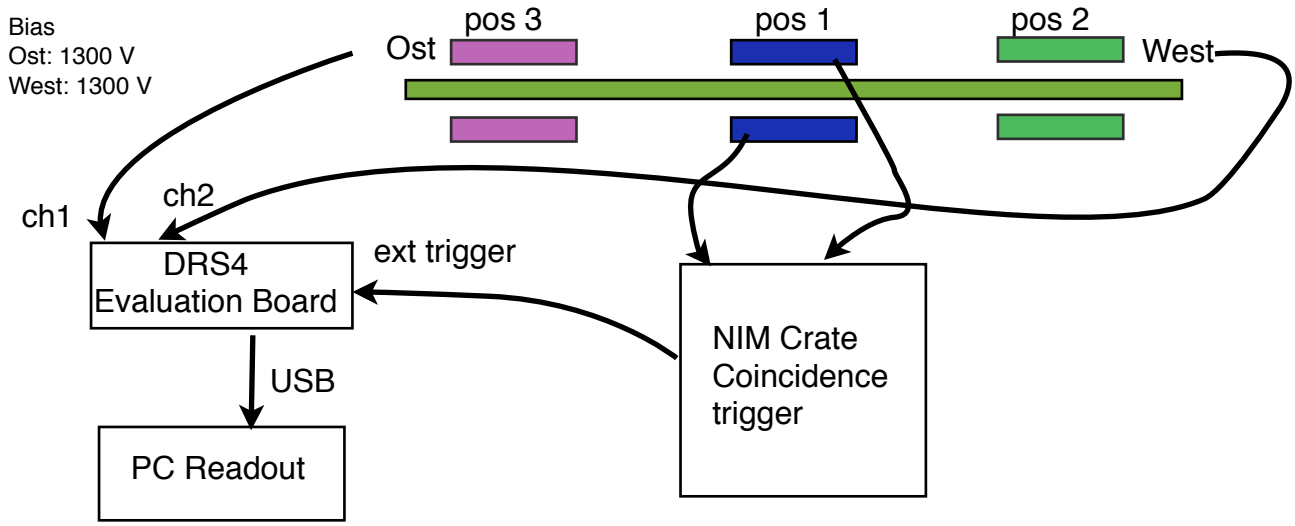
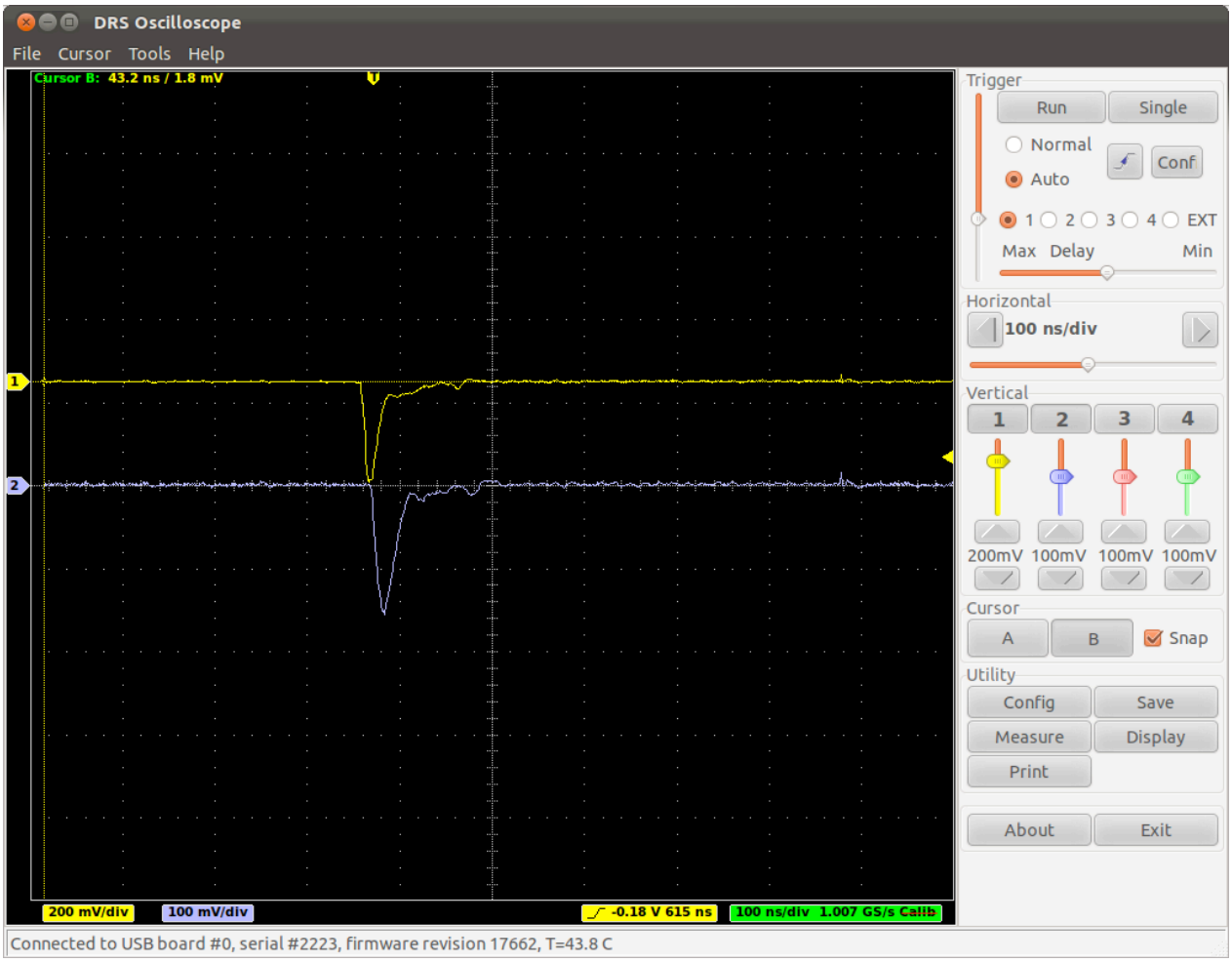
Status of DRS4 Digitizer Tests at KIT. 16 December 2011

