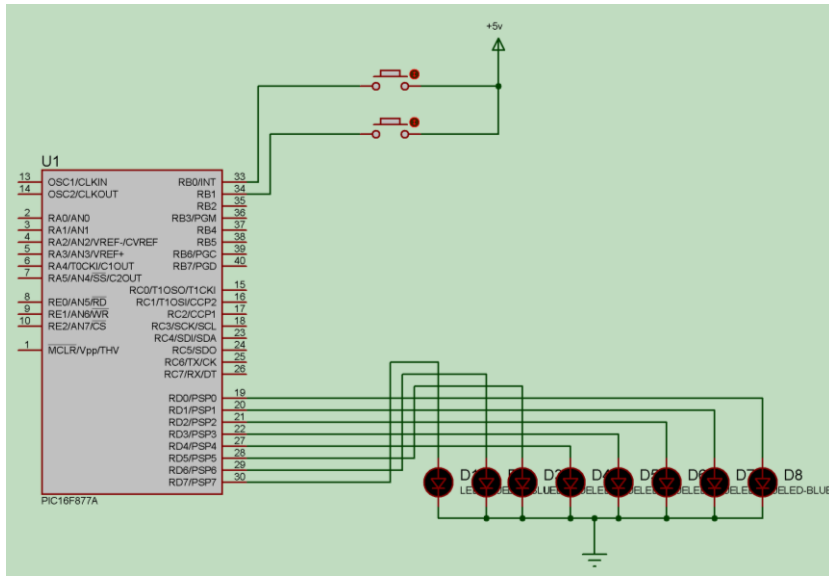


Örnek 1: Sağdan sola ledleri yakmak.

```

program led_uyg1
CONST YAZ AS BYTE[8]=(1,2,4,8,16,32,64,128)
DIM I AS BYTE
main:
TRISD=0
PORTD=0
WHILE TRUE
FOR I=0 TO 7
PORTD=YAZ[I]
DELAY_MS(100)
NEXT I
FOR I=7 TO 0 STEP -1
PORTD=YAZ[I]
DELAY_MS(100)
NEXT I
WEND
end.

```

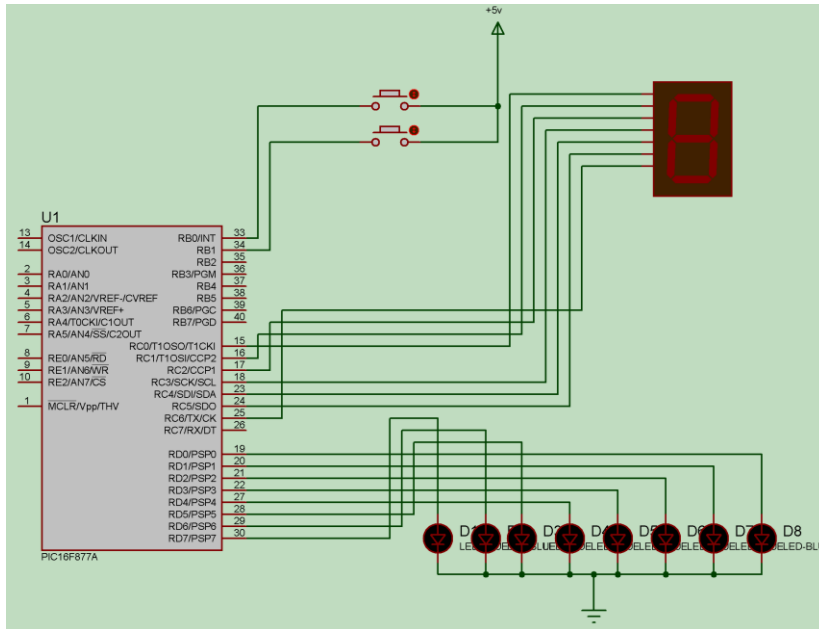


Örnek 2: 1. Butona basıldığında tüm ledleri yakan 2. Butona basıldığında ledleri teker teker söndüren uygulama

```

program button_uyg
CONST YAZ AS BYTE[8]=(1,3,7,15,31,63,127,255)
DIM I AS BYTE
DIM KONTROL AS BYTE
main:
TRISD=0
PORTD=0
TRISB.0=1
TRISB.1=1
DO
IF PORTB.0=1 THEN
KONTROL=1
END IF
IF PORTB.1=1 THEN
KONTROL=2
END IF
IF KONTROL=1 THEN
PORTD=255
END IF
IF KONTROL=2 THEN
FOR I=7 TO 0 STEP -1
PORTD=YAZ[I]
DELAY_MS(100)
NEXT I
END IF
LOOP UNTIL FALSE
end.

