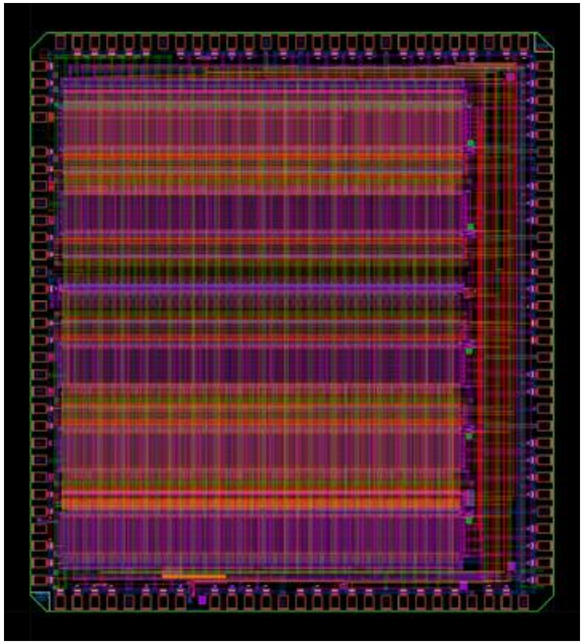
Data Processing, Calibration, and Archiving

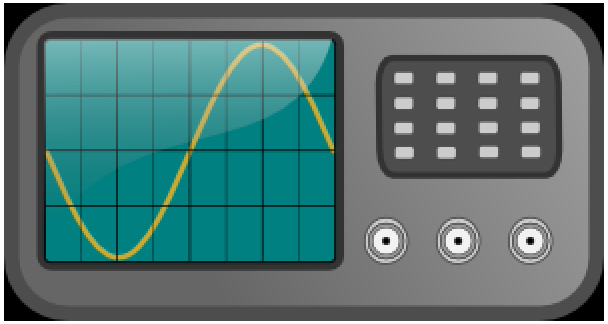
Kurtis Nishimura
University of Hawaii
LAPPD Electronics Godparent Review
May 20, 2011

Oscilloscope on a chip!

- Waveform sampling/digitizing ASICs (like PSEC3/4) are just like an oscilloscope!



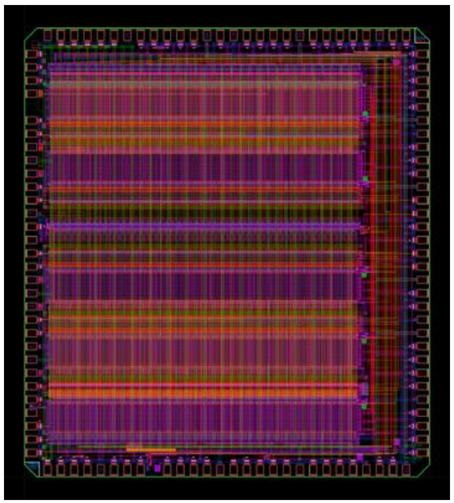
=



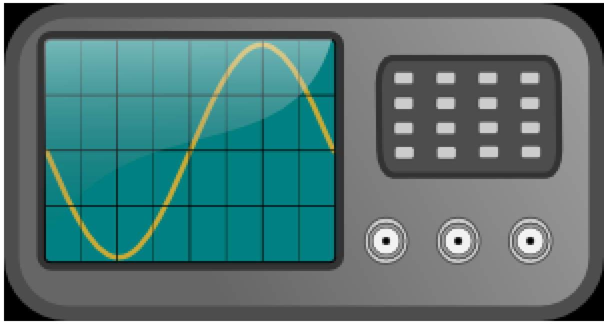
...right?

Oscilloscope on a chip?

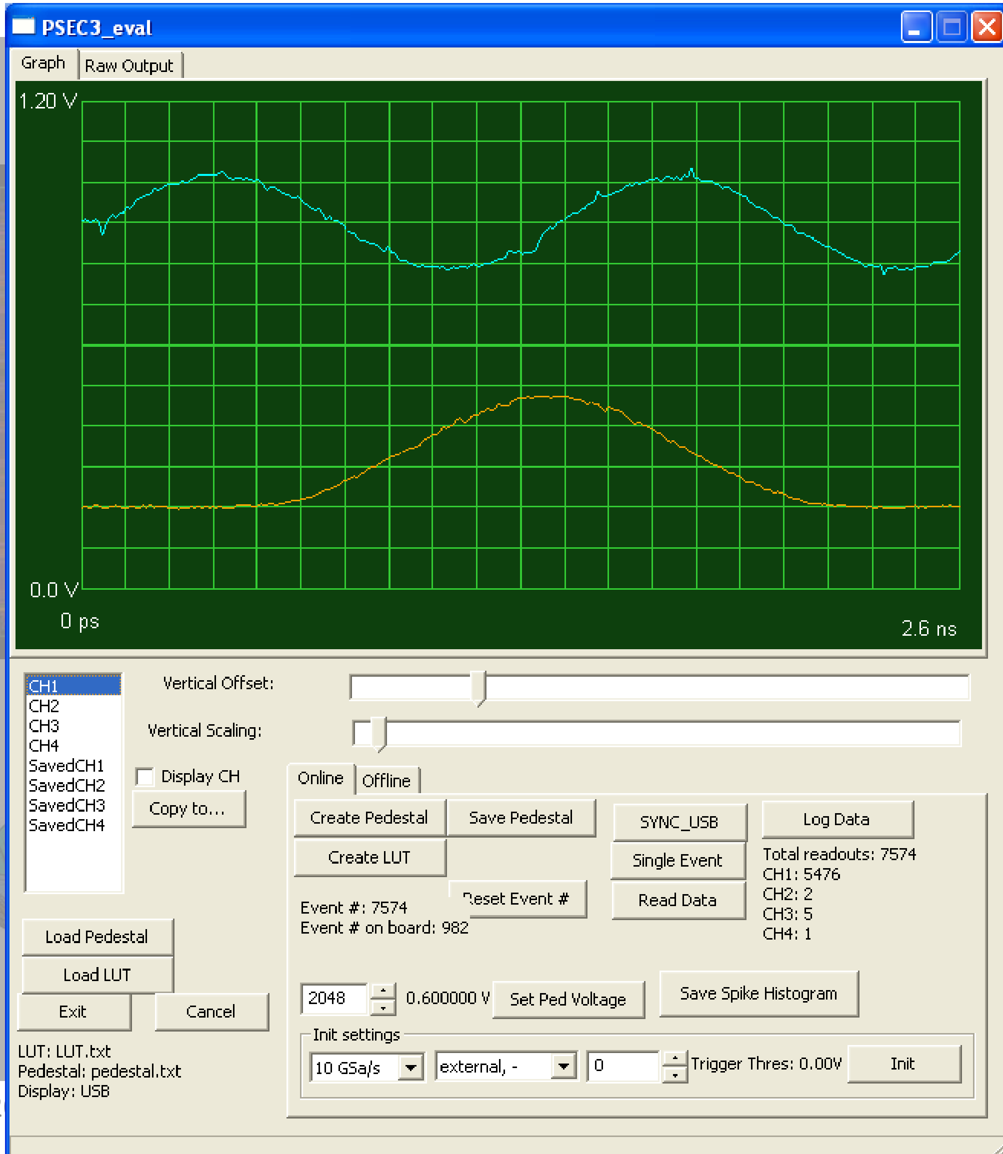
- Modified approximation:

 +  +  + 

+  +  + 

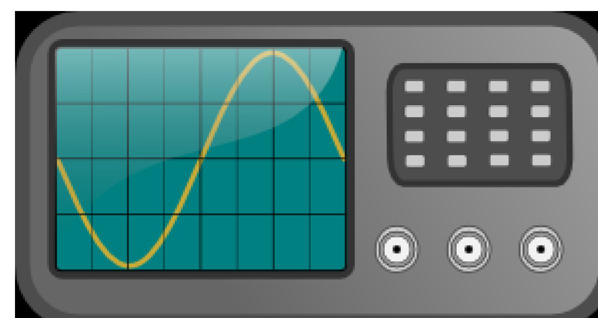
+ … ≈ 

Oscilloscope on a chip?



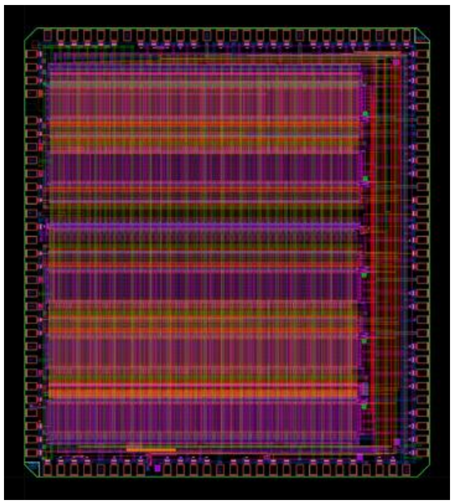
Screenshot from the PSEC3_eval GUI – written by Andrew Wong

≈



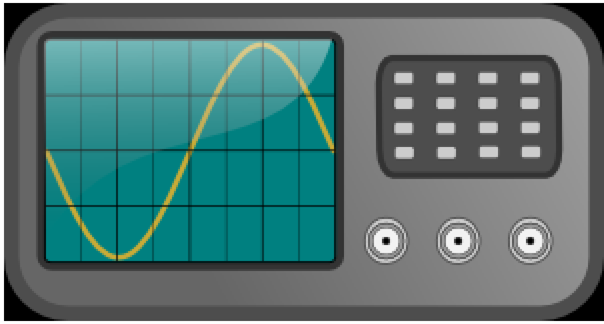
Oscilloscope on a chip?

