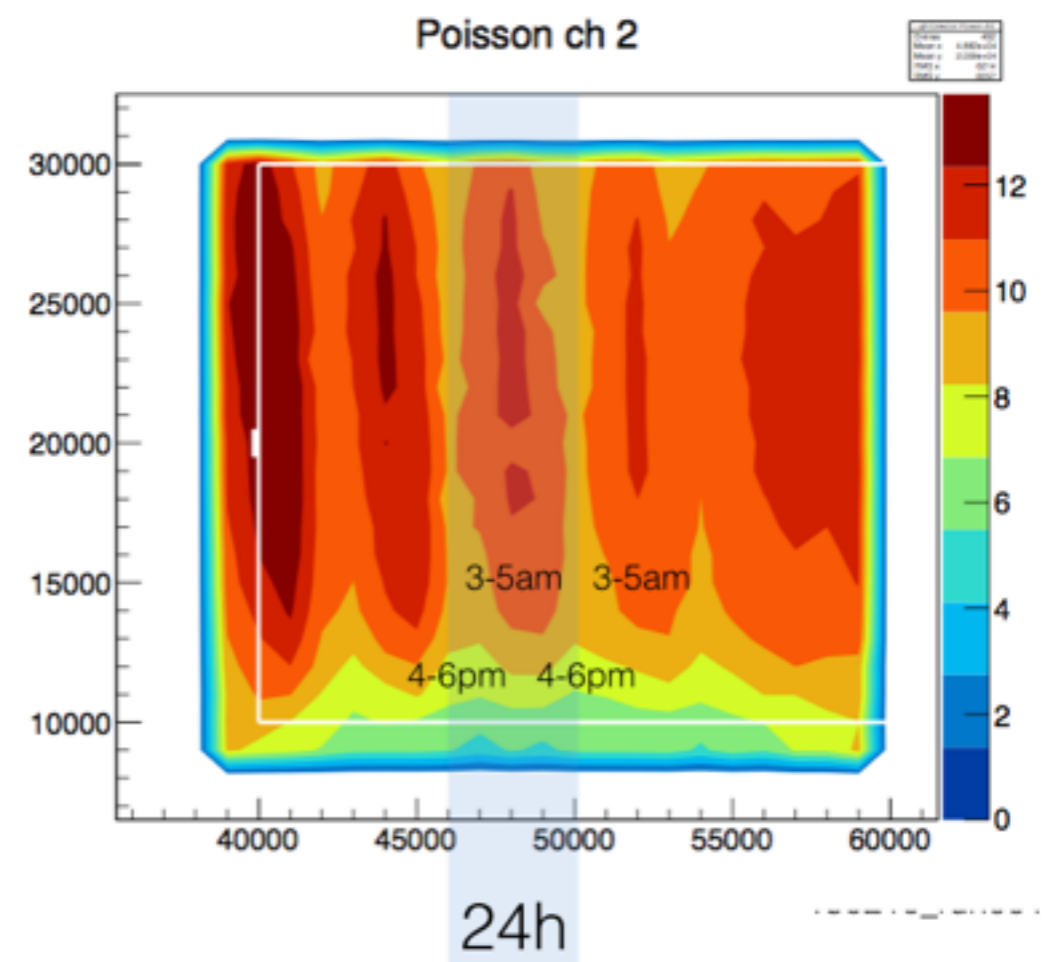


Lab setup update

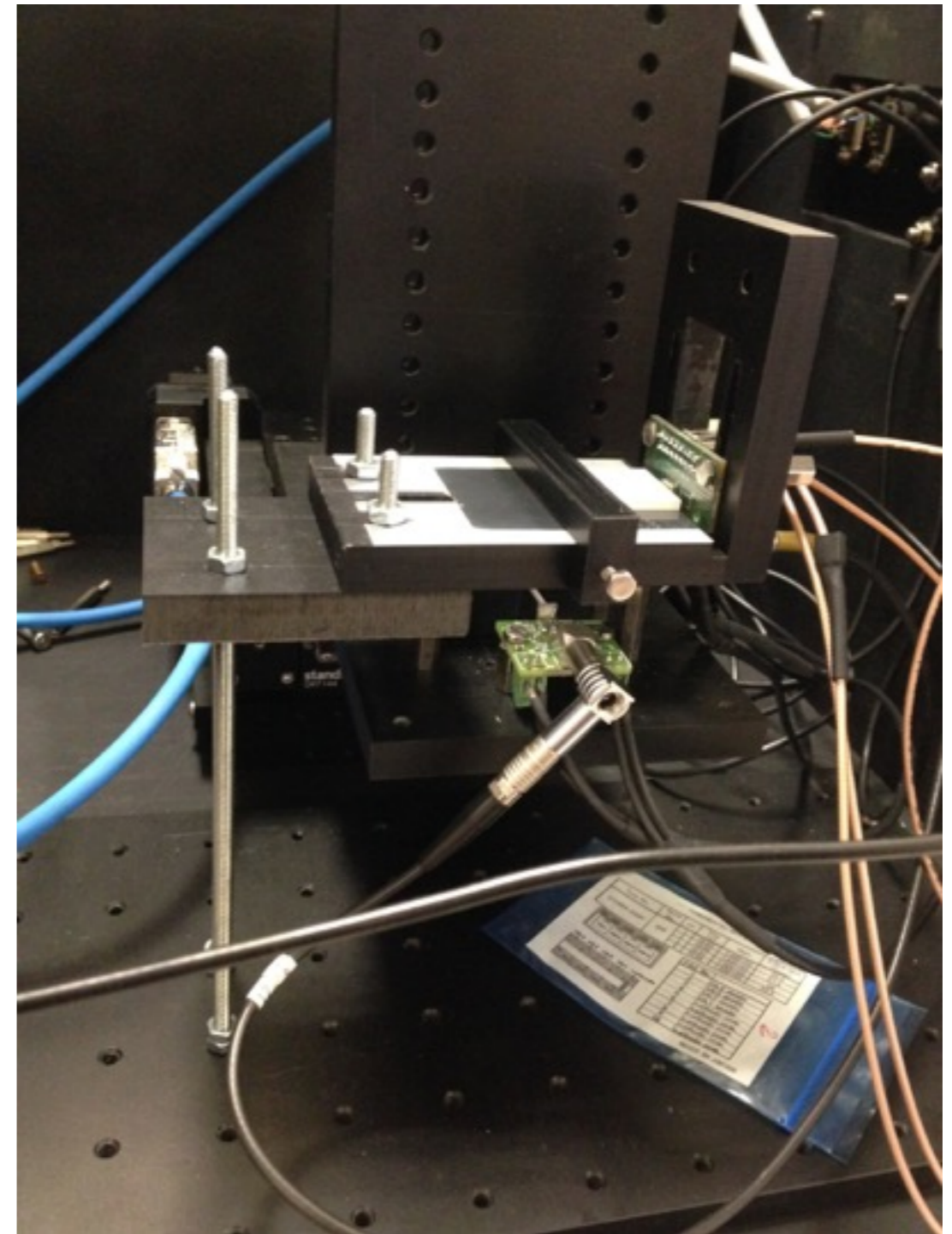
S. Gentile, F. Safai Tehrani, D. Luciani, F. dell'Uomo, S. Fiore, SV
150203

Temperature control

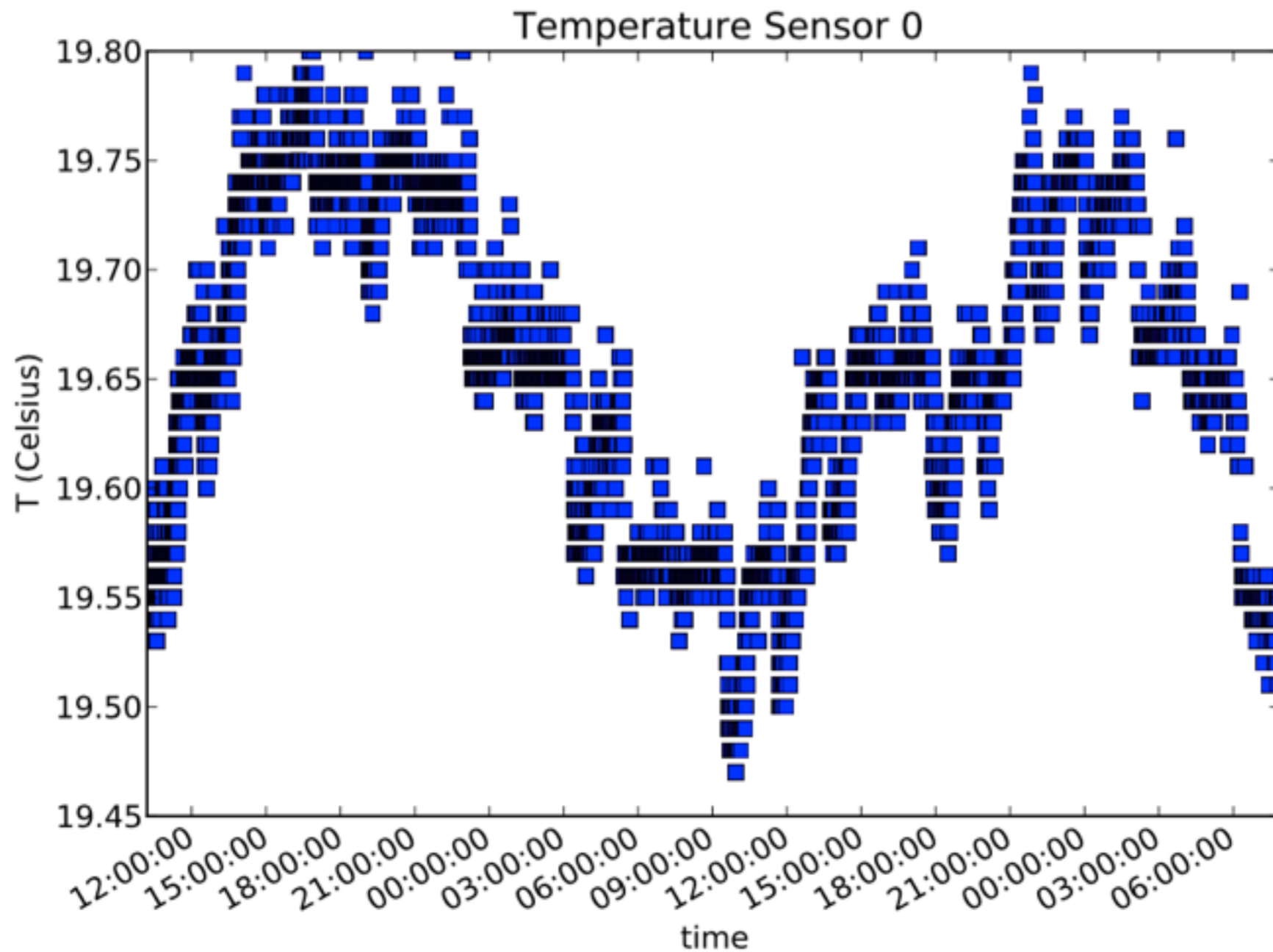
- during last presentation at this LAB meeting, two solution at our reach were discussed and adopted to take into account the important temperature dependance of gain vs temperature of SiPMs:
- the mounting of the measurement box into the thermostatic chamber.
- the introduction of temperature sensors in the measurement setup.



The new setup



Temperature stability



- similar behaviour of our chamber tested so far between 13 and 25 C.

current DAQ setup

The screenshot displays a Mac desktop environment with the following components:

- Figure 1: Temperature Sensor 0**
A scatter plot showing temperature data over time. The y-axis is labeled 'T (Celsius)' and ranges from 24.7 to 25.3. The x-axis is labeled 'time' and ranges from 15:00:00 to 20:00:00. The data points show a peak around 17:00:00.
- Terminal Window (daquser)**
Shows a sequence of commands and responses related to the DAQ system, including 'setArmBusy', 'HandlerBusy Request', and 'fields'.
- Terminal Window (Python)**
Shows a sequence of commands and responses related to the DAQ system, including 'runControl/start', 'startRun', and 'stopRun'.
- MCS SimpleDAQ Test System Control UI**
A web interface showing the status of the DAQ system. It includes sections for Server Configuration, EventData Service, Logger Service, FileWriter Service, TriggerService, EventFilter Service, and RunControlService.

Server Configuration:

- DFQueueDepth: 1000
- TimeQuantumForTriggerQueries: 100000
- HttpServer: http://10.0.0.3:3333
- InputFormat: HexColonData
- HttpRootPath: /Users/daquser/DAQ/daq/httpdroot/
- AsynchronousDF: true
- IncomingEventBufferDepth: 1000
- Port: 3333
- OutputFormat: CharData

EventData Service:

- Input Data Type: HexColonData
- Input Channel Occupancy: 0%
- Input Channel Occupancy (2): 0%

Logger Service:

- Logfile: stdout

FileWriter Service:

- Run name: 150319_run008_x47000_y35000_2520_7070_HVEST
- Current run name: 150319_run008_x47000_y35000_2520_7070_HVEST.1426786708
- Data writing to file: enabled
- Data writing to stream: disabled
- Maximum length of rolling file in bytes (0=unlimited): 0
- Maximum number of events per rolling file (0=unlimited): 0

TriggerService:

- Busy: false
- Started: false
- Number of triggers: 0
- Number of triggers sent: 0
- Interval (us): 1

EventFilter Service:

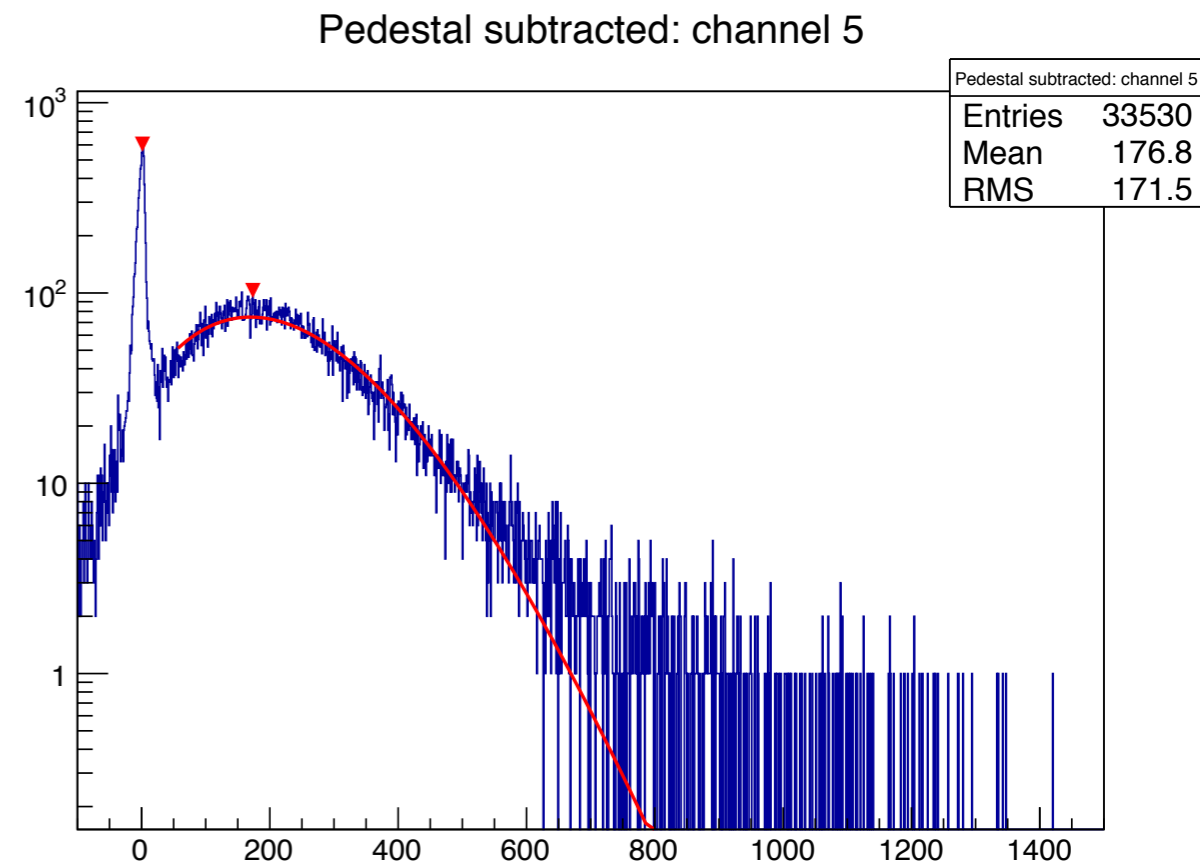
- Started: False
- EventCount: 39593 events
- EventRate: 0 Hz
- OutputType: CharData

