = 71 \text{ remainder } 2 \\ 71 \div 2 = 35 \text{ remainder } 1 \\ 35 \div 2 = 17 \text{ remainder } 1 \\ 17 \div 2 = 8 \text{ remainder } 1 \\ 8 \div 2 = 4 \text{ remainder } 0 \\ 4 \div 2 = 2 \text{ remainder } 0 \\ 2 \div 2 = 1 \text{ remainder } 0 \\ 1 \div 2 = 0 \text{ remainder } 1 \end{array}$$

10001110

Şekil 1.4: Decimalden Binarye geçiş

2. Binary – Hexadecimal dönüşümü: İkili sayı sisteminden on altılı sayı sistemine geçişte her dört bitin hex değeri hesaplanarak yan yana yazılır. Aynı şekilde on altılı sayı sisteminden ikili sayı sistemine geçişte her basamağın Binary değeri hesaplanarak yan yana yazılır. Bu durum Şekil 1.5 ve 1.6'da izah edilmiştir.

$$\begin{array}{c} 11001011 \\ \hline \text{C} \quad \text{B} \end{array} = \text{CB}$$

Şekil 1.5: Binary'den hexadecimal'e geçiş

$$\begin{array}{c} \text{A8} = 10101000 \\ \hline 1000 \\ 1010 \end{array}$$

Şekil 1.6: Hexadecimal'den binary'e geçiş

3. Decimal – Hexadecimal dönüşümü: Onlu sayı sisteminden on altılı sayı sistemine geçişte sayı sürekli on altıya bölünerek kalanlar yan yana yazılır. Aynı şekilde on altılı sayıdan onlu sayıya geçiş için her bir basamak kendi değeri ile çarpılarak çarpımlar toplanır. Böylece dönüşüm gerçekleşir. Bu Şekil 1.7 ve 1.8'de izah edilmiştir.

$$\begin{array}{r} 155 \div 16 = 9 \text{ remainder } 11 \\ \hline 11 \end{array}$$

Şekil 1.7: Decimal'den Hexadecimal'e geçiş

$$\begin{array}{r} 7F2 \\ \begin{array}{l} 2 \times 1 = 2 \\ F \times 16 = 240 \\ 7 \times 256 = 1792 \end{array} \\ + \\ \hline 2 + 240 + 1792 = 2034 \end{array}$$

Şekil 1.8: Hexadecimal'den Decimale geçiş

1.2.5. Komutların Yazılış Biçimi?

Her mikrodenetleyici için kendine has derleyiciler bulunmaktadır. ve Derleyicilerin 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 halinde yazılım koşulunu getirir 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: İki 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 mikrodenetleyicilerinde halt komutu yoktur. Bunun yerine sonsuz döngüler kullanılır.

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	REGISTER, DESTINATION
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 **register (yazmaç)** ile yaptığı işlem sonucunu **Destination (hedef)** ile belirlenen hedefe yazar. Burada **Destination (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	REGISTER, BIT
BCF	STATUS, 5 ;Bu komut STATUS yazmacının 5. biti'ni 0 yapar.

Komut **register (yazmaç)** ile belirtilen yazmacın **bit (hane)** ile belirtilen basamağı ile işlem yapar.

Sabit işleyen 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.

Sayı ve karakter yazılış biçimleri:

- Hexadecimal (On altılı) Sayılar

0x03
03
03h
h'03'

Bu ifadelerin hepsi 03 on altılı sayısını temsil eder.

- Binary (İkili) Sayılar

b'00010101'

İkili sayılar yalnızca yukarıdaki gibi yazılabilir.

- Decimal (Onlu) Sayılar

d'122'

Onlu sayılar sadece yukarıdaki gibi yazılabilir.

- ASCII Karakterler

's'

ASCII karakterleri sadece yukarıdaki gibi yazılabilir.

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.

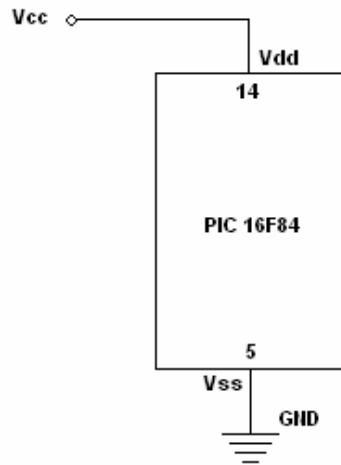
Komutlar		Açıklamalar	Çevrim	Bayraklar	Notlar
ADDWF	f, d	W ve f'i topla	1	C,DC,Z	1,2
ANDWF	f, d	W ile f'i mantıksal VE işlemine tabi tut	1	Z	1,2
CLRF	f	f'i temizle	1	Z	2
CLRW	-	W'yi temizle	1	Z	
COM F	f, d	f'in tersini al	1	Z	1,2
DECF	f, d	f'i azalt	1	Z	1,2
DECFSZ	f, d	f'i azalt, sonuç 0 ise zıpla, sonraki 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'yi f'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'teki nibılların (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
BTFSZ	f, b	f'in bitine bak, 0 ise zıpla, sonraki komutu geç	1(2)	Hiçbiri	3
BTFS	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	
CLRWD	-	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 değişmekle birlikte diğer dijital devrelerle birlikte daha rahat kullanılabilmesi için 5V DC uygun bir değerdir (Şekil 1.9).

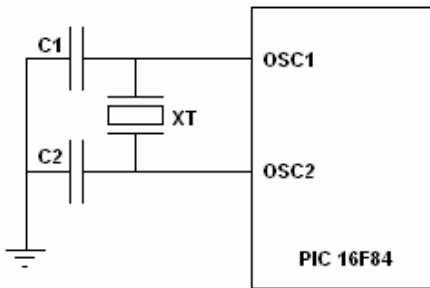


Şekil 1.9: PIC 16F84 besleme gerilimi

1.3.2. Clock Düzeni ve Osilatör Çeşitleri

PIC16F84 mikrodenetleyicisi farklı osilatör tipleri ile çalışabilmektedir. Bu osilatör tipleri şunlardır.

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

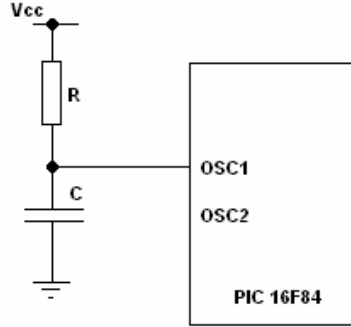


Şekil 1.10: PIC için LP/XT/HS osilatör bağlantısı

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

LP / XT / HS Kristal ya da seramik rezonatör elemanları kullanıldığında en basit devre Şekil 1.10'daki gibi gerçekleştirilebilir. **Eğer seramik rezonatör kullanılıyorsa C1 ve C2 kondansatörlerine gerek yoktur.**

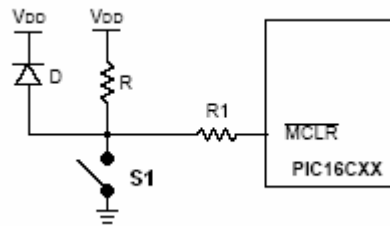


Şekil 1.11: PIC 16F84 için RC Osilatörü bağlantısı

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 K Ω ile 100 K Ω arasında bir değerde olması önerilir. Aynı şekilde kondansatörde 20 pF değerinin üzerinde olmalıdır (Şekil 1.11).

1.3.3. Reset Uçları ve Reset Devresi

PIC mikrodenetleyicisi besleme gerilimi aldığı anda programın başlangıç adresinden itibaren çalışması için dahili Power on Reset devresi bulunmaktadır. Ancak bazen kasıtlı olarak programın çalışmasını başlangıç durumuna almak gerekebilir. Bu durumda Reset uçları kullanılmaktadır (Şekil 1.12).

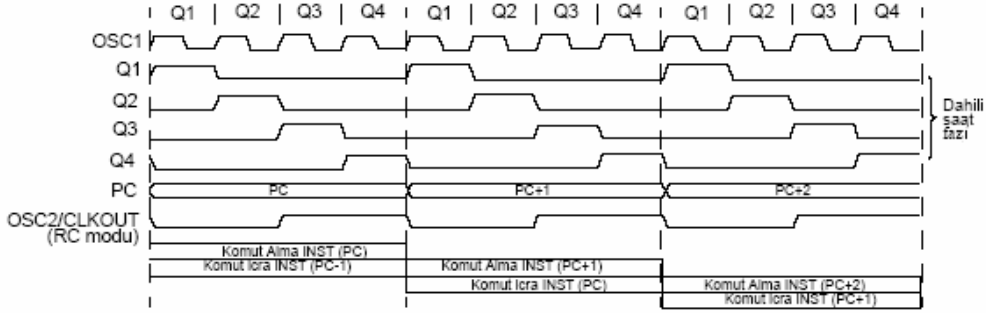


Şekil 1.12: PIC 16F84 için Reset devresi

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 alacak şekilde, komut alınması ve icra edilmesi pipeline (boruhattına) yerleştirilir. Boru hattı 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 arttırılması 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 mikrodnetleyicisinde 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 mikrodnetleyicisinde özel amaçlar için kullanılan bazı yazmaçlar vardır. Bu yazmaçlar şunlardır:

STATUS Register:

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

bit 0 – C (Carry Flag):

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 (Digit Carry Flag):

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 (Power Down Flag):

Enerji kesinti bitidir. Bu bit mikrodnetleyici ilk çalıştığında ve CLRWDT komutu ile Lojik 1 olur. Sleep komutu çalışınca Lojik 0 olur.

bit 4 – TO (Time Out Flag):

Watchdog taşma bitidir. PIC’e enerji verilince ve CLRWDT, SLEEP komutları ile Lojik 1 olur. Watchdog Timer saymayı tamamlayınca Lojik 0 olur.

bit 6:5 – RP1:RP0 (Register Bank Select Bits):

Yazmaç bank seçim biti: Bu bit’ler 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.

OPTION Register:

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

bit 2:0 – PS2,PS1,PS0 (Prescaler 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

bit 3 – PSA (Prescaler Assignment 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 bit’idir. 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 bit’idir. Lojik 0 ise ¼ osilatör frekansı seçilir. Lojik 1 ise harici dijital sinyal (RA4/TOCKI) seçilir.

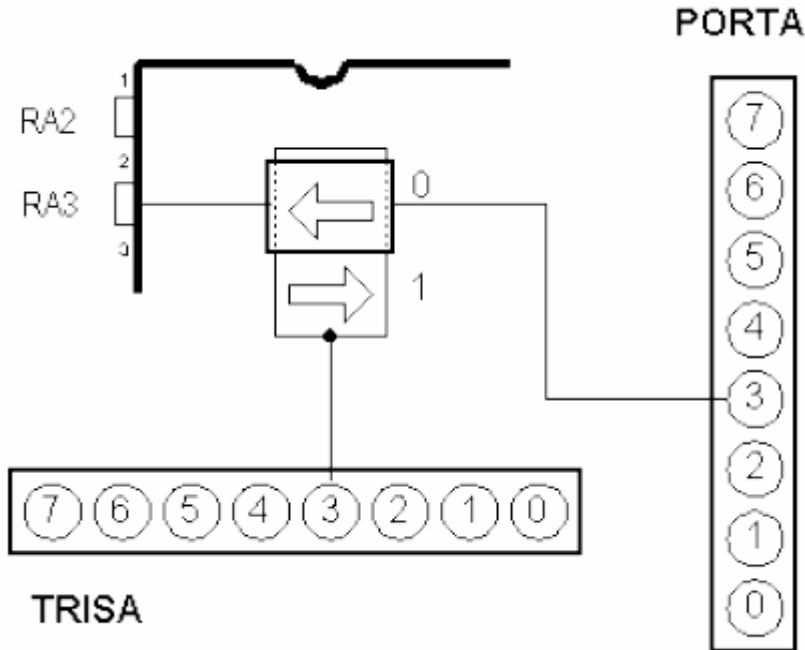
1.3.7. Veri Belleği

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 bit’i 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 external interrupt için kullanılabilir.



