```
;
;               Leilani Electronics
;   program written & tested by William E. Nickles
;           15X35 Blue Dotmatrix Display
;                 rev.1 5/11/2012
;
            list p=16f819
            __config h'3f38'
            radix hex
;
status      equ     0x03
porta       equ     0x05
portb       equ     0x06
intcon      equ     0x0b
option_reg  equ     0x81
trisa       equ     0x85
trisb       equ     0x86
osccon      equ     0x8f
osctune     equ     0x90
adcon1      equ     0x9f
;
countd      equ     0x20
temp0       equ     0x21
count0      equ     0x22
count1      equ     0x23
count2      equ     0x24
;
reg0        equ     0x25
reg1        equ     0x26
reg2        equ     0x27
reg3        equ     0x28
reg4        equ     0x29
reg5        equ     0x2a
reg6        equ     0x2b
;
;   hold registers
dot0        equ     0x2c
dot1        equ     0x2d
dot2        equ     0x2e
dot3        equ     0x2f
dot4        equ     0x30
dot5        equ     0x31
dot6        equ     0x32
dot7        equ     0x33
dot8        equ     0x34
dot9        equ     0x35
dot10       equ     0x36
dot11       equ     0x37
dot12       equ     0x38
dot13       equ     0x39
dot14       equ     0x3a
dot15       equ     0x3b
dot16       equ     0x3c
dot17       equ     0x3d
dot18       equ     0x3e
dot19       equ     0x3f
dot20       equ     0x40
dot21       equ     0x41
dot22       equ     0x42
dot23       equ     0x43
dot24       equ     0x44
dot25       equ     0x45
dot26       equ     0x46
dot27       equ     0x47
dot28       equ     0x48
dot29       equ     0x49
dot30       equ     0x4a
dot31       equ     0x4b
dot32       equ     0x4c
dot33       equ     0x4d
dot34       equ     0x4e
```

```
dot35       equ     0x4f
dot36       equ     0x50
dot37       equ     0x51
dot38       equ     0x52
dot39       equ     0x53
dot40       equ     0x54
dot41       equ     0x55
dot42       equ     0x56
dot43       equ     0x57
dot44       equ     0x58
dot45       equ     0x59
dot46       equ     0x5a
dot47       equ     0x5b
dot48       equ     0x5c
dot49       equ     0x5d
dot50       equ     0x5e
dot51       equ     0x5f
dot52       equ     0x60
dot53       equ     0x61
dot54       equ     0x62
dot55       equ     0x63
dot56       equ     0x64
dot57       equ     0x65
dot58       equ     0x66
dot59       equ     0x67
dot60       equ     0x68
dot61       equ     0x69
dot62       equ     0x6a
dot63       equ     0x6b
dot64       equ     0x6c
dot65       equ     0x6d
dot66       equ     0x6e
dot67       equ     0x6f
dot68       equ     0x70
dot69       equ     0x71
dot70       equ     0x72
dot71       equ     0x73
dot72       equ     0x74
dot73       equ     0x75
dot74       equ     0x76
dot75       equ     0x77
dot76       equ     0x78
dot77       equ     0x79
dot78       equ     0x7a
dot79       equ     0x7b
dot80       equ     0x7c
dot81       equ     0x7d
dot82       equ     0x7e
dot83       equ     0x7f
;
c           equ     0           ; carrier flag
z           equ     2           ; zero flag
rp0         equ     5           ; bank select
;   porta
mclk        equ     0           ; row clock (mic)
moe         equ     1           ; row en/dis (mic)
sdi         equ     2           ; column data(stp)
sclk        equ     3           ; column clock (stp)
le          equ     4           ; column update (stp)
;mclr       equ     5           ; system reset (uC)
str         equ     6           ; row update (mic)
sin         equ     7           ; row data(mic)
;   portb
;int        equ     0           ; system interrupt (uC)
oe          equ     1           ; column en/dis (stp)
;pgc        equ     6           ; icsp clock
;pgd        equ     7           ; icsp data
;
            org     0x00
            goto    start
;
```

```
            org     0x04
            goto    all_off
;
start       clrf    intcon      ; disable interrupts
            clrf    porta       ; clear porta & portb
            clrf    portb
            bsf     status,rp0  ; switch to bank1
            movlw   0x88
            movwf   option_reg
            movlw   0x64        ; option internal osc for 4mhz
            movwf   osccon
            movlw   0x3f        ; tune to max freq
            movwf   osctune
            movlw   0x06        ; disable pic's adc & configure
            movwf   adcon1      ; ports for digital i/o's
            movlw   0x20        ; configure porta i/o's
            movwf   trisa
            movlw   0xc1        ; configure portb i/o's
            movwf   trisb
            bcf     status,rp0  ; switch to bank0
            clrf    porta
            clrf    portb
;
            bcf     intcon,1    ; enable interrupts
            nop
            bsf     intcon,4
            nop
            bsf     intcon,7
;
            bsf     porta,moe   ; disable rows
            nop
            bsf     portb,oe    ; disable columns
            nop
            bcf     porta,le    ; latch/strobe idle low
            nop
            bcf     porta,str
            nop
            bcf     porta,mclk  ; clocks idle low
            nop
            bcf     porta,sclk
            nop
            bcf     porta,sdi   ; data lines idle low
            nop
            bcf     porta,sin
;
            call    wait        ; brief wait...
