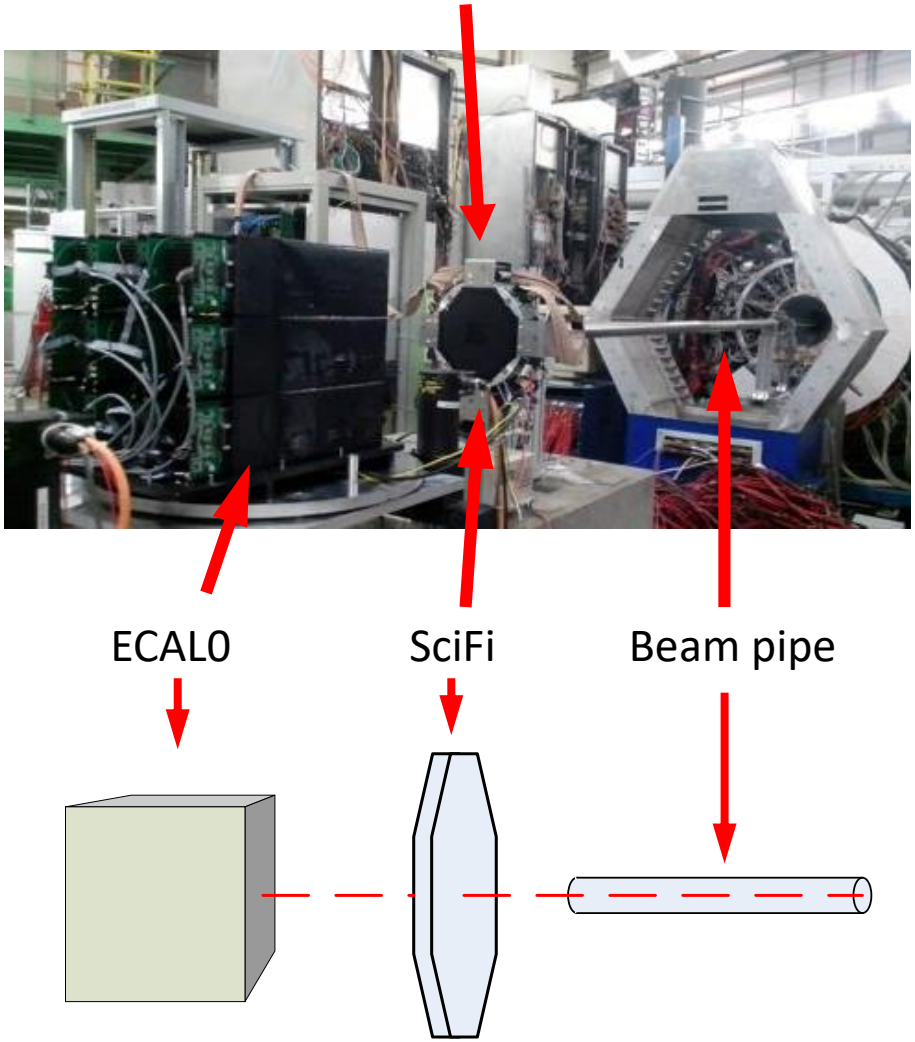
On-line Pulse Parameterization from Sampled Waveforms

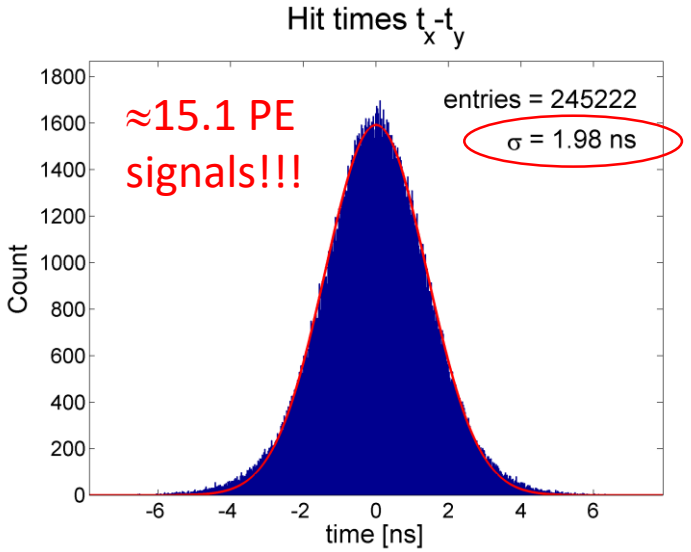
Marcin Ziembicki

Warsaw University of Technology / AstroCeNT

How did it all start?

Small scintillating fiber detector for triggering a test beam experiment for an electromagnetic calorimeter





- Elektronen-Stretcher-Anlage (ELSA)

low-intensity electron source

- Sci-Fi hodoscope

trigger & beam monitoring

- Calorimeter blocks on a movable table (3x3 tower)

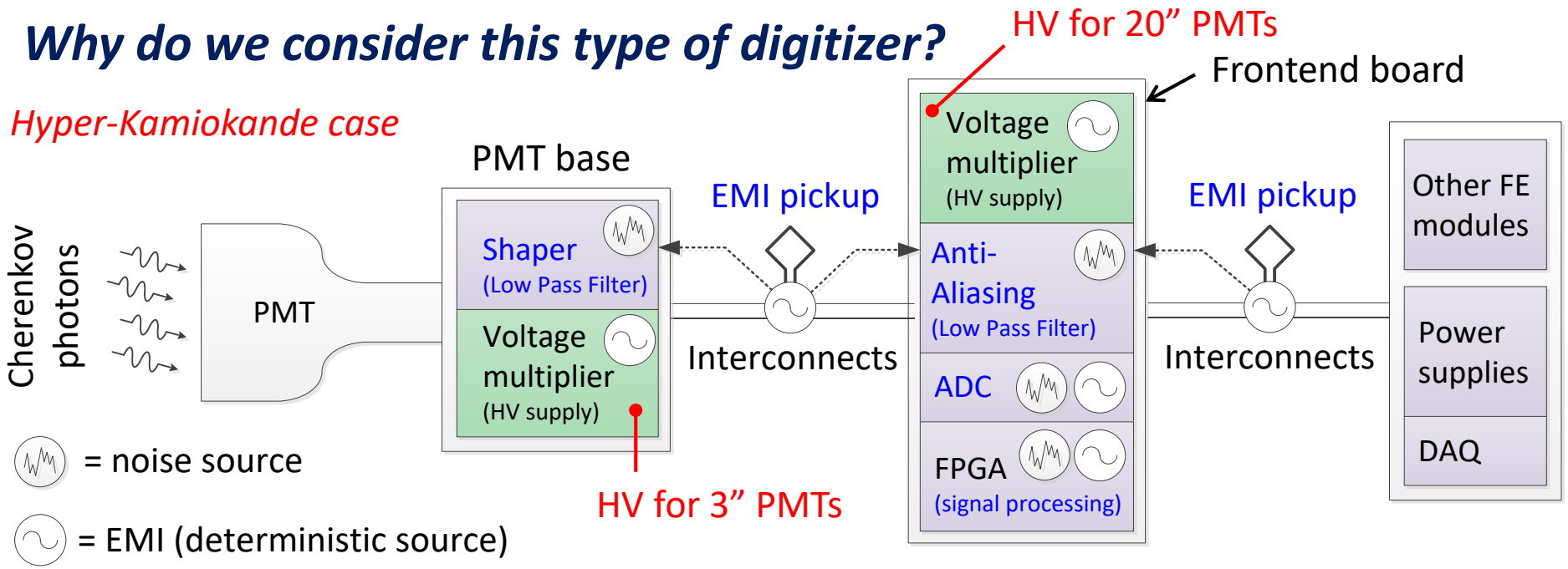
angular and spatial scanning

- DAQ: 80 MSPS ADC

both calorimeter and Sci-Fi

Digitization – Waveform Sampling

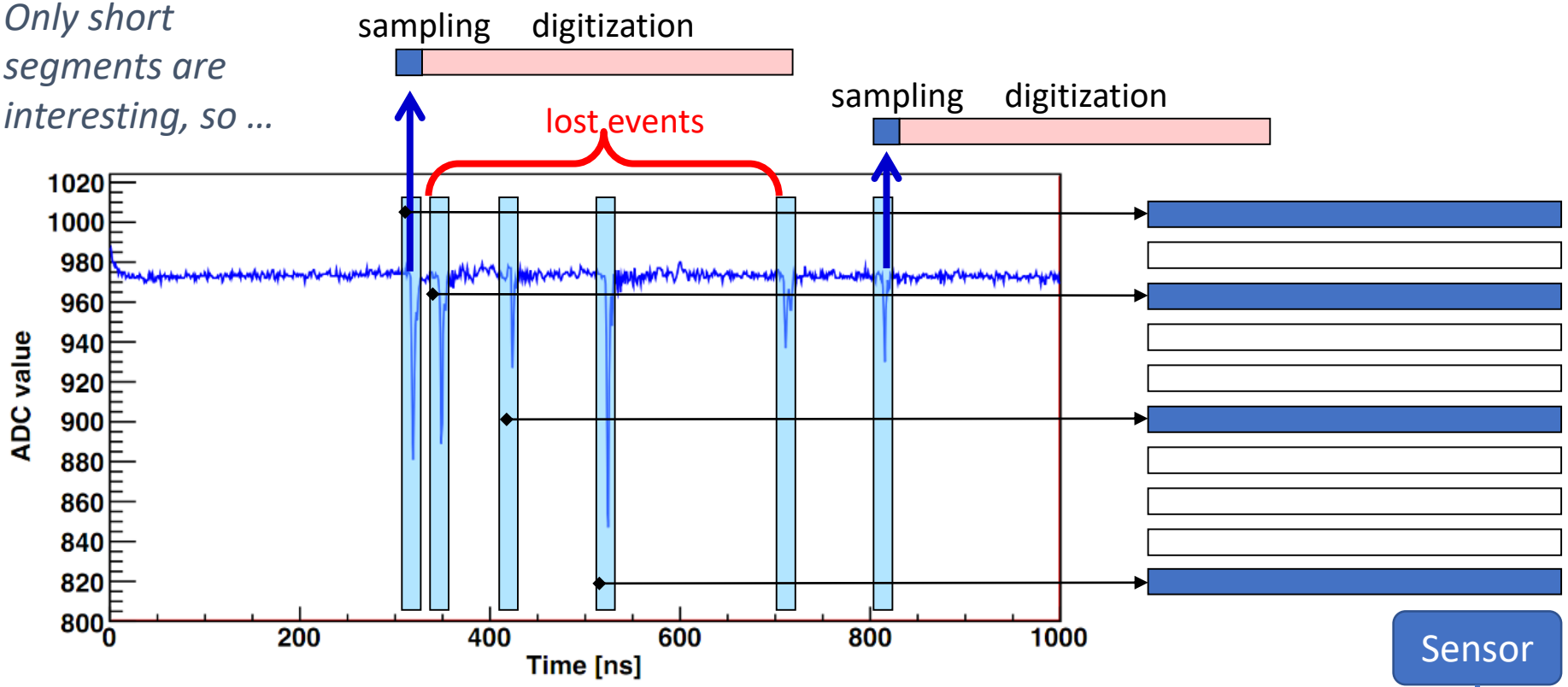
Why do we consider this type of digitizer?

