A system for automated measurement of parameters of large quantities of MPPC detectors

Michał Dziewiecki[1], Robert Kurjata[2], Janusz Marzec, Robert Sulej[3], Krzysztof Zaremba and Marcin Ziembicki

Warsaw University of Technology, Institute of Radioelectronics
Nowowiejska 15/19, 00-665 Warsaw, Poland
E-mail: m.dziewiecki@ire.pw.edu.pl

A new system for fully automated parameter measurements of large quantities of MPPC detectors has been developed. The aim of the project was to create a compact, portable device allowing measurement of gain, breakdown voltage, PDE (relative), dark rate and cross-talk & afterpulse ratio, with minimum operator interaction required. The system consists of two parts: a 32-slot, temperature-controlled MPPC feeder with integrated pre-amplifier and LED flasher and an electronics box with a pulse shaper, a high-speed ADC with on-board FPGA, a two-threshold discriminator, a slow-control unit, a network interface and a power supply. An effort has been made to minimize noise and external interference, as well as to achieve good stability of MPPC supply voltage and temperature. The whole system is controlled via Ethernet network by accompanying software, which also performs on-the-fly data analysis.

---

[1]   Corresponding author
[2]   Speaker
[3]   Currently employed at the Soltan Institute for Nuclear Studies, Hoża 69, 00-681 Warsaw, Poland

1. Introduction

The ND280 detector complex of the T2K experiment [1-3] will be the first large scale (approx. 60000 pcs.) application of multi-pixel photon counters (MPPC), also known as 'silicon photomultipliers'. Since this type of sensors is fairly new, mass testing of all the sensors has been performed, in order to determine their parameters. In this paper, we will present test bench that was used to measure some of the MPPCs (approx. 1200) designated for installation in the SMRD sub-detector [4]. Other test setups as well as results can be found in [5-6].

2. Methods

2.1 Concept

From the very beginning it was clear that, given the number of sensors to measure, one needs to implement some form of an automatic measurement procedure. Initial measurements done using the "quick and dirty" laboratory setup allowed us to create a specification for a new device, which we built from scratch.

First of all, it had to be highly automated and as immune to operator errors as possible. Other desired features were modular architecture (easily serviceable) and compact design (portability). In order to assure identical measurement conditions for all the sensors, we decided to use single measurement channel. The measurement process was to be controlled by PC with dedicated software for data acquisition and control, connected to the device via Ethernet. We also wanted to have an option for raw data collection. The MPPCs were to be placed in a temperature stabilized feeder with 32 slots – this number seemed optimal, since the most time consuming operation is adjusting the temperature and, this way, it would be possible to do it simultaneously for all 32 sensors. A special consideration was given to maintaining high stability of MPPC working conditions (i.e. supply voltage and temperature) as well as to the issue of minimizing noise and interference. The resulting measurement concept is shown in Fig. 1.
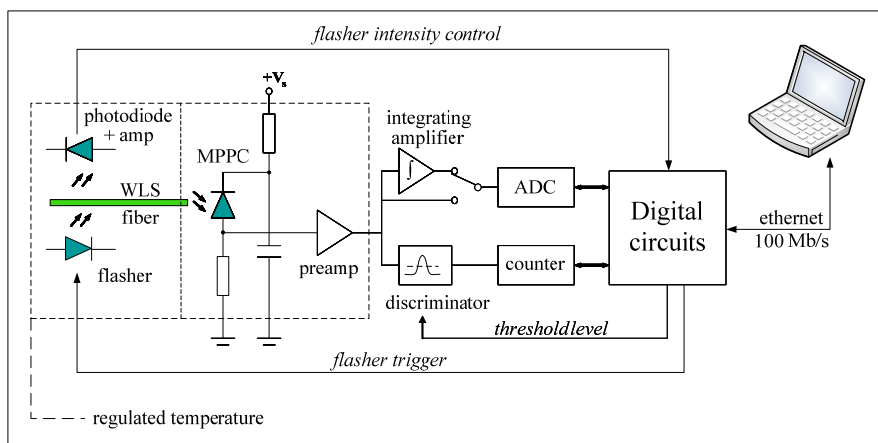

Fig. 1.
Implemented measurement concept.

