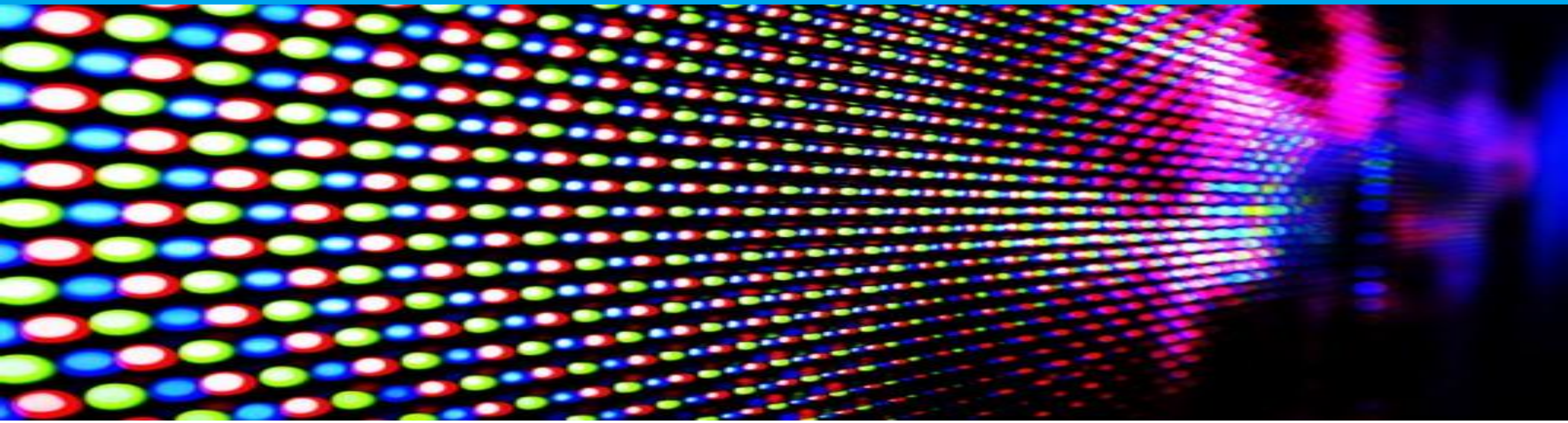
Gain calibration



AHCAL ,Tokyo analysis workshop

07.08.2018

Olin Lloyd Pinto

Need for gain

- SiPMs: Sensitive to changes in temperature and operating voltage.
- LED system - installed to monitor the stability of the readout chain in time. (controlled by CALIB board – HBU)
- Due to limited number of SiPM pixels, leads to non-linear response for large signal.
- SiPMs are monitored during test beam operation - light intensity from zero to the saturation level.
- As the saturation behaviour of the SiPM appears at a pixel scale.
- Need a way to obtain this scale - done via **gain calibration.**
- Offers the possibility of looking directly at the SiPMs and monitoring their performance.

Procedure to obtain the gain of an SiPM

Conversion

Step 1: convert to root

- Convert the **raw slcio/txt file(s) to the root file** required to perform gain calibration.
- You will need these files:
  xml configuration file:
  <processor name="EUDAQEventBuilder2016_woBIF"/>
  <processor name="ProgressHandler"/>
  <processor name="RootTreeGeneratorEUDAQ2016"/>
  <processor name="MyLCIOOutputProcessor"/>

  the script to execute Marlin: myMarlin.sh

- Modifying the input and output in the steering file and then you can run the conversion with: ./myMarlin steering.xml

Get pedestal

Step 2: Pedestal extraction

- Pedestal: LED voltage reduced to 0 V, no light emitted from the LEDs when triggered.
- For every channel, a gaussian-like distribution centered in 400-500 ADC ticks created mainly by “electronic noise”.
- The pedestal extracted - for each channel as well as the memory-cell (to calculate the offset).
- This program extracts the pedestals in ET or AT for each chip, channel and memory-cell.
- It produces a file that can be used as input for the gain calibration and MIP calibration.

Pedestal extraction..

