3-11-2011

Directional Pair-Production Spectrometer Design for Airborne Stand-Off Detection of Special Nuclear Material

William L. Harrell

DIRECTIONAL PAIR-PRODUCTION SPECTROMETER DESIGN FOR AIRBORNE STAND-OFF DETECTION OF SPECIAL NUCLEAR MATERIAL

THESIS

William L. Harrell, 1st Lieutenant, USAF

AFIT/GNE/ENP/11-M09

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GNE/ENP/11-M09

DIRECTIONAL PAIR-PRODUCTION SPECTROMETER DESIGN FOR AIRBORNE STAND-OFF DETECTION OF SPECIAL NUCLEAR MATERIAL

THESIS

Presented to the Faculty

Department of Engineering Physics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Nuclear Engineering

William L. Harrell, BS

1st Lieutenant, USAF

March 2011

AFIT/GNE/ENP/11-M09

DIRECTIONAL PAIR-PRODUCTION SPECTROMETER DESIGN FOR AIRBORNE STAND-OFF DETECTION OF SPECIAL NUCLEAR MATERIAL

William L. Harrell, BS
1st Lieutenant, USAF

Approved:

Maj Benjamin R. Kowash (Chairman)	Date
Larry W. Burggraf, PhD (Member)	Date
Lt Col Christopher S. Williams (Member)	Date

Abstract

The purposes of this research are to experimentally and theoretically prove the concept of a directional pair-production spectrometer to detect and locate the tailings that are created when making Special Nuclear Material (SNM) at stand-off distances from a remotely piloted vehicle (RPV). A directional pair-production spectrometer uses the information garnered from the high energy gamma rays emitted by these SNM manufacturing tailings to perform pair-production spectroscopy and identify the isotope of interest. Through simultaneous operation as a Compton camera, the detection system will be able to measure rudimentary directional information from the medium energy gamma decays. The detector used for this research is constructed of four $LaBr_3$ detectors and operated in coincidence to allow for the reduction of background. The directional efficiency of the detector in measuring the radioactive decay of a 6.7 μCi Co-60 source is validated with a Geant4 simulation. The simulation is used to predict the directional efficiency of a detector system using six detector elements and the pair-production spectrum that would be seen when measuring a higher energy gamma ray source.

Acknowledgements

I would like to express my sincere appreciation to my faculty advisor, Maj. Benjamin Kowash, for his guidance and support during this thesis project. I would also like to extend my gratitude to the members of my thesis committee, Dr. Larry Burggraf and Lt. Col. Christopher Williams, for their additional advice and insight. I must also thank Dr. Abigail Bickley for the many hours spent helping me limp through and battle Geant4 and ROOT and to Mr. Eric Taylor for tolerating my many requests for equipment that enabled the success of this project. Finally, I want to thank my wife for her unconditional support despite the many long hours spent to complete this research. None of this could have been accomplished without her support.

William L. Harrell

Table of Contents

Page

Page

List of Figures

Figure Page

Figure Page

Figure Page

Figure Page

List of Tables

Table Page

DIRECTIONAL PAIR-PRODUCTION SPECTROMETER DESIGN FOR AIRBORNE STAND-OFF DETECTION OF SPECIAL NUCLEAR MATERIAL

I. Introduction

I. 1. Motivation

Recent developments in the Middle East have thrust to the forefront, more than ever, the necessity of being able to determine whether or not a state or non-state actor is attempting to process the materials necessary for the development of a nuclear weapon. Though attempts by the International Atomic Energy Agency (IAEA) to gain access to the nuclear facilities of interest have been helpful in determining whether the supposed peaceful nuclear power programs are indeed what the countries say they are, there remains an air of uncertainty amongst the world powers who strive for peaceful nuclear power. What is needed to help determine the true intentions of these programs is a device that can look for the unique spectroscopic signatures of some of the products created during the development of the special nuclear material (SNM) used in a nuclear weapon. One of the isotopes of interest in the detection of SNM is U-232. Small amounts of U-232 are created when producing U-233 through the irradiation of Th-232. U-232's decay chain emits a 2.6 MeV gamma when Tl-208 β- decays to Pb-208. The high-energy gamma of this decay product is extremely significant and allows the possibility of stand-off detection. The higher energy the gamma, the less likely it is to be attenuated through a given volume of air. A 2.6 MeV gamma has a mean free path (mfp) of approximately 200 meters in air that makes it a prime candidate for stand-off

detection. A gamma of this energy when incident on a high Z material such as $LaBr_3(Ce)$ (the Z of La is 57) is likely to interact with the material via either a Compton or a pair-production process. Multiple detectors of a high Z material, such as $LaBr_3(Ce)$, can be mounted in the front of a remotely piloted vehicle (RPV). Optimal configuration with the proper timing characteristics would allow for accurate detection of coincident Compton and pair-production events, thus creating a system that could successfully perform stand-off spectroscopy while giving rudimentary directional information.

I. 2. Background

A solution to the stand-off detection problem is to construct a detector that is capable of performing gamma-ray spectroscopy using the information garnered from the pair-production and Compton interactions and further uses the information from the Compton events to give the location of the radioactive source. Pair-production spectroscopy is well documented and typically uses timing methods to only select those events which interact by triple coincidence [**1**]. This method simplifies the spectral response towards an ideal single peak [**2**].

I. 3. Previous Work

For a complete background and treatment, the reader is referred to J.H. Winso [**3**]. Space Micro Inc. demonstrated an autonomous gamma source detector vehicle, named Radiation Source Identification and Targeting (RADSITE^{TM}), capable of determining the location of a source to within ±5° azimuthally over a 360° field of view when the source was below approximately a 20° angle above the horizon. The detector geometry used for this application comprised of a 4-element $LaBr_3(Ce)$ detector and is shown in Figure 1. It

was designed to have a geometry that enabled 360° planar detection. The software did not discriminate against any gamma energy and merely used the sensitivity of the detectors and their known orientations to compute the most probable location of the source. The vehicle used was an iRobot ATRV Robotic Platform manufactured by iRobot Co. shown in Figure 2 [**3**]. The geometry and software used by Space Micro Inc. can be improved to enable operation of the detection system as a pair-production spectrometer and a Compton camera providing more accurate isotope identification and isotope location capabilities.

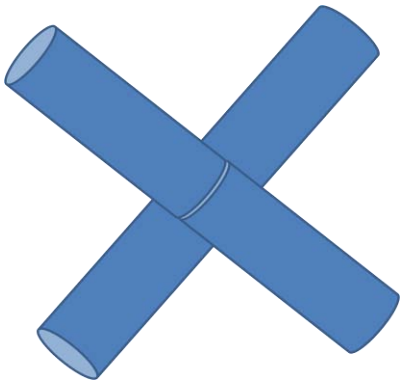

Figure 1. Geometry used by Space Micro Inc. on RADSITE™. This design is capable of locating a source to within ±5° azimuthally over a 360° field of view when the source was below approximately a 20° angle above the horizon [3**].**

Figure 2. Shown is the iRobot ATRV Robotic Platform that the RADSITE™ detector used to maneuver [3**].**

The following section provides the necessary background to understand why pair-production spectroscopy techniques were implemented in this project.

I. 4. Pair-production Spectroscopy

Optimization of the detector to accurately detect pair-production events is the primary objective of this research project. Spectroscopy can then be performed on the source in question to positively confirm the presence of SNM. As is shown in Figure 3, as gamma ray energy increases, the mechanism of interaction for a given material changes from photoelectric effect, to Compton effect, and lastly, to pair-production. Gamma rays with an energy above approximately 0.4 MeV have an increased probability of interacting with Lanthanum, which has a Z (number of protons) of 57, through either Compton scattering.

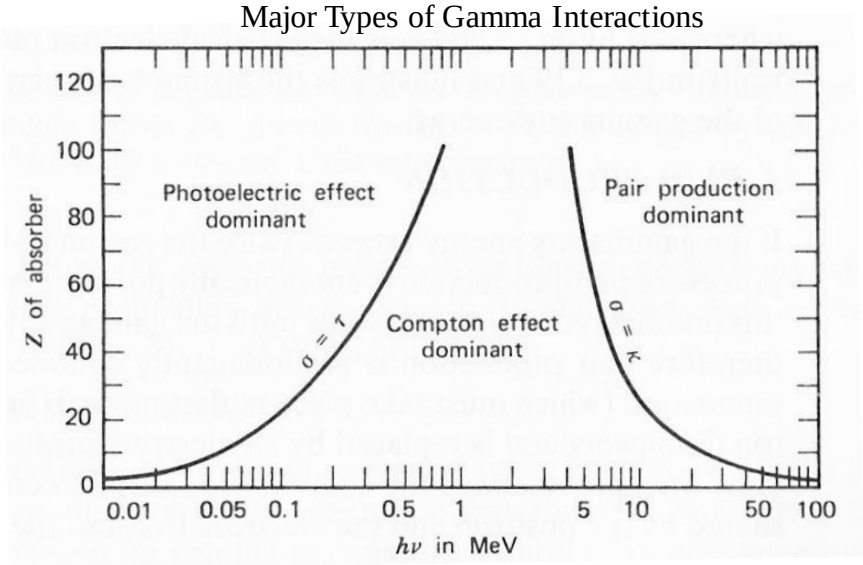

Figure 3. Shown are the three major types of gamma-ray interaction. The two interactions of interest for this application are Compton Effect and pair-production, both of which occur with higher energy gamma ray energies [4, 52].

The cross-sections for the Compton Effect and pair-production for $LaBr_3$ are presented in Figure 4 [**5**]. All of the cross-sections for the energies that are of interest for this project have been tabulated in Table 1. It can be determined from the figure that the point at which it becomes more likely for a gamma to interact predominately by pair-production in Lanthanum is approximately 7 MeV. This does not mean that gammas cannot interact by this mechanism below 7 MeV, but it is less likely than the competing interaction, the Compton Effect.

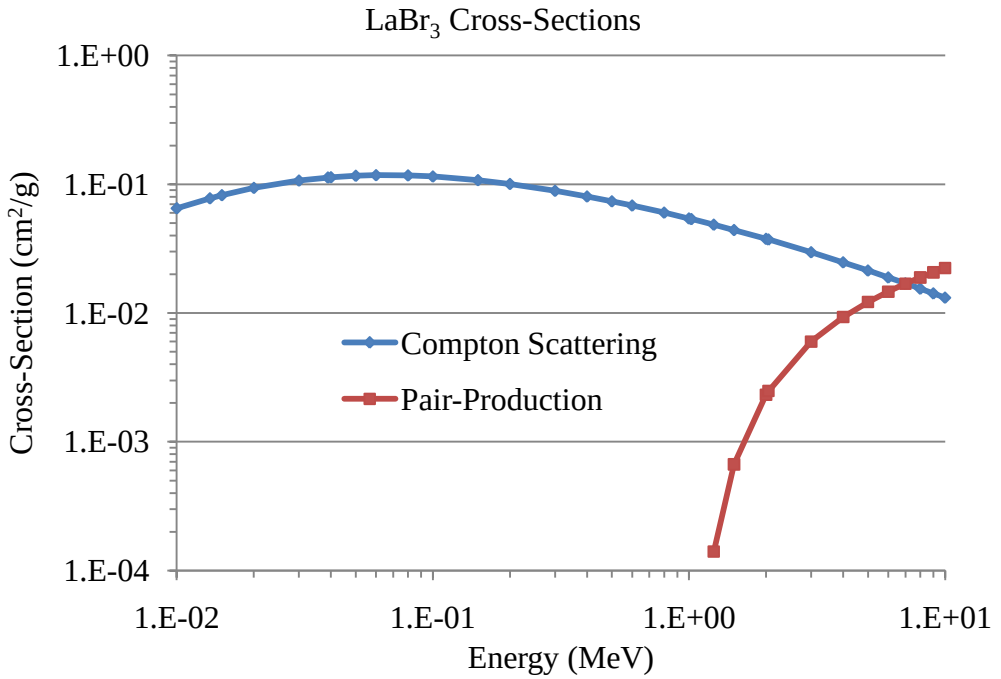

Figure 4. Shown are the cross-sections of interest for this project, namely the Compton Effect and pair-production, in $LaBr_3$ [5].

Table 1. Cross-Sections for 0.511, 1.173, 1.332 and 2.614 MeV Gamma Rays in $LaBr_3(Ce)$[5]

Gamma Ray Energy (keV)	511	1173	1332	2614
Coherent Scattering (cm^2/g)	3.78E-03	7.34E-04	5.70E-04	1.49E-04
Compton Scattering (cm^2/g)	7.31E-02	5.02E-02	4.70E-02	3.23E-02
Photoelectric Absorption (cm^2/g)	1.16E-02	1.85E-03	1.46E-03	4.67E-04
Pair Production (cm^2/g)	0.00E+00	5.25E-05	2.78E-04	4.61E-03
Total Attenuation (w/ Coherent Scattering) (cm^2/g)	08.85E-02	5.28E-02	4.93E-02	3.75E-02
Total Attenuation (w/o Coherent Scattering) (cm^2/g)	8.47E-02	5.20E-02	4.87E-02	3.74E-02

Pair-production spectroscopy typically involves three detectors arranged as shown in Figure 5. When the incoming gamma deposits its energy in the central detector, the resulting positron annihilates with a surrounding electron and will generate two 511 keV photons that are emitted collinearly and with some probability escape from the detector they are born. This process is illustrated below in Figure 5.

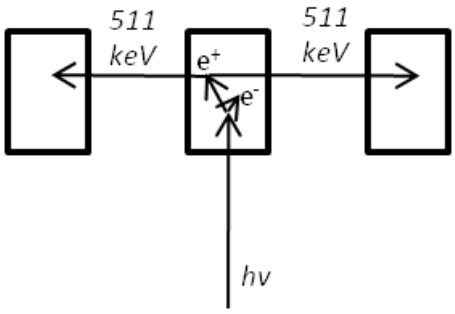

Figure 5. Shown is a block diagram of the typical orientation of a pair-production spectrometer. The proposed spectrometer will attempt to capture one of the 511 keV gamma rays produced by annihilation.

In a perfect scenario, these annihilation photons always escape the primary detector and interact in the two detectors that are positioned around the primary detector. If these three detectors are set up in coincidence, where only events that deposit an initial

amount of energy in the primary detector and two coincident 511 keV gamma rays in the other two detectors are counted, the result is a very specific double escape spectrometer [4, 439]. While the energy spectrum will have much of its background and Compton continuum reduce when operated in coincidence mode than if not, the photopeak will not be the ideal delta function [**2**]. The events that fall in energies above the photopeak are due to the annihilation photons scattering in the initial detector before exiting. The scattered annihilation photons however, retain enough energy to still appear in the annihilation peak of the secondary detectors. The continuum events at the energies below the photopeak are due to the positrons or electrons escaping the initial detector before they have fully stopped [**1**].

I. 5. Compton Camera

Optimization of the detection system to accurately detect Compton scattered gammas involves many geometry and timing challenges. Compton scattering in two detectors occurs when an incident gamma ray interacts in the initial detector, creates a recoil electron and the incident photon is scattered into a second detector. This scattering occurs at an angle, whose probability can be calculated using the Klein-Nishina formula,

$$\frac{d\sigma}{d\Omega} = Zr_0^2 \left(\frac{1}{1+\alpha\ 1-\cos\theta} \right)^2 \left(\frac{1+\cos^2\theta}{2} \right) \left(1+\frac{\alpha^2\ 1-\cos\theta\ ^2}{1+\cos^2\theta\ \left[1+\alpha\ 1-\cos\theta\right]} \right) \quad (1.1)$$

where is the number of protons of the detector, and is the energy of the incident gamma, is the rest mass energy of the electron (0.511 MeV), is the angle of scatter, and is the classical electron radius [4, 51]. This calculation has been completed for the two primary gammas associated with the common calibration source

Co-60, 1173 keV and 1332 keV, as well as 2 MeV and 2.6 MeV sources. The results are shown in Figure 6.

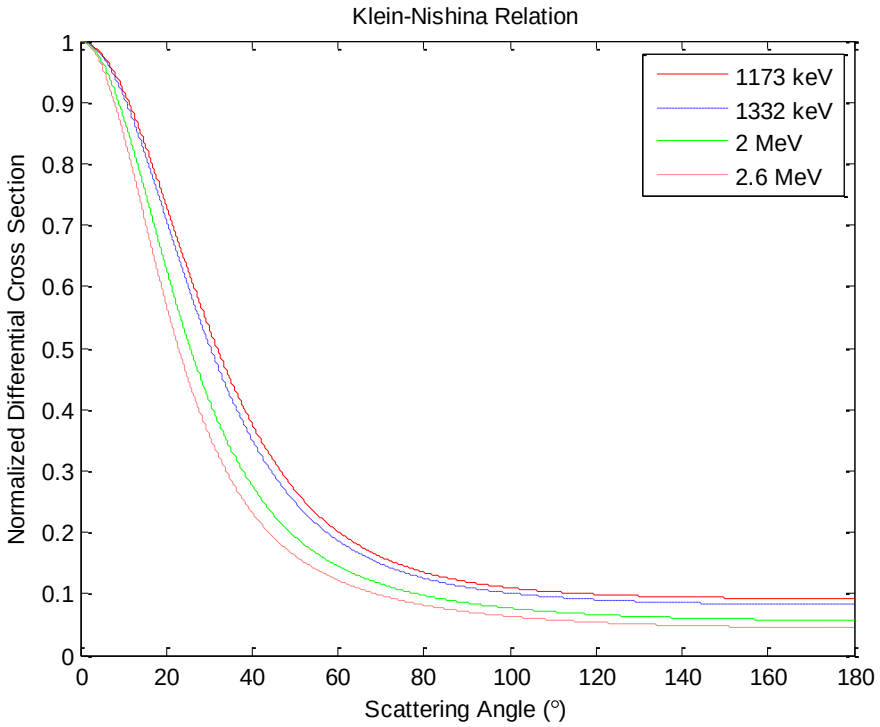

Figure 6. The normalized differential cross section is given by the Klein-Nishina Relation where 0° would indicate a "glancing" scatter and 180 degrees would indicate the incident gamma scattering back in the direction in which it came. Incident gamma energies of 1173 keV, 1332 keV, 2 MeV, and 2.6 MeV are shown [6].

If the angle of scatter is accurately determined, the energy of the incident gamma can be calculated using

$$h\nu = \frac{h\nu'}{1 - \frac{h\nu'}{m_0 c^2}\ 1 - \cos\theta} \tag{1.2}$$

where is the energy of the scattered gamma ray [4, 51]. If the angle of scatter is unknown, as will be the case for this project, where the direction in which the gamma

was incident from is not known, the angle can be calculated using the energy of the incident gamma and scattered gamma [**7**]

$$\cos\theta = 1 + \frac{m_0c^2}{h\nu} - \frac{m_0c^2}{h\nu'} \quad . \tag{1.3}$$

The knowledge of this angle allows the ability to back-project the rim of a cone about the angle into the world. After doing this process for all of the Compton events that occurred during the acquisition, a plot can be created that shows all of the cone rims on one plot. The location where the most intersections occurs is the most likely position of the source. Coincident timing techniques can be used to only select the pulses in the primary detector that occur in coincidence with the pulses in the secondary detector greatly reducing background and the Compton continuum if operating the detector as a Compton spectrometer [4, 325]. A basic Compton camera is shown in Figure 7 [**8**].

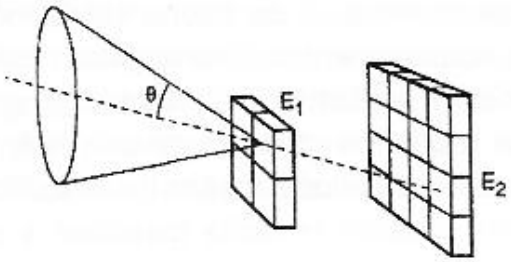

Figure 7. Shown is a simplified block-diagram of a typical Compton camera [8]. The conic projection about the angle is illustrated as well.

I. 6. Objective

The focus of this thesis is to develop and demonstrate a small, low-background, directional pair-production spectrometer capable of implementation in a small RPV. The

spectrometer will utilize four Saint-Gobain BrillanCe 380 detectors. The detectors are well known for their superior energy resolution and durability [**9**].

There are four objectives of this project; one is to determine the difference between a pair-production and Compton event, two is to design and build the directional pair-production spectrometer, three is to characterize the directional efficiency, and four is to create a model to benchmark the current design and provide insight into future designs and applications.

I. 7. General Approach

The development and evaluation of the detection system had multiple steps. First, to produce pair-production and Compton spectrums of a radioactive source, the difference between the two types of events was determined. The implications of not knowing the order of which the pulses from the multiple detectors arrived at the electronics was then determined by performing experiments with a two element detection system. A metric to evaluate pair-production detection efficiency was determined and used to evaluate the directional efficiency of a four element detector. A simulation was then created to benchmark the performance of the four element detector. After verification of the simulation, it was used to estimate the performance of multiple detector geometries and sizes with radioactive sources not present in the laboratory. Lastly, the data sets collected to perform pair-production spectroscopy were used to test the capability of the detector being operated as a Compton camera.

I. 8. Sequence of Presentation

Chapter II provides the reader an introduction to the physics of pair-production and the Compton events and how those and the concept of coincident timing affected the eventual design of the detection system. Chapter III describes the detector, the electronics and the software used in the project. Chapter IV details the experimental and computational procedures used. Chapter V presents the results of the experiments and simulations, and Chapter VI summarizes conclusions and recommendations for further work.

II. Theory

II. 1. Gamma Ray Attenuation

In the scenario of interest, which is to identify and locate SNM from a RPV, gamma rays will interact with the detector after traveling through a given volume of air. For illustrative purposes, a 2 MeV source 100 m from the detection system is used. When propagating through a medium gamma rays can interact either by absorption or by scattering away from the detector. This can be characterized by a fixed probability of occurrence. The sum of these probabilities gives the probability per unit path length that a gamma ray will be removed from consideration. This sum is called the linear attenuation coefficient. According to Knoll, use of the linear attenuation coefficient is rare as it varies with the density of the medium. The mass attenuation coefficient, μ/ρ, is used more often and is shown for dry, sea level air in Figure 8 [4, 53].

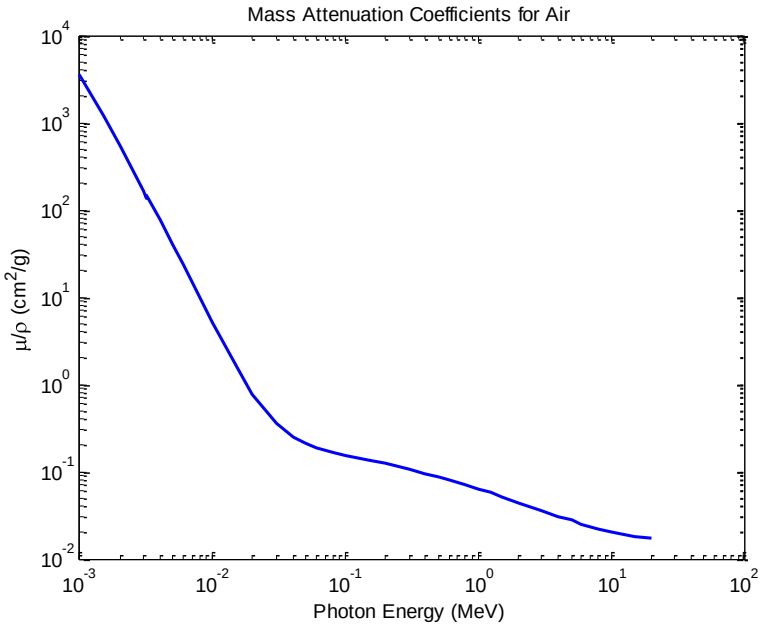

Figure 8. The mass attenuation coefficients for air are shown as a function of photon energy in MeV. At 2 MeV, the attenuation is 0.044 cm^2/g [5].

The mass attenuation coefficient can be used to determine how the attenuation of gamma rays affects detector fluence. The attenuation law for gamma rays is given by:

$$\frac{I}{I_0} = e^{-\mu/\rho\ \rho t} \tag{2.1}$$

where is the number of transmitted gamma rays, is the number of gamma rays incident upon the absorber, and is the mass thickness of the absorber [4, 53]. Using the mass attenuation coefficients from Figure 8 and a nominal 100 m thickness of air, the probability of transmission of uncollided gammas through air can be determined. This is shown in Figure 9.

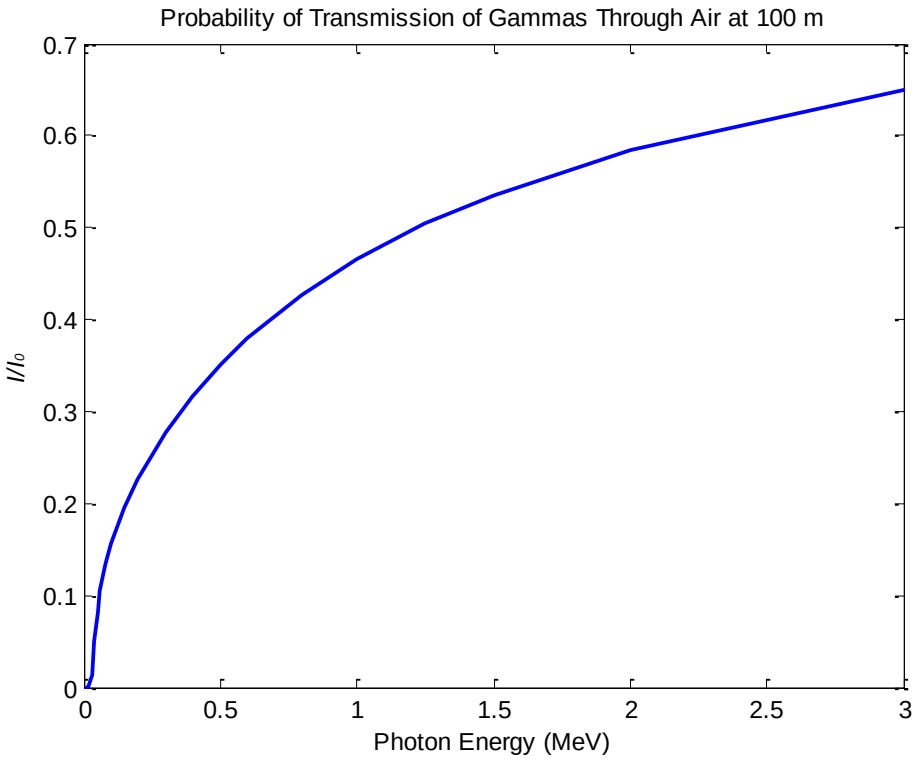

Figure 9. The probability of transmission of uncollided gammas with energy up to and including 3 MeV through 100 m of air is shown. The probability of transmission for a 2 MeV gamma is 58.5%.

It is shown in Figure 9 that gamma rays of higher energy have a higher probability of transmission through air. This relationship makes the case that for stand-off detection a detector optimized for detection of high energy gammas is optimal.

II. 2. Choice of System Geometry

The positions at which the detector elements of the directional pair-production spectrometer are placed relative to each other will be called the system geometry. Efficiency for the detection of pair-production events is of utmost concern for the design of the system geometry. Unfortunately, optimizing the system geometry to detect pair-production events hinders its performance at detecting Compton events due to the forward scattering nature of higher energy gammas that interact through the Compton Effect. Accurate pair-production spectroscopy is more important for this detection system than operation as a Compton camera. Thus, the configuration of the four $LaBr_3$(Ce) element directional pair production spectrometer, shown in Figure 10, is more optimal for the detection of pair-production events.

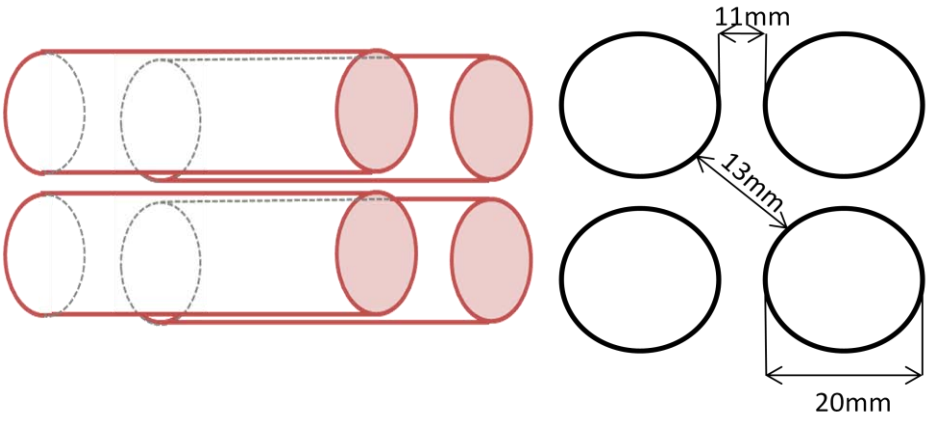

Figure 10. This geometry should have a strong sensitivity to pair-production events. It has the ability to distinguish between pair-production and Compton scatter events to identify a radionuclide through use of coincident timing techniques.

The key to this design is that it can be broken down into eight independent equivalent pair-production spectrometers and eight independent Compton detectors. Each pair of detectors can accomplish both spectroscopy and detection through application of energy filtering. The detectors are spaced as evenly as possible so that each detector will be as close to one neighbor as it is to another. Although this configuration excludes the possibility of detecting both of the 0.511 MeV annihilation photons, greater efficiency is achieved by decreasing the gaps through which the 0.511 MeV photon of interest could escape.

Taking into account the integrated PMTs, there is approximately 11mm separation between the detectors on the adjacent corner. There is approximately 13mm separation between the detectors that are diagonal from each other.

If a gamma ray was to interact in one of the detectors of the array by Compton scattering, then there are potentially three other detectors that its scattered gamma ray can interact with. If a gamma ray with an energy over 1.022 MeV was to interact in one of the detectors by pair-production then the two photons produced from the electron-positron annihilation have three other detectors to interact with.

Coincident techniques are used to provide real-time determination of the incident gamma energy and background reduction. If an event occurs in one detector and another event occurs in a different detector within a certain time window then a Compton event or a pair-production event is known to have occurred. A pair-production event will entail an event in one detector with one coincident 0.511 MeV event in another detector. The incident gamma energy can then be determined by summing the energy of the primary gamma and 1.022 MeV. A Compton event entails an event in one detector followed by a

coincident event in another detector with an energy other than 0.511 MeV. The energy of the incident gamma can be determined by summing the energies in both detectors. There are some cases that interact by means of Compton scattering and incident detector deposition or the scattered gamma will be 0.511 MeV. For the purposes of this research, a coincident event with an energy of 0.511 MeV is designated a pair-production event. The use of these two methods greatly reduces the background in the spectra collected ruling out nearly all but the events mentioned above. Figure 11 shows the possibilities for Compton scatter using a 5% resolution about 0.511 MeV that results in a secondary gamma with an energy between 0.485 MeV and 0.542 MeV with their associated incident gamma energy and angle of scatter. The events shown below will appear in the pair-production spectra as peaks that are 0.511 MeV above the full energy peak of the source.

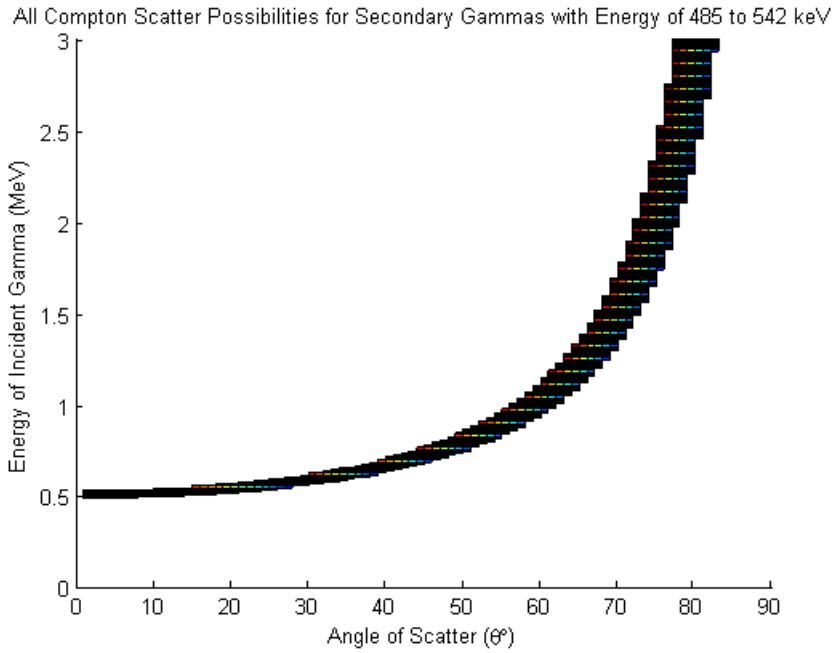

Figure 11. All Compton scatter possibilities are shown for incident gammas of energy 0 MeV to 3 MeV with angles of scatter between 0° and 90° that result in secondary gammas with energies between 485 keV and 542 keV.

It should also be mentioned that these two interactions contribute differently to the overall effectiveness of the detector. The pair-production events will contribute spectroscopic information. Through pair-production spectroscopy, isotopes will be identified. The Compton scattering events will contribute detection information to the system, through operation as a Compton camera. The design will allow the location of events to be determined within a degree of certainty to be characterized later. The ultimate goal is for the system to function as a pair-production spectrometer and Compton camera simultaneously.

II. 2. A. Coincident Timing

Coincidence timing is a crucial part of the operation of this detector. A coincidence detection system is characterized by a coincidence window following the occurrence of an initial event, during which another event will be accepted by the detection system as coincident with the initial. Coincident timing allows for the possibility of determining the difference between a photoelectric event and a Compton or pair-production event. Two events which are not truly coincident can occur within the coincidence window and be indistinguishable by the detection system from true coincidence [**10**]. The maximum time of flight for the photons between detectors and electron stopping times can be calculated to give an ideal coincidence window. If electronics with an adequate sampling rate were used, the chance coincident rate could greatly be decreased by decreasing the coincidence window to near the time for the interaction processes to occur. Chance coincidence is calculated by first measuring the singles' rates for both detectors in a two detector system. The chance coincidence rate can then be calculated using

$$r_{ch} = 2\tau r_{ch1} r_{ch2} \tag{2.2}$$

where τ is the coincidence window, r_{ch1} is the singles rate for channel 1, and r_{ch2} is the singles rate for channel 2 [4, 670]. Chance coincidence rates were measured for background, a 6.7 μCi Co-60 source at 30 cm, and a 15.5 mCi Co-60 source at 10 ft. The results are shown in Table 2.

Table 2. Chance Coincidence Measurements for Various Sources

Measured Source	Singles Rate Channel 1 (cps)	Singles Rate Channel 2 (cps)	Coincidence Window (ns)	Chance Coincidence Rate
Background	9.26	9.41	100	1.74E-05
6.7 μCi at 30cm	78.59	78.07	100	1.23E-03
15.5 mCi at 10ft	1830.4	1799.1	100	6.59E-01

Ideal timing window calculations were completed for both Compton scattered and pair-production scenarios. The scenario for Compton scattering assumes that a gamma enters the primary detector at location A and interacts by Compton scattering immediately. It was assumed that the longest path possible the scattered gamma could travel was 7.28 cm to location B of the adjacent secondary detector at an angle of 46.67°. Locations A and B are shown in Figure 12.

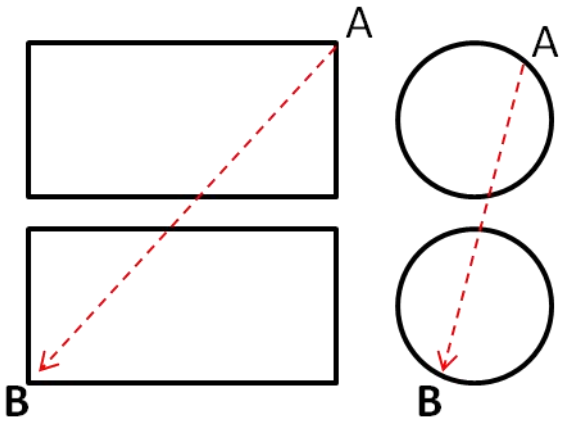

Figure 12. Shown are the locations, A and B, that represent the longest distance a photon could travel between the two detectors.

The energy imparted to the recoil electron can be calculated using

$$E_{re} = \frac{h\nu}{1+\frac{h\nu}{m_0c^2}\ 1-\cos\theta}\ \frac{h\nu}{m_0c^2}\ 1-\cos\theta \tag{2.3}$$

where is the energy imparted to the recoil electron, is the energy of the incident gamma ray, is 0.511 MeV, and is the angle of scatter of the scattered gamma ray [11, 179]. The stopping time of electrons and positrons is determined by first calculating the collisional stopping power

$$\left(-\frac{dE}{dx}\right)^{\pm}_{col} = \frac{4\pi k_0^2 e^4 n}{mc^2\beta^2}\left[\ln\frac{mc^2\tau\sqrt{\tau+2}}{\sqrt{2}I} + F^{\pm}(\beta)\right] \tag{2.4}$$

where

$$F^{-}\ \beta\ = \frac{1-\beta^2}{2}\left[1+\frac{\tau^2}{8}-(2\tau+1)\ln 2\right] \tag{2.5}$$

is used for electrons and

$$F^{+}\ \beta\ = \ln 2 - \frac{\beta^2}{24}\left[23 + \frac{14}{\tau+2} + \frac{10}{\tau+2^{\ 2}} + \frac{4}{\tau+2^{\ 3}}\right] \tag{2.6}$$

for positrons [11, 140]. In the preceding three equations, is 8.99x10^9 Nm^{2}C^{-2}, is the magnitude of the electron charge, is the number of electrons per unit volume in the medium, is the effective atomic number of the medium, is the electron rest mass, is the speed of light in vacuum, is the velocity of the electron or positron relative to the speed of light, is the mean excitation energy of the medium, and is the kinetic energy of the positron or electron divided by . The approximate relation of the radiative stopping power and collisional stopping power is given by

$$\left(-\frac{dE}{dx}\right)^{\pm}_{rad} \sim \left(-\frac{dE}{dx}\right)^{\pm}_{col} \frac{Z_{eff}E}{800} \tag{2.7}$$

where is the energy of the electron or positron in MeV [11, 145]. The stopping rate is the velocity of the positron or electron multiplied by the total stopping power which is merely the sum of the radiative and collisional stopping powers. Dividing this product by the energy of the particle gives the stopping time [11, 150]. Travel times of the photons, electrons, and if applicable, positrons, to travel and stop were calculated for Compton scattering and pair-production. These times represent the ideal timing windows and are tabulated for various energies in Table 3.

Table 3. Total Process Time for Compton Scattering and Pair Production events at various Gamma Ray Energies in $LaBr_3(Ce)$

Interaction Type	Primary Gamma Ray Energy (MeV)	Travel Time (ns)
Compton Scatter	2.6	1.83
Pair Production	2.6	1.59
Compton Scatter	2	1.18
Pair Production	2	0.93
Compton Scatter	1.332	0.77
Pair Production	1.332	0.39
Compton Scatter	1.173	0.78
Pair Production	1.173	0.20

II. 3. Pair-Production Detection Efficiency Calculations

A method for estimating the pair-production detection efficiency was devised for this project. This is calculated much in the same way that intrinsic peak efficiency is calculated. Intrinsic peak efficiency is a metric to quantify the performance of a detector in recording the radiation quanta emitted from a source in the solid angle subtended by the detector at the source position. First, the number of gammas that are emitted in the solid angle subtended by the detector is calculated. At 0°, when the detector is axially aligned with the source as shown in Figure 13, the solid angle can be calculated using the simple point source approximation along the axis of a right circular cylindrical detector. Multiplying that solid angle by the number of detectors that are being used approximates the solid angle for the entire detection system. This is represented with

$$\Omega = 2\pi N\left(1-\frac{d}{\sqrt{d^2+a^2}}\right) \tag{2.8}$$

where is the solid angle subtended by the detector, N is the number of detectors, is the distance from the source to the detector face, and is the radius of the detector [4, 118].

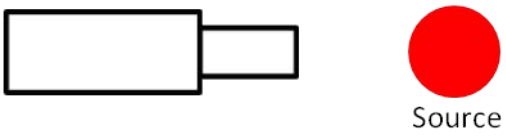

Figure 13. Orientation of a detector and the source when axially aligned (0°).

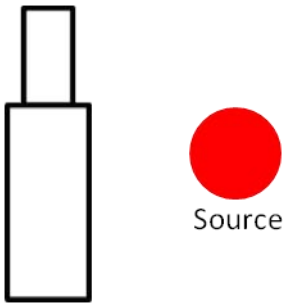

Figure 14. Orientation of a detector and the source when the broad side of the detector is facing the source (90°).

When the detectors are viewed at a 90° orientation, with the broad side of the detector element facing the source as shown in Figure 14, the right circular cylinders as seen by the source are 2cm x 5cm rectangles. The solid angle subtended by a rectangle is represented by

$$\Omega = 4Nz \int_0^{\frac{w}{2}} dx \int_0^{\frac{H}{2}} dy \frac{1}{\left(x^2 + y^2 + z^2\right)^{\frac{3}{2}}} \tag{2.9}$$

where is the width of the detector, is the height of the detector, and is the distance from the source to the detector [**12**]. For the geometry used for this project, the detectors in the rear do not contribute to the solid angle as they are not seen by the source for this

approximation. The solid angle for the detection system for this orientation can be obtained by multiplying the solid angle for one detector by 2; the number of detectors seen by the source. Next, the number of gammas that are not attenuated by the thickness of air between the source and the detector is calculated using (2.1).

Those gammas that are able to make it through the air are then used to determine the number of gammas that will interact by pair-production in the detector using the linear attenuation coefficient and equation (2.1). The constants used in the calculations are tabulated in Table 4. The dimensions of the detector and its distance from the source used for the calculations for the 0° orientation are in 0 and for the 90° orientation are in 0. The total number of gamma rays that should interact by pair-production per hour in the four element detector for both orientations are tabulated in 0 and for the two element detector in 0.

Table 4. Constants Used for Calculations

Energy (keV)	1173	1332	2614
Branching Ratio	0.9986	0.9998	1.00
Source Activity (μCi)	6.7	6.7	6.7
μ/ρ (Air) (cm^2/g)	0.05875	0.05502	0.03855
μ/ρ for Pair-Production ($LaBr_3$) (cm^2/g)	5.25E-05	2.78E-04	4.61E-04
ρ (Air) (g/cm^3)	1.21E-03	1.21E-03	1.21E-03
ρ (LaBr3) (g/cm^3)	5.06	5.06	5.06

Table 5. Dimensions Used for Calculations with Orientation = 0°

Constant	Value
a (cm)	1
d (cm)	27.5, 57.5, 87.5
t (cm)	5

Table 6. Dimensions Used for Calculations with Orientation = 90°

Constant	Value
W (cm)	2
H (cm)	5
z (cm)	27.5, 57.5, 87.5
t (cm)	4

Table 7. Number of Gammas that Interact by Pair-Production per Hour in Four Element Detector

	Orientation (°)	0			90		
	Energy (keV)	1173	1332	2614	1173	1332	2614
d (cm)	27.5	1559	8239	129517	1978	10458	166177
	57.5	356	1882	29606	453	2396	38097
	87.5	153	811	12769	195	1033	16439

Table 8. Number of Gammas that Interact by Pair-Production per Hour in Two Element Detector

	Orientation (°)	0		
	Energy (keV)	1173	1332	2614
d (cm)	27.5	779	4119	64759

The air between the front and rear detectors in the 90° orientation was treated as though it were vacuum and could not attenuate the gammas further. For the rest of this document, $\phi=0°$ refers to the detectors pointing at a source that is on the same horizontal plane. $\phi=45°$ refers to the detectors pointing at a source that is slightly elevated above the plane the detectors are in. The two ϕ orientations are shown in Figure 15.

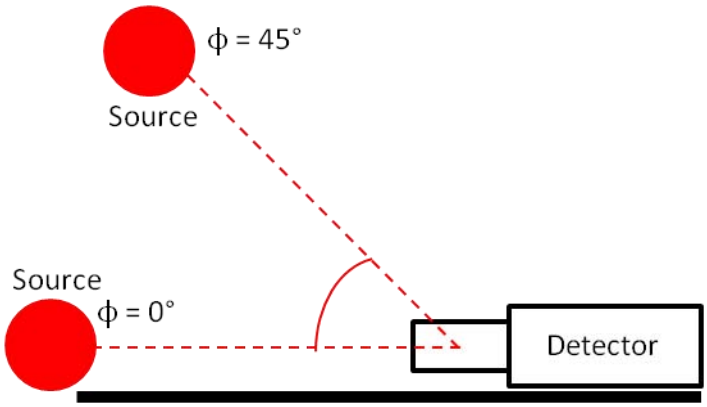

Figure 15. Shown are the two ϕ angles with respect to the detector used.

The solid angles and therefore the efficiency calculations were only completed for the 0° and 90° orientations when ϕ = 0° as the solid angles subtended by the detector as seen by the source when the orientation at other angles and when ϕ = 45° are much too complicated for this simple approximation. The efficiencies reported for the 0° and 90° orientations when ϕ = 0° are simply the number of counts in the full energy peak of interest divided by the theoretical number of pair-production events that should have occurred.

II. 3. A. Extended Pair-Production Efficiency Calculations

These efficiency calculations can be taken further if more constraints are made. The geometry used for the following calculations is the 4 element detector. The first constraint is that all of the pair-production interactions occur in the center of each detector. Secondly, the calculations are constrained to only allow the annihilation photons to propagate in the directions as shown in Figure 16.

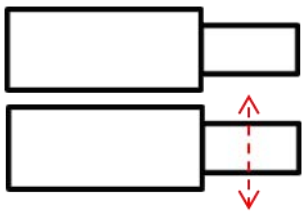

Figure 16. Direction that annihilation photons are constrained to propagate after pair-production for efficiency calculations.

