

## Outline/Table of Contents

1. Introduction
  - a. Introduce the goal of the research conducted here
2. Motivation + Background
  - a. SM and limitations
  - b. Dark Matter
    - i. Introduce the usual suspects leading to DM
  - c. mCPs as a Candidate
    - i. Explain why we need specialized detectors to see them
  - d. MilllQan as a probe for mCPs
    - i. Introduce the location, the probing area, and the two detector geometries.
3. MilllQan Detectors Design
  - a. Principles
  - b. Search Backgrounds
  - c. Detector specifics
    - i. Scintillator material, PMT, Baseboards, LEDs, Wrapping, PMT response, dark rate
    - ii. Detector Electronics (HV, LV, Digitisers, Triggerboard, LV distribution)
4. Slab Detector Construction
  - a. Summer 2024
    - i. Dressing of Slabs
      1. Mu Metal
    - ii. SPE Matching
      1. Methods, Amplifier Boards, "Bad Modes"
    - iii. Physical Construction
      1. Light Leak Checking
    - iv. Cabling
      1. Validation of data into digitizers
5. PCCS - Introduction & Design
  - a. Introduce as a replacement for the LV distribution board.
  - b. Key electronic design elements (Raspberry Pi + Pico)
  - c. Flashing Mechanism
  - d. Physical design
    - i. PCB Overview
    - ii. Structure Overview
  - e. Introduce Interfacing Code Mechanism
  - f. Triggerboard S-Particle

6. Slab Detector Upgrade
  - i. Quick mention of the Early Saturation
  - ii. Installation of PCCS
7. PCCS Use - Remote Flashing
  - a. Description of Remote Access and Usability
  - b. CSV Format
8. Results
  - a. Voltage Sweeps
  - b. Flashing of 25 NPE, SPE
  - c. Detector Triggering on PCCS Injected Data
  - d. Possible Dt Triggering plots
  - e. The timing variable “tdc corrected” and “time”
9. Summary and Conclusions
  - a. Summary of contributions

## Appendices

- A. PCCS Flashing Map
- B. PCCS Constructing a Module
  - a. Guide on 8020, 3D Printed rails
  - b. Installation of PCB onto the Frame
- C. PCCS Design Versions
  - a. Bugs and fixes in each one
- D. PCCS Code Description
- E. PCCS Debugging, Fail Cases, and Previous Blade Replacement
- F. PCCS Testing Board

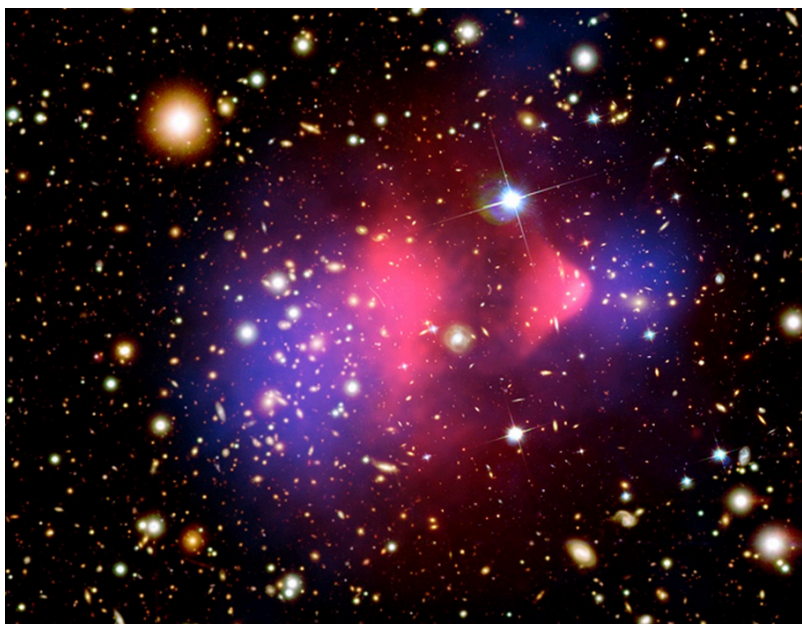


# Chapter 2

The Standard Model of Particle Physics (SM) is the culmination of the past 100 years of formulation and inquiry into the underlying structure of matter in our universe. Utilizing Quantum Field Theory, matter has been described as quantized particles. Standard matter is made of fermions, while force carriers are bosons. This description of nature has been repeatedly tested via observation of particle properties in specialized testing environments. One of the most important centers of particle physics advancements is particle accelerators, which allow for the creation of new particles via highly energetic particle collisions. Current-day particle accelerators such as the Large Hadron Collider (LHC) recreate the environment within microseconds of the Big Bang, allowing us to probe the conditions of the early universe. At the LHC, proton beams are collided together at 99.99999991% the speed of light, for a total collision energy  $\sqrt{s} = 13.6$  TeV. By colliding particles together at higher and higher energies, physicists have discovered 17 different elementary particles, intrinsic quantum numbers such as color, and the imbalances of the universe such as Charge-Parity Violation.

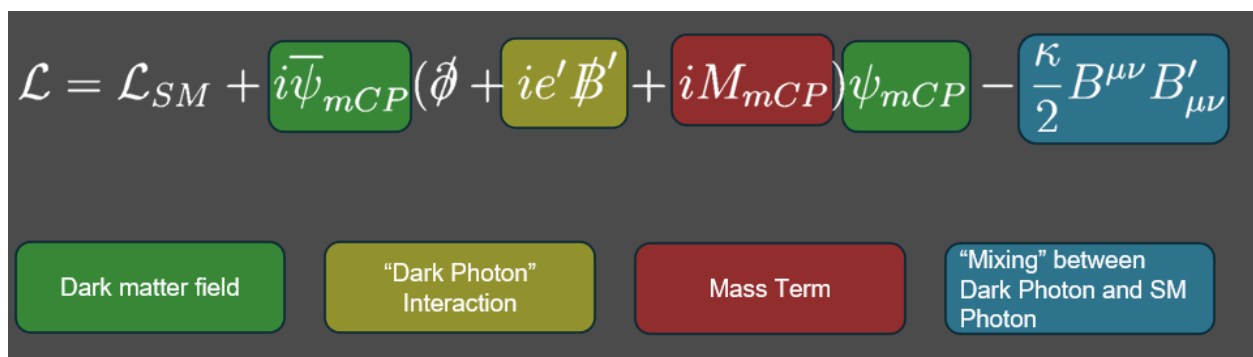
Quantum Field Theory and the local gauge symmetry of the Standard Model,  $U(1)_Y \times SU(2)_L \times SU(3)_C$  works remarkably well at predicting particle properties such as the magnetic moment of the electron and different particle decay branching ratios. The SM is so ubiquitous in particle physics that, since the discovery of the Higgs boson, looking for new physics usually entails having observations that deviate from SM predictions. The SM's extreme accuracy in our particle experiments is diminished by one large flaw: cosmological inconsistencies. While we are able to understand the collisions happening in our detectors, various pieces of evidence suggest that there is more matter in the universe than can be explained by the SM alone.

One such example is the Bullet Cluster gravitational lensing. Gravitational lensing is a consequence of matter curving spacetime via general relativity. When looking at merging galaxy clusters, we can utilize lensing of background objects to predict the matter distribution of the event. We can additionally use traditional telescopes to predict the mass distribution of baryonic



luminous matter via optical and X-ray techniques. A discrepancy arises when the mass inferred from gravitational lensing and luminous mass do not align. The luminous mass, mostly hot gases, interacts with each other electromagnetically and slows down, clumping together, while the mass inferred from gravitational lensing seems unaffected by the merger. Our SM fundamental particles cannot explain this gravitationally lensed mass. The key trait of this matter that differs from our known particles is the ability to clump gravitationally at the macroscopic scale without clear interaction with the other forces. Our stable particles: up quarks, down quarks, electrons, photons, and gluons cannot satisfy these requirements. Neutrinos could compose a fraction of this matter, but not enough and not cold (slow) enough. This new matter, dubbed dark matter (DM), must follow a set of rules set by not only the bullet cluster, but a myriad of other cosmological observations.

These conditions have led to hundreds of different predictions for the true nature of DM. Common hypotheses include weakly interacting massive particles, supersymmetry, and axions. These additional particles all arise from expansion of the standard model by adding additional complex terms to our lagrangians. Some theories of dark matter assume that the bulk missing mass is contained within one particle type. However, it is equally valid to consider DM to be a complex set of interacting particles, a so-called dark sector of matter. Under these assumptions, we can introduce new gauge fields and look at the consequences for detection. One such possibility is introducing a new  $U(1)'$  field, a so-called Dark-Electromagnetism. The consequences on the SM can be seen by looking at the modified Lagrangian below. This new  $U(1)'$  field introduces new massive fermions that satisfy the Dirac equation, along with a new “dark charge”  $e'$  which couples to a new “dark hypercharge field”  $B\text{-Slash}'$ . This is just like how electrons interact with the photon in the SM. As our theory now contains a  $U(1) \times U(1)'$  we can additionally add a coupling term between the original and new photon fields. This term is scaled to a dynamic variable  $\kappa$ . A theoretical calculation of the value of  $\kappa$  can be conducted using single loop corrects which predict:  $\kappa \sim \alpha/\pi \approx 0.00232$ . Experimentally, we still measure a wide range of values of  $\kappa$  up to  $e/6$ .

$$\mathcal{L} = \mathcal{L}_{SM} + \overline{i\psi_{mCP}} (\not{\partial} + ie' \not{B}') + iM_{mCP} \psi_{mCP} - \frac{\kappa}{2} B^{\mu\nu} B'_{\mu\nu}$$


The diagram shows the Lagrangian  $\mathcal{L} = \mathcal{L}_{SM} + \overline{i\psi_{mCP}} (\not{\partial} + ie' \not{B}') + iM_{mCP} \psi_{mCP} - \frac{\kappa}{2} B^{\mu\nu} B'_{\mu\nu}$  with four terms highlighted in colored boxes. Below each box is a label in a matching color box:

- Dark matter field** (green box):  $\overline{i\psi_{mCP}}$
- “Dark Photon” Interaction** (yellow-green box):  $ie' \not{B}'$
- Mass Term** (red box):  $iM_{mCP} \psi_{mCP}$
- “Mixing” between Dark Photon and SM Photon** (blue box):  $-\frac{\kappa}{2} B^{\mu\nu} B'_{\mu\nu}$

When performing a standard field redefinition, a peculiar observation appears. By “diagonalizing” the mixing term, we can observe that within this theory, a suppressed interaction between the dark matter and the photon can exist. This factor contains an interaction between a scaled dark charge ( $\kappa e'$ ) acting as a regular charge to the photon field  $B\text{-slash}$ . These particles would become millicharged particles (mCPs) under our standard definition of charge.

Millicharged particles are interesting, as although they cannot be all of the missing matter in the universe (upper limit of than 0.4%), they still could hint at deeper physics of a hidden sector.

Field Redefinition :

$$B'_\mu \rightarrow B'_\mu - \kappa B_\mu$$

$$\mathcal{L} = \mathcal{L}_{SM} + i\bar{\psi}_{mCP}(\not{\partial} + ie'\not{B}' - i\kappa e'\not{B} + iM_{mCP})\psi_{mCP}$$

Suppressed charge Photon  
Interaction !

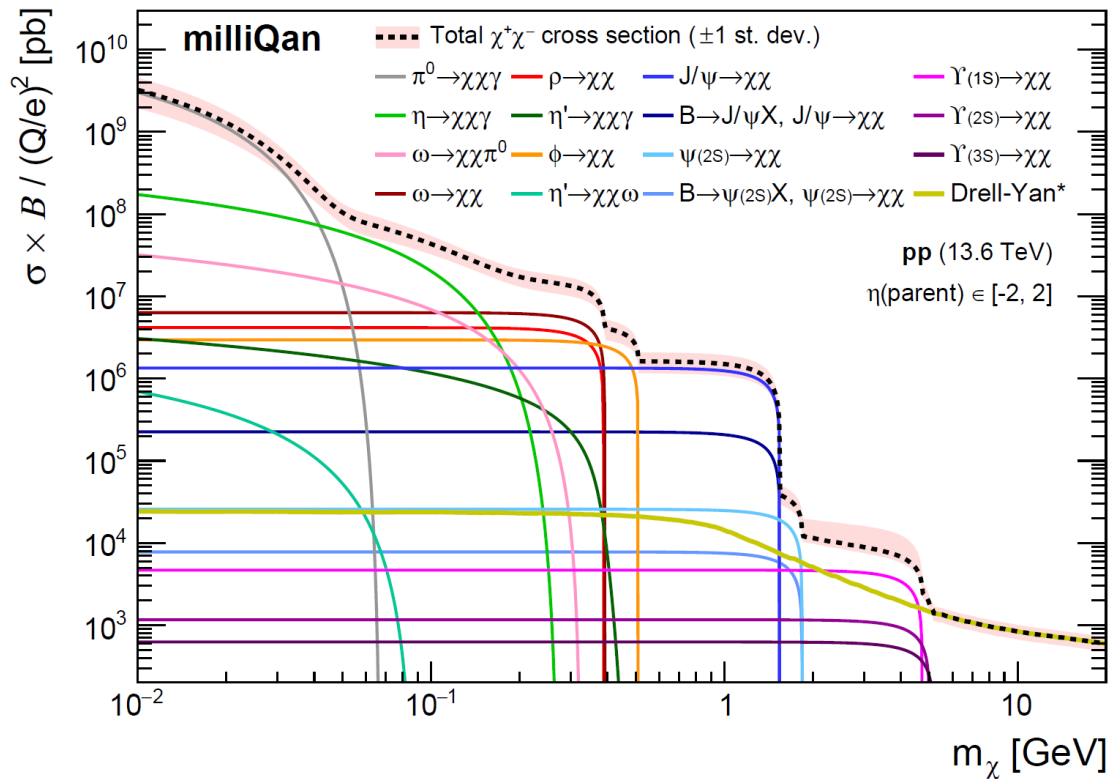
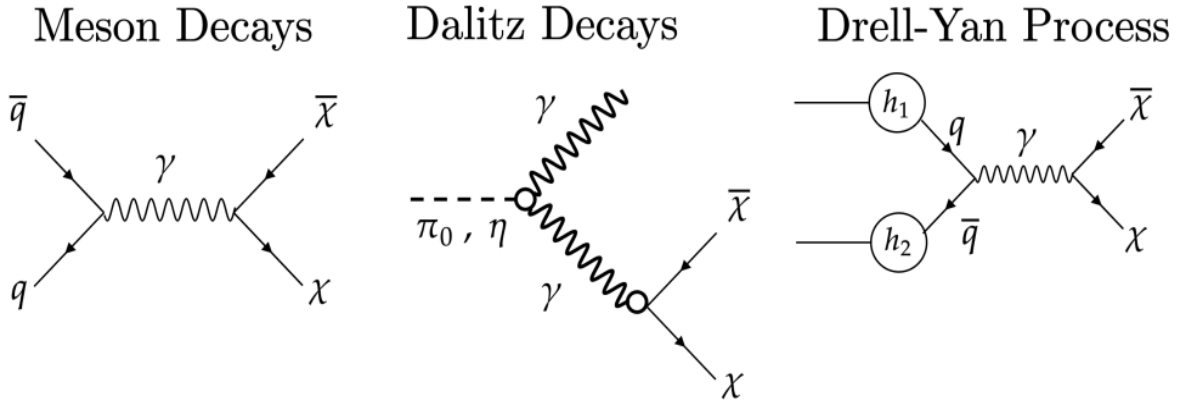
Millicharged particles are particularly interesting as their production methods can be made using many standard QED diagrams since all we have added is another U(1)' gauge symmetry. Since we have now shown that under this theory interactions between SM fields and dark matter can exist, we can now rescale our known interactions and cross sections (described visually as Feynman diagrams) and apply the necessary scaling factor associated with the now smaller charge. This gives us a theoretically simple way to calculate mCP production within our detectors: we first need to have a kinematically valid interaction to create an unknown mass mCP particle (chi). This will give us the mCP production cross section as a function of mass, a sort of probability we use with the runtime of LHC collisions to get the number of mCPs detected as a function of mass. In QED diagrams, the vertex factor that appears at every particle interaction point is proportional to the charge:  $-iQ e \gamma^\mu$ . Where previously  $q$  would be the charge of electrons or quarks,  $\pm 1, \frac{2}{3}, \frac{1}{3}$ , we now have  $Qe = \kappa e$ .

QED vertex:   $= iQe\gamma^\mu$  (A.6)

( $Q = -1$  for an electron)

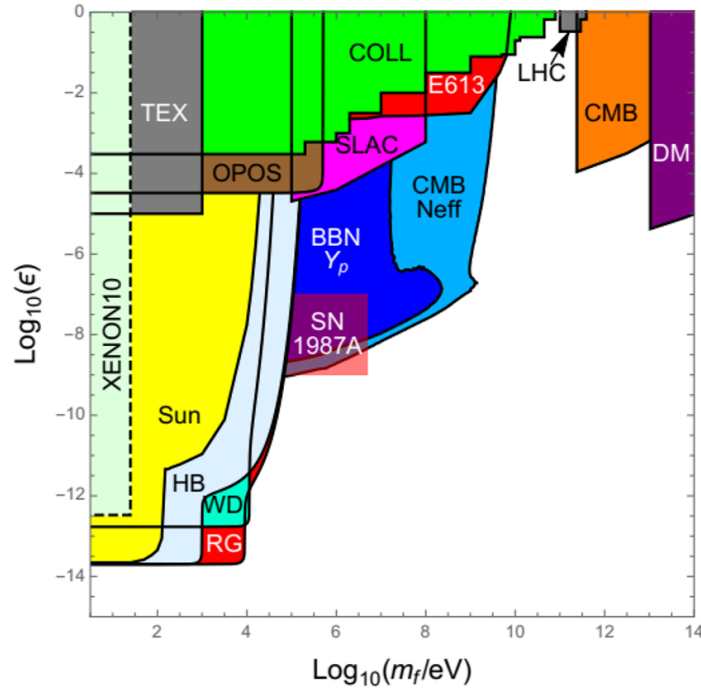
We can then scale down the production and detector efficiency of the mCP with charge. Below is a graph denoting the production of mCPs in slightly weaker 13 TeV proton-proton collisions. Low-mass mCPs can be created in a variety of different neutral meson decay channels. Depending on whether the meson is a pseudoscalar meson or a vector meson, decays proceed through either direct decay or via a Dalitz decay correspondingly. At the higher masses, creation

of mCPs from meson decays becomes kinematically impossible as the decaying particle needs to be twice as massive as the mCP. Instead, we can only produce them from the Drell-Yan process, which is the direct collision of the quarks and anti-quarks in the proton beams, which produces an intermediate gauge boson that has much higher accessible energy depending on the parton energy.

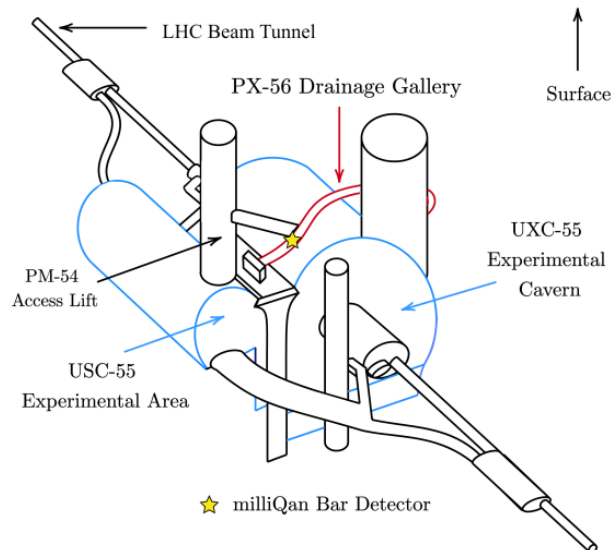


From this graph, we can see that kinetically, the LHC can produce many more low-energy mCPs than high-energy ones. Millicharged particles are characterized by their two fundamental parameters: their mass and fractional charge, epsilon. When looking at the global mCP search effort, that is the previously excluded regions where we believe that mCPs cannot exist. Many of these limits exist not as direct particle searches, but as consequences for observed phenomena, such as SN 1987A, the closest supernova to Earth since 1604. Additionally, properties of the Cosmic Microwave Background also exclude the existence of certain mCPs. Of

course, these limits are not as strong as full direct detection sensitivity, but it does point out a particular region where few existing limits occur. This region happens to be in the electroweak mass range, which is accessible by the LHC.

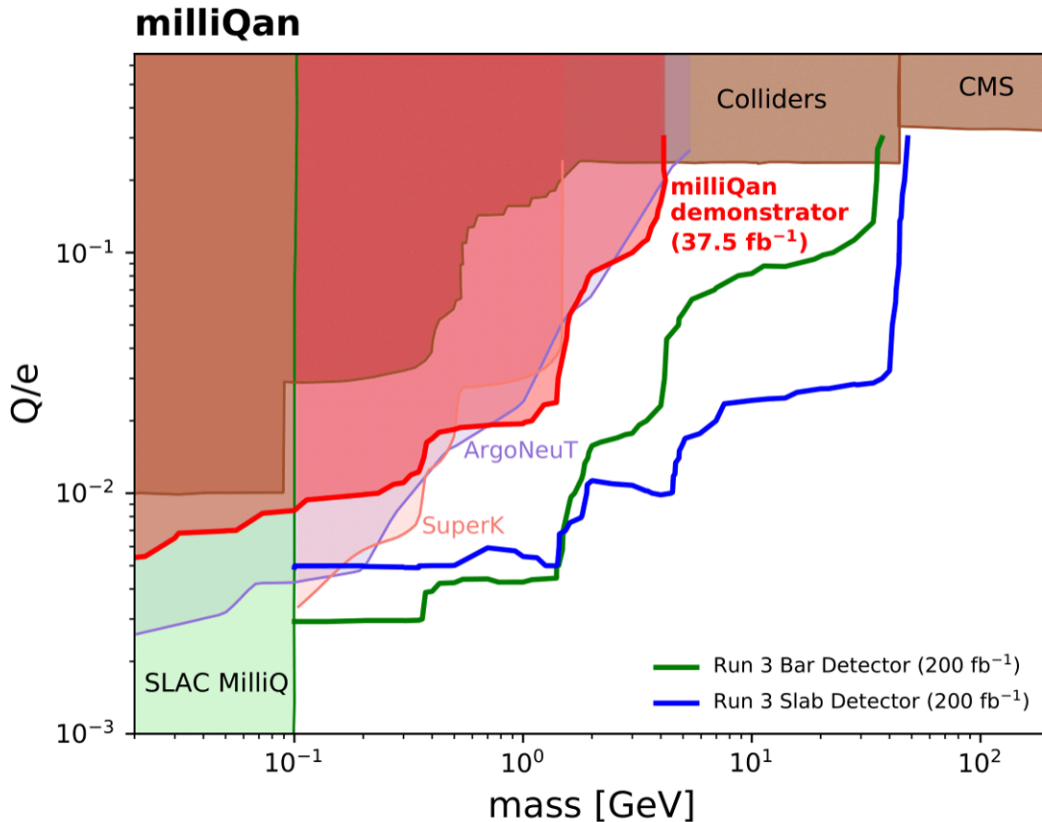


However, although production of these particles can happen at the LHC, that does not mean that the existing general-purpose detectors, CMS and ATLAS, are able to detect them. In their quest to understand a myriad of physical properties, CMS and ATLAS were not designed to search for extremely small signals from mCPs. It is for this reason that in 2016, a new detector experiment specialized for mCPs was proposed: the milliQan Experiment. The milliQan Experiment searches for mCPs produced at the CMS interaction point (IP) in the LHC. As the IP



is surrounded by the 7-meter-radius CMS detector and is located 100 meters underground, careful planning was needed to find a suitable location for mCP detection. The PX-56 drainage gallery, used during the original tunneling efforts, which is located 33 meters away from the interaction point, through 17 meters of rock, was determined to be the best location for milliQan. Being so far from the detection point, and about 70 meters underground, milliQan is shielded from almost all LHC-produced background particles and partially from cosmic events. Shown above is a diagram describing the location of the milliQan Experiment in relation to the CMS cavern. Access to the PX-56 drainage gallery involves descending to CMS -1 floor and climbing 4 flights of stairs, where one will find a small access ladder and door.

For the past ten years, the milliQan Experiment has been developing mCP searches utilizing scintillator detectors. After a proof-of-concept demonstrator detector was built in 2018 during Run 2, the milliQan experiment finalized the design of two detector geometries for Run 3, each probing a different mCP parameter space. Completed in 2023, the Bar detector is designed to target low charge mCPs by utilizing long scintillation bars. Completed in 2024, the Slab detector is designed for high acceptance of flux from the IP by utilizing large slab panels. These designs are complementary to each other and allow for a broad search of mCP parameters. The Bar detector is able to detect the low-mass, low-charge mCPs, which have high production cross section but low detection efficiency. Conversely, the Slab detector can detect the higher charge, high mass mCPs, which are produced infrequently but have large detection efficiency (from the kinematics described above!). Below are the Run 3 projections for the Bar and Slab detectors. Explanations and information on the detectors can be found in the next chapter.



# Chapter 3

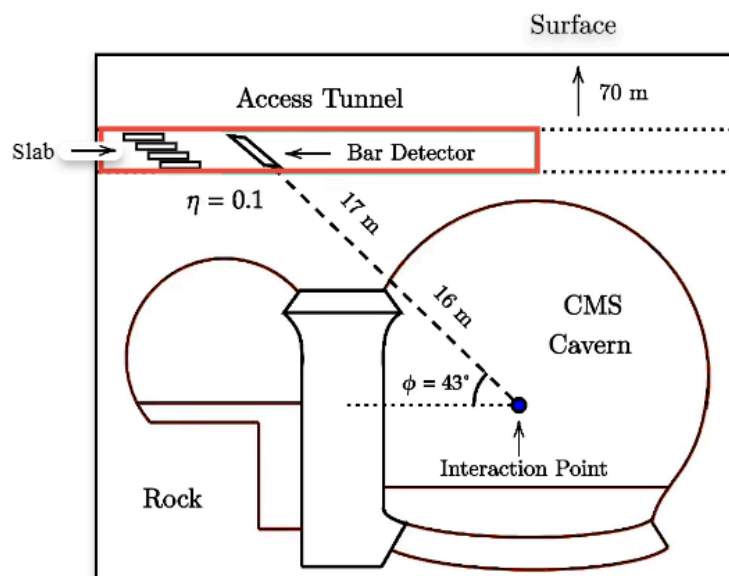


As mentioned earlier, the milliQan detectors are scintillator experiments. Scintillators are a material class that exhibits scintillation, the conversion of ionizing radiation into visible light. Of great importance is that each scintillator has an intrinsic photon yield, the number of photons per MeV of energy deposited. When a charged particle flies through a material, energy deposition can be calculated according to the Bethe-Bloch stopping-power formula. It tells us that energy deposition per unit length,  $dE/dx$ , scales approximately as  $z^2/v^2$ , where  $z$  is the charge of the particle depositing energy and  $v$  is the speed of the particle. In our case, since  $z^2$  becomes increasingly smaller as we reach millicharged particles, deposited energy and photons produced in our scintillator accordingly decrease. Production of mCPs often results in relativistic particles for the masses accessible to milliQan; however, the difference between light mCPs and heavy mCPs will impact velocity and thus energy deposition. However, since we cannot control the mass of the mCP, we can qualitatively reduce the energy deposited in our detector as proportional to  $z^2$ . To detect an mCP flying through our detector, we must have enough photons produced by scintillation. This means we need a certain minimum average energy deposited into the entire scintillator. Since deposited energy scales with  $dE/dx$  and path length, sensitivity to low charge mCPs can be achieved by increasing the amount of scintillator traversed. This leads us to the first principle of the milliQan detectors:

#### Principle #1:

Detection of mCP requires a Single Photo-Electron (SPE) signal to be detected from our scintillator-attached photomultiplier tubes (PMTs, see more below). Smaller charge mCPs need longer interaction length to deposit the energy needed for an SPE signal. Higher charge mCPs (and muons) can use thin slabs and still have large enough signals.

The milliQan Experiment searches for mCPs flying from the interaction point located 33 meters away. Since mCPs have smaller charge, they experience less Coulomb scattering in their journey to our detectors. Furthermore, since the Lorentz force is proportional to charge, they are less affected by CMS's magnetic field. These facts allow for relativistic mCPs to arrive in the access tunnel with minimal deflection from a radial path from the IP. We utilize this fact when designing the milliQan experiments and have aligned our detectors to point in the direction of the IP.

