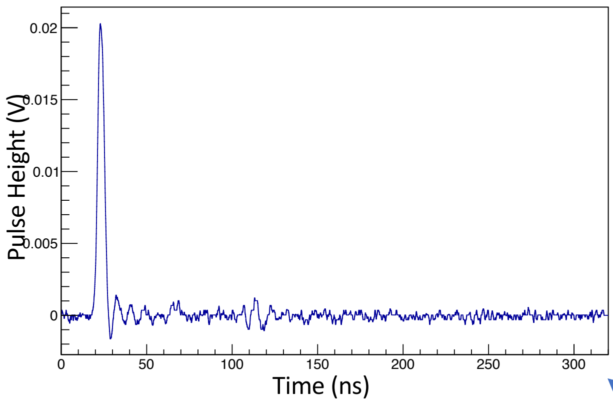
Analog Photon Processor (APP) ASIC

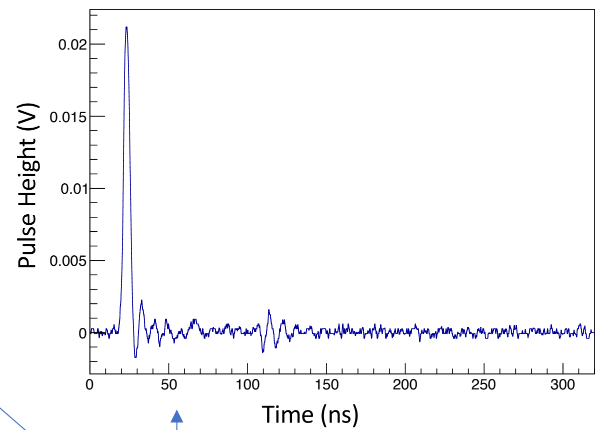
- Photon waveforms and feature extraction
- Analog Photon Processor (APP)
  - Requirements
  - Development
  - Status

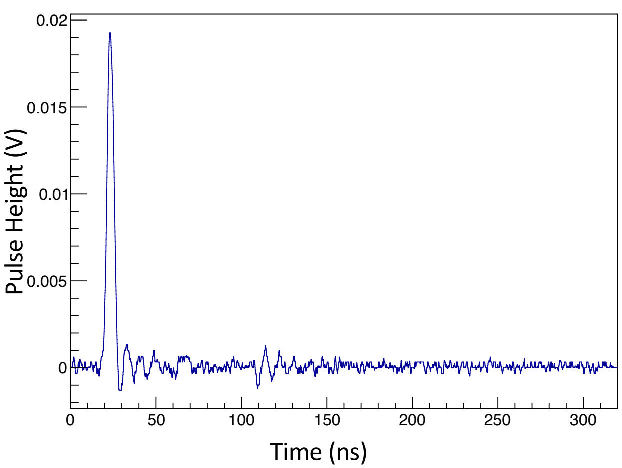
Josh Klein, Rick Van Berg, Nandor Dressnandt, Paul Keener, Adrian Nikolica
University of Pennsylvania

The Goal

- “Waveforms” in photon-based detectors are typically the sum of a small number of similar-looking pulses in each sensor (PMT, SiPM, etc.)







Nyquist sampled at 50 ps
Single photoelectron pulses
=8000 samples * 10 bits
=10 kB/channel/event

E.g., for 20 PMTs hit at 500 Hz = 3 PB/year

Not just a lot to store but a lot to analyze

Nyquist sampling recovers all the information contained within the bandwidth limit.

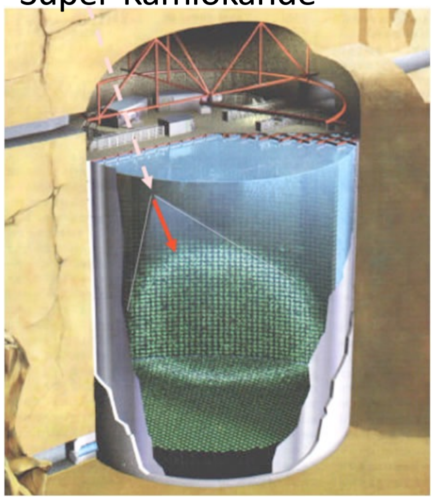
But: we only care about $N_{photons}$ and arrival times $\boldsymbol{t_i}$---pulse shape does not matter here.

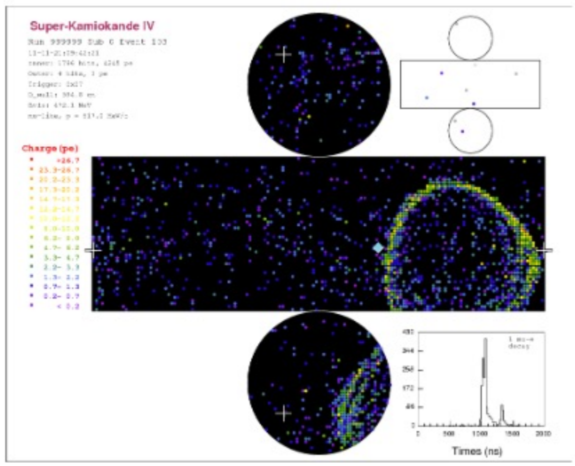
In single pe regime ($N_{photons}$=1), this is just one number.

The Goal

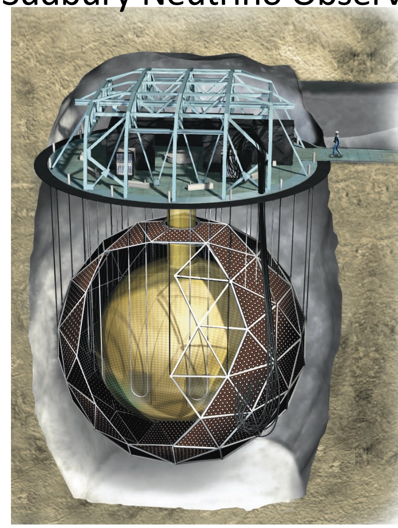
Water Cherenkov neutrino detectors have signals that are primarily single pe with up to 10-20 pe/channel

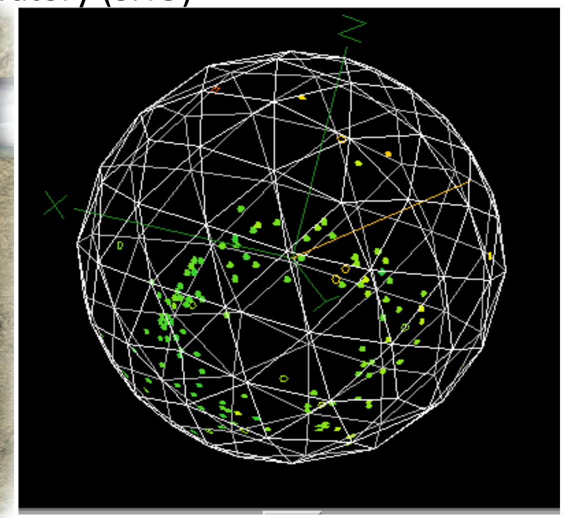
Super-Kamiokande





Sudbury Neutrino Observatory (SNO)

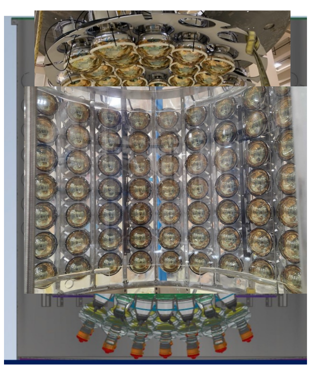


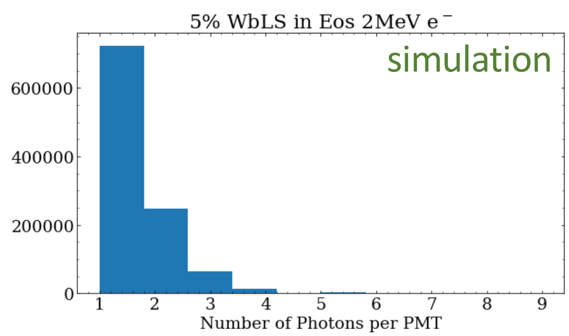


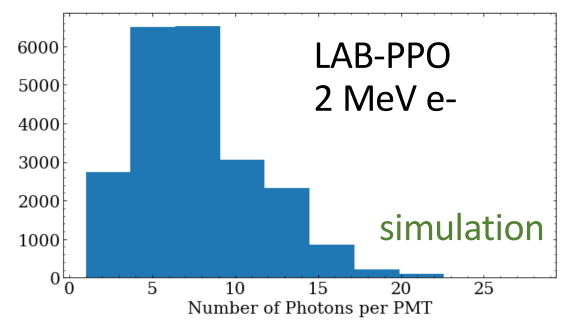
Scintillation or Water-based scintillation have higher occupancies, depending on energy and position

Eos

(240 PMTs)



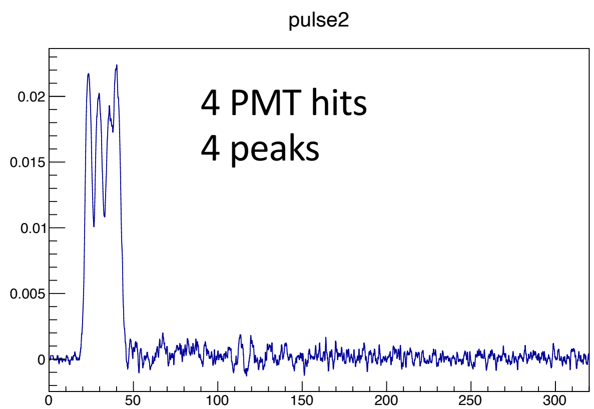




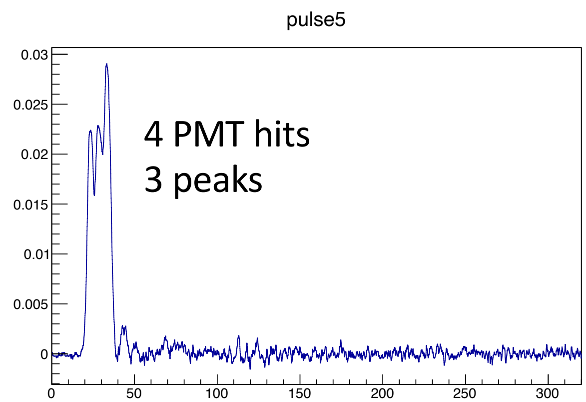
The Goal

- For multi-pe waveforms: $w(t) = \sum_{i=0}^{N_{photons}} p(t - t_i)$

Shape of waveform depends on distribution dN/dt of light

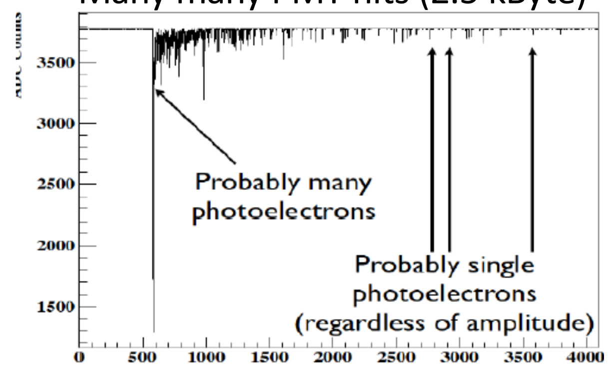


“Forced” uniform pileup



“Forced” uniform pileup

Many many PMT hits (2.5 kByte)

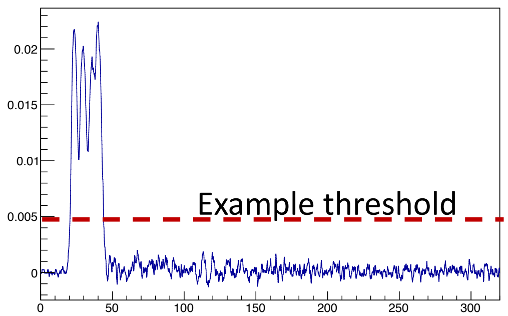


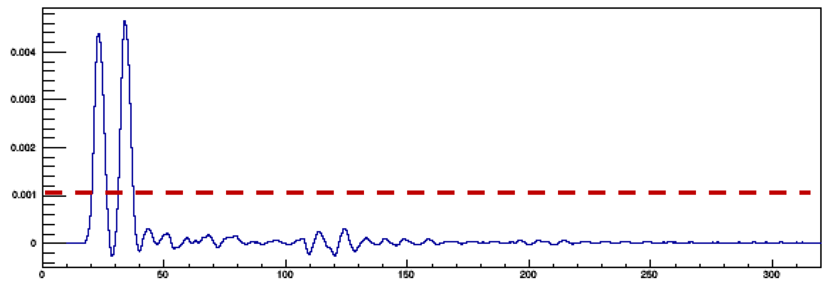
LAr scintillation

Are there features we can extract that give us $N_{photons}$ and t_i well enough that we do not need full waveform digitization, even in the multi-pe regime?

