RUHR
UNIVERSITÄT
BOCHUM

Topix 2.0 measurement results

Darmstadt David-Leon Pohl Mo 14:00, 29.11.2010

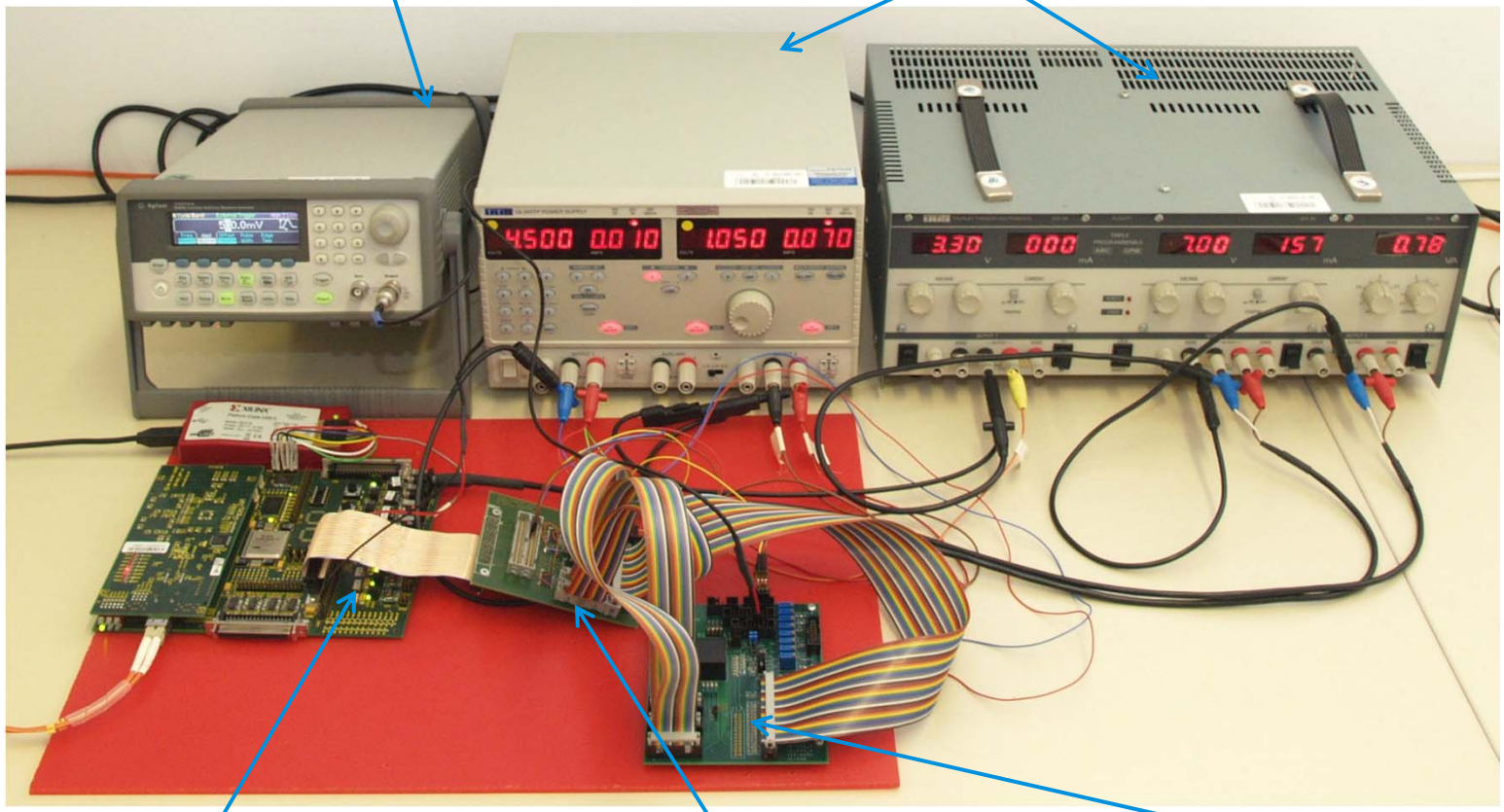
Content

- Experimental setup
- Threshold scan and threshold tuning
  - Method: threshold scan and lookup tables
  - Results
- Pile up effect
  - Observation
  - Method: double pulse
  - Results
- Time over threshold linearity
  - Method: direct injection
  - Results
- Summary

Experimental setup: Overview

signal generator
(Agilent 33250A)

two power supplies
(TTi Ql355TP, TTi PL330TP)

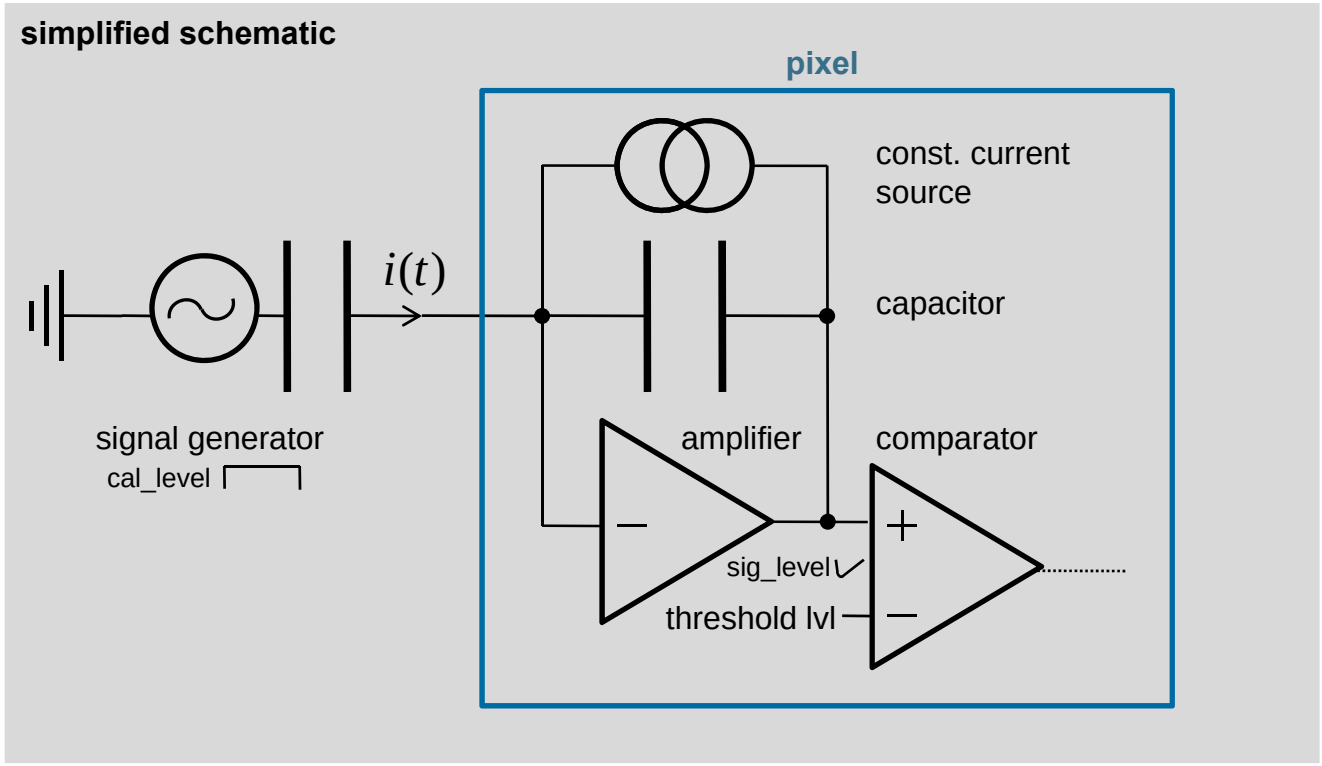


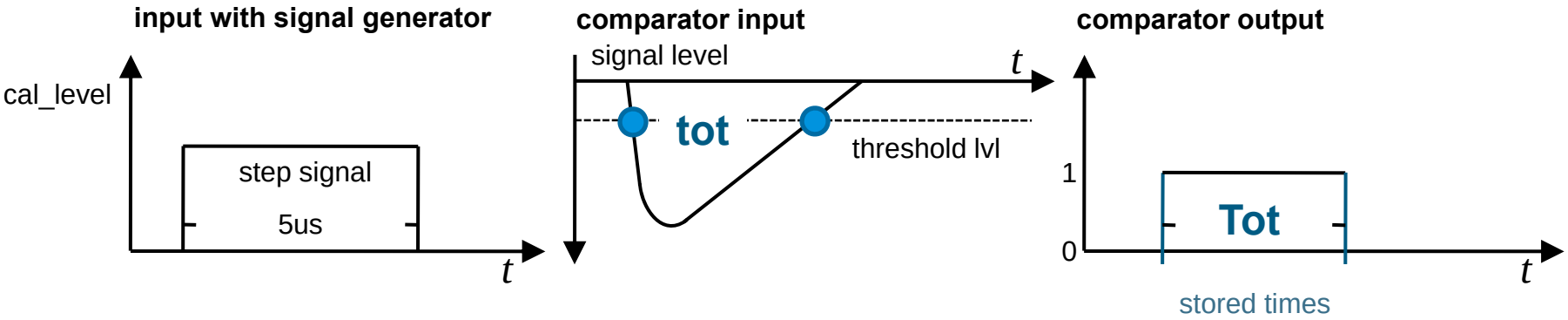
digital readout board
(with FPGA Virtex 4)

adapter board
(with three bus transceiver)

ToPix 2.0 test board
(with 12-bit DAC LTC2620)