- Modified approximation:


+  +  + 

Will focus on the data processing & calibration part for this talk…

+  +  + 

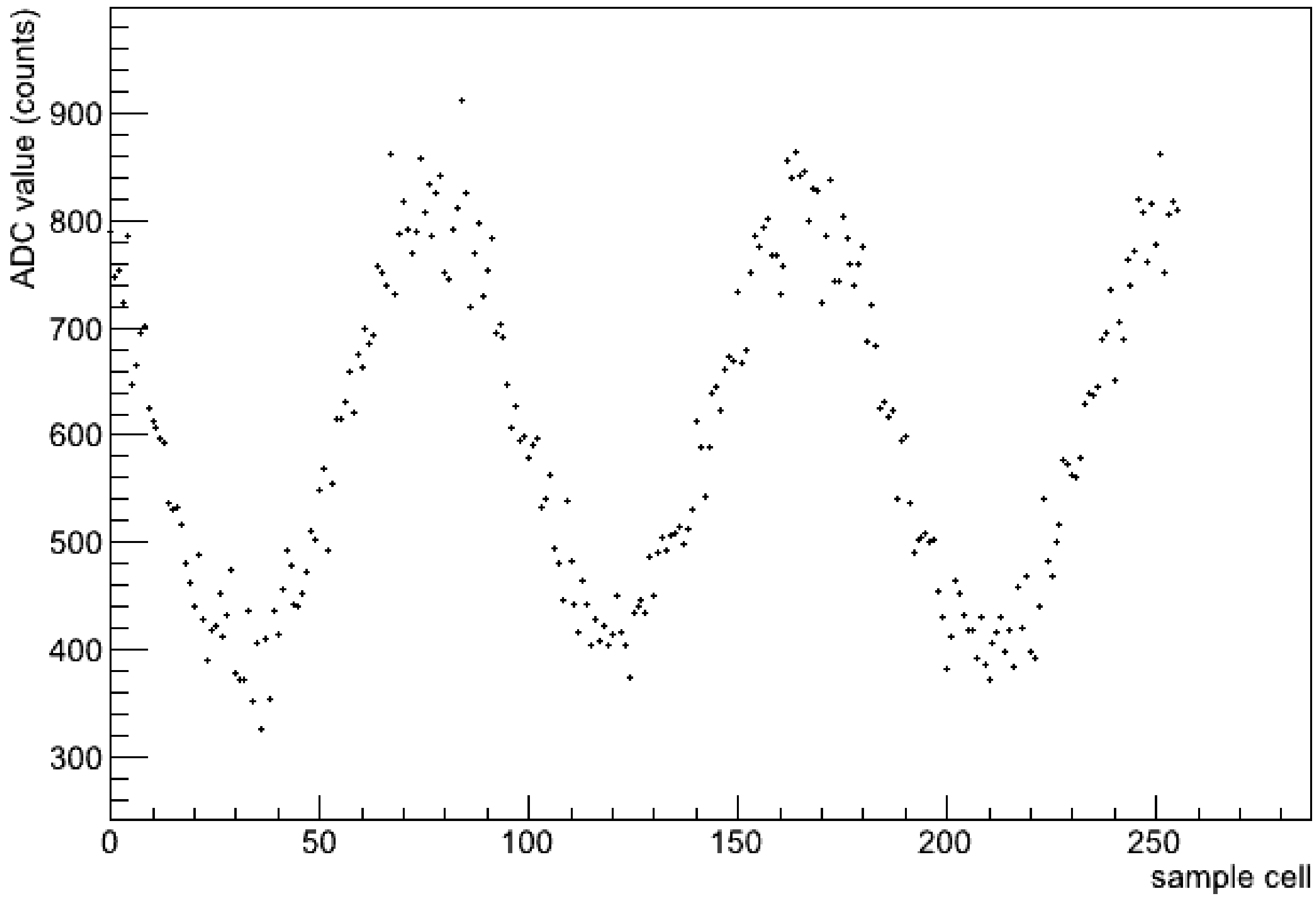
+ … ≈ 

What corrections are required?

1. Voltage conversion:
   - Convert ADC counts to voltage.
2. Pedestal correction:
   - Remove cell-to-cell fixed DC patterns.
3. Time base correction:
   - Keep overall sampling rate constant (or correct for drift).
   - Correct for cell-to-cell variations in sampling rate.

***Some much simpler than others, and not all are required for every chip or application… some extra may be required for some applications.**

A raw PSEC3 readout…



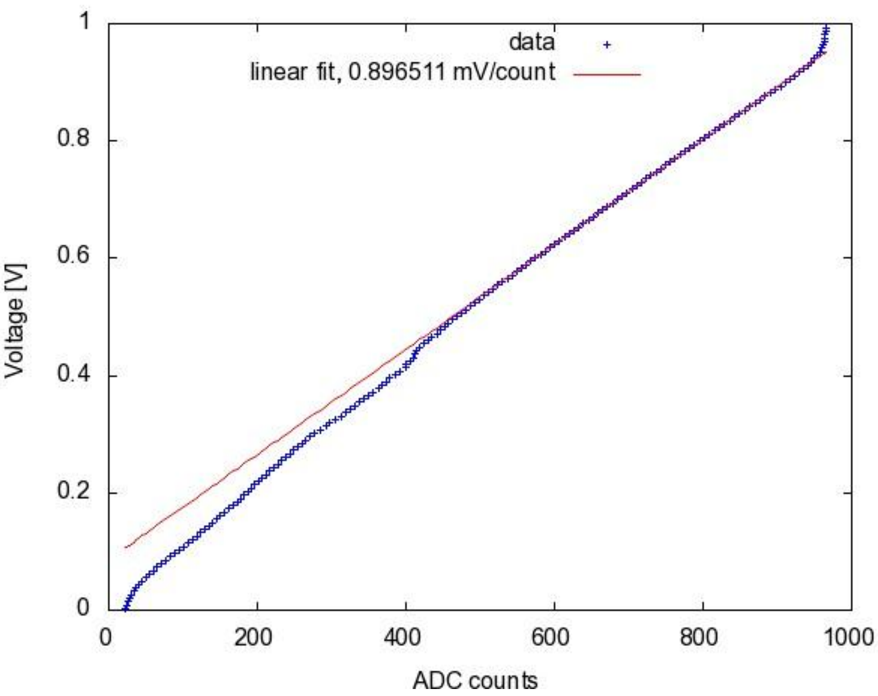
- PSEC3 sampling at ~10 GSa/s.
- 120 MHz, 150 mV_{rms} sine wave input from signal generator.

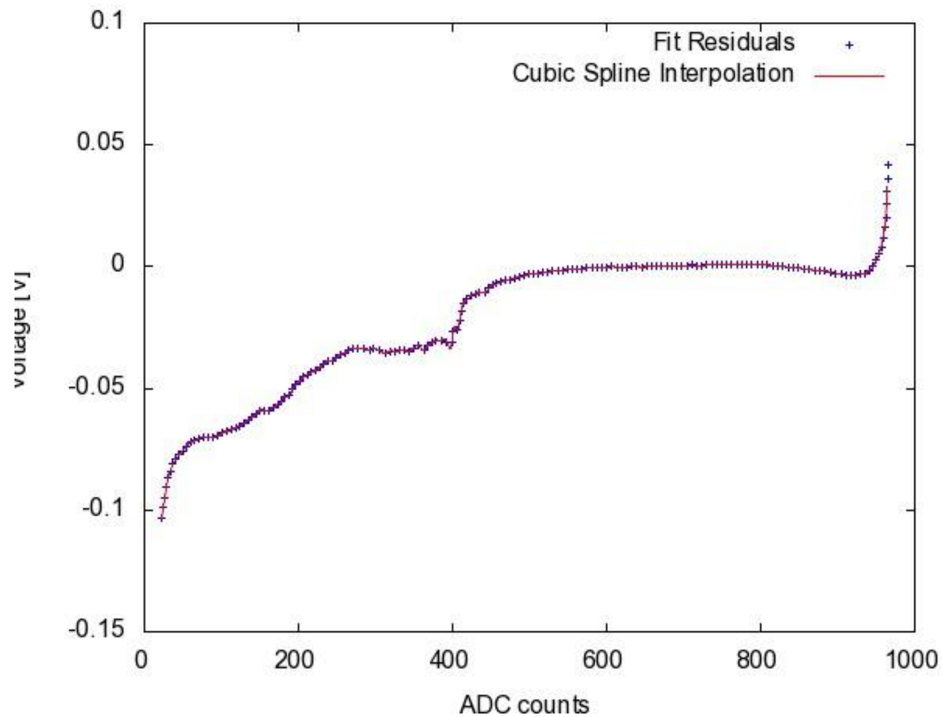
What corrections are required?

1. **Voltage conversion:**
   – **Convert ADC counts to voltage.**
2. Pedestal correction:
   – Remove cell-to-cell fixed DC patterns.
3. Time base correction:
   – Keep overall sampling rate constant (or correct for drift).
   – Correct for cell-to-cell variations in sampling rate.

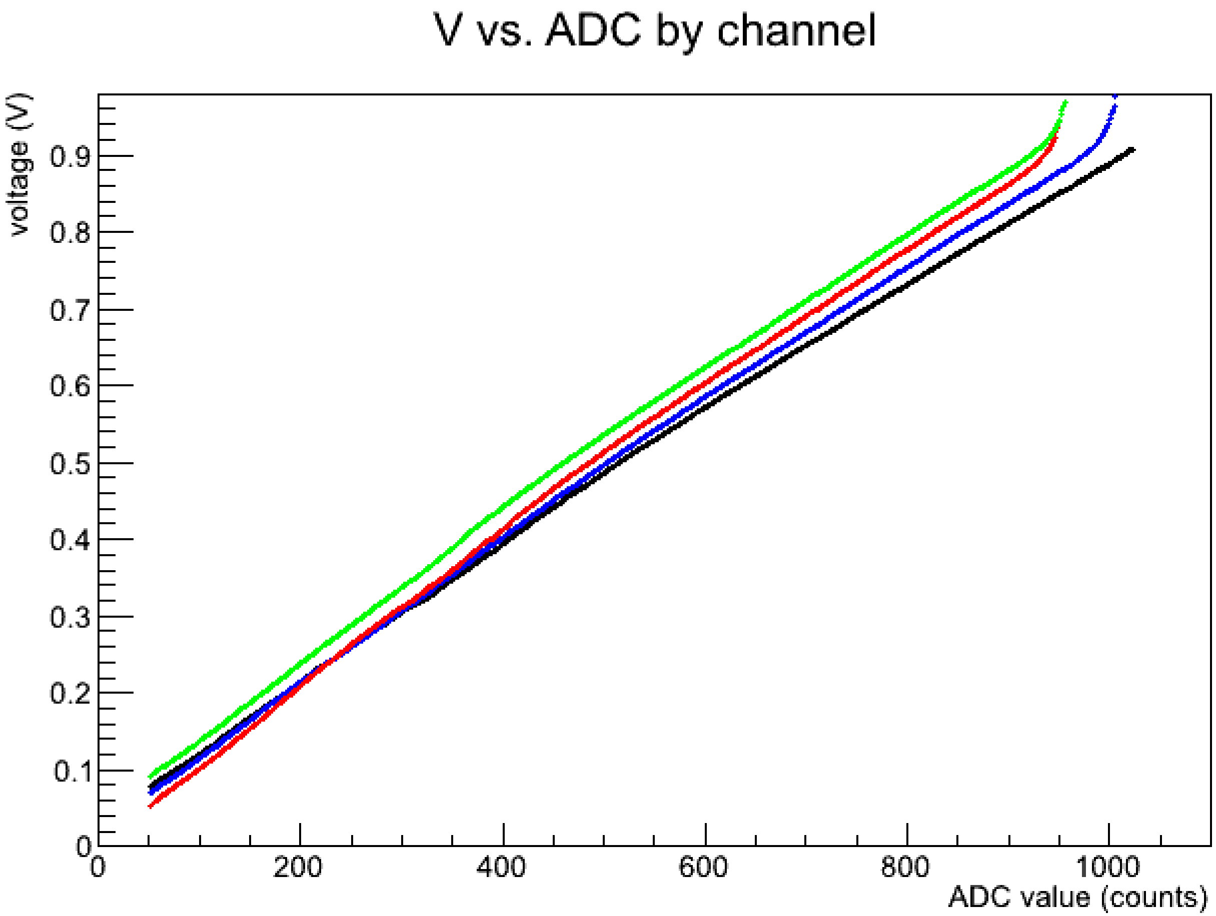
Voltage Calibration

- Straightforward procedure:
  - Scan the input voltage w/ DAC; check ADC output.
  - Eric has shown for PSEC3 already:



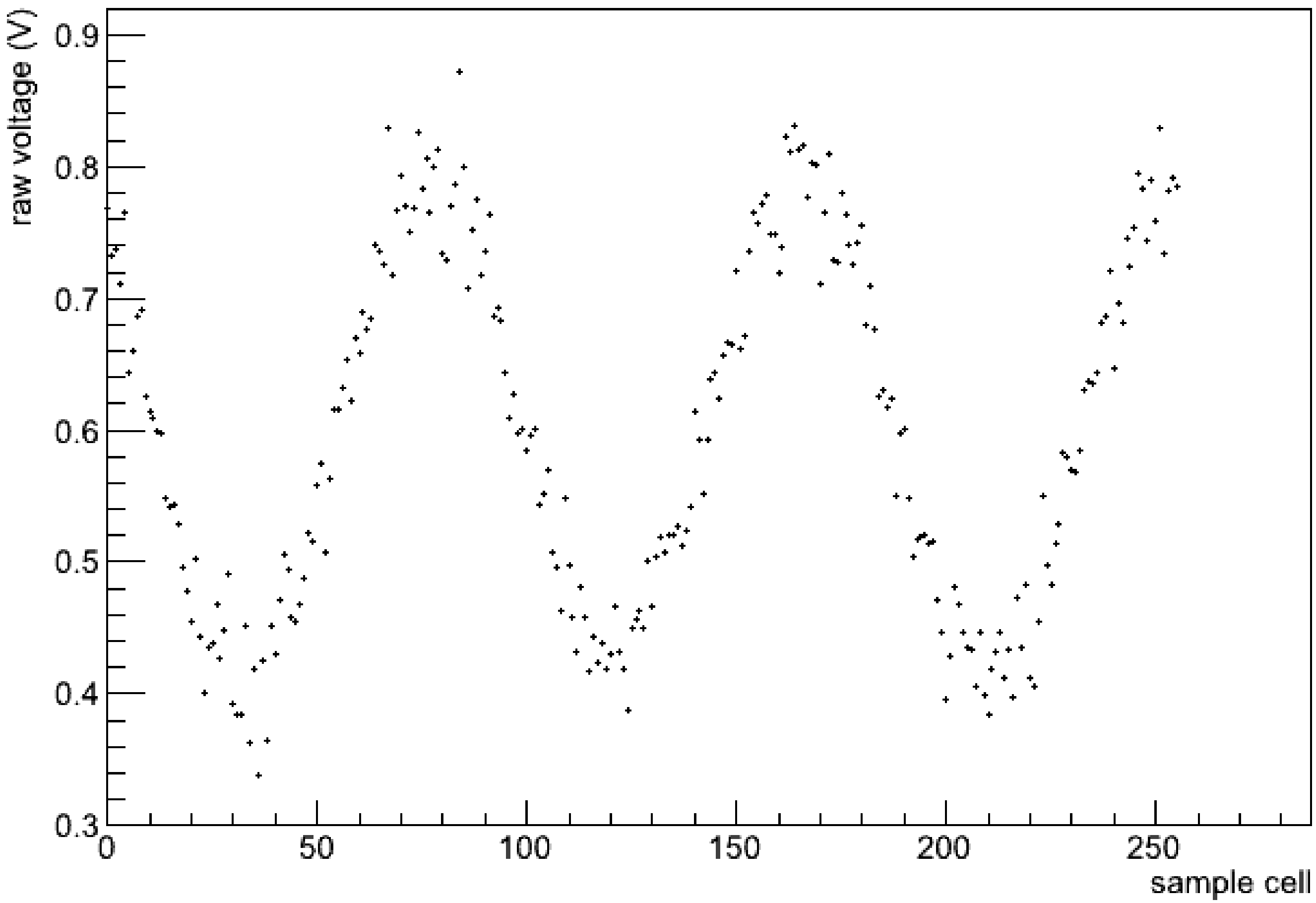


Voltage Calibration



- In readout software, implemented as lookup tables.
  - Independent for each channel to allow for variations.

PSEC3 Readout… +Voltage Calibration



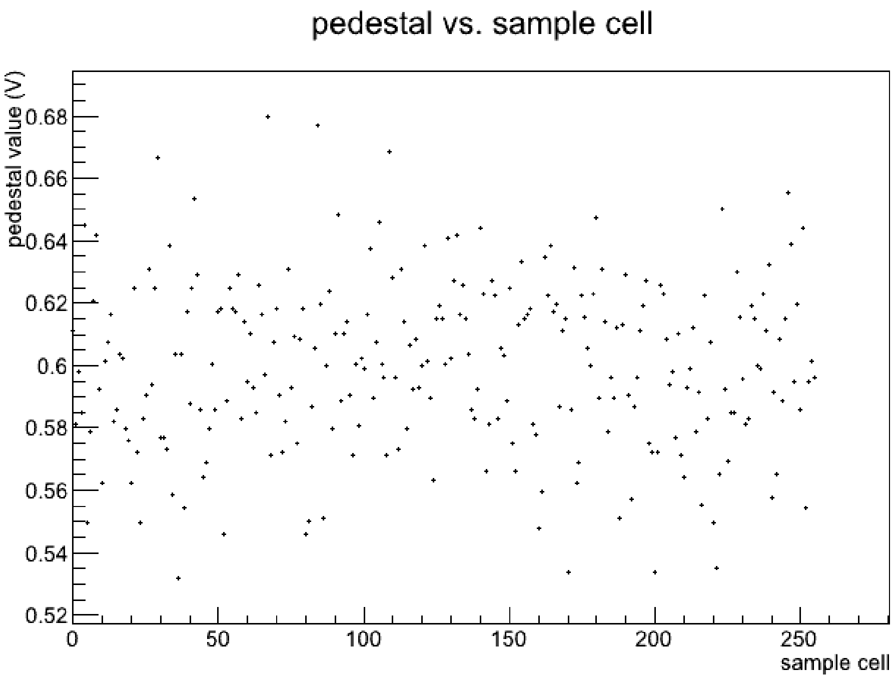
- PSEC3 sampling at ~10 GSa/s.
- 120 MHz, 150 mV_{rms} sine wave input from signal generator.

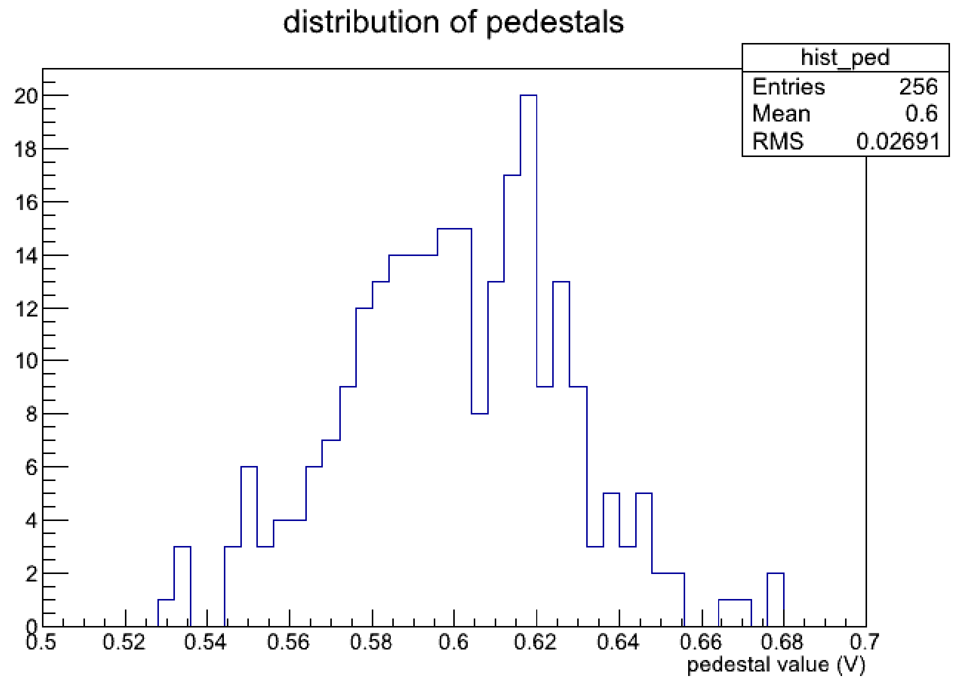
What corrections are required?

1. Voltage conversion:
   – Convert ADC counts to voltage.
2. **Pedestal correction:**
   – **Remove cell-to-cell fixed DC patterns.**
3. Time base correction:
   – Keep overall sampling rate constant (or correct for drift).
   – Correct for cell-to-cell variations in sampling rate.