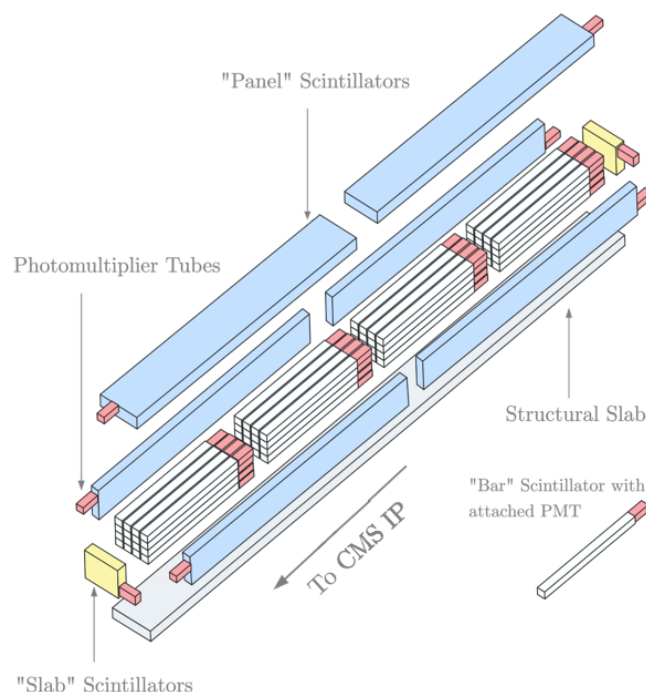


However, imagine if the detector were one big chunk of scintillator placed pointing at the IP with a PMT attached at the end. How could we differentiate the thousands of single-photon events happening in the PMT? Cosmic events vs beam events? Small signals vs intrinsic PMT background? In such a design, it would be impossible to differentiate all the above events.

To remedy this, we can increase the number of channels vertically by splitting the scintillator into four different layers. By stacking these scintillator-pmt layers head to toe, we gain two extremely useful traits. The first is that we can require a signal to light up all four layers of the detector. This allows us to have an extremely basic signal trigger that rejects the majority of dark noise and uniformly distributed cosmic events. It additionally allows us to obtain a direction on the signal, as cosmic events would hit the top layers first, then the bottom layers, and beam events would hit the bottom layers first. However, given that these layers are still large in area, we allow for highly angled trajectories which are not pointing towards the IP to be accepted. By segmenting each layer into multiple channels, we can further narrow down background by requiring signal to fly in a straight path, depositing energy in all four layers. This leads us to the second key principle:

#### Principle #2

Utilizing multiple layers of grid-style detectors allows us to maximize background rejection. Relativistic mCPs (and muons!) coming from the IP will fly in straight lines in the same row and column for each of the four layers. Recording the timing difference of the layer hits, we can determine whether particles are flying upwards or downwards. Our signal trigger then becomes these 4-in-a-row events, which can be considered as (row\*column) number of independent signal event locations.



Shown above is a diagram of the bar detector. It is structurally a 4×4×4 rectangular prism of PMT-coupled scintillator 5cm×5cm×60cm bars. In addition to principles #1 and #2, the bar detector additionally uses scintillator panels (blue) to “shield” our detector from cosmic events by tagging events as non-physics quality. Additionally, the front and back slabs allow us to use principle #1 to gain more information from IP-directed events, specifically the charge. Since the slabs are thin relative to the main detector bars, only muons and high-charge mCPs would deposit enough energy in them, while mCPs would only be detected inside the bars. The bar detector is mounted with ball bearings, allowing for horizontal construction and rotation to point to the IP. Note that the naming of these layers usually starts on a 0 count: ie layer0 - layer3. If anything is named “layer4” it always means the top layer.

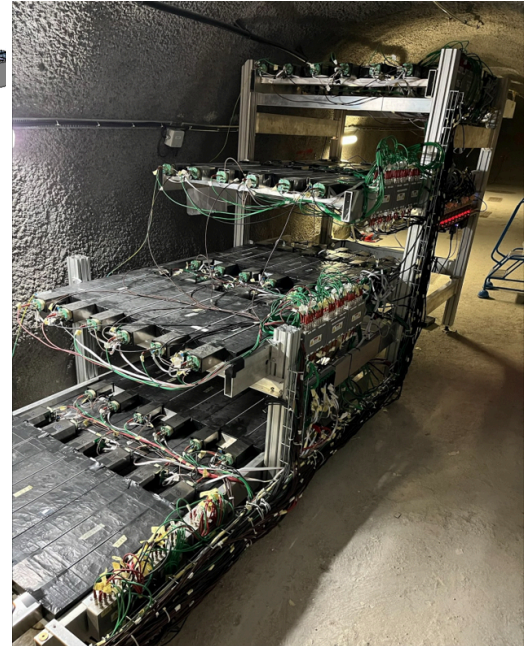
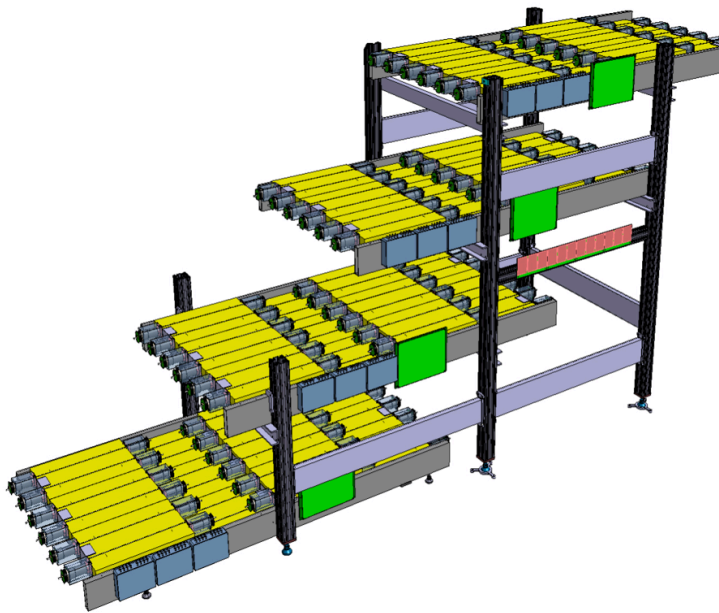
As mentioned in Chapter 2, the mCP dynamic parameters of interest are both the charge and mass of the mCP. We have already discussed how our design can help us probe smaller charges. Since the mCP mass directly affects the accessible production pathways, high-mass mCPs are produced at significantly slower rates than lower-mass mCPs. Considering that milliQan is so far away from the IP, the CMS solid angle reaching the drainage cavern and milliQan is very small. We must then accept that there may be high-mass mCPs we could detect being produced in CMS that will statistically never reach our detector in large enough quantities due to solid angle constraints. The produced mCPs flux cannot be changed; but increasing angular acceptance allows us to accept more high-mass mCPs. Since building a detector and buying scintillators cost money, and we cannot “just” get huge amounts of scintillator and build a massive detector, we are left with the third principle of the MilliQan Experiment:

### Principle #3

High-mass mCP flux is too small for the bar detector angular acceptance to fully satisfy the detector material and readout capabilities. By designing a larger detector that is much thinner, we can detect high-mass, high-charge mCP with lower integrated luminosity (LHC collision time) while keeping construction cost manageable.

It is principle #3 that guides the design of the slab detector. As shown below, the slab detector is constructed of 48 slabs in a 4×3×4 configuration. Each layer contains twelve 40cm×60cm×5cm slabs, each of which has 4 PMTs attached. The increase in angular acceptance has a clear impact on the sensitivity of the slab detector. Although thin slabs can no longer detect the lower-charge mCPs, they outperform the bar detector in charge sensitivity at all masses above the J/Psi decay turnoff ~ 1.5 GeV. Once again, the principle of the slab detector is looking for 4-in-a-row events described in principle #2 in any of the twelve slab paths. The Slab detector's stair-like design also points towards the IP but cannot be adjusted. Each slab layer is separated by a large spacing, which allows for installation of the 48 slab modules from above.

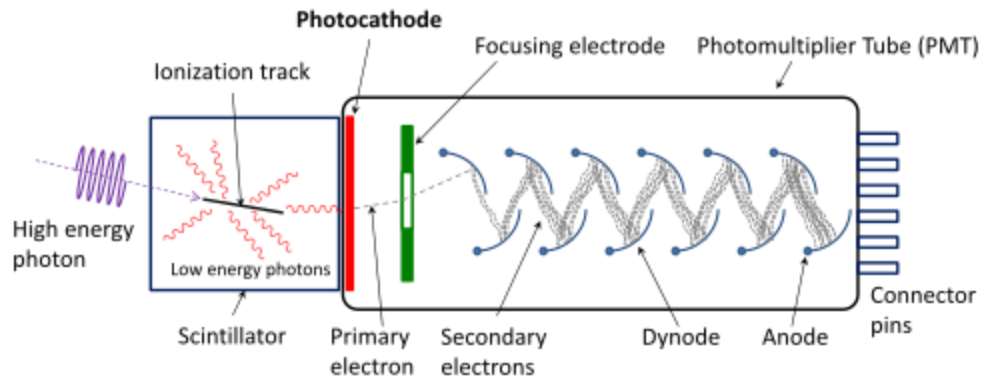
This concludes the general description of the leading ideas behind our detectors. I will now begin describing the specifics of the MilliQan detectors. Although many of the electrical components are used in both detectors, I will focus on the Slab detector application as it is most closely related to my personal expertise as one of the slab detector constructors.



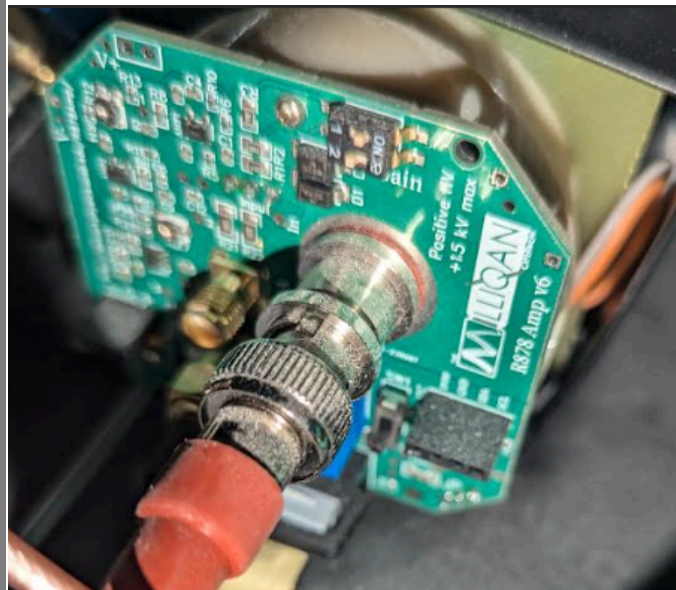
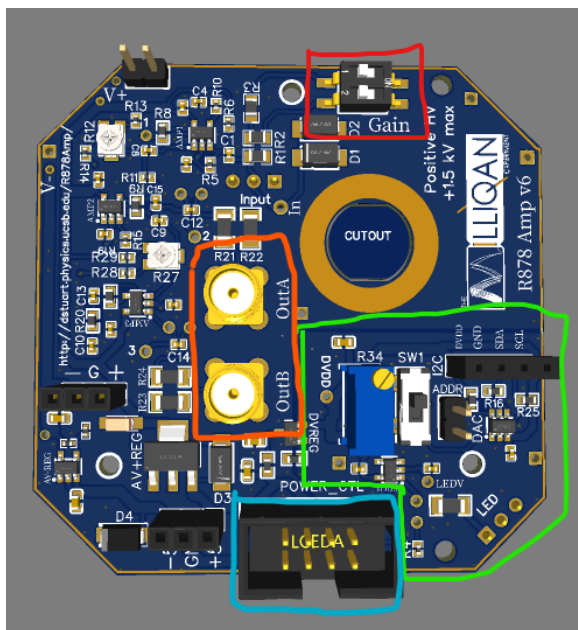
The milliQan detectors utilize EJ-200 plastic scintillators. These are very fast scintillators with a rise time of 1 ns, a decay time of about 2 ns, and a maximum scintillation wavelength of 425 nm. Before use in our detectors, the scintillator needs to be prepared. This is done differently in the slabs and bars, but the main idea was to maximize internal light reflection in the scintillator using reflective or white materials on the first layer and preventing external light by wrapping everything in two layers of thick electrical tape. The EJ-200 is paired with cost-effective Hamamatsu R878 photomultiplier tubes. Photomultiplier tubes are an extremely sensitive light detector, so much so that calling PMTs photon detectors would be a more apt description. PMTs work by first converting photons into electrons via the photoelectric effect at a photocathode. These electrons, called photoelectrons, are then amplified exponentially via cascading dynodes. At each step, the electrons notice a large potential difference and accelerate toward the dynode, in doing so dislodging 3-5 electrons from the new dynode. By having 10 dynodes of increasing voltage, this dislodging process is repeated, and the final electron cloud can be read out as an electrical signal. Our R878 PMTs operate under positive voltage, meaning that the first anode is at 0V and sequential stages have larger voltage up to 1450V. The typical gain of an R878 is one million. The assignment of the voltage at each dynode is custom and is incorporated into a custom PMT base. Below is a PMT diagram along with the milliQan logo where the “m” is created by this avalanche process.







As briefly mentioned, although PMTs can have signal gains in the millions, detection of a single photoelectron event cannot be read out from our R878 directly. Instead, we need to first amplify the signals coming out from our PMTs. This is done using a custom amplifying baseboard, for which a 3D PCB diagram is shown below. There are four important regions for the purpose of this thesis. The 8-pin ribbon cable header circled in blue powers the low-voltage electronics located on the baseboard and allows for communication to the LED flashing hardware in green. The left side of the board contains the amplifying electronics in which the gain can be manipulated using the switches in red. Finally, the amplified signal is output to the SMA connectors in orange. The two outputs differ from each other by a gain of 2, with the “OutB” SMA being the output for physics data.

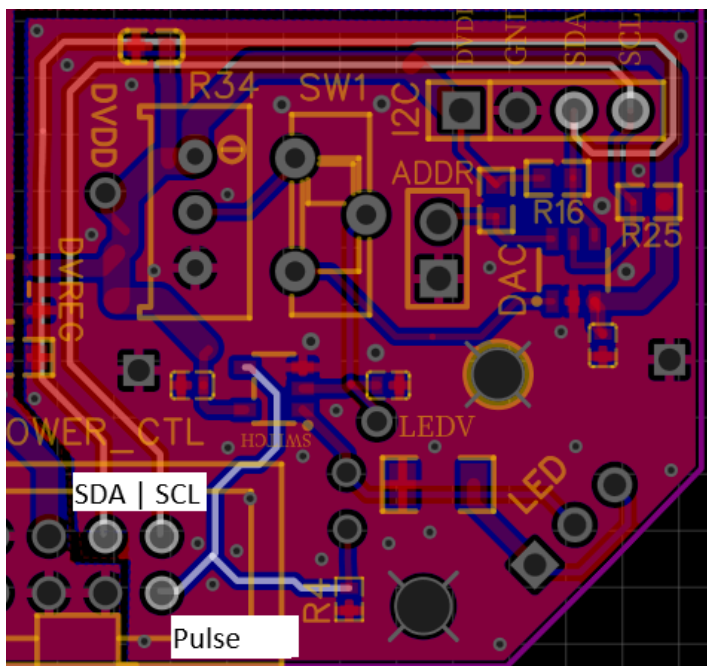


The cutout on the PCB is replaced with a high-voltage bayonet connector, which supplies voltage to the PMT. Hidden under the tan plastic in the real-life photo are a series of resistors and capacitors that implement the dynode voltage settings. The PMT baseboards used in the

Slab detectors are new versions (v5 and v6) compared to the ones used in the bar detector. The older versions contain slightly different IC chips and do not have the same gain switches.

As described above, PMTs get a signal from photons entering the photocathode, ejecting a photoelectron, causing the dynode avalanche. However, electrons can also be emitted from the photocathode by thermionic emission in the presence of photons. Thermal electrons at the photocathode look exactly like SPE events, which are crucial to our analysis. This is both a blessing and a curse, as we can try to calibrate each PMT's SPE response using this thermal dark rate, but we need to also eliminate the dark rate in our mCP search. Additionally, SPE characterisation using thermal electrons is further complicated since thermal emissions happen at any level in the dynode chain. It is for this reason that the LED flashing mechanism was developed for the milliQan detector. By having an LED located next to the PMT face, a fast, low-voltage pulse can turn on the LED for extremely short times, allowing for only a few photons to reach the PMT. Utilizing LED-injected light, the PMT's SPE can be characterized to a higher degree.

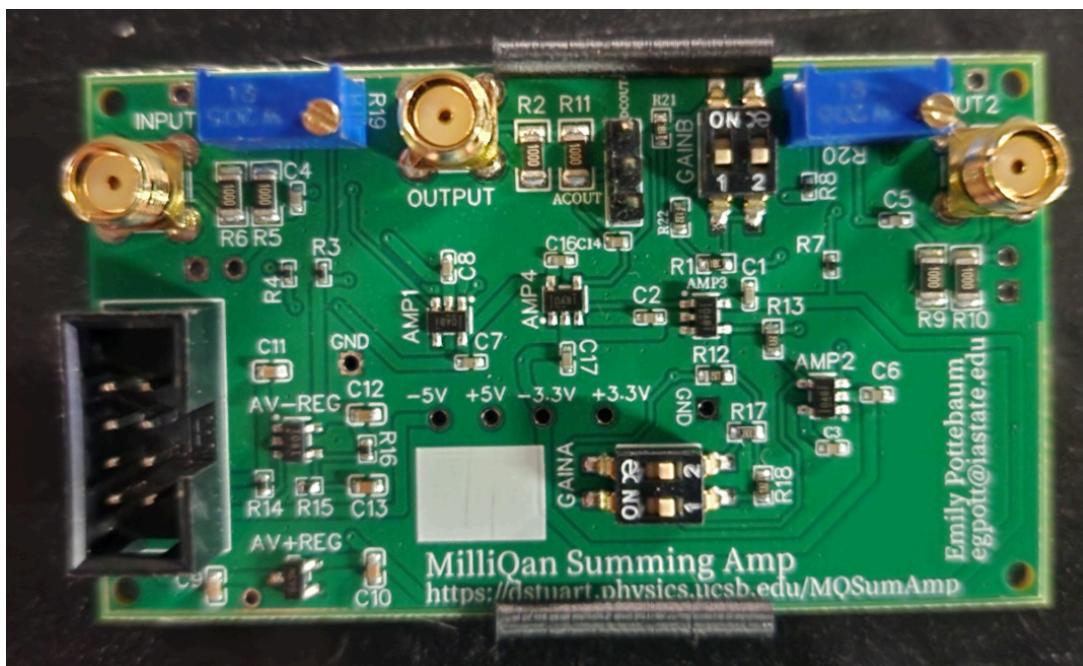
The LED flashing mechanism starts at the ribbon cable header, where three pins are dedicated to controlling the LED: two communication pins for I2C and one “pulse” pin, which can be thought of as the trigger for the LED to be on. The LED voltage is one of two variable values, and it is decided by a switch (SW1). If the switch is in the upward position, the LED voltage is decided by a blue trim-pot; if the switch is turned down, it is set by a DAC that constantly outputs a voltage 0-3.3V depending on internal settings. In any case, the voltage sits idly at a switch waiting for the “pulse” pin. When a pulse is sent, the switch



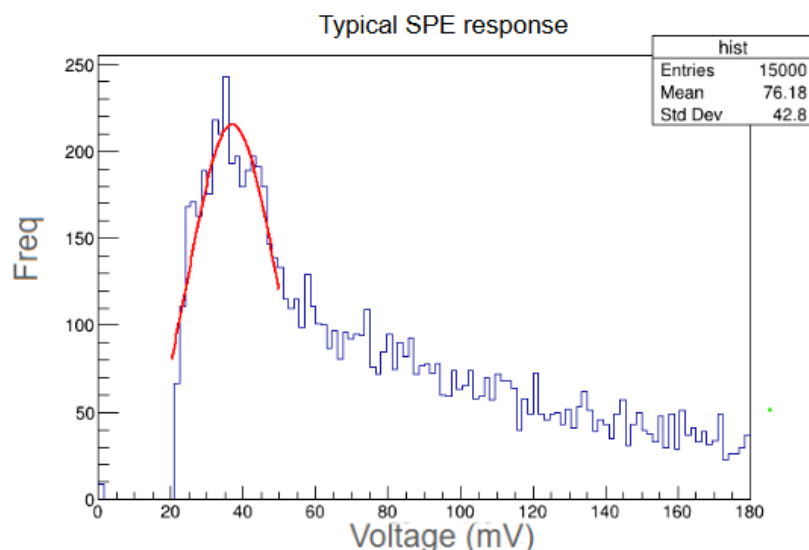
opens, and voltage is passed through a resistor and into the center LED pin on the header. The LED then completes the circuit and is connected to either of the two ground pins on the LED header. The “ADDR” header allows us to change the I2C address of the DAC by one bit, allowing two baseboard LEDs to be used on one ribbon cable (as is the case on the slab detector). During construction, we ensured that one of the two bases going to a summing amp had the address pins grounded.

Unlike in the Bar detector, where each scintillator bar has one PMT, in the slab detector, four PMTs are connected to one slab, but they *do not* behave like four independent channels; the signals of the two PMTs located on the same side are “added” together using a custom summing amp. Thus, only two data channels are obtained from each slab. These summing

amps have two purposes: common gain of the summed signal and channel-specific amplification. One of the difficulties in this approach is that we cannot blindly connect PMT signals together and expect to get a clear output. If two PMTs with very different SPE responses



are connected together, for example, PMT 1 has SPE = 50mV and PMT 2 has SPE = 100mV, it becomes impossible to distinguish two photons entering PMT 1 or a single photon in PMT 2. It is thus essential to match the two channels' SPEs that we are summing together. Specifically, we need that the two SPEs lie within the width of the intrinsic SPE width. A typical SPE response of one of the milliQan PMTs is included. This SPE plot is obtained by collecting and filling 15000 PMT pulses into a histogram. A Gaussian fit is then applied to calculate the best mean of the SPE peak. The strict cutoff below 20 mV is due to triggering thresholds; there are more sub-SPE (noise) events as we decrease the threshold. The thus utilize the channel-specific gain switches on the baseboard and summing amp to match the two SPEs to max/min < 1.35 (about the width of R878 SPE). More information on techniques for SPE matching will be discussed in chapter 4. Note that in the SPE diagram below, the continuation of signal after the SPE is due to true background noise of cosmic events in the scintillator being detected in the PMT.

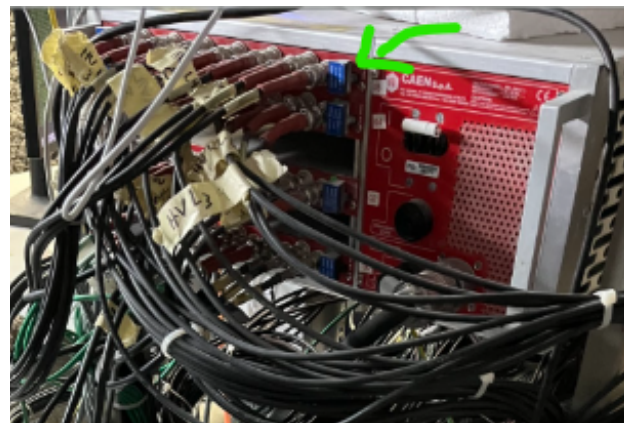
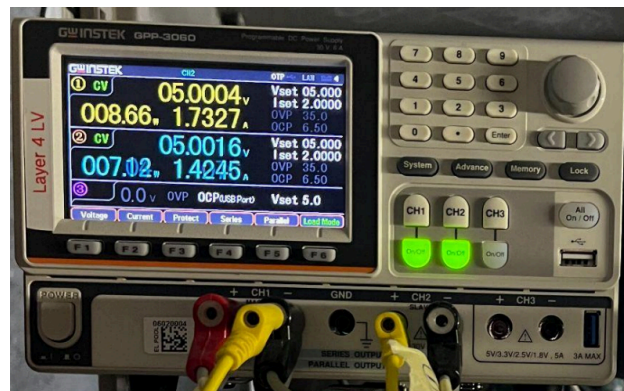
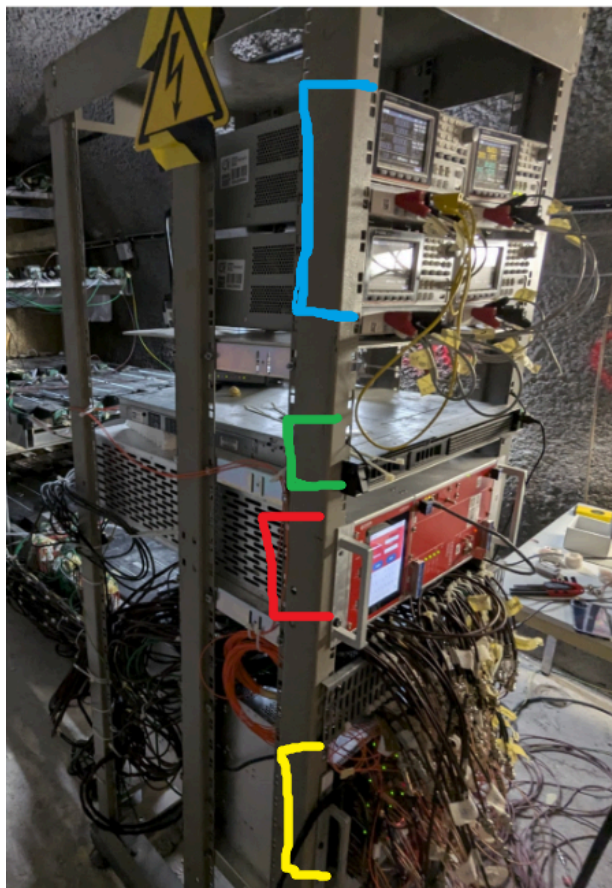


In the slab detector, these two data channels are further combined when triggering the detector, which is used to perform our 4-in-a-row signal trigger. This means that we do not differentiate whether



the front or back of the slab received an event.

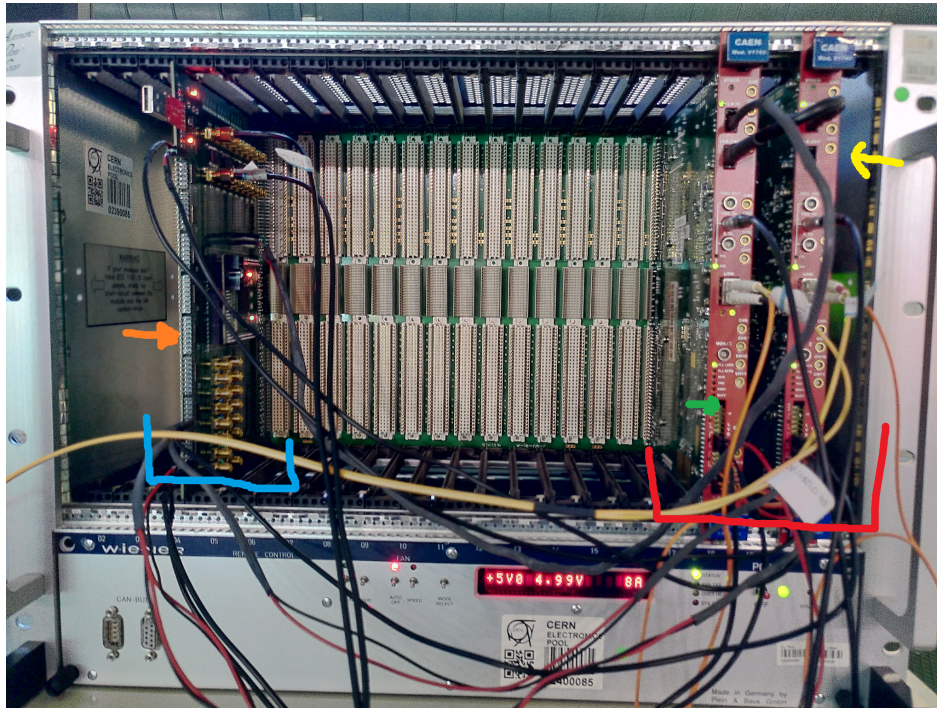
In the final part of this chapter, I will discuss the Slab detector's electrical infrastructure. Located next to the MilliQan Slab detector is a server rack containing various components that power the detector and perform data acquisition (DAQ). The brain of the DAQ system is a Linux PC in green. This is where Slab detector data runs are started. This PC communicates with all the other components using either Ethernet cables or optical fibre. Right above the PC are four identical GW Instek GPP-3060 power supplies, one for each layer of the detector. These two-channel power supplies give each layer its own +5V and -5V needed for all low-voltage operation on baseboards and summing amps. In the image below, a close-up of the top layer LV power supply. Note that the actual nominal current draw is higher than pictured here. Located below the PC in red is the high-voltage power supply crate. Inside this Caen crate, four smaller 12-channel HV blades are plugged in. Each blade controls one layer of the detector, meaning that each slab is controlled by one HV cable. Located on the frame of the slab detector are switching boxes that power four PMTs from one input cable.



Finally, we have the VME crate (yellow). VME is a modular computing bus that allows for high-speed communication between different plugged-in modules. In our case, we have Ceac digitizers and a custom FPGA-based trigger board. Included below is an early photo of the MilliQan VME crate prior to the introduction of all the modules and signal routing.



As we can see, the VME crate operates like a rack with many plug-in bus connections. In this photo, two CAEN Mod. V1743 digitizers are plugged into the crate (red). These digitizers have 16 channels and can thus accommodate 8 slabs' worth of data. In total, 6 digitizer modules are used for the Slab detector. Signal input is conducted via the MCX connectors (yellow arrow). These digitizers have two important outputs: digitized waveforms, which go to the PC via an optical fiber link, and trigger primitives, which are sent from the LVDS outputs (green arrow) to the triggerboard LVDS input (orange arrows). The trigger primitives do not cause a trigger; they are just a sign that a specific channel has gone above threshold. All 96 channels are used in conjunction on the triggerboard to actually trigger the detector.