RunControlService:

- Running: False
- Paused: False

Our beta source

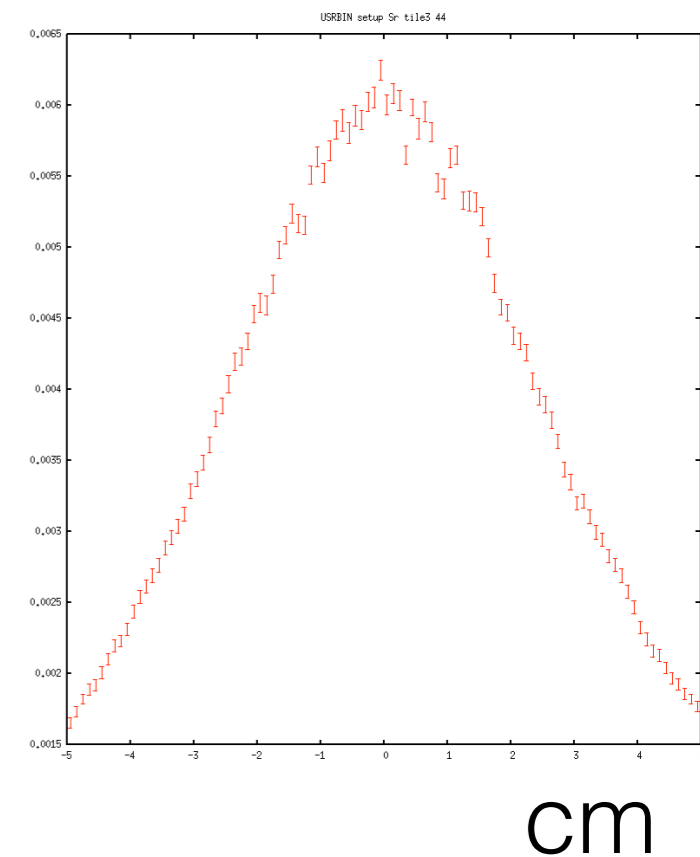
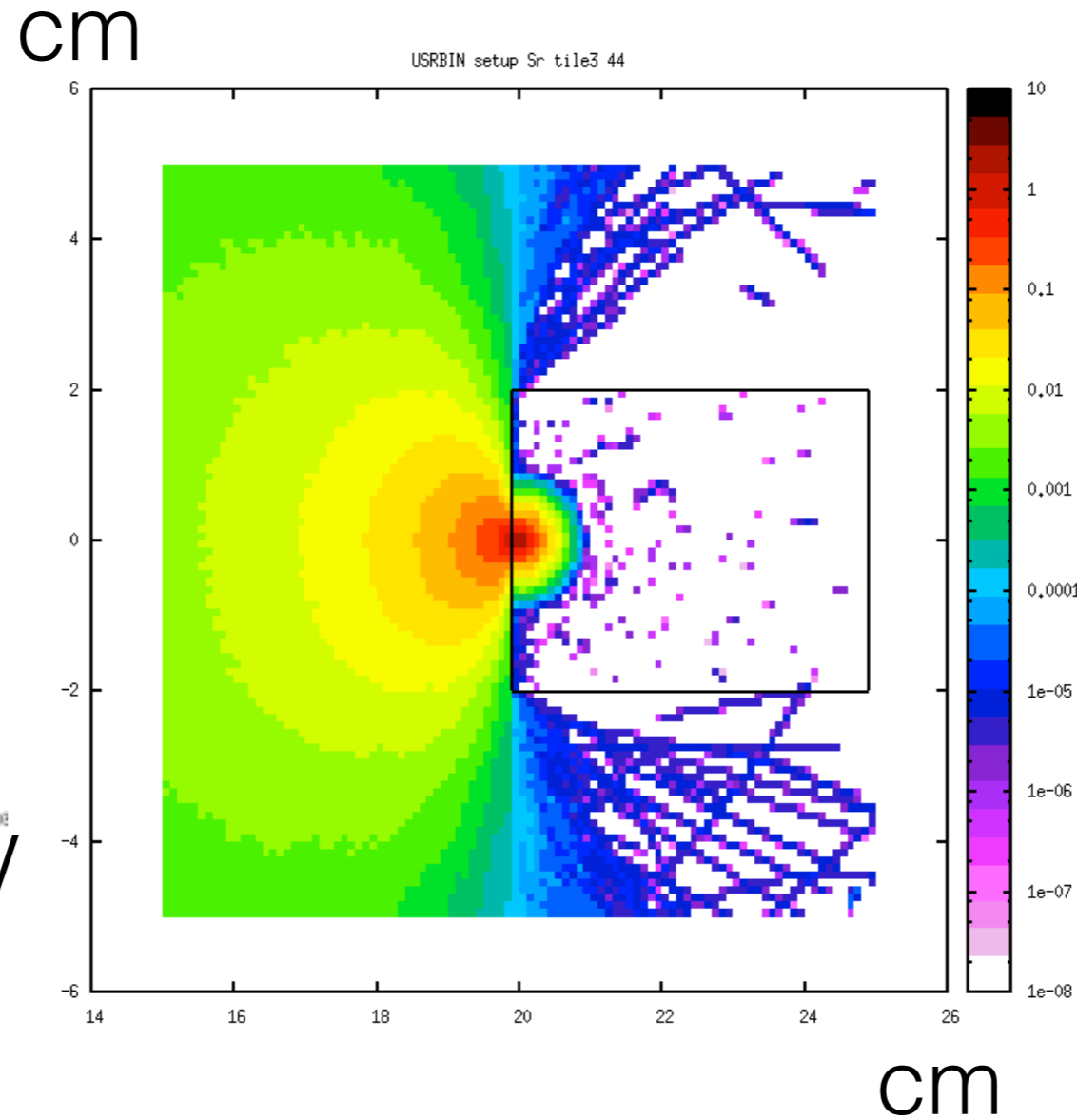
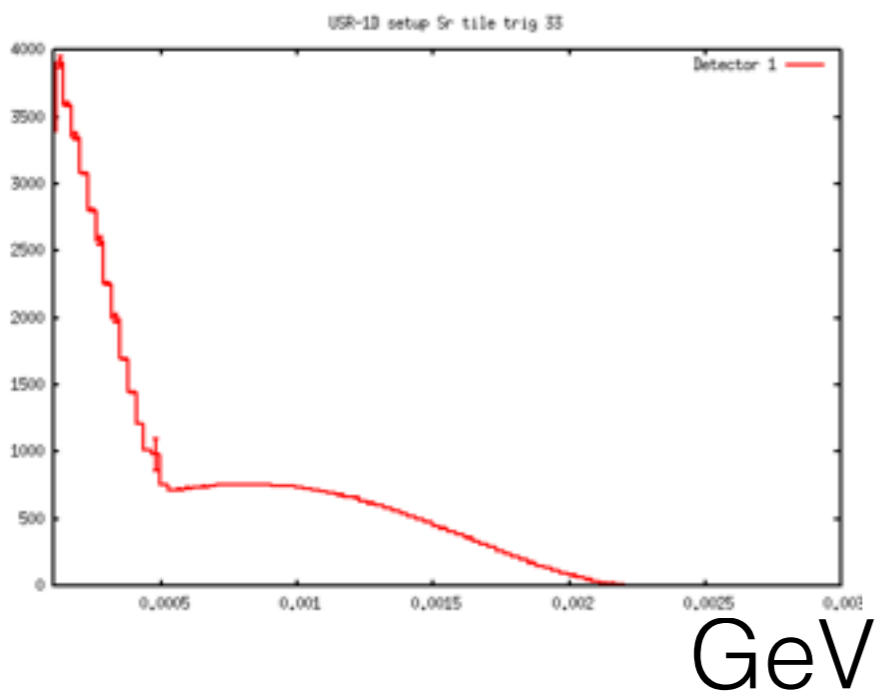
- A discussion done during last meeting suggested to improve the collimation of our Sr90 source.
- S. Fiore setup a Fluka Montecarlo describing our source and measurement setup. We discussed and tested some options.



large tail below pedestal

First simulations: Our source

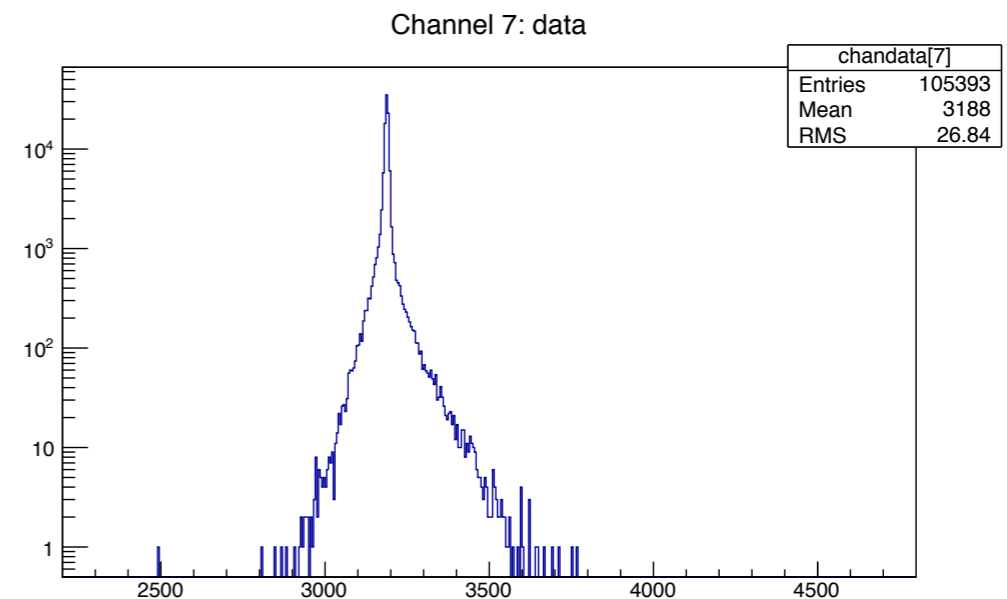
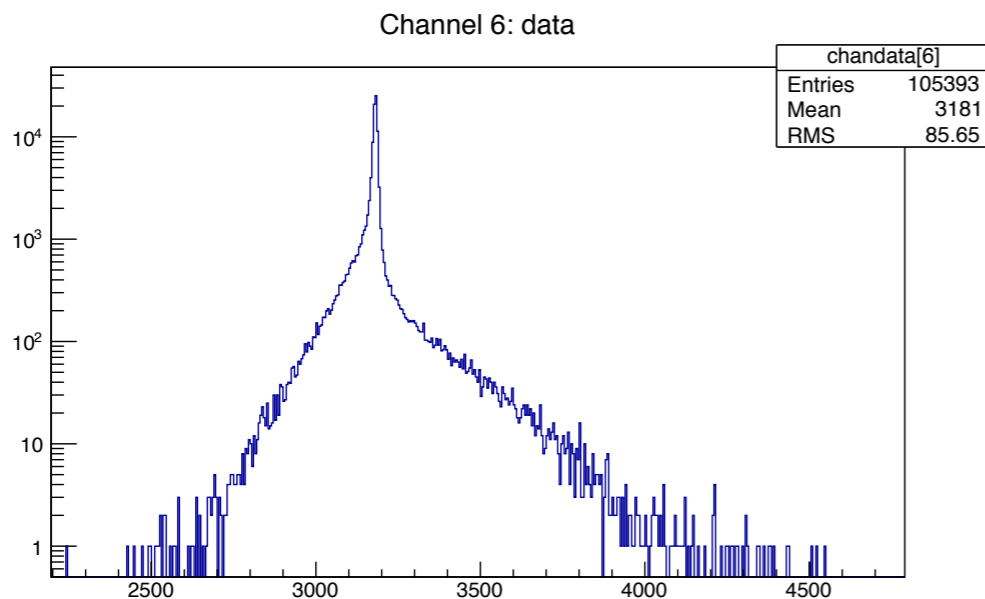
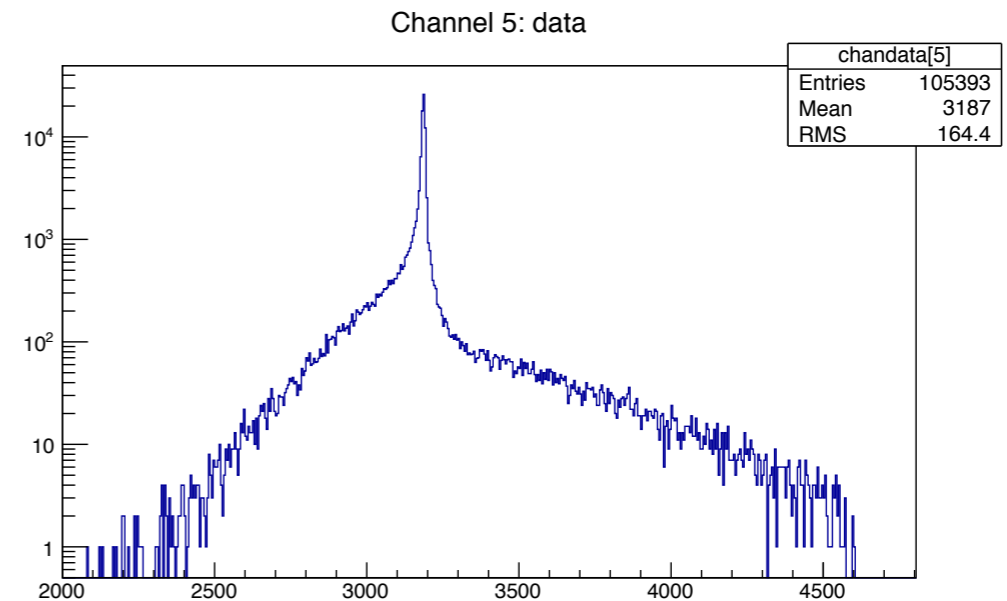
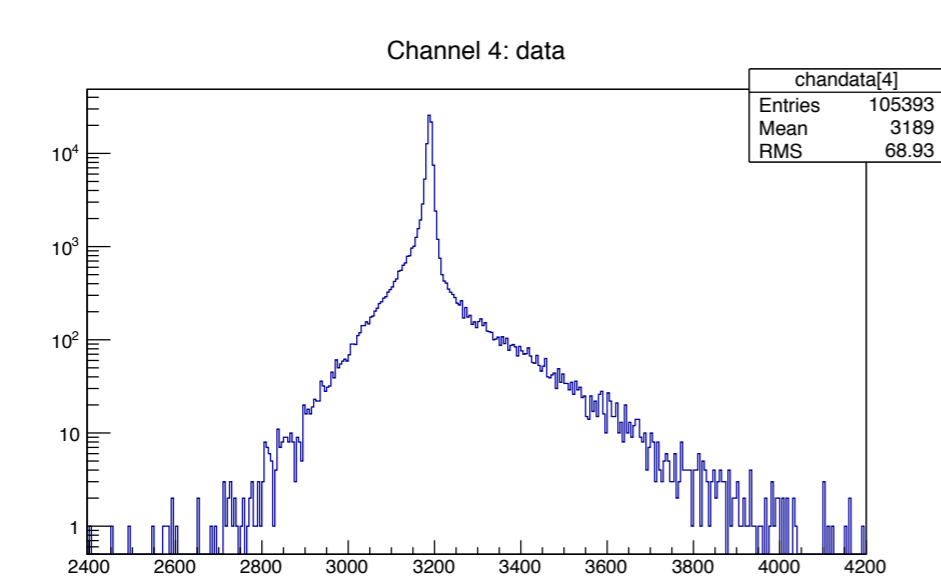
Sr90-Y90 Fluka
spectra