;
            bcf     porta,moe   ; enable drivers
            nop
            bcf     portb,oe
;
matrix      call    leth        ; populate character positions
            call    pos0        ; "hello matrix" message 1
            call    lete
            call    pos1
            call    letl
            call    pos2
            call    letl
            call    pos3
            call    leto
            call    pos4
            call    smile
            call    pos5
            call    letm
            call    pos6
            call    leta
            call    pos7
            call    lett
            call    pos8
            call    letr
            call    pos9
            call    leti
            call    pos10
            call    letx
            call    pos11
;
            movlw   0xe0
            movwf   countd
disp1       call    display     ; update display
            decfsz  countd,f
            goto    disp1
;
            call    blank       ; populate character positions
            call    pos0        ; "how r u?" message 2
            call    leth
            call    pos1
            call    leto
            call    pos2
            call    letw
            call    pos3
            call    blank
            call    pos4
            call    blank
            call    pos5
            call    blank
            call    pos6
            call    letr
            call    pos7
            call    blank
            call    pos8
            call    letu
            call    pos9
            call    blank
            call    pos10
            call    quest
            call    pos11
;
            movlw   0xe0
            movwf   countd
disp2       call    display     ; update display
            decfsz  countd,f
            goto    disp2
;
            call    smile       ; populate character positions
            call    pos0        ; message 3
            call    blank
            call    pos1
            call    letg
            call    pos2
            call    leto
            call    pos3
            call    blank
            call    pos4
            call    smile
            call    pos5
            call    letd
            call    pos6
            call    letu
            call    pos7
            call    letk
            call    pos8
            call    lete
            call    pos9
            call    exc
            call    pos10
            call    exc
            call    pos11
;
            movlw   0xe0
            movwf   countd
disp3       call    display     ; update display
            decfsz  countd,f
            goto    disp3
;
            call    smile       ; populate character positions
            call    pos0        ; message 4
            call    lets
            call    pos1
            call    lett
            call    pos2
            call    leta
            call    pos3
            call    lety
            call    pos4
            call    blank
            call    pos5
            call    blank
            call    pos6
            call    letc
            call    pos7
            call    leto
            call    pos8
            call    leto
            call    pos9
            call    letl
            call    pos10
            call    exc
            call    pos11
;
            movlw   0xe0
            movwf   countd
disp4       call    display     ; update display
            decfsz  countd,f
            goto    disp4
            goto    matrix      ; refresh display
;
display     call    row15       ; displays pos0-pos11
            movf    dot35,w
            call    xmit6
            movf    dot28,w
            call    xmit6
            movf    dot21,w
            call    xmit6
            movf    dot14,w
            call    xmit6
            movf    dot7,w
            call    xmit6
            movf    dot0,w
            call    xmit6
            call    latch
            call    wait
            call    col_clr
            call    row14
            movf    dot36,w
            call    xmit6
            movf    dot29,w
            call    xmit6
            movf    dot22,w
            call    xmit6
            movf    dot15,w
            call    xmit6
            movf    dot8,w
            call    xmit6
            movf    dot1,w
            call    xmit6
            call    latch
            call    wait
            call    col_clr
            call    row13
            movf    dot37,w
            call    xmit6
            movf    dot30,w
            call    xmit6
```

```
movf    dot23,w
call    xmit6
movf    dot16,w
call    xmit6
movf    dot9,w
call    xmit6
movf    dot2,w
call    xmit6
call    latch
call    wait
call    col_clr
call    row12
movf    dot38,w
call    xmit6
movf    dot31,w
call    xmit6
movf    dot24,w
call    xmit6
movf    dot17,w
call    xmit6
