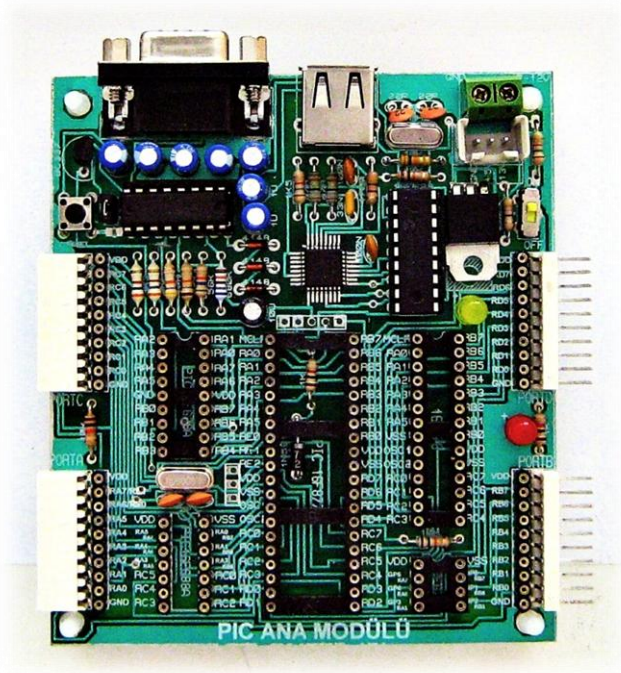


DENEY FÖYÜ

EEM406 MİKRODENETLEYİCİLER
LABORATUVARI DENEY FÖYÜ

YRD.DOÇ.DR.ALİ YILDIZ
ARŞ.GÖR.MUSTAFA B. BIÇER

©2011



[EEM 406 MİKRODENETLEYİCİLER LAB]

MERSİN ÜNİVERSİTESİ, MÜHENDİSLİK FAKÜLTESİ, ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

EEM-406 Mikrodenetleyiciler Laboratuvar Kuralları

1. Laboratuvara gelmeden önce öğrencilerin, o gün yapılacak deneye ait deney f6y6n6 dikkatlice okumaları, deney ile ilgili teorik bilgileri çeşitli kaynaklardan öğrenmiş olmaları gerekir. Deneye başlamadan önce ve deney süresinde her öğrenciye, o gün yapacağı deney ile ilgili sorular sorulur. Bu sorulara verilen cevaplar ve deneyin yürütölüşü sırasında gösterilen başarı, birlikte deęerlendirilerek, öğrenciye her yaptığı deney için not verilir.
2. Ge gelen öğrenciler deneye alınmayacaktır.
3. Laboratuvar ortamı deney sonunda, deneye başlamadan önceki haline getirilmelidir.
4. Laboratuvarda özensiz tutum ve davranışlardan kaçınılmalıdır. Laboratuvar içerisindeki tutum ve davranışlar, laboratuvar notuna etki etmektedir.
5. Laboratuvar not deęerlendirmesi aşığıdaki gibidir:
 - a. **Laboratuvar** deney not ortalaması, quiz ve deney içi performans ile belirlenecektir.
 - b. **Laboratuvar** genel not ortalaması aşığıdaki yüzdeler ile hesaplanacaktır.

Deney Notu	%50
Uygulamalı Final Sınavı	%50

6. Devam zorunludur. En fazla **1** deneye girmeyen öğrenci, devamsızlıktan kalacaktır. Deneylerin bittięi haftadan sonraki hafta, telafi ve mazeret haftasıdır. Telafi hakkını sadece deneylere gerekçeli olarak giremeyenler kullanabilir.

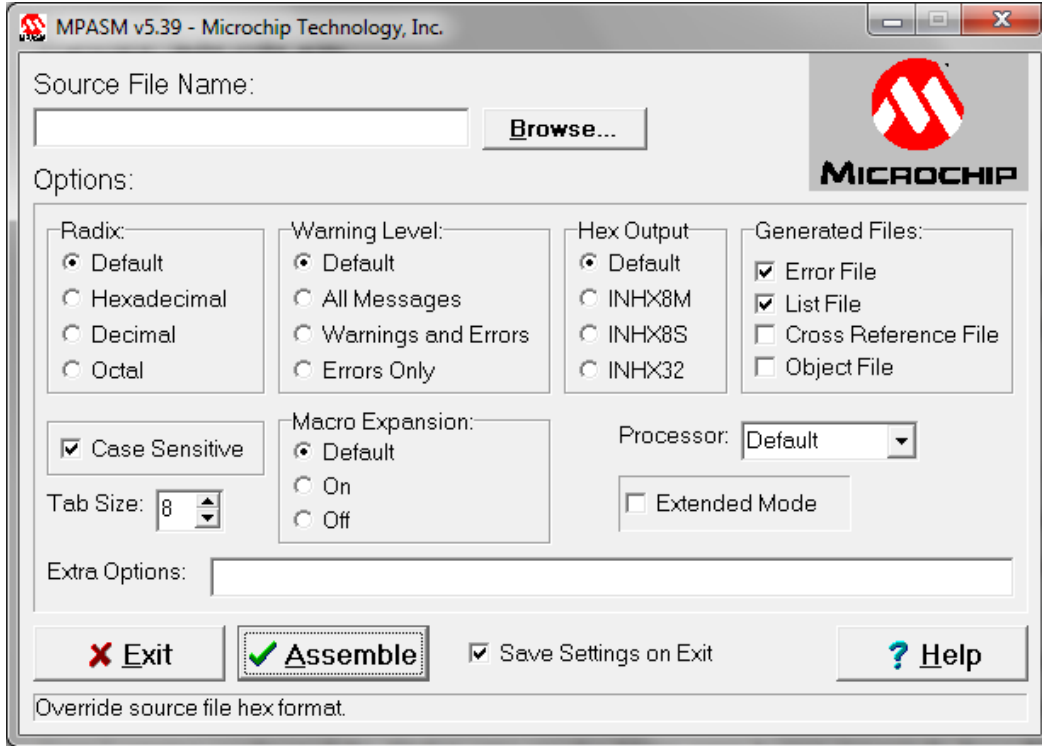
Laboratuvarda Alınması Gereken Güvenlik Önlemleri

- Laboratuvar sorumlusu gözetiminde olmadan; laboratuvarda çalışma yapmayınız.
- Laboratuvar ortamında kullanacağınız deney seti üzerindeki elemanlara, ihtiyaç yoksa dokunmayınız, tahrip olmasına sebep olacak davranışlardan kaçınınız.
- Deney seti üzerindeki elemanların bacaklarını kısa devre yapmaya neden olacak davranışlardan kaçınınız.
- Deney sırasında cep telefonunuzu kapatınız.

GENEL BİLGİLER

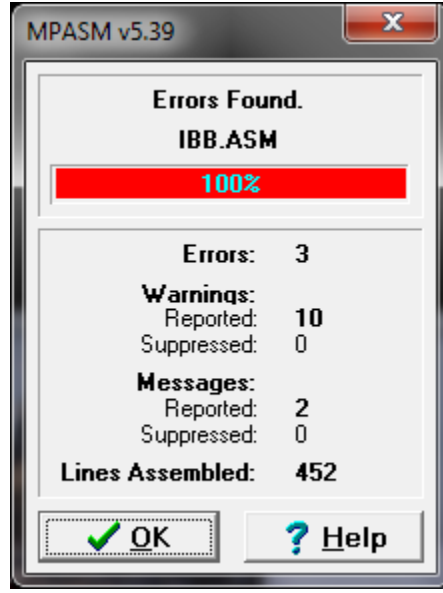
Programın Bilgisayarda Yazılıp Derlenmesi

Program, edit, wordpad veya notepad benzeri bir yazılım ile yazıldıktan sonra, C sürücüsünde Türkçe karakter ve boşluk içermeyen bir klasöre (örneğin *deneyler*), *asm* uzantılı olacak şekilde kaydedilmelidir (örneğin *deney1.asm*). Masaüstünde yer alan MPASMWIN kısayoluna çift tıklanarak, yazılan asm kodunun derleneceği kullanıcı arayüzü açılmalıdır.



Şekil 1 - MPASMWIN Derleyici Ekranı

Yazılan asm dosyasını açmak için Browse... butonuna tıklanmalı ve açılan pencereden, kaynak kodun olduğu klasöre ulaşılarak, derlenmek istenen dosyaya çift tıklanmalıdır. MPASMWIN ekranındaki ayarların değiştirilmesine gerek olmadığından dolayı kodu derlemek için Assemble butonuna tıklanmalıdır. Dosya derlenerek, PIC'e yüklenmesi gereken hex uzantılı dosya ile oluşturulacak diğer dosyalar, derleme sonucunda, derlemenin hatasız olduğunu gösteren ekran görüldüğünde, kaynak kodun olduğu klasöre oluşturulacaktır.



Şekil 2 - Derleme Hatası

Şekildeki ekranın görülmesi, derlemenin başarısız olduğunu göstermektedir ve hata ile ilgili bilgiler, kodun bulunduğu klasörde err uzantılı dosyada yer almaktadır. Bu err uzantılı dosya, herhangi bir metin editörü ile açılarak hatanın nerede olduğu görülebilmektedir.

```

C:\Users\mberkanbicer\Documents\piclab\demo\incbybutton\ibb.ERR - Notepad++
Dosya Düzen Arama Görünüm Kodlama Dil Ayarlar Makro Çalıştır TextFX Ekleniler Pencere 2
ibb.asm | ibb.ERR
1 Warning[205] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 2 : Found directive in column 1. (list)
2 Warning[205] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 3 : Found directive in column 1. (include)
3 Warning[205] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 6 : Found directive in column 1. (__CONFIG)
4 Warning[207] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 10 : Found label after column 1. (portveri)
5 Warning[207] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 11 : Found label after column 1. (d1)
6 Error[122] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 11 : Illegal opcode (EQaU)
7 Warning[207] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 12 : Found label after column 1. (d2)
8 Message[302] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 18 : Register in operand not in bank 0. Ensure
9 Message[302] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 19 : Register in operand not in bank 0. Ensure
10 Warning[203] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 38 : Found opcode in column 1. (goto)
11 Warning[203] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 49 : Found opcode in column 1. (return)
12 Warning[203] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 58 : Found opcode in column 1. (return)
13 Error[113] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 65 : Symbol not previously defined (d1)
14 Error[113] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 69 : Symbol not previously defined (d1)
15 Warning[205] C:\USERS\MBERKANBICER\DOCUMENTS\PICLAB\DEMOS\INCBYBUTTON\IBB.ASM 82 : Found directive in column 1. (END)
16

```

Şekil 3 - ERR Uzantılı Hata Dosyasının İçeriği

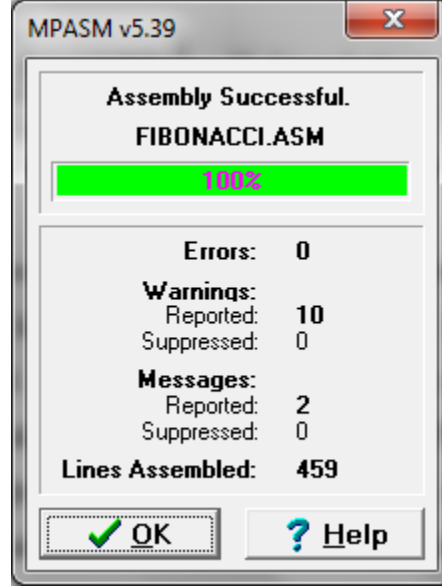
Şekilde, örnek olarak hatalı bir şekilde yazılan ve derlenen kod ile ilgili hata dosyasının içeriği görülmektedir. Kırmızı kutu içerisinde gösterilen satırlar, hataları göstermektedir.

Hata dosyasının yapısı şu şekildedir:

Mesaj Türü[Hata Numarası]	Dosya Adı	Satır Numarası	:	Hata Açıklaması
---------------------------	-----------	----------------	---	-----------------

Yapıya bakıldığında 11'nci satırda Illegal Opcode (EQaU) hatası yer almaktadır. Bu hata, parantez içerisinde yazılan kodun hatalı olduğunu göstermektedir. Bu kodun doğrusu EQU olmalıdır. Kod düzeltildiğinde hatasız bir derleme yapılacaktır.

65 ve 69.ncu satırlarda görülen hata, 11.nci satırdaki hatadan kaynaklanmaktadır.

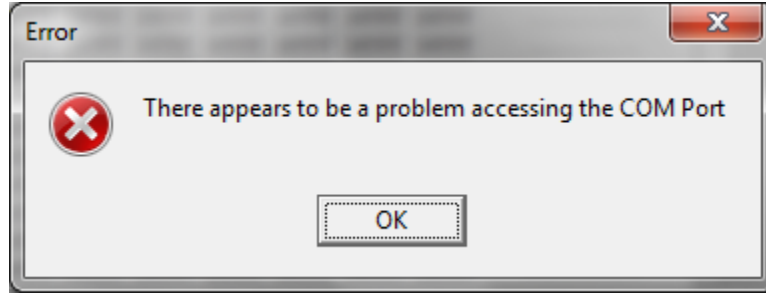


Şekil 4 - Hatasız Derleme İşlemi

Şekildeki gibi bir ekran, derlemenin hatasız olarak yapıldığını göstermektedir.

