Laboratory Course for Master Students of Particle Physics
Experiment: T18
GEANT4 *and the Spectrum of Cosmic Muons*

Prerequisites

- introductory course to the laboratory class
- related sections of the Particle Physics Book

Goal of the experiment

- assembly and usage of a measuring set-up for comic rays
- simulation of the measurement with the GEANT4 toolkit
- comparison between real measurement and simulation

Original Author: Andreas Herten, III. Physikalisches Institut B, RWTH Aachen
Translated by: Markus Vehring, III. Physikalisches Institut B, RWTH Aachen

Contents

References

[1] C. Amsler, M. Doser, M. Antonelli, D. Asner, K. Babu, H. Baer, H. Band, R. Barnett, E. Bergren and J. Beringer, *Review of Particle Physics*, Physics Letters B **667**(1-5), 1–6 (2008)

[2] M. Della Negra, L. Foa, A. Herve, A. Petrilli and The CMS Collaboration, CMS *Physics Technical Design Report*, Technical Design Report CMS. CERN, Geneva (February) (2006)

[3] F. Sauli, *Principles of operation of multiwire proportional and drift chambers*, CERN-77-09 **3**, 92 p (1977)

1 Introduction

This experiment covers the simulation of particles crossing a caesium iodine (CsI) crystal with Geant4 and the comparison with real measurements. Geant4 is an important tool for particle physicists to simulate experiments with computers. It is developed at CERN and optimized for solving complex problems. Because of its high precision this toolkit is now also used for the simulation of experiments in space and medical science.

In the first part of the course an experiment will be set up. In this part cosmic muons will be measured with the help of a CsI crystal[1]. In the second part a Geant4 program for the simulation of the specific experiment will be programmed. For the given detector the pulse height spectrum of cosmic muons crossing the crystal has to be simulated. In the analysis the real measurement and the result of the simulation has to be compared to each other.

---

[1] The crystal is a spare part of the calorimeter of the BaBar experiment. The experiment is located at the American particle accelerator SLAC (Stanford Linear Accelerator Center)

2 Theory

2.1 Background

2.1.1 Particle Physics Book

The main theoretical background of this experiment should be learned with the help of the Particle Physics Books. This book is published by the Particle Data Group and thus is most of the time abbreviated to PDG. You can find the online version at `pdg.lbl.gov`. Information about the theory of cosmic rays can be found in chapter 24[2]. The basics of the calorimeter used in this experiment you can find in chapter 17 (Passage of Particles Though Matter) and chapter 28.10.2 (Particle Detectors: Hadronic Calorimeters). Most of these contents were covered in the introductory course for the laboratory class.

For this experiment you should acquaint yourself with the following topics:

- cosmic rays
- the interaction between cosmic rays and Earth's atmosphere
- the evolution of air showers
- the interaction between charged particles and matter
- hadronic calorimeters
- photomultiplier tubes (PMT)
- absorption

2.1.2 The Landau Distribution

Charged particles going through thin absorption layers lose energy due to the excitation and ionization of atoms. The mathematical distribution for the best description of the average energy loss is in this case a strong asymmetric Landau distribution (Figure 1). A good approximation of this distribution is:

$$f(\lambda) \approx \exp(-\frac{1}{2}(\lambda + e^{-\lambda})) \tag{1}$$

The parameter λ reconsiders the difference between real and most probable energy loss.

$$\lambda = C(\Delta E - \Delta E_w) \tag{2}$$

Here ΔE is the real and ΔE_w the most probable energy loss. The parameter C reconsiders the energy needed for the production of electron-ion pairs for each length unit and the thickness

---

[2] Note! There is only an abridged version of chapter 24 (Cosmic Rays) in the often used **Particle Physics Booklet**, the digested publication of the **Particle Data Group**. Please use the **complete version** you can find for example on the web page.

of the penetrated matter.
Further information about the Landau distribution due to thin absorption layers you can find at chapter 2.6 “Energy Loss Distribution” in [3][3].

