Preliminary Studies on Digital Signal Processing of D0 L1 Calorimeter Trigger Pickoff Signals

Denis Calvet
calvet@hep.saclay.cea.fr

CEA Saclay, 91191 Gif-sur-Yvette CEDEX, France

Saclay, 20 November 2001

Plan

Introduction

Methodology and Analysis Tools

Signal Samples

Algorithm Evaluation Parameters

Proposed Algorithms

Current Results

Summary and Future Work

Introduction

Calorimeter information for D0 L1 Trigger

32 phi x 40 eta Trigger Towers (0.2 x 0.2 segmentation)

Each Trigger Tower: 1 EM channel + 1 HAD channel

2560 channels in total for both EM and HAD

Signal of each Trigger Tower

Differential analog signals delivered by Base Line Subtractor (BLS)

Digital conversion to be done

Estimation of the energy deposited in each tower for each Beam Crossing (132 ns)

Need of (Digital) Signal Processing

Pulse duration larger than BC period

Long rising edge -> risk of premature triggering

Electronic noise and pileup noise rejection

Methodology and Analysis Tools

Measurements on the Detector

Collect samples of original signal on running experiment (BC = 396 ns)

Understand/quantify signal shape, time jitter, noise, etc...

Simulated Signal samples

Based on pulses measured, generate train of pulses of variable amplitude, shape...

Physics simulation of detector @ F_{BC} = 132 ns -> noise, pileup studies (not done here)

Spice model of the electronic chain (not pursued in this study)

Signal Processing Algorithm Definition and Evaluation

Define requirements, propose algorithms and criteria to compare them

Evaluate algorithms on simulated samples

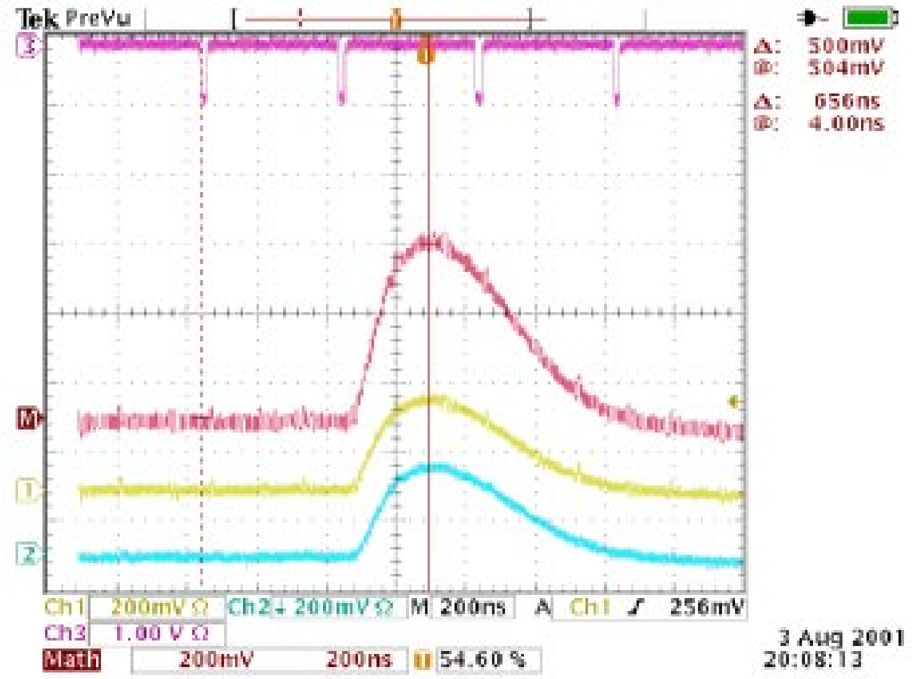
Define procedure for parameter calibration and operation monitoring

Later phase: Implementation

Feed hardware with simulated samples and check

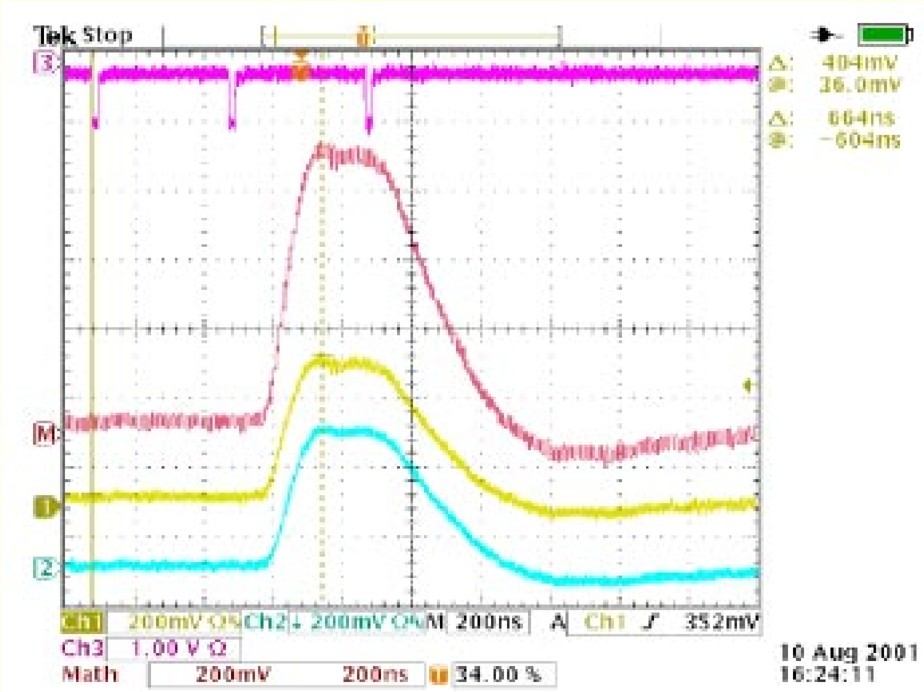
Connect to detector in spy mode...

SIGNAL SAMPLES



Scope trace of one EM channel -- red trace: differential signal; purple: BC clock (396 ns)

SIGNAL SAMPLES (CON'T)



Scope trace of one HAD channel -- note the shape
All traces and information provided by Dan Edmunds at:
http://www.pa.msu.edu/hep/d0/ftp/l1/cal_trig/pictures/trig_pickoff