The third constraint is that the annihilation photons must escape the detector they are born in without interacting in the detector volume. Lastly, one of the annihilation photons must interact by photoelectric effect in another detector. The sum of the angles of propagation that would allow this constraint is approximately 105.2° or 58.46% of 180°. This is assumed to occur within a 2 cm cross-section of the detector volume. The cross-sections for the calculations were given in Table 1.

It was found that for a given number of pair-production events, 58.46% of one of the annihilation photons would propagate in the direction of another detector. Only 34.87% of those will escape the detector they are born in without interaction. Lastly, 88.90% of those remaining annihilation photons will interact by photoelectric effect in another detector and would have a chance of being detected as a pair-production event. If a given number of gammas interact by pair-production in one of the detectors of a 4 element array, this constrained method of calculating efficiency allows 18.12% of those to be given the chance of being detected as a pair-production event. This is a very simplified method and represents only a fraction of the annihilation photons that would interact in the adjacent detectors. A more rigorous examination to find the locations of interactions and directions of propagation would provide much more accurate efficiency

calculations. 0 has been recalculated to reflect the 18.12% probability and is shown below in Table 9.

Table 9. Number of Pair-Produced Annihilation Photons that Deposit in another Detector of Four Element Array

	Orientation (°)	0			90		
	Energy (keV)	1173	1332	2614	1173	1332	2614
d (cm)	27.5	282	1492	23468	358	1895	30111
	57.5	65	341	5364	82	434	6903
	87.5	28	147	2313	35	187	2979

II. 4. Chance Compton Scattering Detection Efficiency Calculations

Chance Compton events are gamma rays that undergo Compton scattering and either the initial or secondary gamma ray results in 511 keV being deposited in a detector. The efficiency for detecting the chance Compton scattered gamma rays can be calculated in a manner that is very similar to that of the pair-production efficiency. This method uses the same geometries and solid angles but different cross-sections. Of interest are those events that deposit within 5% of 511 keV in either the incident or secondary detectors. The probability of an event depositing 511 keV in either detector can be calculated using the normalized Klein-Nishina relation as shown in

$$P_{E_0} = \sum_{x}^{y} \frac{\frac{d\sigma}{d\Omega_{E_0}}}{\sum \frac{d\sigma}{d\Omega_{E_0}}} \qquad (2.10)$$

where P_{E_0} is the probability of scattering within 5% of 511 keV for that incident energy, $\frac{d\sigma}{d\Omega_{E_0}}$ is the result found in the Klein-Nishina relation for that incident energy, and *x* and

y are the lower and upper bound, respectively, of the Compton scattering angle to fall within 5% of 511 keV. The lower and upper bounds for the incident Co-60 gamma rays are calculated using the desired scattered gamma energy in equation (2.10) are shown in Table 10.

Table 10. Compton Scattered Gamma Characteristics that Result in 511 keV

	511 + 0.05*511		511 - 0.05*511		E_0 - 511 - 0.05*511		E_0 - 511 + 0.05*511	
E_0 (keV)	Energy (keV)	Angle (°)	Energy (keV)	Angle (°)	Energy (keV)	Angle (°)	Energy (keV)	Angle (°)
1173	536.55	61.10	485.45	67.48	636.45	50.75	687.55	46.17
1332	536.55	64.45	485.45	70.67	795.45	42.16	846.55	38.74
2614	536.55	75.93	485.45	81.79	2077.45	18.28	2128.55	17.17

The total probability found for the 1173 keV peak to Compton scatter in the manner described above is 5.77%, for the 1332 keV peak it is 6.41%, and for the 2614 keV peak it is 5.48%. Using the method described in section II. 3, the number of gammas that interact by a chance Compton Effect was calculated. The results are shown in Table 11 for a 4 element detector and in Table 12 for a 2 element detector. The chance Compton Effect efficiencies in the results section are calculated the same as the pair-production efficiencies.

Table 11. Number of Gammas that Interact by a Chance Compton Effect per Hour in four Element Detector

	Orientation (°)	0			90		
	Energy (keV)	1173	1332	2614	1173	1332	2614
d (cm)	27.5	48696	52417	35996	68485	73341	49057
	57.5	11123	11975	8228	15689	16804	11247
	87.5	4794	5162	3549	6765	7247	4853

Table 12. Number of Gammas that Interact by a Chance Compton Effect per Hour in two Element Detector

	Orientation (°)	0		
	Energy (keV)	1173	1332	2614
d (cm)	27.5	24348	26208	17998

If the electronics allowed the ability to distinguish the ordering of the pulses, so that the post-processing code only allowed those events whose secondary gamma was within 5% of 511 keV, the number of chance Comptons would decrease greatly. If the 511 keV photon is recorded in the initial detector, the interaction cannot be pair-production. All of these events can be discarded. Equation (2.10) can be used to calculate the probability for the 1173 keV peak to Compton scatter in this manner is 2.71%, for the 1332 keV peak it is 2.46%, and for the 2614 keV peak it is 1.70%. The ability to distinguish the ordering of the incoming pulses results in a 46.97 % reduction in the chance Compton counts from the 1173 keV peak, 38.38% from the 1332 keV peak, and 31.04% from the 2614 keV peak. The results from those calculations are shown in Table 13 for the four element detector and 0 for the two element detector.

Table 13. Number of Gammas that Interact by a Chance Compton Effect per Hour in 4 Element Detector with Pulse Order Information

	Orientation (°)	0			90		
	Energy (keV)	1173	1332	2614	1173	1332	2614
d (cm)	27.5	22871	20116	11167	32165	28146	15218
	57.5	5224	4596	2553	7369	6449	3489
	87.5	2252	1981	1101	3177	2781	1505

Table 14. Number of Gammas that Interact by a Chance Compton Effect per Hour in 2 Element Detector with Pulse Order Information

	Orientation (°)	0		
	Energy (keV)	1173	1332	2614
d (cm)	27.5	11435	10058	5583

II. 5. Compton Camera Images

The design of the detection system used for this project is far from the ideal design for operation as a Compton camera. Typically, Compton cameras use detectors that are not all in the same plane and are pixilated or position sensitive. Figure 17 shows the design and concept of a typical Compton camera which uses two planar pixilated detectors [**13**]. Using a source such as illustrated in Figure 17 (a), the photons emitted undergo Compton scattering in Detector 1 and are absorbed in Detector 2. The angle of scatter can be calculated using equation 1.3. The positions of interaction are known by the pixel location on each detector. The location of the source can be placed on the rim of a cone whose apex is at the point of Compton scatter with an angle of θ and axis of symmetry through both interaction positions. The number of potential vectors that the conical projections can be projected on is limited only by the spatial resolution of the detectors.

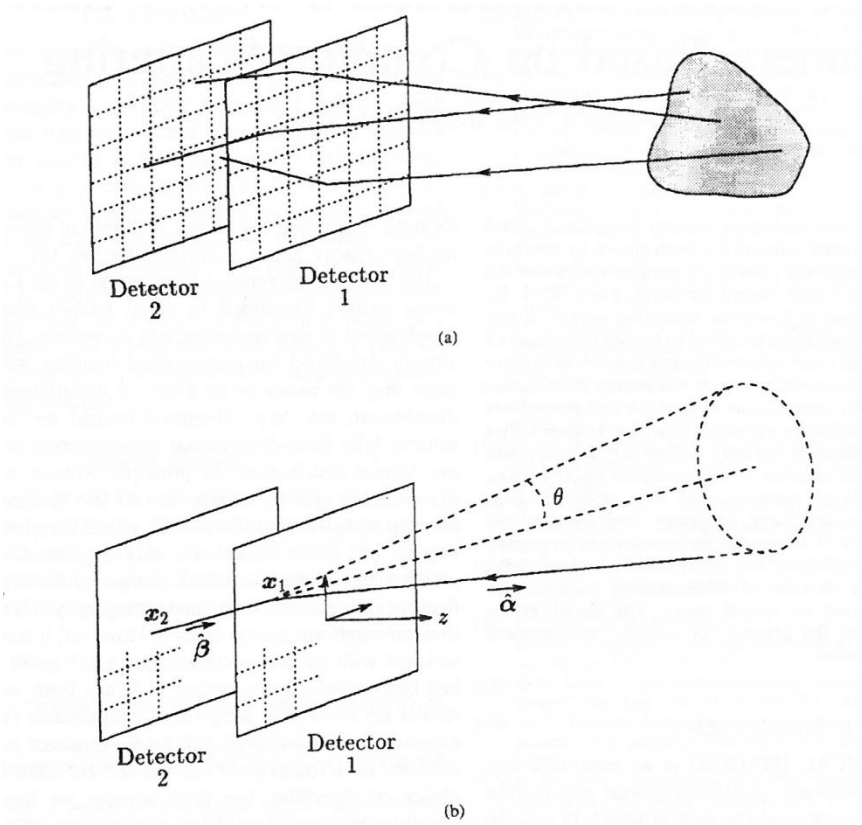

Figure 17. (a) The photons emitted from the source Compton scatter in Detector 1 and are fully absorbed in Detector 2. (b) The location of the source can be backprojected on the rim of a cone whose apex is located at the position of Compton scatter with angle θ and axis of symmetry through both interaction locations [13**].**

The detectors used in this project are neither planar or position sensitive, hence they must be must be modeled as point detectors for the post-processing code. Doing so greatly reduces the accuracy of the Compton camera by only allowing source locations that are perpendicular to the detector. The cone projections expected for this detector are shown in Figure 18. There are 12 cones however the four at the 0°, 90°, 180°, and 270° locations are duplicated. Thus, when plotted there will only be 8 circles. Figure 19shows a profile view of the conical backprojections that shows the inability to get cones, with the exception of those particles with very large angles of scatter, in front of or behind the detectors.

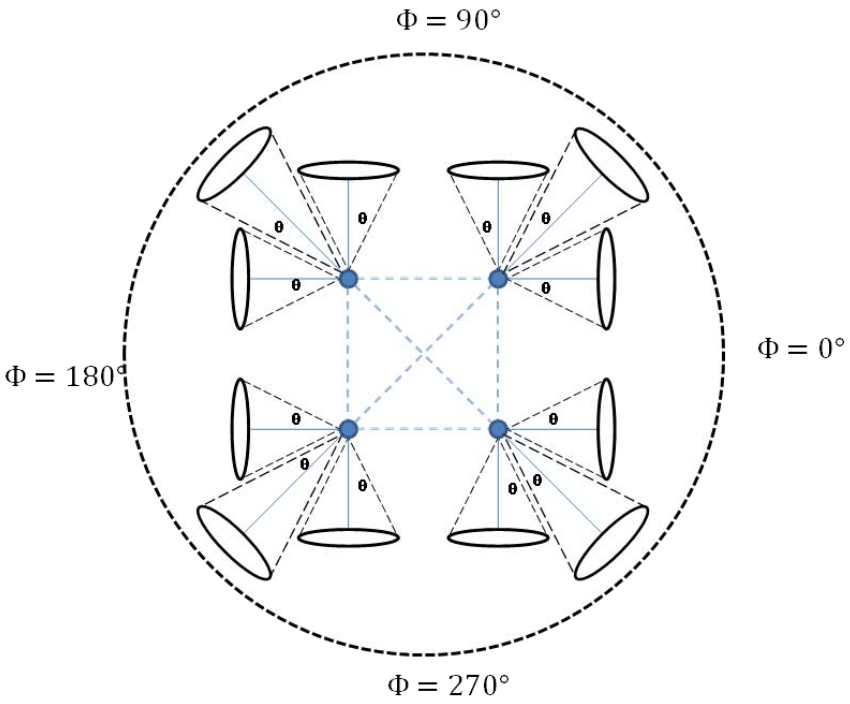

Figure 18. Shown are the conical projections of where the source can be located when modeling the detectors as point detectors for operation as a Compton camera. There are eight cones. The camera should give more accurate directional information for the sources that are in front of the detector.

Figure 19. Shown are the conical backprojections expected from the geometry used for this project. No cones are projected to the right or left of the detectors in the figure.

The best performance that can be expected out of this camera is to be able to tell if the source is to the left, right, above, below, or off to the corners. Sources that are directly in front of or behind the detector will be very difficult, if not impossible to detect with any accuracy in this configuration. This is not to say that the Compton camera will not give an answer, it is just that it may be incorrect. The only direction that will be available for sources that are in front or behind would be achieved if they are given an elevation of some sort.

II. 6. Expected Spectrum Features

$LaBr_3$(Ce) detectors complicate pair-production spectroscopy. Of particular concern with these detectors used for timing operations, is the inherent radioactivity of lanthanum. La-138 naturally occurs with an abundance of 0.09%. The decay scheme of La-138 is shown in Figure 20. The below scheme shows the occurrence of 1.436 MeV and 0.789 MeV gammas. It should be noted that the decay of La-138 to Ce-138 through beta minus emission occurs in coincidence with the beta minus having max energy of 0.255 MeV [**9**].

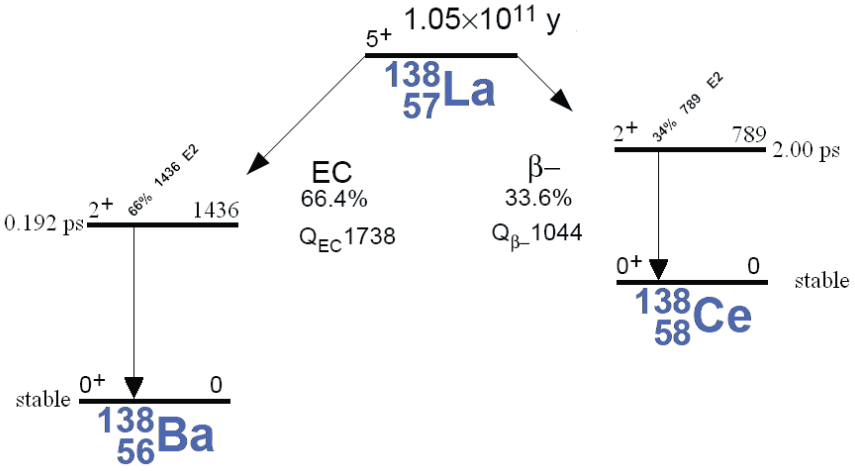

Figure 20. La-138 decay scheme. [9]

The occurrences of La-138 decay in the $LaBr_3(Ce)$ have been characterized by Saint-Gobain, for a 1.5"x1.5" detector. While the detectors are not the same size, the results are assumed to be similar and will not be duplicated in this study for the individual detectors. Their findings are tabulated in [**9**]. The features up from 0 keV to 1600 keV are from the La-138 while the features attributed to the alpha decays as denoted in [**9**], are from Ac-227 contaminates [**9**]. The corresponding background pulse height spectrum provided by Saint-Gobain is shown in Figure 21.

Table 15. Background Count Rates from 1.5"x1.5" Detector [9]

Counts Per Second/cc	Characteristic
0.226	0-255 keV beta continuum
0.065	790 keV-1000 keV
0.068	1436 gamma peak
0.034	Alphas above 1600 keV

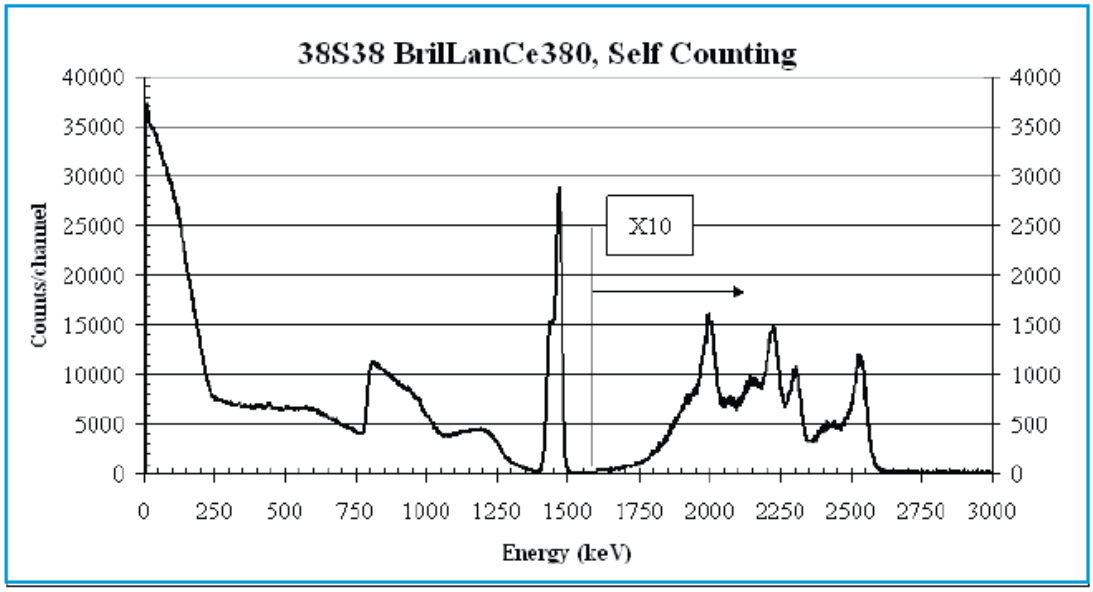

Figure 21. Shown is a self-counting background spectrum supplied by Saint-Gobain [9].

The detectors used in this project have a volume of 15.71 cm^3. Using this volume, the numbers of expected decays for each detector every second are shown in Table 16.

Table 16. Background Count Rates from 1 Detector per Second [9]

Counts Per Second	Characteristic
3.550	0-255 keV beta continuum
1.021	790 keV-1000 keV
1.068	1436 gamma peak
0.534	Alphas above 1600 keV

The intrinsic radioactivity of La-138 greatly complicates the pair-production spectra that are presented in Chapter V. Table 17 shows all of the possible features and peaks that could be present in a Co-60 pair-production spectrum. The interactions in Detector #1 are seen as the pair-production interaction by the electronics whether it really is pair-production or not because another detector records a 511 keV gamma ray within

the designated coincidence window. The final energy is the addition of the energy recorded in Detector #1 and 1022 keV, the annihilation photons energy.

Table 17. Features of Co-60 Pair-Production Spectra

Detector # 1		Detector # 2		Final Energy (keV)
Interaction	Energy (keV)	Interaction	Energy (keV)	
Pair-Production from 1173	151	Pair-Production from 1173	511	1173
Pair-Production from 1332	310	Pair-Production from 1332	511	1332
Chance Compton from 1173	662	Chance Compton from 1173	511	1684
Chance Compton from 1332	821	Chance Compton from 1332	511	1843
Pair-Production from La-138 1436	414	Pair-Production from 1436	511	1436
Chance Compton from La-138 1436	925	Chance Compton from La-138 1436	511	1947
Beta Continuum Contribution	0-255	Pair-Production 1173/1332/1436	511	1022 - 1277
Beta Continuum Contribution	0-255	Chance Compton 1173/1332/1436	511	1022 - 1277

III. Equipment

III. 1. $LaBr_3$(Ce) Detectors

The proposed array uses four Saint-Gobain BrilLanCe 380 detectors with $LaBr_3$(Ce) crystals configured for coincidence detection of pair-production and Compton scatter events. Each $LaBr_3$(Ce) detector consists of a 20mm x 51mm crystals and integrated photomultiplier tube (PMT). The detectors used in this experiment are shown in Figure 22. There were two different operating voltages used for the detectors for this project. The detector high voltages used while operating the oscilloscope as a data acquisition (DAQ) system are given in Table 18. These detector operating voltages were found by adjusting the voltage to have the 1.332 MeV gamma emitted by Co-60 to have a pulse height of approximately 230 mV out of the PMT. The only adjustment to spread out the pulse height spectrum that was available when using the oscilloscope DAQ was the applied high voltage. This is not the case with the XIA and Gamma Vision DAQ's. The software or other NIM electronics supplied with the other DAQ systems have adjustments that can spread out their pulse height spectra. For these systems, the applied high voltages that were given by Saint-Gobain on the individual specification sheets for each detector when measuring Cs-137 were used. The manufacturer specified voltages are given in 0.

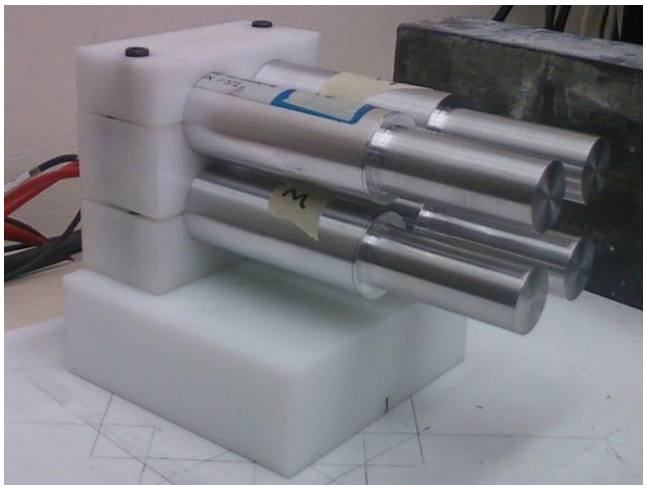

Figure 22. The detectors to be used for this detection system are Saint-Gobain manufactured BrilLanCe 380 series 20mm x 51mm $LaBr_3$(Ce) crystals with integrated PMTs. $LaBr_3$(Ce) scintillator material is known for its exceptional energy resolution, fast emission and excellent linearity.

Table 18. Detector Operating Voltages for use with the Oscilloscope

Detector S/N	HV Setting
570	800
572	852
902	1070
903	1018

Table 19. Detector Operating Voltages for use with Gamma Vision and XIA Electronics

Detector S/N	HV Setting
570	679
572	778
902	823
903	777

Resolution is determined by finding the full width at half maximum (FWHM) of the FEP and dividing by the energy of the FEP. The energy resolution and intrinsic efficiency of the detectors were characterized using Co-57 (0.122 MeV), Sr-85 (0.514 MeV), Cs-137 (0.662 MeV), and Co-60 (1.173 MeV and 1.332 MeV) sources. This was done using the NIM (Nuclear Instrumentation Module) electronics configured as shown

in Figure 23 and the spectral analysis code GammaVision32. The settings used to make the resolution and intrinsic efficiency measurements are shown in Table 20. The settings for the NIM electronics were chosen by forcing the 662 keV FEP to channel 800 of 2048 channels.

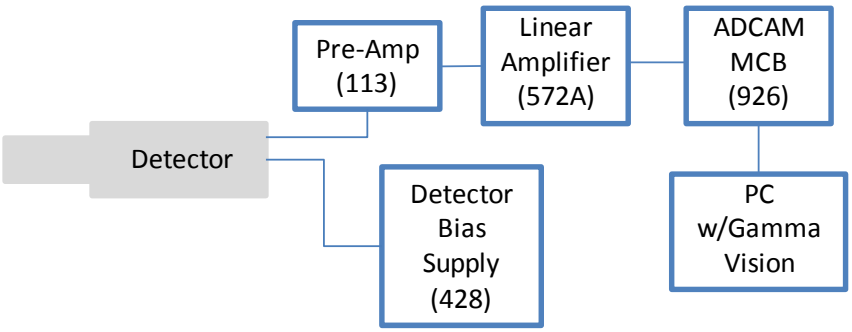

Figure 23. Baseline resolution and efficiency measurements were taken with ORTEC NIM electronics and processed with GammaVision32.

Table 20. Individual Detector Settings for NIM Electronics Measurements

Detector	Linear Amplifier Coarse Gain	Linear Amplifier Fine Gain	Linear Amplifier Shaping Time (μs)	Pre-Amplifier Shaping Time (pf)
P570	50	10.34	2	0
P572	50	8.83	2	0
P902	100	8.92	2	0
P903	100	9.38	2	0

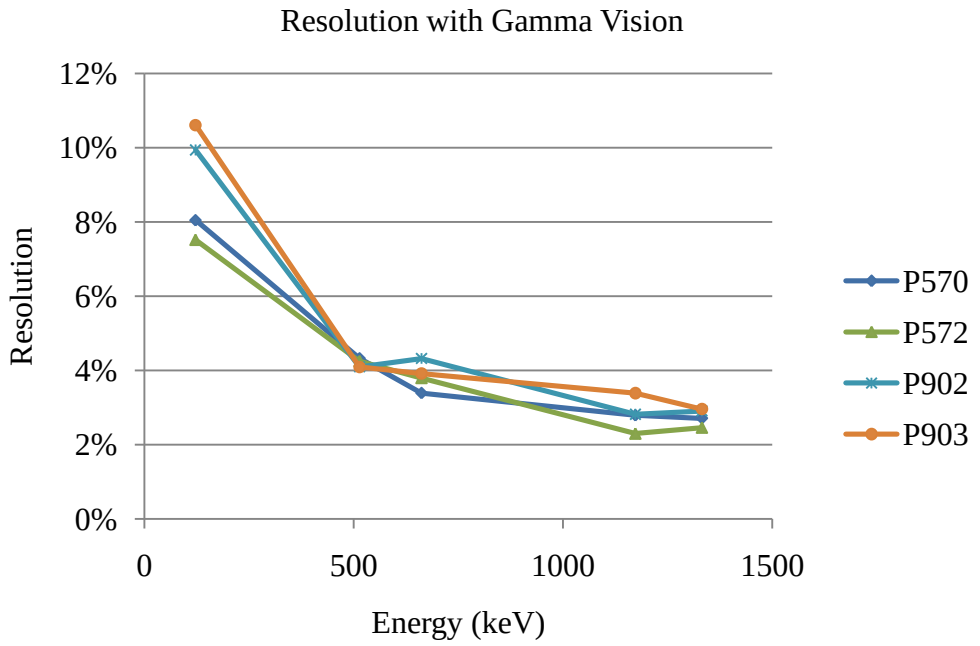

Figure 24. The resolution as computed by the spectra acquisition program, Gamma Vision, is shown for each detector. Spectra were taken for 300 seconds with the sources placed 7 cm from the detectors.

Spectra were taken for 300 seconds with the sources placed 7 cm away from each of the 6 detectors independently. The results are shown above for resolution in Figure 24 and below for intrinsic efficiency in Figure 25. The results for the commonly quoted 662 keV photopeak associated with Cs-137 and the 514 kV photopeak associated with Sr-85 are tabulated in Table 21.

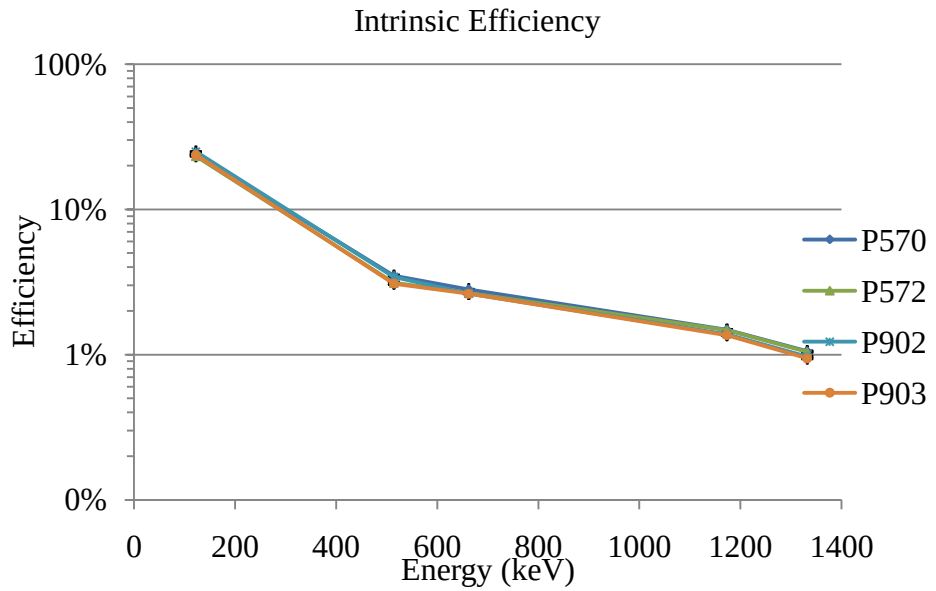

Figure 25. The intrinsic efficiencies for the detectors are shown. Sources were modeled as disks. Spectra were taken for 300 seconds with the sources placed 7 cm form the detectors.

Table 21. Detector Resolution and Intrinsic Efficiency at 662 keV and 514 keV

	662 keV (Cs-137)		514 keV (Sr-85)	
Detector	Resolution	Intrinsic	Resolution	Intrinsic
P570	4.27%	2.93%	4.33%	3.49%
P572	3.78%	2.76%	4.25%	3.12%
P902	4.32%	2.74%	4.10%	3.43%
P903	3.92%	2.73%	4.09%	3.09%

A commonly quoted characteristic of detector performance is the peak-to-Compton ratio. This ratio is defined as the ratio of the count in the highest FEP channel to the count in a channel of the Compton continuum for that peak. The channel in the Compton continuum is taken from the flat portion. The peak-to-Compton ratio is commonly quoted for 662 keV and 1332 keV. The location of the sample taken from the Compton continuum is between 358 keV and 382 keV for the 662 keV FEP and between 1040 keV and 1096 keV for the 1332 keV FEP [4, 428] [**14**]. The peak-to-Compton

ratios for the detectors used in this project are given for 662 keV and 1332 keV in Table 22.

Table 22. Peak-to-Compton Ratios for at 662 keV and 1332 keV

Detector	662 keV	1332 keV
P570	2.82	1.13
P572	2.20	0.87
P902	2.98	1.21
P903	2.93	1.53

Published peak-to-Compton ratios for $LaBr_3(Ce)$ are approximately 5.8 at 662 keV [**15**] . The decreased peak-to-Compton ratio found for these detectors is likely due to lower efficiency detectors used in this project than in the cited literature.

III. 2. Data Acquisition

III. 2. A. Tektronics DPO7104 Oscilloscope

One of the data acquisition systems used for this research consists of a PC with MATLAB and Tektronix's VISA (TekVISA) application installed. The PC is connected to a Tektronix DPO7104 oscilloscope by way of an Ethernet cable as shown in Figure 26.

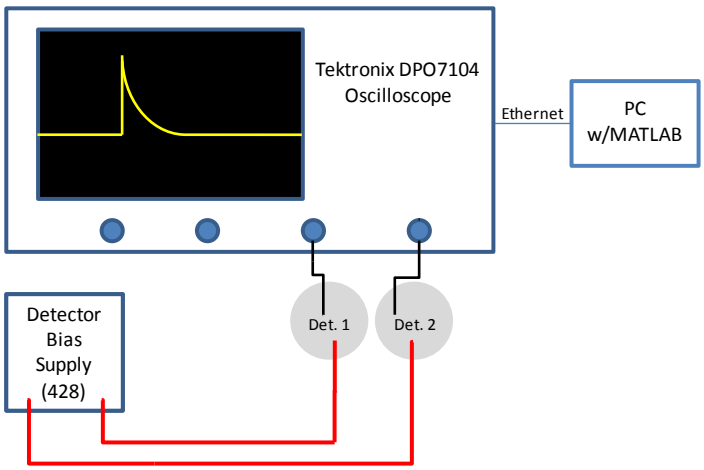

Figure 26. Tektronix DPO7104 Data Acquisition system.

The oscilloscope has the TekVISA application installed as well. The DPO7104 oscilloscope is a very capable instrument with the ability to have four input channels, 1 GHz bandwidth, up to 40 GS/s, a record length of up to 250 million points per second, and to operate in FastFrame mode. FastFrame mode allows the operator to take multiple measurements on the scope and fill up the buffer, and then dump the buffer all at once to the recording PC. A MATLAB script (Appendix A) controls the oscilloscope and reads the waveforms from the oscilloscope and stores them in arrays. This setup provides excellent timing resolution that would be difficult to improve upon. An example capture of a coincident event is shown in Figure 27.

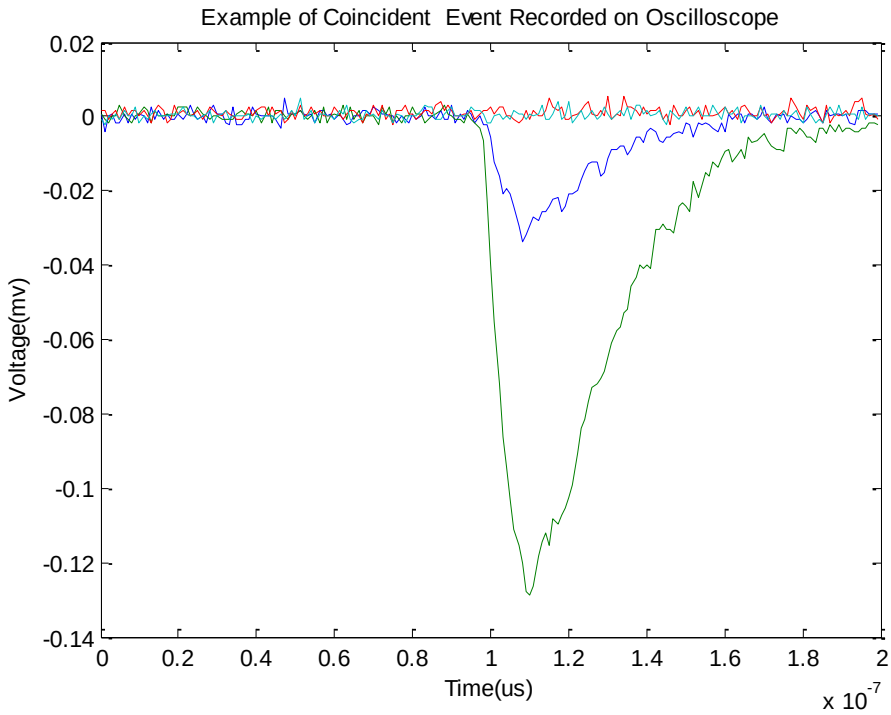

Figure 27. Example of a coincident Compton event in four detectors recorded by the Tektronix DPO7104 Oscilloscope as seen by the PC running MATLAB.

The oscilloscope data acquisition system presented here is used in experiments outlined in section IV. 4, with results presented in section V. 2.

III. 2. B. XIA CAMAC Standard Electronics

The other data acquisition system used for this project is a CAMAC standard system. All CAMAC instruments are installed in a CAMAC crate. The dataway in the rear of the crate provides the method for the control signals, digital data, and power to get to and from the electronics. This project used one X-Ray Instrumentation Associates (XIA) DGF model 4C CAMAC cards (serial number is 1355) for digital data collection. The DGF-4C is a 16-bit four-channel all-digital waveform acquisition and spectrometer card [**16**]. The detector signals are digitized by analog-to-digital converters and then

analyzed by the digital signal processor for pulse shape analysis. The waveforms, timestamps, and pulse shape analysis results are then read by the host system [**16**]. A more detailed explanation of the XIA DGF-4C cards can be found in Sulham [**16**]. The CAMAC crate is controlled by a Jorway 73A-2 Crate Controller, serial number 662 with a SCSI interface to a desktop computer. The desktop is operating Version 6 of Igor Pro with version 3.10 of the DGF Viewer extension installed. The software provides the ability to send set-up commands through the crate controller to the XIA cards and to record the digital data. The CAMAC setup used for this project is illustrated in Figure 28.

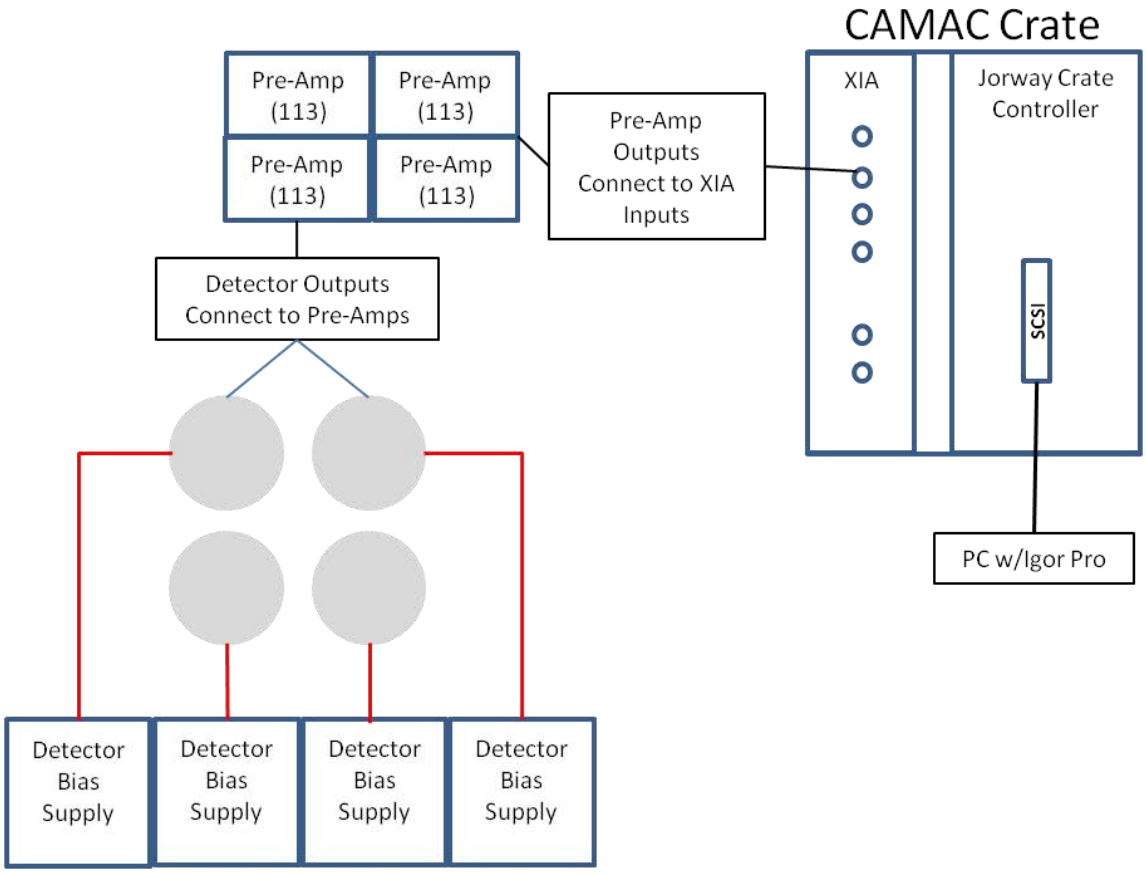

Figure 28. Data acquisition setup using CAMAC standard instrumentation.

The XIA data acquisition system presented here is used in experiments outlined in section IV. 7, with results presented in sections V. 3 and V. 4.

III. 3. Simulation

Simulation of the system geometry in various environments was completed using Geant4 and processed by Root. Geant4 is a toolkit of the simulation of the passage of particles through matter using Monte Carlo methods [**17**]. It was developed by CERN and uses the C++ coding language. The toolkit provides the ability to see the simulation through various viewers. An example of a 6 element detector geometry used for this project is shown with 10 simulated events using the HepRApp Browser in Figure 29. The HepRApp Browser was used in this project to verify detector geometries and source locations and distributions.

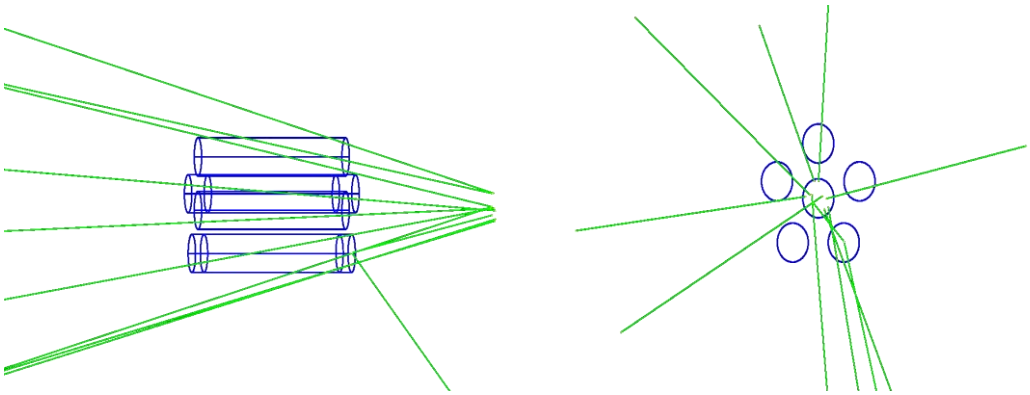

Figure 29. Simulation of 10 simulated events from a planar disk source emitting in a solid angle that covers the front of one of the geometries used for purposes of simulation. The data is shown with the HepRApp Browser visualization tool.

ROOT is a data analysis framework also developed by CERN that is commonly used to process the complicated output from Geant4. ROOT provides the ability to handle large datasets very efficiently and present the results of simulations professionally.

Included are the abilities to histogram in multiple dimensions, fit curves, evaluate functions, minimization, graphics and visualization classes to allow the easy setup of an analysis system that can query and process the data interactively or in batch mode [**18**]. As an example, Figure 30 shows a histogram generated from simulation data generated by Geant4 and processed with ROOT. Similarly, Figure 31 shows ROOT's ability to map where particles interact in specified volumes; in the case of this project, in the detectors.

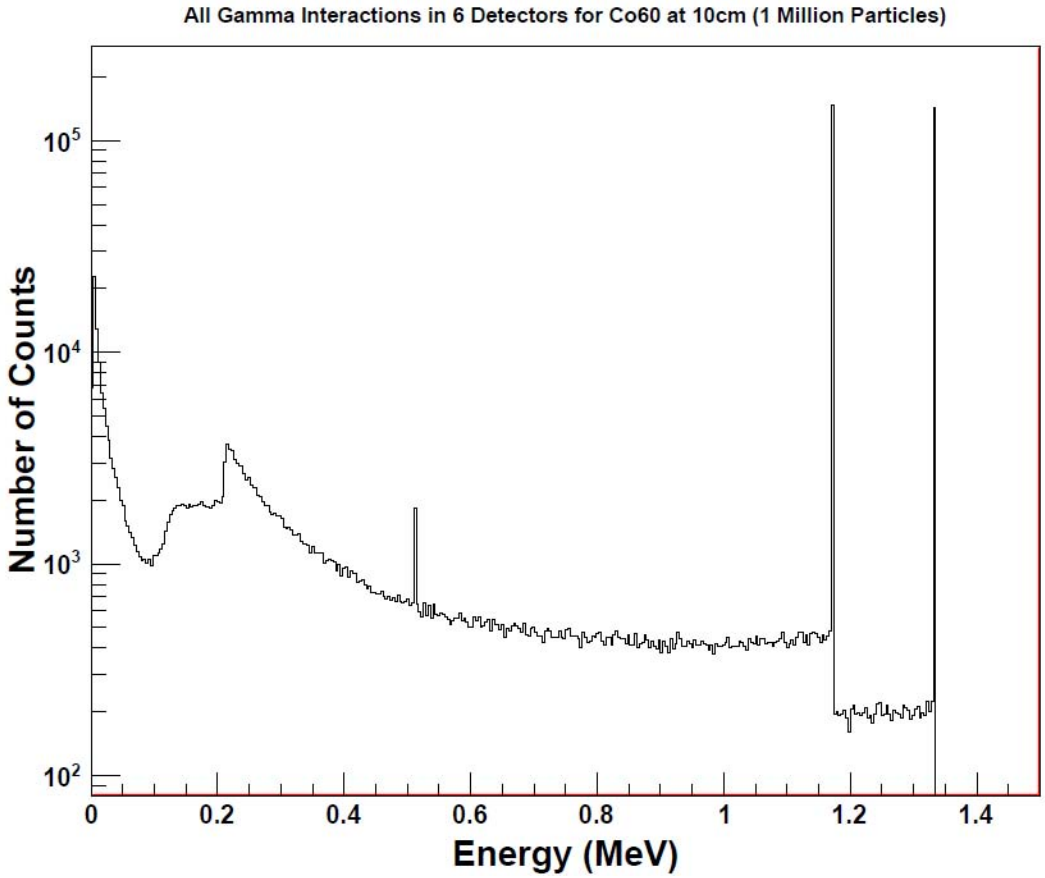

Figure 30. Spectrum of gamma interactions from Co-60 in the system geometry provided by ROOT.

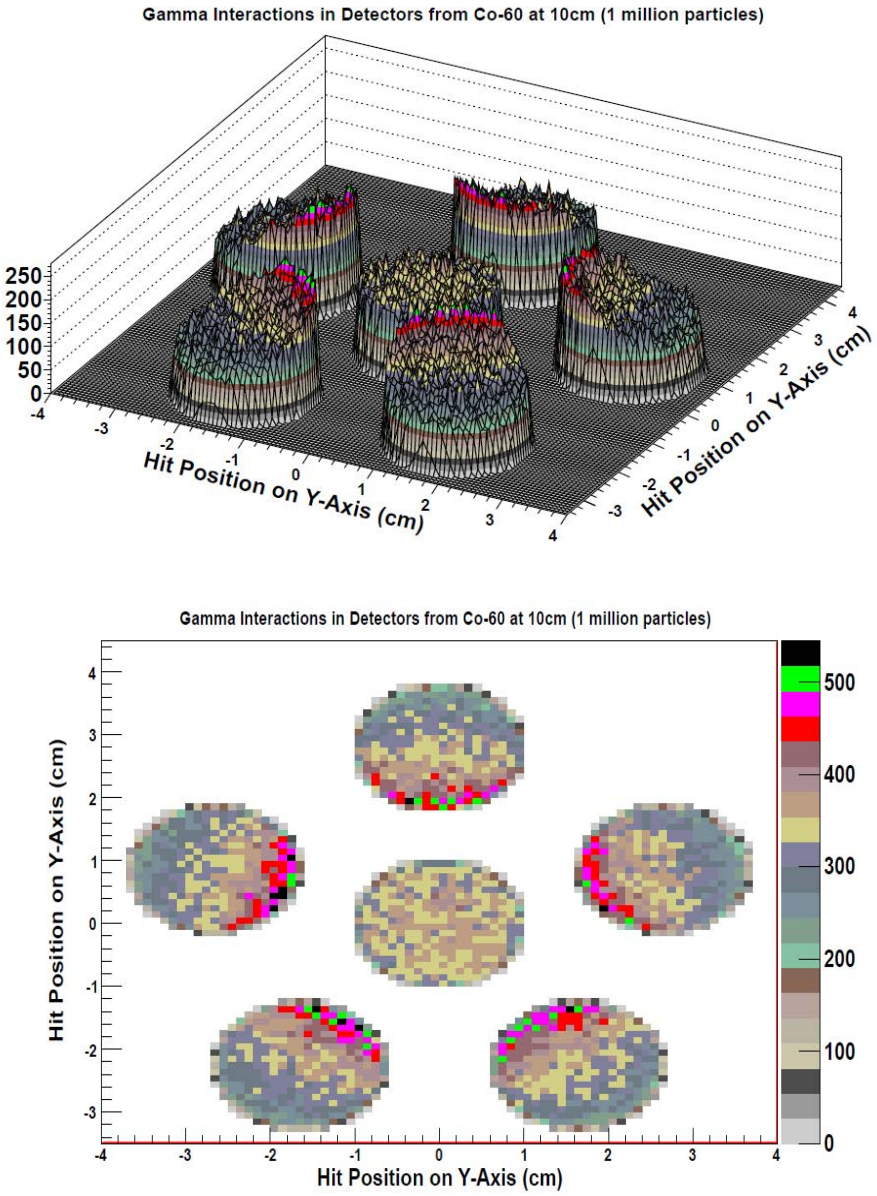

Figure 31. Interactions recorded in the detectors from simulated events generated from Geant4 as presented by ROOT.