- simulation showing uncollimated source illumination

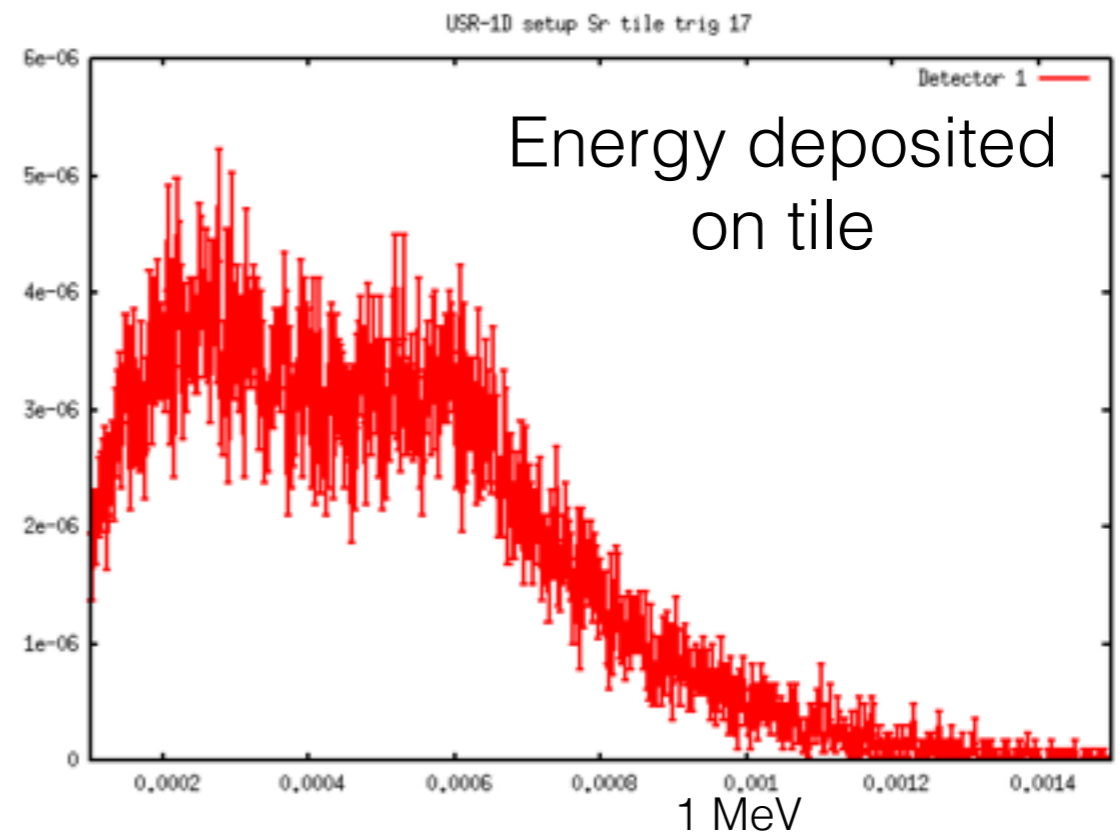
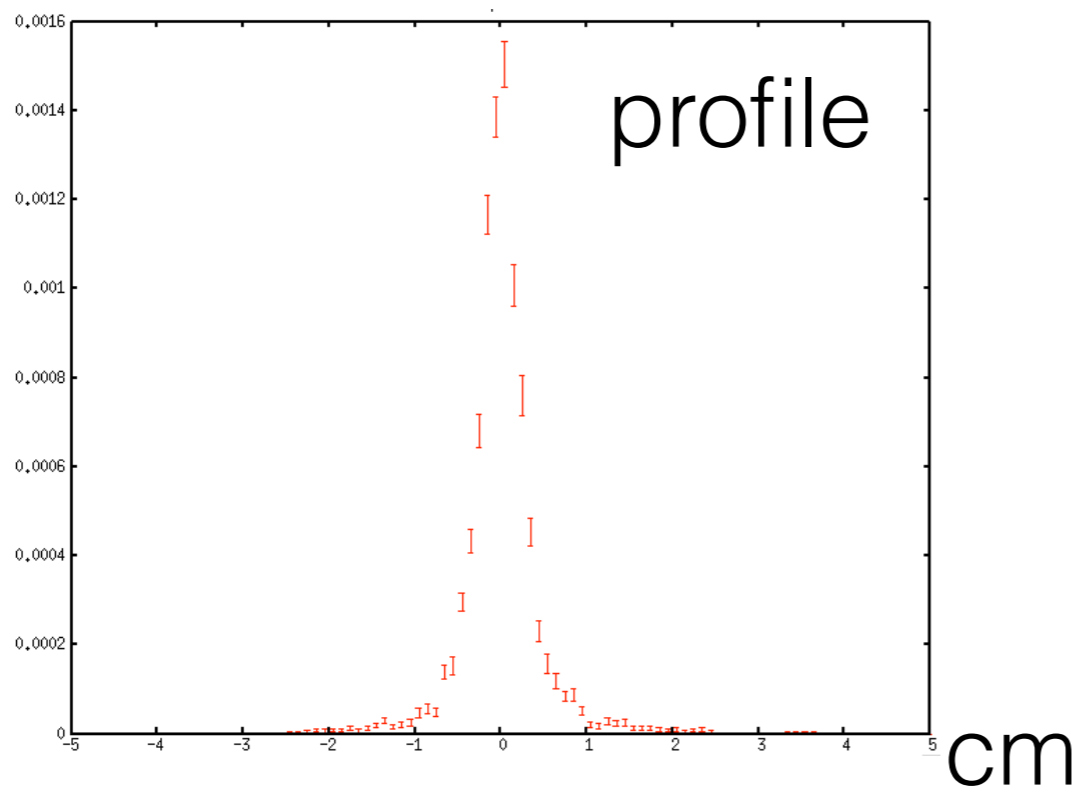
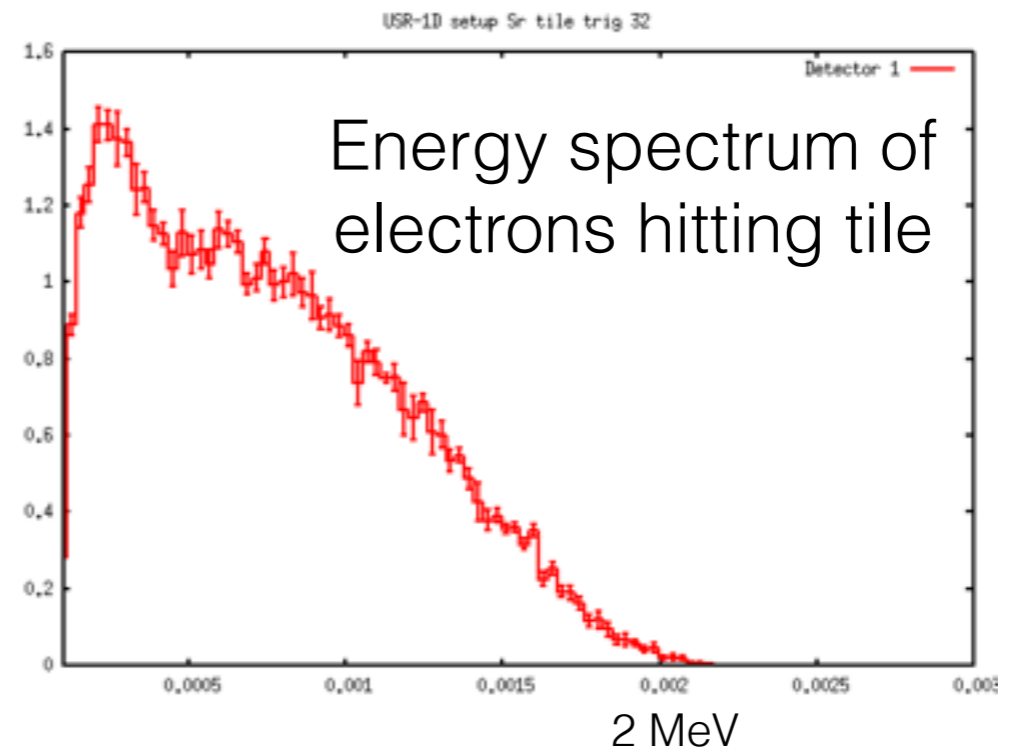
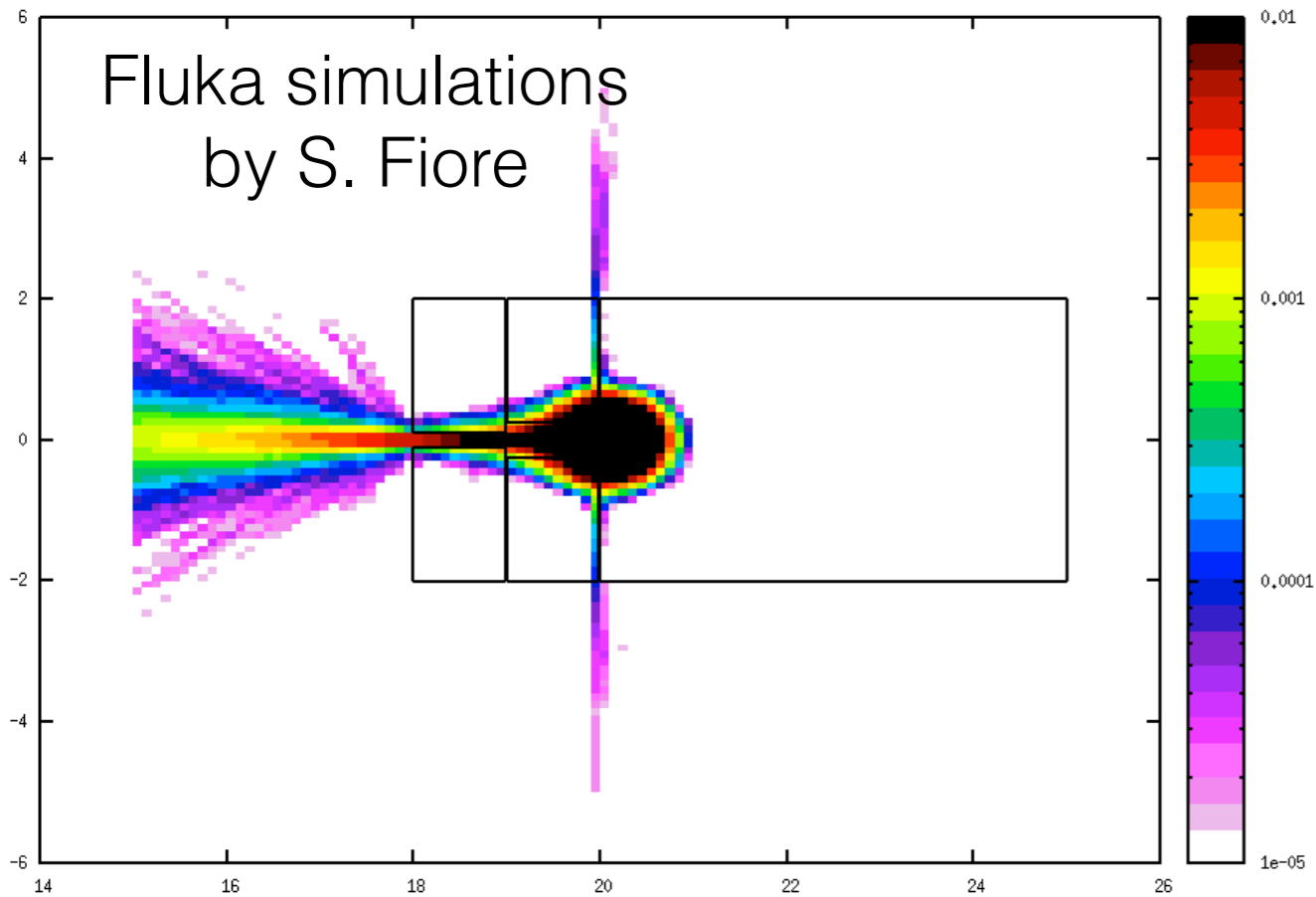
Old collimation: random triggers

1cm plastic, 5mm hole

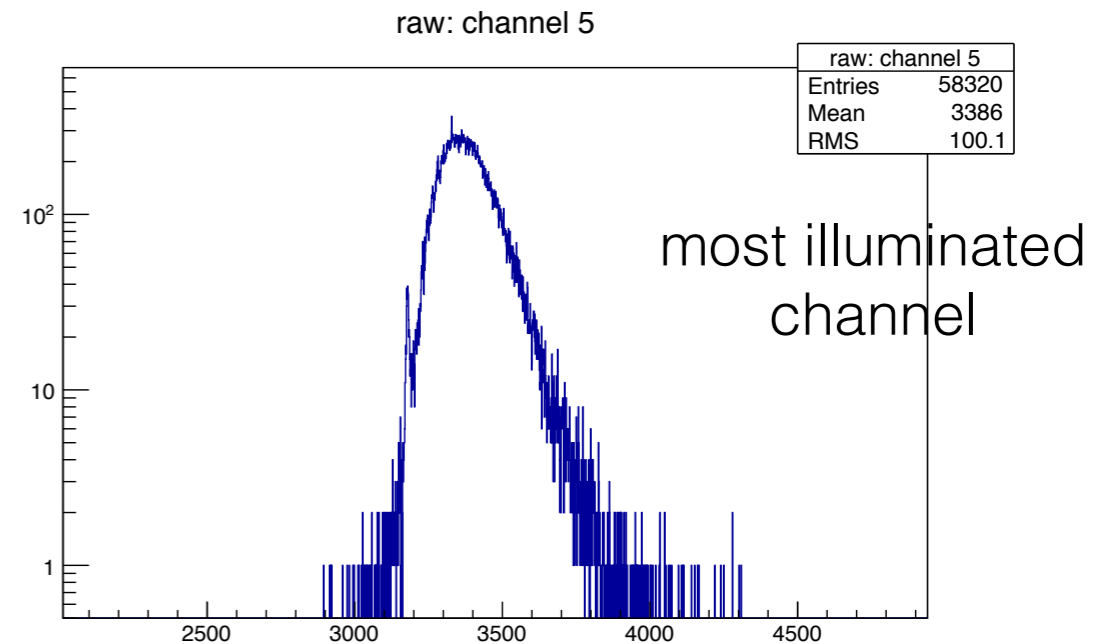
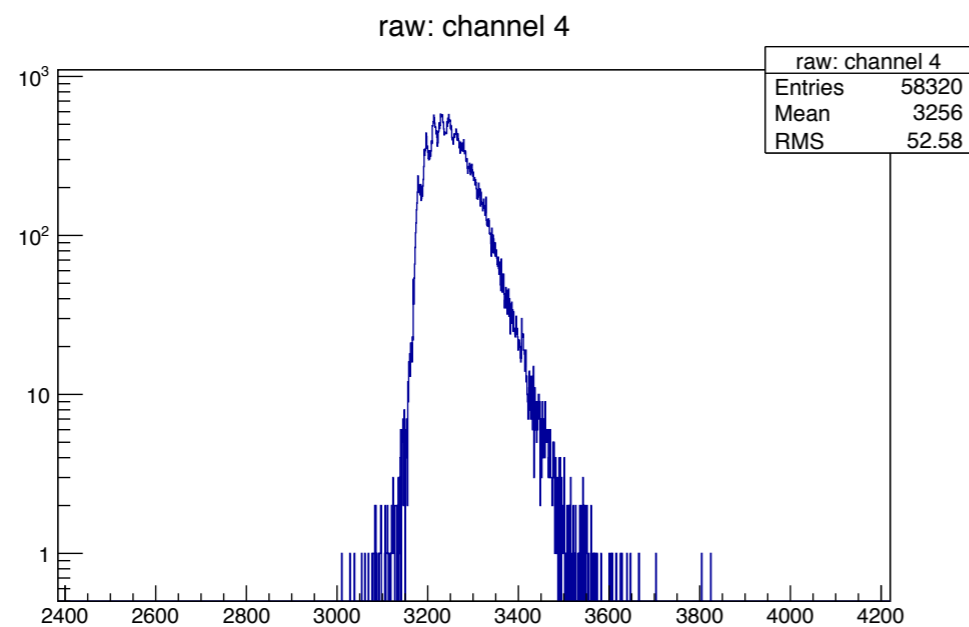


