HATS = Hall A Trigger Simulation

- Technical Details:
  - General Structure
  - Signal Generation (physics)
  - Signal Processing (electronics)
  - Data analysis
- Results compared with data:
  - Overlap rates
  - Tagger deadtime
  - T1 deadtime

.

.

.

General Structure

- ROOT/C++ Design;
- A lot of techniques learned from HAMC
- Simulates standard electronics (DAQ).
- Being used by PVDIS. Easy to adapt for other DAQ system.

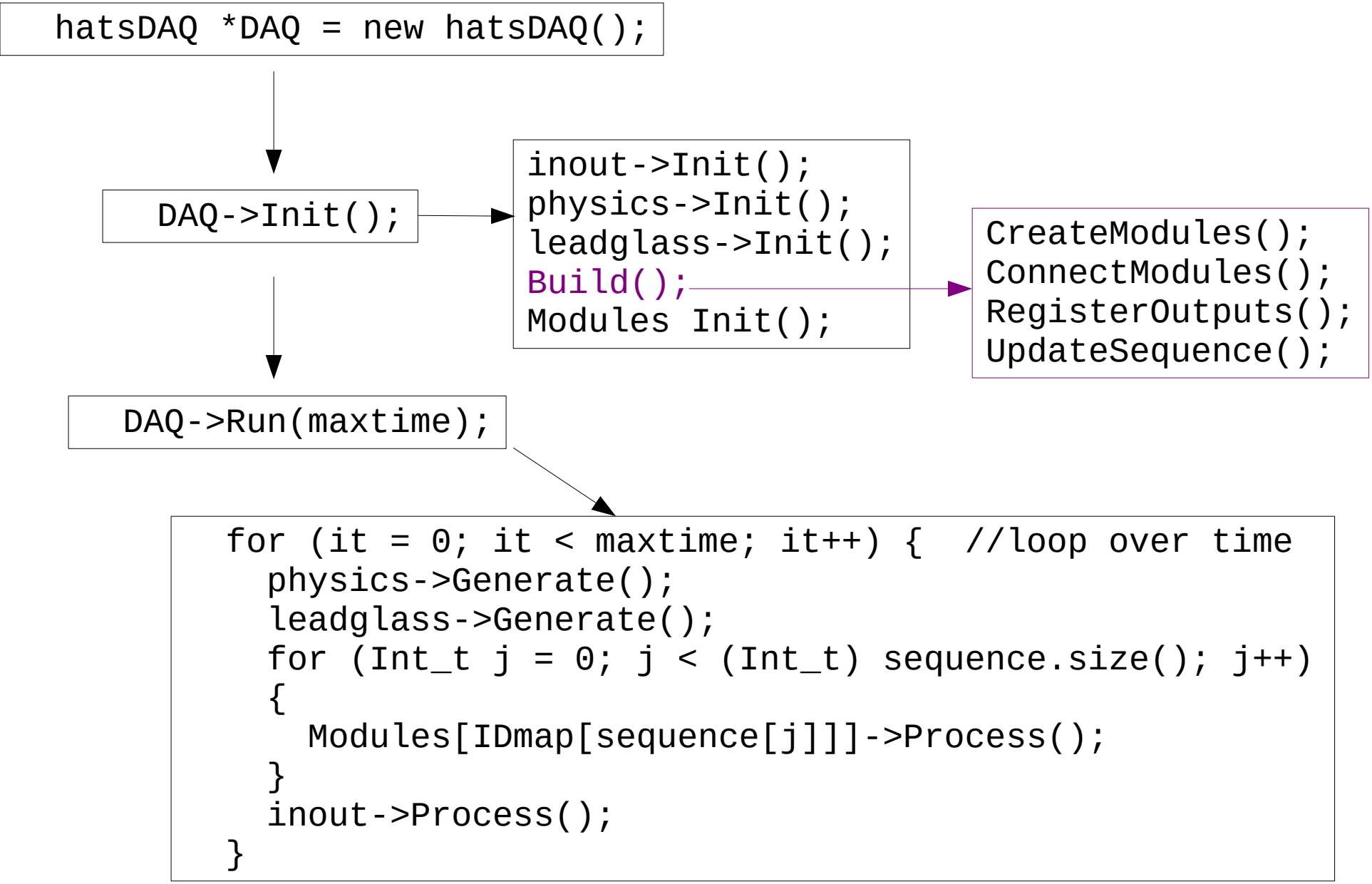
At every time instance (1ns), **Physics** information is generated. **Detectors** (**Leadglass**, **Gas Cerenkov**, **T1** ...) simulates the detector response and generates signals, which are processed by the **DAQ** system (constructed by **Modules**). **I/O** controls input and output.

classes:

- hatsPhysics: generates physics.
- hatsLeadglass, hatsGasCer ..... (abstract class hatsDetector not done yet)
- hatsModules: abstract class

  Derived classes: hatsPhi757, hatsPhi758, hatsSum8, hatsSplitter .....
- hatsDAQ: contains physics, detectors, modules, inout.
- hatsInout: deals with input and output.

General Structure

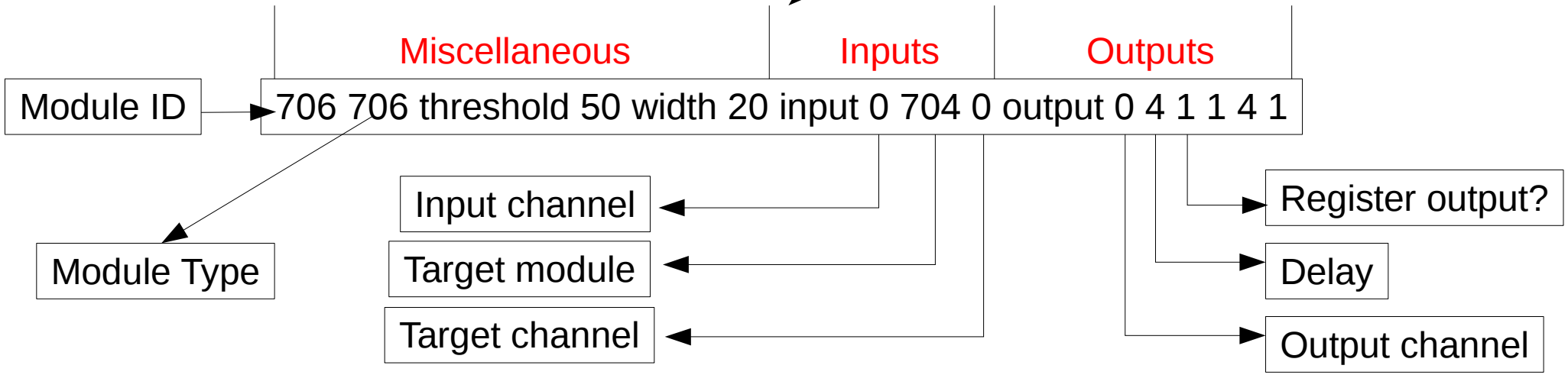


Input and Output

hats.dat:

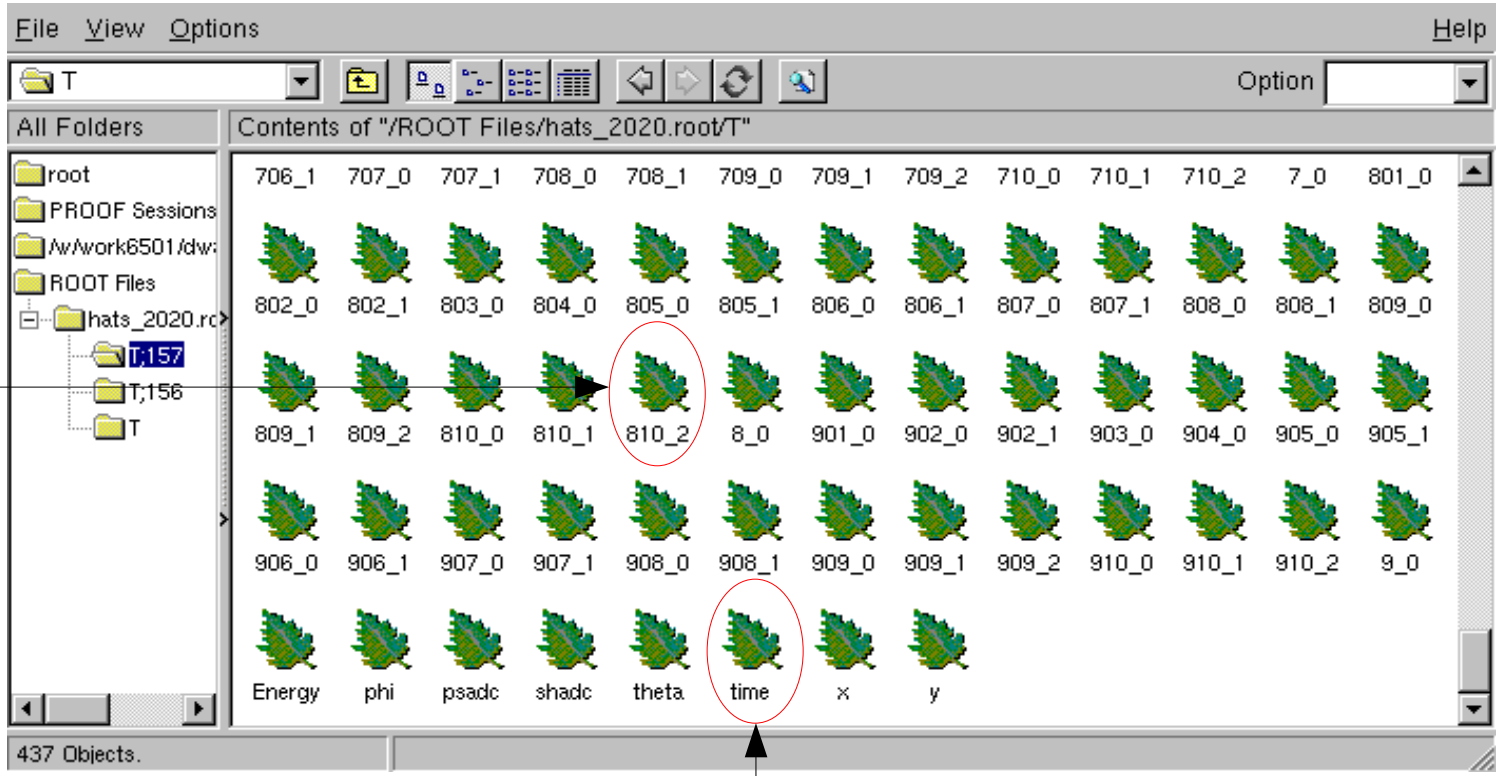
```
# input files
input_files rootfile input.root daqmap DAQmap.dat
rate 1000000
#adc to signal converting coefficient
convert_coeff ps 56.5 sh 208.6
                .............
```

```
.........
704 428F offset 0.0 input 0 702 1 output 0 16 1
705 706 threshold 100 width 20 input 0 703 0 output 0 4 1 1 4 1
706 706 threshold 50 width 20 input 0 704 0 output 0 4 1 1 4 1
707 706 threshold 100 width 100 input 0 705 0 output 0 16 1 1 4 1
                .........
```



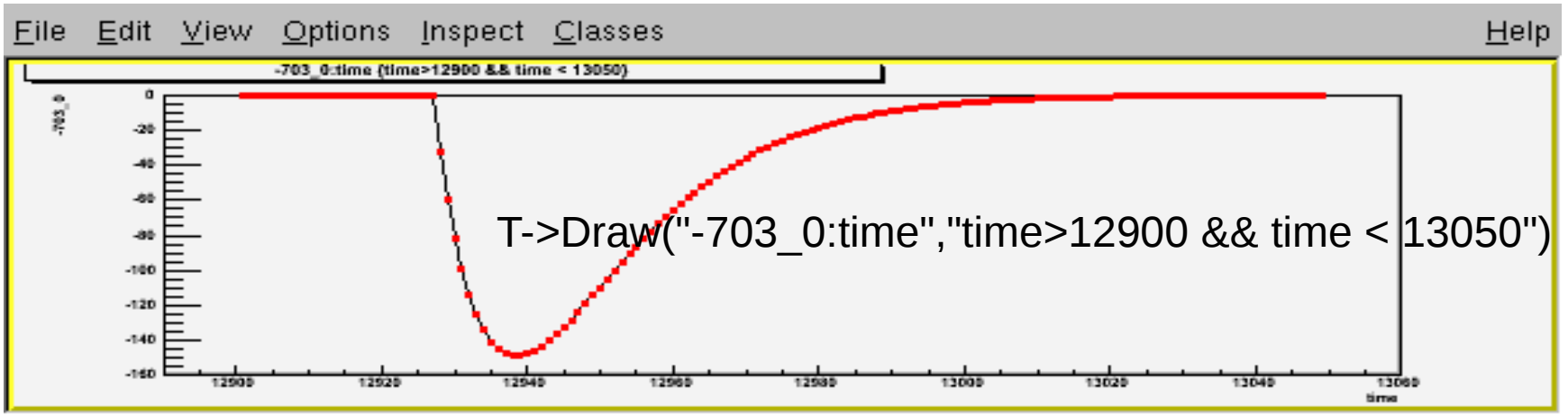
Input and Output

A typical output channel
Module 810, channel 2.