Programın PIC'e Aktarılması

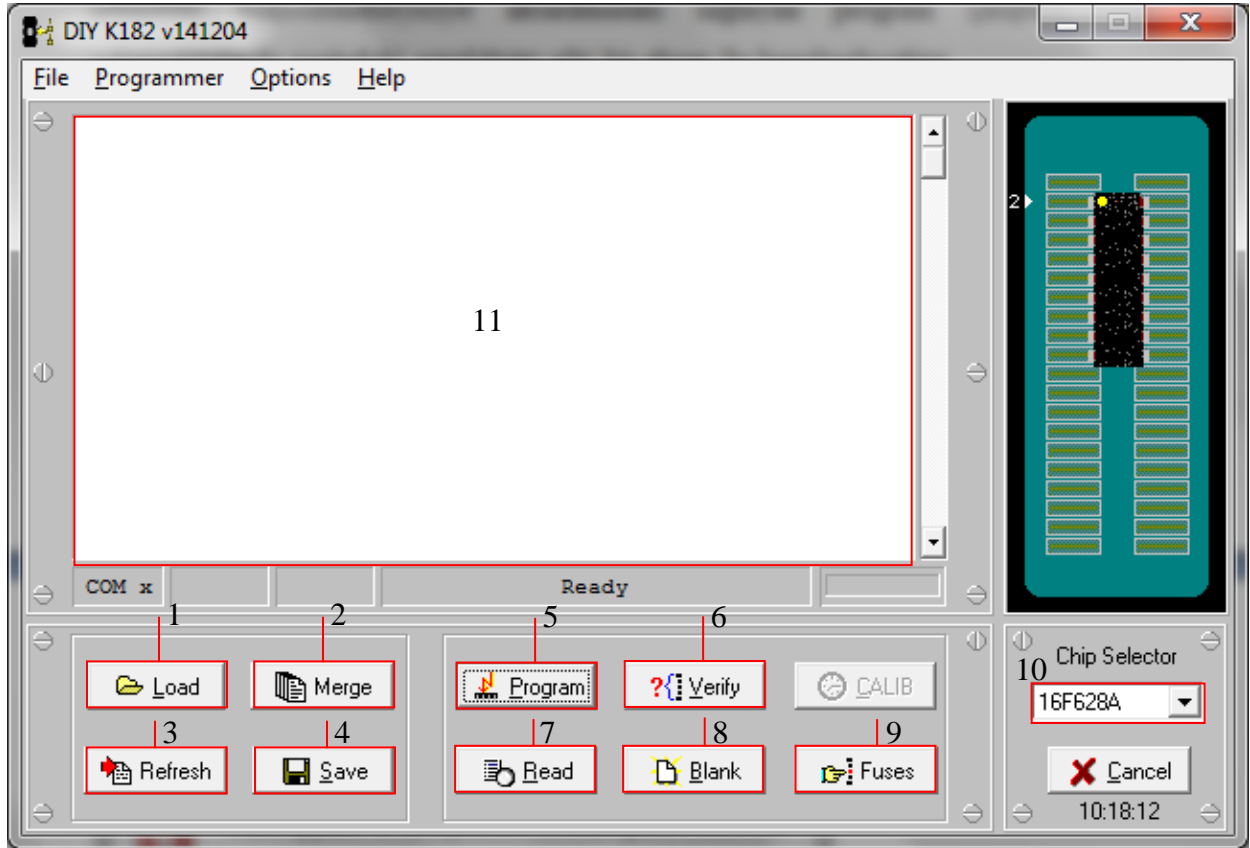
Derlenen programın deney setine atılması için öncelikle deney seti ile bilgisayar arasında bağlantının yapılmış olması gerekmektedir. Bunun için bilgisayarın herhangi bir USB portu ile deney setini USB portu arasında arasında USB kablo bağlantısı yapılmalıdır. Bağlantı yapıldıktan sonra masaüstünde yer alan MicroPro kısayoluna çift tıklanarak, derlenen programın deney setindeki mikrodenetleyiciye aktarılmasını sağlayan program çalıştırılmalıdır. Deney seti bağlantısı yapılmadan program çalıştırılırsa aşağıdaki gibi bir hata ile karşılaşılacaktır.



Şekil 5 - Deney Seti Bağlantısı Olmadığını Bildiren Hata Mesajı

(Not: MicroPro programı, PIC'i seri port kullanarak programlama işi yapabilmektedir. Laboratuvarında kullanılacak deney seti ise USB portu kullanılarak programlanabilmektedir. Bundan dolayı deney seti için bilgisayara COM \leftrightarrow USB dönüştürücü sürücüsü kurulmalıdır.)

Deney seti bağlantısı sağlandıktan sonra program açılırsa herhangi bir hata oluşmayacaktır. Program çalıştırıldığında aşağıdaki görüldüğü gibi bir ekran ile karşılaşılacaktır.



Şekil 6 - MicroPro Programı Ekranı

Şekilde numaralandırılmış butonlar ve kısımlar ile ilgili bilgiler tabloda verilmiştir.

#	İsim	Görev
1	Load	PIC'e aktarılacak olan HEX dosyasını seçme işlevi için kullanılır.
2	Merge	Birden fazla dosyayı birleştirmek için kullanılır. <i>Deney süresinde kullanılmayacaktır.</i>
3	Refresh	Hafızaya alınan kod, tekrar derlenirse bu butona tıklayarak hafızadaki kod yenilenmelidir.
4	Save	Hafızaya alınan ve programda değiştirilen kısımların kaydedilmesi için kullanılır.
5	Program	HEX dosyasının PIC'e aktarılması için kullanılır.
6	Verify	PIC'e aktarılan kodları doğrulamak için kullanılır.
7	Read	PIC'te yüklü olan kodları bilgisayara aktarmak için kullanılır.
8	Blank	PIC içerisindeki kodları silmek için kullanılır.
9	Fuses	PIC'in konfigürasyon ayarlarını görmek/değiştirmek için kullanılır. <i>Deneyler süresinde ayarlar buradan değil, kod içerisinde __CONFIG tanımlaması ile yapılacaktır.</i>
10	Chip Selector	Kodların aktarılacağı PIC modeli seçimi için kullanılır.
11	Code Box	Hafızaya alınan kodların görüntülediği kutudur.

Bir kodu PIC'e aktarmak için önce PIC modeli seçilmeli, sonra yüklenecek kodlar Load butonuna basılarak gelecek ekrandan seçilmeli ve en son olarak Program butonuna basılarak PIC programlanmalıdır. PIC programlandıktan sonra deney seti otomatik olarak çalışma moduna geçecektir ve yüklenen kod çalışmaya başlayacaktır.

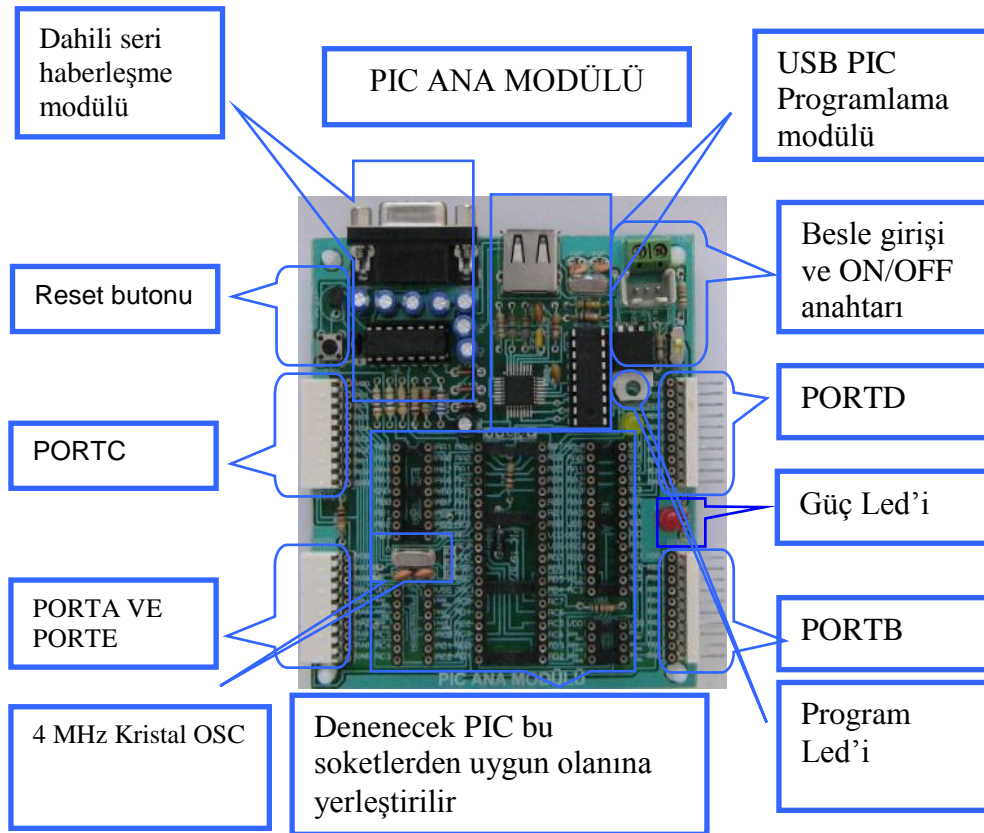
M-PIC MODÜLER DENEY SETİ İLE PROGRAMLAMAYA GİRİŞ VE PORT İŞLEMLERİ

Amaç: Bu deneyde, M-PIC modüler deney setini tanımak, bilgisayarda yazılan bir kodun deney setinde çalıştırılarak, çeşitli port işlemlerinin nasıl yapılacağını uygulayarak görmek amaçlanmıştır.

Açıklayıcı Bilgi

M-PIC modüler deney seti, bir ana modül ve ona ek olarak çeşitli modüllere sahiptir. Bu deney seti kullanılarak yapılacak deneylerde kodlar, bilgisayarda yazılıp MPASMWIN aracılığı ile derlendikten sonra deney setine aktarılarak çalıştırılacaktır. Kodların yazılıp derlenmesi ve deney setine aktarılması ile ilgili bilgiler, “**Genel Bilgiler**” başlığı altında yer almaktadır.

Modüler PIC deney setinin en önemli parçası PIC ana modülüdür. Çalışılacak PIC bu modüle yerleştirilir. Programlama, diğer modüllerle bağlantı, seri haberleşme vb. işlemler bu modül ile yapılır. PIC ana modülü üzerindeki bölümler şekilde görülmektedir.



PIC ana modülünün bilgisayar bağlantısı USB aracılığı ile yapılmaktadır. Bu bağlantı, PIC ana modülü üzerinde yer alan PIC16F628A aracılığı ile yapılmaktadır. Bundan dolayı PIC ana modül üzerinde bulunan PIC16F628A entegresi yerinden çıkarılmamalı, başka bir amaçla kullanılmamalıdır. Aksi halde PIC ana modülü üzerinde yer alan programlayıcı kullanılmaz duruma gelebilir.

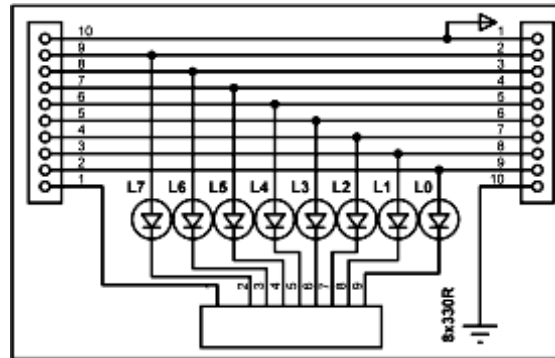
Program ledi, bilgisayar ile PIC ana modülünün haberleştiğini görsel olarak belirten elemandır. USB kablosu deneme kartına her takıldığında yanıp sönerek bilgisayar ile FT232'nin haberleştiğini göstermektedir. Bilgisayar ile haberleşme sağlandığında led yanık kalır. Program ledinin bir diğer görevi de programlama başladığı anda sönerek, programlama bittikten sonra tekrar yanmaktır.

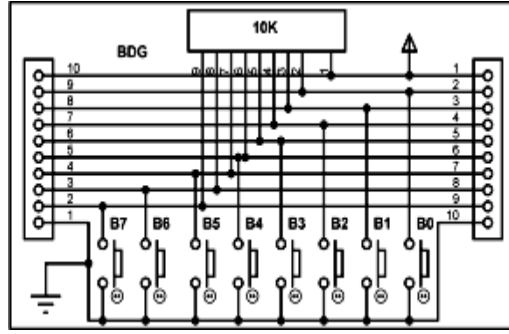
PIC ana modülü ile PIC programlamak için bir güç kaynağına ihtiyaç vardır. Bu güç kaynağı iki şekilde sağlanabilmektedir:

1. ALTAŞ güç kaynağı kullanılabilir. Bu güç kaynağı, ayarlanabilir çıkış vermektedir. Ana modülün sağ üst bölümünde yer alan 3 pinli tunik sokete, bu güç kaynağı direkt bağlanabilmektedir. Gerekli besleme devresi, PIC ana modülün üzerinde mevcuttur.
2. +5V güç kaynağı modülü kullanılabilir. 9 voltluk bir pil ve buna bağlı bir regülatör ile yapılan bu güç kaynağı modülü, PIC ana modülünün beslemesinde kullanılabilir.

Güç ledi, güç kaynağının bağlı olup olmadığını göstermektedir. Güç kaynağı bağlı ve ON/OFF anahtarı ON konumundayken güç ledi yanar. Bu ledin yanması, deneme yapılacak PIC'in beslemesinin var olduğunu gösterir. Programlama yapılırken güç ledinin yandığından emin olunmalıdır.

Aşağıda, port işlemleri ile ilgili çeşitli örnekler yer almaktadır. Bu örnekler için kullanılacak modüllerin devre şemaları şekillerde verilmiştir. Bu devre şemaları dikkate alınmalıdır.





Şekil 8 - Buton Modülü Devre Şeması

Örnek 1 – B Portuna h'55' Verisini Gönderme

```

LIST          P=16F628A
INCLUDE      "P16F628A.INC"

__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

BASLA
MOVLW      h'55'
MOVWF      PORTB

DONGU
GOTO       DONGU

END

```

Örnek 2 – A Portundaki Veriyi B Portuna Gönderme (1 Defa)

```

LIST          P=16F628A
INCLUDE      "P16F628A.INC"

__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

```

```

        CLRF          PORTB
        BANKSEL      TRISB
        CLRF          TRISB
        MOVLW        h'FF'
        MOVWF        TRISA
        BANKSEL      PORTB

        MOVLW        h'07'
        MOVWF        CMCON

BASLA
        MOVF          PORTA, W
        MOVWF        PORTB

DONGU
        GOTO         DONGU

        END

```

Örnek 3 – A Portundaki Veriyi B Portuna Gönderme (Sonsuz Döngü)

```

        LIST          P=16F628A
        INCLUDE      "P16F628A.INC"

        __CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
        _BODEN_OFF & _LVP_OFF & _CP_OFF

        ORG          h'0000'

        CLRF          PORTB
        BANKSEL      TRISB
        CLRF          TRISB
        MOVLW        h'FF'
        MOVWF        TRISA
        BANKSEL      PORTB

        MOVLW        h'07'
        MOVWF        CMCON

BASLA
        MOVF          PORTA, W
        MOVWF        PORTB
        GOTO         BASLA

        END

```

Örnek 4 – A Portundaki Verileri Tersleyerek B Portuna Gönderme

```
LIST          P=16F628A
INCLUDE      "P16F628A.INC"

__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

BASLA

COMF        PORTA, W
MOVWF      PORTB

DONGU

GOTO       DONGU

END
```

Örnek 5 – A Portunun 0. Pinine Basıldığında Led Yakma

```
LIST          P=16F628A
INCLUDE      "P16F628A.INC"

__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

BASLA
```

```

        BTFSS      PORTA, 0
        GOTO      LEDYAK
        GOTO      BASLA

LEDYAK
        MOVLW     h'FF'
        MOVWF     PORTB
        GOTO      BASLA

        END

```

Örnek 6 – A Portunun 0. Ve 1. Pini İle Led Kontrolü

```

        LIST      P=16F628A
        INCLUDE   "P16F628A.INC"

        __CONFIG  _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
        _BODEN_OFF & _LVP_OFF & _CP_OFF

        ORG       h'0000'

        CLRF      PORTB
        BANKSEL   TRISB
        CLRF      TRISB
        MOVLW     h'FF'
        MOVWF     TRISA
        BANKSEL   PORTB

        MOVLW     h'07'
        MOVWF     CMCON

BASLA
        BTFSS     PORTA, 0
        GOTO     LEDYAK
        BTFSS     PORTA, 1
        GOTO     LEDSONDUR
        GOTO     BASLA

LEDYAK
        MOVLW     h'FF'
        MOVWF     PORTB
        GOTO     BASLA

LEDSONDUR
        MOVLW     h'00'
        MOVWF     PORTB
        GOTO     BASLA

        END

