

4/30/2018

Written by Jordan Stomps

I. Launching SpecTcl

Launch SpecTcl by typing the following command from the `~/readout/SpecTcl/` file location:

- `./Launch_SpecTcl`

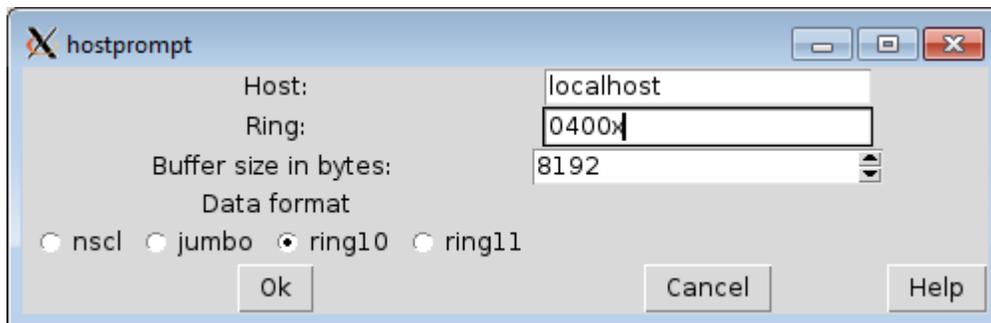
II. Setting-up Online Data Collection

Once SpecTcl has initialized and opened, you can load the preset channel parameters by selecting `Load` in the top-right of the `treegui` window and then selecting `all_ddas.tcl`. You can then press `Ok` and the parameters will populate SpecTcl.

Next, execute the following actions to attach SpecTcl to Readout (**Important:** Remember to launch and setup Readout **before** attaching SpecTcl online. This way it knows what to attach to and what data is being forwarded):

```
Data Source... -> Online... -> Host: localhost -> Ring: 0400x ->
Select "Ring 11" -> Ok
```

Note: Depending on where you are running your terminal from, `localhost` may not be correct. If you are running *on spdaq01*, use `"localhost."` If you are running from any other location, use `"spdaq01"` under `Host:` instead.



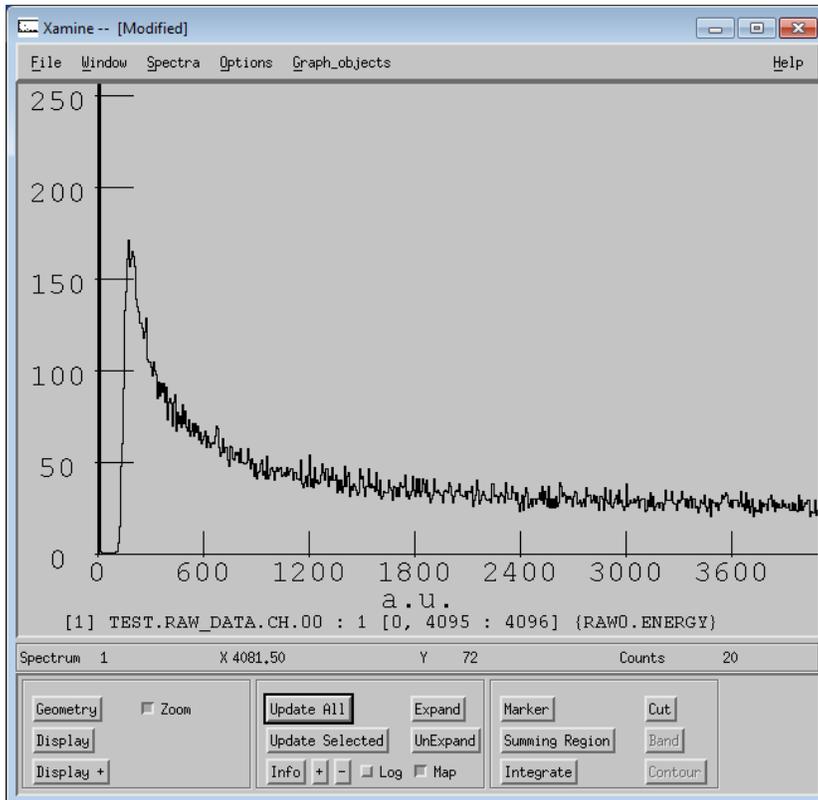
OPTIONAL: If you are unsure whether SpecTcl is actually collecting data, you can see in the bottom right a running total of buffers that have been analyzed. Or, you can open a terminal window and type:

```
$DAQBIN/dumper -source=tcp://localhost/ring
```

