Defining the DAQ System for the DS20k Veto.

Ashlea Kemp; Queen's University

The DAQ System for the Veto.

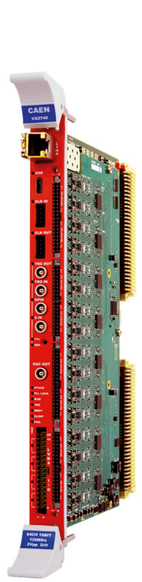
Current Scheme.

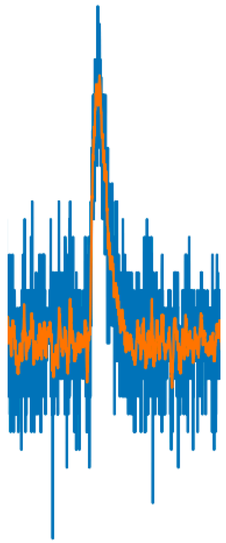
The current scheme being studied for the Veto DAQ System is the following:

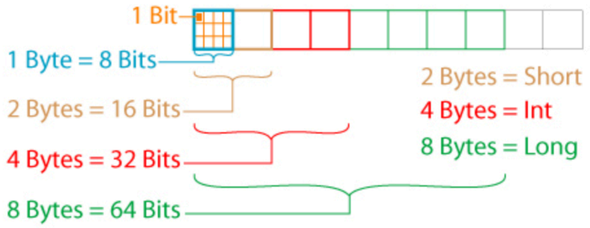
1) DAW/ZLE segments from full waveform transferred from VX to FEP.

2) Hit finding performed (per channel basis) at the FEP. Hit information (charge, prominence, time) transferred to TSP.

3) More sophisticated pulse analysis performed at TSP using hit information across full detector. Pulse information and/or hit information saved as Veto event.







Event Reconstruction, Pulse Analysis, Data Written to Disc.

VX

FEP

TSP

The DAQ System for the Veto.

Current Scheme - Requirements.

🔑 VX-Level
FEP-Level
TSP-Level

Several questions need to be addressed in order to verify whether this scheme is feasible for the Veto.

The following studies will be performed using G4DS simulations and pyreco to answer such questions:

1) What is the average reduction in data transfer between VX and FEP if only sending DAW/ZLE segments, not full WFs - is this within bandwidth limits?

2) What is the single PE detection efficiency for hit finder as a function of threshold/ SNR (noise level), filtering?

3) What is the fake hit rate as a function of threshold/ SNR (noise level), filtering?

4) Average reduction in data transfer from FEP to TSP if sending hit information only - is this within bandwidth limits?

5) Can we perform all the hit finding on the FEP within the available CPU?

6) What is the performance of the reconstruction of a full Veto window with hit information only? Do we still retain all of the physics for accurate energy reconstruction?

7) Can we perform additional pulse analysis at that can be saved as Veto event online, rather than saving all hits and reconstruction offline? Is this within CPU limits at the TSP?

Hit Finding in Pyreco: FEP-Level

For “Veto” Waveforms

Inner Veto is not properly simulated in pyreco *yet*, but we can still simulate single waveforms with the same response (ASIC) as the vPDMs.

Since ARMA filter is not designed for ASIC, and timing is not so important in the Veto, one can test simpler and faster algorithms such as a running average (RA).

Can use RA filter and hit finding algorithm currently implemented in pyreco to test:

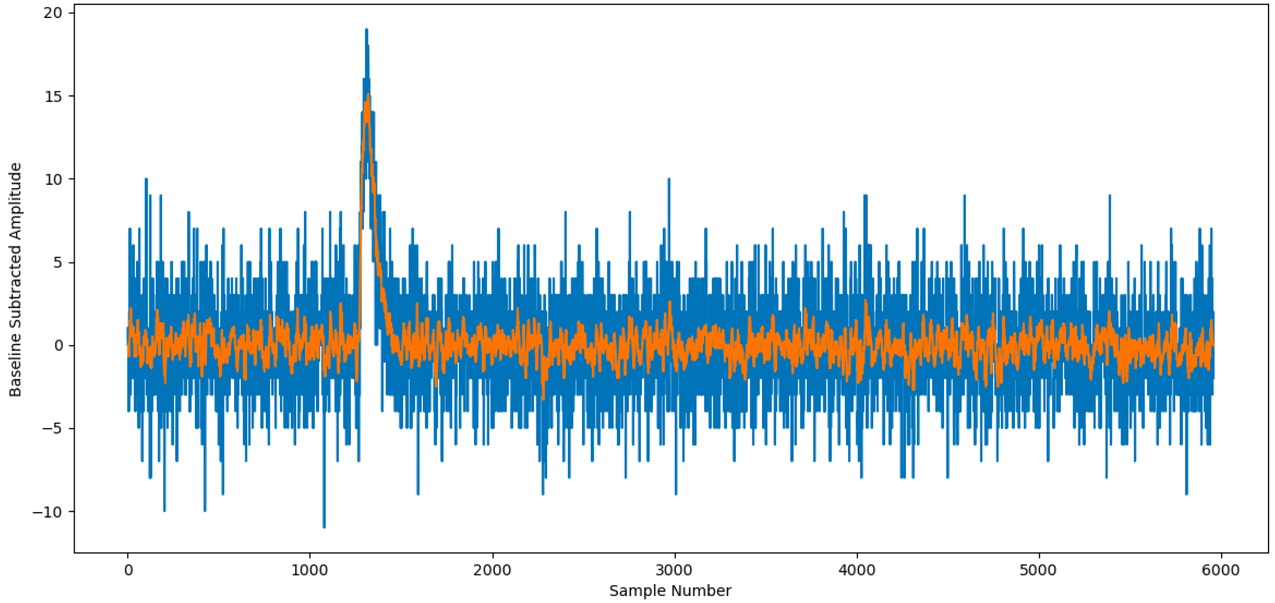
1) Single PE detection efficiency,

2) Fake hit rate,

For different SNR (noise levels), and hit finding thresholds/ variables.

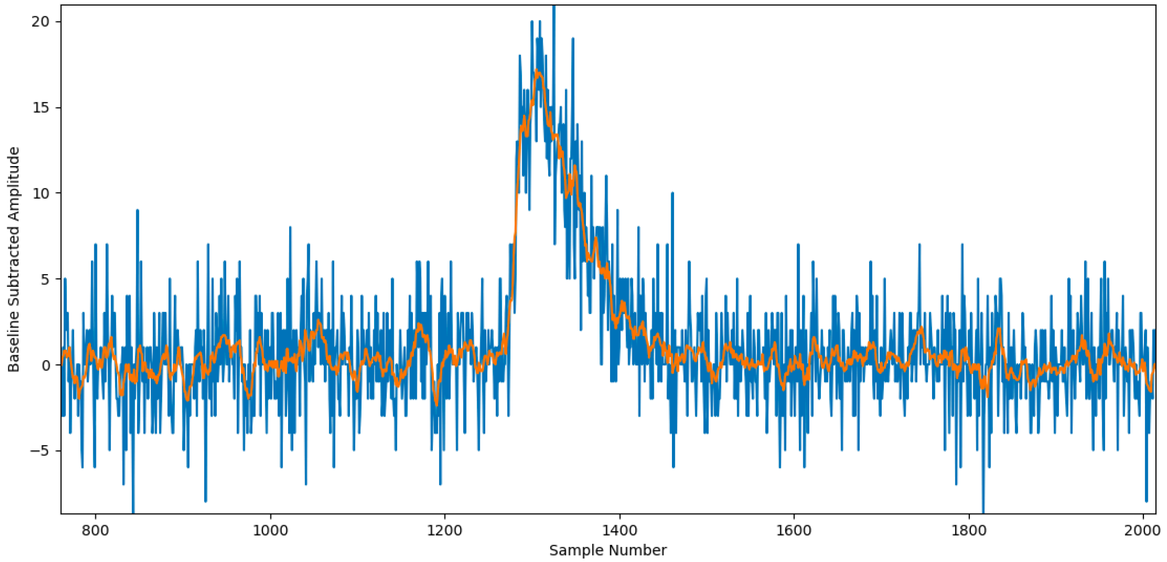
Hit Finding in Pyreco: FEP-Level

For “Veto” Waveforms



Example 50,000 ns long waveform, 1 PE hit with ASIC response, 1.4 mV RMS electronics noise (SNR = 5), RA filtering = 10 bins/ 80 ns.

Zoomed in version around the 1 PE hit.



Hit Finding in Pyreco: VX/FEP-Level

For “Veto” Waveforms

How does the hit finding work?

1) On filtered waveform, search to see if any sample crosses a threshold, typically $\mu_{\mathrm{BL}} + 3\sigma_{\mathrm{BL}}$.

