

CONVERSION ELECTRONS IN $^{154,156}\text{GD}$

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Doctorate of Philosophy
in
Physics

by
Sabrina Yve Strauss

Ani Aprahamian, Director

Graduate Program in Physics

Notre Dame, Indiana

December 2019

© Copyright by

Sabrina Strauss

2019

CC-BY-4.0

CONVERSION ELECTRONS IN $^{154,156}\text{GD}$

Abstract

by

Sabrina Yve Strauss

One of the open questions of nuclear structure is the nature of excited 0^+ states. Probing the 0^+ states in nuclei requires spectroscopy of the 0^+ states with respect to other states. One such probe is conversion electrons. The transitions between two 0^+ states can only be seen via E0 transitions, which are forbidden via γ -rays. Further, E0 components of $J^\pi \rightarrow J^\pi$ transitions can give similar probes of relationships between bands. Information on E0 components are challenging to measure and sparse in nuclear databases.

The rare-earth region provides a rich landscape to probe pure and mixed E0 transitions. The deformed nature of nuclei in the area leads to a large number of nuclear structure phenomena. In ^{154}Gd , 16 0^+ states have been seen in previous studies, the nature of many are unknown. The first excited 0^+ state is of great interest, due to its large E0 strength to the ground state. There is a question of the nature of this state and its potential role as an example of β -vibration, or an indication of shape coexistence. ^{156}Gd is one of the most well-studied nuclei in the rare-earth region. It has a large number of known levels, lifetimes and mixing ratios, allowing for the study of mixed E0 transitions.

This work reports on results for transitions in $^{154,156}\text{Gd}$ following the $^{152,154}\text{Sm}(\alpha,2n)$ reaction using the Internal Conversion Electron Ball (ICEBall) array in coincidence with γ -rays at the University of Notre Dame Nuclear Science Laboratory (NSL).

ICEBall was re-implemented at the NSL 8 years ago and the γ -rays were detected by GEORGINA, a HPGe array at the NSL, and Clovershare, segmented HPGe detectors purchased by the Yale Nuclear Structure Laboratory that are shared by a consortium of universities and laboratories for experimental campaigns.

To my parents, who always encouraged my love of science.

To Jolie, who introduced me to nuclear physics research.

To Ani, who helped keep me from going too far down side paths.

To all the people who put up with my mood swings.

CONTENTS

Figures	vii
Tables	xiii
Chapter 1: Introduction	1
1.1 Nuclear Excitations	10
1.1.1 Collective Excitations	12
1.1.2 Shape Coexistence	12
1.2 Multipole Radiation	15
1.3 Internal Conversion	21
1.3.1 E0 Transitions	26
1.4 The Rare Earth Region: $N = 90, Z = 60$	29
Chapter 2: Experimental Setup	31
2.1 Overview	31
2.2 Nuclear Science Laboratory at Notre Dame	34
2.3 Beam Production	36
2.3.1 Talys Calculations	37
2.3.2 Test Analysis and Results	38
2.4 Electron Detection	43
2.4.1 Internal Conversion Electron Ball Spectrometer	44
2.4.1.1 Mini-Orange Spectrometer	45
2.4.2 Calibration	50
2.5 Gamma Detection	51
2.5.1 GEORGINA	53
2.5.2 CloverShare	54
2.5.3 Calibration	54
2.6 Target Fabrication	58
2.7 Experimental Configuration and Operation	61
2.7.1 Experimental Beam Operation	61
2.7.2 GEORGINA Configuration and Electronics	61
2.7.3 Clovershare Configuration and Electronics	65

Chapter 3: Analysis	73
3.1 Fitting Spectra	73
3.2 Gating	78
3.3 Systematic Effects	81
3.3.1 Electronics-based Integral Non-Linearities	84
3.3.2 Run-based Calibration Corrections	85
3.3.3 Angular Correlations	85
3.4 Upper Limits	89
3.5 Error Analysis	91
 Chapter 4: ^{154}Gd Results	93
4.1 Ground State Band Confirmation	95
4.2 Gates on the Ground State Band	100
4.3 Conversion Coefficients from Singles	102
4.4 $J^\pi \rightarrow J^\pi$ Transitions	115
4.4.1 $0^+ \rightarrow 0^+$ Identification and Population	115
4.4.2 $J^\pi \rightarrow J^\pi$ Transitions	115
4.4.2.1 $K^\pi = 0_2^+$, First excited 0^+ band, 680.6673 keV	125
4.4.2.2 $K^\pi = 0_3^+$, Second excited 0^+ band, 1182.091 keV	130
4.4.2.3 $K^\pi = 0_6^+$, Fifth excited 0^+ band, 1573.9 keV	131
4.4.2.4 $K^\pi = 0_7^+$, Sixth excited 0^+ band, 1650.3 keV	132
4.4.2.5 $K^\pi = 2^+$, γ band, 996.264 keV	133
4.4.2.6 $K^\pi = 2^+$, Second excited 2^+ band, 1531.305 keV	133
4.4.2.7 $K^\pi = 4^+$, First excited 4^+ band, 1645.814 keV	134
 Chapter 5: ^{156}Gd Results	136
5.1 Ground State Band Confirmation	136
5.2 Gates on the Ground State Band	137
5.3 Conversion Coefficients from Singles	149
5.4 $J^\pi \rightarrow J^\pi$ Transitions	150
5.4.1 $0^+ \rightarrow 0^+$ Identification and Population	150
5.4.2 $J^\pi \rightarrow J^\pi$ Transitions	157
5.4.2.1 $K^\pi = 0_3^+$, Second excited 0^+ band, 1168.186 keV	161
5.4.2.2 $K^\pi = 2^+$, Second excited 2^+ band, 1827.841 keV	161
5.4.2.3 $K^\pi = 4^+$, First excited 4^+ band, 1510.594 keV	164
 Chapter 6: Discussion	165
6.1 Dynamic Moments of Inertia	165
6.2 Bands in ^{154}Gd	165
6.2.1 $K^\pi = 0_2^+$, First excited 0^+ band, 680.6673 keV	169
6.2.2 $K^\pi = 0_3^+$, Second excited 0^+ band, 1182.091 keV	169
6.2.3 $K^\pi = 0_6^+$, Fifth excited 0^+ band, 1573.9 keV	169
6.2.4 $K^\pi = 2^+$, γ band, 996.264 keV	169

6.2.5	$K^\pi = 2^+$, Second excited 2^+ band, 1531.305 keV	170
6.2.6	$K^\pi = 4^+$, First excited 4^+ band, 1645.814 keV	170
6.3	Bands in ^{154}Gd	170
6.3.1	$K^\pi = 0_3^+$, First excited 0^+ band, 1049.487 keV	174
6.3.2	$K^\pi = 0_3^+$, Second excited 0^+ band, 1168.186 keV	174
6.3.3	$K^\pi = 2^+$, γ band, 1065.1781 keV	174
6.3.4	$K^\pi = 2^+$, Second excited 2^+ band, 1827.841 keV	174
6.3.5	$K^\pi = 4^+$, First excited 4^+ band, 1510.594 keV	175
Chapter 7: Future Work		176
7.1	Detector Upgrades	176
7.1.1	fIREBAll	176
7.1.2	Magnet Designs	178
7.2	Target Upgrades	179
7.3	Electronics	186
7.4	Additional Experiments	187
7.5	Outlook	188
Appendix A: ^{154}Gd Gated Spectra		190
Appendix B: ^{156}Gd Gated Spectra		268
Appendix C: Analysis Code		294
C.1	Coincidence Code	295
C.1.1	README.md	295
C.1.2	Makefile	296
C.1.3	Queuescript.sh	297
C.1.4	src/main.cxx	298
C.1.5	src/analysis.cxx	300
C.1.6	src/analysis.h	305
C.1.7	src/histograms.cxx	305
C.1.8	include/histograms.h	310
C.1.9	src/timing.cxx	310
C.1.10	include/timing.h	312
C.2	Merge Code	313
C.2.1	README.md	313
C.2.2	Makefile	314
C.2.3	src/main.cxx	315
C.2.4	src/histograms.cxx	317
C.2.5	include/histograms.h	321
C.3	Code shared between Coincidence and Merge	322
C.3.1	src/Coefficients.cxx	322
C.3.2	include/Coefficients.h	326
C.3.3	src/Constraints.cxx	326

C.3.4	include/Constraints.h	328
C.3.5	src/Filelist.cxx	328
C.3.6	include/Filelist.h	330
C.4	Example User Files	331
C.4.1	user/Filelist.dat	331
C.4.2	user/GeCoefficients.dat	331
C.4.3	user/SiLiCoefficients.dat	333
C.4.4	user/BGO.dat	333
C.4.5	user/Timing.dat	334
C.4.6	user/Cut_Files/GeCuts.dat	335
C.4.7	user/Cut_Files/SiLiCuts.dat	336
C.4.8	user/Run_by_Run/GeCoefficients_r*.dat	336
C.4.9	user/Run_by_Run/SiLiCoefficients_r*.dat	337
Appendix D: ROOT Macro Codes		338
D.1	FitterProgram.cxx	338
D.2	skewedsubtraction.cxx	354
D.3	Piecewise_macro.cxx	356
D.4	ROOT2SPE.cxx	359
Bibliography		364

FIGURES

- 1.1 The chart of nuclides. The x -axis is increasing in neutron number from left to right, and the y -axis is increasing in proton number from bottom to top. The lower leftmost corner is the lightest elements, while the upper right are the heaviest, including superheavy elements. Black boxes indicate stable or extremely long lived nuclei. These nuclei are in the "center" of the chart, and known colloquially as the valley of stability. To either side of these stable nuclei are radioactive nuclei, which become shorter lived the farther they are from the valley, indicated by the lightening color. The black boxes are at neutron and proton numbers 8, 20, 28, 50, 82 and 126, marking closed shells for spherical nuclei, and are discussed in further depth within the text.

1.2 The evolution of the shell model and the creation of the closed shells of a spherical nucleus. On the left is the spherical harmonic oscillator. By changing the potential to the Woods-Saxon potential, creating a flat centered potential well, the degenerate states of the harmonic oscillator separate in energy. By further adding a spin-coupling term, the levels separate with energy spacings that create the closed shells and the magic numbers.

1.3 The energy of the first excited 2^+ state, plotted in isotopic chains by mass, distinguished by color. Areas of high excitation energy, around $A \sim 90, 140, 210$ all correspond to doubly magic areas. The areas around $A \sim 110, 170$ have much lower excitation energies and correspond to areas of deformation.

1.4 The Nilsson diagram, showing the evolution of the shell levels of the nucleus with respect to the deformation parameter β . As the nucleus gets more deformed, the closed shells break down. Taken from [17]. . .

1.5 Illustration of different kinds of deformation, where spherical symmetry is lost. Treating the vertical as the axis of symmetry, the prolate nucleus is contracted along this axis, but is still axially symmetric. The oblate nucleus is elongated along this axis, but otherwise axially symmetric. It is also possible for the nucleus to lose axial symmetry. . .

1.6	Plot of the norm of the deformation parameter β_2 along isotopic chains. Deformation can be seen setting in, as isotopic chains transition from spherical to deformed. β_2 is the quadrupole deformation parameter. The norm of β_2 is used, as the method used to calculate β_2 only gives a magnitude, not a sign. A secondary method is needed to distinguish between prolate and oblate.	9
1.7	Plot of the ratio of the first 4^+ excited state energy to the first 2^+ excited state energy in even-even nuclei. On the right, the ratio is broken down into sections, based on structure. This ratio is an excellent diagnostic of the general nuclear structure.	11
1.8	Cartoon of what excited states built upon different types of nuclei looks like. The closed shell has a large gap between the ground state and first excited state. In the vibrator model, this gap is lower, and a gap of comparable size exists before a cluster of states occurs. In the rotor model, the spacing follows a spacing pattern of $J_f(J_f + 1) - J_i(J_i + 1)$, with γ and β vibrations also coming into play. The transitional area between vibrator and rotor shows a shift between the two, with the clustered states of the vibrator model beginning to space out.	13
1.9	Cartoon of the potential well of a nucleus with shape coexistence. A second minima appears at a different β , the deformation parameter. Both of these minima can have excitations built on top of them, leading to shape coexistence. A representation of the shape at β is shown in each minima.	14
1.10	The theoretical K-shell conversion coefficients for the magnetic and electric multipoles for $l = 1, 2, 3$, calculated using BrICC [32]. Each of the multipoles is distinct. Further distinction between multipoles can be seen by plotting the intensity ratios between different electronic shells.	23
2.1	Image of the assembled experimental configuration with Clovershare. One HPGe detector is not visible from this angle. The red line indicates the beam direction.	32
2.2	Comparison between the cross sections of the (α, xn) reactions on ^{154}Sm , calculated using Talys[36]. The less neutrons being lost, the lower the energy before the reaction occurs. Maximizing the cross section with respect to the neutron flux can be done by finding an energy where the desired reaction, $(\alpha, 2n)$ can be maximized, while minimizing the (α, n) and $(\alpha, 3n)$ reactions. Specifically, this means finding an energy before the $3n$ reaction has significant strength, while maximizing the $2n$ reaction, as the n reaction is nearly the same across the energy regime of interest.	33

2.3	Current layout of the Nuclear Science Laboratory at Notre Dame. The red line indicates the path taken by the beam, from ion source, through the accelerator and analyzing magnet, before being sent to ICEBall.	35
2.4	A schematic of how the Helium Ion Source works.	36
2.5	Talys calculation of the highest cross sections divided by the total neutron yield in the natural samarium target. Only the largest contributors are included, as there are a large number of products possible between the isotopes of samarium in the target and the reactions channels open for each. All of the largest contributors are gadolinium isotopes. The ratios for the isotopes of interest appear to plateau around 20-22 MeV.	38
2.6	A cartoon illustration of pulse shape based on radiation type, in the same detecting material. The different types of radiation interact with the detecting material creating a fast and slow excitation. The ratio of these fast and slow excitations is based on the radiation, and expressed in the length of the decay tail. Picture taken from [34].	39
2.7	Plot of the pulse height vs the decay time in the NE212 detectors. Two clear areas are seen. The leftmost, with the shorter decay time, is the gammas, while the more spread out peak on the right is neutrons.	40
2.8	Cartoon of how the bandgap works for insulators, conductors, and semiconductors. Insulators have a large band gap, conductors overlap, and semiconductors have a small bandgap.	43
2.9	Image of ICEBall, open for servicing. The beam would go from left to right in the picture. The target ladder is blank, in the center of ICEBall.	45
2.10	Graphic of the mini-orange filter. The central blocker keeps γ -rays from hitting the detector. The magnets bend electrons toward the detectors, and positrons away from the detectors. Being permanent magnets, they are optimized for a range of electron energies, and can cause overbending or underbending of electrons outside of that energy range, making the magnetic filter a factor in the efficiency. Taken from [44].	46
2.11	An efficiency comparison of magnetic filters of two different strengths. The weak filter has been weakened by 70% compared to the strong filter, using the Pitt split-pole spectrograph. Data from [43].	48
2.12	Linear and quadratic energy calibrations of one Si(Li) detector, using the electron energies listed in Table 2.7. The top graphs show the calibrations, while the graphs beneath show the respective residuals. The calibration is in much better agreement in the quadratic fit, as seen in the residuals.	52

2.13	An example of one of the GEORGINA detectors. The crystal is at a 90° from the cold finger, allowing the long side of the crystal to be placed next to the target, to optimize solid angle coverage. In the experiment, the circular face of the crystal was placed toward the target.	53
2.14	Example of one of the Clovershare detectors.	55
2.15	A characteristic fit of the Clovershare detector efficiencies. The points with large error bars are the ^{152}Eu points, as the scaling of the points caused a large uncertainty, compared to the ^{207}Bi and ^{133}Ba	58
2.16	Efficiency of a CloverShare detector, with the individual leaves' efficiency summed together, compared to the efficiency of the leaves calibrated and summed together. The different colors/shapes indicate the different sources, as labeled.	59
2.17	A look at the energy calibration of an individual leaf in an HPGe Clover detector. The upper graph is a linear calibration. The lower graph is a look at the residuals. For HPGe detectors, ± 0.4 keV is not a good energy calibration. Here, the sawtooth fit that was derived from the differential non-linearity in the electronics is seen fitted to the data.	60
2.18	This is a plot of the HPGe 1 detector time minus the buncher time. Two distinct peaks can be seen in the spectrum, the main timing peak close to channel 800, and the much smaller secondary peak, coming from a second, later bunching signal. The bunching signal only acts as a stop signal for the timing. It can also be used as a veto, as events without a valid buncher time are not real, or are background.	62
2.19	A schematic of the electronics for the ICEBall-Georgina set up. See section 2.7.2 for a detailed explanation. Taken from [5].	64
2.20	ICEBall modeled in Autodesk Inventor [4] with the Clovershare detectors placed around it in locations to optimize efficiency. Once ICEBall had been modeled, the detectors were moved around to make sure they were not behind the tungsten blockers. The beamline was also modeled to know where obstructions existed that prevent the Clovershare detectors from being place there.	66
2.21	The signal processing done by the XIA Pixie-16 electronics. All the processing is done within the module. Taken from [66].	67
3.1	Timing example for the GEORGINA setup	82
3.2	Timing example for the Clovershare setup	83
3.3	Example of calibration drift by run.	86
3.4	An example of the subtraction used to removed the ground state band peaks from spectra.	90
3.5	An example of the global background fit used for peaks that could not be fit using the skewed gaussian function.	91

4.1	Systematics of the Gadolinium isotopes	94
4.2	Singles spectra of ^{154}Gd	96
4.3	Example of comparing gates for level population confirmation	101
4.4	Level scheme and spectrum of ^{154}Gd based on the $2^+ \rightarrow 0^+$ transition.	103
4.5	Level scheme and spectrum of ^{154}Gd based on the $4^+ \rightarrow 2^+$ transition.	105
4.6	Level scheme and spectrum of ^{154}Gd based on the $6^+ \rightarrow 4^+$ transition.	107
4.7	Level scheme and spectrum of ^{154}Gd based on the $8^+ \rightarrow 6^+$ transition.	109
4.8	Electron singles for ^{154}Gd . The energies of transitions between 0^+ states are marked in the plot.	117
5.1	Singles spectra of ^{156}Gd	138
5.2	Level scheme and spectrum of ^{156}Gd based on the $2^+ \rightarrow 0^+$ transition.	140
5.3	Level scheme and spectrum of ^{156}Gd based on the $4^+ \rightarrow 2^+$ transition.	142
5.4	Level scheme and spectrum of ^{156}Gd based on the $6^+ \rightarrow 4^+$ transition.	144
5.5	Level scheme and spectrum of ^{156}Gd based on the $8^+ \rightarrow 6^+$ transition.	146
5.6	Example of comparing gates for level population confirmation	148
5.7	Electron singles for ^{156}Gd . The energies of transitions between 0^+ states are marked in the plot.	152
6.1	The dynamic moments of inertia of the four 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the first excited 0^+ band have very similar moments of inertia.	166
6.2	The dynamic moments of inertia of the non- 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the γ band have similar moments of inertia. However, they do not overlap within their standard deviations.	167
6.3	The dynamic moments of inertia of the four 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the first excited 0^+ band have very similar moments of inertia.	171
6.4	The dynamic moments of inertia of the non- 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the γ band have similar moments of inertia, and overlap within two standard deviations.	172

7.1	A comparison of the singles spectrum of one Si(Li) detector between the GEORGINA and Clovershare experiments. The resolution of the Si(Li) detector has deteriorated between the two experiments (the GEORGINA one was performed first). The LM peaks at \sim 240 keV and \sim 330 keV show this most clearly. The peaks are distinguishable in the GEORGINA data, but not in the Clovershare data.	177
7.2	Map of the magnetic field lines, created through the use of COMSOL, for the original magnetic design [21].	179
7.3	Initial tests of the cross shapes. The left-side of the shape would be the side against the blocker. Two extremes were chosen, with intermediate steps being tested between the two. These are the three sets of extremes used: large triangle to square, triangle with the base against the blocker to triangle with the point against the blocker, and square to truncated diamond.	180
7.4	Sampling of the different hybrid cross shapes tested while examining magnetic designs. The left side of the shape would be the side against the central blocker.	181
7.5	Various width profiles tested for the magnets. The left figure shows the cross-shape for the width profiles. The left side goes against the central blocker. Variable gradients were looked at, starting with linear slopes, but evolving into step functions and piecewise linear slopes as the simulations expanded.	182
7.6	The final shapes chosen for testing. The left side goes against the central blocker. The top design has the best overall efficiency, the middle design has the best high energy efficiency, and the bottom design has the best low energy efficiency. These designs have been commissioned for testing.	183
7.7	Comparison of the thick (self-supporting) and thin (carbon-backed) targets. The spectra were taken during the same experiment. At low energies, the resolution is better in the thin target. Further, the energies of the peaks shift based on which target it is. This spectrum is not energy calibrated.	184
7.8	Trigger rate of the background with respect to time, after running on one of the carbon-backed targets. There is a clear indication of an exponential decay, shown by the fit. The slope corresponds to a half-life of 133(5) seconds. ^{15}O has a half-life of 122 seconds, within an acceptable error with the low statistics of the run.	185
7.9	A schematic representation of the MDPP-16 software in SCP mode. The amplitude and timing are separateing out into individual signals. Signal properties, shaping time, and the threshold can all be adjusted. Taken from [41].	187

TABLES

2.1	Isotope Distribution of Natural Samarium	37
2.2	$4^+ \rightarrow 2^+$ K-Electrons by Isotope	41
2.3	Summary of Energy Tests	42
2.4	ICEBall detector locations	47
2.5	ICEBall magnetic strengths with GEORGINA	49
2.6	ICEBall magnetic strengths with Clovershare	49
2.7	ICEBall calibration sources	50
2.8	GEORGINA calibration sources	56
2.9	^{56}Co calibration information	57
2.10	Target Enrichment and Thickness	61
2.11	GEORGINA detector locations	63
2.12	Clovershare detector locations	68
2.13	Data Structure of Electronics	69
3.1	Si(Li) Efficiencies in GEORGINA Configuration	76
3.2	GEORGINA Efficiency Parameters	76
3.3	Si(Li) Efficiencies in Clovershare Configuration	77
3.4	Clovershare March Run Efficiency Parameters	77
3.5	Clovershare May Run Efficiency Parameters	78
3.6	Timing gates by detector pairing for the ICEBall-GEORGINA Experiment	79
3.7	Relative Angles of Detectors	87
4.1	^{154}Gd Ground State Band $6^+ \rightarrow 4^+$ and $8^+ \rightarrow 6^+$ Extra Electron Peak Contributions in Singles	98
4.2	^{154}Gd Ground State Band Internal Conversion Coefficients from Singles	99
4.3	^{154}Gd Internal Conversion Coefficients from Singles	111
4.4	Uncorrected ^{154}Gd Internal Conversion Coefficients from Singles	112

4.5	^{154}Gd Internal Conversion Electrons without known Multipolarities	113
4.6	0^+ States in ^{154}Gd	116
4.7	$0^+ \rightarrow 0^+$ Transitions in ^{154}Gd	119
4.8	$2^+ \rightarrow 2^+$ Transitions in ^{154}Gd	120
4.9	$4^+ \rightarrow 4^+$ Transitions in ^{154}Gd	122
4.10	$6^+ \rightarrow 6^+$ Transitions in ^{154}Gd	124
4.11	$E0$ Contributions for $J^\pi \rightarrow J^\pi$ Transitions	126
4.12	$q_K^2(E0/E2)$ for $0^+ \rightarrow 0^+$ Transitions	127
4.13	$E0$ Contributions for $J^\pi \rightarrow J^\pi$ Transitions with Known Mixing Ratios	128
4.14	$B(E0)$ Ratios for $J^\pi \rightarrow J^\pi$ Transitions	129
5.1	^{156}Gd Ground State Band Internal Conversion Coefficients from Singles	139
5.2	0^+ States in ^{156}Gd	151
5.3	^{156}Gd Internal Conversion Coefficients from Singles	153
5.4	Uncorrected ^{156}Gd Internal Conversion Coefficients from Singles	154
5.5	^{156}Gd Internal Conversion Electrons without known Multipolarities	156
5.6	$0^+ \rightarrow 0^+$ Transitions in ^{156}Gd	158
5.7	$2^+ \rightarrow 2^+$ Transitions in ^{156}Gd	159
5.8	$4^+ \rightarrow 4^+$ Transitions in ^{156}Gd	160
5.9	$E0$ Contributions for $J^\pi \rightarrow J^\pi$ Transitions	162
5.10	$B(E0)$ Ratios for $J^\pi \rightarrow J^\pi$ Transitions	162
5.11	$E0$ Contributions for $J^\pi \rightarrow J^\pi$ Transitions	163
6.1	Dynamic Moments of Inertia of Bands seen in ^{154}Gd	168
6.2	Dynamic Moments of Inertia of Bands seen in ^{156}Gd	173
7.1	Data event - MDPP-16 (32 bit word)	189
A.1	List of ^{154}Gd gates	190
B.1	List of ^{156}Gd gates	268

CHAPTER 1

INTRODUCTION

Visible matter comprises ~ 5 percent of the universe. This matter makes up all living things and ecosystems, including the stars and planets. This matter exists entirely within the atom, with the nucleus holding more than 99.9 % of all matter. The nucleus is comprised of protons and neutrons in different configurations, held together by the strong force. Understanding this essential building block of the universe is at the heart of understanding how matter is formed and evolves.

The periodicity of phenomena on the atomic level and higher is well documented near stability, from the lattice structures of crystals, to the periodic table of the elements, organized in a way that groups elements that behave similarly due to the structure of the orbiting atomic electrons. The periodic table organizes elements based on similar properties, such as the halogens and noble gases. The elements can be neatly grouped by similar chemical properties. But, this periodic table does not delve into the subatomic level. The periodic table separates out atoms by the number of protons, which equates with the atomic number, but the number of neutrons in a nucleus can also vary, creating different isotopes of an element that is otherwise chemically identical. When mapping these two, we create Figure 1.1, known as the chart of nuclides. These nucleons are held together by the strong force, which has a short range. In the low-mass region of the chart, this creates a linear trend for the line of stability. As mass and the total number of nucleons increases, the line of stability trends toward the neutron side, due to repulsion by the coulomb force in the protons. As the neutrons do not contribute significantly to

the coulomb force, the addition of more neutrons creates larger separation between protons, decreasing the coulomb contribution.

The chart of nuclides has several notable features in this form. The color coding in Figure 1.1 is representative of the lifetimes of the isotopes represented in each box: lighter boxes are shorter-lived, while darker boxes are longer lived, with black boxes being stable isotopes. This creates a black line going up at approximately $N = Z$, known as the valley of stability. Additionally in this figure, several rows and columns are singled out. These rows and columns are a series of numbers referred to as the "magic" numbers, and highlight a form of periodicity in nuclei.

This periodic nature is explained through the shell model. Based on the same idea as atomic electrons, the "shells" occur due to the energy gap in levels. The neutrons and protons each have separate potential wells and levels populated. When examining solutions to this potential well, the magic numbers naturally form after including angular momentum, and taking the spin-orbit interaction into account, as seen in Figure 1.2. These magic numbers represent closed shells of a spherical nucleus model. It is of note, that a doubly-magic nucleus, that is, a nucleus with both neutron and proton close shells, is spherical. However, a nucleus with only a single closed shell is not, necessarily, spherical. These closed shells cause a variety of effects, including high energy of the first excited 2^+ state at closed shells, seen in Figure 1.3. The high energy of the first excited 2^+ state near the closed shells is due to the large energy gaps that occur at closed shells, seen on the right side of Figure 1.2.

Outside of these closed shells, there are large regions that do not work within the paradigm created by this phenomenon. Nuclei near the close shells are generally expected to be spherical in nature. However, the shell model becomes extremely complex away from the closed shells. In some cases, the magic numbers themselves appear to change. This change can be charted using the Nilsson model, which

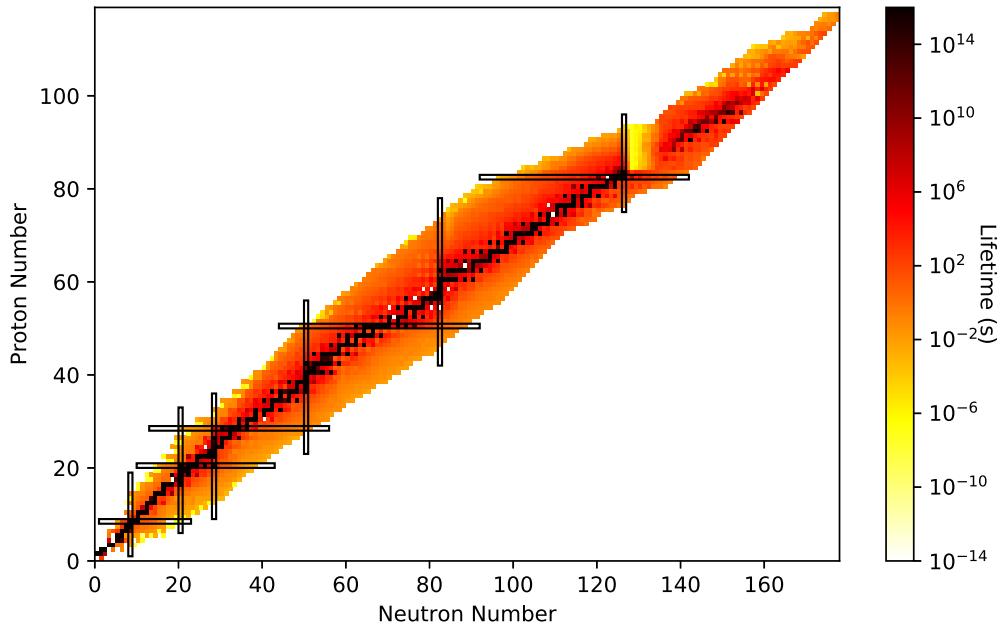


Figure 1.1. The chart of nuclides. The x -axis is increasing in neutron number from left to right, and the y -axis is increasing in proton number from bottom to top. The lower leftmost corner is the lightest elements, while the upper right are the heaviest, including superheavy elements. Black boxes indicate stable or extremely long lived nuclei. These nuclei are in the "center" of the chart, and known colloquially as the valley of stability. To either side of these stable nuclei are radioactive nuclei, which become shorter lived the farther they are from the valley, indicated by the lightening color. The black boxes are at neutron and proton numbers 8, 20, 28, 50, 82 and 126, marking closed shells for spherical nuclei, and are discussed in further depth within the text.

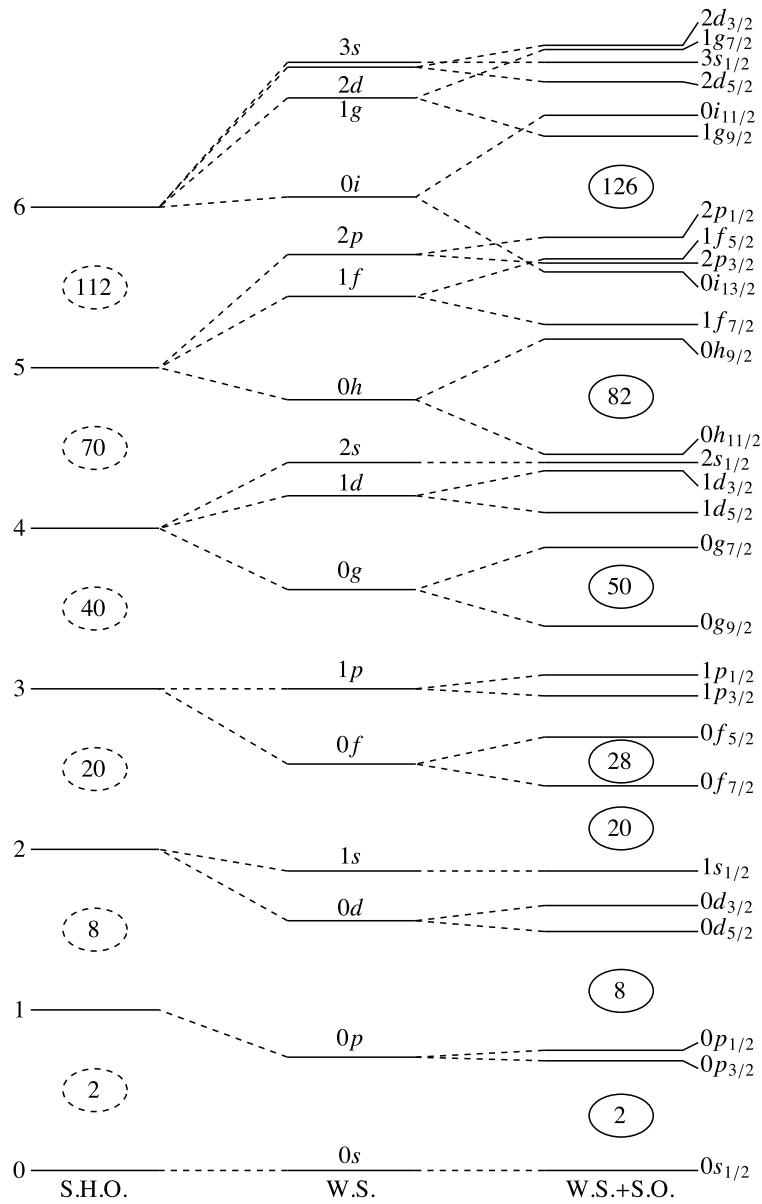


Figure 1.2. The evolution of the shell model and the creation of the closed shells of a spherical nucleus. On the left is the spherical harmonic oscillator. By changing the potential to the Woods-Saxon potential, creating a flat centered potential well, the degenerate states of the harmonic oscillator separate in energy. By further adding a spin-coupling term, the levels separate with energy spacings that create the closed shells and the magic numbers.

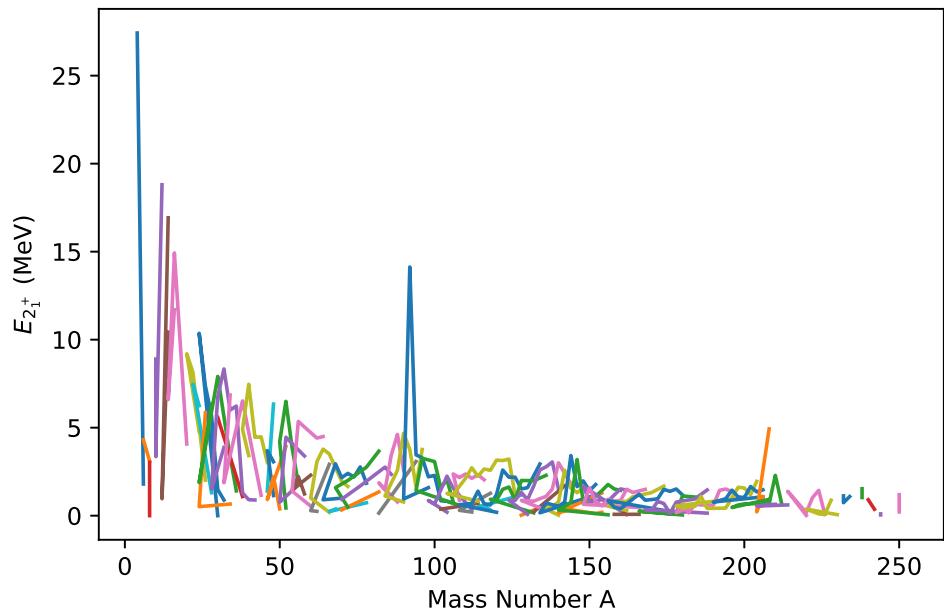


Figure 1.3. The energy of the first excited 2^+ state, plotted in isotopic chains by mass, distinguished by color. Areas of high excitation energy, around $A \sim 90, 140, 210$ all correspond to doubly magic areas. The areas around $A \sim 110, 170$ have much lower excitation energies and correspond to areas of deformation.

evolves the spherical shell model as it deforms into an ellipsoid. The model uses the deformation parameter β in the hamiltonian, allowing the level energies to be seen as a function of β . This leads to a collapse and change of the magic numbers, as seen in Figure 1.4.

In the area around $Z = 60$ and $N = 90$, away from the closed shells, nuclei do not hold a spherical shape. They lose symmetry on the major axes, becoming axially assymmetric. /i/e/ $x = y \neq z$ or $x \neq y \neq z$. The former case has two forms: oblate (pancake-like) or prolate (football-like) depending on whether the nucleus appears compressed (oblate) or elongated (prolate) along the symmetry axis (z), as seen in Figure 1.5. In Figure 1.4, the negative β corresponds to oblate nuclei, while positive β corresponds to prolate nuclei. There are more exotic deformations when the latter assymmetry is true, where symmetry is broken along two or more axes, resulting in pear-shapes and other exotic formations.

There are several ways to look for deformation in a nucleus. The most common is the quadrupole moment, which is directly related to the quadrupole deformation parameter, β_2 (also referred to as β) by

$$Q_0 = \frac{3}{\sqrt{5\pi}} Z R_0^2 \beta_2 (1 + 0.16\beta_2) \quad (1.1)$$

to second order, where Z is the atomic number, and R_0 is the approximate radius based on $R_0 = 1.2A^{1/3}\text{fm}$, where A is the number of nucleons [15]. This can be measured in even-even nuclei using the transition between the first excited 2^+ state and the 0^+ ground state. The onset of deformation can be seen in Figure 1.6, via the rapid increase in β_2 . The equation used to calculate β_2 for this figure is

$$\beta_2 = \frac{4\pi}{3ZR_0^2} [B(E2)]^{1/2} \quad (1.2)$$

where $B(E2)$ is the reduced transition probability for the $0_1^+ \rightarrow 2_1^+$ transition. There

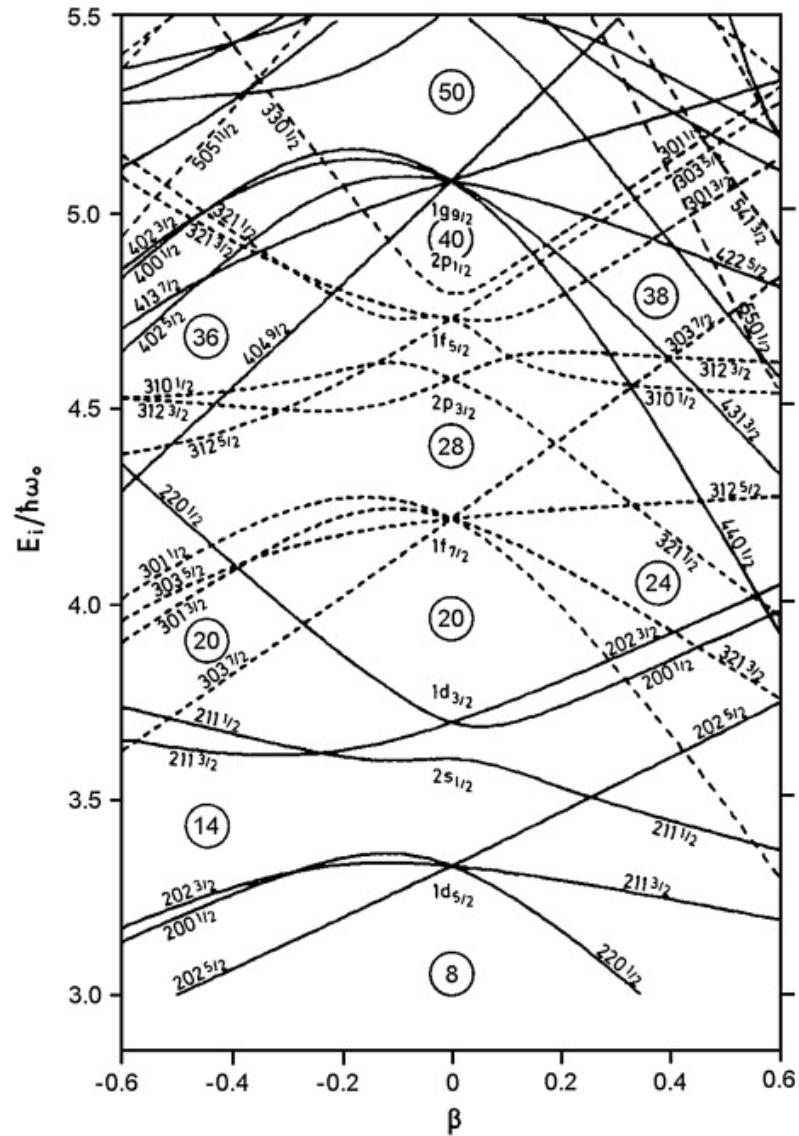


Figure 1.4. The Nilsson diagram, showing the evolution of the shell levels of the nucleus with respect to the deformation parameter β . As the nucleus gets more deformed, the closed shells break down. Taken from [17].

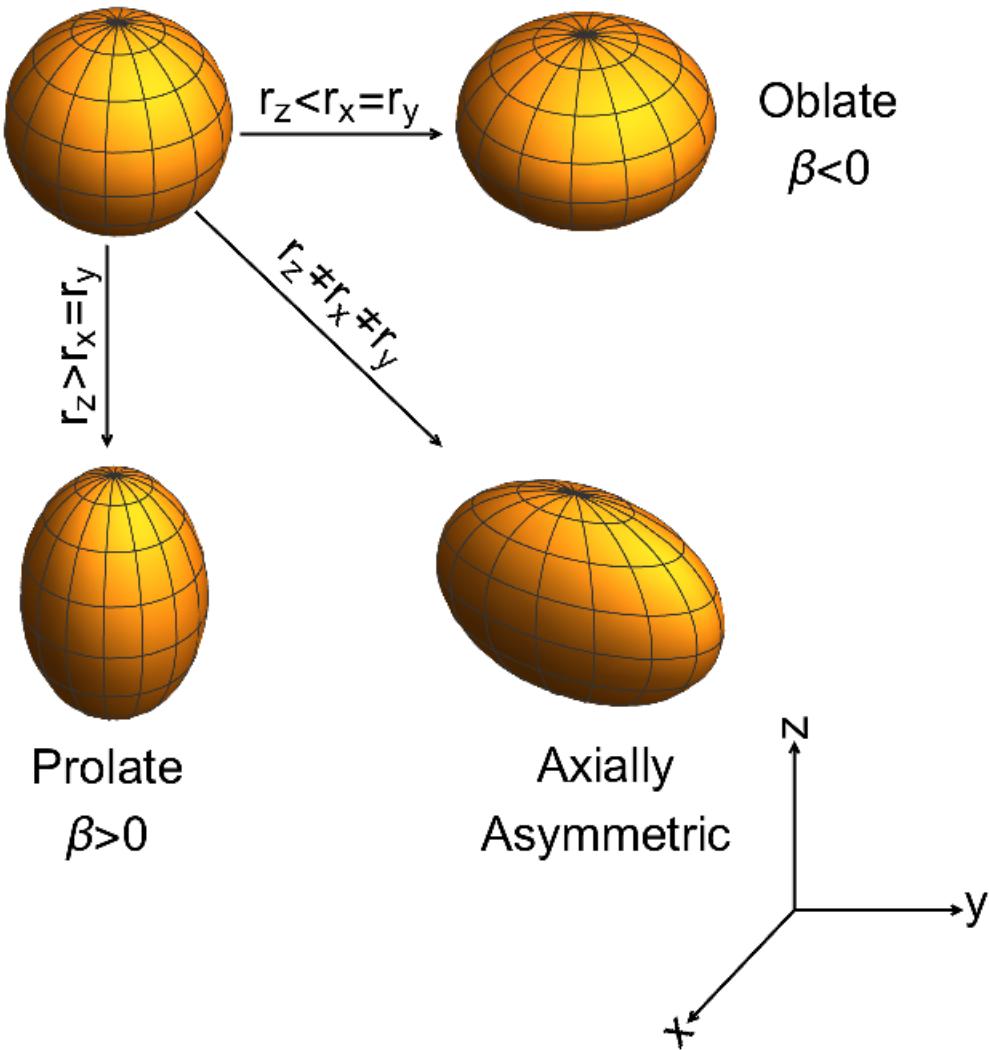


Figure 1.5. Illustration of different kinds of deformation, where spherical symmetry is lost. Treating the vertical as the axis of symmetry, the prolate nucleus is contracted along this axis, but is still axially symmetric. The oblate nucleus is elongated along this axis, but otherwise axially symmetric. It is also possible for the nucleus to lose axial symmetry.

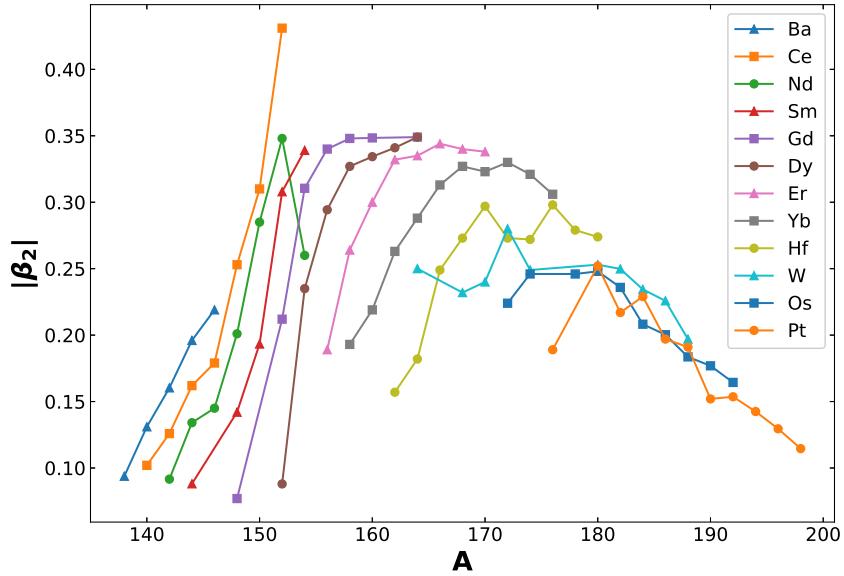


Figure 1.6. Plot of the norm of the deformation parameter β_2 along isotopic chains. Deformation can be seen setting in, as isotopic chains transition from spherical to deformed. β_2 is the quadrupole deformation parameter. The norm of β_2 is used, as the method used to calculate β_2 only gives a magnitude, not a sign. A secondary method is needed to distinguish between prolate and oblate.

is no way to distinguish prolate and oblate using this calculation, as the sign comes from the choice of sign on the root. The choice cannot be decided with the information in the equation.

In the lanthanide region, there are multiple stable isotopes of several elements, including samarium ($Z = 62$) and gadolinium ($Z = 64$). Studies across these elements show the onset of deformation through systematics like the energy of the first excited 2^+ state, and the ratio between the energies of the first 4^+ and 2^+ excited states, seen in Figure 1.7. The $E_{4_1^+}/E_{2_1^+}$ ratio theoretically tops out at 3.3. This ratio gives a measure of the moment of inertia of the ground-state band of the

nucleus, as, for a rigid rotor,

$$E = \frac{\hbar^2}{2\mathcal{J}_{eff}} J(J+1) \quad (1.3)$$

The moment of inertia, \mathcal{J}_{eff} , varies as energy increases. In a perfect rigid rotor, this would not be the case, and $\mathcal{J}_{eff} = \mathcal{J}_{rigid} = \frac{2}{5}Am_Nr_0^2A^{2/3}$. Taking the ratio of two energy levels, $J = 4$ and $J = 2$ in this case, would give a ratio of $10/3$, or 3.3. However, if it is not a rigid rotor, this number deviates from 3.3, as the change in the effective moment of inertia changes. The model used by Bohr and Mottelson shows that \mathcal{J}_{eff} is proportional to the square of the deformation parameter, β_2 , at low deformation, and approaches that of a rigid body at high deformation [11]. Much of these nuclei are successfully modelled with deformation in theoretical calculations [22]. These models can also open up new questions, when a previously identified feature does not fit with the assignment in an otherwise well described system.

1.1 Nuclear Excitations

There are many different ways a nucleus can be excited. While the types of excitations available depend on shape, there are two extremes to consider: a single particle excitation, and a collective excitation. The single particle excitation can be described as a single neutron, proton, or hole from a closed shell being excited to a different level. A collective excitation is usually thought of on a macroscopic level, looking at large numbers of nucleons moving together. These are only extremes, and there are excitations that can be described as multi-particle that would fit in-between. All of these excitations can exist within a nucleus.

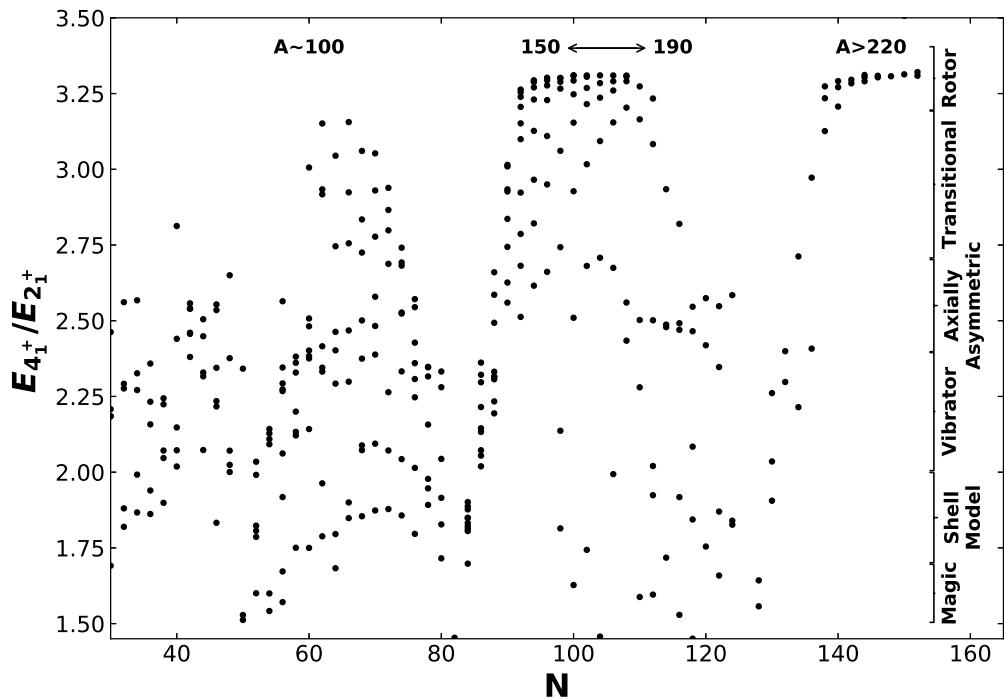


Figure 1.7. Plot of the ratio of the first 4^+ excited state energy to the first 2^+ excited state energy in even-even nuclei. On the right, the ratio is broken down into sections, based on structure. This ratio is an excellent diagnostic of the general nuclear structure.

1.1.1 Collective Excitations

Two types of collective excitations are rotational and vibrational excitations.

Rotational excitations occur in deformed nuclei, as discussed earlier, where spherical symmetry is broken, i.e. $x = y \neq z$ or $x \neq y \neq z$. These are caused by the rotation of the nucleus about the axis of symmetry. Vibrational excitations involve the compression and expansion of the nucleus as a whole. These come in different modes, and have a characteristic even spacing, seen in Figure 1.8.

There are several ways to tell the difference between these types of excitations. The collective models have well-described spacing of the levels, seen in Figure 1.8.

Vibrational excitations will be spaced regularly as $n\hbar\omega$, while rotational excitations will have increasing spacing between the levels as J increases. The reduced transition probabilities also give information about the difference between two different levels. The single-particle strengths for these transitions can be calculated and are known as Weisskopf estimates. The reduced transition probability can then be written in Weisskopf units using the estimate, which gives a measure of the collectivity of the transition. This does have limitations, however, as the Weisskopf estimate assumes a spherical nuclear model.

1.1.2 Shape Coexistence

Another aspect of nuclear shape is that of shape coexistence. In shape coexistence, there is more than one minima of the potential well, at different forms of deformation, as illustrated in Figure 1.9. These two minima represent two different shapes of the nucleus, and both can have excitations built upon them. The two shapes can also interact, causing shape mixing effects.

There are two main theoretical approaches to shape coexistence in nuclei: microscopic shell-model and mean-field descriptions [26]. These two approaches compliment each other. In the shell-model approach, a spherical shell model is

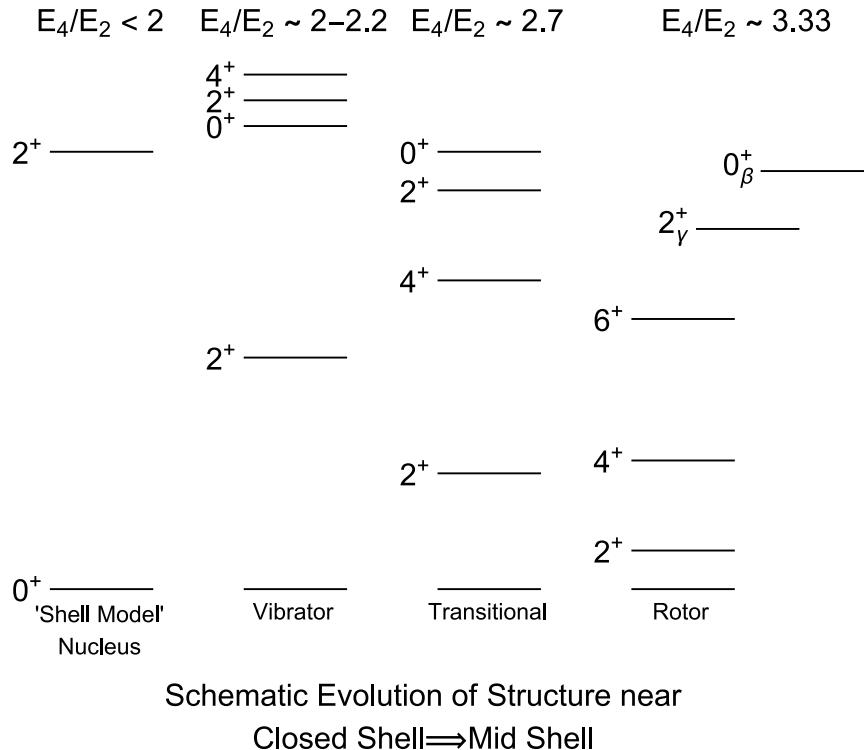


Figure 1.8. Cartoon of what excited states built upon different types of nuclei looks like. The closed shell has a large gap between the ground state and first excited state. In the vibrator model, this gap is lower, and a gap of comparable size exists before a cluster of states occurs. In the rotor model, the spacing follows a spacing pattern of $J_f(J_f + 1) - J_i(J_i + 1)$, with γ and β vibrations also coming into play. The transitional area between vibrator and rotor shows a shift between the two, with the clustered states of the vibrator model beginning to space out.

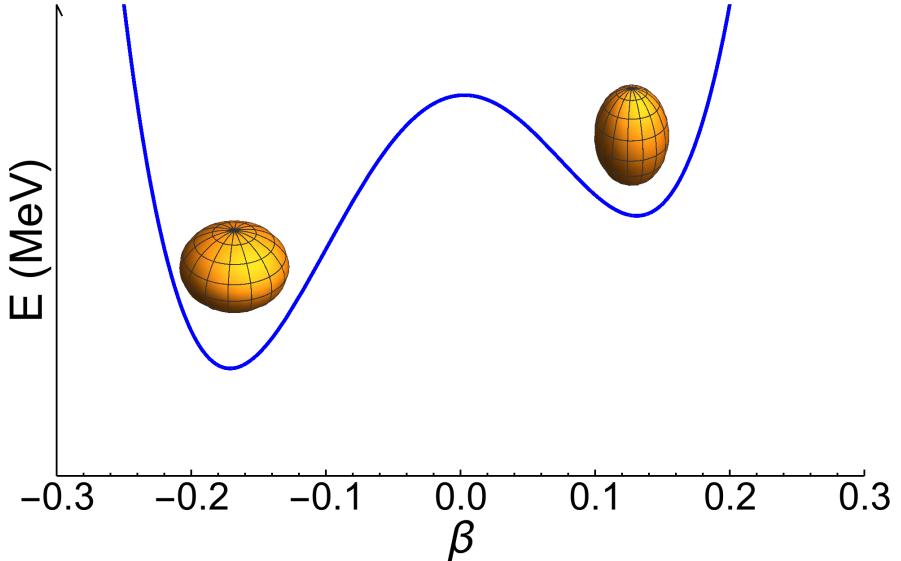


Figure 1.9. Cartoon of the potential well of a nucleus with shape coexistence. A second minima appears at a different β , the deformation parameter. Both of these minima can have excitations built on top of them, leading to shape coexistence. A representation of the shape at β is shown in each minima.

assumed, and particle-hole excitations across closed shells are examined. For an $np-nh$ configuration, it would appear the 0^+ states are too high. However, upon correcting for monopole interaction energy, and then coupling different $np-nh$ subspaces together, lower energy 0^+ states are calculated [16]. For calculations above $A = 80$, nucleons must be treated in pairs built from realistic collective pairs, as the model spaces become too large for current computational power[26]. In this approach, intruder states from even-even particle-hole excitations can create shape-coexisting 0^+ states.

The mean-field approach uses the self-consistent Hartree-Fock-Bogoliubov (HFB) theory. The effective forces are tuned to describe global nuclear properties. When

minimizing the energy, deformation in the nuclear shape can result. Various mean-field states are found, which have broken symmetries needed within the nucleus. These states are projected onto the proper particle numbers, isospin, and angular momentum to produce the physical states observed [26]. To compare with experimental date, the Hamiltonian must be rediagonlized in terms of the physical, projected states.

Shape coexistence, like the collective excitations, has spectroscopic fingerprints that can help determine its existence. As with deformation, large reduced transition rates are a possible sign of shape coexistence, but do not distinguish between static and dynamic deformation [26]. The diagonal $E2$ matrix elements can distinguish between the dynamic and static. The $E0$ transitions present in $J^\pi \rightarrow J^\pi$ may also be an indirect fingerprint of shape coexistence, as it is related to the change in the mean radius of the nucleus, as discussed in Section 1.3.1[64]. The low-lying states of interest can be difficult to observe, as they are not populated strongly in reactions.

1.2 Multipole Radiation

When the nucleus is in an excited state, with energy in excess of its ground state, this excess is stored in the protons and neutrons of the nucleus. These nuclei oscillate, creating electromagnetic fields with their oscillation. The nucleus is then capable of deexciting in different ways. With enough energy, the nucleus will preferentially emit particles, such as neutrons or protons. To do so, it must be at a high enough energy. In the case that enough energy is not present for particle emission, the electromagnetic force becoming the route of de-excitation, via photon emission in the form of γ -rays. These fields are commonly described using expansions of spherical harmonics, allowing for angular momentum identification of the radiation emitted. This expansion is commonly referred to as multipole expansion. Discussion and derivation can be found in detail in many references

[10, 31, 67].

The Maxwell equations are

$$\nabla \cdot \mathbf{E} = 0 \quad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (1.4a)$$

$$\nabla \cdot \mathbf{B} = 0 \quad \nabla \times \mathbf{B} = \frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t} \quad (1.4b)$$

for a vacuum with no source. Radiation from a nuclear transition has a well-defined energy, and therefore, a well-defined frequency ω . The time-dependence of the electric and magnetic field can be expressed as

$$\mathbf{E}(\mathbf{r}, t) = \text{Re} \left\{ \mathbf{E}(\mathbf{r}, t) e^{-i\omega t} \right\} \quad (1.5a)$$

$$\mathbf{B}(\mathbf{r}, t) = \text{Re} \left\{ \mathbf{B}(\mathbf{r}, t) e^{-i\omega t} \right\} \quad (1.5b)$$

Using equations 1.4 and 1.5, the Maxwell equations can be rewritten in the equivalent forms

$$(\nabla^2 + k^2) \mathbf{E} = 0 \quad (1.6a)$$

$$\nabla \cdot \mathbf{E} = 0 \quad (1.6b)$$

$$\mathbf{B} = -\frac{i}{ck} \nabla \times \mathbf{E} \quad (1.6c)$$

and

$$(\nabla^2 + k^2) \mathbf{B} = 0 \quad (1.7a)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (1.7b)$$

$$\mathbf{E} = -\frac{ic}{k} \nabla \times \mathbf{B} \quad (1.7c)$$

As these are equivalent, any pair of fields $\mathbf{B}(\mathbf{r})$ and $\mathbf{E}(\mathbf{r})$ that satisfy equations 1.6 must satisfy equations 1.7 and vice-versa. It can be shown that any arbitrary

solution to the Maxwell equations can be written as a linear combination of a transverse magnetic wave and a transverse electric wave, i.e.

$$\mathbf{r} \cdot \mathbf{E} = 0, \text{ or} \quad (1.8)$$

$$\mathbf{r} \cdot \mathbf{B} = 0 \quad (1.9)$$

When combining equations 1.8 and 1.6, a solution to the Maxwell equations can be derived. When doing the same with equations 1.9 and 1.7, a separate, linearly independent, solution can be derived. These two solutions in combination can express any of the solutions to the Maxwell equations as a linear combination. Because of the linear independence of the spherical harmonics, solutions to these wave equations can be written as multipole expansions, the term multipole in reference to the angular momentum, l , defining the series of the harmonic. The first set of solutions give the magnetic multipole fields, with the solutions

$$(\nabla^2 + k^2) \mathbf{E}^M = 0 \quad \mathbf{B}^M = -\frac{i}{ck} \nabla \times \mathbf{E}^M \quad (1.10a)$$

$$\nabla \cdot \mathbf{E}^M = 0 \quad \mathbf{r} \cdot \mathbf{E}^M = 0 \quad (1.10b)$$

where the M indicates the magnetic multipole component of the field. The second set give the electric multipole fields with the solutions

$$(\nabla^2 + k^2) \mathbf{B}^E = 0 \quad \mathbf{E}^E = -\frac{ic}{k} \nabla \times \mathbf{B}^E \quad (1.11a)$$

$$\nabla \cdot \mathbf{B}^E = 0 \quad \mathbf{r} \cdot \mathbf{B}^E = 0 \quad (1.11b)$$

where the E indicates the electric multipole component of the field. The total fields are then just the two components summed together.

These multipole fields can be written as an expansion of the spherical Hankel

functions, $h_l^{1,2}(kr)$, multiplied by the spherical harmonics, $Y_{lm}(\theta, \phi)$. The fields can then be broken apart into components of l , the angular momentum. The general expressions for \mathbf{E}^M and \mathbf{B}^E fields are

$$\mathbf{E}^M(\mathbf{r}) = \sum_{l,m,i} c_{lm}^{(i)} h_l^{(i)} \mathbf{l} Y_{lm}(\theta, \phi) \quad (1.12a)$$

$$\mathbf{B}^E(\mathbf{r}) = \sum_{l,m,i} d_{lm}^{(i)} h_l^{(i)} \mathbf{l} Y_{lm}(\theta, \phi) \quad (1.12b)$$

with \mathbf{E}^E and \mathbf{B}^M derivable from the curl expressions in equations 1.6 and 1.7. In this equation, the orbital angular momentum is expressed as $\mathbf{l} = -i(\mathbf{r} \times \nabla)$. The created multipole radiation is of E and M types, being referred to as El and Ml radiation. Then the vector spherical harmonics can be introduced in the form

$$\mathbf{X}_{lm} = \frac{1}{\sqrt{l(l+1)}} \mathbf{l} Y_{lm}(\theta, \phi) \quad (1.13)$$

A further simplification can be defined as

$$f_{lm}(kr) = a_{lm}^{(1)} h_l^{(1)}(kr) + a_{lm}^{(2)} h_l^{(2)}(kr) \quad (1.14)$$

leading to

$$\mathbf{B}_l^E(\mathbf{r}) = \sum_m f_{lm}^E(kr) \mathbf{X}_{lm}(\theta, \phi) \quad \mathbf{E}_l^M(\mathbf{r}) = \sum_m f_{lm}^M(kr) \mathbf{X}_{lm}(\theta, \phi) \quad (1.15a)$$

$$\mathbf{E}_l^E(\mathbf{r}) = -\frac{ic}{k} \nabla \times \mathbf{B}_l^E(\mathbf{r}) \quad \mathbf{B}_l^M(\mathbf{r}) = -\frac{i}{ck} \nabla \times \mathbf{E}_l^M(\mathbf{r}) \quad (1.15b)$$

With these definitions, the total electric and magnetic fields of can be written as

$$\mathbf{E}(\mathbf{r}) = \sum_{l=1}^{\infty} \sum m = -l^l a^E(l, m) \mathbf{E}^E(l, m; \mathbf{r}) + a^M(l, m) \mathbf{E}^M(l, m; \mathbf{r}) \quad (1.16a)$$

$$\mathbf{B}(\mathbf{r}) = \sum_{l=1}^{\infty} \sum m = -l^l a^E(l, m) \mathbf{B}^E(l, m; \mathbf{r}) + a^M(l, m) \mathbf{B}^M(l, m; \mathbf{r}) \quad (1.16b)$$

These amplitudes a^E and a^M can be derived [10]. They come out to be

$$a^E(l, m) \simeq -\frac{4\pi}{(2l+1)!!} \left(\frac{l+1}{l}\right)^{1/2} \kappa^{l+2} (Q_{lm} + Q'_{lm}) \quad (1.17a)$$

$$a^M(l, m) \simeq +\frac{4\pi}{(2l+1)!!} \left(\frac{l+1}{l}\right)^{1/2} \kappa^{l+2} (M_{lm} + M'_{lm}) \quad (1.17b)$$

where $\kappa \equiv \omega/c$, the wave number. Then, using the charge distribution, $\rho(\mathbf{r})$, and magnetic distribution $\mathbf{M}(\mathbf{r})$, and current distribution $\mathbf{j}(\mathbf{r})$ we define

$$Q_{lm} = \int r^l Y_{lm}^*(\theta, \phi) \rho(\mathbf{r}) dV \quad (1.18a)$$

$$Q'_{lm} = -\frac{i\kappa}{l+1} \int r^l Y_{lm}^*(\theta, \phi) \nabla(\mathbf{r} \times \mathbf{M}) dV \quad (1.18b)$$

$$M_{lm} = -\frac{1}{c(l+1)} \int r^l Y_{lm}^*(\theta, \phi) \nabla(\mathbf{r} \times \mathbf{j}) dV \quad (1.18c)$$

$$M'_{lm} = -\int r^l Y_{lm}^*(\theta, \phi) \nabla \mathbf{M} dV \quad (1.18d)$$

These are the electric (Q_{lm} and Q'_{lm}) and magnetic (M_{lm} and M'_{lm}) multipole moments. To use them for nuclear transitions, the distributions must take the initial and final states into account. They become

$$\rho(i, f; \mathbf{r}) = e\varphi_f^*(\mathbf{r})\varphi_i(\mathbf{r}) \quad (1.19a)$$

$$\mathbf{M}(i, f; \mathbf{r}) = \frac{e\hbar}{2Mc} \mu \{ \varphi_f^* \boldsymbol{\sigma} \varphi_i \} \quad (1.19b)$$

$$\mathbf{j}(i, f; \mathbf{r}) = \frac{e}{2M} [\varphi_f^*(\mathbf{p}\varphi_i) + (\mathbf{p}\varphi_f)^*\varphi_i] \quad (1.19c)$$

where M and e are the mass and charge of a single particle, $\mathbf{p} = i\hbar\nabla$ is the linear moment operator, μ is the magnetic moment in Bohr magnetons, and $\boldsymbol{\sigma}$ is the Pauli spin operator.

From here, it is possible to obtain the transition probability using Fermi's golden rule,

$$w_{if} = \frac{2\pi}{\hbar} |\langle f | H_{int} | i \rangle|^2 \rho, \quad (1.20)$$

where H_{int} is the interaction Hamiltonian from the generated field. This Hamiltonian is

$$H = -\frac{1}{c} \mathbf{A} \cdot \mathcal{J} \quad (1.21)$$

where \mathcal{J} is the current density and \mathbf{A} is the magnetic vector potential, which can be derived from the solutions above [63]. Combining all of this together with Fermi's golden rule, the transition probabilities are

$$T^E(l, m) = \frac{8\pi(l+1)}{l[(2l+1)!!]^2} \frac{\kappa^{2l+1}}{\hbar} |Q_{lm} + Q'_{lm}|^2 \quad (1.22a)$$

$$T^M(l, m) = \frac{8\pi(l+1)}{l[(2l+1)!!]^2} \frac{\kappa^{2l+1}}{\hbar} |M_{lm} + M'_{lm}|^2 \quad (1.22b)$$

Of note from this derivation, the transition probability decreases with increasing l , making lower angular momentum transitions more favorable. Additionally, looking at the multipole moments, once the wave equations have been added in, all but the Q_{lm} multipole moments have a factor of $1/c$. This makes the electric transition stronger than the magnetic transition of the same l .

There are limitations for the types of multipole radiation that can be emitted due to a transition, known as selection rules. One such limitation is that all of the multipole moments of equations 1.18 go to zero is $l = 0$, making $l = 0$ transitions forbidden via this method. Other rules are based on two pieces of information about the initial and final states: the spins and parities of the states. The spins, J ,

of the states select the allowable l , using the range $|J_i - J_f| \leq l \leq J_i + J_f$, meaning the change in angular momentum cannot exceed the sum of the spins of the two states, or be smaller than the difference of the spins of the two states. For example, a transition between two states of spins 4 and 2 would allow l from 2 to 6. The parity selection now helps select the set of solutions. With the first selection rule, in the example, the possibilities are E2, M2 ... E6, M6. If the parity stays the same between the two states, so positive to positive or negative to negative, the solutions are even El and odd Ml. If the parity changes, the solutions are odd El and even Ml. In an equation form, the parity change is

$$\pi(\text{El}) = (-1)^l \quad \pi(\text{Ml}) = (-1)^{l-1} \quad (1.23)$$

While not a selection rule, multipoles of lower order are stronger, and electric multipoles are stronger than magnetic multipoles, with Ml having about the same intensity as El + 1. There are also forbidden transitions to consider. The name is a misnomer, as these transitions are not strictly forbidden, but are highly suppressed, or must occur through a different mechanism. A relevant example to this text is that of the $l = 0$ transition, previously noted as forbidden. Photons carry an angular momentum of 1, so any transitions that occur via photon emission must have a $\delta J \geq 1$, so l must be at least 1. An E0 transition has $l = 0$, making it forbidden by single photon emission. However, it can still occur, several different ways, including internal conversion.

1.3 Internal Conversion

Not all transitions are equal, with certain multipolarities being preferred or excluded via selection rules. This makes transitions between some states highly improbable or, in the case of photon emission, impossible. The $0^+ \rightarrow 0^+$ transition

cannot occur via photon emission due to $\delta = 0$. To study these transitions, another type of deexcitation that occurs using the mechanisms in electromagnetic radiation must occur, assuming particle emission does not take place. Above 1.022 MeV of excitation, the nucleus can also deexcite via pair-production, the creation of an electron-positron pair. Below this energy, there are two processes to deexcite the nucleus: the aforementioned photon emission and internal conversion. Internal conversion is also an electromagnetic processing, occurring internally within the atom, with the excitation energy being transferred to an orbiting atomic electron. The electron is then ejected from its bound state, as the energy it has been given exceeds the electron binding energy, giving the electron a kinetic energy of

$$T_e = E_\gamma - BE \quad (1.24)$$

where BE is the atomic binding energy of the electron, and E_γ is the transition energy. The electron carries information about the multipolarity of the transition it was ejected from, as a photon would if emitted for the same transition. Unlike photon emission, it is not bound by a change in angular momentum, creating a process through which E0 transitions can occur, giving a method to study such transitions and states. Figure 1.10 shows the internal conversion coefficients of the K-shell electron for several multipolarities in gadolinium, plotted against energy. Both of these processes come from the same overall mechanism, that of the electromagnetic field, but internal conversion also has a component from the nuclear interaction. As a result, internal conversion can be described with the conversion coefficient

$$\alpha = \frac{w_e}{w_\gamma} \quad (1.25)$$

where w_e is the transition probability due to internal conversion and w_γ is the transition probability due to single photon emission for the same transition.

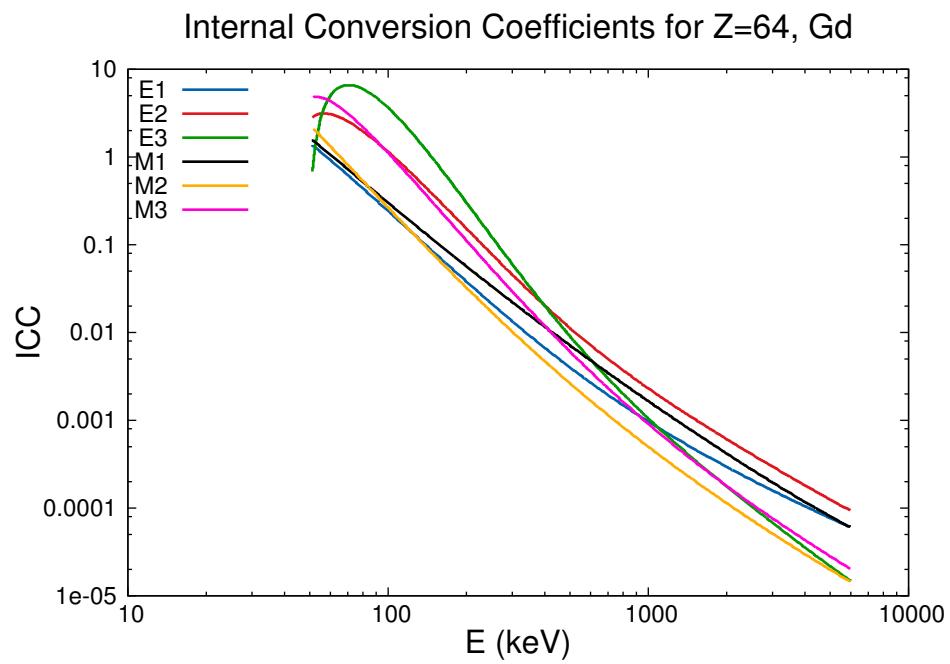


Figure 1.10. The theoretical K-shell conversion coefficients for the magnetic and electric multipoles for $l = 1, 2, 3$, calculated using BrICC [32]. Each of the multipoles is distinct. Further distinction between multipoles can be seen by plotting the intensity ratios between different electronic shells.

Internal conversion is a result of the interaction of the electron's wavefunction with that of the nucleus. The probability for internal conversion can be derived using Fermi's Golden rule, equation 1.20, where H_{int} is the electromagnetic interaction of the combined initial nuclear and electron states, to the combined final nuclear state and free electron, and ρ is the accessible energy state density for the ejected electron. For simplicity, this derivation will be specifically for K -shell electrons, and look at the area outside of the nuclear radius [10, 57, 58].

The electron can be treated as a plane wave once it has been ejected from the K shell, assuming the electron energy is large compared to the binding energy. This makes the initial and final states of the electron

$$\psi_i = \left(\pi \left(\frac{a_0}{Z} \right)^3 \right)^{-1/2} e^{-\frac{RZ}{a_0}} \quad (1.26a)$$

$$\psi_f = V^{-1/2} e^{i\mathbf{k}\cdot\mathbf{R}} \quad (1.26b)$$

where $a_0 = \hbar^2/me^2$ is the Bohr radius of the hydrogen atom, \mathbf{R} is the position vector between the electron and the center of the nucleus, \mathbf{k} is the wave vector of the electron after ejection, and V is the volume of the "box", which should drop out in the final result, as ρ also contains the volume in its formulation. The energy state density can be rewritten in terms of the wave vector \mathbf{k} to be

$$\rho = z(k)d\Omega, \text{ where} \quad (1.27)$$

$$z(k) = V \frac{m\hbar k}{(2\pi\hbar)^3} \quad (1.28)$$

Taking into account there are two electrons in the K -shell, the probability w_{if} can then be rewritten as

$$w_{if} = 2 \frac{2\pi}{\hbar} z(k) \int |\langle f | H_{int} | i \rangle|^2 d\Omega \quad (1.29)$$

For this derivation, the electric multipoles are being considered, so only the electrostatic interaction is being used. By using the magnetic interaction instead, the magnetic multipoles can also be derived. By assuming the interaction is occurring outside of the nucleus, the Hamiltonian can be assumed to be the electrostatic interaction between the protons of the nucleus and the electron or

$$H = \sum_{n=1}^Z \frac{e^2}{|\mathbf{R} - \mathbf{r}_n|} \quad (1.30)$$

where \mathbf{R} is as described above, and \mathbf{r}_n is the proton position from the center of the nucleus. Then the matrix element is

$$|\langle f | H_{int} | i \rangle| = \left(V \pi \left(\frac{a_0}{Z} \right)^3 \right)^{-1/2} \sum_{n=1}^Z \int e^{-i\mathbf{k} \cdot \mathbf{R}} \varphi_f^* \frac{e^2}{|\mathbf{R} - \mathbf{r}_n|} e^{-\frac{RZ}{a_0}} \varphi_i d\Omega \quad (1.31)$$

with φ_i and φ_f being the initial and final wave functions of the nucleus, respectively. The integration must occur over both the coordinates of the electron and over the coordinates in the nuclear wave functions, such that $d\Omega = d^3R d\tau$ where τ represents the nuclear coordinates.

The main contribution to the integral is going to come from $\mathbf{R} > \mathbf{r}_n$, the earlier assumption the interaction is occurring outside of the nucleus. This allows for an expansion of $|\mathbf{R} - \mathbf{r}_n|^{-1}$ in terms of spherical harmonics such that

$$\frac{1}{|\mathbf{R} - \mathbf{r}_n|} = \sum_{l=0}^{\infty} \sum_{m=-l}^l \frac{4\pi}{2l+1} \frac{r_i^l}{R^{l+1}} Y_{lm}(\Theta, \Phi) Y_{lm}^*(\theta_n, \phi_n) \quad (1.32)$$

where Θ and Φ are the polar vector of \mathbf{R} and θ_n and ϕ_n are the same for \mathbf{r}_n .

Substituting this into equation 1.31 gives

$$|\langle f | H_{int} | i \rangle| = \frac{e}{\left(V \pi \left(\frac{a_0}{Z} \right)^3 \right)^{1/2}} \sum_{l=0}^{\infty} \sum_{m=-l}^l \frac{4\pi}{2l+1} Q_{lm}(i, f) J_{lm} \quad (1.33)$$

where $Q_{lm}(i, f)$ is the electric multipole matrix element and J_{lm} is the integral

$$J_{lm} \equiv \int e^{-\frac{RZ}{a_0}} e^{-i\mathbf{k}\cdot\mathbf{R}} R^{-(l+1)} Y_{lm}(\Theta, \Phi) d^3R \quad (1.34)$$

Because of the earlier assumption of the outgoing electron being a plane wave by assuming the energy is sufficiently large, such that $ka_0/Z \gg 1$. With this approximation, $e^{-\frac{RZ}{a_0}} \simeq 1$ and the integral can be evaluated. Treating θ and ϕ as polar angles for \mathbf{k} ,

$$J_{lm} = 4\pi i^{-l} \frac{k^{l-2}}{(2l-1)!!} Y_{lm}(\theta, \phi) \quad (1.35)$$

Substituting all of this into 1.20, the transition probability is

$$w_{if} = 128\pi \frac{me^2 Z^3}{\hbar^2 a_0^3} \sum_{l=0}^{\infty} \sum_{m=-l}^l \frac{k^{2l-3}}{[(2l-1)!!]^2} |Q_{lm}(i, f)|^2 \quad (1.36)$$

There is a strong dependence on atomic number Z and the multipole order l . The magnetic transition follows a similar derivation, with a solution distinct from the electric one just derived. Thus, measuring the conversion coefficient of a transition can be used to determine the multipolarity of the transition. The theoretical K -shell conversion coefficients for $l = 1, 2, 3$ are plotted in Figure 1.10 for comparison.

1.3.1 E0 Transitions

Like the electromagnetic radiation, $l = 0$ transitions are still forbidden in the internal conversion proof above, as the rate directly depends on Q_{lm} , which goes to 0 for $l = 0$, which is relevant to transitions where the initial and final spins are the same. This, however, arises from the previous assumption of looking only outside of the nucleus. While the majority of the interaction between nucleus and electron occurs outside of the nucleus, when this goes to zero, as is the case for $l = 0$, the correction term from inside the nucleus must be taken into account, i.e. $R < r_i$.

This exchanges the positions of R and r_i on the right in the expansion from equation 1.32. Revisiting equation 1.31, and setting $l = 0$ it becomes

$$|\langle f | H_{int} | i \rangle| = \frac{e}{\left(V\pi\left(\frac{a_0}{Z}\right)^3\right)^{1/2}} \sum_{n=1}^Z \int d\tau \varphi_f^* \frac{1}{r_n} \varphi_i \int_{R < r_n} d^3 R e^{-i\mathbf{k} \cdot \mathbf{R}} e^{-\frac{RZ}{a_0}} \quad (1.37)$$

As integration over \mathbf{R} extends only to very small numbers, the exponentials on the right can be approximated as 1, simplifying the integral to just $d^3 R$, making it $4\pi r_n^3/3$. The matrix element is now

$$|\langle f | H_{int} | i \rangle| = \frac{4\pi}{3} \frac{e}{\left(V\pi\left(\frac{a_0}{Z}\right)^3\right)^{1/2}} \sum_{n=1}^Z \int d\tau \varphi_f^* r_n^2 \varphi_i \quad (1.38)$$

The sum is of the order of magnitude of the square of the nuclear charge radius, denoted by \mathcal{R}^2 . Substituting 1.38 into 1.29, the probability for an electric multipole of order $l = 0$ becomes

$$w_{if} = \frac{32}{9} \frac{e^2}{\hbar c} Z^3 \left(\frac{\mathcal{R}^2}{a_0^2}\right)^2 \left(2 \frac{\Delta E}{mc^2}\right)^{1/2} \frac{mc^2}{\hbar} \quad (1.39)$$

where $\Delta E = E_f - E_i - B_K$ is the energy of the electron, with B_K indicating the atomic binding energy of the K -shell. At first glance, this indicates the $l = 0$ (E0) component is a measure of the change in the nuclear charge radius. The diagonal components of the E0 operator are measures of the mean-square charge radius [64]. The off-diagonal terms, which would be the terms probed by conversion electron spectroscopy are a direct probe of the nuclear dynamic. The interpretation of E0 components does not have a complete consensus [27].

One common way to rewrite the E0 transition probability is $\mathcal{W} = \Omega \rho^2$, where Ω are non-nuclear factors, i.e. $\Omega = \sum_i \Omega_i$ where i is the electron subshell, defined by Church and Wesener [19]. This contains the nuclear factors to ρ^2 , referred to as the

nuclear strength parameter [18, 64]. In practice, ρ^2 tends to fall between $10^{-1} - 10^{-3}$, so it is standard to quote $\rho^2 \times 10^3$. For a given transition,

$$\rho_{if}^2 = \left| \frac{\langle f | \sum_n e_n r_n^2 | i \rangle}{eR^2} \right|^2 \quad (1.40)$$

This makes ρ^2 unitless, due to the division by eR^2 , where R is the standard nuclear radial approximation. With this, ρ^2 can be scaled in terms of mass, allowing for comparisons across the nuclear chart [64]. The term $\langle f | \sum_n e_n r_n^2 | i \rangle$ shows the strength of ρ^2 is from two main factors: the nuclear matrix element of the initial and final states, and δr , the change in the nuclear radius. For large ρ^2 , both of these must be large. For small ρ^2 , the matrix element may be large, with only a small change in δr , or the opposite may be true.

Experimentally, this strength parameter can be determined by the equation

$$\rho^2(E0) = q_K^2(E0/E2) \times \mathcal{W}(E2) \times \frac{\alpha_K(E2)}{\Omega_K(E0)} \quad (1.41)$$

where the E2 being referenced is the transition from the 0^+ state to the 2^+ state in the ground state band. $\mathcal{W}_\gamma(E2)$ is the transition rate, the inverse of the partial mean life of the γ -component of the E2 transition. $q_K^2(E0/E2)$ is a ratio of the intensity of the E0 and E2 electrons and $\alpha_K(E2)$ is the theoretical conversion coefficient of the E2 transition. This electronic factor has an energy dependence. This strength parameter is directly related to the reduced transition probability $B(E0)$ by

$$B(E0) = \rho^2(E0)e^2R^4 \quad (1.42)$$

In a mixed transition with an E0 component, the E2 in equation 1.41 would be the E2 component of the transition, and would be calculated using the gamma rays and the theoretical E2 internal conversion coefficient[19]. A second mixing term, ϵ would

be used to represent the ratio between the E0 electrons and E2 gamma-ray components, and would be defined as

$$\epsilon^2 = (\alpha_{exp} - \alpha(E2)) - \delta^2(\alpha(M1) - \alpha_{exp}) \quad (1.43)$$

where δ is the mixing ratio between E2 and M1, obtained from gamma-gamma directional correlation experiments, and the theoretical conversion coefficients for pure M1 and E2 transitions are used. In fact, ϵ^2 is $q_K^2(E0/E2)\alpha_K(E2)$.

These transitions have been seen to manifest via a variety of phenomena, including the shell model, collective models, shape mixing i.e. shape coexistence and intruder states[64]. Depending upon which phenomenon is responsible for a given E0 transition will change ρ^2 , of which estimates can be calculated. Many of these models lead to small E0 rates, the exception being shape mixing. Due to the dominant nature of shape mixing in E0 decays, it is difficult to compare these models with experimental ρ^2 values unless shape-mixing effects have been removed [64].

1.4 The Rare Earth Region: $N = 90, Z = 60$

The region around $N = 90, Z = 60$ is far away from the closed shells, and constitutes the largest region of deformed nuclei in the chart of nuclides. This region of nuclei has a variety of interesting structure features that are not fully characterized, including a large number of bound 0^+ states outside of the ground state [45]. Many of these states have not been fully characterized, including gamma-spectroscopy, electron-spectroscopy and lifetime measurements. With this large number of 0^+ states, electron-spectroscopy becomes vital, as transitions between two 0^+ states must be E0 in nature, which cannot occur via gamma radiation.

The isotopic chains of the rare earth region can be seen plotted against the magnitude of the deformation parameter β in Figure 1.6. There is a large amount of deformation in the region, and several isotopes with large numbers of 0^+ states, including ^{152}Gd , ^{156}Gd , ^{162}Dy , and ^{168}Er , all of which have more than 10 0^+ states below 3 MeV [45], with ^{154}Gd having the most, at 16 states. In addition, ^{158}Gd also has a large number of 0^+ states at 13 [38]. This makes the Gd isotopic chain excellent for studying the nature of these excited 0^+ states in deformed nuclei.

In section 1.1, collective excitations and shape coexistence were discussed. In these deformed nuclei, there are a large number of collective excitations, the nature of which is unknown. E0 transitions in the region have been seen to have large transition strengths, and have been associated with shape coexistence [64].

Confirming shape coexistence cannot be done with a single measurement. Heyde and Wood give an in-depth look at the spectroscopic fingerprints needed to identify shape coexistence [26]. Direct fingerprints for deformation include the diagonal E2 matrix elements, which can be done via Coulomb excitation experiments. Reduced transition probabilities for E2 transitions also show deformation. Such values largely require lifetime measurements, of which there are a variety of possible methods, dependent on the range of the lifetimes being measured, including doppler broadening and doppler shift methods.

While an indirect fingerprint, with the large number of 0^+ states present in this region of the nuclear chart, E0 strengths become a vital spectroscopic fingerprint in interpretation and understanding [26]. These transitions give a model-independent view of the configurations underlying the transition. Other hints to shape coexistence include changes in masses, mean-square radii, and pair occupancies. Systematic patterns in the more direct fingerprints can also hint at possible shape coexistence. Fully understanding the 0^+ states in nuclei will require a systematic study of E0 transition strengths including those from $\Delta J \neq 0$.

CHAPTER 2

EXPERIMENTAL SETUP

2.1 Overview

For E0 transitions, two types of radiation must be measured: conversion electrons and gamma rays. As conversion electron transitions and intensities are usually tabulated as conversion coefficients, which compare the electron intensities to the respective gamma-rays from the transition, both conversion electrons and gamma-rays must be measured in this experiment. Further, to identify pure E0 transitions, the experiment must show no corresponding gamma rays.

Due to the high level density of nuclei in the rare-earth region, detectors with high resolution are favored for these experiments. The detectors must also be placed in a way that allows for simultaneous measurement of both particles and gamma-rays. For good resolution, high purity germanium (HPGe) detectors are the standard for gamma-rays. For electron detection, silicon-based detectors have the best resolution. The electron energies of interest range from less than 100 keV to over 1 MeV. Standard silicon detectors do not have an energy range that high for electrons, so lithium-drifted silicon (Si(Li)) detectors were used, which have a higher top-end electron energy.

To make these measurements, three separate experiments were performed at the Nuclear Science Laboratory (NSL) at the University of Notre Dame using the Internal Conversion Electron Ball (ICEBall) Spectrometer paired with two configurations of HPGe detectors: GEORGINA and Clovershare (Figure 2.1).

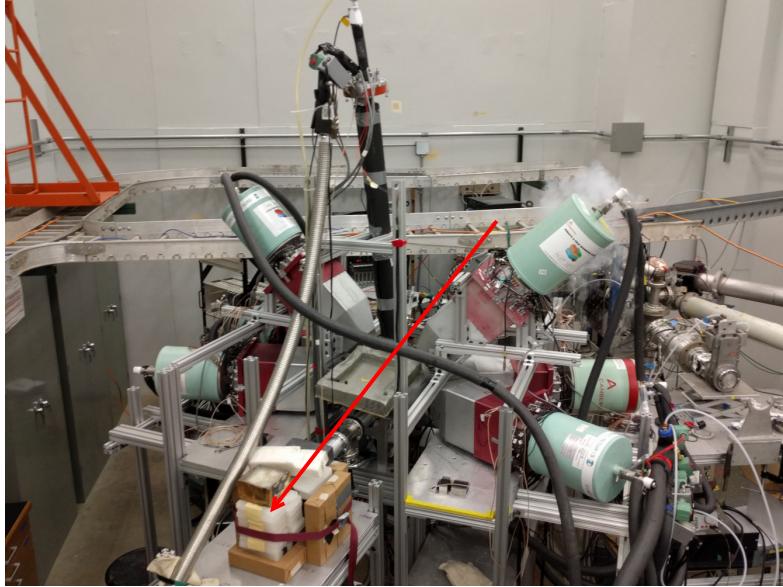


Figure 2.1. Image of the assembled experimental configuration with Clovershare. One HPGe detector is not visible from this angle. The red line indicates the beam direction.

The $(\alpha, 2n)$ reaction favors populating states of low, even- J and positive parity. These states, including the 0^+ states, are the ones of interest in this study, making this a natural reaction to select for the experiment. The Q value for this reaction is -14.774 MeV for ^{154}Gd and -13.638 MeV for ^{156}Gd , meaning an alpha beam of the necessary energy for the reaction to occur is well within the capability of the NSL. A $\text{Sm}(\alpha, 2n)\text{Gd}$ reaction was used with different enriched targets to create the desired Gd isotope, with an α beam of 20 MeV. A bunched beam was used to create timing for coincidence identification. This beam energy was determined by a measurement using a natural Sm target, with ICEBall and two neutron detectors. TALYS was run to determine a range of energies to maximize the cross section of the desired reaction, while minimizing the neutron flux, as neutrons can damage HPGe detectors. Figure 2.2 shows the cross sections for the (α, xn) reactions for ^{154}Sm in the α energy range of interest. Beam energies from 16 MeV to 21 MeV

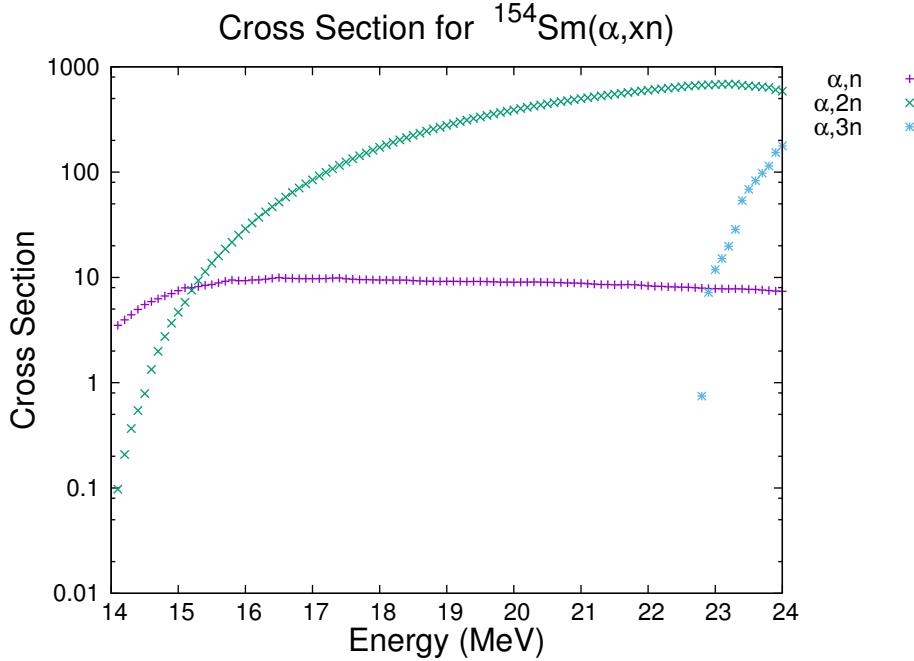


Figure 2.2. Comparison between the cross sections of the (α, xn) reactions on ^{154}Sm , calculated using Talys[36]. The less neutrons being lost, the lower the energy before the reaction occurs. Maximizing the cross section with respect to the neutron flux can be done by finding an energy where the desired reaction, $(\alpha, 2n)$ can be maximized, while minimizing the (α, n) and $(\alpha, 3n)$ reactions. Specifically, this means finding an energy before the $3n$ reaction has significant strength, while maximizing the $2n$ reaction, as the n reaction is nearly the same across the energy regime of interest.

were tested. Measurements were done at 1 MeV intervals. The cross section was calculated using conversion electron measurements from ICEBall, and compared with the neutron flux at that energy.

Two different forms of electronics were used for the data acquisition system. The data acquisition systems used were based on the NSCL DAQ[47, 49]. The electronics used with GEORGINA were shaped by NIM modules before going through ADC VME modules to the computer [13, 39]. The Clovershare data used the XIA Pixie-16 modules, which take the place of the NIM and VME modules for shaping and converting the signals to digital data [66].

The digital files were converted to files for the analysis software using `evt2root` [59]. Analysis, reported in chapter 3, was done using the CERN Root Data Analysis Framework and Radware [12, 50].

2.2 Nuclear Science Laboratory at Notre Dame

The Nuclear Science Laboratory at the University of Notre Dame has been in operation since the 1934, with accelerators in operation since 1937. Currently, the NSL operates locally with three accelerators: the FN Tandem, the 5 MV Single-Ended Sta. Ana accelerator, and the 3 MV 9S Tandem. There is also a fourth accelerator, a 1 MV machine called CASPAR, located in the Homestake Mines in South Dakota. The experiments in this dissertation were performed on the FN Tandem. Figure 2.3 is the current layout of the NSL. The accelerator in the top right of the figure is the FN Tandem.

The FN Tandem has two ion sources: the Helium Ion Source (HIS) and the Multi-Cathode Source of Negative Ions Using Cesium Sputtering (MC-SNICS). Most elements can absorb the electrons from cesium to create negative ions. Helium cannot and needs a special source, as discussed below. For the experiments described, the HIS was used.

Due to the electron binding energy of helium, the HIS uses a duoplasmatron. Within the duoplasmatron, a thin tungsten wire is heated while surrounded by the source gas (helium). The tungsten wire emits electrons, and positively ionizes the helium. This helium is extracted from the duoplasmatron and focused by an einzel lens, where it then goes through the lithium charge exchange. This section of the HIS works as a small two stage acceleration system, although the charge exchange works added electrons instead of stripping them. The positively charged ions are accelerated through a lithium vapor, and some gain electrons, becoming neutral or negatively charged. Those ions that become negatively charged are accelerated out

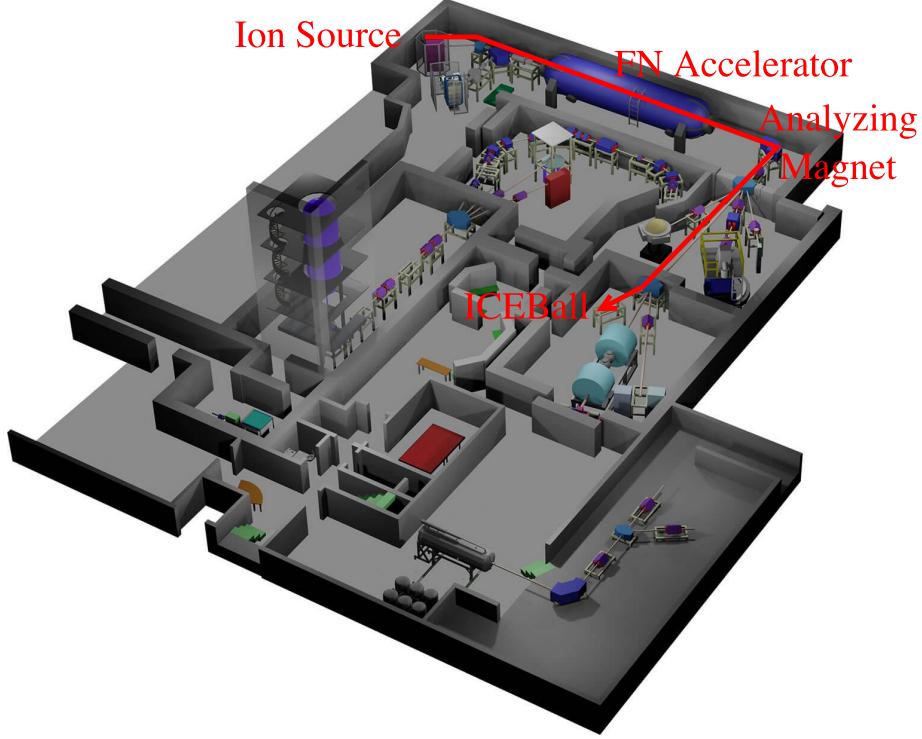


Figure 2.3. Current layout of the Nuclear Science Laboratory at Notre Dame. The red line indicates the path taken by the beam, from ion source, through the accelerator and analyzing magnet, before being sent to ICEBall.

the other end of the HIS, before being sent to the FN Tandem. Figure 2.4 is a schematic of this system.

The FN Tandem has been in operation since 1968. It is a Van de Graaff. Charging chains carrying a static charge deposit the charge on a terminal in the center of the machine. The charging chains is a pelletron type, upgraded from the original insulated rubber belt system in 2000. The pelletron chain consists of metal pellets connected using nylon, allowing for each pellet to be electrically isolated as it carries electrical charge to the terminal [46]. The beamline is kept at vacuum, but the area outside of the beamline, but within the accelerator tank, is filled with a mixture of CO₂ and dry nitrogen gas, at approximately 200 PSI. To create a more

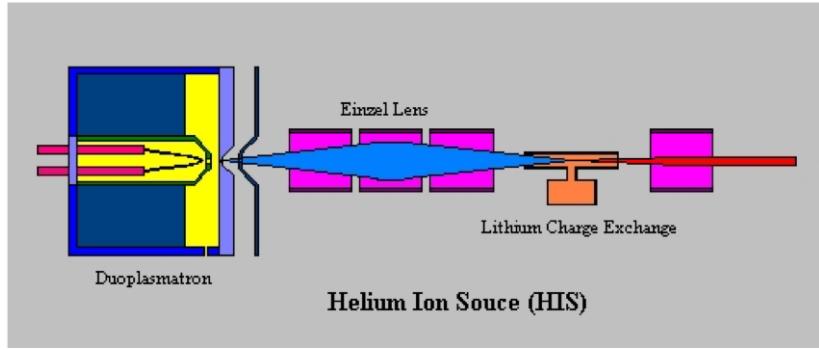


Figure 2.4. A schematic of how the Helium Ion Source works.

uniform acceleration field, the terminal is brought to ground along the beamline using resistor-lined tubes. The resistors used create a uniform electrical field down the tube, allowing for uniform acceleration.

It is known as a tandem because the system is a two-stage acceleration. Negatively charged ions enter the system, accelerating toward a positively charged terminal shell. Inside of the shell, the ions go through carbon stripping foils $3 \mu\text{g}/\text{cm}^2$ thick, becoming positively charged as electrons are pulled off. The ions then accelerate away from the terminal for the second stage. The total energy gained by the ions is the terminal voltage times the quantity of one plus the final charge state.

After being sent through the accelerator, the beam goes through the analyzing magnet. This magnet is set for the specific species, energy, and charge state to bend 90 degrees to the experimental set ups.

2.3 Beam Production

Because of the use of HPGe detectors in an $(\alpha, 2n)$ reaction, the neutron flux must be minimized in comparison to the cross section of the reaction, as the (α, n) reaction is also open. The neutrons being produced will interact with everything, including the HPGe detectors, which can be damaged by the (n, γ) reaction on

TABLE 2.1
ISOTOPE DISTRIBUTION OF NATURAL SAMARIUM

Isotope	Lifetime (y)	Abundance (%)
^{144}Sm	Stable	3.08
^{147}Sm	1.06×10^{11}	15.00
^{148}Sm	7×10^{15}	11.25
^{149}Sm	Stable	13.82
^{150}Sm	Stable	7.37
^{152}Sm	Stable	26.74
^{154}Sm	Stable	22.74

germanium. To do this, a range of energies to test were selected by looking at theoretical cross sections in Talys[36]. These energies were then tested with natural Sm targets, ICEBall, and two liquid scintillators to detect neutrons.

Samarium, as with many even-Z elements in the lanthanide region, has many stable isotopes. A total of five stable isotopes of Samarium exist, with two other isotopes being long-lived. Table 2.1 summarizes the abundances and lifetimes of the various isotopes found in natural Samarium. The two isotopes used our enriched targets are the ones that have the highest natural abundances, $^{152,154}\text{Sm}$.

2.3.1 Talys Calculations

Talys [36] is code for simulation of nuclear reactions dynamically with neutrons.

Cross sections can be estimated using Talys to guide where an experiment may want to run to optimize production, as is the case presently.

Both natural and enriched samarium targets were used and modelled within Talys.

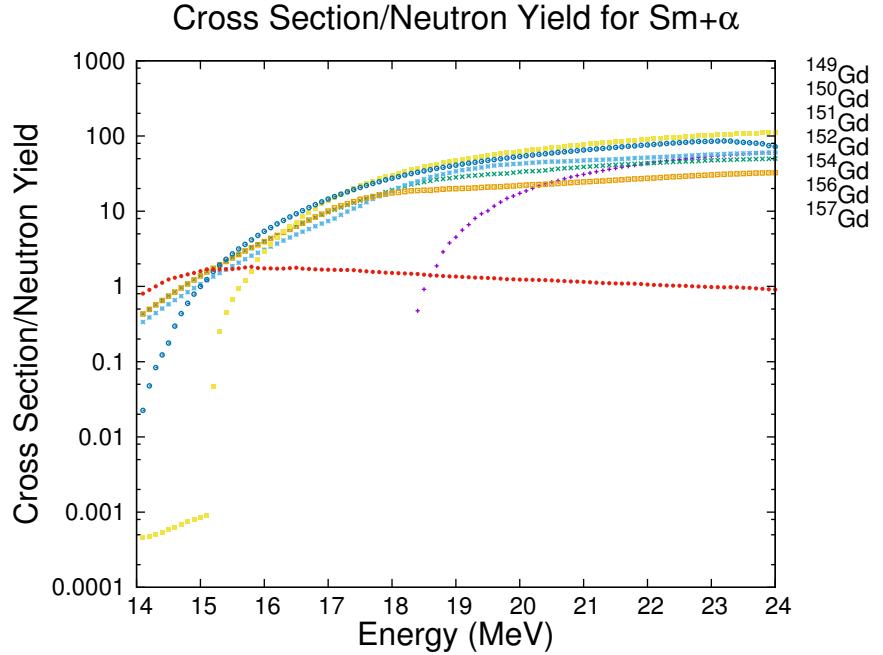


Figure 2.5. Talys calculation of the highest cross sections divided by the total neutron yield in the natural samarium target. Only the largest contributors are included, as there are a large number of products possible between the isotopes of samarium in the target and the reactions channels open for each. All of the largest contributors are gadolinium isotopes. The ratios for the isotopes of interest appear to plateau around 20-22 MeV.

A range of energies, from 14 MeV to 24 MeV α -particles were run. These energies allowed a large number of potential reaction products from elastic, inelastic, transfer, and knockout reactions. Figure 2.5 is a visual summary of the strongest modelled reactions in the natural samarium target. It was decided to test the energies between 16-21 MeV.

2.3.2 Test Analysis and Results

To test these Talys calculations and determine the best running energy, the cross section and neutron flux had to be measured at each energy without the use of the HPGe detectors. This was done using a natural samarium target. ICEBall was used

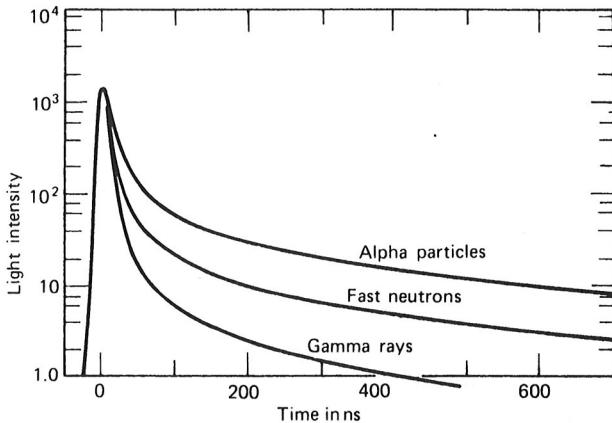


Figure 2.6. A cartoon illustration of pulse shape based on radiation type, in the same detecting material. The different types of radiation interact with the detecting material creating a fast and slow excitation. The ratio of these fast and slow excitations is based on the radiation, and expressed in the length of the decay tail. Picture taken from [34].

to measure the cross section by examining the K-electrons from ground-state band transitions in the reactions of interest, as the ground-state band is the most significantly populated band in the reaction. The DAQ described in section 2.7.2 was used. Spectra from ICEBall were fit using the methods described in 3.1. The peaks were identified under the assumption the ground-state band transitions would be the strongest peaks in the spectra, and populate quickly.

The neutrons were measured using two NE213 liquid organic scintillators. In these detectors, it is possible to do pulse shape discrimination, as the neutrons have longer decay times than the gammas in these detectors due to the nature of the interaction with the scintillating liquid [34]. This produces longer trails in the detectors, leading to different decay shapes, see Figure 2.6.

By plotting the pulse height against the decay tail time, neutrons and gammas can be separated. In this setup, this was done by sending the signal to a Mesytec MPD-4 (Multi-channel Pulse Discriminator) [40]. This is a module built for pulse

17 MeV Neutron Detector

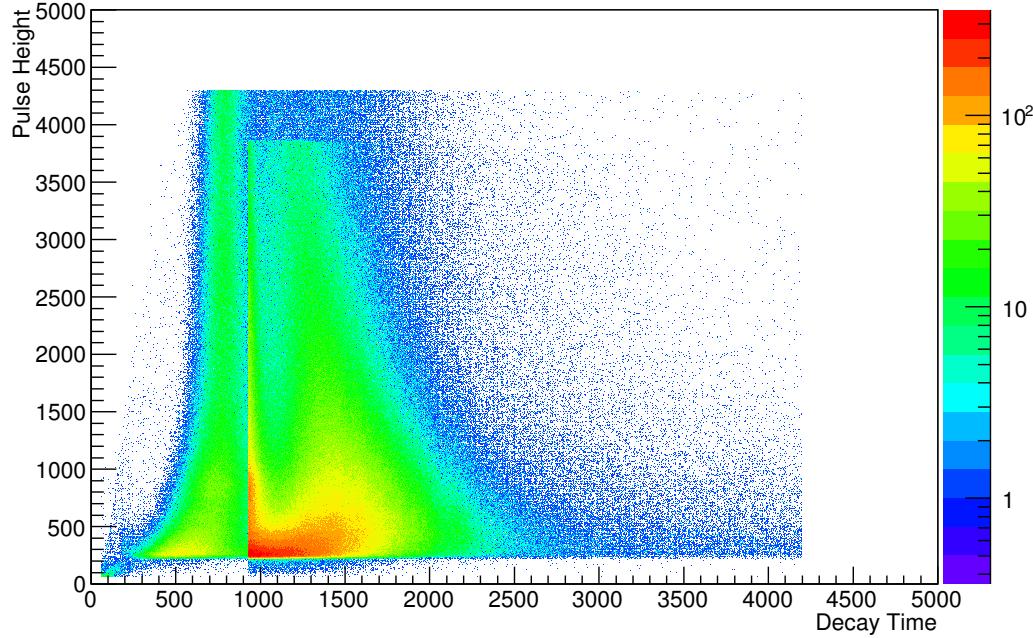


Figure 2.7. Plot of the pulse height vs the decay time in the NE212 detectors. Two clear areas are seen. The leftmost, with the shorter decay time, is the gammas, while the more spread out peak on the right is neutrons.

discrimination. It splits the signal, amplifies for the pulse height, and sends the other signal to a constant fraction discriminator (CFD), which converts the value to an amplitude using a time to analog converter (TAC). A CFD works by measuring the time it takes for a pulse to decrease to a fraction of its original height. Both of these signals (the pulse amplitude and the TAC amplitude) are recorded in the data. Plotting these values in a two-dimensional plot distinguishes the gammas and neutrons, as in Figure 2.7. There is a threshold cut off that removed most gammas from the spectrum.

Once the detectors were calibrated, each energy was run, with a total of four different targets across six energies. For each energy, the peaks in the electron spectra were identified with their respective isotopes, working under the assumption

TABLE 2.2
 $4^+ \rightarrow 2^+$ K-ELECTRONS BY ISOTOPE

Isotope	Energy (keV)
^{154}Gd	197
^{156}Gd	149

that only the lowest-lying states and the ground-state band would be populated significantly enough to see the electrons. These peaks are representative of the reaction cross section for that isotope. The K-electron of the $4^+ \rightarrow 2^+$ ground state transition for both nuclei of interest was clearly visible and clean at all energies, listed in Table 2.2. These K-electrons were used for the cross-section calculation.

To calculate the cross section, the formula

$$\sigma = \frac{Yield}{\delta x * Q * \epsilon} \quad (2.1)$$

was used, where δx is the target thickness, Q is the integrated charge, and ϵ is the efficiency of the Si(Li) detector at the electron energy. The *Yield* is the area under the curve.

This then had to be compared with the neutron production cross section. A bunched beam was used to get a time-of-flight. Calculating the neutron flux was done using the formula

$$\sigma_n = \frac{Neutrons}{\delta x * Q} \quad (2.2)$$

where δx and Q are the same as used for the cross section. The neutron yield is obtained by pulse shape discrimination, as seen in Figure 2.7. The left-most line is

TABLE 2.3
SUMMARY OF ENERGY TESTS

Energy (MeV)	Cross Section (mb)		Neutron Flux (neutrons/s)	Ratio (σ/Φ_n)	
	^{154}Gd	^{156}Gd		^{154}Gd	^{156}Gd
16	0.126 (13)	0.334 (25)	1.247 (4)	0.101 (10)	0.268 (20)
17	0.554 (54)	1.273 (91)	2.142 (13)	0.259 (25)	0.594 (43)
18	3.124 (133)	5.427 (233)	4.281 (30)	0.730 (31)	1.268 (55)
19	3.030 (172)	5.327 (217)	4.205 (25)	0.721 (41)	1.267 (52)
20	5.618 (167)	9.438 (275)	5.806 (36)	0.968 (29)	1.626 (48)
21	7.438 (267)	12.827 (419)	7.410 (54)	1.004 (37)	1.731 (58)

the gamma-flash that comes from the prompt gammas. The secondary gathering of counts, to the right, is the neutron peak. This area can be integrated over to get the number of neutrons seen during the run.

Table 2.3 summarizes the results at each energy. Note, that the cross section is not reflective of the full cross section of the isotopes, but of the K-electrons for that transition. This ratio, of the cross-section to the neutron flux, was ultimately used to determine the running energy. The two higher energies, 20 and 21 MeV, have a ratio that agrees within error for both isotopes, indicating a plateau, in agreement with the expectation from Talys. Ultimately, 20 MeV was chosen for two reasons: to minimize any higher energy reaction channels, and because the neutron flux was lower in rate.

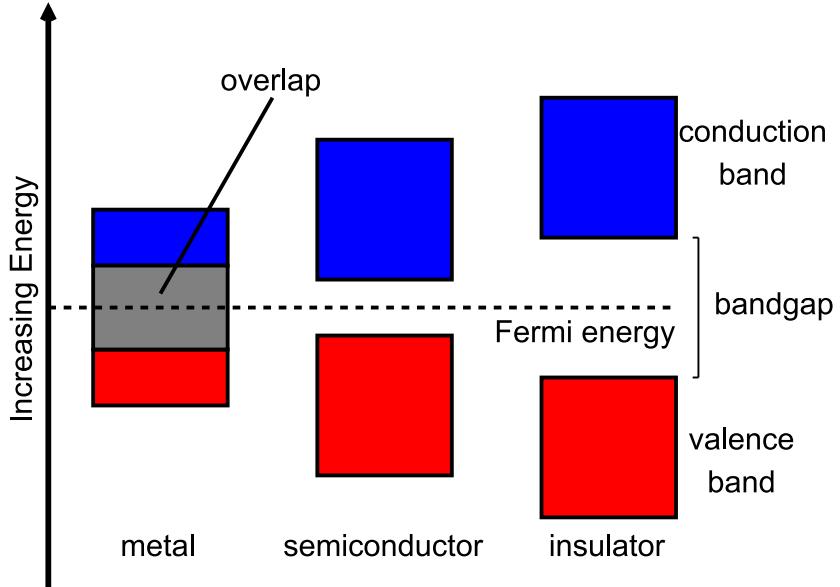


Figure 2.8. Cartoon of how the bandgap works for insulators, conductors, and semiconductors. Insulators have a large band gap, conductors overlap, and semiconductors have a small bandgap.

2.4 Electron Detection

Silicon detectors are a commonly used particle detector, with good energy resolution. The detectors are semiconductors, using a $p - n$ junction to create a depletion region, which becomes the active detection region[34]. Semiconductors have a bandgap around 1eV. A cartoon of the bandgap can be seen in Figure 2.8. The $p - n$ junction takes two semi-conductor materials, one of p -type (an excess of holes in the valence band) and one of n -type (an excess of electrons in the conduction band) together. The holes and electrons mix, creating a depletion region, which becomes the active detection area. This region can be enlarged to some extent by putting a bias across the detector.

Because electrons are much more penetrative than protons or alphas of the same kinetic energy, it takes much more material to fully stop electrons. Silicon detectors only have active regions of 1mm-2mm thick, due to impurities, and cannot be made

thick enough for the purposes of measuring electrons in the energy range of interest. Lithium-drifted silicon, colloquially Si(Li), detectors operate under the same principles as silicon detectors, but can be manufactured with larger depletion regions and a larger detectable energy range of electrons. These work by specifically introducing impurities, or doping, in this case lithium, to the detector, drifting it to create a $p - i - n$ junction. The i region is also known as the intrinsic region, and is created by the drifting of the lithium, creating a highly purified region. The thickness of these detectors is only limited by the distance across which the lithium can be successfully drifted.

2.4.1 Internal Conversion Electron Ball Spectrometer

The Internal Conversion Electron Ball Spectrometer (ICEBall) was developed at the University of Pittsburgh by Metlay et al. [43, 44]. Originally at the Spin Spectrometer at Oak Ridge National Laboratory, it was later stationed at the Wright Nuclear Structure Laboratory at Yale University with the YRAST Ball, until being brought to the University of Notre Dame and stationed in the Nuclear Science Laboratory West Target Room, on a dedicated beamline [5].

Originally designed to go inside of large gamma detector arrays, ICEBall consists of six mini-orange spectrometers, cooled using liquid nitrogen for ideal energy resolution. The liquid nitrogen is administered using an autofill system with two thermal sensors for a start and stop signal. When the sensor closest to the detectors warms enough, liquid nitrogen is pumped into the system. A second sensor at the top of the dewar is used to stop the fill of liquid nitrogen before it overflows. This system fills in 15 to 20 minute intervals. ICEBall can be seen in Figure 2.9, open for servicing.

The Si(Li) detectors inside of ICEBall are 5 mm thick, with a surface area of 750 mm² [44]. Table 2.4 summarizes the locations of the six Si(Li) detectors inside of



Figure 2.9. Image of ICEBall, open for servicing. The beam would go from left to right in the picture. The target ladder is blank, in the center of ICEBall.

ICEBall, using spherical coordinates. A thin aluminized mylar foil is placed in front of the Si(Li) detectors to block low-energy electrons and δ -rays that successfully make it past the mini-orange filter (discussed in 2.4.1.1).

2.4.1.1 Mini-Orange Spectrometer

Between the detectors and the target are mini-orange filters. First designed in the 1970s, mini-orange filters are a permanent magnet array surrounding a high-Z material, as seen in Figure 2.10 [29, 30]. In ICEBall, this material is tungsten, and the magnets are made of SmCo₅, and arranged in groups of 3. The tungsten acts as a blocker, lowering background from the target that can be due to γ rays and heavy charged particles. The magnets create a field that bends electrons toward the detector, while bending positrons away from the detector, further lowering background noise.

Compared to conventional spectrometers that sweep over electron energies by changing the magnetic field of the spectrometer, the mini-orange spectrometers are

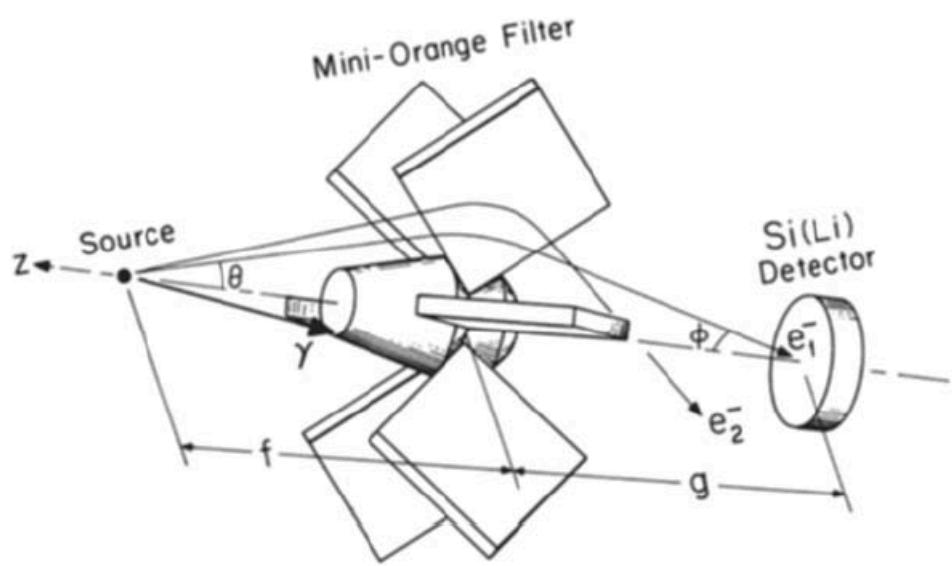


Figure 2.10. Graphic of the mini-orange filter. The central blocker keeps γ -rays from hitting the detector. The magnets bend electrons toward the detectors, and positrons away from the detectors. Being permanent magnets, they are optimized for a range of electron energies, and can cause overbending or underbending of electrons outside of that energy range, making the magnetic filter a factor in the efficiency. Taken from [44].

TABLE 2.4
ICEBALL DETECTOR LOCATIONS

Detector	θ	ϕ
1	90	79.2
2	270	100.8
3	172	129.9
4	198	31.7
5	18	148.3
6	355	50.1

The beam axis is the z-axis. θ is the angle in the xy-plane, where 0 degrees is beam left. ϕ is the azimuthal angle, with respect to the beam axis. All values are in degrees.

able to cover a wide range of energies at once. A drawback of this is the permanent magnets. The field is optimized for a range of energies, and must be pre-selected before the experiment. Switching magnets mid-experiment is not feasible, due to the downtime needed to warm-up the system, bring it to atmosphere, replace the filter, and then return the system to data taking conditions. Additionally, the efficiency of the system is a convolution of the detector's efficiency and the mini-orange filter in front of it, meaning each configuration must have a separate efficiency measurement. Tables 2.5 and 2.6 summarize the magnetic configurations of the mini-orange filters used in the experiments. The higher the magnetic strengths, the higher energy the peak efficiency of the mini-orange spectrometer, seen in Figure 2.11. For the GEORGINA configuration, the magnetic strengths were all of similar values. For the Clovershare configuration, the filters were replaced on three detectors to enhance electron efficiency at higher energies.

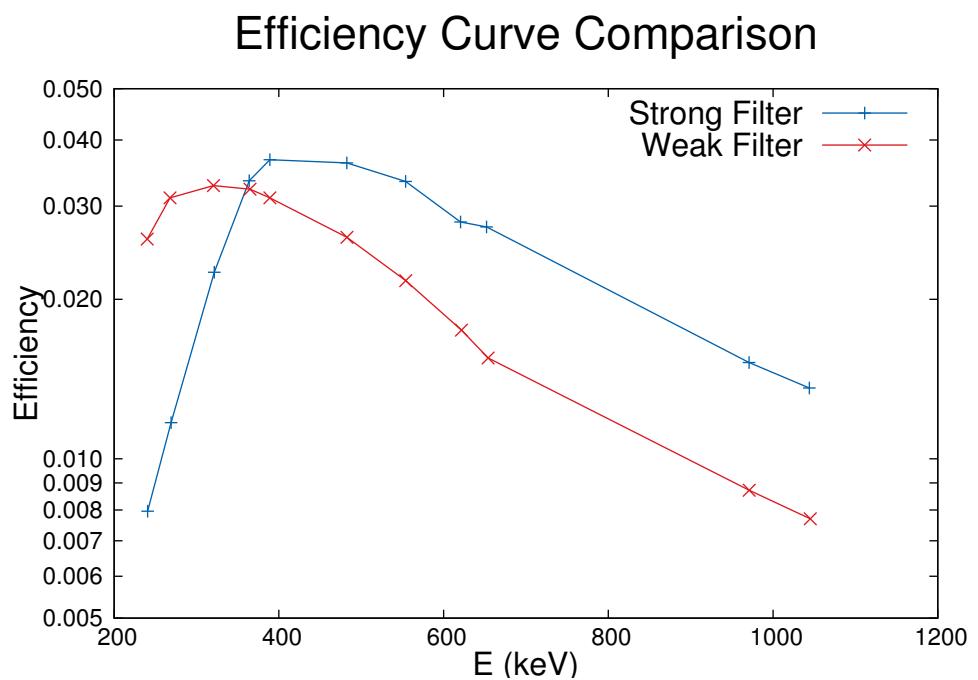


Figure 2.11. An efficiency comparison of magnetic filters of two different strengths. The weak filter has been weakened by 70% compared to the strong filter, using the Pitt split-pole spectrograph. Data from [43].

TABLE 2.5
ICEBALL MAGNETIC STRENGTHS WITH GEORGINA

Detector	Filter	Strengths
1	M13	815,740,830
2	M15	939,911,949
3	M21	850,875,900
4	M14	972,911,992
5	M18	856,913,963
6	M22	845,900,900

Magnetic strengths of the mini-orange filters used in the GEORGINA experiments, listed in Gauss. All the magnetic strengths are similar and geared for efficiency peaking at approximately 300-400 keV.

TABLE 2.6
ICEBALL MAGNETIC STRENGTHS WITH CLOVERSHARE

Detector	Filter	Strengths
1	M13	815,740,830
2	M20	1228,1292,1265
3	M21	850,875,900
4	M2	1411,1420,1410
5	M16	1286,1340,1285
6	M22	845,900,900

Magnetic strengths of the mini-orange filters used in the Clovershare experiments, listed in Gauss. The magnetic strengths are mid-energy range and high-energy range for efficiency.

TABLE 2.7
ICEBALL CALIBRATION SOURCES

Source	Activity	Electron Energy (keV)	Intensity (%)
^{133}Ba	0.331(7) μCi ¹	240.413	0.331
		266.868	0.698
		320.032	1.308
		347.866	0.370
^{207}Bi	0.306(8) μCi ²	481.697	1.562
		554.4	0.469
		975.657	7.243
		1048.1	1.838

Calibration source information for ICEBall. The energies and the respective intensities are listed for each source. Intensities are taken from [62]. The intensity of the 347 keV line in ^{133}Ba is both the 384K and 356L intensities combined.

2.4.2 Calibration

ICEBall is energy and efficiency calibrated using two sources: ^{133}Ba and ^{207}Bi . The specific properties of the two sources are listed in Table 2.7. The ^{207}Bi covers the high-energy regime, with lines around 500 keV and 1000 keV. The ^{133}Ba covered energies from 200 to 400 keV. Both sources are low activity, preventing incomplete or overlapping charge collection from hindering the resolution of the detectors during calibration runs.

The energy calibration is assumed to be quadratic in nature, although both linear and quadratic calibrations are performed. Figure 2.12 shows both of these fits and

¹Measured on May-4-2012

²Measured on May-4-2012

their respective residuals for one of the Si(Li) detectors. These values are summarized in the next chapter for each experiment.

The efficiency calibration is fitted to equation

$$\ln(\epsilon) = p_1 + p_2 \ln(E) + p_3 E \quad (2.3)$$

The efficiency is a convolution of the magnetic configuration and the inherent detector efficiency. Using the efficiency points, the analytic expression for the efficiency was determined empirically in previous work [5]. These experimental values are summarized in the next chapter for each experiment.

2.5 Gamma Detection

Gamma radiation is highly penetrating, making semiconductor detectors of normal purity unusable, as they have limitations on the depth of the depletion region that are nowhere near the necessary depths needed for fully stopping gamma radiation. The depletion depth is inversely proportional to the number of impurities in the material [34]. High-purity germanium, also known as intrinsic germanium, detectors decrease these impurities to increase the depletion depth, resulting in depths of several centimeters.

Like other semi-conductor detectors, there are dead zones that attenuate the radiation. Generally, this attenuation is negligible above 200 keV. These detectors have a high resolution due to the small bandgap of 0.7 eV in germanium. This small bandgap comes at a price, and HPGe detectors can only be operated at liquid nitrogen temperatures, as putting any voltage on a room temperature HPGe would result in high leakage currents and noise, making detection impossible.

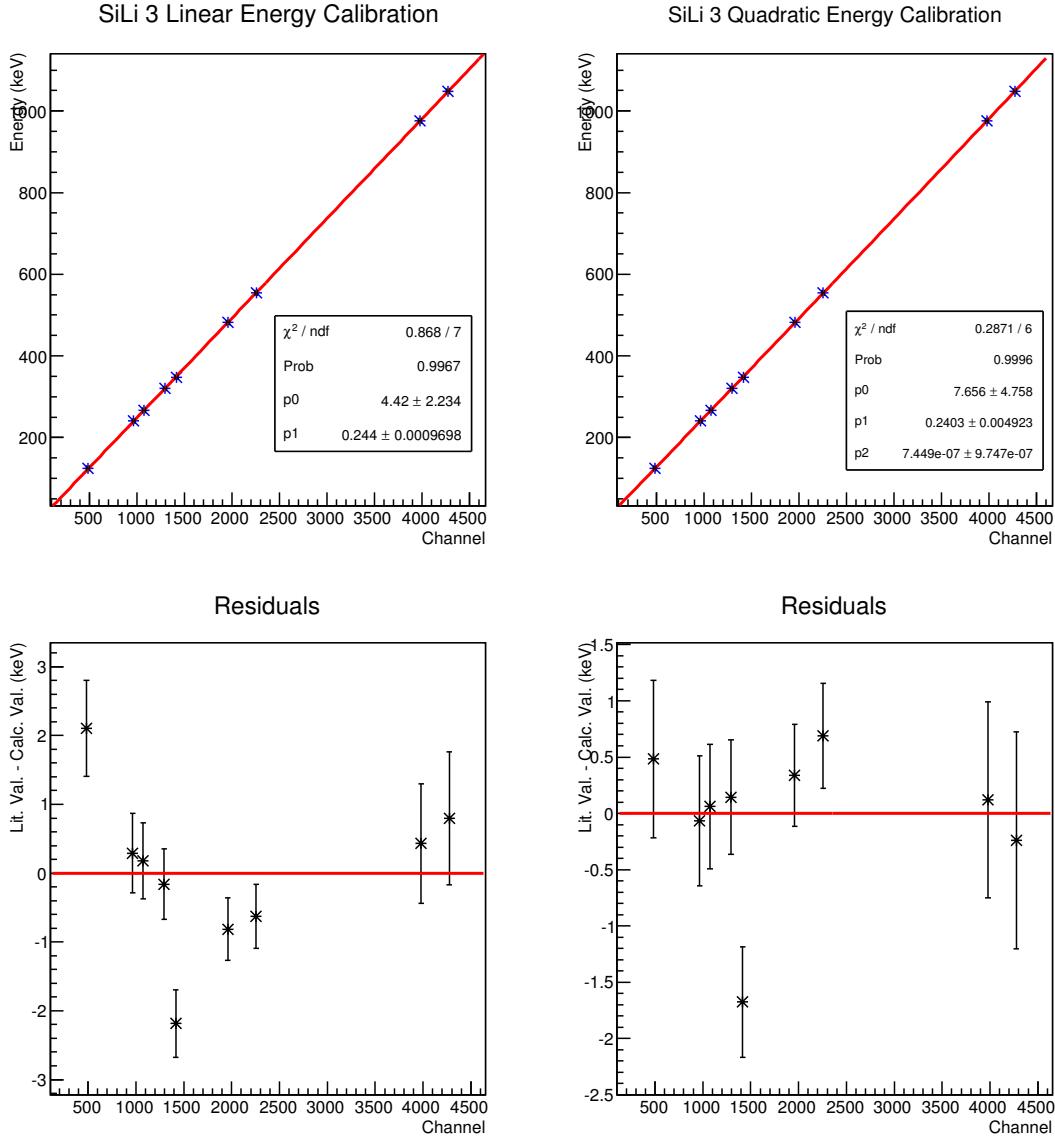


Figure 2.12. Linear and quadratic energy calibrations of one Si(Li) detector, using the electron energies listed in Table 2.7. The top graphs show the calibrations, while the graphs beneath show the respective residuals. The calibration is in much better agreement in the quadratic fit, as seen in the residuals.



Figure 2.13. An example of one of the GEORGINA detectors. The crystal is at a 90° from the cold finger, allowing the long side of the crystal to be placed next to the target, to optimize solid angle coverage. In the experiment, the circular face of the crystal was placed toward the target.

2.5.1 GEORGINA

The GERmanium detector Online aRray for Gamma ray spectroscopy in Nuclear Astrophysics (GEORGINA) is a compact array of high-purity germanium (HPGe) detectors for γ -ray experiments[28]. The detectors are 100% relative efficiency (as compared with a 3" \times 3" NaI detector at 1332 keV). These detectors were designed for use with low cross-section astrophysical capture reactions, meant to cover a large solid angle. There are a total of five detectors. The detectors were designed with the crystal at a 90° from the cold-finger, as seen in Figure 2.13.

One of the two detectors had Bismuth Germanate (BGO) detectors from Argonne National Laboratory around it to cut down on background from incomplete charge collection. BGO detectors have a high efficiency, but poor resolution, and are good to use as a veto system. In the event that a gamma-ray does not deposit its full energy in the HPGe crystal, the BGO detectors surrounding the crystal should pick up the remaining energy. This indicates the event was incomplete in energy and should be excluded from the data. This is a manual cut made during the analysis procedure by looking at the data from the BGO detectors.