When performing physics searches, the coincidence window of these trigger primitives is 80 ns. This means that we need to have a satisfying set of trigger primitives within 80ns, or the trigger will not fire. The triggerboard implements many different backup and commissioning triggers beyond our 4-in-a-row.

```
...
Triggers:
  1. Four Layers
  2. Three in Row
  3. 2 Separate Layers
  4. 2 Adjacent Layers
  5. N Layers
  6. External
  7. N Hits
  8. Internal
  9. 4InRow
 10. FORMOSA
 11. Front/Back Panels
 12. S Particle (not used for bar)
 13. Zero Bias
 14. 3InARowAND4LayersHit
...
```

The current trigger suite of the slab detector is detailed in the 14 trigger bit names. There are two different types of trigger classes: data-first and external.

Data-first triggers are direct results of trigger primitives. These are triggerbits 1,2,3,4,5,7,9 and 14. External trigger bits are caused by factors outside of the digitizer signal, like an external system. These are triggerbits 6,10, 12, and 13.

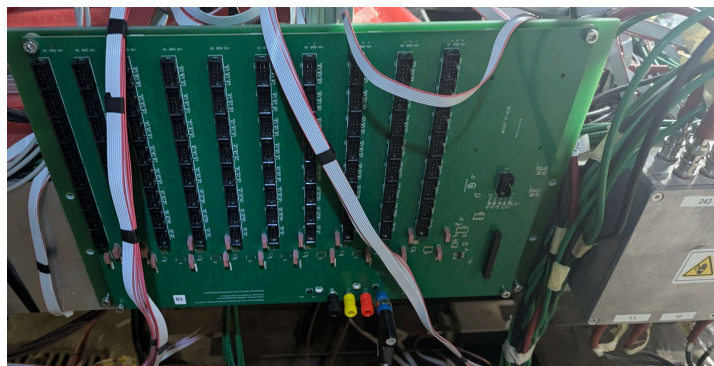
In both cases, once a trigger fires, the triggerboard will request a new event, which causes all digitizer information to be dumped into the PC. At the same time, the triggerboard itself saves information about the trigger bit (or multiple bits) that were fired. In each system, data is written in 1000-event-long files, which we need to match back up together in offline processing.

One problem with all of these trigger bits is that the number of possible trigger primitive combinations is not equal across all triggers. If we ran the detector with all of these triggerbits turned on, we would have millions of events per second, as the combinatorics of 96 channels each with their own dark noise becomes quite large. That is why almost all data-first triggerbits are prescaled, meaning that only a small fraction of the time will the triggerboard fire an event when they are detected. For example, the “2 Separate Layers” triggerbit is prescaled to  $1 \times 10^{-7}$ , one in 10 million. Of the data-first triggers, only triggerbit 9 (signal) and 14 (backup signal) are allowed to be fully efficient, or as we call it, unprescaled. The external triggerbits are usually left unprescaled.

I will now go over the triggerbits that need a little more explanation:

The number of layers needed for triggerbit 5 and hits needed for triggerbit 7 can be chosen in a firmware setting. The external (6), internal (8), and Front/Back Panel (11) triggerbits are unused. Triggerbit 10, FORMOSA, refers to an experimental hodoscope, or particle tracker, that was installed on the slab detector in 2026 utilizing scintillator bars. Triggerbit 10 has never been used to trigger the Slab detector; the FORMOSA-scope is triggered using a spare PMT SMA output. Triggerbit 12, S-Particle, was originally designed to be used for an auxiliary S-Particle (spin-less charged particle) search using the milliQan Slab detector but is currently being used in conjunction with the LED flashing system described in Chapter 5. Finally, we have triggerbit 14, Zero-bias, which operated periodically in clock cycles of the triggerboard, nominally once every two seconds. This essentially gives us randomly sampled data without bias of PMT SPE value and trigger thresholds.

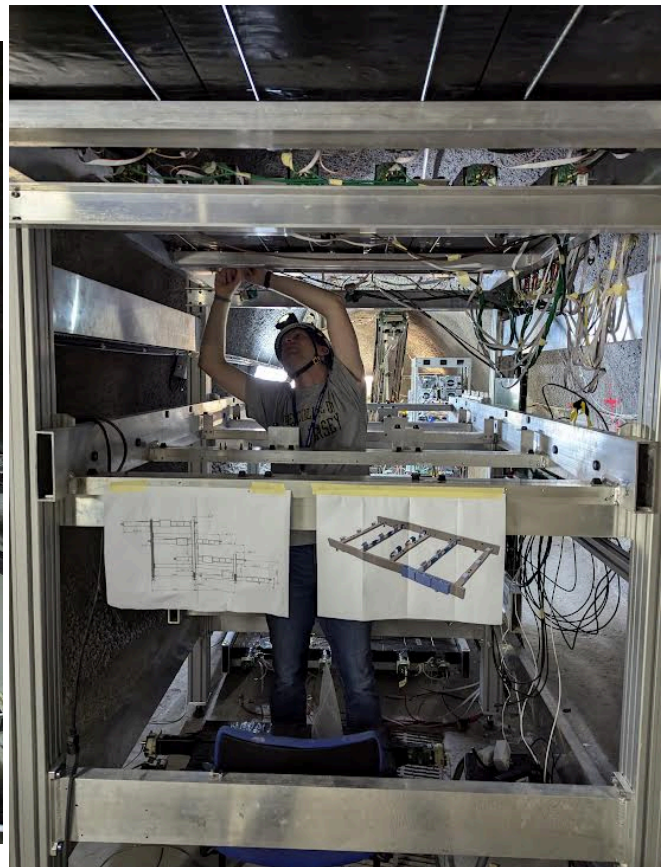
The final detector-specific item I want to talk about before construction information is the LV distribution. As described earlier, each layer of the detector is powered by a separate power supply. Banana cables are routed to the LV distribution boards, where all the ribbon cables are plugged in. There used to be four LV-distribution boards located on the side of the Slab detector prior to the installation of the replacement PCCS system described in Chapter 5. An HV splitter box can be seen and is located to the right of the distribution board. Each ribbon cable used in the slab detector contains 4 headers: one for each PMT, one for the summing amp, and one for connection to the distribution board.



# Chapter 4



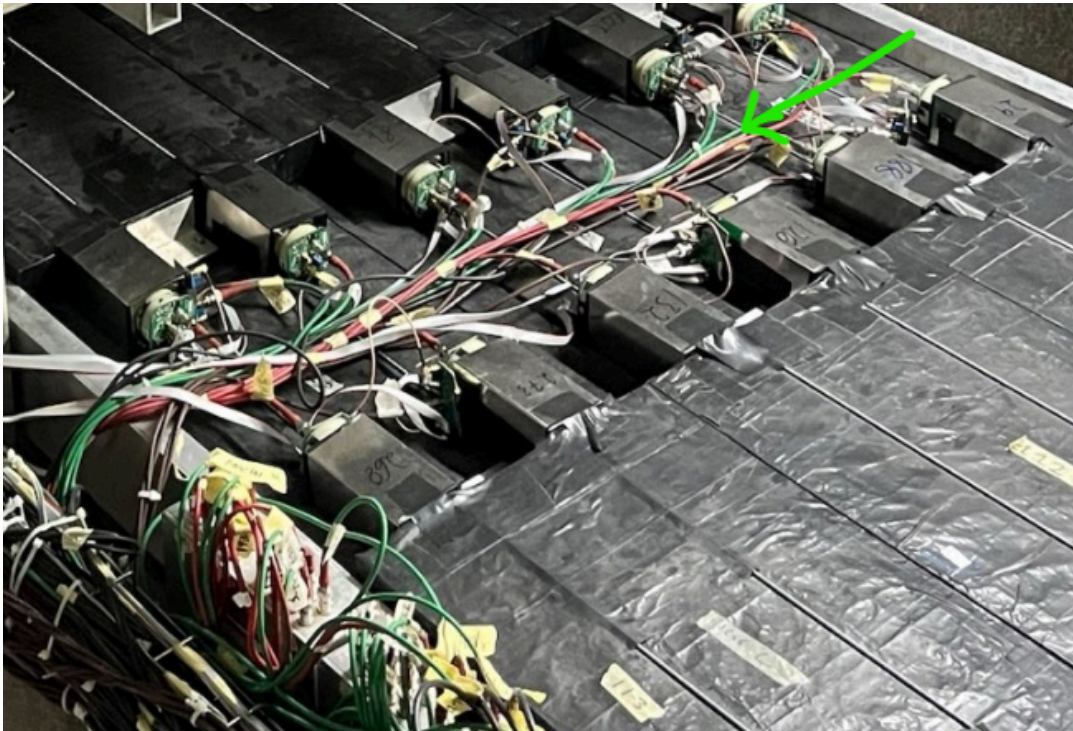
During the 23-24 academic year, I started working on the milliQan Experiment, developing intuition and prototyping automated signal injection using LED flashing on the milliQan baseboards. At the start of the 2024 summer, I had the opportunity to travel to Europe and work on the construction of the milliQan Slab detector alongside an international team of collaborators. I arrived at CERN on June 16th, 2024, where I promptly took the self-mask rescue training needed to enter the underground segments of the LHC on the 17th. After familiarizing myself with the UCSB offices and MilliQan overground lab/storage area, formal construction on the slab detector started the week of June 24. The main summer construction and commissioning team consisted of Zubair Bhatti, Michael Carrigan, Darisuh Imani, Ian Reed, Giacomo Zecchinelli, Matt Joyce, and myself.



When we arrived in the drainage gallery, the slab detector was not much more than a metal frame. Only six slabs had been previously installed and wired to our DAQ system. The majority of the slab modules were lying behind the Slab detector towards the end of the tunnel (pictured above). These modules were the “ready to install” slabs, prepared previously by Sai Neha Santpur and Ryan Schimtz. Of the total 48 slab modules, 25 still needed to be dressed in Mu-metal, a special metallic shielding to protect our PMTs from the influence of CMS’ magnetic field, and 7 slabs still needed SPE matching described in Chapter 3. There were a variety of signal and HV cables that still needed to be made as well. Above is a photo of the Slab detector at the start of summer; Layer 1 and Layer 2 are empty, allowing Ian to climb inside the structure to cable the slabs on Layer 3.

At the start of summer, the collaboration had set the goal to finish installation and commissioning of the entire detector by the time Dariush and I left for UCSB. To enter the drainage cavern, CMS required two members to always be present in case of medical emergencies. Since Dariush and I were the only ones working 100% on the slab detector construction during the summer, progress became much slower once people needed to find aligning times in the schedule to enter the cavern.

We first wanted to finish one layer to start utilizing the new DAQ software adapted from the previously constructed bar detector. We chose layer 3 to be the ideal first layer as it did not impede the construction of the other layers and kept the inside of the metal frame accessible.



Installation of each layer followed a row-by-row procedure. Row 2 and row 4 are inset into the detector frame and thus needed to be installed first. A photo of a completed layer shows row 2 (green arrow) being almost fully covered up by rows 1 and 3. Note that

the layer number listed on the slabs is on a one count, not zero count. Slab 111 is in the bottom right corner of the bottom layer, Slab 443 on top left of top layer.

|                   | Layer 1 |     |     |
|-------------------|---------|-----|-----|
| back of cavern    | 143     | 142 | 141 |
|                   | 133     | 132 | 131 |
|                   | 123     | 122 | 121 |
| front of detector | 113     | 112 | 111 |



The installation of each row of slabs requires optimally 3 people and was as follows:

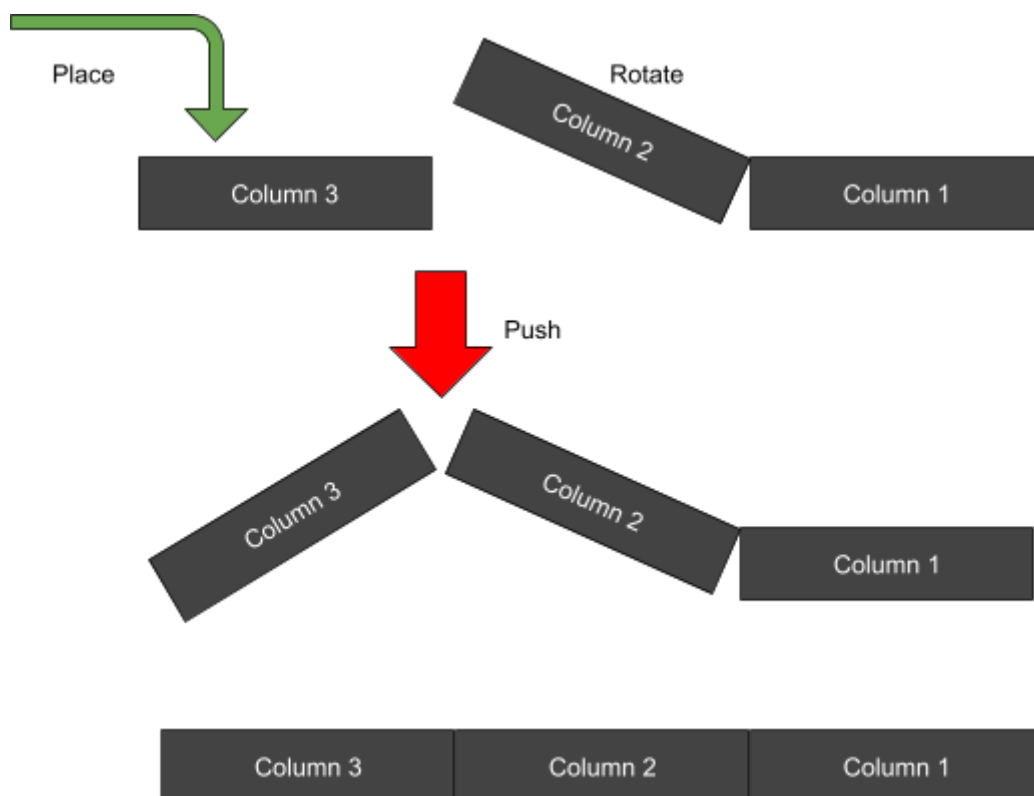
1. Place either column 1 slab or column 3 slab onto the detector. It is best practice for the first placed slab to be the hardest to place as there is still the most room.
  - a. Make sure that the slab is touching the black plastic spacers and metal stoppers located in between the two PMTs
2. Then bring the column 2 slab and place it in the center location, pushing it towards the installed slab.

The tolerances between each slab made it impossible to set the last row easily without ripping the black tape and introducing light leaks. We had to set the last two slabs at the same time using a special technique

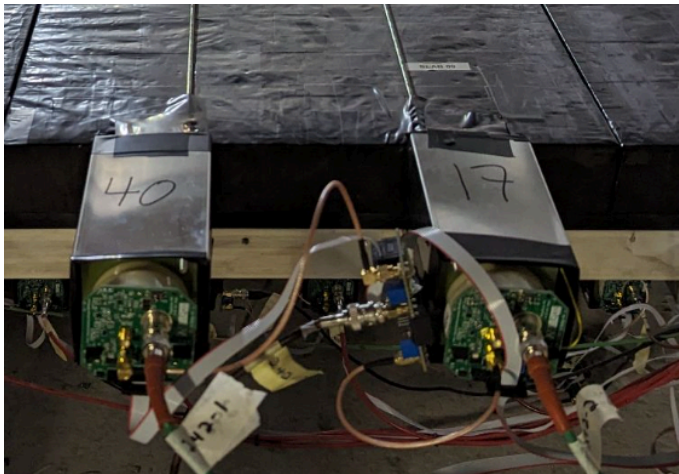
3. For the last slab, have one person lift the column 2 slab to ~ 20 degrees and place the third slab onto its black plastic spacers.
4. Then angle the 3rd slab up until a triangle shape can be formed with the two slabs, and they can almost hold themselves in their angled positions
5. At this point, slowly angle down the two slabs until they are flush with the detector.

Two people should perform the lifting and rotating of the slabs while a third person makes sure that none of the slabs are ripping themselves on any part of the detector frame.

Below is a diagram showing the final installation steps of a single row. Here, the column 1 slab was placed first.



After placement of a row of slabs, all cables were routed. The first step is to wire the PMT outputs to the summing amp inputs 1 and 2 via SMA cables. Since SMA connections require screwing the cable on, it is best to do this as the first step. The summing amp is attached to one of the two PMTs via a plastic strip and Velcro placed on the PMT's Mu-Metal. Usually, a right-angle SMA connector is used for the further PMT going into Input 1 and a straight SMA connector is used for the close PMT (as shown in the example below). Finally, an SMA-to-BNC adapter is placed on the output of the summing amplifier. A custom-cut BNC cable is used to move the signal from the detector area to the DAQ area, where a BNC-to-MCX adapter is used. Then the LV ribbon cables can be plugged in. They are push-fitted and semi-permanently attached using a little plastic bracket to ensure that the cable stays securely placed. Finally, the labeled HV and signal cables can be plugged into the detector and routed to either the HV-splitter box or the large cable routing tray attached on the side of the detector



The cable tray contains all signal cables and the LV banana cables used in each of the LV-distribution boards. Cables were “braided” together in groups of around three with tape to keep the braiding in place to allow for cleaner routing inside the tray. Cables are routed through the metal grid to enter the tray and then zip-tied as they make their way to the DAQ.

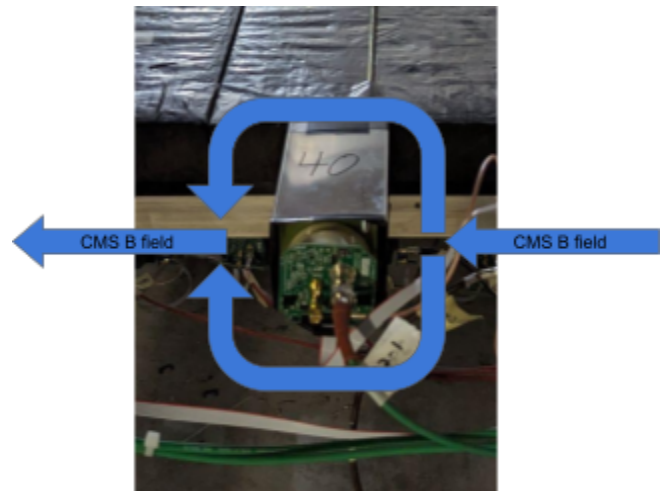


Once wired, signal integrity was checked using a DRS4 Digital Oscilloscope. Every new signal cable was checked on the DRS. For unusually “bright” (high rate) channels, a light leak test was performed by turning the lights off to see if the rate changed. When a light was detected, we further searched for the location by shining flashlights at different angles and locations around the slab while looking at the trigger rate using a. Pictured here is a very large tear in the black tape that happened during the row installation technique. Once we ensured good operation of the new channels, we then connected the signal cable into the MCX port of the digitizer.



Placement of additional slab modules happened only when enough personpower was located in the cavern. When only Dariush and I were in the cavern, we focused on the more menial tasks of dressing the slabs and SPE matching.

The simpler of the two operations, dressing, was the process of placing pre-folded pieces of Mu-metal around the PMTs of the slabs. Since the first photoelectron in the PMT avalanche process is slow, a large magnetic field can easily cause the photoelectron to miss the first dynode. Mu-metal with its very high magnetic permeability can redirect the magnetic field on its edges. This can be verified using the magnetometer built into phones. When the Mu-metal opening is aligned perpendicular to CMS's field, the magnetic reading inside the Mu-metal drops dramatically, and the reading on the surface of the Mu-metal increases as the magnetic field intensity is increased by the displaced magnetic fields. A diagram showing Mu-metal shielding a PMT from the magnetic field is included on the right. Mu-metal is essential to the SPE matching and slab characterization step.

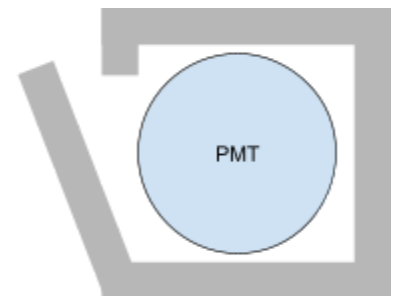


The astute might wonder why, then, so many slabs are missing Mu-metal when only a few of them need to be characterized. This is because Ryan and Neha had performed slab characterization in winter, well after the LHC and CMS' solenoid had been turned off. At that point, only very faint residual magnetic field remained, which our PMTs can survive without issue.

To dress the slabs, the steps are as follows:

1. Grab a Mu-metal plate and three pieces of duct tape, two short and one long one.
2. Open the Mu-metal around the PMT, making sure to write the name/number of the enclosed PMT in Sharpie
3. Close the box by folding the open flap under the lip and covering the long opening with a piece of black tape.
4. Then secure the Mu-metal as close to the scintillator material (and photocathode) as possible.
  - a. To get a good flush seal, one may need to massage the black tape near the PMT's connection to the scintillator
5. Finally, place one piece of black tape connecting the Mu-metal and the Slab on the top and bottom while pushing the shielding.
  - a. Optionally add more pieces of tape to ensure the tape doesn't lift

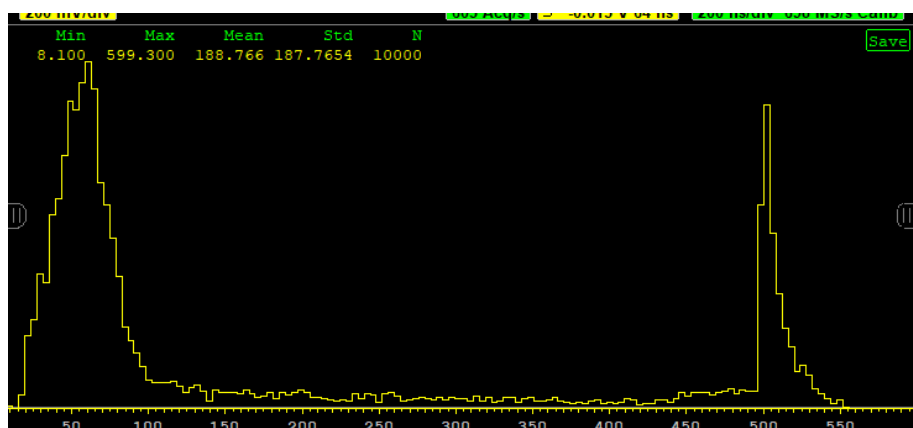
Fold flap under lip and "seal" using electrical tape.



The harder of the two tasks was SPE matching the PMTs on each side of the slab. The importance of SPE matching has already been emphasized previously but cannot be understated. It is so important that if two PMTs were found to be impossible to match within the 1.35 ratio, we would resort to replacing PMTs, a time-consuming task which requires even more time to ensure light tightness.

SPE matching was conducted on a testing table which contained a testing 2-channel power supply and a 4-channel HV power supply (enough for the 4 PMTs). The positioning of the slab on the table was important as we needed to ensure that the Mu-metal was perpendicular to the magnetic field.

SPE matching is an iterative process and heavily utilized the DRS4. We must first get a basic idea of what the SPE of the two PMTs we are trying to match is, with all gain switches turned off. To do this, plug the channel of interest into the summing amp, unplug the other, take a look at the output using the DRS, measure the peak-to-peak height of the pulses, and plot them on a histogram. This would give us a plot looking something like the one below, which is a very clear/pure SPE at ~ 55 mV with saturating pulses at 500mV.



We would mark down our initial value and then start looking at the other PMT. As discussed earlier, we have both switches on the baseboard and the summing amp that we can use to change the gain of the pulses and thus manipulate the value of the SPE peak. When performing this first test, it is normal to have all the switches on the unity (1x) gain.

The two switches on the baseboard attenuate the signal from the PMTs. The notation that I will use is '0' is off and '1' is on, with the first value being switch 1 and the second switch 2. Baseboard Gain table:

- 0 0: 1x gain (default)
- 0 1:  $\frac{1}{2}$  gain
- 1 0:  $\frac{1}{5}$  gain
- 1 1:  $\frac{1}{7}$  gain



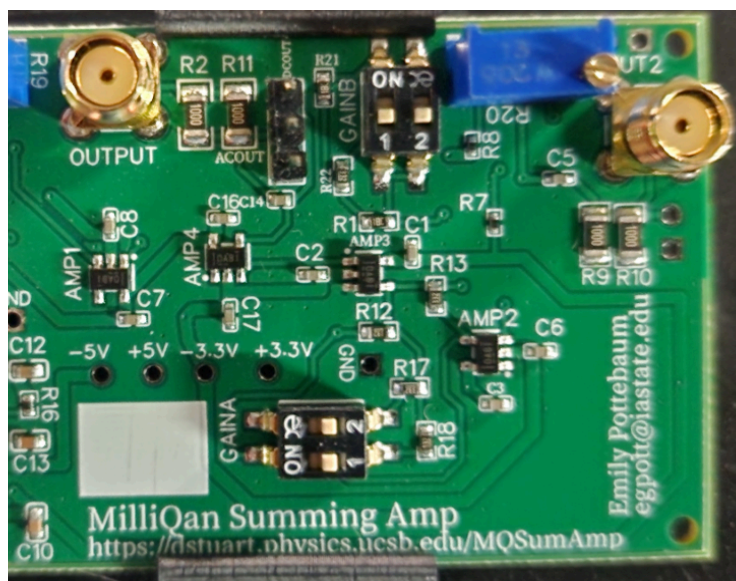


These switches are good when our PMT signal is particularly high. The hidden secondary goal of SPE matching is to match the SPE at the lowest possible value since this allows us to resolve a higher number of photons prior to digitizer voltage saturation.

Then we have the four switches on the summing amplifier. The GainA switches control amplification for input 1 and input 2 separately. Each switch applies a gain of 2x on its respective channel.

Finally, we have the GainB switches, which apply to both inputs. The switch mapping is as follows:

- 0 0: 2x gain
- 1 0: 1x gain (default)
- 0 1: 0.4 gain
- 1 1: 0.3 gain



The values in all of these switch tables are approximate and should be used as starting points for the matching process. For example, switching the baseboard gain to 1/2x and using the GainA switch for 2x gain for the channel should result in 1x gain but in reality can give us 0.8x gain.

This gives us, for a single channel, gain options of: 2x, 1x, ~0.8x, 1/2x, 2/5x, 2/7x, 1/5x, and 1/7x. By combining the gain opportunities of each channel, we can easily deal with ratios (~0.8x, 1.25x, 1.40x, 1.75x, 2x, 2.5x, 2.8x, 3.5x, 4x, 5x, 7x, 10x, 14x) when comparing both channels on 1x gain settings.

Once again, treat these values with a grain of salt, as sometimes the signal switches break the input or make the SPE unresolvable. In the easiest cases, we do not need to move the B switches from their default “1 0” position, but when the only matched SPE values fall above 70mV or below 30mV, we try to bring them back into a good range using the B switches.

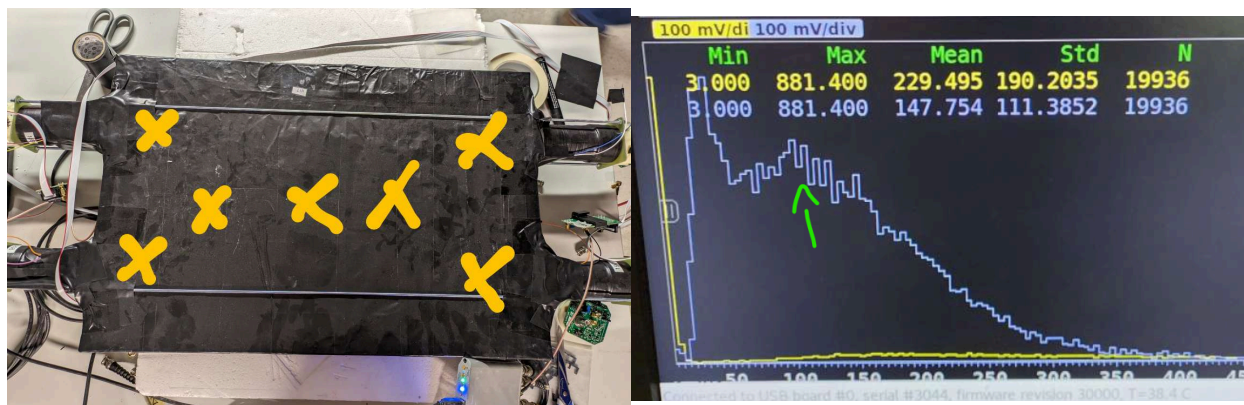
The one final caveat is the potentiometers on the summing amp. These potentiometers control the pedestal (DC bias) of the two PMT inputs. Because of how the summing amp is designed, the negative rail oscillates, which can cause issues depending on the location of the pedestal. The full dynamics of the summing amplifier were not understood during the original construction of the Slab detector, and the tuning of the resistors was an intuition-based problem until the pulses looked high quality

Once we saw what we believed were two compatible SPE distributions, we checked the summed SPE to make sure it was resolvable by plugging both inputs into the summing amp. If it

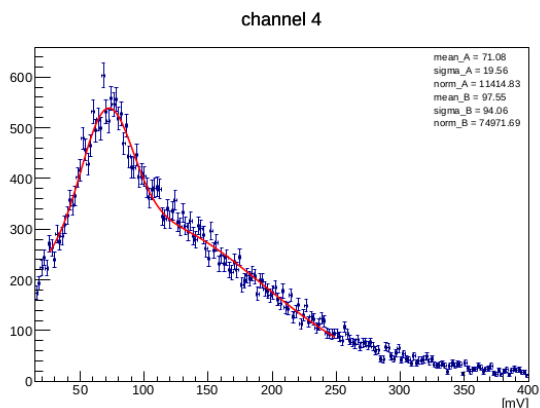


was not resolvable, we would need to try a different matching or even a different summing amp. When we were confident on all sides, we would perform a more thorough data collection of anywhere from 10k-100k pulses and save the data. We would then process the pulse information into a ROOT file for plotting and fitting using ROOT's gauss method. The returned mean of the Gaussian distribution would become the value of SPE for that channel. We then repeated this process with the other channel and compared the values. Assuming that they still lay within a good range, we moved on to the final check of the summed SPE, usually taking a little more data for the combined SPE. At this point, if all the values made sense and were within the criterion of  $\text{max/min} < 1.35$ , we have completed matching one half of a slab. These fitted values were filled out into a spreadsheet.

