Module II: CosmicWatch

The Desktop Muon Detector v3X

Exercise Document

Contacts:
Spencer Axani (saxani@udel.edu)

Revised 03/0/2025

Goals:

Over the next five lab sessions, you will assemble a **CosmicWatch Desktop Muon Detector v3X** and use it to conduct experiments exploring how cosmic rays interact with matter and the environment. Once your detector is built, you will design and carry out investigations on topics such as cosmic ray flux variations with altitude, shielding effects, or angular dependence. You will collect and analyze data, interpret your findings, and present your results, honing both your technical expertise and scientific communication skills.

The GitHub repository contains all pertinent material for this Exercise.
https://github.com/spenceraxani/CosmicWatch-Desktop-Muon-Detector-v3X

Within the repo, you can find an **Instructions Manual** that elaborates on:

1. Sources of ionizing radiation (Ch. 2)
2. How particles interact with matter (Ch. 3)
3. Particle detection methods (Ch. 4)
4. What are the CosmicWatch Desktop Muon Detectors and how to build them (Ch. 5&6)
5. Example measurements, both performed and exploratory (Ch. 7)

Additionally, the GitHub repository includes:

1. The Firmware for the detector
2. Printed Circuit Boards (PCB) information and Gerber file for manufacturing the PCBs
3. Example data, and plotting scripts

You should start off by downloading the GitHub repository to a QSEG830/PHYS646 working directory on the lab computer.

Prior to this lab, we gave an overview of the CosmicWatch project. Today, we build one. Keep track of your work and answer the questions in this document in your **Laboratory Notebook**. It will be submitted later.

Equipment Required

- Arbitrary Function Generator (AFG1062)
- Mixed Domain Oscilloscope (MDO34) & BNC cables and probes
- Printed circuit boards and components prepurchased

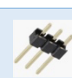
Feel free to rip off the last two page of this document, which includes the circuit diagram for the printed circuit board and rendering.

One word of caution; please do not solder to the PCB while it's connected to a power source.

We'll start by giving you a tutorial through Zoom on soldering surface mount technology (SMT) components and standard integrated circuits (IC).

Part I: Populate the Main printed circuit board up to the first amplifier.

Solder the following surface mount technology (SMT) components to the printed circuit board (PCB). We first build the amplifier on the **Main PCB** and subsequently test it.

	Reference	Value	Component	Description	Notes	Link
1	C3, C19	4.7uF		CAP CER 4.7UF 50V X7R 0805		
2	C21, C22, C18, C6	1uF		CAP CER 1.0UF 50V X7R 0805		
3	C12, C4, C25	0.1uF		CAP CER 0.1UF 50V X7R 0805		
4	C13	470pF		CAP CER 470pF 50V X7R 0805		
5	FB1	Ferrite Bead		FB ULTRA 0805 31 OHM 6A .015DC		
6	R23	100		RES SMD 100 OHM 1% 1/8W 0805		
7	R3, R4	30		RES SMD 30 OHM 1% 1/8W 0805		
8	R5	620		RES SMD 620 OHM 1% 1/8W 0805		
9	R25, R22	1k		RES SMD 1K OHM 1% 1/8W 0805		
10	R26	100k		RES SMD 100K OHM 1% 1/8W 0805		
11	R10	22.1k		RES SMD 22.1K OHM 1% 1/8W 0805		
12	D1	BAT54s		BAT54S diode	Limits input voltage levels to a predefined range	
13	U7	LM4040_2.5V		IC VREF SHUNT 0.5% SOT23	Sets up the 2.5V reference voltage	
14	U2	TPH2506		IC OPAMP 2 CIRCUIT	Has direction!	
15	U4	RP Pico	In Bag 	Raspberry Pi Pico	No Header Pins. Flat against board.	
16	P2	BNC	In Bag 	CONN BNC JACK R/A 50 OHM PCB	Mount on top side of board. Flat against board.	
17	J5	3 pin header for TP1/2/3	In Bag 	Test point connections	Mount long side up on top side of board.	

The amplifier is now complete! Let's do some tests.

Part II: Inject a known pulse into the circuit and see the amplifier's response.

You've been using the AFG1062 for several of the Exercises, however you haven't yet exploited the "A" in the name—the "Arbitrary" part of the function generator. An arbitrary function generator (AFG) can produce an arbitrary electrical waveform by digitally synthesizing signals using a digital-to-analog converter (DAC). The device generates a waveform by storing discrete data points in memory, which define the shape of the desired signal. These data points are sent to the DAC, which converts the digital values into an analog voltage. Let's program the AFG1062 to output the shape of our expected **SiPM signal** waveform to help us understand the amplifier we just built.

Procedure

1. Setting Up the Arbitrary Function

1. Open the Arbitrary Function (Arb) menu on the AFG.
2. Set the window size to define the Arbitrary function:
   - Configure FREQ = **1 MHz** (or PERIOD = **1 µs**).
3. Set the waveform High and Low levels:
   - High level: **1 V,** Low level: **0 V**

2. Defining a New Waveform

1. Click on the **Others** button, then select **New**.
2. Set the number of points within the 1.00 µs window:
   - Number of points: **1000** (results in **1 ns per point**).
3. Ensure **interpolate** is turned ON.
4. Click **Edit Points** and manually add the **Points and Voltages** in Table 1.
5. Click Write -> Inter (Enter) -> USER0 and Save

3. Viewing the Arbitrary Waveform

1. Load the saved waveform:
   - Select USER0 and click **CallOut**, then turn ON **Out1**.
2. Connect **Ch1 of the AFG** to Ch1 of the MDO34 oscilloscope. Set the scope to 1MOhm and AC coupled.
3. Observe the waveform to confirm the correct output.

4. Configuring Burst Mode:

1. Click **Mod** → **Burst**.
2. Set the **Burst Period** to **250 ms**.
3. Set **#_Cycles** to **1**.
4. Enable **triggering** to observe waveform updates.

Let's change the amplitude to a more realistic signal.

Point	Voltage [V]
1	**0**
91	**0**
119	**0.96**
121	**0.98**
129	**1**
139	**0.96**
161	**0.85**
190	**0.7**
248	**0.48**
322	**0.31**
373	**0.22**
426	**0.16**
502	**0.1**
615	**0.05**
785	**0.015**
964	**0**

Table 1: The points and voltages to input into AFG1062

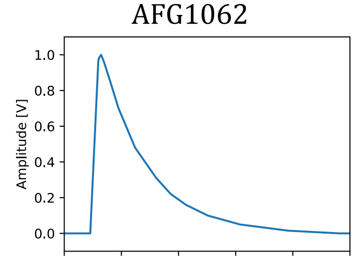


Figure 1: The waveform from Table 1.

5. Adjusting Amplitude

1. Click **Arb**, then use the **High Level** setting to adjust the amplitude. Make sure that the **low-level stays at 0mV**.
    1. Set the High Level to **10 mV**.
2. Connect AFG **Out1 to CH1 of the oscilloscope** and verify the output by setting up the trigger.

We will now inject this waveform into the circuit we just built for further analysis.

Figure 2: Injecting the pulse into the amplifier circuit and measuring it's response on Test Point 1 (TP1)

6. Measuring Circuit Response

1. Use a BNC tee at the AFG1062 to send the pulse to the BNC input of your Main Board. As shown in Fig. 2.
2. Using a BNC mini-grabber cable, connect the red lead to Test Point 1 (TP1), and the black lead to the GND connection on the PCB. Connect this BNC cable to the **CH2 of the oscilloscope (AC coupled, MΩ termination, try using short BNC cables to minimize reflections)**.
3. TP1 is connected to output of the amplifier (after AC coupling and adding a 108mV bias):
    - Estimate the gain of the amplifier by comparing the mV/div scale on CH1 and CH2 [2pt]. Note, you can find tune the scale by clicking the scale knob.
    - Identify and measure any time delay [2pt].
    - Compare the signal shape between input and output [2pt]. Amplifiers have a bandwidth over which they can operate. Frequencies above this range will be

attenuated, similar to the RC circuit in Exercise 2 & 3. Increasing the gain on an amplifier, often reduces the bandwidth over which they can operate. See, for example, Fig. 1 of the TPH2506 op amp datasheet.

