Cosmic Ray Muon Detection

Elise Le Boulicaut
Gustavus Adolphus College

May 29, 2018

Abstract

Muons result from cosmic ray showers, which are streams of energetic charged particles originating from our galaxy and passing through the atmosphere. Muons can be better understood by studying their count rate and energy as a function of altitude. This provides information about where they are created as well as how they interact with the atmosphere. Preliminary muon detection tests were conducted using large plastic scintillator paddles and photomultiplier tubes with Nuclear Instrumentation Modules. To make the experiment more portable, a compact muon detector consisting of a slab of plastic scintillator with a silicon photomultiplier was constructed and placed in a high-altitude balloon. Preparatory work such as simulation, threshold testing, and calibration are presented here. Results from the balloon flight were also analyzed, despite unfortunate circumstances which limited the amount of data available. In addition, a coincidence circuit has been developed for future flights and is discussed.

1 Introduction

Cosmic rays are energetic charged particles passing through Earth's atmosphere, originating from our galaxy and thought to be produced by events such as supernovae or black holes [1]. They are composed primarily of protons but also of helium nuclei as well as a small portion of heavy nuclei [9]. When these cosmic rays interact with the atmosphere, they produce showers of new particles, which can decay in many different ways. In particular, pions and kaons produced in this manner decay to muons at an altitude of about 15,000 meters [12]. These muons then travel in conical showers within an angle of 1° of the original trajectory of the particles which produced them [12]. Muons have a lifetime of $2.2\,\mu s$, after which they decay into an electron, a neutrino and an antineutrino. Without taking into account relativity, a typical muon would only travel approximately 660 m before decaying. However, because of time dilation in the rest frame of the muon, as well as its relatively weak interaction with other particles, it actually travels much farther and can be detected at Earth's surface. Since most other particles are completely stopped when passing through the atmosphere, cosmic ray showers observed at sea level are almost exclusively composed of muons [9, 12]. The muon flux at sea level is known to be approximately 1 per minute per square centimeter per steradian [1, 12].

1.1 Cosmic ray showers and the Pfotzer maximum

When a primary cosmic ray particle (often a proton) collides with an air molecule, it produces secondary cosmic rays which can be divided into three components: nuclear-active, soft, and penetrating [5], as seen in Fig. 1. The nuclear-active component results from the disintegration of nuclei and the formation of unstable particles in the process, such as pions. These interactions occur in cascades, resulting in the original energy being divided into a large number of particles. Thus, individual particles rapidly lose energy as they travel through the atmosphere and few nuclear-active components remain at sea level. The soft component is composed of electrons and photons from the almost instantaneous decay of neutral pions produced by the nuclear processes. These soft components are quickly absorbed and do not reach the ground. Finally, the penetrating component is comprised primarily of muons, which result from the decay of charged pions. Because these muons are very energetic and are not nuclear-active, they interact weekly and reach the ground easily.

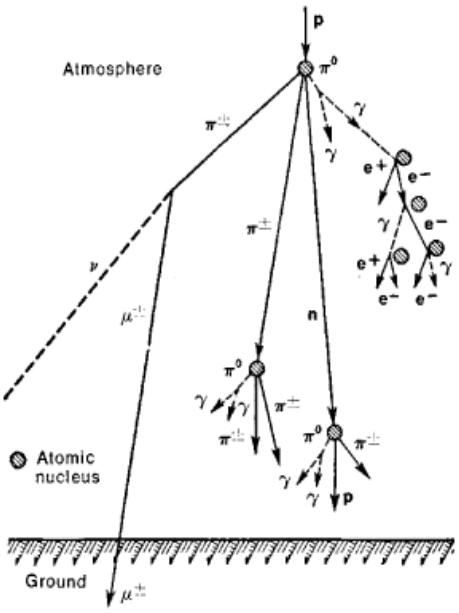

Figure 1: Diagram of cosmic ray showers, showing the various decays that occur upon the interaction of a cosmic ray with the atmosphere. Figure obtained from reference [5].

The altitude at which the production of new secondary cosmic rays balances out with their absorption leads to a maximum in count rate, called the Pfotzer maximum. Fig. 2 shows the data from the first discovery of this phenomenon by Regener and Pfotzer in 1935, using a triple coincidence with Geiger-Müller tubes in a high-altitude balloon [4]. The count rates recorded in this experiment include all ionizing radiation, and therefore all cosmic rays. However, because muons are a prominent part of secondary cosmic rays, we can assume that the height of the Pfotzer maximum corresponds approximately to the mean altitude at which they are created.

From Fig. 2, it can be noted that the count rate appears to decrease exponentially as the altitude decreases and particles are progressively absorbed. This is reminiscent of Beer's law, which states that the absorptive capacity of a dissolved substance is proportional to

its concentration in a solution [2]. In addition, it can be seen in Fig. 2 that there seems to be different stages of exponential decay with different extinction coefficients, perhaps corresponding to the absorption of the different types of secondary cosmic rays.

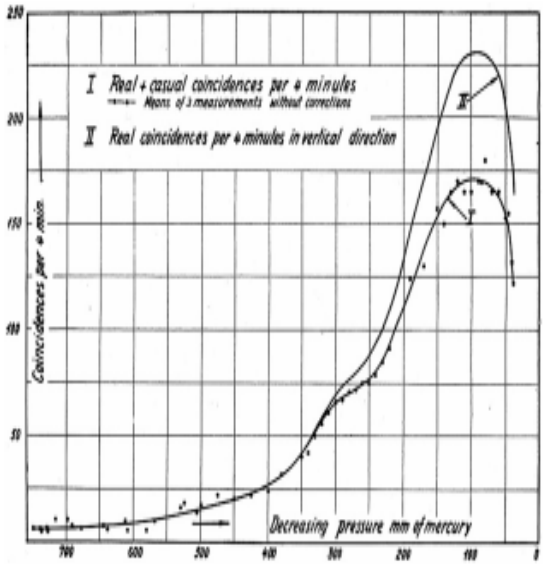

Figure 2: Coincidence rate per 4 min at a solid angle of 20 degrees about the zenith as function of decreasing pressure, obtained by Regener and Pfotzer in 1935. Figure obtained from reference [4].

1.2 Energy loss of muons through the atmosphere

Cosmic ray muons are thought to have an average energy of approximately 6 GeV upon creation and 4 GeV at sea level [12]. The energy loss between these two points is mainly due to interactions with air molecules in the atmosphere. This implies that the peak of a distribution of muon energies would shift towards lower energies with decreasing altitude.

To study the mean rate of energy loss for muons passing through any material, it is appropriate to use the Bethe equation , whose curve is shown in Fig. 3 for different elements [3]. With energies between 4 and 6 GeV, cosmic ray muons are called minimum-ionizing particles because their rate of energy loss is close to the minimum. Applying the Bethe equation to air [7] leads to:

$$-\frac{dE}{dx} = \left(\frac{0.1535}{\beta^2}\right)\left[9.358 + \left(ln\beta^2 + ln\gamma^2 - \beta^2\right)\right] \times MeVg^{-1}cm^2, \tag{1}$$

where $\beta = v/c$, v being the speed of the particle and c the speed of light; $\gamma = \sqrt{\frac{1}{1-\beta^2}}$ is the Lorenz factor. This equation can be used to simulate the energy of a single muon as a function of altitude, given an initial energy and height (see section 2.1).

From Fig. 3, it can be seen that for minimum ionizing particles such as cosmic ray muons, the rate of energy loss decreases with decreasing momentum (and thus energy).

Therefore, we would expect a tendency for muon energies to converge (in a broad sense) and thus their energy distribution to narrow as the particles near the ground. This was verified experimentally by McNichols by taking measurements with a plastic scintillator detector at different points on a mountain in Hawai'i. [11].

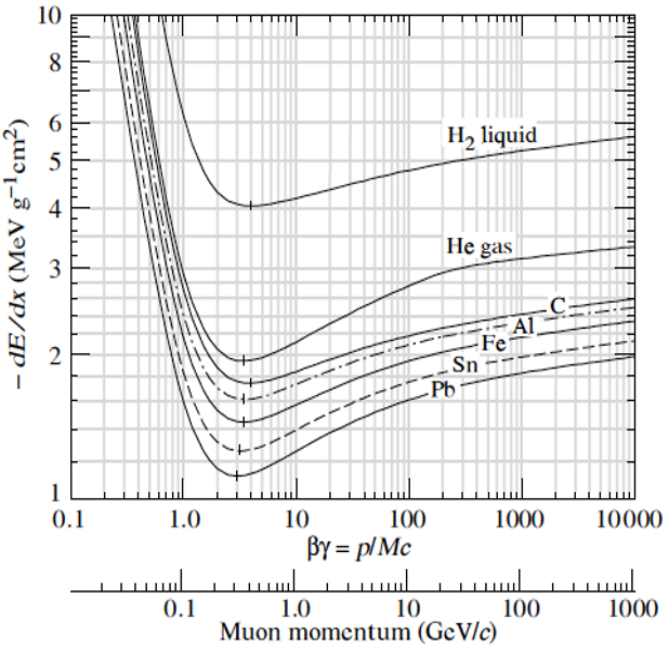

Figure 3: Mean energy loss rate for muons in different materials. Most cosmic ray muons have a momentum around the minimum of the curve, making them minimum ionizing particles. Figure obtained from reference [3]

1.3 Methods of cosmic ray detection

One common way to detect secondary cosmic rays is to use a Geiger-Müller tube. The tube is filled with gas and a voltage is applied to it. When a particle or photon passes through, the gas is ionized and a discharge occurs. This creates a pulse, which can then be counted. Geiger counters record all types of ionizing radiation indiscriminately.

In order to specifically detect muons, it is necessary to impose farther restrictions. Typically, this is carried out by using plastic scintillator detectors, which produce a photon when a charged particle passes through the material. A photomultiplier then produces an electrical signal upon the detection of the photon, where the amplitude of the pulse is proportional to the amount of energy deposited in the detector. Traditionally, this has been done using scintillator detectors in the shape of paddles or cylinders, along with photomultiplier tubes (PMTs). These require a high-voltage supply and the use of Nuclear Instrumentation Module (NIM) electronics. This method can produce satisfactory results [9] but it is inconvenient for altitude measurements. Indeed, the only way to investigate an altitude dependence on count rate and energy with this setup is to take data at different elevations on a mountain.

This produces relatively few data points and cannot reach altitudes high enough to observe the Pfotzer maximum. An alternate approach consists in using a much smaller slab of plastic scintillator, fitting in the palm of a hand, along with a Silicon Photomultiplier (SiPM), which only requires approximately 30V to be powered. Surface-mount electronics make the entire experiment extremely compact, allowing it to be placed in a high-altitude balloon. This allows for an almost continuous data acquisition, as well as a much wider range of altitudes.

Both methods of muon detection were studied for this project and compared to results from a simple Geiger counter. Plastic scintillator paddles and PMTs were used as preliminary work in order to get a better understanding of charged particle detection and characterization. Then, the compact muon detection system was built, calibrated, and sent up in a high-altitude balloon. Unfortunately, the recovered results were sparse, but the ground tests made before the flight make us confident that more satisfactory data can be obtained with this system in the future. A coincidence circuit was also designed as an improvement of this setup and could be used in the next balloon flight. In addition, a computer simulation was created for the energy of muons as a function of altitude and could be compared to experimental results.

2 Simulation of muon energy loss through the atmosphere

2.1 Simulation for a single muon

As a first step in simulating the passage of cosmic ray particles through the atmosphere, the Bethe equation can be used to obtain a plot of energy as a function of altitude for a single muon. This requires an assumption on the initial energy and height of the muon upon creation. These parameters can be adjusted to determine their effect on the shape of the plot and the resulting energy of the muon once it arrives at sea level. The Matlab code used for this is found in Fig. 26 in the Appendix. The method is reminiscent of Euler's method. From the initial energy, the β factor of the particle is calculated, using the relativistic energy expression

$$E_{total} = \gamma E_{rest}, \tag{2}$$

where the rest energy of the muon was taken to be $E_{rest} = 105.6583745$ MeV. The calculated β value is used to determine how far the muon travels (dh) in a fixed small time interval (dt) from

$$dh = \gamma\beta c dt. \tag{3}$$

This distance increment must first be converted to g/cm^2 to be used in the Bethe equation. This is done using

$$dx = dh \times \rho_{air}(h), \tag{4}$$

where dx now has the appropriate units, and $\rho_{air}(h)$ is the density of air depending on altitude h. It is determined using the law of atmospheres, expressed in the following equations [6]:

$$P(h) = P_0 \left(\frac{1 - Lh}{T_0}\right)^{\frac{gM}{RL}} \tag{5}$$

$$T(h) = T_0 - (Lh) \tag{6}$$

$$\rho_{air}(h) = \frac{MP(h)}{RT(h)}, \tag{7}$$

where P represents pressure, and T temperature. The constants used in these equations are the following:

$P_0 = 101325$ Pa (standard pressure at sea level)
$T_0 = 288.15$ K (standard temperature at sea level)
$L = 0.0065$ K/m (temperature lapse rate)
$g = 8.9$ m/s^2 (gravitational acceleration)
$M = 0.0289644$ kg/mol (molar mass of dry air)
$R = 8.31447$ J/(mol·K) (ideal gas constant)

Once dx is obtained from equation 4, the Bethe equation (equation 1) can be used to determine the energy loss for that distance increment. The energy and velocity of the particle are then updated using equation 2 and used in the next iteration. The loop is terminated when the muon reaches sea level. An example of a plot of energy versus altitude is shown in Fig. 4 for an initial energy of 6,000 MeV and initial altitude of 15,000 m. This results in a final energy of 2,848.6545 MeV at sea level, which roughly agrees with what we would expect [12].