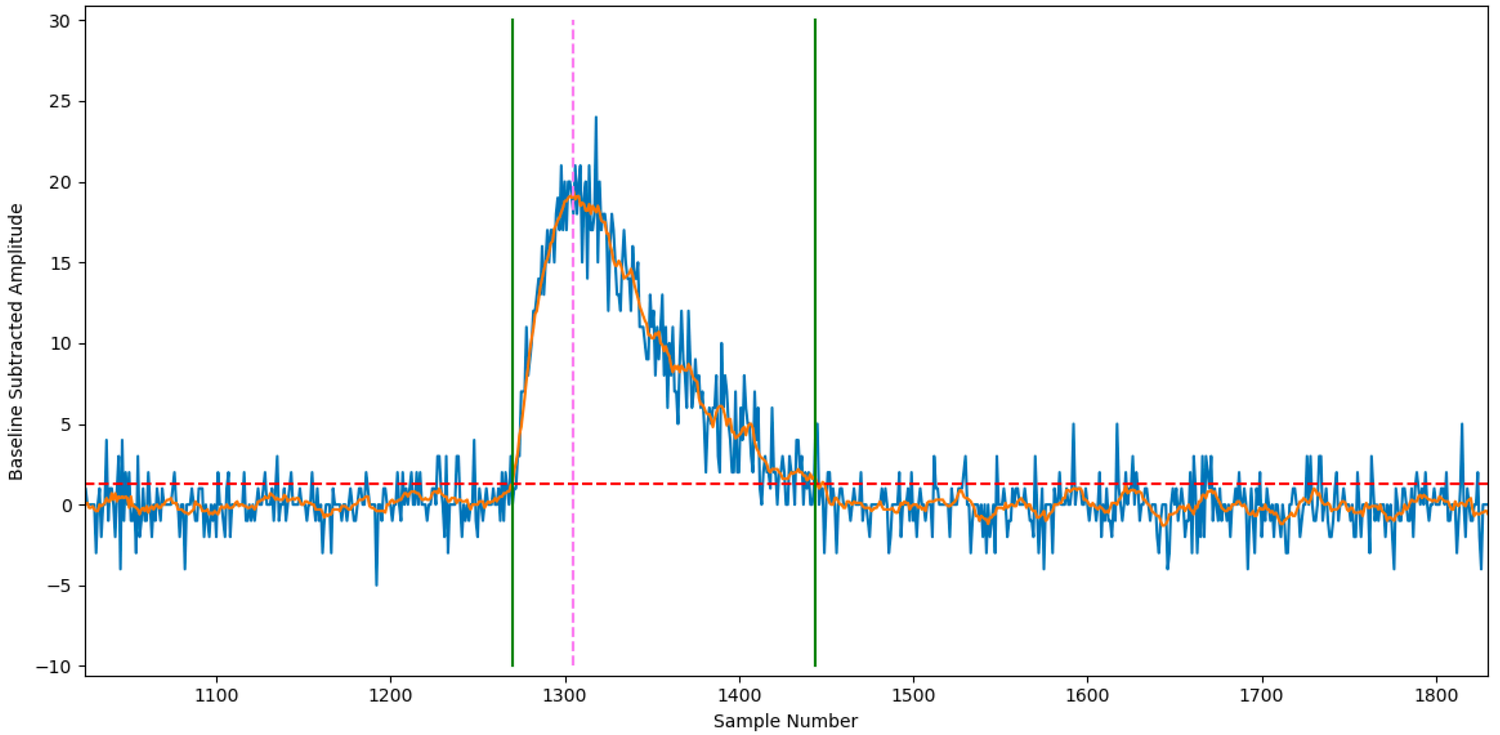
2) If sample crosses this threshold, start summing samples of waveform - continue until sample falls below threshold.

3) Check if the hit is large enough: is the Sum of the hit/ Max amplitude of hit > given value? If yes, record hit, if no, do not record hit.

μ_{BL} and σ_{BL} are calculated in the first 600 samples of the waveform == 4800 ns.