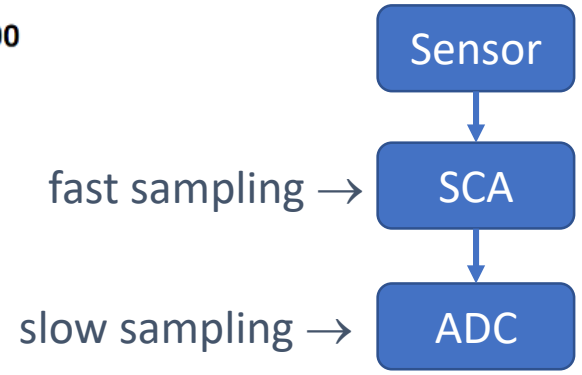


- Possibility to implement completely dead-time free system.
  - Better ability to tag decay electrons that occur at short decay times and high muon energies.
  - E61 case – ability to disentangle in-bunch pile-up
- Pulse processing on-the-fly (i.e. send only time/charge – most of the time)
- **Can subtract off periodic EMI by digital filters implemented in FPGA firmware.**
- There is a price to pay: **power consumption,** cost, **data rate**.
  - We need to reduce all without affecting physics performance

Lowering Power Consumption – Switched Capacitor Arrays (DRS4 example)

Only short segments are interesting, so …





Avoiding dead time in capacitor arrays:

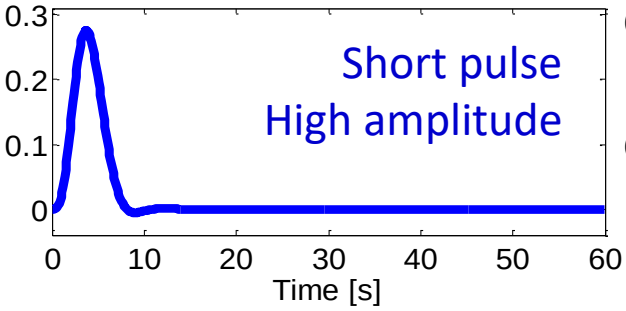
- Use chip with segmented memory
  - Latch only part of array, keep other parts active (DRS5 solution – not yet available)
- Use multiple arrays for single waveform

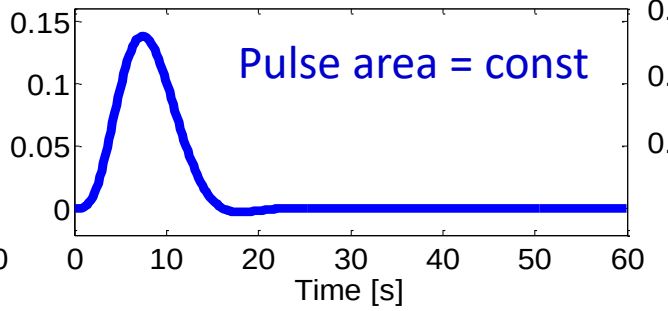
Study of Sampling Systems

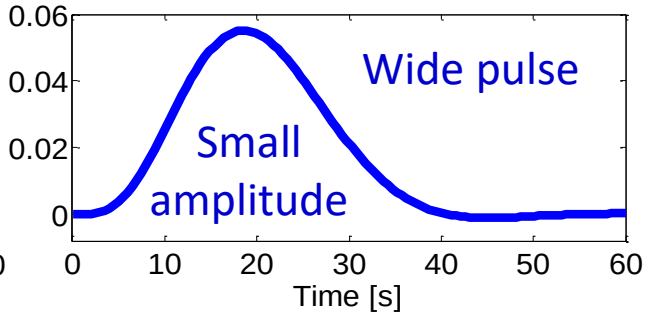
High resolution → Low resolution



How **poor** can the **system specs** be to still be able to tell **when** and how big the **pulse was** with **satisfactory precision**?

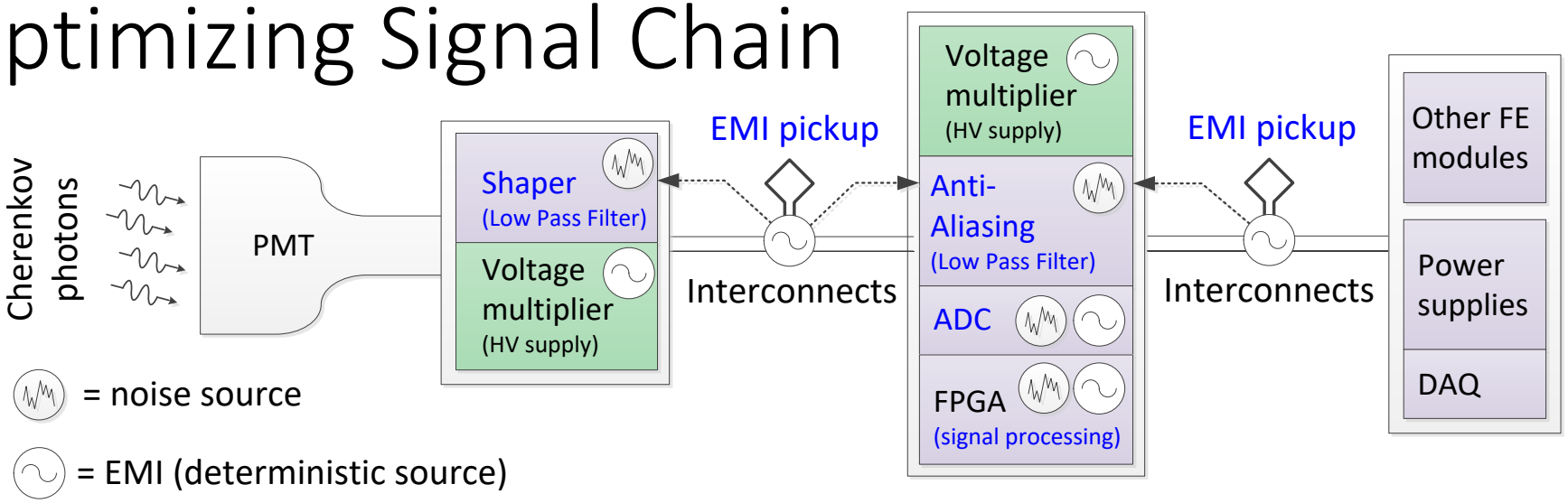






High bandwidth → Low bandwidth

Optimizing Signal Chain



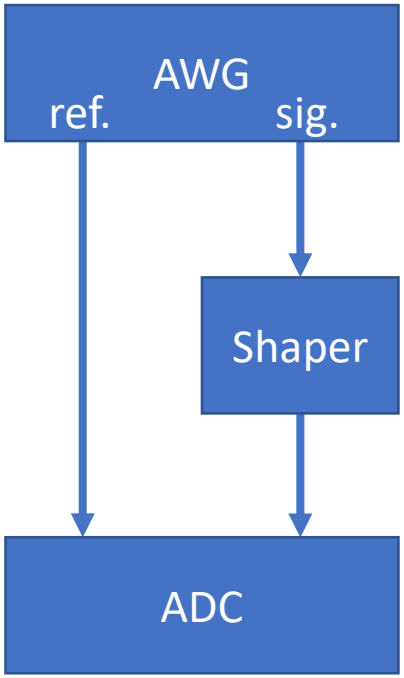
- Type of shaper/anti-aliasing filter?
- Speed and resolution of the ADC?
- Signal processing methods and sharing of signal processing between FPGA and DAQ
- Optimization of resource usage within the FPGA → talk by M. Suchenek
- Quality of time & charge estimates
  - Waveform compression in case this quality is unsatisfactory → talk by G. Pastuszak
- Treatment of pulse pile-up
- **Model of the full signal chain**
  - **Will allow exploration of various variants of shaper/ADC combinations without the need for many prototypes**

Timing Resolution of Sampling Digitizers

PURPOSE OF THE STUDY:

Determine how fast and how precise does a system needs to be to achieve given performance specs?

- Use AWG instead of PMT.
- Use large reference pulse (timing accuracy $\sigma \approx 10$ ps) and small, shaped signal pulse (1 mV ~ 100 mV).
- Apply signal processing methods and calculate time difference Δt between ref. and sig. channels.
- Repeat multiple times and compute RMS of Δt values.
- Two shapers:
  - 15 ns and 30 ns rise time (10% to 90%), 5-th order **Bessel-type** low-pass filters.
- Shared project WUT/TRIUMF



Agilent 33600A (1 GSPS/80 MHz)

Custom shapers

Commercial ADCs (CAEN)

DT5724 (100 MSPS/14b)

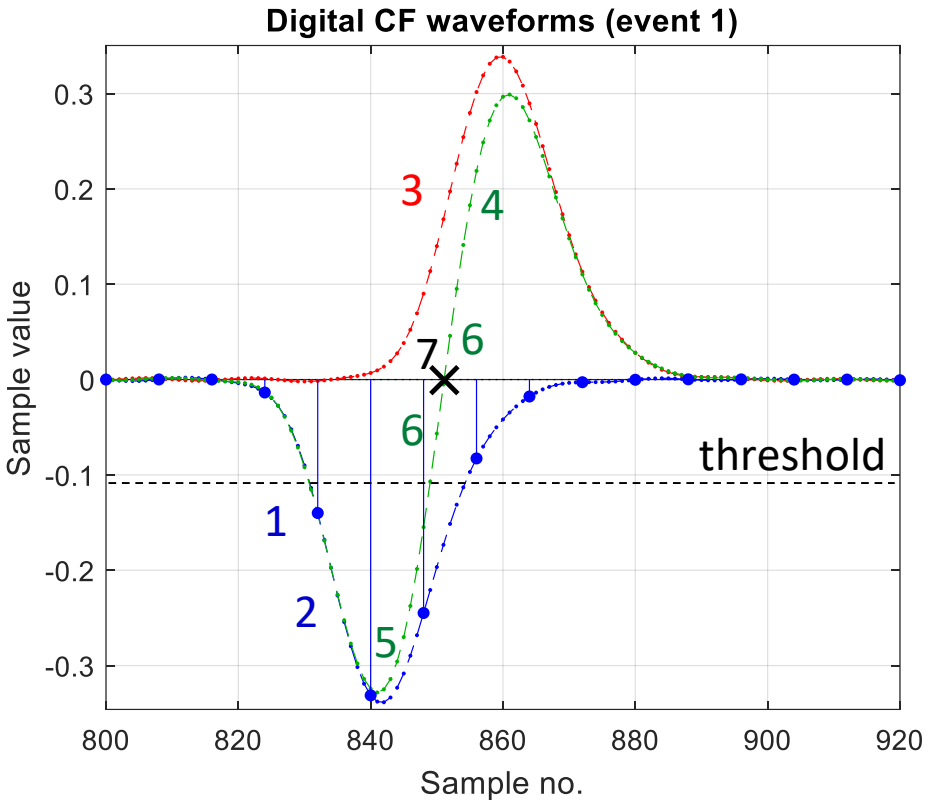
V1720 (250 MSPS/12b)

