ACarbone_GRS_Report

April 7, 2021

1 Lab 4: Gamma Ray Spectroscopy

PHYS3112 - Experimental and Computational Physics

Anthony Carbone - z5260477

Report Due - Wed 7/4/2021

1.1 Aim

To observe and explain some characteristics of a gamma ray spectroscopy setup that uses a thallium doped sodium iodide scintillator as the detector.

1.2 Introduction

There are in this context three ways gamma rays interact with matter: 1. the photoelectric effect, 2. Compton scattering, and 3. pair production.

The emitted gamma rays from the sources used in this experiment are too low for pair production, and the photoelectric effect occurs without detection as all of the gamma photon's energy is transferred to a tightly bound electron, which most likely be brought to rest inside the NaI crystal. [1]

However it is Compton scattering that is detected, and this is the primary behaviour we will observe. This is because the gamma photon is scattered to a lower energy by colliding with a 'nearly free' NaI electron, transferring a small amount of energy in doing so. As this electron de-excitates, it releases a energy proportional photon which is then detected by the PMT, thus inferring the inital gamma ray energy transfer. [1]

2 Experiments

2.1 Pulse Shapes

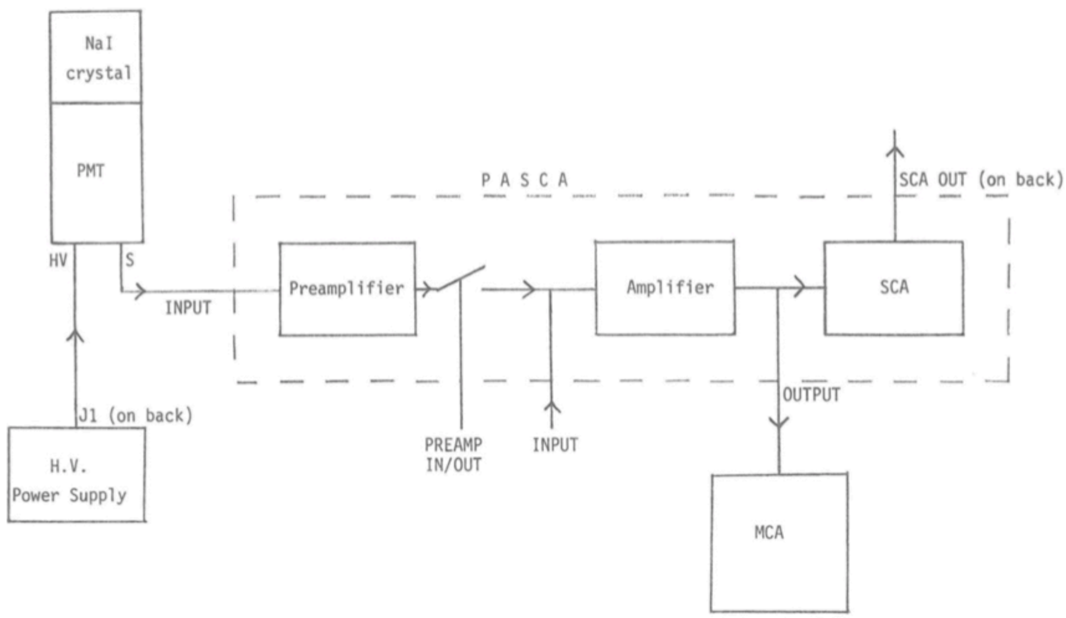


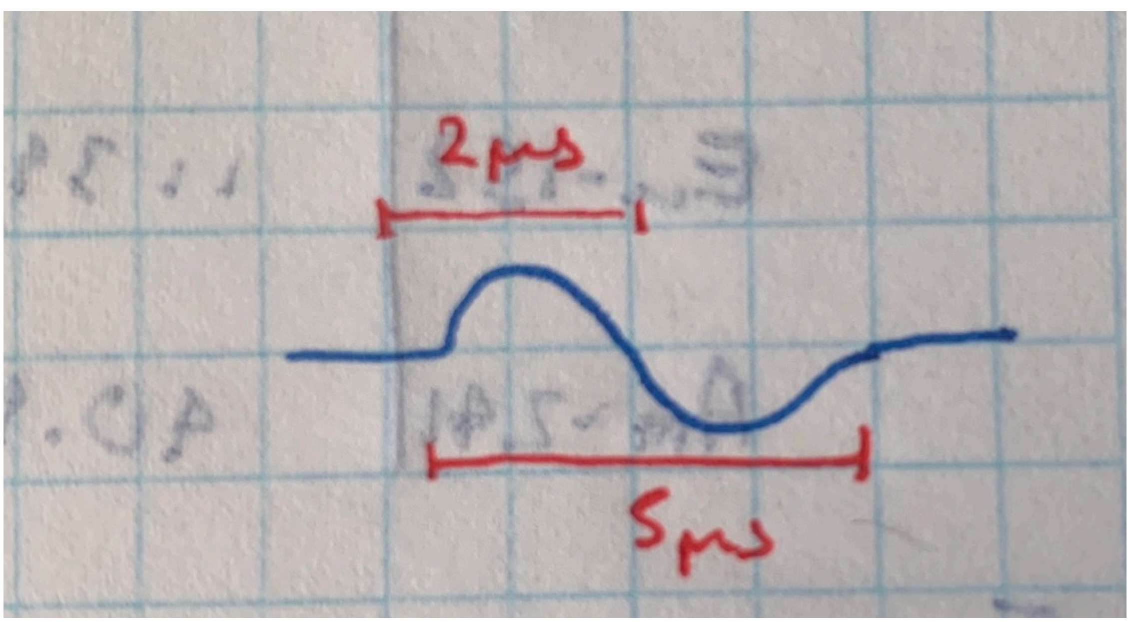
Figure 4: Block diagram of electronics.

[1]

This experiment compared the signal emerging from the preamp to the amp. This was achieved by setting the spectroscope to trigger on every count and observing the average peak voltage V, baseline voltage $V_{b}, and decay time t_D.

For the preamp - $V = 1.60$mV, - $V_b = -15.6$mV, and - $t_D = 60\mu$s,

while for the amp - $V = 2.96$V, - $V_b = 0.00$V, and - t_D had a shape as in the following figure.



This highlights the crucial role the amp performs, which is reducing the decay time substantially, permitting essentially 1 count per 5mus, as opposed to the preamp capacity of 1 count per 60mus. This enables far greater capacity for count rate, which would otherwise have caused overlapping of the signals. This ultimately results in a 'cleaner', more workable signal.

2.2 Cs-137 Spectrum

In this section I analyse the spectra of the Cs-137 source.