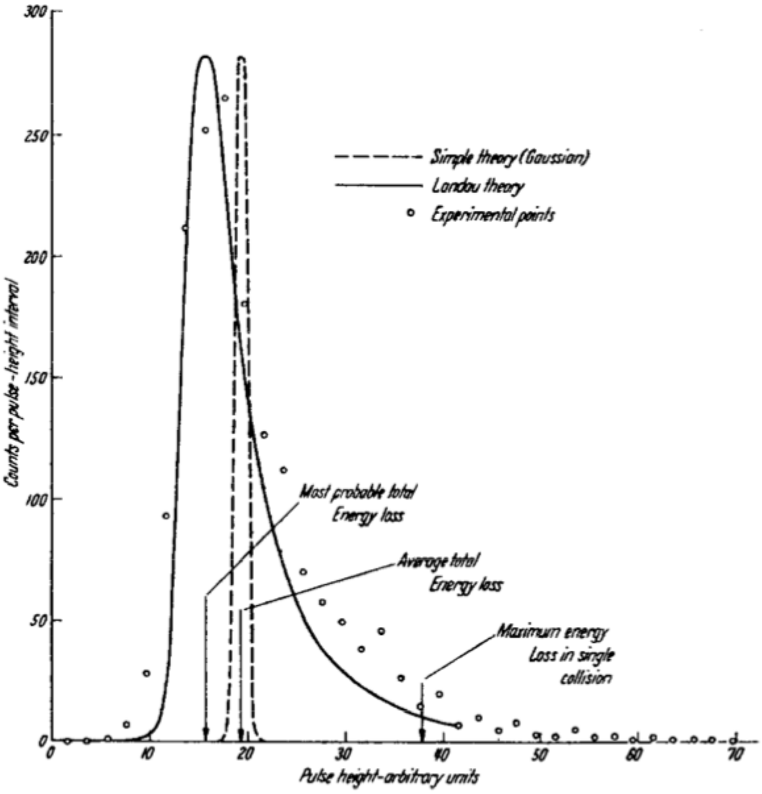

Figure 1: The Landau distribution. Plot from [3].

2.2 Control Questions

To test yourself, to prepare yourself for the discussion before the experiment and as a guidance through the PDG, these questions could be useful:

1. What are cosmic primaries and cosmic secondaries?

2. What is the main component of cosmic radiation?

3. How can you parameterize the energy spectrum of cosmic rays?

4. What happens in the interaction of cosmic primaries with Earth’s atmosphere?

---

[3] For example on the CERN document server under reference number 117989.

5. What particles are produced in an air shower? What is the dominating part at the height of Aachen?
6. What is the angular dependency of the muon spectrum from air showers at Earth's surface? What is the energy of these muons in Aachen?
7. What effects cause the energy loss of muons during propagation through matter?
8. What is the radiation length?
9. What is the average radiation length of muons in CsI?
10. How can you parameterize the energy loss of charged particles in matter?
11. How can you describe the average deviation from the original flight direction of a muon after passing through matter? (extra question)
12. What is the working principle of a calorimeter?
13. How does a PMT work?

Of course the other parts of this instruction manual are prerequisites for the discussion of the laboratory course, too.

2.3 Geant4

GEANT4 is short for “Geometry and Tracking 4“. It is a toolkit for the simulation of particles propagating through matter and their detection.
GEANT4 is written in C++ and an object oriented package of libraries. With them even complex constructions like the experiments of the LHC at CERN can be simulated as a whole.

