university of groningen

kvi - center for advanced
radiation technology

Readout of the PANDA Electromagnetic Calorimeter,

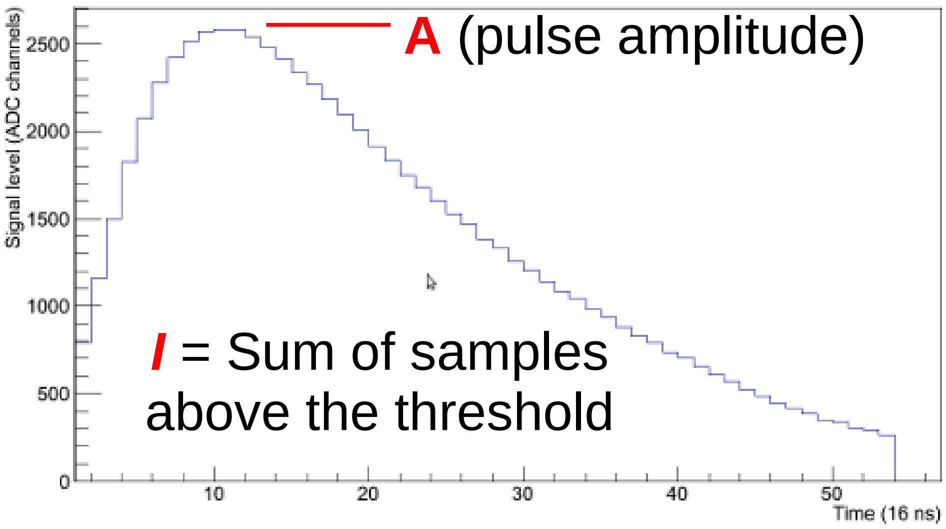
M. Kavatsyuk, O. Kuiken, P. Schakel

KVI-CART, University of Groningen

for the PANDA collaboration

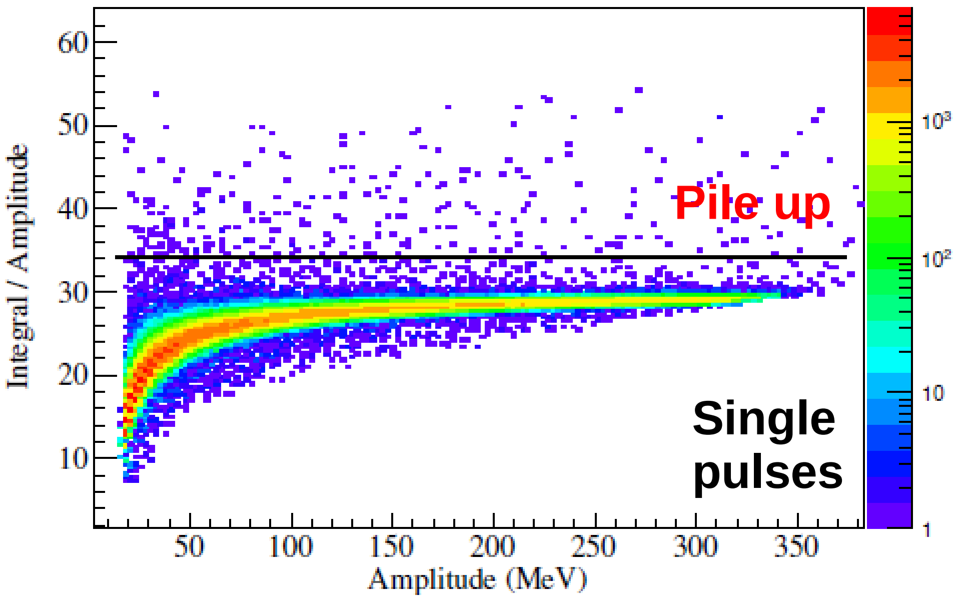
Pulse Detection: Pulse

- Baseline of the signal is continuously subtracted
- Once signal crosses detection **threshold**, samples are collected in a buffer until signal drops below **threshold/4** (one pulse before detection threshold and one sample after end-of-signal threshold are included into the buffer)
- **Pulse is detected if:**
  - Number of samples in the buffer (waveform length) > **min_wf_length**
  - Number of samples in the buffer < **pileup_wf_length**
  - Ratio **Int/A** (pulse area over amplitude) > **min_IA**
  - Ratio **Int/A** (pulse area over amplitude) < **pileup_IA**



A (pulse amplitude)

I = Sum of samples above the threshold



Pile up

Single pulses

Pulse Detection: Pile-up

Pulse pile-up is detected if:

- Number of samples in the buffer > **pileup_wf_length**

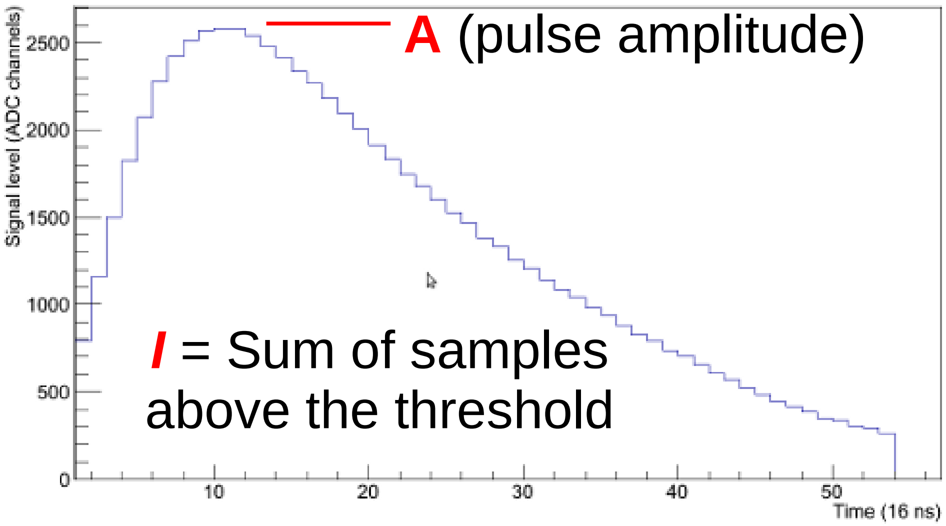
OR

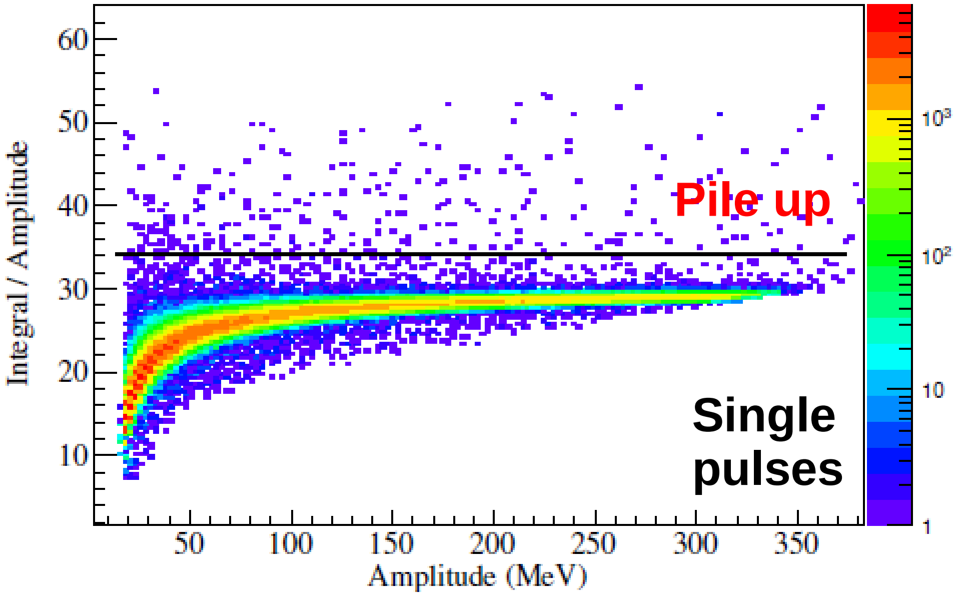
- Ratio **Int/A** (pulse area over amplitude) > **pileup_IA**

If pile-up is detected complete waveform is sent do EMC DC

Hint: if it is necessary to look at waveforms for debugging, set **pileup_wf_length = 1** (all pulses will be treated as pile-up)

