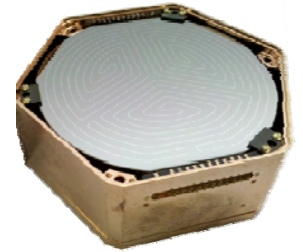


The SuperCDMS-SNOLAB Trigger System (WBS 1.6.2 and 1.6.6.)

David Toback
April 2, 2015
Review at FNAL



Presentation Outline*

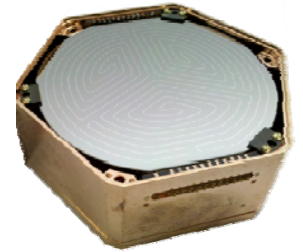


1. Physics Drivers & Triggering Modes
2. The Trigger within the DAQ and the Trigger Path
3. Information storage and L1 Trigger Algorithms
4. Levels 2 and 3
5. Other issues: Randoms, non-DCRC inputs and Risk
6. Summary/Conclusions

* Overview of results vetted by the CDMS Trigger Taskforce that completed its work in late 2014



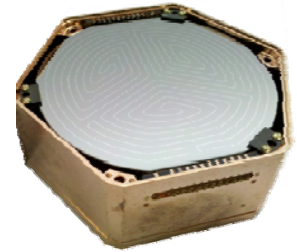
Physics Drivers



- The CDMS Trigger is designed to take data in a way that efficiently triggers for our full set of expected analyses:
 - WIMP search: energy deposition in a single detector
 - LIPs analyses and Nearest-Neighbor searches
- Current specs allow us to read out all detectors for every WIMP trigger in a single detector
 - No need for special triggers for the other searches



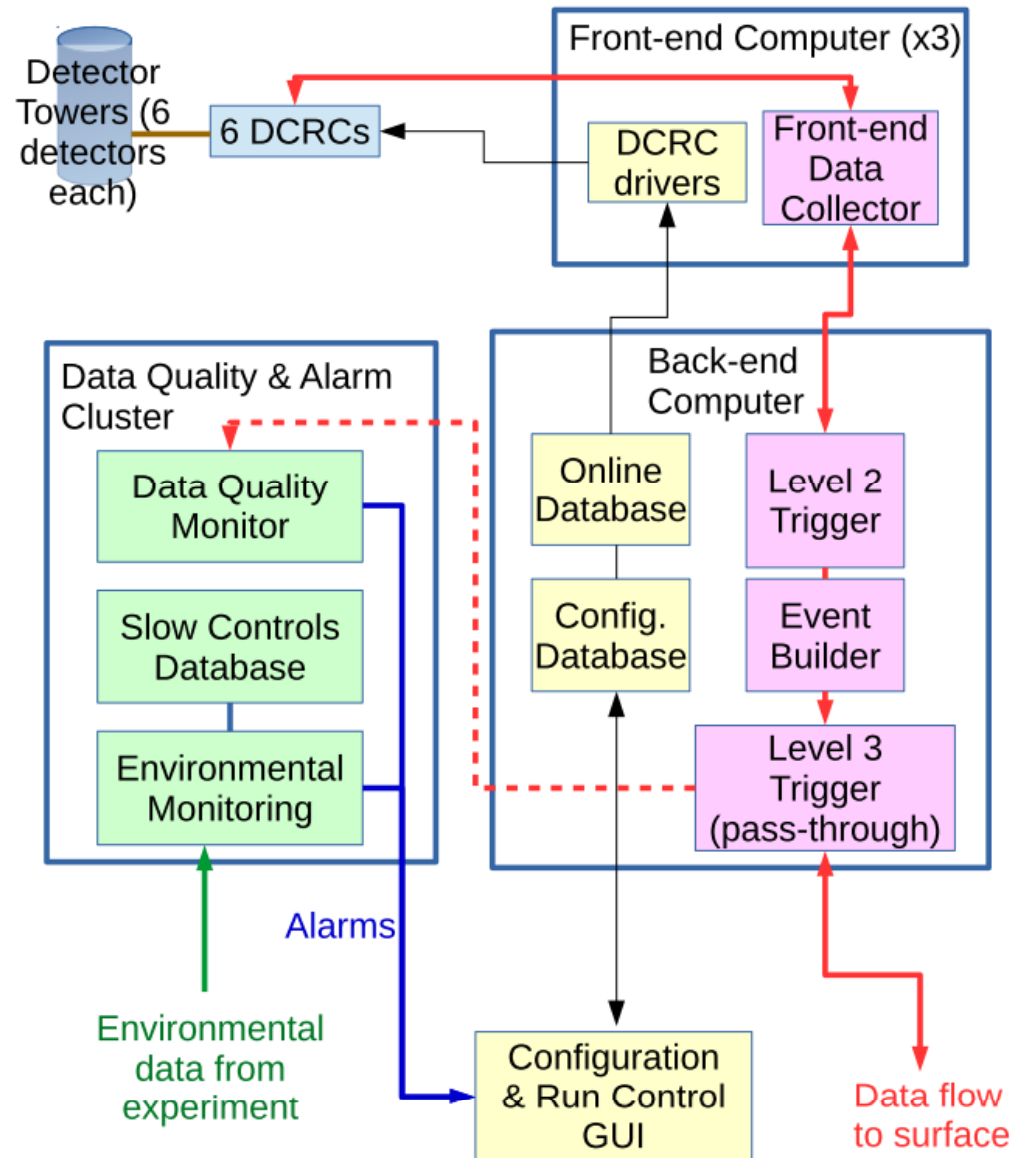
Triggering/Readout Modes



- Also need high-rate calibration runs, and lots of random triggers during data taking for noise studies
- Trigger takes in data from:
 - iZIP detectors, CDMSlite detectors, Accelerometers, and Active Veto detectors (if things change)
- We will read out the detectors in a number of different configurations for these modes:
 - All the towers for the same trigger time (WIMP data taking
 - anticipated to have a trigger rate of ~ 0.03 Hz)
 - Just a single detector or tower that passes the trigger (calibrations)