The simulation software presented here is used in simulations outlined in section IV. 6, with results presented in section V. 5.

IV. Procedure

IV. 1. Chapter Overview

To construct a directional pair-production spectrometer, it was first necessary to set up all of the equipment and adjust the hardware and software settings, so that spectra could be collected for each of the detectors. The detectors then had to be characterized and the energy spectra had to be calibrated, ensuring that each of the detectors had the same energy calibration. A code was written to process the raw data and produce a crude image. A simulation was also created and analyzed to benchmark the constructed detector.

IV. 2. XIA Data Acquisition System Equipment Setup

The first step in setting up the equipment was to install Igor Pro 6 by Wavemetrics on a desktop PC with an Adaptec SCSI card installed. Then the latest version of XIA's DGF viewer software was installed. After the software installation was completed the hardware had to be correctly configured. The Jorway model 73A crate controller was placed in the two right-most slots of the CAMAC standard crate.

The DGF-4C modules were configured according to the user manual supplied with the software. This project utilized one such module and it was placed in slot 18 of the CAMAC crate. Table 23 gives all of the jumper settings for the DGF-4C module used in this project.

Table 23. DGF-4C Module Jumper Settings

Jumper Number	Module 1 - 1344
Jumper 1	Installed
Jumper 2	Installed
Jumper 3	Installed
Jumper 4	Installed
Jumper 5	Board Clock

The system was then booted by first powering on the crate and then booting the host computer. During the start-up process the operator inputs the SCSI bus ID number and the crate number. The inputs for this project had the bus ID number as 0 and the crate number as 3.

To get a similar energy calibration for each of the four channels, the gain setting in DGF viewer was adjusted so that the centroid of a full energy peak (FEP) from a calibration source was at the same location for each channel. This adjustment was done by taking spectra of Cs-137 for 10 minutes. The channel picked to force the FEP from the Cs-137 source was 20,000. The individual gain, offset and Tau settings are shown in Table 24. With these gain settings spectra were then taken of other common calibration sources such as Co-60, Cs-137 and Co-57 to create an accurate energy calibration.

Table 24. XIA Software Settings for Individual Channels

Channel Number	Detector S/N	High Voltage (V)	Gain (V/V)	Offset (V)	Tau (μs)
0	P903	780	7.0063	0.0103	180.2673
1	P572	780	3.1033	0.0602	166.4002
2	P570	680	3.9071	0.0442	151.6836
3	P902	820	7.6668	0.0175	148.4852

DGF viewer has the ability to accept user-defined settings to achieve the optimal resolution. One such setting is the energy filter rise time setting. A longer energy filter rise time optimizes energy resolution but lowers throughput [**19**]. Ten measurements were conducted, each for 900 seconds, while varying the energy filter rise time and then measuring resolution. Cs-137 was used for this resolution optimization calibration. The point was found where the energy resolution no longer improved while the throughput continued to slow. There are many settings within the software that are common across the channels. These are shown in Table 25. More details on what each setting does can be found in [**19**].

Table 25. XIA Software Settings Common for All Channels

Trigger Filter Rise Time (μs)	0.05
Trigger Filter Flat Top (μs)	0.05
Energy Filter Rise Time (μs)	0.8
Energy Filter Flat Top (μs)	1.2
Module CSRA	0x2400
Channel CSRA	0x009D
Threshold	30
Binning Factor	1
Coincidence Pattern	0xFFFF for Calibration otherwise 0xFEE8
Cutoff	0
Pulse Shape Analysis	N/A

The energy resolution of the detectors at 0.511 MeV is of utmost concern. More accurate resolution measurements will allow for more accurate energy filters to be applied in the XIA post-processing code to extract the coincident events of interest. The resolution of each $LaBr_3$(Ce) element was measured by placing a Sr-85 source, which has

a FEP at 0.514 MeV, 10 cm from the detectors and counting for 43,200 seconds. The measured resolution for each detector at 0.514 MeV using the Igor Pro 6 software is given in Table 26.

Table 26. Detector Resolution at 514 keV

Detector S/N	Resolution (%)
P903	4.34
P572	4.43
P570	4.59
P902	4.00

One feature of the XIA electronics that was utilized was the ability to specify detector hit patterns. When using only one module, the operator can specify that there must be at least two valid events within the user-specified coincidence window. A valid event is one that is above the threshold setting. The detector hit patterns used in this project are those that only permit two or more coincident events. This greatly reduces data sets and increases the efficiency of the module reducing the number of times the hardware must read out its buffer during a given run. As an example, a three minute collection was taken of a Co-60 source with the coincidence window set to 100ns and the hit pattern allowing any combination of events to be counted. The output file for that collection was 14.2 MB and the number of detected pair-production events is 80. The plotted results are shown in Figure 32. Not specifying a hit pattern that permits only coincident events allows the software to record any event, even those that occur without another coincident event. Shortly thereafter a collection was taken of the same Co-60 source with the coincidence window set to 100ns but with the hit pattern allowing only

those events that had two or more hits reported. The output file for that collection was 727 KB and the number of detected pair-production events was 176. The plotted results are shown in Figure 33.

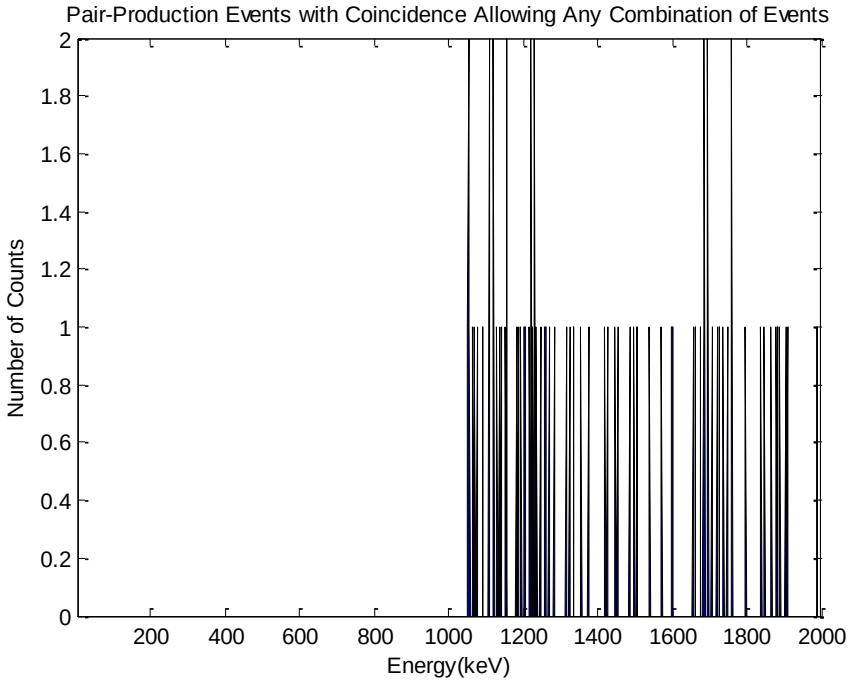

Figure 32. Pair production events from a Co-60 source from a 3 minute collection with a coincidence window of 100ns and the hit pattern allowing any combination of hits. There are 80 total pair-production events.

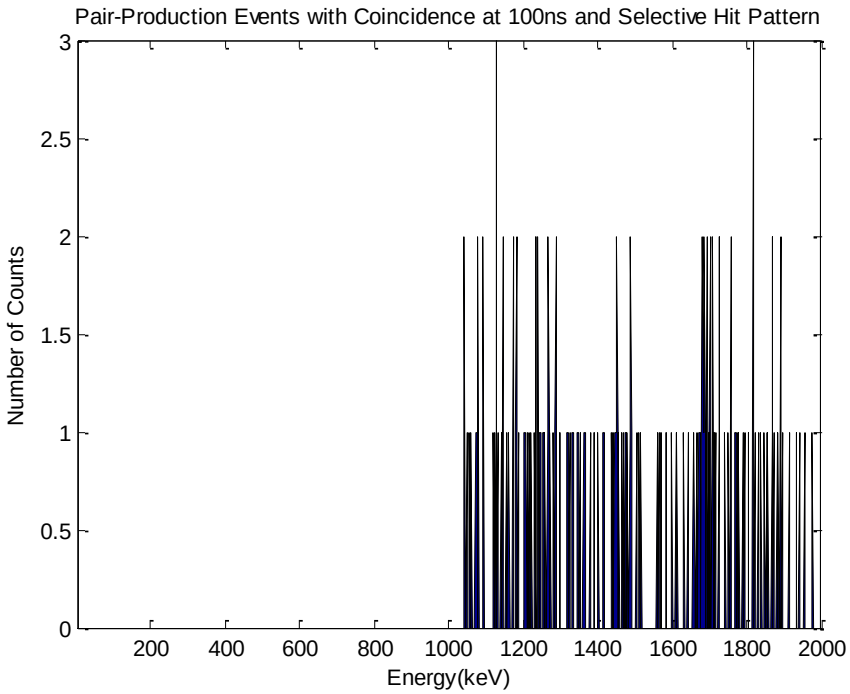

Figure 33. Pair production events from a Co-60 source from a 3 minute collection with a coincidence window of 100ns and the hit pattern allowing only those events that had two or more hits reported. There are 176 total pair-production events.

The results of the measurements taken with the XIA data acquisition system will be presented in sections V. 3, V. 4, and V. 6.

IV. 3. XIA Post-Processing Code in MATLAB

The post-processing code from the XIA output has three main parts. The first part extracts the desired data from the output, eliminating column headers and file identifiers, and organizes it into a more condensed and usable format. The second part of the code scales the data by applying an energy calibration in the form of a second order polynomial. It also performs a series of logic that verifies that the reported events are indeed a coincident event. The code then reports the energies of the coincident events and the detectors that are involved in the event. Next, the coincident events are filtered into

whether they are pair-production or Compton events by using energy filters that discriminate whether the event has a detected gamma whose energy is within the resolution of the detectors about 0.511 MeV. The window about 0.511 MeV is between 0.485 MeV and 0.536 MeV. If the event has a detected gamma that falls in that range, then that event is considered to be a pair-production event, otherwise it is a Compton event. For the Compton events, the code sums the coincident events to find the parent event energy. For the pair-production events, the code identifies which event is outside of the range that encompasses the resolution of the detectors about 0.511 MeV. 1.022 MeV is then added to that energy to give the energy of the parent event. If both events are between 0.485 MeV and 0.536 MeV, the event is given an energy of 1.022 MeV.

The third part of this code calculates the possible backprojected locations of the parent source onto a source map and overlays the rims of those cones to come up with the most likely location of the source. As there is no way to differentiate the ordering of the pulses, the order is chosen based on the most probable Compton scattering angle according to the Klein-Nishina relation. The plots used to visualize these projected images are Mercator maps, like the ones used to map the world on a 2-D map. The ϕ angle is given on the x-axis to facilitate an easier understanding of the location. The indicated θ does not have any variation and indicates the horizontal plane. This is due to the inability to differentiate a θ direction by the Compton camera. The indicated ϕ angle is best described by Figure 34. The detectors in Figure 34 are situated as if they are pointed out of the paper.

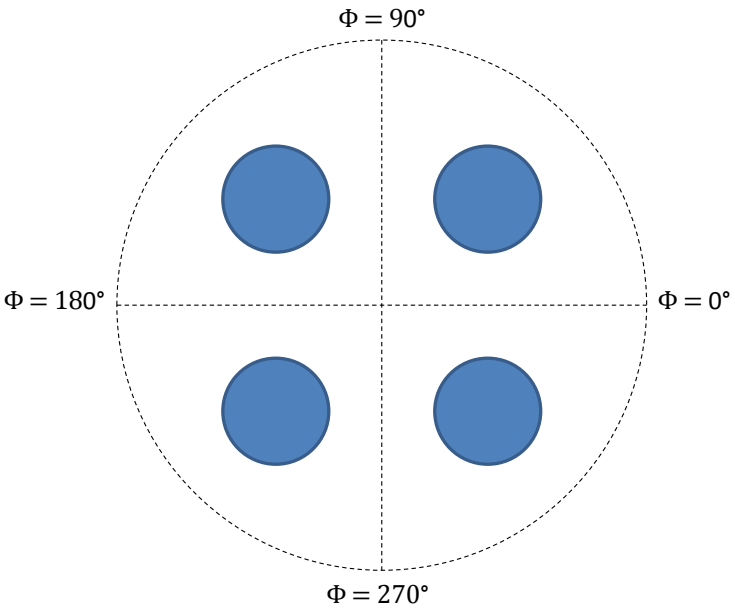

Figure 34. Shown are the ϕ angles as reported by the plots in Chapter V. The detectors are indicated by the shaded circles and are assumed to be pointing out of the paper.

The results of the XIA post-processing code will be presented in sections V. 3, V. 4, and V. 6.

IV. 4. Oscilloscope Data Acquisition System

Using the Tektronix DPO7104 oscilloscope as a data acquisition system allows the operator to harness the high sampling rate capability that the scope offers. While the oscilloscope by itself can store the waveforms on its internal hard drive, certain steps can be taken to turn the oscilloscope into an efficient data acquisition system. These steps are detailed in Appendix A.

The oscilloscope is operated by running a MATLAB script on the host PC. This is also found in Appendix A. Before running the MATLAB script, the operator needs to choose how many waveforms per acquisition are desired. The MATLAB script has the ability to change the enabled channels, the sampling mode, the scale, etc. Two of the

useful commands within the script are the Data:Start and Data:Stop commands. These tell the scope which of the displayed data points to send to the host PC. In the current configuration, there are 1,000 points per waveform however, the waveforms of interest only take up 126 of those points, therefore the Data:Start command is set to 75 and the Data:Stop command is set to 200. If the operator wanted the entire waveform the settings would be 1 and 1000, respectively. The triggering mechanism used in the script is the Pattern trigger.

Pattern trigger is a logic trigger that triggers when logic inputs cause the selected function to become true or false. This trigger can be used with all four of the channels of the oscilloscope. The settings for this project had all of the channels set to HIGH, the logic set to false, and the trigger threshold set to 10mv. A screenshot of the settings for the Pattern trigger is provided in Figure 35.

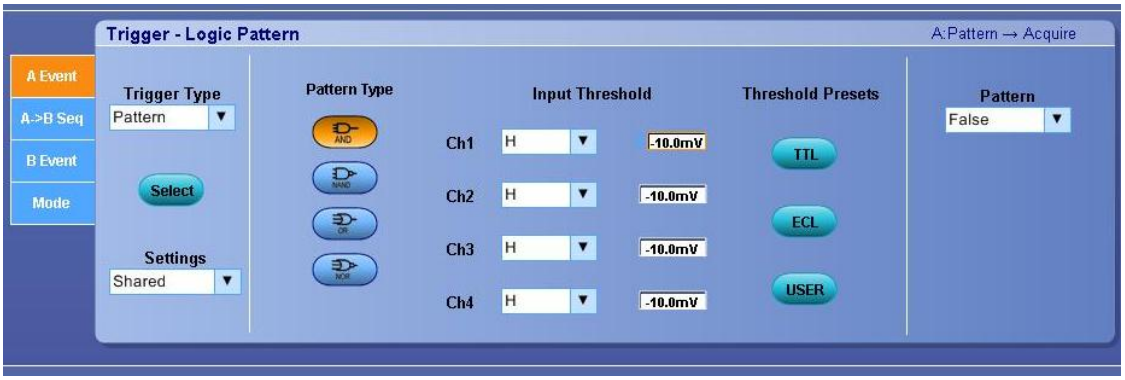

Figure 35. Pattern Trigger settings on oscilloscope. The same settings were used in the MATLAB script to control the oscilloscope.

This allows the initializing trigger to occur on any one of the four channels. Upon receipt of the initializing trigger, the oscilloscope then triggers on the three remaining channels. Unfortunately, when using four channels the Pattern trigger mode cannot be

configured to only trigger when two of the four channels have a triggered event, therefore, many non-coincident events are reported.

The data rate of this data acquisition system is limited. The settings used for this project returned 100 waveforms per channel per second with 126 points per waveform and 1 nanosecond per point. This performance can only be achieved by turning off the "Waveform Display On/Off" option in the Display toolbar on the oscilloscope. These waveforms were then fed into a post-processing code written in MATLAB to analyze and discriminate the data.

The results of the measurements taken with the oscilloscope will be presented in section V. 2.

IV. 5. Oscilloscope Post-Processing Code in MATLAB

The post-processing code analyzes the waveforms that are returned from the oscilloscope. After manipulating the data matrix into a usable format, the data is smoothed using a Savitsky Golay filter with a third order polynomial and a window of seven points [**20**]. The maximum pulse height was then found and an energy calibration was applied to that value to obtain the energy of the deposited gamma on the detector. Timing parameters for the incoming pulses are found by finding the time that corresponds to the midpoint of the leading edge of the pulse. If that point falls between two times from the timing array given by the oscilloscope then the code finds the nearest time to that point. The code then determines whether the pulses are within a user-defined coincidence window.

The pulses that are within the coincidence window are then run through a series of filters to determine if they are pair-production events or Compton events. These filters

call an event pair-production if one of the coincident gamma rays is within was within the user-defined resolution of 0.511 MeV, otherwise they are considered Compton events. The user-defined resolution for this project is 5%. In the case of the pair-production events, the coincident pulse that resulted in an energy other than 0.511 MeV is added to 1.022 MeV which is the energy of the two gammas produced when a positron and electron pair annihilate. In the case of the Compton events, the pulses are simply added up to find the energy of the originating gamma ray. The pair-production events and Compton events are tabulated into separate arrays to allow for further analysis. The results from the post-processing algorithm are placed in histograms to perform spectroscopy on the measured sources.

The results of the measurements taken with the oscilloscope will be presented in section V. 2.

IV. 6. Simulation in Geant4 and Post-Processing in ROOT

IV. 6. A. Geant4 Simulation

The Geant4 simulation was conducted to benchmark the performance of the detector system and to extrapolate the performance of the system to various configurations. The simulation utilizes the class structure that is common to the C++ programming language. The classes that are used in this simulation are Detector Construction, Hit, Event Action, Primary Generator, TrackerSD and Physics List. The Detector Construction class is where the material properties and geometry of the detector system and the world in which the experiment takes place are described. The Hit class serves as a way to control what parameters are reported when a hit occurs in the simulation. The Event Action class controls which parameters are written to the output

file. This class is where changes should be made to modify the ways that various parameters are written to the output file, such as changing a string to a predetermined number as was done when identifying the particular physics process that occurred for an event. The physics process codes used to simplify the output file are shown in Table 27.

Table 27. Physics Process Codes used in Geant4 Simulation

Physics Process	Geant4 Abbreviation	Code
Primary Generator Particles	PrimaryGen	0
Photoelectric Effect	phot	1
Compton Effect	compt	2
Gamma Conversion	conv	3
Annihilation	annihil	4
Bremsstrahlung	eBrem	5
Ionization	eIoni	6

The Primary Generator class is where the manner in which particles are shot at the detector is identified. This class can also be used to direct the program to a macro entitled vis.mac, where it will find the instructions for shooting particles. This approach was taken for this simulation. The vis.mac file is a macro file where the visual parameters of the simulation are controlled. Here the programmer can instruct Geant4 to write the visual output to a file to open in another visual browser, such as the HepRApp program. This output file is also where the General Particle Source (gps) can be described. The gps class allows the specifications of the spectral, spatial and angular distribution of the primary source particles to be defined very easily [**21**]. The user can program the gps to shoot one or multiple particles with specified intensities (branching ratios). In this simulation, the gps class was chosen to model the Co-60 source used as closely as possible. The gps class can also be used to describe the geometric properties

of the source. In this simulation, a circular planar source with a diameter of 3 mm was used to mimic the source used in the lab experiments. Using the gps class in the vis.mac file as opposed to using the typical approach of using the Primary Generator class to shoot the particles gives the advantage of not having to recompile every time after making an adjustment in the vis.mac file.

The Physics List class is where the programmer can include the various kinds of physics that the simulation should allow for. This simulation included the physics for electron multiple scattering, ionization, and Bremsstrahlung . For positrons the physics included was for multiple scattering, ionization, Bremsstrahlung , and annihilation. Lastly, for gammas, the physics included was the photoelectric effect, Compton scattering, and gamma conversion. The Physics List class is also where cut values are set for the simulation. A higher cut value will speed up the simulation and cause Geant4 to not track the particle once its energy or path length drops below the specified cut value. When a particle drops below that value its remaining energy is not lost. It is counted as being locally deposited. In high density bulk scintillator materials such as the LaBr3 used in this project, low energy gammas and electrons will not travel very far so the cuts can be set at higher values than if the detector was a thin, wafer-like semiconductor. The cut length for this project was set to the default value of 1 mm and the energy cut values were set to 20 keV.

The last class used for this project was the TrackerSD (Sensitive Detector) class. The Tracker SD class is where the information that is needed to write the output file for each particle track is specified. Whatever information is specified here, if added properly

in the Hit class, will get written to the output file. The output file is written to a .out file and placed in the directory that the simulation is located.

The output files can be processed in a number of ways using user-written code in FORTRAN, MATLAB or ROOT. ROOT was used for this project. The output files are written in such a way that there is a row of "-1's" separating each event. There is one "-1" for each entry in the output file. ROOT uses this line of "-1's" to identify when an event track ends. Each event will have various numbers of hits, depending on the track. Each of these hits will have the information reported for it that was defined in the Event Action class. Each of these classes has a file in the /include and /src subdirectories.

The results of the Geant4 simulations will be presented in section V. 5.

IV. 6. B. Simulation Post-Processing in Root

Post-processing the output file from the Geant4 simulation is a two step process. First the output file must be translated into a .root file. This is done using the MakeRootTree_hit.C code in Appendix C.17. ROOT files use a tree, branch, leaf structure. For the data produced in the simulation for this project, the hit file itself is defined as a Tree. Each branch is comprised of a particular attribute from the hit file, such as the particle identification number or the physics process. Each leaf is then filled with the actual data from each event. After the .root file is created, ROOT can then be used to process the data. This is done using the ProcessAllPairCompton.C file in Appendix C.18.

This code contains an algorithm to extract the pair production events in a manner that resembles the way that detectors in a laboratory setting see a pair production event. This is done by summing up the deposited energies of the electron tracks for each gamma

produced by the primary generator. If in summing up the electron tracks in one detector, a 0.511 MeV gamma is found, then 1.022 MeV is added to whatever energy, if any, was deposited in another detector. Care has to be taken with how Geant4 accounts for an electron's lost energy when that loss results in a Bremsstrahlung event. In the event that a Bremsstrahlung event occurs, the beginning energy of that Bremsstrahlung photon must be subtracted from the total deposited energy of the primary gamma event. If this is not done, a spectrum of the deposited pair production events will contain non-physical events that are too high in energy to possibly occur. If a qualifying pair-production event is found, the resulting energy is added to a histogram which is then printed to the screen. These histograms can then be saved as .root files allowing future modification and analysis of the data. This can be done by starting up ROOT and opening up the Tree Browser by typing "new TBrowser()" in the command line. The TBrowser allows the programmer to navigate to the desired .root plot and make the desired modifications and analysis. Command line commands can be entered to display more plots and histograms that do not require a specific post processing code to extract the data.

The results of the simulation post processing code will be presented in section V. 5.

IV. 7. Directional Efficiency Measurements

Directional efficiency measurements were completed to characterize from which direction the detector is most efficient at detecting photons. These were completed by first creating a template that traced out the base of the detector mount in 45° increments around a central point. This was done so that the source could remain stationary and the detectors could be repositioned between acquisitions. This system allowed the source to

be the same distance from the central point of the detector system no matter the orientation. In-plane spectra, where ϕ equals 0°, were taken in 45° increments from 0° to 180°. These measurements were repeated in 30 cm increments from 30 cm to 90 cm. A simplified layout of the experiment setup is shown in Figure 36.

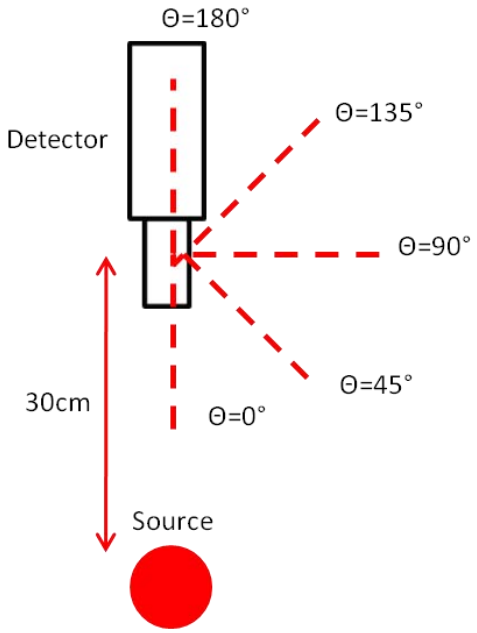

Figure 36. Simplified layout of in-plane measurements, where ϕ = 0°.

Out of plane measurements were taken, where ϕ is equal to 45° in spherical coordinates, using the same procedure as the in-plane measurements. Care was taken to maintain the distance between the source and center of the detector crystals at 30 cm, 60 cm or 90 cm. A simplified layout of the out-of-plane experiment is shown in Figure 37. The experimental setup used to take the out-of-plane measurements is shown in Figure 37.

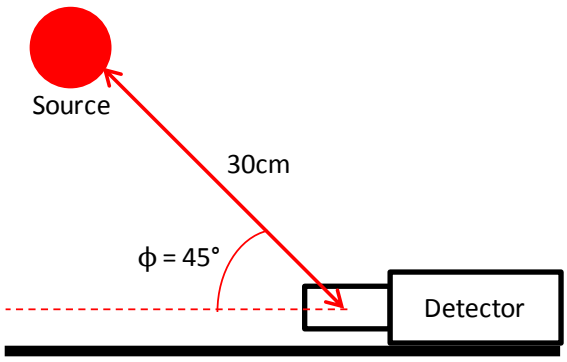

Figure 37. Simplified layout of out-of-plane measurements, where ϕ = 45°.

Background measurements were also taken to quantify the contribution of the inherent radiation of the La-138 and the laboratory and cosmic background to the coincident pair-production and Compton spectra.

The results of the directional efficiency measurements will be presented in section V. 4.

IV. 8. Directional Efficiency Simulation

Simulations were completed in Geant4 to benchmark the measurements taken in the lab. This was done by defining detectors of the same composition and size as those used in the laboratory and replicating the in-plane measurements taken in the laboratory. The simulations took place in a simulated world filled with air and did not take into account the intrinsic radiation of the La-138. After replicating these measurements, simulations were completed with larger detectors than are currently available and with a higher energy source, 2.6 MeV, than were possible in the laboratory.

The results of the directional efficiency simulation will be presented in section V. 5.

V. Results and Analysis

V. 1. Chapter Overview

This chapter presents the results of determining a difference in the pulse-shape between a Compton and a pair-production event. The contributions of knowing the order of the pulses is analyzed. In addition, the efficiency gained by adding additional detector pairs to the detector geometry is analyzed. The results of the directional efficiency measurements of the four element detector are presented and compared to simulations completed with Geant4. Other detector geometries are also simulated and used to approximate their performance with higher energy gamma rays. Lastly, the capability of the detector as a Compton camera is evaluated.

V. 2. Difference between Compton and Pair-Production Events

To determine whether a difference between the pulse shape of a Compton Event and the pulse shape of a Pair-Production event exists, the data acquisition system incorporating the Tektronix DPO7104 oscilloscope was used. A four hour spectrum was taken using two detectors at a distance of 10 cm of a Co-60 source. The source had an activity of 6.7 μCi. The oscilloscope post processing code discussed in section IV. 5 was used to separate the Compton events from the Pair-Production events. These events were then further sorted into arrays based on the energy deposited. If the energy deposited was within a 5% resolution of 511 keV, the event was placed into an array for its particular interaction process to be summed and plotted. Only 92 pair-production events meet these qualifications, therefore, the number of Compton events plotted was restricted to the same number. The Compton event order was determined by calculating the probability of scatter using the Klein-Nishina formula. The deposited energy that was calculated to

be the location of the Compton scatter was used for this analysis. The pulse used for the pair-production events was the energy deposited in the detector where the pair-production occurred. Examples of each type interaction are shown in Figure 38.

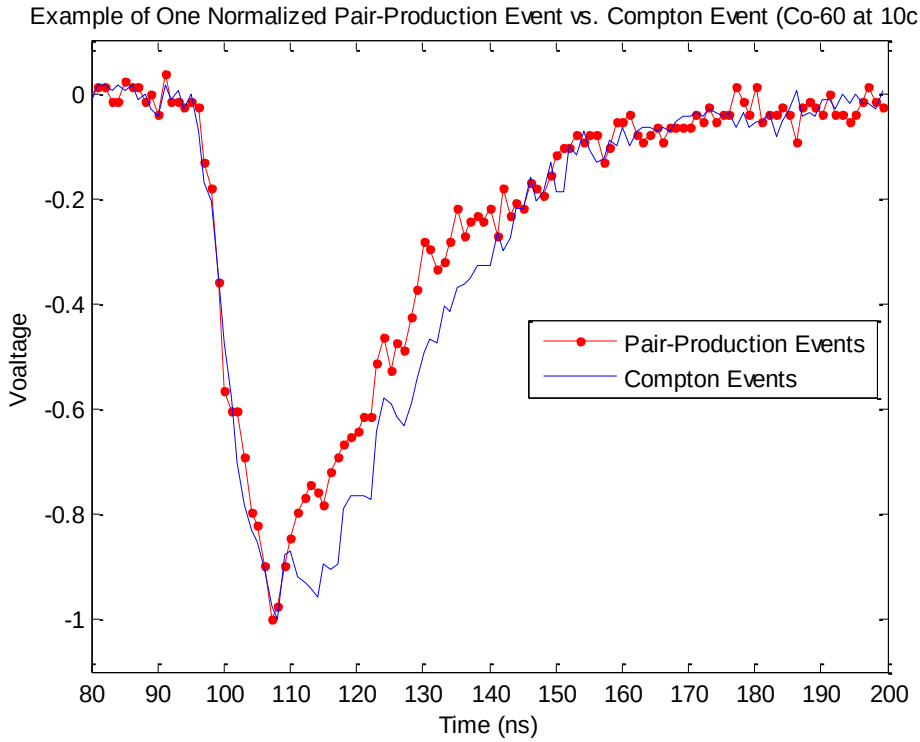

Figure 38. Normalized pair-production and Compton event from a two element detector Co-60 acquisition at 10 cm.

It is difficult to determine if there are any trends that perpetuate from event to event when looking at single events. One technique to allow subtle features of the pulses that are frequently present become apparent is to look at an aggregate pulse. To do this, the pulses for each type of event were summed and normalized to one. The result of this process is shown in Figure 39. Below the aggregate plot is a plot of the absolute difference between the two aggregate pulses. The absolute difference plot shows a significant difference in the rise of the pulses and a slight difference in the fall of the

pulses, however the trend indicates a change from a positive to negative difference. This is likely due to the timing jitter of the oscilloscope. Timing jitter is a source of timing inaccuracy in electronics that use leading edge triggering; the triggering method used in this application. Timing jitter is caused by random fluctuations of noise and will cause pulses to appear to arrive at different times with respect to the centroid of the pulse [4, 659].

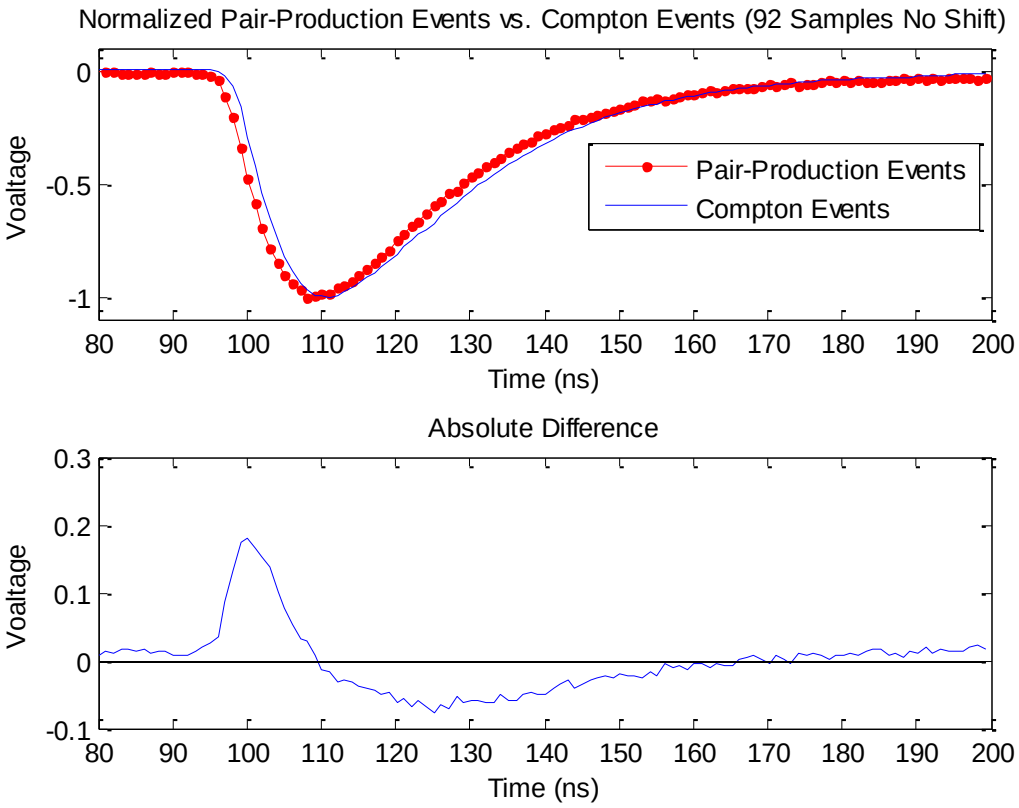

Figure 39. Normalized sum of 92 pair-production and Compton events from a two detector element Co-60 acquisition at 10 cm.

Shifting the aggregate pulses in time provides an example of what the data would look like when correcting for timing jitter. Figure 40 shows the same aggregate pulses shown in Figure 39, but with the Compton events curve shifted earlier in time by 1 ns.

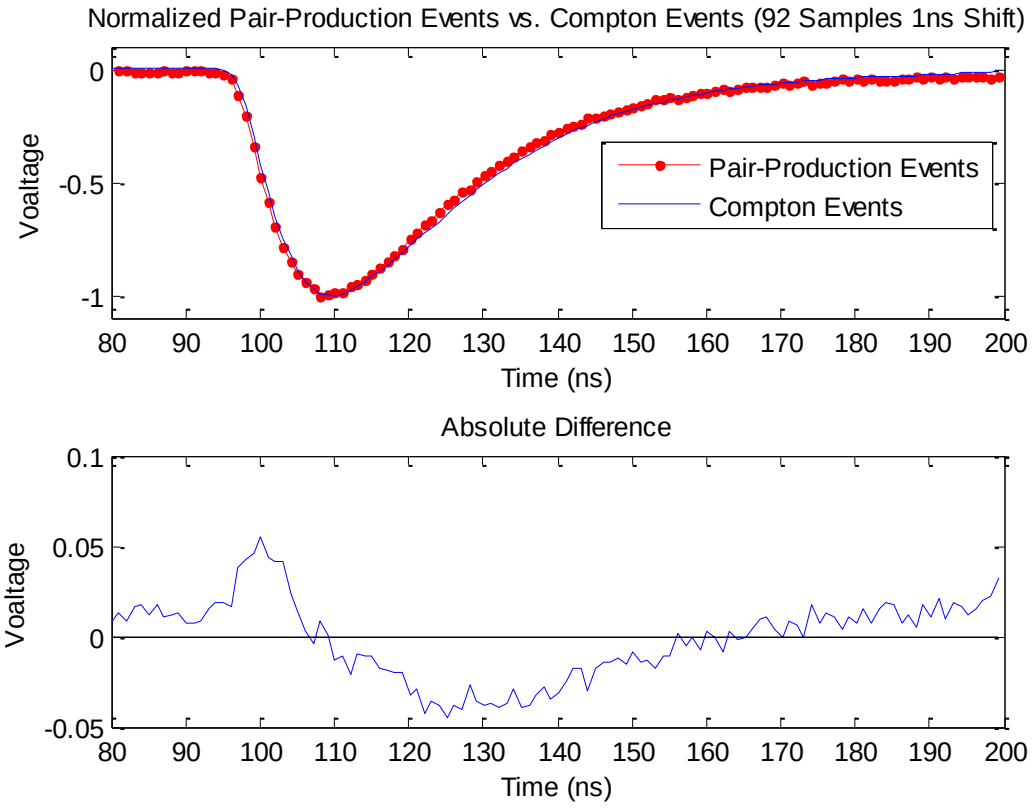

Figure 40. Normalized sum of 92 pair-production and Compton events from a two detector element Co-60 acquisition at 10 cm. The Compton events pulse has been shifted earlier in time by 1 ns.

The 1 ns shifted data indicates a decreased difference between the two aggregate pulses when comparing to the unshifted data. The absolute difference plot still shows a positive difference that transitions into a negative difference at the centroid of the peak. Shifting the Compton events data in time by 1 ns more to 2 ns results in the plots shown in Figure 41.

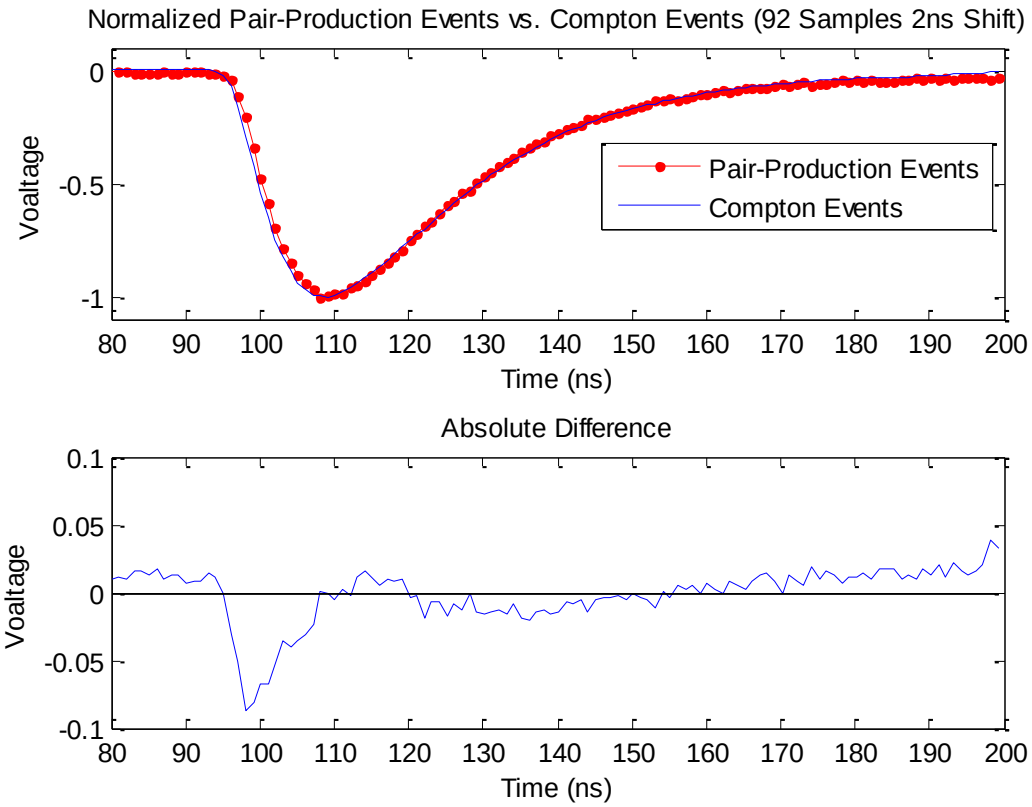

Figure 41. Normalized sum of 92 pair-production and Compton events from a two detector element Co-60 acquisition at 10 cm. The Compton events pulse has been shifted earlier in time by 2 ns.

The 2 ns shifted data indicates a change in the absolute difference plot. The difference in the leading edge of the pulse has changed to a negative difference. The falling edge of the pulse is very close to the zero line of the plot and shows a difference that is approximately 2%. Shifting the data by 1.5 ns from the original data would likely provide the least difference between the two pulses. Unfortunately, the settings used on the oscilloscopes make it difficult to shift the data by 1.5 ns. The data presented in the figures above indicate that there is not a difference between the two pulses, at least that can be discerned with the techniques used. More pulses that could contribute to the summed data should verify this conclusion.

V. 3. Two Detector Experiment

Much of what can be characterized about the four element detector can be using two detectors. Using two detectors can also allow for extensions and extrapolations on the design used in this project. The relative efficiencies of the detectors were computed by summing up the number of counts that fall within 3%, the approximate resolution of the detectors at 1173 keV to 1332 keV, of the two characteristic peaks of Co-60, 1173 keV and 1332 keV. The same was done for the peaks located at 1436 keV, 1684 keV and 1843 keV. The 1436 keV peak is of interest because of its contribution from the intrinsic La-138 radiation in the detectors. The 1684 keV and 1843 keV peaks occur when chance Compton events happen where a primary gamma Compton scatters into 511 keV or whatever the energy is of the primary gamma minus 511 keV. This is counted as a pair-production event.

A background spectrum was first taken for 12 hours to characterize the contribution of the intrinsic radiation of the La-138 in the detectors and the background itself to the spectra to be taken later. The pulse-height spectrum is shown in Figure 42. The beta continuum and peak at 1436 keV are due to the intrinsic radioactivity of the La-138 in the detector. The pair-production background spectrum is shown in Figure 43 and the Compton background spectrum is shown in Figure 44. The number of counts recorded in the peaks of interest are shown in Table 28. All of the spectra, unless otherwise noted, are binned into 1024 channels from 0 to 2000 keV giving 1.953 keV per channel.

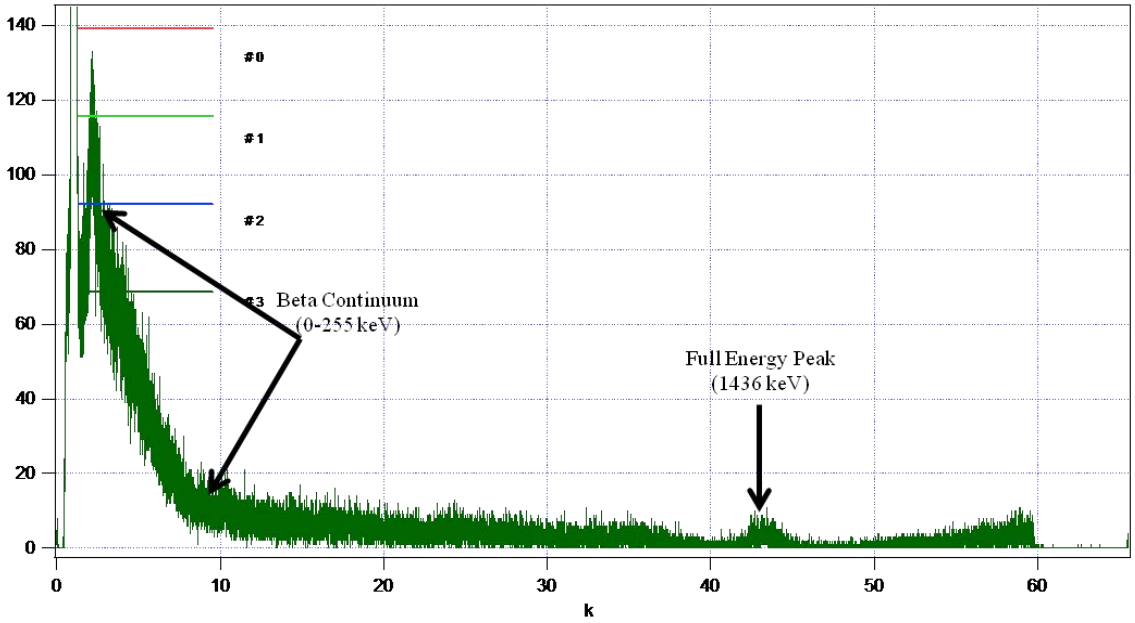

Figure 42. Pulse-height spectrum of 12 hour background acquisition for one detector. Shown are two features of the spectrum that are due to the intrinsic radioactivity from La-138. The horizontal axis is channel number in thousands and the vertical axis is the number of counts.