V1730 (500 MSPS/14b)

Signal Processing Methods

Digital Constant Fraction Discriminator:

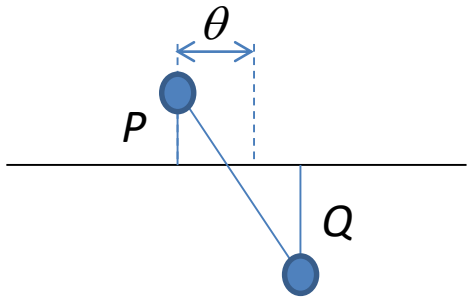
- Simple processing → needs little FPGA resources
- Does not make any assumption as to the pulse shape
- Favors high sampling rate, but some improvements are possible for low sampling rates if pulse shape is invariant
- Poor performance in low SNR conditions



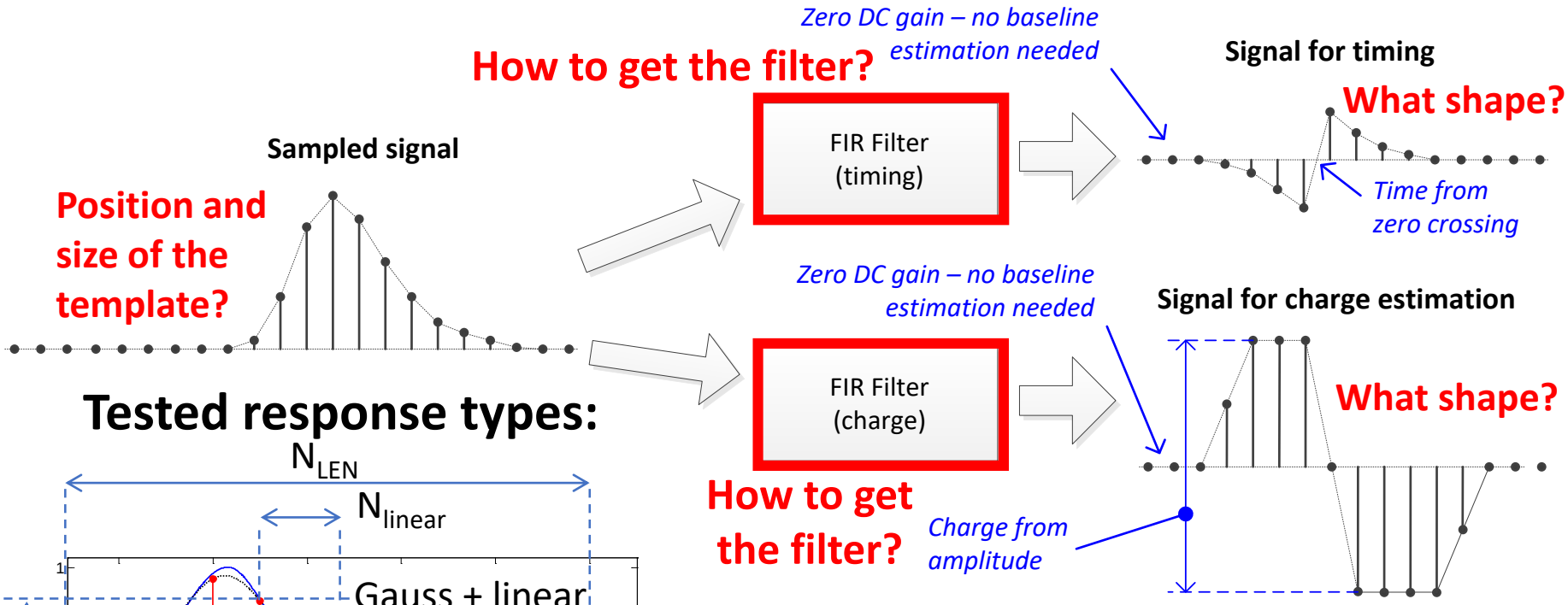
Time errors and possible correction

θ - actual sub-sample shift

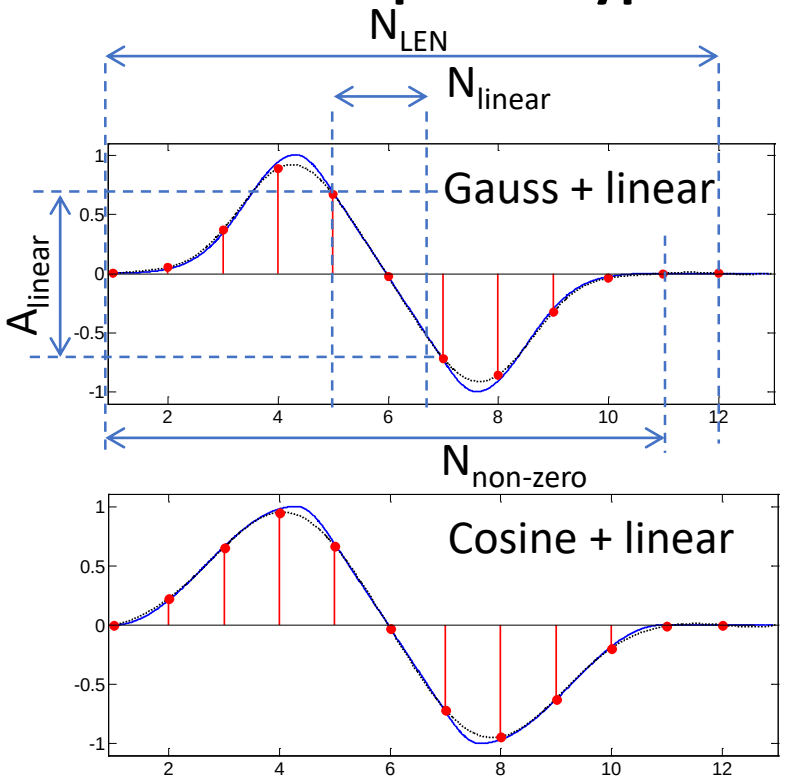
$$CR = \frac{P}{P - Q}$$



Signal Processing – FIR DPLMS



Tested response types:

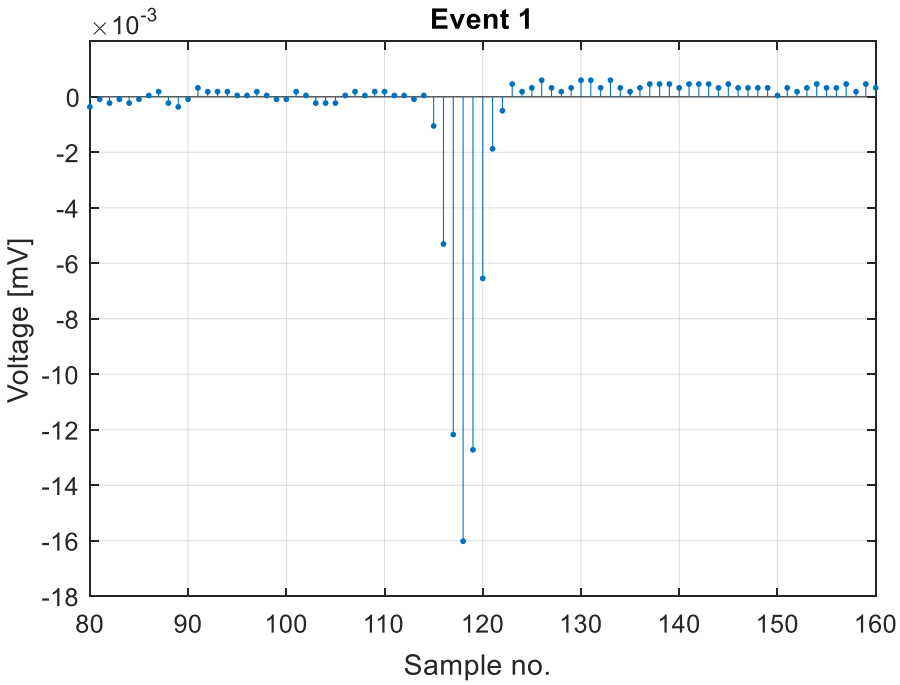


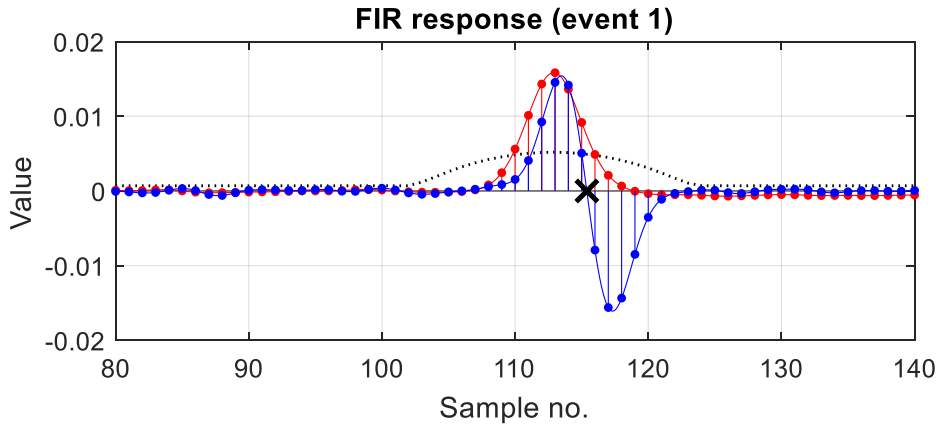
… or simply subtract pedestal and integrate.