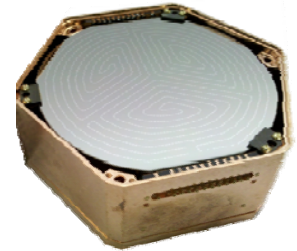
The Trigger within the DAQ

- Readout/triggering done in a single *Detector Control and Readout Card (DCRC)*
 - 1 Detector = 1 DCRC
- Level 1 in DCRC digitizes inputs, creates trigger primitives and reports its output to L2/DAQ
- Integrated Trigger/DAQ design

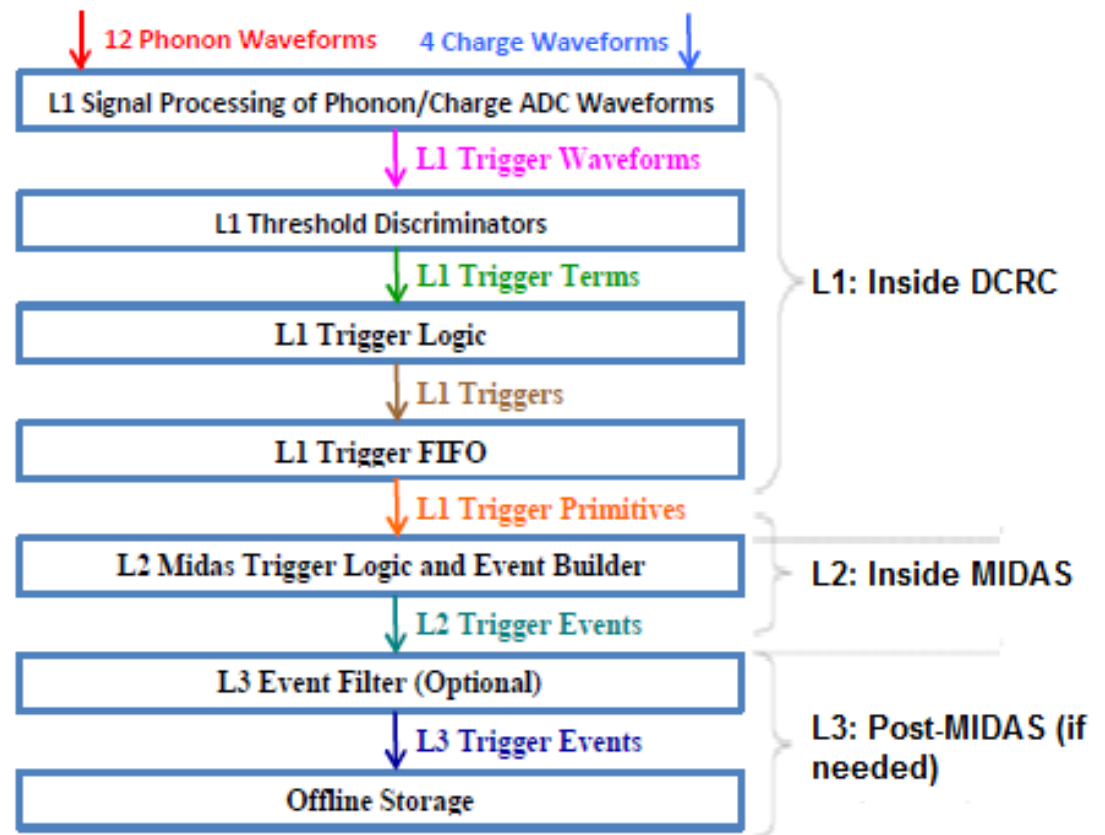




Overall Trigger Path (Part 1)

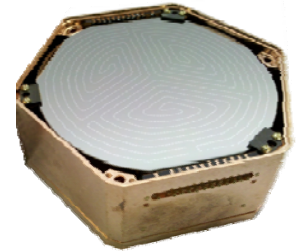


- The trigger is made of three systems known as Level 1, Level 2 and Level 3
- Will walk through details of these various steps after an overview of the overall goals that this system achieves

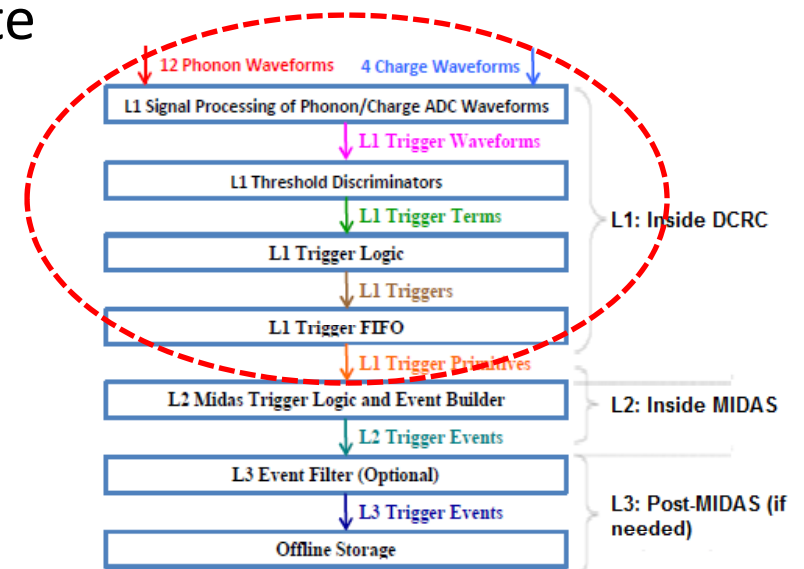




Detailed Trigger Steps (Part 2 - DCRC)