```

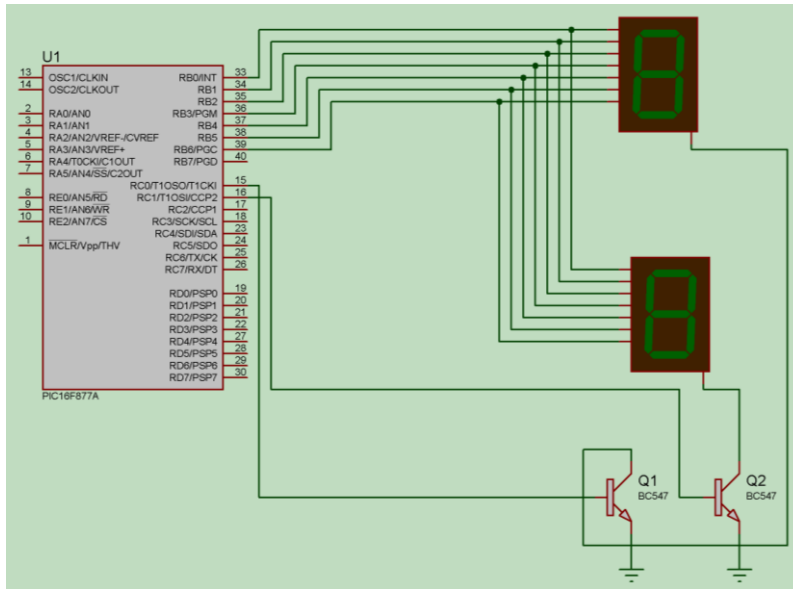


Örnek 3: 1. Butona basıldığında display 0-9 arasındaki sayıları gösterir, 2. Butona basıldığında ledleri teker teker söndüren uygulama

```

program dijital_uyg
CONST YAZ AS BYTE[8]=(1,3,7,15,31,63,127,255)
DIM I AS BYTE
DIM KONTROL AS BYTE
CONST SAYI AS BYTE[10]=(63,6,91,79,102,109,125,7,127,111)
main:
TRISD=0
PORTD=0
TRISB.0=1
TRISB.1=1
TRISC=0
PORTC=0
DO
IF PORTB.0=1 THEN
KONTROL=1
END IF
IF PORTB.1=1 THEN
KONTROL=2
END IF
IF KONTROL=1 THEN
PORTD=255
FOR I=0 TO 9
PORTC= SAYI[I]
DELAY_MS(300)
NEXT I
END IF
IF KONTROL=2 THEN
FOR I=7 TO 0 STEP -1
PORTD=YAZ[I]
DELAY_MS(100)
NEXT I
END IF
LOOP UNTIL FALSE
end.

```



Örnek 4: Sırasıyla displayleri 0-9 arası saydıran uygulama

program Display1

```
CONST YAZ AS BYTE[10]=(63,6,91,79,102,109,125,7,127,103)
```

```
DIM I AS BYTE
```

```
main:
```

```
TRISB=0
```

```
TRISC=%11000000
```

```
WHILE TRUE
```

```
FOR I=0 TO 9
```

```
PORTC=1
```

```
PORTB=YAZ[I]
```

```
DELAY_MS(200)
```

```
NEXT I
```

```
FOR I=0 TO 9
```

```
PORTC=2
```