- large tails around pedestal have been measured (150227_run002)

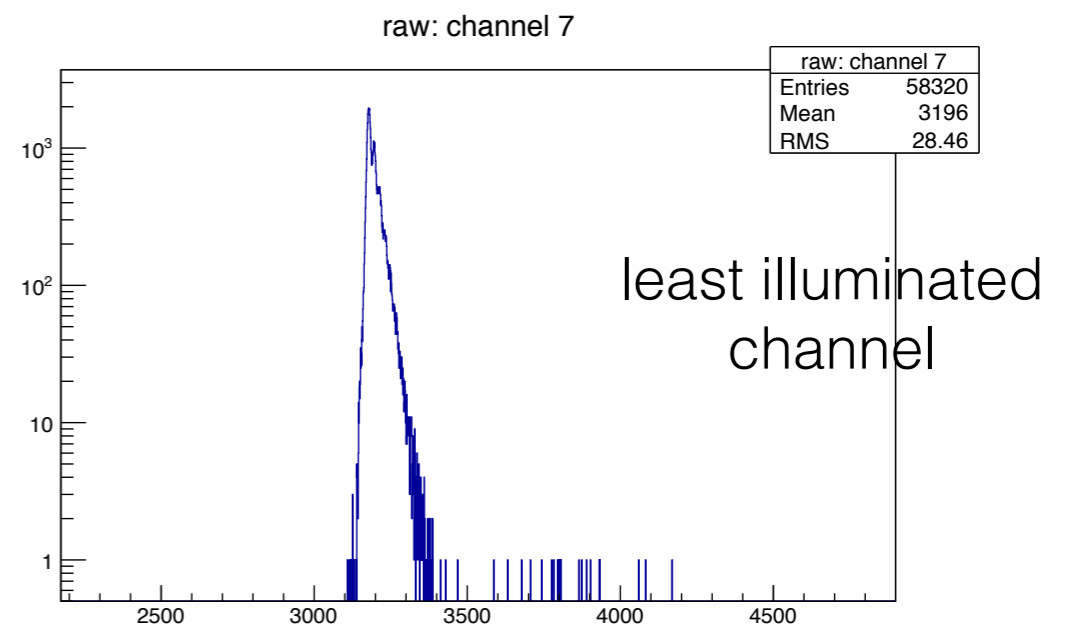
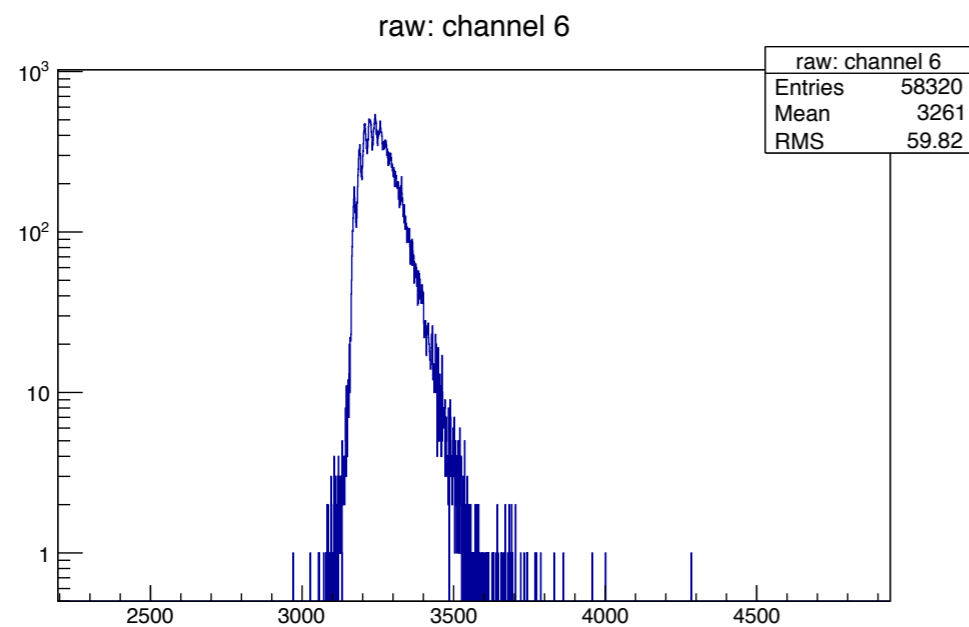
Source Collimation



New collimation: data



1cm plastic, 5mm hole + 1cm plastic, 2mm hole



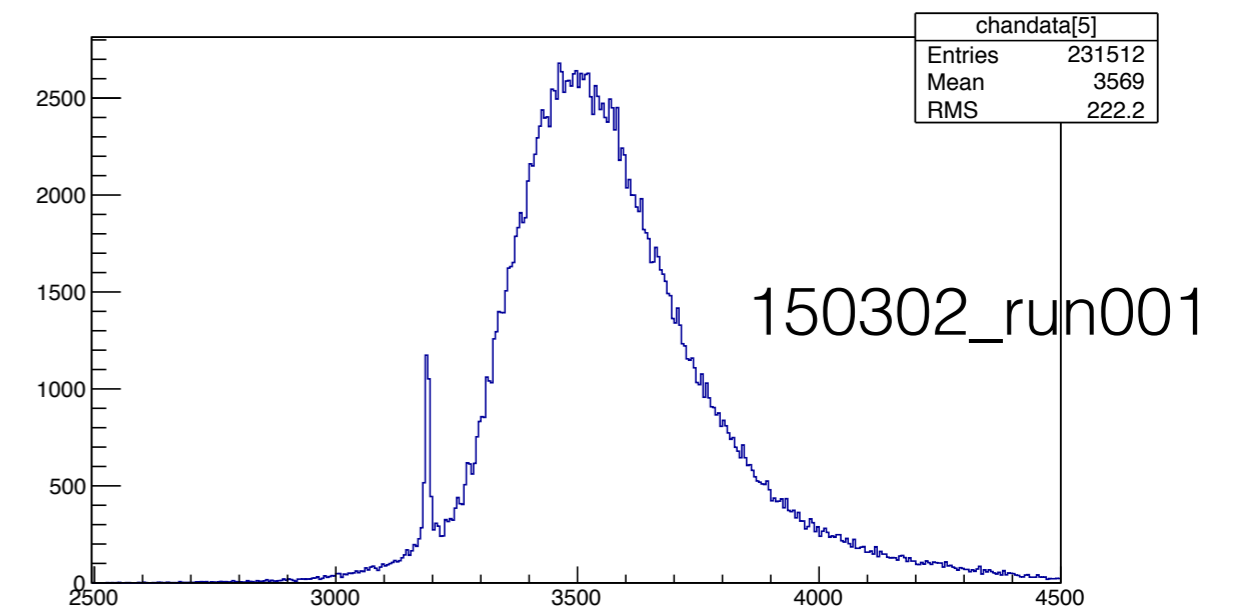
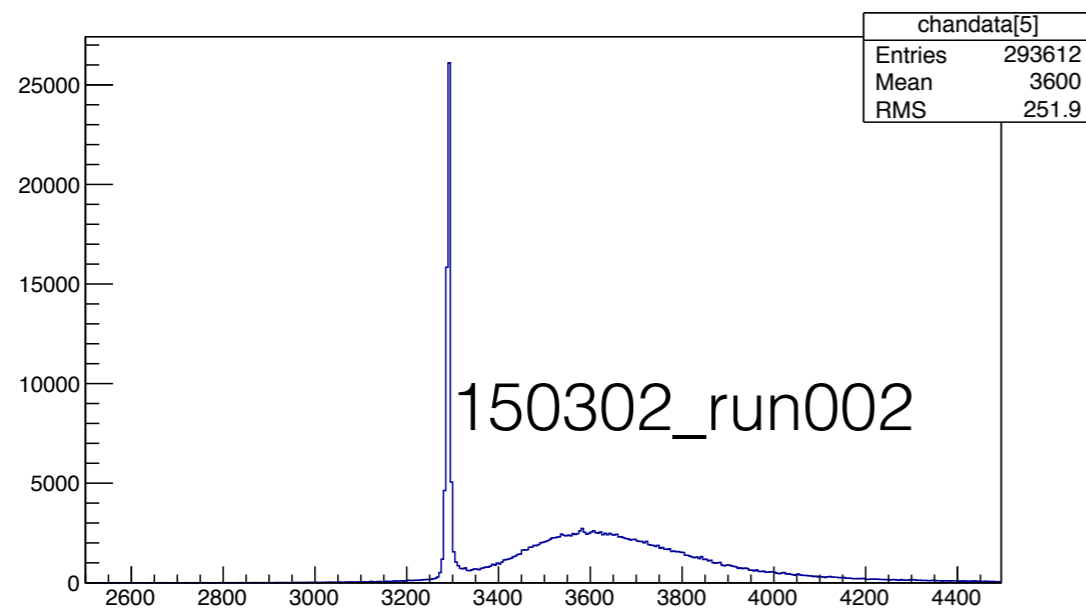
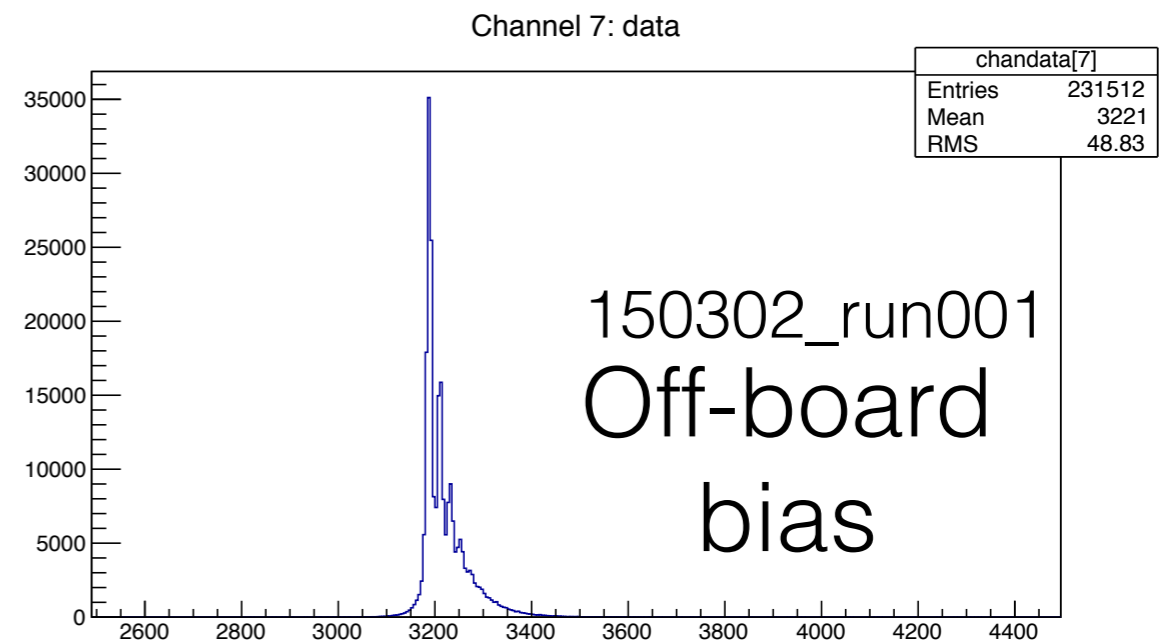
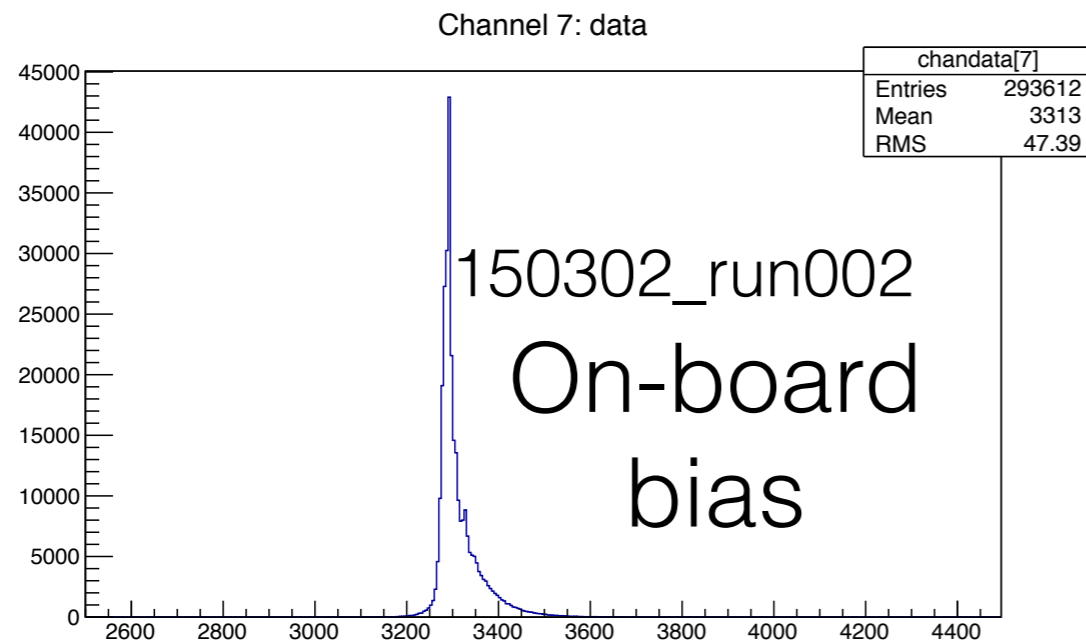
- Collimation is effective to cleanup our spectra

Pedestal

- During last presentation, a large ratio of pedestal events over total number of events was found, even in case our trigger was selecting electrons in the middle of the tile.