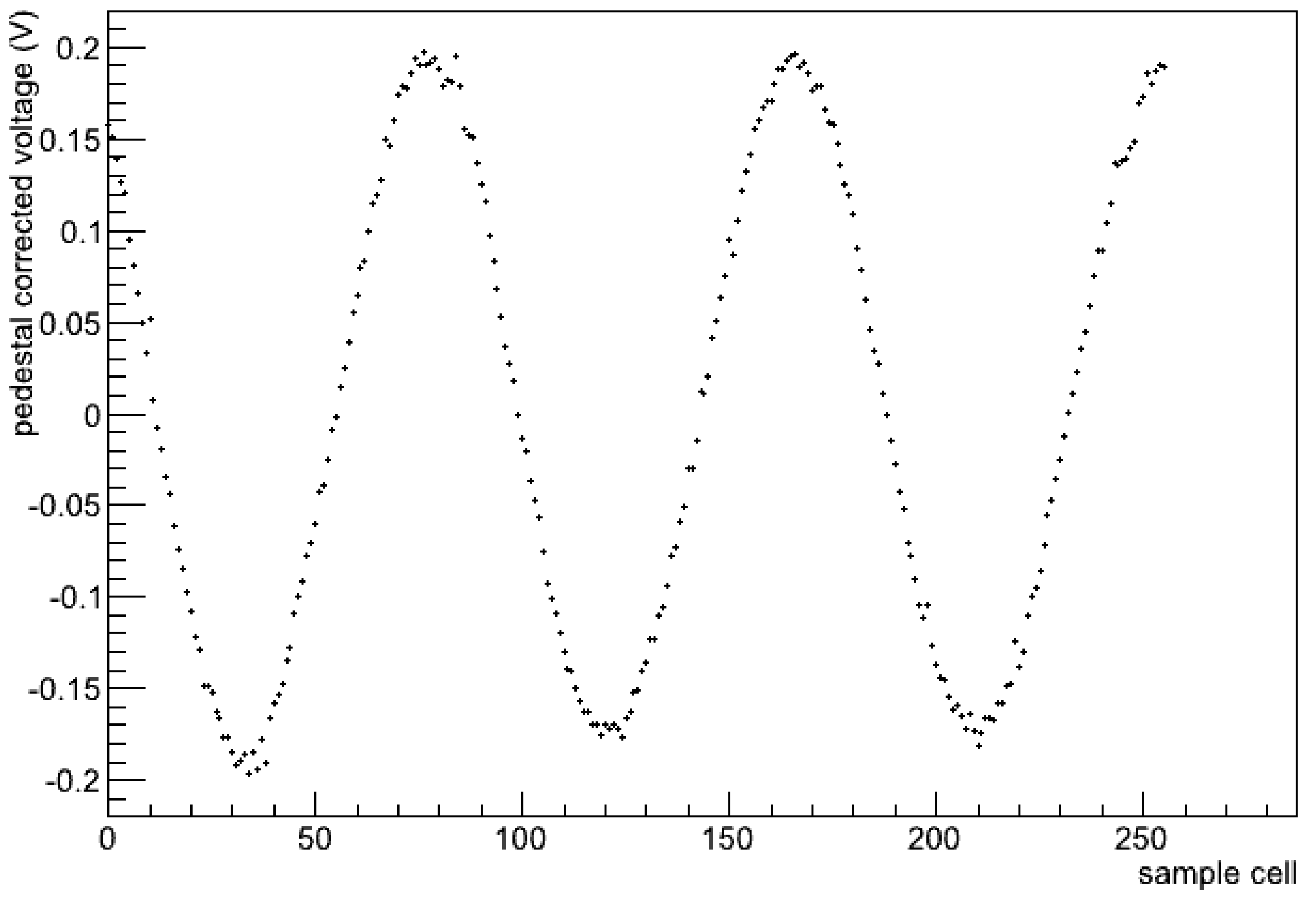
Pedestal Correction

- Another straightforward procedure…
  - Read out values of samples with no input.
  - Determine average voltage.
  - This is the “pedestal” to subtract from future readouts.





PSEC3 Readout… + Pedestal Subtraction



➔ This is the current limit of the calibrations in the PSEC3_eval readout software.

What corrections are required?

1. Voltage conversion:
   – Convert ADC counts to voltage.
2. Pedestal correction:
   – Remove cell-to-cell fixed DC patterns.
3. **Time base correction:**
   – **Keep overall sampling rate constant (or correct for drift).**
   – **Correct for cell-to-cell variations in sampling rate.**

*Whereas 1 and 2 are well understood and already implemented in readout software, this last piece represents work in progress…

Time Base Corrections

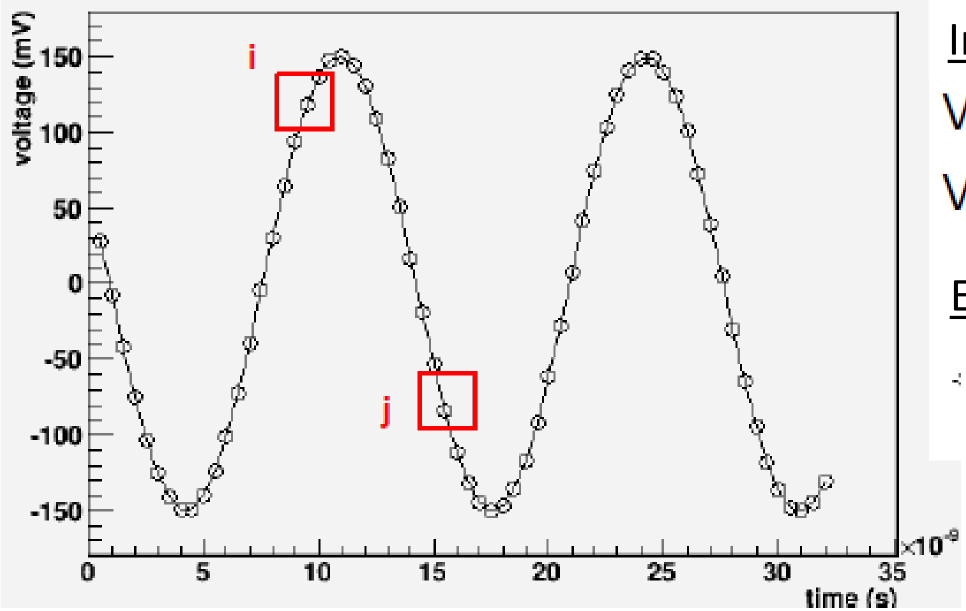
- Overall time base drift:
  - ☑ **On PSEC3, overall sampling rate is locked by DLL.**
- Still needed:
  - ❑ Measurement of the sampling rate.
  - ❑ Calibration of individual time delays between sample cells ($\Delta t_{i,i+1}$).

- Sine wave inputs of known frequency are a powerful tool.

  ➔ We've developed a correlation-based technique for timing calibration…

Timing Calibration w/ Correlations

- Plot correlations between pairs of samples:
  - To determine Δt_{ij}, plot $V_i - V_j$ versus $V_i + V_j$



Input signals given by:

$V_i = A \sin(\omega t_i + \phi)$

$V_j = A \sin(\omega t_j + \phi)$

Effectively rotate by 45°:

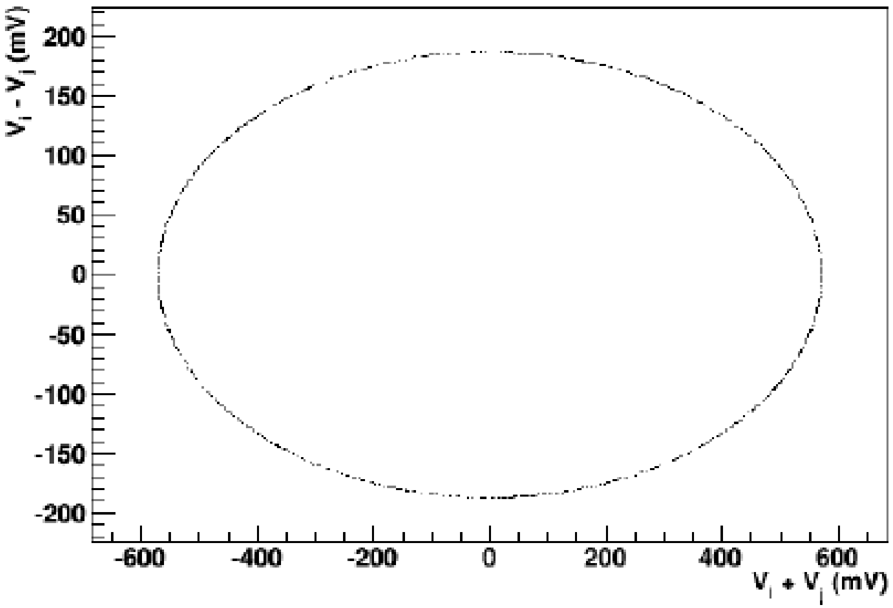
- $x := V_i + V_j$
- $y := V_i - V_j$

$$\rightarrow \frac{x^2}{4A^2\cos^2(\omega\delta t/2)} + \frac{y^2}{4A^2\sin^2(\omega\delta t/2)} = 1$$

- **i and j can be adjacent (or not), but cycle ambiguities exist if > 1 period apart.**

Timing Calibration w/ Correlations

- Plot correlations between pairs of samples:
  - To determine Δt_{ij}, plot $V_i - V_j$ versus $V_i + V_j$



<u>Input signals given by</u>:

$V_i = A \sin(\omega t_i + \phi)$

$V_j = A \sin(\omega t_j + \phi)$

<u>Effectively rotate by 45°</u>:

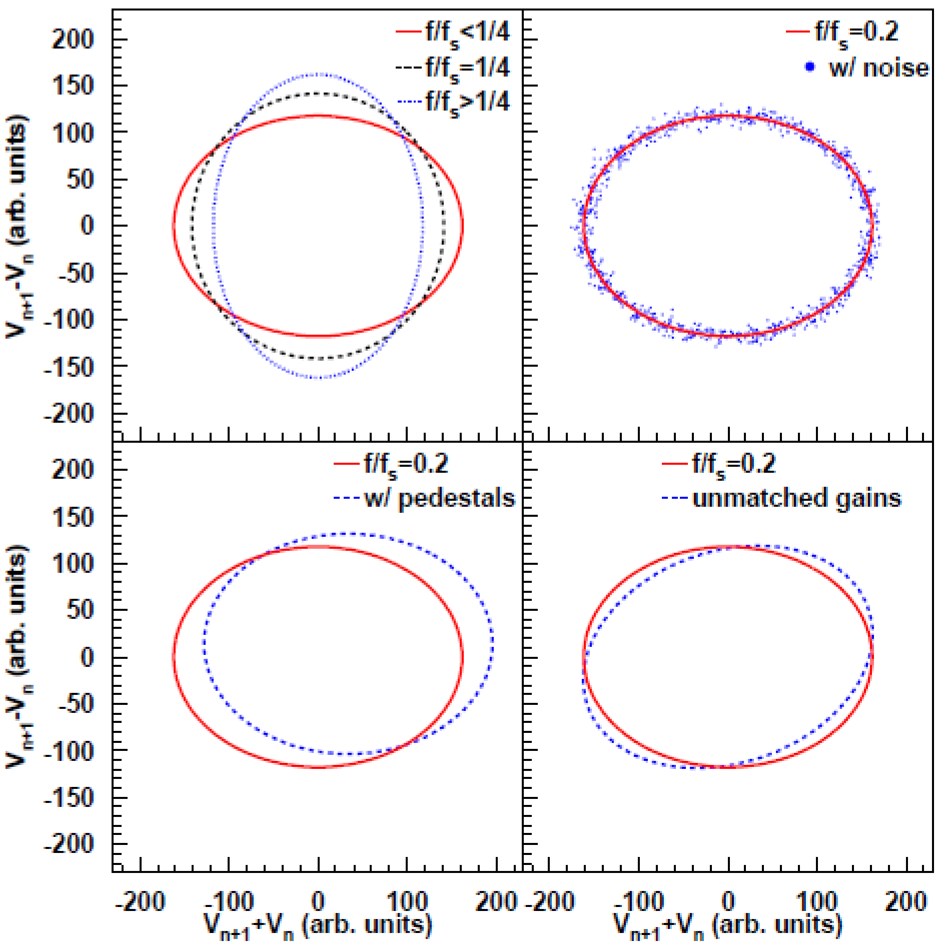
- $x := V_i + V_j$
- $y := V_i - V_j$

➔ $$\frac{x^2}{4A^2\cos^2(\omega\delta t/2)} + \frac{y^2}{4A^2\sin^2(\omega\delta t/2)} = 1$$

- **i and j can be adjacent (or not), but cycle ambiguities exist if > 1 period apart.**

Timing Calibration w/ Correlations

- Ellipse features:



1) Different Δt (for known sampling frequency) give different major/minor radii.
2) Noise makes ellipse "fuzzy"
3) Nonzero pedestals shift origin
4) Difference in gain between two cells causes a rotation.

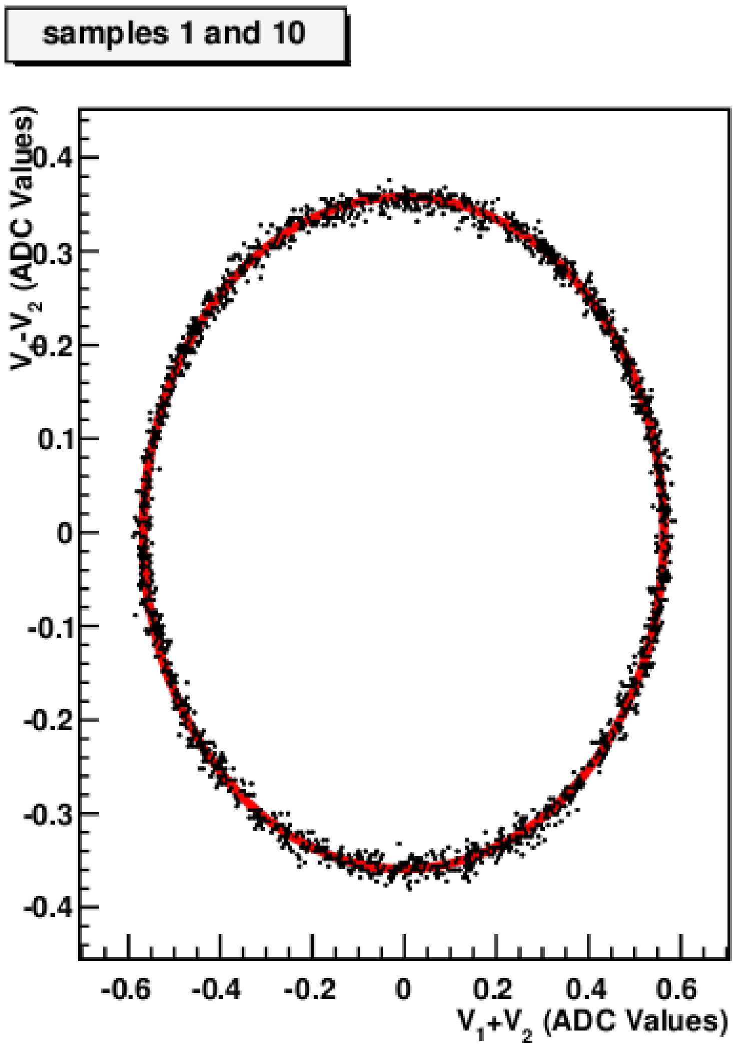
➔ We have written an ellipse fitter to perform this method.

➔ Even without fitting, it provides nice qualitative check on results.

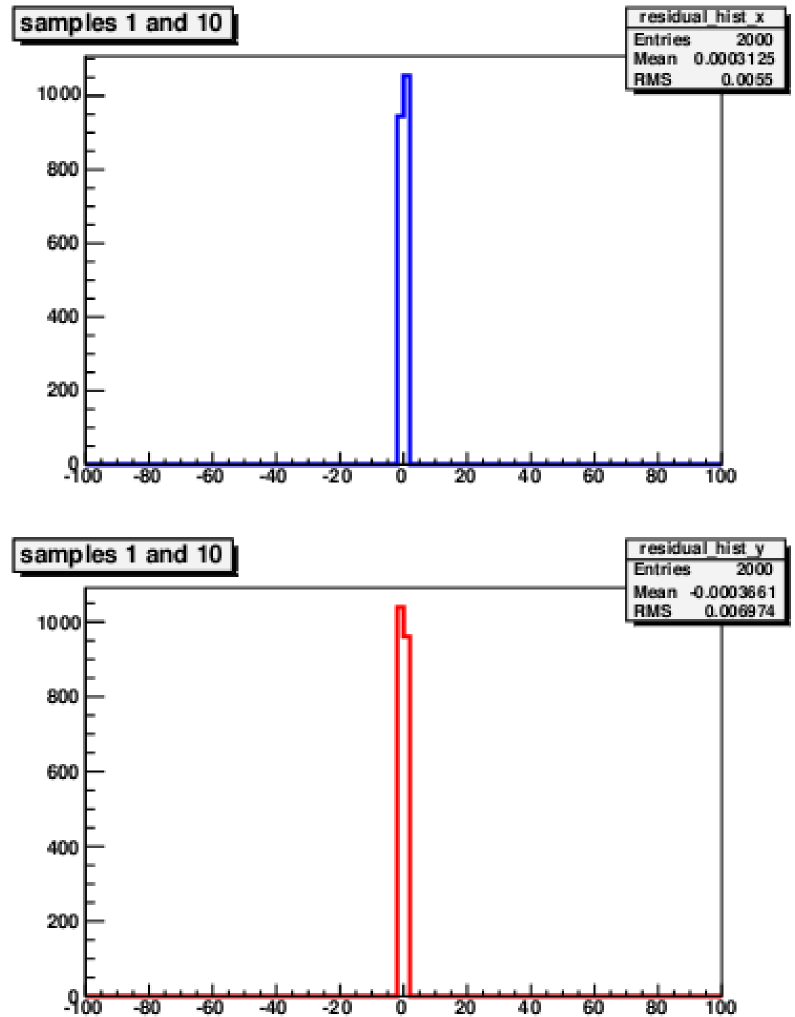
- **<u>Procedure tested on fast scope data…</u>**

Example Fit (TDS6804B)

Data and fit



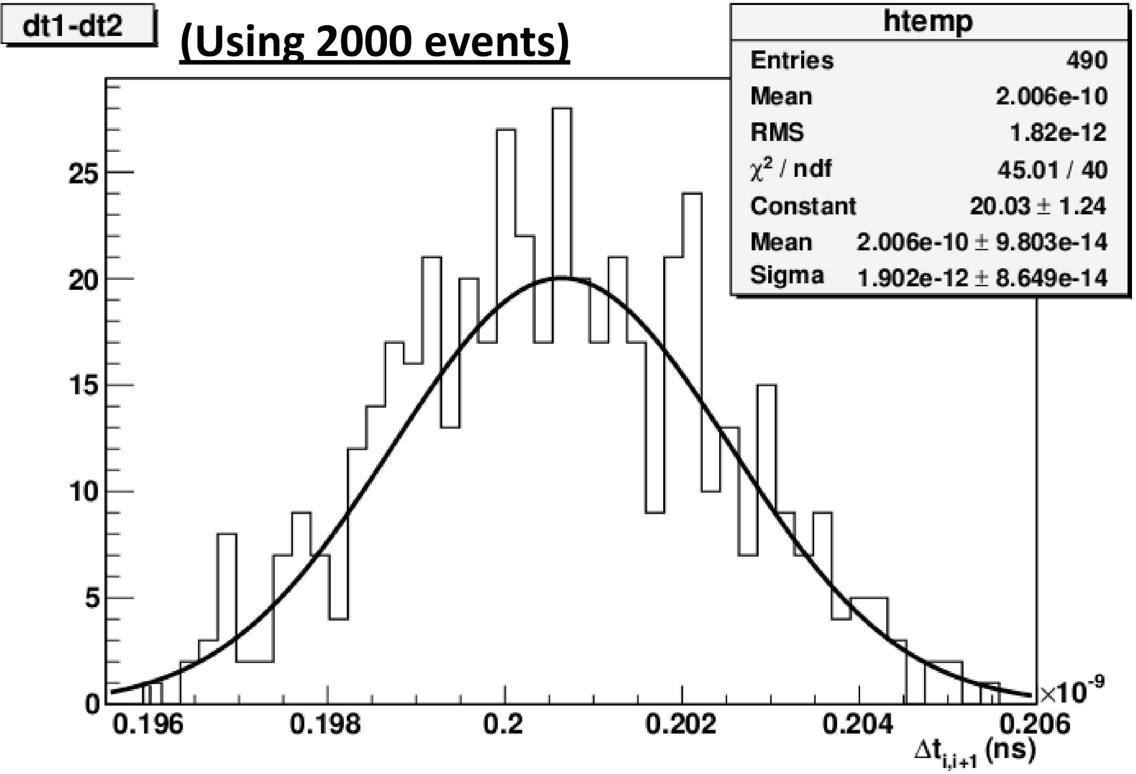
Residuals in x, y



➔ Fits very well behaved. No obvious outliers or fit failures.

