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

' File......DS1620_1.pbp
' Started....5/10/08
' Microcontroller used:  Microchip Technology 16F88
'                        microchip.com
' PicBasic Pro Code, micro-Engineering Labs, Inc.
'                    melabs.com

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

' The program reads Dallas DS1620 3-wire digital
' temperature device and displays results on an LCD.
' The DS1620 measures temperatures from -55°C to +125°C
' in 0.5°C increments

'-----New PicBasic Pro Command-----

' SHIFTIN
' See page 145 at: http://www.melabs.com/downloads/pbpm304.pdf
'
' SHIFTOUT
' See page 148 at: http://www.melabs.com/downloads/pbpm304.pdf

'-------------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-------------

    temp    VAR     WORD        ' WORD to store temperature variable,
                                ' temp
    temp1   VAR     BYTE        ' BYTE to store 8-bit temp1


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

    TRISB = 0           ' Set pins B7-B0 of PORTB 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

    LOW DSRST           ' Reset the DS1620

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

start:

' Convert temperature from DS1620

    DSRST = 1           ' Enable DS1620

    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$ee]
                        ' Send initiate temperature conversion
                        ' command, $ee, on data pin DSDQ,
                        ' synchronized by clock pin DSCLK, shift
                        ' data out lowest bit first, LSBPRE

    DSRST = 0           ' Reset the DS1620 to enable conversion

    PAUSE 1000          ' Pause 1 second to complete conversion

 ' Read temperature from DS1620

    DSRST = 1           ' Enable DS1620

    SHIFTOUT DSDQ, DSCLK, LSBFIRST, [$aa]
                        ' Send read command, $aa

    SHIFTIN DSDQ, DSCLK, LSBPRE, [temp\9]
                        ' Read 9-bit temperature.
                        ' Shifts in 9 bits of variable temp,
                        ' [temp\9], on data pin DSDQ,
                        ' synchronized by clock pin DSCLK,
                        ' shift data in lowest bit first,
                        ' LSBPRE

    DSRST = 0           ' Reset the DS1620

' Check to see if temp is below 0°C

    IF temp > $0191 THEN print_temp_below_zero

' Display temperature as a decimal

    LCDOUT $fe, 1, DEC (temp >> 1), ".", DEC (temp.0*5), " Degrees C"

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

```
                        ' by a decimal from  the entry "."

    GOTO start          ' Jumps to loop label, starts all over

print_temp_below_zero:

' Express temp in the 2's complement form:

    temp1 = ~ temp + 1  ' temp1 is the 2's complement form of temp.
                        ' temp1 is an 8-bit variable to truncate
                        ' the upper 8-bits of the 16-bit temp.

    LCDOUT $fe, 1,"-", DEC (temp1 >> 1),".",DEC (temp1.0*5)," Degrees C"

    GOTO start

    END
```