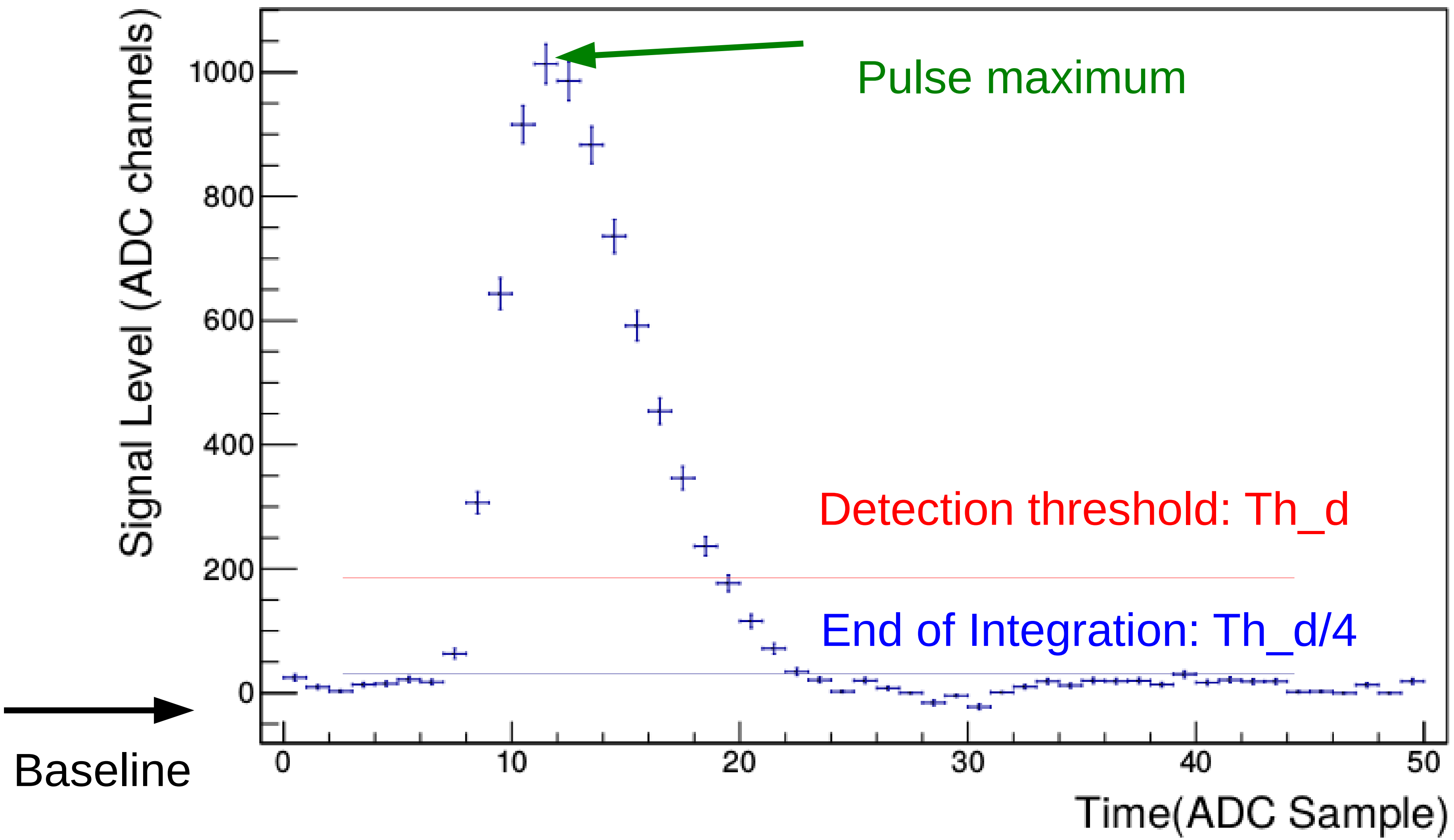
Hint: In waveform mode it is possible to request RAW ADC data (before subtraction of the baseline)





Pulse Detection: long pulses

ADC samples after base-line subtraction



Pulse maximum

Detection threshold: Th_d

End of Integration: Th_d/4

Baseline

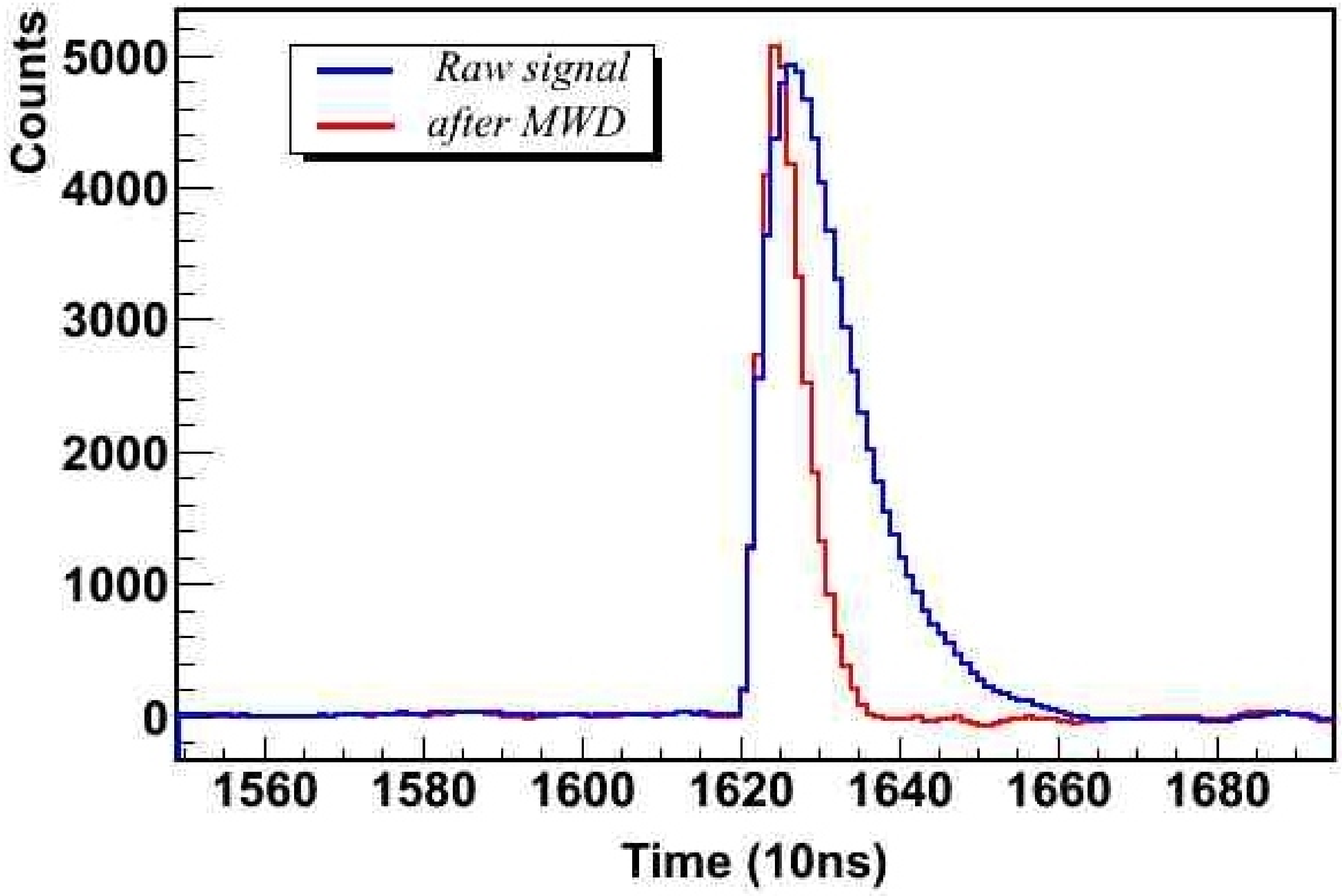
Integration includes: one sample before Th_d till last sample above Th_d/4

If pulse has long tail it might become difficult to discriminate pile-up

MWD Filtering

Pulse-Length reduction

Short Moving-Window Deconvolution is useful for shaped pulses as well



MWD filtering (optional) is done on raw ADC data

Dual-Gain Readout

Pulse-detection is independent on **High** and **Low-gain channels:**

- Algorithms have different threshold but all other detection parameters are the same for each high- low-gain pair;
- If pulse is detected in both High and Low-gain channels then:
  if High-gain signal is **close to clipping** (ADC_MAX_VALUE-300), the **High-gain hit is discarded**, otherwise the **low-gain hit is discarded**
- It is possible to enable only high or low-gain hit data
  - in this case the clipping threshold has no effect;
  - effective for both hit and waveform data

Calibration and Combining

Hit-data can be “calibrated” at the data concentrator**:**

- Calibration is linear: E = a + b*ADC_value
- Output is unsigned 16 bit data
- Calibration constants are different for high and low-gain channels

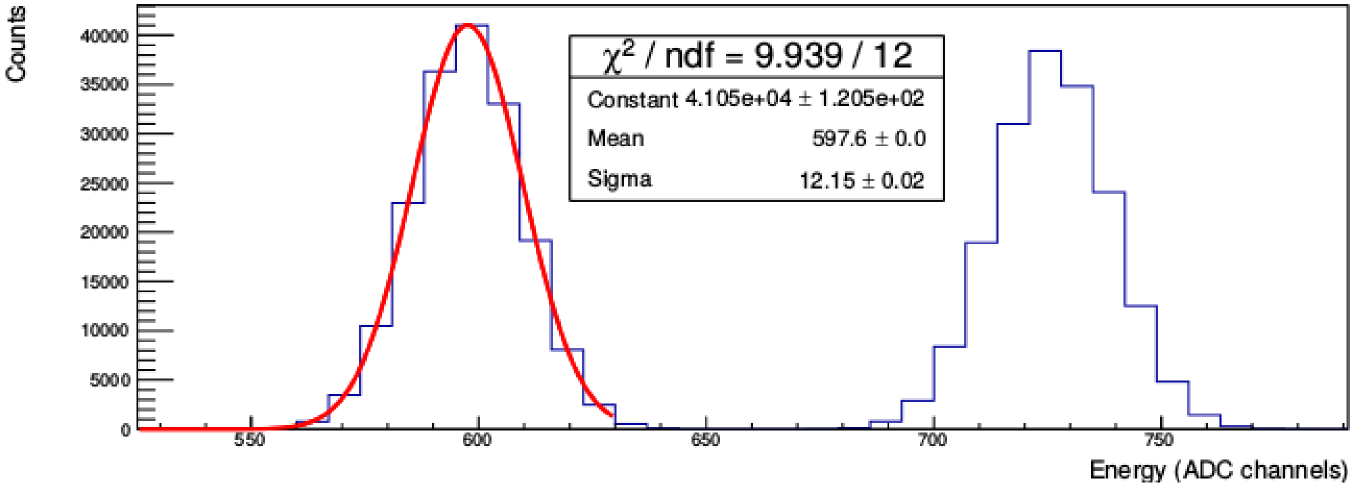
After calibration data from two photosensors (reading out same crystal) can be combined:

- “Combining” map is hard-coded:

  channel #X at the DC input #0 is combined with
  channel #X at the DC input #1
- If pulse-amplitude of both hits is close (difference < 50%) average of energy and time-stamps are taken; new hit is marked as combined data
- Else, both hits are passed with corresponding flag-bit

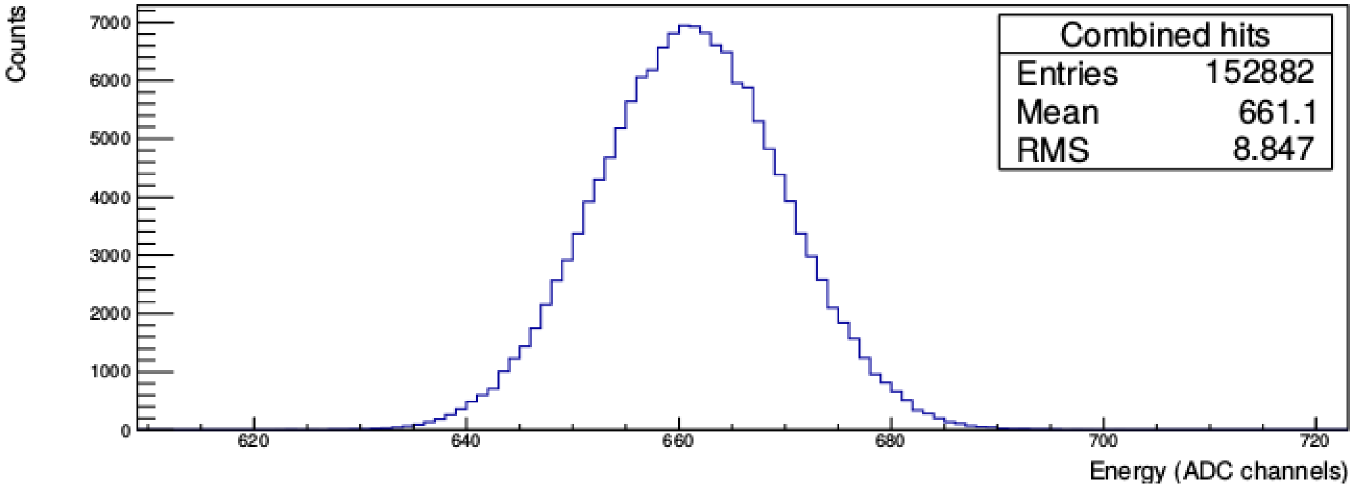
Calibration and Combining
(in action)

Not combined hits (sigma for both channels ~12)



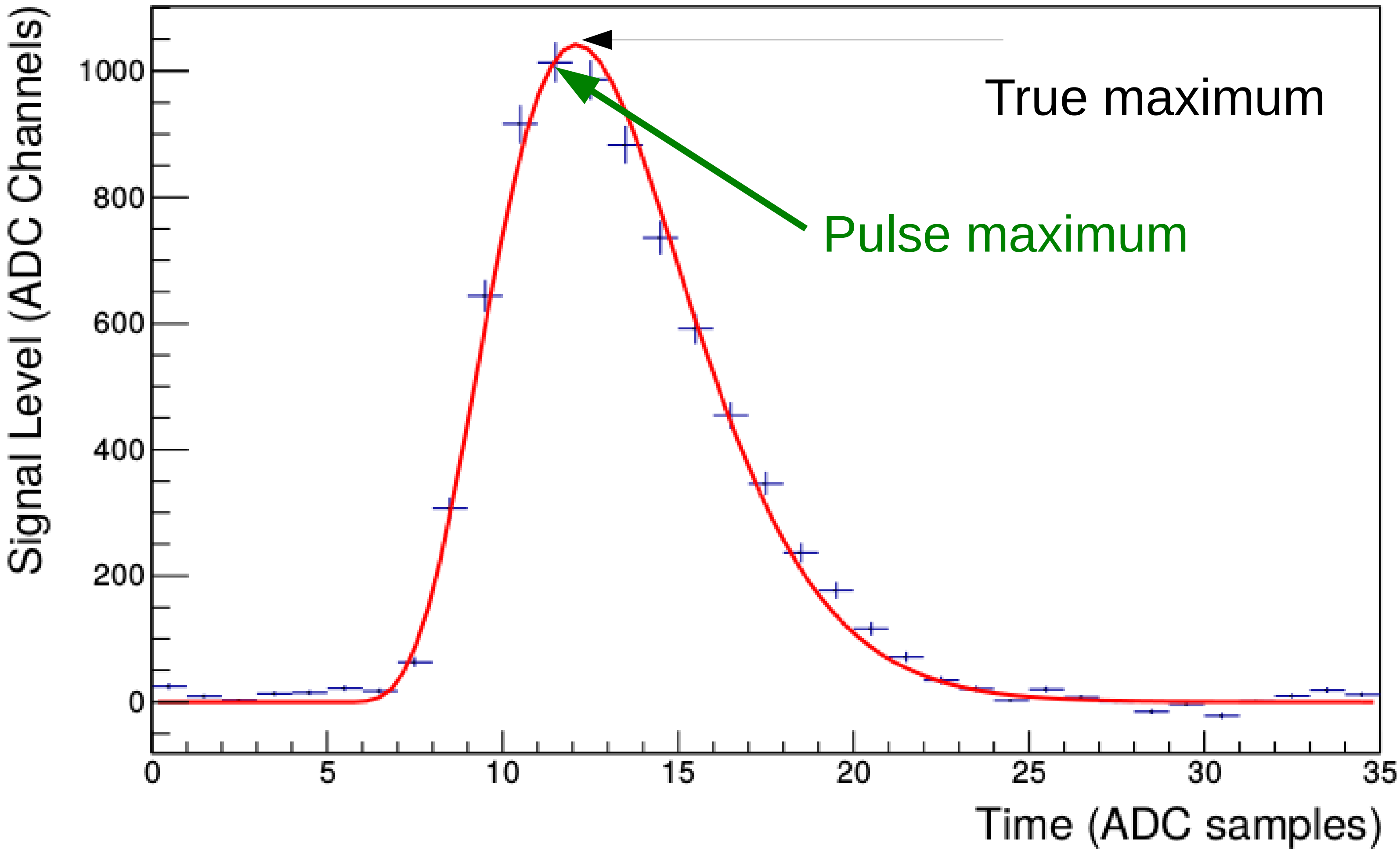
For this test calibration was disabled

Combined hits (sigma ~9)