The time branch, can plot any output channel versus time

eg.



Signal Generation

Currently, getting ADC signal from data

Converting coefficients from ADC to analog signal

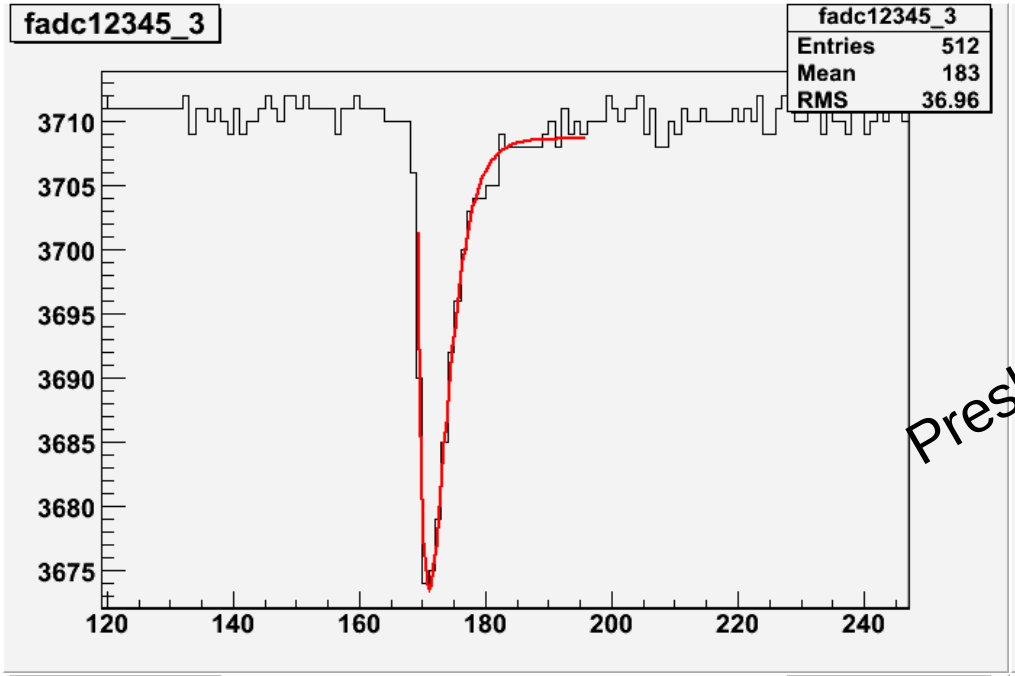
$$Analog\ signal = A \cdot t \cdot e^{-\frac{t}{\tau}},\ where\ A\ is\ related\ to\ energy\ deposit$$

$$threshold = A \cdot \frac{\tau}{e}$$

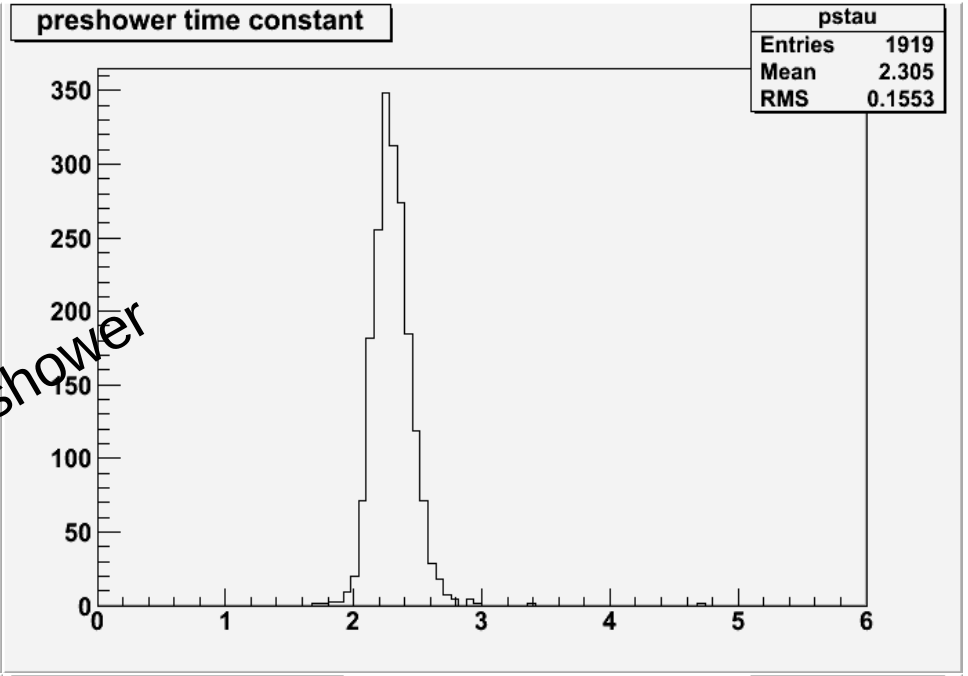
$$cut\ on\ ADC = k \int A \cdot t \cdot e^{-\frac{t}{\tau}} = k \cdot A \cdot \tau^2$$