2.5.2 CloverShare

CloverShare is a group of nine HPGe clover detectors with BGO shields, that originated at Yale University as part of the YRAST Ball array [6]. They were sent to various laboratories and universities for series of experiments, including the NSL. Two campaigns of experiments were run with CloverShare, each one involving an ICEBall experiment.

The CloverShare detectors are large, segmented HPGe detectors, with fitted BGO shields, seen in Figure 2.14. Each detector is segmented into four crystals, resembling a clover. Due to the "add-back" capability of these detectors, summing over the close-proximity crystals, the detectors each have a total relative efficiency $\tilde{1}50\%$, whereas each individual crystal is approximately 20% efficient. In the first of the two experiments, 7 detectors were used. In the second, 5 were used, as two experienced pre-amplifier problems during the previous campaign. The detectors used biases between -3000 and -4000 V, and were kept cold using a liquid nitrogen autofill system.

2.5.3 Calibration

Both sets of detectors were calibrated using a ^{152}Eu source, in addition to the ICEBall sources. The information about the sources is in Table 2.8. The ^{152}Eu source was not designed to sit perfectly on the target ladder, while the two ICEBall sources were. So, even though the Eu source was attached to the target ladder for calibration, using it as a measure of absolute efficiency was not possible. To use the ^{152}Eu for efficiency, a linear extrapolation of the efficiency of the 344 keV line was done using the 303 keV and 356 keV lines in ^{133}Ba . All points in the Eu were then scaled based on this. Additionally, several background lines could be used to extend the energy calibration up to 2700 keV. The energy calibration was fit to a polynomial, and fits up to the fifth order were done to find the best calibration



Figure 2.14. Example of one of the Clovershare detectors.

without overfitting. For examples of these fits, see the next chapter.

The ^{152}Eu source was placed on the individual detectors instead of centered in the trap, as it does not mount well to the target ladder. To use the ^{152}Eu for efficiency, a linear extrapolation of the efficiency of the 344 keV line was done using the 303 keV and 356 keV lines in ^{133}Ba . All points in the Eu were then scaled based on this. A characteristic fit of the efficiency of these detectors can be seen in Figure 2.15.

The efficiency calibration is fitted to equation

$$\ln(\epsilon) = a_0 - (a_1 + a_2 \times e^{-a_3 \times E}) \times E^{-a_4 \times E} \times \ln(E) \quad (2.4)$$

A characteristic fit can be seen in Figure 2.15. The CloverShare detectors were also calibrated using a ^{56}Co source that was created on the FN accelerator days before

³Measured on May-4-2012

⁴Measured on May-4-2012

TABLE 2.8
GEORGINA CALIBRATION SOURCES

Source	Activity	γ Energy (keV)	Intensity (%)
^{133}Ba	0.331(7) μCi^3	80.997	0.3406
		276.398	7.164
		302.853	18.33
		356.017	62.05
		383.851	8.94
^{207}Bi	0.306(8) μCi^4	569.702	97.75
		1063.662	74.09
^{152}Eu		121.7825	28.65
		244.6989	7.582
		344.281	26.6
		411.115	2.262
		443.965	3.125
		778.903	13.017
		867.39	4.26
		964.055	14.758
		1085.842	10.062
		1089.7	1.738
		1112.087	13.587
		1408.022	20.945

Calibration source information for GEORGINA. The energies and the respective intensities are listed for each source. Intensities are taken from [62]. The ^{152}Eu does not have activity and intensity, as it was normalized to the ^{133}Ba .

TABLE 2.9
 ^{56}CO CALIBRATION INFORMATION

Activity	γ Energy (keV)
6.44 μCi^5	846.77
	977.372
	1037.843
	1175.101
	1238.288
	1360.212
	1771.357
	2015.215
	2034.791
	2598.5

Energies used in ^{56}Co for calibration, as well as activity.

the experiment. The information about the source is in Table 2.9.

As the clovers are comprised of four separate crystals, the efficiency was taken for the individual leaves, and once the detectors had been calibrated, they were summed together. Figure 2.16 shows the comparison. As is expected, there is a small improvement in the higher energies. This is due to the summation of otherwise incomplete charge collection due to compton scattering within individual crystals. This did cause the efficiency at lower energies to appear lower, but this is due to the summing of cascades of energies over the segments. The summed crystals were used for analysis, and will be used in spectra from here on forward.

⁵Measured on Mar-16-2016

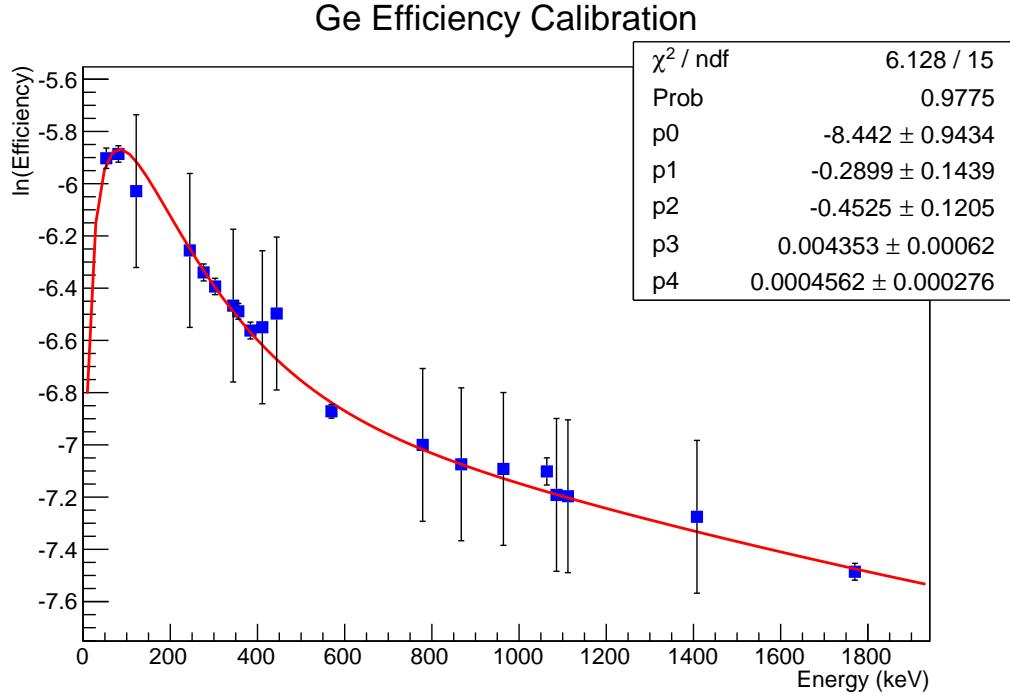


Figure 2.15. A characteristic fit of the Clovershare detector efficiencies. The points with large error bars are the ^{152}Eu points, as the scaling of the points caused a large uncertainty, compared to the ^{207}Bi and ^{133}Ba .

Each leaf of the clovers was energy calibrated. An unusual trend in the residuals was found in all cases, as seen in Figure 2.17. This trend could not be corrected for by using a higher-order polynomial, as is usually the case for integral non-linearities in multi-channel analyzers [34]. Instead, this appears to be a differential non-linearity in the electronics, resulting in discontinuities. As will be discussed in the next section, the electronics used for the experiment are not primarily used with detectors of this sensitivity.

2.6 Target Fabrication

Electrons must escape the target for detection. All materials have some stopping power for radiation, based on the material, thickness, and type of particle. The

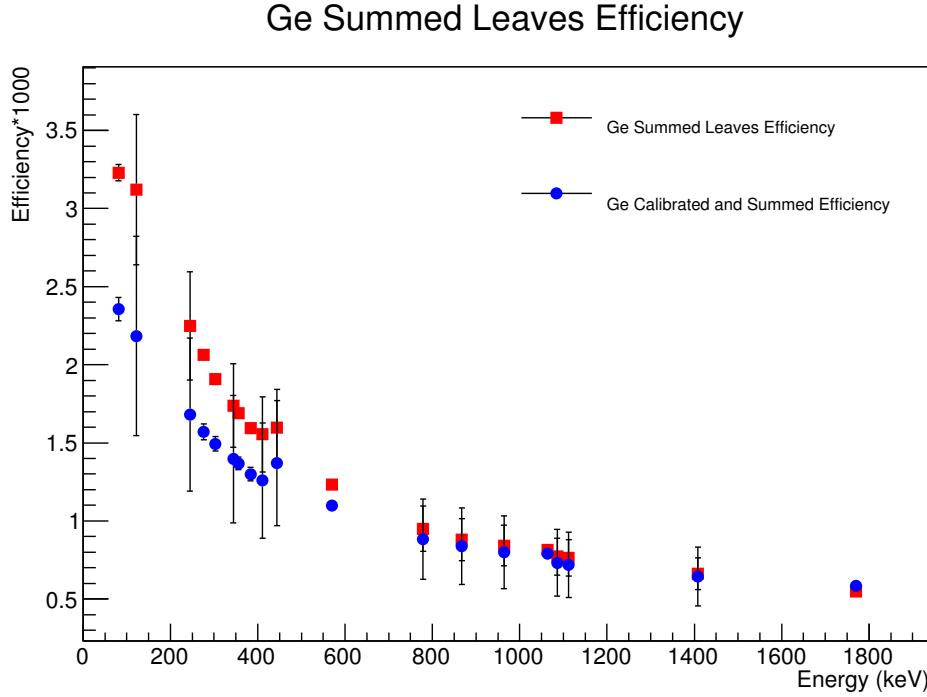
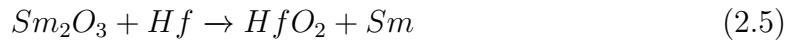


Figure 2.16. Efficiency of a CloverShare detector, with the individual leaves' efficiency summed together, compared to the efficiency of the leaves calibrated and summed together. The different colors/shapes indicate the different sources, as labeled.

Beta-Bloch formula can be used to calculate this stopping power, with several programs, including SRIM, created to make and tabulate these values[69]. A thicker target causes energy loss and straggling effects, blurring out the conversion electron spectrum, meaning a thinner target is a more ideal situation. Enriched, self-supporting samarium targets were used for this experiment. Table 2.10 has the enrichment and thickness of the targets used. The material started as Sm_2O_3 . Using the reaction



at a temperature of 1520-1820 K the samarium was extracted as a metal [20]. Once

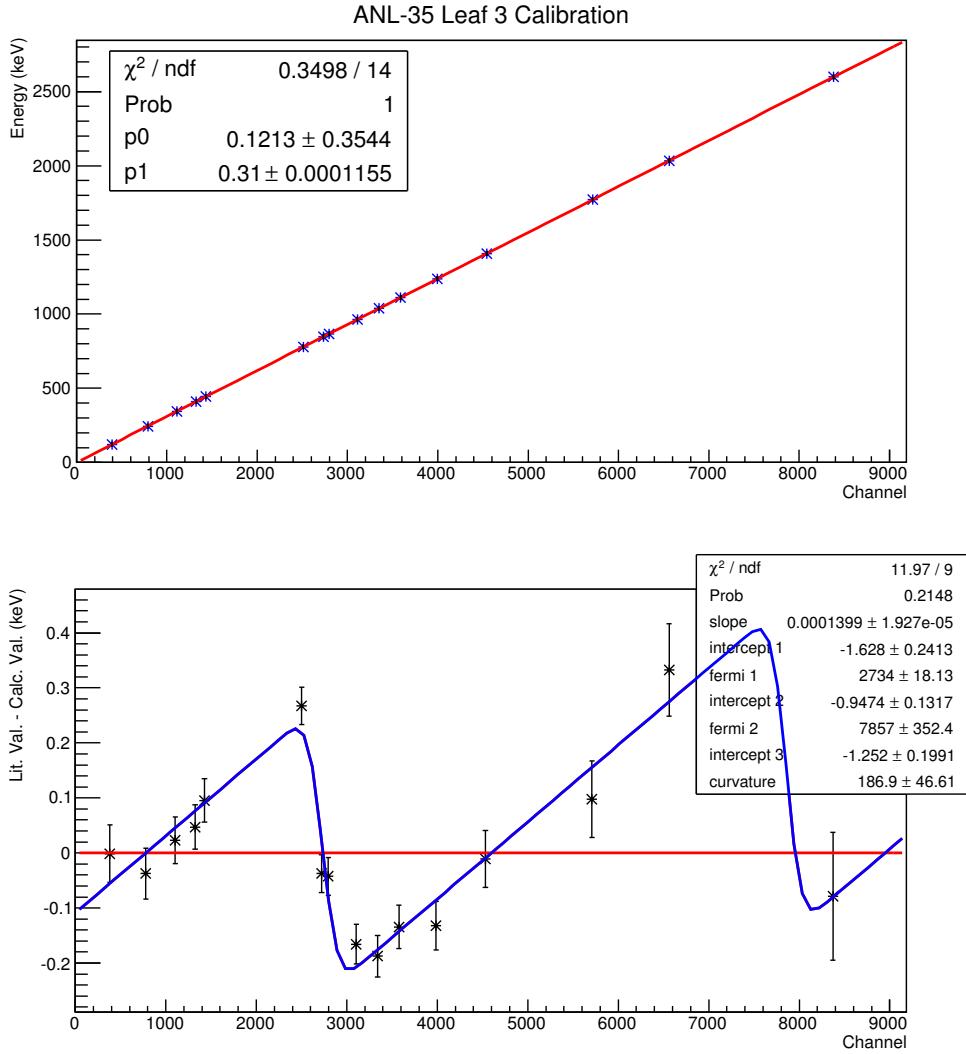


Figure 2.17. A look at the energy calibration of an individual leaf in an HPGe Clover detector. The upper graph is a linear calibration. The lower graph is a look at the residuals. For HPGe detectors, ± 0.4 keV is not a good energy calibration. Here, the sawtooth fit that was derived from the differential non-linearity in the electronics is seen fitted to the data.

TABLE 2.10
TARGET ENRICHMENT AND THICKNESS

Isotope	Enrichment (%)	Thickness (mg/cm ²)
¹⁵² Sm	> 98	1.44
¹⁵⁴ Sm	> 98	1.7

cooled, it was rolled as thin as possible. Samarium easily oxidizes, stagnating the rolling process, as too much work on the material at once could cause instantaneous oxidization, resulting in the loss of the target. Rolling had to be done incrementally, giving the material time to rest.

2.7 Experimental Configuration and Operation

2.7.1 Experimental Beam Operation

The experiments with GEORGINA were done using a bunched beam, creating a set timing signal to help with coincidence. The beam was bunched before the accelerator, with a time separation of 600 ns between bunches and a full-width half-maximum of 16 ns for the bunch. The timing of these bunches is shown in Figure 2.18. Clear separation in of the bunches is visible. The accelerator was run with a terminal voltage of 6.64 MV, with ⁴He²⁺ being sent through the machine, resulting in a beam of energy 20 MeV, including the initial 60 kV kick from the HIS.

2.7.2 GEORGINA Configuration and Electronics

While the GEORGINA detectors were designed for large efficiency up to 12 MeV, they were used in this experiment for energies up to 4 MeV. Two GEORGINA

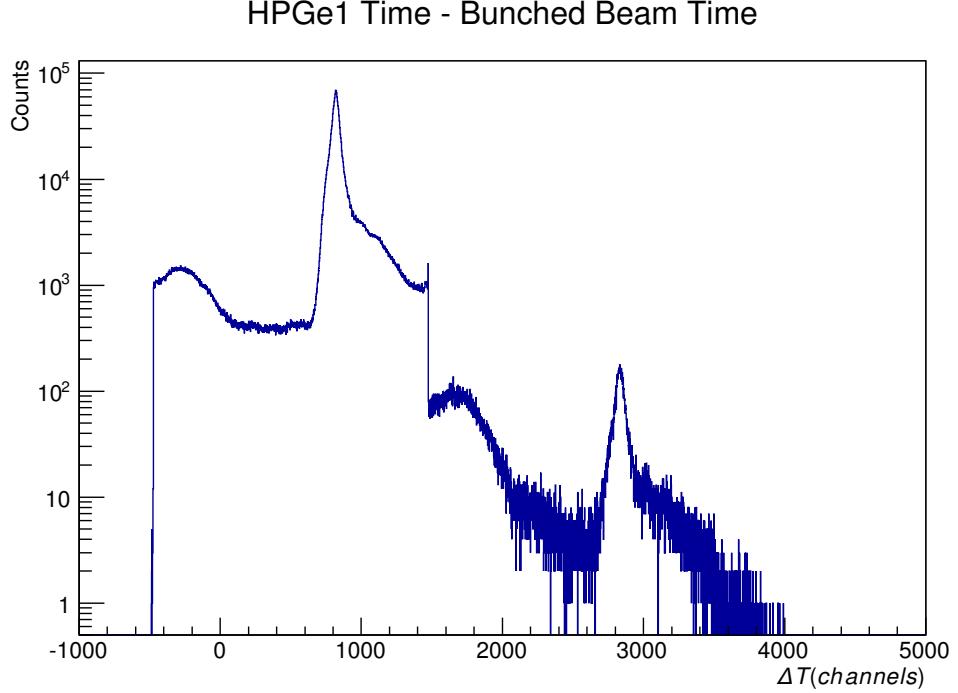


Figure 2.18. This is a plot of the HPGe 1 detector time minus the buncher time. Two distinct peaks can be seen in the spectrum, the main timing peak close to channel 800, and the much smaller secondary peak, coming from a second, later bunching signal. The bunching signal only acts as a stop signal for the timing. It can also be used as a veto, as events without a valid buncher time are not real, or are background.

detectors were used in the experiment with ICEBall. The location of the detectors is given in Table 2.11. They were placed with the face of the detector against ICEBall. The ICEBall-GEORGINA data was taken the electronics set-up from previous experiments[5]. Signals from the Si(Li), HPGe, and BGO detectors were split into two signals, one for timing and one for energy, as seen in Figure 2.19. For the Si(Li) and HPGe, the energy signals are run through an amplifier NIM (Nuclear Instrumentation Modules) before being fed into the Mesytec analog-to-digital converter (MADC-32) VME (Versa Module Europa) bus module for analog-to-digital conversion [39]. This module has 13 bit resolution. The timing

TABLE 2.11
GEORGINA DETECTOR LOCATIONS

Detector	θ	ϕ
1	0	90
2	180	90

The beam axis is the z-axis. θ is the angle in the xy-plane, where 0 degrees is beam left. ϕ is the azimuthal angle, with respect to the beam axis. All values are in degrees.

signals went through a Timing Filter Amplifier (TFA) NIM and a Constant Fraction Discriminator (CFD) NIM to clean the signal before being converted using a Caen V775 time-to-digital converter (TDC) module, with 12 bit resolution[13]. The TFA module shapes the pulse signal and amplifies it. The CFD module allows for triggering on the same part of the slope of a signal, regardless of the height of the signal, giving more consistent timing. A third module, the Caen V830 32 Channel Latching Scaler, was used to keep track of detector rates for deadtime, beam current, and trigger rates[14] via a second signal sent from the CFD module. Logic was done using a third signal from the CFD module and logic NIM modules that employ an emitter-couple logic circuit. These can be adjusted for either "and" or "or" logic. This logic created the trigger that acted as the start signal. The data structures of the events in these modules are explicitly written in Tables 2.13(a), 2.13(b), 2.13(c).

The BGO detectors went through an amplifier before being fed directly into the MADC-32. Both the Si(Li)s and the HPGe detectors went into a preamplifier before going into an amplifier, and then the MADC-32. The amplifier allowed for the adjustment of the gain, to optimize the energy regime of interest. The timing signals were sent through a TFA before going through a CFD and being delayed

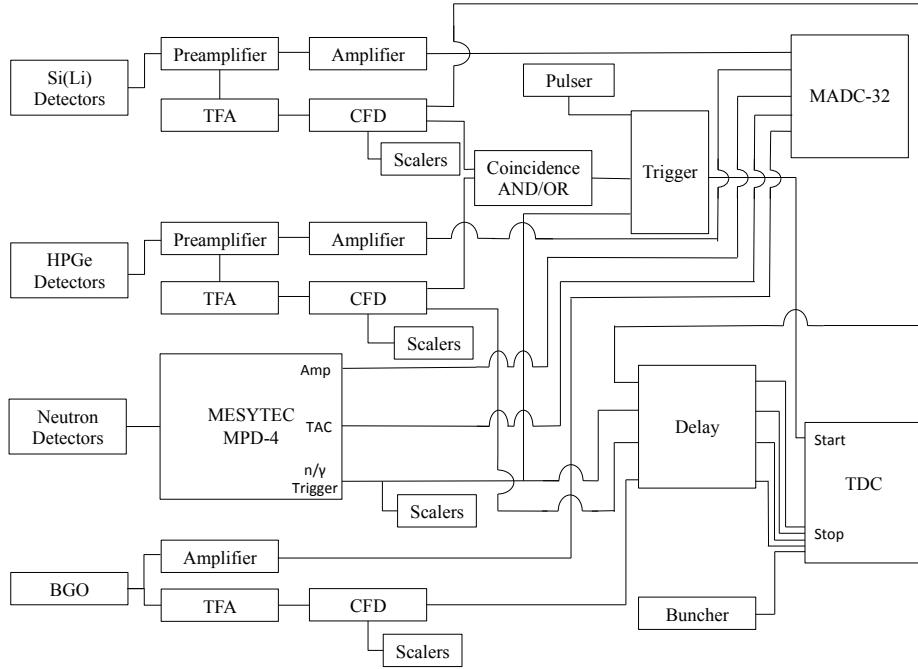


Figure 2.19. A schematic of the electronics for the ICEBall-Georgina set up. See section 2.7.2 for a detailed explanation. Taken from [5].

and sent as timing stop signals, as well as being recorded into the scalers. The stop signal was delayed by $\tilde{5}00$ ns to prevent self-triggering. Start timing signals were put into a logic coincidence for the trigger. The trigger could be adjusted if the count rate were too high, but the ideal case was the "OR" coincidence, where a Si(Li) or HPGe detector could trigger the start. When this occurred, the signal was sent to a trigger, which sent the TDC a start signal. Stop signals could come from any of the three types of detectors, or the bunched beam being used.

The data was collected from the VME modules using the Michigan State University NSCL data acquisition system (DAQ)[47]. The data file, known as an event or "evt" file, is only compatible with the analysis software SpecTcl[47]. For online

analysis, SpecTcl was used. However, for the purposes of gating and fitting, it is inadequate. These evt files were instead converted into "root" files, for use with the CERN Root Data Analysis Framework[12]. This open source software is programmable through C++, allowing for a robust set of features. This conversion was done using a program called `evt2root`[59].

2.7.3 Clovershare Configuration and Electronics

Due to the fixed nature of the ICEBall detectors, the clovers were limited in the angles they could be placed at to optimize efficiency. With the tungsten blockers in front of the Si(Li) detectors to block gamma-rays and x-rays, placing the clovers behind these detectors would drastically reduce efficiency. The system was modeled in AutoDesk Inventor [4] to visualize the detector placement, and find an ideal setup to optimize the HPGe detectors. Figure 2.20 shows the resulting model and placements, as well as several unmoveable obstructions, like cable trays, that needed to be worked around. Figure 2.1 shows the final placements for the seven-detector configuration, in the experimental hall. Table 2.12 lists the detector placements for the seven and five detector configurations, labeled as Experiments 1 and 2.

The electronics used for the Clovershare series of experiments were designed for use with the High Efficiency TOTal absorption spectrometeR (HECTOR), a NaI(Tl) detector array[53]. The HECTOR data acquisition system is based on the Michigan State University NSCL digital data acquisition system (DDAS). This system uses three XIA Pixie-16 modules [66] which are 16 channel 14-bit 100 MSPS digitizers. The data structure of this system can be seen in Table 2.13(d). Signals from detectors were fed directly from the preamplifiers to the Pixie modules, which did the amplification, shaping, timing, and logic triggers within the module through adjustments on the data acquisition computer, seen in Figure 2.21. In the ICEBall setup, these functions were done using the NIM electronics. NaI(Tl) detectors have

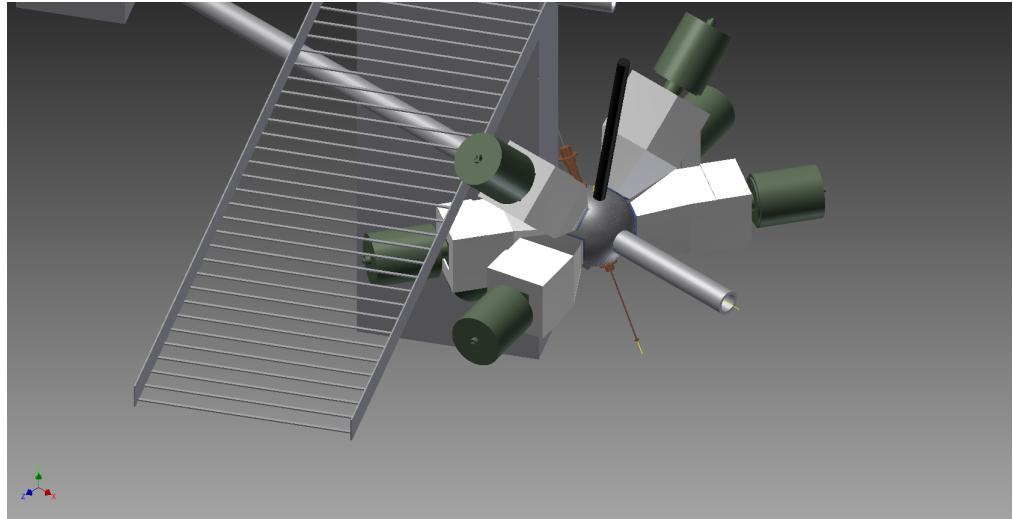


Figure 2.20. ICEBall modeled in Autodesk Inventor [4] with the Clovershare detectors placed around it in locations to optimize efficiency. Once ICEBall had been modeled, the detectors were moved around to make sure they were not behind the tungsten blockers. The beamline was also modeled to know where obstructions existed that prevent the Clovershare detectors from being placed there.

far lower resolution than HPGe detectors, so the differential non-linearities are not noticeable in the HECTOR data.

The Pixie modules were unable to produce enough gain in the amplifier section for the Si(Li) detectors to cover a significant number of channels for resolution, so the detectors' signals were put through fast filter amplifiers [2] to boost the signal before being fed into the DDAS. Fast amplifiers were used due to the fast timing and acquisition nature of the Pixie-16 modules and DDAS.

As with the ICEBall data, these files were event files that were then converted into root files for further analysis.

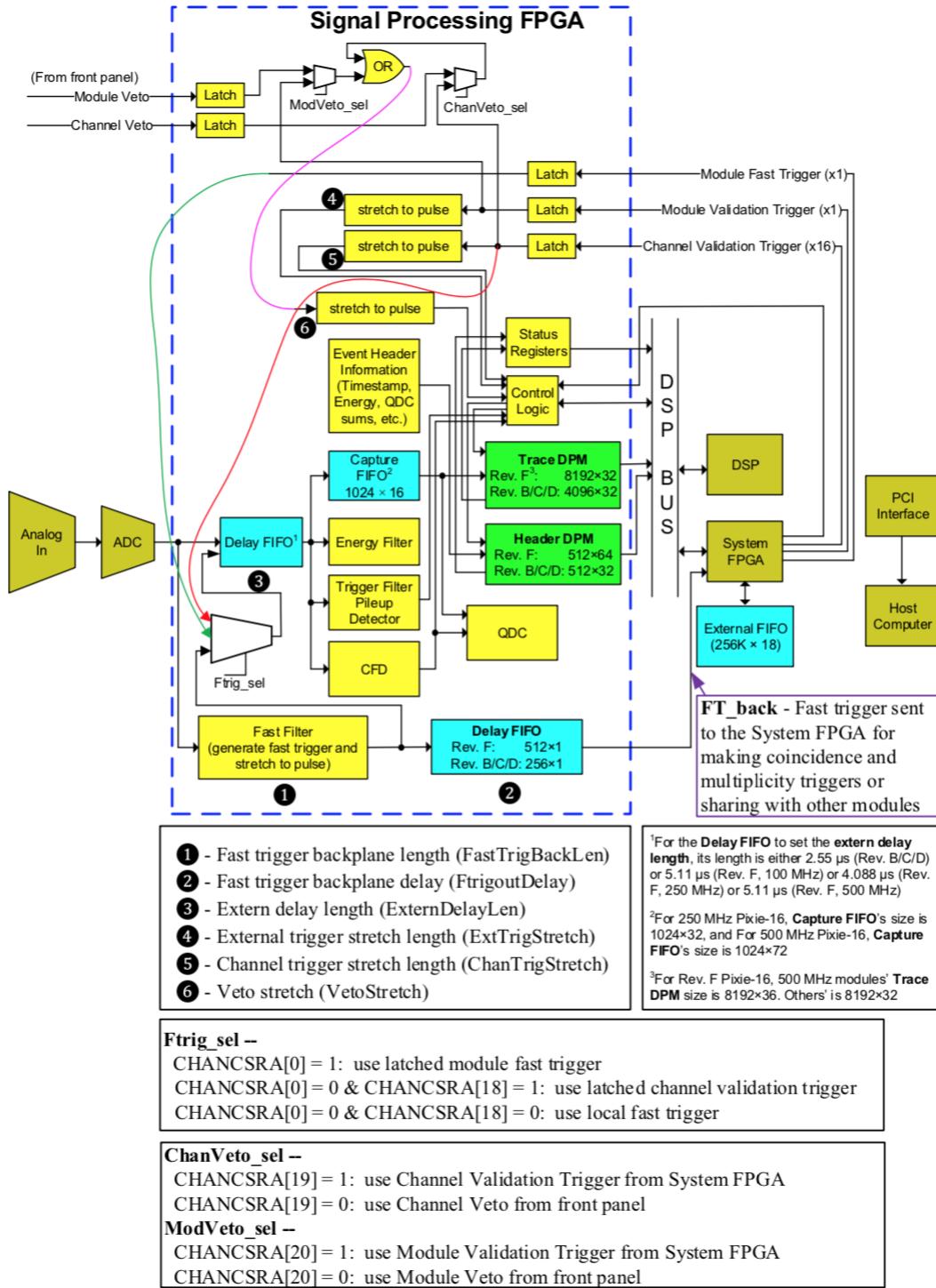


Figure 2.21. The signal processing done by the XIA Pixie-16 electronics.
 All the processing is done within the module. Taken from [66].

TABLE 2.12
CLOVERSHARE DETECTOR LOCATIONS

	Experiment 1		Experiment 2	
Detector	θ	ϕ	θ	ϕ
ANL-35	270	145	270	145
LBL-7	0	111	0	111
ANL-23	0	66	Not Used	
LBL-6	180	111	Not Used	
LBL-10	180	66	180	71
ANL-31	45	90	45	90
ANL-18	135	90	180	111

The detectors are listed by identification name. The angles for the detectors are given for both experiments run with the Clovershare detectors. ANL-23 and LBL-6 were not used in the second experiment due to preamplifier issues. The beam axis is the z-axis. θ is the angle in the xy-plane, where 0 degrees is beam left. ϕ is the azimuthal angle, with respect to the beam axis. All values are in degrees.

TABLE 2.13
DATA STRUCTURE OF ELECTRONICS

(a) Data event - MADC32 (32 bit word)

Header						
2 header signature	6 subheader	8 module id	1 output format	3 adc resolution	2	10 number of data words
b01	b000000		bx	bxxx	b00	number of 32 bit data words
Data Word						
2 data-sig	9	5	1	1 out of range	1..3	11..13
b00	00 0100 000	channel number	b0	Oor	b00	ADC Amplitude
End of Event						
2				30		
b11				event counter/time stamp		

70

(b) Data event - V775 (32 bit word)

Header							
5	3	8	2	6			8
Geo Address[5]	010	Crate Number[8]	00	Converted Channels[6]			
Data Word							
5	3	3	5	1	1	1	12
Geo Address[5]	000		Channel Number	Valid	Under-Threshold	Overflow	ADC Amplitude[12]
End of Block							
5	3					24	
Geo Address[5]	100					Event Counter[24]	

(c) Data event - V830 (32 bit word)

Header					
5	1	2	6	2	16
Geo Address	1		Number of Enabled Channels	Source Source	Trigger Number
Data Word					
32					
Channel Counter[32]					

(d) Data event - Pixie-16 (32 bit word)

Index	Header						
0	1	14	5	4	4	4	
	Finish Code	Event Length	Header Length	CrateID	SlotID	Chan#	
1			32				
			EVTTIME_LO[32]				
2	1	15		16			
	CFD forced trigger bit	CFD Fractional Time[15] x 32768		EVTTIME_HI[16]			
3	1	15		16			
	Trace Out-of-Range Flag	Trace Length		Event Energy			
Data Word							
n	16		16				
	ADC Data (2n+1)[16]		ADC Data (2n)[16]				

Tables of the 32-bit word data structures of the different electronics used in the experiments, including the MADC (a), the CAEN 775 TDC (b), CAEN V830 Scaler (c) and the Xia Pixie-16 (d). The headers, event word structure, and end of event structure, where needed, are all listed. In the tables, the first row in a given block is the number of bits used by that piece of data, while the second row is a description of the data. Bits go in descending order from left to right.

CHAPTER 3

ANALYSIS

The acquisition of the data, as discussed in the previous chapter, was done using the NSCL DAQ [47, 49]. These files were converted into files compatible with ROOT [12]. In ROOT, calibrations, gates, and fitting were done. Fits of the peaks were also done in RADWARE [50] and compared. RADWARE was used as the main fitting tool, as the built-in framework for the Fano factor [23] created more consistent global fits, and a better estimate of the skew factors for the conversion electron spectra. Appendix C contains the analysis code used, and Appendix D contains the ROOT macros discussed in the text.

3.1 Fitting Spectra

Whenever possible, all data were fit using the same methodology. If another methodology needed to be used (as will be discussed in Section 3.4) the methodology was tested. This was done by taking both fitting routines to the same peaks and comparing the results. If they were in agreement, the secondary method could be used where the primary method failed.

In a completely ideal scenario, the spectrum from an experiment should look like a series of delta functions. This is never the case in a real experiment, as detectors do not have infinite resolution, and do not cover an infinitesimal angle. Instead, an ideal experimental spectrum would have the peaks look like gaussian functions. The

normal gaussian formula is

$$f(x) = h e^{-\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)^2} \quad (3.1)$$

where h is the height of the peak, μ is the centroid of the peak, and σ^2 is the variation of the peak. This function can be renormalized using the integral to give the area as a fitting parameter instead of the height as

$$A = h\sigma\sqrt{2\pi} \quad (3.2)$$

The HPGe spectra were fitted using the gaussian formula. Fitting multiple peaks at once in a spectrum requires a secondary consideration: the Fano factor. The Fano factor is a measure of dispersion within a detector[23]. It takes into account that the energy loss in the detector is not purely statistical, as the gaussian distribution would assume. This goes into the resolution of the detector, which is related to the energy of the particle being detected. As a result, the resolution changes with energy. This is reflected in the variation (σ^2) in the fit. To do this in ROOT, the fitting function must be written to reflect this information, tying the width of every peak together under certain assumptions. As there is not an iterative way to create such a function in ROOT for x number of peaks, an individual fitting function must be written for each number x of peaks to fit. This becomes exceedingly tedious. However, RADWARE has this energy adjustment built into the fitting code, allowing the correlation between peaks to be turned off if needed for up to 35 peaks. If there are background peaks that should not have a correlation, such as the case of a Compton peak, it can be turned off for individual peaks.

In the previous chapter, the results of the calibrations were discussed, without going into detail into the calibrations themselves. In particular, the calibrations of the Si(Li) detectors were important for the fitting of data peaks. Due to incomplete

charge collection, the Si(Li) detectors have a pronounced low-energy tail that can be modeled using a skewed gaussian. This skewed gaussian formula adds two more fitting parameters: R and β . R is the ratio of the tail height to the peak height, and β is a "skewedness" parameter. Both of these values are floated during the calibrations to find optimal values, and then fixed from these values for the data fits. The skewed gaussian formula is

$$f(x) = h \left(1 - \frac{R}{100} \right) e^{-\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)^2} + h \left(\frac{R}{100} \right) e^{\frac{x-\mu}{\beta}} \operatorname{erfc} \left(\frac{x-\mu}{\sqrt{2}\sigma} + \frac{\sigma}{\sqrt{2}\beta} \right) \quad (3.3)$$

with the variables as previously discussed. Of note, R can only range from 0 to 100, and β must be non-zero when used. In general, β should be positive. Similar to the Fano factor, R and β are related to the detector. R may vary slightly with energy, but can be held fixed, while *beta* is usually fixed for the detector.

A third component, a smoothed step function, can also be used to increase the background on the low energy side of the peak that occurs from the Compton scattering of photons into the detector. This adds one more fitting parameter, the step height, to the fitting function, that would need to be fixed from the calibration data. Fits were done with and without this floating in the calibration data. It made no impact, and in some cases, found a negative step height, which is unphysical. The step height was therefore fixed to zero.

The peak fits in RADWARE were used to get the efficiency parameters for the functions 2.3 and 2.4. The parameters for the ICEBall-GEORGINA set up are in Tables 3.1 and 3.2. The parameters for the ICEBall-Clovershare runs are in Tables 3.3, 3.4 and 3.5. The Si(Li) efficiencies did not change between the two Clovershare runs, but the HPGe detectors were changed and moved, resulting in different values.

TABLE 3.1
SI(LI) EFFICIENCIES IN GEORGINA CONFIGURATION

Detector	p_1	p_2	p_3
1	-9.753	1.127	-0.003429
2	-20.33	3.155	-0.006227
3	-15.96	2.309	-0.004813
4	-8.542	0.9793	-0.003405
5	-14.39	2.186	-0.006035
6	-11.41	1.485	-0.003527

TABLE 3.2
GEORGINA EFFICIENCY PARAMETERS

Detector	a_0	a_1	a_2	a_3	a_4
0	-8.247	-0.4383	-0.2526	0.003371	0.000269
1	-8.581	-0.5057	-0.3439	0.003742	0.000235

TABLE 3.3
SI(LI) EFFICIENCIES IN CLOVERSHARE CONFIGURATION

Detector	p_1	p_2	p_3
1	-11.39	1.424	-0.003961
2	-17.75	2.542	-0.003829
3	-14.55	2.101	-0.004718
4	-20.39	3.16	-0.005671
5	-16.79	2.426	-0.004415
6	-15.88	2.331	-0.004863

TABLE 3.4
CLOVERSHARE MARCH RUN EFFICIENCY PARAMETERS

Detector	a_0	a_1	a_2	a_3	a_4
0	-8.193	-0.3404	-0.4493	0.005554	0.0007445
1	-7.43975	-0.289389	-0.374205	0.00649538	0.000929357
2	-8.317	-0.2887	-0.5504	0.008845	0.0008342
3	-9.611	-0.4577	-0.5905	0.004659	0.0004296
4	-7.863	-0.327	-0.447	0.008795	0.00098
5	-8.297	-0.3503	-0.4463	0.006396	0.0007243
6	-8.225	-0.3443	-0.4478	0.005464	0.0007393

TABLE 3.5
CLOVERSHARE MAY RUN EFFICIENCY PARAMETERS

Detector	a_0	a_1	a_2	a_3	a_4
0	-8.185	-0.3043	-0.4174	0.004383	0.0006837
1	-7.286	-0.3525	-0.334	0.007523	0.001481
4	Not calculated due to detector malfunction				
5	-7.724	-0.2955	-0.2346	0.004545	0.0001513
6	-15.55	-1.496	-1.132	0.004681	0.0001785

3.2 Gating

Two types of gates were used in the analysis: energy and timing. Energy gates were used to look for coincidence with specific transitions. This allowed for confirmation of excited state population using known level schemes. Generally, the energy gates were used on the gamma-ray spectra, defined using the centroid of the gamma and a range of $\pm 2\sigma$. Only one HPGe detector would be gated on so both gamma-ray and conversion electron spectra could be viewed with the same constraints. The Si(Li) detectors were gated on, but both due to the low-energy skewed tails and the lower resolution of the Si(Li) detectors, clean gates were not achievable.

The timing gates were unique to pairs of detectors. In the GEORGINA data, a pulsed beam was used, allowing for a distinct timing structure, shown in Figure 2.18 from the previous chapter. As is seen, the structure is complicated. By using an energy gate, the background is cut down, and the timing structure becomes much clearer, as seen in Figure 3.1. There is an energy dependence, also shown in the figure. Thus, the timing gates were made to cover the full range of possibilities. A second timing gate, covering a different region of the same time length, is used to

subtract off the background the timing gate does not exclude. Table 3.6 summarizes the gates for the different detector pairings. In some cases, the background gates needed two sections, to cover the same range as the original timing gate.

TABLE 3.6
TIMING GATES BY DETECTOR PAIRING FOR THE
ICEBALL-GEORGINA EXPERIMENT

Detector 1	Detector 2	Main Gate		Background Gate	
		Start	End	Start	End
HPGe-HPGe Pairings					
0	1	-1500	1000	-2500	-1500
				1000	2500
HPGe-Si(Li) Pairings					
0	0	-1500	0	0	1500
0	1	-600	1000	1000	2000
				-1600	-1000
0	2	-400	1000	1000	2400
0	3	-500	1000	1000	2500
0	4	-200	1100	-1300	-200
0	5	-700	1000	1000	2000
				-1400	-700
1	0	-1200	1200	-2400	-1200
				1200	2400
1	1	-600	1500	1500	2100
				-2100	-600
1	2	-200	1400	1400	1600

TABLE 3.6 (CONTINUED)

Detector 1	Detector 2	Main Gate		Background Gate	
		Start	End	Start	End
1	3	-500	1400	-1600	-200
				-1900	-500
	4	-100	1600	1400	1900
1	5	-600	1500	-1700	-100
				1600	1700
				-2100	-600
				1500	2100
Si(Li)-Si(Li) Pairings					
1	2	-200	200	200	600
1	3	-200	200	200	600
1	4	0	200	200	400
1	5	-600	600	600	1800
1	0	-1200	200	200	1600
2	3	-100	100	100	300
2	4	0	200	200	400
2	5	-600	200	200	1000
2	0	-1200	200	200	1600
3	4	0	200	200	400
3	5	-600	400	400	1000
3	0	-1200	200	200	1600
4	5	-700	100	100	900
4	0	-1200	100	100	1400
5	0	-1200	200	200	1600

TABLE 3.6 (CONTINUED)

		Main Gate		Background Gate	
Detector 1	Detector 2	Start	End	Start	End

A table of the timing gates used in the GEORGINA experiment. Detectors are indexed as the code read them in. The indexes worked in both orders, to avoid redundancy issues, i.e. (0,1) and (1,0) are the same as far as the code is concerned. All start and end values are channel number, and in several cases, there are two background gates, due to the size of the main gate and the available channel range.

In the Clovershare data, the timing was done within the electronics, leading to an clean structure, seen in Figure 3.2. Although the same code structure was used for the timing, the gates were all the same within the Clovershare data. The main gate used was ± 0.3 and the background ranged from 0.3 to 0.9, using the same scaling as seen in Figure 3.2.

3.3 Systematic Effects

Several systematic effects had to be taken into account at various stages of analysis. Each will be discussed in further detail in this section. At the calibration stage, there are two major corrections in the Clovershare data. The first, as discussed in section 2.5.3, is due to the integral non-linearities of the electronics. Second, the instability of the Clovershare detectors required a run-by-run calibration correction. In the analysis stage, corrections on extracted values needed to be done based on the angular correlations due to the detectors being at different angles relative to each other. If the angle between the two HPGe detectors were the same as the angle between the HPGe detector and Si(Li) detector(s), this correction would divide out and be unneeded.

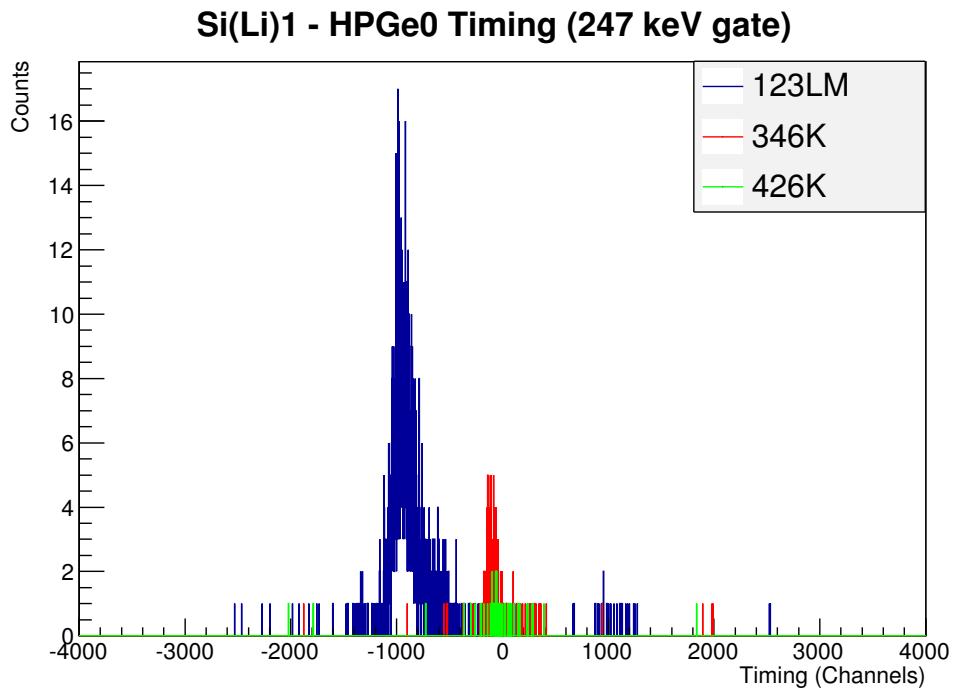


Figure 3.1. A plot of the energy-gated timing for a HPGe-Si(Li) detector pairing. The HPGe detector was gated at 247.9 keV, one of the ground state band transitions in the ^{145}Gd nucleus. The conversion electrons for other transitions in the ground state band were gated on in the Si(Li) detector. Plotted here are the time differences between the two detectors when both peaks were seen in the respective detectors. The 123LM peak was used instead of the K peak due to the high threshold of this detector. As can be seen, the timing has an energy dependence. This dependence is most prominent at lower energies.

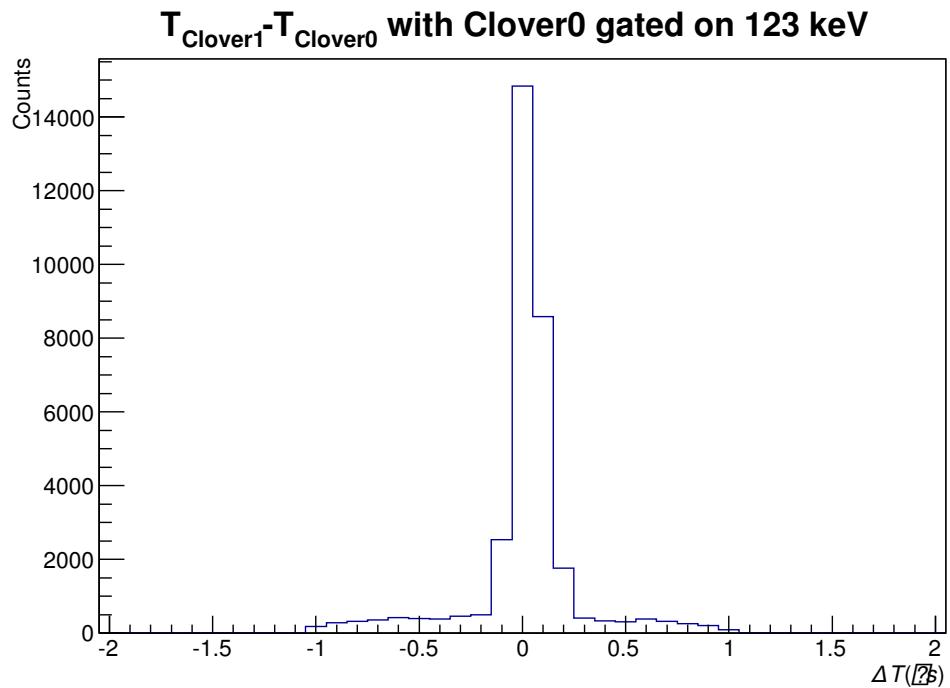


Figure 3.2. Timing between two of the Clovershare detectors, with one detector gated on 123 keV, the lowest transition in the ground state band of ^{154}Gd . In this data, the gates all looked similarly symmetric, and the timing gates did not need to be varied as they had for the GEORGINA set up.

3.3.1 Electronics-based Integral Non-Linearities

There are two kinds of non-linearities from electronics: integral and differential [34].

These two non-linearities are interrelated. Integral linearity is most easily seen in the energy calibration. Differential non-linearity can be seen by looking at the count rate with respect to channel number. Generally, the non-linearity can be expressed in calibration by a higher order correction to an otherwise linear calibration. In the GEORGINA detectors, this was the case. In the Clovershare detectors, this was not the case.

The Pixie-16 MCA used in the Clovershare experiments has non-linearity specifications for 12-bit and 14-bit modes, but not 16-bit, according to the XIA datasheet [66]. These are listed in terms of the difference of the least significant bit (LSB). In an ideal ADC, this number is 0, meaning the difference between two adjacent channels is exactly one LSB. A larger number means the gap is that much greater than one, and a negative number is that much less. When such non-linearities exist, a best case scenario is for the difference to be constant, leading to the polynomial correction. In the case of the Clovershare experiments, the LSB difference between channels does not appear to be constant.

Plotting the residuals of the calibration after doing a linear fit shows what appears to be a sawtooth pattern to the energy points, as seen in Figure 2.17. To fit this sawtooth pattern, it was assumed that the upward slope of the various sections was the same, and the sections were connected via the error function and complimentary error function, to allow for smoothing of the discontinuity. This results in the function

$$E_{res} = m * x + b_1 * \operatorname{Erfc}\left(\frac{x - f_1}{c}\right) + b_2 * \operatorname{Erf}\left(\frac{x - f_1}{c}\right) * \operatorname{Erfc}\left(\frac{x - f_2}{c}\right) + b_3 * \operatorname{Erfc}\left(\frac{x - f_2}{c}\right) \quad (3.4)$$

where m is the slope, b_i are the various intercepts to shift the correction, f_i is the location of the shift, and c is the curvature of the error function.

3.3.2 Run-based Calibration Corrections

Run-by-run corrections are usually needed when there is drift or instability in the electronics or pre-amplifier. In the Si(Li) and GEORGINA detectors, there was no noticeable drift. The Clovershare data does appear to need a correction for the HPGe detectors. The best way to track these instabilities was to look at known lines in each run, and plot the residuals after calibration, compared to the known value. This can be seen in Figure 3.3. Several well defined peaks were taken in run to define a linear correction for each run. These linear corrections were done on top of the original energy calibration and correction.

The peaks used for the run-by-run correction were a combination of the ground-state band peaks in the isotope of interest at low energies, and background gammas at higher energies, such as 1460 keV from ^{40}K and 1764 keV from ^{214}Bi . These are naturally occurring from the concrete walls of the NSL.

3.3.3 Angular Correlations

Angular correlations are a well studied phenomenon that arises when trying to look at two types of radiation in coincidence. A thorough exploration of different radiation pairings is explored in [8]. For this work, the pairings of interest are $\gamma - \gamma$ and $\gamma - e$. Triple correlations were also explored, as there were several cases where the intermediate transition was not seen.

Because the detectors are at different angles with respect to each other, when computing conversion coefficients, or subtracting off conversion electrons from the ground state band within the spectra, a correlation must be made based on these relative angles. Table 3.7 lists these angles from the ICEBall-Georgina setup, with

Clover 0 Residuals by Run

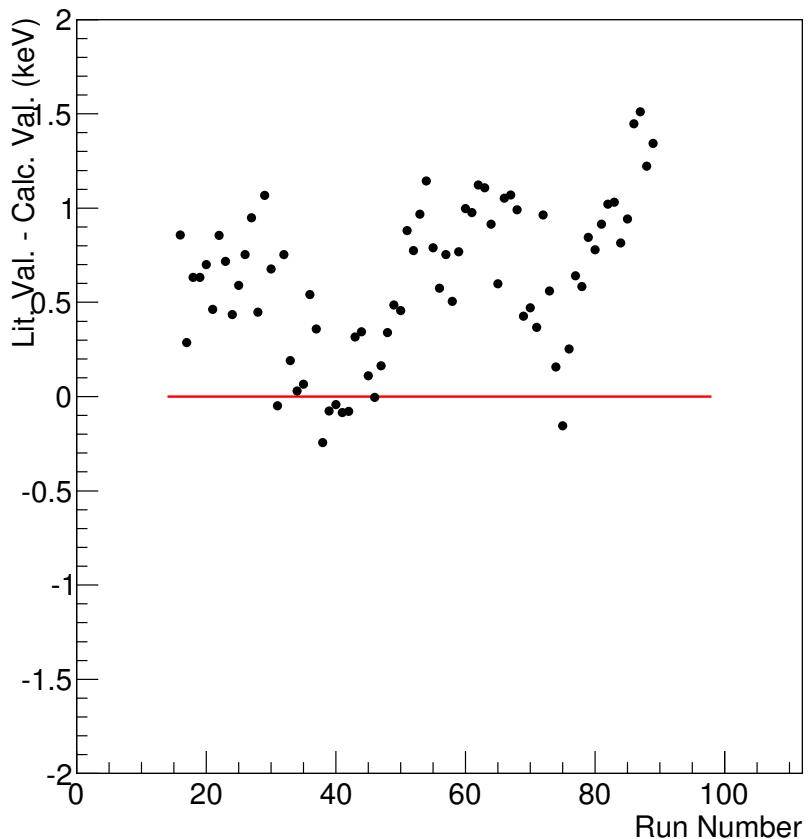


Figure 3.3. The difference between the literature value and the calibrated value (before run-by-run correction) of the naturally occurring ^{40}K background peak at 1460 keV, plotted by run for one leaf of one HPGe detector. The red line shows $\Delta E = 0$.

TABLE 3.7
RELATIVE ANGLES OF DETECTORS

Detector	θ'
HPGe 2	180
SiLi 1	90
SiLi 2	90
SiLi 3	122.5
SiLi 4	110.7
SiLi 5	69.3
SiLi 6	57.3

Relative angles of the detectors with respect to HPGe 1 in the ICEBall-GEORGINA set up. The angles were calculated using Tables 2.4 and 2.11 and equation 3.5. All angles are in degrees.

respect to Detector 1.

These angles can be calculated using

$$\theta' = \cos^{-1}\left(\frac{\sin\theta_1\cos\phi_1\sin\theta_2\cos\phi_2 + \cos\theta_1\cos\phi_1\cos\theta_2\cos\phi_2 + \sin\phi_1\sin\phi_2}{\sqrt{2}}\right) \quad (3.5)$$

with the previous definitions given for the angles in Tables 2.4 and 2.11.

Angular correlation functions for $\gamma - \gamma$ are usually written as a series of Legendre polynomials, multiplied by a correlation coefficient A_ν , or

$$w(\beta) = \sum_{\nu} A_{\nu} P_{\nu}(\cos\beta) \quad (3.6)$$

In the case of $\gamma - e$ correlations, the equation becomes

$$\begin{aligned} w_m(\beta) &= \sum_{\nu} b_{\nu}(LLm) A_{\nu} P_{\nu}(\cos\beta), \\ w_e(\beta) &= \sum_{\nu} b_{\nu}(L+1, L+1, e) A_{\nu} P_{\nu}(\cos\beta) \end{aligned} \quad (3.7)$$

depending on if the radiation is electric or magnetic in nature. The b_{ν} values are based off the same matrix elements used to calculate theoretical conversion coefficients, and are well understood[55, 56].

Both of these cases assume pure multipole transitions. In the case of mixed transitions, the correlation function becomes

$$W = w_I + \delta^2 w_{II} + 2\delta w_{III} \quad (3.8)$$

where w_I and w_{II} are the pure multipole correlation functions, and w_{III} is a correlation mixing the two multipoles. The variable δ is known as the mixing ratio. These correlations can be further extrapolated to look at cases not just of direct correlations, but of indirect correlations, such as three γ -rays in cascade. If the middle radiation is not observed, the first and third radiations still have a determinable correlation. This is derived for pure multipoles by [8] and for mixed transitions by [48, 54].

For the experiments in this text, the ratios of these angular correlations at the detectors respective angles is needed to adjust for angular effects. Thus, the ratio becomes a factor, C_{\angle} , the ratio of the Si(Li) detector angle to the HPGe detector angle with respect to the detector gated on. This factor is divided by to correct the conversion coefficient. In the singles, a similar value, based on the angular distribution instead of the angular correlation, must be used for correction.

3.4 Upper Limits

In ^{156}Gd , there were several possible transitions of interest at energies less than 250 keV. In that energy range, the ground-state band transitions cover much of the spectrum. Some transitions could be removed through the use of gating on gamma-rays of parallel transitions, but not all the transitions could be removed. This was due, in part, to the gate transitions being in sequence with some of the ground-state band.

To get an upper limit for the transitions of interest, the ground state band transitions that were in sequence with the gate transition were subtracted off of the conversion electron spectrum through the following method. An example of this subtraction can be seen in Figure 3.4. The skewed gaussian fitting function described earlier was given parameters derived from the data and subtracted from the conversion electron spectrum.

The area of the skewed gaussian is obtained by getting the area of the gamma peak, and adjusting by several factors: efficiencies (ϵ), conversion coefficient (α , as derived from theory), and correlation coefficients (W). This gives the following:

$$A_{ce} = A_\gamma \times \frac{\epsilon_{ce}}{\epsilon_\gamma} \times \alpha \times \frac{W_{ce}}{W_\gamma} \quad (3.9)$$

The height of the skewed gaussian can be derived in terms of the area, R , σ and β to be

$$H = A \frac{100}{2 * e^{-\frac{\sigma^2}{2\beta^2}} R\sigma - \sqrt{2\pi}(R - 100)\sigma}; \quad (3.10)$$

These three variables are all unique to the detector, and come directly from fitting the calibration data, as previously discussed. From there, the parameters can be fed into the equation, along with the centroid of the peak, and subtracted directly off the spectrum. Any remaining counts are counted by taking sections on either side

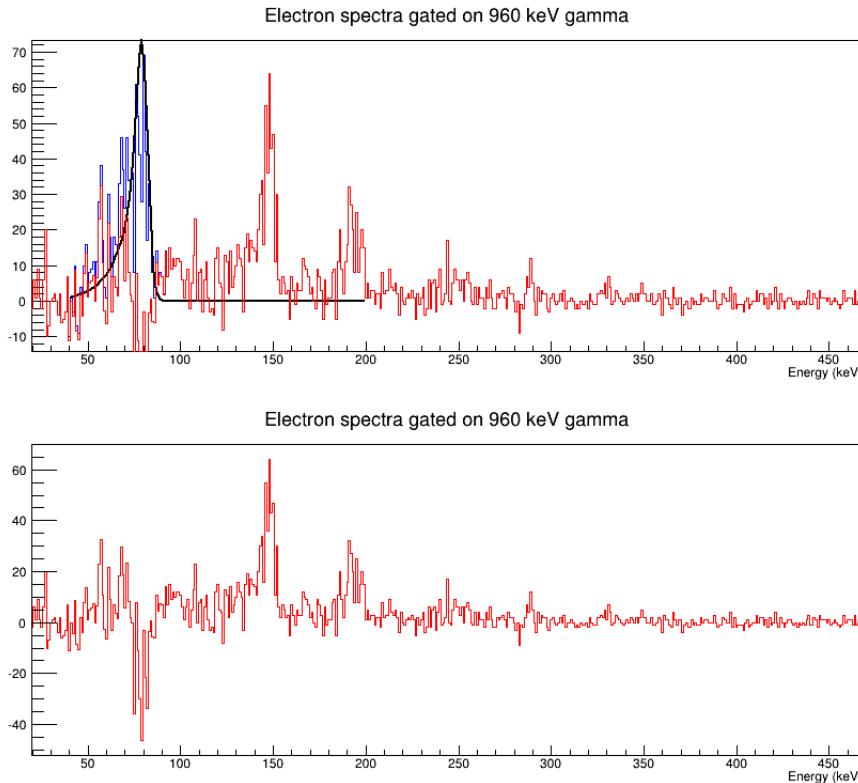


Figure 3.4. An example of the subtraction used to removed the ground state band peaks from spectra. To subtract the conversion electron peak off, the area of the corresponding gamma peak was taken from the same gate. Using the conversion coefficient and the efficiencies of both detectors, the area of the peak could be calculated. From there, this value, along with fixed R , β , σ , and centroid were used to subtract the peak off. The black line is the function used for subtraction. The blue line is the original spectrum, and the red line is after subtraction. This code can be found in Appendix D

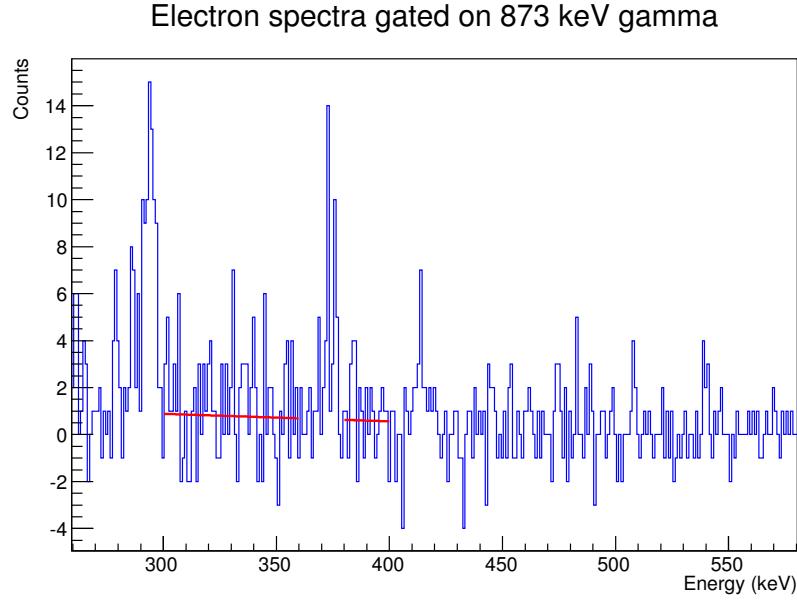


Figure 3.5. An example of the global background fit used for peaks that could not be fit using the skewed gaussian function. The areas on either side of the peak are selected by the user. Once the background fit is done, the central area between the two sections is treated as the peak and the background is subtracted off to give a peak area. The red line shows the fit for the background area, and the two sections used for the background fit.

See D for the global-fit function used.

of the area of interest and using a global linear fit for the background, as seen in Figure 3.5. This linear fit is subtracted off bin-by-bin in the region of interest. The remaining area is then taken as an upper limit on the transition of interest.

3.5 Error Analysis

When areas of peaks were found, the error is found with the fit. In the cases of the peaks that needed to be fit by estimating the background and subtracting it off, the error of the area is assumed to be the square root of the level.

These areas are propagated through calculations via the equation

$$\sigma_f^2 = \sum_i \left(\sigma_{x_i} \frac{\delta f}{\delta x_i} \right)^2 \quad (3.11)$$

where f is a function of all x_i in the sum [7]. If the function is related to all variables via some form of a polynomial, i.e. $(x_i)^a$ where a is a constant, then it can be rewritten as

$$\sigma_f^2 = f * \sum_i \left(\frac{\sigma_{x_i}}{a * x_i} \right)^2 \quad (3.12)$$

Errors on the areas of both gamma-rays and electrons were propagated through the data. The error on the efficiency was estimated by shifting the efficiency value $\pm 0.5\sigma$ and calculating upper and lower limit functions for the efficiency.

In the final results, the uncertainties were kept separate into statistical and systematic effects. Statistical effects are caused by the areas of the peaks.

Systematic effects are from the uncertainty in the efficiency of both the HPGe and Si(Li) detectors, as well as an uncertainty in the angular correlations, based on the solid angle covered by the detectors.

CHAPTER 4

^{154}Gd RESULTS

^{154}Gd is a transitional nucleus with a large number of 0^+ states, 16, and high number of levels in the nucleus. The nature of these 0^+ states is of interest for this study. The lowest lying excited 0^+ state in the nucleus has an anomalously large transition to the ground state, as compared with other nuclei in the region, shown in Figure 4.1. For this 0^+ state, there is a question of the state representing a β -vibration, or being a sign of shape-coexistence in the nucleus.

The ^{154}Gd data was taken across three separate runs; one with ICEBall and GEORGINA, and two with ICEBall and Clovershare, in two different configurations, as discussed in Chapter 2. The GEORGINA experiment allowed for the HPGe detectors to be closer to the target, at a distance of 30 cm. This experiment yielded the most conversion electron data, and results in this chapter are quoted from that data. Although the Clovershare detectors has to be placed 5-10 cm farther from the target, the larger number of detectors, 7 for the first run and 5 for the second, the $\gamma - \gamma$ coincidence data was better, so identification of transition placement and level population was confirmed. All of the experiments used the same enriched ^{152}Sm target of 1.44 mg/cm^2 thickness, discussed in Chapter 2 and shown in Table 2.10. A complete catalog of spectra used in analysis can be found in Appendix A.

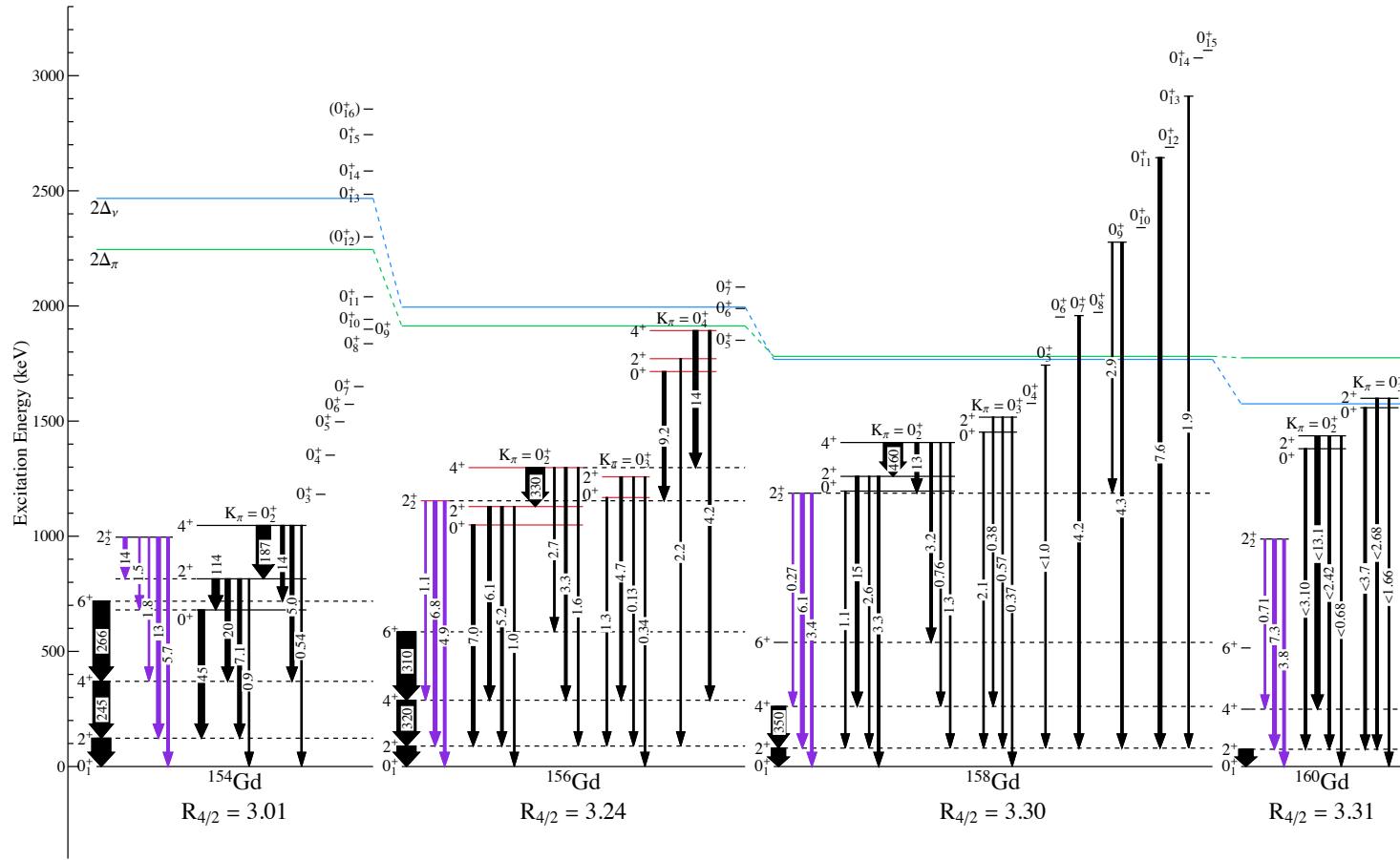


Figure 4.1. Systematics of the Gadolinium isotopes. ^{154}Gd has an anomalously large connection between the first excited 0^+ state and the ground state.

4.1 Ground State Band Confirmation

In the singles spectrum of, several prominent peaks stand out, seen in Figure 4.2. In the γ -spectrum, there are four prominent peaks from 100 to 500 keV. These peaks are the ground state band. The strong peak just beyond 500 keV is the 511 keV annihilation peak, and the peaks below 100 keV are x-rays. In the conversion electron spectrum of Figure 4.2, there are clear peaks until approximately 400 keV in, and then again in the range of 550 keV to 650 keV. The peaks up to 400 keV correspond to these same ground-state lines that are prominent in the γ -spectrum. The peaks in the range of 550 keV to 650 keV correspond to a strong series of $J^\pi \rightarrow J^\pi$ transitions from the first excited 0^+ band.

These ground state band transitions are all pure E2 multipoles, making them an excellent diagnostic to compare with theoretical conversion coefficients.

In the gated spectra, the internal conversion coefficients were in agreement with theory, meaning the conversion electron peaks did not have contributions from other transitions when gated, after the angular correlation correction was applied, discussed in section 3.3.3. In the singles, to remove the contaminant the gamma-ray had to be seen and identified with known multipolarity. The conversion coefficient was then assumed from theory, and the contribution to the area was calculated to be

$$A_e = \alpha \epsilon_e \frac{A_\gamma}{\epsilon_\gamma} C_\angle \quad (4.1)$$

where ϵ_i are the efficiencies of the electrons and gammas respectively. C_\angle is the angular correction for this particular transition, discussed in section 3.3.3. Here, it is multiplied, and not divided, so it can be subtracted from the uncorrected area. For the $6^+ \rightarrow 4^+$ and $8^+ \rightarrow 6^+$ transitions, electron contributions from smaller gamma lines of similar energy had to be subtracted. This was determined by looking at the conversion coefficients of these transitions in gated spectra, as

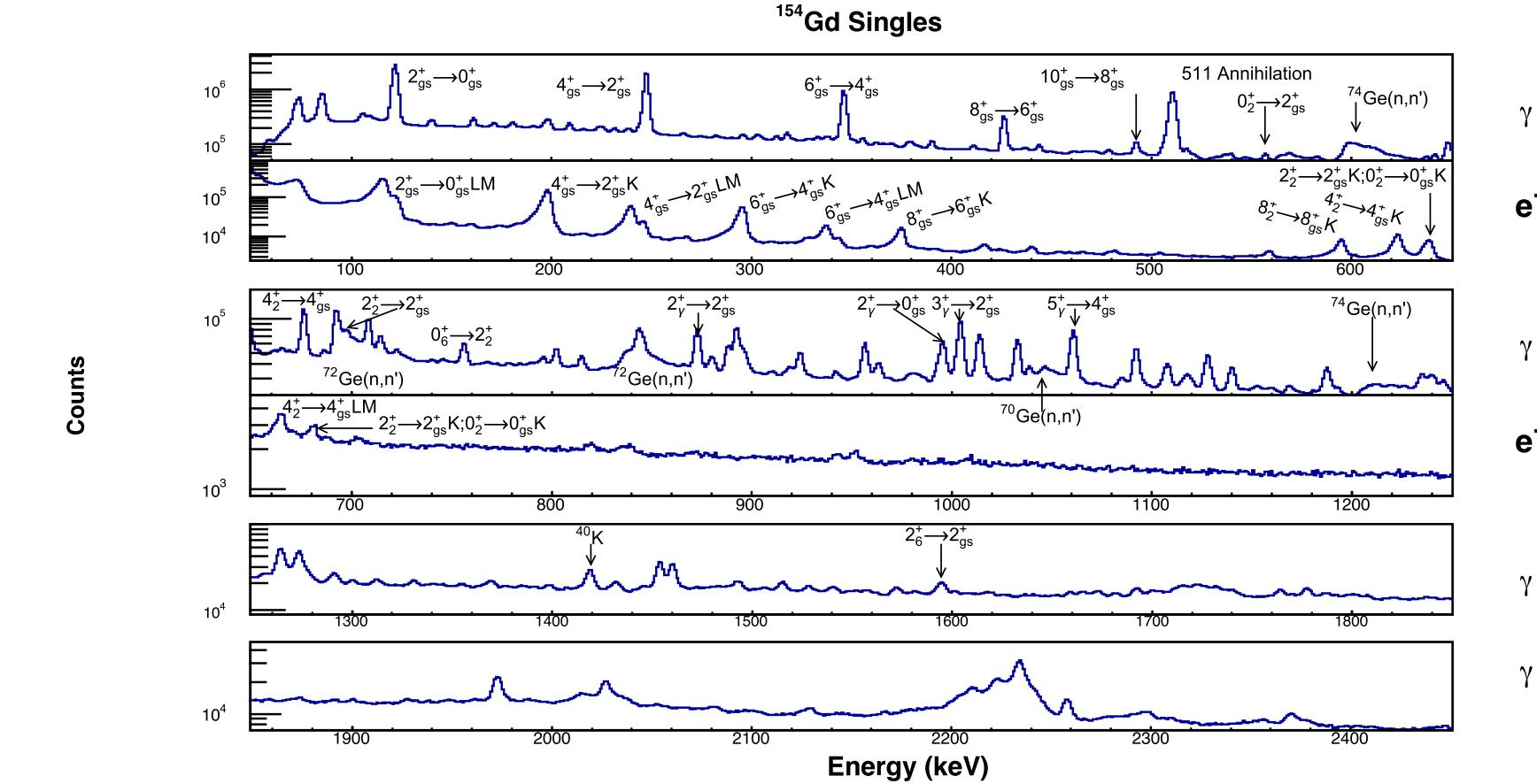


Figure 4.2. Singles spectra of ^{154}Gd . Spectra are labeled with the particles being detected, the energies of the γ and electron spectra aligned for identification. In the γ spectrum, several lines of note are labeled. The large peak at low energy in the electron spectrum is cut off due to the threshold. It is a combination of background and the 123K peak.

Transitions in the higher energy regime of the γ spectrum cannot be determined without gating, due to additional background from the experimental room. The electron detectors have a maximum energy of approximately 1250 keV, so there are no conversion electron spectra paired with the higher energy gamma spectra.

discussed in the next section. Looking at Figure 4.2, it is clear there are many transitions around the same energy, so the contributions may be many. However, only lines from reactions in the target material will contribute. As will be discussed in section 4.2, gating identifies these lines within the known level scheme.

Multipolarities could then be assigned based on the known data, and the contribution could be calculated based on the gamma area. The contributions were subtracted off the fitted area of the electron peak, before the conversion coefficient was calculated. This meant the error also include contributions from the errors of these gamma peaks and the errors on the theoretical α values.

There were two types of transitions that acted as contributed to the conversion electron peak areas: those of similar energy, and higher energies whose K-electron energies were similar to the L and M peaks of the lines of interest. The peaks are summarized in Table 4.1. The electron-shell of the transition is listed in each case, and total L and M contributions are used, without need to look at subshell contributions. In several transitions, the mixing ratio had to be assumed to get a conversion coefficient for the subtraction.

The $6^+ \rightarrow 4^+$ K-conversion coefficient went from $0.0546(1)(14)$ to $0.0335(1)_{-10}^{+11}$, in agreement with previous measurements and theory from BrICC[32]. Similarly, the $8^+ \rightarrow 6^+$ K-conversion coefficient went $0.0271(2)(7)$ to $0.0180(2)(5)$, also in agreement with theory calculations. The contributing transitions that were subtracted are summarized in Table 4.1. The conversion coefficient was then calculated, with an angular distribution correction based on the multipolarity of the transition. The transitions up to the 10^+ are summarized and compared with theory in Table 4.2.

TABLE 4.1

$$^{154}\text{GD GROUND STATE BAND } 6^+ \rightarrow 4^+ \text{ AND } 8^+ \rightarrow 6^+ \text{ EXTRA ELECTRON}$$

$$\text{PEAK CONTRIBUTIONS IN SINGLES}$$

E	E_i	E_f	$J_i \rightarrow J_f$	Multipolarity	δ	Shell	α
346.59 keV, $6^+ \rightarrow 4^+$							
338.28	1770	1432	$5_{4+}^+ \rightarrow 5_\gamma^+$	E0,E2,M1	-0.004	K	0.10 (1)
					$\frac{K(E0)}{K(E2)} \approx 1.0$	L	0.01210 (12)
						M	0.00180 (3)
342.18	2254	1911	$8_{4+}^+ \rightarrow 6_{4+}^+$	E2		K	0.0315 (5)
						L	0.0069 (1)
						M	0.001554 (22)
390.73	1756	1366	$8_2^+ \rightarrow 6_2^+$	E2		K	0.0218 (3)
426.84 keV, $8^+ \rightarrow 6^+$							
422.12	1418	996	$2_3^+ \rightarrow 2_\gamma^+$	E0,E2,M1	Assumed ≈ 1	K	0.114 (16)
						L	0.0161 (23)
						M	0.0049 (12)
466.99	1719	1251	$2_{2-}^- \rightarrow 3_{0-}^-$	M1,E2	Assumed ≈ 1	K	0.019 (6)
478.27	1719	1241	$2_{2-}^- \rightarrow 1_{0-}^-$	E2		K	0.01272 (18)

Table 4.1: A list of the transitions in the ^{154}Gd Singles data that were contributing to the electron peaks corresponding to the ground state band lines. Conversion coefficients were assumed using BrICC[32]. The bands for each level are listed as subscripts. Whole numbers indicate 0^+ bands.

TABLE 4.2

 ^{154}GD GROUND STATE BAND INTERNAL CONVERSION COEFFICIENTS FROM SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	Shell	α (This Work)	α (Th)[32]	α (Spits)[60]	α (Gono)[25]
122.23	$2^+ \rightarrow 0^+$	123.0709	0	1184000	E2	K	$0.7759 (34) {}^{+148}_{-146}$	0.656 (10)	0.61 (3)	
						L	$0.3788 (26) {}^{+81}_{-80}$	0.411 (6)		
						M	$0.1323 (3) (29)$	0.0963 (14)		
247.85	$4^+ \rightarrow 2^+$	370.9998	123.0709	45600	E2	K	$0.1044 (2) {}^{+31}_{-30}$	0.1000 (12)	0.080 (3)	0.0827 (119)
						L	$0.0272 (1) (8)$	0.0225 (4)		
						M	$0.0096 (1) (3)$	0.00513 (8)		
346.59 [†]	$6^+ \rightarrow 4^+$	717.662	370.9998	8260	E2	K	$0.0335 (1) {}^{+11}_{-10}$	0.0304 (5)	0.029 (1)	0.0306*
						L	$0.0087 (1) (3)$	0.00662 (10)		
						M	$0.0027 (1) (1)$	0.001491 (21)		
426.84 [†]	$8^+ \rightarrow 6^+$	1144.44	717.662	2570	[E2]	K	$0.0180 (2) (5)$	0.01716 (24)		0.0170 (22)
						L	$0.0030 (1) (1)$	0.00332 (5)		
						M	$0.0008 (1) (1)$	0.000741 (11)		
493.171	$10^+ \rightarrow 8^+$	1637.05	1144.44	1110	E2	K	$0.0124 (4) (4)$	0.01179 (17)		0.0124 (21)
						L	$0.0067 (4) (2)$	0.00213 (3)		

Table 4.2: A list of ground state band conversion coefficients from ^{154}Gd . Multipolarities and mixing ratios were taken from the data sheets[51]. Unless otherwise stated, the α values are α_K . An angular distribution correction has been applied based on multipolarities for pure transitions, and those with known mixing ratios. The first error is statistical, the second is systematic. Numbers are compared with Spits et al.[60] and Gono et al.[25]. The starred value was used as an absolute calibration of the conversion electron detector in the Gono work. The dagger lines had contaminant lines subtracted from the conversion electrons. See the text for more details.

4.2 Gates on the Ground State Band

To confirm the levels populated in the experiment, the γ components of the ground state transitions were used as energy gates. Figures 4.4, 4.5, 4.6, and 4.7 are the result of these gates. Band structure in these figures was taken from the data sheets[51]. No band assignments were made in this work. In the cases of cascades, secondary gates were done to confirm assignments. The spectra from these gates are available in Appendix A.

Determining which transitions went uniquely into a given ground state level was done by comparing the outgoing ground state transition for that level with the incoming transition i.e. for the 2^+ level, the $2^+ \rightarrow 0^+$ (123 keV) gate was compared directly with the $4^+ \rightarrow 2^+$ (247 keV) gate. The γ -spectra corresponding to two of these gates can be seen in Figure 4.3. Lines in the gamma spectrum that were present only in the outgoing spectrum, but not the incoming, were assumed to be populating that level. For example, in Figure 4.3, several areas have been highlighted. The line at 557 keV is present in the $2^+ \rightarrow 0^+$ gate, but not the $4^+ \rightarrow 2^+$ gate. This line corresponds to the $0_2^+ \rightarrow 2_{gs}^+$ transition, putting it in parallel with the $4^+ \rightarrow 2^+$ transition, explaining why it remains unseen. These identified transitions were then gated on to confirm the assignment, and to search for cascades from higher states.

There were nine bands seen in the gating, 4 of which are bands with excited 0^+ band heads. These are seen in the $2^+ \rightarrow 0^+$ gate (Figure 4.4). Three of these bands are seen in the $4^+ \rightarrow 2^+$ gate (Figure 4.5), and two are seen in the $6^+ \rightarrow 4^+$ gate (Figure 4.6). Only one is seen in the $8^+ \rightarrow 6^+$ gate (Figure 4.7). This drop off is unsurprising for multiple reasons. Studies have shown the $(\alpha, 2n)$ reaction to stop populating spin states with a significant cross section beyond $J^\pi = 12^+$ [65]. In the ground state band of ^{154}Gd , the 12^+ state sits at 2184.69 keV, giving an approximate cut off for populating higher energy states in the nucleus.

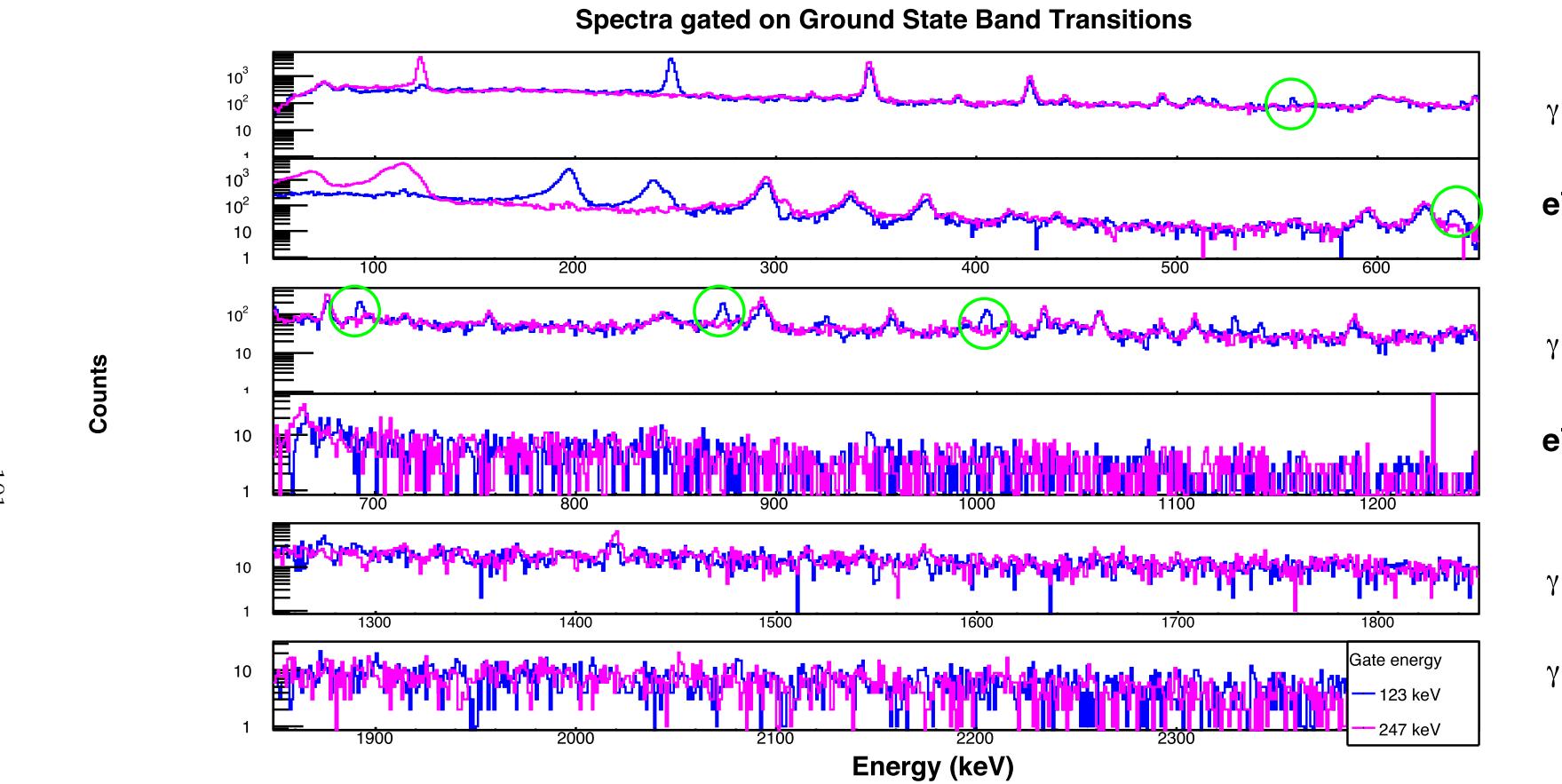


Figure 4.3. Spectra gated on 123 keV ($2^+ \rightarrow 0^+$) and 247 keV ($4^+ \rightarrow 2^+$), the first two ground state band lines of ^{154}Gd . As can be seen, some lines do not appear in different gates. Comparison of these gates, yields a list of transitions that directly populate the interim level (in this example, the 2^+ state). Several peaks are circled in green that appear in the $2^+ \rightarrow 0^+$ and not the $4^+ \rightarrow 2^+$ transition.

Additionally, due to the lack of known higher spin states in the higher energy 0^+ bands, identification of populating such bands in the higher gates becomes difficult.

4.3 Conversion Coefficients from Singles

Conversion coefficients were calculated from the singles spectra. Transitions that could be clearly identified and distinguished and identified or from near identical energy transitions are found in Table 4.3. Contamination concerns were taken into account by using the identified transitions to look for gamma-rays in close proximity to those in the table. Several transitions with E0 components can be found in Table 4.3. Most of the conversion electrons seen are the K -shell electron. These transitions were corrected for angular effects based on the multipolarity of the transition.

Table 4.4 holds conversion coefficients that have been left uncorrected in the singles for one of two reasons: either there were multiple known assignments to the gamma-ray energy, or the transition is considered a mixed transition, but no mixing ratio, δ has been previously measured. Table 4.5 contains the conversion coefficients that could not be corrected because the transition has no multipole assignment. It contains allowable and reasonable theoretical conversion coefficients to compare the values against, calculated using BrIcc [32].

In Table 4.3, there are several conversion coefficients that are high, not only compared to theory, but compared to previous measurements (232, 329, 444, and 557 keV). For the 232 keV transition, the large error hints at a lack of statistics. For the other three transitions, the transition indicates a change from smaller to larger J^π . The angular distribution correction for such transitions is less than 1, inflating the value. This may be an indication the angular distribution correction is not wholly accurate. Further, due to these values being from singles data, unknown contaminants to the electron spectrum cannot be ruled out. While these transitions

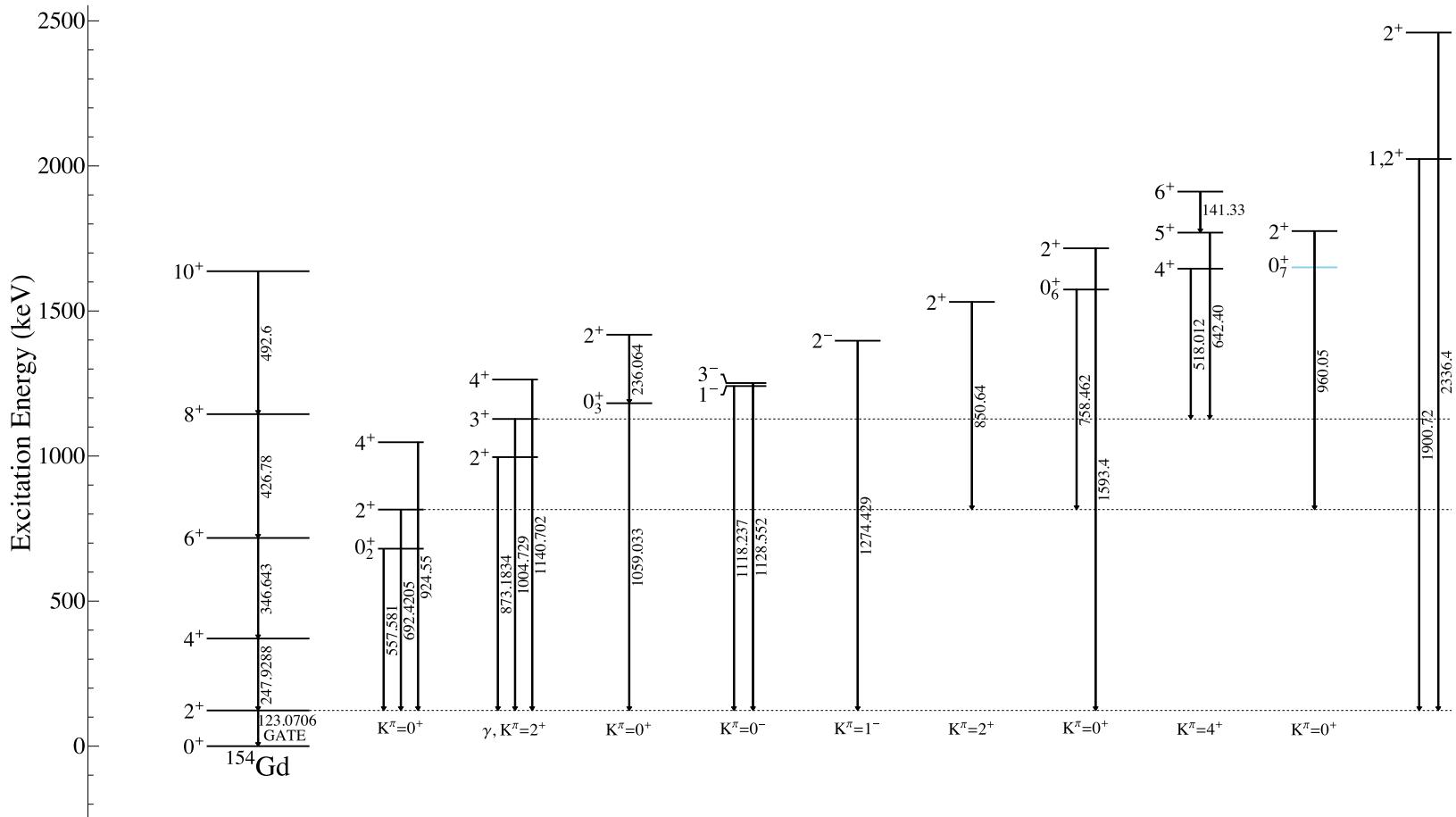


Figure 4.4(a) Level Scheme of ^{154}Gd . The gate on the $2^+ \rightarrow 0^+$ transition (123 keV) γ -component in the ground state. The lines shown are in coincidence. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

123 keV gated spectra

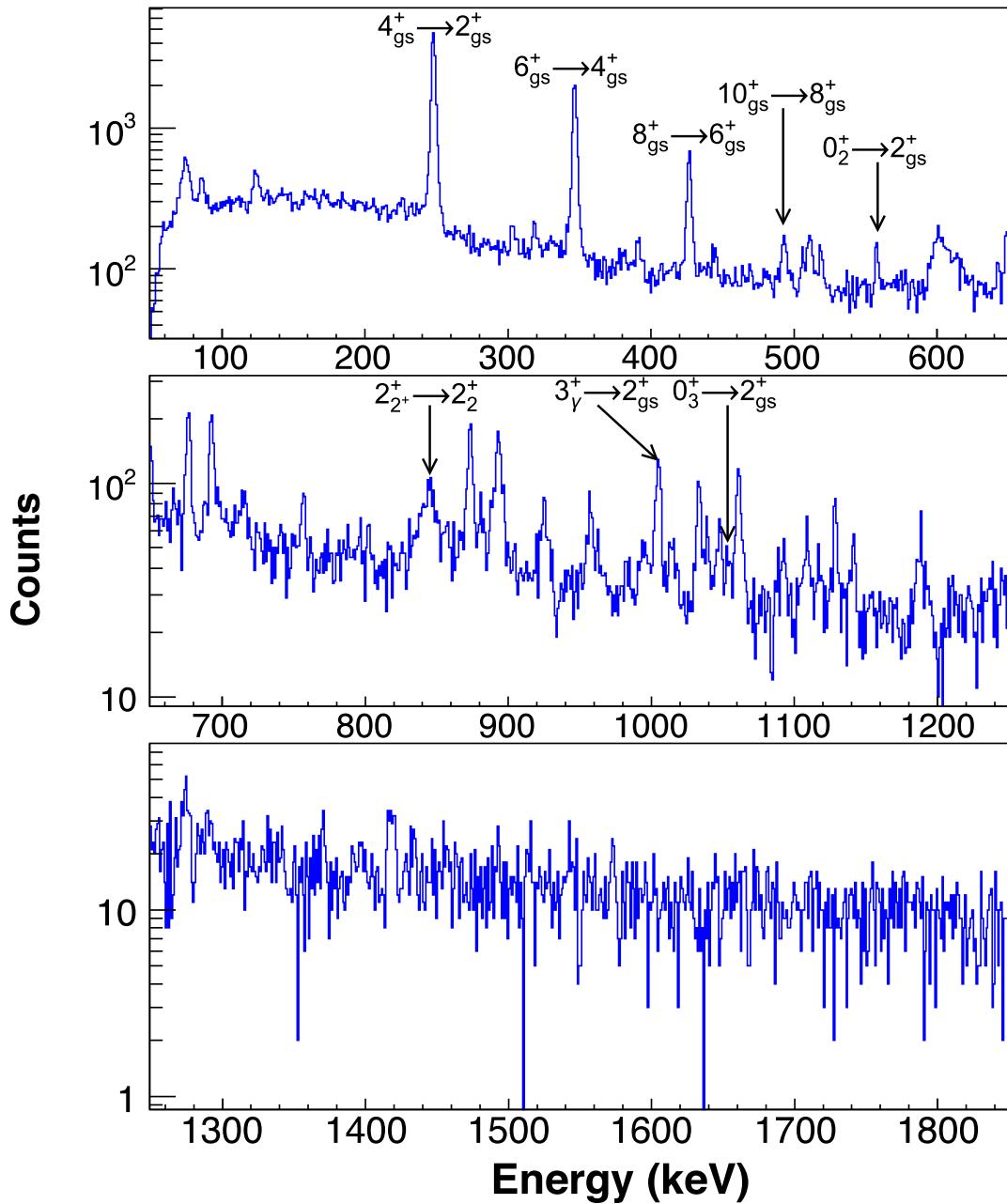


Figure 4.4(b) Gamma spectrum gated on 123 keV, corresponding to the $2^+ \rightarrow 0^+$ transition. Several transitions have been labeled, corresponding to the level scheme.

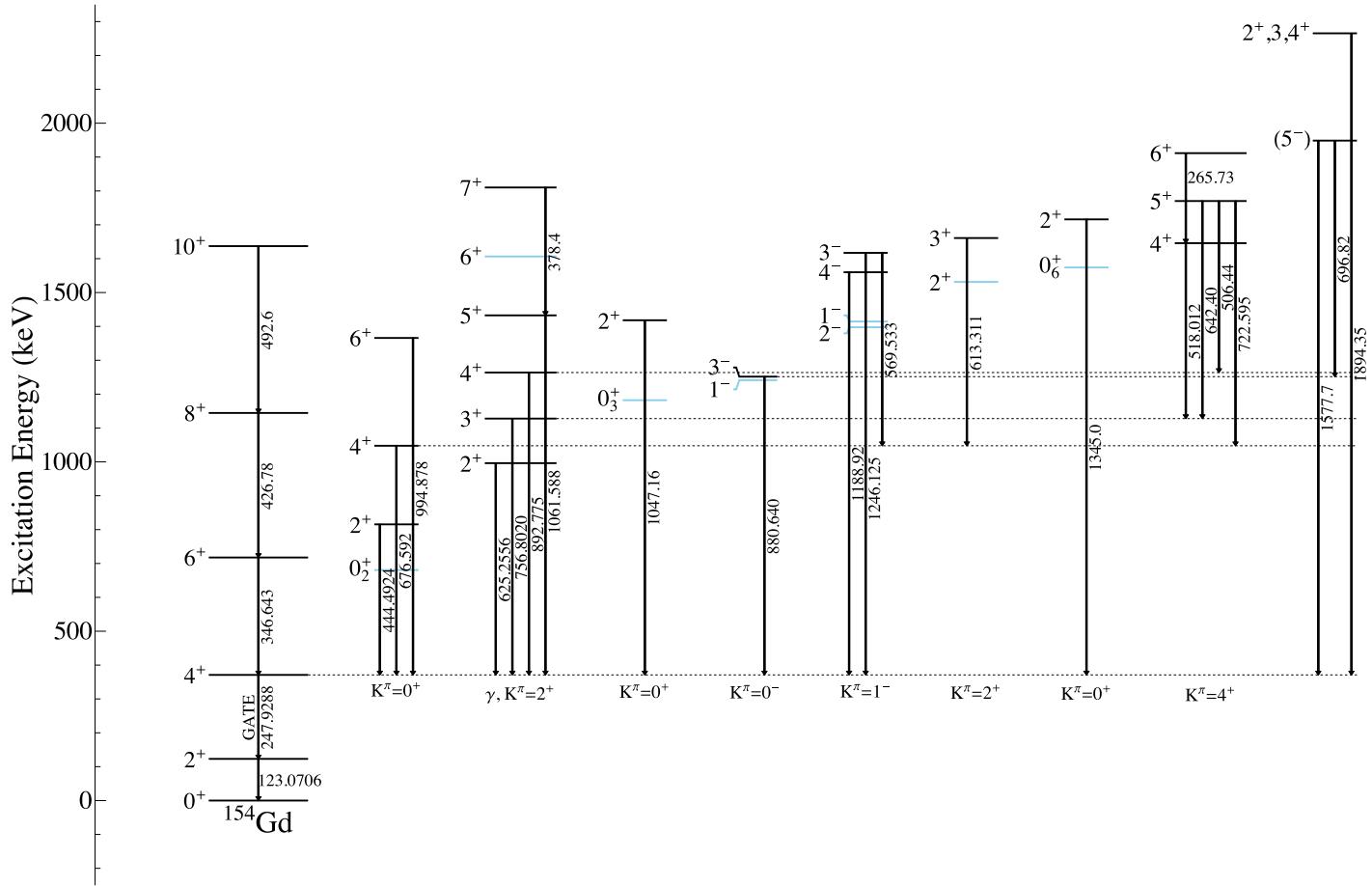


Figure 4.5(a) Level Scheme of ^{154}Gd . The gate on the $4^+ \rightarrow 2^+$ transition (247 keV) γ -component in the ground state. The lines shown are in coincidence. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

247 keV gated spectra

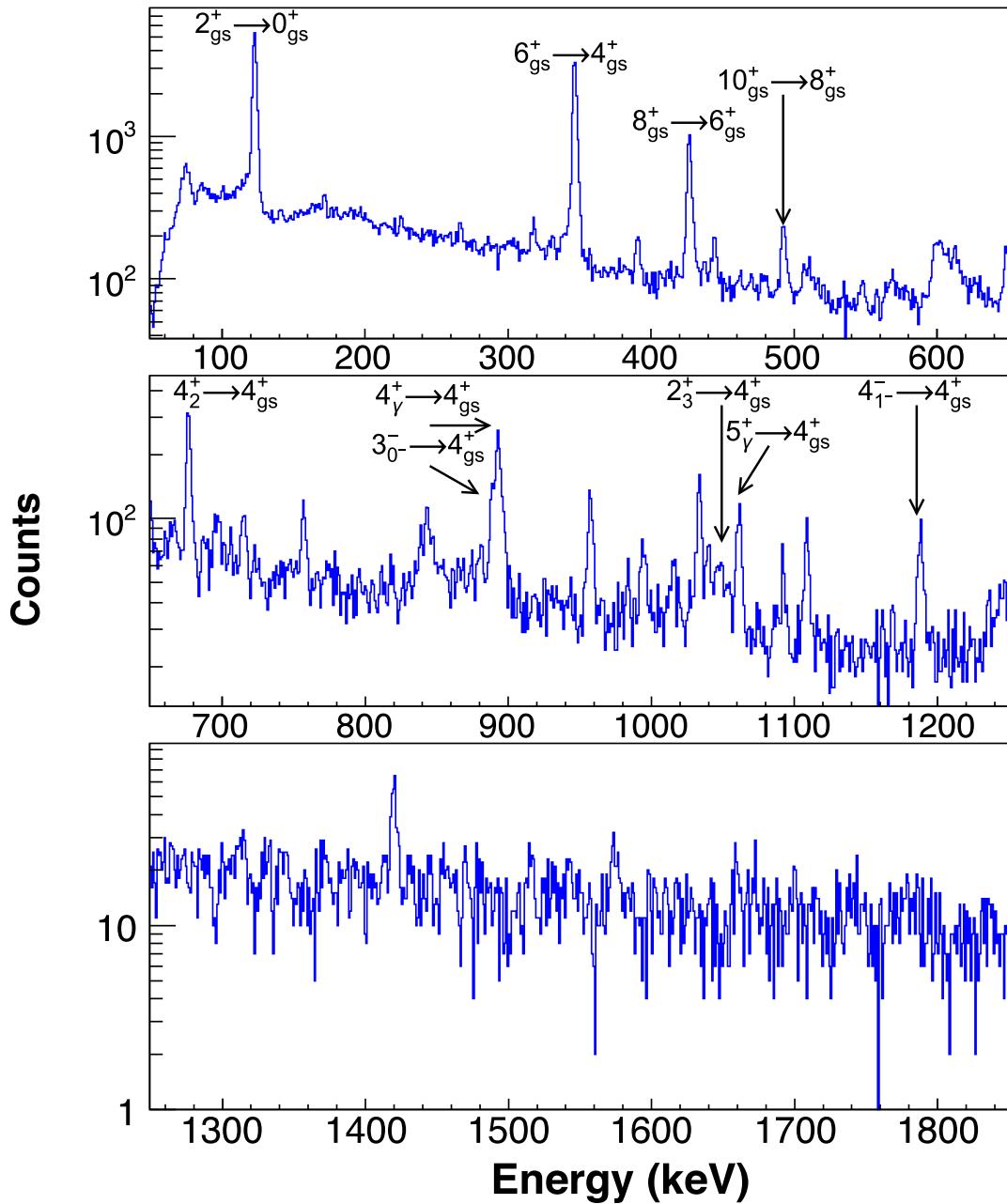


Figure 4.5(b) Gamma spectrum gated on 247 keV, corresponding to the $4^+ \rightarrow 2^+$ transition. Several transitions have been labeled, corresponding to the level scheme.

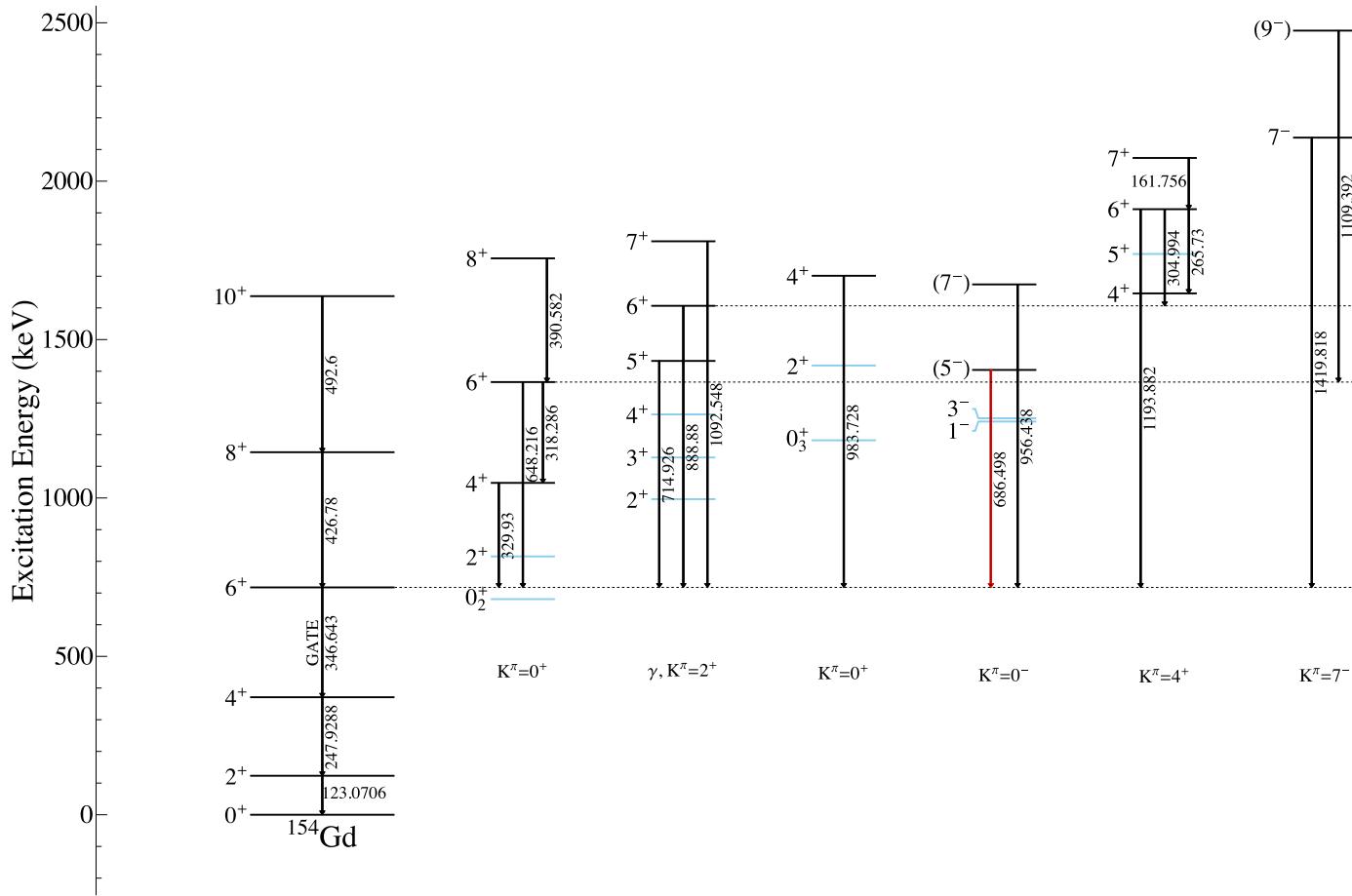


Figure 4.6(a) Level Scheme of ^{154}Gd . The gate on the $6^+ \rightarrow 4^+$ transition (346 keV) γ -component in the ground state. The lines shown are in coincidence. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue. The red line in the $K^\pi = 0^-$ band has not been previously seen, but coincidence gates place it there.

346 keV gated spectra

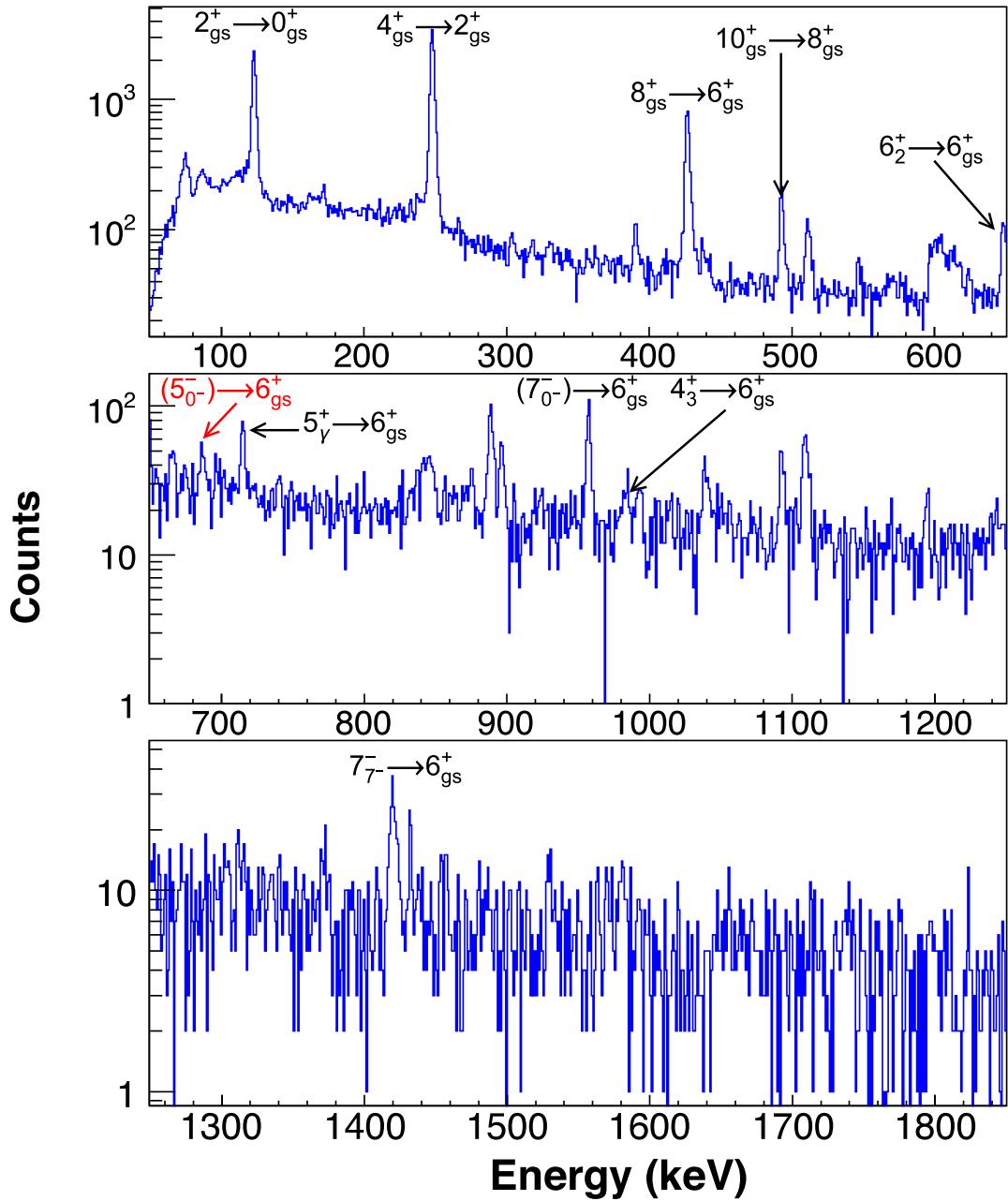


Figure 4.6(b) Gamma spectrum gated on 346 keV, corresponding to the $6^+ \rightarrow 4^+$ transition. Several transitions have been labeled, corresponding to the level scheme. The red label corresponds to the red line in the level scheme.

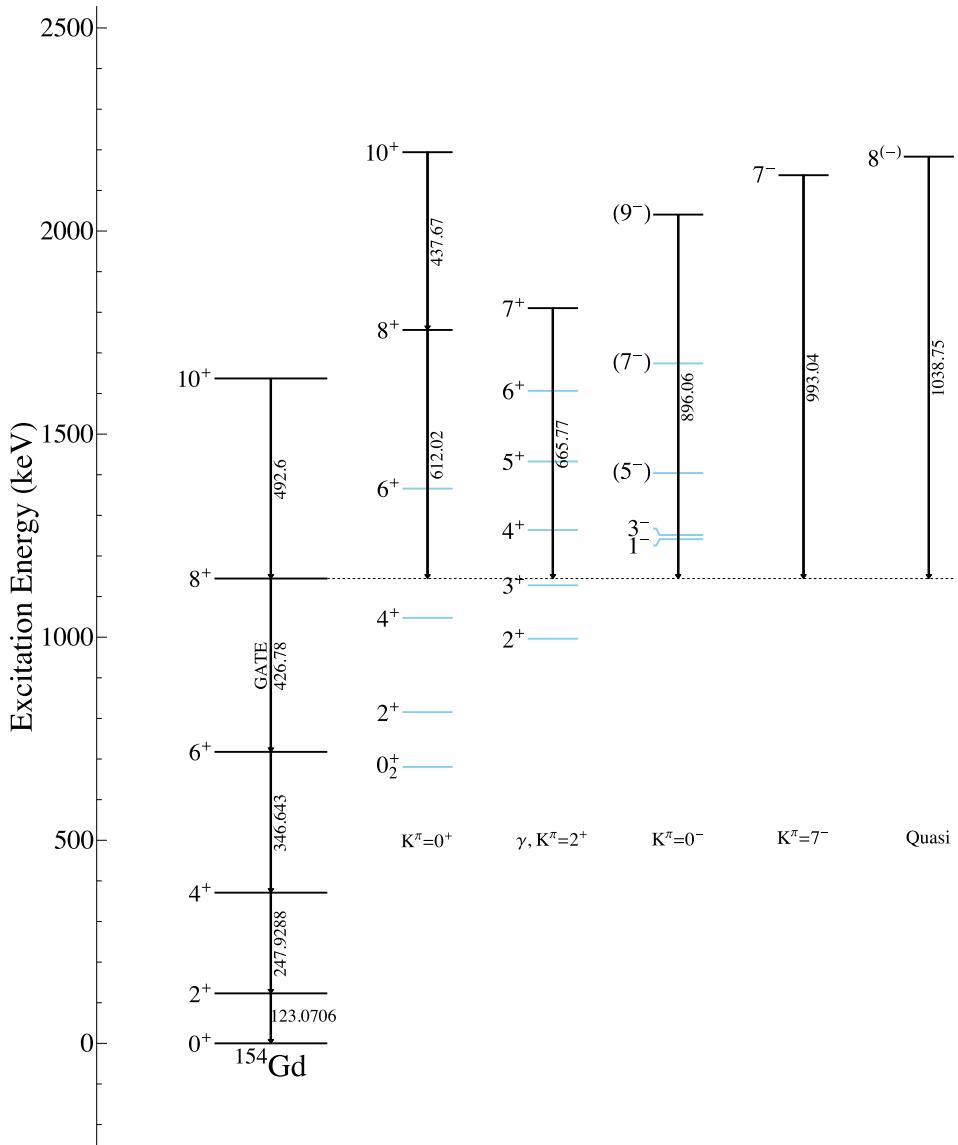


Figure 4.7(a) Level Scheme of ^{154}Gd . The gate on the $8^+ \rightarrow 6^+$ transition (426 keV) γ -component in the ground state. The lines shown are in coincidence. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

426 keV gated spectra

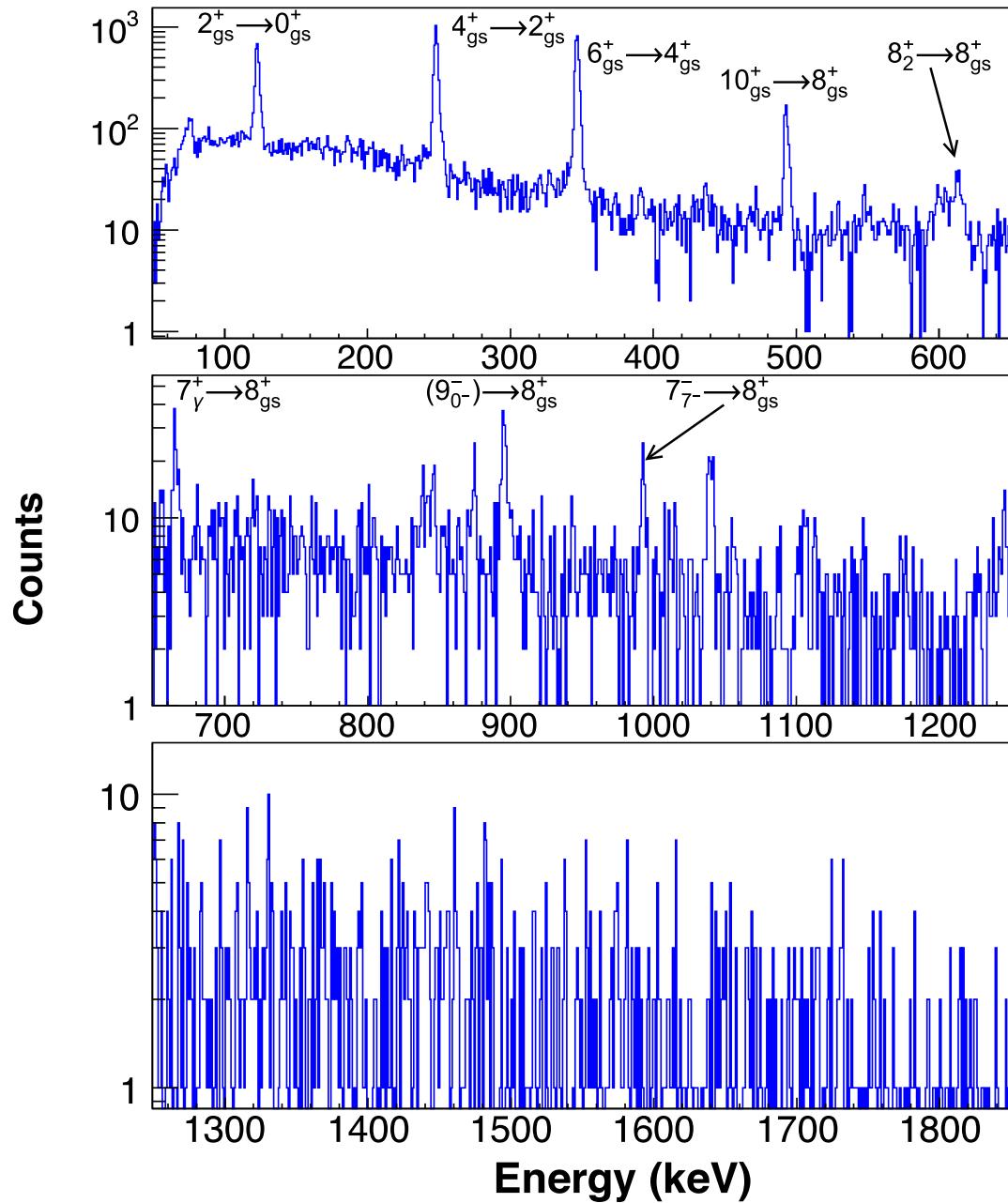


Figure 4.7(b) Gamma spectrum gated on 426 keV, corresponding to the $8^+ \rightarrow 6^+$ transition. Several transitions have been labeled, corresponding to the level scheme.

TABLE 4.3

 ^{154}GD INTERNAL CONVERSION COEFFICIENTS FROM SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	δ	Shell	α (This Work)	α (Th)[32]	α (Spits)[60]	α (Gono)[25]
232.44	$4_2^+ \rightarrow 2_2^+$	1047.592	815.4917	7600	E2		K	0.287 (103) $^{+83}_{-82}$	0.0982 (14)	0.100 (8)	
							LM	0.0450 (46) (13)	0.0288 (4)		
329.49	$4_2^+ \rightarrow 6_{gs}^+$	1047.592	717.662	7600	E2		K	0.1665 (94) (18)	0.0352 (5)	0.034 (3)	
349.89	$2_{2+}^+ \rightarrow 0_{gs}^+$	1531.305	1182.091		[E2]		K	0.0298 (8) $^{+8}_{-7}$	0.0296 (5)	< 0.097	
416.79	$4_6^+ \rightarrow 3_{2+}^+$	2080.23	1660.903		(M1)		K	0.0331 (47) (6)	0.03442 (5)		
444.19	$2_2^+ \rightarrow 4_\gamma^+$	815.4917	370.9998	6400	E2		K	0.0556 (34) (6)	0.01543 (22)	0.014 (1)	
506.41	$5_4^+ \rightarrow 4_\gamma^+$	1770.187	1263.778		E2		K	0.008 (4) (1)	0.01098 (16)	0.0100 (11)	
515.92	$4_{4+}^+ \rightarrow 3_{\gamma}^+$	1645.814	1127.802		E2+M1	-7 (3)	K	0.0069 (5) (1)	0.0107 (4)	0.0113 (9)	
557.58	$0_2^+ \rightarrow 2_{gs}^+$	680.6673	123.0709	4560	E2		K	0.0452 (39) (6)	0.00864 (12)	0.009 (1)	0.0091 (16)
610.71	$8_2^+ \rightarrow 8_{gs}^+$	1756.49	1144.44		E0+M1+E2	-0.69 (14)	K	0.0236 (9) (6)	0.0110 (6)		0.053 (7)
							L	0.0153 (8) (4)	0.00158 (7)		
676.70	$4_2^+ \rightarrow 4_{gs}^+$	1047.592	370.9998	7600	E0+M1+E2	+2.9 (4)	K	0.0281 (4) (10)	0.00593 (17)	0.0460 (46)	0.040 (7)
693.47	$2_2^+ \rightarrow 2_{gs}^+$	815.4917	123.0709	6400	E2+M1+E0	7.5 (4)	K	0.0017 (2) (1)	0.00522 (8)	0.0421 (4)	
873.54	$2_\gamma^+ \rightarrow 2_{gs}^+$	996.2568	123.0709	950	E2+M1+E0	-9.4 (4)	K	0.0021 (3) (1)	0.00311 (5)	0.0035 (1)	
889.61	$6_\gamma^+ \rightarrow 6_{gs}^+$	1606.55	717.662		E2+M1	> 1.8	K	0.0044 (6) (2)	0.00349 (5)	0.0033 (2)	
894.40	$4_\gamma^+ \rightarrow 4_{gs}^+$	1263.778	370.9998		E0+M1+E2	-3.8 (3)	K	0.0019 (2) (1)	0.00307 (5)	0.0039 (3)	
924.85	$4_2^+ \rightarrow 2_{gs}^+$	1047.592	123.0709	7600	E2		K	0.0033 (9) (1)	0.00273 (4)	0.0031 (1)	
996.33	$2_\gamma^+ \rightarrow 0_{gs}^+$	996.2568	0	950	E2		K	0.0021 (4) (1)	0.00234 (4)	0.0025 (1)	
1005.12	$3_\gamma^+ \rightarrow 2_{gs}^+$	1127.8018	123.0709		E2+M1	-7.4 (4)	K	0.0019 (1) (1)	0.00233 (4)	0.0024 (1)	

Table 4.3: A list of conversion coefficients from ^{154}Gd . Multipolarities and mixing ratios were taken from the nuclear data sheets[51]. Unless otherwise stated, the α values are α_K . An angular distribution correction has been applied based on multipolarities for pure transitions, and those with known mixing ratios. The first error is statistical, the second is systematic. Numbers are compared with Spits et al.[60] and Gono et al.[25] The starred value was used as an absolute calibration of the conversion electron detector in the Gono work. The bands for each level are listed as subscripts. Whole numbers indicate 0^+ bands.

TABLE 4.4

 UNCORRECTED ^{154}GD INTERNAL CONVERSION COEFFICIENTS FROM
 SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	Multipolarity	δ	Shell	α (This Work)	α (Th)[32]	α (Spits)[60]	α (Gono)[25]
198.311	$3_{1-}^- \rightarrow 2_3^+$ $(9)_{4+}^+ \rightarrow (8)_{4+}^+$	1617.123	1418.16	[E1]		K	0.0842 (26) (19)	0.0393 (6)		
313.21	$3_\gamma^+ \rightarrow 2_2^+$	1127.802	815.492	[M1,E2]		K	0.0805 (47) (20)			
641.92	$5_{4+}^+ \rightarrow 3_\gamma^+$	1770.187	1127.8018	M1,E2		K	0.0303 (7) (8)		0.0086 (8)	
648.951	$6_2^+ \rightarrow 6_{gs}^+$	1365.87	717.662	E0+M1+E2	+1.30 (20)		0.0079 (5)		0.039 (7)	
715.198	$2_{2+}^+ \rightarrow 2_2^+$	1531.305	815.492	E0+M1+E2		K	0.0199 (5) (5)		0.0070 (5)	
						L	0.0041 (4) (1)			
1061.59	$0_3^+ \rightarrow 2_{gs}^+$ $5_\gamma^+ \rightarrow 4_{gs}^+$	1182.091	123.0709	E2		K	0.0022 (3) (1)	0.0021 (1)		
		1432.588	370.9998	E2+M1	-4.3_{-26}^{+12}		0.0021 (1)	0.0019 (4)		

Table 4.4: A list of conversion coefficients from ^{154}Gd . Multipolarities and mixing ratios were taken from the nuclear data sheets[51]. Unless otherwise stated, the α values are α_K . No angular distribution correction has been applied, either due to unknown mixing ratios, or multiple assignments of the gamma-ray. None of the above transitions have known half-lives. The first error is statistical, the second is systematic. Numbers are compared with Spits et al.[60] and Gono et al.[25]. The bands for each level are listed as subscripts. Whole numbers indicate 0^+ bands.

TABLE 4.5

 ^{154}GD INTERNAL CONVERSION ELECTRONS WITHOUT KNOWN
MULTIPOLEARITIES

E (keV)	$J_i \rightarrow J_f$	E_i (keV)	E_f (keV)	Shell	α (This Work)	Theory[32]			
						$\alpha(\text{M1})$	$\alpha(\text{E2})$	$\alpha(\text{E1})$	α (Spits)[60]
266.37	$6_{4+}^+ \rightarrow 4_{4+}^+$	1911.544	1645.814	K	0.2074 (74) (50)		0.0654 (10)		
303.89	$(7^+)_{4+} \rightarrow 5_{4+}^+$	2073.30	1770.187	K	0.1183 (40) $^{+30}_{-29}$		0.0444 (7)		
318.382	$6_2^+ \rightarrow 4_2^+$	1365.87	1047.592	K	0.0736 (18) (18)		0.0388 (6)		
				L	0.0371 (12) (9)		0.00892 (13)		
379.55	$7_\gamma^+ \rightarrow 5_\gamma^+$	1810.21	1432.588	K	0.1120 (60) (31)		0.0236 (4)		
433.12	$4_6^+ \rightarrow 4_{4+}^+$	2080.23	1645.814	K	0.0571 (42) (15)	0.0310 (5)	0.01650 (24)		0.0220 (45)
687.05	$(5^-)_{0-} \rightarrow 6_{gs}^+$	1404.16	717.662	K	0.3538 (116) (90)			0.00203 (3)	
722.64	$5_{4+}^+ \rightarrow 4_2^+$	1770.187	1047.592	K	0.0166 (12) (42)	0.00856 (12)	0.00468 (7)		
1033.91	$3^- \rightarrow 4_2^+$	2080.791	1047.592	K	0.0015 (4) (1)			0.000916 (13)	

Table 4.5: A list of conversion coefficients from ^{154}Gd without known multipolarities. As a result, an angular distribution correction term has not been applied. None of the above transitions have known half-lives. The first error is statistical, the second is systematic. Numbers are compared with theoretical coefficients for allowed and reasonable polarities, as well as results from Spits et al. [60] The bands for each level are listed as subscripts. Whole numbers indicate 0^+ bands. The 3^- for $E = 1033.91$ keV has no band placement.

were identified clearly and separably, low-intensity gamma-rays may have been missed in the gates. There are also three transitions that are lower than the theoretical values (515, 873 and 894 keV). In all of these cases, δ is negative. Finally, there are several $J^\pi \rightarrow J^\pi$ transitions seen in the singles. For the $4_2^+ \rightarrow 4_{gs}^+$ between the first excited 0^+ band and the ground state band at 676 keV, there is clearly an E0 component, although not as big at that seen by Spits or Gono [25, 60]. It is a similar situation for the $8^+ \rightarrow 8^+$ at 610 keV compared to Gono. Additionally, there is a $6^+ \rightarrow 6^+$ transition at 889 keV that is higher than number from Spits and the theoretical conversion coefficient. The $4^+ \rightarrow 4^+$ was also found in gating, as will be discussed in the next section.

In Table 4.4, the $2^+ \rightarrow 2^+$ transition at 715 keV appears to have a large E0 component. This transition is examined in the next section. For the two entangled transitions around 645 keV, the $6^+ \rightarrow 6^+$ transition is likely the dominant contribution, as the measured value in the singles is in good agreement with the measured value in the gated data, see Table 4.10. Two separate α values are listed in this part of Table 4.4, as the gamma-rays were separable, but the conversion electron peaks were not.

In Table 4.5, the $4^+ \rightarrow 4^+$ transition at 433 keV either indicates a mixed transition with an E0 component. The angular correction for a pure E2 transition would make the conversion coefficient approximately double the theoretical conversion coefficient from BrICC [32]. This angular correction is the largest correction that can be made. Mixed component angular corrections result in values larger than the theoretical M1 transition. Together, this points toward an E0 component to the transition. Unfortunately, this transition could not be found in gates, as the levels are too high in energy to be populated significantly enough for gates.

4.4 $J^\pi \rightarrow J^\pi$ Transitions

4.4.1 $0^+ \rightarrow 0^+$ Identification and Population

With the large number of 0^+ states in ^{154}Gd , it was necessary to identify which were populated. Table 4.6 contains a list of 0^+ states from Meyer et al, with indications as to which 0^+ have been observed outside of the discovery paper, and which have been previously observed in the reaction used in this work [45].

Many of the states were left unseen in this work as is to be expected. As previously mentioned, the energy of the 12^+ in the ground state band, 2184.69 keV, is the rough cut off for populating states. The levels seen in this work exceed previous $(\alpha, 2n)$ experiments and agree with the levels seen in Spits et al, a thermal neutron capture experiment[60]. Until Spits, there was no known spectroscopy for the 1182, 1573 and 1650 keV states, having only been seen previously in (p, t) and (t, p) experiments. No $(\alpha, 2n)$ experiments with $\gamma - \gamma$ coincidence, looking for such levels have been run since Spits et al was published. The energies of the $0^+ \rightarrow 0^+$ transitions from these states are marked in the electron singles in Figure 4.8. To obtain information in these areas, and for possible $J^\pi \rightarrow J^\pi$ transitions, energy gates were used. A full list of gates and spectra can be found in Appendix A.

