DLP-UTH8
**LEAD-FREE**

# 8-CHANNEL DATA COLLECTION HUB

## 1.0  INTRODUCTION

The DLP-UTH8 is a low-cost, easy-to-use active hub used for connecting multiple temperature and humidity sensors to one PC featuring 4th generation USB silicon from FTDI.  All operational power is taken from the host PC via the USB port.  The temperature and humidity sensor modules (shown below) connect to the DLP-UTH8 using standard Cat-5 intranet cable.
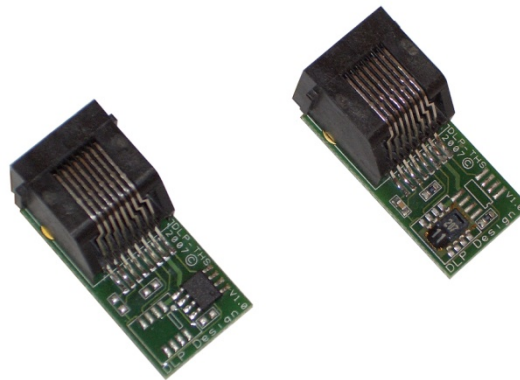


The software running in the hub continuously checks all ports for the presence of sensor modules and automatically reads the data from all connected sensors.  The most current data is held in memory until requested by the host PC.

## 2.0  SPECIFICATIONS

The DLP-UTH8 is compatible with two sensor modules: the DLP-THS-T and DLP-THS-H.  The specifications for each module are shown here.

| Module | DLP-THS-T | DLP-THS-H |
|---|---|---|
| | **(Temperature Only)** | **(Temperature and Humidity)** |
| Temperature | -55 to 125°C (-67 to 257°F) | -40 to 123.8°C (-40 to 254.9°F) |
| Humidity | | 0 to 100% RH |
| Accuracy (T) | ±0.5°C (-10 to 85°C) | ±2.0°C (-30 to 80°C) |
| Accuracy (H) | | ±3%RH (20 to 80%), ±5%RH (0 to 100%) |
| | | |

## 3.0  ABSOLUTE MAXIMUM RATINGS

Stresses above those listed here may cause permanent damage to the DLP-UTH8 hub:

 ➢  Operating Temperature:  0-70°C

Only DLP-THS sensor modules are compatible with the DLP-UTH8 hub.  Connecting any other type device may cause permanent damage to the hub.


## 4.0  USB DRIVERS

USB drivers for the following operating systems are available for download from the DLP Design website:

| Windows XP x64 | Mac OSX |
|---|---|
| Windows Server 2003 | Mac OS9 |
| Windows 2000 | Mac OS8 |
| Windows 98, ME | Linux |

These drivers are available for download from the following page:  http://www.dlpdesign.com/DNLD8/.

NOTE:  If you are using the dual-mode drivers from FTDI (CDM2.00.00) and wish to use the Virtual COM Port (VCP) drivers instead, then it may be necessary to disable the D2XX drivers first via Device Manager.  Right click on the entry under USB Controllers that appears when the DLP-UTH8 is connected, select Properties, select the Advanced tab, check the option for "Load VCP" and click OK.  Once you unplug and then replug the DLP-UTH8, a COM port should appear in Device Manager under Ports (COM & LPT).


## 5.0  USING THE DLP-UTH8

Simply connect the DLP-UTH8 to the PC to initiate the loading of drivers.  Once the drivers are loaded, the DLP-UTH8 is ready for use.  All commands are single-byte commands.

You can either utilize a simple terminal-emulator program or write your own program in your language of choice.  Begin by opening the COM port, set the baud rate to 9600 (1 start bit, no parity, 8 data bits, 1 stop bit), and send single-byte commands as shown below.  The Ping command can be used to locate the correct COM port to be used for communicating with the DLP-UTH8, or you can look in Device Manager to see which port Windows has assigned to the DLP-UTH8.