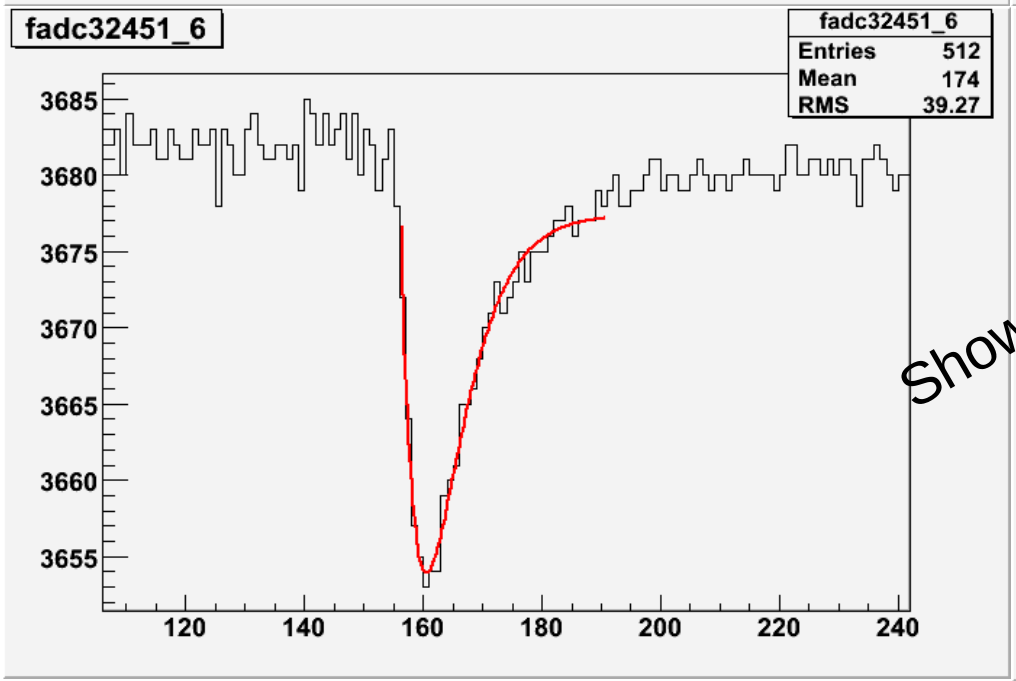
$$\rightarrow \quad k = \frac{cut\ on\ ADC}{e \cdot \tau \cdot threshold}$$

Time Constants: FADC Calibration

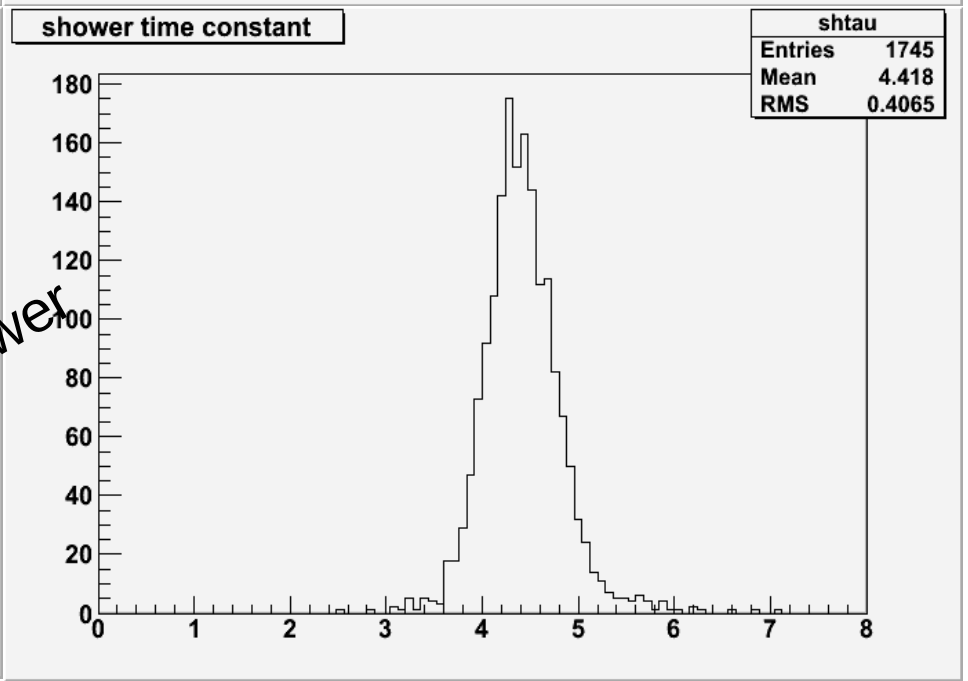


Preshower

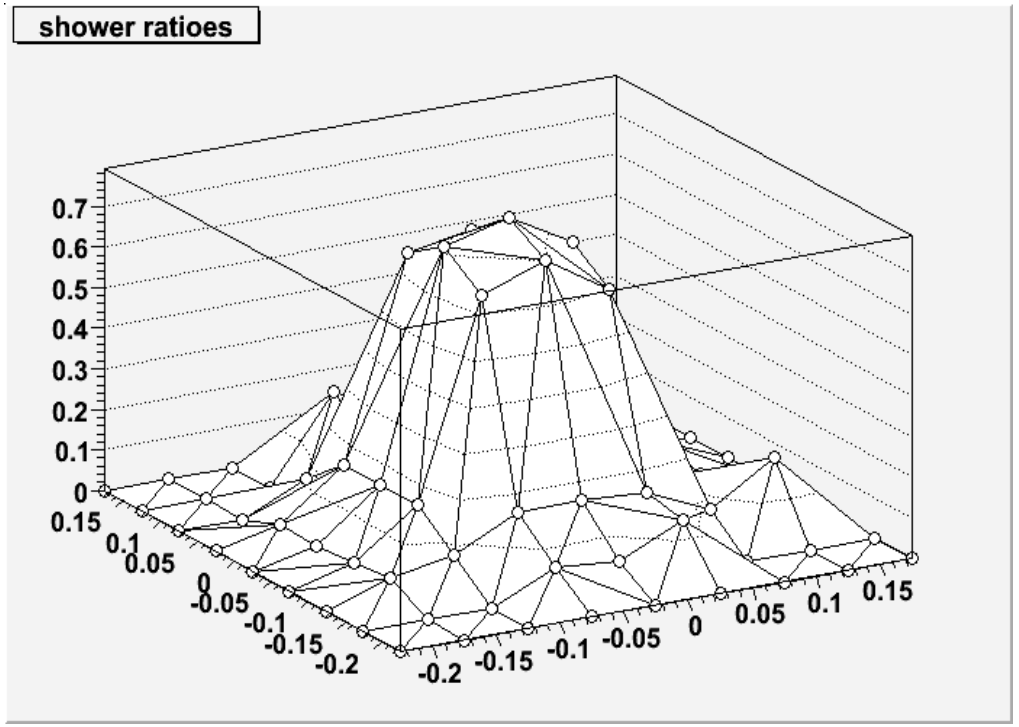




Shower



Approximately 1x1 cluster



x_cent, y_cent: center of the block i ;
x_hit = x_foc + th_foc * drift_sh;
y_hit = y_foc + ph_foc * drift_sh;
Deltax = x_hit - x_cent;
Deltay = y_hit - y_cent;