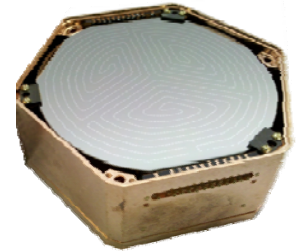


- **Inputs:** Digitized and stored as 16-bit words in a circular buffer
- **Level 1 (L1) Signal Processing:** Decimate (downsample) phonon inputs, sums them, and create *L1 trigger waveforms*
- **L1 Threshold Discriminator:** Decision based on *L1 trigger waveforms*
- **L1 Trigger Logic:** Make Level 1 trigger decisions and create *trigger primitives*
- **L1 Trigger FIFO:** Stores primitives for use by Level 2 (L2) trigger

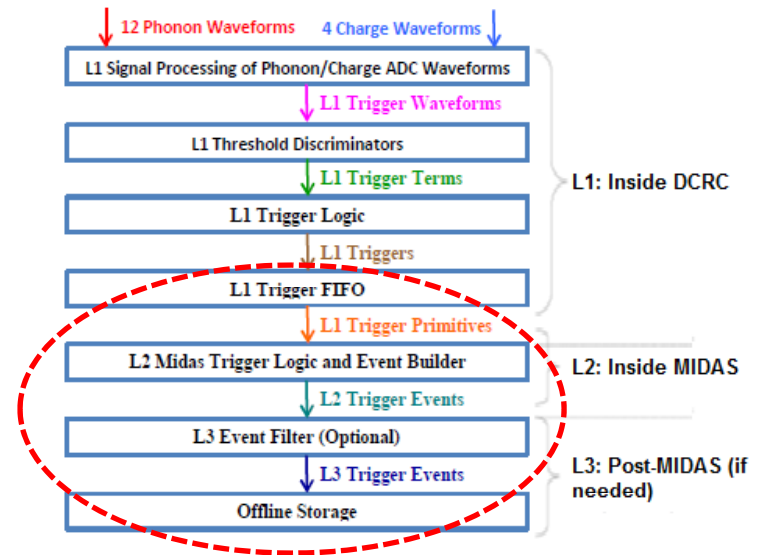




Detailed Trigger Steps (Part 3 – Level 2)

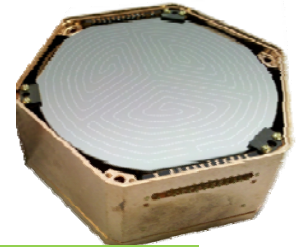


- **Level 2 (L2) Trigger Logic:** Examine the L1 trigger primitives for all detectors, decides which detectors to read out into the front-end computers
- **Event Builder:** Assembles data into the event structure in back-end computer
- **Level 3 (L3) Event Filter:** Optional software filter that could use event data to further reduce the trigger rate
 - Currently the L3 is just a pass-through to **Offline Storage**