4.4.2 $J^\pi \rightarrow J^\pi$ Transitions

After identification of the bands and transitions, gates were put on the transitions both entering and leaving states of interest, namely even- J^+ states. Gates entering these states did not have enough statistics to glean information on transitions of interest from the resulting spectra. However, gates leaving these states allowed enough statistics to look for transitions from states of the same J^π . In many cases, two separate gates could be used to look at the same state. To look for these transitions, other transitions, namely those of the ground state band, were

TABLE 4.6

 0^+ STATES IN ^{154}GD

Energy (keV)	Seen Previously [45]	Seen in $(\alpha, 2n)$	Seen in This Work
0	×	×	×
680.4(3)	×	×	×
1181.9(3)	×		×
1352.9(3)			×
1497.7(3)			
1573.7(3)	×		×
1650.6(4)	×		×
1836.7(4)			
1899.3(4)			
1942.9(4)			
2039.8(4)			
2299.9(5)	×		
2485.1(5)			
2585.3(5)			
2744.5(5)	×		
2855.0(5)	×		

Table 4.6: A list of all 0^+ states in ^{154}Gd . All the states listed were seen in Meyer et al, a (p, t) experiment[45]. 0^+ states above approximately 2184 keV would not be seen in this experiment, the energy of the 12^+ ground state, as that is the energy cut off in the nucleus for state population, discussed in the text. The levels seen in this exceed previous $(\alpha, 2n)$ experiments and agree with the levels seen in Spits [60]. Until Spits, there was no known spectroscopy for the 1182, 1573 and 1650 keV states, having only been seen previously in (p, t) and (t, p) experiments. No $(\alpha, 2n)$ experiments with $\gamma - \gamma$ coincidence, looking for such levels have been run since Spits et al was published.

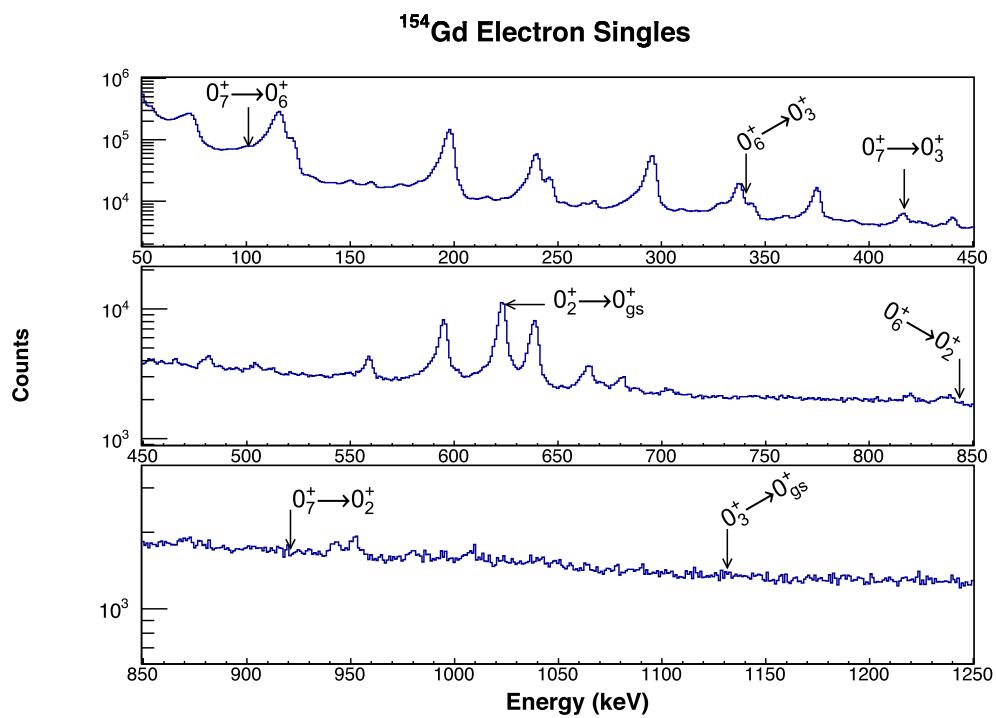


Figure 4.8. Electron singles for ^{154}Gd . The energies of transitions between 0^+ states are marked in the plot.

subtracted from the regions of interest as described in section 3.4. This subtraction technique causes the results to be treated as limits. Tables 4.7 - 4.10 are the tabulated results. Where previous measurements have been taken, they have been listed. Additionally, the theoretical conversion coefficients have been listed for $M1$ and $E2$ transitions, as taken from BrIcc [32].

As is expected, Table 4.7 is only lower limits. The pure $E0$ transition should not have a gamma component, leading to a conversion coefficient of infinity. A lower limit in this table is an indication of the noise in the gamma spectrum that acts as the limit for the ability to see a line in the spectrum. Although $M1$ and $E2$ transitions are not allowed, they have been included in the table for the sake of comparison. The lower limits, expectedly, rule out the $M1$ and $E2$ transitions. The measured limits agree with the ones seen by Spits [60].

In Tables 4.8, 4.9 and 4.10 many of the transitions could be seen in two different gates. In many cases, only lower limits could be seen due to the noise limit in the gamma spectrum. In other cases, only upper limits could be seen due to the same effect in the electron spectrum.

The majority of these states do not have lifetimes, so $B(E0)$ values cannot be calculated. However, the relative intensities of these values can be compared, assuming they are coming from the same state, as the lifetime would divide out (see equation 1.41 and 1.42). The contributions from the individual components of the transition must be separated out.

To compare the $E0$ components, the other two major contributing components ($M1$ and $E2$) had to be subtracted out. This was done by calculating ϵ^2 via equation 1.43. Transitions with known δ mixing ratios are in Table 4.13. Transitions without known δ mixing ratios, had δ assumed to be 1. In some cases, this left a negative value, which has been excluded from the table of results. These values are listed in Table 4.11. The $0^+ \rightarrow 0^+$ transitions are in Table 4.12. This subtraction was not

TABLE 4.7

 $0^+ \rightarrow 0^+$ TRANSITIONS IN ^{154}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		
							α (M1)	α (E2)	α (Spits)[60]
1182.091	3	680.6673	2	501.427	557.581	> 0.0283	0.0213 (3)	0.01126 (16)	> 0.2
1573.9	6	680.6673	2	893.9	557.581	> 0.0183	0.00510 (8)	0.00294 (5)	
1573.9	6	1182.091	3	391.9	1059.033	> 0.0529	0.0402 (6)	0.0216 (3)	> 0.1
1650.3	7	680.6673	2	970.3	557.581	> 0.0209	0.00419 (6)	0.00247 (4)	> 0.027
1650.3	7	1182.091	3	468.3	1059.033	> 0.0922	0.0254 (4)	0.01343 (19)	

Table 4.7: A list of conversion coefficients from ^{154}Gd for $0^+ \rightarrow 0^+$ transitions seen in the gated data. All are lower limits. Numbers are compared with Spits et al.[60] and theoretical coefficients for M1 and E2 transitions. All coefficients are K-electrons.

TABLE 4.8

 $2^+ \rightarrow 2^+$ TRANSITIONS IN ^{154}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		
							α (M1)	α (E2)	α (Spits)[60]
815.4917	2	123.0709	GS	692.4205	123.0706	0.0430 (3) (9)	0.00952 (14)	0.00516 (8)	0.0421 (4)
996.264	γ	815.4917	2	180.72	692.4205 444.4924	> 1.0570 > 0.9718	0.320 (5)	0.210 (3)	
1418.16	3	815.4917	2	602.688	692.4205 444.4924	> 0.0125 > 0.0093	0.01343 (19)	0.00715 (10)	0.025 (3)
1418.16	3	996.2568	γ	421.893	873.1834 625.2556	> 0.0367 > 0.0463	0.0332 (5)	0.01170 (25)	0.114 (16)
1531.305	2^+	815.4917	2	715.819	692.4205 444.4924	0.0146 (40) $^{+43}_{-33}$ 0.0234(80) $^{+68}_{-52}$	0.00877 (13)	0.00478 (7)	0.0070 (4)
1531.305	2^+	996.2568	γ	535.050	873.1834 625.2556	0.0204 (70) $^{+54}_{-41}$ > 0.0183	0.0181 (3)	0.00956 (14)	0.093 (11)
1716.050	6	815.4917	2	900.5583	692.4205 444.4924	< 0.0105 < 0.0531	0.00501 (7)	0.00289 (4)	

TABLE 4.8 (CONTINUED)

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		
							α (M1)	α (E2)	α (Spits)[60]
1716.050	6	996.2568	γ	719.80	873.1834	0.0113 (46) $^{+33}_{-25}$	0.00865 (13)	0.00472 (7)	
					625.2556	0.0501 (260) $^{+147}_{-113}$			
1775.429	7	815.4917	2	960.05	692.4205	> 0.0221	0.00430 (6)	0.00253 (4)	
					444.4924	> 0.0231			
1775.429	7	996.2568	γ	779.165	873.1834	0.0206 (112) $^{+60}_{-46}$	0.00712 (10)	0.00396 (6)	
					625.2556	0.0745 (521) $^{+217}_{-165}$			

Table 4.8: A list of conversion coefficients from ^{154}Gd for $2^+ \rightarrow 2^+$ transitions seen in the gated data. The first error is statistical, the second is systematic. Numbers are compared with theoretical K-shell conversion coefficients for M1 and E2 transitions, as well as results from Spits et al.[60] All coefficients are K-electrons. Whole numbers for the bands are the 0^+ bands.

TABLE 4.9

 $4^+ \rightarrow 4^+$ TRANSITIONS IN ^{154}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		α (Spits)[60]	α (Gono)[25]
							α (M1)	α (E2)		
1047.592	2	370.9998	GS	676.593	247.9288	0.0550 (2) $^{+12}_{-11}$ 0.0131 (1) (3)	0.01007 (15) 0.001384 (20)	0.00544 (8) 0.000870 (13)	0.0460 (46)	0.040 (7)
1263.778	γ	1047.592	2	216.186	676.593 924.55	< 0.1250 < 0.1033	0.196 (3)	0.1222 (18)		
1645.814	4^+	1047.592	2	598.22 924.55	676.593	< 0.0092 < 0.0142	0.01368 (20)	0.00728 (11)	< 0.067	
1645.814	4^+	1263.778	γ	382.025 1140.702	892.775 1140.702	< 0.0360 < 0.0494	0.0429 (6)	0.0232 (4)	0.033 (5)	
1701.39	3	1047.592	2	653.7 924.55	676.593	< 0.0093 < 0.0301	0.01097 (16)	0.00590 (9)	0.0220 (62)	
1701.39	3	1263.778	γ	437.612 1140.702	892.775 1140.702	< 0.0585 < 0.0511	0.0302 (5)	0.01605 (23)		
1789.17	2^+	1047.592	2	740.91 924.55	676.593	< 0.0124 < 0.0447	0.00806 (12)	0.00443 (7)		
1789.17	2^+	1263.778	γ	525.392	892.775	< 0.0168	0.0190 (3)	0.01001 (14)		

TABLE 4.9 (CONTINUED)

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		α (Spits)[60]	α (Gono)[25]
							α (M1)	α (E2)		
				1140.702		< 0.0161				

Table 4.9: A list of conversion coefficients from ^{154}Gd for $4^+ \rightarrow 4^+$ transitions seen in the gated data. The first error is statistical, the second is systematic. Numbers are compared with theoretical K-shell conversion coefficients for M1 and E2 transitions, as well as results from Spits et al.[60] and Gono et al.[25] All coefficients are K-electrons, except for the transition from 1047 keV. The second value is the LM peak. Whole numbers for the bands are the 0^+ bands.

TABLE 4.10

 $6^+ \rightarrow 6^+$ TRANSITIONS IN ^{154}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory[32]		α (Gono)[25]
							α (M1)	α (E2)	
1365.878	2	717.662	GS	648.3	346.643	0.0778 (4) (16)	0.01120 (16)	0.00601 (9)	0.039 (7)
1606.55	γ	1365.878	2	240.672	648.3	> 0.9065	0.1462 (21)	0.0885 (13)	
					994.9	> 1.1070			
1911.544	4^+	1365.878	2	545.7	648.3	< 0.0209	0.01723 (25)	0.00911 (13)	
					994.9	< 0.0189			
1911.544	4^+	1606.55	γ	304.75	888.69	< 0.0794	0.0777 (11)	0.0440 (7)	0.042 (6)

Table 4.10: A list of conversion coefficients from ^{154}Gd for $6^+ \rightarrow 6^+$ transitions seen in the gated data. The first error is statistical, the second is systematic. Numbers are compared with theoretical K-shell conversion coefficients for M1 and E2 transitions, as well as results from Gono et al.[25] All coefficients are K-electrons. Whole numbers for the bands are the 0^+ bands.

done for the 0^+ transitions, as $M1$ and $E2$ transitions are not allowed. Most of the values calculated are upper or lower limits, as the original α obtained was an upper or lower limit.

With these values, two transitions from the same level can be compared using the $B(E0)$ formula to take the energy adjustment into account (equation 1.42). This ratio must then be corrected by the ratio of the gate efficiencies, as the intensities of the transitions are unknown, and the efficiency correction gives absolute numbers to compare. Because some transitions could be seen in multiple gates, these ratios could be calculated using several corrected numbers. These results are summarized in Table 4.14.

4.4.2.1 $K^\pi = 0_2^+$, First excited 0^+ band, 680.6673 keV

The $J^\pi \rightarrow J^\pi$ transitions between the first excited 0^+ band and the ground state band appear to be quite strong. Lifetimes beyond the 4^+ state in the band are unknown, inhibiting the ability to calculate ρ^2 for these transitions. In gating, the transitions between the 2^+ , 4^+ , and 6^+ states to the respective J^π of the ground state were seen. All three transitions indicated a strong E0 component. The $0^+ \rightarrow 0^+$ and $8^+ \rightarrow 8^+$ transitions could be seen in the singles, but not enough statistics were present in gates. The $2^+ \rightarrow 2^+$ transition agreed with previous measurements from Spits [60]. The $4^+ \rightarrow 4^+$ transition appears to be higher than the measurements from Spits and Gono [25]. Similarly, the $6^+ \rightarrow 6^+$ transition appears to be larger than Gono.

Looking at Table 4.13, the first 0^+ excited band to the ground state has several transitions with known mixing ratios and lifetimes. The E0 mixing ratio, ϵ^2 and the matrix element $\rho^2(E0)$ are calculated.

TABLE 4.11
E0 CONTRIBUTIONS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS

E_i (keV)	Band	E_f (keV)	Band		E (keV)	Gate	$q^2\alpha(E2)$
$2^+ \rightarrow 2^+$							
996.264	γ	815.4917	2	602.688	692.4205		> 1.584
				444.4924			> 1.414
1418.16	3	815.4917	2	602.688	692.4205		> 0.0044
1418.16	3	996.2568	γ	421.893	625.2556		> 0.0477
					873.1834		> 0.0285
1531.305	2^+	815.4917	2	715.819	692.4205		$0.0157 (43)_{-35}^{+46}$
					444.4924		$0.0333 (114)_{-74}^{+97}$
1531.305	2^+	996.2568	γ	535.050	873.1834		$0.0131 (45)_{-26}^{+35}$
					625.2556		> 0.0089
1716.050	6	815.4917	2	900.5583	692.4205		< 0.0131
					444.4924		< 0.0983
1716.050	6	996.2568	γ	719.80	873.1834		$0.0092 (38)_{-20}^{+19}$
					625.2556		$0.0868 (451)_{-196}^{+255}$
1775.429	7	815.4917	2	960.05	692.4205		> 0.0374
					444.4924		> 0.0394
1775.429	7	996.2568	γ	779.165	873.1834		$0.0301 (164)_{-67}^{+88}$
					625.2556		$0.1379 (965)_{-305}^{+401}$
$4^+ \rightarrow 4^+$							
1645.814	4^+	1263.778	γ	382.025	892.775		< 0.0059
					1140.702		< 0.0033
1701.39	3	1047.592	2	653.7	676.593		< 0.0017
					924.55		< 0.0433
1701.39	3	1263.778	γ	437.612	892.775		< 0.0708
					1140.702		< 0.0560
1789.17	2^+	1047.592	2	740.91	676.593		< 0.0123
					924.55		< 0.0769

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	$q^2\alpha(E2)$
1789.17	2 ⁺	1263.778	γ	525.392	892.775	< 0.0046
					1140.702	< 0.0319
$6^+ \rightarrow 6^+$						
1606.55	γ	1365.878	2	240.672	648.3	> 1.5783
					994.9	> 1.9793
1911.544	4 ⁺	1365.878	2	545.7	648.3	< 0.0155
					994.9	< 0.0115
1911.544	4 ⁺	1606.55	γ	304.75	888.69	< 0.0371

Table 4.11: A list of $E0$ contributions in ^{154}Gd , assuming a mixing ratio between the E2 and M1 contributions of $\delta = 1$. These values have not been normalized, as the lifetime of the states are unknown. Table 4.14 compares values between two transitions of the same initial state. Only non-negative values are listed in the table, and δ was assumed to be 1, as no mixing ratios are known for these transitions. For $\alpha(exp)$, $\alpha(M1)$, and $\alpha(E2)$ used in these calculations, please refer to Tables 4.8-4.10. Bands are listed. Whole numbers refer to 0^+ bands.

TABLE 4.12

$q_K^2(E0/E2)$ FOR $0^+ \rightarrow 0^+$ TRANSITIONS

E_i (keV)	Transition	$E0$ (keV)	Transition	$E2$ (keV)	$q_K^2(E0/E2)$
1182.091	$0_3^+ \rightarrow 0_2^+$	501.427	$0_3^+ \rightarrow 2_{gs}^+$	1059.033	0.0023 (5)
1573.9	$0_6^+ \rightarrow 0_3^+$	391.85	$0_6^+ \rightarrow 2_{gs}^+$	1451.7	0.0521 (119)
1573.9	$0_6^+ \rightarrow 0_2^+$	893.9	$0_6^+ \rightarrow 2_{gs}^+$	1451.7	0.0168 (77)
1650.3	$0_7^+ \rightarrow 0_3^+$	468.3	$0_7^+ \rightarrow 2_{gs}^+$	1527.1	0.2082 (345)
1650.3	$0_7^+ \rightarrow 0_2^+$	970.3	$0_7^+ \rightarrow 2_{gs}^+$	1527.1	0.0402 (192)

Table 4.12: A list of $q_K^2(E0/E2)$ contributions in ^{154}Gd for the $0^+ \rightarrow 0^+$ transitions. These values cannot be converted to nuclear strengths, ρ^2 as the lifetimes are unknown.

TABLE 4.13

E0 CONTRIBUTIONS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS WITH KNOWN
MIXING RATIOS

E_i (keV)	Band	E_f (keV)	Band	E (keV)	δ	$t_{1/2}$ (ps)	Gate	ϵ^2	$\rho^2(E0)$
$2^+ \rightarrow 2^+$									
815.4917	2	123.0709	GS	692.4205	7.5 (4)	6.4 (4)	123.0706	1.9211 (1112) (402)	53.74 (900) (112)
$4^+ \rightarrow 4^+$									
1047.592	2	370.9998	GS	676.593	2.9 (4)	7.6 (4)	247.9288	0.4274 (60) $^{+93}_{-85}$	14.46 (335) $^{+31}_{-29}$
1645.814	4^+	1047.592	2	598.22	0.65 (20)		676.593	< 0.00003	
							924.55	< 0.00714	
$6^+ \rightarrow 6^+$									
1365.878	2	717.662	GS	648.3	1.30 (20)		346.643	0.1843 (286) (38)	

Table 4.13: A list of $E0$ contributions in ^{154}Gd for data with known mixing ratios, δ . These values have not been normalized, as the lifetime of the states are unknown. Table 4.14 compares values between two transitions of the same initial state. Lifetimes are from [61]. For $\alpha(exp)$, $\alpha(M1)$, and $\alpha(E2)$ used in these calculations, please refer to Tables 4.8-4.10. Whole numbers indicate 0^+ bands.

TABLE 4.14
 $B(E0)$ RATIOS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS

E_i (keV)	Band	$E_{0_2^+}$ (keV)	Gate $_{0_2^+}$	E_γ (keV)	Gate $_\gamma$	$B(E0)$ Ratio
$0^+ \rightarrow 0^+$						
1573.9	6	680.6673	557.581	1182.091	1059.033	0.219
1650.3	7	680.6673	557.581	1182.091	1059.033	0.090
$2^+ \rightarrow 2^+$						
1418.16	3	815.4917	692.4205	996.2568	625.2556	0.068
					873.1834	0.099
1531.305	2 ⁺	815.4917	444.4924	996.2568	625.2556	< 2.416
					873.1834	1.438 (698) ₋₄₃₁ ⁺⁵⁶⁵
		692.4205			625.2556	< 1.363
					873.1834	0.812 (358) ₋₂₄₅ ⁺³²¹
1716.050	6	815.4917	444.4924	996.2568	625.2556	< 0.786
					873.1834	< 6.473
		692.4205			625.2556	< 0.126
					873.1834	< 0.786
1775.429	7	815.4917	444.4924	996.2568	625.2556	> 0.201
					873.1834	> 0.807
		692.4205			625.2556	> 0.229
					873.1834	> 0.912
$4^+ \rightarrow 4^+$						
1645.814	4 ⁺	1047.592	924.55	1263.778	892.775	0.817
					1140.702	0.133
1701.39	3	1047.592	676.593	1263.778	892.775	0.015

E_i (keV)	Band	$E_{0_2^+}$ (keV)	Gate $_{0_2^+}$	E_γ (keV)	Gate $_\gamma$	$B(E0)$ Ratio
			924.55		1140.702	0.0167
					892.775	0.416
					1140.702	0.474
1789.17	2 ⁺	1047.592	676.593	1263.778	892.775	1.702
					1140.702	2.207
			924.55		892.775	12.052
					1140.702	15.622
$6^+ \rightarrow 6^+$						
1911.544	4 ⁺	1365.878	648.3	1606.55	888.69	0.205
			994.9			0.181

Table 4.14: Ratios of the $B(E0)$ values in ^{154}Gd . Only ratios between two transitions of the same state are listed, as the lifetime of the states are unknown. Table 4.11 lists the values that were used in the calculation. The gates are included, as an efficiency correction was made on the ratio based on the gates. In many cases, only upper or lower limits for the values could be used for this calculation. Errors are not given on these values. Those values marked with errors or as limits had defined values instead of limits. Whole numbers for bands indicate 0⁺ bands.

4.4.2.2 $K^\pi = 0_3^+$, Second excited 0⁺ band, 1182.091 keV

All of the measurements of transitions coming out of the second excited state were limits. For the 0⁺ state (1182.091 keV), the gates were used to look for the transition to the first excited 0⁺ state. The lower limit on the transition is lower than that previously measured by Spits. The limit in this work comes from the background of the HPGe detector. No further transitions could be seen above 4⁺, as the band is only known up to this J^π .

For the 2⁺ state (1418.16 keV), two transitions were examined, one to the 2⁺ in the first 0⁺ excited state and one to the head of the γ -band. Each of these transitions

was measured with two different gates. The lower limits are consistent with the Spits data. For the transition to the γ -band, there appears to be an E0 component. The lower limit in the transition to the first excited 0^+ state is not conclusive. For the 4^+ state (1701.39 keV), two transitions were examined, one to the 2^+ in the first 0^+ excited state and one to the head of the γ -band. Each of these transitions was measured with two different gates. These transitions only have upper limits. For the transition to the γ -band, the upper limit is not close to the M1 conversion coefficient, indicating there may be some E0 strength. This cannot be stated conclusively, as it is only an upper limit. The upper limits in the transition to the first excited 0^+ state from the two different gates disagree. One upper limit agrees with the previous measurement from Spits, and would indicate there may be E0 strength [60]. The other upper limit indicates a mixed M1+E2 transition with no or very little E0 strength.

Looking at Table 4.14, the second 0^+ excited band seems to strongly favor transitioning to the γ -band, over transitioning to the first 0^+ excited band. None of the transitions examined have known mixing ratios, so the q^2 and αq^2 can be found in Tables 4.11 and 4.12.

4.4.2.3 $K^\pi = 0_6^+$, Fifth excited 0^+ band, 1573.9 keV

For the 0^+ state (1573.9 keV), the gates were used to look for the transition to the first and second excited 0^+ states. The lower limit on the transition to the second excited 0^+ state is lower than that previously measured by Spits. The limit in this work comes from the background of the HPGe detector. The transition to the first excited 0^+ state had not previously been measured, making this lower limit the first measurement. No further transitions could be seen above 4^+ , as the band is only known up to this J^π . The 4^+ state is not seen, as it is at approximately the same energy as the population cut off.

For the 2^+ state (1716.050 keV), two transitions were examined, one to the 2^+ in the first 0^+ excited state and one to the head of the γ -band. Each of these transitions was measured with two different gates. The upper limits to the first excited 0^+ band would indicate a small E0 component if one exists. For the transition to the γ -band, the values were measurable. The conversion coefficients found in both gates indicate an E0 component. Not enough statistics were present to see any transition between this 2^+ state and the 2^+ state of the second excited 0^+ band.

Looking at Table 4.14, the fifth 0^+ excited band seems to favor transitioning to the γ -band, over transitioning to the first 0^+ excited band. None of the transitions examined have known mixing ratios, so the q^2 and αq^2 can be found in Tables 4.11 and 4.12.

4.4.2.4 $K^\pi = 0_7^+$, Sixth excited 0^+ band, 1650.3 keV

For the 0^+ state (1650.3 keV), the gates were used to look for the transition to the first and second excited 0^+ states. The transition to the fifth state would emit an electron lower than the thresholds on the detectors, making the measurement infeasible with the current set up. The lower limit on the transition to the first excited 0^+ state is lower than that previously measured by Spits, but of similar magnitude. The transition to the second excited 0^+ state had not previously been measured, making this lower limit the first measurement. No further transitions could be seen above 2^+ , as the band is only known up to this J^π , as proposed by Spits[60].

For the 2^+ state (1775.429 keV), two transitions were examined, one to the 2^+ in the first 0^+ excited state and one to the head of the γ -band. Each of these transitions was measured with two different gates. The lower limits to the first excited 0^+ band are in good agreement, and would indicate an E0 component if one exists. For the transition to the γ -band, the values were measurable. The

conversion coefficients found in both gates indicate an E0 component. Not enough statistics were present to see any transition between this 2^+ state and the 2^+ state of the second excited 0^+ band.

Looking at Table 4.14, the sixth 0^+ excited band seems to favor transitioning to the γ -band, over transitioning to the first 0^+ excited band, but as the ratio is a lower limit, this cannot be confirmed. None of the transitions examined have known mixing ratios, so the q^2 and αq^2 can be found in Tables 4.11 and 4.12.

4.4.2.5 $K^\pi = 2^+, \gamma$ band, 996.264 keV

Transitions for the 2^+ (996.264 keV), 4^+ (1263.778 keV), and 6^+ (1606.55 keV) to the first excited 0^+ band could be seen in the gates. The $2^+ \rightarrow 2^+$ transition are lower limits, indicating a large E0 component. The $6^+ \rightarrow 6^+$ transition shows a similar lower limit, indicating a large E0 component. Conversely, the $4^+ \rightarrow 4^+$ transition has upper limits pointing to a largely E2 transition. This makes conclusions difficult to draw.

None of the states in the γ -band had multiple outgoing transitions measured. No transitions examined have known mixing ratios, so the q^2 and αq^2 can be found in Table 4.11.

4.4.2.6 $K^\pi = 2^+$, Second excited 2^+ band, 1531.305 keV

Transitions for the 2^+ (1531.305 keV) and 4^+ (1789.17 keV) to the first excited 0^+ band and γ -band could be seen in the gates. For the 2^+ transition to the first excited 0^+ band, the values are large for both gates, disagreeing with the value found by Spits [60]. It is worth noting the large error on the values from this work indicates the disagreement may be purely statistical. This would indicate a small E0 component if true. For the transition to the γ -band, there is one value and one lower limit. The lower limit agrees with the measured value, which is much smaller

than Spits, but would indicate an E0 component, as Spits' value does.

The transitions for the 4^+ state are all upper limits. To the first excited 0^+ band, the upper limits do not rule out the possibility of an E0 component. For the transition to the γ -band, the upper limits from the two gates agree, firmly placing the transition as an M1+E2 transition, if not a pure E2.

Looking at Table 4.14, the second 2^+ excited band seems to either not favor transitioning to the γ -band over the first 0^+ excited band, or prefer the first excited 0^+ band. None of the transitions examined have known mixing ratios, so the q^2 and αq^2 can be found in Table 4.11.

4.4.2.7 $K^\pi = 4^+$, First excited 4^+ band, 1645.814 keV

Transitions for the 4^+ (1645.814 keV) and 6^+ (1911.544 keV) to the first excited 0^+ band and γ -band could be seen in the gates. All the values are upper limits. For the 4^+ transition to the first excited 0^+ band, the upper limits are lower than the one set by Spits [60]. The lower of the two upper limits would indicate a mostly E2 transition. The higher of the two would indicate a mixed M1+E2, with little to no E0 component. The upper limits on the transition to the γ -band agrees with the value measured by Spits, which would indicate a mixed M1+E2 transition. Without a known mixing ratio, it is not possible to tell if there is an E0 component.

For the 6^+ state, the upper limits on the transition to the first excited 0^+ band indicate there may be a small E0 component, as both upper limits are higher than the M1 theory strength. Only one gate could be used to look at the transition to the γ -band. The upper limit agrees with the previous measurement from Gono, which would indicate an E2 transition [25]. This work's upper limit does not eliminate the possibility of a mixed transition.

Looking at Table 4.14, the first 4^+ excited band seems to favor transitioning to the γ -band over the first 0^+ excited band. One of the transitions examined has a known

mixing ratio, δ^2 , which allows for the calculation of the E0 mixing ratio, ϵ^2 , listed in Table 4.13. For the other transitions, the q^2 and αq^2 can be found in Table 4.11.

CHAPTER 5

^{156}Gd RESULTS

^{156}Gd is one of the most well studied nuclei in the rare-earth region. It has a large number of known lifetimes, having been studied by both (n, γ) and (n, e^-) [33]. Only two other nuclei, ^{162}Dy and ^{168}Er have been studied to a similar extent. It is known to have 6 excited 0^+ states, three with known lifetimes, enabling the direct calculation of the nuclear matrix element with the observation of an E0. This nucleus has not been studied by (p, t) since 1973, long before the renaissance of 0^+ states being discovered via (p, t) in the early 2000s [24, 38, 45]. Further lifetimes and known δ mixing ratios enable similar calculations for E0 components of $J^\pi \rightarrow J^\pi$ transitions.

This data was taken with ICEBall and GEORGINA. The experiment used an enriched ^{154}Sm target of 1.7 mg/cm^2 thickness, as discussed in Chapter 2 and shown in Table 2.10. A complete catalog of spectra used in analysis can be found in Appendix B.

5.1 Ground State Band Confirmation

The singles spectra show several prominent peaks in Figure 5.1. In the γ -spectrum on top, there are three prominent peaks from 100 to 400 keV. These peaks are the ground state band transitions from the 2^+ state up to the 8^+ state. The transition from the 2^+ state to the ground state is only 89 keV, and while visible, suffers from a steep efficiency drop-off at that energy, and a large background due to x-rays.

The strong peak just beyond 500 keV is the 511 keV annihilation peak. The

conversion electron spectrum beneath has clear K -shell and L -shell peaks corresponding to these ground state band transitions.

The transitions in the ground state band are used for confirmation of the data.

These are all pure E2 multipole transitions, making them a diagnostic to compare with the theoretical coefficients from BrIcc[32]. Singles data had to be corrected for angular distribution effects, discussed in section 3.3.3. The transitions from 2^+ to 10^+ are summarized and compared with theory in Table 5.1.

Unlike the ^{154}Gd data, there do not appear to be contaminants in these ground state lines, outside of the $2_{gs}^+ \rightarrow 0_{gs}^+$ transition. This transition is also considered unreliable, as the low energy of the γ -ray puts it in the energy region where the efficiency decreases due to the attenuation of the photon before it reaches the active region of the detector. The change in efficiency in the HPGe detectors is steep in this region, leading to large uncertainty.

5.2 Gates on the Ground State Band

The transitions of the ground state band are the most prominent peaks in the spectra (Figure 5.1). The high number of counts in these peaks made them prime candidates to gate on and confirm the levels populated. Only the γ components were gated on, as the electron components' wide width made it difficult to gate on the whole of the peak without contaminants. Figures 5.2, 5.3, 5.4, and 5.5 are the result of these gates. Unlike the gates in ^{154}Gd , the $4_{gs}^+ \rightarrow 2_{gs}^+$ gate was the most fruitful, due to the challenges the attenuation of the $2_{gs}^+ \rightarrow 0_{gs}^+$ transition posed. Determining which transitions went uniquely into a given ground state level was done by comparing the outgoing ground state transition for that level with the incoming transitions, i.e. the $4_{gs}^+ \rightarrow 2_{gs}^+$ (199 keV) gate was compared directions with the $6_{gs}^+ \rightarrow 4_{gs}^+$ (296 keV) gate. The γ -spectra corresponding to gates of two of these lines can be seen in Figure 5.6. Lines in the gamma spectrum that were

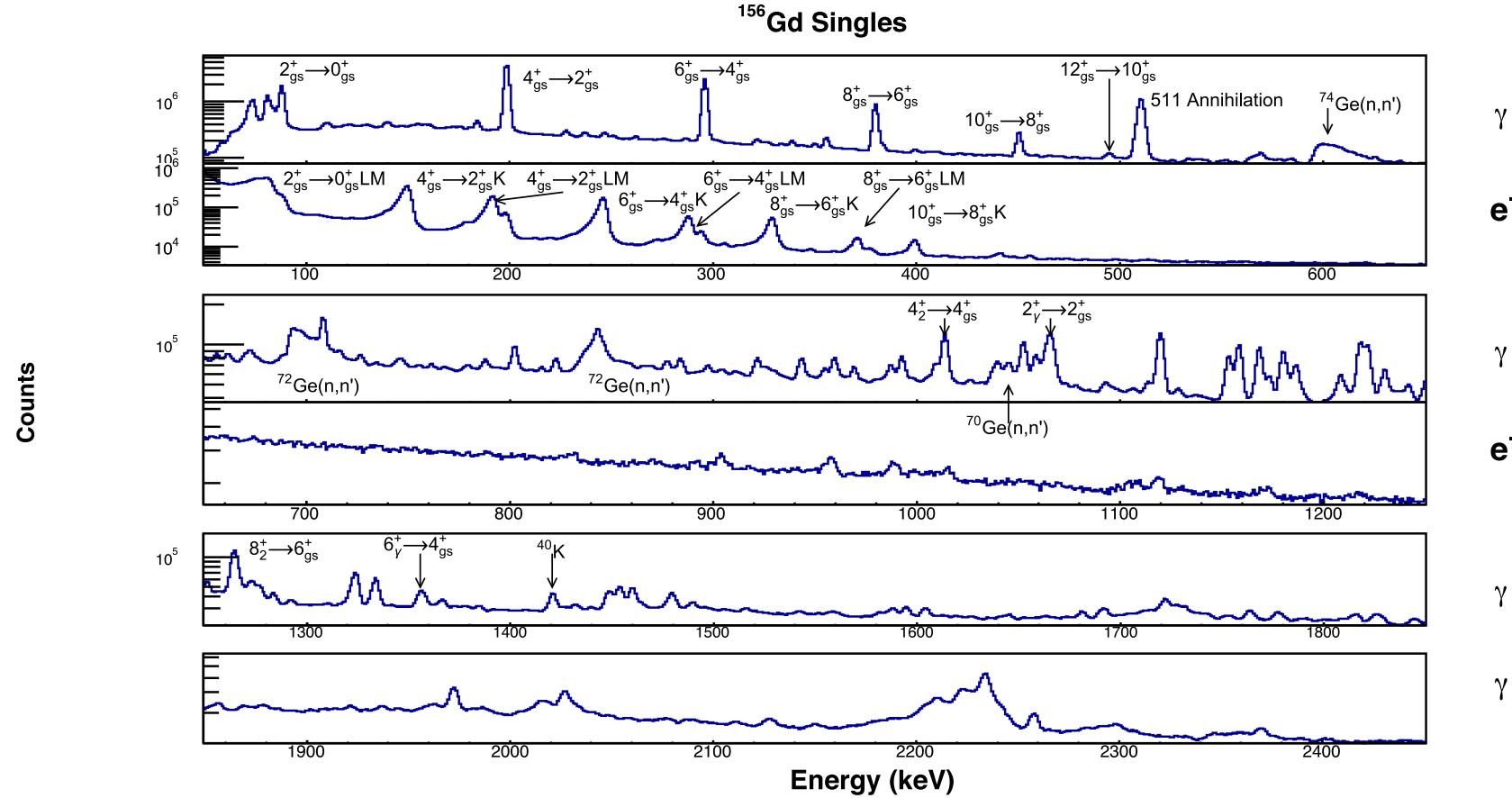


Figure 5.1. Singles spectra of ^{156}Gd . Spectra are labeled with the particles being detected, the energies of the γ and electron spectra aligned for identification. In the γ spectrum, several lines of note are labeled. The large peak in the electron spectrum at low energy is cut off due to the threshold. It is a combination of background and the 89L peak.

Transitions in the higher energy regime of the γ spectrum cannot be determined without gating, due to additional background from the experimental room. The electron spectrum only goes to 1250 keV, as that is the limit of the detectors.

TABLE 5.1

$$^{156}\text{GD}$$
 GROUND STATE BAND INTERNAL CONVERSION COEFFICIENTS FROM
SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	Shell	α (This Work)	α (Th)	α (Konijn)
198.58	$4^+ \rightarrow 2^+$	288.187	88.97	111900	E2	K	$0.1667 (4)^{+46}_{-45}$	0.1565 (22)	0.199 (36)
						L	$0.0537 (1)^{+16}_{-15}$	0.0531 (8)	
						M	$0.0170 (1) (5)$	0.0122 (2)	
296.04	$6^+ \rightarrow 4^+$	584.715	288.187	15800	E2	K	$0.0572 (1) (18)$	0.0477 (7)	0.04683*
						L	$0.0121 (1) (4)$	0.0115 (2)	
						M	$0.0036 (1) (1)$	0.0026 (1)	
379.92	$8^+ \rightarrow 6^+$	965.134	584.715	4320	E2	K	$0.0274 (1) (9)$	0.0235 (4)	0.038 (10)
						L	$0.0050 (1) (2)$	0.0048 (1)	
						M	$0.0013 (1) (1)$	0.0011 (1)	
450.64	$10^+ \rightarrow 8^+$	1416.078	965.134	1900	E2	K	$0.0152 (2) (5)$	0.01483 (21)	0.0145*
						L	$0.0028 (1) (1)$	0.00279 (4)	
						M	$0.0010 (1) (1)$	0.000621 (9)	

A list of the ground state conversion coefficients from ^{156}Gd . Multipolarities and mixing ratios were taken from NNDC. Unless otherwise stated, the α values are α_K . An angular distribution correction has been applied based on multipolarities for pure transitions, and those with known mixing ratios. The first error is statistical, the second is systematic. Numbers are compared with Konijn et al. [35] Starred values in the Konijn data were used as calibration points.

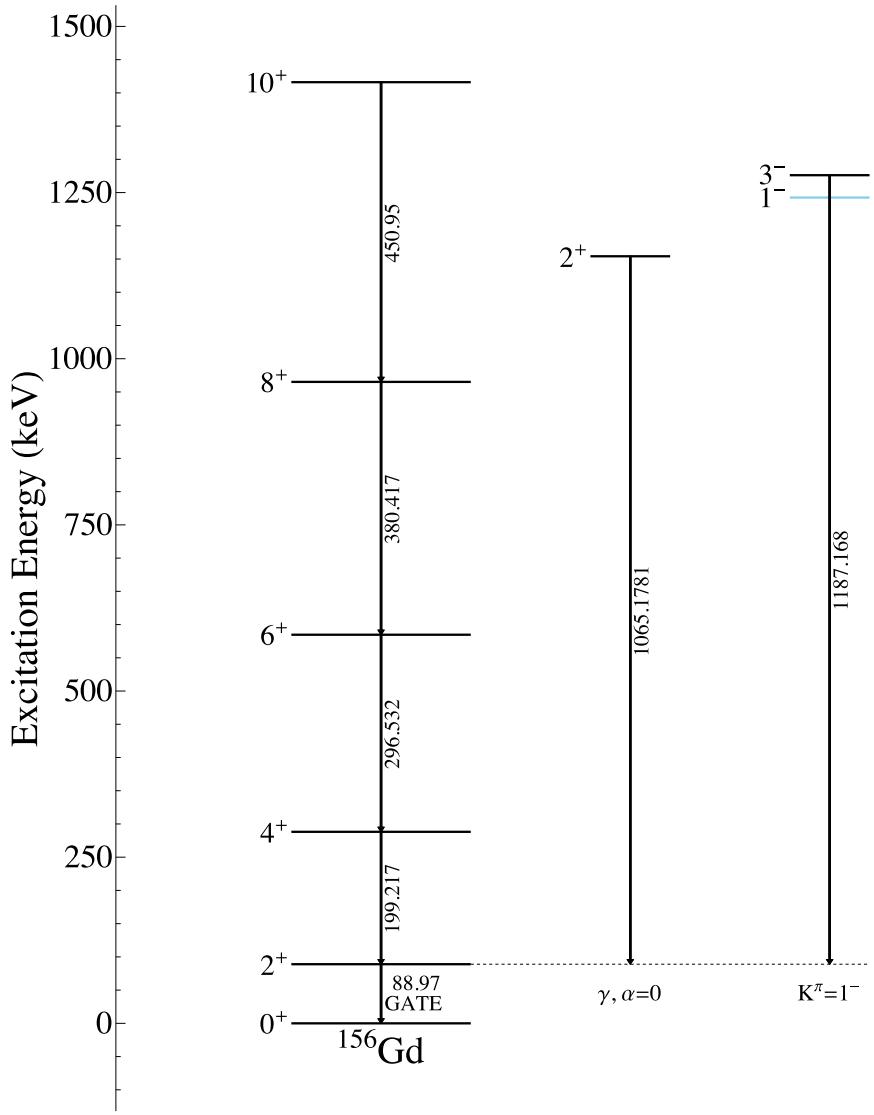


Figure 5.2(a) Level Scheme of ^{156}Gd . The gamma ray of the $2^+ \rightarrow 0^+$ (89 keV) transition in the ground state was gated on. It was then compared with the gated spectrum from the gamma ray of the $4^+ \rightarrow 2^+$ (199 keV) transition in the ground state. Peaks only appearing in the first gate were assumed to go into the 2^+ state, and assignments were made. Due to the low energy of the $2^+ \rightarrow 0^+$ transition, the efficiency was lower, and it is likely that transitions into the 2^+ state were missed. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

89 keV gated spectra

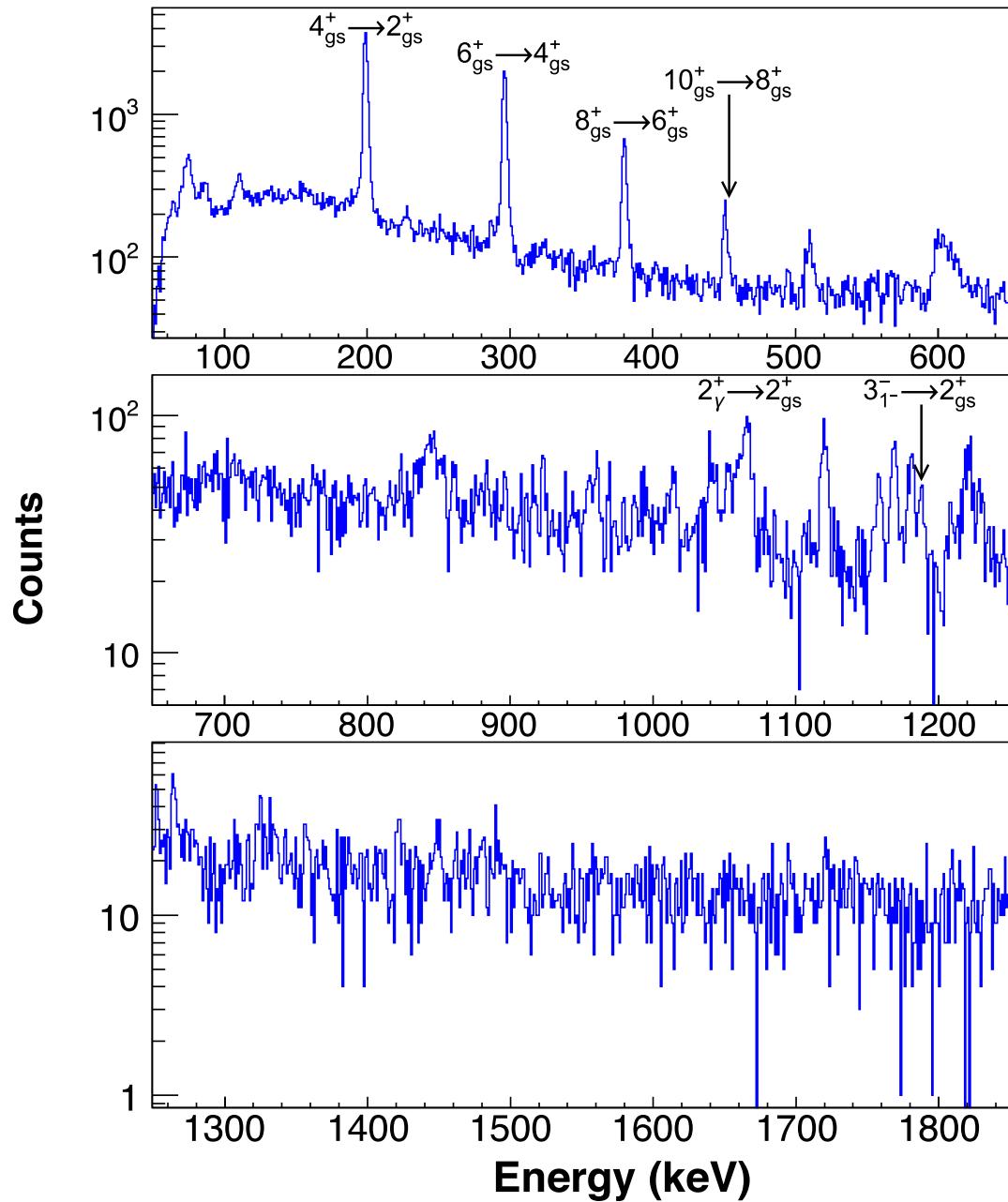


Figure 5.2(b) Gamma spectrum gated on 89 keV, corresponding to the $2^+ \rightarrow 0^+$ transition. Several transitions are marked according to the level scheme.

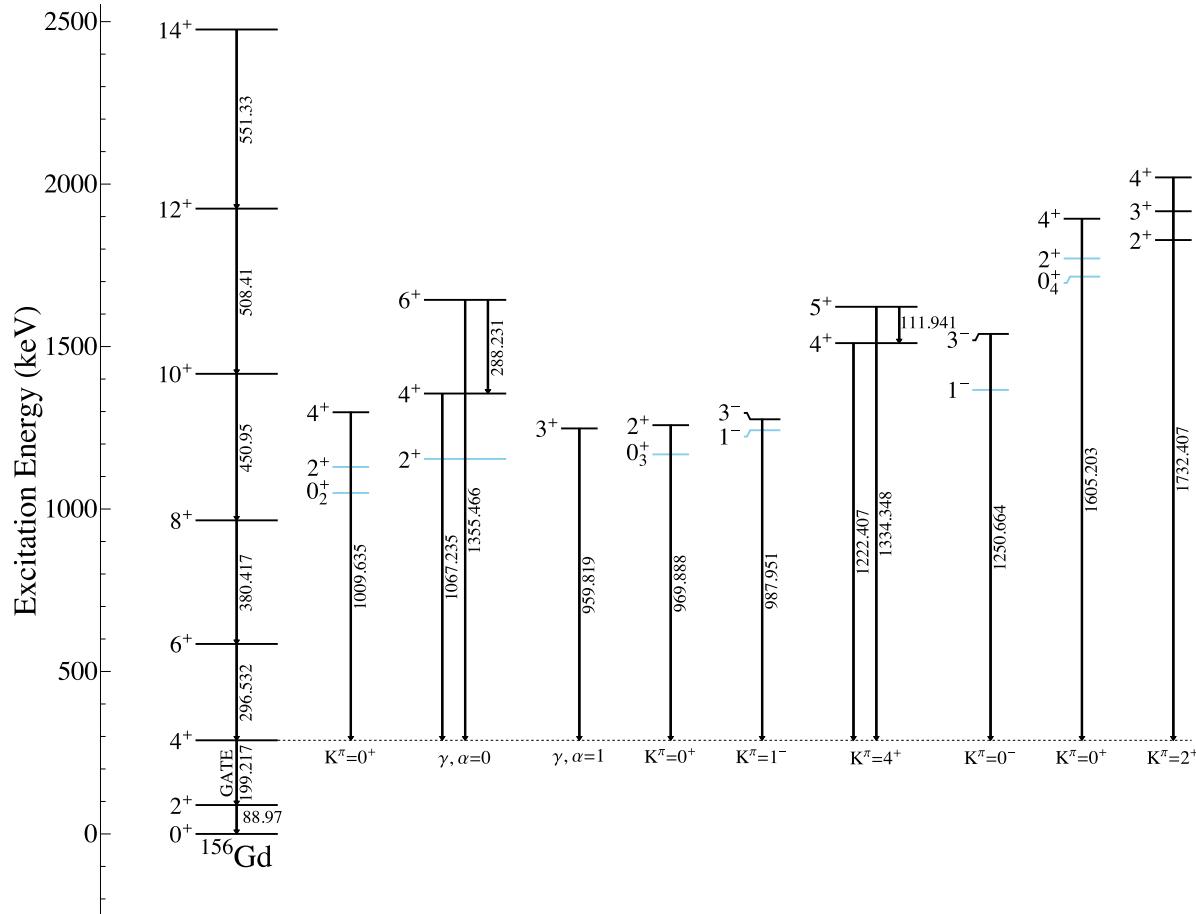


Figure 5.3(a) Level Scheme of ^{156}Gd . The gamma ray of the $4^+ \rightarrow 2^+$ (199 keV) transition in the ground state was gated on. It was then compared with the gated spectrum from the gamma ray of the $6^+ \rightarrow 4^+$ (296 keV) transition in the ground state. Peaks only appearing in the first gate were assumed to go into the 4^+ state, and assignments were made. Additionally, these peaks were also gated on, to look for cascades leading into the 4^+ state, which were found in several cases. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

199 keV gated spectra

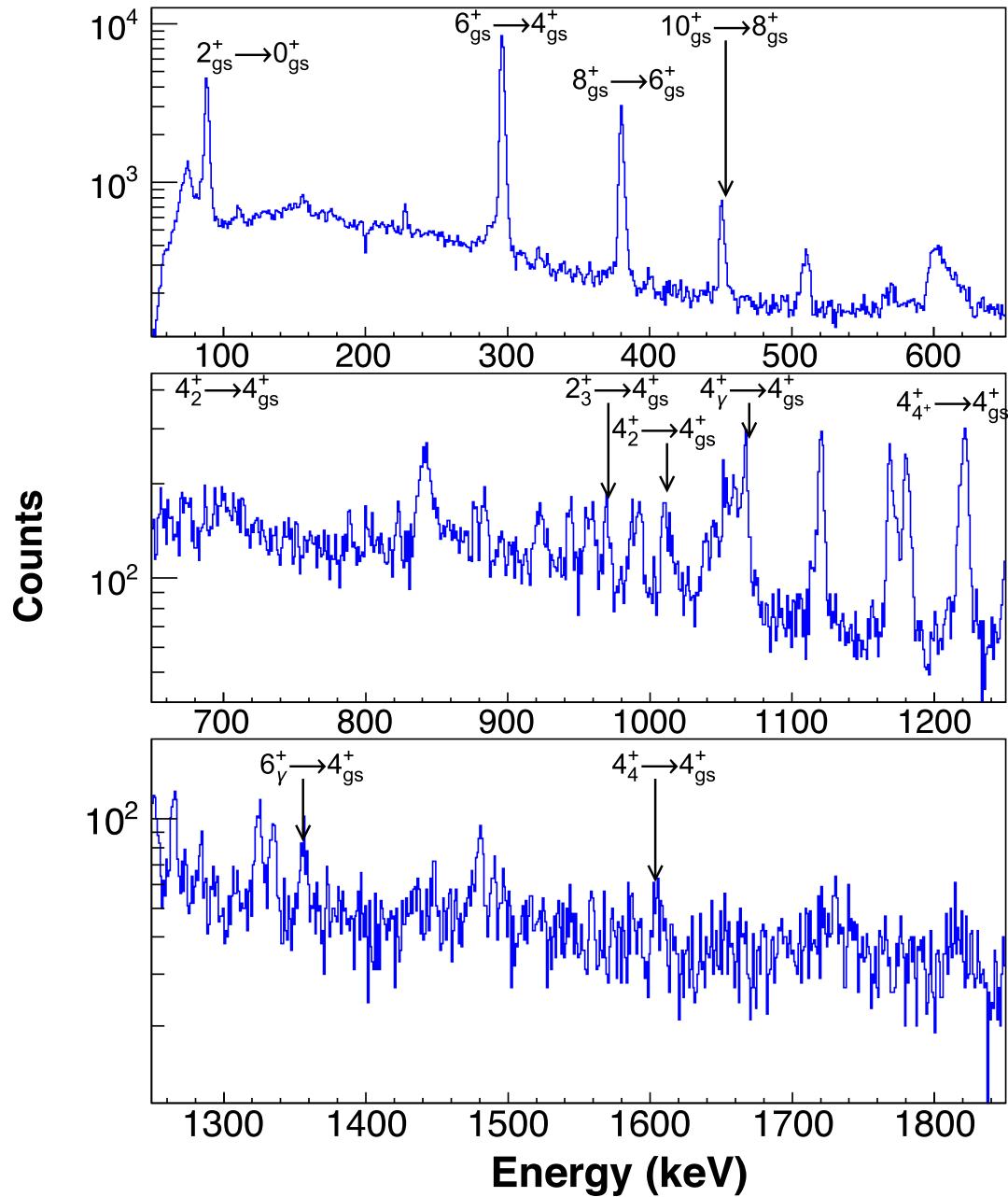


Figure 5.3(b) Gamma spectrum gated on 199 keV, corresponding to the $4^+ \rightarrow 2^+$ transition. Several transitions are marked according to the level scheme.

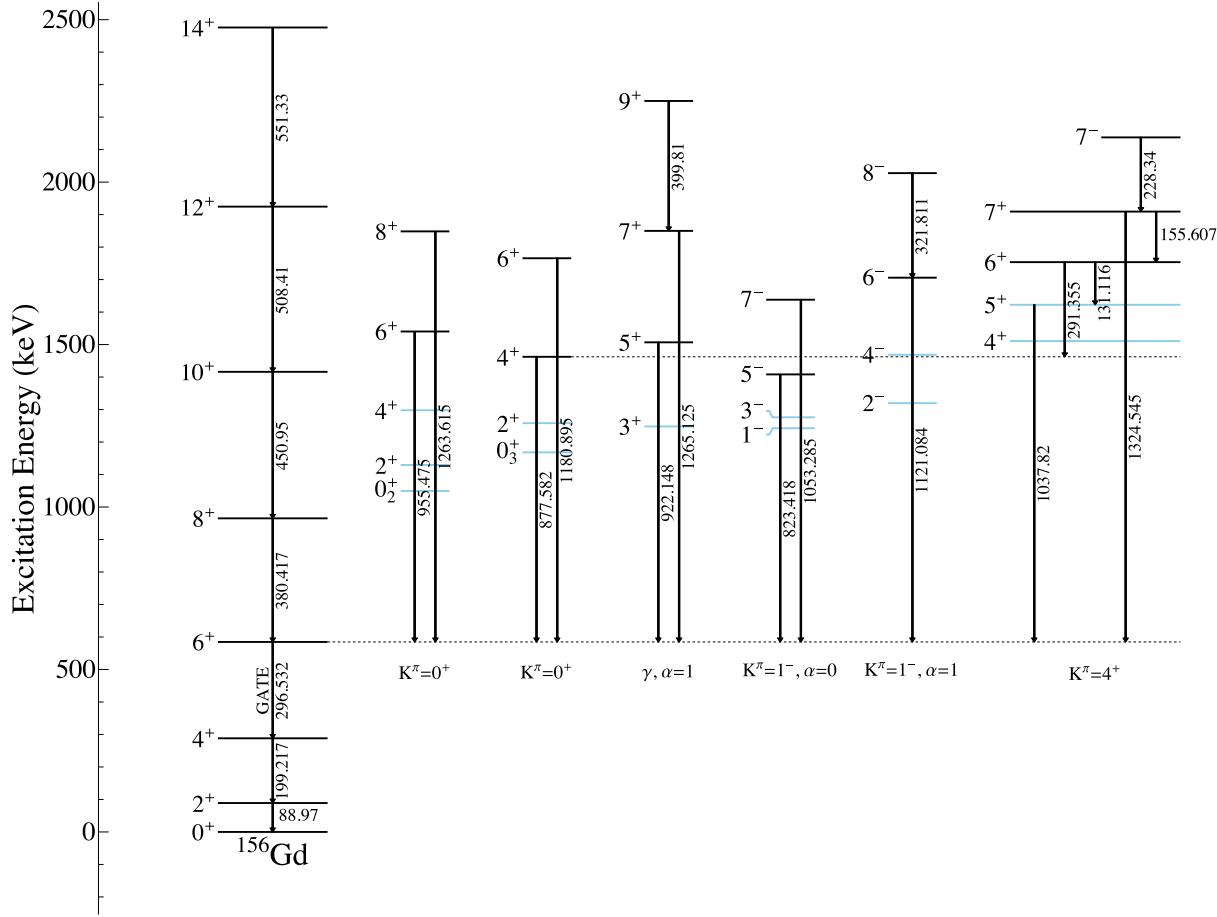


Figure 5.4(a) Level Scheme of ^{156}Gd . The gamma ray of the $6^+ \rightarrow 4^+$ (296 keV) transition in the ground state was gated on. It was then compared with the gated spectrum from the gamma ray of the $8^+ \rightarrow 6^+$ (380 keV) transition in the ground state. Peaks only appearing in the first gate were assumed to go into the 6^+ state, and assignments were made. Additionally, these peaks were also gated on, to look for cascades leading into the 6^+ state, which were found in several cases. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

296 keV gated spectra

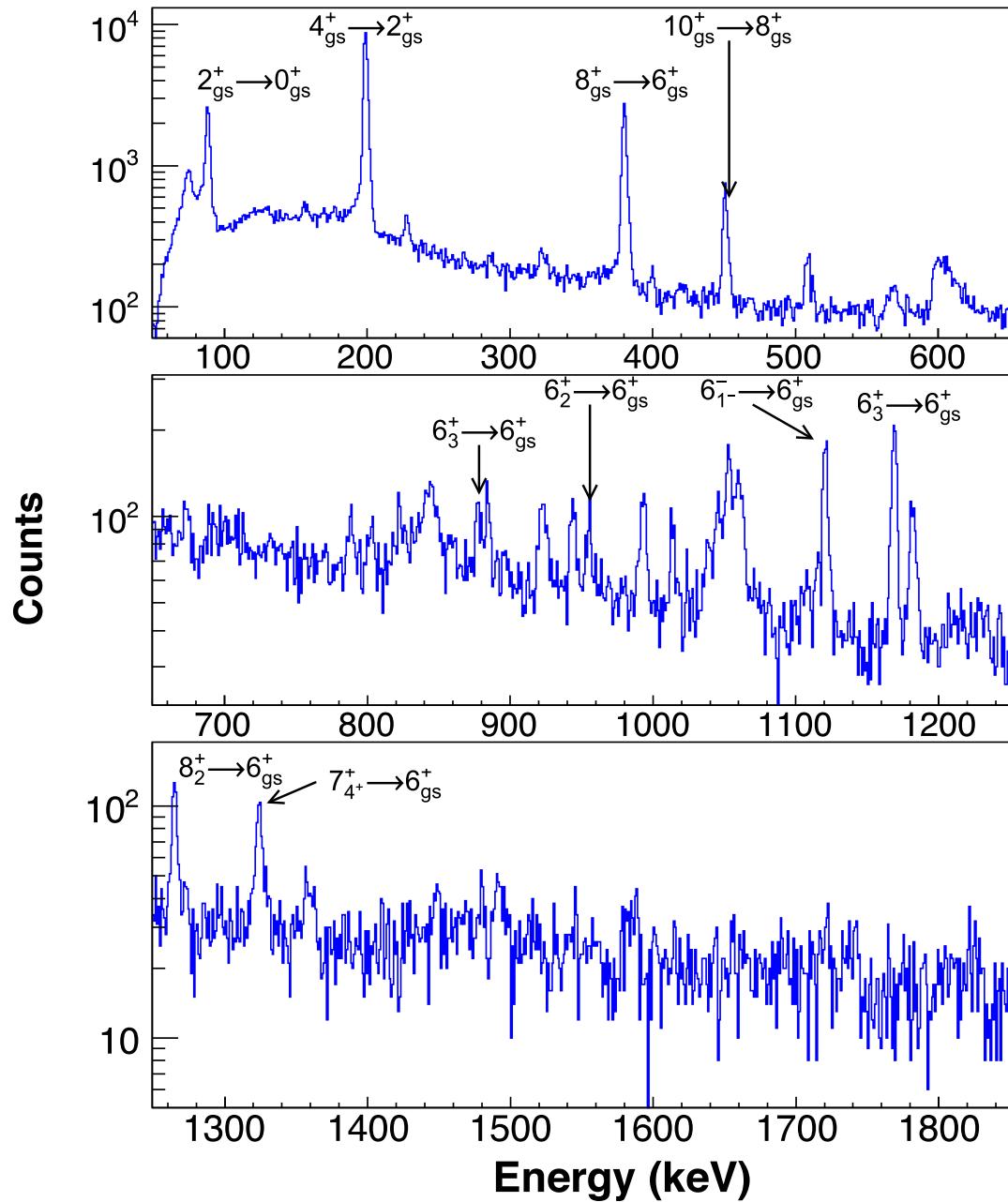


Figure 5.4(b) Gamma spectrum gated on 296 keV, corresponding to the $6^+ \rightarrow 4^+$ transition. Several transitions are marked according to the level scheme.

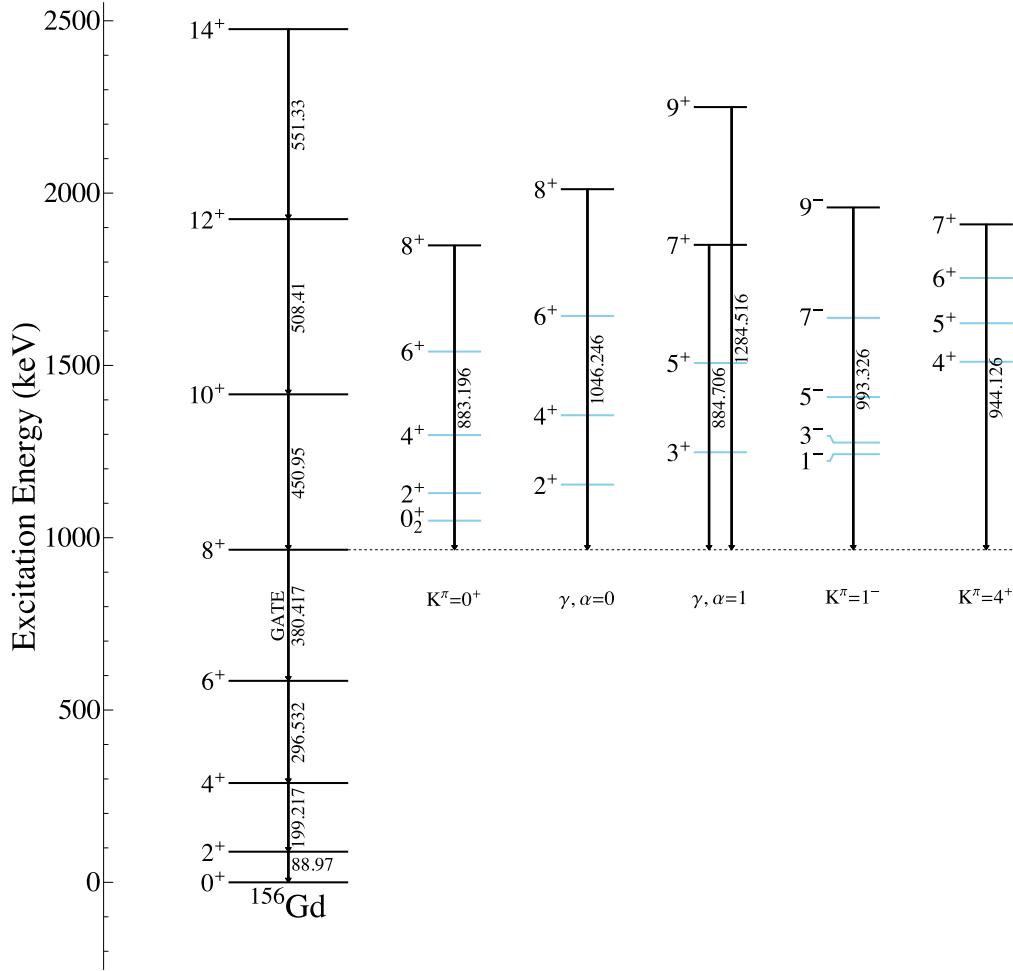


Figure 5.5(a) Level Scheme of ^{156}Gd . The gamma ray of the $8^+ \rightarrow 6^+$ (380 keV) transition in the ground state was gated on. It was then compared with the gated spectrum from the gamma ray of the $10^+ \rightarrow 8^+$ (451 keV) transition in the ground state. Peaks only appearing in the first gate were assumed to go into the 8^+ state, and assignments were made. Additionally, these peaks were also gated on, to look for cascades leading into the 8^+ state, which were found in several cases. The levels are organized by band. The lower levels of the band, unseen by gamma rays in this gate, are in blue.

present only in the outgoing spectrum, but not the incoming spectrum, are going into that level (the 4^+_gs in the example above). Several areas have been circled to demonstrate this effect. Identified transitions were then gated on to confirm assignments and search for cascades from higher energy states.

380 keV gated spectra

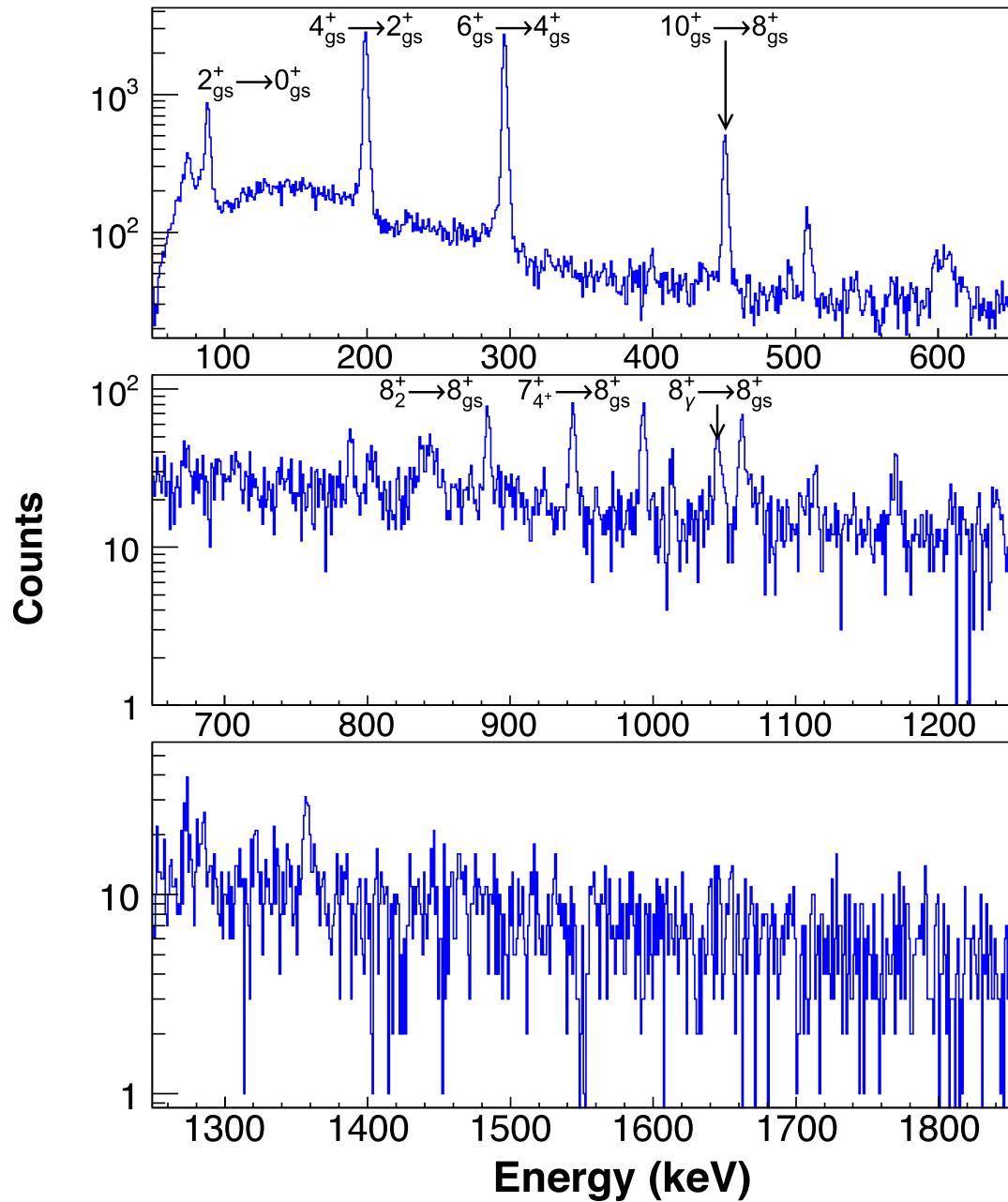


Figure 5.5(b) Gamma spectrum gated on 380 keV, corresponding to the $8^+ \rightarrow 6^+$ transition. Several transitions are marked according to the level scheme.

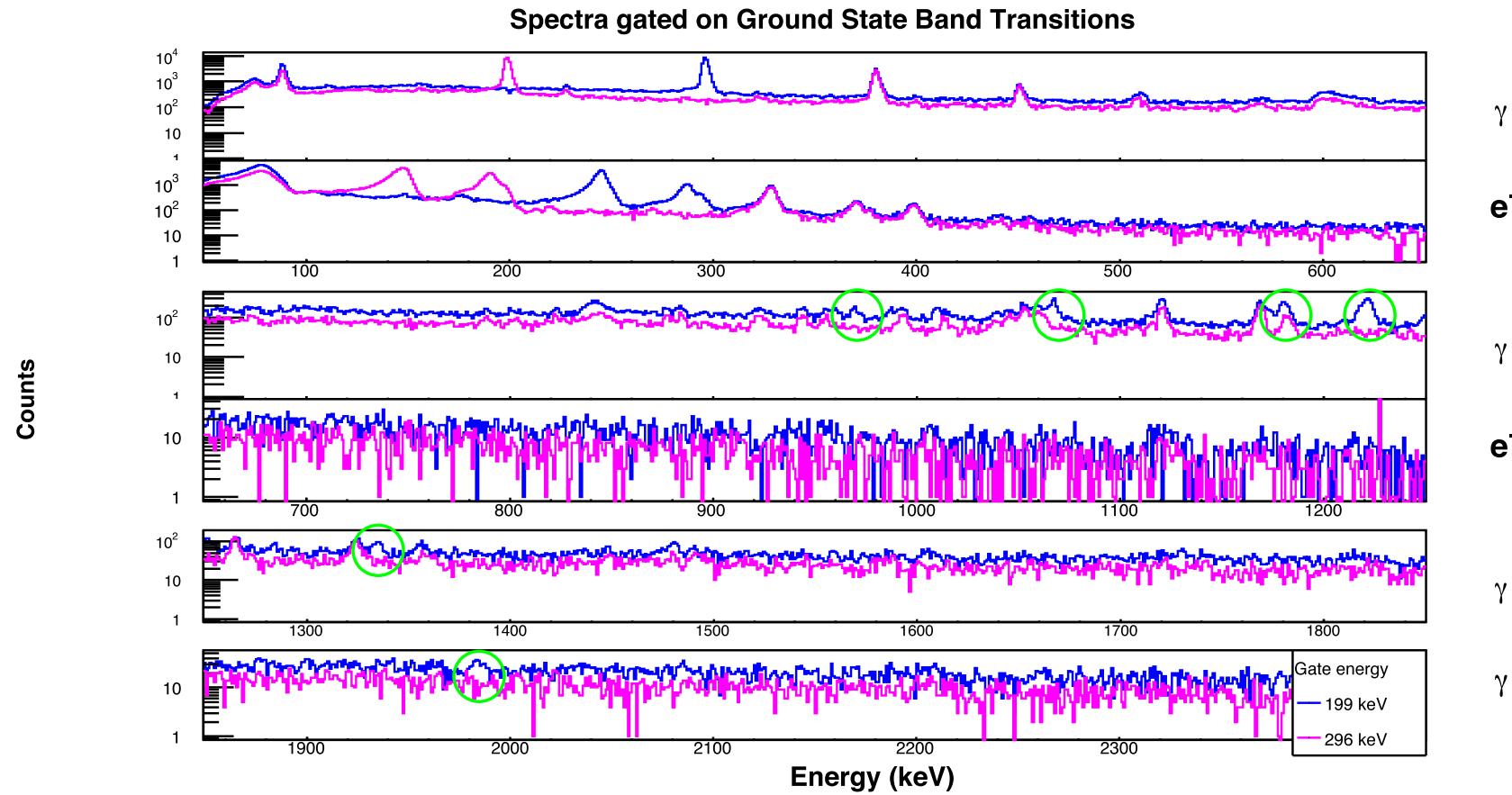


Figure 5.6. Spectra gated on two of the ground state band lines of ^{156}Gd . As highlighted in green, some lines do not appear in different gates. Comparison of these gates yields a list of transitions that directly populate the interim level.

In this data, the secondary gate check is especially important, as the attenuation of the $2_{gs}^+ \rightarrow 0_{gs}^+$ transition would mean weaker lines going into the 2_{gs}^+ state could be falsely assigned going into the 4_{gs}^+ state. As the ground state band transitions are so prolific, the transitions showed up when gating on weaker peaks in the spectra, while the inverse was not always true. The spectra from these gates are available in Appendix B.

Little could be seen in the $2_{gs}^+ \rightarrow 0_{gs}^+$ (89 keV) gate, as it reflected in Figure 5.2. The $4_{gs}^+ \rightarrow 2_{gs}^+$ (199 keV) gate shows far more, seen in Figure 5.3, including evidence of populating 3 excited 0^+ bands. In total, nine bands were seen in the gating. The number of bands seen drops off with higher gates, both due to the drop off in populating higher energy states and the drop off in populating higher J states. Studies have shown the $(\alpha, 2n)$ reaction to stop populating spin states with a significant cross section beyond about $J^\pi = 12^+$ [65]. In the ground state band of ^{156}Gd , the 12^+ state sits at 1924.49 keV, giving an approximate cut off for populating higher energy states in the nucleus. Additionally, due to the lack of known higher spin states in the higher energy 0^+ bands, identification of populating such bands in the higher gates becomes difficult.

5.3 Conversion Coefficients from Singles

With the gamma-rays now identified through gates, conversion coefficients could be calculated from singles spectra. Tables 5.3, 5.4, and 5.5 are these results. Where available, results were compared with Konijn et al[35]. Results are compared against this data, as it also examined the nucleus via an $(\alpha, 2n)$ reaction. Transitions that could be clearly identified and separated are found in Table 5.3. Contamination concerns were taken into account by using the identified transitions to look for gamma-rays in close proximity to those in the table. These transitions all have known multipole assignments and mixing ratios as needed, allowing for angular

corrections. Two transitions with possible $E0$ components are seen in this table. One, the $2^+ \rightarrow 2^+$ transition, appears to have a clear $E0$ component. Although this transition was not used in Figure 5.2, the band the state exists in is populated, as seen in Figures 5.3, 5.4, and 5.5. The $6^+ \rightarrow 6^+$ transition does not appear to have an $E0$ component. While the value agrees with Konijn[35], it does not agree with the theoretical value for an $E2$, the multipole the transition has been assigned. Table 5.4 holds conversion coefficients that have been left uncorrected for angular distribution in the singles for one of two reasons: either there were multiple known assignments to the gamma-ray energy, or the exact multipole mixing-ratio δ was unknown. Several of these transitions are high energy $J^\pi \rightarrow J^\pi$ transitions if the assignment is correct without extra contamination. This includes a $0^+ \rightarrow 0^+$ transition. The Si(Li) efficiency at these energies was too low to see the conversion electrons in gated spectra.

Table 5.5 has conversion coefficients that could not be corrected for angular distribution due to the transitions having unknown multipole assignments. Allowable and reasonable theoretical conversion coefficients from BrIcc[32] are listed in the table.

5.4 $J^\pi \rightarrow J^\pi$ Transitions

5.4.1 $0^+ \rightarrow 0^+$ Identification and Population

With the large number of 0^+ states in ^{156}Gd , it was necessary to identify which were populated. Table 5.2 contains a list of 0^+ states from NNDC, with notes as to which 0^+ states have been previously observed in the reaction used in this work. The states seen in this work are consistent with those seen in other $(\alpha, 2n)$ experiments and include higher states. The energies of the possible $0^+ \rightarrow 0^+$ transitions from these states are marked in the electron singles in Figure 5.7.

TABLE 5.2

 0^+ STATES IN ^{156}Gd

Energy (keV)	Seen in $(\alpha, 2n)$	Seen in This Work
0	×	×
1049.487(2)	×	×
1168.186(7)	×	×
1715.211(4)		×
1851.278(7)		×
1988.5(2)		
2082.0		

Table 4.6: A list of all 0^+ states in ^{156}Gd . 0^+ states above approximately 1924 keV would not be seen in this experiment, the energy of the 12^+ ground state, as that is the energy cut off in the nucleus for state population, discussed in the text. The levels seen in this exceed previous $(\alpha, 2n)$ experiments.

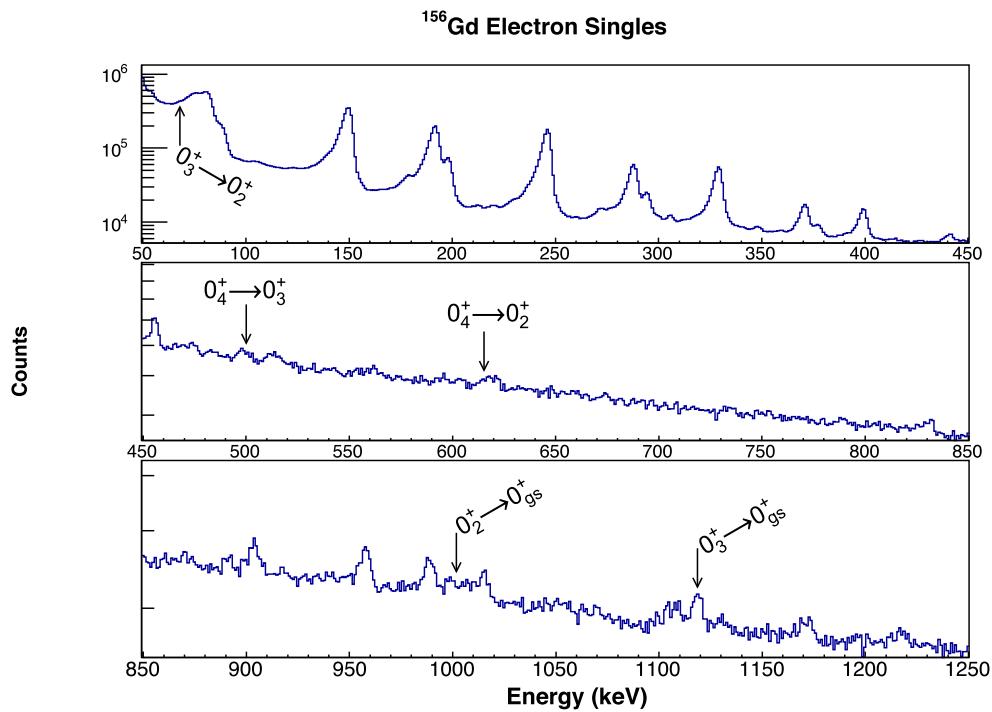


Figure 5.7. Electron singles for ^{156}Gd . The energies of transitions between 0^+ states are marked in the plot.

TABLE 5.3

 ^{156}GD INTERNAL CONVERSION COEFFICIENTS FROM SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	δ	Shell	α (This Work)	α (Th)	α (Konijn)
227.90	$7_{7-}^- \rightarrow 7_{4+}^+$	2137.6	1909.26	13000000000	E1		K	0.4704 (50) $^{+85}_{-84}$	0.0272 (4)	0.063 (13)
							LM	0.1077 (20) (20)	0.0049 (6)	
321.92	$8_{1-}^- \rightarrow 6_{1+}^-$	2027.1	1705.799		E2		K	0.0283 (13) (9)	0.0378 (6)	0.025 (7)
355.87	$4_{4+}^+ \rightarrow 2_\gamma^+$	1510.594	1154.152	189000	E2		K	0.0156 (6) (5)	0.0281 (4)	
399.56	$9_\gamma^+ \rightarrow 7_\gamma^+$	2249.65	1849.84		E2		K	0.0075 (8) (3)	0.0205 (3)	0.026 (5)
921.83	$5_\gamma^+ \rightarrow 6_{gs}^+$	1506.863	584.715	400	E2		K	0.0043 (9) (5)	0.0028 (1)	0.0030 (7)
1040.470	$2_2^+ \rightarrow 2_{gs}^+$	1129.437	88.970		E2+E0+M1	-5.9^{+14}_{-28}	K	0.0152 (10) (2)	0.0022 (1)	0.014 (3)
1059.31	$6_\gamma^+ \rightarrow 6_{gs}^+$	1643.653	584.715		E2		K	0.0011 (4) (1)	0.0021 (1)	0.0013 (8)

A list of conversion coefficients from ^{156}Gd . Multipolarities and mixing ratios were taken from the nuclear date sheets[52]. Unless otherwise stated, the α values are α_K . An angular distribution correction has been applied based on multipolarities for pure transitions, and those with known mixing ratios. The first error is statistical, the second is systematic. Numbers are compared with Konijn et al[35]. Whole numbers are 0^+ bands.

TABLE 5.4

 UNCORRECTED ^{156}GD INTERNAL CONVERSION COEFFICIENTS FROM
 SINGLES

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	δ	α (This Work)	α (Th)	α (Konijn)
154.94	$4_{4+}^+ \rightarrow 4_\gamma^+$	1510.594	1355.422	189000	M1+E2	0.48	0.4635 (183) $^{+98}_{-97}$	0.460 (7)	
	$7_{4+}^+ \rightarrow 6^+$	1909.26	1753.653		(M1+E2)	0.29		0.474 (7)	
883.86	$8_2^+ \rightarrow 8_{gs}^+$	1848.33	965.134		E0+E2		0.0057 (7) (1)	0.0030 (1)	> 0.0092
	$7_\gamma^+ \rightarrow 8_{gs}^+$	1849.84	965.134		E2(+M1)			0.0030 (1)	< 0.0052
955	$6_2^+ \rightarrow 6_{gs}^+$	1540.19	584.715		E0+E2		0.0065 (4) (5)	0.0026 (1)	0.020 (8)
959.88	$0_2^+ \rightarrow 2_{gs}^+$	1049.487	88.97	1800	E2			0.0025 (1)	0.0045 (24)
	$3_\gamma^+ \rightarrow 4_{gs}^+$	1248.006	288.197		E2+M1	-12		0.0025 (1)	
1009.33	$4_2^+ \rightarrow 4_{gs}^+$	1297.822	288.197	1600	E0+E2,M1		0.0173 (9) (4)		0.0164 (29)
1045.48	$8_\gamma^+ \rightarrow 8_{gs}^+$	2011.38	965.134		E2(+M1)		0.0012 (2) (2)	0.0021 (1)	0.0025 (6)
1052.61	$7_{1-}^- \rightarrow 6_{gs}^+$	1638	584.715		E1			0.0009 (1)	
1065.74	$2_\gamma^+ \rightarrow 2_{gs}^+$	1154.152	88.97	568	E2+M1	-16	0.0023 (2) (1)	0.0021 (1)	0.0025 (9)
	$4_\gamma^+ \rightarrow 4^+$	1355.422	288.187		E2+M1	-4		0.0021 (3)	0.0021 (51)
1158.65	$2_\gamma^+ \rightarrow 0_{gs}^+$	1154.152	0	568	E2		0.0020 (3) (1)	0.0017 (1)	0.0023 (3)

TABLE 5.4 (CONTINUED)

E (keV)	$J^\pi \rightarrow J^\pi$	E_i (keV)	E_f (keV)	$T_{1/2}$ (fs)	Multipolarity	δ	α (This Work)	α (Th)	α (Konijn)
	$3_\gamma^+ \rightarrow 2_{gs}^+$	1248.006	88.97	580	E2+M1	-11.8		0.0017 (1)	
1168.69	$0_3^+ \rightarrow 0_{gs}^+$	1168.186	0	5000	E0		0.0045 (3) (1)		> 0.0035
	$2_3^+ \rightarrow 2_{gs}^+$	1258.075	88.97	1540	E2+M1+E0	0.38		0.0026 (1)	
1222.22	$4_{4+}^+ \rightarrow 4_{gs}^+$	1510.594	288.197	189000	M1+E2	-1.7	0.0028 (4) (1)	0.0018 (1)	0.00174*
	$5_\gamma^+ \rightarrow 4_{gs}^+$	1506.863	288.197	400	E2			0.001560 (22)	
1264.85	$4_\gamma^+ \rightarrow 2_{gs}^+$	1355.422	88.97	540	E2		0.0017 (3) (1)	0.0014 (1)	
	$8_2^+ \rightarrow 6_{gs}^+$	1848.33	584.715						
	$7_\gamma^+ \rightarrow 6_{gs}^+$	1849.84	584.715						

A list of conversion coefficients from ^{156}Gd . Multipolarities and mixing ratios were taken from the nuclear data sheets[52]. Unless otherwise stated, the α values are α_K . No angular distribution correction has been applied, either due to unknown mixing ratios, or multiple assignments of the gamma-ray. The first error is statistical, the second is systematic. Numbers are compared with Konijn et al. [35] Starred values were used as calibration points in the Konijn paper. All coefficients are K-shell electrons. Whole numbers in the subscript indicate 0^+ bands.

TABLE 5.5

$$^{156}\text{GD INTERNAL CONVERSION ELECTRONS WITHOUT KNOWN}$$

$$\text{MULTIPOLEARITIES}$$

E (keV)	$J^\pi \rightarrow J^\pi$	E_f (keV)	E_i (keV)	α (This Work)	Theory			
					$\alpha(\text{M1})$	$\alpha(\text{E2})$	$\alpha(\text{E1})$	α (Konijn)
671.41	$7_{1^-}^- \rightarrow 8_{gs}^+$	1638	965.134	0.0064 (6) (2)			0.00213 (3)	
838.83	$9_\gamma^+ \rightarrow 10_{gs}^+$	2249.65	1416.078	0.0009 (3) (1)	0.00595 (9)	0.00337 (5)		
943.732	$7_{4^+}^+ \rightarrow 8_{gs}^+$	1909.26	965.134	0.0022 (6) (1)	0.00448 (7)	0.00262 (4)		0.0025 (3)

A list of conversion coefficients from ^{156}Gd without known multipolarities. As a result, an angular distribution correction term has not been applied. None of the states have known half lives. The first error is statistical, the second is systematic. Numbers are compared with theoretical values for allowed multipolarities and results from Konijn et al. [35]. All coefficients are K-shell electrons.

To obtain information in these areas, and for possible $J^\pi \rightarrow J^\pi$ transitions, energy gates were used. A full list of gates and spectra can be found in Appendix B.

5.4.2 $J^\pi \rightarrow J^\pi$ Transitions

After identification of the bands and transitions, gates were put on the transitions both entering and leaving states of interest, namely even- J^+ states. Gates entering these states (incoming transitions) were not fruitful. This is unfortunate for the 0^+ states, as only incoming transitions can be used for transitions to the ground state. Gates leaving the states (outgoing transitions), had more statistics, and results could be obtained. To look for these transitions of interest, other transitions, namely those of the ground state band, were subtracted from the regions of interest as described in section 3.4. This subtraction technique causes the results to be treated as limits. $J^\pi \rightarrow J^\pi$ transitions up to 4^+ were seen. Tables 5.6, 5.7, and 5.8 are the tabulated results. These values are compared with Konijn[35] where available. Additionally, the theoretical conversion coefficients have been listed for the $M1$ and $E2$ transitions, as taken from BrIcc[32].

Due to the lack of lifetimes of these states, $B(E0)$ values cannot be calculated. However, the relative intensities of these values can be compared, assuming they are coming from the same state, as the lifetime would divide out (see equations 1.41 and 1.42). The contributions from the individual components of the transition must be separated out for the 2^+ and 4^+ transitions. This was done by getting the q^2 values multiplied by the theoretical $\alpha(E2)$, which gives an estimate of $E0$ intensity (see section 1.3.1). None of the transitions have known δ mixing ratios, so δ was assumed to be 1, and the theoretical mixed $M1$ and $E2$ α was subtracted. In some cases, this left a negative value, which has been excluded from the table of results, Table 5.9. All values calculated are upper or lower limits, as the α calculated in the previous tables were upper and lower limits.

TABLE 5.6

 $0^+ \rightarrow 0^+$ TRANSITIONS IN ^{156}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory		
							$\alpha(M1)$	$\alpha(E2)$	α (Konijn)
1168.186	3	1049.487	2	118.71	960.50771	> 1.1970	1.042 (15)	0.726 (11)	

A list of conversion coefficients from ^{156}Gd for $0^+ \rightarrow 0^+$ transitions seen in the gated data. All listed theoretical values are for the K-shell internal conversion coefficient. Numbers are compared with theoretical values for allowed multipolarities and results from Konijn et al.[35] All coefficients are K-shell electrons. Whole numbers for the bands indicate excited 0^+ bands.

TABLE 5.7

 $2^+ \rightarrow 2^+$ TRANSITIONS IN ^{156}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory		
							$\alpha(\text{M1})$	$\alpha(\text{E2})$	α (Konijn)
1258.075	3	1129.437	2	128.638	1040.470	> 0.5325	0.830 (12)	0.578 (8)	
1258.075	3	1154.152	γ	103.92	1065.1781	> 2.9695	1.524 (22)	1.049 (15)	
1827.841	2^+	1129.437	2	698.407	1040.470	< 0.0208	0.00932 (13)	0.00506 (7)	
1827.841	2^+	1154.152	γ	673.684	1065.1781	< 0.0228	0.01018 (15)	0.00550 (8)	
1827.841	2^+	1258.075	3	569.771	1169.087	< 0.0013	0.01545 (22)	0.00819 (12)	0.006 (4)

A list of conversion coefficients from ^{156}Gd for $2^+ \rightarrow 2^+$ transitions seen in the gated data. All listed theoretical values are for the K-shell internal conversion coefficient. Numbers are compared with Konijn et al.[35] All coefficients are K-shell electrons. Whole numbers for bands indicate 0^+ excited bands.

TABLE 5.8

 $4^+ \rightarrow 4^+$ TRANSITIONS IN ^{156}GD

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	α (This Work)	Theory		
							α (M1)	α (E2)	α (Konijn)
1462.297	3	1297.822	2	164.469	1009.649	> 0.4870	0.416 (6)	0.279 (4)	
1462.297	3	1355.422	γ	106.88	1067.2325	> 0.7233	1.405 (20)	0.972 (14)	
1510.594	4^+	1297.822	2	212.771	1009.649	> 0.0704	0.204 (3)	0.1282 (18)	
1510.594	4^+	1355.422	γ	155.168	1067.2325	> 0.0981	0.490 (7)	0.333 (5)	

A list of conversion coefficients from ^{156}Gd for $4^+ \rightarrow 4^+$ transitions seen in the gated data. All listed theoretical values are for the K-shell internal conversion coefficient. Numbers are compared with Konijn et al. [35] All coefficients are K-shell electrons. Whole numbers for bands refer to excited 0^+ bands.

With these values, two transitions from the same level can be compared using the $B(E0)$ formula to take the energy adjustment into account. It is also adjusted by the ratio of the rate efficiencies. Only one 2^+ level had transitions to other 2^+ states that could be compared. This ratio is in Table 5.10.

The first excited 0^+ band and the γ band are not discussed, as no $J^\pi \rightarrow J^\pi$ transitions could be observed leaving these levels.

5.4.2.1 $K^\pi = 0_3^+$, Second excited 0^+ band, 1168.186 keV

Transitions from this band to both the first 0^+ band and the γ band could be observed. All values observed are lower limits. The 0^+ state in the band has a known lifetime, allowing for the $0^+ \rightarrow 0^+$ nuclear matrix element to the first excited 0^+ state be calculated[3]. Table 5.11 contains the calculated q^2 and ρ^2 value for the $0_3^+ \rightarrow 0_2^+$ transition.

For the other transitions to the first excited 0^+ band, the $2^+ \rightarrow 2^+$ transition lower limit did not allow for any conclusions to be drawn as to the nature of the transition. The $4^+ \rightarrow 4^+$ transition lower limit indicates an E0 component.

Assuming both transitions have an E0 component and δ is 1, Table 5.9 has the lower limits tabulated for the $q^2\alpha(E2)$ values.

For the transitions to the γ band, the $2^+ \rightarrow 2^+$ transition lower limit indicates an E0 component. The $4^+ \rightarrow 4^+$ transition lower limit did not allow for any conclusions to be drawn as to the nature of the transition.

5.4.2.2 $K^\pi = 2^+$, Second excited 2^+ band, 1827.841 keV

The transitions from the head of this band to the 2^+ states of the γ band and the first and second 0^+ excited states were observed. All three coefficients are upper limits. The transitions to the γ band and the first excited 0^+ band may have E0 components, as the upper limits do not rule them out. Assuming both transitions

TABLE 5.9

 $E0$ CONTRIBUTIONS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS

E_i (keV)	Band	E_f (keV)	Band	E (keV)	Gate	$q^2\alpha(E2)$
$2^+ \rightarrow 2^+$						
1258.075	3	1154.152	γ	103.92	1065.1781	> 3.366
1827.841	2^+	1129.437	2	698.407	1040.47	< 0.02722
1827.841	2^+	1154.152	γ	673.684	1065.1781	< 0.02992
$4^+ \rightarrow 4^+$						
1462.297	3	1297.822	2	164.469	1009.649	> 0.279

A list of $E0$ contributions in ^{156}Gd , assuming a mixing ratio between the E2 and M1 contributions of $\delta = 1$. These values have not been normalized, as the lifetime of the states are unknown. Table 5.10 compares values between two transitions of the same initial state. Only non-negative values are listed in the table, and δ was assumed to be 1, as no mixing ratios are known for these transitions. For $\alpha(\text{exp})$, $\alpha(M1)$, and $\alpha(E2)$ used in these calculations, please refer to Tables 5.7-5.8. Whole numbers for bands refer to excited 0^+ bands.

TABLE 5.10

 $B(E0)$ RATIOS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS

E_i (keV)	Band	$E_{0_2^+}$ (keV)	Gate $_{0_2^+}$	E_γ (keV)	Gate $_\gamma$	$B(E0)$ Ratio
$2^+ \rightarrow 2^+$						
1827.841	2^+	1129.437	1040.47	1154.152	1065.1781	1.128

Ratios of the $B(E0)$ values in ^{156}Gd . Only ratios between two transitions of the same state are listed, as the lifetime of the states are unknown. Table 5.9 lists the values that were used in the calculation. The gates are included, as an efficiency correction was made on the ratio based on the gates. In many cases, only upper or lower limits for the values could be used for this calculation. Errors are not given on these values. Those values marked with errors or as limits had defined values instead of limits.

E0 CONTRIBUTIONS FOR $J^\pi \rightarrow J^\pi$ TRANSITIONS

TABLE 5.11

E_i (keV)	Transition	$E0$ (keV)	Transition	$E2$ (keV)	$t_{1/2}$ (ps)	$q_K^2(E0/E2)$	$\rho^2(E0)$
1168.186	$0_3^+ \rightarrow 0_2^+$	118.71	$0_3^+ \rightarrow 2_{gs}^+$	1079.216	$0.9031 < t_{1/2} < 4.4221$	1.995 (15)	$22.78 < \rho^2 < 111.53$

Table 5.11: A list of $q_K^2(E0/E2)$ and $\rho^2(E0)$ contributions in ^{156}Gd for the $0^+ \rightarrow 0^+$ transitions. Lifetime from [3].

have an E0 component and δ is 1, Table 5.9 has the upper limits tabulated for the $q^2\alpha(E2)$ values. The values are quite small. Table 5.10 compares the strengths of the two transitions as a ratio. The ratio is close to 1, indicating similar transition strengths from this 2^+ excited band to the γ band the first excited 0^+ band. The upper limit on the transition to the second excited 0^+ band does rule out an E0 component. The upper limit is lower than the measurement by Konijn, although not significantly so when looking at the error on the conversion coefficient[35].

5.4.2.3 $K^\pi = 4^+$, First excited 4^+ band, 1510.594 keV

The transitions from the head of this band to the 4^+ states of the γ band and the first excited state were observed. Both coefficients are lower limits. No conclusions can be drawn for the transitions, as the lower limits do not rule any type of transition out.

CHAPTER 6

DISCUSSION

6.1 Dynamic Moments of Inertia

The dynamic moment of inertia of a band of excitations in a nucleus gives direct insight into the shape of a nucleus. Within the nucleus, two bands having the same dynamic moment of inertia can indicate a relationship between the bands, such as a two-phonon excitation built upon a one-phonon band.

Combining the dynamic moment of inertia with measures of the ρ^2 of E0 transitions can give further insight into the nuclear matrix element. While the E0 matrix element is generally seen as a change in the radius of the nucleus between two states of the same J^π , it is, more accurately, a combination of this radial change and a measure of the overlap of the nuclear states. Comparing the dynamic moments of inertia of the two bands involved in an E0 transition can aid in examining the strength of the nuclear overlap.

6.2 Bands in ^{154}Gd

To examine the structure of the bands seen in this experiment, the dynamic moments of inertia can be compared. This is usually done with higher spins, but examination at lower spins can allow for more bands to be compared. The energy difference tracks linearly, with the slope being directly correlated to the moment of inertia of the band, as seen in Figure 6.1 and Figure 6.2. The slopes of these bands are summarized in Table 6.1. Slopes with an error had enough points to calculate

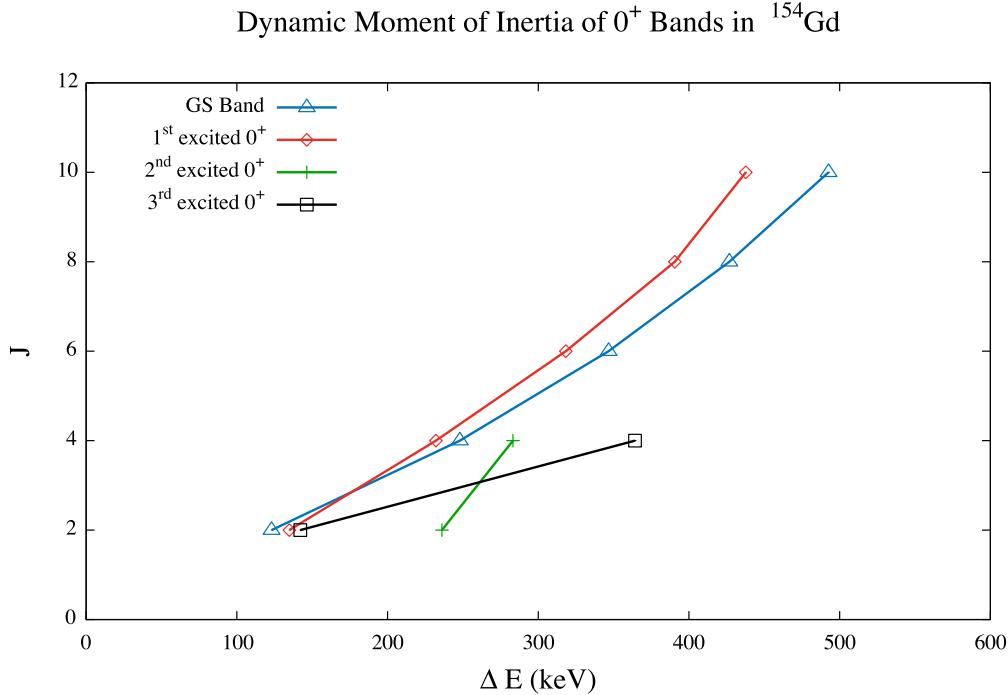


Figure 6.1. The dynamic moments of inertia of the four 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the first excited 0^+ band have very similar moments of inertia.

the error on the slope. Those without error had only two points or, in the case of the second excited 2^+ band, one point and the origin. These points each require two levels of the band to be known for calculation. The intercept was left to float, but not included, as it was in agreement with 0 in all cases where error could be calculated.

Figure 6.1 shows the 0^+ bands. The two bands agree within 2σ . While the γ -band also appears similar in slope to the ground state band in Figure 6.2, they do not overlap within 3σ . The first 4^+ excited band does have a slope within 2σ of the γ -band.

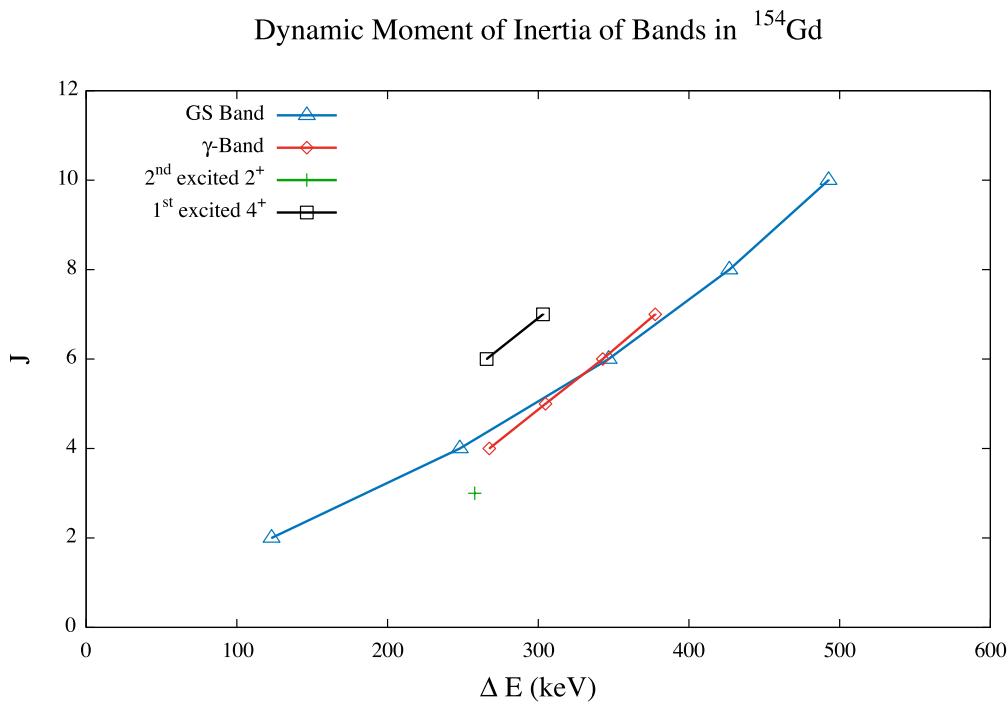


Figure 6.2. The dynamic moments of inertia of the non- 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the γ band have similar moments of inertia. However, they do not overlap within their standard deviations.

TABLE 6.1
DYNAMIC MOMENTS OF INERTIA OF BANDS SEEN IN ^{154}Gd

Band	Moment of Inertia
Ground State	0.0214 (15)
1st excited 0^+	0.0258 (19)
2nd excited 0^+	0.0424
5th excited 0^+	0.0090
γ -band	0.0271 (3)
2nd excited 2^+	0.0155
1st excited 4^+	0.0267

Table 6.1: List of the moments of inertia of the bands seen in ^{154}Gd in this experiment. The moment of inertia is the slope in a least-squares linear regression. Those without error only had one or two points of energy difference, so the standard deviation of the slope could not be calculated. The ground state band and the first excited 0^+ band agree within two standard deviations.

6.2.1 $K^\pi = 0_2^+$, First excited 0^+ band, 680.6673 keV

The first excited 0^+ band is the only band with lifetimes and mixing ratios such that ρ^2 could be calculated, seen in Table 4.12. The first excited 0^+ band has a very similar slope to the ground state band, as seen both in Figure 6.1 and Table 6.1, indicating similar moments of inertia. With the large ρ^2 values calculated for the $2^+ \rightarrow 2^+$ and $4^+ \rightarrow 4^+$ transitions to the ground state, the small change in the moment of inertia would indicate a large overlap in the nuclear states.

6.2.2 $K^\pi = 0_3^+$, Second excited 0^+ band, 1182.091 keV

The second excited 0^+ band has a much larger dynamic moment of inertia, compared to the ground state, first excited 0^+ state and γ -bands. With the data collected in this work, comparisons can be made between the latter two, via Table 4.14, where the strength of the ratio of transitions heavily favor the γ -band.

6.2.3 $K^\pi = 0_6^+$, Fifth excited 0^+ band, 1573.9 keV

The fifth excited 0^+ band has the smallest slope of all the bands looked at in this nucleus. It has a stronger connection to the second excited 0^+ band than the first.

6.2.4 $K^\pi = 2^+$, γ band, 996.264 keV

The γ band has a very similar slope to the ground state band, as seen both in Figure 6.1 and Table 6.1, indicating similar moments of inertia, although the slope is higher than that seen by the ground state band or first excited 0^+ band. While measures of the transitions between the γ band and ground state band could not be done in this work, measures between the γ -band and first excited 0^+ were done.

The $2^+ \rightarrow 2^+$ and $6^+ \rightarrow 6^+$ transitions indicate potentially large E0 components for the conversion coefficients. The $4^+ \rightarrow 4^+$ transition indicates small E0 components. Of these transitions, the $2^+ \rightarrow 2^+$ has a previously measured relative γ -intensity of

0.043 (6)% [51]. The low intensities of these transitions and the lack of mixing and branching ratios make it difficult to draw conclusions.

6.2.5 $K^\pi = 2^+$, Second excited 2^+ band, 1531.305 keV

The dynamic moment of inertia of the 2nd excited 2^+ state is very different from any of the previously discussed bands. This band has indications of E0 components to both the γ -band and the first excited 0^+ band for the $2^+ \rightarrow 2^+$ transition, but only indications to the first excited 0^+ band for the $4^+ \rightarrow 4^+$ transition.

Comparisons of the transitions between the two bands in Table 4.14 indicate a stronger connection to the first excited 0^+ band than the γ -band. This does not confirm it to be built on the first excited 0^+ band, though evidence points to this, including the branching ratios to the ground state and the first excited 0^+ band.

6.2.6 $K^\pi = 4^+$, First excited 4^+ band, 1645.814 keV

The dynamic moment of inertia for the first excited 4^+ band is almost identical to the γ -band dynamic moment of inertia. The ratios in 4.14 indicate a stronger connection to the γ band. The E0 components would not be large for these transitions, indicating a potentially small ρ^2 . The large connection but small ρ^2 would make sense if the first excited 4^+ band is the 4^+ component of the $\gamma\gamma$ excitations.

6.3 Bands in ^{154}Gd

To examine the structure of the band seen in this experiment, the dynamic moments of inertia can be compared. This is usually done at higher spins, but examination at lower spins can allow for more bands to be compared. The energy difference tracks linearly, with the slope being directly correlated to the moment of inertia of the band, as seen in Figures 6.3 and 6.4. The slopes of these bands are summarized

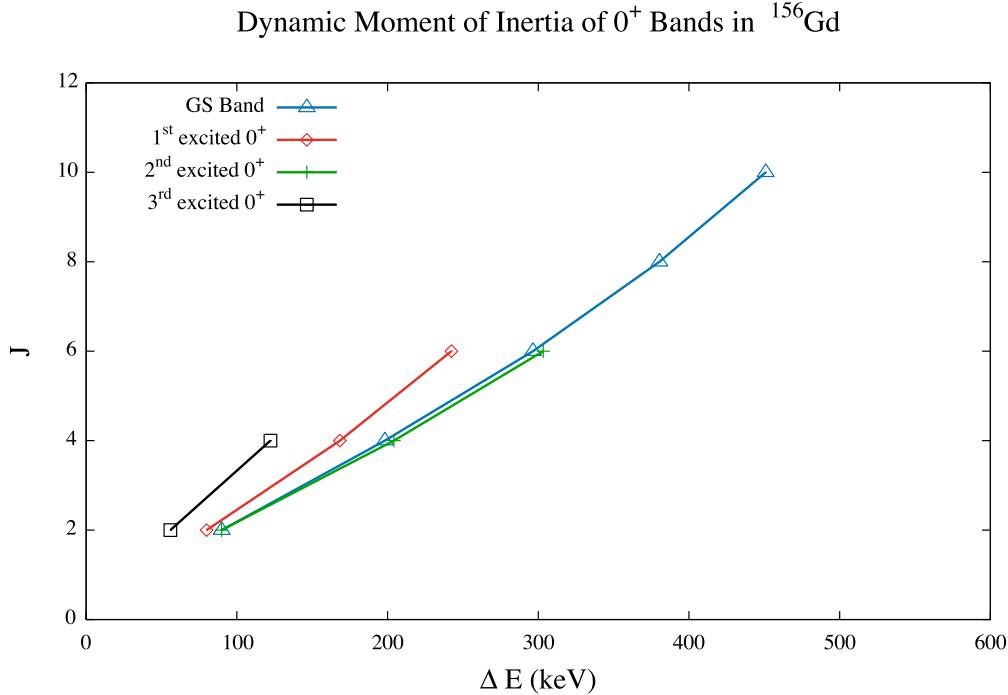


Figure 6.3. The dynamic moments of inertia of the four 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the first excited 0^+ band have very similar moments of inertia.

in Table 6.2. Slopes with an error had enough points to calculate the error on the slope. Those without error had only two points or, in the case of the second excited 2^+ band, one point and the origin. These points each require two levels of the band to be known for calculation. The intercept was left to float, but not included, as it was in agreement with 0 in all cases where error could be calculated.

Figure 6.3 shows the moments of inertia of the 0^+ bands, including the ground state band. The first excited 0^+ band has a very similar slope to the ground state band, as seen both in Figure 6.3 and Table 6.2. The two bands agree within 2σ . The γ -band and the excited 4^+ band also appear similar in slope to the ground state band in Figure 6.4, and overlap within 2σ . The first 4^+ excited band is almost identical in slope to the first excited 0^+ band. Both of these bands are about 1σ

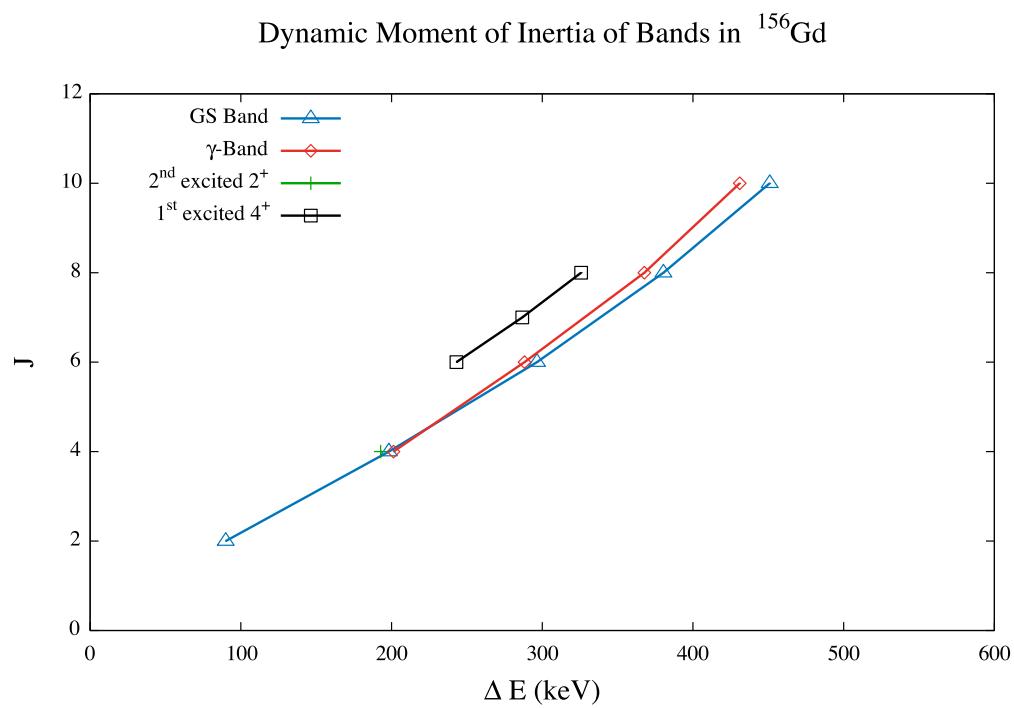


Figure 6.4. The dynamic moments of inertia of the non- 0^+ bands seen in the experiment. As is seen visually and with the slopes, the ground state band and the γ band have similar moments of inertia, and overlap within two standard deviations.

TABLE 6.2
DYNAMIC MOMENTS OF INERTIA OF BANDS SEEN IN ^{156}Gd

Band	Moment of Inertia
Ground State	0.0220 (11)
1st excited 0^+	0.0245 (13)
2nd excited 0^+	0.0187 (8)
3rd excited 0^+	0.0301
γ -band	0.0259 (13)
2nd excited 2^+	0.0208
1st excited 4^+	0.0242 (10)

Table 6.2: List of the moments of inertia of the bands seen in ^{156}Gd in this experiment. The moment of inertia is the slope in a least-squares linear regression. Those without error only had one or two points of energy difference, so the standard deviation of the slope could not be calculated. The ground state band and the first excited 0^+ band agree within two standard deviations. The same is true of the ground state band and both the γ -band and first excited 4^+ band.

apart.

6.3.1 $K^\pi = 0_3^+$, First excited 0^+ band, 1049.487 keV

The first excited 0^+ band has a very similar slope to the ground state band, as seen both in Figure 6.3 and Table 6.2. The two bands agree within 2σ . Unfortunately, without more statistics, data could not be collected for outgoing transitions from the first excited 0^+ band.

6.3.2 $K^\pi = 0_3^+$, Second excited 0^+ band, 1168.186 keV

The second excited 0^+ band could be seen going to both the γ band and first excited 0^+ band. It has a very different dynamic moment of inertia compared to those two. This change in shape may account for the large ρ^2 value between the first and second 0^+ excited states. The transitions for the 2^+ and 4^+ states in this band could not be determined to have E0 components or not to the first excited 0^+ state, and the 2^+ state may have an E0 component to the γ -band.

6.3.3 $K^\pi = 2^+$, γ band, 1065.1781 keV

The γ band has a very similar slope to the ground state band, as seen both in Figure ?? and Table 6.2, indicating similar moments of inertia, although the slope is higher than that seen by the ground state band or first excited 0^+ band. The transitions between the first excited 0^+ band and the γ band are not energetically capable of conversion electrons for the $2^+ \rightarrow 2^+$ transition, and below the energy threshold of the detectors in this work for the $4^+ \rightarrow 4^+$ transition.

6.3.4 $K^\pi = 2^+$, Second excited 2^+ band, 1827.841 keV

The second excited 2^+ band has a smaller slope than the ground state band, first excited 0^+ band and the γ band. It is a slightly larger moment of inertia than the

second excited 0^+ band. Transitions to this band do not indicate an E0 component. Transitions to the first excited 0^+ band and the γ band may have an E0 component, but with only upper limits, it is difficult to determine. These limits indicate approximately equal strengths to both bands, as seen in Table 5.10.

6.3.5 $K^\pi = 4^+$, First excited 4^+ band, 1510.594 keV

Only the first state in the first excited 4^+ band was able to get electron spectroscopy. This band has a similar slope to the ground state band, first excited 0^+ band and the γ band, although it is least like the ground state band dynamic moment of inertia. The lower limits on the conversion coefficients to the first excited 0^+ band and the γ band tell very little, as all forms of multipolarity, including a pure E2, are still accessible. Further, the lack of information on transition to the ground state band obfuscates interpretation.

CHAPTER 7

FUTURE WORK

Throughout the course of this experiment, there have been a variety of areas that were adjusted to try and optimize the data. In this chapter, continued work and optimizations for future experiments are discussed.

7.1 Detector Upgrades

In a given experiment, the detectors are one of the single most important pieces of equipment. Two different styles of HPGe detectors were used, to varying success, and different magnetic configurations were used to optimize efficiency, discussed in Section 2.4.1.1. Further steps are being taken for the future of conversion electron experiments at Notre Dame in both areas to improve the quality of data.

7.1.1 fIREBAll

The La Crosse Internal ConveRsion Electron Ball Array (fIREBAll) will be the next generation of ICEBall[37]. The Major Research Instrumentation (MRI) proposal to the National Science Foundation (NSF) for funding has been approved. The University of Wisconsin-La Crosse submitted the proposal. The ICEBall beamline at the NSL will become the fIREBAll beamline. The focus of the proposal was detector upgrades, both for the Si(Li) detectors, and the HPGe detectors. With this proposal, there will be dedicated HPGe detectors for the array, with two Bismuth Germanate (BGO) anti-Compton suppression shields. The current Si(Li)

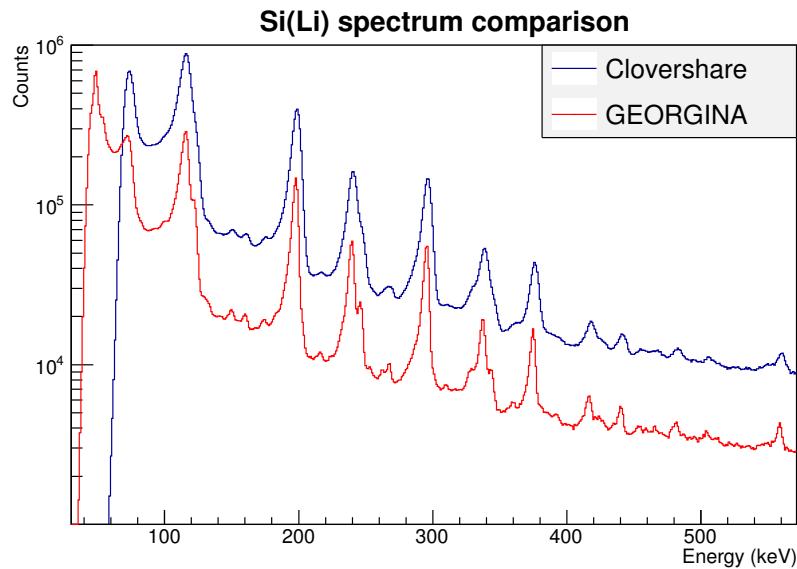


Figure 7.1. A comparison of the singles spectrum of one Si(Li) detector between the GEORGINA and Clovershare experiments. The resolution of the Si(Li) detector has deteriorated between the two experiments (the GEORGINA one was performed first). The LM peaks at ~ 240 keV and ~ 330 keV show this most clearly. The peaks are distinguishable in the GEORGINA data, but not in the Clovershare data.

detectors would be replaced with two Si(Li) detectors, for a total of twelve detectors each. There would still be six detection planes, as the Si(Li) detectors would be stacked in a telescope formation, to allow for the measurement of higher energy conversion electron transitions, including E0 transitions. The current Si(Li) detectors in use are 5 mm thick. All of the new detectors would be 5 mm thick, making for an active stopping depth of 10 mm when stacked. Additionally, the detectors would be 800 mm^2 surface area instead of 750 mm^2 of the current detectors. Over the past 25 years of use, the current Si(Li) detectors have begun to deteriorate. As seen in Figure 7.1, there is a clear decline in the resolution of the detector between the different experiments.

7.1.2 Magnet Designs

To improve the detection efficiency of the conversion electrons, the magnetic configurations can be changed. This was done in the experiments to optimize for higher energies. When the experiments in this work were performed, the permanent magnets used were $36 \times 36 \times 5$ mm SmCo₅ squares. These thin squares are not, necessarily, creating the most optimal magnetic field.

Using GEANT4 and COMSOL, a variety of magnetic shapes were modeled, varying the cross-shape profile, the thickness, and the number of magnets used [1, 21].

Figure 7.2 shows what the magnetic fields look like when modelled in COMSOL.

This figure is of the original magnets used in this work. Efficiencies of the new shapes were modeled from 0.1 to 1.0 MeV. The magnets were assumed to have a maximal side length of 37 mm. All of the magnets simulated were assumed to be Neodymium, specifically Nd₂Fe₁₄B. In all of the shape testing, the number of magnets was varied to find idealized scenarios, with the magnets symmetrically spaced. The first variation was the cross-shape profile. Two extremes would be chosen, with intermediate steps between the extremes modelled and tested, as shown in Figure 7.3. Several hybrids of the intermediate steps were also examined, seen in Figure 7.4. The number of magnets was varied from 3 to 6 for each design, and evenly spaced.

After picking the best cross-shapes, the thickness was varied between uniform and a gradient. Initially, this was a linear, uniform gradient. As simulations continued, variations in the form of the gradient were tested, including a step function, and a piecewise linear design, seen in Figure 7.5. As with the cross-shape, the number of magnets was varied for each design, ranging from 3 to 6 magnets.

Three designs were chosen after extensive simulations, to maximize efficiencies at low energies (0.2 to 0.4 MeV), high energies (0.7-1.0 MeV), and overall (0.2 to 1.0 MeV). These designs are shown in Figure 7.6 with simulated efficiencies for six

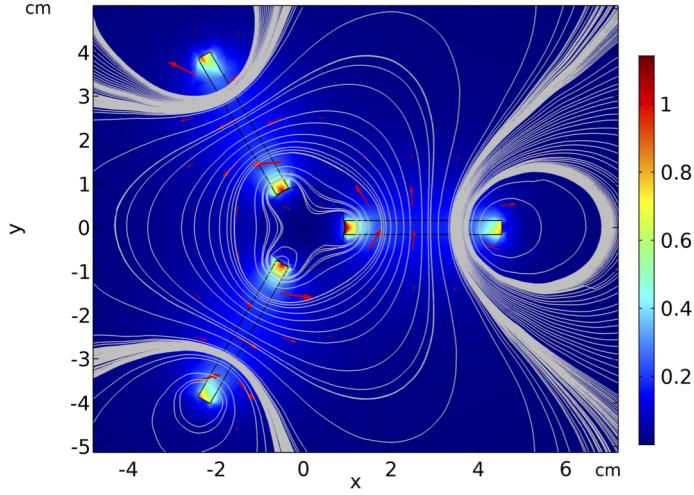


Figure 7.2. Map of the magnetic field lines, created through the use of COMSOL, for the original magnetic design [21].

magnet configurations. These designs have been ordered for testing with the detectors.

7.2 Target Upgrades

One of the concerns and limitations of the original experiments was the thickness of the target. Ideally, the target being used would be less than $500 \mu\text{g}/\text{cm}^2$, but the ones used were $1.7 \text{ mg}/\text{cm}^2$ and $1.44 \text{ mg}/\text{cm}^2$. While these targets were self-supporting, they were thick enough that electron straggling occurred, smearing out the spectrum.

To create a thinner target, it was decided to explore a carbon-backing. The Sm was evaporated onto a backing $20 \mu\text{g}/\text{cm}^2$ thick. In the targets made, the Sm was measured to be $170 \mu\text{g}/\text{cm}^2$, ten times thinner than the original targets used.

To compare the old and new targets, an experiment was run with both targets inside of ICEBall. The data compared was taken with the same electronics (discussed in the next section) and setup, to minimize differences that could impact the data.

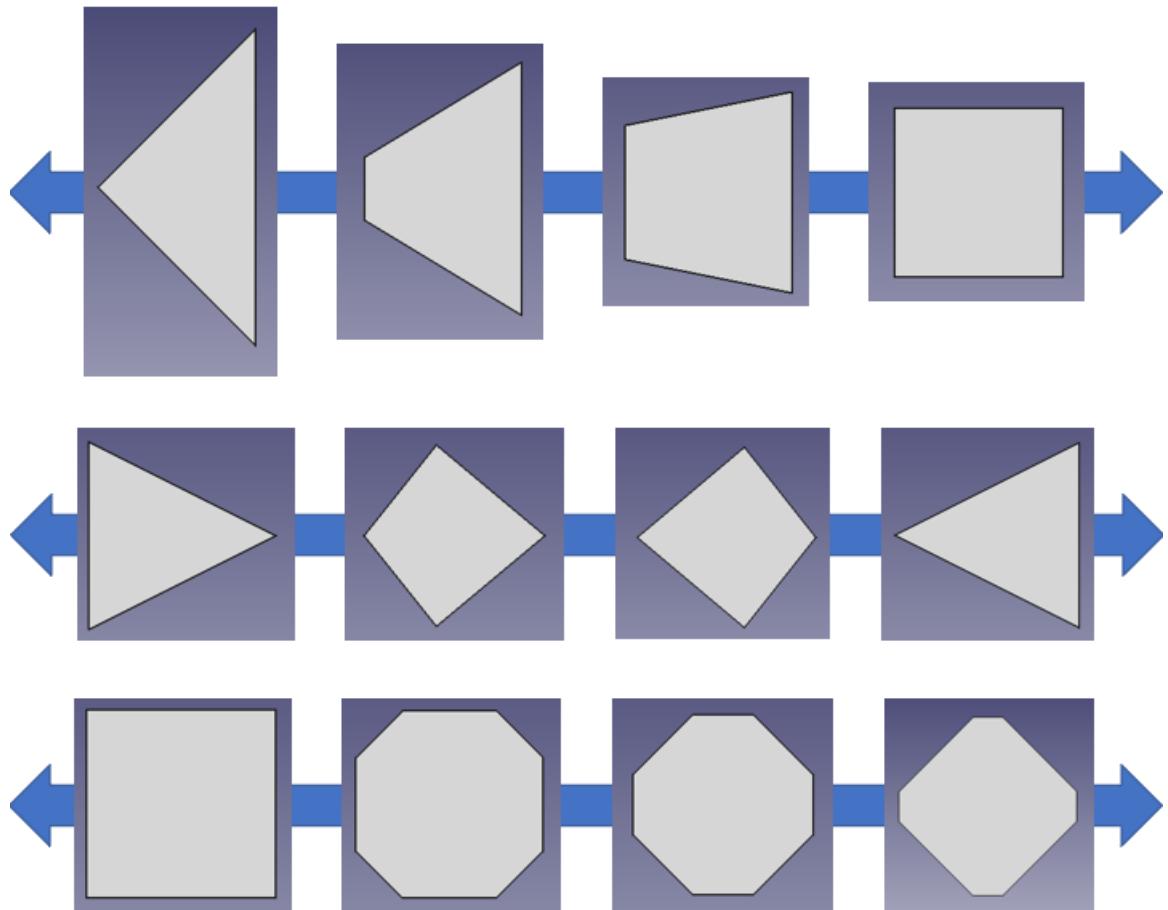


Figure 7.3. Initial tests of the cross shapes. The left-side of the shape would be the side against the blocker. Two extremes were chosen, with intermediate steps being tested between the two. These are the three sets of extremes used: large triangle to square, triangle with the base against the blocker to triangle with the point against the blocker, and square to truncated diamond.

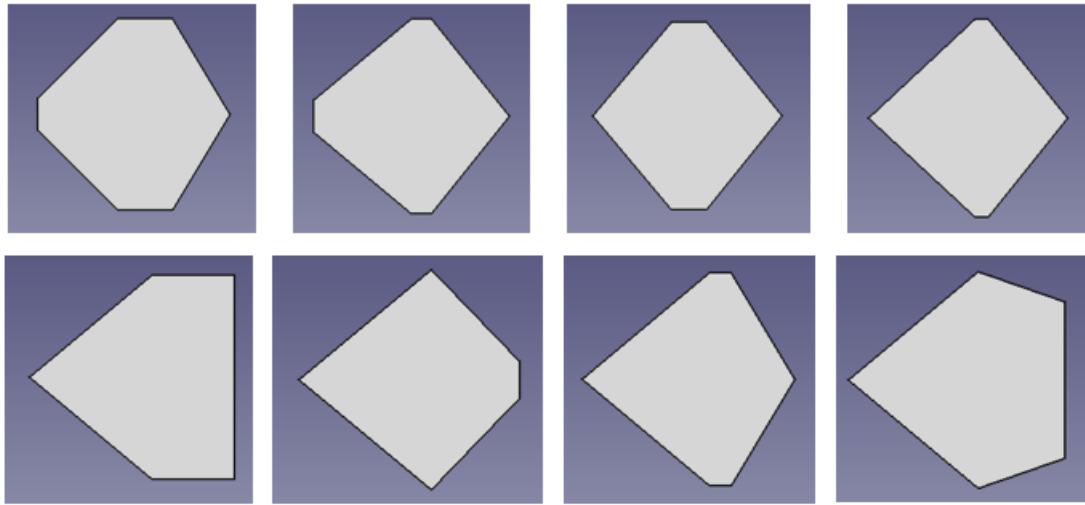


Figure 7.4. Sampling of the different hybrid cross shapes tested while examining magnetic designs. The left side of the shape would be the side against the central blocker.

The new targets could be run with higher beam currents to get similar rates in the detectors, at what appeared to be a proportional rate to the target thickness. At lower electron energies, the carbon-backed targets had far better resolution, as seen in Figure 7.7. The centroids of the peaks also shifted in comparison between the thick and thin target, which follows logically from the thicker target causing greater energy loss.

Also of note in Figure 7.7, the thin target appears to have a higher background, causing the signal-to-noise ratio to decrease. To understand this increase in background, the beam was cut off and a new run was taken. Figure 7.8 shows the detector trigger rate with respect to time during this background run. The background appears to show an exponential decay with a half-life on the order of two minutes.