movf    dot10,w
call    xmit6
movf    dot3,w
call    xmit6
call    latch
call    wait
call    col_clr
call    row11
movf    dot39,w
call    xmit6
movf    dot32,w
call    xmit6
movf    dot25,w
call    xmit6
movf    dot18,w
call    xmit6
movf    dot11,w
call    xmit6
movf    dot4,w
call    xmit6
call    latch
call    wait
call    col_clr
call    row10
movf    dot40,w
call    xmit6
movf    dot33,w
call    xmit6
movf    dot26,w
call    xmit6
movf    dot19,w
call    xmit6
movf    dot12,w
call    xmit6
movf    dot5,w
call    xmit6
call    latch
call    wait
call    col_clr
call    row9
movf    dot41,w
call    xmit6
movf    dot34,w
call    xmit6
movf    dot27,w
call    xmit6
movf    dot20,w
call    xmit6
movf    dot13,w
call    xmit6
movf    dot6,w
```

```
        call    xmit6
        call    latch
        call    wait
        call    col_clr
        call    row7
        movf    dot77,w
        call    xmit6
        movf    dot70,w
        call    xmit6
        movf    dot63,w
        call    xmit6
        movf    dot56,w
        call    xmit6
        movf    dot49,w
        call    xmit6
        movf    dot42,w
        call    xmit6
        call    latch
        call    wait
        call    col_clr
        call    row6
        movf    dot78,w
        call    xmit6
        movf    dot71,w
        call    xmit6
        movf    dot64,w
        call    xmit6
        movf    dot57,w
        call    xmit6
        movf    dot50,w
        call    xmit6
        movf    dot43,w
        call    xmit6
        call    latch
        call    wait
        call    col_clr
        call    row5
        movf    dot79,w
        call    xmit6
        movf    dot72,w
        call    xmit6
        movf    dot65,w
        call    xmit6
        movf    dot58,w
        call    xmit6
        movf    dot51,w
        call    xmit6
        movf    dot44,w
        call    xmit6
        call    latch
        call    wait
        call    col_clr
        call    row4
        movf    dot80,w
        call    xmit6
        movf    dot73,w
        call    xmit6
        movf    dot66,w
        call    xmit6
        movf    dot59,w
        call    xmit6
        movf    dot52,w
        call    xmit6
        movf    dot45,w
        call    xmit6
        call    latch
        call    wait
        call    col_clr
        call    row3
        movf    dot81,w
        call    xmit6
```

```
            movf    dot74,w
            call    xmit6
            movf    dot67,w
            call    xmit6
            movf    dot60,w
            call    xmit6
            movf    dot53,w
            call    xmit6
            movf    dot46,w
            call    xmit6
            call    latch
            call    wait
            call    col_clr
            call    row2
            movf    dot82,w
            call    xmit6
            movf    dot75,w
            call    xmit6
            movf    dot68,w
            call    xmit6
            movf    dot61,w
            call    xmit6
            movf    dot54,w
            call    xmit6
            movf    dot47,w
            call    xmit6
            call    latch
            call    wait
            call    col_clr
            call    row1
            movf    dot83,w
            call    xmit6
            movf    dot76,w
            call    xmit6
            movf    dot69,w
            call    xmit6
            movf    dot62,w
            call    xmit6
            movf    dot55,w
            call    xmit6
            movf    dot48,w
            call    xmit6
            call    latch
            call    wait
            call    col_clr
            return
;
; charaters
;
leta        movlw   0x08         ; letter 'a'
            movwf   reg0
            movlw   0x14
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x3e
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
letb        movlw   0x3c         ; letter 'b'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x3c
            movwf   reg6
            return
letc        movlw   0x1c        ; letter 'c'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x20
            movwf   reg3
            movlw   0x20
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
letd        movlw   0x3c        ; letter 'd'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x3c
            movwf   reg6
            return
lete        movlw   0x3e        ; letter 'e'
            movwf   reg0
            movlw   0x20
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x20
            movwf   reg4
            movlw   0x20
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
letf        movlw   0x3e        ; letter 'f'
            movwf   reg0
            movlw   0x20
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x20
            movwf   reg4