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

UYGULAMA FAALİYETİ

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"> ➤ Veri kitapçıklarını temin ederken, www.microchip.com , www.intel.com , www.atmel.com gibi internet bağlantılarından yararlanınız.
 <ul style="list-style-type: none"> ➤ Veri kitapçılarından denetleyicilerin minimum donanımlarını çiziniz. Denetleyicilerin birbirine göre avantaj ve dezavantajlarını defterinize not ediniz. 	<ul style="list-style-type: none"> ➤ Farklı denetleyicilerin maliyet açısından farklılıklarını kıyaslayınız.
 <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

A. OBJEKTİF TESTLER (ÖLÇME SORULARI)

Aşağıdaki soruların cevaplarını doğru ve yanlış olarak değerlendiriniz.

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 ya da değiştirmek mümkün değildir. (.....)
5. RAM bellekler hızlıdır fakat devrenin enerjisi kesildiğinde içindeki veriler silinir. Bu nedenle geçici bellek olarak kullanılırlar. (.....)
6. İkili sayı sisteminde ifade edilen sayının her bir basamağı bir binary digit olarak değerlendirilir. (.....)

Aşağıdaki cümlelerde bulunan boşlukları en uygun şekilde doldurunuz.

7. (1100101)₂ sayısını decimal ve hexadecimal sistemde gösteriniz. _____, _____
8. (3F8)₁₆ sayısını binary ve decimal sistemde gösteriniz. _____, _____
9. MPASM derleyicisinde satır başına ; (noktalı virgül) kullanıldığında o satır _____ olarak değerlendirilir.
10. Bazı komutlarda doğrudan rakamlar kullanılabilir. Bunlara _____ denilir.
11. _____ deyimi iki amaç için kullanılabilir. Birincisi programın başlangıç adresini belirlemek için. İkincisi ise _____ alt programı başlangıç adresini belirlemektir.
12. _____ yönlendirmeli komutlar 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 anahtarı ile karşılaştırınız. Doğru cevap sayınızı belirleyerek kendinizi değerlendiriniz. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt yaşadığınız sorularla ilgili konuları faaliyete geri dönerek tekrar inceleyiniz

ÖĞRENME FAALİYETİ-2

AMAÇ

PIC 16F84 mikrodnetleyicisi için basit bir programlama kartı yapımını gerçekleştirecek ve programlamak için gerekli olan yazılımı inceleyebileceksiniz.

ARAŞTIRMA

Bu faaliyet öncesinde yapmanız gereken öncelikli araştırmalar şunlardır:

- Çeşitli mikrodnetleyiciler için programlama kartları ve yazılımları hakkında araştırma yapınız.
- Birden fazla mikrodnetleyiciye uygun programlama kartlarını araştırınız avantaj ve dezavantajlarını belirleyiniz.
- Mikrodnetleyici programlayabilmek için gerekli olan yazılımları internetten temin ediniz.
- Araştırma işlemleri için bilgisayar laboratuvarında veya ev ortamında bulunan hazır bir bilgisayarı ve internet ortamını kullanınız. Kazanmış olduğunuz bilgi ve deneyimleri arkadaş gurubunuz ile paylaşınız.

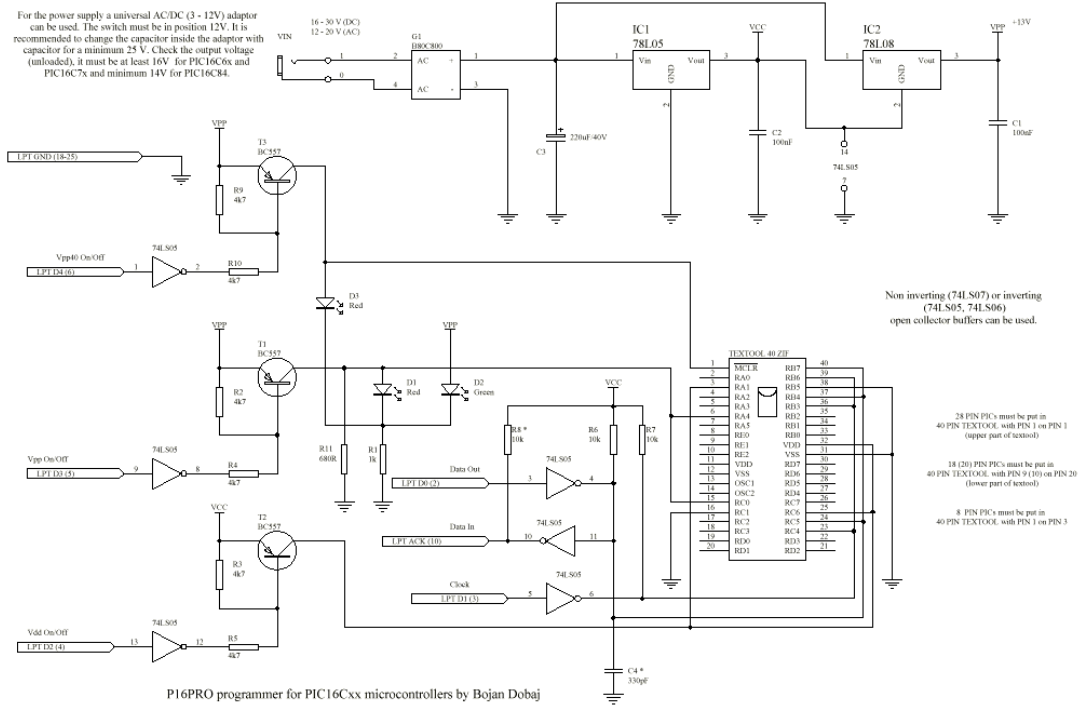
2. MİKRODNETLEYİCİ PROGRAMLAMA KARTINI YAPMAK

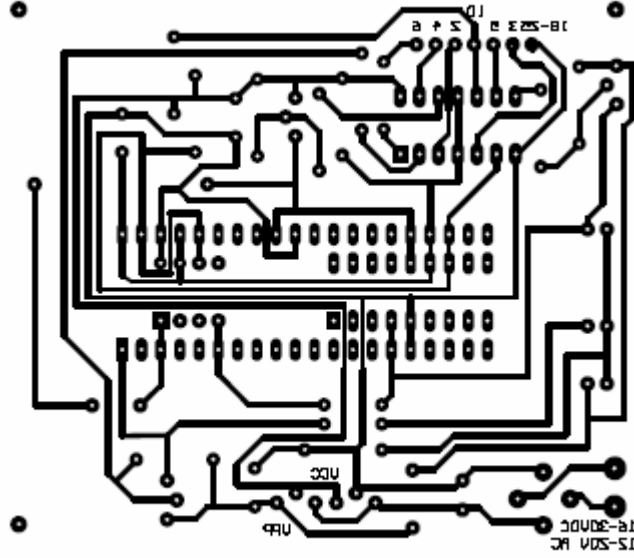
2.1. Programlama Kartı

2.1.1. Programlama Devresinin Yapımı

Mikrodnetleyicilerle yapılan devrelerin dizaynı yapılıp uygulama aşamasına gelindiğinde, denetleyici içerisine yazılım yüklemek için çeşitli programlama cihazlarına gereksinim vardır. Ancak PIC mikrodnetleyicileri için bu cihazlar uygulaması basit ve etkili devrelerdir.

Şekilde paralel port ile haberleşen P16Pro programlama kartının devre şeması görülmektedir.





Resim 2.5: P16PRO baskılı devre alt yüzeyi




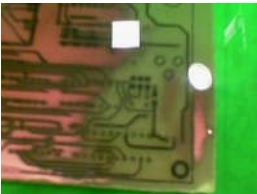
Malzeme Listesi:

No	Adet	Malzeme Adı	Tip
1	1	TEXTTOOL 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

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

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 ütüleyiniz. 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 kalemi ile düzeltiniz.</p>
<p>➤ Hazırladığınız plaketi asit içerisine atarak (3:1 oranında tuz ruhu ve Perhidrol karışımı) boyanmamış bölümlerin çözülerek kaybolmasını bekleyiniz.</p> 	<p>➤ Asitin çözülmesi esnasında açığa çıkan gazı solumayınız ve açık havada işlemi gerçekleştirin. Aksitakdirde 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 kullanın.</p>
<p>➤ Lehimleme işlemini yaparak devreyi hazırlayınız.</p> 	<p>➤ Lehimleme yaptıktan sonra kalan lehim atıklarını temizleyiniz.</p>

ÖLÇME VE DEĞERLENDİRME

A. OBJEKTİF TESTLER (ÖLÇME SORULARI)

Aşağıdaki soruların cevaplarını doğru ve yanlış olarak değerlendiriniz.

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 anahtarı ile karşılaştırınız. Doğru cevap sayınızı belirleyerek kendinizi değerlendiriniz. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt yaşadığınız sorularla ilgili konuları faaliyete geri dönerek tekrar inceleyiniz

ÖĞRENME FAALİYETİ-3

AMAÇ

Mikrodenetleyiciler için hazırlanmış programları fiziksel testlerden önce simülasyon yöntemi ile analiz etmek için kullanılan programların temellerini öğrenip, mikrodenetleyici kodlarının derlenmesi için kullandığımız derleyicilerin çalıştırılmasını ve kullanımını öğrenebileceksiniz.

ARAŞTIRMA

Bu faaliyet öncesinde yapmanız gereken öncelikli araştırmalar şunlardır:

1. Mikrodenetleyici temeli devrelerin çizimini ve simülasyonunu gerçekleştiren programlar hakkında bir ön araştırma yapınız. Elde ettiğiniz bilgileri sınıf ortamında tartışınız.

2. Araştırma sonucunda belirlediğiniz bazı programların birbirlerine göre avantaj ve dezavantajlarını tartışınız.

3. MPASM programının yapısı ve çalışması hakkında bir ön araştırma yapınız.

Araştırma işlemleri için bilgisayar laboratuvarında veya ev ortamında bulunan hazır bir bilgisayarı ve internet ortamını kullanınız. Kazanmış olduğunuz bilgi ve deneyimleri arkadaş gurubunuz ile paylaşınız.

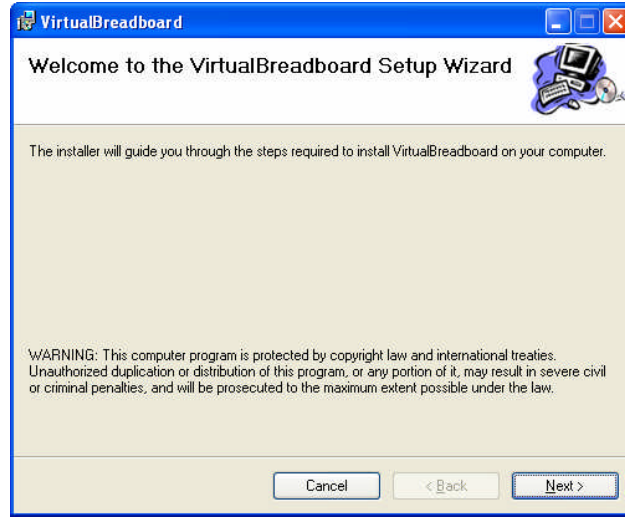
3. MİKRODENETLEYİCİ PROGRAMLAMA EDITÖRÜNÜ KULLANMAK

3.1. Elektronik Devre Çizim Programı ile Denetleyici

3.1.1. Çizim Programının Kurulması

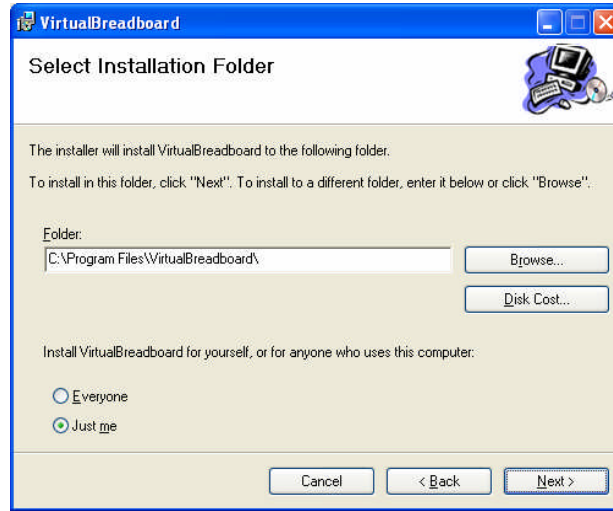
Çeşitli mikrodenetleyiciler için farklı türde ve birçok simülatör ve devre editörü olmasına rağmen, kullanım kolaylığı ve gerçek zamanlı simulasyon yapması nedeni ile VirtualBreadBoard yazılımı tercih edilebilir bir seçenek olarak karşımıza çıkmaktadır. Programın avantajı birçok noktadaki devre davranışını çok basit araçlarla izleyebilmek mümkündür.