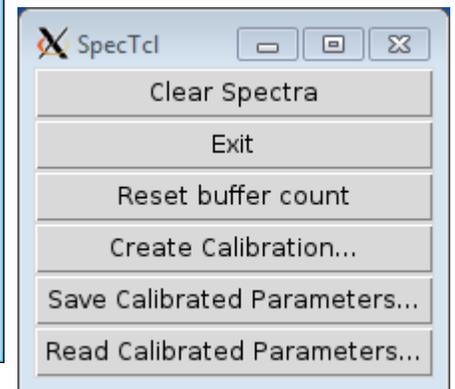
If nothing appears, then Readout is not collecting data. If something does appear, you can press `Ctrl + C` to end the data dump.

III. Displaying Data

SpecTcl allows you to display one channel/parameter at a time. First open the GUI window named *Xamine*. In the bottom left select *Display +*. Here you can select the channel to be displayed and press *Apply* and *Ok*.



Important: Remember that this is a static picture of the data. If you want to see the spectrum updated, you must press *Update All* (unless Section IV is followed).

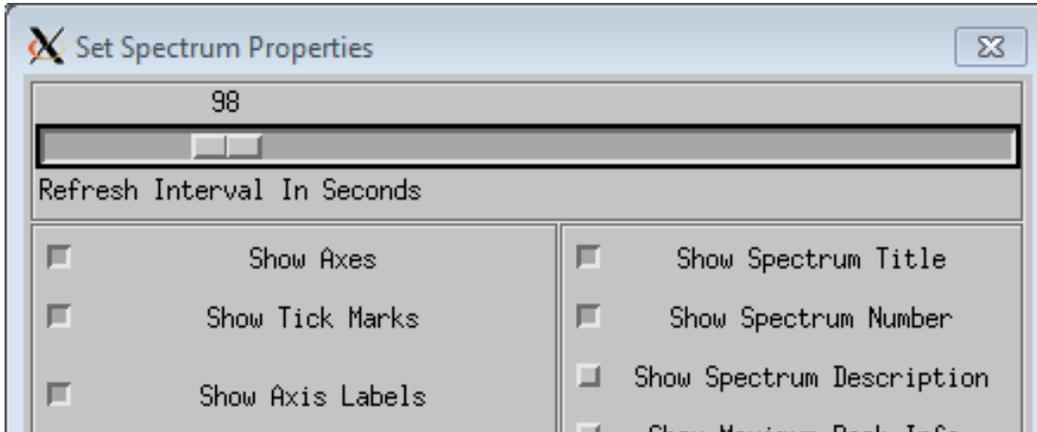


If you want to scale the y-axis up or down, you can press *Alt + -* to increase the y-axis and press *Alt + +* to decrease the y-axis.

To display **multiple spectra**, select the *Geometry* button in the bottom left and select the number of rows and columns. You can then display a spectrum in each box.

IV. Adding a Refresh Interval to Xamine

To avoid pressing *Update All* every time an update is needed, you can set a refresh interval to the spectrum in question in Xamine. Simply go to *Spectra -> Properties...* and then move the slider at the top of the page to the desired seconds of interval. Then press *Ok*.