            movlw   0x20
            movwf   reg5
            movlw   0x20
            movwf   reg6
            return
letg        movlw   0x1c        ; letter 'g'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x2c
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
leth        movlw   0x22        ; letter 'h'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x3e
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
leti        movlw   0x3e        ; letter 'i'
            movwf   reg0
            movlw   0x08
            movwf   reg1
            movlw   0x08
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
letj        movlw   0x3e        ; letter 'j'
            movwf   reg0
            movlw   0x04
            movwf   reg1
            movlw   0x04
            movwf   reg2
            movlw   0x04
            movwf   reg3
            movlw   0x04
            movwf   reg4
            movlw   0x24
            movwf   reg5
            movlw   0x18
            movwf   reg6
            return
letk        movlw   0x22        ; letter 'k'
            movwf   reg0
            movlw   0x24
            movwf   reg1
            movlw   0x28
            movwf   reg2
            movlw   0x30
            movwf   reg3
            movlw   0x28
            movwf   reg4
            movlw   0x24
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
letl        movlw   0x20        ; letter 'l'
            movwf   reg0
            movlw   0x20
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x20
            movwf   reg3
            movlw   0x20
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
letm        movlw   0x22        ; letter 'm'
            movwf   reg0
            movlw   0x36
            movwf   reg1
            movlw   0x2a
            movwf   reg2
            movlw   0x2a
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
letn        movlw   0x22        ; letter 'n'
            movwf   reg0
            movlw   0x32
            movwf   reg1
            movlw   0x2a
            movwf   reg2
            movlw   0x26
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
leto        movlw   0x1c        ; letter 'o'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
letp        movlw   0x3c        ; letter 'p'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x20
            movwf   reg4
            movlw   0x20
            movwf   reg5
            movlw   0x20
            movwf   reg6
            return
letq        movlw   0x1c        ; letter 'q'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x2a
            movwf   reg4
            movlw   0x24
            movwf   reg5
            movlw   0x1a
            movwf   reg6
            return
letr        movlw   0x3c        ; letter 'r'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x28
            movwf   reg4
            movlw   0x24
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
lets        movlw   0x1c        ; letter 's'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x1c
            movwf   reg3
            movlw   0x02
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
lett        movlw   0x3e        ; letter 't'
            movwf   reg0
            movlw   0x08
            movwf   reg1
            movlw   0x08
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x08
            movwf   reg6
            return
letu        movlw   0x22        ; letter 'u'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x26
            movwf   reg5
            movlw   0x1a
            movwf   reg6
            return
letv        movlw   0x22         ; letter 'v'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x22
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x14
            movwf   reg5
            movlw   0x08
            movwf   reg6
            return
letw        movlw   0x22         ; letter 'w'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x2a
            movwf   reg3
            movlw   0x2a
            movwf   reg4
            movlw   0x36
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
letx        movlw   0x22         ; letter 'x'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x14
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x14
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x22
            movwf   reg6
            return
lety        movlw   0x22         ; letter 'y'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x14
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x08
            movwf   reg6
            return
letz        movlw   0x3e         ; letter 'z'
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x04
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x10
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
exc         movlw   0x14            ; "!"