External vs Internal Bias

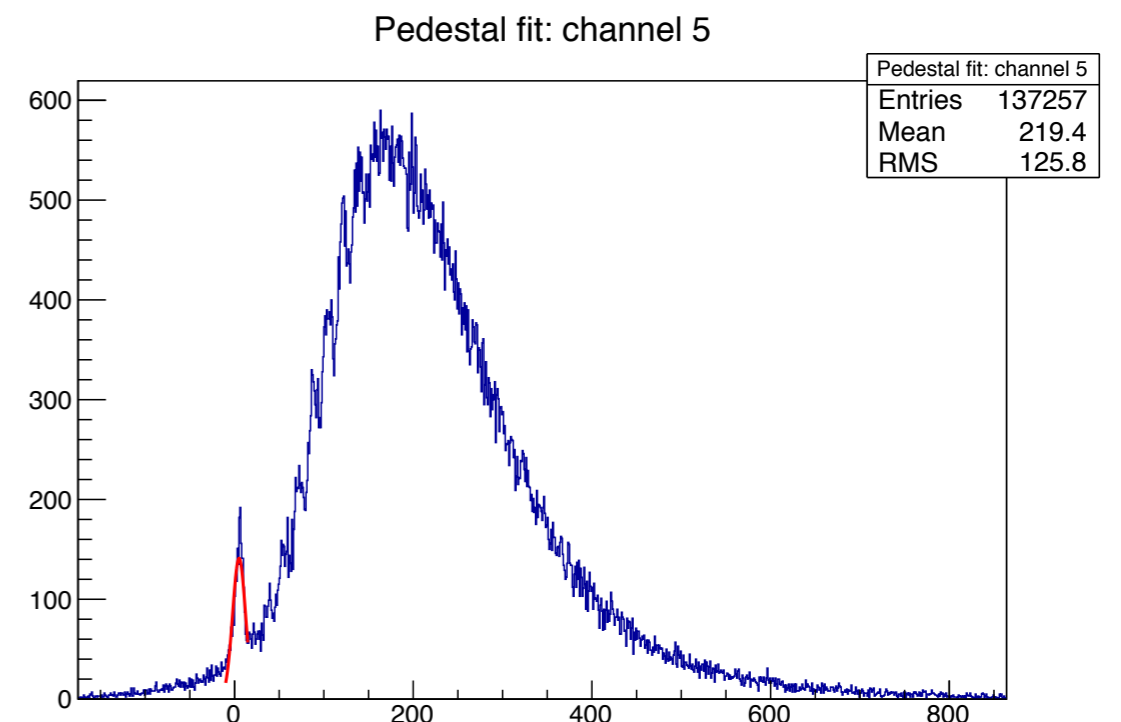
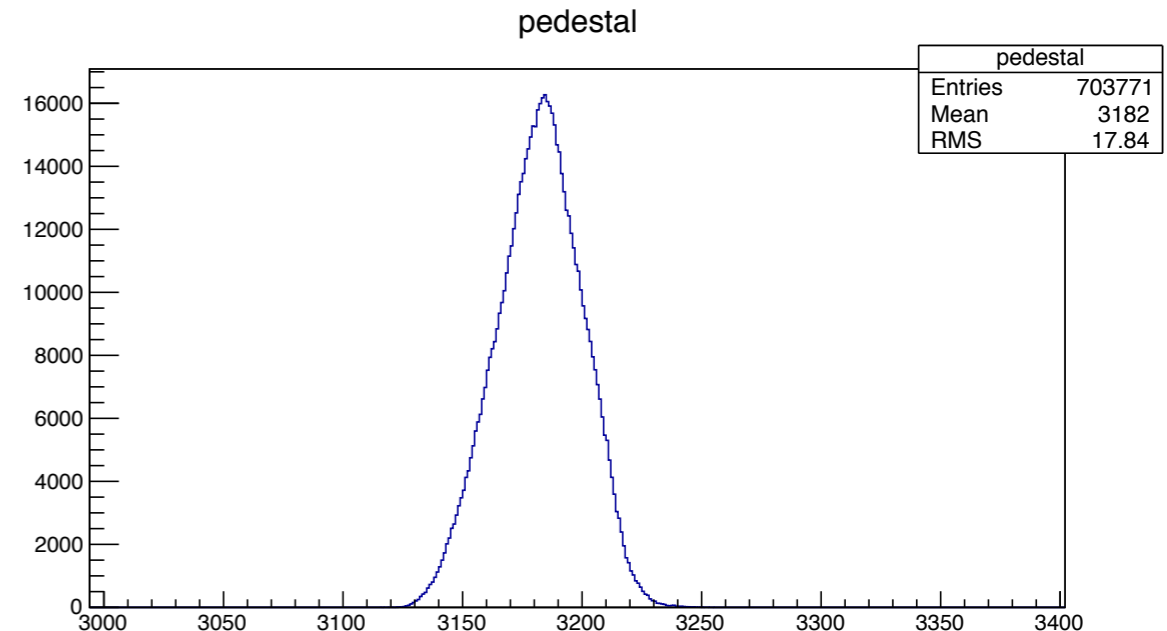
- large pedestal peaks have been observed when on-board bias is used.



- roughly same behaviour if external bias is given with precision power supply or from a second readout board
- it seems that our bias circuit is ok, but some crosstalk is present if bias and front-end live on same board.

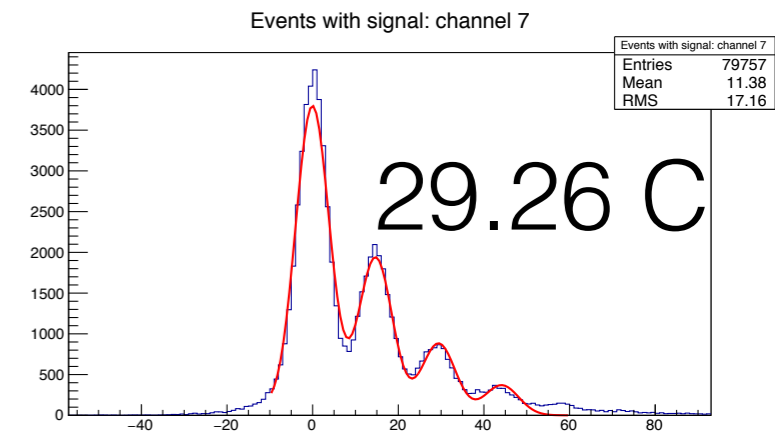
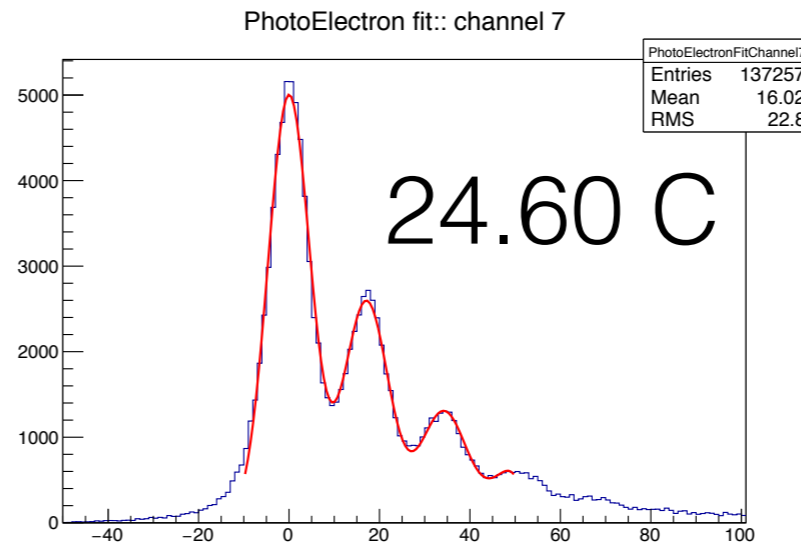
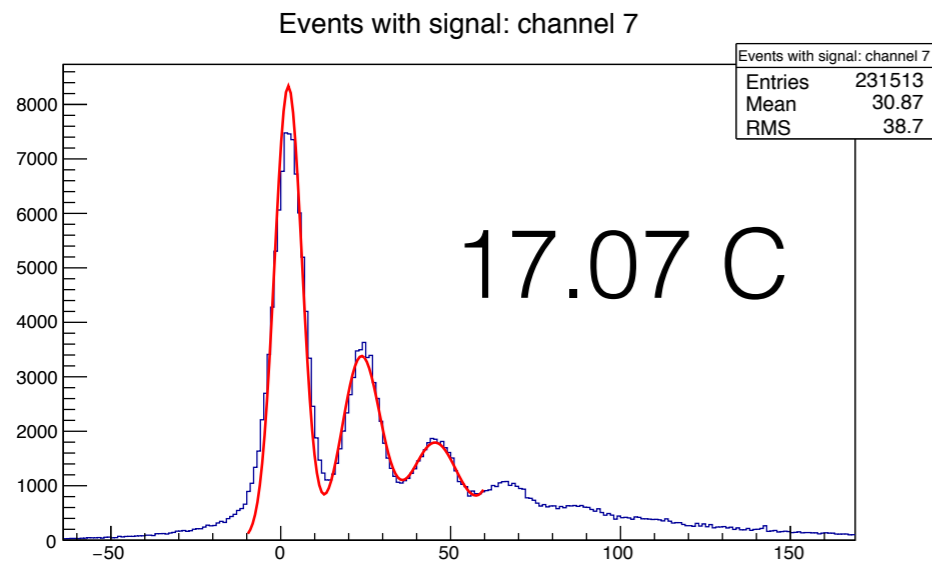
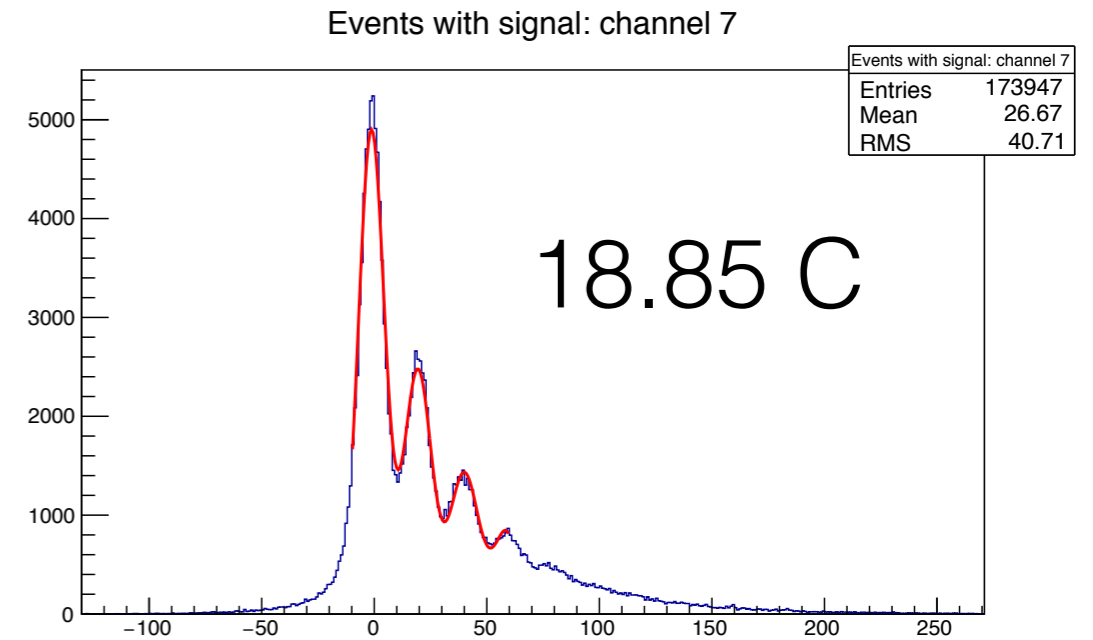
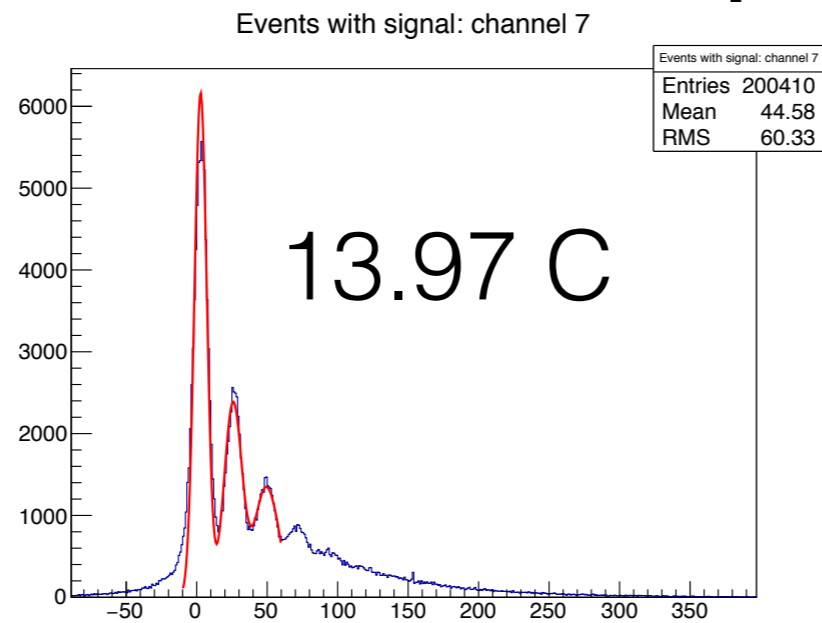
New scans: data analysis

- the 32 input EASIROC inputs are divided into « signal inputs » and « pedestal inputs ».
- an event-by-event average pedestal is calculated using « pedestal inputs » and subtracted to all channels to reduce noise. New histogram is produced.
- From each new histogram, a channel pedestal is fit and subtracted to raw data.
- resulting final distributions are used for calibration and measurements.

