

Table of Contents

Introduction.....	2
Connecting to the PWM-CTA	2
Experiment Setup.....	2
Loading / Saving.....	3
Running the Graph.....	4
Channel Adjustment.....	5
Ea	5
Data Acquisition	5
File Format.....	5
Data Processing.....	5
Post Processing in PWM Comm.....	5
Matlab Routines	6

User Guide

Introduction

This document is not meant to be a comprehensive manual. It is walk through for the steps required in creating and running an experiment with the PWM-CTA.

Connecting to the PWM-CTA

Before using PWM Comm it is important to have installed the Datal card and drivers. If PWM Comm is started, and detects that the Datal data acquisition board is not properly installed, a prompt will appear allowing the software to continue loading, but without the data acquisition (graph) functionality.

When PWM Comm is started it will default to using comm. 1, which is generally the first serial port on a given machine. This can be changed by choosing a different comm port from the Settings Menu. The PWM Comm program will display an error as seen in XXX if it cannot successfully send commands to the PWM-CTA. If this window shows up it generally means that an incorrect comm. port has been selected or the PWM-CTA is not turned on. If neither of these are the case, choosing Reset Communications from the File Menu will send a reset signal to the PWM-CTA.



Figure 1: Comm Error

Experiment Setup

Each set of channel parameters is saved as an experiment file with a “.pwm” extension. The creation of an experiment is done through the Channel Setup window, which can be accessed through the File Menu by choosing Channel Setup and can be seen in Figure 2. The Channel Setup window will show 18 channels regardless of the number of channels installed in the PWM-CTA. The channels are numbered sequentially from left to right on the box and any non-existent channels will be ignored by the software.

Experiment Description / Channel Title

Both of these boxes hold organizational information for the user, but have no effect on the operation of the PWM-CTA. The experiment description box holds a brief description of the experiment and can be changed to describe the setup being created. The Channel Title boxes hold individual descriptions for each channel.

Function

The function box selects the functionality for a given channel. All channels will have a full set of functions listed. In general, all functions except Off, Normal (PWM), and Ext A/D can be ignored. Channels that are set to do only External A/D will return invalid data if set to Normal, Resistance, or Offset modes.

Frequency

The frequency at which the PWM-CTA takes data can be set through the drop down box. Available frequencies are integer divisions of 100kHz. The PWM-CTA runs all channels simultaneously at the chosen frequency.

RCold

RCold shows the cold resistance of a wire. It cannot be entered manually, but must be measured using the update button.

A/D Range

When used as an analog to digital converter, a channel has four possible gain settings. Decreasing the range will allow higher sensitivity for small input signals.

Update

After setting a channel to Normal(PWM) mode, or making any adjustments to a PWM channel, clicking the update button will measure RCold and calculate Ea and Eref based on the rest of the parameters. It will have no effect on channels in any other mode.

For most applications these are the only necessary changes, more details on the other settings can be found in the full manual.