Definition of "Resolved Pulse Packet"

Critical to feature extraction

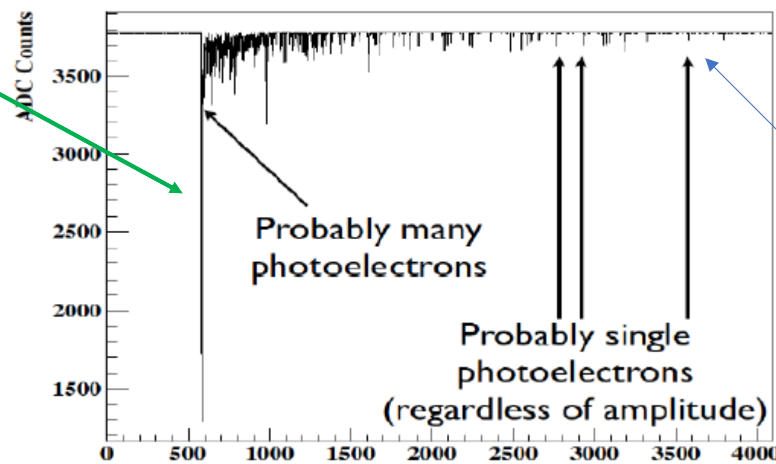




"Fully resolved" means pulses that fall below threshold and then have a second rising edge above threshold

Not resolved

Analysis of fully resolved pulses just recapitulates analysis of unresolved pulses ---resolved pulses each can have their own analysis



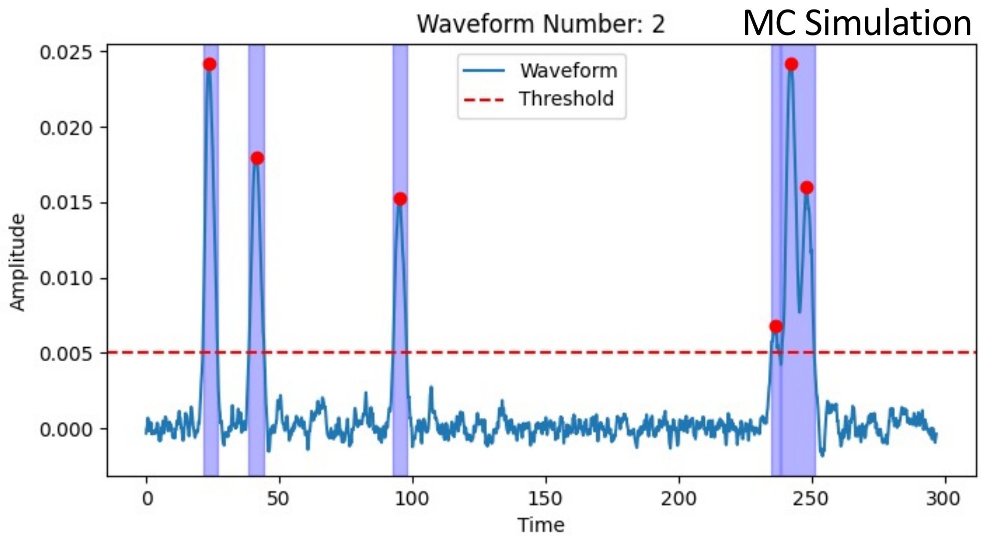
Fully resolved

Feature List

Within each resolved pulse packet measure:

- Start of rising edge t_r
- End of falling edge t_f (t_f-t_i = time over threshold)
- Peak times and values
- Total integral

MC Simulation



"retriggerability" necessary to start analysis of each pulse packet

MC Simulation

```
Waveform Data:
  Number of Resolved Packets: 5
Packet Info:
    Packet:
      Start Time: 21.257383966244728
      Integral: 0.09355681358849272
      Time Over Threshold: 5.6160337552742625
      Number of Peaks: 1
      Peak:
        Time: 23.624472573839665
        Height: 0.024190603362512775

Packet Info:
    Packet:
      Start Time: 38.616033755274266
      Integral: 0.07561360795008586
      Time Over Threshold: 5.755274261603375
      Number of Peaks: 1
      Peak:
        Time: 41.58649789029536
        Height: 0.017928273926372 64

Packet Info:
    Packet:
      Start Time: 92.82700421940929
      Integral: 0.05996750654964445
      Time Over Threshold: 5.105485232067508
      Number of Peaks: 1
      Peak:
        Time: 95.14767932489453
        Height: 0.015224094975565095

Packet Info:
    Packet:
      Start Time: 234.8523206751055
      Integral: 0.020924396618506546
      Time Over Threshold: 3.10970464135022
      Number of Peaks: 1
      Peak:
        Time: 236.75527426160338
        Height: 0.006779438317607855

Packet Info:
    Packet:
      Start Time: 238.8438818565401
      Integral: 0.1741331640922245
      Time Over Threshold: 12.206751054852333
      Number of Peaks: 2
      Peak:
        Time: 242.46413502109706
        Height: 0.02419060538522899
      Peak:
        Time: 248.12658227848104
        Height: 0.0160305989702465
```

Analog Feature Extraction

Clearly, features could be found with FPGA operating on digitized data but:

- High-end FPGAs are not cheap
- Firmware is frustrating (or expensive)
- Fast sampling ADCs are power-hungry and expensive
- Bandwidth must be limited to sample at Nyquist---limits timing precision

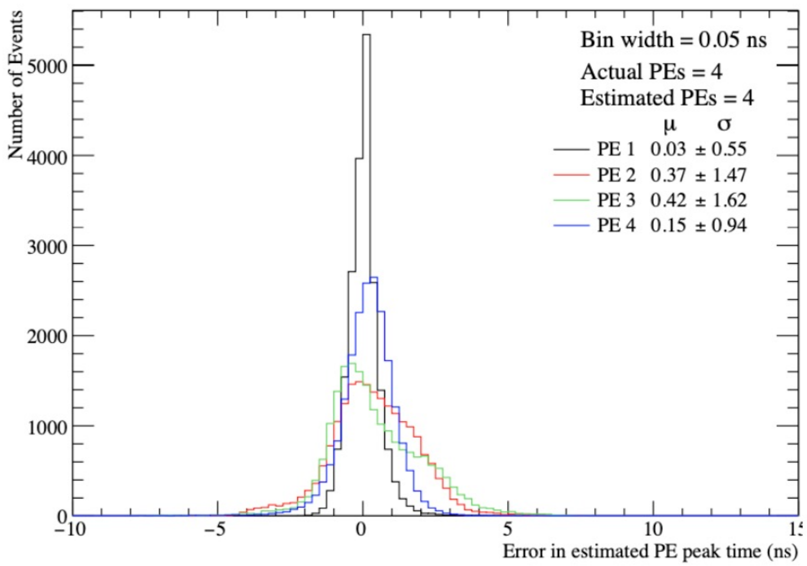
Analog approaches can measure all the features and:

- Much cheaper ($5/channel?)
    - ADCs can be very slow (1-10μs conversions)
- All "useful" information but with much smaller data size---faster data analysis, less storage
    - Up to 7 measurements/packet (for two peaks) @ 12 bits ~ **11 Bytes/packet**
- Can have better timing precision because marginal Nyquist restrictions
- Little to no deadtime
    - Only during switch from one resolved packet to another---could be zero

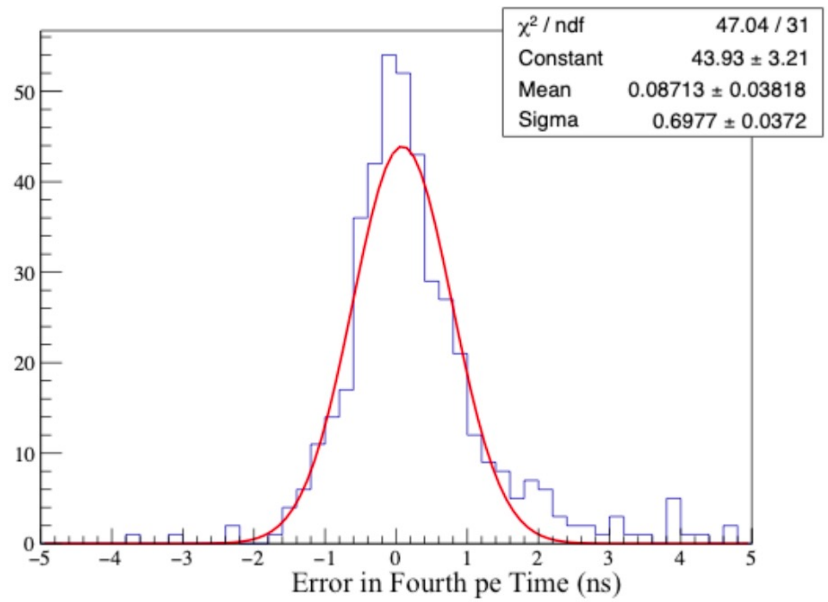
Comparison to Full Waveform for t_i

Uses real PMT waveforms with forced pileup

Modeled Analog Feature Extraction



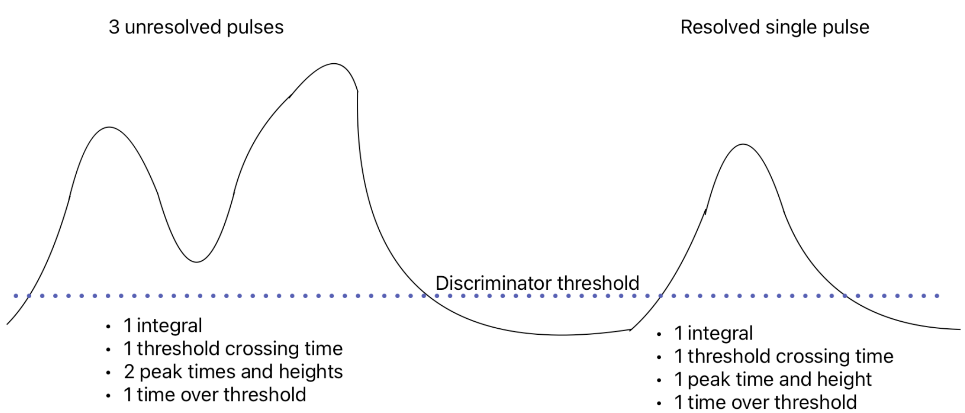
Iterative Matched Filter Approach on full 50 ps waveform



