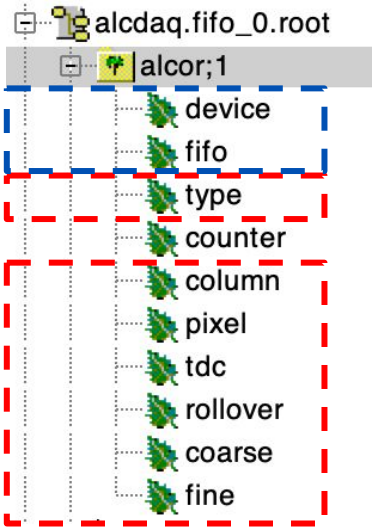
Summary of beam test data format

Nicola Rubini[(1)]

[(1)]INFN Bologna

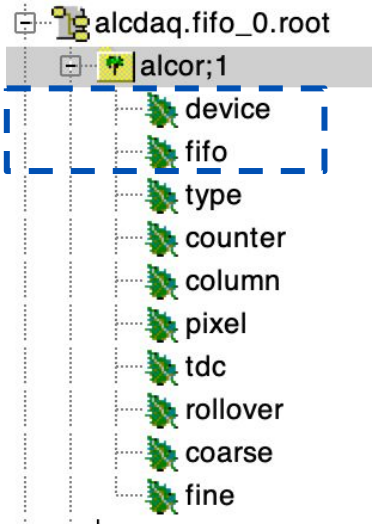
7th March 2025

ALCOR decoded raw data



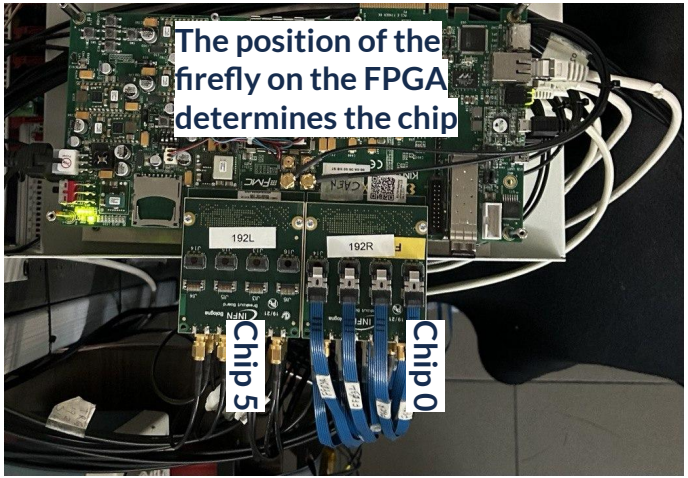
ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

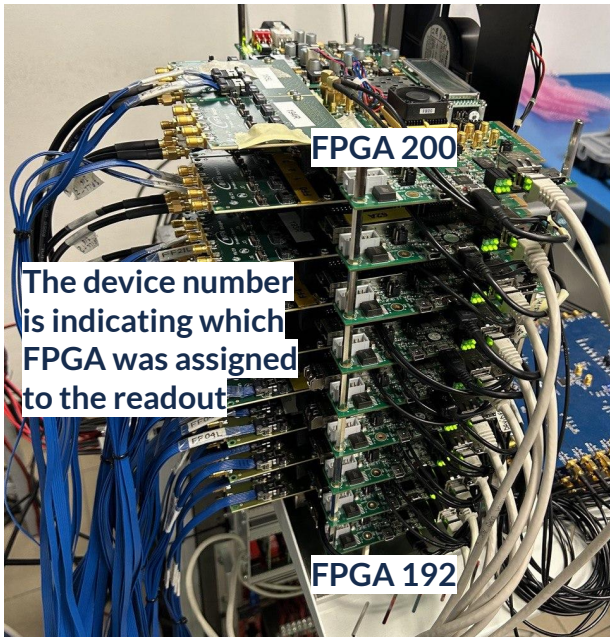
ALCOR decoded raw data



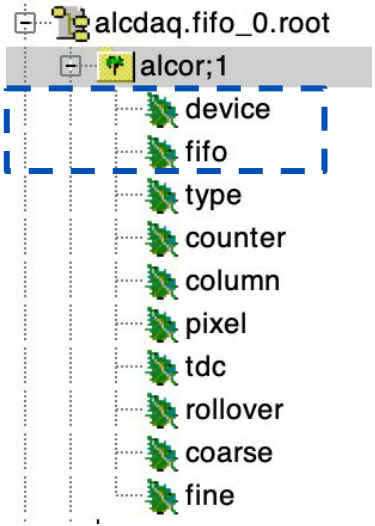
ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on Beam Test configuration, the information is stored as follows