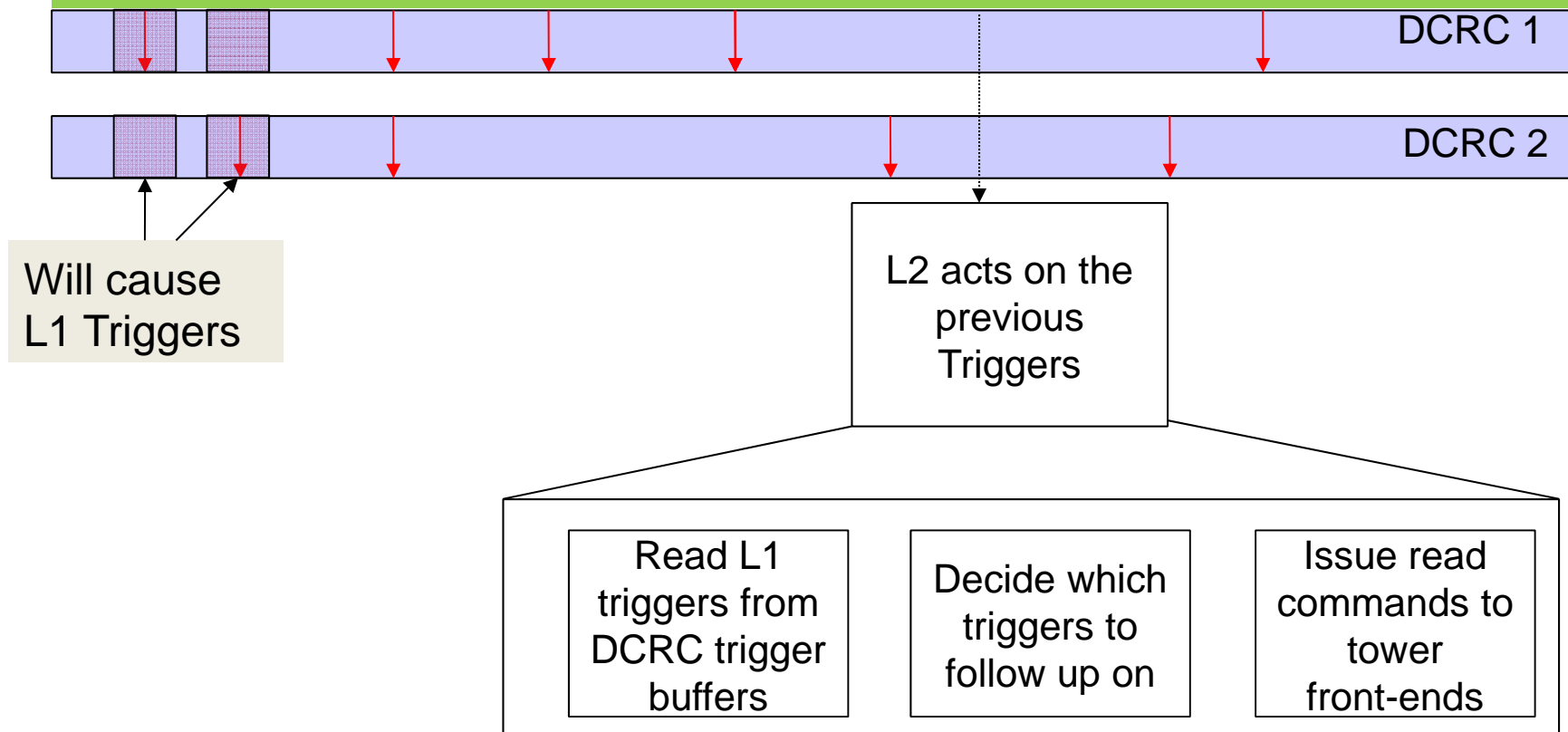




Watching Events Pass Through the System

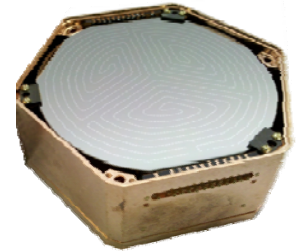


- DCRCs continuously write digitized waveforms into memory
- L1 trigger algorithm notes interesting signals (red arrows).
- The L2 trigger system periodically collects the L1 trigger times from all DCRCs and decides what to do with them – No interruption of L1 trigger operation.





More details: Detector Readout/DCRC Information Storage



As part of Level 1:

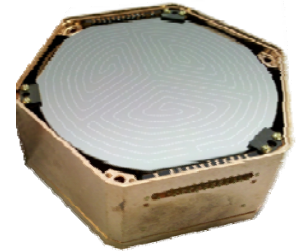
- Digitize phonon and charge channel at 625 kHz and 2.5 MHz
- Stores waveforms in an ~ 3.35 s long circular buffer

As part of Level 2:

- DCRC is polled about once per second by the L2
- Readout is 52 ms centered on the physics pulse
 - ~ 2 ms at nominal digitization speed, ~ 25 ms each of pre-pulse and post-pulse waveforms with $16\times$ slower
 - Allows for measurement of fast time and slow time components (as low as 19 Hz)



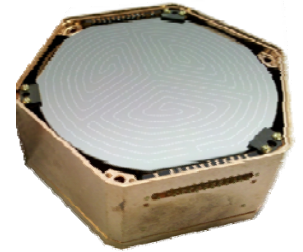
L1 Trigger Decision Algorithm in the FPGA



1. Sum the 12 phonon channels as input
2. Apply an *Finite Impulse Response* filter to the last N samples of the input to produce a “filtered” trigger waveform
 - a) Coefficients of FIR are chosen based on noise
 - b) FPGA Designed to allow for easy changing
3. If result goes above threshold start a peak-search algorithm to find the peak and wait until waveform falls below a lower, “off” threshold
4. Check whether trigger is otherwise vetoed (or we are doing a pre-scale)
5. Write peak time and value (and other info) to L1 trigger FIFO for readout by Level 2



Level 1 Trigger Algorithm

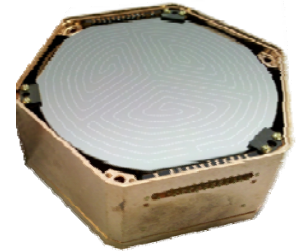


- Using a Finite Impulse Response (FIR) filter in the FPGA allows us to choose between many choices like running an Optimal Filter*, a Boxcar filter or a bandpass filter
- We have chosen an FPGA with sufficient resources to run the trigger algorithms in real time, keeping the amount of data passed to the L2 trigger reasonably small, and the requirement that the trigger firmware be easy to maintain and update

Note: We will do this with four different FIR filters.