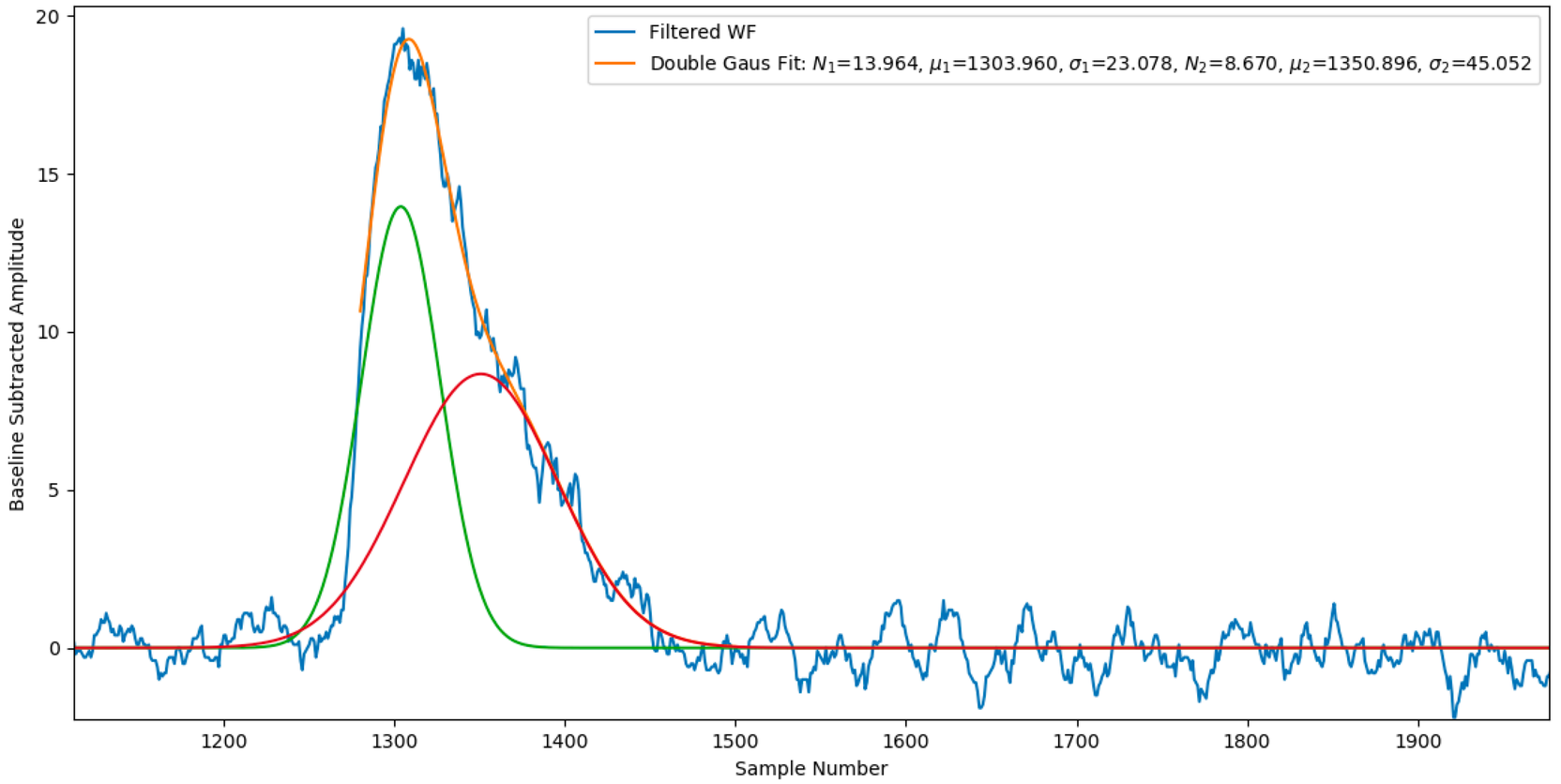
Example WF with SNR = 8, RA = 10 bins, Thr = $3\sigma_{\mathrm{BL}}$, Sum/Max = 25

Note: Sum/Max variable is somewhat arbitrary at the moment…



Single PE Detection Efficiency / Fake Hit Rate: VX/FEP-Level

For “Veto” Waveforms - Update to ToT Requirement



Fit a single PE pulse (ASIC) with double Gaussian to get an estimate for width of pulse (for ToT requirement).