Tools and Simulation Chain

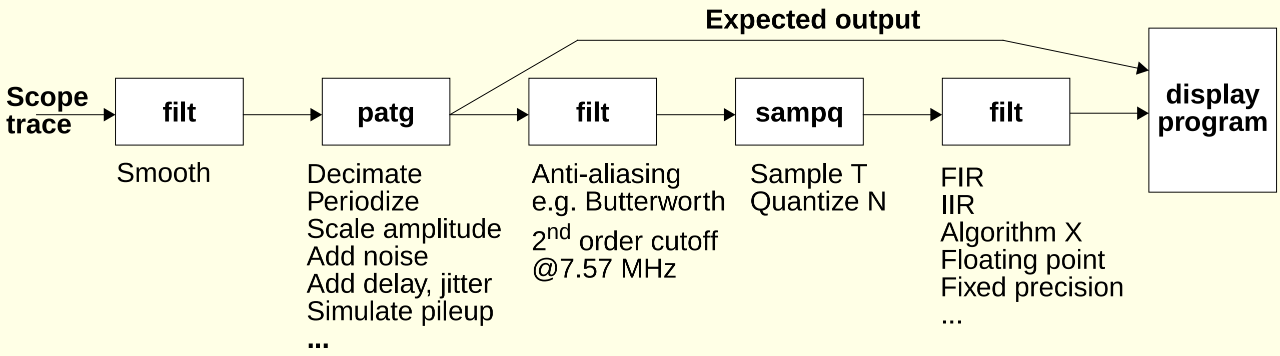
Programs developed

Pattern generator: patg

Sampling and Quantization: sampq

Digital Filtering: filt

Simulation chain



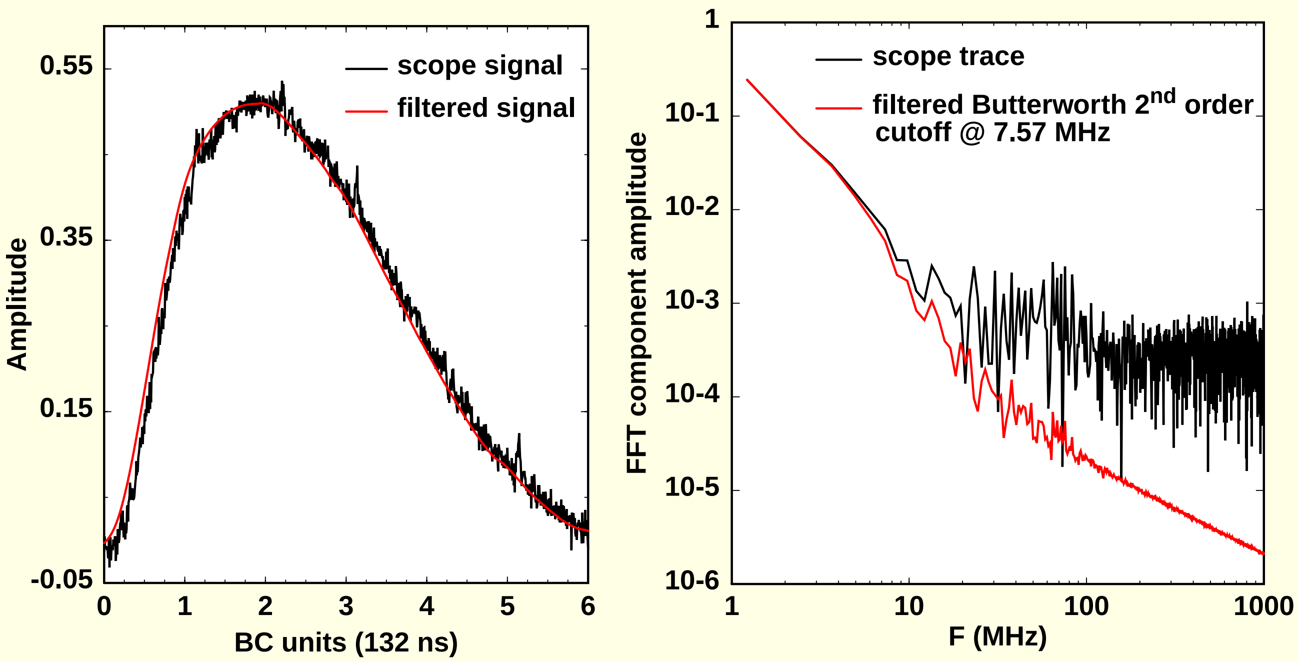
All programs standalone; written in standard C; parameter file and command line options

Input, output and intermediate files in ASCII format

I/O with data files or stdin/stdout (can use UNIX pipes between programs)

Your choice for post-processing and display program: Excel, xvgr, ...

Signal Spectrum



EM trace tek00101.csv -- 4096 point FFT

Most of the energy of the signal is in a band between 0 and ~8 MHz

Some HF noise was picked by the scope in measurements

A simulated 2nd order Butterworth lowpass filter with cutoff @ 7.57 MHz cleans up the signal

ALGORITHM EVALUATION PARAMETERS

Algorithm features

Dependent or not on trigger tower type and/or trigger tower number
Number of parameters to adjust, individually or not, procedure for parameter calculation
Algorithm intrinsic latency
Input sampling frequency and precision
Baseline correction, behavior under saturation

Algorithm quality

Precision on amplitude for the BC concerned and residual error on adjacent BCs
Time/amplitude resolution - separation of adjacent pulses
Probability on undetected pulse (e.g. small amplitude)
Probability of pulse assignment to the wrong BC

Sensitivity of the algorithm

Electronic noise, pileup noise
Signal phase and jitter compared to sampling clock
Dependence on signal shape
Limited precision arithmetic, coefficient truncation, input quantization

Implementation

Hardware resources
Operating frequency of each part
Effective algorithm latency

PROPOSED ALGORITHMS

Finite Impulse Response Filter (FIR) deconvolution

Pros: optimal for pileup rejection - linear (detection of small and large pulses in same way)
Cons: sensitivity to signal shape, phase, jitter,... - per trigger tower parameter set

Peak detector + weighted average around peak

Pros: less sensitive to signal phase and jitter - few parameters to adjust
Cons: no pileup rejection - risk of double detection for 2 peaks shaped signals (HAD calo.)

Matched filter + peak detector

Pros: optimal for white noise rejection
Cons: not optimal for pileup correction, but coeff can be tuned - per tower parameter set

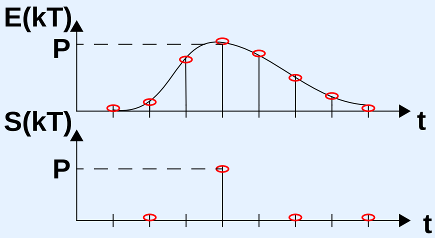
Simplified correlator + weighted average around peak

Pros: simple peak localization - little sensitivity to signal shape, phase and jitter
Cons: no pileup rejection - risk of no detection for small signals

Other ideas?

FIR DECONVOLUTION

Principle



$$\text{Error} = \begin{bmatrix} a_4 & a_5 & 0 & 0 & 0 & 0 \\ a_2 & a_3 & a_4 & a_5 & 0 & 0 \\ a_0 & a_1 & a_2 & a_3 & a_4 & a_5 \\ 0 & 0 & a_0 & a_1 & a_2 & a_3 \\ 0 & 0 & 0 & 0 & a_0 & a_1 \end{bmatrix} \begin{bmatrix} E_0 \\ E_1 \\ E_2 \\ E_3 \\ E_4 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ P \\ 0 \end{bmatrix}$$

Sample signal at $T = T_{BC} / N$ T_{BC}: 132 ns; N_{min}=2 (Shannon's sampling theorem); N_{max}~3-4

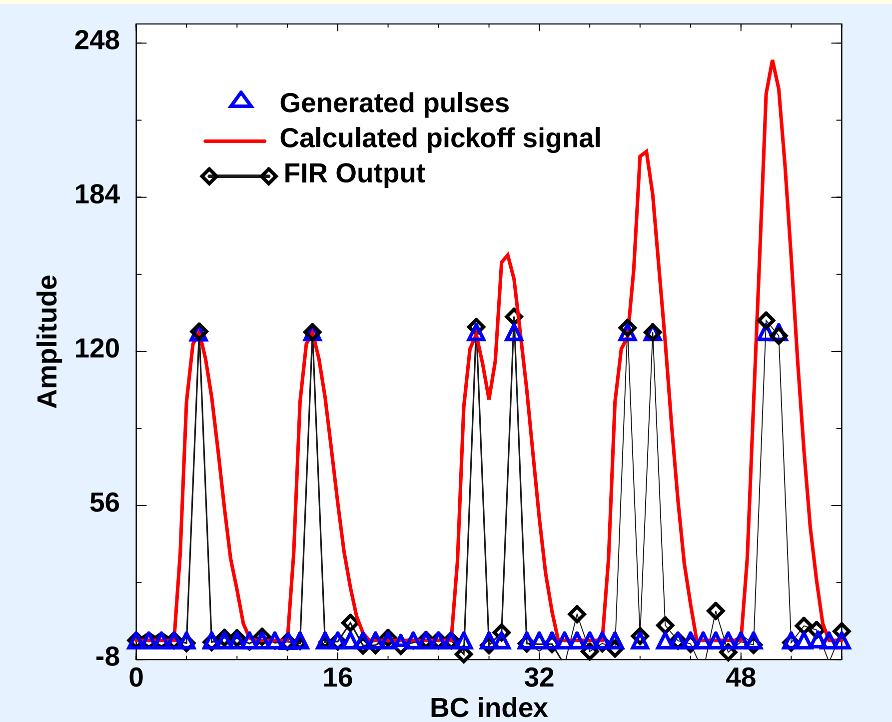
Evaluate convolution for $t = k^* T_{BC}$

Adjust coefficients a_i so that Error.Error^T is minimum (iterate over input vectors)