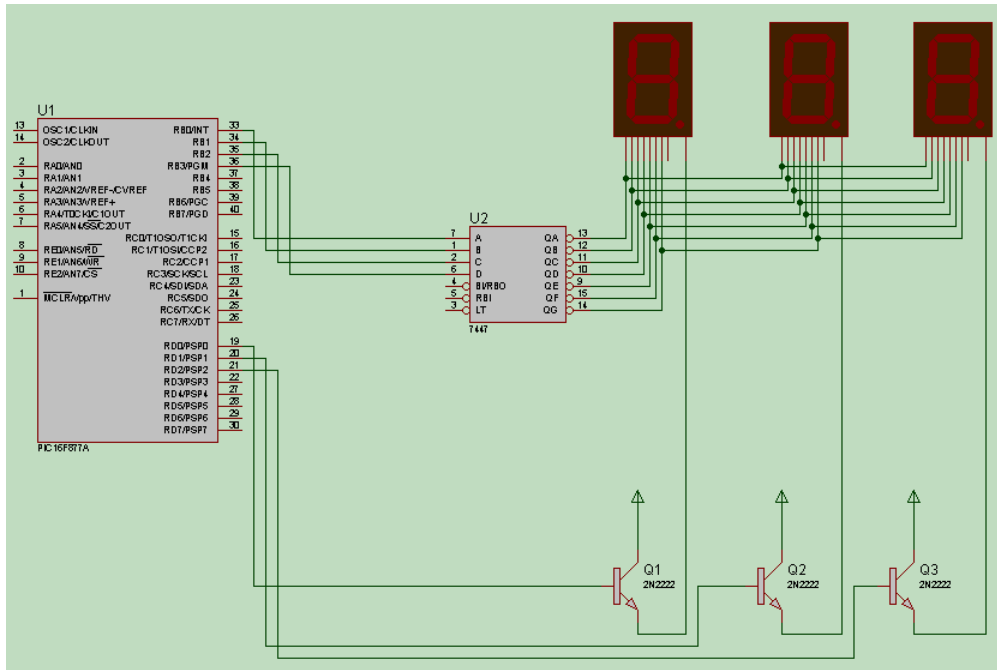
```
PORTB=YAZ[I]
```

```
DELAY_MS(200)
```

```
NEXT I
```

```
WEND
```

```
end.
```

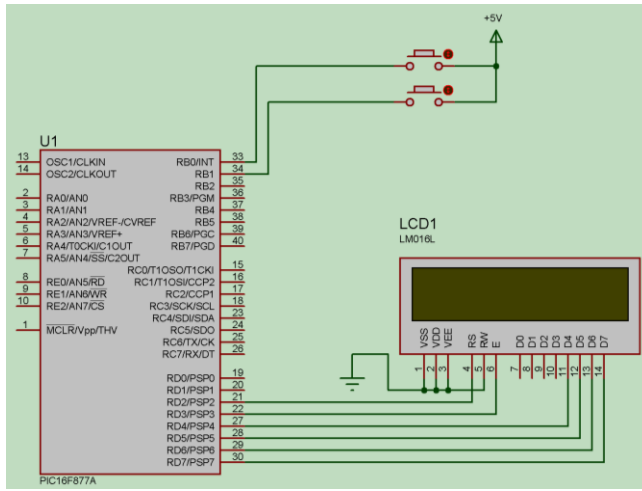


Örnek 5: Displayda 124 yazdıran uygulama

```

program Display2
main:
TRISB=0
TRISD=0
PORTB=0
PORTD=0
WHILE TRUE
PORTD=1
PORTB=0
DELAY_MS(20)
PORTD=2
PORTB=0
DELAY_MS(20)
PORTD=4
PORTB=0
DELAY_MS(20)
WEND
end.

```



Örnek 6: Ekranın farklı satır sütunlarına değer yazdıran uygulama

program lblcd1

dim

LCD_RS as sbit at Rd2_bit

LCD_EN as sbit at Rd3_bit

LCD_D7 as sbit at Rd7_bit

LCD_D6 as sbit at Rd6_bit

LCD_D5 as sbit at Rd5_bit

LCD_D4 as sbit at Rd4_bit

dim

LCD_RS_Direction as sbit at TRISd2_bit

LCD_EN_Direction as sbit at TRISd3_bit

LCD_D7_Direction as sbit at TRISd7_bit

LCD_D6_Direction as sbit at TRISd6_bit

LCD_D5_Direction as sbit at TRISd5_bit

LCD_D4_Direction as sbit at TRISd4_bit

main:

TRISD=0

LCD_INIT()

LCD_CMD(_LCD_CLEAR)

LCD_CMD(_LCD_CURSOR_OFF)

LCD_OUT(1,1, "BILGISAYAR")

LCD_OUT_CP("5")

LCD_CHR(2,7,"X")

LCD_CHR_CP("Y")

end.