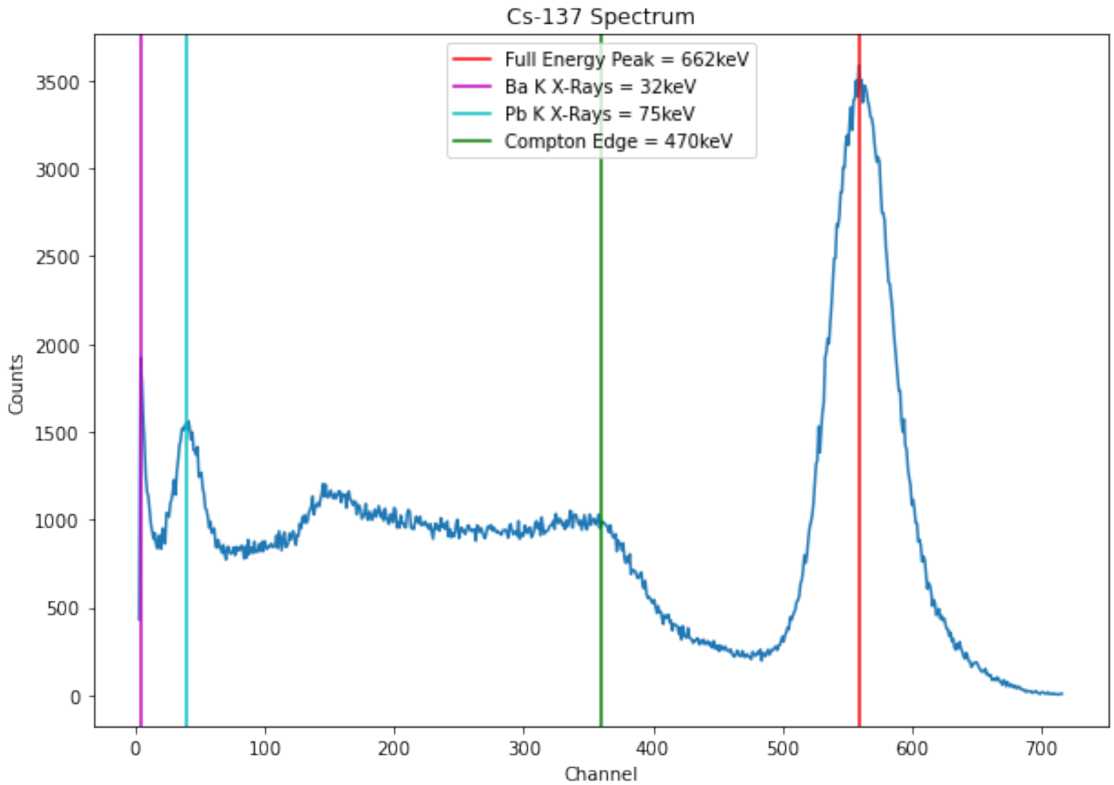
[1]:
```
# Imports
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
```

[2]:
```
# Function to import the awfully formatted spectrum data
def read_spectrum(fname, skiprows=0):
    raw_spectrum = pd.read_csv(fname, names=[0, 1, 2, 3, 4, 5, 6, 7],␣
 ↪skipfooter=1, skiprows=skiprows, delim_whitespace=True)
    raw_spectrum[0] = raw_spectrum[0].str.replace(':', ' ').astype(int)
    chn = np.arange(raw_spectrum[0][0],raw_spectrum[0].iloc[-1]+7)
    counts = np.array([])
    for row in range(0, len(raw_spectrum)):
        for col in range(1, 8):
            counts = np.append(counts, raw_spectrum[col][row])
    return chn, counts
```

[3]:
```
chn_Cs137, counts_Cs137 = read_spectrum('Data/1_Cs137.txt')

plt.figure(figsize=(10,7))
plt.plot(chn_Cs137, counts_Cs137)
plt.title('Cs-137 Spectrum')
plt.xlabel('Channel')
plt.ylabel('Counts')

# Overlay notable features, all recorded in my lab book
full_energy_Cs137 = 559
plt.axvline(x=full_energy_Cs137, label='Full Energy Peak = 662keV', color='r')
Ba_k_Cs137 = 5
plt.axvline(x=Ba_k_Cs137, label='Ba K X-Rays = 32keV', color='m')
Pb_k_Cs137 = 40
plt.axvline(x=Pb_k_Cs137, label='Pb K X-Rays = 75keV', color='c')
compton_edge_Cs137 = 360
plt.axvline(x=compton_edge_Cs137, label='Compton Edge = 470keV', color='g')
plt.legend()

print(f'Notable Channels: Full energy = {full_energy_Cs137}, Ba K X-rays =␣
 ↪{Ba_k_Cs137}, Compton edge = {compton_edge_Cs137}')
```

```
<ipython-input-2-c1405c006cda>:3: ParserWarning: Falling back to the 'python'
engine because the 'c' engine does not support skipfooter; you can avoid this
warning by specifying engine='python'.
  raw_spectrum = pd.read_csv(fname, names=[0, 1, 2, 3, 4, 5, 6, 7],
skipfooter=1, skiprows=skiprows, delim_whitespace=True)
```

```
Notable Channels: Full energy = 559, Ba K X-rays = 5, Compton edge = 360
```



Photoelectric interactions occur with the Pb shielding causing characteristic x-rays to be emitted, these are indicated in the above figure in cyan.

Ba x-rays are also detected as they are produced in the decay of Cs-137.

Compton scattering occurs in the detector and the excited electron emits a photon equal to the energy transferred from the gamma ray during collision. Thus, this photon energy is greatest when the Compton angle $\theta = 180°$. This forms the Compton edge, indicated in the above figure with the green line. This backscatter peak continues for all angles and is present as a continuum to the left of the Comptone edge.

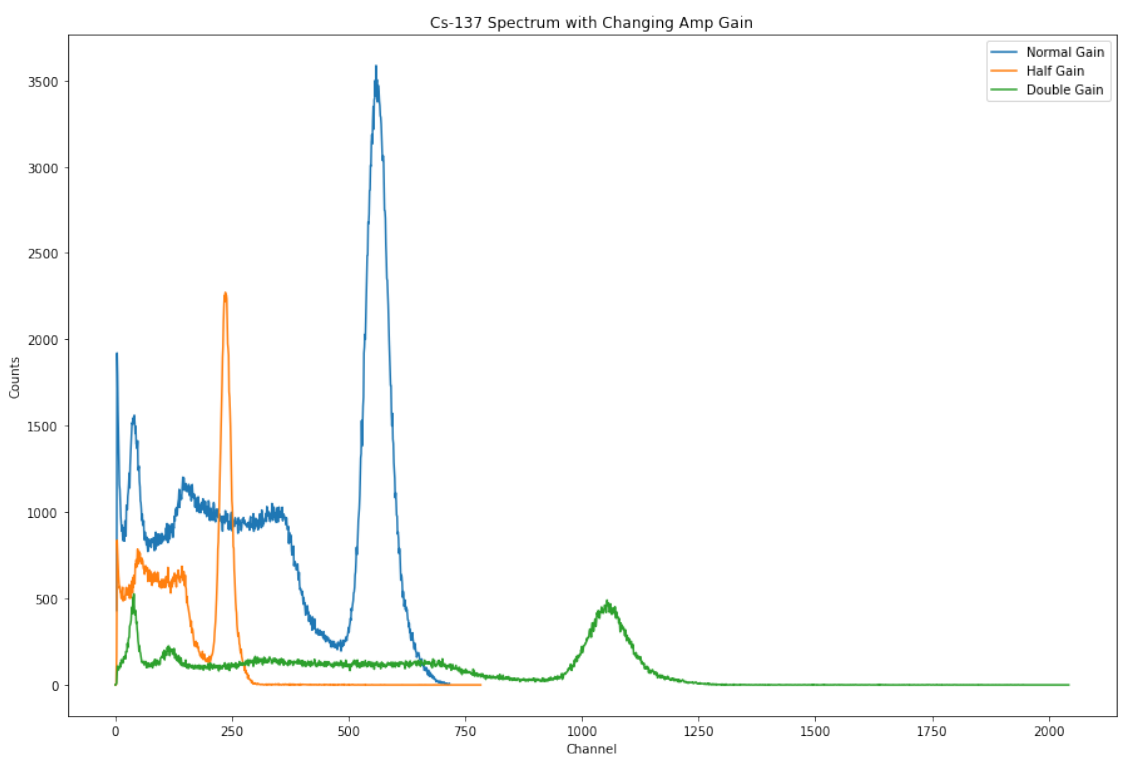
2.3 Amp Gain

Now we look at the effect on the spectrum of altering the amp gain.