Once we had completed the two halves of a slab, Cd109 source testing was conducted by connecting both sides of the slab to two channels of the DRS4, resting the source on the slab, and obtaining 50k events triggering on the OR of the two summing amps. This was repeated at seven locations, 5cm in front of each PMT, the center of the slab, and 15 cm from the center point of the PMTs, pictured below. These locations allow us to get an idea about how the location ionizing events effect the signal in the four PMTs. The sources in front of a PMT should be able to recover something more like the source testing of a single PMT, while the center of the slab should be more of a combination between both channels. Source data has an additional source peak (green arrow) which is related to the characteristic energy of the ionizing radiation and detection efficiency of the PMTs. Similar to SPE peaks, these source peaks were fit using a double Gaussian method in ROOT. If source peaks were extremely unresolvable, or too close to the SPE that it would just form a shoulder bump, we would try to increase the number of events, and if nothing helped, go back to the SPE-matching or replace a PMT. An example of a bad source peak is included (bottom right).



When there were more of us in the cavern, installation and cabling were more common, while if it was only Dariush and me, we usually worked through the time-consuming SPE matching and performing source tests on the installed slabs. In-Situ source testing allows us to more easily compare the test bench values to the real setup.



The installation of the Slab detector was as follows (all dates in 2024):

June 26: Start heavy access for construction purposes

July 3: Layer 4 done, Layer 2 slabs dressed

July 10: Layer1 completed, Row 2 and 4 of Layer 2 placed, half of Layer 3 dressed

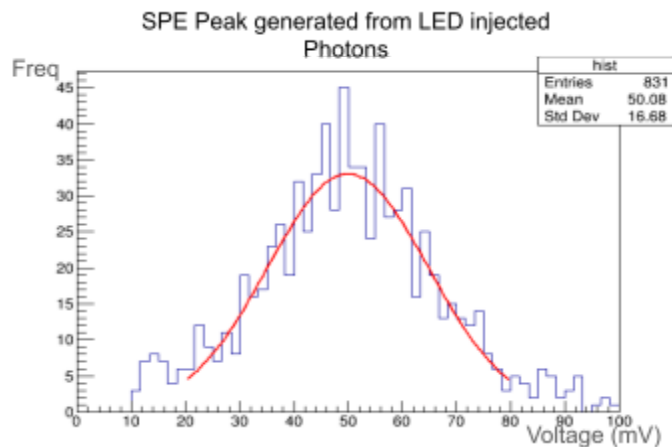
July 17: Layer 2 completed, one row of Layer 3 installed, characterization of slab in progress

July 25: Layer 3 completed; Detector fully constructed and cabled

For information about the slab detector, can be found [here](#) and [here](#).

In the last weeks before Dariush and I left in mid-August, time was spent finalizing in situ source testing and fixing hot channels inside the detector. Additionally, I was able to perform the first signal injection tests on the slab detector using a prototype LED flashing system. Testing was conducted on slab 19, where we were able to gain an extremely clean SPE Gaussian peak with only 830 pulses. This paved the way for future work and eventual installation described in Chapter 5 and Chapter 6.

| Channel Numbers                  |        |        |  |
|----------------------------------|--------|--------|--|
| (Offline Channels - front, back) |        |        |  |
| Layer 0                          |        |        |  |
| 22, 23                           | 14, 15 | 6, 7   |  |
| 20, 21                           | 12, 13 | 4, 5   |  |
| 18, 19                           | 10, 11 | 2, 3   |  |
| 16, 17                           | 8, 9   | 0, 1   |  |
| Layer 1                          |        |        |  |
| 46, 47                           | 38, 39 | 30, 31 |  |
| 44, 45                           | 36, 37 | 28, 29 |  |
| 42, 43                           | 34, 35 | 26, 27 |  |
| 40, 41                           | 32, 33 | 24, 25 |  |
| Layer 2                          |        |        |  |
| 70, 71                           | 62, 63 | 54, 55 |  |
| 68, 69                           | 60, 61 | 52, 53 |  |
| 66, 67                           | 58, 59 | 50, 51 |  |
| 64, 65                           | 56, 57 | 48, 49 |  |
| Layer 3                          |        |        |  |
| 94, 95                           | 86, 87 | 78, 79 |  |
| 92, 93                           | 84, 85 | 76, 77 |  |
| 90, 91                           | 82, 83 | 74, 75 |  |
| 88, 89                           | 80, 81 | 72, 73 |  |



The photo on the left was taken with the completed slab detector. Left to right: Zubair Bhatti, Eliott Schaffer, Dariush Imani. Dinner near CMS with members of the construction team. Left to right: Ian Reed, Eliott Schaffer, Dariush Imani, Michael Carrigan, Matt Joyce.



# Chapter 5

## Assumptions (MilliQan Detector Background, DAC, PMT, I2C)

### Chapter 5: PCCS Introduction and Design

As introduced in previous chapters, the MilliQan Detectors contain the capability of an advanced pulse injection scheme utilising a distributed DAC located on each of the PMTs. MilliQan was, in fact, planned with the utilization of the pulse injection in mind. A specialized low-voltage distribution board (LV Board) that powers the PMT's amplifier boards and allows for injection was originally designed. Due to flaws within the board's PCB design, utilizing the pulse injection became impossible. These LV distribution boards were, however, still capable of powering the detector without the flashing functionality. As pulse injection was a low priority within the MilliQan Experiment, and with integrated luminosity to the Bar Detector steadily increasing, maintaining the working bar detector and analysis took precedence.

The Power Control Calibration System (PCCS) was born as an extension of the LV Board used on the MilliQan Bar Detector. By first understanding the issues, both electrical and mechanical, of the previous LV board, a new design specification was created as the core of the PCCS.

The goal of the PCCS Design is:

#### 1. Fully automated pulse injection:

Correctly implement the necessary control to the over sixty Inter-Integrated Circuit (I2C) interfaced DACs and localization of the pulses.

#### 2. Removable channels

In the previous LV Board, sometimes channels broke during electrical events in the MilliQan cavern. These events include power trips and thunderstorms. As a precautionary measure, a design was developed that allowed quick deployment of new, ready-to-use channels without needing to replace the entire PCB.

#### 3. Detector safety

Not only should channels be removable, but optimally, if some anomalous event happened, there should be enough protection to ensure the safety of the detector. These removable channels should themselves act as the first defense mechanism, localizing issues for minimum impact.

#### 4. Larger spacing between different channels

As requested by the construction team responsible for the Slab Detector's construction, the inter-channel spacing on the current LV Board made it hard to cable.

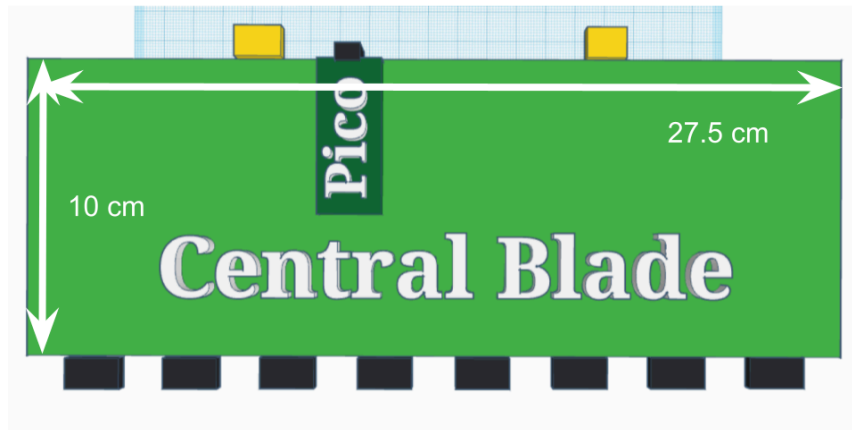


The challenge in designing the PCCS came from the sheer number of devices connected. Integral to the pulse injection is the DAC located on each baseboard, which allows for custom voltage within each of the injection sites. These DACs communicate via I2C [See Appendix], utilising hardware addresses to communicate with specific devices on a 2-wire protocol. I2C only functions when there is exactly one device with the same address connected on a bus. For the Bar Detector's 72 channels, there were 72 DACs, all with the same address, which would lead to communication failure if all were wired together. Thankfully, I2C multiplexers, which allow for the choice of one of eight different isolated I2C buses have been created and can be utilized. These devices, I2C-interfaceable themselves, only had two available address bits, allowing for a maximum of 32 I2C buses. As this was not even half of the total number of channels required, further care was taken to allow more channels. Possible solutions included different daisy-chaining architectures and the utilization of multiple different multiplexer devices with new addresses.

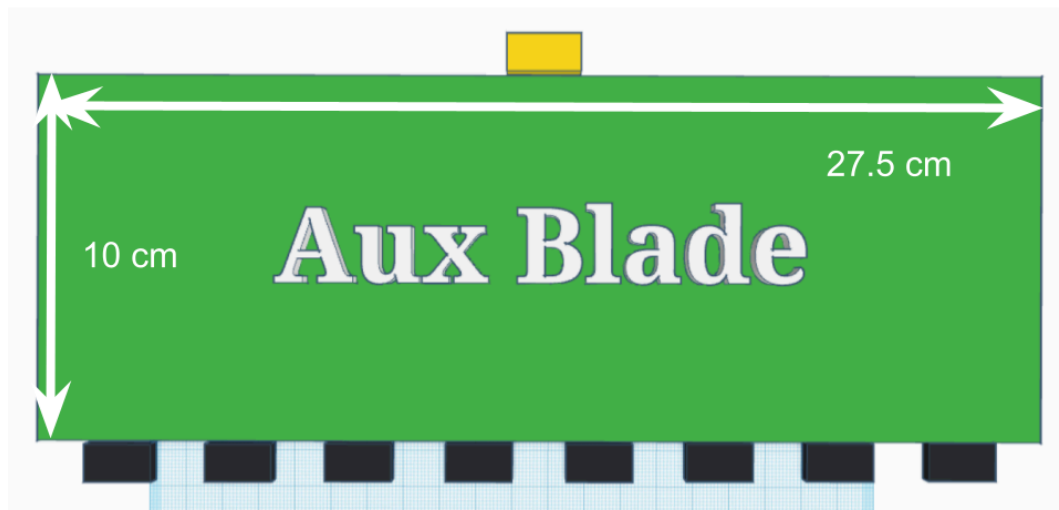
Additionally, for fully customizable injected signals, we should have the freedom to set any combination of connected outputs. If we have 72 outputs, that would require 72 GPIO Pins without using multiplexers, which would further increase the complexity of such a system. Utilizing I2C GPIO multiplexers would further increase the I2C overhead and would lead to additional I2C multiplexing issues, as they would again share addresses.

For the PCCS, a hierarchical method was utilized, which, instead of incorporating more integrated circuit multiplexers, incorporates independent microcontroller systems to handle the large number of channels. Thus, by utilizing a combination of a single Raspberry Pi 5 (RPi5) and multiple Raspberry Pi Picos (Picos), we could handle more I2C channels, get more GPIO pins, and create a clear segmentation of the system at the Pico level. This led to the key characteristic design of the PCCS, which resembles a backplane and blade servers used in data centers. A single backplane (henceforth motherboard) contains an RPi5, the hardware to create the injected signal, and the routing of signals and voltages for a set of pin headers.

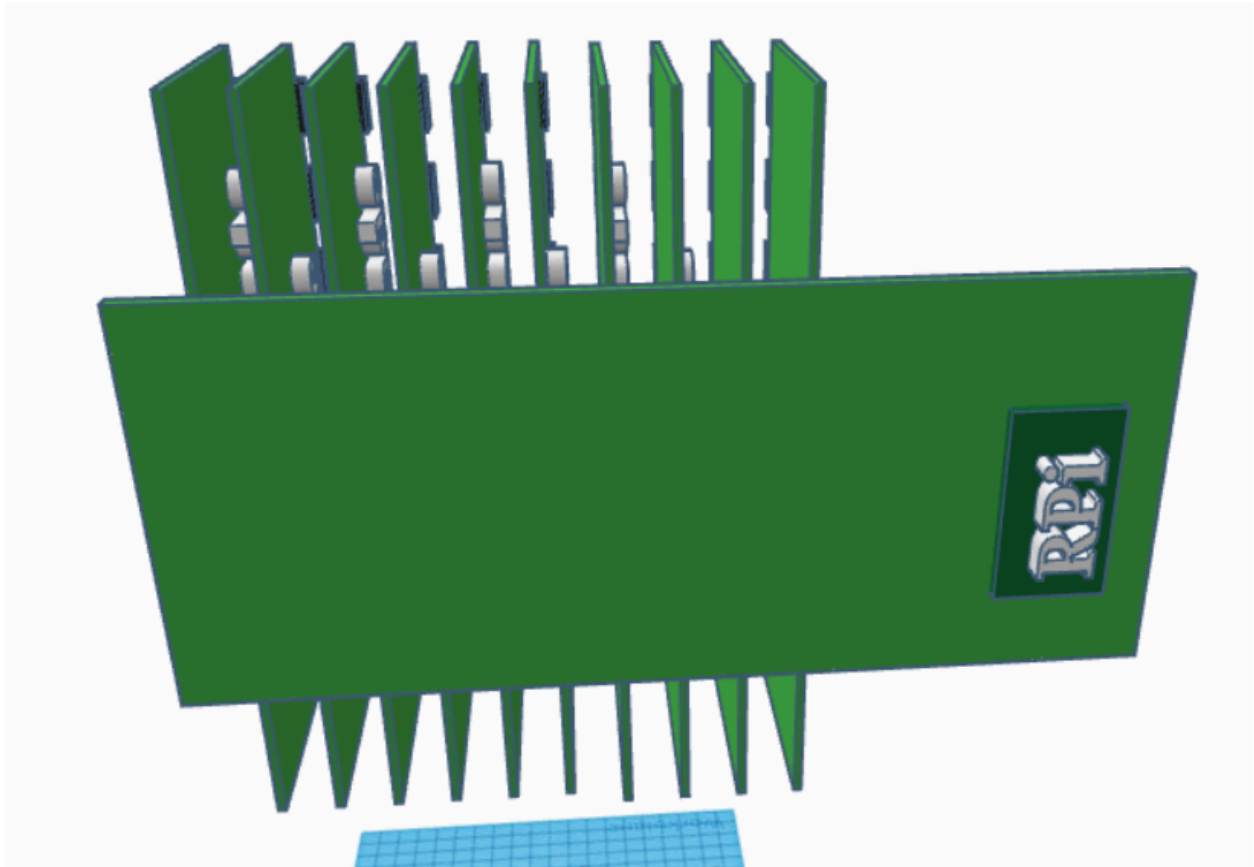
In these pin headers, Blade Pairs can be plugged in. Two Blades make a blade pair: a Pico Blade and an Auxiliary Blade. A Pico Blade contains a Pico microcontroller and directly communicates with the RPi5 for information about the injected signal. The Auxiliary Blade does not communicate with the RPi5 and is interfaced by its respective Pico through the motherboard. Each individual Blade has 8 LV connectors, one 1-to-8 I2C Multiplexer, and one 8-channel 3-state buffer. Thus, each blade pair controls 16 LV channels, equivalent to one layer of the Bar Detector. Up to 5 blade pairs can be connected to one PCCS Module for a total of 80 available LV channels.



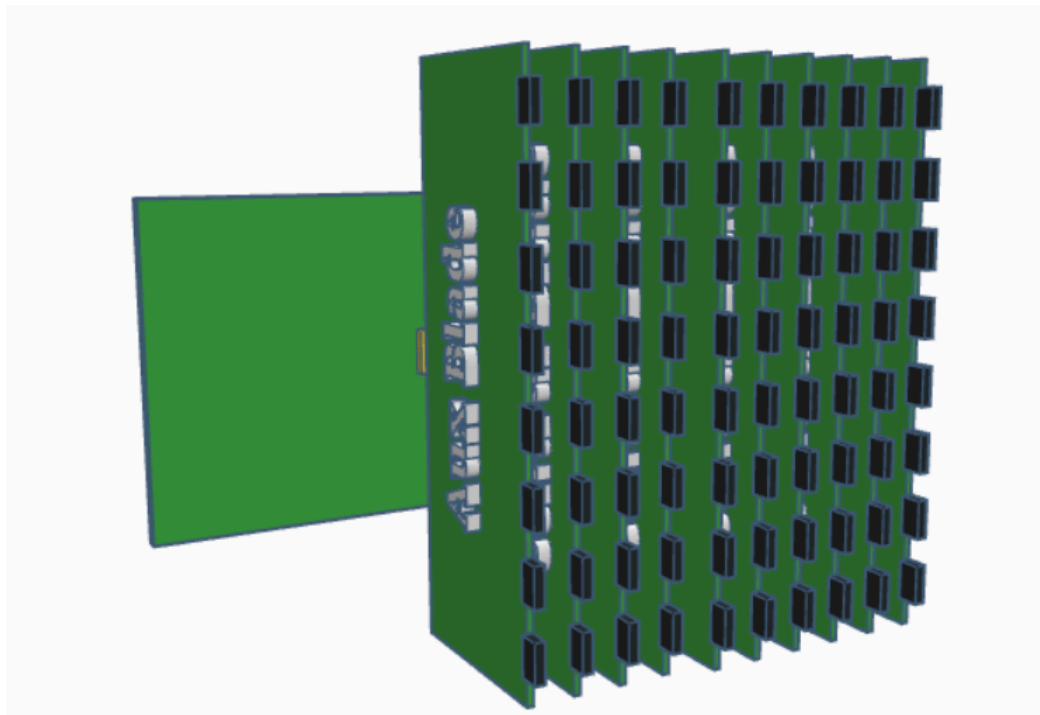
Central Blade: Early 3D Diagram, the inputs/outputs from the backplane are in yellow, and the outputs to the PMT baseboards are in black. The left header contains the pins related to the Auxiliary Blade, and the right header contains the communication between the pico and the RPi5



Aux Blade: Early Diagram, notice that the two connectors to the motherboard are different to prevent misconnection.

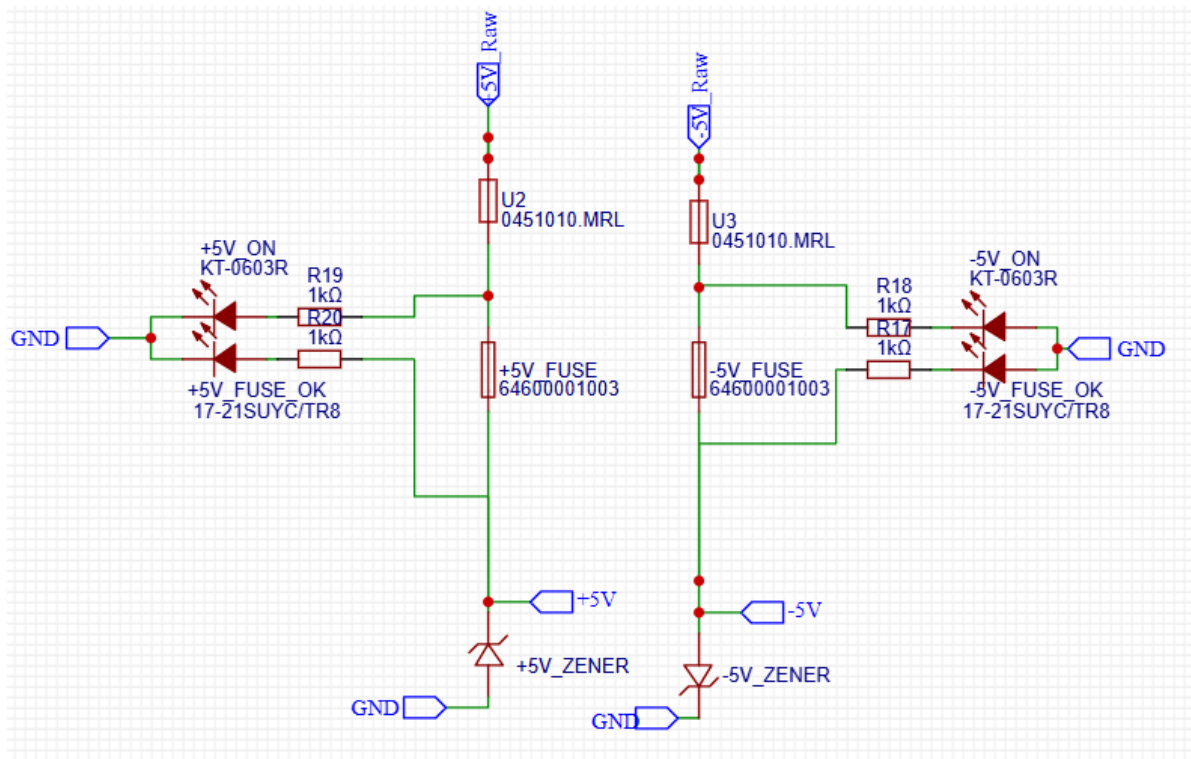


Motherboard: Early Diagram showing the RPI on the front and Blades plugged into the back.



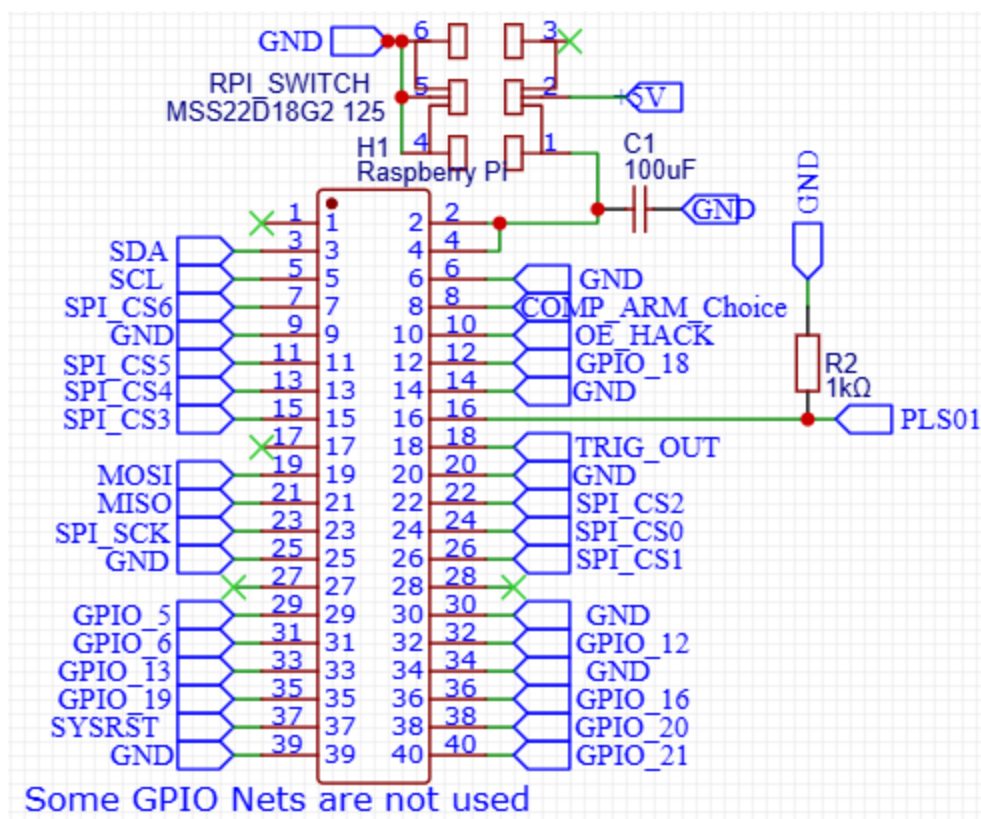
The RPi5 communicates to the Picos utilizing SPI and a custom frame-like data structure [See appendix] which contains the voltage settings and binary on/off logic for all 16 channels. The Picos then slice up with the data structure and set both the AND gates and the LED voltage using the DACs located on each baseboard.

Shifting over to the large structures of the motherboard PCB schematic, we start at the power input of the PCCS. Since the PCCS needs to power the detector's amplification electronics, fuses with LEDs and Zener diodes were added to create a clean, safe voltage for the detector. Two fuses in series are utilized: a large non-replaceable 10A fuse (U2, U3), and a small fuse holder, which we put a 3A fuse inside (+5/-5 Fuse). When the power of either voltage is on, the corresponding red LED should turn on. If the fuse is plugged in and not blown, a second yellow LED will turn on, indicating that the voltage is reaching the entire PCCS. The voltage is stepped down to 3.3V for logic and communication ICs on the motherboard and blades independently.





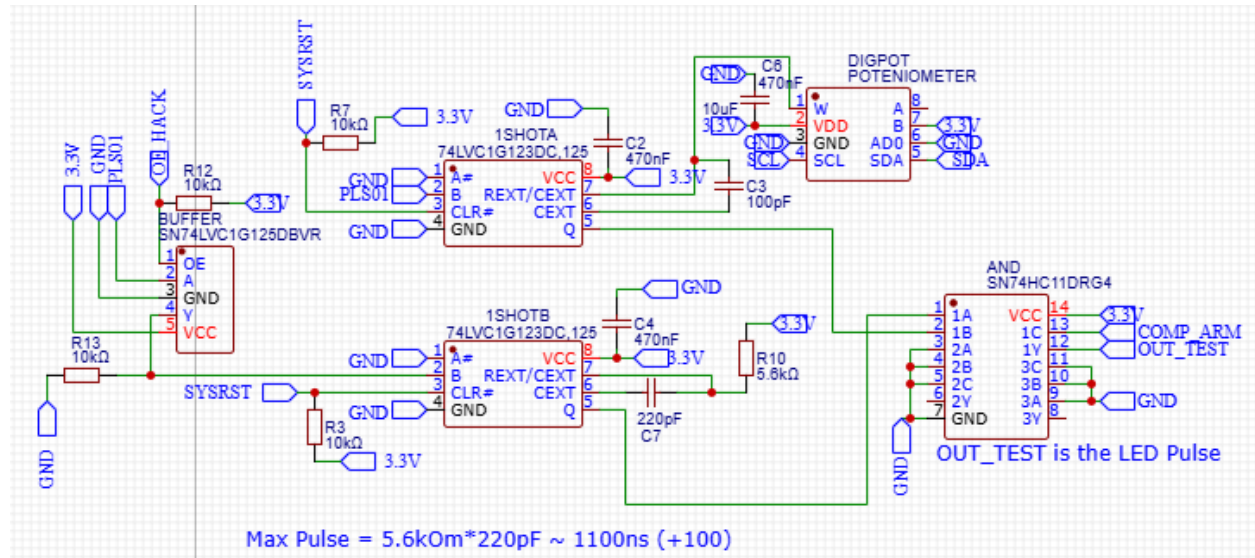
Before entering into the pulse creation mechanism, let's first look at how the interaction of the PCCS motherboard is done. An RPi5 is connected to the PCCS using a 40-pin ribbon cable.