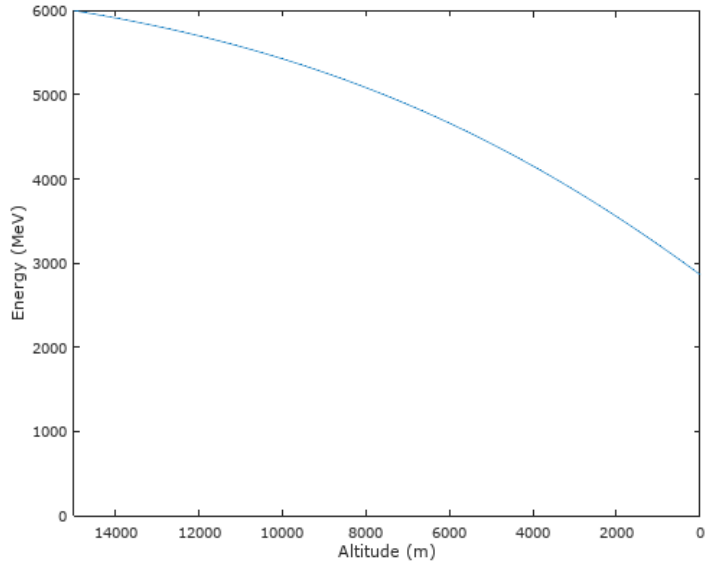

Figure 4: Plot of energy as a function of altitude for a single muon passing through the atmosphere using the simulation discussed in section 2.1. The initial conditions used were an energy of 6 GeV and altitude of 15,000 m. The resulting energy at sea level is 2,848.6545 MeV.

2.2 Simulation for multiple muons

The simulation described in the previous part is insightful but not realistic, because muons are actually created with a range of energies and at a range of altitudes. Therefore,

the single muon simulation was run multiple times, for a set of random Gaussian distributed values for the initial energy and height. The code used for this is found in Fig. 27 in the Appendix. Now, instead of choosing a single value for each initial condition, we choose a mean and standard deviation. All the solutions are plotted on the same graph, to visualize the range of energies and altitudes of the resulting muons, as shown in Fig. 5. This plot was obtained for 100 randomly generated muons having a mean intial energy of 6 GeV, with a standard deviation of 1 GeV, created at a mean altitude of 15,000 m, with a standard deviation of 2,000 m. This is a more accurate representation what occurs in reality, even though the number of muons passing though the atmosphere at a given time would of course be much greater than 100. It is interesting to note from Fig. 5 that the range of energies appears to be slightly narrower at lower altitudes, as expected from the Bethe equation.

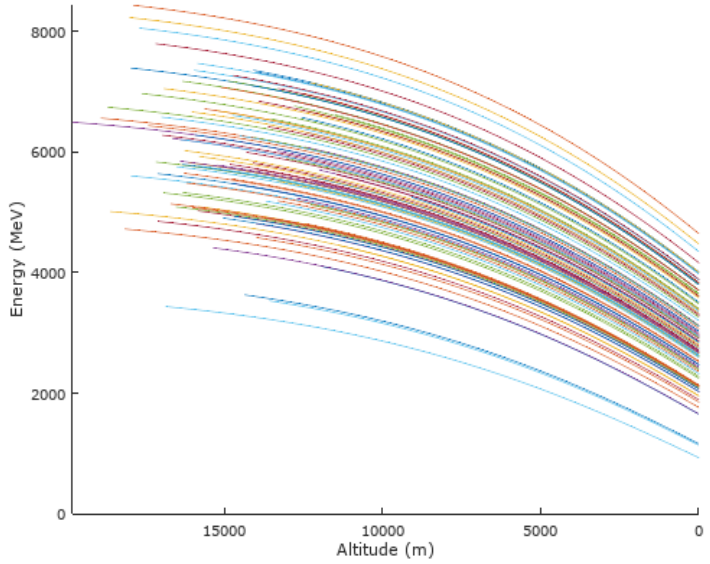

Figure 5: Plot of energy as a function of altitude for 100 randomly generated muons passing through the atmosphere using the simulation discussed in section 2.1. The initial conditions used were a mean energy of 6 GeV, with a standard deviation of 1 GeV, and an altitude of 15 km, with a standard deviation of 2 km.

2.3 Energy distributions at various altitudes

In order to study in more detail the dependence of the range of energies on altitude, the simulation in section 2.2 was used to obtain histograms of the energies of the randomly generated muons at various altitudes. This can be seen as taking a vertical slice through a plot such as the one in Fig. 5. The code for this is found in Fig. 28 in the Appendix. Each time the simulation is run for a single muon (with a random initial energy and altitude), the energy value corresponding to the altitude closest to the one we wish to obtain the histogram for is selected and stored. The built-in Matlab "histogram" function is then used with a fixed bin width of 500 MeV to obtain the distribution. Examples of these are shown

in Fig. 6. They were obtained from a set of 500 randomly generated muons with the same initial conditions as in Fig. 5. We can see from this figure that as the muons fall through the atmosphere and loose energy, their energies remain Gaussianly distributed, but the peak and width of the distribution changes. In order to quantify these changes, Matlab was used to obtain the best fit mean and standard deviation for the histograms at each altitude. The results are found in Table 1.

Altitude (m)	Mean energy (MeV)	Standard deviation in energy (MeV)
15,000	5,906	963
10,000	5,402	947
5,000	4,401	883
0	2,859	758

Table 1: Best fit mean and standard deviation in energy for the histograms found in Fig. 6.

The first element to note from Table 1 is that at the mean creation altitude (15,000 m), the mean and standard deviation in energy are very close to the initial conditions (mean of 6,000 MeV and standard deviation of 1,000 MeV), as expected. As the altitude decreases, the peak of the distribution shifts to lower energies, consistent with what was observed in Fig. 5. Also, Table 1 shows that the standard deviation in energy decreases with decreasing altitude, indicating a narrowing of the distributions, as can also be seen in Fig. 5 and 6. This confirms the predictions made in section 1.2 based on the Bethe equation. The fact that the narrowing effect was not very large could be due to the initial conditions, particularly the standard deviation in the initial energy.

Constructing these simulations helps us to better understand the process of energy loss of muons as they fall through the atmosphere and provides us with a model to which experimental results can be compared.

Some shortcomings of these simulations include our uncertainty in the most realistic values of the initial energy and altitude as well as their standard deviations. Also, it was assumed that the energies and altitudes of creation followed a Gaussian distribution, which is not necessarily the case due to the complexity of the creation process. All these parameters would affect the distributions shown in Fig. 6.

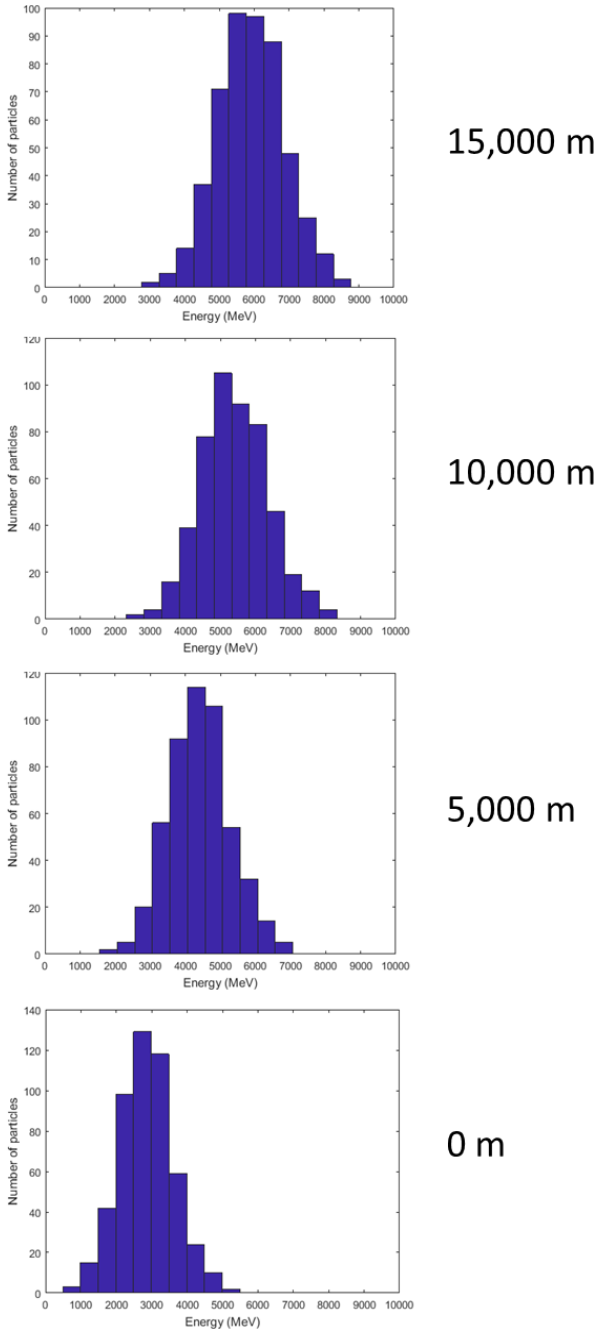

Figure 6: Histograms of energies of 500 randomly generated muons at various altitudes, using the method described in section 2.3. The initial conditions used were a mean energy of 6 GeV, with a standard deviation of 1 GeV, and an altitude of 15,000 m, with a standard deviation of 2,000 m.

3 Cosmic muon detection using plastic scintillator paddles, photomultiplier tubes and NIM electronics

3.1 Setup

This first method was used as an introduction to plastic scintillator detectors in order to gain a full understanding of how muon detection works in general. The setup for this method is shown in Fig. 7. The paddles are slabs of plastic scintillator of dimensions 8" × 8", two of which have a thickness of 1/2" and the third a thickness of 1/4". The photons resulting from the interaction of the incident charged particle with the plastic are directed to the PMT through a light guide. Optical gel was used at the junction between the light guide and the PMT to improve the optical connection by matching indices of refraction, and black tape was used on both devices to block out light from the surroundings. The role of the PMT is to convert the collected light into an electrical signal in the form of a current pulse. These PMTs must be powered with a negative high voltage supply and the signal can be read from the anode. In this case, two voltage supplies were used so that two of the paddles shared a supply. The three paddles and PMTs used in our experiment were obtained from TRIUMPH approximately 20 years ago. When viewed on an oscilloscope, the signal from the anode of the PM tube is a negative-going pulse with a width of approximately 20 ns and an amplitude varying from several hundred millivolts to several volts, depending on how much energy was deposited by the incident particle.

In order to obtain a fast logic signal from the raw PMT pulses, a Constant Fraction discriminator (CFd) was used. Our model was a Phillips Scientific Five Channel Timing Discriminator Model 715. The role of the CFd is to output a negative pulse of desired width (set to 10 ns for most of our experiments) and constant amplitude (800 mV) for every input pulse which exceeds a certain threshold. Threshold characterizations will be discussed later.

The three CFd outputs are then sent to a LeCroy Triple 4-Fold Coincidence Unit Model 465. Push buttons can be used to enable or disable any number of inputs, which allowed us to use the unit either as a buffer, a coincidence between a pair of detectors or a coincidence between all three detectors. The output pulse is also a fast logic signal, with a width equal to the overlap of the coincident input pulses [10].

Finally, the output of the coincidence unit is sent to a Canberra Dual Counter Timer Model 2071A, which displays on a screen the number of fast logic pulses received in a given amount of time.

3.2 Supply voltage characterization

The first step in characterizing the setup is to find an ideal voltage at which to operate the PMTs, as this can greatly affect the amplitude of the pulses and therefore the count rate. Two criteria are important in making this decision: a small change in operating voltage must not lead to a significant change in pulse amplitude and the pulses from all three PMTs should have the same height. Because we were using two supplies (Hamamatsu Model C3350 dual supply and Canberra Model 3002 single supply) for three tubes, our goal was to find a combination of voltages and PMTs that would best satisfy both criteria. To do this, the anode pulse for each detector was measured using an oscilloscope, while varying

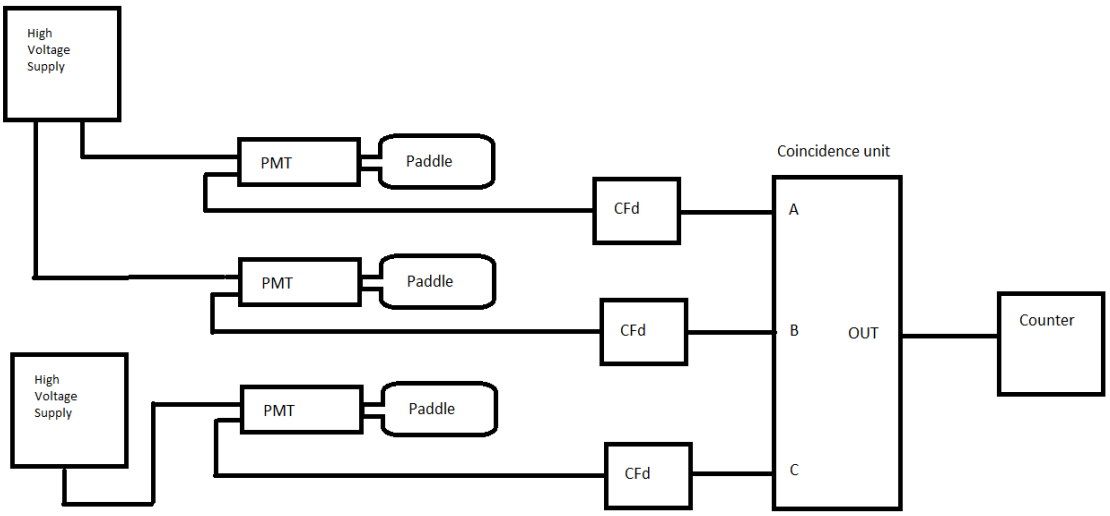

Figure 7: Setup used for detecting muons using three large plastic scintillator paddle detectors, photomultiplier tubes (PMT), and NIM electronics. The PMTs are powered with high voltage power supplies and their output is sent to a CFd (Constant Fraction discriminator) to obtain a fast logic signal. A coincidence unit is then used, for which any of the inputs (A,B or C) can be enabled or disabled. A Dual Counter Timer then displays the number of registered pulses within a given time.

supply voltages. Five Na-22 button sources were placed on each paddle when making these measurements in order to obtain a large influx of mono-energetic particles. The result is shown in Fig. 8, where the numbers 28, 85 and 88 refer to the serial numbers printed on the PMTs and are used for identification.

