Working With Cosmic Rays

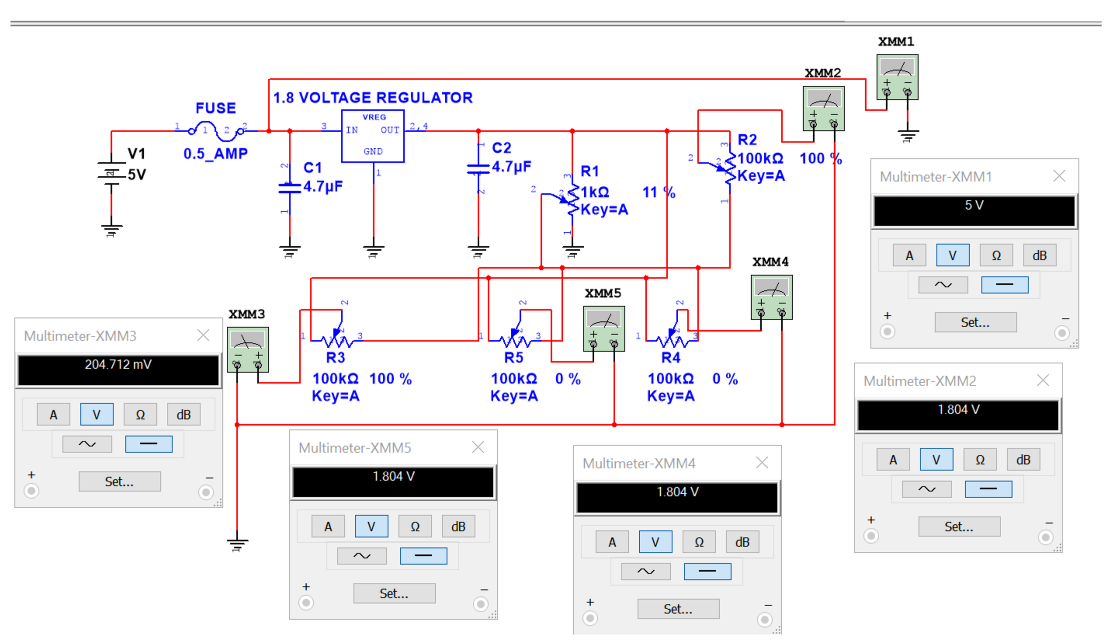
Robert Li, Computer Engineering Technology, Queensborough Community College, Bayside, NY
Jude Okocha, Electrical Engineering Technology, Queensborough Community College, Bayside, NY
Reginald Lesane, Dept. of Physics, St. John's University, Jamaica, NY
Raul Armendariz PhD, Dept. of Physics, Queensborough Community College, Bayside, NY
Daniel Garbin PhD, Dept. of Math, Queensborough Community College, Bayside, NY

QCC Undergraduate Research Day Conference, Dec. 8th, 2017, Queensborough Community College

Building a power distribution unit (PDU)

A power distribution unit is an electrical circuit including a voltage divider that generates power to different electronic devices such as photomultiplier tubes (PMTs) . The PDU contains a voltage divider that takes in 5V, drops 0.2mv – 0.25mv, and provides 1.8V; four variable potentiometers are used to provide 0.2V to 1.8V to power up to 4 PMTs. Each PMT module has an internal low voltage DC to high voltage DC converter which amplies the voltage 1V:1kV.

Results of Multisim simulation of pdu variable/regulated voltages



The various multimeters shows voltage ouputs from different potentiometers.

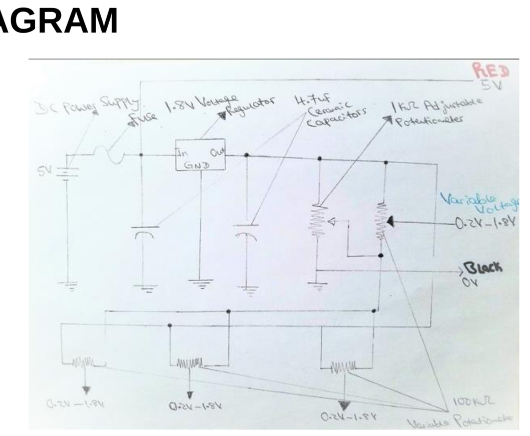
It shows the range of 0.2V to 1.8V, 5V from the power supply and ground can be measured by connecting a wire to the central ground of the entire circuit.