Experimental setup: Time over threshold

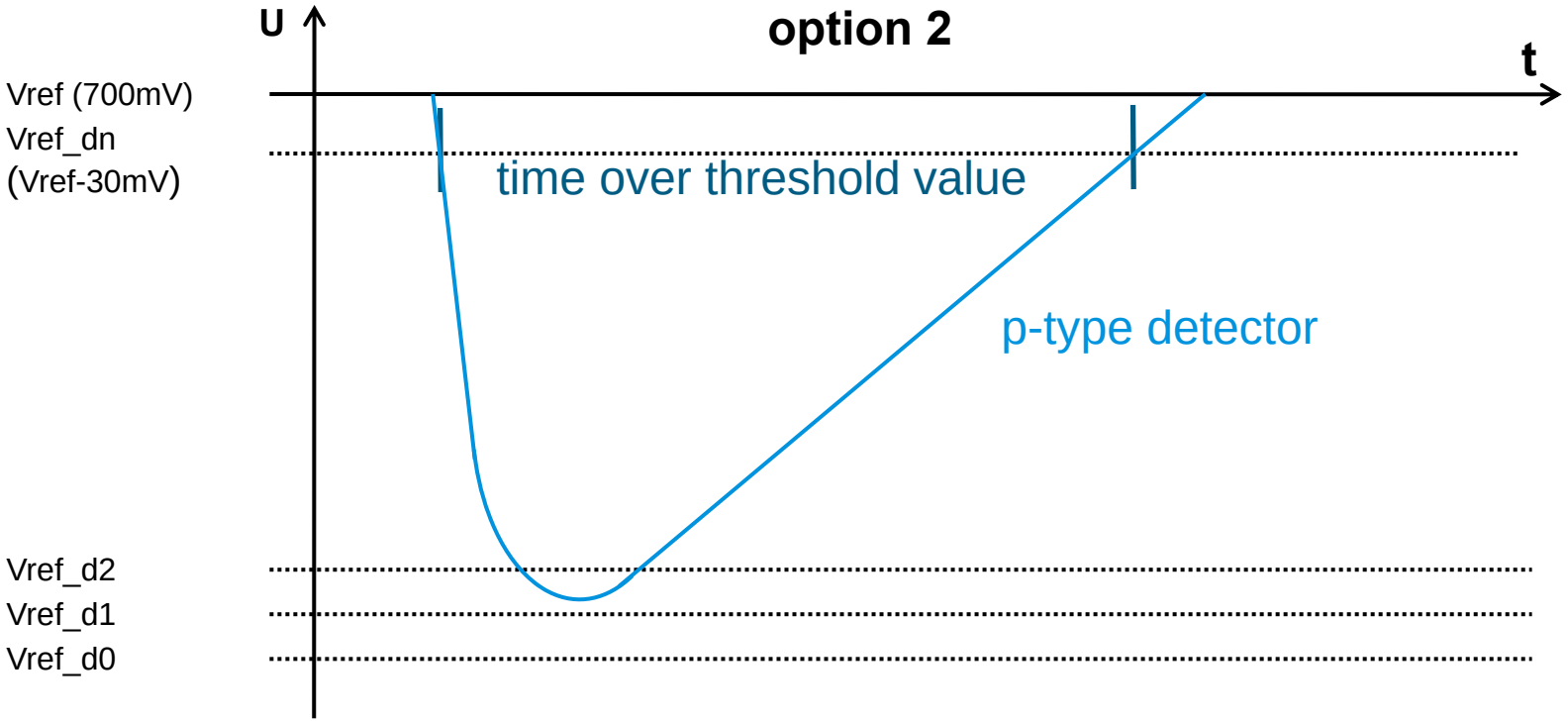




Threshold scan: Method

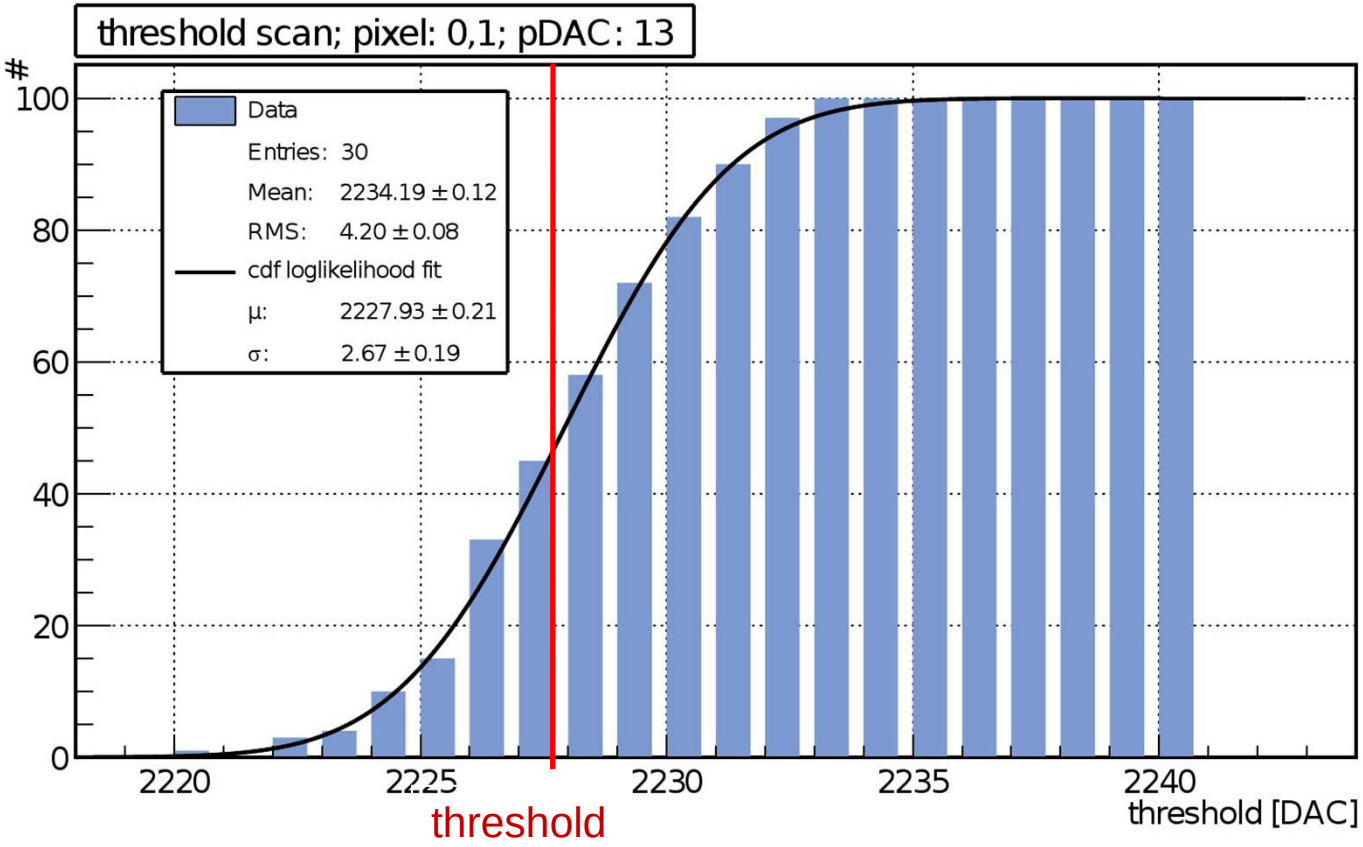
three options:

1. keep threshold constant, change signal level
   here **not possible** due to different non linearity between cal_level <-> charge injected for the pixels
2. keep injected charge constant, change threshold level
   here: threshold can be changed by a 12-bit DAC ($\Delta V = 0.3mV$)
3. there are 4 wire bonded pixels where method 1. is possible



here: measure $\frac{\text{pixel answers}}{\text{number of injection}}$ for different threshold levels (option 2)

Threshold scan: Threshold, noise determination

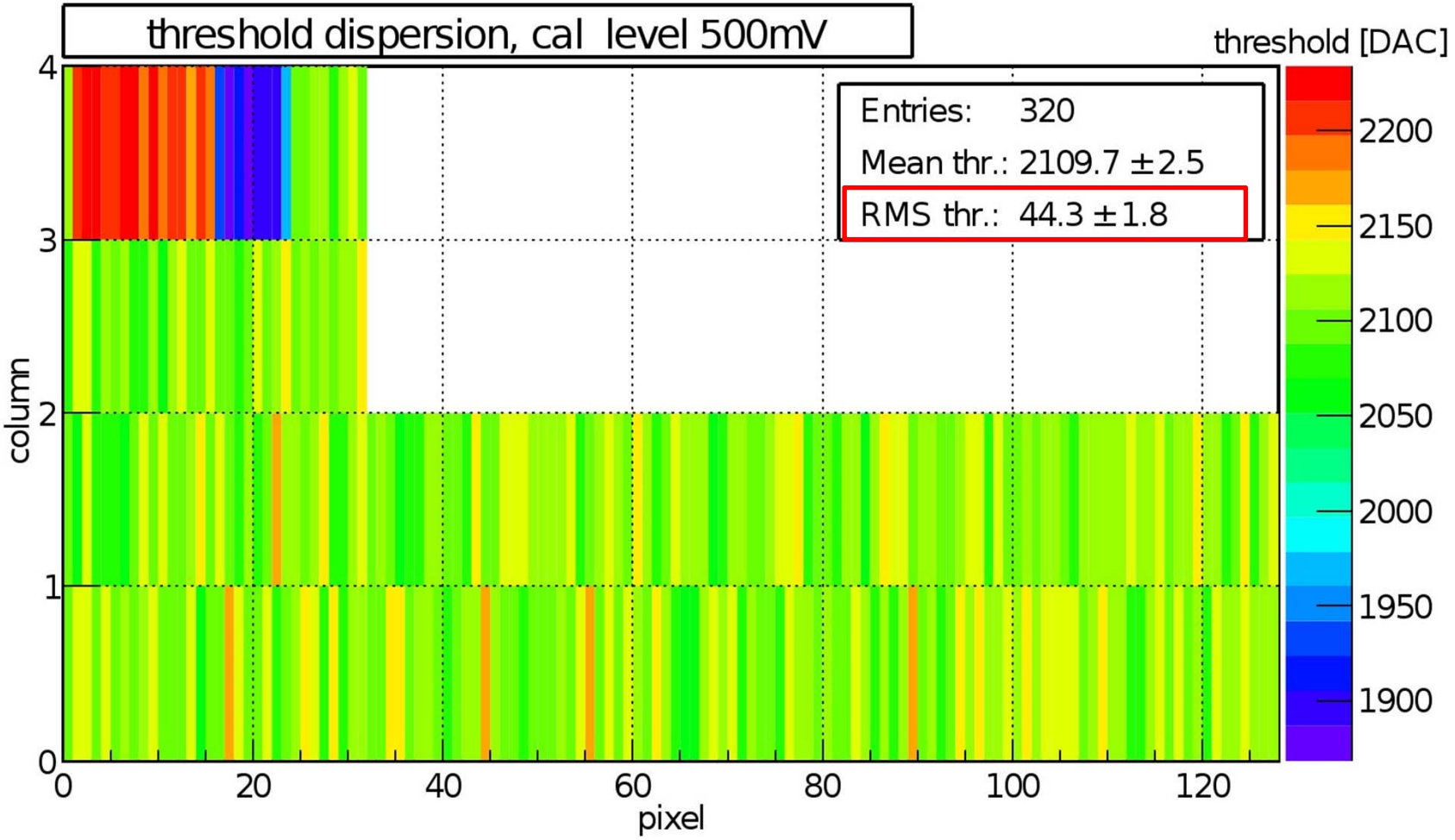


Fit: cumulative distribution function (cdf) of Gaussian distribution:

$$cdf(x) = \frac{N_{inj}}{2}\left(1 + erf\left(\frac{x-\mu}{\sqrt{2\sigma^2}}\right)\right) \quad \rightarrow \quad \begin{matrix}\mu : \text{threshold} \\ \sigma : \text{noise}\end{matrix}$$

Error threshold: 0.2 DAC for 100 injections per bin or more

Threshold scan: Result for all untuned pixels



goal: tune these 320 pixels to one threshold level

motivation: pixels with smaller threshold → higher resolution
not too close to zero level → avoid fake hits due to noise

Threshold tuning: Method

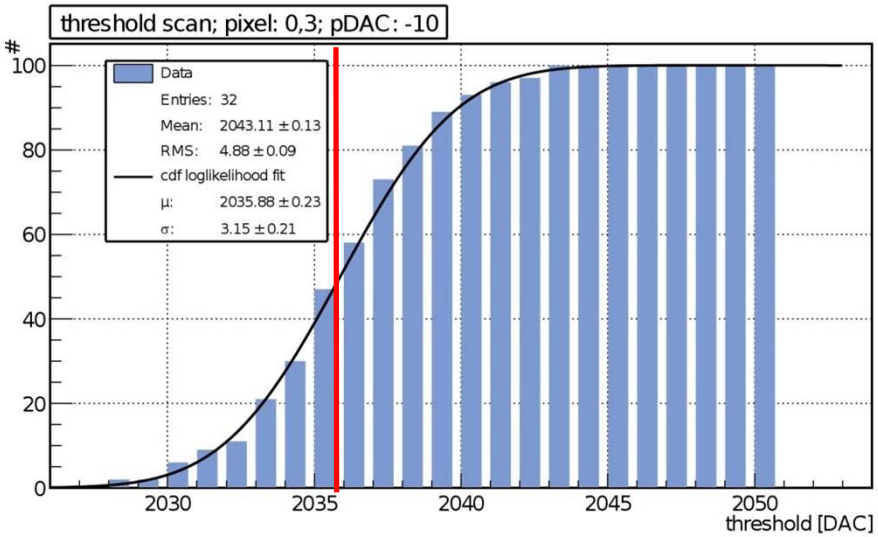
- every pixel has a 5-bit DAC to change its threshold level

