

VME dds curve editor.

B. A. Schupbach
email: schupbach@fnal.gov

November 2016

Abstract

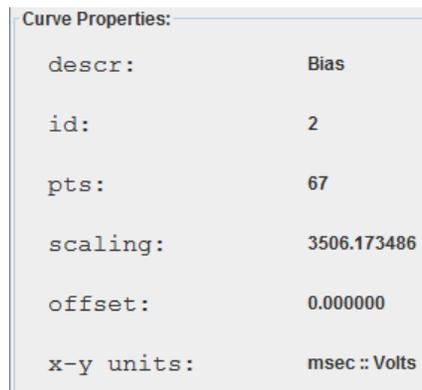
Java console application to load, edit, import, and export shared memory curves to or from a Booster llrf vme-dds board.

1 Introduction

This document should provide enough information about the program so a new user can successfully load, edit, import, and export a shared memory curve to or from the vme-dds board.

2 The basics

A shared memory curve, as defined by the vme-dds board, is a sequence of 1023 possible time-value pairs. Each curve begins with a header that contains a short descriptive name, index, length, scaling, and offset. The desired time-value pairs may be floating point numbers, the internal representation uses scaled integers.



Curve Properties:	
descr:	Bias
id:	2
pts:	67
scaling:	3506.173486
offset:	0.000000
x-y units:	msec : Volts

Figure 1: Shared memory curve header info.

With this in mind, an operational Booster LLRF vme-dds board has the following pre-defined shared memory curves:

descr	id	len	notes
Freq	1	1023	the frequency curve, 37-52 MHz
Bias	2	190	the bias curve 0.5-9 V
Inject	3	4	beginning of interpolated freq curve
Spare_32B	4	1022	not used
POFF1	5	1022	full cycle phase offset curve, channel A
POFF2	6	1022	full cycle phase offset curve, channel B
PARA	7	512	A-B injection paraphase

Users can launch the editor directly from the [Java Controls](#) web launch service or from ACNET via index page **B30**. Upon start-up the program should display the following:

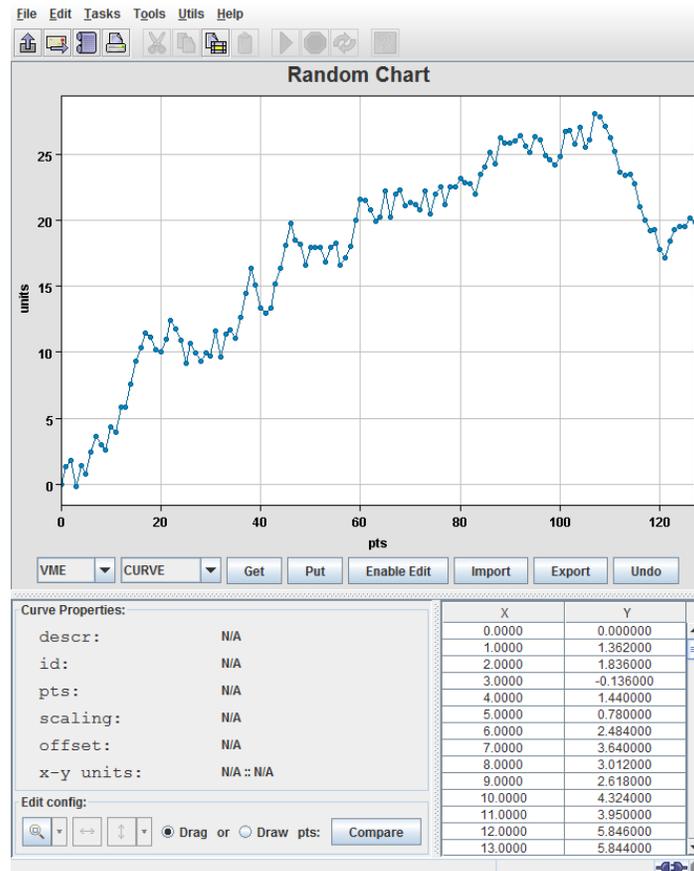


Figure 2: Start-up.