MATH/WIRING DIAGRAM

$$VOLTAGE\ RANGE = \left(\frac{R2}{R1+R2} \times Vreg\right) - Vdeducted$$

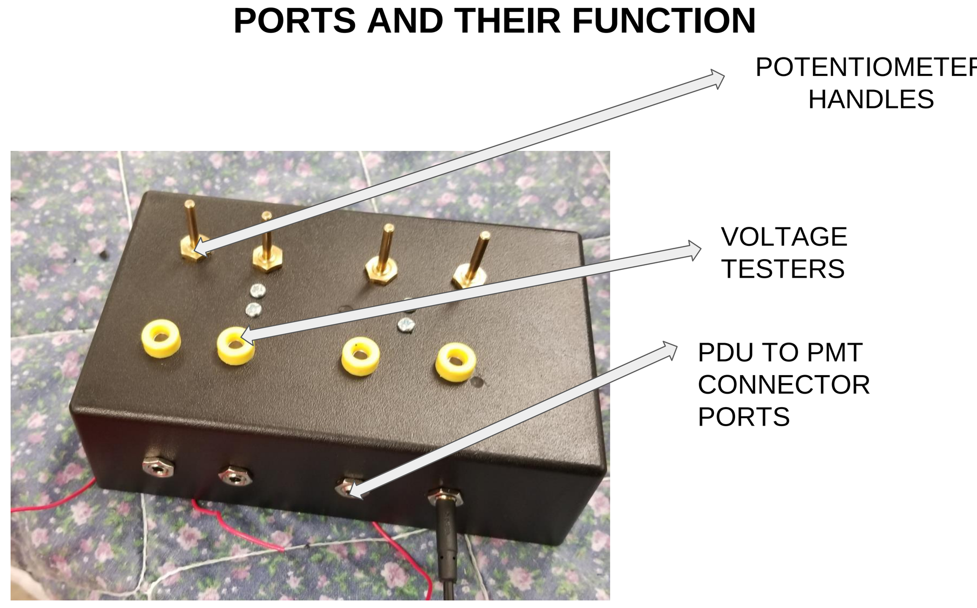
$$\left(\frac{100k\Omega}{1k\Omega + 100k\Omega} \times 1.8V\right) - 0.2V = 0.6V$$

$$1.8V - 0.2V = 0.6V$$

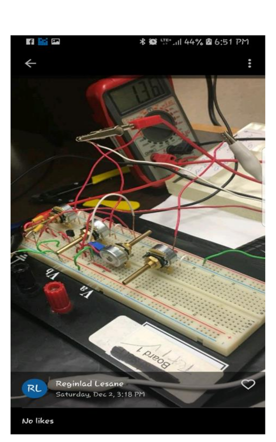
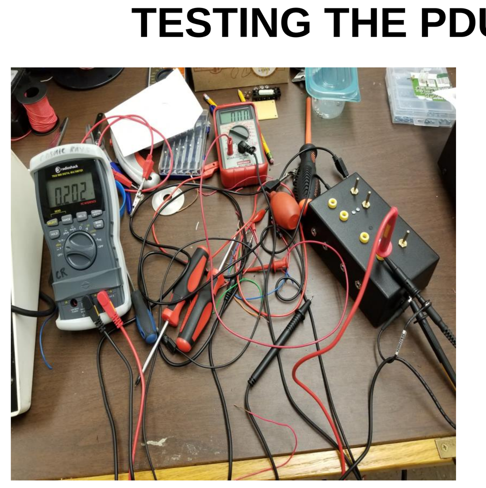
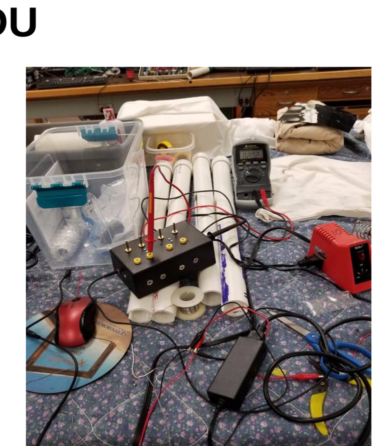