            movwf   reg0
            movlw   0x14
            movwf   reg1
            movlw   0x14
            movwf   reg2
            movlw   0x14
            movwf   reg3
            movlw   0x14
            movwf   reg4
            movlw   0x00
            movwf   reg5
            movlw   0x14
            movwf   reg6
            return
one         movlw   0x08            ; "1"
            movwf   reg0
            movlw   0x18
            movwf   reg1
            movlw   0x08
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
two         movlw   0x1c            ; "2"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x04
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x10
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x3e
            movwf   reg6
            return
three       movlw   0x1c            ; "3"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x02
            movwf   reg2
            movlw   0x1c
            movwf   reg3
            movlw   0x02
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
four        movlw   0x22        ; "4"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x3e
            movwf   reg3
            movlw   0x02
            movwf   reg4
            movlw   0x02
            movwf   reg5
            movlw   0x02
            movwf   reg6
            return
five        movlw   0x3e        ; "5"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x20
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x02
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
six         movlw   0x04        ; "6"
            movwf   reg0
            movlw   0x08
            movwf   reg1
            movlw   0x10
            movwf   reg2
            movlw   0x3c
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
seven       movlw   0x3e        ; "7"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x02
            movwf   reg2
            movlw   0x04
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x10
            movwf   reg5
            movlw   0x10
            movwf   reg6
            return
eight       movlw   0x1c        ; "8"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x1c
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x22
            movwf   reg5
            movlw   0x1c
            movwf   reg6
            return
nine        movlw   0x1c         ; "9"
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x22
            movwf   reg2
            movlw   0x1e
            movwf   reg3
            movlw   0x04
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x10
            movwf   reg6
            return
slash1      movlw   0x00         ; "/"
            movwf   reg0
            movlw   0x20
            movwf   reg1
            movlw   0x10
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x04
            movwf   reg4
            movlw   0x02
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
slash2      movlw   0x00         ; "\"
            movwf   reg0
            movlw   0x02
            movwf   reg1
            movlw   0x04
            movwf   reg2
            movlw   0x08
            movwf   reg3
            movlw   0x10
            movwf   reg4
            movlw   0x20
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
smile       movlw   0x14         ; ":-)"
            movwf   reg0
            movlw   0x14
            movwf   reg1
            movlw   0x00
            movwf   reg2
            movlw   0x2a
            movwf   reg3
            movlw   0x22
            movwf   reg4
            movlw   0x1c
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
eq          movlw   0x00         ; "="
            movwf   reg0
            movlw   0x00
            movwf   reg1
            movlw   0x3e
            movwf   reg2
            movlw   0x00
            movwf   reg3
            movlw   0x3e
            movwf   reg4
            movlw   0x00
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
l_ar        movlw   0x00        ; "<--"
            movwf   reg0
            movlw   0x08
            movwf   reg1
            movlw   0x10
            movwf   reg2
            movlw   0x3e
            movwf   reg3
            movlw   0x10
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
r_ar        movlw   0x00        ; "-->"
            movwf   reg0
            movlw   0x08
            movwf   reg1
            movlw   0x04
            movwf   reg2
            movlw   0x3e
            movwf   reg3
            movlw   0x04
            movwf   reg4
            movlw   0x08
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
blank       movlw   0x00
            movwf   reg0
            movlw   0x00
            movwf   reg1
            movlw   0x00
            movwf   reg2
            movlw   0x00
            movwf   reg3
            movlw   0x00
            movwf   reg4
            movlw   0x00
            movwf   reg5
            movlw   0x00
            movwf   reg6
            return
quest       movlw   0x1c
            movwf   reg0
            movlw   0x22
            movwf   reg1
            movlw   0x02
            movwf   reg2
            movlw   0x04
            movwf   reg3
            movlw   0x08
            movwf   reg4
            movlw   0x00
            movwf   reg5
            movlw   0x08
            movwf   reg6
            return
;
```

```
; positions
;
pos0        movf    reg0,w
            nop
            movwf   dot0
            movf    reg1,w
            nop
            movwf   dot1
            movf    reg2,w
            nop
            movwf   dot2
            movf    reg3,w
            nop
            movwf   dot3
            movf    reg4,w
            nop
            movwf   dot4
            movf    reg5,w
            nop
            movwf   dot5
            movf    reg6,w
            nop
            movwf   dot6
            return