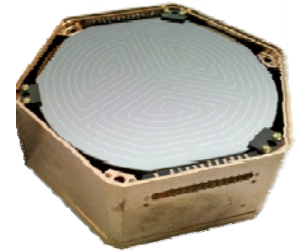
Level 2 Implementation Notes



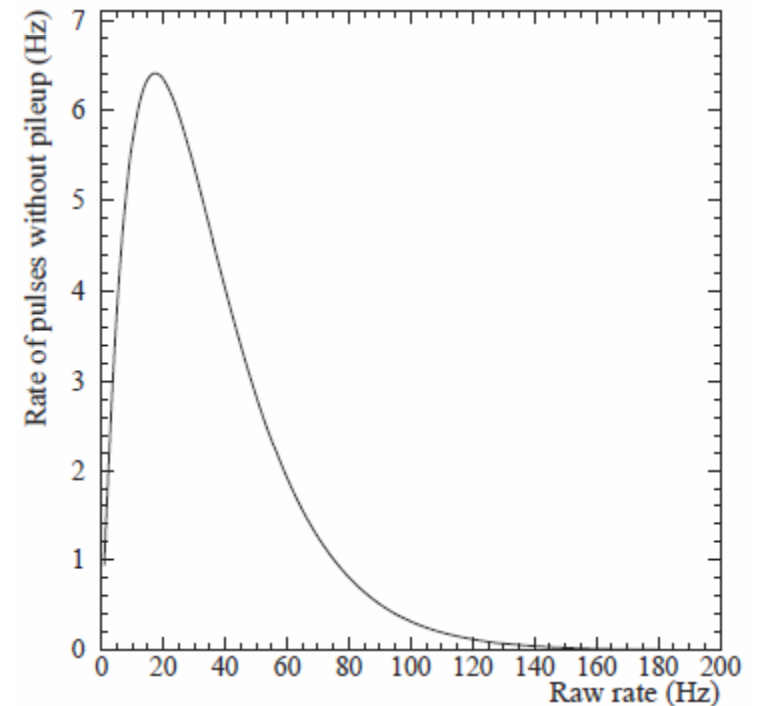
- L2 trigger is purely software, runs on the back-end computer
- Retrieves L1 trigger buffer information once per sec
- L2 algorithms are flexible and fully customizable
- Controls random trigger in software and collects randoms from L1
- Additional features:
 1. Performs Pileup rejection during calibrations
 2. Does Trigger merging
 3. Does Pre-scaling of triggers



L2 and Pileup rejection for Calibrations

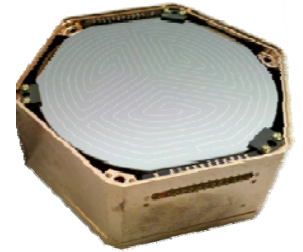


- During calibrations, with a powerful source, we can have two energy deposits at the same time
- Call this “pileup” ... Can’t use in offline analysis. Can’t just infinitely increase the source strength
- L2 will reject piled-up events if there is no evidence that there is a multiple scatter event.
- Calibration rate expected to be ~ 5 Hz per detector, or ~ 240 Hz across the whole array





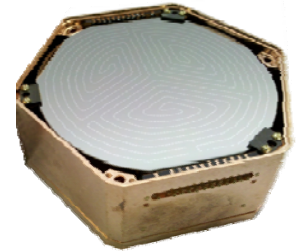
Level 3



- Design allows for a Level 3 trigger that can do additional event rejection
- Current design doesn't need this so we leave L3 in “passthrough” mode



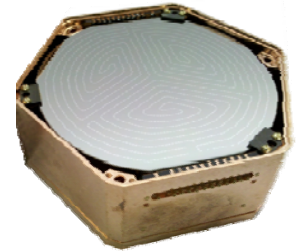
Randoms



- Need lots of random-noise pulses for offline analysis and Triggering
- Noise can, and often does, change in time
- Plan: take randoms before, during, and after the run:
 - During Run: Trigger at constant rate of ~ 0.1 Hz – 360 events in 3 hours
 - For 3 hr runs will take 500 randoms at the beginning and ending of a run
 - Implemented will be created by a pseudo-random trigger generation process on the DCRCs



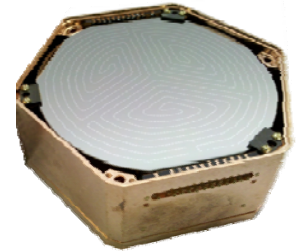
Trigger interface with non-DCRC hardware



- Three other inputs to the Trigger:
 1. DCRCs has a LEMO input to provide an externally imposed L1 trigger (Mostly for test facilities)
 2. The Active Veto system, if implemented, will be examined by L2 and will produce L2 triggers
 3. Accelerometers (used to monitor mechanical vibrations throughout the experiment) can provide L1 inhibits
- DCRC will record a timestamp at the beginning and end of each inhibit signal for later use



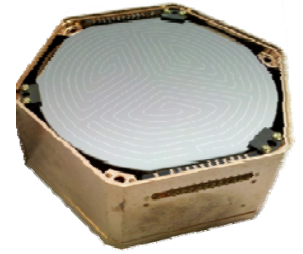
Risks and Mitigation



- **Manpower risks:** Luckily not many people needed. Fortunately Soudan Trigger experts remain committed to project and have trained next generation
- **DCRC Risk:** The primary Trigger board, DCRC, is being designed by a professional engineer at FNAL, with extensive testing on versions A-C already done
 - Extensive testing expected for Revision D both in-house as well as at many teststands
- **MIDAS Software Risk:** Much of the Trigger software, in MIDAS, is already being exercised and used at teststands.



