DLP-VLOG

LEAD-FREE
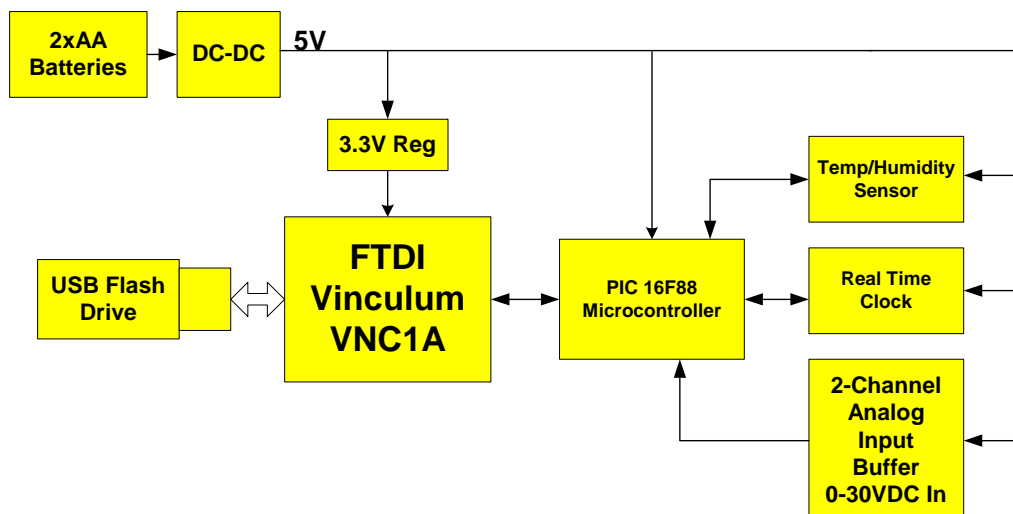
VINCULUM-BASED TEMPERATURE / HUMIDITY / VOLTAGE DATA LOGGER

FEATURES:

➢ Virtually Unlimited Data Storage Utilizing FTDI's New Vinculum™ USB Host IC

➢ Data Logged to USB Flash Drive

➢ Low-Power Modes for Long Battery Life

➢ System Power via Two AA Batteries

➢ Lead-Free; RoHS Compliant

DEVELOPMENT KIT INCLUDES:

➢ DLP-VLOG Module

➢ C Source Code for the Microcontroller Firmware

➢ Windows Support Software for Real Time Clock Setup

➢ Windows Data Conversion Software

➢ C Compiler and Device Programmer/Debugger

1.0 INTRODUCTION

The DLP-VLOG is intended as both a demonstration platform for the new Vinculum IC as well as a learning tool. The Development Kit is designed to log temperature, humidity and voltage data to a USB Flash drive. Using the Vinculum IC from FTDI, this system demonstrates how one can provide virtually unlimited data storage for a small microcontroller.

This system can be beneficial to electronics enthusiasts with virtually any level of technical expertise. Based on the Vinculum Host Controller IC and the Microchip PIC16F88 Microcontroller, the DLP-VLOG can function as a digital design trainer, a microprocessor development system, or as a rapid path to learning how to implement a USB Host Controller for interfacing a USB Flash drive to a small microcontroller.

Programming headers on the board allow for firmware updates to both the microcontroller and the Vinculum IC so that the board can be updated with additional features developed by the user as well as firmware updates for the Vinculum.

A highly optimized C compiler from Custom Computer Services is provided with the kit that supports the microcontroller firmware. Changes made to the firmware can be debugged and reprogrammed into the microcontroller using the CCS ICD Programmer/Debugger module, which is also included with the kit.

The sensor used in this design is the Senserion SHT11. While this sensor will acquire temperature data in the range of –40 to 123.8°C (-40 to 255°F), it is recommended that you restrict exposure of the module to the commercial temperature range of 0-70°C. The humidity range for the sensor is 0-100% (non-condensing).