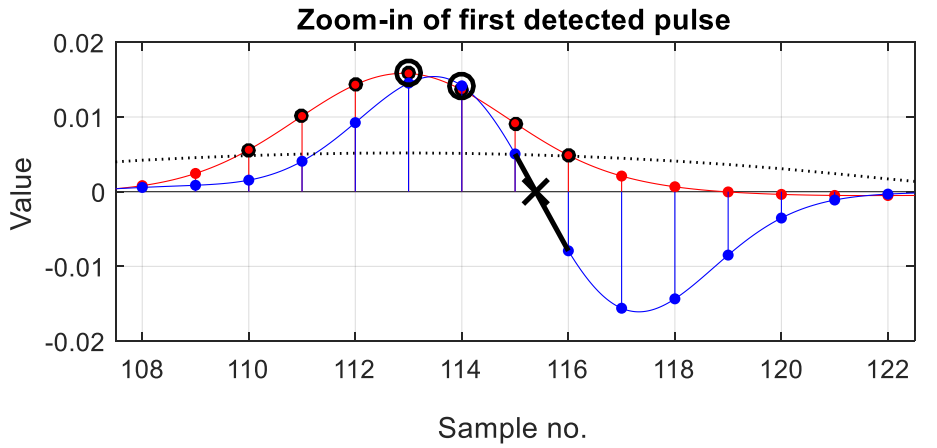
- FIR = Finite Impulse Response
- 'Black-box' approach → transform **known** input into desired output, don't care how.
- Arbitrary filter characteristic possible.
- Filter should be 'optimal' → **minimize certain cost function**.

Gatti E., et al., "*Digital Penalized LMS method for filter synthesis with arbitrary constraints and noise*", NIM A523, 167-185, 2004

Signal Processing - FIR Filters







- Trigger on ‘gate’ filter response (red)
- Use adaptive threshold to prevent false positives (dotted black line)
- Timing using ‘timing’ filter response (blue)
- Apply correction to counteract non-linear shape of the waveform near zero-crossing.

Method assumes that shape is invariant

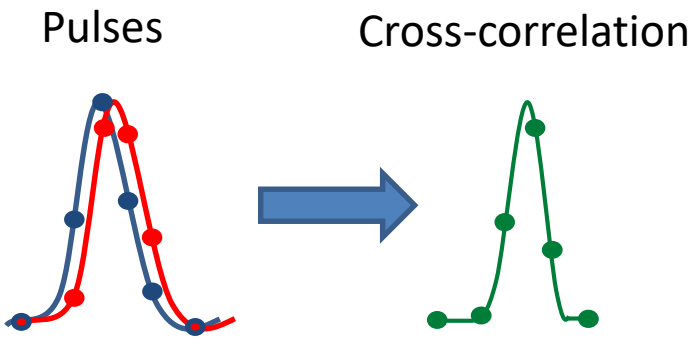
Need on-line Quality Factor to judge accuracy of estimation

Signal Processing – Continued

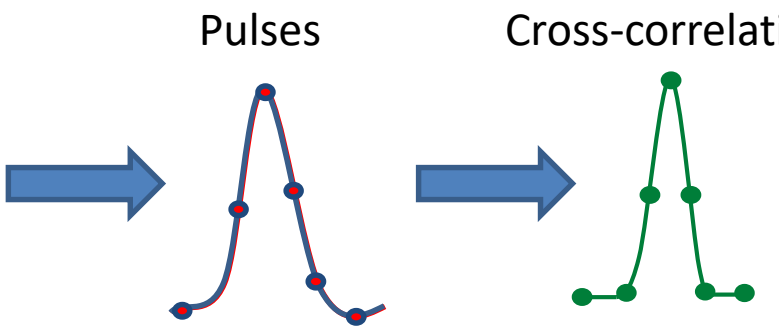
Matched FIR Filter and Cross-Correlation Processing:

- Much more complex processing
  - Works well with filter orders of 9-12
- **Assumes that shape is invariant**
- Similar timing performance to zero-average FIR filter
- Relatively easy to disentangle piled-up pulses

Misaligned pulses

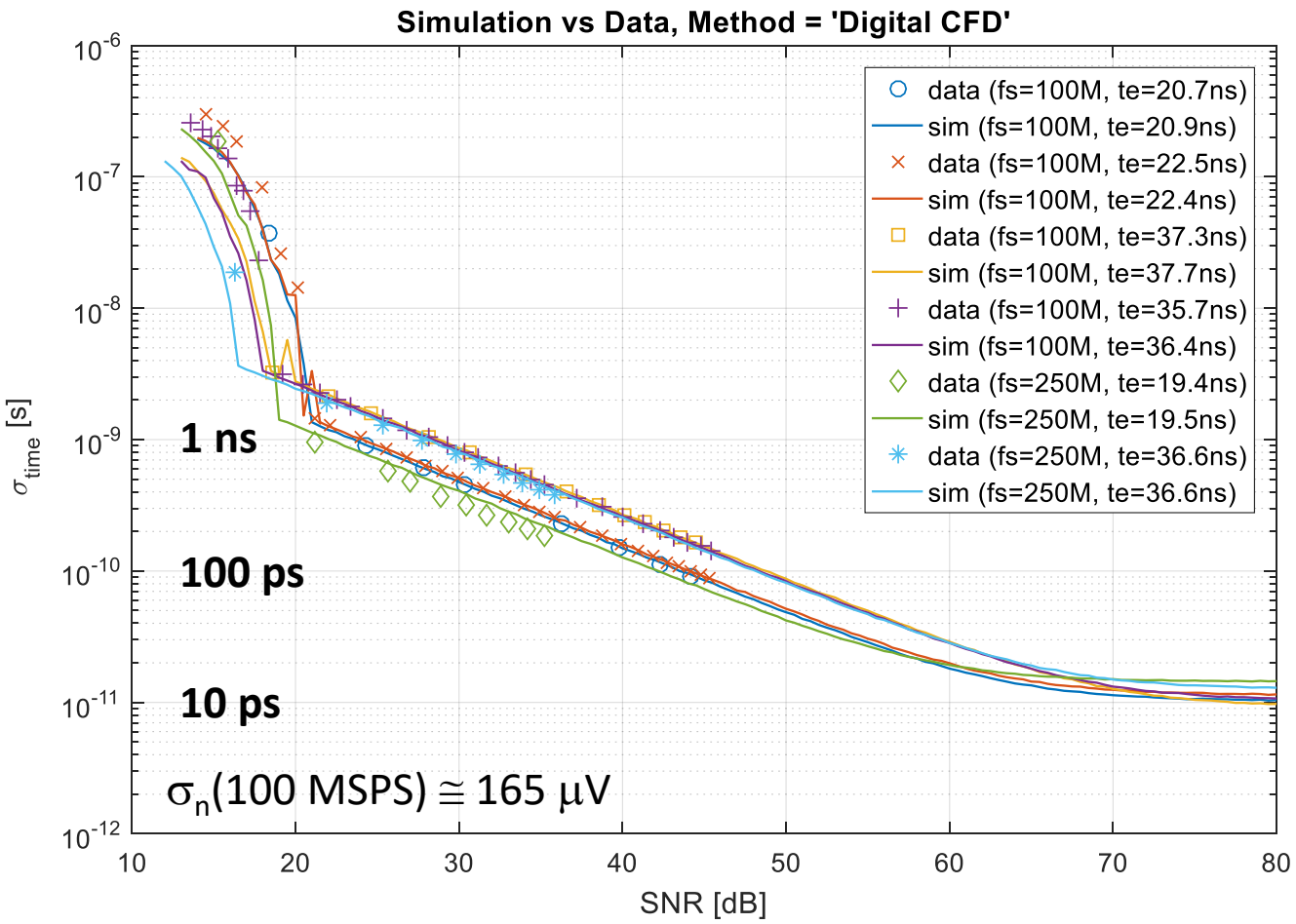


Aligned pulses



Sub-sample shifts done using windowed sinc interpolation (Blackman window). FFT interpolation also possible if shifting impulse response.

Results – Digital CFD



mV → 0.5 1.7 5.2 16.5 52.3 165.3 523 1653

SNR ≥ 20 dB

Good match of model and data for 100 MHz ADC, slightly worse for 250 MHz ADC

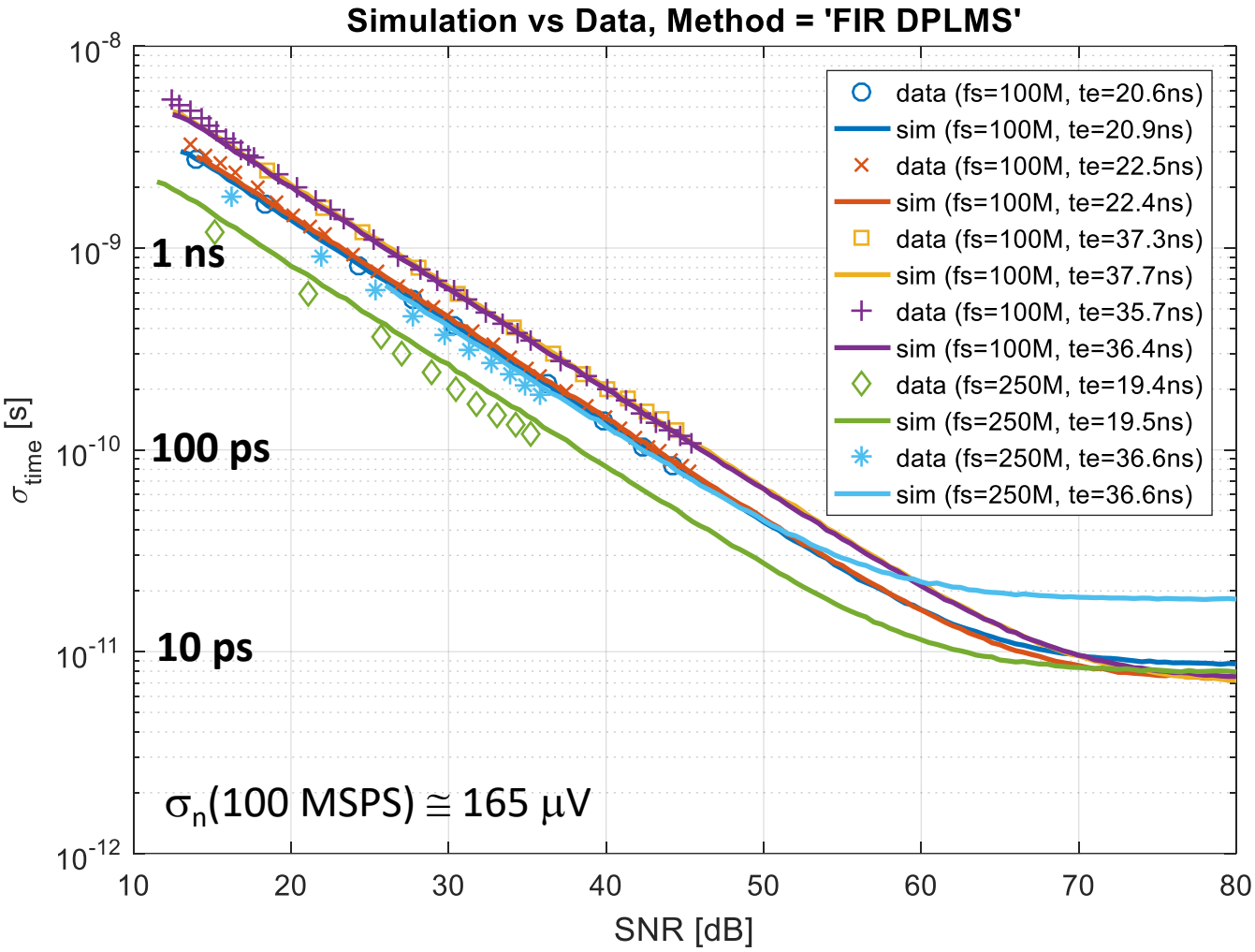
SNR < 20 dB

Poor match, data worse than model. Not a useful range anyway, as we need $\sigma_{time} < 1$ ns.