Bu programı kurmak için www.virtualbreadboard.com işletim sisteminize uygun olarak hazırlanmış ücretsiz programı bilgisayarınıza indirmelisiniz. Daha sonra çıkan sıkıştırılmış dosyayı bilgisayarınızda herhangi bir klasöre açarak oluşan Setup.exe isimli uygulamayı çalıştırın. Şekildeki gibi bir pencere açılacaktır.



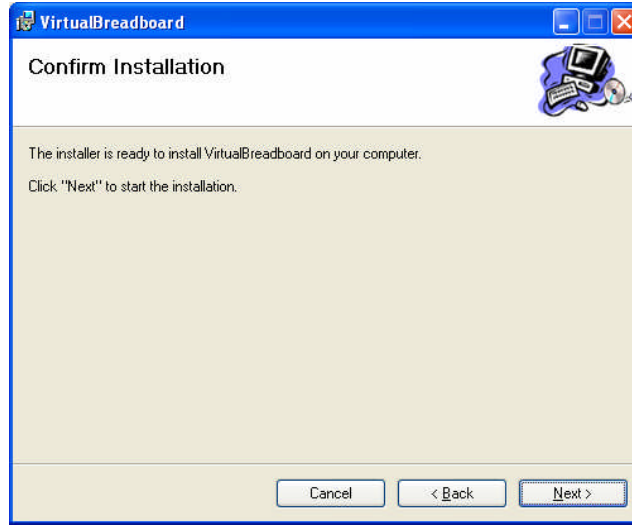
Şekil 3.1: VirtualBreadboard Simülatörünün kurulumu adım 1

Yönergeleri izleyerek kurulumla Next komutu ile devam ediyoruz.



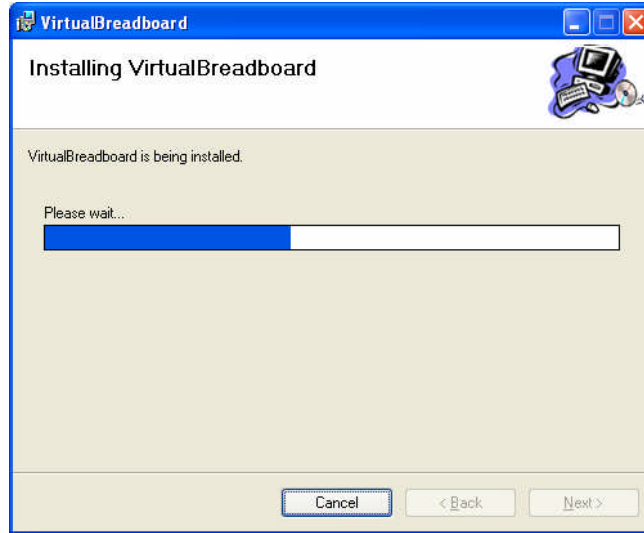
Şekil 3.2: VirtualBreadboard Simülatörünün kurulumu adım 2

Bu kısımda programın kurulumunun yapılacağı klasör adını belirtmek gerekmektedir. Eğer önerilen klasör dışında bir klasöre kurulum yapmak isterseniz Browse komutu ile seçebilirsiniz. Next ile devam ediyoruz.



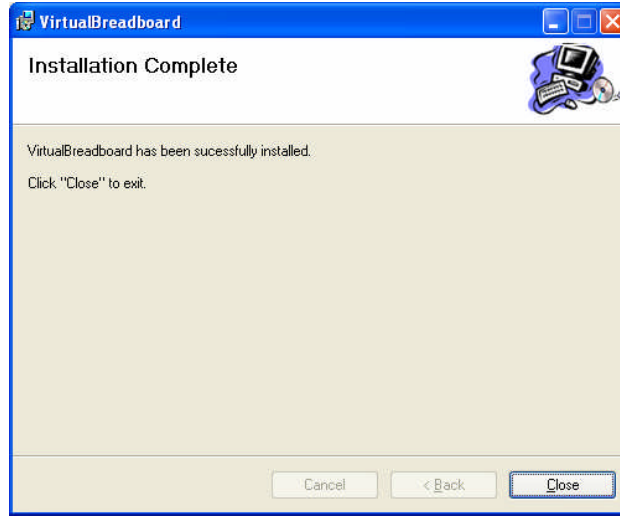
Şekil 3.3: VirtualBreadboard Simülatörünün kurulumu adım 3

Kurulumu onaylamak için Next komutunu seçiyoruz.



Şekil 3.4: VirtualBreadboard Simülatörünün kurulumu adım 4

Dosyaların kopyalanması işlemi başlamış oluyor, bu esnada herhangi bir şey yapmayacağız.

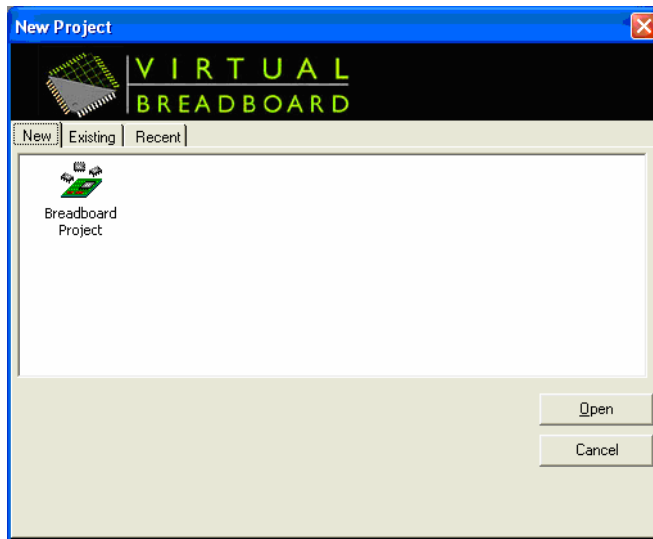


Şekil 3.5: VirtualBreadboard Simülatörünün kurulumu adım 5

Kopyalama işlemi tamamlanınca çıkan ekranda VirtualBreadboard has been successfully installed mesajını alırsak program sorunsuzca kurulmuş demektir. Close ile kurulumu tamamlıyoruz.

3.1.2. Denetleyici ile Devre Oluşturulması

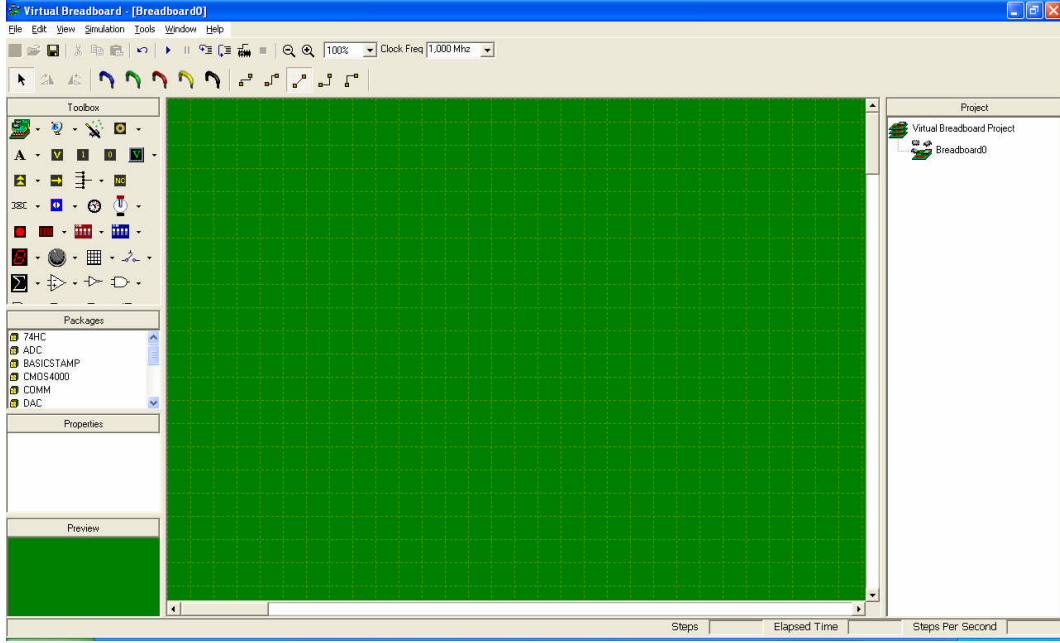
Bunun için VirtualBreadboard uygulaması Başlat menüsünden ilgili program tıklanarak çalıştırılır. Şekildeki gibi bir ekran karşımıza çıkacaktır



Şekil 3.6: VirtualBreadboard Simülatörü ilk açılış

Yeni bir proje hazırlayacak isek Breadboard Project komutunu seçmeliyiz. Aksi takdirde Open komutu ile daha önceden hazırlamış olduğumuz projeler üzerinde çalışmaya devam edebiliriz.

İstediğimiz seçimi yaptıktan sonra programı başlatınca şekildeki gibi bir çalışma ortamı ile karşılaşırız.

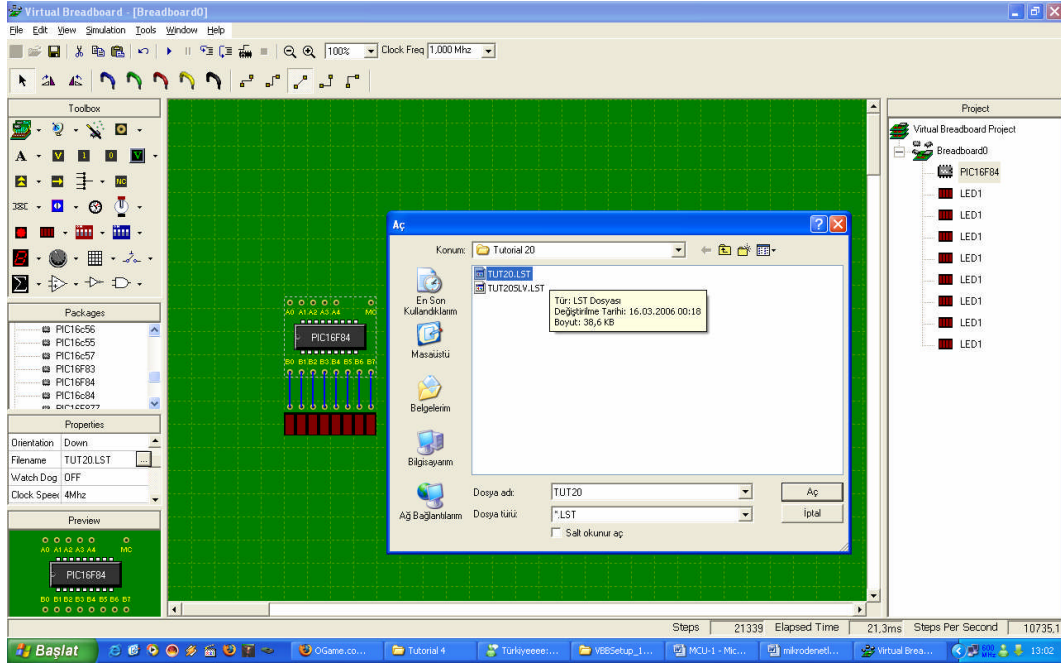


Şekil 3.7: VirtualBreadboard Simülatörü çalışma sayfası

Burada bizim için en önemli kısım sol taraftaki devre elemanlarının simgeleri ve Packages ile Properties kutularıdır. Devre elemanının türü Packages'ten seçilir ve özellikleri Properties'ten ayarlanır. Devre elemanlarını sıra ile seçerek board üzerine yani yeşil alana bırakıyoruz.