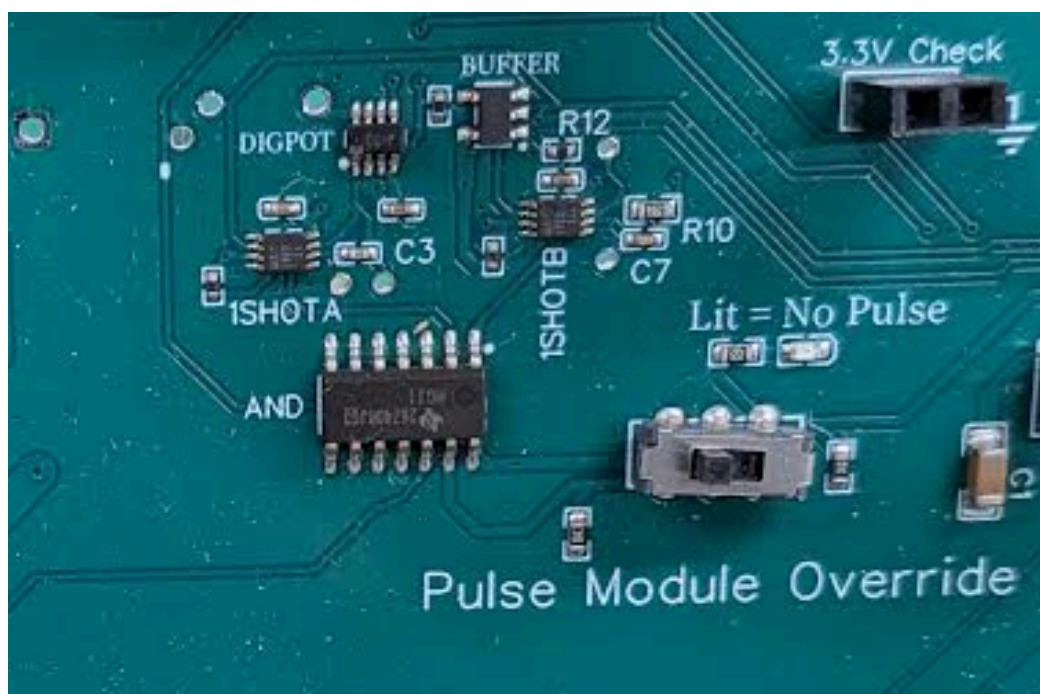
Many of the pins are used for basic PCCS functions such as pulse injection and blade communication; however, some are used for an expansion header where a PCB can be installed. The power of the RPi5 can be supplied directly from the motherboard or via its own wire. A switch near the 40-pin header controls whether the +5V is shared or not. In the installed version of the PCCS, the power is attained using Power over Ethernet (POE) hats.

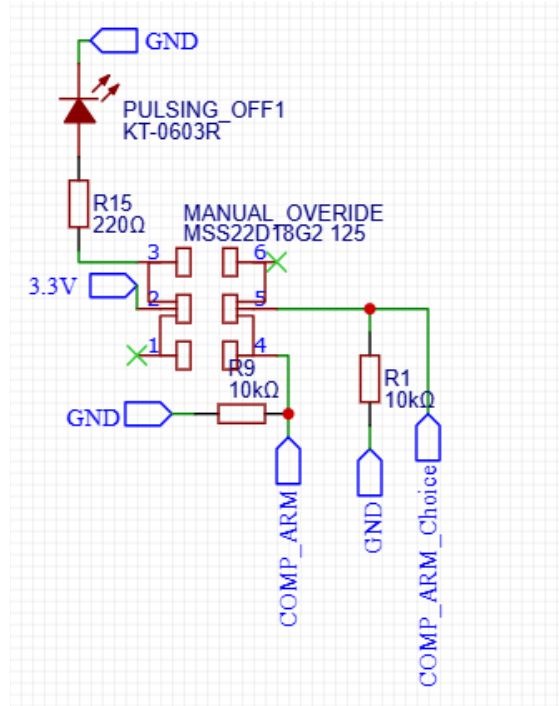
Signal injection into the MilliQan Detector utilizes a short burst of voltage on LEDs to produce photons near the PMTs. These events, named flashes, correspond to  $O(100\text{ns})$  flashes of an LED. This short pulse is created on the motherboard by the schematic below. At the core, it utilizes two different one-shots, each creating a logic pulse based on an RC time constant. By having a digital potentiometer on one of the one-shots and requiring the AND of both one-shots at the end, we are able to control the RC and thus the outgoing flash pulse length. Two different safety measures are located within the pulse creation circuit, which avoid PMT damage by preventing accidental flashing of the LEDs and limiting the maximum pulse length. To create a pulse, the one-shot's B input must go high. To do this, we utilize one of the RPi5's GPIO pins, named PLS01. This line is pulled down by a 1k resistor on the motherboard. On the bottom one-shot, PLS01 must first pass through a 3-state buffer, which is enabled by a different GPIO pin named OE\_Hack going low. This ensures that even if PLS01 is set to high by some strange

occurrence, OE\_Hack would prevent 1ShotB from creating its pulse, and thus the final AND Gate would fail. Finally, the AND gate contains an additional input called COMP\_ARM, which contains two different security features - an additional GPIO pin and a physical hardware switch that can override and turn off the PCCS flashing.



Schematic of the Pulse Generation. 1SHOTA controls the time of the pulse, while 1SHOTB controls the maximum possible time with its constant RC. The AND3 gate at the end requires all elements to have fired to allow the pulse to the blades. A real photo of the system is shown below.



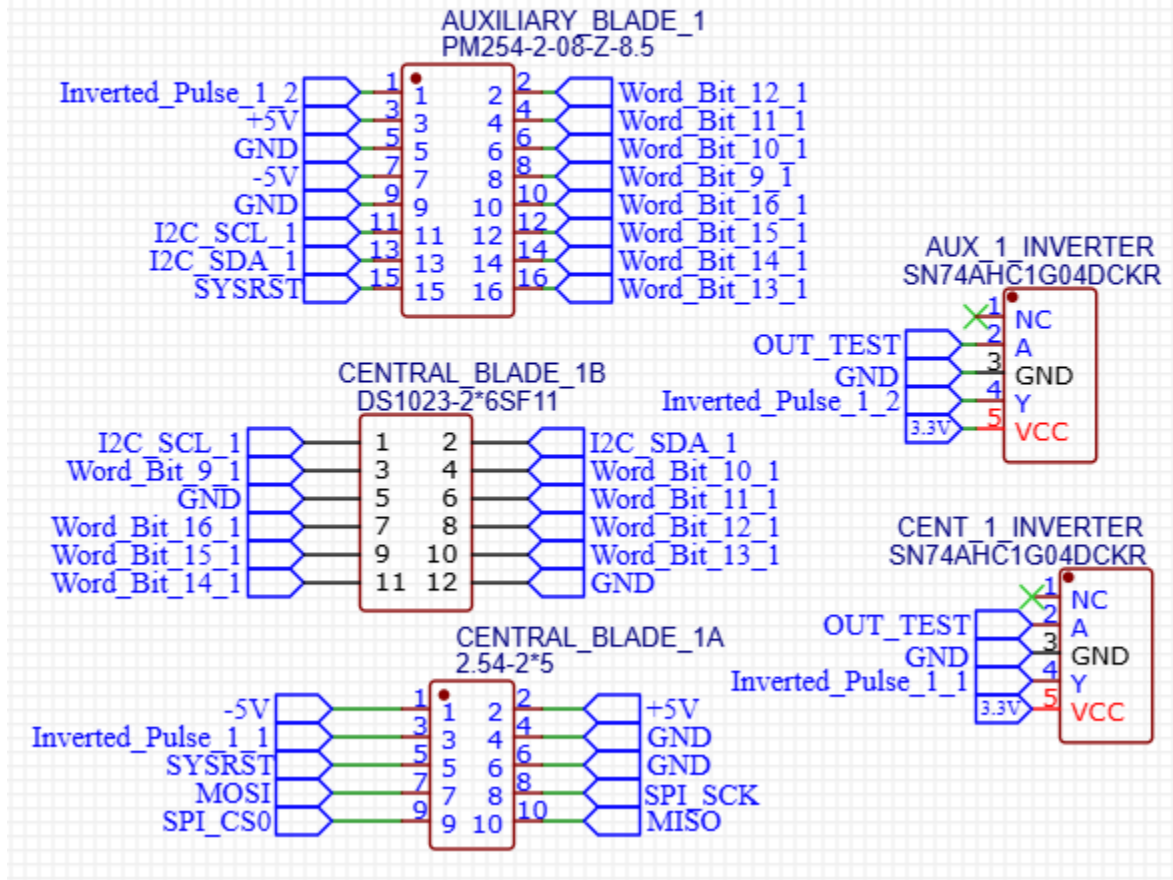


The manual override of PCCS Flashing. A bright LED turns on when the PCCS can no longer flash.

The final pulse, OUT\_TEST, is then distributed to the blade pairs and accessible through an SMA out. Additionally, a controllable trigger out is created using a buffer and an additional AND2 Gate, which is controlled by a GPIO. There is an onboard fix required to make the “trigger out” work properly. [See appendix]. Currently, this output is plugged into the MilliQan Slab Detector’s Triggerboard’s S-Particle SMA input.

Moving over to the last main component of the motherboard, the blade connectors. As shown on the 3D models, there are two connectors to the Pico blade, also known as central blade, and one connector to the auxiliary blade. Pictured are the headers for one blade pair, with included inverters. While the pulse generators come in as positive pulses on a ground line, the three-state buffer requires an inverted pulse; thus, an inverter is located next to each individual blade. On the header, Central\_Blade\_1A, we have the two voltages (1,2), the inverted pulse (3), the system reset pulse (5), and the SPI communication (7,8,9,10). The SPI communication is handled using three shared pins, MOSI, MISO, and SPI\_CLK, and one unique chip select, in this case, SPI\_CS0.

## IO Pins for one set of central + auxiliary blades 5 of these in total with sheet 2

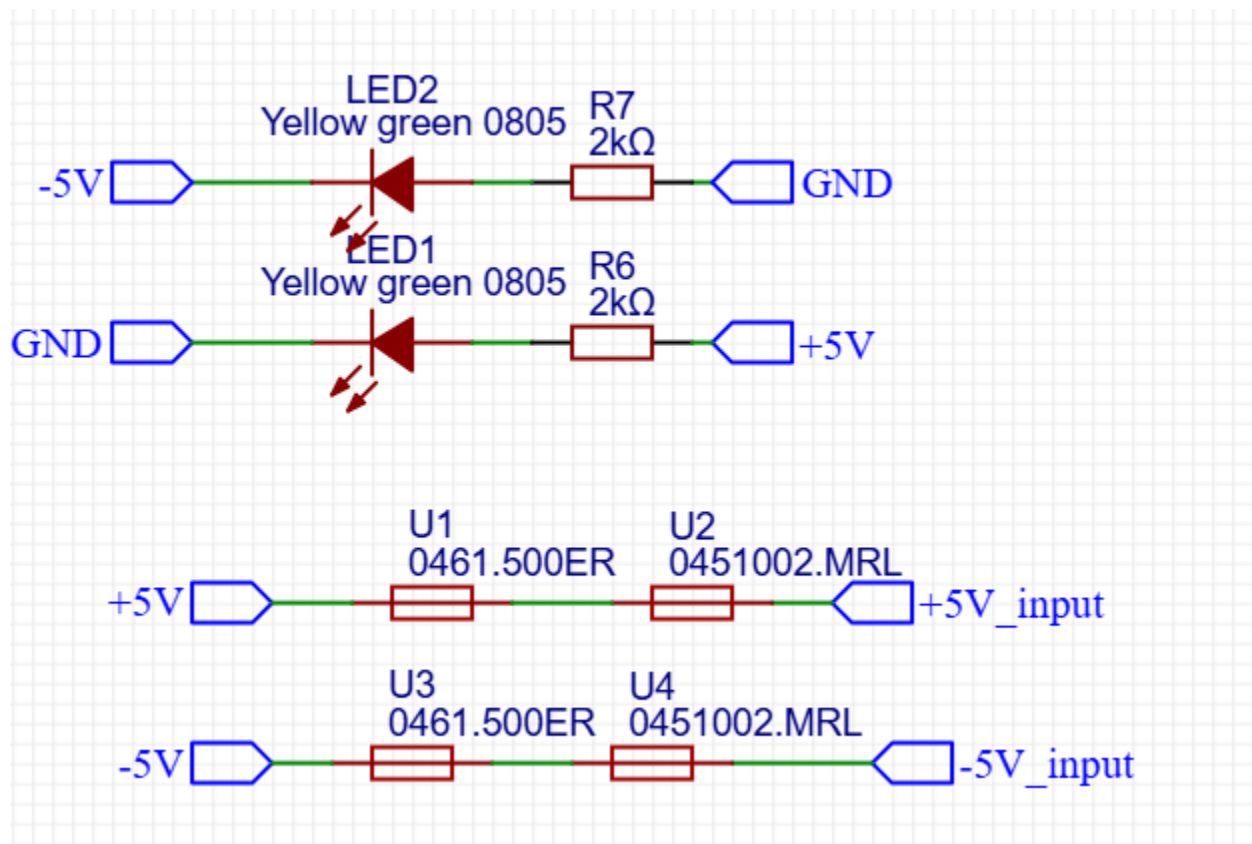


The schematic of the headers on the motherboard for each blade pair.

The second central blade connector contains all the pins needed to control the auxiliary blade. This includes the eight independent GPIO pins Word Bit 9-16, along with the two I2C communication pins SDA and SCL. On the Auxiliary Blade's header, we input all of the word bits 9-16, along with the I2C pins, the System resets, the voltage levels, and finally the inverted pulse.

Moving on to the blade pairs, the blades act as the distribution method for the core +5/-5V to the baseboards. Thus, as a first security measure, both voltage levels enter two fuses, a slow-blow 500mA fuse and a fast-blow 2A fuse. The nominal current draw from the installed blade pairs is actually around 700mA. The characteristic time of the slow-blow fuse is 100 seconds at 900mA with exponential behavior below, giving us a stable but sensitive fuse. During electrical events in

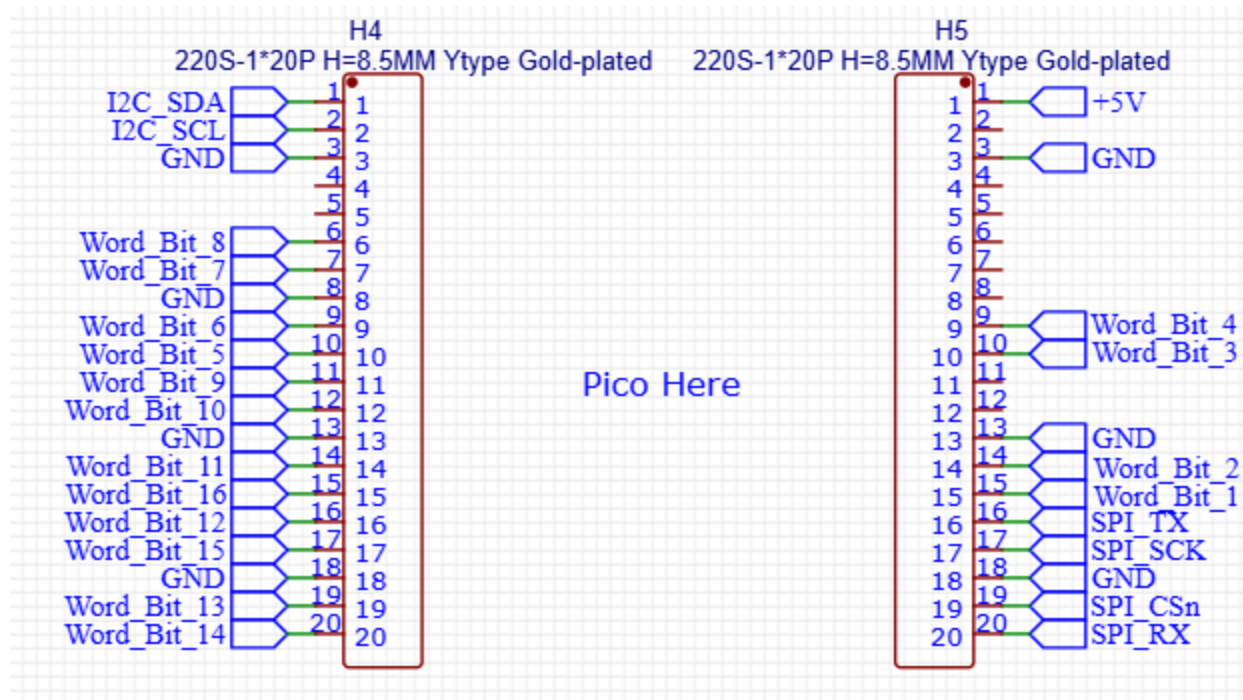
the cavern, we have seen that the slow-blow fuse can be blown, indicated by turned-off +5V and -5V LEDs located on the blades. See more in the Debugging Appendix []



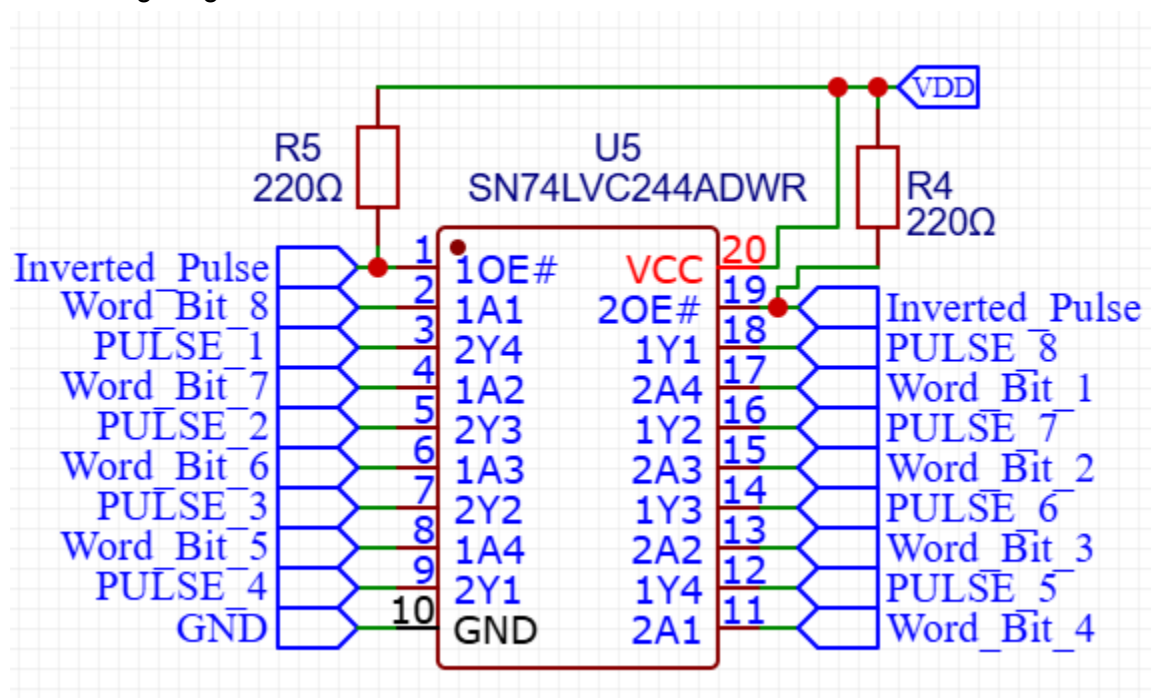
The first security functions of the blades. The two fuses are fed the “input” voltages from the pin headers and supply the board voltages as long as no error has occurred. This “stable” voltage is then fed to the LEDs.

Unique to the Pico blade is the actual Pico plug-in header. The Pico is powered directly from the +5V located on the blade through Pin 1. The Pico has dedicated pins for I2C, SPI, and the 16 on/off GPIO pins for pulse location selection.

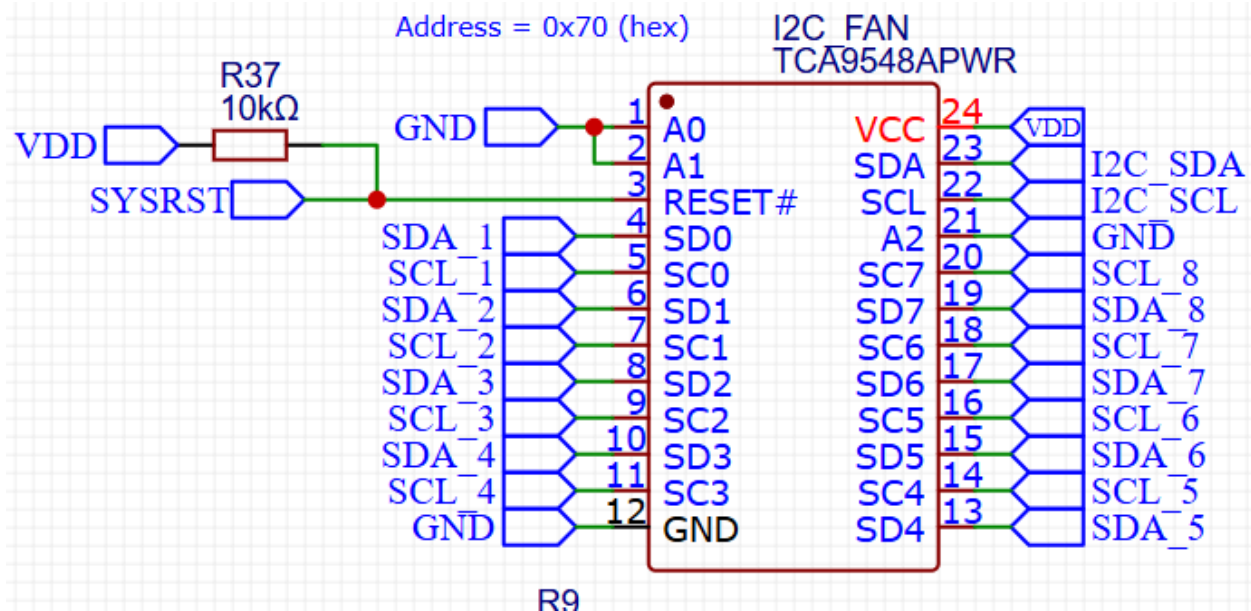




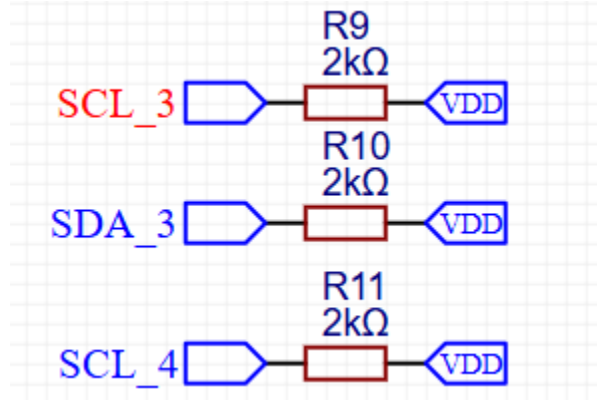
On both the Pico and Auxiliary blade, an SN74LVC244A 8-channel three-state buffer converts the blade's inverted pulse into eight independent pulses, which are then connected to each of the eight LV headers. On the auxiliary blade, word bits 9-16 are used with the outputs going to eight new LV headers. In the Pico diagrams, VDD is created by the onboard AMS1117 5V to 3.3V voltage regulator.



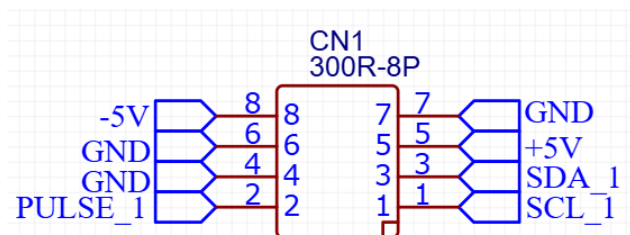
The second function of the blades is the I2C fanout; to do this, we utilize TCS9548A I2C multiplexers. Two of these multiplexers are located on each Pico's global I2C line: the Pico Blade's with I2C address 0x70, and the Auxiliary Blade's with I2C Address 0x71. When trying to communicate with one specific baseboard, let's say the one in channel 4 for this blade pair, we close the 0x71 address multiplexer and open the 4th channel on the 0x70 multiplexer. All of these functions are defined within the Pico's codebase.



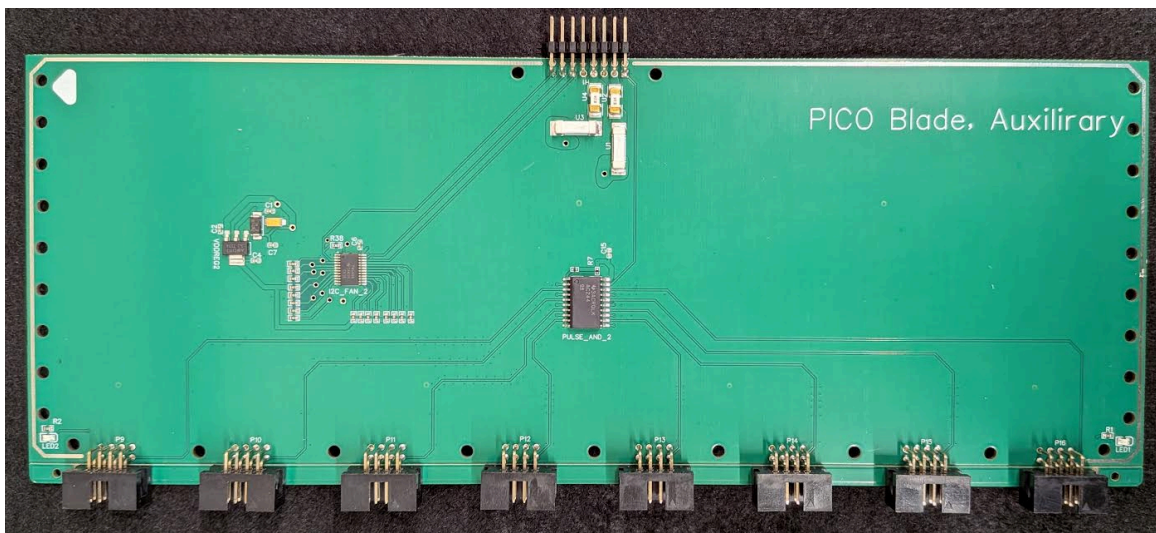
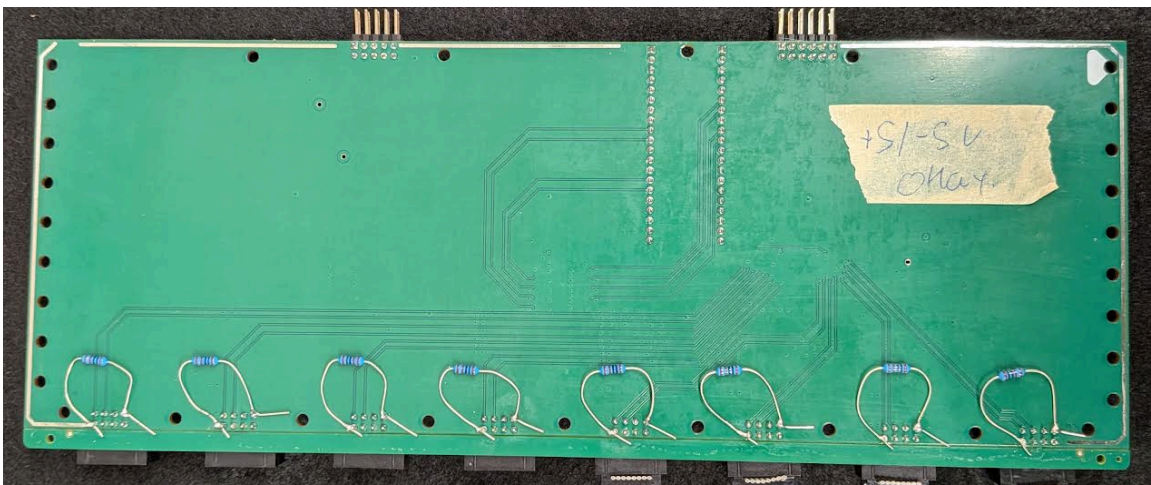
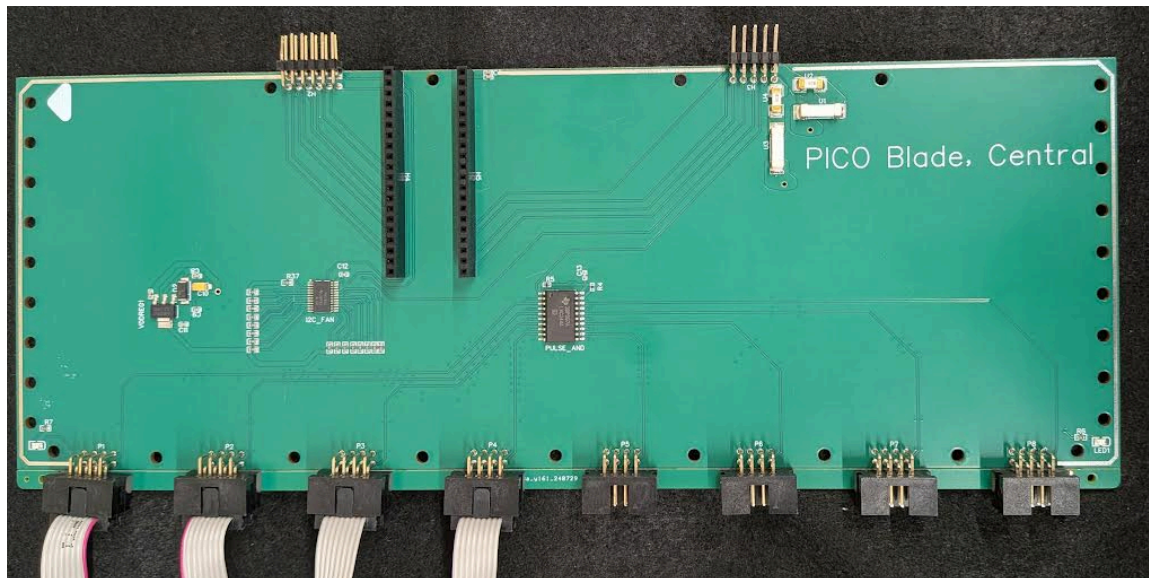
The I2C fanout chip. Note that there is a 2kOhm resistor that pulls up each of the SCL and SDA lines separately, as is needed by the I2C protocol.