The chart area provides a visual representation of the time-value pairs, the curve properties area contains the curve header info along with the X-Y units, and the adjacent table shows each time-value pair defined for the curve.

The shared memory curve editor supports the following functionality:

- *download a curve from a vme-dds board for inspection*
- *upload a curve to a vme-dds board*
- *import and export a curve to csv file*
- *edit the curve time-value pairs in various ways*

3 Inspection

To download a shared memory curve from a vme-dds board;

1. Select the appropriate VME node with the **VME** pull-down menu.
2. Select the desired curve with the **CURVE** pull-down menu.
3. Download the curve from the selected VME node with the **GET** action button.
4. Upon successful download of the curve the chart area, properties panel, and time-value pair table should update appropriately.
5. Enable the zoom function by clicking on the magnifying glass in the edit config pane.



Figure 3: Download a curve.

4 Editing

The editor uses familiar gui components which allow the user to change the curve time-value pairs. Changing the time-value pairs can be done via the chart area using the mouse or by editing cells in the table. Each series of edits is cached by the program and can be undone.

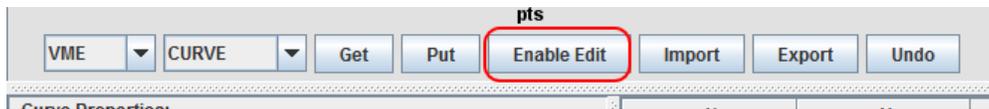


Figure 4: Enable editing of time-value pairs.

To change a table cell directly:

1. Enable editing with the **Enable Edit** action button.
2. Double click (left mouse button) on the desired cell.
3. Enter the desired value.
4. Confirm the change by hitting enter.

Editing the curve time-value pairs via the chart area involves two distinct modes: **Drag** and **Draw**. Each transforms the input of the mouse in a different manner.

In **Drag** mode, the user can select a point or a number of points from the chart area using the left mouse button. Each selected point can then be *dragged* in the X direction or Y direction (default) using the mouse.

To edit the time-value pairs in Drag mode:

1. Enable editing with the **Enable Edit** action button.
2. Select the points to edit.
3. Drag the selected points up or down to the desired value.

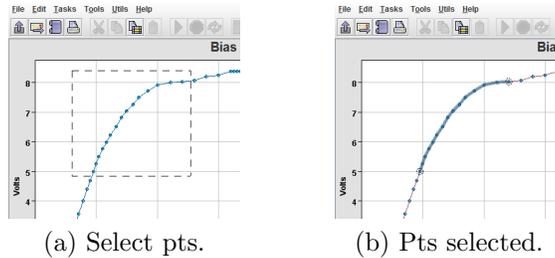


Figure 5: Selecting points in Drag mode.

4. Mouse over the selected points and drag them with the left mouse button.

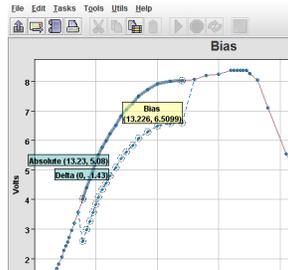


Figure 6: Drag the selected points in Y.

Drag mode supports moving points in the X and Y (default) directions separately. To drag the points around in time, select the \leftrightarrow action button in the Edit Config pane.



Figure 7: Drag the selected points in X.

The default editing mode is **Drag**, to switch to **Draw** mode, select it with the radio button in the Edit Config pane.



Figure 8: Switch mouse input modes.

In **Draw** mode, the Y values of the curve take on the Y values of the mouse position as the pointer moves across the chart area. For example the user can trace out the desired Y values of the curve while holding down the mouse wheel button.

To edit the time-value pairs in Draw mode:

1. Enable editing with the **Enable Edit** action button.
2. Select **Draw** mode using the radio button in the Edit Config pane.
3. Hold down the *mouse wheel button* and draw out the desired path for the Y values of the curve.

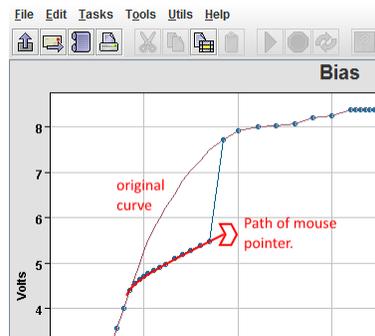


Figure 9: Draw out desired Y values using the mouse.