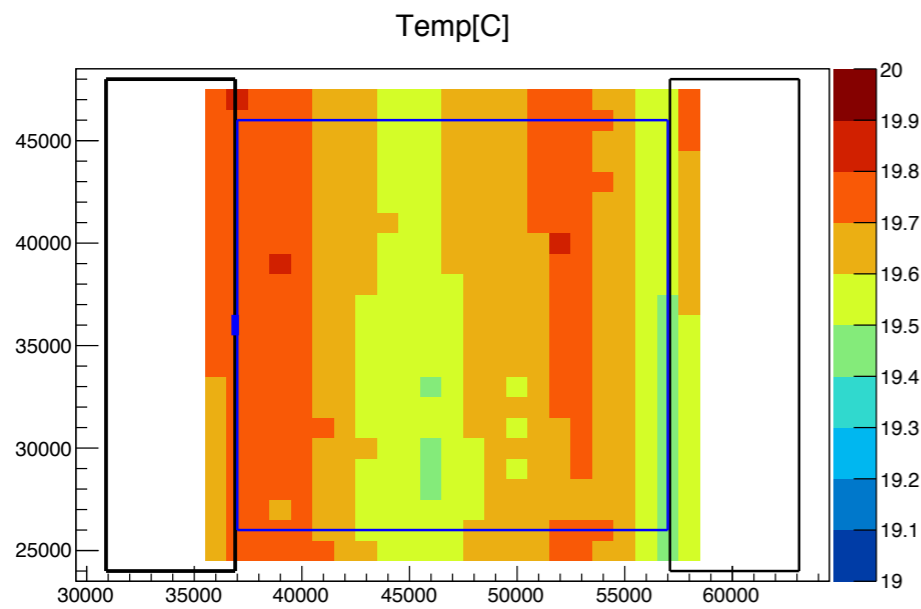
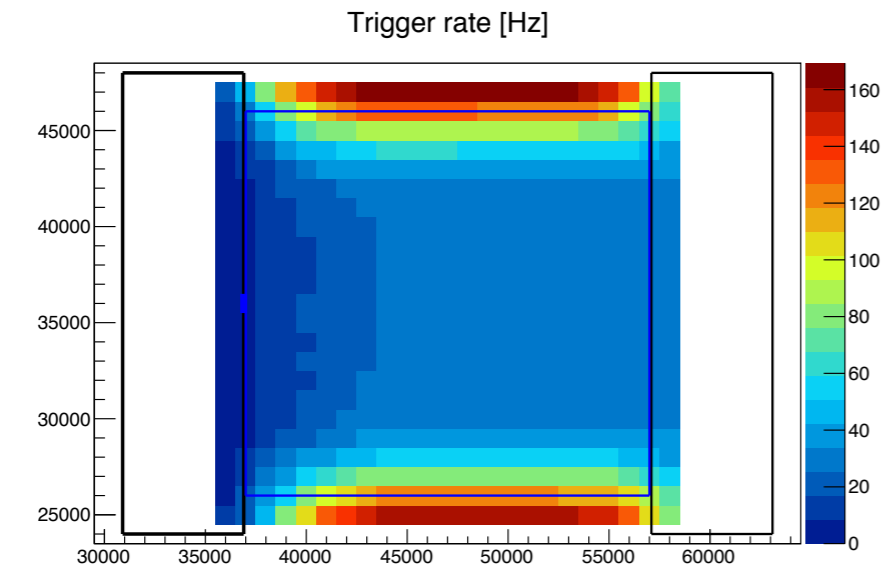


Calibration procedure

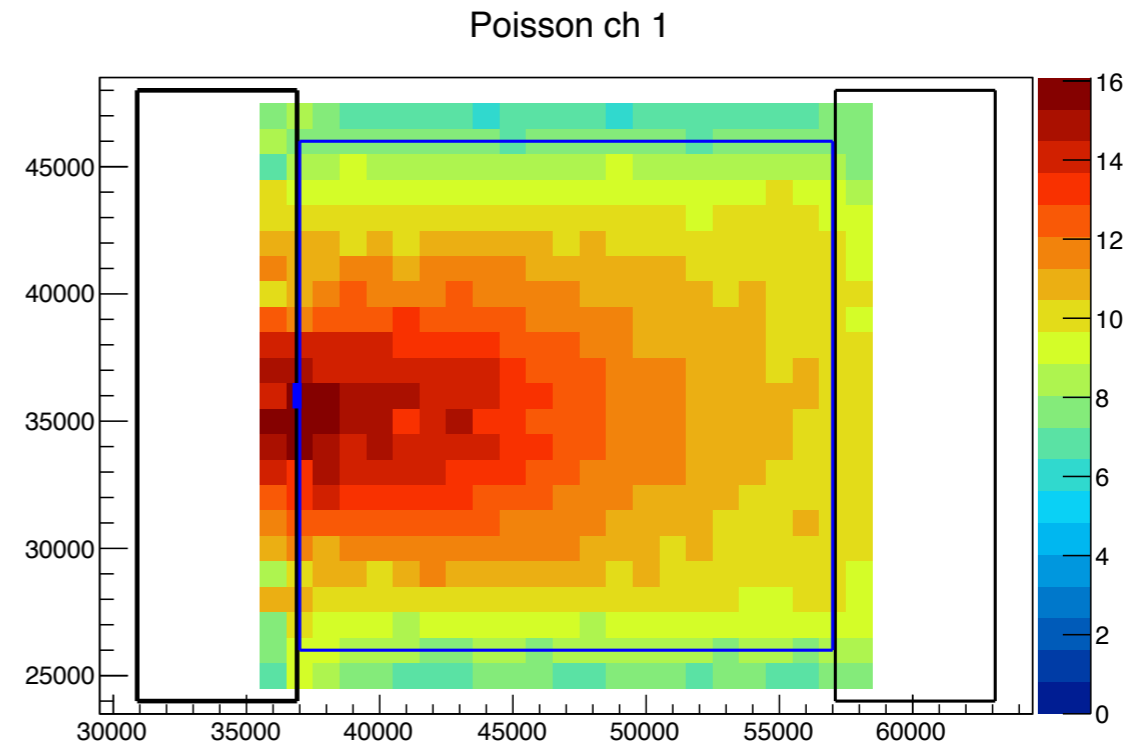
- calibration values range 13-24 ADC counts/P.E.
- bad $P(\chi^2)$: need to improve on fitting



Scan results, external trigger



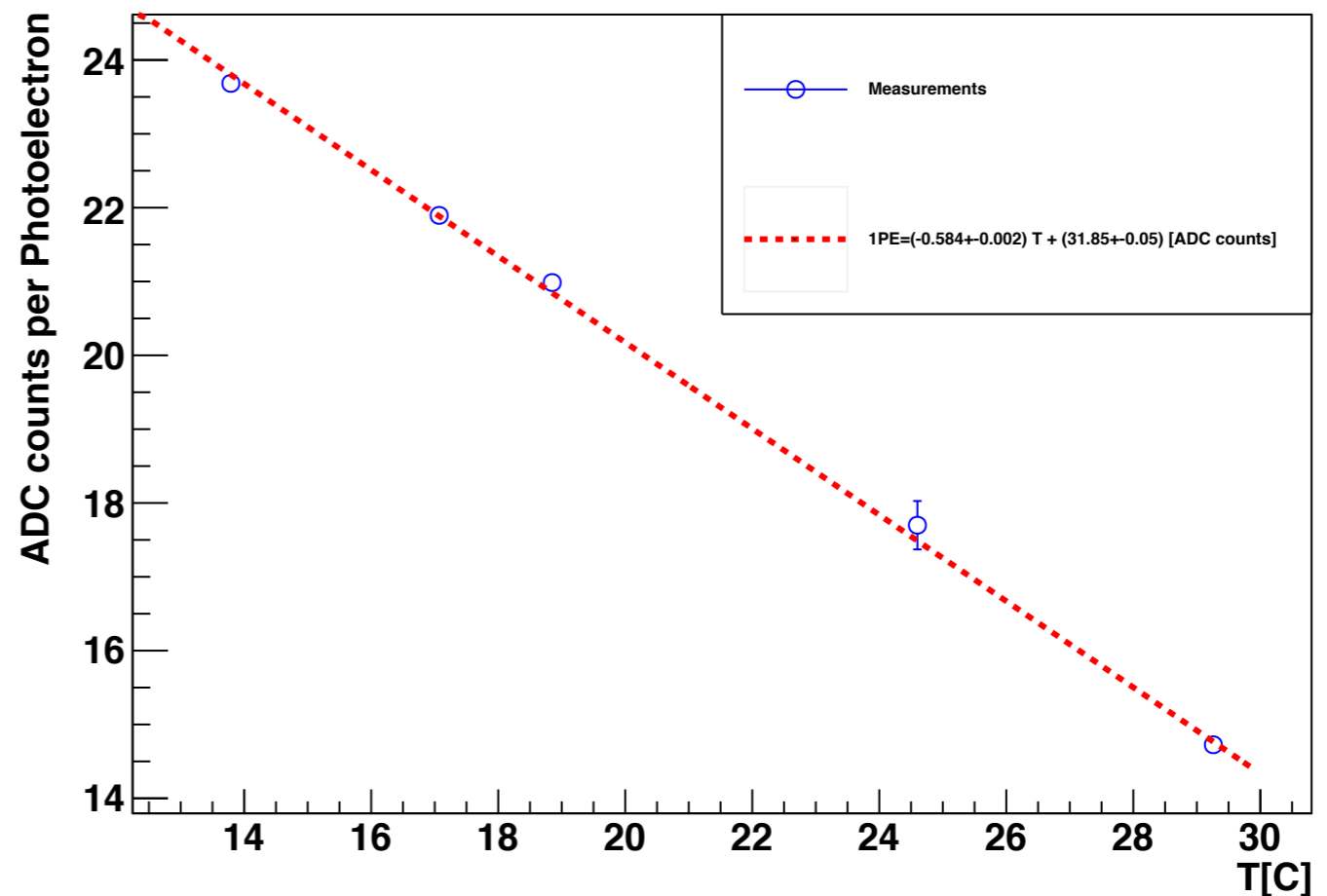
300s/point



Gain vs Temperature

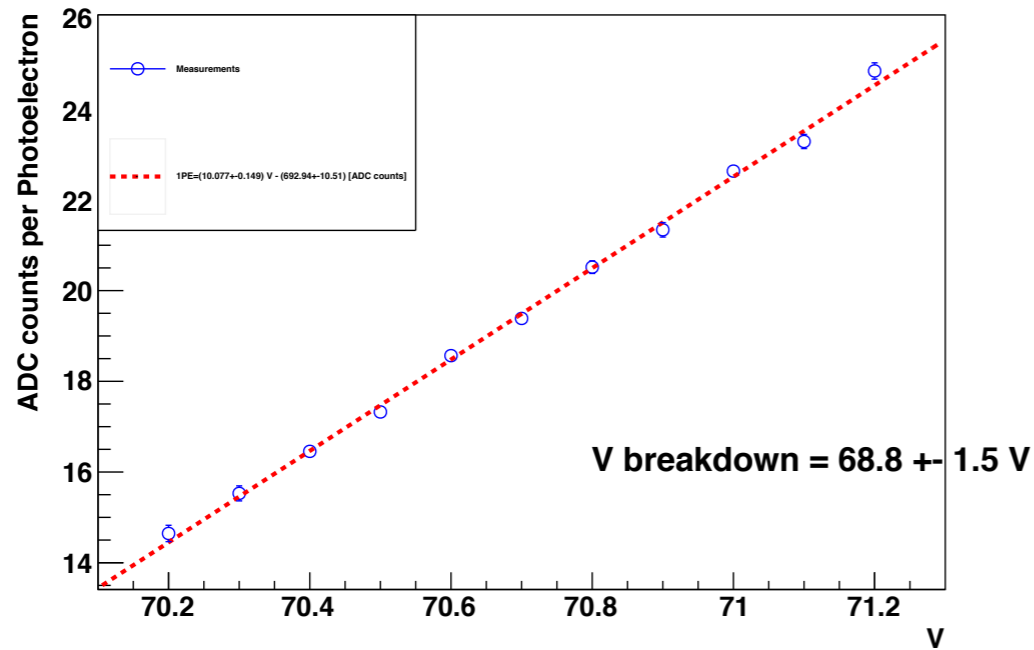
- First studies have started to implement an active control of bias voltage.