NOTE: If a COM port is not present after installing the CDM drivers, connect the USB device and open Control Panel > System > Device Manager. Right click on DLP-UTH8 under USB Controllers and select Properties, then the Advanced tab. Check the box marked "Load VCP" then click OK. Unplug and re-plug the DLP-UTH8 and the COM port will be added.

| COMMAND | DESCRIPTION | DATA RETURNED |
|---|---|---|
| ' (27h) | Ping | Z (5Ah) |
| ? (3Fh) | Help | ASCII help text (Terminal Emulator required) |
| ] (5Dh) | Select ASCII data return | None |
| \ (5Ch) | Select Binary data return | None |
| L (4Ch) | Select Degrees F | None |
| ; (3Bh) | Select Degrees C | None |
| | | |
| 1 (31h) | Return CH1 temperature data | ASCII or Binary data |
| 2 (32h) | Return CH2 temperature data | ASCII or Binary data |
| 3 (33h) | Return CH3 temperature data | ASCII or Binary data |
| 4 (34h) | Return CH4 temperature data | ASCII or Binary data |
| 5 (35h) | Return CH5 temperature data | ASCII or Binary data |
| 6 (36h) | Return CH6 temperature data | ASCII or Binary data |
| 7 (37h) | Return CH7 temperature data | ASCII or Binary data |
| 8 (38h) | Return CH8 temperature data | ASCII or Binary data |
| | | |
| Q (51h) | Return CH1 humidity data | ASCII or Binary data |
| W (57h) | Return CH2 humidity data | ASCII or Binary data |
| E (45h) | Return CH3 humidity data | ASCII or Binary data |
| R (52h) | Return CH4 humidity data | ASCII or Binary data |
| T (54h) | Return CH5 humidity data | ASCII or Binary data |
| Y (59h) | Return CH6 humidity data | ASCII or Binary data |
| U (55h) | Return CH7 humidity data | ASCII or Binary data |
| I (49h) | Return CH8 humidity data | ASCII or Binary data |

## 6.0  RETURN DATA TYPES

By default, the DLP-UTH8 returns data to the host PC in the form of ASCII text such that it can be easily displayed using a simple terminal emulator.  Alternatively, data can be returned in binary form requiring the user's host application to calculate the current temperature.

## 7.0  CALCULATION METHODS

The following example code shows how to calculate temperature and humidity if returning data in the binary mode:

```
void calctemp(int mode, int MSByte, int LSByte, float *degc)
{
        int degc_temp;
        int isnegative;

        if(mode == 0) // Humidity Sensor
        {
                degc_temp = LSByte | (MSByte << 8);
                *degc = (float)(( -40.00 + (0.01 * (double)degc_temp)));
        }

        if(mode == 1) // Temperature only sensor
        {
                degc_temp = MSByte | (LSByte << 8);
                isnegative=0;
                if( (degc_temp & 0x8000) == 0x8000 )//if MSBit is set then negative temperature
                {
                  degc_temp &= 0x07ff;
                  isnegative=1;
                  degc_temp = 0x800 - degc_temp;
                }

                degc_temp &= 0x07ff;
                *degc = (float)((float)degc_temp/16.0);
                if(isnegative)
                  *degc *= -1.0;
        }
}

double calchum(int MSByte, int LSByte, float degc)
{
        //the degc parameter is the temperature from the humidity sensor

        int hum_temp;
        double rhtrue, rhlin;
        double tempC;

        hum_temp = LSByte | (MSByte << 8);
        tempC = (float)(( -40.00 + (0.01 * (double)degc)));
        rhlin = -4.0 + (.0405*(float)hum_temp) + (-2.8e-6*(float)hum_temp*(float)hum_temp);
        rhtrue = (tempC - 25.0) * (.01 + .00008 * hum_temp) + rhlin;

        return rhtrue;
}
```

## 8.0  DISCLAIMER

© DLP Design, Inc., 2008

Neither the whole nor any part of the information contained herein nor the product described in this manual may be adapted or reproduced in any material or electronic form without the prior written consent of the copyright holder.

This product and its documentation are supplied on an as-is basis, and no warranty as to their suitability for any particular purpose is either made or implied.  DLP Design, Inc. will not accept any claim for damages whatsoever arising as a result of the use or failure of this product. Your statutory rights are not affected.  This product or any variant of it is not intended for use in any medical appliance, device, or system in which the failure of the product might reasonably be expected to result in personal injury.

This document provides preliminary information that may be subject to change without notice.


## 12.0   CONTACT INFORMATION

DLP Design, Inc.
1605 Roma Lane
Allen, TX  75013

Phone:   469-964-8027
Fax:       415-901-4859

Email Sales:      sales@dlpdesign.com
Email Support:   support@dlpdesign.com
Website URL:     http://www.dlpdesign.com