4. Vary the input signal amplitude:
   - As a function of pulse input, from 2mV to 100mV, document the gain [5pt].
   - At an input pulse of ~300mV, the amplifier is hitting the maximum output range of the amplifier – the output is saturated. Describe the waveform as you keep increasing the pulse size beyond ~300mV [2pt].
     - Often, front ends electronics include several amplifiers at different amplification levels, such that they can accommodate a larger input range. This improves the "**dynamic range**," i.e. the range over which you can take data. For example, the IceCube neutrino detector's single photon sensors have three different gain levels in-case of saturation.

From first order approximations (what you would find in a standard electronics textbook), the voltage gain of this **non-inverting amplifier** circuit is:

$$A = V_{out} / V_{in} = 1 + (R5/R3) = 1+620/30 = 21.6$$

Do your measurements agree with this? [2pt]

In reality, particularly for high-speed electronics, there are many other effects that can change the effective gain. For example, parasitic (unwanted) capacitance and inductance due to the physical design of the PCB, or the frequency response of the amplifier itself. Designing the analog electronics (particularly at high frequencies) is often referred to as Black Magic or an art form, as first order approximations rarely hold, and integrated circuits (like our operational amplifier TPH2502) require several dozen parameters to roughly describe their proper operation. We learn about this in PHYS645 Electronics for Scientists.

Convert your measured gain voltage to decibels. [2pt]

Part III: Finish building the analog circuit

After amplification, the amplified signal feeds into a comparator circuit and a precision peak detector circuit. The comparator circuit acts as a "**discriminator**," which outputs a digital HIGH pulse to the microcontroller (MCU) if an amplified signal is above some reference voltage. The **precision peak detector** holds the peak of the amplified pulse for sufficient time that we can measure it. Let's now build these circuits.

	Reference	Value	Component	Description	Notes	Link
1	U5	TPH2506		IC OPAMP 2 CIRCUIT		
2	R14	200k 226k		RES SMD 200K OHM 1% 1/8W 0805		
3	R27, R24, R19, R21, R17, R12, R30	1k		RES SMD 1K OHM 1% 1/8W 0805		
4	R6	0		RES SMD 0 OHM		
5	C5	100pF		CAP CER 100pF 50V X7R 0805		
6	C16	0.1uF		CAP CER 0.1uF 50V X7R 0805		
7	C17	4.7uF		CAP CER 4.7uF 50V X7R 0805		
8	C11	1uF		CAP CER 1uF 50V X7R 0805		
9	D4	BAT54S		DIODE ARR SCHOT 30V 200MA SOT233		
10	R8, R18, R28, R29	100		RES SMD 100 OHM 1% 1/8W 0805		
10	R7	22.1k		RES SMD 22.1k OHM 1% 1/8W 0805		
11	R31	2k		RES SMD 2K OHM 1% 1/8W 0805		
12	Q2	2N7002		MOSFET SOT23 N 60V 5OHM 150C		

Ideally, we aim to extract is the **SiPM peak voltage,** since this should be roughly proportional to the energy deposited by the particle interacting in our detector.

Why not simply sample the SiPM waveform at giga samples per second (GSPS) speeds to measure the SiPM peak voltage? Like what could be done using the MDO34 oscilloscope? [2pt]

We do not use a fancy ADC to do this, instead we will use a **low-power (0.5W) microcontroller (MCU) costing just $4**, the Raspberry Pi Pico, which samples at **1 MSPS—2,500 times slower** than the high-end MDO34 oscilloscope. Obviously, this is too slow to be able to extract the SiPM peak voltage directly from the raw waveform. This is where the **peak detector** comes in. By holding the peak value of the amplified signal some time, our slower MCU can accurately measure its amplitude. With a known gain, we can then back-calculate the original **SiPM peak voltage**.

1. Peak Detector Analysis (TP2)

1. TP2 is connected to the **peak detector output**, which holds the (amplified peak voltage)/2 for measurement. Inject 20mV pulses from the AFG1062.
2. Move the BNC mini-grabber from TP1 to TP2 (1MOhm termination is necessary for this).
    1. Use the oscilloscope to describe the waveform [2pt]. If we always trigger the ADC of the MCU at the same time (on the rising edge of the amplified pulse), we should be able to get a consistent measurement from the peak detector.

The **trigger threshold** on the comparator circuit is controlled by the MCU, so the next step requires uploading some firmware. Locate the **TestingTriggerThreshold.uf2** file in the Software folder. This file configures the MCU to set the trigger voltage to a specific value.

2. Firmware update

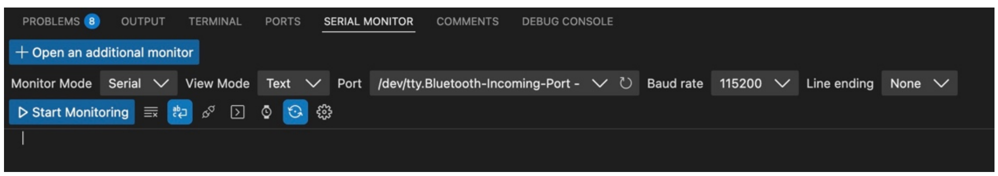
1. Hold the white boot select button down on the RP Pcio, while plugging it into the computer via a micro USB cable.
2. The RP Pico will appear as a removable drive in My PC.
3. Drag and drop (or copy and paste) the **Exercises/TestTriggerThreshold.uf2** into removable drive.
4. The Firmware is now uploaded, and the removable drive will now disappear. The MCU will reboot and run the code. The code sets a trigger threshold.

3. The Trigger Signal (TP3)

1. When the amplified pulse exceeds the trigger threshold, a **digital HIGH (3.3V) pulse** is generated to trigger the RP Pico's data acquisition, just like the oscilloscope's trigger threshold.
2. Change the input pulse and observe the trigger behavior:
    - At 20 mV input from the AFG, describe the TP3 output. [2pt]
    - Decrease the size of the input pulse until the trigger pulse goes below roughly 2.5V. Anything below this, will not be considered HIGH. At what input voltage does this happen? [2pt].

4. ADC Measurement

1. Roughly 2us after triggering, the firmware commands the MCU to measures the peak detector waveform on its ADC.
2. The MCU will then print the measured ADC value to the Serial Port (the USB connection). That means, if you setup your computer to listen to the asynchronous data flowing through that cable, you should see the ADC measurements in real-time.
3. Read the ADC value through VSCode:
    1. Open VSCode through the Anaconda Navigator.
    2. On the left side pane, click on the Extensions button  .
    3. Type "Serial Monitor" and install. This added a way to listen to a serial port in the Serial Monitor, as shown below.



    4. With the detector plugged into a USB port on the computer, it will be listed as a COM port in the **Port** pull down menu. Once you figure out which COM port it is plugged into (perhaps by unplugging and plugging back in), you can click the **Start Monitoring** button. The Serial Monitor should now be showing the data flowing out of your detector, specifically the measured ADC amplitude. With some patience you could make the following figure. This figure would allow you to convert the ADC value to an input pulse amplitude. This is one step in calibrating a particle detector.