Refer to the datasheet for the SHT11 for detailed specifications.

2.0 DOCUMENTATION

The following items are available from the VLOG download page (http://www.dlpdesign.com/vdn9/):

- An expanded version of this datasheet/manual

- The electrical schematic in both PDF and Orcad formats

- Microcontroller firmware source code

- The modified version of the DS1302.c source code file

- A Windows utility for creating the Real Time Clock setup file

- A Windows utility for converting DLP-VLOG output data to floating point values

3.0 FLOW DIAGRAMS

3.1 STARTUP SEQUENCE

```
┌─────────────────────┐
│  Power Up / Reset    │
│                      │
│  Flash LEDs          │
└─────────────────────┘
            │
            ▼
┌─────────────────────┐   #use rs232(baud=9600, xmit=PIN_B5, rcv=PIN_B2, FORCE_SW)
│                      │   //send 'E' until echoed back
│                      │   putc('E');
│  Power Up and        │   putc(0x0D);
│  Synchronize with    │
│  Vinculum            │   //send 'e' once
│                      │   putc('e');
│                      │   putc(0x0D);
└─────────────────────┘
            │
            ▼              putc(0x0D);
      ◇◇◇◇◇◇◇◇◇            // "D:\>" or "No Disk" returned     ┌──────────────────────────┐
     ◇         ◇                                                │  Power Off Vinculum      │
    ◇ USB Flash ◇         N                                     │  ------------------------ │
    ◇ Drive Present ◇──────────────────────────────────────▶   │  Sleep 10 Seconds         │
     ◇    ?    ◇                                                │  (100uA)                  │
      ◇◇◇◇◇◇◇◇◇                                                 └──────────────────────────┘
            │
            ▼  Y
```
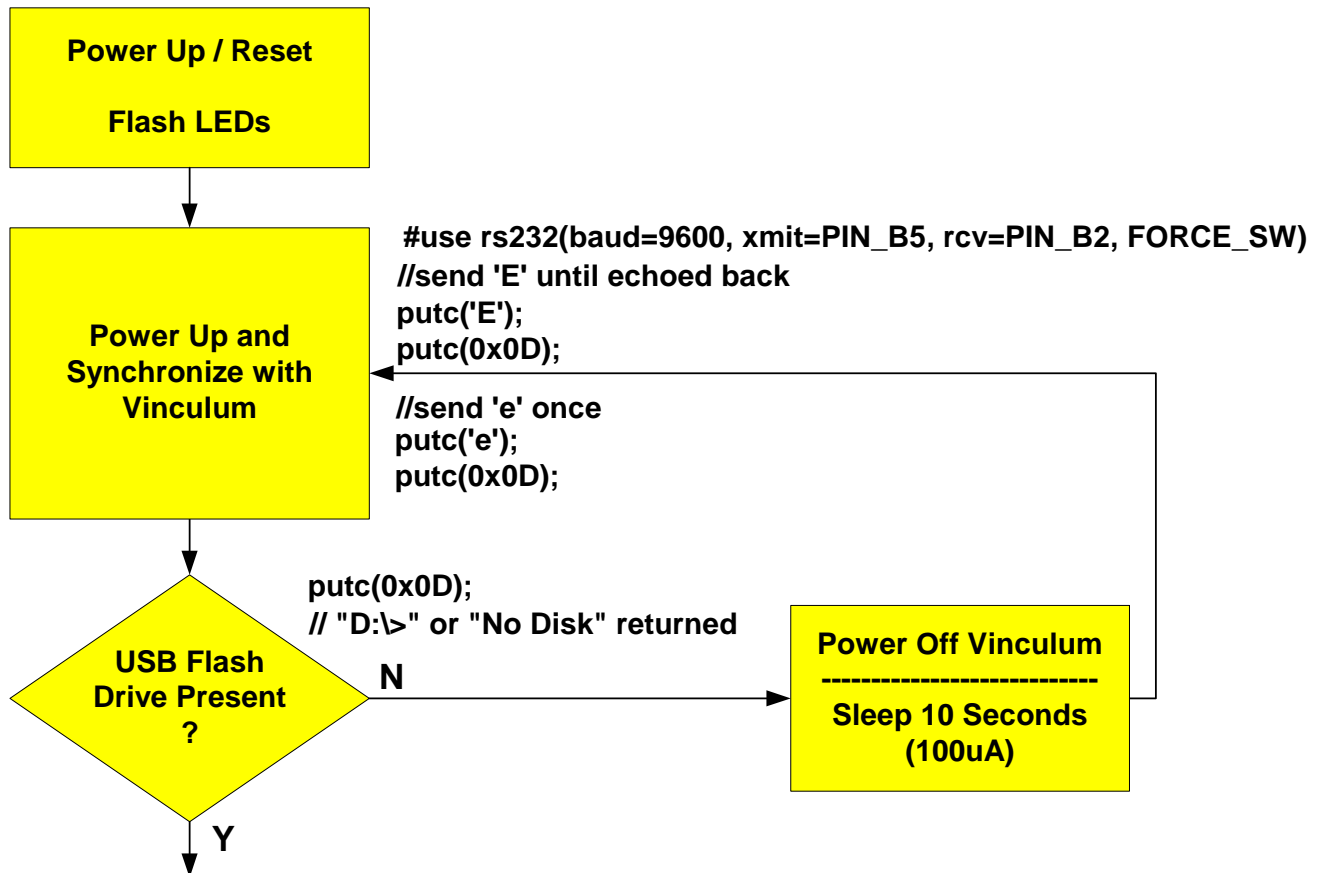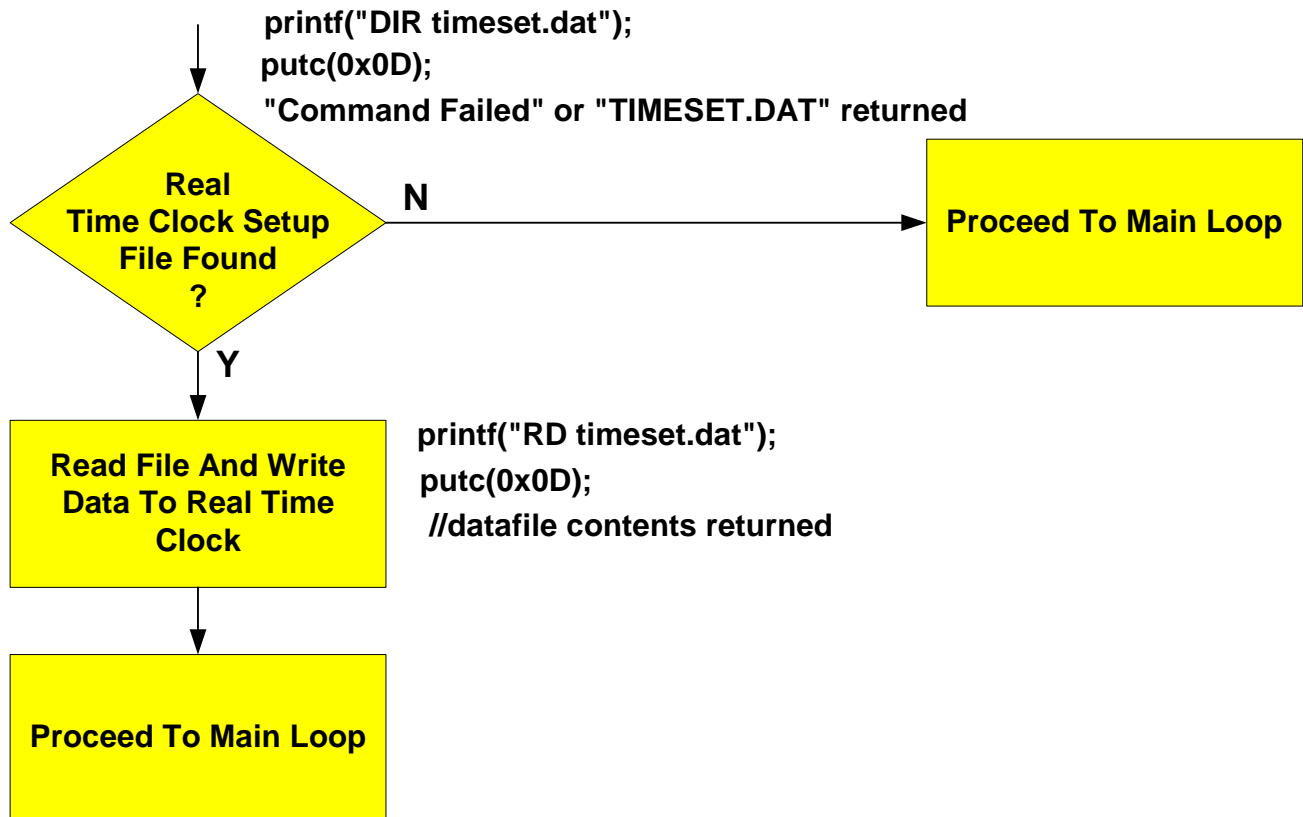
On power up or after a device reset, the firmware needs to know if a USB Flash drive is connected to the board. In order to do this, the microcontroller must first synchronize with the Vinculum. This is accomplished by sending an upper-case 'E' to the Vinculum via the serial interface until it is returned by the Vinculum. Once the 'E' is returned, a lower-case 'e' is sent to complete the process.