```

Örnek 7 – Basılan Butona Göre Led Yakma

```
LIST          P=16F628A
INCLUDE       "P16F628A.INC"

__CONFIG     _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

BASLA
BTFSS      PORTA, 0
GOTO       LED1YAK
BTFSS      PORTA, 1
GOTO       LED2YAK
BTFSS      PORTA, 2
GOTO       LED3YAK
BTFSS      PORTA, 3
GOTO       LED4YAK
BTFSS      PORTA, 4
GOTO       LED5YAK
BTFSS      PORTA, 5
GOTO       LED6YAK
BTFSS      PORTA, 6
GOTO       LED7YAK
BTFSS      PORTA, 7
GOTO       LED8YAK
GOTO       BASLA

LED1YAK
MOVLW      h'01'
MOVWF      PORTB
GOTO       BASLA

LED2YAK
MOVLW      h'02'
MOVWF      PORTB
GOTO       BASLA

LED3YAK
MOVLW      h'04'
MOVWF      PORTB
```

```

        GOTO      BASLA
LED4YAK
        MOVLW    h'08'
        MOVWF    PORTB
        GOTO      BASLA
LED5YAK
        MOVLW    h'10'
        MOVWF    PORTB
        GOTO      BASLA
LED6YAK
        MOVLW    h'20'
        MOVWF    PORTB
        GOTO      BASLA
LED7YAK
        MOVLW    h'40'
        MOVWF    PORTB
        GOTO      BASLA
LED8YAK
        MOVLW    h'80'
        MOVWF    PORTB
        GOTO      BASLA
        END

```

Örnek 8 – 1'er Saniye Aralıklarla Ledleri Yakıp Söndürme

```

        LIST      P=16F628A
        INCLUDE   "P16F628A.INC"

        __CONFIG  _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
        _BODEN_OFF & _LVP_OFF & _CP_OFF

        ORG       h'0000'

        SAYAC1    EQU    h'20'
        SAYAC2    EQU    h'21'
        SAYAC3    EQU    h'22'

        CLRF      PORTB
        BANKSEL   TRISB
        CLRF      TRISB
        MOVLW     h'FF'
        MOVWF     TRISA
        BANKSEL   PORTB

        MOVLW     h'07'
        MOVWF     CMCON

        CLRF      SAYAC1
        CLRF      SAYAC2
        CLRF      SAYAC3

```

```

BASLA
    MOVLW    h'FF'
    MOVWF   PORTB
    CALL    GECIKME
    MOVLW    h'00'
    MOVWF   PORTB
    CALL    GECIKME
    GOTO    BASLA

GECIKME
    ;999990 PERİYOT
    MOVLW    0x07
    MOVWF   SAYAC1
    MOVLW    0x2F
    MOVWF   SAYAC2
    MOVLW    0x03
    MOVWF   SAYAC3
GECIKME_0
    DECFSZ  SAYAC1, f
    GOTO    $+2
    DECFSZ  SAYAC2, f
    GOTO    $+2
    DECFSZ  SAYAC3, f
    GOTO    GECIKME_0

    ;6 PERİYOT
    GOTO    $+1
    GOTO    $+1
    GOTO    $+1

    ;4 PERİYOT (CALL İLE BİRLİKTE)
    RETURN
    END

```

Örnek 9 – 1'er Saniye Aralıklarla Ledleri Artan Şekilde Yakma

```

LIST        P=16F628A
INCLUDE     "P16F628A.INC"

__CONFIG   _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG        h'0000'

SAYAC1     EQU    h'20'
SAYAC2     EQU    h'21'
SAYAC3     EQU    h'22'

VERI       EQU    h'23'

```

```
CLRF      PORTB
BANKSEL   TRISB
CLRF      TRISB
MOVLW    h'FF'
MOVWF     TRISA
BANKSEL   PORTB
```

```
MOVLW    h'07'
MOVWF     CMCON
```

```
CLRF      SAYAC1
CLRF      SAYAC2
CLRF      SAYAC3
```

```
MOVLW    h'01'
MOVWF     VERI
```

BASLA

```
MOVF      VERI, W
MOVWF     PORTB
CALL      GECIKME
MOVLW    h'00'
MOVWF     PORTB
CALL      GECIKME
RLF       VERI, W
IORLW    h'01'
MOVWF     VERI
GOTO     BASLA
```

GECIKME

```
                ;999990 PERİYOT
MOVLW    0x07
MOVWF     SAYAC1
MOVLW    0x2F
MOVWF     SAYAC2
MOVLW    0x03
MOVWF     SAYAC3
```

GECIKME_0

```
DECFSZ   SAYAC1, f
GOTO     $+2
DECFSZ   SAYAC2, f
GOTO     $+2
DECFSZ   SAYAC3, f
GOTO     GECIKME_0
```

```
                ;6 PERİYOT
GOTO     $+1
GOTO     $+1
GOTO     $+1
```

```
                ;4 PERİYOT (CALL İLE BİRLİKTE)
```

RETURN
END

Deneyin Yapılışı:

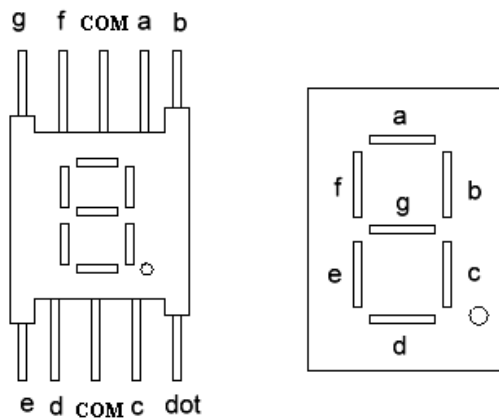
1. 50 Hz, aralarında 120° faz farkı bulunan 3 faz kare dalga üretilmek istenmektedir. Bunun için gerekli programı yazınız. Çıkış olarak B portunu kullanınız.
2. Basılan iki butonun değerini toplayarak LED'lerde binary olarak gösteren programı yazınız.
3. Basılan iki butonun değerini VE'leyerek sonucu LED'lerde binary olarak gösteren programı yazınız.
4. Basılan iki butonun değerini VEYA'layarak sonucu LED'lerde binary olarak gösteren programı yazınız.
5. Basılan iki butonun değerini XOR'layarak sonucu LED'lerde binary olarak gösteren programı yazınız.
6. Basılan butonun değeri kadar LED'i aynı anda yakan programı yazınız.
7. A portunun 0. pinine bağlı olan butona her basışta PORTB'yi bir bit sola kaydıran programı yazınız.
8. Basılan butonun değerinden geriye doğru 1'er saniye aralıklarla sayan ve değer 0 olunca sayma işlemini durduran programı yazınız.

7 SEGMENT UYGULAMALARI

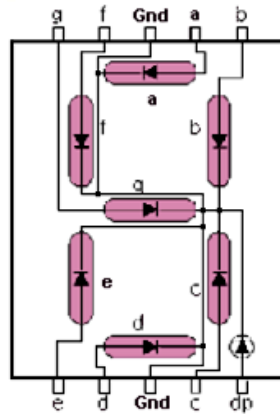
Amaç: Bu deneyde, M-PIC ana modülünü ve 7 segment modüllerini kullanarak çeşitli işlemler sonucu elde edilen bilgileri göstermek amaçlanmıştır.

Açıklayıcı Bilgi

Deney setinde tekli ve dörtlü olmak üzere iki adet ortak katot 7 segment modülü ve üzerinde 7448 bulunan binary→7 segment dönüştürücü modülü mevcuttur. Ortak katot 7 segment display yapısı ve iç yapısı şekilde görülmektedir.



Şekil 9 - Ortak Katot 7 Segment Display



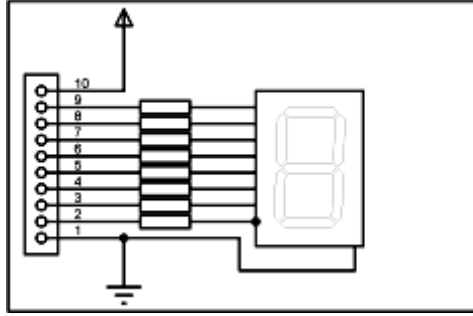
Şekil 10 - Ortak Katot 7 Segment Display İç Yapısı

Şekilden de görüldüğü gibi bir 7 segment display, görüntüyü oluşturacak 7 adet led ve noktayı oluşturacak 1 adet led olmak üzere 8 led içermektedir. Herhangi bir görüntünün oluşturulabilmesi için COM/GND uçlarının topraklanması ve ışık vermesi istenen lede uygun gerilimin uygulanması gerekmektedir. 0-9 arasındaki sayıların oluşturulabilmesi için, (ek olarak nokta da kullanılacaksa) toplam 8 pine ihtiyaç vardır ve rakamların oluşması için gerekli olan

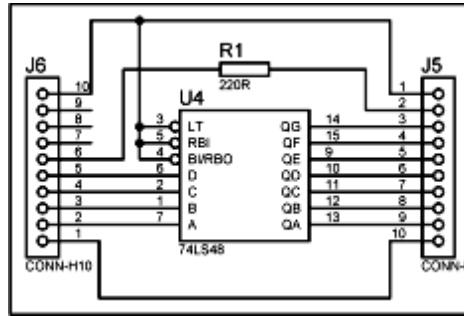
dönüşüm tablolarının programa dahil edilmesi gerekmektedir. Bununla ilgili uygulamalar, deney kapsamında yapılacaktır.

Deney setinde de modül olarak yer alan ve 7448 entegresi içeren binary→7 segment dönüştürücü modülü kullanarak, PIC'in sadece 4 pini aracılığı ile bu işlemin gerçekleştirilebildiği de deney sırasında uygulanacaktır.

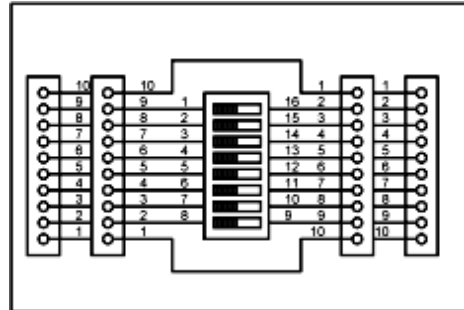
Aşağıda, 7 segment uygulamaları ile ilgili çeşitli örnekler yer almaktadır. Bu örnekler için kullanılacak modüllerin devre şemaları şekillerde verilmiştir. Bu devre şemaları dikkate alınmalıdır.



Şekil 11 - Ortak Katot 7 Segment Modülü Devre Şeması



Şekil 12 - Binary→7 Segment Dönüştürücü Modülü Devre Şeması



Şekil 13 - Port Çaprazlama/Bağlantı Kesme Modülü

Örnek 1 – Butona Bastıkça Artan Değeri 7 Segmentte Gösterme

```
LIST          P=16F628A
INCLUDE      "P16F628A.INC"

__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

SAYAC1      EQU    h'20'
SAYAC2      EQU    h'21'
SAYAC3      EQU    h'22'

VERI        EQU    h'23'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

CLRF        SAYAC1
CLRF        SAYAC2
CLRF        SAYAC3

CLRF        VERI

BASLA
BTFSS      PORTA, 0
CALL      ARTIR
GOTO      BASLA

ARTIR
BTFSS      PORTA, 0
GOTO      $-1
MOVF      VERI, W
CALL      SEG7TABLE
MOVWF     PORTB
INCF     VERI, W
ANDLW    h'0F'
MOVWF     VERI
RETURN

SEG7TABLE
ADDWF     PCL, F
RETLW    h'3F'
```

```
RETLW    h'06'  
RETLW    h'5B'  
RETLW    h'4F'  
RETLW    h'66'  
RETLW    h'6D'  
RETLW    h'7D'  
RETLW    h'07'  
RETLW    h'7F'  
RETLW    h'6F'  
RETLW    h'77'  
RETLW    h'7C'  
RETLW    h'39'  
RETLW    h'5E'  
RETLW    h'79'  
RETLW    h'71'
```

```
END
```

Örnek 2 – Basılan Butonun Değerini 7 Segmentte Gösterme

```
LIST      P=16F628A  
INCLUDE   "P16F628A.INC"  
  
__CONFIG  _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &  
_BODEN_OFF & _LVP_OFF & _CP_OFF  
  
ORG       h'0000'  
  
SAYAC1    EQU    h'20'  
SAYAC2    EQU    h'21'  
SAYAC3    EQU    h'22'  
  
VERI      EQU    h'23'  
  
CLRF      PORTB  
BANKSEL   TRISB  
CLRF      TRISB  
MOVLW    h'FF'  
MOVWF    TRISA  
BANKSEL   PORTB  
  
MOVLW    h'07'  
MOVWF    CMCON  
  
CLRF      SAYAC1  
CLRF      SAYAC2  
CLRF      SAYAC3  
  
CLRF      VERI
```

BASLA

```
    BTFSS    PORTA, 0
    CALL     YAZ0
    BTFSS    PORTA, 1
    CALL     YAZ1
    BTFSS    PORTA, 2
    CALL     YAZ2
    BTFSS    PORTA, 3
    CALL     YAZ3
    BTFSS    PORTA, 4
    CALL     YAZ4
    BTFSS    PORTA, 5
    CALL     YAZ5
    BTFSS    PORTA, 6
    CALL     YAZ6
    BTFSS    PORTA, 7
    CALL     YAZ7
    GOTO     BASLA
```

YAZ0

```
    MOVLW   h'00'
    MOVWF   VERI
    CALL    GOSTER
    RETURN
```

YAZ1

```
    MOVLW   h'01'
    MOVWF   VERI
    CALL    GOSTER
    RETURN
```

YAZ2

```
    MOVLW   h'02'
    MOVWF   VERI
    CALL    GOSTER
    RETURN
```

YAZ3

```
    MOVLW   h'03'
    MOVWF   VERI
    CALL    GOSTER
    RETURN
```

YAZ4

```
    MOVLW   h'04'
    MOVWF   VERI
    CALL    GOSTER
    RETURN
```

YAZ5

```
    MOVLW   h'05'
```

```

        MOVWF    VERI
        CALL     GOSTER
        RETURN

YAZ6
        MOVLW   h'06'
        MOVWF   VERI
        CALL     GOSTER
        RETURN

YAZ7
        MOVLW   h'07'
        MOVWF   VERI
        CALL     GOSTER
        RETURN

GOSTER
        BTFSS   PORTA, 0
        GOTO    $-1
        MOVF    VERI, W
        CALL    SEG7TABLE
        MOVWF   PORTB
        RETURN

SEG7TABLE
        ADDWF   PCL, F
        RETLW   h'3F'
        RETLW   h'06'
        RETLW   h'5B'
        RETLW   h'4F'
        RETLW   h'66'
        RETLW   h'6D'
        RETLW   h'7D'
        RETLW   h'07'
        RETLW   h'7F'
        RETLW   h'6F'
        RETLW   h'77'
        RETLW   h'7C'
        RETLW   h'39'
        RETLW   h'5E'
        RETLW   h'79'
        RETLW   h'71'

        END