Features

- Set of coefficients determined on a per tower basis
- Training vector set can include shape distortion, time jitter, noise...
- If N even, number of coefficients should also be even
- Coefficients count must be sufficient to keep output at 0 after pulse
- Time position of peak adjustable: low intrinsic latency (can be 0 or even <0)
- Optimum linear solution; separation of adjacent pulses (pileup rejection)
- Signed coefficients and arithmetic

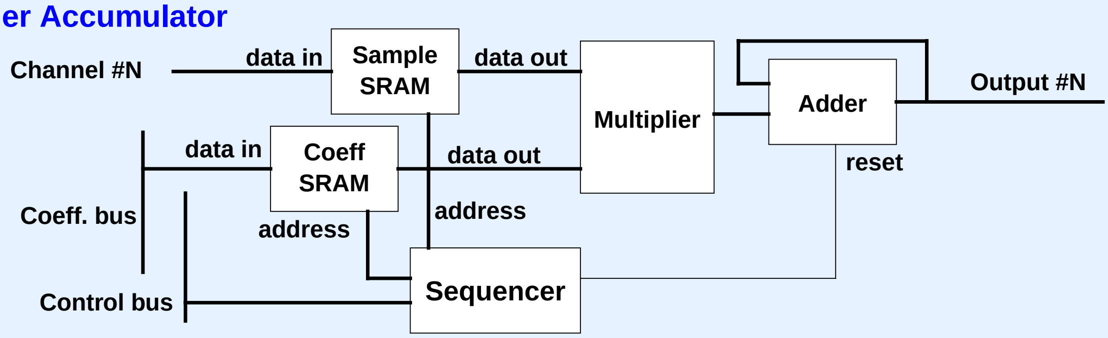
FIR DECONVOLUTION - EXAMPLE



Conditions: input: EM trace tek00101.csv -- signal sampling @F_{BC} x 2 -- 8 bit samples
FIR 12 Coefficients -- 32 bit floating point arithmetic -- algorithm intrinsic latency: 0

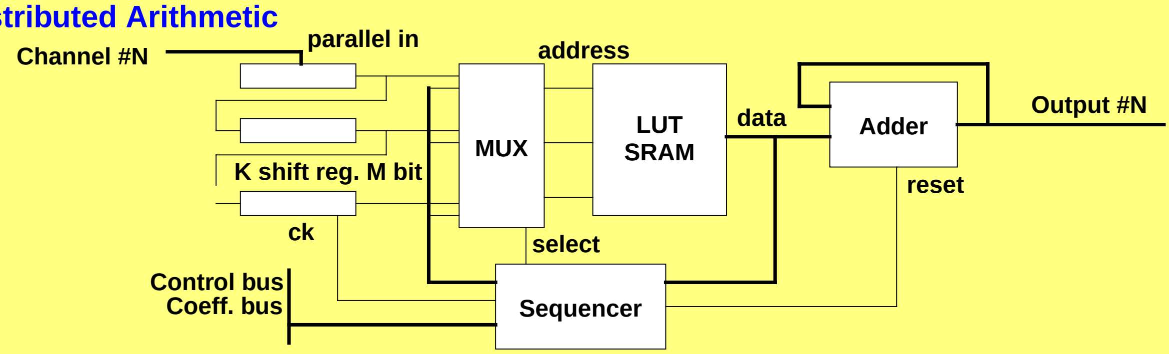
FIR DECONVOLUTION - IMPLEMENTATION

Multiplier Accumulator



K coefficients: K word SRAMs (single port or separate I/O) or parallel shift registers
On some FPGA's: dedicated 18 x 18 bit Multiplier blocks (32 on Xilinx 500K gates Virtex 2)

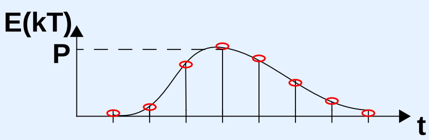
Distributed Arithmetic



K shift registers; 2**K word SRAM Look Up Table
input running at M * sampling rate; SRAM and adder running at M * beam crossing clock

PEAK DETECTOR + AVERAGING

Principle



Sample signal at $T = T_{BC} / N$ T_{BC}: 132 ns; $N_{typ} = 2$; N_{max}~3-4

Apply series of conditions to determine presence of peak; e.g.:

peak at (k-1)T IF
[E(kT) < E((k-1) T)]
AND [E((k-1) T) >= E((k-2) T)]
AND [E((k-2) T) >= E((k-3) T)]

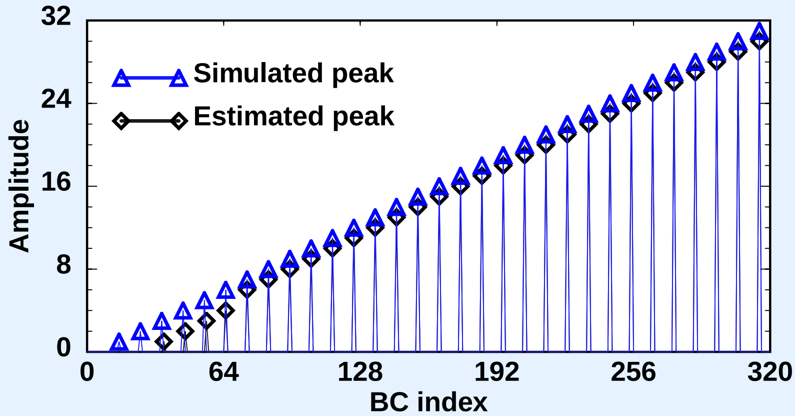
Assign value to output:
IF (!peak) R(kT)=0; ELSE R(kT) = A * (Sum of M samples around peak)

Features

- Presence of peak found by dE/dt -> oversampling degrades performance; sensitive to noise
- No rule to determine optimal set of conditions for peak detection: "try and see"
- Some separation of pulses close in time but no pileup rejection
- Output=0 for BC not concerned - but risk of missing pulses or assignment to wrong BC
- For white noise; signal almost flat around peak: noise reduction by ~sqrt(M)
- Simple hardware; few parameters; low intrinsic latency (1 sampling period in the example)
- Tolerant to pulse time jitter, pulse shape distorsion, sampling clock phase

PEAK DETECTOR + AVERAGING (CON'T)

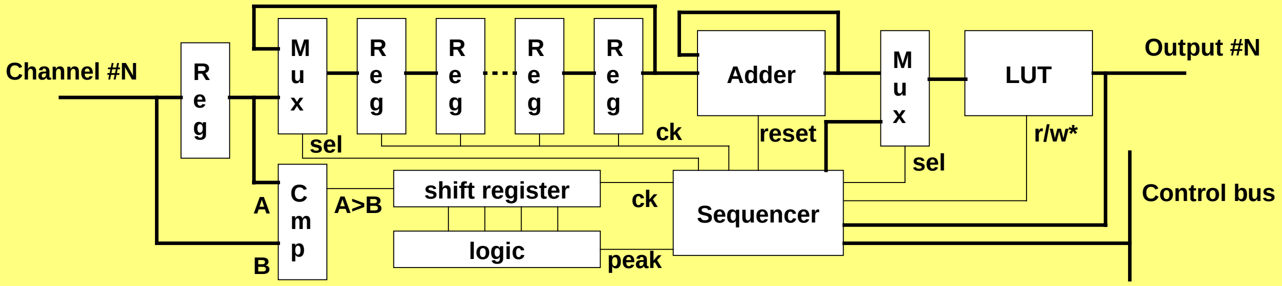
Example



Conditions:
EM trace tek00101.csv
Sampling @F_{BC} x 3
8 bit samples
8 bit arithmetic
Averaging: 4 samples
Latency: 1 T_{BC}

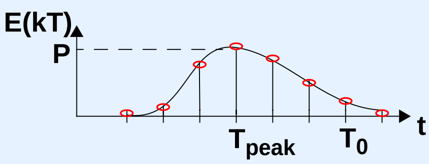
output = 0 outside BC of interest; errors for small peaks (wrong BC or no detection)