Next, the firmware sends the carriage return control code 0xD. The Vinculum will respond with either "D:\>" indicating that there is a Flash drive connected and ready or "No Disk", which indicates that no drive is connected.
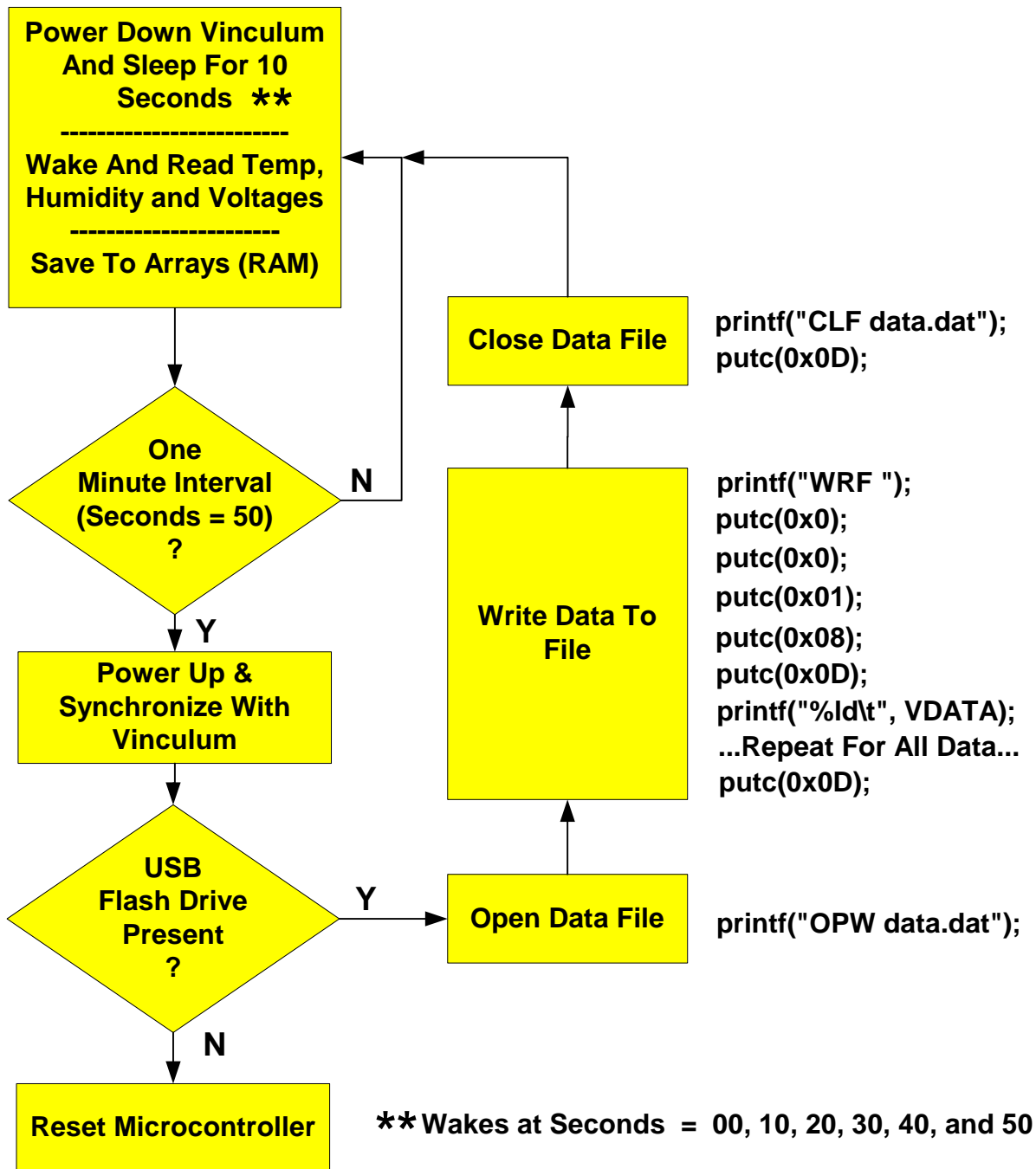
If no drive is found, then the firmware remains in a loop waiting for a drive to be connected so that data can be saved. While the system waits for a drive, it will spend most of its time sleeping with the Vinculum powered off to conserve battery power.