3.1.3. Derlenmiş Programın Denetleyiciye Tanıtılması

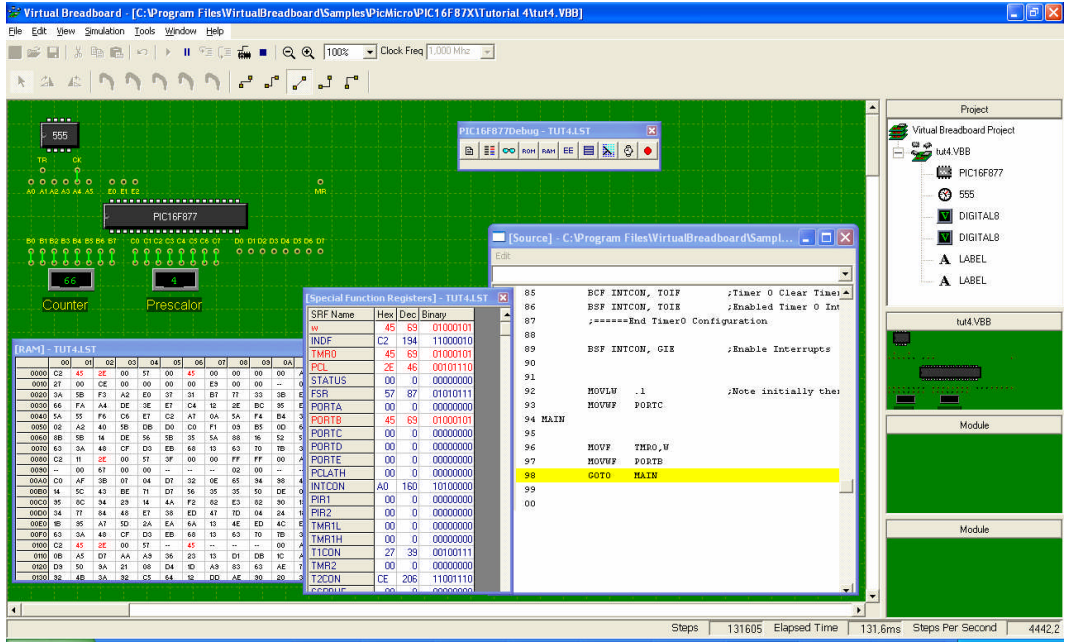
Derlenmiş programı denetleyiciye aktarmak için Properties kutusundan Filename seçeneği ile Hex uzantılı derlenmiş binary dosyayı ya da lst uzantılı derleyici çıktısını (komutların derlenmemiş halini içermektedir) aşağıdaki gibi seçiyoruz. Böylece devremiz simülasyona hazır hale gelmektedir.



Şekil 3.10: Derlenmiş programın denetleyiciye tanıtılması

3.1.4. Çizim Programında Programın Denenmesi

Programın çalışmasının izlenmesi için bütün hazırlıklar tamamlandıktan sonra program deneme aşamasına başlanabilir. Bunun için F9 tuşuna basmak ya da araç çubuğundan Play simgesine basmak yeterlidir. Şekilde simülasyonu yapılan bir tasarım görülmektedir.



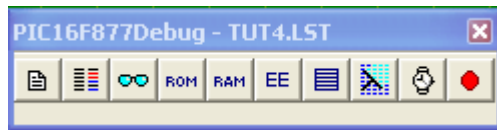
Şekil 3.10: Tasarımı tamamlanmış bir programın simüle edilmesi

Şimdi bu ekrandakileri sırayla açıklayalım.



Şekil 3.11: Start, Pause, Step, Reset ve Stop komutları

Sırasıyla çalıştır, duraklat, adım kipi, adım kipini kapat, tekrar başlat ve durdur komutları simülasyonun akışını kontrol etmektedir.



Şekil 3.12: DEBUG araç çubuğu

Programın çalışması esnasında çeşitli izleme işlemleri için kullanılır. Simgeleri sıra ile açıklayalım.

- İlk simge programın çalışma esnasında kaynak kodları üzerinde hangi komutu işlemekte olduğunu izlemek içindir. Şekildeki pencereyi açar. Sarı renkte olan satır o an çalıştırılmakta olan komuttur.

```

85      BCF INTCON, TOIF      ;Timer 0 Clear Timer
86      BSF INTCON, TOIE      ;Enabled Timer 0 Int
87      ;=====End Timer0 Configuration
88
89      BSF INTCON, GIE        ;Enable Interrupts
90
91
92      MOVLW .1               ;Note initially the
93      MOVWF PORTC
94 MAIN
95
96      MOVF TMR0,W
97      MOVWF PORTE
98      GOTO MAIN
99
00

```

Şekil 3.13: Program kodlarını simülasyon esnasında izleyebilirsiniz

- 2. simge SFR yani özel fonksiyonlara sahip kaydedicileri izlemek için kullanılmaktadır. Şekildeki pencereyi açar. Kırmızı renkte olan kaydedicilerde o an bir işlem yapıldığı anlaşılır.

SRF Name	Hex	Dec	Binary
W	45	69	01000101
INDF	C2	194	11000010
TMR0	45	69	01000101
PCL	2E	46	00101110
STATUS	00	0	00000000
FSR	57	87	01010111
PORTA	00	0	00000000
PORTB	45	69	01000101
PORTC	00	0	00000000
PORTD	00	0	00000000
PORTE	00	0	00000000
PCLATH	00	0	00000000
INTCON	A0	160	10100000
PIR1	00	0	00000000
PIR2	00	0	00000000
TMR1L	00	0	00000000
TMR1H	00	0	00000000
T1CON	27	39	00100111
TMR2	00	0	00000000
T2CON	CE	206	11001110
CCP1F	00	0	00000000

Şekil 3.14: Özel yazmaçlar: Programın simülasyonu esnasında izleyebilirsiniz

- 3. simge kendimize ait özellikle izlemek istediğimiz bir nokta devre elemanı ya da devre ile ilgili parametre var ise onu izlemek içindir.
- 4. - 5. ve 6. simgeler RAM, ROM ve EEPROM hafızasının durumunu izlemek içindir. Şekilde veri hafızasının (RAM) bir bölümü görülmektedir.

[RAM] - TUT4.LST										
	00	01	02	03	04	05	06	07	08	09
0000	C2	45	2E	00	57	00	45	00	00	00
0010	27	00	CE	00	00	00	00	E3	00	--
0020	3A	5B	F3	A2	E0	37	31	B7	77	33
0030	66	FA	A4	DE	3E	E7	C4	12	2E	BC
0040	5A	55	F6	C6	E7	C2	A7	0A	5A	F4
0050	02	A2	40	5B	DB	D0	C0	F1	03	B5
0060	8B	5B	14	DE	56	5B	35	5A	88	16
0070	63	3A	48	CF	D3	EB	68	13	63	70
0080	C2	11	2E	00	57	3F	00	00	FF	FF
0090	--	00	67	00	00	--	--	--	02	00
00A0	C0	AF	3B	07	04	D7	32	0E	65	34
00B0	14	5C	43	BE	71	D7	56	35	35	50
00C0	35	8C	34	29	14	4A	F2	82	E3	82
00D0	34	77	84	48	E7	38	ED	47	7D	04
00E0	1B	35	A7	5D	2A	EA	6A	13	4E	ED
00F0	63	3A	48	CF	D3	EB	68	13	63	70
0100	C2	45	2E	00	57	--	45	--	--	--
0110	0B	A5	D7	AA	A9	36	23	13	D1	DB
0120	D9	50	3A	21	08	D4	1D	A9	83	63

Şekil 3.15: Veri hafızası: Simülasyon esnasında izleyebilirsiniz

- 7. simge STACK hafızasını izlemek içindir.
- 8. simge belleğin herhangi bir adresindeki bilgiyi program çalışırken görüntülemek/değiştirmek için kullanılır.
- 9. simge bir kronometre çalıştırarak zamanla ilgili devrelerde analizi kolaylaştırır. Devre analizini hareketli yapar.

Görüldüğü gibi VirtualBreadboard programı ile sanal olarak hazırlanan PIC mikrodeneleyici yazılımları test edilip incelemeleri yapılabilmektedir. Bu da bize zaman ve maliyet kazancı olarak dönecektir.

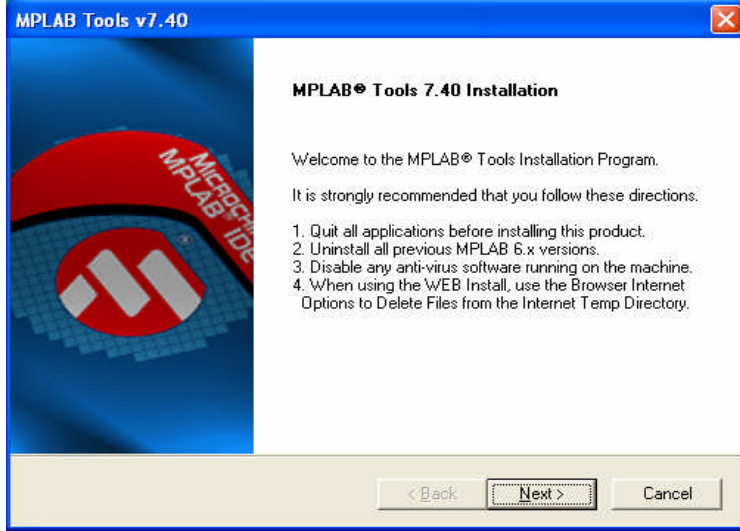
3.2. Mikrodeneleyici Program Editörünün Kurulumu

3.2.1. Derleyici Editörün Yüklenmesi

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

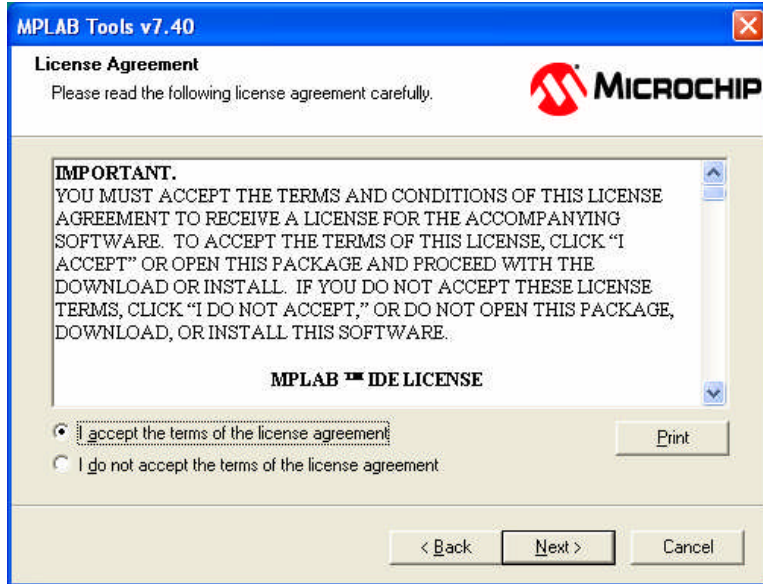
MPLAB Programını kurmak için üreticinin kendi WEB sitesinden (www.microchip.com) ücretsiz kurulum paketini bilgisayarınıza indirin.

Bu modülün yazıldığı esnada son sürümü olan 7.40'ı kurmak için indirdiğiniz sıkıştırılmış dosyayı bilgisayarınızda bir klasöre açarak Install_MPLAB_v740.exe dosyasını çalıştırın. Kurulum programı hazırlıklarını yapacaktır. Bir süre sonra şekildeki gibi bir ekranla karşılaşacaksınız.