```

Örnek 3 – 1'er Saniye Aralıkla Artan Değeri 7 Segmentte Gösterme

```
LIST          P=16F628A
INCLUDE       "P16F628A.INC"

__CONFIG     _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG          h'0000'

SAYAC1      EQU   h'20'
SAYAC2      EQU   h'21'
SAYAC3      EQU   h'22'

VERI        EQU   h'23'

CLRF        PORTB
BANKSEL     TRISB
CLRF        TRISB
MOVLW      h'FF'
MOVWF      TRISA
BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

CLRF        SAYAC1
CLRF        SAYAC2
CLRF        SAYAC3

CLRF        VERI

BASLA
CALL        GOSTER
CALL        GECIKME
INCF       VERI, W
ANDLW     h'0F'
MOVWF     VERI
GOTO      BASLA

GOSTER
BTFSS     PORTA, 0
GOTO     $-1
MOVF     VERI, W
CALL     SEG7TABLE
MOVWF    PORTB
RETURN

SEG7TABLE
ADDWF    PCL, F
RETLW   h'3F'
```

```
RETLW    h'06'  
RETLW    h'5B'  
RETLW    h'4F'  
RETLW    h'66'  
RETLW    h'6D'  
RETLW    h'7D'  
RETLW    h'07'  
RETLW    h'7F'  
RETLW    h'6F'  
RETLW    h'77'  
RETLW    h'7C'  
RETLW    h'39'  
RETLW    h'5E'  
RETLW    h'79'  
RETLW    h'71'
```

GECIKME

```
                ;999990 PERİYOT  
MOVLW    0x07  
MOVWF    SAYAC1  
MOVLW    0x2F  
MOVWF    SAYAC2  
MOVLW    0x03  
MOVWF    SAYAC3
```

GECIKME_0

```
DECFSZ   SAYAC1, f  
GOTO     $+2  
DECFSZ   SAYAC2, f  
GOTO     $+2  
DECFSZ   SAYAC3, f  
GOTO     GECIKME_0
```

```
                ;6 PERİYOT  
GOTO     $+1  
GOTO     $+1  
GOTO     $+1
```

```
                ;4 PERİYOT (CALL İLE BİRLİKTE)  
RETURN  
END
```

Örnek 4 – 1'er Saniye Aralıkla Artan Değeri 7 Segmentte Gösterme (7448 Entegreli)

```
LIST      P=16F628A  
INCLUDE   "P16F628A.INC"  
  
__CONFIG  __INTRC_OSC_NOCLKOUT & __WDT_OFF & __PWRTE_ON & __MCLRE_OFF &  
__BODEN_OFF & __LVP_OFF & __CP_OFF  
  
ORG       h'0000'
```

```

SAYAC1    EQU    h'20'
SAYAC2    EQU    h'21'
SAYAC3    EQU    h'22'

VERI      EQU    h'23'

CLRF      PORTB
BANKSEL   TRISB
CLRF      TRISB
MOVLW    h'FF'
MOVWF    TRISA
BANKSEL   PORTB

MOVLW    h'07'
MOVWF    CMCON

CLRF      SAYAC1
CLRF      SAYAC2
CLRF      SAYAC3

CLRF      VERI

```

BASLA

```

MOVF     VERI, W
MOVWF    PORTB
CALL     GECIKME
INCF     VERI, W
ANDLW    h'0F'
MOVWF    VERI
GOTO     BASLA

```

GECIKME

```

;999990 PERİYOT
MOVLW    0x07
MOVWF    SAYAC1
MOVLW    0x2F
MOVWF    SAYAC2
MOVLW    0x03
MOVWF    SAYAC3

```

GECIKME_0

```

DECFSZ   SAYAC1, f
GOTO     $+2
DECFSZ   SAYAC2, f
GOTO     $+2
DECFSZ   SAYAC3, f
GOTO     GECIKME_0

```

```

;6 PERİYOT
GOTO     $+1
GOTO     $+1
GOTO     $+1

```

```

;4 PERİYOT (CALL İLE BİRLİKTE)
RETURN
END

```

Örnek 5 – Kesme ve 4x7 Segment Display Uygulaması

Bu örnekte mikrodenetleyici olarak PIC16F877A, modül olarak B7SEG, Port Çaprazlama-Bağlantı Kesme ve 4x7 Segment Display modülleri kullanılacaktır. Saniye hesabı için TMR0 kesmesi kullanılacaktır. TMR0'a yüklenen değer,

$$TMR0 = 256 - \left(\frac{\text{Gecikme Süresi} \times f_{osc}}{4 \times \text{Preskalar}} \right)$$

formülü ile hesaplanarak, 4MHz kristal osilatör, 256 preskalar ve 10ms gecikme süresi için değer, 216.9 olarak elde edilmektedir. Bu da yaklaşık olarak 217 kabul edilebilir.

```

LIST          P=16F877A
INCLUDE       "P16F877A.INC"

__CONFIG     _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

ORG          h'0000'
GOTO         PROGRAM
ORG          h'0004'
GOTO         KESME

PROGRAM

CBLOCK      h'20'
    SAYAC1
    SAYAC2

    BIRLER
    ONLAR
    YUZLER
    BINLER
    ARTFLAG
    SAY10MS
    SAKLA_W
    SAKLA_S
ENDC

CLRF        PORTB
CLRF        PORTD
BANKSEL     TRISB
CLRF        TRISB
CLRF        TRISD

MOVLW      b'00010111'
MOVWF      OPTION_REG

```

BANKSEL	PORTB
BSF	INTCON, 7
BSF	INTCON, 5
CLRF	SAYAC1
CLRF	SAYAC2
CLRF	ARTFLAG
CLRF	SAKLA_W
CLRF	SAKLA_S
CLRF	BIRLER
CLRF	ONLAR
CLRF	YUZLER
CLRF	BINLER
BASLA	
BTFSC	ARTFLAG, 0
CALL	SAYIART
CLRF	PORTB
MOVLW	h'01'
MOVWF	PORTB
MOVF	BIRLER, W
MOVWF	PORTD
CALL	GECIKME
CLRF	PORTB
MOVLW	h'02'
MOVWF	PORTB
MOVF	ONLAR, W
MOVWF	PORTD
CALL	GECIKME
CLRF	PORTB
MOVLW	h'04'
MOVWF	PORTB
MOVF	YUZLER, W
MOVWF	PORTD
CALL	GECIKME
CLRF	PORTB
MOVLW	h'08'
MOVWF	PORTB
MOVF	BINLER, W
MOVWF	PORTD
CALL	GECIKME
GOTO	BASLA
SAYIART	
INCF	BIRLER, W
ANDLW	H'0F'
MOVWF	BIRLER
MOVF	BIRLER, W

```

        SUBLW    h'0A'
        BTFSC   STATUS, Z
        CALL    ONLARART

        MOVF    ONLAR, W
        SUBLW   h'0A'
        BTFSC   STATUS, Z
        CALL    YUZLERART

        MOVF    YUZLER, W
        SUBLW   h'0A'
        BTFSC   STATUS, Z
        CALL    BINLERART

        BCF     ARTFLAG, 0
        RETURN

ONLARART
        CLRF    BIRLER
        INCF    ONLAR, F
        RETURN

YUZLERART
        INCF    YUZLER, W
        ANDLW   h'0F'
        MOVWF   YUZLER
        CLRF    BIRLER
        CLRF    ONLAR
        RETURN

BINLERART
        INCF    BINLER, W
        ANDLW   h'0F'
        MOVWF   BINLER
        CLRF    BIRLER
        CLRF    ONLAR
        CLRF    YUZLER
        RETURN

GECIKME
        MOVLW   0x1F
        MOVWF   SAYAC1
        MOVLW   0x04
        MOVWF   SAYAC2
GECIKME_0
        DECFSZ  SAYAC1, F
        GOTO    $+2
        DECFSZ  SAYAC2, F
        GOTO    GECIKME_0

        GOTO    $+1

```

```

RETURN

KESME
MOVWF    SAKLA_W
SWAPF    STATUS, W
MOVWF    SAKLA_S

BCF      INTCON, 2
MOVLW    h'D2' ; 217 SAYISINDAN BASLASIN
MOVWF    TMRO
INCF     SAY10MS, F
MOVLW    D'100'
SUBWF    SAY10MS, W
BTFSS    STATUS, Z
GOTO     KESMESONU
CLRF     SAY10MS
BSF      ARTFLAG, 0
KESMESONU
SWAPF    SAKLA_S, W
MOVWF    STATUS
SWAPF    SAKLA_W, F
SWAPF    SAKLA_W, W
RETFIE
END

```

Deneyin Yapılışı:

1. Basılan tuşun karesini alarak, sonuç 1 haneli ise sonucu, 2 haneli ise birler basamağını display üzerinde gösteren programı yazınız.
2. Sonucu 0-F aralığında olacak şekilde butonlara basılan 2 sayının toplamını display üzerinde gösteren programı yazınız.
3. Sonucu 0-F aralığında olacak şekilde butonlara basılan 2 sayının çarpımını display üzerinde gösteren programı yazınız.
4. Fibonacci serisini **hesaplayarak**, sonuç 1 haneli ise sonucu, 2 haneli ise birler basamağını display üzerinde gösteren programı yazınız.
5. 4'lü 7 segment display modülünü kullanarak 9999'dan geriye 1'er saniye arayla sayan program yazınız.

KAYAN YAZI UYGULAMALARI

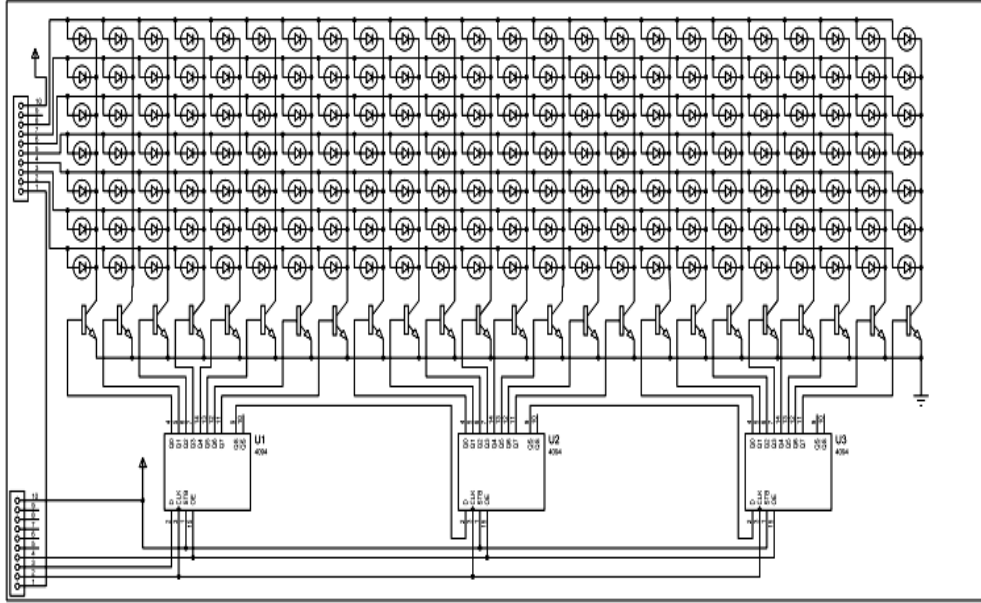
Amaç: Bu deneyde, M-PIC deney setinde yer alan kayan yazı modülü kullanılarak, matris biçiminde dizilmiş onlarca led üzerinde tarama yöntemiyle görüntü oluşturma uygulamaları yapılacaktır.

Açıklayıcı Bilgi

M-PIC deney setinde yer alan kayan yazı modülü, satır ve sütun taramaları için iki port kullanmaktadır. Bir sütunda her bir satır için veri, D portundan 8-bit olarak uygulanmaktadır. Sütunların taranması için ise B portunun 0., 1. ve 2. pinleri kullanılmaktadır. Sütun taramaları için modül üzerinde, shift register entegresi olan 4094 entegreleri mevcuttur. Her bir entegrenin 8 çıkışı olduğundan ve modül üzerinde 24 sütun yer aldığından toplam 3 adet shift register yeterli olmaktadır. Sütun tarama işlemi şu şekilde yapılır:

1. 2. pin lojik 1 seviyesine çekilir.
2. 0. pin lojik 1 seviyesine çekilir.
3. 1. pin clock pinidir ve her yükselen kenarda o an 0. pinde yer alan değer, shift registerin ilk pinine atanır ve clock ucunun her yükselen kenarında bu veri kayar. Bunun için ilk veri 1 olarak verildikten sonra bütün sütunlar taranana kadar veri 0 olarak verilir. 9. yükselen kenarda veri, 4094 entegrelerinin Q çıkışlarının birbirlerinin Data pinlerine bağlı olmasından dolayı, ikinci shift registerde kaymaya devam eder. Veri, 24. yükselen kenarda son shift registerin son pinine ulaşır. Bundan sonra taramanın tekrar başa dönmesi için işlemler birinci adımdan itibaren tekrarlanmalıdır.

Sütun taraması yapılırken her kaydırmada, aktif olan sütuna, o sütunun verisi gönderilmelidir. Şekilde, deney setinde yer alan kayan yazı modülünün devre şeması yer almaktadır. Sonraki sayfalarda, bu konu ile ilgili örnekler yer almaktadır. Kayan yazı modülünün daha rahat kullanılması için PIC16F877A tercih edilmiştir.