Timing resolution is proportional to

$$\frac{t_{rise}}{SNR}$$

Results – FIR DPLMS



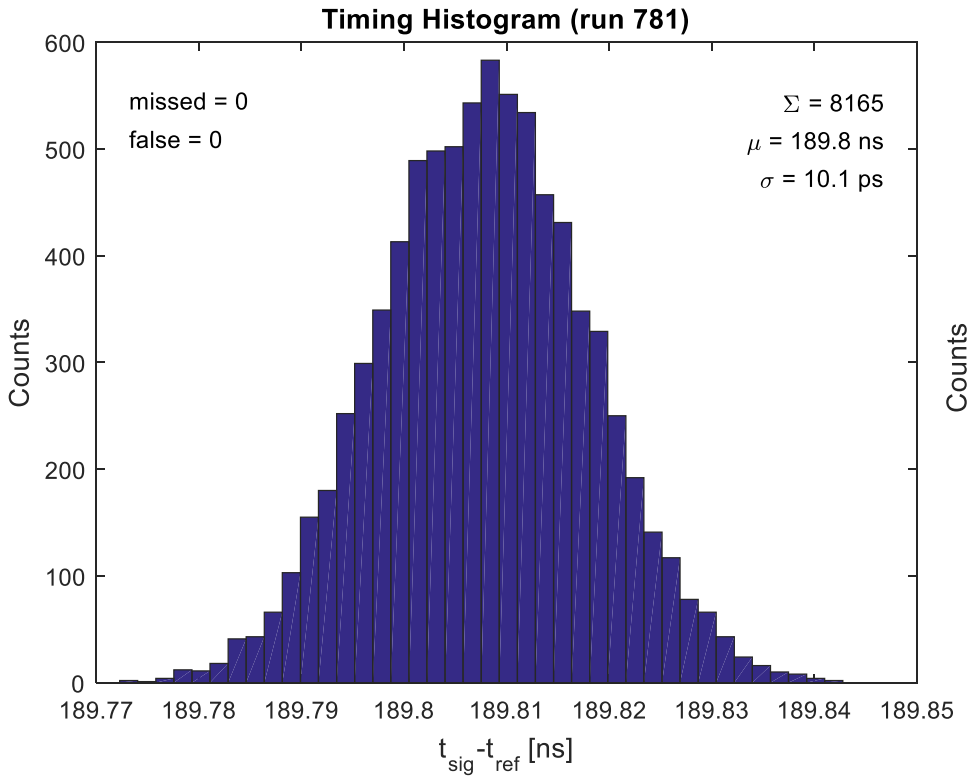
Good match of model and data for 100 MHz ADC, slightly worse for 250 MHz ADC

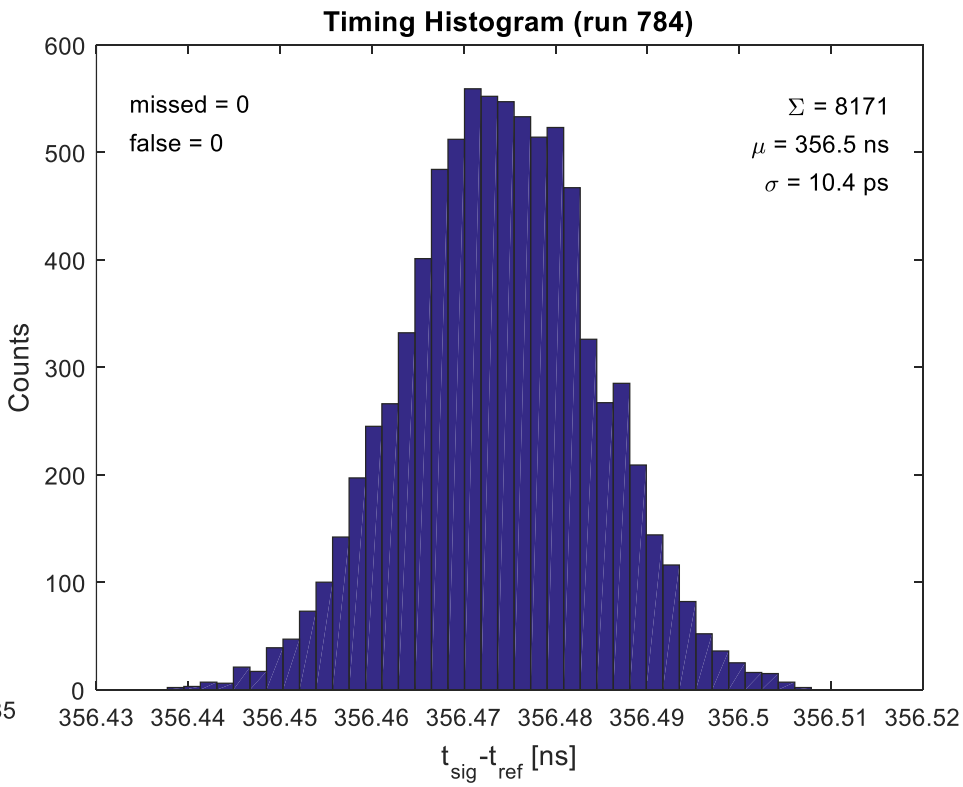
250 MHz data better than model – possibly due to some correlation which is not reflected by simulation.