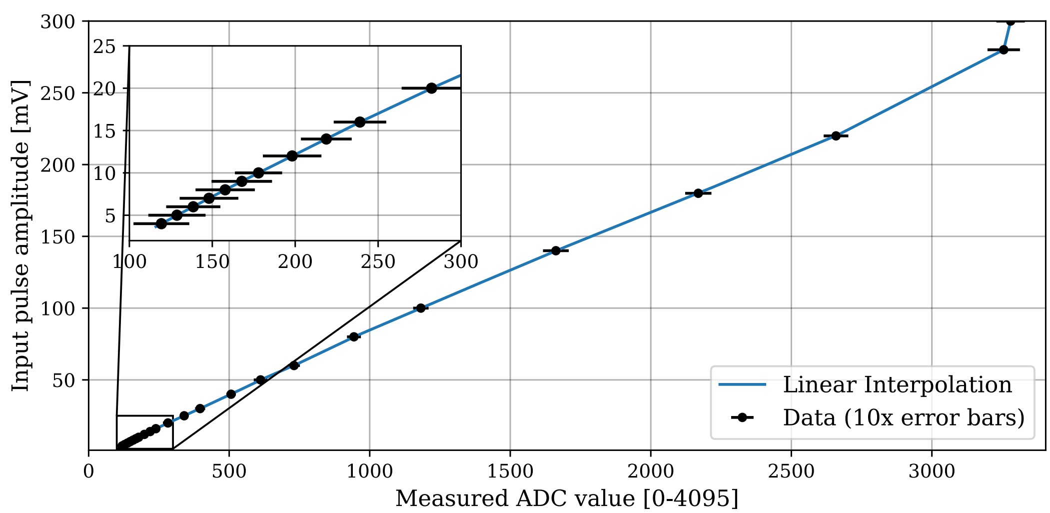


Figure 2 The calibration curve for the detector, which converts a measured ADC value from the peak detector to the input pulse amplitude. The blue line is a linear interpolation between the datapoints. The error bars are drawn with 10x the scale, for visual purposes.

If the ADC reported a value of 1,500, what SiPM peak voltage would that correspond to [2pt]?

Later, when we are taking data with our detector, this curve is used to convert what the ADC measurements to a SiPM Peak voltage, through linear interpolation.

Part IV: DC-DC booster power supply for the SiPM

We will now populate the part of the circuit responsible for suppling the +30.2V bias voltage for single photon sensor, the SiPM. This is known as a DC-DC booster.

	Reference	Value	Component	Description	Notes	Link
1	U1	MAX5026		DC-DC Booster, IC REG BOOST ADJ 260MA SOT6	Important, has direction! Look for little dot. Long line on silkscreen	
2	L1	47uH		FIXED IND 47UH 190MA 4.86OHM SMD		
3	D3	BAT54WS		DIODE SCHOTTKY 30V 200MA SOD323	Has direction. Hard to see line on component.	
4	R2	158k		RES SMD 158k OHM 0.5% 1/8W 0603		
5	R1	6.65k		RES SMD 6.65k OHM 0.5% 1/8W 0603		
6	R9	100		RES SMD 100 OHM 1% 1/8W 0805		
7	C1, C14	1uF		CAP CER 1UF 50V Y5V 0805		
7	C2	4.7uF		CAP CER 4.7UF 50V X7R 0805		
8	6-pin header	2.54mm 2x3 pin	In Bag 	CONN SOCKET 6POS 0.1 GOLD PCB	Top side of board	

Once you supply power to the MCU via the micro-USB cable, the DC-DC booster will output ~30.2V between the HV and GND connection (see 6-pin header silk screen markings) going to the SiPM. Use the multimeter to measure and confirm the correct voltage [2pt].

After verifying the voltage, we are safe to test our circuit using the SiPM. Use a completed SiPM-scintillator combo from the lab, plug it into your 6-pin header. Connect the BNC output to the oscilloscope (50 Ohm terminated and AC coupled). Set the trigger threshold to 10mV, and the scales to 1us/div and 10mV/div.

Do you see any pulses? Trigger on a few of them and describe them [2pt]?

There's also a population of small sub-mV pulses. Can you see them? It can help to set the Ch1 bandwidth to 20MHz to reduce high frequency noise. Those pulses are thermionic emission from the silicon. They look identical to actual single photons, and we commonly refer to them as the **Dark Noise Rate**. Count how many you get in a window and estimate the **Dark Noise Rate** [2pt].

Look up the **Dark Noise Rate** in the MicroFJ 60035 SiPM datasheet [2pt].

Unlike other types of single photon detectors, SiPMs have a very large dark rate, which can make them rather problematic for single photon detection. If you see a single photon, how do you know it wasn't thermionic emission? For context, a 2" diameter photomultiplier tube (PMT, a common single photon device) may have a dark rate of 100Hz, five hundred thousand times lower per unit area than the SiPM.

Larger pulses originate from brighter events in the scintillator – that is, more energy deposited. You can connect a mini-grabber to Ch2 to have another look at TP1, TP2, and TP3. Verify that these test points provide the amplification, peak detection, and trigger that we need to move forward.

Part V: The Digital side of the board

Let's finalize the **Main PCB** by adding the topside digital components. These components are used to perform various digital operations. For example, the RJ45 connector (Ethernet connector), is used to send digital HIGH pulses to connected detectors to tell them that your detector triggered on something. Other components, like the BMP280 sensor, digitally communicates with the MCU to report the temperature and pressure.

	Reference	Value	Component	Description	Notes	Link
1	Coincidence connector	RJ45	In Bag 	RJ45, 8p8c right angle	Top side	
2	Reset Button	Reset Button	In Bag 	SWITCH TACTILE SPST-NO 6x6x9mm	Top side	
3	Buzzer	Buzzer	In Bag 	BUZZER MAGNETIC	Top side – has direction	
4	Temp/Pressure sensor	BMP280	In Bag 	BMP280-3.3V	Top side – has direction	
5	LED 3mm	LED 3mm	In Bag 	3mm LED	Top Side, note how far to put it in . Note direction	
6	LED 5mm	LED 5mm	In Bag 	5mm LED	Top Side, not how far to put it in. note direction!	
7	OLED screen	OLED screen	In Bag 	128x64 Yellow Blue OLED	Top side. Note GND connection of OLED must be one of the side pins.	
8	U8	SD Card socket	In Bag 	Micro SD Memory Card Slot Holder Sockets	Bottom side	

Let's now upload the final firmware to the detector.

2. Final Firmware update

1. Repeat the firmware upload procedure from Part III, with the firmware file: **Firmware/CosmicWatchFirmware.uf2**

3. Use the Serial Monitor to troubleshoot

1. When the detector boots up, it performs a few checks and prints the results to the Serial port. Have a look for any errors. It may complain that there isn't an SD card inserted, feel free to grab one from the lab.

3. Adjusting detector settings

1. You do not have access to modify the firmware. Instead, a config.txt file is created on the micro SD card that allows you to modify user variables. At the moment, these only include:
    1. Detector Name: The name of your detector.
    2. The trigger threshold value: the default threshold is set to 155mV.

Open the microSD card on your computer. Try renaming your detector "Luke." There are a few other hidden easter eggs as well. Feel free to let us know if you have a good idea on other names☺ Feel free to change it back to something else. Or if you have suggestions for other user options that you would like control of.

Description of digital components:

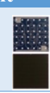
1. Ethernet RJ45 jack: this is the coincidence connection. It's used to communicate between detectors to identify coincidence events and provide power.
2. Reset button: used to reset the detector: occasionally, it doesn't work, and you may have to hard reset by manually unplugging and plugging the detectors back in.
3. Buzzer: enables users to play sounds.
4. BMP280: the temperature and pressure sensor.
5. LEDs: they flash on every event (white 5mm) and on coincident events (blue 3mm)
6. OLED screen: writes data to screen for realtime rate measurements.
7. MicroSD card socket: used to save data automatically to a SDcard.

Part VI: The SiPM board

The last part is to build the SiPM board and interface it with the scintillator. A **Silicon Photomultiplier (SiPM)** is a highly sensitive **solid-state photon detector** that operates based on **avalanche photodiodes (APDs)** working in **Geiger mode**. It is designed to detect very low light levels, down to **single photons**, making it a key technology in **particle physics, medical imaging, and astrophysics**. Please be careful with the SiPM, it represents half the cost of the full detector. Please avoid scratch the surface.