Örnek 7: 1. Butona basıldığında 1.satır 5.sütuna “BILGISAYAR” yazdıran, 2.butona basıldığında “ELEKTRONIK” yazdıran uygulama

```
program lblcd2
dim
  LCD_RS as sbit at Rd2_bit
  LCD_EN as sbit at Rd3_bit
  LCD_D7 as sbit at Rd7_bit
  LCD_D6 as sbit at Rd6_bit
  LCD_D5 as sbit at Rd5_bit
  LCD_D4 as sbit at Rd4_bit
dim
  LCD_RS_Direction as sbit at TRISd2_bit
  LCD_EN_Direction as sbit at TRISd3_bit
  LCD_D7_Direction as sbit at TRISd7_bit
  LCD_D6_Direction as sbit at TRISd6_bit
  LCD_D5_Direction as sbit at TRISd5_bit
  LCD_D4_Direction as sbit at TRISd4_bit
main:
  TRISD=0
  LCD_INIT()
  LCD_CMD(_LCD_CLEAR)
  LCD_CMD(_LCD_CURSOR_OFF)
  TRISB.0=1
  TRISB.1=1
  WHILE TRUE
  IF PORTB.0=1 THEN
  LCD_CMD(_LCD_CLEAR)
  LCD_OUT(1,5, "BILGISAYAR")
  END IF
  IF PORTB.1=1 THEN
  LCD_CMD(_LCD_CLEAR)
  LCD_OUT(1,5, "ELEKTRONIK")
  END IF
  WEND
end.
```

Örnek 8: Sağdan sola karakterleri tek tek kaydıran uygulama

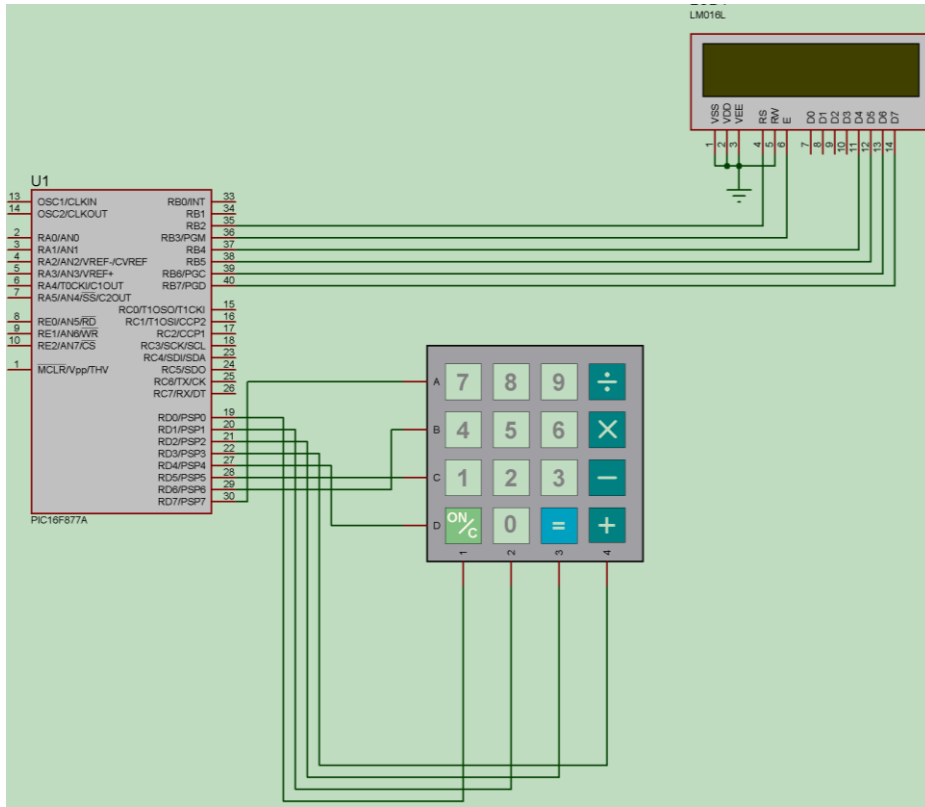
```
program lblcd3
dim
  LCD_RS as sbit at Rd2_bit
  LCD_EN as sbit at Rd3_bit
  LCD_D7 as sbit at Rd7_bit
  LCD_D6 as sbit at Rd6_bit
  LCD_D5 as sbit at Rd5_bit
  LCD_D4 as sbit at Rd4_bit
dim
  LCD_RS_Direction as sbit at TRISd2_bit
  LCD_EN_Direction as sbit at TRISd3_bit
  LCD_D7_Direction as sbit at TRISd7_bit
  LCD_D6_Direction as sbit at TRISd6_bit
  LCD_D5_Direction as sbit at TRISd5_bit
  LCD_D4_Direction as sbit at TRISd4_bit
  DIM A AS CHAR[16]
  DIM B AS CHAR[1]
  DIM I AS BYTE
  DIM J AS BYTE
main:
  TRISD=0
  LCD_INIT()
  LCD_CMD(_LCD_CLEAR)
  LCD_CMD(_LCD_CURSOR_OFF)
  A="MIKRO BILGISAYAR"
  WHILE TRUE
  FOR I=0 TO 15
  FOR J=1 TO 16
  B[0]=A[I]
  LCD_OUT(1,(16-J)+1+I,B)
  LCD_OUT_CP(" ")
  DELAY_MS(40)
  NEXT J
  NEXT I
  NEXT I
  WEND
end.
```