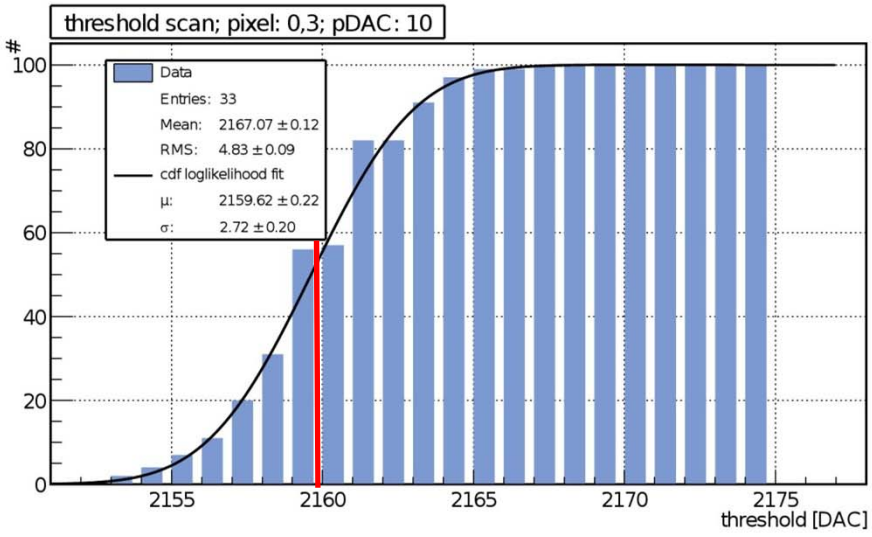
1	0	0	1	0

1 change direction

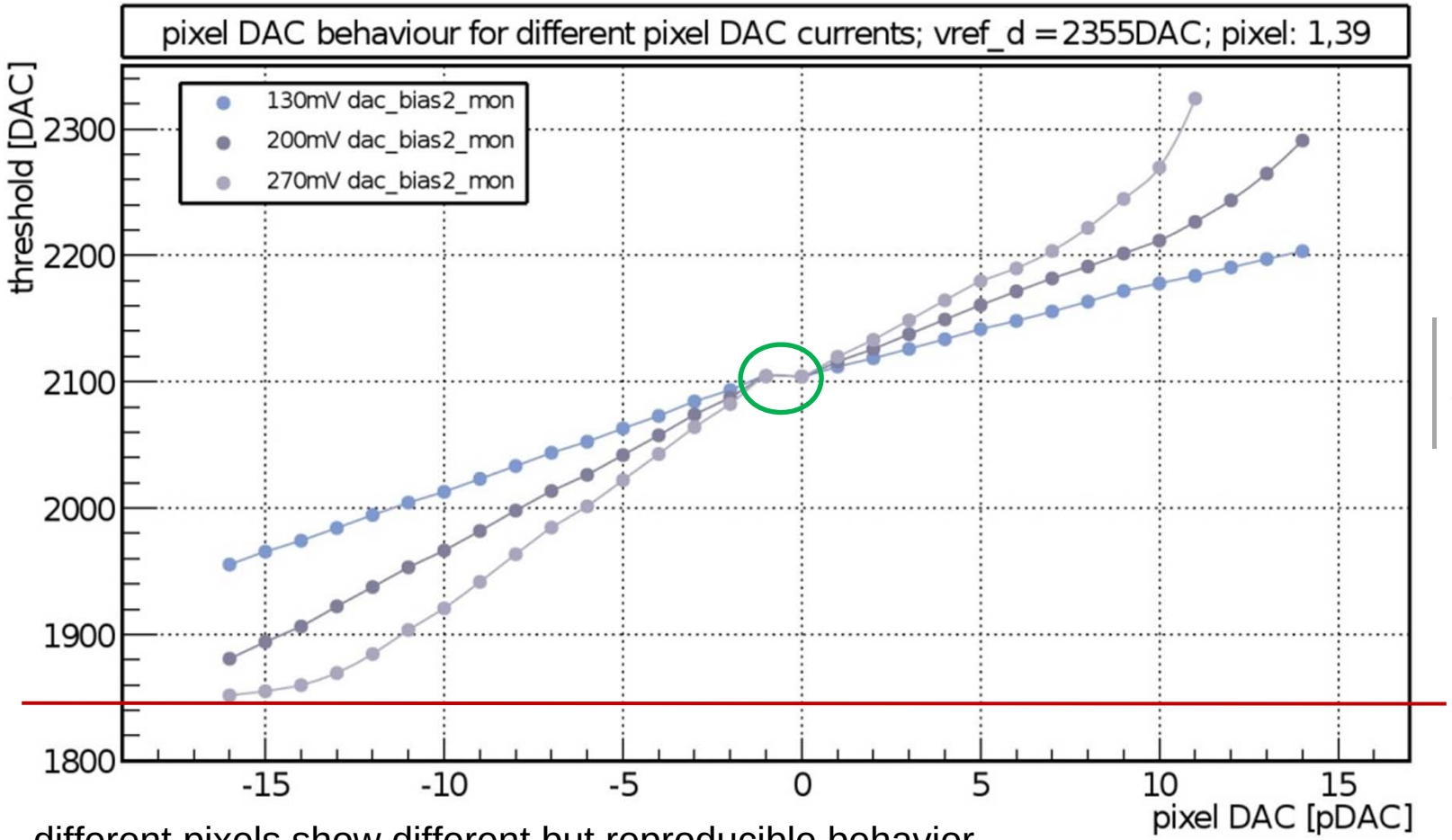
1 change amount

- here:
  - -16..-2 lower threshold
  - -1..0 direction bit change
  - 1..15 higher threshold
- determine the threshold for every pixel and pixel DAC setting (>9000 scans)
- example: left pixel DAC = -10, right pixel DAC = 10





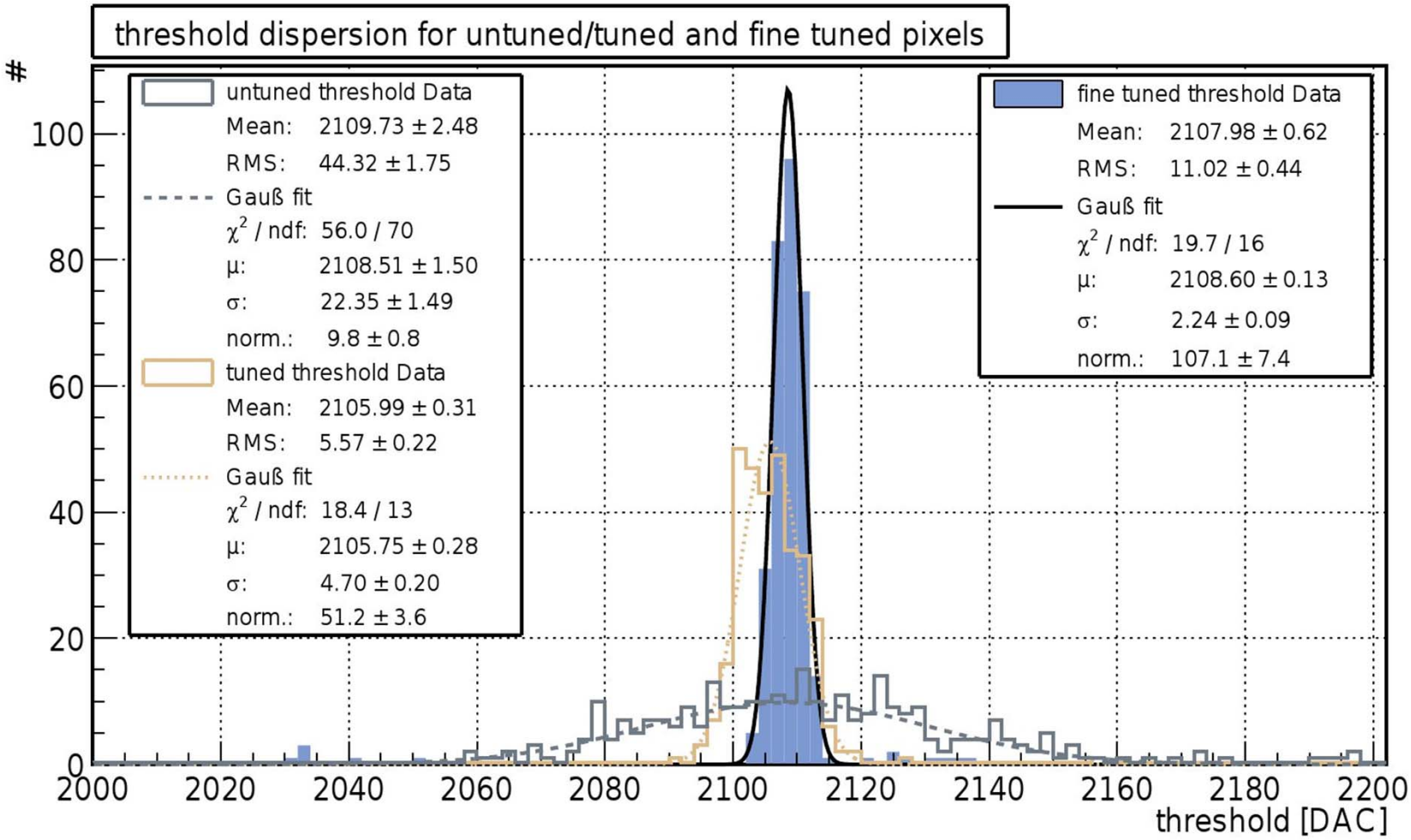
Threshold tuning: Lookup tables



RMS thr. dis.

- different pixels show different but reproducible behavior
- pixel threshold added/subtracted bit change
- saturation
- nominal range is to high (RMS threshold dispersion: 45DAC=13.5mV)
- best possible choice with the ToPix 2.0A test board, one could do even better! (<1DAC)

Threshold tuning: Result



result: - threshold dispersion σ = 2.24DAC = 0.67mV is smaller than the noise with 2.8DAC = 0.84mV

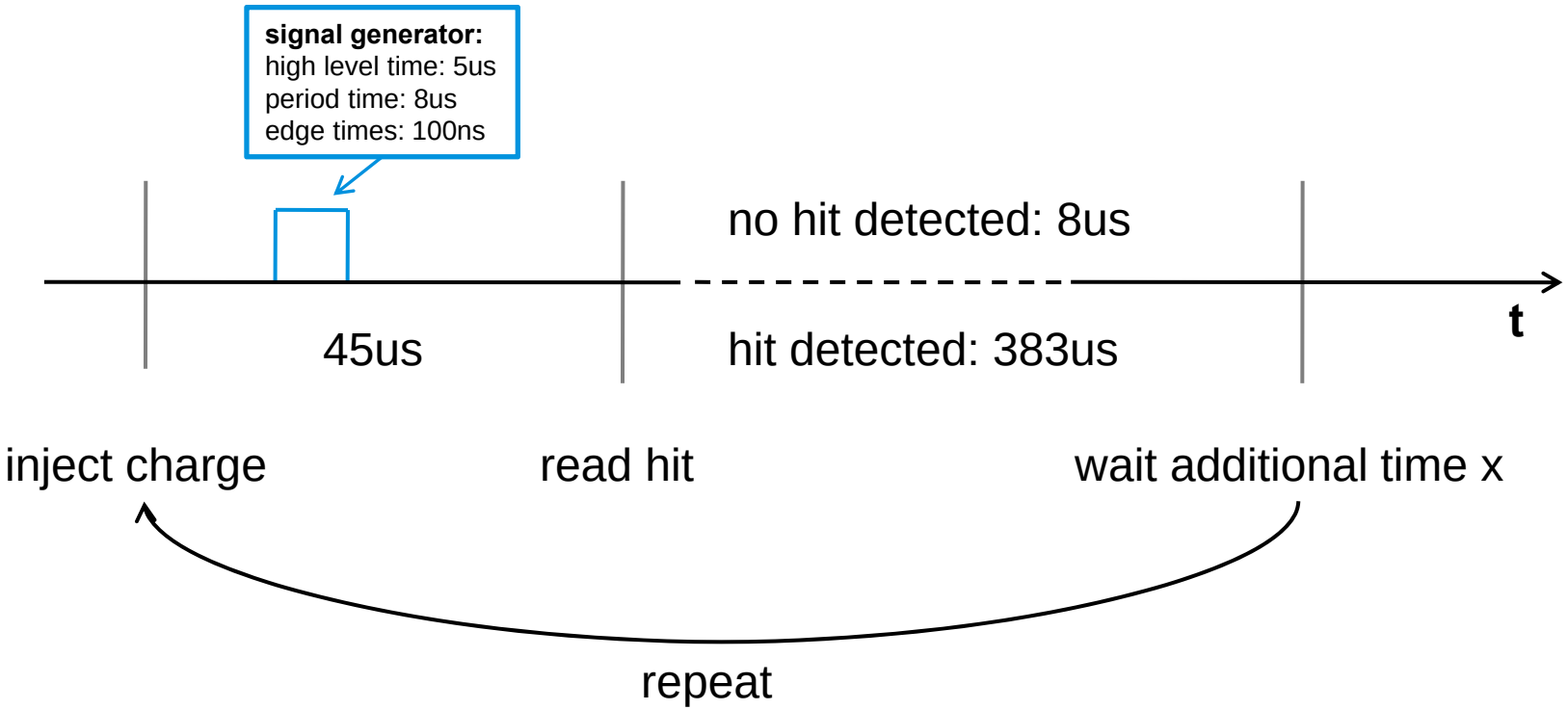
- a 5-bit pixel DAC provides sufficient range for tuning

Pile up: Timings for charge injection

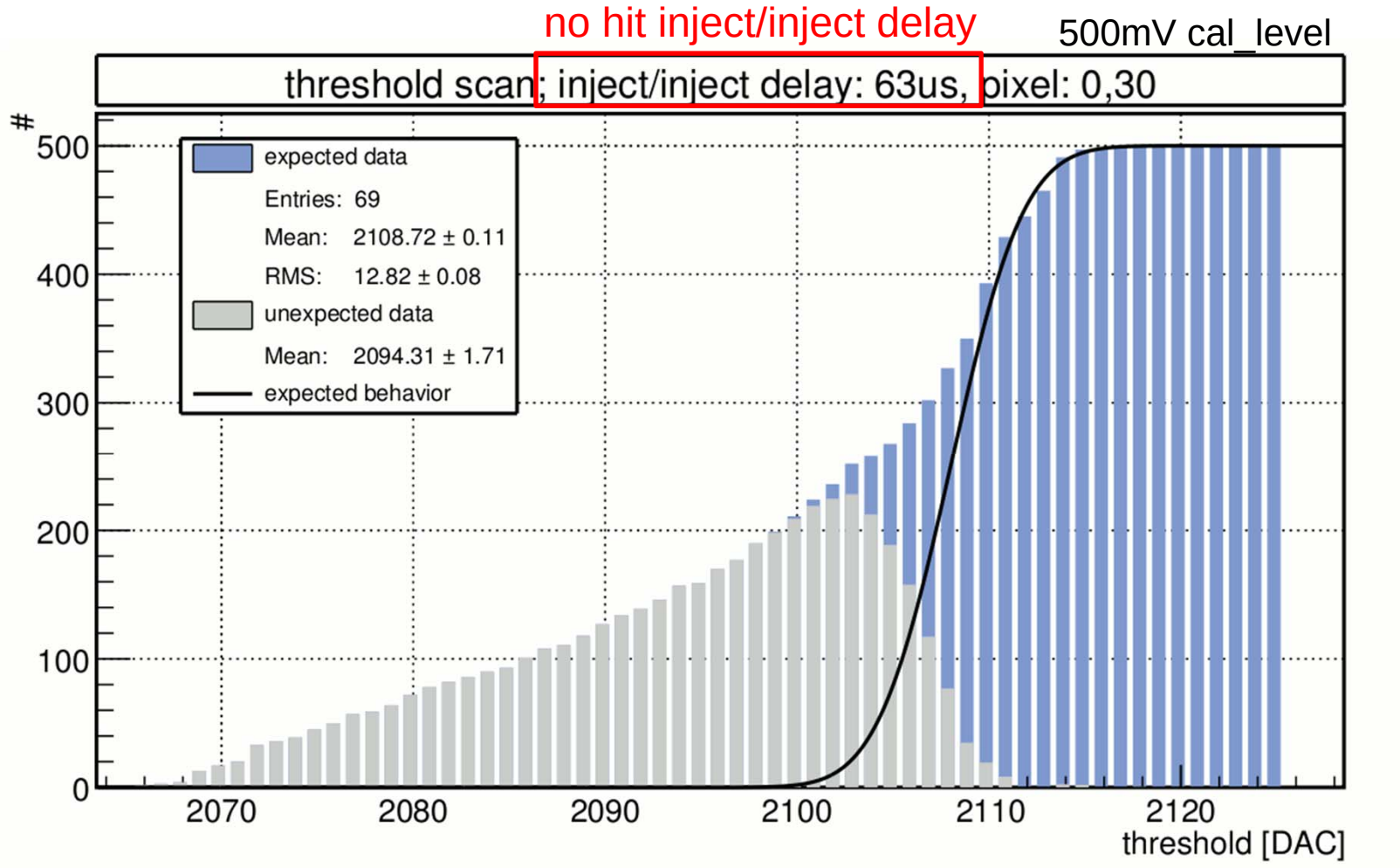
INJECT/READHIT LOOP{

inject charge (takes ca. 35us)
wait 10us
read hit (takes ca. 383us for a hit readout, ca. 8us for no hit)
wait additional time x for testing