	Reference	Value	Component	Description	Notes	Link
1	R11	10		RES SMD 10 OHM 1% 1/8W 0805		
2	R13	100		RES SMD 100 OHM 1% 1/8W 0805		
3	C15, C8	0.1uF		CAP CER 0.1UF 50V X7R 0805		
4	C24, C9	10nF		CAP CER 10000PF 50V X7R 0805		
5	C10. C23	10pF		CAP CER 10PF 50V C0G/NP0 0805		

Before adding the SiPM, let's go through the procedure together in class.

	Reference	Value	Component	Description	Notes	Link
1	U6	SiPM	In Bag 	SiPM_MicroFJ−60035−TSV	Direction is extremely important	
2	6 pin SMT (on SiPM board)	2x3 pins	In Bag 	CONN HEADER SMD 6POS 2.54MM	SiPM PCB, align it well with the footprint.	
3	2x Standoffs on SiPM PCB	Standoff	In Bag 	1/8" Hex Size, 7/16"" Length, 0-80 Thread Size	Mount on bottom side (non SiPM side),inner two holes (91780A029)	
4	2x screws to mount the standoffs to SiPM board	Standoff	In Bag 	0-80 Thread Size, 1/4" Long	Screw through SiPM side in board.	

A **scintillator** is a material that emits **visible or ultraviolet (~420nm) light (scintillation)** when excited by ionizing radiation. This property makes scintillators crucial for **particle detection**, where they convert high-energy radiation into detectable light signals.

1. Add reflective foil to scintillator

1. Wrap the scintillator with the aluminum foil, leaving a space open for the SiPM. Use a small amount of black electrical tape to hold it in place.
2. Add some optical gel to the scintillator interface surface. And the SiPM Surface.
3. Add a silicone pad in-between.
4. "Use the #2 screws to attach the SiPM PCB to the scintillator.
5. Wrap the full assembly in several layers of black electrical tape. Please, it's important to do a very good job wrapping the tape smoothly. Cut the corners such that the tape folds over. Wrap in perpendicular directions to ensure good light tightness. We are working with a single photon sensor, so the smallest hole will prevent the detector from working.

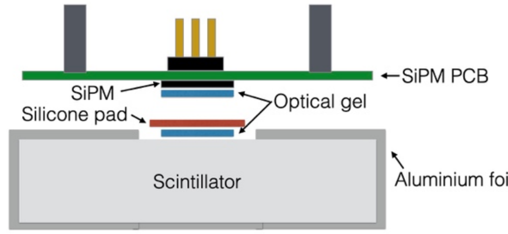


Figure 3 The SiPM-scintillator assembly.

	Reference	Value	Component	Description	Notes	Link
1	Plastic Scintillator	50x50x10mm	In Lab	Drill holes for #2 screws. Drill using #52 bit		
2	Reflective foil	Alum foil	In Bag	Reflective tape for scintillator		
3	Optical Gel	Optical gel	In Lab	Optical coupling		
4	Silicon pad	0.3mm thick silicone	In Bag	Optical coupling		
5	#2 screw for SiPM PCB/scintillator	#2 5/16"	In Bag 	18-8 Stainless Steel, Number 2 Size, 5/16" Long	Screw through SiPM side in board.	
6	Black electrical tape	Tape	In Lab 	Black tape	Optical isolation	

Once complete, you can plug the SiPM-scintillator assembly into your completed main board. Once powered on, verify that you see Hz-level count rates.

Congratulations! If all went well, you should now have a general-purpose particle detector.

The OLED will report the detector trigger rate and the Poissonian statistical uncertainty. Pay attention to the uncertainty. Take sufficient data. For example, if you are looking for a 1% effect, you want the statistical uncertainty to be less than 1%. For a Poisson process, this is 10,000 events! That is, the uncertainty on the number of random events, when the average is 10,000 is sqrt(10,000) = 100, which is 1%. You essentially need more than 10,000 events percent level effect!

With 5 minutes of data, record the master count rate (TOTAL) from your detector from the OLED screen [2pt].

The rate accounts for the detector deadtime. **Detector deadtime** is the period after detecting an event during which a detector is unable to record another event. It occurs due to limitations in the detector's electronics, signal processing, or recovery time after a pulse. Deadtime is a critical factor in high-rate experiments, as it affects counting efficiency and can lead to event losses if not properly accounted for. The **Detector livetime** is defined as the amount of time that the detector is able to take data. It amounts to the total run time minus the deadtime.

Livetime = (run time of measurements – deadtime)

Rate [Hz] = (number of events)/Livetime

The deadtime is extremely important for all detectors which measure rate. In the firmware, the amount of time it takes to make any calculation which prevents the CPU from monitoring for pluses, is added to the deadtime. Literally, when the firmware adds 2+2, we measure how long that calculation takes and add it to the deadtime.

The detectors use dual cores on the CPU to reduce the deadtime. One core monitors for triggers and records the peak detector for pulses, adds those events to a ring buffer for the SDCard, and handles communication between detectors. The second core operates peripherals that induce large deadtime (the OLED screen, BMP280, and writing to the ring buffer to the SD card). A **ring buffer** is a **circular memory structure** used to store and manage events in a continuous, efficient way. Since particle detectors often generate **high-**

rate event streams, ring buffers help handle data flow without overwhelming memory or losing critical information.

Part VII: Natural radioactivity.

Radiation is all around us, naturally present in our environment and emitted from both cosmic and terrestrial sources. It comes in different forms, including **alpha, beta, and gamma radiation**, each with distinct properties and interactions. **Alpha particles** consist of two protons and two neutrons, making them relatively heavy and highly ionizing, but they have very short range and can be stopped by something as thin as a sheet of paper or even human skin. **Beta particles** are high-energy electrons or positrons, which are more penetrating than alpha radiation but can still be blocked by a few millimeters of plastic or aluminum. **Gamma rays**, on the other hand, are high-energy electromagnetic waves that can travel long distances and penetrate deeply into materials, requiring dense shielding like lead or thick concrete to reduce exposure.

While radiation is a natural part of life, excessive exposure can pose health risks, making radiation detection and shielding important in many scientific and medical applications.

Even common materials can have a measurable amount of radiation with these detectors.
Record the rate with Potassium salt on top of the scintillator and compare to the master rate you recorded previously [2pt]. Is it statistically significant [2pt]?
Potassium salt contains a small amount of Potassium-40, an isotope that emits a 1.4MeV gamma ray when it decays. If you have a bunch of bananas, with sufficient data, and optimum banana placement, you can also see the same radiation.

Bring your detector near the Fiesta ware in the front of the lab. These are rare antique plates from the 40s-60s that were coated in an orange Uranium Dioxide coating that is rather radioactive. Please be careful with them, they are my personal plates☺
Record the rate with the detector sitting atop the plate [2pt].

While waiting for your partner to complete their detector, feel free to borrow other radioactive source to see how the detector responds. You may notice that at high count rates, the LED pauses for a bit, this is when the ring buffer is full, and we add extra deadtime for the SD card to read out the buffer. At even higher rates, the detector may stop working – we're working updating the firmware to handle this.

Once you have two detectors, return the radioactive sources and continue.

The BNC on the Main PCB connects up directly to the SiPM output. Connect this to the oscilloscope. Set the trigger to "Normal," 50Ohm termination, and limit the bandwidth to 20Mhz. Connect your partners detector to the other oscilloscope channel. Place the detectors one-on-top of the other. Do you ever see events at the same time on the oscilloscope (coincident events)? [1pt]
Do these events have any other notable characteristic compared to the non-coincident events? [2pt]

While outdoors, most of the down-going ionizing radiation consists of cosmic ray muons, though some low-energy electrons and positrons are also present. However, when inside a building shielded by several feet of concrete, these lower-energy electromagnetic contributions are effectively eliminated, leaving just the highly penetrating cosmic ray muons. The coincident events you observed originate from muons produced 15–20 km above sea level, which travel through the atmosphere, penetrate the building, and pass through both detectors. This ionizing radiation is also passing though you, destroying DNA and ionizing cells.

