ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

IPA

Influence of Wrapping on the Light Output of BGO

Till Dieminger & Ronan Schwarz

Semester project in the group of Prof. G. Dissertori
Supervised by Dr. Francesca Nessi-Tedaldi

September 17, 2019

Abstract. We measured the effect of 14 different wrapping materials on the light output and energy resolution of a Bismuth Germanate scintillating crystal. Most of the wrappings increased the light output significantly with respect to the bare crystal. Using 3M Vikuiti™ ESR foil, we achieved almost triple the light output of the unwrapped crystal.

Contents

CERN-THESIS-2019-120
/ /2019

I Introduction

Scintillators are a vital part of todays detectors, used in many areas from particle physics to modern medicine. They are used to detect ionising radiation, since they transform a high energy photon into a shower of low energy photons, conveniently measurable by photomultiplier tubes or avalanche photo diodes. The main working principle of scintillators is the ionisation of atoms in the crystal. The resulting free electrons then are captured by so-called scintillation centers, emitting a photon [1].

Many treatments of cancer require a precise map of the tumor's position and size. This can be achieved using positron emission tomography. In this method, a radioactive tracer, often Fluorine-18, is introduced into the body. The tracer accumulates in the cancer cells and decays under emission of a positron, which annihilates with an electron from the surrounding tissue. This annihilation produces two back-to-back photons with a characteristic energy of 511 keV. Registering this radiation with two detectors gives information on the position of the annihilation and therefore also on that of the tumor. For this, an accurate and precise measurement of the energy, as well as the time delay of the incoming photons is needed. This is often done using scintillating crystals.

One of the commonly used scintillators for this application is Bismuth Germanate (BGO). Its light yield, i.e. the number of emitted photons per keV of the incoming radiation, is relatively high. However, the light yield cannot be fully exploited, since most of the emitted light escapes through the sides of the crystal, lowering the effective light output of the crystal towards the photo detector. To increase the light output with respect to the light yield, one can wrap the crystal in different materials to guide the light into the detector.

In our experiment, we used standard materials like paper or Tyvek® [2] for this purpose, but also some more exotic materials like Polytetrafluoroethylene (PTFE), or Vikuiti™ foil from 3M [3], which already is used for PET scanners [4]. We found that most materials lead to a significant increase of the light output. Of the materials we studied, Tyvek®, aluminised Mylar®, PTFE and Vikuiti™ delivered the best results. We used a Sodium-22 source for our measurements, which decays mainly through a β^+ transition, similar to the previously mentioned Fluorine-18, see appendix A.1.

Scintillation

A scintillator is a material in which the energy loss of an ionising radiation is emitted in the form of low energy photons, preferably in a wavelength range detectable by the PMT used. For this, the scintillator needs to contain luminescent centres, for example doping ions or lattice defects [5].

The incoming ionising radiation excites the electrons from the occupied valence band into the conduction band. For each of these interactions, an electron–hole pair is created. If the energy of the radiation is high enough to reach this ionisation threshold, we have free carriers which will move randomly in the crystal until they are absorbed by a defect or recombine on a luminescent centre. The photons resulting from these interactions need to have an energy in the band gap, to avoid reabsorption of the emitted light.

Bismuth germanate (BGO, $Bi_4Ge_3O_{12}$) is a crystal which exhibits scintillation properties. The emitted light has a wavelength of 300 – 600 nm with a maximum at 480 nm. Its decay time is approximately 350 ns. Its primary advantages are a high light yield of around 8500 photons per MeV and the high energy resolution. The light output has a temperature dependence of around −1.2% per Kelvin, while centered at 300 K. A further advantage is the high transparency of BGO for visible light [6].

II Set-up and methods

II.1 Measuring set-up

For our measurements we used two photomultiplier tubes (PMT) – one with our BGO crystal [7], and a smaller one equipped with a YSO (Y_2SiO_5:Ce) scintillator [8]– mounted back-to-back without any optical couplant, to maximize reproducibility of the placement. The BGO crystal was placed directly on top of the PMT window. About 4 cm above this crystal, we placed a Sodium-22 source (see appendix A.1), and another 4 cm above it, we positioned the YSO crystal of the second PMT. To reduce noise in the measurements, we exploited the fact that the two photons of the annihilation are back-to-back. We placed the two detector signals in coincidence, giving us the possibility to trigger only on the annihilation events. In the energy spectrum we therefore would expect a dominant peak at 511 keV, as well as a suppressed one at 1786 keV, since the probability for simultaneous passing of both a 511 keV and the 1275 keV photon through the BGO is small. Anyhow, there still is accidental triggering, such that we also measure only the 1275 keV photon, though this also is suppressed.

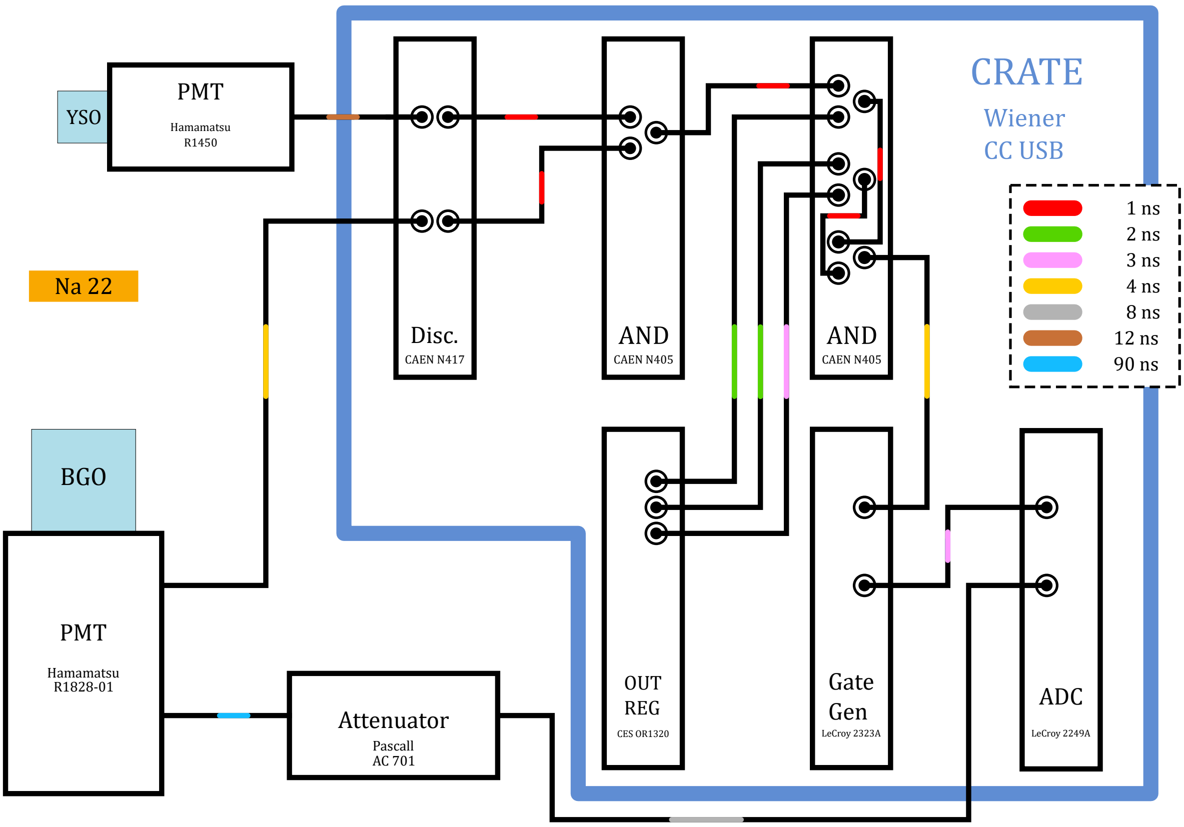


Figure 1: The setup we used for measurements and data acquisition.