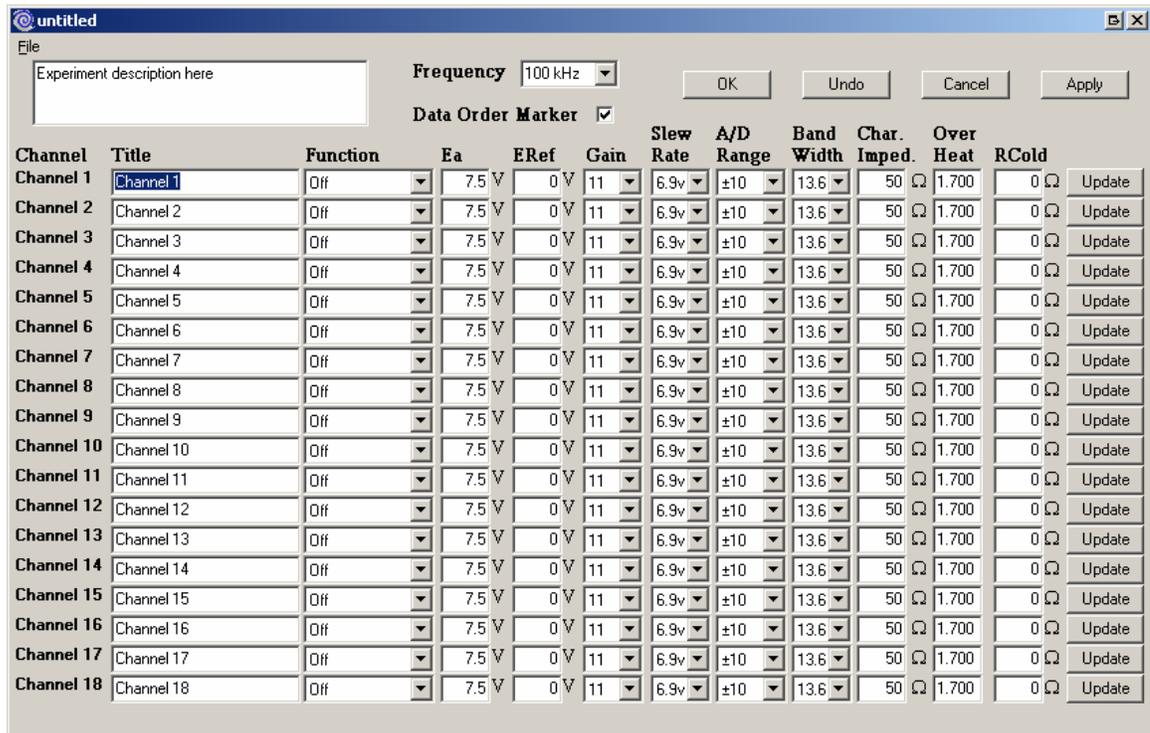


Figure 2: PWM Comm Channel Setup Window

Loading / Saving

If an experiment has already been defined, it may be loaded through File -> Open in either the main GUI window or the Channel Setup window. If a new experiment has

been defined, clicking OK, Apply, or choosing Save from the File menu will bring up a file selection window. If an experiment is modified, clicking OK, Apply or File -> Save will update the experiment file.

The naming scheme for data output from the PWM-CTA is fixed by the experiment file name. Data files will have a counter appended to the file name and the file extension will be changed to .pdt. The counter will default to 001 when a new experiment is loaded and will increment every time data is acquired. The number of digits used as well as the next number can be set by choosing File -> File Number from the main PWM Comm window. The data files will be saved to the same location as the experiment file.

Running the Graph

Opening the channel setup window will automatically stop all running wires. To restart wires, click the Run Active Channels button in PWM Comm (Figure 3). The indicator next to the button will turn green and the red LEDs on the PWM-CTA chassis will light up if the channel is in Normal(PWM) mode. It is important not to disconnect a wire while it is running (LED on).

An individual channel can be viewed by selecting it from the Channel dropdown box. Real time data can be shown for a channel by clicking the Show Graph button. This will start the graph running and display some basic statistics below it if available. If the data is not shown on the graph, right clicking it will allow adjustment of the range. PWM Comm will attempt to customize the range to the selected channel if a single channel is selected.