Assume the half-life of the muon is 1.56us and that they are travelling near the speed of light, c. That is, half of the muons will decay after 1.56us. Without special relativity, how far would the average muon travel before decaying [2pt]? If time dilation and special relativity wasn't true, the muons you observed would represent a stupendously small fraction of the muons produced in the upper atmosphere. How many decay lengths does 10km represent [2pt]? 10 km is roughly the altitude of a typical commercial airplane flight.
If you assume a muon flux at sea level of 1 muon per centimeter squared per minute, what would be that equivalent flux at 10km [2pt]?

About 10-15% of your annual radiation dose comes from cosmic ray muons punching through your body. Estimate the number of equivalent years of radiation dose you would receive during an eight hour flight had special relativity been incorrect [2pt]. Thank goodness for Einstein. You can verify special relativity over rather small altitudes differences. Simply comparing the muon rate on top of a relatively high building to ground level is sufficient.

Figure 4 Prof. Jungfleisch measuring the muon rate atop the Eiffle tower.

Part VII: Extracting cosmic ray muons (coincidence mode).

When you and your partner have completed your particle detector, we can then set them up in **coincidence mode** to extract cosmic ray muons. Muons are highly penetrating particles – they are the reason why we often build particle detectors kilometers underground. For example, the IceCube neutrino detector is buried 2.45km under the South Pole ice sheet to help shield against these down-going high energy particles. The average cosmic ray muon (4GeV) will easily penetrate through two detectors, depositing 2-3MeV per cm travelled in material with 1g/cm^3, like the scintillator.

2. Setup in coincidence mode

1. Connect two detectors together via an ethernet cable.
2. Power on one of the detectors (the other will be powered through the ethernet cable)
3. As they boot up, the first thing they do is check to see if another detector is also connected.
4. To indicate that they entered coincidence mode, they will both flash the LED light for 1 second.
5. Have a look at the screen, it should now read out the coincidence rate (MUONS) as well as the master rate (TOTAL). The coincidence rate corresponds to the muon trigger rate.

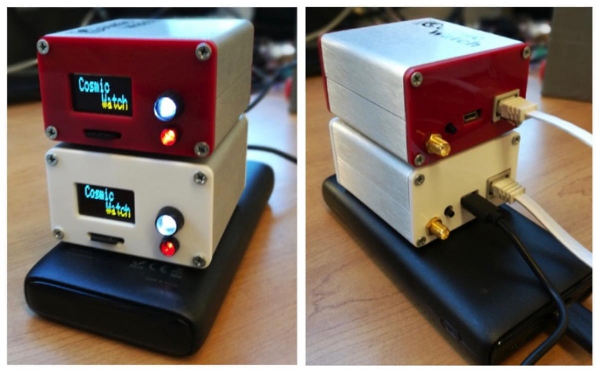


Figure 5 Setting up detectors in Coincidence mode.

You can also simply reset the detectors using the reset buttons, within ~1 second of each other to put them in coincidence mode.

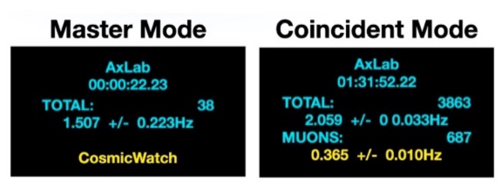


Figure 6 The OLED screen readout.

When the two scintillators are far away from each other, the rate is extremely low compared to when they are touching. Why is this [2pt]?

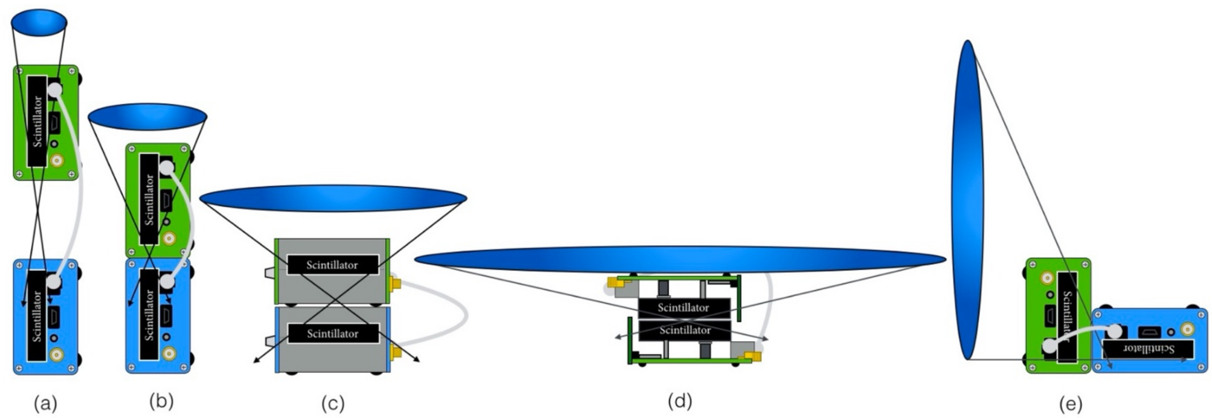


Figure 6 Different configurations for the detectors. Along with an illustration of the solid angle over which a muon could trigger both detectors.

The above figure shows several configurations for setting up the detectors in coincidence mode. Moving from left to right the solid angle subtended between detectors increases (illustrated as the blue ovals above the detectors) as well as the coincidence rate as measured by the coincident detector. The physical configuration of the two detectors allows us to look at different parts of the sky.

Place the detectors one-atop-the-other (Configuration (d) in Fig. 6) such that their scintillators are touching.

Measure the "full sky" muon rate [1pt]. Also record the Master rate [1pt].

Why do both detectors have the same muon rate [2pt]? Are these independent datasets?

How long would you have to take data to measure the full sky muon rate to within 10% statistical uncertainty [2pt].

We've put a lead enclosure in the front of the laboratory. Two inches of lead greatly decreases the incident terrestrial radiation but has little impact on the average cosmic ray muon.

Measure the master and coincidence rate inside the lead tower [2pt].

Wash your hands after working with the lead.

Coincidence mode also greatly reduces the terrestrial radiation. Terrestrial radiation (gamma, beta, alpha decays) tends to be lower energy (MeV) and is extremely unlikely to be able to deposit sufficient energy into both detectors. This is because they like to scatter after interacting, as well as the fact that they would have to deposit sufficient energy in both detectors to trigger them.

- **Alpha particles** will not penetrate a single detector (either the aluminum enclosure or even the black electrical tape) and therefore cannot trigger both the master and coincident detector at the same time.
- **Beta particles** can be significantly attenuated by the aluminum case, and have a significant chance of scattering, thus losing energy. It's unlikely that the beta particle will be able to deposit sufficient energy within the scintillator of the master, exit, then depositing sufficient energy in the coincidence detector.
- **Gamma rays** can penetrate the aluminum enclosure and plastic scintillator; however they have a significant chance of Compton scattering, which will change the direction. If a gamma ray does interact with both detectors, this means that it likely Compton scattered off the scintillator slabs, lost sufficient energy to trigger the detector, then Compton scattered or photoelectrically absorbed in the second detector, also depositing sufficient energy to trigger the detector.

Coincidence measurements, in particular for high count rates, can be significantly influenced by "**accidental coincidences.**" **Accidental coincidences** occur when two or more **unrelated** particles produce signals in multiple detectors within a short time window, leading to a **false coincidence event**. These coincidences happen due to random background radiation, or cosmic rays, rather than a true correlated physical process.