The Large Hadron Collider

The main motivation for the LHC project and with this the huge detectors is the study of the breaking of electro weak symmetry for the Higgs mechanism. The experimental study of the higgs mechanism explores the consistency of the standard model at energy ranges up to above 1 TeV.
Also. there are many other reasons motivating the research at the TeV scale. Possible alternatives or additions to the standard model predict symmetries, new interactions and new particles. Its possible that important discoveries which give indications to a grand unified theory (GUT) are in the reach of the LHC. The expected discoveries at the LHC and its detectors (on of these is the CMS experiment in which the RWTH is also involved in) can appear in the form of super symmetric add-ons to the standard model or extra dimensions, which would imply a modified gravitational force at the TeV scale. In the past it was shown, that hadron accelerators are very good for the study of new energy ranges.

Geant4 at the CMS Experiment

While the CMS experiment was in its planing stage, the performance of the single detectors was simulated with GEANT4. Simultaneously feedback from this test environment was provided to the programmers to enhance the toolkit. Already at the phase of construction and before the start of data taking multiple physicists around the globe worked with results which were expected from a calibrated and running experiment. Their analyses were written for data which were simulated with computers under the preconditions of the experiment.

For the data generation, datasets are produced with Monte Carlo generators (e.g. PYTHIA[4]). As a form of beam collision, these datasets deliver the theoretical backbone. In the next step the propagation of the particles produced in the collision through the complete detector is simulated. Included in this are all of the sub-detectors like the tracker, the calorimeter (electromagnetic and hadronic) and the muon chambers. This simulation is performed by GEANT4, in which the complete detector is replicated as a precise virtual representation.

The result of this simulation serves as a base for analyses. In these conclusions on the initial and intermediate state are drawn from the end states measured in the different virtual subdetectors. If the results in respect to the input parameters are satisfying, this analysis can later be applied to the real data.

How the simulation works in detail, how the different sub-detectors were verified with GEANT4 and even more about this subject you can find in the CMS `Physics Technical Design Report (Volume I: Detector Performance and Software)`[5] in Chapter 2.5: `Simulation` [2].

Structure of the Toolkits

At this point only a rough sketch of the structure of GEANT4 shall be given. Please use the secondary literature listed at the end of this section to get more information.

A GEANT4 simulation is constructed in multiple steps.

The first step is the definition of the `world volume`, the world in which all other elements are placed. Here the detector (absorber) and the trigger can be defined. They can be activated as active volume and red out.

The material of all the components should be defined. It can be created from formerly defined elements with these material property.

In a `physics list` single particles can be defined or can be loaded from former projects.

The `primary generator action` defines which particles cross the whole environment and under which angles and energies to simulate them.

You can find more about GEANT4 on the official web page of Cern under `http://geant4.web.cern.ch/geant4/`. The `User Documentation` and the `Getting Started` have to be pointed out in particular.

---

[4] PYTHIA is a Monte Carlo generator created for the usage in particle physics. More information about PYTHIA you can find on `http://home.thep.lu.se/~torbjorn/Pythia.html`.

[5] PDF version: `http://cern.ch/aherten/praktikum/CMSTDR1.pdf`

3 Execution

3.1 Preparation

The crystal that is used for the measurement is already connected to a PMT and together with it placed in a brass housing. During the laboratory course, this housing should not be opened. Pictures of the calorimeter can be found in the appendix (Figure 3).
In the laboratory the calorimeter should be placed into a metal frame. Also, two triggering scintillators and various lead bricks are provided for this experiment (Figure 2).