~900 ps precision vs. ~700 ps for 4th piled up pulse

Analog Photon Processor Requirements

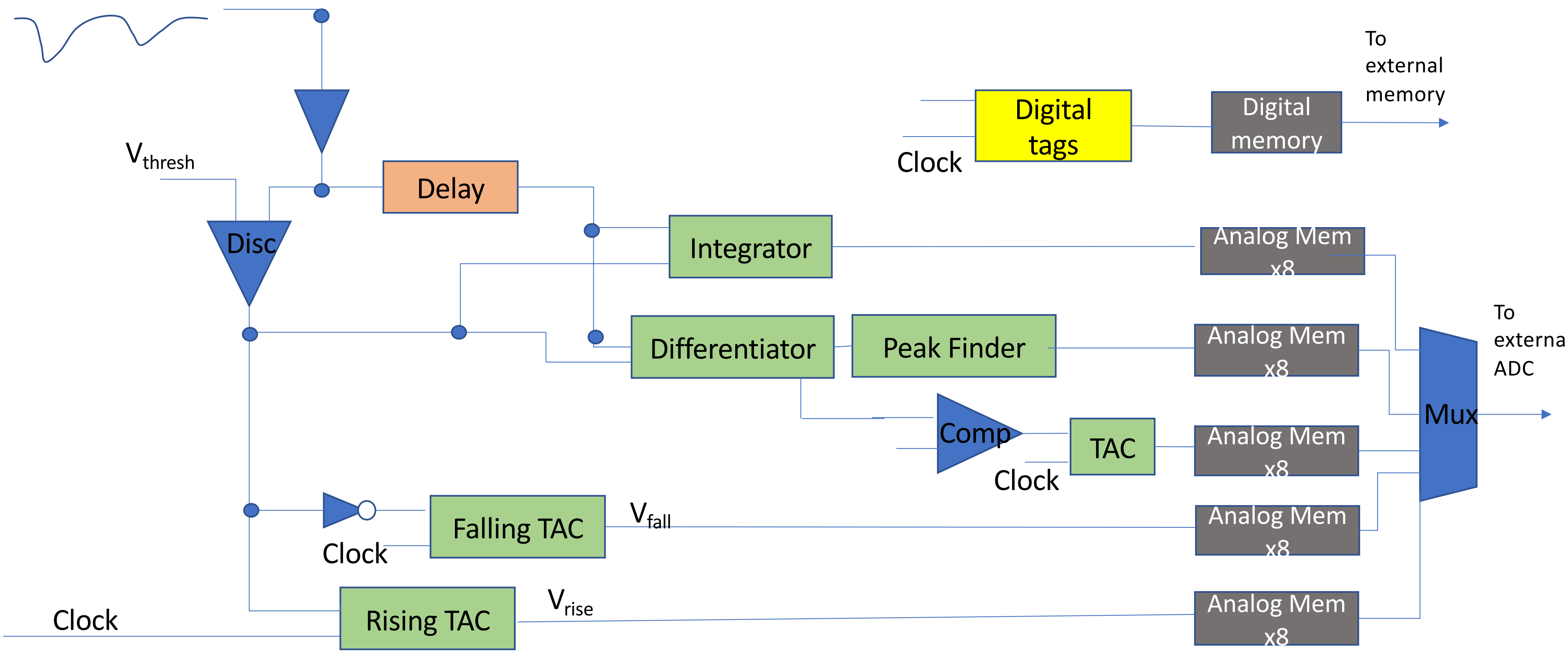
- Retriggerability – to create resolved pulse packets
  - In practice this will depend on realistic bandwidths
- Integrator dynamic range 100 pe (extendable with multiple gain paths)
  - INL <1% after calibration
- SNR > 10-20
- Peak detection for at least two peaks/resolved packet
  - Amplitude and time
- Time resolution and precision 20 ps for threshold crossing



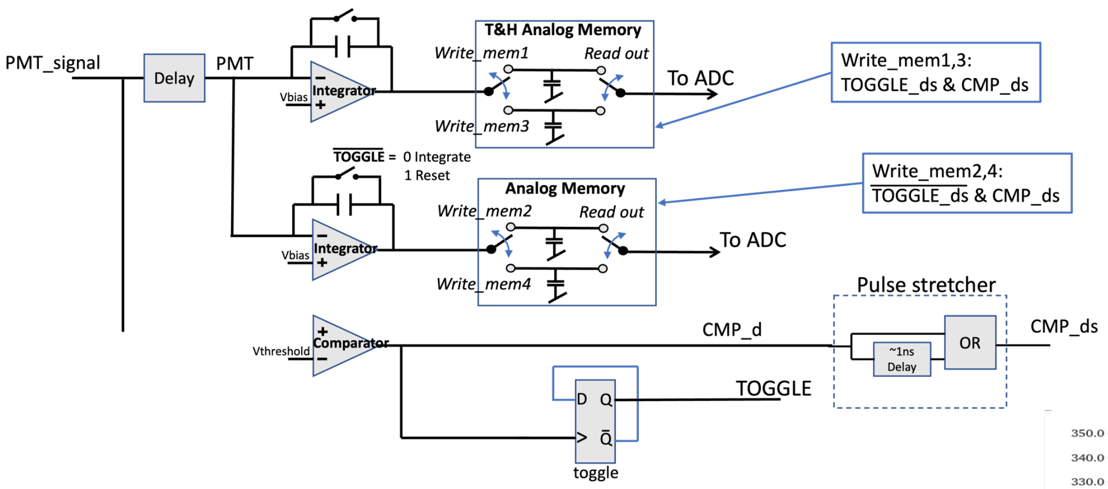
We have worked within TSMC's 65 nm process to ensure we are fast enough

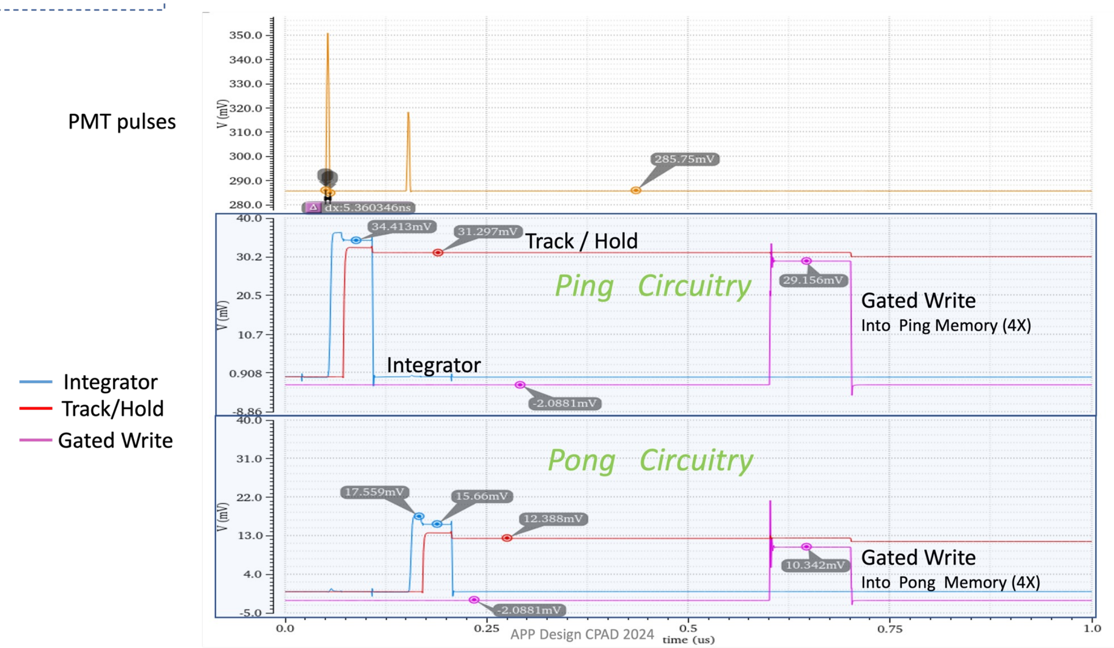
Analog Photon Processor ASIC

Feature Block Diagram

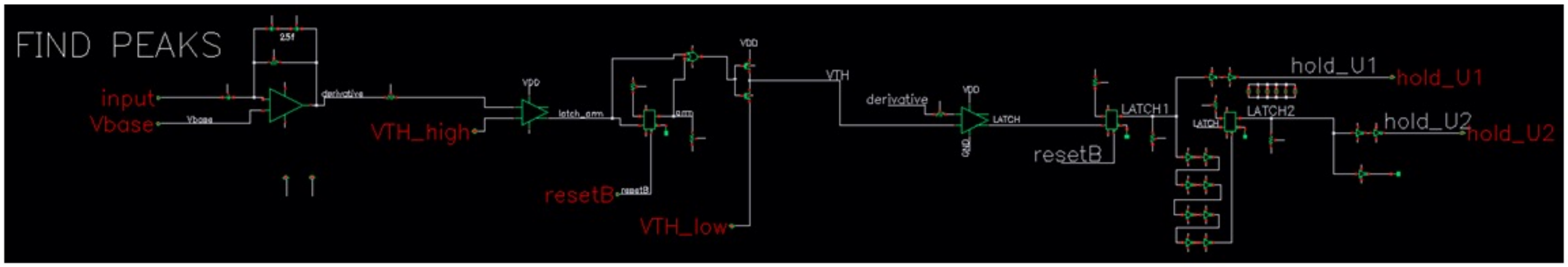


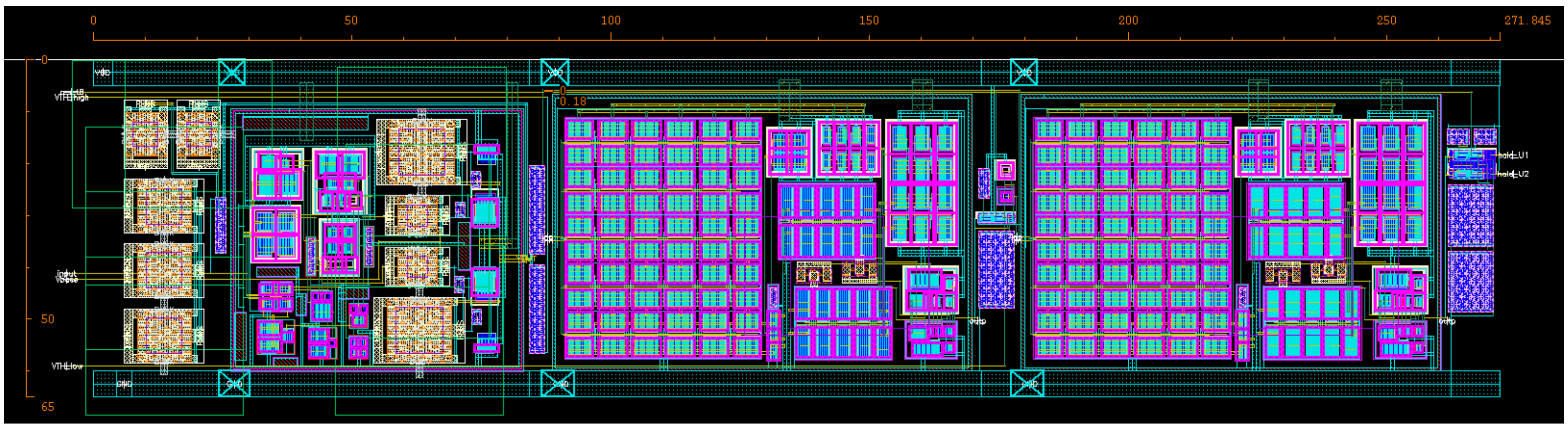
Retriggerability Using Ping-Pong Logic



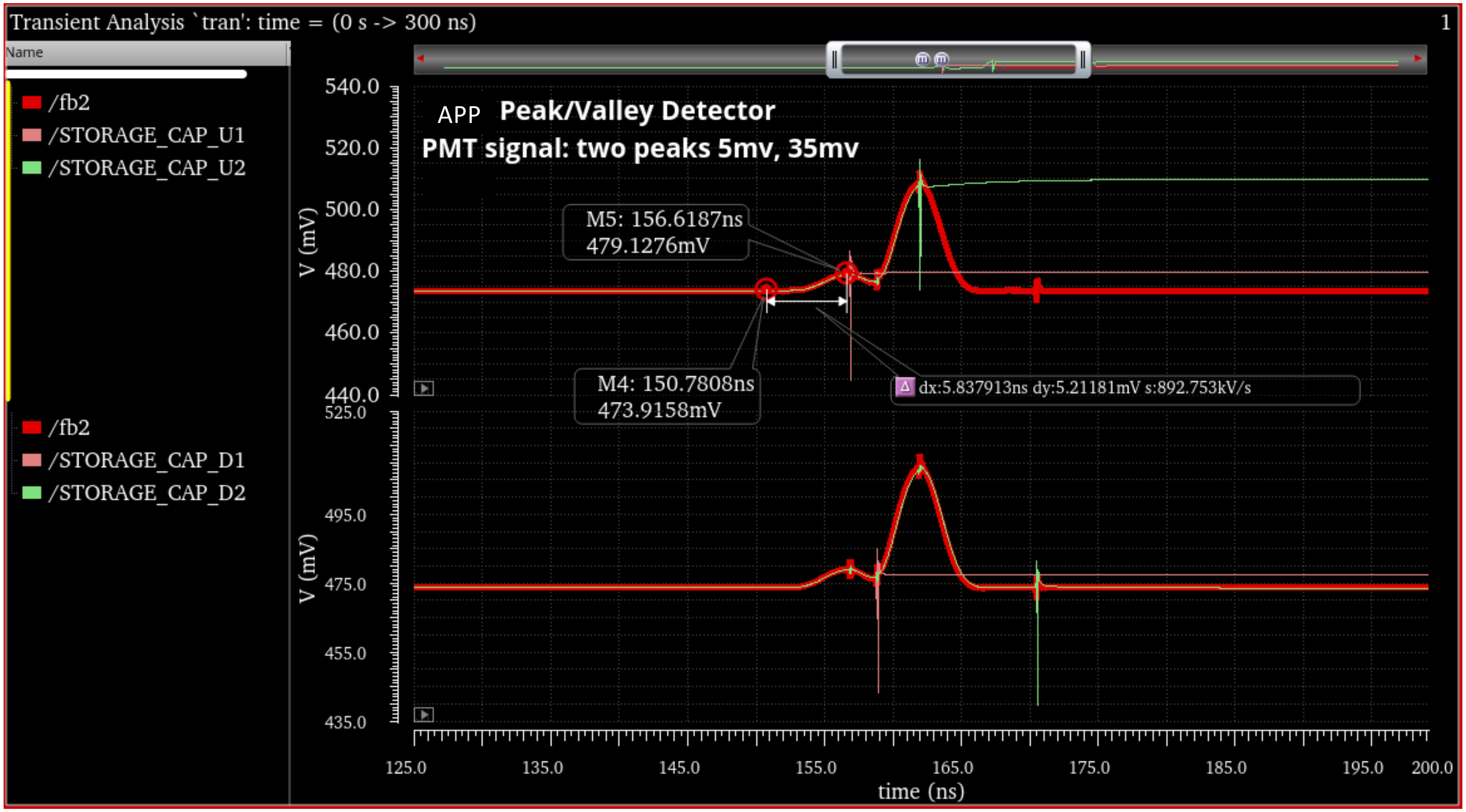


Find peak (Layout & schematic)