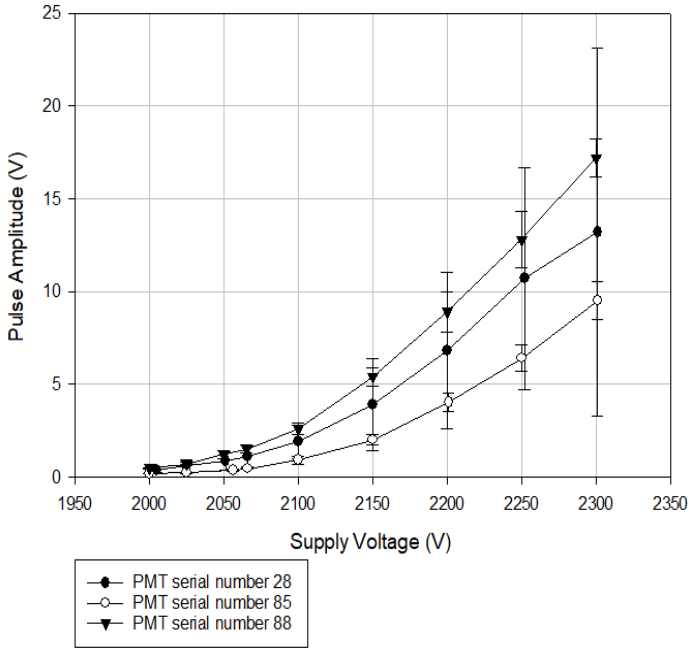

Figure 8: Supply voltage characterization: amplitude of pulse at the anode of each PMT as a function of supply voltage. Pulse amplitudes and uncertainties were measured using the cursors on a Tektronix 2212 with five Na-22 button source placed on each paddle. The numbers 28, 85 and 88 refer to the serial number on each PMT and are used for identification. A Hamamatsu C3350 dual supply was used for tubes 85 and 28 and a Canberra 3002 single supply was used for tube 88.

First of all, it is evident from this figure that the three PMTs used do not function identically because the pulse amplitudes vary from one tube to the other for a given supply voltage. However, all three curves do have similar shapes, indicating comparable responses to changes in supply voltage. Fig. 8 shows us that we should avoid operating the tubes at more than 2,150 V, as pulse amplitudes vary greatly with small changes in supply voltage in this region. In addition, we found that the ideal arrangement for having pulses of equal amplitudes was to use the Hamamatsu supply at 2,040 V for tubes 28 and 88 and the Canberra supply at 2,105 V for tube 85. This resulted in pulses of amplitude 0.70 ± 0.10 V, 0.90 ± 0.10 V, and 0.80 ± 0.10 V respectively. We considered this sufficient to proceed.

3.3 CFd threshold characterization

The next goal was to characterize the count rate response to different CFd thresholds. Indeed, this threshold determines which raw detector pulses will produce logic pulses when passed through the CFd and therefore how many events will be registered by the counter. If

the threshold is too low, random fluctuations and electronic noise will be counted as muon events; but if it is too high, a large portion of muons will be ignored. In order to determine a reasonable setting, we recorded counts in one minute for varying CFd thresholds, from -25 mV to -1 V for each detector. Note that since the pulses are negative-going, a threshold is considered "low" if it is close to 0. Radioactive sources were removed so that cosmic rays were the only source of charged particles detected. The result is shown in Fig.9.

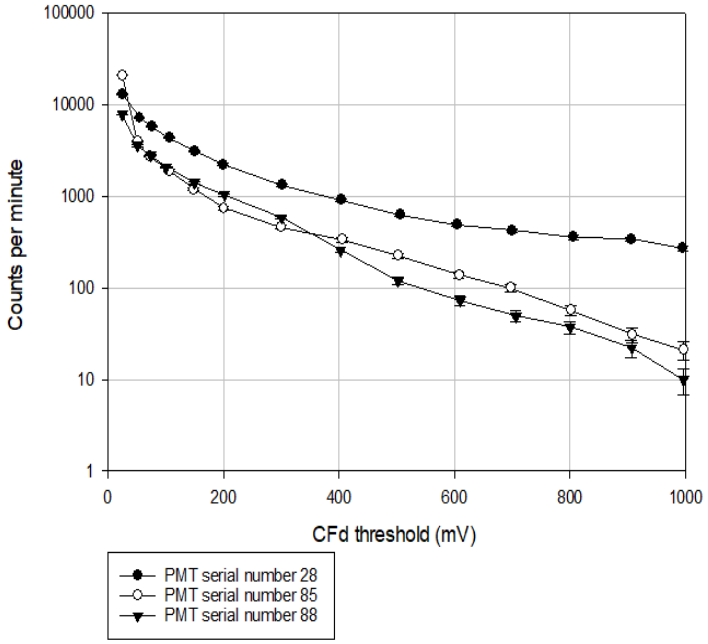

Figure 9: CFd threshold characterization: count rate for each detector as a function of the absolute value of the CFd threshold. Error bars correspond to the square root of the number of counts. A log scale is used for the y-axis to accommodate large differences in count rate. These results were obtain without any radioactive sources and with tubes 28 and 85 sharing the Hamamatsu supply at 2,066 V and tube 28 supplied by the Canberra at 2,025 V. Note that these are not ideal operating conditions for comparing detectors but were considered sufficient for this purpose.

Unfortunately, this test was done before the voltage characterization and thus the operating conditions were not optimal for matching pulse amplitudes. In this case, tubes 28 and 85 were supplied by the Hamamatsu at 2,066 V and tube 88 by the Canberra at 2,025 V. This resulted in pulse amplitudes of 1.30 ± 0.10 V, 0.60 ± 0.10 V and 0.80 ± 0.10 V respectively. This explains why the count rates for PMT 28 are much greater than the other two. However, since we are mostly interested in the shape of the graph, we will consider this result sufficient for our purposes. We can see from Fig. 9 that count rates vary wildly depending on the threshold, which is why a logarithmic scale is used for the y-axis. The ideal threshold is situated in a region passed the "knee" of the curve, but not too far from it. This is the point at which noise no longer dominates, without cutting off too many real events. From our graph, we determined that a good threshold would be 200 mV. Since ideally the amplitude of the pulses would be matched (even though this was not quite the

case in Fig. 9), the same threshold of 200 mV was subsequently used for all three detectors. When conducting a similar experiment, McNichols argued that setting the discriminator to a value of 30% of an average pulse height ensures the detection of almost exclusively muons but also ignores a large portion of them [11]. With our pulse amplitudes ideally matched as described in section 3.2, a discriminator of 200 mV would correspond to 29% of the average pulse amplitude for tube 28, 22% for tube 88 and 25% for tube 85. These percentages being right below the 30% mark farther supports our decision.

3.4 Coincidence measurements

Once the ideal supply voltage and CFd threshold conditions were established, count rates were recorded for single detectors, pairs of detectors using a double coincidence, and all three detectors using a triple coincidence. The supply voltage and pulse amplitudes were the optimal conditions described in section 3.2 and the CFd threshold was set to -200 mV for all three channels of the CFd. Counts were recorded for a duration of 8 minutes. Coincidences were recorded with the paddles stacked vertically, separated by 14cm. Table 2 summarizes these results. Double coincidence counts were differentiated between adjacent paddles (top/middle or middle/bottom) and non-adjacent paddles (top/bottom) as we would expect the change in distance to have an effect on the count rate. In addition, accidentals were measured by recording coincidence counts when all three paddles were lying flat next to each other. Again, for double coincidences, count rates for adjacent and non-adjacent paddles were separated because of the difference in distance between them. For each configuration, multiple runs were conducted with different paddles at different positions relative to each other. The results were averaged to obtain the rates shown in Table 2 and the uncertainty was taken to be the standard deviation of these data sets. The dimensions of the paddles were used to obtain the flux in counts per minute per cm^2, in order to compare with the accepted value of the cosmic ray muon flux of 1/min/cm^2.

Type of measurement	Description	Counts per minute per cm^2
Single detector	PMT 85	5.029 ± 0.012
	PMT 28	3.367 ± 0.008
	PMT 88	3.241 ± 0.008
Double coincidence	stacked adjacent	0.4226 ± 0.0010
	stacked non-adjacent	0.2051 ± 0.0005
	flat adjacent	0.01184 ± 0.00003
	flat non-adjacent	0.005268 ± 0.000013
Triple coincidence	stacked	0.1897 ± 0.0005
	flat	0.0006055 ± 0.0000015

Table 2: Count rates for different configurations showing the effect of coincidence. Counts were recorded over 8 minutes, using the optimal supply voltage and threshold conditions described in sections 3.2 and 3.3. Each rate is taken as the average of multiple trials and the uncertainty corresponds to the standard deviation of the mean.

The single detector count rates show that one of the PMTs (85) recorded many more counts than the other two, which could be due to the fact that it was powered with a

different voltage supply. While performing these experiments, it was also found that single detector rates varied greatly depending on the positioning of the paddle (for example if it was placed on the ground or propped up some distance off the floor). This demonstrated the necessity for coincidence measurements. Indeed, the flux obtained from double coincidence for stacked adjacent paddles was relatively close to the expected muon flux and was also found to be consistent, independent of which paddles were used. The count rate for stacked non-adjacent paddles was slightly lower, most likely because of the greater distance between them. However, it is surprising that this effect should be so pronounced because cosmic ray muons are expected to be so energetic that a very large portion of those incident at the top paddle should survive to reach the bottom one. Furthermore, the time necessary for them to travel from the middle to the bottom paddle should be on the order of 0.5 ns, which should not considerably effect the overlap of the two pulses. Nevertheless, it is clear that cosmic rays are indeed being measured because the coincidence rates for paddles lying flat next to each other are very small, as they should be if particles are incident from the vertical. Finally, it was found that a triple concidence yielded similar results to a double coincidence for non-adjacent paddles. This is reasonable, since it is expected that if a particle passes through the top and bottom paddles, it also passes through the middle paddle. However, this flux measurement is almost an order of magnitude lower than the expected muon flux, which suggests that perhaps the restrictions imposed by the triple coincidence are too strict. Again, the very low rate of accidentals in this case confirms that the measured particles are indeed traveling vertically.

3.5 Effect of lead shielding on count rate

Our goal for this part of the experiment was to investigate the stopping effect of various thicknesses of lead on incident cosmic ray muons. Since it was determined in section 3.4 that coincidences were the most reliable way to make count rate measurements, it was decided to record triple coincidence counts over a period of 3 hours with the paddles stacked vertically, placed under varying numbers of lead bricks. A maximum of five lead bricks were used, each 2" thick, leading to a range of thicknesses from 0 to 10". Note that supply voltage and threshold conditions were not optimized when taking these measurements. PMTs 88 and 28 were supplied with 2,066 V from the Hamamatsu supply, PMT 85 with 2,000 V from a Canberra 3102, and all CFd thresholds were set to -50 mV. Even though these were not ideal settings for matching pulse amplitudes and cutting out noise, it was found that triple coincidence count rates were still comparable to the ones obtained in section 3.4. Results are shown in Fig.10.

We can see from this figure that within the range of thicknesses used in this experiment, the shielding had no observable effect on count rate. Indeed, the flux was found to be essentially identical for the largest thickness of lead than with no lead at all. This tells us that most of the incident muons were energetic enough not to be stopped by the lead. This can be confirmed by using the Bethe equation applied to lead [7]:

$$-\frac{dE}{dx} = \left(\frac{0.1216}{\beta^2}\right)\left[7.168 + \left(ln\beta^2 + ln\gamma^2 - \beta^2\right)\right] \times MeVg^{-1}cm^2 \tag{8}$$

From the simulation discussed in section 2, assuming a creation energy of 6 GeV at an altitude

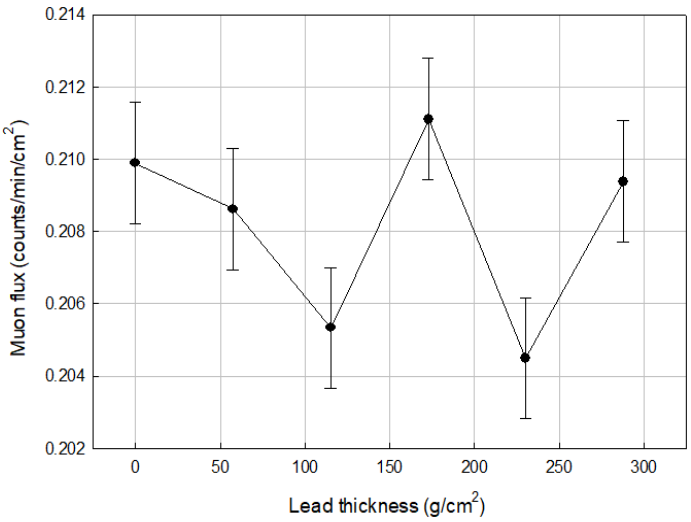

Figure 10: Cosmic ray muon flux for varying thicknesses of lead, obtained from 3 hour runs with a triple coincidence. Thickness in g/cm^2 was determined from the number of lead bricks used along with the density of lead. Flux was calculated from the measured count rates and the dimensions of the paddles.

of 15,000 m, the energy of a muon at sea level would be 2,823 MeV, which corresponds to $\beta = 0.99931$. Using this value in equation 8 with $dx = 288.0360$ g/cm^2 for the largest thickness used, it is found that the total amount of energy lost is 453 MeV. Thus, we can conclude that essentially all muons detected had an energy greater than this value, which is consistent with expectations. This also confirms that our setup was indeed counting primarily cosmic rays, since any other source of radiation would be much less energetic.