The last dynode of the main PMT was coincided with the signal of the secondary PMT using an AND gate [9]. To trigger the counting, the signals passed a discriminator beforehand, which singled out signals above a certain threshold [10].

Since the data acquisition backend needs to have finished with the previous data processing, we furthermore correlated its state with the PMT coincidence signal using a second AND gate. The output of this logic unit then triggered a precise gate generator opening a gate for the main PMT's anode signal integration.

This anode was connected to an integrating 10-bit ADC [11] via a delay line, to compensate for the time lag introduced by the logic units and discriminator. We also needed to consider the effective opening and closing times of the ADC. These are given by the manufacturer as 2 ns each. Since the maximal time delay that the manufacturer indicates for an accurate measurement is 500 ns and we integrate over 1000 ns, the 4 ns introduced due to the opening times is below the relevant timescale. We furthermore introduced an attenuator in order to scale the signal to the dynamic range of the ADC, which was 256 pC.

The backend was a custom LabVIEW® environment for the Wiener CC-USB Crate Controller[12], which we used to house our data acquisition set-up. A depictation of our set-up is shown in figure 1.

The BGO we used for our measurements was produced by the Shanghai Institute of Ceramics in the 1980's as a prototype for the L3 detector at the LEP [13]. It measures 29 mm × 26 mm × 31 mm with the effective side facing the PMT window being 29 mm × 26 mm.

To wrap this scintillating crystal, we used standard household – as well as a heavier – aluminium foil, normal printer paper, Tyvek®, PTFE and polyvinyl chloride (PVC) tape, the alveolar structure used in the CMS experiment to house the Lead Tungstate crystals [14], aluminised Mylar® and 3M Vikuiti™ ESR foil. For the aluminium foils, which had a thickness of 5 μm and 18 μm respectively, we used both, the shiny and the matt side. We furthermore used the same foil just crumpled up. For each material, we formed a cap covering the whole crystal except for the ground side, which faced the PMT window. For the Vikuiti™ foil, we cut 5 pieces which we placed on the sides of the crystal using a cap made out of Tyvek®. We used both, the intended front side, as well as the back side of the foil. Since the Mylar® foil was slightly translucent, we used two layers of it for the wrapping of the crystal, to prevent any light loss.

Parameter	Value
Threshold BGO	11.1(1) mV
Threshold YSO	20.0(1) mV
Voltage BGO	2305(1) V
Voltage YSO	1100(1) V
Gatewidth	1000 ns
Attenuation	19 – 26 dB

Table 1: The optimised detector and DAQ settings used during the measurements.

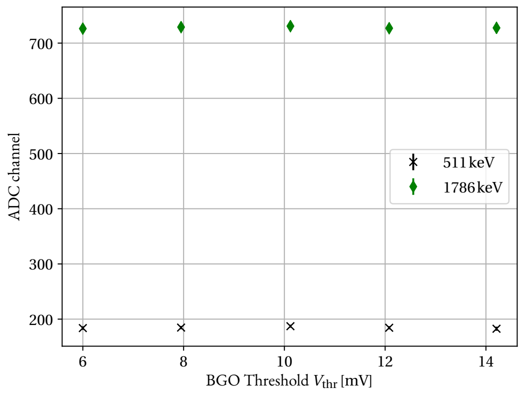


Figure 2: Peak position versus discriminator threshold value.

II.2 Optimisation

We needed to optimise several parameters of our setup prior to data taking: The discriminator threshold voltages, the gate width of the gate generator, the attenuation and the operating voltages of the PMTs. We chose the voltages of the two PMTs such that a significant signal could be seen. We did not see any significant effect of the threshold on the peak position nor the resolution (see figures 2, 3). However, a lower threshold led to the accumulation of noise in the lower part of the spectrum, while a higher threshold reduced the rate significantly. In figure 4, we depicted this situation: Figure 4a shows a measurement taken with a rather low threshold. At the lower end of the spectrum a lot of noise accumulated, which lacks in the high threshold measurement shown in figure 4b. We therefore set it to around 11 mV, where we had low noise and still had a sensible rate.

On the other hand, we saw a strong effect on the integrated current for gate widths up to 1000 ns (see also figure 5). This can be explained considering the time constant

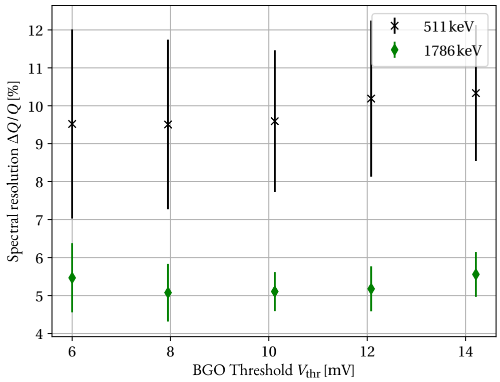
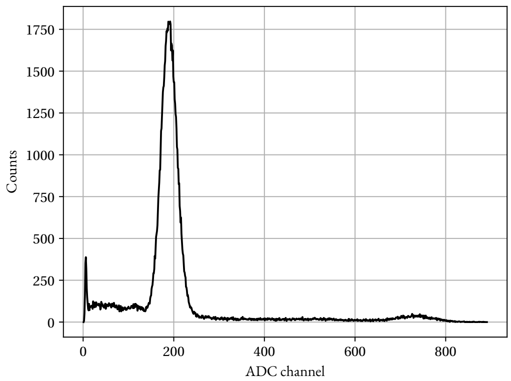
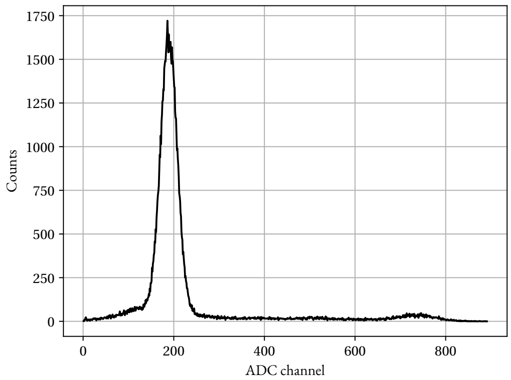


Figure 3: Spectral resolution versus discriminator threshold value



(a) A low threshold measurement at 6 mV.



(b) A high threshold measurement at 14 mV.

Figure 4: ADC spectrum for two different discrimination threshold values.