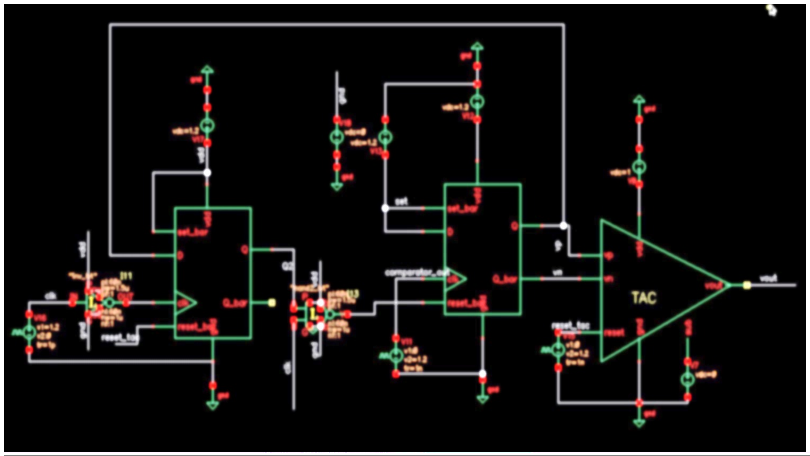


Peak Detector Full TSMC 65 nm Model Simulation



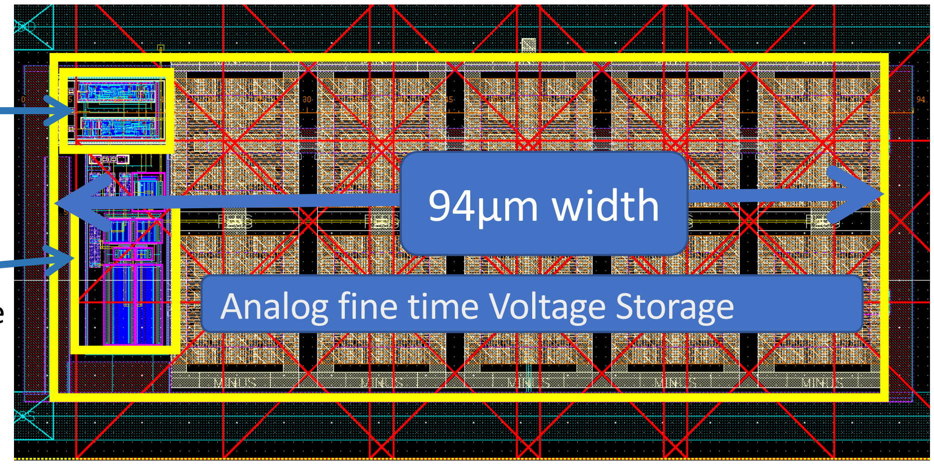
T A C -- Time to Analog Converter

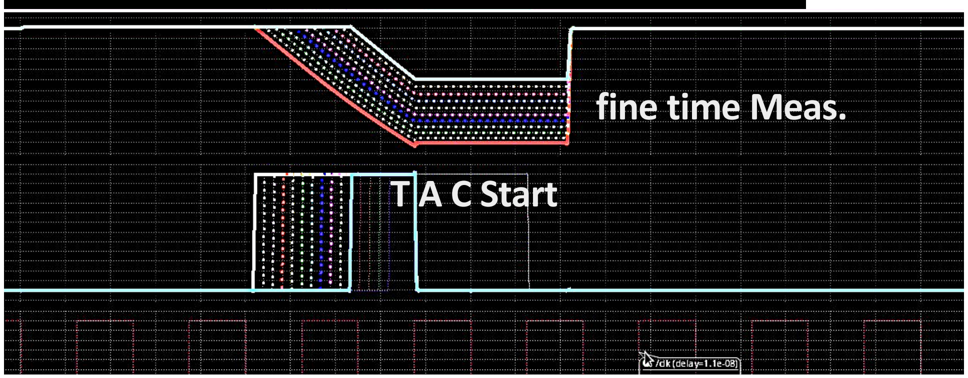
Constant Current Capacitor Discharge (½-1½ Clock pulses)



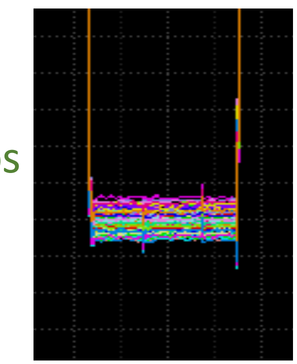
Digital Latch & Clocking Circuit

Analog Circuit Switches current source





Trans Noise for max *ft*(1.5CLK)
σ=158.17uV @ 13mV/ns ~ σ(t) 12ps



TSMC 65 nm simulation

Detailed Performance (TSMC 65 nm Model)

Discriminator vs. over-voltage (before calibration)

Input Override Voltage	Output		
Vth = 2.5mV	Delay of leading edge	Width	Rise time
2.5mV	2.37 ns	2.05 ns	454.5 ps
5mV	1.79 ns	3.27 ns	454.7 ps
7.5mV	1.57 ns	4.01 ns	452.7 ps
10 mV	1.44 ns	4.45 ns	454.2 ps
20mV	1.23 ns	5.68 ns	455.9 ps
50mV	1.04 ns	6.38 ns	460.3 ps
100 mV	932.7 ps	6.8 ns	463.8 ps
200mV	844.91 ps	7.44 ns	469.5 ps

Skew/walk will be calibrated

Integrator non-linearity (before calibration)

Input Voltage(mV)	|INL| (%)	Input Voltage(mV)	|INL| (%)
5	2.67	300	1.73
10	1.07	350	1.9
20	2.33	400	1.53
50	1.07	450	1.51
100	0.53	500	2.21
150	1.07	550	1.67
200	1.133	600	2.51
250	1.6		

APP Block Status: November 2024

	Schematic Design	Schematic simulations	Layout
Peak detector	done	done	Near completion
Discriminator	done	done	done
Integrator	done	ongoing	Started
Analog Delay	done	done	Near completion
TAC	done	done	done
Analog Memory (Track and Hold)	done	done	ongoing
Analog Mux	tbd		
Fully Differential Opamp	done	done	done
LVDS Driver	tbd		
LVDS Receiver	tbd		
DAC	done	done	done
Compiled Logic	Started		

First tapeout is possible by Spring 2025 (4 channels)

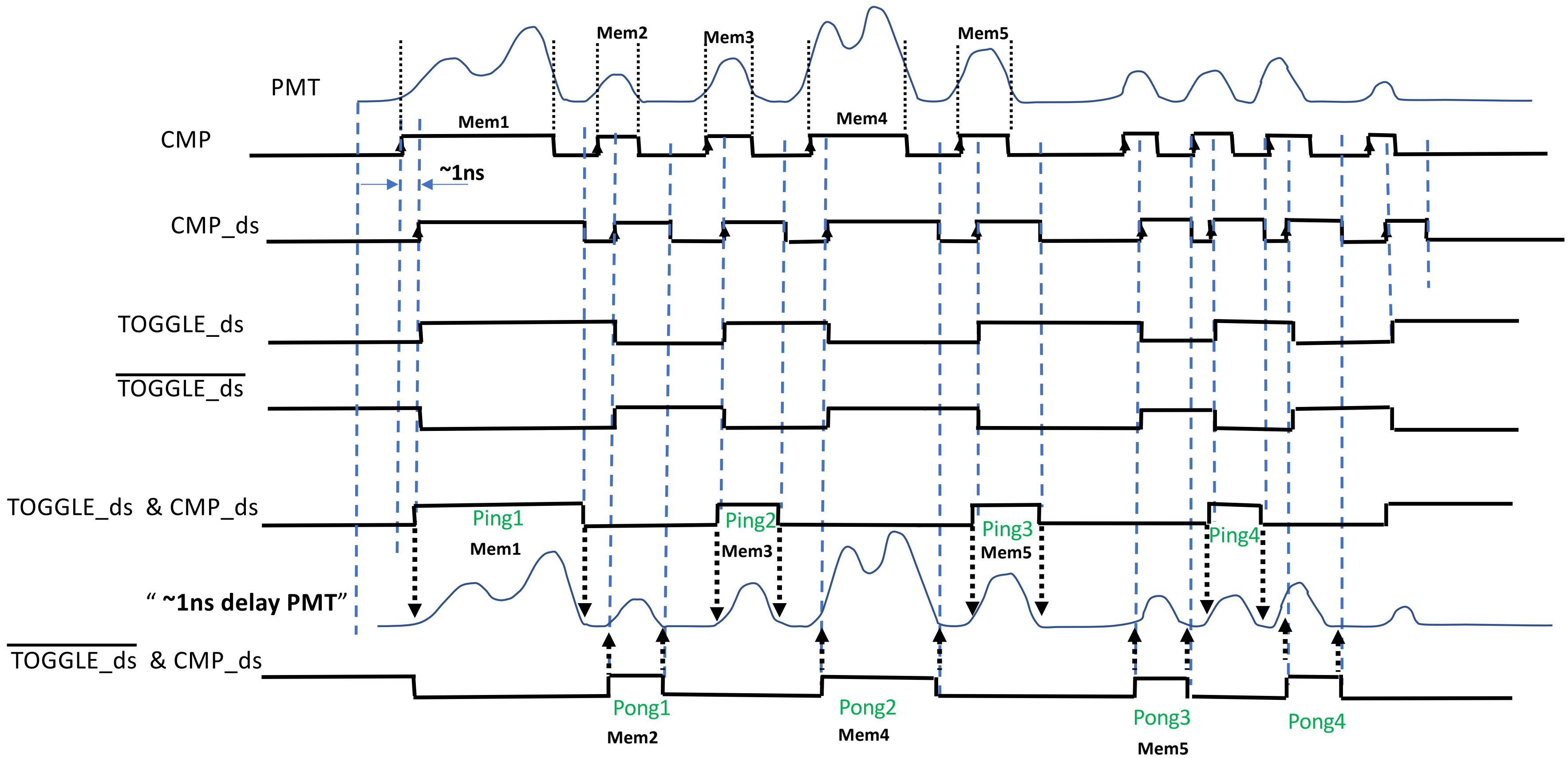
Summary

- Analog feature extraction allows measurements of $N_{photons}$ and t_i in large ν detectors
- Much smaller data volume
- Lower cost
- Lower power
- Wider bandwidth than fast sampling yields improved resolution on t_i
- With faster PMTs and hybrid Cher/scint detectors being designed, can provide a better front-end solution than thousands of WFD channels

Funded by DOE/HEP award number DE-SC0007901

Backups

Timing Diagram for the Dead timeless Ping Pong Integrate & Hold



Analog Photon Processor:
ping-pong logic for
Integrated Time-Over-Threshold
8/27/2024 Penn