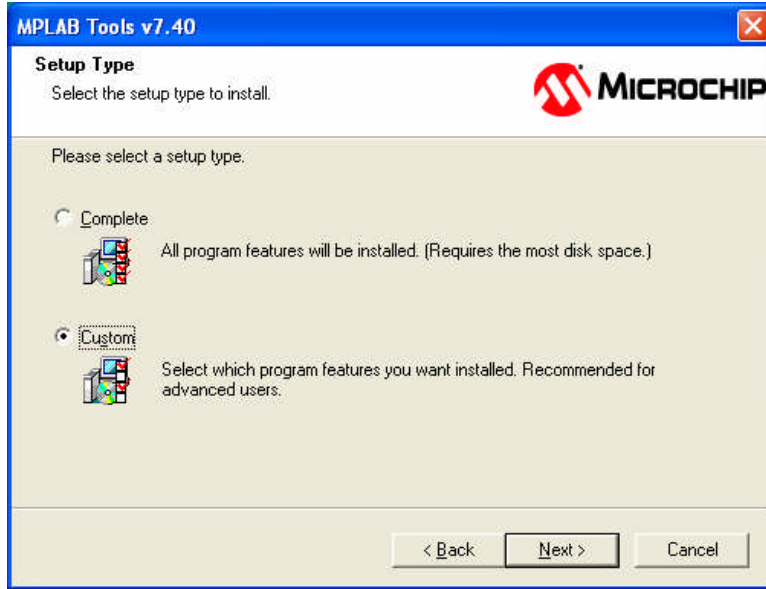
Şekil 3.16: MPLAB kurulumu. adım 1

Next komutu ile devam ediyoruz.



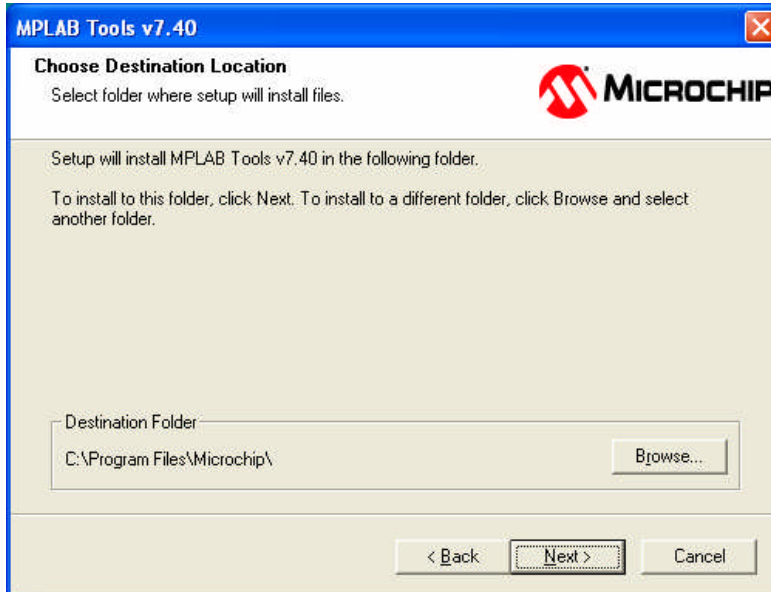
Şekil 3.17: MPLAB kurulumu. adım 2

Üretici firmanın lisans anlaşmasını kabul etmeniz gerekir aksi takdirde programı kullanamazsınız. Lisans anlaşmasının şartlarını kabul edin ve Next konutuna tıklayın.



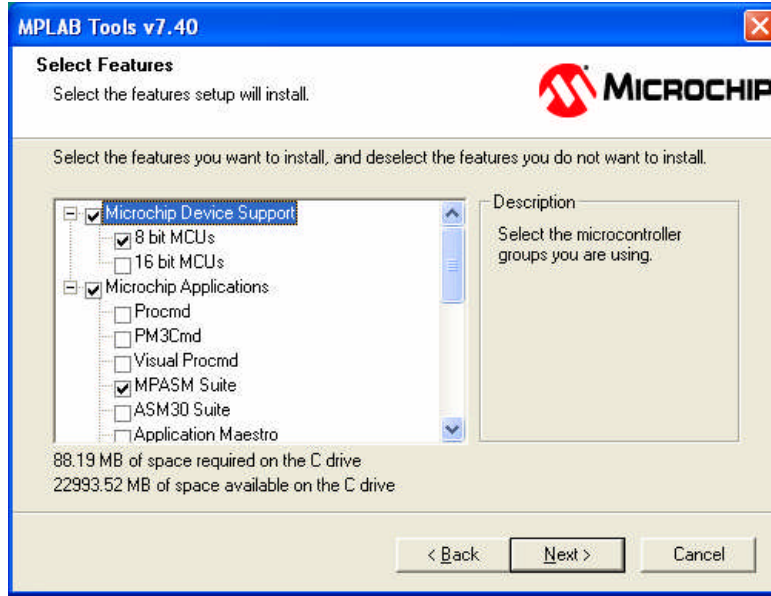
Şekil 3.18: MPLAB kurulumu. adım 3

Kurulum türünü seçerek devam ediyoruz. Eğer programın tüm bileşenlerini yüklemek isterseniz Complete seçeneğini seçin. Custom seçeneği ile sadece istediğiniz bileşenleri yükleyebilirsiniz. Next komutuna tıklıyoruz.



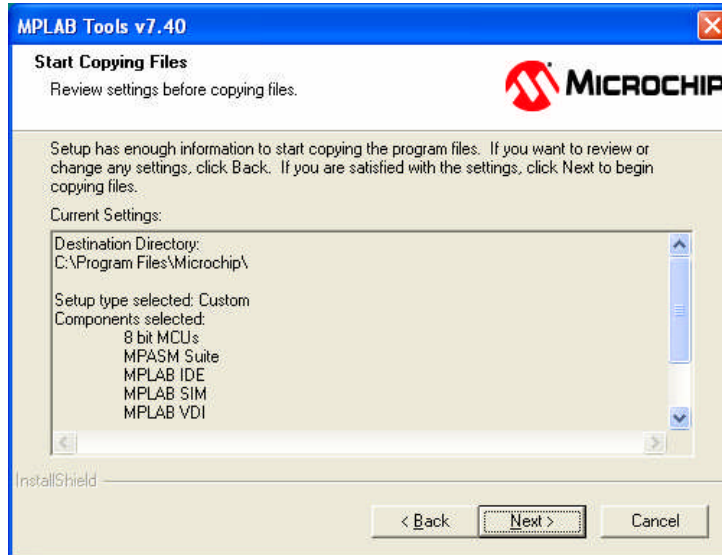
Şekil 3.19: MPLAB Kurulumu. Adım 4

Programın kurulacağı dizini seçip Next komutuna tıklıyoruz.



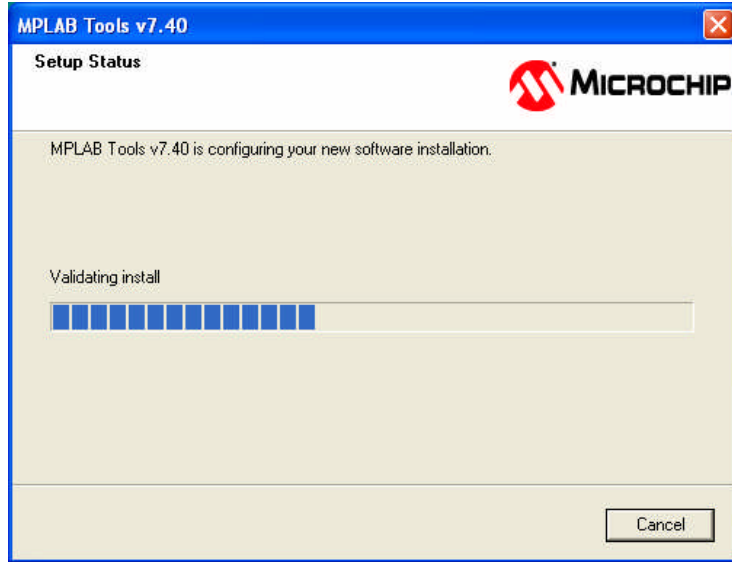
Şekil 3.20: MPLAB Kurulumu. Adım 5

Programın hangi bileşenlerini yükleyip yüklemeyeceğimizi seçiyoruz ve Next komutuna tıklıyoruz.



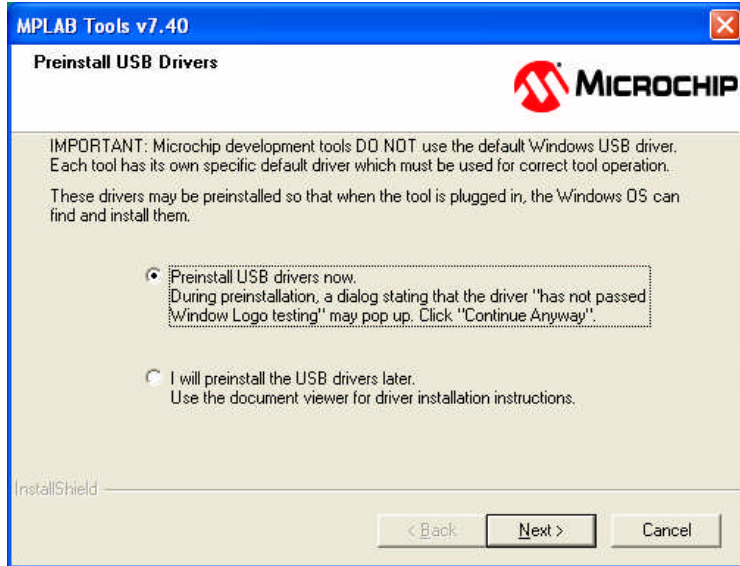
Şekil 3.21: MPLAB kurulumu. adım 6

Kurulum özeti. Next ile devam ediyoruz.



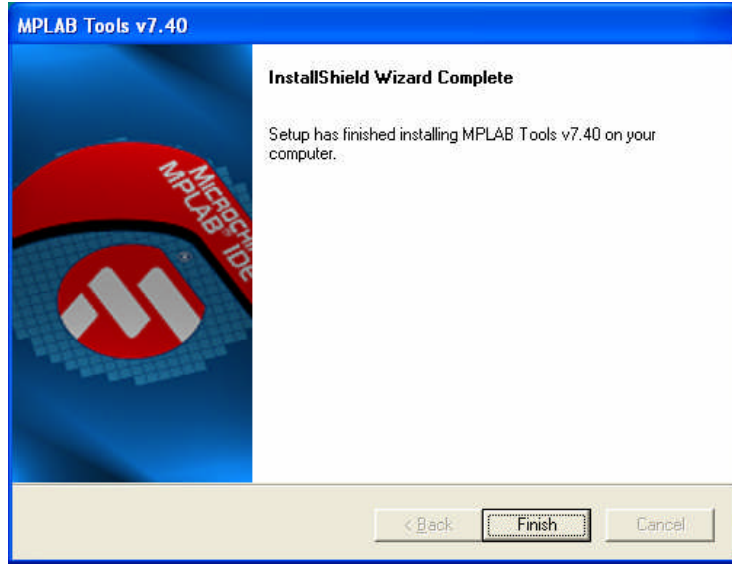
Şekil 3.22: MPLAB kurulumu. adım 7

Dosyaların kopyalanmasına başlıyor ve bekliyoruz.



Şekil 3.23: MPLAB kurulumu. adım 8

Kopyalama işlemi bitti. Eğer USB arabirim ile haberleşen bir programlama kartı, debug kartı veya deney kartı gibi bir cihaza sahip iseniz (Firmanın kendi sitesinde bu setler tanıtılmaktadır ve ücrete tabidir.) USB sürücülerini yüklemelisiniz aksi takdirde yüklemenize gerek yoktur. Biz alttaki seçeneği seçerek Next komutuna tıklıyoruz.