3.2 REAL TIME CLOCK SETUP

```
printf("DIR timeset.dat");
putc(0x0D);
"Command Failed" or "TIMESET.DAT" returned
```

Real Time Clock Setup File Found ?

N → Proceed To Main Loop

Y

Read File And Write Data To Real Time Clock

```
printf("RD timeset.dat");
putc(0x0D);
 //datafile contents returned
```

Proceed To Main Loop

Once a Flash drive has been connected to the system, the next step is to determine whether the user has placed a setup file for the Real Time Clock on the Flash drive. If the file is present, it is read from the drive, the data is written to the Real Time Clock, and then the file is deleted so as to prevent the time data from being overwritten in the Real Time Clock the next time the module is reset.

3.3 MAIN LOOP

```
Power Down Vinculum
And Sleep For 10
Seconds **
---------------------------
Wake And Read Temp,
Humidity and Voltages
---------------------------
Save To Arrays (RAM)
```

One Minute Interval (Seconds = 50)?

N

Power Up & Synchronize With Vinculum

Y

USB Flash Drive Present ?

Y

Open Data File

N

Reset Microcontroller

Close Data File

```
printf("CLF data.dat");
putc(0x0D);
```

Write Data To File

```
printf("WRF ");
putc(0x0);
putc(0x0);
putc(0x01);
putc(0x08);
putc(0x0D);
printf("%ld\t", VDATA);
...Repeat For All Data...
putc(0x0D);
```

```
printf("OPW data.dat");
```

** **Wakes at Seconds = 00, 10, 20, 30, 40, and 50**

Once a Flash drive is connected and the Real Time Clock is set with the current time, the main loop can begin taking readings. Again, the system remains in sleep mode between readings to conserve power.

The microcontroller comes out of sleep mode about once every 600mS to read the Real Time Clock and see if it is time to wake up. If a new ten-second interval has begun, then all readings are taken

and stored to microcontroller RAM in raw form as A/D counts. If the Real Time Clock indicates the end of a one-minute period, then the data is written to the Flash drive.