[4]:
```
chn_halfGain_Cs137, counts_halfGain_Cs137 = read_spectrum('Data/
↪1_Cs137_halfCoarseGain.txt', skiprows=4)
chn_doubleGain_Cs137, counts_doubleGain_Cs137 = read_spectrum('Data/
↪1_Cs137_doubleCoarseGain.txt', skiprows=4)

plt.figure(figsize=(15,10))
plt.plot(chn_Cs137, counts_Cs137, label='Normal Gain')
plt.plot(chn_halfGain_Cs137, counts_halfGain_Cs137, label='Half Gain')
plt.plot(chn_doubleGain_Cs137, counts_doubleGain_Cs137, label='Double Gain')
plt.title('Cs-137 Spectrum with Changing Amp Gain')
plt.xlabel('Channel')
plt.ylabel('Counts')
plt.legend()
```

```
<ipython-input-2-c1405c006cda>:3: ParserWarning: Falling back to the 'python'
engine because the 'c' engine does not support skipfooter; you can avoid this
warning by specifying engine='python'.
  raw_spectrum = pd.read_csv(fname, names=[0, 1, 2, 3, 4, 5, 6, 7],
skipfooter=1, skiprows=skiprows, delim_whitespace=True)
```

[4]:
```
<matplotlib.legend.Legend at 0x7fecf8893640>
```



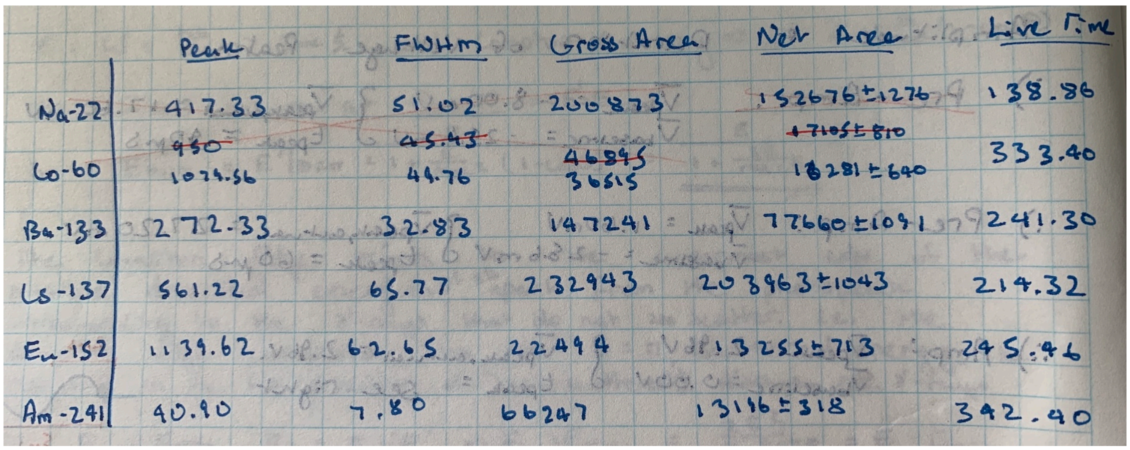
It can be observed from the above figure that, as expected, decreasing the amp gain essentially scales the spectrum horizontally. This corresponds to the voltage entering the MCA to be lower, and thus register in lower channels.

What is unexpected is that by doubling the amp gain, the spectrum scales horizontally in the same direction as that in the half gain case, by an even greater amount. I suspect what is occuring is the MCA driver changes modes recognising the lower significance of the lower voltages, and expands its channel set.

2.4 All Spectrums

Next we look at the spectrums for Co-60, Ba-133, Cs-137, Eu-152, and Am-241.

First, here is the recorded data from the spectra of the sources.

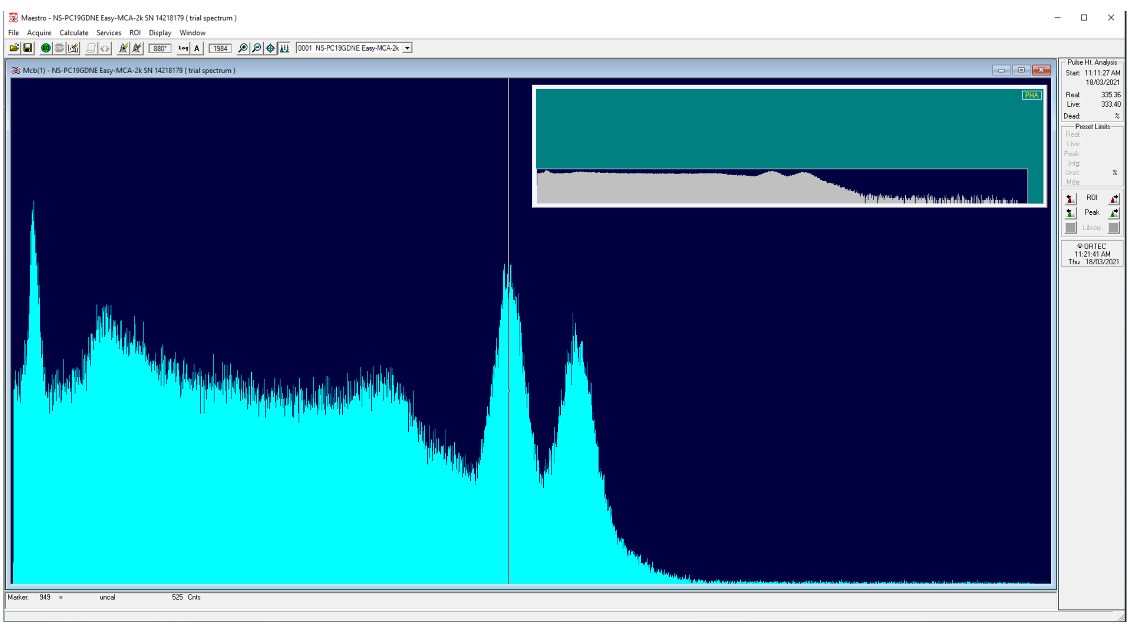


	Peak	FWHM	Gross Area	Net Area	Live Time
Na-22	417.33	51.02	200873	152676±1276	138.86
Co-60	~~450~~ 1079.56	~~45.43~~ 44.76	~~46845~~ 36515	~~47105±810~~ 18281±640	333.40
Ba-133	272.33	32.83	147241	77660±1091	241.30
Cs-137	561.22	65.77	232943	203963±1043	214.32
Eu-152	1139.62	62.65	22494	13255±713	245.46
Am-241	40.90	7.80	66247	13196±318	342.40