2.2 Hardware

The system's architecture (Fig. 2) consists of three main parts – the temperature stabilized feeder for 32 MPPCs with pre-amplifier and LED flasher (Fig. 3), an electronics box, also containing a power supply and finally the control software (see section 2.3). The feeder is made from an aluminum alloy and has three parts – a fixed outer one, connected to heating / cooling elements, a rotating inner one with MPPC slots (for Hamamatsu S10362 type sensors, ceramic case) and a cover with a LED flasher [7] along with light intensity monitoring circuitry. Signals from each sensor are conducted to the bottom of the rotating part, where a PCB with gold-plated pads is mounted. Then, once the correct feeder position is selected, the sensor is connected to the pre-amplifier via a set of brushes. The connection proved reliable, it also provides very short distance between the sensor and the pre-amplifier. After initial amplification, the signal is transmitted to the electronics box. The temperature stabilization is achieved by using power transistors for heating and Peltier elements for cooling, both placed at the bottom of the outer part (see Fig. 3). Total heating/cooling power is 200W, the whole feeder is also insulated from ambient by polyurethane foam (Fig. 4a). Temperature is regulated using a master/slave PID loop, with the part containing the sensors being the master and the whole outer part being the slave. The master part's temperature is measured using two class A PT-1000 sensors (Fig. 4a), which allowed for system's absolute temperature accuracy of ±0.5°C[1]. The outer part's temperature is measured using standard class B sensor. Achieved regulation quality was better than 0.1°C (Fig. 4b). The position control was done using a self-blocking worm gear and a stepper motor, a reference 'zero' position was found using two sensors placed on top of the rotating part.

The electronics box contains the power supply and all the electronics modules – a shaping amplifier (quasi-Gaussian shaping, FWHM 30 ns, pole-zero pulse tail cancellation, optional analog integrator), a fast ADC board (FADC), a dual-threshold discriminator with scalers (20ns dead time), a slow control board, a control interface and an Ethernet switch (since we have two Ethernet capable boards). In order to avoid ground loops, power supply is floating and all signal connections between the box and the feeder are symmetrical. All power elements are linear, including fans / heating /
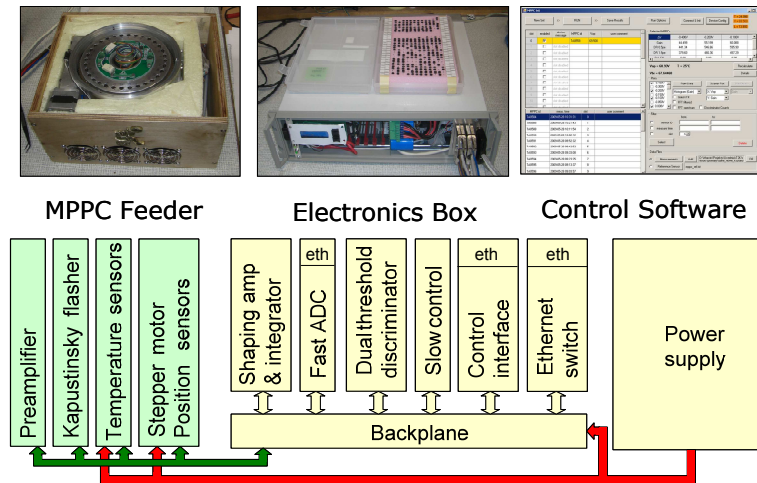

Fig. 2.
System architecture. Green blocks belong to the feeder, modules marked yellow are inside the electronics box.