Same effect observed for the time-difference distributions

Precise Amplitude Measurement

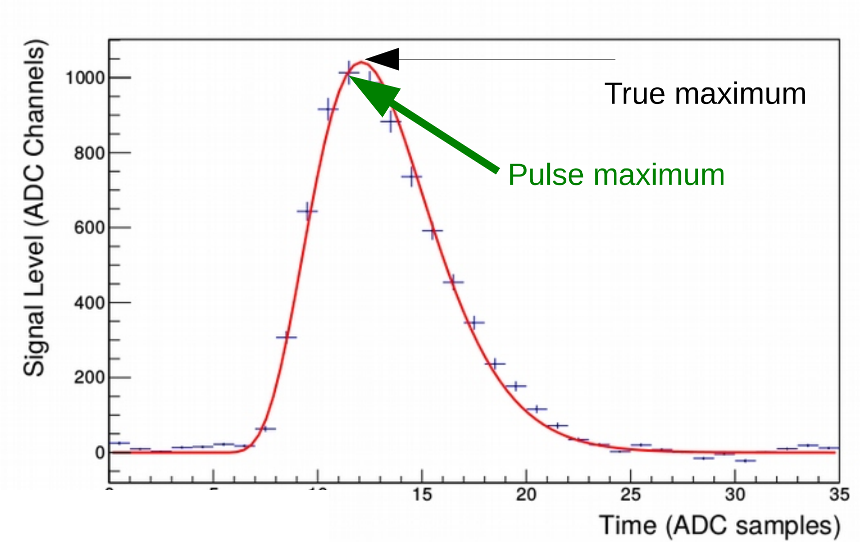


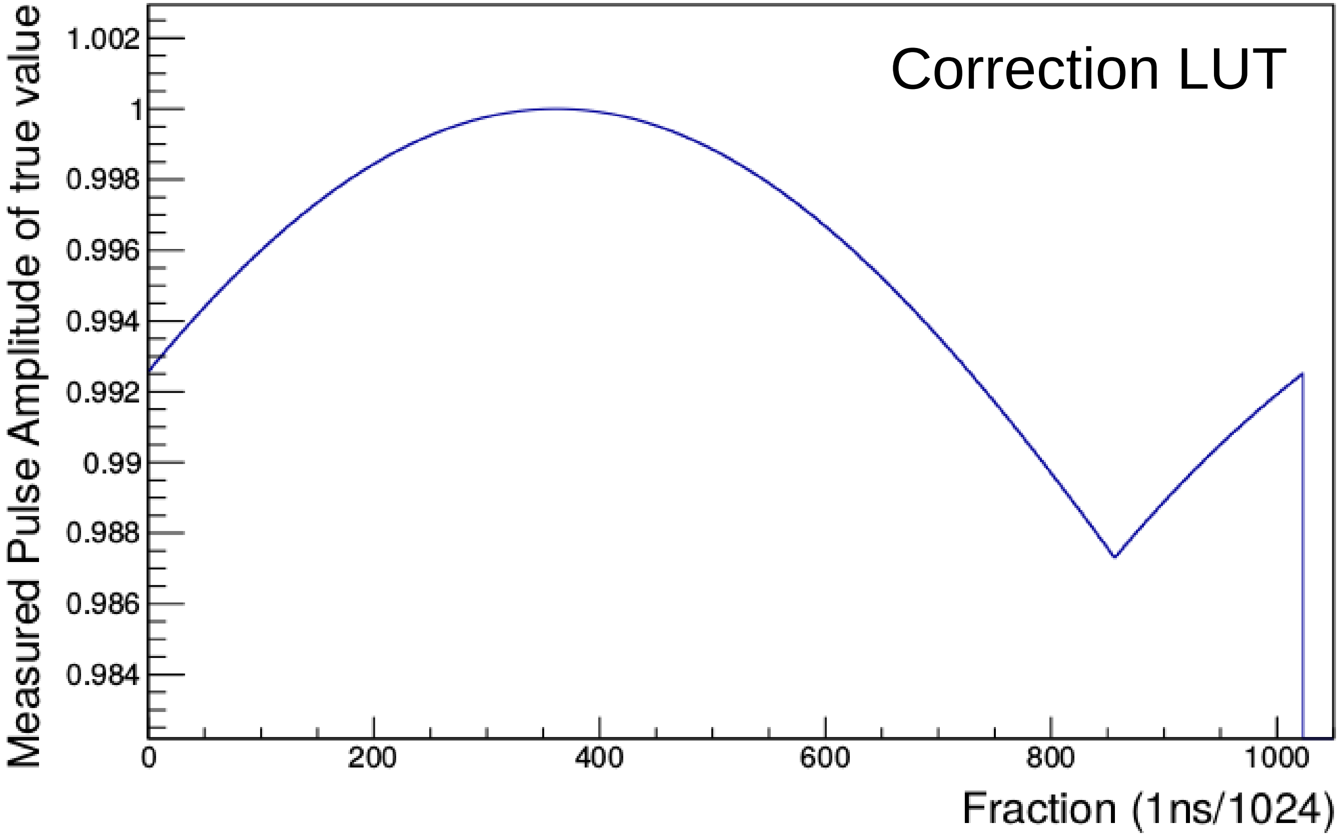
Measured “maximum value” is always smaller than the pulse amplitude:

- Difference depends on pulse phase (fraction) between start of the pulse and phase of sampling clock

Correction for “Maximum”

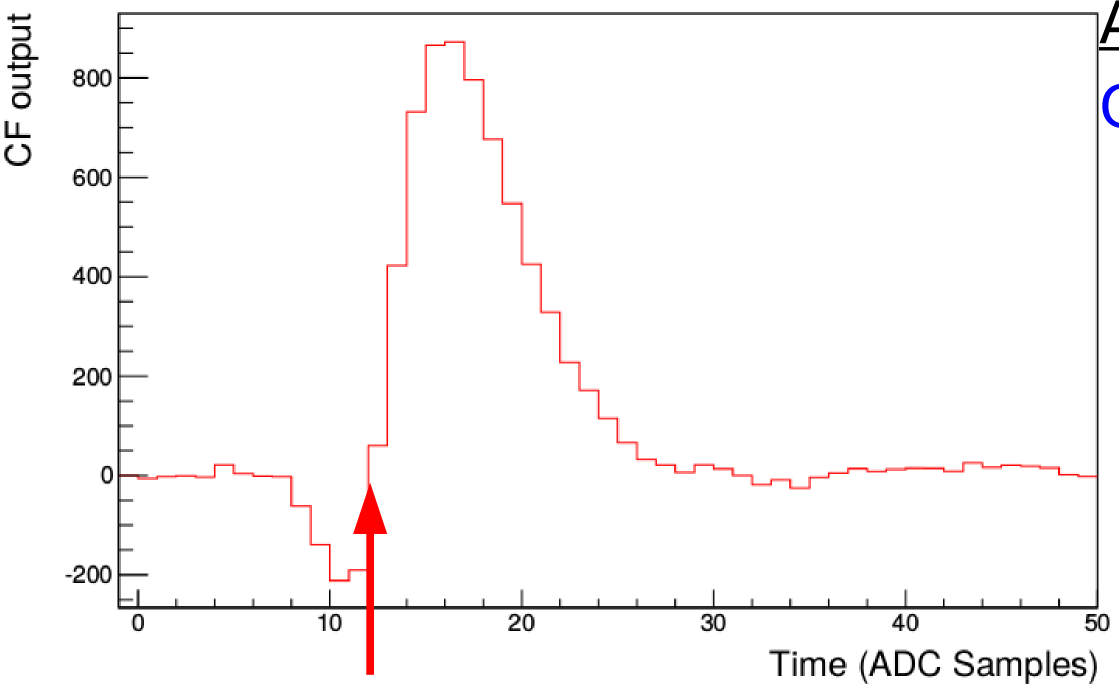
Correction LUT can be constructed to calculate true amplitude of a pulse





- No threshold effects as with “integration”
- Very sensitive to pulse shape (no gain in resolution for realistic case)

Time Measurement



Time stamp: zero-crossing
(linear regression)

Analogue-like implementation:

CFT(n) = MWD(n-*d*) – *R*·MWD(n)

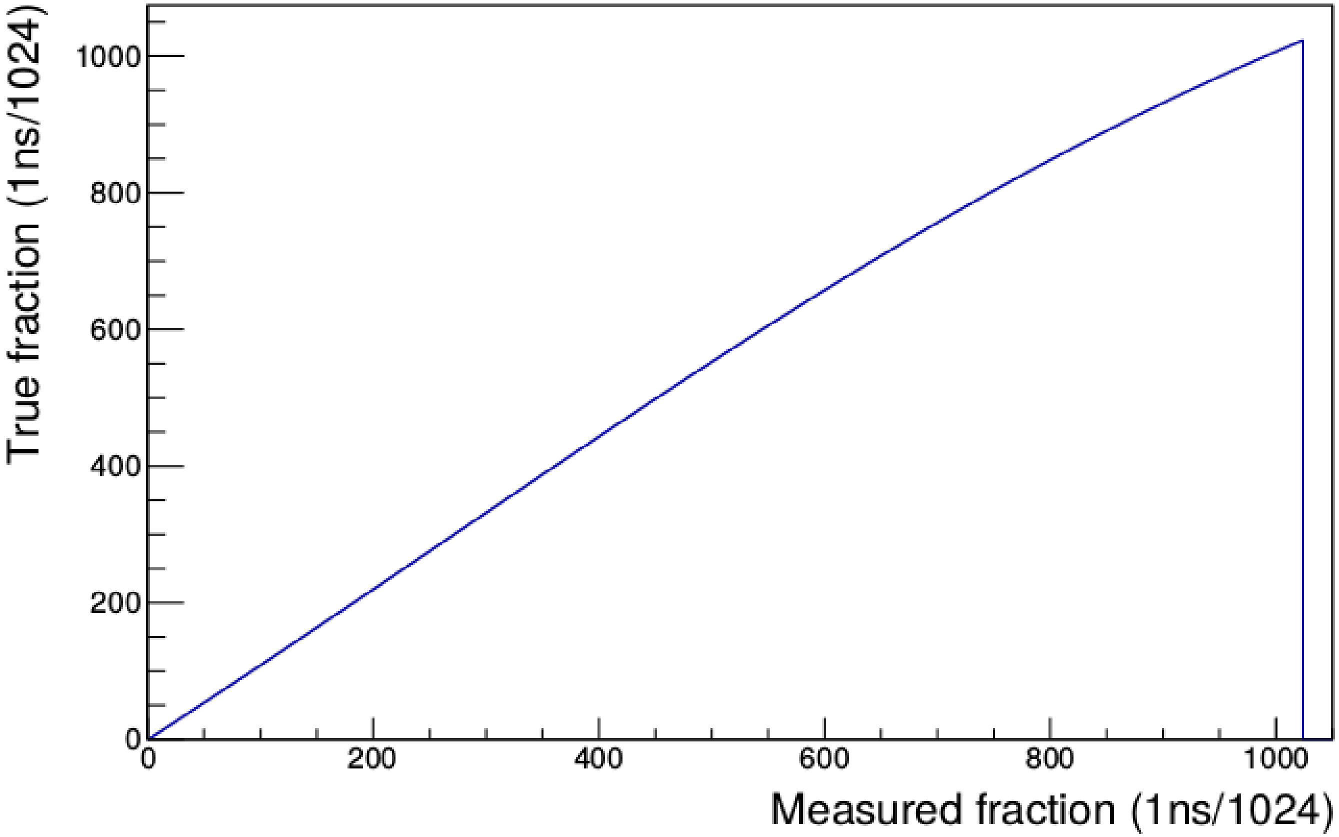
- Delay *d* = signal rise time
- Fraction *R* – to select most linear part of the signal leading edge (**R=1/4**)
- *N* – number for the linear regression
- Symmetry against zero level