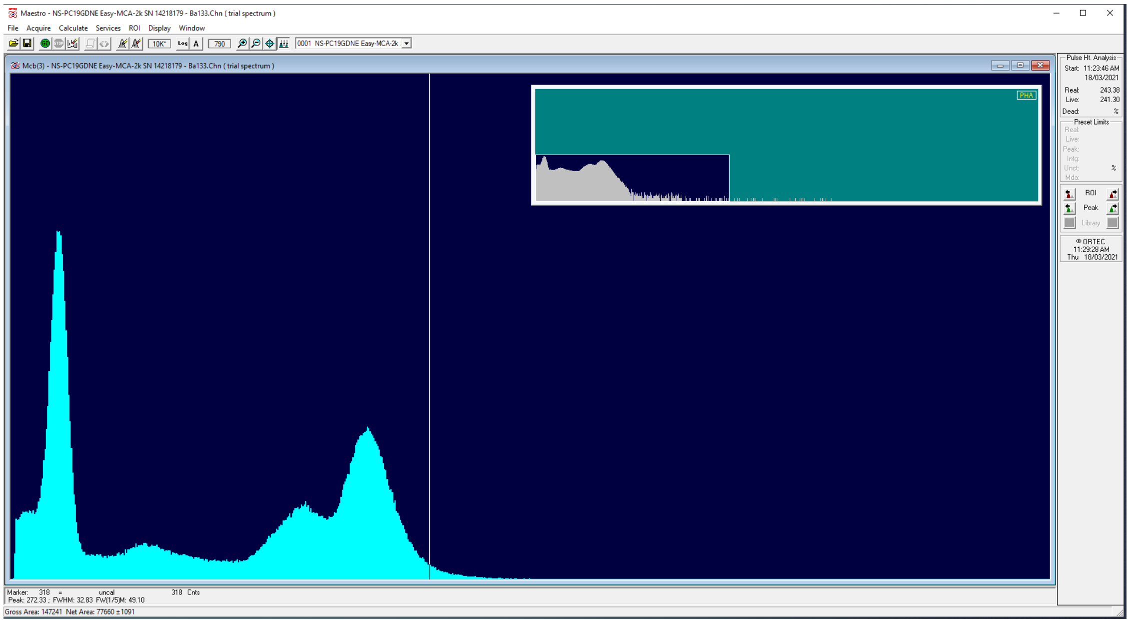
Results for each spectrum

Following is the spectrum plot for each source, with the peak highlighted. The channel can be seen in the bottom right.

Note the Compton continuum in each of the plots.