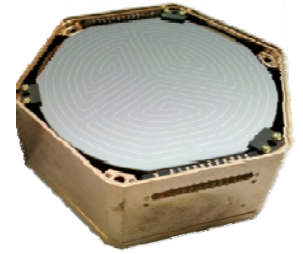
Summary/Conclusions



- Trigger system is low risk, builds on what worked for previous experiments and will have its primary board through four iterations before deployment
- The infrastructure is well supported and the software framework is well-tested
- The Trigger Framework underwent a long internal review by a Trigger Taskforce, and the algorithms and plans are now well vetted
- We have a plan of staged development and deployment at test facilities, which will both debug the system and give collaborators familiarity running something close to the SNOLAB DAQ

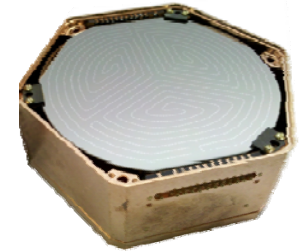


- Backups





Let the Subject Begin



Talk guidance
provided by Dean
Hoffer, FNAL Project
Office

Guidance for Plenary Session Presentation Slides for Director's Reviews

General Guidelines on Slides:

- Stay within you time. Plan on questions and leave about 25% of your allocated time for questions.
- Plan on approximately 0.75 slides per minute of talk time allocated
- Absolute limit is 1 slide per minute
- Keep text and figures large
- Limit complexity of slides
- Focus on import points, skip complicated details or agree to discuss them in a breakout session

Slide Content:

- Explain what the system(s) are that are being presented and the WBS that it is associated with.
- Technical issues to be addressed.
 - Performance Specifications
 - Design Concept
 - Design Issues
 - R&D Program
 - Past Results
 - Future plans
- Show pictures of real hardware if possible.
- Explain why you conclude that your part is technically feasible.
 - I.E. (Someone built similar widgets elsewhere. Built similar widgets here.)
- Show the Organization structure for this WBS deliverable.
- **Show cost estimate with contingency for this WBS deliverable.**
- **Show the schedule and key milestones associated with this WBS deliverable.**
- Show key risks associated with this WBS deliverable.
- **Explain why you can say the costs are understood.**
 - I.E. Similar to widgets build at lab X, costs based on that
- Conclude that your system is technically and financially feasible.
 - My Part is Technically feasible
 - **My Part is Financially feasible**
 - The Risks are Understood
 - The R&D planned will Reduce the Risk
 - We have a Plan for what to do next

Cost and
schedule can be
skipped for April

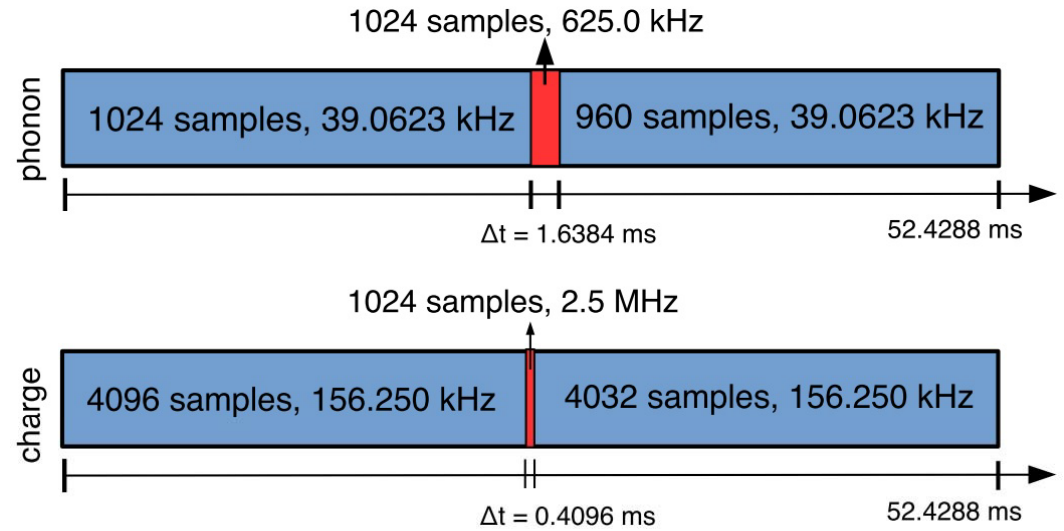


Waveform Readout

Want fast digitization speed in DCRC, & long waveforms to filter low-frequency noise.

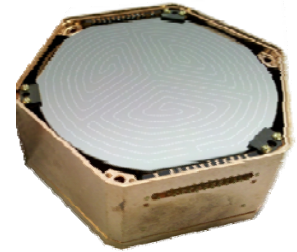
This results in large data volume, possibly exceeding SNOLAB network capacity.

Solution: downsample waveforms during readout in pre-pulse and post-pulse regions by averaging every 16 points. Preserves fast timing in main pulse, and sensitivity to low-frequency components of signal and noise.





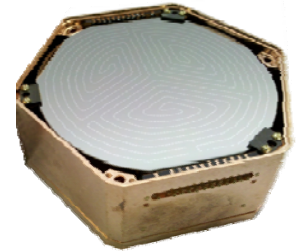
Within the Algorithm



- Each trigger term is associated with a bit and the algorithm for setting and lowering the bit is done using two programmable (one high and one low) thresholds.
 - When the selected waveform goes above the first threshold, the bit is asserted; when it drops below a lower, second threshold, the bit is reset. During the time that any bit associated with a particular waveform is asserted, the DCRC will search for the peak amplitude of that waveform, peak time. The time between assertions constitutes the “peak search window”.
- Whenever at least one trigger bit is asserted, the DCRC will create a 16-bit trigger word. During the peak-search window, the DCRC will track whether any trigger bits (including those associated with other waveforms) are asserted, even if they are not asserted at the waveform’s peak.
- Once the peak search window is ended, the DCRC will produce a 16-bit trigger word representing the state of all the trigger bits during the peak-search window. Each of the 8 trigger bits (no matter which waveform it is associated with) will be represented twice in this word: once to show whether that bit was asserted at the time of the peak, and again to show whether the bit was ever asserted during the peak search window.
- This word will be compared to a mask defining which bits are required, and another mask defining which bits are vetoed. Any bits not appearing in the requirement or veto masks are ignored.