Let us assume that N_1 and N_2 are the individual master count rates of two detectors in a twofold coincidence arrangement and the coincidence window (the time window in which an event will be labeled as a coincident event) be τ. You measured N_1 and N_2 in the end of Part VI.
Given a time window, $\tau = 2\mu s$, and the count rates N_1 and N_2, what is the mean number of events you expect in the time window [2pt]?

If individual count rates follow Poisson statistics and that the two detectors are independent, the probability of observing n events, when the mean number of events is expected to be μ is described by the Poisson distribution:

$$f(n, \mu) = \frac{\mu^n e^{-\mu}}{n!}$$

The probability that Detector 2 gives no signal in the time interval τ is:

$$f(0, N_2\tau) = e^{-N_2\tau}$$

Correspondingly, the chance of getting an uncorrelated count in this window is 1 - $f(0, N_2\tau)$:

$$P = 1 - \mathrm{e}^{-N_2\tau}$$

Since normally $N_2\tau \ll 1$, we can Taylor expand:

$$P \approx N_2\tau$$

This is the probability of Detector 2 observing a count in time window τ.

The rate of which the time window τ is opened is N_1 (Detector 1 trigger rate). Therefore, the rate at which Detector 2 observes a count in coincidence with Detector 1 is:

$$R_2 = N_1 N_2 \tau$$

Finally, Detector 2 can also have a signal before Detector 1 within the resolving time of the coincidence circuit, the total accidental coincidence rate is:

$$R_{acc} = 2N_1 N_2 \tau$$

Pictorially, I've drawn below where that 2 in the above equation comes from.

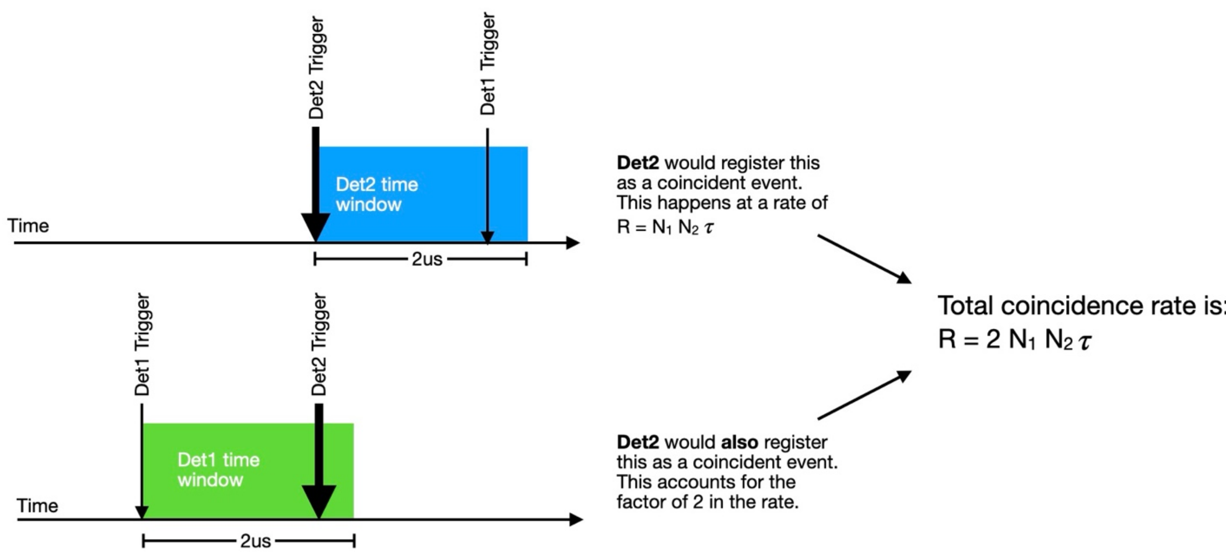


Figure 7 Illustration of the coincidence time windows generated after a detector is triggered.

Calculate the accidental coincidence rate given your measured master rates [2pt].

Compare that rate to the full sky coincidence rate you measured [2pt].

How many accidental coincidences do you expect per day [2pt]?

Part VII: Taking data

You've already become familiar with the OLED screen, but this is only useful in reading out the rates. What if you want event-by-event detail. There are two ways to access this data.

- **(Recommended) Through a microSD card**: Each time the detector is reset or powered on, a new file on the microSD card is created with a filename that counts sequentially upwards from the previous file. An "M" or "C" indicates whether the detector was in master mode (M), indicating that a coincidence detector was not used during startup, or coincident mode (C), indicating that a second detector was observed.
    - **Benefits**: extracting data using the microSD card is least prone to error and likely the simplest.
- **Directly to a computer serial port through a microUSB cable**: When the detector is plugged into a computer USB port, and the import_data.py is run using Python 3 (with appropriate libraries installed), the user is prompted to supply the path and name of the file to where the data is to be stored. It will then begin recording the data in real-time to the output file.
    - **Benefits**: is the serial port adds a computer time stamp to the events. This is useful if you need to keep accurate track of time.

- **In real-time via a serial monitor**: When a detector is plugged into the computer, the detector sends the data through the serial connection (USB micro cable with data transmission capabilities). If you have a serial monitor, you can read the data. We communicate at baud rate is 115200 bits per second. The simplest serial monitor is in VSCode or [ArduinoIDE](). The data can be seen accumulating in real-time in the serial monitor. The data can be copied and pasted into a text editor for later analysis.
    - **Benefits:** Realtime monitoring of data.

The data on the SD card or through the import_data.py script is always saved in a simple to use .TXT file. Comments are always indicated with the "#" marker for ease of removing, and each collum is tab delimited for easy data parsing. Each row is a different event, and each column represents some different property of the measurement. The figure below shows the first five events from a microSD card.

```
##########################################################################################
#                         CosmicWatch: The Desktop Muon Detector v2.1
#                                  Questions? saxani@udel.edu
#                                     Detector Name: AxLab
# Event   Time[s]  Coincident[bool]  ADC[0-4095]  SiPM[mV]  Deadtime[s]  Temp[C]  Pressure[Pa]
##########################################################################################
1       0.026389        0       145     6.7     0.000055        18.6    102165
2       0.191943        0       155     7.7     0.000471        18.6    102164
3       0.788561        0       143     6.5     0.000993        18.6    102164
4       1.676145        0       190     11.2    0.001363        18.6    102164
5       1.754871        0       281     19.9    0.001716        18.6    102164
```

- **ADC** [0-4095]: The ADC measurement for the event. The RP Pico has a 12-bit ADC, meaning the values reported are from 0-4095. The ADC is referenced between ground and 2.5V.
- **SiPM** [mV]: The calculated SiPM peak voltage based on the ADC measurement. It represents a number roughly proportional to the number of photons that triggered the SiPM. The conversion is from Fig. 2.
- **Dead-time** [s]: The cumulative dead-time since the detector start. This must be accounted for when making any rate measurement. To calculate the event rate, divide the total counts by the livetime, where the livetime = (total run time minus total deadtime).
- **Temp** [∘C]: The measured temperature of the detector via the on-board BMP280 temperature sensor. Measured in degrees Celsius.
- **Pressure** [Pa]: The measured temperature of the detector via the on-board BMP280
- pressure sensor. Measured in degrees Pascals.

Insert a microSD card. With the detectors in coincidence mode, take 5 minutes of data. Load the file from the SD card onto your computer and identify the coincident events [2pt].

Now, instead run the import_data.py script on your computer.

1. Launch VSCode through Anaconda Navigator

1. In VSCode, make a new terminal:
    - Terminal -> new terminal. Then use the terminal to launch import_data.py script:
      >> python import_data.py
2. You will be asked to select which COM port you want to record from.
3. Provide a file a name or press enter to save a default filename to the current directory

Take some data through import_data.py, and verify that indeed it produces a file with data [2pt].