The range voltage range here is calculated using the voltage divider rule (VDR) and basic math

PORTS AND THEIR FUNCTION



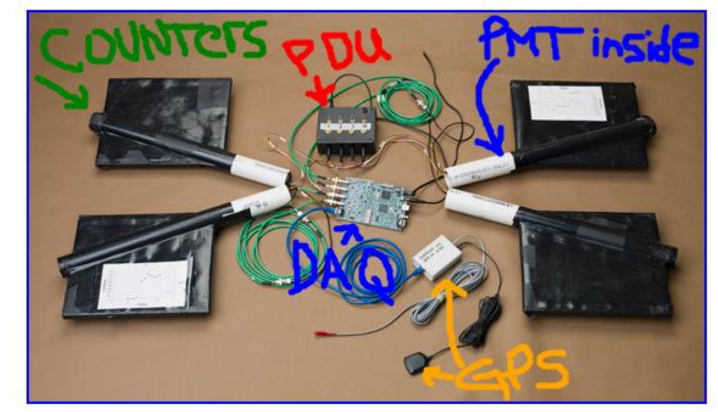
TESTING THE PDU

The PDU is being tested here between its highest range and lowest range. The lowest voltage shown here is displayed on the true RMS multimeter which is approximately what we needed. The value shown on the other multiplier is the peak voltage of the variable voltage; which is approximately 1.8V.

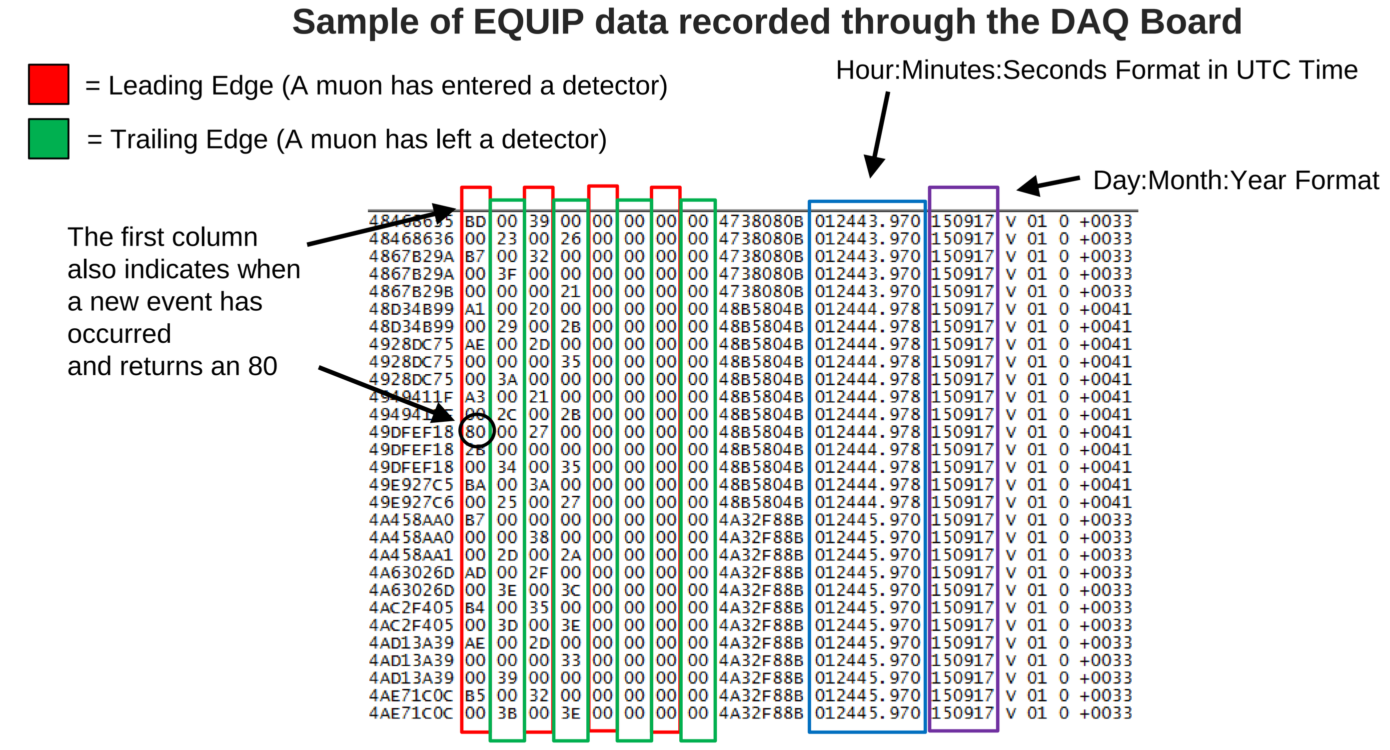
Writing Python software to analyze Quark Net DAQ cosmic ray data