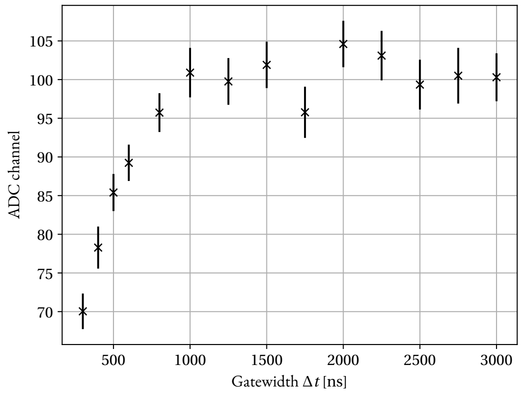


Figure 5: The dependence of the 511 keV peak position on the ADC integration gatewidth.

of BGO, which is about 350 ns [5]. After three time constants, $1 - e^{-3} \approx 95\%$ of the scintillation light was emitted. Therefore the integrated current plateaus for even higher gate widths. These, however, would have resulted in a better spectral resolution, but also would have increased the occurrence of signal pile-up. As a compromise we set the gate width to a value of 1000 ns, where the peak position began to plateau with respect to the gate width. This indicates that we indeed detetected a huge majority of the photons.

The attenuation was chosen dependend on the experiment. For the bare crystal, we used 19 dB, while for the wrappings, we used 26 dB. The final optimisation parameters used can be seen in table 1.

II.3 Reproducibility

To quantify the reproducibility of the results, we repeated a measurement of the energy spectrum without any wrapping several times, each of them with 250 000 counts. Between the runs, we opened the PMT light-tight cover, removed the crystal, put it back and sealed the tube again, in order to mimic the wrapping procedure. Like in the later measurements, we used a stencil to place the crystal as close as possible to the original position. The nine-measurement series indicates a relative systematic error of about 2%. This is about what we have expected from a relocation of the crystal. The temperature-corrected positions of the 511 keV peaks for each run are depicted in figure 6, where we corrected the values for the dependence of BGO's scintillation light yield on temperature, which is $-1.2\%\mathrm{K}^{-1}$

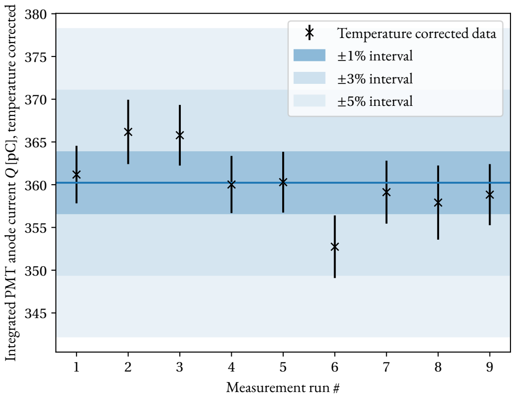


Figure 6: The 511 keV peak position for each of the runs for the reproducibility study. We normalised the data to a temperature of 18 °C.

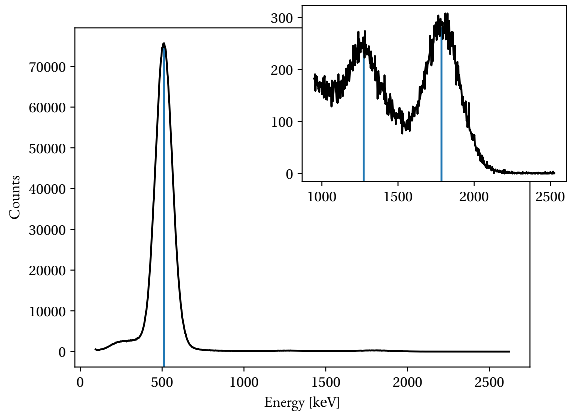


Figure 7: The recorded energy spectrum without wrapping. We have dominant peak at 511 keV and two other peaks, shown in an enlarged view, which are highly suppressed. The blue lines mark the expected peaks at 511 keV, 1275 keV, and 1786 keV.

at room temperature [1]. Therefore – respecting the relative systematic error of 2% – the results are reproducible.

III Results

III.1 Energy spectrum

In figure 7, a measurement of the energy spectrum of our Sodium-22 source taken without a wrapping of three million counts over 15 hours is shown. We clearly have a dominant peak, and since we triggered on back-to-back events, we can identify it with a 511 keV photon resulting from the electron – positron annihilation. Using this identification and assuming it to be linear, we thus had a correspondence between ADC channels and energy. The results justify this assumption, as the data fits the expected peak energy values, which are depicted by the blue lines in figure 7.

The measured peak positions plotted against the energy identification of the peaks is shown in figure 8. Again, we have an almost perfect linear dependence which further consolidates the identification made. In principle, the correspondence between ADC channels and energy should be proportional. We have, however, a significant positive offset, i.e. the ADC channel zero corresponds to a finite energy of about 100 keV. This might be due to insufficient energy containment, which would result in measuring fewer photons than we would have in an array of crystals.

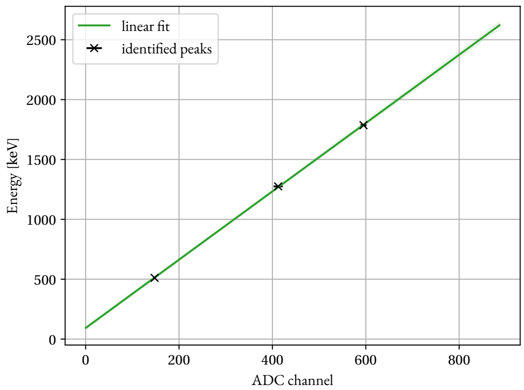


Figure 8: The measured peak positions in the spectrum versus the expected energy values. The green line is a linear fit. Note that errors are included but too small to be visible. The error on the fit is depicted by the pale green area around the fit, but also quite small. Note the small but significant offset.

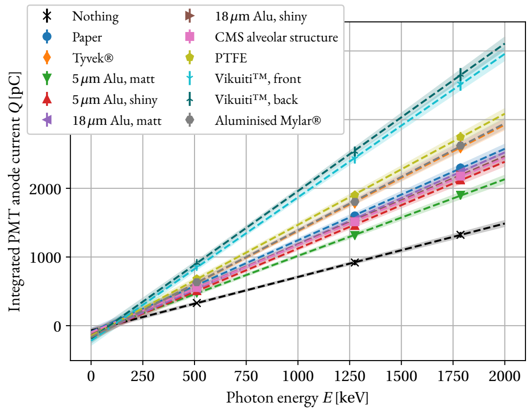


Figure 9: The measured PMT charge yield of the three peaks for each material. The dotted lines are linear fits.

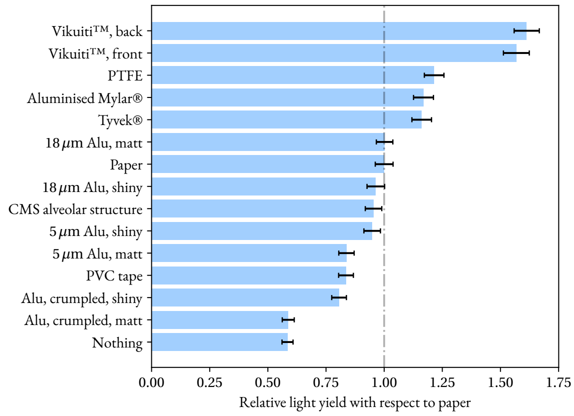


Figure 10: The light yields of all the materials we studied, as extracted from the linear fits of the three peaks. The values are relative with respect to a paper wrapping.

III.2 Wrapped light output

For each material, we measured an energy spectrum with 3 million counts in total, resulting in about 15 hours of data taking. A comparative plot of the positions of the three identified peaks for the different wrapping materials we used is shown in figure 9. Most of the materials we studied were quite similar in terms of charge yield at the PMT anode, which corresponds directly to the light output. In figure 10, the relative light yields with respect to paper, as extracted from the linear fits, are shown.