$$ratio = \frac{ADC\ of\ block\ i}{ADC\ total}$$

3D

Delta y

0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.054	0.043	0.035	0.037	0.000	0.000	0.000
0.000	0.000	0.083	0.155	0.155	0.154	0.070	0.000	0.000
0.000	0.073	0.158	0.643	0.714	0.648	0.131	0.205	0.000
0.000	0.061	0.159	0.701	0.768	0.675	0.143	0.066	0.000
0.000	0.133	0.150	0.601	0.666	0.594	0.118	0.105	0.000
0.000	0.000	0.080	0.113	0.119	0.124	0.064	0.000	0.000
0.000	0.000	0.181	0.035	0.037	0.066	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

2D

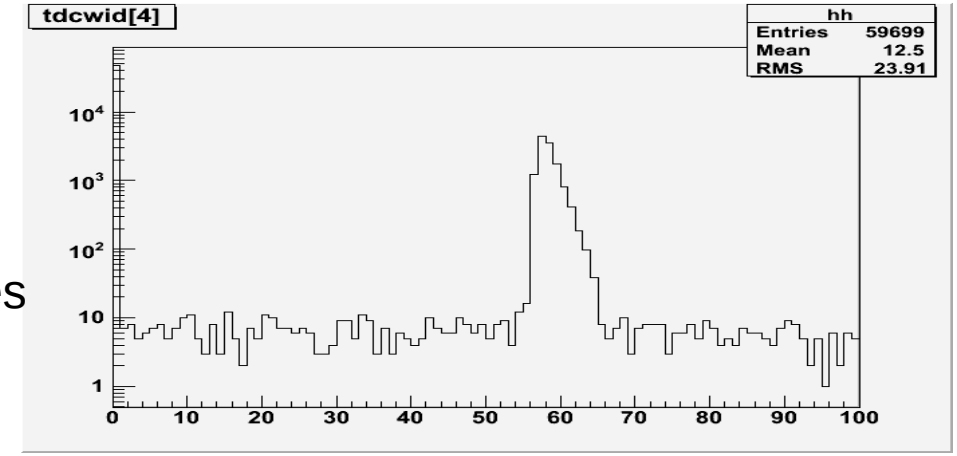
Delta X

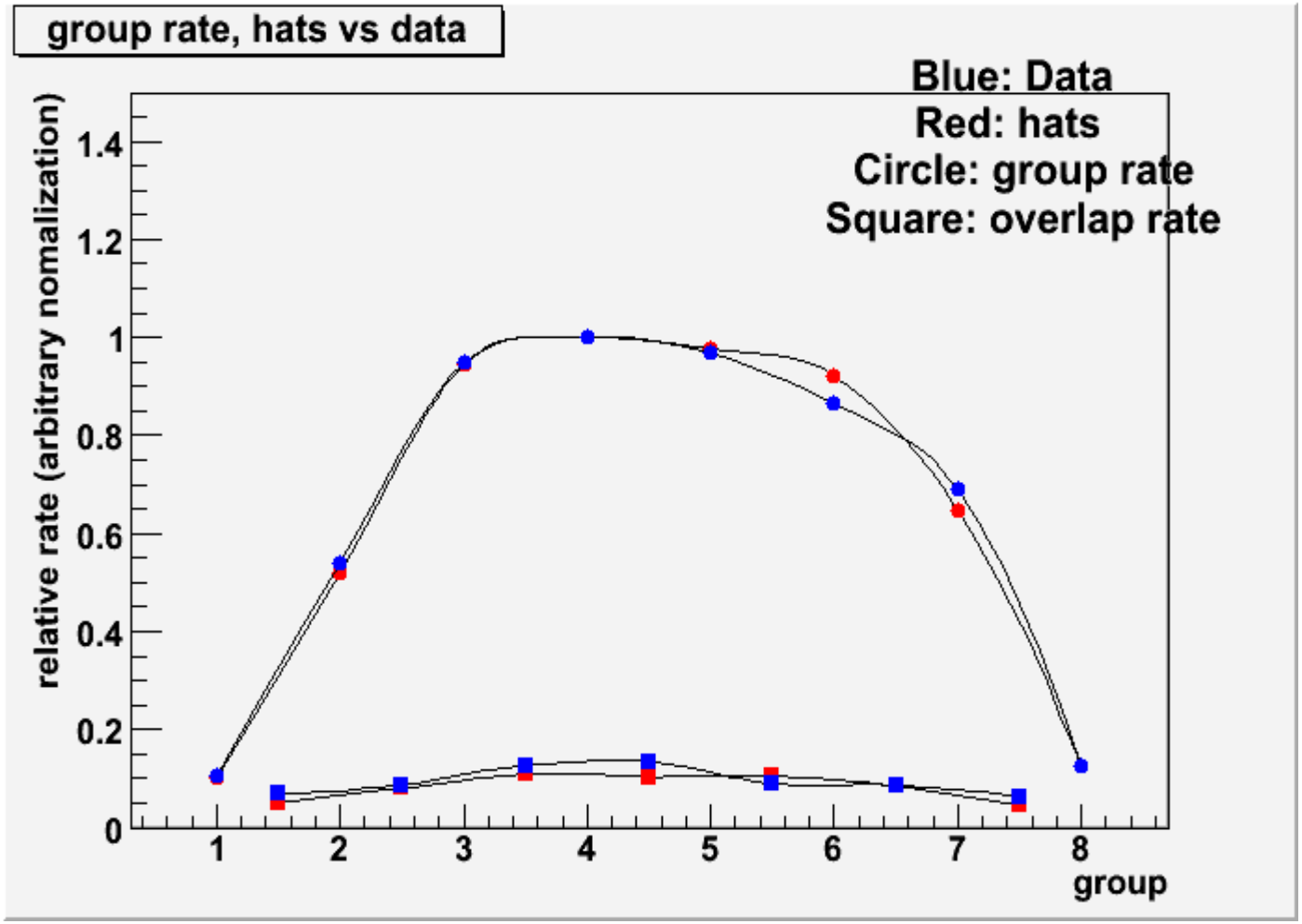
Signal Processing

Data Analysis

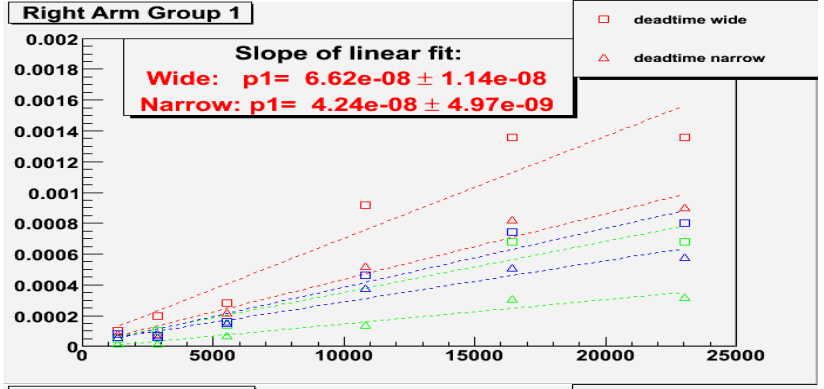
Overlap rate

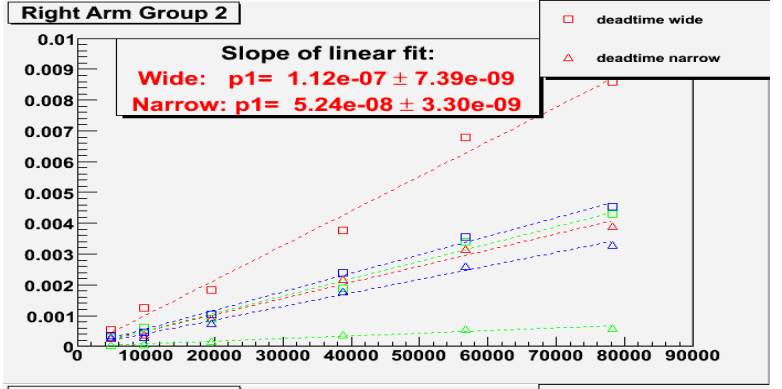
- Use tdc analysis to get tdc spectrum
- Cut on the main peak to get group rates
- Cut on group(i) && group(i+1) to get overlap rates