pos1        movf    reg0,w
            nop
            movwf   dot7
            movf    reg1,w
            nop
            movwf   dot8
            movf    reg2,w
            nop
            movwf   dot9
            movf    reg3,w
            nop
            movwf   dot10
            movf    reg4,w
            nop
            movwf   dot11
            movf    reg5,w
            nop
            movwf   dot12
            movf    reg6,w
            nop
            movwf   dot13
            return
pos2        movf    reg0,w
            nop
            movwf   dot14
            movf    reg1,w
            nop
            movwf   dot15
            movf    reg2,w
            nop
            movwf   dot16
            movf    reg3,w
            nop
            movwf   dot17
            movf    reg4,w
            nop
            movwf   dot18
            movf    reg5,w
            nop
            movwf   dot19
            movf    reg6,w
            nop
            movwf   dot20
            return
pos3        movf    reg0,w
            nop
            movwf   dot21
            movf    reg1,w
            nop
            movwf   dot22
            movf    reg2,w
            nop
            movwf   dot23
            movf    reg3,w
            nop
            movwf   dot24
            movf    reg4,w
            nop
            movwf   dot25
            movf    reg5,w
            nop
            movwf   dot26
            movf    reg6,w
            nop
            movwf   dot27
            return
pos4        movf    reg0,w
            nop
            movwf   dot28
            movf    reg1,w
            nop
            movwf   dot29
            movf    reg2,w
            nop
            movwf   dot30
            movf    reg3,w
            nop
            movwf   dot31
            movf    reg4,w
            nop
            movwf   dot32
            movf    reg5,w
            nop
            movwf   dot33
            movf    reg6,w
            nop
            movwf   dot34
            return
pos5        movf    reg0,w
            nop
            movwf   dot35
            movf    reg1,w
            nop
            movwf   dot36
            movf    reg2,w
            nop
            movwf   dot37
            movf    reg3,w
            nop
            movwf   dot38
            movf    reg4,w
            nop
            movwf   dot39
            movf    reg5,w
            nop
            movwf   dot40
            movf    reg6,w
            nop
            movwf   dot41
            return
pos6        movf    reg0,w
            nop
            movwf   dot42
            movf    reg1,w
            nop
            movwf   dot43
            movf    reg2,w
            nop
            movwf   dot44
            movf    reg3,w
            nop
            movwf   dot45
            movf    reg4,w
            nop
            movwf   dot46
            movf    reg5,w
            nop
            movwf   dot47
            movf    reg6,w
            nop
            movwf   dot48
            return
pos7        movf    reg0,w
            nop
            movwf   dot49
            movf    reg1,w
            nop
            movwf   dot50
            movf    reg2,w
            nop
            movwf   dot51
            movf    reg3,w
            nop
            movwf   dot52
            movf    reg4,w
            nop
            movwf   dot53
            movf    reg5,w
            nop
            movwf   dot54
            movf    reg6,w
            nop
            movwf   dot55
            return
pos8        movf    reg0,w
            nop
            movwf   dot56
            movf    reg1,w
            nop
            movwf   dot57
            movf    reg2,w
            nop
            movwf   dot58
            movf    reg3,w
            nop
            movwf   dot59
            movf    reg4,w
            nop
            movwf   dot60
            movf    reg5,w
            nop
            movwf   dot61
            movf    reg6,w
            nop
            movwf   dot62
            return
pos9        movf    reg0,w
            nop
            movwf   dot63
            movf    reg1,w
            nop
            movwf   dot64
            movf    reg2,w
            nop
            movwf   dot65
            movf    reg3,w
            nop
            movwf   dot66
            movf    reg4,w
            nop
            movwf   dot67
            movf    reg5,w
            nop
            movwf   dot68
            movf    reg6,w
            nop
            movwf   dot69
            return
pos10       movf    reg0,w
            nop
            movwf   dot70
            movf    reg1,w
            nop
            movwf   dot71
            movf    reg2,w
            nop
            movwf   dot72
            movf    reg3,w
            nop
            movwf   dot73
            movf    reg4,w
            nop
            movwf   dot74
            movf    reg5,w
            nop
            movwf   dot75
            movf    reg6,w
            nop
            movwf   dot76
            return
pos11       movf    reg0,w
            nop
            movwf   dot77
            movf    reg1,w
            nop
            movwf   dot78
            movf    reg2,w
            nop
            movwf   dot79
            movf    reg3,w
            nop
            movwf   dot80
            movf    reg4,w
            nop
            movwf   dot81
            movf    reg5,w
            nop
            movwf   dot82
            movf    reg6,w
            nop
            movwf   dot83
            return
;
; row selection
;
row0        movlw   0x00
            call    xmitm
            return
row1        call    row0
            movlw   0x80
            call    xmitm
            call    strobe
            return
row2        call    row0
            movlw   0x40
            call    xmitm
            call    strobe
