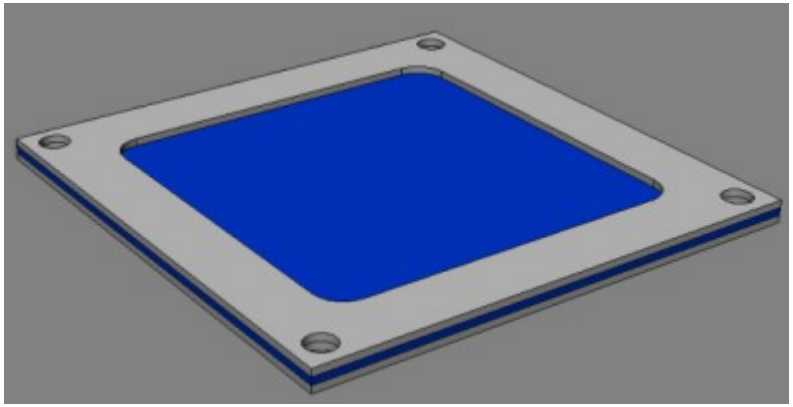
Method for pile up events removal

Giacomo Ubaldi - 04/05/2022

Start Counter

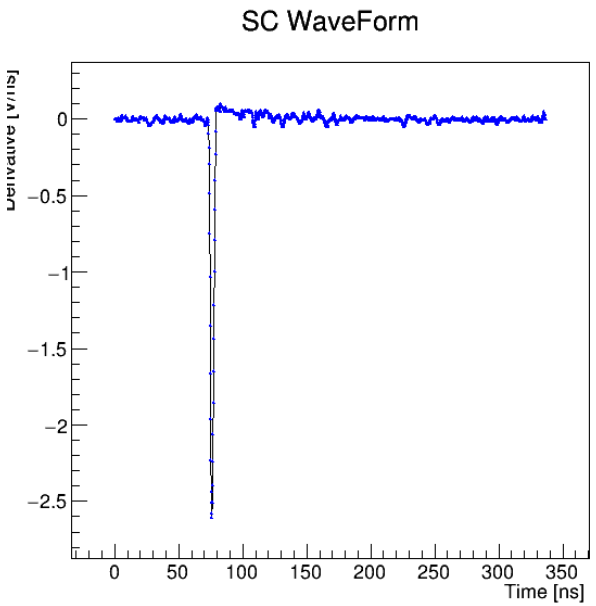
- The Start Counter gives a **signal** which is the result of the **sum of 8 channels** connected to the WaveDAQ system
- From it, we obtain the first *time stamp* and *charge* information of an event
- SC needs to operate at very low beam rates, i.e. ~ **10 kHz,** so the flux of the beam has to be low.
  However it is not constant: there is the possibility of **pile up events**.



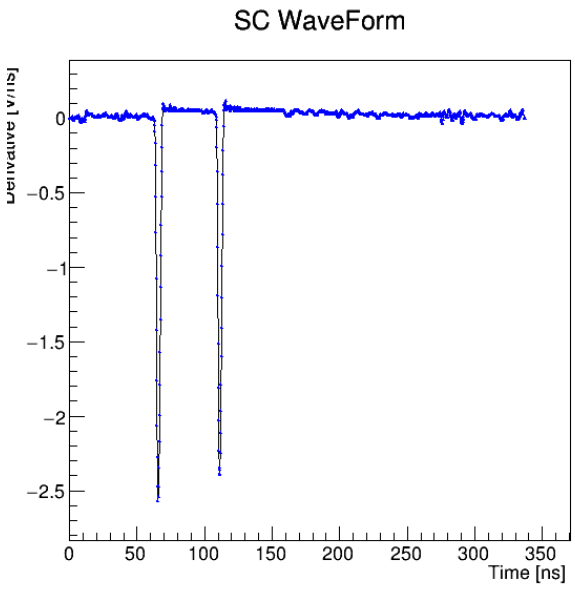
Start Counter Pile Up

Pile up is the superimposition of more than a projectile in the SC time window, which is **~ 350 ns.**

It means that the signal from the SC could have more than one peak in pile up cases:



Single projectile



Pileup projectile

It is fundamental to filter out pile up events because they worsen the overall analysis.