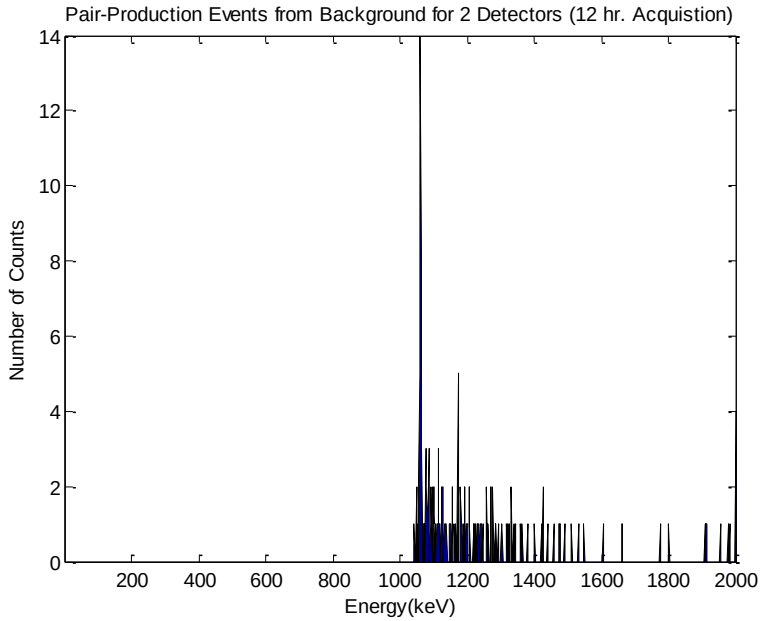

Figure 43. Pair-production event background spectrum from a 12 hour acquisition using a 2 element detector and allowing the 511 keV gamma to occur in either detector.

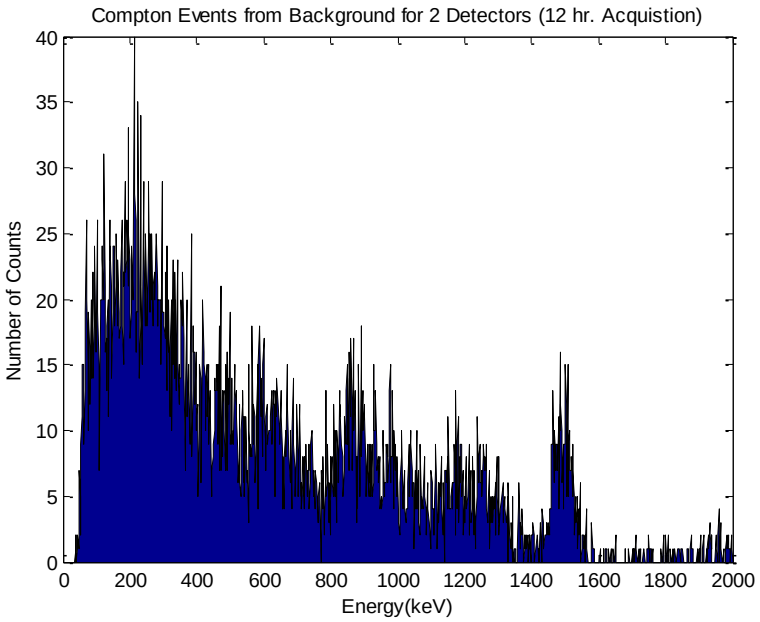

Figure 44. Compton event background spectrum from a 12 hour acquisition using a 2 element detector.

Table 28. Number of Counts in Peaks of Interest from 12 hr. Background Acquisition and 2 Detectors

	511 keV in Either Detector		511 keV in Detector 1		511 keV in Detector 2	
Peak (keV)	Compton Events	Pair Production Events	Compton Events	Pair Production Events	Compton Events	Pair Production Events
1173	205 ± 14.3	52 ± 7.2	206 ± 14.4	25 ± 5	206 ± 14.4	27 ± 5.2
1332	130 ± 11.4	27 ± 5.2	131 ± 11.4	12 ± 3.5	131 ± 11.4	15 ± 3.9
1436	224 ± 15.0	19 ± 4.4	228 ± 15.1	9 ± 3	230 ± 15.2	10 ± 3.2
1684	N/A	4 ± 2	N/A	1 ± 1	N/A	3 ± 1.7
1843	N/A	2 ± 1.4	N/A	1 ± 1	N/A	1 ± 1

To characterize the performance of a two element detector, a Co-60 source of activity 6.7 μCi was measured at 15 cm from the center of the detector crystals. The uncollimated source was used to establish the performance of a 2 element detector without knowing the order of the pulses. Data was taken for 12 hours. The pulse-height spectrum from one detector is shown in Figure 45. Important features of the pulse-height

spectrum are pointed out in the figure. The double escape peak for the 1332 keV gamma ray is of great interest as that is the location of the events that were caused by pair-production of the 1332 keV gamma. The double escape peak for the 1173 keV gamma ray is hidden underneath the beta continuum from the La-138, the low energy backscatter gamma rays, and the characteristic x-ray peak of the lead shielding around the detector. The data was processed in a manner to allow detection of the 511 keV gamma in either detector one or detector two. The pair-production spectrum from the uncollimated 2 element detector acquisition is shown in Figure 46 and the Compton spectrum in Figure 47.

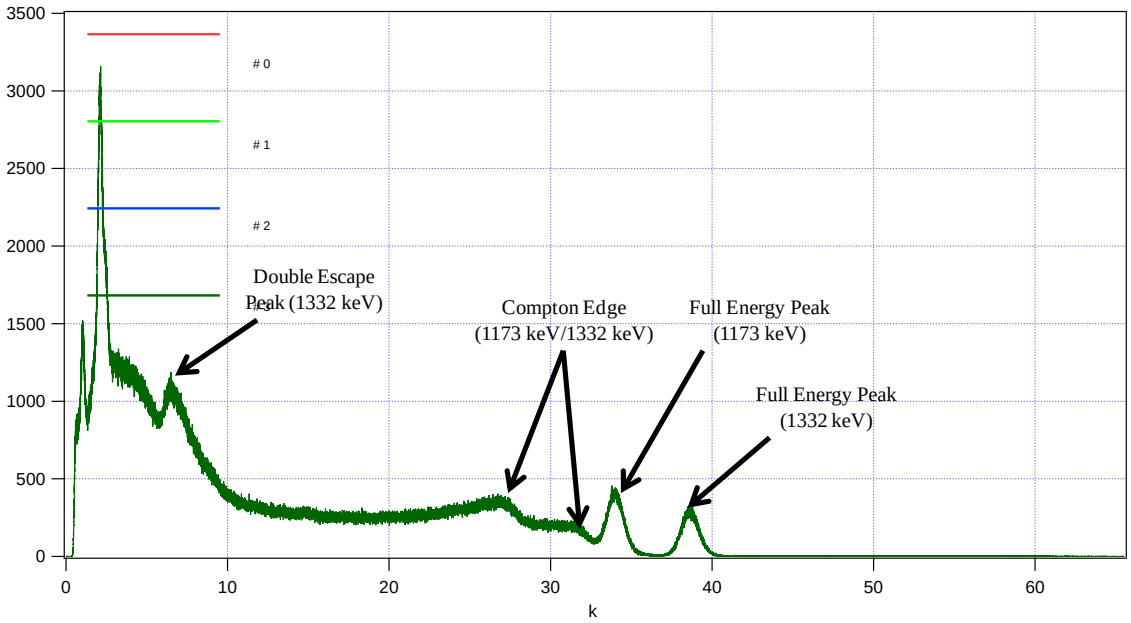

Figure 45. Pulse-height spectrum of 12 hour Co-60 acquisition for one detector at 15cm. The horizontal axis is channel number in thousands and the vertical axis is the number of counts.

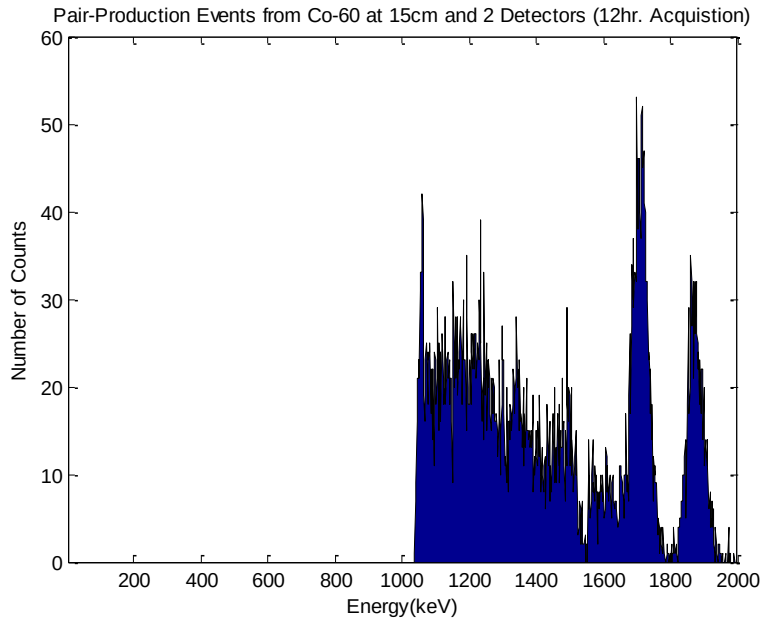

Figure 46. Pair-production events from a Co-60 acquisition for 12 hours at 15cm from two detectors.

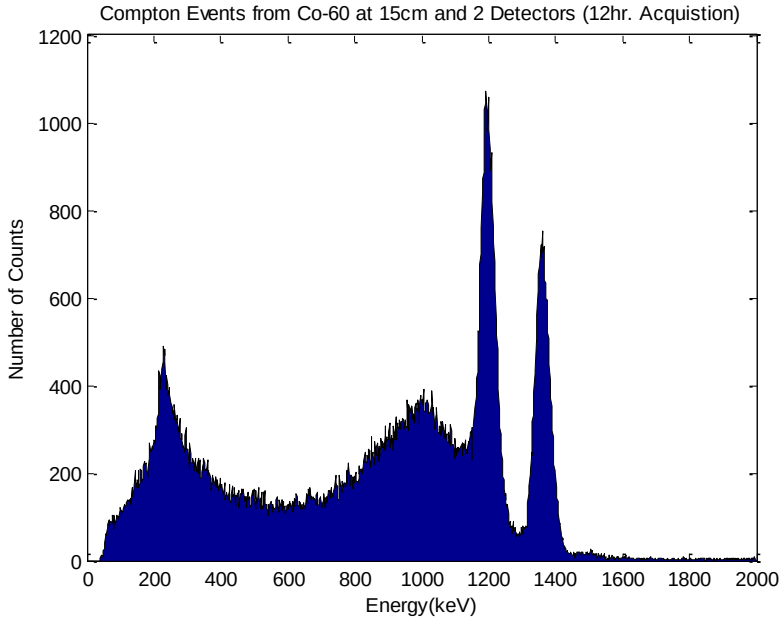

Figure 47. Compton events from a Co-60 acquisition for 12 hours at 15cm from two detectors.

The number of counts recorded in the peaks of interest, with and without background, are shown in Table 29.

Table 29. Number of Counts in Peaks of Interest from 12 hr. Acquisition of Co-60 at 15cm and 2 Detectors

	Including Background		Less Background	
Peak (keV)	Compton Events	Pair Production Events	Compton Events	Pair Production Events
1173	20729 ± 144.0	767 ± 27.7	20524 ± 143.3	715 ± 26.7
1332	15222 ± 123.4	724 ± 26.9	15092 ± 122.8	697 ± 26.4
1436	2901 ± 53.9	664 ± 25.8	2677 ± 51.7	645 ± 25.4
1684	N/A	1298 ± 36.0	N/A	1294 ± 36.0
1843	N/A	744 ± 27.3	N/A	742 ± 27.2

The true pair-production events seen in the barely distinguishable 1173 keV and 1332 keV peaks are dwarfed by the chance Compton events that contribute to the 1684 keV and 1843 peaks. The two element detector, over the period of 12 hours, should have had 6.8 E+7 recorded gammas emitted by the Co-60 source in the solid angle subtended by the detector. Of those, there should have been a total of 4.5E+4 pair-production events contributed by the 1173 keV gamma and 2.4E+5 pair-production events contributed by the 1332 keV gamma. Dividing the number of counts observed within 3% of the FEP by the number predicted by theory gives efficiencies of 3.4% for the 1173 keV peak and 0.61% for the 1332 keV peak. The efficiency for the 1684 keV chance Compton peak is 0.10% and for the 1843 keV peak is 0. 065%. Theory indicates that many more pair-production and chance Compton events should have been recorded than are indicated in the plots. This is likely due to the inefficiencies of capturing the 511 keV annihilation photons and chance Compton secondaries.

In the four element detector, it is impossible to know which detector sees the initial gamma first with the electronics used for this project. An acquisition using the same source as before but collimated to one detector with a 5 cm thick lead brick, as shown in Figure 48, was run for 12 hours. The collimated two element detector setup was used to characterize the contribution that knowing the ordering of the pulses would contribute to the overall efficiency of the detection system.

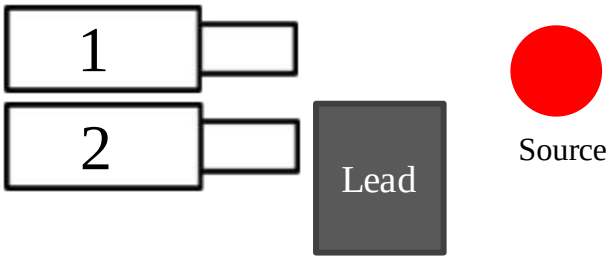

Figure 48. Shown is the orientation of the collimated two detector experiment. The unshielded detector is number 1 and the shielded detector is number 2.

The unshielded detector will be known from here on as detector one and the shielded detector will be known as detector two. The data was processed by filtering out the pair-production events from the Compton events. This was done by allowing a pair-production event to be one that not only meets the energy and coincidence timing requirements described previously but it also must deposit its 511 keV annihilation gamma in detector two. The only chance Compton events that get counted when processing the data in this way are those that meet the same criteria. The spectra that result from this are shown in Figure 49 for the pair-production events and Figure 50 for the Compton events. Binning the pair-production spectrum into fewer bins allows the features of the 1332 keV peak to become more apparent while the 1173 keV peak remains hidden. Figure 51 shows the pair-production spectrum plotted in 256 bins.

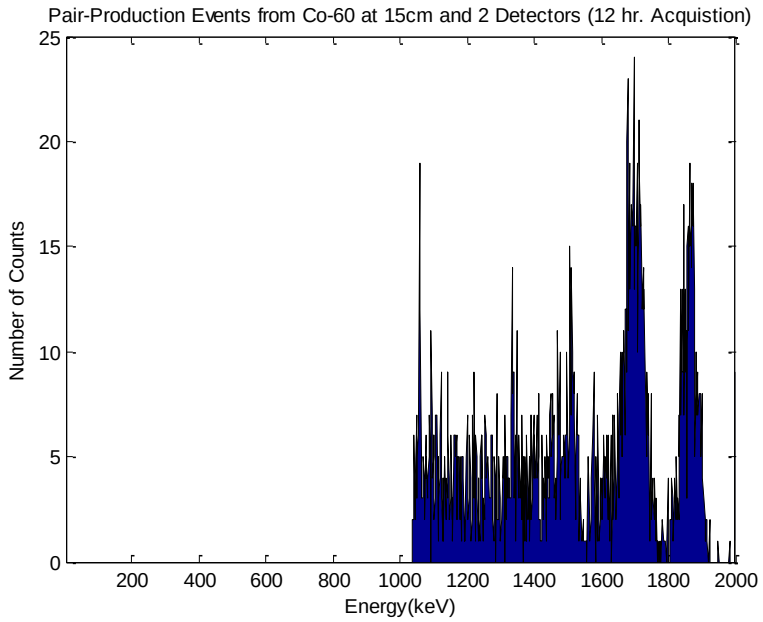

Figure 49. Pair-production events from a Co-60 acquisition for 12 hours at 15cm from two detectors. Plotted in 1024 bins. Events are called pair-production events if the 511 keV annihilation gamma deposited in detector 2.

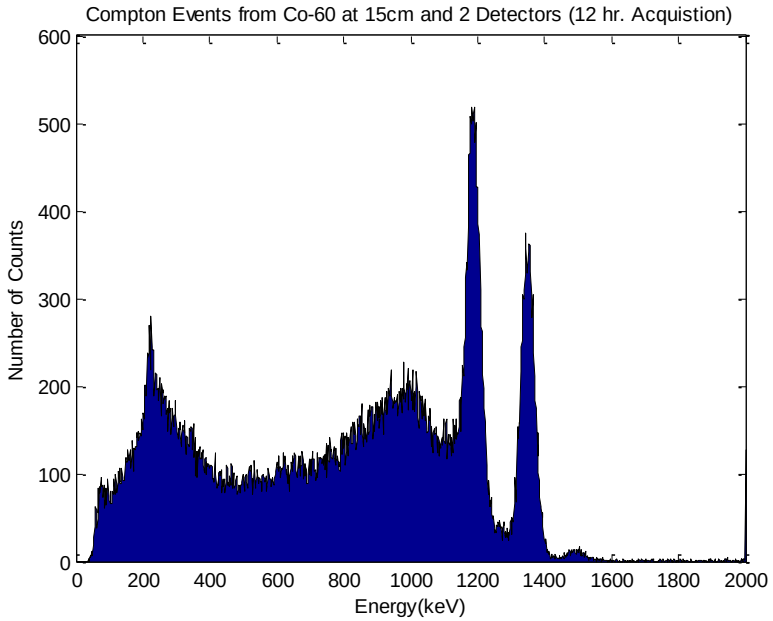

Figure 50. Compton events from a Co-60 acquisition for 12 hours at 15cm from two detectors. Events are called pair-production events if the 511 keV annihilation gamma deposited in detector 2.

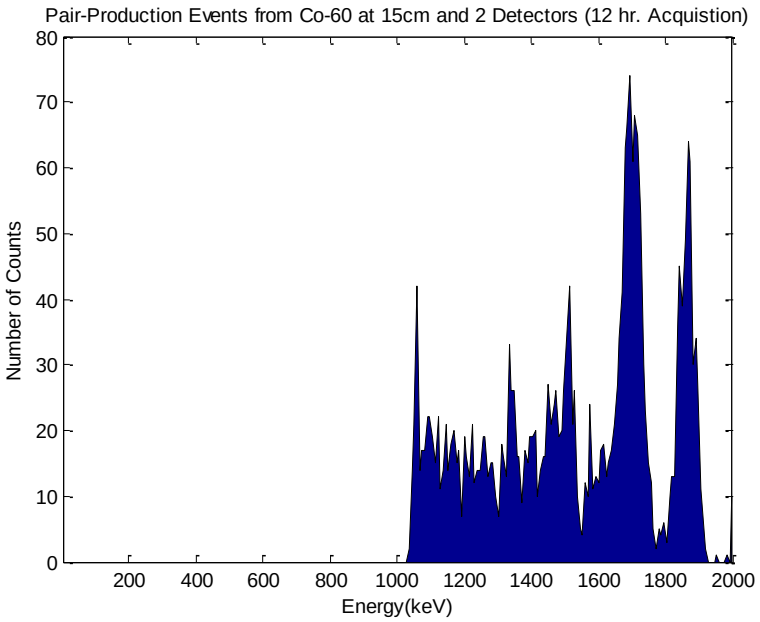

Figure 51. Pair-production events from a Co-60 acquisition for 12 hours at 15cm from two detectors. Plotted in 256 bins. Events are called pair-production events if the 511 keV annihilation gamma deposited in detector 2.

The number of counts recorded in the peaks of interest, with and without background, are shown in Table 30.

Table 30. Number of Counts in Peaks of Interest from 12 hr. Acquisition of Co-60 at 15cm and 2 Detectors with Annihilation Gamma Deposited in Detector 2

	Including Background		Less Background	
Peak (keV)	Compton Events	Pair Production Events	Compton Events	Pair Production Events
1173	11598 ± 107.7	137 ± 11.7	11392 ± 106.7	110 ± 10.5
1332	8470 ± 92.0	188 ± 13.7	8339 ± 91.3	173 ± 13.2
1436	585 ± 24.2	248 ± 15.7	355 ± 18.8	618 ± 24.9
1684	N/A	628 ± 25.1	N/A	625 ± 25
1843	N/A	423 ± 20.6	N/A	422 ± 20.5

While the two chance Compton peaks still dwarf the true pair-production peaks, the true pair-production peaks can at least be distinguished when knowing the ordering of the pulses. The two element detector, over the period of 12 hours should have had 3.4

E+7 gammas emitted by the Co-60 source in the solid angle subtended by the detector. Of those, there should have been a total of 2.3 E+4 pair-production events contributed by the 1173 keV gamma and 1.2 E+5 pair-production events contributed by the 1332 keV gamma. Dividing the number seen within 3% of the FEP by the number that are predicted by theory gives efficiencies of 0.61% for the 1173 keV peak and 0.16% for the 1332 keV peak. The efficiency for the 1684 keV chance Compton peak is 0.20% and for the 1843 keV peak is 0.17%.

The data was processed once again but this time only allowing the 511 keV annihilation gamma to be deposited in detector 1. The data processed in this way provides a way to see how many chance Compton events occur as the true pair-production events should be eliminated. The Compton events spectrum is similar to that of the other two methods of processing data and will not be presented below, however the pair-production spectrum is very different and is shown in Figure 52. The number of counts recorded in the peaks of interest are shown in Table 31.

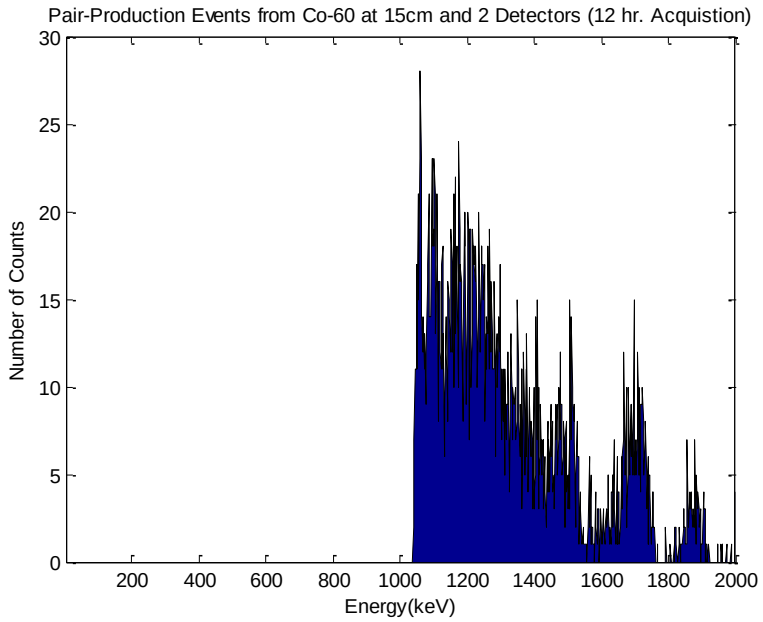

Figure 52. Pair-production events from a Co-60 acquisition for 12 hours at 15cm from two detectors. Events are called pair-production events if the 511 keV annihilation gamma deposited in detector 1.

Table 31. Number of Counts in Peaks of Interest from 12 hr. Acquisition of Co-60 at 15cm and 2 Detectors with Annihilation Gamma Deposited in Detector 1

	Including Background		Less Background	
Peak (keV)	Compton Events	Pair Production Events	Compton Events	Pair Production Events
1173	11868	515	11662	490
1332	8789	391	8658	379
1436	583	337	353	328
1684	N/A	323	N/A	322
1843	N/A	98	N/A	97

When treating a 511 keV annihilation gamma as any 511 keV gamma deposited in detector 1, a lot of background is produced that makes spectroscopy of the source nearly impossible. The chance Compton peaks at 1684 keV and 1843 keV still dominate the spectrum as the only prominent peaks. The efficiency for the 1684 keV chance Compton peak is 0.10% and for the 1843 keV peak is 0. 032%. The lack of any true pair-

production peaks shows that in order to have even a remotely decent pair-production spectrometer, the order of the pulses needs to be known. Figure 52 also shows the background pulses that are seen by counting pulses that are not ordered in the correct fashion are a significant detriment to the ability to perform spectroscopy.

V. 4. Four Element Detector Directional Efficiency

Before characterizing a four element detector for directional efficiency, the background was characterized in a similar manner as the two element detector. A 12 hour acquisition yielded the pair-production background spectrum in Figure 53 and the Compton background spectrum in Figure 54. The number of counts in the peaks of interest are tabulated in Table 32.

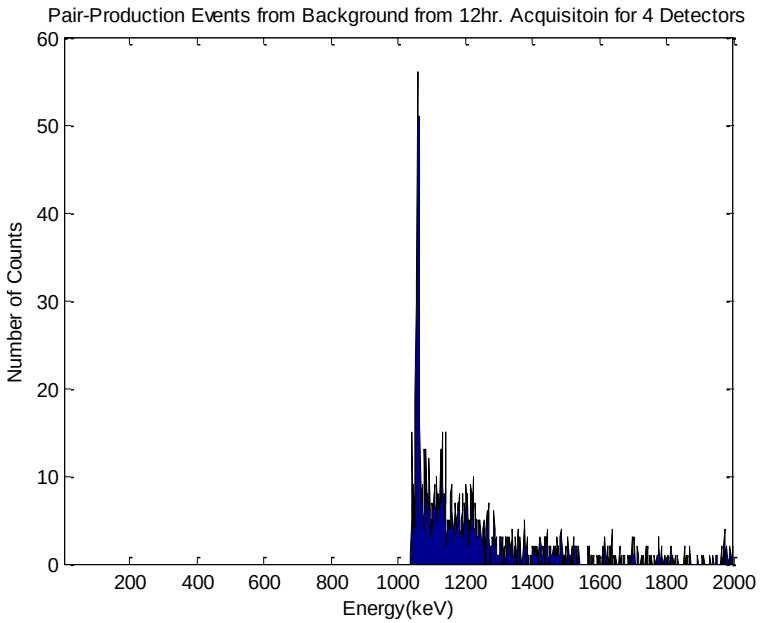

Figure 53. Pair-production event background spectrum from a 12 hour acquisition using a 4 element detector.

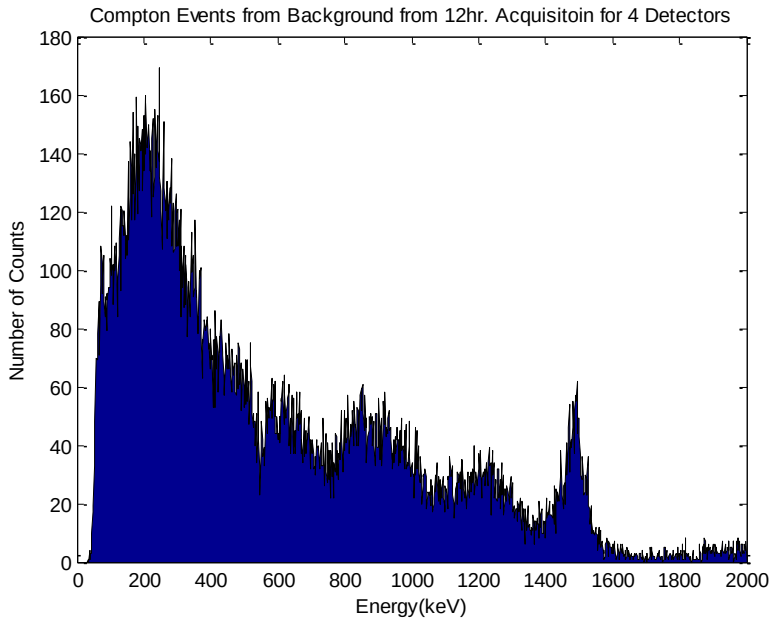

Figure 54. Compton event background spectrum from a 12 hour acquisition using a 4 element detector.

Table 32. Number of Counts in Peaks of Interest from 12 hr. Background Acquisition and 4 Detectors

Peak (keV)	Compton Events	Compton Events/hr	Pair-Production Events	Pair-Production Events/hr
1173	955 ± 30.9	79.58	199 ± 14.1	16.58
1332	669 ± 25.9	55.75	71 ± 8.4	5.917
1436	1415 ± 37.6	117.92	82 ± 9.1	6.83
1684	N/A	N/A	31 ± 5.6	2.58
1843	N/A	N/A	20 ± 4.5	1.67

The directional efficiency was characterized following the procedure described in Chapter 4. Acquisitions were one hour in length using Co-60 with an activity of 6.7 μCi and at various angles and distances. The number of counts reported on the following plots, are once again the number of counts within 3% of the FEPs of interest, namely 1173 keV and 1332 keV. The results of the acquisitions at 30 cm for $\phi = 0°$ and $\phi = 45°$

are shown in Figure 55and Figure 56, respectively. The error bars indicated on the plots below are computed as

$$\sigma = \sqrt{N} \tag{2.11}$$

where is the standard deviation and is the number of counts recorded for that peak of interest [4, 84].

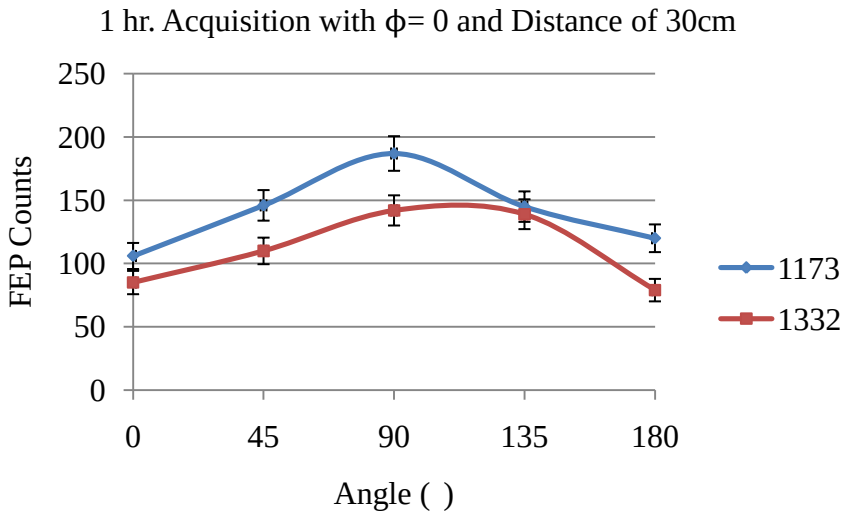

Figure 55. Number of pair-production counts within 3% of the full energy peaks (FEP) of Co-60 for an in plane one hour acquisition at a distance of 30cm and at various angles.

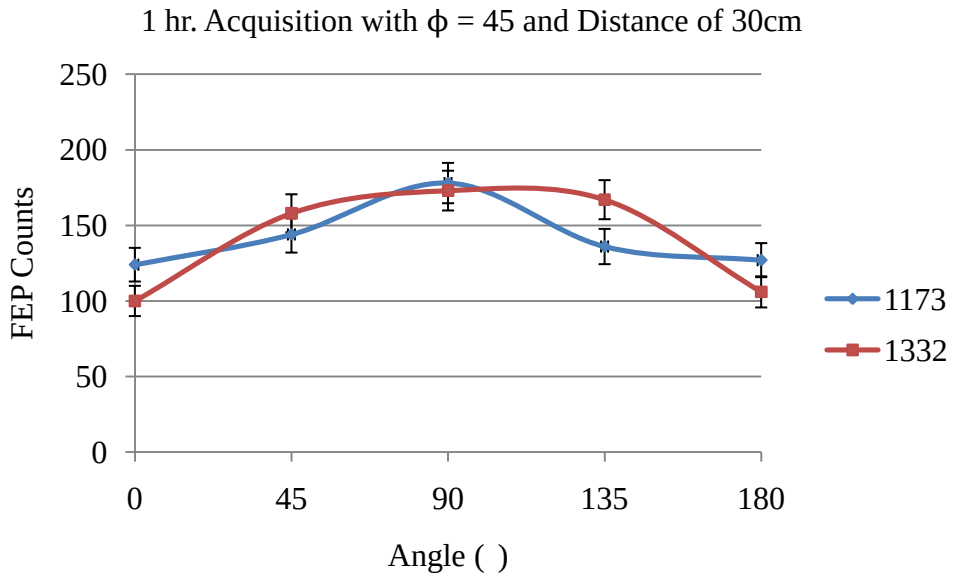

Figure 56. Number of pair-production counts within 3% of the full energy peaks (FEP) of Co-60 for an out of plane one hour acquisition at a distance of 30cm and at various angles.

Typical spectra for the directional efficiency measurements are shown in Figure 57 for the pair-production spectrum and Figure 58 for the Compton spectrum.

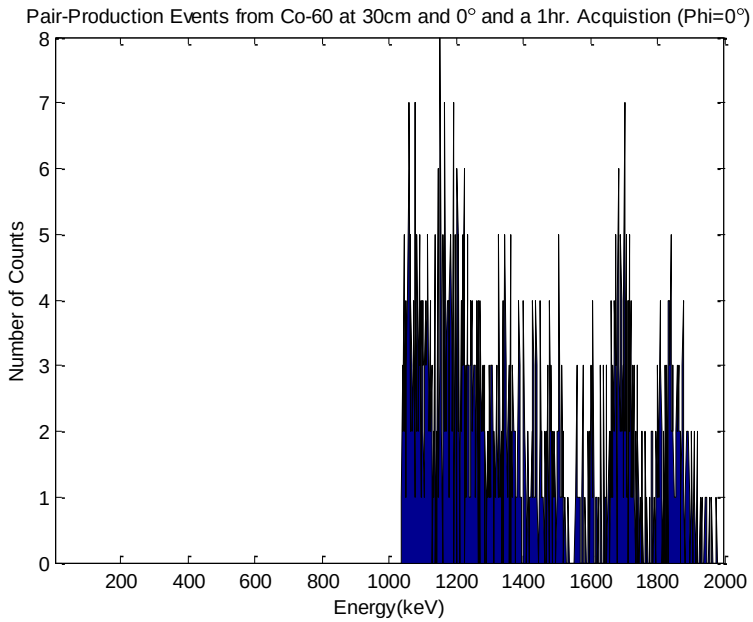

Figure 57. Pair-production events from a 1 hour Co-60 acquisition at 30cm. The orientation was 0° and ϕ = 0°.

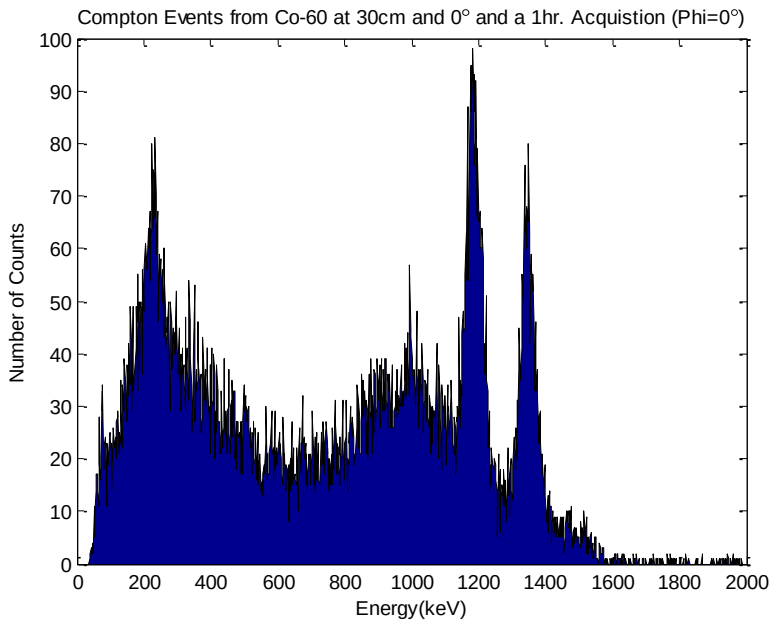

Figure 58. Compton events from a 1 hour Co-60 acquisition at 30cm. The orientation was 0° and ϕ = 0°.

As is shown in Figure 55 and Figure 56 where the directional efficiency measurements are plotted, the detector geometry is 76% more efficient at detecting 1173 keV pair-production gammas and 67% for 1332 keV pair-production gammas when oriented at 90° with respect to the source than when oriented at 0°. This is most likely due to the increased solid angle subtended by the detector in the 90° orientation. There is also a negligible difference in efficiency between ϕ = 0° and ϕ = 45°. Theory predicts that the 4 element detector should be 26% more efficient at detecting pair-production 1173 keV and 1332 keV gammas when the orientation is 90° than when it is 0°. The discrepancy between theory and the laboratory experiments are likely due to the beta continuum which was not taken into account for the theory calculations as well as the small angle Compton scattering that the gammas undergo before pair-production.

The trends mentioned above hold true as the 4 element detector is moved farther from the source. Measurements were taken at 30 cm, 60 cm and 90 cm for $\phi = 0°$ and $\phi = 45°$. The orientation of the 4 element detector to the source was changed in 45° increments from 0° to 180°. The results of these measurements are shown in Figs. 54 – 57.

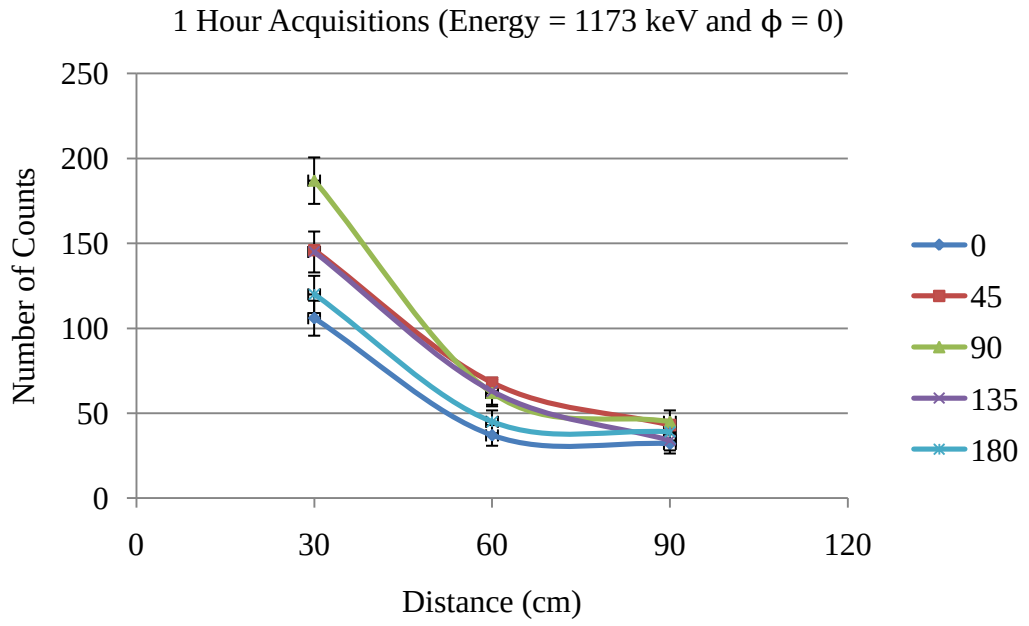

Figure 59. Number of pair-production counts within 3% of the 1173 keV FEP of Co-60 for an in plane one hour acquisition at a distance of 30cm, 60cm and 90cm at various angles.

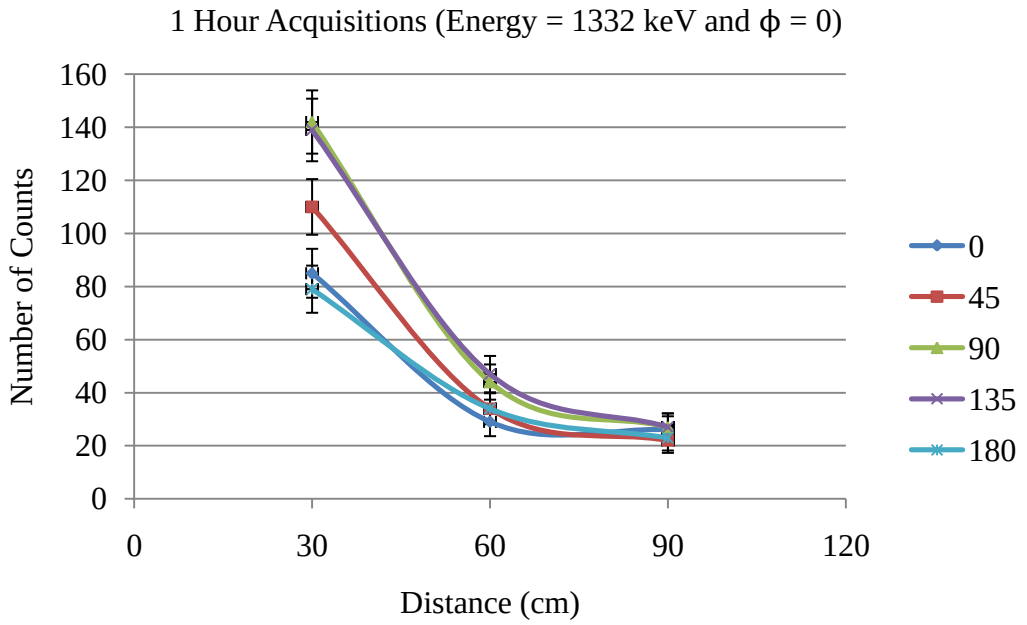

Figure 60. Number of pair-production counts within 3% of the 1332 keV FEP of Co-60 for an in plane one hour acquisition at a distance of 30cm, 60cm and 90cm at various angles.

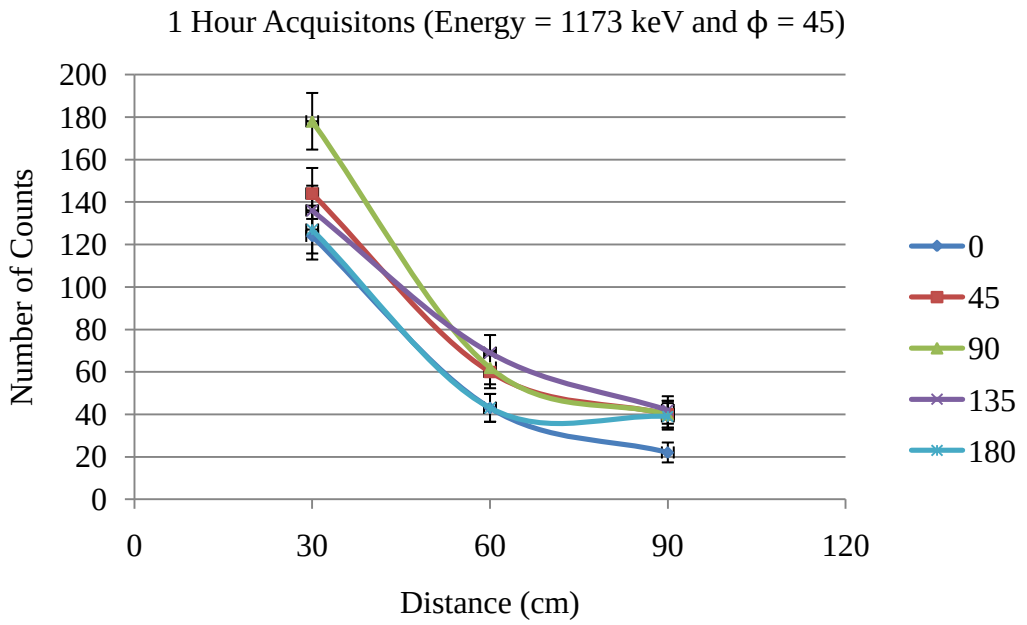

Figure 61. Number of pair-production counts within 3% of the 1173 keV FEP of Co-60 for an out of plane one hour acquisition at a distance of 30cm, 60cm and 90cm at various angles.

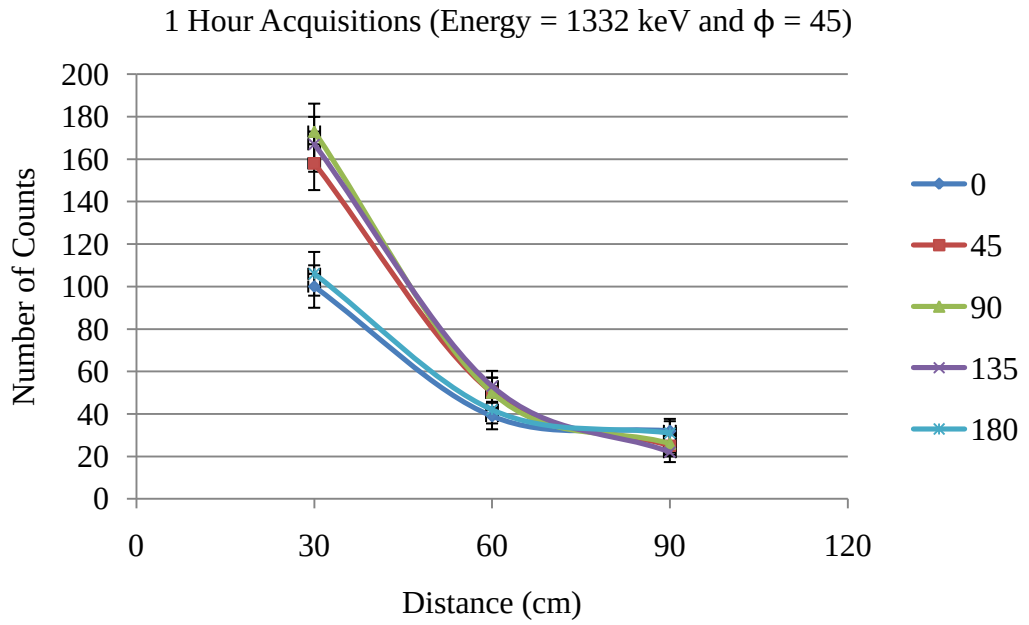

Figure 62. Number of pair-production counts within 3% of the 1332 keV FEP of Co-60 for an out of plane one hour acquisition at a distance of 30cm, 60cm and 90cm at various angles.