Linear regression induces some error in the time measurement

Has to be corrected

Time-Stamp Correction

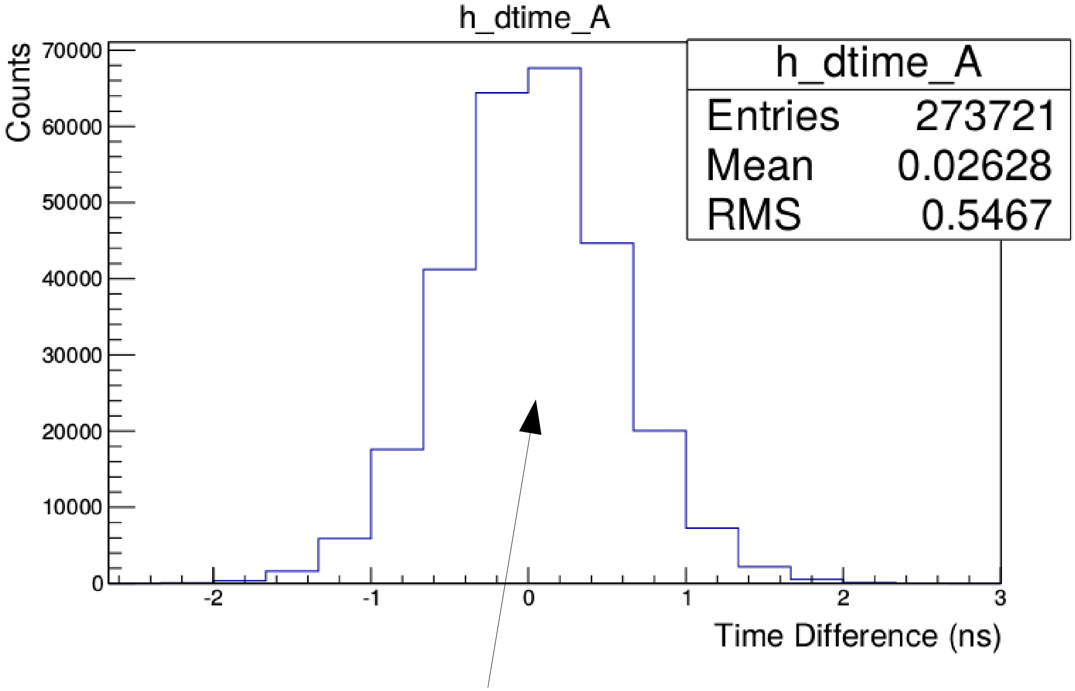
- A correction LUT is calculated using measured waveform
- Correction is performed at the data concentrator
- There is one LUT for one DC (channel-to-channel pulse-shape variations can not be corrected)



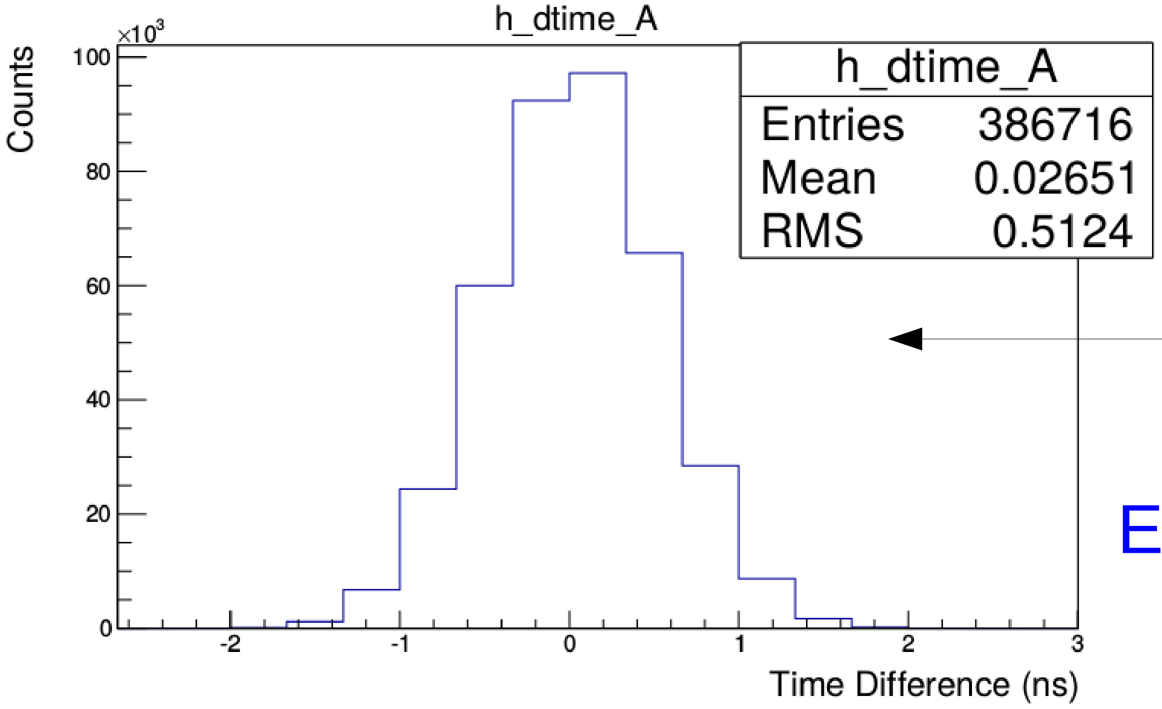
Correction in Action

Measurement with LED pulser and EMC prototype

Time difference between two channels



No Correction



With on-line correction

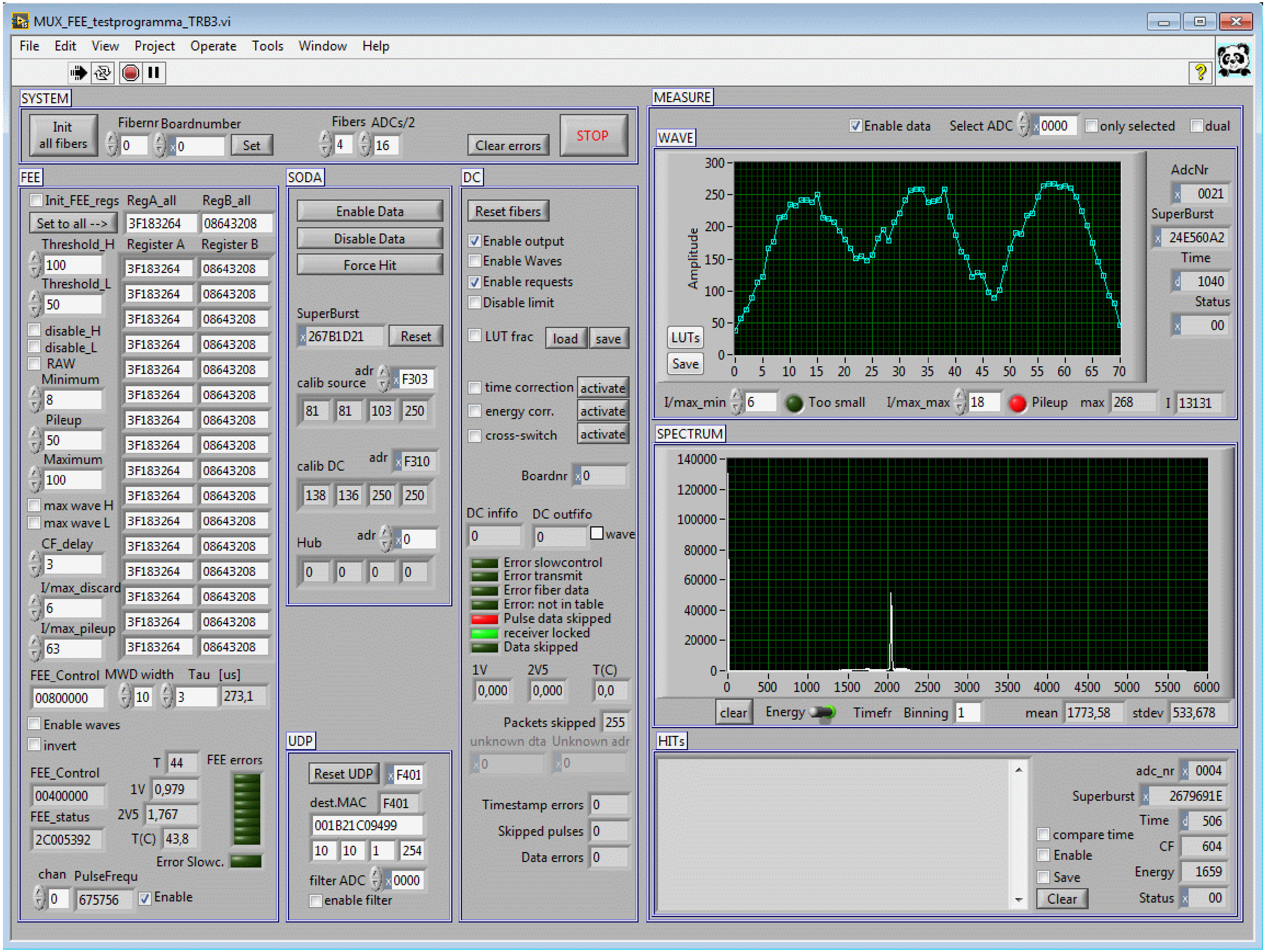
It Works!