Outstanding results were achieved with the 3M Vikuiti™ foil, which has a mirror-like, very reflective surface. The front side is stated to be slightly more reflective than the back side [3], but we achieved considerably higher light outputs with its back side. The next best wrappings were PTFE and Tyvek®, both very white, diffuse surfaces. Similarly, standard white paper delivered a good light output. We also tested aluminised Mylar® foil, which – like Vikuiti™ – has a very reflective, mirror-like surface. However, it did not perform as good as the 3M foil, but rather achieved results not significantly different from those of Tyvek®.

For the aluminium foils we obtained the best results with the slightly heavier version, where the matt side resulted in a slightly higher light output. While the shiny side of the lighter aluminium foil delivered comparable results to that of the heavier one, its matt side was one of the worst wrappings we studied in terms of light output. We furthermore crumpeled up the lighter foil to increase its diffusivity, but this resulted in light outputs comparable to the the crystal without any wrapping. We therefore did not include the respective data in figure 9 for clarity.

Also tested, but also not included in the plot was self-sticking white PVC tape. We wanted to reduce the total reflection in the crystal by applying the tape directly on its surface. However, it gave virtually the same results as the matt side of standard household aluminum foil.

We furthermore wrapped the crystal in the material from which the alveolar structure holding the $PbWO_4$ crystals in the ECAL of the CMS detector was made. This led to a light output comparable to the heavy aluminium foil. Hence it was in the middle range of the materials studied.

III.3 Energy resolution for wrapped BGO

We also considered the influence of wrapping on the energy resolution of the scintillator, which is of great importance in many applications. In figure 11, we have a plot of the charge yield as extracted from the linear fits, plotted against the resolution for each material. The data shows a negative correlation between those two: For a material with higher charge yield, we also have sharper peaks. This was to be expected, as a higher light output increases the total charge yield for each peak, but should not broaden the peak itself. However, some materials with similar light outputs seem to differ slightly in their resolutions, for example Paper and the matt side of the heavy aluminium foil, or Tyvek® and the aluminised Mylar®. Respecting the errors we assessed on the other hand, this difference is not significant.

As before, the back side of the Vikuiti foil™ performed best. Its resolution of 5 – 6% is about twice as good as for

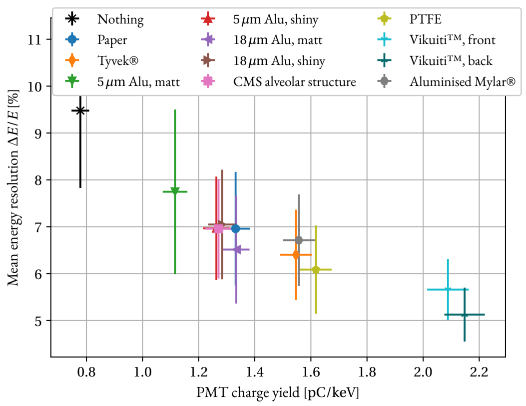


Figure 11: The charge yield plotted against the mean energy resolution. As we have three peaks with known energies, we calculated the arithmetic mean of their resolutions for each material. We calculated the mean of these.

the unwrapped crystal. The other materials all resolved the energy spectrum with about 6 – 8%, which is also considerably better than the bare crystal.

Again, we omitted the data for the crumpled aluminium foil as well as the self-sticking PVC tape in the figure for more clarity.

III.4 Interpretation

The highly reflective Vikutiti™ foil shows the best results, since virtually all light escaping the crystal is reflected back; almost none is absorbed by the foil. BGO is very transparent for its own scintillation light, resulting in nearly no decrease in intensity at all. After few reflections, the light will enter the photo detector with a comparable intensity to that it originally had. This causes the high light output observed in the measurements.

The very diffusive materials like PTFE and Tyvek® also have a strong effect on the light output. Even though they are not as reflective as Vikuiti™, a certain part of the escaping photons is always scattered towards the PMT. This increases the light output significantly. The diffusive materials we used were Tyvek®, Paper, PTFE and PVC tape. Since PTFE is the most diffusive of these four, we expected it to perform the best, as we indeed have seen in our measurements. Similarly, the very diffusive Tyvek® also delivered a comparably high light output. The less diffusive paper and PVC performed considerably worse.

The perfomance of the aluminium foils is highly dependend on the smoothness of their surface. For the heavier, smoother aluminium foil, we have a light output similar to that of paper. Even though aluminium is reflective, its reflectivity is not sufficient to guide the light into the photo detector: A high portion of the photons are absorbed by the material, resulting in a lower intensity of the reflected ones. It is also not diffusive enough to distribute the light evenly through the whole crystal, like PTFE and Tyvek® have. Anyway, both effects are still present, and their sum leads to a considerable benefit compared to the bare crystal. For the CMS alveolar structure, consisting out of an aluminised carbon fiber structure, we have a similar situation: The aluminium reflects some of the light, while the carbon fiber structure absorb a considerable amount. This leads to a performance slightly worse than that of paper.

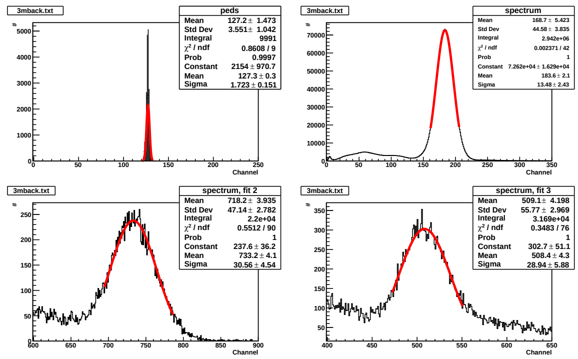


Figure 12: The result of our fitting routine using ROOT and the code in appendix B on the data of the backside of the 3M Vikuiti ™. The plots show clockwise the pedestals, the 511 keV, the 1275 keV and the 1786 keV peak. The red lines are the fitted curves.

We originally crumpled the aluminium foil to increase its diffusivity. However, it performed worse than all of the other materials, its matt side not being significantly different from no wrapping at all. The reason is that some of the light gets reflected into the creases, where it will escape only with a heavily reduced intensity, as the reflectivity of the foil was not very high. The matt side of the crumpled aluminium foil therefore absorbed almost all of the light, while its specular side achieved a performance slightly worse than that of the PVC tape. This makes the crumpled aluminium the worst material we studied in terms of light output.

IV Data analysis

For the data analysis we used ROOT and Python. The spectrum was fitted with the ROOT routine shown in appendix B. From these fits we extracted the peak positions and standard deviations, an example for which can be seen in figure 12. These values then were further analysed with the standard Python modules *NumPy* and *SciPy*, while for the error propagation we used the package *uncertainties*. We used the pseudo-inverse from *uncertainties* for the linear fits, which corresponds to the least square method. It also allowed us to compute errors for the fitting parameters. As discussed in appendix A.4, the systematic error resulting from the peak shift due to the Compton edge is neglible compared to the errors introduced by the fitting rootine. Further, concentrating on the right hand side of the gaussian peak, we could avoid the problem of the peak shift. This is because the Compton edge altered predominantly the left hand side of the peak. The systematic errors due to the inconsistency of the crystal placement on the PMT is discussed in II.3. We created the all images with Inkscape, while we utilised *matplotlib* for our plots.