Adam Cox - adam.cox@kit.edu

Section 1. Setup and initial tests of operating the DRS4 and data extraction

The test bench set up at KIT includes a ~3-m plastic scintillator “Edelweiss” panel and two smaller scintillator panels to act as a coincidence trigger. At each end of the Edelweiss panel are two PMT groups, labeled Ost and West, each with a summed signal output that are recorded by the DRS4.

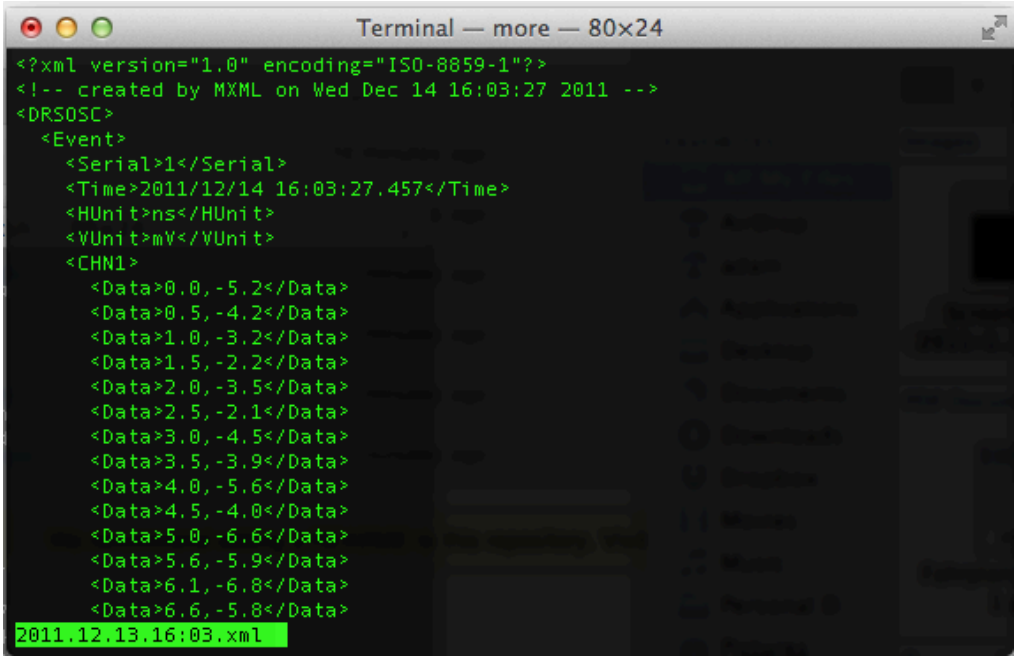


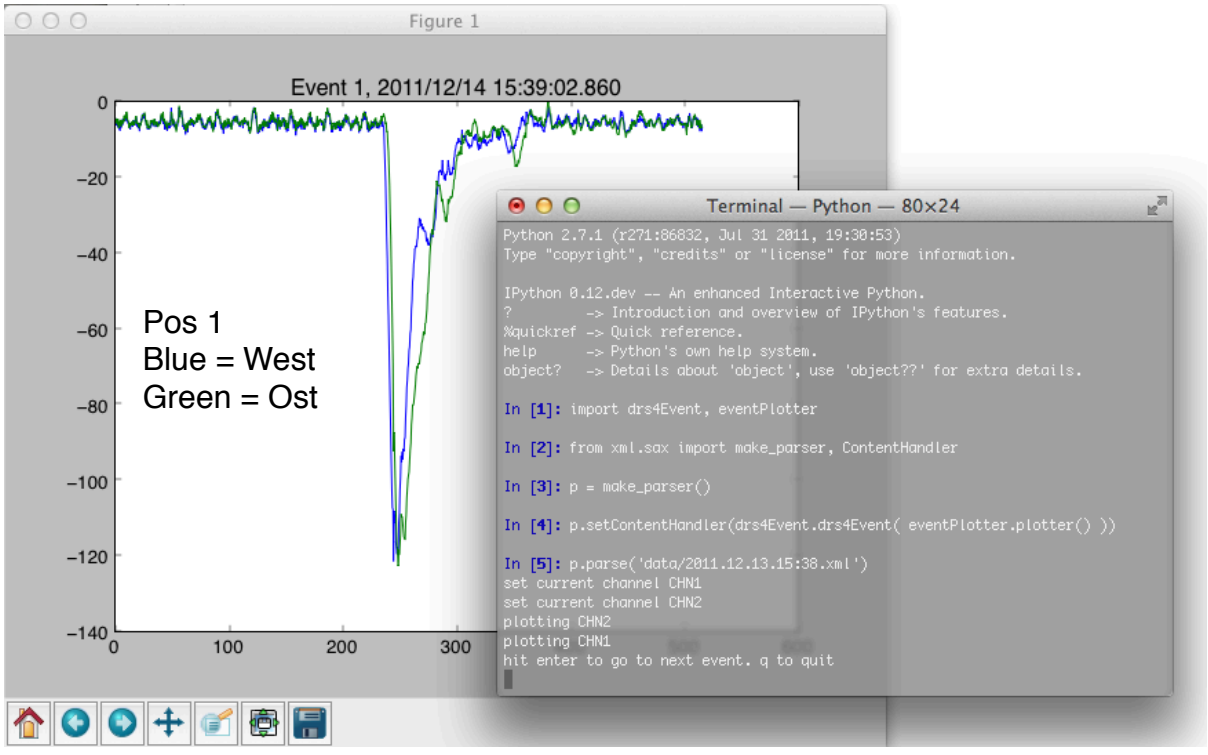


The PC readout software was supplied by the PSI group that produces the DRS4 chip. This allows one to view the output of the DRS4 in “oscilloscope” mode and to save the triggered data to disk in a binary or XML format. While XML data files are significantly larger than binary data files, they are programmatically easier to read. Some Python code was written to read these XML data. They are currently located at the following public repository: https://github.com/gadamc/drs4

The image above is a screenshot of the PSI group’s software. The software was installed on a PC running the Ubuntu 11.10 linux distribution.