---

[1] Class A PT-1000 sensor's absolute accuracy is ±0.15°C, but it is necessary to account for errors due to electronics noise, contact resistance and calibration.

cooling drivers (no interference caused by PWM[2]). Slow control has been implemented using a single microcontroller, an external ADC (24-bit, Δ-Σ) and two DACs – an 8-bit one for fans/heating/cooling and a 16-bit one for MPPC/flasher voltage. The ADC is used to read temperature sensors and to provide feedback data for digital regulators controlling MPPC and flasher voltages. Drivers for these voltages have been placed on the slow control board. Voltage accuracy and stability is better than ± 10 mV at 70V (calibrations done using 6-digit HP 34401A multimeter). Signal digitization is done by the FADC board, utilizing a 200 Msps, 10 bit analog to digital converter (AD9601), a Xilinx Spartan 3E series FPGA chip (XC3S500E), controlling most of the board's functionality, and a microcontroller, which is responsible for communication with the backplane and FPGA programming. The FPGA receives data from ADC, selects data samples, forms network frames and sends acquired data via 100MBit network to the PC, using layer 2 raw Ethernet frames. It also controls measurement cycle by creating all necessary trigger signals. Currently, only 30% of FPGA chip is used, thus future expansion of functionality is possible. Obtained pulse waveforms are shown in Fig. 5, sample ADC spectra is shown in Fig. 6.

2.3 Software

A standalone PC application has been developed (*C#* language) to setup and monitor the device parameters, as well as managing data acquisition and analysis. *WinPcap* library [8] was used to capture raw Ethernet packets in data acquisition mode. Device is controlled through the simple TCP/IP based protocol. Parameters (operating voltage V_{op}, temperature, light source, feeder motor, amplifier, discriminator, FADC) may be set up manually or by running scripts. Continuous monitoring is performed for the V_{op}, the temperature and the light source intensity. Trend plots are available for these parameters.
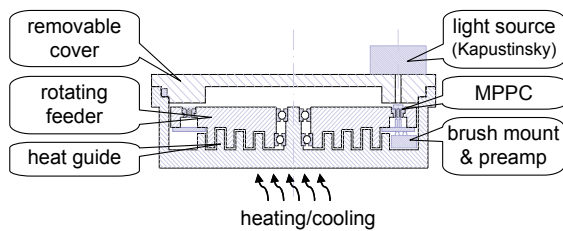
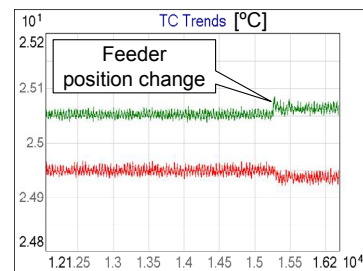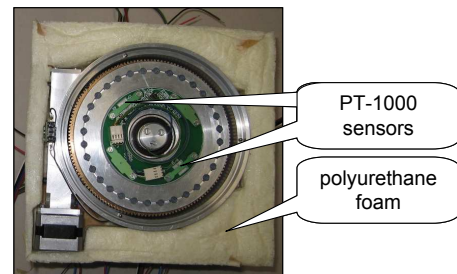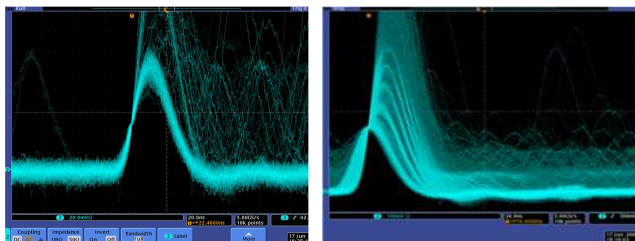

Fig. 3.
Feeder cross-section.

Fig. 5.
Registered MPPC pulse waveforms
at the output of the shaping amplifier.
(a) Dark rate measurement, trigger at 0.5 p.e.
(b) Measurement with light, trigger at ~3 p.e.

Fig. 4.
(a) MPPC feeder with marked positions
of temperature sensors.
(b) Temperature trend plot, both sensors.