Finally, we have the LV headers. Each blade contains eight LV headers. Each header contains the shared +5V, -5V, and GND, along with a unique SDA, SCL, and Pulse pin. When designing the PCB, care was taken to ensure a direct path for the pulse trace to each of the headers to ensure low noise and feedback on this important line.

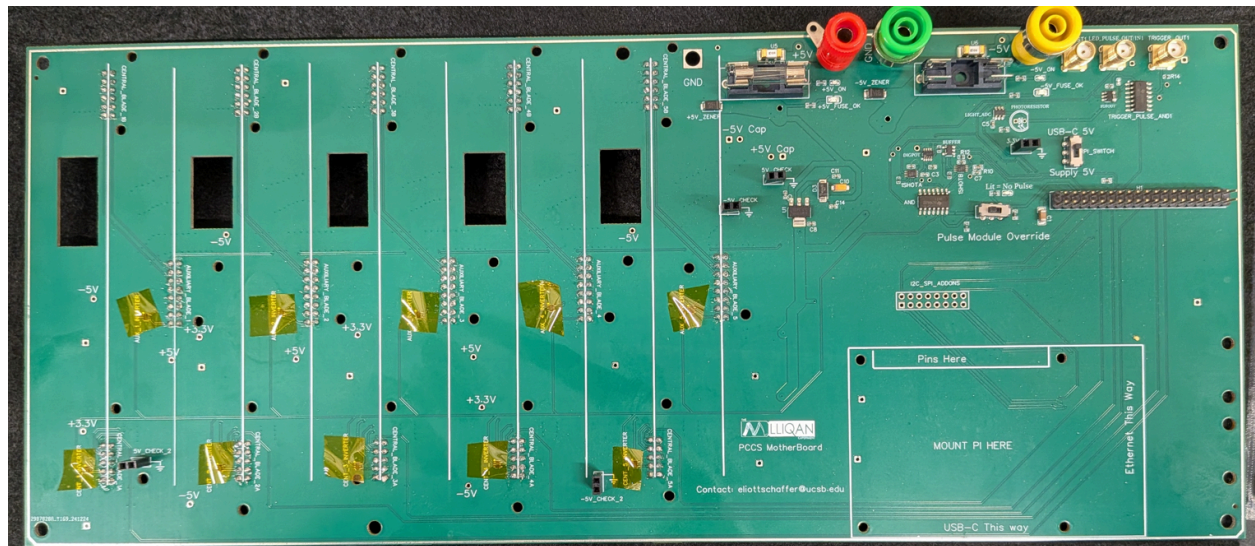


Attached are photos of the central (front and back) and auxiliary blades. Four ribbon cables are plugged into the central blade. Resistors are soldered between one of the ground pins and the pulse pin on the back of the blades to ensure a fast RC decay of the square pulse.

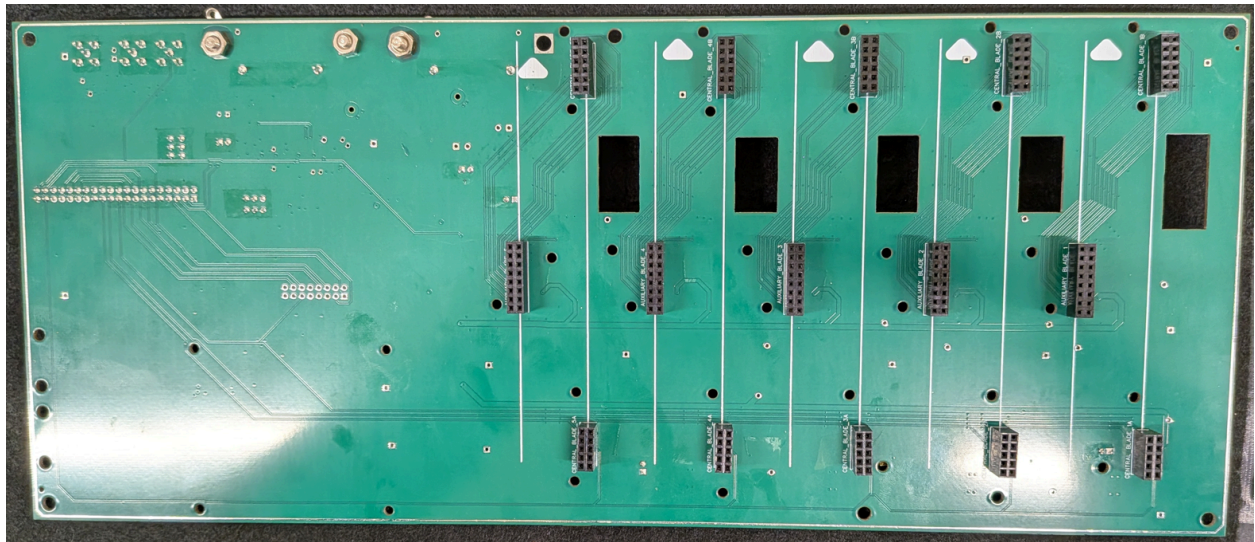




Below is a photo of the PCCS motherboard. The +5V/-5V headers are located at the top. The Raspberry Pi mount and 40 pin header are located near the pulsing mechanism. The electrical tape is placed on the blade pulse inverters before soldering the blade pin headers to prevent any damage. On the back of the motherboard, we see the 5 pairs of blade connectors. The triangle at the top corresponds to the directionality, which is matched with a triangle on each blade.



The front of the PCCS motherboard, notice the fuse holder here contains a fuse for the +5V but not the -5V. If power were to be turned on, we would see both LEDs light up for +5V, but only one of the two for the -5V.



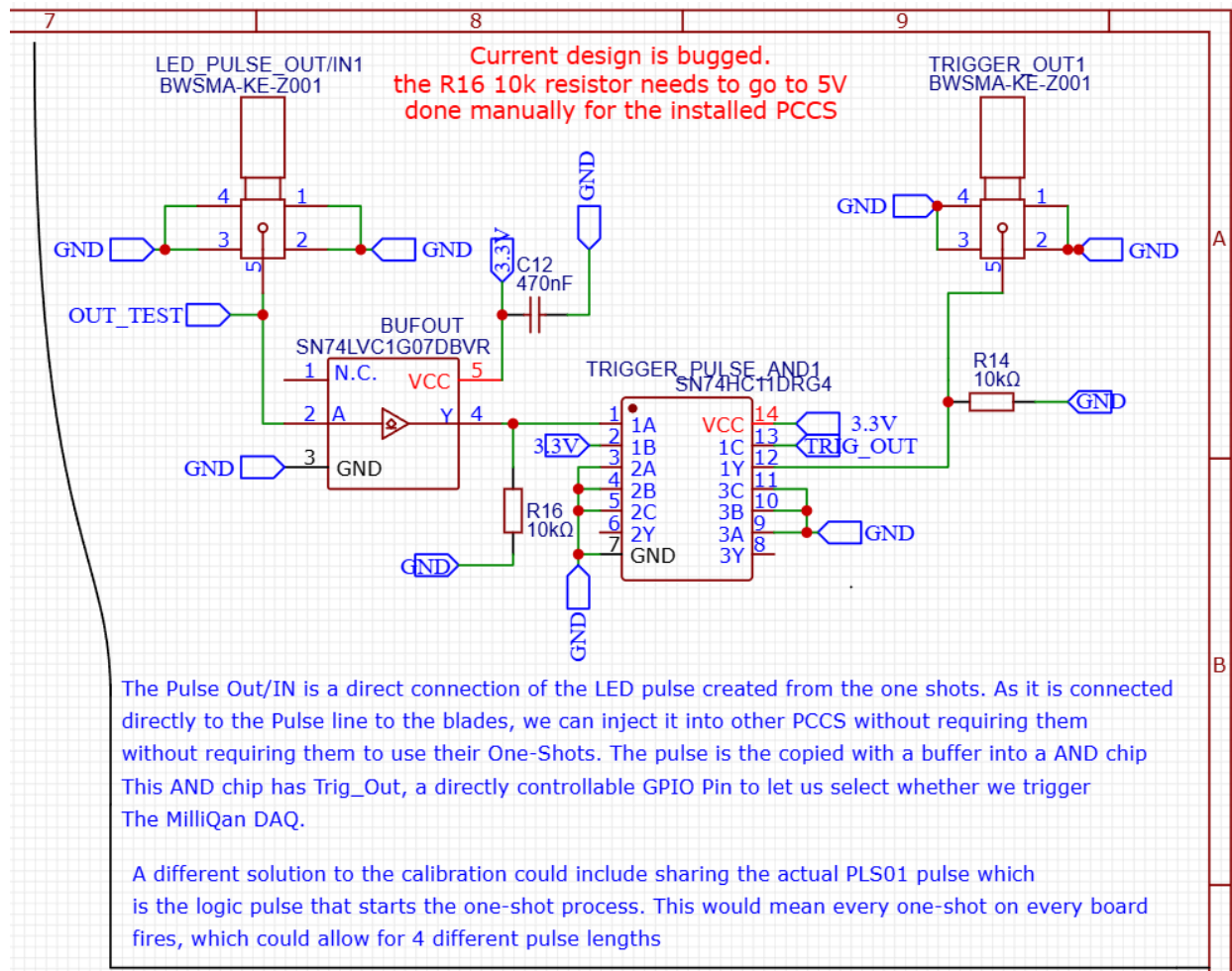
Moving on to the physical construction of the PCCS. The PCCS PCBs are housed within an 8020 frame. The frame is made of 18 in, 12 in, and 6 in 1515 pieces of aluminum extrusion. The frame is assembled such that the dimensions are 18 in x 15 in x 9 in. Within the frame, three

metal supports screw into the PCCS's mounting holes. The installed PCCS's mountings include two full-length bars and one cut-out rectangle, which allows for the 40-pin header access and pulse-out access on the motherboard. The blades are then plugged into the female headers and supported by 3D-printed supports. For more information, see [Appendix]



Multiple PCCS Modules can also be utilized together for an even greater number of channels and for systems that need physical separation, such as the Slab Detector. This utilization, dubbed Multi-PCCS, is the current prevalent use case of the PCCS. The four layers of the Slab Detector each have their own PCCS module and utilize 24 LV connectors. This means that there are four RPi5 and eight Pico microcontrollers used in the current installation. Each RPi can only communicate with the two Picos that are plugged into the motherboard, but utilizing MQTT, the RPi5 can communicate with the shared common network. When utilizing Multi-PCCS, only one of the flashing modules is used, and the pulse created is distributed to all other PCCS. The PCCS, which contains the intact pulse creation, is named the PCCS Flasher, while all other PCCS have their pulse creation module disabled by removing the final AND chip. A diagram of the pulse sharing mechanism is included along with information about the design bug regarding R16 in the selectable triggerout SMA. The other PCCS where the pulses are shared are called PCCS Sub 1-3, each denoting the layer it controls.



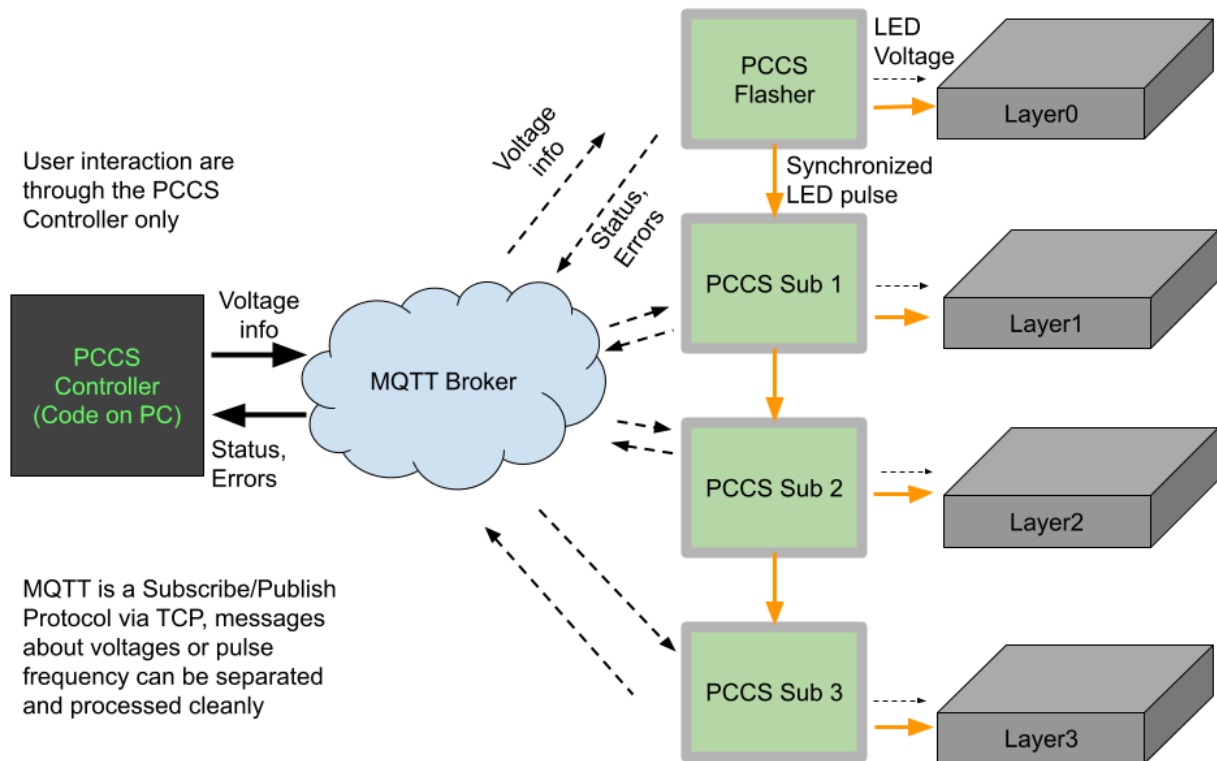


The trigger out of the PCCS can be used to trigger the detector's S-Particle trigger, one of the external triggers in the milliQan trigger board. This gives us a lot of flexibility in the range of tests that can be performed with the PCCS.

The multi-PCCS system works using two different RPi5 Python programs and the flashed compiled C Pico code. The two Python programs are PCCS\_Controller.py and receiver.py and can be thought of as boss and employee. The PCCS\_Controller.py contains all the information about the flashing configuration via an imported CSV file [See Chapter 7] and relays information regarding what each of the four PCCS modules needs to do and which channels to avoid. When running the PCCS to perform flashing, you ideally only need to run the controller script. The receiver.py script runs in a service that continuously listens for commands from the PCCS\_Controller. Once connected, the controller keeps track of the service devices connected and won't start flashing until all 4 PCCS modules have responded in a timely manner.

The system works using Message Queuing Telemetry Transport (MQTT), a subscribe/publish communication protocol which allows for custom automated data flows. When the controller wants to send data to the receiver service on each PCCS module, it does not open

communication with each of them; instead, it publishes the information to a “data” board via an MQTT broker. The MQTT broker then turns around and distributes the information to all connected devices that are subscribed to the “data” board, in this case, the receiver services. The Multi-PCCS uses many different types of channels for specific communications. Below is a diagram of the MQTT communication. Note that currently the PCCS controller code is on the RPi5 on the PCCS flasher, not the Slab PC (this doesn't affect anything).



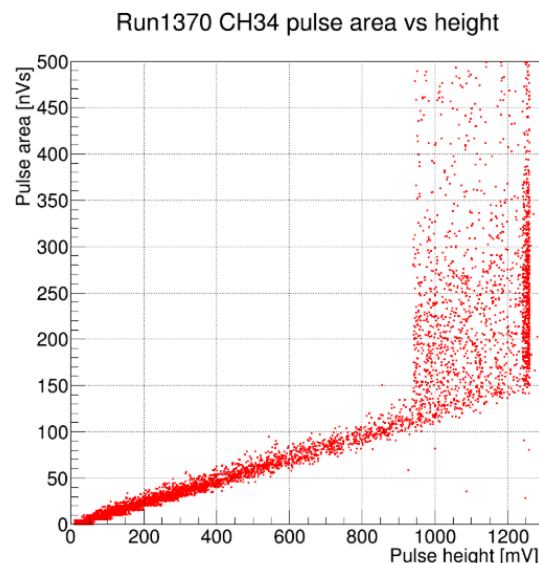
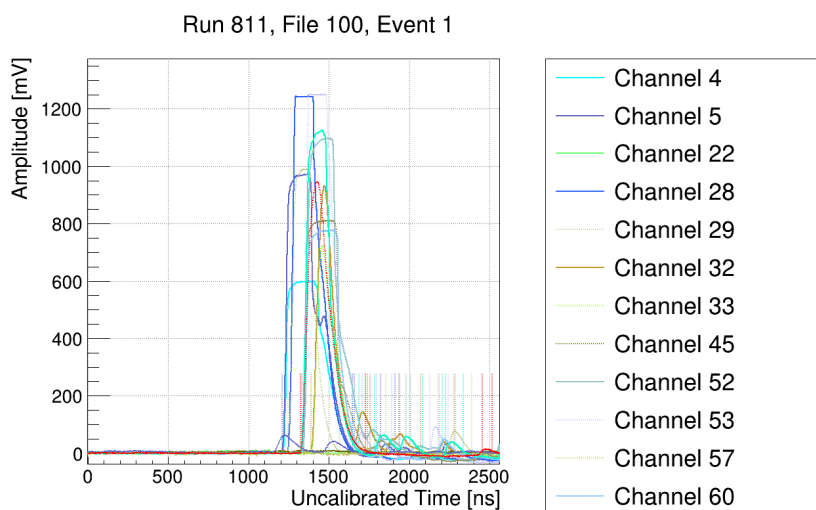
Once the receivers have received their instructions, they quickly start to relay them to the Picos, who go channel by channel, changing the DAC voltages and setting the AND gate in the correct position. Once a few seconds have passed without error, the PCCS module reports back to the controller that it is ready for the flash to be sent. At this point, the controller will ask the PCCS Flasher to create a flash and then move on to the next value in the CSV file. If a single config is requested to be repeated more than 5 times in a row, the PCCS controller will treat the first 5 flashes by going through all the motions, confirming no issues, and then issuing a command to the flasher to just start creating LED pulses at the desired frequency without re-initialising all the baseboard LEDs and AND gate. We call this mode the fast-flashing mode, and it is extensively used to collect enough statistics for analysis.

# Chapter 6

After installation in Summer 2024, the slab detector ran mostly smoothly into the winter. Some data quality issues were found with caused by the potentiometers on the summing amps, which reduced the sensitivity of a large number of channels in the detector. Intervention during the LHC 2024 Year-End Technical Stop was attempted, but access to the drainage cavern was impossible due to construction on the stairs leading up to the cavern. During the summer of 2025, the opportunity opened itself once again for Dariush and me to travel to CERN and work as a focus upgrade team on both the Slab detector. We additionally worked on slight adjustments to the Bar detector that were found during the first search. The 2025 milliQan July intervention was a quick one-month plan with a few clear goals:

1. Fix the early saturation issues causing data quality issues throughout the detector
2. Install a new IP-controlled Fan Tray for the VME Crate (allows remote power switching)
3. Installation of the PCCS

One of the recurring issues we noticed within the Slab detector data was that sometimes signals saturated at voltages below our digitizer's voltage limit. This was deemed very strange, as the amplifiers used on the baseboards and summing amps were using +3.3/-3.3V logic and the milliQan digitizers could only go up to 1.25V. We should have, in theory, never been able to see the rail saturation from the amplifiers being an issue on our digitizers. On the left graph are photos showing the visual early saturation effects. We can see a few truly saturating pulses going to 1250mV but also other pulses which seem to abruptly end at a flat top at values well below saturating. As we know, the characteristic shape of our PMTs does not include such flat-top pulses; we knew something was wrong. The right graph shows this early saturation. We expect the pulse height and pulse area scatterplot to be linear until we reach the saturation point, where the voltage is maxed out but even larger pulses can saturate the digitizer for longer, giving higher pulse area. However, we see non-linear pulse voltage and pulse areas fall within range. This is what we dubbed early saturation. This was particularly impactful as it degraded our ability to distinguish high charge mCPs and muons, which is one of the key goals of the slab detector.





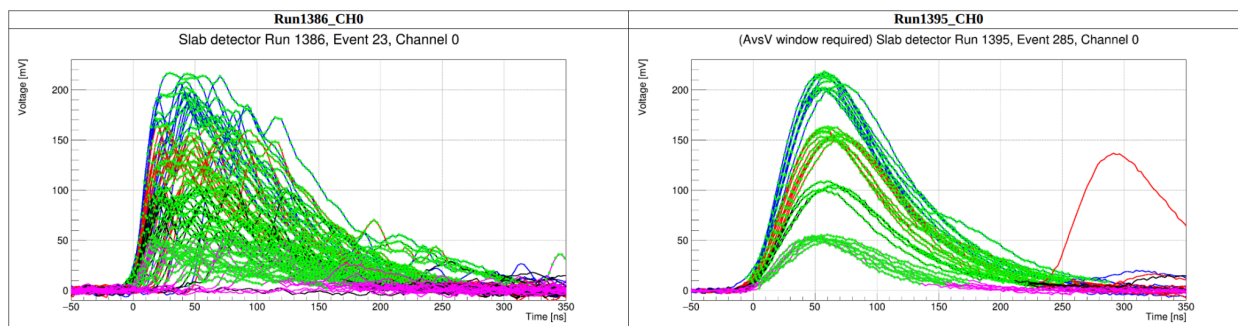
Early saturation that we observed was particularly interesting because it did not happen at a single point; it was in a range of locations. For example, the signal could saturate at 1000mV in the first pulse, but then 1200 mV in the next, with some particularly bad channels having saturation points down at 600 mV. We knew that static saturation could be caused by the DC bias from the potentiometers, but the key breakthrough was discovering that the negative rail on the summing amplifier (which should be a clean -3V stepped down from -5V) was actually oscillating from -3V to -2V at 1.7kHz.

Pictured here is the negative rail of the summing amplifier. It was determined that the early saturation was a direct cause of amplified pulses hitting the negative rails at different times during the oscillation. Since our pulses essentially had a random chance of having early saturation in the affected channel, a fix was determined to be adjusting the DC bias to be high in positive voltages to ensure that the negative pulses would saturate the digitizer before hitting the negative rail.



Since this required access to the summing amplifiers, we were able to tune potentiometers on almost all channels. The slabs on Row 2 and Row 4 of Layer 1 were unable to be accessed as they are covered by the upper rows and too close to the ground to be adjusted.

In addition to fixing early saturation, we also replaced many older generation (V1) summing amps, which started to cause anomalous ringing behavior in the pulse waveforms. Shown below is the before and after swapping a noisy V1 summing amp and the newer V3 amps.



The final data quality fix on the slab detector was fixing the extremely loud remaining channels, which we were able to apply summing amp GainB switches to reduce the output gain. This gave us a more uniform detector and prevented some channels from being overrepresented in triggers.

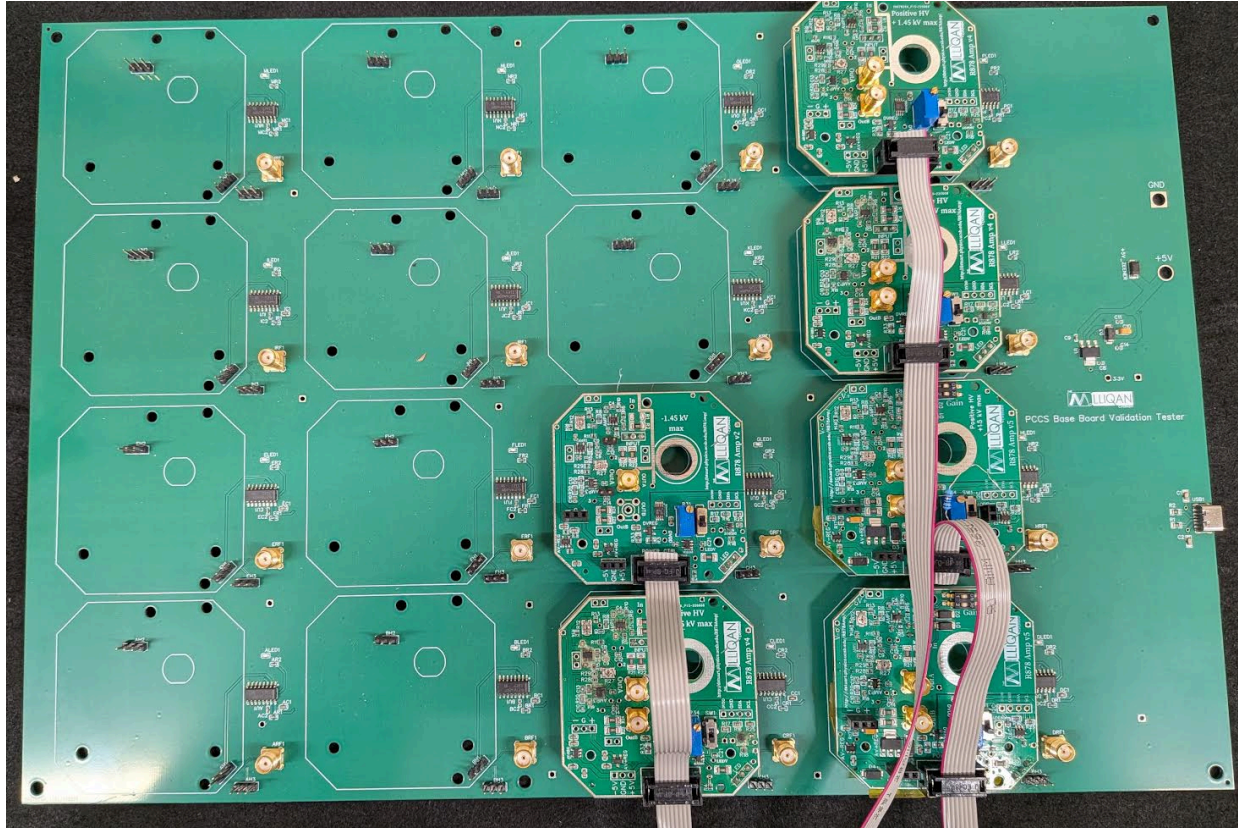
The majority of these fixes were completed within two weeks of our June 29, 2025 arrival. While Dariush took the lead on Slab detector fixes, I worked on getting remote access to the VME crate. At the time, there were two fail cases that required in-person intervention in the cavern: the triggerboard and digitizers being stuck in a “bad mode” and the drainage gallery breaker tripping. As getting people inside the cavern is not the easiest task, many valuable LHC collision hours were lost due to the inability to power cycle the VME crates. I was tasked with figuring out how to interface with our new Ethernet-controllable crate to ensure we would never lose data due to a power cycling issue. The newly installed Kontron WIENER UEP6000 allows for interaction using Simple Network Management Protocol (SNMP) and a hosted web server at the fan tray’s local IP address.

To get the server crate to connect to the cavern’s IP network, the first task was getting the crates into DHCP mode, where their IPs will be automatically assigned by the network switches. The Bar Detector Fan Tray’s IP is 128.141.91.56, and the Slab Detector Fan Tray’s IP is 128.141.91.58. Once I had successfully connected and found the IP addresses of the fan trays, I started getting SNMP to work. After reading the public [documentation](#), I found that I needed to download custom software from the manufacturer’s file [archive](#). Specifically, I needed to download a Management Information Base (MIB) file, which SNMP uses to communicate and perform different queries.

After successfully setting up both the Bar detector’s PC and the Slab detector’s, I aliased the VME functions for easy future use, as shown in the terminal window below. The Main switch controls the power to all connected VME devices without turning off the power to the fantray, allowing us to toggle VME power once and for all.

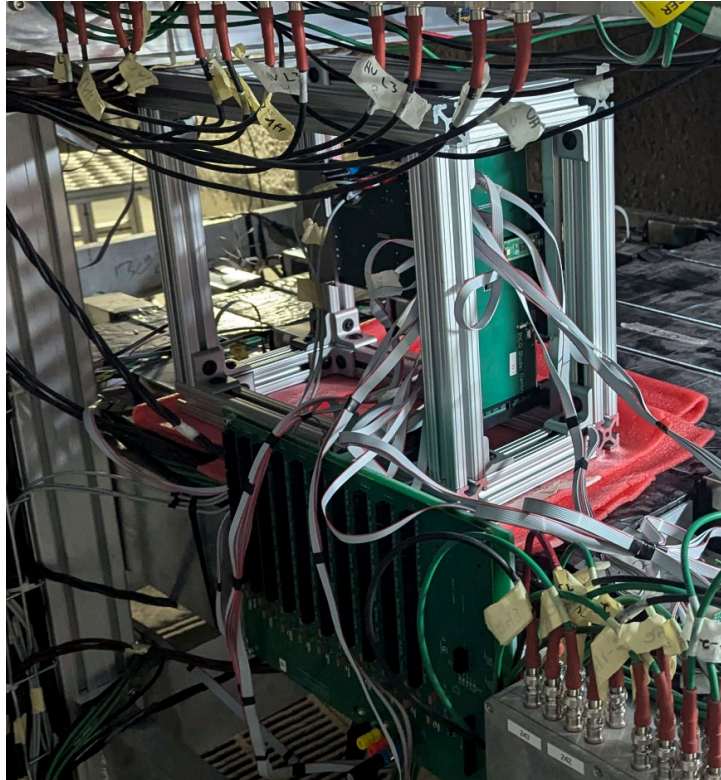
```
#####  
MILLIQAN  
Welcome to the Bar Detector!  
#####  
  
milliqan@milliqan-01:~$ VMEStatus  
WIENER-CRATE-MIB::sysMainSwitch.0 = INTEGER: on(1)  
milliqan@milliqan-01:~$ VMEOff  
WIENER-CRATE-MIB::sysMainSwitch.0 = INTEGER: off(0)  
milliqan@milliqan-01:~$ VMEOn  
WIENER-CRATE-MIB::sysMainSwitch.0 = INTEGER: on(1)
```

By July 9, my focus shifted over to the installation of the completed PCCS on the detector. By finalizing the Multi-PCCS testing using our baseboard validator board, which allows us to test the flashing and voltages of the PCCS. More information on the validation tester can be found in [Appendix].