If the Flash drive is not present, then the user has removed it for data analysis and the microcontroller simply resets itself to restart the process at the end of the one-minute interval.

Note that neither removing the Flash drive nor resetting the microcontroller via the reset button will erase the time in the Real Time Clock. Removing power from the module is the only way to erase the current time.

4.0 CIRCUIT DESCRIPTION

4.1 CURRENT MONITOR PORT

Great care has been taken in the design of this system to ensure that battery power is conserved wherever possible. When no Flash drive is connected and the system is powered and waiting for the drive from the user, it remains in sleep mode. The system also stays in sleep mode between readings. While in sleep mode, the current draw from the batteries is approximately 100uA.

The 2xAA battery pack is mounted to the bottom of the PCB. Power enters the PCB from the battery pack and immediately connects to R10; a zero-Ohm resistor. To monitor the current draw from the batteries, remove R10 from the board and connect a current meter in series at JP1.

4.2 DC-DC CONVERTER

U3 is a synchronous boost converter that will convert the 3 volts from the batteries to regulated 5-volt power. This device was chosen in part due its low power consumption. When the DLP-VLOG is in sleep mode and only drawing 100uA from the batteries, U3 is only consuming 25uA itself.

4.3 REAL TIME CLOCK

U7 is the Real Time Clock. Three I/O port pins are required for communication with the micro-controller. Once the time and date are written to the registers in the IC, it will keep the correct time so long as power is applied.

4.4 SENSOR

U1 is the temperature and humidity sensor. Only two I/O port pins are required for communication with the microcontroller. The data I/O line on the Real Time Clock and the data line for the sensor both share a port pin on the microcontroller (DATA/PGC).

4.5 MICROCONTROLLER CLOCK

U4 is a Microchip PIC16F88 Microcontroller. Since all communication with the devices on the board are either asynchronous or at 9600 baud, no crystal or ceramic resonator is required to clock the microcontroller. Instead, its internal oscillator is run at 8MHz.

4.6 VINCULUM POWER SOURCE

Power for the Vinculum and Flash drive is controlled by the microcontroller via Q1. At the one-minute interval, the microcontroller pulls VPWR low to turn on Q1 and check for the presence of the Flash drive. This action supplies 5-volt power to the Flash drive and 3.3-volt power to the Vinculum (via the U5 regulator).

4.7 PROGRAMMING INTERFACES

Firmware is written to the program memory in the microcontroller via the CCS ICD Programmer/ Debugger and IDE software. A small adapter PCB is provided that connects to both the 5-pin header (H1) on the DLP-VLOG and the modular cable connected to the ICD.

Should the firmware in the Vinculum need to be updated, a 6-pin header (H2) is provided for connection to the TTL-232R 3.3V USB-to-serial cable. This cable is available for purchase from Mouser Electronics under part number TTL-232R-3V3, or from the FTDI on-line store. Be sure to use the 3.3 volt version of this cable. The Windows software that performs this programming process is available for download from the Vinculum website (http://www.vinculum.com/).

To update the Vinculum's firmware, start by removing the batteries from the battery holder. Place a shorting jumper across JP3 (MCLR) to hold the microcontroller in the reset state, and another shorting jumper across PROG# and GND at JP2. Then connect the TTL-232R cable to header H2 with the black wire going to pin 1 of the header. Finally, run the VPROG.exe utility to erase and reprogram the Vinculum.

4.8 ANALOG BUFFERS

Analog voltages can be measured on the CH1 and CH2 inputs. Voltages present at these inputs are first divided by R11, R12, R16 and R17 to provide an input range of 0-30 volts, and then buffered by U8, which has been configured for unity gain.

5.0 FIRMWARE DESCRIPTION

This section will only cover items that some developers may find confusing. Questions that arise from the source code listing that are not covered here should be directed to support@dlpdesign.com.