}

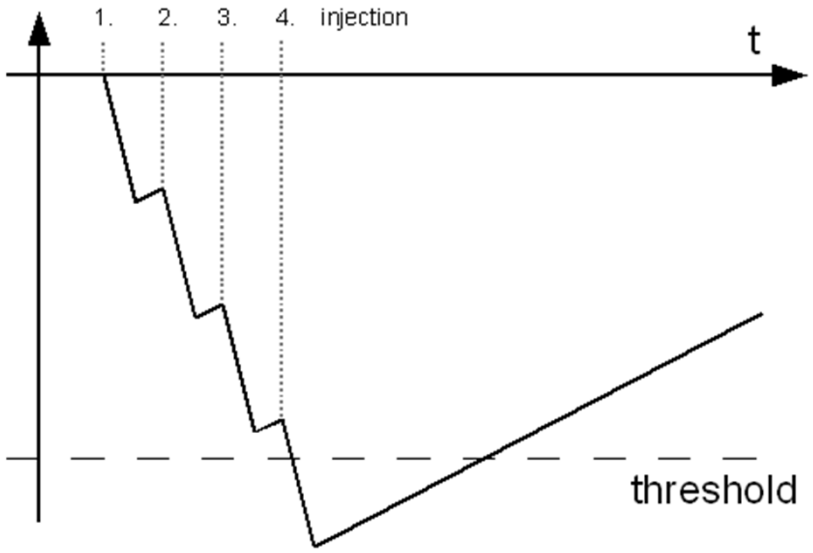


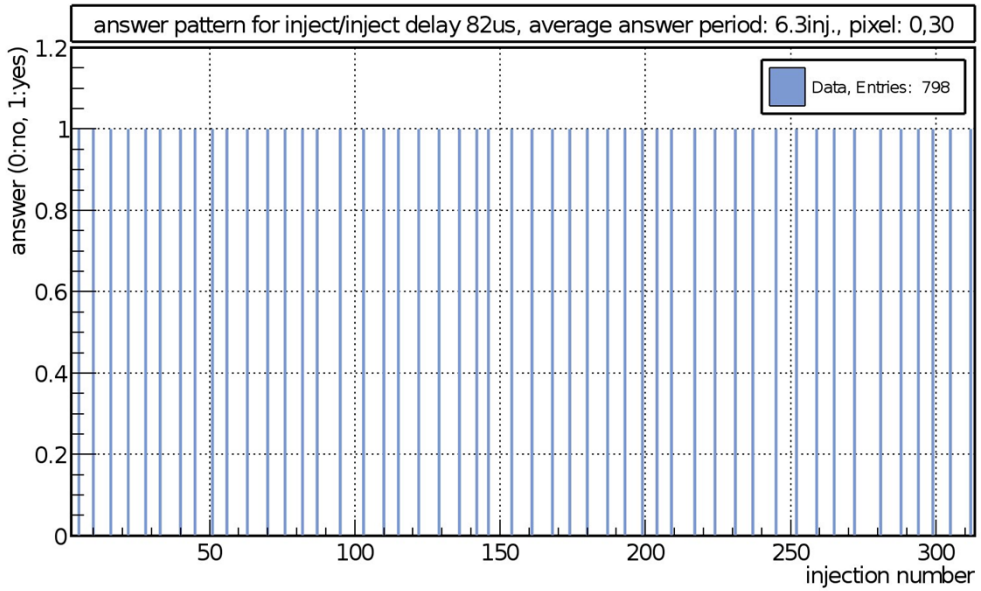
Pile up: First observation in threshold scan



- threshold scan depends on the delay time
- a hit read out takes additional time → pile up cannot be seen if a hit is detected

Pile up: Observation of a periodic pattern





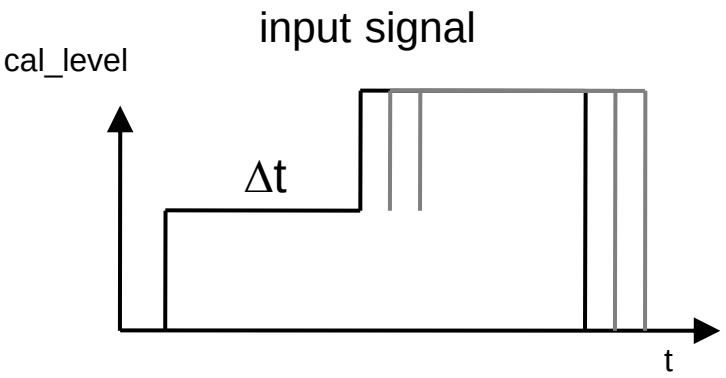
- periodic pattern: inject, inject, inject, hit detected, read out, delay due to readout, →baseline restored, inject, inject, inject…
- pile up in the order of several 100us till ms, tot values in the order of us
- → baseline restoring **under** threshold takes a long time

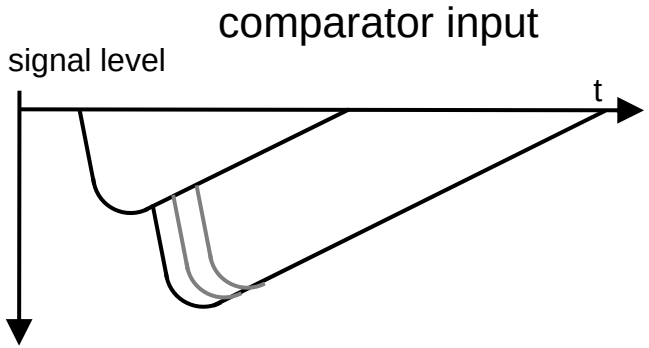
Pile up detection method: Double pulse

challenge: - no possibility to measure the analog signal with the ToPix 2.0 test board
- the only two measurable values are: TOT, pixel responses or not

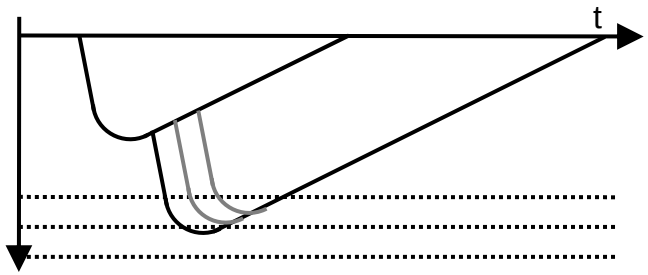
solution: **double pulse method:**
use double pulse with different delays combined with a threshold scan:

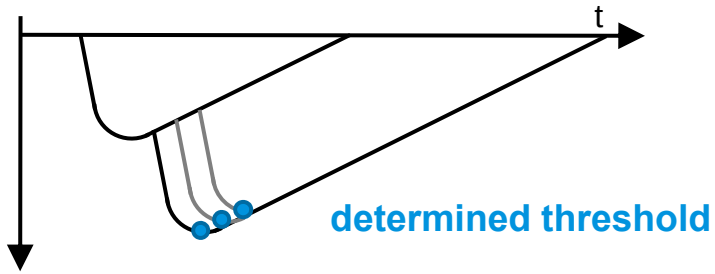
1. inject signal two times





2. do a threshold scan

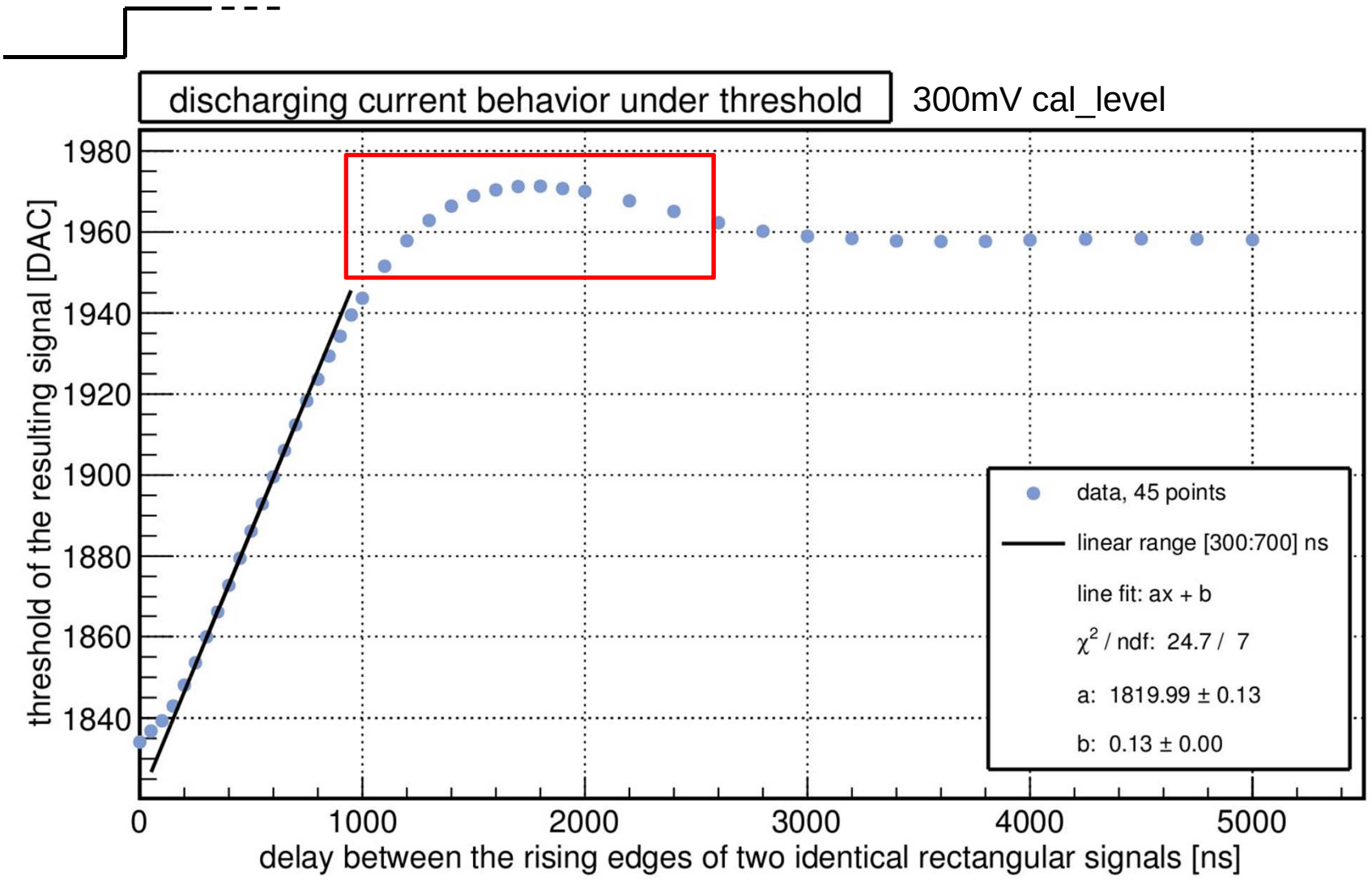




3. repeat for 1. and 2. for different rising edge delays Δt
4. threshold(Δt) shows the shape of the first signal