Once I felt confident that the PCCS was able to control the detector in a safe and controlled manner, we started the replacement of the LV-Board with the PCCS. We started by reconfirming PCCS operation on the PMT flashed in 2024 and then moved slowly, adding more LV ribbon cables until we filled out a whole blade. Once we validated that PCCS and LV-Dist gave identical data, we moved to a full-scale replacement on Layer 2. Before moving the ribbon cables from the old board to the PCCS, I ensured to record the existing positions, as many LV-Dist headers give anomalous voltages and cause failures in the detector. When replacing the ribbon cables onto the PCCS, each of the three blades had control of one column (4) slabs. Having a very well-defined plugging order, it allows us to more easily track issues in the detector and isolate potential voltage shorts/surges. Previously, the ribbon cables were scattered around the 80 LV headers, making it difficult to pinpoint an LV-Dist issue. When a PCCS blade fuse blows, it is immediately evident that one column of the detector is dead. Each blade has a unique layer and column, with the row and front/back side of the slab changing. In the picture below, we see 213-12 followed by 213-34, which are the front (34) and back (12) sides of the slab. We then move on to 223-X, 233-X, and 243-X. This blade controls all PMT amps on layer 2, column 3.





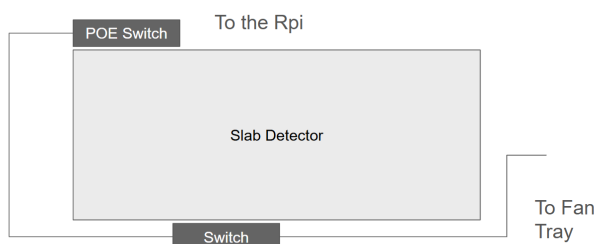
By July 16, PCCS modules were installed on Layer 2 and Layer 3. Full-scale flashing test waited until the Multi-PCCS was constructed. We were, however, able to start scanning the I2C-connected DAC. In doing so, we repeatedly found baseboards that had I2C communication failures. Since these channels would cause the Pico microcontrollers to crash, the Picos needed to avoid them. To do this, I added a hard-coded setup file on the RPi5, which is sent to the Pico prior to any I2C communication attempt. The Picos can then avoid opening the I2C multiplexer to the bad channels regardless of whether we try to change the voltage of the LED. Attached is one such Pico printout of the working and broken channels. We see that in well-behaving channels there are two different I2C addresses, as we had previously connected 50% of the solder-bridge pins during construction.

The Layer 4 PCCS module was installed by July 23, leaving the Layer 1 PCCS module (which

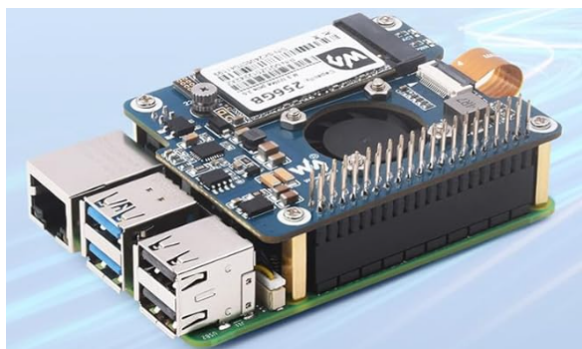
```
Summary of detected devices:
Channel 0: 0x4D 0x4C
Channel 1: 0x4D 0x4C
Channel 2: 0x4D 0x4C
Channel 3: 0x4D 0x4C
Channel 4: No devices
Channel 5: No devices
Channel 6: No devices
Channel 7: 0x4D 0x4C
Channel 8: No devices
Channel 9: 0x4D 0x4C
Channel 10: 0x4D 0x4C
Channel 11: 0x4D 0x4C
Channel 12: 0x4D 0x4C
Channel 13: 0x4D 0x4C
Channel 14: 0x4D 0x4C
Channel 15: No devices
```

contained the intact flashing module) as the last module to install. As we increased the number of PCCS, it became clear that we could no longer power the PCCS' RPi using our own USB-C chargers. Since we knew that we needed to hook up the RPi to the cavern network, we looked into using Power over Ethernet (PoE), which allows Ethernet cables to power the Raspberry Pi in conjunction with a specialized RPi PoE hat. These hats contain a transformer that steps the Ethernet's 48V to the 5V used by the Raspberry Pi while retaining access to all the pins that the PCCS motherboard needs. We used a POE M.2 HAT+ for RPi 5 (pictured).

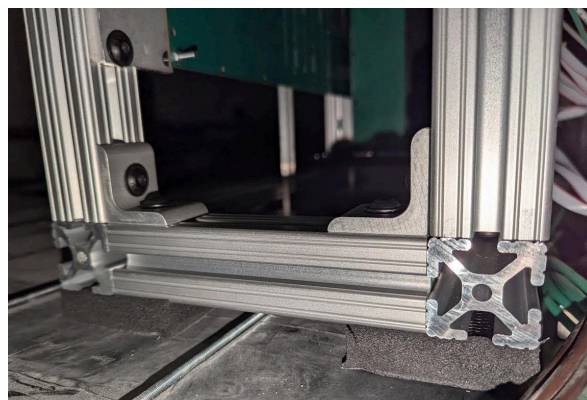
Since PoE needs specialized switches, two new switches were installed on the slab detector. One for general-purpose, more access ports, and the other being a Netgear GS305PV3 PoE-enabled switch. After that, we were ready to install the last PCCS module and start all channel flashing.



completed on July 30th, along with scans of the available I2C modules. After connecting the RPi to the PoE switch, permanent access to their IP addresses became possible, as previous tests utilized phone hotspots to SSH into the RPis. When installing the final PCCS, we retrofitted the flammable pink bubble wrap the PCCS were sitting on with black thermal isolating foam feet.



Installation of the last PCCS module was

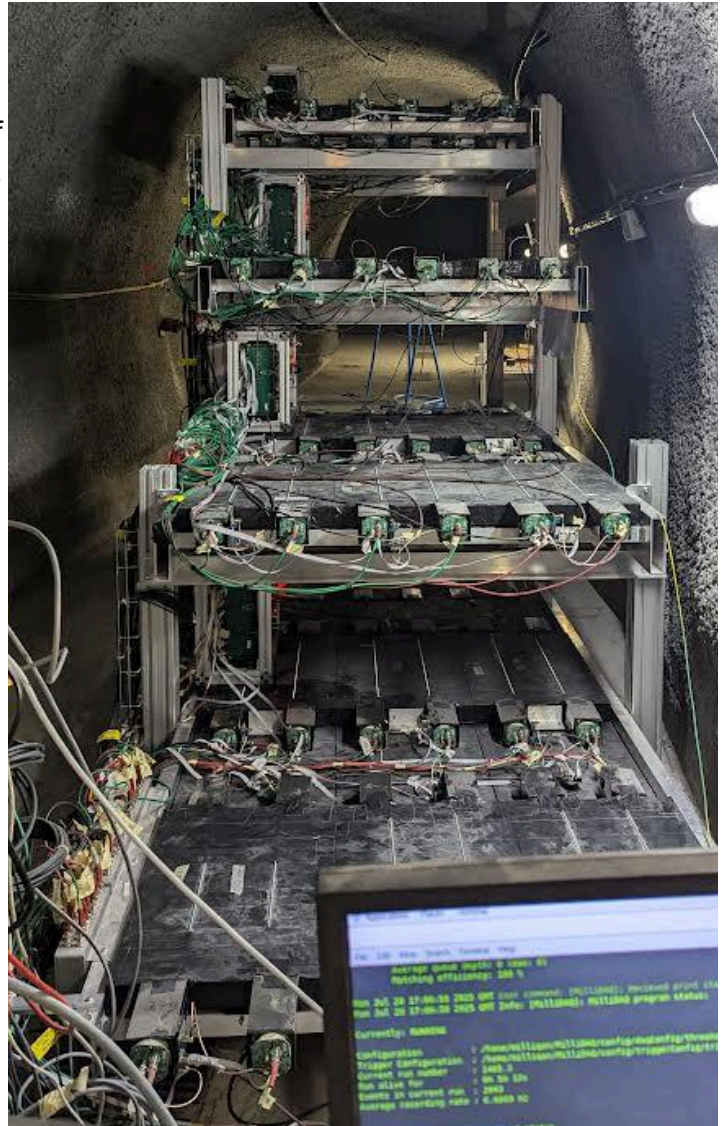


On the final days of our stay at CERN, we worked through debugging the PCCS and getting ready for remote operation of the system while at UCSB. We performed the first global flashing tests, which gave us a map of all the working LEDs. Unfortunately, the LED availability inside the detector is not 100%. This is due to a few different reasons, the first being that many PMTs simply do not have LEDs installed. Since the slabs were created, there had been no plan or intention to use the LED flashing. A systematic testing issue with the installed LEDs caused some channels to be plugged in backwards, in which they became inaccessible after completion of the detector. Some channels are inaccessible due to I2C failure considerations. Finally, the dynamics of the LED location naturally lead to a high chance of photons failing to reach the PMT face. The LED photons need to bounce between the black tape and PMT-holding plastic before reaching the PMT face. In the end, only 18 of the 48 slabs had a single working LED. Thankfully, two four-in-a-row slab paths were available for signal trigger emulation. The results in Chapter 8 are due to these two available paths (back left X43 and back right X41).



Dariusz and I left CERN on August 1st, 2025. We had successfully completed all of our tasks on the slab detector, as well as performed a realignment and introduction of a larger front-end panel on the Bar detector. A photo of the PCCS upgraded Slab detector as we left it in August is shown on the right.

| Known flashed Slabs |        |        |
|---------------------|--------|--------|
|                     |        |        |
| Layer 0             |        |        |
| 22, 23              | 14, 15 | 6, 7   |
| 20, 21              | 12, 13 | 4, 5   |
| 18, 19              | 10, 11 | 2, 3   |
| 16, 17              | 8, 9   | 0, 1   |
|                     |        |        |
| Layer 1             |        |        |
| 46, 47              | 38, 39 | 30, 31 |
| 44, 45              | 36, 37 | 28, 29 |
| 42, 43              | 34, 35 | 26, 27 |
| 40, 41              | 32, 33 | 24, 25 |
|                     |        |        |
| Layer 2             |        |        |
| 70, 71              | 62, 63 | 54, 55 |
| 68, 69              | 60, 61 | 52, 53 |
| 66, 67              | 58, 59 | 50, 51 |
| 64, 65              | 56, 57 | 48, 49 |
|                     |        |        |
| Layer 3             |        |        |
| 94, 95              | 86, 87 | 78, 79 |
| 92, 93              | 84, 85 | 76, 77 |
| 90, 91              | 82, 83 | 74, 75 |
| 88, 89              | 80, 81 | 72, 73 |



# Chapter 7

## Guide to PCCS Utilization:

To start a PCCS flashing run, you must SSH into the Flasher's RPi5, PCCS@128.141.91.10, while connected to the drainage cavern's network using CERN's LXPlus. The IP addresses for all the PCCS modules are:

Layer0 (Flasher) 128.141.91.10

Layer1 128.141.91.11

Layer2 128.141.91.12

Layer3 128.141.91.13

```
(base) milliqa@milliqaipc-02:~$ ssh pi@128.141.91.10
pi@128.141.91.10's password:
Linux raspberrypi 6.6.20+rpt-rpi-2712 #1 SMP PREEMPT Debian 1:6.6.20-1+rpt1 (2024-03-07) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Apr 26 09:32:29 2026 from 128.141.91.73
pi@raspberrypi:~/MilliQan-Experiment-PCCS/Source $
```

First, you must restart the receiver service on the flasher module, as there is a bug that causes it to lock up. Each PCCS module runs a receiver service, but only the flasher needs to be restarted at the start of each run. The receiver service responds to the PCCS Controllers' MQTT requests and commands.

```
sudo systemctl stop receiver.service
sudo systemctl start receiver.service
```

Then, utilizing a flashing config CSV (see next section), run "python PCCS\_Control.py /path/to/your/.csv".

```
python PCCS_Control.py ~/PCCS_In_Situ_New_Triggerbit_Validation.csv
```

The console will show the sender scripts' attempts to communicate with all four modules' receiver services, and once all have connected, it will start to send the pulse information to each Sub PCCS. For flashing runs where we send thousands of flashes with the same config, once the sender has confirmed five good flashing attempts, it will enter the quick flashing mode, where instead of sending the information, it will just start creating the flashing pulses at a desired frequency chosen within the CSV config. The flashing can be terminated at any time by stopping or restarting the receiver service on the PCCS Flasher (using the systemctl commands above).

If planning on triggering on S-Particle, you must go to the Slab detector's trigger config and turn on the bit for S-Particle in the config file by uncommenting the nominal trigger config or making a custom one.

```
#triggerBoard.trigger = 0b0001000001011111
triggerBoard.trigger = 0b0000100000000000 #S Particle
#triggerBoard.trigger = 0b0000000100000000 #Four in a row
#triggerBoard.trigger = 0b0001000101010011 #Nominal running menu
#triggerBoard.trigger = 0b0000000100000011
#triggerBoard.trigger = 0b0001000000000000 #zero bias only
```

## PCCS CSV Format:

The PCCS CSV format contains the information needed to program every single channel and flashing characteristic in a run. It is split up into two sections: the first 5 columns denote global settings needed for flashing characteristics and triggering. It is then followed by 192 values, one for each of the four PMTs on all 48 slabs, where you can enter a value 0-4000 for the DAC count voltage going to the LED.

|   | A                   | B               | C       | D           | E      | F       | G       | H       | I       | C |
|---|---------------------|-----------------|---------|-------------|--------|---------|---------|---------|---------|---|
| 1 | Detector (bar/slab) | Flashing Repeat | Rate Hz | Trigger 0/1 | Length | Chan0-R | Chan0-L | Chan1-R | Chan1-L |   |
| 2 | slab                | 5000            | 5       | 1           | 100    | 0       | 0       | 0       | 0       |   |
| 3 | slab                | 5000            | 5       | 1           | 100    | 0       | 0       | 0       | 0       |   |
| 4 | slab                | 5000            | 5       | 1           | 100    | 0       | 0       | 0       | 0       |   |
| 5 | slab                | 5000            | 5       | 1           | 100    | 0       | 0       | 0       | 0       |   |

A description of the global settings is included:

**Detector:** Selects the mode in which the PCCS is operating, either single PCCS or Multi-PCCS. For all applications on the slab detector, it must be set to “slab” as shown above.

**Flashing Repeat:** Denotes the number of times the flashing configuration requested will be injected into the detector. If you want 5 files of this particular injected data, then select 5000 flashes.

**Rate:** The rate at which you want to flash the number of repeats. Caution: There is no hard cap currently; going too quickly could damage the detector.

**Trigger:** Whether you want to trigger the MilliDAQ’s S-Particle external trigger. “0” means no triggering; “1” means triggering. This will force the readout of the PCCS-injected signal independent of the actual triggerboard logic.

**Length:** Controls the length of the LED flash in nanoseconds. The minimum is 100 ns, and the maximum is 1100 ns.

Currently, when in Multi-PCCS Mode, only one row will be performed successfully before the multi-pccs locks up. One needs to edit the CSV and delete a row before continuing on to the next configuration. This is a bug with the Multi-PCCS mode, as the PCCS has previously been used with long and varied CSV sweep configurations. See [Appendix] for more info.

```
vi ~/PCCS_In_Situ_New_Triggerbit_Validation.csv
```

To import the CSV into the MilliQan cavern utilizing SCP from a local device to the slab detector PC, then from the slab PC to the flashing PCCS, or to modify any of the existing configuration files.

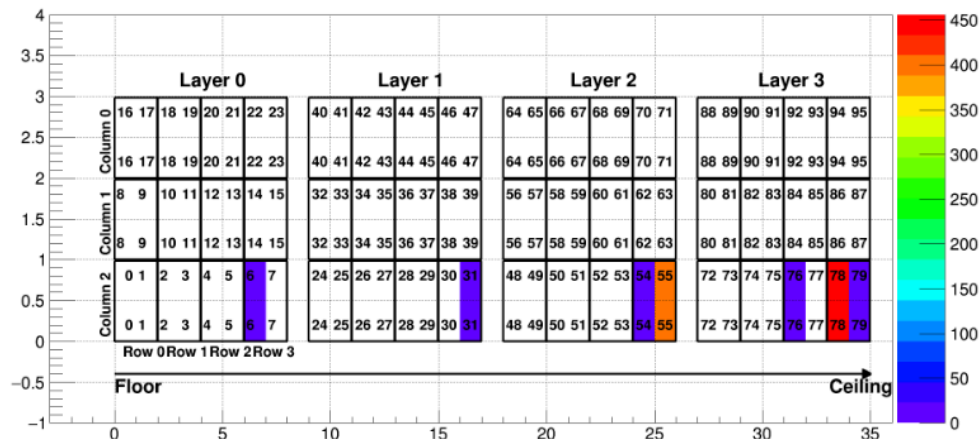
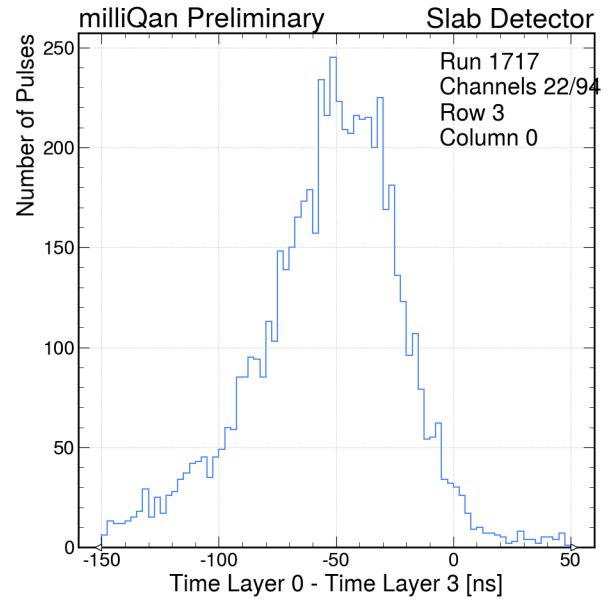
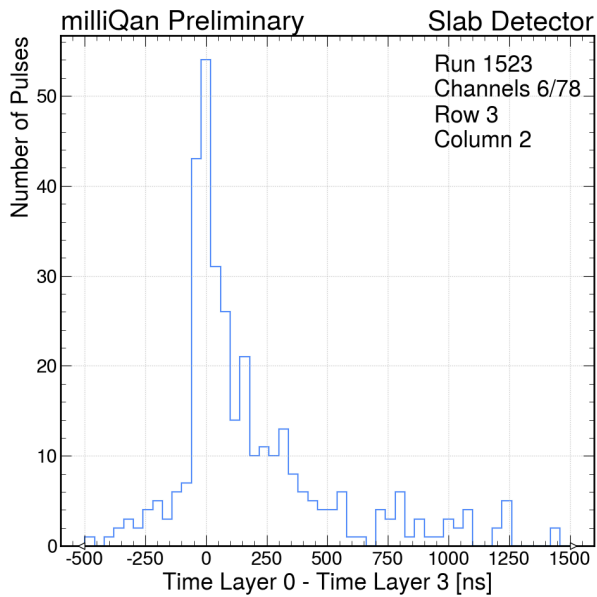


# Chapter 8

Flashing the Slab detector started in Fall 2025, during the short time between LHC beam dumps and ramp-up. This limited the scale of flashing that we could perform, but gave us plenty of time to start studying the LEDs in the detector and their response. Attached is a waveform display of sending a 1000 ns-long pulse to the LED of Chan 31. We see many different pulses, which complicate the studies we wanted to conduct.

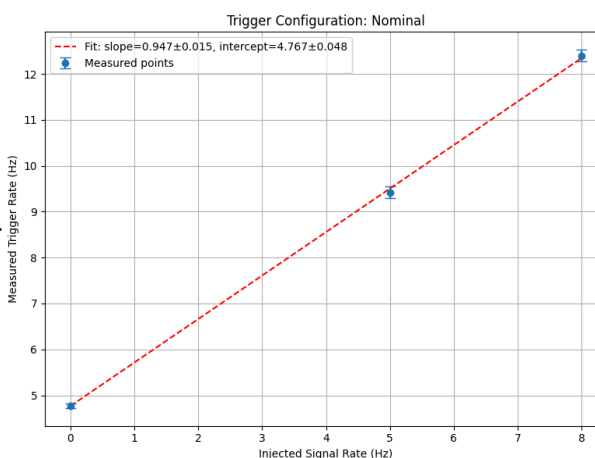


One of the key observations of the first tests was that shorter LED pulses generated from the one-shot were more useful, as they had the tightest timing distribution and could more easily be scaled up to longer LED flashing. Below are two different PCCS pulse timing distributions, one using a 1000ns pulse (left) and the other using 100ns pulses (right). These distributions are looking at different channel pairs, but the effect remains real.



The timing distributions on the PCCS-injected signals are, on average, much larger than beam muons and our signal mCPs, which move near the speed of light through our detector. Since the coincidence window of the trigger board, 80ns, is set based on real conditions, many PCCS-injected signals failed to trigger the detector. By increasing the coincidence window to 400ns (larger than the pulse timing distribution), we were able to trigger on the PCCS-injected signals reliably and without using S-Particle. That is, the detector itself was able to detect the PCCS signals without any prompting.

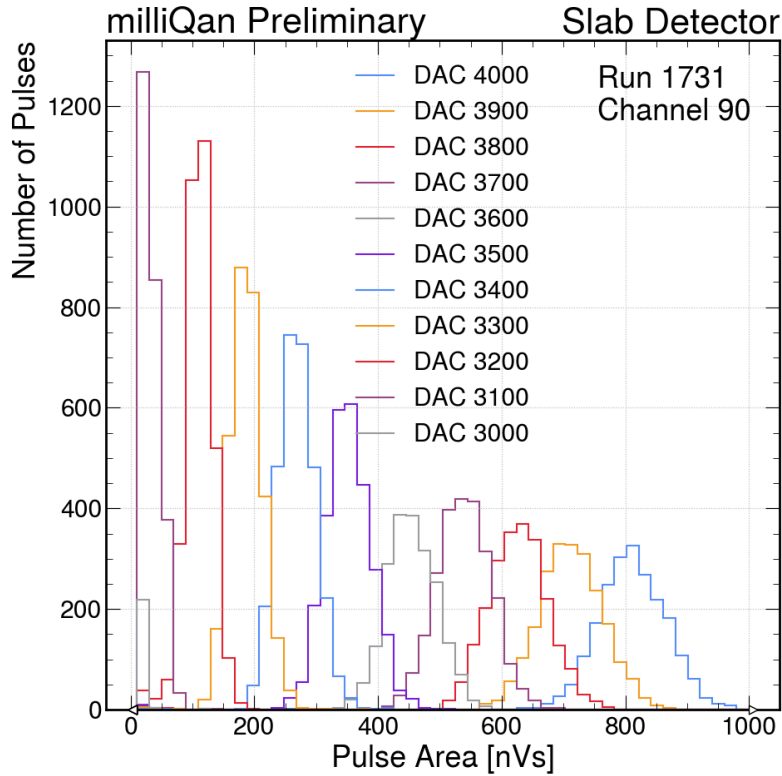
This can be seen in a PCCS-injected event display and by looking at the increase in trigger rate as the flashing rate increased, which gives us around ~95% triggering efficiency on the “saturating” injected signals. This test was performed with the nominal Slab detector trigger suite.



These first quick tests confirmed that the Slab detector was performing to our standards and didn't need emergency intervention during fall. The large PCCS tests were performed during the winter LHC shutdown, where the time pressure of the beam ramping was no longer a concern. These first tests utilized large (4000 DAC counts) uniform LED voltage. Since each LED is connected to a unique PMT, to create a uniform signal in the detector, the translation of DAC voltage to the number of photoelectrons received in each PMT was essential. To characterize the LEDs, we performed a DAC count sweep in all connected LEDs. By having the PCCS cause the detector to trigger on S-Particles, we were able to look at LED-injected pulses without having to worry about satisfying any of the data-forward triggers.

For Slab detector Run 1731, Dariush and I prepared a CSV file sweeping all channels from the maximum DAC voltage, 4000, down to 3000 counts in steps of ten. At each DAC value, 5 files (5000) worth of signal injection flashes were collected at 5 Hz and then plotted. A photo of the CSV file, PCCS\_In\_Situ\_Pulse\_Voltage\_sweep\_4000\_to\_3000.csv, is shown below. Access to this file and all CSV test files can be found on the milliQan Slab PC.

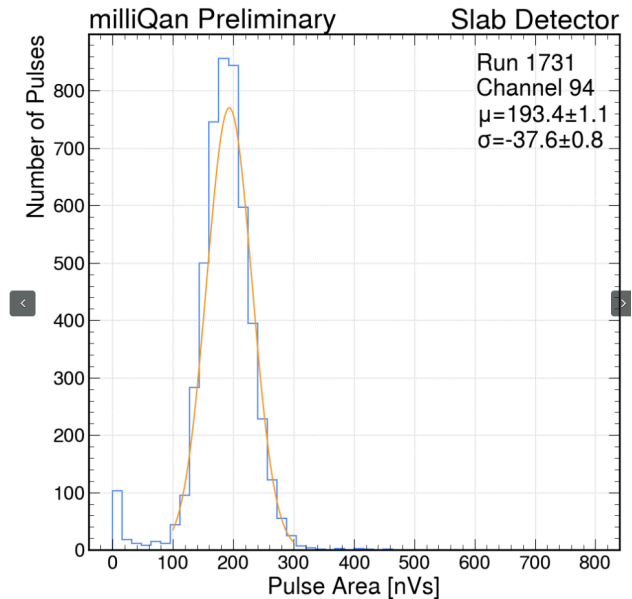
| Detector ( | Flashing R | Rate Hz | Trigger 0/1 | Length | Chan0-R | Chan0-L | Chan1-R | Chan1-L |
|------------|------------|---------|-------------|--------|---------|---------|---------|---------|
| slab       | 5000       | 5       | 1           | 100    | 4000    | 4000    | 4000    | 400     |
| slab       | 5000       | 5       | 1           | 100    | 3900    | 3900    | 3900    | 390     |
| slab       | 5000       | 5       | 1           | 100    | 3800    | 3800    | 3800    | 380     |
| slab       | 5000       | 5       | 1           | 100    | 3700    | 3700    | 3700    | 370     |
| slab       | 5000       | 5       | 1           | 100    | 3600    | 3600    | 3600    | 360     |
| slab       | 5000       | 5       | 1           | 100    | 3500    | 3500    | 3500    | 350     |
| slab       | 5000       | 5       | 1           | 100    | 3400    | 3400    | 3400    | 340     |
| slab       | 5000       | 5       | 1           | 100    | 3300    | 3300    | 3300    | 330     |



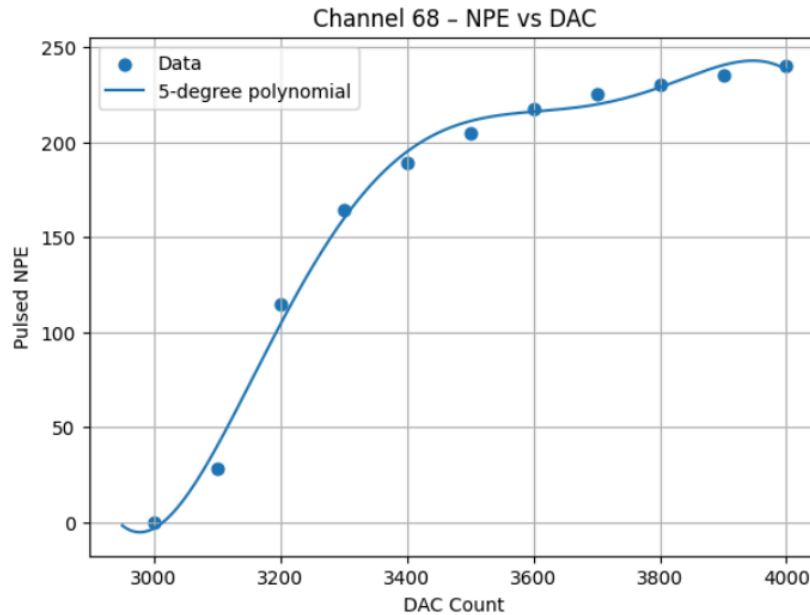
By plotting the area (proxy for NPE) of each different DAC count value in a separate histogram on a shared plot, we can see the change in NPE as a function of selected DAC counts. Going back and fitting each of the histograms with a Gaussian fit, we were able to extract the average NPE flashed at each DAC point. We converted the pulse area into NPE using each channel's specific SPE value.