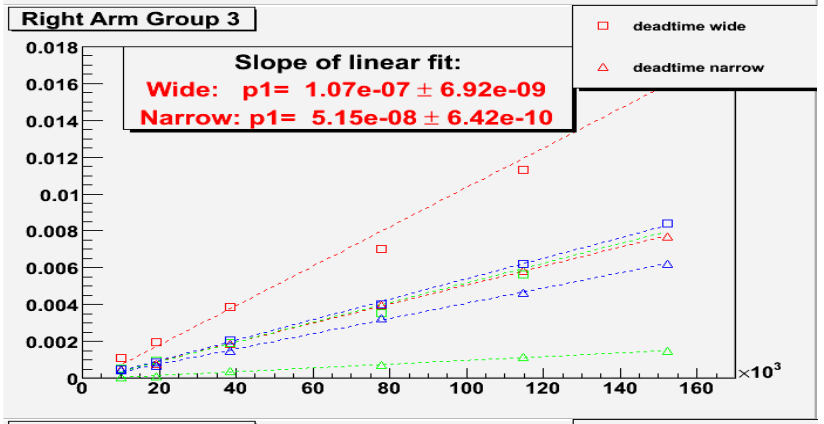


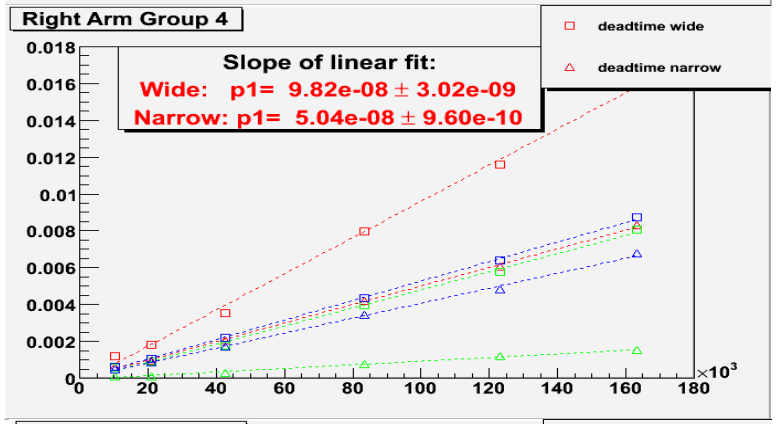


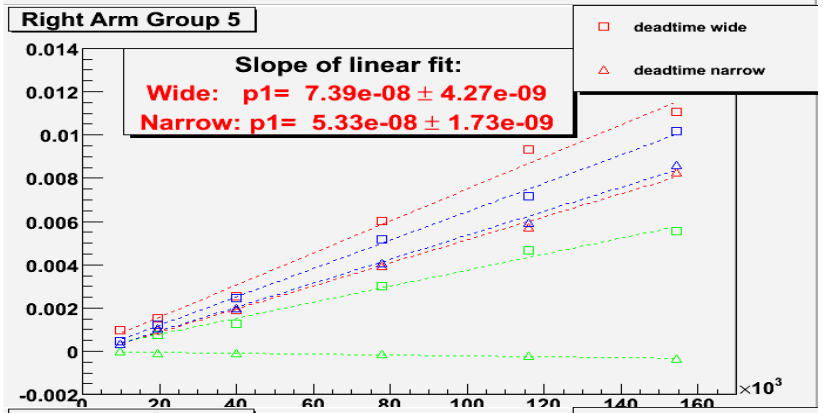
Tagger Deadtime

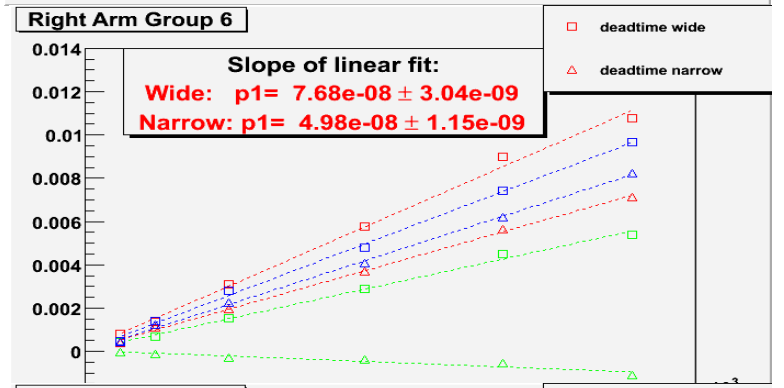


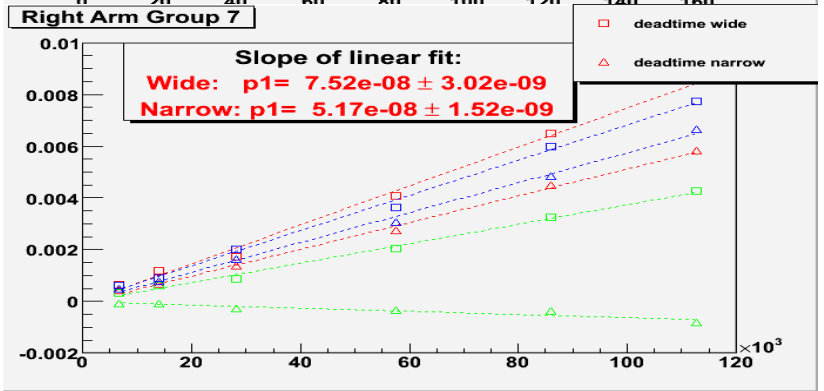