- The program **pedestal_memcell** takes the root file as input and provides a .tsv file as output.

- Execute and run:

./bin/pedestal_memcell [InputFilePath] [OutputDirectoryPath] Flag (HG/LG(0) or HG/TDC(1)).

Output files:

- First file, **pedestal_offsets_in.tsv** containing information of: chip, channel, pedestal position, pedestal width, pedestal offsetcell1 ... Pedoffsetcell16.

- Second file, **Plots_comp.root**, containing pedestal distributions for all channels, chips and memory cells.

Skip no pedestal

- If pedestal subtraction not performed then you have fig.1.
- If the procedure is well followed, well peaks due to noise suppression.

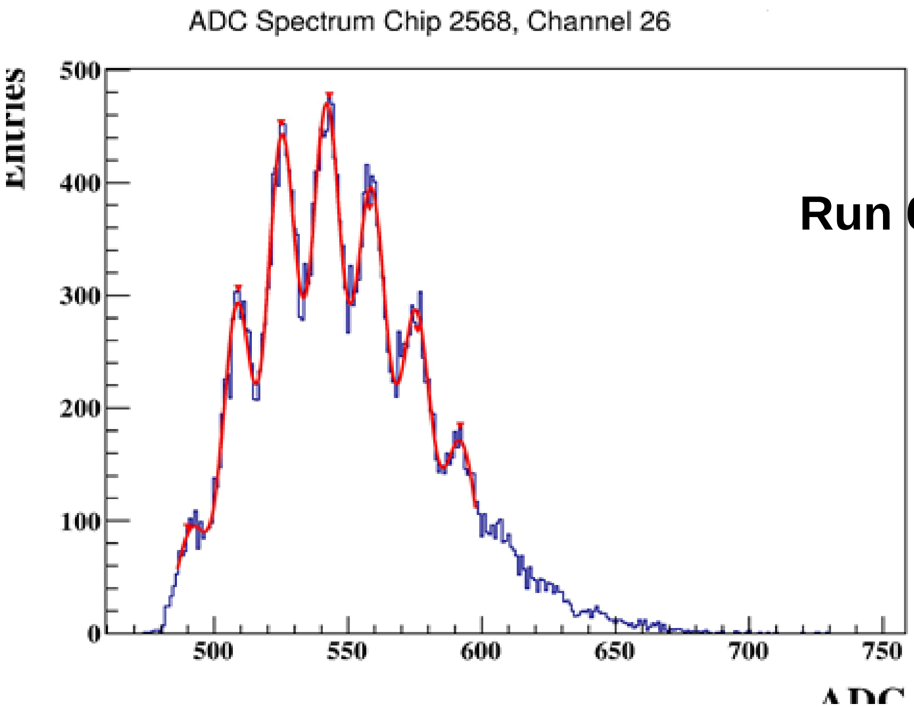


Figure 1

Run 60096, 6V

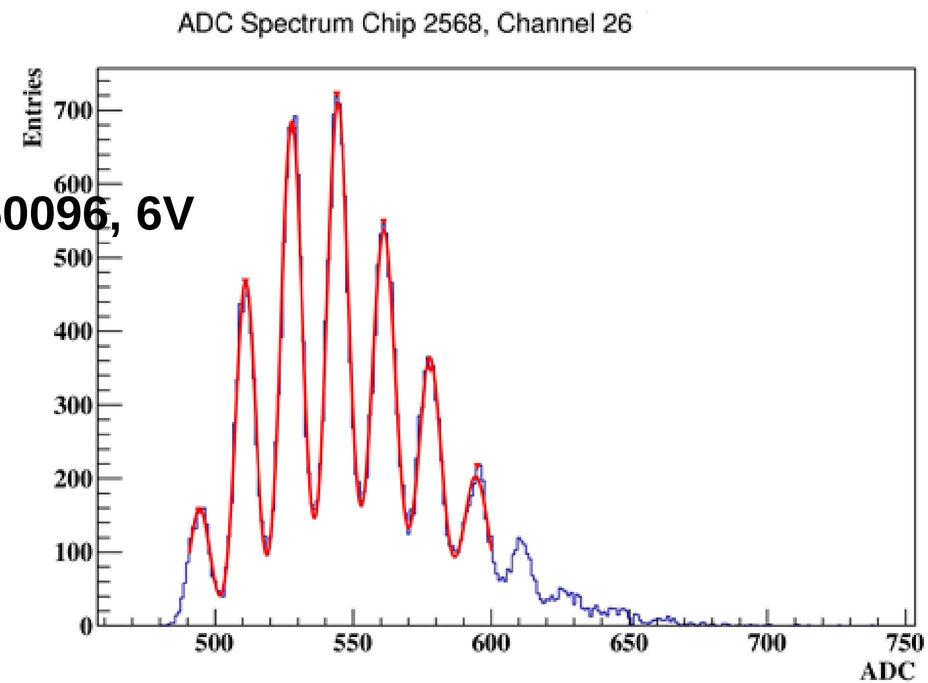


Figure 2

Calibration

Step 3: Peak information and checks

- Best determination of the gain - requires a photo-electron spectrum.

- At first, a **peak finder** is used to set each peak the mean value and the position to an approximate location of the corresponding photoelectron peak.
- Tspectrum: Automatically identifies the peak position in a spectrum in presence of background/statistical fluctuations – noise.
- Check the spectrum to see if it is fitable or not.
- Resolution: Determines resolution of the neighbouring peaks, default value is 1, corresponds to 3 sigma distance between peaks. Higher values allow higher resolution.

