



Bridgeport Instruments, LLC, 11740 Jollyville Rd., Ste 500, Austin, TX 78759,
www.BridgeportInstruments.com

By Michael Momayezi
Phone 512-553-9933
Fax 512-553-9934
email momayezi
@bridgeportinstruments.com

MCA-3000 Users Manual

June, 2020

Summary

The MCA-3000 is a family of high-performance multi-channel analyzers (MCA) for measuring gamma-ray radioactivity with a scintillator.

The PMT-3000 includes a operating voltage supply and plugs directly onto a photomultiplier.

The SiPM-3000 includes the SiPM operating voltage supply and plugs directly onto an SiPM-array.

Using its embedded 32-bit ARM processor, it provides accurate gamma-ray spectroscopy and gain stabilization together with many advanced features such as automatic background subtraction and alarm computations.

This document describes how to use the MCA-3000 from the graphical user interface and from Python scripts. Since all commands are formatted as human readable JSON strings, developers will find they can program the MCA-3000 in any language they want.

Feature summary – PMT-3000

- More than 4Mcps histogramming rate
- Suitable for all scintillators via programmable integration time
- Positive and negative operating voltage and different PMT pinouts.

Feature summary – SiPM-3000

- Suitable for 500kcps on NaI(Tl) or faster scintillators
- Suitable for many scintillators via programmable integration time
- Very low-noise integrated SiPM voltage supply
- Very low trigger threshold: <10keV with 2MeV energy range.

Feature summary – Common characteristics

- Histogram: 4K×32; Loss-less two-bank mode with 2× 2K×32;
- List mode, pulse shape capture
- On the fly pulse shape discrimination
- Built-in programmable gain stabilization performed by embedded ARM processor
- USB interface; Serial interface on PCB; default 115200 baud
- FPGA customization possible for application specific product
- ARM C-code customization possible for application specific product
- Power consumption is only 300mW (5V@60mA)

Popular FPGA firmware options

- Loss-less dual bank list mode buffer with 2047 events per buffer.
- 100ms time slice operation with 20-deep buffer, events and 1k histogram per slice.

Popular software options

- Sample vs background measurements with alarm computation
- Radiation Portal mode with automatic background tracking and programmable alarm

Table of Contents

1. Supporting Documentation
3. Energy Spectrum
4. Sample vs Background
5. Mathematics of Errors and Alarms
6. Radiation Portal Monitor
7. Analog
8. Calibration
9. Gain Stabilization
10. Mechanical and Pinouts
11. Ordering

1. Supporting Documentation

Open-source software The software is open source and mostly written in Python. Bridgeport Instruments provides a software installer, which by default will create a C:\BPISoftV3 directory. That folder includes Python 3.7 with three added packages: ZMQ (www.zeromq.net), wxPython and Matplotlib v 3.2. ZMQ is used to implement the client-server behavior and it is accessible from more than 40 programming languages. Matplotlib is used for the graphics interface, and wxPython is used to create a traditional user interface with pull down menus.

The wxMCA folder that contains the MCA software can be placed anywhere on the hard disk. Inside is a folder called documentation and the best starting page is wxMCA/documentation/english/introduction/introduction.html which you can open in any web-browser. This set of linked documents contains a description of every control variable and every data field used by the MCA-3000.

2. Getting started

2.1 Using the USB interface

For USB communication the MCA-3000 uses libusb0.1 (Linux) and libusb_win32 on Windows 10. The Windows 10 libusb dll is digitally signed. When the MCA hardware is present during the installation, the installer will automatically link the MCA-3000 to libusb_win32.

However, it is also possible to simply copy the BPISoftV3 folder onto the C: drive and install the Windows USB-driver manually. This needs to be done only once, and Windows will later recognize any other MCA-3000, by its USB vendor ID of 0x1FA4 and the USB product ID of 0x0103 (PMT-3000) or 0x203 (SiPM-3000).

Open the Windows Device Manager and connect the MCA-3000 to a USB port. Wait until it appears in the Device Manager as an unknown device of type armMorpho or sipmMorpho.

Open the C:\BPISoftV3 folder and launch zadig-2.4.exe. Use Options/List All Devices to refresh the device list and select the device with the BPI vendor ID of 0x1FA4. To the right of the green arrow, select libusb-win32 (v 1.2.6.0) and click the big 'Install Driver' button below. Note that it may take up to 20 seconds for the screen to update - be patient. Once the 'Driver Installation Successful' message appears you can close the window. In the Device Manager you will now see the MCA-3000 listed under 'libusb-win32 devices'.

2.2 Launching the software

First launch the MCA Data Server. Look inside the wxMCA folder. Under Windows, double-click on run_mds.cmd. Under Linux, launch wxMCA/mds/mds_server.py making sure you use a python 3.6 or higher installation

that also includes ZMQ, wxPython and Matplotlib.

If the MDS reports no MCA found, the OS may have been slow in enumerating the MCA on the USB bus. Kill the MDS and launch it again.

Then launch the User Interface. Under Windows, double-click on run_wxMCA.cmd. Under Linux, launch wxMCA/wxGUI/MCA_Main.py

The items in the menu bar are mostly self-explanatory. You can view results and edit instrument settings in a spread-sheet environment. To send the changed values to the instrument, you need to click on File →to MCA.

3. Energy Spectrum

The MCA provides fast and accurate measurements of energy spectrum and count rates. To acquire an energy histogram, select Display→Histogram from the menu bar. In response, a histogram panel will open. On that panel use "New" to erase the old histogram and count rate data and begin a new acquisition. The panel does not automatically refresh, so click the "Refresh" button to get an updated energy spectrum.

Use "Save" to append the energy histogram and count rate data to a default data data file. The file name is automatically generated and is of the form xyz_histo_status.json where xyz is the detector serial number. By default that file is located in wxMCA/user/mca3k/data. Alternatively, use the Display→Save Histogram As to save the histogram where you want it.

3.1 Adjusting operating voltage

You need to adjust the operating voltage to set the maximum measurable energy.

Then adjust the digital gain to achieve the desired MCA calibration in keV per bin

The maximum measurable energy is independent of the number of histogram bins used. Users need to adjust the operating voltage to change the gain of the SiPM or PMT. In the table below we recommend maximum measurable energies for different applications.

| <i>Max. Energy</i> | <i>Comment</i> |
|--------------------|--|
| 1.0MeV | High gain for measuring low energies down to 20keV. Good for measuring contamination with Cs-137 and Cs-140 (fresh fission product), or Radium isotopes. |
| 2.0MeV | Measure NORM up to K-40 at 1460keV |
| 3.0MeV | Wide range covers all naturally occurring gamma-rays, even to Tl-208 at 2615keV |

Table 1: Cs-137 calibration for different tasks; NORM: Naturally occurring radioactive materials

How to determine the maximum measurable energy. In the **DSP** panel set **ha_mode** to 1 to measure the pulse height. The maximum measurable pulse height is 900. Start a new histogram and record the peak position (P) of a known isotope; eg E=662keV of Cs-137. The maximum measurable energy is computed as

$$E_{\max} = E \cdot \frac{900}{P}$$

Access the operating voltage control by opening the **Gain** panel, via a click on **Gain** on the dashboard. Enter a new operating voltage in the voltage field and hit enter. Note that the mouse must be inside the text input field for the software to accept that change. As a visual feedback you will see the little square in the bottom left hand corner of the panel change its color when the new entry has been processed.