V Conclusion

We studied the influence of wrapping on the light output, as well as on the energy resolution of a BGO scintillating crystal. The best performance was achieved using the 3M Vikuiti™ foil, which almost tripled the light output and halved the energy resolution with respect to the unwrapped crystal. More common wrapping materials like PTFE, Tyvek® and Mylar® also resulted in a considerable improvement of the scintillation properties: They doubled the light output, and achieved an energy resolution of $6-7\%$ (as opposed to $\sim 10\%$ without wrapping).

There seemed to be no preference for diffusive or reflective surfaces, as both the very white, diffusive and the highly reflective surfaces – namely PTFE and Tyvek® respectively Vikuiti™ and Mylar® – considerably improved the scintillating light output and energy resolution. However, the extremely reflective Vikuiti™ foil performed best by a clear margin.

Further improvements could be achieved using an array of crystals to guarantee energy containment, which would result in an even better resolution.

Acknowledgments

We would like to thank Ms. Gabriele Kogler, for her generous help in organising our stay. We also want to express our deep thanks to Dr. Francesca Nessi-Tedaldi, for her helpful insights and her valuable suggestions, which encouraged us to endure the long measurements. Furthermore, we want to thank Prof. Dr. Günther Dissertori for the opportunity to work at his group.

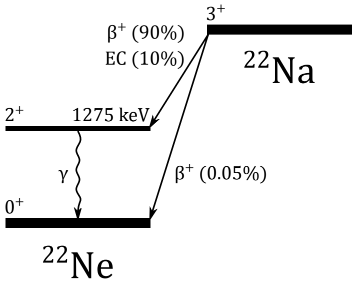


Figure 13: The decay scheme of ^{22}Na [15]. The dominant decay is into an excited ^{22}Ne state, which decays into the ground state under emission of a 1275keV photon. The decay directly into the ground state is possible, but extremely suppressed, as it is a $\Delta J = 3$ transition.

A Appendix

A.1 Radioactive source

We used a ^{22}Na source with an activity of about 22 kBq. This isotope has a half life of 2.60 a and decays dominantly either by β^+-decay or electron capture into an excited ^{22}Ne state, which decays into the ground state under emission of gamma radiation, as can be seen in the term scheme (fig. 13). In the case of the β^+-decay, one would register two back-to-back 511 keV photons due to the positron annihilation, as well as a 1275 keV photon resulting from the decay into the ground state. Another possible decay is a β^+-decay directly into the ground state of ^{22}Ne, such that only the two 511 keV annihilation photons can be measured. However, this decay is highly supressed, as it involves an angular momentum change of $\Delta J = 3$.

A.2 Radiation protection

To estimate our radiation exposure, we calculated the ambient dose equivalent rate $\dot{H}^*_{10}(r)$ using the formula

$$\dot{H}^*_{10}(r) = \frac{A \cdot \Gamma_{H^*}}{r^2},$$

where A is the activity of the source, r the distance, and Γ_{H^*} the ambient dose equivalent rate constant, which is $0.333\,\mathrm{nSv\,m^2\,kBq^{-1}h^{-1}}$ in the case of ^{22}Na [16]. We furthermore measured it using a dose rate metre. The results are listed in table 2. If we assume working at a distance of 30 cm to the source most of the time, our radiation exposition is comparable to the natural radiation background, which lies between 80 and 190 nSv/h in Genève [17].

Distance	$\dot{H}^*_{10}$, calculated	$\dot{H}^*_{10}$, measured
10 cm	733 nSv/h	590 nSv/h
20 cm	183 nSv/h	190 nSv/h
30 cm	81 nSv/h	112 nSv/h

Table 2: Measured and calculated ambient equivalent dose rates for ^{22}Na in various distances.

The intensity of gamma radiation in matter is attenuated according to

$$I = I_0\,\mathrm{e}^{-\rho d/\lambda},$$

where ρ is the material's density, d its thickness, and λ is the attenuation length. In the case of lead, which we used for radiation protection, it is $\lambda \approx 6\,\mathrm{g/cm^2}$ for 511 keV, and $\lambda \approx 15\,\mathrm{g/cm^2}$ for 1275 keV photons [1]. Our 5 cm lead bricks would therefore reduce the intensity of the 511 keV and 1275 keV photons to 0.79% and 2.28% respectively. This is sufficient for a reduction of our radiation exposure to less than 1% of the natural background.

A.3 Photomultiplier tube

A photomultiplier tube (PMT) is a measurement device for the detection of photons, mostly from the visible part of the electromagnetic spectrum. Its main working principle is the multiplication of a single photo electron using secondary emission. The PMT used consisted out of a bialkali coated photocathode and several dynodes at different potentials. A photon enters the PMT through the window, resulting in a single photo electron emmision. This electron then is focused and accelerated towards the first dynode. Hitting the dynode, the electron releases several (k) secondary electrons. These are then accelerated towards the next dynode where, through the same mechanisms, they are again multiplied (see fig. 14). After n dynodes, we get k^n electrons leading to a measureable current at the anode. If the PMT's gain is known, integrating this current for a specific event gives the number of photoelectrons which is proportional the light output of the scintillator. To get the light yield of the scintillator, one needs to correct this data for the quantum efficiency of the PMT and the geometrical solid angle, the PMT covers of the scintillator.

A.4 Estimation of Compton background

For a crude estimate of the systematic error on the peak position due to the underlying Compton edge, we modeled it

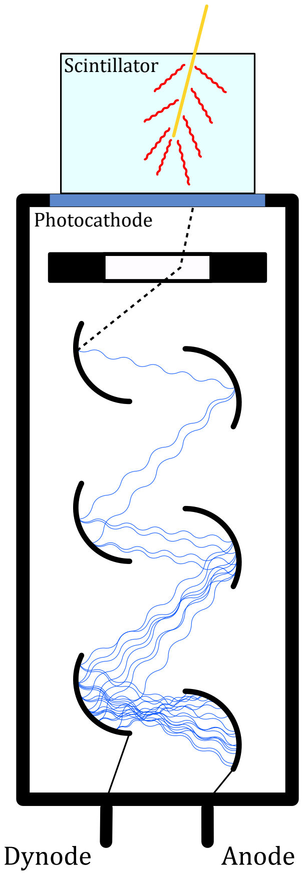
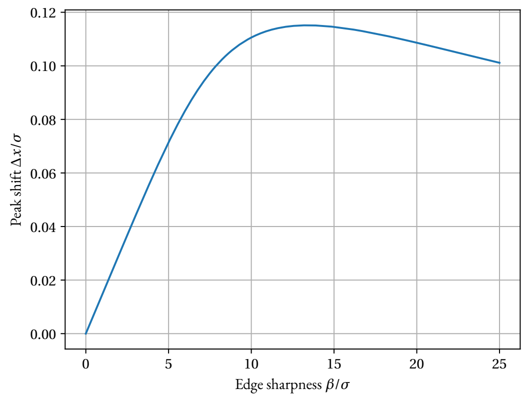
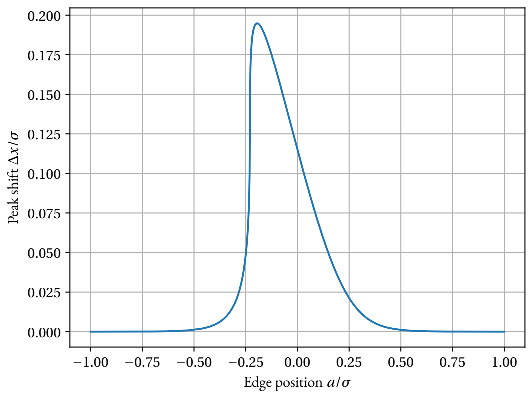


Figure 14: Schematics of a PMT. The scintillating crystal emits several low energy photons, which result in the emission of photo electrons at the photo cathode. These get multiplied using a series of dynodes, and then registered as a current at the anode.