Effect is more prominent for higher amplitudes

EMC Readout
from the point of view
# of a user

Labview GUI

Small systems with one data concentrator can be controlled with GUI



SYSTEM: Fibernr 0, Boardnumber 0, Fibers 4, ADCs/2 16

FEE:

□ Init_FEE_regs

Register A	Register B
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208
3F183264	08643208

RegA_all 3F183264, RegB_all 08643208

Threshold_H 100
Threshold_L 50
□ disable_H
□ disable_L
□ RAW
Minimum 8
Pileup 50
Maximum 100
□ max wave H
□ max wave L
CF_delay 3
I/max_discard 6
I/max_pileup 63

FEE_Control 00800000, MWD width 10, Tau 3, [us] 273,1
□ Enable waves
□ invert
FEE_Control 00400000
FEE_status 2C005392
T 44, 1V 0,979, 2V5 1,767, T(C) 43,8
FEE errors
Error Slowc.
chan 0, PulseFrequ 675756, ☑ Enable

SODA:
SuperBurst 267B1D21
calib source adr F303: 81, 81, 103, 250
calib DC adr F310: 138, 136, 250, 250
Hub adr 0: 0, 0, 0, 0

UDP:
F401
dest.MAC F401
001B21C09499
10, 10, 1, 254
filter ADC 0000
□ enable filter

DC:
☑ Enable output
□ Enable Waves
☑ Enable requests
□ Disable limit
□ LUT frac
□ time correction
□ energy corr.
□ cross-switch
Boardnr 0
DC infifo 0, DC outfifo 0, □ wave

Error slowcontrol
Error transmit
Error fiber data
Error: not in table
🟥 Pulse data skipped
🟩 receiver locked
Data skipped

1V 0,000, 2V5 0,000, T(C) 0,0
Packets skipped 255
unknown dta 0, Unknown adr 0
Timestamp errors 0
Skipped pulses 0
Data errors 0

MEASURE: ☑ Enable data, Select ADC 0000, □ only selected, □ dual

WAVE: AdcNr 0021, SuperBurst 24E560A2, Time 1040, Status 00
I/max_min 6, ● Too small, I/max_max 18, ● Pileup, max 268, I 13131

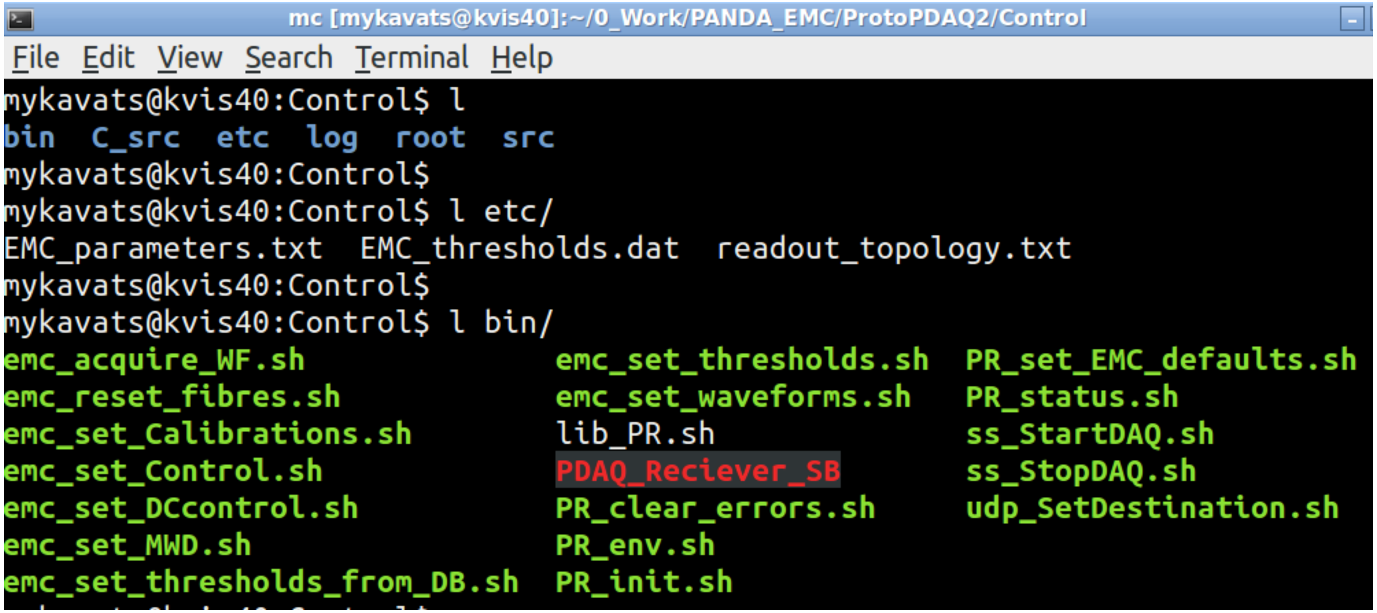
SPECTRUM: Energy, Timefr, Binning 1, mean 1773,58, stdev 533,678

HITs: adc_nr 0004, Superburst 2679691E, Time 506, CF 604, Energy 1659, Status 00
□ compare time
□ Enable
□ Save

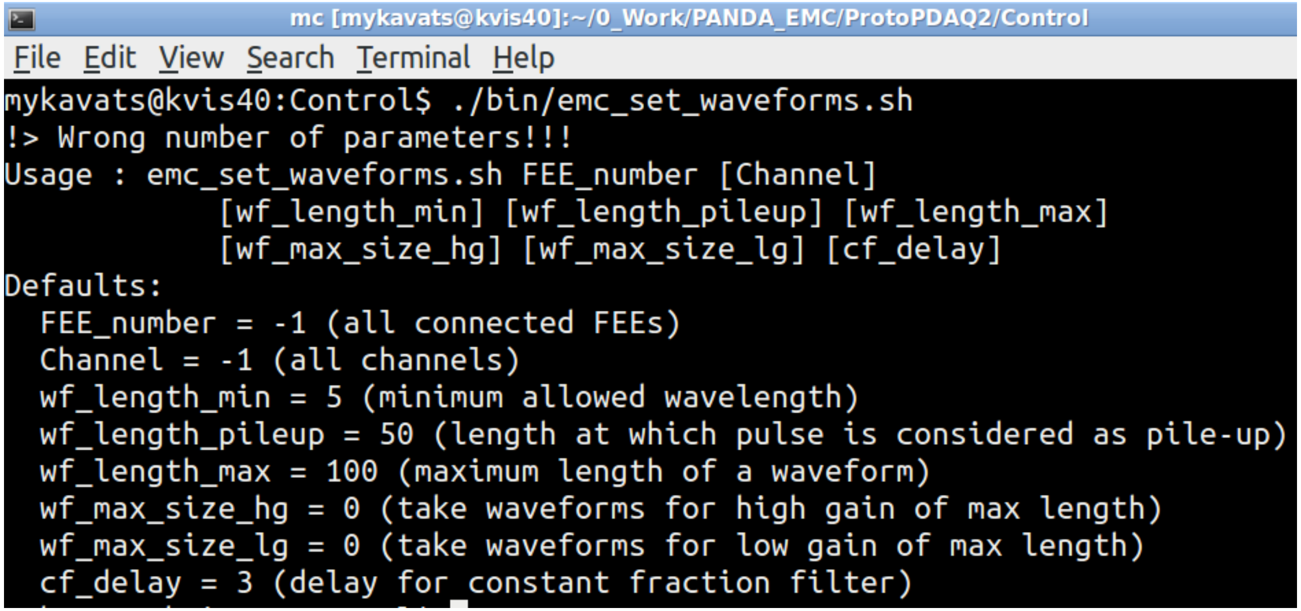
Requires additional Windows PC

CLI for Complex Systems

The CLI package allows to control systems of any complexity:


mc [mykavats@kvis40]:~/0_Work/PANDA_EMC/ProtoPDAQ2/Control
File Edit View Search Terminal Help

```
mykavats@kvis40:Control$ l
bin  C_src  etc  log  root  src
mykavats@kvis40:Control$
mykavats@kvis40:Control$ l etc/
EMC_parameters.txt  EMC_thresholds.dat  readout_topology.txt
mykavats@kvis40:Control$
mykavats@kvis40:Control$ l bin/
emc_acquire_WF.sh              emc_set_thresholds.sh  PR_set_EMC_defaults.sh
emc_reset_fibres.sh            emc_set_waveforms.sh   PR_status.sh
emc_set_Calibrations.sh        lib_PR.sh              ss_StartDAQ.sh
emc_set_Control.sh             PDAQ_Reciever_SB       ss_StopDAQ.sh
emc_set_DCcontrol.sh           PR_clear_errors.sh     udp_SetDestination.sh
emc_set_MWD.sh                 PR_env.sh
emc_set_thresholds_from_DB.sh  PR_init.sh
```