Go back to the **Histogram** panel and click **New** to start a new acquisition. If you change the operating voltage by 1%, the PMT gain will change by about 6%. For the SiPM-3000 the gain equation is

$$\text{gain} = 1.0e6 \cdot (V - 28.6V)$$

Now adjust the digital gain to achieve the desired MCA calibration in keV per bin.

Once the operating voltage has been established, set **ha_mode** back to 0, and start a new histogram

Determine the measured peak position (P) and use the desired peak position (E) to determine the new **digital_gain**

$$dg_{\text{new}} = dg \cdot \frac{E}{P}$$

With that the detector has been calibrated. See the calibration section 8 for a much more detailed description.

3.2 Count rate measurement

Observe how the count rate accuracy improves with measurement time. The MCA reports count rates together with their statistical errors. This gives users a useful tool. Manually, or programmatically, they can end a measurement precisely when the desired accuracy has been reached, which saves time and money.

The error (in %) is computed from the number of events as $100 \cdot 2 / \sqrt{N}$, where N is the number of events. This is called the statistical $2\text{-}\sigma$ error. The true count rate lies within this error range with a 95% probability.

Note that the MCA corrects the recognized count rate using the known dead time per event. Hence the reported count rate is greater than the recognized count rate. The reported count rate is also an accurate estimate of the true number of counts per second. In fact the reported count rate should match the true input count rate with about 1% accuracy (systematic error) for input count rates up to 100kcps. The statistical count rate error is computed correctly using the number of recognized events.

3.3 Histogramming speed

The MCA has a non-extendable dead time equal to the hold_off_time per recognized event. For NaI(Tl) between 5°C and 65°C the MCA-3000 recommends a fixed hold off time for NaI of 1.2 μ s. The resulting throughput is shown in the next figure.

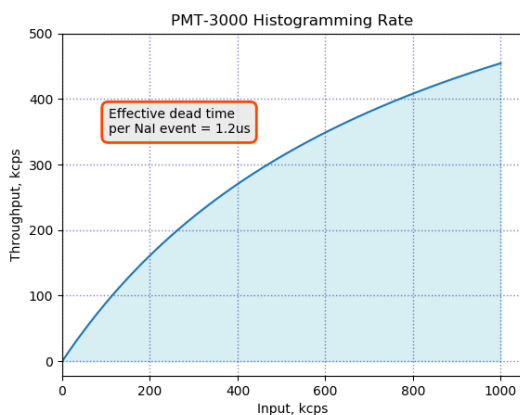


Fig. 1: Expected histogramming rate vs input count rate for a hold_off_time = 1.2 μ s; (ie typical for NaI(Tl) at room temperature.

The SiPM-3000 will match the speed of the PMT-3000 for NaI. For LaBr₃ the PMT-3000 can go even faster – at a fixed hold off time of only 0.3 μ s. The resulting throughput is shown in the next figure.

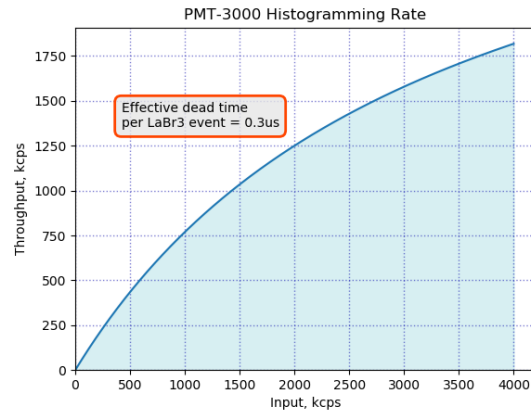


Fig. 2: Expected histogramming rate vs input count rate for a hold_off_time = 0.3 μ s; (ie typical for LaBr₃).

3.4 High count rates

There may be a small gain shift at high count rates. Users should be aware that at input count rates above a few 10kcps there may occur small gain shifts in the energy spectrum. Typically, at 50kcps the gain may fall by about 1%, but the exact gain shift will depend on the PMT that is used on the detector.

4. Sample vs Background

Non-standard software option

The instrument provides automatic background subtraction. To accurately measure the radioactivity of a weak source, or a weakly contaminated sample can be a complicated process. However, in all cases the process requires a precise measurement of the background and the ability to correctly subtract that background spectrum from the sample spectrum. And the MCA-3000 greatly simplifies this task.

The MCA-3000 stores sample and background data on the instrument and reports the difference. From the dashboard, open the "Sample – Bck" panel. From top to bottom you will see three spectrum displays. The one at the top is the background spectrum. Initially this will be empty. The middle panel has the sample spectrum, and initially it will show the previously acquired spectrum. Finally, the bottom panel has the difference

spectrum, which initially equals the middle spectrum.

To acquire a background spectrum, remove all unwanted radioactive materials from near the detector and click on "New" next to the top spectrum panel. You can increase the accuracy of the results by measuring the background spectrum 4 times longer than a typical sample spectrum. Click "Ref" next to the top spectrum to update the display.

Once you have counted the background long enough, click on "New" next to the middle spectrum panel. This starts a new sample spectrum acquisition and stops the background acquisition. The background spectrum will now have stopped and remain unchanged.

If you simply acquire a sample spectrum under the same conditions that you acquired the background spectrum, you will see a similar spectrum grow; click "Ref" next to the middle spectrum to update the display.

Every time the display is updated, the software reads three spectra from the MCA: the background, the sample and the difference spectrum. You will notice, that even the absence of any new radioactivity the difference

spectrum is not exactly a flat, empty spectrum. The reason for that is that the gamma-rays arrive randomly and there is therefore a natural statistical variation between acquire spectra.

The instrument computes probabilities and alarms. The MCA computes the probability that the count rate in the difference spectrum was caused by nothing but background. The message box next to the difference spectrum shows the result of that computation. You can confine this computation to a region of interest (ROI) by adjusting `roi_low` and `roi_high` in the alarm panel. For instance you can increase the sensitivity to Cs-137 by setting [`roi_low`, `roi_high`] to [580, 780]. The ROI is entered as keV and the software converts that into MCA bins using the `keV_bin` variable from the `autocal.json` file.

Finally, you can set an alarm threshold. If the probability that the measured sample radioactivity is caused by nothing but background is too low, the MCA can raise an alarm. The alarm threshold can be set in the alarm panel via the `alarm_thr` variable. For example, a value of $1.0e-3$ is interpreted as a probability of 1 in 1000.