CMP_ds = Delayed and stretched w.r.t CMP

TOGGLE_ds = Delayed and stretched w.r.t CMP

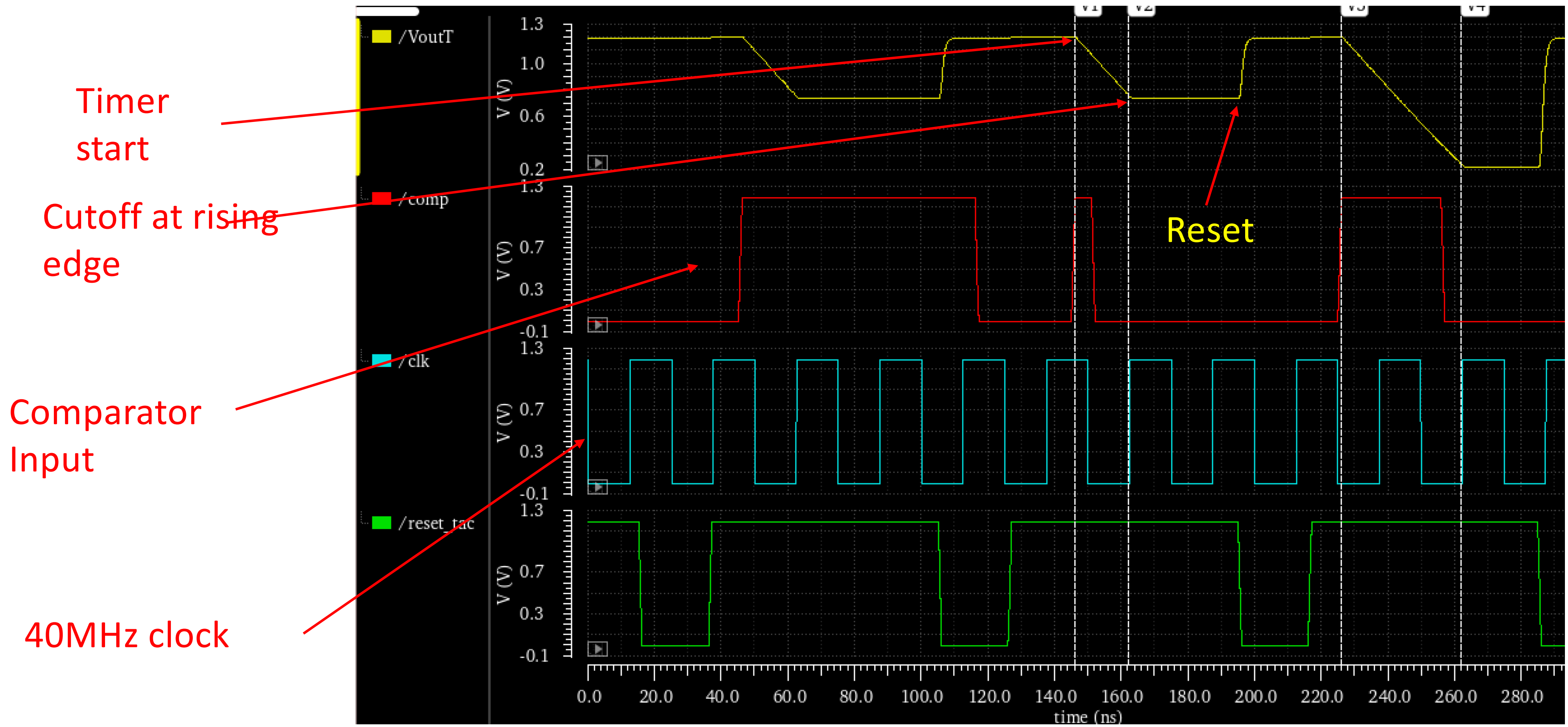
TOGGLE_ds & CMP_ds = 'Ping' Write

$\overline{\text{TOGGLE\_ds}}$ & CMP_ds = 'Pong' Write

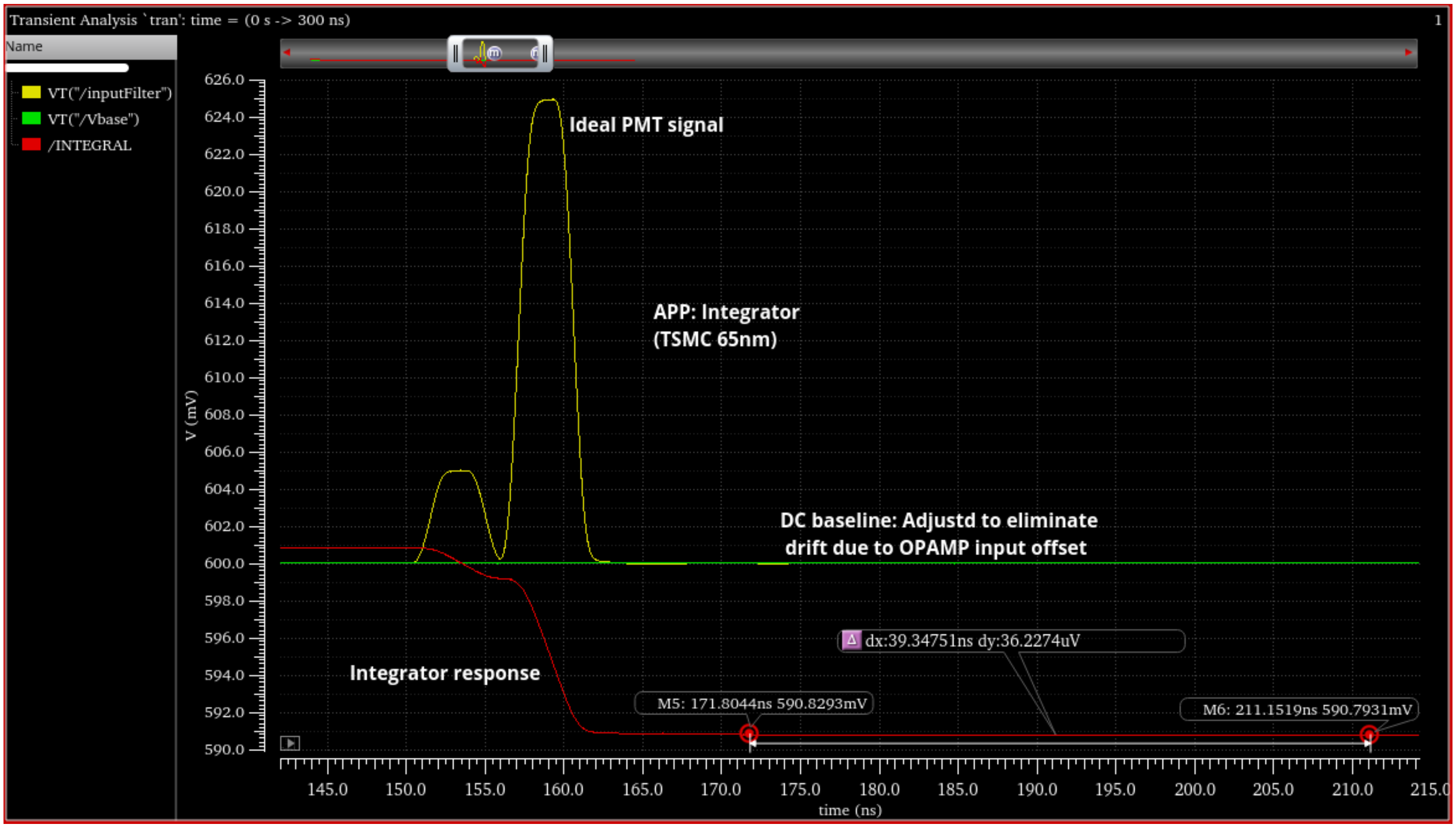
Time-to-Amplitude Converter (TAC)

65 nm TSMC Model

12 bits yields < 20 ps timing resolution

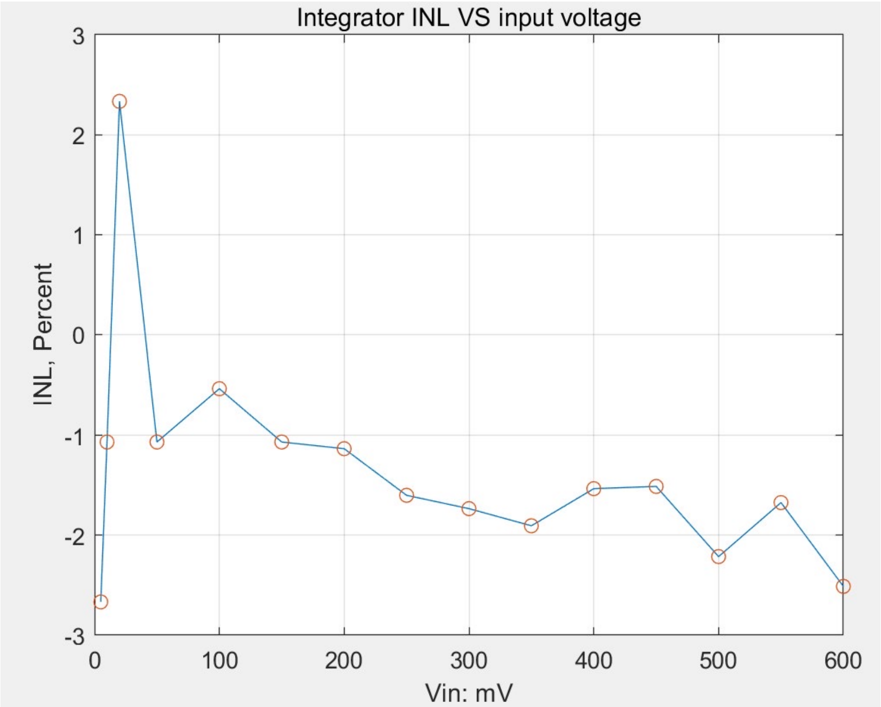


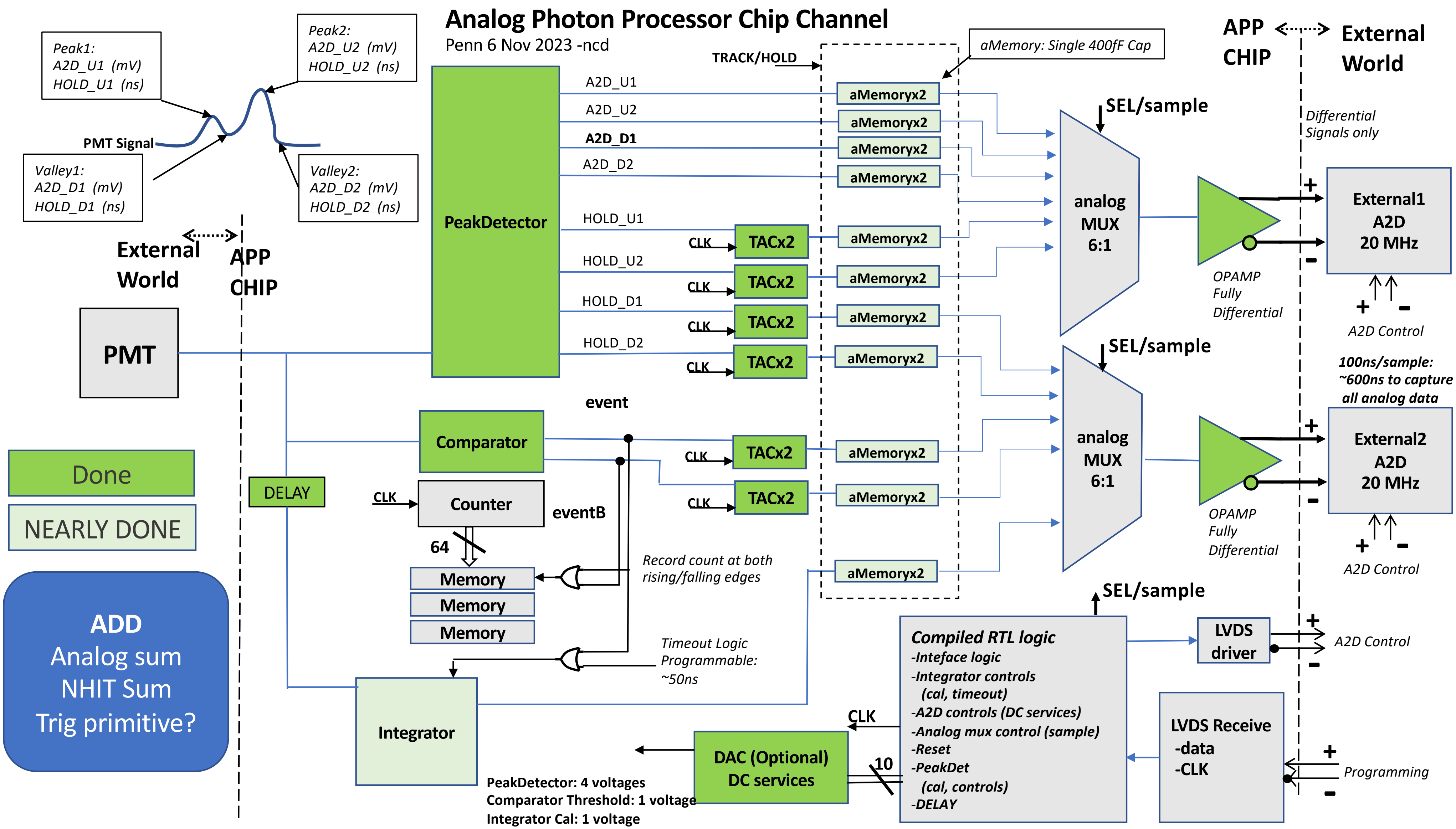
Integrator Full TSMC 65 nm Model Simulation