(a)



(b)

Figure 15: The shift of the peak position as a function of the (a) hardness β and (b) position a (for the worst case β) of the Compton edge. Even in the worst case scenario, the peak is shifted less than 20% of the standard deviation.

using a Fermi function

$$\frac{1}{e^{\beta(x-a)}+1}.$$

We furthermore used a standard Gaussian $Ae^{-x^2/2\sigma^2}$ to model the peak. We estimated the amplitude of the Compton background to about 5% of the peak's height, which matches our data for the 511 keV peak.

The peak shift due to a Compton edge at $a = 0$ in dependence of the sharpness β is shown in figure 15a. It peaks at about $\beta = 13$, where the shift is not quite 12%. In this worst case scenario, we calculated the effect the relative position a of the edge on peak position, as shown in figure 15b. Assuming the case where the peak shift is maximal, the systematic error is still below 20% of the standard deviations. Compared to our other errors, for example due to fitting of the peak, this is relatively small. Therefore, we concluded that the effect of the Compton background is neglible.

B Code for fitting

```
// F. Nessi, rev. November 2018
// rev. June 2019 by
// T. Dieminger, R. Schwarz
// ROOT version of hisana.kumac.
// For analysis of spectra from
// LeCroy 2249A, 2038 lines ASCII,
// header, 1024, ped data, 1024,
// spectrum data

#include "TSpectrum2.h"
#include "TRandom.h"
#include "TH2.h"
#include "TF2.h"
#include "TMath.h"
#include "TROOT.h"
#include <TFile.h>
#include <TNtuple.h>
#include <TH2.h>
#include <TProfile.h>
#include <TCanvas.h>
#include <TFrame.h>
#include <TROOT.h>
#include <TSystem.h>
#include <TRandom3.h>
#include <TBenchmark.h>
#include <TInterpreter.h>

void hisana1024_v6()
{
gROOT->Reset();
gROOT->SetStyle("Plain");
gStyle->SetOptStat(1002201);
gStyle->SetOptFit(1111);

const int nchan=1024;
int i=0;
const int fdata=2060.;
float f1min=10.;
float f1max=100.;
float f2min=50.;
float f2max=150.;
Double_t f1entries=0.;
Double_t f2entries=0.;
Double_t f3entries=0.;
Double_t f4entries=0.;
Double_t Ped[nchan];
Double_t Spect[nchan];
Double_t SubSpect[nchan];
Double_t Raw[fdata];

Double_t x_value[nchan];
Double_t y_value[nchan];
Double_t x_value_err[nchan];
Double_t y_value_err[nchan];

int Sub_Ped;
float Cos_Fit;
TString answ = "␣";
TString answ2 = "␣";
```

```
TString answp = "␣";

int ntxtstr=0;

char txtstr[50];

//------------- Reading data from file
char hisname[100];
char filename[100];
char outname[]="fitresults.txt";
char str[] = "-268.14␣Temperature_1(C)";
char string[] = "";
float Value;
cout <<
"Please␣ENTER␣the␣name␣of␣data␣file:␣";
cin >> filename;
cout << filename << endl;
ifstream infileabs (filename,ios::in);
if(!infileabs.is_open()){
cout << "No␣inputfile␣to␣open!" << endl;
}

ofstream outfileabs (outname,ios::app);
if(!outfileabs.is_open()){
cout<<"No␣outputfile␣to␣open!"<<endl;
}

while(infileabs.good() && i<fdata){
// extract line into a string:
infileabs.getline(str,256);
sscanf(str, "%f␣%s", &Value, string);
Raw[i] = Value;
i++;
}
infileabs.close();

for(int z=0;z<fdata;z++){
if(z>=11 && z<=1034){   // Pedestal
Ped[z-11]=Raw[z];}
if(z>=1036){            // Spectrum
Spect[z-1036]=Raw[z];}
}

for(int z=0;z<nchan;z++){
x_value[z]=z;
}

// Introduce Variables for fitting range
int nped;
int nspect;
int nspectlow = 0;

for(int n=nchan-1; Ped[n]==0; n--){
nped = n;
}
for(int n=nchan-1; Spect[n]==0; n--){
nspect = n;
}

f1min = nped/3;
f1max = 2*nped/3;
f2min = nspect/4;
f2max = 3*nspect/4;

//---------- 1st Window: Pedestal
TCanvas *c1 = new TCanvas("c1","Graph",
                          0,0,1536,1152);
c1->Divide(2,2);
c1->cd(1);
TH1F *h301 = new TH1F("peds",filename,
                      1024,0.,1024.);
for(int z=0;z<nchan;z++){
h301->Fill(x_value[z],Ped[z]);
f1entries += Ped[z];
}
cout << "ped␣entries␣=␣"
     << f1entries << endl;
h301->GetXaxis()->SetTitle("Channel");
h301->GetYaxis()->SetTitle("#");
h301->GetXaxis()->SetTitleOffset(1.0);
h301->GetYaxis()->SetTitleOffset(1.0);
h301->GetXaxis()->SetRange(0,nped);
// Define the fit function:
TF1 *f1
= new TF1("fit1","gaus",f1min,f1max);
f1->SetLineColor(2);
h301->Fit("fit1","R");
c1->Update();
gSystem->ProcessEvents();
Sub_Ped=f1->GetParameter(1);
cout << "Fitted␣pedestal␣value␣=␣"
     << Sub_Ped << endl;
cout<< "Pedestal␣fit␣OK?␣[y/n]:␣";
cin >> answ;
cout << "Answer␣=␣" << answ << endl;
if  (answ!="y")
{
check0:
cout <<
"Manually␣select␣fitting␣range?␣[y/n]";

cin >> answ2;
if(answ2=="n"){}
else if(answ2=="y"){
cout << "Give␣upper␣bound:␣";
cin >> nped;
h301->GetXaxis()->SetRange(0,nped);
}
else {
goto check0;
}
do
{
cout <<
"Give␣start␣value␣for␣pedestal␣fit:␣";
cin >> f1min;
printf ("pedmin:␣%4.0f␣\n",f1min);
cout <<
"Give␣end␣value␣for␣pedestal␣fit:␣";
cin >> f1max;
printf ("pedmax:␣%4.0f␣\n",f1max);
TF1 *f1 = new TF1("fit1","gaus",
                  f1min,f1max);
f1->SetLineColor(2);
```

```
h301->Fit("fit1","R");
// ("R" restricts the fitting range)
c1->Update();
gSystem->ProcessEvents();
Sub_Ped=f1->GetParameter(1);
cout <<
"Fitted␣pedestal␣value␣(rounded)␣=␣"
      << Sub_Ped<<endl;
cout << "Pedestal␣fit␣OK?␣[y/n]␣";
cin >> answ;
} while (answ!="y");
}
gSystem->ProcessEvents();

// 2. Window: Pedestal-subtracted Spectrum
for(int z=0; z<nchan-Sub_Ped; z++){
// Pedestal subtraction by spectrum shift:
SubSpect[z] = Spect[z+Sub_Ped];
}
for(int z=nchan-Sub_Ped; z<nchan; z++){
SubSpect[z] = 0;
}
c1->cd(2);
TH1F *h201 = new TH1F("spectrum",filename,
                      1024,0.,1024.);
for(int z=0; z<nchan; z++){
h201->Fill(x_value[z],SubSpect[z]);
f2entries += SubSpect[z];
}
h201->GetXaxis()->SetTitle("Channel");
h201->GetYaxis()->SetTitle("#");
h201->GetXaxis()->SetTitleOffset(1.0);
h201->GetYaxis()->SetTitleOffset(1.0);
h201->GetXaxis()
    ->SetRange(nspectlow,nspect);

h201->SetFillColor(0);
h201->Draw("hist");
// fetch the integral of bin contents
Double_t hisentries = h201->Integral();
//Define the fit function
TF1 *f2
= new TF1("fit2","gaus",f2min,f2max);
f2->SetLineColor(2);
h201->Fit("fit2","R");
ntxtstr
= sprintf(txtstr,"entries␣=␣%7.0f",
                  hisentries);

TLatex latex;
latex.SetTextFont(42);
latex.DrawLatex(150.,10.,txtstr);
f2->Draw("same");
c1->Update();
gSystem->ProcessEvents();
Cos_Fit=f2->GetParameter(1);

cout << "Spectrum␣peak␣value␣␣=␣"
      << Cos_Fit << endl;
cout << "Peak␣fit␣OK?␣[y/n␣]:␣";
cin >> answ;
if  (answ!="y")
{
check:
cout <<
"Manually␣select␣fitting␣range?␣[y/n]";

cin >> answ2;
if(answ2=="n"){}
else if(answ2=="y"){
cout << "Give␣lower␣bound:␣";
cin >> nspectlow;
cout << "Give␣upper␣bound:␣";
cin >> nspect;
h201->GetXaxis()
    ->SetRange(nspectlow,nspect);
}
else {
goto check;
}
do
{
cout << "Give␣start␣value␣for␣peak␣fit:␣";
cin >> f2min;
printf("lower␣limit:␣%4.0f␣\n", f2min);
cout <<
"Give␣(dotted)␣end␣value␣for␣peak␣fit:␣";
cin >> f2max;
printf ("upper␣limit:␣%4.0f␣\n",f2max);
TF1 *f2 = new TF1("fit2","gaus",
                  f2min,f2max);
f2->SetLineColor(2);
h201->Fit("fit2","R");
ntxtstr
= sprintf(txtstr,"entries␣=␣%7.0f",
                  hisentries);
h201->Draw("hist");
f2->Draw("same");
c1->Update();
gSystem->ProcessEvents();
Cos_Fit=f2->GetParameter(1);