There may be any number of edits, using the Undo action button will eventually remove all of the cached curve edits.

To undo an edit to the curve time-value pairs:

1. Undo the most recent curve edit with the **Undo** action button.



Figure 10: Undo a change to the time-value pairs.

The **Drag** and **Draw** modes allow the user to change a select number of time-value pairs. While this is useful in making small changes to the curve, there may exist a need to shift the entire curve in X or Y. This can be done using the **Interp** dialog found in the **Utils** menu.



Figure 11: Shift or offset an entire curve.

To shift in time or offset in amplitude:

1. Raise the **Interp** dialog from the **Utils** file menu (a).
2. Enter a \pm **delay** or \pm **offset** into the dialog text box (b).
3. Confirm the edit with the **Delay** and **Offset** action buttons.
4. The **Interp** dialog will accumulate any shifts in time or amplitude offsets made to the curve providing the window remains open.
5. The Undo action button will revert the most recent change in sequence.

The **Smooth** function is experimental at this stage and may not behave as one would expect. It is a work in progress. It should be noted that in order to shift the entire curve in time, the curve has to be linearly interpolated. This process may introduce small differences from the original curve if the time between points is fairly large compared to the amount of shift in X & Y.

5 Importing and Exporting

Curve time-value pairs can be imported from and exported to a comma separated file. The file however has to include a proper header followed by the time-value pairs. An example follows: the line numbers **On>** are for illustration only and need not be included.

```

01> CURVE
02> Bias
03> 2
04> 67
05> 3506.173486
06> 0.000000
07> 0.000000,0.549775
08> 0.008000,0.549775
.
.
.
.
.
.
72> 63.000000,0.549775
73> 65.000000,0.549775

```

Lines **01**→**06** define the curve header and lines **07**→**73** define the time-value pairs of the curve. Line **02** is the short description, **03** the curve ID, **04** the length, **05** the scaling, and **06** the offset.

To import and export a curve from or to a file:

1. To **Import** a curve use the action button.
2. Select the file you want to import with the file Open dialog.
3. To **Export** a curve use the action button.
4. Browse for an appropriate file location with the Save dialog, give it a name.

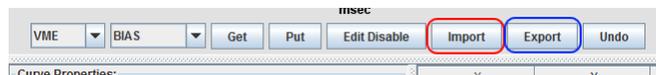


Figure 12: Import or export to a csv file.

Remote file access:

Users should note that if the application is started via ACNET page **B30**, file access is limited to the clx node. This means that any curves that are exported will reside on the clx node. As of now the files are written to a shared folder with the following path:

```
/usr/local/cbs_files/cns_write/brfdds
```

An editor launched via the **Java Controls** web service has local file access only. I intend to localize curve export to a Controls database to bridge the gap between local and remote file access; this is a work in progress. Once the database has been setup there will be no differences in file access between applications.

5.1 Directory Structure

The *brfdds* directory on the *clx* node has the following structure. Curves exported to the *ops* subdirectory should contain copies of and or save files of operational curves that can be used as a restore point following a shutdown or a power outage.

Corruption of the non-volatile fpga memory should be considered a rare event but is not out of the question, having proper files to restore from will make recovering a non-issue. Exporting the curve prior to editing makes for an easy restore point as well.

```
\brfdds
  |--\ops
  |   |
  |   +--> "operational curves", secret sauce goes here
  |--\cache
  |   |
  |   +--> "brfdds-log.txt" and cached curves
  |--\debug
  |   |
  |   +--> "may be useful, not for operations"
  |--\testing
  |   |
  |   +--> "user beware"
  |--\vxi
  |   |
  |   +--> "current copies of the vxi curves"
```

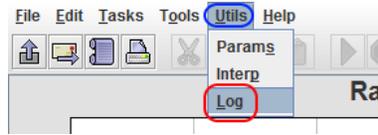
Each time a curve is uploaded to an operational vme node ie. **BLRFDS**, a copy of the curve is exported with a timestamp to the *cache* subdirectory. This will create a digital history of which curves were uploaded and when.