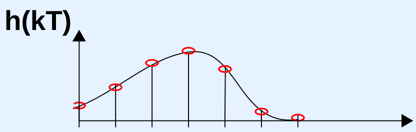
Implementation



MATCHED FILTER + PEAK DETECTOR

Principle





Matched filter: best detection E(kT) in white noise: filter impulse response h(kT) = E(T_0 - kT)
Sample signal at T= T_{BC} / N T_{BC}: 132 ns; N_{typ} = 2; N_{max}~3-4
Use FIR for convolution of input by impulse response
Output of filter != 0 for BCs around that of interest: use peak detector after filter

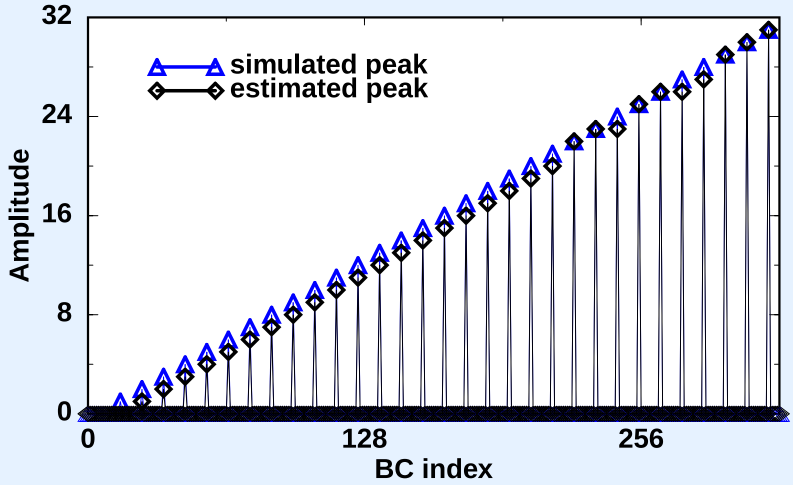
IF [S(kT) < S((k-1) T)] AND [S((k-2) T) < S((k-1) T)]
THEN R(kT) = S((k-1) T)
ELSE R(kT) = 0

Features

Optimum SNR (if white noise)
Some separation of pulses close in time (depends on signal shape)
Coefficient tuning for best trade-off between electronic noise and pileup noise rejection
Output = 0 for BC not concerned - few risks of missing pulses or assignment to wrong BC
Matching intrinsic latency = T_0 - T_{peak} i.e. several sampling periods (+1 for peak detection)
One parameter set per channel
Tolerance to pulse time jitter, pulse shape distortion (double peak), sampling clock phase

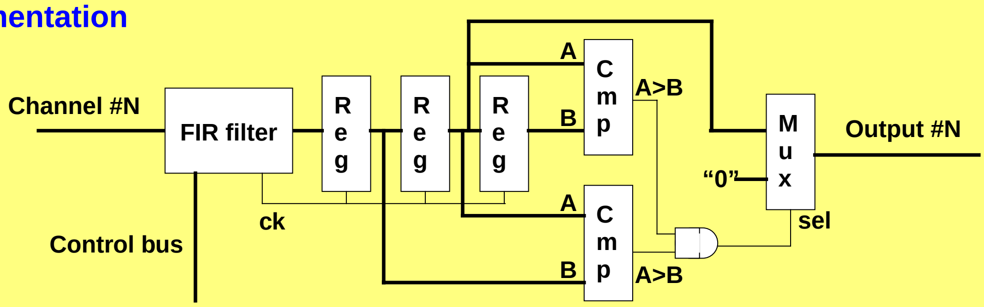
MATCHED FILTER + PEAK DETECTOR (CON'T)

Example



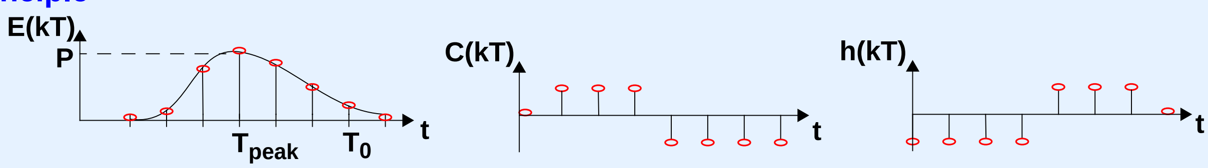
Conditions:
HAD trace tek00401.csv
Sampling @F_{BC} x 2
8 bit samples
6 coefficients 6 bit
Latency: 2 T_{BC}

Implementation



SIMPLIFIED CORRELATOR + PEAK DETECTOR

Principle



Matched filter on non-linear transformation of signal + peak detector + local averaging

Make non linear transformation of input:

IF [E(kT) < E((k-1) T)] THEN C(kT) = -1
ELSE IF [E(kT) > E((k-1) T)] THEN C(kT) = +1
ELSE C(kT) = 0

Convolve C(kT) with h(kT) -- If local maximum found: peak is present

Assign value to output:

IF (!peak) R(kT) = 0; ELSE R(kT) = A * (Sum of M samples around peak)

Features

Improved variant of peak detector + averaging around peak

Output = 0 for BC not concerned - few risks of missing pulses or assignment to wrong BC

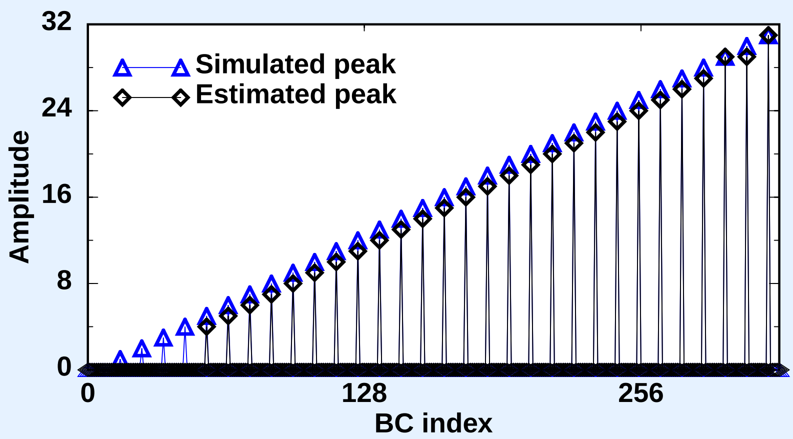
Intrinsic latency = differentiate + matching + find peak = 1 + (T_0 - T_{peak}) + 1

Few parameters specific to each channel

Tolerance to pulse time jitter, pulse shape distorsion (double peak), sampling clock phase

SIMPLIFIED CORRELATOR + PEAK DETECTOR (CON'T)

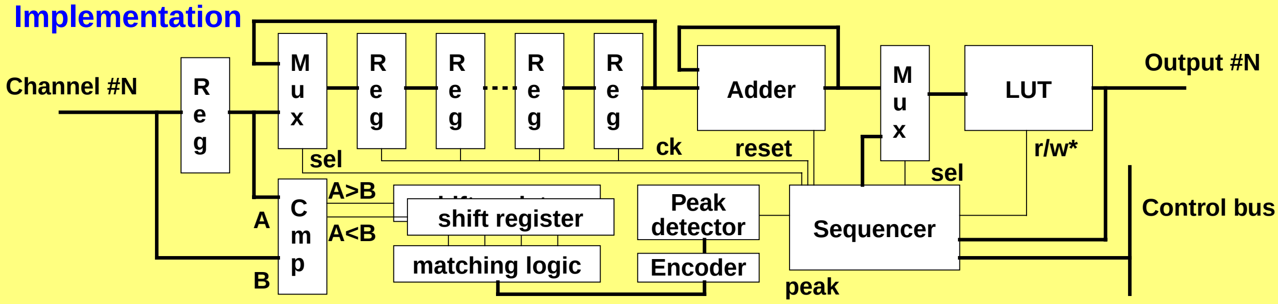
Example



Conditions:
EM trace tek00101.csv
Sampling @F_{BC} x 2
8 bit samples
Matching:
8 ternary coefficients
Averaging: 4 samples
8 bit arithmetic
Latency: 3 T_{BC}

errors for small peaks: no detection

Implementation



ALGORITHM COMPARATIVE STUDIES

DIGITIZATION

Sampling at F_{BC} x 2, F_{BC} x 3 or F_{BC} x 4

Use 8 bit of 8 or 10 bit ADC

PARAMETERS TO BE STUDIED

Signal peak amplitude over full range

Signal phase wrt sampling clock over [-1/2 T_{BC}, +1/2 T_{BC}]