Calculated Δt values for TDS6804B

(Using 2000 events)



w/ scope set to 5 GSa/s:

Δt = 200.6 ps
$\sigma_{\Delta t}$ = 1.9 ps

Excellent resolution even with a small dataset.

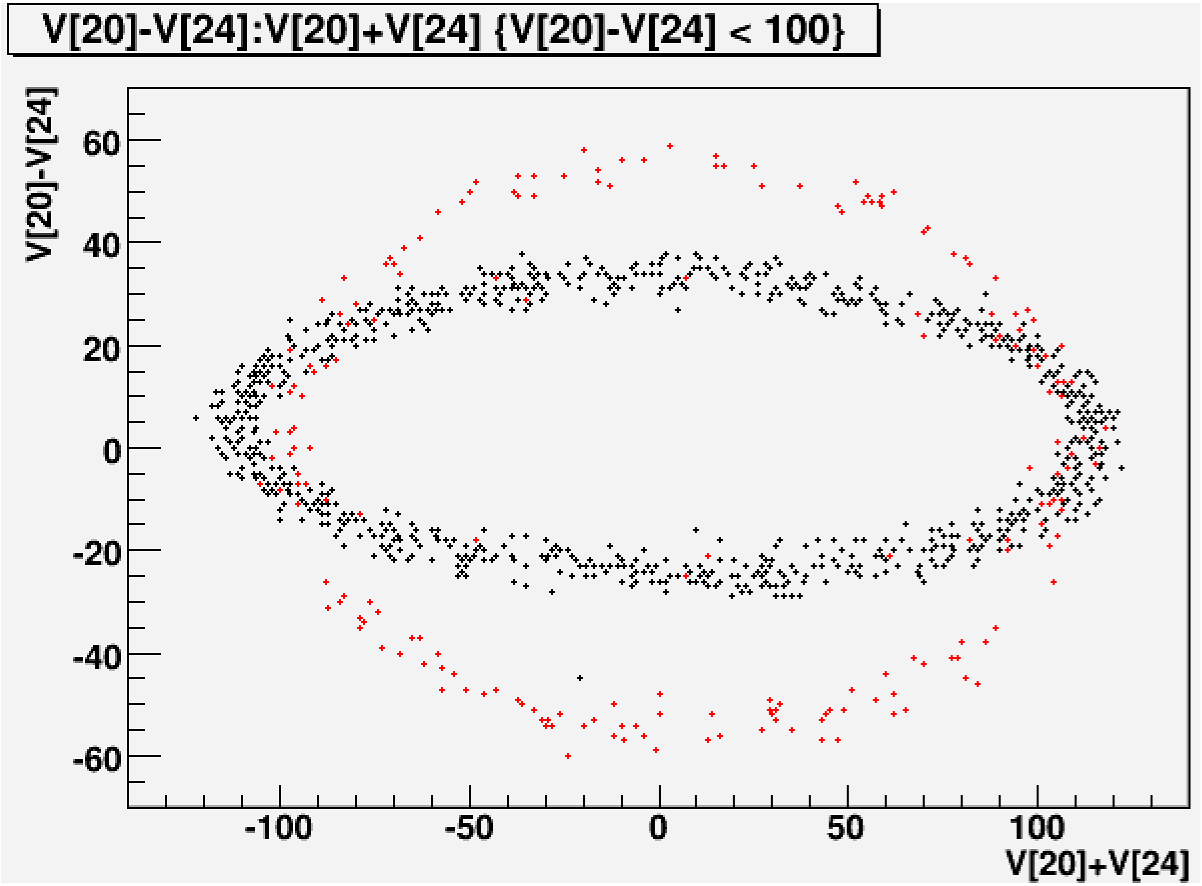
Technique is working… next tried PSEC3 data provided by Eric.

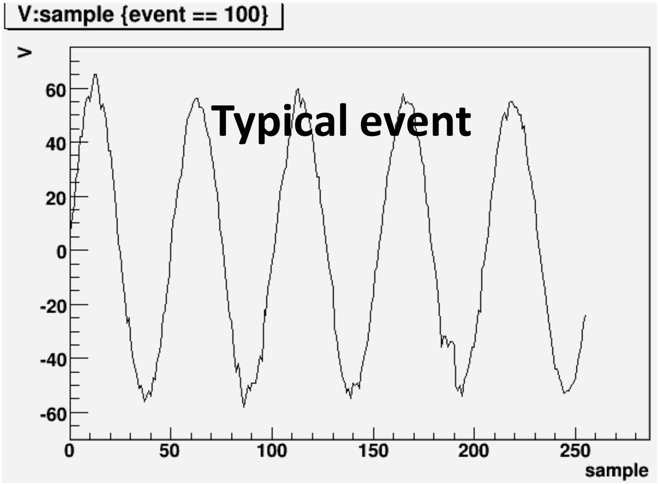
TDS6804B Datasheet

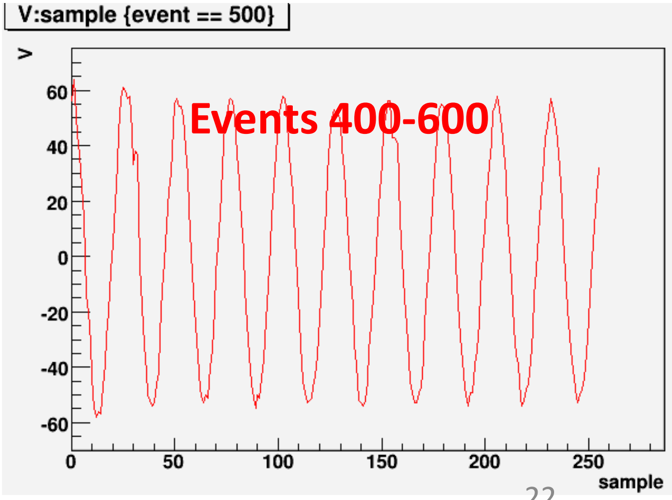
Aperture uncertainty, typical	Short term: ≤1.5 ps rms, records having duration ≤100 ms ≤800 fs rms, records having duration ≤10 µs Long term: ≤15 parts per trillion rms, records having duration ≤1 minute

Correlation-based Timing w/ PSEC3

- First attempts at timing calibration w/ PSEC3, 5 GSa/s:





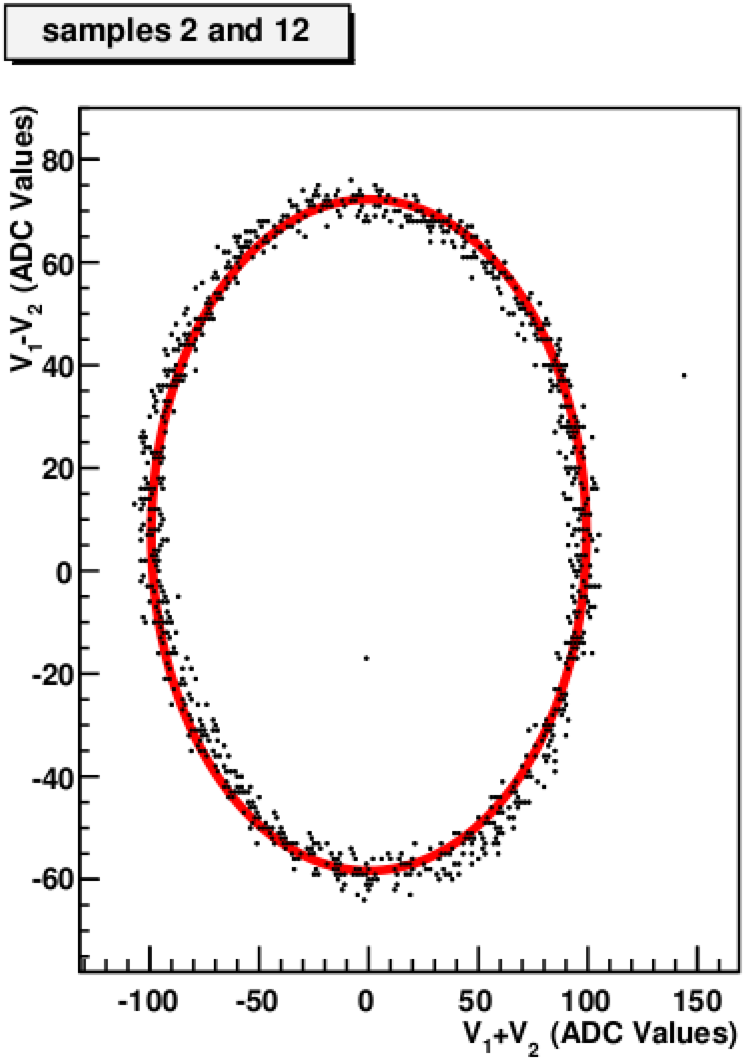


- Sampling rate slipped in events **~400-600**:

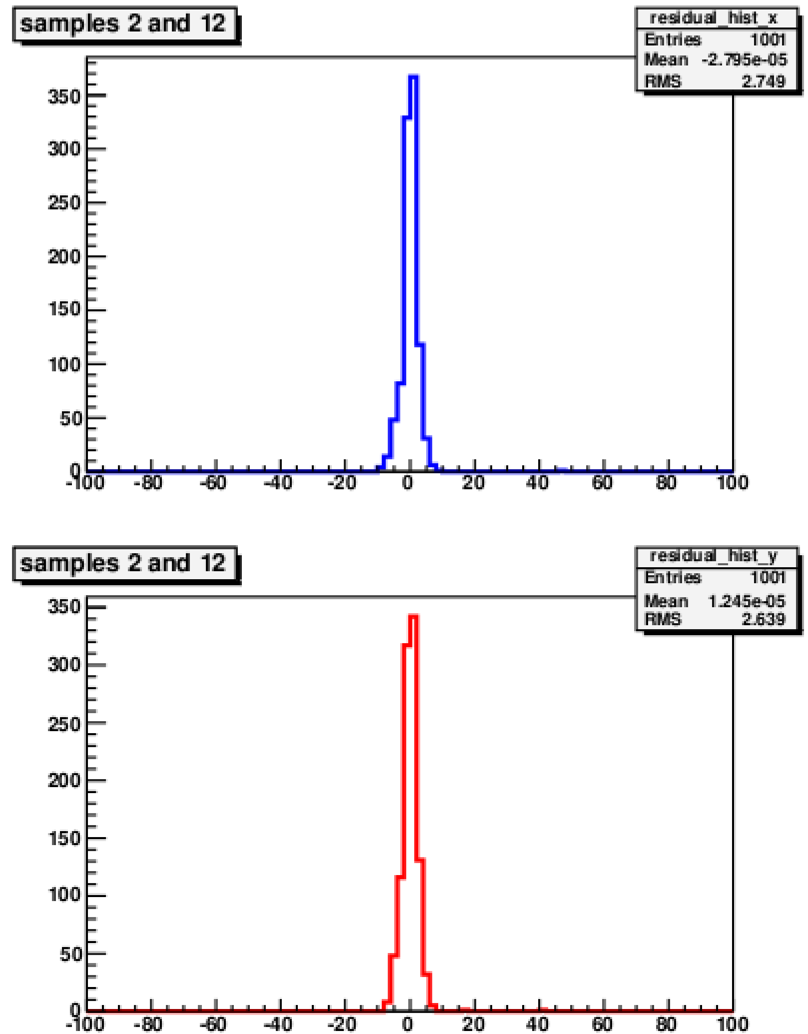
➔ **This technique was able to pick this out immediately.**