Şekil 14 - Kayan Yazı Modülü Devre Şeması

Örnek 1 – Harfleri Kayan Yazı Modülünde Sabit Olarak Gösterme

```

LIST          P=16F877A
INCLUDE       "P16F877A.INC"

__CONFIG     _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

ORG          h'0000'

SAYAC1      EQU    h'20'
SAYAC2      EQU    h'21'

SUTSAY      EQU    h'22'

CLRF        PORTB
CLRF        PORTD
BANKSEL     TRISB
CLRF        TRISB
CLRF        TRISD
BANKSEL     PORTB

CLRF        SAYAC1
CLRF        SAYAC2

CLRF        SUTSAY

BSF         PORTB, 2
BSF         PORTB, 0
BSF         PORTB, 1
BCF         PORTB, 1

```

```

        BCF          PORTB, 0

BASLA
        MOVF        SUTSAY, W
        SUBLW      d'24'
        BTFSC      STATUS, Z
        CALL       SIFIRLA

        MOVF        SUTSAY, W
        CALL       YAZITABLO
        MOVWF      PORTD

        INCF       SUTSAY, F

        CALL       GECIKME

        BCF        PORTB, 0
        BSF        PORTB, 1
        BCF        PORTB, 1

        GOTO       BASLA

SIFIRLA
        CLRF       SUTSAY

        BSF        PORTB, 0
        BSF        PORTB, 1
        BCF        PORTB, 1
        BCF        PORTB, 0
        RETURN

YAZITABLO
        ADDWF      PCL, F
        RETLW     b'00011111'
        RETLW     b'00100100'
        RETLW     b'01000100'
        RETLW     b'00100100'
        RETLW     b'00011111'
        RETLW     b'00000000'
        RETLW     b'01111111'
        RETLW     b'01001001'
        RETLW     b'01001001'
        RETLW     b'01001001'
        RETLW     b'00110110'
        RETLW     b'00000000'
        RETLW     b'00011100'
        RETLW     b'00100010'
        RETLW     b'01000001'
        RETLW     b'01000001'
        RETLW     b'00100010'
        RETLW     b'00000000'

```

```

RETLW    b'01111111'
RETLW    b'01000001'
RETLW    b'01000001'
RETLW    b'00100010'
RETLW    b'00011100'
RETLW    b'00000000'

GECIKME
    MOVLW    0xC7
    MOVWF    SAYAC1
GECIKME_0
    DECFSZ   SAYAC1, F
    GOTO     GECIKME_0

    GOTO     $+1
    RETURN

    END

```

Örnek 2 – Harfleri Kayan Yazı Modülünde Kayar Şekilde Gösterme

```

LIST      P=16F877A
INCLUDE   "P16F877A.INC"

__CONFIG  _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

ORG       h'0000'
GOTO     PROGRAM
ORG       h'0004'
GOTO     KESME

PROGRAM

SAYAC1    EQU    h'20'
SAYAC2    EQU    h'21'

SUTSAY    EQU    h'22'
BASLNG    EQU    h'23'

SAY10MS   EQU    h'24'
SAKLA_W   EQU    h'25'
SAKLA_S   EQU    h'26'

CLRF      PORTB
CLRF      PORTD
BANKSEL   TRISB
CLRF      TRISB
CLRF      TRISD
MOVLW    b'00010111'

```

```

MOVWF    OPTION_REG
BANKSEL  PORTB

BSF      INTCON, 7
BSF      INTCON, 5

CLRF     SAYAC1
CLRF     SAYAC2

CLRF     SUTSAY
CLRF     BASLNG

CLRF     SAKLA_W
CLRF     SAKLA_S
CLRF     SAY10MS

BSF      PORTB, 2
BSF      PORTB, 0
BSF      PORTB, 1
BCF      PORTB, 1
BCF      PORTB, 0

BASLA
MOVF     SUTSAY, W
SUBLW   d'24'
BTFSC   STATUS, Z
CALL    SIFIRLA

MOVF     SUTSAY, W
ADDWF   BASLNG, W
CALL    YAZITABLO
MOVWF   PORTD

INCF    SUTSAY, F

CALL    GECIKME

BCF     PORTB, 0
BSF     PORTB, 1
BCF     PORTB, 1

GOTO    BASLA

SIFIRLA
CLRF    SUTSAY

BSF     PORTB, 0
BSF     PORTB, 1
BCF     PORTB, 1
BCF     PORTB, 0

```

RETURN

YAZITABLO

ADDWF	PCL, F
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00000000'
RETLW	b'00011111'
RETLW	b'00100100'
RETLW	b'01000100'
RETLW	b'00100100'
RETLW	b'00011111'
RETLW	b'00000000'
RETLW	b'01111111'
RETLW	b'01001001'
RETLW	b'01001001'
RETLW	b'01001001'
RETLW	b'00110110'
RETLW	b'00000000'
RETLW	b'00011100'
RETLW	b'00100010'
RETLW	b'01000001'
RETLW	b'01000001'
RETLW	b'00100010'
RETLW	b'00000000'
RETLW	b'01111111'
RETLW	b'01000001'
RETLW	b'01000001'
RETLW	b'00100010'
RETLW	b'00011100'


```

MOVF      BASLNG, W
SUBLW    d'48'
BTFSC    STATUS, Z
CLRF     BASLNG
INCF     BASLNG, F

```

KESMESONU

```

SWAPF    SAKLA_S, W
MOVWF    STATUS
SWAPF    SAKLA_W, F
SWAPF    SAKLA_W, W
RETFIE

```

END

Örnek 3 – 1'er Saniye Aralıkla Artan Değeri Kayan Yazı Modülünde Sabit Olarak Gösterme

Bu örnekte saniye hesabı için TMR0 kesmesi kullanılacaktır. TMR0'a yüklenen değer,

$$TMR0 = 256 - \left(\frac{\text{Gecikme Süresi} \times f_{osc}}{4 \times \text{Preskalar}} \right)$$

formülü ile hesaplanarak, 4MHz kristal osilatör, 256 preskalar ve 10ms gecikme süresi için değer, 216.9 olarak elde edilmektedir. Bu da yaklaşık olarak 217 kabul edilebilir.

```

LIST      P=16F877A
INCLUDE   "P16F877A.INC"

__CONFIG  _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

ORG       h'0000'
GOTO     PROGRAM
ORG       h'0004'
GOTO     KESME

PROGRAM

CBLOCK   h'20'
    SAYAC1
    SAYAC2

    BIRLER
    ONLAR
    YUZLER
    BINLER
    ARTFLAG
    SAY10MS
    SAKLA_W
    SAKLA_S

```

```

        TMP1
        TMP2
        TMP3
ENDC

CLRF    PORTB
CLRF    PORTD
BANKSEL TRISB
CLRF    TRISB
CLRF    TRISD
MOVLW  b'00010111'
MOVWF  OPTION_REG
BANKSEL PORTB

BSF     INTCON, 7
BSF     INTCON, 5

CLRF    SAYAC1
CLRF    SAYAC2

CLRF    ARTFLAG
CLRF    BIRLER
CLRF    ONLAR
CLRF    YUZLER
CLRF    BINLER

CLRF    SAKLA_W
CLRF    SAKLA_S
CLRF    SAY10MS
CLRF    TMP1
CLRF    TMP2
CLRF    TMP3

BSF     PORTB, 2
BSF     PORTB, 0
BSF     PORTB, 1
BCF     PORTB, 1
BCF     PORTB, 0

```

BASLA

```

BTFSC  ARTFLAG, 0
CALL   SAYIART

MOVF   BINLER, W
CALL   SAYIYAZ

MOVF   YUZLER, W
CALL   SAYIYAZ

MOVF   ONLAR, W
CALL   SAYIYAZ

```

```

        MOVF      BIRLER, W
        CALL     SAYIYAZ

        BSF      PORTB, 0
        BSF      PORTB, 1
        BCF      PORTB, 1
        BCF      PORTB, 0

        GOTO     BASLA

SAYIYAZ
        CLRF     TMP1
        CLRF     TMP2
        CLRF     TMP3
SAYIYAZ_1
        MOVWF    TMP1
        MOVWF    TMP2
        CALL     SAYIOFFSETBUL

        MOVLW   h'06'
        MOVWF    TMP3
SAYIYAZ_2
        CALL     GECIKME
        MOVF     TMP2, W
        CALL     YAZITABLO
        MOVWF    PORTD
        INCF     TMP2, F

        BCF      PORTB, 0
        BSF      PORTB, 1
        BCF      PORTB, 1

        DECFSZ   TMP3, F
        GOTO     SAYIYAZ_2
        RETURN

SAYIOFFSETBUL
        MOVLW   h'05'
        MOVWF    TMP3
SAYIOFFSETBUL_1
        MOVF     TMP1, W
        ADDWF    TMP2, F
        DECFSZ   TMP3, F
        GOTO     SAYIOFFSETBUL_1
        MOVF     TMP2, W
        ADDLW   h'01'
        MOVWF    TMP2
        RETURN

```

SIFIRLA

```
BSF      PORTB, 0
BSF      PORTB, 1
BCF      PORTB, 1
BCF      PORTB, 0
RETURN
```

SAYIART

```
INCF     BIRLER, W
ANDLW    h'0F'
MOVWF    BIRLER

MOVF     BIRLER, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     ONLARART

MOVF     ONLAR, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     YUZLERART

MOVF     YUZLER, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     BINLERART

BCF      ARTFLAG, 0
RETURN
```

ONLARART

```
CLRF     BIRLER
INCF     ONLAR, F
RETURN
```

YUZLERART

```
INCF     YUZLER, W
ANDLW    h'0F'
MOVWF    YUZLER
CLRF     BIRLER
CLRF     ONLAR
RETURN
```

BINLERART

```
INCF     BINLER, W
ANDLW    h'0F'
MOVWF    BINLER
CLRF     BIRLER
CLRF     ONLAR
CLRF     YUZLER
RETURN
```

YAZITABLO

```

ADDWF      PCL, F

;0
RETLW     b'00000000'
RETLW     b'00111110'
RETLW     b'01000001'
RETLW     b'01000001'
RETLW     b'00111110'
RETLW     b'00000000'

;1
RETLW     b'00000000'
RETLW     b'00000000'
RETLW     b'00100001'
RETLW     b'01111111'
RETLW     b'00000001'
RETLW     b'00000000'

;2
RETLW     b'00000000'
RETLW     b'00100001'
RETLW     b'01000011'
RETLW     b'01000101'
RETLW     b'00111001'
RETLW     b'00000000'

;3
RETLW     b'00000000'
RETLW     b'00100010'
RETLW     b'01001001'
RETLW     b'01001001'
RETLW     b'00110110'
RETLW     b'00000000'

;4
RETLW     b'00000000'
RETLW     b'00001100'
RETLW     b'00110100'
RETLW     b'01111111'
RETLW     b'00000100'
RETLW     b'00000000'

;5
RETLW     b'00000000'
RETLW     b'01110010'
RETLW     b'01001001'
RETLW     b'01001001'
RETLW     b'01000110'
RETLW     b'00000000'

```

```

;6
RETLW    b'00000000'
RETLW    b'00111110'
RETLW    b'01001001'
RETLW    b'01001001'
RETLW    b'00100110'
RETLW    b'00000000'

```

```

;7
RETLW    b'00000000'
RETLW    b'01000000'
RETLW    b'01000111'
RETLW    b'01001000'
RETLW    b'01110000'
RETLW    b'00000000'

```

```

;8
RETLW    b'00000000'
RETLW    b'00110110'
RETLW    b'01001001'
RETLW    b'01001001'
RETLW    b'00110110'
RETLW    b'00000000'

```

```

;9
RETLW    b'00000000'
RETLW    b'00110010'
RETLW    b'01001001'
RETLW    b'01001001'
RETLW    b'00111110'
RETLW    b'00000000'
RETLW    b'00000000'

```

```

GECIKME
    MOVLW    0xFF
    MOVWF    SAYAC1
GECIKME_0
    DECFSZ   SAYAC1, F
    GOTO     GECIKME_0

    GOTO     $+1
    RETURN

```

```

KESME
    MOVWF    SAKLA_W
    SWAPF    STATUS, W
    MOVWF    SAKLA_S

    BCF     INTCON, 2
    MOVLW    h'D2' ; 217 SAYISINDAN BASLASIN

```

```
MOVWF    TMR0
INCF     SAY10MS, F
MOVLW   D'100'
SUBWF   SAY10MS, W
BTFS    STATUS, Z
GOTO    KESMESONU
CLRF    SAY10MS
```

```
BSF     ARTFLAG, 0
```

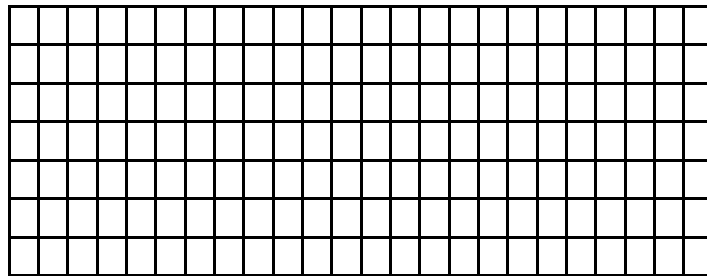
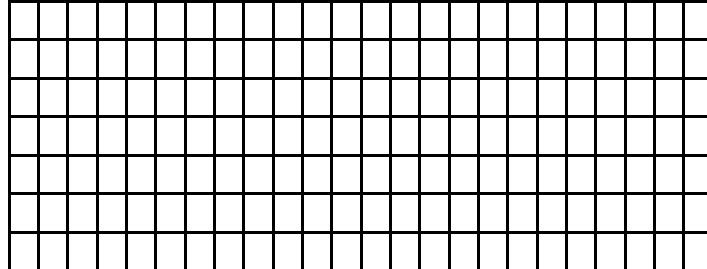
KESMESONU

```
SWAPF   SAKLA_S, W
MOVWF   STATUS
SWAPF   SAKLA_W, F
SWAPF   SAKLA_W, W
RETFIE
```

END

Deneyin Yapılışı:

1. İsminizi sabit bir şekilde kayan yazı modülünde gösteren programı yazınız.
2. İsminizi kayan yazı şekilde kayan yazı modülünde gösteren programı yazınız
3. 9999'dan geriye doğru 1'er saniye aralıklarla sayan ve değeri kayan yazı modülünde sabit bir şekilde gösteren programı yazınız.

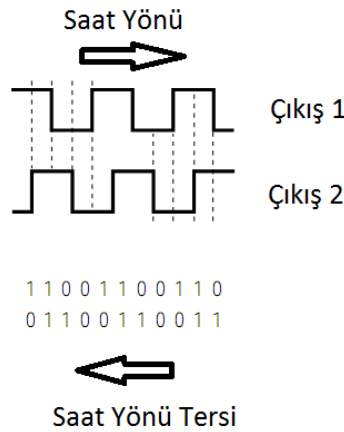


ROTARY ENCODER, ADC VE 7 SEGMENT UYGULAMALARI

Amaç: Bu deneyde, M-PIC deney setinde yer alan rotary encoder, adc, B7SEG, port çaprazlama/bağlantı kesme ve 4x7 segment display modülleri kullanılarak, rotary encoder dönüş sayısı ve analog ölçüm değerlerinin 7 segment display üzerinde görüntülemesi uygulamaları yapılacaktır.

Açıklayıcı Bilgi

M-PIC deney setinde yer alan rotary encoder modülü, üzerine bir sonsuz turlu rotary encoder ile 2 led ve direnç içermektedir. Rotary encoderin her bir adımında ledler yanıp sönmektedir. Ledlerin yanıp sönme sırası, rotary encoder dönüş yönünü göstermektedir. Ledlerdeki her bir değişim, bir adım atıldığını göstermektedir. Bu değişimleri ve dönüş yönünü algılamak için rotary encoderin 2 ucu kullanılmaktadır. Rotary encoderin dönüşü sırasında uçlarda görülen sinyaller şekilde görülmektedir.