V. Writing a Summing Gate

If you want to take multiple DDAS channels and histogram their incremented events together, you must type this into the SpecTcl command terminal using the following commands:

- `spectrum name g1 {channel-parameters} {{low high bins}} (1)`
- `sbind -all (2)`

Here is a breakdown of the commands:

- (1) This creates a new spectrum called *name*. It is a gamma 1-D spectrum (*g1*). *Channel-parameters* must be listed the same as they appear in the GUI of all desired channels included in the summing gate (e.g. `{raw0.energy raw1.energy raw2.energy raw3.energy}`). The last portion is the limits for graphing.
- (2) This command binds the spectrum to Xamine so it can be visualized.

From here you can select the spectrum in Xamine when you press *Display* or *Display+*.

VI. Writing a Coincident Gate

If you want to make a gate so that only events for a channel are plotted if they occurred when another channel also fired, you use the following steps:

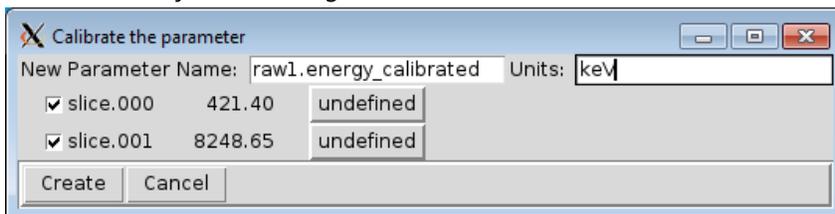
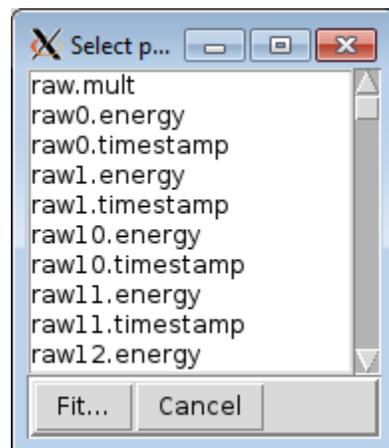
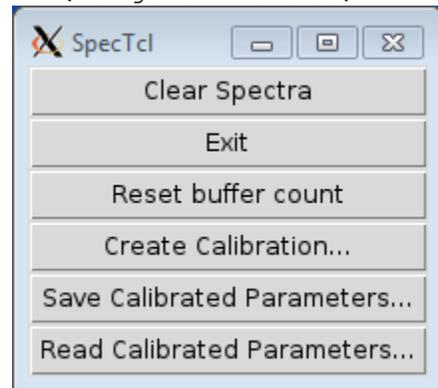
- Select the *Gates* tab in SpecTcl.
- Type a name for the slice of the coincident gate (e.g. `slice.ch1`).
- Select *slice* as the Gate type.
- Type the gate info in the following format: `parameter-name {range-of-values}`.
 - If we were writing a coincident gate for channel 3 when channel 1 fired, it may be `raw1.energy {0 65536}`.
- Go to the *Parameter* tab of SpecTcl.
- Double-Click the DDAS channel that this gate will be applied to (e.g. `raw3.energy`).
- Select the Slice Gate in the *Gates* drop-down menu.
- Select *Apply*.

- Update or Display the graph in Xamine.

VII. Writing a Linear Energy Calibration

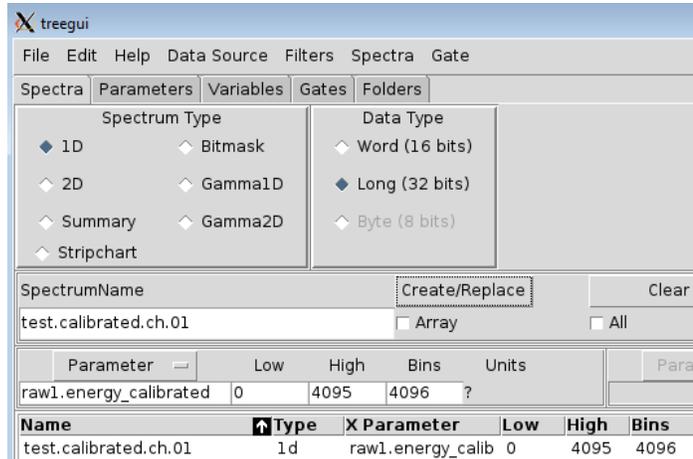
You can apply a linear least squares fit energy calibration to a DDAS channel by using the following steps:

- First make slices of the all the peaks you identify in that channel/parameter:
 - Select the *Gates* tab in SpecTcl.
 - Type a name for the slice of the coincident gate (e.g. slice.ch1.000).
 - Select *slice* as the Gate type.
 - Type the gate info as follows: `parameter-name {range-of-values}`.
 - In our example, it may be `raw1.energy {0 1000}` where the bounds are the approximate range of the peak (no need to be exact).
 - Repeat until all peaks desired have been identified.
- Open up the window on the right (may need to be resized) and select *Create Calibration...*
- Select the channel/parameter by which the calibration is to be calculated for and select *Fit...*
- Now select all peaks desired to be included in the fit calibration. You can name the new parameter that is created and define the units. Make sure to select the areas labelled *Undefined* and select a corresponding energy value for that peak. Then click *Create*. NOTE: If the correct values have not been included in the drop-down menu, refer to the section labelled *VIII. Making a Table of Reference Energies*.



- Make sure you remember the Parameter name from this window prompt, because you have to enter that manually into SpecTcl (if you forget it, you may be able to find it by typing `parameter -list` in the SpecTcl console). Use the following commands:

- Type a name for the spectrum under *SpectrumName*.
- Type the explicit name of the parameter you created (see above).
- Type the low and high limits as well as the bins for viewing (these now correspond to your new units you created).
- Select Create/Replace.
- Finally, open Xamine and plot the new calibrated parameter (use the instructions found in *Displaying Data*). As new data is collected, this histogram will be populated based on the new calibration.



VIII. Making a Table of Reference Energies

In order to have the correct values available in the dropdown menu for creating an energy calibration, the values have to be manually entered using the following steps:

- Make sure you are in the `~/readout/SpecTcl` directory and type `./Edit_Calibration_Points`.
- This will launch a text editor allowing you to add or delete values for the dropdown menu.
- Type in values as needed, one numeral value per line and nothing else.
 - Note: Keep units in mind. If one peak corresponds to 1 MeV, but you are doing calibration units in KeV, make sure you type 1000 as your value.
- Make sure to save your edits on the file and close the program.
- If you are in SpecTcl, you can now press *Refresh Calibration Points*. NOTE: This currently copies all values that were originally written in the calibration from before, so you may see duplicates.
- You can now continue with your calibration in SpecTcl.

IX. Saving and Loading Parameters and Gates

Save time setting up SpecTcl by saving your parameters and gates before you close the program:

If you have executed any calibrations that you would like to save, follow these steps:

- Select *Save Calibrated Parameters* in the SpecTcl Command Window.
- Type a name for your file you are saving (making sure it has a .tcl ending).
- Select *Save* on the default location.

You can load these parameters using these steps:

- Select *Read Calibrated Parameters* in the SpecTcl Command Window.
- Select the .tcl definitions files you want to use.

- Click *Open*.

If you don't have any calibrations, follow these steps:

- Select *Save* under the *Definition file*: section of the main SpecTcl GUI.
- Type a name for your file you are saving (making sure it has a .tcl ending).
- Select *Save* on the default location.