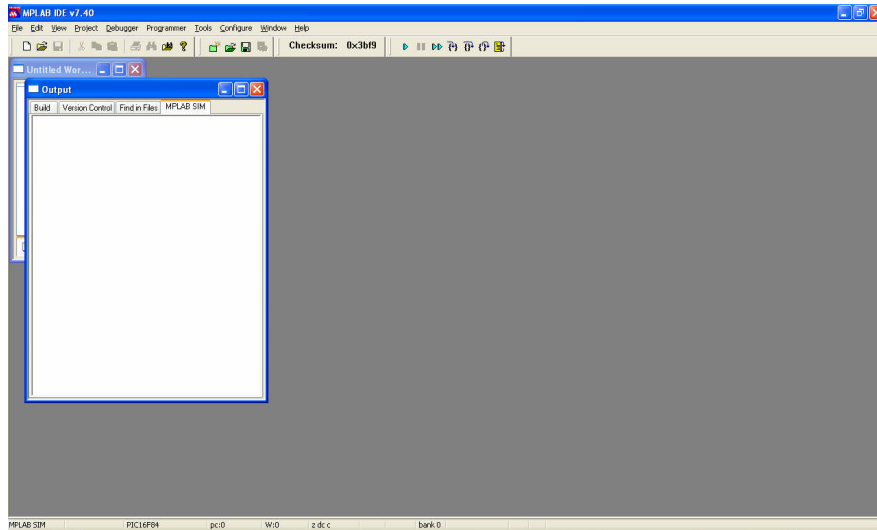


Şekil 3.24: MPLAB kurulumu. adım 9

Kurulum tamamlanmıştır. Finish komutu ile sonlandırabiliriz ve programı kullanmaya başlayabiliriz.

3.2.2. Editörün Ayarlarının Yapılması

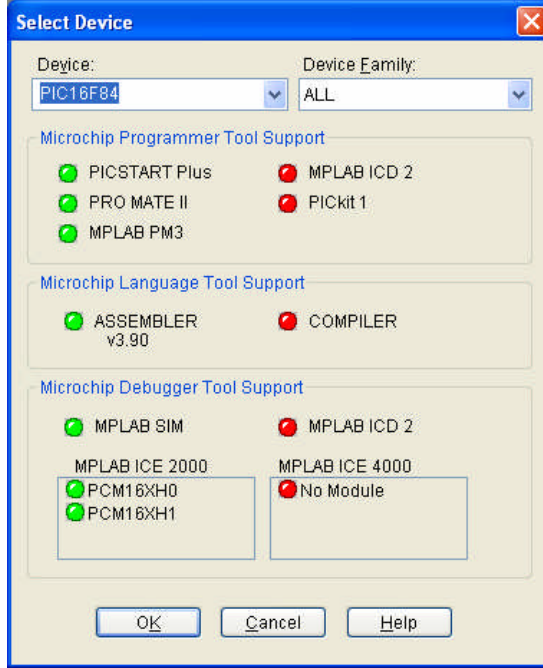
MPLAB programı başlatıldığında şekildeki gibi bir ekranla karşılaşırız. Programın genel görünümü bu şekildedir.



Şekil 3.25: MPLAB ana ekranı

MPLAB programını kullanmadan önce yazılım geliştireceğimiz denetleyiciye göre programda bazı ayarlamalar yapmamız gerekir bunlar şunlardır.

Öncelikle Debugger menüsünden Select Tool komutu ile MPLAB SIM seçilir. Bu sayede simülatör kullanılabilir. Daha sonra Configure menüsünden Select Device komutu ile kullanmak istediğimiz denetleyici seçilmelidir.



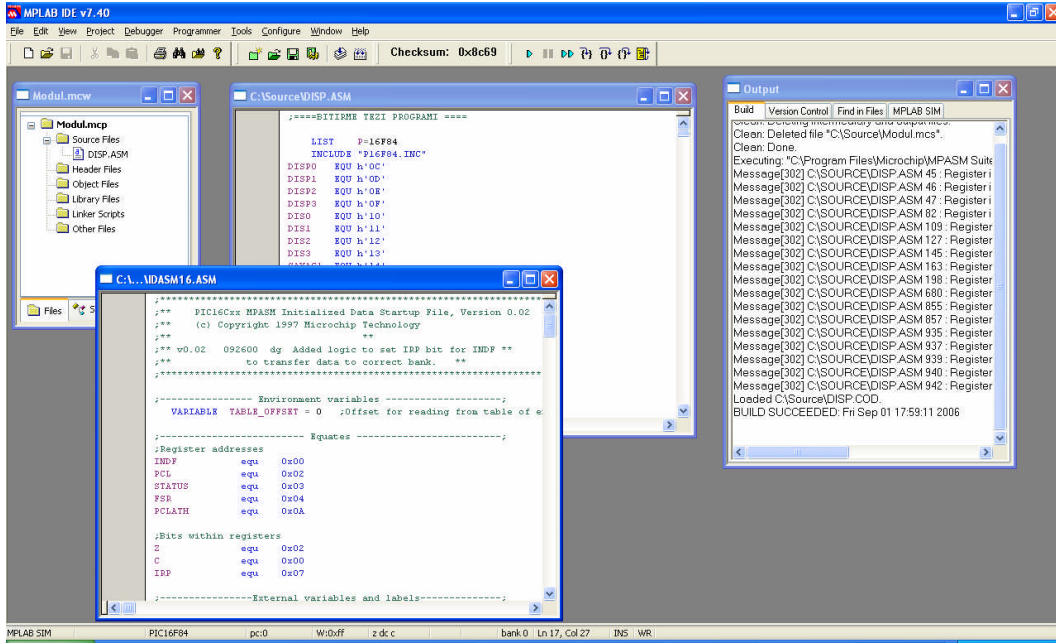
Şekil 3.26: MPLAB; denetleyici ve simülatör seçimi

Bu aşamadan sonra denetleyicinin yapılandırma bit'lerini ayarlamak için Configure menüsünden Configuration Bits komutu verilerek denetleyicinin yapılandırma bitleri ayarlanır.

3.2.3. 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.

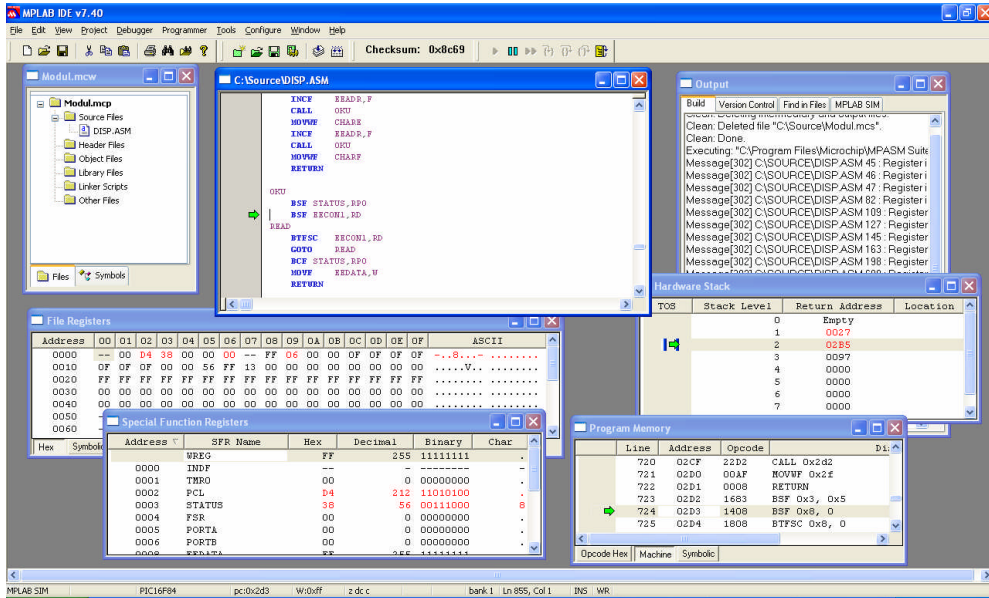
Editing yani düzenleme ve program kodlarını yazma safhasında olan bir projenin görünümü şekildeki gibidir.



Şekil 3.27: MPLAB; editör özellikleri

Kodlar renklendirilerek hata oranı azaltılmıştır.

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

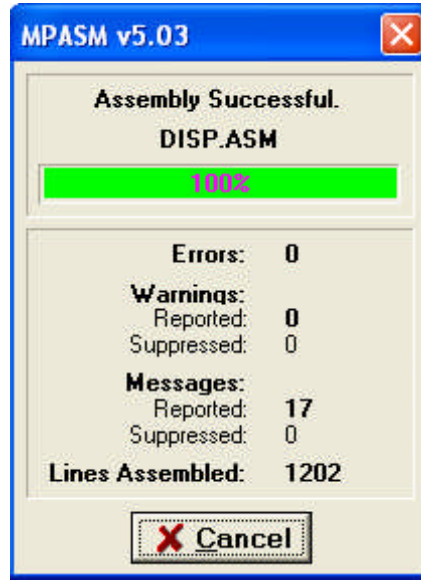


Şekil 3.28: MPLAB; simülasyon durumundaki görünüm

Bu pencereler sırası geldikçe anlatılacaktır.

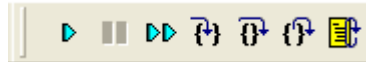
3.2.4. Denetleyiciye Yüklenecek Programın Derlenmesi ve Denenmesi

MPLAB programında hazırlanmış bir projeyi denetleyiciye yükleyebilmek için derlenmiş Binary dosyalara ihtiyaç vardır. Bunu MPASM aracılığı ile dışarıda haricen yapabileceğimiz gibi aynı zamanda MPLAB içerisinde Project menüsünden Quickbuild, Make, Build All gibi komutlarla derleme işlemini gerçekleştirebiliriz. Derleme işlemi tamamlanınca şekildeki gibi bir mesajla karşılaşılır.



Şekil 3.28: 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.

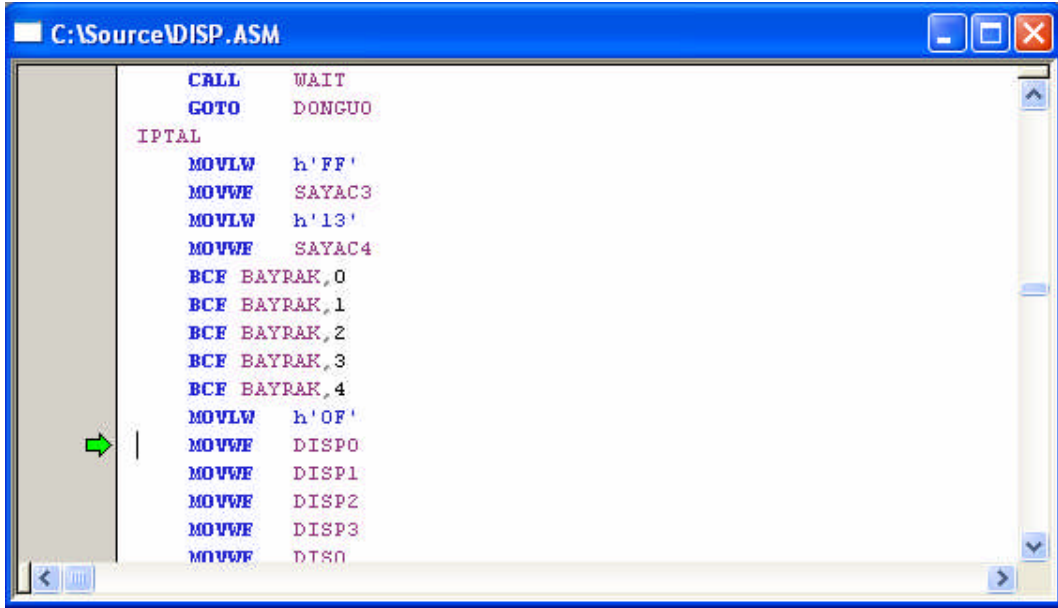


Şekil 3.29: MPLAB; simülasyon komutları

Burada 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.

Simülasyon esnasında programı daha iyi analiz etmek için kullanılan pencereleri birer birer görelim.