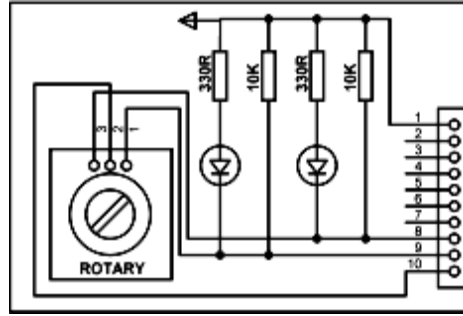
Şekil 15 - Rotary Encoder Çıkış Sinyalleri

Görüldüğü gibi rotary encoderin çıkışlarında görülen sinyaller arasında 90 derecelik faz farkı bulunmaktadır. Sağa doğru döndürme işlemi gerçekleştirildiğinde çıkışlar (10, 11, 01, 00, 10, 11, 01, 00, ...) şeklinde, sola doğru döndürme işlemi gerçekleştirildiğinde çıkışlar (00, 01, 11, 10, 00, 01, 11, 10, ...) şeklinde olmaktadır. Bunun Gray kodlaması olduğuna dikkat ediniz. Bu çıkışlar PIC'in portuna uygulandığında, çıkışların her iki değişimi Şekil 14'te görüldüğü gibi 1 adım olarak kabul edilecektir. Dönüş yönünü tespit etmek için ise aşağıdaki işlemler yapılır:

1. Portta görülen değer, 0x03 değeri ile VE'lenerek ilk değişkene kaydedilir.

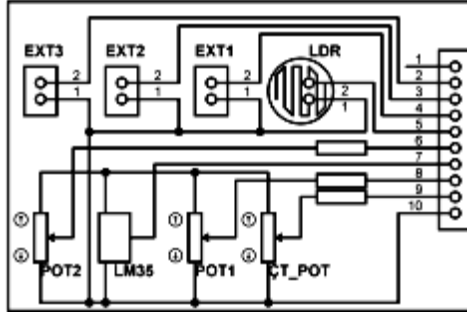
2. Döngü içerisinde port değeri alınarak 0x03 ile VE'lenir ve ilk değişken ile karşılaştırılır. Eğer ilk değişken ile aynı değer görülüyorsa rotary encoder'da herhangi bir değişim meydana gelmemiştir.
3. Görülen değer, ilk değişkendeki değerden farklıysa bir değişim gerçekleşmiştir. Bu durumda ilk değişkendeki değer 0x01 ile, porttan yeni alınan ve ikinci değişkene (aynı zamanda üçüncü değişkene) kaydedilen yeni değer 0x03 ile VE'lenir.
4. İkinci değişkendeki değer sağa kaydırılır.
5. İlk değişkendeki değer ile ikinci değişkendeki değer XOR'lanır. Sonuç 1 ise saat yönünde, sonuç 0 ise saat yönünün tersinde bir dönüş gerçekleşmiş demektir. Bu duruma göre display üzerinde görüntülenen değer arttırılır veya azaltılır.
6. Değer display üzerinde görüntülenir.
7. Üçüncü değişkendeki değer ilk değişkene aktarılarak döngü birinci adımdan başlayacak şekilde düzenlenir.

Rotary encoder modülü devre şeması Şekil 16'da verilmiştir.



Şekil 16 – Rotary Encoder Modülü Devre Şeması

Analog uygulama için ise analog giriş modülü kullanılacaktır. Analog giriş modülü, A portunun 8 pinini analog giriş olarak kullanmak üzere tasarlanmış ve üzerinde 3 adet ayarlı direnç, 1 adet LM35 sıcaklık sensörü, 1 adet LDR bulunmaktadır. Geriye kalan 3 pin için ise kullanıcının dışarıdan analog sinyal uygulayabilmesi için soketler yerleştirilmiş ve boş bırakılmıştır. Analog giriş modülünün devre şeması Şekil 17'de görülmektedir.



Şekil 17 - Analog Giriş Modülü Devre Şeması

4x7 segment modülü, rotary encoder modülü ve adc modüllerinin daha rahat kullanılması için PIC16F877A tercih edilmiştir.

Örnek 1 – Rotary Encoder Uygulaması

```

LIST          P=16F877A
INCLUDE       "P16F877A.INC"

__CONFIG     _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

CBLOCK       h'20'
             SAYAC1
             SAYAC2

             BIRLER
             ONLAR
             YUZLER
             BINLER

             TMP1
             TMP2
             TMP3

             ENCPOS
ENDC

CLRF         PORTD
CLRF         PORTB
CLRF         PORTA
BANKSEL     TRISB

MOVLW       h'06'
MOVWF       ADCON1

CLRF        TRISB
CLRF        TRISD

MOVLW       h'FF'

```

```

MOVWF    TRISA

BANKSEL  PORTB

CLRF     SAYAC1
CLRF     SAYAC2
CLRF     BIRLER
CLRF     ONLAR
CLRF     YUZLER
CLRF     BINLER
CLRF     ENCPOS

MOVF     PORTA, W
ANDLW   b'00000011'
MOVWF   TMP1

BASLA
MOVF     PORTA, W
ANDLW   b'00000011'
MOVWF   TMP2
MOVWF   TMP3
XORWF   TMP1, W
BTFSC   STATUS, Z
GOTO    BASLA_1

MOVF     TMP1, W
ANDLW   h'01'
MOVWF   TMP1

BCF     STATUS, C
RRF     TMP2, W
ANDLW   h'01'
XORWF   TMP1, W
BTFSS   STATUS, Z
GOTO    EARTIR
GOTO    EAZALT

EARTIR
MOVF     ENCPOS, W
XORLW   h'FF'
BTFSS   STATUS, Z
INCF   ENCPOS, F
GOTO    BASLA_1

EAZALT
MOVF     ENCPOS, W
XORLW   h'00'
BTFSS   STATUS, Z
DECF   ENCPOS, F

BASLA_1

```

```
MOVF      TMP3, W
MOVWF     TMP1

MOVF      ENCPOS, W
MOVWF     TMP3

CLRF      BIRLER
CLRF      ONLAR
CLRF      YUZLER
CLRF      BINLER
```

BASLA_2

```
MOVF      ENCPOS, W
XORLW    h'00'
BTFSC    STATUS, Z
GOTO     DEVAM

CALL     SAYIART
DECFSZ   TMP3, F
GOTO     $-2
```

DEVAM

```
CLRF      PORTB
MOVLW    h'01'
MOVWF     PORTB
MOVF      BIRLER, W
MOVWF     PORTD
CALL     GECIKME
CLRF      PORTB
MOVLW    h'02'
MOVWF     PORTB
MOVF      ONLAR, W
MOVWF     PORTD
CALL     GECIKME
CLRF      PORTB
MOVLW    h'04'
MOVWF     PORTB
MOVF      YUZLER, W
MOVWF     PORTD
CALL     GECIKME
CLRF      PORTB
MOVLW    h'08'
MOVWF     PORTB
MOVF      BINLER, W
MOVWF     PORTD
CALL     GECIKME
GOTO     BASLA
```

SAYIART

```
INCF     BIRLER, W
ANDLW    H'0F'
```

```

MOVWF    BIRLER

MOVF     BIRLER, W
SUBLW   h'0A'
BTFSC   STATUS, Z
CALL    ONLARART

MOVF     ONLAR, W
SUBLW   h'0A'
BTFSC   STATUS, Z
CALL    YUZLERART

MOVF     YUZLER, W
SUBLW   h'0A'
BTFSC   STATUS, Z
CALL    BINLERART

RETURN

ONLARART
CLRF    BIRLER
INCF    ONLAR, F
RETURN

YUZLERART
INCF    YUZLER, W
ANDLW  h'0F'
MOVWF   YUZLER
CLRF    BIRLER
CLRF    ONLAR
RETURN

BINLERART
INCF    BINLER, W
ANDLW  h'0F'
MOVWF   BINLER
CLRF    BIRLER
CLRF    ONLAR
CLRF    YUZLER
RETURN

GECIKME
MOVLW  0x1A
MOVWF  SAYAC1
MOVLW  0x02
MOVWF  SAYAC2
GECIKME_0
DECFSZ SAYAC1, F
GOTO   $+2
DECFSZ SAYAC2, F
GOTO   GECIKME_0

```

```
GOTO      $+1
RETURN
END
```

Örnek 2 – Analog Giriş Uygulaması

Bu örnek, aynı anda tek bir analog girişin sonucunu göstermek üzere hazırlanmıştır. Analog girişin farklı kanallarının ölçümünü yaparak göstermek için açıklama simgesi ile derleyicinin görmemesi sağlanan satırlar incelenmeli, ilgili satırların simgeleri silinerek diğerleri, açıklama simgesi ile açıklama haline getirilmelidir.

```
LIST      P=16F877A
INCLUDE   "P16F877A.INC"

__CONFIG  _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

CBLOCK    h'20'
          SAYAC1
          SAYAC2

          BIRLER
          ONLAR
          YUZLER
          BINLER

          ANRESH
          ANRESL

          TMP
ENDC

CLRF      PORTD
CLRF      PORTB
CLRF      PORTA
BANKSEL   TRISB

CLRF      ADCON1

CLRF      TRISB
CLRF      TRISD

MOVLW    h'FF'
MOVWF    TRISA

BANKSEL   PORTB

CLRF      TMP
CLRF      SAYAC1
CLRF      SAYAC2
```

```
CLRF      BIRLER
CLRF      ONLAR
CLRF      YUZLER
CLRF      BINLER
```

BASLA

```
; MOVLW      B'01000001' ;Fosc/8 [7-6], A/D ch0 [5-3], a/d on [0]
; MOVLW      B'01000001' ; KANAL 0
MOVLW     B'01001001' ; KANAL 1
; MOVLW      B'01010001' ; KANAL 2
; MOVLW      B'01011001' ; KANAL 3
; MOVLW      B'01100001' ; KANAL 4
MOVWF     ADCON0
```

```
BSF      ADCON0,GO ;Start A/D conversion
```

BEKLE

```
BTFSC    ADCON0,GO ;Wait for conversion to complete
GOTO     BEKLE
```

```
MOVF     ADRESH, W
;RLF     ADRESH, W ; KANAL 2 için
CALL     CARP100
RRF      ANRESH,W
ADDLW    h'01'
MOVWF    TMP
```

```
CLRF     BIRLER
CLRF     ONLAR
CLRF     YUZLER
CLRF     BINLER
```

```
CALL     SAYIART
DECFSZ   TMP, F
GOTO     $-2
```

```
CLRF     PORTB
MOVLW    h'01'
MOVWF    PORTB
MOVF     BIRLER, W
MOVWF    PORTD
CALL     GECIKME
CLRF     PORTB
MOVLW    h'02'
MOVWF    PORTB
MOVF     ONLAR, W
MOVWF    PORTD
BSF      PORTD, 4
CALL     GECIKME
CLRF     PORTB
MOVLW    h'04'
```

```
MOVWF    PORTB
MOV      YUZLER, W
MOVWF    PORTD
BCF      PORTD, 4
CALL     GECIKME
CLRF     PORTB
MOVLW   h'08'
MOVWF    PORTB
MOV      BINLER, W
MOVWF    PORTD
CALL     GECIKME
GOTO     BASLA
```

SAYIART

```
INCF     BIRLER, W
ANDLW    h'0F'
MOVWF    BIRLER
```

```
MOV      BIRLER, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     ONLARART
```

```
MOV      ONLAR, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     YUZLERART
```

```
MOV      YUZLER, W
SUBLW    h'0A'
BTFSC    STATUS, Z
CALL     BINLERART
```

```
RETURN
```

ONLARART

```
CLRF     BIRLER
INCF     ONLAR, F
RETURN
```

YUZLERART

```
INCF     YUZLER, W
ANDLW    h'0F'
MOVWF    YUZLER
CLRF     BIRLER
CLRF     ONLAR
RETURN
```

BINLERART

```
INCF     BINLER, W
ANDLW    h'0F'
```

```

MOVWF    BINLER
CLRF     BIRLER
CLRF     ONLAR
CLRF     YUZLER
RETURN

GECIKME
    MOVLW    0x1A
    MOVWF    SAYAC1
    MOVLW    0x02
    MOVWF    SAYAC2
GECIKME_0
    DECFSZ   SAYAC1, F
    GOTO     $+2
    DECFSZ   SAYAC2, F
    GOTO     GECIKME_0

    GOTO     $+1
    RETURN

CARP100:
    MOVWF    ANRESL
    CLRF     ANRESH
    CALL     CARP10
CARP10:
    BCF     STATUS, C
    RLF     ANRESL, F
    RLF     ANRESH, F
    MOVF    ANRESH, W
    MOVWF   TMP
    MOVF    ANRESL, W
    BCF     STATUS, C
    RLF     ANRESL, F
    RLF     ANRESH, F
    BCF     STATUS, C
    RLF     ANRESL, F
    RLF     ANRESH, F
    ADDWF   ANRESL, F
    MOVF    TMP, W
    BTFSC   STATUS, C
    ADDLW   1
    ADDWF   ANRESH, F
    RETURN
    END

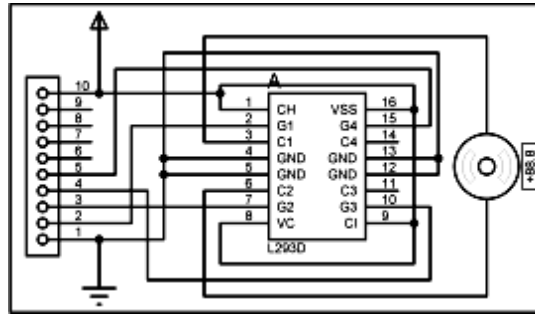
```

ADC İLE DC MOTOR VE STEP MOTOR KONTROLÜ

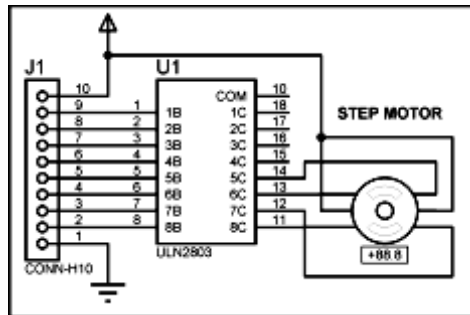
Amaç: Bu deneyde, M-PIC deney setinde yer alan adc ve rotary encoder modüllerine ek olarak dc motor modülü ve step motor modülü kullanılarak, rotary encoder aracılığı ile dc motorun hız kontrolü yapılacak, yine rotary encoder ile step motor döndürülecek ve adc modülünde AN0 ve AN3 kanalları karşılaştırılarak, karşılaştırılma durumuna göre step motor dönüşü sağlanacaktır.

Açıklayıcı Bilgi

Bu deneyde, M-PIC deney setinde rotary encoder ve analog giriş modüllerine ek olarak dc motor modülü ve step motor modülünü kullanarak motor kontrolü yapılacaktır. Bu deney süresinde PIC16F628A kullanılacaktır. Bundan dolayı analog girişler değer olarak ölçülmeyecek, sadece iki analog değer karşılaştırması olarak kullanılacaktır. DC motor modülünün ve step motor modülünün devre şeması şekillerde görülmektedir.