---

[2] Pulse Width Modulation

4

Pulse waveforms are sent to the application in the raw Ethernet packets; each waveform is 256 samples length. Pulse charge is calculated as a field under the waveform in a narrow time window adjusted to the pulse position. Histogram of the charge amplitudes forms a result of the basic measurement (Fig. 6).

Photosensor parameters are measured for selected operating voltages. The protocol of the measurement is as follows:

- *Measurement with light source turned on*

Obtained charge amplitude histogram contains pedestal peak and a ten to fifteen well separable photo-electron peaks. Peaks are assumed to be Gaussians. Peak positions and amplitudes are initially estimated using adaptive FFT filtering procedure and then Gaussian fit is fine tuned (Fig. 6). Calculated Gaussian peak parameters are used for the further analysis.

Fig. 6.
Charge amplitude histogram (blue curve) obtained with light source turned on. Red curve corresponds to the Gaussian fit.

First, photosensor gain is calculated as:

$$gain = peak\_dist \cdot calib\_const,$$

where *peak_dist* is the mean distance between photo-electron peaks and *calib_const* is a constant used to express gain in absolute units (based on device calibrations). Photosensor gain increases linearly with operating voltage. Basing on this assumption, a line fit is calculated for $gain(V_{op})$ measured at several operating voltages (Fig. 7). Obtained fit allows estimation of the V_{op} for a given gain. The breakdown voltage V_{br} is calculated as a voltage at which estimated gain is zero.

- *Measurement with low light condition (light intensity set to produce 1-2 p.e. pulses).*

Charge amplitude histogram contains five or six separable photo-electron peaks (Fig. 8). This measurement, together with dark spectra, is used to calculate relative photon detection efficiency. The following formula is used:

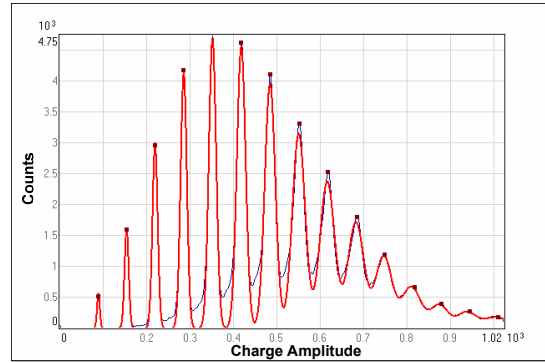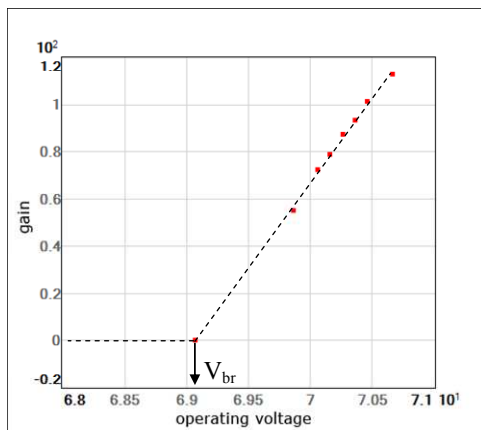$$relative\_pde = p\_light – p\_dark,$$

Fig. 7.
Photosensor gain calculated for several V_{op}, with marked line fit that allows V_{br} estimation.
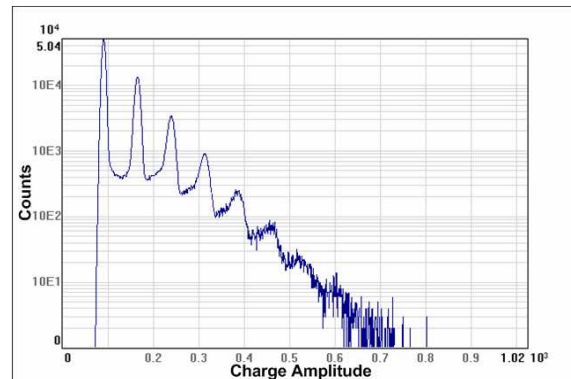
Fig. 8.
Charge amplitude histogram for low light conditions.

where *p_light* is a ratio of peaks above 0.5 p.e. level to the total counts in the histogram for low light level measurement and *p_dark* is the same ratio taken from dark spectra.