Nevertheless, caution must be used when interpreting these results, as the supply voltage and threshold settings were not ideal and it is possible that the bricks were not placed directly on top of the detectors.

This method for muon detection was shown to produce satisfactory results, especially when using a coincidence between two or three paddles. Indeed, the measured muon flux for double coincidences was shown to be on the order of the expected value. In addition, it was found that lead shielding up to a thickness of 288.0360 g/cm^2 (corresponding to 10 inches) had very little effect on the measured count rate, confirming that the detected particles were indeed very energetic and thus most probably muons.

However, some difficulties did arise with this setup, one of them being the inconsistencies in count rate for single detectors. These varied significantly depending on the location of the paddle and which side of it was facing up. This made us believe that there could have been a bad optical connection, either between the plastic scintillator and the light guide, or the light guide and the PMT. In addition, this method of muon detection does not allow for altitude dependence measurements, as it uses high voltage and NIM electronics, which are not easily portable. This is why a second, more compact method of muon detection was studied.

4 Cosmic muon detection using compact plastic scintillator, silicon photomultiplier, and surface-mount electronics

4.1 Setup

The setup for this method of muon detection is largely based off of the work of Axani et al. (see reference [1]). The circuit diagram of the system is shown in Fig. 11. In this case, charged particles are detected by a plastic scintillator in the same way as described in section 3.1, except that this slab of plastic is much smaller, with dimensions of $4.0 \times 4.5 \times 0.70$ cm. It was polished, wrapped in reflective foil, covered in black tape, and placed inside a black plastic box to avoid interference from surrounding light. In place of large photomultiplier tubes, we used a small Silicon photomultiplier (SiPM), which comes in the form of a very thin 6 mm^2 sheet. The model used was a C-Series 60035-SMT SensL SiPM, mounted on a small circuit board (which we will call the SiPM PCB), providing connections to the anode, cathode, and ground. In contrast with the previous setup, no light guide was used since the SiPM was pressed directly against the plastic scintillator by two screws drilled through the SiPM PCB and the scintillator. Again, optical gel was used at the interface between the SiPM and the scintillator to match indices of refraction. The SiPM consists of a large number of micro-cells each composed of silicon P-N junctions and has a breakdown voltage of 24.7 V, to which an over-voltage of up to 5.0 V can be added [1]. In our case, we applied a positive voltage of 29.3 V to the cathode. A DC-DC boost converter was used to provide this voltage from a 5 V input.

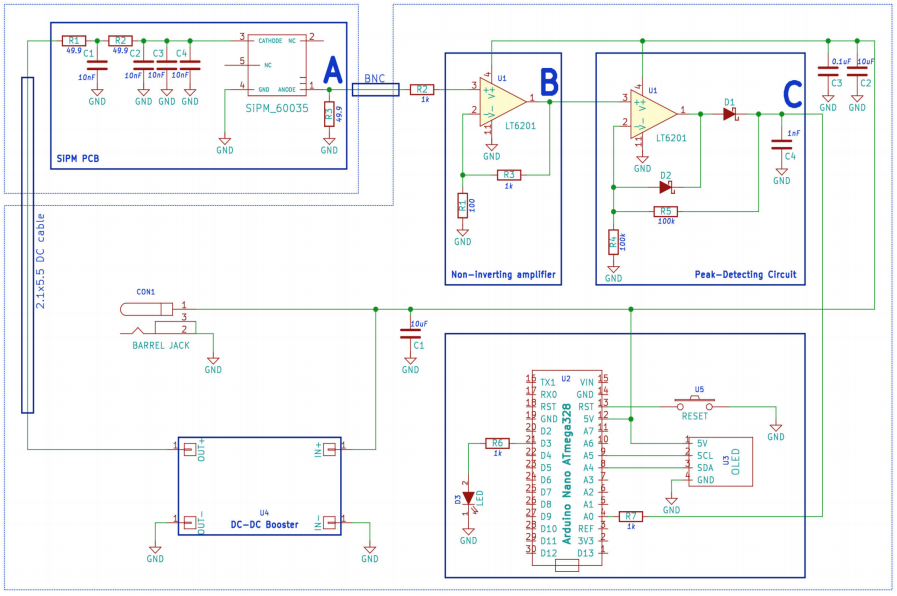

Figure 11: Circuit for the compact muon detector setup. This figure is taken from the github link provided by reference [1]. Our setup is essentially identical, except that we did not use the BNC cable or the reset button.

As can be seen in Fig. 11, the first part of the circuit consists of the SiPM circuit. It is contained on the custom-made SiPM PCB. Its goal is to filter out the noise from the high-speed switching of the DC-DC boost converter by a series of low-pass filters. In addition, a "pull-down" resistor is connected to the anode of the SiPM to hold the line at ground when there is no signal [1]. All resistors and capacitors were specifically designed for surface-mounting and were soldered onto the PCB. Connections between the components are built into the board. The SiPM PCB was connected to the rest of the circuit by soldered wires.

Another custom-made PCB, which we will call the main PCB, contains all the other parts of the circuit. First, a DC Jack was soldered onto the main PCB to provide all components with 5 V, which was supplied by a 12 V battery with a voltage regulator. The DC-DC boost converter was also soldered onto the main PCB. Its role is to take in the 5 V provided by the DC Jack to output the 29.3 V necessary to power the SiPM. Wires were connected from the output of the DC-DC boost converter to the SiPM PCB for this purpose. The signal from the SiPM was also connected to the main PCB. This raw signal is then sent through a traditional non-inverting amplifier, with a theoretical gain of 58.6, using a LT6201 op-amp, specifically designed to handle fast signals. The amplified signal then goes through a "peak-detecting circuit", which is designed to conserve the height of the pulse but extend its fall time by a factor of approximately 100. This is done so that the Arduino, which will be recording the data, has time to make multiple samples of the signal. Similarly to the SiPM PCB, all components used on the main PCB were specifically designed for surface-mounting and were soldered onto the board at the appropriate places. Connections between the components as well as all ground connections on the board are built in. The output of the peak-detecting circuit is then sent to one of the analog inputs of an Arduino Nano, which was soldered onto the main PCB. In addition, a 0.96" Yellow and Blue 128×64 OLED I2C Serial LCD screen was connected to the Arduino to display the information recorded by the device. A case for this screen was 3D-printed so that it could be placed outside of the balloon payload box in order to verify that the detector was functioning properly upon takeoff.

The Arduino is programmed using the Arduino IDE software. The code was obtained from a github link provided in reference [1]. Its main features include counting muon events and calculating the amplitude of the pulse for each event. This is done by comparing the voltage at the analog input to a threshold, set at the beginning of the program. If the threshold is exceeded, the Arduino takes five samples of the voltage, 5.8 μs apart. An exponential regression fit is then used to extrapolate the amplitude of the pulse. This is done because the voltage recorded by the Arduino at the instant the signal exceeds the threshold may not be the actual amplitude of the pulse. From this, the amplitude of the original SiPM pulse (raw signal) can be calculated using calibration parameters determined experimentally in reference [1]. We did not have the means to repeat the calibration and therefore used the same parameters, since our circuit was identical. In addition, the Arduino program also records the total number of counts along with time stamps for each event. By keeping track of dead-time, a count rate with uncertainty is calculated. The OLED screen displays the following information: total counts, run time, count rate (with uncertainty) and a histogram (resembling: - - - -) representing the relative amplitude of the most recently detected pulse. In order to record this information for future analysis, the Arduino Nano was connected to an Arduino Mega with a micro-SD shield on it. The Arduino Mega reads the information from the Arduino Nano by serial communication. It also reads information

from a GPS and saves the time, latitude, longitude, and altitude if a count is recorded by the Nano. The muon data as well as the GPS data are then saved to the SD card.

4.2 Signal Testing

We started by testing the output of the SiPM circuit by visualizing the pulses on a scope. In order to observe a large number of pulses with comparable amplitudes, we taped a Cs-137 button source to the plastic box. Pulses still appeared with a wide range of amplitudes, from approximately 10 mV to several hundreds of mV, with a concentration of pulses around 50 mV. The width of the signal was 500 ± 50ns. These results are consistent with those obtained by Axani et al., namely a "positive pulse, whose width is $\mathcal{O}(0.5\mu s)$ and height is typically between 10-100 mV" [1].

We then monitored the output of the non-inverting amplifier on the same scope, still using the Cs-137 button source. Again, pulses were found to have a wide range of amplitudes but now ranged from approximately 300 mV to slightly over 1 V. It was hard to calculate an actual gain, since pulses were constantly changing and it was difficult to determine which amplified signal corresponded to which SiPM signal. However, it was found that generally the gain seemed lower than the theoretical 58.6. In addition, the shape of the pulse was slightly modified: the peak was rounded and the width was increased to 1.5 ± 0.2 μs. These effects are due to the fact that the bandwidth frequency for this op-amp at a gain of 58.6 is approximately 2.5 MHz and the 0.5 μs width of the SiPM signal corresponds to a frequency of 2 MHz [1]. Thus the op-amp is operating at a frequency close to its bandwidth, which explains the distortion and the reduced gain. These effects should not have a significant incidence on the rest of the experiment.

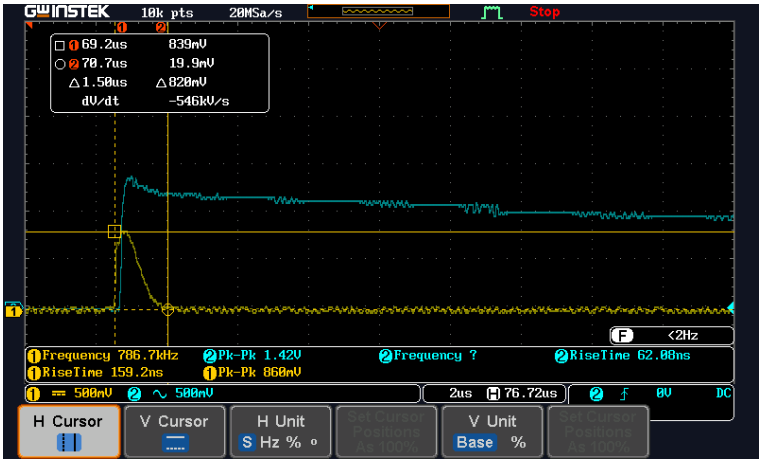

Figure 12: Screen-shot of oscilloscope trace showing the amplified pulse (Channel 1, in yellow) and the output from the peak detecting circuit (Channel 2, in blue). Cursors show the measurements for the height and width of the amplified pulse. These results were obtained by triggering on Channel 2 and with a Cs-137 button source taped to the detector box.

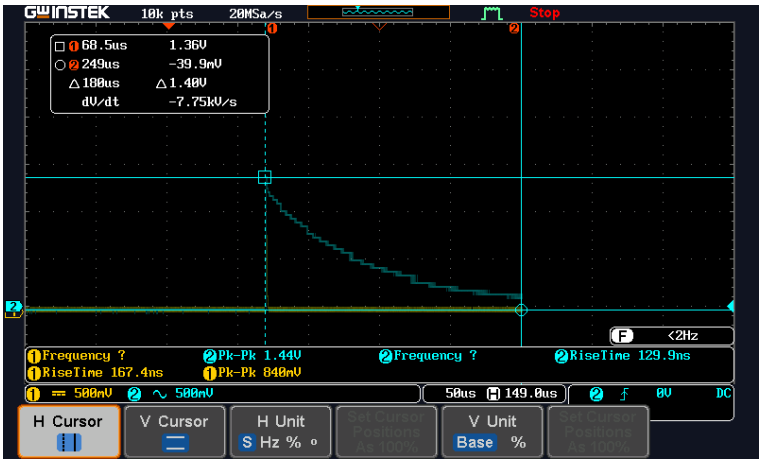

Figure 13: Screen-shot of oscilloscope trace showing the output of the peak-detecting circuit. This is the same pulse as in Fig.12 but with a larger time scale to see the entire fall time. Cursors show the measurements of height and width of the pulse.

Next, the scope was used to monitor both the amplified pulse and the output from the peak-detection circuit. A screen-shot of the scope trace is found in Fig. 12. Fig. 13 shows only the output of the peak-detecting circuit with a wider time scale in order to see the entire fall time. As expected, the peak-detecting circuit extends the width of the signal to 200 ± 50 μs. However, it appeared on the oscilloscope that the amplitude of the peak-detecting pulse was larger than the amplified pulse, which could be an effect of scope triggering or of the electronics. This is why, when calculating the correspondence between peak-detecting pulse amplitudes and SiPM pulse amplitudes, the Arduino code uses an equation obtained from a calibration and not simply the gain of the amplifier.