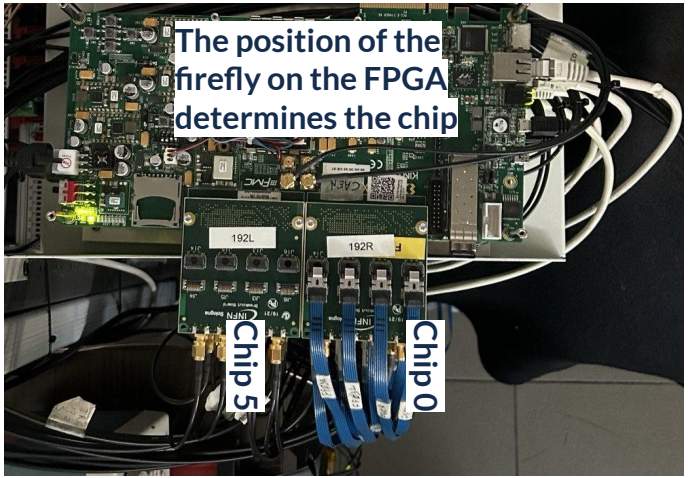


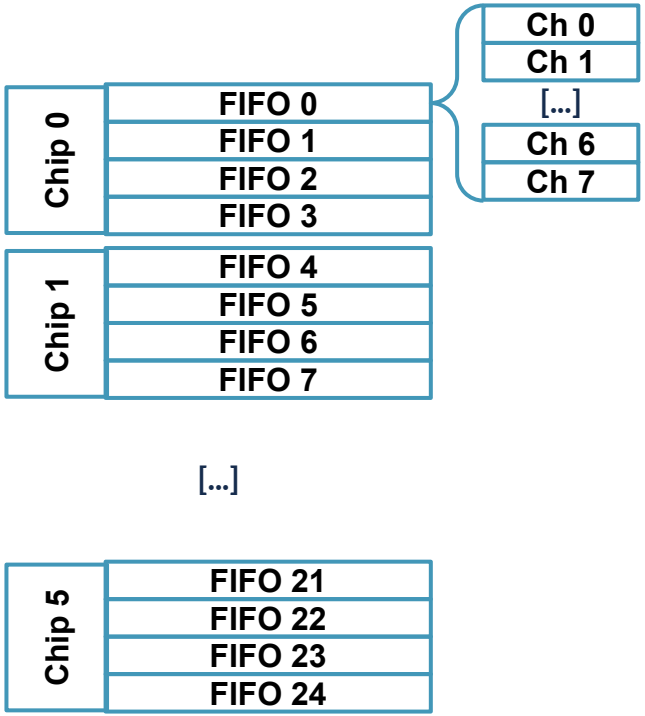
ALCOR decoded raw data



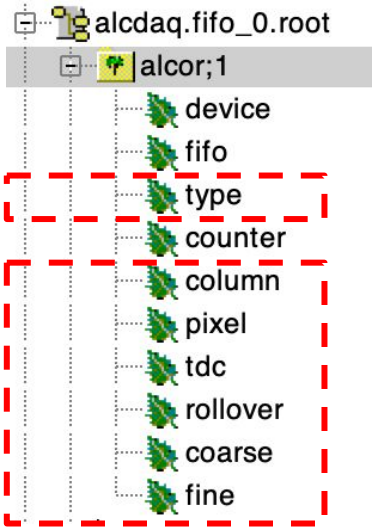
ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on Beam Test configuration, the information is stored as follows





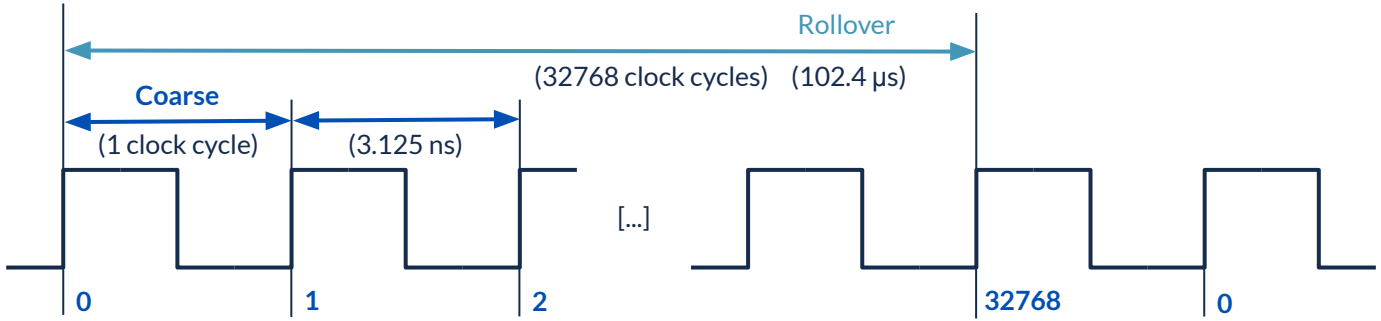
ALCOR decoded raw data



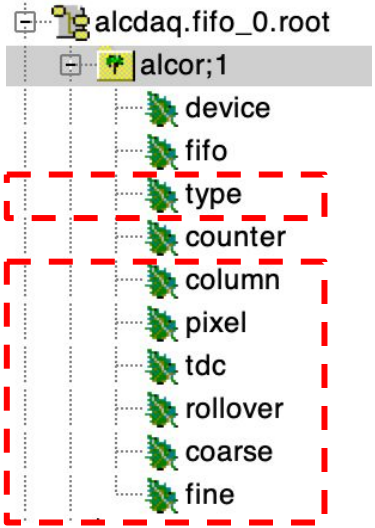
ALCOR clock
(320 MHz)

ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on ALCOR data, the chip works as follows



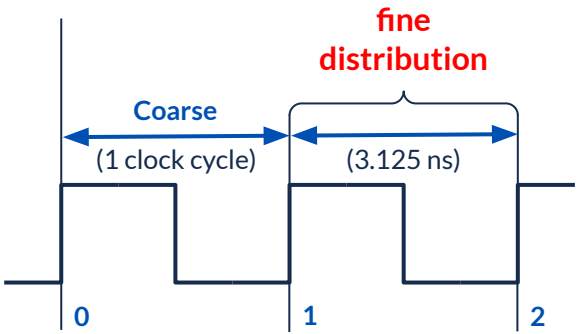
ALCOR decoded raw data



ALCOR clock
(320 MHz)

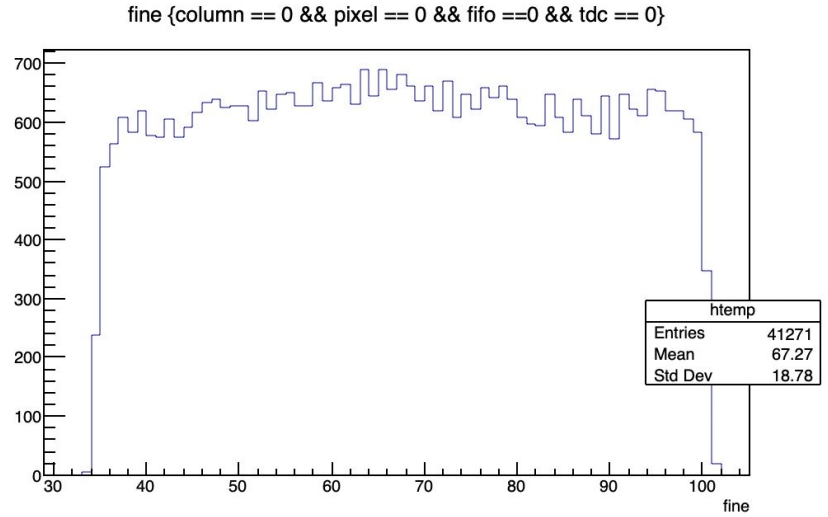
ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on ALCOR data, the chip works as follows