▪ *Measurement with no light.*

Dark rates are calculated using discriminator counts with no light conditions. Two thresholds are adjusted to count pulses above the level of 0.5p.e. and 1.5p.e. Charge amplitude histograms are also collected (Fig. 9).

▪ *Calculation of the cross-talk and after-pulses (CT&AP) fraction.*

CT&AP fraction is calculated using charge

Fig. 9.
Charge amplitude histogram obtained with no light conditions. Pedestal and 1p.e. peak positions are estimated using Gaussian fit calculated for corresponding measurement with light source turned on.

amplitude histogram collected with no light conditions (Fig. 9), according to the following formula:

$$CT \& AP = \frac{t_{1p.e.} - n_{1p.e.}}{\sum n - n_{0p.e.}}, \quad t_{1p.e.} = -n_{0p.e.} \cdot \log\left(\frac{n_{0p.e.}}{\sum n}\right),$$

where: Σn is the sum of all counts in the histogram; $n_{0p.e.}$ is the counts number in the pedestal peak; $n_{1p.e.}$ is the counts number in the one photo-electron peak; $t_{1p.e.}$ is the "true" one photo-electron count number estimated from the assumed Poisson distribution of the peak amplitudes.

3. Results

The main purpose of mass testing was finding faulty sensors, defined as the ones with parameters far from the average. A total of 1202 MPPCs were measured, all at t=25°C and Hamamatsu specified V_{op} for gain 7.5×10^5. A smaller subset of sensors was measured at multiple voltages and temperatures. The sensor quality was excellent, with only four bad, of which three were damaged by us. For single temperature, single voltage, 500k counts per histogram measurements the device throughput was 32 sensors per 45 minutes, while for 7 voltages, single temperature and 1M counts per histogram it was 32 sensors per 12 hours. It has been found that these values could be significantly improved by further software changes (more parallelism) and possibly by moving some functionality to FPGA, which could allow for higher trigger rate. Results are shown in Fig. 10 to Fig. 12.
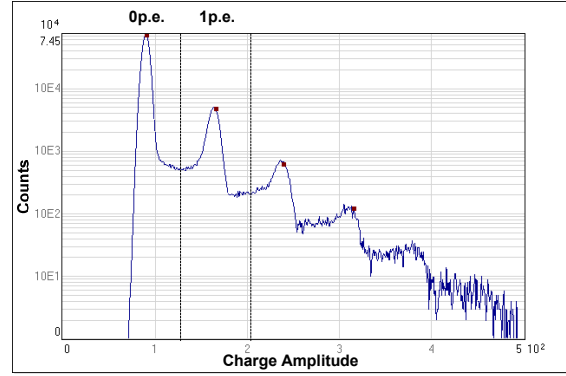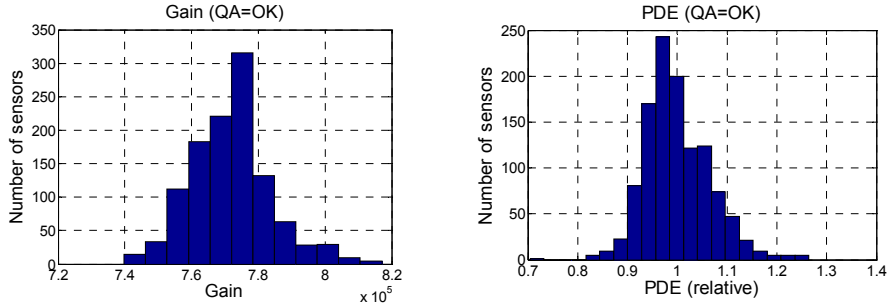
Fig. 10.
Gain (left) and relative histograms (right) for 1202 sensors.
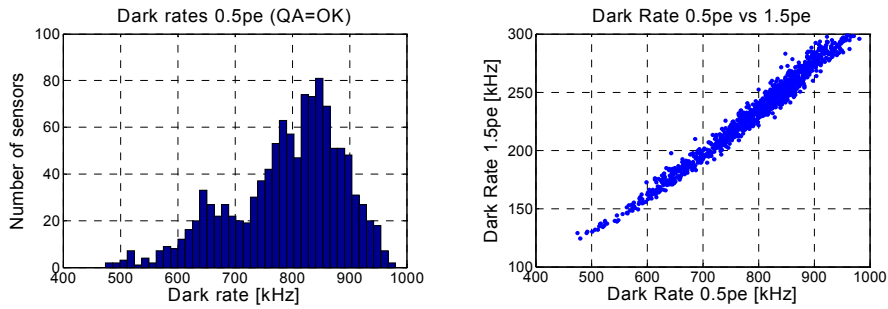All sensors measured at Hamamatsu specified V_{op} for gain 7.5×10^5, t=25°C.

