Abstract

In this note we present the second part of the Full Simulation Package that describes the response of the NESTOR detector deployed in March of 2003 at a depth of 4000 m. This package includes the simulation of the specific functions of all the detector units and the operations of the Read Out and Data Acquisition system.

1. Introduction

In March of 2003, a NESTOR floor along with the sea bottom station was deployed at a depth of 4000 m. A NESTOR floor [1,2] is made of titanium and looks like a star with 6 arms. The diameter of the star is about 32 meters and at the center there is a large titanium sphere that houses the floor electronics (Figure 1). Each arm holds a pair of photomultiplier tubes (PMTs) one looking upwards and the other downwards. The 15” PMTs are enclosed in spherical glass shielding and the signals are collected from the floor electronics, where they are digitized by 5 ATWDs¹. When a trigger occurs, the PMT waveforms as well as other operational and environmental parameters are transmitted to the shore laboratory where they are recorded [3,4].

Figure 1

A Nestor floor during deployment.
In the 2003 Run more than five million events were recorded and analyzed in order to study the performance of the detector. The simulation tool that we used in the analysis was the Full Simulation Package (FSP), which describes in detail the detector architecture and all the physical processes sensitive to an underwater neutrino telescope.

The FSP consists of six stages:

- The simulation of the muon flux near the detector volume.
- The simulation of the interactions of the primary muon as well as its secondaries.
- The production of the Cherenkov photons that hit the PMT photocathode.
- The simulation of the detector response.
- The background simulation.
- The simulation of the Data Acquisition System.

In a previous note [5] we have described the three first stages of the FSP. In the following we describe the last three stages, which depend on the specific detector functionality (i.e. Photomultiplier tube characteristics, Data Acquisition System architecture).

The simulation is coded in Fortran or C++ and uses the software libraries developed in the European Center of Nuclear Research (CERN). The host operating system is Linux and for the simulation we have used the main computer farm of the Physics Lab of the School of Science and Technology in the Hellenic Open University.

This software package takes as input the Monte Carlo events produced by the first part of the FSP, which contain information about the Cherenkov photons that hit the photocathode of the photomultipliers.

2. Simulation of the PMT Response

The electrical pulses that are produced when one or more Cherenkov photons hit the photo cathode of the PMT are simulated taking into account:

- The collective and quantum efficiency of the PMTs.
- The transit time distribution of the PMT electronics, and

1 Analog Transient Waveform Digitiser
- The pulse height distribution for each PMT when one only photoelectron is emitted from the photocathode.

The quantum efficiency of the PMTs (the probability that a photon will produce the emission of an electron from the photocathode) depends on the energy of the incident photon as it is demonstrated in Figure 2 [6].

![Quantum Efficiency Graph](image)

**Figure 2**

*The quantum efficiency of the photo-cathode with respect to the photon energy.*
On the other hand the probability that this electron (called photoelectron) will reach the first dynode of the PMT (collective efficiency) depends on the position of the incident photon on the surface of the photocathode. The most efficient part of the photocathode is the top part (zero polar angle) and (according to the supplier’s datasheets) is 70.7%. The dependence of the collective efficiency on the polar and azimuth angles (with respect to the PMTs axis) has been measured at the lab by the NESTOR collaboration [6] and it is demonstrated in Figure 3 for three different PMTs.

![Figure 3](image)

*The collective efficiency for three different PMTs with respect to the polar angle of the incident photon.*

Due to the quantum and collective efficiency of the PMTs the number of photoelectrons that contribute to the waveform is much smaller than the number of photons that actually hit the photocathode (Figure 4).
Each photoelectron collected in the first dynode, contributes to the resulting waveform of the PMT. For example if n is the number of the (collected by the first dynode) photoelectrons which left the photocathode at times $T_i$ ($i=1,\ldots,n$) then the waveform of the PMT can be expressed as $V(t) = \sum_{k=1}^{n} U(t - T_k)$.

The function $U(t)$ describes the electrical pulse produced by a single photoelectron and can be approximated as $U(t) = g f(t)$, where $f(t)$ is the sum of two one-tailed gaussians with rise time 8 ns and falling time 15 ns respectively (see Figure 5), and $g$ is the height of the pulse.

The height ($g$) of the produced electrical pulse when a single photoelectron is emitted from the photocathode follows a Polya distribution:

$$P(g) = \frac{(a \cdot g/M)^a}{g \cdot \Gamma(a)} \cdot e^{-gM}$$
where the parameters $a$ and $M$ have been measured [7] at the lab for each PMT separately. This single photoelectron distribution is shown in Figure 6.