mV → 0.5 1.7 5.2 16.5 52.3 165.3 523 1653

Example Histograms – FIR Timing

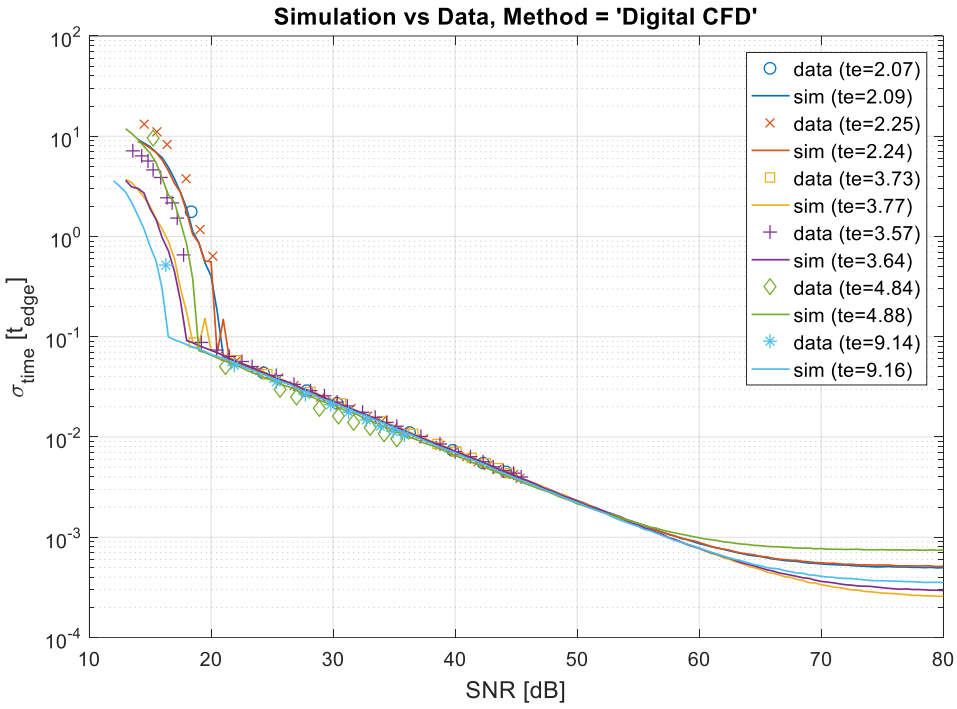
Large SNR case

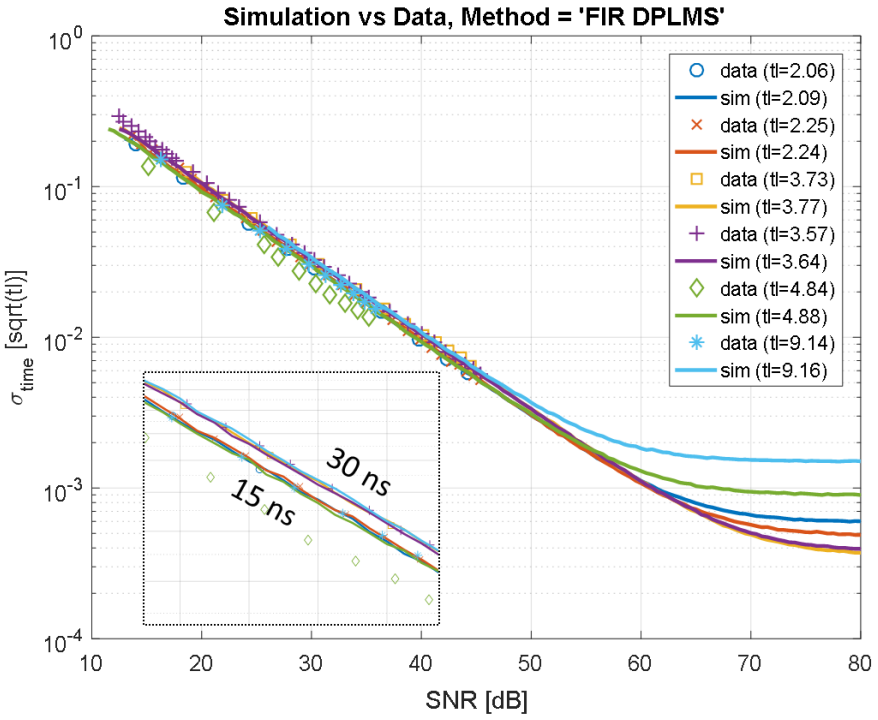




100 MSPS ADC, 14-bit, 15 ns shaper

Digital CFD / FIR DPLMS – Normalized



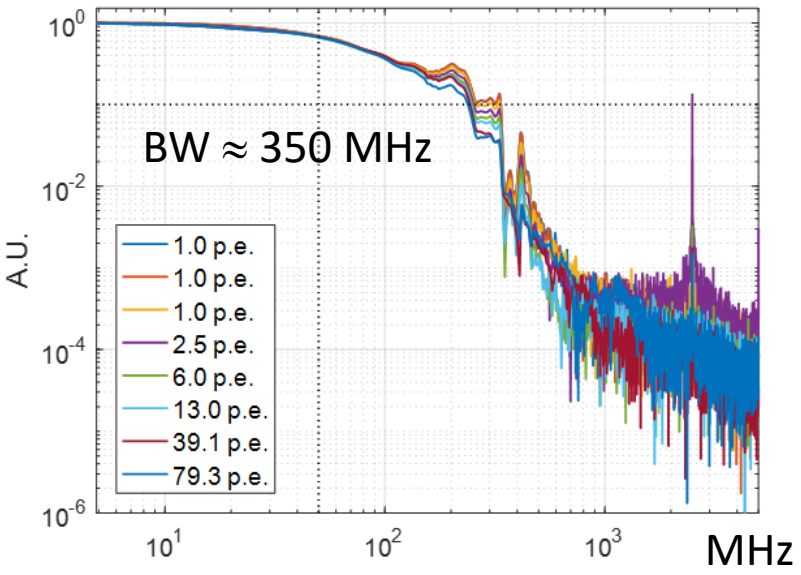


- Don’t need extremely high sampling rates to maintain good timing resolution, as long as SNR is sufficient
- It seems that it is better to maintain sharp edge → logical, as we don’t cut bandwidth of the signal that still has valid information
  - Sharp edges help in pile-up resolution
- Oversampling help only in case of FIR-based algorithms → SNR gets better

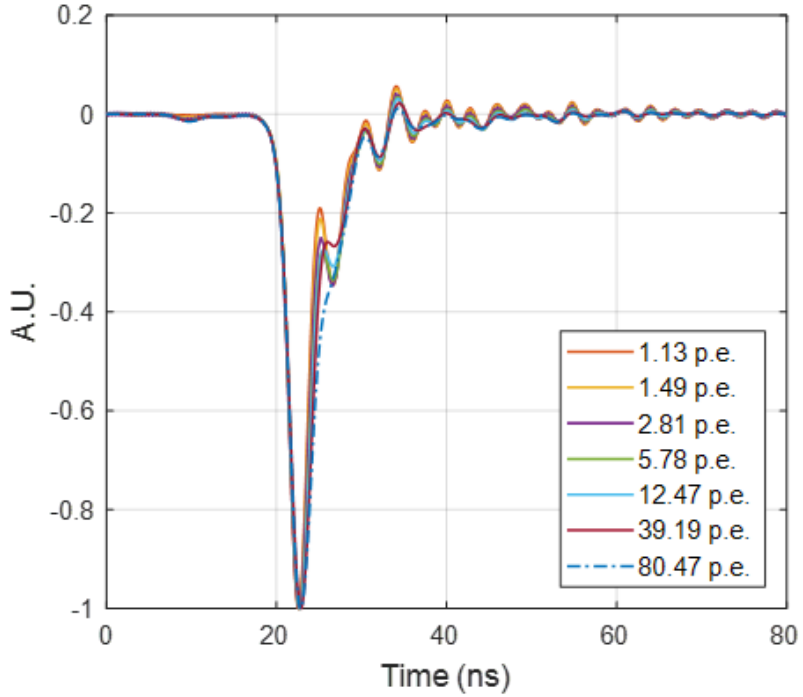
R14347 – Waveforms

- Visible dependence of waveform shape on position of the light source on the photocathode
- $t_{rise} \in$ (1.9 ns, 3.0 ns), FWHM $\in$ (3.0 ns, 4.7 ns); both increase with PE level (expected)

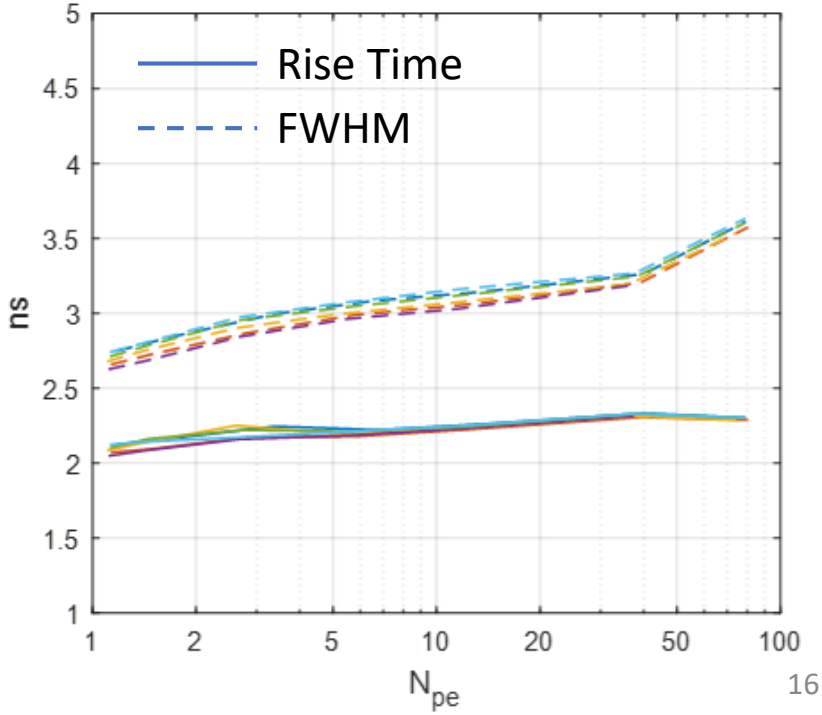
Normalized Amplitude Spectrum



Normalized templates

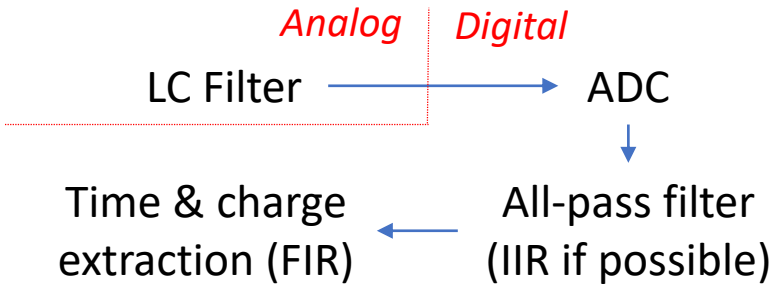


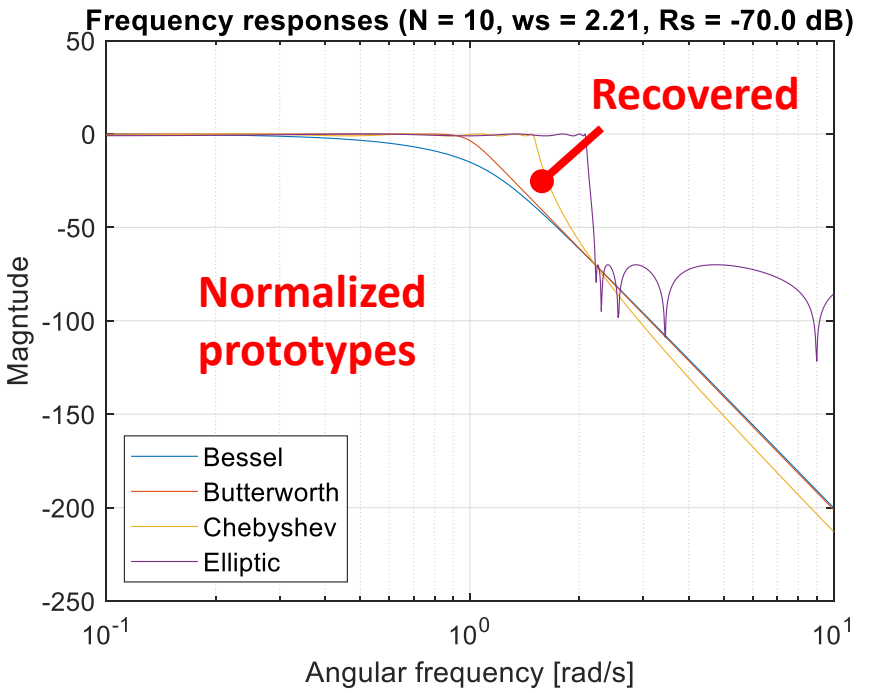
Rise Time & FWHM

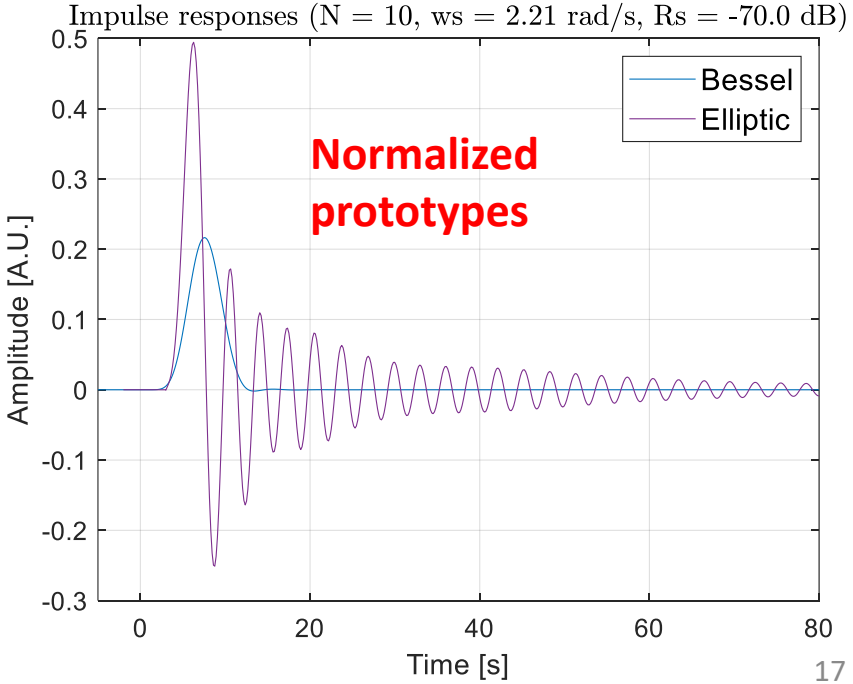


Where are we now?