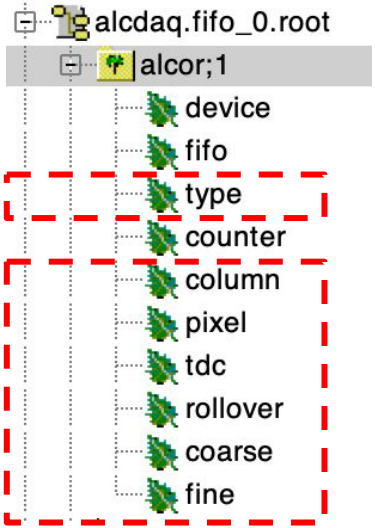


Fine distribution subdivide the coarse counter (clock cycle) into roughly ~70 parts bringing down the resolution potentially to ~45 ps

w/ fine info > fine data
w/o fine info > coarse data



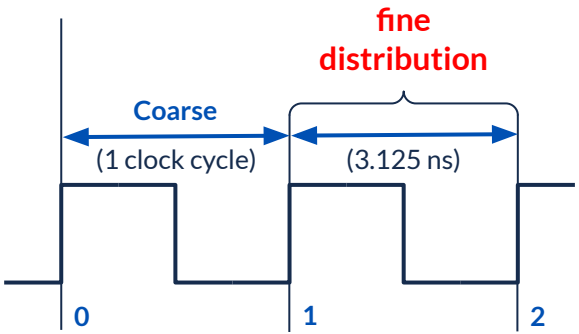
ALCOR decoded raw data



ALCOR clock
(320 MHz)

ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on ALCOR data, the chip works as follows



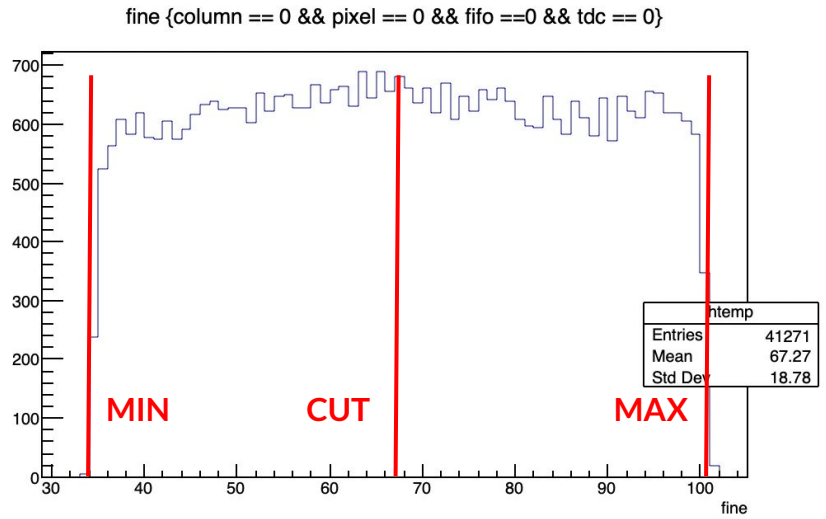
Fine correction is done by:
$t_{corr} = t_{coarse}$ - phase

ALCOR v1:

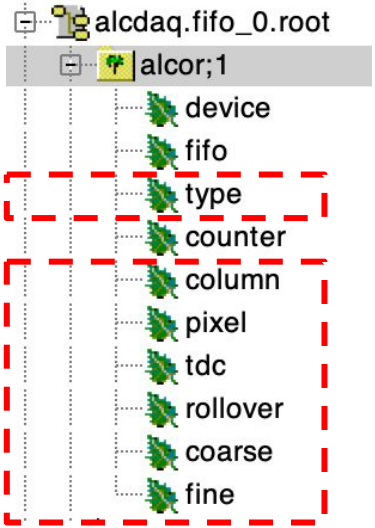
phase = (fine-MIN)/(MAX-MIN) + (fine > CUT) + off

ALCORv2:

phase = (fine-MIN)/(MAX-MIN) + off



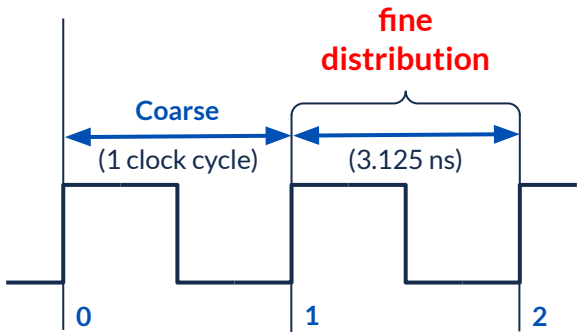
ALCOR decoded raw data

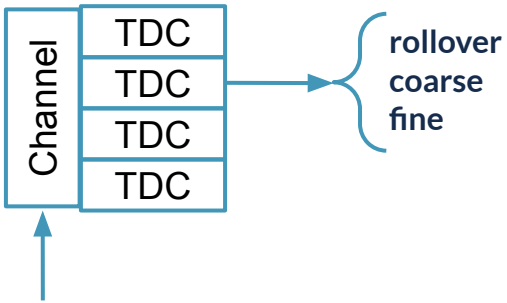


ALCOR clock
(320 MHz)

ALCOR data is decoded in a TTree structure. This is the purely ALCOR data, device and fifo are informations added later related to Beam Test configuration