Integrator Non-linearity (Before Calibration)

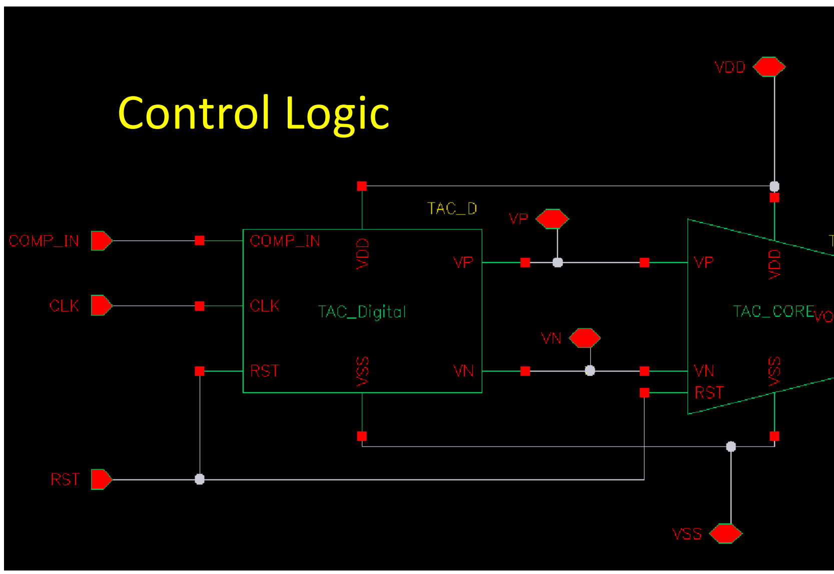
Input Voltage(mV)	|INL| (%)	Input Voltage(mV)	|INL| (%)
5	2.67	300	1.73
10	1.07	350	1.9
20	2.33	400	1.53
50	1.07	450	1.51
100	0.53	500	2.21
150	1.07	550	1.67
200	1.133	600	2.51
250	1.6		

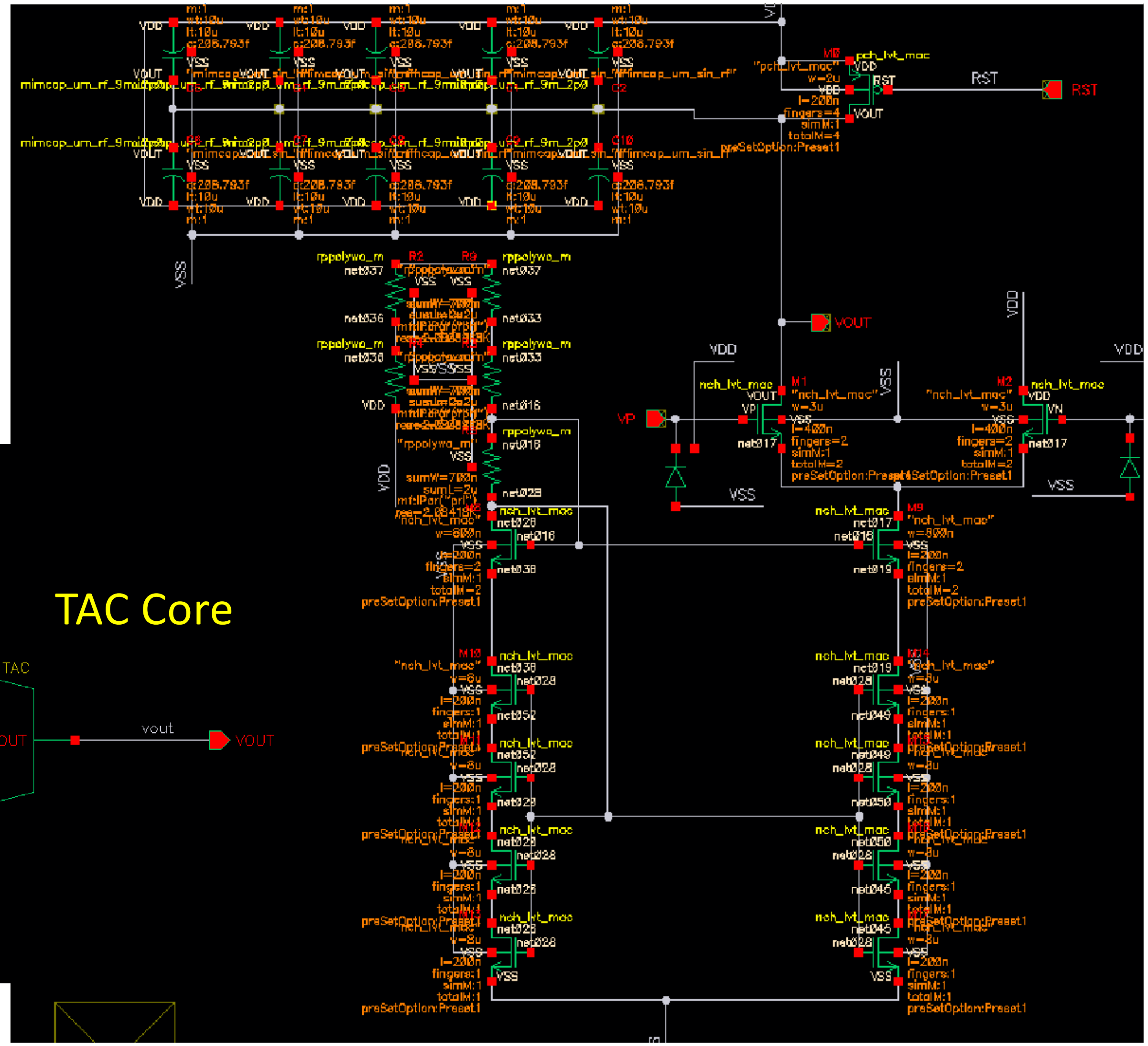




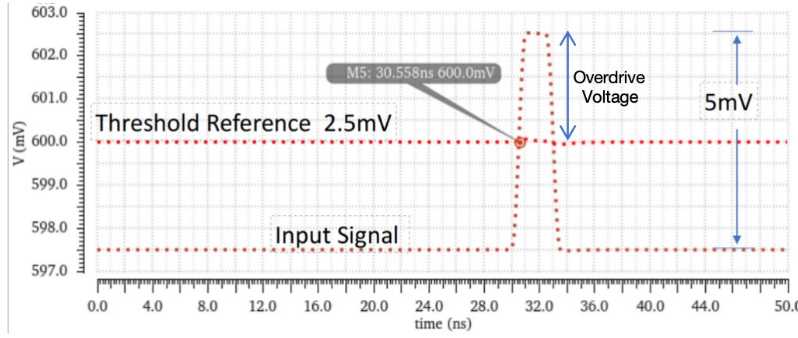
Analog Photon Processor Chip Channel
Penn 6 Nov 2023 -ncd
Peak1: A2D_U1 (mV) HOLD_U1 (ns)
Peak2: A2D_U2 (mV) HOLD_U2 (ns)
Valley1: A2D_D1 (mV) HOLD_D1 (ns)
Valley2: A2D_D2 (mV) HOLD_D2 (ns)
PMT Signal
External World
APP CHIP
PMT
Done
NEARLY DONE
ADD
Analog sum
NHIT Sum
Trig primitive?
DELAY
PeakDetector
A2D_U1
A2D_U2
A2D_D1
A2D_D2
HOLD_U1
HOLD_U2
HOLD_D1
HOLD_D2
TRACK/HOLD
CLK
TACx2
aMemoryx2
aMemory: Single 400fF Cap
event
Comparator
eventB
Counter
64
Memory
Memory
Memory
Record count at both rising/falling edges
Timeout Logic Programmable: ~50ns
Integrator
DAC (Optional) DC services
10
PeakDetector: 4 voltages
Comparator Threshold: 1 voltage
Integrator Cal: 1 voltage
SEL/sample
analog MUX 6:1
OPAMP Fully Differential
APP CHIP
External World
Differential Signals only
External1 A2D 20 MHz
External2 A2D 20 MHz
A2D Control
100ns/sample: ~600ns to capture all analog data
Compiled RTL logic
-Inteface logic
-Integrator controls (cal, timeout)
-A2D controls (DC services)
-Analog mux control (sample)
-Reset
-PeakDet (cal, controls)
-DELAY
LVDS driver
LVDS Receive
-data
-CLK
Programming

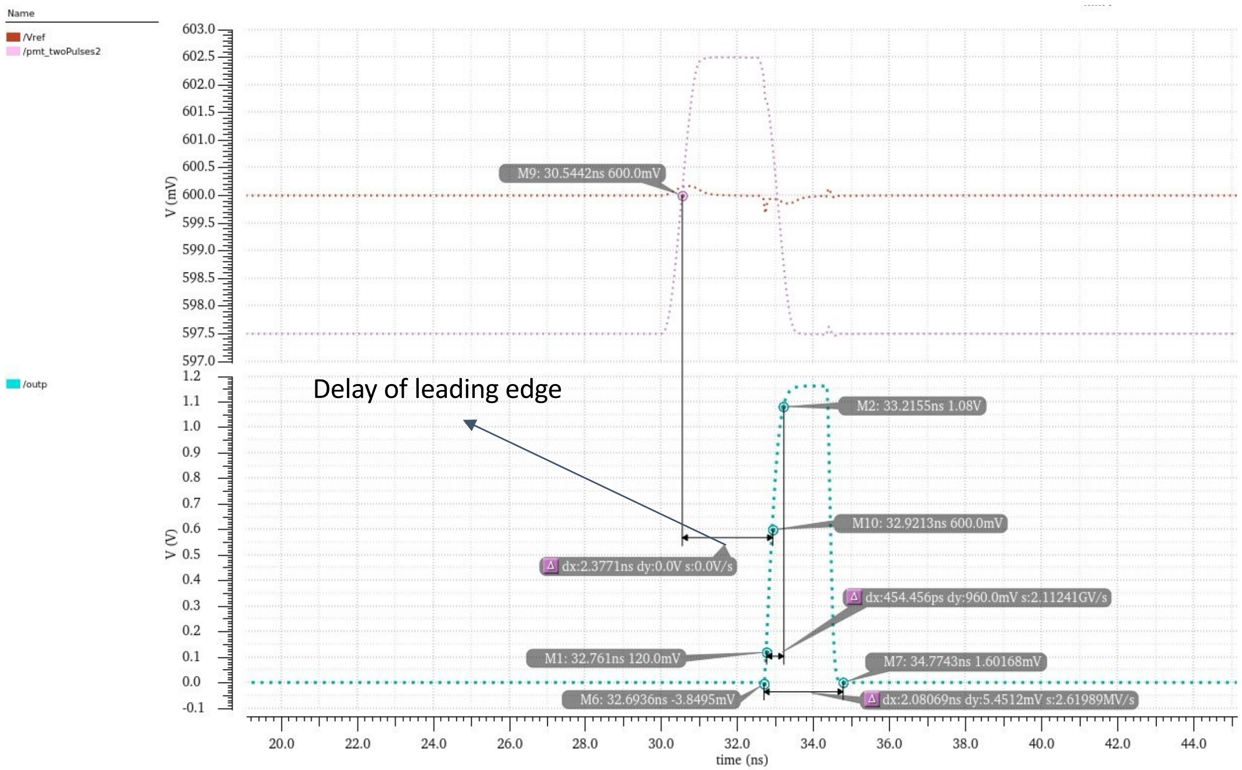
TAC (schematic)