            return
row3        call    row0
            movlw   0x20
            call    xmitm
            call    strobe
            return
row4        call    row0
            movlw   0x10
            call    xmitm
            call    strobe
            return
row5        call    row0
            movlw   0x08
            call    xmitm
            call    strobe
            return
row6        call    row0
            movlw   0x04
            call    xmitm
            call    strobe
            return
row7        call    row0
            movlw   0x02
            call    xmitm
            call    strobe
            return
row8        call    row0
            movlw   0x01
            call    xmitm
            call    strobe
            return
row9        movlw   0x80
            call    xmitm
            call    row0
            call    strobe
            return
row10       movlw   0x40
            call    xmitm
            call    row0
            call    strobe
            return
row11       movlw   0x20
            call    xmitm
            call    row0
            call    strobe
            return
row12       movlw   0x10
            call    xmitm
            call    row0
            call    strobe
            return
row13       movlw   0x08
            call    xmitm
            call    row0
            call    strobe
            return
row14       movlw   0x04
            call    xmitm
            call    row0
            call    strobe
            return
row15       movlw   0x02
            call    xmitm
            call    row0
            call    strobe
            return
;
clocks      bsf     porta,sclk  ; column clock
            nop
            nop
            nop
            nop
            nop
            bcf     porta,sclk
            return
clockm      bsf     porta,mclk  ; row clock
            nop
            nop
            nop
            nop
            nop
            bcf     porta,mclk
            return
;
latch       bsf     porta,le    ; latch in column data
            nop
            nop
            nop
            nop
            nop
            bcf     porta,le
            return
strobe      bsf     porta,str   ; latch in row data
            nop
            nop
            nop
            nop
            nop
            bcf     porta,str
            return
;
; clear columns
col_clr     movlw   0x00        ; xmit '36' zero's
            call    xmits
            movlw   0x00
            call    xmits
            movlw   0x00
            call    xmits
            movlw   0x00
            call    xmits
            movlw   0x00
            call    xmit4
            call    latch
            return
;
; clear rows
row_clr     movlw   0x00        ; xmit '16' zero's
            call    xmitm
            movlw   0x00
            call    xmitm
            call    strobe
            return
;
; column select
;
col1        call    stp1
            call    stp93
            call    off_4
            call    latch
            return
col2        call    stp2
            call    stp93
            call    off_4
            call    latch
            return
col3        call    stp3
            call    stp93
            call    off_4
            call    latch
            return
col4        call    stp4
            call    stp93
            call    off_4
            call    latch
            return
```

```
col5        call    stp5
            call    stp93
            call    off_4
            call    latch
            return
col6        call    stp6
            call    stp93
            call    off_4
            call    latch
            return
col7        call    stp7
            call    stp93
            call    off_4
            call    latch
            return
col8        call    stp8
            call    stp93
            call    off_4
            call    latch
            return
col9        call    stp9
            call    stp1
            call    stp92
            call    off_4
            call    latch
            return
col10       call    stp9
            call    stp2
            call    stp92
            call    off_4
            call    latch
            return
col11       call    stp9
            call    stp3
            call    stp92
            call    off_4
            call    latch
            return
col12       call    stp9
            call    stp4
            call    stp92
            call    off_4
            call    latch
            return
col13       call    stp9
            call    stp5
            call    stp92
            call    off_4
            call    latch
            return
col14       call    stp9
            call    stp6
            call    stp92
            call    off_4
            call    latch
            return
col15       call    stp9
            call    stp7
            call    stp92
            call    off_4
            call    latch
            return
col16       call    stp9
            call    stp8
            call    stp92
            call    off_4
            call    latch
            return
col17       call    stp92
            call    stp1
            call    stp9
            call    off_4
            call    latch
            return
col18       call    stp92
            call    stp2
            call    stp9
            call    off_4
            call    latch
            return
col19       call    stp92
            call    stp3
            call    stp9
            call    off_4
            call    latch
            return
col20       call    stp92
            call    stp4
            call    stp9
            call    off_4
            call    latch
            return
col21       call    stp92
            call    stp5
            call    stp9
            call    off_4
            call    latch
            return
col22       call    stp92
            call    stp6
            call    stp9