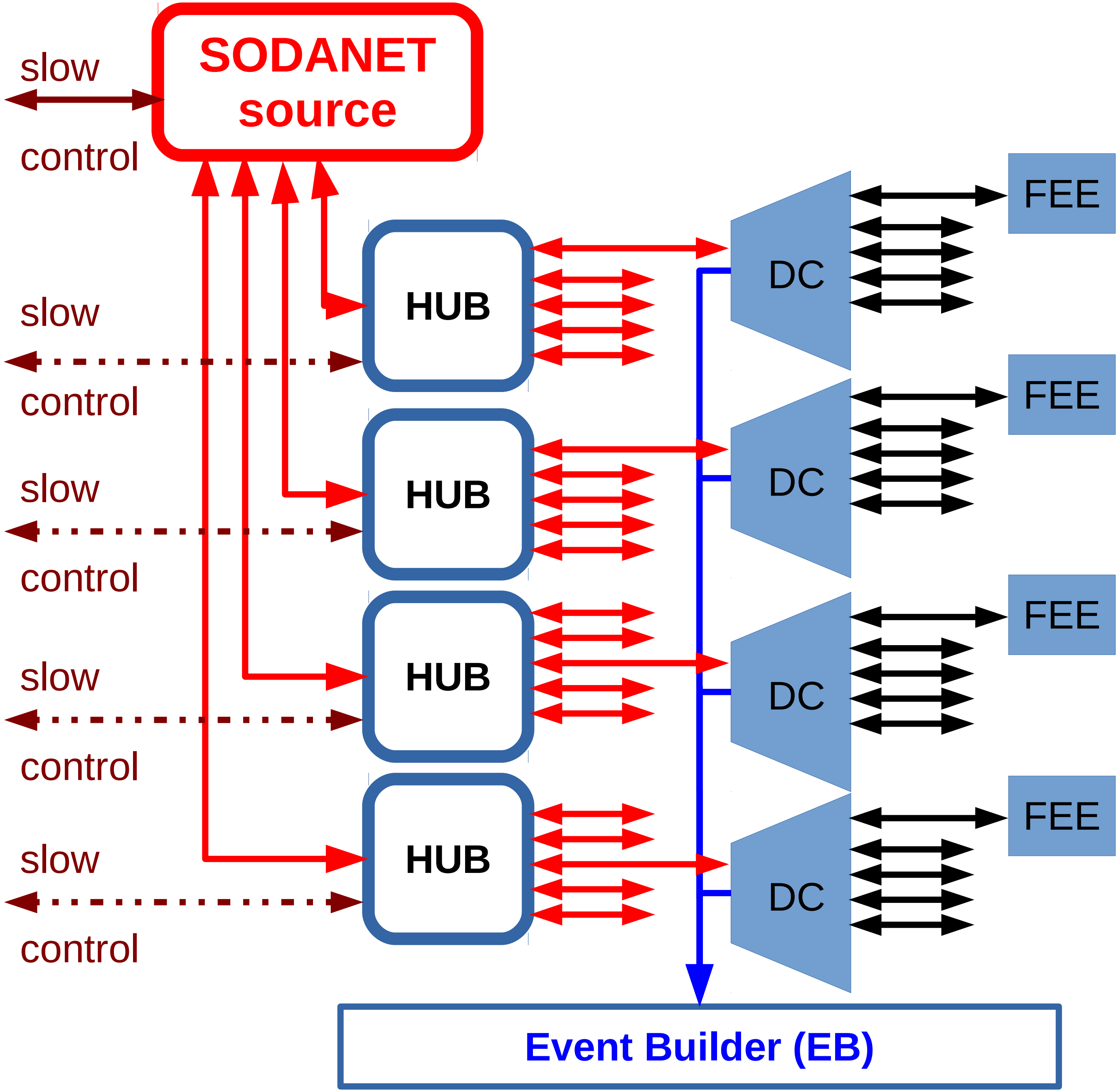

mc [mykavats@kvis40]:~/0_Work/PANDA_EMC/ProtoPDAQ2/Control
File Edit View Search Terminal Help

```
mykavats@kvis40:Control$ ./bin/emc_set_waveforms.sh
!> Wrong number of parameters!!!
Usage : emc_set_waveforms.sh FEE_number [Channel]
            [wf_length_min] [wf_length_pileup] [wf_length_max]
            [wf_max_size_hg] [wf_max_size_lg] [cf_delay]
Defaults:
  FEE_number = -1 (all connected FEEs)
  Channel = -1 (all channels)
  wf_length_min = 5 (minimum allowed wavelength)
  wf_length_pileup = 50 (length at which pulse is considered as pile-up)
  wf_length_max = 100 (maximum length of a waveform)
  wf_max_size_hg = 0 (take waveforms for high gain of max length)
  wf_max_size_lg = 0 (take waveforms for low gain of max length)
  cf_delay = 3 (delay for constant fraction filter)
```

- Allows to set registers for all available FEEs or for selected channel
- Provides complete logging of changes
- Can be extended to control another SODANET compatible subsystems

All tools use **trbcmd** command-line tool – once PANDA run control is established should be rewritten to use **trblib** and standard DB

SODANET Topology



SODANET link:

- Bidirectional
- Synchronous (only in one direction)
- Transfer:
  - source → DC: synchronization information and FEE configuration
  - DC → source: slow control, used for time calibration

Data link (DC → EB):

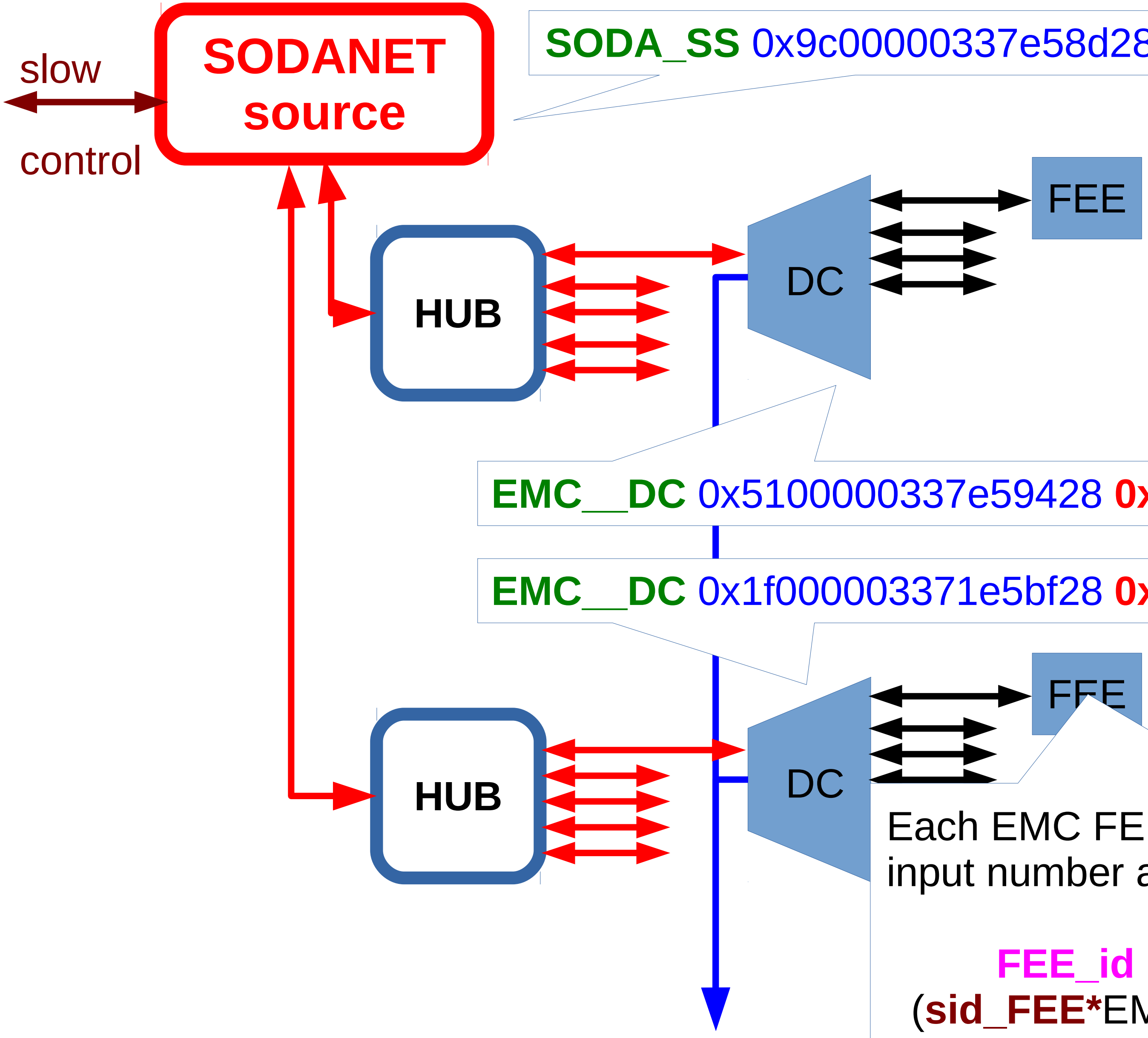
- Unidirectional Ethernet

Link DC ↔ FEE:

- Bidirectional, synchronous
- Protocol up to subsystem

SODANET Topology

(etc/readout_topology.txt)



SODA_SS 0x9c00000337e58d28 **0xf303**

EMC__DC 0x5100000337e59428 **0xf301 0**

EMC__DC 0x1f000003371e5bf28 **0xf302 1**

Each EMC FEE is addressed by the input number at the DC (**fibre_numb**)

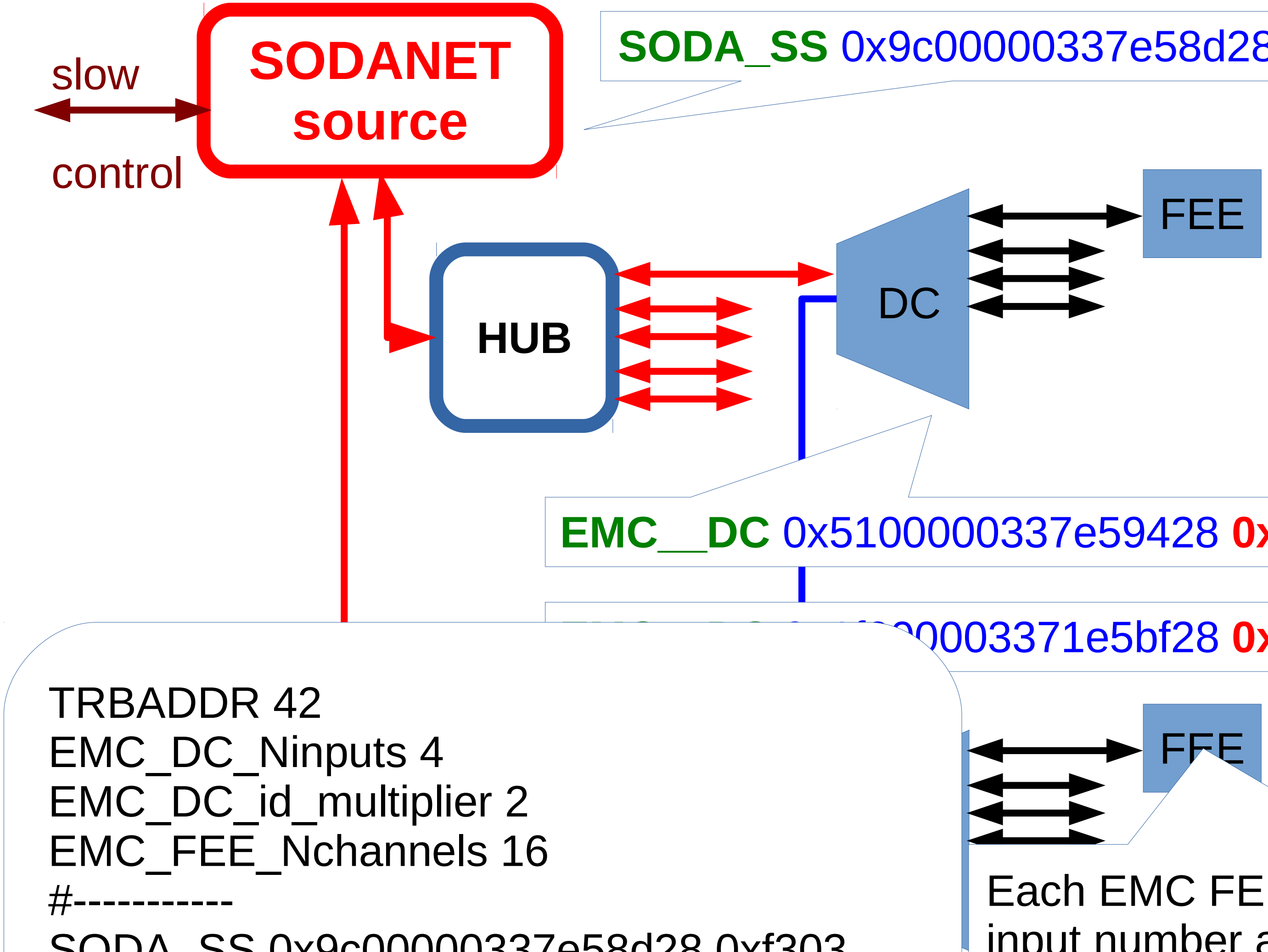
FEE_id = EMC_DC_id_multiplier * (**sid_FEE***EMC_DC_Ninputs + **fibre_numb**)

ADC_channel = FEE_id*16 + channel#

- **Sub-system identifier** – used by shell scripts
- **FPGA unique identifier** – used once to set TRB address
- **TRB address** – used to access registers
- **Seed for FEE unique ID (sid_FEE)**

SODANET Topology

(etc/readout_topology.txt)



SODA_SS 0x9c00000337e58d28 **0xf303**

EMC__DC 0x5100000337e59428 **0xf301 0**

00003371e5bf28 **0xf302 1**

```
TRBADDR 42
EMC_DC_Ninputs 4
EMC_DC_id_multiplier 2
EMC_FEE_Nchannels 16
#-----------
SODA_SS 0x9c00000337e58d28 0xf303
#-----------
EMC__DC 0x5100000337e59428 0xf301 0
EMC__DC 0x1f000003371e5bf28 0xf300 1
```

Each EMC FEE is addressed by the input number at the DC (**fibre_numb**)

FEE_id = EMC_DC_id_multiplier * (**sid_FEE***EMC_DC_Ninputs + **fibre_numb**)

ADC_channel = FEE_id*16 + channel#

- **Sub-system identifier** – used by shell scripts
- **FPGA unique identifier** – used once to set TRB address
- **TRB address** – used to access registers
- **Seed for FEE unique ID (sid_FEE)**

Default Settings
(etc/EMC_parameters.txt)

If parameters are not specified directly for the command the default values from etc/EMC_parameters.txt are been used

```
================================================
# emc_set_thresholds.sh
thr_low_gain=50
thr_high_gain=100
mode=0
IA_min=6
IA_max=18
Raw_ADC_data=0
======================
# emc_set_waveforms.sh
wf_length_min=5
wf_length_pileup=50
wf_length_max=100
wf_max_size_hg=0
wf_max_size_lg=0
cf_delay=3
```

0 - enable low and high gains
1 - disable high gain
2 - disable low gain
3 - only SODA forces hits

If 1, raw ADC data will be acquired in waveforms

If 1, waveforms of maximum length will be acquired for selected channel (high/low agin)

Should be ½ of the pulse rise time

Default Settings

(etc/EMC_parameters.txt)

```
========================================================
# emc_set_MWD.sh
mwd_width=0
mwd_tau=1
```

Parameters of the MWD filtering.
If mwd_width=0, filtering is off

```
========================================================
# emc_set_Control.sh
reset_all=0
invert_inputs=0
clear_errors=1
enable_waveforms=1
channel_freq_meas=0
res_fpga_mon=1
fpga_mon_addr=0
```

Invert ADC inputs

```
========================================================
# emc_set_DCcontrol.sh
enable_data_cn=1
enable_wfdata_cn=1
disable_packet_limit=0
source_mux_stat=0
```

Allow data flow to compute nodes

Allow waveform-data flow to compute nodes

Typical Session

Enter working directory (optional)

```
> cd ~/Control
```

Set shell environment

```
> .   ./bin/PR_env.sh
```

Initialize log file (log/daq.log)
Sets TRB addresses to all devices
Sets FEE_id for all EMC FEEs

```
> ./bin/PR_init.sh
```

Sets default values for all EMC FEEs and DCs

```
> ./bin/PR_set_EMC_defaults.sh
```

If compute nodes not used, sets destination for UDP packets

```
> ./bin/udp_SetDestination.sh 0x682346923 192 168 1 1
```

Starts data-receiving program on “DAQ” PC

```
> ./bin/PDAQ_Reciever_SB root/data.root
```

Enable data taking by FEEs

```
> ./bin/ss_StartDAQ.sh
```

```
> ./bin/ss_StopDAQ.sh
```

Additional Tools

- On-line monitoring of incoming data:
  - Waveform data
  - Hit data
  - For all available ADC channels, or only for selected

- Automatic tool which collects given number of (raw-data) waveforms from all available ADC channels (high and low gain), calculates noise level, deduces thresholds (in terms of multiple of RMS value), stores new threshold values in DB (ASCII) and sets values to all available FEEs

- Root macros which allow:
  - Calculate threshold values from given waveforms
  - Calculate correction LUTs (amplitude and timing)
  - Simulate feature-extraction algorithms on waveform data

Summary

The EMC readout is ready for non-expert operation and should be used by different groups to find possible bugs

Some parameters should be optimised for different amplifiers:

- Energy measurement by amplitude or integral?
- What is the best MWD filtering?
- Is pulse-shape stable enough to benefit from correction LUTs?
- What are the optimal hit-detection parameters
- Can “hits-combining” be used to decrease hit-detection thresholds?

Optimisations should be done using measurements with EMC prototypes and simulations (simulation root macro is provided)