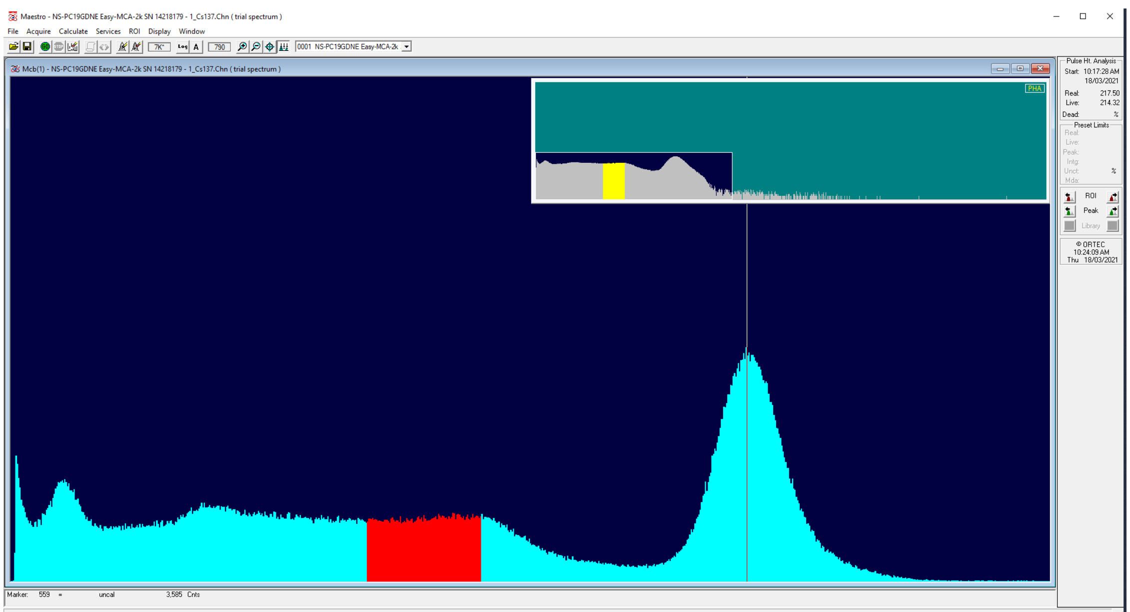


Co-60 Spectrum



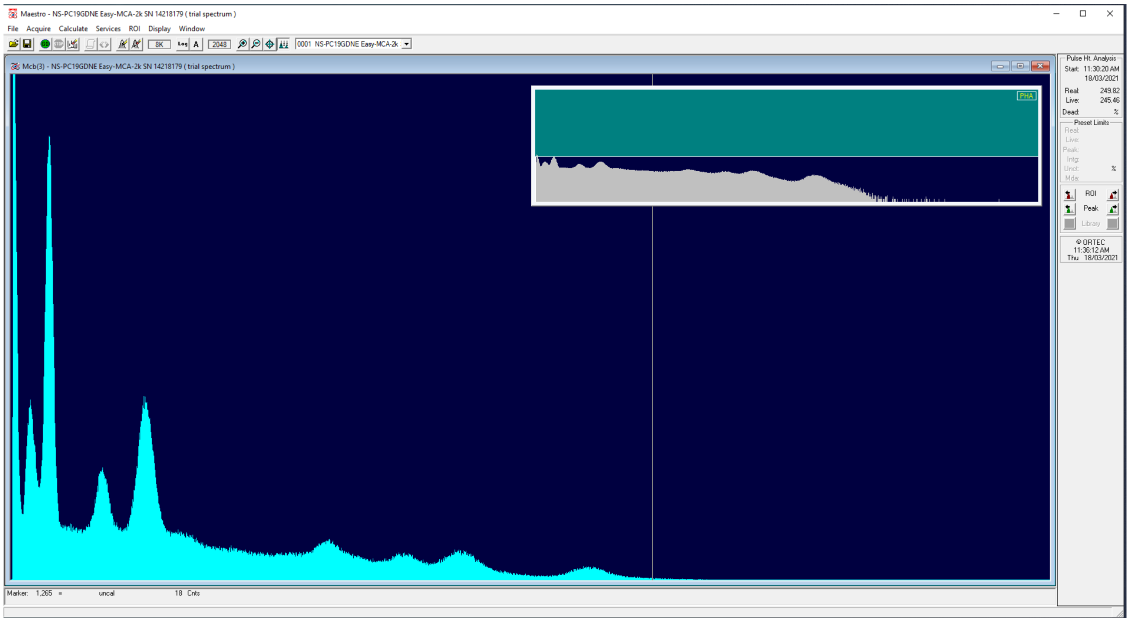
Ba-133

Spectrum

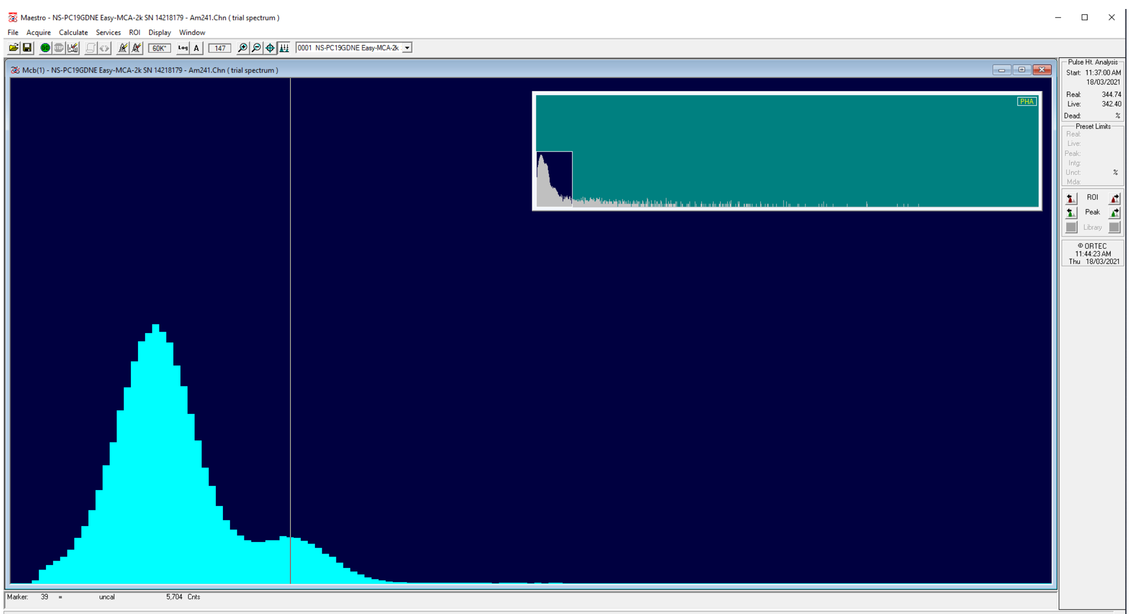


Cs-137

Spectrum



Eu-152 Spectrum



Am-241 Spectrum

2.5 Energy Calibration

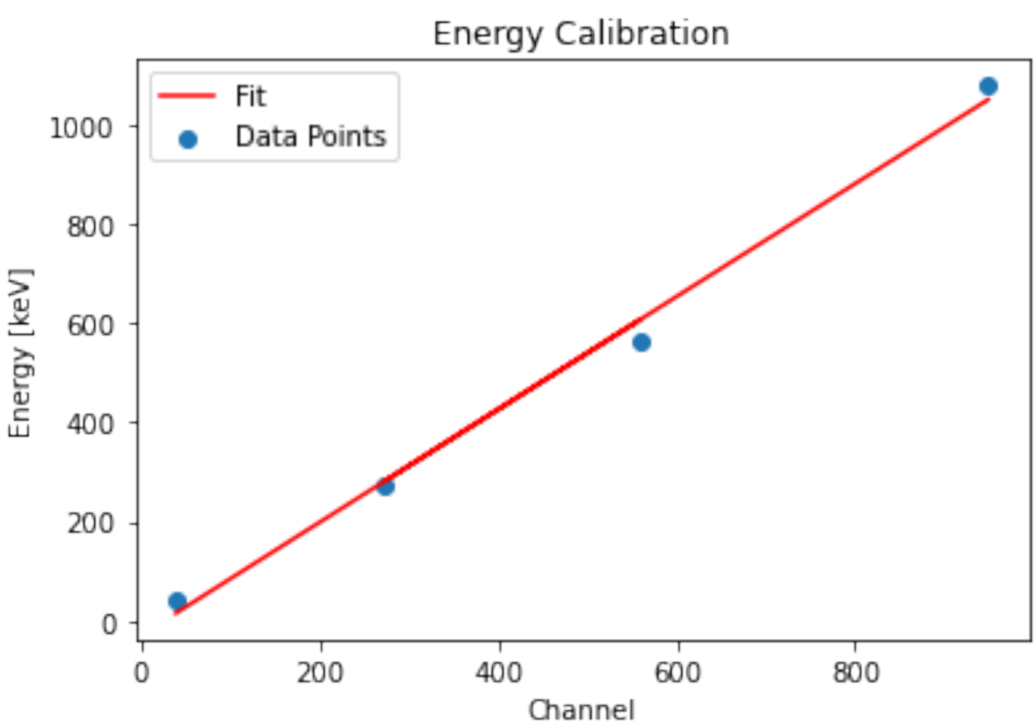
I obtained an energy calibration on the channel number by curve fitting source energy vs channel. This of course assumes a linear relationship between channel and energy.

Note, I unfortunately lost the Na-22 and Eu-152 data, but the remaining data was sufficient to perform the calibration calculation.

[5]:
```
source_df = pd.DataFrame({'Channel': [949, 272.33, 559,39],
                          'Energy': [1079.56, 272.33, 561.22, 40.90]
```

```
            })

source_df.index = ['Co60','Ba133','Cs137', 'Am241']

source_df.index.name = 'Source'
source_df.reset_index(inplace=True)

def line(x, a, b):
    return a*x+b

params, cov = curve_fit(line, source_df['Channel'], source_df['Energy'])

plt.scatter(source_df['Channel'], source_df['Energy'], label='Data Points')
plt.plot(source_df['Channel'], line(source_df['Channel'], *params), color='r',␣
 ↪label='Fit')
plt.title('Energy Calibration')
plt.xlabel('Channel')
plt.ylabel('Energy [keV]')
plt.legend()

def calibrate(channel):
    return line(channel, *params)

print(f'True Energy = {params[0]:.2f} * [Channel] - {-params[1]:.2f}')
```

```
True Energy = 1.14 * [Channel] - 29.40
```



Thus, the 'true energy' E can be calculated using $E = 1.14C - 29.40$, where C is the channel number.

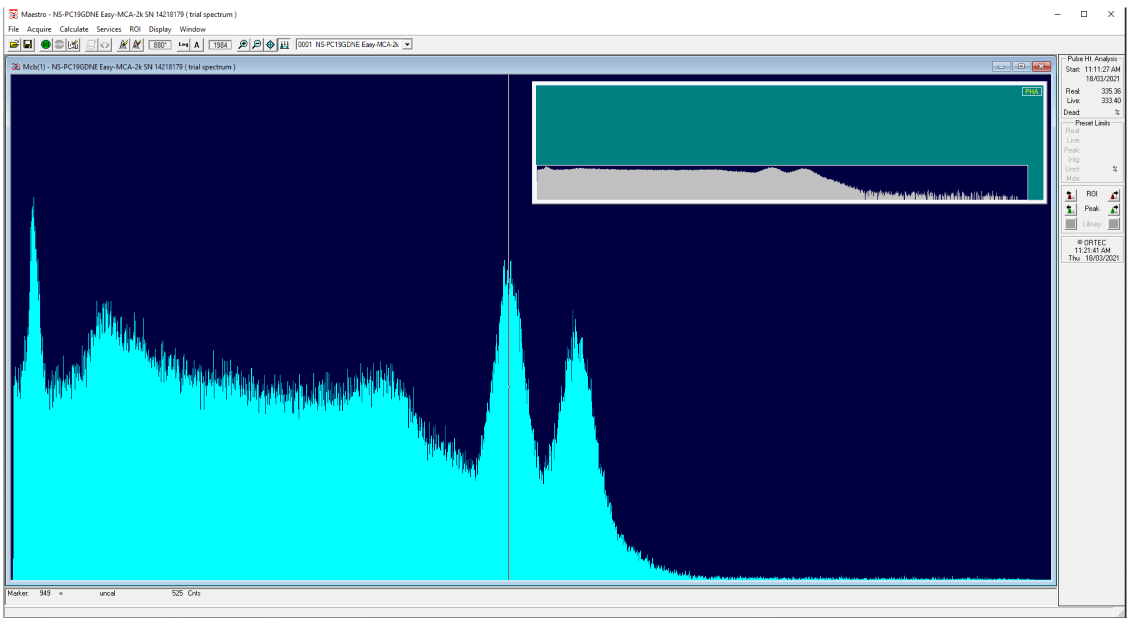
2.6 Energies of Co-60 Gamma Rays

[6]:
```
print(f'Calibrated Co-60 (lower) photopeak = {calibrate(949):.2f} keV')
```

```
Calibrated Co-60 (lower) photopeak = 1051.19 keV
```

This is somewhat reasonable given the theoretical photopeak occurs at 1079.56. This is most likely due to the linear fit lacking enough data to calculate the calibration effectively (due to losing two of the data points).

Note with the Co-60 spectrum, the two photopeaks are expected to have the same intensity, but are at different heights. This is due to the lower of the two photopeaks being counted in addition to the Compton phenomena of the higher photopeak, resulting in more counts.