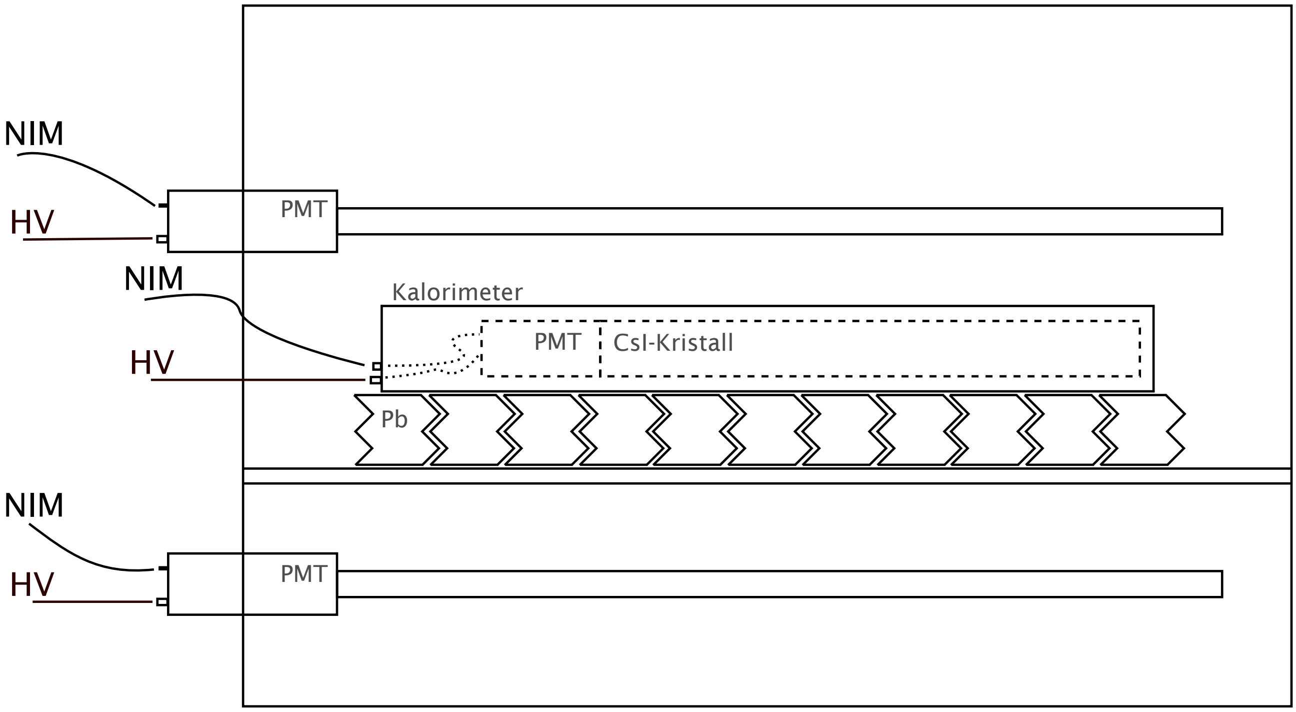

Figure 2: Schematic of the experiment with (top to bottom) top triggering scintillator, calorimeter, lead bricks and bottom triggering scintillator.

3.2 Interconnect the Components

All of the used PMTs are driven by a negative high voltage (HV) of maximum 1900 V. Individual PMTs behave differently and an adjustment off the different voltages may be required.
The signal output of the PMTs is readout over a NIM crate[6].

---

[6] NIM is an international norm for the regulation of electronic modules. The NIM modules used in the laboratory class utilize NIM pulses, which interpret a voltage of -0.7 V as a logical 1 and a voltage of 0 V as a logical 0.

3.2.1 Schematic of the Circuit

Goal of this part of the experiment is the measurement of muons which completely penetrate the experimental structure top to bottom while they deposit energy in the calorimeter.
Following NIM plug-ins are provided:

- Discriminator
- Gate-Delay-Generator
- Coincidence
- Amplifier
- Multichannel Analyzer (MCA)[7]

Furthermore, an oscilloscope is located in the vicinity of the experiment. With it you can (and should) verify every step while interconnecting the hardware.

3.3 Measurement

For the measurement a PC with the software Genie 2000 is provided. Also two radioactive compounds (Na-22, Co-60) are available in the laboratory.
While the measurement is taken, you can observe the low event rate of the experiment. It can take a while until you can see a pulse height spectrum of the deposited energy.
The measurement will run until the next day.