One method to discriminate pile up events is the **derivative method.**

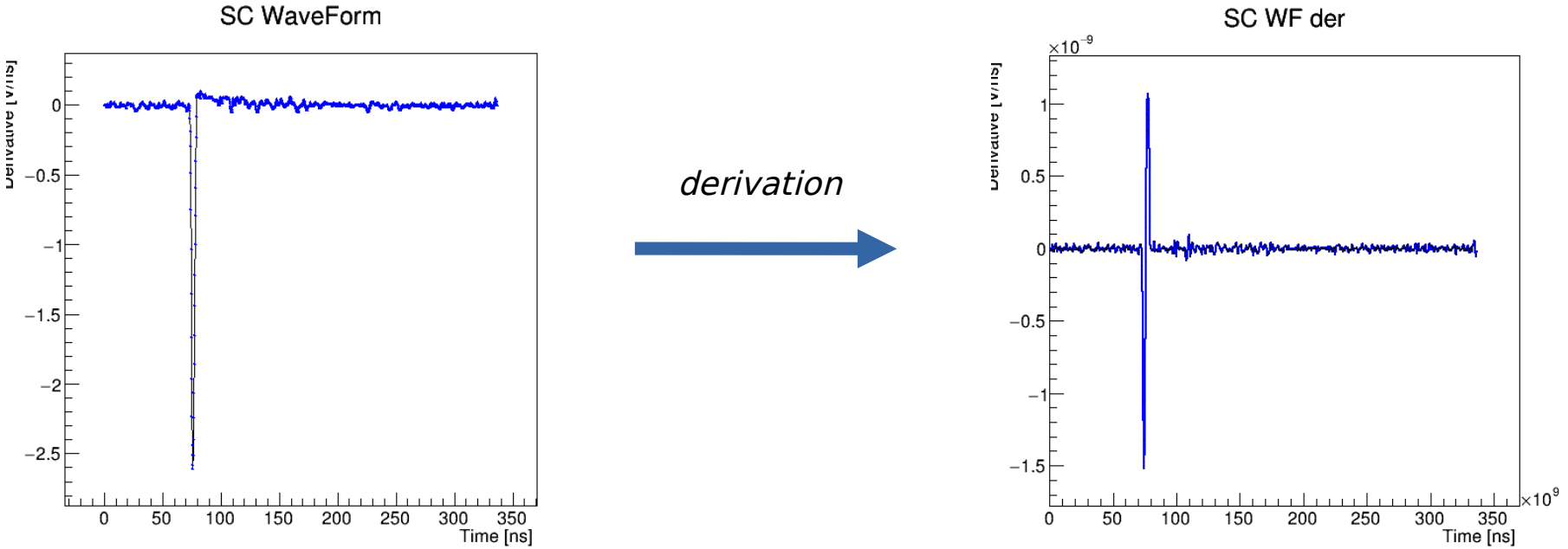
I implemented on SHOE the version developed by Roberto Zarrella

Derivative method, 1

At first, the derivative is made by an algorithm which iterates on every point of the SC signal. The derivative range is of 5 points.