Pile up detection method: Validation

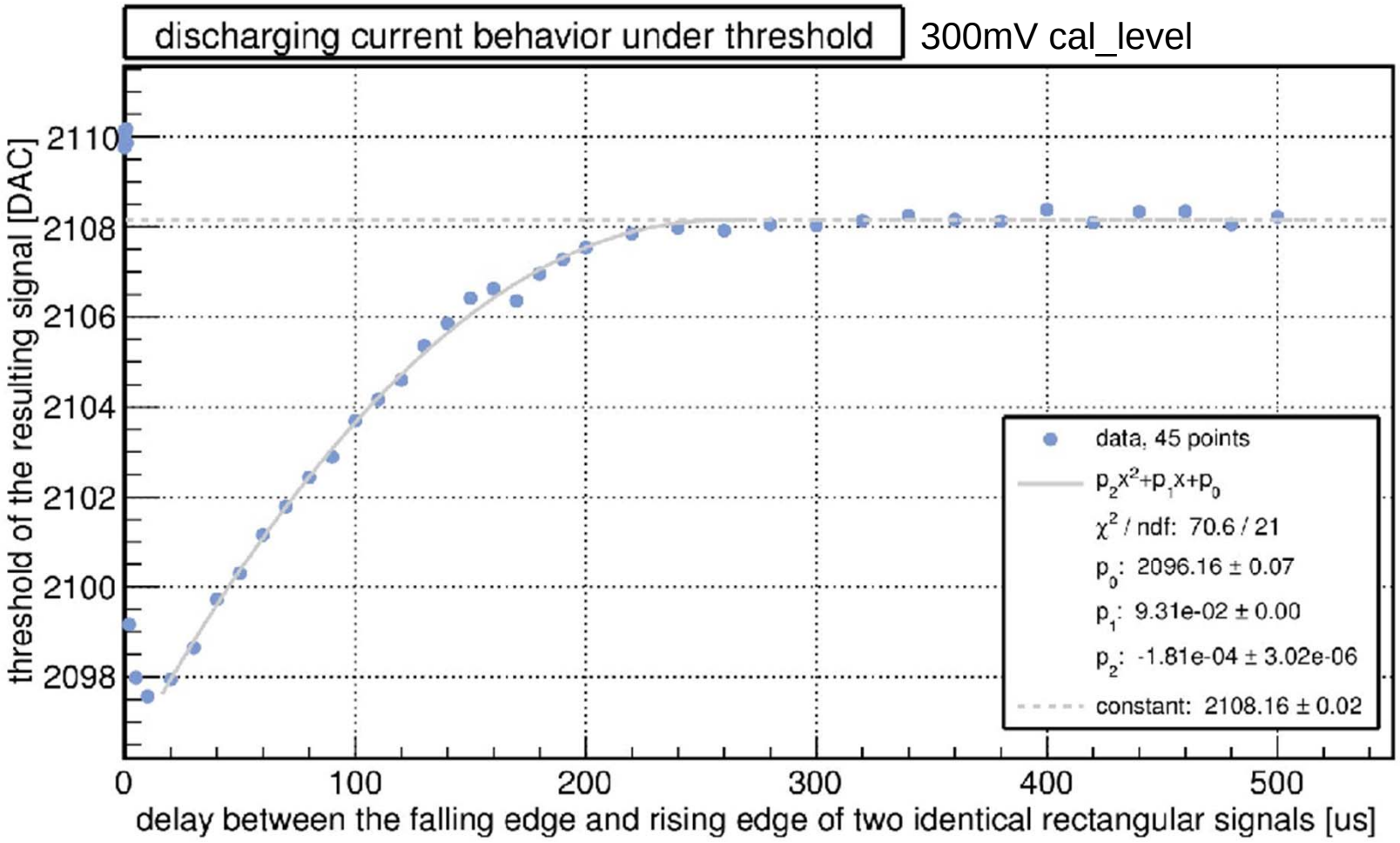
- signal after rising edge:



- one can see the constant discharging current (straight line)
- there is an overshoot

Pile up detection method: Baseline restoring

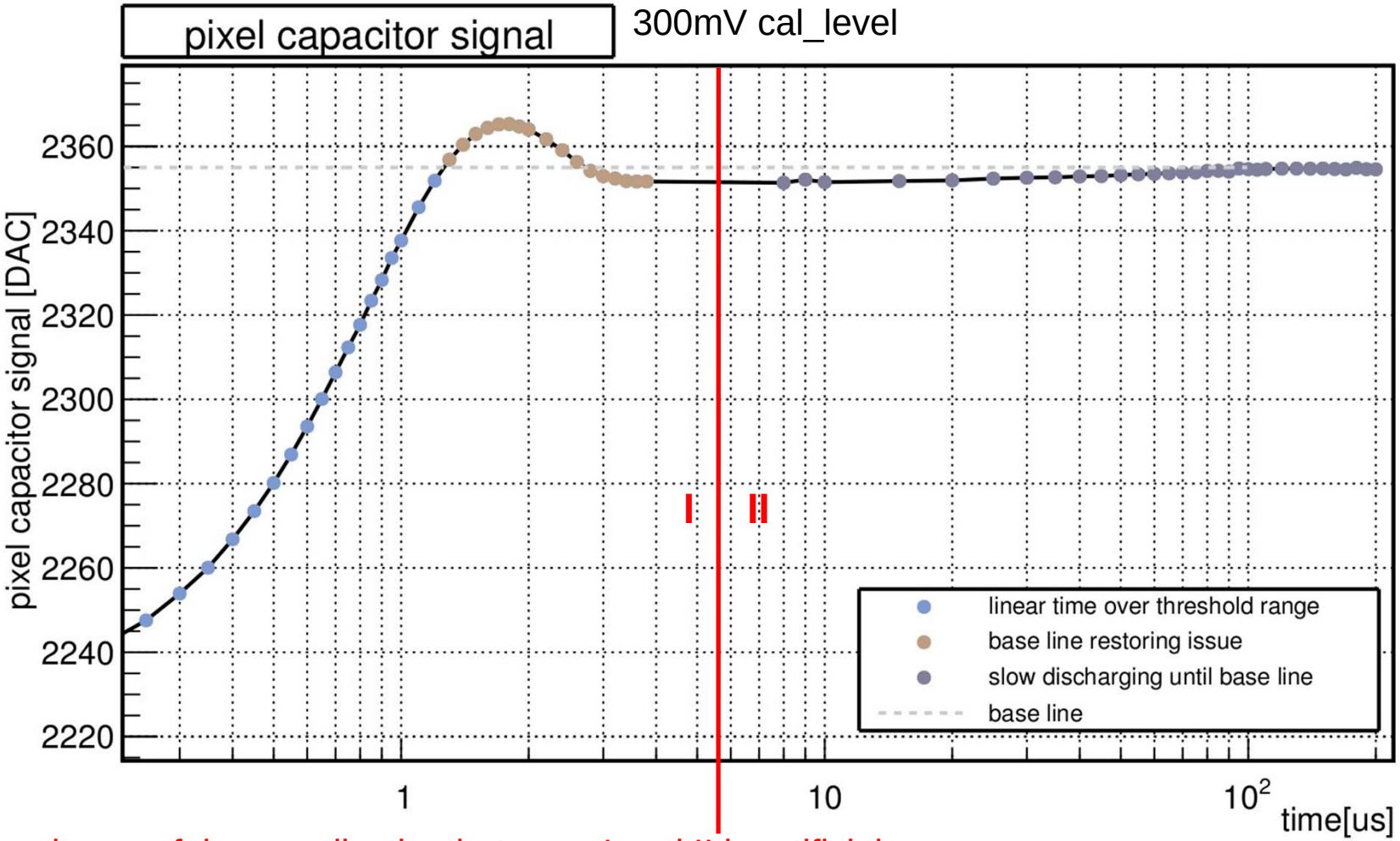
- signal after falling edge:



- impact for experiment: not clear, there is no falling edge

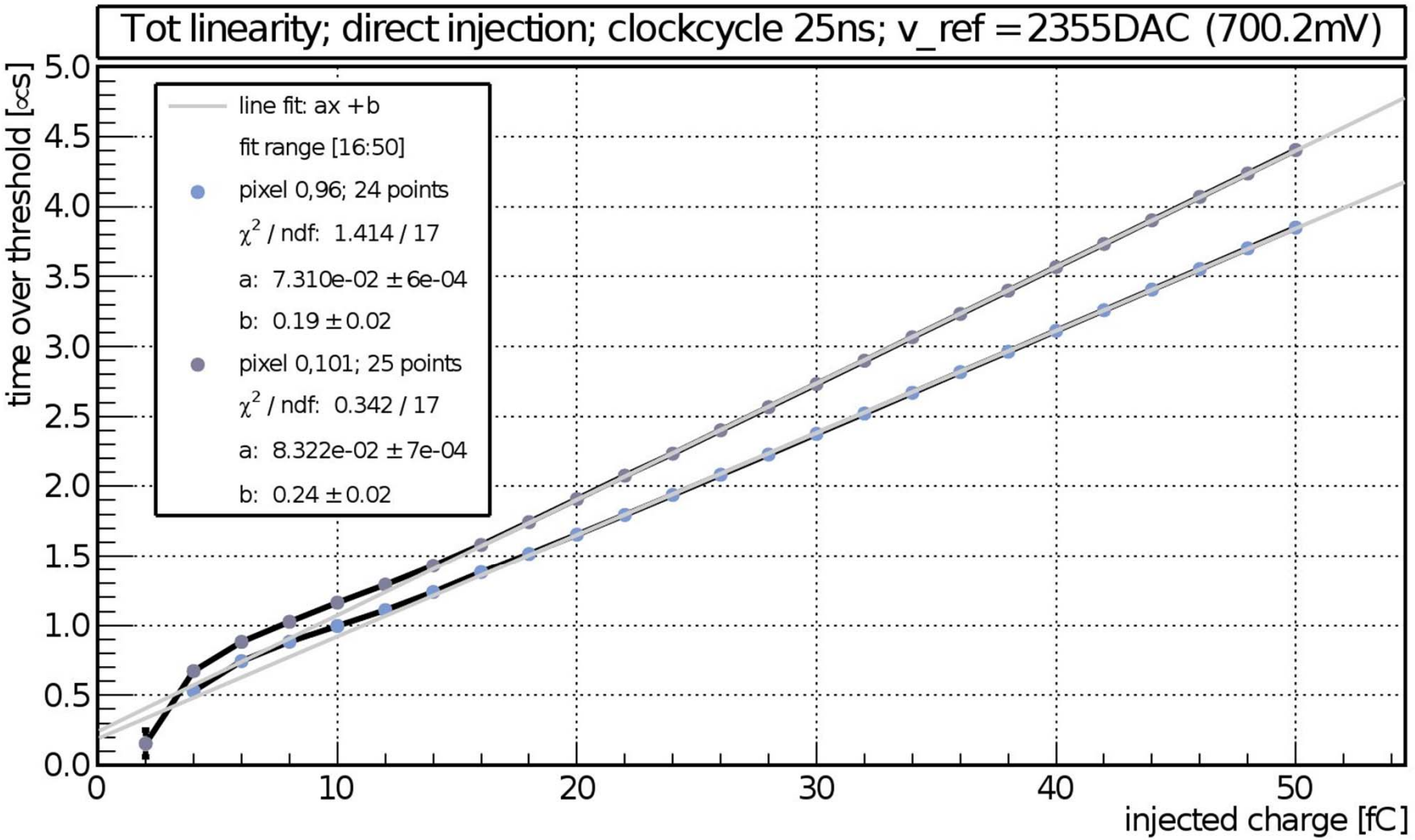
Pile up effect: Results

- time scale of the signal:



- be careful: normalization between I and II is artificial
- baseline level is probably not valid for I

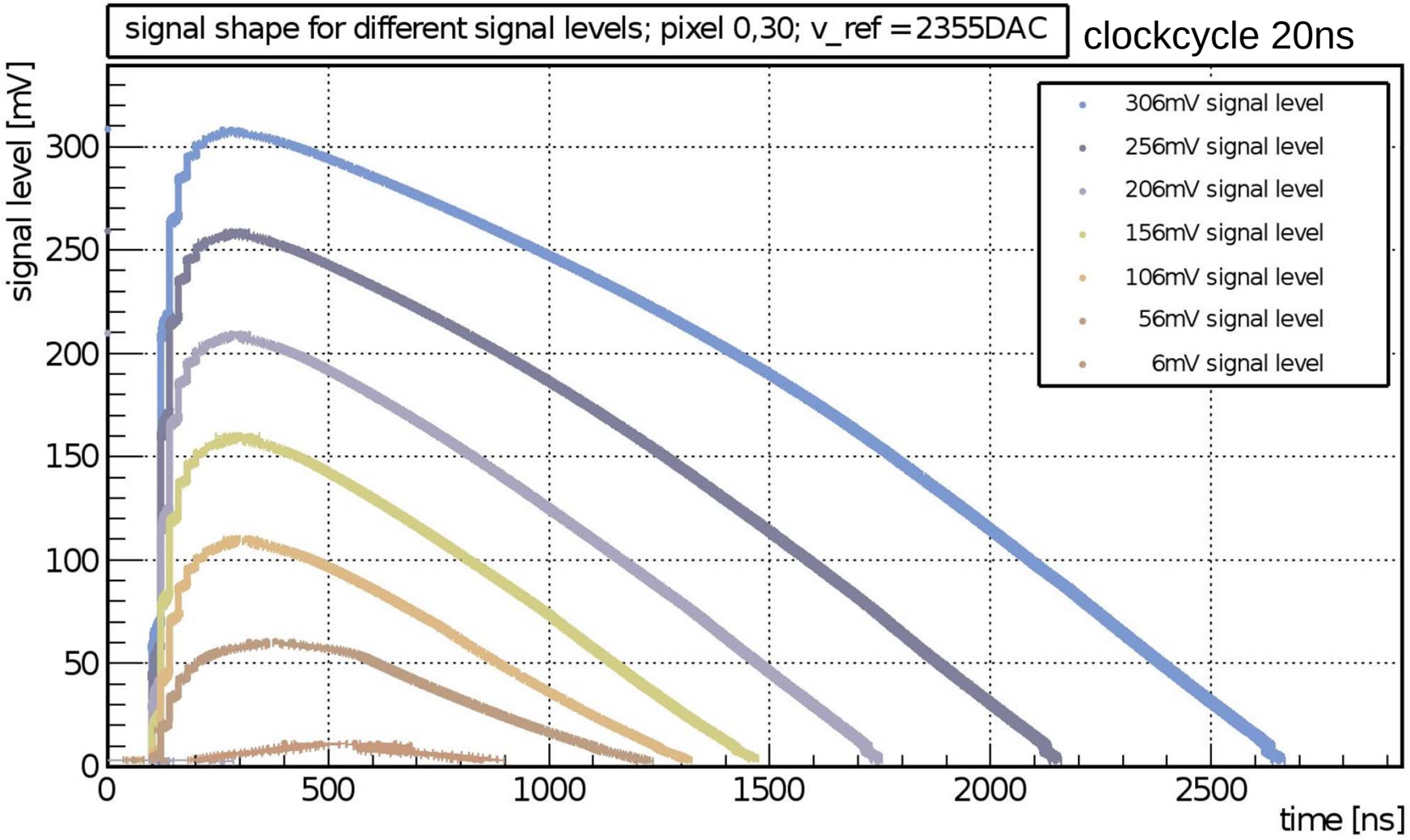
Tot linarity: Direct injection in wire bonded pixels