3.4 Geant4 Simulation

3.4.1 Installation of the Geant4 Libraries

The installation of GEANT4 is not trivial, especially outside of the Scientific Linux environment. Because of this, we advise you work in the CIP-Pool.

Files for the Laboratory Course

You can login into your account in the CIP-Pool (room 28A 202, 203 and 204).
Create a new folder (e.g. named `geant4`)[8]. This will be your workfolder from now on. On the webpage for the laboratory class[9], you can find the necessary files for this part of the experiment. Please download and extract them into a subfolder of the GEANT4 workfolder created in the last step[10].

---

[7] 14-bit ADC (16384 channels)

[8] For example with the following shell command: `mkdir $HOME/geant4`

[9] `http://www.institut3b.physik.rwth-aachen.de/go/id/gblg/?lidx=1`

[10] For example by entering `tar -xzf softwareT18.tar.gz` into the shell.

Initialization of the Geant4 Environment

To use Geant4, you have to set some environment variables. The easiest way to do so, is to load them in your bashrc. Open the file named `.bashrc` in your home directory (mind the dot in the beginning, it is a hidden file) and enter the following lines:

```
# use gcc for c++11 with Geant4.10.2
export LCG_ROOT=/cvmfs/sft.cern.ch/lcg/contrib
source ${LCG_ROOT}/gcc/4.9.3/x86_64-slc6/setup.sh ${LCG_ROOT}

# Geant4 Version 10.2.p02
source /net/software/g4rt/sl7/10.02.p02/geant4.10.02.p02-install/share/
    Geant4-10.2.2/geant4make/geant4make.sh > /dev/null
```

If you have an open terminal, close it afterwards. Every newly opened terminal will find all necessary variables.

3.4.2 Usage of Geant4

Before we start with the explanation of the present code base, some general remarks about the usage of the toolkit.

Coordinate System

Geant4 uses a special coordinate system. All of the size definitions assigned to one object are applied in both directions. Thus the point of origin is always located in the middle.
If for example one wants to create a box with a length of 20 cm, the appropriate parameter in the code has to be set to $x_{\text{Box}} = 10 * \text{cm}$. The x axis points from left to right in horizontal direction, the z axis points from the left side out of the screen and the y axis points vertically up.
A code example for the creation of a box of 4 times 6 times 9 cm^3 is:

```
G4Box* Box = new G4Box("Box1", 2.0*cm, 3.0*cm, 4.5*cm);
```

As you can see it is necessary to append a unit of measurement in Geant4. Necessary conversions are performed independently by the toolkit. This means you can also write 0.02*m instead of 2.0*cm.

Compilation, Execution and Visualization

Geant4 is a toolkit which contains commands for the simulation of particle tracks and is based to 100 % on C++. Like with any other C++ program the simulations created with Geant4 have to be compiled.
The name of the program is declared in the file `CMakeLists.txt` Additionally to this, there are further *flags* and links set in this file to embed the Geant4 libraries into the C++ environment.
To compile your program, enter the (at the beginning empty) build directory and type the following command in a shell:

```
cmake ..
```

That adds a lot of files into your build directory. This includes a makefile which you have to run with

```
make
```

to compile your program. Now you can execute it with

```
./e166calor
```

You wont be able to compile the program before you did some of the following implementation tasks.

To display and visualize the geometry and particle tracks GEANT4 utilizes different implementations. We use an interactive graphical output called OpenGL. If the program is executed it will open a viewer with a command line. You can set the parameters of the program with various commands and start and control the graphical output.
By entering `help` you get an overview of the possible control commands.
Per default the commands stated in the macro files `ExecutionMacro.mac` and `Viewer.mac` are executed at runtime. With the interactive prompt you can now overwrite the default values. For example with

```
/vis/filtering/trajectories/create/chargeFilter
/vis/filtering/trajectories/chargeFilter-0/add -1
```