The numbers of counts in the peaks of interest falls off as the detector is moved farther from the source due to the smaller solid angle subtended by the detectors. The efficiencies were calculated in the manner described previously for the 0° and 90° orientations with ϕ = 0° and are tabulated in Table 33. The extended efficiency calculations that take into account the probability of capturing one of the annihilation photons in another detector are shown in Table 34.

Table 33. Efficiency Calculations for Pair Production

	Orientation (°)	0		90	
	Energy (keV)	1173	1332	1173	1332
d (cm)	27.5	6.80%	1.03%	9.45%	1.36%
	57.5	10.39%	1.54%	13.68%	1.84%
	87.5	20.84%	3.20%	23.03%	2.61%

Table 34. Extended Efficiency Calculations for Pair Production

	Orientation (°)	0		90	
	Energy (keV)	1173	1332	1173	1332
d (cm)	27.5	37.59%	5.70%	52.23%	7.49%
	57.5	56.92%	8.50%	75.61%	10.14%
	87.5	114.29%	17.69%	128.57%	14.44%

Table 35. Efficiency Calculations for Chance Compton Events

	Orientation (°)	0		90	
	Energy (keV)	1684	1843	1684	1843
d (cm)	27.5	0.24%	0.22%	0.30%	0.26%
	57.5	0.37%	0.33%	0.44%	0.35%
	87.5	0.74%	0.68%	0.74%	0.50%

The efficiency of the 4 element detector for detecting a pair-production event created by a 1332 keV gamma is noticeably lower than that for the 1173 keV gamma. This is most likely due to the increased background that is built up around the 1173 keV peak. This background likely has to do with the beta continuum produced by the intrinsic radiation of the detectors. If the system observes a supposed 511 keV annihilation gamma ray and not the incident gamma energy deposited in the first detector, but a coincident beta decay, it would get reported as a pair-production event. According to the data in Table 33, the efficiency of detecting a pair-production event increases as the distance increases. This is likely due to the reduction of all of the clutter from the small-angle Compton scattering taking place.

Lastly, a measurement was taken of a 15.5 mCi Co-60 source at a distance of 10 feet for 3 hours to see if the increased number of counts would allow any visible full energy peaks to form on the pair-production events spectrum. A plot with well formed peaks would provide the ability to perform pair-production spectroscopy and confirm or

deny the identification, not location, of a pair-production capable radioactive source. The results of this measurement are shown in Figure 63.

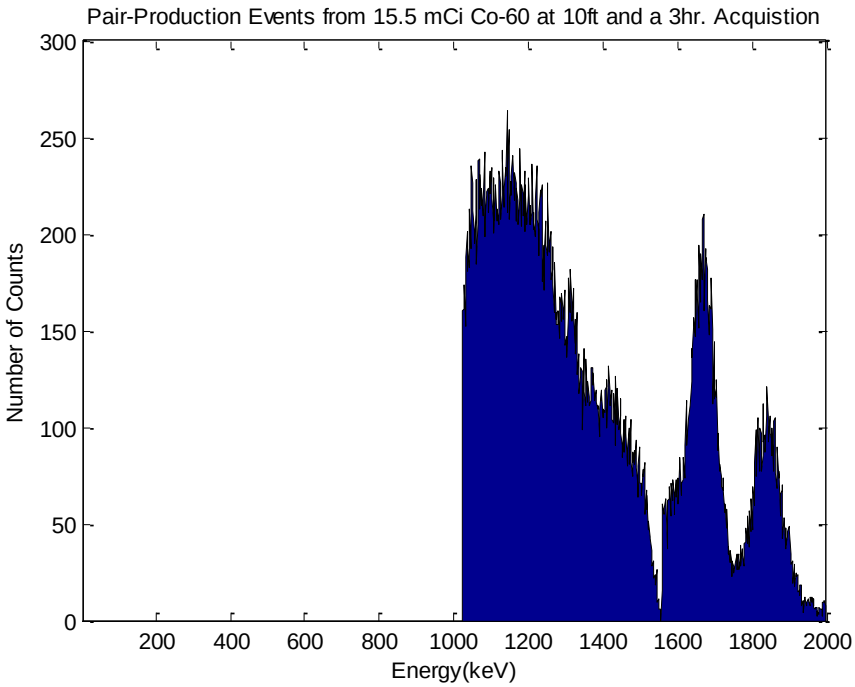

Figure 63. Pair-production events spectrum from a 15.5 mCi Co-60 source measurement at 10 feet for 3 hours.

Unfortunately, there are no significant peaks that formed at 1173 keV and 1332 keV, though the expected chance Compton peaks did form. There are slight indications of potential peaks forming at 1173 keV and 1332 keV; however, rebinning the data did not prove useful. A longer acquisition resulting in more counts would likely display similar results with peaks that cannot be resolved, only with counts which are greater in magnitude.

V. 5. Simulation Results

A simulation was first completed to benchmark the results of the pair-production laboratory data experiments. This was done by simulating the system to be in the 0° orientation with $\phi = 0°$. A 3 mm circular planar source of Co-60 was simulated by shooting 6.2E+6 random gamma rays over an angle equal to 5.5° at the 4 element detector. The simulated spectrum is shown in Figure 64.

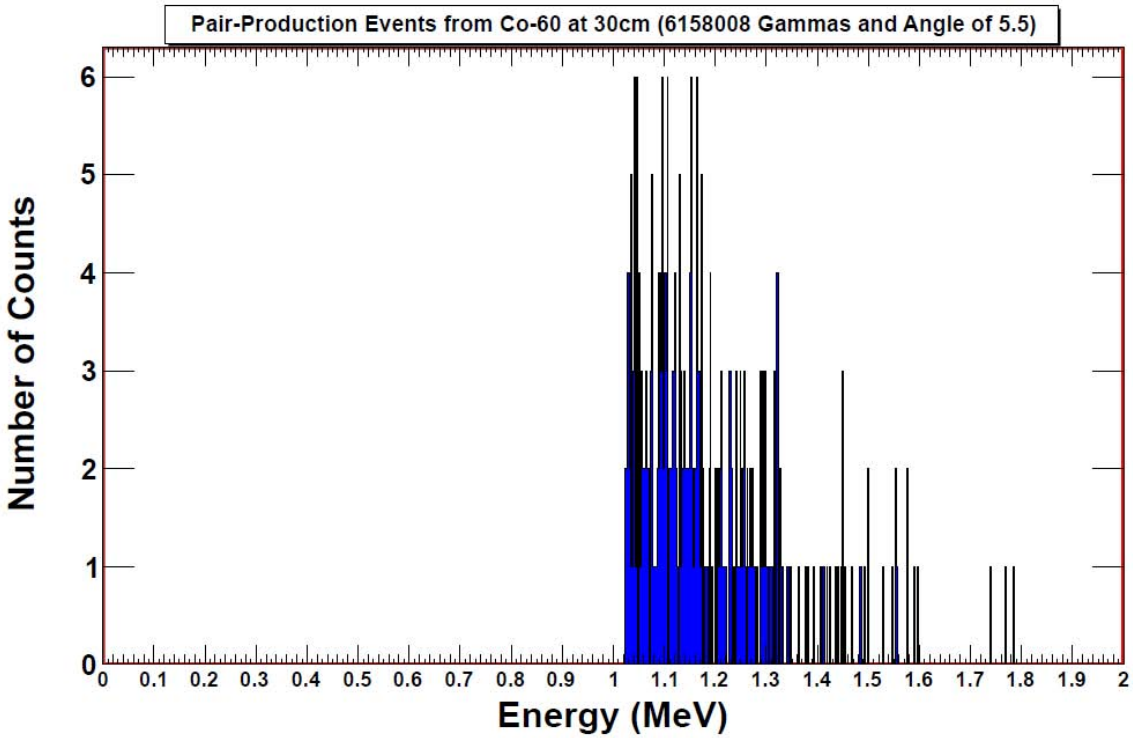

Figure 64. Simulated spectrum of pair-production events from 6158008 Co-60 gamma rays shot at a 4 element detector. This was done in a 0° orientation with $\phi = 0°$.

There were 69 counts within 3% of the 1173 keV peak and 34 counts within 3% of the 1332 keV peak. The counts located in the 1450 keV range are caused by gammas undergoing Bremsstrahlung interactions and not the intrinsic radioactivity of La-138 because the intrinsic radiation was not modeled. The number of counts in the FEPs of

interest are not the same as those seen in the laboratory. The fewer simulated counts are most likely due to the unmodeled background and intrinsic radiation. The background spectrum taken for 12 hours in section V. 4 yielded count rates of approximately 17 counts/hour in the 1173 keV peak and 6 counts/hour in the 1332 keV peak. This simulation is supposed to represent a 1 hour acquisition at 30 cm. Subtracting the background counts from the results of the 1 hour lab acquisition yields 89 counts in the 1173 keV peak and 79 counts in the 1332 keV peak. Another contribution to the discrepancy is the near coincidence that the Co-60 gamma rays in the lab frame of reference are emitted in. The coincidence between the 1173 keV and 1332 keV gamma rays is 0.9 picoseconds [**22**]. This is well within the 100ns coincidence window that the Igor Pro software is operating at. In the simulation, the gamma rays are emitted one at a time, thus, not allowing different gamma rays to contribute to the same event. This also most likely causes the differences in the sizes of the chance Compton peaks at 1684 keV and 1843 keV.

The simulation was further tested by duplicating the measurements taken at the same distance but with the orientation of the 4 element detector varying in 45° increments with respect to the source. As stated earlier, it is very difficult to calculate the number of particles that should be emitted from the source to replicate the laboratory experiments. To validate the directional efficiency measurements, the number of particles emitted was kept the same and the 4 element detector was oriented in 0°, 45°, and 90°. The detector was not oriented at 135° and 180° as those are the same as the 0° and 45° orientations for the simulation. This is because only the scintillation crystals were modeled and there is not a PMT to interfere when simulating at 135° and 180°. Simulations with those

orientations would only be useful if the photomultiplier tube and detector mount were modeled as well. 15E+6 Co-60 gamma rays were simulated over a 10° angle at the detector in the various orientations. The results are shown in Figure 65.

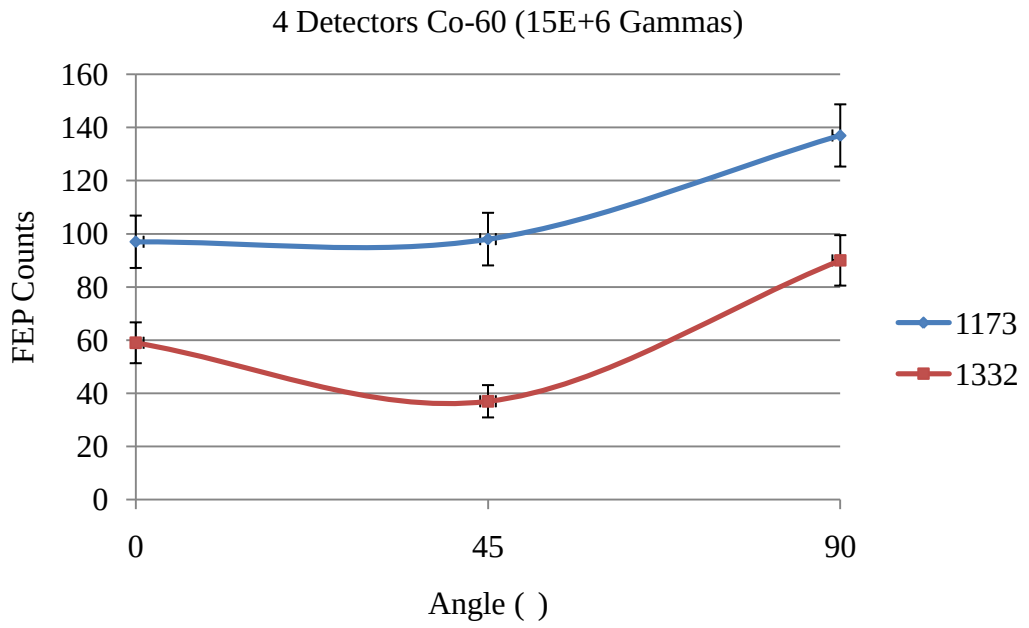

Figure 65. Number of pair-production counts within 3% of the full energy peaks (FEP) of Co-60 for an in plane simulation at a distance of 30cm and at various angles for a 4 element detector. 15E+6 gamma rays were emitted by the source in an angle of 10°.

The simulations show that the 4 element detector is 81% more efficient at detecting 1173 keV pair-production gammas and 86% for the 1332 keV gammas when the orientation is 90° than when it is 0°. These are similar to the efficiencies found in the laboratory where they were 76% for the 1173 keV pair-production gammas and 67% for the 1332 keV gammas but far from the 26% predicted in theory.

Figure 67 shows the results from a Co-60 source at 30cm with 15E+6 gamma rays emitted at a 4 element detector and a 6 element detector in an 11° angle. The 6 element detector had the orientation shown in Figure 66.

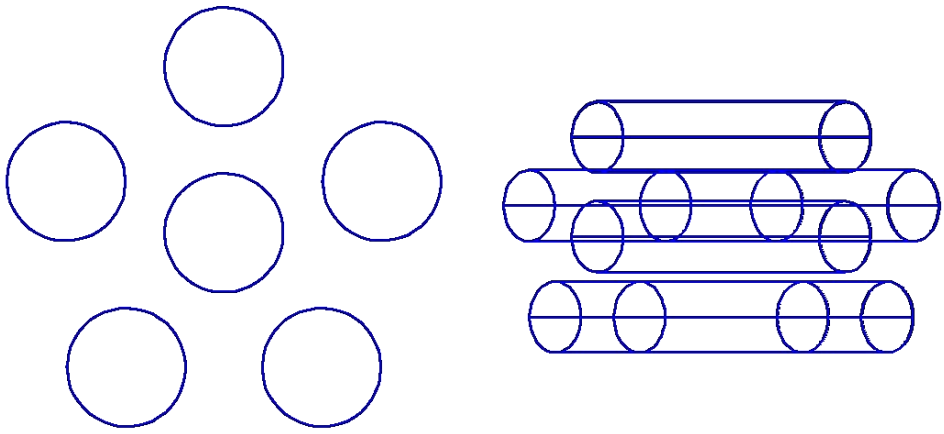

Figure 66. Geometry used for the simulated 6 element detector.

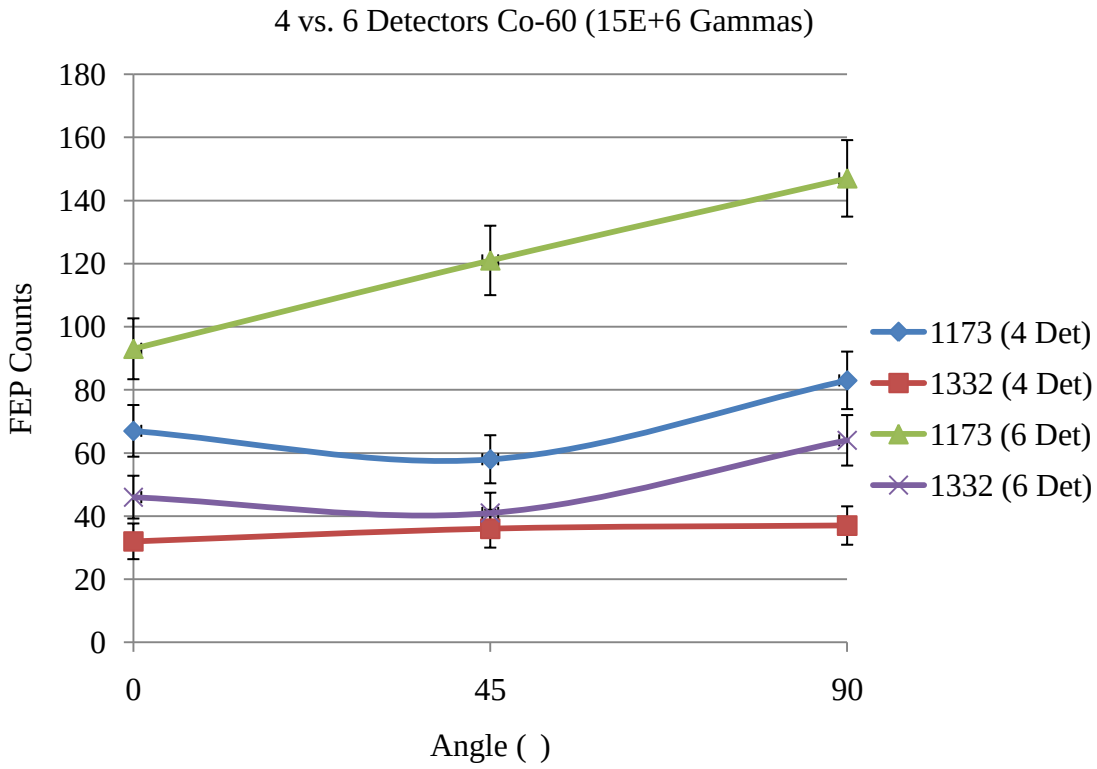

Figure 67. Number of pair-production counts within 3% of the full energy peaks (FEP) of Co-60 for an in plane simulation at a distance of 30cm and at various angles for a 4 element detector. 10E+6 gamma rays were emitted by the source in an angle of 11°.

The 6 element detector is a much more efficient pair-production spectrometer and allows the ability to perform spectroscopy by providing distinguishable peaks more quickly than the 4 element detector. The improvements are shown in Table 36. The average improvement at detecting the 1173 keV pair-production gammas is 75% and it is 44% at detecting the 1332 keV pair-production gammas.

Table 36. 4 Detector vs. 6 Detector Directional Efficiency Simulation Results

	4 Detectors		6 Detectors		Improvement (1173 keV)	Improvement (1332 keV)
Angle	1173 keV	1332 keV	1173 keV	1332 keV		
0	67 ± 8.2	32 ± 5.7	93 ± 9.6	46 ± 6.8	39%	44%
45	58 ± 7.6	36 ± 6	121 ± 11	41 ± 6.4	109%	14%
90	83 ± 9.1	37 ± 6.1	147 ± 12.1	64 ± 8	77%	73%

Simulations were completed using a source that emitted 2.6 MeV gammas. A 6 element detector with the same size $LaBr_3$(Ce) crystals, was used with a source emitting 10E+6 gamma rays at an angle of 11°. The simulated pair-production spectrum is provided below in Figure 68.

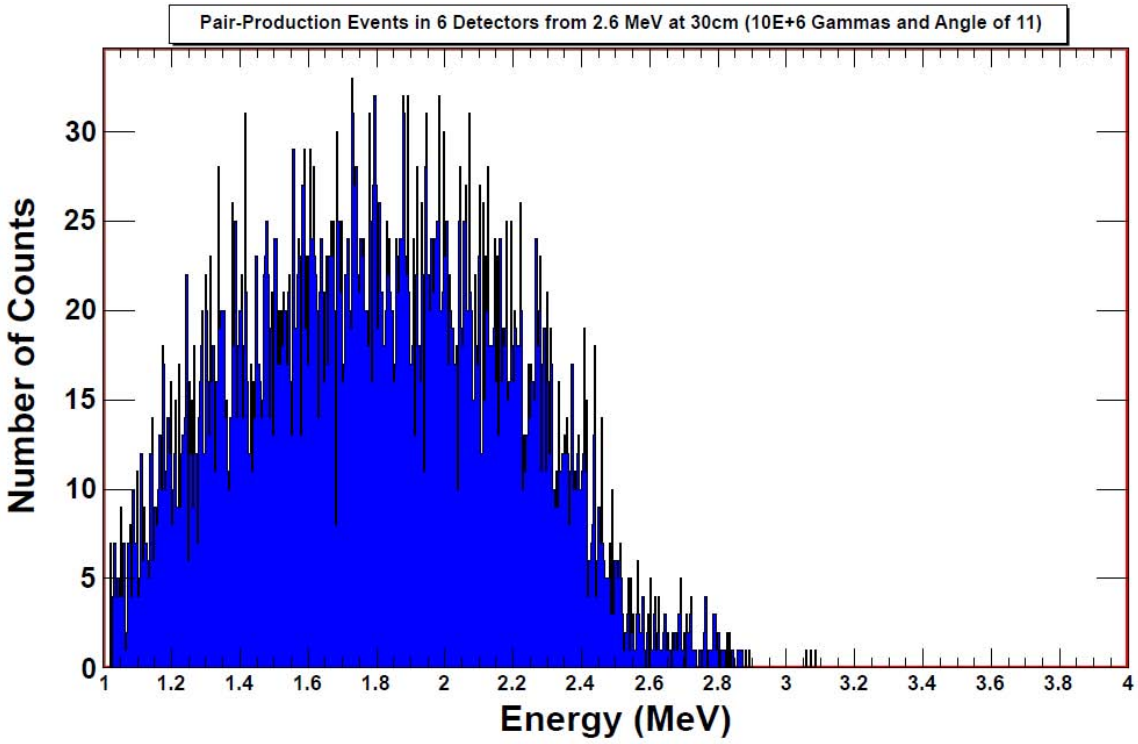

Figure 68. Simulated spectrum of pair-production events from 10E+6, 2.6 MeV gamma rays shot at a 6 element detector. This was done in a 0° orientation with $\phi = 0°$.

The inability to distinguish a peak at 2.6 MeV is readily apparent. An attempt to remedy this with larger detectors proved helpful, though it still did not solve the problem. The spectrum shown in Figure 69 is from 5E+6, 2.6 MeV gamma rays incident on 6 $LaBr_3$ detectors that are 8 cm in diameter and 10 cm in length, reflecting an increase in volume by a factor of 32. The angle that the source gamma rays were emitted in was 25°. There were 203 counts within 3% of the 2.6 MeV peak. The large hump to the left of the 2.6 MeV mark is most likely due to the Compton scattering of the higher energy gamma rays before the pair-production interaction occurs. If the detection of both of the 511 keV annihilation photons was demanded of the detector, then the pair-production spectrum should change into more of a delta function at 2.6 MeV. Modifying the system to do so, would greatly decrease the efficiency of the detector.

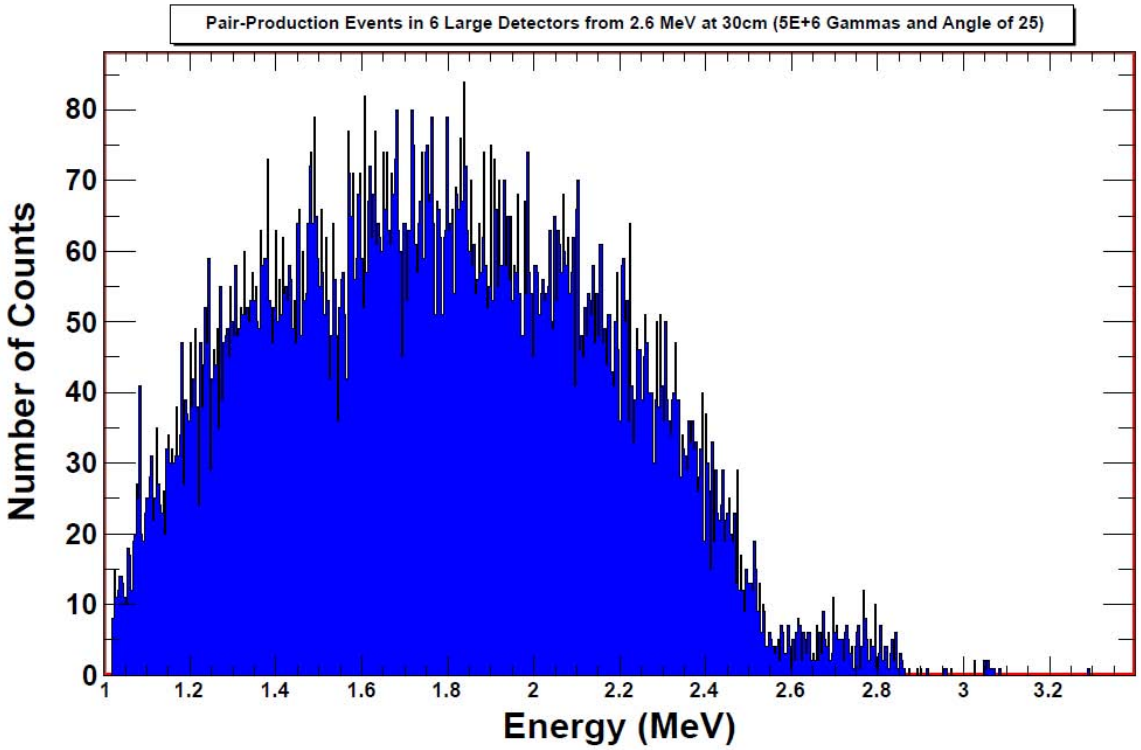

Figure 69. Simulated spectrum of pair-production events from 5E+6, 2.6 MeV gamma rays shot at a 6 element detector made up of substantially larger crystals. This was done in a 0° orientation with $\phi = 0°$.

A comparison by operating a detector as a double coincident detector, as was done for the rest of this project, and a triple coincident detector that mandated detection of both 511 keV annihilation photons. A 5 element detector with a geometry shown in Figure 70 was used. A Co-60 source was simulated and the detection system had 230E+6 gammas incident on it. The results of operating the detection system in a double coincident mode are shown in Figure 71. The results of operating the detection system in a triple coincident mode are shown in Figure 72. The triple coincident operation results were rebinned to bring out the features of the spectrum and are shown in Figure 73.

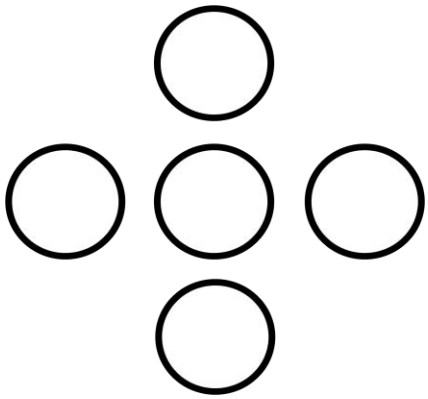

Figure 70. Geometry used to compare double coincident operation to triple coincident operation of the detection system.

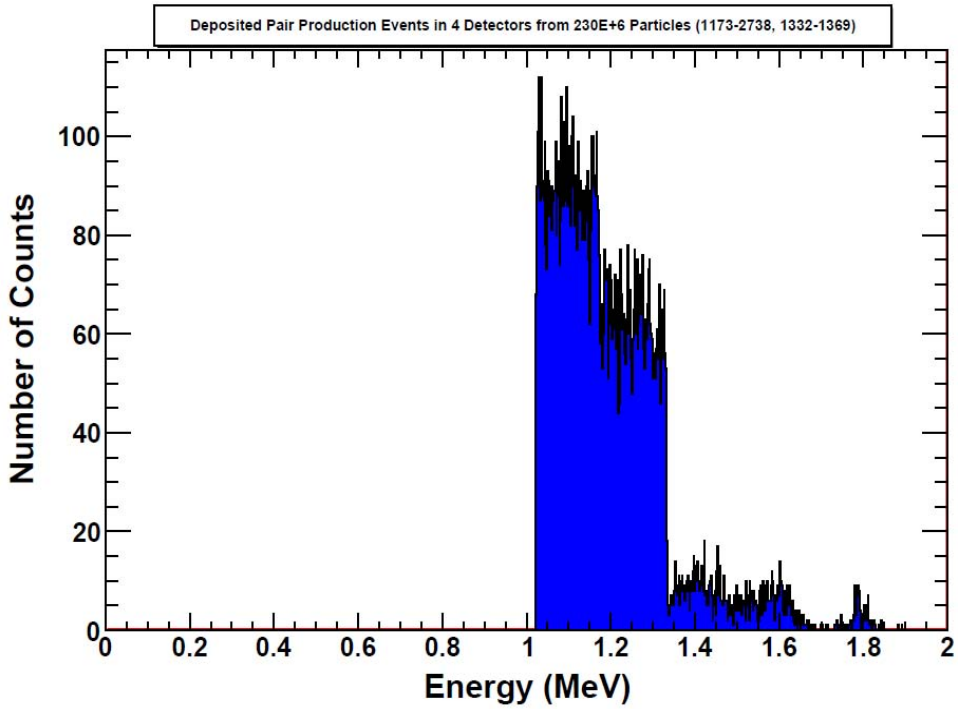

Figure 71. Simulated spectrum of pair-production events from 230E+6, Co-60 gamma rays shot at a 5 element detector. This was done in a 0° orientation with $\phi = 0°$. The detection system was operated in double coincidence mode.

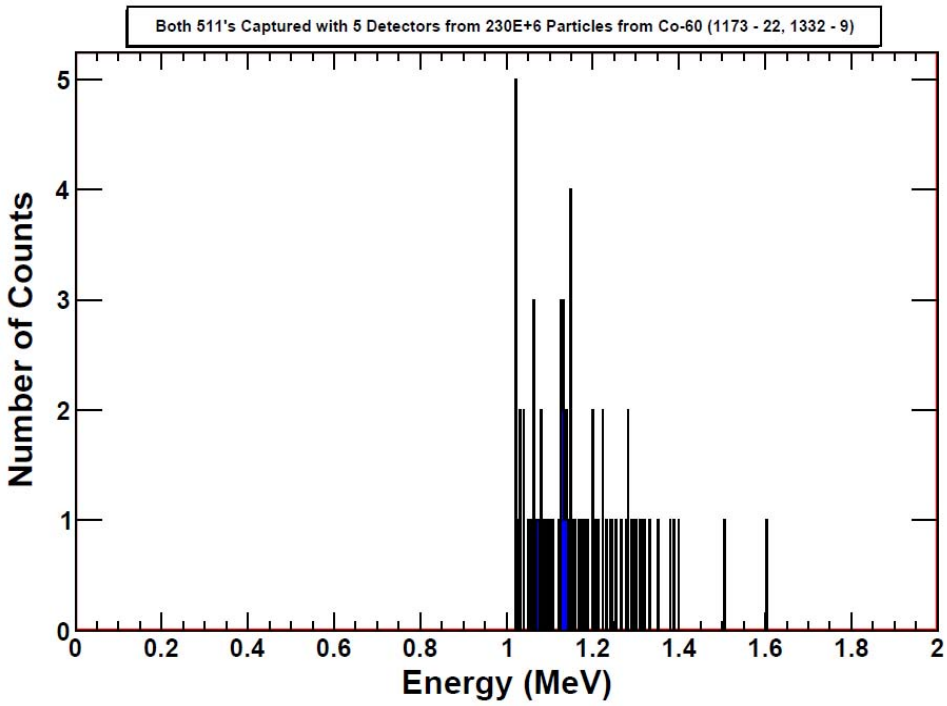

Figure 72. Simulated spectrum of pair-production events from 230E+6, Co-60 gamma rays shot at a 5 element detector. This was done in a 0° orientation with $\phi = 0°$. The detection system was operated in triple coincidence mode.

Operating the detection system in triple coincident mode proved to be 124 times less efficient for 1173 keV pair-production gamma detection and 154 times less efficient for 1332 keV pair-production gamma detection when compared to operating in double coincident mode. Figure 73 shows some features at 1173 keV and 1332 keV beginning to appear however many more events would need to be run for the features to become prominent.

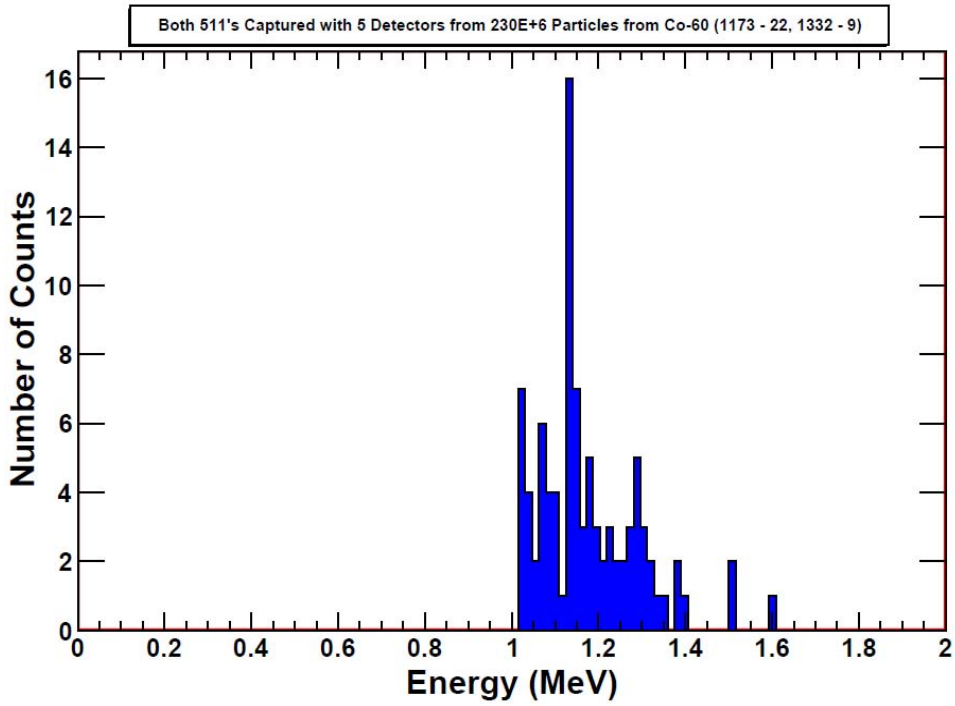

Figure 73. Rebinned simulated spectrum of pair-production events from 230E+6, Co-60 gamma rays shot at a 5 element detector. This was done in a 0° orientation with ϕ = 0°. The detection system was operated in triple coincidence mode.

V. 6. Compton Camera Performance

The intrinsic background that was presented earlier contributes to the degradation of the performance of the detector as a Compton camera as well. Characterizing this background and subtracting it from the image produced by the raw data of a run can help bring out the true image. The background was characterized for 10 minutes and the results are shown in Figure 74.

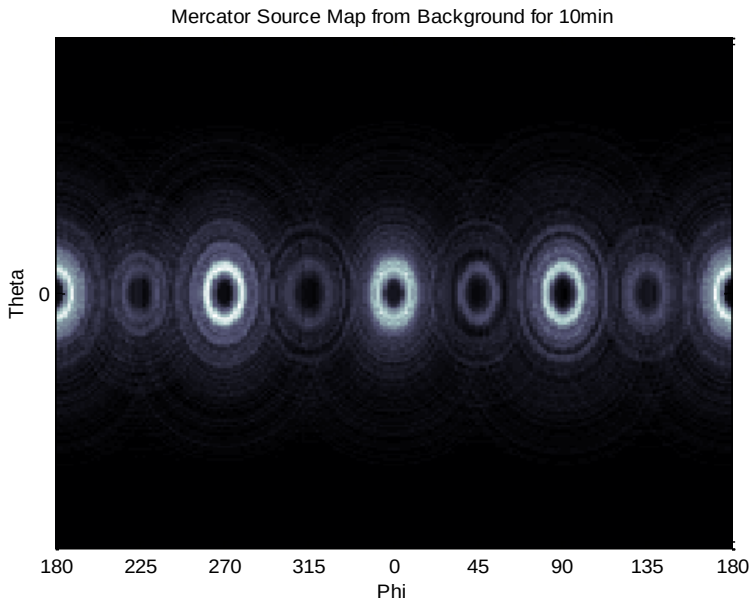

Figure 74. Shown is a Compton camera image from a 10 minute background characterization. The locations with the colors closer to white are more likely than those that are black.

The rings along the equator of the image are due to the assumption that the detectors are points and are all in the same plane. Each pixel and ring is 1.8° thick in the φ direction and 0.9° thick in the θ direction. There are twelve different cone directions that can be projected on the map. Eight of these cones are unique. The cones that point orthogonal to the detector, namely 0°, 90°, 180°, and 270°, are duplicated. The rings are also equally spaced at approximately 45° intervals because of the Compton scattering options for the incident gamma rays.

The performance of operating the detector as a Compton camera was evaluated by processing the data that was taken over the period of 10 minutes of the 6.7 μCi Co-60 source as the detectors were rotated from 0° to 180° in orientation with respect to the source that was at a distance of 30 cm from the detector. This was done for in-plane and out-of-plane measurements.

The measurements taken that best illustrate the performance of the Compton camera are presented below. Figure 75 shows the measurement for θ=0° and ϕ=0°. Figure 76 shows the measurement for θ=0° and ϕ=45°. Figure 77 shows the measurement for θ=90° and ϕ=0°. Lastly, Figure 78 shows the measurement for θ=90° and ϕ=45°.

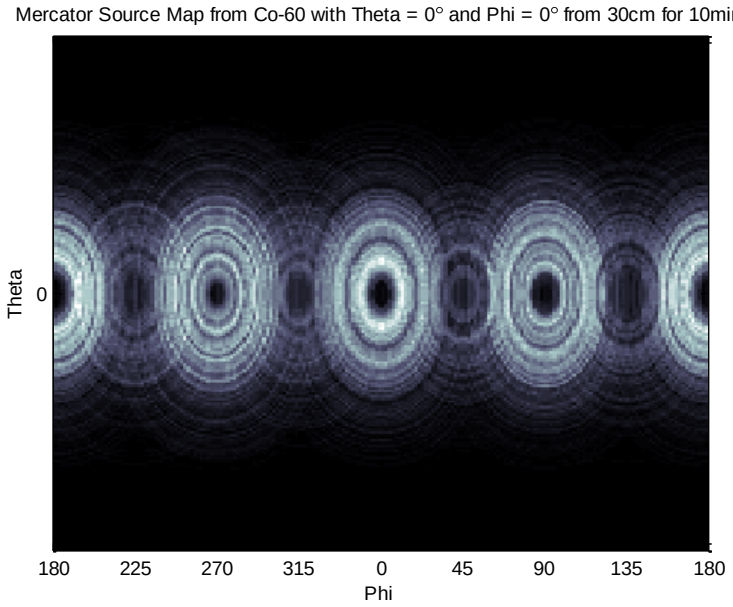

Figure 75. Shown is the image from operating the directional pair-production spectrometer as a Compton camera. The measurement taken was of a Co-60 source at a distance of 30cm for 10 minute at θ=0° and ϕ=0°. It is difficult to discern a dominating location, which is expected.

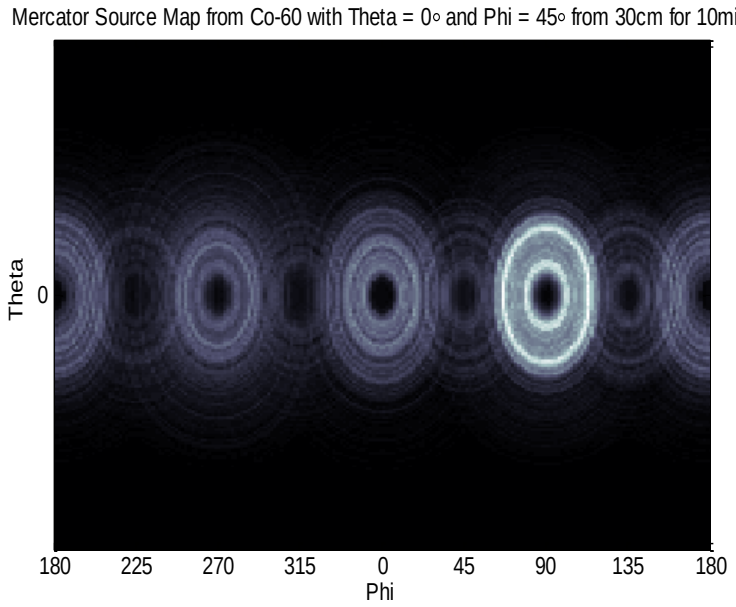

Figure 76. Shown is the image from operating the directional pair-production spectrometer as a Compton camera. The measurement taken was of a Co-60 source at a distance of 30cm for 10 minute at θ=0° and ϕ=45°. The dominating ring location is at ϕ=90°, indicating a source that is in the northern hemisphere of the map.

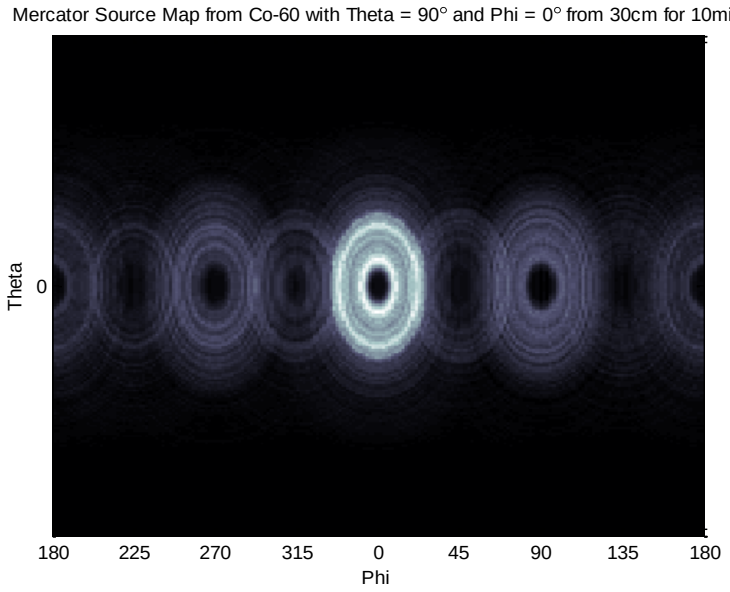

Figure 77. Shown is the image from operating the directional pair-production spectrometer as a Compton camera. The measurement taken was of a Co-60 source at a distance of 30cm for 10 minute at θ=90° and ϕ=0°. The dominating ring location is at ϕ=0°, indicating a source that is dead center on the right of the detector.

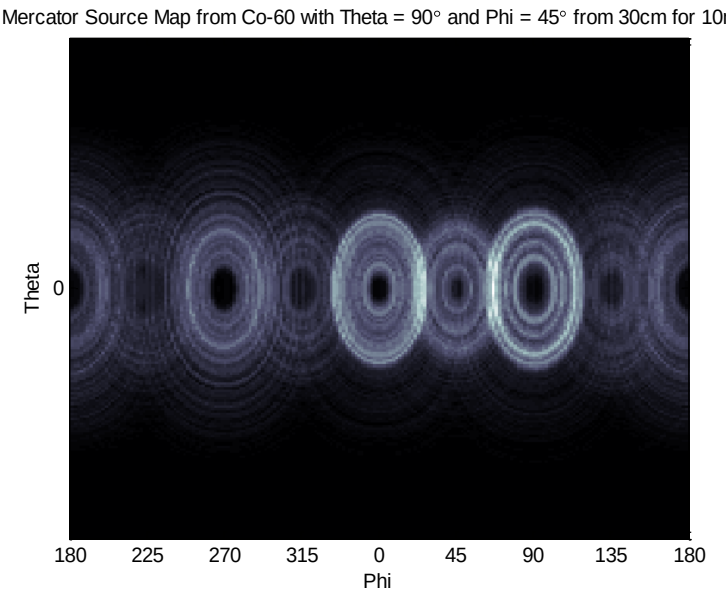

Figure 78. Shown is the image from operating the directional pair-production spectrometer as a Compton camera. The measurement taken was of a Co-60 source at a distance of 30cm for 10 minute at θ=90° and ϕ=45°. The dominating ring locations are at ϕ=0°, ϕ=45°, and ϕ=90°, indicating a source that is off to the right and elevated in the northern hemisphere of the detector.

It is apparent from Figs. 69 - 72 that the 4 detector array, can indeed function as a crude Compton camera. In Figure 75 the camera shows its inability to distinguish a direction when a source is directly in front or behind the detector, though it could be argued that the likelihood is that it is either in front of or behind the detector due to the near equivalent intensity rings located at 0°,90°,180°, and 270°. In Figure 76 the camera indicates that the source is most likely at 90°, while having strong rings at 0° and 180°. The camera's indication of having a source in the northern hemisphere and most likely one that is directly above it is correct as this was for a source located at θ=0° and ϕ=45°. The strongest indication of the Compton Camera's performance is shown in Figure 77 when measuring a source located at θ=90° and ϕ=0°. The image indicates, without much doubt that the source is located at ϕ=0°. Lastly, the source was moved to a

location of θ=90° and ϕ=45° and its image is shown in Figure 78. The image indicates a source that is in the upper-right hemisphere of the detector with its most intense rings at 0°, 45°, and 90°. The rings at the ϕ=45° location are noticeably more faint than those at the 0° and 90° locations as would not be expected as the source is at the 45° location. This is due to the likelihood of a gamma being able to deposit its energy in the two detectors that are diagonally opposed. If a gamma Compton scatters in the detector on the upper right of the detector array in a direction that would send it diagonally across the detector, it has a decreased chance of traversing that distance without depositing its energy in one of the adjacent detectors. The decreased probability is caused by the overlap of the detectors as shown in Figure 79 that was designed into the array to keep it as compact as possible.

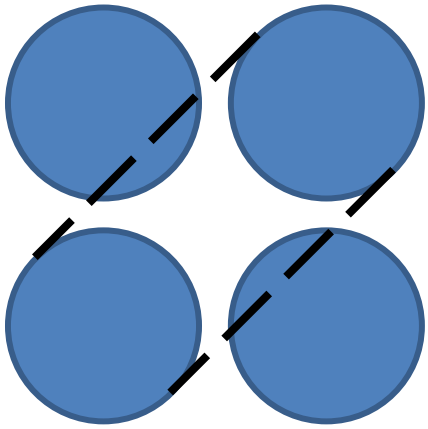

Figure 79. Shown is the overlap that is caused by the design of trying to make the detector array as compact as possible. A Compton scattered gamma ray is less likely to deposit its energy into the detector diagonal from it than it is to deposit its energy into an adjacent detector.

VI. Conclusions and Recommendations

VI. 1. Chapter Overview

The primary objective of this project was to determine the difference between a pair-production and Compton event, design and build the directional pair-production spectrometer, characterize the directional efficiency, create a model to benchmark the current design and provide insight into future designs and applications, and lastly evaluate the performance of the detector as a Compton camera. It is informative to examine the results of each of the objectives and form conclusions. Recommendations are made for future work.