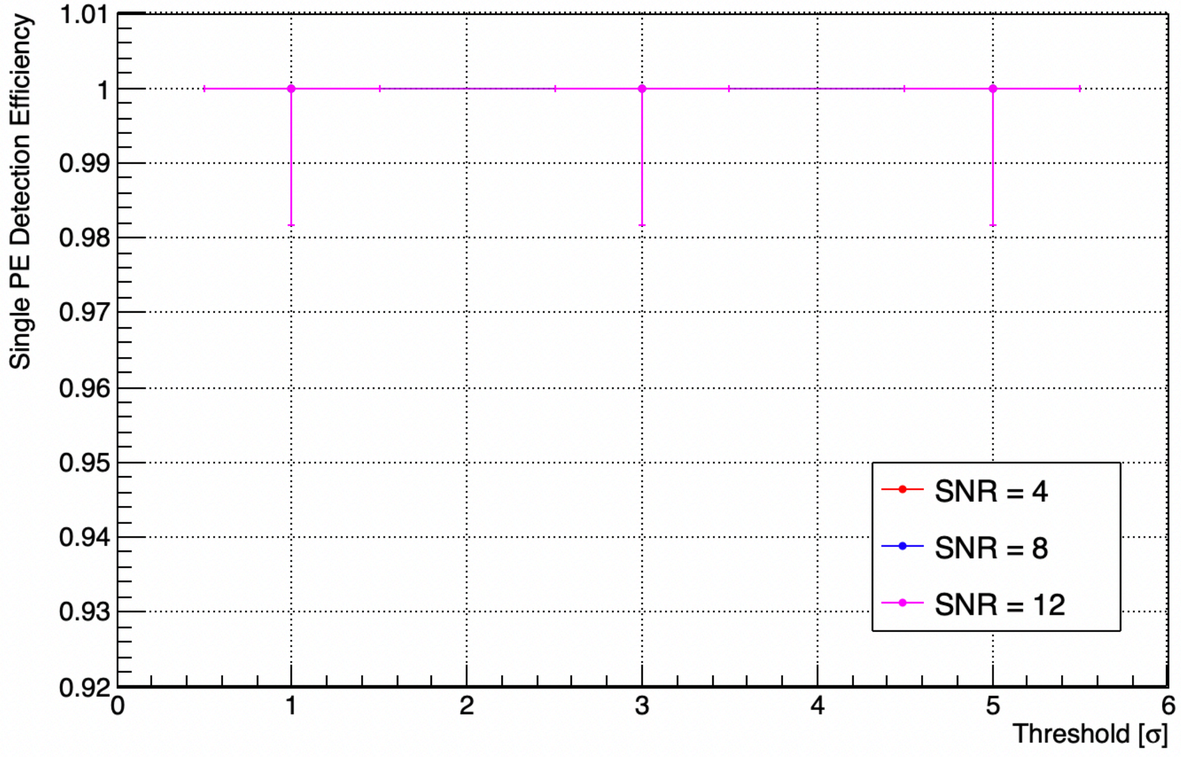
Taking into account fitted σ_1, σ_2, ρ_{12} values, width of summed gaussian is 50 bins long.

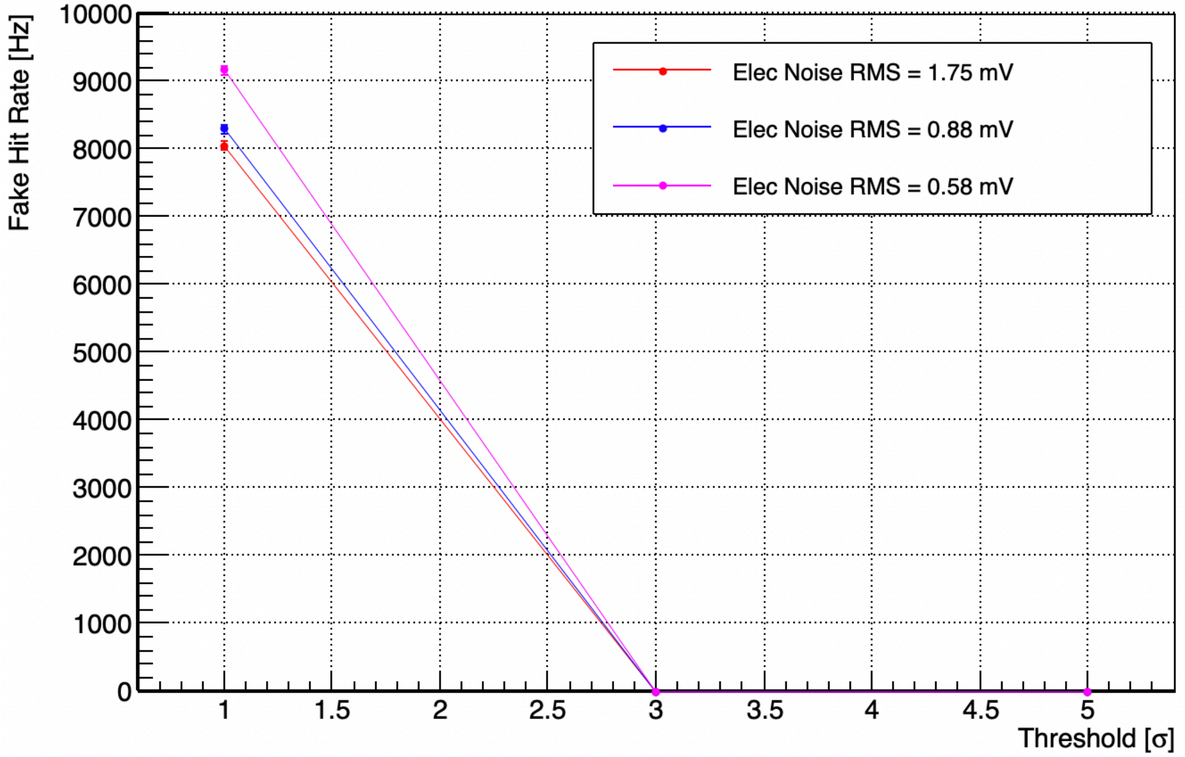
Calculate single PE detection efficiency and fake hit rate using single PE waveforms based on ToT requirements of $0.5\sigma_{pulse}$, $1\sigma_{pulse}$, $2\sigma_{pulse}$