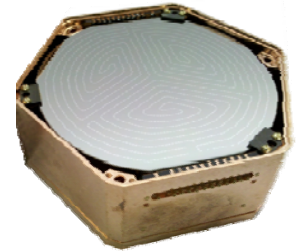
DCRC Configuration



- DCRC configuration is done through a set of web-based tools that access all detector settings using a custom graphical user interface (GUI) that interfaces with MIDAS's central online database, which contains all details of the experimental setup. This is discussed in Section 9.5. These settings can be saved to an offline database for future use, and default settings for different running modes will be defined to allow quick reconfiguration of the detector array.



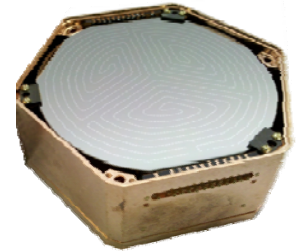
Notes: Where do these go?



- DCRCs continuously digitize signals and apply detector-level trigger algorithm.
- DAQ combines detector triggers to decide which waveforms to retrieve from DCRC memory.
- Tower-level processes collect detector triggers and read out waveforms. Back-end processes run central trigger, configuration DB, event builder.



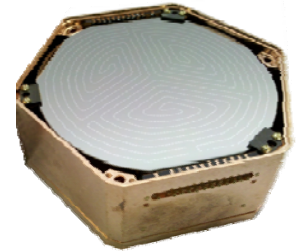
Off-scale stuff



- In the event of exceptional conditions during data-taking or calibrations, such as an off-scale pulse or a saturation event (sufficiently large energy deposition to take the TESs into normalconducting mode), the waveforms may remain above threshold for a very long time, and the peak might be indeterminate. In order to read out the rising edge, the timestamp in the trigger FIFO will be placed a set (configurable) time after the waveform first rose above threshold. (That is to say, there should be a maximum length of the peak-search window which, if exceeded, will cause the DCRC to just report the start time of the peak-search window as the time for the L1 trigger rather than the peak of the event.) The design of the trigger is such that it will be easy to create and implement the final settings based on the noise environment at SNOLAB. During SNOLAB operations, we will set and adjust as needed the weights for the linear combinations of the raw waveforms, the associations between trigger terms and trigger waveforms, the turn-on and turn-off thresholds used to create the trigger terms, the association between the trigger mask pairs and trigger waveforms, the contents of the mask pairs, and the pre-scale thresholds. Procedures for determining these settings are well underway.



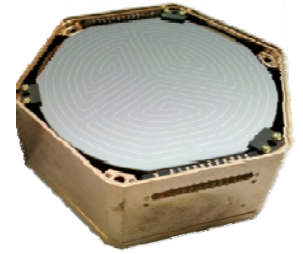
Multiple FIR Filters in Parallel



- Detail: The firmware in the DCRC will run multiple FIR filters in parallel, and will be designed so that it is straightforward to change the filter coefficients via a simple set of commands from the DCRC drivers in the front-end computers without recompiling the firmware.
- The capability to run multiple FIR filters in parallel allows us to tune the trigger to best pick out the expected signal on top of the experimental noise, as well as to apply special-purpose filters to trigger on abnormal signals, such as pulses that saturate either the transition edge sensors or the digitizers. This flexibility also grants the capability to easily adjust the trigger parameters to accommodate changes in the experimental conditions over time. High-voltage detectors may also require special trigger consideration, and FIR filters will be capable of handling this task as well, so no special high-voltage mode trigger hardware is needed. The L1 trigger permits a flexible implementation of an FIR filtering trigger that will allow us to trigger on different choices of waveforms (e.g. on individual channels or on weighted sums of channels), to apply simple logic conditions to trigger terms (e.g. requiring triggers on both sides of a detector), and to be controllable by a relatively small number of parameters that can be loaded through registers into the DCRC without requiring firmware changes.
- The design includes four elements:
 1. Four FIR filters to be run in parallel, each producing a trigger waveform that will be used in triggering
 2. Eight trigger terms (logical bits used to decide if a trigger will be issued), each generated from applying threshold checks to a trigger waveform in a configurable manner
 3. Trigger logic to determine a set of eight configurable trigger definitions that will generate triggers only if certain programmable combinations of trigger terms are set (or not set, in the case of vetos)
 4. Procedures for deciding what information to include in the L1 Trigger FIFO for each trigger, for use by the L2 trigger system, including trigger time determined by a peak-finding algorithm.
- The DCRC will compute 4 trigger algorithm waveforms, which are configurable linear combinations of the 12 phonon and 4 ionization channels. Each of these waveforms will be processed by its own FIR filter algorithm to produce a filtered waveform. (For each trigger waveform we must specify the 16 linear weights for the individual raw channels that define the linear combination, as well as the FIR filter coefficients that are applied to that linear combination to produce the trigger waveform. The WIMP search data will only trigger on the phonon terms.) The choice of having four FIR filters is a compromise between FPGA size, power draw, firmware complexity, and trigger flexibility. We have chosen a Cyclone IV FPGA, which will have 200 multipliers running at 200 MHz. This would be sufficient to carry out 64,000 operations between each phonon digitization step. That is enough to calculate four FIRs in real time with a length of order 104 samples each.
- Four FIRs would be enough, for example, to run one optimal filter trigger on the summed phonon channels, another on side A, another on side B, with a fourth running with a simpler filter such as a boxcar filter. Using a larger FPGA would be more expensive and complicate the hardware and firmware design, while four independent filter waveforms seems enough to provide enough flexibility. After the creation of the 4 FIR trigger waveforms, the DCRC will calculate 8 trigger terms which will be used as input to define a trigger word (defined below).



