

Temperature.tb

```

1 '
2 ' Program Name: Temperature
3 ' Date       : 6/20/2010
4 ' Description : Read the temperature from the Texas Instruments TMP112 sensor
5 '              connected to the I2C interface. Output the temperature value
6 '              to UART0. Build a UDP data packet and transmit it on to a
7 '              preset IP address and port.
8
9 '              The Program Configuration Panel is used to setup the parameters
10 '             to configure UART0 and set the wireless network SSID and channel.
11
12 ' Variables
13 DIM   iTemp           AS INTEGER   ' Holds the final temperature value.
14 DIM   iCharCnt        AS INTEGER
15 DIM   iXmitDataCnt    AS INTEGER
16 DIM   cidSend         AS INTEGER   ' Holds the Connection ID.
17
18 DIM   abyCmd(8)       AS BYTE      ' Holds the command string to the I2C.
19 DIM   abyData(9)      AS BYTE      ' Holds the returned data from the I2C.
20 DIM   abyDataPacket(35) AS BYTE    ' Array to build the wireless data packet.
21 DIM   sI2CReading     AS STRING    ' Holds the string that is send out UART0.
22
23 ' Constants
24 CONST TSENSOR_ADRS   = 0x48       ' I2C bus address of Temperature sensor.
25 CONST TSENSOR_P_TEMP = 0          ' Pointer to Temperature register
26 CONST TSENSOR_P_CONFIG = 1       ' Pointer to Configuration register.
27 CONST TSENSOR_P_TLOW  = 2        ' Pointer to Low Temp trip register.
28 CONST TSENSOR_P_THIGH = 3        ' Pointer to hight Temp trip register.
29
30 TASK   ' Program statements here.
31   ON_ERROR Ehandler                ' Declare an error handler.
32
33   GOSUB RdTemp                      ' Call the subroutine to read the temperature
34                                     ' sensor.
35   ' Convert the temperature value read to a string and print it out UART0.
36   sI2CReading = "Temp Register = " & STR$(iTemp) & CHR$(13)
37   OUTPUT UART0 sI2CReading
38
39   ' Check to see if we are connected to a wireless network.
40   IF LINKED? THEN
41     GOSUB SendTempReading           ' We are connected so transmit the temperature
42                                     ' to the preconfigured IP address and port.
43   ENDIF
44 REPEAT_TASK                          ' Now go to sleep until we wake up at TASK.
45
46 ' Define subroutines and error handlers here.
47
48 ' RdTemp - Read the temperature from the TI TMP112. Data is read out in big-endian format,
49 '          so we need to normalize it and put it into little-endian format.
50 '          The Temp Sensor is located on the I2C bus at address 0x48.
51 SUB   RdTemp
52   abyCmd(0) = TSENSOR_P_CONFIG     ' Build the 2-byte I2C command
53   abyCmd(1) = 0x80                 ' to read the temperature.
54   OUTPUT I2C TSENSOR_ADRS, abyCmd(0), 2 ' Output the command on the I2C interface
55   DELAY(50)                        ' to start conversion. Wait for it to complete.
56
57   abyCmd(0) = TSENSOR_P_TEMP       ' Build the 1-byte I2C command to point to
58                                     ' the register in the TMP112 where the
59                                     ' temperature reading is stored.
60   OUTPUT I2C TSENSOR_ADRS, abyCmd(0), 1 ' Output the pointer to Temp register.
61
62   iCharCnt = INPUT I2C TSENSOR_ADRS, abyData,2 ' Now read temperature data from the TMP112.
63
64   ' We have the 2-bytes of temperature data in the Array abyData().
65   ' We need to convert it into a 12-bit value that is stored into an integer.
66   iTemp = abyData(0) SHL 4          ' Shift the upper 4-bits into iTemp.
67   iTemp = iTemp + (abyData(1) SHR 4) ' Add in the lower 8-bits.
68   RETURN
69 ENDSUB
70

```

Temperature.tb

```

71 ' SendTempReading - Build a UDP packet, as defined below. Then call SendData to
72 '                   transmit the data packet.
73 '   byte    0    1    2    3 - 19    20    21
74 '   -----
75 '         | N | P | 80 | MAC-adrs | LSB | MSB |
76 '   -----
77 SUB SendTempReading
78   abyDataPacket(0) = 0x4E           ' N
79   abyDataPacket(1) = 0x50           ' P
80   abyDataPacket(2) = 0x80           ' Temp sensor packet type.
81   macAddrStr = MAC_ADDRESS
82   abyDataPacket(3) = TOBYTEARRAY(macAddrStr) ' Convert MAC address to string data.
83   abyDataPacket(20) = iTemp AND 0xFF ' Temperature value: LSB
84   abyDataPacket(21) = iTemp SHR 8   ' Temperature value: MSB
85
86   iXmitDataCnt = 22                 ' Number of characters in data packet.
87   GOSUB SendData
88   RETURN
89 ENDSUB
90 '
91 '   SendData - Open a connection if not already open. Then Transmit the
92 '             data packet using the preset packet length.
93 '
94 '             abyDataPacket() = Holds the data to transmit.
95 '             iXmitDataCnt    = Holds the number of characters to transmit.
96 '
97 SUB SendData
98   IF cidSend = 0 THEN
99     cidSend = OPEN UDP "230.1.1.100", 2200
100  ENDIF
101
102  IF cidSend <> -1 THEN
103    SEND cidSend, abyDataPacket(0), iXmitDataCnt
104  ENDIF
105
106  RETURN
107 ENDSUB
108
109
110 END

```