4.3 Threshold testing

In order to determine an ideal threshold to set in the Arduino code for which pulses to count as muon events, we collected data from three 50 minute runs without the Cs-137 source, with different thresholds. The threshold is applied to the peak-detecting pulse and is set by entering a number in the range from 0 to 1023, which corresponds to voltages between 0 and 5 V. The calibration equation obtained by Axani et al. can be used to calculate equivalent threshold voltages on the SiPM (raw) signal, which may be more meaningful, even though the threshold condition is not directly applied to this signal. The data was saved to the SD card, as described in section 4.1 and was then imported into Matlab. It was then straightforward to generate histograms for SiPM pulse amplitudes and peak-detecting pulse amplitudes. We also produced a plot of SiPM pulse amplitudes over time. Finally, we used the time stamps to calculate the time between successive counts and displayed these in a histogram (see Fig. 29 in the Appendix for the Matlab code). Fig. 14 through 17 show the results for the three different thresholds listed in Table 3.

Threshold	Code	Signal voltage threshold (V)	SiPM voltage threshold (mV)
1	100	0.488	36.08
2	150	0.733	47.28
3	200	0.978	58.73

Table 3: Thresholds used in Fig. 14 through 17 for counting muons using compact electronics. "Code" represents the number entered in the Arduino program, which the analog input recognizes as a voltage. "Signal voltage threshold" is the translation of the "code" into a voltage and is applied to the output of the peak-detecting circuit. "SiPM voltage threshold" is calculated from the "signal voltage threshold" using a calibration equation found in [1]. It is the equivalent threshold on the SiPM signal, even though it is not directly applied to this signal.

In Fig. 14 and 15 , we can see that the cutoffs in the histograms match the thresholds shown in Table 3, which gives us confidence that the Arduino was functioning as it should. In addition, these figures show that the majority of pulses have amplitudes between 50 mV and 100 mV for the SiPM voltage and around 1 V for the signal voltage (output of the peak-detecting circuit), which agrees with what we had observed on the oscilloscope. For threshold 1, it is visible in both Fig. 14 and 15 that the lowest bins are the largest, indicating that a fair amount of noise was being measured. We therefore conclude that this threshold is too low. On the other hand, for threshold 3, we suspected that the histograms are possibly being cut off in the middle of a peak, which indicates that the threshold might be too high. Thus we concluded that threshold 2, which corresponds to 0.733 V for the peak-detection pulse, was the most appropriate.

The four thresholds used are also evident in Fig. 16, as the plots are shifted higher and higher. We also notice that the amount of counts decreases as the threshold is increased, as expected.

In Fig. 17, we can see that for all distributions, the most probable value for time between counts is 1-2 seconds. However, as the threshold is increased, the histograms show longer tails, indicating that longer times between counts become more probable. For example, with threshold 3, it can occur that up to 40 seconds go by without detecting any counts, which is far from being the case with threshold 1. This confirms that threshold 3 is most likely too high.

From these tests, threshold 2 was selected for use in the balloon flight.

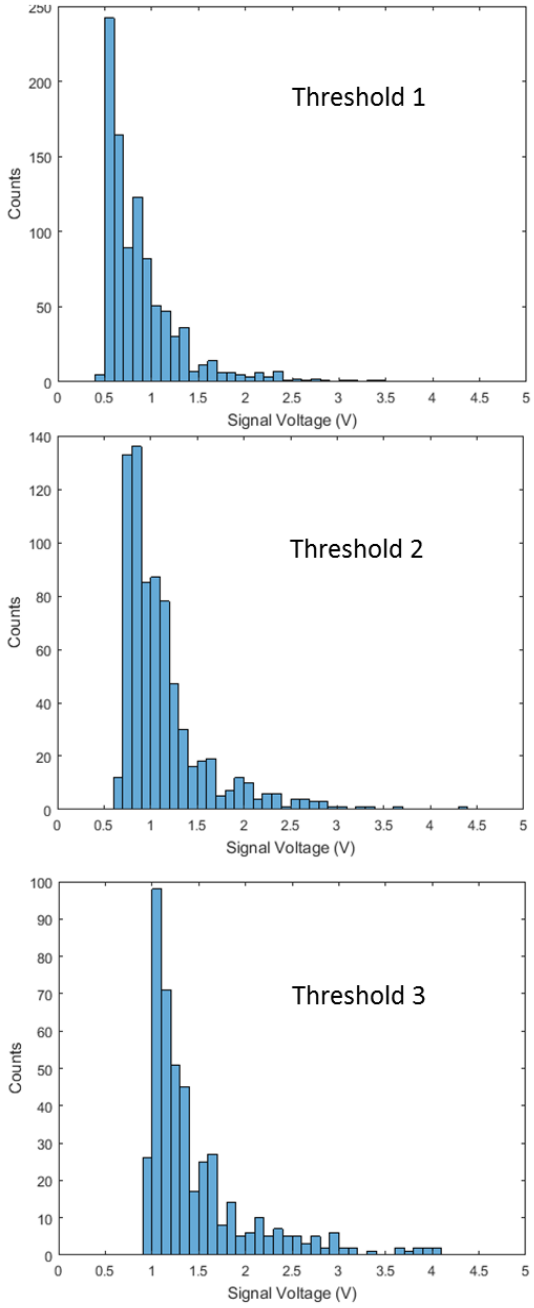

Figure 14: Histograms of SiPM pulse amplitudes for different thresholds, calculated from the amplitudes of the peak-detecting pulses using the calibration equation found in reference [1]. Each histogram represents approximately 50 minutes of run time with no radioactive sources present. Thresholds are listed in Table 3.

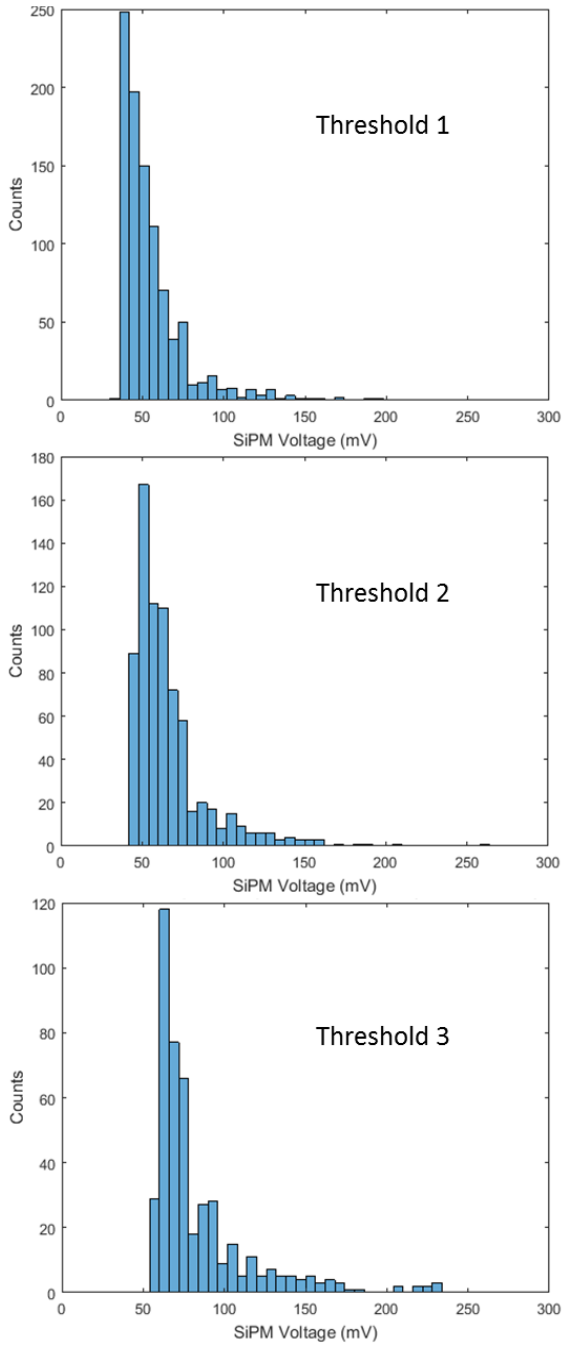

Figure 15: Histograms of signal pulse amplitudes (output of the peak-detecting circuit) for different thresholds, calculated from an exponential regression fit using five samples of the voltage, taken 5.8 μs apart. Each histogram represents approximately 50 minutes of run time with no radioactive sources present. Thresholds are listed in Table 3.

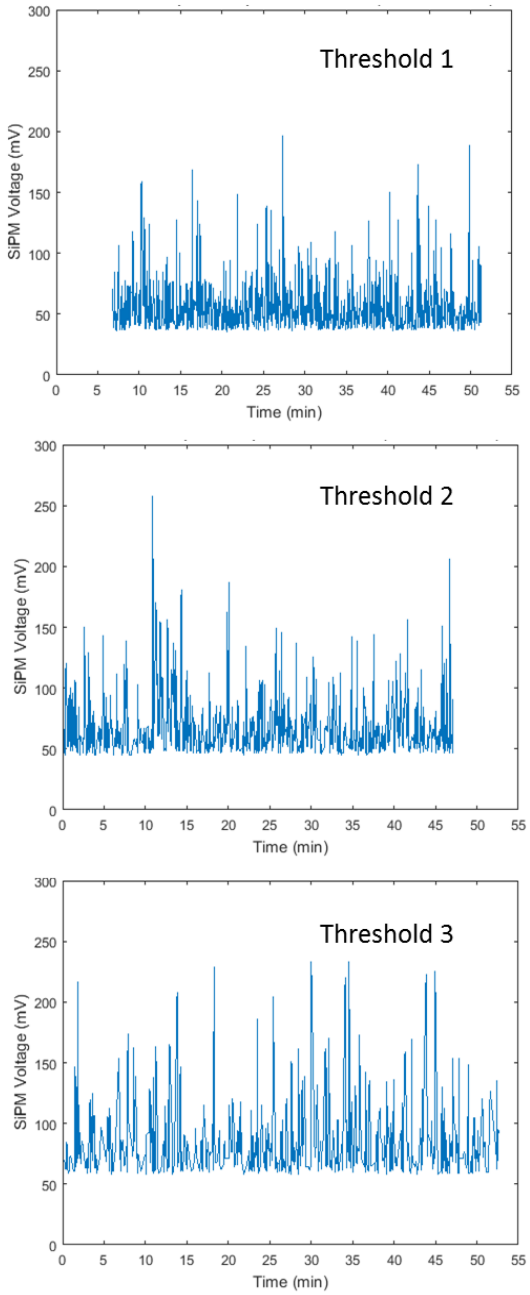

Figure 16: Plots of SiPM pulse amplitudes over time for different thresholds, calculated from the amplitudes of the peak-detecting pulses using the calibration equation found in reference [1]. Each plot represents approximately 50 minutes of run time with no radioactive sources present. Thresholds are listed in Table 3.

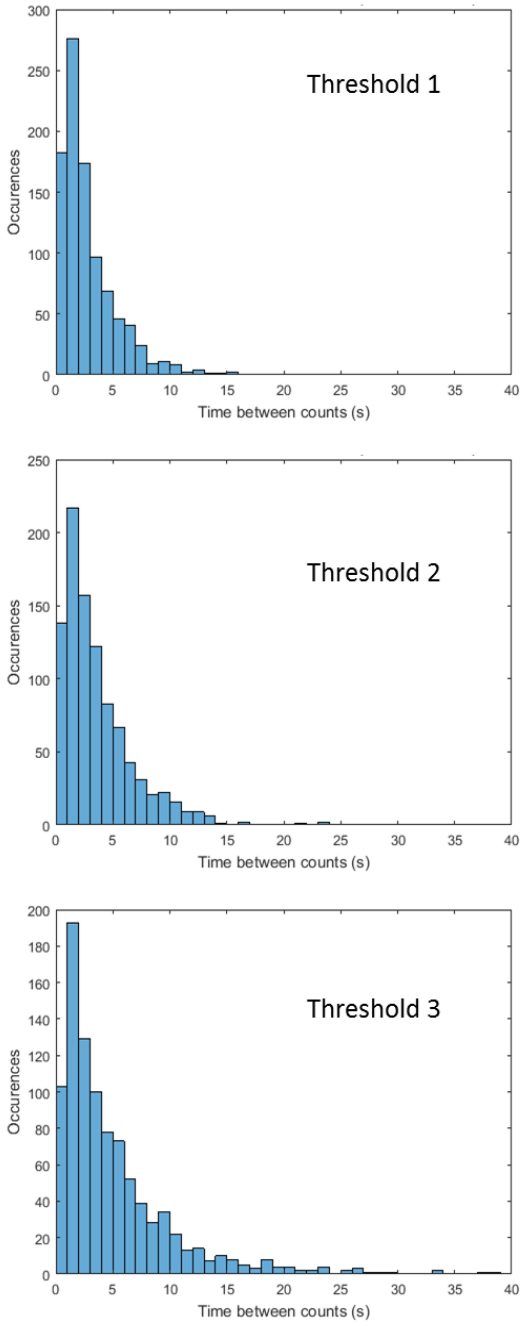

Figure 17: Histograms of time between successive counts for different thresholds, using the time stamps recorded by the Arduino. Each histogram represents approximately 50 minutes of run time with no radioactive sources present. Thresholds are listed in Table 3.

4.4 Calibration

In order to convert the amplitude of the recorded pulses into energy deposited in the detector, a calibration was conducted using known radioactive button sources. These were placed on top of the plastic box containing the detector and counts were recorded for 30 minutes to an hour. From this, histograms of the peak-detecting pulse amplitude such as those shown in Fig. 14 were obtained. In addition, a one hour run was conducted without any sources and with lead shielding around the detector to measure background. The resulting histogram was then subtracted from the ones obtained with the sources. The voltages at which the distributions seemed to show a peak in the number of counts were recorded for each sources. These were then plotted against the known energies of the decay products and fit to a linear regression. Table 4 lists the sources used with their associated energies and Fig. 18 shows the result of the linear calibration fit.

Source	Energies (MeV)
^{137}Cs	0.662
^{60}Co	1.173 and 1.333
^{22}Na	0.511 and 1.275
^{90}Sr	0.546

Table 4: List of radioactive button sources used for calibration with associated energies of decay products.