VI. 2. Distinguish Difference between Compton and Pair-Production Events

Using the Tektronix DPO7104 oscilloscope as a data acquisition system to distinguish pair-production events from Compton events proved useful. The high sampling rate of the oscilloscope that provided one data point for every nanosecond gave the ability to filter out many non-coincident events that could get counted in a system such as the XIA system used in this project. This high sampling rate also gave the ability to better determine the order of the pulses, thus rejecting more of the chance Compton events. Data analysis by averaging the pulses for both event types resulted in no difference in pulse shape between the Compton and pair-production events. A disadvantage of the system was the method of transferring the data via Ethernet connection to a host PC reduced the potential capabilities of the oscilloscope. The settings that were used gave the highest vertical resolution. Increasing the sampling rate greatly reduced the vertical resolution, making accurate spectroscopy difficult at best. The Ethernet cable also reduced the achievable data rates to below other data acquisition

systems. The settings used could at best achieve 100 waveforms per second per channel. Operating a similar data acquisition code embedded on the oscilloscope may provide a higher vertical resolution, and if using a low overhead programming language such as C, a much faster data rate.

VI. 3. Pulse Order Impact and Directional Efficiency Characterization

Knowing which pulse occurred first in the detector proved very useful. This knowledge eliminated much of the background associated with the chance Compton events. The directional efficiency at detecting pair-production events in the detector was found to be at its greatest when the source was placed at a 90° orientation with respect to the detector. The efficiencies improved by 76% for the 1173 keV pair-production gammas and 67% for the 1332 keV pair-production gammas. This contradicts theoretical calculations that predict an improvement of only 26%. The increased efficiency at 90° was expected as the solid angle subtended by the detector when it is positioned at a 90° orientation is larger than when it is placed at 0°. The results, at least for the 0° and 90° orientations, follow theory as well. Directional efficiency for detecting pair-production events was not found to be impacted by moving the source to an out-of-plane position.

VI. 4. Simulation Evaluation

Replication of the theoretical number of gammas that should be emitted in the direction of the detector by the source based on the activity of the source and the solid angle subtended was found to be difficult in the Geant4 simulation. The method used to "aim" the conical source beam supposedly follows the law of cosines however that was found to not be correct. Thus, it was difficult to replicate the experiments in the Geant4

simulation as closely as would have been preferred. Despite these difficulties, the simulation replicated the directional efficiency trends seen in the laboratory experiments with the 90° orientation being more efficient than the 0° orientation. The efficiencies improved by 81% for the 1173 keV pair-production gammas and 86% for the 1332 keV pair-production gammas. These improvements were higher than were seen in the laboratory. The spectra, while somewhat sparse, were not as cluttered by the background associated with the La-138 decay in the detectors or the confusion of having the two characteristic gammas of Co-60, 1173 keV and 1332 keV, emitted in coincidence. Simulating a system with two more detector elements in a 6 element detector geometry should be more efficiently detect pair-production events by giving 10 pairs of detectors instead of the eight that are seen when using the four element detector as well as an increased solid angle subtended by the detector system. This assumption proved to be correct. The direction efficiency increased an average of 74% for the 1173 keV pair-production gammas and 44% for the 1332 keV pair-production gammas. Simulation of a high energy gamma source provided a spectrum that significantly differed from the spectra that were seen in simulations and experiments with the standard Co-60 source. The high energy spectra were extremely sensitive to the Compton events even when using much larger detectors. A 5 element detector was simulated as well and operated in double and triple coincidence modes. Double coincidence mode refers to the detection of one of the 511 keV annihilation photons and triple coincidence mode refers to the detection of both of the 511 keV annihilation photons. Operation in triple coincidence mode should greatly reduce the efficiency of the detector. This was seen when simulating 230E+6 Co-60 gamma rays. Operating the detector in triple coincidence

mode proved to be 124 times less efficient than when operating in double coincidence mode. The expected features of the spectrum are more prominent in the triple coincidence spectra than the double coincidence spectra, however, the greatly reduced efficiency makes operation in triple coincidence mode unfeasible for the intended application of this detection system.

VI. 5. Compton Camera Performance

The design for the directional pair-production spectrometer worked fairly well as a Compton camera. It was found that the detector was able to distinguish the location of the source as long as it was placed in the ϕ plane. Sources that were placed directly in front or behind the detector were nearly impossible to locate with much certainty.

VI. 6. Recommendations for Future Work

The design of a directional pair-production spectrometer such as the one in this project is a complex optimization problem with many tradeoffs. One such trade-off is the optimization of the detector to be a pair-production spectrometer or a Compton camera. Designing the detector to work better for one mechanism lessens its ability to perform well in the other capacity. A design study to see what geometry of the detectors performs best as a pair-production spectrometer and which as a Compton camera would be beneficial to see what features of the two may work well together and give the best hybrid solution. The performance of the Compton camera should also be characterized to be able to quantify the angular resolution that is important when trying to make any type of conclusions on the location of a source.

Performing spectroscopy on a higher energy source in the realm of the one simulated in this project would help to validate the results seen in the simulation. This would also drive further design and algorithm modifications if both annihilation photons must be detected to consider an event a pair-production event.

Using electronics that allow more input channels and a fast sampling rate to determine pulse orders would greatly benefit this project. This would help reduce much of the background observed in the pair-production spectrum. It would also make for a more accurate Compton camera, by providing more potential conical projections to be made. Not only would it benefit from the added channels, but knowing the true order of the event would be much better than using the highest probability for scatter as used in this project. Lastly, detectors that have an energy resolution of the LaBr3 detectors or better but do not have intrinsic radiation would also help to lessen the background in the pair-production spectrum.

Appendix A. Tektronix DPO7104 Oscilloscope Operation as Data Acquisition System

The following steps detail how to set up a Tektronix DPO7104 Oscilloscope as a four channel data acquisition system. The first step is to make sure the oscilloscope has the latest version of firmware and TekVISA installed. These can both be found on Tektronix's website. TekVISA is a collection of software for remote control of instruments regardless of instrument brand and physical connection. This software must first be installed on the host PC and the oscilloscope before the oscilloscope will accept remote control from the host PC [23]. TekVISA is only compatible with 32-bit operating systems and is the preferred VISA application for this project as the host PC used is a 32-bit system. If a 64-bit operating system is being used, Tektronix recommends using the National Instruments VISA application, NI VISA. The oscilloscope and host PC are connected by means of an Ethernet cable.

At this point, it is recommended to try to establish a connection utilizing TekVISA before attempting to establish one through MATLAB. This step will eliminate the possibility of an error in the MATLAB instrument driver or MATLAB script used to execute the data acquisition should problems arise later. The first step to establishing a connection with the oscilloscope is to boot up the oscilloscope and host PC and then connect the Ethernet cable. Creating a TCP/IP connection between the oscilloscope and host PC is begun by enabling the socket server on the oscilloscope by right clicking in the bottom right of the oscilloscope screen on the TekVISA LAN Server Control application and starting the server. The location of this command is shown in Figure 80.

Figure 80. The first step to starting a TCP/IP connection between the oscilloscope and the host PC is to start the socket server. The location on oscilloscope to start the socket server is shown.

Once the socket server is enabled, the OpenChoice Instrument Manager must be started on the host PC. Clicking on the update button in the Instrument List section of the GUI will indicate if the oscilloscope and host PC can ping each other. Clicking on the Identify button in the Instrument section of the GUI indicates if the host PC can identify the oscilloscope. If a secondary window with the oscilloscope information in it opens, a connection has been established. Clicking OpenChoice Talker Listener in the Applications and Utilities section of the GUI and then the Start Application or Utility button opens up the OpenChoice Talker Listener application. This application is useful to try out the various oscilloscope commands before inserting them into a MATLAB program.

To take multiple consecutive acquisitions, the host PC must also have MATLAB installed with the instrument driver for the particular model of oscilloscope being used. The Tektronix DPO7104, the model of oscilloscope being used for this project, unlike most of Tektronix's oscilloscopes, does not have a custom made instrument driver created by Mathworks. The instrument driver used with the DPO7104 is a modified

version of the TDS7104 driver. This modification was completed by Jesse Foster as part of an earlier project [24]. The instrument driver, a .mdd file, must be placed in the following subdirectory of the host computer: /Program Files/ MATLAB/R2010a/toolbox/instrument/instrument/drivers. To make sure the instrument driver for the oscilloscope is installed in the correct location and functioning properly, utilize MATLAB's instrument control toolbox. The GUI interface provided by MATLAB's instrument control toolbox, which can be accessed by typing "tmtool" in the command line interface of MATLAB, provides the capability to control most of the oscilloscope's functions and is the recommended tool for getting familiar with the command structure used to communicate with the oscilloscope. Expanding the Hardware and then the TCPIP (VX-11) menus should display text with the word Tektronix in it. Left click on that text and on the right of the screen a menu for control of the scope should open up. Click connect. In the same window as with the OpenChoice Talker Listener application, the operator should be able to issue the scope commands. The instrument control toolbox while reliable and useful, is not very flexible and does not permit multiple continuous acquisitions. To accomplish this, a MATLAB script was written. This script was begun by Jesse Foster as well [24]. In its current form, the MATLAB script used to control the oscilloscope utilizes Tektronix's FastFrame mode. The FastFrame mode allows the operator to have the oscilloscope capture multiple waveforms until the buffer gets filled. The buffer is then dumped by means of the Ethernet cable to the host PC. The host PC then writes the waveform data to a text file. The script also utilizes Tektronix's High Resolution mode. This mode increases the vertical resolution typically available by the oscilloscope by running at the highest

sampling rate of the digitizer and calculating and displaying the average of all the values in each sample interval [25]. This mode provides the maximum detail in the acquired waveform.

The following files are MATLAB scripts that allow a host PC to pull data off of the oscilloscope. Not included here is the instrument driver code.

A.1. Pulls Data Off of Scope and writes to .txt file.

```
%%This code uses the Tektronix DPO7104 oscilloscope as the data
acquisition
%%system for 2 channels.  Data is read from oscilloscope using Fast
Frame
%%Mode.
clc
clear all
close all
Run_Start_time = fix(clock) % Get Time for starting acquisition
dpo.InputBufferSize = 500000; %Buffer size
dpo = instrfind('Type', 'tcpip', 'RemoteHost', '169.254.235.51',
'RemotePort', 4000, 'Tag', ''); %Identifies Oscilloscope

if isempty(dpo)
    dpo = tcpip('169.254.235.51', 4000);  % IP Address of DPO7104
else
    fclose(dpo);
    dpo = dpo(1);
end

deviceObj = icdevice('tektronix_dpo7104_r1.mdd', dpo);  %Connecting to
scope using driver called tektronix_dpo_7104_r1.mdd

%Connect to the scope
connect(deviceObj);


nRecords = 10; %Number of FastFrame waveforms to capture for each time
the buffer dumps
minutes_run = .1667;  %  Sets how many minutes to acuire data for

%% Scope Settings
fprintf(dpo,'Select:Ch1 On') % Turns input for channel 1 on
fprintf(dpo,'Select:Ch2 On') % Turns input for channel 2 on
fprintf(dpo,'CH1:Position 0') % Aligns channel 1 at 0 V
fprintf(dpo,'CH2:Position 0') % Aligns channel 2 at 0 V
fprintf(dpo,'Horizontal:Position 10') % Places signals at designated
horizontal location (10)
```

```
fprintf(dpo,'ch1:Scale .1');  % Sets vertical scale for channel 1 (.1
V/div)
fprintf(dpo,'ch2:Scale .1');  % Sets vertical scale for channel 2 (.1
V/div)
% Trigger Setting
fprintf(dpo,'Trigger:Enhance Off');  % Toggles enhanced triggering mode
fprintf(dpo,'Trigger:A:Type Logic'); % Sets the A trigger mode as a
Logic
fprintf(dpo,'Trigger:A:Logic:Class Pattern');  % Sets the A trigger to
use the Pattern Mode
fprintf(dpo,'Trigger:A:Logic:Function And');% {AND|NANd|NOR|OR}  Sets
boolean logic to perform on triggered events
% AND specifies to trigger if all conditions are true.
% NANd specifies to trigger if any of the conditions are false.
% NOR specifies to trigger if all conditions are false.
% OR specifies to trigger if any of the conditions are true.
fprintf(dpo,'Trigger:A:Logic:Pattern:Input:CH1 High'); % {HIGH|LOW|X}
Sets importance level for channel 1 (High)
fprintf(dpo,'Trigger:A:Logic:Pattern:Input:CH2 High'); % Sets
importance level for channel 2 (High)
fprintf(dpo,'Trigger:A:Logic:Input:Format Binary'); % Sets data format
to Binary--Can also be 'Hexadecimal'
fprintf(dpo,'Trigger:A:Logic:Pattern:When False'); %
{TRUe|FALSe|LESSThan|MOREThan} Sets condition for meeting boolean logic
fprintf(dpo,'Trigger:A:Logic:Threshold:CH1 -10e-3'); % Sets trigger
level for Channel 1
fprintf(dpo,'Trigger:A:Logic:Threshold:CH2 -10e-3'); % Sets trigger
level for Channel 2
fprintf(dpo,'Trigger:A:Mode Normal'); % Sets trigger mode to Normal
(Other option is Auto -> forces trigger at timeout period)
fprintf(dpo,'ACQUIRE:MODE HiRes'); %Sets sampling mode to High
Resolution Mode

NumberOfPoints = str2num(query(dpo,'HORIZONTAL:MODE:RECORDLENGTH?')); %
Queries the record length (number of points per waveform)
SampleRate = str2num(query(dpo,'HORIZONTAL:MODE:SAMPLERATE?')); %
Queries the sample rate of the scope
TimeSpan = NumberOfPoints/SampleRate; % Returns seconds per point
timearray = linspace(0,TimeSpan,NumberOfPoints);  % Array of time
fprintf(dpo,['HORIZONTAL:MODE MANUAL'])
fprintf(dpo,'Data:Encdg Ribinary');  % Specifies signed integer-data
point representation with the most significant byte transferred first
(Try SIRBINARY)
fprintf(dpo,'WFMOutpre:BYT_NR 2'); % Sets the binary width of the data
fprintf(dpo,'DATA:START 75'); % Sets the beginning of the waveform data
to transfer
fprintf(dpo,'DATA:STOP 2500'); % Sets the end of the waveform data to
transfer
fprintf(dpo,'WFMOutpre:ENCDG BIN') % Sets incoming data to binary
format
fprintf(dpo,'WFMInpre:ENCDG BINARY') % Sets outgoing data to binary
format (Lose?)
fprintf(dpo,'WFMInpre:BN_FMT RI'); % Specifiies integer-data point
representaoitn (Lose?)
fprintf(dpo,'WFMInpre:BYT_NR 2');%Set 2 byte format = int16  (Lose?)
```

```
fprintf(dpo,'Horizontal:FastFrame:State 1'); %Turns on FastFrame Mode
fprintf(dpo,['Horizontal:FastFrame:Count ' num2str(nRecords)]); %Sets
the number of waveforms to take for each buffer dump
nFrames = str2num(query(dpo,'Horizontal:FastFrame:Count?')); %Returns
the number of frames

fprintf(dpo,'DATA:SOURCE Ch1'); % Sets data source to Channel 1
Ch1Ymult = str2num(query(dpo,'WFMOutpre:YMult?'));  % Returns vertical
scale factor
Ch1Yoff = str2num(query(dpo,'WFMOutpre:Yzero?'));  % Returns the offset
factor
fprintf(dpo,'DATA:SOURCE Ch2'); % Sets data source to Channel 2
Ch2Ymult = str2num(query(dpo,'WFMOutpre:YMult?')); % Returns vertical
scale factor
Ch2Yoff = str2num(query(dpo,'WFMOutpre:Yzero?'));  % Returns the offset
factor

fprintf(dpo,'DATA:SOURCE Ch1,CH2');  % Reinitiates data source to
specified channels
nSamples=5500000; % Sets the number of sammples to acquire (set really
large to allow for time acquisition instead of number of acquisitions)
seconds_run = minutes_run*60;  % Changes length of acuisition to
seconds
tstart = tic;  % Start acquistion timer
for i = 1:nSamples
    fprintf(dpo,'Acquire:StopAfter Sequence'); %Tells oscilloscope to
stop acquiring until Acquire:State On command is given again
    fprintf(dpo,'Acquire:State On') %Begins Acquisition
    fprintf(dpo,'*wai'); %Ensures scope does not perform next command
until the previous has been completed
    fprintf(dpo,'CURVE?');     % Call waveform (writes out in list mode
Channel 1 first then Channel2)
    Channel(:,1) = Ch1Ymult*binblockread(dpo, 'int16')+Ch1Yoff; % Reads
Channel 1 and scales it
    Channel(:,2) = Ch2Ymult*binblockread(dpo, 'int16')+Ch2Yoff; % Reads
Channel 2 and scales it
    fprintf(dpo,'*wai'); % Makes scope wait to finish tranfer before
executing next command
    fread(dpo,1); % Clears extra terminator from 'CURVE?' cmd
    fprintf(dpo,'*wai'); % Makes scope wait to finish tranfer before
executing next command

dlmwrite('Ch_2Dec_10sec_highres_40ns_500.txt',Channel,'precision',16,'-
append') %Writes data to file appending after each acquisition
    if toc(tstart) > seconds_run  % Exit "for" loop when timing
criteria has been met
        break
    end
end
Sample_Collection_Time = toc(tstart) % Gives how long acquisition took

%Disconnect and close Oscilloscope communication portal
disconnect(deviceObj);
```

```
delete(deviceObj);
delete(dpo);
Run_Stop_time = fix(clock) % Gives current time
```

A.2. Post-Processing Code for Oscilloscope Data.

```
close all
clear all
clc
%%This code processes the data taken from the oscilloscope from two
%%channels
Run_Start_time = fix(clock)
clear segarray;
block_size = 126; %Set to number of points transfered per waveform
(1+Data:stop - Data:start)
file_id = fopen('Ch_23Nov_60min.txt');
%Pre-allocate Arrays
count = 0;
compton_events_energy = [];
pair_production_events_energy = [];
Channel1pair = [];
Channel2pair = [];
Channel1comp = [];
Channel2comp = [];
%Processes oscilloscope data by sorting out pair-production and Compton
%events
while ~feof(file_id)
    count = count+1;
    segarray = textscan(file_id,'%n%n',block_size,'delimiter',',');

[compton_events_energy_new,pair_production_events_energy_new,coincident
_pulse_peak, Channel1pair_new, Channel2pair_new, Channel1comp_new,
Channel2comp_new]=process_data_func(segarray);
    compton_events_energy = [compton_events_energy;
compton_events_energy_new];
    pair_production_events_energy = [pair_production_events_energy;
pair_production_events_energy_new];
    Channel1pair = [Channel1pair,Channel1pair_new]; %Channel 1 pair-
production events
    Channel2pair = [Channel2pair,Channel2pair_new]; %Channel 2 pair-
production events
    Channel1comp = [Channel1comp,Channel1comp_new]; %Channel 3 Compton
events
    Channel2comp = [Channel2comp,Channel2comp_new]; %Channel 4 Compton
events
end

fclose(file_id); %Close file

%Create historgram information for pair-production and Compton events
[all_coincidences(1,:),xoutpair]=hist(pair_production_events_energy(:,5
),1024);
[all_coincidences(2,:),xoutcomp]=hist(compton_events_energy(:,5),1024);
```

```
[all_coincidences(3,:),xoutall] = hist([compton_events_energy(:,5);
pair_production_events_energy(:,5)],1024);
[pairelem,paircol] = size(Channel1pair);
[compelem,compcol] = size(Channel1comp);

%Information from oscilloscope needed to scale data correctly
NumberOfPoints = 1000;
SampleRate = 1.0000e+009;
TimeSpan = NumberOfPoints/SampleRate;
time = linspace(0,TimeSpan,NumberOfPoints);
timechange = time(75:200);

%Finds lowerbound and upperbound of 511 keV peak
det_res_511 = 0.05;
lowerbound_511 = 511-511*det_res_511;
upperbound_511 = 511+511*det_res_511;
pair_511 = [];
%Finds pair-produciton events that are within 3% of the 1173 keV and
1332
%keV peaks then stores those events in arrays for the respective
detector
for k = 1:paircol
    if ((pair_production_events_energy(k,5) > 1137 &
pair_production_events_energy(k,5) < 1208) |
(pair_production_events_energy(k,5) > 1292 &
pair_production_events_energy(k,5) < 1370))
        if (pair_production_events_energy(k,2)>lowerbound_511 &&
pair_production_events_energy(k,2)<upperbound_511)
            pair_511= [pair_511,Channel1pair(:,k)];
        end
    full_en_pair1(:,k) = Channel1pair(:,k);
    full_en_pair2(:,k) = Channel2pair(:,k);
    end
end
compton_511 = [];
%Finds Compton events that are within 3% of the 1173 keV and 1332
%keV peaks then stores those events in arrays for the respective
detector
for k = 1:compcol
    if ((compton_events_energy(k,5) > 1137 & compton_events_energy(k,5)
< 1208) | (compton_events_energy(k,5) > 1292 &
compton_events_energy(k,5) < 1370))
        if (compton_events_energy(k,1)>lowerbound_511 &&
compton_events_energy(k,1)<upperbound_511)
            compton_511 = [compton_511,Channel2comp(:,k)];
        end
    full_en_comp1(:,k) = Channel1comp(:,k);
    full_en_comp2(:,k) = Channel2comp(:,k);
    end
end

%Sums all of the waveforms with energy of 511 keV from the full energy
peaks that are either
%pair-produciton or Compton events
[~,pair_size] = size(pair_511);
```

```
[~,compton_size] = size(compton_511);
sum_pair = sum(pair_511,2);
sum_comp = sum(compton_511,2);
if pair_size<compton_size
    small_length = pair_size;
else
    small_length = compton_size;
end
sum_comp_small = sum(compton_511(:,1:small_length),2);

%Plots normalized pair-produciton 511 keV events
figure (4)
plot(timechange,-sum_pair/min(sum_pair),timechange,-
sum_comp_small/min(sum_comp_small))
title('Normalized Pair-Production Events vs. Compton Events (1 hr.
Acquisition of Co-60 at 10cm)')
xlim([.8E-7 .2E-6])
ylim([-1 .1])
xlabel('Time (ns)')
ylabel('Voaltage')
set(gca,'XTick',.8E-7:10E-9:.2E-6)
set(gca,'XTickLabel',{'80','90','100','110','120','130','140','150','16
0','170','180','190','200'})
legend('Pair-Production Events','Compton Events','Location','East')

%Plots normalized Compton 511 keV events
figure(5)
plot(timechange,-sum_comp_small/min(sum_comp_small),'xb')
title('Normalized Compton Events (1 hr. Acquisition of Co-60 at 10cm)')
xlim([.8E-7 .2E-6])
ylim([-1 .1])
xlabel('Time (ns)')
ylabel('Voaltage')
set(gca,'XTick',.8E-7:10E-9:.2E-6)
set(gca,'XTickLabel',{'80','90','100','110','120','130','140','150','16
0','170','180','190','200'})

%Plots normalized pair-produciton and Compton 511 keV events
figure(6)
plot(timechange,-sum_pair/min(sum_pair),'.r')
title('Normalized Pair-Production Events (1 hr. Acquisition of Co-60 at
10cm)')
xlim([.8E-7 .2E-6])
ylim([-1 .1])
xlabel('Time (ns)')
ylabel('Voaltage')
set(gca,'XTick',.8E-7:10E-9:.2E-6)
set(gca,'XTickLabel',{'80','90','100','110','120','130','140','150','16
0','170','180','190','200'})

%Plot pair-production event spectrum
% figure(2)
% plot(xoutpair,all_coincidences(1,:))
% xlabel('Energy (keV)')
```

```
% ylabel('Number of Counts')
% title('Pair Production Events of Co60 over 1 hour at 10cm (2ns
Coincidence Window)')
% xlim([0 2000])

%Plot Compton event spectrum
% figure(3)
% plot(xoutcomp,all_coincidences(2,:))
% xlabel('Energy (keV)')
% ylabel('Number of Counts')
% title('Compton Effect Events of Co60 over 1 hour at 10cm (2ns
Coincidence Window)')
% xlim([0 2000])

Run_Stop_time = fix(clock)
```

A.3. Process Data Function called by Oscilloscope Post-Processing Program

```
%This function takes the data from the Oscilloscope_Spectroscopy.m file
and
%Filters it out to contain the desired data
function [compton_events_energy,pair_production_events_energy,
coincident_pulse_peak, channel1pair_out, channel2pair_out,
channel1comp_out, channel2comp_out] = process_data(segarray)
Channel1 = cell2mat(segarray(:,1));
Channel1 = reshape(Channel1,126,[]);

Channel2 = cell2mat(segarray(:,2));
Channel2 = reshape(Channel2,126,[]);

clear segarray
NumberOfPoints = 126;  % Number of points in waveform
SampleRate = 1.0000e+009; % Sample rate of scope
TimeSpan = NumberOfPoints/SampleRate; % Returns seconds per point
timearray = linspace(0,TimeSpan,NumberOfPoints); % Array of time
detnum=2; % Number of detectors
coincidence_window = 25e-9; % Width of coincidence window in seconds
trigger_level = 10e-3; % Trigger level in V

[num_points,num_wfms] = size(Channel1); % Determines how many waveforms
were recorded
%Use Savitsky-Golay filter to smooth the data
Channel1filt = sgolayfilt(Channel1,3,7);
Channel2filt = sgolayfilt(Channel2,3,7);
%Find the minimum(peak) of the filtered data
[pulse_peak_ch1,min_index_ch1] = min(Channel1filt);  % Find the pulse
height of each pulse and the index of that pulse height for channel 1
[pulse_peak_ch2,min_index_ch2] = min(Channel2filt);  % Find the pulse
height of each pulse and the index of that pulse height for channel 2
%Apply energy calibration to the trigger level of the data
```

```
trigger_level_ch1 = -7448*trigger_level^3 + 2859*trigger_level^2 +
5104.4*trigger_level - 0.1771;  % Energy calibration for channel 1 for
the trigger level
trigger_level_ch2 = -26841*trigger_level^3 + 8977.7*trigger_level^2 +
4930.1*trigger_level - 1.8421;  % Energy calibration for channel 2 for
the trigger level
%Pre-Allocate arrays
midrise_time_ch1 = zeros(1,num_wfms);
midrise_time_ch2 = zeros(1,num_wfms);
pulse_order_time_comp = zeros(num_wfms,2);
pulse_order_index = zeros(num_wfms,2);
energy_order_index = zeros(num_wfms,2);
% Calls function "find_midrise_time" and gives the timming for each
% triggered event(time is off of the midpoint of the rise of the pulse)
for i = 1:num_wfms

[midrise_time_ch1(i)]=find_midrise_time(Channel1filt(:,i),timearray,tri
gger_level);

[midrise_time_ch2(i)]=find_midrise_time(Channel2filt(:,i),timearray,tri
gger_level);
end

pulse_peak_ch1 = -pulse_peak_ch1;  % Changes pulse values from negative
to positive for channel 1
en_cal_ch1 = -7448.*pulse_peak_ch1.^3 + 2859.*pulse_peak_ch1.^2 +
5104.4.*pulse_peak_ch1 - 0.1771; % Energy calibration for pulse heights
for channel 1
pulse_peak_ch2 = -pulse_peak_ch2;  % Changes pulse values from negative
to positive for channel 2
en_cal_ch2 = -26841.*pulse_peak_ch2.^3 + 8977.7.*pulse_peak_ch2.^2 +
4930.1.*pulse_peak_ch2 - 1.8421;  % Energy calibration for pulse
heights for channel 2
%Filter out data that is below the trigger level
for j = 1:num_wfms
        if en_cal_ch1(j) < trigger_level_ch1  % If waveform peak is
below trigger level then call it a NaN
            en_cal_ch1(j) = NaN;  % If waveform peak is below trigger
level then call it a NaN
            midrise_time_ch1(j) = NaN;  % If waveform peak is below
trigger level then call it a NaN
        end
        if en_cal_ch2(j) < trigger_level_ch2  % If waveform peak is
below trigger level then call it a NaN
            en_cal_ch2(j) = NaN;  % If waveform peak is below trigger
level then call it a NaN
            midrise_time_ch2(j) = NaN; % If waveform peak is below
trigger level then call it a NaN
        end
        %Sorts the pulses according to time
        pulse_order_time_comp(j,1) = midrise_time_ch1(j);  % Duplicate
timing values to place in array to sort pulses
        pulse_order_time_comp(j,2) = midrise_time_ch2(j);  % Duplicate
timing values to place in array to sort pulses
```

```
        [pulse_order_time_comp(j,:),pulse_order_index(j,:)] =
sort(pulse_order_time_comp(j,:),'ascend');  % Sorts pulse orders by
time for each triggerd event
        energy_order_index(j,:) = pulse_order_index(j,:);  % Duplicates
index of pulse orders
        for k = 1:detnum
            if energy_order_index(j,k) == 1
                energy_order_index(j,k) = en_cal_ch1(j);  % Writes
energy of event into index location for Channel 1
            elseif energy_order_index(j,k) == 2
                energy_order_index(j,k) = en_cal_ch2(j);  % Writes
energy of event into index location for Channel 2
            end
        end

         %  Determines if pulses are within coincidence window and then
         %  writes them to a coincidence matrix or sets them to NaN if
not
         %  coincident
        if abs(pulse_order_time_comp(j,1) -
pulse_order_time_comp(j,detnum))<= coincidence_window
            coincident_pulse_times(j,1:detnum) =
pulse_order_time_comp(j,1:detnum);
            coincident_pulse_peak(j,1:detnum) =
energy_order_index(j,1:detnum);
        else coincident_pulse_times(j,1:2) = NaN;
            coincident_pulse_peak(j,1:2) = NaN;
        end
end
 %If no waveforms are deemed either pair-production events or Compton
 %events then write out blank arrays
 if num_wfms == 0
     coincident_pulse_peak = [];
     compton_events_energy =[];
     pair_production_events_energy = [];
     channel1pair_out = [];
     channel2pair_out = [];
     channel1comp_out = [];
     channel2comp_out = [];
     return
 end
 %Sort out which events are Pair-production and which are Compton
[coincident_energy_row,coincident_energy_col] =
find(coincident_pulse_peak(:,1)>0);
coincident_pulse_peak = coincident_pulse_peak(coincident_energy_row,:);
coincident_pulse_peak(:,5) = sum(coincident_pulse_peak,2);
coincident_pulse_times =
coincident_pulse_times(coincident_energy_row,:);
[num_elements_times,num_columns_times] = size(coincident_pulse_times);
% Finds the size of the coincidence time matrix
lowerbound_511 = 511-511*.05;
upperbound_511 = 511+511*.05;
[pair_prod_row,pair_prod_column] =
find(coincident_pulse_peak(:,2)>lowerbound_511 &
coincident_pulse_peak(:,2)<upperbound_511);
```

```
pair_production_events_energy = coincident_pulse_peak(pair_prod_row,:);
pair_production_events_time = coincident_pulse_times(pair_prod_row,:);
x=0;
%If events are pair-production, add 1022 to the non 511 keV event
if (pair_production_events_energy(:,1)<lowerbound_511 |
pair_production_events_energy(:,1)>upperbound_511)
    pair_production_events_energy(:,5) =
pair_production_events_energy(:,1) + 1022;
    x=1;
elseif (pair_production_events_energy(:,2)<lowerbound_511 |
pair_production_events_energy(:,2)>upperbound_511)
    pair_production_events_energy(:,5) =
pair_production_events_energy(:,2) + 1022;
    x=1;
else
    pair_production_events_energy(:,5) = 1022;
end
if pair_prod_row==1
    channel1pair_out = Channel1;
    channel2pair_out = Channel2;
else
    channel1pair_out = [];
    channel2pair_out = [];
end
%Determine if events are Compton events and sum the energies
[compton_row,compton_column] =
find(coincident_pulse_peak(:,2)<=lowerbound_511 |
coincident_pulse_peak(:,2)>=upperbound_511);
compton_events_energy = coincident_pulse_peak(compton_row,:);
compton_events_time = coincident_pulse_times(compton_row,:);
compton_events_energy(:,5) =
compton_events_energy(:,1)+compton_events_energy(:,2)+compton_events_en
ergy(:,3)+compton_events_energy(:,4);
compton_events_energy(:,6) = acosd(1-
((compton_events_energy(:,5)./compton_events_energy(:,2))-1)./511);
compton_events_order = pulse_order_index(compton_row,:);
compton_events_order(:,5) = compton_events_energy(:,6);
if compton_row==1
    channel1comp_out = Channel1;
    channel2comp_out = Channel2;
else
    channel1comp_out = [];
    channel2comp_out = [];
end
end
```

A.3. Find Midrise Time Function called by Process-Data Function

```
%%This code finds the time of the pulse by looking at the middle of the
%%rise of the pulse
function [halfpeak_time] =
find_midrise_time(Channel,timearray,trigger_level)
```

```
    if min(Channel) <= -trigger_level %Make sure the event is a valid
event and is below trigger level
        percent50 = .5*min(Channel); %Find 50 percent of the peak
height
        midrise = percent50;
        indexabove = find(Channel<midrise,1,'first'); %Look for the
first point that is greater than the middle of the rise
        if indexabove > 1
            indexbelow = indexabove - 1; %If the middle point is found
to be greater than 1 pick it
        else
            indexbelow = 1; %If the middle of the rise is the first
point the index is the first
        end
        %Find which timing point is closest to the middle of the rise
and
        %pick it to be the time
        abovediff = abs(Channel(indexabove)-midrise);
        belowdiff = abs(Channel(indexbelow)-midrise);
        if abovediff < belowdiff
            halfpeak_time = timearray(indexabove);
        else
            halfpeak_time = timearray(indexbelow);
        end
    else
        halfpeak_time = NaN;
    end
end
```

Appendix B. Post-Processing Code for XIA Data Acquisition System

The following files are MATLAB scripts that post process the data acquired by the XIA DAQ system. The data is sorted into pair-production and Compton events. The code allows spectroscopy and imaging to be performed.

B.1. Takes .dat file from XIA software and transforms it into file more easily read by MATLAB

```
close all
clear all
clc
%%This code takes a .dat file generated from a data acquisition using
the
%%XIA CAMAC electronics that has had no modifications done to it and
%%Transforms it into a file that can be used in MATLAB

Run_Start_time = fix(clock)
clear segarray;
fclose('all') %Closes any files that may still be open.
delete('Module_1_data.txt'); %Deletes previously processed files

file_id = fopen('Co60_60cm_180deg_1hr0001.dat'); %Name of file to
process
%Reads in Header information at top of file.  Makes manually deleting
file
%headers unnecessary.
block_size = 3;
headertext_mod1 = textscan(file_id,'%s',block_size,'delimiter','\n');
block_size = 1;
headertext_mod1_2 =
textscan(file_id,'%s%n',block_size,'delimiter','\t');
Run_start_mod1 = headertext_mod1_2(2);
Run_start_mod1 = cell2mat(Run_start_mod1);
block_size = 3;
headertext_mod1_3 = textscan(file_id,'%s',block_size,'delimiter','\n');

%Reads in data and stores in text file named Module_1_data.txt.  Each
read
%appends to the file
block_size = 50000;
count1 = 0;
while ~feof(file_id)
    count1 = count1+1;
   segarraymod1 =
textscan(file_id,'%n%n%n%n%n%n',block_size,'delimiter','\t');
   [mod_1_data_new]=process_data_xia(segarraymod1);
   dlmwrite('Module_1_data.txt',mod_1_data_new,'precision',12,'-
append')
end
```

```
fclose(file_id); %Close the file

Run_Stop_time = fix(clock)
```

B.2. Post-Processing code for the XIA electronics that performs spectroscopy on the data.

```
close all
clear all
clc
%%This code takes the file that was modified by XIA_strip.m and
performs
%%spectroscopy on it.  It also prepares a .mat file to be used in the
%%Backprojection_xia.m code.

Run_Start_time = fix(clock)

file_id_mod = fopen('Module_1_data.txt'); %Open the file

%Initialize arrays
energy_mat = [];
pos_mat = [];
compton_events_en = [];
pair_events_end = [];
ch1 = [];
ch2 = [];
ch3 = [];
ch4 = [];
count_data = 0;
block_size = 4; %Number of channels of data

%Reads textfile and sends to function process_data_xia then stores
%processed data in arrays by appending to the previous.
while ~feof(file_id_mod)
   count_data = count_data+1;
   seg_mod =
textscan(file_id_mod,'%n%n%n%n',block_size,'delimiter',',');
   [energy_mat_new,pos_mat_new,ch1_new,ch2_new,ch3_new,ch4_new] =
process_data_xia(seg_mod);
    ch1 = [ch1;ch1_new];
    ch2 = [ch2;ch2_new];
    ch3 = [ch3;ch3_new];
    ch4 = [ch4;ch4_new];
    energy_mat = [energy_mat;energy_mat_new]; %Energies deposited
    pos_mat = [pos_mat;pos_mat_new]; %Detector number that gamma
deposited in
end

fclose(file_id_mod); %Close the file
delete('compton_events.mat'); %Delete compton_events.mat from previous
runs
```

```
res = 0.05; %Resolution of the detectors at 511 keV
lowerbound_511 = 511 - 511*res; %Lower bound of 511 keV peak
upperbound_511 = 511 + 511*res; %Upper bound of 511 keV peak
[energy_matelem,~] = size(energy_mat); %Determine how many events will
need to be sorted

%Initialize Arrays
pair_events_en = [];
pair_events_pos = [];
compton_events_en = [];
compton_events_pos = [];

%Sorts events into Pair-Production or Compton Event arrays.  If either
%event one, two or three are within the 511 keV window then the event
is a
%Pair-production event.  Otherwise, it is a Compton event.
for i = 1:energy_matelem
    if (((energy_mat(i,1) > lowerbound_511) & (energy_mat(i,1) <
upperbound_511)) | ((energy_mat(i,2) > lowerbound_511) &
(energy_mat(i,2) < upperbound_511)) | ((energy_mat(i,3) >
lowerbound_511) & (energy_mat(i,3) < upperbound_511)))
        pair_events_en = [pair_events_en;energy_mat(i,:)]; %Pair-
Production Energy Array
        pair_events_pos = [pair_events_pos;pos_mat(i,:)]; %Pair-
Production Detector Number Array
    else
        compton_events_en = [compton_events_en;energy_mat(i,:)];
%Compton Energy Array
        compton_events_pos = [compton_events_pos;pos_mat(i,:)];
%Compton Detector Number Array
    end
end
%Delete unecessary arrays to save space
clear energy_mat
clear pos_mat
%Determine how many compton and pair-production events there are
[compton_events_en_elem,~] = size(compton_events_en);
[pair_events_en_elem,~] = size(pair_events_en);

if compton_events_en_elem | pair_events_en_elem > 0 %Check to make sure
event deposited positive amount of energy and is therefore real
    %Inner loop to look at Compton Evnets
    if compton_events_en_elem >0
        %Condense array to rid holes left from filtering out
        %Pair-production events
        [comptoneventselem,~] = find(compton_events_en(:,1));
        compton_events_en = compton_events_en(comptoneventselem,:);
        compton_events_pos = compton_events_pos(comptoneventselem,:);
        %Sum energies of first two events to come up with the initial
        %energy of the incident gamma ray
        compton_events_en(:,4) =
compton_events_en(:,1)+compton_events_en(:,2);
        %Compute the angles of scatter for both possible orderings of
the
        %events
```

```
        compton_events_en(:,5) = acosd(1+511./compton_events_en(:,4) -
511./compton_events_en(:,1));
        compton_events_en(:,6) = acosd(1+511./compton_events_en(:,4) -
511./compton_events_en(:,2));
        %Rid the array of non-physical scatters and keep only the
"real"
        %numbers.  If both possible scattering angles are complex then
the
        %entire event is thrown out.
        for m=1:compton_events_en_elem
            if isreal(compton_events_en(m,5))
                compton_events_en(m,1:4) =
real(compton_events_en(m,1:4));
                compton_events_en(m,5) = real(compton_events_en(m,5));
            else
                compton_events_en(m,5) = NaN;
            end
            if isreal(compton_events_en(m,6))
                compton_events_en(m,1:4) =
real(compton_events_en(m,1:4));
                compton_events_en(m,6) = real(compton_events_en(m,6));
            else
                compton_events_en(m,6) = NaN;
            end
            if  isnan(compton_events_en(m,5:6))
                compton_events_en(m,:) = NaN;
            end
        end
        %Resize array to only have "real" events and save file to
perform
        %backprojection in another M-file
        real_vec = isnan(compton_events_en(:,1));
        [real_compton_elem,~] = find(real_vec==0);
        real_compton = compton_events_en(real_compton_elem,:);
        savefile1 = 'compton_events.mat';
        save(savefile1,'real_compton','compton_events_pos')
        clear comptoneventselem
        clear comptoneventscol
        %Plot Compton Spectrum and Print to Command window the number
of
        %counts within 3% of the peaks of interest
        nbins = 1024;
        vec = linspace(0,2000,nbins);
        [compty,comptx] = hist(real_compton(:,4),vec);
        compty(1024) = 0;
        figure(9)
        area(comptx,compty)
        title('Compton Events from Co-60 at 30cm and 90\circ and a
12hr. Acquistion (Phi=0\circ)')
        xlabel('Energy(keV)')
        ylabel('Number of Counts')
        xlim([0 2000])
        Num_1173 = sum(compty(583:616))
        Num_1332 = sum(compty(662:703))
        Num_1468 = sum(compty(713:764))
```

```
    end
    %Inner loop to look at Compton Evnets
    if pair_events_en_elem > 0
        %Condense array to rid holes left from filtering out
        %Compton events
        [paireventselem,~] = find(pair_events_en(:,1));
        pair_events_en = pair_events_en(paireventselem,:);
        pair_events_pos = pair_events_pos(paireventselem,:);
        clear paireventselem
        clear paireventscol
        pair_events_en(:,4) = 0;
        [pair_elem,~] = size(pair_events_en);
        %Finds the gamma that is not the 511 keV gamma and adds 1022
keV to
        %it
        for j = 1:pair_elem
            tempa = [0 0 0 0];
            tempa = sort(pair_events_en(j,:));
            if (((tempa(1)<lowerbound_511) | (tempa(1)>upperbound_511))
& tempa(1) ~=0)
                pair_events_en(j,4) = tempa(1)+1022;
            elseif (((tempa(2)<lowerbound_511) |
(tempa(2)>upperbound_511)) & tempa(2) ~=0)
                pair_events_en(j,4) = tempa(2)+1022;
            elseif (((tempa(3)<lowerbound_511) |
(tempa(3)>upperbound_511)) & tempa(3) ~=0)
                pair_events_en(j,4) = tempa(3)+1022;
            elseif (((tempa(4)<lowerbound_511) |
(tempa(4)>upperbound_511)) & tempa(4) ~=0)
                pair_events_en(j,4) = tempa(4)+1022;
            else
                tempaindex = find(tempa>0,1,'first');
                pair_events_en(j,4) = tempa(tempaindex)+1022;
            end
        end
        %Plot Pair-Production Spectrum and Print to Command window the
number of
        %counts within 3% of the peaks of interest
        nbins = 1024;
        vec = linspace(0,2000,nbins);
        [pairy,pairx] = hist(pair_events_en(:,4),vec);
        pairy(1024) = 0;
        figure(10)
        area(pairx,pairy)
        title('Pair-Production Events from Co-60 at 30cm and 90\circ
and a 12hr. Acquistion (Phi=0\circ)')
        xlabel('Energy(keV)')
        ylabel('Number of Counts')
        xlim([3 2000])
        Num_1173 = sum(pairy(583:616))
        Num_1332 = sum(pairy(662:703))
        Num_1468 = sum(pairy(713:764))
        Num_1684 = sum(pairy(836:889))
        Num_1843 = sum(pairy(915:972))
    end
```

```
end

Run_Stop_time = fix(clock)
```

B.3. Post-processing function that is called by the post-processing code for the XIA electronics

```
%%This code process the data by applying energy calibrations and
sorting
%%the events
function [energy_mat, pos_mat,ch1,ch2,ch3,ch4] =
process_data_xia(segmod1)

event1 = cell2mat(segmod1(:,:));
%Gives detector number to columns
event1(1,2) = 1;
event1(2,2) = 2;
event1(3,2) = 3;
event1(4,2) = 4;

%Performs energy calibration on event based on detector
if event1(1,3)>0
    ch1a = 7.436541e-9;ch1b = 0.032221;ch1c = -0.0826102;
    event1(1,3) = ch1a*event1(1,3)^2+ch1b*event1(1,3) + ch1c;
end
if event1(2,3)>0
    ch2a = 9.848932e-9;ch2b = 0.032487;ch2c = 1.659473;
    event1(2,3) = ch2a*event1(2,3)^2+ch2b*event1(2,3) + ch2c;
end
if event1(3,3)>0
    ch3a = 7.677146e-9;ch3b = .033423;ch3c = 2.863917;
    event1(3,3) = ch3a*event1(3,3)^2+ch3b*event1(3,3) + ch3c;
end

if event1(4,3)>0
    ch4a = 1.788921e-9;ch4b = 0.032826;ch4c = 0.440832;
    event1(4,3) = ch4a*event1(4,3)^2+ch4b*event1(4,3) + ch4c;
end
%Makes sure energy calibration did not give an energy of less than one.
If
%it did this sets that channel blank.
if event1(1,3)>0
    ch1 = event1(1,3);
else
    ch1 = [];
end

if event1(2,3)>0
    ch2 = event1(2,3);
else
    ch2 = [];
end
```

```
if event1(3,3)>0
    ch3 = event1(3,3);
else
    ch3 = [];
end

if event1(4,3)>0
    ch4 = event1(4,3);
else
    ch4 = [];
end
if event1(1,3)<0
    event1(1,3)=0;
end
if event1(2,3)<0
    event1(2,3)=0;
end
if event1(3,3)<0
    event1(3,3)=0;
end
if event1(4,3)<0
    event1(4,3)=0;
end

%Finds the non-zero events and forms them into an array to be sent out
of
%funciton
index_event1 = find(event1(:,3)>1);
nonz_events1 = event1(index_event1,:);
nonz_events = [nonz_events1];
[num_nonz_eventselem,~] = size(nonz_events);
if num_nonz_eventselem==2
    event1en = nonz_events(1,:);
    event2en = nonz_events(2,:);
    event3en = [0,0,0,0];
elseif num_nonz_eventselem==3
    event1en = nonz_events(1,:);
    event2en = nonz_events(2,:);
    event3en = nonz_events(3,:);
else
    event1en = [];
    event2en = [];
    event3en = [];
end
[event_elem,event_col] = size(event1en);
if event_elem>0
    energy_mat = [event1en(1,3),event2en(1,3),event3en(1,3)];
    pos_mat = [event1en(1,2),event2en(1,2),event3en(1,2)];
else
    energy_mat = [];
    pos_mat = [];
end
end
```