Şekil 18 - DC Motor Modülü Devre Şeması



Şekil 19 - Step Motor Modülü Devre Şeması

Örnek 1 – Rotary Encoder ile DC Motor Hız Kontrolü

```
LIST          P=16F628A
INCLUDE      "P16F628A.INC"
```

```
__CONFIG    _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF
```

```

ORG          h'0000'

CBLOCK      h'20'
            SAYAC1
            SAYAC2

            TMP1
            TMP2
            TMP3
            TMP4
            TMP5

            ENCPOS
            PERLO
            PERHI

            SAKLA_W
            SAKLA_S
ENDC

CLRF        PORTB
CLRF        PORTA
BANKSEL     TRISB

CLRF        TRISB

MOVLW      h'FF'
MOVWF      TRISA

BANKSEL     PORTB

MOVLW      h'07'
MOVWF      CMCON

CLRF        SAYAC1
CLRF        SAYAC2

CLRF        ENCPOS
CLRF        PERLO
CLRF        PERHI

MOVF        PORTA, W
ANDLW      b'00000011'
MOVWF      TMP1

BSF        PORTB, 0

BASLA
MOVF        PORTA, W
ANDLW      b'00000011'

```

```

MOVWF    TMP2
MOVWF    TMP3
XORWF    TMP1, W
BTFSC    STATUS, Z
GOTO     BASLA_1

MOVF     TMP1, W
ANDLW    h'01'
MOVWF    TMP1

BCF      STATUS, C
RRF      TMP2, W
ANDLW    h'01'
XORWF    TMP1, W
BTFSS    STATUS, Z
GOTO     EARTIR
GOTO     EAZALT

EARTIR
MOVF     ENCPOS, W
XORLW    H'FF'
BTFSS    STATUS, Z
INCF     ENCPOS, F
GOTO     BASLA_1

EAZALT
MOVF     ENCPOS, W
XORLW    H'00'
BTFSS    STATUS, Z
DECF     ENCPOS, F

BASLA_1
MOVF     TMP3, W
MOVWF    TMP1

MOVF     ENCPOS, W
MOVWF    PERLO

SUBLW    H'FF'
MOVWF    PERHI

BCF      PORTB, 1
MOVF     PERLO, W
XORLW    h'00'
BTFSC    STATUS, Z
GOTO     BASLA_2
MOVF     PERLO, W
CALL     GECIKME

BASLA_2
BSF      PORTB, 1

```

```

MOVF      PERHI, W
XORLW    h'00'
BTFSC    STATUS, Z
GOTO     BASLA
MOVF      PERHI, W
CALL     GECIKME

GOTO     BASLA

GECIKME
MOVWF    SAYAC1
GECIKME_0
DECFSZ   SAYAC1, F
GOTO     GECIKME_0

RETURN

END

```

Örnek 2 – Rotart Encoder ile Step Motor Kontrolü

```

LIST      P=16F628A
INCLUDE   "P16F628A.INC"

__CONFIG  _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG       h'0000'

CBLOCK    h'20'
          SAYAC1
          SAYAC2

          TMP1
          TMP2
          TMP3
          TMP4
          TMP5

          ENCPOS

          MOTORPOS
ENDC

CLRF      PORTA
BANKSEL   TRISB

CLRF      TRISB

MOVLW    h'FF'

```

```

MOVWF    TRISA

BANKSEL  PORTB

MOVLW   h'07'
MOVWF   CMCON

CLRF    SAYAC1
CLRF    SAYAC2
CLRF    MOTORPOS
CLRF    ENCPOS

MOVF    PORTA, W
ANDLW   b'00000011'
MOVWF   TMP1

BASLA
MOVLW   H'37'
CALL    GECIKME
MOVF    PORTA, W
ANDLW   b'00000011'
MOVWF   TMP2
MOVWF   TMP3
XORWF   TMP1, W
BTFS    STATUS, Z
GOTO    BASLA_1

MOVLW   h'01'
ANDWF   TMP1, F

BCF     STATUS, C
RRF     TMP2, W
ANDLW   h'01'
XORWF   TMP1, W
BTFS    STATUS, Z
GOTO    SOLADON
GOTO    SAGADON

SOLADON
INCF    MOTORPOS, F
MOVF    MOTORPOS, W
ANDLW   b'00000111'
CALL    ADIMTBL
ANDLW   b'00001111'
MOVWF   PORTB
GOTO    BASLA_1

SAGADON
DECF    MOTORPOS, F
MOVF    MOTORPOS, W
ANDLW   b'00000111'

```

```

CALL      ADIMTBL
ANDLW    b'00001111'
MOVWF    PORTB
GOTO     BASLA_1

BASLA_1
MOVF     TMP3, W
MOVWF    TMP1
GOTO     BASLA

ADIMTBL
ADDWF    PCL, F
RETLW    b'0001'
RETLW    b'1001'
RETLW    b'1000'
RETLW    b'1010'
RETLW    b'0010'
RETLW    b'0110'
RETLW    b'0100'
RETLW    b'0101'

GECIKME
MOVWF    SAYAC1
GECIKME_0
DECFSZ   SAYAC1, F
GOTO     GECIKME_0
RETURN

END

```

Örnek 3 – AN0 ve AN3 Gerilimlerinin Karşılaştırılması ile Step Motor Kontrolü

```

LIST      P=16F628A
INCLUDE   "P16F628A.INC"

__CONFIG  _INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_ON & _MCLRE_OFF &
_BODEN_OFF & _LVP_OFF & _CP_OFF

ORG       h'0000'

CBLOCK    h'20'
          SAYAC1
          SAYAC2

          MOTORPOS

          SAKLA_W
          SAKLA_S

ENDC

```

```

        CLRF        PORTA
        CLRF        PORTB

        MOVLW      B'00011100'
        MOVWF      CMCON
        BANKSEL    TRISB
        CLRF        TRISB
        BANKSEL    PORTB

BASLA
        BTFSC      CMCON, 6
        GOTO       BASLA

        INCF        MOTORPOS, F
        MOVF        MOTORPOS, W
        ANDLW      b'00000111'
        CALL        ADIMTBL
        ANDLW      b'00001111'
        MOVWF      PORTB

        MOVLW      D'13'
        MOVWF      SAYAC1
        MOVLW      D'251'
        MOVWF      SAYAC2

LOOP
        DECFSZ     SAYAC2, 1
        GOTO       LOOP
        DECFSZ     SAYAC1, 1
        GOTO       LOOP

        GOTO       BASLA

ADIMTBL
        ADDWF      PCL, F
        RETLW      b'0001'
        RETLW      b'1001'
        RETLW      b'1000'
        RETLW      b'1010'
        RETLW      b'0010'
        RETLW      b'0110'
        RETLW      b'0100'
        RETLW      b'0101'

GECIKME
        MOVWF      SAYAC1
GECIKME_0
        DECFSZ     SAYAC1, F
        GOTO       GECIKME_0
        RETURN
        END

```

Deneyin Yapılışı:

1. DC motorun, sonsuz bir döngü halinde 1'er saniye arayla ve 1'er saniye süreyle çalışıp durmasını sağlayan programı yazınız.
2. DC motorun 2 saniye sağa, 1 saniye sola dönmesini sağlayan programı yazınız.
3. Step motorun 10 adım ileri, 5 adım geri dönmesini sağlayan programı yazınız.
4. Step motorun, fibonacci sayı dizisine göre aşağıda örneği verildiği şekilde dönmesini sağlayacak programı yazınız.
 - a. 1 adım ileri
 - b. 1 adım geri
 - c. 2 adım ileri
 - d. 3 adım geri
 - e. 5 adım ileri
 - f. 8 adım geri
 - g. 13 adım ileri
 - h. 21 adım geri
 - i. ...
5. DC motorun, bir önceki soruda sorulduğu şekilde, fibonacci serisine göre sürelerle dönmesini sağlayan programı yazınız.
 - a. 1 saniye ileri
 - b. 1 saniye geri
 - c. 2 saniye ileri
 - d. 3 saniye geri
 - e. 5 saniye ileri
 - f. 8 saniye geri
 - g. 13 saniye ileri
 - h. 21 saniye geri
 - i. ...
6. Rotary encoder ile step motorun hızının kontrolünü sağlayacak programı yazınız.

TUŞ TAKIMI VE LCD UYGULAMALARI

Amaç: Bu deneyde, M-PIC deney setinde yer alan tuş takımı modülü, 7 segment display modülü ve lcd modülü kullanarak 7 segment ve lcd üzerinde metin ve rakam görüntüleme uygulamaları yapılacaktır.

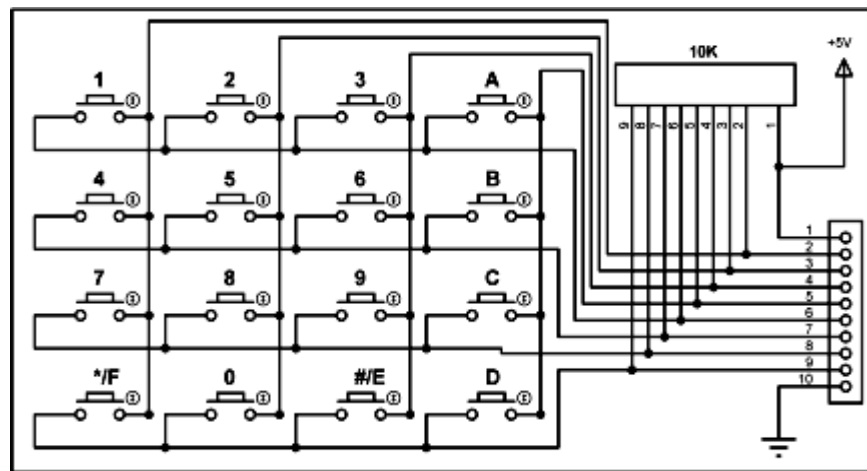
Açıklayıcı Bilgi

M-PIC deney setinde yer alan tuş takımı modülü üzerinde 16 adet buton bulunmaktadır. Butonların uçları 5V'a çekilmiştir. PIC ile hangi butona basıldığını anlamak için satır ve sütun taraması yapılmalıdır. Bu uygulamada sütunlar sırasıyla 0V'a çekilerek o sütundaki butonlardan hangisine basıldığı, ilgili sütundaki butonların lojik seviyelerine bakılarak karar verilir. Tuş takımının devre şeması Şekil 20'de görülmektedir.

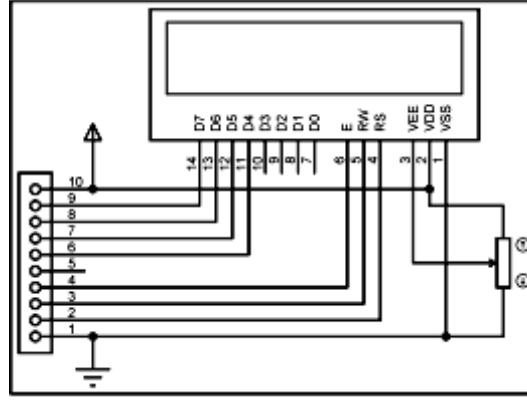
Uygulamada kullanılacak olan LCD ekran, 2 satır ve 8 sütuna sahiptir. Modül, 4-bit kullanım için ayarlandığından dolayı uygulama süresince kodlar, 4-bit olarak sürecek şekilde yazılacaktır.

8-bit sürme: LCD'ye, verilerin ve komutların 8-bit olarak, bir seferde gönderilme işlemidir.

4-bit sürme: LCD'ye, verilerin ve komutların 4-bit olarak 2 seferde gönderilme işlemidir.



Şekil 20 - Tuş Takımı Modülü Devre Şeması



Şekil 21 - 2x8 LCD Modülü Devre Şeması

Örnek 1 – 2x8 LCD Kullanarak Metin Görüntüleme

```

LIST          P=16F877A
INCLUDE      "P16F877A.INC"
__CONFIG    _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

CBLOCK      0x20
            LCD_DATA
            LCD_TMP0
            LCD_TMP1
            LCD_LINE
            LCD_POS

            DELAY_S_DATA
            DELAY_DATA
ENDC

ORG         0
PAGESEL    ANA_PROGRAM
GOTO       ANA_PROGRAM

ORG         4
GOTO       KESME

MESAJLAR
ADDWF      PCL, F
MSG0       DT    "LCD DEMO", 0
MSG1       DT    "EEMERSIN", 0

INCLUDE    "lcd_drive.inc"

KESME
RETFIE

```

```

ANA_PROGRAM
    CLRF          PORTC
    BANKSEL      TRISC
    MOVLW        H'F0'
    MOVWF        TRISC
    BANKSEL      PORTC

    CALL         LCD_INIT
ANA_J0:
    CALL         LCD_CLEAR
    CALL         LCD_CURSOROFF

    LCD_LOCATE   0, 0
    MOVLW        MSG0-6
    CALL         LCD_SENDMESSAGE
    LCD_LOCATE   1, 0
    MOVLW        MSG1-6
    CALL         LCD_SENDMESSAGE

    GOTO         ANA_J0

    END

```

Örnek 2 – Tuş Takımında Basılan Tuşu LCD İle Görüntüleme

```

LIST          P=16F877A
INCLUDE       "P16F877A.INC"
__CONFIG     _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

CBLOCK       0x20
    LCD_DATA
    LCD_TMP0
    LCD_TMP1
    LCD_LINE
    LCD_POS

    DELAY_S_DATA
    DELAY_DATA

    BUTON
    ADET
    VERI
    BUTONFLAG
ENDC

ORG          0
PAGESEL     ANA_PROGRAM
GOTO        ANA_PROGRAM

```

```

        ORG          4
        GOTO        KESME

MESAJLAR
        ADDWF       PCL, F
        MSG0        DT    "LCD DEMO", 0
        MSG1        DT    "EEMERSIN", 0

        INCLUDE     "lcd_drive.inc"

KESME
        RETFIE

ANA_PROGRAM

        CLRF        PORTC
        BANKSEL     TRISC
        MOVLW       H'F0'
        MOVWF       TRISC
        BANKSEL     PORTC

        CLRF        BUTONFLAG

        CALL        LCD_INIT
ANA_J0:
        CALL        LCD_CLEAR
        CALL        LCD_CURSOROFF

        LCD_LOCATE  0, 0
        MOVLW       MSG0-6
        CALL        LCD_SENDMESSAGE
        LCD_LOCATE  1, 0
        MOVLW       MSG1-6
        CALL        LCD_SENDMESSAGE

        MOVLW       .1
        CALL        DELAY_S

        CALL        LCD_CLEAR
ANA_J1:
        CALL        BUTONKONTROL
        BTFSC       BUTONFLAG, 0
        GOTO        VERIGONDER
        GOTO        DEVAM

VERIGONDER
        MOVWF       VERI
        CALL        LCD_SENDCHAR
        INCF        ADET, F
        MOVF        ADET, W
        SUBLW       D'08'