An entry into the application log file will be made for each upload, import, and export operation detailing who, when, and from where the operation took place.

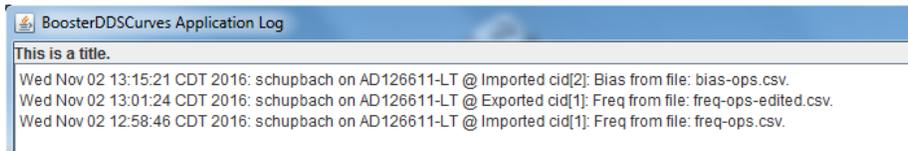
It should be noted that the vme front-end does not keep track of any file information associated with a given curve and that a curve that has been uploaded to the vme may not have an operational save file associated with it.

To view the application log file:

1. Raise the **Log** dialog from the **Utils** file menu (a).
2. A window with a timestamped list of log items should appear.



(a) Utils file menu



(b) Log file window

Figure 13: Display Log file.

6 Comparing curves

To compare a curve that has been loaded into the application buffer (from a vme or file import) with another file on disk, use the **Compare** action button from the Edit Config pane. A file open dialog will appear, select the file you wish to compare to.



Figure 14: Compare the current curve buffer to a file.

A new chart window will appear with each curve and the difference between them. The chart has two Y-scales, the left gives positive and negative difference while the right is in units appropriate to the curves of interest. You can zoom in to inspect each curve if need be.

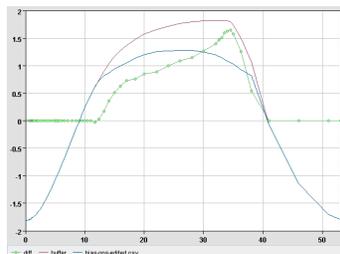


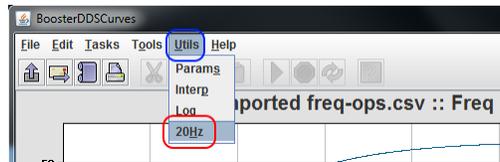
Figure 15: Curve compare chart.

7 20 Hz Curves

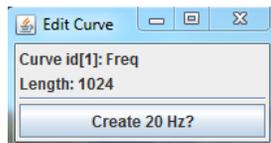
Running the Booster at 20 Hz to increase throughput may become a reality in the near future. To facilitate the testing and debugging process for the LLRF system, I've added a function to create 20 Hz curves from an existing 15 Hz curve using a linear scaling. This will decrease the time steps of the 15 Hz curve by %75.

To create a 20 Hz curve from an existing 15 Hz curve:

1. Download or import the existing 15 Hz curve into the curve buffer.
2. Raise the **20 Hz** dialog window from the **Utils** file menu (a).
3. The **Create 20 Hz?** action button in the dialog window will create a new 20 Hz curve. (b)



(a) Utils file menu



(b) dialog window

Figure 16: Create a 20 Hz curve for testing.

Any edits to the new 20 Hz curve may be undone using the **Undo** button as before. Comparing a 20 Hz and 15 Hz curve is not supported as they are now of different time scales.

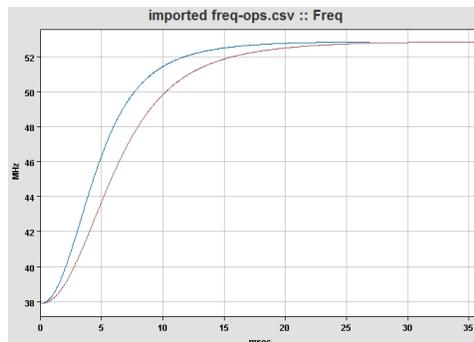


Figure 17: A new 20 Hz freq curve.

8 Conclusion

The vme-dds curve editor allows one to inspect, upload/download, edit, compare, and import/export curves using familiar gui based interactions. Any changes made to a curve are cached so edits can be easily redacted. The application is available via ACNET page **B30** and the [Java Controls](#) web launch service. Comments, suggestions, and rants will be considered.