cout << "peak␣value␣␣=␣"
     << Cos_Fit << endl;
cout << "peak␣fit␣fit␣OK?␣[y/n]";
cin >> answ;
} while (answ!="y");
}

// Defining something, otherwise problemos
TH1F *h205
= new TH1F("spectrum,␣fit␣2", filename,
           1024.,0.,1024.);
float f3min;
float f3max;
TF1 *f3 = new TF1("fit3","gaus",
                  f3min,f3max);
Double_t hisentries2;
TLatex latex2;

TH1F *h207
= new TH1F("spectrum,␣fit␣3",filename,
           1024.,0.,1024.);
float f4min;
float f4max;
```

```
TF1 *f4 = new TF1("fit4","gaus",
                  f4min, f4max);
Double_t hisentries3;
TLatex latex3;

check1:
cout << "Another␣plot?␣[y/n]␣";

cin >> answp;
if (answp=="y"){
c1->cd(3);

for(int z=0; z<nchan; z++){
h205->Fill(x_value[z],SubSpect[z]);
f3entries += SubSpect[z];
}

//define limits
int nspect2low;
int nspect2;
f3min = 1.*nspect2low;
f3max = 1.*nspect2;
cout << "Give␣lower␣bound␣for␣2nd␣plot:␣";
cin >> nspect2low;
cout << "Give␣upper␣bound␣for␣2nd␣plot:␣";
cin >> nspect2;


h205->GetXaxis()->SetTitle("Channel");
h205->GetYaxis()->SetTitle("#");
h205->GetXaxis()->SetTitleOffset(1.0);
h205->GetYaxis()->SetTitleOffset(1.0);
h205->GetXaxis()
    ->SetRange(nspect2low,nspect2);
h205->Draw("hist");
// fetch the integral of bin contents
hisentries2 = h205->Integral();

// Define the fit function
f3->SetLineColor(2);
h205->Fit("fit3","R");

ntxtstr=sprintf(txtstr,"entries␣=␣%7.0f",
                hisentries2);
latex2.SetTextFont(42);
latex2.DrawLatex(150.,10.,txtstr);

f3->Draw("same");
c1->Update();

gSystem->ProcessEvents();
Cos_Fit=f3->GetParameter(1);

cout << "Spectrum␣peak␣value␣␣=␣"
     << Cos_Fit << endl;
cout << "Peak␣fit␣OK?␣[y/n␣]:␣";
cin >> answ;
if(answ!="y")
{
check2:
cout << "Reselect␣range?␣[y/n]␣";
cin >> answ2;
```

```
if(answ2=="n"){}
else if(answ2=="y"){
cout << "Give␣lower␣bound:␣";
cin >> nspectlow;
cout << "Give␣upper␣bound:␣";
cin >> nspect;
h205->GetXaxis()
    ->SetRange(nspectlow,nspect);
}
else {
goto check2;
}
do
{
cout <<
"Give␣start␣value␣for␣peak␣fit:␣";
cin >> f3min;
printf("lower␣limit:␣%4.0f␣\n",f3min);
cout <<
"Give␣end␣value␣for␣peak␣fit:␣";
cin >> f3max;
printf ("upper␣limit:␣%4.0f␣\n",f3max);
TF1 *f3 = new TF1("fit3","gaus",
                  f3min,f3max);
f3->SetLineColor(2);
h205->Fit("fit3","R");
ntxtstr=sprintf(txtstr,"entries␣=␣%7.0f",
                hisentries2);
h205->Draw("hist");
f3->Draw("same");
c1->Update();
gSystem->ProcessEvents();
Cos_Fit=f3->GetParameter(1);
cout << "peak␣value␣␣=␣"
     << Cos_Fit << endl;
cout<< "peak␣fit␣fit␣OK?␣[y/n]␣";
cin >> answ;
} while (answ!="y");
}
check3:
cout << "Another␣plot?␣[y/n]␣";

cin >> answp;
if (answp=="y"){
c1->cd(4);

for(int z=0;z<nchan;z++){
h207->Fill(x_value[z],SubSpect[z]);
f4entries += SubSpect[z];
}

//define limits
int nspect3low;
int nspect3;
cout << "Give␣lower␣bound␣for␣3rd␣plot:␣";
cin >> nspect3low;
cout << "Give␣upper␣bound␣for␣3rd␣plot:␣";
cin >> nspect3;
f4min = 1.*nspect3low;
f4max = 1.*nspect3;

h207->GetXaxis()->SetTitle("Channel");
```

```
h207->GetYaxis()->SetTitle("#");
h207->GetXaxis()->SetTitleOffset(1.0);
h207->GetYaxis()->SetTitleOffset(1.0);
h207->GetXaxis()
    ->SetRange(nspect3low,nspect3);
h207->Draw("hist");
// fetch the integral of bin contents
hisentries3 = h207->Integral();
// Define the fit function

f4->SetLineColor(2);
h207->Fit("fit4","R");

ntxtstr=sprintf(txtstr,"entries␣=␣%7.0f",
                hisentries3);

latex3.SetTextFont(42);
latex3.DrawLatex(150.,10.,txtstr);

f4->Draw("same");
c1->Update();

gSystem->ProcessEvents();
Cos_Fit=f4->GetParameter(1);

cout << "Spectrum␣peak␣value␣␣=␣"
     << Cos_Fit << endl;
cout << "Peak␣fit␣OK?␣[y/n␣]:␣";
cin >> answ;
if  (answ!="y")
{
check4:
cout << "Reselect␣range?␣[y/n]␣";

cin >> answ2;
if(answ2=="n"){}
else if(answ2=="y"){
cout << "Give␣lower␣bound:␣";
cin >> nspectlow;
cout << "Give␣upper␣bound:␣";
cin >> nspect;
h207->GetXaxis()
    ->SetRange(nspectlow,nspect);
}
else {
goto check4;
}
do
{
cout <<
"Give␣start␣value␣for␣peak␣fit:␣";
cin >> f4min;
printf("lower␣limit:␣%4.0f␣\n",f4min);
cout <<
"Give␣(dotted)␣end␣value␣for␣peak␣fit:␣";
cin >> f4max;
printf("upper␣limit:␣%4.0f␣\n",f4max);
TF1 *f4
= new TF1("fit4","gaus",f4min, f4max);
f4->SetLineColor(2);
h207->Fit("fit4","R");
ntxtstr=sprintf(txtstr,"entries␣=␣%7.0f",
                hisentries3);
h207->Draw("hist");
f4->Draw("same");
c1->Update();
gSystem->ProcessEvents();
Cos_Fit=f4->GetParameter(1);
cout << "peak␣value␣␣=␣"
     << Cos_Fit << endl;
cout << "peak␣fit␣fit␣OK?␣[y/n]␣";
cin >> answ;
} while (answ!="y");
}
}
else if (answp=="n"){}
else {
goto check3;
}
}
else if (answp=="n"){}
else {goto check1;}

End:
// Here we want to apologize to all
// the people, who made it up to this
// point, for the excessive use of
// GoTos in this code.
outfileabs.setf(ios_base::right);
outfileabs.width(13);
outfileabs << filename;
outfileabs.setf(ios_base::right,
                ios_base::fixed);
outfileabs.precision(4);
outfileabs.width(7);
outfileabs << f1->GetParameter(1);
outfileabs.width(7);
outfileabs << f1->GetParameter(2);
outfileabs.width(7);
//
outfileabs << "␣h201␣";
outfileabs.precision(7);
outfileabs.width(8);
outfileabs << f2entries;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f2min;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f2max;
outfileabs.width(9);
outfileabs << setprecision(5)
           << f2->GetParameter(1);
outfileabs.width(8);
outfileabs << setprecision(3)
           << f2->GetParError(1);
outfileabs.width(9);
outfileabs << setprecision(5)
           << f2->GetParameter(2);

if (answp=="y"){
outfileabs << "␣␣h205␣";
outfileabs.precision(7);
outfileabs.width(8);
```

```
outfileabs << f3entries;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f3min;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f3max;
outfileabs.width(9);
outfileabs << setprecision(5)
           << f3->GetParameter(1);
outfileabs.width(8);
outfileabs << setprecision(3)
           << f3->GetParError(1);
outfileabs.width(9);
outfileabs << setprecision(5)
           << f3->GetParameter(2);
outfileabs << "␣h207␣";
outfileabs.precision(7);
outfileabs.width(8);
outfileabs << f4entries;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f4min;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f4max;
outfileabs.width(9);
outfileabs << setprecision(5)
           << f4->GetParameter(1);
outfileabs.width(8);
outfileabs << setprecision(3)
           << f4->GetParError(1);
outfileabs.width(9);
outfileabs << setprecision(5)
           << f4->GetParameter(2);
}
else if (answp=="n"){
outfileabs << "␣␣h205␣";
outfileabs.precision(7);
outfileabs.width(8);
outfileabs << f3entries;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f3min;
outfileabs.width(6);
outfileabs << setprecision(4)
           << f3max;
outfileabs.width(9);
outfileabs << setprecision(5)
           << f3->GetParameter(1);
outfileabs.width(8);
outfileabs << setprecision(3)
           << f3->GetParError(1);
outfileabs.width(9);
outfileabs << setprecision(5)
           << f3->GetParameter(2);
}

outfileabs << endl;
outfileabs.close();
c1->Draw();
char plotfilename[100];
cout <<
"Give␣plot␣file␣name␣to␣print:␣";
cin >> plotfilename;
c1->Print(plotfilename,"pdf");
}
```