Assuming a linear relationship between energy deposited in the detector (E) and amplitude of the peak-detection pulse (V) of the form $V = aE + b$, it was found that the best fit parameters were $a = 0.75 \pm 0.03$ V/MeV and $b = 0.24 \pm 0.02$ V. This calibration equation was used in subsequent analysis.

It should be noted that this calibration is based on many estimations and assumptions. Indeed, the ability to clearly identify peaks in the histograms is based on the premise that all decay products deposit all their energy in the detector. However, since the plastic scintillator is thin, it is likely that this did not occur, particularly for energies greater than 1 MeV. Therefore, the peaks identified were very small and thus their voltage is uncertain. In addition, according to this calibration, the peaks for the decay products with energies less than 1 MeV should occur at voltages that approach the threshold of $0.586V$ used in these runs. Therefore, it is possible that the peak recorded was actually mostly noise, or that the peak was cut off by the threshold. Nevertheless, it is much more meaningful to conduct the analysis in terms of energy deposited rather than signal amplitude. And since the relationship between the two was found to be approximately linear, we will consider this calibration sufficient for our purposes.

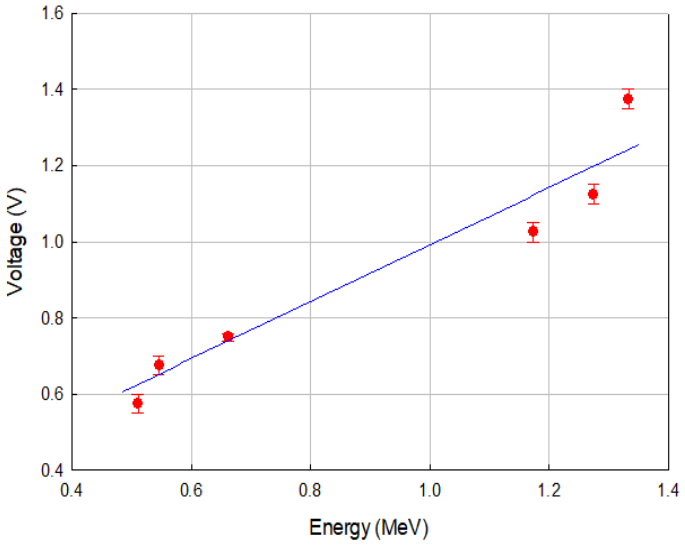

Figure 18: Calibration conducted to convert pulse amplitudes to energy deposited in the detector. Data was obtained by identifying peaks in the histograms of the peak-detection pulses for runs with different button sources placed on top of the detector, and matching them with the known energies associated with the sources (see Table 4). The linear fit yielded the following calibration equation relating voltage (V) to energy (E): $V = (0.75 \pm 0.03)E + (0.24 \pm 0.02)$.

4.5 Coincidence

From our work with the larger paddles and NIM electronics, it was determined that using a coincidence was a more reliable way to count muon events, as this reduces the rate of false positives due to electrical noise or low-energy particles. Therefore, a coincidence circuit was developed using two identical muon detectors, similar to the one discussed so far. The circuit diagram for this setup is shown in Fig. 19. This system was not used in the balloon flight, as it is still a work in progress. As can be seen in Fig. 19, both detectors are powered by a single DC-DC booster, as before. The raw detector pulses are then amplified using the same non-inverting amplifier as in Fig. 11. The amplified pulses are then sent to the non-inverting inputs of two comparators. The inverting inputs are connected to a common potentiometer, which serves as a variable threshold. The comparator model used was a LT1011, which is meant to handle fast signals. It produces a TTL logic pulse with a width equal to the time that the input signal is above the threshold. The comparator outputs then become the inputs to a logic AND gate, which produces a single TTL pulse with a width equal to the overlap between the two inputs. The model used was a NC7SZ08P5X, which is also designed for fast signals. Since the overlap between the two comparator pulses can be very small and is not regular, a monostable multivibrator was used to produce a TTL pulse with a set width for each coincidence count. The output of the AND gate is passed through a differentiator, which produces the downward-going pulse necessary to trigger the timer. The width of the resulting pulse is set by resistor R and capacitor C (see Fig. 19) using the

equation $\Delta t = 1.1RC$.

This setup was constructed on a standard circuit board. The detectors were connected to the board by coaxial cables to ensure mechanical stability and reduce electrical noise. The 5 V and ±12 V were supplied by a breadboard, for testing purposes. If the circuit were to be placed in the balloon, these supplies would come from batteries. The system was tested by monitoring the signals at different places in the circuit using an oscilloscope. It was found that the comparator pulses indeed resembled TTL logic pulses (although they were slightly rounded because of the high frequency of the signal) with a width ranging from 200 to 600 ns. The output of the AND gate was even narrower and it was difficult for the scope to trigger on it, which made testing a challenge. However, the differentiator did output the expected downward-going pulse from 5V to 0, with a width of approximately 190 ± 10 μs. Finally, the output of the 555 timer was found to have the expected square shape and a width almost exactly equal to the theoretical value of $1.1RC$.

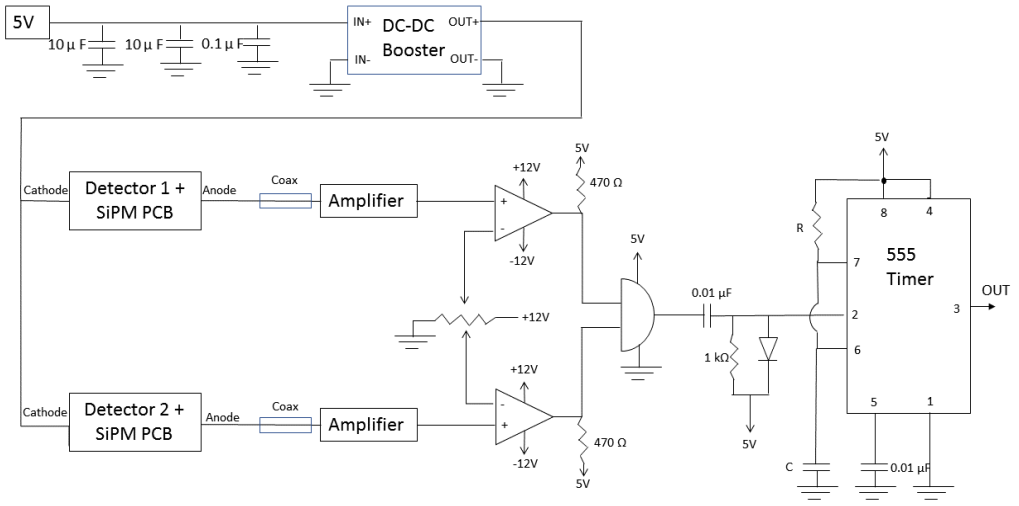

Figure 19: Coincidence circuit for compact muon detector setup. Amplified pulses are sent into fast comparators, having a common variable threshold set by a potentiometer. The comparator pulses then become the inputs of a fast logic AND gate. In order to extend and unify the width of the resulting logic pulse, a monostable multivibrator was included using a differentiator and a 555 timer.

Similarly to when using a single detector, threshold tests were conducted for this coincidence circuit. In contrast to the previous setup, where the threshold was set in the software, here is it a hardware threshold, set by the potentiometer. Because of the additional restriction of the coincidence as well as the different conditions on this circuit board, it was found that the hardware threshold needed to be set lower than what had been set previously in the software. The output of the AND gate was connected to a frequency counter to record coincidences over several hours. Ideally, the output of the 555 timer should have been used, but this test was conducted before the monostable multivibrator component was added to

Threshold (mV)	Counts per minute per cm^2
150	0.593 ± 0.010
200	0.515 ± 0.007
250	0.445 ± 0.010
300	0.434 ± 0.010
350	0.478 ± 0.009
400	0.399 ± 0.005
450	0.351 ± 0.010

Table 5: Measured muon flux for the coincidence circuit shown in Fig. 19 using different thresholds. Counts were recorded with a frequency counter over several hours and the corresponding flux was calculated from the dimensions of the detectors. Uncertainties result from taking the square root of the total number of counts.

the circuit. Each detector was placed in a black plastic box and the boxes were stacked on top of each other. Table 5 shows the measured flux for the 7 thresholds tested.

This table shows that the flux generally decreases as the threshold increases, as expected. The only exception to this trend is the 350 mV threshold, which yielded a greater count rate than the 300 mV threshold. This could simply be due to random fluctuations. Also, we notice that these values are very close to the ones obtained using the larger paddles (see Table 2), in addition to being on the order of the expected muon flux of $1/\text{min}/\text{cm}^2$. This confirms the validity and proper operation of the coincidence circuit. Comparing these values to the ones in Table 2, it would seem that thresholds of 250 mV to 350 mV produce the best agreement between both methods of muon detection. Therefore, thresholds in this range should be considered in future work.

The next step in developing this method would be to connect the output of the 555 timer to a digital pin on an Arduino Nano, which would also be measuring the amplitude of one or both of the detector pulses after passing them through the peak-detecting circuit. The TTL pulse produced by the coincidence circuit would then be used as a way to flag counts as coincidences. However, it was found that the Arduino could not read the digital pin fast enough to record all coincidence pulses. The values of R and C in the monostable circuit had to be increased to 100 kΩ and 10 μF, resulting in $\Delta t = 1.1$ s before the Arduino was able to detect the digital pin going high. This value for Δt is obviously too long, as it would result in missing a large fraction of counts. Alternate methods for reading digital pins in the Arduino software are currently being investigated to resolve this problem.

5 Balloon flight

5.1 Logistics

On Sunday, April 29, the compact muon detector discussed in Sections 4.1 through 4.4 was sent up in a high-altitude balloon to investigate the effect of altitude on the measured count rate and energy deposited in the detector. The system was divided into two black plastic boxes, one containing the plastic scintillator and SiPM PCB, and the other con-

taining the main PCB. Holes were drilled in the boxes to allow connections between the different parts of the circuit. The boxes were then taped together with black tape so ensure mechanical stability as well as minimal background light pickup. These were then zip-tied to a platform inside a Styrofoam payload box. The Arduino Mega was directly zip-tied to the same platform. The GPS probe connected to the Arduino Mega was secured to the side of the Styrofoam box. The battery supplying both the main PCB and the Arduino Mega was placed under the platform, at the bottom of the box. A Geiger counter, equipped with its own Arduino Mega and GPS, was also included in the payload, as well as two Go-Pro type cameras. Fig. 20 shows a picture of the payload. Hanging below the Styrofoam box was a GPS transmitter, which would allow us to track the balloon via the APRS website. A parachute was attached to the lid of the payload box to ensure a smooth descent after the balloon burst. On the parachute was another GPS which could be tracked via the Internet. The balloon was filled with hydrogen and helium gas and was launched at 10am. Fig. 21 shows a picture of it a couple seconds after takeoff.

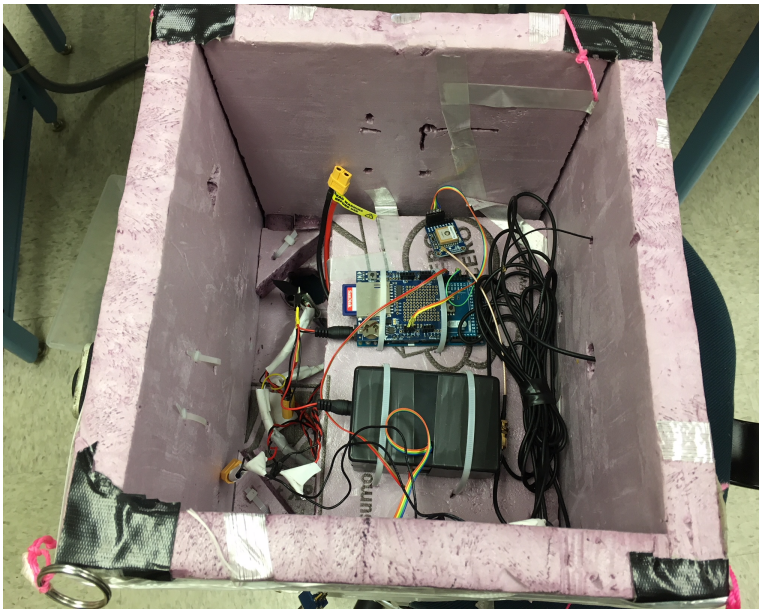

Figure 20: Picture of the Styrofoam payload box containing the compact muon detector system (in the black boxes), the Arduino Mega with the SD shield and GPS, and the portable batteries. The Geiger counter and cameras were added after taking this picture.

Unfortunately, the flight did not go exactly as planned. After approximately 50 minutes of flight, the payload box and the GPS transmitter were ripped apart and presumably disconnected from the balloon. The GPS transmitter was immediately retrieved in a field. However, the payload box was only found a week later by a local farmer, not far from where the transmitter had fallen. The lid and parachute were missing, as well as the cameras. The rest of the equipment was intact. It is still unclear whether the balloon and parachute continued flying after the fall of the payload and transmitter. The last location recorded by the GPS on the parachute was sent out approximately 15 minutes after the fall and corresponded to a location approximately 5-10 miles away. However, that location was searched

Figure 21: Picture of the balloon seconds after takeoff.

and nothing was found. The most probable hypothesis for these unfortunate circumstances is the very high winds, which could have torn the different components apart.

5.2 Data Analysis

The analysis for the data recovered from this flight presented some challenges. However, by combining the information from multiple instruments, we were able to conduct the desired analysis.