- Re-designing the shaper
  - Old shaper used for tests was too noisy, had too low cutoff frequency
  - Decided to switch to fully passive design (LC-ladder)
  - Switch from Bessel to elliptic (hopefully)
- Need additional digital all-pass filter to correct passband ripple and phase





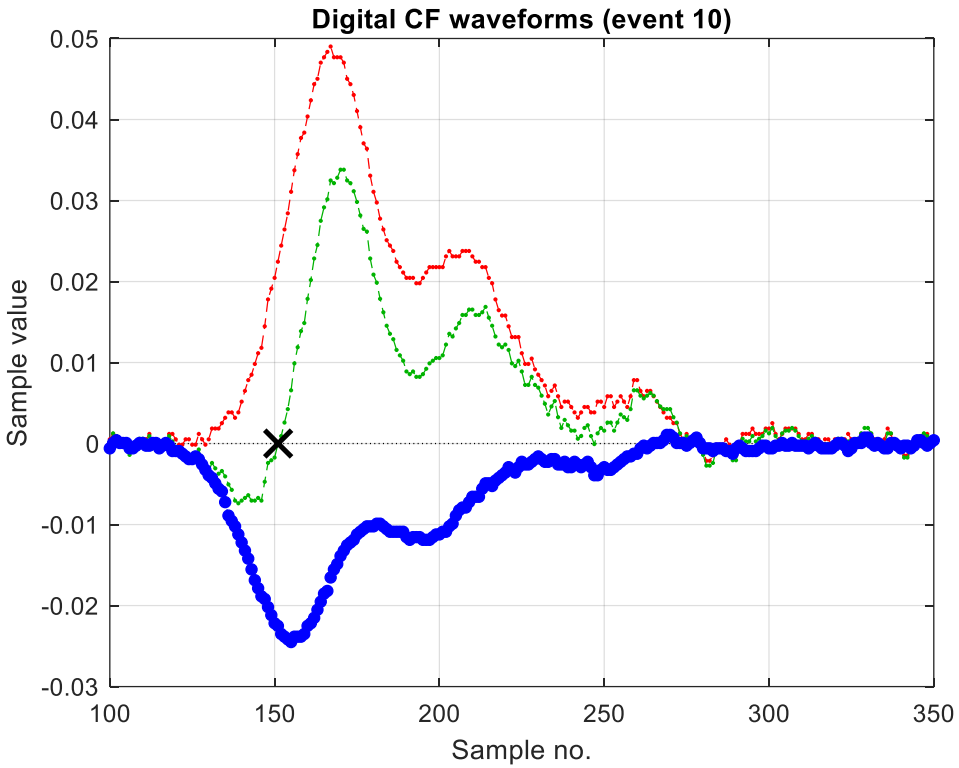


Conclusions

- Much work already done
- Pulse shape not guaranteed to be constant – need to deal with this for FIR-based methods
- Need to foresee that in FIR-based methods the estimate may be completely wrong in case of non-standard shape (for ex. pile-up)
  - Need quality factor for each time/charge estimate
  - Should send full waveform for off-line processing
- Need better shaper
  - Lower noise
  - Sharper rolloff (higher order)
  - Fully passive (LC-ladder)

Revised time estimation

- Digital CFD – limit shift to leading edge only
- For FIR-based method, need to parameterize impulse response of the filter wrt. charge

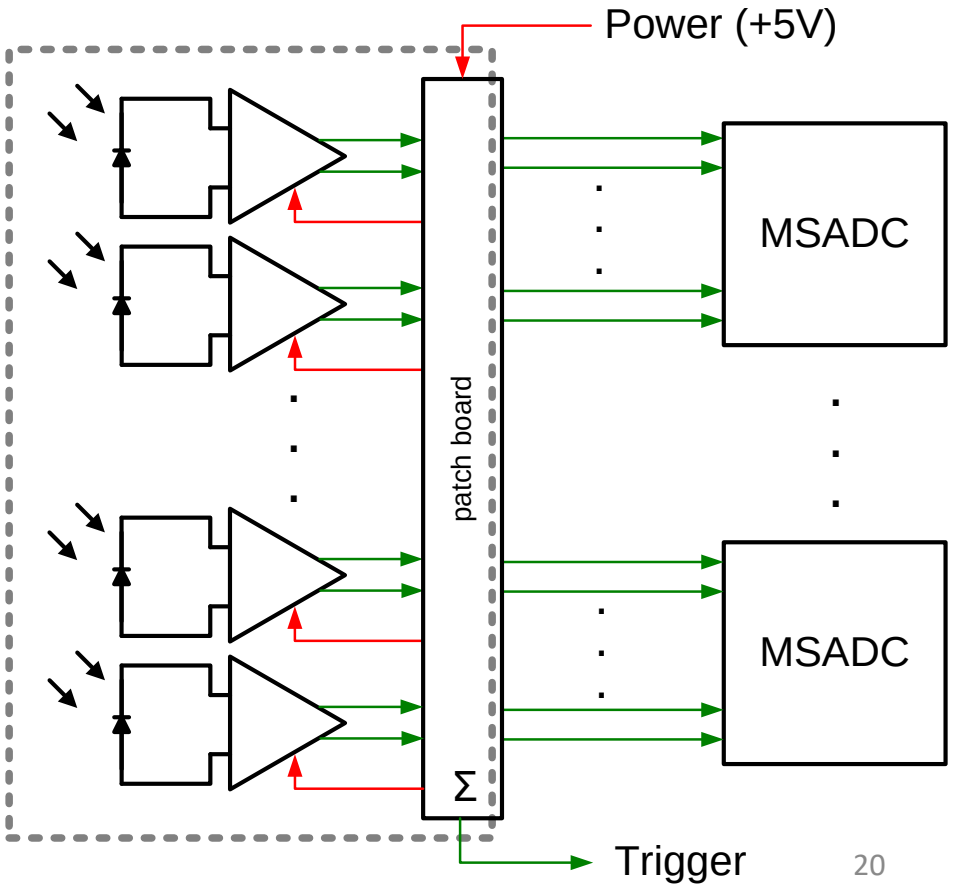


Significant increase in data rate – need efficient coding and possibly lossy waveform compression (already working on this)

BACKUP

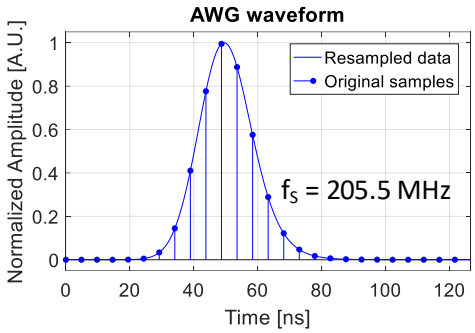
ECAL0 – Digitizer & Acquisition

- DAQ via MSADC cards, 80 MSPS, 12 bits, 64 channels/board
- 9-channel pre-amplifier for single Shashlyk module
- On-detector signal amplification & shaping
  (3 poles, 1 pole-zero,
  40 ns peaking time)
- Differential signal output (analog)
- Sum output for analog trigger



System Model (each channel)

Used 250 MHz data to determine actual AWG f_S



$$\sigma_{final} = \sqrt{{\sigma_{ref}}^2 + {\sigma_{sig}}^2}$$

sqrt(noise periodogram)

$\mathcal{F}^{-1}$

White noise

*

Semi-analog simulation, T_S=1 ps

AWG pulse

Anti-aliasing filter

Shaper

Sampling

\+

Fit

Digital CFD

FIR zero-cross

FIR matched

\-

Error

Random sub-sample shift

Waveform samples: T. Lindner/M. Walters Analog bandwidth of AWG is 80 MHz.

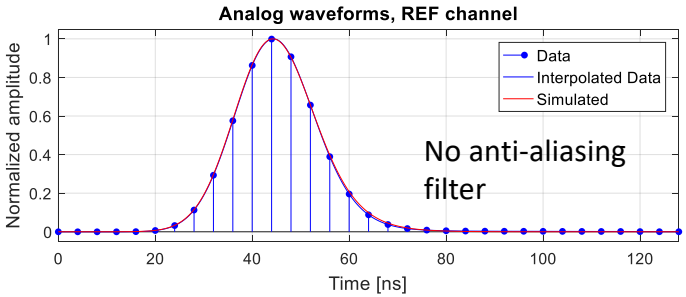
Depends on ADC. 2 real poles for DT5724 Not needed for V1720

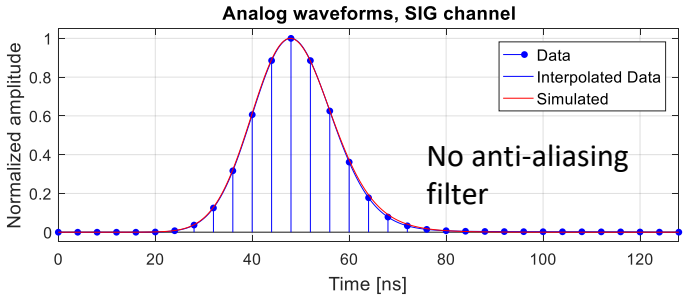
5-th order Bessel filter and CR differentiator. Not used for reference channels.

All transfer functions (*TF*) calculated in s-domain, then used $\mathcal{L}^{-1}$ to calculate impulse response.