Simulated electronic noise level

Simulated pileup noise

Pulses close in time

Generic set of parameters for all TT in EM or HAD

Number of samples/coefficients

Finite precision arithmetic

Behavior under saturation...

study each parameter individually then combined

Sampling Rate

Various rates to consider

ADC conversion rate: $F_{ADC} = N . F_{BC}$ N in [1, ~7] (F_{BC}: 7.57 MHz; 132 ns period)

Digital signal processing: input rate: $F_{in} = F_{ADC} / M$ M=1 or 2, 3 -- output rate: $F_{out} = F_{BC}$

Practical limits

Peak detector or simple correlation + averaging: peak detection runs @ F_{in} -- made by conditions on dV/dt: limits F_{in}<= ~3.F_{BC}; averaging @ F_{out}

Deconvolution FIR: multiply accumulate @ F_{out} but impulse response must cover duration of complete pulse: $F_{in} = 2 . F_{BC}$ to avoid too many coefficients

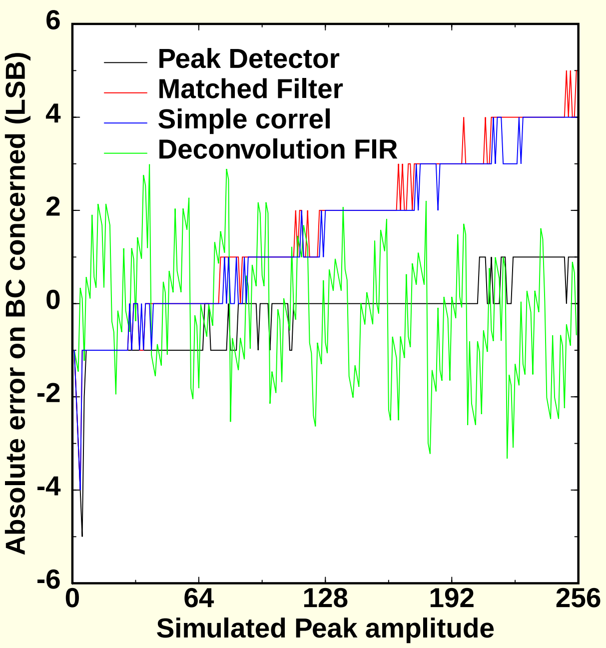
Matched filter + peak detector: all logic runs @ F_{in} : limits number of coefficients/input rate. Typical value with 100 MHz logic: 10 coeff. with $F_{in} = F_{BC}$ or 5 coeff. with $F_{in} = 2. F_{BC}$

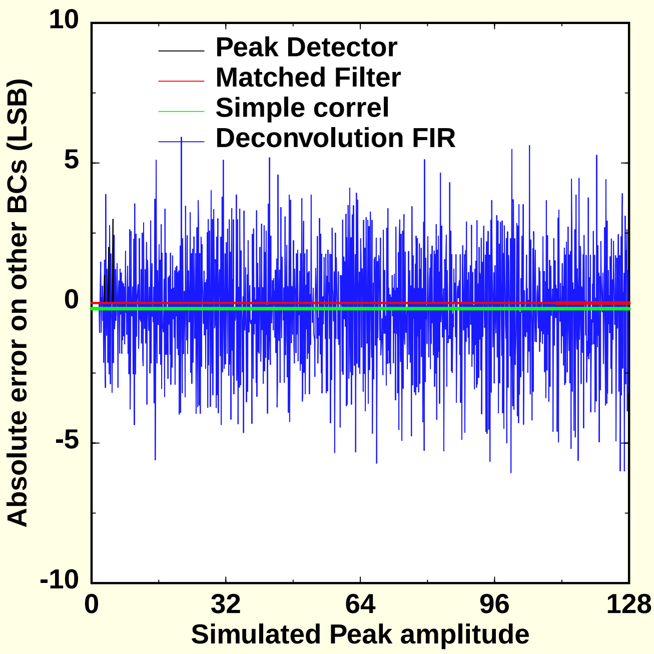
Analog to digital conversion latency

Pipeline ADCs of interest have a typical latency of 3 sampling clock for 8 bit models and 5 sampling clock for 10 bit models

$F_{ADC} = 4 . F_{BC}$ could be good choice (ADC latency = 99-165 ns) with decimation by 2 before digital signal processing -- selecting which samples to take can be used to make coarse phase adjustment of signal with respect to sampling clock

Amplitude Scaling



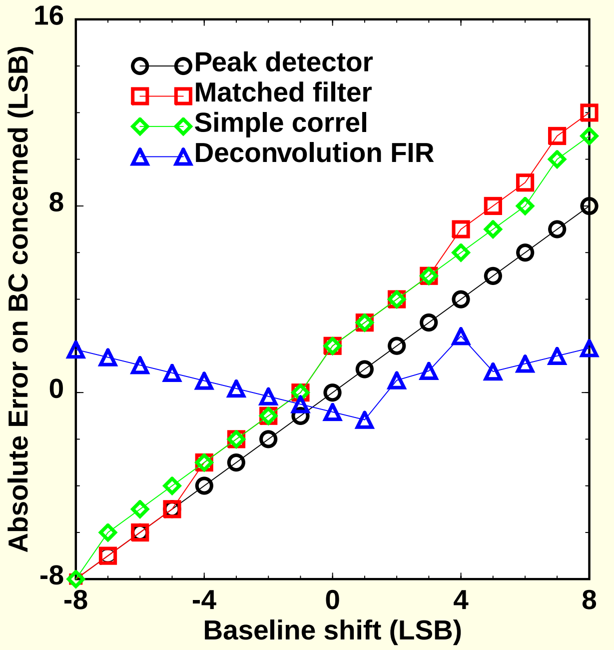


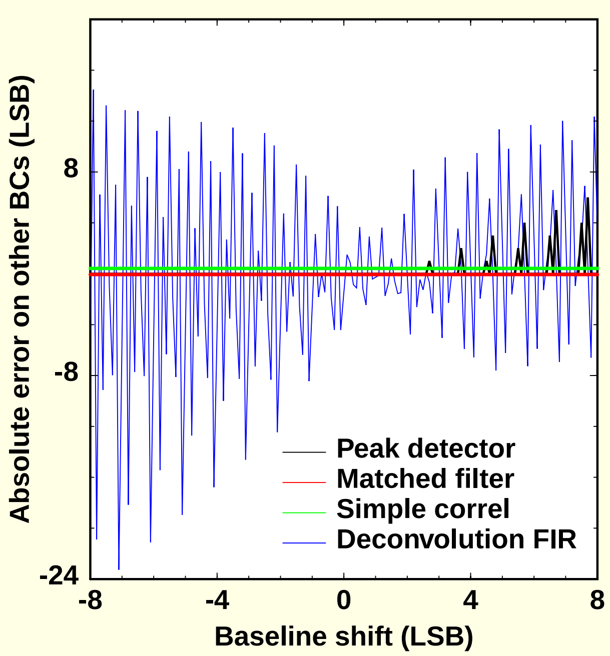
Input: series of pulses (EM tek00101.csv) with peak amplitude from 0 to 255