The data from the GPS transmitter turned out to be very useful in recovering lost information. It provided time stamps with corresponding latitude, longitude and altitude. The Geiger counter was set to record count rates every minute, along with latitude, longitude, and altitude from its GPS. However, the altitude data was found to be non-sensible, and since there were no time stamps, we could not directly infer the altitude from the transmitter information. Therefore, the latitude and longitude as a function of time were plotted for both the Geiger counter data and the transmitter data. By overlaying these two plots on the same graph and sliding the Geiger counter data until the two curves overlapped, we were able to determine the start time for this data set and therefore match a time stamp to each count rate value. As discussed in Section 4.1, the Arduino Nano on the muon detector was set to

record the pulse amplitude for each count, along with a time stamp in milliseconds since the last Arduino reset, and the dead-time for each measurement. For each count, the GPS was supposed to record an absolute time stamp, latitude, longitude, and altitude. However, for some unknown reason, the GPS stopped providing data at precisely 10:00am, which was the moment of the launch. The most likely explanation for this is that the motion due to the high winds prevented the probe from getting a fix. Time stamps for all the subsequent muon data were therefore calculated by assuming a start time of 10:00am. In addition, we had to account for the fact that the Arduino was found that have reset 14 times during the flight and thus the time stamps were initially not cumulative. Once this was done, it appeared that the detector only recorded data for a total of 7 minutes after takeoff.

Once time stamps were obtained for both the Geiger counter and muon data, altitudes could be determined. To do this, a plot of altitude as a function of time was produced from the transmitter data and fit to a linear spline function (a piecewise liner function passing through each data point), as seen in Fig. 22. From this plot, it is clearly visible that the GPS transmitter dropped after reaching an altitude of 33,000 m. It is also possible that the single point at 33,000 m resulted from an error in the transmitter, in which case the balloon would have only reached approximately 20,000 m.

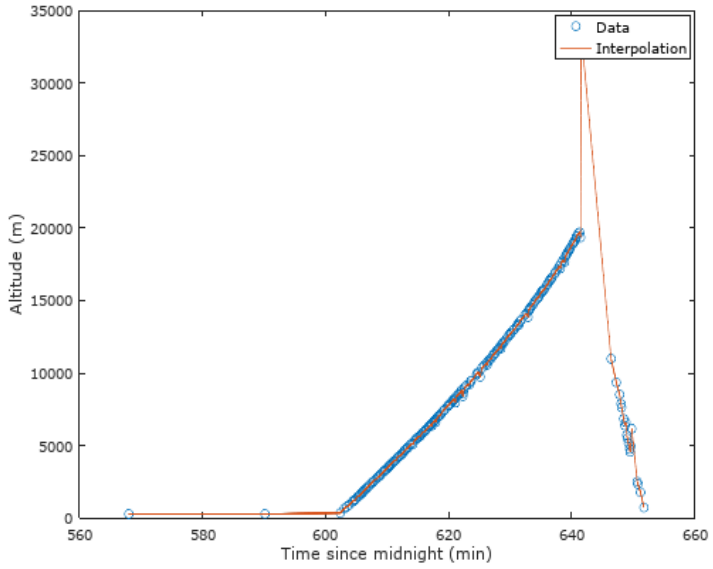

Figure 22: Plot of altitude as a function of time from the GPS transmitter data, with a linear spline fit. This shows that the transmitter fell to the ground after reaching a maximum altitude of 33,000 m.

The resulting interpolation was then used to calculate the altitude corresponding to each time stamp in the Geiger counter and muon data. From this, a plot of the count rate as a function of altitude for the Geiger counter was obtained and is seen in Fig. 23. We can see that the count rate increases with altitude at first, as expected. However, a sharp drop is observed at the end. It is too large and sudden to correspond to the Pfotzer maximum and most likely corresponds to the fall of the payload box. However, the interpolated altitude is still increasing during the drop. This is very surprising, as it would suggest that the GPS

transmitter was still rising while the payload box was falling, which does not seem possible. In addition, since the altitudes in Fig. 23 only range from 0 to 16,000 m, this is effect cannot due to the outlier point in the transmitter data. We therefore consider the Geiger counter data taken after the drop (corresponding to an altitude of about 11,000 m) to be non-reliable. Because the Pfotzer maximum was expected to be observed around 15,000 m, this issue prevented us from confirming the hypothesis.

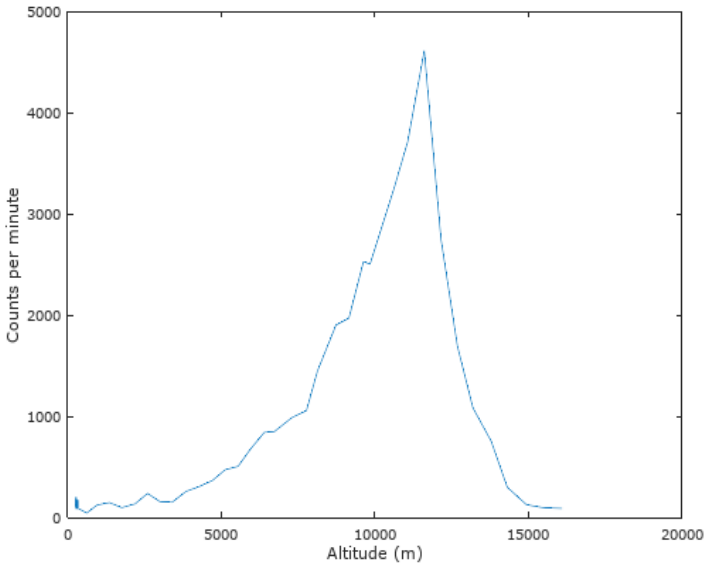

Figure 23: Plot of count rate as a function of altitude from the Geiger counter. Altitude was obtained using the interpolation found in Fig. 22. Data recorded after 11,000 m is considered unreliable.

The same interpolation for the altitude as a function of time was used for the muon detector data to obtain a plot of the count rate as a function of altitude. In this case, all counts recorded within 100 m intervals were selected and the count rate r was calculated using the following equation

$$r = \frac{N}{\Delta t - t_{dead}}, \tag{9}$$

where Δt is the time between the first and last count in that interval, t_{dead} is the total deadtime, and N is the number of counts. The corresponding altitude was then taken to be the average value over that interval. The resulting plot is found in Fig. 24. Fist, it can be seen that data was only collected up to 2,100 m, which is much less than we would have wanted. In addition, the observed trend is in complete disagreement with what we would expect. The count rate seems to decrease with altitude, with its highest value being at the very beginning of the flight. One explanation for this would be that the SiPM became loose mid-flight, which would reduce the amplitude of the pulses and thus the number of counts exceeding the threshold. It is also possible that there were some time lags between resets of the Arduino, resulting in time stamps and thus altitudes being incorrectly matched. Finally, some errors could have occurred in calculating the dead-time at the beginning of the flight,

which would lead to abnormally high count rates if the dead-time was too large. Because of all this, no conclusion can be made regarding muon count rate dependence on altitude.

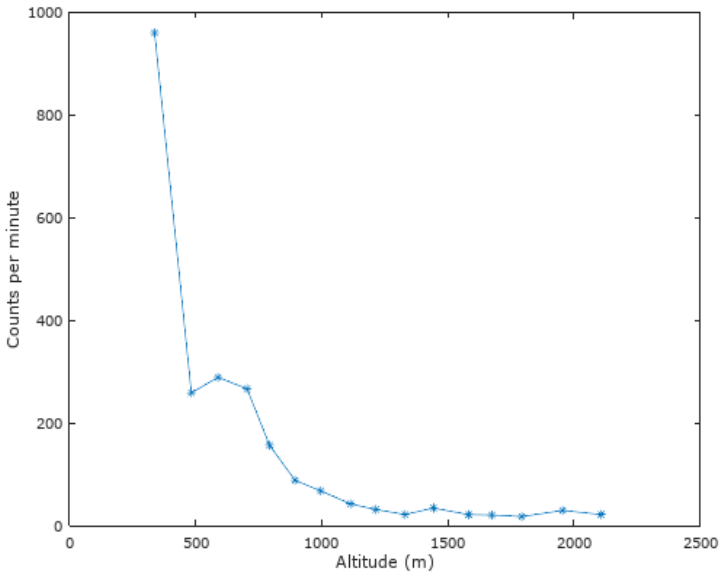

Figure 24: Plot of count rate as a function of altitude, as measured by the muon detector. The rate was calculated over a range of 100 m and the altitude was obtained from the interpolation found in Fig. 22.

From the pulse amplitude information provided by the muon detector, a plot of the energy deposited in the detector as a function of altitude was obtained. Pulse amplitudes were averaged over intervals of 100 m and were converted to energies using the calibration equation obtained in Section 4.4. The plot is shown in Fig. 25. These results are also rather inconclusive, as the energy seems to fluctuate independently of altitude. As with the count rate, the largest energies seem to occur at the lower altitudes. This would make sense if the SiPM had indeed developed a bad connection with the plastic scintillator mid-flight. Also, it is important to note that only the amount of energy deposited in the detector is recorded, not the total energy of the particle. Because muons are so energetic, they only loose a very small fraction of their energy when passing through the detector. Therefore, although muons loose some energy between the 2,100 m (the maximum altitude found in Fig 25) and the ground, this difference might not be large enough to produce a significant change in the amount of energy deposited. If data had been obtained at higher altitudes, this hypothesis could have been tested.

In summary, the results obtained from the flight were not reliable enough to confirm any of the hypothesis made initially or compare them to the simulations in section 2. However, the threshold and calibration tests recorded on the ground make us confident that more satisfactory results can be obtained in the future.

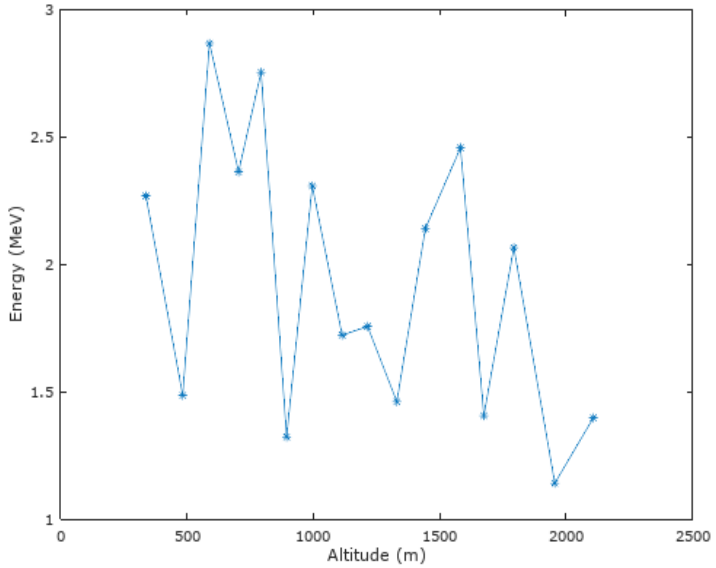

Figure 25: Plot of energy deposited in the detector as a function of altitude from the muon counter. Altitude was obtained using the interpolation found in Fig. 22. Energies were obtained from the pulse amplitudes using the calibration found in Fig. 18.

6 Conclusion

Throughout this study, we used plastic scintillators to detect charged particles from cosmic ray showers. We are fairly confident that we were detecting primarily muons because the count rates obtained did not differ significantly from the accepted value of the muon flux.

For the large paddle detectors, several different types of tests were done to characterize their behaviors and it was found that single detector count rates could vary widely depending on location or orientation, which we suspect could be due to bad optical connections. However, coincidence count rates were much more reliable and consistent with our understanding of muon flux.

The compact detector was found to produce satisfactory results as well when tested in the laboratory. Because it was determined with the paddles that coincidences improved our results, a coincidence circuit was developed for future use.

When the compact muon detector was placed in the high-altitude balloon, it did not produce the expected results. First of all, it only recorded data for a very short period of time and thus did not cover the desired range of altitudes. In addition, we suspect that either a bad optical connection between the SiPM and the scintillator occurred mid-flight, or altitudes were incorrectly matched because the values obtained for count rates and energy deposited in the detector followed opposite trends than what we would expect.

In the future, if this system were to be placed in another balloon, it would be important to make sure that all components are secured and that the SiPM is pressed tightly against the scintillator. In addition, tests should be conducted on the various GPS systems to make sure that they display the desired information and that they can withstand the harsh conditions of the flight.

Despite the inconclusive results from the flight, our ground tests demonstrate that the system is functional and that more satisfactory results could be obtained. These could then be compared to the results from the simulations discussed in Section 2 to test the accuracy of the models used, and modify them if needed.

Acknowledgements

I would like to thank everyone who contributed to this project, particularly Joshua Weisenfeld, Chuck Niederriter, Steve Mellema, and Tom Huber. I also thank Gustavus Adolphus College for providing the necessary funding and equipment.

References

[1] Axani S.N., Conrad J.M., and Kirby C. The Desktop Muon Detector: A simple, physics motivated machine and electronics shop project for university students. Massachusetts Institute of Technology.
http://studylib.net/doc/18745059/arxiv-1606.01196v3--physics.ed-ph--25-aug-2016

[2] Beer's law.
https://www.britannica.com/science/Beers-law

[3] Beringer J. et al. (Particle Data Group). Passage of particles through matter. *Phys. Rev. D* **86**, 2012.

[4] Carlson, P. and Watson, A. A. Erich Regener and the ionisation maximum of the atmosphere. *Hist. Geo Space Sc* **5**, 2014.

[5] Cosmic Rays. *The Great Soviet Encyclopedia, 3rd Edition*, 1979.

[6] Density of air.
https://en.wikipedia.org/wiki/Density of air

[7] Easwar, N. and MacIntire, D.A. Study of the Effect of Relativistic Time Dilation on Cosmic Ray Muon Flux - An undergraduate modern physics experiment. *Am. J. Phys* **59** (7), 1991.