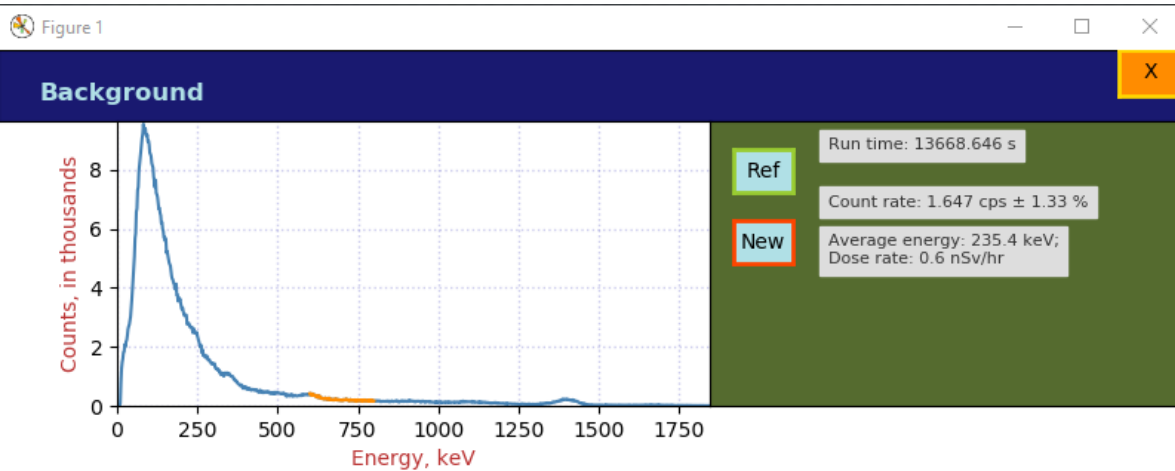


Fig. 3: A background histogram

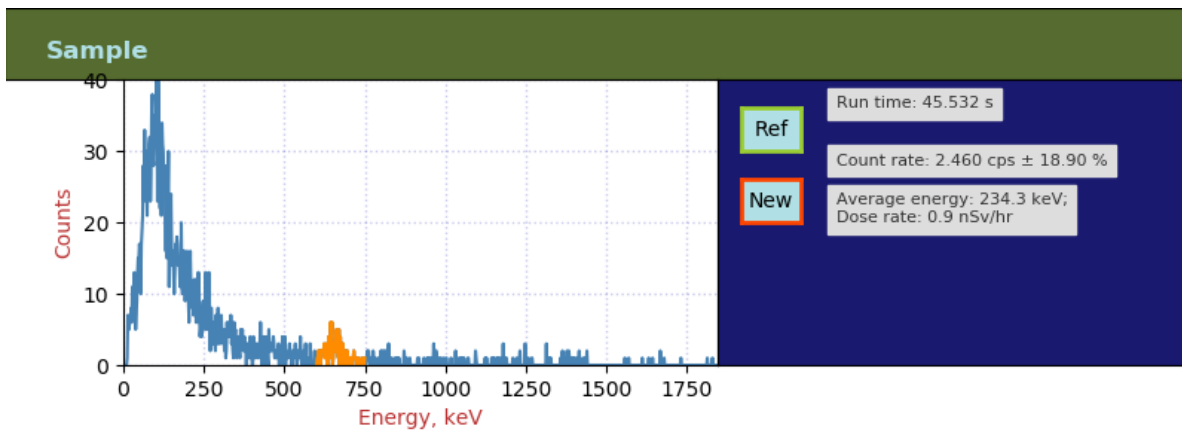


Fig. 4: A sample histogram with a 162kBq (4.4µCi) Cs-137 at 2.5m distance.

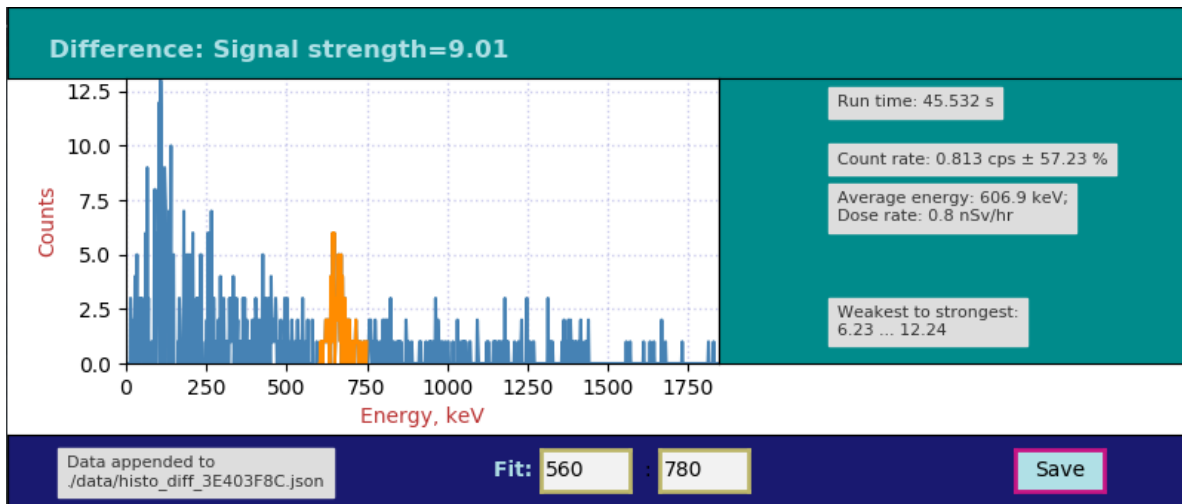


Fig. 5: The back-ground subtracted histogram with an indication of the signal strength. Even though many bins will have negative values, only the positive values are shown for clarity. In this case, the signal was a 162kB (4.4µCi) Cs-137 source at 2.5m distance.

The instrument reports the signal strength and confidence interval. The excess of count rate over background caused by a small amount of radioactivity is called the signal. And like a weak radioactivity signal embedded in noise, the signal strength is what the instrument reports. Here the strength is reported as the negative logarithm of the above-mentioned probability. The instrument also reports the 95% confidence interval for the measured source strength. Users can thus make an informed choice on how to set alarm levels and thresholds for taking a decision. The mathematics is described in the math box below.

5. Mathematics of Errors and Alarms

This section is for the reader with an interest in the mathematics that is used by the instrument.

5.1 Count rates and their errors

When measuring count rates, the instruments counts events and the elapsed time. Systematic errors for measuring the run time are very small and are ignored here. The dominant variation come from the fact that the number of events during a given time interval is Poisson-distributed. When the total number of events counted is greater than 100, we can approximate the resulting Poisson distribution with Gaussian normal distribution with an average of (μ = number of events) equal to the number of events and a standard deviation of $\sigma = \sqrt{\mu}$ For a given measurement the instrument reports the 2- σ relative count rate error

$$\epsilon = 2/\sqrt{\mu}$$

Dead time does not contribute to statistical error. The instrument knows the incurred dead time per recognized event and computes a live

time as run time minus dead time. The reported count rate is events/live_time.

5.2 Background subtraction

Background time can be different from sample time. The instrument can subtract a background spectrum from a sample spectrum even when the two have been acquired for different amounts of time. The calculation for the difference histogram bins is

$$diff = sample - \frac{sample_time}{back_time} \cdot back$$

Here, *diff*, *sample*, and *back* are the histogram bin contents of the three energy spectra.

Count rates are measured by summing all events within the alarm region of interest, and dividing by the respective live times. Count rate errors are computed from the number of events in the region of interest.

The difference count rate is computed as a direct difference, $R_D = R_S - R_B$, from the dead-time corrected sample and background rates.