Next time you want to use these parameter definitions:

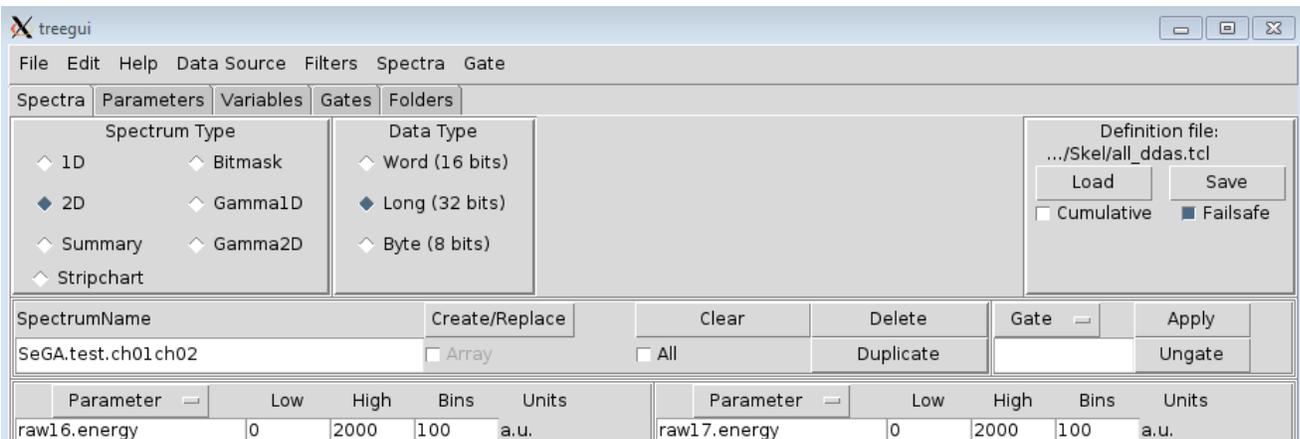
- Select *Load* in the same area of the main SpecTcl GUI.
- Select the .tcl definitions files you want to use.
- Click *Open*.

These steps of saving should save any gates you have written. If they don't show up in the *Gates* tab after loading the file, select *Update Gate List* in the bottom-left of the window. If they still do not appear, you will have to enter the values manually.

X. Writing a 2D Spectrum

To create a parameter in SpecTcl that plots to channels against each other (2D spectrum), use the following steps:

- Select *2D* in the *Spectrum Type* box of the treegui.
- Type a name for your spectrum in the *SpectrumName* box (In this example that may be "SeGA.test.ch01ch02.")
- Select the appropriate channels from the *Parameter* dropdown. If you need reference for what SpecTcl parameters go with what channels, see the Appendix or the parameters provided in the *all_ddas.tcl* definitions file (what you probably loaded at the start).
- Double check that you change the low, high, and bins limits to accurately reflect what you are trying to see.
- Press *Create/Replace*.
- You can then display this new spectrum in Xamine.



Appendix

- NOTE: If you ever run into the situation of clicking a SpecTcl button and then not being able to press anything else, press *Esc* and then you should be able to operate the GUI again.
- For information regarding the calibration plugin, see here: <http://docs.nscl.msu.edu/daq/specplugins/calibparams.pdf>
- Full documentation for SpecTcl, DDAS, and NSCLDAQ can be found here: <http://docs.nscl.msu.edu/daq/index.php>
- The following map of parameters was written by Tamas Budner showing how each detector parameter relates to each DDAS/DAQ channel:

Module	DDAS channel	Source channel	
0	0	A	Proton Detector
0	1	B	SeGA
0	2	C	
0	3	D	
0	4	E	
0	5	F	
0	6	G	
0	7	H	
0	8	I	
0	9	J	
0	10	K	
0	11	L	
0	12	M	
0	13	Gating grid	
0	14	Gating grid	
0	15		16
1	0		1
1	1		2
1	2		3
1	3		4
1	4		5
1	5		6
1	6		7
1	7		8
1	8		9
1	9		10
1	10		11
1	11		12
1	12		13
1	13		14
1	14		15
1	15		
2	0	Silicon detector	