Focusing on ALCOR data, the chip works as follows



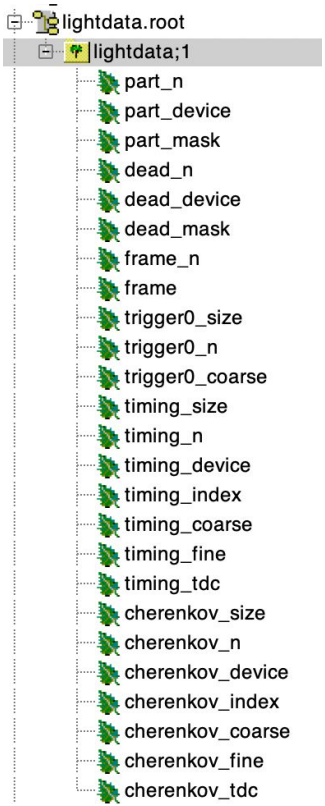


electronics oriented:
pixel + 4*column

type:
Defines whether the hit is a SiPM signal, a spill start/stop, a trigger, etc.

Lightdata

Each event in the tree is a spill



With the knowledge of the connections each device and fifo can be labeled as trigger, cherenkov, timing etc.
ALCOR gives a continuous readout which means that we need a way to optimise reading through the data

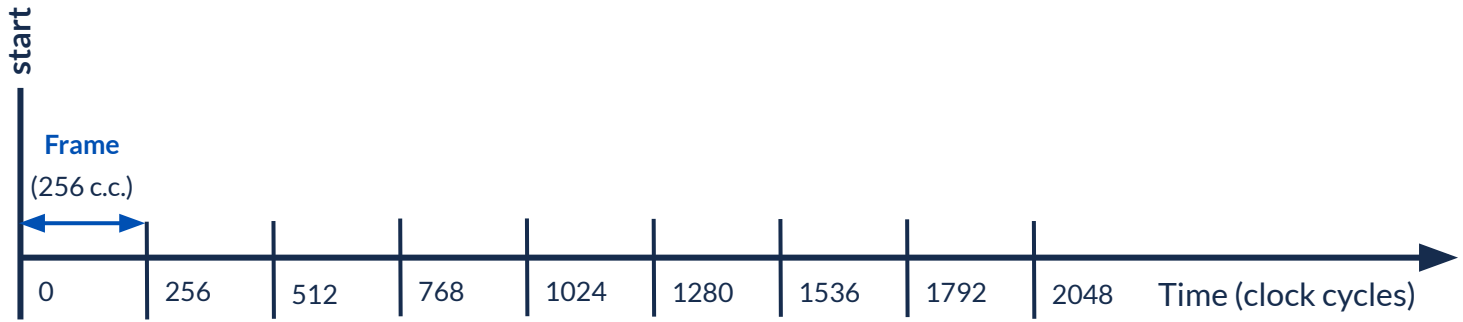
Lightdata

Each event in the tree is a spill

*this number is arbitrary and can be changed

- lightdata.root
  - lightdata;1
    - part_n
    - part_device
    - part_mask
    - dead_n
    - dead_device
    - dead_mask
    - frame_n
    - frame
    - trigger0_size
    - trigger0_n
    - trigger0_coarse
    - timing_size
    - timing_n
    - timing_device
    - timing_index
    - timing_coarse
    - timing_fine
    - timing_tdc
    - cherenkov_size
    - cherenkov_n
    - cherenkov_device
    - cherenkov_index
    - cherenkov_coarse
    - cherenkov_fine
    - cherenkov_tdc

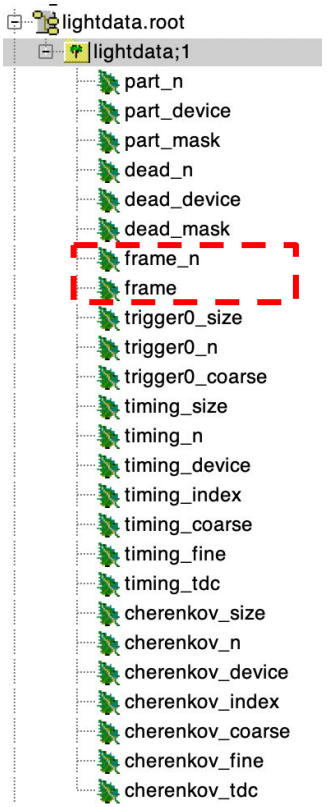
With the knowledge of the connections each device and fifo can be labeled as trigger, cherenkov, timing etc.
ALCOR gives a continuous readout which means that we need a way to optimise reading through the data
This is achieved by subdividing the time in frames of 256* clock cycles (800 ns)



Lightdata

Each event in the tree is a spill

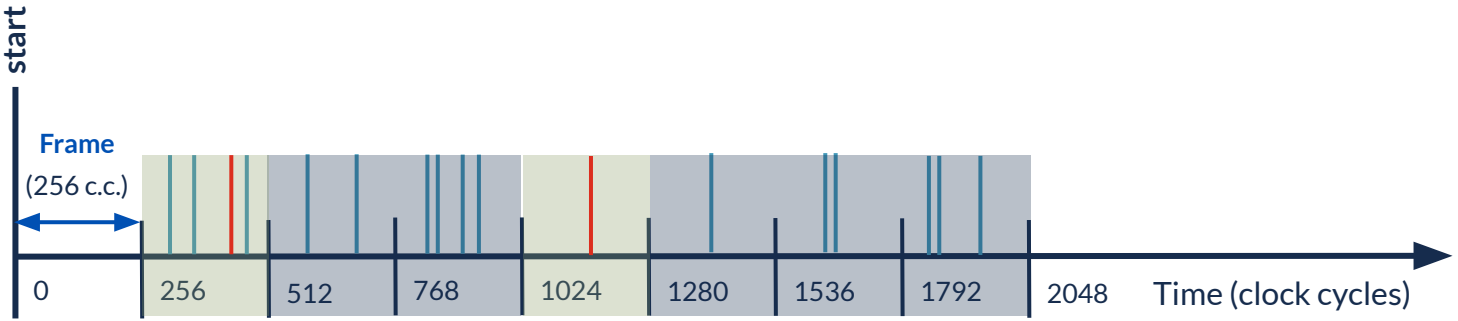
*this number is arbitrary and can be changed