It was theorized this decay may be from ^{15}O , which would be created from the $^{12}\text{C}(\alpha, n)$ reaction. Using the cross section, obtained from the EXFOR database,

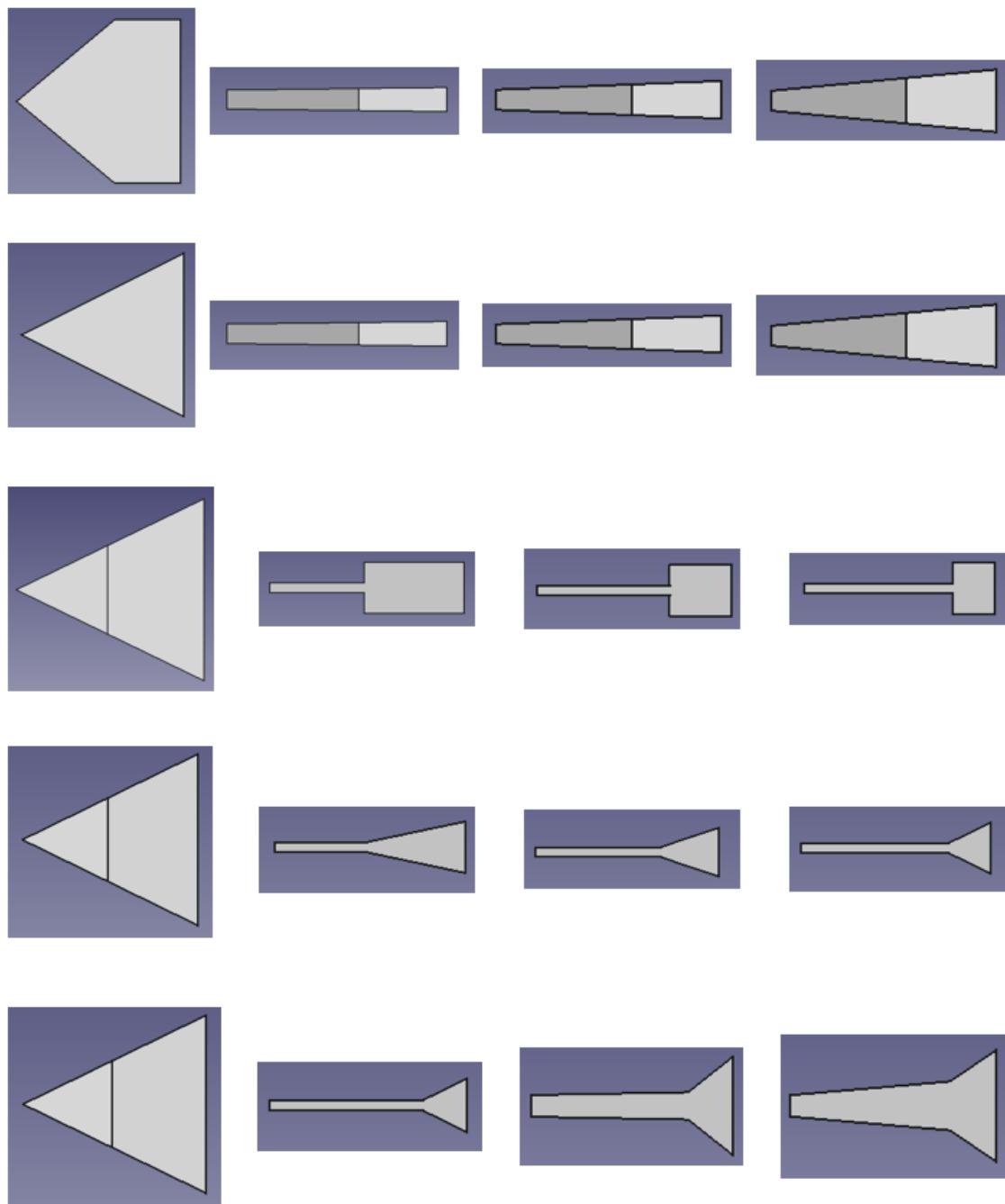


Figure 7.5. Various width profiles tested for the magnets. The left figure shows the cross-shape for the width profiles. The left side goes against the central blocker. Variable gradients were looked at, starting with linear slopes, but evolving into step functions and piecewise linear slopes as the simulations expanded.

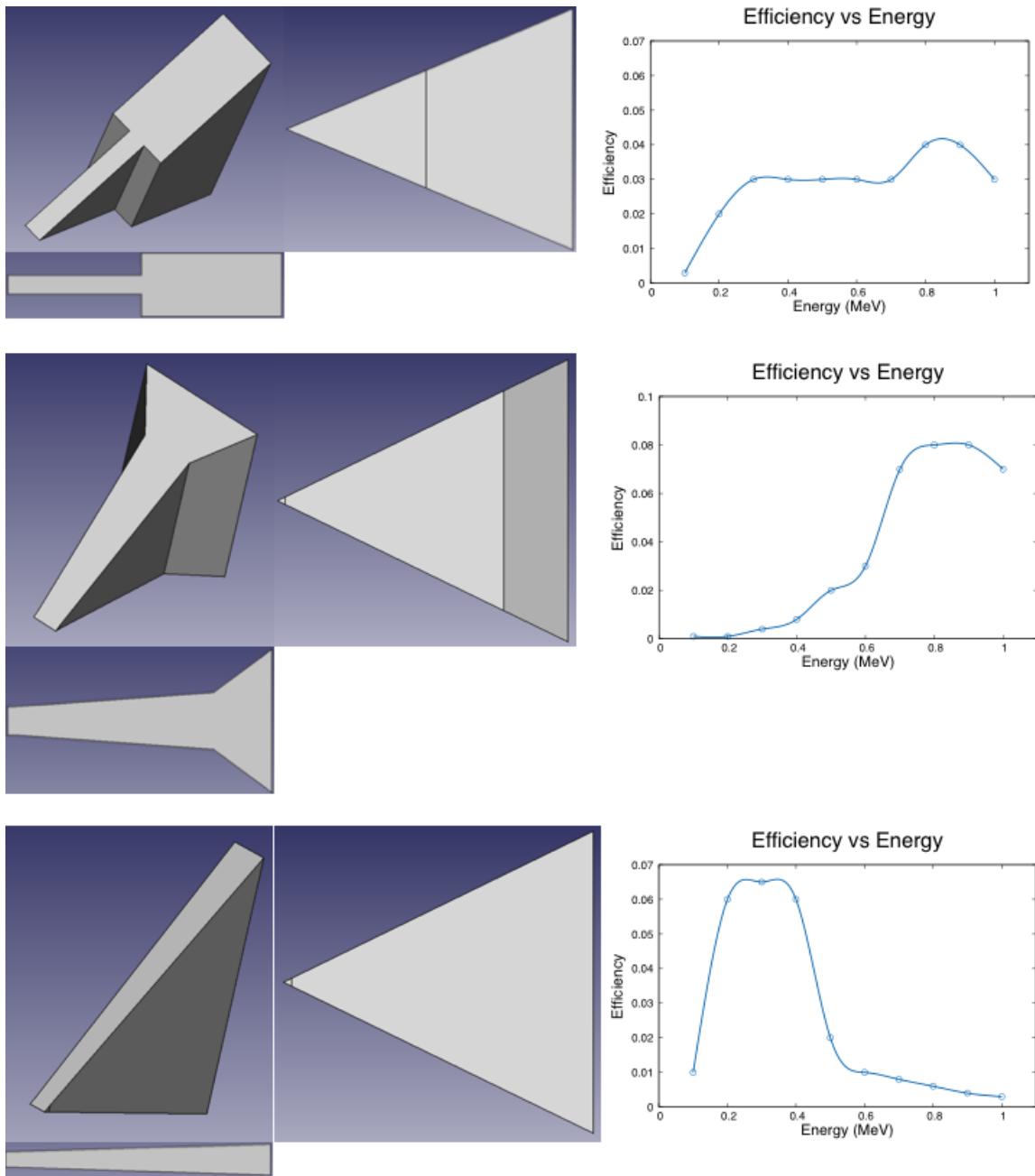


Figure 7.6. The final shapes chosen for testing. The left side goes against the central blocker. The top design has the best overall efficiency, the middle design has the best high energy efficiency, and the bottom design has the best low energy efficiency. These designs have been commissioned for testing.

Target Tests SiLi 3

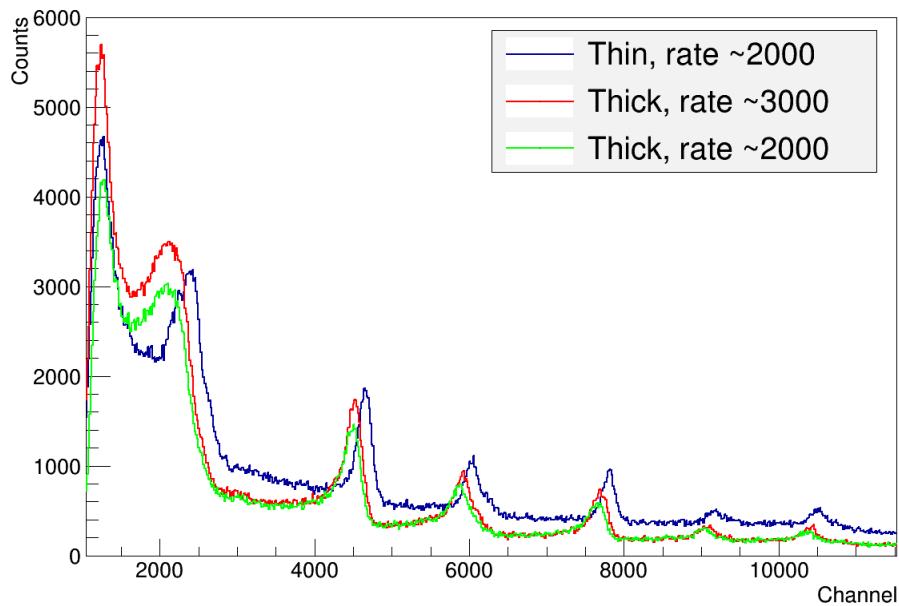


Figure 7.7. Comparison of the thick (self-supporting) and thin (carbon-backed) targets. The spectra were taken during the same experiment. At low energies, the resolution is better in the thin target. Further, the energies of the peaks shift based on which target it is. This spectrum is not energy calibrated.

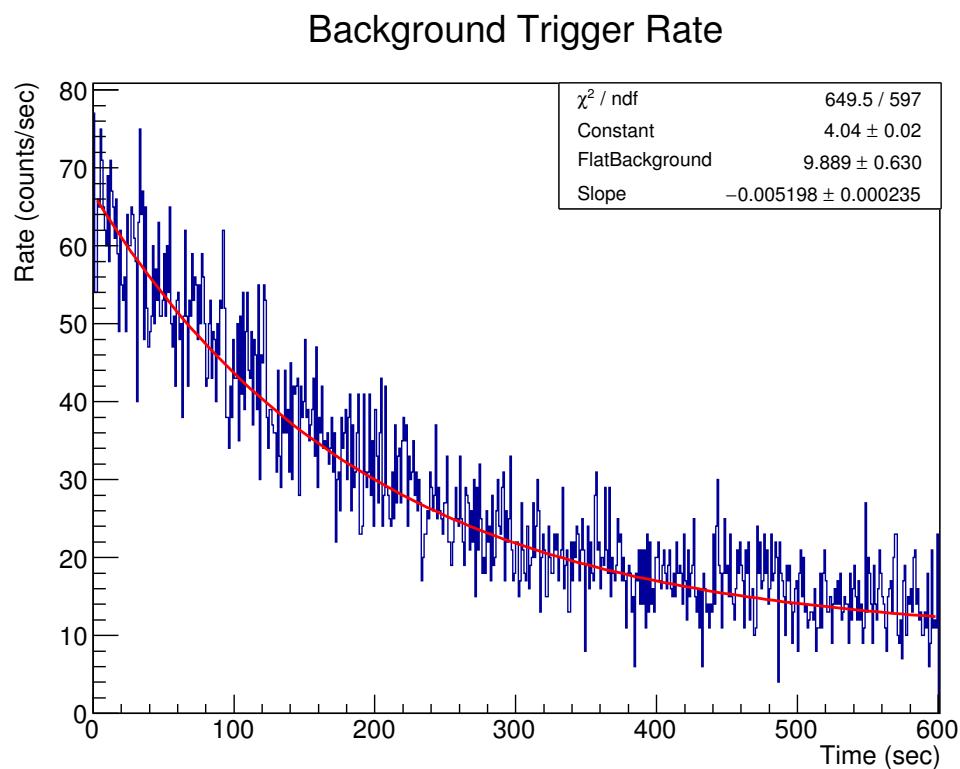


Figure 7.8. Trigger rate of the background with respect to time, after running on one of the carbon-backed targets. There is a clear indication of an exponential decay, shown by the fit. The slope corresponds to a half-life of 133(5) seconds. ^{15}O has a half-life of 122 seconds, within an acceptable error with the low statistics of the run.

the cross section for this reaction at 20 MeV was obtained [9, 68]. This was used to calculate the rate in the detectors if it was ^{15}O being created. These values matched, providing evidence of the new background element. One way this background can be decreased is by using thinner carbon-backing, as $12 \mu\text{g}/\text{cm}^2$ is commercially available. This should reduce the ^{15}O contribution to the background by 40 percent.

7.3 Electronics

After experiments with two different electronics set ups, a third electronics set up was used, with the plan for it to become the standard electronics for future FIREBALL experiments. These electronics use the Mesytec MDPP-16 VME module and the Mesytec MVME data acquisition software [41, 42]. The MVME software from Mesytec has built-in online analysis tools, allowing for real time analysis of the data and can be used with multiple types of VMEs. The MDPP-16 modules replace the conventional NIM modules, replacing external CFDs and TDCs. The gain, threshold, shaping time, and signal rise and decay times are all set within the module.

The MDPP-16 modules can be run in multiple modes, depending on the firmware: QDC, SCP, and RCP modes. The firmware can be changed using a switch on the module, allowing a given module to hold all three firmware versions. For these tests, the SCP, or standard charge sensitive preamplifier, mode is used. Figure 7.9 shows the schematic configuration of the module in SCP mode, and Table 7.1 shows the breakdown of the event words in the system.

One of the benefits of the MDPP-16 modules is the compatibility with other VME modules. Scaler values can be read in with a separate module, such as the CAEN V830, as was done with the GEORGINA setup, discussed in Section 2.7.2. Many of the scaler values can be read in using the MDPP-16, but the addition of the V830 allows for more versatility in rates to record, including comparing rates between the

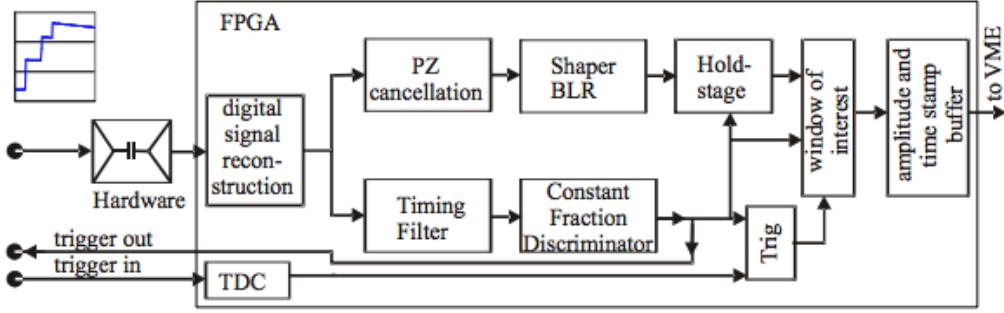


Figure 7.9. A schematic representation of the MDPP-16 software in SCP mode. The amplitude and timing are separateing out into individual signals. Signal properties, shaping time, and the threshold can all be adjusted. Taken from [41].

MDPP-16 and the V830 for deadtime corrections.

7.4 Additional Experiments

In section 1.4, several other spectroscopic fingerprints essential to understanding shape coexistence in this nucleus and other nuclei in the region were mentioned. To that end, for a full and complete understanding of ^{154}Gd , additional spectroscopy experiments would benefit the interpretation of the data. No new gamma spectroscopy has been taken for ^{154}Gd since the discovery of the ten new 0^+ states[45]. The 0^+ states found have no current gamma spectroscopy. Additionally, only one excited 0^+ state in the nucleus has a lifetime, making additional lifetime experiments important.

Unlike ^{154}Gd , ^{156}Gd has not had a large number of states recently found without spectroscopy information, and recent lifetime measurements have been published [3]

7.5 Outlook

Testing with the new magnetic configurations is currently underway, while the new detectors are being manufactured. The electronics have been tested and used extensively at the NSL. Future experiments with the new fIREBall system will have greater sensitivity, contributing to the effort to probe the nuclear chart for E0 transitions in the search for understanding collective modes and shape coexistence.

TABLE 7.1

DATA EVENT - MDPP-16 (32 BIT WORD)

Header							
2 header signature	2 subheader	4	8 module id	3 TDC resolution	3 ADC resolution	10 number of following words	
b01	b00	xxxx	module id	bxxx	bxxx	number of 32 bit data words	
Data Word							
2 data-sig	2	4	2	1	5	16	
b00	01	xxxx	(pu,ov)	Trigger Flag	channel number	ADC Value	
2 data-sig	2	6		1	5	16	
b00	01	xxxxxx		Trigger Flag	channel number + 16	TDC time difference	
End of Event							
2				30			
b11				Event counter/time stamp			

Table of the 32-bit word data structures of the Mesytec MDPP-16 in the SCP Firmware. The headers, event word structure, and end of event structure, where needed, are all listed. In the tables, the first row in a given block is the number of bits used by that piece of data, while the second row is a description of the data. Bits go in descending order from left to right. Data has two words, one for the amplitude and one for the corresponding timing. The (pu,ov) bits are for pile up, underflow, and overflow flags.

APPENDIX A

^{154}GD GATED SPECTRA

The following is a compendium of spectra from the gates on the ^{154}Gd data. A table with the gates and the corresponding background gates used in subtraction is listed. Unless noted, the energy listed is the center of the gate, with a range of ± 1.5 keV.

TABLE A.1

LIST OF ^{154}GD GATES

E_{gate} (keV)	E_{bgd} (keV)	Description
60	130	Due to the low energy, this can be from a variety of places. It is in coincidence with a large number of transitions. One possible transition: $2^- \rightarrow 3^+$ transitioning between the $K^\pi = 2^-$ octopole-vibrational band and the $K^\pi = 2^+$ excited band.
88	130	^{156}Gd contamination
107	130	Two possibilities. $4^- \rightarrow 3^-$ with no known band assignments and $4^+ \rightarrow 3^+$ from a $K^\pi = 1^+$ excited band
110	130	Unable to place.
111.8	130	Unable to place.
123	130	$2_{gs}^+ \rightarrow 0_{gs}^+$
134.8	150	$2_2^+ \rightarrow 0_2^+$ in the 1 st excited 0 ⁺ band.
141	150	$6^+ \rightarrow 5^+$ in $K^\pi = 4^+$ band

TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
162	206	$(7^+) \rightarrow 6^+$ in $K^\pi = 4^+$ band
165	206	$9^- \rightarrow 7^-$ in $K^\pi = 7^-$ band
167	206	No assignment other than the same for 165 gate.
171	187	$8^- \rightarrow 7^-$ in $K^\pi = 7^-$ band
181	206	Three possible transitions. $2_\gamma^+ \rightarrow 2_2^+$ with end state in the 1 st excited 0 ⁺ band. $8^+ \rightarrow 7^+$ in $K^\pi = 4^+$ band. Third transition is too high an energy and spin to be reasonably considered.
188	206	$2^- \rightarrow 2^+$ transitioning between the $K^\pi = 2^-$ octopole-vibrational band and the $K^\pi = 2^+$ excited band.
190	205	Used ±4 keV. Wide gate check.
192	206	Unable to assign.
199	206	¹⁵⁶ Gd contamination; $3^- \rightarrow 2_3^+$ between $K^\pi = 1^-$ octopole-vibrational band and 2 nd excited 0 ⁺ band.
202	206	$3^- \rightarrow 1^-$ in $K^\pi = 1^-$ octupole-vibrational band.
210	290	Possibly $3^- \rightarrow 5^-$ from the $K^\pi = 1^-$ octupole-vibrational band to the $K^\pi = 0^-$ octupole-vibrational band.
218	290	$3^- \rightarrow 2^-$ from the $K^\pi = 1^-$ octupole-vibrational band
224	290	$7^- \rightarrow 6^+$ from the $K^\pi = 7^-$ to the $K^\pi = 4^+$ band
227	290	No assignment other than the same for 224 gate.
232.1	290	$4_2^+ \rightarrow 2_2^+$ in 1 st excited 0 ⁺ band
235	278	$2_3^+ \rightarrow 0_3^+$ in the 2 nd excited 0 ⁺ band
238	278	Unable to place. Seen as going into 2_{gs}^+ .
247.9	290	$4_{gs}^+ \rightarrow 2_{gs}^+$

TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
265	290	$6^+ \rightarrow 4^+$ in K=4 ⁺ band
267.5	290	$4_\gamma^+ \rightarrow 2_\gamma^+$
274	290	Unable to place
282	290	$4_3^+ \rightarrow 2_3^+$ in 2 nd excited 0 ⁺ band.
285	290	Unable to place
297	326	¹⁵⁶ Gd contamination; $4^- \rightarrow 4_\gamma^+$ where 4 ⁻ is in the $K^\pi = 1^-$ octupole-vibrational band.
303	326	Best agreement $7^+ \rightarrow 5^+$ from K=4 ⁺ band.
306	326	Coincidence indicates $6^+ \rightarrow 6^+$ for K=4 ⁺ band to γ -vibrational band.
315.6	326	$2_\gamma^+ \rightarrow 0_2^+$
318.3	326	$6_2^+ \rightarrow 4_2^+$ in 1 st excited 0 ⁺ band
325	400	Unable to place. Seen as going into 6_{gs}^+ .
333	400	May be $0_6^+ \rightarrow 1^-$ where 1 ⁻ is from the $K^\pi = 0^-$ octopole-vibrational band.
339	400	$5^+ \rightarrow 5_\gamma^+$ where 5 ⁺ is from the $K^\pi = 4^+$ band
343.0	400	$6_\gamma^+ \rightarrow 4_\gamma^+$
346.6	400	$6_{gs}^+ \rightarrow 4_{gs}^+$
381	400	¹⁵⁶ Gd contamination; $4^+ \rightarrow 4_\gamma^+$ where 4 ⁺ is from the $K^\pi = 4^+$ band
390.6	400	$8_2^+ \rightarrow 6_2^+$ in 1 st excited 0 ⁺ band
408	421	$3^+ \rightarrow 3^-$ between 2 nd excited 2 ⁺ band and $K^\pi = 0^-$ octopole-vibrational band.
412	421	Unknown J^π to 6 ⁺ in $K^\pi = 4^-+$ band.
417	421	$4^- \rightarrow 5^+$ with 4 ⁻ unassigned and 5 ⁺ in $K^\pi = 4^-+$ band.

TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bkgd} (keV)	Description
426.8	460	$8_{gs}^+ \rightarrow 6_{gs}^+$
435	460	Three possibilities. $3^- \rightarrow 2_2^+$ between $K^\pi = 0^-$ octopole-vibrational band and 1^{st} excited 0^+ band; $4_7^+ \rightarrow 4^+$ between 6^{th} excited 0^+ band and $K^\pi = 4^+$ band; $10^- \rightarrow 8^-$ in $K^\pi = 7^-$ band.
437.7	460	$10_2^+ \rightarrow 8_2^+$ (1^{st} excited 0^+ band)
451	460	$8^{(-)} \rightarrow 7^-$ with $8^{(-)}$ in the two-quasineutron band and no band assignment for 7^- .
465	500	$2_7^+ \rightarrow 3^-$ between the 6^{th} excited 0^+ band and $K^\pi = 0^-$ octopole-vibrational band
467	500	$2^- \rightarrow 3^-$ between the $K^\pi = 2^-$ octopole-vibrational band and $K^\pi = 0^-$ octopole-vibrational band.
471	500	Only agreement is levels with unknown J^π
480	500	$6^+ \rightarrow 5_\gamma^+$ with 6^+ from the $K^\pi = 4^+$ band.
492	500	$10_{gs}^+ \rightarrow 8_{gs}^+$
501	530	$0_3^+ \rightarrow 0_2^+$ gamma equivalent.
505	530	$5^+ \rightarrow 4_\gamma^+$ with 5^+ from the $K^\pi = 4^+$ band.
508	530	Unable to place
557.6	587.5	$0_2^+ \rightarrow 2_{gs}^+$ with the 0^+ from the 1^{st} excited 0^+ band.
568	588	$3^- \rightarrow 4_{gs}^+$ where 3^- is from the $K^\pi = 1^-$ octopole-vibrational band
577	631.5	$0_7^+ \rightarrow 2_\gamma^+$ with the 0^+ from the 6^{th} excited 0^+ band.
596	631.5	$0_9^+ \rightarrow 1^-$ with the 0^+ from the 8^{th} excited 0^+ band and 1^- from the $K^\pi = 0^-$ octopole-vibrational band.
598	631.5	$4^+ \rightarrow 4_2^+$ between the $K^\pi = 4^+$ band and 1^{st} excited 0^+ band

TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
600	631.5	$1^- \rightarrow 2_2^+$ between the $K^\pi = 1^-$ octopole-vibrational band and 1 st excited 0 ⁺ band
604	631.5	$2^+ \rightarrow 2_2^+$ between the 2 nd excited 0 ⁺ band and 1 st excited 0 ⁺ band
612	631.5	$8_2^+ \rightarrow 8_{gs}^+$ with the 8 ⁺ from the 1 st excited 0 ⁺ band
630	734	Unable to place
637	734	$(8^+, 9^+) \rightarrow 10_{gs}^+$
642	734	$5^+ \rightarrow 3_\gamma^+$ with the 5 ⁺ state in the $K^\pi = 4^+$ band
648.3	734	$6_2^+ \rightarrow 6_{gs}^+$ with the 6 ⁺ from the 1 st excited 0 ⁺ band
650	734	$4^+ \rightarrow 2_\gamma^+$ with the 4 ⁺ state in the $K^\pi = 4^+$ band
654	734	$4_3^+ \rightarrow 4_2^+$ between the 2 nd excited 0 ⁺ band and 1 st excited 0 ⁺ band
676.6	734	$4_2^+ \rightarrow 4_{gs}^+$ with the 4 ⁺ from the 1 st excited 0 ⁺ band
680.7	734	Gated on as γ equivalent of $0^2 \rightarrow 0_{gs}^+$
682	734	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
685	750	Unable to place with a known transition. However, the energy lines up extremely well for $(5^-) \rightarrow 6_{gs}^+$ where the (5 ⁻) is from the $K^\pi = 0^-$ octopole-vibrational band
692.4	734	$2_2^+ \rightarrow 2_{gs}^+$ with the 2 ⁺ from the 1 st excited 0 ⁺ band
696	750	$(5^-) \rightarrow 4_{gs}^+$
722	750	$5^+ \rightarrow 4_2^+$ between the $K^\pi = 4^+$ band and the 1 st excited 0 ⁺ band.
724	734	$2^- \rightarrow 2_\gamma^+$ where the 2 ⁻ is from the $K^\pi = 2^-$ octopole-vibrational band. Also possibly $5^+ \rightarrow 4_2^+$ between the $K^\pi = 4^+$ band and the 1 st excited 0 ⁺ band.

TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bkgd} (keV)	Description
758.4	776	$0_7^+ \rightarrow 2_2^+$ between the 6 th excited 0 ⁺ band and 1 st excited 0 ⁺ band. Also $3_\gamma^+ \rightarrow 4_{gs}^+$
761.8	776	$(1, 2^+) \rightarrow 0_3^+$ where the 0 ⁺ is from the 2 nd excited 0 ⁺ band.
797	802	No reasonable known agreement. Seen in 123 and 1004 gates.
811	820	Unable to place. Seen as going into 4_{gs}^+ .
834	905	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
837	905	$2^+ \rightarrow 2^+$ with no band assignments.
840	905	Unable to place.
842	914	Unable to place. Seen as going into 6_{gs}^+ .
844	905	$3^+ \rightarrow 2_2^+$ from 2 nd excited 2 ⁺ band to
850.6	905	$2^+ \rightarrow 2_2^+$ where 2 ⁺ is from the 2 nd excited $K^\pi = 2^+$ band
874	905	$2_\gamma^+ \rightarrow 2_{gs}^+$
891	905	$_\gamma^+ \rightarrow 4_{gs}^+$
894	905	Possibly the same as 891. Also $(9^-) \rightarrow 8_{gs}^+$ with (9^-) from the $K^\pi = 0^-$ octopole-vibrational band.
904	914	Unable to place. Seen as going into 6_{gs}^+ .
962	973	$2_7^+ \rightarrow 2_2^+$ between the 6 th and 1 st excited 0 ⁺ bands.
996	1020	$2_\gamma^+ \rightarrow 0_{gs}^+$ and $6_2^+ \rightarrow 4_{gs}^+$ where 6 ⁺ is from the 1 st excited 0 ⁺ band.
1004.7	1020	$3_\gamma^+ \rightarrow 2_{gs}^+$
1021	1062	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1043	1062	Gated on as a possible $0^+ \rightarrow 2^+$ transition.

TABLE A.1 (CONTINUED)

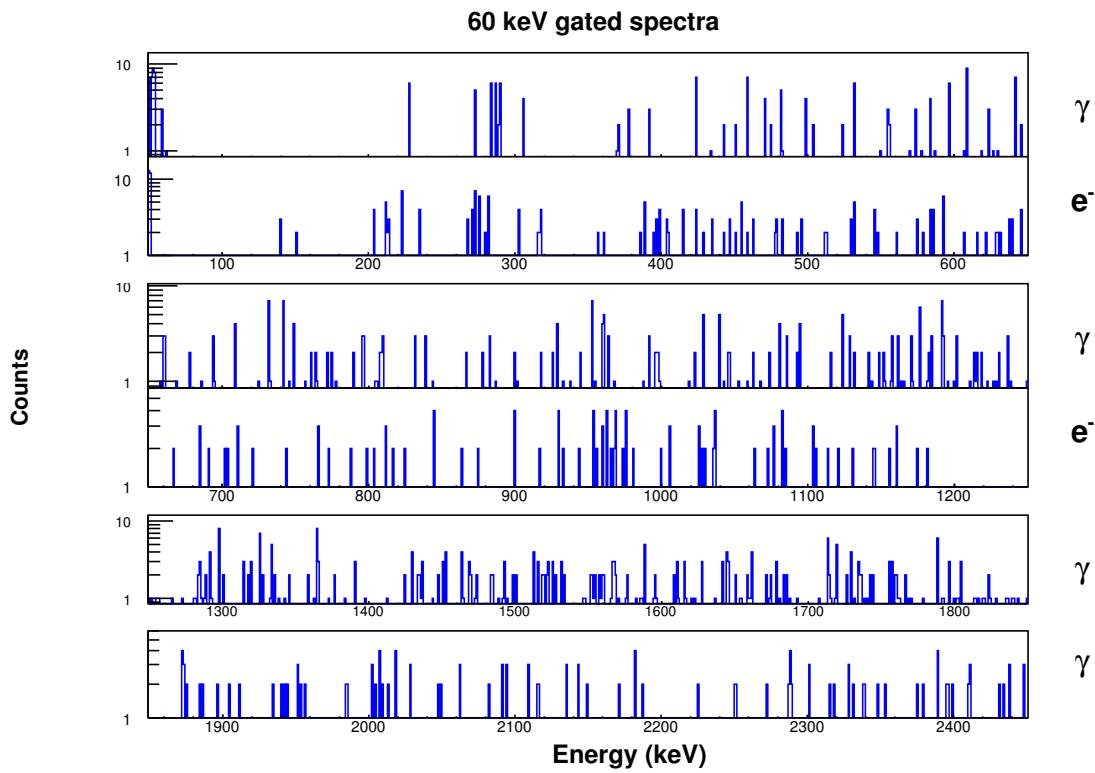
E_{gate} (keV)	E_{bkgd} (keV)	Description
1059	1074	$0_3^+ \rightarrow 2_{gs}^+$ where 0^+ is from the 2 nd excited 0^+ band. Also $5_\gamma^+ \rightarrow 4_{gs}^+$
1083	1100	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1127	1135	$3^- \rightarrow 2_{gs}^+$ where 3^- is from the $K^\pi = 0^-$ octopole-vibrational band. Also a possible $0^+ \rightarrow 2^+$ transition.
1140	1250	$4_\gamma^+ \rightarrow 2_{gs}^+$
1194	1250	$6^+ \rightarrow 6_{gs}^+$ with first 6^+ from K=4 ⁺ band
1213	1250	Unable to place.
1217	1250	Unknown J^π to 4_2^+ from the 1 st excited 0^+ band.
1229	1250	$3 \rightarrow 4_2^+$ from the 1 st excited 0^+ band. Initiate state parity unknown.
1237	1250	$6_\gamma^+ \rightarrow 4_{gs}^+$
1245	1323	$3^- \rightarrow 4_{gs}^+$ where 3^- is from K=1 ⁻ octopole-vibrational band.
1303	1325	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1323	1390	Unknown J^π to 2^- from the $K^\pi = 1^-$ octopole-vibrational band.
1339	1390	$3^- \rightarrow 2_\gamma^+$ where the 3^- has no assignment.
1346	1390	$2_6^+ \rightarrow 4_{gs}^+$ where 2_6^+ is from the 0_6^+ band.
1370	1390	$1^+ \rightarrow 2_2^+$ between a $K^\pi = 1^+$ band and the 1 st excited 0^+ band.
1390	1440	Two possibilities. $4^+ \rightarrow 2_\gamma^+$ where the 4^+ has no assignment; $2^+ \rightarrow 4_\gamma^+$ where the 2^+ has no assignment.
1421	1440	$7^- \rightarrow 6_{gs}^+$ where 7^- is from the $K^\pi = 7^-$ band
1431	1565	Unable to place. Seen as going into 6_{gs}^+ .

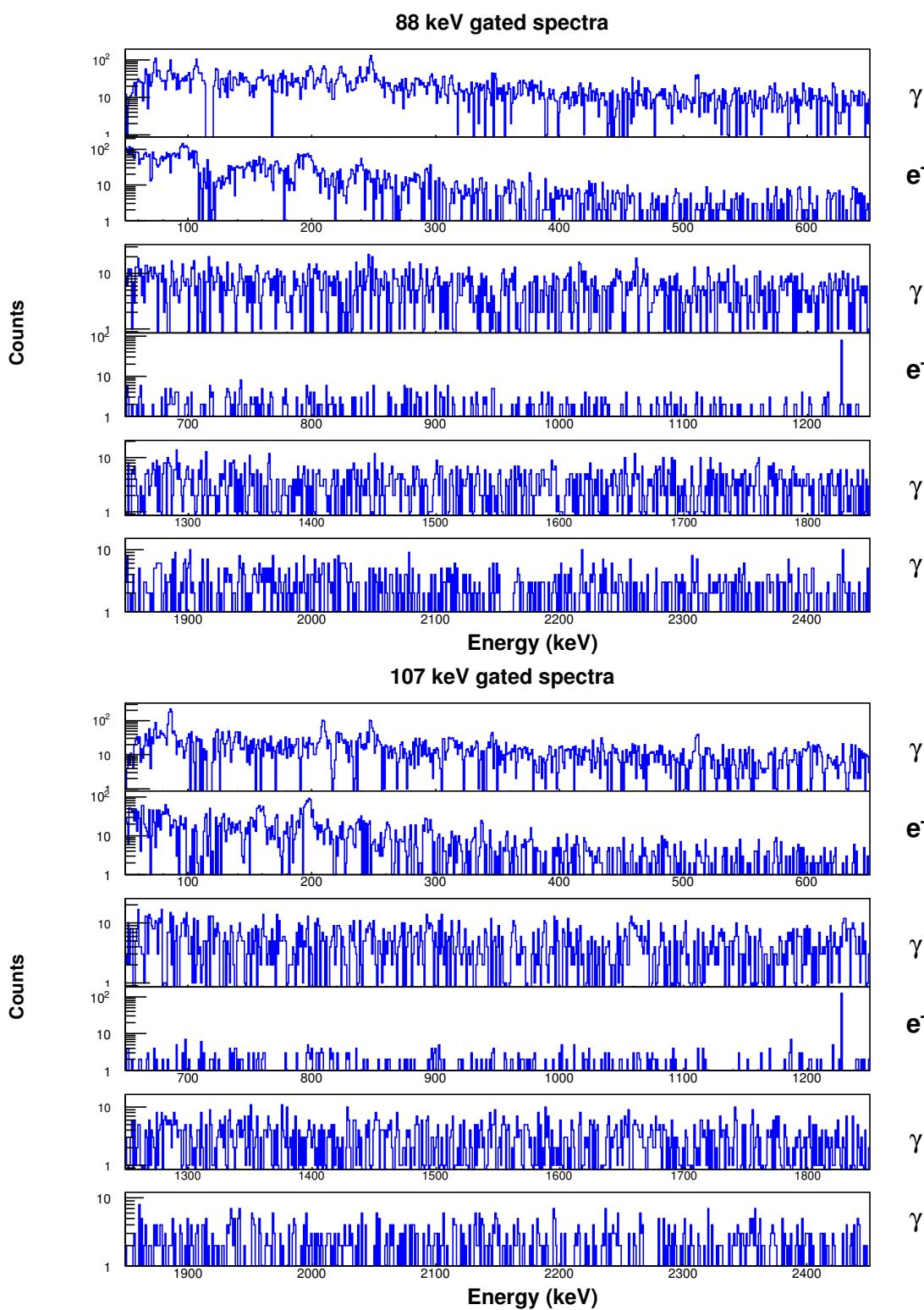
TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
1450	1470	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1488	1500	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1514	1565	Unable to place. Seen as going into 4_{gs}^+ .
1527	1550	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1572	1632	Unable to place. Seen as going into 4_{gs}^+ .
1589	1645	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1658	1750	Unable to place. Seen as going into 4_{gs}^+ .
1669	1720	Gated on as a possible $0^+ \rightarrow 2^+$ transition.
1671	1750	Unable to place. Seen as going into 4_{gs}^+ .
1700	1750	Unable to place. Seen as going into 4_{gs}^+ .
1713.5	1750	Gated on as going into 2_{gs}^+ . Unable to make placement..
1788	1800	Gated on as going into 2_{gs}^+ . Unable to make placement.
1810	1830	Gated on as going into 2_{gs}^+ . Unable to make placement.
1895	1919	State with unknown assignment ($2^+, 3, 4^+$) to 4_{gs}^+
1940	2000	Unable to place. Seen as going into 2_{gs}^+ .
2007	2107	Unable to place. Seen as going into 2_{gs}^+ .
2025	2042	Gated on as going into 2_{gs}^+ . Unable to make placement.
2173	2264	Unable to place. Seen as going into 2_{gs}^+ .
2310	2420	Unable to place. Seen as going into 2_{gs}^+ .
2435	2453	Unable to place. Seen as going into 2_{gs}^+ .
2628	2820	Unable to place. Seen as going into 2_{gs}^+ .

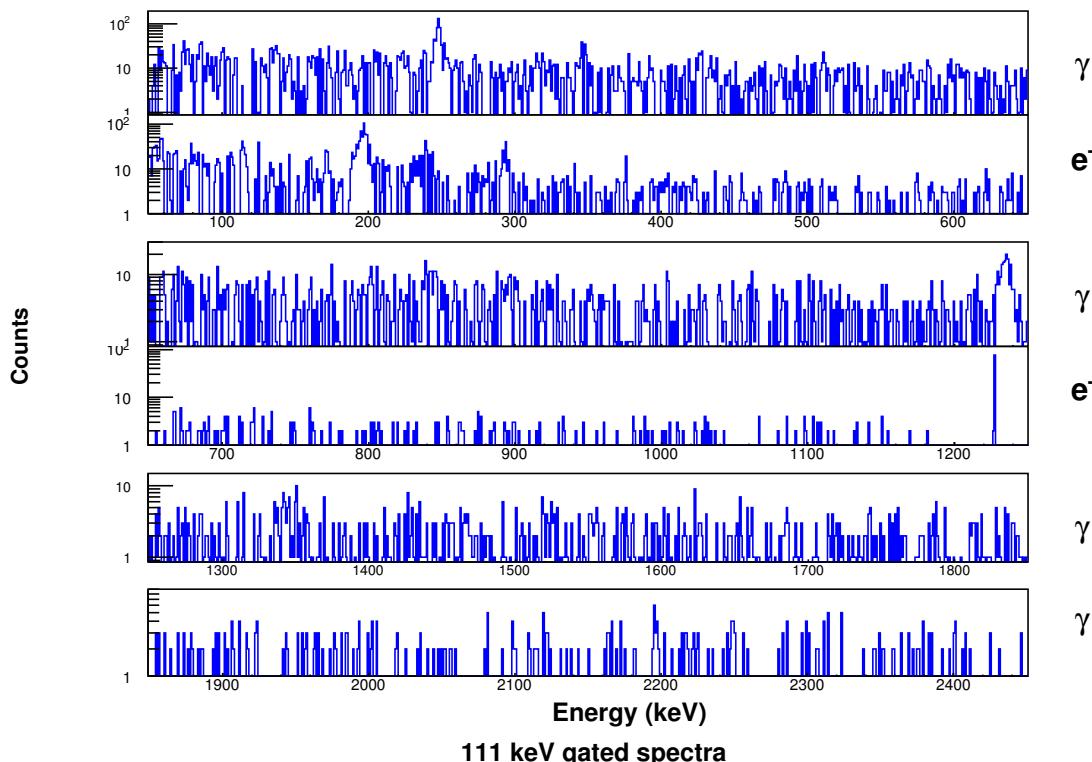
TABLE A.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
2648	2733	Unable to place.
2682	2733	Unable to place.