In particular,

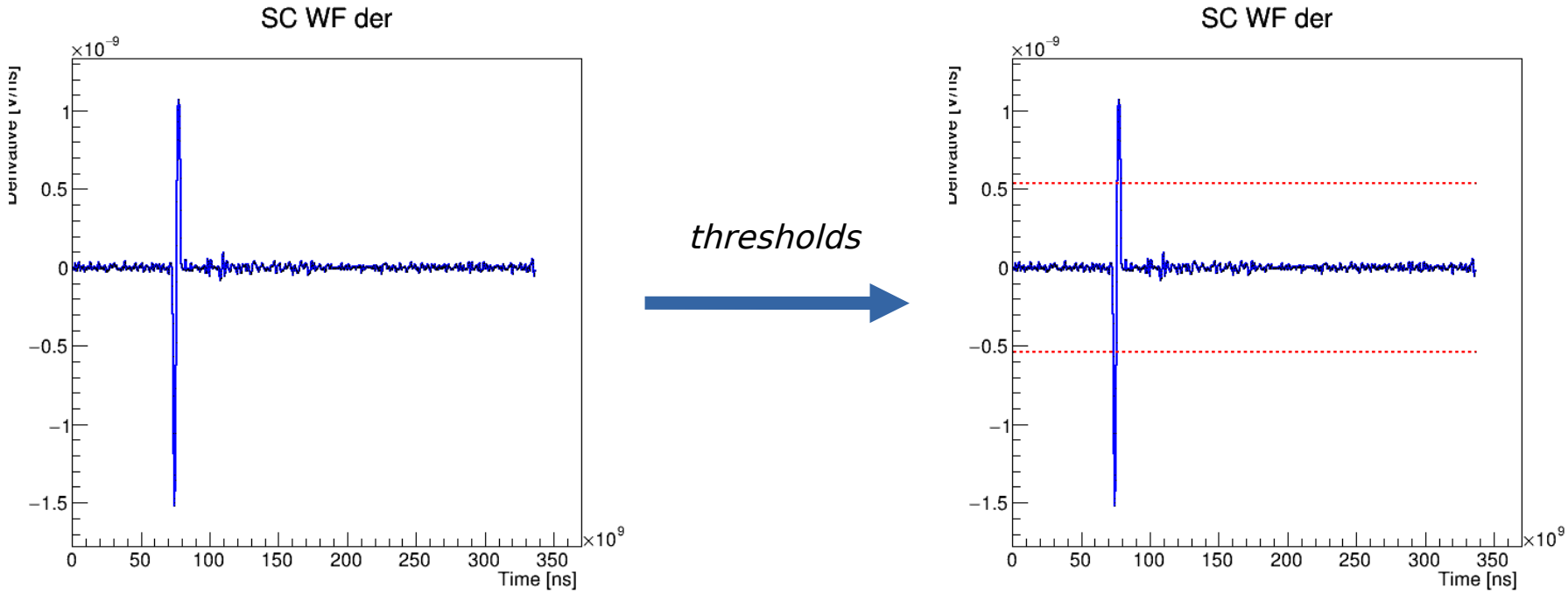
∀ point i: $D(i)=\frac{\Delta y}{\Delta x}=\frac{y[i+5]-y[i]}{x[i+5]-x[i]}$



- From every peak of the signal, we obtain **two** peaks in the derivative, whose height depends on the steepness of the primitive signal
- Baseline noise and overshoot are negligible with this method using 5 consecutive points in the elaboration of the derivative.

Derivative method, 2

Then a **threshold** is defined as the 50% of the positive peak height.

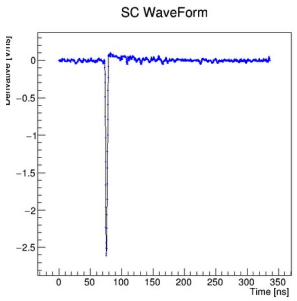


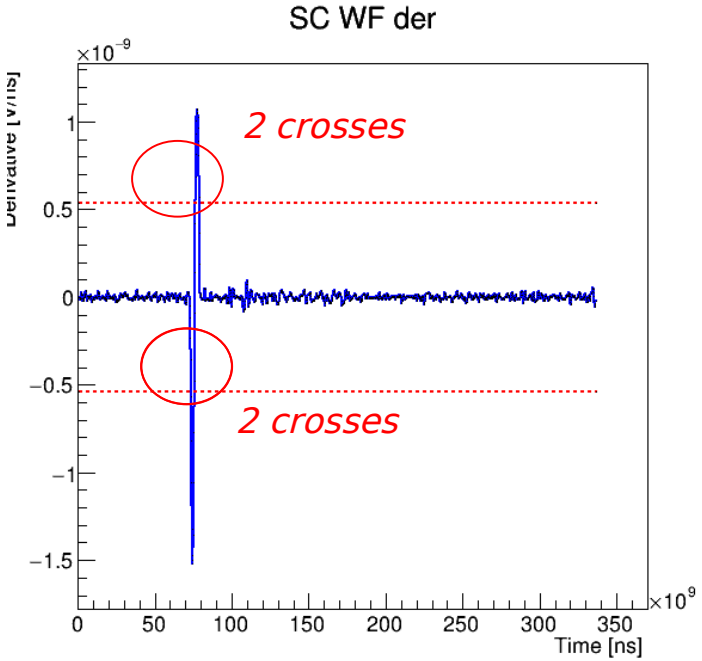
- The symmetric value is used for the negative threshold.
- 50% is choosen to have a good distance between baseline and so for a good discrimination