The next screenshot shows the data in XML format.

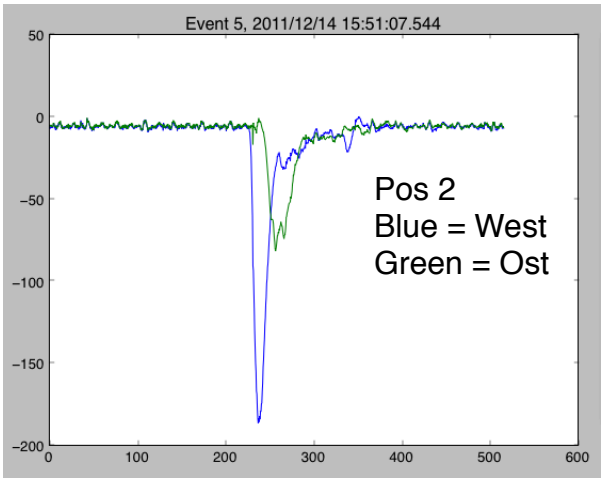


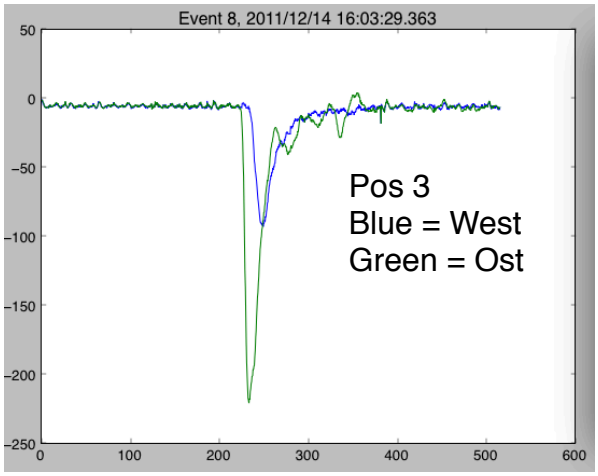


Using the Python code, one can scan through the data, event by event. Once you've downloaded the code, open a terminal in the directory where the code is located. Start a Python shell and issue the following commands:

```
In [1]: import drs4Event, eventPlotter

In [2]: from xml.sax import make_parser

In [3]: p = make_parser()

In [4]: p.setContentHandler(drs4Event.drs4Event( eventPlotter.plotter() ))

In [5]: p.parse('/path/to/data//2011.12.13.15:38.xml')
```

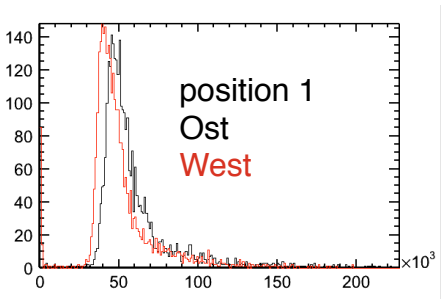
Data was taken with the coincidence scintillators placed in three different locations about the Edelweiss scintillator. The position and amplitude of the pulse peaks shift in time accordingly.