Part VII: Plotting data

Finally, let's plot some actuall data. We've included some example data in the Data/ directory. The plot.py script takes a -i argument to import a file. If you supply the example data file ExampleData/AxLab_000.txt, a la:

>> python plot.py -i ExampleData/AxLab_000.txt

The master and coincidence rate will be printed to the terminal. This is calculated including the detector deadtime.

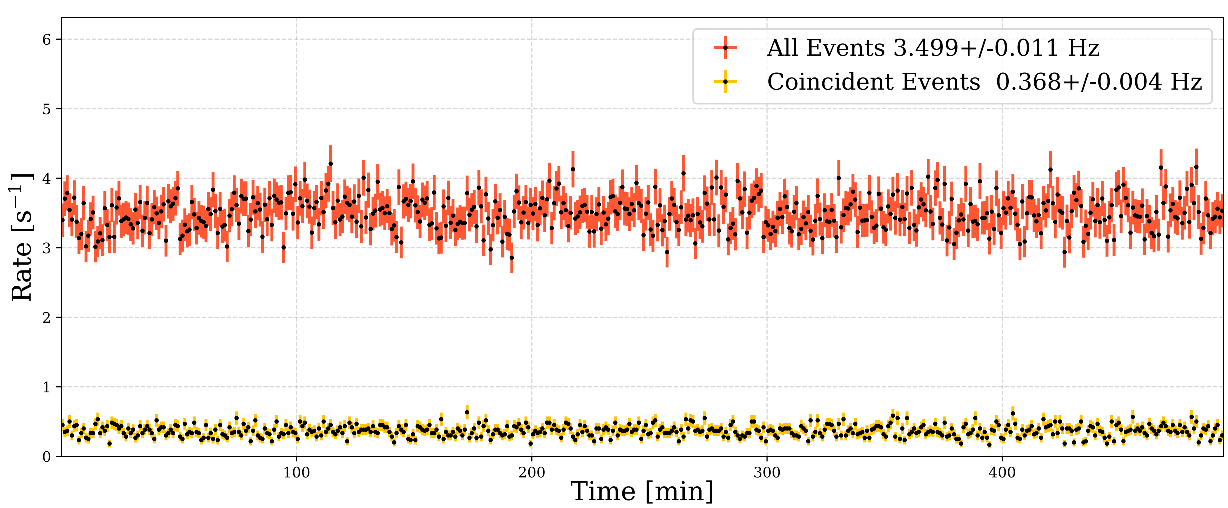
Additionally, four plots will be made.

1. **The count rate as a function of time**. In the plot.py script, where the file is uploaded:

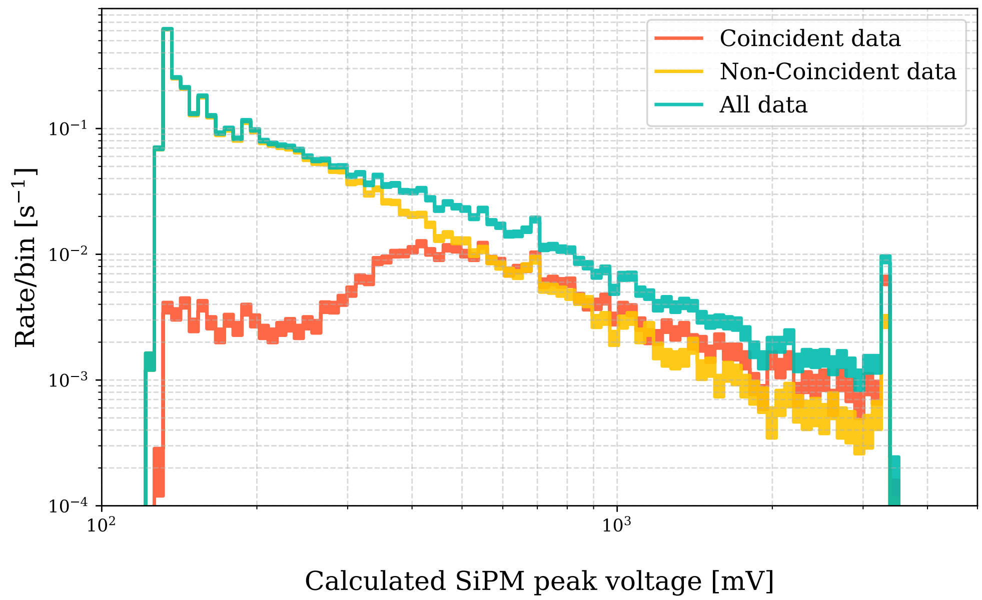
    f1 = CWClass(file_path, bin_size=60)

You can change the bin_size over which each datapoint in calculated. For example, in the following plot, each datapoint represents 60 seconds bin of runtime. The rate for each point is calculated as the number of events in that bin, divided by the livetime of the bin: (bin_size-deadtime in bin).

The error bars correspond to the statistical uncertainty. They are calculated as the sqrt(number of events in bin)/livetime.

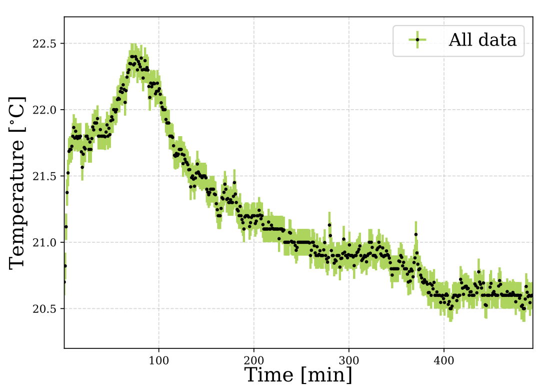


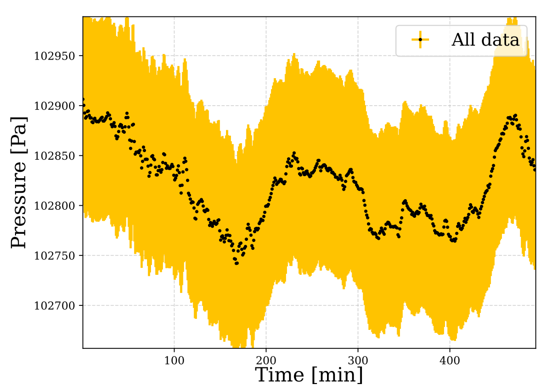
2. **A histogram of the SiPM Peak voltage.**



This is a cool plot. Larger pulses tend to be coincident events. Why is this [4pt]?

3. **The temperature and pressure, as measured by the BMP280**. The error bars come from the uncertainty quoted for the device, +/- 0.1C and +/- 100Pa, respectively.