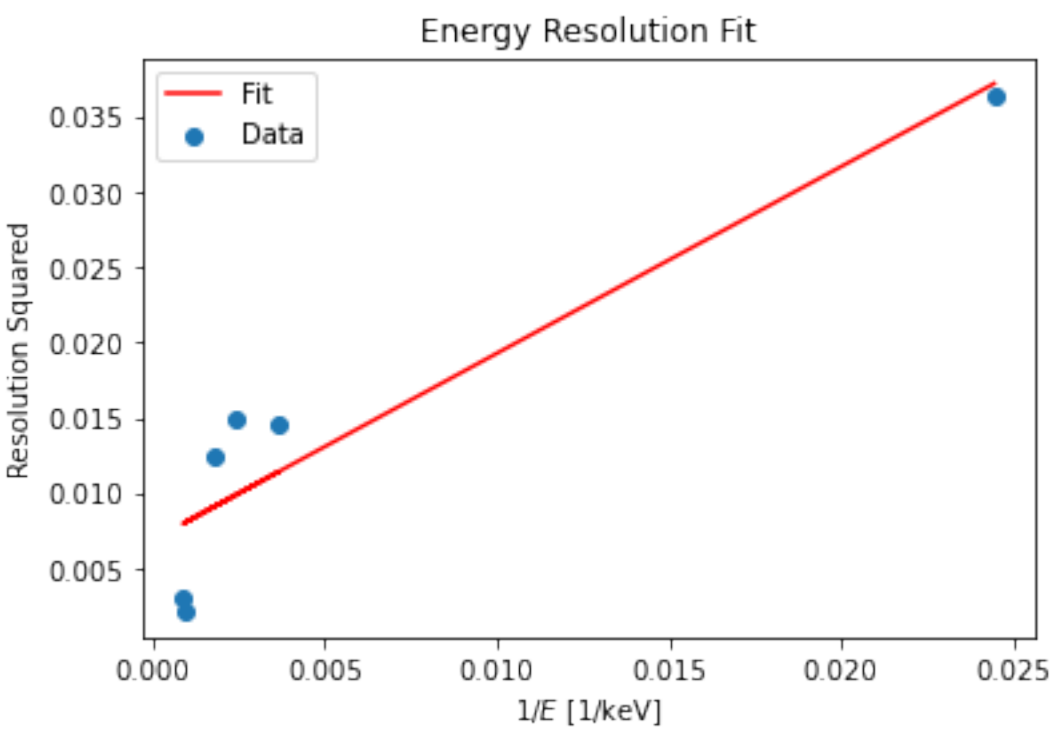


2.7 Energy Resolution

[7]:
```
# Add back the two datapoints, and add FWHM
source_df = pd.DataFrame({'Channel': [0, 949, 272.33, 0, 559, 39],
                          'Energy': [417.33, 1079.56, 272.33, 561.22, 1139.62,␣
 ↪40.90],
                          'FWHM': [51.02, 49.76, 32.83, 62.77, 62.65, 7.8]
          })

source_df.index = ['Na22', 'Co60','Ba133','Cs137', 'Eu152', 'Am241']

source_df['Resolution'] = source_df['FWHM']/source_df['Energy']
```

```
resolution_fit, resolution_cov = curve_fit(line, 1/source_df['Energy'],␣
↪source_df['Resolution']**2)
# B = resolution_fit[0]
# w_l = np.sqrt(resolution_fit[1])

plt.scatter(1/source_df['Energy'], source_df['Resolution']**2, label='Data')
plt.plot(1/source_df['Energy'], line(1/source_df['Energy'], *resolution_fit),␣
↪label='Fit', color='r')
plt.title('Energy Resolution Fit')
plt.xlabel('1/$E$ [1/keV]')
plt.ylabel('Resolution Squared')
plt.legend()
```

[7]: <matplotlib.legend.Legend at 0x7fecf90c7640>



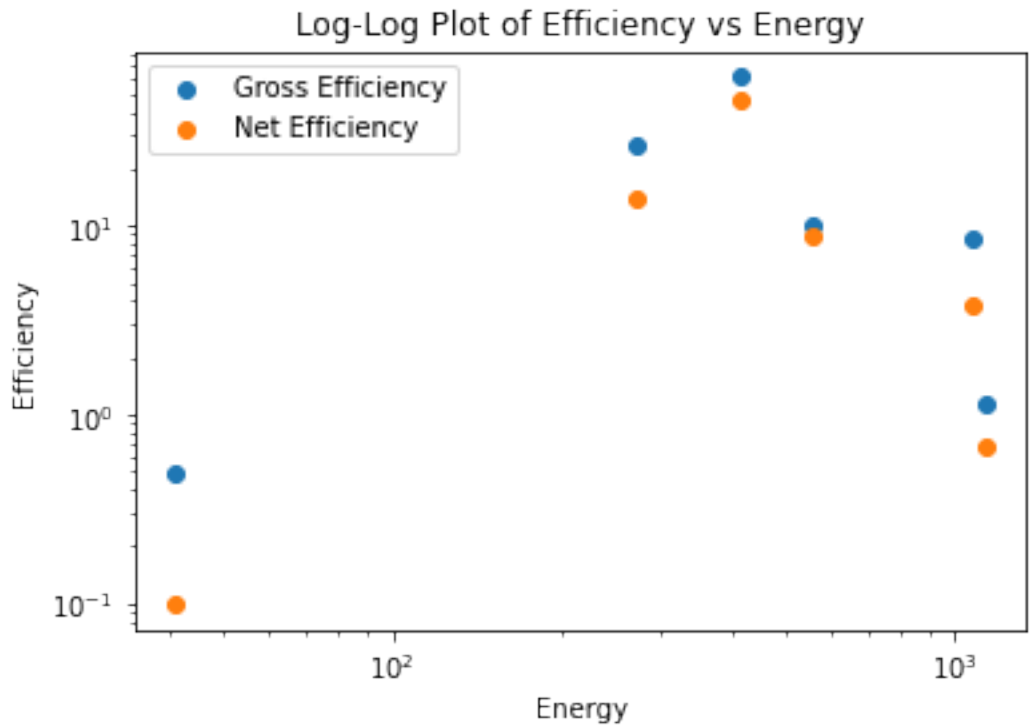
As can be seen, the fit is by no means conclusive. It may highlight that one might also have to calibrate the FWHM as well to obtain a reasonable fit.

2.8 Relative Full Energy Peak Efficiency

This section looks to calculate the net and gross efficiency of the detector using the reference activity and date.