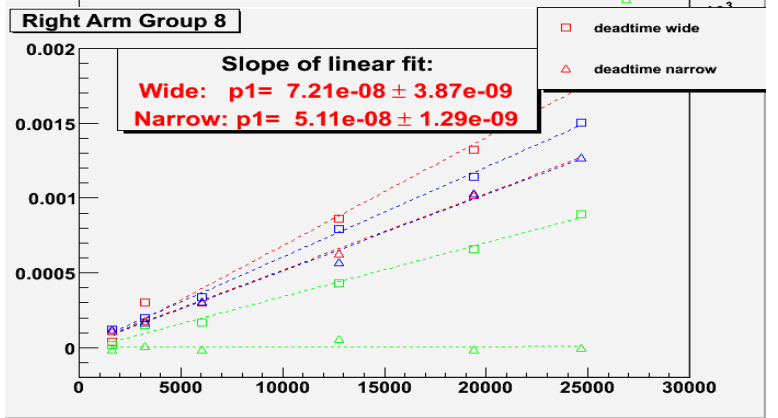




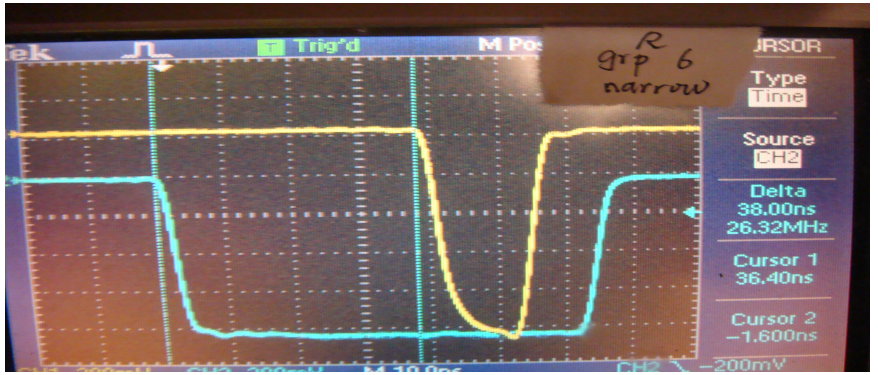


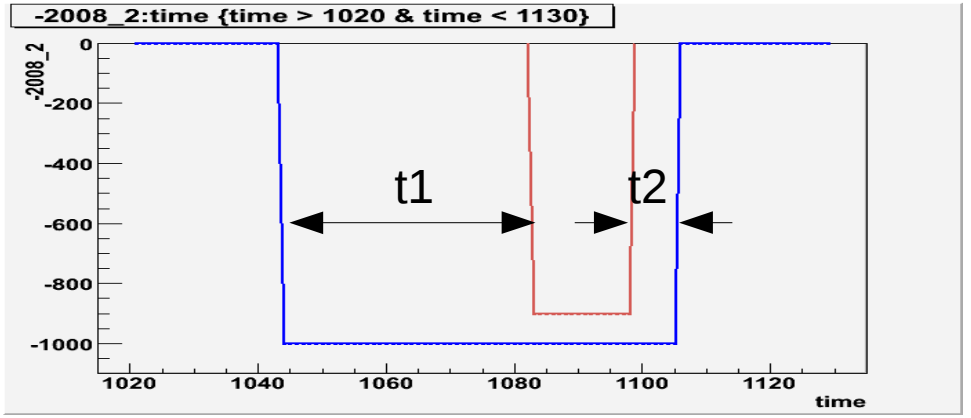


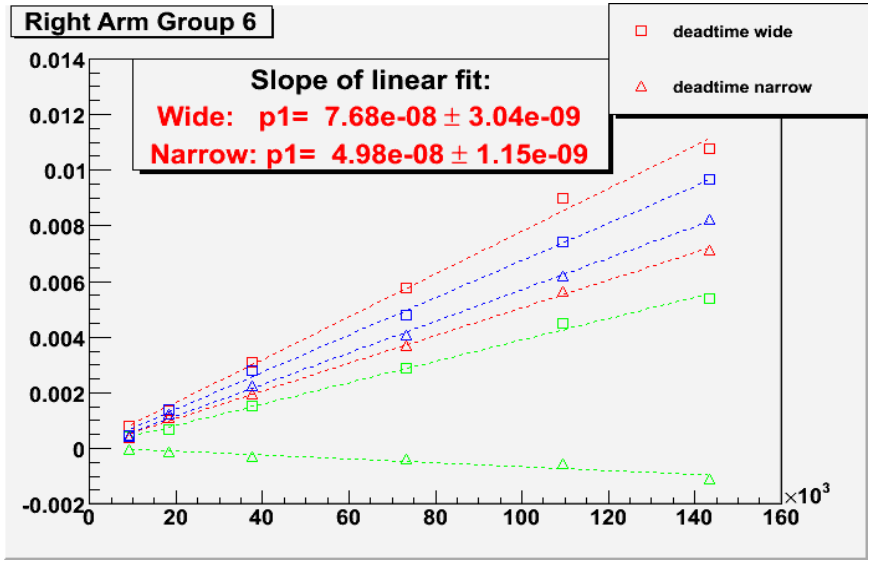


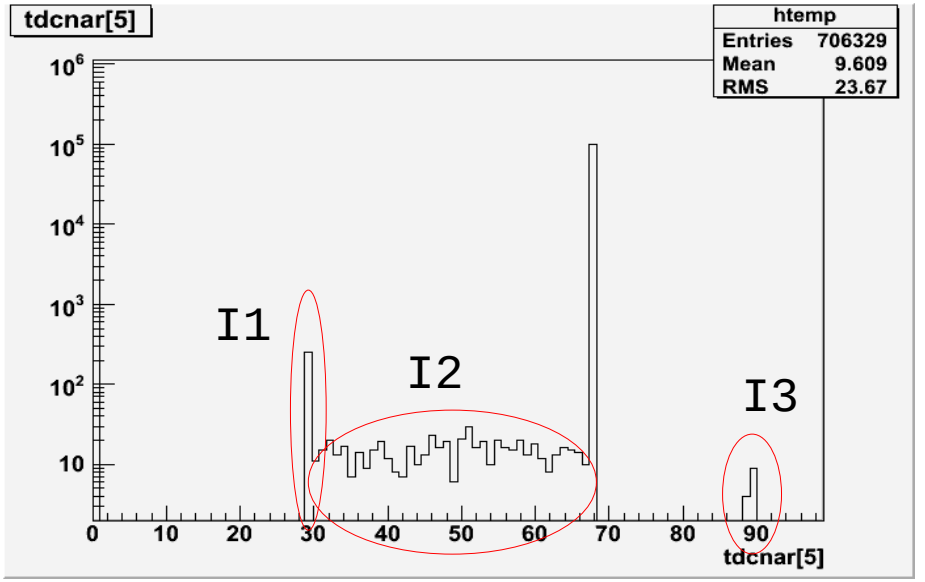