Single PE Detection Efficiency / Fake Hit Rate: VX/FEP-Level

For “Veto” Waveforms - Update to ToT Requirement

ToT = 25 bins ($0.5\sigma_{pulse}$)

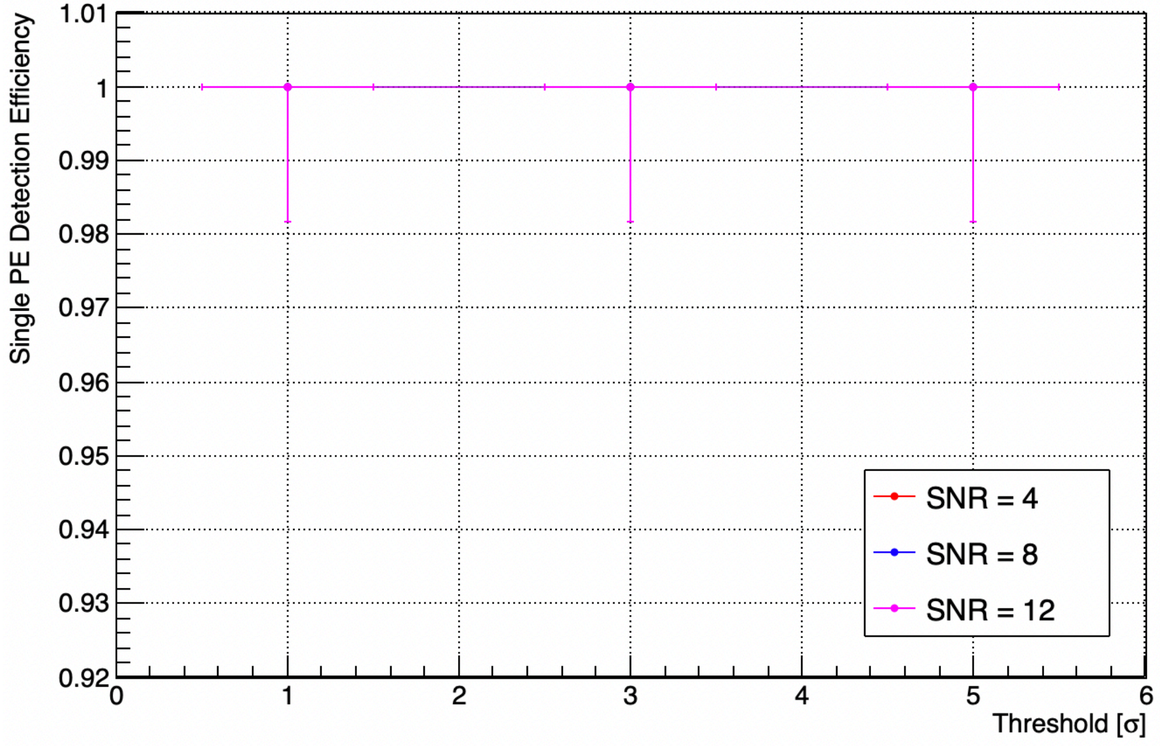


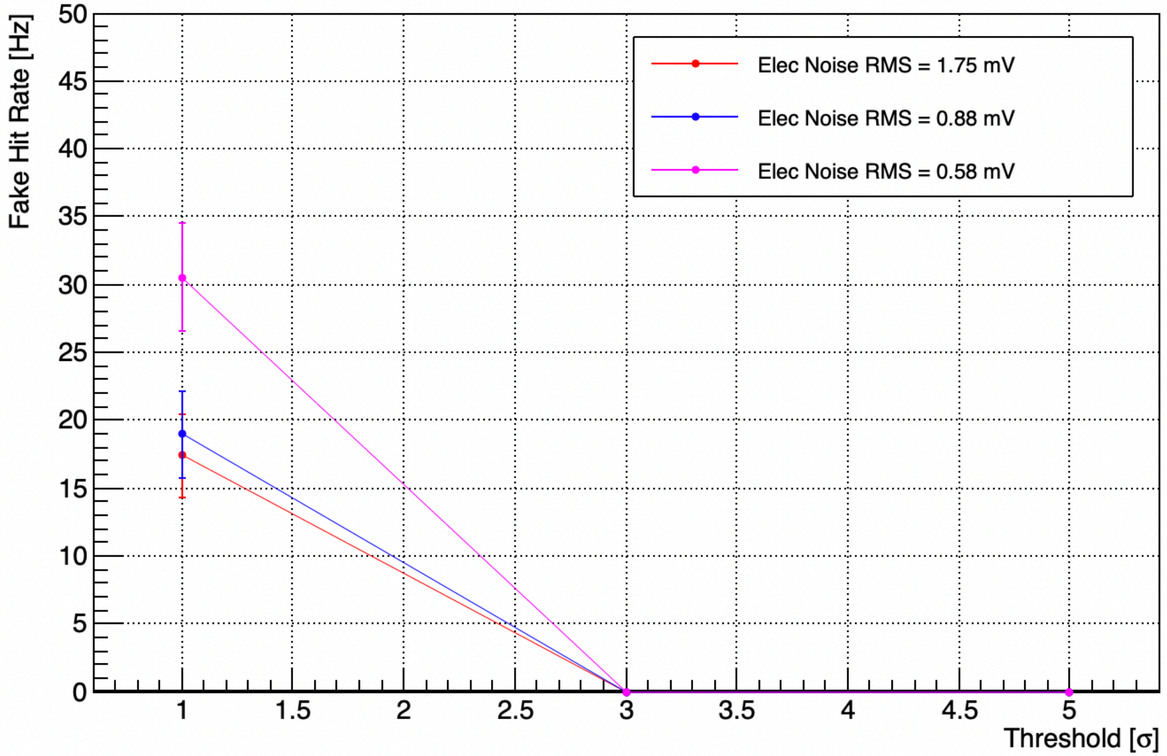


Single PE Detection Efficiency / Fake Hit Rate: VX/FEP-Level

For “Veto” Waveforms - Update to ToT Requirement

ToT = 50 bins ($1\sigma_{pulse}$)