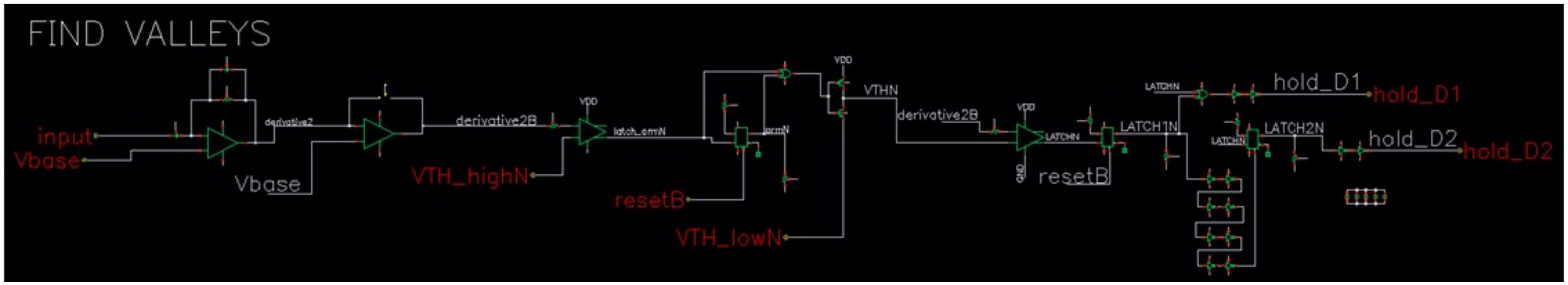


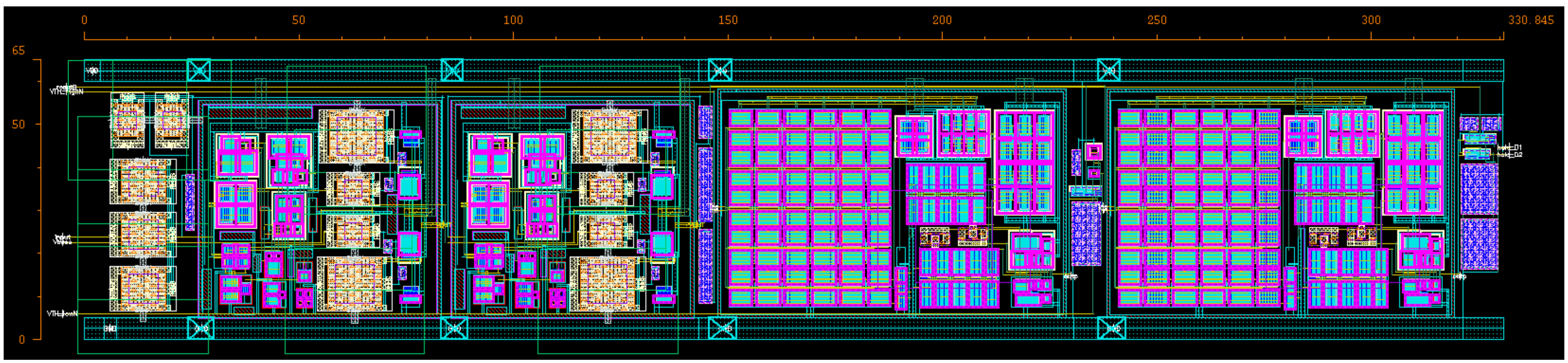
Comparator(Simulation)

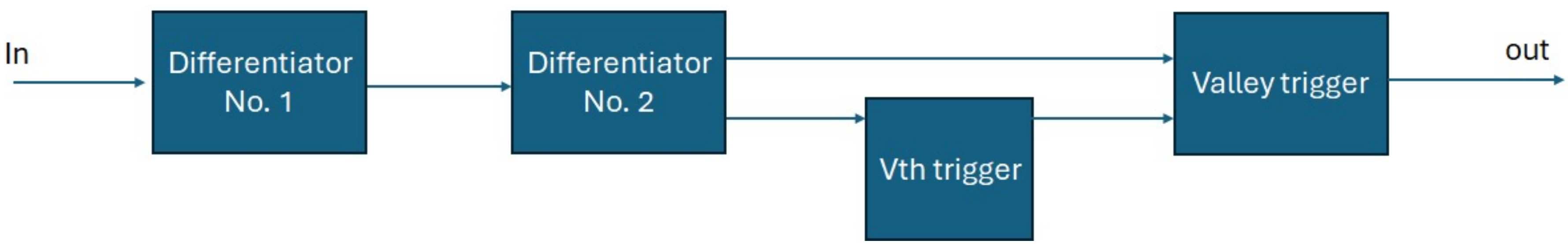


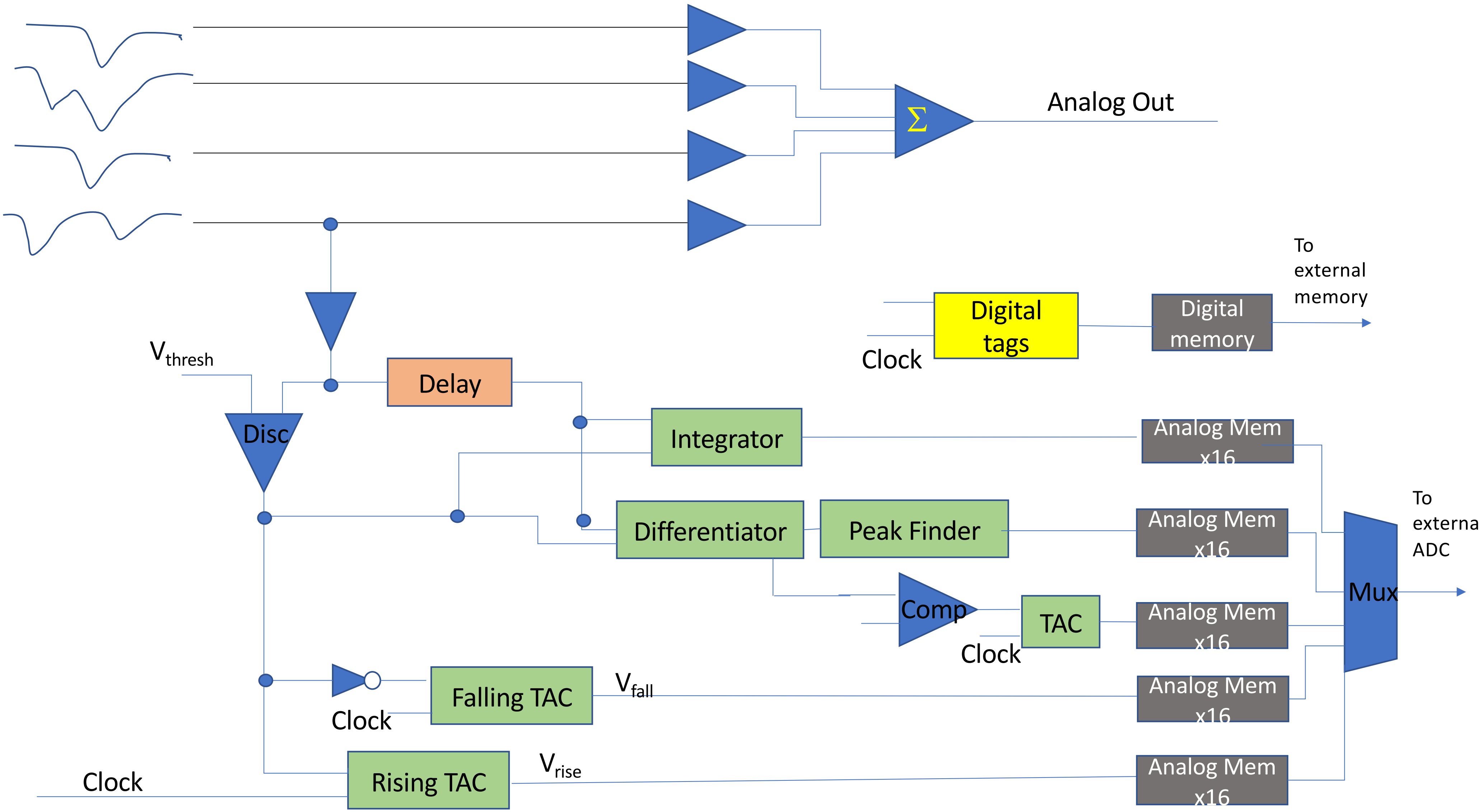


Find valley (Layout & schematic)



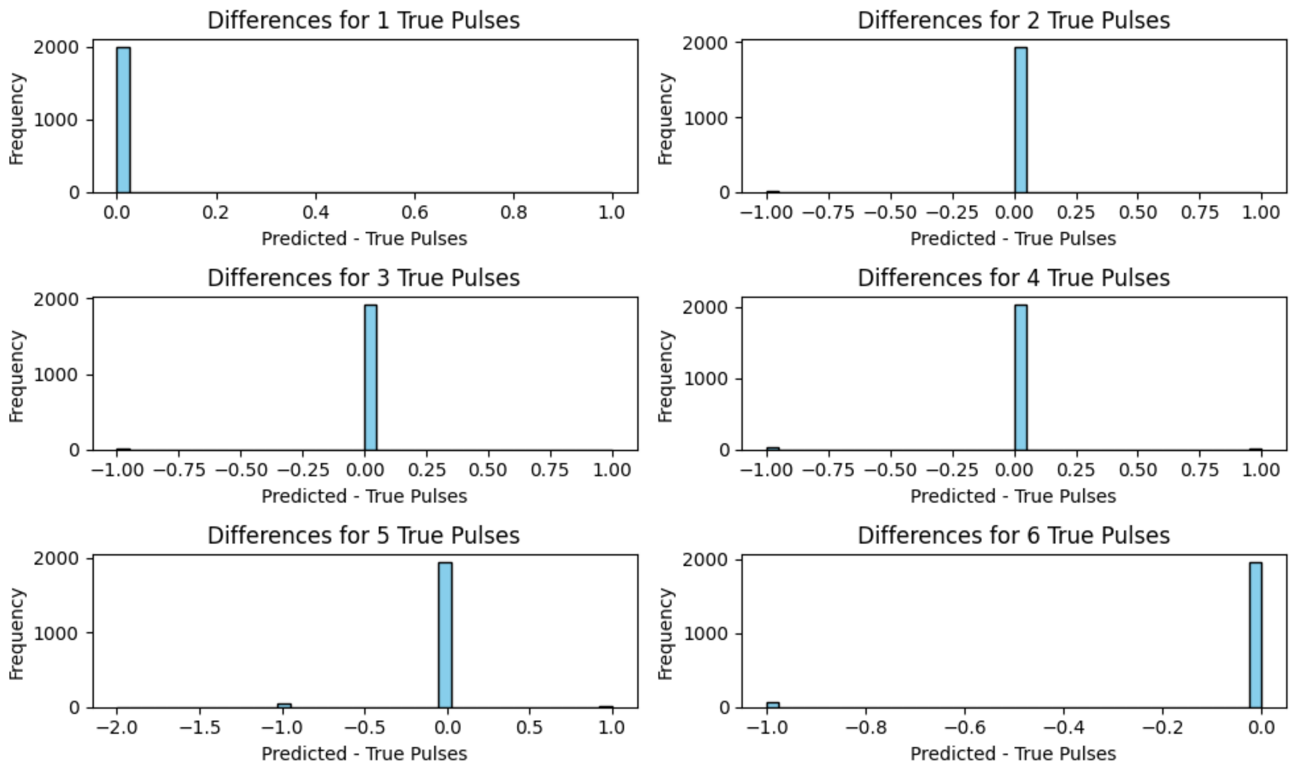






Analog Out
To
external
memory
Digital
tags
Digital
memory
Clock
V
thresh
Delay
Disc
Integrator
Analog Mem
x16
Differentiator
Peak Finder
Analog Mem
x16
To
external
ADC
Mux
Comp
TAC
Analog Mem
x16
Clock
Falling TAC
V
fall
Analog Mem
x16
Clock
Clock
Rising TAC
V
rise
Analog Mem
x16