Fig. 11.
Distribution of dark rate 0.5 p.e. (left) and plot of dark rate 1.5 p.e. vs 0.5 p.e. (right).
All sensors measured at Hamamatsu specified V_{op} for gain 7.5×10^5, t=25°C.
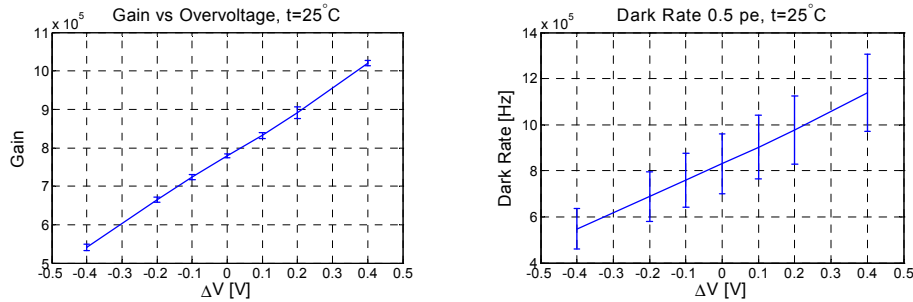

Fig. 12.
Gain (left) and dark rate (right) as a function of overvoltage (160 sensors).

4. Conclusion

A device capable of automated measurements of MPPC sensors has been described. It was used to perform mass testing of 1202 MPPCs designated for installation in SMRD detector. Results of the tests were presented. The performance of the device was found satisfactory, although further optimizations are possible.

Acknowledgements

References

[1] D. Karlen et. al., *Near detectors for the T2K experiment*, Nucl. Phys. B (Proc. Suppl.) 159 (2006) 91-96

[2] Y. Kudenko et. al., The near neutrino detector for the T2K experiment, Nucl. Instr. Meth. A 598 (2009) 289-295.

[3] Y. Itow et al., *The JHF-Kamioka neutrino project*, `[arXiv:hep-ex/0106019v1]`

[4] A. Izmaylov, et. al., *Scintillator counters with WLS fiber/MPPC readout for the side muon range detector (SMRD)of the T2K experiment*, `[arXiv:0904.4545v1]`

[5] M. Yokoyama et. al., *Mass production test of Hamamatsu MPPC for T2K neutrino oscillation experiment*, `[arXiv:0807.3147v1]`

[6] M. Yokoyama et. al., *Application of Hamamatsu MPPCs to T2K neutrino detectors*, `[arXiv:0807.3145v1]`

[7] Omar Veledar et. al., *Simple techniques for generating nanosecond blue light pulses from light emitting diodes*, Meas. Sci. Technol. 18 (2007) 131–137.

[8] *WinPcap*, open source packet capture library, www.winpcap.org.