![Figure 5](image)

*Figure 5*

The function $f(t)$ that describes the waveform produced by a single photoelectron. The rise-time is about 8 ns and the falling time is about 15 ns.

![Figure 6](image)

*Figure 6*

The single photoelectron pulse height distribution of a PMT.

Finally the arrival time of an electrical pulse produced by one photoelectron varies according to a normal distribution (transit time distribution), which has also been
estimated at the lab using the LED calibration system [8]. For example in Figure 7 it is shown the transit time distribution of a specific PMT.

For a demonstration in Figure 8 it is shown the resulting pulse shape of a PMT where 4 photoelectrons contribute to the waveform. This is a special example where some of the Cherenkov photons arrive delayed with respect to the rest.

3. Simulation of the Background Light

The only background source, which contributes significantly to the experimental signal, comes from the K\(^{40}\) decay. Sea water contains a small amount of potassium (K\(^{40}\))...
0.038 ppm) which is composed of three isotopes, one of which is K\textsuperscript{40} (0.0118%), a radioactive isotope with a half-life of 1.28 x 10\textsuperscript{9} years. K\textsuperscript{40} decays mainly\textsuperscript{2} via beta decay according to the reaction:

\[
\text{^{40}K} \rightarrow \text{^{40}Ca} + e^- + \overline{v}_e
\]

The emitted electron from the K\textsuperscript{40} beta decay may have enough kinetic energy (above 240 keV) in order to produce Cherenkov photons in the water. This contribution is added to the simulation by producing more photoelectrons to each PMT, according to a Poisson distribution with mean the expected number of photoelectrons in the experimental time window. This rate has been measured during the 2003 Run for each PMT separately (see Figure 9) and is about 50 kHz [9]. Then these photoelectrons are treated in the same way like the photoelectrons produced with the full Geant Simulation.

![Figure 9](image)

*Figure 9*

*The measured counting rates of the 12 PMTs due to K\textsuperscript{40} decays.*

### 4. Simulation of the Data Acquisition System

The simulation of the Data Acquisition system can be divided in three steps:

1. The simulation of the trigger formation.

\textsuperscript{2} branching ration about 89%
2. The simulation of the response function of the electronics.
3. The data formation according to the raw data protocol.

4.1 Simulation of the trigger formation

When all the waveforms are generated, the next step of the simulation program is to form the trigger. This procedure simulates the function of the floor electronics, which form a trigger when more than N pulses (where N is the majority trigger logic) lay inside the experimental trigger time window. In the simulation, we examine how many pulses exist with height greater than the threshold value defined for each PMT\(^3\). Then we calculate the time of occurrence of the threshold crossing and by arranging these values in time we look for at least N pulses from different PMTs to lay within the experimental trigger window\(^4\) (W).

For example in Figure 10 it is shown the produced waveforms of a Monte Carlo event. There are 12 charts representing the electrical pulses of the 12 PMTs for the detector prototype deployed in March of 2003. The PMTs are arranged in six arms and in each arm there is a pair of PMTs one looking up (upper chart) and one looking down (lower chart). A solid line demonstrates the time that the pulses crossed the threshold value, while the experimental trigger window is demonstrated by the two dashed lines. The trigger logic was set to select events with four or more PMT pulses in the trigger window. In the Figure it is shown that the trigger signal is generated from PMT pulses, from the arm 1 and 2.

\(^3\) The PMT threshold values are defined to the input file.
\(^4\) The majority trigger logic (N) and the trigger time window W are defined by the user to the input file.
Figure 10
An example of the trigger formation of the simulation program. The solid lines indicate the time of the threshold crossing while the dashed lines the experimental trigger window. The pulses from the PMTs of the first and second arm form a 4-fold trigger.

4.2 Simulation of the response of the electronics

Apart from the transit time distribution, the arrival time of each pulse depends on the additional delays due to the electronic circuits. In addition all pulses are shifted in order to form the trigger at 197.5 ns within the experimental time window (0-465 ns) according to the corresponding function of the floor electronics. The additional delays have been estimated during the calibration procedure at the lab.

Furthermore, the electronic circuits and especially the passive delay lines deform the shape of the waveforms. As it is described in another note [10], the response function of the delay units can be estimated by means of Fourier transformations. Then by using this response function and the inverse Fourier transformation we can simulate the effect of the delay lines. For demonstration, in Figure 11 it is shown a PMT waveform before and after the simulation of the attenuation caused by the passive delay lines.
Figure 11

The effect of the delay units to the pulse shape. The black small dots represent the waveform before the delay lines and the red big stars the waveform after the delay line attenuation.