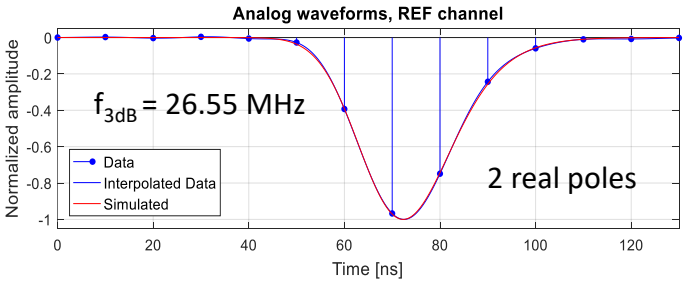
Signal Models

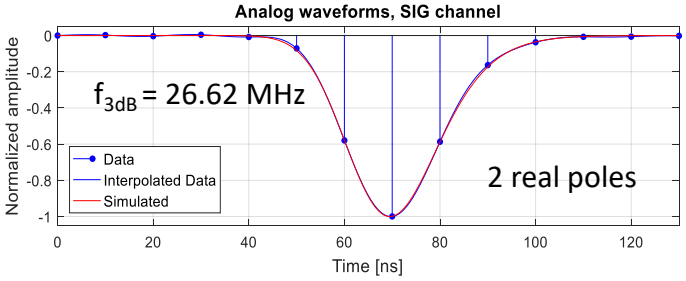
250 MHz,
CH1 = ref, CH2 = ref



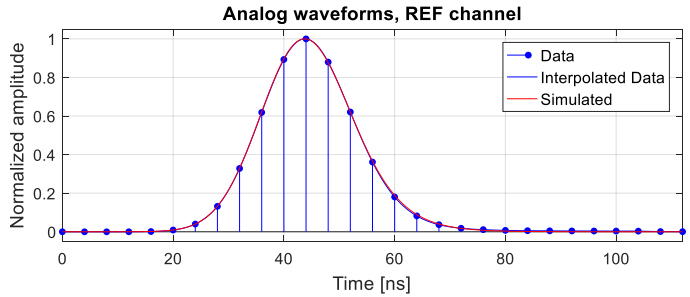


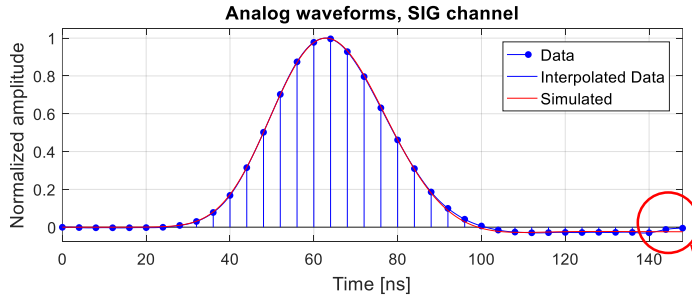
100 MHz,
CH1 = ref, CH2 = ref



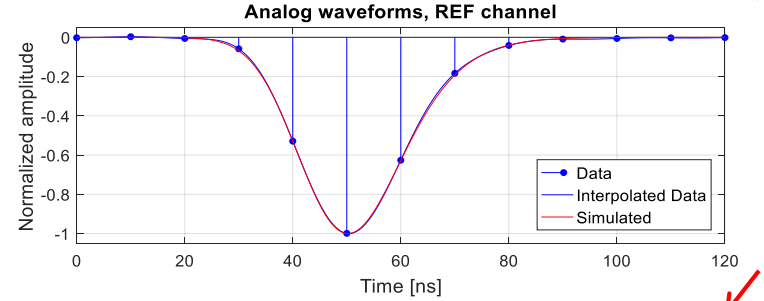


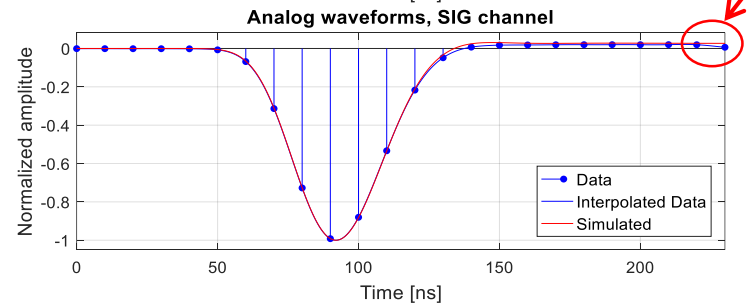
250 MHz,
CH1 = ref, CH2 = sig (15 ns)





CH1 = ref,
CH2 = sig (15 ns low power)

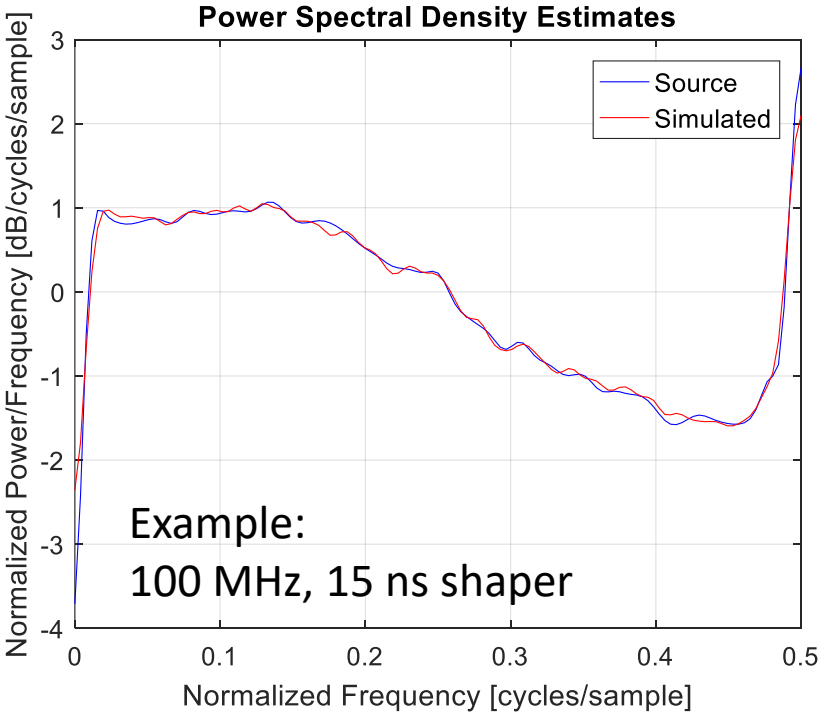


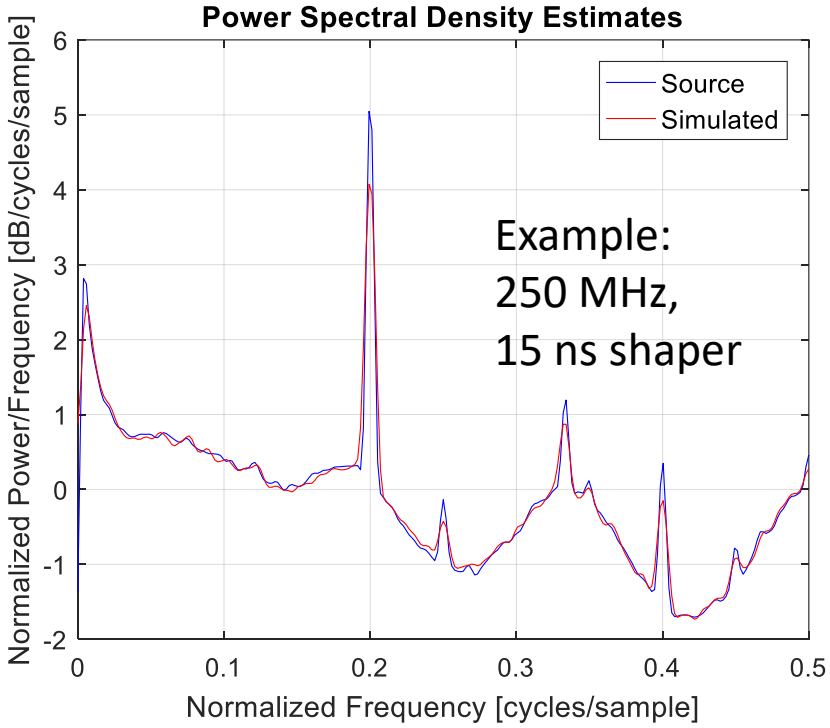


All pulses matched by FWHM

Interpolation artefacts

Noise models

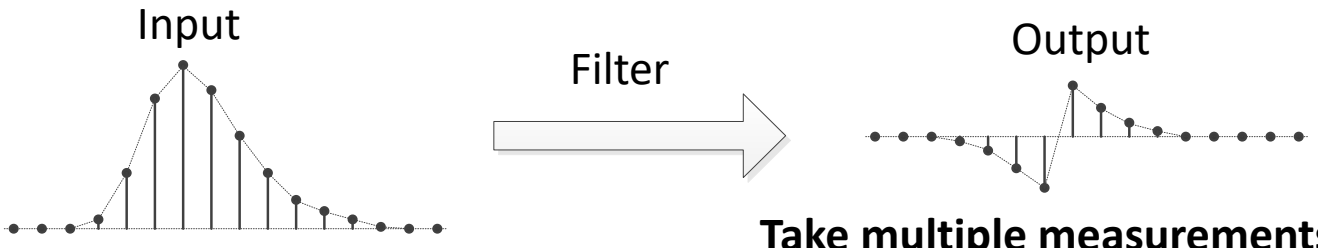




- Good match of simulated periodogram with an experimental one.
- Potential problem:
  - Some of the deterministic components (peaks in spectrum) do not have random phase, but are correlated to sampling clock.

Synthesizing FIR filter – Method 1

Digital Penalized LMS Method



Take multiple measurements, then:

Minimize overall variance of the response:

input signal *noiseless signal (our template)* *stationary noise*

$$x[n] = x'[n] + x''[n]$$

Sought filter

$$Var(y) = \boldsymbol{h}^{1,N} \cdot \boldsymbol{R}^{N,N} \cdot \boldsymbol{h}^{N,1}$$

Noise auto-covariance matrix

number of filter taps *impulse response of the filter*

Filter is **linear**, so the output signal is:

$$y[n] = \sum_{l=0}^{N-1} h[l] \cdot x'[n-l] + \sum_{l=0}^{N-1} h[l] \cdot x''[n-l]$$

Therefore, we can deal with noise and signal components separately

Minimize difference between filter response and our desired response

$$(E(y[k] - v_k))^2 = (\boldsymbol{h}^{1,N} \cdot \boldsymbol{x}'(k)^{N,1} - v_k)^2$$

Value of k-th sample of the response to x'

N past samples of x', starting from k

Gatti E., et al., "*Digital Penalized LMS method for filter synthesis with arbitrary constraints and noise*", NIM A523, 167-185, 2004

Synthesizing FIR filter – Method 1 (cont.)

Digital Penalized LMS Method

Add additional constraints for frequency response, including gain at DC …

Add constraints related to bit-gain (i.e. how well we are supposed to reject quantization noise) …

Finally, build the error functional and minimize it:

$$Area(FIR) = \frac{Area(y)}{Area(x)}$$

Constraint for variance

Constraints for shape of response to pulse template

Frequency constraints

$$\varepsilon^2 = Var(y) + \sum_{k=1}^{N} \alpha_k \cdot (E(y[k]) - v_k)^2 + \sum_{l=1}^{L} \beta_l \cdot \left(|F\{\boldsymbol{h}\}|_{\omega_l}\right)^2 + \varphi \cdot (F\{\boldsymbol{h}\}_{\omega=0} - Area(FIR))^2 + \gamma \cdot \sum_{n} (h[n])^2$$

DC gain (i.e. area) constraint

Bit-gain constraint

All components are square functions, so there exists a global minimum – just need to properly choose $\boldsymbol{N}$, $\vec{\boldsymbol{v}}$, $\vec{\alpha}$, $\vec{\beta}$, φ and γ → papers don't say much about that