The count rate error for the difference spectrum has to be computed from the two uncorrelated errors of the sample and the background counting:

$$\epsilon_d = \frac{\sqrt{(R_S \cdot \epsilon_S)^2 + (R_B \cdot \epsilon_B)^2}}{R_S - R_B}$$

Notice that for very small difference count rates the resulting relative error can be quite large. Further, a difference rate with a large relative error, eg $1.0\text{cps} \pm 200\%$, simply means that the result is compatible with a difference of 0cps.

5.3 Computing probabilities

For a given sample measurement time, T_S , the MCA knows how many background events to expect on average: $N_B = T_S \cdot R_B$. It compares that number to the number of actually measured events, N_S . Using correct Poisson statistics, not a Gaussian approximation, it calculates the probability that N_S could have been caused by the known background: $P = P(N \geq N_S | N_B)$. If $N_S \gg N_B$, this probability will be very small. If it falls below the given alarm probability

(**alarm_thr** in `arm_ctrl`), the MCA can raise an alarm. In other words, a stronger signal causes a lower probability.

However, a user might prefer to use a measure of the signal strength that increases with the signal strength. Hence we define a signal strength as $A = \log_{10}(1/P)$

For the confidence interval of the signal strength, the instrument reports a narrower measure. It computes the two opposites we get by assuming $1-\sigma$ errors in the background count and the sample count, but pointing in opposite directions. The resulting rectangular two-dimensional confidence interval contains about 71% of all cases.

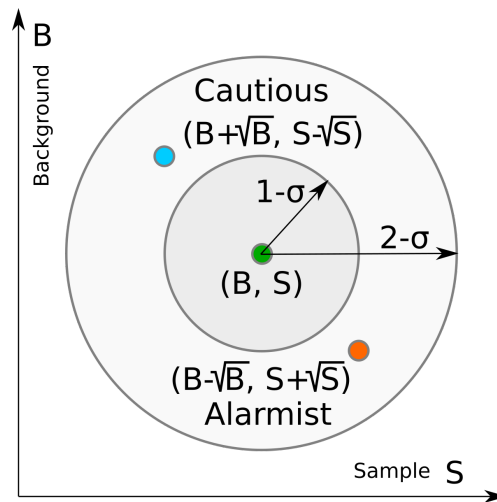


Fig. 6: Two dimensional confidence interval.

6. Radiation Portal Monitor

Non-standard software option

The MCA can act as a radiation portal monitor with background tracking and a programmable alarm function. In a Radiation Portal Monitor (RPM) the data acquisition unit must perform a number of tasks:

- Continuous background measuring;
- Deliver an alarm within a few seconds after the time of closest approach by a source;
- Keep a few seconds of alarm history so that a polling host will not miss an alarm;
- Automatically reset the system on a continuous alarm to remain operational;
- Support a programmable alarm threshold based on a false-alarm rate;
- Support an adjustable background averaging time, eg 30s for a big gamma-ray detector vs 5 minutes for a neutron detector,
- Be able to recognize passing sources without the aid of an occupancy detector.

The MCA-3000 implements all that functionality within its embedded 32-bit ARM processor. An alarm can be sent as digital pulse with a programmable width, from 1 μ s to infinity. Alarms can also be read via USB or the serial communications interface.

6.1 Theory of operation

In the discussion below, times are given in units of time slices. Typically a time slice is 100ms long and the processor perform one RPM computation step every time slice. In some software versions, time slices can be set in multiples of 50ms.

At first the instrument measures the background. On start, after power on or a reset, the ARM processor begins to measure backgrounds. Until backgrounds are know with sufficient precision, the ability to alarm is disabled. This wait time is user programmable, cf **wait** in the RPM panel. Typically the wait time is a fraction of the background averaging time, eg 20% to 50%. A shorter wait time yields an active RPM earlier at an elevated risk for a false alarm until a full background averaging time has passed.

As long as there is no alarm, the events counted during one time slice are considered background events. Using the region of interest (**roi_low, roi_high**) in the RPM panel allows the user to constrain the attention on a part of the energy spectrum.

The background counts per time slice are averaged using geometric averaging. Using the $w = 1/Bck_avg$ from the alarm panel the processor applies the formula:

$B_{n+1} = B_n + w \cdot (N - B_n)$). The background averages are then stored in a 128-long FIFO. This way the instrument can look back 128 time slices to find an untainted background after an alarm has occurred.

We have $0 \leq w \leq 1$ for the weight. If the background at time $t = 0$ changes from B_0 to B_1 , The background average responds to a step function with an exponential function of $B(t) = (B_1 - B_0) \cdot (1 - \exp(-t/\tau))$ where $\tau = 1/w$. The standard deviation of the averaged background, ie its noisiness, improves as if $1/w = Bck_avg$ samples had been averaged.

What happens when a passing source causes an alarm? The instrument continuously computes a moving window sum of the last L time slices and compares that sum to the number of expected background counts during L time slices. For every time slice the instrument computes the probability that the observed counts could have been caused by the known background, cf the mathematics section above. It computes $P = P(N \geq N_L | N_B)$, with $N_B = L \cdot B$. If that probability is less than **epsilon** on the alarm panel, it will trigger an alarm.

An alarm is typically raised no later than $L/2$ time slices after the closest encounter.

For a very strong signal, that alarm may be raised right at the leading edge of the L period; for a weak signal the alarm may be raised about $L/2$ time slices after the source has passed the point of closest approach. In many applications the allowed latency $L/2$ is about 2 seconds, which allows for a 4s summation time. At a time slice length of 100ms this means $L = 40$. In those applications, the time during which the radiation signal is detectable is 4s to 8s and and simulations show that 4s to 6s summation times produce the highest sensitivity; ie lowest minimum detectable activity.

No missed alarms. The instrument keeps a history of the alarm status with a maximum length of 128 time slices. This is controlled by the **history** parameter in the alarm panel. A polling device, via serial interface or USB, will be informed if there was an alarm present in the last **history** time slices. This way a polling host may have a latency of 128 time slices (12.8s) and still will not miss an alarm.

As long as there is an alarm present in the alarm history FIFO, the instrument will suspend background updates and use the oldest background average in its memory as the best estimator of the background.

Automatic device reset ensures the instrument remains functional when the background suddenly increases. Consider the case of a radiation detection backpack. The wearer walks into a room where radioactive material is present, and this causes an alarm. After **history** time slices, the instrument automatically resets and starts to accept the elevated radiation level as the new background. After a **wait** period (30s typically) the instrument will again be ready to alarm if the operator suddenly encounters an even higher radiation level.

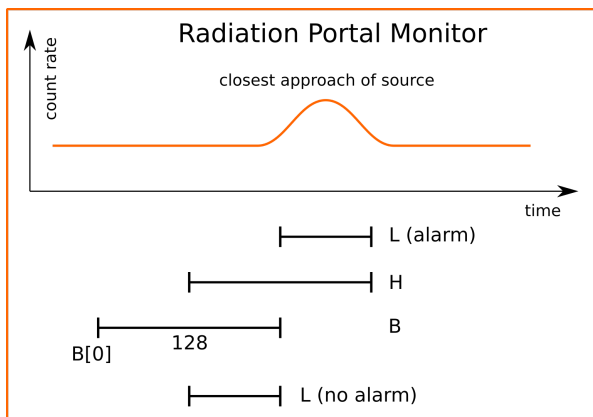


Fig. 7: Programmer's model of the Radiation Portal Monitor software.