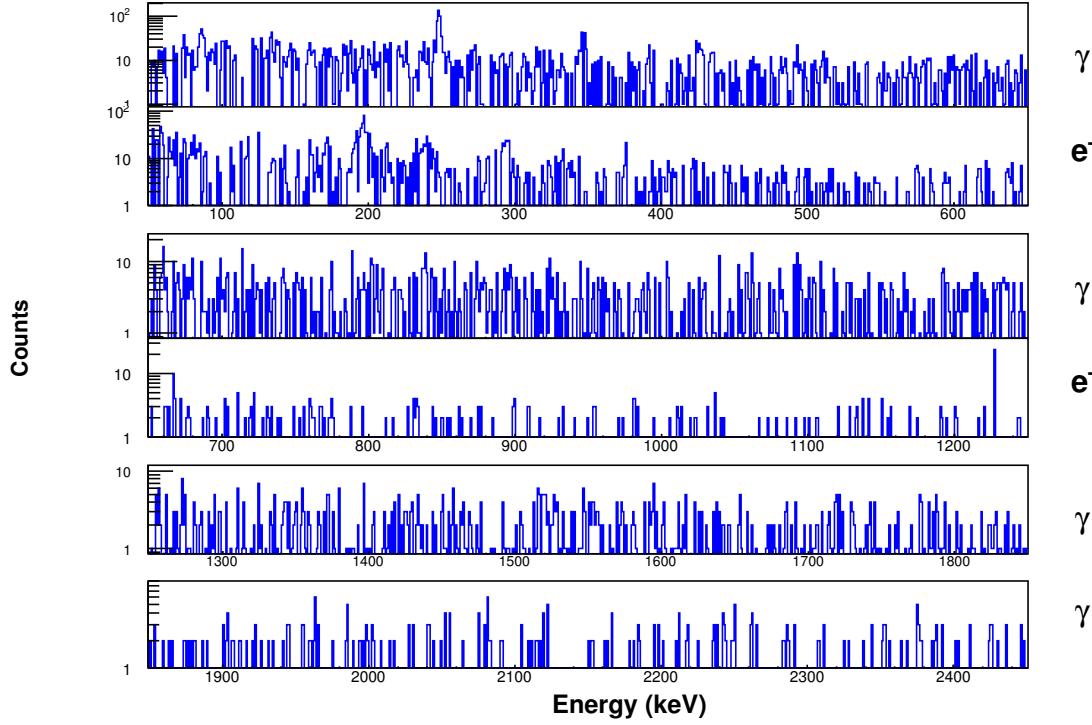




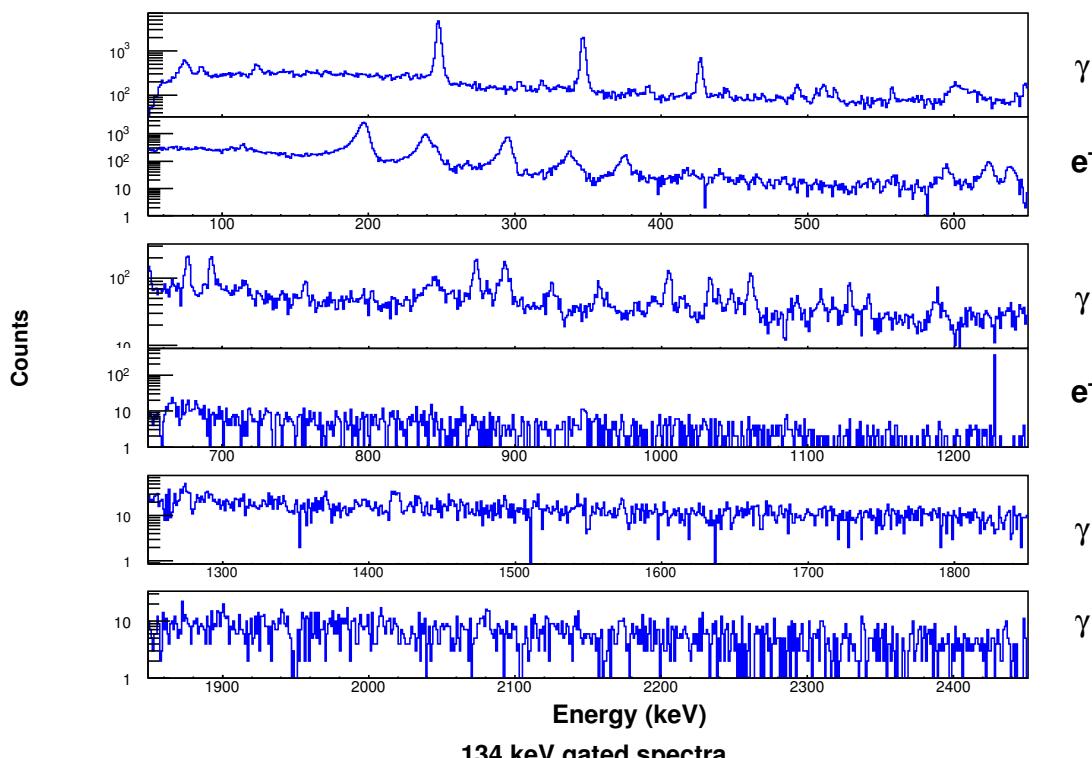
110 keV gated spectra



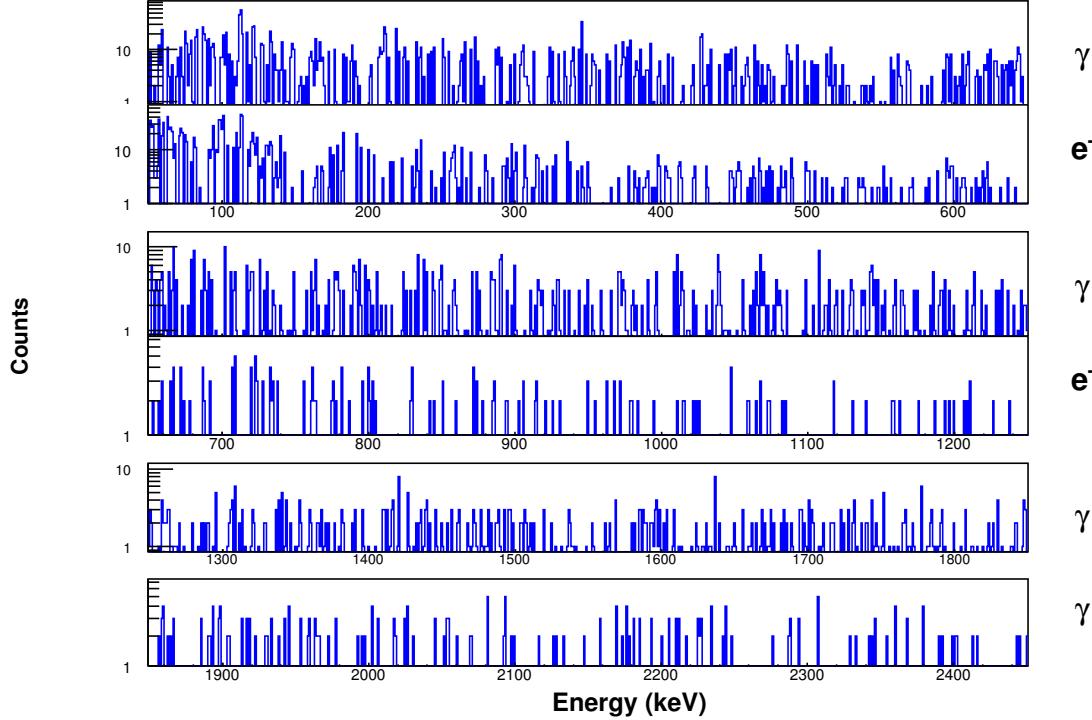
111 keV gated spectra



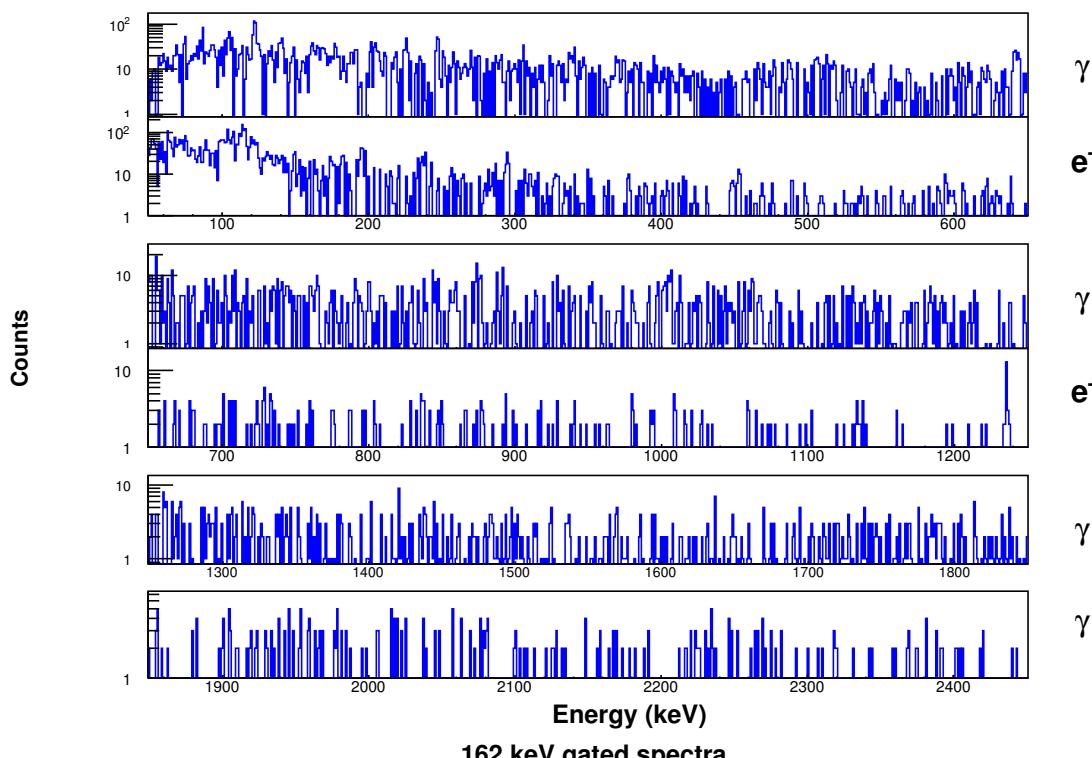
123 keV gated spectra



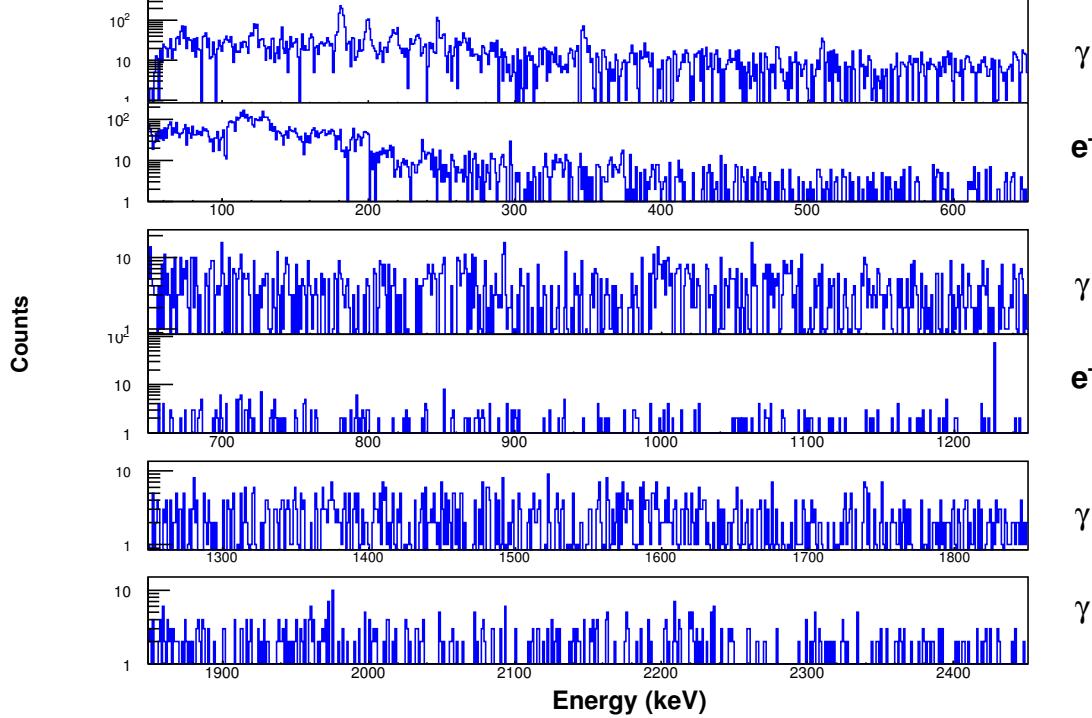
134 keV gated spectra



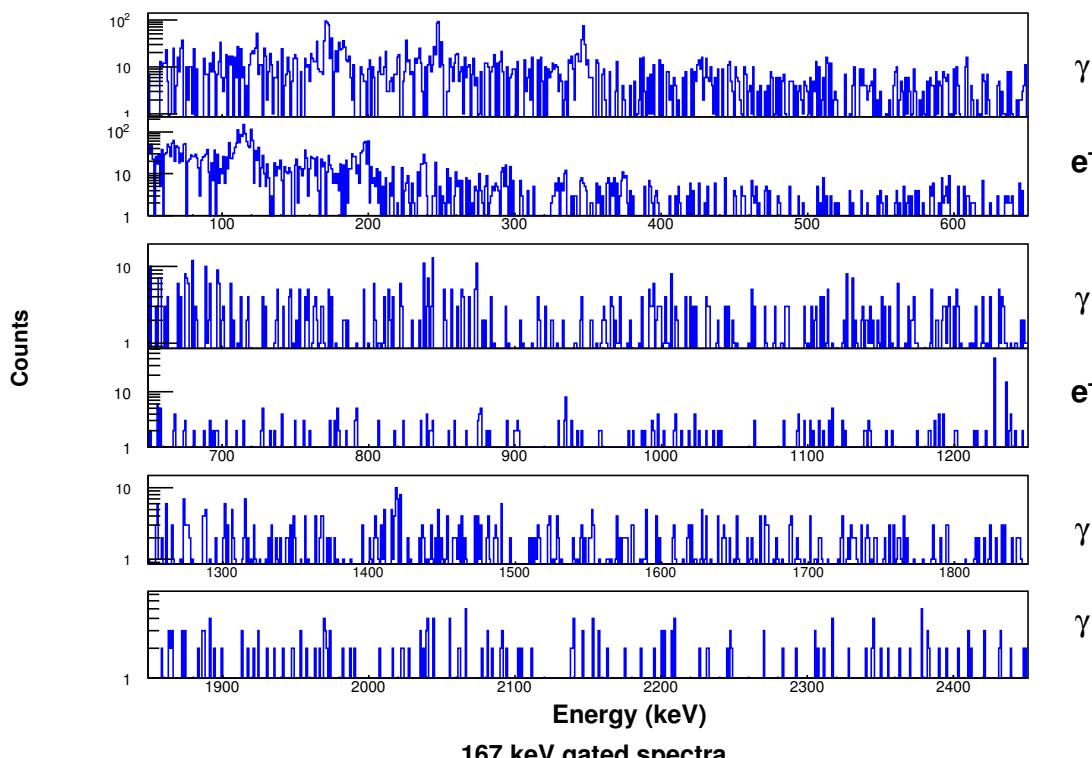
141 keV gated spectra



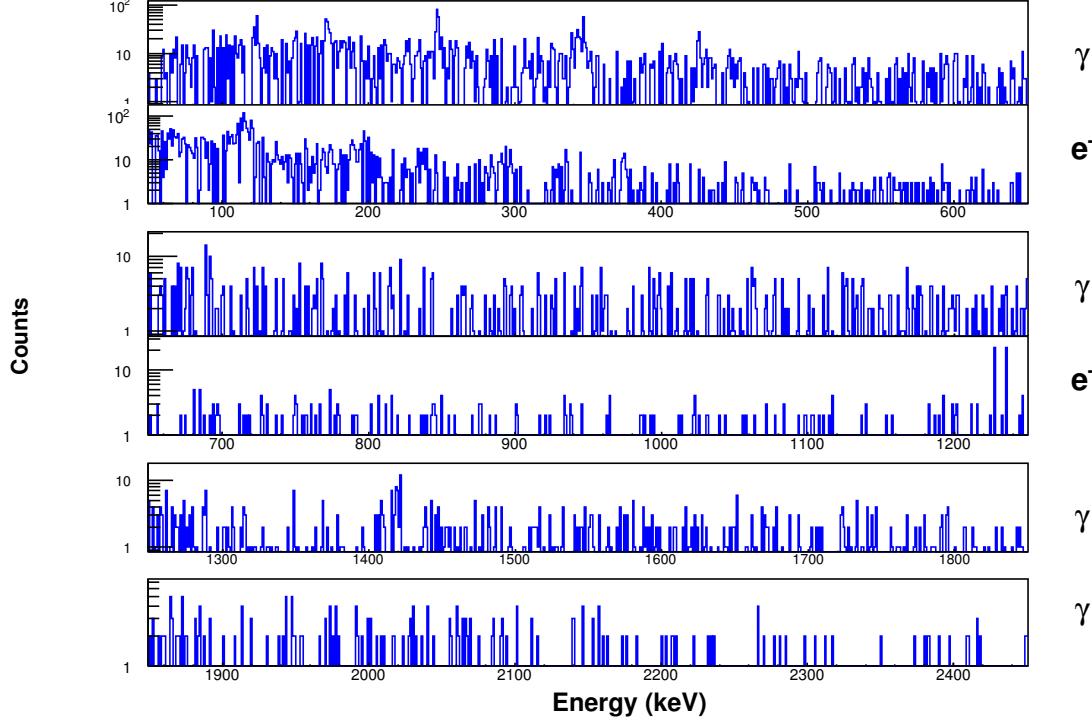
162 keV gated spectra

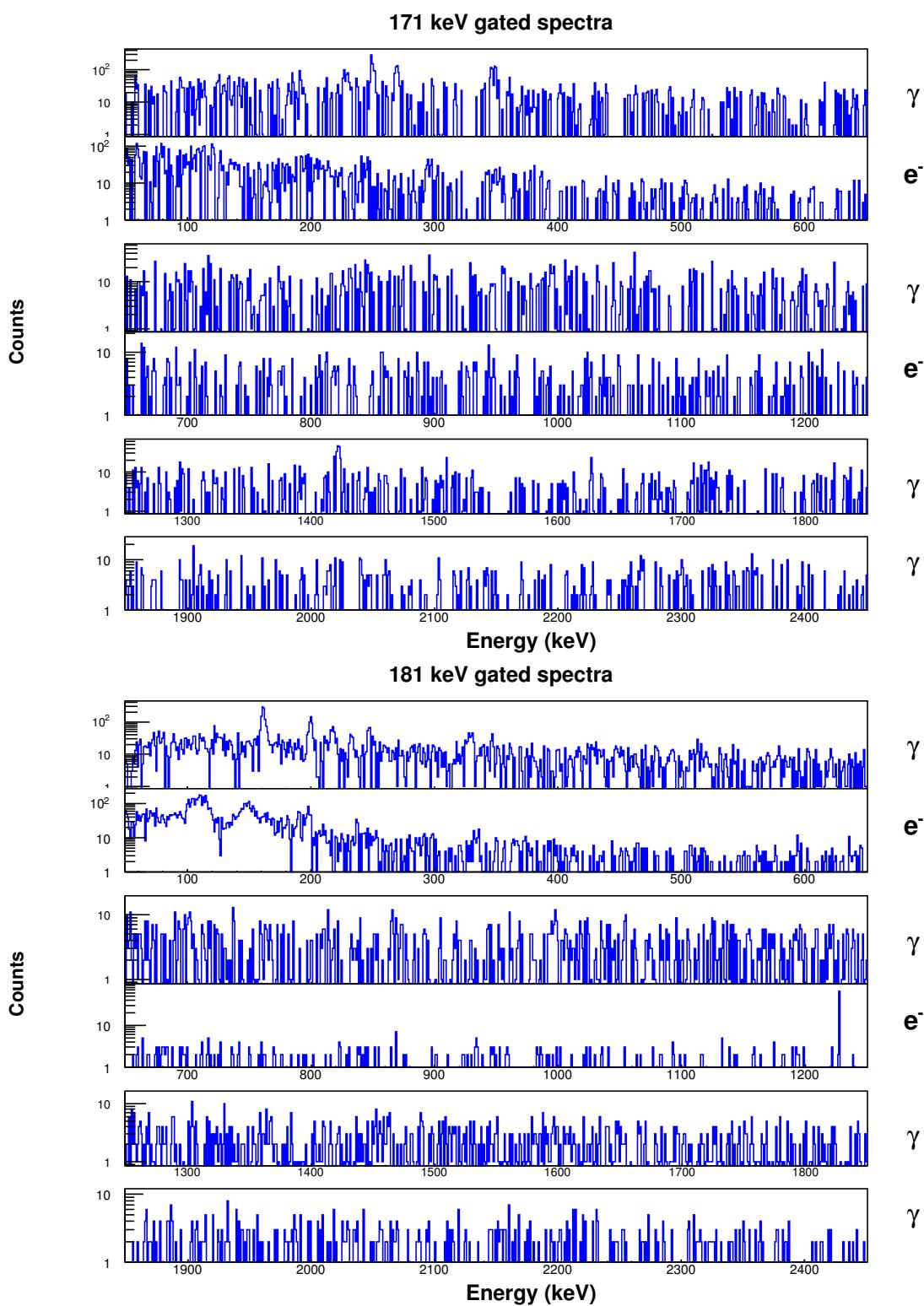


165 keV gated spectra

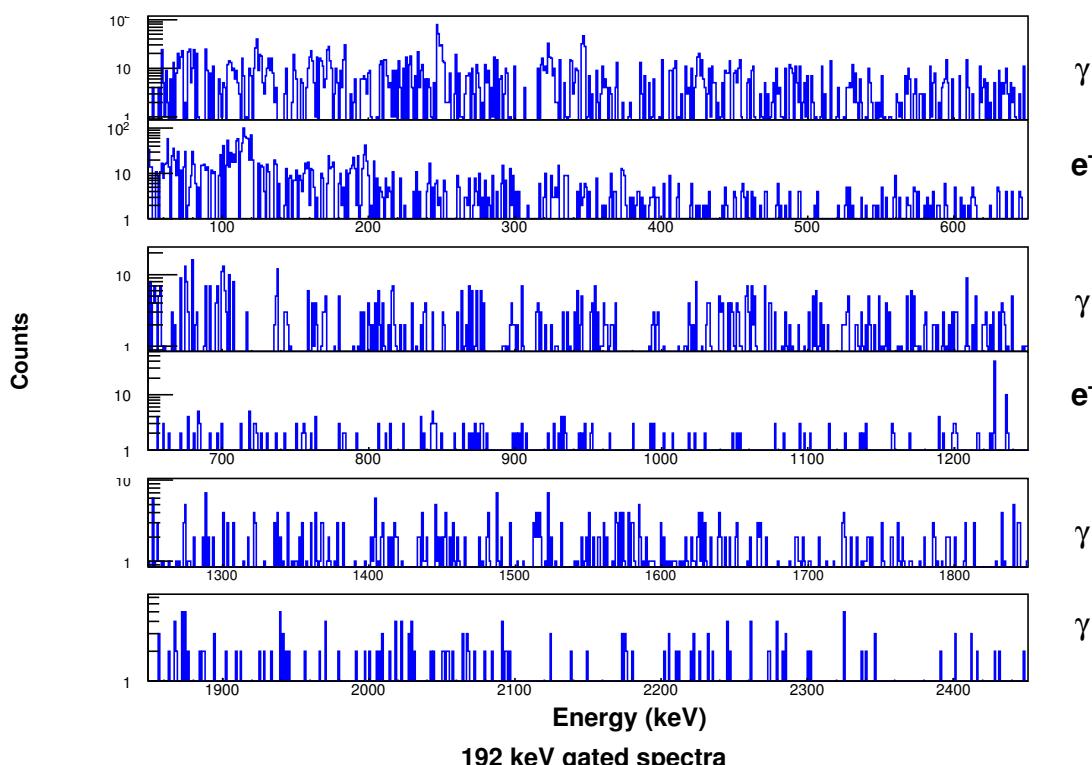


167 keV gated spectra

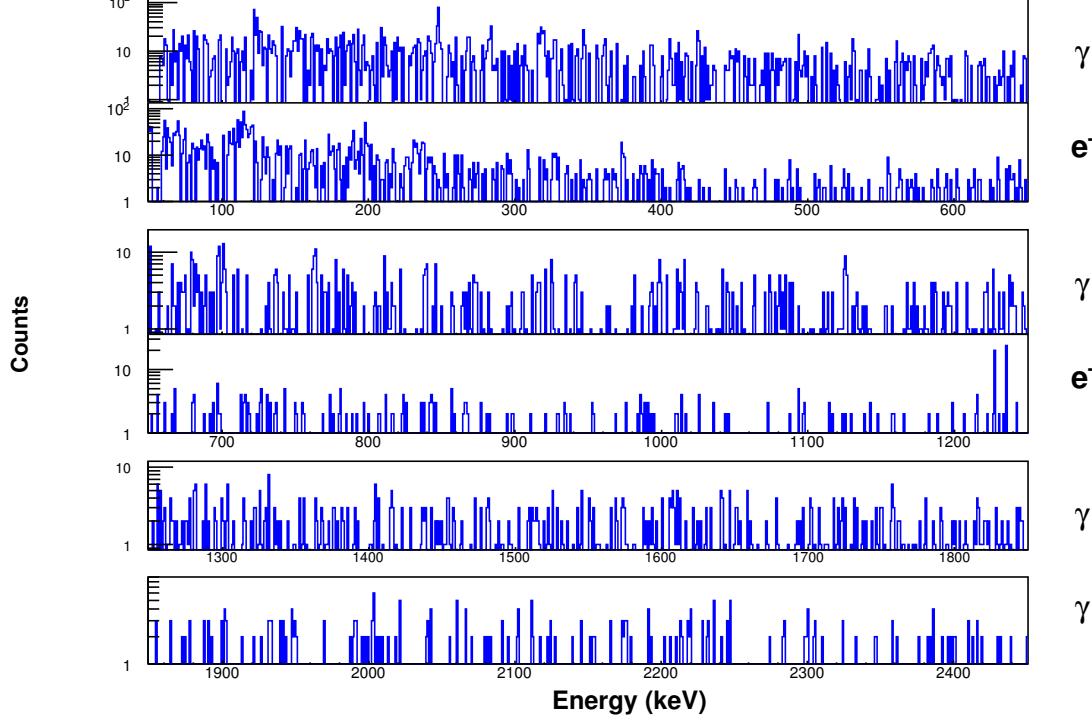




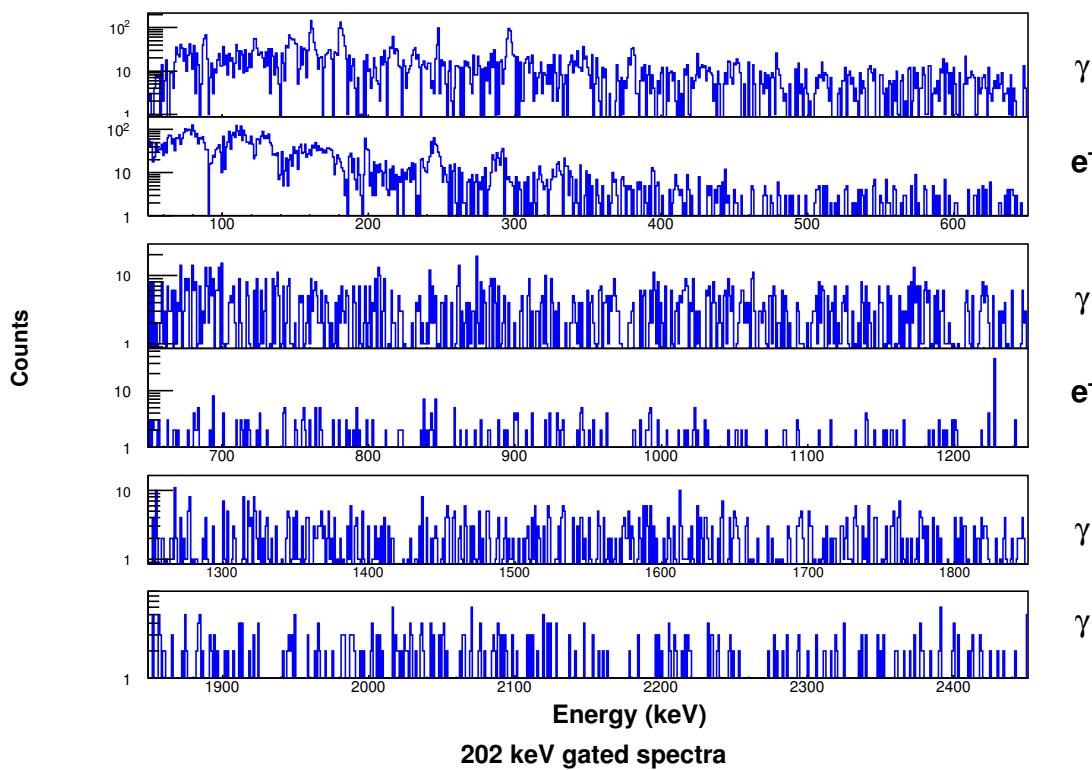
188 keV gated spectra



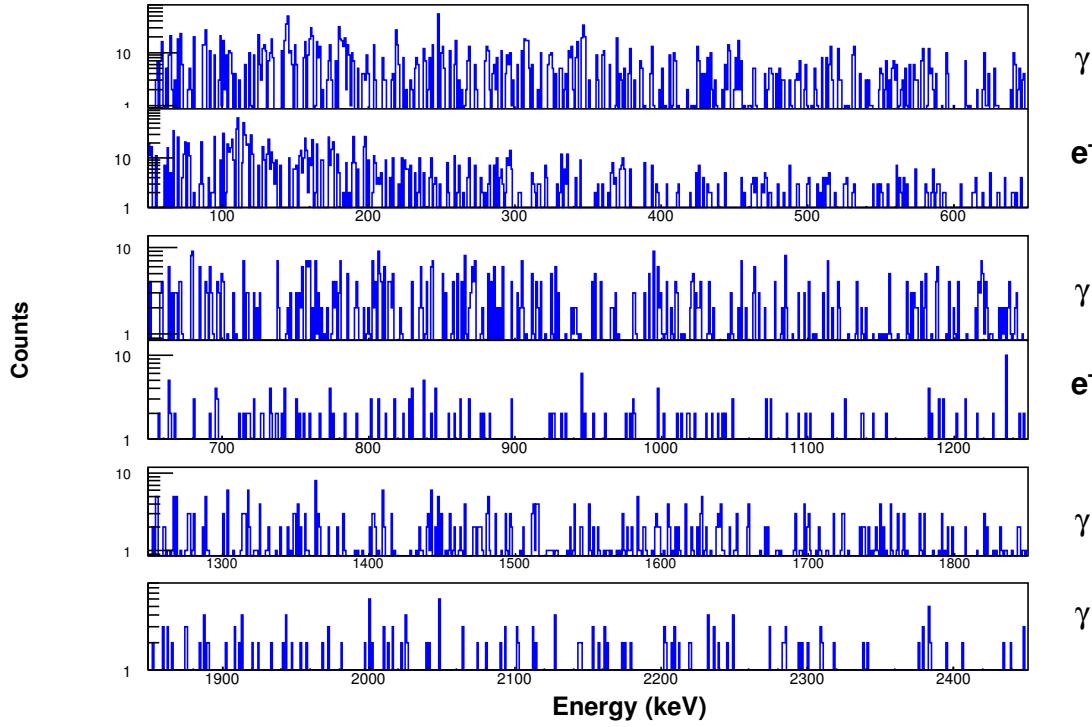
192 keV gated spectra

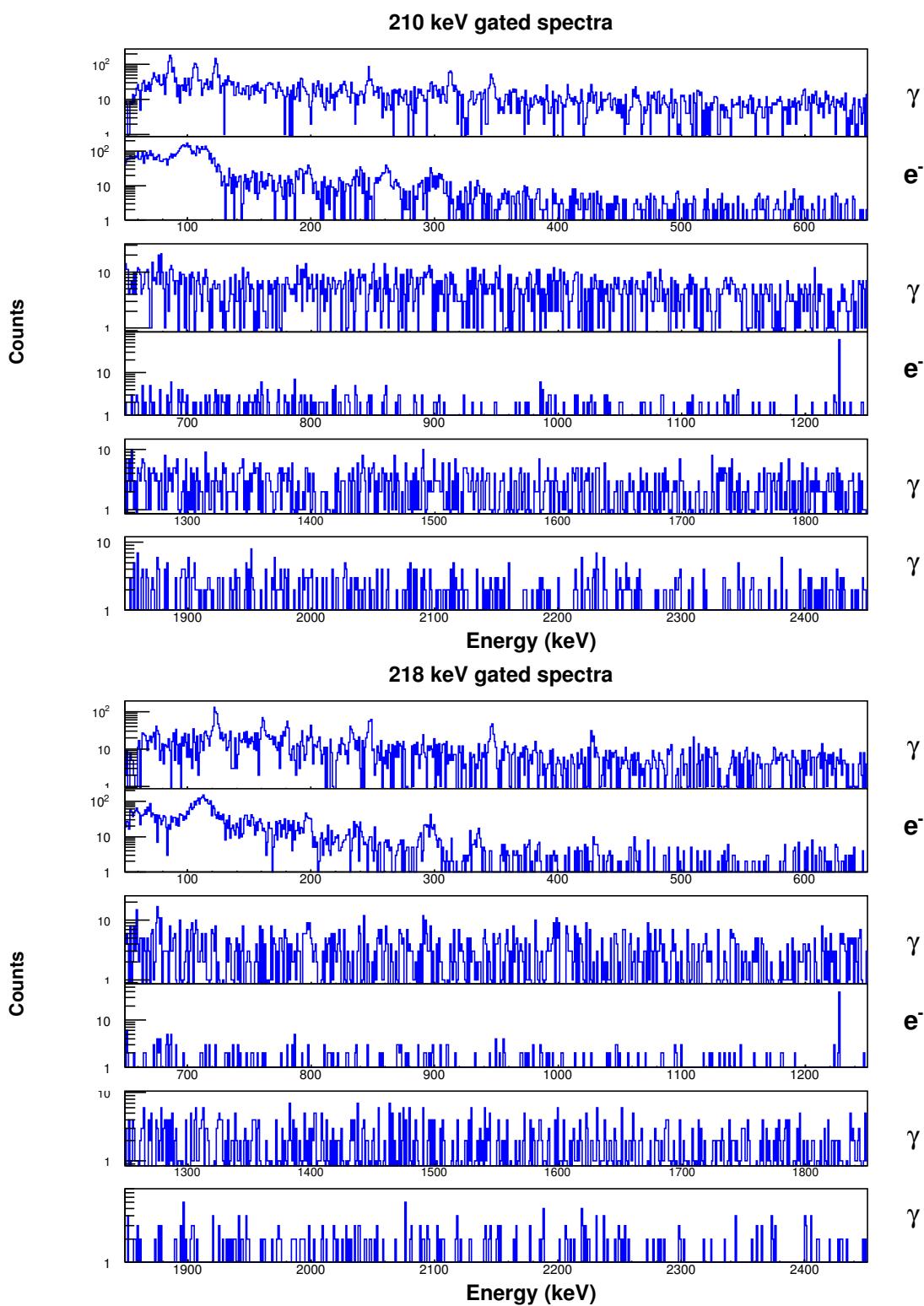


199 keV gated spectra

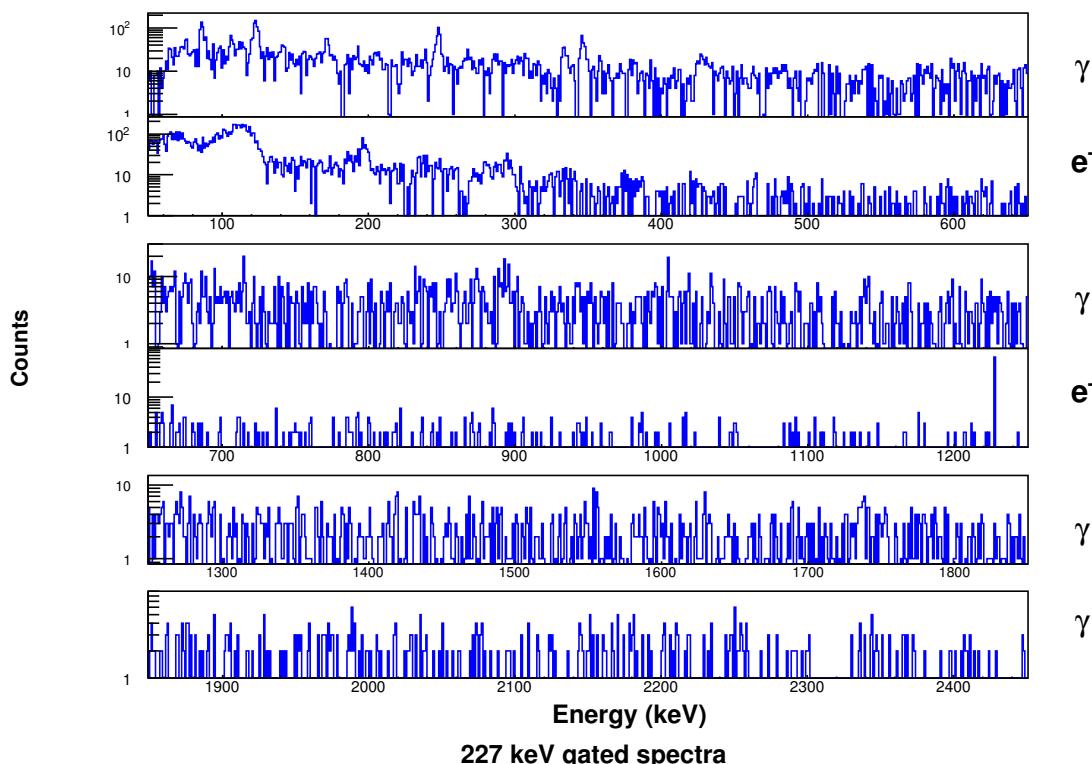


202 keV gated spectra

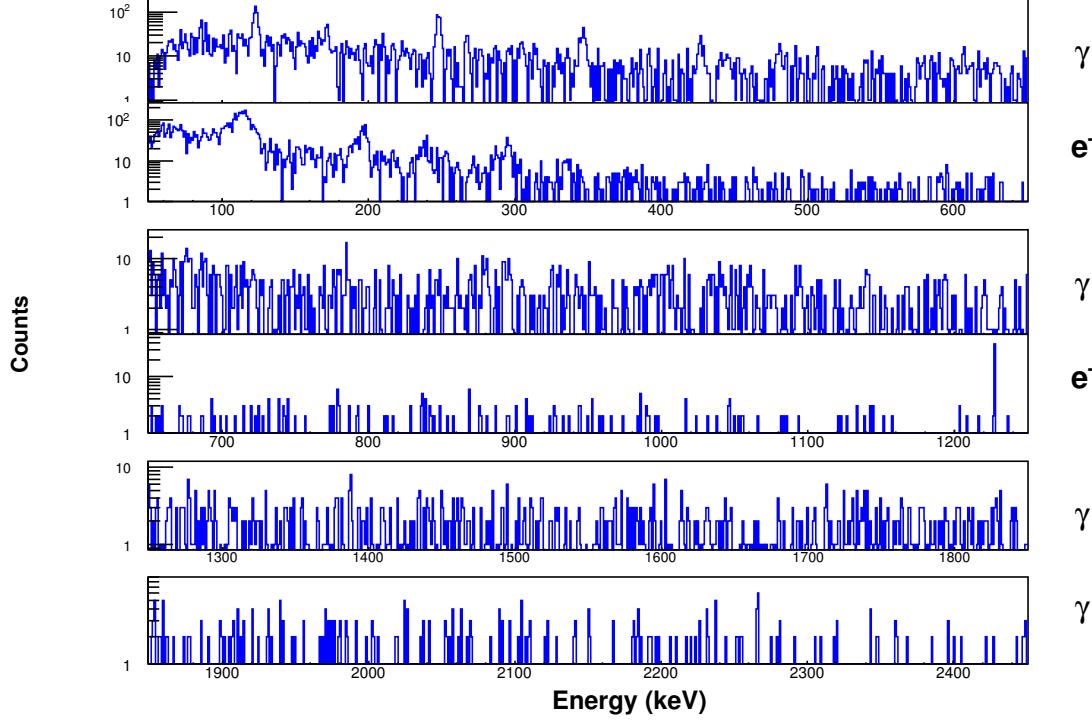




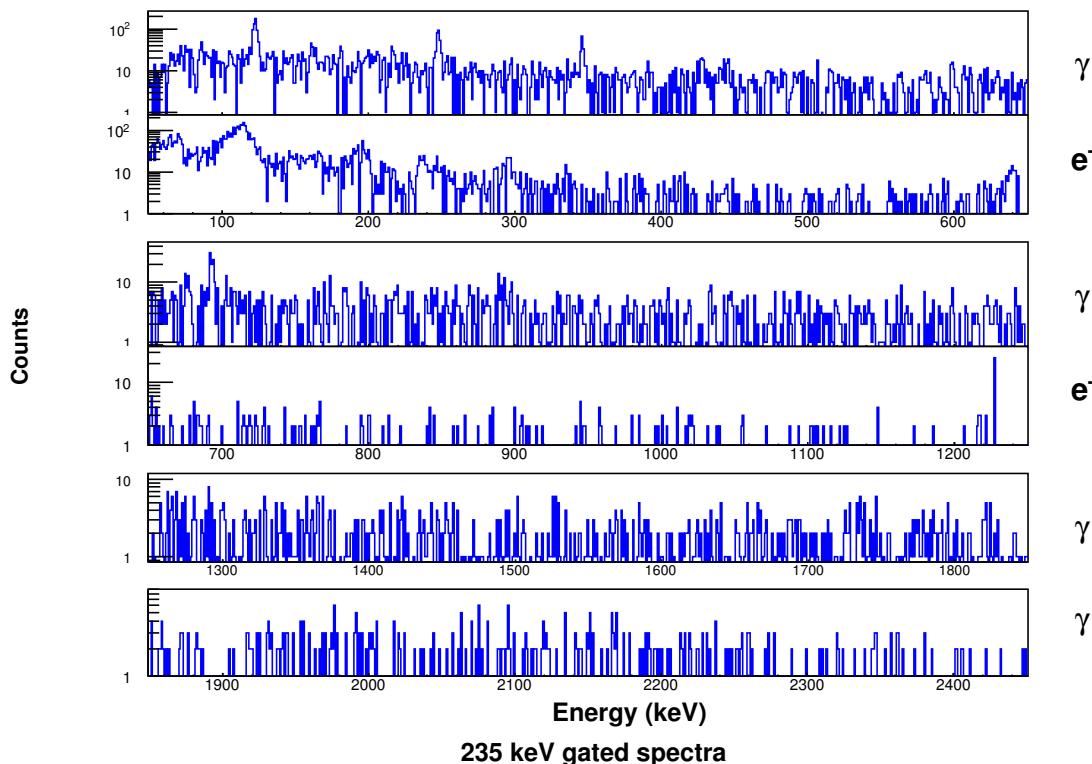
224 keV gated spectra



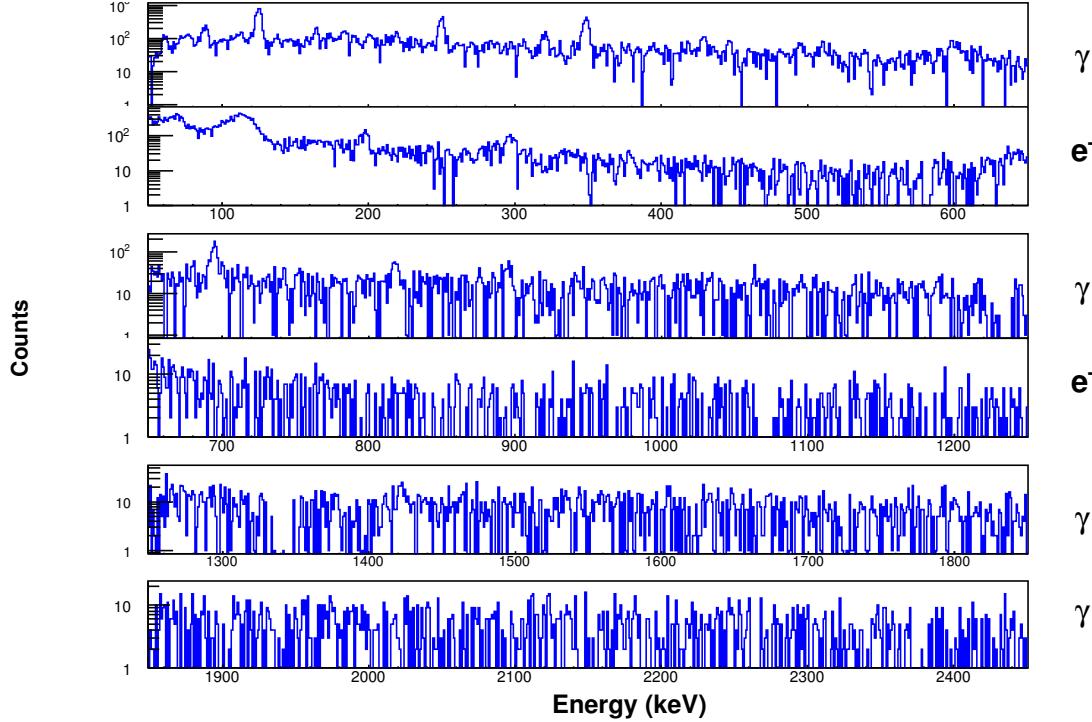
227 keV gated spectra



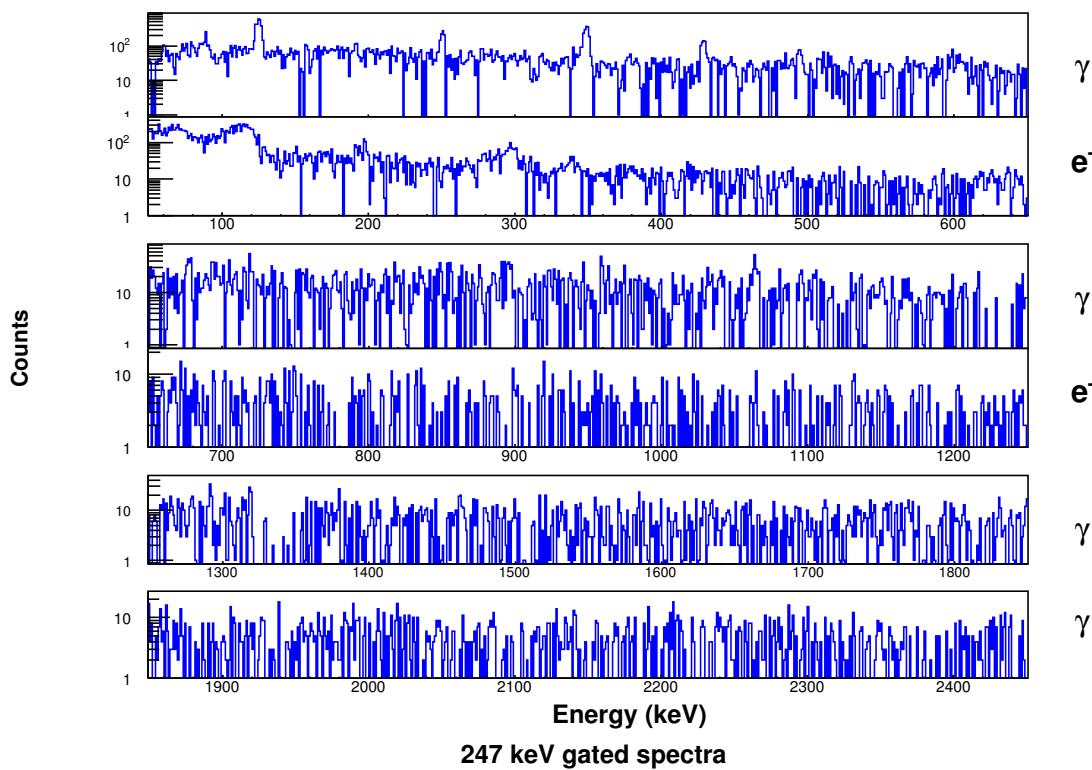
232 keV gated spectra



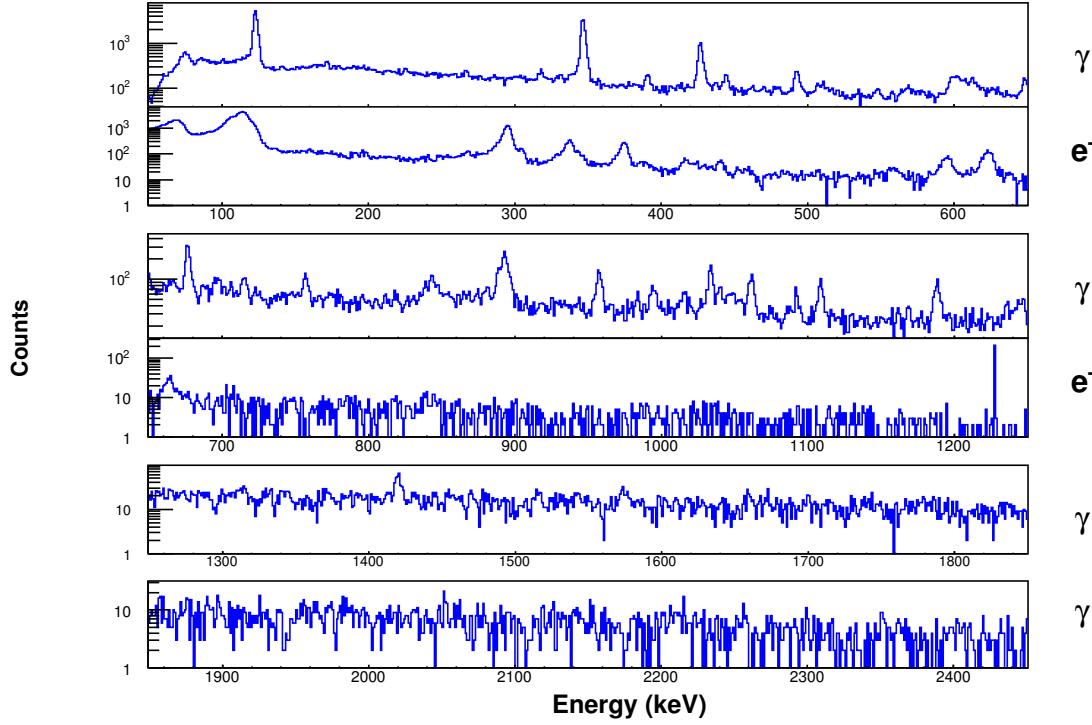
235 keV gated spectra



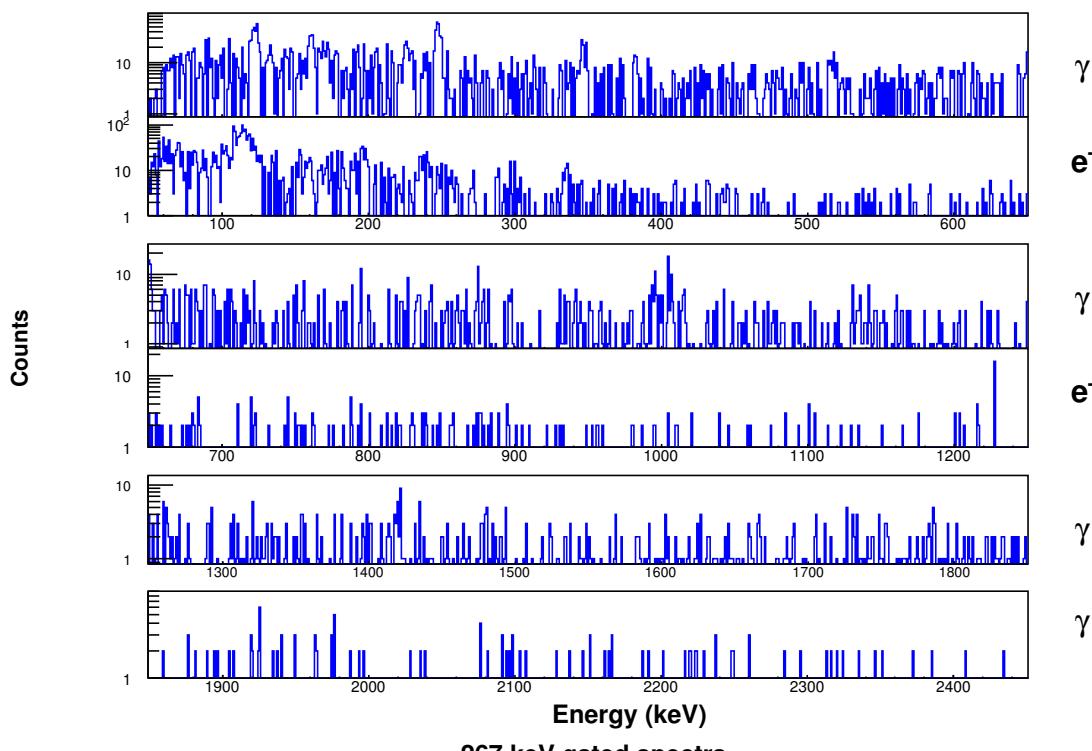
238 keV gated spectra



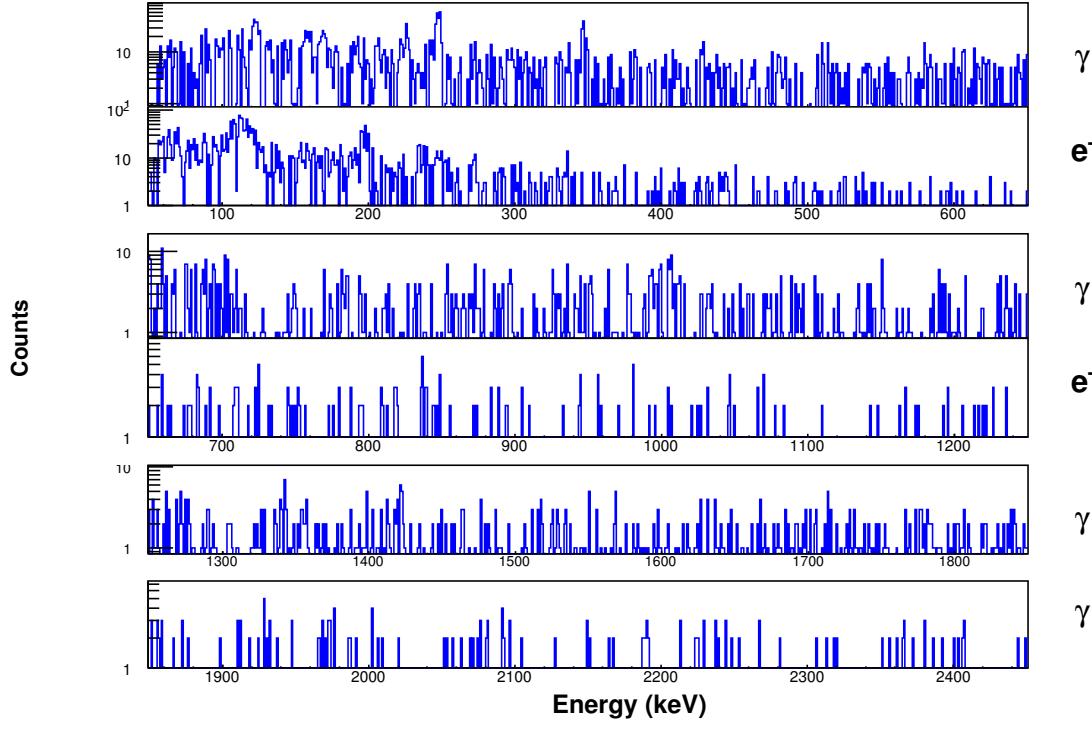
247 keV gated spectra



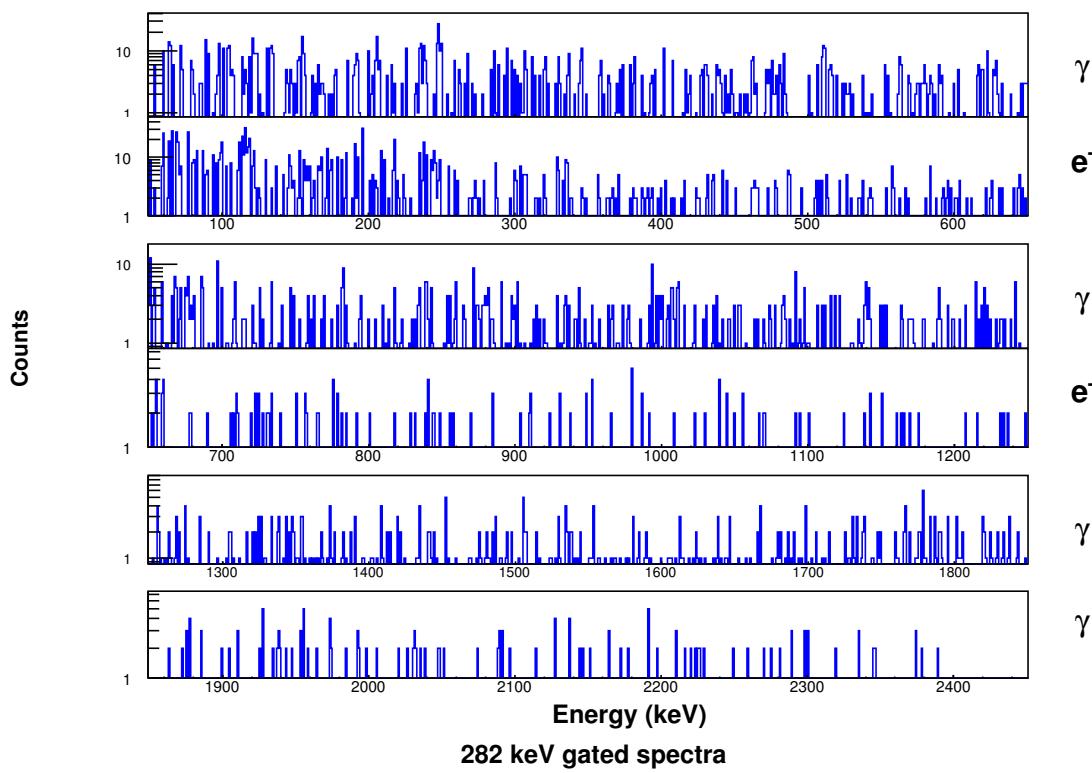
265 keV gated spectra



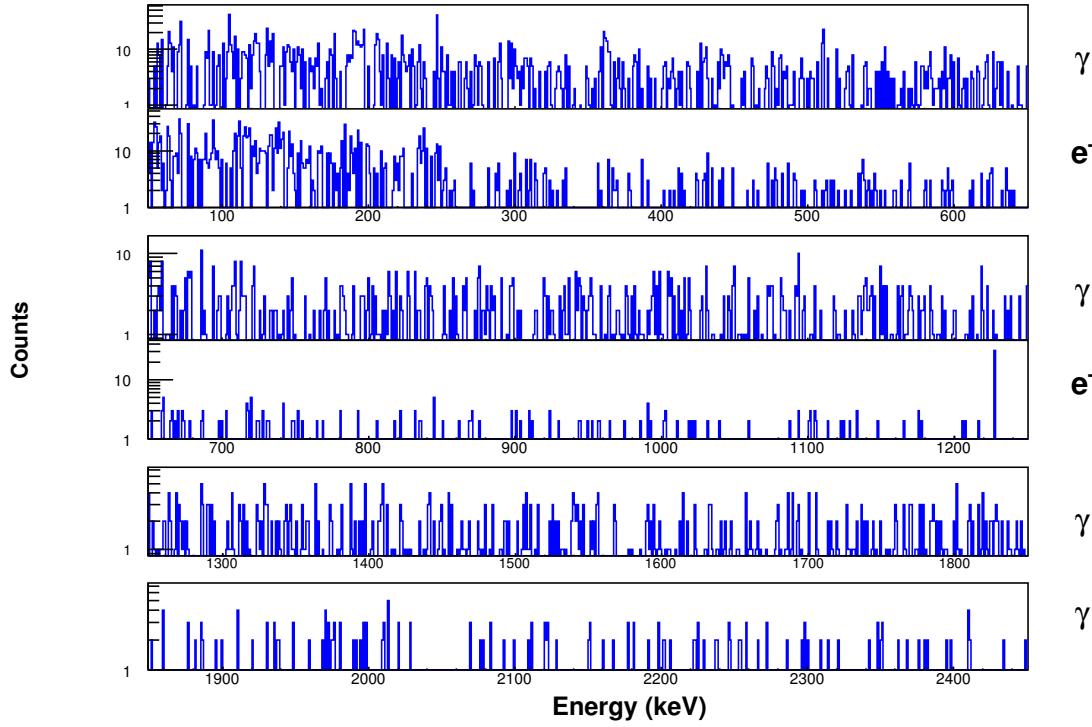
267 keV gated spectra



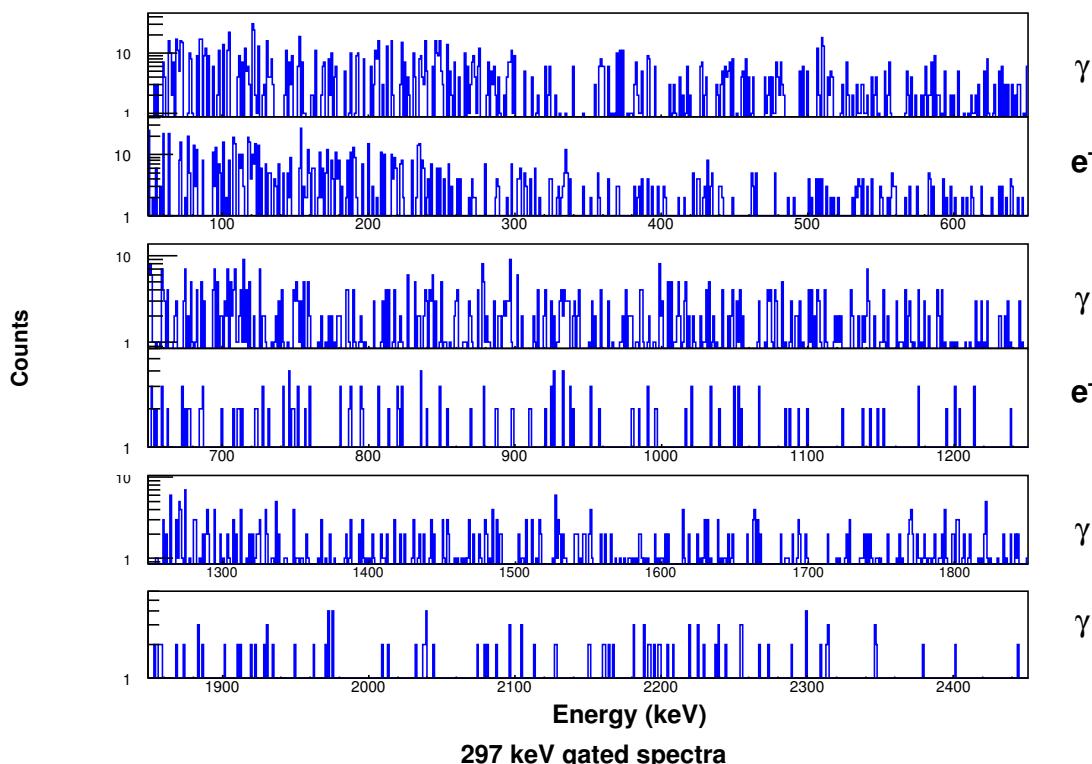
274 keV gated spectra



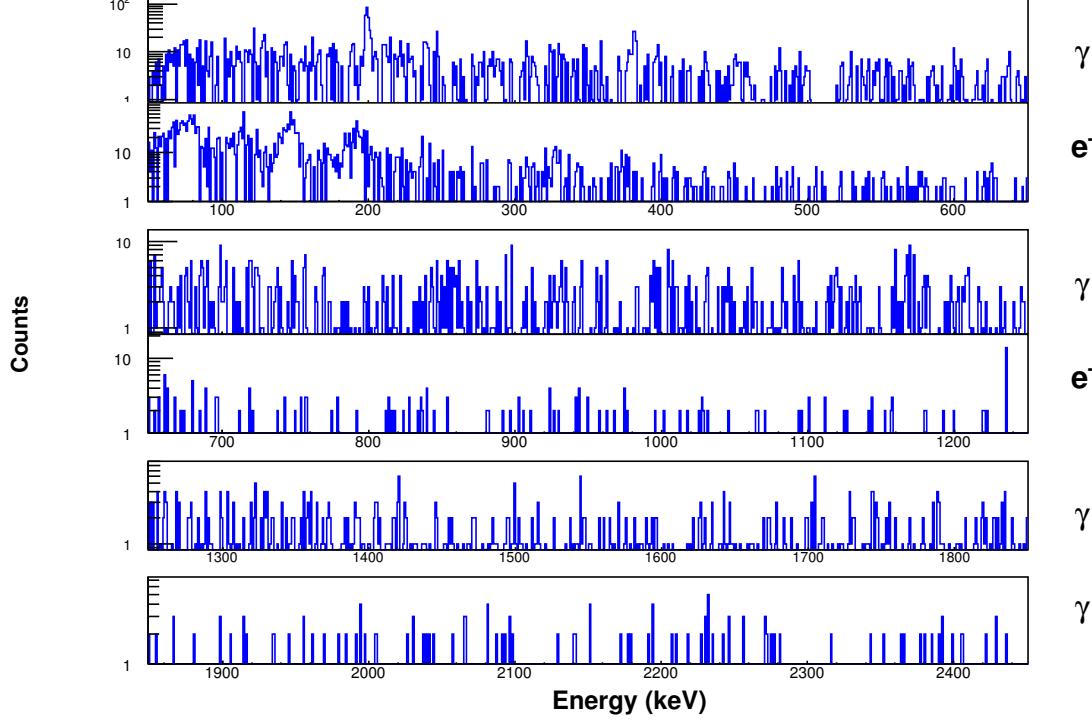
282 keV gated spectra



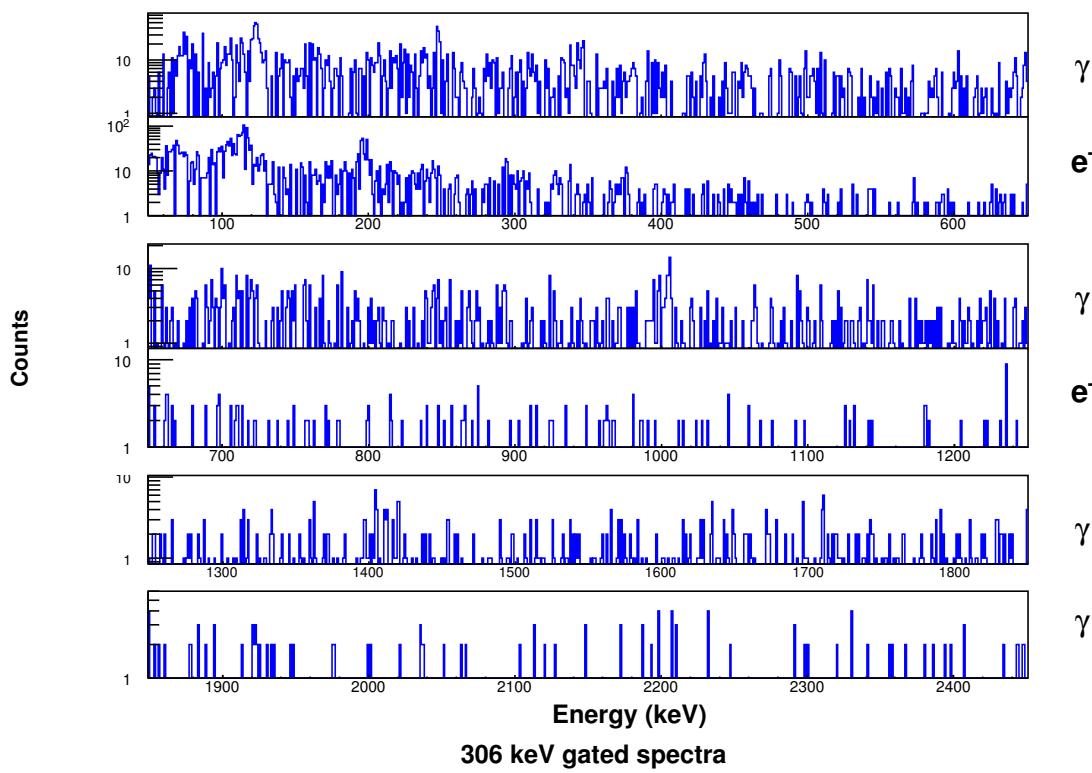
285 keV gated spectra



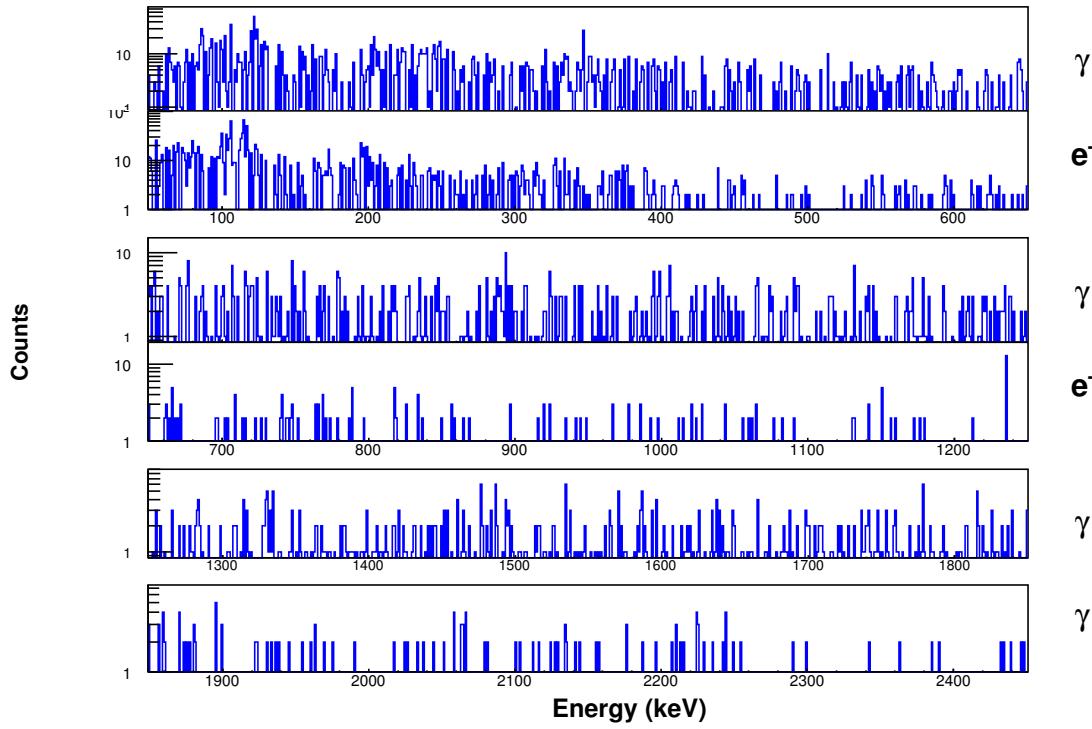
297 keV gated spectra



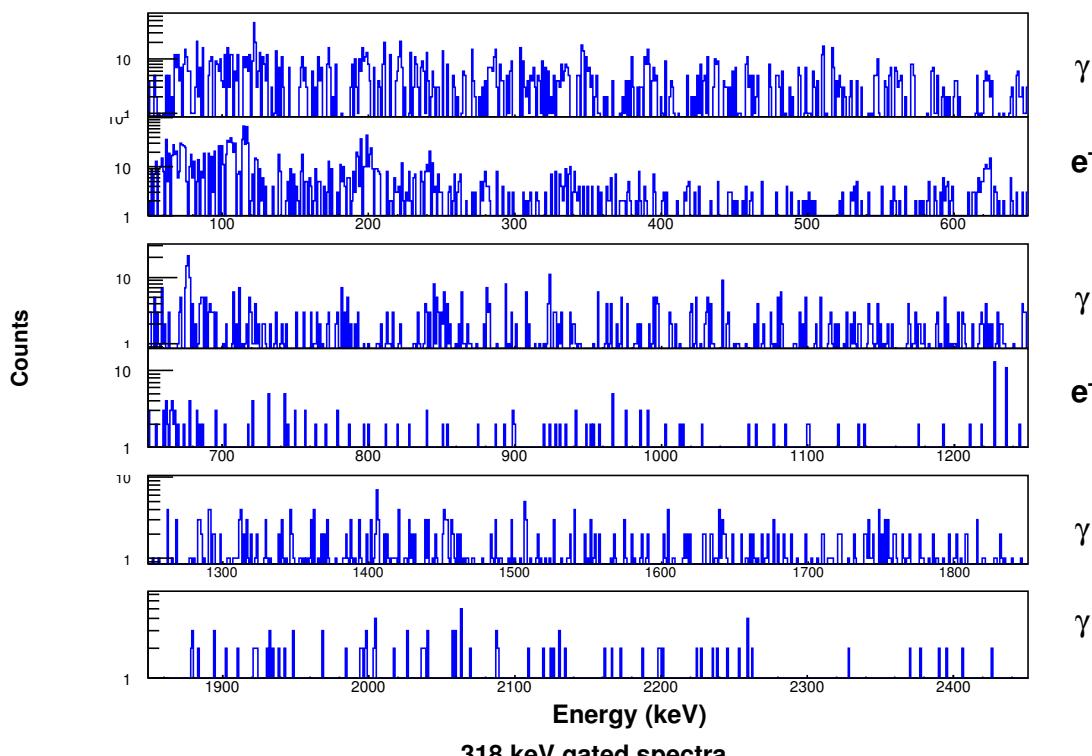
303 keV gated spectra



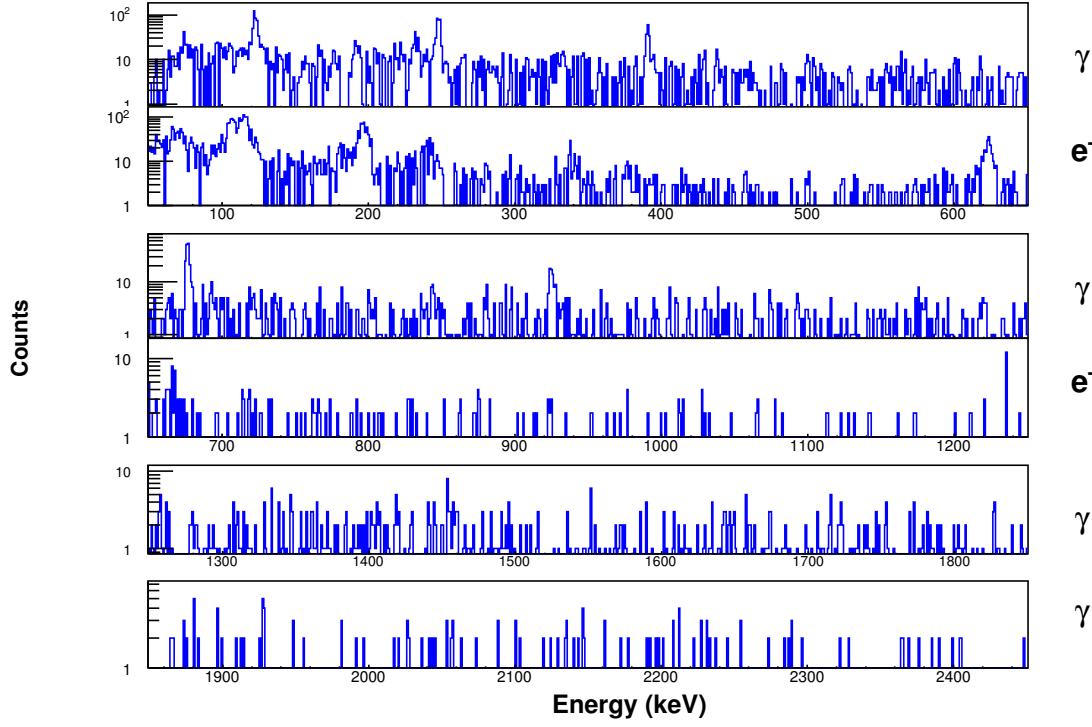
306 keV gated spectra



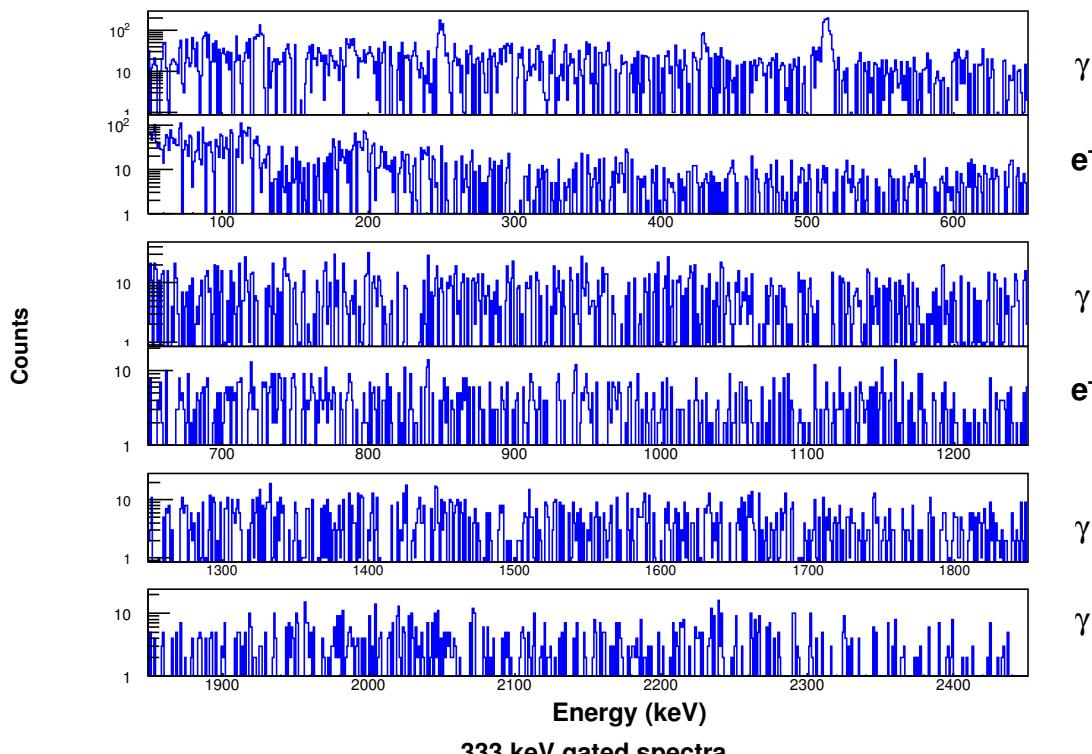
315 keV gated spectra



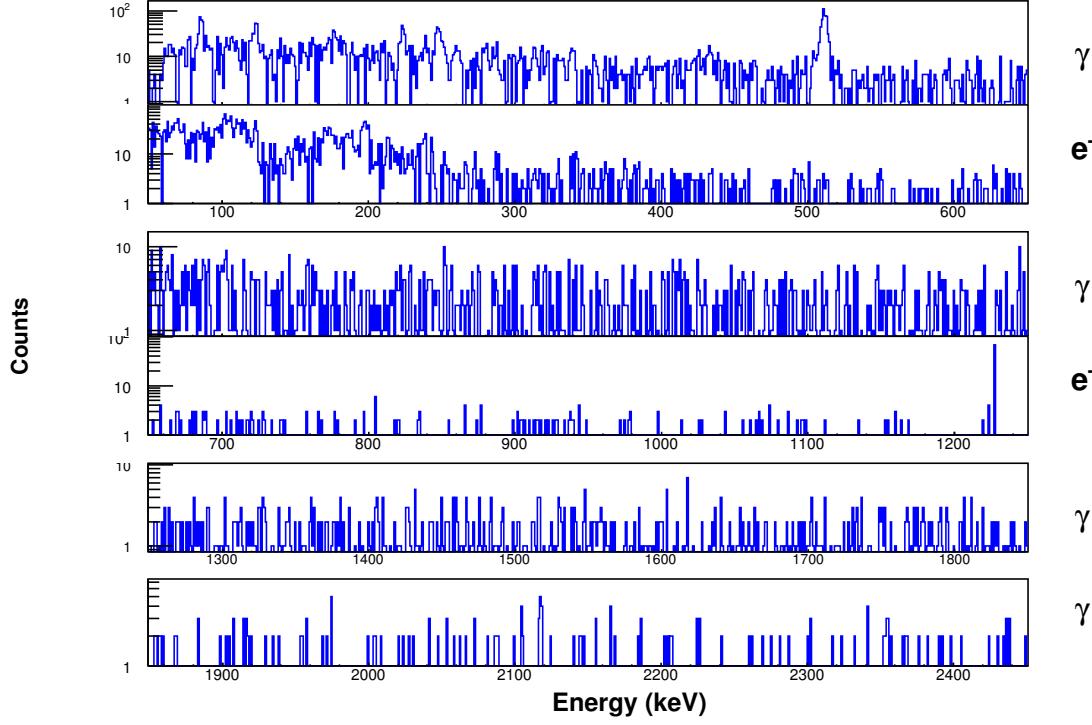
318 keV gated spectra



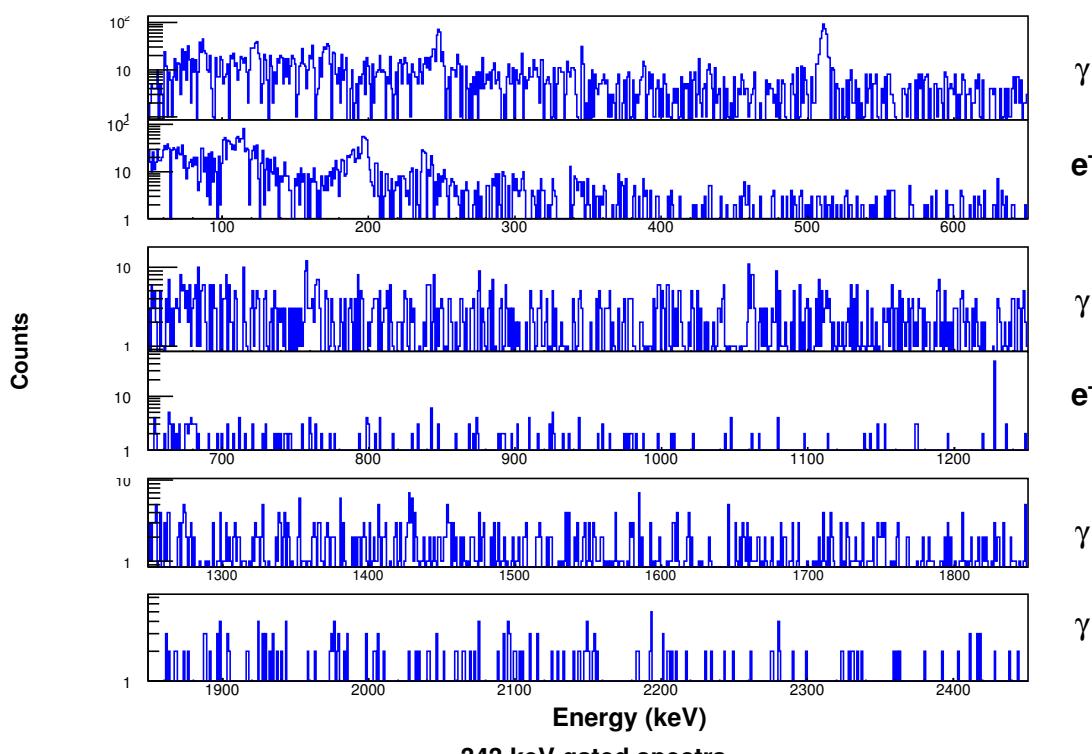
325 keV gated spectra



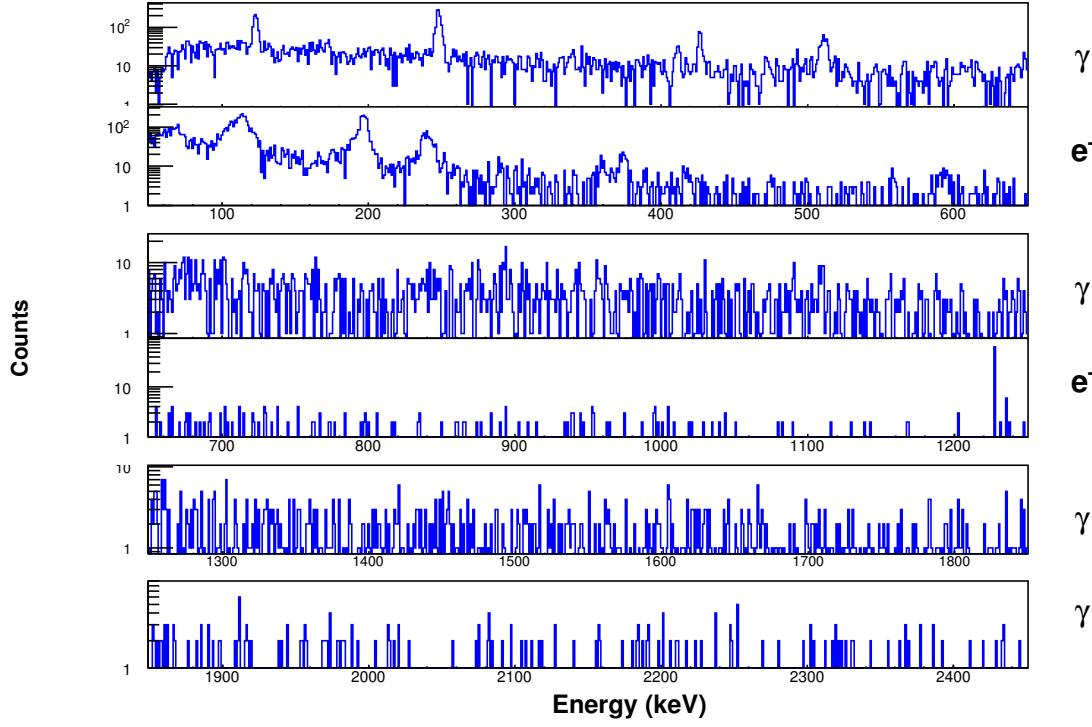
333 keV gated spectra



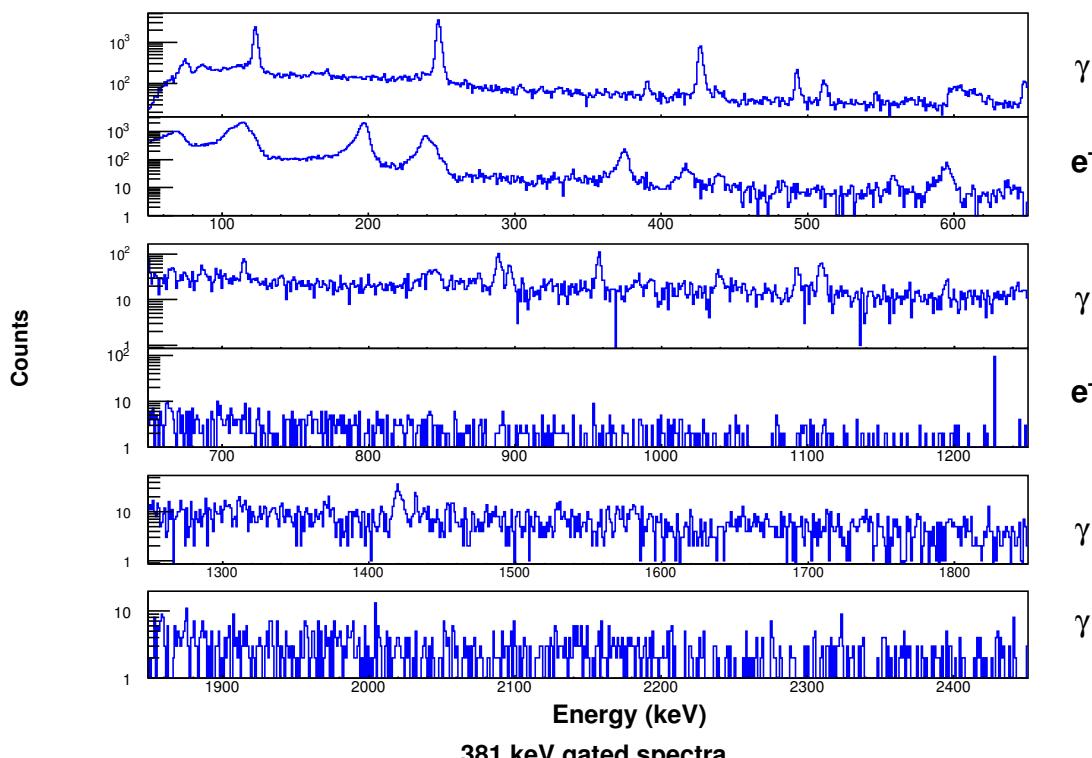
339 keV gated spectra



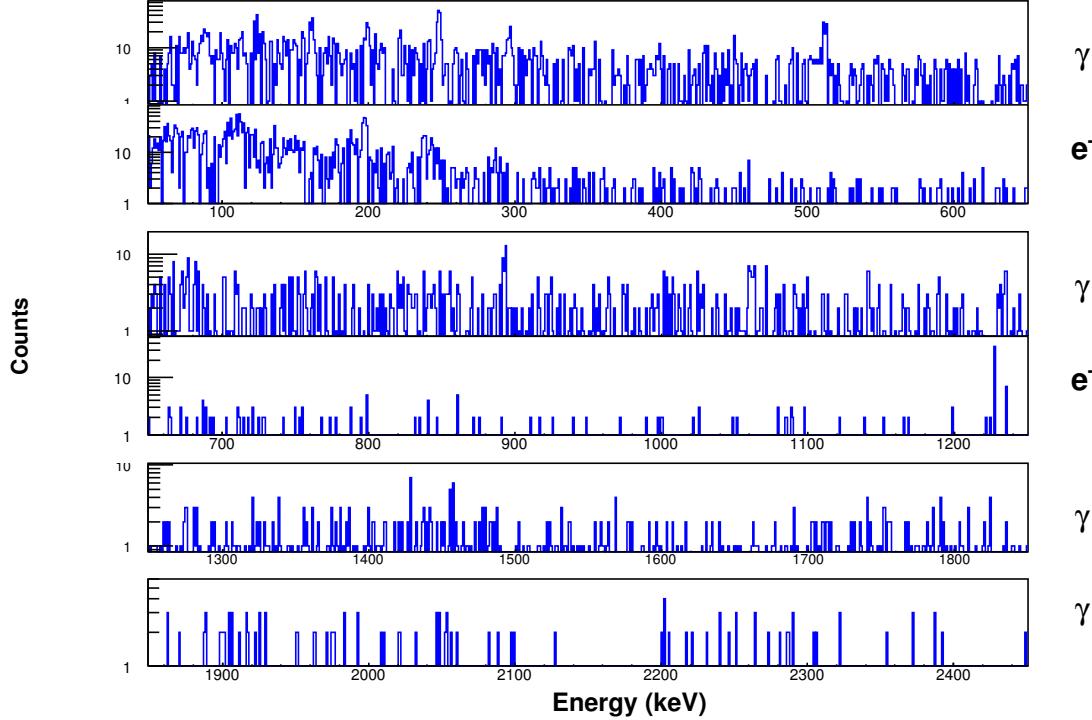
343 keV gated spectra



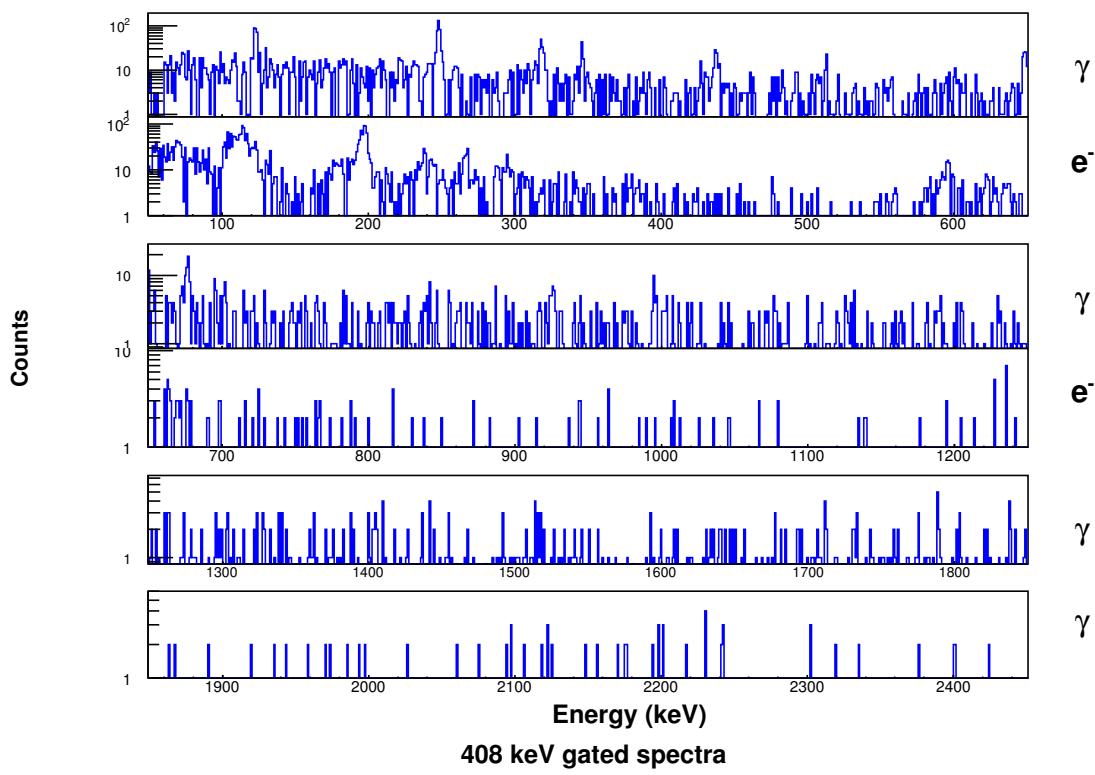
346 keV gated spectra



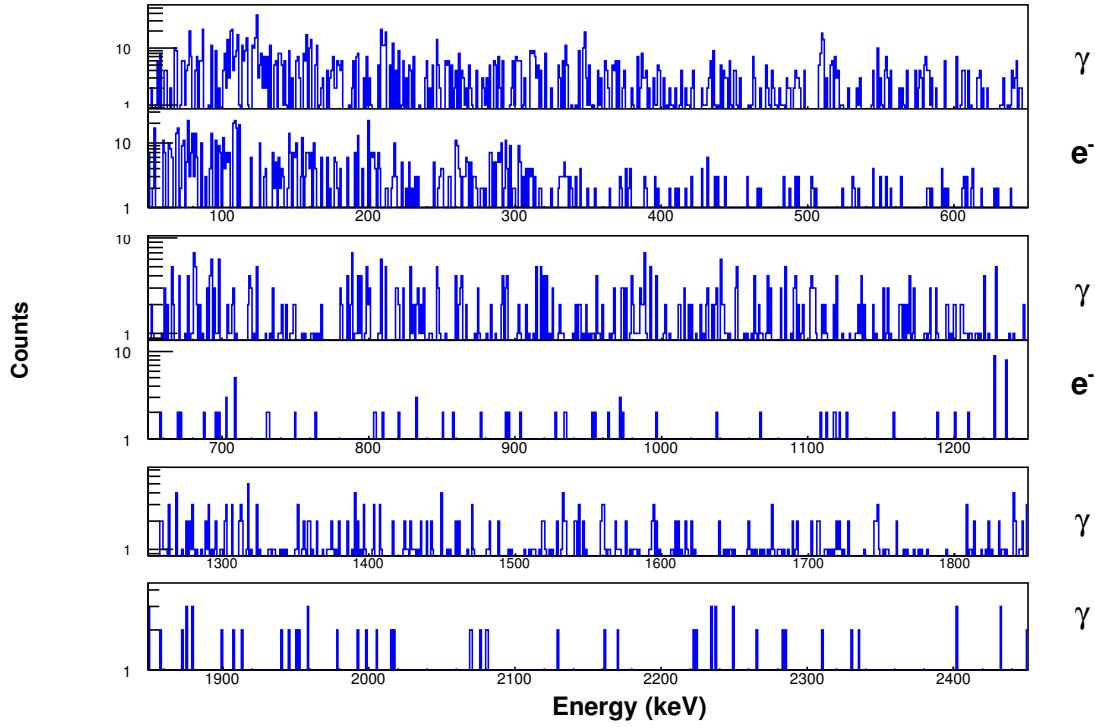
381 keV gated spectra



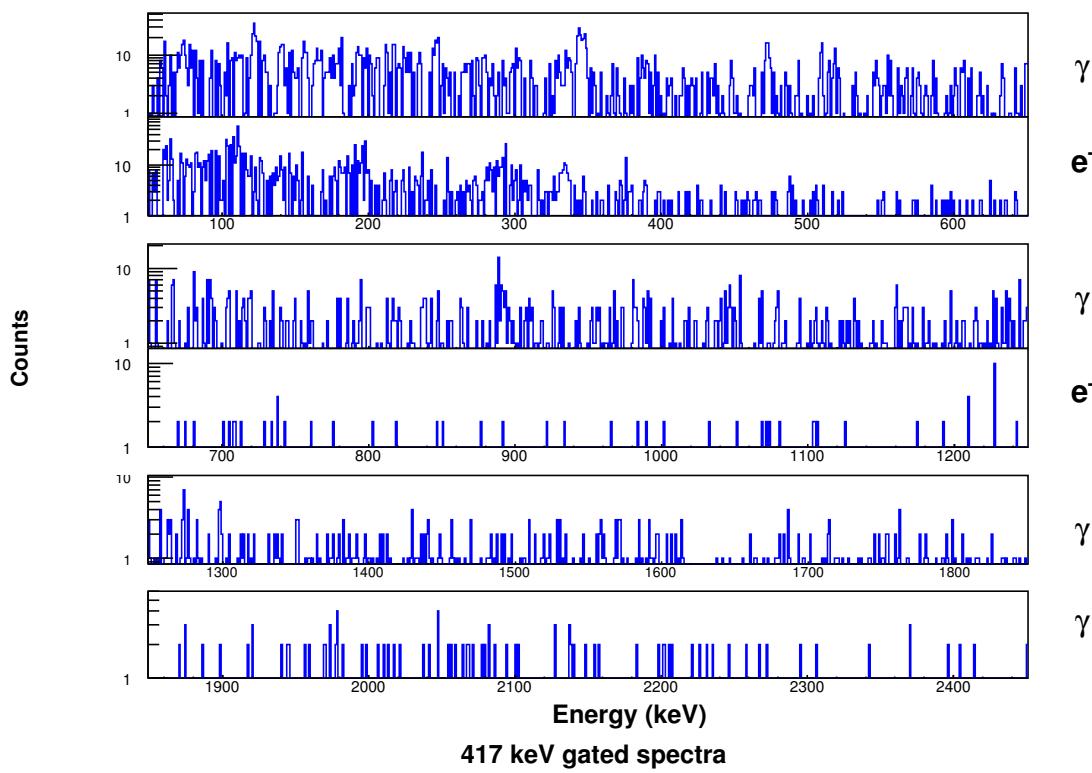
390 keV gated spectra



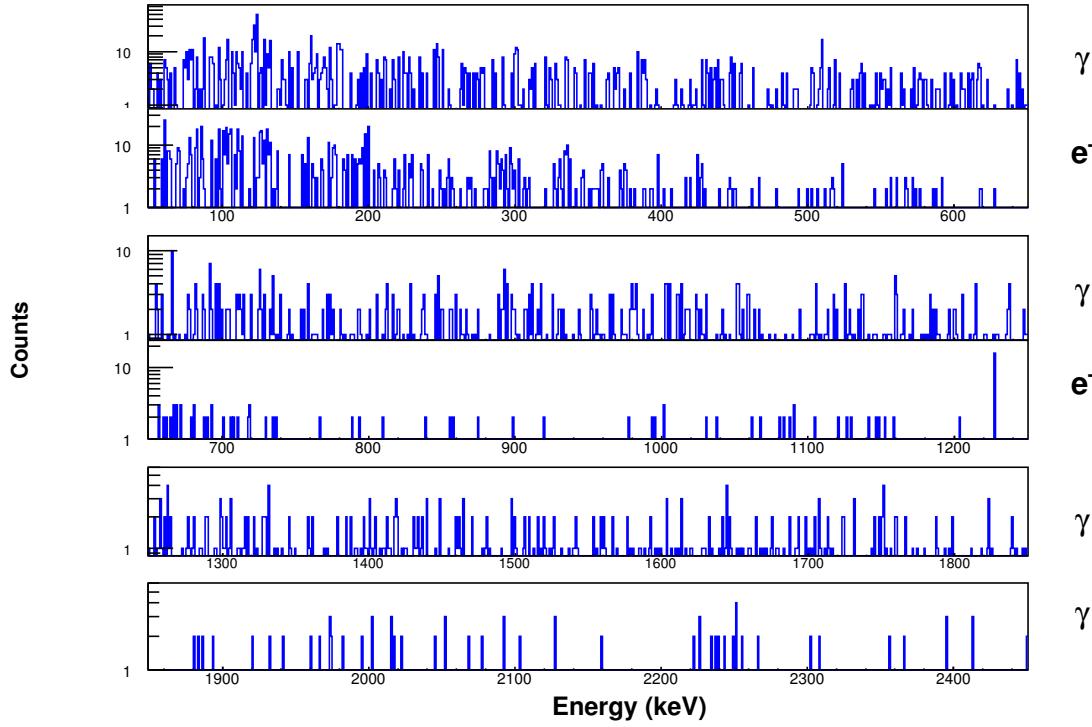
408 keV gated spectra



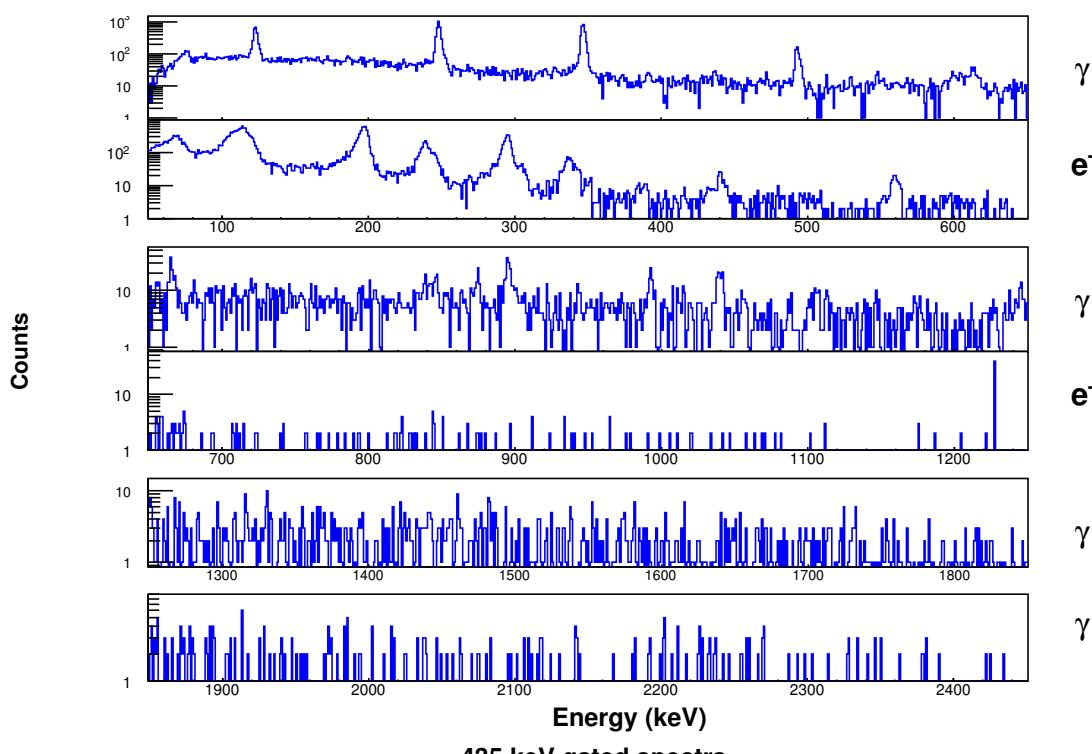
412 keV gated spectra



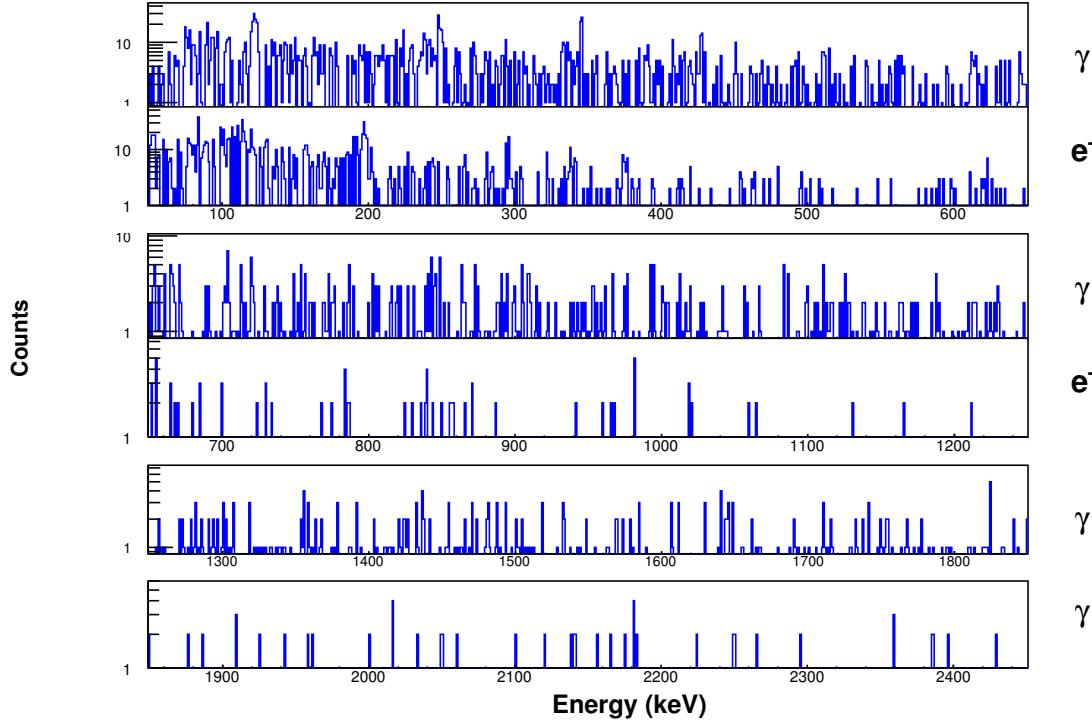
417 keV gated spectra



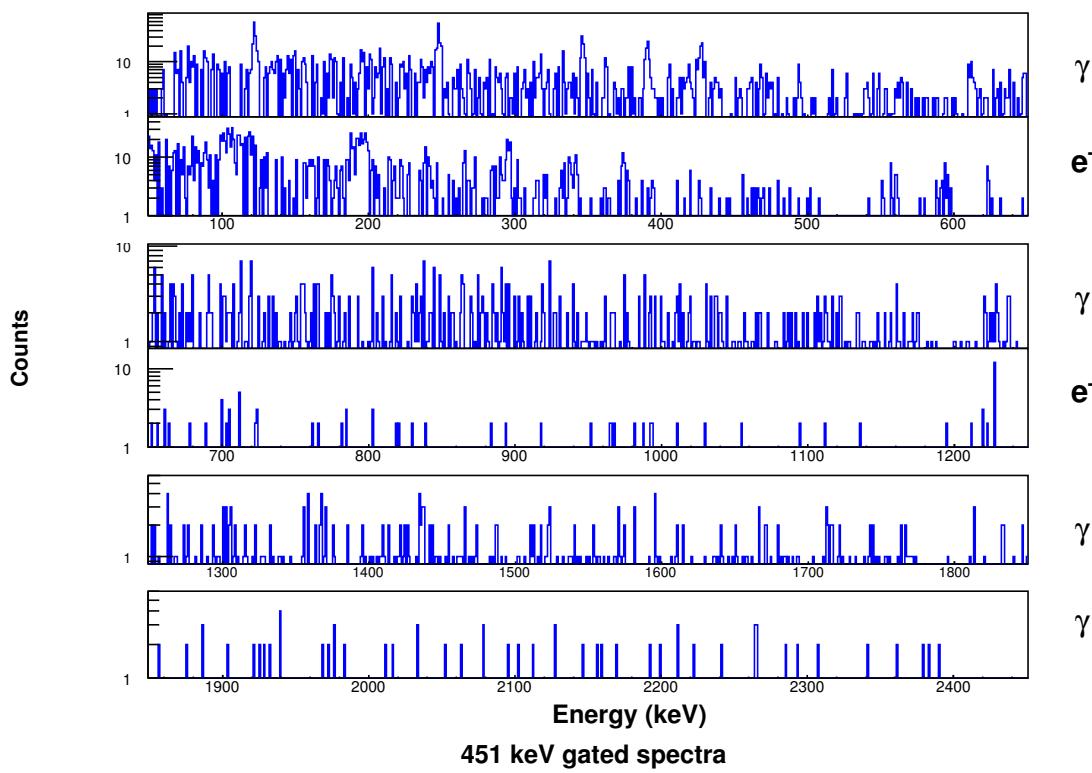
426 keV gated spectra



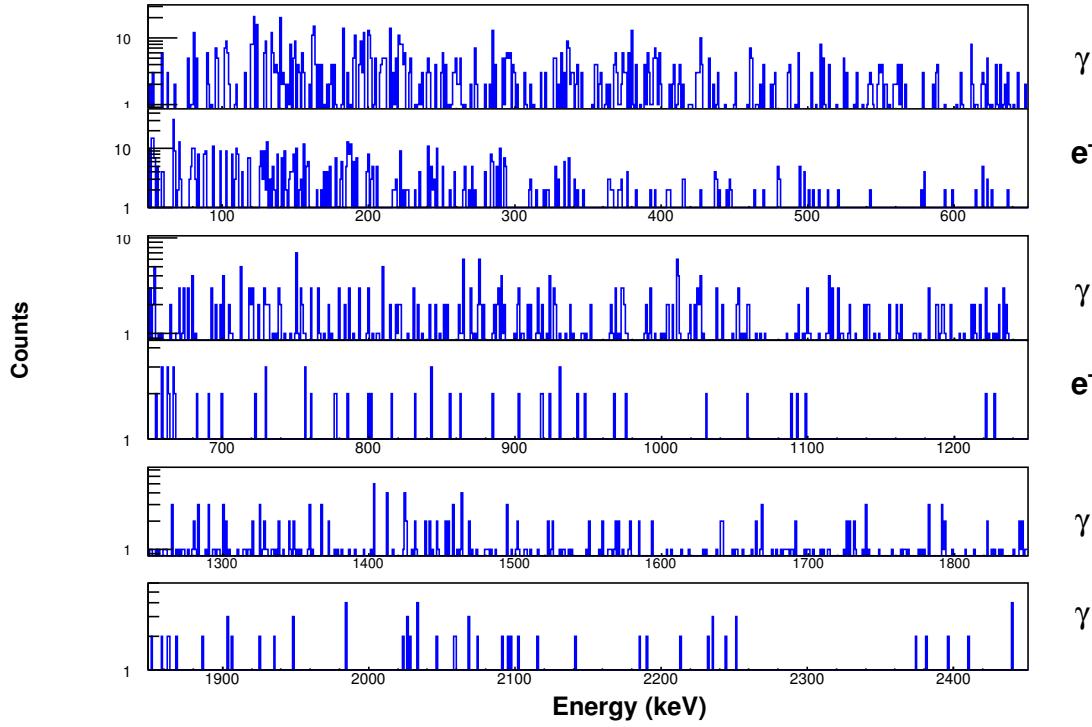
435 keV gated spectra



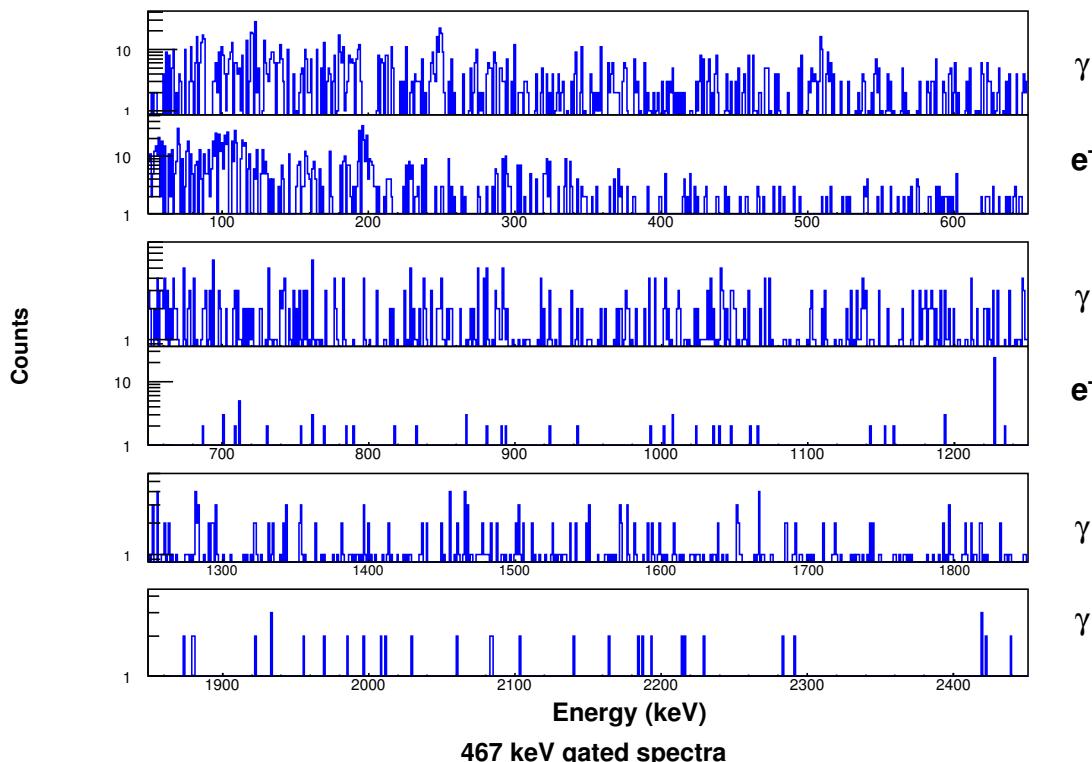
437 keV gated spectra



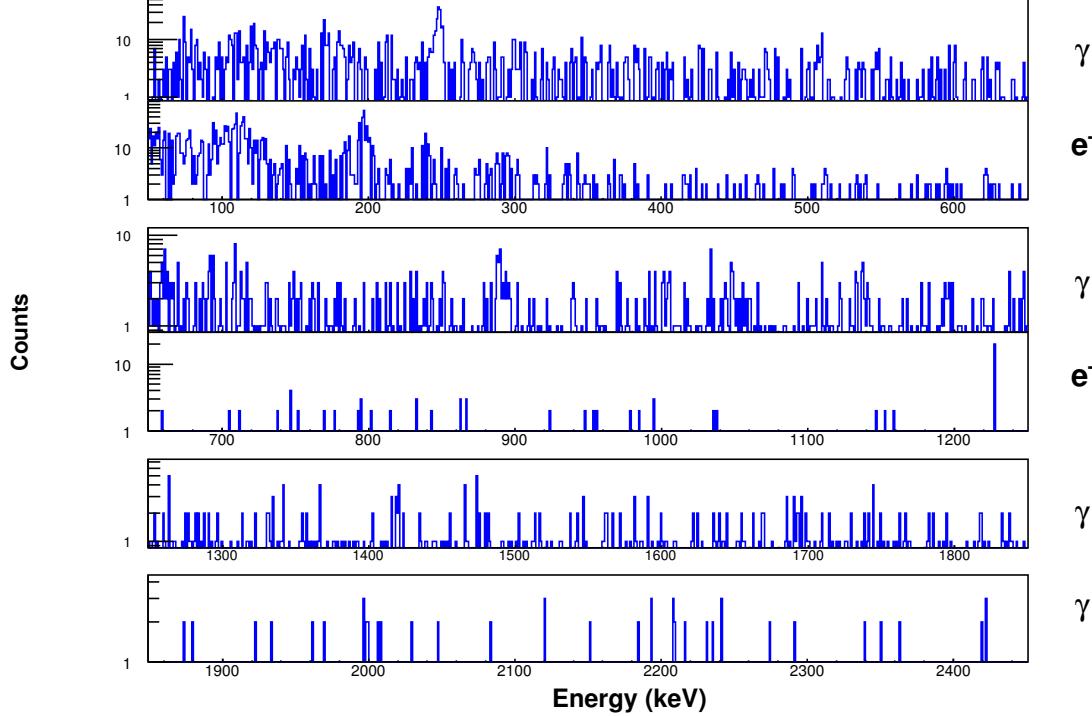
451 keV gated spectra



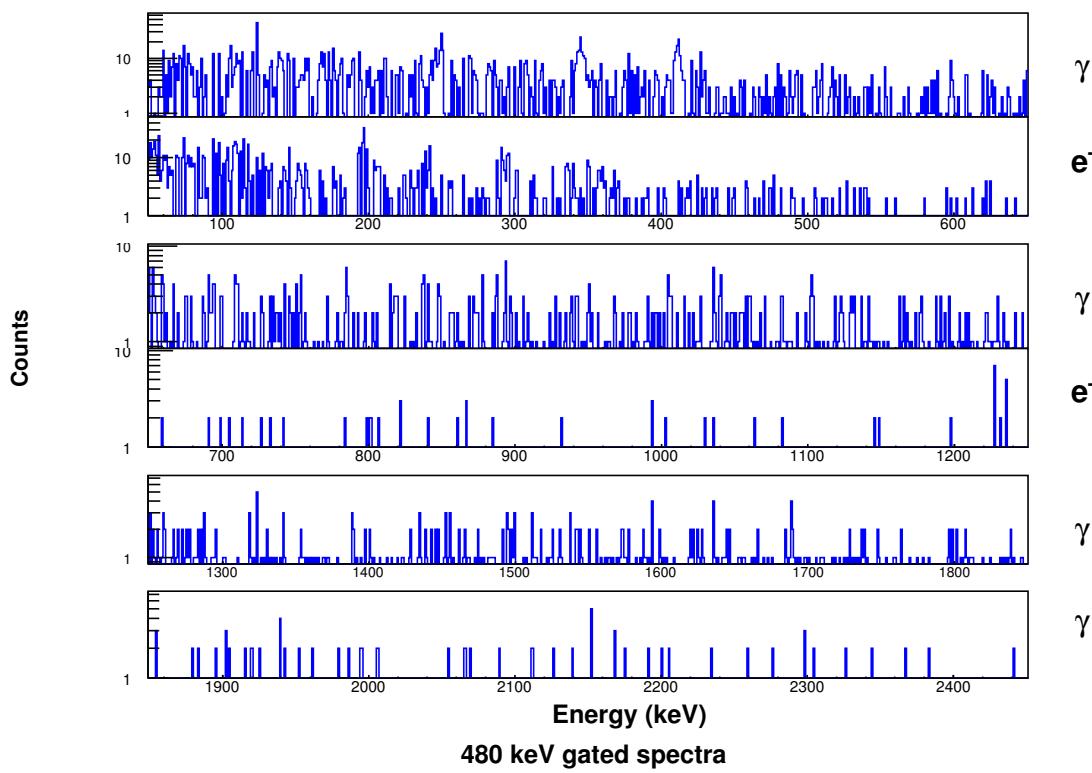
465 keV gated spectra



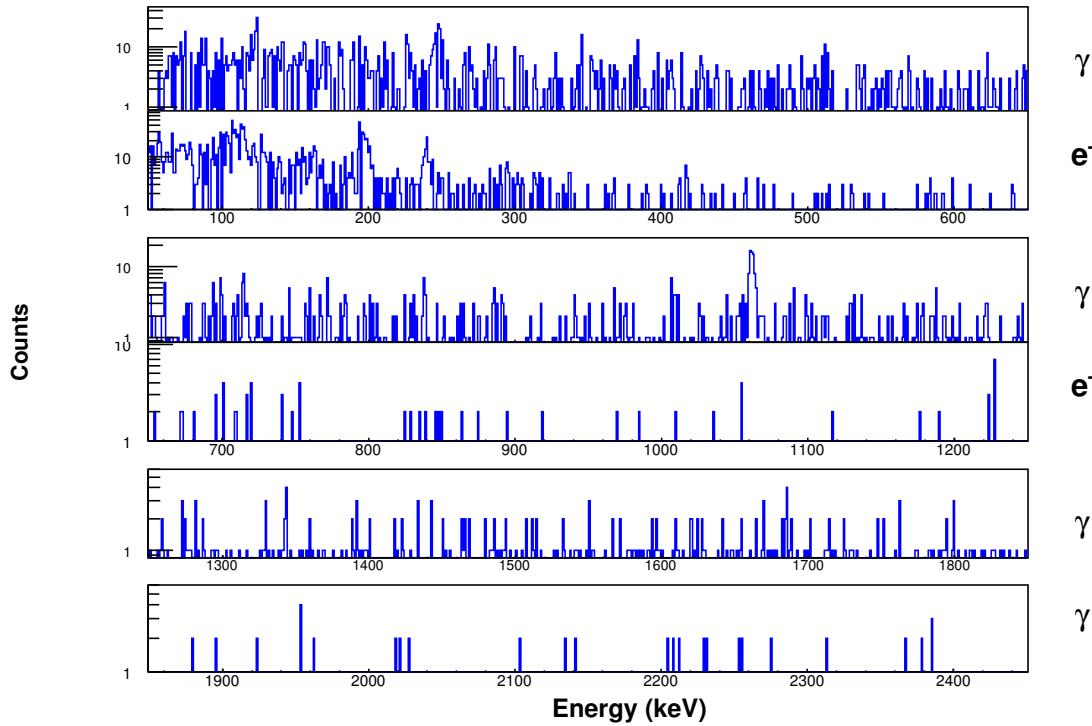
467 keV gated spectra



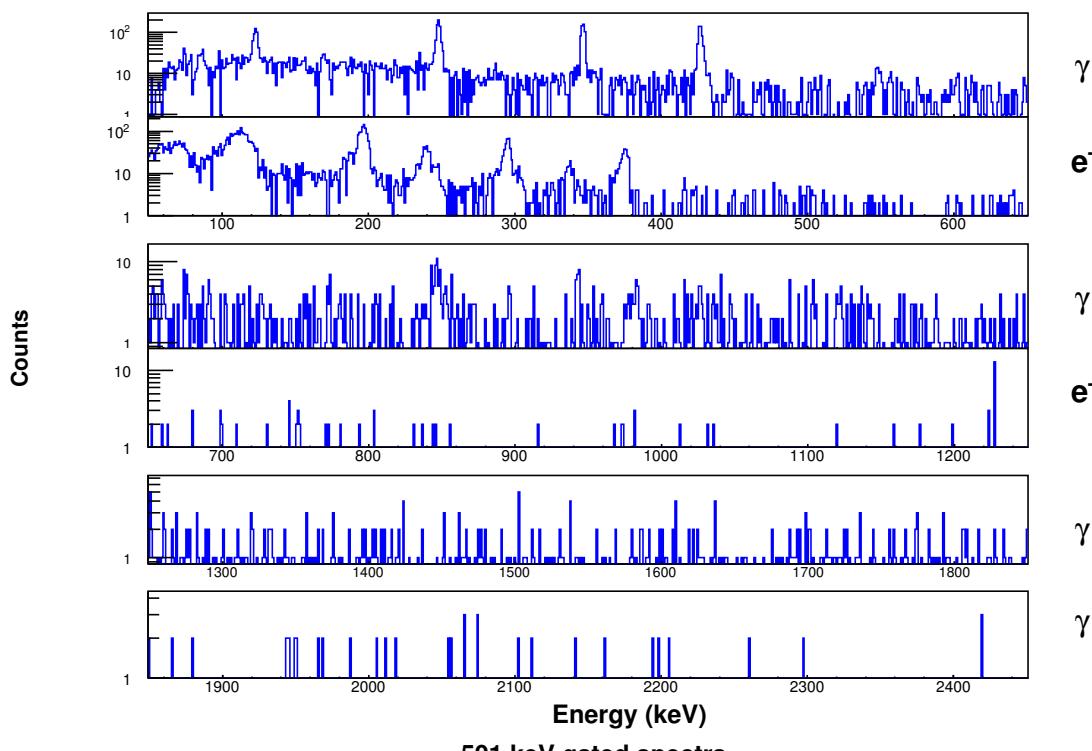
471 keV gated spectra



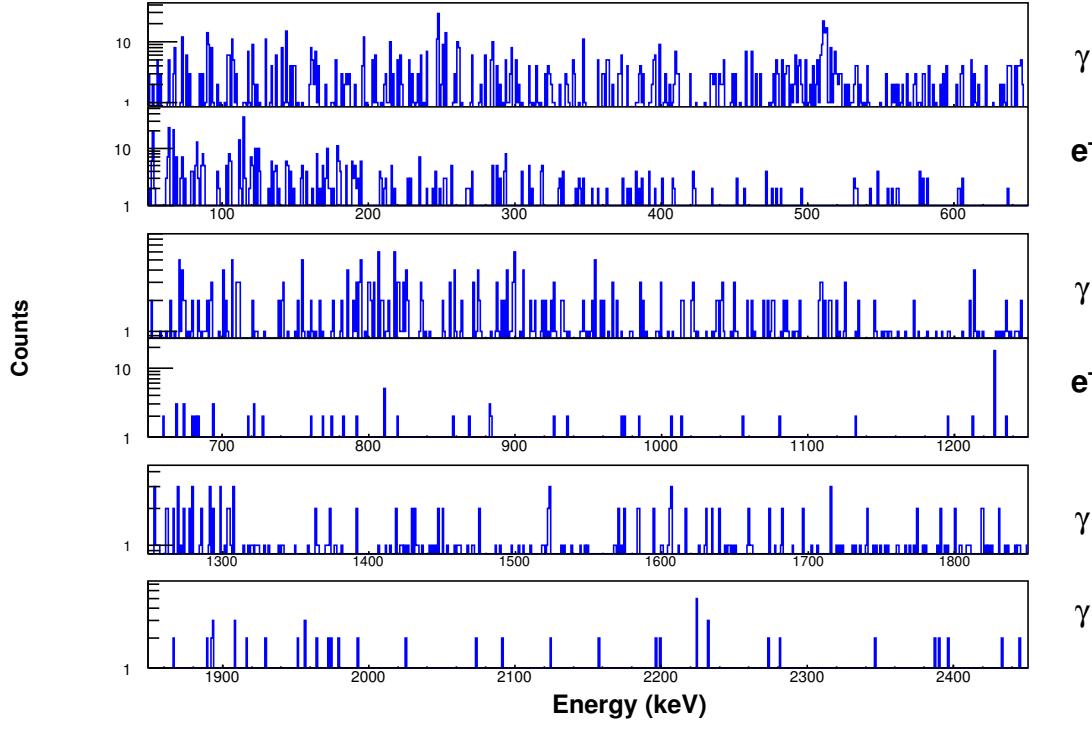
480 keV gated spectra



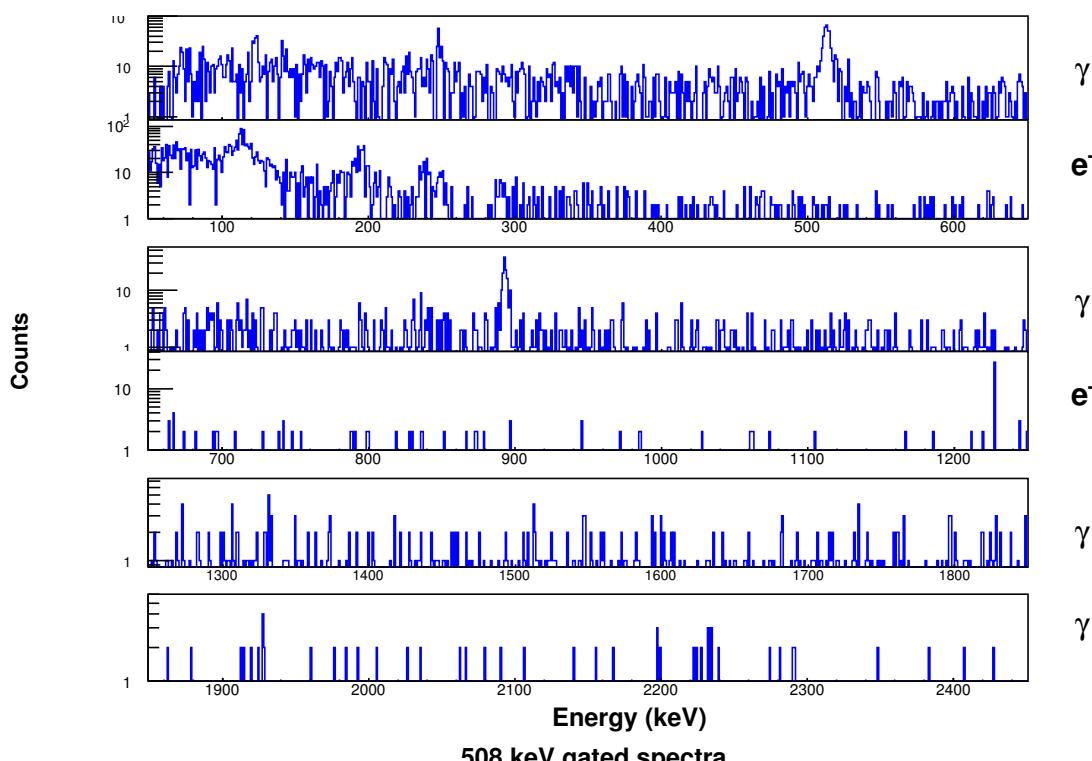
492 keV gated spectra



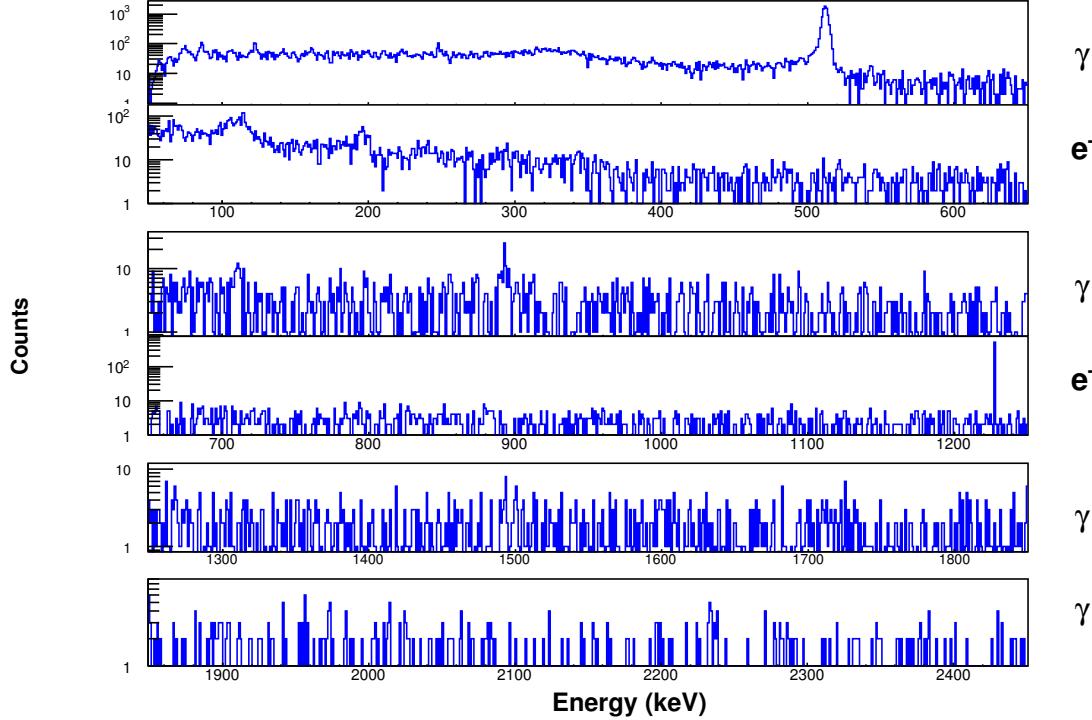
501 keV gated spectra



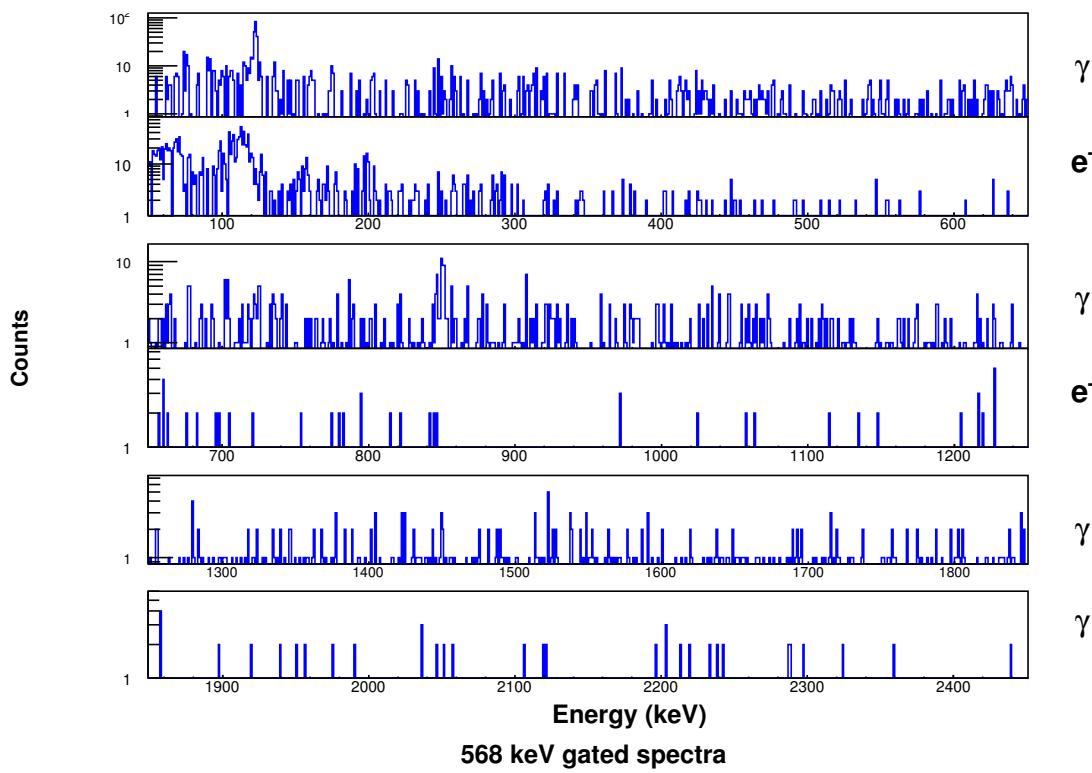
505 keV gated spectra



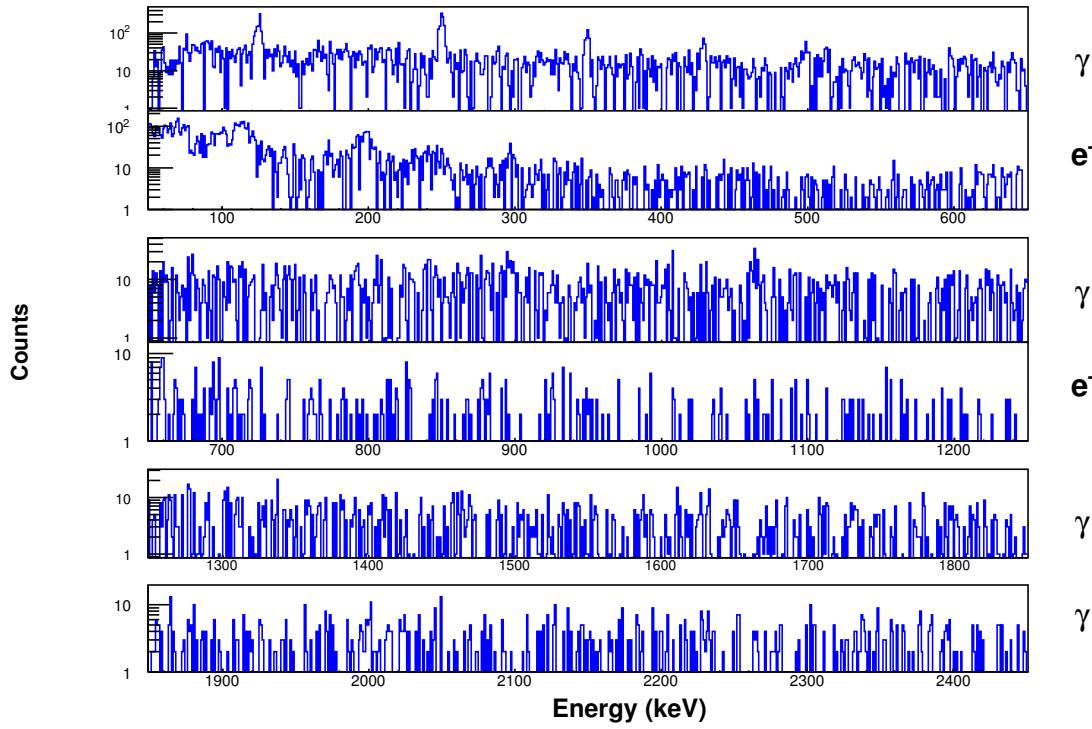
508 keV gated spectra



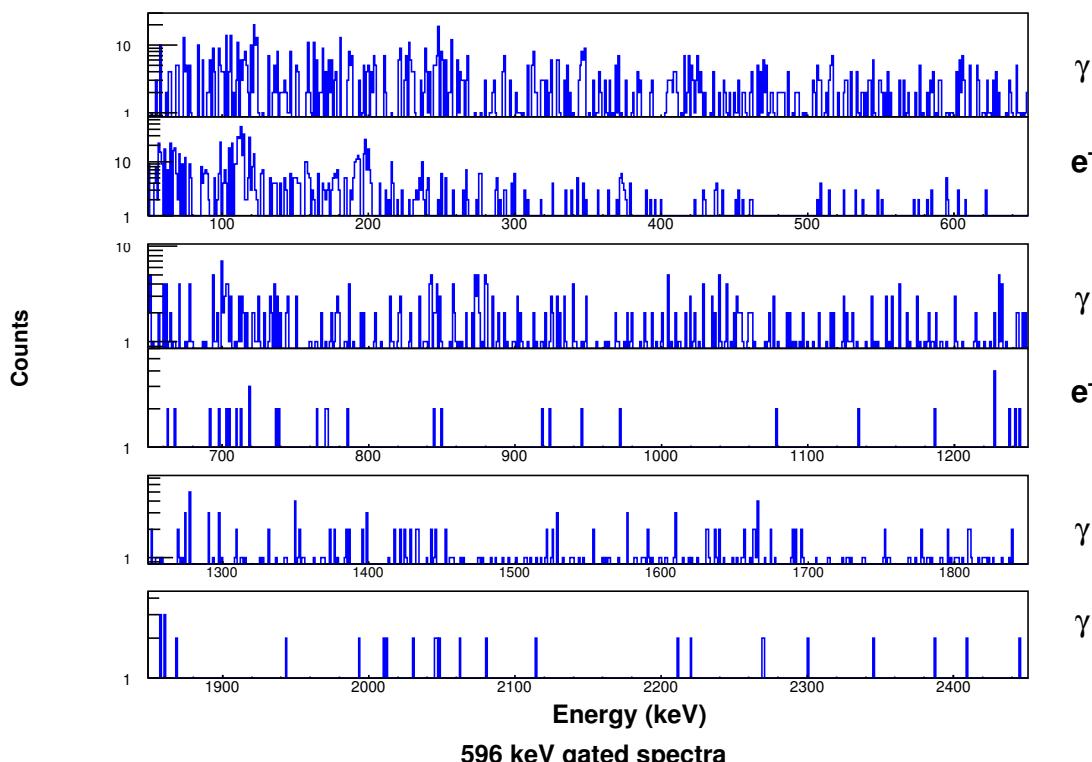
557 keV gated spectra



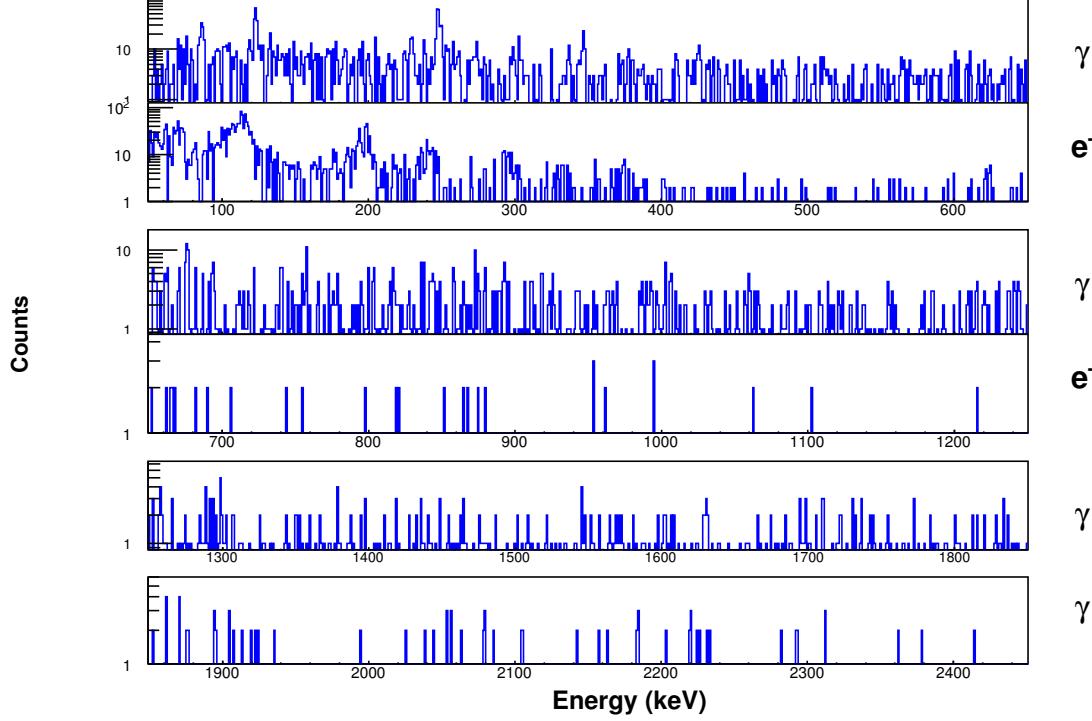
568 keV gated spectra



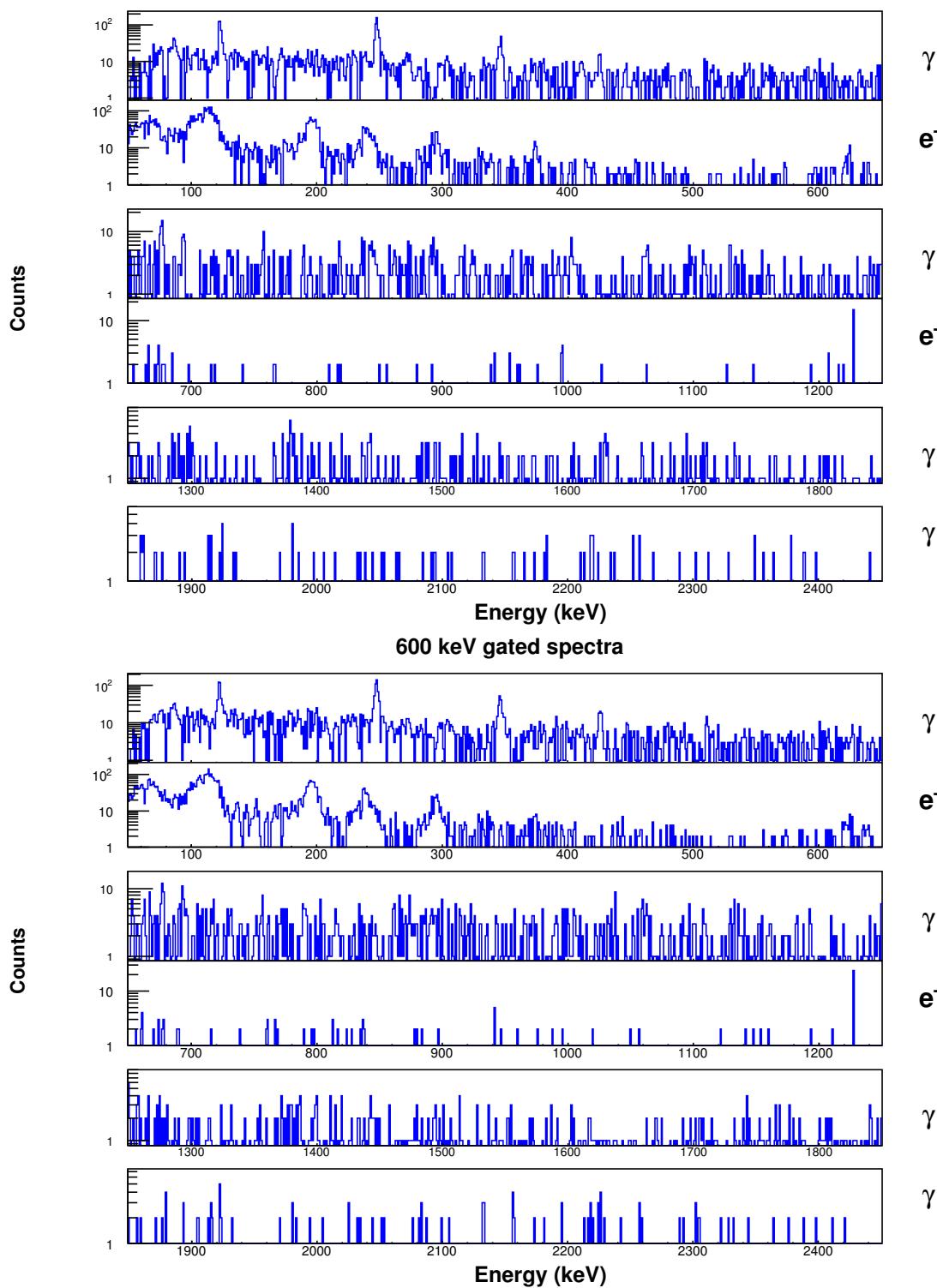
577 keV gated spectra



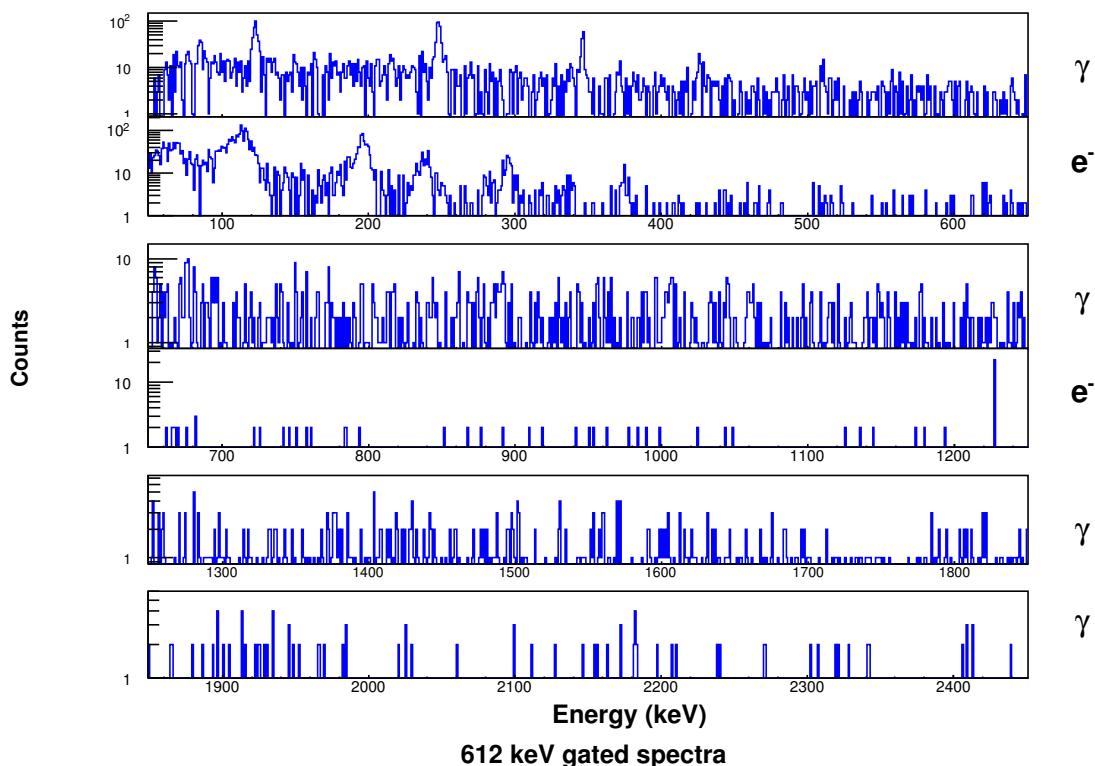
596 keV gated spectra



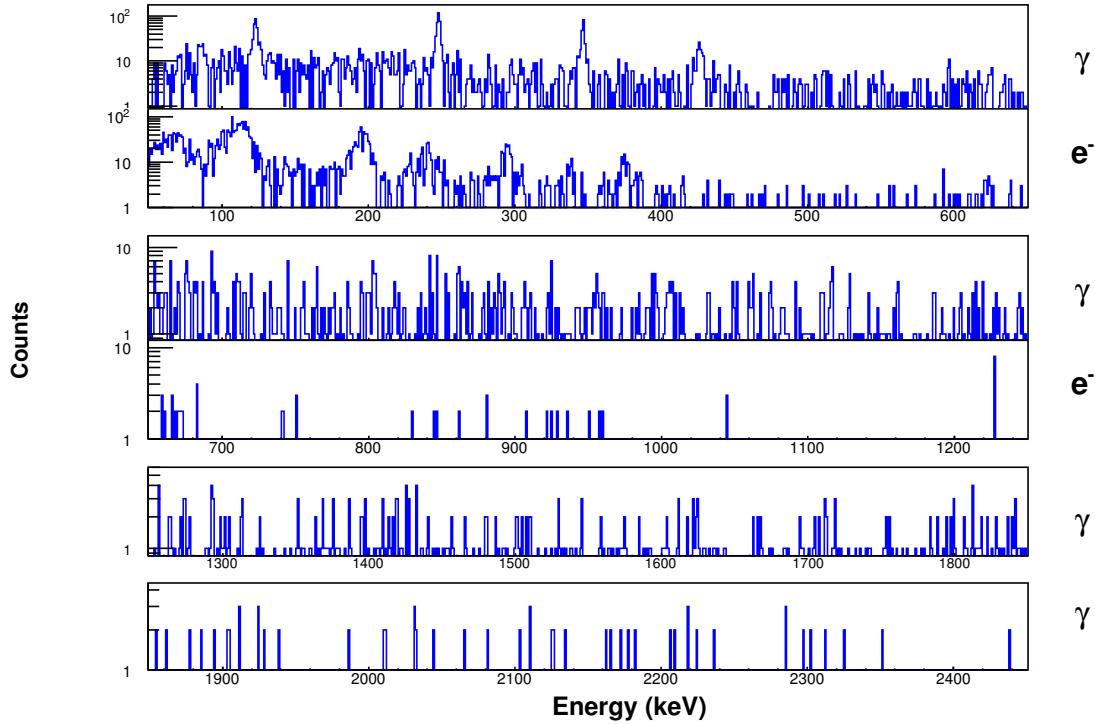
598 keV gated spectra



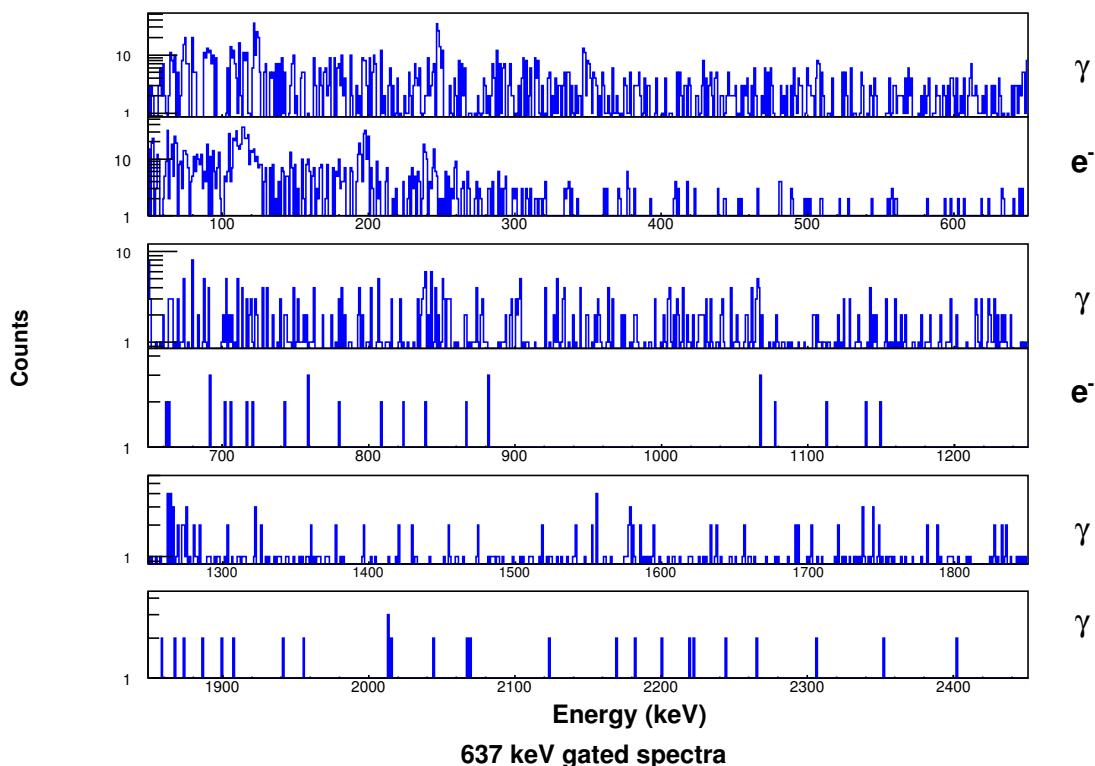
604 keV gated spectra



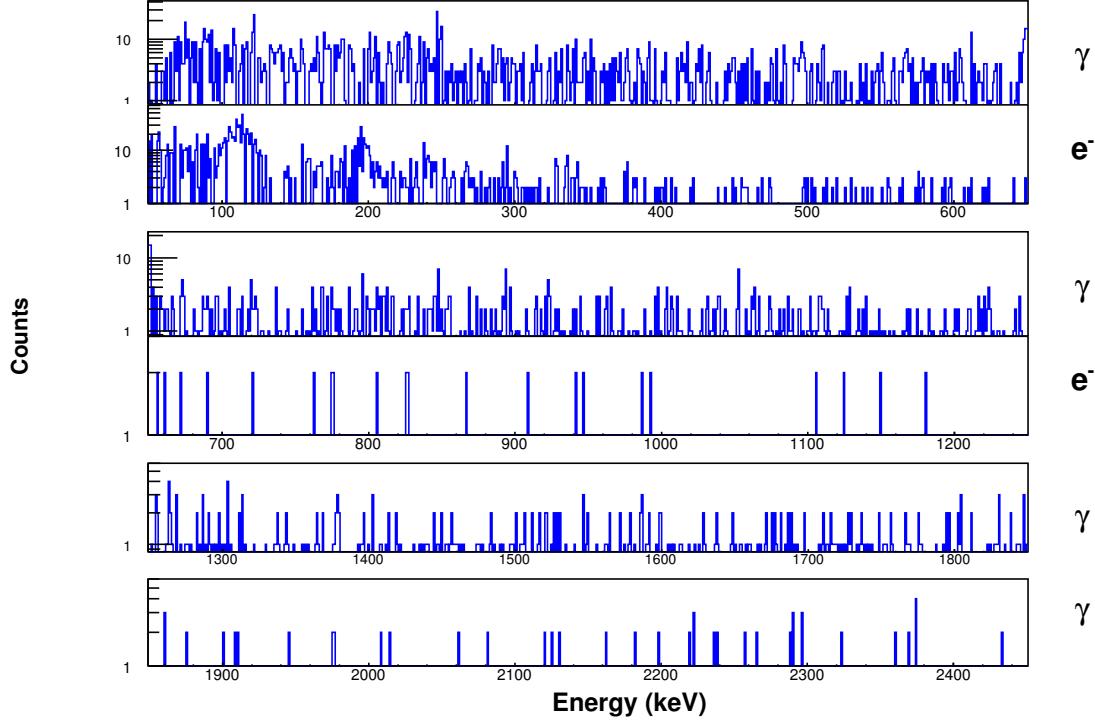
612 keV gated spectra



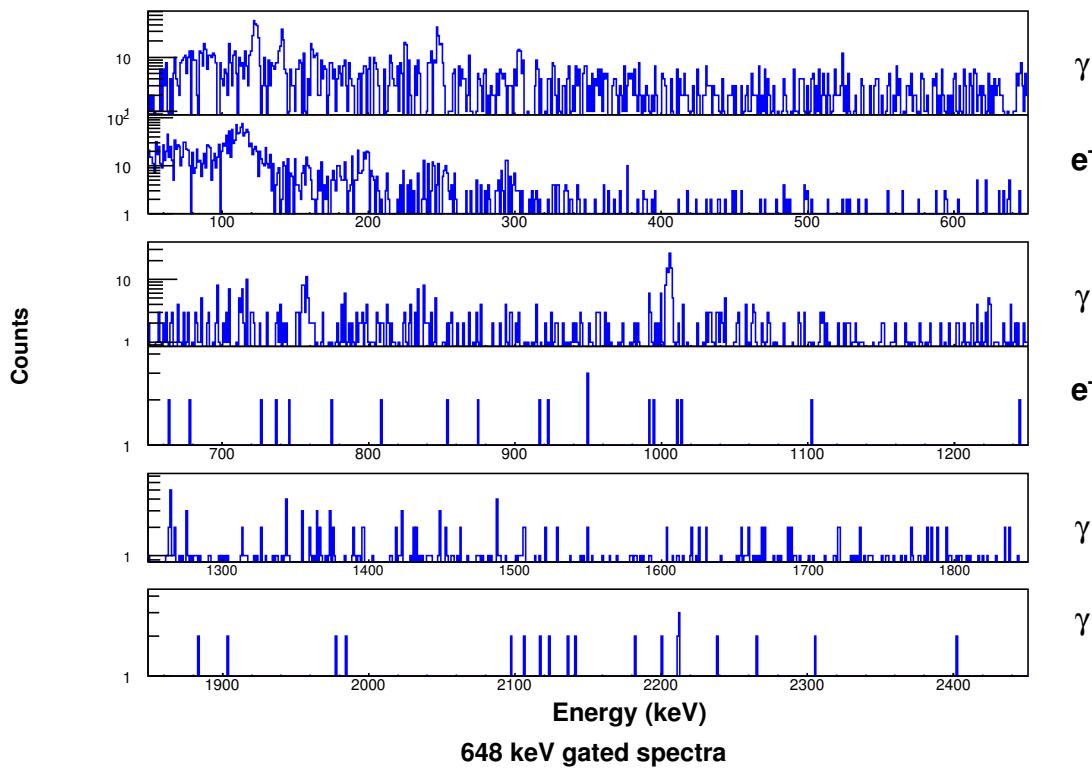
630 keV gated spectra



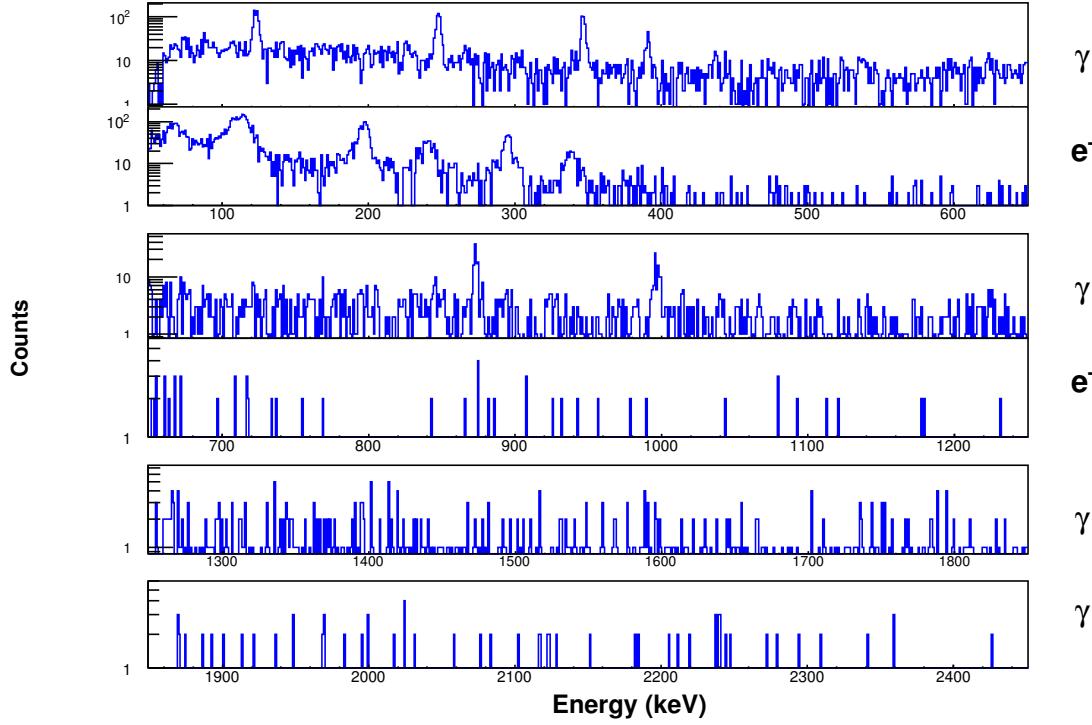
637 keV gated spectra



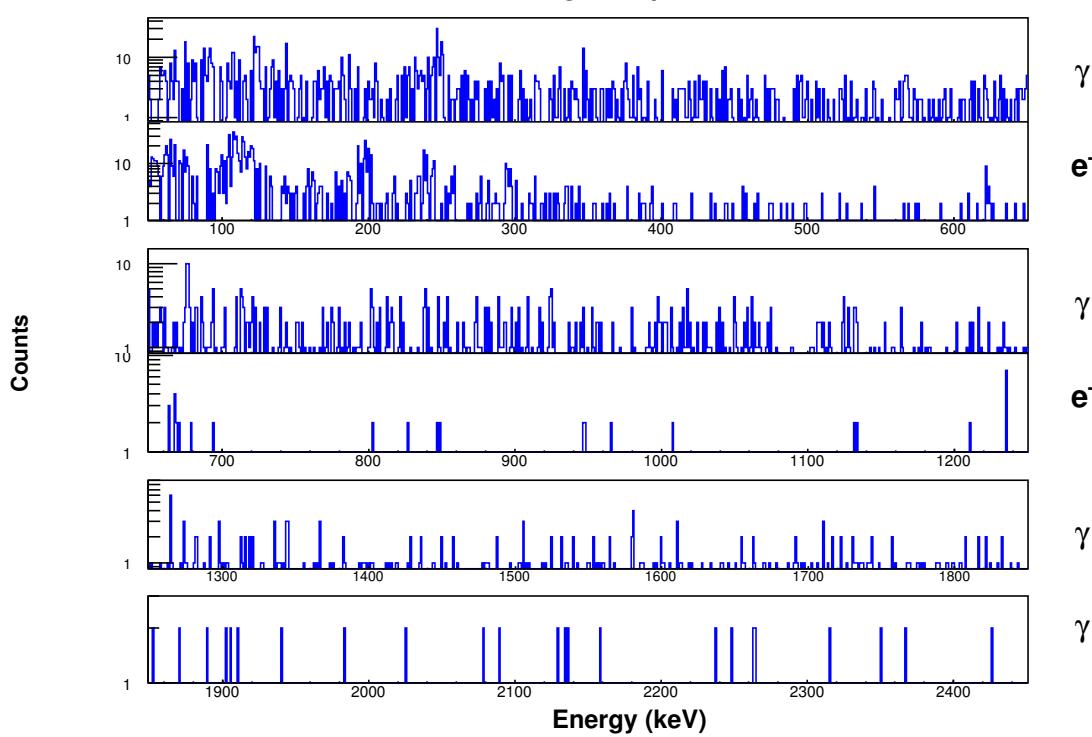
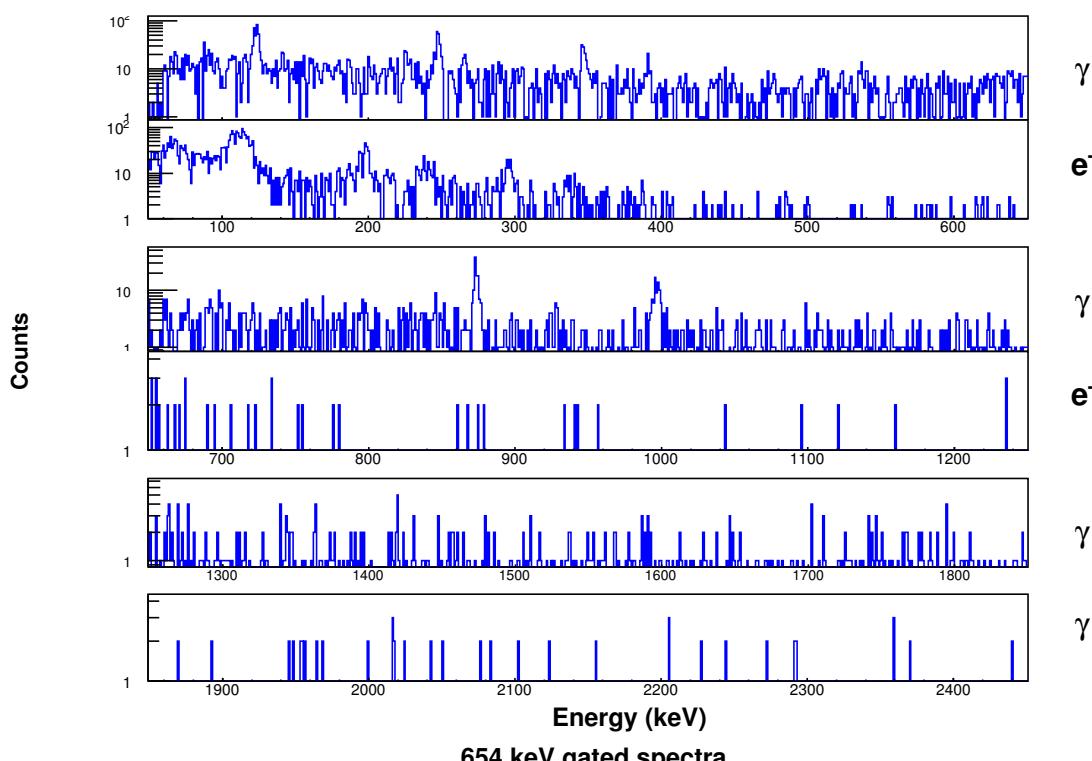
642 keV gated spectra