The next step of the simulation procedure is to generate the digitized waveforms of the trigger and the clock channels\(^5\) of the 5 ATWDs [11]. Due to a small interference of the clock channel with the signal channels the waveforms are a little bit deformed from the addition of crosstalk noise of the level of about 15 mV. This noise, which has been measured at the lab and has been parameterized for each specific ATWD channel separately, is added before the digitization procedure of the waveforms [10]. Then the waveforms of the PMTs are digitized according to the sampling rate of the ATWDs. In this step the gain and the offset of each ATWD channel is taken into account.

Finally the digitized signals are written to text files each containing 2613 events according to the Data Acquisition System specifications. These text files are processed by another program in order to produce binary files according to the raw data protocol.

\(^5\) The clock frequency of the Floorboard is 40 MHz
4.3 Formation according to the raw data protocol

The NESTOR Floor electronics generate 2560 words of data at the occurrence of a trigger. These data words include the outputs of the ATWDs and data from the environmental sensors.

In particular each of the five ATWDs of the Floor Board has four channels each giving 128 10-bit samples of the respective digitized pulses. These 10-bit words are sifted to the left to form \((5\times4\times128 = 2560)\) 16bit words. The six Less Significant Bits (LSB) of these words, are used to carry other information. The first two bits are check bits that prove the correctness of the received data and point out the beginning of each data packet and data block. The sixth bit is used to load the environmental data, the counter values (2, 3 and 4-fold trigger counters, event counter, PMT counter) and information about the status of the electronic boards.

A program named `raw.cpp` has been developed in C++ for the generation of data files identical to those produced by the Floor electronic boards. The program takes as input the text files produced in the previous step of the simulation and creates binary files. The input file starts with 2 bytes corresponding to the majority trigger level and goes on with 2 bytes for the coincidence window, two dozens of 2-byte words, corresponding to the PMT thresholds (12x2 bytes) and the PMT high voltage (12x2 bytes) and 11 4-byte words which contain Monte Carlo information. The last but most informative of the input files consists of repeated blocks of the digitized pulses of the 4 channels of the 5 ATWDs. Each block starts with 2 4-byte numbers, which correspond to the number of the ATWD and the number of the channel and goes on with the 128 2-byte samples. This is repeated 18 times (two channels of the twenty in total do not output any information and therefore are not needed at this step).

The program reads the values of the digitized pulses, converts them to gray code and then shifts the words to the left by 6 bits (remember that each sample of the ATWDs is 10-bit long, so there is no loss of information of course). It should be noted that information not used in the subsequent stages of analysis is random numbers since simulating the temperature for example or the ATWD working status does not fall within the scope of this analysis.
5. Program Execution

The software package uses the output of the first part of FSP; a text file containing information about the photon hits on the photocathode of the photomultipliers. The file is read from unit 92. The user has to supply some essential information by editing an auxiliary input file (named “inbot”), which contains:

- The rms of the transit time spectrum of the PMTs.
- The quantum efficiency of the PMTs.
- The PMT collective efficiency.
- The $\mu$-metal shadowing.
- The K$_{40}$ background rate (in kHz).
- The trigger multiplicity.
- The trigger time window (in ns).
- The voltage thresholds of the PMTs (in mV).

The program also uses a lot of parameter files containing information concerning the following functional characteristics:

1. The single photoelectron distributions of the PMTs
2. The parameterization of the interference of the clock and signal ATWD channels
3. The delays of the cables
4. The response function of the delay lines
5. The gain and the offset of the ADC channels
6. The sampling time interval of the ATWDs

The source code (simu.f) is compiled and linked by executing the script $\texttt{compisimu}$ (bash-shell commands) and the executable file is called $\texttt{simu}$. For example:

```bash
./compisimu
./simu
```

The output of the program is a set of text files starting from unit 21. Then the executable of the raw.cpp program reads each file separately and produces the binary raw data file in identical format with the real data. For example:

```bash
g++ -o raw raw.cpp
./raw fort.21 r_1_1
```
6. Conclusions

In this note we presented the second stage of the Full Simulation Package, which describes the response of the NESTOR detector deployed in March 2003. The input of this stage is the file produced by the first stage of the simulation package. The simulation generates electrical pulses according to the characteristics of the PMTs, and simulates the trigger and the digitisation procedures. The background contribution from the $^{40}\text{K}$ decays and the distortion of the PMT waveforms due to the electronics are also included in the simulation while the output file is formatted according to the Data Acquisition protocol.

Acknowledgments

The authors of this note wish to thank the members of the NESTOR collaboration, the stuffs of the NESTOR Institute as well as the academic and technical personnel of the School of Science and Technology of the Hellenic Open University for their help, scientific, technical and financial support.

References