[47]:
```
source_df['Gross Counts'] = [200873, 36515, 147241, 232943, 22494, 66247]
source_df['Net Counts'] = [152676, 16281, 77660, 203963, 13255, 13196]
source_df['Live Time'] = [138.86, 333.40, 241.30, 214.32, 245.46, 342.4]

source_df['Gross Count Rate'] = source_df['Gross Counts']/source_df['Live Time']
source_df['Net Count Rate'] = source_df['Net Counts']/source_df['Live Time']

source_df['Reference Date'] = ['1/8/2014', '1/2/2003', '25/2/1994', '5/5/1994',␣
↪'1/3/1998', '1/3/1999']
source_df['Reference Date'] = pd.to_datetime(source_df['Reference Date'],␣
↪dayfirst=True)
source_df['Reference Activity'] = [296, 400, 303, 266, 441, 411]

source_df['Half-Life'] = [2.62, 5.27, 10.52, 30.00, 13.52, 432.0] # Years

collection_day = pd.to_datetime('18/3/2021', dayfirst=True)

source_df['Years'] = (collection_day - source_df['Reference Date']).dt.days/365

source_df['Activity'] = source_df['Reference Activity'] * np.
↪exp(-source_df['Years']/source_df['Half-Life'])

source_df['Gross Efficiency'] = source_df['Gross Count Rate']/
↪source_df['Activity']
source_df['Net Efficiency'] = source_df['Net Count Rate']/source_df['Activity']

plt.scatter(source_df['Energy'], source_df['Gross Efficiency'], label='Gross␣
↪Efficiency')
plt.scatter(source_df['Energy'], source_df['Net Efficiency'], label='Net␣
↪Efficiency')
plt.xlabel('Energy')
plt.ylabel('Efficiency')
plt.yscale('log')
plt.xscale('log')
plt.legend()
plt.title('Log-Log Plot of Efficiency vs Energy')
```

[47]:
```
Text(0.5, 1.0, 'Log-Log Plot of Efficiency vs Energy')
```



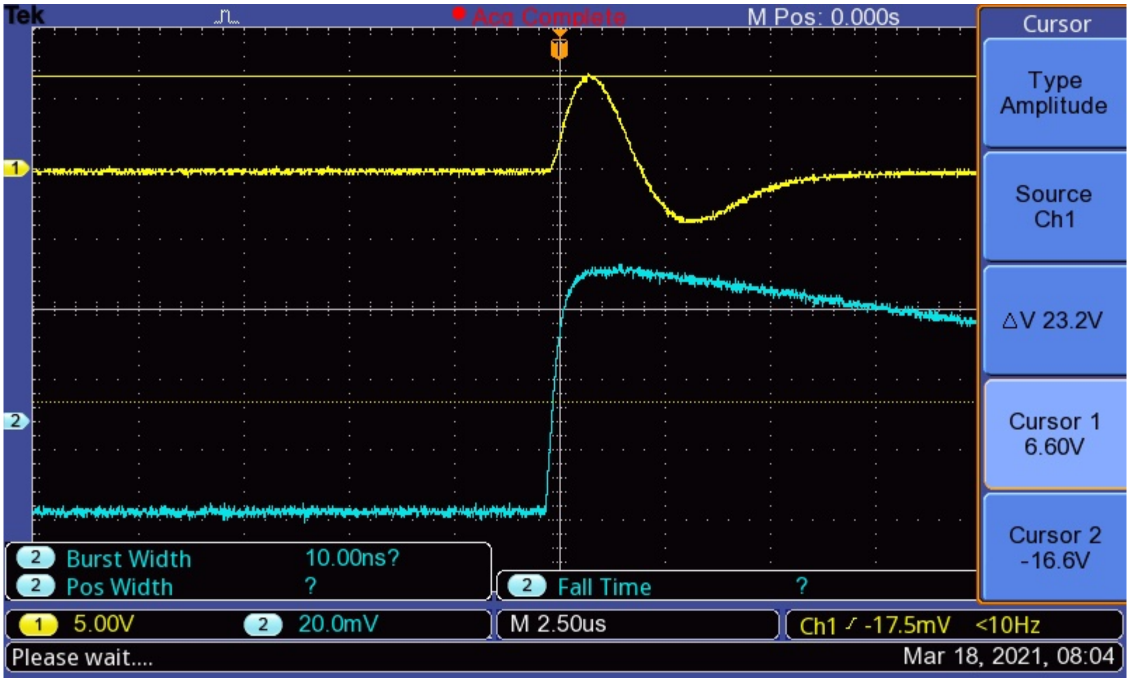
There seems to be little agreeance between the above figure and Figure 1 in the student notes, a plot of the absorbtion coefficient. I cannot recognise any obvious errors in my calculations nor data so it may benefit from further investigation.

2.9 Changing the PMT's Voltage

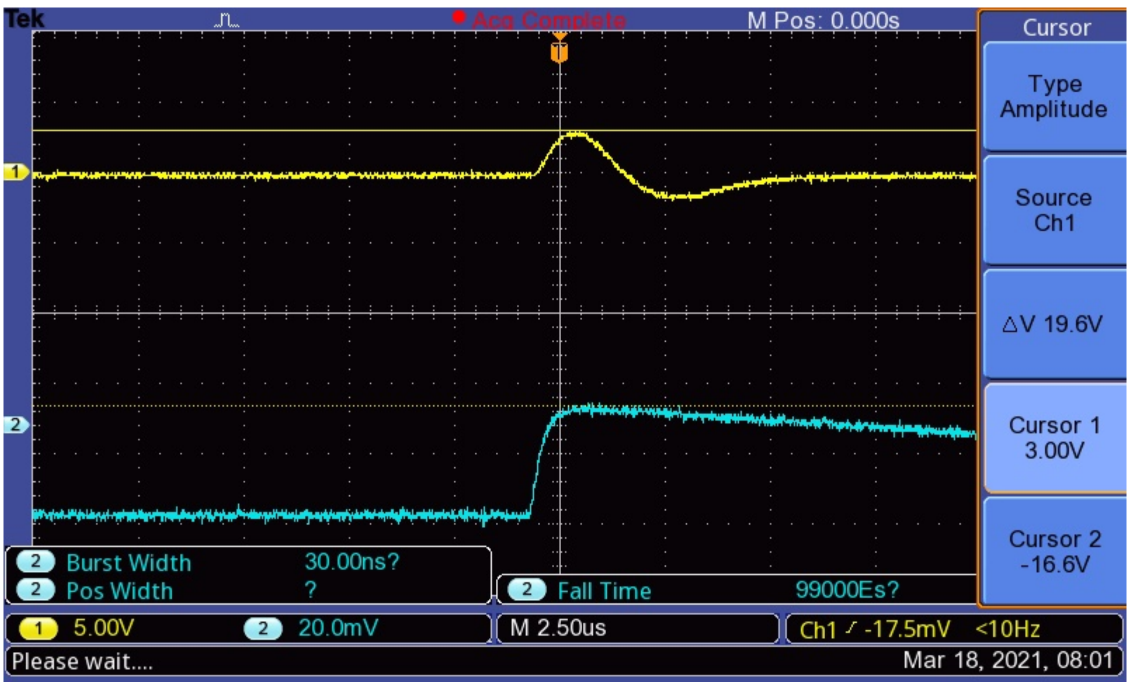
Following are screen captures from the spectroscope of the Cs-137 source with differing PMT voltages of 800V, 1000V, and 1100V. Note the lower blue line is the preamp signal, and the yellow line the amp signal.

What might be expected is the amp signal voltage increases as the PMT voltage increases. Why this is not observed may be due to these observations being made on single, and different, counts. This means each count inherently has a different energy, despite my best efforts to capture seemingly 'average' profiles.