QuarkNetter Reports: The Fermilab Detector

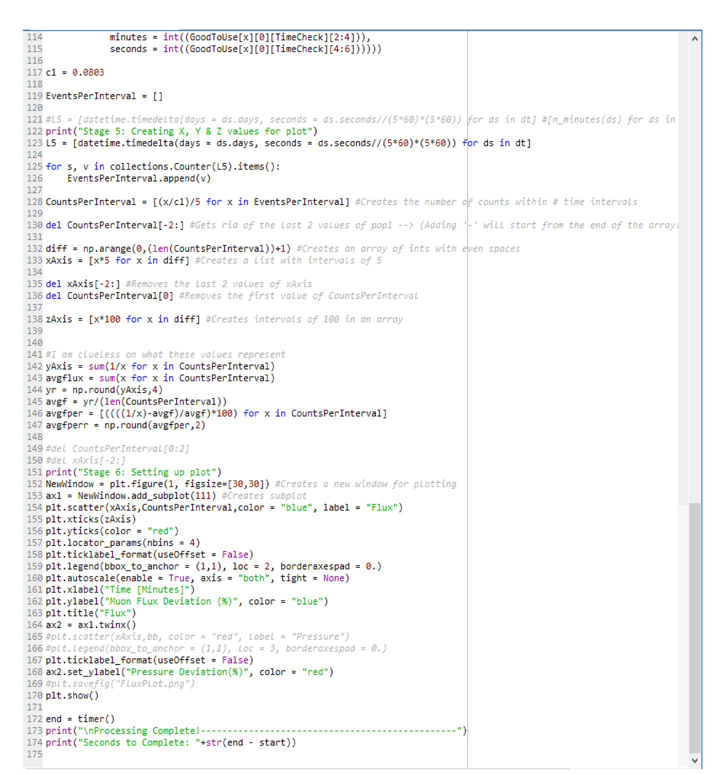
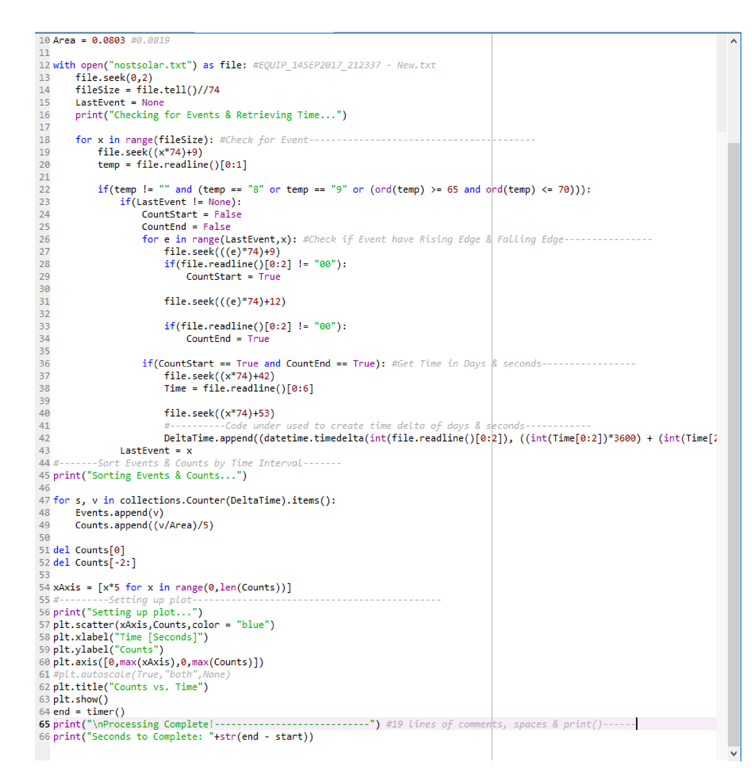


- Our cosmic ray detectors use the QuarkNet DAQ board with hexadecimal data output.
- The DAQ reads a GPS satellite receiver to time stamp each particle detection with a UTC time to about a millisecond resolution.
- The DAQ data consists of up to 4 photomultiplier (PMT) outputs.
- Each PMT output is pulse leading edge and trailing edge time of occurrence; the DAQ can distinguish time up to 1.25 nsec resolution.