5.1 INTERNAL CLOCK

As mentioned earlier, since the serial interface to the Vinculum runs at 9600 baud, the internal oscillator in the PIC 16F88 will work just fine for the system clock source. The internal oscillator is enabled and selected via the CONFIG register when firmware is programmed into the microcontroller.

The source code line **"#use delay(clock=8000000,restart_wdt)"** tells the compiler the speed at which the micro is running and enables use of the delay functions.

5.2 SERIAL PORT SETUP

The PIC16F88 Microcontroller has an on-chip UART that can be utilized for communication with the Vinculum. Please note that while the UART has interrupt capability, interrupts were not used for this project in an attempt to make the code easier to understand.

The "#use RS232" pre-processor directive is shown but not called, so the Vinculum can remain powered off in low-power mode. The "Force Software" parameter will cause software serial I/O routines to be generated even though the UART pins on the micro are specified.

As soon as the #use RS232 directive is issued, the TX line will be taken high to the idle state. If this line is high while the MOSFET is off and no power is applied to the Vinculum, there will be current draw into the Vinculum via the TX line through the protection diodes.

5.3 CONSTANTS

A few constants are defined towards the top of the code listing; for example, **#define S1 "OPW data.dat".** This was done as opposed to using the SPRINTF or PRINTF commands so as to save program memory.

6.0 REAL TIME CLOCK UTILITY

At power up, the data registers in the Real Time Clock IC contain invalid time and date data. The correct date and time must be written to these registers in order for valid time data to be logged to the data file.

Setting the date and time in the Real Time Clock is accomplished by running the software utility "SetupFileGen.exe" that is located on the Flash drive supplied with the kit. When run, this program creates a binary file containing both date and time information. You can simply copy the binary file to the root folder in the Flash drive, and when the Flash drive is reconnected to the DLP-VLOG module, the Real Time Clock will be updated with the correct time.

You will note that there is a several-second delay from when the reset button on the DLP-VLOG is pressed to when the data is actually read and stored into the Real Time Clock.

7.0 DATA CONVERSION UTILITY

In an effort to conserve program memory in the microcontroller, no floating-point calculations are performed. All temperature, humidity and voltage data is written to the Flash drive in raw form as A/D counts.

The software utility "VLOG File Convert.exe" is provided to convert the raw data file to usable floating-point data. To utilize this, simply run the application, select the file to be converted, click on Convert & Save and select a location for the output file.

The utility will convert the raw data from this: **08/26/06 15:26:00 6705 1091 0819 0226 0552**

To this: **08/26/06 15:26:00 27.38 84.14 37.1 26.59 12.15 2.661**

The nine data elements produced are as follows:

- Date
- Time
- Degrees C
- Degrees F
- Humidity %
- Voltage CH1
- Voltage CH2
- Voltage Battery

8.0 DISCLAMER

©DLP Design, Inc., 2006

Neither the whole nor any part of the information contained herein nor the product described in this datasheet 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 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.

9.0 CONTACT INFORMATION

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

Phone: 469-964-8027
Fax: 415-901-4859
Email: support@dlpdesign.com
Internet: http://www.dlpdesign.com

AMEYA360
Components Supply Platform

Authorized Distribution Brand :

ROHM	Sunlord	SSM SUSUMU	AVERLOGIC	NXP	Elprotronic
2Pai Semi	Ambarella	CanaanTek	Firstohm	GigaDevice	威寨半导体

Website :

Welcome to visit   www.ameya360.com

Contact Us :

➤ Address :

 401 Building No.5, JiuGe Business Center, Lane 2301, Yishan Rd
Minhang District, Shanghai , China

➤ Sales :

Direct   +86 (21) 6401-6692

Email   amall@ameya360.com

QQ   800077892

Skype   ameyasales1 ameyasales2

➤ Customer Service :

Email   service@ameya360.com

➤ Partnership :

Tel   +86 (21) 64016692-8333

Email   mkt@ameya360.com