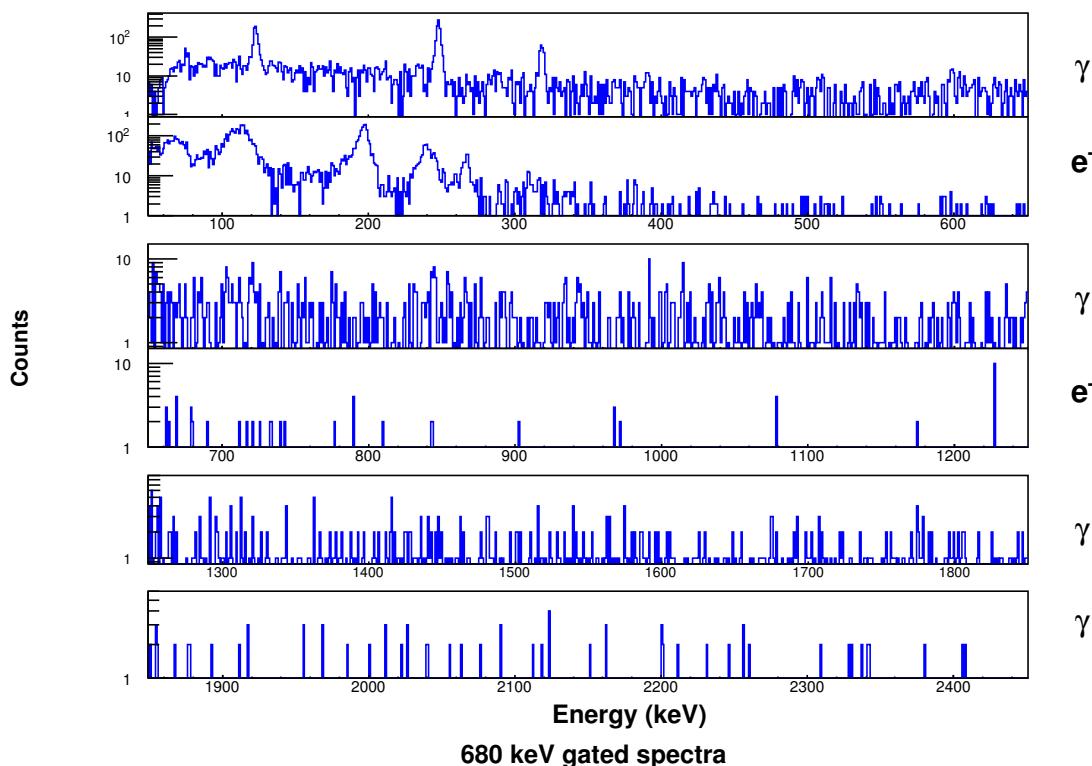
648 keV gated spectra



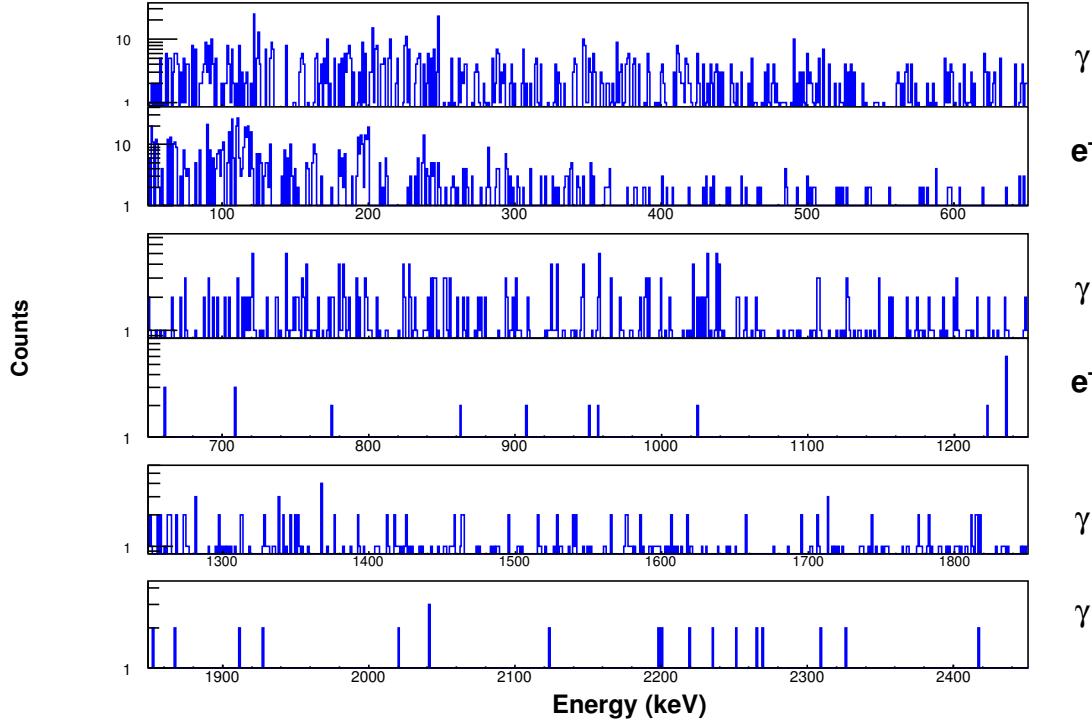
650 keV gated spectra



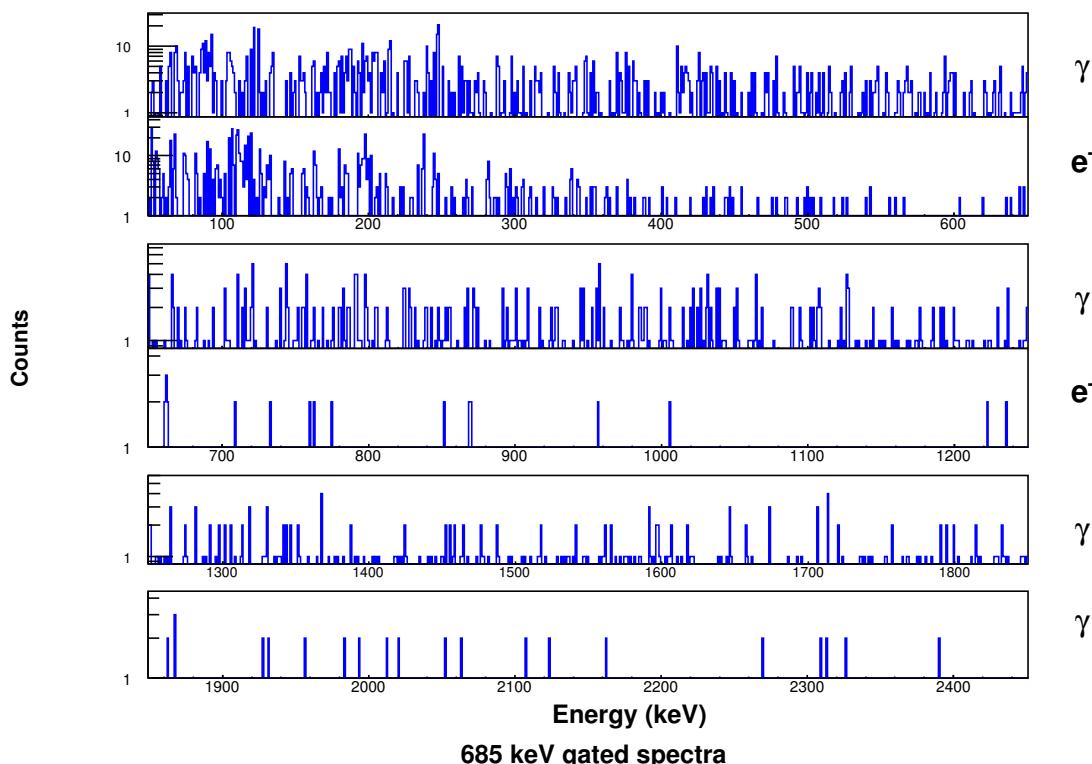
676 keV gated spectra



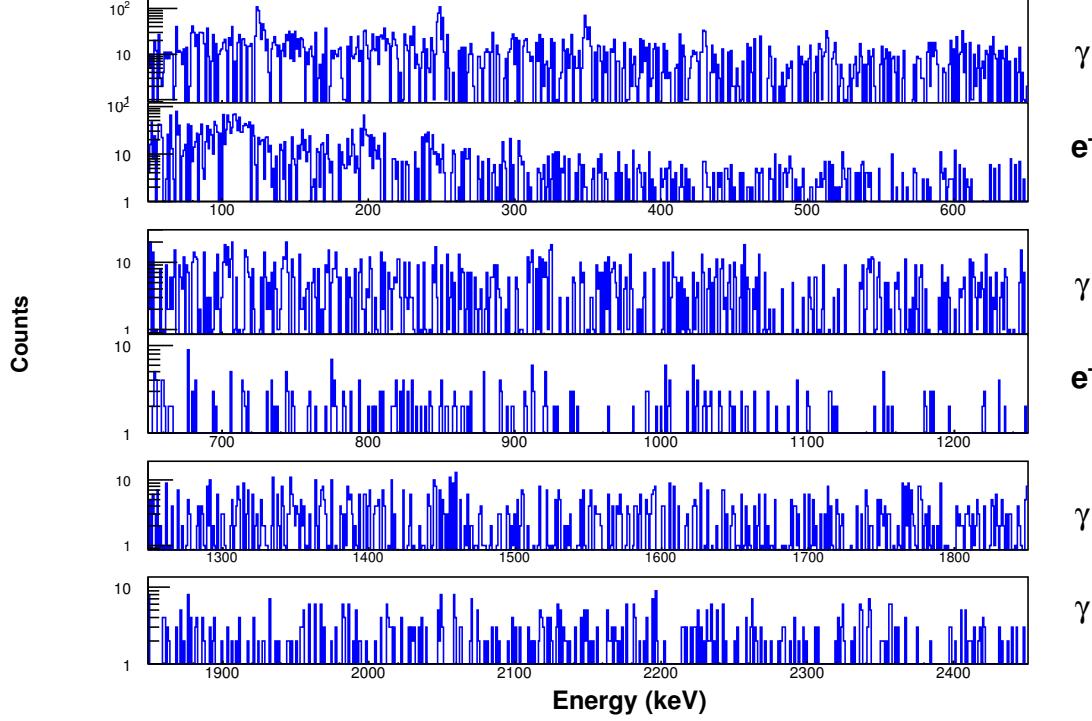
680 keV gated spectra



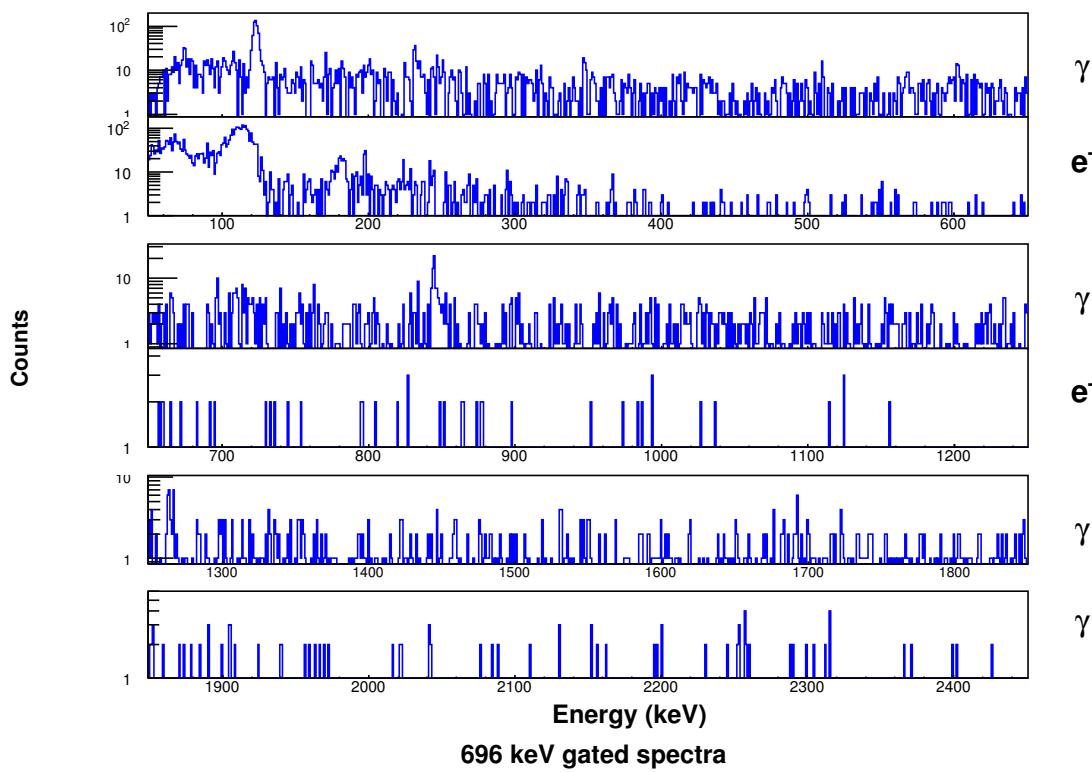
682 keV gated spectra



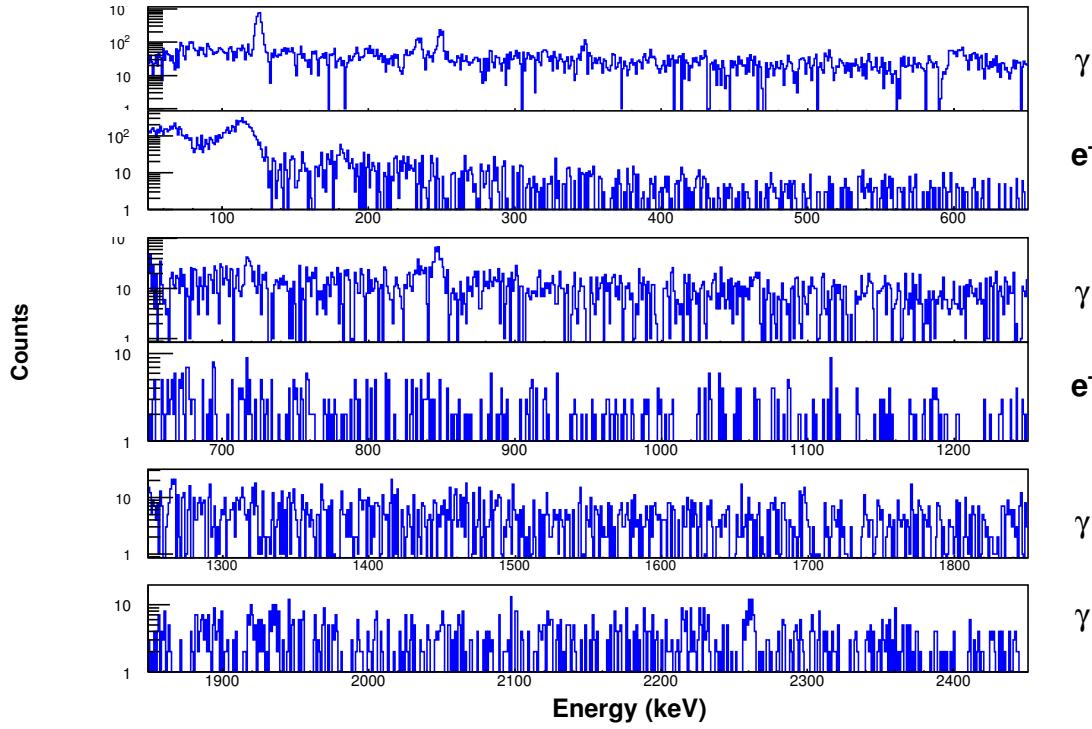
685 keV gated spectra



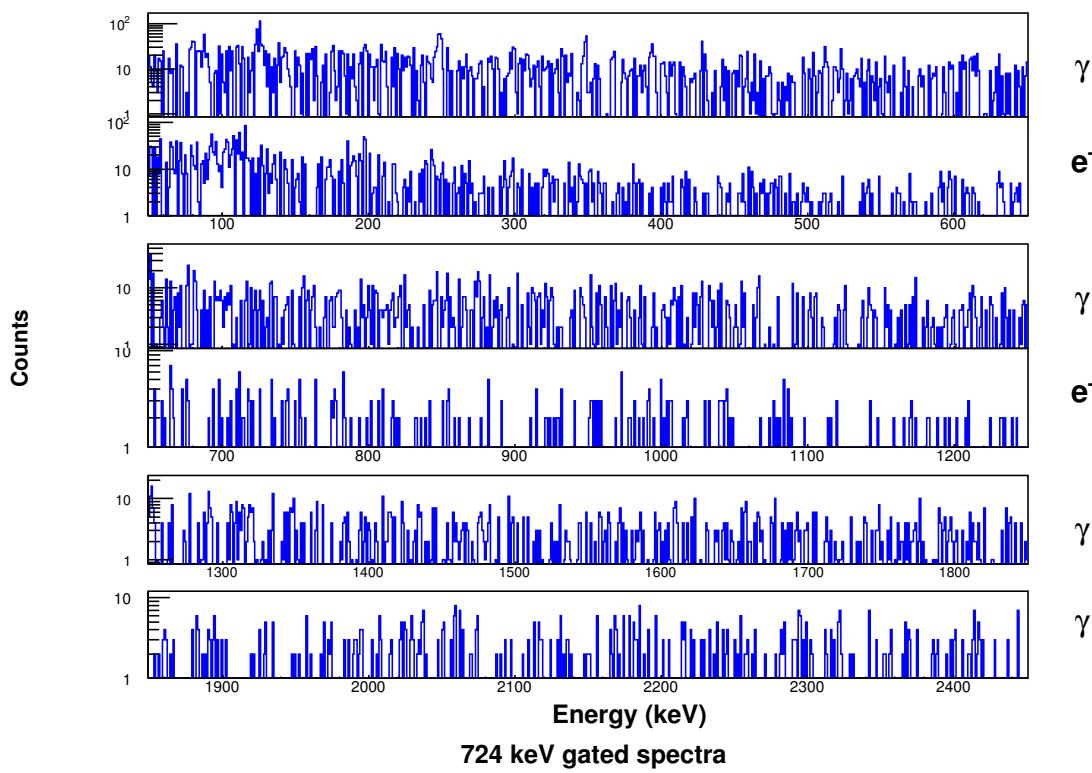
692 keV gated spectra



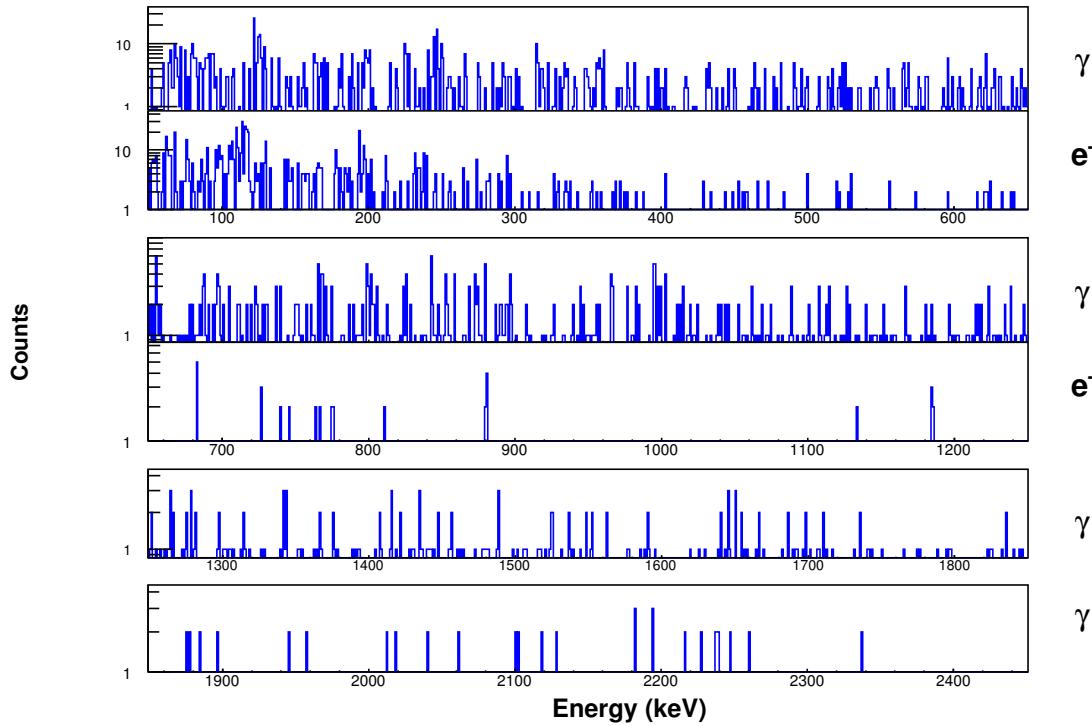
696 keV gated spectra



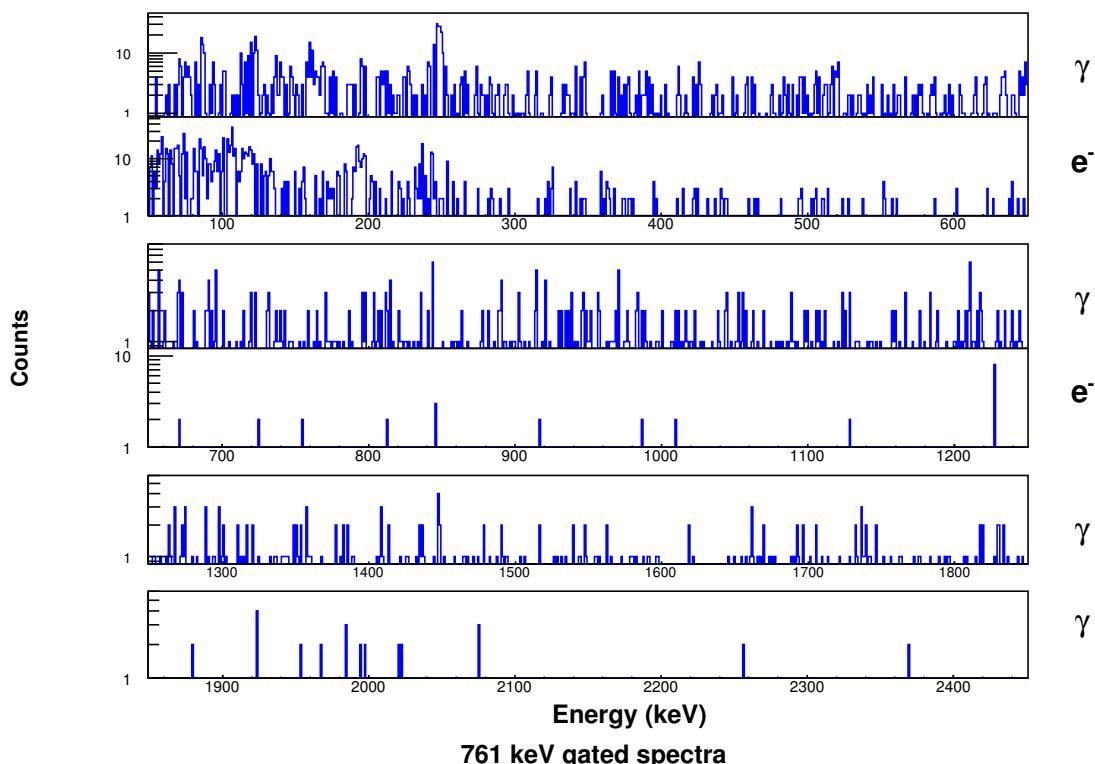
722 keV gated spectra



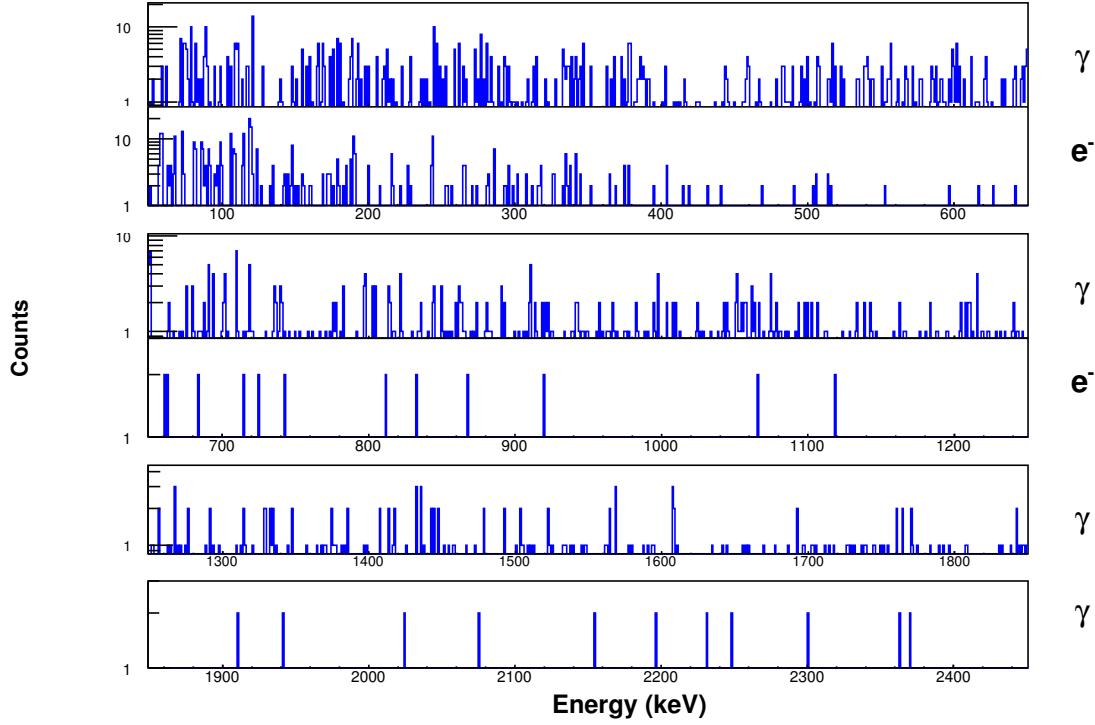
724 keV gated spectra



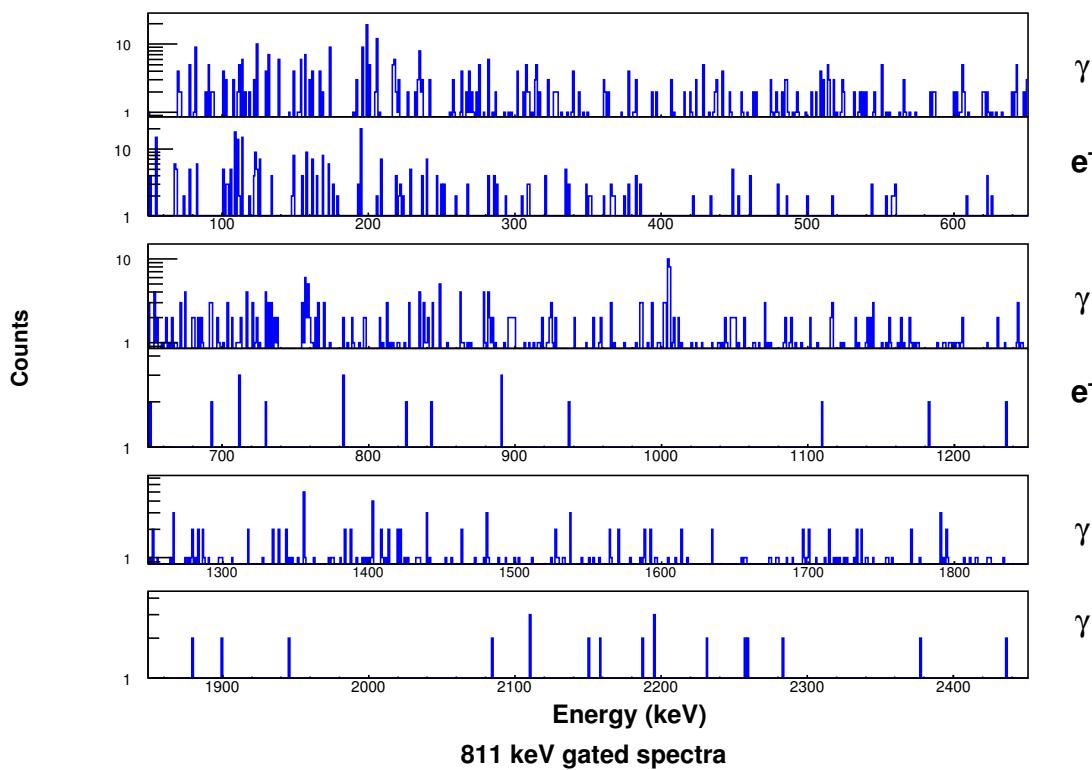
758 keV gated spectra



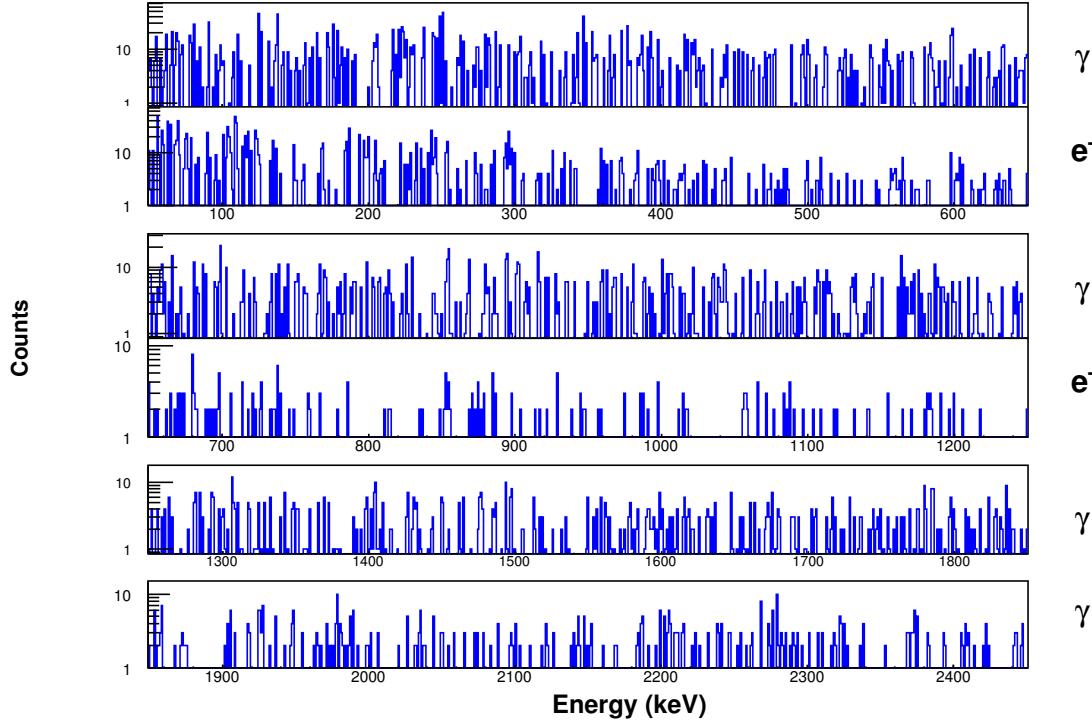
761 keV gated spectra



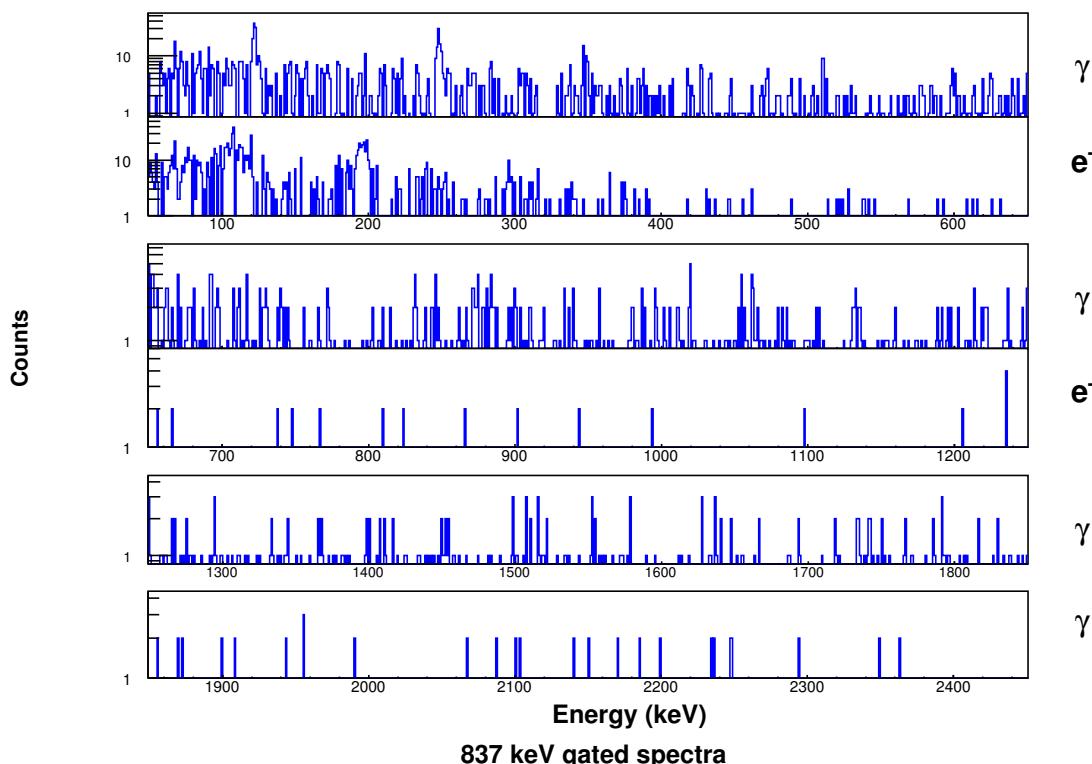
797 keV gated spectra



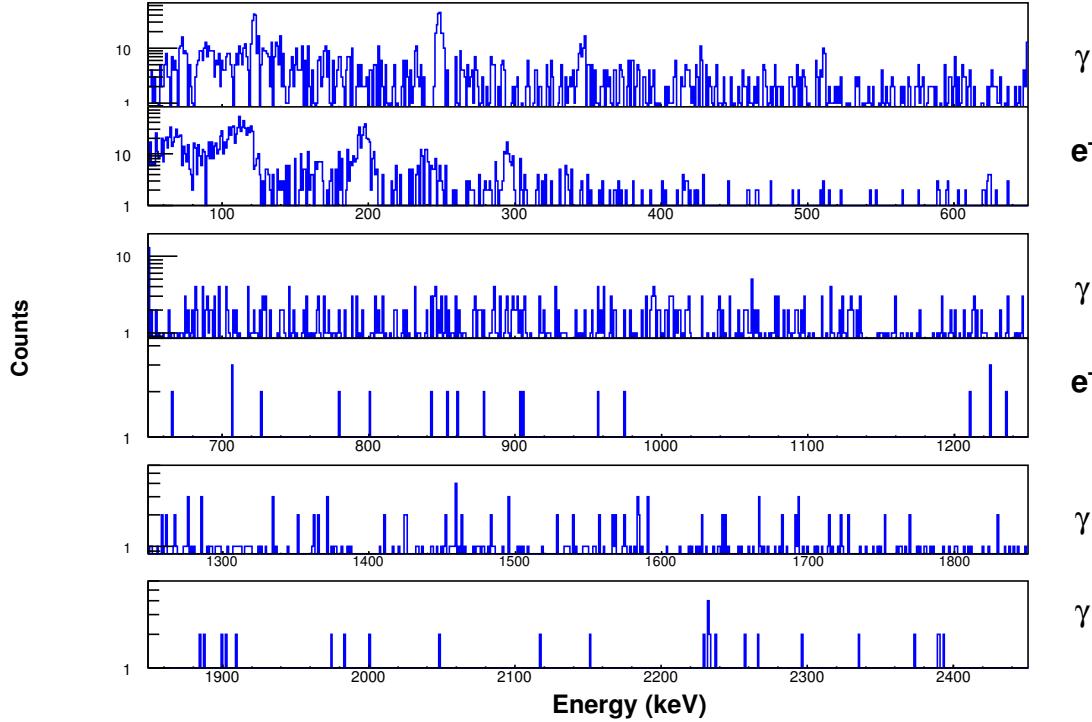
811 keV gated spectra

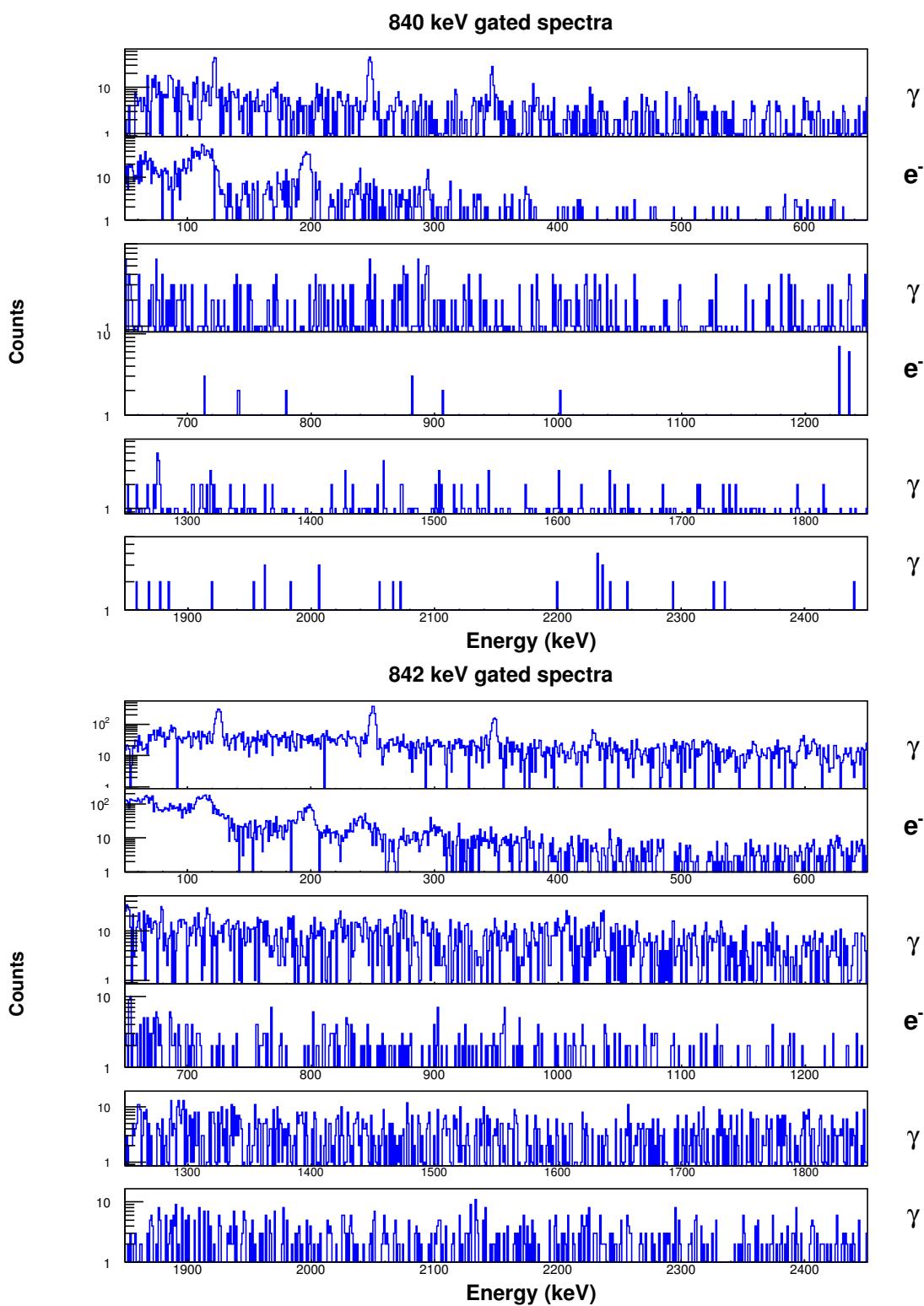


834 keV gated spectra

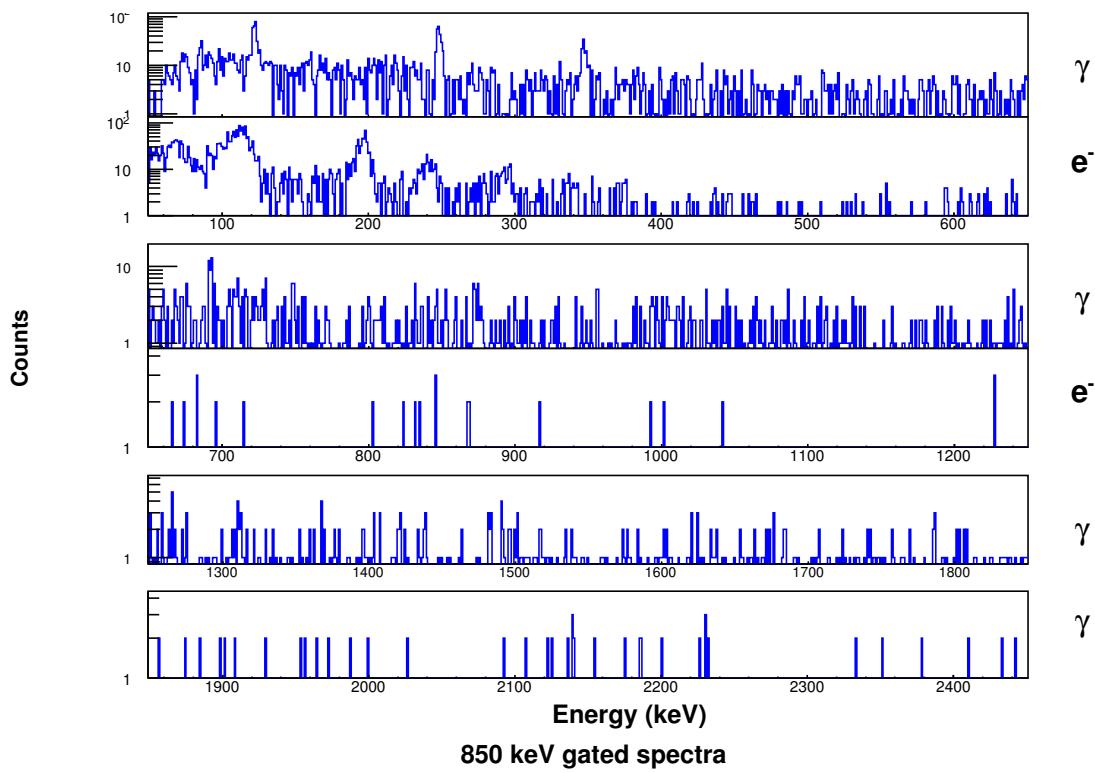


837 keV gated spectra

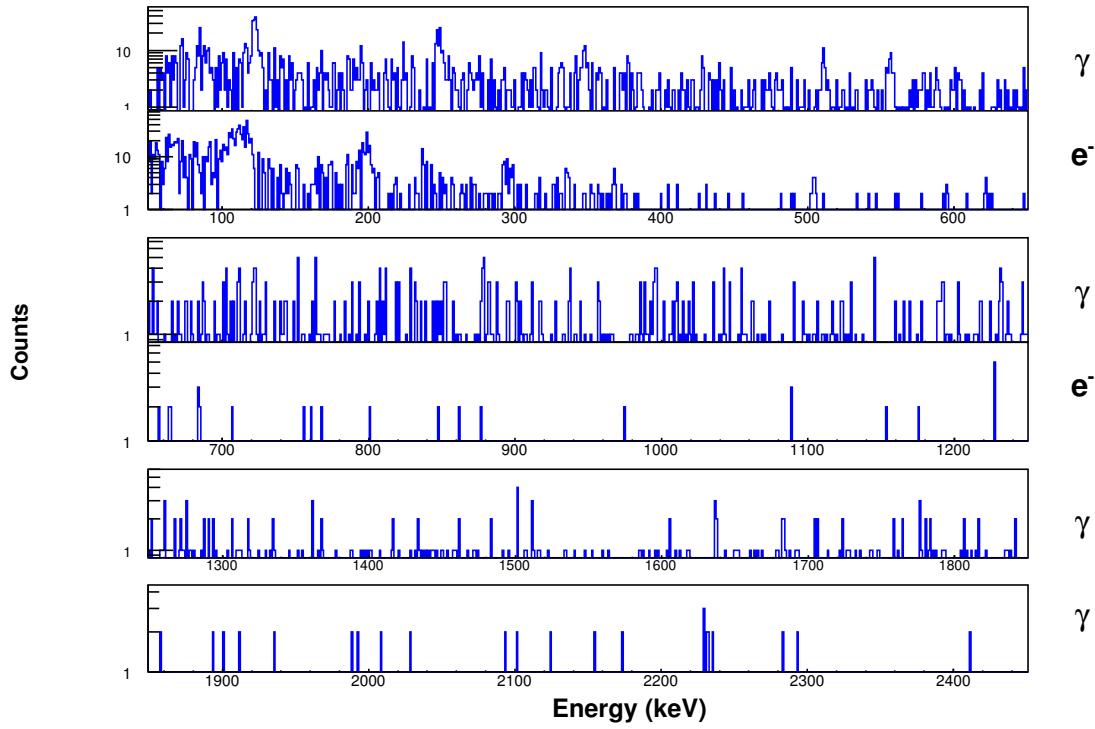




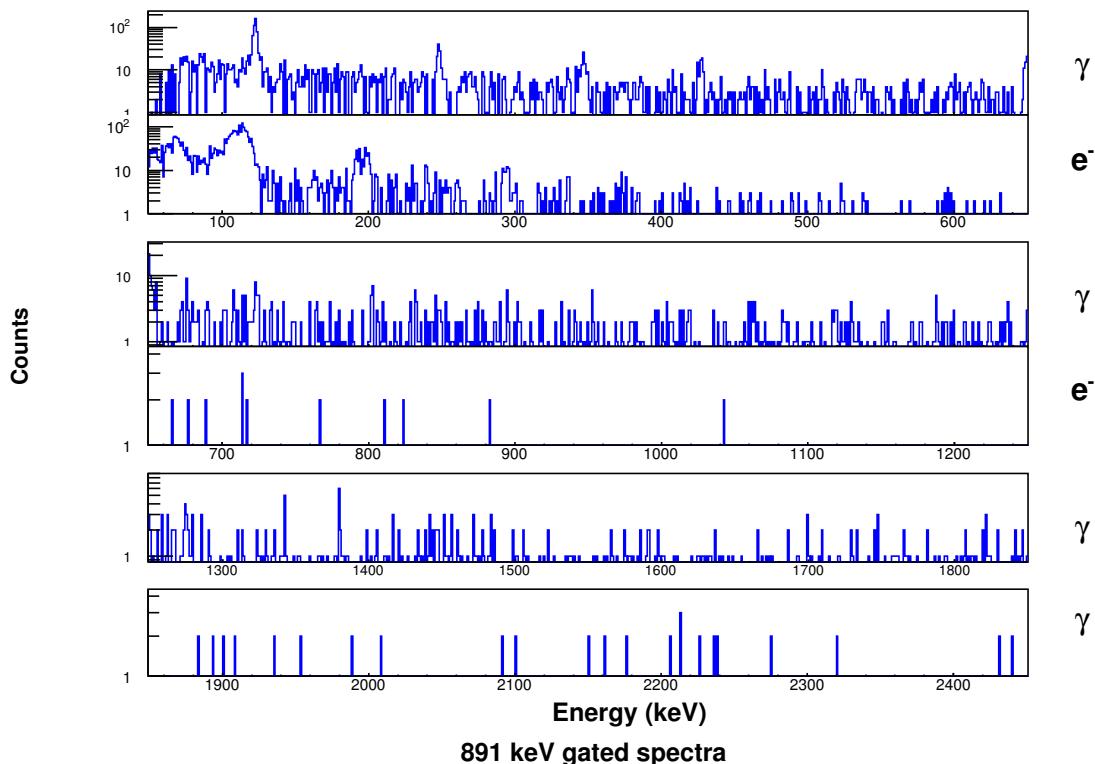
844 keV gated spectra



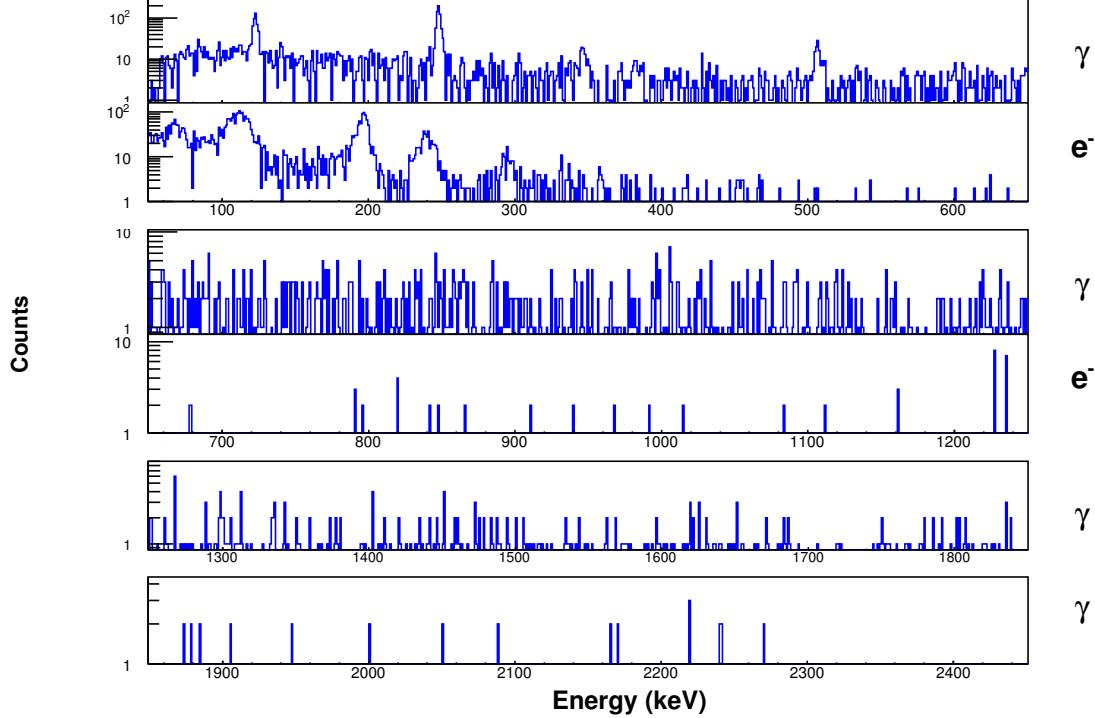
850 keV gated spectra



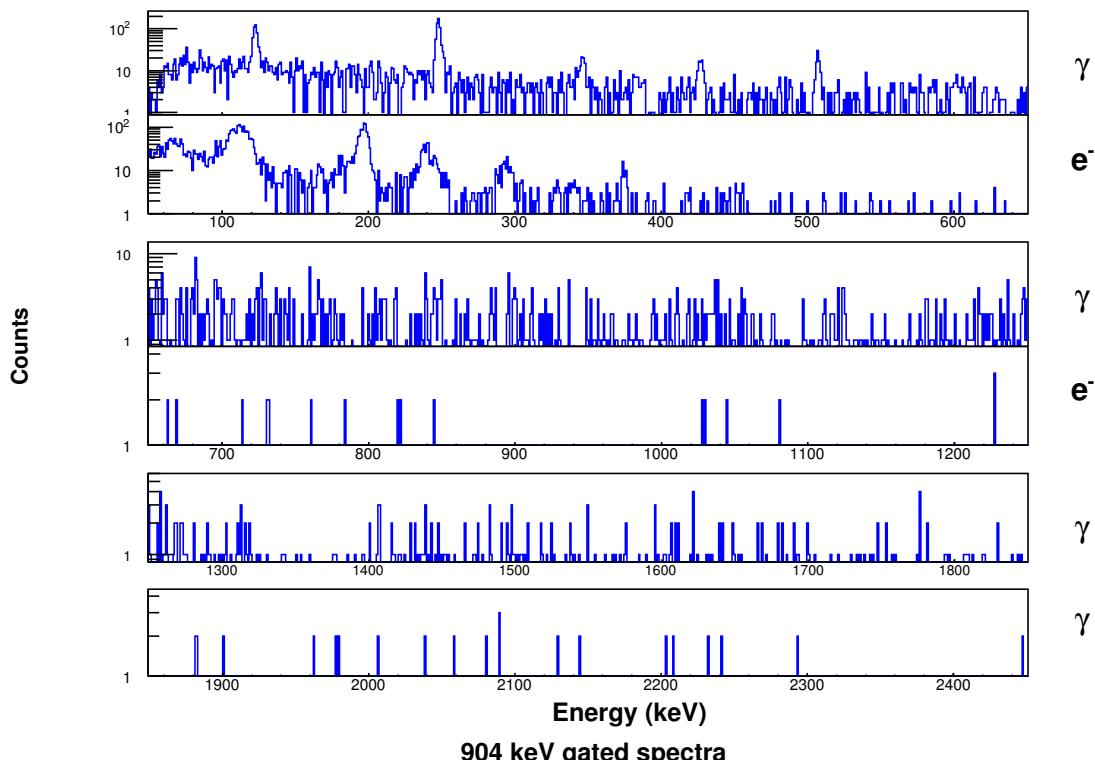
874 keV gated spectra



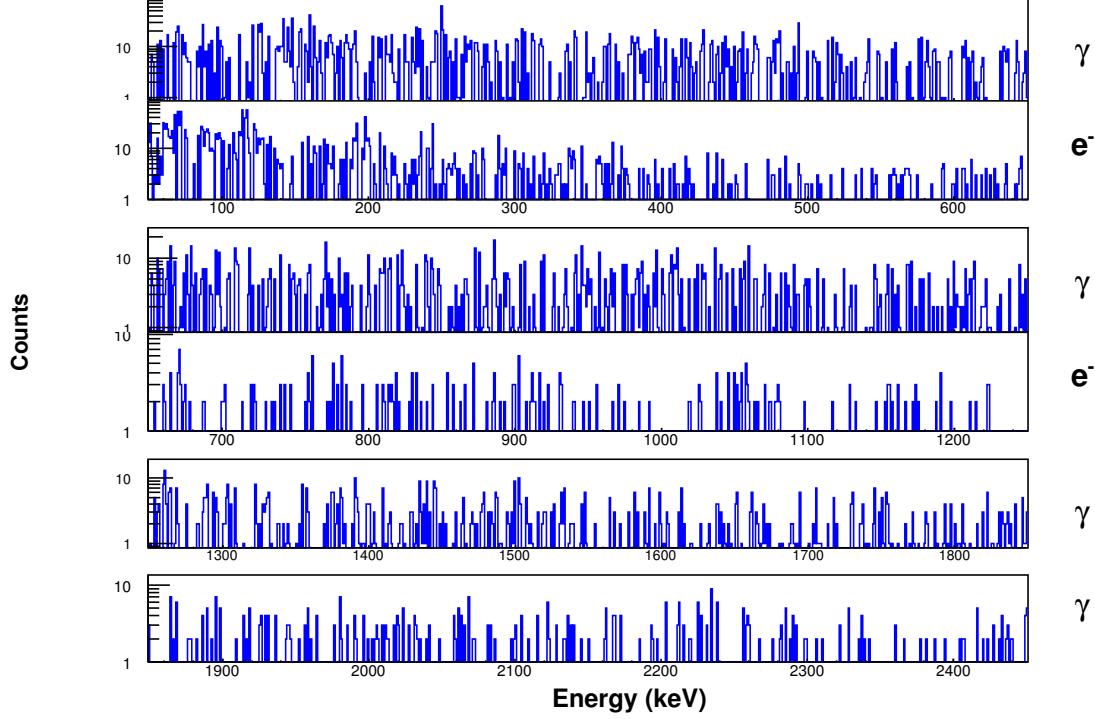
891 keV gated spectra

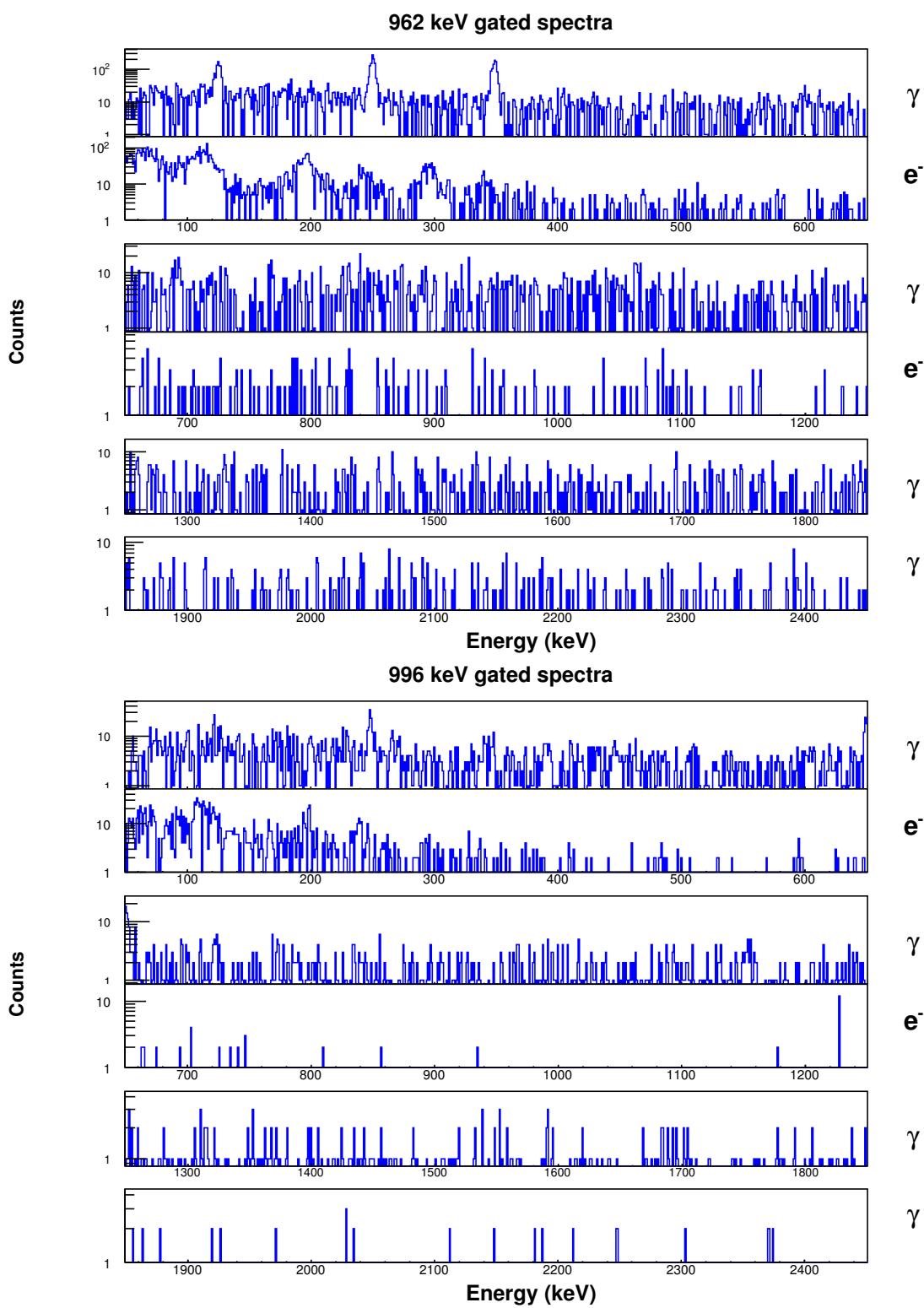


894 keV gated spectra

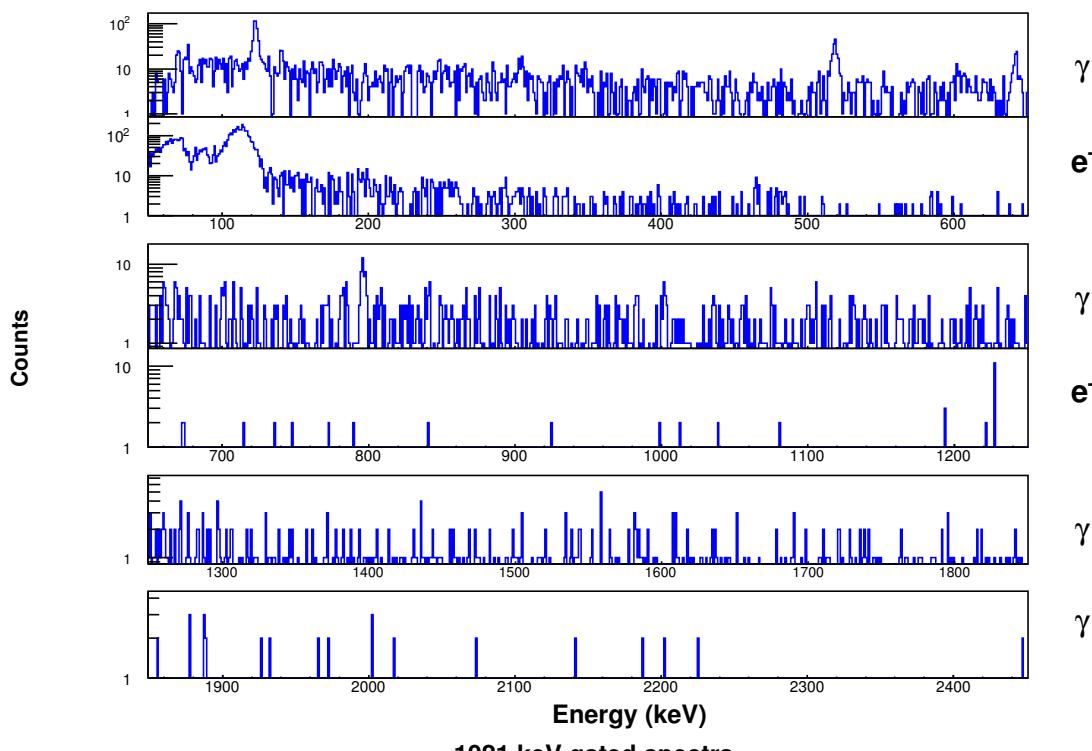


904 keV gated spectra

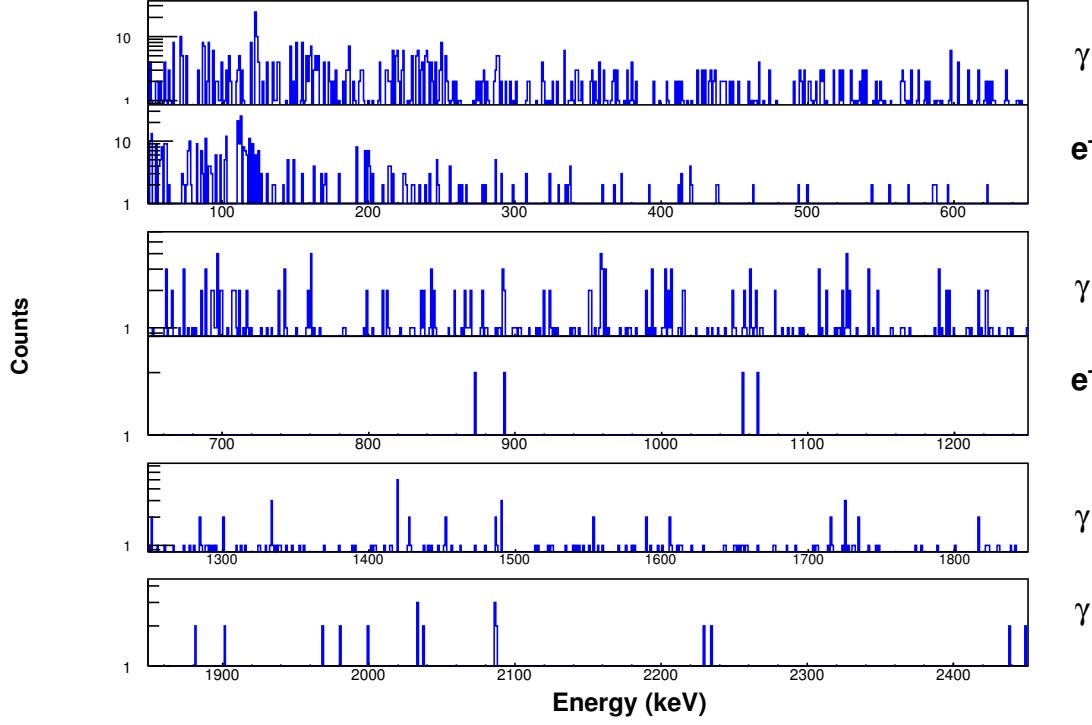




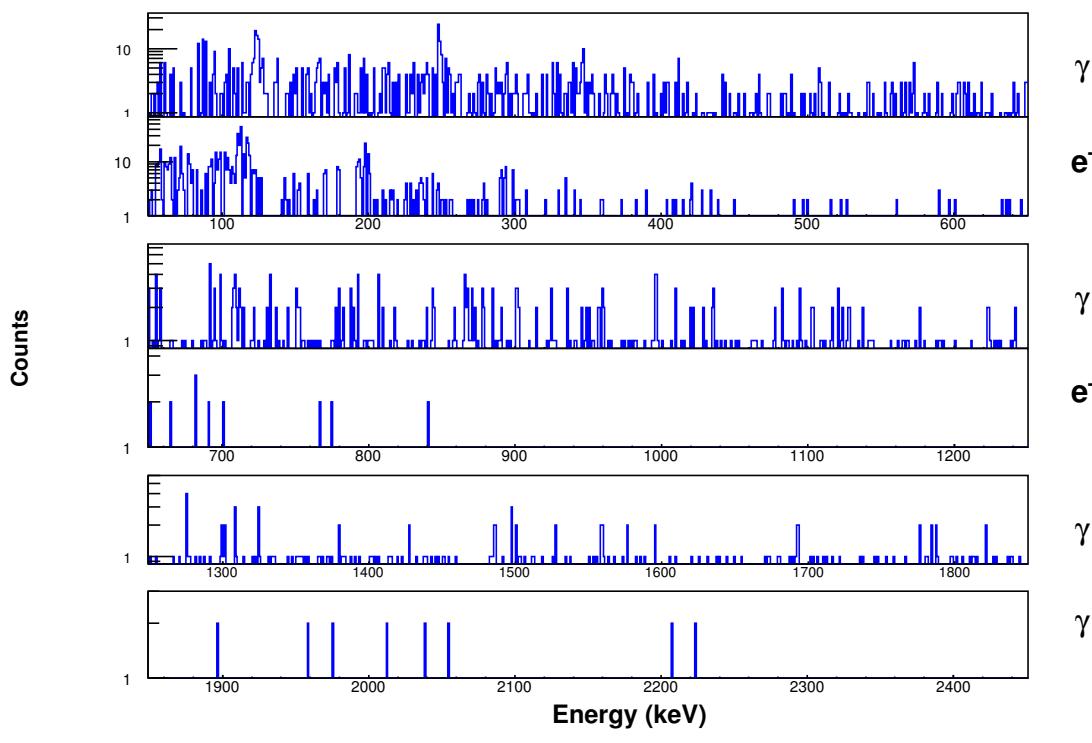
1004 keV gated spectra



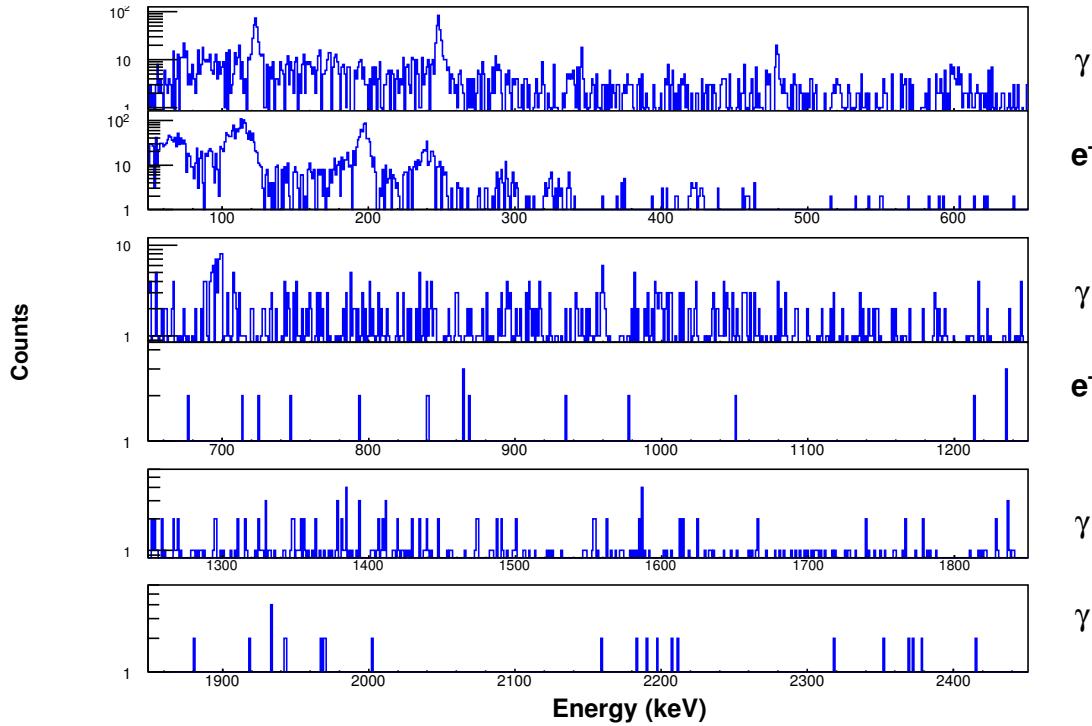
1021 keV gated spectra



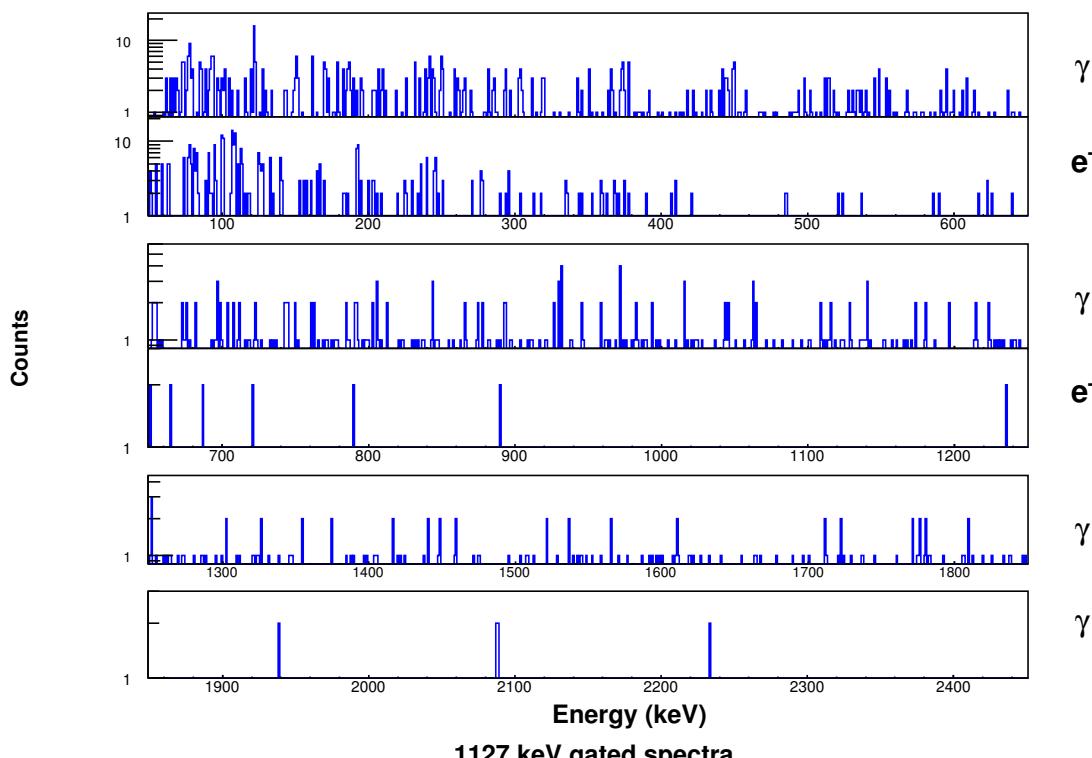
1043 keV gated spectra



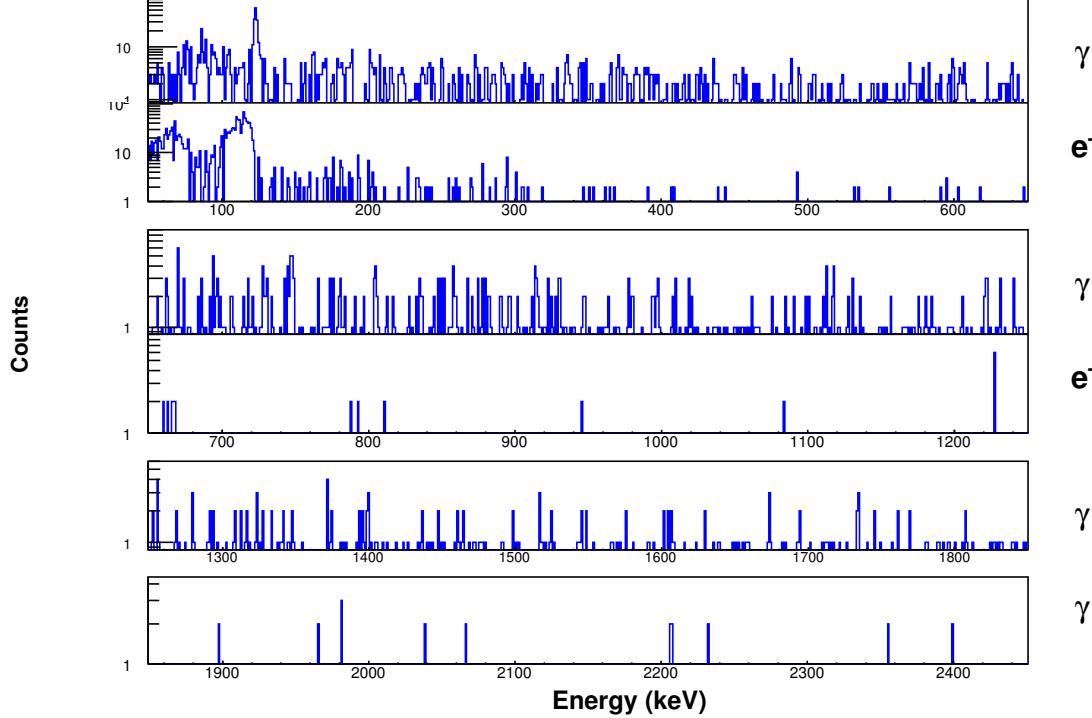
1059 keV gated spectra



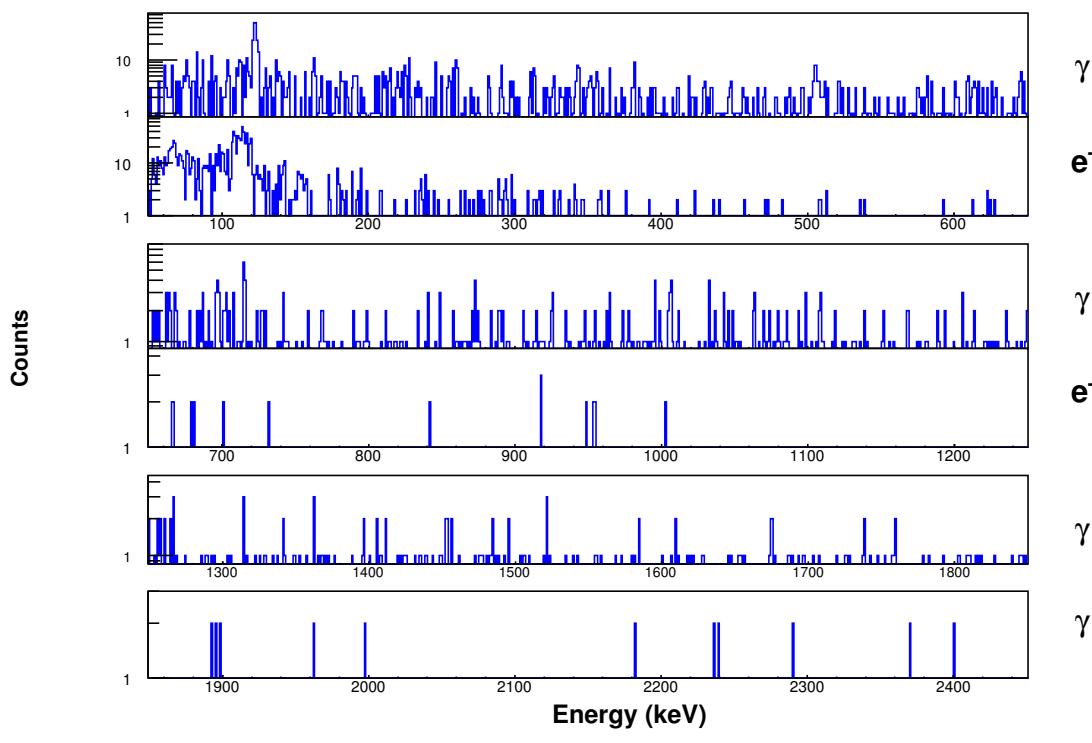
1083 keV gated spectra



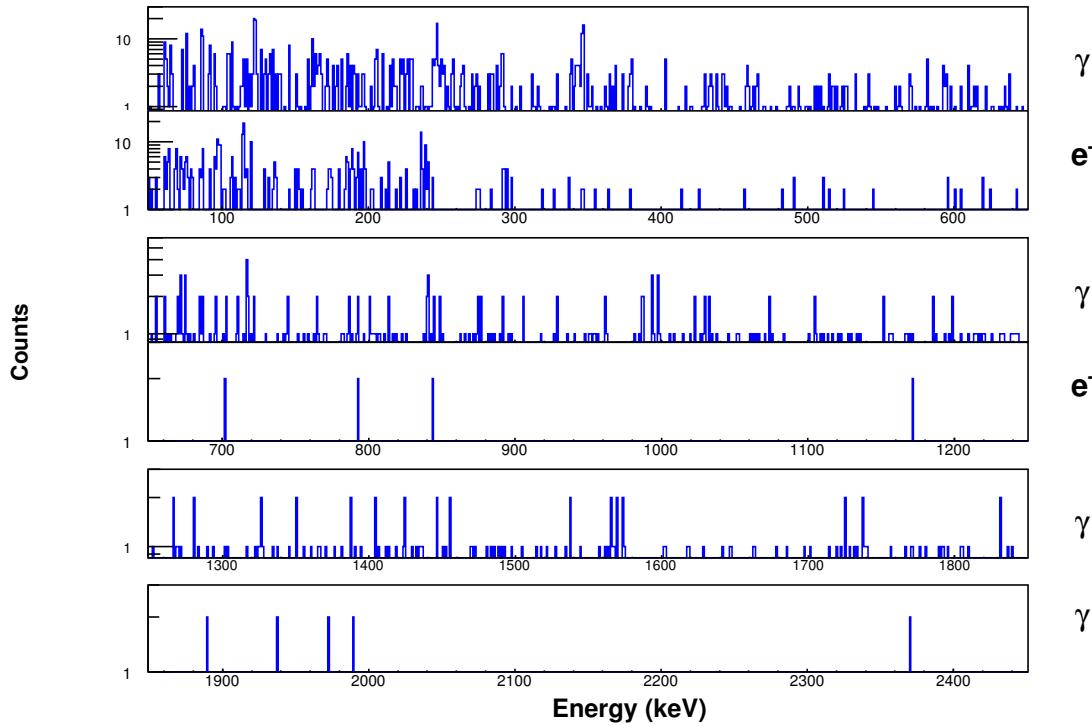
1127 keV gated spectra



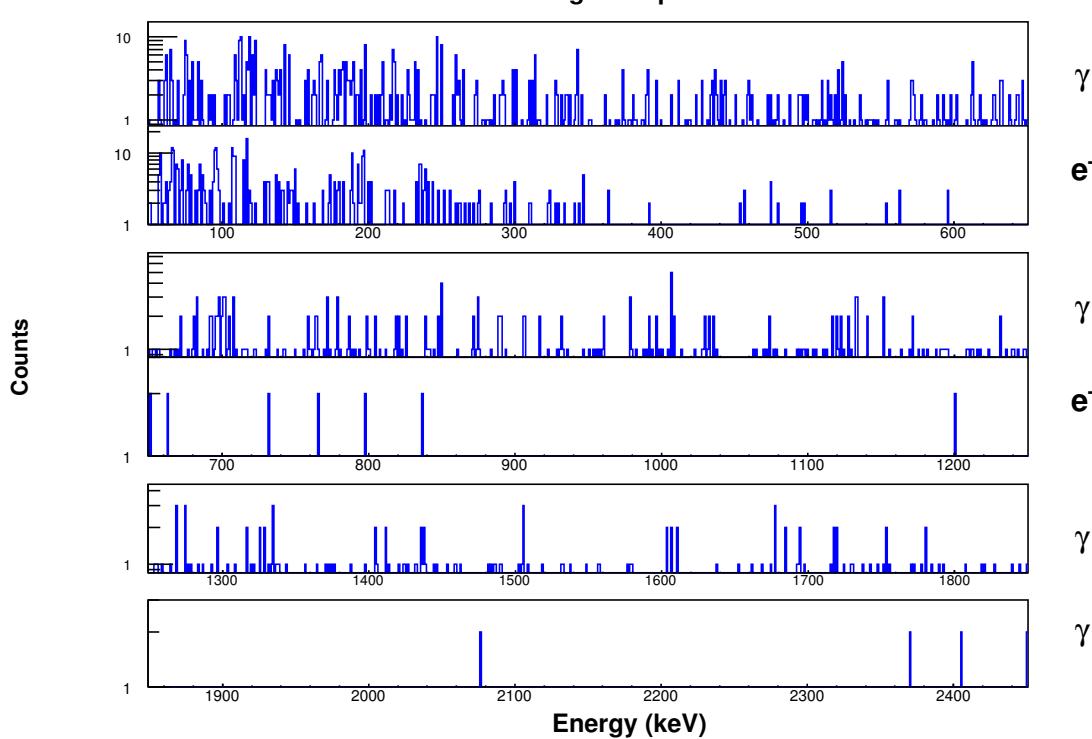
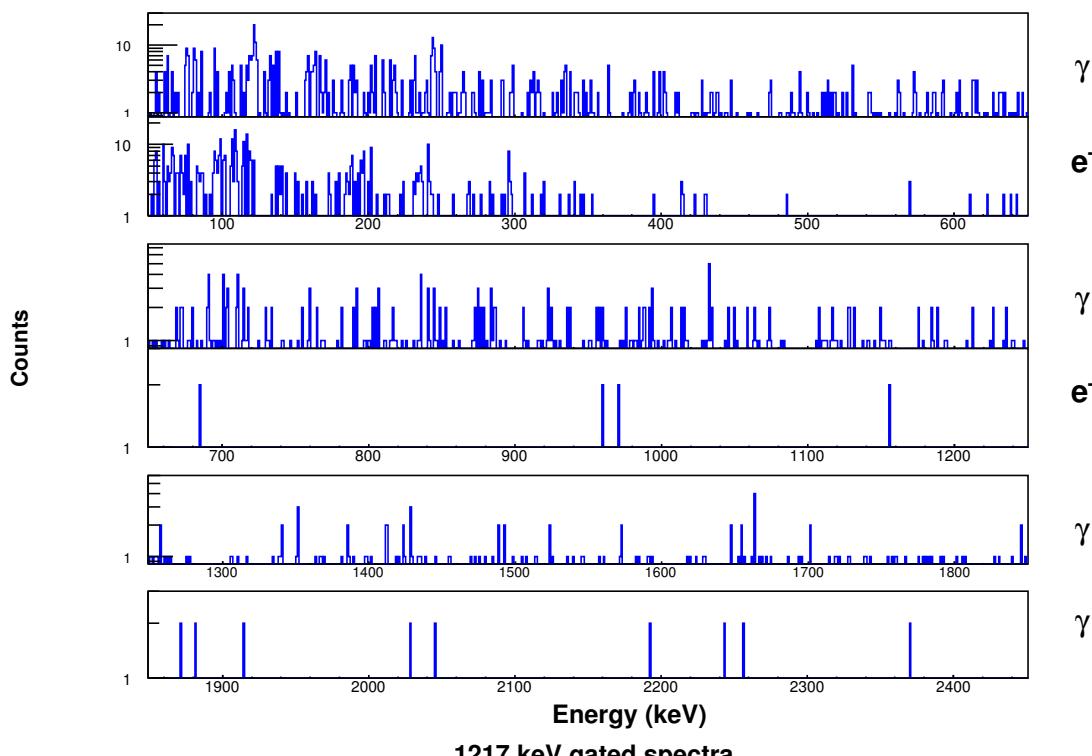
1140 keV gated spectra



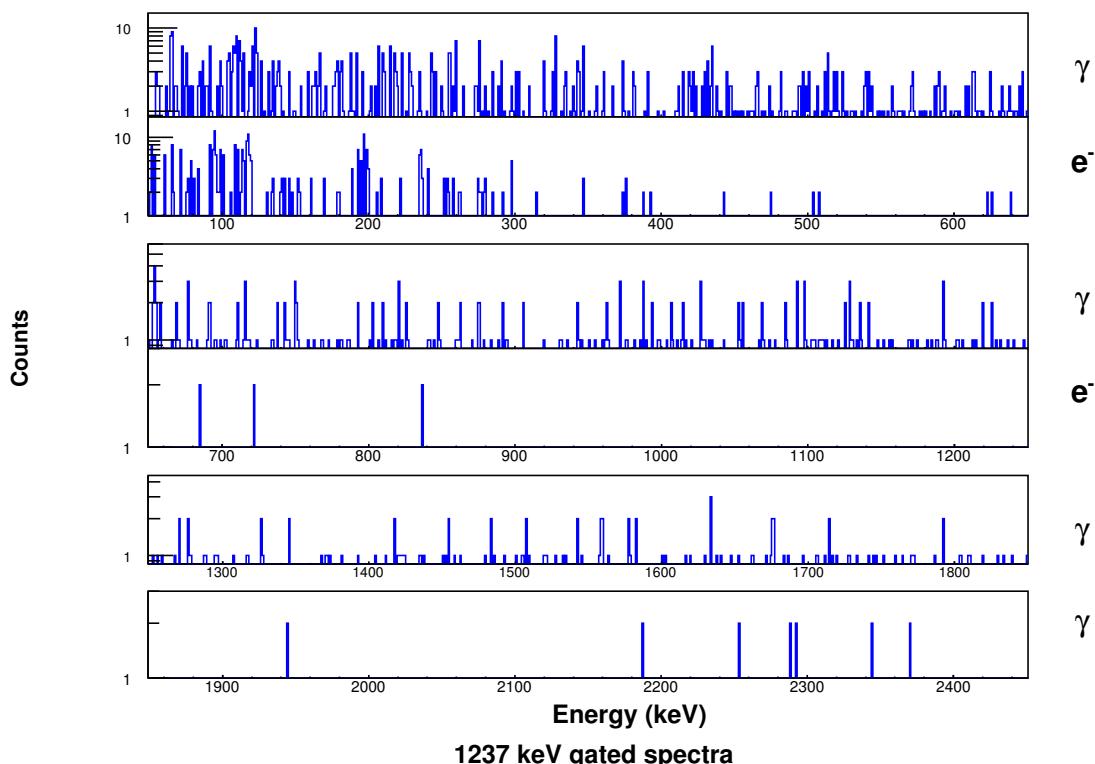
1194 keV gated spectra



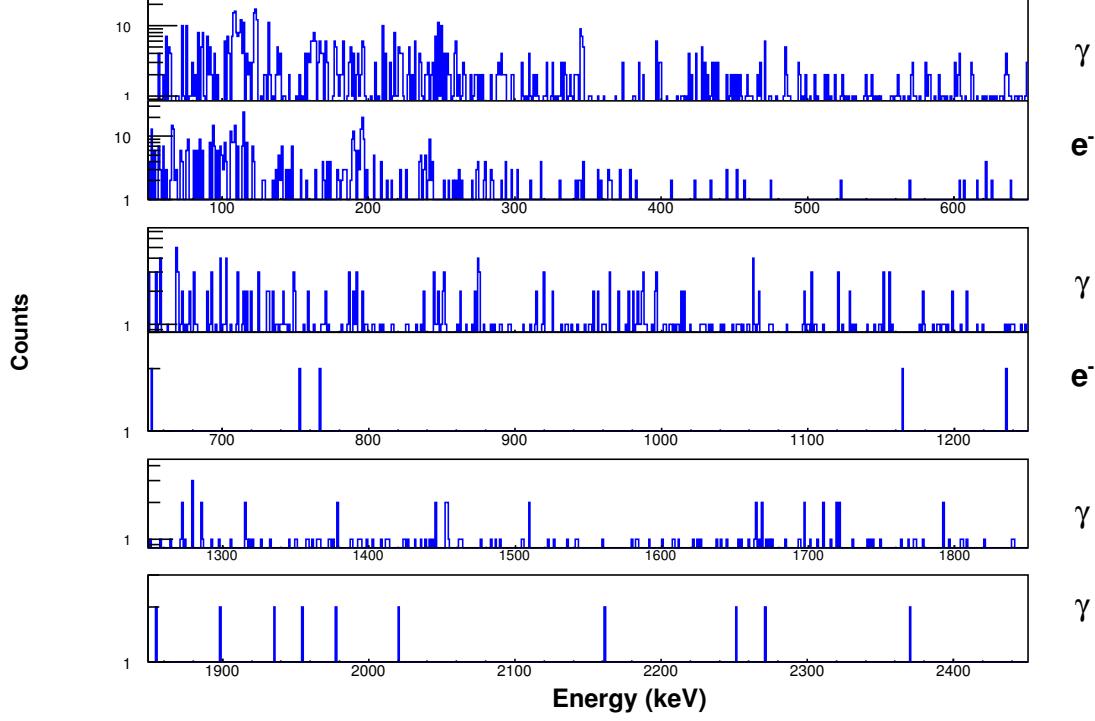
1213 keV gated spectra



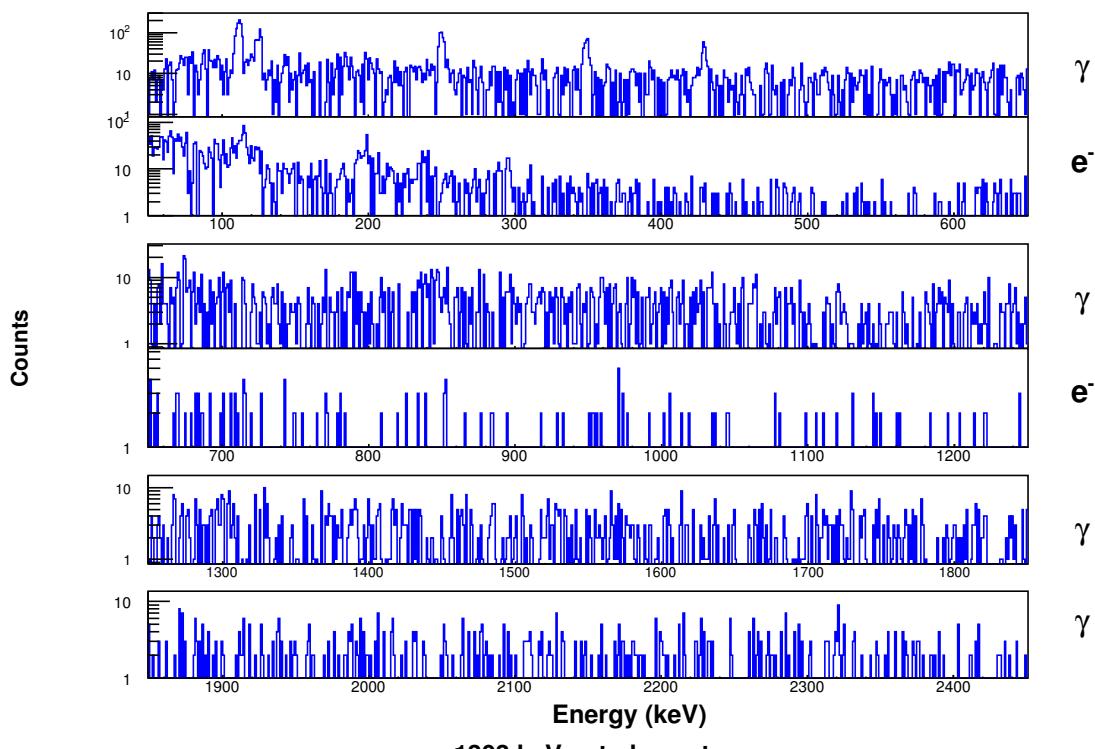
1229 keV gated spectra



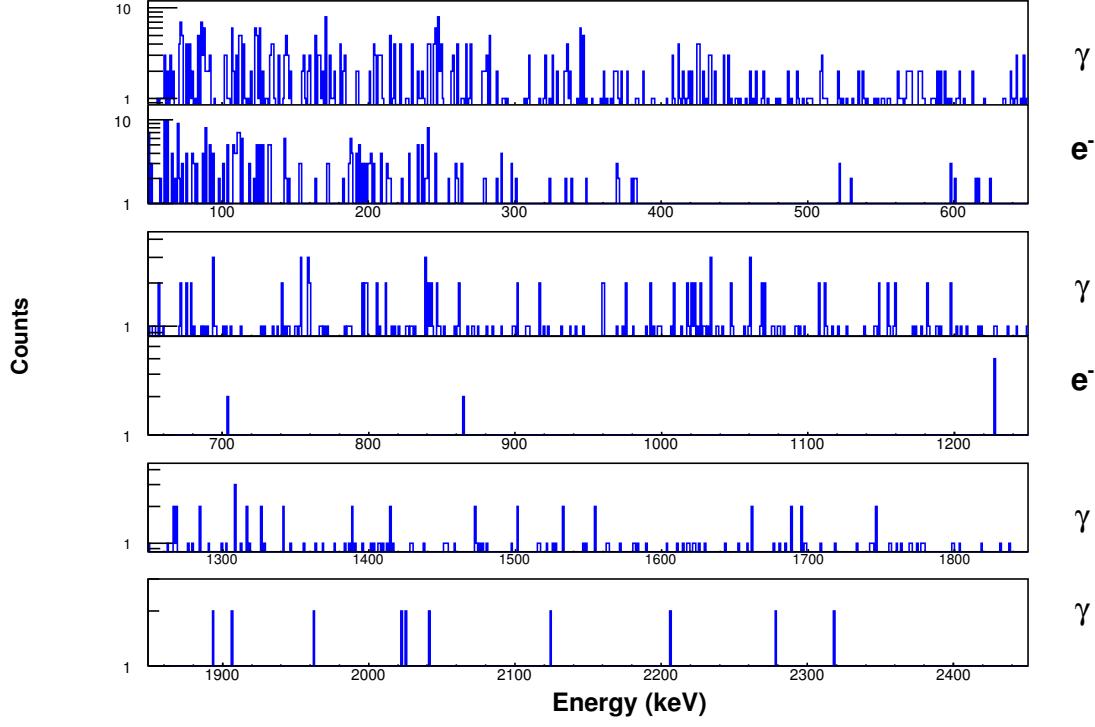
1237 keV gated spectra



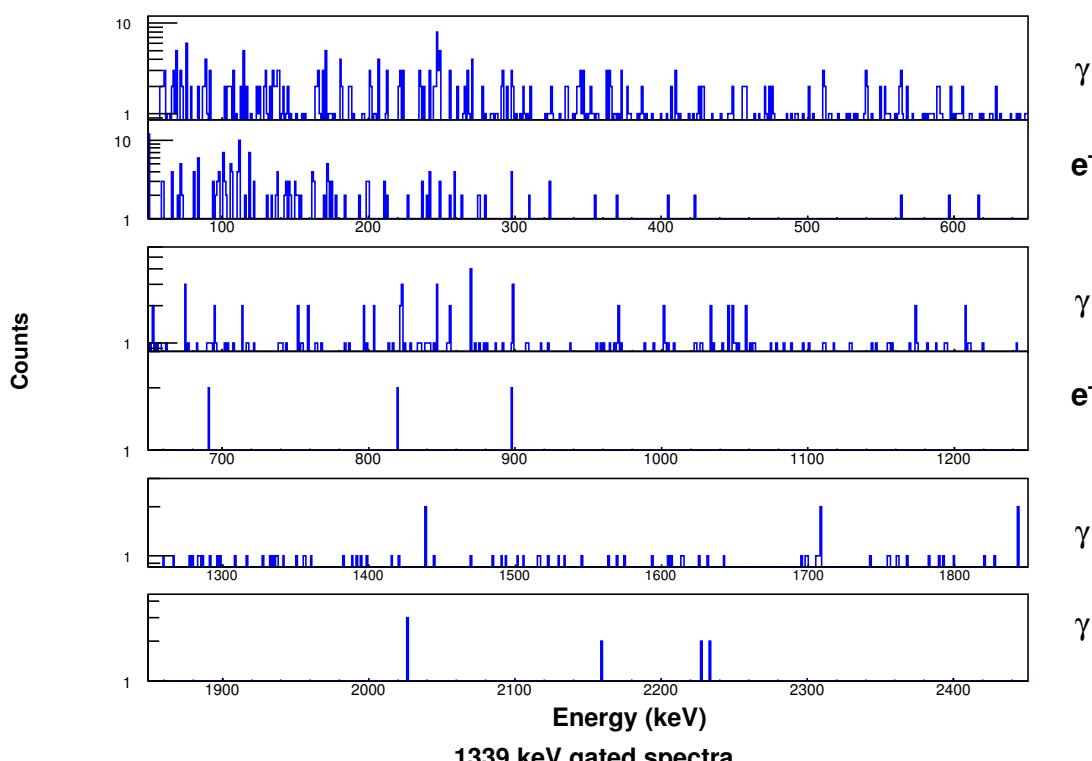
1245 keV gated spectra



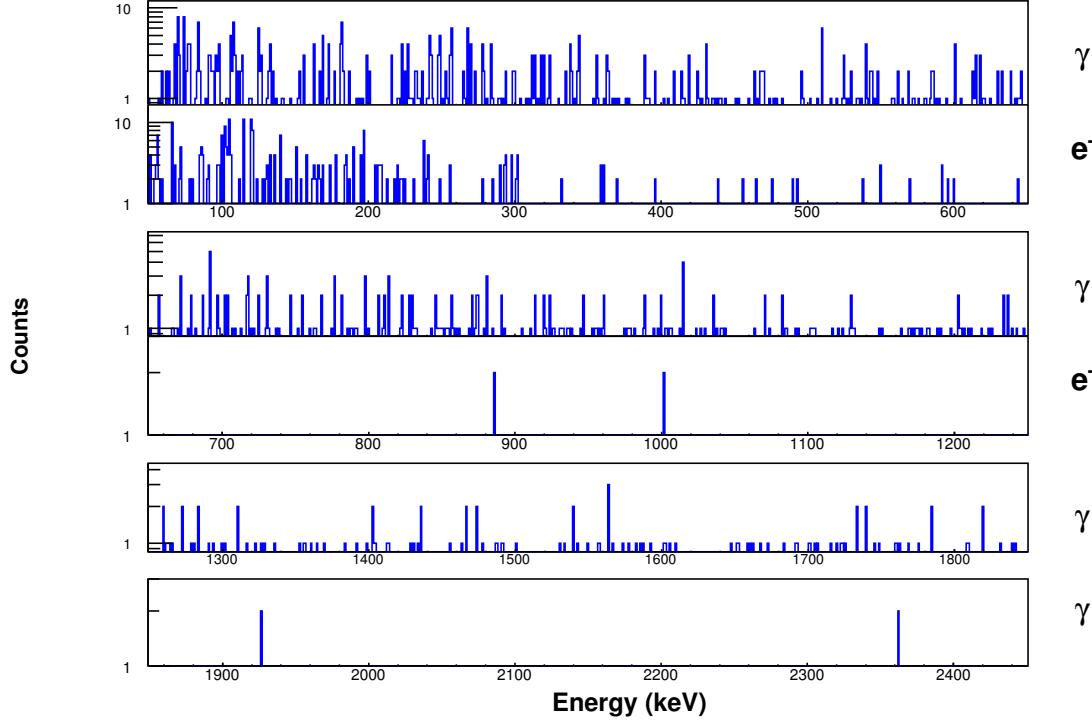
1303 keV gated spectra



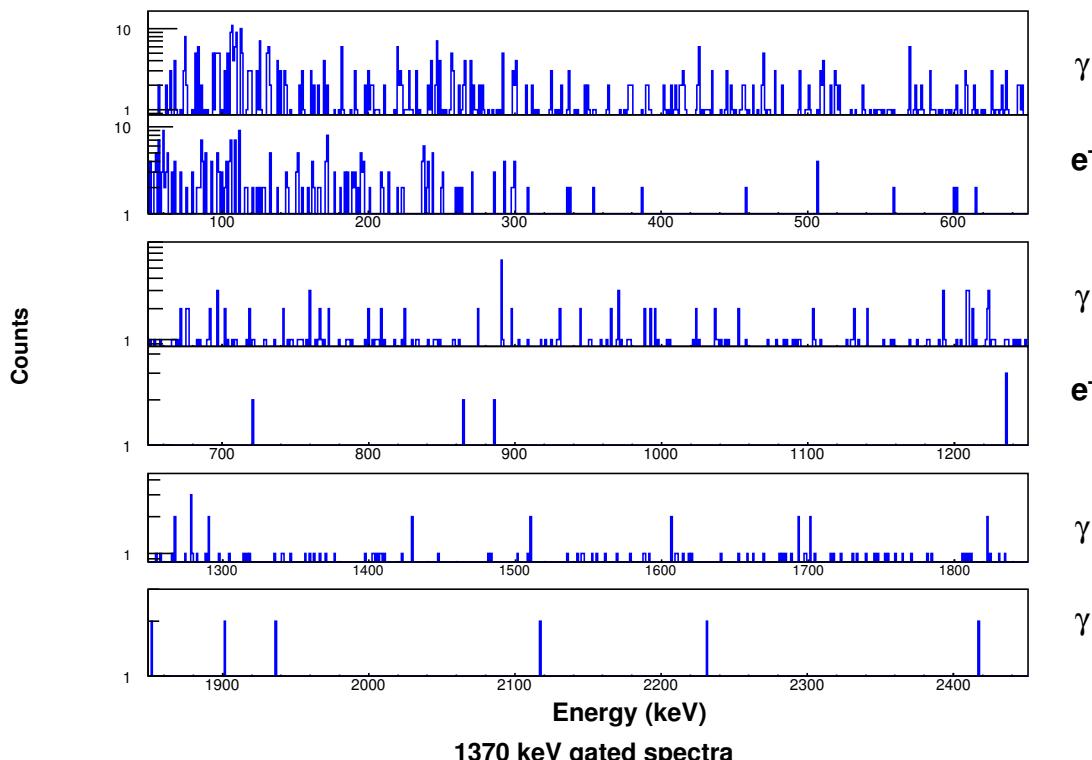
1323 keV gated spectra



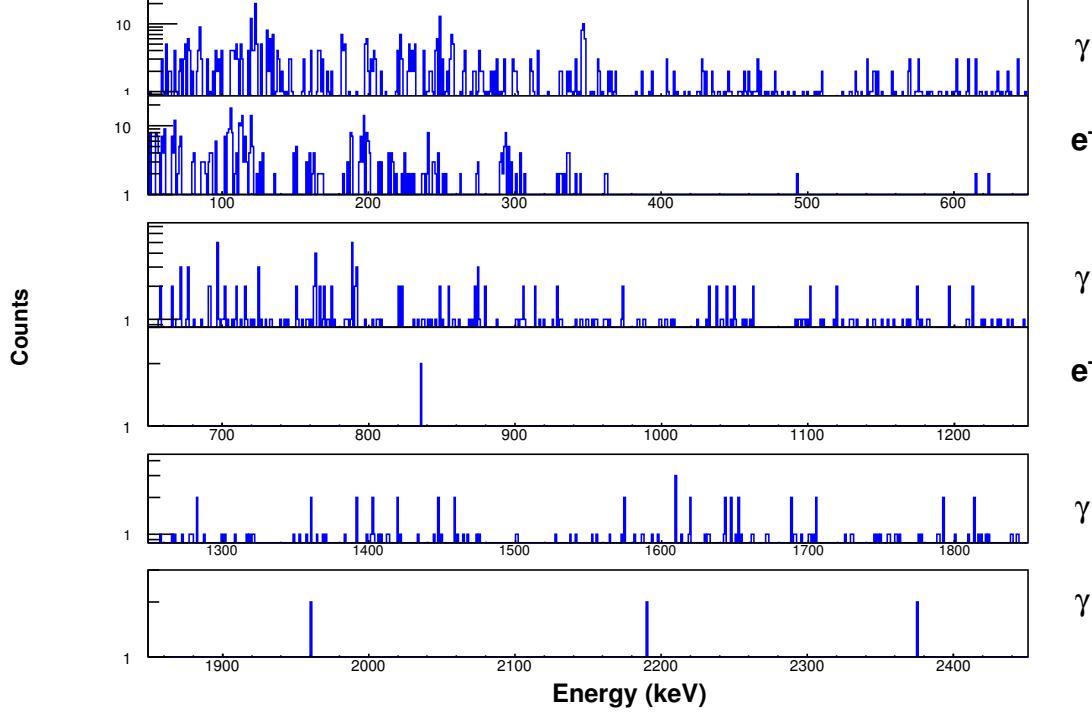
1339 keV gated spectra



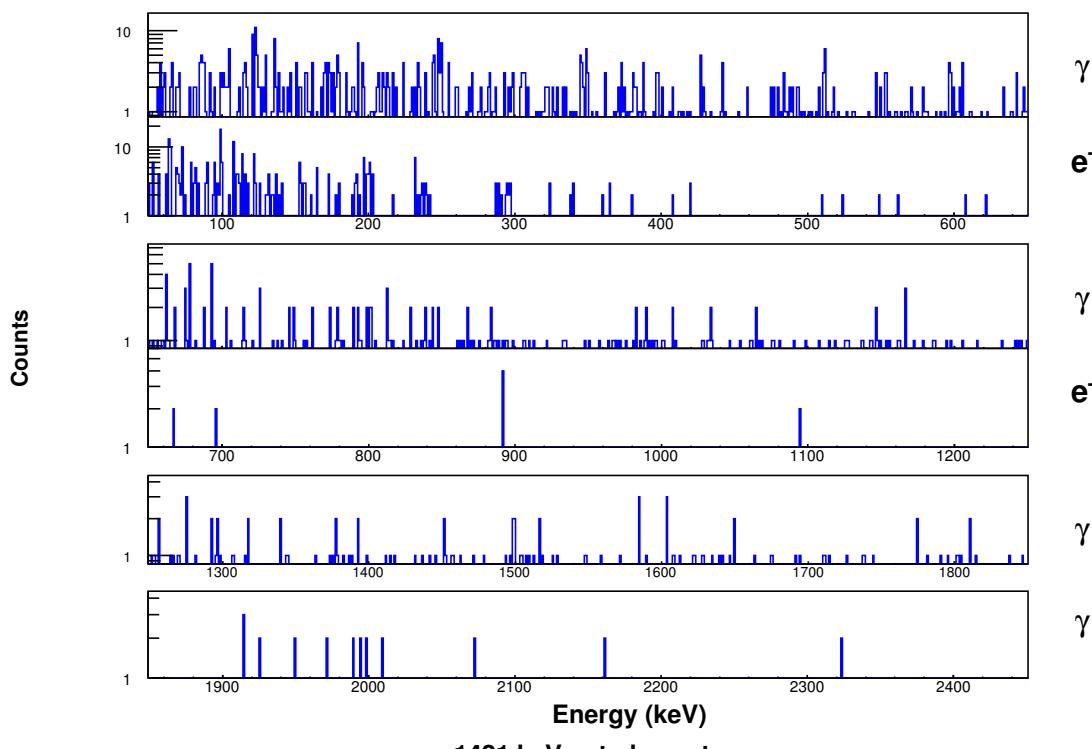
1346 keV gated spectra



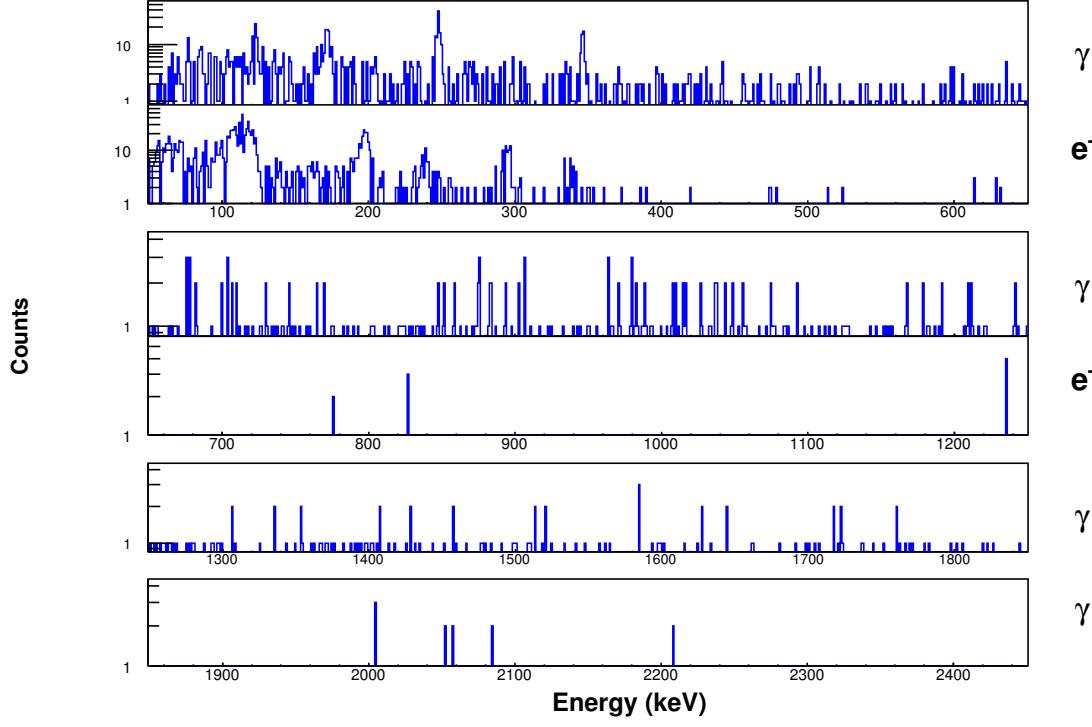
1370 keV gated spectra



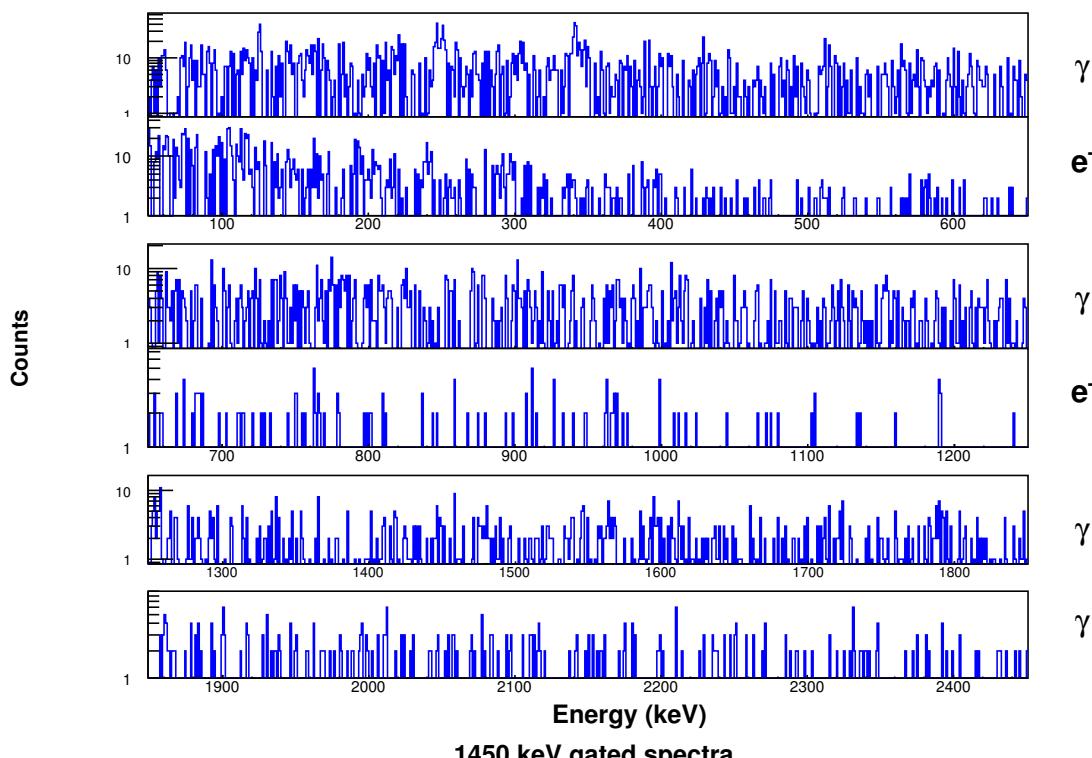
1390 keV gated spectra



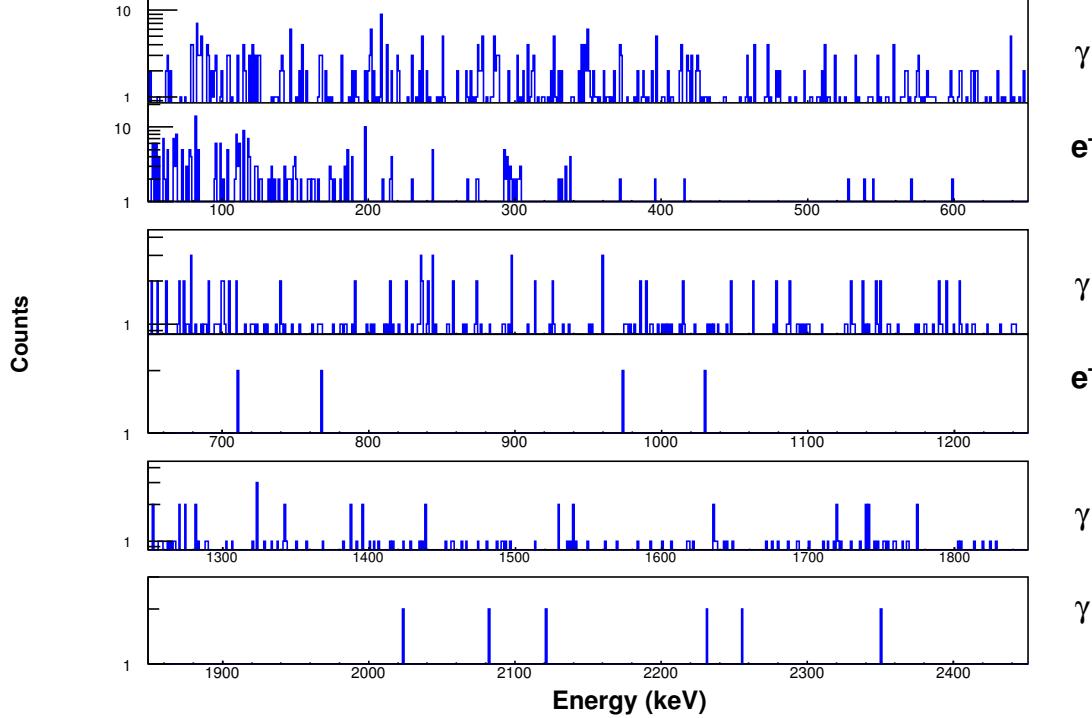
1421 keV gated spectra



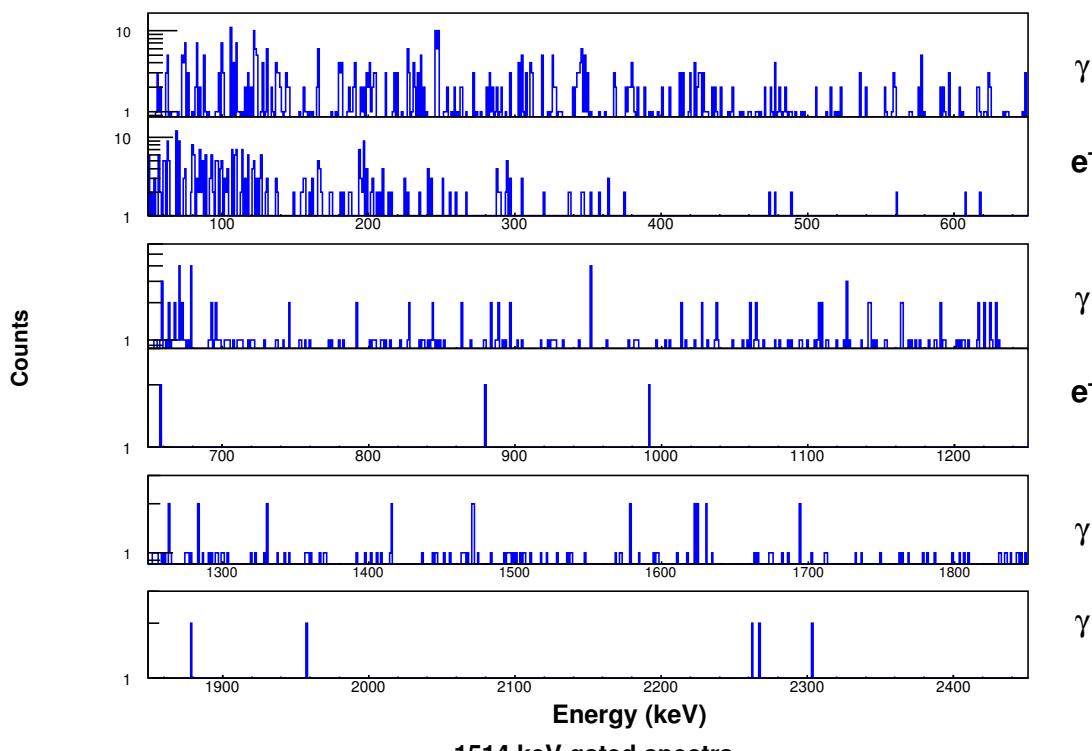
1431 keV gated spectra



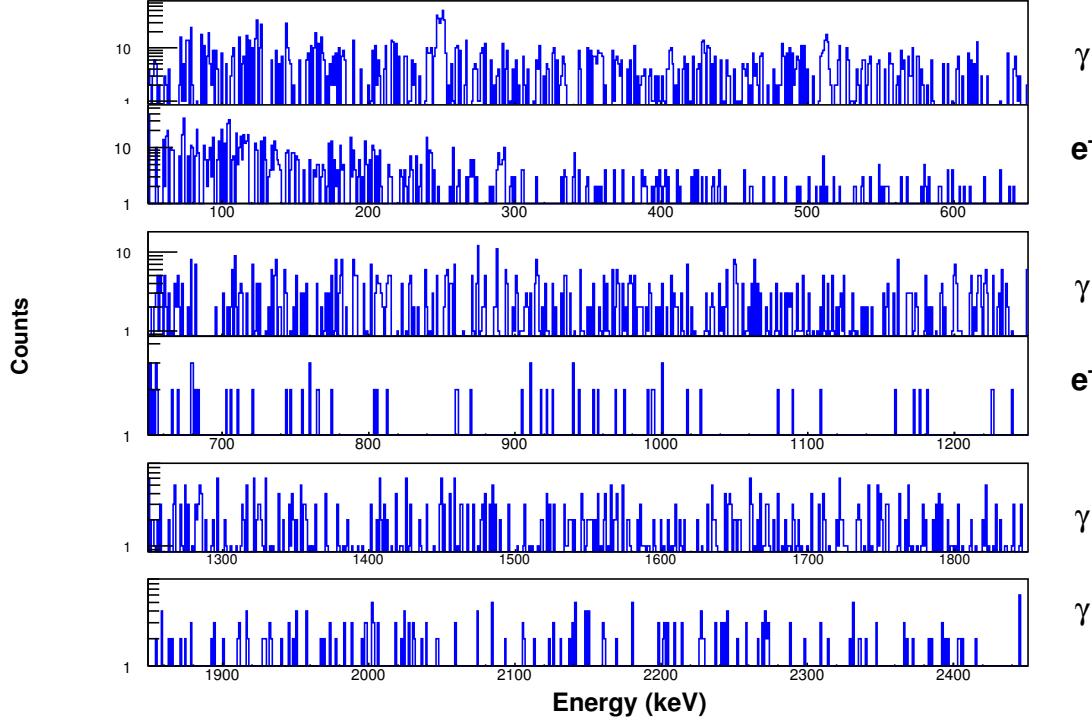
1450 keV gated spectra



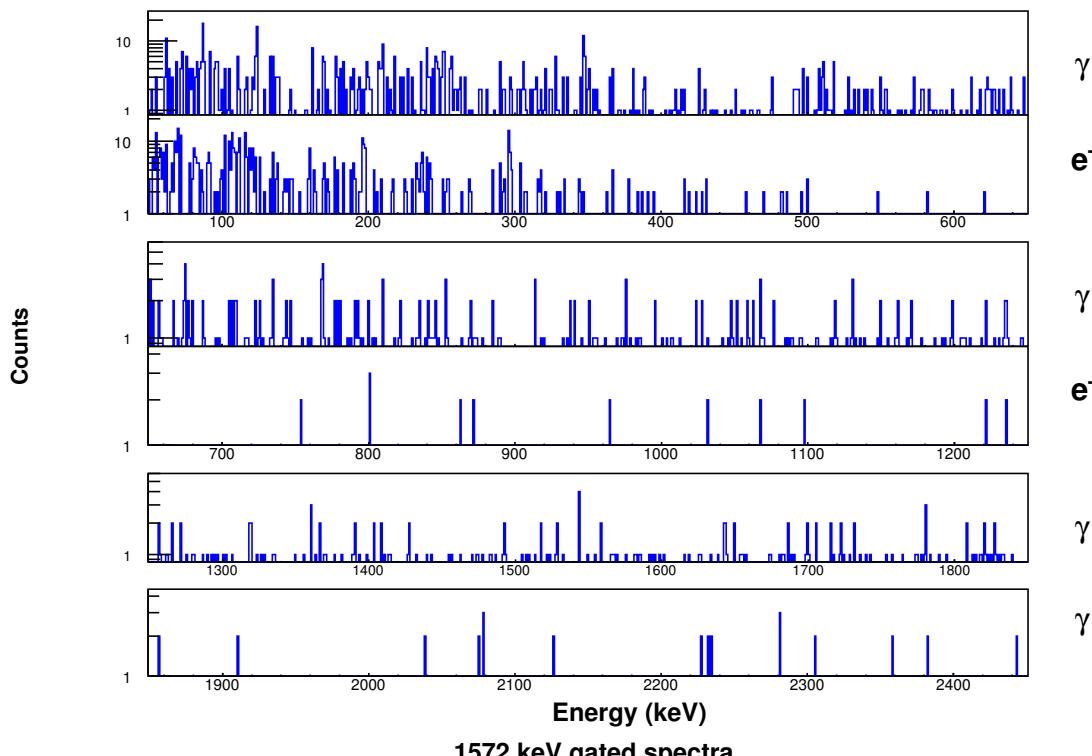
1488 keV gated spectra



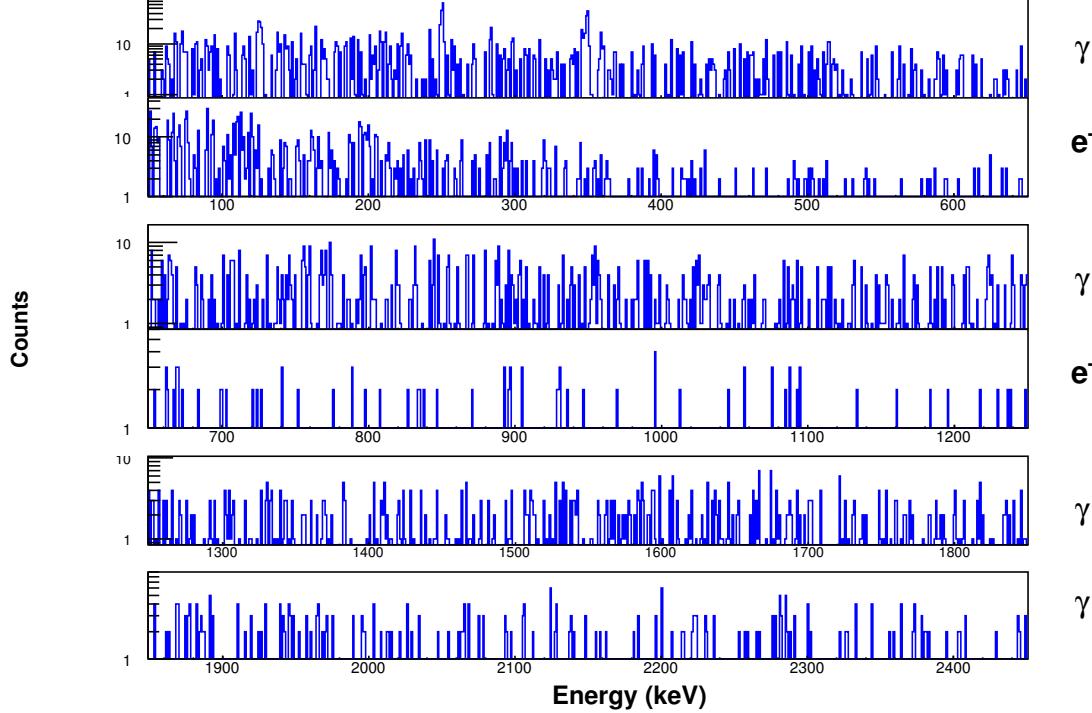
1514 keV gated spectra



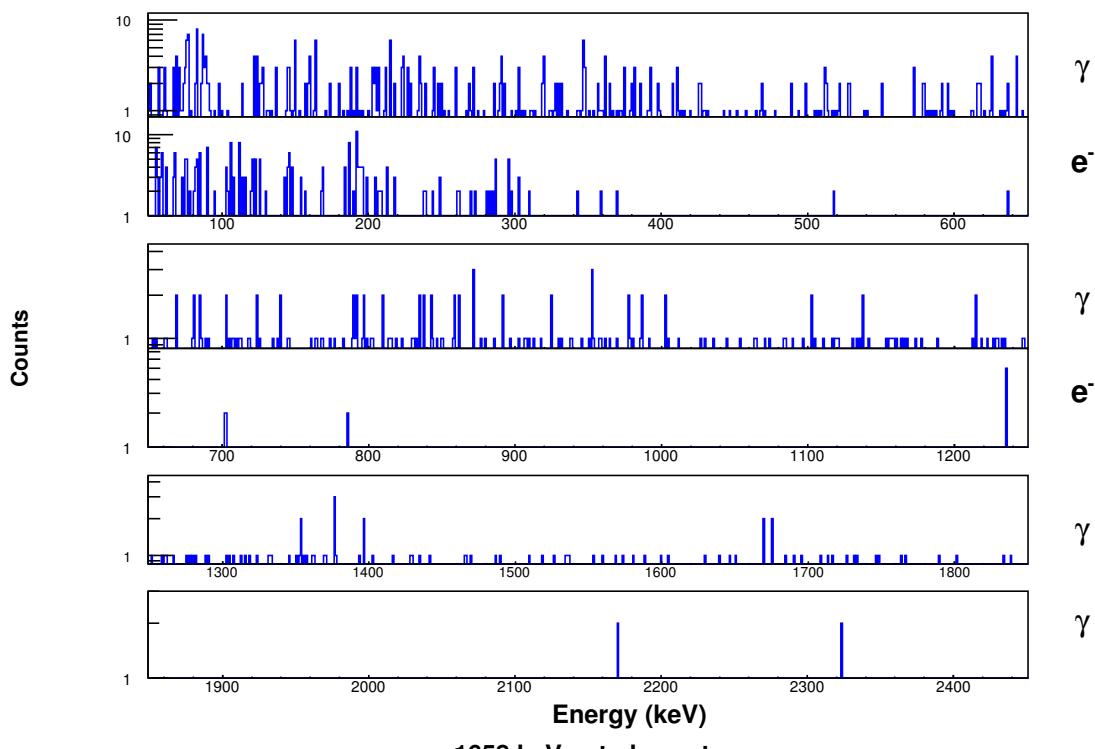
1527 keV gated spectra



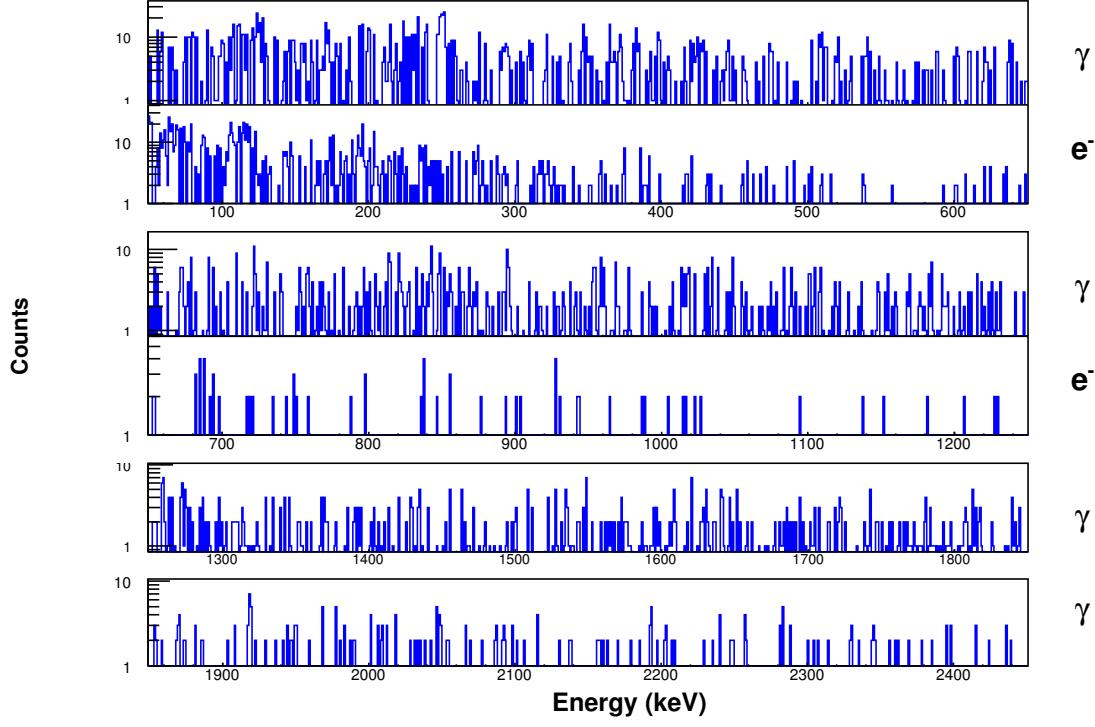
1572 keV gated spectra



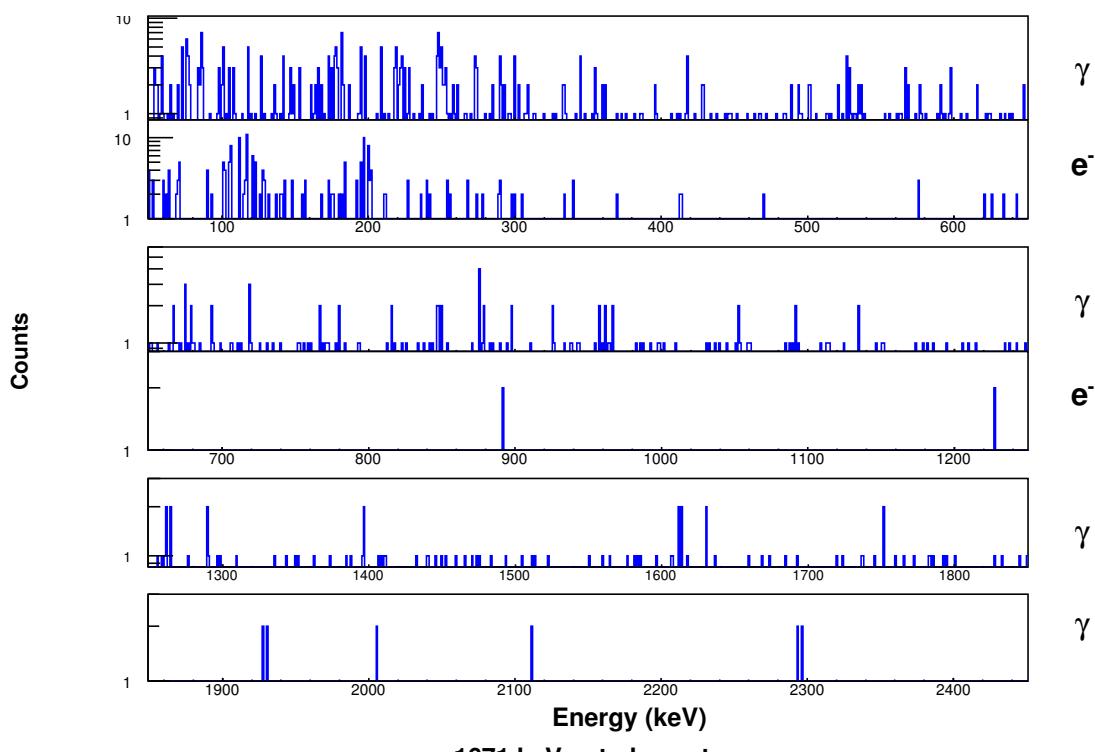
1589 keV gated spectra



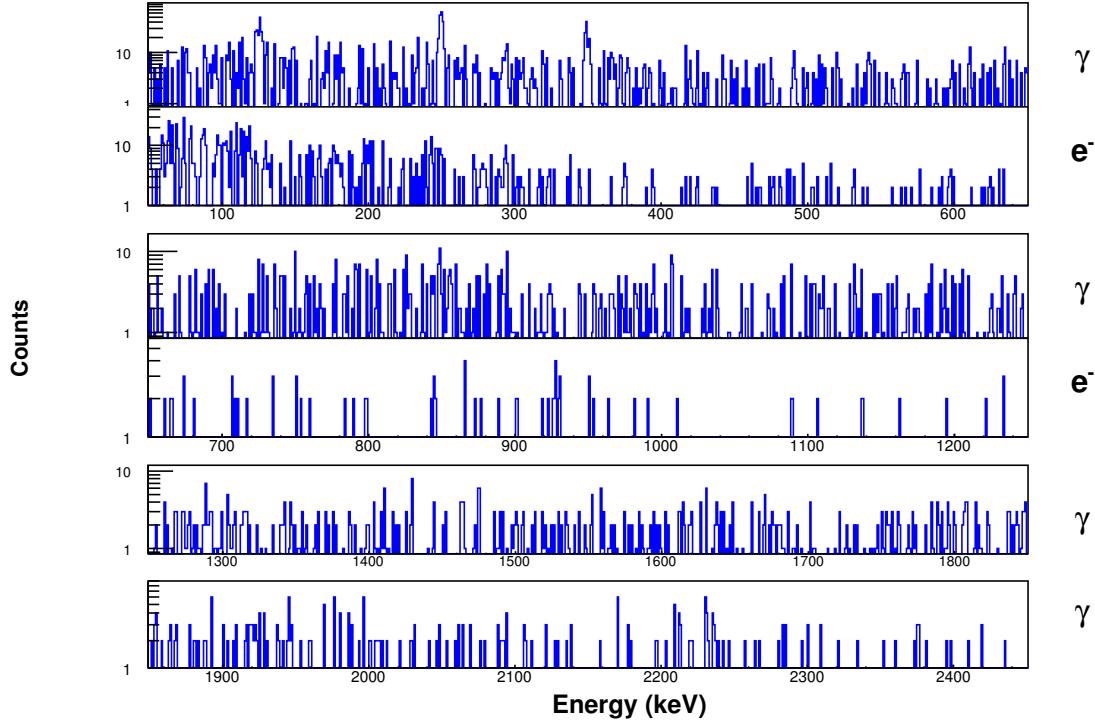
1658 keV gated spectra



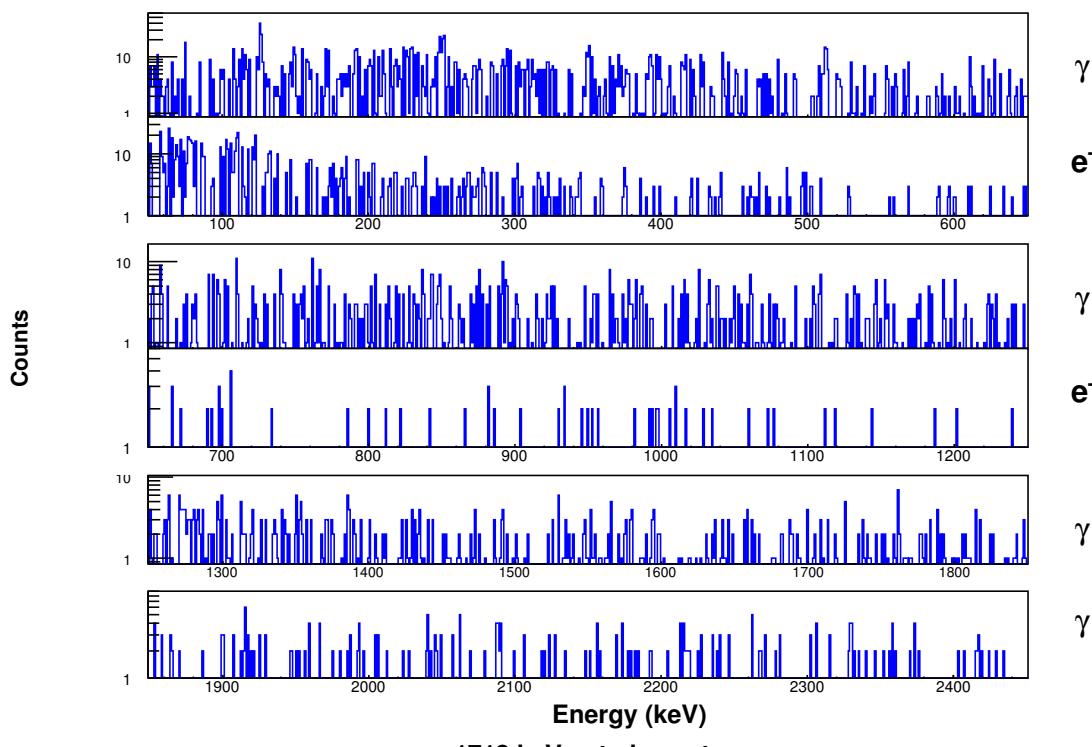
1669 keV gated spectra



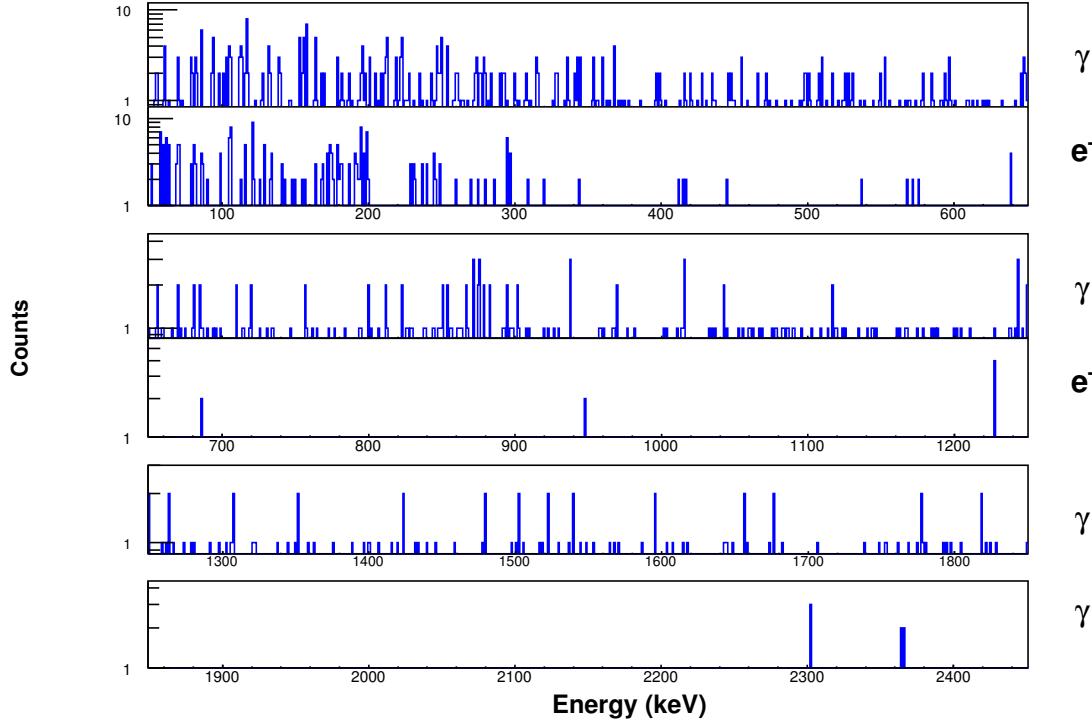
1671 keV gated spectra



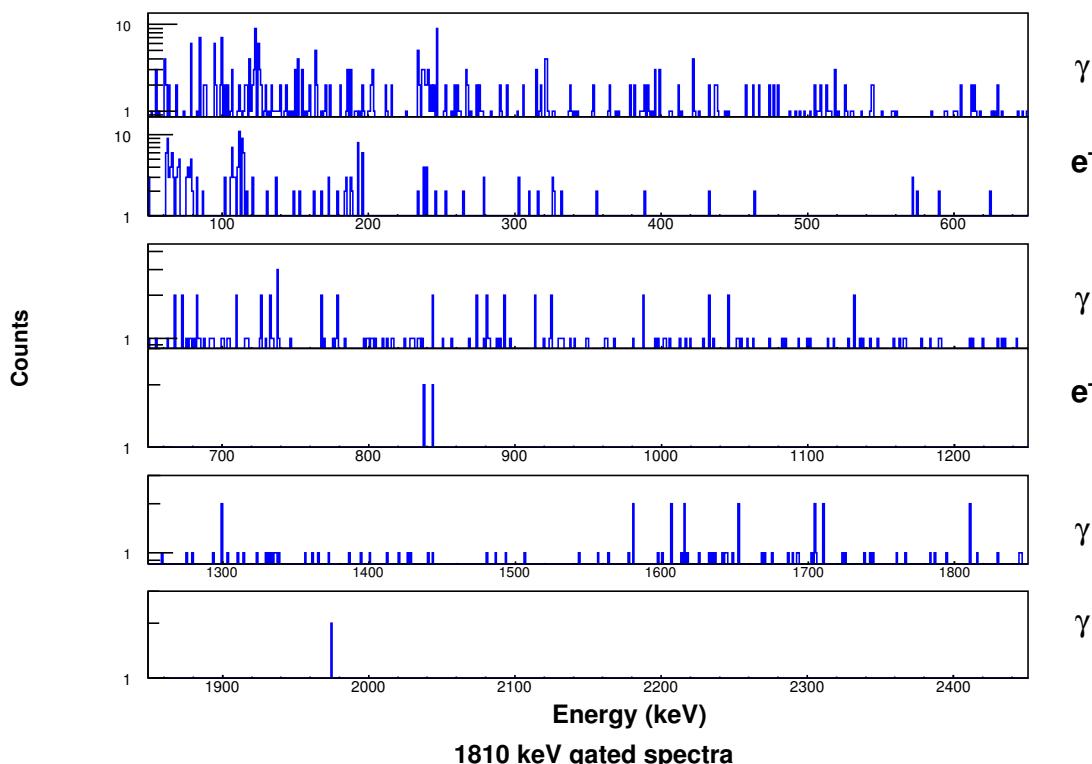
1700 keV gated spectra



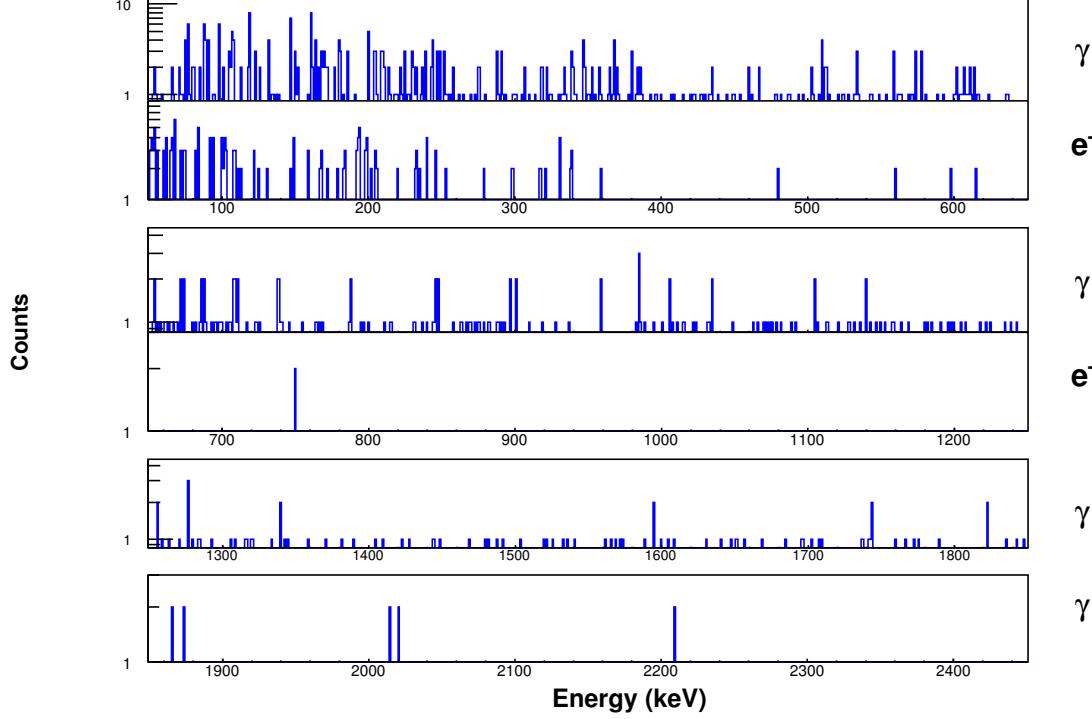
1713 keV gated spectra



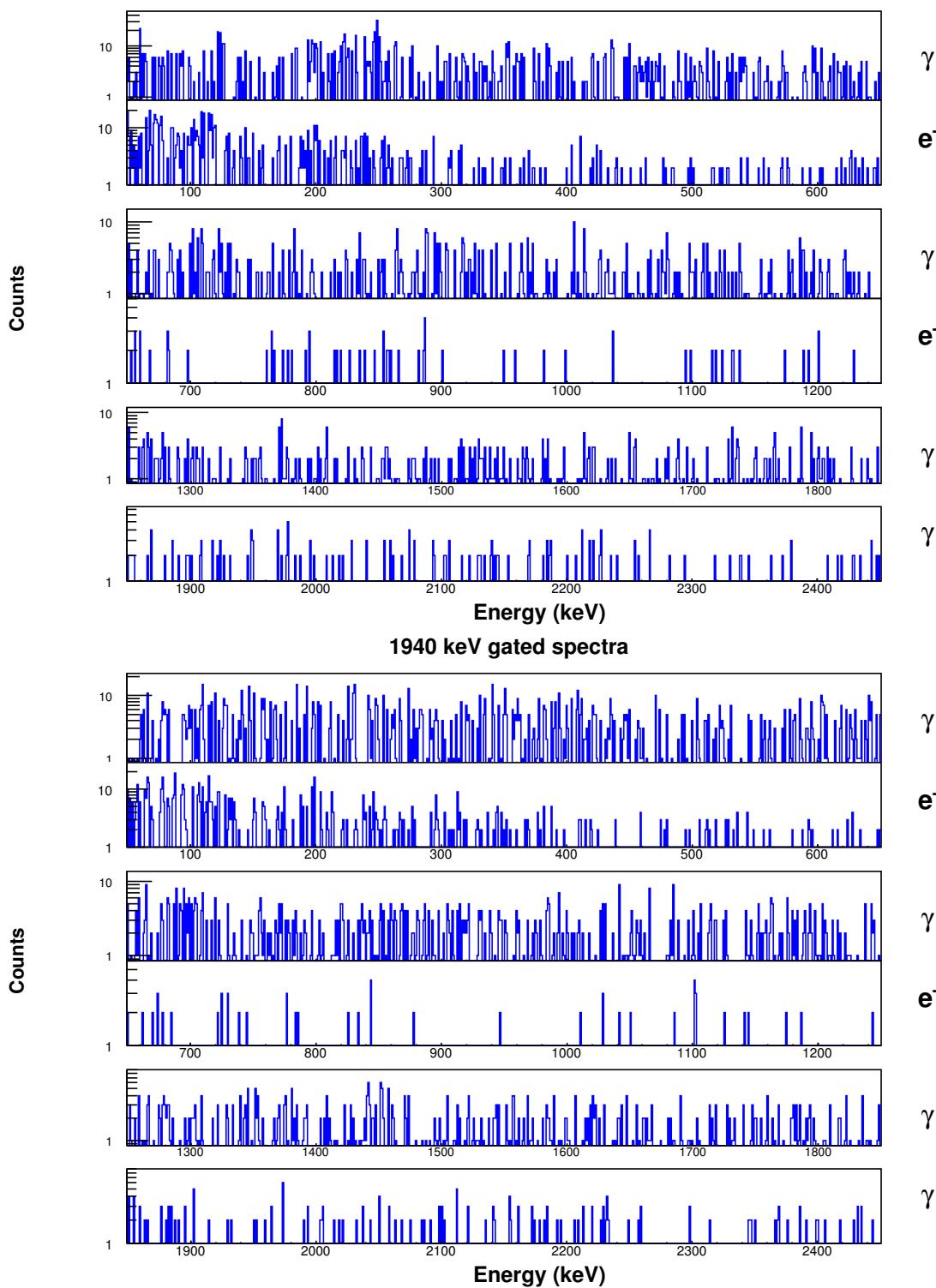
1788 keV gated spectra



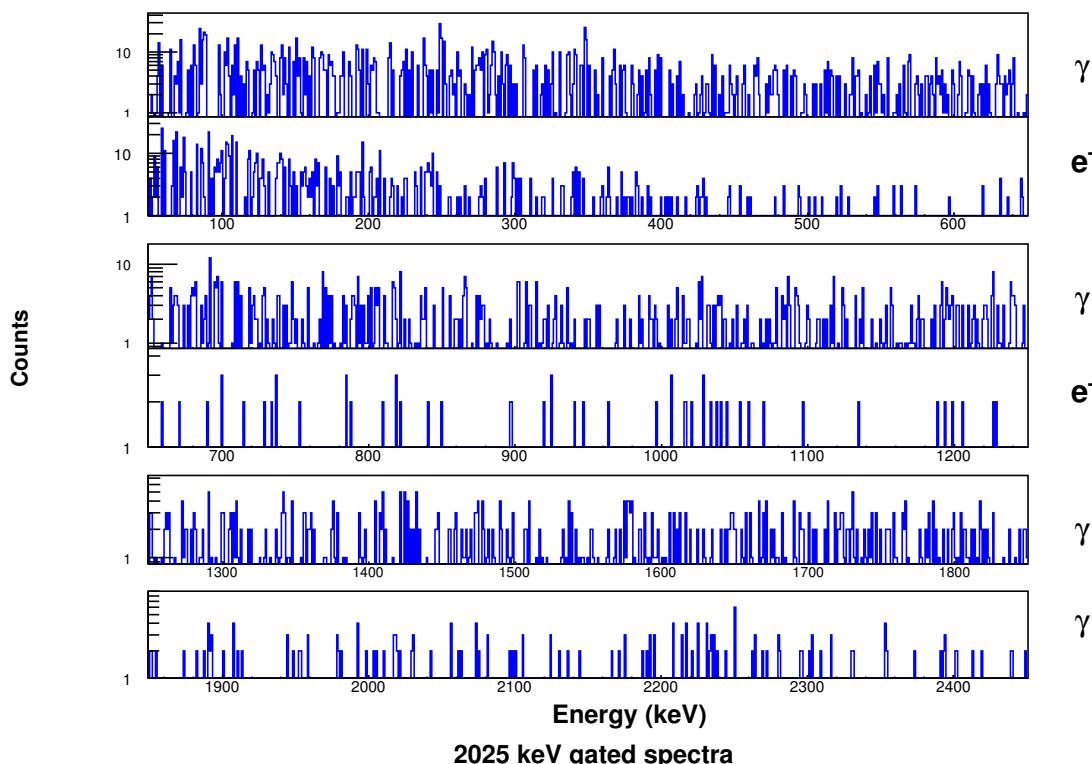
1810 keV gated spectra



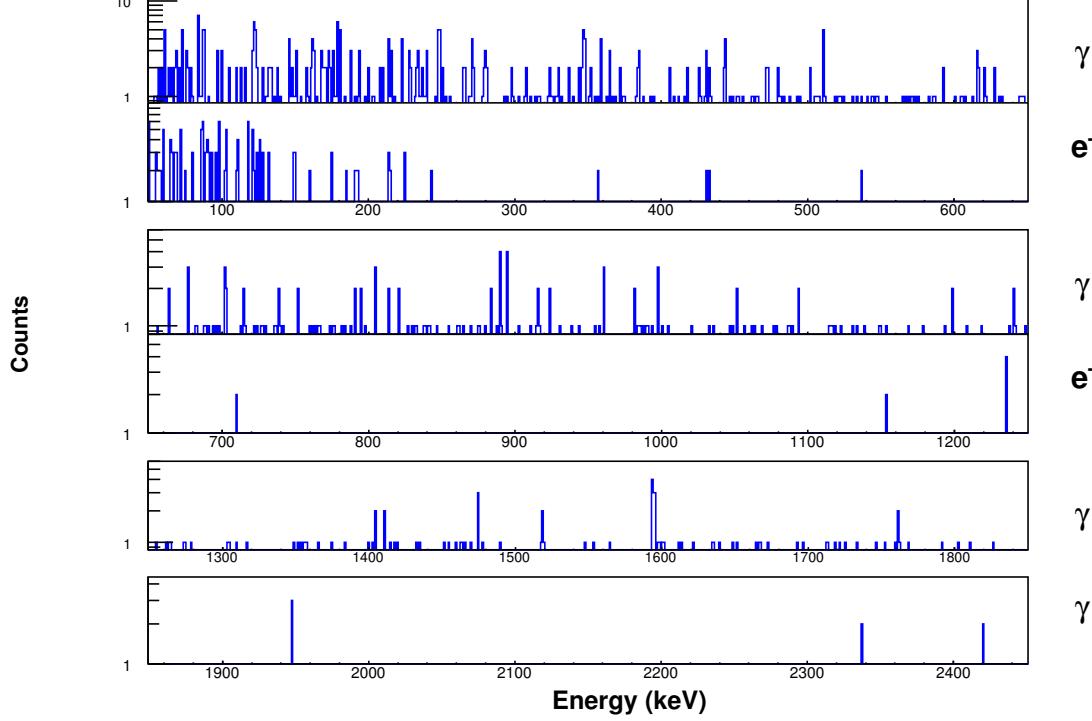
1895 keV gated spectra



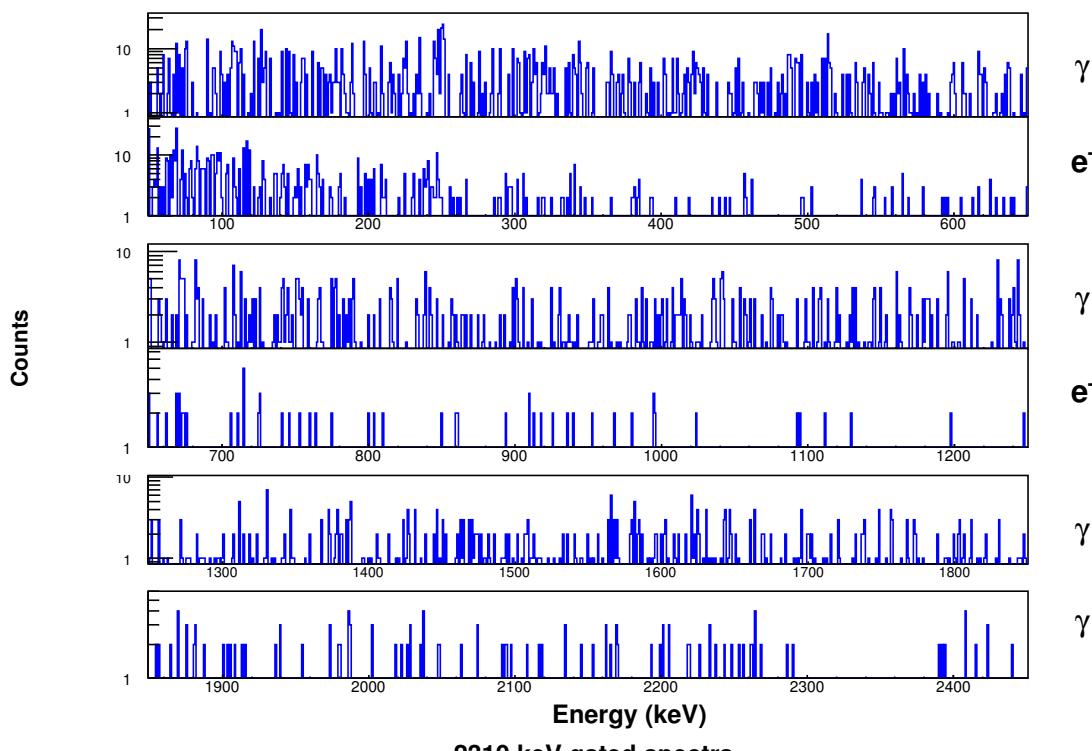
2007 keV gated spectra



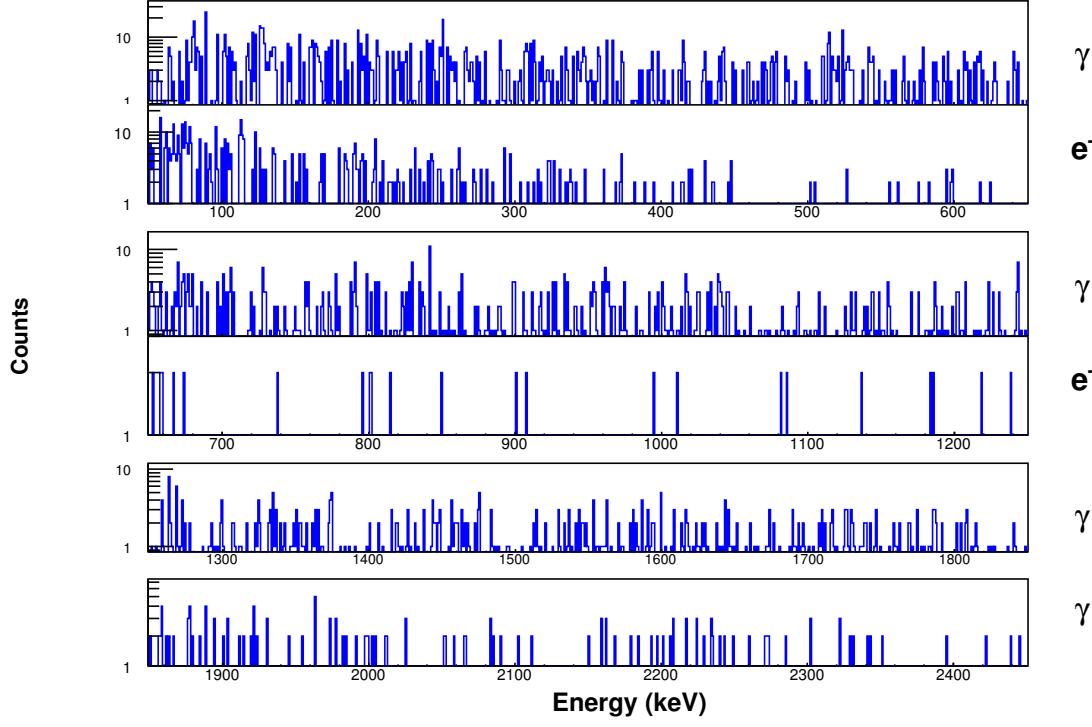
2025 keV gated spectra



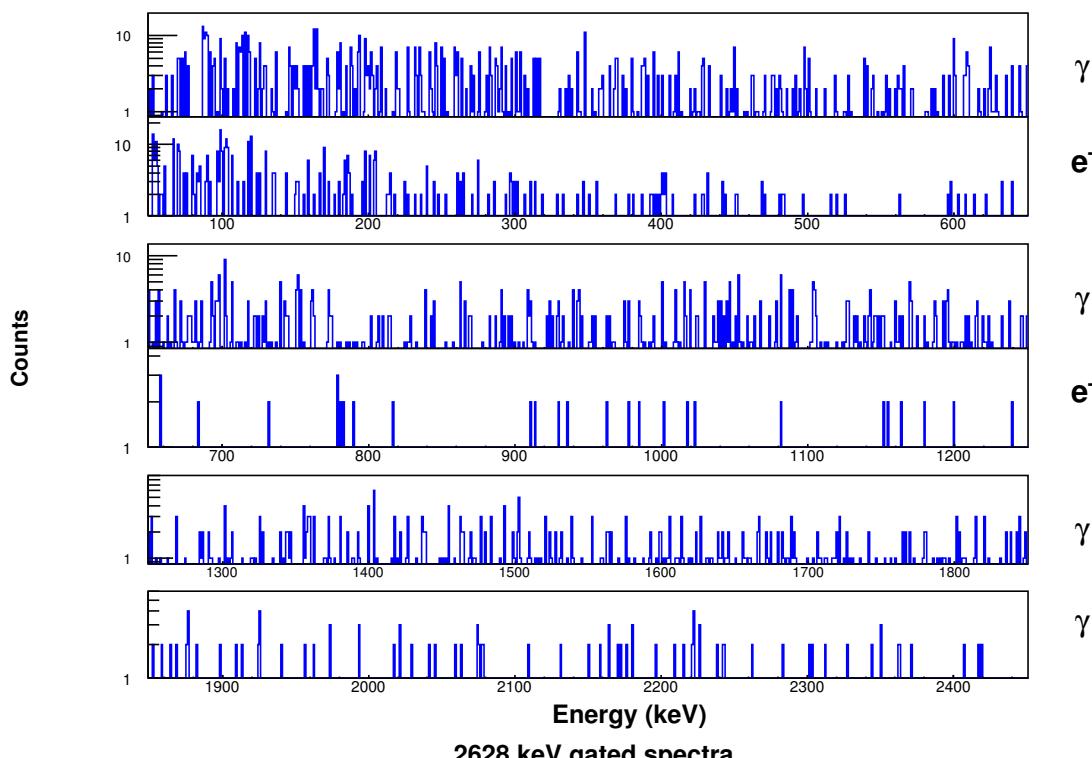
2173 keV gated spectra



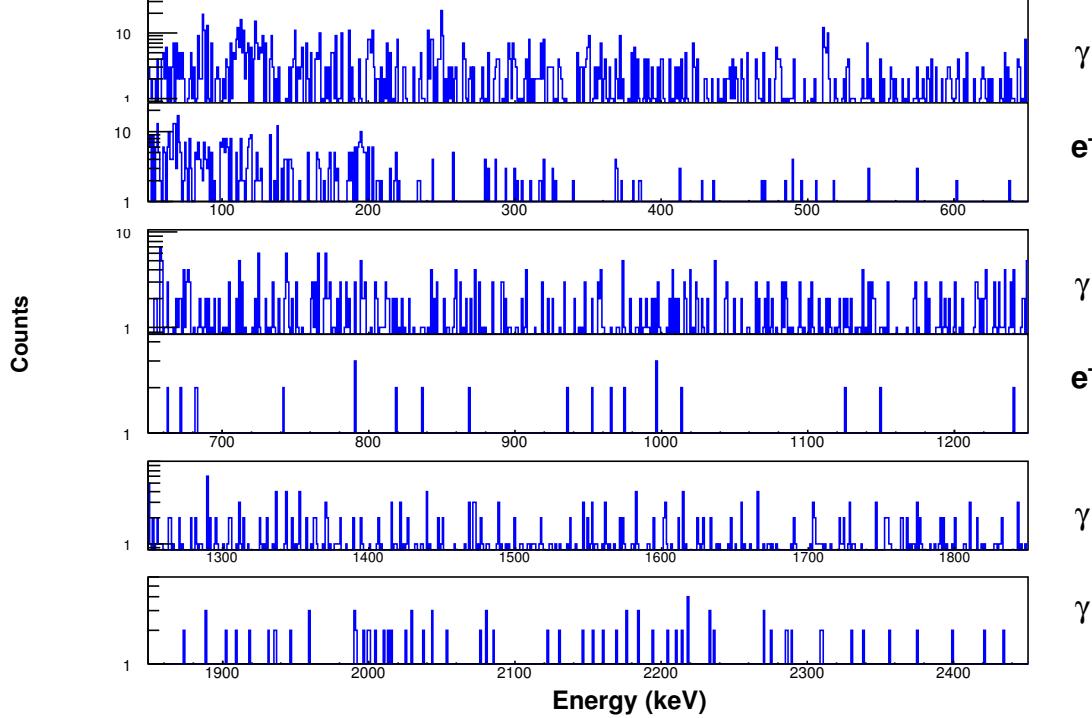
2310 keV gated spectra



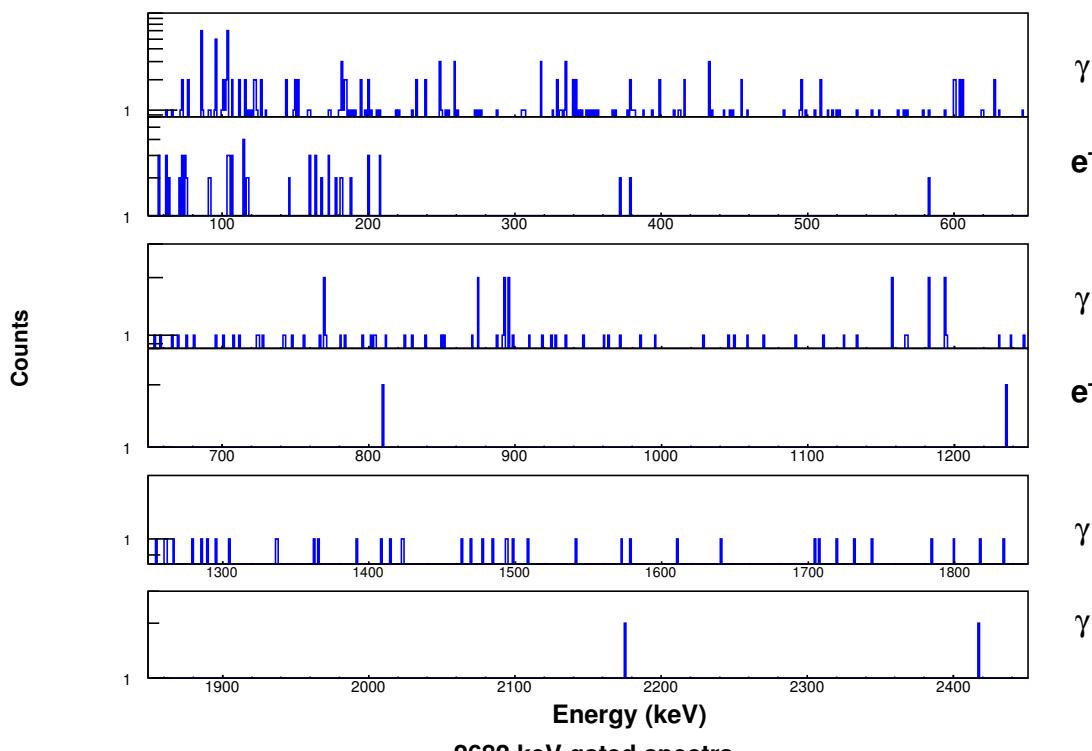
2435 keV gated spectra



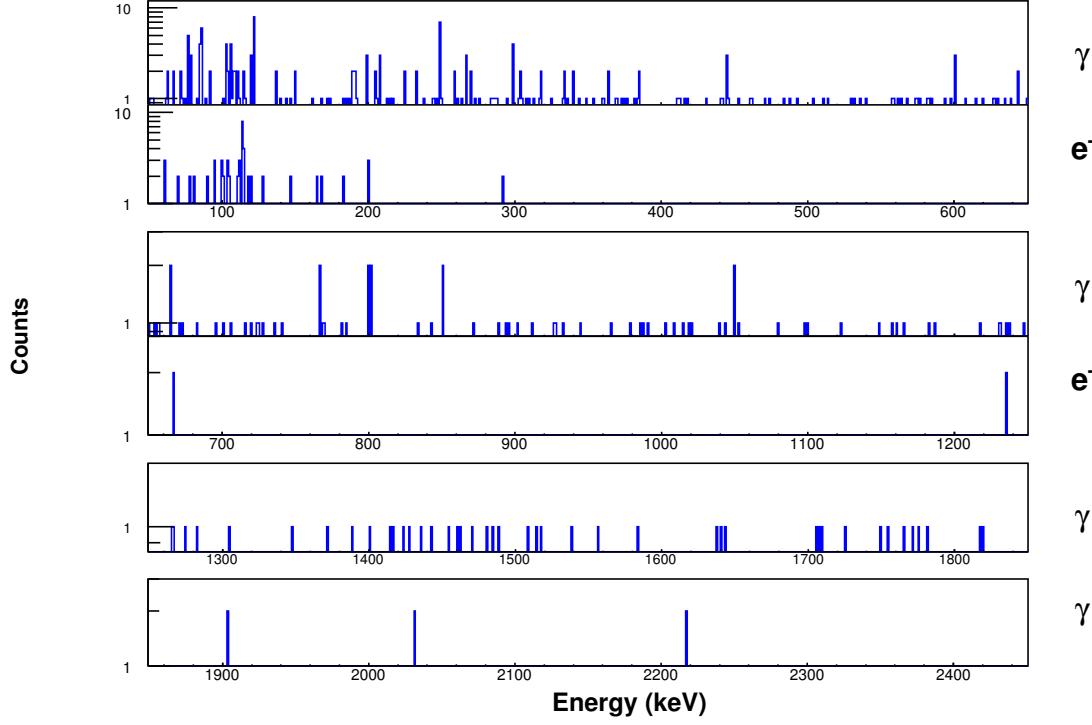
2628 keV gated spectra



2648 keV gated spectra



2682 keV gated spectra



APPENDIX B

^{156}GD GATED SPECTRA

The following is a compendium of spectra from the gates on the ^{156}Gd data. A table with the gates and the corresponding background gates used in subtraction is listed. Unless noted, the energy listed is the center of the gate, with a range of ± 1.5 keV.