- charge injected: $Q_{\text{inj}} \approx C \cdot \Delta U \approx 1\text{pF} \cdot \Delta U$
- linearity for injected charge > 16fC = 100000 electrons

Tot linarity: Signal shape

- each point with error bar is a Gaußfit over 1000 injections

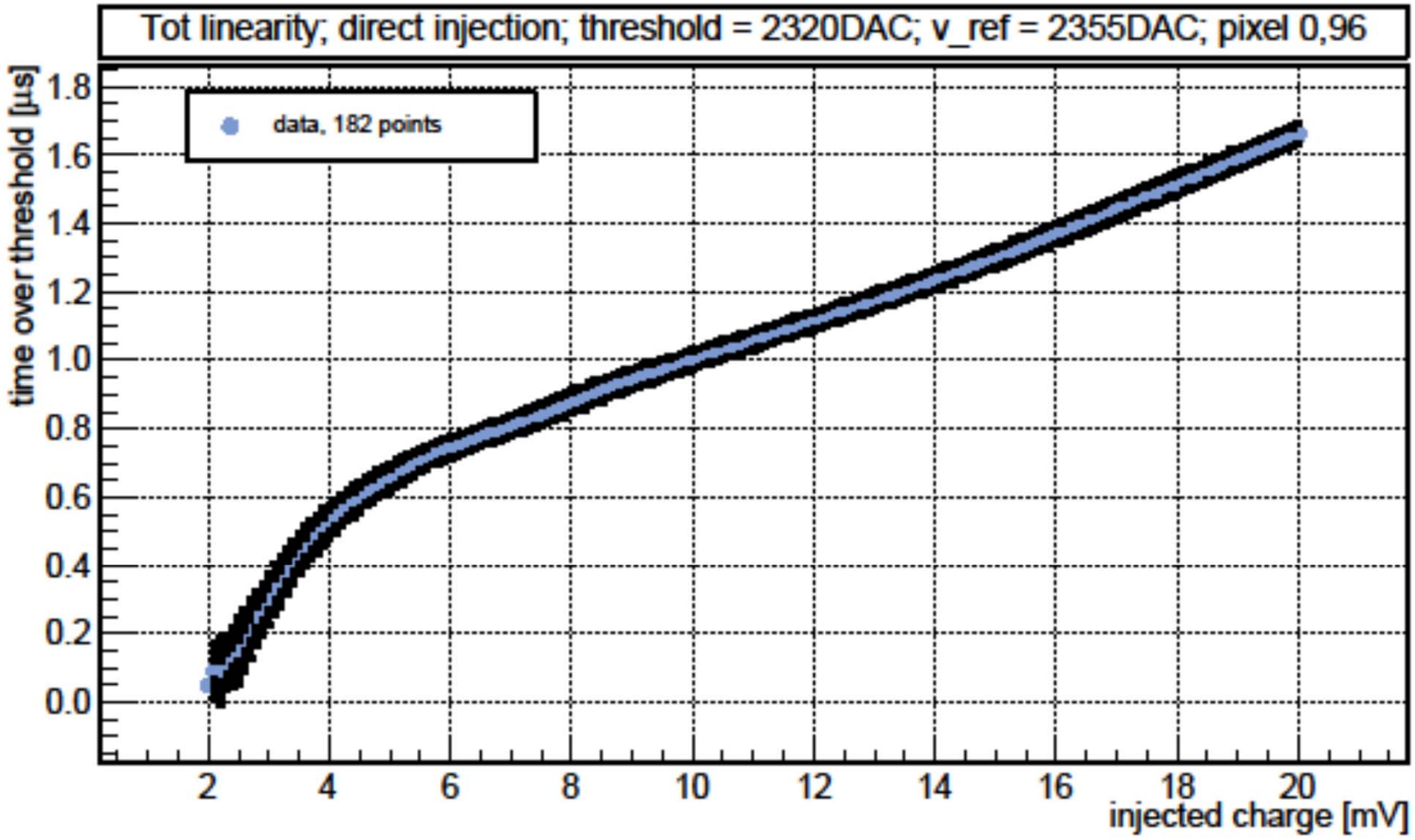


- **Reason for non linearity?**: signal shape for small charges injected looks different

Summary

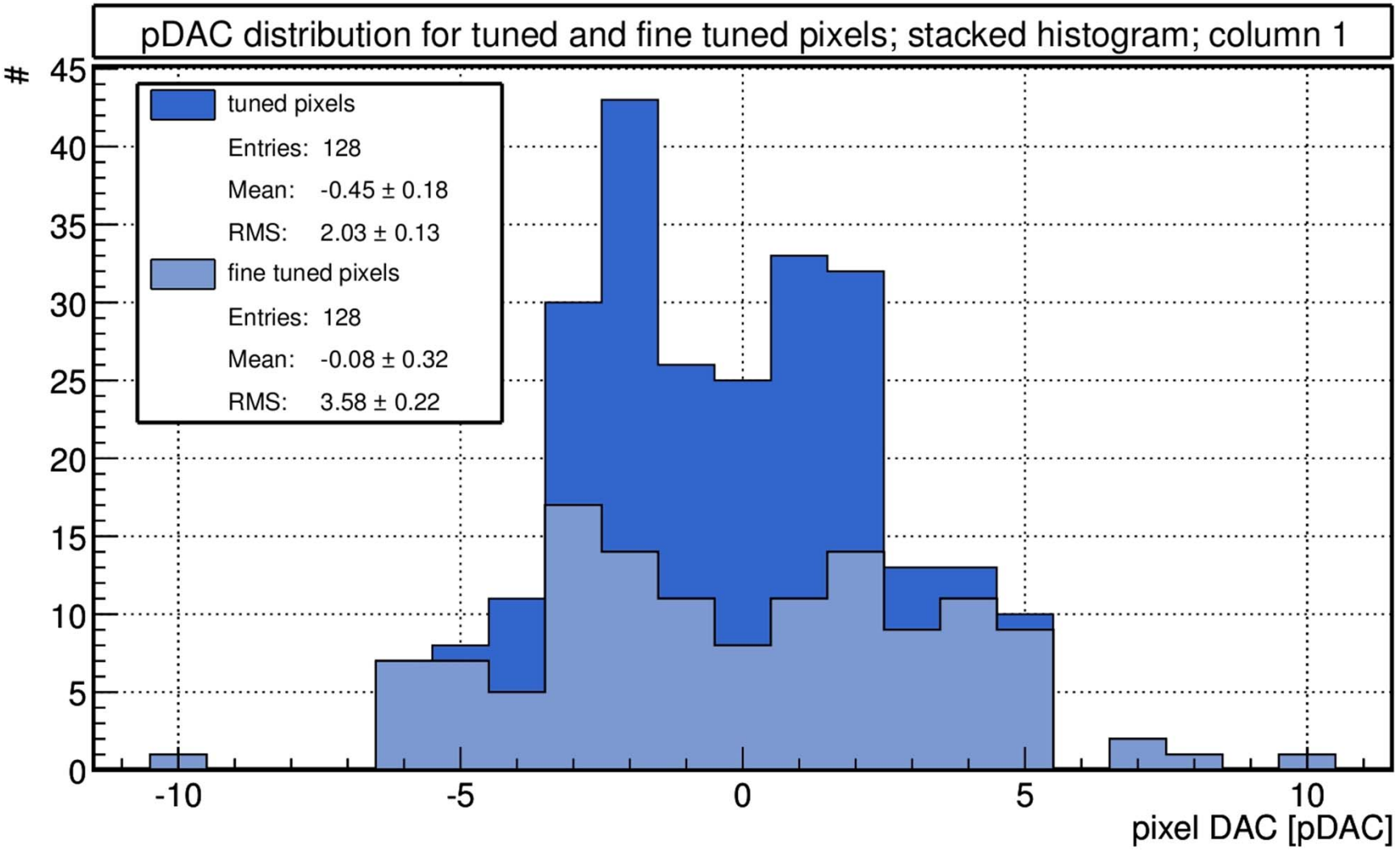
- Threshold tuning
  - threshold dispersion σ = 2.24DAC < noise = 2.8DAC
  - 5-bit pixel DAC is sufficient
  - still better tuning possible (< 1DAC)
- Pile up
  - double pulse method is an effective way to identify pile up
  - large dead time when testing: up to several ms
  - impact for usage with wire ponded pixels: unknown
- Time over threshold linearity
  - nice linearity for injected charges > 16fC
  - signal shape prevents linearity for smaller charge injected

Appendix: Time over threshold linearity



time over threshold linearity, zoom on non linear part

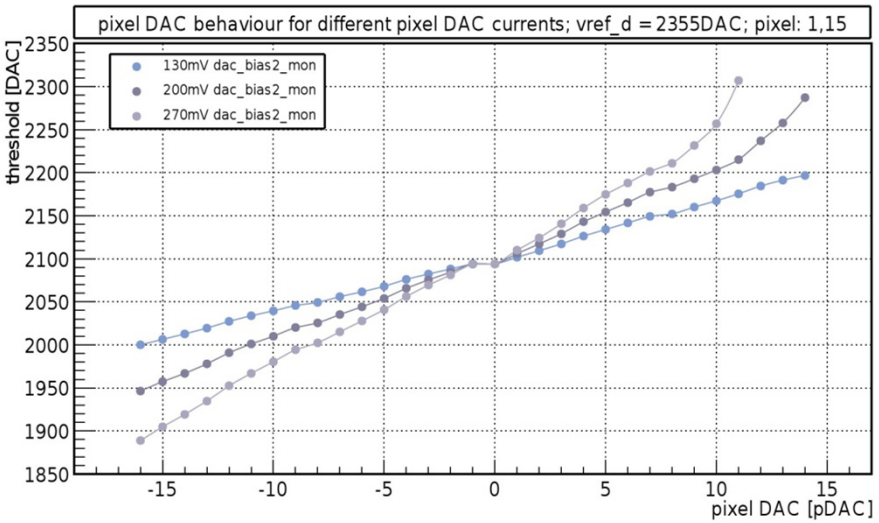
Appendix: Pixel DAC distribution

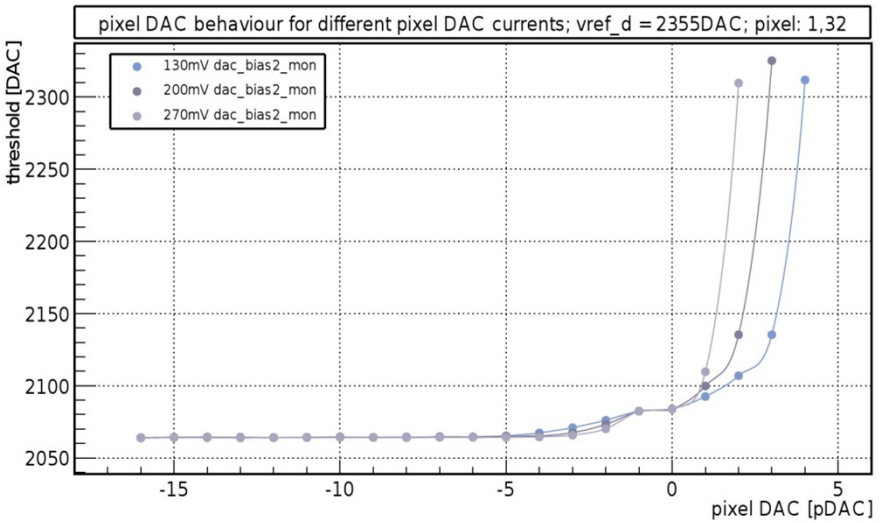


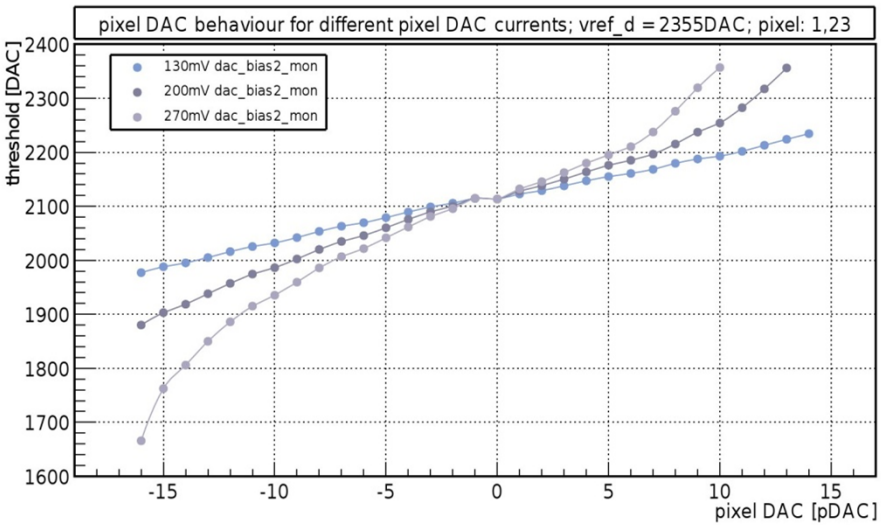
the given range is used much more efficient for fine tuned pixels