Example Fit w/ PSEC3 Data

Data and fit

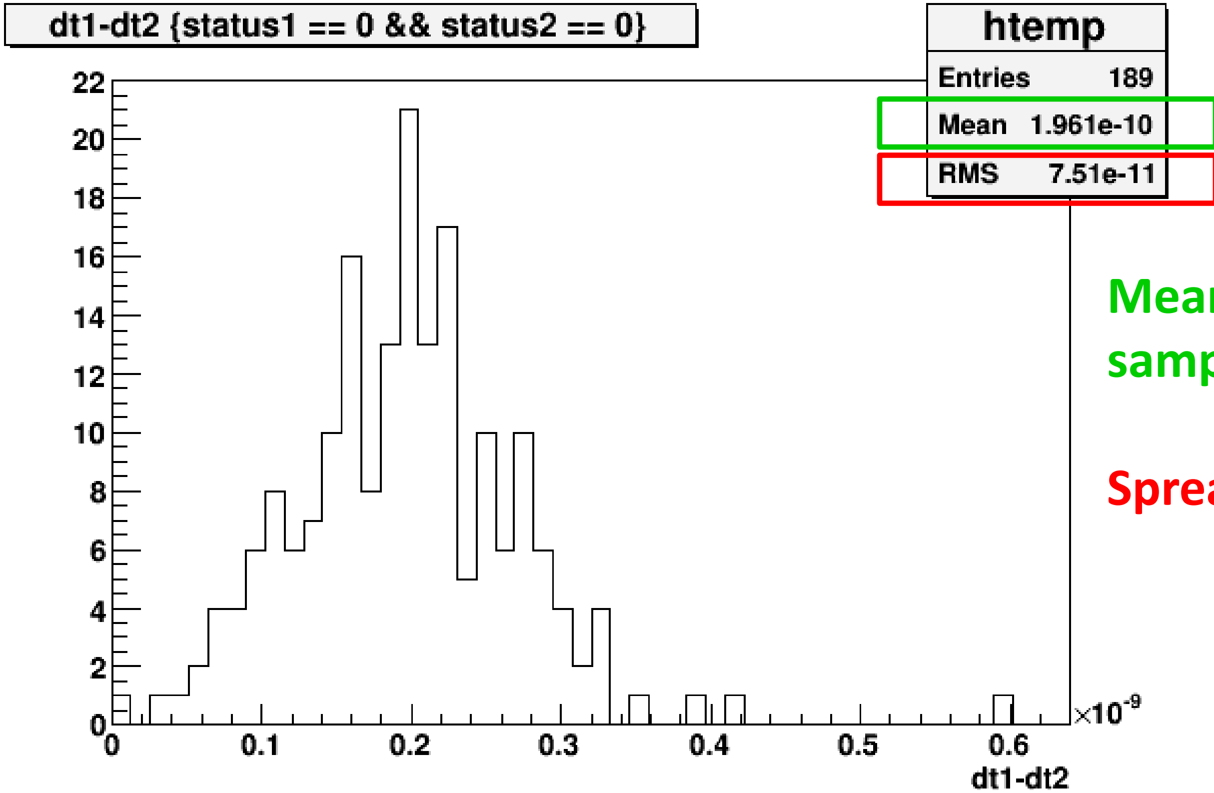


Residuals in x, y



- **Fits are well behaved, converge nicely.**
- **An obvious place to improve: outlier removal.**

Derived Distribution of $\Delta t_{i,i+1}$



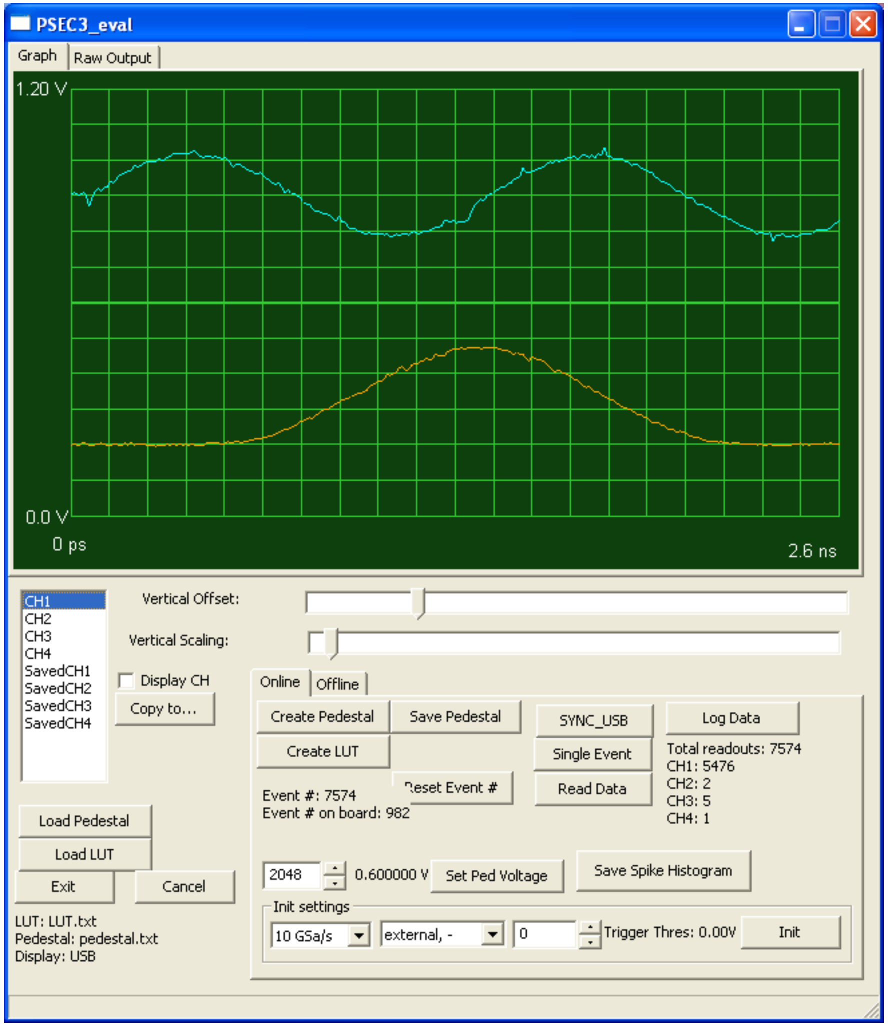
Mean is consistent with expected sampling rate, ~5 GSa/s.

Spread in values is ~ 35%.

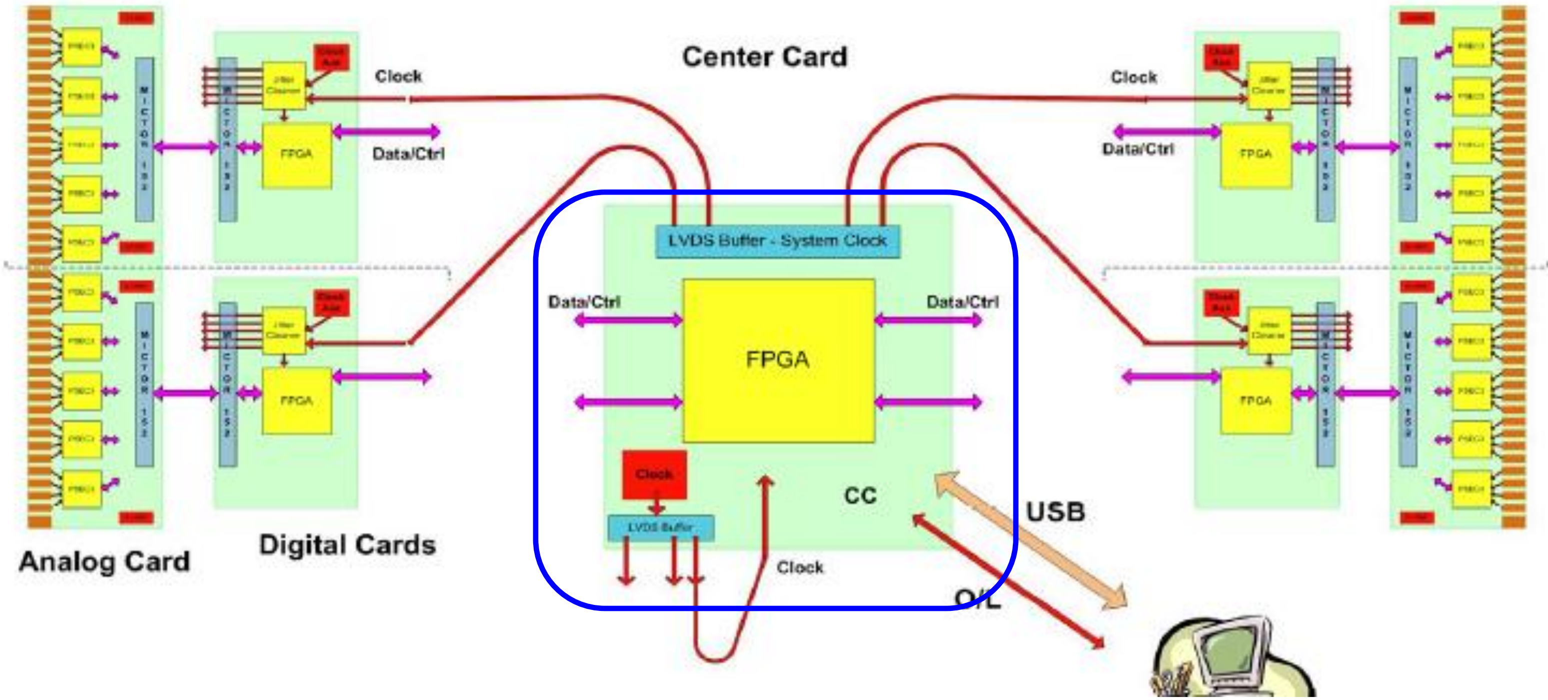
- Very promising: can measure Δt with O(~10^3) events:
  - 10^4-10^5 or more required with some techniques.
- Next few weeks… continue to improve:
  - Validation of calculated Δt values ➔ measure timing precision!