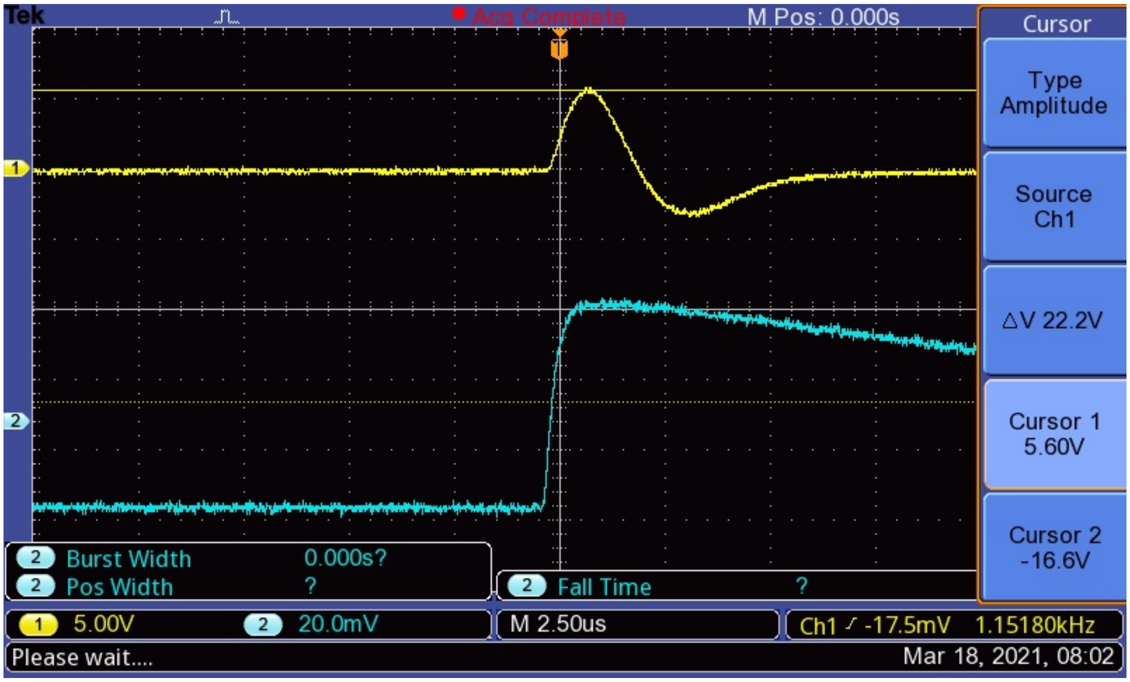
This may benefit from further investigation.



PMT set at 800V



PMT set at 1000V



PMT set at 1100V

3 References

[1] Gamma Ray Spectrometry Student Notes - UNSW School of Physics

Gamma Ray Spectroscopy

TIWKS Thu 2021 [illegible]/02/21

Aim: Study characteristics of gamma-ray spectrometry system, ~~using~~ a thallium doped NaI scintillator.

Notes:

- 3 ways ~~ga~~ γ rays ~~to~~ interact with matter:
  1) photoelectric effect
  2) Compton scattering
  3) pair production

LHK Q1) min energy of scattered γ ray with incident energy E in collision with a stationary e^- of mass m_e

~~$E \propto \frac{1}{\lambda}$ and~~

$E = hf = \frac{hc}{\lambda} \Rightarrow E \propto \frac{1}{\lambda}$ and Compton scattering $\Rightarrow \lambda' = \lambda + \frac{h}{m_e c}(1-\cos\theta)$

so λ'_{max} occurs at $\underline{\theta = \pi} \Rightarrow E' = E_{min}$

so $E_{min} = E'|_{\theta=\pi} = \frac{E}{1+\frac{E}{m_e c^2}(1-\cos\theta)} = \frac{E}{1+\frac{2E}{m_e c^2}}$

LHK Q2) The Compton Edge refers to the uppermost edge of the possible detected energies of e^-'s apart from the photopeak corresponding to the γ-rays that do not ~~so~~ scatter. i.e. the Compton edge corresponds to the $\theta = \pi$, scattered energy, ~~to~~ resulting in the full detection of 'back-scattered' ~~γ-rays~~ ~~et~~. γ-rays

ie. ~~$E_c = E_{min}$~~ $E_T = E - E' \Rightarrow E_c = E_{T,max} = \frac{2E^2}{m_e c^2 + 2E} = E - E_{min}$

$$E - \frac{E}{1+\frac{2E}{m_e c^2}} = E\left[\frac{\cancel{1} + \frac{2E}{m_e c^2} - \cancel{1}}{1+\frac{2E}{m_e c^2}}\right] = E\left(\frac{2E}{m_e c^2 + 2E}\right) = \frac{2E^2}{m_e c^2 + 2E}.$$

Q3) PMT multiply ~~the~~ greatly the number of incident photons that enter the tube, resultining in noticeable currents at the output.

Q4) Don't know, why $\left(\frac{\Delta P}{P}\right)^2 = \frac{B}{E}$ only occurs in purely random statistical processes

Note: $\left(\frac{\Delta P}{P}\right)^2 = \frac{B}{E} + W_I^2$

FWHM; Peak; const. (B); Corr. E (E); intrinsic width (W_I)

Set-up:

- Power supply - 1000 V - +ve - 'LOCAL' (back)
- Amplifier - Fine gain = 7 - Coarse gain = 16 - input = pos - preamp = in

Amplifier output → MCA input; → oscilloscope CH1

- Check on radioactive source

- Capture -> Print -> Measure approx. amplitude & duration of typical V pulses a
  i) Preamp
  ii) Amp
  * adjust fine gain for a solid line of pulses ~ 3 V

- Start MAESTRO MCA
  - Destroy Calibration
  - Clear Detector / Buffer, vert-scale = 4K

* Acquire data to top of screen & SAVE

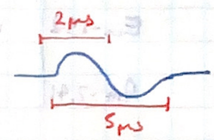
4.2.1 Pulse Shapes

Amplitude & Duration of Voltage Peaks

i) ~~Pre-Amp:~~ ~~$\bar{V}_{peak} = -8.00 mV$~~ ~~$V_{baseline} = -25.6 mV$~~ } ~~$\bar{V}_{peak,actual} \approx 17.60 mV$~~ ~~$t_{peak} \approx 40 \mu s$~~

ii) Pre-Amp: $\bar{V}_{peak} = 1.60 mV$, $\bar{V}_{baseline} = -25.6 mV$ } $\bar{V}_{peak,actual} = 27.20 mV$, $t_{peak} = 60 \mu s$

iii) Amp: $\bar{V}_{peak} = 2.96 V$, $V_{baseline} = 0.00 V$ } $\bar{V}_{peak,actual} = 2.96 V$. t_{peak} = see right



Q5) Pre-Amp: ~~±~~ ~ 1 count per 60μs

Amp: ~ 1 count per 5 μs } big difference!

4.2.2.2 Cs137 Spectrum Saved '1_Cs137.txt' & *.Chn'

i) Full energy (662eV) ≈ Chn 554

ii) Ba K x-rays ~ Chn 40 ~~72.87~~ eV 32.1936 eV 74.969 eV

iii) Pb K x-rays ~ ?

iv) Compton Edge ~ Chn 360

backscatter peak

Q6) Pb K x-rays - Pb shielding
Ba K x-rays - Ba container??

Q7) Backscatter Peaks - Lowest energy scattered??

Lower Gain - Location of peak + Lower peak

4.2.2.3 ROI & Peak Analysis

Cs137 - Normal Coarse Gain (16)

Gross Area / Gross Area Total (higher) = 231515 / 650317

4.3 Everything Spectrometery

Cs137

Peak = 561.22 = 662.37 keV
FWHM = 65.77
~~Ex~~ Gross Area = 232943
Net Area = 203963 ± 1043
Live time = 214.32

	Peak	FWHM	Gross Area	Net Area	Live Time
Na-22	417.33	51.02	200873	152676 ± 1276	138.86
				~~17105 ± 810~~	
Co-60	~~950~~ 1074.56	~~45.43~~ 44.76	~~46845~~ 36515	18281 ± 640	333.40
Ba-133	272.33	32.83	147241	77660 ± 1091	241.30
Cs-137	561.22	65.77	232943	203963 ± 1043	214.32
Eu-152	1139.62	62.65	22494	13255 ± 713	245.46
Am-241	40.90	7.80	66247	13136 ± 318	342.40

4.3.3 Energy Calibration

Q9) Too many datapoints, non-linear, hard to fit.

Q10) ?

Q11) __

4.3.4 Co-60 Energies

4.3.5 Energy Resolution