Small estimation errors on BC concerned for all algorithms

Non null estimation on adjacent BCs for deconvolution FIR

BASELINE VARIATION



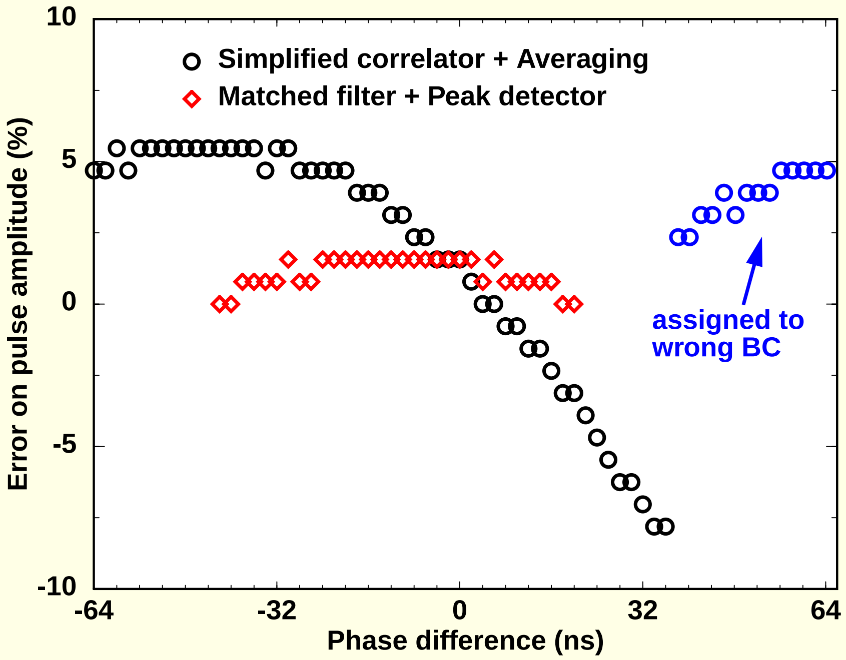


Input: series of pulses (EM tek00101.csv) Peak=128 -- baseline shifted by -8 to +8 LSB

Linear (~) estimation error for correct BC with all algorithms but FIR which is better

On adjacent BCs matching algorithms output 0; errrors with deconvolution FIR

SIGNAL PHASE WRT BC REFERENCE CLOCK



Conditions:
- **EM trace tek00101.csv**
- **Sampling @F_{BC} x 2**
- **8 bit samples**
- **Pulse peak amplitude: 128**

Simplified correlator:
- **8 ternary coefficients**
- **Averaging: 4 samples**
- **8 bit arithmetic**
- **Latency: 3 T_{BC}**

Matching filter:
- **8 coefficients 6 bits**
- **Latency: 2 T_{BC}**

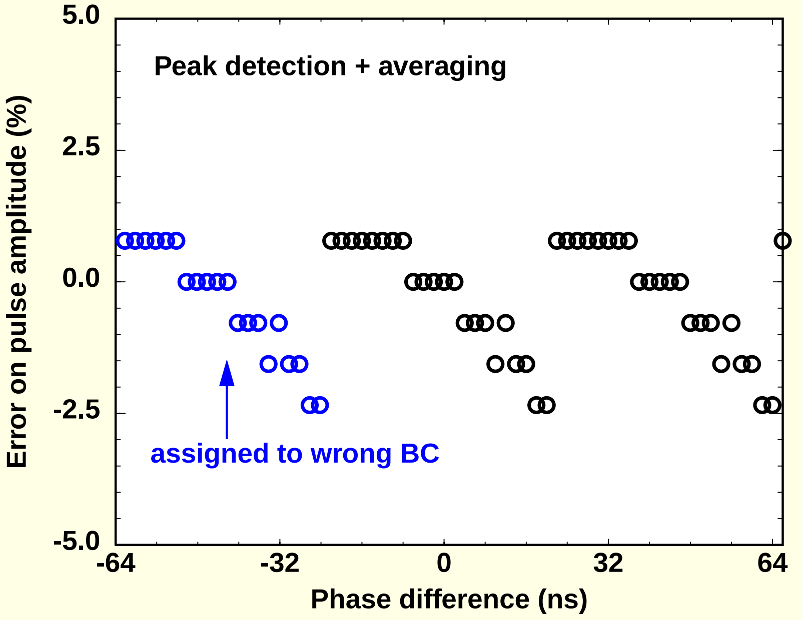
Phase difference = 0 when peak of signal phase aligned with BC clock (and sampling clock)

Simplified correlator: error of +-5% on -64, +20 ns range -- wrong assignment if phase > 32 ns

Matched filter: ~2% error on -30, +20 ns range -- peak finder misses pulse if too much shift

Algorithms robust against phase jitter and may not need compensation for static phase shift

SIGNAL PHASE WRT BC REFERENCE CLOCK (CON'T)



Conditions:
- EM trace tek00101.csv
- Sampling @F_{BC} x 3
- 8 bit samples
- Pulse peak amplitude: 128

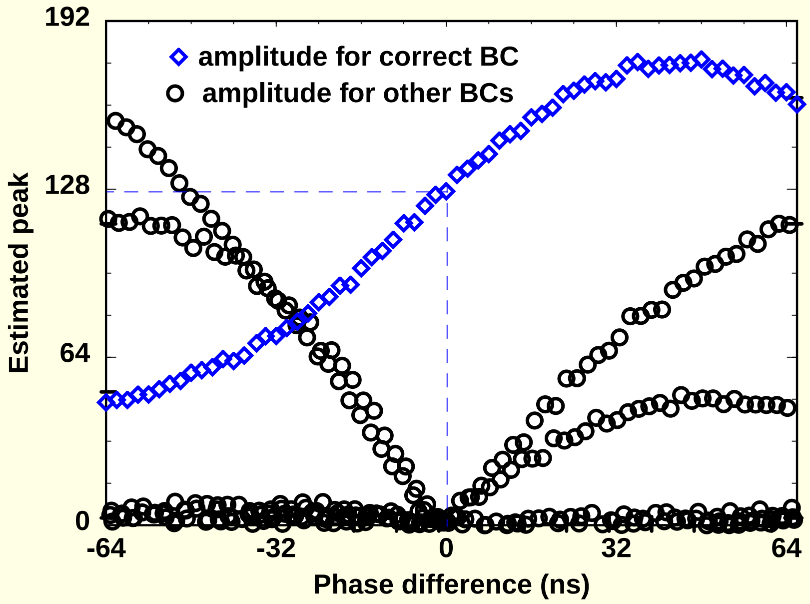
Peak detection:
- 3 conditions

Averaging: 4 samples
- 8 bit arithmetic

Latency: 1 T_{BC}

Signal sampling at F_{BC} x 3 for improved peak detection and flatter signal around peak
~2% error on estimated peak on phase range from -20 to +60 ns
peak is assigned to previous BC if phase shift in -20, -66 ns

SIGNAL PHASE WRT BC REFERENCE CLOCK (CON'T)



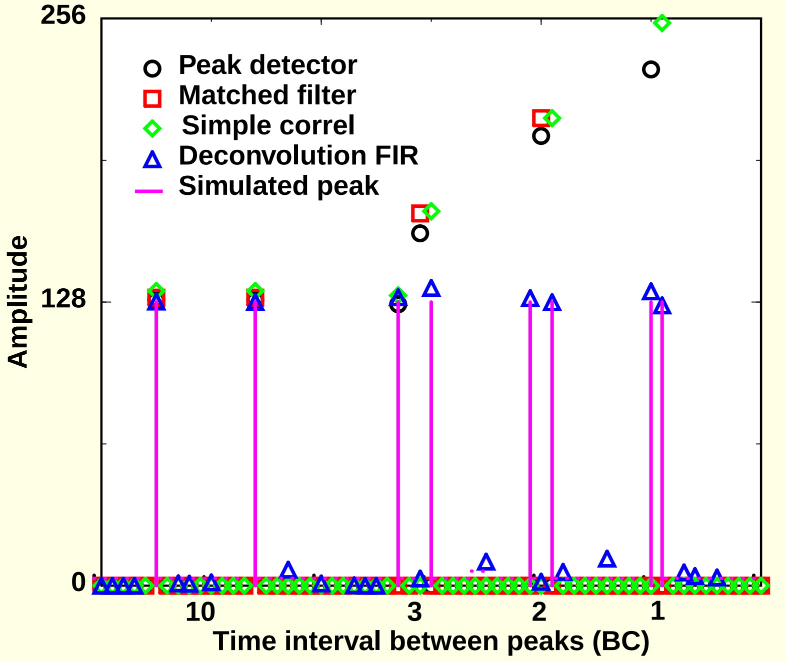
Conditions:
- EM trace tek00101.csv
- Sampling @F_{BC} x 2
- 8 bit samples
- Pulse peak amplitude: 128