Derivative method, 3

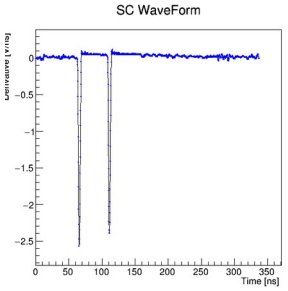
A single signal intersects the thresold lines **4 times**: two for the positive and two for the negative ones.
For a **pileup** signal the intersections are **more than 4**.

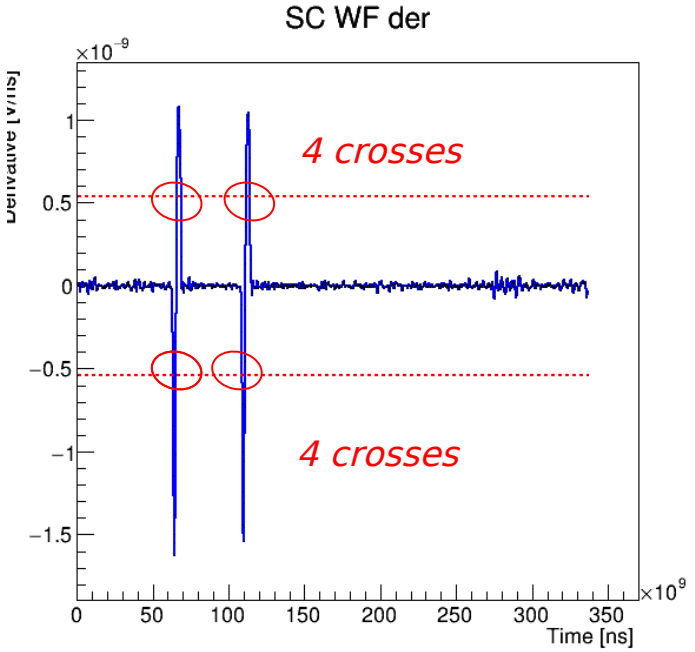
- This is at the basis of the **constant threshold discrimination method**.





