Mammal Tracking Project - Current Meter Board Instructions

Emery Newlon

Overview

The Current Meter Project consists of two boards, a Freescale Freedom board that uses a Kinetis L class microcontroller, and a Resistor Board that measures current by measuring the voltage across a small resistor in series with a load. The Freedom board converts this voltage to a digital difference that is sampled about 2000 times a second. These samples are sent via the board's OpenSDA USB connector to a PC where they are logged by a communication program, such as PuTTY, to a file. This file can then be read and its information displayed via MATLAB functions.

Current Meter Board

The Freedom Board is referred to as the Current Meter Board (CMB) and is powered by the USB connection through the OpenSDA USB port. Its Analog to Digital Converter uses an Analog Reference that can be smoothed by a capacitor/resistor combination to limit power supply noise on the Freedom Board from affecting the current measurements. If this is not considered a problem, the capacitor/resistor combination can be left out. The board can be externally powered rather than powered through the USB port. If this option is not wanted the external Vcc pin can be left out. The required pins are thus only Ground, the V+, and the V- pins from the Resistor Board.

There is a set of eight pins on the J1 header that can be used as 3.3V level inputs to the board. The value of these pins is transmitted with the current samples allowing additional information to be logged with the current information. A pin on the J2 header follows the logging state. When data is being logged it is set (3.3V) and when data is not being logged it is clear (0V). This pin can be used to control operations on the device being sampled if desired. It syncs closely with the samples being taken.

Further information about the Current Meter Board is available in the "Current Meter Board Pins" document.

Resistor Board

The Resistor Board contains a resistor network used to measure the voltage across a resistor placed between a power source and a load. Power and Ground come from the power source as Vin and GND. Power and Ground go to the destination load as Vout and GND. The voltage across the series resistor is passed to the Current Meter Board via the pins V+, V-, and Ground. (All grounds are connected together.) The Resistor Board information is available in the "Current Meter Board Pins" document.

Current Meter Software

The Current Meter software was produced using Freescale's CodeWarrior development package in the C language and using the Processor Expert. An S-Record file "CurrentMeter.srec" was produced that can be loaded directly into the Freedom Board using the MSD-FRDM-KL25Z-Pemicro_v???.SDA application (??? is the version number) located in the FRDM-KL25Z_QSP.zip archive available from Freescale through their website via the www.freescale.com/webapp/sps/site/prod_summary.jsp?code=FRDM-KL25Z link.

NOTE: The directions for downloading applications and examples from this file allow the user to drag a file from the .zip archive directly to a Freedom Board folder. This will not work. The file must first be dragged to a non-compressed folder (not a .zip archive), then dragged from there to the Freedom Board folder for the operation to succeed.

Current Meter Logging

Once the Current Meter software has been installed on the Freedom Board, every time it reboots it immediately opens a COM port on the PC that it is connected to via the OpenSDA USB port. This port may be COM5 or COM4. A serial communication program such as PuTTY can be used to talk to the Current Meter software via this program. Its settings should be:

```
Connection → Serial
     Serial line to connect to   COM5
     Speed (baud)                115200
     Data bits                   8
     Stop bits                   1
     Parity                      None
     Flow control                None
Session → Logging
     All session output
     Browse…
          current_meter.log      in a desired directory
     Always overwrite it
Session
     Connection Type             Serial
     Saved Sessions              Current Meter
     Save
```

These setting can be reloaded when PuTTY is started by:

```
Connection
     Saved Sessions              select “Current Meter”
     Load
     Open
```

The Current Meter software displays:

```
Current Measurement
 use ‘+’ key to start logging ‘-’ key to stop logging.
```

This may not be evident if PuTTY’s “Open” key was not pushed immediately after the Current Meter Board was booted (plugged in to the USB) as that text would be lost otherwise.

At this point, when the ‘+’ key is pressed on the keyboard (or keypad if it is in NumLock mode) the Current Meter Board will start displaying current measurements. When the ‘-‘ key is pressed measurements will stop being displayed. These operations can be repeated any number of times. They can be used to log the current information when a specific action is taken at the current load.

When measurements are finished the Source Power should be shut off. Then the Current Meter Board should be disconnected from the USB cable. Finally the PuTTY window should be closed. At this point the "current_meter.log" file should be copied or renamed to prevent it from being overwritten the next time PuTTY is run.

Current Meter Display

There are several MATLAB functions that can be used to process a Current Meter log file. The `current_samples` function will load the samples from a log file into three variables when given the name of a log file and the scaling of samples in it in milliamps. The later scaling factor is calculated from the Calibration Factor (CF) measured as described in the "Current Meter Board Pins" document (e.g. 1.01) multiplied by the Analog Reference Voltage (e.g. 3.3) divided by 32,768 times 1000 to put it into milliamps.

```
scale = 1.01 * 3.3 / 32768 * 1000 ;
[mamps, stimes, segs, sdata] =
            current_samples ('current_meter.log', scale) ;
```

The variable `mamps` receives all the current samples in milliamps into it. The variable `stimes` has the times the samples were taken at in microseconds. (It is not very useful.) The variable `segs` has the start indices of the segments the samples are broken into. This latter operation is done by comparing the sample times of adjacent samples. When that time is greater than one second a new segment is recognized. This effectively creates a segment for each pair of '+'/'-' characters used when logging was being done. The variable `sdata` contains the external data inputs at the time each sample was taken.

The function `current_graph` will graph all the samples of a display on a line plot.

```
current_graph (mamps, sdata, segs, segno, zones) ;
```

The `mamps`, `sdata`, and `segs` arguments are the variables returned by the `current_samples` function. The `segno` argument specifies which segment as defined by the `segs` variable is to be displayed. The `zones` argument is optional. It is a vector of multipliers to the standard deviation of the samples that is used to separate the samples displayed into different colored zones.

The `sdata` information is display as the lower four bits at the bottom of the graph and the upper four bits at the top of the graph. If a bit is displayed in a dark color (black or blue) it is zero. If it is displayed in a light color (white or green) it is one. Unconnected input pins will be displayed as ones as they each have pull-up resistors attached to them internally.

The function `current_display` will display the samples as a set of individual points or averaged points.

```
current_display (mamps, segs, segno, display_spec, rows, columns) ;
```

The first three arguments are the same as those named in the `current_graph` function. The rest are optional. The number of `rows` and `columns` specifies the number of samples to display on a page and how they are laid out. Default is 5 and 200. Changing this to 5 and 400 will display a full second on a page. The `display_spec` argument can either be `pageno` or `[start-millisec end-millisec]`. The `pageno` argument specifies which page of this size in the segment is to be displayed. If it is zero all samples will be averaged to produce a single page. The time vector specifies a time range to display the samples for.