With the knowledge of the connections each device and fifo can be labeled as trigger, cherenkov, timing etc.
ALCOR gives a continuous readout which means that we need a way to optimise reading through the data
This is achieved by subdividing the time in frames of 256* clock cycles (800 ns)

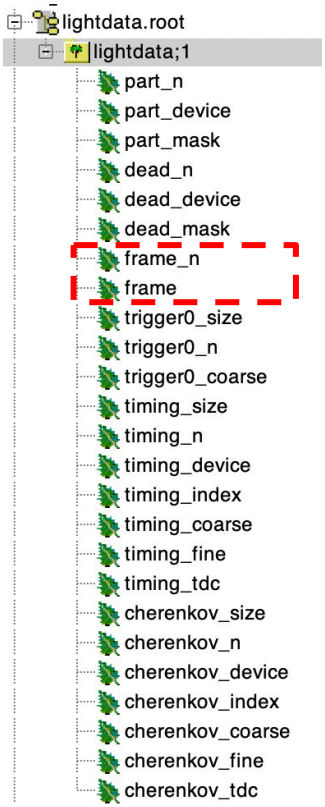
In each frame you can have cherenkov hits and trigger signal

Then, only frames with a trigger are accepted into the lightdata.
This format essentially filters for frames with a trigger.



Lightdata

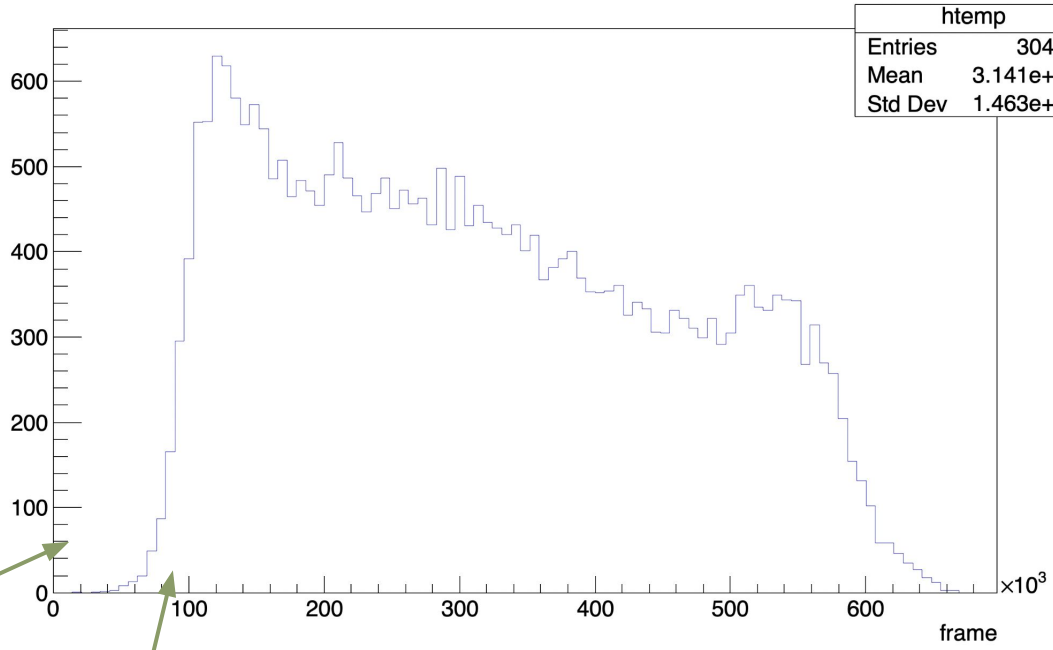
Each event in the tree is a spill

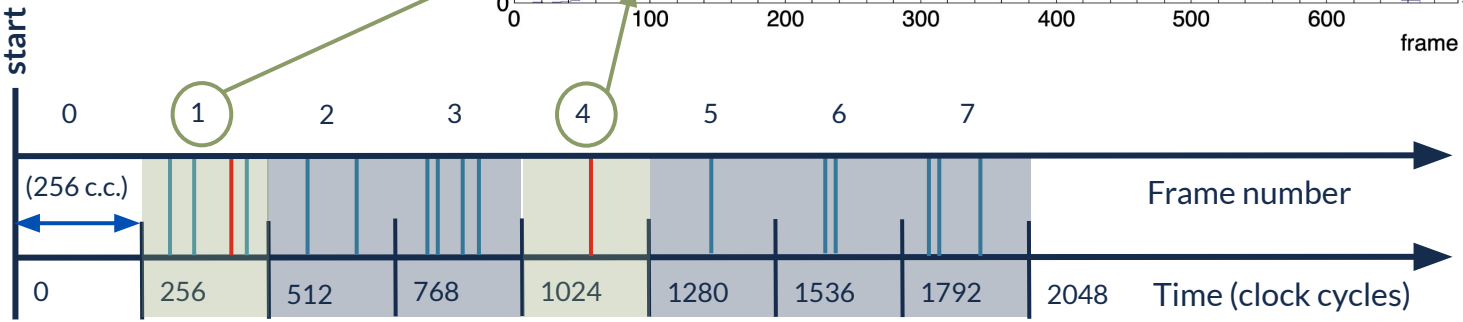


The event in the TTree is a spill.
In frame_n the number of accepted frames is stored.
In frame the array of the frames saved. In this case:

frame_n: 2

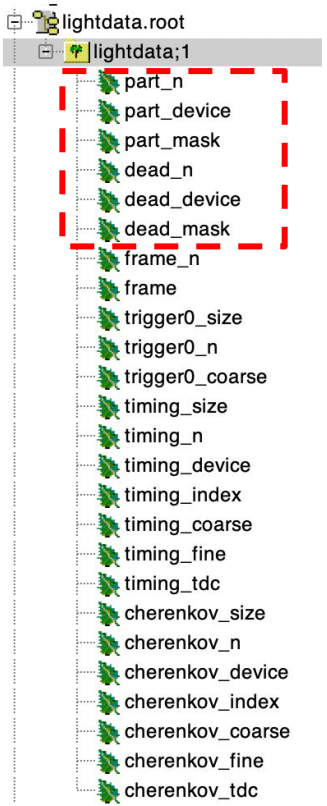
frame: {1,4}





Lightdata

Each event in the tree is a spill



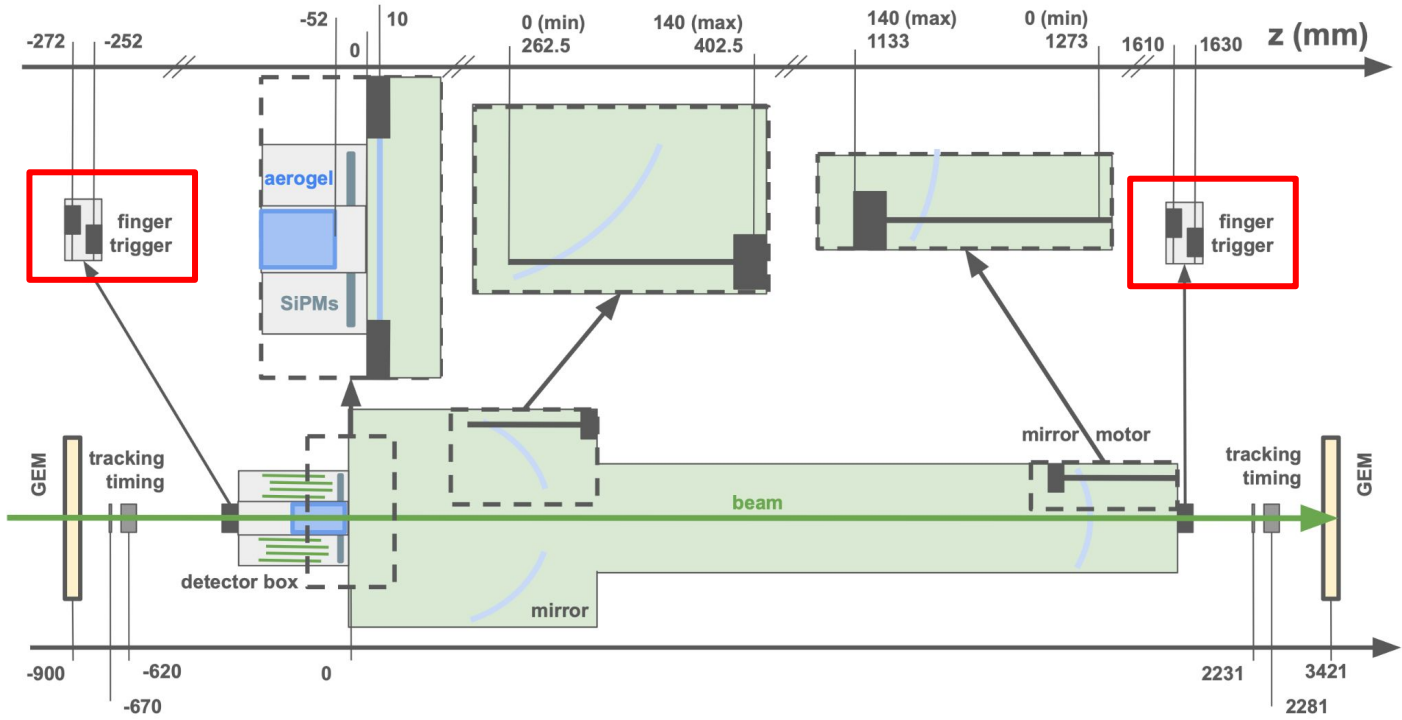
part indicates which device and FIFO is participating in the run. The device refers to the reading FPGA IP address [192-208] and the mask is a bit map for the participating FIFOs:
(1001010111) > (1,4,6,8,9,10)

dead follows the same pattern but refers to FIFOs which reported issues during the run

Lightdata

Each event in the tree is a spill

- lightdata.root
  - lightdata;1
    - part_n
    - part_device
    - part_mask
    - dead_n
    - dead_device
    - dead_mask
    - frame_n
    - frame
    - trigger0_size
    - trigger0_n
    - trigger0_coarse
    - timing_size
    - timing_n
    - timing_device
    - timing_index
    - timing_coarse
    - timing_fine
    - timing_tdc
    - cherenkov_size
    - cherenkov_n
    - cherenkov_device
    - cherenkov_index
    - cherenkov_coarse
    - cherenkov_fine
    - cherenkov_tdc



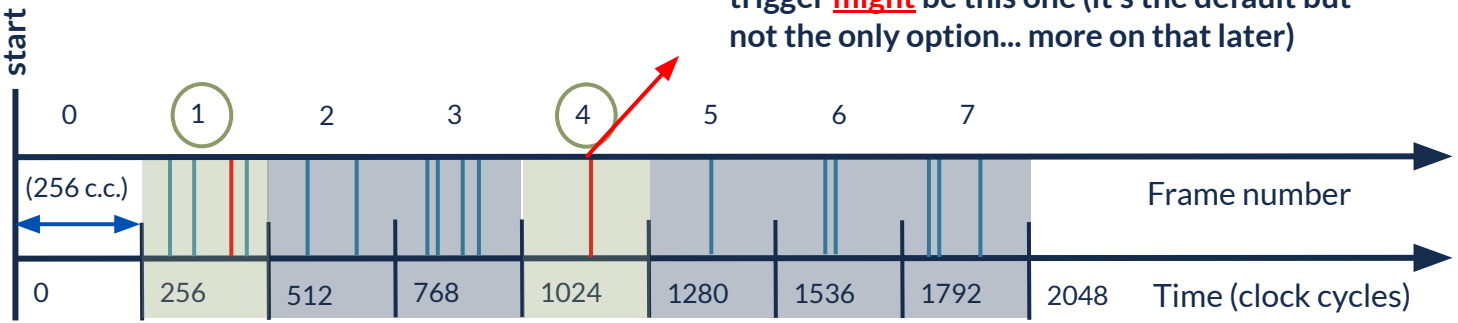
Lightdata

Each event in the tree is a spill

- lightdata.root
  - lightdata;1
    - part_n
    - part_device
    - part_mask
    - dead_n
    - dead_device
    - dead_mask
    - frame_n
    - frame
    - trigger0_size
    - trigger0_n
    - trigger0_coarse
    - timing_size
    - timing_n
    - timing_device
    - timing_index
    - timing_coarse
    - timing_fine
    - timing_tdc
    - cherenkov_size
    - cherenkov_n
    - cherenkov_device
    - cherenkov_index
    - cherenkov_coarse
    - cherenkov_fine
    - cherenkov_tdc