only tracks of negative charged particles are drawn. Other useful commands can be found out commented in the `ExecutionMacro.mac`. Perhaps the most important input is the activation of the virtual particle beam. With

```
/run/beamOn 10
```

you can activate the beam and simulate ten events with the `ParticleGun`. With the internal tools of the program you can manipulate the 3D visualization. You can move, rotate and zoom the object.
In the left part, you can display and hide single parts of the simulation. This is true not only for elements of the set-up including world volume, but also for every single particle beam.
The measured energies are automatically written into the file `absorber_N.dat`[11] in the directory `data`. Note that this file is overwritten if you run your program again. If a high number of events (> 1000) is simulated, displaying all particle trajectories can take a long time. It is recommended to simulate larger numbers of events not before the programming of the simulation is finished and use smaller test runs (100 events) to test the functionality. When you start your simulation with a larger number of events, disable the drawing with the command `/vis/disable` in the file `ExecutionMacro.mac`.

Refinishing

In principle the post processing for the different exercises can be done with every suitable program. We advise you to use the data analyzing tool ROOT[12] which has become a de facto

---

[11] Here `N` is the number of times the program was called (0, 1, ...).

[12] More about ROOT you can find on http://root.cern.ch.

standard in particle physics.
To convert the `absorber_0.dat` into a ROOT data format, a file named `display6.C` resides in the same directory like the program. With

```
root -l display6.C
```

you can display the histogrammed energy distribution. The histogram is initialized under the name `peakhist` and can be modified with the interactive ROOT shell. Have a look at the contents of `display6.C`.

3.4.3 Programming Exercises

An integral part of the experiment is to program a virtual representation of the experimental set-up with GEANT4. Please complete the simulation following the guideline below.

Note: There are comments in the code to point out where the programming exercise has to be implemented. The comments have the form `//### TASK:`.

Construct the Geometries - `E166CalorDetectorConstruction.cc`

Interesting for the set-up of the experiment is the file `E166CalorDetectorConstruction.cc`. In it the size of individual objects and their physical properties are defined, as well as their placement in the set-up.

- **Object Creation**
  In the upper part you can find an example for the definition of the world volume all the other volumes are subordinated:

  ```
  WorldSizeX = 1.1*cm;
  WorldSizeY = 80.*cm;
  WorldSizeZ = 22.5*cm;
  ```

  Please reinsert the commented lines and assign the correct values to the variables. What values are to choose here?
  In addition to the world volume three variables each have to be defined for the crystal (AbsorberX, AbsorberY, AbsorberZ) and the trigger (TriggerX ...)[13]. Here the values from the specific components of the experimental set-up have to be entered. Estimate the size of the crystal; the precise values will be measured in the course of the experiment. The brass case will not be reconsidered in the simulation.

- **Defining the Material Properties**
  The material creation happens in three steps: First the elements are defined, after that a material and its density and at last the elements are added to the (up until now blank) material.

---

[13] Here it is enough to construct the triggering scintillator only once. Since both scintillators are identical in construction, it will be inserted two times.

  - Defining the Elements
    Define according to the definition of oxygen

    ```
    a = 16.00*g/mole;
    G4Element* O = new G4Element(name="Oxygen", symbol="O", z=8., a);
    ```

    the needed elements. Apart from oxygen these are caesium, iodine, carbon, sodium, nitrogen and hydrogen.

  - Material Creation
    With the help of the `G4Material` object create the involved materials of the three components crystal, triggering scintillator and surrounding air. Following the definition of the scintillator material

    ```
    density = 1.032*g/cm3;
    G4Material* pScint = new G4Material(name="pScintillator", density,
      ncomponents=2);
    ```

    implement caesium iodine as `E166Calor` and `Air`. Use $1.290\,\frac{\mathrm{mg}}{\mathrm{cm}^3}$ for the density of air.