Deconvolution FIR:
12 coefficients
- 32 bit floating point arithmetic

Latency: 0

FIR coefficient calculations made for only one pulse at phase = 0
-> improvement to be done: error minimization with multiple pulses shifted in time
Large errors (~50%) on phase drift by +-30 ns; non null amplitude for non relevant BCs
Coefficient tuning to correct phase of each channel -- but algorithm remains sensitive to jitter

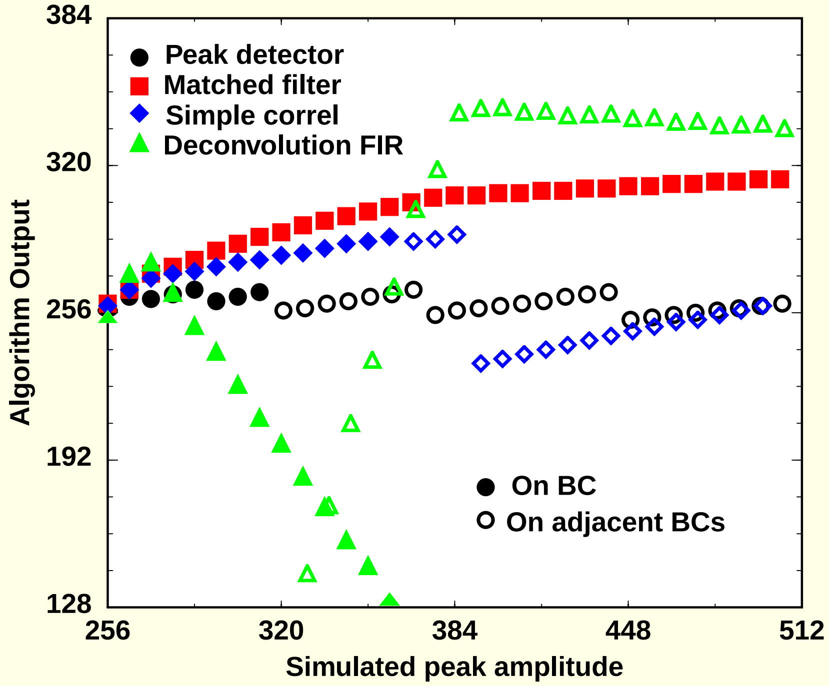
Separation of Pulses Close in Time



Input: series of pulses (EM tek00101.csv) Peak=128 -- time interval between peaks 1-10 BCs

Only deconvolution FIR gives correct results in this case

Behavior Under Saturation

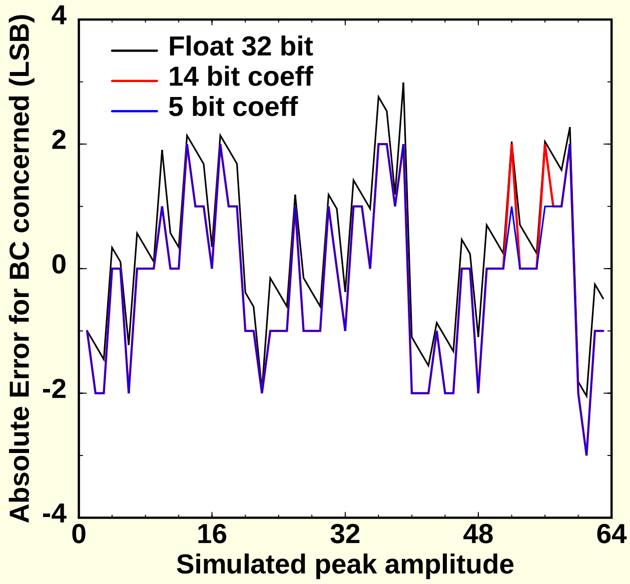


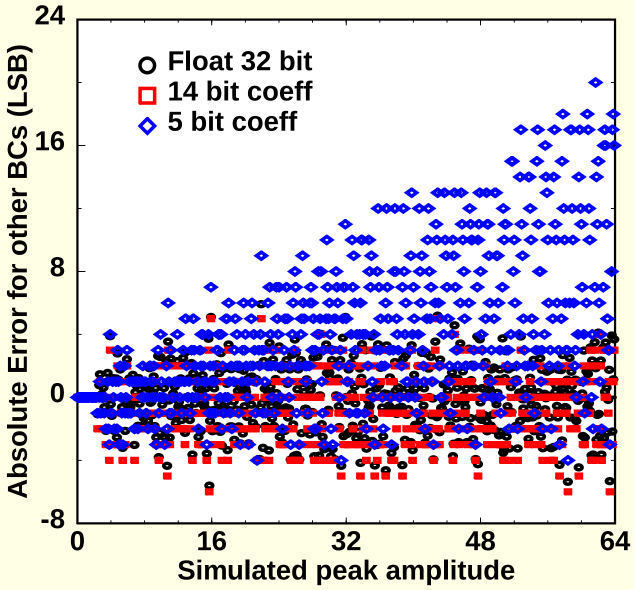
Input: series of pulses (EM tek00101.csv) Peak from 256 to 512 -- ADC saturation at 255

Best results with matched filter -- acceptable output for others if saturation <~320

For any algorithm, take care of overflow in finite precision arithmetic to avoid roll-over

FIXED-POINT ARITHMETIC - FIR

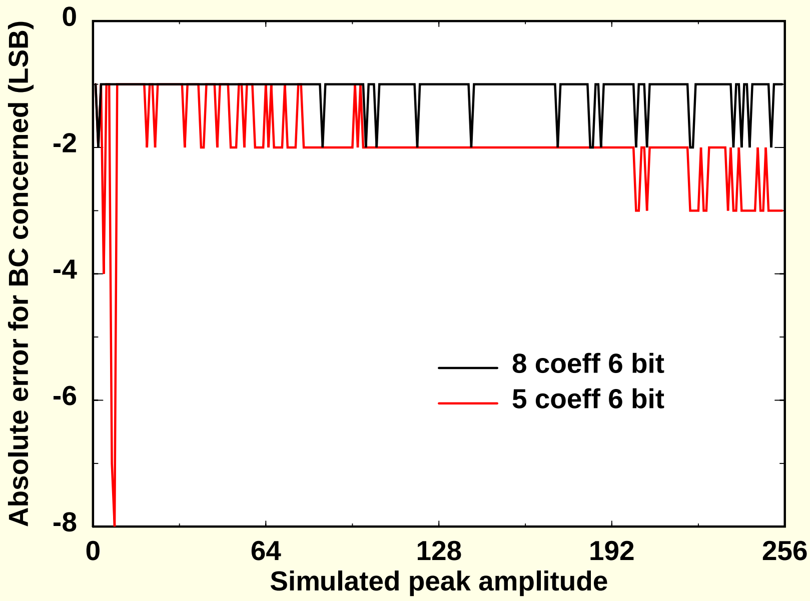




Conditions: **EM trace tek00101.csv** **Pulse peak amplitude: 1 to 63** **8 bit samples** **Sampling @F_{BC} x 2** **FIR 12 coefficients**

Not too much influence seen for estimation error on BC concerned
Residual error on adjacent BCs unacceptable if coarse coefficient quantization
Algorithm sensitive to precision of arithmetic -- signed coeff: 1 bit consumed for sign
Try to take into account number representation when coeff are calculated?