trigger indicates the recorded number and time the finger triggers fired*

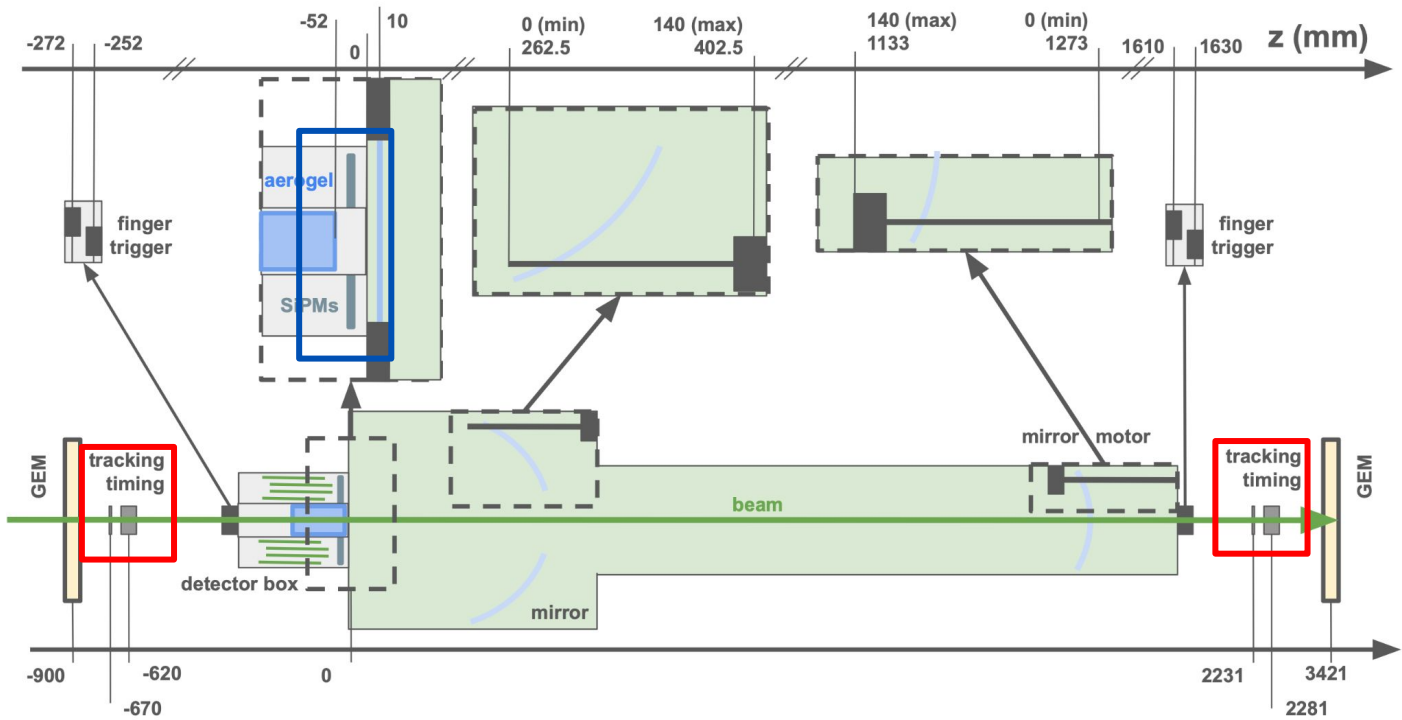
(*) if you remember the frame system, the trigger might be this one (it's the default but not the only option... more on that later)



Lightdata

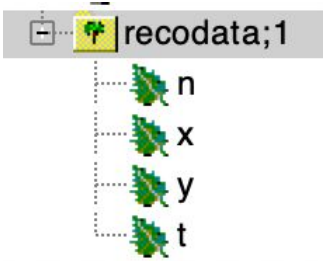
Each event in the tree is a spill

- lightdata.root
  - lightdata;1
    - part_n
    - part_device
    - part_mask
    - dead_n
    - dead_device
    - dead_mask
    - frame_n
    - frame
    - trigger0_size
    - trigger0_n
    - trigger0_coarse
    - timing_size
    - timing_n
    - timing_device
    - timing_index
    - timing_coarse
    - timing_fine
    - timing_tdc
    - cherenkov_size
    - cherenkov_n
    - cherenkov_device
    - cherenkov_index
    - cherenkov_coarse
    - cherenkov_fine
    - cherenkov_tdc



Recodata

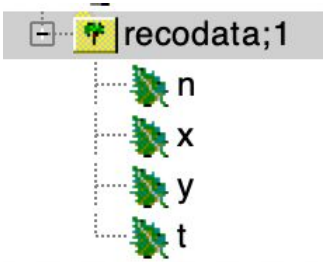
Each event in the tree is a frame



Recodata is the most filtered and analysis ready data format:
each TTree event is a frame and contains the hit *x*, *y* position, the time delta (*t*) with the reference time* and how many hits (*n*) are in the frame