Sample of EQUIP data recorded through the DAQ Board



Old Program vs. Optimized Program

-This code is rewritten from a Version 1 pre-existing program written by my colleague David Buitrago to provide a better understanding of plotting the data.

-Version 1 can display other data such as atmospheric pressure and % Deviation.

-Currently the program will retrieve the # of counts over time by looking at the trailing edge and leading edge from the given file and produce a scatter plot in 5-minute intervals.

-The program takes time to generate a plot when loading text files into tables that average around 200MB or more. This is also due to some of the code that creates unnecessary lists with repeated data and take up memory.

-Using a 230MB Equip file, the average time it takes to generate a plot is around 30 minutes.

-With the help of Dr. D. Garbin, more efficient methods were implemented into the new Program with less than 50 lines of code compared To the previous program, which had 100-150 lines.

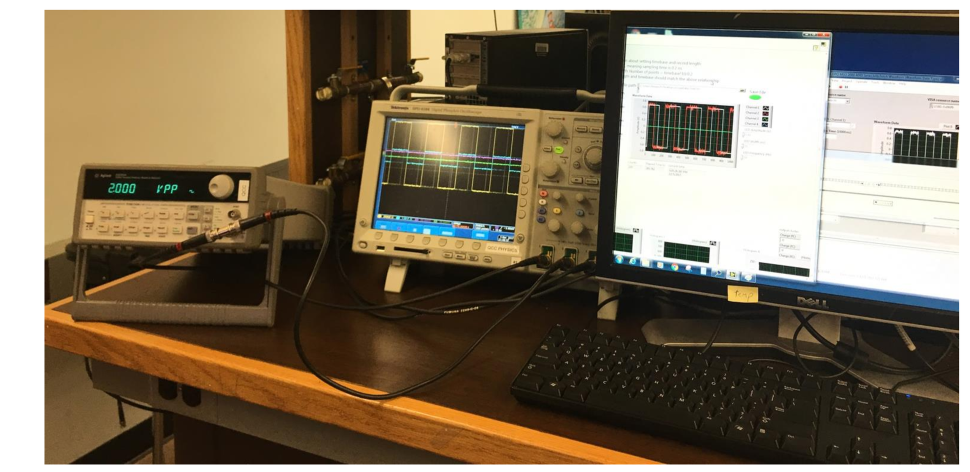
-Rather than storing all the data into multiple lists and taking up memory with repeated data, the idea is to only read and take specific columns of the file in one loop.

-Only three lists are declared and used – Counts, Events & Delta Time. Only one copy of each data is stored.

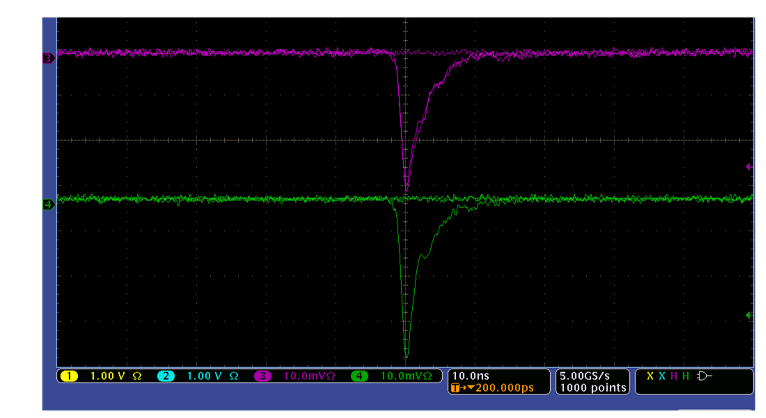
-Since the EQUIP file will have a fixed structure of columns and rows, the Seek() function will be used to reposition where the compiler will begin reading lines of data and skip redundant data, reducing memory usage.