Where do we go from here?

- Currently, calibrations are applied entirely in software.
  - Still to add: Δt corrections.
  - Currently, **raw data** and **calibration constants** are saved.
    - Allows application of new constants if desired.
    - Good flexibility as we develop our methods.
  - Adding flexibility for tagging files with meta-data for identification and portability.
- What about larger systems?



Calibrations in FPGA

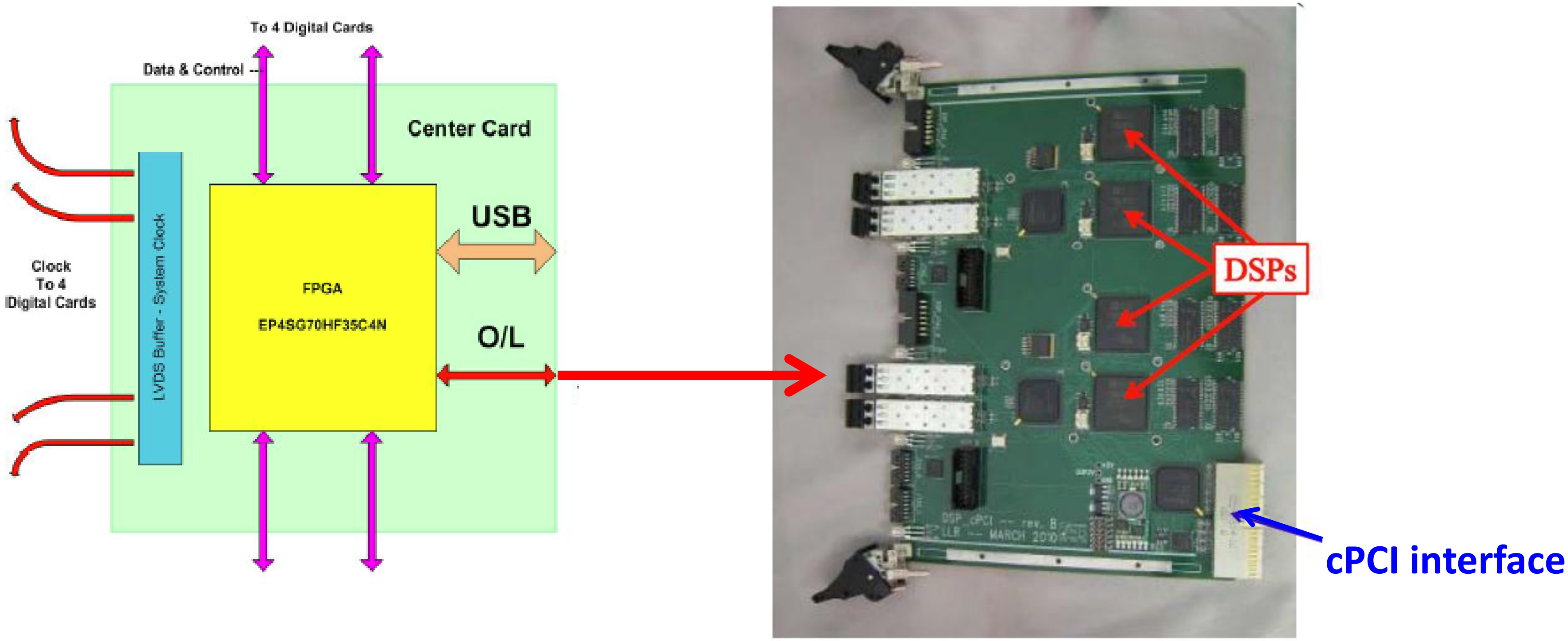


- **<u>Baseline FPGA processing model</u>**: the **central FPGA** performs necessary corrections.

➔ End user sees corrected waveforms out.

- No new hardware required.
- Voltage look-up-tables, pedestal subtractions relatively easy.
- Time-base corrections need more serious study.

Calibrations in DSP



- **<u>DSP processing model</u>**: raw data sent by fiberoptic to dedicated digital signal processing modules.

➔ End user sees corrected waveforms or feature extracted data only.

- Extra hardware needed.
- Algorithms adapted from existing software.
- Development ongoing in Hawaii for other projects (e.g., Belle II).

Summary

- Significant calibration & processing is required before ASIC waveforms are ready for use.
  - The required techniques have been developed.
- Further studies needed/ongoing:
  - Stability of calibration constants.
  - Timing precision that can be obtained.
  - How to implement such calibrations in hardware.
    - If a DSP approach is later desired, a hardware basis for it exists in Hawaii.

BACKUP

Pedestals

- Straightforward, except…
  - If ADC is nonlinear… for a given cell i:
    - $V_i^{ped} \neq (1/N) \sum ADC_i$
    - $V_i^{ped} = (1/N) \sum LUT(ADC_i)$
      - Where LUT is the lookup table from ADC counts to voltage.
      - In worst case, this LUT can vary significantly from cell-to-cell.
  - Other possible artifacts:
    - If distribution of noise is bimodal or multimodal, the mean is not the right measure of the most likely voltage for the cell.
  - Temperature dependence:
    - The pedestals may drift with temperature or other environmental variables.

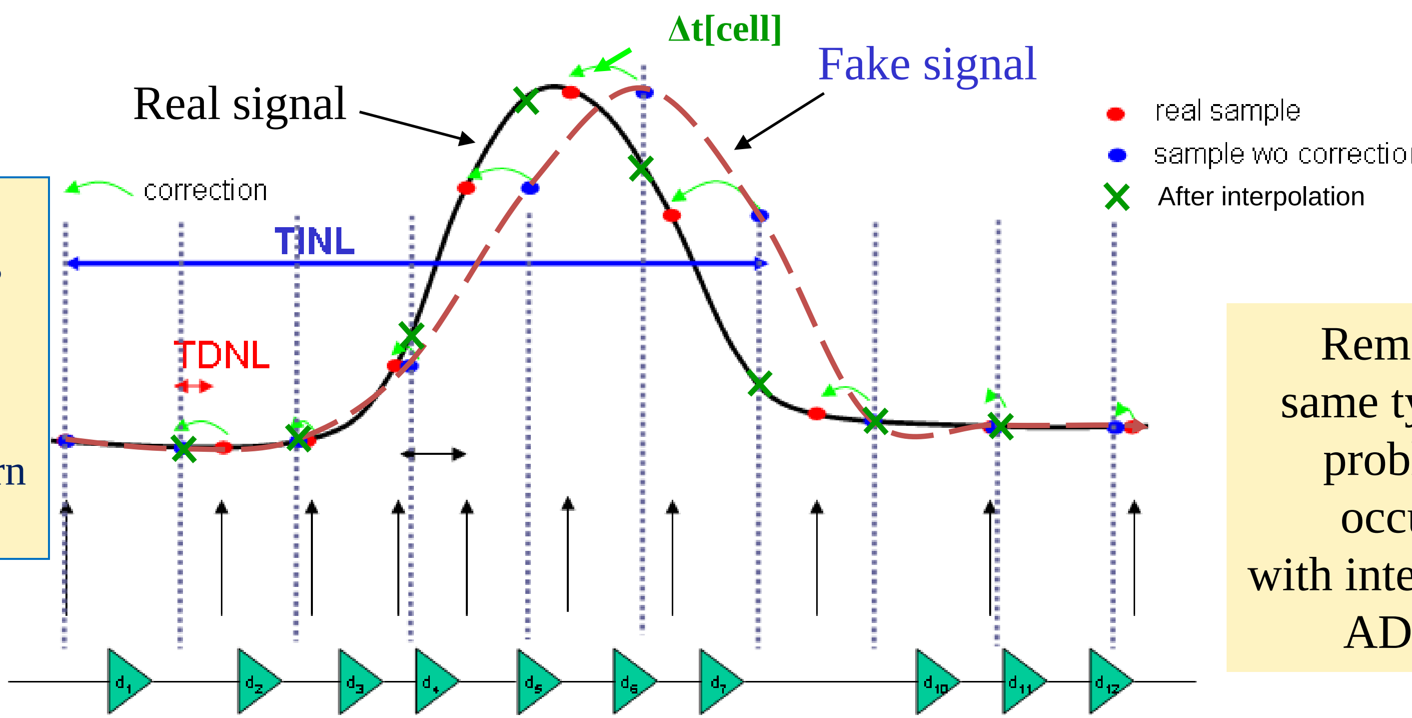
Time Non_Linearities

- Dispersion of single delays => time DNL
- **Cumulative effect** => **time INL**. Gets worse with delay line length.
- **Systematic & fixed effect** => non equidistant samples => Time Base Distortion

If we can measure it => we can correct it !

But calibration and even more correction have to remain “reasonable”.

In a Matrix system, DNL is mainly due to signal splitting into lines => modulo 16 pattern if 16 lines



Remark:
same type of problem occurs
with interleaved ADCs

Timing Calibration w/ Zero Crossings

- Occupancies
  - Use sine wave of known period.
    - Track how many times a zero crossing appears in the interval between two adjacent sampling cells, store in a histogram.
    - Normalize to average bin occupancy.
    - Use half of the average occupancy along with the known input period to get the average Δt.
    - Multiply normalized occupancies by average Δt to get bin-by-bin Δt values.
  - Caveats:
    - Since only the locations of the zero crossings themselves are useful, this requires a large data sample.
    - If pedestals are miscalibrated or drift, the accuracy will be affected.
      - Best to zero-mean each waveform before binning zero crossings, but this only corrects for pedestal drifts that are common to all cells.

*technique used by LAL group w/ WaveCatcher/SAM

Timing Calibration w/ Zero Crossings

- Amplitudes
  - Similar to previous, but exploit that sin(x) $\approx$ x for small amplitude.
    - Procedures are the same as before, but rather than binning occupancies, when a zero crossing appears weight the binning by the voltage difference along the zero crossing.
  - Caveats:
    - Same as w/ occupancies only, but statistics needed are smaller since the voltage level is being used.