Negative Tagger Loss









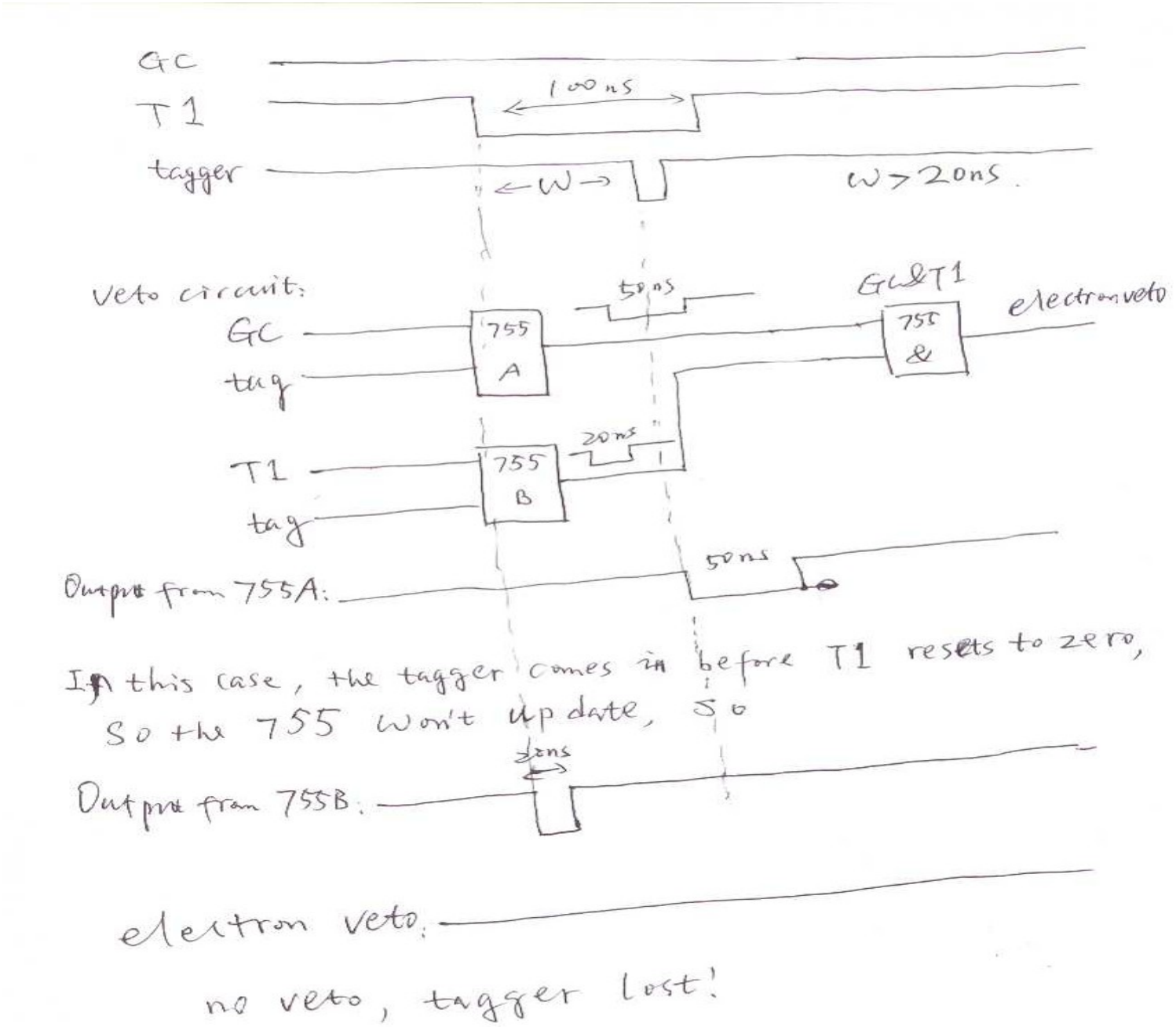
I1 = Rate X signal width (20 ns)
I2 = Rate X t1
I3 = Rate X t2

(20 + t1 + t2) > path width

↓

Negative tagger loss

T1 Deadtime: Problem revisited



To be continued.....