7. Analog

7.1 PMT-3000

The PMT-3000 combines a PMT operating voltage supply and an MCA data acquisition board with a structure as shown in the figure below.

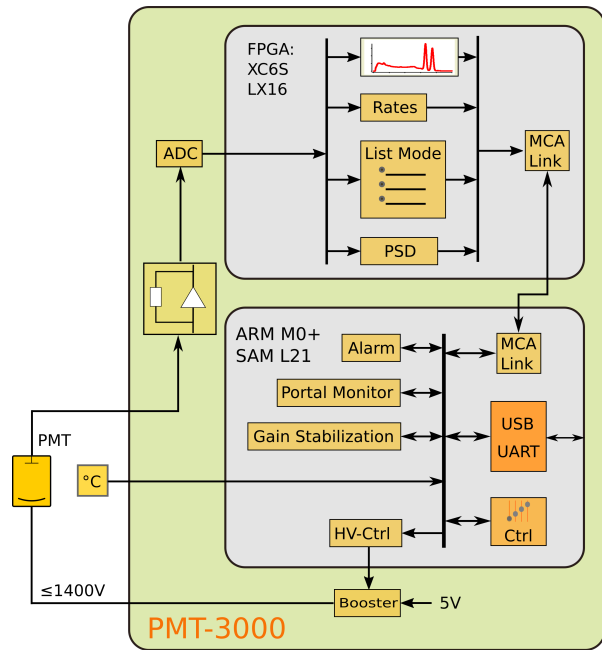


Fig. 8: The components of the PMT-3000.

The analog input of the PMT-3000 uses an I-to-V converter with four programmable gain resistors as shown in the figure below. A gain_select of 0, 1, 2, 4, and 8 creates a transimpedance of 100Ω, 430Ω, 1100Ω, 3400Ω, and 10100Ω, respectively.

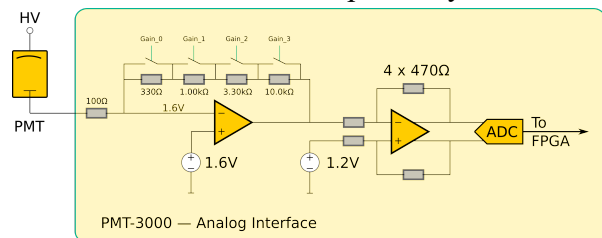


Fig. 9: The input amplifier of the PMT-3000.

7.2 SiPM-3000

The SiPM-3000 combines a very low-noise SiPM operating voltage supply and an MCA data acquisition board with a structure as shown in the figure below.

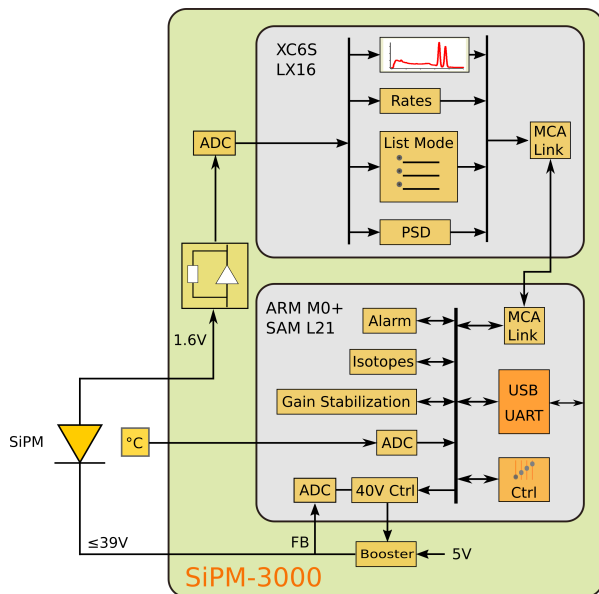


Fig. 10: The components of the SiPM-3000.

The analog input of the SiPM-3000 uses an I-to-V converter with four programmable gain resistors as shown in the figure below. A gain_select of 0, 1, 2, 4, and 8 creates a transimpedance of 5Ω, 20Ω, 55Ω, 155Ω, and 505Ω, respectively.

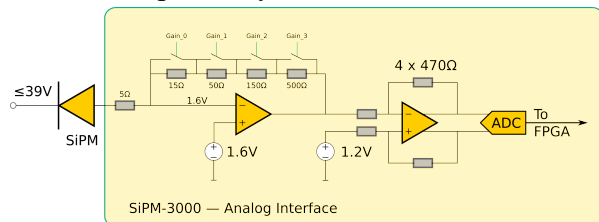


Fig. 11: The input amplifier of the SiPM-3000.

8. Calibration

8.1 Theory of operation

The MCA-3000 uses a switched-gain input amplifier stage followed by an ADC with a 1V input range. The ADC samples the amplifier output voltage at a rate of typically 40MSPS. In the PMT-3000, the sampling rate can reach 120MHz.

For proper operation, the amplifier output voltage must fall within the 1.0V input range of the ADC, as shown in the two figures below.

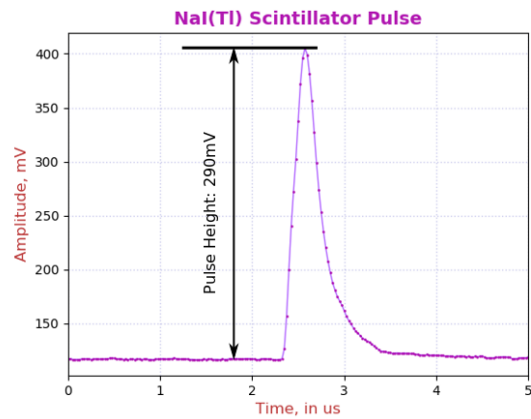


Fig. 12: A regular NaI(Tl) pulse.

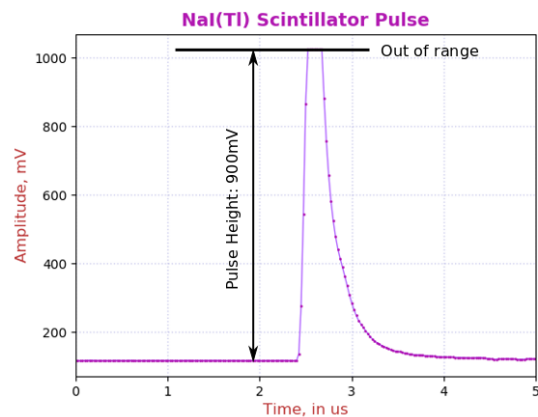


Fig. 13: An out of range NaI(Tl) pulse. Note how it cannot reach beyond 1023mV.

Pulse height is a measure of the difference between the maximum pulse voltage and the baseline from which it rises.

There is a built-in DC-offset of around 128mV and the maximum measurable voltage is shown as 1.023V. Hence the maximum practical signal pulse height range is around 900mV.

The first step of calibration at a chosen electronics gain, is to make sure that the maximum wanted energy can be measured by the ADC. At a fixed electronics gain, the applied operating voltage will determine the maximum measurable energy.