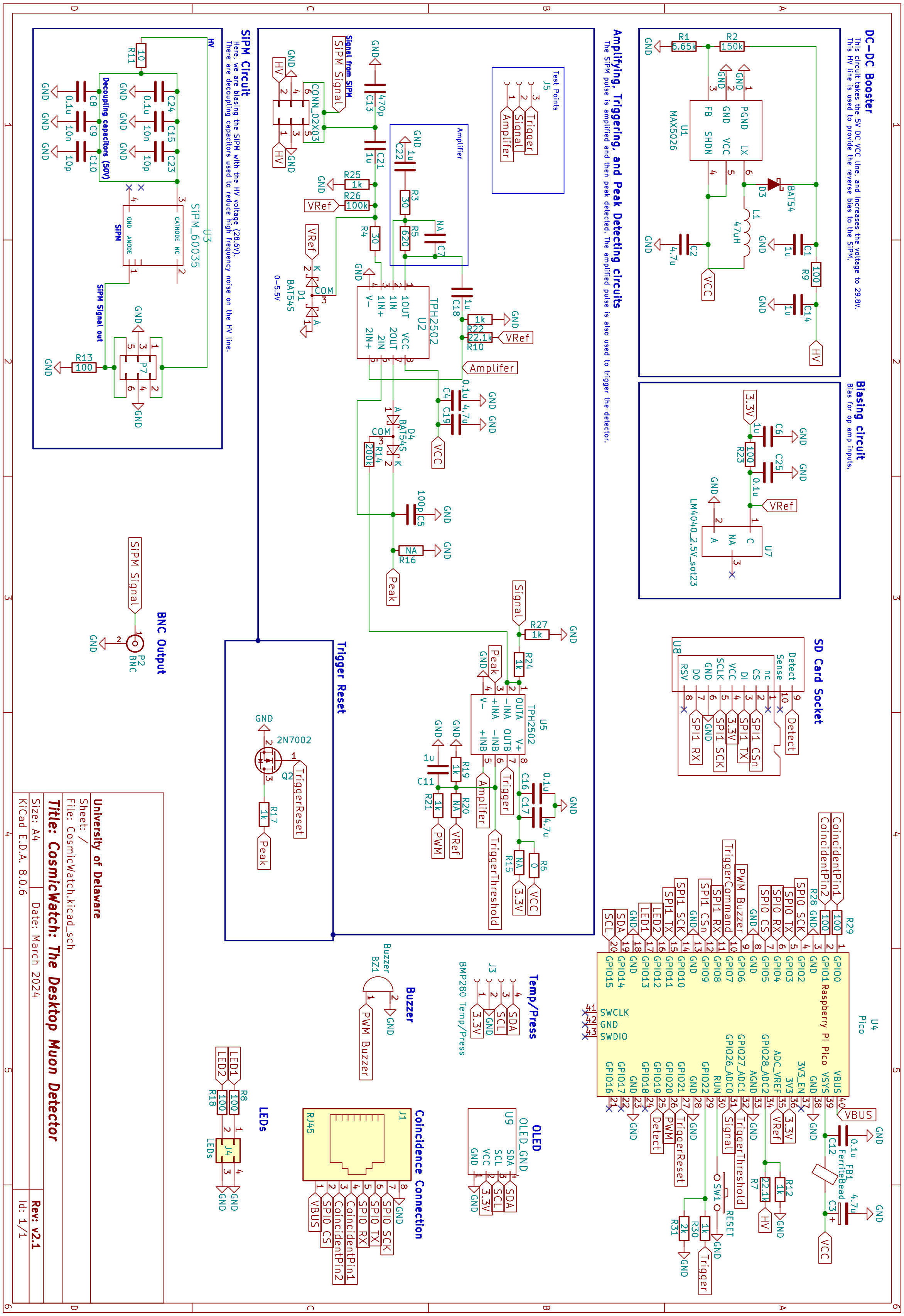
Create these same figures with your coincidence data from Part VII [8pt].

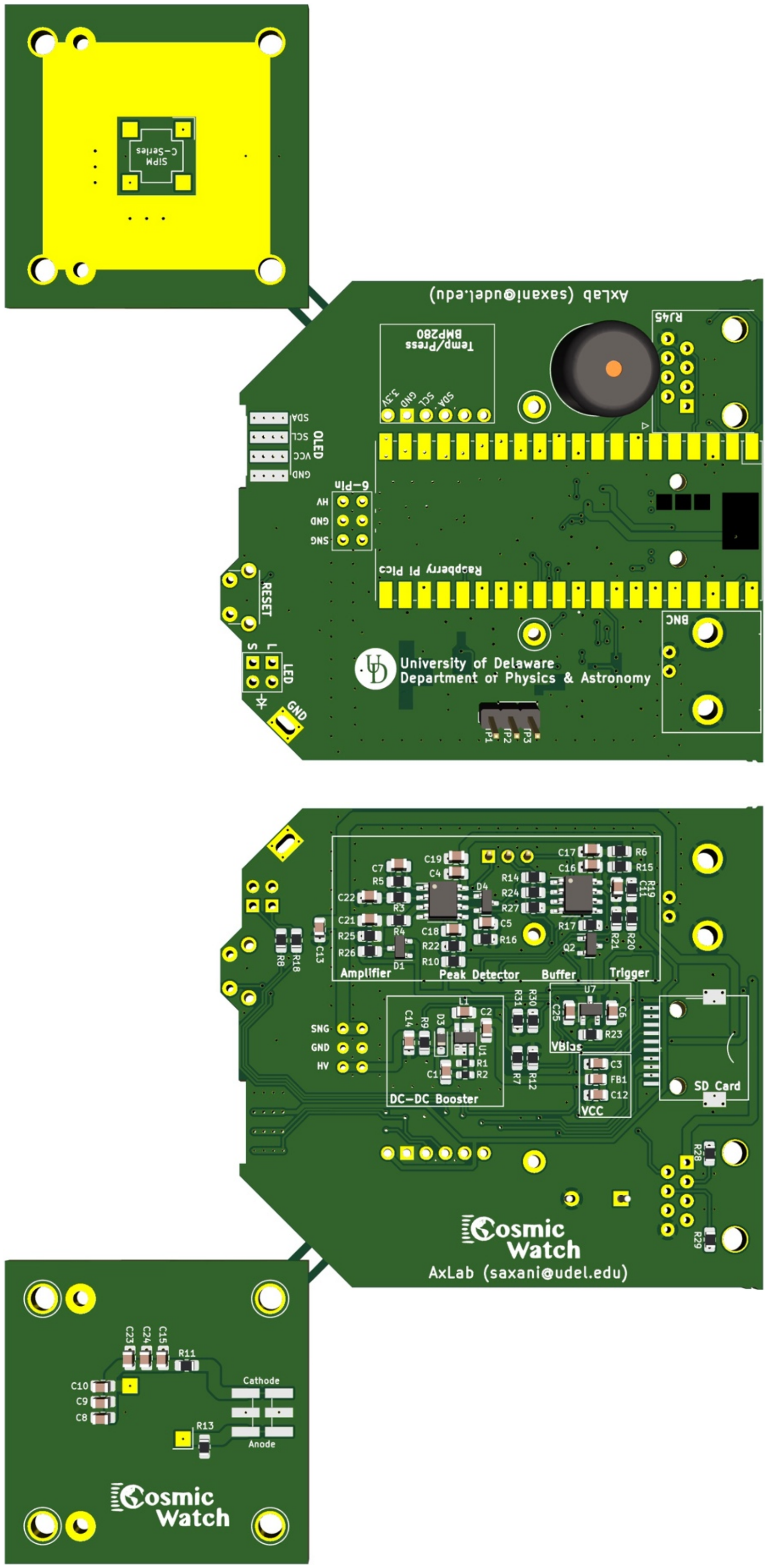
Feel free to modify plot.py with any changes you'd like to make.

Congratulations! You've reached the end of the Exercise!
You can insert your detector into a case if you like.

Next up is for you and your partner to design your own experiment. It's now time to transition over to the Instruction Manual, where there are a ton of different example measurements.

Discuss your idea with the instructor and come up with a plan. In a month, you will be presenting on your experiment/measurement in a conference-style presentation.


DC-DC Booster
Biasing circuit
SD Card Socket
Amplifying, Triggering, and Peak Detecting circuits
SiPM Circuit
Trigger Reset
BNC Output
Temp/Press
Buzzer
OLED
Coincidence Connection
LEDs
University of Delaware
Sheet: /
File: CosmicWatch.kicad_sch
Title: CosmicWatch: The Desktop Muon Detector
Size: A4
Date: March 2024
KiCad E.D.A. 8.0.6
Rev: v2.1
Id: 1/1


AxLab (saxani@udel.edu)
Temp/Press
BMP280
RJ45
OLED
6-Pin
Raspberry Pi Pico
RESET
LED
GND
BNC
University of Delaware
Department of Physics & Astronomy
Amplifier
Peak Detector
Buffer
Trigger
DC-DC Booster
VBias
VCC
SD Card
Cosmic Watch
AxLab (saxani@udel.edu)
Cathode
Anode
Cosmic Watch