  - Adding Elements to a Material
    With the command `AddElement` elements can be added to the individual materials. Like in:

    ```
    pScint->AddElement(C, natoms=9);
    pScint->AddElement(H, natoms=10);
    ```

    Apart from the above defined `pScint` implement this for `E166Calor` and `Air`. The crystal is build up from five atoms caesium and iodine each, while air consists to 70 % out of nitrogen and 30 % out of oxygen. For the later one variables of the form `fractionmass=0.3` have to be used instead of the above `natoms=9`.

  - Analysis Exercise: Sodium Iodine instead of Caesium Iodine
    For this analysis you need sodium iodine as the crystals material. Prepare following lines for sodium iodine, but leave them commented for the time being:

    - a new `G4Material` (to keep it simple with the same name `E166Calor`)
    - one line with the appropriate `AddElement`

- **Placement of the Objects**
  In the code you can find following example for the placement of the world volume:

  ```
  solidWorld = new G4Box("World", WorldSizeX, WorldSizeY, WorldSizeZ);
  logicWorld = new G4LogicalVolume(solidWorld, WorldMaterial, "World");
  physiWorld = new G4PVPlacement(0,
              G4ThreeVector(0, 0, 0), //at (0,0,0)
              "World",                //its name
              logicWorld,             //its logical volume
              0,                      //its mother volume
              false,                  //no boolean operation
              0);                     //copy number
  ```

Following this schema add `solidAbsorber`, `logicAbsorber` and `physiAbsorber` for the crystal and `solidTrigger1`, `logicTrigger1` and `physiTrigger1` for the top, as well as with a 2 indexed for the bottom triggering scintillator.
With `G4ThreeVector(x*cm,y*cm,z*cm)` you can define the positions of the individual components. Contrary to the example from above (`physiWorld`) the crystal and the scintillators possess different, nonzero mother volumes. Think about what this can be and discuss it with your tutor.

Source Configuration - `E166CalorPrimaryGeneratorAction.cc`

In the experimental set-up muons are measured. They are not per se on hand for Geant4, but have to be programmed as a particle source. The following steps occur in the file `E166CalorPrimaryGenerat` `.cc`.

- **Selecting μ^+/μ^-**
  Geant4 uses predefined particles from a table of particles and adds them to the simulation. The source for this experiment are cosmic muons, which have the same probability to exist as particle μ^+ and antiparticle μ^-.
  Implement a program code that randomly (uniform distributed) throws particles or antiparticles.
  With

  ```
  particleTable->FindParticle(particleName="mu+")
  particleTable->FindParticle(particleName="mu-")
  ```

  you can choose a specific muon species. With the help of the command

  ```
  particleGun->SetParticleDefinition(particle)
  ```

  you can finalize the selection and hand the selected `particle` over to the `ParticleGun`.

  Tip: Die Root function `gRandom->Rndm()` provides random numbers[14]. More information regarding Root's `TRandom` package you can find in the documentation.

- **Energy Distribution of the Source**
  The cosmic muons have to be generated with a specific energy distribution. Here a Gaussian probability function with a mean of 4 GeV and $\sigma = 0.5$ GeV should be assumed.
  With

  ```
  particleGun->SetParticleEnergy(primaryEnergy*GeV);
  ```

  you can set the particles energy. Implement `primaryEnergy` like described.

---

[14] Since Root and Geant4 are both toolkits based on C++, you can use your Root packages without any problems. But keep in mind to include the specific package in the document header. Your tutor will be glad to advise you if necessary.

- **Angular Distribution of Entering Particles**
  In question six of subsection 2.2 you studied the angular distribution of entering muons at the Earth's surface.
  Implement this distribution by defining the correct `xCoordinate`, `yCoordinate` and `zCoordinate` with the following `set` call:

  ```
  particleGun->SetParticleMomentumDirection(G4ThreeVector(xCoordinate,
   yCoordinate, zCoordinate));
  ```

  Note: Choose a $\cos^2$ distribution for the angular distribution.