A very common calibration is to set the operating voltage such that the full-energy peak of Cs-137 (661.66keV) corresponds to a pulse height of 187mV. The maximum measurable energy is then $900\text{mV}/187\text{mV} * 662\text{keV} = 3.18\text{MeV}$.

After adjusting the high voltage to set the maximum measurable energy, the user can now vary the digital gain to map the entire energy

spectrum into an energy histogram of the desired size and calibration.

For example, we assume 40MHz ADC sampling rate and the recommended integration time of 1.2µs for NaI(Tl). We also assume that the Cs-137 662keV peak has an average pulse height of 187mV. In that case, choose a digital gain of 4700 to place the peak in the energy histogram at 662keV, ie have an MCA calibration of 1.0 keV per bin.

Below we describe the calibration steps in detail.

8.2 Calibration file

Calibration controls are in autocal.json. This file can be found in the `./rad_config/Matplotlib_gui/config/` folder. Its contents are explained in the table below.

| <i>Variable</i> | <i>Description</i> |
|------------------|--|
| cal_energy | The energy, in keV, for the calibration peak; eg 662 for Cs-137. |
| cal_pulse_height | The pulse height for the calibration peak; eg 187 for Cs-137 and 3.2MeV maximum measurable energy. |
| auto_update | 0→ Off; 1→ Apply the new operating voltage and restart the mca after user clicks on "Calibrate" button. |
| keV_bin | Desired MCA calibration, as keV per MCA bin. |
| gain_exp | <p>PMT's have a power law for their gain vs voltage function:</p> $\frac{gain}{gain_0} = \left(\frac{HV}{HV_0} \right)^{gain_exp}$ <p>Typical values are 5.6 for R6231, 6.0 for CR105 and 7.5 for many other 10-stage PMT. For SiPM this parameter is ignored.</p> |
| mass | The mass of the NaI crystal, in kg; used for computing the deposited dose rate from the count rate and deposited energy. |

Fig. 13: The autocal.json data explained.

For use with Broadcom SiPM we use the following gain formula:

$$gain = 1.0e6 \cdot (V - 28.6V)$$

The 28.6V are the sum of the typical 27V break through voltage and the 1.6V offset of the SiPM-3000 input amplifier.

You can manually calibrate the detector, in which case you have more freedom on how to do it. If you want to use the **Calibrate** button on the **Sample** panel, then the software will always pick the tallest peak in the histogram, beyond 50 MCA bins, to use in the calibration computations. This works best for isotopes with a well-separated medium-energy peak; eg Cs-137 and Na-22. Co-60 and K-40 are not recommended for the software-assisted calibration.

8.3 Step 1 of the calibration

First, adjust the operating voltage to set the maximum measurable energy. Open the **DSP** panel and set the **ha_mode** control to 1. Start a new histogram acquisition on the **Sample** panel by clicking on **New**. Wait until there are around 1000 counts in the peak maximum. Then click **Calibrate** on the **Sample** panel.

After 1 second, you will receive a message with the suggested new operating voltage. The software has applied the new operating voltage already if *auto_update* is set to 1.

There is some variation between the SiPM or PMT and you may have to repeat a second time to achieve the desired accuracy. Note that even if you do not know the exact gain exponent of your PMT, the process will still converge. It may just take another iteration or two.

Here is how to compute the *cal_pulse_height* (*cph*):

$$cph = 900 * \frac{cal_energy}{E_{max}}$$

8.4 Step 2 of the calibration

Now change the digital gain to achieve the desired MCA calibration in keV per MCA bin. From here on, the operating voltage and the electronic gain *gain_select* should be kept constant. Instead we use the *digital_gain* from the **Gain** panel.

Set the **ha_mode** control to 0 to return to normal energy measurements, and start a new histogram acquisition. Wait for sufficient accuracy and click **Calibrate**.

This time, you will receive a notice of the new *digital_gain* .

The new value was computed from a linear formula

$$dg_{\text{new}} = dg \cdot \frac{\text{cal_energy}}{E_{\text{peak}}}$$

and it will be accurate on the first try.

However, keep in mind that there is a statistical uncertainty in determining the actual energy of the histogram peak. You can compute the 1- σ error from the fwhm (in %) and the number of net counts (N) above background in the peak as

$$\varepsilon = \frac{\text{fwhm}}{2.3655 \cdot N}$$

9. Gain Stabilization

All MCA-3000 offer gain stabilization that keeps the gain, the trigger threshold and the maximum measurable energy constant.

NaI(Tl) is the most commonly used scintillator, and it also one of the most challenging. Its brightness and pulse shape change with temperature, while the application engineer wants to keep constant not only the MCA gain (keV per bin) but also the trigger threshold and maximum measurable energy that is independent of temperature.

All MCA-3000 offer a user-programmable set of lookup tables to change the operating voltage and the digital gain to achieve this goal. In addition, for NaI(Tl) the MCA also vary the hold-off time with temperature to avoid retriggering on the tail end of the same pulse.

Since code and data needed for the gain stabilization are embedded in the MCA, no user software is required and the calibration moves with the detector.

The PMT-3000 also offers LED-based gain stabilization to counter PMT aging. Vacuum photomultiplier tubes lose gain over time, even under light to moderate loads. For example the popular R6231 PMT loses half its gain after an accumulated anode charge of about 50C. Distributed over one year, this equivalent to an average anode current of just 1.6 μ A, with a 6%

gain loss per month. With a built-in LED that injects light into the back of the photomultiplier, the PMT-3000 can gain-stabilize on the LED response and counteract the PMT aging.

9.1 General theory of operation

All types of gain stabilization use lookup tables, and users can supply their own. The standard code includes a 64-long float array with three look up tables (**LUT**). The MCA frequently determines the temperature and adjust the operating parameters according to the values found in the lookup tables.

The LUT depend on the scintillator and one DSP setting, namely the integration time.

For NaI(Tl) the LUT supplied by the factory use an integration time of 1.2 μ s.

The NaI(Tl) integration time is fixed. For NaI(Tl) the LUT supplied by the factory was derived for an integration time of 1.2 μ s. That value remains fixed over the entire temperature range, as it provides the best energy resolution for small and large energies at all temperatures

But the hold-off time has to be varied.

NaI(Tl) pulses become very long when the crystal temperature falls below 0°C.

Consequently, the hold-off time has to be increased at low temperatures to avoid retriggering on the tail end of the same pulse.

Since this phenomenon is determined by the scintillator physics, it has been implemented as an immutable function in the MCA, instead of a lookup table. The function is used if `arm_ctrl["fields"]["cal_scint"]==1`.

The implemented hold-off vs time function for NaI(Tl) is

$$\max(1.25, (0.634 + 1.27 * \exp(-T/32.26)))$$

in μ s and with T being the temperature in °C.

How the LUT are constructed. Developers may find that they need to construct the correct LUT for their detectors, as they may use a different scintillator or a different PMT. To this end the detectors needs to be run through a temperature cycle. The detector is recalibrated to desired specification at each temperature and the operating voltage, digital gain and possibly the LED value are recorded. Using interpolation on then constructs the LUT at the fixed temperature steps.

Keep in mind that many crystals don't allow for rapid temperature changes or they will crack. 10°C per hour is a safe bet for a 50mm NaI.