Calibration vs Temperature: VBias=71.0 V

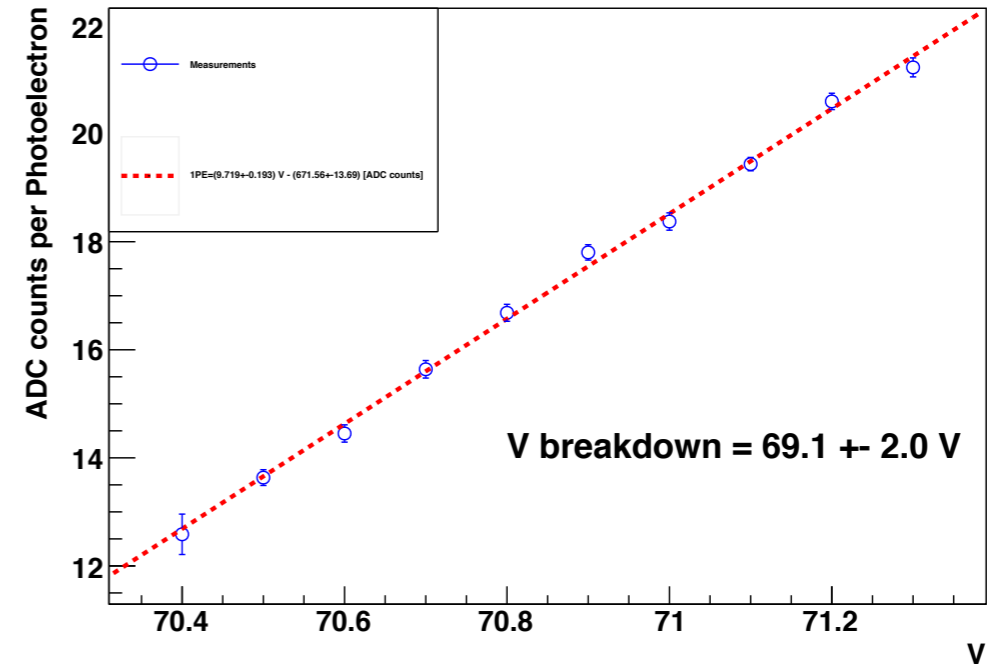


Gain vs Bias

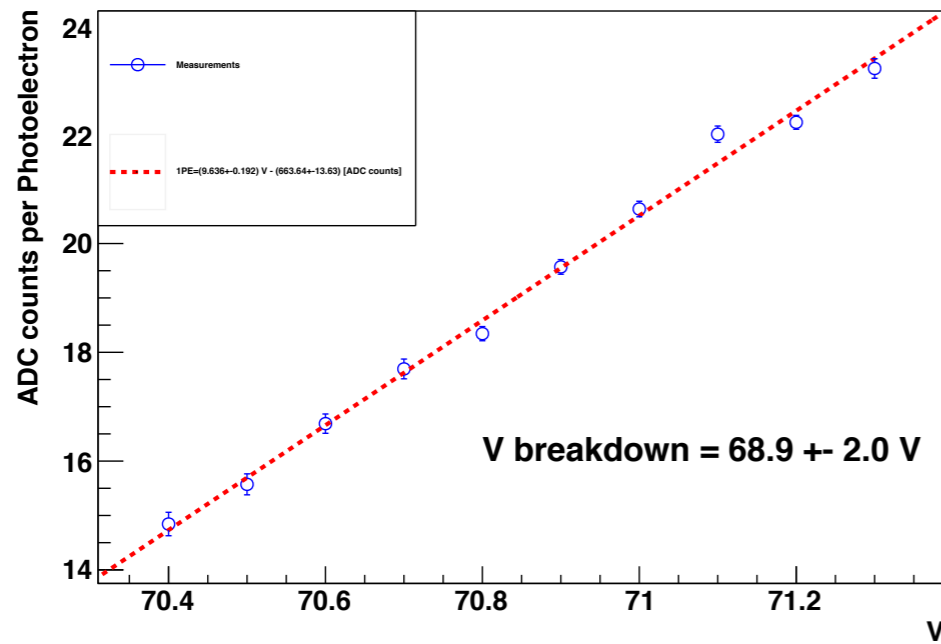
Calibration vs Temperature: T=13.8 C



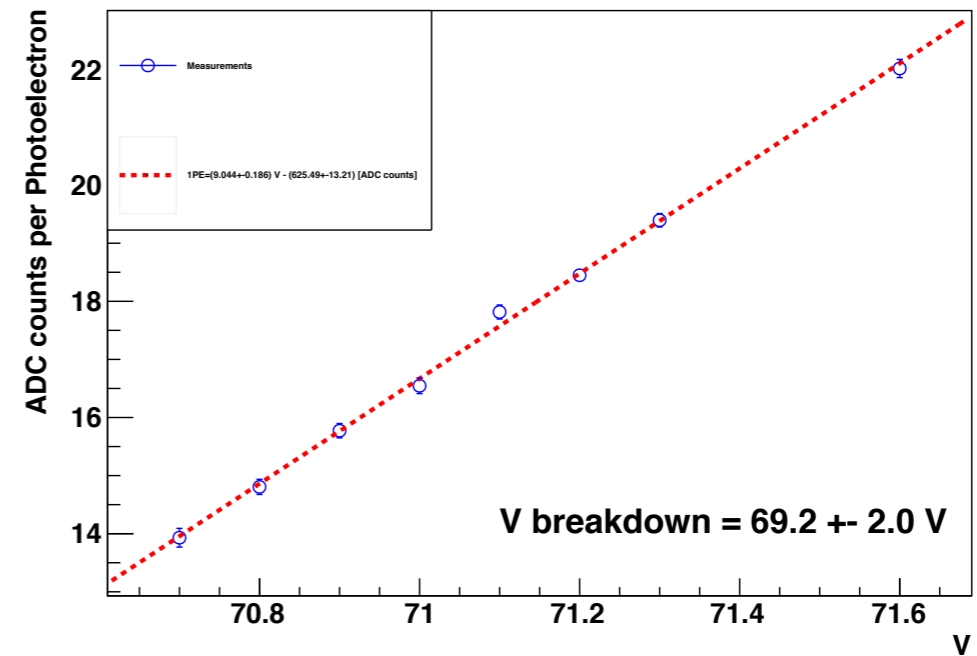
Calibration vs Temperature: T=20 C



Calibration vs Temperature: T=17 C



Calibration vs Temperature: T=23 C

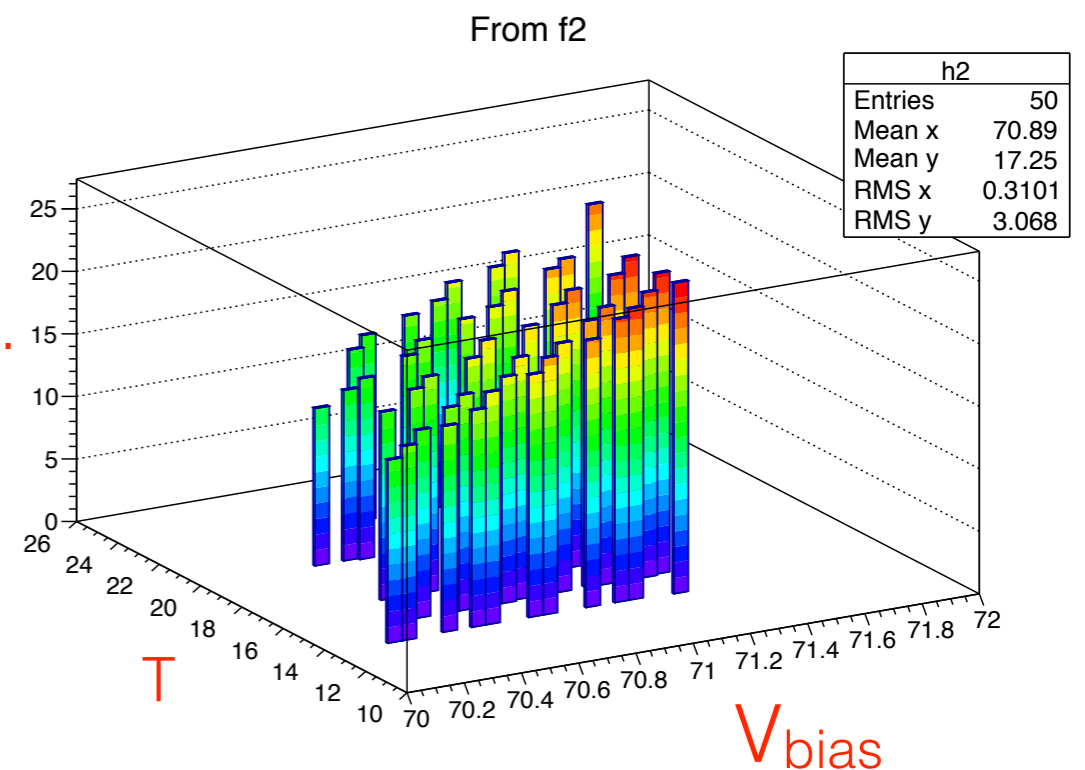


- same analysis used as for XY scan.

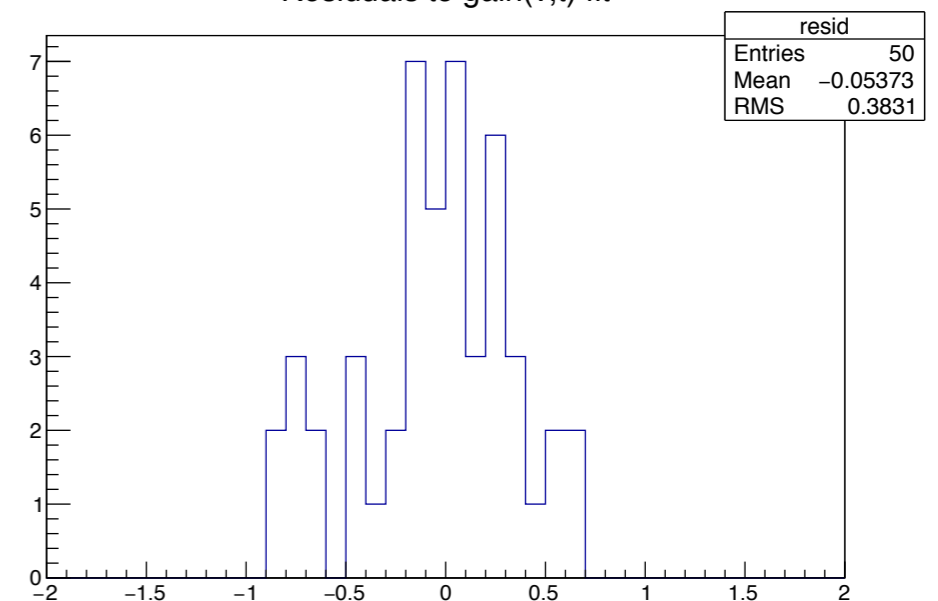
Towards a realtime gain control

- since:
 - Gain = alpha ($V_{\text{bias}} - V_{\text{overvoltage}}$)
 - $V_{\text{overvoltage}} = \text{beta} + \text{gamma} (T)$
- Previous data sets have been used in a common fit to:
 - $\text{Gain}(V,T) = a + b V + c T$
- Choosing an appropriate gain, Voltage corrections vs Temperature can be applied online.
- Current status.
 - Temperature sensors able to achieve better than 0.1 C resolution (better by averaging O(100) measurements).
 - Granularity on Vbias achievable is O(100 mV).
 - A Python script will be our realtime gain control running on ZYNQ, accessing temperature sensor over Ethernet and controlling on-board bias system.

ADC counts/P.E.



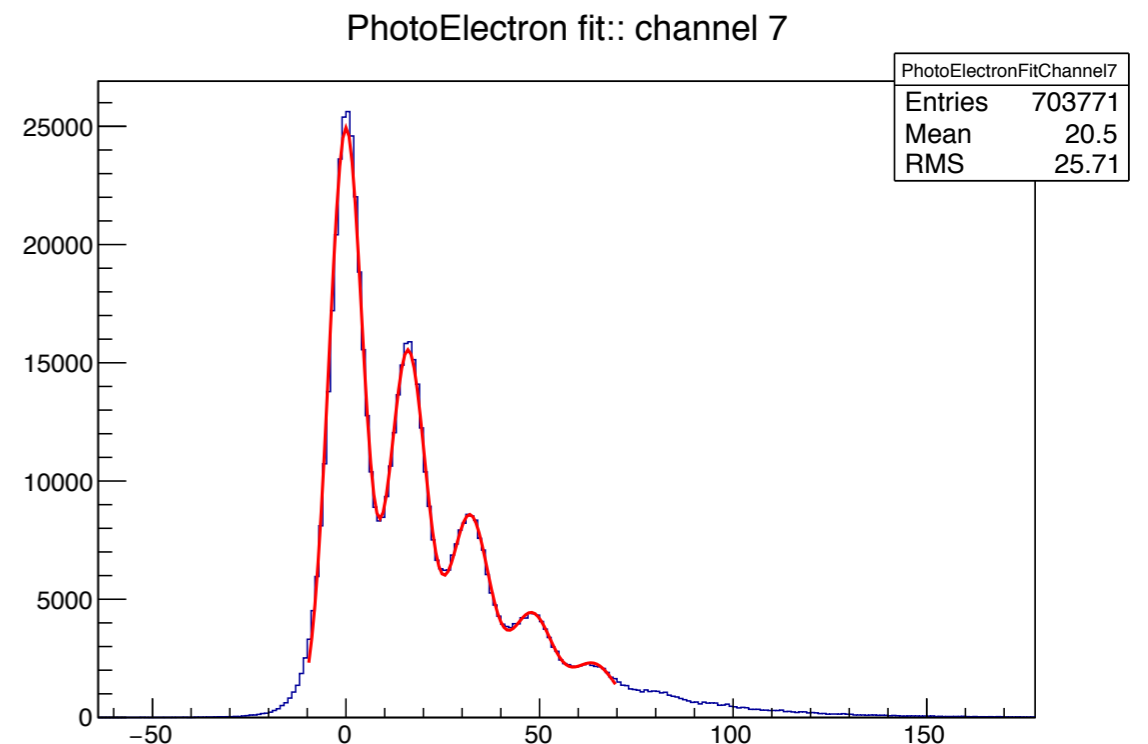
Residuals to gain(v,t) fit



ADC counts

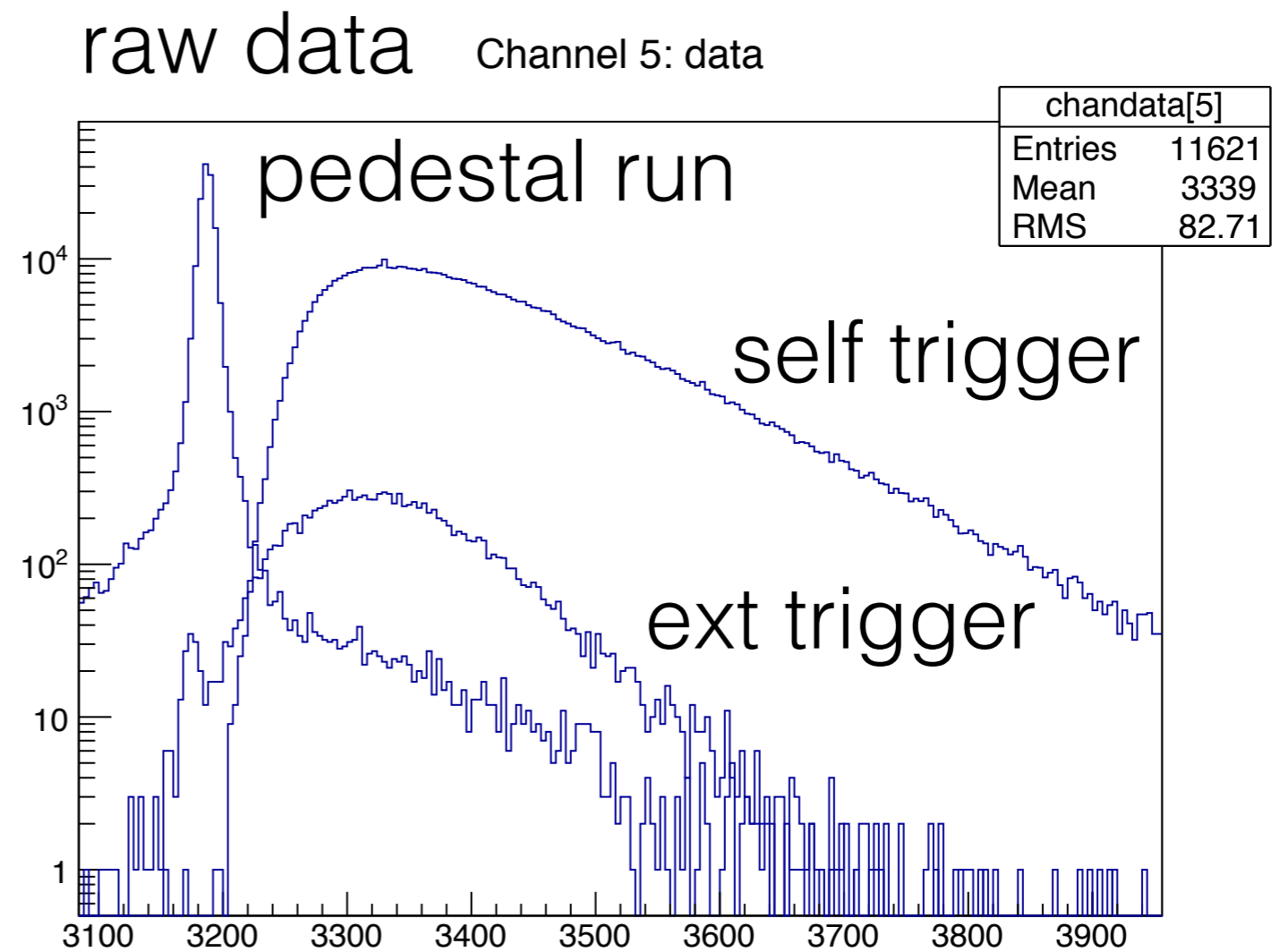
Fine scan with self-trigger

- trying to improve trigger rate 10 Hz -> 10 kHz by using self-trigger capabilities of EASIROC.
- trigger on most illuminated SiPM signal.
- Distributions are now different. Pedestal values measured from dedicated pedestal run using a random trigger.
- calibration constant from long run at T and V scan values from poorly illuminated SiPM (channel 3).