Peak checks

Step 3: contd..

- It passes a parameter sigma to the peak searching function and is selective to the peaks with a given sigma.
- Finally, the number of peaks and their positions are extracted.
- Peaks with amplitude less than threshold/highest peaks are discarded.
- Amplitude < 0.1 * highest peak are discarded in the peak search. This value is not optimized but avoids to find to many peak which are not real.
- Mean Condition which checks the ADC spectrums and accepts the mean < 1000 to 3000 ADCs.

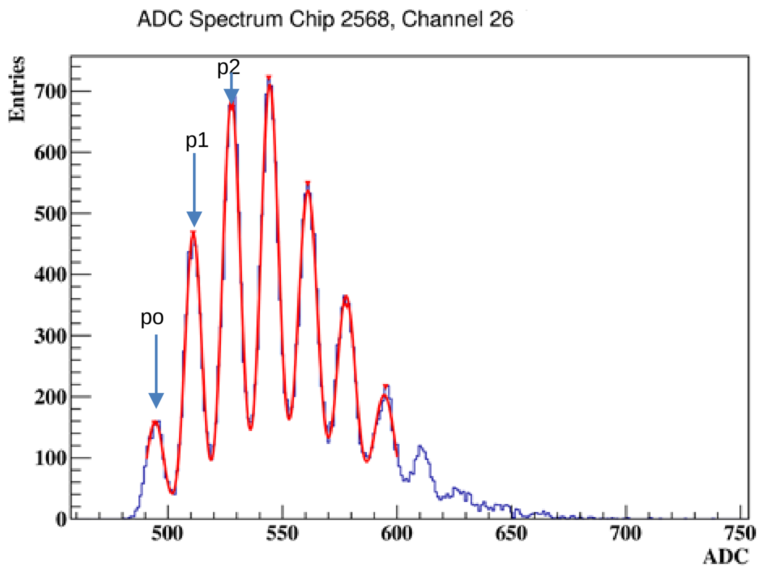
Multi-gaussian: Fit to peaks

Step 3: Multi-Gauss Fitting

- SiPM gain is from the fit result as the distance between pedestal (peak zero) and second peak divided by two.
- The Fit is in the form :

mg += par[2*i+3]*TMath::Gaus(xx,gain*i+p0,par[2*i+4])

Peak normalization **Pedestal positon** **Peak width**



HELMHOLTZ | ASSOCIATION

DESY

Comparative check

- An additional consistency check is performed to ensure the distance between pedestal and first peak agrees with the defined gain.
- Fits to peaks higher than the 3rd one are not directly used in determining the gain, but their proper description helps to improve the stability of the fits.
- The uncertainty in gain determination is mainly due to the fit.