B.4. Backprojection code that produces image from XIA electronics data.

```
close all
clear all
clc

%% This code takes the .mat file that was generated by the
%% XIA_spectroscopy.m code and performs backprojection on it.
%Load and Rename files
load compton_events.mat
energy_mat = real_compton;
det_num = compton_events_pos;
clear real_compton
clear compton_events_pos
load 1min_background_xy.mat
background_image = Image;
clear Image

%Rename columns of array for easier tracking
Einc = energy_mat(:,4);
[energy_matelem,energy_matcol] = size(energy_mat(:,4));
ThetaScatter1 = energy_mat(:,5);
ThetaScatter2 = energy_mat(:,6);
Zeff = 45.5673; %Z effective of LaBr3
PScatter1 = zeros(energy_matelem,1);
PScatter2 = zeros(energy_matelem,1);
%Finds the probability of scattering for both orderings of angles
for k=1:energy_matelem
    Pscat180 =
KleinNishina(Einc(k,1),Zeff)./sum(KleinNishina(Einc(k,1),Zeff));
    if isnan(ThetaScatter1(k,1))
        PScatter1(k,1) = 0;
    else
        PScatter1(k,1) = Pscat180(ceil(ThetaScatter1(k,1)));
    end
    if isnan(ThetaScatter2(k,1))
        PScatter2(k,1) = 0;
    else
        PScatter2(k,1) = Pscat180(ceil(ThetaScatter2(k,1)));
    end
end
%Coordinates of Detectors
% %X-Y Plane
x1 = [1.4 1.4 0];
x2 = [-1.4 1.4 0];
x3 = [-1.4 -1.4 0];
x4 = [1.4 -1.4 0];

x = [x1;x2;x3;x4];
nDet =4; %Number of Detectors
tolerance = 0.005; %Tolerance used to determine rim of cone location on
world sphere

R_S = 10; %Radius of the sphere
```

```
Phi_S = linspace(0,180,200); %Number of points that makes up the Phi
grid of the world sphere
nPhi = length(Phi_S);
Theta_S = linspace(0,360,200); %Number of points that makes up the
Theta grid of the world sphere
nTheta = length(Theta_S);
%% Pre-allocate X, Y and Z arrays
x_s = zeros(nPhi,nTheta);
y_s = x_s;
z_s = x_s;
k = 0;
%Convert Theta and Phi into X,Y,Z coordinates
for i=1:nPhi
    for j=1:nTheta
        k=k+1;
        x_s(k) = sind(Phi_S(i))*cosd(Theta_S(j));
        y_s(k) = sind(Phi_S(i))*sind(Theta_S(j));
        z_s(k) = cosd(Phi_S(i));
    end
end
%Pre-Allocate Image array
Image = zeros(nPhi,nTheta);
most_prob = find(((PScatter1(:)>0.005|PScatter2(:)>0.005)));
 ThetaScatter1=ThetaScatter1(most_prob,1);
 ThetaScatter2=ThetaScatter2(most_prob,1);
 PScatter1 = PScatter1(most_prob,1);
 PScatter2 = PScatter2(most_prob,1);
 det_num = det_num(most_prob,:);
 energy_mat = energy_mat(most_prob,:);
[energy_matelem,energy_matcol] = size(energy_mat(:,4));
%Determine rim of cone and backproject it onto world sphere
% for i=1:ceil(energy_matelem)
for i =1:1
    Pscatter = 0;
    %Determine most probable scatter and get values as shown
    if PScatter1(i,1)>PScatter2(i,1)
        ThetaScatter = ThetaScatter1(i,1);
        xinc = x(det_num(i,1),:);
        xscat = x(det_num(i,2),:);
        PScatter = PScatter1(i,1);
    else
        ThetaScatter = ThetaScatter2(i,1);
        xinc = x(det_num(i,2),:);
        xscat = x(det_num(i,1),:);
        PScatter = PScatter2(i,1);
    end
    %Determine Characteristics of Backprojected Cone
    CosThetaScatter = cosd(ThetaScatter);
    Vn = xinc-xscat;
    Vn_hat = Vn/norm(Vn);
    loop = 0;
    %Backproject Rim of cone onto world sphere and add probability to
the
    %rim
    for k=1:nPhi % Phi Pixels
```

```
        for L = 1:nTheta %Theta Pixels
            loop=loop+1;
            Vs = [x_s(loop) y_s(loop) z_s(loop)];
            Vs_hat = Vs/norm(Vs);
            CosThetaNS = dot(Vn_hat,Vs_hat);
            if(and(CosThetaNS <= CosThetaScatter + tolerance,...
                    CosThetaNS > CosThetaScatter - tolerance))
                Image(k,L)=Image(k,L)+PScatter;
            end
        end
    end
end
image_less_background = Image-background_image;

% createfigure1(image_less_background);
figure;imagesc(Theta_S,Phi_S,image_less_background);

%Find array location of max values of Theta and Phi
[~,thetaMax] = max(max(Image))
[~,phiMax] = max(max(Image'))
%Print to screen the maximum Theta and Phi location using spherical
%coordinates in degrees
Theta = Theta_S(thetaMax)
Phi = Phi_S(phiMax)
%Convert max values of theta and Phi to Cartesian Coordinates
Z = R_S*cosd(Phi_S(phiMax))
X = R_S*sind(Phi_S(phiMax))*cosd(Theta_S(thetaMax))
Y = R_S*sind(Phi_S(phiMax))*sind(Theta_S(thetaMax))
Max_Probability = max(max(Image))
[~,Theta_less_back] = max(max(image_less_background))
[~,Phi_less_back] = max(max(image_less_background'))
Max_Probability_less_back = max(max(image_less_background))
Theta = Theta_S(Theta_less_back)
Phi = Phi_S(Phi_less_back)
```

B.5. Klein-Nishina Function Called by the backprojection code for the XIA electronics

```
%This function computes the differential cross-section for a given
energy
%and material Z from the Klein-Nishina relatoin.
function DiffXSec = KleinNishina(E,Z)
    re = 2.81794e-15; %classical e- radius
    theta = linspace(0,pi,180)';
    a = E/511;
    Term1 = (1./(1+a.*(1-cos(theta)))).^2;  % wrong in Evans Thesis?
No looks like Krane is wrong
    Term2 = (1+cos(theta).^2)./2;
    Term3 = 1+(((a.^2).*(1-
cos(theta)).^2)./((1+cos(theta).^2).*(1+a.*(1-cos(theta)))));
    DiffXSec = Z*re.^2*Term1.*Term2.*Term3;
end
```

Appendix C. Geant4 Simulation and Post-Processing in ROOT

The following files are Geant4 and ROOT scripts that create the simulation of the pair-production spectrum using the 4 element detector in this project. The files will be presented as they are grouped in a file structure. This structure with folder names is:

-include
- --thesisDetectorConstruction.hh
- --ThesisEventAction.hh
- --ThesisHit.hh
- --thesisPhysicsList.hh
- --thesisPrimaryGenerator.hh
- --ThesisTrackerSD.hh

-src
- --thesisDetectorConstruction.cc
- --ThesisEventAction.cc
- --ThesisEventParameters.icc
- --ThesisHit.cc
- --thesisPhysicsList.cc
- --thesisPrimaryGeneratorAction.cc
- --ThesisTrackerSD.cc

-GNUmakefile.mk
-thesis.cc
-vis.mac
-MakeRootTree_hit.C
-ProcessAllPairCompton.C

C.1. thesisDetectorConstruction.hh

```
// thesisDetectorConstruction.hh
// Created by W.L. Harrell

#ifndef thesisDetectorConstruction_H
#define thesisDetectorConstruction_H 1

class G4LogicalVolume;
class G4VPhysicalVolume;

#include "G4VUserDetectorConstruction.hh"

class thesisDetectorConstruction : public G4VUserDetectorConstruction
{
  public:

    thesisDetectorConstruction();
    ~thesisDetectorConstruction();
```

```
    G4VPhysicalVolume* Construct();

  private:

    // Logical volumes
    //
    G4LogicalVolume* experimentalHall_log;
    G4LogicalVolume* detector1_log;
      G4LogicalVolume* detector2_log;
      G4LogicalVolume* detector3_log;
      G4LogicalVolume* detector4_log;

    // Physical volumes
    //
    G4VPhysicalVolume* experimentalHall_phys;
    G4VPhysicalVolume* detector1_phys;
      G4VPhysicalVolume* detector2_phys;
      G4VPhysicalVolume* detector3_phys;
      G4VPhysicalVolume* detector4_phys;
};

#endif
```

C.2. ThesisEventAction.hh

```
// ThesisEventAction.hh
//Created by: W.L. Harrell

#ifndef ThesisEventAction_h
#define ThesisEventAction_h 1

#include "G4UserEventAction.hh"
#include "G4ThreeVector.hh"
#include "globals.hh"

class ThesisEventAction : public G4UserEventAction
{
public:
  ThesisEventAction();
  ~ThesisEventAction();

public:
  void BeginOfEventAction(const G4Event*);
  void EndOfEventAction(const G4Event*);
  void writeTheHitFile(void);
  void writeTheTrajFile(void);
  void drawTracks(const G4Event*);

private:
  G4int trackerCollID;
  G4int EventID;
  G4int hitID;
  G4int trackID;
  G4int parentID;
```

```
  G4int geantID;
  G4int n_trajectories;
  G4int nEvtTraj;
  G4int planeID;
  //G4int printModulo;

  G4double dE_tpc;
  G4double hit_time;
  G4double particle_energy;
  G4double trkLength;
  G4double  particle_charge;  //might be useful to include in hit also
  G4double array_position_x;
  G4double array_position_y;

  G4ThreeVector hitPosition;
  G4ThreeVector trkMomentum;
  G4ThreeVector particle_p;
  G4ThreeVector localPosIn;
  G4ThreeVector localPosOut;
  G4ThreeVector vertexPosition;

  G4String particle_name;
  G4String PhysicsProcess;
  G4int PhysicsProcessInt;

};

#endif
```

C.3. ThesisHit.hh

```
// Thesis_hit.hh
// Created by W.L. Harrell

#ifndef ThesisHit_h
#define ThesisHit_h 1

#include "G4VHit.hh"
#include "G4THitsCollection.hh"
#include "G4Allocator.hh"
#include "G4ThreeVector.hh"

class ThesisHit : public G4VHit
{
  public:

      ThesisHit();
      ~ThesisHit();
      ThesisHit(const ThesisHit &right);
      const ThesisHit& operator=(const ThesisHit &right);
      int operator==(const ThesisHit &right) const;


      inline void *operator new(size_t);
```

```
      inline void operator delete(void *aHit);

      void Draw();
      void Print();

private:
  G4int         trackID;
  G4int         parentID;
  G4double      edep;
  G4ThreeVector pos;
  G4ThreeVector momentum;
  G4double      time;
  G4double      particleEnergy;
  G4String      particleName;
  G4int         geantID;
  G4double      trackLength;
  G4int         planeID;
  G4ThreeVector localPosIn;
  G4ThreeVector localPosOut;
  G4ThreeVector vertexPosition;
  G4String      PhysicsProcess;
  G4int     PhysicsProcessInt;

  public:
  inline void SetTrackID  (G4int track)      { trackID = track; };
  inline G4int GetTrackID()    { return trackID; };

  inline void SetParentID  (G4int parent)      { parentID = parent; };
  inline G4int GetParentID()    { return parentID; };

  inline void SetEdep(G4double de){ edep = de; }
  inline void AddEdep(G4double de){ edep += de; }
  inline G4double GetEdep(){ return edep; }

  inline void SetPos(G4ThreeVector xyz){ pos = xyz; }
  inline G4ThreeVector GetPos(){ return pos; }

  inline void SetMomentum(G4ThreeVector pxyz){ momentum = pxyz; }
  inline G4ThreeVector GetMomentum(){ return momentum; }

  inline void SetParticleEnergy(G4double e1){ particleEnergy = e1; }
  inline G4double GetParticleEnergy(){ return particleEnergy; }

  inline void SetParticleName(G4String name){ particleName = name; }
  inline G4String GetParticleName(){ return particleName; }

  inline void SetGeantID(G4int pid){ geantID = pid; }
  inline G4int GetGeantID(){ return geantID; }

  inline void SetTime(G4double htime){time = htime;}
  inline G4double GetTime(){return time;}

  inline void SetTrackLength(G4double trklength){trackLength =
trklength;}
  inline G4double GetTrackLength(){return trackLength;}
```

```
  inline void SetPlaneID(G4int planeid){ planeID = planeid; }
  inline G4int GetPlaneID(){ return planeID; }

  inline void SetLocalPosIn(G4ThreeVector inxyz){ localPosIn = inxyz; }
  inline G4ThreeVector GetLocalPosIn(){ return localPosIn; }

  inline void SetLocalPosOut(G4ThreeVector outxyz){ localPosOut =
outxyz; }
  inline G4ThreeVector GetLocalPosOut(){ return localPosOut; }

  inline void SetVertexPosition(G4ThreeVector vtxxyz){ vertexPosition =
vtxxyz; }
  inline G4ThreeVector GetVertexPosition(){ return vertexPosition; }

  inline void SetPhysicsProcess(G4String physprocess){ PhysicsProcess =
physprocess; }
  inline G4String GetProcessName(){ return PhysicsProcess; }
};

typedef G4THitsCollection<ThesisHit> ThesisHitsCollection;

extern G4Allocator<ThesisHit> ThesisHitAllocator;

inline void* ThesisHit::operator new(size_t)
{
  void *aHit;
  aHit = (void *) ThesisHitAllocator.MallocSingle();
  return aHit;
}

inline void ThesisHit::operator delete(void *aHit)
{
  ThesisHitAllocator.FreeSingle((ThesisHit*) aHit);
}

#endif
```

C.4.thesisPhysicsList.hh

```
// thesisPhysicsList.hh
// Created by W.L. Harrell

#ifndef thesisPhysicsList_h
#define thesisPhysicsList_h 1

#include "G4VUserPhysicsList.hh"
#include "globals.hh"

class thesisPhysicsList: public G4VUserPhysicsList
{
public:
  thesisPhysicsList();
```

```
  ~thesisPhysicsList();

protected:
  // Construct particle and physics process
  void ConstructParticle();
  void ConstructProcess();
  void ConstructEM();
  void SetCuts();

};

#endif
```

C.5. thesisPrimaryGenerator.hh

```
// thesisPrimaryGeneratorAction.hh
// Created by W.L. Harrell

#ifndef thesisPrimaryGeneratorAction_h
#define thesisPrimaryGeneratorAction_h 1

#include "G4VUserPrimaryGeneratorAction.hh"

class G4GeneralParticleSource;
class G4Event;

class thesisPrimaryGeneratorAction : public
G4VUserPrimaryGeneratorAction
{
  public:
    thesisPrimaryGeneratorAction();
    ~thesisPrimaryGeneratorAction();

  public:
    void GeneratePrimaries(G4Event* anEvent);

  private:

    G4GeneralParticleSource* particleGun;
};

#endif
```

C.6. ThesisTrackerSD.hh

```
// ThesisTrackerSD.hh
// Created by: W.L. Harrell

#ifndef ThesisTrackerSD_h
#define ThesisTrackerSD_h 1

#include "G4VSensitiveDetector.hh"
```

```
#include "ThesisHit.hh"

class G4Step;
class G4HCofThisEvent;
class G4TouchableHistory;

class ThesisTrackerSD : public G4VSensitiveDetector
{
public:
  ThesisTrackerSD(G4String name);
  ~ThesisTrackerSD();

  G4bool ProcessHits(G4Step* aStep, G4TouchableHistory* ROhist);

  void Initialize(G4HCofThisEvent*);
  void EndOfEvent(G4HCofThisEvent*);

private:
  ThesisHitsCollection* trackerCollection;

  G4int HitID;
};


#endif
```

C.7. thesisDetectorConstruction.cc

```
// thesisDetectorConstruction.cc
// Created by W.L. Harrell

#include "thesisDetectorConstruction.hh"

#include "G4Material.hh"
#include "G4Box.hh"
#include "G4Tubs.hh"
#include "G4LogicalVolume.hh"
#include "G4ThreeVector.hh"
#include "G4PVPlacement.hh"
#include "globals.hh"
#include "G4NistManager.hh"
#include "G4SDManager.hh"
#include "ThesisTrackerSD.hh"

// required for visualizer (aab)
#include "G4VisAttributes.hh"
#include "G4Colour.hh"


thesisDetectorConstruction::thesisDetectorConstruction()
 :  experimentalHall_log(0), experimentalHall_phys(0),
    detector1_log(0), detector2_log(0), detector3_log(0),
      detector4_log(0),
```

```
    detector1_phys(0), detector2_phys(0), detector3_phys(0),
      detector4_phys(0)
{;}

thesisDetectorConstruction::~thesisDetectorConstruction()
{
}

G4VPhysicalVolume* thesisDetectorConstruction::Construct()
{

  //--------------------------------------------------- materials

  G4double a;  // atomic mass
  G4double z;  // atomic number
  G4double density;

  G4Material* Ar =
  new G4Material("ArgonGas", z= 18., a= 39.95*g/mole, density=
1.782*mg/cm3);

  G4Material* Al =
  new G4Material("Aluminum", z= 13., a= 26.98*g/mole, density=
2.7*g/cm3);

  //LaBr3
  G4Element* La = new G4Element("Lanthanum", "La", z=57, a=
138.90547*g/mole);
  G4Element* Br = new G4Element("Bromide", "Br", z=35, a=
79.904*g/mole);
  G4Element* N  = new G4Element("Nitrogen", "N", z=7, a=14.01*g/mole);
  G4Element* O  = new G4Element("Oxygen", "O", z=8, a=16.00*g/mole);

  G4Material* LaBr3 = new G4Material("LaBr3", density= 5.08*g/cm3, 2);
  LaBr3->AddElement(La, 1);
  LaBr3->AddElement(Br, 3);

  G4NistManager* man = G4NistManager::Instance();
  man->SetVerbose(1);

  //Air
  G4Material* Air = man->FindOrBuildMaterial("G4_AIR");

  //------------------------------------------------------ volumes

  //------------------------------ experimental hall (world volume)
  G4double expHall_x = 100.0*cm;
  G4double expHall_y = 100.0*cm;
  G4double expHall_z = 100.0*cm;
  G4Box* experimentalHall_box
    = new G4Box("expHall_box",expHall_x,expHall_y,expHall_z);
  experimentalHall_log = new G4LogicalVolume(experimentalHall_box,
                                             Air,"expHall_log",0,0,0);
  experimentalHall_phys = new G4PVPlacement(0,G4ThreeVector(),
```

```
  experimentalHall_log,"expHall",0,false,0);

  //------------------------------ detector 1

  G4double innerRadiusOfdet1 = 0.*cm;
  G4double outerRadiusOfdet1 = 1.0*cm;
  G4double hightOfdet1 = 5.0*cm;
  G4double startAngleOfdet1 = 0.*deg;
  G4double spanningAngleOfdet1 = 360.*deg;
  G4Tubs* detector1_tube = new G4Tubs("det1_tube",innerRadiusOfdet1,
                                      outerRadiusOfdet1,hightOfdet1,

startAngleOfdet1,spanningAngleOfdet1);
  detector1_log = new
G4LogicalVolume(detector1_tube,LaBr3,"det1_log",0,0,0);
  G4double det1Pos_x = 1.54*cm;
  G4double det1Pos_y = 1.54*cm;
  G4double det1Pos_z = 0.0*cm;
  detector1_phys = new G4PVPlacement(0,
            G4ThreeVector(det1Pos_x,det1Pos_y,det1Pos_z),
            detector1_log,"det1",experimentalHall_log,false,0);

   //------------------------------ detector 2

  G4double innerRadiusOfdet2 = 0.*cm;
  G4double outerRadiusOfdet2 = 1.0*cm;
  G4double hightOfdet2 = 5.0*cm;
  G4double startAngleOfdet2 = 0.*deg;
  G4double spanningAngleOfdet2 = 360.*deg;
  G4Tubs* detector2_tube = new G4Tubs("det2_tube",innerRadiusOfdet2,
                                      outerRadiusOfdet2,hightOfdet2,

startAngleOfdet2,spanningAngleOfdet2);
  detector2_log = new
G4LogicalVolume(detector2_tube,LaBr3,"det2_log",0,0,0);
  G4double det2Pos_x = -1.54*cm;
  G4double det2Pos_y = 1.54*cm;
  G4double det2Pos_z = 0.0*cm;
  detector2_phys = new G4PVPlacement(0,
            G4ThreeVector(det2Pos_x,det2Pos_y,det2Pos_z),
            detector2_log,"det2",experimentalHall_log,false,0);

  //------------------------------ detector 3

  G4double innerRadiusOfdet3 = 0.*cm;
  G4double outerRadiusOfdet3 = 1.0*cm;
  G4double hightOfdet3 = 5.0*cm;
  G4double startAngleOfdet3 = 0.*deg;
  G4double spanningAngleOfdet3 = 360.*deg;
  G4Tubs* detector3_tube = new G4Tubs("det3_tube",innerRadiusOfdet3,
                                      outerRadiusOfdet3,hightOfdet3,

startAngleOfdet3,spanningAngleOfdet3);
  detector3_log = new
G4LogicalVolume(detector3_tube,LaBr3,"det3_log",0,0,0);
```

```
  G4double det3Pos_x = -1.54*cm;
  G4double det3Pos_y = -1.54*cm;
  G4double det3Pos_z = 0.0*cm;
  detector3_phys = new G4PVPlacement(0,
            G4ThreeVector(det3Pos_x,det3Pos_y,det3Pos_z),
            detector3_log,"det3",experimentalHall_log,false,0);

   //------------------------------ detector 4

  G4double innerRadiusOfdet4 = 0.*cm;
  G4double outerRadiusOfdet4 = 1.0*cm;
  G4double hightOfdet4 = 5.0*cm;
  G4double startAngleOfdet4 = 0.*deg;
  G4double spanningAngleOfdet4 = 360.*deg;
  G4Tubs* detector4_tube = new G4Tubs("det4_tube",innerRadiusOfdet4,
                                      outerRadiusOfdet4,hightOfdet4,

startAngleOfdet4,spanningAngleOfdet4);
  detector4_log = new
G4LogicalVolume(detector4_tube,LaBr3,"det4_log",0,0,0);
  G4double det4Pos_x = 1.54*cm;
  G4double det4Pos_y = -1.54*cm;
  G4double det4Pos_z = 0.0*cm;
  detector4_phys = new G4PVPlacement(0,
            G4ThreeVector(det4Pos_x,det4Pos_y,det4Pos_z),
            detector4_log,"det4",experimentalHall_log,false,0);

 //============================
 // Define Sensitive Detector
 //============================
  G4SDManager* SensitiveDetectorMan = G4SDManager::GetSDMpointer();
  G4String Detectorname = "/thesis/detectorsRO";
  ThesisTrackerSD* DetectorRO = new ThesisTrackerSD(Detectorname);
  SensitiveDetectorMan->AddNewDetector(DetectorRO);
  detector1_log->SetSensitiveDetector(DetectorRO);
  detector2_log->SetSensitiveDetector(DetectorRO);
  detector3_log->SetSensitiveDetector(DetectorRO);
  detector4_log->SetSensitiveDetector(DetectorRO);

  //--------- Visualization attributes -------------------------------
  // required for visualization (aab)
  G4Colour white(1.0,1.0,1.0);
  G4Colour gray(1.0,1.0,1.0);
  G4Colour black(1.0,1.0,1.0);
  G4Colour red(1.0,1.0,1.0);
  G4Colour green(1.0,1.0,1.0);
  G4Colour blue(1.0,1.0,1.0);
  G4Colour cyan(1.0,1.0,1.0);
  G4Colour magenta(1.0,1.0,1.0);
  G4Colour yellow(1.0,1.0,1.0);


  G4VisAttributes* VisAtt_white= new
G4VisAttributes(G4Colour(1.0,1.0,1.0));
```

```
  G4VisAttributes* VisAtt_red= new
G4VisAttributes(G4Colour(1.0,0.0,0.0));
  G4VisAttributes* VisAtt_blue= new
G4VisAttributes(G4Colour(0.0,0.0,1.0));

  experimentalHall_log  ->SetVisAttributes(VisAtt_white);
  detector1_log->SetVisAttributes(VisAtt_blue);
  detector2_log->SetVisAttributes(VisAtt_blue);
  detector3_log->SetVisAttributes(VisAtt_blue);
  detector4_log->SetVisAttributes(VisAtt_blue);

  //------------------------------------------------------------------
  return experimentalHall_phys;

}
```

C.8. ThesisEventAction.cc

```
// ThesisEventAction.cc
// Created by: W.L. Harrell

#include "ThesisEventAction.hh"
#include "ThesisHit.hh"
#include "ThesisEventParameters.icc"

#include "G4Event.hh"
#include "G4EventManager.hh"
#include "G4HCofThisEvent.hh"
#include "G4VHitsCollection.hh"
#include "G4TrajectoryContainer.hh"
#include "G4Trajectory.hh"
#include "G4VVisManager.hh"
#include "G4UnitsTable.hh"
#include "G4SDManager.hh"
#include "G4UImanager.hh"
#include "G4ios.hh"
#include "G4ThreeVector.hh"
#include "G4VProcess.hh"

#include "Randomize.hh"
#include <fstream>
#include <iomanip>
#include <iostream>

using namespace std;


ThesisEventAction::ThesisEventAction()
{
  trackerCollID = -1;
}


ThesisEventAction::~ThesisEventAction()
```

```
{;}


void ThesisEventAction::BeginOfEventAction(const G4Event*)
{
   G4SDManager * SensitiveDetector1 = G4SDManager::GetSDMpointer();
 if(trackerCollID<0)
   {
     G4String colNam;
     trackerCollID = SensitiveDetector1-
>GetCollectionID(colNam="trackerCollection");
   }

}

void ThesisEventAction::EndOfEventAction(const G4Event* evt)
{

  EventID = evt->GetEventID();
  G4cout << ">>> Event " << EventID << G4endl;

  if(trackerCollID<0) return;
  G4HCofThisEvent* HCE = evt->GetHCofThisEvent();
      ThesisHitsCollection* THC = 0;

  if(HCE)
    THC = (ThesisHitsCollection*)(HCE->GetHC(trackerCollID));

  if(THC)
    {
      int n_hit = THC->entries();
      G4cout << "     " <<n_hit<< " hits are stored in
ThesisHitsCollection." << G4endl;
      hitID = -999;
      trackID = -999;
      parentID = -999;
      dE_tpc = -999;
      hitPosition = -999;
      trkMomentum = -999;
      hit_time = -999/ns;
      particle_energy = -999;
      particle_name = -999;
      geantID = -999;
      trkLength = -999;
      planeID = -999;
      localPosIn = -999;
      localPosOut = -999;
      vertexPosition = -999;
      PhysicsProcess = -999;
      PhysicsProcessInt = -999;

      if (n_hit==0)
      {
        //track did not interact with sensitive detector
        G4String filename="thesis_hit.out";
```

```
        ofstream evfile(filename, ios::app);
        evfile <<"-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1"<<G4endl;
        evfile.close();
      }
      else {
      for(int i=0;i<n_hit;i++)
        {
          hitID = i;
          trackID = (*THC)[i]->GetTrackID();
          parentID = (*THC)[i]->GetParentID();
          dE_tpc = (*THC)[i]->GetEdep();
          hitPosition = (*THC)[i]->GetPos();
          trkMomentum = (*THC)[i]->GetMomentum();
          hit_time = (*THC)[i]->GetTime()/ns;
          particle_energy = (*THC)[i]->GetParticleEnergy();
          particle_name = (*THC)[i]->GetParticleName();
          geantID = (*THC)[i]->GetGeantID();
          trkLength = (*THC)[i]->GetTrackLength()/cm;
          planeID = (*THC)[i]->GetPlaneID();
          localPosIn = (*THC)[i]->GetLocalPosIn();
          localPosOut = (*THC)[i]->GetLocalPosOut();
          vertexPosition = (*THC)[i]->GetVertexPosition();
          PhysicsProcess = (*THC)[i]->GetProcessName();
          // Give the physics processes codes to make post-processing
simpler
          if(PhysicsProcess == "PrimaryGen"){
            PhysicsProcessInt = 0;
          }
          if(PhysicsProcess == "phot"){
            PhysicsProcessInt = 1;
          }
          if(PhysicsProcess == "compt"){
            PhysicsProcessInt = 2;
          }
          if(PhysicsProcess == "conv"){
            PhysicsProcessInt = 3;
          }
          if(PhysicsProcess == "annihil"){
            PhysicsProcessInt = 4;
          }
          if(PhysicsProcess == "eBrem"){
            PhysicsProcessInt = 5;
          }
          if(PhysicsProcess == "eIoni"){
            PhysicsProcessInt = 6;
          }
          //find track entry point into plane for primaries
          if (localPosIn[0]==0 && localPosIn[1]==0 && localPosIn[2]==0
&& parentID==0) {
            for (int j=0;j<n_hit;j++) {
            G4ThreeVector tmpIn = (*THC)[j]->GetLocalPosIn();
            if ((trackID == (*THC)[j]->GetTrackID()) && (planeID ==
(*THC)[j]->GetPlaneID()) && (tmpIn[0]!=0 || tmpIn[1]!=0 ||
tmpIn[2]!=0)) {
```

```
              localPosIn[0] = tmpIn[0];
              localPosIn[1] = tmpIn[1];
              localPosIn[2] = tmpIn[2];
              break;
            }
            }
          }

          //find track entry point into plane for secondaries
          //equivalent to vertex position
          if (localPosIn[0]==0 && localPosIn[1]==0 && localPosIn[2]==0
&& parentID!=0) {
            localPosIn[0] = vertexPosition[0];
            localPosIn[1] = vertexPosition[1];
            localPosIn[2] = vertexPosition[2];
          }

          //find track exit point from plane
          if (localPosOut[0]==0 && localPosOut[1]==0 &&
localPosOut[2]==0) {
            for (int j=0;j<n_hit;j++) {
            //primary
            if (parentID==0){
              G4ThreeVector tmpOut = (*THC)[j]->GetLocalPosOut();
              if ((trackID == (*THC)[j]->GetTrackID()) && (planeID ==
(*THC)[j]->GetPlaneID()) && (tmpOut[0]!=0 || tmpOut[1]!=0 ||
tmpOut[2]!=0)) {
                localPosOut[0] = tmpOut[0];
                localPosOut[1] = tmpOut[1];
                localPosOut[2] = tmpOut[2];
                break;
              }
            }

            //secondary
            if (parentID!=0){
              //if ((trackID == (*THC)[j]->GetTrackID()) && ((trackID
!= (*THC)[j+1]->GetTrackID()) || (j==n_hit-1))) {
              if ((trackID == (*THC)[j]->GetTrackID())) {
                //find position of last hit of track = localPosOut
                G4ThreeVector tmpOut = (*THC)[j]->GetPos();
                localPosOut[0] = tmpOut[0];
                localPosOut[1] = tmpOut[1];
                localPosOut[2] = tmpOut[2];
              }
            }

            }
          }
          G4int npadrow   = 100;
          G4int npadcol = 100;
          G4double col_offset = -25*cm;
          G4double row_offset = -25*cm;
```

```
          //if hit occurs in gap between pads then array position will
be -999
          array_position_x = -999;
          array_position_y = -999;

          for (G4int i=1; i<=npadrow; i++) {
            G4double pad_min_y = row_offset+0.5*cm*(i-1);
            G4double pad_max_y = row_offset+0.5*cm*i;
            //G4double pad_min_z = row_offset+(2.50)*cm*(i-1)-1.2*cm;
            //G4double pad_max_z = row_offset+(2.50)*cm*(i-1)+1.2*cm;
            if (hitPosition[1]>pad_min_y && hitPosition[1]<pad_max_y)
            {
              array_position_y = i;
              break;
            }
          }

          for (G4int j=1; j<=npadcol; j++) {
            G4double pad_min_x = col_offset+0.5*cm*(j-1);
            G4double pad_max_x = col_offset+0.5*cm*j;
            //G4double pad_min_x = col_offset+(0.47)*cm*(j-1)-0.21*cm;
            //G4double pad_max_x = col_offset+(0.47)*cm*(j-1)+0.21*cm;
            if (hitPosition[0]>pad_min_x && hitPosition[0]<pad_max_x)
            {
              array_position_x = j;
              break;
            }
          }
          //write output file
          if (writeHitFile)
            writeTheHitFile();
        }
      //last line of hit file needs to be distinguishable
      G4String filename="thesis_hit.out";
      ofstream evfile(filename, ios::app);
      evfile <<"-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -
1 -1 -1 -1 -1 -1 -1"<<G4endl;
      evfile.close();
      if(drawSDTracks)
        drawTracks(evt);
      }

    }
  /* //Uncomment to write Trajectory file
  // extract the stored trajectories of all primaries and secondaries.
  G4TrajectoryContainer* trajectoryContainer = evt-
>GetTrajectoryContainer();
  n_trajectories = 0;
  if (trajectoryContainer) n_trajectories = trajectoryContainer-
>entries();

  if (G4VVisManager::GetConcreteInstance())
    {
      nEvtTraj = n_trajectories;
```

```
      for (G4int i=0; i<n_trajectories; i++)
        {
        G4Trajectory* trj = (G4Trajectory*)((*(evt-
>GetTrajectoryContainer()))[i]);
        trj->DrawTrajectory(1000);

        trackID = trj->GetTrackID();
        parentID = trj->GetParentID();
        particle_name = trj->GetParticleName();
        geantID = trj->GetPDGEncoding();
        particle_charge = trj->GetCharge();
        //particle_energy = trj->GetEnergy();
        particle_p = trj->GetInitialMomentum();
        //PhysicsProcess = trj->GetProcessName();
        //write output file
        if (writeTrajFile)
          writeTheTrajFile();
        }
    }

  */


}

void ThesisEventAction::writeTheHitFile(void)
{
  G4String filename="thesis_hit.out";
  ofstream evfile(filename, ios::app);
  // ofstream evfile(filename);
  evfile <<hitID<<" "
       <<trackID<<" "
       <<parentID<<" "
       <<particle_name << " "
       <<geantID<< " "
       <<particle_energy/MeV << " "
       <<dE_tpc/MeV << " "
       <<hit_time/ns << " "
       <<hitPosition[0]/cm << " "
       <<hitPosition[1]/cm << " "
       <<hitPosition[2]/cm << " "
       <<trkMomentum[0]/MeV << " "
       <<trkMomentum[1]/MeV << " "
       <<trkMomentum[2]/MeV << " "
       <<trkLength/cm << " "
       <<array_position_x << " "
       <<array_position_y << " "
       <<planeID << " "
       <<localPosIn[0]/cm << " "
       <<localPosIn[1]/cm << " "
       <<localPosIn[2]/cm << " "
       <<localPosOut[0]/cm << " "
       <<localPosOut[1]/cm << " "
       <<localPosOut[2]/cm << " "
       <<PhysicsProcessInt << " "
```

```
        <<G4endl;


  evfile.close();

}
/* //Uncomment to write trajectory file
void ThesisEventAction::writeTheTrajFile(void)
{
  G4String filename="thesis_traj.out";
  ofstream evfile(filename, ios::app);

  evfile <<EventID<<" "
       <<nEvtTraj << " "
       <<trackID << " "
       <<parentID << " "
       <<particle_name << " "
       <<geantID << " "
       <<particle_charge << " "
       <<particle_p[0] << " "
       <<particle_p[1] << " "
       <<particle_p[2] << " "
       //<<PhysicsProcessInt << " "
       <<G4endl;
  evfile.close();

}
*/

// commented to turn off visualization
void ThesisEventAction::drawTracks(const G4Event* evt) {

  if(G4VVisManager::GetConcreteInstance())
    {
      G4UImanager::GetUIpointer()-
>ApplyCommand("/vis/scene/notifyHandlers");
      G4TrajectoryContainer* trajContainer = evt-
>GetTrajectoryContainer();
      G4int n_trajectories = 0;

      if(trajContainer) n_trajectories = trajContainer->entries();
      for (G4int i=0; i<n_trajectories; i++) {
      G4Trajectory* trj = (G4Trajectory*)(*trajContainer)[i];
        {
          trj->DrawTrajectory();
        }
      }
    }
}
```

C.9. ThesisEventParameters.icc

```
// ThesisEventParameters.icc
// Created by W.L. Harrell
//*******************************************************
//---------------Event Action Parameters

  G4bool writeHitFile = true;   // Write Individual read out plane Hits
  G4bool writeTrajFile = true;    // Write all trajectories for each
event
  G4bool readFile = true;    // generate primaries from BUU
distributions

  G4bool drawSDTracks = true;  // Draw Tracks in the crystal
  G4int VerboseLevel= 1;
```

C.10. ThesisHit.cc

```
// ThesisHit.cc
// Created by W.L. Harrell

#include "ThesisHit.hh"
#include "G4UnitsTable.hh"
#include "G4VVisManager.hh"
#include "G4Circle.hh"
#include "G4Colour.hh"
#include "G4VisAttributes.hh"
#include "G4VProcess.hh"
#include <iomanip>
//#include "g4std/iomanip"


G4Allocator<ThesisHit> ThesisHitAllocator;

ThesisHit::ThesisHit()
{
  trackID = -999;
  parentID = -999;
  edep=0;
  pos = G4ThreeVector(0.,0.,0.);
  momentum = G4ThreeVector(0.,0.,0.);
  time = 0.;
  particleEnergy = 0.;
  geantID = 0;
  trackLength = -999;
  planeID = -999;
  localPosIn = G4ThreeVector(0.,0.,0.);
  localPosOut = G4ThreeVector(0.,0.,0.);
  vertexPosition = G4ThreeVector(0.,0.,0.);
  PhysicsProcessInt = -999;
  //PhysicsProcess = 0;

}
```

```
ThesisHit::~ThesisHit()
{;}

ThesisHit::ThesisHit(const ThesisHit &right): G4VHit(right)
{
  trackID = right.trackID;
  parentID = right.parentID;
  edep = right.edep;
  pos = right.pos;
  momentum = right.momentum;
  time = right.time;
  particleEnergy = right.particleEnergy;
  particleName = right.particleName;
  geantID = right.geantID;
  trackLength = right.trackLength;
  planeID = right.planeID;
  localPosIn = right.localPosIn;
  localPosOut = right.localPosOut;
  vertexPosition = right.vertexPosition;
  PhysicsProcess = right.PhysicsProcess;
  PhysicsProcessInt = right.PhysicsProcessInt;

}

const ThesisHit& ThesisHit::operator=(const ThesisHit &right)
{
  trackID = right.trackID;
  parentID = right.parentID;
  edep = right.edep;
  pos = right.pos;
  momentum = right.momentum;
  time = right.time;
  particleEnergy = right.particleEnergy;
  particleName = right.particleName;
  geantID = right.geantID;
  trackLength = right.trackLength;
  planeID = right.planeID;
  localPosIn = right.localPosIn;
  localPosOut = right.localPosOut;
  vertexPosition = right.vertexPosition;
  PhysicsProcess = right.PhysicsProcess;
  PhysicsProcessInt = right.PhysicsProcessInt;

  return *this;
}

int ThesisHit::operator==(const ThesisHit &right) const
{
  return 0;
}

void ThesisHit::Draw()
{
    G4VVisManager* pVVisManager = G4VVisManager::GetConcreteInstance();
```

```
  if(pVVisManager)
  {
    G4Circle circle(pos);
    circle.SetScreenSize(0.4);
    circle.SetFillStyle(G4Circle::filled);
    G4Colour colour(1.,0.,0.);
    G4VisAttributes attribs(colour);
    circle.SetVisAttributes(attribs);
    pVVisManager->Draw(circle);
    }
}

void ThesisHit::Print()
{;}
```

C.11. thesisPhysicsList.cc

```
// thesisPhysicsList.cc
// Created by W.L. Harrell

#include "thesisPhysicsList.hh"
#include "G4ParticleTypes.hh"

// required for ConstructEM (aab)
#include "G4ProcessManager.hh"

thesisPhysicsList::thesisPhysicsList()
{;}

thesisPhysicsList::~thesisPhysicsList()
{;}

void thesisPhysicsList::ConstructParticle()
{
  // Ionization in material requires e- (aab)
  G4Electron::ElectronDefinition();

  // Ionization in material requires e+ (aab)
  G4Positron::PositronDefinition();

  // Brem & pair prod requires gamma (aab)
  G4Gamma::GammaDefinition();
}

void thesisPhysicsList::ConstructProcess()
{
  // Define transportation process
  AddTransportation();

  // Add processes that you expect the proton to experience (aab)
  ConstructEM();
}

// Add electron physics (aab)
```

```
#include "G4eMultipleScattering.hh"
#include "G4eIonisation.hh"
#include "G4eBremsstrahlung.hh"

// Add e+ physics (aab)
#include "G4eplusAnnihilation.hh"

// Add gamma physics (aab)
#include "G4PhotoElectricEffect.hh"
#include "G4ComptonScattering.hh"
#include "G4GammaConversion.hh"

void thesisPhysicsList::ConstructEM()
{
  theParticleIterator->reset();
  while( (*theParticleIterator)() ){
    G4ParticleDefinition* particle = theParticleIterator->value();
    G4ProcessManager* pmanager = particle->GetProcessManager();
    G4String particleName = particle->GetParticleName();

    if (particleName == "e-") {
      //electron
      pmanager->AddProcess(new G4eMultipleScattering, -1, 1, 1);
      pmanager->AddProcess(new G4eIonisation,         -1, 2, 2);
      pmanager->AddProcess(new G4eBremsstrahlung,     -1, 3, 3);

    }
    else if (particleName == "e+") {
      //positron
      pmanager->AddProcess(new G4eMultipleScattering, -1, 1, 1);
      pmanager->AddProcess(new G4eIonisation,         -1, 2, 2);
      pmanager->AddProcess(new G4eBremsstrahlung,     -1, 3, 3);
      pmanager->AddProcess(new G4eplusAnnihilation,    0,-1, 4);

    }
    else if (particleName == "gamma") {
      // gamma
      pmanager->AddDiscreteProcess(new G4PhotoElectricEffect);
      pmanager->AddDiscreteProcess(new G4ComptonScattering);
      pmanager->AddDiscreteProcess(new G4GammaConversion);

    }
  }
}

SetVerboseLevel(0);

void thesisPhysicsList::SetCuts()
{
  if (verboseLevel >0){
    G4cout << "thesisPhysicsList::SetCuts:";
    G4cout << "CutLength : " << G4BestUnit(defaultCutValue,"Length") <<
G4endl;
  }
  // Set cutvalues for gamma rays, electrons, and positrons
```

```
  SetCutValue(1.*keV, "gamma");
  SetCutValue(20.*keV, "e-");
  SetCutValue(100.*keV, "e+");
  if (verboseLevel>0) DumpCutValuesTable();
}
```

C.12. thesisPrimaryGeneratorAction.cc

```
// thesisPrimaryGeneratorAction.cc
// Created by W.L. Harrell

#include "thesisPrimaryGeneratorAction.hh"

#include "G4Event.hh"
#include "G4GeneralParticleSource.hh"
#include "G4ParticleTable.hh"
#include "G4ParticleDefinition.hh"
#include "globals.hh"

//This simulation uses the General Particle Source (gps)
//It is defined in the vis.mac file
thesisPrimaryGeneratorAction::thesisPrimaryGeneratorAction()
{
   particleGun = new G4GeneralParticleSource();
}

thesisPrimaryGeneratorAction::~thesisPrimaryGeneratorAction()
{
  delete particleGun;
}

void thesisPrimaryGeneratorAction::GeneratePrimaries(G4Event* anEvent)
{
  particleGun->GeneratePrimaryVertex(anEvent) ;
}
```