-Using the same 230MB file used in the previous Program, the average time to generate a plot is 7 minutes.

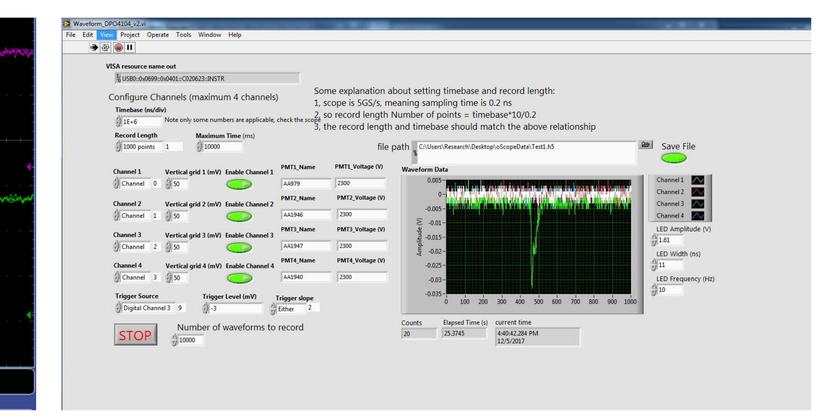
LabView as a Data Acquisition System from an Oscilloscope



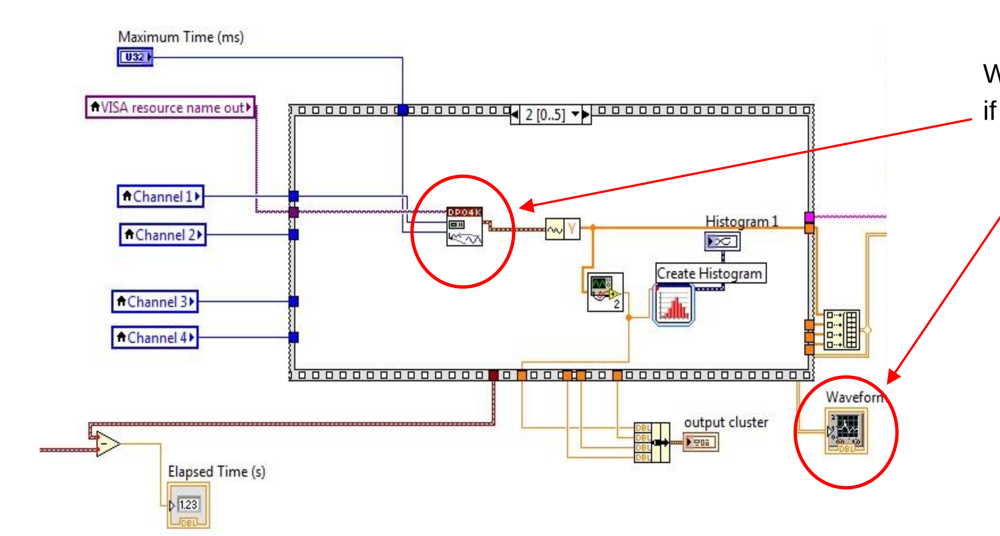
We're using a Labview program to acquire signals from an oscilloscope to test photomultiplier tubes (PMTs). The LabView program written by Dr. Zhang from BNL retrieves signals at a 1Hz rate. We're trying to make this faster. To do this we are using a function generator to send a signal into a DPO4104 oscilloscope, and LabView is installed on a PC to control the oscilloscope and acquire the signals.



This is a DPO screenshot of two PMT noise signals coinciding >50ns. Both signals are captured in Run/Stop mode.