References

[1] M. Tanabashi *et al.* (Particle Data Group), Phys. Rev. D 98, 030001 (2018).

[2] *Tyvek®*, DuPont, Wilmington, 974 Centre Rd. DE 19805 U.S.A.

[3] *Vikuiti™ Enhanced Specular Reflector, Application Guidelines*, 3M Company, St. Paul, MN 55144-1000, U.S.A., 2003.

[4] W.H. Wong *et al.*, Proc. IEEE Nuc. Sci. Sym. Conf. rec., Vol. 6, M9-63 (2004), pp. 3407-3411.

[5] Paul Lecoq *et al.*, *Inorganic Scintillators for Detector Systems*, Springer 2006.

[6] M.J. Weber and R.R. Monchamp, J. Appl. Phys. 44, 5495 (1973).

[7] *Photomultiplier tube R1828-01*, HAMAMATSU PHOTONICS K.K., Electron Tube Division 314-5, Shimokanzo, Iwata City, Shizuoka Pref., 438-0193, Japan, 2010.

[8] *Photomultiplier tube R1450*, HAMAMATSU PHOTONICS K.K., Electron Tube Division 314-5, Shimokanzo, Iwata City, Shizuoka Pref., 438-0193, Japan, 1998.

[9] *Mod. N405 3-Fold Logic Unit*, CAEN S.p.A., 55049 Viareggio (LU), Italy.

[10] *Mod. N417 8-Channel Low Threshold Discriminator*, CAEN S.p.A., 55049 Viareggio (LU), Italy.

[11] *Model 2249A 12-channel ADC*, LeCroy Corporation, Research Systems Division, Chestnut Ridge, NY 10977-6499, U.S.A., 1992.

[12] *Wiener CC-USB Crate Controller*, Wiener Power Electronics GmbH, Burscheid, Germany.

[13] B. Adeva *et al.*, Nucl. Instr. Meth. Phys. Res. A 289 (1990), pp. 35-102.

[14] CMS Collaboration, JINST 3 S08004 (2008).

[15] C.M. Lederer, J.M. Hollander and I. Perlman, *Table of Isotopes*, p. 162, John Wiley & Sons 1967.

[16] Hanno Krieger, *Grundlagen der Strahlungsphysik und des Strahlenschutzes*, pp. 331-365, Springer 2017.

[17] Nationale Alarmzentrale, *Measured values of the NADAM station in Genève-Cointrin* [online], 5th of September 2019, https://www.naz.ch/en/aktuell/zeitverlauf_GVE.shtml