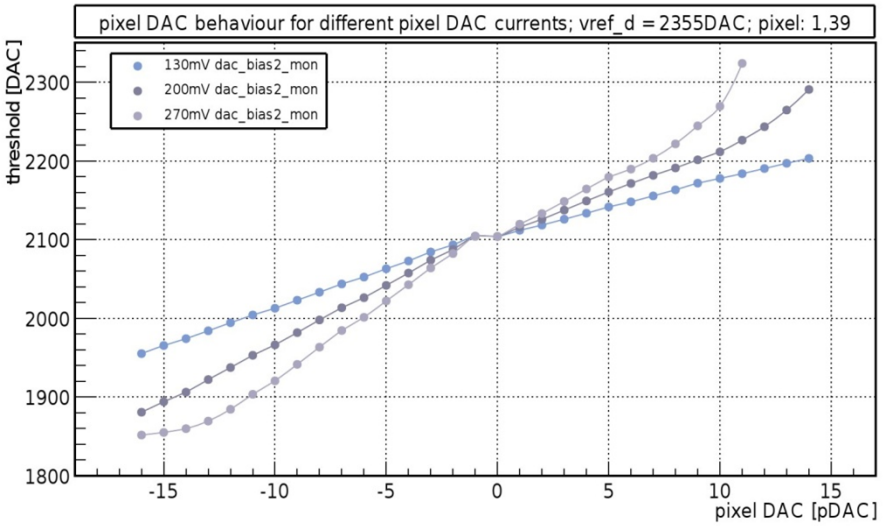
Appendix: Pixel DACs behavior

very small range provided for this pixel



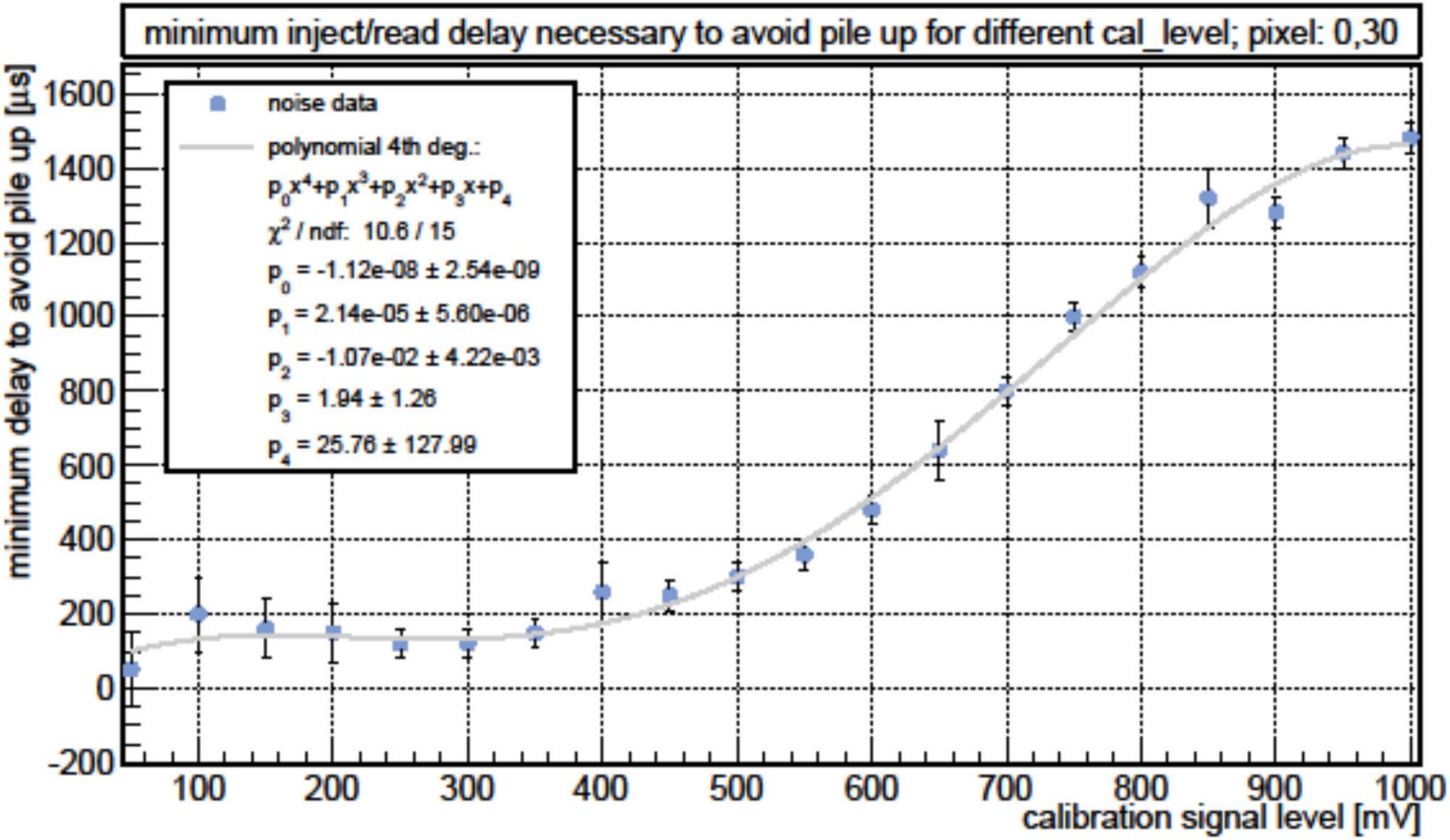






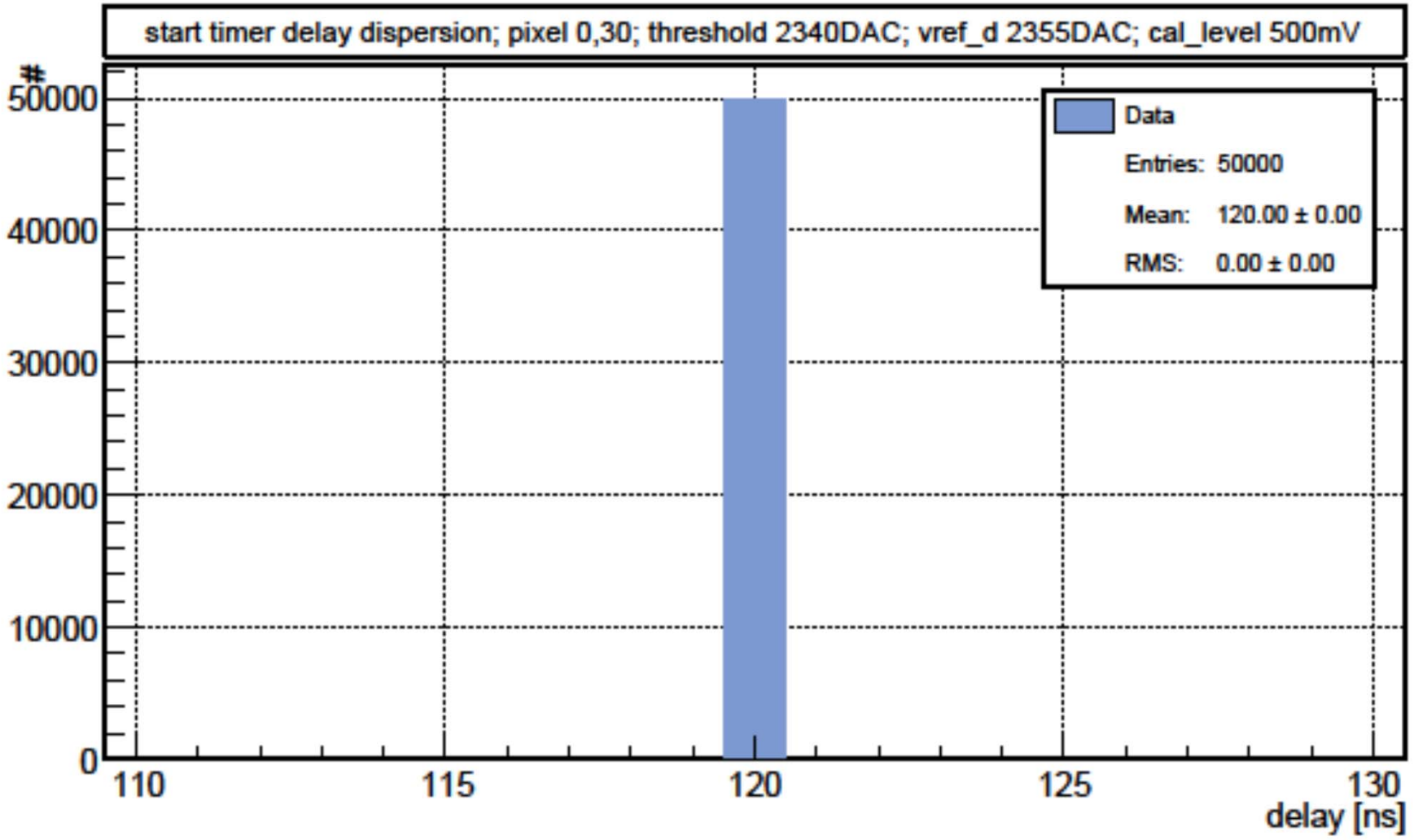
very different behavior for different pixels