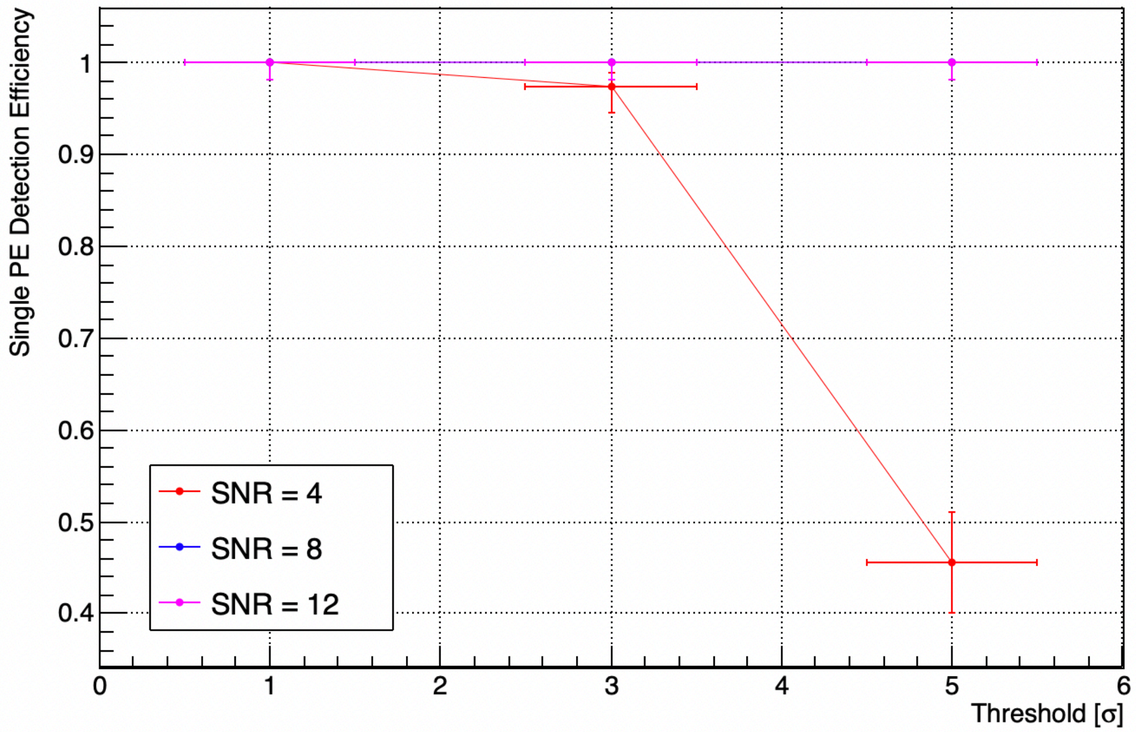


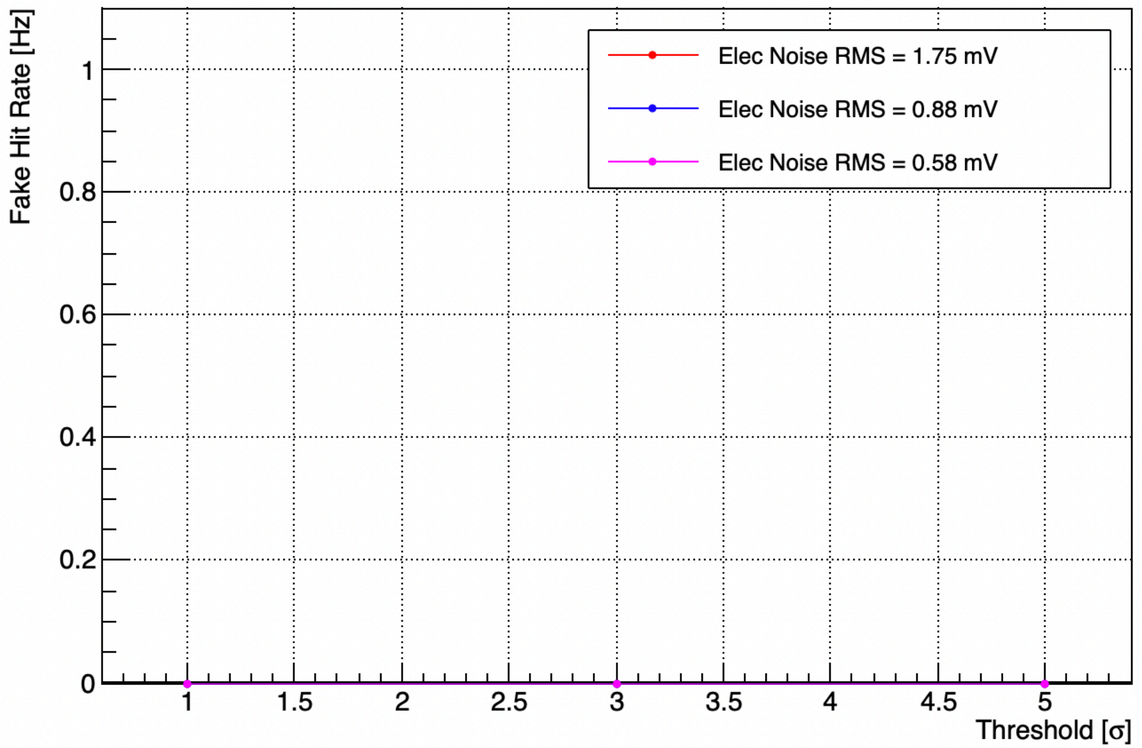


Single PE Detection Efficiency / Fake Hit Rate: VX/FEP-Level

For “Veto” Waveforms - Update to ToT Requirement

ToT = 100 bins ($2\sigma_{pulse}$)





Next Steps

Optimise hit finding algorithm using simple single PE ASIC waveforms.

Implement hit finding algorithm studied here in FEP in vslice to test FEP CPU performance, and study output of FEP and whether it is consistent with input. Study average event size of veto events. Study viability of 1) veto-triggered events (Rate and Size??) and 2) TPC-veto window events.

Will need to repeat studies considering channel summation.

Contact Shawn regarding Outer Veto, and how this ties in with the Veto DAQ system.