Collecting all ten points and plotting them, I was able to fit a curve so that we can flash any voltage that we want. Utilizing SciPy curve fit, a variety of different functions were tested. Since we only had ten points and we wanted to avoid overfitting artifacts, a 5th degree polynomial was

decided to be the most accurate representation of "real" distribution. The plotted and fitted curve for channel 68 is attached below. In future flashing, I tried to flash the same amount of NPE light in each channel; the numbers for DAC counts were obtained using these fits. A summary of the sweeping tests with the raw data used in the fits can be found [here](#). A summary of the channel polynomial weights can be found in [Appendix].







The most important distributions were the channels where we could utilize the 4-in-a-Row trigger. As one of the few unprescaled triggers, this allows us to perform rigorous trigger validations of the DAQ. The back left 4-in-a-row flashable channels, Chan 22, 46, 70, 94, were all characterized in order to create a function that allows us to easily get the DAC counts for each channel at a specified NPE. In this path, we had a dynamic flashing range from SPE to 40 NPE. I utilized this to study how the detector triggers on differently sized pulses. The other available path had a channel which would only ever give SPE-level light, preventing us from performing NPE-dependent tests. Specifically, what we wanted to first study was the triggering efficiency of the Slab detector DAQ on PCCS-injected signals. To do this, we came up with a plan:

- Inject light in a 4-in-a-Row configuration
- At the same time, trigger the DAQ utilizing the S-Particle and allow it to trigger on the 4-in-a-Row signal.
- Since the triggering of S-Particle should not affect whether the DAQ triggers on 4-in-a-Row, we can collect every LED-injected signal and decide whether the detector correctly or incorrectly decided to trigger on the signal
- Repeat at different NPE levels

By creating an offline code method to check all received pulses and see if they should have triggered the detector, we can then verify if the triggering of the Slab detector. The first attempt to do this was using a previously created method made by Dariush, which took all the pulses in

```
] : get_dac_count(10)
```

DAC for NPE = 10

Channel 22: 3331.9196177227495

Channel 46: 3227.2790655931367

Channel 70: 3335.086493409141

Channel 94: 3026.489093783516

```
] : get_dac_count(40)
```

DAC for NPE = 40

Channel 22: 3999.999935778624

Channel 46: 3650.7450522430604

Channel 70: 3702.7017207405033

Channel 94: 3087.9500697359763

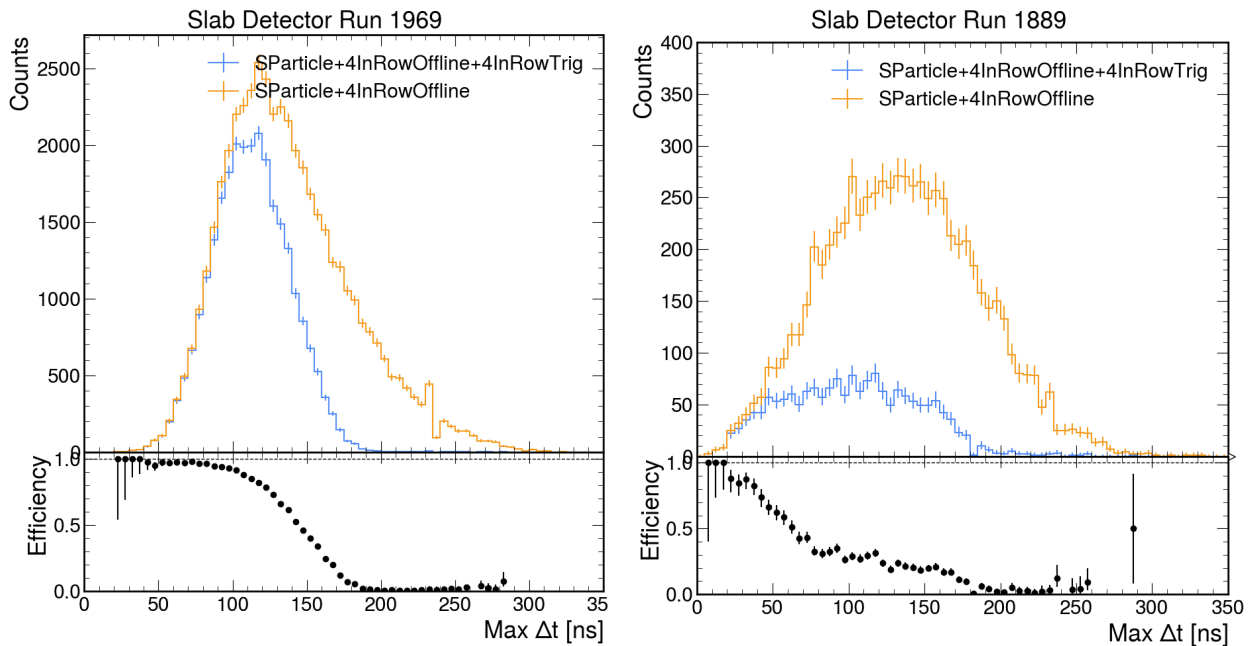
an event and tried to see if any of them lay within a 4-in-a-Row path. This method worked very well at detecting the 4-in-a-Row events, but since we are looking at S-Particle events, we need to ensure that the pulses have the ability to trigger the detector within the coincidence window.

To do this, the first attempt was to store information about the four pulses that Dariush's script found to satisfy the four-in-a-row requirement and calculate the longest possible time difference between the pulses, which we call maxDt. The issue with this method is that it can only calculate maxDt when there is only one pulse in each layer, as it cannot actually know which pulses caused the detector to trigger.

To calculate the efficiency of the detector, I always used two different masks, a numerator mask and a denominator mask, where the numerator is a strict subset of the denominator. In the following plots, the denominator mask selected events where there was an S-particle flash and that satisfied the offline requirement at some maxDt. The numerator contained all events from the denominator that actually triggered the detector. We would expect that only events with maxDt smaller than the coincidence window would actually trigger the detector.

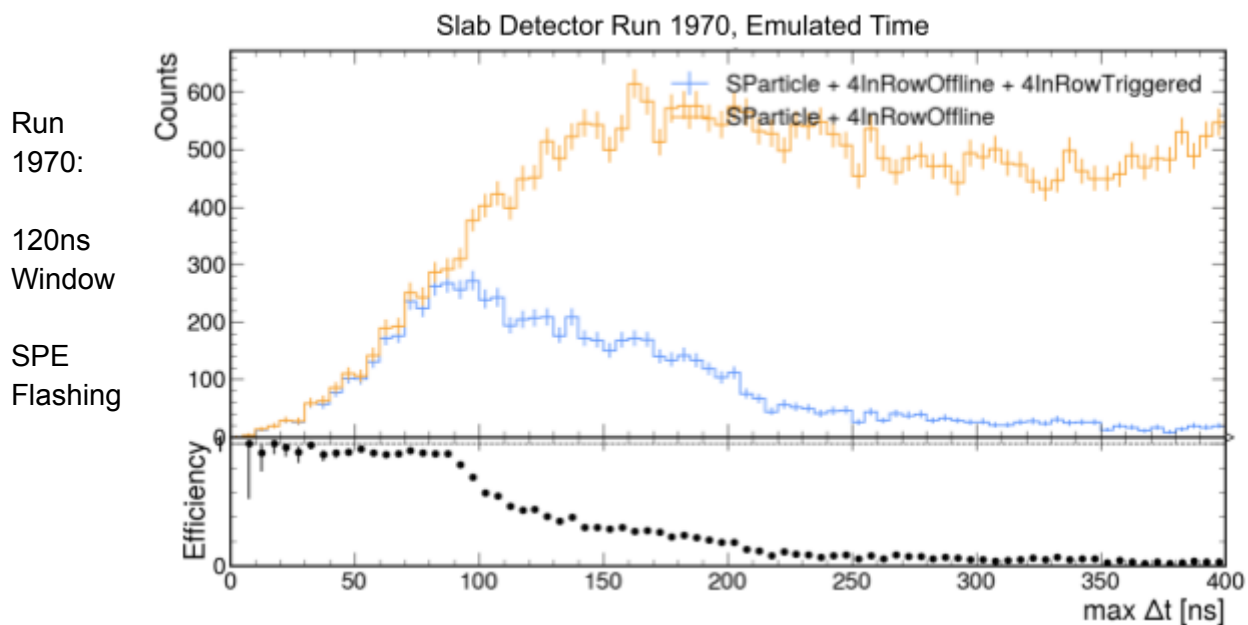
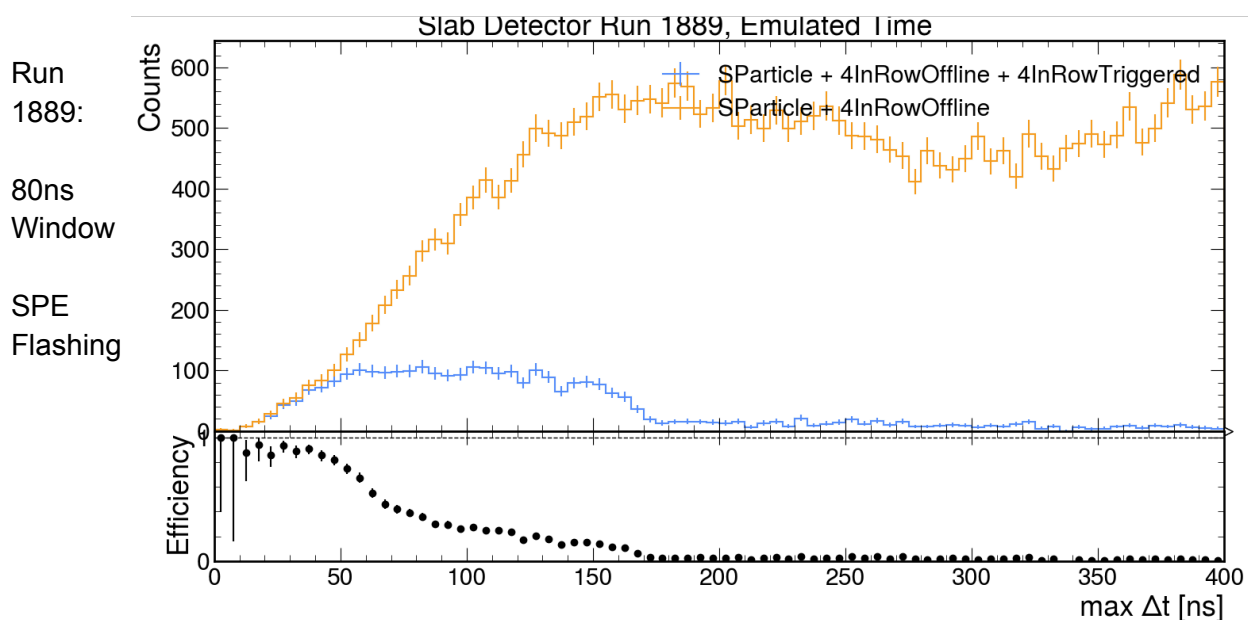
$$\varepsilon = \#(4\_in\_a\_Row\_Offline \& S\_Particle \& Triggered) / \#(4\_in\_a\_Row\_offline \& S\_Particle).$$

In the two histograms below, runs with 40 NPE flashing (left) and SPE flashing (right) at an 80 ns coincidence window. We see that the triggering efficiency remains very high after the supposed 80ns cutoff for 40 NPE, while the efficiency of the SPE flashing falls very quickly and is already around 30% at 80ns. The first hypothesis was to change the way we defined the time by introducing a more sophisticated offline requirement which would give us something closer to what the detector itself experiences.

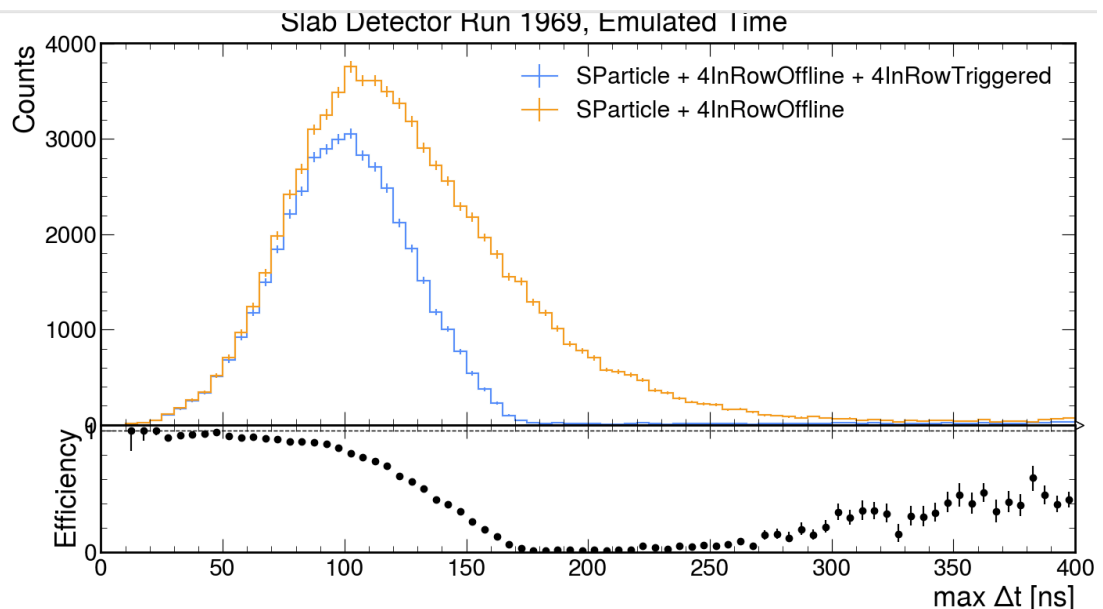


Our method to improve the quality of the time variable being plotted was by going back to the raw waveforms and emulating what is happening in the triggerboard. Utilizing the digitizer pulse thresholds, I created my own trigger primitives in the back left path. Since the digitizers are clocked at 2.5ns and the triggerboard is clocked at 8ns, it was very simple to see whether all four layers had successfully sent a trigger primitive in our trigger window requirements. Additionally, this let me know exactly which pulses caused the trigger, allowing me to have a better time variable for the event. The emulation system also allows us to take a look at SPE flashed data, as in SPE flashing multiple single-photon flashes in one event were common.

This emulation script is able to spit out ROOT files in the same format as our offline analysis files, allowing for easy inclusion of the new branches containing the emulation-calculated time and pulse times. The following plots contain triggering efficiency utilizing the emulation offline requirement.



Run  
1969:  
  
80ns  
Window  
  
40 NPE  
Flashing



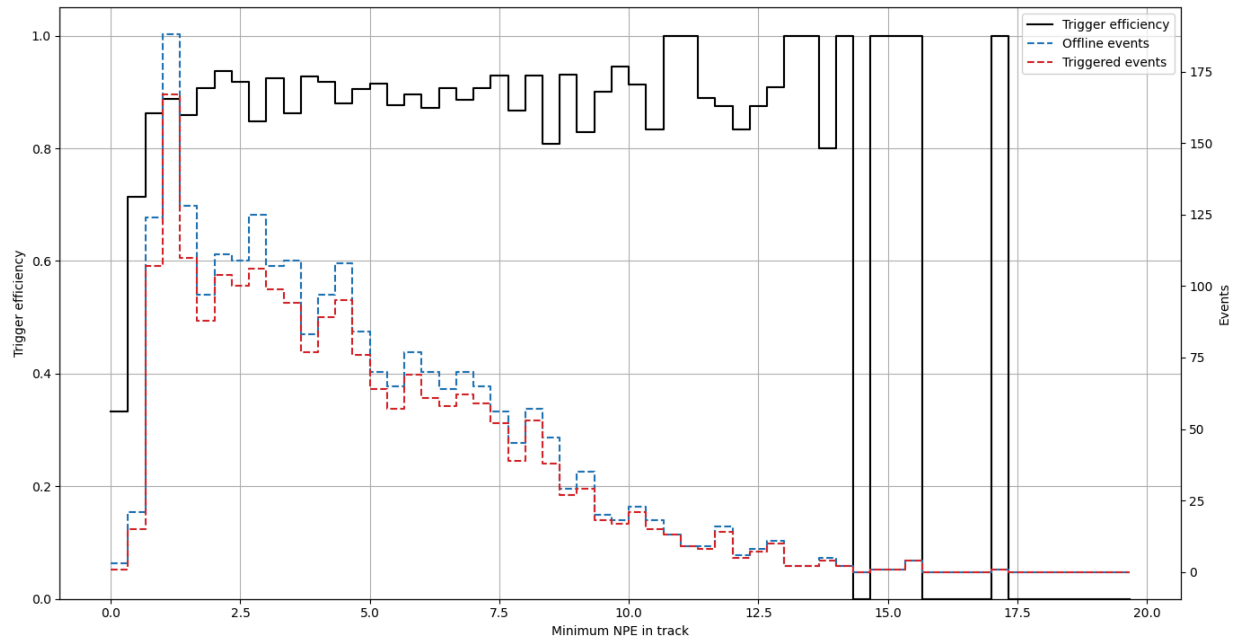
The SPE flashed data still has an efficiency drop below the coincidence window but with a slightly better-looking falloff, while the 40 NPE data appears almost unchanged from the previous timing method. The 40 NPE data being unchanged is not too surprising, as PCCS saturating pulses have a very well-defined time.

We can see this difference in timing effect by looking at the denominator of the SPE and 40 NPE flashed data. Where the 40 NPE denominator decays to around zero after 250ns, the SPE emulated time continues past 400ns. This is because when flashing SPE light, we essentially have the single photons enter the PMT with a uniform distribution related to the time of the LED pulse.

One remark that is important to make is that the efficiency of the SPE flashing data is very high for very quick pulses. This is a very important conclusion because it tells us that the detector works extremely well for particles moving near the speed of light. Almost all muons that are detected using the slab detector only need a 20 ns coincidence window to be detected. Since we expect mCPs also at highly relativistic speeds, it is safe to assume that mCPs moving even at half the speed of light would fall in this highly efficient region.

To further confirm that the behavior of the SPE flashing was working as intended, I returned to Run 1970, where I took a look at the highly efficient region below 100ns. To confirm that the detector triggering performance worked all the way down to the SPE level, I took a look at the four triggered pulses' NPE values, specifically calculating the minimum NPE in the 4-in-a-Row track. By plotting the numerator and denominator events on a histogram, I was able to show that the detector triggering efficiency works almost uniformly for min NPE in track > SPE.





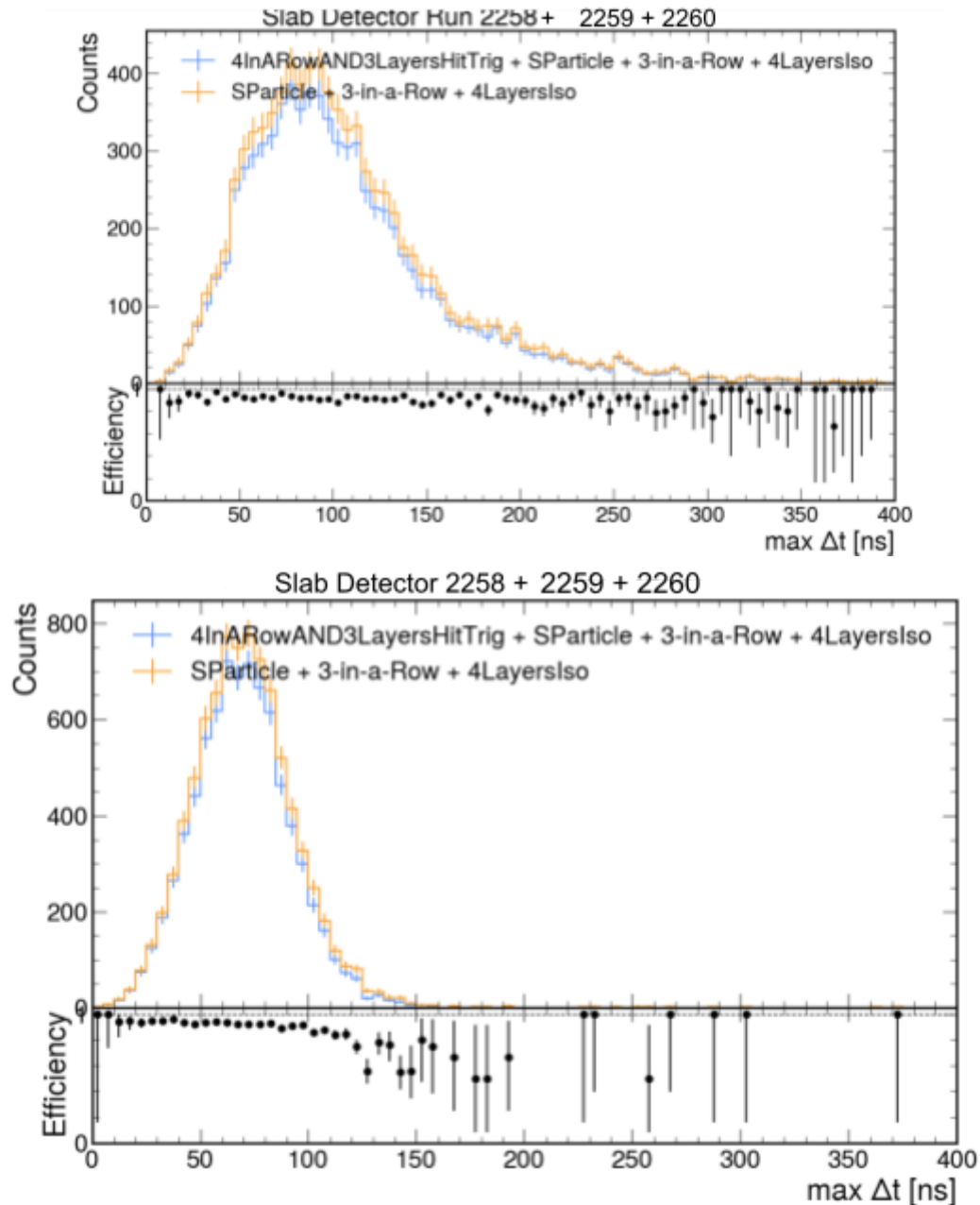
The last validation that I performed was looking at the efficiency of the unprescaled backup trigger, 3inaRowAND4LayersIso, which are events that have a 3-in-a-Row and have exactly one pulse in the last isolated layer. By triggering on S-Particle and flashing three of the channels in the back left path along with all other possible slabs, we could confirm the behavior of the backup trigger with a wider range of channel inputs.

Keeping the core channels in the blue four-in-a-row path, we cycled through all other slabs and triggered in 3-in-a-Row plus one extra slab. The CSV flashed 5 files at 5 Hz of 40 NPE signal in the main 3-in-a-Row and flashed a 4000 DAC count signal in the last channel.

To perform the offline validation, I modified Dariush's 4-in-a-Row detection to look for a 3-in-a-Row and a pulse in the flashed off-track channel using a command flag. The backup trigger seemed to work in a variety of different flashed channels. Specifically, we found successful triggering with channels: 33, 43, 45, and 68. The other channels we tried to flash had no photon output due to configuration changes made in the detector since the creation of the LED mapping. These changes are detailed in Appendix A.

|         |        |        |
|---------|--------|--------|
| Layer 0 |        |        |
| 22, 23  | 14, 15 | 6, 7   |
| 20, 21  | 12, 13 | 4, 5   |
| 18, 19  | 10, 11 | 2, 3   |
| 16, 17  | 8, 9   | 0, 1   |
| Layer 1 |        |        |
| 46, 47  | 38, 39 | 30, 31 |
| 44, 45  | 36, 37 | 28, 29 |
| 42, 43  | 34, 35 | 26, 27 |
| 40, 41  | 32, 33 | 24, 25 |
| Layer 2 |        |        |
| 70, 71  | 62, 63 | 54, 55 |
| 68, 69  | 60, 61 | 52, 53 |
| 66, 67  | 58, 59 | 50, 51 |
| 64, 65  | 56, 57 | 48, 49 |
| Layer 3 |        |        |
| 94, 95  | 86, 87 | 78, 79 |
| 92, 93  | 84, 85 | 76, 77 |
| 90, 91  | 82, 83 | 74, 75 |
| 88, 89  | 80, 81 | 72, 73 |

When plotting the data, at first it seemed that there was once again no sign of any type of turnoff in triggering efficiency. However, this is when plotting using the time variable “timeFit\_module\_calibrated,” a processed time variable that is designed to account for different types of time jitter in the digitizers while also fitting the start time of the pulse rather than using a voltage threshold.



However, when looking at the same exact data using the raw “time” variable, we see that the distribution of timing has dramatically decreased, along with the semblance of a well-behaved turnoff at 125ns. This tells us that in the future there will be advantages to working with the raw “time” variable during analysis when trying to understand the requirements of the triggerboard.

# Appendix A: PCCS Flashing Map

Described in this appendix are the current working LED locations.

Although the PCCS was designed with the ability to control all of the 192 LEDs that could be installed in the detector, many of the slabs do not have LEDs installed or have them installed in the wrong direction due to a systematic bug in the LED testing protocol. Furthermore, the dynamics of photon injection require photons to bounce around to reach the PMT face. As a result, only a small proportion of all LEDs can be used for signal injection. Furthermore, when performing detector interventions, some of the blades were swapped with blades that do not contain the soldered LV-headers, making them unsafe for PCCS Flashing. These blades have been disabled in the code, and channels with working LEDs have been moved into convenient locations to retain flashing functionality. Below are the mappings of the flashable slabs (left) and the soldered blade map (right), along with the channel swaps that have been performed during interventions. On the left, we see that there are two paths open for 4-in-a-Row operations: the back left path and the back right path.

| Known flashed Slabs |        |        |
|---------------------|--------|--------|
| Layer 0             |        |        |
| 22, 23              | 14, 15 | 6, 7   |
| 20, 21              | 12, 13 | 4, 5   |
| 18, 19              | 10, 11 | 2, 3   |
| 16, 17              | 8, 9   | 0, 1   |
| Layer 1             |        |        |
| 46, 47              | 38, 39 | 30, 31 |
| 44, 45              | 36, 37 | 28, 29 |
| 42, 43              | 34, 35 | 26, 27 |
| 40, 41              | 32, 33 | 24, 25 |
| Layer 2             |        |        |
| 70, 71              | 62, 63 | 54, 55 |
| 68, 69              | 60, 61 | 52, 53 |
| 66, 67              | 58, 59 | 50, 51 |
| 64, 65              | 56, 57 | 48, 49 |
| Layer 3             |        |        |
| 94, 95              | 86, 87 | 78, 79 |
| 92, 93              | 84, 85 | 76, 77 |
| 90, 91              | 82, 83 | 74, 75 |
| 88, 89              | 80, 81 | 72, 73 |

| PCCS Pluggings (Plugged in chan order top to bottom) |             |                 |
|------------------------------------------------------|-------------|-----------------|
| Layer 0                                              |             |                 |
| Central blade 2                                      | Aux blade 1 | Central blade 1 |
| Layer 1                                              |             |                 |
| Central blade 8                                      | Aux blade 7 | Central blade 7 |
| Layer 2                                              |             |                 |
| Central blade 4                                      | Aux blade 3 | Central blade 3 |
| Layer 3                                              |             |                 |
| Central blade 6                                      | Aux blade 5 | Central blade 5 |
| Ch 31                                                |             |                 |
| (241-34)                                             |             | Ch 78,          |
| Swapped                                              |             | (441-12)        |
| Swap to 242-34                                       |             | Swapped         |
|                                                      |             | Swap to 433-34  |

# Appendix B



## PCCS Constructing a Module

- a. Guide on 8020, 3D-printed rails
- b. Installation of PCB onto the Frame

Tab 11

Current bugs in the PCCS do not impede the use of the system. They are all features that would simplify use or ensure better system robustness. One of the key features of the Picos is that they will automatically reboot themselves if something goes wrong, ensuring that any bad state does not last. The RPi can all be accessed using SSH; they cannot be stuck in a bad mode unless the internet goes off in the cavern.

Bug Log:

**Bug:** The Multi-PCCS locks up after one row of config CSV; to stop a fast flash, we need to start the service

**Info:** It seems that the flasher receiver (the code that created the fast flashes) locks up and doesn't send the MQTT command "flash done".

**Work done:** A new untested feature expects Ctrl-C interruptions from the PCCS\_Controller Python code and sends a kill code via MQTT to the receiver service. This has not been ported to the installed production variants.

**Bug:** RPi Pico failing to communicate its full status back to the RPi5

**Info:** In January 2025, I developed a way to fix desync from the Picos by having the Pico communicate information about the last received data back to the RPi during the SPI communication. This feature was being used not only to fix the Picos' SPI buffer when an issue is detected, but also to detect whether Picos are connected.

**Bug:** PCCS\_Control.py Error manager not fully verified

Tab 12

The final big contribution I bring to the milliQan calibration is thus a glimpse into the complex dynamics of the triggerboard, while confirming that it has performed well during the final months of Run3 data collection.