TABLE B.1

LIST OF ^{156}GD GATES

E_{gate} (keV)	E_{bgd} (keV)	Description
89	100	$2_{gs}^+ \rightarrow 0_{gs}^+$
110.6	120	$5^+ \rightarrow 4^+$ in $K^\pi = 4^+$ band
156.7	175	$7^+ \rightarrow 6^+$ in $K^\pi = 4^+$ band
199.2	215	$4_{gs}^+ \rightarrow 2_{gs}^+$
227.4	250	$7^- \rightarrow 7^+$ with 7^+ in $K^\pi = 4^+$ band
287.6	310	$6_\gamma^+ \rightarrow 4_\gamma^+$
296.5	310	$6_{gs}^+ \rightarrow 4_{gs}^+$
322.1	350	$8^- \rightarrow 6^-$ in $K^{pi} = 1^-$ octopole-vibrational band
357.7	370	$4^+ \rightarrow 2_{gs}^+$ with 4^+ in $K^\pi = 4^+$ band
380.4	390	$8_{gs}^+ \rightarrow 6_{gs}^+$
399.9	410	$9_\gamma^+ \rightarrow 7_\gamma^+$
451	465	$10_{gs}^+ \rightarrow 8_{gs}^+$

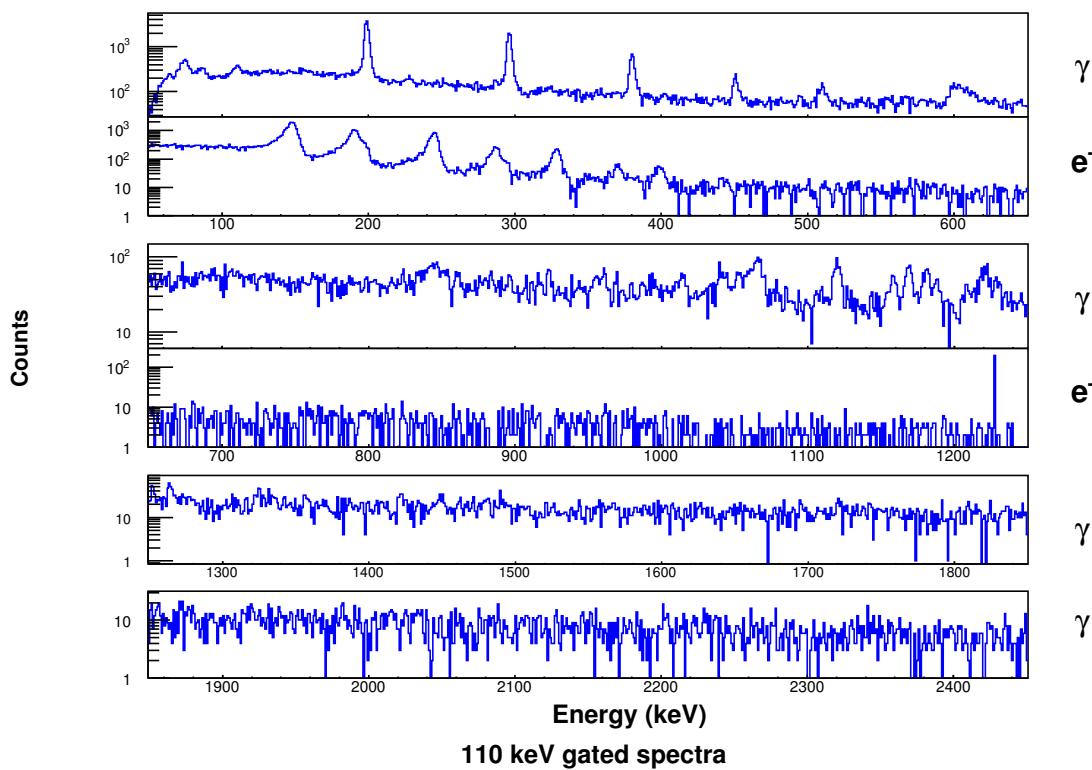
TABLE B.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
568.3	640	$2^- \rightarrow 1^-$ with 2^- in $K^\pi = 2^-$ and 1^- in $K^\pi = 0^-$ octopole vibrational bands.
789.7	795	$3^-+ \rightarrow 4^-$ with 3^+ in $K^\pi = 1^+$ band and 4^- in $K^\pi = 1^-$ octopole vibrational band
823.6	900	$5^- \rightarrow 6_{gs}^+$ with 5^- in $K^\pi = 1^-$ octopole vibrational band
875.9	900	$4_3^+ \rightarrow 6_{gs}^+$ with 4^+ in 2 nd excited 0 ⁺ band
883.6	900	$8_2^+ \rightarrow 8_{gs}^+$ with 8^+ in 1 st excited 0 ⁺ band and $7_\gamma^+ \rightarrow 8_{gs}^+$
922.6	1000	$5_\gamma^+ \rightarrow 6_{gs}^+$
943.6	980	$7^+ \rightarrow 8_{gs}^+$ with 7^+ in $K^\pi = 4^+$ band
954.5	1000	$6_2^+ \rightarrow 6_{gs}^+$ with 6^+ in 1 st excited 0 ⁺ band
960.2	980	$3_\gamma^+ \rightarrow 4_\gamma^+$
969.2	980	$2_3^+ \rightarrow 4_{gs}^+$ with 2^+ in 2 nd excited 0 ⁺ band
988	1000	$3^- \rightarrow 4_{gs}^+$ with 3^- in $K^\pi = 1^-$ octopole vibrational band
992.8	1000	$9^- \rightarrow 8_{gs}^+$ with 9^- in $K^\pi = 1^-$ octopole vibrational band
1010.7	1100	$4_2^+ \rightarrow 4_{gs}^+$ with 4^+ in 1 st excited 0 ⁺ band
1013.3	1100	Likely a transition into 10_{gs}^+
1039.4	1100	$5^+ \rightarrow 6_{gs}^+$ with 5^+ in $K^\pi = 4^+$ band
1044.5	1090	$8^+\gamma \rightarrow 8_{gs}^+$
1052.2	1100	$7^- \rightarrow 6_{gs}^+$ with 7^- in $K^\pi = 1^-$ band
1058.5	1090	$6^+\gamma \rightarrow 6_{gs}^+$
1067.5	1100	$4^+\gamma \rightarrow 4_{gs}^+$ and $2^+\gamma \rightarrow 2_{gs}^+$

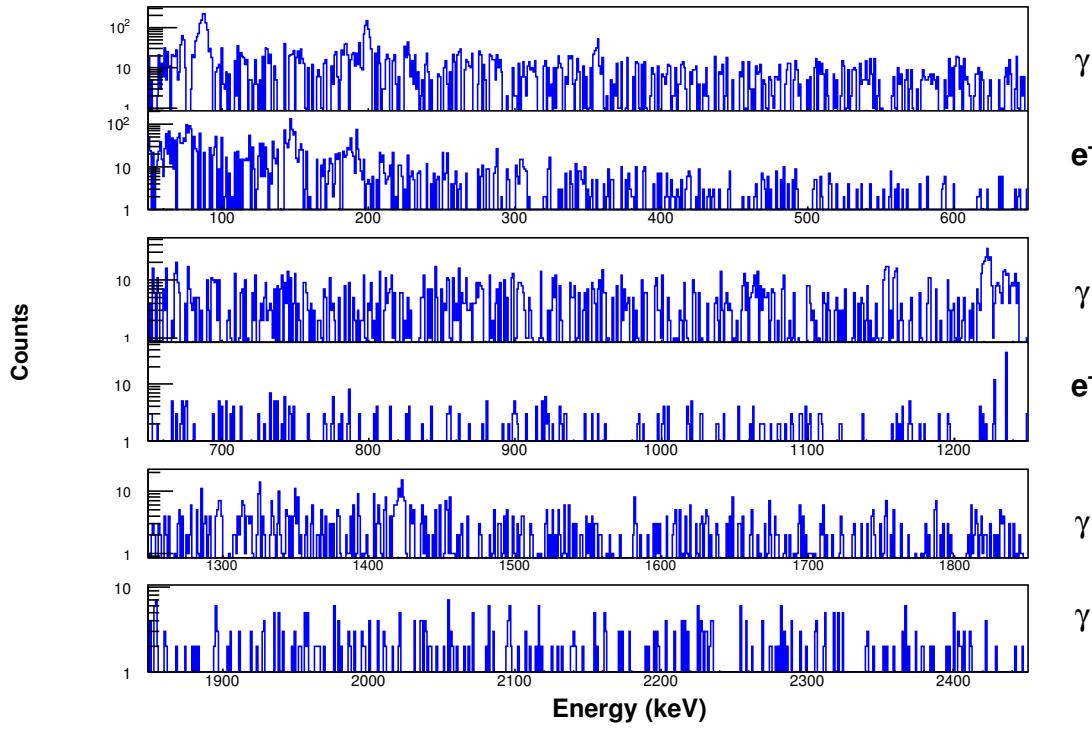
TABLE B.1 (CONTINUED)

E_{gate} (keV)	E_{bgd} (keV)	Description
1119.2	1150	$6^- \rightarrow 6_{gs}^+$ with 6^- in $K^\pi = 1^-$ octopole vibrational band
1156.8	1200	$1^- \rightarrow 2_{gs}^+$ with 1^- in $K^\pi = 1^-$ octopole vibrational band and $2_2^+ \rightarrow 0_{gs}^+$ with 2^+ in 1 st excited 0 ⁺ band
1168.2	1200	$2_3^+ \rightarrow 2_{gs}^+$ with 2^+ in 2 nd excited 0 ⁺ band and $0_3^+ \rightarrow 0_{gs}^+$ gamma-equivalent
1180.6	1200	$6_3^+ \rightarrow 6_{gs}^+$ with 2^+ in 2 nd excited 0 ⁺ band
1188.3	1200	$3^- \rightarrow 2_{gs}^+$ with 1^- in $K^\pi = 1^-$ octopole vibrational band
1221.7	1240	$4^+ \rightarrow 4_{gs}^+$ with 4^+ in $K^\pi = 4^+$ band
1250	1300	$3^- \rightarrow 4_{gs}^+$ with 3^- in $K^\pi = 0^-$ octopole vibrational band
1265.1	1300	$7_\gamma^+ \rightarrow 6_{gs}^+$ and $8_2^+ \rightarrow 6_{gs}^+$ with 8^+ in 1 st excited 0 ⁺ band
1284.7	1300	$9_\gamma^+ \rightarrow 6_{gs}^+$
1325.8	1400	$7^+ \rightarrow 6_{gs}^+$ with 7^+ in $K^\pi = 4^+$ band
1334.9	1348	$5^+ \rightarrow 4_{gs}^+$ with 5^+ in $K^\pi = 4^+$ band
1355.9	1400	$6_\gamma^+ \rightarrow 4_{gs}^+$
1479.7	1500	$10_\gamma^+ \rightarrow 8_{gs}^+$ and $6_3^+ \rightarrow 4_{gs}^+$ with 6^+ in 2 nd excited 0 ⁺ band
1494	1500	No known transitions in agreement, but appears in coincidence with ground state band transitions.
1605	1630	$4_4^+ \rightarrow 4_{gs}^+$ with 4^+ in 3 rd excited 0 ⁺ band
1731.1	1750	$4_4^+ \rightarrow 4_{gs}^+$ with 4^+ in $K^\pi = 2^+$ band

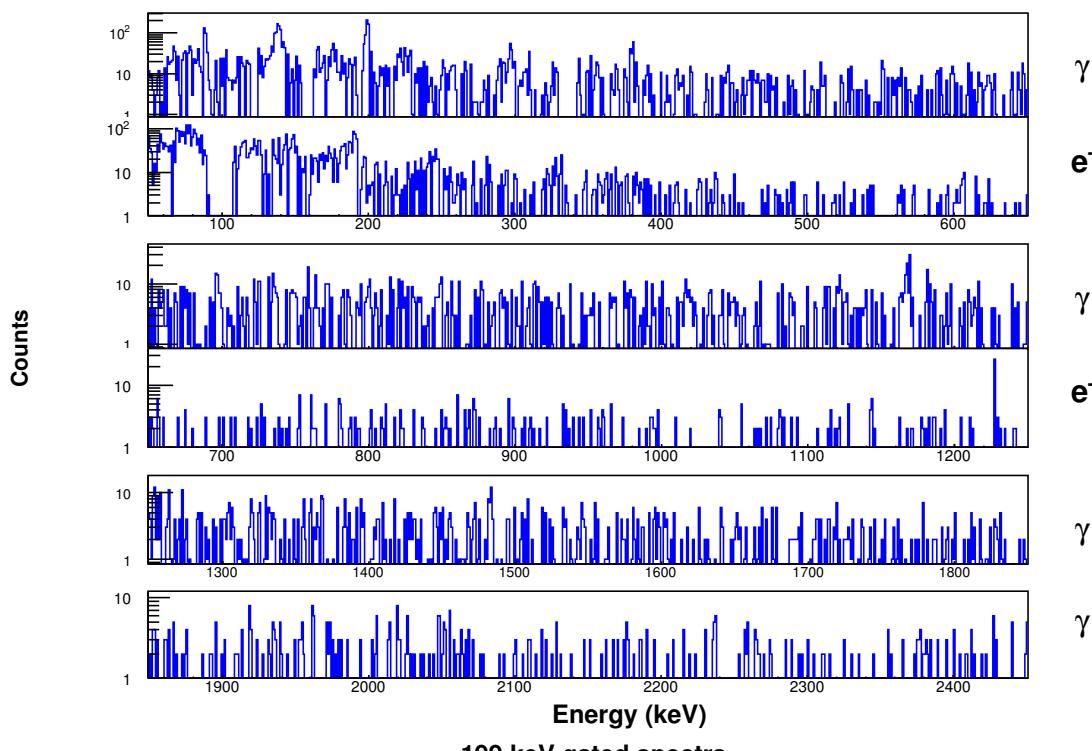
89 keV gated spectra



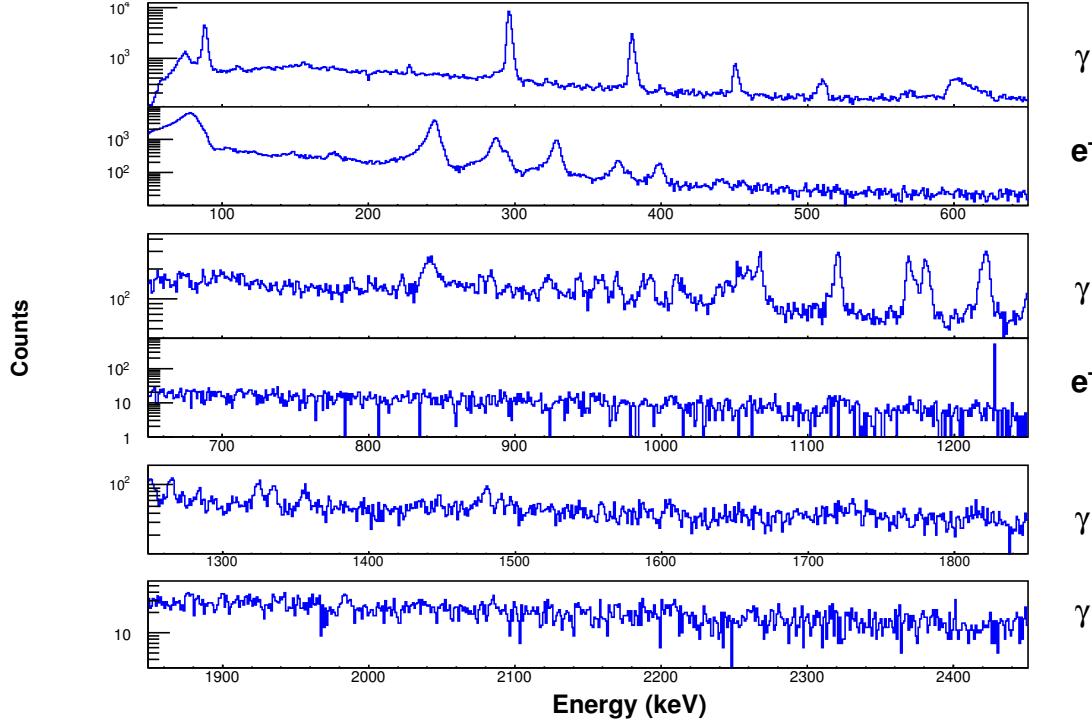
110 keV gated spectra



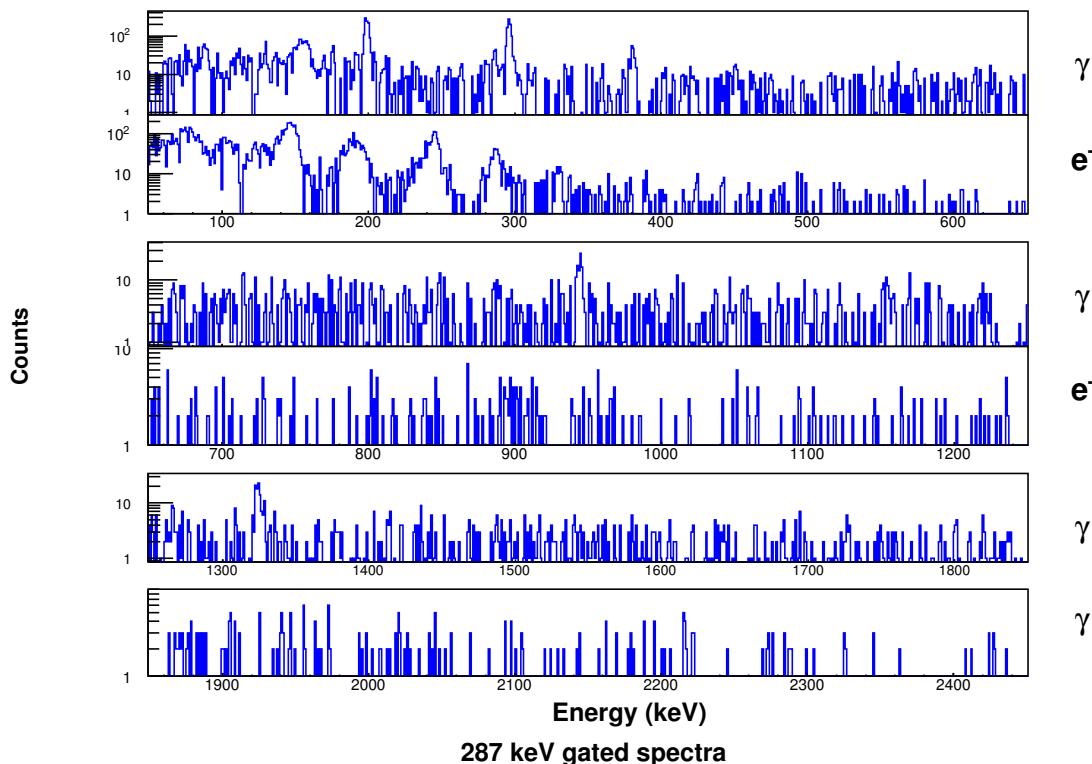
156 keV gated spectra



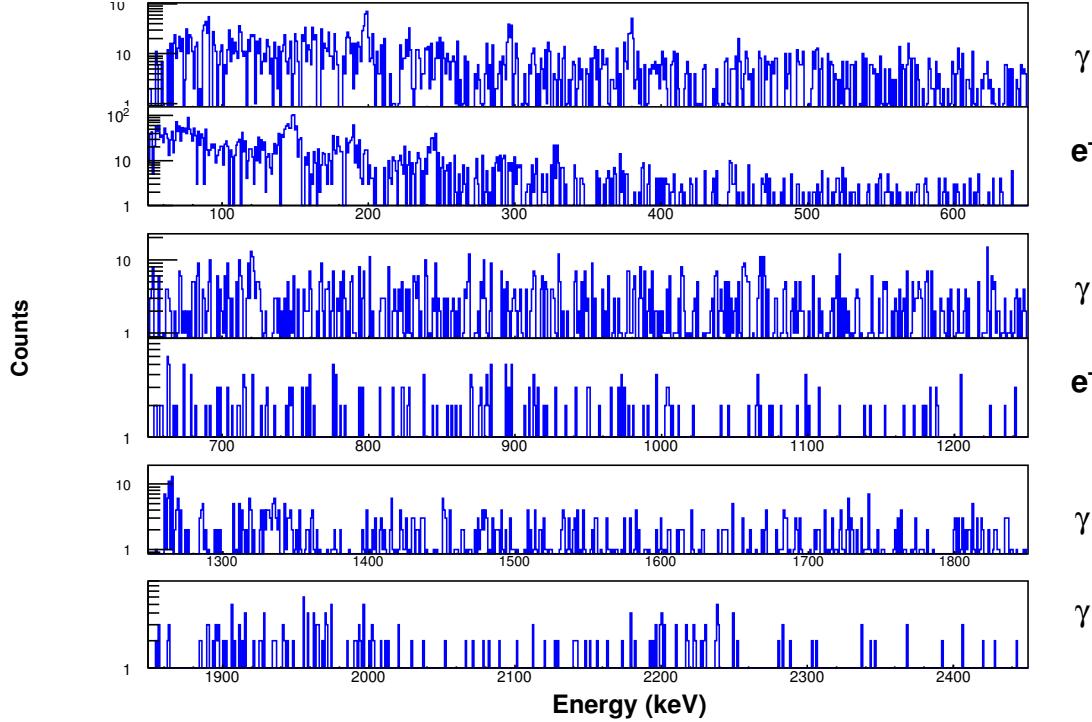
199 keV gated spectra

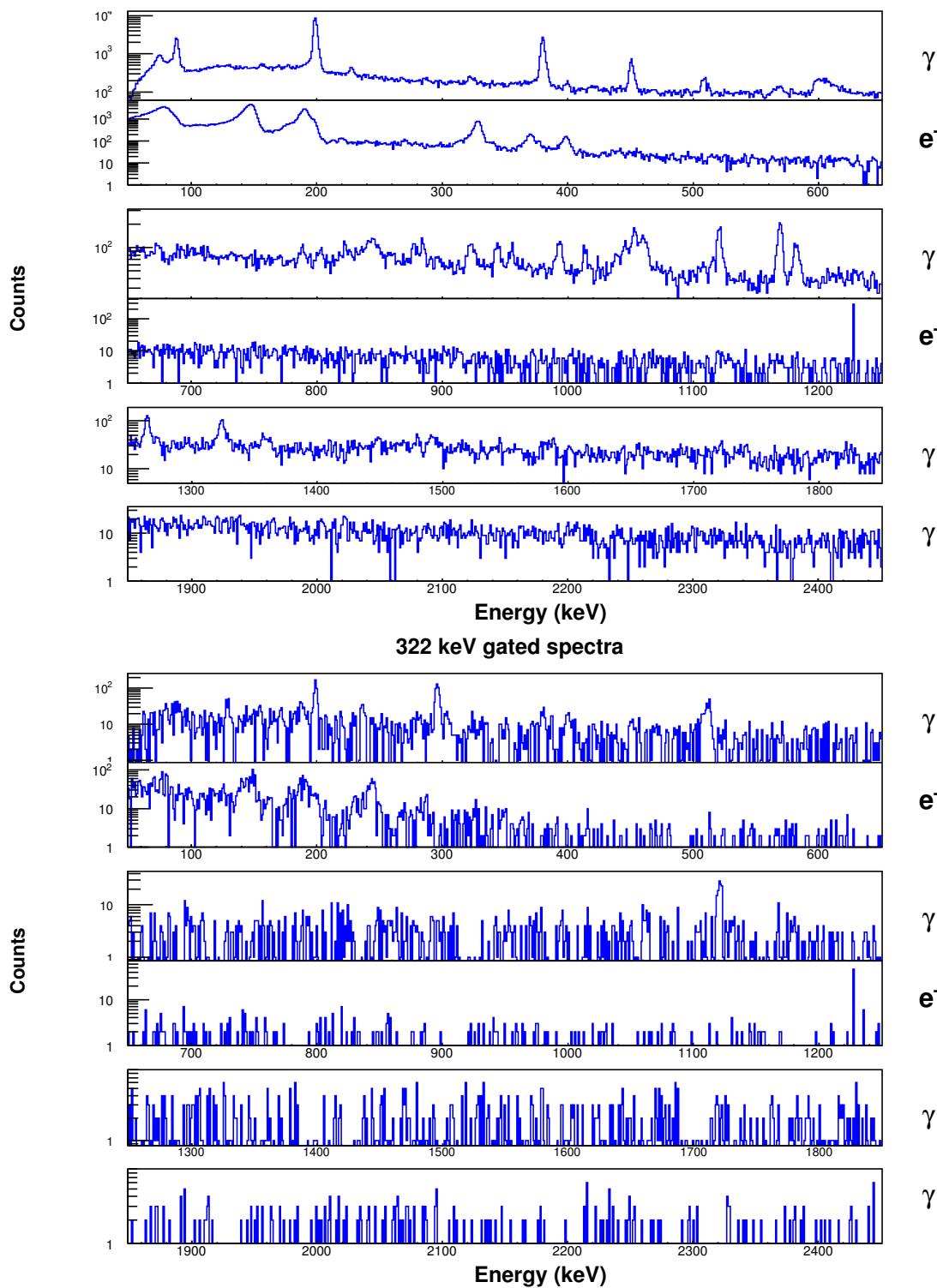


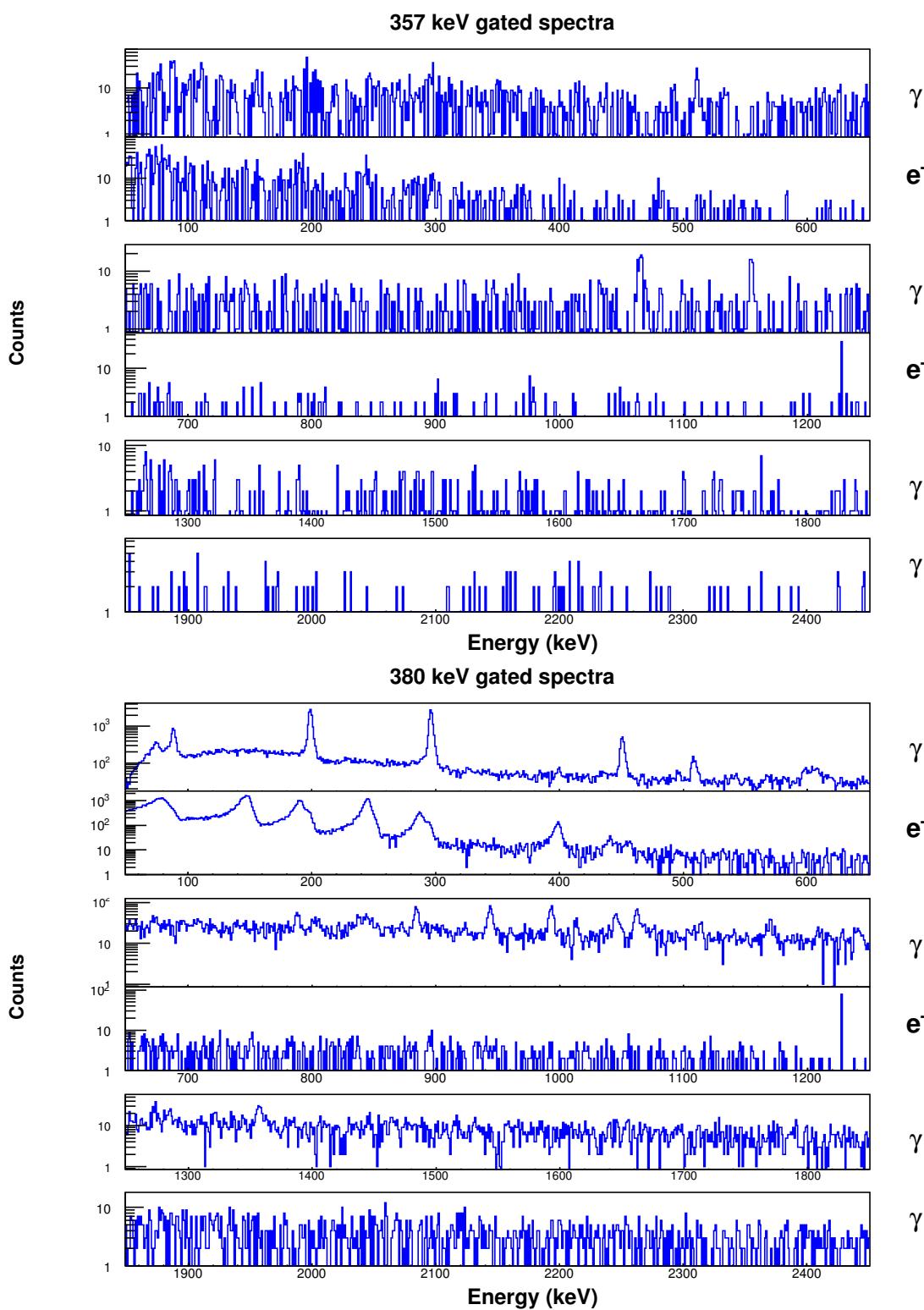
227 keV gated spectra



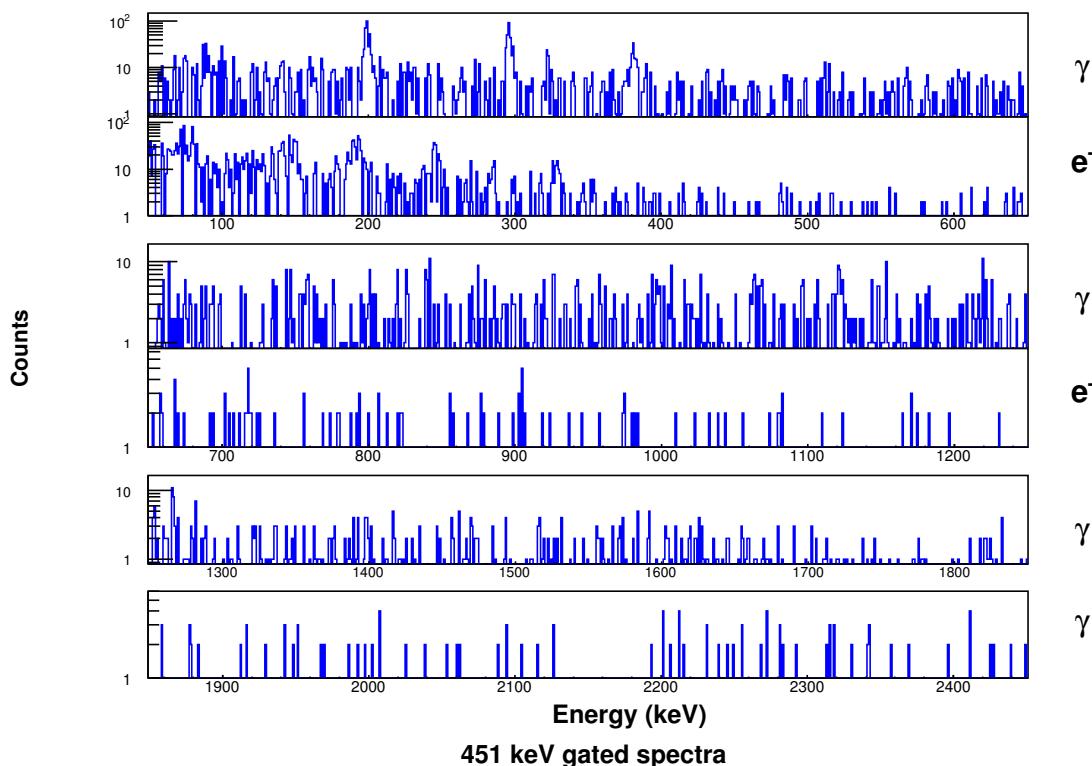
287 keV gated spectra



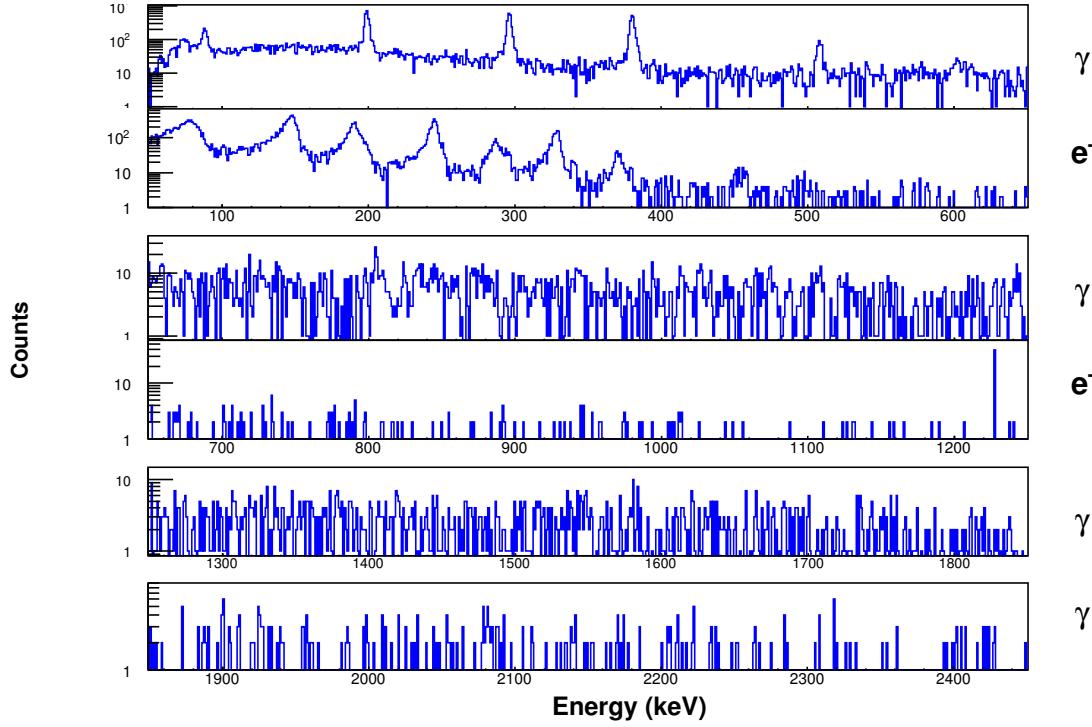


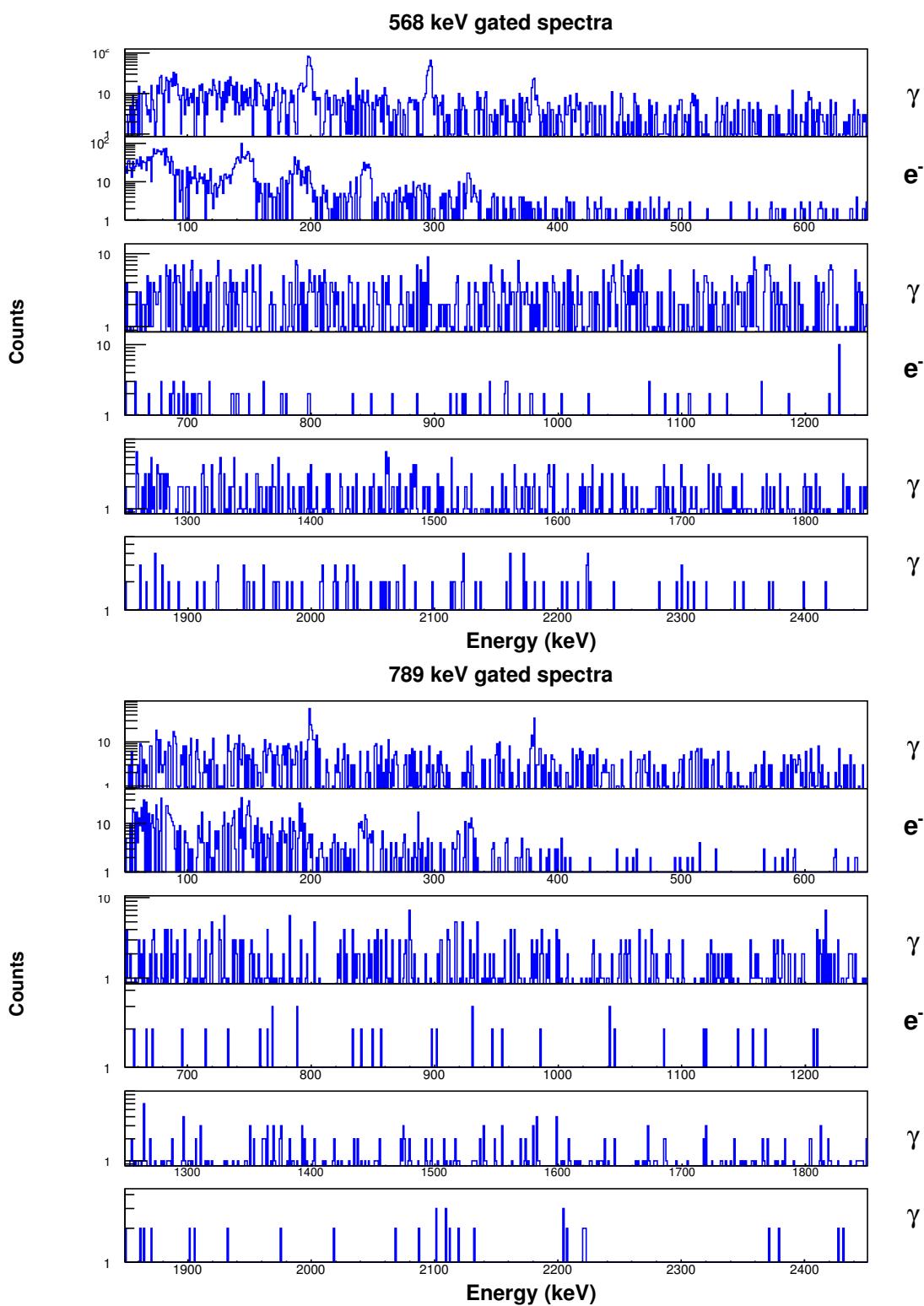


399 keV gated spectra

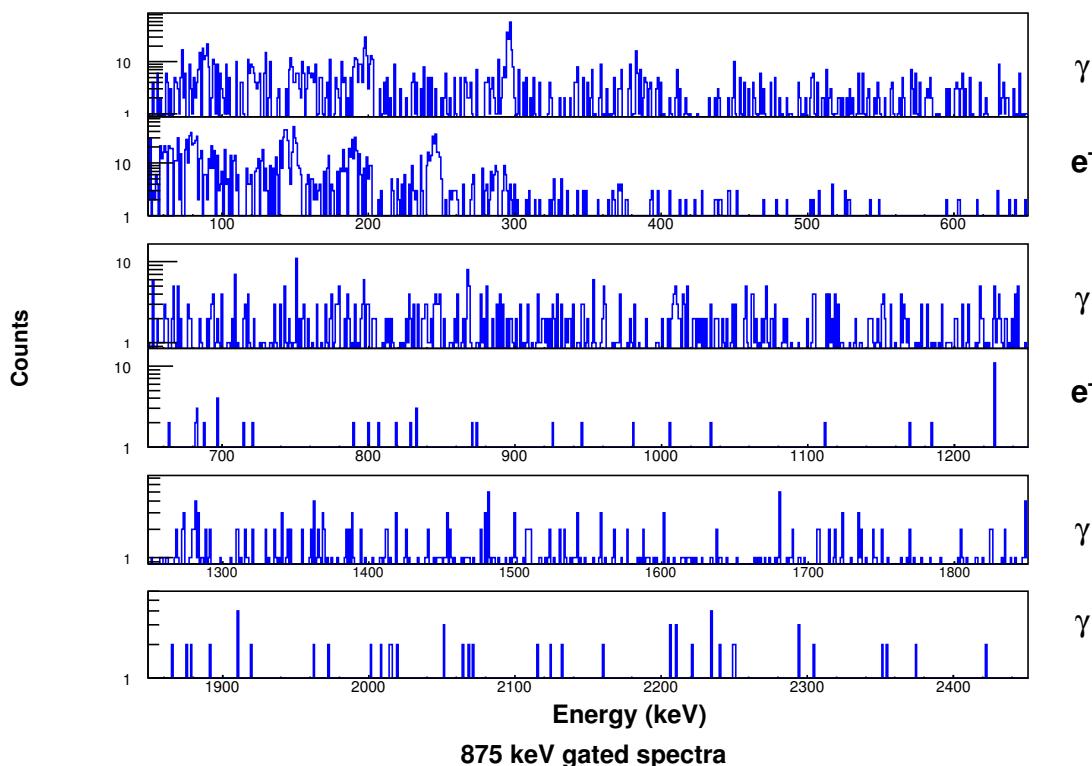


451 keV gated spectra

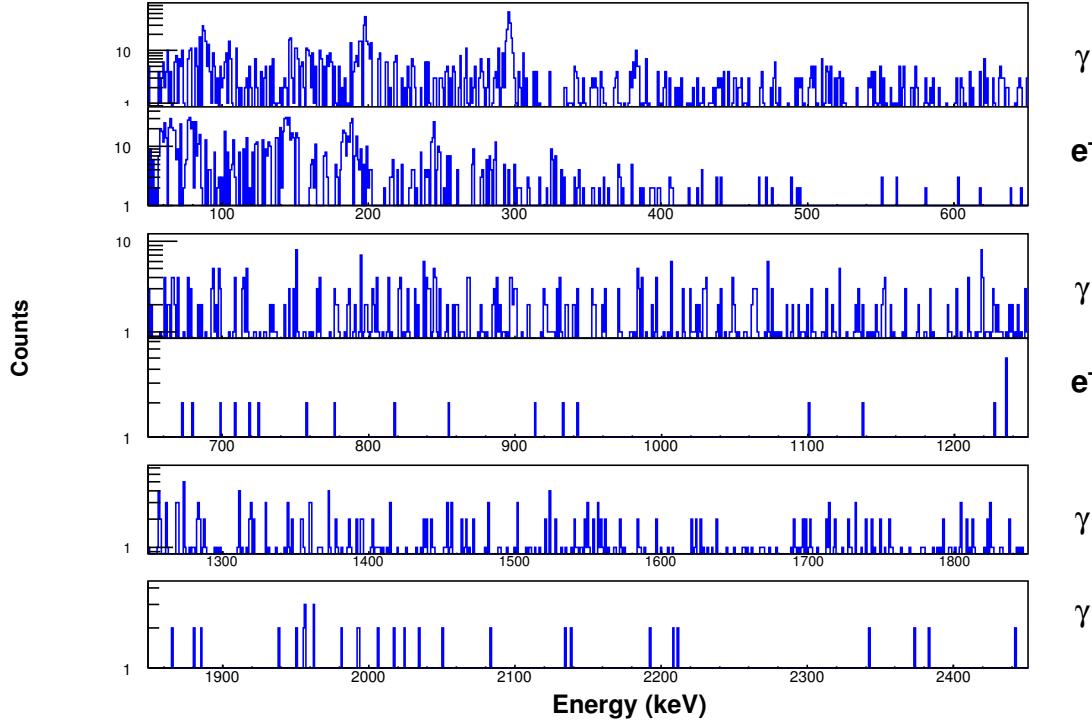




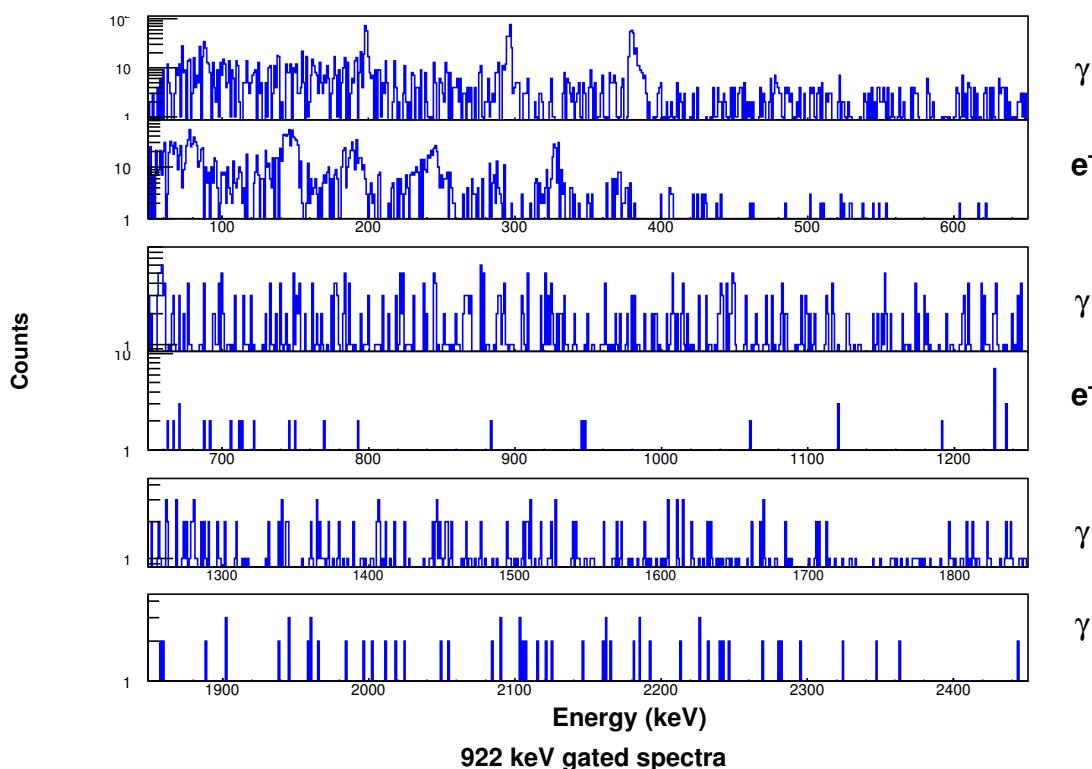
823 keV gated spectra



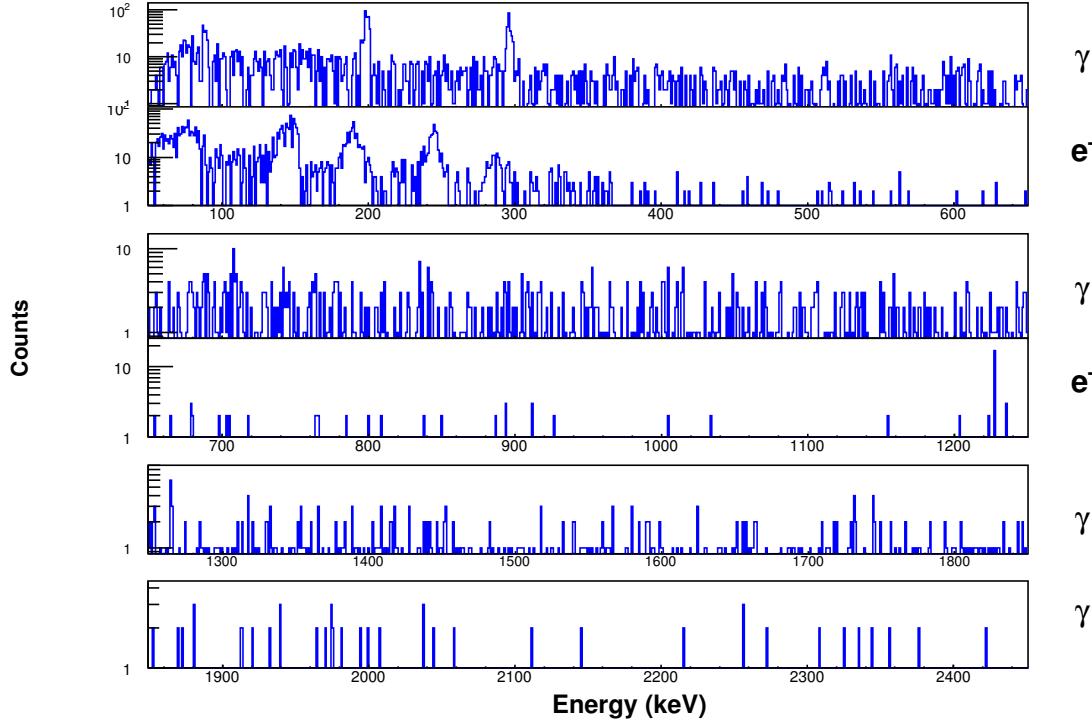
875 keV gated spectra



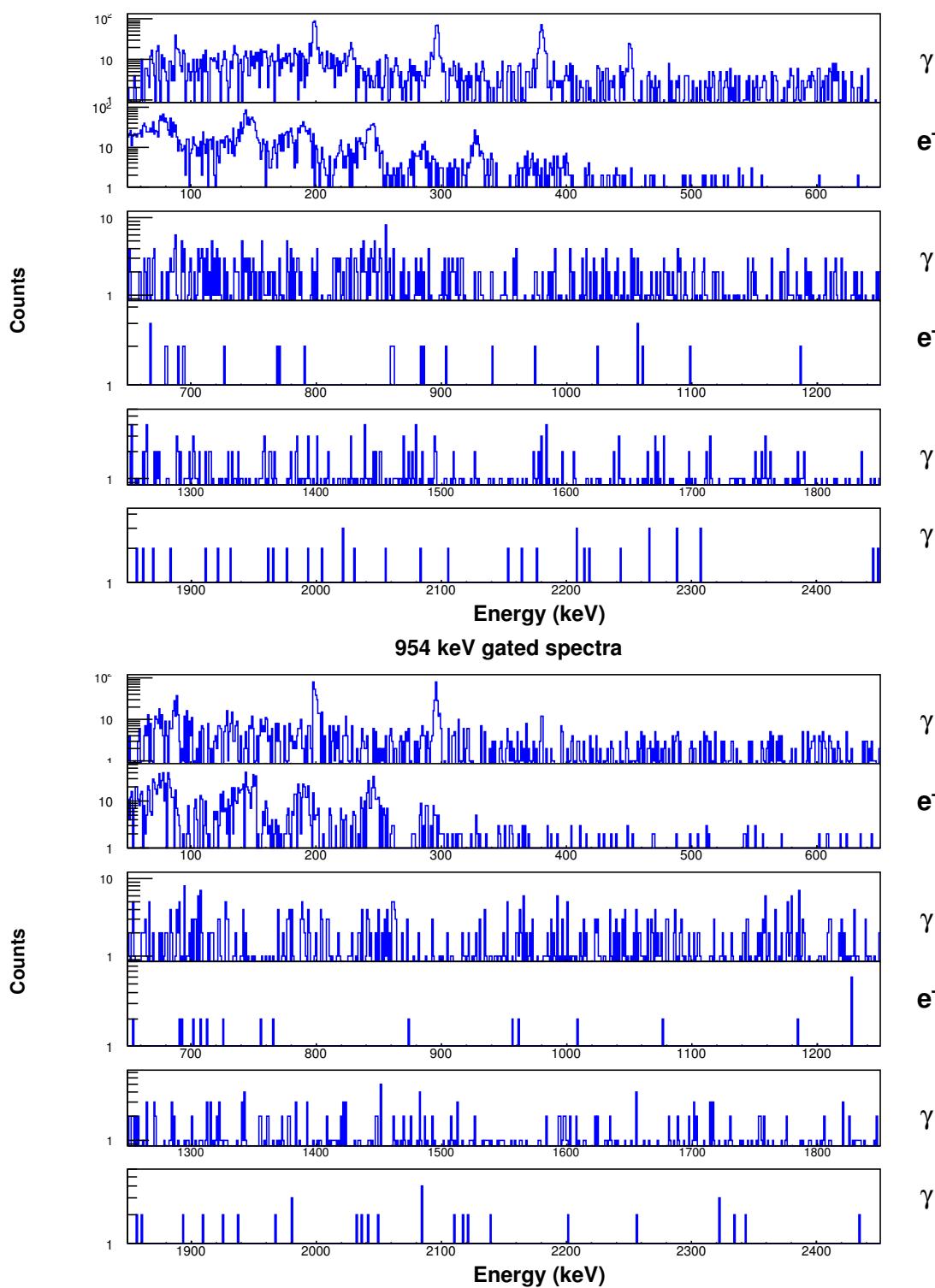
883 keV gated spectra



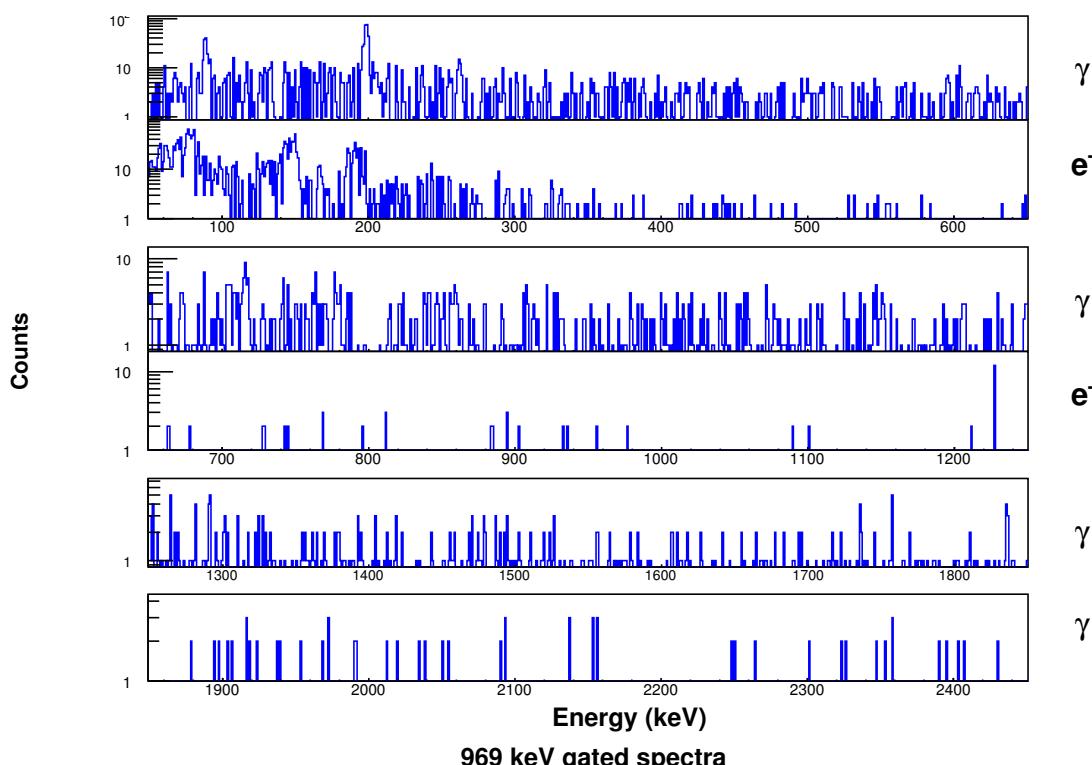
922 keV gated spectra



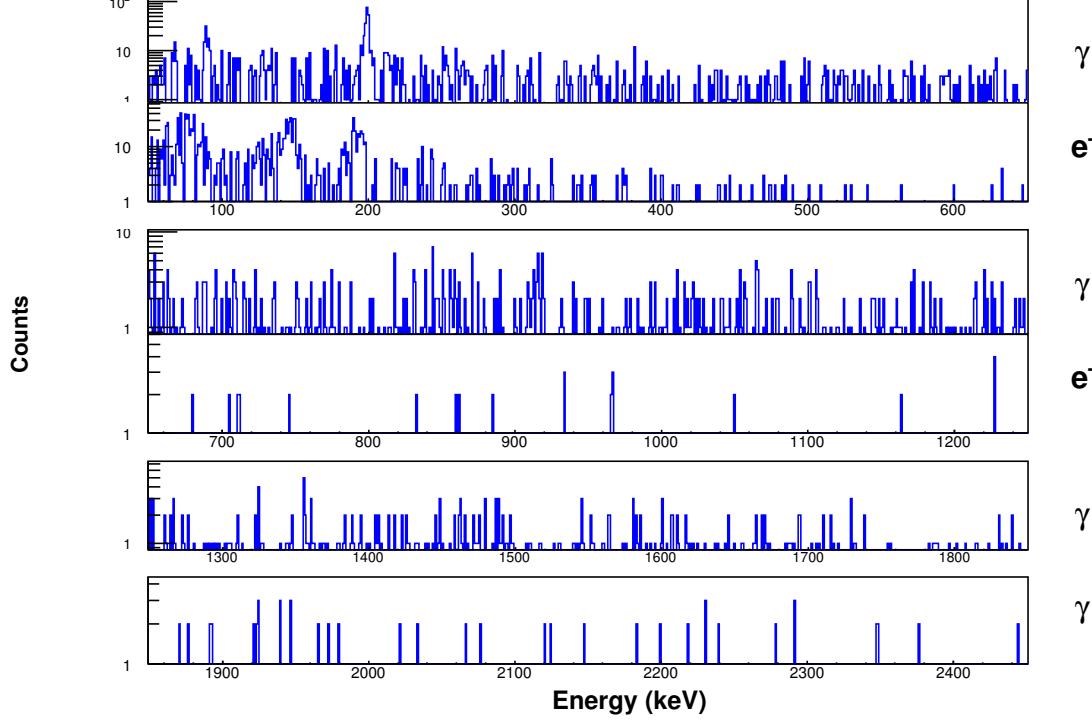
943 keV gated spectra



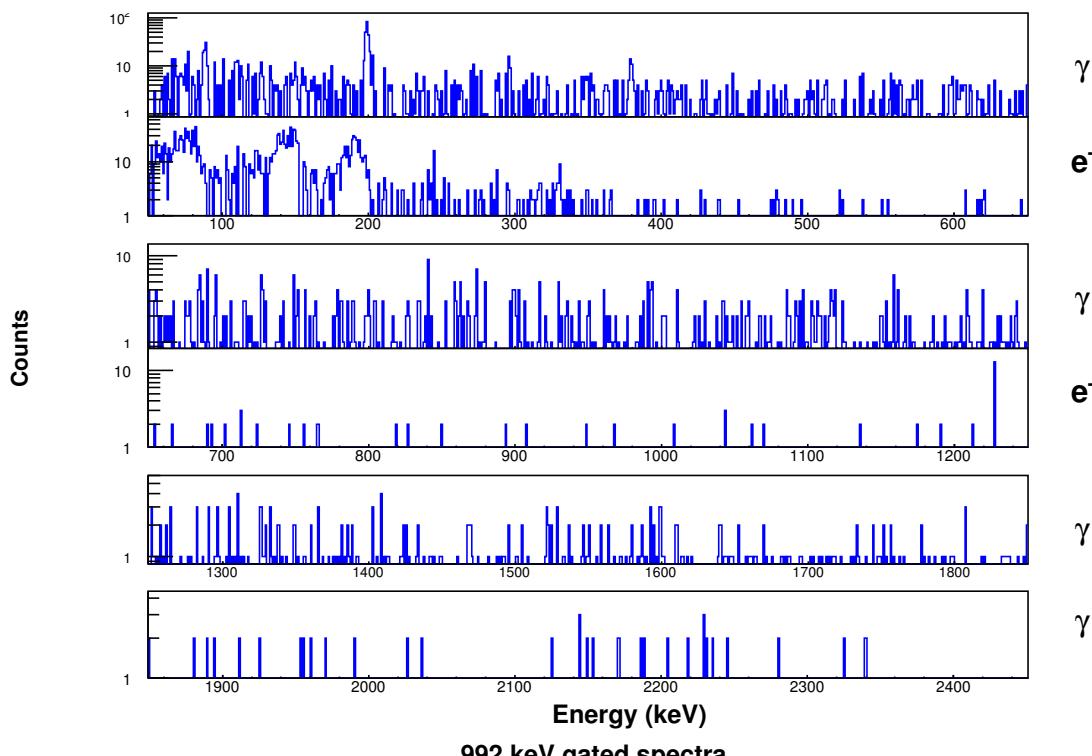
960 keV gated spectra



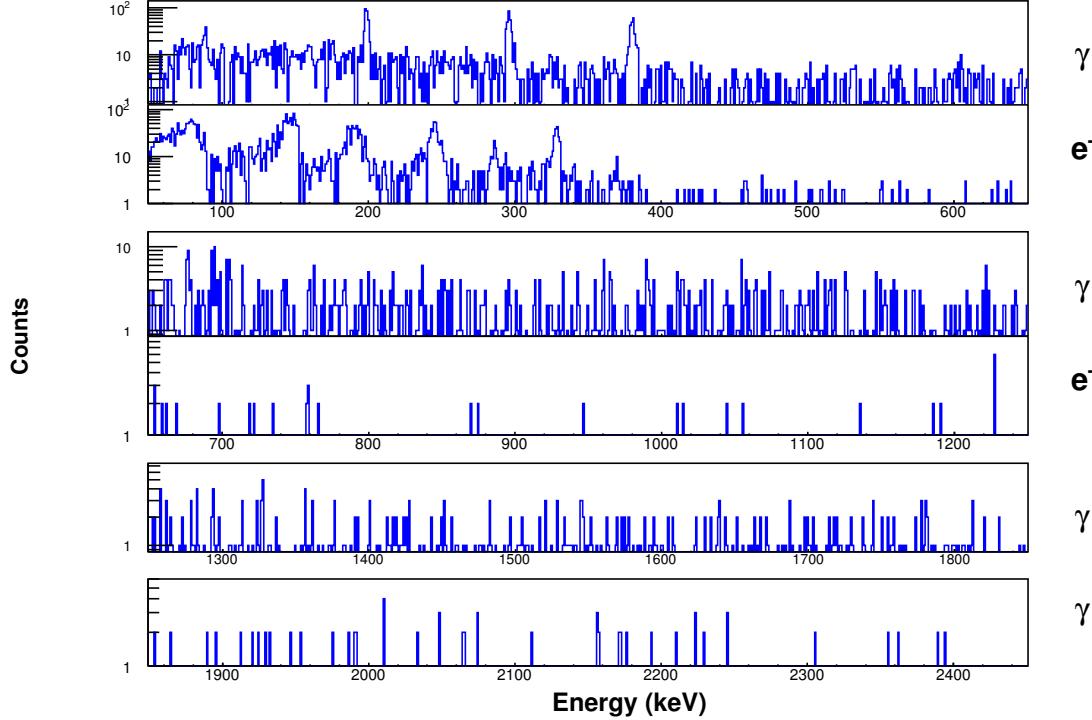
969 keV gated spectra



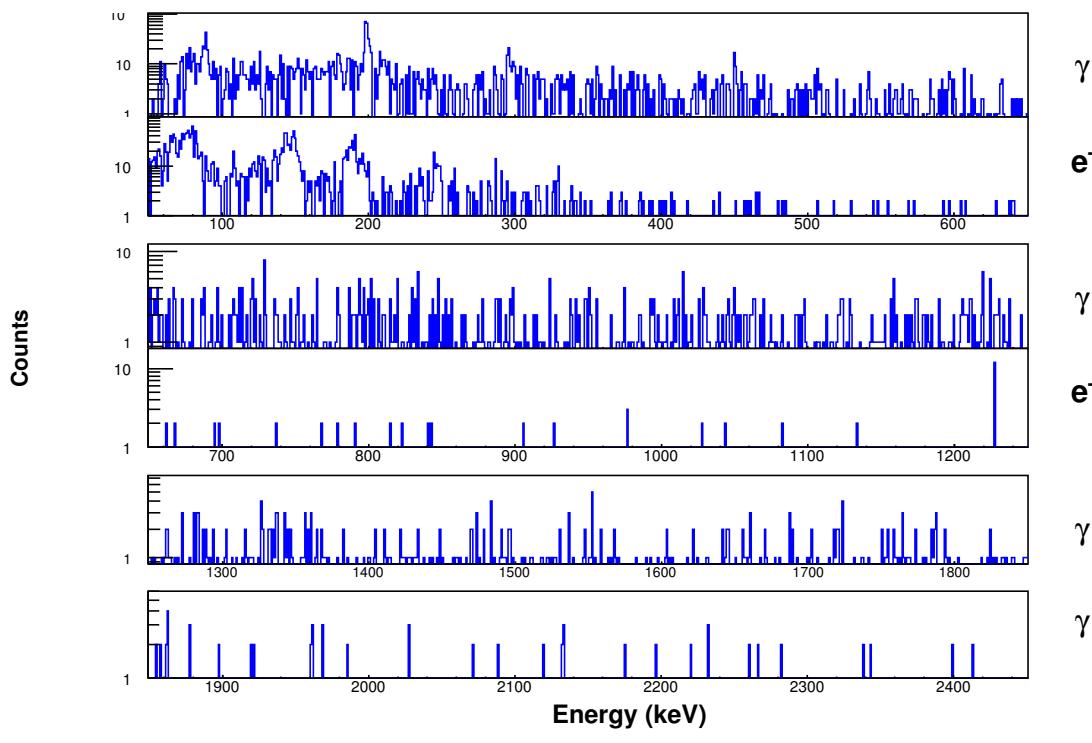
988 keV gated spectra



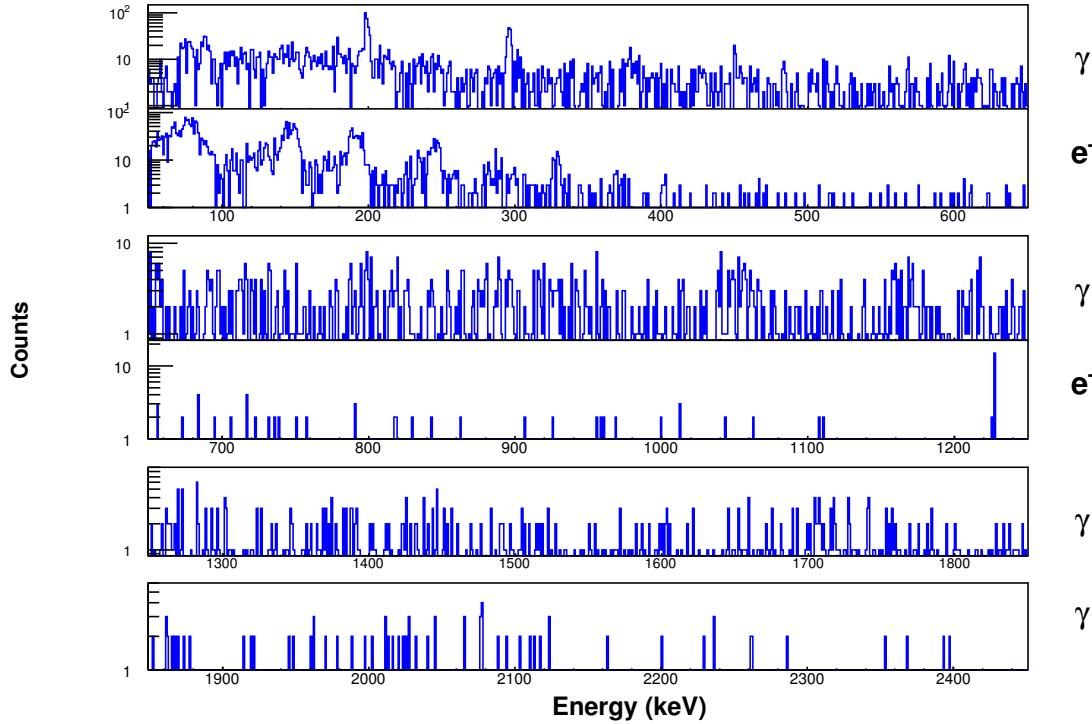
992 keV gated spectra



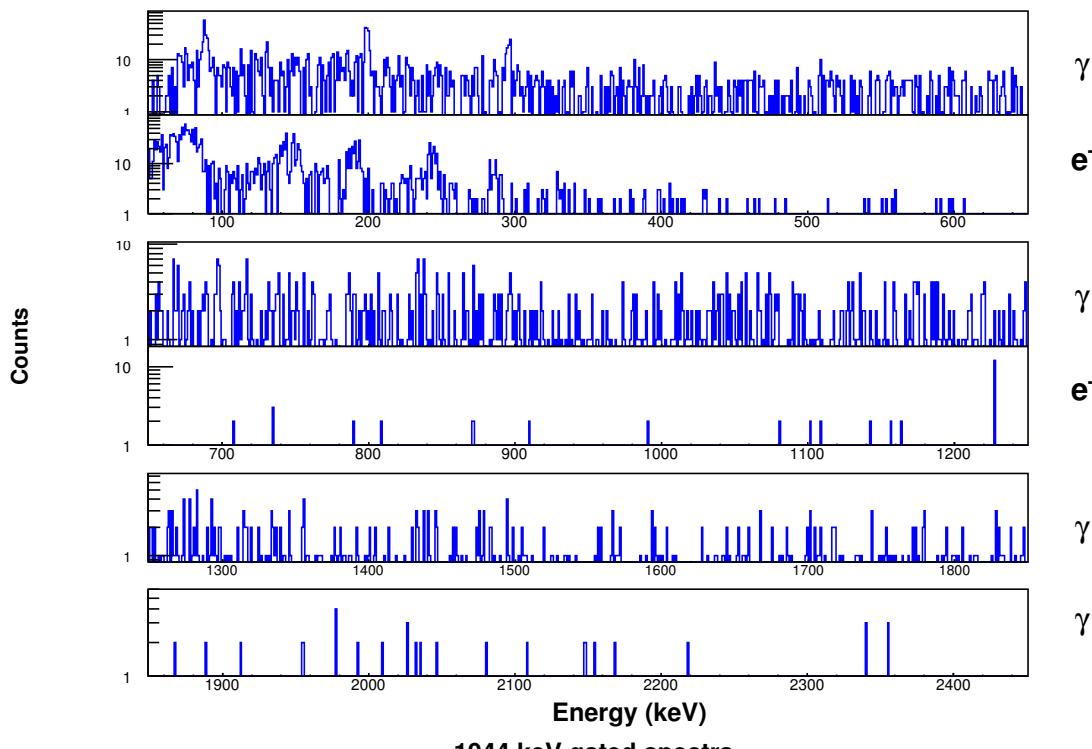
1010 keV gated spectra



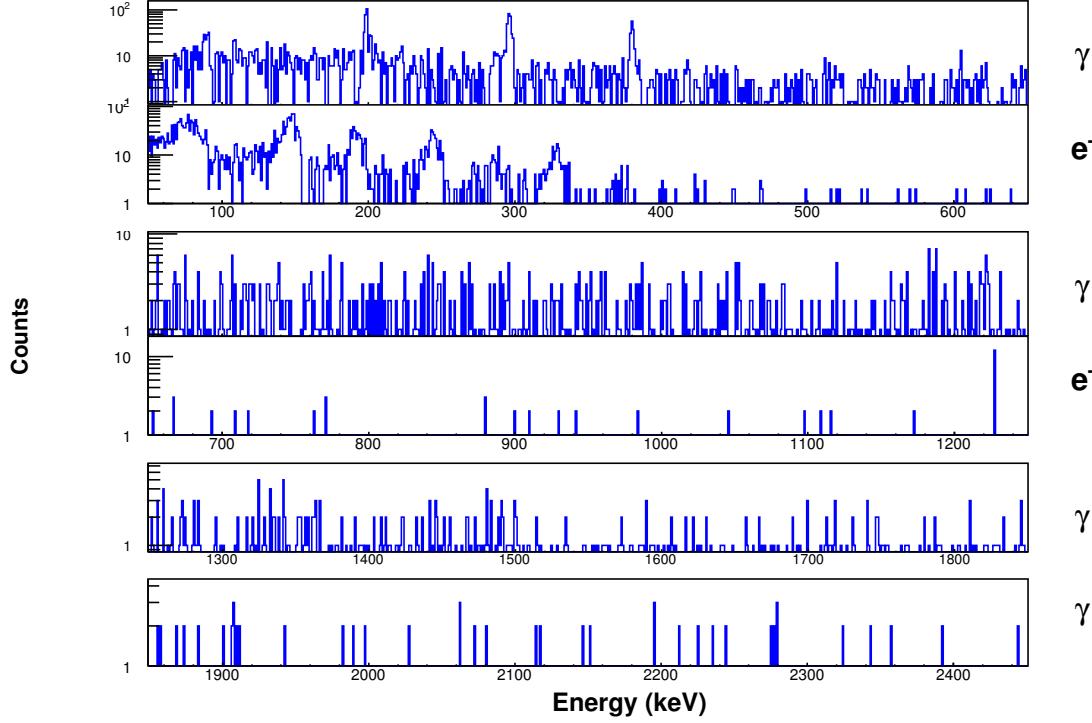
1013 keV gated spectra



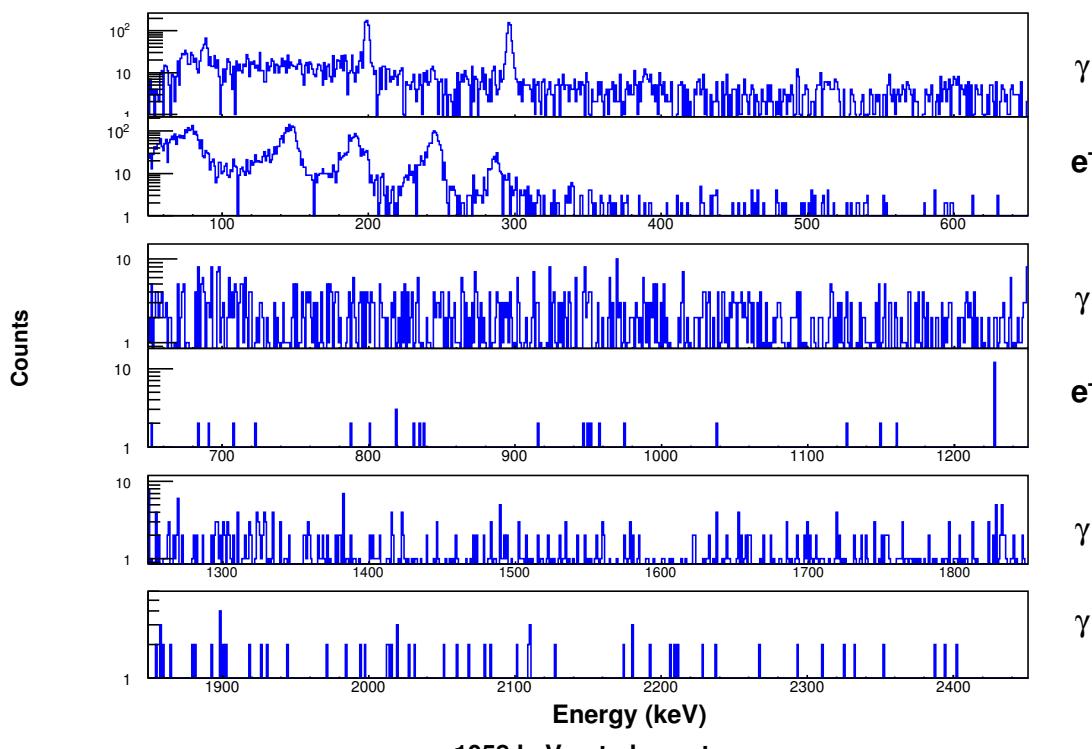
1039 keV gated spectra



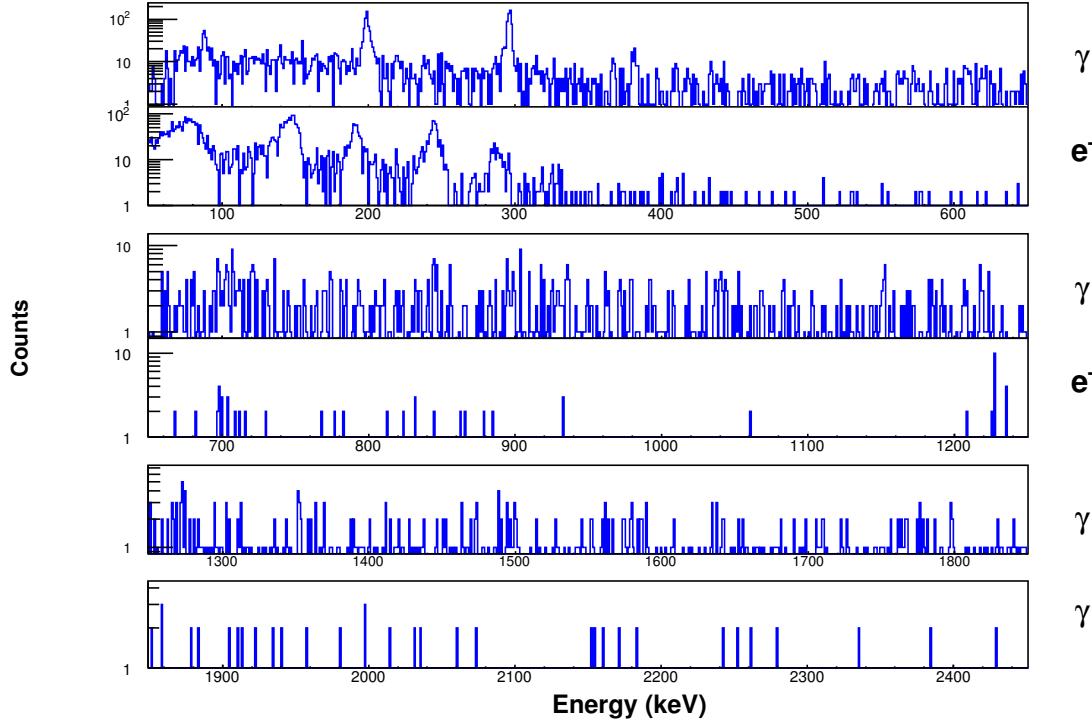
1044 keV gated spectra



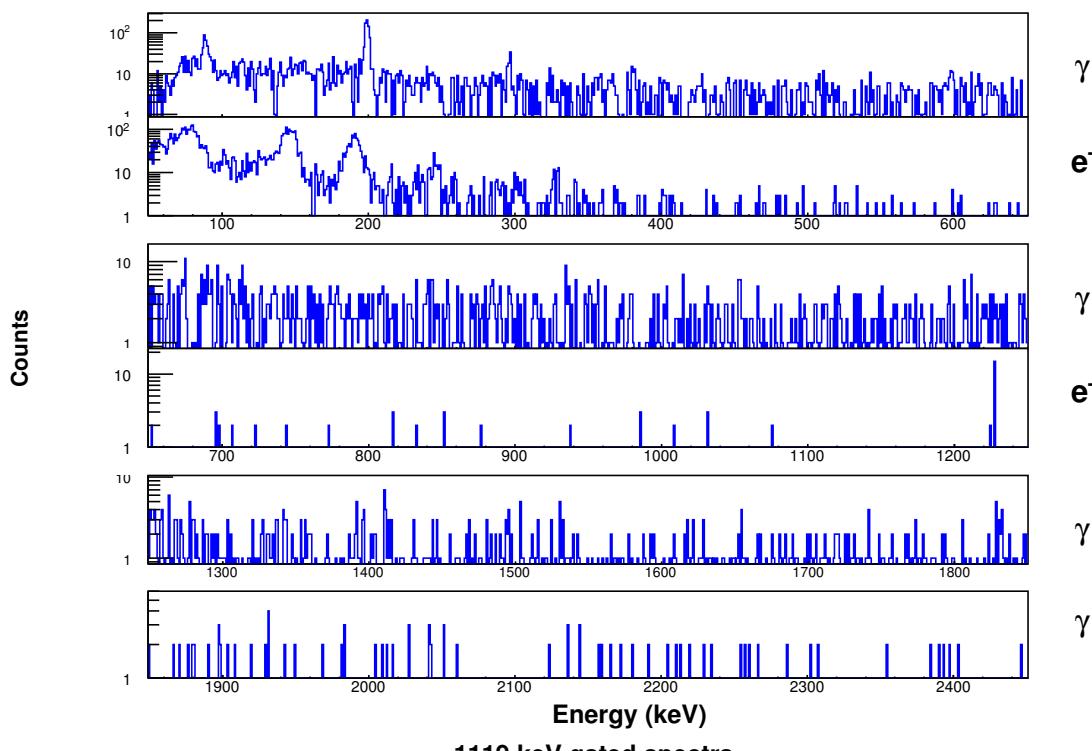
1052 keV gated spectra



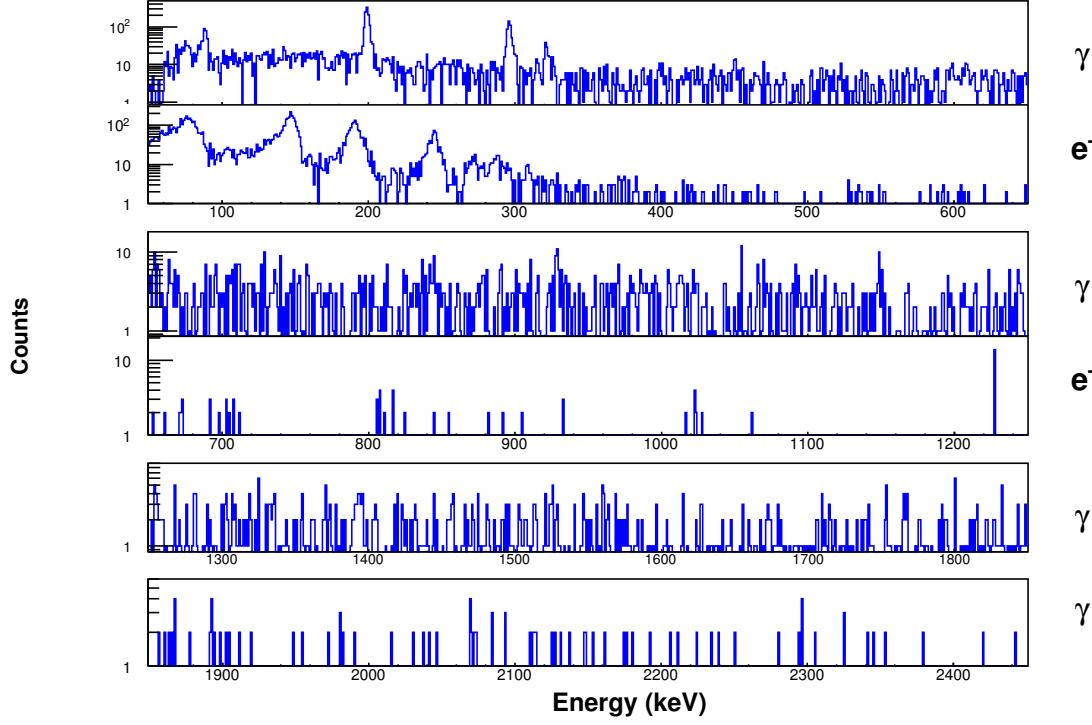
1058 keV gated spectra



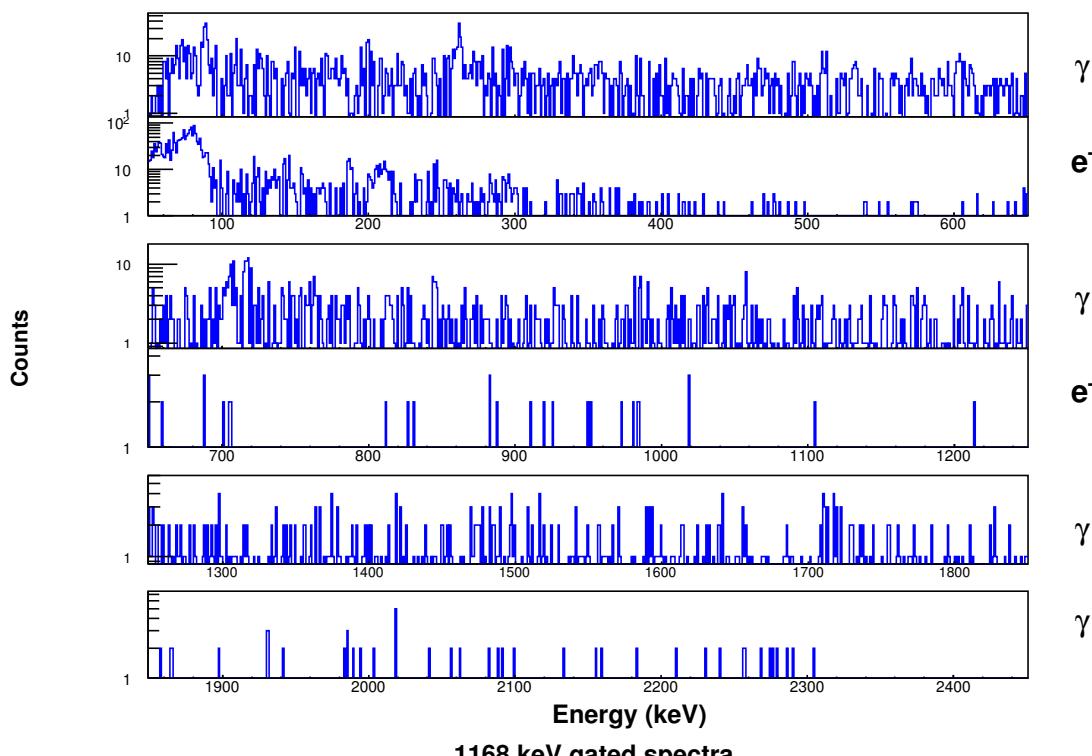
1067 keV gated spectra



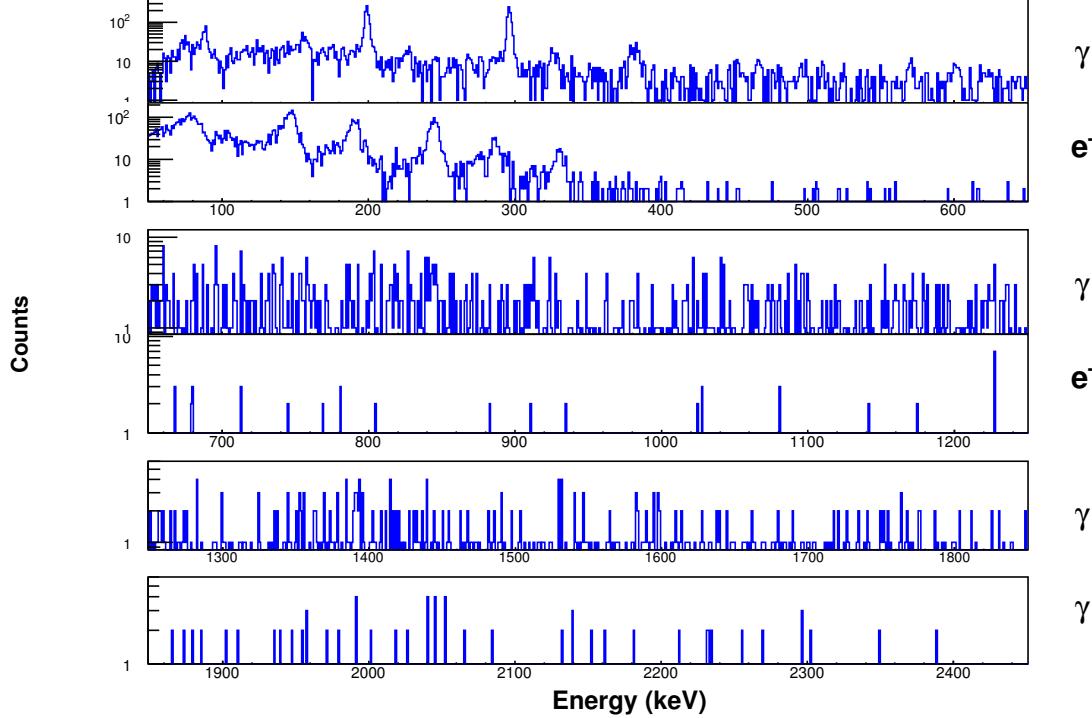
1119 keV gated spectra



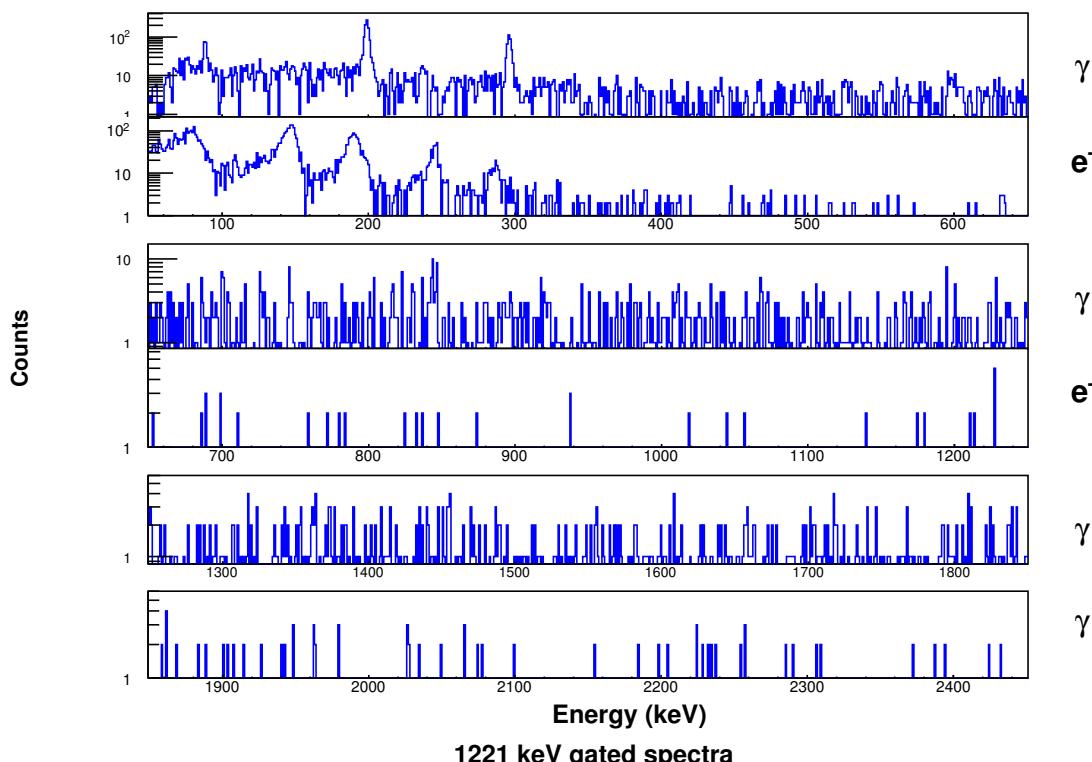
1156 keV gated spectra



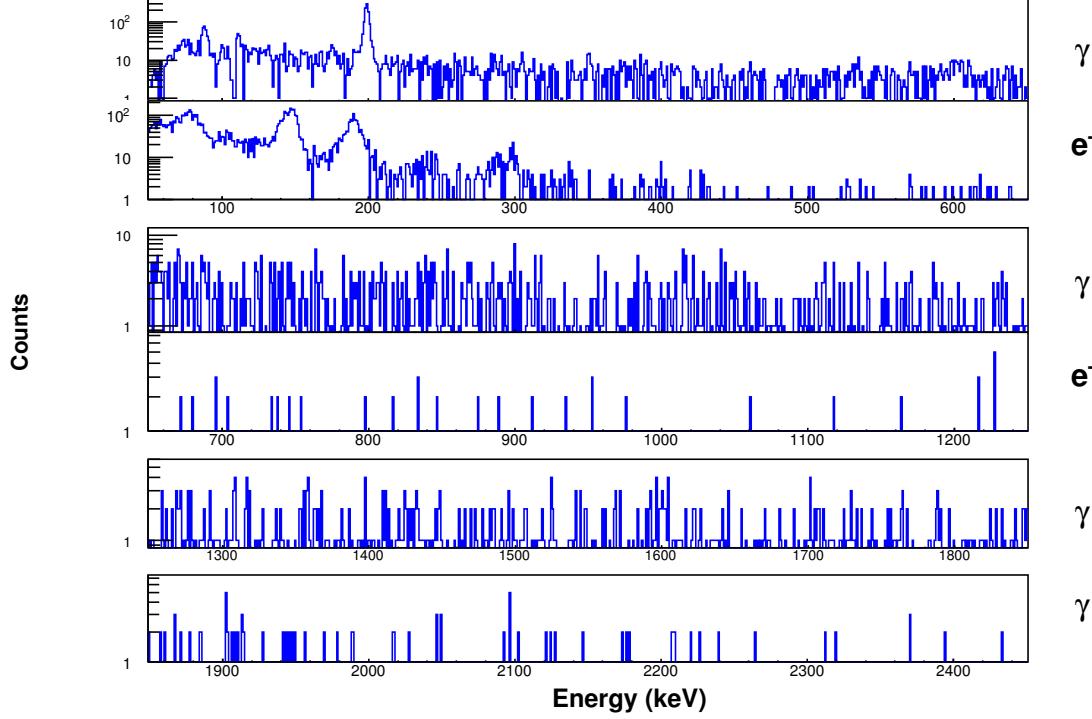
1168 keV gated spectra



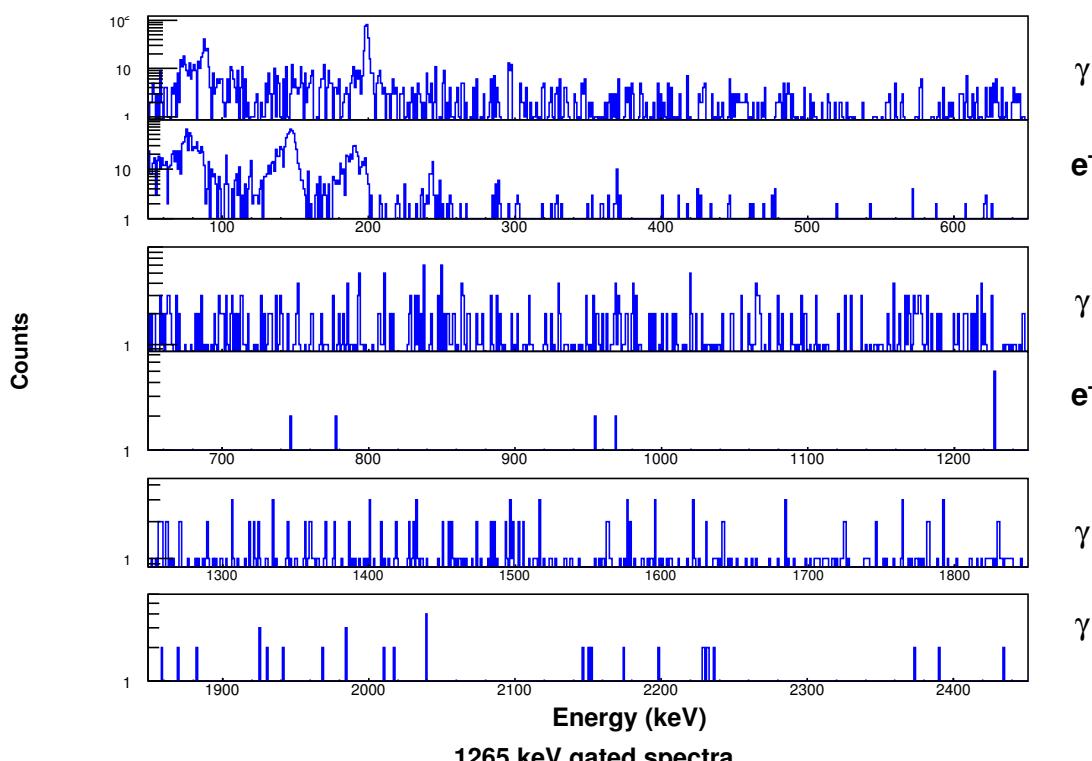
1180 keV gated spectra



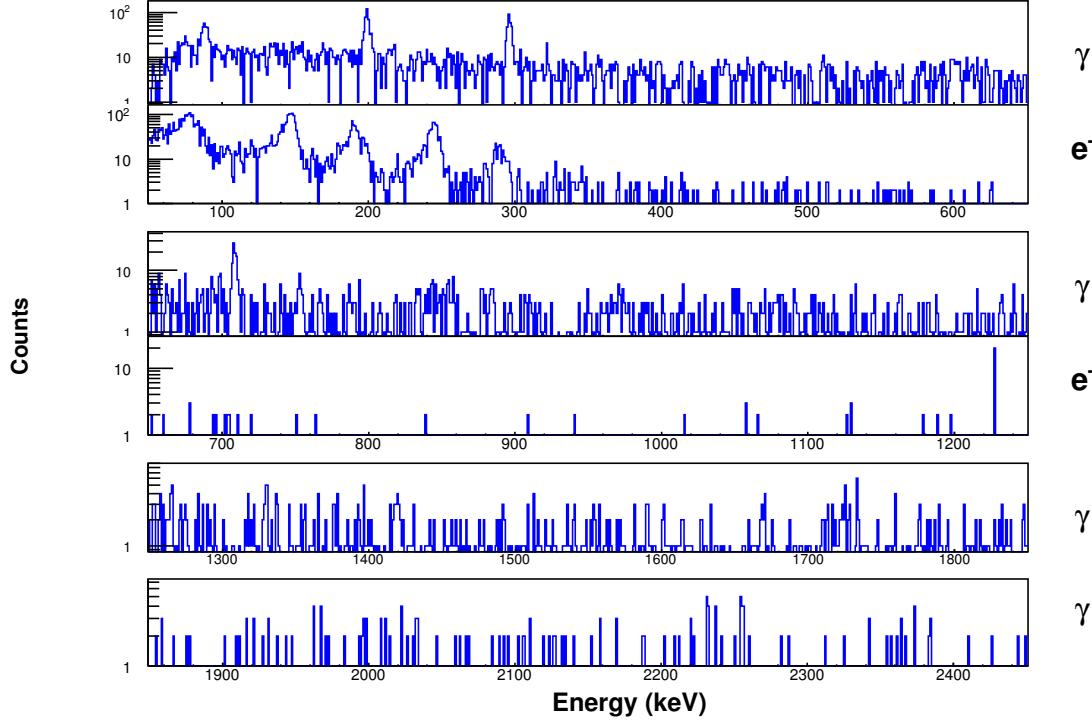
1221 keV gated spectra



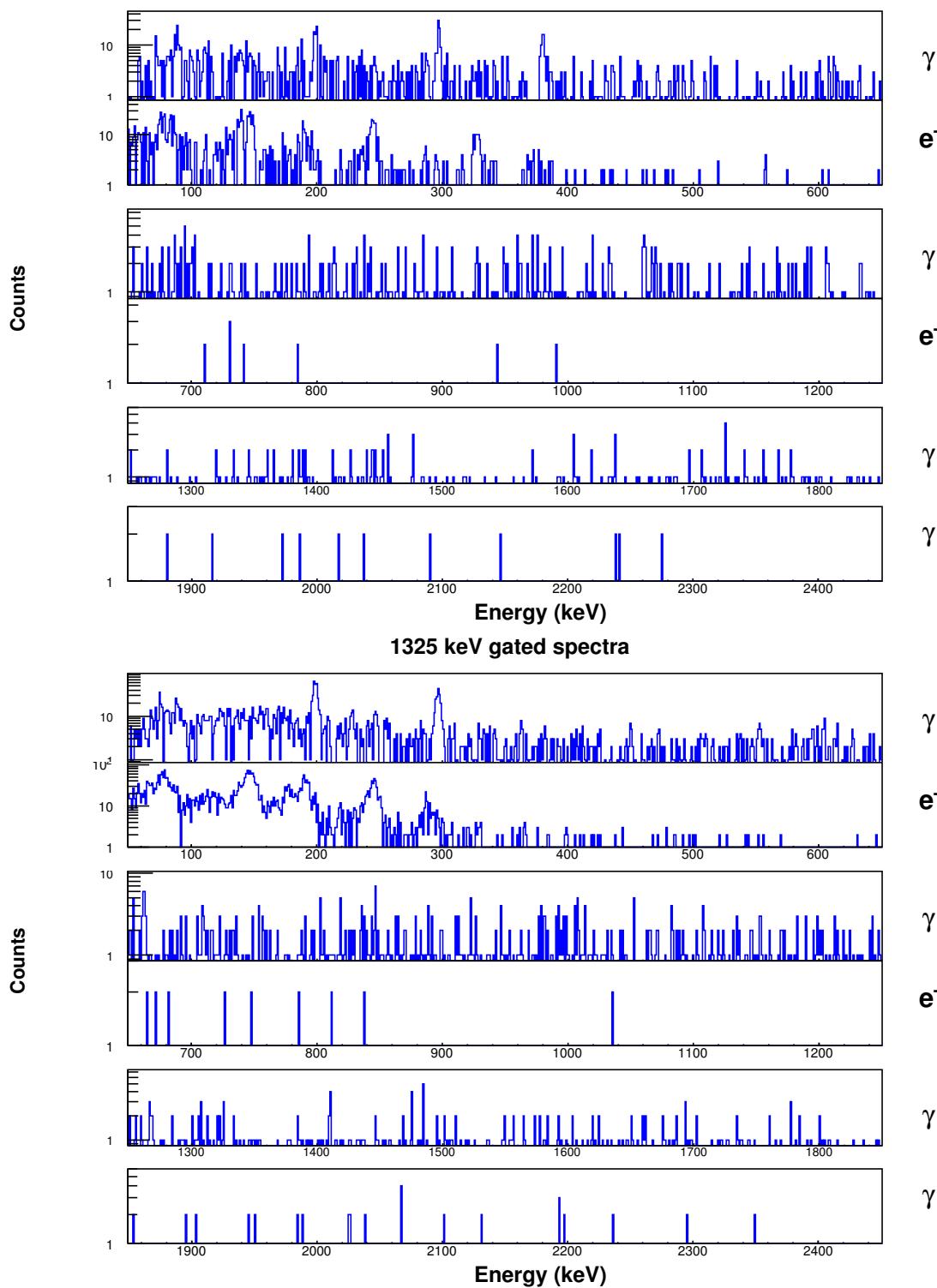
1250 keV gated spectra



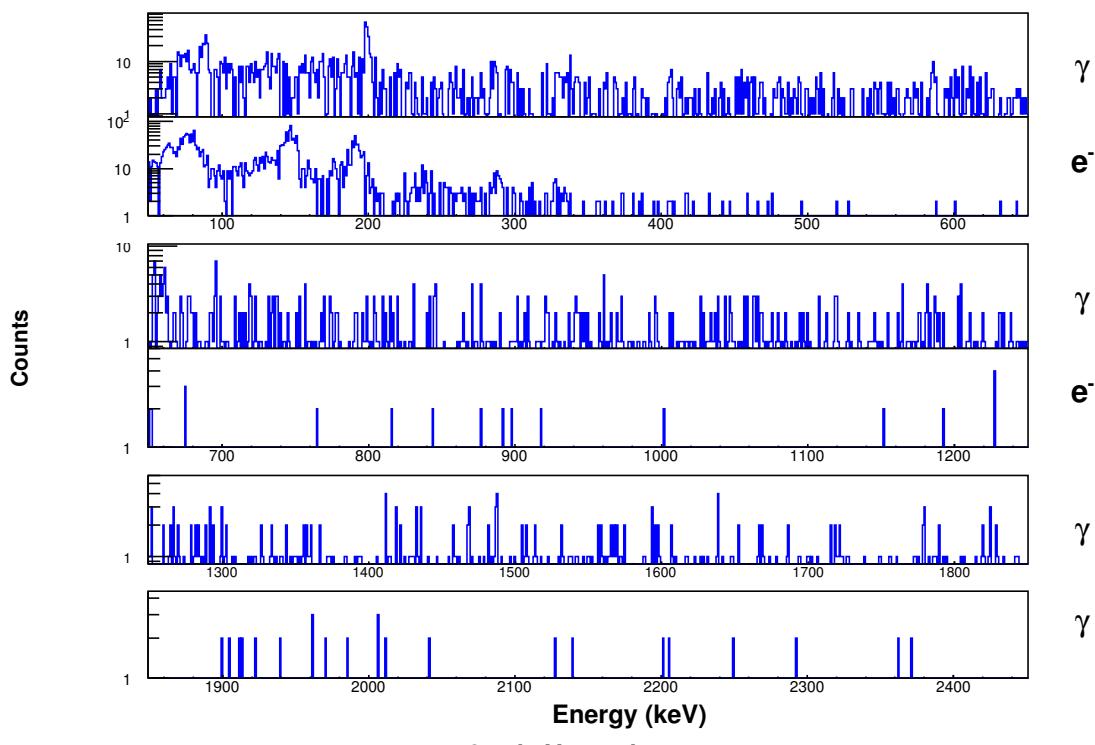
1265 keV gated spectra



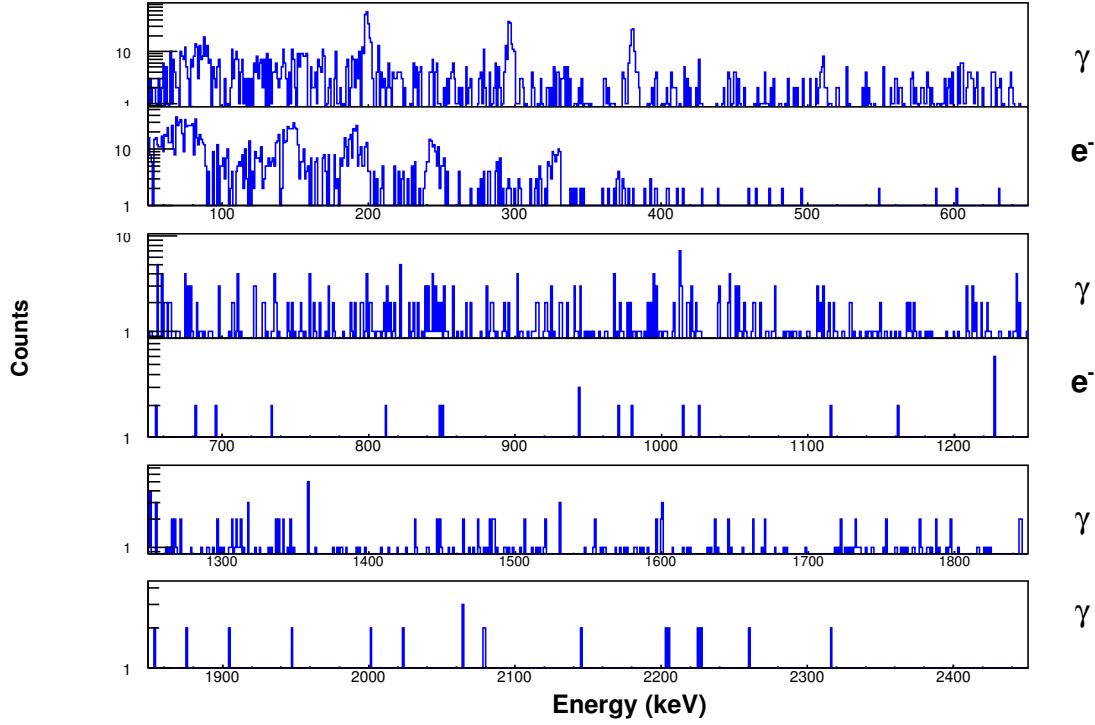
1284 keV gated spectra



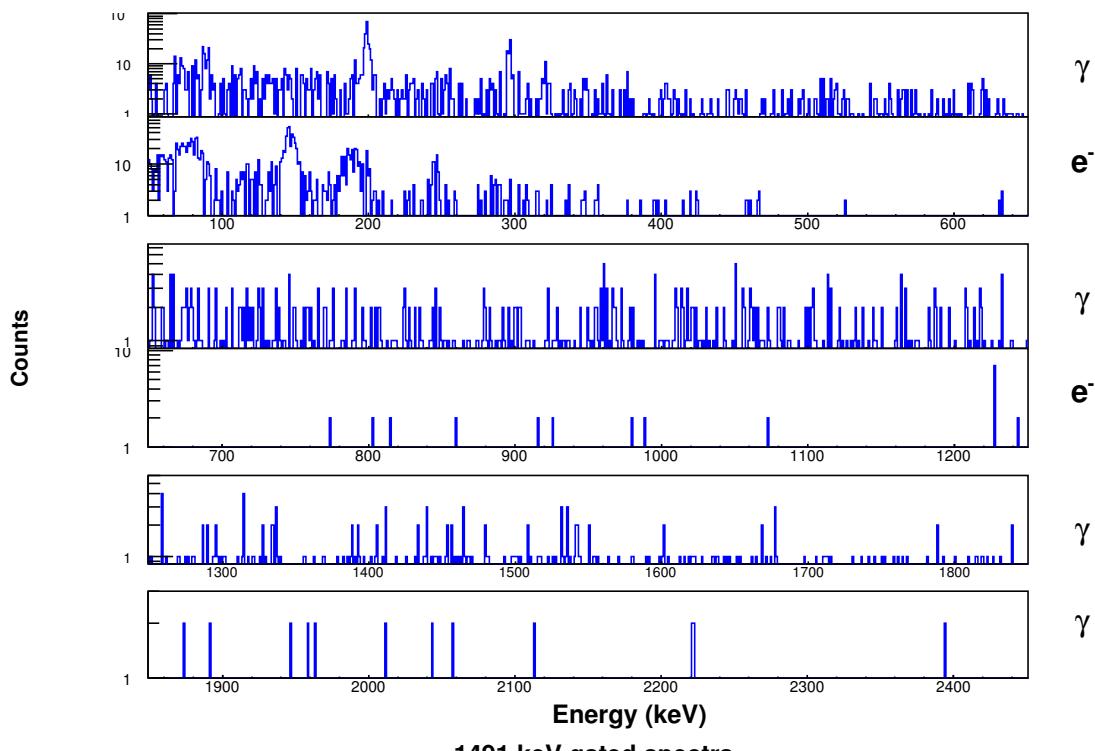
1334 keV gated spectra



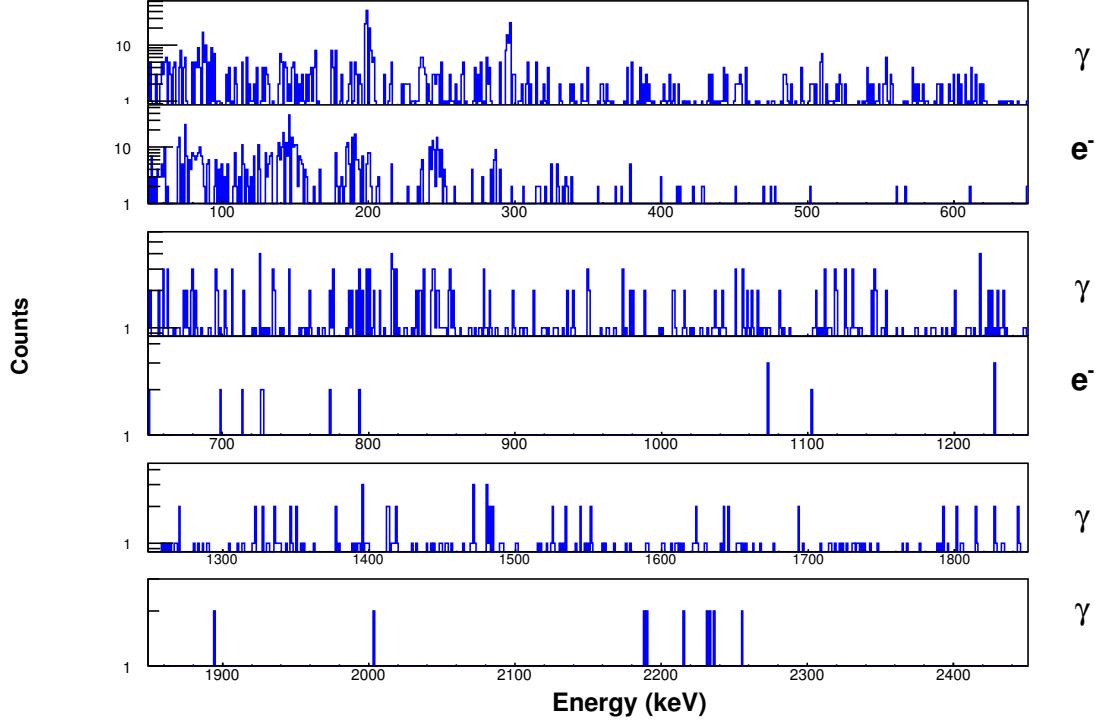
1355 keV gated spectra



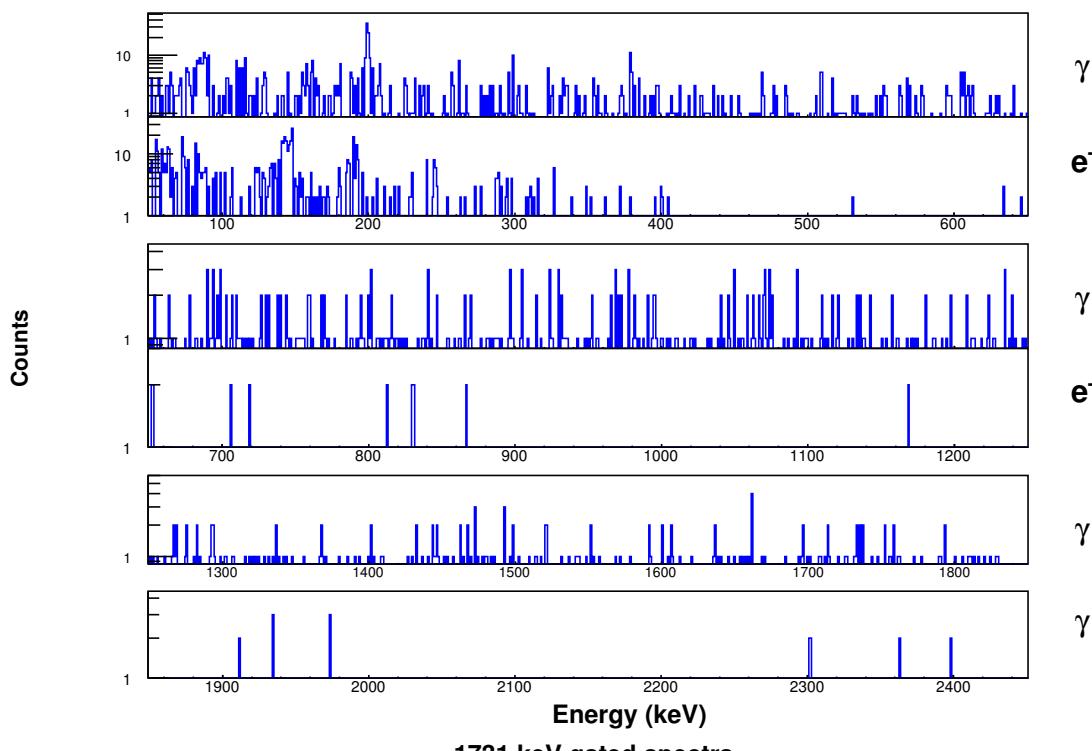
1479 keV gated spectra



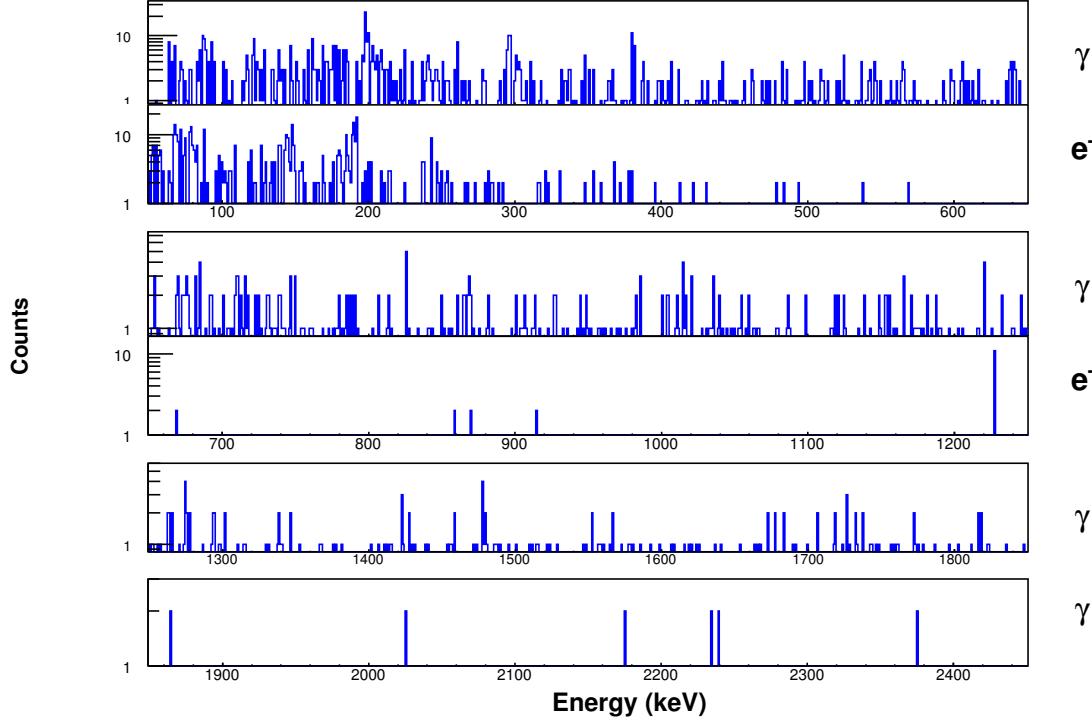
1491 keV gated spectra



1605 keV gated spectra



1731 keV gated spectra



APPENDIX C

ANALYSIS CODE

This appendix contains the analysis code used to gate on rootfiles. It was compiled using ROOT 5.34.19, and later ROOT 6.02 libraries. The GCC compiler version 7.1.0. Included in this appendix are the makefiles and example input files for parameters.

There are two codes used for analysis: the coincidence gating, and the code to merge the analysis of the individual runs together. These two codes were written to share multiple elements. The code in this appendix is broken into four sections: the code unique to the coincidence gating, the code unique to the merge code, the shared code between the two, and example input files from the `user` folder.

The folder structure for the code is in the following list. Some files listed are not used in the merge code, and would not be present in the directory. The topmost bullets are all in the main code directory.

- README.md
- Makefile
- QueueScript.sh
- include
 - Coefficients.h
 - Constraints.h
 - Filelist.h
 - analysis.h
 - histograms.h
 - timing.h
- logs
- src
 - Coefficients.cxx
 - Constraints.cxx

- Filelist.cxx
- analysis.cxx
- histograms.cxx
- main.cxx
- timing.cxx
- user
 - BGO.dat
 - Cut_Files
 - GeCut.dat
 - SiLiCut.dat
 - Filelist.dat
 - GeCoefficients.dat
 - Run_by_Run
 - GeCoefficients_r*.dat
 - SiLiCoefficients_r*.dat
 - SiLiCoefficients.dat
 - Timing.dat

Please note, "logs" is an otherwise empty folder that is used for putting output files from running code on the CRC backend. Should one wish to run code locally in the terminal window, this folder would be unnecessary, and the shell script named "QueueScript.sh" would be adjusted for local use.

C.1 Coincidence Code

The code in this appendix is expressly for the ICEBall-GEORGINA data. The ICEBall-Clovershare data used a different version of the `evt2root` converter, which no longer required the `libExpEvent.so` that was generated by the converter. The `analysis.cxx` file was also changed to reflect the different tree structure created by the converter. While not replicated here, the area of change is referred to in that subsection.

C.1.1 README.md

Simple included explanation of code inputs for running.

```

1 # ICEBall-Coincidence-Code
2
3 This code is to be used in conjunction with Merge-Parallel.
4
5 Inputs for the code are as follows (must be in this order)
6 1. Run Number
7 2. File output Name
8 3. Cut File Name
9 4. Timing File Name

```

C.1.2 Makefile

In this code, CodeDirectory must be replaced with the path to the code directory.

-L/CodeDirectory/libExpEvent.so

could be removed in the Clovershare data due to the converter changes.

```

1 #to compile using root libraries
2 ROOTCFLAGS := $(shell root-config --cflags)
3 ROOTLIBS := $(shell root-config --libs)
4 ROOTGLIBS := $(shell root-config --glibs)
5
6 #this is the compiler that will be used:
7 CXX = g++
8 #these are some compiler flags
9 CXXFLAGS = -O2 -fPIC -g $(ROOTCFLAGS)
10
11 #folder including all the header files
12 INCLUDES := -Iinclude -Iuser
13 #folder with all the source files
14 SOURCES := src
15 #source files
16 FILES := Coefficients.cxx Constraints.cxx histograms.cxx Filelist.cxx timing.cxx
   ↳ analysis.cxx main.cxx
17 #this is the executable name
18 EXECUTABLES := main
19 #libraries
20 LIBS = -L/CodeDirectory/libExpEvent.so $(ROOTLIBS) $(SYSLIBS)
21 #this are object files (take the names of sources and change the extensions)
22 OBJECTS := $(FILES:.cxx=.o)
23
24 #don't do anything to files named 'clean' and 'rmobjects'
25 .PHONY: clean rmobjects
26
27 $(EXECUTABLES): $(OBJECTS)
28   $(CXX) $(LIBS) $(INCLUDES) $(CXXFLAGS) $(OBJECTS) -o $(EXECUTABLES)
29

```

```

30  #compile all source files and generate object files
31  # %.cxx = any file that matches the pattern
32  # $< = name of the first dependency file
33  # $@ = name of the target file
34  %.o: $(SOURCES)/%.cxx
35      $(CXX) $(INCLUDES) $(CXXFLAGS) -c $< -o $@
36
37  #this will remove the executable file and objects
38  clean:
39      rm -rf $(EXECUTABLES) $(OBJECTS)
40
41  rmobjects:
42      rm -rf $(OBJECTS)

```

C.1.3 Queuescript.sh

This shell script is written for executing the code on the CRC backend computers.

It can be used locally by removing the commented lines down to line 9, and replacing

`$SGE_TASK_ID`

with the run number. The terms "netID", "Nickname", "runStart", "runEnd", "CodeDirectory", "FileOut", "CutFile", and "TimingFile" must all be replaced with the correct information.

```

1 #!/bin/tcsh
2
3 #$ -M netID@nd.edu
4 #$ -m a
5 #$ -q long
6 #$ -N Nickname
7 #$ -o logs/
8 #$ -t runStart-runEnd
9 #$ -j y
10
11 module load /afs/crc.nd.edu/user/n/ns1/nuclear/startup/ns1
12 module load root/6.02
13 root-config --version
14 cd CodeDirectory
15 echo "start task ${SGE_TASK_ID}"
16 date
17 ./main $SGE_TASK_ID FileOut CutFile TimingFile
18 echo "ended task ${SGE_TASK_ID}"

```

```

19  date
20
21  exit $?

```

C.1.4 src/main.cxx

This is the main code that calls all of the subroutines. The gSystem line is necessary to load the library to read the tree structure of the root files. "Directory" and "OutputDirectory" should be replaced with the proper filepaths.

```

1  /*
2  Please check the README for more information about this code and it's purpose.
3  */
4
5  //C++ libraries
6  #include <iostream>
7  #include <fstream>
8  #include <string>
9  #include <cmath>
10 #include <cstdio>
11 #include <cstdlib>
12 #include <vector>
13
14 //ROOT libraries
15 #include "TSystem.h"
16 #include "TROOT.h"
17 #include "TH2.h"
18 #include "TStyle.h"
19 #include "TCanvas.h"
20 #include "TString.h"
21 #include "TFile.h"
22 #include "TChain.h"
23 #include "TH1F.h"
24 #include "TTree.h"
25 #include "TRandom3.h"
26 #include " TBranch.h"
27 #include "TLeaf.h"
28 #include "TObject.h"
29 #include "TObjArray.h"
30 #include "TMath.h"
31
32 //Object files
33 #include "analysis.h"
34 #include "Constraints.h"
35 #include "histograms.h"
36 #include "Coefficients.h"
37 #include "Filelist.h"
38 #include "timing.h"

```

```

39
40 //Data tree files
41 // #include "eventData.h"
42 // #include "eventScaler.h"
43 // #include "ExpEventDictionary.h"
44 // #include "ExpEventLinkDef.h"
45
46 using namespace std;
47
48 //File path info
49 extern std::string sFilepath;
50 extern std::string sTree;
51 extern std::string sRun;
52 extern std::string sFType;
53
54 extern TChain* chain;
55 extern std::vector<std::vector<TH1F*>> ge_en_ge_cut;
56 extern std::vector<std::vector<TH1F*>> sili_en_ge_cut;
57 extern std::vector<std::vector<TH1F*>> ge_en_sili_cut;
58 extern std::vector<std::vector<TH1F*>> sili_en_sili_cut;
59
60
61 extern int nGeOrder; //Order of calibration i.e. 1 = linear.
62 extern int nGeDets; //Total number of signals from Germanium detectors
63 extern int nGeSegments; //number of segments in a single Germanium crystal, for
   ↳ adding purposes
64 extern std::vector<std::vector<double>> dGeCoefficients; //Ge Coefficients
65 extern std::vector<std::vector<double>> dGeCoeffRes; // Ge Residual Coefficients
66 extern std::vector<std::vector<double>> dGeRunCorr; //Run Correction
   ↳ Coefficients
67
68 extern int nSiLiOrder; //Order of calibration i.e. 1 = linear.
69 extern int nSiLiDets; //Total number of signals from SiLi
70 extern int nSiLiPlace; //Start of SiLis in generalized array detectors
71 extern std::vector<std::vector<double>> dSiLiCoefficients; //Coefficients
72 extern std::vector<std::vector<double>> dSiLiRunCorr; //Run Correction
   ↳ Coefficients
73
74 std::vector<std::vector<double>> dGeBounds; //bounds for cuts
75 std::vector<std::vector<double>> dSiLiBounds; //bounds for cuts
76 int nGeConstraints;
77 int nSiLiConstraints;
78
79 Timing tGates;
80
81 int main(int argc, char* argv[]) //Order of arguments: run #, output filename,
   ↳ cut filename, time low, time high
82 {
83     char buffer[50];
84     int nRunNum;
85     nRunNum = atoi(argv[1]); //Run to do the cuts on
86     char* sOut = argv[2]; //file title to write to
87     char* sCut1 = argv[3]; //Cut file name indicator
88     char* sTime = argv[4]; //Cut file name indicator

```

```

89 gSystem->Load("$Directory$/libExpEvent.so","", kTRUE);
90 readPaths(); //From Filelist.cxx
91 makeChain(nRunNum); //From Filelist.cxx
92 defineGeCoeff(); //From Coefficients.cxx
93 defineGeCoeff(nRunNum); //From Coefficients.cxx, correction terms
94 defineSiLiCoeff(); //From Coefficients.cxx
95 defineSiLiCoeff(nRunNum); //From Coefficients.cxx
96 defineBGO(); //From constraints.cxx
97 tGates = Timing(nGeDets/nGeSegments, nSiLiDets, sTime);
98 //Debugging for the Timing class
99 std::cout << "Germanium pairs: " << tGates.geGe.size() << std::endl;
100 //for (int i = 0; i < tGates.geGe.size(); i++) std::cout << tGates.geGe[i][0]
101 //<<"\t" << tGates.geGe[i][1] <<std::endl;
102 std::cout << "Germanium-SiLi pairs: " << tGates.geSiLi.size() << std::endl;
103 //for (int i = 0; i < tGates.geSiLi.size(); i++) std::cout <<
104 //<< tGates.geSiLi[i][0] <<"\t" << tGates.geSiLi[i][1] <<std::endl;
105 std::cout << "SiLi pairs: " << tGates.siliSiLi.size() << std::endl;
106 //for (int i = 0; i < tGates.siliSiLi.size(); i++) std::cout <<
107 //<< tGates.siliSiLi[i][0] <<"\t" << tGates.siliSiLi[i][1] <<std::endl;
108 analysis ana(chain); //analysis class. Main part of code.
109 sprintf(buffer,"GeCut_%s.dat",sCut1); //File name to input
110 nGeConstraints = defineConstraints(buffer,dGeBounds); //From constraints.cxx
111 sprintf(buffer,"SiLiCut_%s.dat",sCut1); //File name to input
112 nSiLiConstraints = defineConstraints(buffer,dSiLiBounds);
113 makeHistograms(nGeDets/nGeSegments,nGeConstraints,nSiLiDets,nSiLiConstraints);
114 //from histograms.cxx
115 ana.Loop(Form("$OutputDirectory%/%s_run_00%i.root",sOut,nRunNum),nRunNum,
116 //fOut is in Filelist.h
117 }
```

C.1.5 src/analysis.cxx

This is the main analysis section of the code. The skeleton is built using the MakeClass routine in ROOT.[12] If the structure of the tree varies, this is where the changes would reflect, from lines 154-206, where the event tree values are assigned to local variables for manipulation in the code.

```

1 #define analysis_cxx
2 #include "analysis.h"
3
4 //C++ libraries
5 #include <iostream>
6 #include <fstream>
7 #include <string>
8 #include <cmath>
9 #include <vector>
10
```

```

11 //ROOT libraries
12 #include <TSystem.h>
13 #include <TROOT.h>
14 #include <TH2.h>
15 #include <TStyle.h>
16 #include <TCanvas.h>
17 #include < TString.h>
18 #include <TFile.h>
19 #include <TH1F.h>
20 #include <TTree.h>
21 #include <TRandom3.h>
22 #include <TMath.h>
23
24 //object files
25 #include "histograms.h"
26 #include "Constraints.h"
27
28 //Data tree files
29 // #include "eventData.h"
30 // #include "eventScaler.h"
31 // #include "ExpEventDictionary.h"
32 // #include "ExpEventLinkDef.h"
33
34 //From Coefficients.cxx
35 extern int nGeOrder; //Order of calibration i.e. 1 = linear.
36 extern int nGeDets; //Total number of signals from Germanium detectors
37 extern int nGeSegments; //number of segments in a single Germanium crystal, for
   ↳ adding purposes
38 extern std::vector<std::vector<double>> dGeCoefficients; //Ge Coefficients
39 extern std::vector<std::vector<double>> dGeCoeffRes; // Ge Residual Coefficients
40 extern std::vector<std::vector<double>> dGeRunCorr; //Run Correction
   ↳ Coefficients
41
42 extern TChain* chain;
43
44 extern int nSiLiOrder; //Order of calibration i.e. 1 = linear.
45 extern int nSiLiDets; //Total number of signals from SiLi
46 extern int nSiLiPlace; //Start of SiLis in generalized array detectors
47 extern std::vector<std::vector<double>> dSiLiCoefficients; //Coefficients
48 extern std::vector<std::vector<double>> dSiLiRunCorr; //Run Correction
   ↳ Coefficients
49
50 //From Constraints.cxx
51 extern std::vector<std::vector<double>> dGeBounds; //bounds for cuts
52 extern std::vector<std::vector<double>> dSiLiBounds; //bounds for cuts
53 extern int nGeConstraints;
54 extern int nSiLiConstraints;
55
56 extern int nBGODets; //Total number of signals from BGO detectors
57 extern int nBGOPlace; //Start of BGOs in generalized array
58 extern std::vector<double> dBГОThreshold;
59
60 //From histograms.cxx
61 extern std::vector<std::vector<TH1F*>> ge_en_ge_cut;

```

```

62 extern std::vector<std::vector<TH1F*> > sili_en_ge_cut;
63 extern std::vector<std::vector<TH1F*> > ge_en_sili_cut;
64 extern std::vector<std::vector<TH1F*> > sili_en_sili_cut;
65
66 using namespace std;
67
68 void analysis::Loop(const char* fileName, int nRunNum, bool bTripleCoin)
69 {
70     // In a ROOT session, you can do:
71     //   Root > .L analysis.C
72     //   Root > analysis t
73     //   Root > t.GetEntry(12); // Fill t data members with entry number 12
74     //   Root > t.Show();      // Show values of entry 12
75     //   Root > t.Show(16);    // Read and show values of entry 16
76     //   Root > t.Loop();     // Loop on all entries
77     //
78
79     // This is the loop skeleton where:
80     // jentry is the global entry number in the chain
81     // ientry is the entry number in the current Tree
82     // Note that the argument to GetEntry must be:
83     //   jentry for TChain::GetEntry
84     //   ientry for TTree::GetEntry and TBranch::GetEntry
85     //
86     // To read only selected branches, Insert statements like:
87     // METHOD1:
88     //   fChain->SetBranchStatus("*",0); // disable all branches
89     //   fChain->SetBranchStatus("branchname",1); // activate branchname
90     // METHOD2: replace line
91     //   fChain->GetEntry(jentry);      //read all branches
92     // by   b_branchname->GetEntry(ientry); //read only this branch
93     if (fChain == 0) return;
94
95     //Don't keep writing to fOut, so switch to the "user" space instead
96     gROOT->cd();
97
98
99     TRandom3 *randgen = new TRandom3(1); //For uniformity over bin, to remove
100    ↵ artifacting
101    std::vector<double> dGeEn;
102    std::vector<double> dSiLiEn;
103    std::vector<double> dBGO;
104
105    //Times
106    std::vector<double> dT_GeEnSeg;
107    std::vector<double> dT_GeEn;
108    std::vector<double> dT_SiLiEn;
109    std::vector<double> dT_BGO;
110
111    //Ge vector
112    for(int i=0; i<nGeDets/nGeSegments;i++)
113    {
114        dGeEn.push_back(0);
115        dT_GeEn.push_back(0);

```

```

115     }
116     for(int i=0; i<nGeDets;i++)
117     {
118         dT_GeEnSeg.push_back(0);
119     }
120     for(int i=0; i<nSiLiDets;i++)
121     {
122         dSiLiEn.push_back(0);
123         dT_SiLiEn.push_back(0);
124     }
125     for(int i=0; i<nBGODets;i++)
126     {
127         dBGO.push_back(0);
128         dT_BGO.push_back(0);
129     }
130
131     //main loop through the data
132     Long64_t nentries = fChain->GetEntriesFast();
133
134     //main loop through the data
135     Long64_t nbytes = 0, nb = 0;
136     for (Long64_t jentry=0; jentry<nentries;jentry++) {
137         Long64_t ientry = LoadTree(jentry);
138         if (ientry < 0) break;
139         nb = fChain->GetEntry(jentry);   nbytes += nb;
140         if (jentry % 1000000 == 0) {
141             cout << "." << flush;
142         }
143         // if (Cut(ientry) < 0) continue;
144         //Germanium calibration
145         for(int i=0 ; i < nGeDets ; i++)
146         {
147             if(i % nGeSegments == 0)
148             {
149                 dGeEn[i/nGeSegments] = 0; //Reset the total clover energy
150                 dT_GeEnSeg[i] = 0; //Reset segment time
151                 dT_GeEn[i/nGeSegments] = 0; //Reset total time
152             }
153             double eneC = 0, dADC = 0;
154             dADC = ge_en_raw[i] + 0.5*(1 - 2*randgen->Rndm());
155             if(ge_time[i] == 0 || buncher_time == 0)
156             {
157                 dT_GeEnSeg[i] = -10000;
158             }
159             else
160             {
161                 dT_GeEnSeg[i] = ge_time[i]-buncher_time;
162             }
163             for(int j=0 ; j<nGeOrder+1 ; j++)
164             {
165                 eneC = eneC + dGeCoefficients[i][j]*pow(dADC,j);
166             }
167             eneC = dGeRunCorr[i][0]+dGeRunCorr[i][1]*eneC; //Run correction factor
168             dGeEn[i] = eneC;

```

```

169         dT_GeEn[i] = dT_GeEnSeg[i]; //This is the line that should fix the
170         ↵ previous problem!
171         // if(i%nGeSegments!=0 && i!=14 && i!=15)
172         //{
173             // dGeEn[i/nGeSegments] = dGeEn[i/nGeSegments]+eneC; //Add energies
174             // if(dT_GeEn[i/nGeSegments] == 0) dT_GeEn[i/nGeSegments] =
175             ↵ dT_GeEnSeg[i]; //Timing
176             //}
177             // if(dADC > 0.5) std::cout << dADC << std::endl;
178         }
179         //SiLi Calibration
180         for(int i=0 ; i < nSiLiDets ; i++)
181     {
182             dSiLiEn[i]=0;
183             dT_SiLiEn[i] = 0;
184             double dADC = 0;
185             dADC = sili_en_raw[i] + 0.5*(1 - 2*randgen->Rndm());
186             if(sili_time[i] == 0 || buncher_time == 0)
187             {
188                 dT_SiLiEn[i] = -10000;
189             }
190             else
191             {
192                 dT_SiLiEn[i] = sili_time[i]-buncher_time;
193             }
194             for(int j=0 ; j<nSiLiOrder+1 ; j++)
195             {
196                 dSiLiEn[i] = dSiLiEn[i] + dSiLiCoefficients[i][j]*pow(dADC,j);
197             }
198             dSiLiEn[i] = dSiLiRunCorr[i][0]+dSiLiRunCorr[i][1]*dSiLiEn[i];
199             ↵ //Run-by-run correction
200         }
201         //BG0s, done dumb because of T. Battaglia not using arrays?!?
202         dBGO[0] = bgo1_en;
203         dBGO[1] = bgo4_en;
204         dBGO[2] = bgo5_en;
205         dBGO[3] = bgo6_en;
206         dT_BGO[0] = bgo1_time-buncher_time;
207         dT_BGO[1] = bgo4_time-buncher_time;
208         dT_BGO[2] = bgo5_time-buncher_time;
209         dT_BGO[3] = bgo6_time-buncher_time;
210
211         //Run the constraints subroutine. At this time, it does use timing gates
212         fillHistograms(nGeConstraints, dGeBounds, dGeEn, dGeEn, dSiLiEn, dBGO,
213             ↵ dT_GeEn, dT_SiLiEn, dT_BGO, true);
214         fillHistograms(nSiLiConstraints, dSiLiBounds, dSiLiEn, dGeEn, dSiLiEn,
215             ↵ dBGO, dT_GeEn, dT_SiLiEn, dT_BGO, false);
216     }
217
218     //File to write out to
219     TFile* fOut = new TFile(fileName,"RECREATE");
220
221     //write to file
222     fOut->cd();

```

```

218     writeHistToFile(fOut);
219     fOut->Close();
220 }
```

C.1.6 src/analysis.h

This header file has not been included, as it is procedurally generated by the MakeClass routine in ROOT. It changes based on the tree directory.