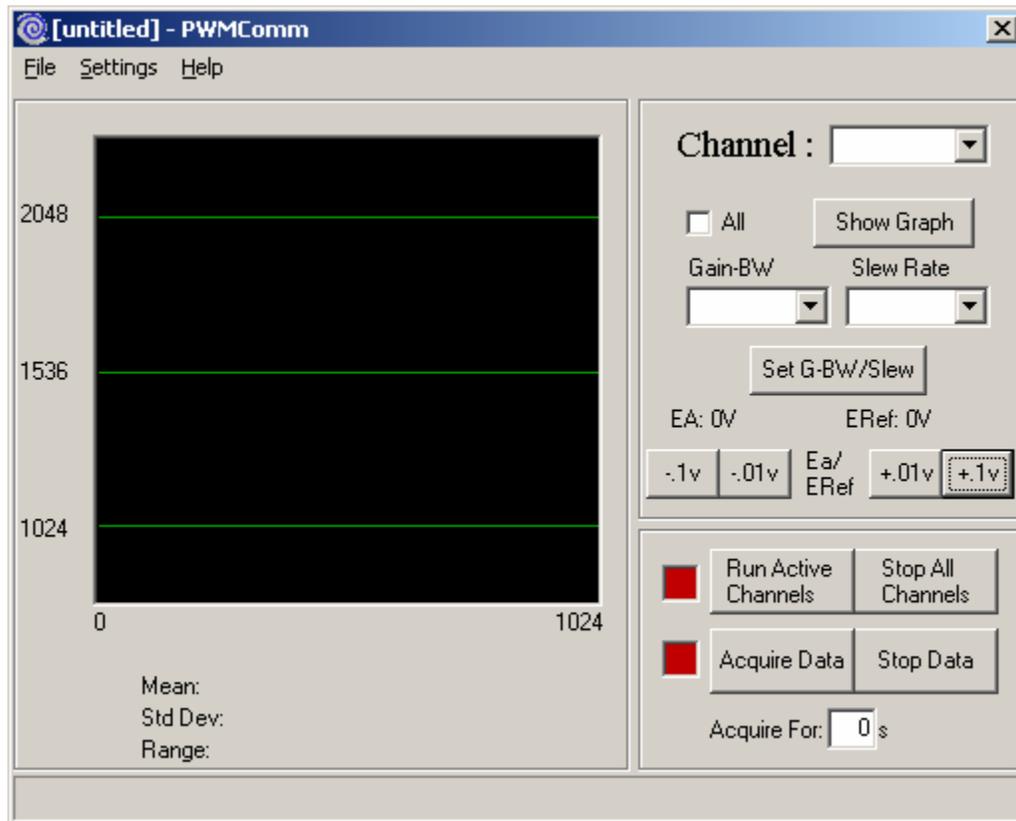


Figure 3: PWM Comm Main Window

Channel Adjustment

Ea

Channels set to Normal mode will most likely require some fine adjustment to achieve the maximum dynamic range. To do this, the wire should be placed in the maximum expected flow while the voltage (Ea) is adjusted to bring the counts near but not exceeding the maximum counts for the wire. Maximum counts can be calculated as:

$$MaxCounts = \frac{65536}{32 \left(\frac{Freq}{100,000} \right)}$$

To adjust the drive voltage for a channel, select the channel from the Channels box and set the vertical range of the graph to show MaxCounts/2 to Maxcounts. With the channel running, clicking the -.1v or -.01v buttons will raise the counts, while the +.1v or +.01v buttons will lower the counts.

Data Acquisition

After all channels are adjusted, data may be acquired at any time. Data must be acquired for a pre-determined amount of time set at the bottom of the PWM Comm window. Clicking the Acquire Data button will automatically take the proper number of samples to fill the time and save them to a file. The file will have the same name as the experiment file with an added automatically incrementing counter. The file extension will be changed to .pdt.

File Format

The data from the PWM-CTA is stored to a .pdt file as binary values. Also stored to the file is a timestamp from when the data set was started. The time is saved as a 64-bit floating point value which holds the number of seconds that have passed since midnight. After the timestamp the data is sequentially written to the file. Each channel's data consists of one 16-bit integer which must later be converted to a pulse width or voltage. If the data order marker is turned on (which it generally should be) an extra channel's worth of data is written before the first channel to assist with sorting the data later.

Data Processing

At this point the data is ready for processing. If you are using your own processing code, this section is unnecessary, otherwise some basic processing is offered through PWM Comm and Matlab.

Post Processing in PWM Comm

Since the PWM-CTA does not output formatted data, a routine has been included to convert the raw data files to a more useful format. This routine can be accessed by selecting Postprocess Data from the File menu.

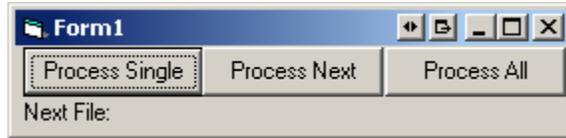


Figure 4: Postprocessing Window

The postprocessing interface is shown in Figure 4. Process Single will prompt the user to choose a .pdt file then convert it, Process Next will process the next .pdt file in sequence (assuming it is available and the default numbering scheme was used), and Process All will prompt for a .pwm file and convert all .pdt files found with the same name.

Regardless of the option used, post processing will require the .pwm file to be in the same location as the .pdt files and the output will be a .pdp file of the same name as the input file. This processed file will no longer contain a timestamp or the data order marker. All PWM channels will be left as integers while all A/D channels will be converted to a single precision floating point value based on the range chosen in the .pwm file.

Matlab Routines

The included Matlab routines will read in a processed .pdp file and create an array of the data. They will also convert the PWM counts to flow rate if desired.