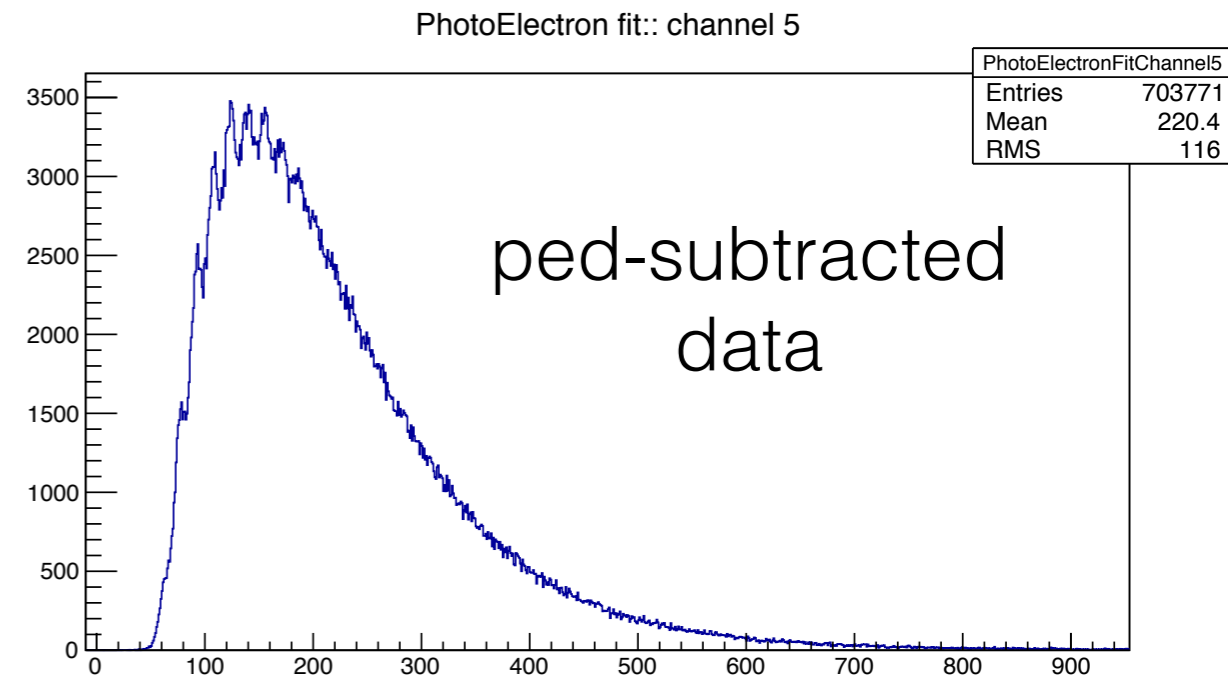
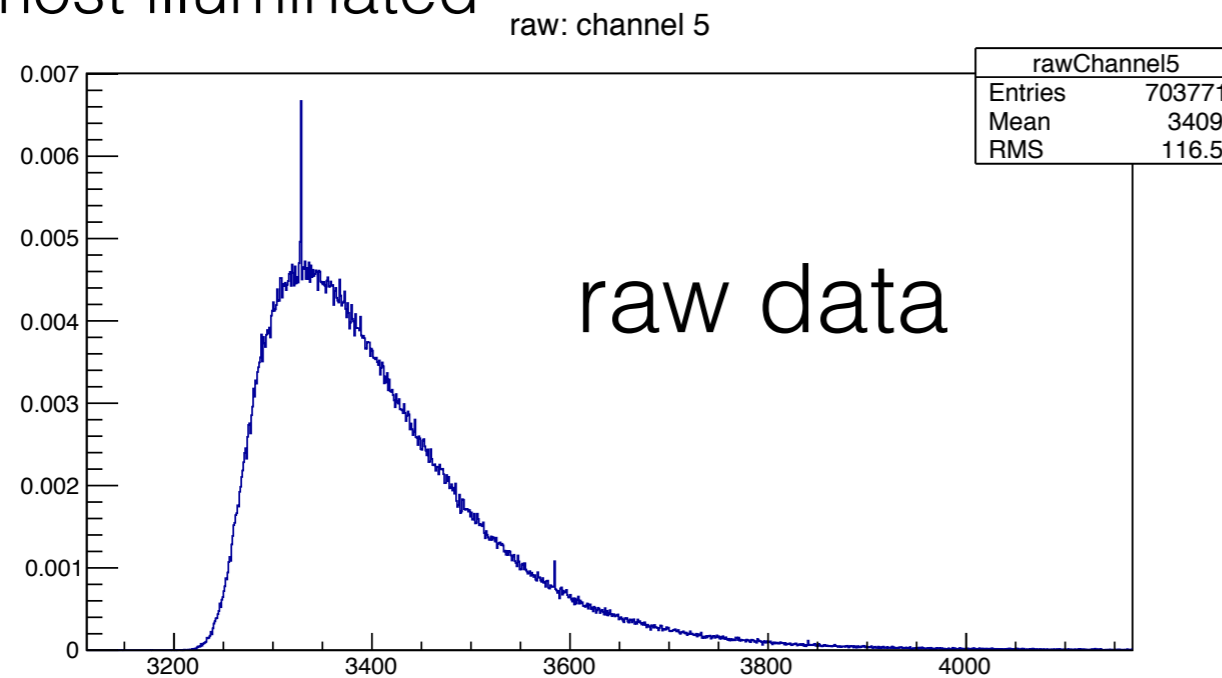
Self-triggered event sample

- XY scan is obtained by plotting average of distribution, in units of P.E.
- Threshold is set just above pedestal, see figure.

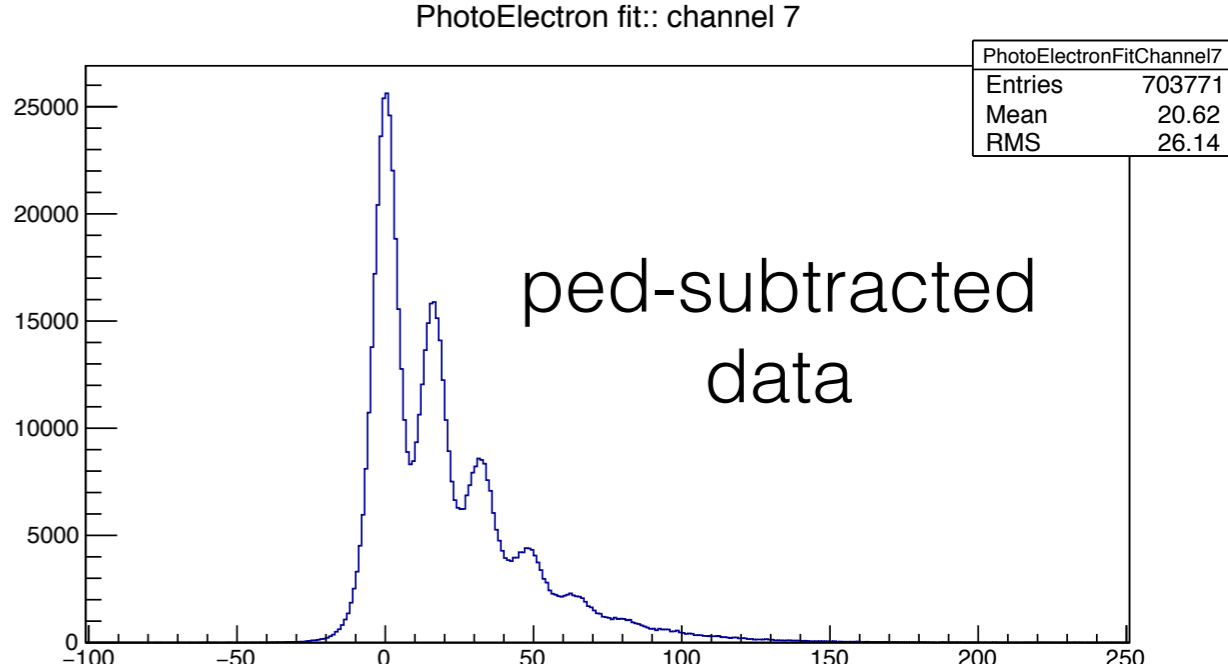
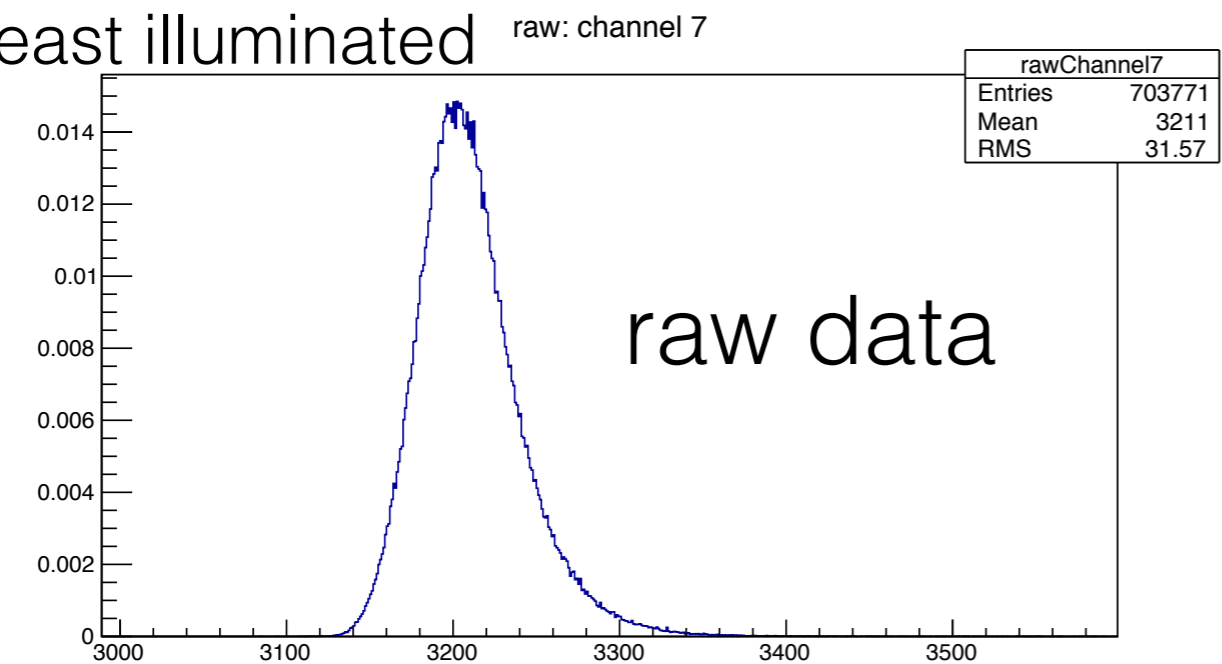


Event-by-event pedestal subtraction

ch5: most illuminated

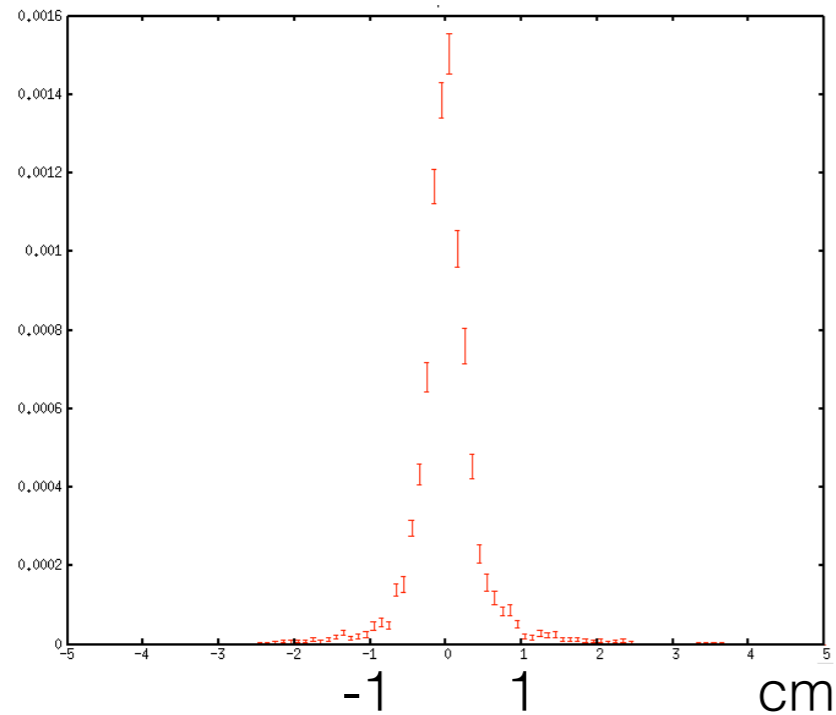


ch7: least illuminated



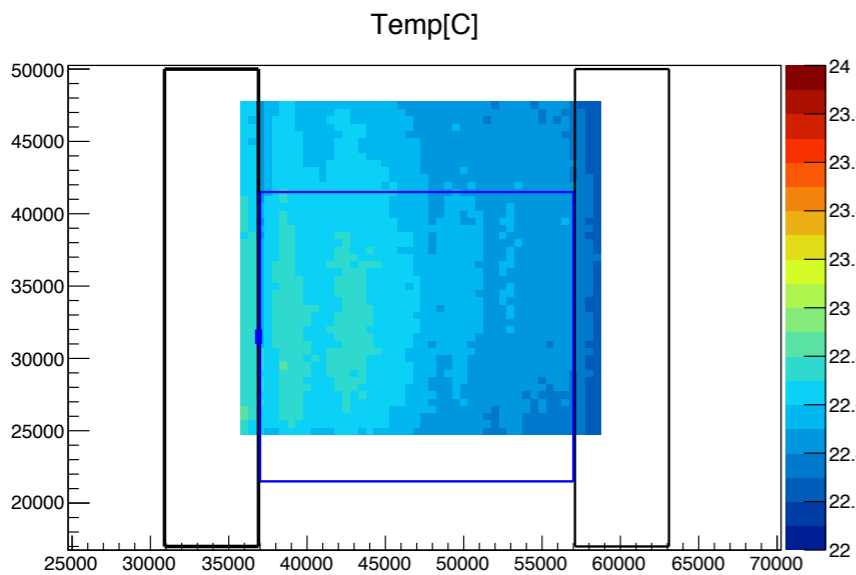
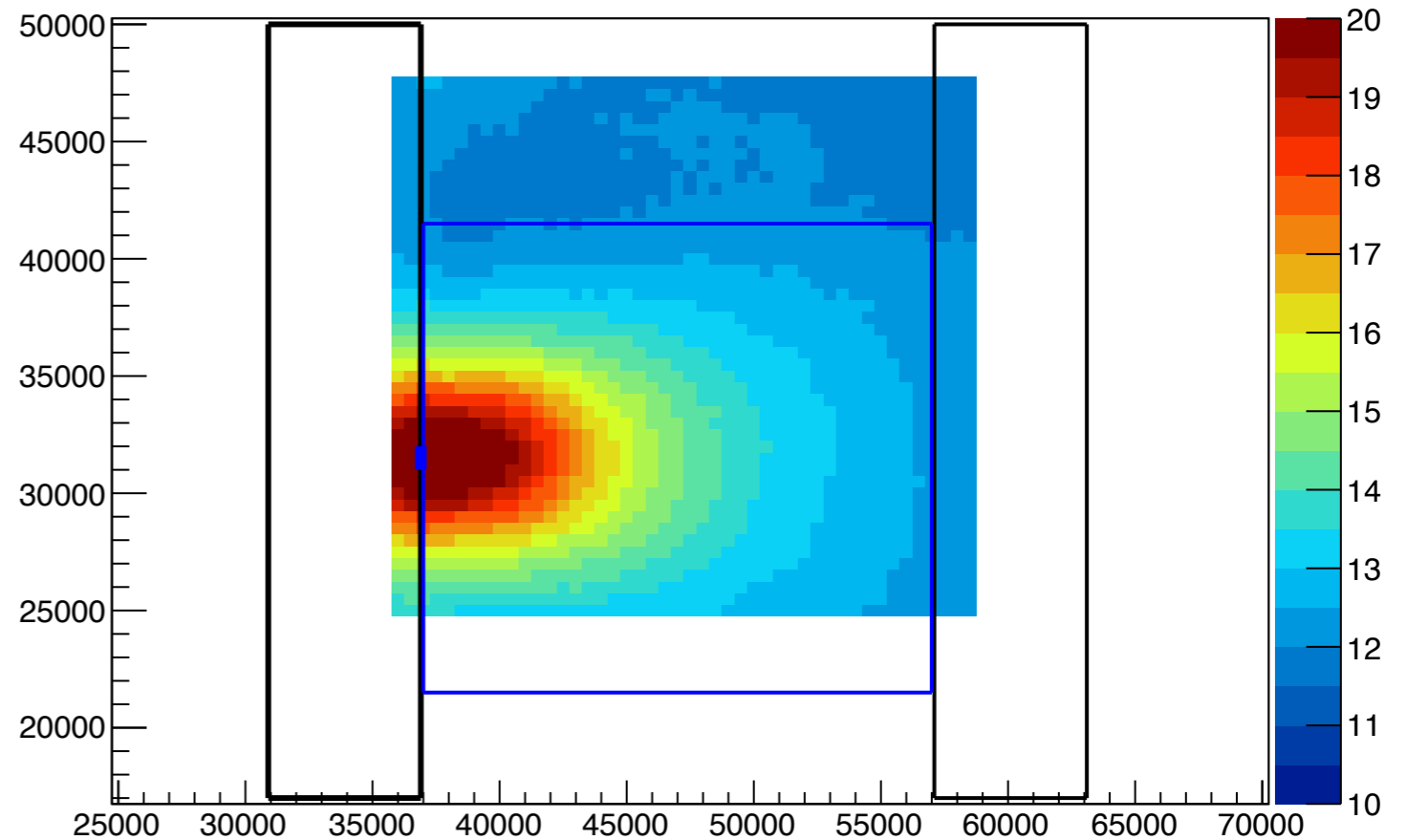
- it works, similar to baseline subtraction from waveform

Fine scan, first results with self-trigger



Sr90 source collimation
from Fluka

Average ch 1



- data acquired in 5 seconds, about 50k events per point
- distribution averages are plotted, calibrated but no fit