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



```
C:\Source\DISP.ASM

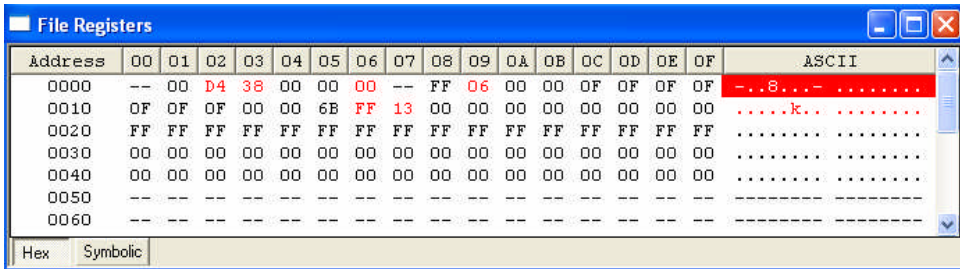
CALL    WAIT
GOTO    DONGUO

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   DISPO
MOVWF   DISP1
MOVWF   DISP2
MOVWF   DISP3
MOVWF   DISO
```

Şekil 3.30: MPLAB; program kodları

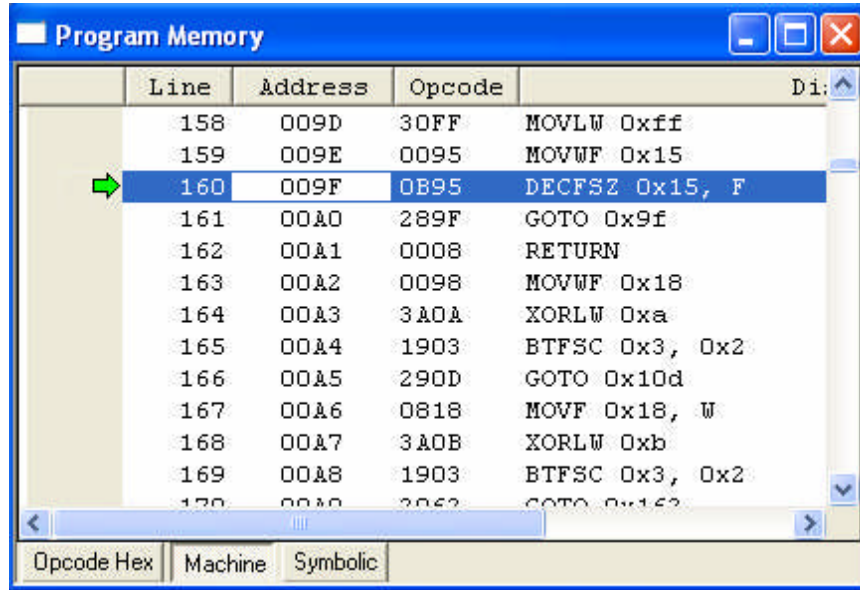
- **File Registers:** Bununla veri hafızasının durumu çalışma zamanında izlenebilir. Kırmızı renkli hücreler en son işlem gören hücrelerdir.



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	0F	00	00	6B	FF	13	00	00	00	00	00	00	00	...k...
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 3.31: MPLAB; File Registers

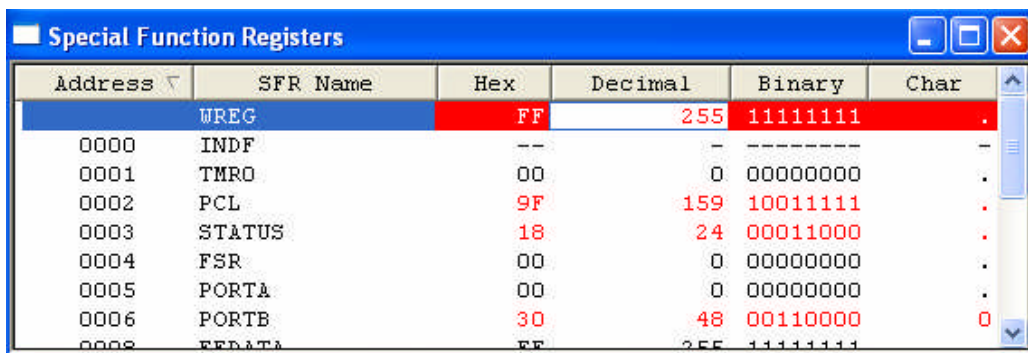
- **Program Memory:** Bununla da program hafızasının durumu çalışma zamanında izlenmektedir. Yeşil işaret hangi komutun icra edildiğini göstermektedir.



Line	Address	Opcode	Disassembly
158	009D	30FF	MOVLW 0xff
159	009E	0095	MOVWF 0x15
160	009F	0B95	DECFSZ 0x15, F
161	00A0	289F	GOTO 0x9f
162	00A1	0008	RETURN
163	00A2	0098	MOVWF 0x18
164	00A3	3A0A	XORLW 0xa
165	00A4	1903	BTFSC 0x3, 0x2
166	00A5	290D	GOTO 0x10d
167	00A6	0818	MOVF 0x18, W
168	00A7	3A0B	XORLW 0xb
169	00A8	1903	BTFSC 0x3, 0x2
170	00A9	289F	GOTO 0x163

Şekil 3.32: MPLAB; program hafızası

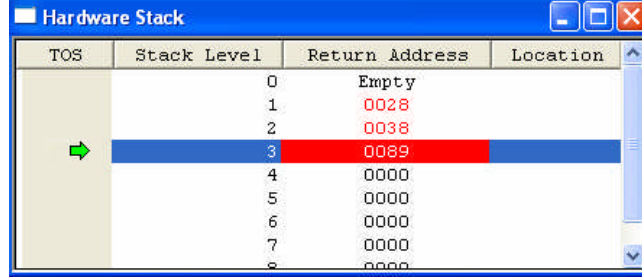
- **Special Function Registers:** Özel amaçlı kaydedicileri izlemek içindir. Kırmızı renkte olanlar son işlem gören kaydedicilerdir.



Address	SFR Name	Hex	Decimal	Binary	Char
0000	WREG	FF	255	11111111	.
0001	INDF	--	--	-----	-
0002	TMRO	00	0	00000000	.
0003	PCL	9F	159	10011111	.
0004	STATUS	18	24	00011000	.
0005	FSR	00	0	00000000	.
0006	PORTA	00	0	00000000	.
0007	PORTB	30	48	00110000	0
0008	PORTC	FF	255	11111111	.

Şekil 3.33: MPLAB; özel fonksiyon registerleri

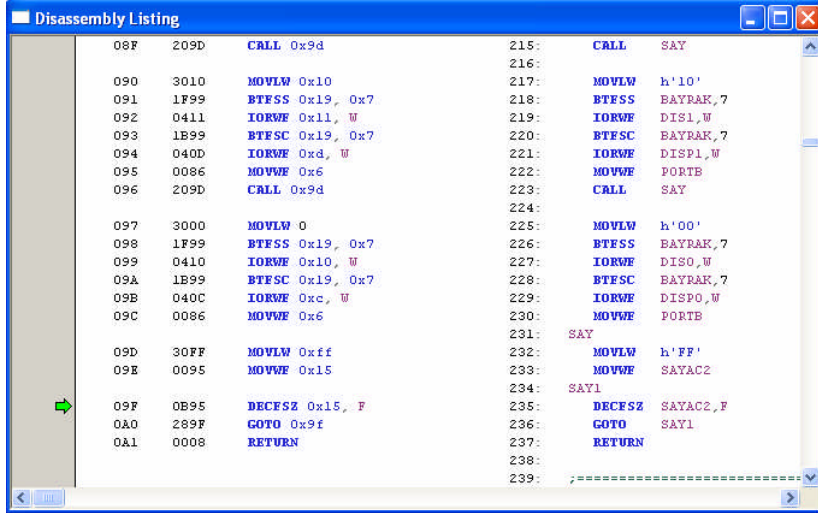
- **Hardware Stack:** Stack hafızasını gösterir.



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 3.34: MPLAB; yığın yazmaçları

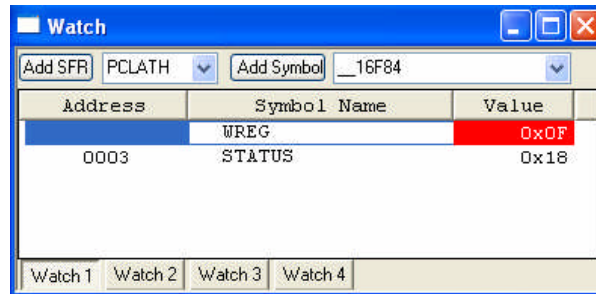
- **Diassembly Listing:** Derlenmiş dosyadan tekrar elde edilen komut dosyasını gösterir. Aktif komut yeşil gösterge ile işaretlenmiştir.



Address	Instruction	Value
08F	CALL 0x9d	215: CALL SAY
090	MOVLW 0x10	216: MOVLW h'10'
091	BTFSS 0x19, 0x7	217: BTFSS BAYRAK,7
092	IORWF 0x11, W	218: IORWF DIS1,W
093	BTFSC 0x19, 0x7	219: BTFSC BAYRAK,7
094	IORWF 0xd, W	220: IORWF DIS1,W
095	MOVWF 0x6	221: MOVWF PORTB
096	CALL 0x9d	222: CALL SAY
097	MOVLW 0	223: MOVLW h'00'
098	BTFSS 0x19, 0x7	224: BTFSS BAYRAK,7
099	IORWF 0x10, W	225: IORWF DIS0,W
09A	BTFSC 0x19, 0x7	226: BTFSC BAYRAK,7
09B	IORWF 0xc, W	227: IORWF DIS0,W
09C	MOVWF 0x6	228: MOVWF PORTB
09D	MOVLW 0xff	229: MOVLW h'FF'
09E	MOVWF 0x15	230: MOVWF SAYAC2
09F	DECFSZ 0x15, F	231: SAY1
0A0	GOTO 0x9f	232: MOVLW h'FF'
0A1	RETURN	233: MOVWF SAYAC2
		234: SAY1
		235: DECFSZ SAYAC2,F
		236: GOTO SAY1
		237: RETURN
		238: ;=====
		239: ;=====

Şekil 3.35: MPLAB; Diassembly listeleme

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

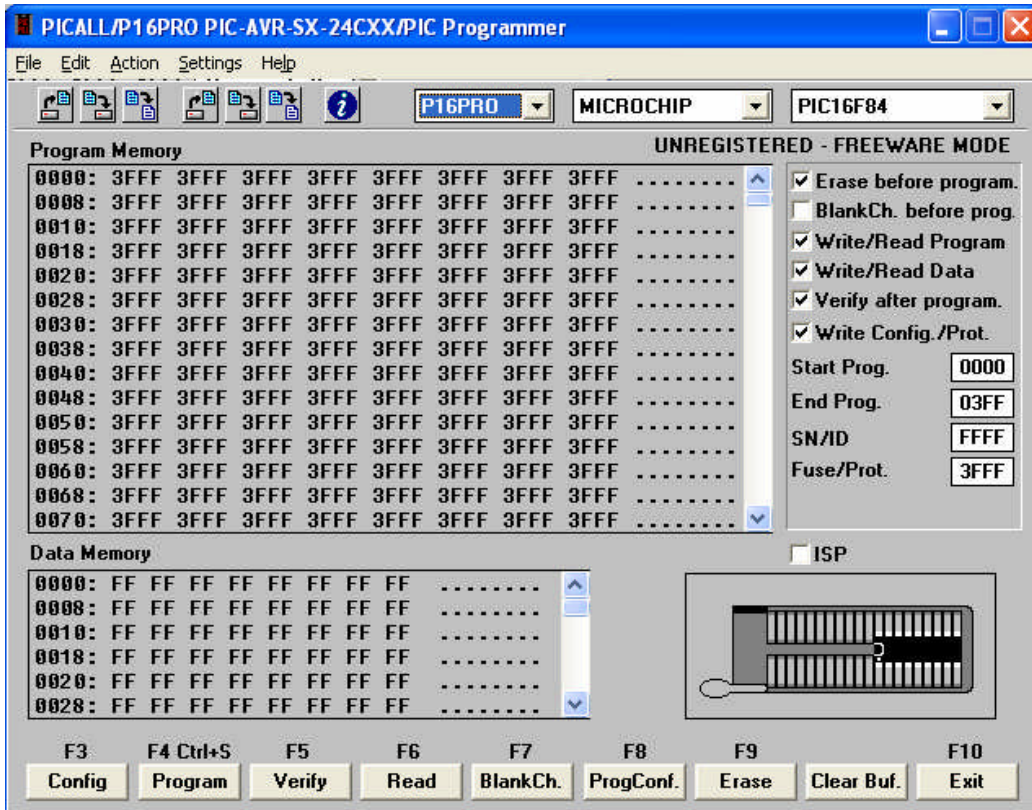


Address	Symbol Name	Value
0003	STATUS	0x18

Şekil 3.36: MPLAB; isteğe bağlı register izleme

3.2.5. Denetleyiciye Yüklenecek Programın Derlenmesi ve Denenmesi

Hazırlanan kodların derleme işi bittikten sonra bunları PIC mikrodenetleyicisinin program hafızasına aktarmak gerekmektedir. Bunun için daha önceki konularda anlatılan programlayıcı devrelere ihtiyaç vardır. Her ne kadar MPLAB içerisinde bazı programlayıcı devrelere destek olsa dahi bunların çoğu ücretli ve temini zor olduğu için kendi kullanmak istediğimiz programlayıcıya ait özgün programları kullanmak daha akılcı olacaktır. Önceki bölümlerde anlatılan P16PRO programlayıcısı için hazırlanmış PICALL yazılımını inceleyelim. Programın görünümü aşağıdaki gibidir.



