```
'---------------Title--------------

' File......DS1620_2.pbp
' Started....5/14/08
' Microcontroller used:  Microchip Technology 16F88
'                        microchip.com
' PicBasic Pro Code, micro-Engineering Labs, Inc.
'                    melabs.com

'--------Program Desciption--------

' The program sets up the Dallas DS1620 digital
' temperature device as a thermostst.  It writes both
' the 9-bit TH, high tempersture limit, and the 9-bit TL,
' the low temperature limit, and displays them on an LCD.
' The DS1620 measures temperatures from -55°C to +125°C
' in 0.5°C increments

'-------------Includes-------------

    INCLUDE "Modedefs.bas"       ' The Mode names for SHIFTIN and
                                 ' SHIFTOUT are defined in the
                                 ' file Modedefs.bas

'--------DS1620 Control Pins-------

    DSRST   VAR     PORTB.0      ' Name PORTB.0 as DSRST (DS1620 Reset)
    DSDQ    VAR     PORTB.1      ' Name PORTB.1 as DSDQ (DS1620 Data)
    DSCLK   VAR     PORTB.2      ' Name PORTB.2 as DSCLK (DS1620 Clock)

'------------Variables-------------

    tH    VAR     WORD           ' WORD to store  high temperature
                                 ' variable, tH
    tH1   VAR     BYTE           ' BYTE to store tH1
    tL    VAR     WORD           ' WORD to store  low temperature
                                 ' variable, tL
    tL1   VAR     BYTE           ' BYTE to store tL1


'---------Initialization--------

    TRISB = 0            ' Set all PORTB pins as outputs

    ANSEL = 0            ' Configure all pins to digital
                         ' operation since not using ADC
                         ' (Analog to Digital Converter)

    OSCCON = $60         ' Sets the internal oscillator in the
                         ' 16F88 to 4 MHz

'--------Main Code--------

    PAUSE 1000           ' Pause 1 second to allow LCD to setup
    DSRST = 0            ' Reset the DS1620
```

```
' Write TH and TL to DS1620

    DSRST = 1           ' Enable DS1620
    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$01,$0032\9]
                        ' Send write high temperature limit
                        ' command, $01, and 9-bit high temperature,
                        ' $0032\9 (+25°C - see table below)
                        ' on data pin DSDQ, synchronized by
                        ' clock pin DSCLK, shift data out
                        ' lowest bit first, LSBPRE.

'    Temperature    Binary Digital Output  Hex Digital Output
'       +125°C          0 11111010             00FA
'       +25°C           0 00110010             0032
'       +½°C            0 00000001             0001
'       0°C             0 00000000             0000
'       -½°C            1 11111111             01FF
'       -25°C           1 11001110             01CE
'       -55°C           1 10010010             0192

    DSRST = 0           ' Reset the DS1620
    DSRST = 1           ' Enable DS1620
    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$02,$00CE\9]
                        ' Send write low temperature limit
                        ' command, $02, and 9-bit low temperature,
                        ' $01CE\9 (-25°C - see table above)
    DSRST = 0           ' Reset the DS1620

' Main loop to read temperature from the DS1620 and then
' display it on the LCD.

' Read high and low temperature limits from DS1620

    DSRST = 1           ' Enable DS1620
    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$A1]
                        ' Send read high temperature limit
                        ' command, $A1
    SHIFTIN DSDQ, DSCLK, LSBPRE, [tH\9]
                        ' Read 9-bit high temperature limit.
                        ' Shifts in 9 bits of variable, tH
                        ' on data pin DSDQ, synchronized by
                        ' clock pin DSCLK, shift data in
                        ' lowest bit first, LSBPRE
    DSRST = 0           ' Reset the DS1620
    DSRST = 1           ' Enable DS1620
    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$A2]
                        ' Send read low temperature limit
                        ' command, $A2
    SHIFTIN DSDQ, DSCLK, LSBPRE, [tL\9]
                        ' Read 9-bit low temperature limit.
                        ' Shifts in 9 bits of variable, tL

    DSRST = 0           ' Reset the DS1620
```

```
' Check to see if TH is below 0°C

    IF tH > $0191 THEN print_tH_below_zero

' If tH is not below 0°C, display high temperature limit, TH,
' as a decimal.

    LCDOUT $fe, 1,"TH = ", DEC (tH >> 1),".",DEC (tH.0*5)," Deg C"

                        ' Shift tH to right one position, (tH >> 1),
                        ' to display the integer portion of tH then
                        ' multiply bit 0 of tH by 5 (tH.0*5) to
                        ' display decimal portion of tH, (.0 or .5).
                        ' The bit tH.0 is either a 0 or 1,
                        ' so (tH.0*5) is either 0 or 5 proceeded
                        ' by a decimal from the entry "."

print_low_temp:

' Check to see if TL is below 0°C

    IF tL > $0191 THEN print_tL_below_zero


' If tL is not below 0°C, display low temperature limit, TL,
' as a decimal.

    LCDOUT $fe, $c0,"TL = ",DEC (tL >> 1),".",DEC (tL.0*5)," Deg C"

    END

' If tH is below 0°C, display high temperature limit, TH,
' as a decimal.

print_tH_below_zero:

' Express tH in the 2's complement form:

    tH1 = ~ tH + 1      ' tH1 is the 2's complement form of tH.
                        ' tH1 is an 8-bit variable to truncate
                        ' the upper 8-bits of the 16-bit tH.

    LCDOUT $fe, 1,"TH = -", DEC (tH1 >> 1),".",DEC (tH1.0*5)," Deg C"
    GOTO print_low_temp

' If tL is below 0°C, display low temperature limit, TL,
' as a decimal.

print_tL_below_zero:

' Express tL in the 2's complement form:

    tL1 =~ tL + 1       ' tL1 is the 2's complement form of tL.
                        ' tL1 is an 8-bit variable to truncate
                        ' the upper 8-bits of the 16-bit tL.
```

```
    LCDOUT $fe,$C0,"TL = -", DEC (tL1 >> 1),".",DEC (tL1.0*5)," Deg C"
    END
```