Secondly, it takes a while for a detector to attain steady state. The vacuum PMT needs an hour and the crystal characteristic time depends on the material and the size. For a 50mm NaI it takes about half an hour, for a 76mm NaI it takes 1.5hrs. Hence, it is necessary for a precise measurement, to maintain the detector at a fixed temperature for 2 to 4 hours before moving to the next.

Thermal equilibrium is necessary. Keep in mind that the PMT and the scintillator both have different temperature coefficients. If the two are not in equilibrium, there is no easy way to provide gain stabilization. Hence an outdoor detector should be packaged with good thermal insulation to achieve thermal time constants of 2 hours or more – to ensure that all components are in a steady state situation. However, care must be taken not to insulate the electronics. Even if it only dissipates 300mW, that small amount of power stills needs to drain away to avoid self heating.

When using SiPM: SiPM-based systems react more quickly, because the SiPM-array has very little thermal mass. In uncooled systems its temperature may be close to the crystal or halfway between the electronics temperature and the crystal temperature, depending on the assembly.

In cooled systems the SiPM-array is well insulated from the crystal and its temperature remains constant as long as the Peltier cooling loop remains in regulation.

As a result, insulation for SiPM-based systems needs to be, at the minimum, just good enough to avoid cracking the crystal through temperature shock.

| <i>arm_cal registers and fields</i> | |
|-------------------------------------|--|
| <i>Index: name</i> | <i>Description</i> |
| 0: lut_len | Number of entries in the LUT; default is 19, 2..19 are allowed |
| 1: lut_tmin | Minimum temperature in the lookup table; Typically -30°C |
| 2: lut_dt | Temperature step size in the lookup table; Typically 5°C |
| [3:22]: lut_ov | Change of operating voltage vs temperature |
| [23:42]: lut_dg | Change of digital gain vs temperature |
| [43:62]: lut_led | Change of LED target vs temperature |
| 63: lut_mode | int(lut_mode)&0x1 → lock bit, set to 1 to prevent the user from reading the arm_cal data from the MCA. |

The arm_cal registers and fields.

9.2 Gain stab. summary

Standard software recognizes these gain stabilization modes (gsm):

- 0 ⇒ Off; Use when calibrating a detector.
- 7 ⇒ Suspend; Keep all parameters as they are.
- 1 ⇒ Use temperature lookup for change of operating voltage and digital gain; adjust hold-off if required.
- 2 ⇒ Compare measured LED value to expected value and adjust operating voltage accordingly. Use LUT for digital gain and compute hold-off time as needed.

| <i>arm_ctrl for detector calibration</i> | | |
|--|------------|---|
| <i>Name</i> | <i>gsm</i> | <i>Description</i> |
| cal_ov | 0,1,2 | Operating voltage when the detector was calibrated |
| cal_temp | 1,2 | Temperature (in deg C) at which the detector was calibrated |
| cal_dg | 1,2 | Digital gain when the detector was calibrated |
| cal_scint | 1,2 | Scintillator type; 1⇒ NaI(Tl), adjust hold-off time vs temperature. |
| cal_target | 2 | Target value for response to LED; used with gain_stab=2 |

The arm_ctrl registers and fields concerning detector calibration; The gsm column lists the gain stabilization modes for which this parameter is used.

| <i>LUT needed for detector calibration</i> | | |
|--|------------|--------------------|
| <i>Name</i> | <i>gsm</i> | <i>Description</i> |
| lut_ov | 1 | Operating voltage |
| lut_dg | 1,2 | Digital gain |
| lut_led | 2 | LED average |

The lookup tables (LUT) needed detector calibration; The gsm column lists the gain stabilization modes for a given LUT is used.

9.3 Gain stab. mode: gsm = 0

Off. Gain stabilization is turned off and the target operating voltage the MCA will set is the one requested in arm_ctrl["fields"]["cal_ov"].

9.4 Gain stab. mode: gsm = 7

Suspend. This is different from gsm=0. During gsm ≠ 0 the gain stabilization algorithm may have changed the target operating voltage to a temperature dependent value that is different from arm_ctrl["fields"]["cal_ov"]. Setting gsm=0 would revert to that original value, while gsm=7 will leave the value unchanged. The same applies to the digital gain and the hold-off setting.

9.5 Gain stab. mode: gsm = 1

This mode relies on just the look up tables and the measured temperature. It is implemented for all MCA-3000. The units ship with LUT determined for NaI(Tl) at an integration time of 1.25µs

9.6 Gain stab. mode: gsm = 2

This mode adjusts the voltage using an LED measurement It is only implemented for the PMT-3000 series. The units ship with LUT determined for NaI(Tl) at an integration time of 1.25µs

The driving circuit for the LED used in the gain stabilization is electronically temperature compensated, but the compensation is not perfect. Since also the scintillator brightness changes with temperature, the ratio of the LED value and the full energy peak from a gamma-ray will be temperature dependent. Hence, there needs to be a lookup table to tell by how much the LED target value needs to be shifted as the temperature changes, in order to keep a full-energy gamma-peak constant.

Use an LED when expecting PMT aging. The purpose of the LED is to counteract PMT aging. This applies mostly to remotely installed detectors where a frequent recalibration is not possible. Whenever a detector can be recalibrated monthly and only sees light to moderate loads (anode current < 1µA with round the clock operation), then an LED system is not required.

Since aging is not a problem for SiPM, the SiPM-3000 series does not offer an LED option.

Adjusting the LED. In the LED panel you will find the controls shown in the table below

| <i>LED controls</i> | |
|---------------------|--|
| <i>Name</i> | <i>Description</i> |
| Period | LED frequency = 2441Hz/(P + 1) |
| Width | Driver pulse width = W · 0.050µs |
| Attenuate | Attenuation factor = 1/2 ^A |
| On | 0→ Off; 1→ On; |
| Select | 0→ LED pulses are hidden; 1→ Only show LED in histogram and traces |

The LED controls.

Manual vs algorithmic operation of the LED. When gain stabilization is off the user can set the the LED frequency and pulse width. When the detector has been calibrated, read the LED value from the RR_10 field on the Status panel

and enter it into the `cal_target` field on the `Gain` panel.

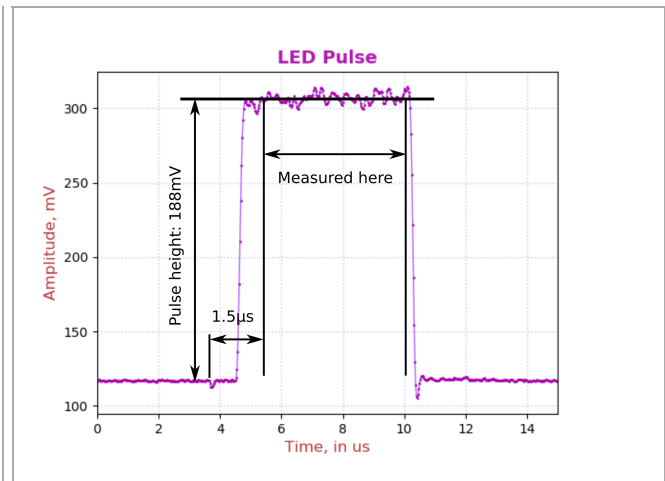
Don't run the LED too fast. Keep in mind that the LED pulses also cause a load on the PMT which may reduce its gain slightly (1% to 5% worst case). Hence, keep the LED running at $\leq 100\text{Hz}$ ($P>24$) during calibration. The PMT-3000 will update the measured LED average every 1024 pulses. At a rate of 100Hz, it will take 10s between updates.