NUMBER OF COEFFICIENTS - MATCHED FILTER

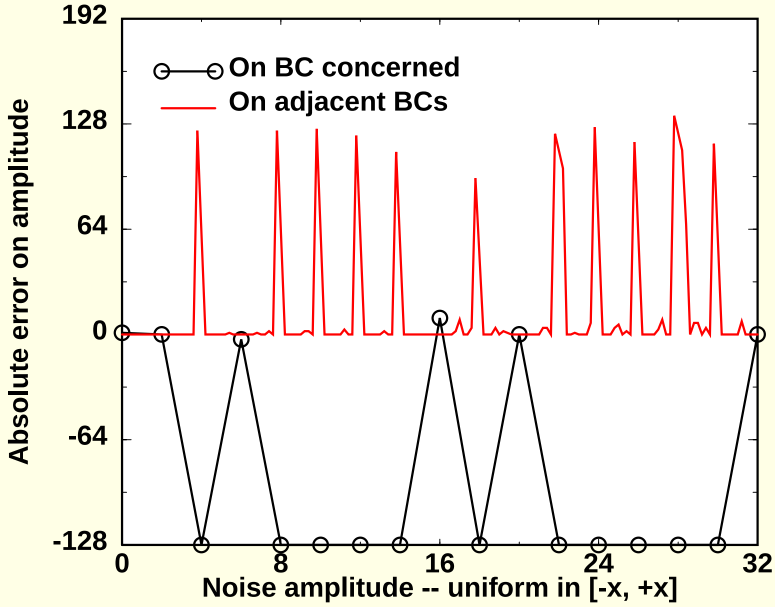


With 8 samples:
Latency = 2 T_{BC}
With 5 samples:
Latency = 1 T_{BC}

Conditions: **EM trace tek00101.csv** **Pulse peak amplitude: 1 to 255** **Sampling @ F_{BC} x 2** **Matched filter + Peak detector**

Limited precision, small number of coefficients OK -- unsigned coeff: all bits used for mantissa
Residual error on adjacent BCs null in this test
More tests needed to conclude, but very good properties seen so far

ADDING WHITE NOISE TO SIGNAL

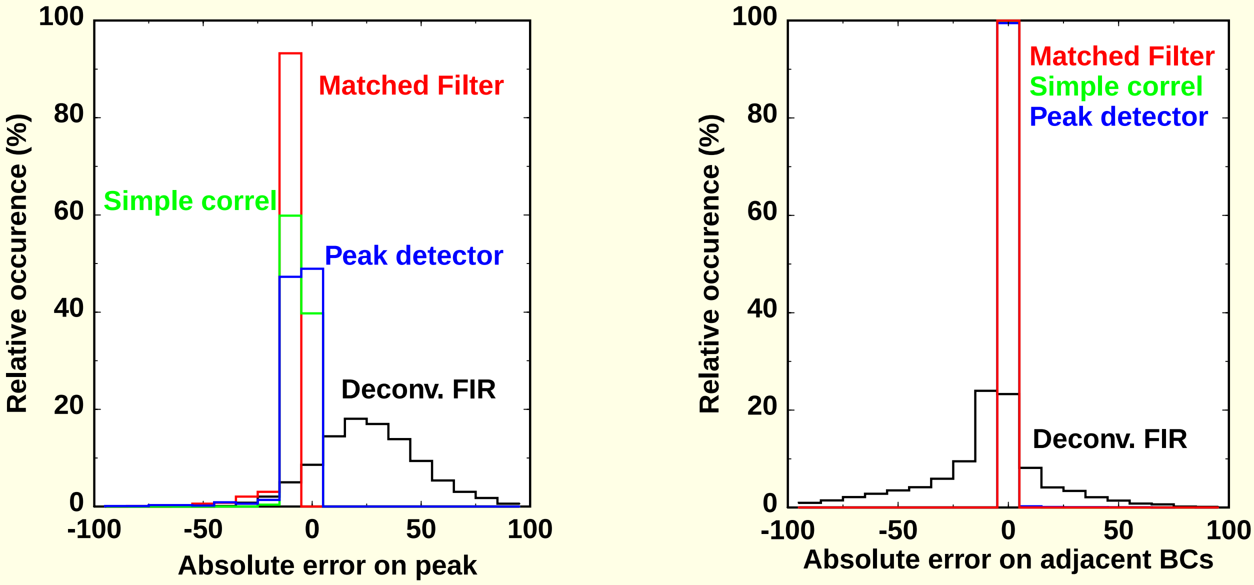


Conditions:

EM trace tek00101.csv
Pulse peak amplitude: 128
Sampling @ T_{BC} x 3
Peak Detector + averaging
Wide-band uniform dist noise
added before anti-aliasing filter

Peak detector: worst -- undetected pulses and assignment to wrong BC occur
Matched filter: ~2 LSB error max on BC; 0 on adjacent BCs
Simple correl + averaging: ~3 LSB error max on BC; 0 on adjacent BCs
Deconvolution FIR (32 FP arithm.): <~8 LSB if noise less than [-16, +16]
Noise model unrealistic; this test is for demo only
Good model of noise needed; more work to be done in this area

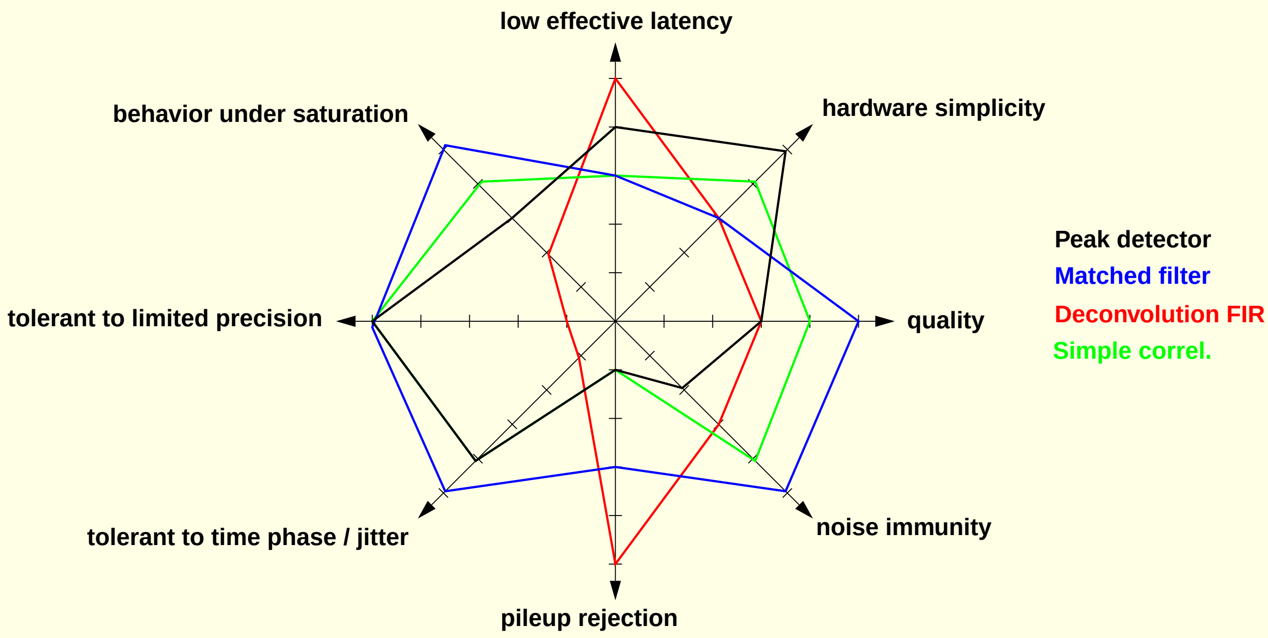
DEMO TEST WITH ALL EFFECTS COMBINED



Conditions: EM trace tek00101.csv Peak amplitude: uniform in [8, 255] Noise: uniform in [-8, +8] LSB
Pulse inter-delay: fixed 8 T_{BC} Pulse time jitter: uniform in [-10, +10] ns

A set of “good” input parameters was picked-up to show a working demo...
but parameters need to be set to realistic values; e.g. time distribution of pulse inter-arrival is unrealistic and results depend critically on pileup
Model of noise, signal jitter, pileup and tower occupancy needed

COMPARISON OF ALGORITHMS



Matched filter + Peak detector best choice -- (like theory says and other experiments do...)
Coefficient tuning for optimal pileup/noise rejection
Deconvolution FIR: no satisfactory results obtained -- ***independent study needed***

SUMMARY

Simulation chain developped

Sample generation from scope trace measurements, filtering and sampling
Allows to study algorithms off-line on a standalone PC or workstation

4 algorithms described

Linear: deconvolution with FIR
Non linear: Peak detection (2 methods) + local averaging or Matched filter + peak detection

Performance evaluation of algorithms

Large set of parameters to play with
So far matched filter + peak detection shows best results

Future work on simulation

Need to quantify incoherent noise, pileup noise, ...
Refined simulations with realistic input parameters

Still a lot to understand and do...