Appendix: Delay times needed to avoid pile up

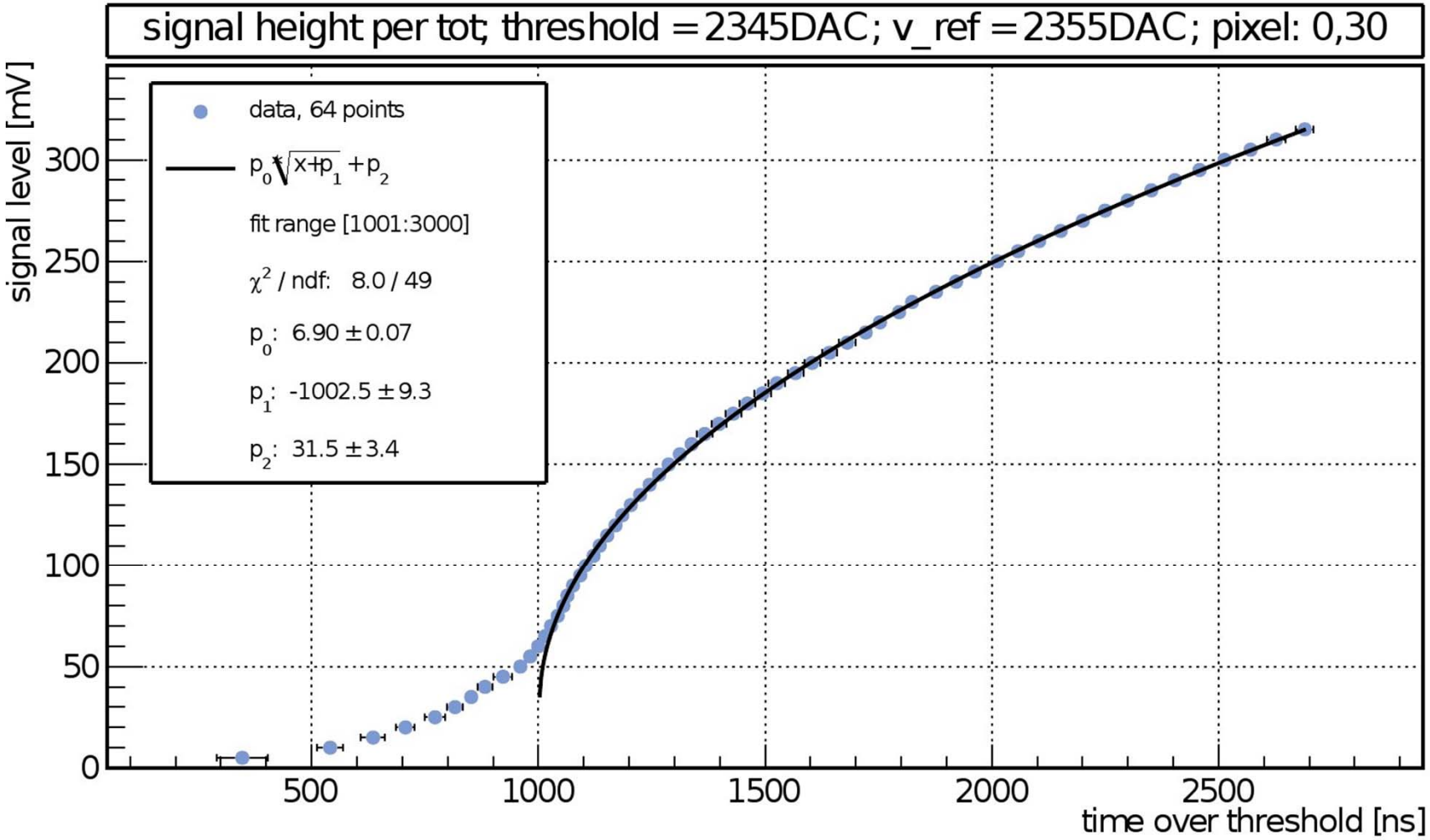


Appendix: Offset for rising edge

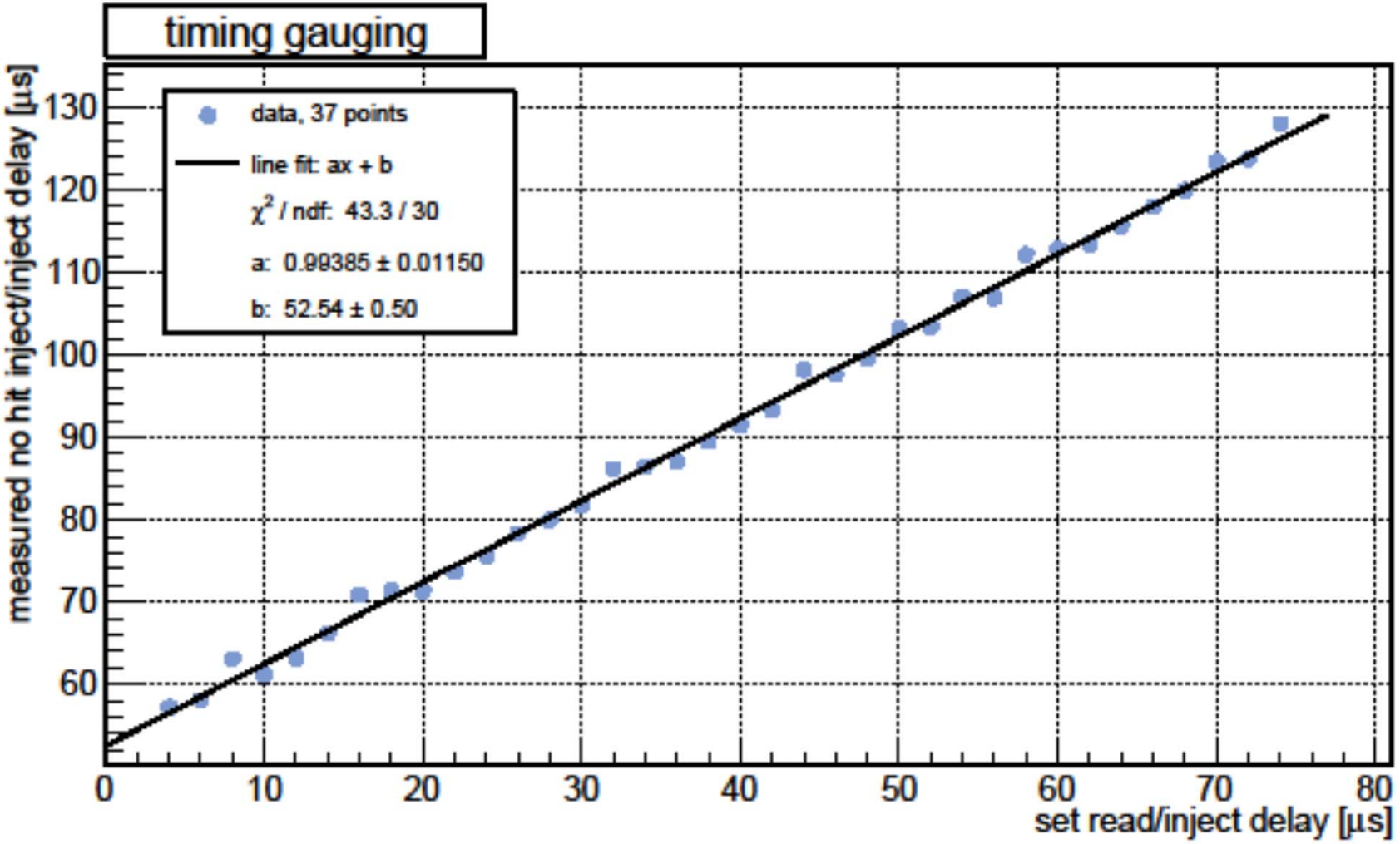
constant offset for rising time measurement



Appendix: Signal level for charge injected

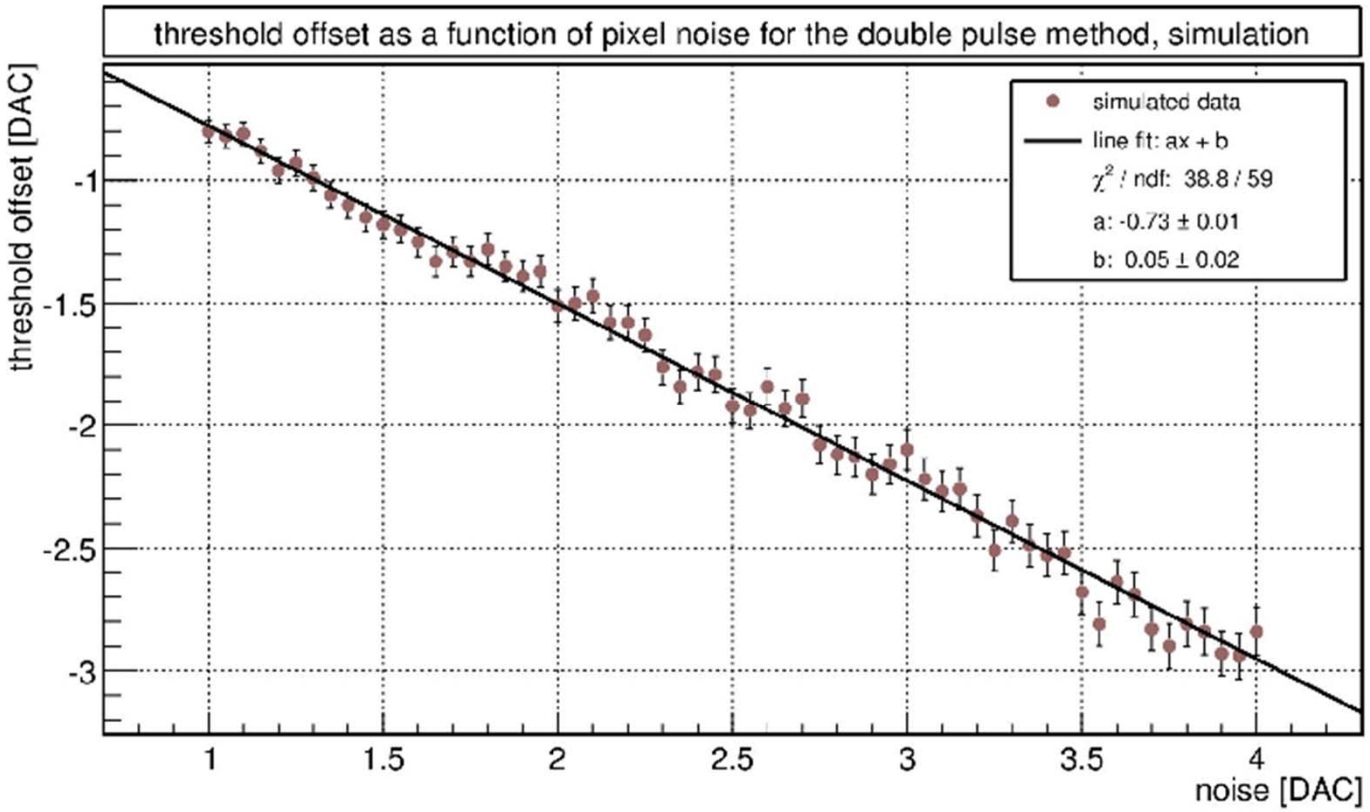


Appendix: Software trigger linearity

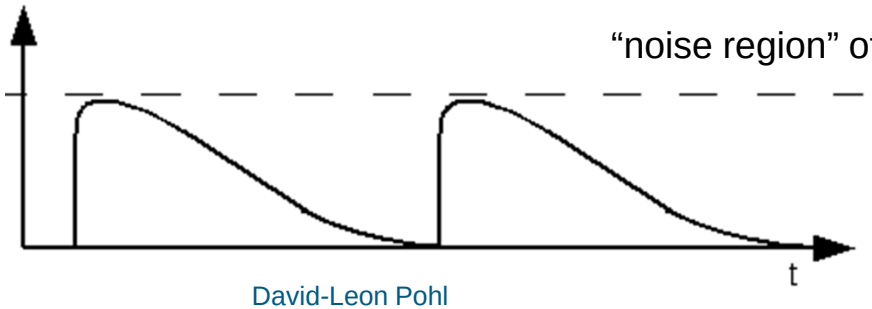


Appendix: Double pulse

double pulse generates additional noise source

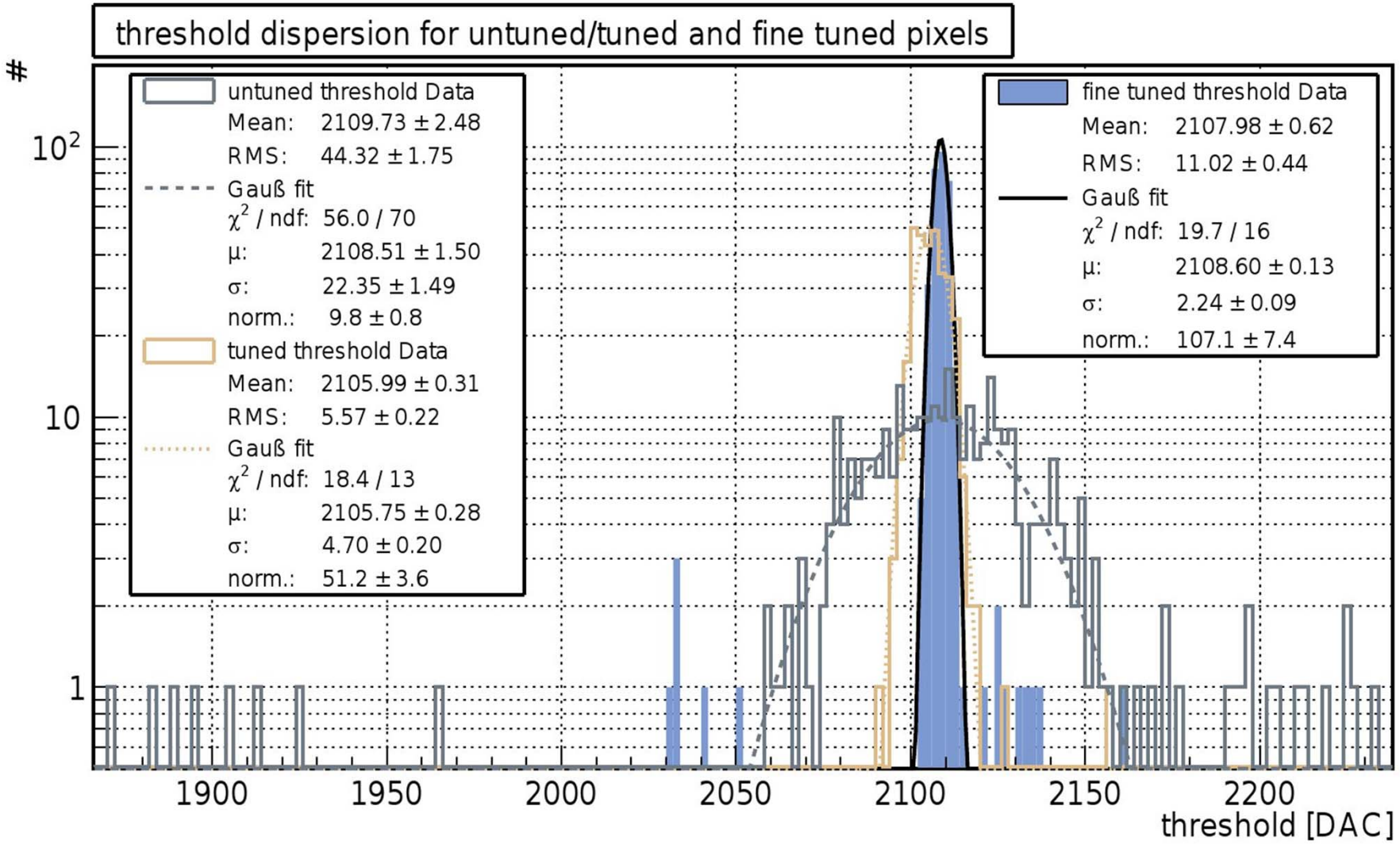


signal level



"noise region" of both signals

Appendix: Result of page 10 with all entrees



Appendix: Input signal (cal_level)