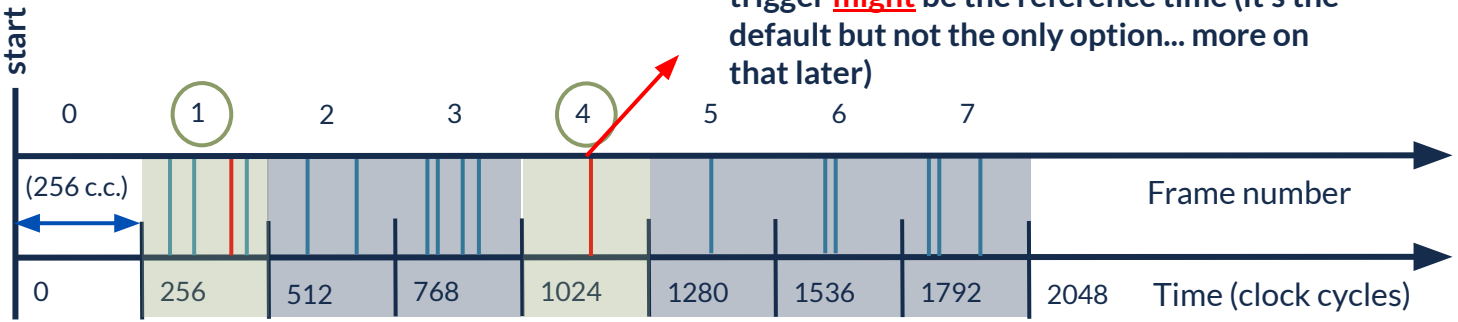
Recodata

Each event in the tree is a frame



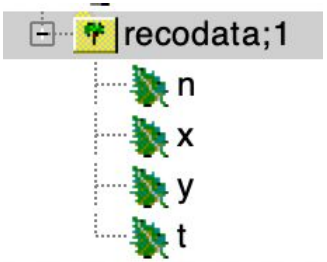
Recodata is the most filtered and analysis ready data format:
each TTree event is a frame and contains the hit *x*, *y* position, the time delta (*t*) with the reference time* and how many hits (*n*) are in the frame

(*) if you remember the frame system, the trigger might be the reference time (it's the default but not the only option... more on that later)

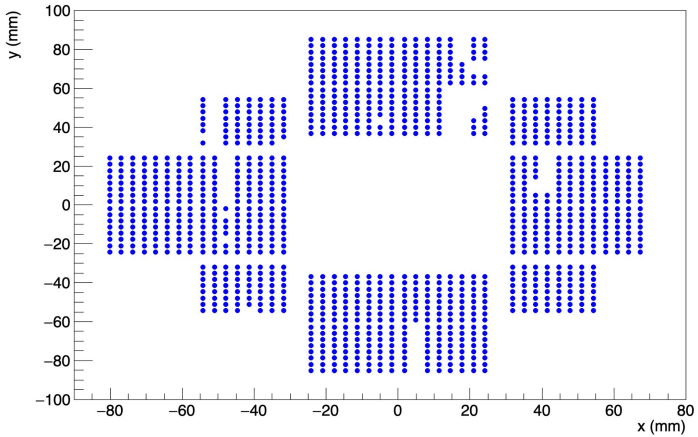


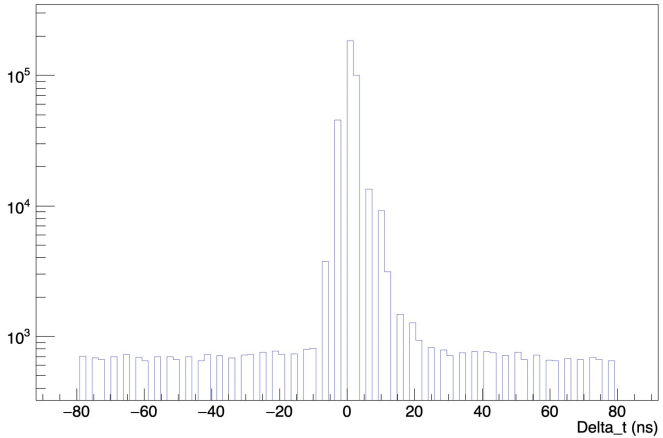
Recodata

Each event in the tree is a frame



Recodata is the most filtered and analysis ready data format:
each TTree event is a frame and contains the hit *x*, *y* position, the time delta (*t*) with the reference time* and how many hits (*n*) are in the frame





Known readout issues

Fine Tune

[2023BT][Timing][Fixed->ALCORv2]

- Fine tune has an ambiguity at the cut threshold where a wrong clock cycle is assigned (effectively shifting the phase of one clock cycle)

Clock cycle shift

[All][Filtered->Upon req.]

- Data sometimes start shifted by a clock cycle, making the coincidence of cherenkov light (or anything else) more difficult or cut-out if not taken into consideration

Rollover shift

[All][Filtered->Upon req.]

- Data sometimes start shifted by a rollover word, making the coincidence of cherenkov light (or anything else) more difficult or cut-out if not taken into consideration

Other known issues

Background during spill

[All][Still under inv.]

- Fluctuating levels of background have been found in both beam tests, a more thorough investigation is in progress

Analysis targets

SPR and Photon yield

[NR][Ongoing]

- Develop a standard approach to systematically and precisely measure SPR and photon yield of cherenkov light

Afterpulse

[TBA]

- Develop a standard analysis to measure and/or treat afterpulse impact in data. It's possible to match results with characterisation results.

Cross-talk

[TBA]

- Develop a standard analysis to measure and/or treat cross-talk impact in data. It's possible to match results with characterisation results.

Mirror scan

[TBA]

- Develop an analysis to measure and evaluate impact of focusing mirrors on SPR and photon yield

Other investigations

Background

[NR][Ongoing]

- Characterise the bkg and evaluate its impact on the SPR and find source

GEM-available runs

[NR][Stopped][Available]

- Merge GEM and ALCOR data with QA to enhance precision

SiPM Timing

[NR][Stopped][Available]

- Work on ALCOR based timing to minimise time reference in recodata

SiPM Tracking

[NR][Stopped][Available]

- Work on ALCOR based tracking

Thank you!

Back-up