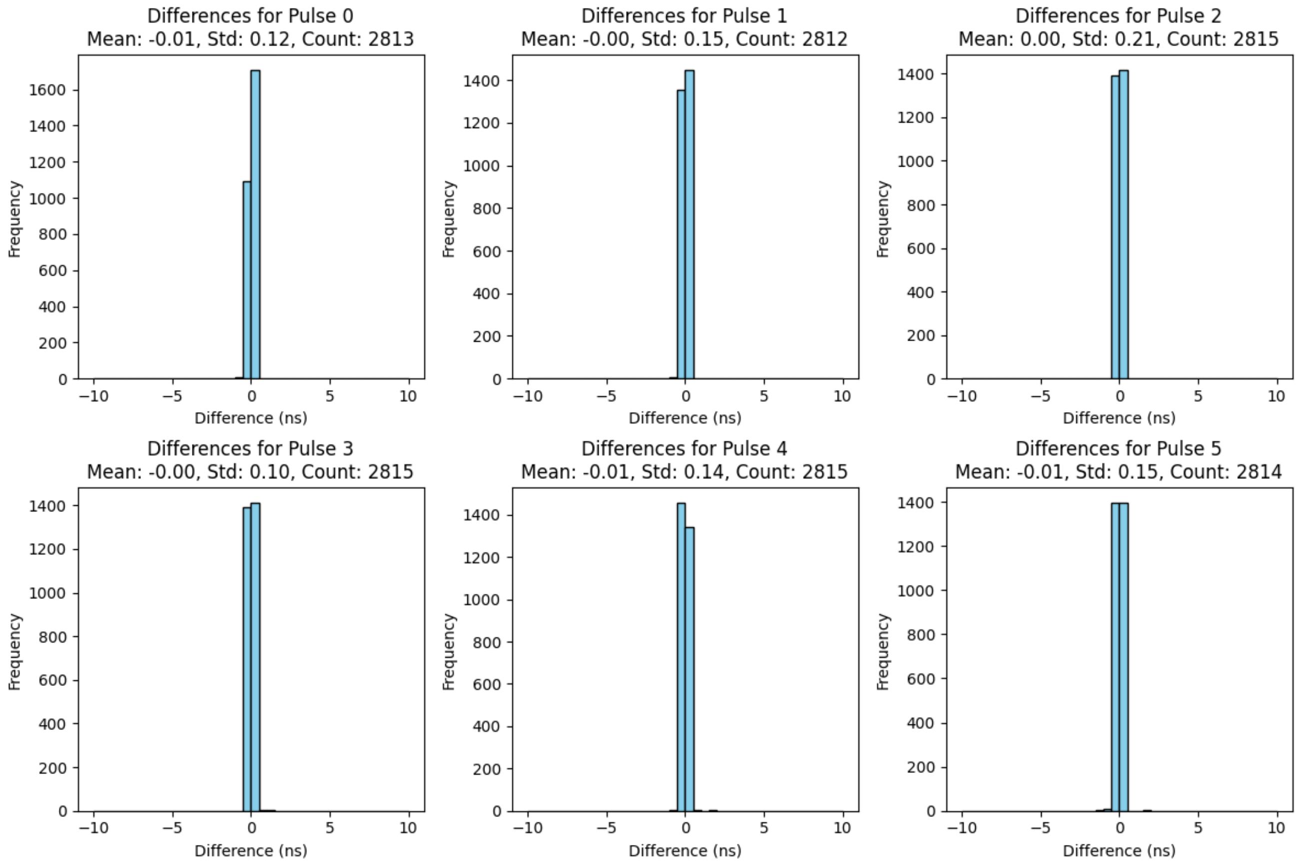
Performance on Predicting N



No surprise that it does extremely well here

Performance on Predicting t_i

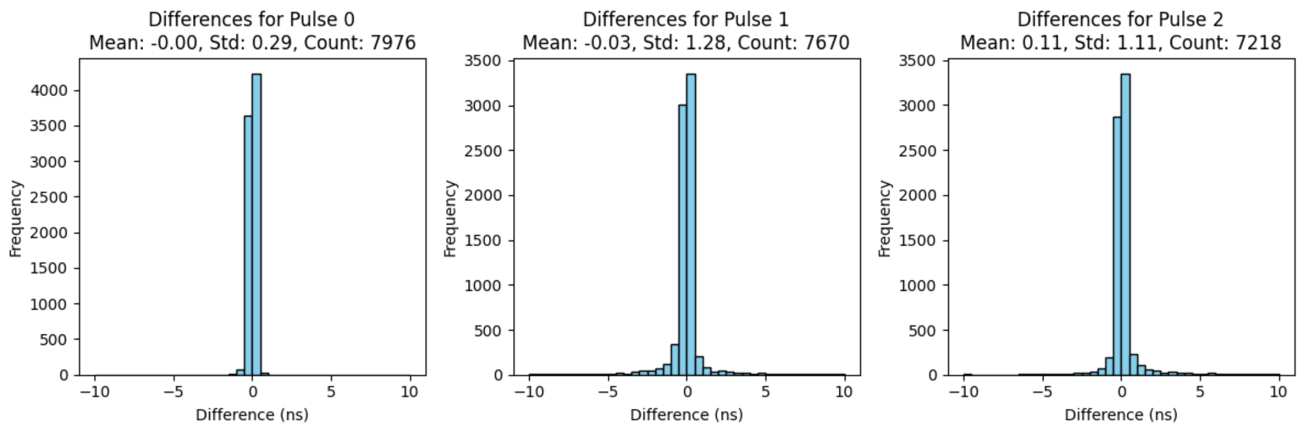
PMT-like pulses
with σ=2.5 ns



Only cases where all 6 pulses are in resolved packets (no pileup)

Performance on Predicting t_i

Real PMT pulses



Algorithm

How well can you do for resolved+unresolved pulses (distributed with LAB-PPO time profile)?

- This is what I used:

```
# Initialize the MultiOutputRegressor with RandomForestRegressor
model = MultiOutputRegressor(RandomForestRegressor(n_estimators=100, random_state=4
2))
```

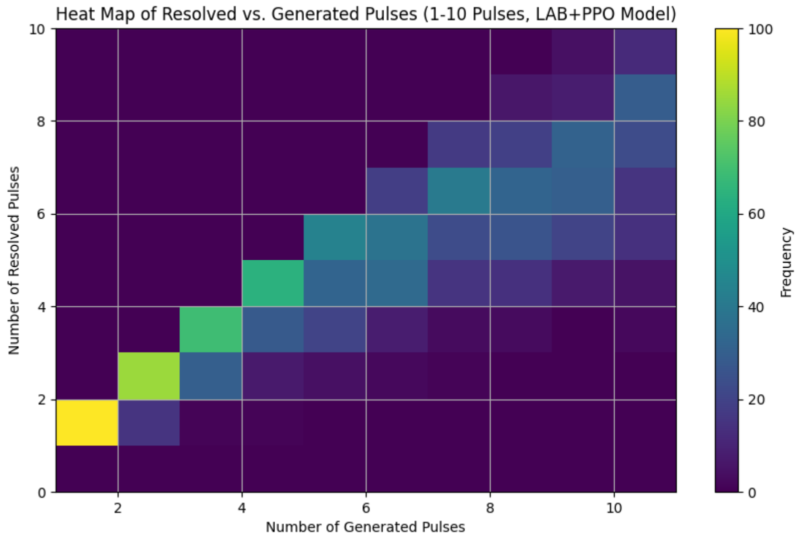
```
# Train the model
model.fit(X_train, y_train)

# Make predictions and evaluate
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error: {mse}")
```

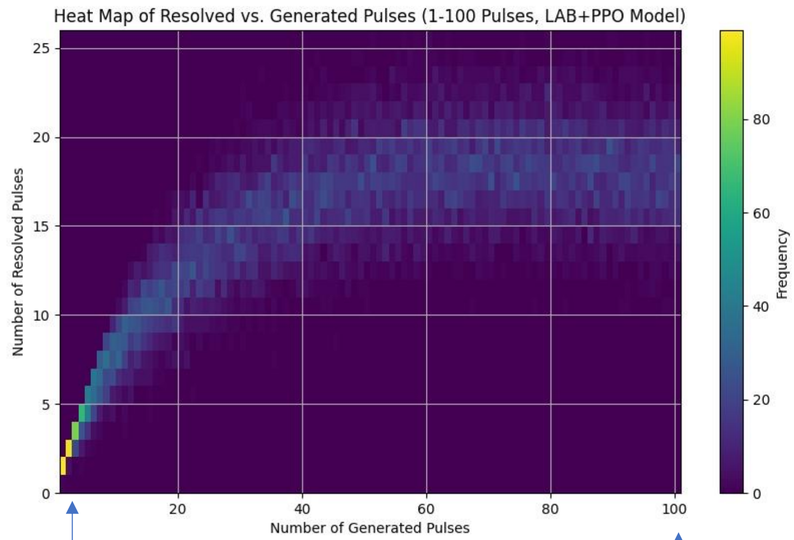
- X_train is the combined pulse data---our features (peak, integral, etc.)
- y_train are the “labels” (the true times)
- y_test is the sample to test the predictions
- y_pred are the actual predictions
- I know nothing about how RandomForest works and I really don’t care right now

Times of Resolved Packets are Much Better

How Many Resolved Packets vs. Total Photons?

PMT-like pulses with 2.5 ns σ



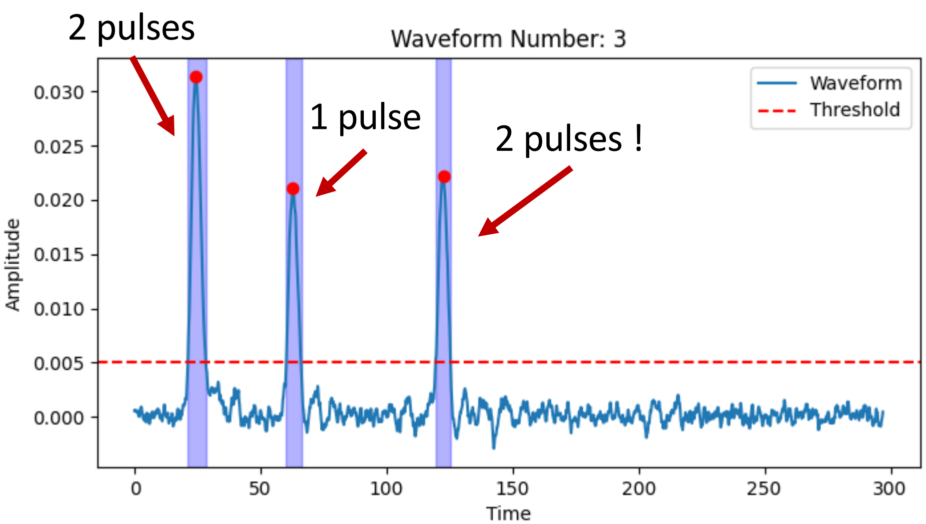


Light Yield for Solar ν in Theia

Light Yield for 1 GeV Muon going through Theia

Feature Extraction Looks Good

Real R14688 Pulses (added together)




```
Waveform Truth Info:
  Waveform Number: 3
  Number of Pulses: 6
  Mean Times of Pulses: [100.81012658 101.97046414  41.07594937   3.57383966   2
.45991561
   8.91139241]

Waveform Data:
  Number of Resolved Packets: 3
Packet Info:
    Packet:
      Start Time: 21.11814345991561
      Integral: 0.1414729926058468
      Time Over Threshold: 7.147679324894515
      Number of Peaks: 1
      Peak:
        Time: 24.27426160337553
        Height: 0.03138168921577744

Packet Info:
    Packet:
      Start Time: 60.198312236286924
      Integral: 0.08750698247860653
      Time Over Threshold: 5.848101265822777
      Number of Peaks: 1
      Peak:
        Time: 62.84388185654009
        Height: 0.021134238806553185

Packet Info:
    Packet:
      Start Time: 119.42194092827005
      Integral: 0.09267808521570586
      Time Over Threshold: 5.987341772151893
      Number of Peaks: 1
      Peak:
        Time: 122.53164556962025
        Height: 0.02222540555158048
```

But note that noise increases like sqrt(N)because of pulse addition