[8] Gaussian distributed random numbers.
https://www.mathworks.com/matlabcentral

[9] Ho C.Y.E. Cosmic Ray Muon Detection using NaI Detectors and Plastic Scintillators. University of Virginia.
http://home.fnal.gov/ group/WORK/muonDetection.pdf

[10] LeCroy Corporation data sheets.
http://teledynelecroy.com/lrs/dsheets/365al.htm

[11] McNichols A.T. Variable Altitude Muon Detection and Energy Dependence of Cosmic Ray Muons. University of Hawai'i.
http://www.spacegrant.hawaii.edu/reports/23_SUM14-SP14/AMcNichols_S14.pdf

[12] Muon basics.
http://www2.fisica.unlp.edu.ar/ veiga/experiments.html

Appendix

```
function [xscaled Evec]=MuonEnergy(E,x)
% This function outputs vectors containing the position (xscaled) and energy (Evec)of a muon
% as it travels through the atmosphere, given an initial energy (E) and altitude (x)

%Constants:

Erest=105.6583745; %MeV
c=3*(10^10); %cm/s

p0=101325; %Pa
T0=288.15; %K
g=9.8; %m/s
L=0.0065; %K/m
R=8.31447; %J/(mol·K)
M=0.0289644; %kg/mol

%Functions used throughout the program:

fE=@(b) (fgamma(b)*Erest); %energy from beta
fgamma=@(b) (1/sqrt(1-(b)^2)); %gamma from beta
fdE=@(b) (0.1535/b^2)*(9.358+(log(b^2))+(log(fgamma(b))^2)-(b^2)); %energy loss from beta (Mev*cm^2/g)
fbeta=@(E) sqrt(1-(Erest/E)^2); % beta from energy
fp=@(x) (p0*((1-(L*x)/T0))^((g*M)/(R*L))); %pressure at altitude x (Pa)
fT=@(x) (T0-(L*x)); %temperature at altitude x (K)
frho=@(x) ((fp(x)*M)/(R*fT(x))); %density at altitude x (kg/m^3)
```

```
% Initializations:

dt=10^-8; %sec
i=1;
t=0;

Evec(1)=E; %initial energy
xvec(1)=x; %initial height
b=fbeta(E); %initial velocity

%Loop until the particle reaches the ground

while x>=0

  dx=fgamma(b)*b*c*dt; %distance traveled in time dt
  x=x-dx; %update position
  dE=fdE(b)*dx*(frho(x/100)/1000); %energy lost in distance dx, convert x to meters then convert density to g/cm^3
  E=E-dE; %update energy
  b=fbeta(E); %update velocity
  i=i+1;
  xvec(i)=x;
  Evec(i)=E;

end

xscaled=xvec./100; %convert to meters
plot(xscaled,Evec)
xlim([0 xscaled(1)])
ylim([0 Evec(1)])
xlabel('Altitude (m)')
ylabel('Energy (MeV)')

set ( gca, 'xdir', 'reverse' )
```

Figure 26: Matlab code for simulating the energy loss of a single muon passing through the atmosphere, used to obtain Fig. 4.

```
% This script generates plots of energy vs altitude for N muons with random initial energies and altitudes.

clear all

%Mean and standard deviation for random Gaussian distributions for initial enegergy and altitude
x0=1500000;
xsigma=200000;
E0=6000;
Esigma=1000;
N=100; %number of random generations

%Create vectors containing all random values
xrand=RandomGauss(x0,xsigma,N);
Erand=RandomGauss(E0,Esigma,N);

hold on
%Plot all solutions
for i=1:N
  [xscaled Evec]=MuonEnergy(Erand(i),xrand(i));
  plot(xscaled,Evec)
end

 hold off

 xlim([0 (max(xrand)/100)])
 ylim([0 max(Erand)])
 xlabel('Altitude (m)')
 ylabel('Energy (MeV)')

 set (gca,'xdir','reverse')
```

Figure 27: Matlab code for simulating the energy loss of multiple muons passing through the atmosphere, with random initial energies and altitudes. The function "RandomGauss" is a simple subroutine for generating a vector containing a given number of random values within a Gaussian distribution with known mean and standard deviation (found from reference [8]). This program was used to obtain Fig. 5.

```
% This script produces histograms the energy of N randomly generated muons at different altitudes.
clear all

alt=[0 5000 10000 15000]; % Select altitudes for histograms
numalt=length(alt);
%Mean and standard deviation for random Gaussian distributions for initial enegergy and altitude
x0=1500000;
xsigma=200000;
E0=6000;
Esigma=1000;
N=500; %number of random generations

%Create vectors containing all random values
xrand=RandomGauss(x0,xsigma,N);
Erand=RandomGauss(E0,Esigma,N);

Edist=zeros(N,numalt);

% Loop through the desired altitudes
for a=1:numalt
  % Loop through the N randomly generated events
  for i=1:N
    [xscaled Evec]=MuonEnergy(Erand(i),xrand(i));
    [closest index]=min(abs(xscaled-alt(a))); % Find index of altitude closest to the desired value
    Edist(i,a)=Evec(index);
  end

  % Plot the histogram with a fixed bin width of 500 MeV
  lastbinedge=min(Edist(:,a))+(500*(ceil((max(Edist(:,a))-min(Edist(:,a)))/500))); %adapt last bin edge
  edges=min(Edist(:,a)):500:lastbinedge;
  figure;hist(Edist(:,a),edges)
  h=fitdist(Edist(:,a),'Normal')
  title(['Energy distribution at ' num2str(alt(a)) 'm'])
  xlabel('Energy (MeV)')
  ylabel('Number of particles')
  xlim([0 E0+(4*Esigma)])
end
```

Figure 28: Matlab code for obtaining histograms of the energy of randomly generated muons at various altitudes, as seen in Fig. 6.

```
% This program generates histograms of signal pulse amplitude, raw detector pulse amplitude,
% and time between successive counts, as well as a plot of the raw detector pulse amplitude as a function of time
% from the muon detector data.

save_mode=0; %set to 1 to save files

% names of output files: the last 2 numbers represent the signal threshold
% (in mV) and the run  number for that threshold, respectively
thresh='1V';
file_out1=['Vsipm_hist_' thresh '.fig'];
file_out2=['Vsignal_hist_' thresh '.fig'];
file_out3=['Vsipm_vs_time_' thresh '.fig'];
file_out4=['time_between_counts_hist_' thresh '.fig'];

[Mdata endrow]=MuonData('muon.txt');

time=Mdata(:,2);
time_min=time./60000; % Convert from ms to minutes
Vsignal=Mdata(:,3);
Vsipm=Mdata(:,4);

% Calculate time between successive counts
for i=1:length(time)-1
    time_between_counts_ms(i)=time(i+1)-time(i);
end
time_between_counts_s=time_between_counts_ms./1000;

% Create and label the figures
edges1=0:6:300;
h1 = figure;
hist(Vsipm)
xlabel('SiPM Voltage (mV)')
ylabel('Counts')
title(['Raw detector pulse amplitude distribution (' thresh ' threshold)'])

edges2=0:0.1:5;
h2 = figure;
hist(Vsignal)
xlabel('Signal Voltage (V)')
ylabel('Counts')
title(['Peak detection pulse amplitude distribution (' thresh ' threshold)'])

h3 = figure;
plot(time_min,Vsipm);
xlabel('Time (min)')
ylabel('SiPM Voltage (mV)')
title(['Raw detector pulse amplitude over time (' thresh ' threshold)'])

edges=0:1:40;
h4 = figure;
hist(time_between_counts_s)
xlabel('Time between counts (s)')
ylabel('Occurences')
title(['Distribution of time between counts(' thresh ' threshold)'])

% Save the figures
if save_mode==1
    savefig(h1,file_out1)
    savefig(h2,file_out2)
    savefig(h3,file_out3)
    savefig(h4,file_out4)
end
```

Figure 29: Matlab used to obtain the histograms for threshold testing, found in Fig. 14 through 17.

```
function f=Altitude

% This program produces a function handle consisting of a linear interpolation
% of the altitude of the balloon as a function of time.

BRBdata=importdata('BRBdata.mat'); % data from GPS transmitter
x=(BRBdata(:,1).*1440)-300; % time stamps, converted to minutes from midnight
y=BRBdata(:,7); % altitude
u=x(1):0.01:x(end); % time values to plot the interpolation
f=@(u) piecelin2(x,y,u); % interpolating function
interpvals=arrayfun(f,u); % altitude values to plot the interpolation

figure;plot(x,y,'o',u,interpvals,'-')
xlabel('Time since midnight (min)')
ylabel('Altitude (m)')
legend('Data','Interpolation')
```

Figure 30: Matlab code obtaining a piecewise linear interpolation of the altitude as a function of time, using the GPS transmitter data. The function "BRBdata" is a subroutine written to parse the data from the text file. The function "piecelin2" is found in Fig. 31. This script was used to obtain Fig. 22.

```
function v = piecelin2(t,y,u)
%  This function finds the piecewise linear value of S(u)
%  with S(t(j)) = y(j) and returns v = S(u).
% (code written by Michael Hvidsten)

%  Find subinterval index i so that t(i) <= u < t(i+1) or u= t(i+1)
 n = length(t);
 i=0;
 for j = 1:n-1
      if  (t(j) <= u && u <  t(j+1)) || u==t(j+1);
            i = j;
      end
 end
if i==0 error('u value is not in spline interval');
end
%  Evaluate spline at u
v = y(i) + (u-t(i))*(y(i+1)-y(i))/(t(i+1)-t(i));
```

Figure 31: Matlab code for creating and evaluating any piecewise linear interpolation, given a set of data. It is used in the code found in Fig. 30.

```
% This script produces a plot of count rate as a function of altitude
% from the Geiger counter data

Gdata=GeigerData('GeigerCounterData.txt'); % Geiger Counter data
Gtime=Gdata(:,1); % Time (in minutes since midnight)
Gcounts=Gdata(:,2); % Counts per minute
n=length(Gtime);

Galt=zeros(n,1);

f=Altitude; % Piecewise linear interpolation of altitude as a function of time
Galt=arrayfun(f,Gtime);
Gtime_plot=Gtime-582; % Convert time to minutes from the beginning of data collection

% Plot altitude as a function of time for testing purposes
%figure;plot(Gtime_plot,Galt);
%xlabel('Time (min)')
%ylabel('Altitude (m)')

% Plot count rate as a function of altitude
figure;plot(Galt,Gcounts);
xlabel('Altitude (m)')
ylabel('Counts per minute')
```

Figure 32: Matlab code for obtaining a plot of count rate as a function of altitude for the Geiger counter data, as seen in Fig. 23. The function "GeigerData" is a subroutine written to parse the data from the text file.

```
% This program produces a plot of count rate as a function of altitude from the muon detector data

Mdata=MuonData('FLYDATA2.TXT'); % data from muon detector
Mtime_ms=Mdata(:,2); % Time stamps in milliseconds
Mdeadtime=Mdata(:,5); % Deadtime in milliseconds for each count
Mtime=(Mdata(:,2)./60000)+600; % Time converted to minutes since midnight
n=length(Mtime);

Malt=zeros(n,1);

f=Altitude; % Piecewise linear interpolation of altitude as a function of time
Malt=arrayfun(f,Mtime);

% Loop through all recorded counts
start=1;
  i=1;
  while start<n

    indices=find(Malt(start)<=Malt & Malt<(Malt(start)+100)); % Find indices corresponding to 100m altitude intervals
    first=indices(1);
    last=indices(end);
    deltat=Mtime_ms(last)-Mtime_ms(first); % Time interval
    deadtime=sum(Mdeadtime(first+1:last)); % Total deadtime (excluding first count)
    deltat_true=deltat-deadtime; % Time interval, taking into account deadtime
    deltat_true_min=deltat_true./60000;
    counts=length(indices)-1; % Number of counts in the interval (excluding first count)
    Mcountrate(i)=counts/deltat_true_min; % Counts per minute
    Malt_avg(i)=mean(Malt(indices)); % Average altitude over the interval

   start=indices(end)+1; % move to new 100m altitude interval
   i=i+1;

 end

 figure;plot(Malt_avg,Mcountrate,'-*')
 xlabel('Altitude (m)')
 ylabel('Counts per minute')
```

Figure 33: Matlab code for obtaining a plot of count rate as a function of altitude for the muon detector data, as seen in Fig. 24. The function "MuonData" is a subroutine written to parse the data from the text file.

```
% This program produces a plot of the energy deposited in the muon detector
% as a function of altitude.

Mdata=MuonData('FLYDATA2.TXT'); % data from muon detector
Mvolt=Mdata(:,3); % pulse amplitude for each count
Mtime=(Mdata(:,2)./60000)+600; % time, converted to minutes since midnight
n=length(Mtime);

Malt=zeros(n,1);

f=Altitude; % Piecewise linear interpolation of altitude as a function of time
Malt=arrayfun(f,Mtime);
figure;plot(Mtime,Malt); % Plot altitude as a function of time for testing purposes

  % Loop through all recorded counts
  start=1;
  i=1;
  while start<n

    indices=find(Malt(start)<=Malt & Malt<(Malt(start)+100)); % Find indices corresponding to 100m altitude intervals
    Mvolt_avg(i)=mean(Mvolt(indices)); % average of pulse amplitude
    Malt_avg(i)=mean(Malt(indices)); % average altitude

   start=indices(end)+1; % move to the newt 100m altitude
   i=i+1;

 end

 Menergy=(Mvolt_avg-0.2438)./0.74868; % Convert voltage to energy using the calibration equation

 figure;plot(Malt_avg,Menergy,'-*')
 xlabel('Altitude (m)')
 ylabel('Energy (MeV)')
```

Figure 34: Matlab code for obtaining a plot of energy deposited in the muon detector as a function of altitude for the muon detector data, as seen in Fig. 25.