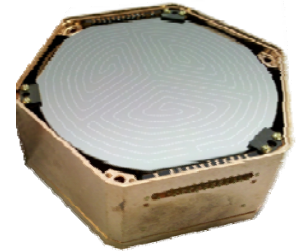
Trigger system design goals



- Deadtime-free triggering
- Data volume within limits of SNOLAB network
- Ability to handle high trigger rates in calibration mode, to maximize livetime
- As low of a trigger threshold as possible
- Flexible, programmable software trigger
- Simple, intuitive detector configuration tools, to minimize operator errors



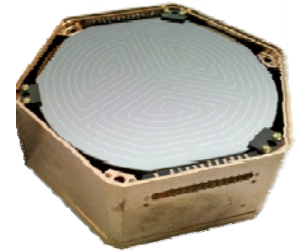
Overview of Integrated L1 and L2



- Each detector has a DCRC that produces local trigger decisions at L1
- This detector-level (Level 1) trigger is implemented in firmware in an FPGA running a finite impulse response (FIR) filter, which typically is applied to the summed signals from all 12 phonon channels
- The FIR filter will provide the flexibility to use many different trigger algorithms via easily programmable filter-weight coefficients that can be loaded using the DAQ
- These algorithms can range from a simple threshold-crossing to a sophisticated optimal filter that uses measured noise characteristics to trigger with the lowest possible thresholds on each detector
- The Level 1 trigger runs continuously in real time, as waveforms are constantly digitized inside the DCRC's circular memory buffer
- Times at which Level 1 triggers occur (filtered pulses go above threshold) are recorded in a trigger FIFO buffer in a deadtime-free manner.



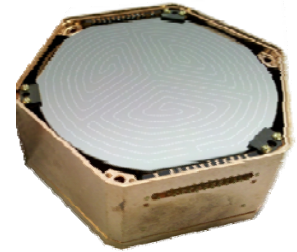
Level 2 stuff



- We note that in L1 each detector is a self-contained object, but at L2 one detector can influence the readout of another. For example, a trigger decision could be made based on a coincident signal in multiple detectors or information from the neutron veto detectors could cause (or inhibit) the readout of the iZIPs.
- It receives Level 1 trigger information from each tower's frontend process and decides which waveforms should be retrieved from each tower. This includes applying any coincidence or veto requirements between towers, such as initiating the readout of other towers when one tower generates a trigger.
- Note: The Level 2 trigger also is responsible for injecting random software triggers to read noise traces from the DCRCs, and for rejecting events, during calibration data-taking, that show evidence of pileup within a single detector readout.



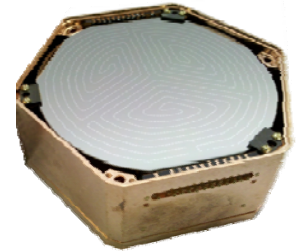
Level 2 and High-Rate Calibration Data Taking



- As discussed in Section 9.3.3, there's a maximum calibration source strength we can tolerate due to issues with multiple overlapping pulses in the waveform trace. This rate limitation has nothing to do with triggering, and so we can't simply cut hard in trigger and arbitrarily increase the source strength to compensate.
- Rejecting pileup events (events with multiple pulses within the readout window) during calibration data-taking, to ensure high efficiency for calibration data.
- Calibration mode is expected to have event interaction rates in detectors that are so high that multiple pulses may overlap in time in a single detector. We refer to pulses that occur before the previous pulse has completed as pile-up events.
- In calibration mode only the triggering detector will be read out, for reasons of data volume. In addition, algorithms running the Level 2 trigger will be employed to reject pile-up events, so that they don't swamp the DAQ bandwidth with events that cannot be analyzed.



Level 1 Stored Information and Output



If the energy deposit creates a Level 1 Trigger, Trigger information is stored:

- If all required bits are set, and no vetoed bits are set, then an enable bit and a pre-scale bit will also be checked.
- The enable bit allows each mask pair to be turned on or off.
- Pre-scale functionality is affected as a comparison between a pseudo-random number generator and a threshold, allowing trigger entries to be randomly rejected at a certain rate.
- If the enable bit and pre-scale bit are both set, then a trigger primitive will be created and placed as an entry in the trigger FIFO, containing:
 - • 32 bits: timestamp of trigger waveform peak
 - • 16 bits: amplitude of trigger waveform at its peak
 - • 16 bits: trigger word
 - • 8 bits: bits indicating which mask pairs caused the FIFO entry (which also clearly indicates which waveforms were involved)
- The DCRC will also use and respond to external information, including external triggers or veto signals to deal with potential noise changes, or episodic noise like the Soudan cryocooler noise.