```

```

    BTFSC     STATUS, Z
    GOTO      IKINCISATIR

    MOVF      ADET, W
    SUBLW    D'17'
    BTFSC     STATUS, Z
    GOTO      TEMIZLE
    GOTO      DEVAM

IKINCISATIR
    LCD_LOCATE 1, 0
    GOTO      DEVAM

TEMIZLE
    CALL      LCD_CLEAR
    LCD_LOCATE 0, 0
    MOVF      VERI, W
    CALL      LCD_SENDCHAR
    MOVLW    H'01'
    MOVWF    ADET

DEVAM:
    BCF       BUTONFLAG, 0
    MOVLW    .200
    CALL     DELAY_MS
    GOTO     ANA_J1

BUTONKONTROL
    MOVLW    B'00001110'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW   H'F0'
    MOVWF    BUTON

    BTFSS   BUTON, 4
    GOTO    BAS1
    BTFSS   BUTON, 5
    GOTO    BAS4
    BTFSS   BUTON, 6
    GOTO    BAS7
    BTFSS   BUTON, 7
    GOTO    BASF

    MOVLW    B'00001101'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW   H'F0'
    MOVWF    BUTON

```

```

    BTFSS    BUTON, 4
    GOTO     BAS2
    BTFSS    BUTON, 5
    GOTO     BAS5
    BTFSS    BUTON, 6
    GOTO     BAS8
    BTFSS    BUTON, 7
    GOTO     BAS0

    MOVLW    B'00001011'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS    BUTON, 4
    GOTO     BAS3
    BTFSS    BUTON, 5
    GOTO     BAS6
    BTFSS    BUTON, 6
    GOTO     BAS9
    BTFSS    BUTON, 7
    GOTO     BASE

    MOVLW    B'00000111'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS    BUTON, 4
    GOTO     BASA
    BTFSS    BUTON, 5
    GOTO     BASB
    BTFSS    BUTON, 6
    GOTO     BASC
    BTFSS    BUTON, 7
    GOTO     BASD
    RETURN

BAS0
    BSF      BUTONFLAG, 0
    RETLW    '0'

BAS1
    BSF      BUTONFLAG, 0
    RETLW    '1'

BAS2
    BSF      BUTONFLAG, 0
    RETLW    '2'

```

```
BAS3
    BSF        BUTONFLAG, 0
    RETLW     '3'
BAS4
    BSF        BUTONFLAG, 0
    RETLW     '4'
BAS5
    BSF        BUTONFLAG, 0
    RETLW     '5'
BAS6
    BSF        BUTONFLAG, 0
    RETLW     '6'
BAS7
    BSF        BUTONFLAG, 0
    RETLW     '7'
BAS8
    BSF        BUTONFLAG, 0
    RETLW     '8'
BAS9
    BSF        BUTONFLAG, 0
    RETLW     '9'
BASA
    BSF        BUTONFLAG, 0
    RETLW     'A'
BASB
    BSF        BUTONFLAG, 0
    RETLW     'B'
BASC
    BSF        BUTONFLAG, 0
    RETLW     'C'
BASD
    BSF        BUTONFLAG, 0
    RETLW     'D'
BASE
    BSF        BUTONFLAG, 0
    RETLW     'E'
BASF
    BSF        BUTONFLAG, 0
    RETLW     'F'
END
```

Not: Include dosyası deney sırasında bilgisayarlarda hazır bulunacaktır.

Örnek 3 – Tuş Takımından Basılan Tuşu 7 Segment İle Gösterme

```
LIST        P=16F877A
INCLUDE     "P16F877A.INC"

__CONFIG   _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF
```

```

ORG          h'0000'

SAYAC1      EQU   h'20'
SAYAC2      EQU   h'21'
SAYAC3      EQU   h'22'

BUTON EQU    h'23'
BUTONFLAG   EQU   h'24'

CLRF        PORTB
CLRF        PORTC
BANKSEL     TRISB
CLRF        TRISB
MOVLW      H'F0'
MOVWF      TRISC
BANKSEL     PORTB

CLRF        BUTONFLAG

CLRF        SAYAC1
CLRF        SAYAC2
CLRF        SAYAC3

BASLA
CALL        BUTONKONTROL
BTFSC      BUTONFLAG, 0
GOTO       VERIGONDER
GOTO       DEVAM

VERIGONDER
CALL        SEG7TABLE
MOVWF     PORTB

DEVAM
BCF        BUTONFLAG, 0
CALL       GECIKME
GOTO       BASLA

SEG7TABLE
ADDWF     PCL, F
RETLW    h'3F'
RETLW    h'06'
RETLW    h'5B'
RETLW    h'4F'
RETLW    h'66'
RETLW    h'6D'
RETLW    h'7D'
RETLW    h'07'
RETLW    h'7F'
RETLW    h'6F'

```

```

RETLW    h'77'
RETLW    h'7C'
RETLW    h'39'
RETLW    h'5E'
RETLW    h'79'
RETLW    h'71'

GECIKME
    MOVLW    0x66
    MOVWF    SAYAC1
GECIKME_0
    DECFSZ   SAYAC1, f
    GOTO     GECIKME_0
    RETURN

BUTONKONTROL
    MOVLW    B'00001110'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS   BUTON, 4
    GOTO    BASF
    BTFSS   BUTON, 5
    GOTO    BAS7
    BTFSS   BUTON, 6
    GOTO    BAS4
    BTFSS   BUTON, 7
    GOTO    BAS1

    MOVLW    B'00001101'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS   BUTON, 4
    GOTO    BAS0
    BTFSS   BUTON, 5
    GOTO    BAS8
    BTFSS   BUTON, 6
    GOTO    BAS5
    BTFSS   BUTON, 7
    GOTO    BAS2

    MOVLW    B'00001011'
    MOVWF    PORTC

```

```

        MOVF      PORTC, W
        ANDLW    H'F0'
        MOVWF    BUTON

        BTFSS   BUTON, 4
        GOTO    BASE
        BTFSS   BUTON, 5
        GOTO    BAS9
        BTFSS   BUTON, 6
        GOTO    BAS6
        BTFSS   BUTON, 7
        GOTO    BAS3

        MOVLW   B'00000111'
        MOVWF   PORTC

        MOVF      PORTC, W
        ANDLW    H'F0'
        MOVWF    BUTON

        BTFSS   BUTON, 4
        GOTO    BASD
        BTFSS   BUTON, 5
        GOTO    BASC
        BTFSS   BUTON, 6
        GOTO    BASB
        BTFSS   BUTON, 7
        GOTO    BASA
        RETURN

BAS0
        BSF      BUTONFLAG, 0
        RETLW   D'0'

BAS1
        BSF      BUTONFLAG, 0
        RETLW   D'1'

BAS2
        BSF      BUTONFLAG, 0
        RETLW   D'2'

BAS3
        BSF      BUTONFLAG, 0
        RETLW   D'3'

BAS4
        BSF      BUTONFLAG, 0
        RETLW   D'4'

BAS5
        BSF      BUTONFLAG, 0
        RETLW   D'5'

BAS6
        BSF      BUTONFLAG, 0
        RETLW   D'6'

```

```

BAS7
    BSF      BUTONFLAG, 0
    RETLW    D'7'
BAS8
    BSF      BUTONFLAG, 0
    RETLW    D'8'
BAS9
    BSF      BUTONFLAG, 0
    RETLW    D'9'
BASA
    BSF      BUTONFLAG, 0
    RETLW    D'10'
BASB
    BSF      BUTONFLAG, 0
    RETLW    D'11'
BASC
    BSF      BUTONFLAG, 0
    RETLW    D'12'
BASD
    BSF      BUTONFLAG, 0
    RETLW    D'13'
BASE
    BSF      BUTONFLAG, 0
    RETLW    D'14'
BASF
    BSF      BUTONFLAG, 0
    RETLW    D'15'
    END

```

Örnek 4 – Tuş Takımından Basılan İki Tuşun Farkını Alarak 7 Segment İle Gösterme

```

LIST      P=16F877A
INCLUDE   "P16F877A.INC"

__CONFIG  _XT_OSC & _WDT_OFF & _PWRTE_ON & _BODEN_OFF & _LVP_OFF &
_CP_OFF

ORG       h'0000'

CBLOCK    0x20
    SAYAC1
    SAYAC2
    SAYAC3

    BUTON
    BUTONFLAG

    SAYI1
    SAYI2
    TMP

```

SAYIFLAG

ENDC

```

CLRF      PORTB
CLRF      PORTC
BANKSEL   TRISB
CLRF      TRISB
MOVLW    H'F0'
MOVWF    TRISC
BANKSEL   PORTB
BANKSEL   PORTC
    
```

```

CLRF      BUTONFLAG
CLRF      SAYI1
CLRF      SAYI2
CLRF      SAYIFLAG
    
```

```

CLRF      SAYAC1
CLRF      SAYAC2
CLRF      SAYAC3
    
```

BASLA

```

CALL      BUTONKONTROL
BTFSS    BUTONFLAG, 0
GOTO     BASLA
    
```

ILKSAYI

```

MOVWF    SAYI1
CALL     SEG7TABLE
MOVWF    PORTB
CALL     GECIKME
MOVLW   B'01000000' ; -
MOVWF    PORTB
CALL     GECIKME
BCF      BUTONFLAG, 0
    
```

BASLA_1

```

CALL      BUTONKONTROL
BTFSS    BUTONFLAG, 0
GOTO     BASLA_1

MOVWF    SAYI2
CALL     SEG7TABLE
MOVWF    PORTB
CALL     GECIKME
MOVLW   B'01001000' ; =
MOVWF    PORTB
CALL     GECIKME
BCF      BUTONFLAG, 0

CALL     SONUCGOSTER
    
```

```

        GOTO          BASLA

SONUCGOSTER
        MOVF          SAYI2, W
        SUBWF         SAYI1, W
        BTFSS         STATUS, C
        GOTO          NEGATIFGOSTER

POZITIFGOSTER
        MOVF          SAYI2, W
        SUBWF         SAYI1, W
        GOTO          SON

NEGATIFGOSTER
        MOVLW         B'01000000' ; -
        MOVWF         PORTB
        CALL          GECIKME

        MOVF          SAYI1, W
        MOVWF         TMP
        MOVF          SAYI2, W
        MOVWF         SAYI1
        MOVF          TMP, W
        SUBWF         SAYI1, W

SON
        CALL          SEG7TABLE
        MOVWF         PORTB
        CALL          GECIKME
        CALL          GECIKME
        CALL          GECIKME
        MOVLW         H'00'
        MOVWF         PORTB
        RETURN

SEG7TABLE
        ADDWF         PCL, F
        RETLW         h'3F'
        RETLW         h'06'
        RETLW         h'5B'
        RETLW         h'4F'
        RETLW         h'66'
        RETLW         h'6D'
        RETLW         h'7D'
        RETLW         h'07'
        RETLW         h'7F'
        RETLW         h'6F'
        RETLW         h'77'
        RETLW         h'7C'
        RETLW         h'39'
        RETLW         h'5E'

```

```

RETLW      h'79'
RETLW      h'71'

GECIKME
;999990 PERİYOT
MOVLW      0x07
MOVWF      SAYAC1
MOVLW      0x2F
MOVWF      SAYAC2
MOVLW      0x03
MOVWF      SAYAC3
GECIKME_0
DECFSZ     SAYAC1, f
GOTO       $+2
DECFSZ     SAYAC2, f
GOTO       $+2
DECFSZ     SAYAC3, f
GOTO       GECIKME_0

;6 PERİYOT
GOTO       $+1
GOTO       $+1
GOTO       $+1

;4 PERİYOT (CALL İLE BİRLİKTE)
RETURN

BUTONKONTROL
MOVLW      B'00001110'
MOVWF      PORTC

MOVF       PORTC, W
ANDLW     H'FO'
MOVWF     BUTON

BTFSS     BUTON, 4
GOTO     BASF
BTFSS     BUTON, 5
GOTO     BAS7
BTFSS     BUTON, 6
GOTO     BAS4
BTFSS     BUTON, 7
GOTO     BAS1

MOVLW      B'00001101'
MOVWF      PORTC

MOVF       PORTC, W
ANDLW     H'FO'
MOVWF     BUTON

```

```

    BTFSS    BUTON, 4
    GOTO     BAS0
    BTFSS    BUTON, 5
    GOTO     BAS8
    BTFSS    BUTON, 6
    GOTO     BAS5
    BTFSS    BUTON, 7
    GOTO     BAS2

    MOVLW    B'00001011'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS    BUTON, 4
    GOTO     BASE
    BTFSS    BUTON, 5
    GOTO     BAS9
    BTFSS    BUTON, 6
    GOTO     BAS6
    BTFSS    BUTON, 7
    GOTO     BAS3

    MOVLW    B'00000111'
    MOVWF    PORTC

    MOVF     PORTC, W
    ANDLW    H'F0'
    MOVWF    BUTON

    BTFSS    BUTON, 4
    GOTO     BASD
    BTFSS    BUTON, 5
    GOTO     BASC
    BTFSS    BUTON, 6
    GOTO     BASB
    BTFSS    BUTON, 7
    GOTO     BASA
    RETURN

BAS0
    BSF      BUTONFLAG, 0
    RETLW    D'0'

BAS1
    BSF      BUTONFLAG, 0
    RETLW    D'1'

BAS2
    BSF      BUTONFLAG, 0
    RETLW    D'2'

```

```
BAS3
    BSF      BUTONFLAG, 0
    RETLW    D'3'
BAS4
    BSF      BUTONFLAG, 0
    RETLW    D'4'
BAS5
    BSF      BUTONFLAG, 0
    RETLW    D'5'
BAS6
    BSF      BUTONFLAG, 0
    RETLW    D'6'
BAS7
    BSF      BUTONFLAG, 0
    RETLW    D'7'
BAS8
    BSF      BUTONFLAG, 0
    RETLW    D'8'
BAS9
    BSF      BUTONFLAG, 0
    RETLW    D'9'
BASA
    BSF      BUTONFLAG, 0
    RETLW    D'10'
BASB
    BSF      BUTONFLAG, 0
    RETLW    D'11'
BASC
    BSF      BUTONFLAG, 0
    RETLW    D'12'
BASD
    BSF      BUTONFLAG, 0
    RETLW    D'13'
BASE
    BSF      BUTONFLAG, 0
    RETLW    D'14'
BASF
    BSF      BUTONFLAG, 0
    RETLW    D'15'
END
```

Deneyin Yapılışı:

1. Tuş takımından basılan iki tuşun değerinin toplamını LCD ekranda gösteren programı yazınız.
 - a. İlk butona basınca ekranda (örneğin 2'ye basılmış olsun)
2+
görülmelidir.

b. İkinci butona basıldığında (örneğin 4'e basılmış olsun)

$$2+4=6$$

şeklinde görülmelidir.

2. Basılan butonun karesini ve küpünü **hesaplayarak** sonucu LCD'de gösteren programı yazınız.
3. Basılan iki butonun değerini toplayarak sonucu, örnekteki formatta, 7 segment display üzerinde gösteren programı yazınız. Sonuç iki haneli ise önce ilk hanesi, 1 saniye sonra ikinci hanesi görüntülenecektir.
4. Basılan iki butonun değerini çarparak sonucu, örnekteki formatta, 7 segment display üzerinde gösteren programı yazınız. Sonuç iki haneli ise önce ilk hanesi, 1 saniye sonra ikinci hanesi görüntülenecektir.