- **Positioning of Particles in the Sky**
  After you programmed the angular distribution, you have to position the particles uniformly on the virtual sky. This can be done with:

  ```
  particleGun->SetParticlePosition(G4ThreeVector(XGun*cm, height*cm,
   ZGun*cm));
  ```

  Implement the uniform distributed variables `XGun` and `ZGun`. Choose a sensible value for the fix `height`.

Arrival Direction - `E166CalorEventAction.cc`

As a check the arrival direction of the muons at the absorber, is saved to the output file.

- **Calculating the arrival angle**
  The arrival angle of the cosmic muons need to be calculated with respect to the chosen coordinate system in the detector construction. For this purpose you should use the coordinates of the arrival direction

  ```
  HitMomentum.x(), HitMomentum.y(), HitMomentum.z();
  ```

3.5 Start of the Simulation

Start the simulation of the CsI crystal and simulate an adequate number of particle tracks. Make sure you have a minimum of 1000 particles crossing the crystal and with this an equal number of lines in the `absorber_N.dat` file.

3.5.1 Analysis of Different Materials

Apart from the CsI you should also produce datasets for NaI and `pScint` for the active detector material `E166Calor`.

3.5.2 Analysis of Different Sizes

It is not possible to take precise measurements of the CsI crystals dimensions because of the surrounding brass housing. Based on your above estimation generate different widths for the CsI crystal[15].

4 Evaluation

After successful experimentation you have measured and simulated pulse spectra.
In the experimental part you should have measured the following spectra:

- the spectrum of cosmic muons measured with the CsI crystal
- possible additional measurements with artificial sources

In the simulation part, you simulated the following spectra:

- the spectrum of cosmic muons, with...
  - ...CsI for the active detector material and your estimation for the size of the crystal
  - ...NaI and plastic (material of the triggering scintillators) as an alternative detector material
  - ...CsI for the active detector material and different measurements for the size of the detecting crystal

Before you can work with the measured data you have to calibrate it. Use the Co-60 and Na-22 spectra to map the channels of the MCA to energies. Present your results in an adequate form (table or plot).
After successful calibration fit Landau distributions to the measured and simulated pulse spectra. Quote the parameters of these fits together with their error[16]. How well do your predictions and these spectra match? (χ^2)
Finally make a meaningful comparison between your measured and simulated pulse spectra, for example with a plot, a ratio or a difference.
Eventually determine the most probable crystal size by comparing the simulations of different crystal sizes with the real measurement.
Please explain in your protocol how you implemented the angular distribution of entering particles and additionally hand in the code of the programming exercises.

---

[15] Do not forget to write down the used values.

[16] For example with ROOT or PyLab.

A Appendix

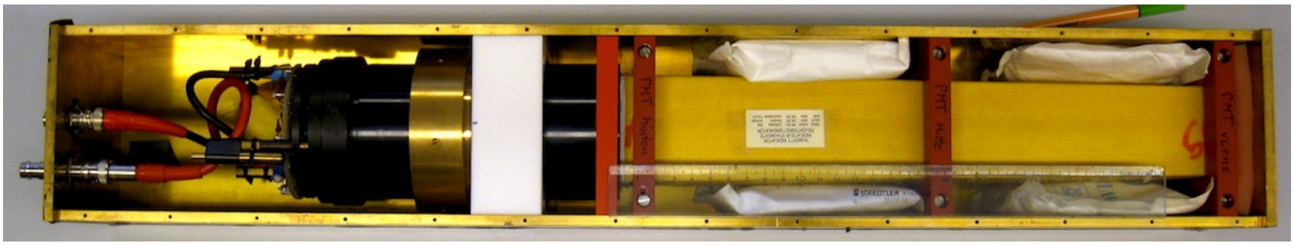

Figure 3: Picture of the detectors brass housing while opened. On the right side: the CsI crystal with a ruler; on the left side: the photo multiplier. To bind possible moisture a small package with silicone balls is inserted into the case.