Keep the LED pulses short, especially for high count rates. When using `Select=1`, you can see the LED values histogrammed. Mostly they will fall into a narrow peak of with an "energy resolution" of 1% to 2% fwhm. At high input count rates from a source ($> 20\text{kHz}$) you will also see a small fraction of events classified as LED that are a pile up of an LED pulse with a gamma-ray. These are rare, but they do slightly affect the accuracy of the LED average measurement. Keeping the LED pulses short helps to suppress these unwanted events.

The LED pulse is shorter than the drive pulse. Not only is the LED light pulse about $0.9\mu\text{s}$ shorter than the driver pulse, the FPGA logic also begins measuring the LED light only $1.5\mu\text{s}$ after the drive pulse starts. Hence if you the the LED width $W<30$ you will see a measured LED average of zero.

Similarly, the FPGA stops measuring the LED light about $0.15\mu\text{s}$ before the pulse actually ends. These two measure ensure that the LED light intensity is only measured during the flat top of the pulse where the behavior is most predictable.

Usually the pulse height is adjusted at the factory to about 200mV when the detector is calibrated to have a maximum energy range of 3.2MeV. This allows operation with a higher gain and a lower energy range of down to 1MeV max, while keeping the LED pulse within ADC range.

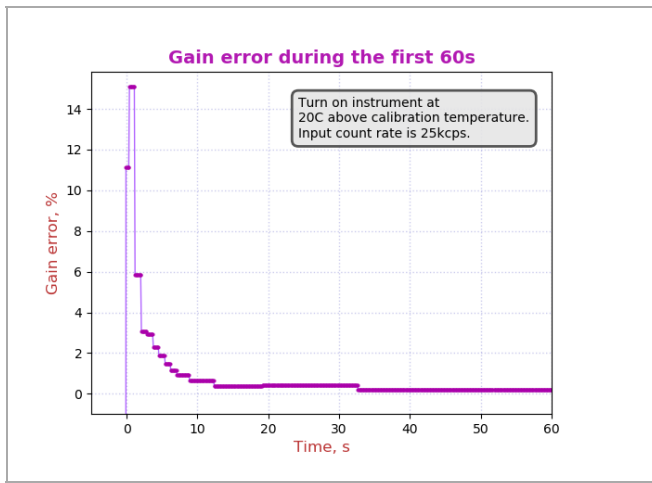


A wide LED pulse to demonstrate how it is being measured.

9.7 Dynamical performance for $gsm = 2$

The gain error falls to $< 1\%$ in 10s. If an instrument that was calibrated at room temperature is left in a hot vehicle and turned on at a temperature much different from the calibration temperature, its initial gain settings will be all wrong for the new temperature. For $gsm=1$ the adjustment will be immediate, as the instrument only requires one temperature lookup to adjust to the new situation.

For LED-based gain stabilization ($gsm=2$) there is a short time lag. The instrument needs to measure the new LED value and then step by step adjust the high voltage to make the LED value meet the computed LED target. As shown in the figure below, this process takes less than 10 seconds. In the beginning the LED blinks at 1220Hz, and the high voltage is updated about every 0.8s. Once the LED value is close to the LED target (within 1%) The instrument gradually reduces the LED blinking frequency to 19Hz to reduce the load on the PMT and avoid accelerated PMT-aging.



This is the response of the gain stabilization algorithm to turning on the detector at a temperature that is different by 20°C from the calibration temperature. Within 10 seconds the gain error falls back to below 1%.

10. Mechanical and Pin Outs

10.1 8-Pin Connector

The PMT-3000 uses a Bulgin PX0447 mini-B USB connector for power and USB communication. Mating cables are the Bulgin PX0441 (USB-A) and PX0442 (USB mini-A) series cables. They come in lengths from 2m to 4.5m.

The PMT-3000 uses a Switchcraft EN3P8MPX 8-pin connector for GPIO and serial UART communication. The mating connector is part of the EN3C8 series.

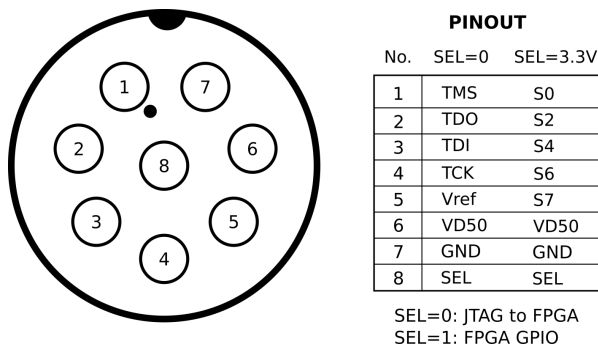


Fig. 14: Pinout of the EN3P8 connector.

11. Product and Part Numbers

11.1 Product numbers

The PMT-3000 is part of a product series that all contain an ARM M0+ 32-bit processor for communication and slow control, such as gain stabilization. In the -1000 series the MCA is implemented by software in the ARM processor. In the -3000 series the MCA is

implemented independently within an FPGA for very high-speed operation. The -3000 series also uses a waveform-digitizing ADC and offers detailed pulse capture and real time pulse shape discrimination.

Both types of devices are available for vacuum photomultiplier tubes (PMT) and Si-photomultipliers (SiPM). In both cases, the device generates the operating voltage for the photo-sensor from the incoming 5V.

| <i>P/N</i> | <i>Sensor</i> | <i>FPGA</i> |
|------------|---------------|-------------|
| PMT-3000 | PMT | Yes |
| SiPM-3000 | SiPM | Yes |

Table : Part numbers.

11.2 USB-ID

On the USB bus devices are recognized by their Vendor ID (VID), Product ID (PID) and Serial Number (SN). The vendor ID for Bridgeport Instruments is 0x1FA4. The Product ID's are shown in the table below. Within a product the serial number is fixed, unless BPI makes a custom device that requires a non-standard driver. Note that simple extensions, such as adding a variable to the controls, does not require a new driver.

The BPI software recognizes individual devices by the unique serial number burnt into each ARM processor. The device reports that when the host reads **arm_version**. The serial number communicated in response to USB setup commands is fixed for each part, to avoid that the host keeps adding every new device to an ever longer list of devices requiring a designated USB driver.

| <i>P/N</i> | <i>PID</i> | <i>SN</i> |
|------------|------------|----------------|
| PMT-3000 | 0x0103 | armMorpho0001 |
| SiPM-3000 | 0x0203 | sipmMorpho0001 |

Table : Product ID and USB bus serial numbers. The vendor ID is always VID=0x1FA4.

11.3 Device serial numbers

Each ARM processor has an immutable 128-bit unique serial number, which can be printed as a 32-character hexadecimal string. The MCA Data Server always uses the complete 32-character string to identify the device. Because

of space constraints, the serial number printed onto the device is shortened to 8 hexadecimal

characters.