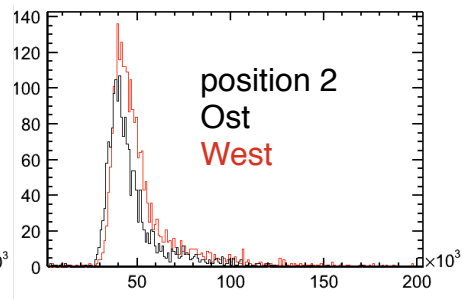


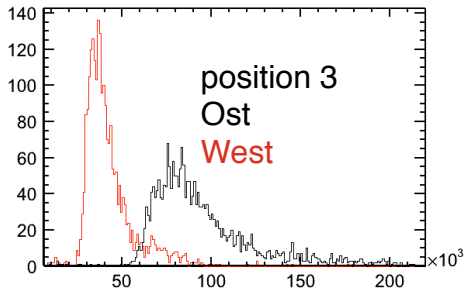


In addition to displaying the pulses, code was written to analyze each event with a trapezoidal filter to estimate the amplitude. This is performed by the drs4Analyzer class found in the dataAnalysis.py file.

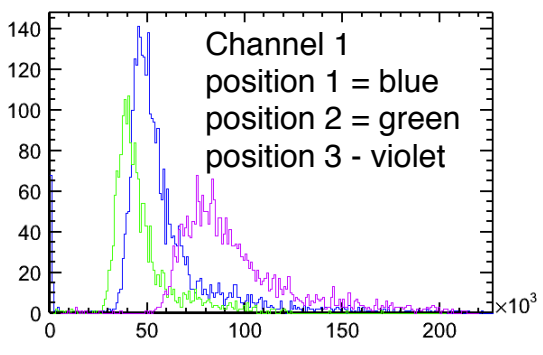
The energy spectra distributions are found in the following figures. The Ost channel is in black and the West channel is in red (sorry for changing the color scheme!). The energy spectra for each channel at the three different positions of the trigger modules are also shown: position 1 is in blue, position 2 in green and position 3 in violet. The distribution of

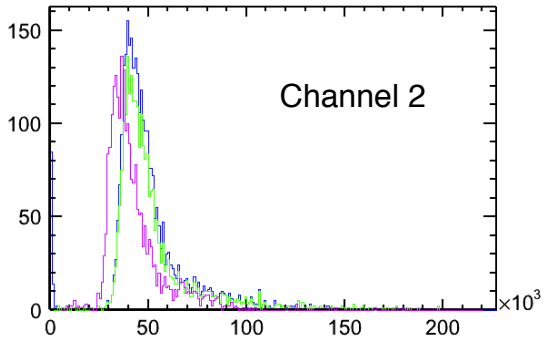


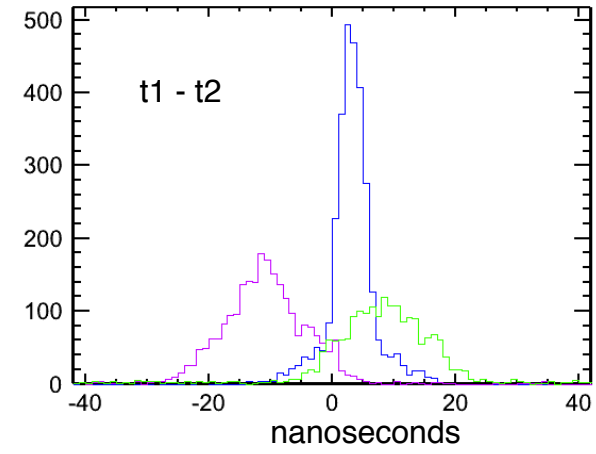




the difference between the peak position times (channel 1 - channel 2) are shown in the last plot with the same color scheme for each position of the trigger modules.







Section 2 - Tests of the DRS4

Current Problems:

Some events are not readout properly by the XML Python code. The python code, for an unknown reason, doesn't find both x,y values for a point in the event. Instead it only sees one value. Those events are skipped in the analysis.

The sampling rate is not necessarily a rational number. As a result, in the data the value for the time of each event does not appear to be consistently spaced. However, I believe that this is most likely a rounding error. I assume in the analysis that each data point is equally spaced in time.