            call    off_4
            call    latch
            return
col23       call    stp92
            call    stp7
            call    stp9
            call    off_4
            call    latch
            return
col24       call    stp92
            call    stp8
            call    stp9
            call    off_4
            call    latch
            return
col25       call    stp93
            call    stp1
            call    off_4
            call    latch
            return
col26       call    stp93
            call    stp2
            call    off_4
            call    latch
            return
col27       call    stp93
            call    stp3
            call    off_4
            call    latch
            return
col28       call    stp93
            call    stp4
            call    off_4
            call    latch
            return
col29       call    stp93
            call    stp5
            call    off_4
            call    latch
            return
col30       call    stp93
```

```
            call    stp6
            call    off_4
            call    latch
            return
col31       call    stp93
            call    stp7
            call    off_4
            call    latch
            return
col32       call    stp93
            call    stp8
            call    off_4
            call    latch
            return
col33       call    stp92
            call    stp92
            call    on_1
            call    latch
            return
col34       call    stp92
            call    stp92
            call    on_2
            call    latch
            return
col35       call    stp92
            call    stp92
            call    on_3
            call    latch
            return
;
on_1        movlw   0x80
            call    xmit4
            return
on_2        movlw   0x40
            call    xmit4
            return
on_3        movlw   0x20
            call    xmit4
            return
off_4       movlw   0x00
            call    xmit4
            return
;
stp1        movlw   0x80
            call    xmits
            return
stp2        movlw   0x40
            call    xmits
            return
stp3        movlw    0x20
            call    xmits
            return
stp4        movlw   0x10
            call    xmits
            return
stp5        movlw   0x08
            call    xmits
            return
stp6        movlw   0x04
            call    xmits
            return
stp7        movlw   0x02
            call    xmits
            return
stp8        movlw   0x01
            call    xmits
            return
stp9        movlw   0x00
            call    xmits
            return
stp92       call    stp9
```

```
            call    stp9
            return
stp93       call    stp9
            call    stp9
            call    stp9
            return
;
;
wait        movlw   0x01
            movwf   count0
repeat      movlw   0x01
            movwf   count1
loop1       decfsz  count1,f
            goto    loop1
            decfsz  count0,f
            goto    repeat
            return
; wait 0.20 seconds ...
wait1       movlw   0xff
            movwf   count0
again       movlw   0xff
            movwf   count1
loop        decfsz  count1,f
            goto    loop
            decfsz  count0,f
            goto    again
            return
;
; xmit data to row drivers
xmitm       movwf   temp0
            movlw   0x08
            movwf   count2
oled0       bcf     status,c
            rrf     temp0,f
            btfss   status,c
            goto    oled1
            bsf     porta,sin
            goto    oled2
oled1       bcf     porta,sin
oled2       call    clockm
            decfsz  count2,f
            goto    oled0
            return
;
; xmit data to column drivers
xmits       movwf   temp0
            movlw   0x08
            movwf   count2
xmts0       bcf     status,c
            rrf     temp0,f
            btfss   status,c
            goto    xmts1
            bsf     porta,sdi
            goto    xmts2
xmts1       bcf     porta,sdi
xmts2       call    clocks
            decfsz  count2,f
            goto    xmts0
            return
;
; xmit 4 bits
xmit4       movwf   temp0
            movlw   0x04        ; set counter to '4'
            movwf   count2
            bcf     status,c    ; clear carrier flag
oled01      rrf     temp0,f     ; rotate out data bit
            btfss   status,c
            goto    oled11
            bsf     porta,sdi   ; c-bit is high, clock out a high bit
            goto    oled21
oled11      bcf     porta,sdi   ; c-bit is low, clock out a low bit
```

```
oled21      call    clocks      ; toggle oled clock line
            decfsz  count2,f    ; all bits sent out?
            goto    oled01      ; no, keep transmitting..
            return
; xmit 6 bits
xmit6       movwf   temp0
            movlw   0x06        ; set counter to '6'
            movwf   count2
            bcf     status,c    ; clear carrier flag
oled07      rrf     temp0,f     ; rotate out data bit
            btfss   status,c
            goto    oled17
            bsf     porta,sdi   ; c-bit is high, clock out a high bit
            goto    oled27
oled17      bcf     porta,sdi   ; c-bit is low, clock out a low bit
oled27      call    clocks      ; toggle oled clock line
            decfsz  count2,f    ; all bits sent out?
            goto    oled07      ; no, keep transmitting..
            return
;
; clear display and turn off
all_off     bcf     intcon,7    ; disable interrupts
            nop
            bcf     intcon,4
            nop
            bcf     intcon,1
            call    col_clr     ; clear col/rows
            call    row_clr
dot         bsf     porta,moe   ; disable drivers
            nop
            bsf     portb,oe
            goto    dot         ; wait for reset...
            retfie
            end
```