- A Fast Fourier Transform (FFT) fit is performed in order to get a comparison with the gain values.

Execution

Step 4: Gain calculation

- Program used to extract the gain in external trigger
- The program can be executed by:

./gain_ledcalib [input file] [input pedestal file] [output folder] [mode 0/1] [runnumber] [vcalib]

- If the vcalib is not known but only the Run number, put the last parameter to -1. This will set the output names with the Run number instead of the vcalib.

Peak information

nPeaks CheckSpectrum = 5
peak 0 = 445 ADC
peak 1 = 461 ADC
peak 2 = 478 ADC
peak 3 = 494 ADC
peak 4 = 510 ADC

Peaks & position

x[1] = 461 x[0] = 445
ix 1 diffX 16 sigmaX 4 diffX-1: 0 sigmaX-1 : 0
first peak: verification not started yet
x[2] = 478 x[1] = 461
ix 2 diffX 17 sigmaX 4.12311 diffX-1: 16 sigmaX-1 : 4
x[3] = 494 x[2] = 478
ix 3 diffX 16 sigmaX 4 diffX-1: 17 sigmaX-1 : 4.12311
x[4] = 510 x[3] = 494
ix 4 diffX 16 sigmaX 4 diffX-1: 16 sigmaX-1 : 4
nPeaksToCheckMean = 5
nPeaks < 8: Condition satisfied

Fit Result

Result: gainFit = 16.1941 gainFFT = 16.5549
Result: Chi2/NDF = 1.20219 , Run Number: 60624
cntFitted = 9

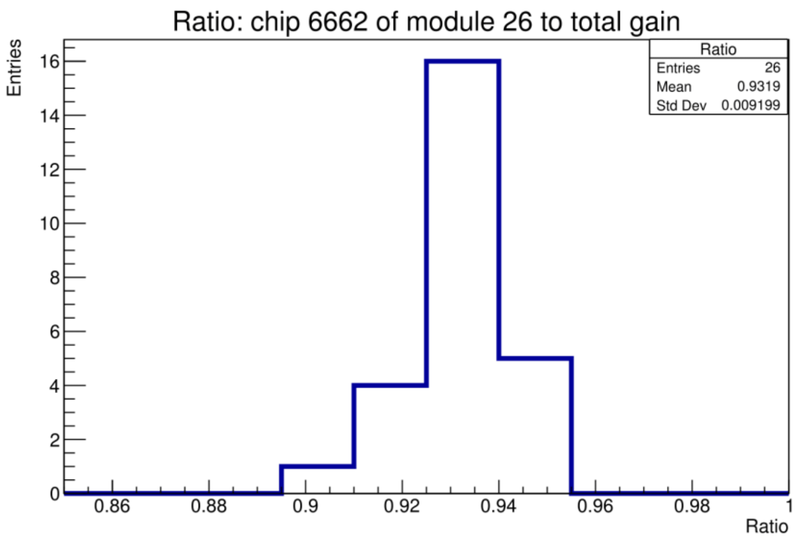
Output files

- Test.root: results related to the quality of the fits (chi^2).
- HighSignal.root: ADC spectrum(for all channels and chips).
- Plot_comp.root: fitted spectrums that satisfy minimum requirements.
- gainfits_XXXX.tsv: text file with the gain values. Five rows: Chip, channel, gain (multigaussian), gain (FFT), errorgain.
- Listbadfits : text file with chip and channel which don't fit.

Final gain constants

- Compute over all the gain_fits.tsv file.
- Calculates the mean gain for all chips and the corresponding channels.
- Missing channels: Provide the mean value of the corresponding chip.

Date[y.day.mm]	Gain[ADC]
20180705	16.6
20180805	16.58
20180905	16.58
20181005	16.59
20181605	16.57
20181905	16.21



Ratio of one chip to the mean gain of the 38 layers

Outlook

- Software for pedestal extraction and gain calibration successfully working.
- Need to optimize the procedure and the fitting techniques.

THANK YOU