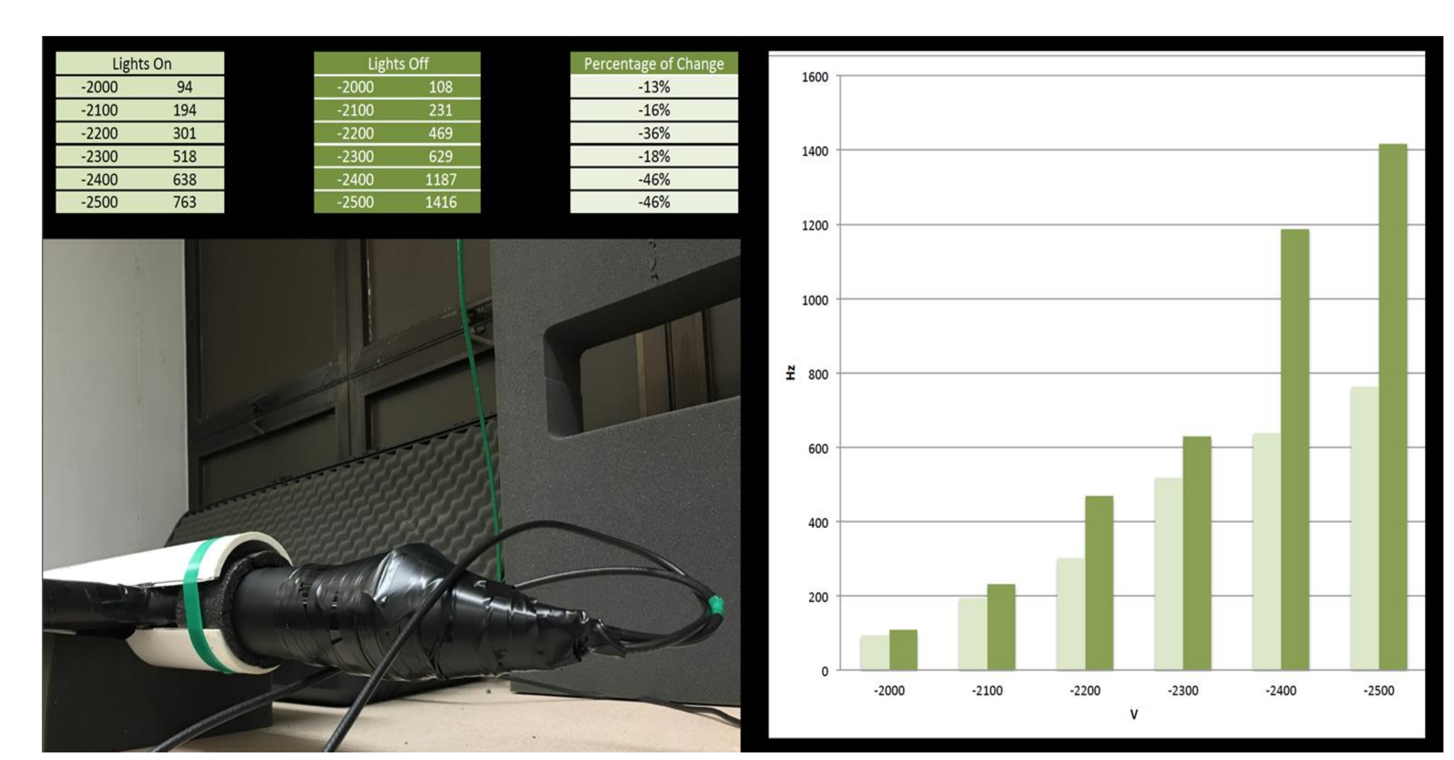
This is a LabView screenshot of two PMT noise signals coinciding >50ns. However the signals are captured in Single mode which displays.



We are going to modify the DPO SubVI to see if it will acquire more waveforms per second.

This is the LabView block diagram showing SubVI's, clusters, waveform charts and nodes. These components define how LabView communicates with the oscilloscope and downloads data. Also a National Instrument technician is assisting us.

Building a Cosmic Ray Muon Detector and Sealing it from Ambient Light



Lights On	
-2000	94
-2100	194
-2200	301
-2300	518
-2400	638
-2500	763

Lights Off	
-2000	108
-2100	231
-2200	469
-2300	629
-2400	1187
-2500	1416

Percentage of Change
-13%
-16%
-36%
-18%
-46%
-46%

A PMT and scintillator need to be wrapped light proof to detect reliable signals. A Lights On/ Lights Off test determine if light interferes with PMT noise signals. This is tested by turning room lights on and off to find the percentage of change in PMT noise signals. This chart has a small percent of change which indicates the equipment is light proof.