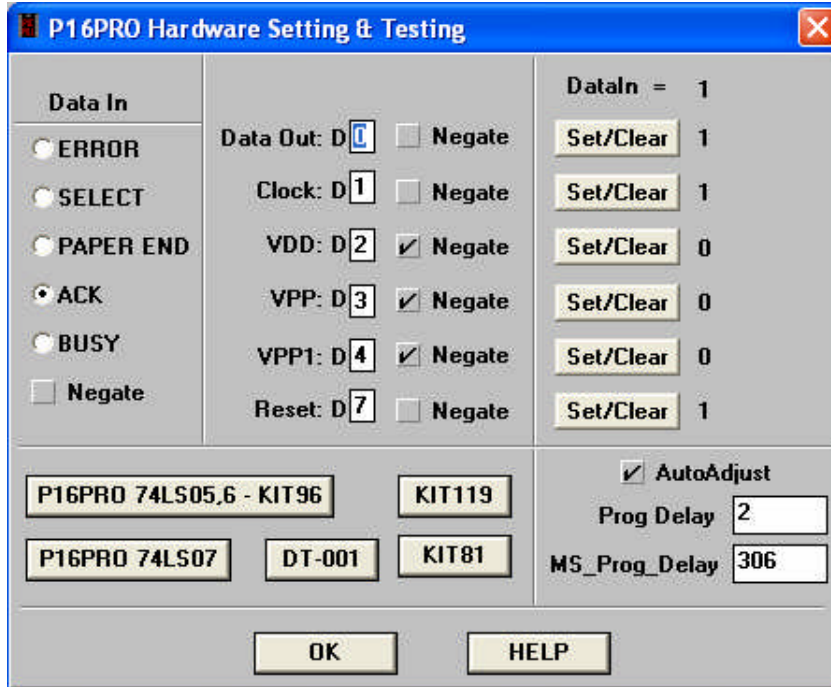
Şekil 3.37: PICALLW; genel görünüm

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. Daha önceki konularda anlatılan P16PRO için KIT96 seçilmelidir. Ayrıca kod koruma osilatör türü vb. ayarlar için Edit menüsünden Configuration komutu çalıştırılır. Şekildeki gibi bir ekran gelecektir burada istenilen özellikler ayarlanarak programlama işlemine hazır hale gelinir.



Şekil 3.38: PICALLW; sigorta, kod koruma ve osilatör seçimi



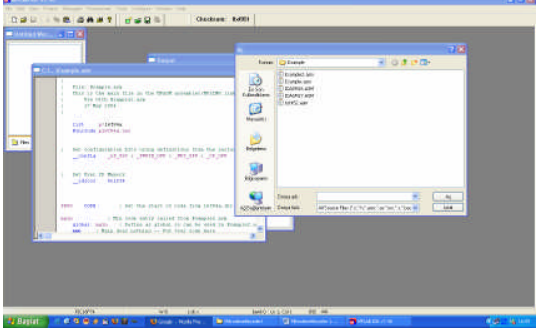
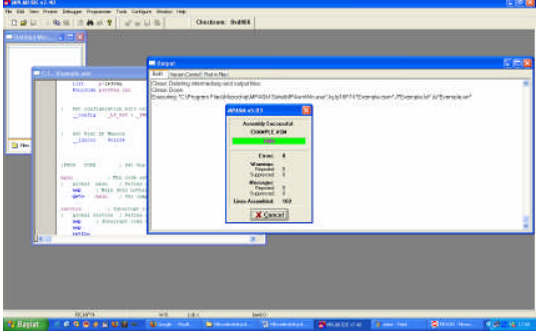
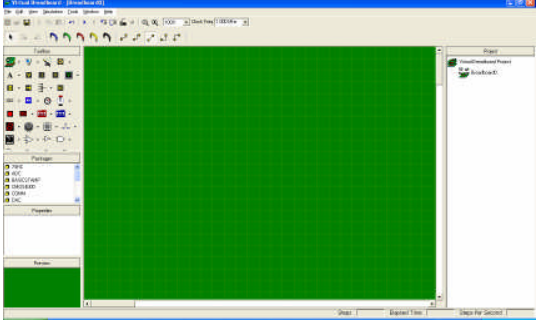
Şekil 3.39: PICALLW; programlayıcı donanımı özelliklerinin seçimi

F4 tuşuna basarak ya da Program komutunu çalıştırarak binary kodların denetleyiciye aktarımı sağlanmış olur.

3.2.6. Denetleyiciye Yüklenecek Programın Derlenmesi ve Denenmesi

Mikrodenetleyicinin programlanması ve simülasyonu tamamlandıktan 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

İşlem Basamakları	Öneriler
<p>➤ MPASM programını 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.</p> <p>➤ Derleme sonundaki hata ve uyarı mesajlarını izleyerek hatalar bulunduğu takdirde düzeltmeye çalışınız.</p>
<p>➤ Virtual BreadBord programını açınız.</p> 	<p>➤ Açarken Başlat – Programlar – VirtualBreadBoard – VirtualBreadBoard.exe yolunu izleyiniz</p>

➤ Packages kutusundan PIC 16F84 Mikrodenetleyicisini form üzerine sürükleyiniz.	➤ Denetleyiciye simulasyon için güç ve osilatör devresi hazırlamaya gerek yoktur.
➤ Properties penceresinden osilatör frekansını ve türünü Clockspeed ve Oscillator seçeneklerinden XT ve 4 MHz olarak ayarlayınız.	➤ Daha yavaş çalışmasını istediğimiz devrelerde daha düşük osilatör frekansları kullanılabilir.

ÖLÇME VE DEĞERLENDİRME

A. OBJEKTİF TESTLER (ÖLÇME SORULARI)

Aşağıdaki soruların cevaplarını doğru ve yanlış olarak değerlendiriniz.

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. (.....)
4. Virtual BreadBoard ile programlanan devrenin çalışma esnasındaki tepkisi görülerek hata bulma işlemi hızlandırılır. (.....)
5. Simulasyonun başlaması için Simulation menüsünden Run komutu çalıştırılır. (.....)
6. Simülasyon adım adım izlenecekse Step komutu seçilir. (.....)

Aşağıdaki cümlelerde bulunan boşlukları en uygun şekilde doldurunuz.

7. PICALLW uygulamasında Osilatör ve kod koruma seçenekleri F3 tuşuna basılarak , _____ komutu çalıştırılır ve ayarlanır.
8. PIC denetleyicisi ile hazırlanmış bir program kodu _____ ' programı ile simüle edilebilir.
9. _____ programı MPLAB paketinin içerisinde derleyici görevi yapan bölümüdür.
10. _____ programı MPLAB paketinin içinde simülatör görevi yapan bölümüdür.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarı ile karşılaştırınız. Doğru cevap sayınızı belirleyerek kendinizi değerlendiriniz. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt yaşadığınız sorularla ilgili konuları faaliyete geri dönerek tekrar inceleyiniz

Tüm sorulara doğru cevap verdiyseniz modül değerlendirmeye geçiniz.

MODÜL DEĞERLENDİRME

PERFORMANS TESTİ (YETERLİK ÖLÇME)

Modül ile kazandığınız yeterliği aşağıdaki ölçütlere göre değerlendiriniz.

KONTROL LİSTESİ

Değerlendirme Ölçütleri	Evet	Hayır
1. Mikrodenetleyici Seçimi Yapmak ve Mikrodenetleyicileri Tanımak		
A) PIC 16F84 mikrodenetleyicisini Board üzerine doğru bir şekilde bağlayabildiniz mi?		
B) Besleme devresini ve voltajını doğru verdiniz mi?		
C) Osilatör seçimini ve montajını doğru yaptınız mı?		
2. Mikrodenetleyici Programlama Kartı Yapmak		
A) P16PRO mikrodenetleyicisinin baskılı devresini doğru bir şekilde hazırladınız mı?		
B) Paralel bağlantı kablosunu doğru olarak bağladınız mı?		
C) Programlayıcı kartının asit ile baskılı devre hazırlama işlemini doğru bir şekilde yaptınız mı?		
D) Programlayıcı kartının lehimleme işlemini doğru bir şekilde yaptınız mı?		
3. Hata Mesajlarını Çözebilmek		
A) MPLAB programında yazılmış kodları doğru şekilde yaptınız mı?		
B) MPSIM ile yazılan bir kodu doğru olarak denediniz mi?		
C) PICALLW programı ile derlenmiş yazılımı P16PRO programlayıcısı ile doğru olarak veri aktarımı yaptınız mı?		
D) Hazırlanan devreleri VirtualBradboard ile denediniz mi?		

DEĞERLENDİRME

Yaptığınız değerlendirme sonucunda eksikleriniz varsa öğrenme faaliyetlerini tekrarlayınız.

Modülü tamamladınız, tebrik ederiz. Öğretmeniniz size çeşitli ölçme araçları uygulayacaktır. Öğretmeninizle iletişime geçiniz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1CEVAP ANAHTARI

1	D
2	D
3	Y
4	Y
5	D
6	D
7	$(101)_{10}, (65)_{16}$
8	$(1111111000)_2,$ $(1016)_{10}$
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 CEVAP ANAHTARI

1	Y
2	D
3	Y
4	Y
5	D
6	Y
7	D

ÖĞRENME FAALİYETİ-3 CEVAP ANAHTARI

1	D
2	D
3	Y
4	D
5	D
6	D
7	Config
8	MPLAB, ViertualBradboard
9	MPASM
10	MPSIM

Cevaplarımızı cevap anahtarları ile karşılaştırarak kendinizi değerlendiriniz.

ÖNERİLEN KAYNAKLAR

- <http://www.microchip.com>
- <http://www.antrag.org.tr>
- <http://www.hitachi.com>
- <http://www.elektronikhobi.net>
- <http://www.picproje.org>
- <http://www.cpu-world.com>
- <http://www.cupages.com>
- <http://www.picallw.com>

KAYNAKÇA

- KATAR İdris, Mikrodenetleyiciler Ders Notları
- <http://www.microchip.com>
- <http://www.mikroelektronika.co.yu/english/product/books/PICbook/picbook.htm>
- <http://www.antrag.org.tr>
- <http://www.hitachi.com>
- <http://www.elektronikhobi.net>
- <http://www.picallw.com>
- <http://home.iae.nl/users/pouweha/lcd/lcd2.shtml>
- <http://www.phanderson.com/PIC>
- http://www.winpicrog.co.uk/pic_tutorial3.htm