C.1.7 src/histograms.cxx

These are the subroutines for the creation, adding to, and writing to file of the histograms created in the gating process.

```

1 //This file is for the creation of the arrays of histograms to be used with the
2 // cuts code. It also contains the fill histograms command.
3
4 #include <histograms.h>
5
6 //C++ libraries
7 #include <vector>
8 #include <iostream>
9
10 //ROOT libraries
11 #include <TH1F.h>
12 #include <TString.h>
13 #include <TFile.h>
14 #include <TDirectory.h>
15 #include < TObject.h>
16
17 //Libraries within Code
18 #include "timing.h"
19
20 //From Coefficients.cxx
21 extern int nGeOrder; //Order of calibration i.e. 1 = linear.
22 extern int nGeDets; //Total number of signals from Germanium detectors
23 extern int nGeSegments; //number of segments in a single Germanium crystal, for
24 // adding purposes
25
26 extern int nSiLiOrder; //Order of calibration i.e. 1 = linear.
27 extern int nSiLiDets; //Total number of signals from SiLi
28 extern int nSiLiPlace; //Start of SiLis in generalized array detectors
29
30 //From Constraints.cxx
31 extern std::vector<std::vector<double>> dGeBounds; //bounds for cuts
```

```

30  extern std::vector<std::vector<double> > dSiLiBounds; //bounds for cuts
31  extern int nGeConstraints;
32  extern int nSiLiConstraints;
33
34  extern int nBGODets; //Total number of signals from BGO detectors
35  extern int nBGOPlace; //Start of BGOs in generalized array
36  extern std::vector<double> dBГОThreshold;
37
38 //From main.cxx
39 extern const Timing tGates;
40
41 //From this file
42 std::vector<std::vector<TH1F*> > ge_en_ge_cut;
43 std::vector<std::vector<TH1F*> > sili_en_ge_cut;
44 std::vector<std::vector<TH1F*> > ge_en_sili_cut;
45 std::vector<std::vector<TH1F*> > sili_en_sili_cut;
46
47
48 //For regular coincidence
49 void makeHistograms(int nGeDets, int nGeCuts, int nSiLiDets, int nSiLiCuts)
50 {
51     //Vector to make a new row
52     std::vector<TH1F*> row;
53     //Loop through and construct them all.
54     for(int j=0 ; j < nGeDets ; j++)
55     {
56         ge_en_ge_cut.push_back(row);
57         ge_en_sili_cut.push_back(row);
58         for(int i=0 ; i < nGeCuts ; i++)
59         {
60             ge_en_ge_cut[j].push_back( new TH1F(Form("ge_en_ge_cut[%i] [%i]",j,i),Form(" "
61             → Clover_%i",j),4001,0.5,4000.5));
62         }
63         for(int i=0 ; i < nSiLiCuts ; i++)
64         {
65             ge_en_sili_cut[j].push_back(new TH1F(Form("sili_en_ge_cut[%i] [%i]",j,i),For )
66             → m("Clover_%i",j+1),4001,0.5,4000.5));
67         }
68     }
69     for(int j=0 ; j < nSiLiDets ; j++)
70     {
71         sili_en_ge_cut.push_back(row);
72         sili_en_sili_cut.push_back(row);
73         for(int i=0 ; i < nGeCuts ; i++)
74         {
75             sili_en_ge_cut[j].push_back(new TH1F(Form("ge_en_sili_cut[%i] [%i]",j,i),For )
76             → m("SiLi_%i",j),4001,0.5,4000.5));
77         }
78         for(int i=0 ; i < nSiLiCuts ; i++)
79         {
80             sili_en_sili_cut[j].push_back(new TH1F(Form("sili_en_sili_cut[%i] [%i]",j,i) )
81             → ,Form("SiLi_%i",j+1),4001,0.5,4000.5));
82         }
83     }
84 }
```

```

80    }
81
82    //Regular, singular coincidence
83    void fillHistograms(int nConstraints, std::vector<std::vector<double>>
84        >dConstraints, std::vector<double> dEnCut, std::vector<double> dGeDetectors,
85        std::vector<double> dSiLiDetectors, std::vector<double> dBGO,
86        std::vector<double> dGeV, std::vector<double> dSiLiT, std::vector<double>
87        dBGOT, bool GeorSiLi/*true = ge gate, false = sili gate*/)
88    {
89        for(int i=0; i<nConstraints ; i++)
90        {
91            if((dEnCut[(int)dConstraints[i][0]] > dConstraints[i][1] && //gate detector
92                ← lower limit
93                dEnCut[(int)dConstraints[i][0]] < dConstraints[i][2] && //gate detector
94                ← upper limit
95                GeorSiLi
96                //dBGO[(int)dConstraints[i][0]] < dBГОThreshold[(int)dConstraints[i][0]])
97                // //BGO Threshold met
98                )|| //OR (above is Ge constraint, below is SiLi)
99                (dEnCut[(int)dConstraints[i][0]] > dConstraints[i][1] && //gate detector
100                ← lower limit
101                dEnCut[(int)dConstraints[i][0]] < dConstraints[i][2] && //gate detector
102                ← upper limit
103                !GeorSiLi) //is SiLi
104            {
105                //Fill Germanium histograms, includes BGO thresholds and timing
106                for(int j = 0 ; j < nGeDets/nGeSegments ; j++)
107                {
108                    if(GeorSiLi && (int)dConstraints[i][0] != j) // if Ge gate AND gate
109                    ← detector is not this detector AND BGO is under threshold)
110                    {
111                        //Timing check - ge-ge
112                        for(int k = 0; k < tGates.geGe.size(); k++)
113                        {
114                            //if( (((dGeV[(int)dConstraints[i][0]]-dGeV[j])>=tGates.geGe[k][2] &&
115                            ← (dGeV[(int)dConstraints[i][0]]-dGeV[j])<=tGates.geGe[k][3]) ||
116                            ← ((dGeV[j]-dGeV[(int)dConstraints[i][0]])>=tGates.geGe[k][2] &&
117                            ← (dGeV[j]-dGeV[(int)dConstraints[i][0]])<=tGates.geGe[k][3]))
118                            ← std::cout << "Within timing gate for Ge-Ge" << std::endl;
119                            if((int)dConstraints[i][0]==tGates.geGe[k][0] && j==tGates.geGe[k][1])
120                            {
121                                if( (dGeV[(int)dConstraints[i][0]]-dGeV[j])>=tGates.geGe[k][2] &&
122                                    ← (dGeV[(int)dConstraints[i][0]]-dGeV[j])<=tGates.geGe[k][3] &&
123                                    dGeDetectors[j] > 5) ge_en_ge_cut[j][i]->Fill(dGeDetectors[j]);
124                            }
125                            else if(j==tGates.geGe[k][0] &&
126                                (int)dConstraints[i][0]==tGates.geGe[k][1])
127                            {
128                                if( (dGeV[j]-dGeV[(int)dConstraints[i][0]])>=tGates.geGe[k][2] &&
129                                    ← (dGeV[j]-dGeV[(int)dConstraints[i][0]])<=tGates.geGe[k][3] &&
130                                    dGeDetectors[j] > 5) ge_en_ge_cut[j][i]->Fill(dGeDetectors[j]);
131                            }
132                        }
133                    }
134                }

```

```

115     else if (!GeorSiLi)
116     {
117         //Timing check ge-sili
118         for(int k = 0; k < tGates.geSiLi.size(); k++)
119         {
120             if((int)dConstraints[i][0]==tGates.geSiLi[k][1] &&
121                 → j==tGates.geSiLi[k][0])
122             {
123                 if( (dGeT[j]-dSiLiT[(int)dConstraints[i][0]])>=tGates.geSiLi[k][2]
124                     → &&
125                     → (dGeT[j]-dSiLiT[(int)dConstraints[i][0]])<=tGates.geSiLi[k][3]
126                     → && dGeDetectors[j] > 5)
127                     → ge_en_sili_cut[j][i]->Fill(dGeDetectors[j]);
128             }
129         }
130     }
131 }
132 }
133 //Fill SiLi histograms, includes timing
134 for(int j = 0 ; j < nSiLiDets ; j++)
135 {
136     if(GeorSiLi)
137     {
138         //Timing check ge-sili
139         for(int k = 0; k < tGates.geSiLi.size(); k++)
140         {
141             if(j==tGates.geSiLi[k][1] &&
142                 → (int)dConstraints[i][0]==tGates.geSiLi[k][0])
143             {
144                 if( (dGeT[(int)dConstraints[i][0]]-dSiLiT[j])>=tGates.geSiLi[k][2]
145                     → &&
146                     → (dGeT[(int)dConstraints[i][0]]-dSiLiT[j])<=tGates.geSiLi[k][3]
147                     → && dSiLiDetectors[j] > 20)
148                     → sili_en_ge_cut[j][i]->Fill(dSiLiDetectors[j]);
149             }
150         }
151     }
152 }
```

```

152         if( (dSiLiT[j]-dSiLiT[(int)dConstraints[i][0]])>=tGates.siliSiLi[k] )
153             ↵ [2] &&
154             ↵ (dSiLiT[j]-dSiLiT[(int)dConstraints[i][0]])<=tGates.siliSiLi[k] )
155             ↵ [3] && dSiLiDetectors[j] > 20)
156             ↵ sili_en_sili_cut[j][i]->Fill(dSiLiDetectors[j]);
157     }
158 }
159 }
160
161
162 void writeHistToFile(TFile* fOut)
163 {
164     for(int i=0; i<nGeConstraints ; i++)
165     {
166         fOut->cd();
167         gDirectory->mkdir(Form("Ge Cut Detector %i, Peak
168             ↵ %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
169         fOut->cd(Form("Ge Cut Detector %i, Peak
170             ↵ %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
171         for(int j=0; j<nGeDets/nGeSegments ; j++)
172         {
173             ge_en_ge_cut[j][i]->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
174         }
175         for(int j=0; j<nSiLiDets ; j++)
176         {
177             sili_en_ge_cut[j][i]->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
178         }
179     }
180     for(int i=0; i<nSiLiConstraints ; i++)
181     {
182         fOut->cd();
183         gDirectory->mkdir(Form("SiLi Cut Detector %i, Peak
184             ↵ %f", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
185         fOut->cd(Form("SiLi Cut Detector %i, Peak
186             ↵ %f", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
187         for(int j=0; j<nGeDets/nGeSegments ; j++)
188         {
189             ge_en_sili_cut[j][i]->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
190         }
191         for(int j=0; j<nSiLiDets ; j++)
192         {
193             sili_en_sili_cut[j][i]->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
194         }
195     }
196     fOut->Write();
197 }
```

C.1.8 include/histograms.h

Header file for histograms.cxx.

```
1  #ifndef HISTOGRAMS_H
2  #define HISTOGRAMS_H
3
4  #ifndef _HISTOGRAMS_CXX
5  #define EXTERNAL extern
6  #else
7  #define EXTERNAL
8  #endif
9
10 //C++ libraries
11 #include <vector>
12
13 //ROOT libraries
14 #include <TH1F.h>
15 #include <TString.h>
16 #include <TFile.h>
17
18 void makeHistograms(int nGeDets, int nGeCuts, int nSiLiDets, int nSiLiCuts);
19 void fillHistograms(int nConstraints, std::vector<std::vector<double>>
20                     >dConstraints, std::vector<double> dEnCut, std::vector<double> dGeDetectors,
21                     std::vector<double> dSiLiDetectors, std::vector<double> dBGO,
22                     std::vector<double> dGeV, std::vector<double> dSiLiT, std::vector<double>
23                     dBGOT, bool GeorSiLi);
24 void writeHistToFile(TFile* fOut);
25
26 #endif
```

C.1.9 src/timing.cxx

This is a class for keeping track of the timing gates used between pairs of detectors.

```
1  #define timing_cxx
2  #include "timing.h"
3
4  //C++ libraries
5  #include <iostream>
6  #include <cstdio>
7  #include <fstream>
8  #include <string>
9  #include <cmath>
10 #include <vector>
11
12 //ROOT Libraries
```

```

13  #include "TString.h"
14
15  // class Timing
16 //{
17  // public:
18  //     std::vector<std::vector<double>> geGe;
19  //     std::vector<std::vector<double>> geSiLi;
20  //     std::vector<std::vector<double>> siliSiLi;
21  //     Timing(int,int,char* );
22  // };
23
24  Timing::Timing ()
25  {
26      //Just make a thing
27  }
28
29  Timing::Timing (int nGe, int nSiLi, char* cTiming)
30  {
31      std::string buffer;
32      int nCounter = 0;
33      if(strcmp(cTiming,"") == 0) //No file provided
34      {
35          cTiming = "Timing";
36      }
37      std::fstream fTiming(Form("user/%s.dat",cTiming)); //Timing File
38      if(!fTiming.is_open())
39      {
40          std::cout << "Timing file did not open. Ending program." << std::endl;
41          //Need to add a flag somehow that this is void.
42      }
43      getline (fTiming,buffer); //Label Line
44      getline (fTiming,buffer); //Timing Line
45      std::vector<double> row; //Row for adding a level in.
46      while(strncmp(buffer.c_str(),"//ge-sili",2) != 0) //Loop through all the ge
→ Detectors
47      {
48          geGe.push_back(row); //Put a new row of timing
49          for(int k=0; k< 4; k++) //Loop through detector 1, detector 2, low time,
→ high time.
50      {
51          geGe[nCounter].push_back(std::atof(buffer.substr(0,buffer.find_first_
→ of(',',0)).c_str())); //read in jth number in
→ line
52          buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_firs_
→ t_of('\n',0)); //make a substring of the rest of the
→ line
53      }
54      nCounter++;
55      getline (fTiming,buffer); //Line with timing in it
56  }
57  std::cout << "Ge-ge timing read in" << std::endl;
58  getline (fTiming,buffer); //Line with timing in it
59  nCounter = 0;
60  while(strncmp(buffer.c_str(),"//sili-sili",2) != 0)

```

```

61     {
62         geSiLi.push_back(row); //Put a new row of timing
63         for(int k=0; k< 4; k++) //Loop through detector 1, detector 2, low time,
64             // high time.
65         {
66             geSiLi[nCounter].push_back(std::atof(buffer.substr(0,buffer.find_firs_]
67             // t_of(',',0).c_str())); //read in jth number in
68             // line
69             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_firs_]
70             // t_of('\n',0)); //make a substring of the rest of the
71             // line
72         }
73         nCounter++;
74         getline (fTiming,buffer); //Line with timing in it
75     }
76     std::cout << "Ge-sili timing read in" << std::endl;
77     nCounter=0;
78     getline (fTiming,buffer); //Line with timing in it
79     while(!fTiming.eof()) //Loop through all the sili Detectors
80     {
81         siliSiLi.push_back(row); //Put a new row of timing
82         for(int k=0; k< 4; k++) //Loop through detector 1, detector 2, low time,
83             // high time.
84         {
85             siliSiLi[nCounter].push_back(std::atof(buffer.substr(0,buffer.find_fi_]
86             // rst_of(',',0).c_str())); //read in jth number in
87             // line
88             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_firs_]
89             // t_of('\n',0)); //make a substring of the rest of the
90             // line
91         }
92         nCounter++;
93         getline (fTiming,buffer); //Line with timing in it
94     }
95     std::cout << "Sili-sili timing read in" << std::endl;
96 }
```

C.1.10 include/timing.h

Header file for timing.cxx.

```

1 ///////////////////////////////////////////////////////////////////
2 // This class has been automatically generated on
3 // Thu Jan 11 12:37:54 2018 by ROOT version 5.34/19
4 // from TTree evtTree/133Ba_90deg_facingdown_trig(coin1(siliall+ge1+ge2) +
5 // pulser)
6 // found on file: rootFiles/Data/run1.root
7 ///////////////////////////////////////////////////////////////////
8 #ifndef timing_h
```

```

9  #define timing_h
10
11 //C++ libraries
12 #include <iostream>
13 #include <fstream>
14 #include <string>
15 #include <cmath>
16 #include <vector>
17
18 // Fixed size dimensions of array or collections stored in the TTree if any.
19
20 class Timing
21 {
22     public:
23         std::vector<std::vector<double>> geGe;
24         std::vector<std::vector<double>> geSiLi;
25         std::vector<std::vector<double>> siliSiLi;
26     Timing();
27     Timing(int,int,char*);
28 };
29
30 #endif
31
32 #ifdef timing_cxx
33 #endif // #ifdef timing_cxx

```

C.2 Merge Code

C.2.1 README.md

Simple included explanation of code inputs for running. Includes important commentary on inputs when used in conjunction with the coincidence code.

```

1 # Merge-Parallel
2
3 This code is to be used in conjunction with ICEBall-Coincidence-Code.
4
5 Inputs for the code are as follows (must be in this order)
6 1. First sequential run
7 2. Last sequential run
8 3. File output name (should be the same as the name used for
    ↳ ICEBall-Coincidence-Code)
9 4. Cut File Name (Should be the same as the name used for
    ↳ ICEBall-Coincidence-Code)

```

C.2.2 Makefile

The compilation code.

```
1  #to compile using root libraries
2  ROOTCFLAGS := $(shell root-config --cflags)
3  ROOTLIBS := $(shell root-config --libs)
4  ROOTGLIBS := $(shell root-config --glibs)
5
6  #this is the compiler that will be used:
7  CXX = g++
8  #these are some compiler flags
9  CXXFLAGS = -O2 -fPIC -g $(ROOTCFLAGS)
10
11 #folder including all the header files
12 INCLUDES := -Iinclude -Iuser
13 #folder with all the source files
14 SOURCES := src
15 #source files
16 FILES := Constraints.cxx Coefficients.cxx histograms.cxx Filelist.cxx main.cxx
17 #this is the executable name
18 EXECUTABLES := main
19 #libraries
20 LIBS = $(ROOTLIBS) $(SYSLIBS)
21 #this are object files (take the names of sources and change the extensions)
22 OBJECTS := $(FILES:.cxx=.o)
23
24 #don't do anything to files named 'clean' and 'rmobjects'
25 .PHONY: clean rmobjects
26
27 $(EXECUTABLES): $(OBJECTS)
28   $(CXX) $(LIBS) $(INCLUDES) $(CXXFLAGS) $(OBJECTS) -o $(EXECUTABLES)
29
30 #compile all source files and generate object files
31 # %.cxx = any file that matches the pattern
32 # $< = name of the first dependency file
33 # $@ = name of the target file
34 %.o: $(SOURCES)/%.cxx
35   $(CXX) $(INCLUDES) $(CXXFLAGS) -c $< -o $@
36
37 #this will remove the executable file and objects
38 clean:
39   rm -rf $(EXECUTABLES) $(OBJECTS)
40
41 rmobjects:
42   rm -rf $(OBJECTS)
```

C.2.3 src/main.cxx

This is the main code that calls all the subroutines. The main subroutine used is in the histograms code. "\$OutputDirectory\$ must be replaced with the output directory.

```
1  /*
2  Please check the README for more information about this code and it's purpose.
3  */
4
5  //C++ libraries
6  #include <iostream>
7  #include <fstream>
8  #include <string>
9  #include <cmath>
10 #include <cstdio>
11 #include <cstdlib>
12 #include <vector>
13
14 //ROOT libraries
15 #include <TROOT.h>
16 #include "TH2.h"
17 #include "TStyle.h"
18 #include "TCanvas.h"
19 #include "TString.h"
20 #include "TFile.h"
21 #include "TChain.h"
22 #include "TH1F.h"
23 #include "TTree.h"
24 #include "TRandom3.h"
25 #include " TBranch.h"
26 #include " TLeaf.h"
27 #include " TObject.h"
28 #include " TObjArray.h"
29
30 //Object files
31 #include "Coefficients.h"
32 #include "Constraints.h"
33 #include "histograms.h"
34 #include "Filelist.h"
35
36 using namespace std;
37
38 //File path info
39 extern std::string sFilepath;
40 extern std::string sTree;
41 extern std::string sRun;
42 extern std::string sFType;
43
```

```

44 //from histograms.cxx
45 extern std::vector<std::vector<TH1F*>> ge_en_ge_cut;
46 extern std::vector<std::vector<TH1F*>> sili_en_ge_cut;
47 extern std::vector<std::vector<TH1F*>> ge_en_sili_cut;
48 extern std::vector<std::vector<TH1F*>> sili_en_sili_cut;
49
50 //From coefficients.cxx
51 extern int nGeOrder; //Order of calibration i.e. 1 = linear.
52 extern int nGeDets; //Total number of signals from Germanium detectors
53 extern int nGeSegments; //number of segments in a single Germanium crystal, for
→ adding purposes
54
55 extern int nSiLiOrder; //Order of calibration i.e. 1 = linear.
56 extern int nSiLiDets; //Total number of signals from SiLi
57 extern int nSiLiPlace; //Start of SiLis in generalized array detectors
58
59 std::vector<std::vector<double>> dGeBounds; //bounds for cuts
60 std::vector<std::vector<double>> dSiLiBounds; //bounds for cuts
61 int nGeConstraints;
62 int nSiLiConstraints;
63
64 std::vector<std::vector<double>> dGeGeBounds; //bounds for cuts
65 std::vector<std::vector<double>> dGeSiLiBounds; //bounds for cuts
66 std::vector<std::vector<double>> dSiLiSiLiBounds; //bounds for cuts
67 int nGeGeConstraints;
68 int nGeSiLiConstraints;
69 int nSiLiSiLiConstraints;
70
71 int main(int argc, char* argv[]) //Order of arguments: first run, last run,
→ output file name, cut file name
72 {
73     char buffer[50];
74     int nRunStart = atoi(argv[1]);
75     int nRunEnd = atoi(argv[2]);
76     char* sOut = argv[3];
77     char* sCut1 = argv[4];
78     char* sCut2 = argv[5];
79     readPaths();
80     defineGeCoeff(); //From Coefficients.cxx
81     defineSiLiCoeff(); //From Coefficients.cxx
82     if(strcmp(sCut2,"0")==0)
83     {
84         sprintf(buffer,"GeCut_%s.dat",sCut1); //File name to input
85         nGeConstraints = defineConstraints(buffer,dGeBounds); //From constraints.cxx
86         sprintf(buffer,"SiLiCut_%s.dat",sCut1); //File name to input
87         nSiLiConstraints = defineConstraints(buffer,dSiLiBounds);
88         std::cout << "Regular Coincidence" << std::endl;
89         writeHistToFile(Form("/scratch365/sstrauss/%s.root",sOut),sOut,nRunStart,nRun ]
→ End, false); //fOut is in
→ Filelist.h
90     }
91     else
92     {
93         sprintf(buffer,"Cut_Ge_%s_Ge_%s.dat",sCut1,sCut2); //File name to input

```

```

94     nGeGeConstraints = defineTripleConstraints(buffer,dGeGeBounds); //From
95     ↪   constraints.cxx
96     sprintf(buffer,"Cut_Ge_%s_SiLi_%s.dat",sCut1,sCut2); //File name to input
97     nGeSiLiConstraints = defineTripleConstraints(buffer,dGeSiLiBounds); //From
98     ↪   constraints.cxx
99     sprintf(buffer,"Cut_SiLi_%s_SiLi_%s.dat",sCut1,sCut2); //File name to input
100    nSiLiSiLiConstraints = defineTripleConstraints(buffer,dSiLiSiLiBounds);
101    ↪   //From constraints.cxx
102    std::cout << "Triple Coincidence" << std::endl;
103    writeHistToFile(Form("/scratch365/sstrauss/%s.root",sOut),sOut,nRunStart,nRun ]
104      ↪   End, true); //fout is in
105      ↪   Filelist.h
106  }
107  return 0;
108 }
```

C.2.4 src/histograms.cxx

This is the code that combines the histograms of the separate files into one.

```

1  //This file is for the creation of the arrays of histograms to be used with the
2  ↪   cuts code. It also contains the fill histograms command.
3
4
5  //C++ libraries
6  #include <vector>
7  #include <iostream>
8
9  //ROOT libraries
10 #include <TH1F.h>
11 #include <TString.h>
12 #include <TFile.h>
13 #include <TDirectory.h>
14 #include < TObject.h>
15 #include <TROOT.h>
16
17 using namespace std;
18
19 //File path info
20 extern std::string sFilepath;
21 extern std::string sTree;
22 extern std::string sRun;
23 extern std::string sFType;
24
25 //From Constraints.cxx
26 extern std::vector<std::vector<double> > dGeBounds; //bounds for cuts
27 extern std::vector<std::vector<double> > dSiLiBounds; //bounds for cuts
28 extern int nGeConstraints;
29 extern int nSiLiConstraints;
```

```

30
31 //From coefficients.cxx
32 extern int nGeOrder; //Order of calibration i.e. 1 = linear.
33 extern int nGeDets; //Total number of signals from Germanium detectors
34 extern int nGeSegments; //number of segments in a single Germanium crystal, for
   ↵ adding purposes
35
36 extern int nSiLiOrder; //Order of calibration i.e. 1 = linear.
37 extern int nSiLiDets; //Total number of signals from SiLi
38 extern int nSiLiPlace; //Start of SiLis in generalized array detectors
39
40 //From this file
41 void writeHistToFile(char* fileName, char* sOut, int nRunStart, int nRunStop,
   ↵ bool bTripleCoin)
42 {
43     TH1F* temp;
44     TH1F* toAdd;
45     TFile* fOut = new TFile(fileName, "RECREATE");
46     std::cout << fileName << std::endl;
47     TFile* fIn = new
        ↵ TFile(Form("%s%s_run_00%i.root", sFilepath.c_str(), sOut, nRunStart));
48     std::cout << Form("%s%s_run_00%i.root", sFilepath.c_str(), sOut, nRunStart) <<
        ↵ std::endl;
49     for(int i=0; i<nGeConstraints ; i++)
50     {
51         fIn->cd(Form("Ge Cut Detector %i, Peak
           ↵ %f", (int)dGeBounds[i][0], (dGeBounds[i][1]+dGeBounds[i][2])/2));
52         fOut->cd();
53         gDirectory->mkdir(Form("Ge Cut Detector %i, Peak
           ↵ %f", (int)dGeBounds[i][0], (dGeBounds[i][1]+dGeBounds[i][2])/2));
54         fOut->cd(Form("Ge Cut Detector %i, Peak
           ↵ %f", (int)dGeBounds[i][0], (dGeBounds[i][1]+dGeBounds[i][2])/2));
55         for(int j=0; j<nGeDets/nGeSegments ; j++)
56         {
57             /*(TH1F*)*/(fIn->Get(Form("Ge Cut Detector %i, Peak
           ↵ %f/Clover_%i", (int)dGeBounds[i][0], (dGeBounds[i][1]+dGeBounds[i][2])/2, j))
           ↵ )->Write(Form("Clover_%i", j), TObject::kOverwrite, 1E8);
58         }
59         for(int j=0; j<nSiLiDets ; j++)
60         {
61             /*(TH1F*)*/(fIn->Get(Form("Ge Cut Detector %i, Peak
           ↵ %f/SiLi_%i", (int)dGeBounds[i][0], (dGeBounds[i][1]+dGeBounds[i][2])/2, j+1))
           ↵ )->Write(Form("SiLi_%i", j+1), TObject::kOverwrite, 1E8);
62         }
63     }
64     for(int i=0; i<nSiLiConstraints ; i++)
65     {
66         fIn->cd(Form("SiLi Cut Detector %i, Peak
           ↵ %f", (int)dSiLiBounds[i][0]+1, (dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
67         fOut->cd();
68         gDirectory->mkdir(Form("SiLi Cut Detector %i, Peak
           ↵ %f", (int)dSiLiBounds[i][0]+1, (dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
69         fOut->cd(Form("SiLi Cut Detector %i, Peak
           ↵ %f", (int)dSiLiBounds[i][0]+1, (dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));

```

```

70     for(int j=0; j<nGeDets/nGeSegments ; j++)
71     {
72       /*(TH1F*)*/(fIn->Get(Form("SiLi Cut Detector %i, Peak
73         ↪  %f/Clover_%i", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][
74           ↪  2])/2,j)))->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
75     }
76     for(int j=0; j<nSiLiDets ; j++)
77     {
78       /*(TH1F*)*/(fIn->Get(Form("SiLi Cut Detector %i, Peak
79         ↪  %f/SiLi_%i", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][
80           ↪  2])/2,j+1)))->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
81   }
82   for(int nRunNum = nRunStart+1; nRunNum <=nRunStop; nRunNum++)
83   {
84     TFile* fIn = new
85       ↪  TFile(Form("%s%s_run_00%i.root",sFilepath.c_str(),sOut,nRunNum));
86     std::cout << Form("%s%s_run_00%i.root",sFilepath.c_str(),sOut,nRunNum) <<
87       ↪  std::endl;
88     if(fIn->IsZombie()) continue;
89     for(int i=0; i<nGeConstraints ; i++)
90     {
91       fIn->cd(Form("Ge Cut Detector %i, Peak
92         ↪  %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
93       fOut->cd();
94       fOut->cd(Form("Ge Cut Detector %i, Peak
95         ↪  %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
96       for(int j=0; j<nGeDets/nGeSegments ; j++)
97       {
98         gROOT->cd();
99         temp = (TH1F*)(fOut->Get(Form("Ge Cut Detector %i, Peak %f/Clover_%i",(in_
100           ↪  t)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j)));
101         toAdd = (TH1F*)(fIn->Get(Form("Ge Cut Detector %i, Peak %f/Clover_%i",(in_
102           ↪  t)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j)));
103         temp->Add(toAdd);
104         fOut->cd(Form("Ge Cut Detector %i, Peak
105           ↪  %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
106         temp->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
107         //for(int bin=0; bin < temp->GetSize(); bin++)
108           ↪  temp->AddBinContent(bin,toAdd->GetBinContent(bin));
109       }
110     for(int j=0; j<nSiLiDets ; j++)
111     {
112       gROOT->cd();
113       temp = (TH1F*)(fOut->Get(Form("Ge Cut Detector %i, Peak %f/SiLi_%i", (int)j
114         ↪  dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j+1)));
115       toAdd = (TH1F*)fIn->Get(Form("Ge Cut Detector %i, Peak %f/SiLi_%i", (int)d_
116         ↪  GeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j+1));
117       temp->Add(toAdd);
118       fOut->cd(Form("Ge Cut Detector %i, Peak
119         ↪  %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
120       temp->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
121     }
122   }

```

```

109     for(int i=0; i<nSiLiConstraints ; i++)
110     {
111         fIn->cd(Form("SiLi Cut Detector %i, Peak
112             ↪ %f", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
113         fOut->cd();
114         fOut->cd(Form("SiLi Cut Detector %i, Peak
115             ↪ %f", (int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
116         for(int j=0; j<nGeDets/nGeSegments ; j++)
117         {
118             gROOT->cd();
119             temp = (TH1F*)(fOut->Get(Form("SiLi Cut Detector %i, Peak %f/Clover_%i", (int)
120                 ↪ dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j)));
121             toAdd = (TH1F*)fIn->Get(Form("SiLi Cut Detector %i, Peak %f/Clover_%i", (int)
122                 ↪ dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j));
123             temp->Add(toAdd);
124             fOut->cd(Form("SiLi Cut Detector %i, Peak %f", (int)dSiLiBounds[i][0]+1,(d
125                 ↪ SiLiBounds[i][1]+dSiLiBounds[i][2])/2));
126             temp->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
127         }
128         for(int j=0; j<nSiLiDets ; j++)
129         {
130             gROOT->cd();
131             temp = (TH1F*)(fOut->Get(Form("SiLi Cut Detector %i, Peak %f/SiLi_%i", (int)
132                 ↪ dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j+1)));
133             toAdd = (TH1F*)fIn->Get(Form("SiLi Cut Detector %i, Peak %f/SiLi_%i", (int)
134                 ↪ dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j+1));
135             temp->Add(toAdd);
136             fOut->cd(Form("SiLi Cut Detector %i, Peak %f", (int)dSiLiBounds[i][0]+1,(d
137                 ↪ SiLiBounds[i][1]+dSiLiBounds[i][2])/2));
138             temp->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
139         }
140         fIn->Close();
141         delete fIn;
142     }
143     for(int i=0; i<nGeConstraints ; i++)
144     {
145         fOut->cd();
146         fOut->cd(Form("Ge Cut Detector %i, Peak
147             ↪ %f", (int)dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2));
148         for(int j=0; j<nGeDets/nGeSegments ; j++)
149         {
150             temp = (TH1F*)(fOut->Get(Form("Ge Cut Detector %i, Peak %f/Clover_%i", (int)
151                 ↪ dGeBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j)));
152             temp->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
153             //for(int bin=0; bin < temp->GetSize(); bin++)
154             //    temp->AddBinContent(bin,toAdd->GetBinContent(bin));
155         }
156         for(int j=0; j<nSiLiDets ; j++)
157         {
158             temp = (TH1F*)(fOut->Get(Form("Ge Cut Detector %i, Peak %f/SiLi_%i", (int)dG
159                 ↪ eBounds[i][0],(dGeBounds[i][1]+dGeBounds[i][2])/2,j+1)));
160             temp->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
161         }

```

```

151     }
152     for(int i=0; i<nSiLiConstraints ; i++)
153     {
154         fOut->cd();
155         fOut->cd(Form("SiLi Cut Detector %i, Peak
156             ↪ %f",(int)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2));
157         for(int j=0; j<nGeDets/nGeSegments ; j++)
158         {
159             temp = (TH1F*)(fOut->Get(Form("SiLi Cut Detector %i, Peak %f/Clover_%i",(int)
160                 ↪ t)dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j)));
161             temp->Write(Form("Clover_%i",j),TObject::kOverwrite,1E8);
162         }
163         for(int j=0; j<nSiLiDets ; j++)
164         {
165             temp = (TH1F*)(fOut->Get(Form("SiLi Cut Detector %i, Peak %f/SiLi_%i",(int)
166                 ↪ dSiLiBounds[i][0]+1,(dSiLiBounds[i][1]+dSiLiBounds[i][2])/2,j+1)));
167             temp->Write(Form("SiLi_%i",j+1),TObject::kOverwrite,1E8);
168         }
169     }
170     std::cout << "Finishing it up" << std::endl;
171     //fOut->Write();
172     fOut->Close();
173     std::cout << "File closed" << std::endl;
174     return;
175 }
```

C.2.5 include/histograms.h

Header file for histograms.cxx.

```

1  #ifndef HISTOGRAMS_H
2  #define HISTOGRAMS_H
3
4  #ifndef _HISTOGRAMS_CXX
5  #define EXTERNAL extern
6  #else
7  #define EXTERNAL
8  #endif
9
10 //C++ libraries
11 #include <vector>
12
13 //ROOT libraries
14 #include <TH1F.h>
15 #include <TString.h>
16 #include <TFile.h>
17
18 void writeHistToFile(char* fileName, char* sOut, int nRunStart, int nRunStop);
19
20 #endif
```

C.3 Code shared between Coincidence and Merge

C.3.1 src/Coefficients.cxx

These are the subroutines for reading in calibration coefficients for the Si(Li) and HPGe detectors.

```
1 #include "Constraints.h"
2
3 //C++ libraries
4 #include <iostream>
5 #include <fstream>
6 #include <string>
7 #include <cmath>
8 #include <vector>
9 #include <cstdlib>
10
11 //ROOT libraries
12 #include <TString.h>
13
14 using namespace std;
15
16 int nGeOrder; //Order of calibration i.e. 1 = linear.
17 int nGeDets; //Total number of signals from Germanium detectors
18 int nGeSegments; //number of segments in a single Germanium crystal, for adding
→ purposes
19 std::vector<std::vector<double> > dGeCoefficients; //Coefficients
20 std::vector<std::vector<double> > dGeCoeffRes; //Residual Coefficients
21 std::vector<std::vector<double> > dGeRunCorr; //Run Correction Coefficients
22
23 int nSiLiOrder; //Order of calibration i.e. 1 = linear.
24 int nSiLiDets; //Total number of signals from SiLi
25 int nSiLiPlace; //Start of SiLis in generalized array detectors
26 std::vector<std::vector<double> > dSiLiCoefficients; //Coefficients
27 std::vector<std::vector<double> > dSiLiRunCorr; //Run Correction Coefficients
28
29 void defineGeCoeff() //Get Ge coefficients. Does not include run-by-run
→ corrections
30 {
31     //First thing: read in the coefficients for this run.
32     fstream fCoeff("user/GeCoefficients.dat"); //Coefficient File, before
→ run-by-run corrections
33     if(!fCoeff.is_open())
34     {
35         cout << "Ge file did not open" << endl;
36         return;
37     }
38     string buffer;
39     std::getline(fCoeff,buffer);
```

```

40     nGeOrder = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
41     //So this clusterfuck is due to gcc being a pain. it basically converts a
42     //substring of the string into a char to convert into an int because
43     //apparently it won't just go string to int
44     getline (fCoeff,buffer);
45     nGeDets = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
46     getline (fCoeff,buffer);
47     nGeSegments = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
48     getline (fCoeff,buffer); //Label Line
49     std::vector<double> row; //Row for adding a level in.
50     //Okay, here, we get into the nitty gritty
51     for(int i=0; i<nGeDets;i++) //Loop through all the detectors
52     {
53         getline (fCoeff,buffer); //Line with coefficients on it.
54         dGeCoefficients.push_back(row); //Put a new row in for the detector
55         for(int j=0; j<= nGeOrder; j++) //Loop through the coefficients
56         {
57             dGeCoefficients[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of(
58                 ',' ,0)).c_str())); //read in jth
59             // coefficient
60             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_first_of(
61                 '\n',0)); //make a substring of the rest of the
62             // coefficients
63         }
64     }
65     getline (fCoeff,buffer); //Line with label on it
66     for(int i=0; i<nGeDets;i++) //Loop through all the detectors
67     {
68         getline (fCoeff,buffer); //Line with coefficients on it.
69         dGeCoeffRes.push_back(row); //Put a new row in for the detector
70         for(int j=0; j< 7; j++) //Loop through the residual coefficients. Currently
71             // hardcoded in.
72         {
73             dGeCoeffRes[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of(
74                 ',' ,0)).c_str())); //read in jth
75             // coefficient
76             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_first_of(
77                 '\n',0)); //make a substring of the rest of the
78             // coefficients
79         }
80     }
81     fCoeff.close(); //Close the coefficients file
82 }
83
84 void defineGeCoeff(int nRunNum) //Ge Coefficients for the run based corrections.
85 {
86     //First thing: read in the coefficients for this run.
87     fstream fCoeff(Form("user/Run_by_Run/GeCoefficients_r%i.dat",nRunNum));
88     //Coefficient File
89     if(!fCoeff.is_open())
90     {
91         cout << "Ge run file did not open, setting correction to y=x" << endl;
92         std::vector<double> row; //Row for adding a level in.
93         for(int i=0; i<nGeDets;i++)

```

```

81     {
82         dGeRunCorr.push_back(row);
83         dGeRunCorr[i].push_back(0);
84         dGeRunCorr[i].push_back(1);
85     }
86     return;
87 }
88 string buffer;
89 std::getline(fCoeff,buffer); //Label
90 std::vector<double> row; //Row for adding a level in.
//Okay, here, we get into the nitty gritty
91 for(int i=0; i<nGeDets;i++) //Loop through all the detectors
92 {
93     getline (fCoeff,buffer); //Line with coefficients on it.
94     dGeRunCorr.push_back(row); //Put a new row in for the detector
95     for(int j=0; j<= 1; j++) //Loop through the coefficients, assuming linear
96     {
97         dGeRunCorr[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of('
98             → ,',0)).c_str())); //read in jth
99             → coefficient
100            buffer = buffer.substr(buffer.find_first_of(' ',0)+1,buffer.find_first_o
101                → f('\n',0)); //make a substring of the rest of the
102                    → coefficients
103    }
104 }
105 fCoeff.close(); //Close the coefficients file
106 }
107 void defineSiLiCoeff() //Get SiLi coefficients. Does not include run-by-run
108   → corrections
109 {
110     //First thing: read in the coefficients for this run.
111     fstream fCoeff("user/SiLiCoefficients.dat"); //Coefficient File
112     if(!fCoeff.is_open())
113     {
114         cout << "SiLi file did not open" << endl;
115         return;
116     }
117     string buffer;
118     std::getline(fCoeff,buffer);
119     nSiLiOrder = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
120         → //So this clusterfuck is due to gcc being a pain. it basically converts a
121         → substring of the string into a char to convert into an int because
122         → apparently it won't just go string to int
123     getline (fCoeff,buffer);
124     nSiLiDets = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
125     getline (fCoeff,buffer);
126     nSiLiPlace = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
127     getline (fCoeff,buffer); //Label Line
128     std::vector<double> row; //Row for adding a level in.
//Okay, here, we get into the nitty gritty
129     for(int i=0; i<nSiLiDets;i++) //Loop through all the detectors
130     {
131         getline (fCoeff,buffer); //Line with coefficients on it.

```

```

127     dSiLiCoefficients.push_back(row); //Put a new row in for the detector
128     for(int j=0; j<= nSiLiOrder; j++) //Loop through the coefficients
129     {
130         dSiLiCoefficients[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of(
131             ',' ,0)).c_str())); //read in jth
132             coefficient
133             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_first_of(
134                 '\n',0)); //make a substring of the rest of the
135                 coefficients
136     }
137     fCoeff.close(); //Close the coefficients file
138 }
139
140 void defineSiLiCoeff(int nRunNum) //SiLi Coefficients for the run based
141     → correction
142 {
143     //First thing: read in the coefficients for this run.
144     ifstream fCoeff(Form("user/Run_by_Run/SiLiCoefficients_r%i.dat",nRunNum));
145     → //Coefficient File
146     if(!fCoeff.is_open())
147     {
148         cout << "SiLi run file did not open, setting correction to y=x" << endl;
149         std::vector<double> row; //Row for adding a level in.
150         for(int i=0; i<nSiLiDets;i++)
151         {
152             dSiLiRunCorr.push_back(row);
153             dSiLiRunCorr[i].push_back(0);
154             dSiLiRunCorr[i].push_back(1);
155         }
156         return;
157     }
158     string buffer;
159     std::getline(fCoeff,buffer); //Label
160     std::vector<double> row; //Row for adding a level in.
161     //Okay, here, we get into the nitty gritty
162     for(int i=0; i<nSiLiDets;i++) //Loop through all the detectors
163     {
164         getline (fCoeff,buffer); //Line with coefficients on it.
165         dSiLiRunCorr.push_back(row); //Put a new row in for the detector
166         for(int j=0; j<= 1; j++) //Loop through the coefficients, assuming linear
167         {
168             dSiLiRunCorr[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of(
169                 ',' ,0)).c_str())); //read in jth
170                 coefficient
171             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_first_of(
172                 '\n',0)); //make a substring of the rest of the
173                 coefficients
174         }
175     }
176     fCoeff.close(); //Close the coefficients file
177 }

```

C.3.2 include/Coefficients.h

Header file for Coefficients.cxx.

```
1  #ifndef COEFFICIENTS_H
2  #define COEFFICIENTS_H
3
4  #ifndef _COEFFICIENTS_CXX
5  #define EXTERNAL extern
6  #else
7  #define EXTERNAL
8  #endif
9
10 void defineGeCoeff();
11
12 void defineGeCoeff(int nRunNum);
13
14 void defineSiLiCoeff();
15
16 void defineSiLiCoeff(int nRunNum);
17
18 #endif
```

C.3.3 src/Constraints.cxx

These are the subroutines for reading in coincidence gates for the BGO, Si(Li), and HPGe detectors.

```
1  #include "Constraints.h"
2
3  //C++ libraries
4  #include <iostream>
5  #include <fstream>
6  #include <string>
7  #include <cmath>
8  #include <vector>
9  #include <cstdlib>
10
11 //ROOT libraries
12 #include <TString.h>
13
14 using namespace std;
15
16 int nBGODets; //Total number of signals from BGO detectors
17 int nBGOPlace; //Start of BGOs in generalized array
18 std::vector<double> dBGOThreshold;
```

```

19
20 int defineConstraints(char* cCuts, std::vector<std::vector<double> > &dBounds)
21 {
22     //First thing: read in the cuts for this run.
23     fstream fCuts(Form("user/Cut_Files/%s",cCuts)); //Coefficient File
24     if(!fCuts.is_open())
25     {
26         cout << "Cuts file did not open" << endl;
27     }
28     string buffer;
29     std::getline(fCuts,buffer);
30     int nConstraints = std::atoi(buffer.substr(0,buffer.find_first_of(
31         ',',0)).c_str()); //So this clusterfuck is due to gcc being a pain. it
32         //basically converts a substring of the string into a char to convert into
33         //an int because apparently it won't just go string to int
34     getline (fCuts,buffer); //Label Line
35     std::vector<double> row; //Row for adding a level in.
36     //Okay, here, we get into the nitty gritty
37     for(int i=0; i<nConstraints;i++) //Loop through all the constraints
38     {
39         getline (fCuts,buffer); //Line with cuts on it.
40         dBounds.push_back(row); //Put a new row of constraints
41         for(int j=0; j< 3; j++) //Loop through the parts of the constraints
42         {
43             dBounds[i].push_back(std::atof(buffer.substr(0,buffer.find_first_of(',',j
44                 ,0)).c_str())); //read in jth
45                 //coefficient
46             buffer = buffer.substr(buffer.find_first_of(',',0)+1,buffer.find_first_o
47                 f('\n',0)); //make a substring of the rest of the
48                 //coefficients
49         }
50         //Make low and high bounds
51         dBounds[i][1]=dBounds[i][1]-dBounds[i][2];
52         dBounds[i][2]=dBounds[i][1]+2*dBounds[i][2];
53     }
54     fCuts.close();
55     return nConstraints;
56 }
57
58 void defineBGO()
59 {
60     //First thing: read in the cuts for this run.
61     fstream fBGO("user/BGO.dat"); //Coefficient File
62     if(!fBGO.is_open())
63     {
64         cout << "BGO file did not open" << endl;
65     }
66     string buffer;
67     std::getline(fBGO,buffer);
68     nBGODets = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
69     //So this clusterfuck is due to gcc being a pain. it basically converts a
70     //substring of the string into a char to convert into an int because
71     //apparently it won't just go string to int
72     std::getline(fBGO,buffer);

```

```

63     nBGOPlace = std::atoi(buffer.substr(0,buffer.find_first_of(' ',0)).c_str());
64     std::getline(fBGO,buffer); //Label Line
65     //Okay, here, we get into the nitty gritty
66     for(int i=0; i<nBGODets;i++) //Loop through all the constraints
67     {
68         std::getline(fBGO,buffer); //Line with cuts on it.
69         dBГОThreshold.push_back(std::atof(buffer.c_str())); //read in ith detector
69         ← threshold
70     }
71     fBGO.close();
72 }
```

C.3.4 include/Constraints.h

Header file for Constraints.cxx.

```

1  #ifndef CONSTRAINTS_H
2  #define CONSTRAINTS_H
3
4  #ifndef _CONSTRAINTS_CXX
5  #define EXTERNAL extern
6  #else
7  #define EXTERNAL
8  #endif
9
10 #include <vector>
11
12 int defineConstraints(char* cCuts, std::vector<std::vector<double> > &dBounds);
13 int defineTripleConstraints(char* cCuts, std::vector<std::vector<double> >
13     ← &dBounds);
14 void defineBGO();
15
16 #endif
```

C.3.5 src/Filelist.cxx

These are the subroutines for reading in the directory to the data, the generalized run name, the file type, and the tree name. The tree name is not used in the merge code.

```

1  #include "Filelist.h"
2
3  //C++ libraries
```

```

4  #include <vector>
5  #include <string>
6  #include <fstream>
7  #include <iostream>
8  #include <cstdlib>
9
10 //ROOT libraries
11 #include <TChain.h>
12 #include <TString.h>
13 #include <TSystem.h>
14
15
16 //sFilepath,sRun and sFType are all in the header file Filelist.h
17 std::string sFilepath;
18 //Change sTree for the program to look at a different tree in the files.
19 std::string sTree;
20 //Formatting for the beginning and end of the rootfile name
21 std::string sRun;
22 std::string sFType;
23
24 TChain* chain;
25
26 using namespace std;
27
28 void readPaths()
29 {
30     fstream fFile("user/Filelist.dat"); //Coefficient File
31     if(!fFile.is_open())
32     {
33         cout << "Filelist did not open" << endl;
34     }
35     string buffer;
36     std::getline(fFile,buffer); //Label
37     std::getline(fFile,sFilepath); //Filepath
38     sFilepath = sFilepath.substr(0,sFilepath.size()-1); //make a substring without
39     ↵ a return
40
41     std::getline(fFile,buffer); //Label
42     std::getline(fFile,sTree); //Tree
43     sTree = sTree.substr(0,sTree.size()-1); //make a substring without a return
44
45     std::getline(fFile,buffer); //Label
46     std::getline(fFile,sRun); //Run
47     std::getline(fFile,sFType); //Type
48     sRun = sRun.substr(0,sRun.size()-1); //make a substring without a return
49     sFType = sFType.substr(0,sFType.find_first_of('\n')-1); //make a substring
50     ↵ without a return
51
52     fFile.close();
53 }
54
55 void makeChain(int nRunNum)
56 {
57     //Set up the chain

```

```

56   chain = new TChain(sTree.c_str());
57   chain->SetCacheSize(1E8);
58   chain->AddBranchToCache("*");
59
60   //Add all the chains to the file. Change the list by adding or subtracting the
61   //→ rootfiles using the same syntax.
62   //For the May 2016 run, data taking begins at run 16
63   //Skipping runs 23-26 -> high beam current affected SiLi resolution
64   //NOTE: from run 63 onward, SiLi 6 was removed from the DAQ, as it became too
65   //→ noisy. This will have to be taken into account at some point
66   //Some kind of crash. Runs 64 and 65 do not exist
67   //Data runs finish at run 90.
68
69   //Don't go through the loop for the runs marked above
70
71   //NOTE: needs to be adjusted at some point to include single or triple digits
72   bRealRun = gSystem->IsFileInIncludePath(Form("%s%s%i%s", sFilepath.c_str(),
73   ← sRun.c_str(), nRunNum, sFType.c_str()));
74   cout << Form("%s%s%i%s", sFilepath.c_str(), sRun.c_str(), nRunNum,
75   ← sFType.c_str()) << "\t" << bRealRun << endl;
74   chain->Add(Form("%s%s%i%s", sFilepath.c_str(), sRun.c_str(), nRunNum,
75   ← sFType.c_str()));
75 }
```

C.3.6 include/Filelist.h

Header file for Filelist.hxx.

```

1  #ifndef FILELIST_H
2  #define FILELIST_H
3
4
5  //C++ libraries
6  #include <iostream>
7  #include <string>
8  #include <cstdio>
9
10 //ROOT libraries
11 #include <TROOT.h>
12 #include "TFile.h"
13 #include "TTree.h"
14 #include "TChain.h"
15 #include "TString.h"
16 #include " TBranch.h"
17 #include "TLeaf.h"
18 #include "TObject.h"
19 #include "TObjArray.h"
20
```

```

21  #ifndef _FILELIST_CXX
22  #define EXTERNAL extern
23  #else
24  #define EXTERNAL
25  #endif
26
27  void makeChain(int nRunNum);
28  void readPaths();
29
30 #endif

```

C.4 Example User Files

C.4.1 user/Filelist.dat

This is read in by the `Filelist.cxx` subroutines. It includes the path to the raw data files, the name of the tree, and the formatting for the beginning and end of the data files, including the file extension. "DataDirectory" should be replaced with the correct filepath. The comments must be left in for proper spacing to be read by the program.

```

1 //Change sfilepath to change the location the program looks for the rootfiles
2 DataDirectory
3 //Change sTree for the program to look at a different tree in the files.
4 evtTree
5 //Formatting for the beginning and end of the rootfile name
6 run
7 .root

```

C.4.2 user/GeCoefficients.dat

This is read in by the `Coefficients.cxx` subroutines. It includes the number of HPGe detectors, the polynomial order of the energy calibration, and the energy calibration itself. Coefficients are in order of increasing polynomial (0^{th} , 1^{st} , etc.). It also is set up for segmented detectors, as in the Clovershare runs. In the case of the different tree structure in the Clovershare data, it also contains the start index of the HPGe detectors in the `evt` tree array. As a special correction function was used

on the residuals to deal with the differential non-linearity of the electronics, as discussed in section 2.7.3 and 3.3.1, these coefficients are also included. They can be excluded if not needed. The comments must be left in for proper spacing to be read by the program.

```

1 1 //Order of calibration i.e. 1 = linear.
2 28 //Total number of signals from Germanium detectors
3 4 //number of segments in a single Germanium crystal, for adding purposes
4 Germanium coefficients, individual signals
5 -0.1341,0.302906
6 0.2797,0.307566
7 0.2768,0.318397
8 0.1213,0.309977
9 0.3823,0.369697
10 0.4855,0.310194
11 0.5756,0.255518
12 0.8542,0.255371
13 0.3598,0.312358
14 0.2325,0.316021
15 1.115,0.320741
16 0.5961,0.305995
17 0.3897,0.314955
18 0.1004,0.312015
19 0.147581,0.314969
20 -0.504784,0.303855
21 0.4491,0.328692
22 0.2944,0.331888
23 0.3586,0.327496
24 0.463,0.314045
25 0.2192,0.308175
26 0.1438,0.310634
27 -0.0039,0.303053
28 0.3204,0.304322
29 0.1457,0.333777
30 0.1074,0.333716
31 0.2118,0.319745
32 0.2501,0.347206
33 Residual Germanium Coefficients
34 0.0002045,-2.794,3291,-1.604,7886,-2.117,335.7
35 0.0001749,-2.524,3374,-1.436,7501,-1.959,62.84
36 0.0003222,-3.928,3019,-2.283,7516,-3.026,13.21
37 0.0001399,-1.626,2734,-0.9474,7857,-1.2852,186.9
38 0.0001379,-0.03692,3661,-0.1894,0,0,33.82
39 0.0003149,-3.532,2556,-2.088,7644,-2.743,275.2
40 0.0001943,-1.873,2827,-1.126,6000,-1.361,16.09
41 0.0001881,-1.867,3040,-1.086,6240,-1.312,51.4
42 0.0002371,-3.204,3451,-1.849,6680,-2.417,196.8
43 0.0001754,-1.94,2463,-1.135,7197,-1.544,2.549
44 0.0002692,-3.208,2985,-1.833,6392,-2.449,154.9

```

```

45 0.00004704,-0.6761,3491,-0.3824,7500,-0.5222,2.679
46 0.0002221,-2.528,2443,-1.466,7500,-1.991,59.55
47 0.0001218,-1.258,3453,-0.7873,0,0,178.9
48 0,0,0,0,0,0
49 0,0,0,0,0,0
50 0.000151,-1.967,3154,-1.111,7433,-1.536,2.431
51 0.0001272,-1.559,2988,-0.8972,7504,-1.229,120.3
52 0.0002146,-6.801,2300,-3.603,8466,-6.3,788.8
53 0.0001937,-2.351,3039,-1.352,7500,-1.79,108.8
54 0.00007026,-0.07328,2723,-0.4305,7500,-0.5336,25.13
55 0.00009658,-1.205,3422,-0.7118,7500,-0.8693,125.3
56 0.000141,-1.512,2498,-0.8911,8022,-1.185,76.33
57 0.0001528,-1.793,2765,-1.051,7984,-1.39,163.7
58 0.0001754,-1.754,2204,-1.014,6987,-1.374,3.612
59 0.0002673,-2.724,1851,-1.582,6055,-2.226,358.8
60 0.000141,-1.688,2250,-0.982,6898,-1.403,0.3633
61 0.0001584,-2.322,2466,-1.285,7473,-1.95,153.9

```

C.4.3 user/SiLiCoefficients.dat

This is read in by the `Coefficients.cxx` subroutines. It includes the number of Si(Li) detectors, the polynomial order of the energy calibration, and the energy calibration itself. Coefficients are in order of increasing polynomial (0^{th} , 1^{st} , etc.).

In the case of the different tree structure in the Clovershare data, it also contains the start index of the Si(Li) detectors in the `evt` tree array. The comments must be left in for proper spacing to be read by the program.

```

1 2 //Order of calibration i.e. 1 = linear.
2 6 //Total number of signals from SiLi detectors
3 48 //Start of SiLis in generalized array, redundant from CloverShare
4 //SiLi Coefficients
5 -20.0687, 0.186828, -3.03149E-06
6 9.46919, 0.153857, 6.276E-07
7 3.84721, 0.163458, 3.61702E-07
8 8.41864, 0.160942, 4.86406E-07
9 5.5835, 0.163278, 3.71974E-07
10 -2.28321, 0.173382, 2.06782E-07

```

C.4.4 user/BGO.dat

This is read in by the `Coefficients.cxx` subroutines. It includes the number of BGO detectors and the thresholds for each detector. In the case of the different tree

structure in the Clovershare data, it also contains the start index of the BGO detectors in the `evt` tree array. The comments must be left in for proper spacing to be read by the program.

```

1 4 //Total number of signals from BGO detectors
2 36 //Start of BGOs in generalized array, redundant from clovershare
3 //BGO Thresholds
4 10
5 10
6 10
7 10

```

C.4.5 user/Timing.dat

This is read in by the `Timing.cxx` subroutines. It includes the timing cut for each pairing of detectors. Each pairing has index of each detector and the upper and lower limit of the timing gate. Indexes for detectors start at 0. If the number of detectors is incorrect in either the `GeCoefficients.cxx` or the `SiLiCoefficients.cxx`, this file will not be read in correctly. The order of pairing groups (ge-ge, ge-sili, etc.) must be kept the same. The comments must be left in for proper spacing to be read by the program. This file name can be changed so multiple timing files can exist, as the timing file is specified as input when running the program.

```

1 //ge-ge
2 0, 1, -1500, 1000
3 //ge-sili
4 0, 0, -1500, 0
5 0, 1, -600, 1000
6 0, 2, -400, 1000
7 0, 3, -500, 1000
8 0, 4, -200, 1100
9 0, 5, -700, 1000
10 1, 0, -1200, 1200
11 1, 1, -600, 1500
12 1, 2, -200, 1400
13 1, 3, -500, 1400
14 1, 4, -100, 1600

```

```

15 1, 5, -600, 1500
16 //sili-sili
17 1, 2, -200, 200
18 1, 3, -200, 200
19 1, 4, 0, 200
20 1, 5, -600, 600
21 1, 0, -1200, 200
22 2, 3, -100, 100
23 2, 4, 0, 200
24 2, 5, -600, 200
25 2, 0, -1200, 200
26 3, 4, 0, 200
27 3, 5, -600, 400
28 3, 0, -1200, 200
29 4, 5, -700, 100
30 4, 0, -1200, 100
31 5, 0, -1200, 200

```

C.4.6 user/Cut_Files/GeCuts.dat

This is read in by the `Constraints.cxx` subroutines. It includes the number cuts to run on the HPGe detectors. Each cut is then listed as the detector index, the center of the gate, and half the width of the gate. This is done to mirror the centroid and width of peaks, for readability. The comments must be left in for proper spacing to be read by the program. This file name can be changed so multiple timing files can exist, as the timing file is specified as input when running the program.

```

1 14 //Total number of Ge cuts to make
2 //Ge number(0-6), centroid, sigma
3 0,123,1.5
4 1,123,1.5
5 2,123,1.5
6 3,123,1.5
7 4,123,1.5
8 5,123,1.5
9 6,123,1.5
10 0,247.9,1.5
11 1,247.9,1.5
12 2,247.9,1.5
13 3,247.9,1.5
14 4,247.9,1.5
15 5,247.9,1.5
16 6,247.9,1.5

```

C.4.7 user/Cut_Files/SiLiCuts.dat

This is read in by the `Constraints.cxx` subroutines. It includes the number cuts to run on the HPGe detectors. Each cut is then listed as the detector index, the center of the gate, and half the width of the gate. This is done to mirror the centroid and width of peaks, for readability. The comments must be left in for proper spacing to be read by the program. This file name can be changed so multiple timing files can exist, as the timing file is specified as input when running the program.

```
1 12 //Total number of SiLi cuts to make
2 SiLi number(0-6),centroid, sigma
3 0,630.5,4
4 1,630.5,4
5 2,630.5,4
6 3,630.5,4
7 4,630.5,4
8 5,630.5,4
9 0,672.3,4
10 1,672.3,4
11 2,672.3,4
12 3,672.3,4
13 4,672.3,4
14 5,672.3,4
```

C.4.8 user/Run_by_Run/GeCoefficients_r*.dat

This is read in by the `Coefficients.cxx` subroutines. It includes a linear correction for the HPGe detectors based on the run specified where * is in the file name. The run correction is assumed to be linear, and the number of detectors is assumed from the base calibration files. The comments must be left in for proper spacing to be read by the program. This file name can be changed so multiple run files can exist, as the run file is based on the run being analyzed. These files can be excluded and the correction will be set to $y=x$ automatically.

```
1 Germanium run correction coefficients, individual signals
2 0.584057642 , 0.999282044
```

```

3 0.971306146 , 0.999064637
4 2.143313887 , 0.99916084
5 0.952878608 , 0.998954085
6 0.14849556 , 0.999183925
7 0.026878147 , 0.999284696
8 -0.072866577 , 0.999213095
9 -0.269457227 , 0.999457012
10 0.980689919 , 0.999131285
11 2.111134837 , 0.998652167
12 0.684397991 , 0.998943934
13 1.251808042 , 0.998663883
14 -0.030330086 , 0.999770215
15 2.862946017 , 0.997447239
16 0 , 0
17 0 , 0
18 1.154940607 , 0.998653804
19 1.088098405 , 0.998668786
20 -0.394580103 , 1.000881362
21 0.443448664 , 0.99895749
22 -0.55139315 , 0.999311609
23 0.860640139 , 0.999007238
24 1.052573998 , 0.998685627
25 0.877030797 , 0.99920026
26 1.112452034 , 0.998738007
27 1.102602529 , 0.9988922
28 1.022320775 , 0.99899084
29 0.736591679 , 0.998994596

```

C.4.9 user/Run_by_Run/SiLiCoefficients_r*.dat

This is read in by the `Coefficients.cxx` subroutines. It includes a linear correction for the Si(Li) detectors based on the run specified where * is in the file name. The run correction is assumed to be linear, and the number of detectors is assumed from the base calibration files. The comments must be left in for proper spacing to be read by the program. This file name can be changed so multiple run files can exist, as the run file is based on the run being analyzed. These files can be excluded and the correction will be set to $y=x$ automatically.

APPENDIX D

ROOT MACRO CODES

This appendix contains macros used in ROOT for various pieces of analysis, such as subtracting off peaks or piecewise background fits. All codes were run using ROOT version 5.34.19 [12].

D.1 FitterProgram.cxx

This code is a series of fitting macros for various kinds of spectra and peak types.

These fits were compared with RADWARE[50] for consistency.

```
1 #include <TFile.h>
2 #include <TH1F.h>
3 #include <TF1.h>
4 #include <TMath.h>
5 #include <TRandom3.h>
6 #include <TFitResultPtr.h>
7
8 #include <sstream>
9 #include <string>
10 #include <iostream>
11
12 //linear+gaussian+skew+step guess, one iteration
13 void FitterSk1(TH1D* h1, double cent, double width, double rrr, double beta,
14                 double step, int xmin, int xmax)
15 {
16     double pi = 3.14159265359;
17     // TH1F* h1 = (TH1F*)inFile->Get(hist);
18     TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]*((1-[5])/100)*exp(-((x-[3])/(sqrt(
19         t(2.0)*[4]))**2)+TMath::Erfc((x-[3])/sqrt(2.0)*[4])+[4]/(sqrt(2.0)*[6]))*[5]/100*exp((x-[3])/[6])+TMath::Erfc((x-[3])/sqrt(2.0)*[4])*[7]/100)");
20     lingaus->SetParameters(0,0,0,cent,width,rrr,beta,step);
21     lingaus->FixParameter(3,cent);
22     lingaus->FixParameter(4,width);
23     lingaus->FixParameter(5,rrr);
```

```

22 lingaus->FixParameter(6,beta);
23 lingaus->FixParameter(7,step);
24 // lingaus->SetParLimits(5,0,100.0); //R
25 // lingaus->SetParLimits(7,0,100.0); //Step
26 h1->Fit("lingaus","BLS","",xmin,xmax);
27 lingaus->SetParLimits(3,cent-10,cent+10);
28 lingaus->SetParLimits(4,0,width+10);
29
30 /* for(int i=1; i<=500; i++)
31 {
32     h1->Fit("lingaus","BLQM","",xmin,xmax);
33 }
34 h1->Fit("lingaus","BLS","",xmin,xmax);
35 */
36 }
37
38 //linear+gaussian+skew+step guess
39 void FitterSk(TH1D* h1, double cent, double width, double rrr, double beta,
40 ← double step, int xMin, int xMax)
41 {
42     double pi = 3.14159265359;
43     // TH1F* h1 = (TH1F*)inFile->Get(hist);
44     TF1* lin = new
45     ← TF1("lin",[0]+[1]*x+[2]*TMath::Erfc((x-[3])/(sqrt(2.0)*[4]))*[5]/100");
46     TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]*((1-[5]/100)*exp(-((x-[3])/
47     ← t(2.0)*[4])*2)+TMath::Erfc((x-[3])/(sqrt(2.0)*[4])+[4]/(sqrt(2.0)*[6]))*[ ]
48     ← [5]/100*exp((x-[3])/[6])+TMath::Erfc((x-[3])/(sqrt(2.0)*[4]))*[7]/100)");
49     lingaus->SetParameters(0,0,0,cent,width,rrr,beta,step);
50     lingaus->FixParameter(3,cent);
51     lingaus->FixParameter(4,width);
52     lingaus->FixParameter(5,rrr);
53     lingaus->FixParameter(6,beta);
54     lingaus->FixParameter(7,step);
55     h1->Fit("lingaus","BLS","",xMin,xMax);
56     lingaus->SetParLimits(5,0,100.0); //R
57     lingaus->SetParLimits(6,0,1000.0); //beta
58     lingaus->SetParLimits(7,0,100.0); //Step
59     lingaus->SetParLimits(3,cent-10,cent+10);
60     lingaus->SetParLimits(4,0,width+10);
61
62     for(int i=1; i<=100; i++)
63     {
64         h1->Fit("lingaus","BLQM","",xMin,xMax);
65     }
66     h1->Fit("lingaus","BLS","",xMin,xMax);
67
68     TFitResultPtr r = h1->Fit("lingaus","LLS","",xMin,xMax);
69     lin->SetParameter(0, lingaus->GetParameter(0));
70     lin->SetParameter(1, lingaus->GetParameter(1));
71     lin->SetParameter(2, lingaus->GetParameter(2));
72     lin->SetParameter(3, lingaus->GetParameter(3));
73     lin->SetParameter(4, lingaus->GetParameter(4));
74     lin->SetParameter(5, lingaus->GetParameter(7));

```

```

72 lin->SetParError(0, lingaus->GetParError(0));
73 lin->SetParError(1, lingaus->GetParError(1));
74 lin->SetParError(2, lingaus->GetParError(2));
75 lin->SetParError(3, lingaus->GetParError(3));
76 lin->SetParError(4, lingaus->GetParError(4));
77 lin->SetParError(5, lingaus->GetParError(7));
78 Double_t c0_err = lingaus->GetParError(0);
79 Double_t c1_err = lingaus->GetParError(1);
80 Double_t bkgd_integral_error = TMath::Sqrt(
81   ↳ TMath::Power((xMax-xMin)*c0_err,2.0) +
82   ↳ TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
83 Double_t bkgd_integral = lin->Integral(xMin,xMax);
84 Double_t fn_integral = lingaus->Integral(xMin,xMax);
85 Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
86   ↳ r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
87 Double_t net_area = fn_integral - bkgd_integral;
88 Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
89   ↳ TMath::Power(bkgd_integral_error,2.0) );
90 r->GetCovarianceMatrix().Print();
91 printf("\n\nFit area = %10.6f+/-%10.6f\nBkgd area = %10.6f+/-%10.6f\nNet area =
92   ↳ %10.6f+/-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ ]
93   ↳ ral_error,net_area,net_area_error);
94 }
95
96 //linear+gaussian guess
97 void FitterG(TFile* inFile, char* hist, double cent, double width, int xmin, int
98   ↳ xmax)
99 {
100   double pi = 3.14159265359;
101   TH1F* h1 = (TH1F*)inFile->Get(hist);
102   TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] ]"
103   ↳ )/(sqrt(2.0)*[4]))**2)");
104   lingaus->SetParNames("intercept","slope","area","centroid", "sigma");
105   lingaus->SetParameters(0,0,0,cent,width);
106   lingaus->FixParameter(3,cent);
107   lingaus->FixParameter(4,width);
108   h1->Fit("lingaus","BLS","",xmin,xmax);
109   lingaus->SetParLimits(3,cent-5,cent+5);
110   lingaus->SetParLimits(4,width-5,width+5);
111
112   for(int i=1; i<=500; i++)
113   {
114     h1->Fit("lingaus","BLQM","",xmin,xmax);
115   }
116   h1->Fit("lingaus","BLS","",xmin,xmax);
117 }
118
119 //linear+gaussian*2 guess
120 void FitterG(TFile* inFile, char* hist, double cent, double width, double cent2,
121   ↳ double width2, int xmin, int xmax)
122 {
123   double pi = 3.14159265359;
124   TH1F* h1 = (TH1F*)inFile->Get(hist);

```

```

117 TF1* lingaus = new
118   ↪ TF1("lingaus", "[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3])/(sqrt(2.0)*[4] )
119   ↪ ])**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/(sqrt(2.0)*[7]))**2)");
120 lingaus->SetParNames("intercept","slope","area 1","centroid 1", "sigma 1","area
121   ↪ 2","centroid 2", "sigma 2");
122 lingaus->SetParameters(0,0,0,cent,width);
123 lingaus->FixParameter(3,cent);
124 lingaus->FixParameter(4,width);
125 lingaus->FixParameter(6,cent2);
126 lingaus->FixParameter(7,width2);
127 h1->Fit("lingaus","BLS","",xmin,xmax);
128 lingaus->SetParLimits(3,cent-5,cent+5);
129 lingaus->SetParLimits(4,width-5,width+5);
130 lingaus->SetParLimits(6,cent2-5,cent2+5);
131 lingaus->SetParLimits(7,width2-5,width2+5);
132
133 for(int i=1; i<=500; i++)
134 {
135   h1->Fit("lingaus","BLQM","",xmin,xmax);
136 }
137 h1->Fit("lingaus","BLS","",xmin,xmax);
138
139 //linear+gaussian*3 guess
140 void FitterG(TFile* inFile, char* hist, double cent, double width, double cent2,
141   ↪ double width2, double cent3, double width3, int xmin, int xmax)
142 {
143   double pi = 3.14159265359;
144   TH1F* h1 = (TH1F*)inFile->Get(hist);
145   TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] )
146   ↪ )/(sqrt(2.0)*[4]))**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/(sqrt(2.0)*[7] ) )
147   ↪ **2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/(sqrt(2.0)*[10]))**2)");
148   lingaus->SetParNames("intercept","slope","area 1","centroid 1", "sigma 1","area
149   ↪ 2","centroid 2", "sigma 2","area 3","centroid 3", "sigma 3");
150   lingaus->SetParameters(0,0,0,cent,width);
151   lingaus->FixParameter(3,cent);
152   lingaus->FixParameter(4,width);
153   lingaus->FixParameter(6,cent2);
154   lingaus->FixParameter(7,width2);
155   lingaus->FixParameter(9,cent3);
156   lingaus->FixParameter(10,width3);
157   h1->Fit("lingaus","BLS","",xmin,xmax);
158   lingaus->SetParLimits(3,cent-5,cent+5);
159   lingaus->SetParLimits(4,width-5,width+5);
160   lingaus->SetParLimits(6,cent2-5,cent2+5);
161   lingaus->SetParLimits(7,width2-5,width2+5);
162   lingaus->SetParLimits(9,cent3-5,cent3+5);
163   lingaus->SetParLimits(10,width3-5,width3+5);
164
165   for(int i=1; i<=500; i++)
166   {
167     h1->Fit("lingaus","BLQM","",xmin,xmax);
168   }
169   h1->Fit("lingaus","BLS","",xmin,xmax);

```

```

164    }
165
166 //guess for gamma singles, one old, one new peak
167 void FitterGS(TFile* inFile, char* hist, double cent, double width, double cent2,
168   → double width2, int xmin, int xmax)
168 {
169     double pi = 3.14159265359;
170     TH1F* h1 = (TH1F*)inFile->Get(hist);
171     TF1* lingaus = new
172       → TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3])/(sqrt(2.0)*[4]
173       → ]))**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/(sqrt(2.0)*[7]))**2)");
172     lingaus->SetParameters(0,0,0,cent,width);
173     lingaus->FixParameter(3,cent);
174     lingaus->FixParameter(4,width);
175     lingaus->FixParameter(6,cent2);
176     lingaus->FixParameter(7,width2);
177     h1->Fit("lingaus","BLS","","",xmin,xmax);
178     lingaus->SetParLimits(6,cent2-5,cent2+5);
179     lingaus->SetParLimits(7,width2-5,width2+5);
180
181     for(int i=1; i<=500; i++)
182     {
183         h1->Fit("lingaus","BLQM","","",xmin,xmax);
184     }
185     h1->Fit("lingaus","BLS","","",xmin,xmax);
186 }
187
188 //guess for gamma singles, one old, two new peaks
189 void FitterGS(TFile* inFile, char* hist, double cent, double width, double cent2,
190   → double width2, double cent3, double width3, int xmin, int xmax)
190 {
191     double pi = 3.14159265359;
192     TH1F* h1 = (TH1F*)inFile->Get(hist);
193     TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3])
194       → )/(sqrt(2.0)*[4]))**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/(sqrt(2.0)*[7]))
194       → )**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/(sqrt(2.0)*[10]))**2)");
194     lingaus->SetParameters(0,0,0,cent,width);
195     lingaus->FixParameter(3,cent);
196     lingaus->FixParameter(4,width);
197     lingaus->FixParameter(6,cent2);
198     lingaus->FixParameter(7,width2);
199     lingaus->FixParameter(9,cent3);
200     lingaus->FixParameter(10,width3);
201     h1->Fit("lingaus","BLS","","",xmin,xmax);
202     lingaus->SetParLimits(6,cent2-5,cent2+5);
203     lingaus->SetParLimits(7,width2-5,width2+5);
204     lingaus->SetParLimits(9,cent3-5,cent3+5);
205     lingaus->SetParLimits(10,width3-5,width3+5);
206
207     for(int i=1; i<=500; i++)
208     {
209         h1->Fit("lingaus","BLQM","","",xmin,xmax);
210     }
211     h1->Fit("lingaus","BLS","","",xmin,xmax);

```

```

212 }
213
214 //guess for gamma singles, two old, one new peaks
215 void FitterGS2(TFile* inFile, char* hist, double cent, double width, double
216    ↪ cent2, double width2, double cent3, double width3, int xmin, int xmax)
217 {
218     double pi = 3.14159265359;
219     TH1F* h1 = (TH1F*)inFile->Get(hist);
220     TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] ]
221        ↪ )/(sqrt(2.0)*[4]))**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/sqrt(2.0)*[7]) )
222        ↪ )**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[10]))**2)");
223     lingaus->SetParameters(0,0,0,cent,width);
224     lingaus->FixParameter(3,cent);
225     lingaus->FixParameter(4,width);
226     lingaus->FixParameter(6,cent2);
227     lingaus->FixParameter(7,width2);
228     lingaus->FixParameter(9,cent3);
229     lingaus->FixParameter(10,width3);
230     h1->Fit("lingaus","BLS","",xmin,xmax);
231     lingaus->SetParLimits(9,cent3-5,cent3+5);
232     lingaus->SetParLimits(10,width3-5,width3+5);
233
234     for(int i=1; i<=500; i++)
235     {
236         h1->Fit("lingaus","BLQM","",xmin,xmax);
237     }
238     h1->Fit("lingaus","BLS","",xmin,xmax);
239
240 //linear+sine+gaussian, sine guess
241 void FitterGTS(TFile* inFile, char* hist, double cent, double width, double
242    ↪ scale, double freq, double phase, int xmin, int xmax)
243 {
244     double pi = 3.14159265359;
245     TH1F* h1 = (TH1F*)inFile->Get(hist);
246     TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6] ]
247        ↪ )/(sqrt(2.0)*[7]))**2)*(1+[2]*sin([3]*x+[4]))");
248     lingaus->SetParameters(0,0,0,cent,width);
249     lingaus->FixParameter(6,cent);
250     lingaus->FixParameter(7,width);
251     lingaus->FixParameter(2,scale);
252     lingaus->FixParameter(3,freq);
253     lingaus->FixParameter(4,phase);
254     h1->Fit("lingaus","BLS","",xmin,xmax);
255     lingaus->SetParLimits(2,scale-scale,2);
256     lingaus->SetParLimits(3,0.25,0.45);
257     lingaus->SetParLimits(4,0,2*pi);
258
259     for(int i=1; i<=500; i++)
260     {
261         h1->Fit("lingaus","BLQM","",xmin,xmax);
262     }
263     h1->Fit("lingaus","BLS","",xmin,xmax);
264 }
```

```

261
262 //linear+sine+gaussian*2, sine guess
263 void FitterGTS(TFile* inFile, char* hist, double cent, double width, double
264   → cent2, double width2, double scale, double freq, double phase, int xmin, int
265   → xmax)
266 {
267   double pi = 3.14159265359;
268   TH1F* h1 = (TH1F*)inFile->Get(hist);
269   TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
270   → ])/(sqrt(2.0)*[7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/(sqrt(2.0)*[1]
271   → 0]))**2)*(1+[2]*sin([3]*x+[4]))");
272   lingaus->SetParameters(0,0,0,cent,width);
273   lingaus->FixParameter(6,cent);
274   lingaus->FixParameter(7,width);
275   lingaus->FixParameter(9,cent2);
276   lingaus->FixParameter(10,width2);
277   lingaus->FixParameter(2,scale);
278   lingaus->FixParameter(3,freq);
279   lingaus->FixParameter(4,phase);
280   h1->Fit("lingaus","BLS","",xmin,xmax);
281   lingaus->SetParLimits(2,scale-scale,2);
282   lingaus->SetParLimits(3,0.25,0.45);
283   lingaus->SetParLimits(4,0,2*pi);
284
285   for(int i=1; i<=500; i++)
286   {
287     h1->Fit("lingaus","BLQM","",xmin,xmax);
288   }
289   h1->Fit("lingaus","BLS","",xmin,xmax);
290 }

291 //linear+sine+gaussian*3, sine guess
292 void FitterGTS(TFile* inFile, char* hist, double cent, double width, double
293   → cent2, double width2, double cent3, double width3, double scale, double freq,
294   → double phase, int xmin, int xmax)
295 {
296   double pi = 3.14159265359;
297   TH1F* h1 = (TH1F*)inFile->Get(hist);
298   TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
299   → ])/(sqrt(2.0)*[7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/(sqrt(2.0)*[1]
300   → 0]))**2)+[11]/(sqrt(2.0*pi)*[13])*exp(-((x-[12])/(sqrt(2.0)*[13]))**2)*(1+
301   → [2]*sin([3]*x+[4]))");
302   lingaus->SetParameters(0,0,0,cent,width);
303   lingaus->FixParameter(6,cent);
304   lingaus->FixParameter(7,width);
305   lingaus->FixParameter(9,cent2);
306   lingaus->FixParameter(10,width2);
307   lingaus->FixParameter(12,cent3);
308   lingaus->FixParameter(13,width3);
309   lingaus->FixParameter(2,scale);
310   lingaus->FixParameter(3,freq);
311   lingaus->FixParameter(4,phase);
312   h1->Fit("lingaus","BLS","",xmin,xmax);
313   lingaus->SetParLimits(2,scale-scale,2);

```

```

306 lingaus->SetParLimits(3,0.25,0.45);
307 lingaus->SetParLimits(4,0,2*pi);
308
309 for(int i=1; i<=500; i++)
310 {
311     h1->Fit("lingaus", "BLQM", "", xmin,xmax);
312 }
313 h1->Fit("lingaus", "BLS", "", xmin,xmax);
314 }
315
316 //linear+sine+gaussian, gaussian guess
317 void FitterGTG(TFile* inFile, char* hist, double cent, double width, double
318     → scale, double freq, double phase, int xmin, int xmax)
319 {
320     double pi = 3.14159265359;
321     TH1F* h1 = (TH1F*)inFile->Get(hist);
322     TF1* lingaus = new TF1("lingaus", "[0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
323         → ])/(sqrt(2.0)*[7]))**2)*(1+[2]*sin([3]*x+[4]))");
324     lingaus->SetParameters(0,0,0,cent,width);
325     lingaus->FixParameter(6,cent);
326     lingaus->FixParameter(7,width);
327     lingaus->FixParameter(2,scale);
328     lingaus->FixParameter(3,freq);
329     lingaus->FixParameter(4,phase);
330     h1->Fit("lingaus", "BLS", "", xmin,xmax);
331     lingaus->SetParLimits(6,cent-5,cent+5);
332     lingaus->SetParLimits(7,width-5,width+5);
333
334     for(int i=1; i<=500; i++)
335     {
336         h1->Fit("lingaus", "BLQM", "", xmin,xmax);
337     }
338
339 //linear+sine+gaussian*2, gaussian*2 guess
340 void FitterGTG(TFile* inFile, char* hist, double cent, double width, double
341     → cent2, double width2, double scale, double freq, double phase, int xmin, int
342     → xmax)
343 {
344     double pi = 3.14159265359;
345     TH1F* h1 = (TH1F*)inFile->Get(hist);
346     TF1* lingaus = new TF1("lingaus", "[0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
347         → ])/(sqrt(2.0)*[7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[1]
348         → 0]))**2)*(1+[2]*sin([3]*x+[4]))");
349     lingaus->SetParameters(0,0,0,cent,width);
350     lingaus->FixParameter(6,cent);
351     lingaus->FixParameter(7,width);
352     lingaus->FixParameter(9,cent2);
353     lingaus->FixParameter(10,width2);
354     lingaus->FixParameter(2,scale);
355     lingaus->FixParameter(3,freq);
356     lingaus->FixParameter(4,phase);
357     h1->Fit("lingaus", "BLS", "", xmin,xmax);

```

```

354     lingaus->SetParLimits(6,cent-5,cent+5);
355     lingaus->SetParLimits(7,width-5,width+5);
356     lingaus->SetParLimits(9,cent2-5,cent2+5);
357     lingaus->SetParLimits(10,width2-5,width2+5);
358
359     for(int i=1; i<=500; i++)
360     {
361         h1->Fit("lingaus","BLQM","",xmin,xmax);
362     }
363     h1->Fit("lingaus","BLS","",xmin,xmax);
364 }
365
366 //linear+sine+gaussian*3, gaussian*3 guess
367 void FitterGTG(TFile* inFile, char* hist, double cent, double width, double
368     ↪ cent2, double width2, double cent3, double width3, double scale, double
369     ↪ freq, double phase, int xmin, int xmax)
370 {
371     double pi = 3.14159265359;
372     TH1F* h1 = (TH1F*)inFile->Get(hist);
373     TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
374     ↪ ])/(sqrt(2.0)*[7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[1]
375     ↪ 0]))**2)+[11]/(sqrt(2.0*pi)*[13])*exp(-((x-[12])/sqrt(2.0)*[13]))**2)*(1+
376     ↪ [2]*sin([3]*x+[4]))");
377     lingaus->SetParameters(0,0,0,cent,width);
378     lingaus->FixParameter(6,cent);
379     lingaus->FixParameter(7,width);
380     lingaus->FixParameter(9,cent2);
381     lingaus->FixParameter(10,width2);
382     lingaus->FixParameter(12,cent3);
383     lingaus->FixParameter(13,width3);
384     lingaus->FixParameter(2,scale);
385     lingaus->FixParameter(3,freq);
386     lingaus->FixParameter(4,phase);
387     h1->Fit("lingaus","BLS","",xmin,xmax);
388     lingaus->SetParLimits(6,cent-5,cent+5);
389     lingaus->SetParLimits(7,width-5,width+5);
390     lingaus->SetParLimits(9,cent2-5,cent2+5);
391     lingaus->SetParLimits(10,width2-5,width2+5);
392     lingaus->SetParLimits(12,cent3-5,cent3+5);
393     lingaus->SetParLimits(13,width3-5,width3+5);
394
395     for(int i=1; i<=500; i++)
396     {
397         h1->Fit("lingaus","BLQM","",xmin,xmax);
398     }
399     h1->Fit("lingaus","BLS","",xmin,xmax);
400 }
401
402 //linear+gaussian
403 void FitterI(TFile* inFile, char* hist, double cent, double width, int xMin, int
404     ↪ xMax)
405 {
406     double pi = 3.14159265359;
407     TH1F* h1 = (TH1F*)inFile->Get(hist);

```

```

402 // TF1* lingaus = new TF1("lingaus","pol1(0)+gaus(2)");
403 TF1* lin = new TF1("lin","[0]+[1]*x");
404 TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] ]
405 → )/(sqrt(2.0)*[4]))**2)");
406 lingaus->SetParameters(-20,0.022,800,0,0);
407 lingaus->FixParameter(3,cent);
408 lingaus->FixParameter(4,width);
409 // h1->Fit("lingaus","BLQM","","xMin,xMax");
410 // lingaus->SetParLimits(3,cent-5,cent+5);
411 // lingaus->SetParLimits(4,width-5,width+5);
412 // h1->Fit("lingaus","BLM","","xMin,xMax");

413 for(int i=1; i<=500; i++)
414 {
415     h1->Fit("lingaus","LLQM","","xMin,xMax");
416 }
417 TFitResultPtr r = h1->Fit("lingaus","LLS","","xMin,xMax");
418 lin->SetParameter(0, lingaus->GetParameter(0));
419 lin->SetParameter(1, lingaus->GetParameter(1));
420 Double_t bkgd_integral = lin->Integral(xMin,xMax);
421 Double_t c0_err = lingaus->GetParError(0);
422 Double_t c1_err = lingaus->GetParError(1);
423 Double_t bkgd_integral_error = TMath::Sqrt(
424 → TMath::Power((xMax-xMin)*c0_err,2.0) +
425 → TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
426
427 Double_t fn_integral = lingaus->Integral(xMin,xMax);
428 Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
429 → r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
430 Double_t net_area = fn_integral - bkgd_integral;
431 Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
432 → TMath::Power(bkgd_integral_error,2.0) );
433 r->GetCovarianceMatrix().Print();
434 printf("\n\nFit area = %10.6f+-%10.6f\nBkgd area = %10.6f+-%10.6f\nNet area =
435 → %10.6f+-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ ]
436 → ral_error,net_area,net_area_error);
437 }

438 //linear+gaussian*2
439 void FitterI(TFile* inFile, char* hist, double cent, double width, double cent2,
440 → double width2, int xMin, int xMax)
441 {
442     double pi = 3.14159265359;
443     TH1F* h1 = (TH1F*)inFile->Get(hist);
444     // TF1* lingaus = new TF1("lingaus","pol1(0)+gaus(2)");
445     TF1* lin = new TF1("lin","[0]+[1]*x");
446     TF1* lingaus = new
447     → TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3])/sqrt(2.0)*[4]
448     → )**2)+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/sqrt(2.0)*[7]))**2)");
449     lingaus->SetParameters(-20,0.022,800,0,0);
450     lingaus->FixParameter(3,cent);
451     lingaus->FixParameter(4,width);
452     lingaus->FixParameter(6,cent2);

```

```

446     lingaus->FixParameter(7,width2);
447     // h1->Fit("lingaus","BLQM","","xMin,xMax");
448     // lingaus->SetParLimits(3,cent-5,cent+5);
449     // lingaus->SetParLimits(4,width-5,width+5);
450     // h1->Fit("lingaus","BLM","","xMin,xMax");
451
452     for(int i=1; i<=500; i++)
453     {
454         h1->Fit("lingaus","LLQM","","xMin,xMax");
455     }
456     TFitResultPtr r = h1->Fit("lingaus","LLS","","xMin,xMax");
457     lin->SetParameter(0, lingaus->GetParameter(0));
458     lin->SetParameter(1, lingaus->GetParameter(1));
459     Double_t bkgd_integral = lin->Integral(xMin,xMax);
460     Double_t c0_err = lingaus->GetParError(0);
461     Double_t c1_err = lingaus->GetParError(1);
462     Double_t bkgd_integral_error = TMath::Sqrt(
463         → TMath::Power((xMax-xMin)*c0_err,2.0) +
464         → TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
465
466     Double_t fn_integral = lingaus->Integral(xMin,xMax);
467     Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
468         → r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
469     Double_t net_area = fn_integral - bkgd_integral;
470     Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
471         → TMath::Power(bkgd_integral_error,2.0) );
472     r->GetCovarianceMatrix().Print();
473     printf("\n\nFit area = %10.6f+-%10.6f\nBkgd area = %10.6f+-%10.6f\nNet area =
474         → %10.6f+-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ ]
475         → ral_error,net_area,net_area_error);
476
477 }
478
479 //linear+gaussian*3
480 void FitterI(TFile* inFile, char* hist, double cent, double width, double cent2,
481     → double width2, double cent3, double width3, int xMin, int xMax)
482 {
483     double pi = 3.14159265359;
484     TH1F* h1 = (TH1F*)inFile->Get(hist);
485     // TF1* lingaus = new TF1("lingaus","pol1(0)+gaus(2)");
486     TF1* lin = new TF1("lin","[0]+[1]*x");
487     TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] ]
488         → )/(sqrt(2.0)*[4]))**2+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/sqrt(2.0)*[7]) ]
489         → )**2+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[10]))**2)");
490     lingaus->SetParameters(-20,0.022,800,0,0);
491     lingaus->FixParameter(3,cent);
492     lingaus->FixParameter(4,width);
493     lingaus->FixParameter(6,cent2);
494     lingaus->FixParameter(7,width2);
495     lingaus->FixParameter(9,cent3);
496     lingaus->FixParameter(10,width3);
497     // h1->Fit("lingaus","BLQM","","xMin,xMax");
498     // lingaus->SetParLimits(3,cent-5,cent+5);
499     // lingaus->SetParLimits(4,width-5,width+5);

```

```

491 // h1->Fit("lingaus", "BLM", "", xMin, xMax);
492
493 for(int i=1; i<=500; i++)
494 {
495     h1->Fit("lingaus", "LLQM", "", xMin, xMax);
496 }
497 TFitResultPtr r = h1->Fit("lingaus", "LLS", "", xMin, xMax);
498 lin->SetParameter(0, lingaus->GetParameter(0));
499 lin->SetParameter(1, lingaus->GetParameter(1));
500 Double_t bkgd_integral = lin->Integral(xMin, xMax);
501 Double_t c0_err = lingaus->GetParError(0);
502 Double_t c1_err = lingaus->GetParError(1);
503 Double_t bkgd_integral_error = TMath::Sqrt(
    ↳ TMath::Power((xMax-xMin)*c0_err,2.0) +
    ↳ TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
504
505 Double_t fn_integral = lingaus->Integral(xMin, xMax);
506 Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
    ↳ r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
507 Double_t net_area = fn_integral - bkgd_integral;
508 Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
    ↳ TMath::Power(bkgd_integral_error,2.0) );
509 r->GetCovarianceMatrix().Print();
510 printf("\n\nFit area = %10.6f+/-%10.6f\nBkgd area = %10.6f+/-%10.6f\nNet area =
    ↳ %10.6f+/-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ
    ↳ ral_error,net_area,net_area_error);
511
512 }
513
514 //linear+gaussian*
515 void FitterI(TFile* inFile, char* hist, double cent, double width, double cent2,
    ↳ double width2, double cent3, double width3, double cent4, double width4, int
    ↳ xMin, int xMax)
516 {
517     double pi = 3.14159265359;
518     TH1F* h1 = (TH1F*)inFile->Get(hist);
519 // TF1* lingaus = new TF1("lingaus","pol1(0)+gaus(2)");
520 TF1* lin = new TF1("lin","[0]+[1]*x");
521 TF1* lingaus = new TF1("lingaus","[0]+[1]*x+[2]/(sqrt(2.0*pi)*[4])*exp(-((x-[3] )
    ↳ )/(sqrt(2.0)*[4]))**2+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/sqrt(2.0)*[7]) )
    ↳ )**2+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[10]))**2+[11]/(sqrt(2.0)*
    ↳ t(2.0*pi)*[13])*exp(-((x-[12])/sqrt(2.0)*[13]))**2");
522     lingaus->SetParameters(-20,0.022,800,0,0);
523     lingaus->FixParameter(3,cent);
524     lingaus->FixParameter(4,width);
525     lingaus->FixParameter(6,cent2);
526     lingaus->FixParameter(7,width2);
527     lingaus->FixParameter(9,cent3);
528     lingaus->FixParameter(10,width3);
529     lingaus->FixParameter(12,cent4);
530     lingaus->FixParameter(13,width4);
531 // h1->Fit("lingaus", "BLQM", "", xMin, xMax);
532 // lingaus->SetParLimits(3,cent-5,cent+5);
533 // lingaus->SetParLimits(4,width-5,width+5);

```

```

534 // h1->Fit("lingaus", "BLM", "", xMin, xMax);
535
536 for(int i=1; i<=500; i++)
537 {
538     h1->Fit("lingaus", "LLQM", "", xMin, xMax);
539 }
540 TFitResultPtr r = h1->Fit("lingaus", "LLS", "", xMin, xMax);
541 lin->SetParameter(0, lingaus->GetParameter(0));
542 lin->SetParameter(1, lingaus->GetParameter(1));
543 Double_t bkgd_integral = lin->Integral(xMin, xMax);
544 Double_t c0_err = lingaus->GetParError(0);
545 Double_t c1_err = lingaus->GetParError(1);
546 Double_t bkgd_integral_error = TMath::Sqrt(
547     ↪ TMath::Power((xMax-xMin)*c0_err,2.0) +
548     ↪ TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
549
550 Double_t fn_integral = lingaus->Integral(xMin, xMax);
551 Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
552     ↪ r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
553 Double_t net_area = fn_integral - bkgd_integral;
554 Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
555     ↪ TMath::Power(bkgd_integral_error,2.0) );
556 r->GetCovarianceMatrix().Print();
557 printf("\n\nFit area = %10.6f+/-%10.6f\nBkgd area = %10.6f+/-%10.6f\nNet area =
558     ↪ %10.6f+/-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ
559     ↪ ral_error,net_area,net_area_error);
560
561 //linear+gaussian+sine
562 void FitterT(TFile* inFile, char* hist, double cent, double width, double scale,
563     ↪ double freq, double phase, int xMin, int xMax)
564 {
565     double pi = 3.14159265359;
566     TH1F* h1 = (TH1F*)inFile->Get(hist);
567     TF1* linsine = new TF1("linsine", "[0]+[1]*x)*(1+[2]*sin([3]*x+[4]))");
568     TF1* lingaus = new TF1("lingaus", "[0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
569     ↪ ])/(sqrt(2.0)*[7]))**2)*(1+[2]*sin([3]*x+[4]))");
570     lingaus->SetParameters(0,0,0,0,0,0,0,0);
571     lingaus->FixParameter(6,cent);
572     lingaus->FixParameter(7,width);
573     lingaus->FixParameter(2,scale);
574     lingaus->FixParameter(3,freq);
575     lingaus->FixParameter(4,phase);
576
577     for(int i=1; i<=500; i++)
578     {
579         h1->Fit("lingaus", "BLQM", "", xMin, xMax);
580     }
581     TFitResultPtr r = h1->Fit("lingaus", "LLS", "", xMin, xMax);
582     linsine->SetParameter(0, lingaus->GetParameter(0));
583     linsine->SetParameter(1, lingaus->GetParameter(1));
584     linsine->SetParameter(2,scale);
585     linsine->SetParameter(3,freq);

```

```

580     linsine->SetParameter(4,phase);
581     Double_t bkgd_integral = linsine->Integral(xMin,xMax);
582 //  Double_t c0_err = lingaus->GetParError(0);
583 //  Double_t c1_err = lingaus->GetParError(1);
584 //  Double_t bkgd_integral_error = TMath::Sqrt(
585 //    TMath::Power((xMax-xMin)*c0_err,2.0) +
586 //    TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
587     Double_t bkgd_integral_error = TMath::Sqrt(bkgd_integral);
588
589     Double_t fn_integral = lingaus->Integral(xMin,xMax);
590     Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
591     r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
592     Double_t net_area = fn_integral - bkgd_integral;
593     Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
594     TMath::Power(bkgd_integral_error,2.0) );
595     r->GetCovarianceMatrix().Print();
596     printf("\n\nFit area = %10.6f+/-%10.6f\nBkgd area = %10.6f+/-%10.6f\nNet area =
597     %10.6f+/-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ
598     ral_error,net_area,net_area_error);
599 }
600
601 //linear+gaussian*2+sine
602 void FitterT(TFile* inFile, char* hist, double cent, double width, double cent2,
603   double width2, double scale, double freq, double phase, int xMin, int xMax)
604 {
605   double pi = 3.14159265359;
606   TH1F* h1 = (TH1F*)inFile->Get(hist);
607   TF1* linsine = new TF1("linsine","([0]+[1]*x)*(1+[2]*sin([3]*x+[4]))");
608   TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
609     ])/(sqrt(2.0)*[7]))**2+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[1]
610     ])**2)*([1]+[2]*sin([3]*x+[4]))");
611   lingaus->SetParameters(0,0,0,0,0,0,0,0,0);
612   lingaus->FixParameter(6,cent);
613   lingaus->FixParameter(7,width);
614   lingaus->FixParameter(9,cent2);
615   lingaus->FixParameter(10,width2);
616   lingaus->FixParameter(2,scale);
617   lingaus->FixParameter(3,freq);
618   lingaus->FixParameter(4,phase);
619
620   for(int i=1; i<=500; i++)
621   {
622     h1->Fit("lingaus","BLQM","",xMin,xMax);
623   }
624   TFitResultPtr r = h1->Fit("lingaus","LLS","",xMin,xMax);
625   linsine->SetParameter(0, lingaus->GetParameter(0));
626   linsine->SetParameter(1, lingaus->GetParameter(1));
627   linsine->SetParameter(2,scale);
628   linsine->SetParameter(3,freq);
629   linsine->SetParameter(4,phase);
630   Double_t bkgd_integral = linsine->Integral(xMin,xMax);
631 //  Double_t c0_err = lingaus->GetParError(0);
632 //  Double_t c1_err = lingaus->GetParError(1);

```

```

624 // Double_t bkgd_integral_error = TMath::Sqrt(
625   ↵ TMath::Power((xMax-xMin)*c0_err,2.0) +
626   ↵ TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
627   Double_t bkgd_integral_error = TMath::Sqrt(bkgd_integral);
628
629   Double_t fn_integral = lingaus->Integral(xMin,xMax);
630   Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
631   ↵ r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
632   Double_t net_area = fn_integral - bkgd_integral;
633   Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
634   ↵ TMath::Power(bkgd_integral_error,2.0) );
635   r->GetCovarianceMatrix().Print();
636   printf("\n\nFit area = %10.6f+-%10.6f\nBkgd area = %10.6f+-%10.6f\nNet area =
637   ↵ %10.6f+-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ ]
638   ↵ ral_error,net_area,net_area_error);
639 }
640
641 //linear+gaussian*3+sine
642 void FitterT(TFile* inFile, char* hist, double cent, double width, double cent2,
643   ↵ double width2, double cent3, double width3, double scale, double freq, double
644   ↵ phase, int xMin, int xMax)
645 {
646   double pi = 3.14159265359;
647   TH1F* h1 = (TH1F*)inFile->Get(hist);
648   TF1* linsine = new TF1("linsine","([0]+[1]*x)*(1+[2]*sin([3]*x+[4]))");
649   TF1* lingaus = new TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6]
650   ↵ ])/(sqrt(2.0)*[7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[1]
651   ↵ 0]))**2)+[11]/(sqrt(2.0*pi)*[13])*exp(-((x-[12])/sqrt(2.0)*[13]))**2)*(1+
652   ↵ [2]*sin([3]*x+[4]))");
653   lingaus->SetParameters(0,0,0,0,0,0,0,0,0);
654   lingaus->FixParameter(6,cent);
655   lingaus->FixParameter(7,width);
656   lingaus->FixParameter(9,cent2);
657   lingaus->FixParameter(10,width2);
658   lingaus->FixParameter(12,cent3);
659   lingaus->FixParameter(13,width3);
660   lingaus->FixParameter(2,scale);
661   lingaus->FixParameter(3,freq);
662   lingaus->FixParameter(4,phase);
663
664   for(int i=1; i<=500; i++)
665   {
666     h1->Fit("lingaus","BLQM","",xMin,xMax);
667   }
668   TFitResultPtr r = h1->Fit("lingaus","LLS","",xMin,xMax);
669   linsine->SetParameter(0, lingaus->GetParameter(0));
670   linsine->SetParameter(1, lingaus->GetParameter(1));
671   linsine->SetParameter(2,scale);
672   linsine->SetParameter(3,freq);
673   linsine->SetParameter(4,phase);
674   Double_t bkgd_integral = linsine->Integral(xMin,xMax);
675   // Double_t c0_err = lingaus->GetParError(0);
676   // Double_t c1_err = lingaus->GetParError(1);

```

```

666 // Double_t bkgd_integral_error = TMath::Sqrt(
667   ↵ TMath::Power((xMax-xMin)*c0_err,2.0) +
668   ↵ TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
669   Double_t bkgd_integral_error = TMath::Sqrt(bkgd_integral);
670
671   Double_t fn_integral = lingaus->Integral(xMin,xMax);
672   Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
673   ↵ r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
674   Double_t net_area = fn_integral - bkgd_integral;
675   Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
676   ↵ TMath::Power(bkgd_integral_error,2.0) );
677   r->GetCovarianceMatrix().Print();
678   printf("\n\nFit area = %10.6f+/-%10.6f\nBkgd area = %10.6f+/-%10.6f\nNet area =
679   ↵ %10.6f+/-%10.6f\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ ]
680   ↵ ral_error,net_area,net_area_error);
681 }
682
683 //linear+gaussian*4+sine
684 void FitterT(TFile* inFile, char* hist, double cent, double width, double cent2,
685   ↵ double width2, double cent3, double width3, double cent4, double width4,
686   ↵ double scale, double freq, double phase, int xMin, int xMax)
687 {
688   double pi = 3.14159265359;
689   TH1F* h1 = (TH1F*)inFile->Get(hist);
690   TF1* linsine = new TF1("linsine","([0]+[1]*x)*(1+[2]*sin([3]*x+[4]))");
691   TF1* lingaus = new
692   ↵ TF1("lingaus","([0]+[1]*x+[5]/(sqrt(2.0*pi)*[7])*exp(-((x-[6])/sqrt(2.0)*[ ]
693   ↵ [7]))**2)+[8]/(sqrt(2.0*pi)*[10])*exp(-((x-[9])/sqrt(2.0)*[10]))**2)+[11]/( )
694   ↵ sqrt(2.0*pi)*[13])*exp(-((x-[12])/sqrt(2.0)*[13]))**2)+[14]/(sqrt(2.0*pi)*
695   ↵ [16])*exp(-((x-[15])/sqrt(2.0)*[16]))**2)*(1+[2]*sin([3]*x+[4]))");
696   lingaus->SetParameters(0,0,0,0,0,0,0,0);
697   lingaus->FixParameter(6,cent);
698   lingaus->FixParameter(7,width);
699   lingaus->FixParameter(9,cent2);
700   lingaus->FixParameter(10,width2);
701   lingaus->FixParameter(12,cent3);
702   lingaus->FixParameter(13,width3);
703   lingaus->FixParameter(15,cent4);
704   lingaus->FixParameter(16,width4);
705   lingaus->FixParameter(2,scale);
706   lingaus->FixParameter(3,freq);
707   lingaus->FixParameter(4,phase);
708
709   for(int i=1; i<=500; i++)
710   {
711     h1->Fit("lingaus","BLQM","",xMin,xMax);
712   }
713   TFitResultPtr r = h1->Fit("lingaus","LLS","",xMin,xMax);
714   linsine->SetParameter(0, lingaus->GetParameter(0));
715   linsine->SetParameter(1, lingaus->GetParameter(1));
716   linsine->SetParameter(2,scale);
717   linsine->SetParameter(3,freq);
718   linsine->SetParameter(4,phase);
719   Double_t bkgd_integral = linsine->Integral(xMin,xMax);

```

```

708 // Double_t c0_err = lingaus->GetParError(0);
709 // Double_t c1_err = lingaus->GetParError(1);
710 // Double_t bkgd_integral_error = TMath::Sqrt(
711 //   → TMath::Power((xMax-xMin)*c0_err,2.0) +
712 //   → TMath::Power(0.5*(xMax-xMin)*(xMax-xMin)*c1_err,2.0) );
713 Double_t bkgd_integral_error = TMath::Sqrt(bkgd_integral);
714
715 Double_t fn_integral = lingaus->Integral(xMin,xMax);
716 Double_t fn_integral_error = lingaus->IntegralError( xMin, xMax,
717   → r->GetParams(), r->GetCovarianceMatrix().GetMatrixArray() );
718 Double_t net_area = fn_integral - bkgd_integral;
719 Double_t net_area_error = TMath::Sqrt( TMath::Power(fn_integral_error,2.0) +
720   → TMath::Power(bkgd_integral_error,2.0) );
721 r->GetCovarianceMatrix().Print();
722 printf("\n\nFit area = %10.6f+-%10.6f\nBkgd area = %10.6f+-%10.6f\nNet area =
723   → %10.6f+-%10.6f\n\n",fn_integral,fn_integral_error,bkgd_integral,bkgd_integ |
724   → ral_error,net_area,net_area_error);
725 }
```

D.2 skewedsubtraction.cxx

This code is used for subtracting off conversion electron peaks, typically those from the ground state band in this work, from the conversion electron spectrum to look for smaller peaks and upper limits.

```

1 /* This macro is meant to subtract off a skewed gaussian of the given
2 parameters.
3
4 The conversion coefficient, area of the gamma, efficiency of the
5 gamma detector, and efficiency of the electron detector are used to calculate
6 the area of the skewed gaussian, and give an input to the height.
7
8 A best guess based on other data must be used for the width/sigma. R and beta
9 are taken from calibration data. The centroid is based off two things: the gamma
10 centroid and the electron orbital given.*/
11
12 TH1F* Subtraction(TH1F* hist, double dICC, double dGeEff, double dSiLiEff, double
13   → dGeArea, double dCorrRatio, double dGeCentroid, int nOrbital, double dR,
14   → double dBeta, double dSigma, int xMin, int xMax)
15 {
16     double dSiLiCentroid = 0;
17     double dSkewCentroid = 0;
18     double dSiLiArea = 0;
19     double dHeight = 0;
20     TH1F* hSubtracted = hist->Clone();
21     //Getting sili centroid
22     switch(nOrbital)
```

```

21     {
22         case 0:
23             dSiLiCentroid = dGeCentroid - 50.239;
24             break;
25         case 1:
26             dSiLiCentroid = dGeCentroid - 7.9303;
27             break;
28         case 2:
29             dSiLiCentroid = dGeCentroid - 1.3;
30             break;
31         default:
32             cout << "Invalid electron orbital" << endl;
33             return nullptr;
34     }
35     //calculating area under sili peak
36     dSiLiArea = dICC*dGeArea/dGeEff*dSiLiEff*1/dCorrRatio;
37     cout << "SiLi Area: " << dSiLiArea << endl;
38     //calculating the height from the area.
39     dHeight = dSiLiArea*100/(2*exp(-dSigma**2/(2*dBeta**2))*dR*dBeta-sqrt(2*3.141
40     ↳ 5926)*(dR-100)*dSigma);
41     cout << "SiLi Height: " << dHeight << endl;
42     //calculating the skewed shift in the centroid
43     dSkewCentroid = (2*dBeta*dBeta*exp(-dSigma*dSigma/(2*dBeta*dBeta)))/((dR-100)
44     ↳ *sqrt(2*3.14159)*dSigma+2*dR*dBeta*exp(-dSigma*dSigma/(2*dBeta*dBeta)));
45     std::cout << "Centroid shift: " << dSkewCentroid << std::endl;
46     dSkewCentroid = dSiLiCentroid-dSkewCentroid;
47     //Skewed gaussian function being set
48     TF1* skewed = new TF1("skewed","[0]*(1-[3]/100)*exp(-((x-[1])/(sqrt(2.0)*[2])
49     ↳ )**2)+[0]*[3]/100*exp((x-[5])/[4])*TMath::Erfc((x-[5])/(sqrt(2.0)*[2])+[2]
50     ↳ ]/(sqrt(2.0)*[4]))",
51     ↳ xMin,xMax);
52     skewed->SetParNames("height","peak","sigma","R","beta","centroid");
53     skewed->SetParameters(dHeight,dSiLiCentroid,dSigma,dR,dBeta,dSiLiCentroid);
54
55     //begin the subtraction
56     for (int i=xMin; i < xMax; i++)
57     {
58         hSubtracted->SetBinContent(i,hSubtracted->GetBinContent(i)-skewed->Eval(h
59         ↳ Subtracted->GetBinCenter(i)));
60     }
61
62     TCanvas* c1 = new TCanvas();
63     c1->Divide(1,2);
64     c1->cd(1);
65     hist->Draw();
66     skewed->SetLineColor(1);
67     skewed->Draw("same");
68     hSubtracted->SetLineColor(2);
69     hSubtracted->Draw("same");
70     c1->cd(2);
71     hSubtracted->Draw();
72     return hSubtracted;
73 }
```

D.3 Piecewise_macro.cxx

This code was used for fitting the background on either side of a peak and outputting the area under the peak after subtracting the background off. It must be compiled by root for use by being converted into a shared library by ACLiC.

```
1 //+ Combined (simultaneous) fit of two histogram with separate functions
2 // and some common parameters
3 //
4 // See http://root.cern.ch/phpBB3/viewtopic.php?f=3&t=11740#p50908
5 // for a modified version working with Fumili or GSLMultiFit
6 //
7 // N.B. this macro must be compiled with ACLiC
8 //
9 //Author: L. Moneta - Dec 2010
10
11 #include "Fit/Fitter.h"
12 #include "Fit/BinData.h"
13 #include "Fit/Chi2FCN.h"
14 #include "TH1.h"
15 #include "TList.h"
16 #include "Math/WrappedMultiTF1.h"
17 #include "HFitInterface.h"
18 #include "TCanvas.h"
19 #include "TStyle.h"
20 #include "TMath.h"
21
22
23 // definition of shared parameter
24 // background function, left
25 int iparBL[3] = { 0,           // Constant background value
26                  1,           // Linear background value
27                  2, //Shift of "0"
28 };
29
30 //background function, right
31 int iparBR[3] = { 0, // Constant background value
32                   1, // Linear background value
33                   2, //Shift of "0"
34 };
35
36 struct GlobalChi2 {
37     GlobalChi2( ROOT::Math::IMultiGenFunction & f1,
38                 ROOT::Math::IMultiGenFunction & f2) :
39         fChi2_1(&f1), fChi2_2(&f2) {}
40
41 // parameter vector is first background (in common 1 and 2)
42 // and then is signal (only in 2)
43     double operator() (const double *par) const {
```

```

44     double p1[3];
45     for (int i = 0; i < 3; ++i) p1[i] = par[iparBL[i] ];
46
47     double p2[3];
48     for (int i = 0; i < 3; ++i) p2[i] = par[iparBR[i] ];
49
50     return (*fChi2_1)(p1) + (*fChi2_2)(p2);
51 }
52
53 const ROOT::Math::IMultiGenFunction * fChi2_1;
54 const ROOT::Math::IMultiGenFunction * fChi2_2;
55 };
56
57 void combinedFit(TH1F* hist, double xMin, double xStart, double xEnd, double
58   → xMax) {
59   // TH1D * hB = new TH1D("hB","histo B",100,0,100);
60   // TH1D * hSB = new TH1D("hSB","histo S+B",100, 0,100);
61
62   // TF1 * fB = new TF1("fB","expo",0,100);
63   // fB->SetParameters(1,-0.05);
64   // hB->FillRandom("fB");
65
66   // TF1 * fS = new TF1("fS","gaus",0,100);
67   // fS->SetParameters(1,30,5);
68
69   // hSB->FillRandom("fB",2000);
70   // hSB->FillRandom("fS",1000);
71
72   // perform now global fit
73
74   //TF1* fBL = new TF1("fBL","pol1",xMin,xStart);
75   //TF1* fBR = new TF1("fBR","pol1",xEnd,xMax);
76
77   TF1* fBL = new TF1("fBL","[0]+[1]*(x-[2])",xMin,xStart);
78   TF1* fBR = new TF1("fBR","[0]+[1]*(x-[2])",xEnd,xMax);
79   fBL->FixParameter(2,xMin);
80   fBR->FixParameter(2,xMin);
81   // TF1 * fSB = new TF1("fSB","expo + gaus(2)",0,100);
82
83   ROOT::Math::WrappedMultiTF1 wfBL(*fBL,1);
84   ROOT::Math::WrappedMultiTF1 wfBR(*fBR,1);
85
86   ROOT::Fit::DataOptions opt;
87   ROOT::Fit::DataRange rangeBL;
88   // set the data range
89   rangeBL.SetRange(xMin,xStart);
90   ROOT::Fit::BinData dataBL(opt,rangeBL);
91   ROOT::Fit::FillData(dataBL, hist);
92
93   ROOT::Fit::DataRange rangeBR;
94   rangeBR.SetRange(xEnd,xMax);
95   ROOT::Fit::BinData dataBR(opt,rangeBR);
96   ROOT::Fit::FillData(dataBR, hist);

```

```

97
98     ROOT::Fit::Chi2Function chi2_BL(dataBL, wfBL);
99     ROOT::Fit::Chi2Function chi2_BR(dataBR, wfBR);
100
101    GlobalChi2 globalChi2(chi2_BL, chi2_BR);
102
103    ROOT::Fit::Fitter fitter;
104
105    const int Npar = 3;
106    double par0[Npar] = {0,0,xMin};
107
108    // create before the parameter settings in order to fix or set range on them
109    fitter.Config().SetParamsSettings(3,par0);
110    // // fix 5-th parameter
111    fitter.Config().ParSettings(2).Fix();
112    // // set limits on the third and 4-th parameter
113    // fitter.Config().ParSettings(2).SetLimits(-10,-1.E-4);
114    // fitter.Config().ParSettings(3).SetLimits(0,10000);
115    // fitter.Config().ParSettings(3).SetStepSize(5);
116
117    fitter.Config().MinimizerOptions().SetPrintLevel(0);
118    fitter.Config().SetMinimizer("Minuit2","Migrad");
119
120    // fit FCN function directly
121    // (specify optionally data size and flag to indicate that is a chi2 fit)
122    fitter.FitFCN(3,globalChi2,0,dataBL.Size()+dataBR.Size(),true);
123    ROOT::Fit::FitResult result = fitter.Result();
124    result.Print(std::cout);
125    hist->Draw();
126    fBL->SetFitResult(result, iparBL);
127    fBL->SetRange(rangeBL().first, rangeBL().second);
128    fBL->Draw("same");
129    fBR->SetFitResult(result, iparBR);
130    fBR->SetRange(rangeBR().first, rangeBR().second);
131    fBR->Draw("same");
132
133    TF1* fB = new TF1("fB","[0]+[1]*(x-[2])");
134    fB->SetFitResult(result, iparBL);
135    double dBkdSigma =
136         $\sqrt{(\text{Power}((xEnd-xStart)*fB->\text{GetParError}(0),2)+\text{Power}(0.5*fB->\text{GetParError}(1)*(\text{Power}(xEnd,2)-\text{Power}(xStart,2)-2*xMin),2))}$ ;
137    //double dBkdArea =
138         $fB->\text{GetParameter}(0)*(xEnd-xStart)+fB->\text{GetParameter}(1)*(0.5*xEnd*xEnd - 0.5*xStart*xStart - xMin);$ 
139    double dBkdArea = fB->Integral(xStart,xEnd);
140
141    double dArea = 0;
142    for(int i=xStart; i <= xEnd; i++)
143    {
144        if(hist->GetBinContent(i)>0)
145        {
146            dArea = dArea + hist->GetBinContent(i)-fB->Eval(i+0.5);

```

```

145     // std::cout << i << "\t" << hist->GetBinContent(i) << "\t" <<
146     // fB->Eval(i+0.5) << "\t" << hist->GetBinContent(i)-fB->Eval(i+0.5) <<
147     // std::endl;
148 }
149 std::cout << "Centroid: " << (xStart+xEnd)/2.0 << std::endl;
150 std::cout << "Area: " << dArea << " +/- " << sqrt(dArea) << " +/- " <<
151     // sqrt(dBkdArea) << std::endl;
152 std::cout << "Background Area: " << dBkdArea << std::endl;
153
154 // TCanvas * c1 = new TCanvas("Simfit", "Simultaneous fit of two histograms",
155 //                             10, 10, 700, 700);
156 // c1->Divide(1,2);
157 // c1->cd(1);
158 // gStyle->SetOptFit(1111);
159
160 // fB->SetFitResult(result, iparB);
161 // fB->SetRange(rangeB().first, rangeB().second);
162 // fB->SetLineColor(kBlue);
163 // hB->GetListOfFunctions()->Add(fB);
164 // hB->Draw();
165
166 // c1->cd(2);
167 // fSB->SetFitResult(result, iparSB);
168 // fSB->SetRange(rangeSB().first, rangeSB().second);
169 // fSB->SetLineColor(kRed);
170 // hSB->GetListOfFunctions()->Add(fSB);
171 // hSB->Draw();
172
173 }
174 void Piecewise(TH1F* hist, double xMin, double xStart, double xEnd, double xMax)
175 {
176     TF1* piecewise = new TF1("piecewise", "[0]+[1]*x)*(TMath::Erfc((x-[2])*1000)+
177     // TMath::Erf((x-[3])*1000))", xMin,
178     // xMax);
179     piecewise->FixParameter(2,xStart);
180     piecewise->FixParameter(3,xEnd);
181     piecewise->FixParameter(1,0);
182     piecewise->FixParameter(0,36223);
183     hist->Fit("piecewise","BLS","", xMin, xMax);
184 }
```

D.4 ROOT2SPE.cxx

This code was used for converting ROOT histograms into .spe files for use by RADWARE [50]. It uses an existing conversion in the RADWARE source code to

convert from ascii to spe.

```
1  /* Quick macro to convert root histogram to ascii*/
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <unistd.h>
5  #include <math.h>
6  #include <string.h>
7
8  /* ===== */
9  int file_error(char *error_type, char *filename)
10 {
11     /* write error message */
12     /* cannot perform operation error_type on file filename */
13
14     if (strlen(error_type) + strlen(filename) > 58) {
15         warn1("ERROR - cannot %s file\n%s\n", error_type, filename);
16     } else {
17         warn1("ERROR - cannot %s file %s\n", error_type, filename);
18     }
19     return 0;
20 } /* file_error */
21
22 /* ===== */
23 int put_file_rec(FILE *fd, void *data, int numbytes)
24 {
25     /* write one fortran-unformatted style binary record into data */
26     /* returns 1 for error */
27
28 #ifdef VMS /* vms */
29     int j1;
30     short rh[2];
31     char *buf;
32
33     buf = data;
34     j1 = numbytes;
35     if (numbytes <= 2042) {
36         rh[0] = numbytes + 2; rh[1] = 3;
37     } else {
38         rh[0] = 2044; rh[1] = 1;
39         while (j1 > 2042) {
40             if (fwrite(rh, 2, 2, fd) != 2 ||
41                 fwrite(buf, 2042, 1, fd) != 1) return 1;
42             rh[1] = 0; j1 -= 2042; buf += 2042;
43         }
44         rh[0] = j1 + 2; rh[1] = 2;
45     }
46     if (fwrite(rh, 2, 2, fd) != 2 ||
47         fwrite(buf, j1, 1, fd) != 1) return 1;
48     /* if numbytes is odd, write an extra (padding) byte */
49     if (2*(numbytes>>1) != numbytes) {
50         j1 = 0;
```

```

51     fwrite(&j1, 1, 1, fd);
52 }
53
54 #else /* unix */
55
56 if (fwrite(&numbytes, 4, 1, fd) != 1 ||
57     fwrite(data, numbytes, 1, fd) != 1 ||
58     fwrite(&numbytes, 4, 1, fd) != 1) return 1;
59 #endif
60     return 0;
61 } /*put_file_rec */
62
63 /* ===== */
64
65 int wspec(char *filnam, float *spec, int idim)
66 {
67     /* write spectra in gf3 format
68      filnam = name of file to be created and written
69      spec = spectrum of length idim */
70
71     char buf[32];
72     int j, c1 = 1, rl = 0;
73     char namesp[8];
74     FILE *file;
75
76     file = fopen(filnam, "w");
77     //if (!(file = open_new_file(filnam, 0))) return 1;
78     //strncpy(namesp, filnam, 8);
79     //if (j < 8) memset(&namesp[j], ' ', 8-j);
80
81     /* WRITE(1) NAMESP, IDIM, 1, 1, 1 */
82     /* WRITE(1) SPEC */
83     #define W(a,b) { memcpy(buf + rl, a, b); rl += b; }
84     W(namesp,8); W(&idim,4); W(&c1,4); W(&c1,4); W(&c1,4);
85     #undef W
86     cout << "trying to put it in the file..." << endl;
87     if (put_file_rec(file, buf, rl) ||
88         put_file_rec(file, spec, 4*idim)) {
89         file_error("write to", filnam);
90         fclose(file);
91         return 1;
92     }
93     cout << "idk man" << endl;
94     fclose(file);
95     return 0;
96 }
97
98 /* ===== */
99
100 /* Macro to convert root histogram to spe for radware*/
101
102 void r2s(TH1F* inName, char* fileOut)
103 {
104     float spec[16384];

```

```

105     int idim1;
106     int i, numch;
107
108     TH1F* specToConv = inName;
109     int n = specToConv->GetNbinsX();
110
111     /** if (file_out == NULL)
112         printf("Sorry, but the ascii output file did not open.");
113     */
114     i = 0;
115     while (i<=(n+1) && i < 16384)
116     {
117         //file_out << "\t" << specToConv->GetBinCenter(i) << ", " << "\t" <<
118         //specToConv->GetBinContent(i) << endl;
119         spec[i++] = specToConv->GetBinContent(i);
120     }
121     printf("%i lines read...\n", i);
122
123     numch = idim1 = i;
124     printf(" %i channels..\n", numch);
125
126     wspec(Form("%s.spe",fileOut), spec, numch);
127     /* tell user that the file has been converted */
128     printf(" histogram ==> %s.spe, %i chs.\n", fileOut, numch);
129 }
130 /* ===== */
131
132 void r2a(TH1F* inName, char* fileOut)
133 {
134     ofstream file_out(fileOut);
135     TH1F* specToConv = inName;
136     int n = specToConv->GetNbinsX();
137
138     /** if (file_out == NULL)
139         printf("Sorry, but the ascii output file did not open.");
140     */
141     for (int i=0; i<=(n+1); i++)
142     {
143         file_out << "\t" << specToConv->GetBinCenter(i) << ", " << "\t" <<
144         specToConv->GetBinContent(i) << endl;
145     }
146     file_out.close();
147 }
148
149 void fileR2S(char* fileIn, char* fileOut, int nStart, int nEnd)
150 {
151     TFile* file = new TFile(fileIn);
152     cout << fileIn << " open" << endl;
153     for(int i=nStart;i<nEnd;i++)
154     {
155         //r2s((TH1F*)file->Get(Form("clover_raw_%d",i)),Form("%s%d%d",fileOut,(int)i]
156         //      ↵ /4,i%4));

```

```

156     //r2s((TH1F*)file->Get(Form("Clover_%d",i)),Form("%s%d%d",fileOut,(int)i/4,i]
157     //→    %4)); //Special case right
158     //→    now
159 }
160
161 void fileR2S(char* fileIn, char* fileOut)
162 {
163     TFile* file = new TFile(fileIn);
164     cout << fileIn << " open" << endl;
165     for(int i=0;i<2;i++)
166     {
167         r2s((TH1F*)file->Get(Form("Clover_%d",i)),Form("%s_C%d",fileOut,i));
168     }
169     for(int i=1;i<7;i++)
170     {
171         r2s((TH1F*)file->Get(Form("SiLi%d",i)),Form("%s_S%d",fileOut,i));
172     }
173     file->Close();
174 }
```

BIBLIOGRAPHY

1. J. Allison, K. Amako, J. Apostolakis, P. Arce, M. Asai, T. Aso, E. Bagli, A. Bagulya, S. Banerjee, G. Barrand, B. R. Beck, A. G. Bogdanov, D. Brandt, J. M. C. Brown, H. Burkhardt, P. Canal, D. Cano-Ott, S. Chauvie, K. Cho, G. A. P. Cirrone, G. Cooperman, M. A. Cortés-Giraldo, G. Cosmo, G. Cuttone, G. Depaola, L. Desorgher, X. Dong, A. Dotti, V. D. Elvira, G. Folger, Z. Francis, A. Galoyan, L. Garnier, M. Gayer, K. L. Genser, V. M. Grichine, S. Guatelli, P. Guèye, P. Gumplinger, A. S. Howard, I. Hřivnáčová, S. Hwang, S. Incerti, A. Ivanchenko, V. N. Ivanchenko, F. W. Jones, S. Y. Jun, P. Kaitaniemi, N. Karakatsanis, M. Karamitros, M. Kelsey, A. Kimura, T. Koi, H. Kurashige, A. Lechner, S. Lee, F. Longo, M. Maire, D. Mancusi, A. Mantero, E. Mendoza, B. Morgan, K. Murakami, T. Nikitina, L. Pandola, P. Paprocki, J. Perl, I. Petrović, M. G. Pia, W. Pokorski, J. M. Quesada, M. Raine, M. A. Reis, A. Ribon, A. R. Fira, F. Romano, G. Russo, G. Santin, T. Sasaki, D. Sawkey, J. I. Shin, I. I. Strakovský, A. Taborda, S. Tanaka, B. Tomé, T. Toshito, H. N. Tran, P. R. Truscott, L. Urban, V. Uzhinsky, J. M. Verbeke, M. Verderi, B. Wendt, H. Wenzel, D. H. Wright, D. M. Wright, T. Yamashita, J. Yarba, and H. Yoshida. Recent developments in Geant4. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 835:186 – 225, 2016. ISSN 0168-9002. doi: <https://doi.org/10.1016/j.nima.2016.06.125>. URL <http://www.sciencedirect.com/science/article/pii/S0168900216306957>.
2. AMETEK ORTEC. 579 Fast-Filter Amplifier. <https://www.ortec-online.com/products/electronics/amplifiers/579>.
3. A. Aprahamian, R. C. de Haan, S. R. Lesher, C. Casarella, A. Stratman, H. G. Börner, H. Lehmann, M. Jentschel, and A. M. Bruce. Lifetime measurements in ^{156}Gd . *Phys. Rev. C*, 98:034303, September 2018. doi: 10.1103/PhysRevC.98.034303. URL <https://link.aps.org/doi/10.1103/PhysRevC.98.034303>.
4. Autodesk. Inventor professional-grade 3d cad software for product design and engineering. <https://www.autodesk.com/products/inventor/overview>.
5. A. Battaglia. *Conversion Coefficients Measurements of ^{176}Lu Using ICEBall*. PhD thesis, University of Notre Dame, May 2015.

6. C. Beausang, C. Barton, M. Caprio, R. Casten, J. Cooper, R. Krücken, B. Liu, J. Novak, Z. Wang, M. Wilhelm, A. Wilson, N. Zamfir, and A. Zilges. The YRAST Ball array. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 452(3):431 – 439, 2000. ISSN 0168-9002. doi: [https://doi.org/10.1016/S0168-9002\(00\)00449-6](https://doi.org/10.1016/S0168-9002(00)00449-6). URL <http://www.sciencedirect.com/science/article/pii/S0168900200004496>.
7. P. R. Bevington and D. K. Robinson. *Data Reduction and Error Analysis for the Physical Sciences*. McGraw-Hill, 3rd edition, 2003. ISBN 0-07-247227-8.
8. L. C. Biedenharn and M. E. Rose. Theory of angular correlation of nuclear radiations. *Reviews of Modern Physics*, 25(3):729–777, July 1953.
9. J. Black, W. Caelli, D. Livesey, and R. Watson. Search for the lowest $T = 2$ state of ^{8}Be : Experimental. *Physics Letters B*, 30(2):100 – 102, 1969. ISSN 0370-2693. doi: [https://doi.org/10.1016/0370-2693\(69\)90408-0](https://doi.org/10.1016/0370-2693(69)90408-0). URL <http://www.sciencedirect.com/science/article/pii/0370269369904080>.
10. J. M. Blatt and V. F. Weisskopf. *Theoretical Nuclear Physics*. Springer-Verlag New York Inc., 1979.
11. A. Bohr and B. Mottelson. Moments of Inertia of Rotating Nuclei. *Kgl. Danske Videnskab. Selskab, Mat.-fys. Medd.*, 30(1), 1955.
12. R. Brun and F. Rademakers. Root - an object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 389(1):81 – 86, 1997. ISSN 0168-9002. doi: [https://doi.org/10.1016/S0168-9002\(97\)00048-X](https://doi.org/10.1016/S0168-9002(97)00048-X). URL <http://www.sciencedirect.com/science/article/pii/S016890029700048X>. New Computing Techniques in Physics Research V.
13. CAEN. CAEN V775. <https://www.caen.it/products/v775/>, .
14. CAEN. CAEN V830. <https://www.caen.it/products/v830/>, .
15. R. F. Casten. *Nuclear Structure from a Simple Perspective*. Oxford University Press, Inc., 1st edition, 1990.
16. E. Caurier, J. Menéndez, F. Nowacki, and A. Poves. Coexistence of spherical states with deformed and superdeformed bands in doubly magic ^{40}Ca : A shell-model challenge. *Phys. Rev. C*, 75:054317, May 2007. doi: 10.1103/PhysRevC.75.054317. URL <https://link.aps.org/doi/10.1103/PhysRevC.75.054317>.
17. G. Choppin, J.-O. Liljenzin, J. Rydberg, and C. Ekberg. *Radiochemistry and Nuclear Chemistry*. Academic Press, Oxford, 4th edition, 2013.

18. E. L. Church and J. Weneser. Electric-monopole transitions in atomic nuclei. *Physical Review*, 103(4):1035–1044, August 1956.
19. E. L. Church, M. E. Rose, and J. Weneser. Electric-monopole directional-correlation experiments. *Physical Review*, 109(4):1299–1306, February 1958.
20. S. Clifford, X. Guo-ji, C. Ingelbrecht, and M. J. Pomeroy. Processes for the production of ultra-pure metals from oxide and their cold rolling to ultra-thin foils for use as targets and as reference materials. *Nuclear Instruments and Methods in Physics Research A*, (480):29–35, 2002.
21. COMSOL Inc. COMSOL Multiphysics Software.
<https://www.comsol.com/comsol-multiphysics>.
22. J. P. Delaroche, M. Girod, J. Libert, H. Goutte, S. Hilaire, S. Péru, N. Pillet, and G. F. Bertsch. Structure of even-even nuclei using a mapped collective hamiltonian and the d1s gogny interaction. *Physical Review C*, 81:014303, Jan 2010.
23. U. Fano. Ionization yield of radiation ii. the fluctuations of the number of ions. *Physical Review*, 72(1):26–29, July 1947.
24. D. G. Fleming, C. Günther, G. Hagemann, B. Herskind, and P. O. Tjøm. Study of the (p, t) reaction on the even gadolinium nuclei. *Phys. Rev. C*, 8:806–818, Aug 1973. doi: 10.1103/PhysRevC.8.806. URL
<https://link.aps.org/doi/10.1103/PhysRevC.8.806>.
25. Y. Gono and T. T. Sugihara. Electric monopole admixtures in interband transitions of ^{154}Gd . *Physical Review C*, 10(6):2460–2466, 1974.
26. K. Heyde and J. L. Wood. Shape coexistence in atomic nuclei. *Reviews of Modern Physics*, 83:1467–1521, 2011.
27. G. Ilie and R. F. Casten. $E0$ transitions in deformed nuclei. *Physical Review C*, 84:064320, 2011.
28. Institute for Nuclear Astrophysics. GEORGINA A compact Ge-detector Array.
<https://isnap.nd.edu/research/facility/experimental/georgina/>, 2018.
29. J. Van Klinken and K. Wissak. Conversion Electrons Separated from High Background. *Nuclear Instruments and Methods*, 98:1–8, 1972.
30. J. Van Klinken, S. J. Feenstra, K. Wissak and H. Faust. Mini-Orange Spectrometers for In- and Off-Beam Observation of Conversion Electrons. *Nuclear Instruments and Methods*, 130:427–441, 1975.
31. J. D. Jackson. *Classical Electrodynamics*. John Wiley & Sons Inc., 3rd edition, 1999.

32. T. Kibédi, T. W. Burrows, M. B. Trzhaskovskaya, P. M. Davidson, and C. W. N. Jr. Evaluation of theoretical conversion coefficients using bricc. *Nuclear Instrumentation and Methods A*, 589:202–229, 2008.
33. J. Klora, H. G. Börner, T. von Egidy, R. Georgii, J. Jolie, S. Judge, V. A. Khitrov, B. Krusche, V. A. Libman, H. Lindner, L. L. Litvinsky, U. Mayerhofer, A. V. Murzin, S. J. Robinson, A. M. Sukhovojs, and H. Trieb. Nuclear structure of ^{156}Gd studied with (n, γ) , (n, e) , (d, p) , (d, t) reactions and lifetime measurements. *Nuclear Physics A*, 561(1):1 – 73, 1993. ISSN 0375-9474. doi: [https://doi.org/10.1016/0375-9474\(93\)90165-T](https://doi.org/10.1016/0375-9474(93)90165-T). URL <http://www.sciencedirect.com/science/article/pii/037594749390165T>.
34. G. F. Knoll. *Radiation Detection and Measurement*. John Wiley & Sons Inc., 3rd edition, 2000.
35. J. Konijn, F. W. N. D. Boer, A. V. Poelgeest, W. H. A. Hesselink, M. J. A. D. Voigt, and H. Verheul. The level structure of ^{156}Gd studied by means of the $(\alpha, 2n\gamma)$ reaction. *Nuclear Physics A*, 352:191–220, 1981.
36. A. J. Koning, S. Hilaire, and M. C. Duijvestijn. Talys-1.0. In O. Bersillon, F. Gunsing, E. Bauge, R. Jacqmin, and S. Leray, editors, *Proceedings of the International Conference on Nuclear Data for Science and Technology*, pages 211–214, Nice, France, April 22-27 2007. EDP Sciences.
37. S. Lesher, A. Aprahamian, and W. Tan. MRI: Acquisition of Si(Li) Detectors and Two BGO Compton Suppression Shields for the Development of the La Crosse FIREBall. NSF Major Research Instrumentation Award Number 1919364, 2019.
38. S. R. Lesher, A. Aprahamian, L. Trache, A. Oros-Peusquens, S. Deyliz, A. Gollwitzer, R. Hertenberger, B. D. Valnion, and G. Graw. New 0^+ states in ^{158}Gd . *Physical Review C*, 66:051305, Nov 2002.
39. Mesytec. Mesytec MADC-32.
<https://www.mesytec.com/products/nuclear-physics/MADC-32.html>, .
40. Mesytec. Mesytec MPD-4.
<https://www.mesytec.com/products/nuclear-physics/MPD-4.html>, .
41. Mesytec. Mesytec MDPP-16.
<https://www.mesytec.com/products/nuclear-physics/MDPP-16.html>, .
42. Mesytec. mvme - Mesytec VME data acquisition.
<https://www.mesytec.com/downloads/mvme.html>, .
43. M. P. Metlay. *Development Of A Multiple-Element Conversion-Electron Spectrometer Array And Investigation Of Transition Multipolarities In ^{130}Ce* . PhD thesis, University of Pittsburgh, May 1992.

44. M. P. Metlay, J. X. Saladin, I. Y. Lee, and O. Dietzschi. The ICEBall: a multiple element array for in-beam internal conversion electron spectroscopy. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 336(1):162–170, November 1993.
45. D. A. Meyer, V. Wood, R. F. Casten, C. R. Fitzpatrick, G. Graw, D. Bucurescu, J. Jolie, P. von Brentano, R. Hertenberger, H.-F. Wirth, N. Braun, T. Faestermann, S. Heinze, J. L. Jerke, R. Krücken, M. Mahgoub, O. Möller, D. Mücher, and C. Scholl. Enhanced density of low-lying 0^+ states: A corroboration of shape phase transitional behavior. *Physics Letters B*, 638(1):44 – 49, 2006.
46. National Electrostatics Corporation. Pelletron Charging System. <http://www.pelletron.com/products/pelletron-charging-chains/>.
47. National Superconducting Cyclotron Laboratory. NSCL Data Acquisition Documentation. <http://docs.nscl.msu.edu/daq/>.
48. R. K. Osborn and M. E. Rose. $1-3\gamma$ correlation with mixtures and polarization. Technical Report 1560, Oak Ridge National Laboratory, June 1953.
49. C. Prokop, S. Liddick, B. Abromeit, A. Chemey, N. Larson, S. Suchyta, and J. Tompkins. Digital data acquisition system implementation at the National Superconducting Cyclotron Laboratory. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 741:163 – 168, 2014. ISSN 0168-9002. doi: <https://doi.org/10.1016/j.nima.2013.12.044>. URL <http://www.sciencedirect.com/science/article/pii/S0168900213017488>.
50. D. C. Radford. Notes on the use of the program gf3. <http://radware.phy.ornl.gov/gf3/>, May 2000.
51. C. Reich. Nuclear data sheets for $a = 154$. *Nuclear Data Sheets*, 110(10):2257 – 2532, 2009. ISSN 0090-3752. doi: <https://doi.org/10.1016/j.nds.2009.09.001>. URL <http://www.sciencedirect.com/science/article/pii/S0090375209000805>.
52. C. Reich. Nuclear data sheets for $a = 156$. *Nuclear Data Sheets*, 113(11):2537 – 2840, 2012. ISSN 0090-3752. doi: <https://doi.org/10.1016/j.nds.2012.10.003>. URL <http://www.sciencedirect.com/science/article/pii/S0090375212000798>.
53. C. S. Reingold, O. Olivas-Gomez, A. Simon, J. Arroyo, M. Chamberlain, J. Wurzer, A. Spyrou, F. Naqvi, A. C. Dombos, A. Palmisano, T. Anderson, A. M. Clark, B. Frentz, M. R. Hall, S. L. Henderson, S. Moylan, D. Robertson, M. Skulski, E. Stech, S. Y. Strauss, W. P. Tan, and B. Vande Kolk. High

- efficiency total absorption spectrometer hector for capture reaction measurements. *The European Physical Journal A*, 55(77), 2019. doi: 10.1140/epja/i2019-12748-8.
54. M. E. Rose. Supplementary remarks on angular correlation. Technical Report 1555, Oak Ridge National Laboratory, June 1953.
 55. M. E. Rose, G. H. Goertzel, B. I. Spinrad, J. Harr, and P. Strong. The internal conversion coefficients. i. the k -shell. *Physical Review*, 83(1):79–87, July 1951.
 56. M. E. Rose, L. C. Biedenharn, and G. B. Arfken. Internal conversion angular correlations. *Physical Review*, 85(1):5–16, January 1952.
 57. R. R. Roy and B. P. Nigam. *Nuclear Physics Theory and Experiment*. John Wiley & Sons Inc., 1st edition, 1967.
 58. E. Segré. *Nuclei and Particles*. Basic Books, 2nd edition, 1977.
 59. K. Smith. evt2root Package. <https://github.com/ksmith0/evt2root/>, March 2014.
 60. A. M. J. Spits, P. H. M. V. Assche, H. G. Börner, W. F. D. K. Schreckenbach, G. G. Colvin, R. C. Greenwood, C. W. Reich, P. O. Lipas, J. Suhonen, P. Sinkko, and A. Bäcklin. Spectroscopy of the ^{153}Gd and ^{154}Gd Isotopes, 1996. in BLG-703, edited by A. M. J. Spits and A. Bäcklin.
 61. D. Tonev, A. Dewald, T. Klug, P. Petkov, J. Jolie, A. Fitzler, O. Möller, S. Heinze, P. von Brentano, and R. F. Casten. Transition probabilities in ^{154}Gd : Evidence for X(5) critical point symmetry. *Phys. Rev. C*, 69:034334, Mar 2004. doi: 10.1103/PhysRevC.69.034334. URL <https://link.aps.org/doi/10.1103/PhysRevC.69.034334>.
 62. W. H. Trzaska. Recommended data on selected gamma-ray and conversion-electron calibration sources. *Nuclear Instruments and Methods in Physics Research A*, 297(2):223–229, 1990.
 63. S. S. M. Wong. *Introductory Nuclear Physics*. Prentice-Hall, Inc., 1st edition, 1990.
 64. J. L. Wood, E. F. Zganjar, C. D. Coster, and K. Heyde. Electric monopole transitions from low energy excitations in nuclei. *Nuclear Physics A*, (651): 323–368, 1999.
 65. X. Wu. *Investigation of Multiphonon Vibrational States in Deformed Nuclei*. PhD thesis, University of Notre Dame, December 1993.
 66. Xia. Pixie-16, 16-channel pxi digital pulse processor. https://www.xia.com/DGF_Pixie-16.html, 2016.

67. A. Zangwill. *Modern Electrodynamics*. Cambridge University Press, 1st edition, 2013.
68. V. Zerkin and B. Pritychenko. The experimental nuclear reaction data (EXFOR): Extended computer database and Web retrieval system. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 888:31 – 43, 2018. ISSN 0168-9002. doi: <https://doi.org/10.1016/j.nima.2018.01.045>. URL <http://www.sciencedirect.com/science/article/pii/S0168900218300627>.
69. J. F. Ziegler, M. Ziegler, and J. Biersack. SRIM - The stopping and range of ions in matter (2010). *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 268(11):1818 – 1823, 2010. ISSN 0168-583X. doi: <https://doi.org/10.1016/j.nimb.2010.02.091>. URL <http://www.sciencedirect.com/science/article/pii/S0168583X10001862>. 19th International Conference on Ion Beam Analysis.

*This document was prepared & typeset with pdfL^AT_EX, and formatted with
NDDiss2_ε classfile (v3.2017.2[2017/05/09])*