Örnek 9: ekranın etrafına “X” karakterini dolandıran uygulama

```
program lablcd1
dim
  LCD_RS as sbit at RB2_bit
  LCD_EN as sbit at RB3_bit
  LCD_D7 as sbit at RB7_bit
  LCD_D6 as sbit at RB6_bit
  LCD_D5 as sbit at RB5_bit
  LCD_D4 as sbit at RB4_bit
dim
  LCD_RS_Direction as sbit at TRISB2_bit
  LCD_EN_Direction as sbit at TRISB3_bit
  LCD_D7_Direction as sbit at TRISB7_bit
  LCD_D6_Direction as sbit at TRISB6_bit
  LCD_D5_Direction as sbit at TRISB5_bit
  LCD_D4_Direction as sbit at TRISB4_bit
dim satir as byte
dim sutun as byte
main:
  Trisb=0
  Lcd_init()
  lcd_cmd(_lcd_clear)
  lcd_cmd(_lcd_cursor_off)
  satir=1
  sutun=1
  while true
  if satir=1 then
  lcd_chr(satir,sutun, "x")
  delay_ms(50)
  lcd_cmd(_lcd_clear)
  sutun=sutun+1
  if sutun=16 then
  satir=satir+1
  end if
  end if
  if satir=2 then
  lcd_chr(satir,sutun, "x")
  delay_ms(50)
  lcd_cmd(_lcd_clear)
  sutun=sutun-1
  if(satir=2) and (sutun=1) then
  satir=1
  sutun=1
  end if
  end if
  wend
end.
```

Örnek 10: “X” karakterine yukarı aşağıya oynatan uygulama

```
program lablcd2
dim
  LCD_RS as sbit at RB2_bit
  LCD_EN as sbit at RB3_bit
  LCD_D7 as sbit at RB7_bit
  LCD_D6 as sbit at RB6_bit
  LCD_D5 as sbit at RB5_bit
  LCD_D4 as sbit at RB4_bit
dim
  LCD_RS_Direction as sbit at TRISB2_bit
  LCD_EN_Direction as sbit at TRISB3_bit
  LCD_D7_Direction as sbit at TRISB7_bit
  LCD_D6_Direction as sbit at TRISB6_bit
  LCD_D5_Direction as sbit at TRISB5_bit
  LCD_D4_Direction as sbit at TRISB4_bit
dim i as byte
dim j as byte
main:
  Trisb=0
  Lcd_init()
  lcd_cmd(_lcd_clear)
  lcd_cmd(_lcd_cursor_off)
  while true
    for j=1 to 16
      for i=1 to 2
        lcd_chr(i,j, "x")
        delay_ms(100)
        lcd_cmd(_lcd_clear)
      next i
    next j
  wend
end.
```



Örnek 11: tuş takımında basılan tuşu ekrana yazdıran uygulama

```

program lblcd3
dim
LCD_RS as sbit at RB2_bit
LCD_EN as sbit at RB3_bit
LCD_D7 as sbit at RB7_bit
LCD_D6 as sbit at RB6_bit
LCD_D5 as sbit at RB5_bit
LCD_D4 as sbit at RB4_bit
dim
LCD_RS_Direction as sbit at TRISB2_bit
LCD_EN_Direction as sbit at TRISB3_bit
LCD_D7_Direction as sbit at TRISB7_bit
LCD_D6_Direction as sbit at TRISB6_bit
LCD_D5_Direction as sbit at TRISB5_bit
LCD_D4_Direction as sbit at TRISB4_bit
dim KeypadPort as byte at PORTD
dim tus as byte
main:
Trisb=0
Lcd_init()
lcd_cmd(_lcd_clear)
lcd_cmd(_lcd_cursor_off)
Keypad_Init()
while true
tus=0
while(tus=0)
tus=Keypad_Key_Click()
delay_ms(10)
wend
select case tus

```

```
case 1
lcd_cmd(_lcd_clear)
case 2
tus=48
case 3
tus = 61
case 4
tus=43
case 16
tus= 47
end select
lcd_chr_cp(tus)
wend
end.
```