C.13. ThesisTrackerSD.cc

```
// ThesisTrackerSD.cc
// Created by: W.L. Harrell

#include "ThesisTrackerSD.hh"
#include "ThesisHit.hh"

#include "G4HCofThisEvent.hh"
#include "G4TouchableHistory.hh"
#include "G4Track.hh"
#include "G4Step.hh"
#include "G4ios.hh"

#include "G4SDManager.hh"
```

```
#include "G4ParticleDefinition.hh"
#include "G4ParticleTypes.hh"
#include "G4Ions.hh"
#include "G4ThreeVector.hh"
#include "G4VProcess.hh"


ThesisTrackerSD::ThesisTrackerSD(G4String
name):G4VSensitiveDetector(name)
{
  G4String HCname;
  collectionName.insert(HCname="trackerCollection");
}

ThesisTrackerSD::~ThesisTrackerSD(){ }


void ThesisTrackerSD::Initialize(G4HCofThisEvent* HCE)
{
  trackerCollection = new
ThesisHitsCollection(SensitiveDetectorName,collectionName[0]);
  static G4int HitID = -1;
}

G4bool ThesisTrackerSD::ProcessHits(G4Step* aStep,G4TouchableHistory*)
{
  G4double edep = aStep->GetDeltaEnergy();
  if(edep==0.) return false;

  G4ParticleDefinition* particleType = aStep->GetTrack()-
>GetDefinition();

  ThesisHit* newHit = new ThesisHit();
  newHit->SetTrackID        (aStep->GetTrack()->GetTrackID());
  newHit->SetParentID       (aStep->GetTrack()->GetParentID());
  newHit->SetEdep           (edep);
  newHit->SetPos            (aStep->GetPostStepPoint()->GetPosition());
  newHit->SetMomentum       (aStep->GetTrack()->GetMomentum());
  newHit->SetParticleName   (particleType->GetParticleName());
  newHit->SetParticleEnergy (aStep->GetPreStepPoint()-
>GetKineticEnergy());
  newHit->SetTime           (aStep->GetTrack()->GetGlobalTime());
  newHit->SetTrackLength    (aStep->GetTrack()->GetTrackLength());
  newHit->SetGeantID        (particleType->GetPDGEncoding());
  newHit->SetPlaneID        (aStep->GetTrack()->GetVolume()-
>GetCopyNo());
  newHit->SetVertexPosition (aStep->GetTrack()->GetVertexPosition());
  if(aStep->GetTrack()->GetCreatorProcess())
  newHit->SetPhysicsProcess (aStep->GetTrack()->GetCreatorProcess()-
>GetProcessName());
  else
  newHit->SetPhysicsProcess ("PrimaryGen");

  //collect info on pre and post points
  G4StepPoint* point1 = aStep->GetPreStepPoint();
```

```
  G4String thePrePVname = point1->GetPhysicalVolume()->GetName();

  G4StepPoint* point2 = aStep->GetPostStepPoint();
  G4String thePostPVname = point2->GetPhysicalVolume()->GetName();

  G4int itrk = aStep->GetTrack()->GetTrackID();
  // check if trk is entering the SD plane
  if(point1->GetStepStatus() == fGeomBoundary) {
   newHit->SetLocalPosIn (aStep->GetPreStepPoint()->GetPosition());
   G4ThreeVector LocalPosIn = (aStep->GetPreStepPoint()-
>GetPosition());
  }

  // check if trk is exiting the SD plane
  if(point2->GetStepStatus() == fGeomBoundary) {
    newHit->SetLocalPosOut (aStep->GetPostStepPoint()->GetPosition());
   G4ThreeVector LocalPosOut = (aStep->GetPostStepPoint()-
>GetPosition());
  }

  HitID = trackerCollection->insert( newHit );
  return true;
}


void ThesisTrackerSD::EndOfEvent(G4HCofThisEvent*HCE)
{
  G4String HCname = collectionName[0];

  static G4int HCID = -1;
  if(HCID<0)
    HCID = G4SDManager::GetSDMpointer()->GetCollectionID(HCname);
  HCE->AddHitsCollection(HCID,trackerCollection);

  if (verboseLevel>0) {
     G4int NbHits = trackerCollection->entries();
     G4cout << "\n-------->Hits Collection: in this event they are " <<
NbHits
            << " hits in the tracker chambers: " << G4endl;
     for (G4int i=0;i<NbHits;i++) (*trackerCollection)[i]->Print();
    }
}
```

C.14. GNUmakefile.mk

```
# $Id: GNUmakefile for Directional Pair Spectrometer
# --------------------------------------------------------------
# GNUmakefile.  W.L. Harrell
# --------------------------------------------------------------

name := thesis
G4TARGET := $(name)
G4EXLIB := true

ifndef G4INSTALL
```

```
  G4INSTALL = ../../..
endif

.PHONY: all
all: lib bin

include $(G4INSTALL)/config/binmake.gmk
```

C.15. thesis.cc

```
// thesis.cc
// Created by W.L. Harrell
// --------------------------------------------------------------
//      GEANT 4 - Thesis_Simulation
//      Directional Pair Spectrometer
// --------------------------------------------------------------

#include "G4RunManager.hh"
#include "G4UImanager.hh"

#include "thesisDetectorConstruction.hh"
#include "thesisPhysicsList.hh"
#include "thesisPrimaryGeneratorAction.hh"
#include "ThesisEventAction.hh"

// required for visualizer
#include "G4UIterminal.hh"
#include "G4UItcsh.hh"
#ifdef G4VIS_USE
#include "G4VisExecutive.hh"
#endif

int main()
{
  // Construct the default run manager
  //
  G4RunManager* runManager = new G4RunManager;

  // set mandatory initialization classes
  //
  G4VUserDetectorConstruction* detector = new
thesisDetectorConstruction;
  runManager->SetUserInitialization(detector);
  //
  G4VUserPhysicsList* physics = new thesisPhysicsList;
  runManager->SetUserInitialization(physics);

  // set mandatory user action class
  //
  G4VUserPrimaryGeneratorAction* gen_action = new
thesisPrimaryGeneratorAction;
  runManager->SetUserAction(gen_action);
```

```
  //Set user action classes
  ThesisEventAction* event_action = new ThesisEventAction;
  runManager->SetUserAction(event_action);

  // Initialize G4 kernel
  //
  runManager->Initialize();

  // Get the pointer to the UI manager and set verbosities
  //
  G4UImanager* UI = G4UImanager::GetUIpointer();
  // handle run setup through vis.mac file (aab)
  //UI->ApplyCommand("/run/verbose 1");
  //UI->ApplyCommand("/event/verbose 1");
  //UI->ApplyCommand("/tracking/verbose 1");


  // required for visualizer (aab)
#ifdef G4VIS_USE
  G4VisManager* visManager = new G4VisExecutive;
  visManager->Initialize();
#endif

  G4UIsession * session = 0;
#ifdef G4UI_USE_TCSH
  session = new G4UIterminal(new G4UItcsh);
#else
  session = new G4UIterminal();
#endif
#ifdef G4VIS_USE
  UI->ApplyCommand("/control/execute vis.mac");
#endif
  session->SessionStart();
  delete session;

#ifdef G4VIS_USE
  delete visManager;
#endif
  delete runManager;

  return 0;
}
```

C.16. vis.mac

```
#
# Macro file for the initialization phase of "thesis.cc"
# when running in interactive mode
#
# Created by W.L. Harrell
#
# Sets some default verbose
#
/control/verbose 2
/control/saveHistory
```

```
/run/verbose 2
#
# create empty scene
#
/vis/scene/create
#
# Create a scene handler for a specific graphics system
# (Edit the next line(s) to choose another graphic system)
#
# Use this open statement to get an .eps and .prim files
# suitable for viewing in DAWN.
#/vis/open DAWNFILE
#
# Use this open statement instead for OpenGL in immediate mode.
#/vis/open OGLIX
#
# Use this open statement instead to get a HepRep version 1 file
# suitable for viewing in WIRED.
#/vis/open HepRepFile
#
# Use this open statement instead to get a HepRep version 2 file
# suitable for viewing in WIRED.
#/vis/open HepRepXML
#
# Output an empty detector
/vis/viewer/set/viewpointThetaPhi 70 20 deg
/vis/viewer/flush
#
# output all track steps
/tracking/verbose 0
# Draw trajectories at end of event, showing trajectory points as
# markers of size 2 pixels
/vis/scene/add/trajectories
/vis/modeling/trajectories/create/drawByCharge
/vis/modeling/trajectories/drawByCharge-0/default/setDrawStepPts true
/vis/modeling/trajectories/drawByCharge-0/default/setStepPtsSize 2
# (if too many tracks cause core dump => /tracking/storeTrajectory 0)
#
# Draw Hits
/vis/scene/add/hits
#
# To draw gammas only
#/vis/filtering/trajectories/create/particleFilter
#/vis/filtering/trajectories/particleFilter-0/add gamma
#
# To draw charged particles only
#/vis/filtering/trajectories/particleFilter-0/invert true
#
# Many other options available with /vis/modeling and /vis/filtering.
# For example, select colour by particle ID
#/vis/modeling/trajectories/create/drawByParticleID
#/vis/modeling/trajectories/drawByParticleID-0/set e- red
#
#/event/verbose 0
```

```
# This is where the GPS source is defined.  Adjust the below to change
the
# source characteristics.  First rot1, and centre setting is for 0
degree
# orientation.  Second is for 45 degree orientation.  Third is for 90
degree
# orientation.
/gps/verbose 2
/gps/source/clear
/gps/source/add .0075
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
/gps/ene/type Mono
/gps/ene/mono .34714 MeV

/gps/source/add .0076
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
/gps/ene/type Mono
/gps/ene/mono .82610 MeV

/gps/source/add 99.85
#/gps/source/add 100
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
```

```
/gps/ene/type Mono
#/gps/ene/mono 2.6 MeV
/gps/ene/mono 1.173228 MeV

/gps/source/add 99.9826
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
/gps/ene/type Mono
#/gps/ene/mono 10. MeV
/gps/ene/mono 1.332492 MeV

/gps/source/add .00120
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
/gps/ene/type Mono
/gps/ene/mono 2.15857 MeV

/gps/source/add .000002
/gps/particle gamma
/gps/pos/type Plane
/gps/pos/shape Circle
#/gps/pos/rot1 -1. 0. 0.
/gps/pos/rot1 -1. 0. 1.
#/gps/pos/rot1 0. 0. 1.
/gps/pos/radius 0.3 cm
#/gps/pos/centre 0. 0. 32.5 cm
/gps/pos/centre 27.65 0. 27.65 cm
#/gps/pos/centre 32.5 0. 0 cm
/gps/ang/type cos
/gps/ang/maxtheta 11 deg
/gps/ene/type Mono
/gps/ene/mono 2.505692 MeV


/vis/scene/endOfEventAction accumulate
```

```
#
# At end of each run, an automatic flush causes graphical output.
#/run/beamOn 1
# When you exit Geant4, you will find a file called scene-0.heprep.zip.
# Unzipping this will give you three separate HepRep files suitable for
# viewing in WIRED.
# The first file will contain just detector geometry.
# The second file will contain the detector plus one event.
# The third file will contain the detector plus ten events.
```

C.17. MakeRootTree_hit.C

```
//
// MakeRootTree_hit.C v1.0 April 13, 2010
// Created by: A. A. Bickley
// read in Geant hit file and create root tree

gROOT->Reset();

#include <string>
#include <fstream>
#include "Riostream.h"
#include "math.h"


void MakeRootTree_hit(const char *inputFile =
"thesis_hit_45_try2.out",const char
*outputFile = "thesis_hit_45_try2.root")
{

  //=======
  // Files
  //=======
  string inputFileName(inputFile);
  ifstream inFile(inputFile);
  if (!inFile)    {
    cout<<"failed to find inFile......."<<endl;
    break;
  }

  string outputFileName(outputFile);
  TFile *f = new TFile(outputFileName.c_str(), "recreate");


  //=============
  // Definitions
  //=============
  Double_t tmp[25] =
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
  Int_t nEvents = 0;
  Int_t nEntries = 0;
  Int_t ihit = 0;
  Int_t ihit_prev = 0;
```

```
Bool_t IsFirstEvent = kTRUE;

const Int_t nhit = 1000;
Int_t evt[nhit] = {0};
Int_t hitID[nhit] = {0};
Int_t trackID[nhit] = 0;
Int_t parentID[nhit] = 0;
Char_t particleName[10];
Double_t geantID[nhit] = {0.0};
Double_t particle_energy[nhit] = {0.0};
Double_t dE_tpc[nhit] = {0.0};
Double_t hit_time[nhit] = {0.0};
Double_t hitPositionX[nhit] = {0.0};
Double_t hitPositionY[nhit] = {0.0};
Double_t hitPositionZ[nhit] = {0.0};
Double_t trkMomentumX[nhit] = {0.0};
Double_t trkMomentumY[nhit] = {0.0};
Double_t trkMomentumZ[nhit] = {0.0};
Double_t trkLength[nhit] = {0.0};
Double_t array_position_x[nhit] = {0.0};
Double_t array_position_y[nhit] = {0.0};
Double_t planeID[nhit] = {0.0};
Double_t localPosInX[nhit] = {0.0};
Double_t localPosInY[nhit] = {0.0};
Double_t localPosInZ[nhit] = {0.0};
Double_t localPosOutX[nhit] = {0.0};
Double_t localPosOutY[nhit] = {0.0};
Double_t localPosOutZ[nhit] = {0.0};
Double_t PhysicsProcess[nhit] = {0.0};


//====================
// Initialize Arrays
//====================
for (Int_t i=0; i<nhit; i++)
  {
    evt[i] = -999;
    hitID[i] = -999;
    trackID[i] = -999;
    parentID[i] = -999;
    geantID[i] = -999;
    particle_energy[i] = -999;
    dE_tpc[i] = -999;
    hit_time[i] = -999;
    hitPositionX[i] = 9999;
    hitPositionY[i] = -999;
    hitPositionZ[i] = -999;
    trkMomentumX[i] = -999;
    trkMomentumY[i] = -999;
    trkMomentumZ[i] = -999;
    trkLength[i] = -999;
    array_position_x[i] = -999;
    array_position_y[i] = -999;
    planeID[i] = -999;
    localPosInX[i] = -999;
```

```
      localPosInY[i] = -999;
      localPosInZ[i] = -999;
      localPosOutX[i] = -999;
      localPosOutY[i] = -999;
      localPosOutZ[i] = -999;
      PhysicsProcess[i] = -999;

   }


  //==========
  // Hit Tree
  //==========
  TTree *t = new TTree("Hits", "hits in sensitive detector");
  TBranch *b;

  Int_t nHitsPerEvent = 0;
  b=t->Branch("nHitsPerEvent",&nHitsPerEvent,"nHitsPerEvent/I");
  b=t-
>Branch("hit",&evt,"evt[nHitsPerEvent]/I:hitID[nHitsPerEvent]/I:trackID
[nHitsPerEvent]/I:parentID[nHitsPerEvent]/I:geantID[nHitsPerEvent]/D:pa
rticle_energy[nHitsPerEvent]/D:dE_tpc[nHitsPerEvent]/D:hit_time[nHitsPe
rEvent]/D:hitPositionX[nHitsPerEvent]/D:hitPositionY[nHitsPerEvent]/D:h
itPositionZ[nHitsPerEvent]/D:trkMomentumX[nHitsPerEvent]/D:trkMomentumY
[nHitsPerEvent]/D:trkMomentumZ[nHitsPerEvent]/D:trkLength[nHitsPerEvent
]/D:array_position_x[nHitsPerEvent]/D:array_position_y[nHitsPerEvent]/D
:planeID[nHitsPerEvent]/D:localPosInX[nHitsPerEvent]/D:localPosInY[nHit
sPerEvent]/D:localPosInZ[nHitsPerEvent]/D:localPosOutX[nHitsPerEvent]/D
:localPosOutY[nHitsPerEvent]/D:localPosOutZ[nHitsPerEvent]/D:PhysicsPro
cess[nHitsPerEvent]/D");
  b->GetLeaf("hitID")->SetAddress(&hitID);
  b->GetLeaf("trackID")->SetAddress(&trackID);
  b->GetLeaf("parentID")->SetAddress(&parentID);
  b->GetLeaf("geantID")->SetAddress(&geantID);
  b->GetLeaf("particle_energy")->SetAddress(&particle_energy);
  b->GetLeaf("dE_tpc")->SetAddress(&dE_tpc);
  b->GetLeaf("hit_time")->SetAddress(&hit_time);
  b->GetLeaf("hitPositionX")->SetAddress(&hitPositionX);
  b->GetLeaf("hitPositionY")->SetAddress(&hitPositionY);
  b->GetLeaf("hitPositionZ")->SetAddress(&hitPositionZ);
  b->GetLeaf("trkMomentumX")->SetAddress(&trkMomentumX);
  b->GetLeaf("trkMomentumY")->SetAddress(&trkMomentumY);
  b->GetLeaf("trkMomentumZ")->SetAddress(&trkMomentumZ);
  b->GetLeaf("trkLength")->SetAddress(&trkLength);
  b->GetLeaf("array_position_x")->SetAddress(&array_position_x);
  b->GetLeaf("array_position_y")->SetAddress(&array_position_y);
  b->GetLeaf("planeID")->SetAddress(&planeID);
  b->GetLeaf("localPosInX")->SetAddress(&localPosInX);
  b->GetLeaf("localPosInY")->SetAddress(&localPosInY);
  b->GetLeaf("localPosInZ")->SetAddress(&localPosInZ);
  b->GetLeaf("localPosOutX")->SetAddress(&localPosOutX);
  b->GetLeaf("localPosOutY")->SetAddress(&localPosOutY);
  b->GetLeaf("localPosOutZ")->SetAddress(&localPosOutZ);
  b->GetLeaf("PhysicsProcess")->SetAddress(&PhysicsProcess);
```

```
  //============
  // Hit Loop
  //============
  cout << "Filling hit branch.... "<<endl;
  while (!inFile.eof())
    {
      //if (nEvents>0) break;
      if (nEntries%1000==0) cout << "***** Event "<<nEvents<<" *****
Reading Hits " << nEntries << "*******"<<endl;

      //fill tree from GEANT4 output file
      inFile >> tmp[0] >> tmp[1] >> tmp[2] >> particleName >> tmp[3] >>
tmp[4] >> tmp[5] >> tmp[6] >> tmp[7] >> tmp[8] >> tmp[9] >> tmp[10] >>
      tmp[11] >> tmp[12] >> tmp[13] >> tmp[14] >> tmp[15] >> tmp[16] >>
tmp[17] >> tmp[18] >> tmp[19] >> tmp[20] >> tmp[21] >> tmp[22] >>
      tmp[23];
      //cout<<tmp[0]<<", "<<tmp[1]<<", "<<tmp[2]<<", "<<tmp[3]<<",
"<<tmp[4]<<", "<<tmp[5]<<", "<<tmp[6]<<", "<<tmp[7]<<", "<<tmp[8]<<",
"<<tmp[9]<<endl;

      if


(tmp[0]==-1&&tmp[1]==-1&&tmp[2]==-1&&tmp[3]==-1&&tmp[4]==-1&&tmp[5]==-
1&&tmp[6]==-1&&tmp[7]==-1&&tmp[8]==-1&&tmp[9]==-1&&tmp[10]==-
1&&tmp[11]==-1&&tmp[12]==-1&&tmp[13]==-1&&tmp[14]==-1&&tmp[15]==-
1&&tmp[16]==-1&&tmp[17]==-1&&tmp[18]==-1&&tmp[19]==-1&&tmp[20]==-
1&&tmp[21]==-1&&tmp[22]==-1&&tmp[23]==-1&&ihit!=0)
{
      //end of event => time to fill the tree
      nEvents++;
      nHitsPerEvent = ihit;
      ihit=0;
      t->Fill();
      continue;
      //break;
      }
      if (tmp[0]<0) {
      cout<<"empty event or end of geant hits file"<<endl;
      nEvents++;
      ihit=0;
      continue;
      }


      evt[ihit] = nEvents;
      hitID[ihit] = tmp[0];
      trackID[ihit] = tmp[1];
      parentID[ihit] = tmp[2];
      geantID[ihit] = tmp[3];
      particle_energy[ihit] = tmp[4];
      dE_tpc[ihit] = tmp[5];
      hit_time[ihit] = tmp[6];
      hitPositionX[ihit] = tmp[7];
      hitPositionY[ihit] = tmp[8];
```

```
      hitPositionZ[ihit] = tmp[9];
      trkMomentumX[ihit] = tmp[10];
      trkMomentumY[ihit] = tmp[11];
      trkMomentumZ[ihit] = tmp[12];
      trkLength[ihit] = tmp[13];
      array_position_x[ihit] = tmp[14];
      array_position_y[ihit] = tmp[15];
      planeID[ihit] = tmp[16];
      localPosInX[ihit] = tmp[17];
      localPosInY[ihit] = tmp[18];
      localPosInZ[ihit] = tmp[19];
      localPosOutX[ihit] = tmp[20];
      localPosOutY[ihit] = tmp[21];
      localPosOutZ[ihit] = tmp[22];
      PhysicsProcess[ihit] = tmp[23];



      ihit_prev = ihit;
      IsFirstEvent = kFALSE;
      nEntries++;
      ihit++;
      //if (ihit>10) break;
    }

  cout<<"Total events processed: "<<nEvents-1<<endl;
  t->Write();
  inFile.close();
  f->Close();


}
```

C.18. ProcessAllPairCompton.C

```
#include <string>
#include <fstream>
#include "Riostream.h"
#include "math.h"

void ProcessAllPairCompton(const char
*inputFile="thesis_hit_45_try2.root")
{


  gROOT->Reset();
  gStyle->SetOptStat(kFALSE);
  gStyle->SetOptTitle(kTRUE);
  gStyle->SetLabelSize(0.04,"X");
  gStyle->SetTitleSize(0.05,"X");
  gStyle->SetLabelSize(0.04,"Y");
  gStyle->SetTitleSize(0.05,"Y");
  gStyle->SetPadTopMargin(0.05);
  gStyle->SetPadLeftMargin(0.15);
```

```
gStyle->SetPadRightMargin(0.05);
gStyle->SetPadBottomMargin(0.15);


//=======
// Files
//=======
string inputFileName(inputFile);
ifstream inFile(inputFile);
if (!inFile)    {
  cout<<"failed to find inFile......."<<endl;
  break;
}
else   TFile *fin = new TFile(inputFileName.c_str(), "read");


//=============
// Definitions
//=============
const Int_t nhit = 3000000;
Int_t evt[nhit] = {0};
Int_t hitID[nhit] = {0};
Int_t trackID[nhit] = 0;
Int_t parentID[nhit] = 0;
Char_t particleName[10];
Double_t geantID[nhit] = {0.0};
Double_t particle_energy[nhit] = {0.0};
Double_t dE_tpc[nhit] = {0.0};
Double_t hit_time[nhit] = {0.0};
Double_t hitPositionX[nhit] = {0.0};
Double_t hitPositionY[nhit] = {0.0};
Double_t hitPositionZ[nhit] = {0.0};
Double_t trkMomentumX[nhit] = {0.0};
Double_t trkMomentumY[nhit] = {0.0};
Double_t trkMomentumZ[nhit] = {0.0};
Double_t trkLength[nhit] = {0.0};
Double_t array_position_x[nhit] = {0.0};
Double_t array_position_y[nhit] = {0.0};
Double_t planeID[nhit] = {0.0};
Double_t localPosInX[nhit] = {0.0};
Double_t localPosInY[nhit] = {0.0};
Double_t localPosInZ[nhit] = {0.0};
Double_t localPosOutX[nhit] = {0.0};
Double_t localPosOutY[nhit] = {0.0};
Double_t localPosOutZ[nhit] = {0.0};
Double_t PhysicsProcess[nhit] = {0.0};
Double_t primary_energy[nhit] = {0.0};
Double_t trkarray[nhit] = {0.0};
Double_t array511[nhit] = {0.0};
Double_t dep_primary_en[nhit] = {0.0};
Int_t trkcounter[nhit] = {0};
Bool_t alreadyfilled = kFALSE;
Double_t dep_gamma_en[nhit] = {0.0};
Double_t dep_511_prod[nhit] = {0.0};
Int_t nHitsPerEvent = 0;
```

```
Float_t histedenergy = 0.;
Int_t incident_det = 0;
Int_t detnum = 0;
Int_t inc_tracker = 0;
Int_t elec_track = 0;
Double_t dE_incgamma[nhit] = {0.0};
Double_t total_gam_dep[nhit] = {0.0};
Int_t Num_1173 = 0;
Int_t Num_1332 = 0;
Bool_t  already_subtracted = kFALSE;
Int_t eBrem_tracker = 0;

//===================
// Initialize Arrays
//===================
for (Int_t i=0; i<nhit; i++)
  {
      evt[i] = 0.0;
    hitID[i] = 0.0;
    trackID[i] = 0;
    parentID[i] = 0;
    geantID[i] = 0.0;
    particle_energy[i] = 0.0;
    dE_tpc[i] = 0.0;
    hit_time[i] = 0.0;
    hitPositionX[i] = 0.0;
    hitPositionY[i] = 0.0;
    hitPositionZ[i] = 0.0;
    trkMomentumX[i] = 0.0;
    trkMomentumY[i] = 0.0;
    trkMomentumZ[i] = 0.0;
    trkLength[i] = 0.0;
    array_position_x[i] = 0.0;
    array_position_y[i] = 0.0;
    planeID[i] = 0.0;
    localPosInX[i] = 0.0;
    localPosInY[i] = 0.0;
    localPosInZ[i] = 0.0;
    localPosOutX[i] = 0.0;
    localPosOutY[i] = 0.0;
    localPosOutZ[i] = 0.0;
    PhysicsProcess[i] = 0.0;
    primary_energy[i] = 0.0;
    trkcounter[i] = 0;
    trkarray[i] = 0;
    array511[i] = 0.0;
    dep_primary_en[i] = 0.0;
    dep_gamma_en[i] = 0.0;
    dep_511_prod[i] = 0.0;
    detnum[i] = 0;
    incident_det[i] = 0;
    inc_tracker[i] = 0;
    dE_incgamma[i] = 0.0;
    elec_track [i] = 0;
    total_gam_dep[i] = 0.0;
```

```
    Num_1173[i] = 0;
    Num_1332[i] = 0;
  }


//===========
//Input Tree
//===========
TTree *tin = (TTree*) fin->Get("Hits"); assert(tin);
TLeaf *lnHitsPerEvent = tin->GetLeaf("nHitsPerEvent");
TLeaf *levt = tin->GetLeaf("evt");
TLeaf *lhitID = tin->GetLeaf("hitID");
TLeaf *ltrackID = tin->GetLeaf("trackID");
TLeaf *lparentID = tin->GetLeaf("parentID");
TLeaf *lgeantID = tin->GetLeaf("geantID");
TLeaf *lparticle_energy = tin->GetLeaf("particle_energy");
TLeaf *ldE_tpc = tin->GetLeaf("dE_tpc");
TLeaf *lhit_time = tin->GetLeaf("hit_time");
TLeaf *lhitPositionX = tin->GetLeaf("hitPositionX");
TLeaf *lhitPositionY = tin->GetLeaf("hitPositionY");
TLeaf *lhitPositionZ = tin->GetLeaf("hitPositionZ");
TLeaf *ltrkMomentumX = tin->GetLeaf("trkMomentumX");
TLeaf *ltrkMomentumY = tin->GetLeaf("trkMomentumY");
TLeaf *ltrkMomentumZ = tin->GetLeaf("trkMomentumZ");
TLeaf *ltrkLength = tin->GetLeaf("trkLength");
TLeaf *larray_position_x = tin->GetLeaf("array_position_x");
TLeaf *larray_position_y = tin->GetLeaf("array_position_y");
TLeaf *lplaneID = tin->GetLeaf("planeID");
TLeaf *llocalPosInX = tin->GetLeaf("localPosInX");
TLeaf *llocalPosInY = tin->GetLeaf("localPosInY");
TLeaf *llocalPosInZ = tin->GetLeaf("localPosInZ");
TLeaf *llocalPosOutX = tin->GetLeaf("localPosOutX");
TLeaf *llocalPosOutY = tin->GetLeaf("localPosOutY");
TLeaf *llocalPosOutZ = tin->GetLeaf("localPosOutZ");
TLeaf *lPhysicsProcess = tin->GetLeaf("PhysicsProcess");

lnHitsPerEvent->SetAddress(&nHitsPerEvent);
levt->SetAddress(&evt);
lhitID->SetAddress(&hitID);
ltrackID->SetAddress(&trackID);
lparentID->SetAddress(&parentID);
lgeantID->SetAddress(&geantID);
lparticle_energy->SetAddress(&particle_energy);
ldE_tpc->SetAddress(&dE_tpc);
lhit_time->SetAddress(&hit_time);
lhitPositionX->SetAddress(&hitPositionX);
lhitPositionY->SetAddress(&hitPositionY);
lhitPositionZ->SetAddress(&hitPositionZ);
ltrkMomentumX->SetAddress(&trkMomentumX);
ltrkMomentumY->SetAddress(&trkMomentumY);
ltrkMomentumZ->SetAddress(&trkMomentumZ);
ltrkLength->SetAddress(&trkLength);
larray_position_x->SetAddress(&array_position_x);
larray_position_y->SetAddress(&array_position_y);
lplaneID->SetAddress(&planeID);
```

```
  llocalPosInX->SetAddress(&localPosInX);
  llocalPosInY->SetAddress(&localPosInY);
  llocalPosInZ->SetAddress(&localPosInZ);
  llocalPosOutX->SetAddress(&localPosOutX);
  llocalPosOutY->SetAddress(&localPosOutY);
  llocalPosOutZ->SetAddress(&localPosOutZ);
  lPhysicsProcess->SetAddress(&PhysicsProcess);


  //====================
  // Output Histograms
  //====================
  //gammas
  TH1F* hgamma=new TH1F("hgamma","all gammas",1024,0,2.5);
  hgamma->GetXaxis()->SetTitle("Energy (MeV)");
  hgamma->GetYaxis()->SetTitle("Counts");

  TH1F* hgamma_zoom=new TH1F("hgamma_zoom","all gammas",200,0,1.0);
  hgamma_zoom->GetXaxis()->SetTitle("Energy (MeV)");
  hgamma_zoom->GetYaxis()->SetTitle("Counts");

  //pair_produced
  TH1F* pair_produced=new TH1F("pair_produced","(90 deg w/11 cos)
Deposited Pair Production Events in 4 Detectors from 10E+6 Particles
(1173-, 1332-)",1024,0,2);
  pair_produced->GetXaxis()->SetTitle("Energy (MeV)");
  pair_produced->GetYaxis()->SetTitle("Number of Counts");

  //electrons
  TH1F* helectron=new TH1F("helectron","all electrons",200,0,1.5);
  helectron->GetXaxis()->SetTitle("Energy (MeV)");
  helectron->GetYaxis()->SetTitle("Counts");

  TH1F* helectron_zoom=new TH1F("helectron_zoom","all
electrons",300,0,1);
  helectron_zoom->GetXaxis()->SetTitle("Energy (MeV)");
  helectron_zoom->GetYaxis()->SetTitle("Counts");

  //===========
  //Event Loop
  //===========
  Double_t prev_hit = -1;
  Double_t prev_evt = -1;
  Int_t nEntries = tin->GetEntries();
  cout<<"starting event loop, nEntries= "<<nEntries<<endl;
  for (Int_t ientry=0; ientry<nEntries; ientry++) {
       //if(ientry>90) break;
    if (ientry%3000000==0) cout << "***** Reading Particle " << ientry
<< "*******"<<endl;
    prev_evt = evt[0];
    tin->GetEntry(ientry);
    alreadyfilled = kFALSE;
    already_subtracted = kFALSE;
    eBrem_tracker = 0;
    //Code to extract Pair-Production events and fill a histogram
```

```
        for (Int_t i=0; i<nhit; i++)
        {
              trkcounter[i] = 0;}

      for (Int_t nhit=0; nhit<nHitsPerEvent; nhit++){
      if(hitPositionX[nhit]>0.539 && hitPositionX[nhit]<2.541 &&
hitPositionY[nhit]>0.539 && hitPositionY[nhit]<2.541 &&
hitPositionZ[nhit]> -0.001 && hitPositionZ[nhit]<5.001){
            detnum[nhit] = 1;}
      if(hitPositionX[nhit]<-0.539 && hitPositionX[nhit]>-2.541 &&
hitPositionY[nhit]>0.539 && hitPositionY[nhit]<2.541 &&
hitPositionZ[nhit]> -0.001 && hitPositionZ[nhit]<5.001){
            detnum[nhit] = 2;}
      if(hitPositionX[nhit]<-0.539 && hitPositionX[nhit]>-2.541 &&
hitPositionY[nhit]<-0.539 && hitPositionY[nhit]>-2.541 &&
hitPositionZ[nhit]> -0.001 && hitPositionZ[nhit]<5.001){
            detnum[nhit] = 3;}
      if(hitPositionX[nhit]>0.539 && hitPositionX[nhit]<2.541 &&
hitPositionY[nhit]<-0.539 && hitPositionY[nhit]>-2.541 &&
hitPositionZ[nhit]> -0.001 && hitPositionZ[nhit]<5.001){
            detnum[nhit] = 4;}
      if(geantID[nhit]==22 && (dE_tpc[nhit]>-.485 || dE_tpc[nhit]<-
.537)){
            incident_det[ientry] = detnum[nhit];
            inc_tracker[ientry] = trackID[nhit];
            dE_incgamma[ientry] = dE_tpc[nhit];
            break;}
      else{
            incident_det[ientry] = 0;
            inc_tracker[ientry] = -1;
            break;}
      }
      for (Int_t mhit=0; mhit<nHitsPerEvent; mhit++){
            if(geantID[mhit] == 11 && particle_energy[mhit] == -
dE_incgamma[ientry]){
            elec_track[ientry] = trackID[mhit];}}
      if(elec_track[ientry] == 0){
            elec_track[ientry] = 2;}

      for (Int_t jhit=0; jhit<nHitsPerEvent; jhit++){
      if(hitPositionX[jhit]>0.539 && hitPositionX[jhit]<2.541 &&
hitPositionY[jhit]>0.539 && hitPositionY[jhit]<2.541 &&
hitPositionZ[jhit]> -0.001 && hitPositionZ[jhit]<5.001){
            detnum[jhit] = 1;}
      if(hitPositionX[jhit]<-0.539 && hitPositionX[jhit]>-2.541 &&
hitPositionY[jhit]>0.539 && hitPositionY[jhit]<2.541 &&
hitPositionZ[jhit]> -0.001 && hitPositionZ[jhit]<5.001){
            detnum[jhit] = 2;}
      if(hitPositionX[jhit]<-0.539 && hitPositionX[jhit]>-2.541 &&
hitPositionY[jhit]<-0.539 && hitPositionY[jhit]>-2.541 &&
hitPositionZ[jhit]> -0.001 && hitPositionZ[jhit]<5.001){
            detnum[jhit] = 3;}
      if(hitPositionX[jhit]>0.539 && hitPositionX[jhit]<2.541 &&
hitPositionY[jhit]<-0.539 && hitPositionY[jhit]>-2.541 &&
hitPositionZ[jhit]> -0.001 && hitPositionZ[jhit]<5.001){
```

```
            detnum[jhit] = 4;}
      if(geantID[jhit]==11 && parentID[jhit]==inc_tracker[ientry] &&
detnum[jhit]==incident_det[ientry] && trackID[jhit] ==
elec_track[ientry]){
                  dep_gamma_en[ientry]+=-dE_tpc[jhit];}
            if(geantID[jhit]==22 && (dE_tpc[jhit]<-.485 &&
dE_tpc[jhit]>-.537)){
                  trkcounter[jhit]=trackID[jhit];}
            else{
                  trkcounter[jhit] = -1;}}
      for (Int_t qhit=0; qhit<nHitsPerEvent; qhit++){
            if(geantID[qhit] == 22 && PhysicsProcess[qhit] == 5 &&
parentID[qhit] == elec_track[ientry] && eBrem_tracker !=
trackID[qhit]){
            eBrem_tracker = trackID[qhit];
            dep_gamma_en[ientry] = dep_gamma_en[ientry] -
particle_energy[qhit];}}
      for (Int_t ihit=0; ihit<nHitsPerEvent; ihit++){
            if(alreadyfilled == kTRUE){
                  break;}
            if(geantID[ihit]==11){
                  for(Int_t khit=0; khit<ihit; khit++){
                        if(parentID[ihit]==trkcounter[khit] &&
parentID[ihit]!=inc_tracker[ientry] && trackID[ihit] !=
elec_track[ientry]){
                        dep_511_prod[ientry] +=- dE_tpc[ihit];
                        }}}
            if(dep_511_prod[ientry] >.485 && dep_511_prod[ientry] <.537
&& dep_gamma_en[ientry]>0){
                  histedenergy = dep_gamma_en[ientry]+1.022;
                  cout<<" The Histed Energy is "<<histedenergy<<endl;
                  pair_produced->Fill(histedenergy);
                  if(histedenergy>(1.173-1.173*.03) &&
histedenergy<(1.173+1.173*.03)){
                        Num_1173 = Num_1173 + 1;}
                  if(histedenergy>(1.332-1.332*.03) &&
histedenergy<(1.332+1.332*.03)){
                        Num_1332 = Num_1332 + 1;}
                  already filled = kTRUE;
                  break;}
            }
    for (Int_t jhit=0; jhit<nHitsPerEvent; jhit++) {
      if ((hitID[jhit]!=prev_hit || evt[jhit]!=prev_evt)) {
      if(particle_energy[jhit]!=0) {
        if (geantID[jhit]==22) hgamma->Fill(particle_energy[jhit]);
        if (geantID[jhit]==22) hgamma_zoom-
>Fill(particle_energy[jhit]);
        if (geantID[jhit]==11) helectron->Fill(particle_energy[jhit]);
        if (geantID[jhit]==11) helectron_zoom-
>Fill(particle_energy[jhit]);
        if (parentID[jhit]==1) {
        }
        else {
        }
      }
```

```
      prev_hit = hitID[jhit];
      prev_evt = evt[jhit];
      }
    }
    //if(histedenergy>1.9){
    // cout << "***** Reading Particle " << ientry << "*******"<<endl;}
  }
  cout<<"Number of 1173 in histogram is "<<Num_1173<<endl;
  cout<<"Number of 1332 in histogram is "<<Num_1332<<endl;

  //=================
  // Output Canvases
  //=================
  //create canvases to contain the histogram

  //gammas
  TCanvas *canvas0 = new TCanvas("canvas0","AllGammas",400,400);
  canvas0->SetLogy();
  canvas0->SetTicks(1,1);
  canvas0->Draw();
  hgamma->Draw();

  TCanvas *canvas1 = new TCanvas("canvas1","AllGammas",400,400);
  canvas1->SetLogy();
  canvas1->SetTicks(1,1);
  canvas1->Draw();
  hgamma_zoom->Draw();

  TCanvas *canvas2 = new TCanvas("canvas2","Deposited Pair Production
Events with at least 1 511 Capture",400,400);
  canvas2->SetLogy();
  canvas2->SetTicks(1,1);
  canvas2->Draw();
  pair_produced->Draw();

  TCanvas *canvas3 = new TCanvas("canvas3","AllElectrons",400,400);
  canvas3->SetLogy();
  canvas3->SetTicks(1,1);
  canvas3->Draw();
  helectron->Draw();

  TCanvas *canvas4 = new TCanvas("canvas4","AllElectrons",400,400);
  canvas4->SetLogy();
  //canvas4->SetLogx();
  canvas4->SetTicks(1,1);
  canvas4->Draw();
  helectron_zoom->Draw();

  canvas0->Update();
  canvas1->Update();
  canvas2->Update();
  canvas3->Update();
  canvas4->Update();
}
```

Appendix D. Radiation Sources Used

Source ID Number	Isotope	Initial Activity (mCi)	Date of Initial Activity
T-113	Co-57	0.05271	1-Jan-06
T-133	Sr-85	0.09679	15-Aug-09
T-089	Cs-137	0.0102	1-Aug-98
T-125	Co-60	0.01026	15-Sep-07
00097	Co-60	24500	3-Nov-54

Bibliography

[1] Marion L. Stelts and John C. Browne, "A Neutron Capture gamma-Ray Spectrometer for use at an Electron LINAC," *Nuclear Instruments and Methods*, vol. 133, pp. 35-49, November 1976.

[2] A. Robertson, G. C. Cormick, T. J. Kennett, and W. V. Prestwich, "Continuum Reduction in the Spectral Response of Ge(Li) Pair Spectrometers," *Nuclear Instruments and Methods*, vol. 127, pp. 373 - 379, May 1975.

[3] J.H. Winso, E.S. Ackermann, M. Fennell, R. Perez, and J. Rolando, "Geometrically Optimized, LaBr3: Ce Scintillation SensorArray for Enhanced Stand-Off Direction Finding of Gamma Radiation Sources," in *Nuclear Science Symposium Conference Record, 2007. NSS '07. IEEE*, vol. 2, Honolulu, HI, 2007, pp. 1009-1015.

[4] Glenn F Knoll, *Radiation Detection and Measurement*, 3rd ed. New Jersey: John Wiley & Sons, 2000.

[5] M.J. Berger et al. (2010, April) The National Institue of Standards and Technolgoy (NIST). [Online]. http://www.nist.gov/physlab/data/xcom/index.cfm

[6] Brian L. Evans, "A Portable Compton Gamma-Ray Camera Design," WPAFB, OH, MS Thesis 1995.

[7] J. van der marel and B. Cederwall, "Backtracking as a way to reconstruct Compton scattered gamma-rays," *Nuclear Instruments and Methods in Physics Research A*, vol. 437, pp. 538-551, August 1999.

[8] Gary W. Phillips, "Gamma-ray imaging with Compton cameras," *Nuclear Instruments and Methods in Physics Research B*, vol. 99, pp. 674 - 677, 1995.

[9] Saint-Gobain. (2009) BrilLanCe Scintillators Performance Summary. [Online]. http://www.detectors.saint-gobain.com/uploadedFiles/SGdetectors/Documents/Technical_Information_Notes/BrilLanCe-Scintillators-Performance-Summary.pdf

[10] E.B. Shera, K. J. Casper, and B.L. Robinson, "Analysis of Chance Coincidences in Fast-Slow Coincidence Systems," *Nuclear Instruments and Methods*, vol. 24, pp. 482-492, April 1963.

[11] James E. Turner, *Atoms, Radiation, and Radiation Protection*, 3rd ed. Weinheim, Germany: Wiley-VCH, 2007.

[12] Ronald E. Pevey. (1998) NE406 Radiation Protection and Shielding. [Online]. http://web.utk.edu/~rpevey/NE406/lesson2.htm

[13] Michael J. Cree and Phillip J. Bones, "Towards Direct Reconstruction from a Gamma Camera Based on Compton Scattering," *IEEE Transactions on Medical Imaging*, vol. 13, no. 2, pp. 398-407, June 1994.

[14] Nuclear and Plasma Sciences Society, "325-1986 IEEE Standard Test Procedures for Germanium Gamma-Ray Detectors," IEEE, 325-1986, 1987.

[15] Agnieszka Syntfeld et al., "Comparison of a LaBr3(Ce) Scintillation Detector With a Large Volume CdZnTe Detector," *IEEE Transactions on Nuclear Science*, vol. 53, no. 6, pp. 3938-3943, December 2006.

[16] Clifford Sulham, *Special Nuclear Material Imaging Using a High Purity Germanium Double Sided Strip Detector*. Dayton, United States of America: Air University, 2004.

[17] (2010, November) Geant4. [Online]. http://geant4.web.cern.ch/geant4/

[18] (2010) ROOT. [Online]. http://root.cern.ch/drupal/

[19] LLC XIA, User's Manual: Digital Gamma Finder (DGF) DGF-4C Revision F, July 2009, Version 4.03.

[20] MathWorks. Sgolayfilt. [Online]. http://www.mathworks.com/help/toolbox/signal/sgolayfilt.html

[21] C Ferguson. (2000, February) Geant4 General Particle Source Users Manual. [Online]. http://reat.space.qinetiq.com/gps/new_gps_sum_files/gps_sum.htm

[22] J. K. Tuli. (2003) National Nuclear Data Center- Chart of the Nuclides. [Online]. http://www.nndc.bnl.gov/chart/decaysearchdirect.jsp?nuc=60CO&unc=nds

[23] (2010, January) Tektronix. [Online]. http://www2.tek.com/cmswpt/faqdetails.lotr?ct=FAQ&cs=faq&ci=6253&lc=EN

[24] Jesse Foster, "Digital Signal Processing with Tektronix Oscilloscope and Review of FGPA Application in Nuclear Detection," Air Force Institute of Technology, Wright-Patterson AFB, OH, 2010.

[25] (2007, May) Tektronix. [Online]. http://www2.tek.com/cmswpt/faqdetails.lotr?ct=FAQ&cs=faq&ci=6335&lc=EN

Vita

1st Lieutenant William L. Harrell was born on 30 July 1985 at Brandon City Hospital in Brandon, Florida. In 2003 he graduated from Lake Mary High School in Lake Mary, Florida. He entered undergraduate studies at The Citadel in Charleston, South Carolina, and graduated in May 2007 with a Bachelor of Science degree in Physics. He received his Air Force commission on 4 May 2007. His first assignment was to the Air Force Office of Scientific Research in Arlington, Virginia where he worked as an Assistant Program Manager in the Physics and Electronics Directorate. In August, 2009 he entered the School of Engineering, Air Force Institute of Technology.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
24-03-2011	Master's Thesis	August 2009 – March 2010

4. TITLE AND SUBTITLE

Directional Pair-Production Spectrometer Design for Airborne Stand-Off Detection of Special Nuclear Material

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Harrell, William L., First Lieutenant, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GNE/ENP/11-M09

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Mr. William Ulicny
DHS/DNDO
245 Murray Dr. B410
Washington, DC 20528

10. SPONSOR/MONITOR'S ACRONYM(S)

DHS/DNDO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The purposes of this research are to experimentally and theoretically prove the concept of a directional pair-production spectrometer to detect and locate the tailings that are created when making Special Nuclear Material (SNM) at stand-off distances from a remotely piloted vehicle (RPV). A directional pair-production spectrometer uses the information garnered from the high energy gamma rays emitted by these SNM manufacturing tailings to perform pair-production spectroscopy and identify the isotope of interest. Through simultaneous operation as a Compton camera, the detection system will be able to measure rudimentary directional information from the medium energy gamma decays. The detector used for this research is constructed of four $LaBr_3$ detectors and operated in coincidence to allow for the reduction of background. The directional efficiency of the detector in measuring the radioactive decay of a 6.7 μCi Co-60 source is validated with a Geant4 simulation. The simulation is used to predict the directional efficiency of a detector system using six detector elements and the pair-production spectrum that would be seen when measuring a higher energy gamma ray source.

15. SUBJECT TERMS

Nuclear, Detection, Special Nuclear Material, Detector, Gamma Ray, Pair-Production, Compton Camera, Spectrometer, Spectroscopy

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Maj Benjamin R. Kowash AFIT/ENP
U	U	U	UU	199	19b. TELEPHONE NUMBER *(Include area code)* 937-785-3636 x4571 email: benjamin.kowash@afit.edu

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18