Single projectile: 4 crosses

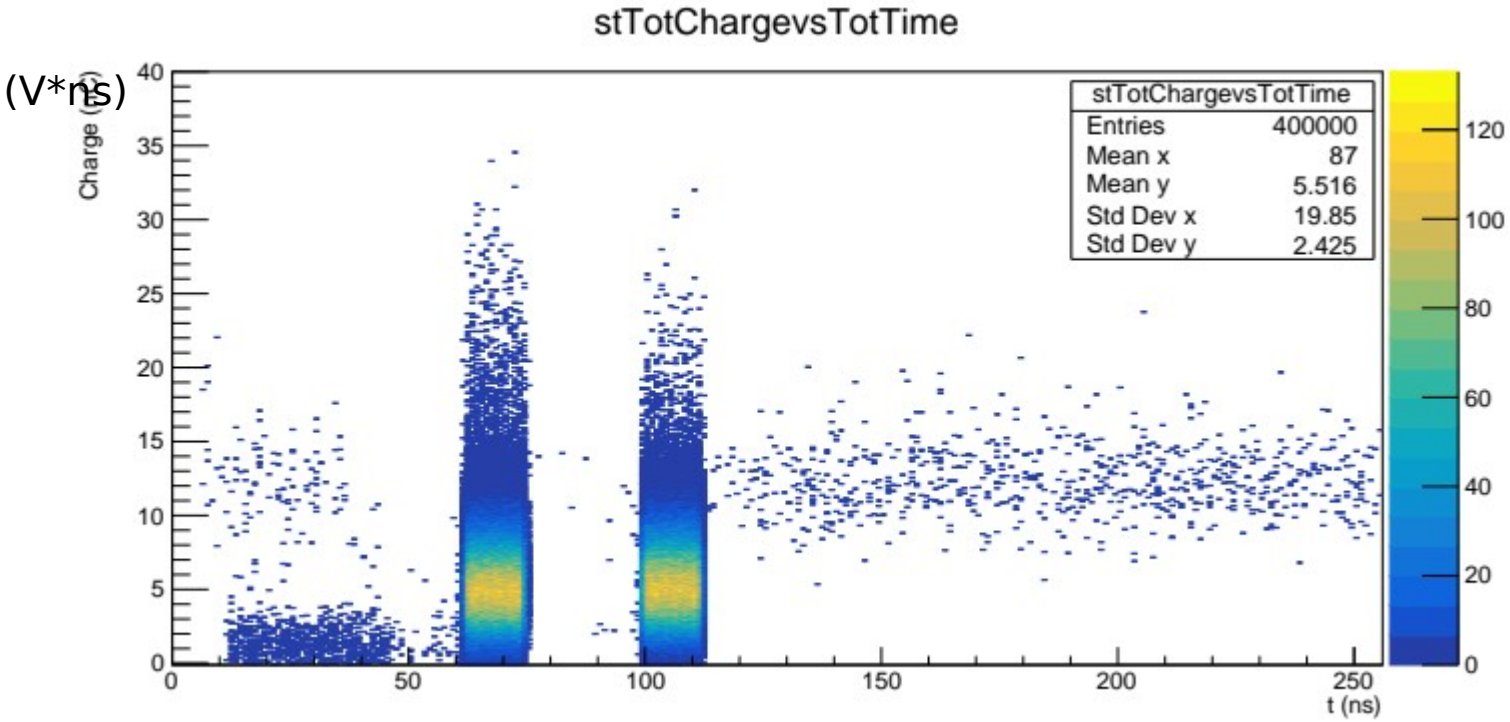




pileup projectile: >4 crosses

Results, 1

- The method was tested on real **data taken at GSI** of beam of **O16** of 400 MeV/n against a target of C
(DataGSI2021/data_test.00004310.physics_foot.daq.RAW._lb0000._FOOT-RCD._0001.data)

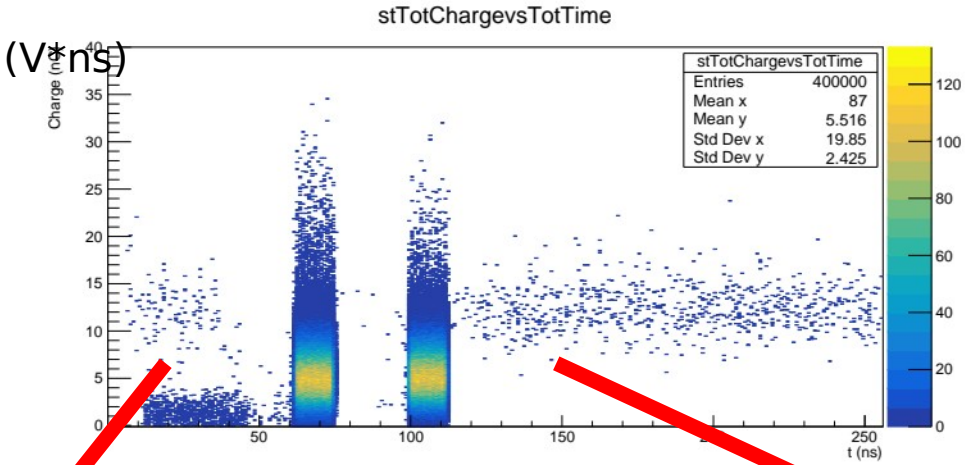


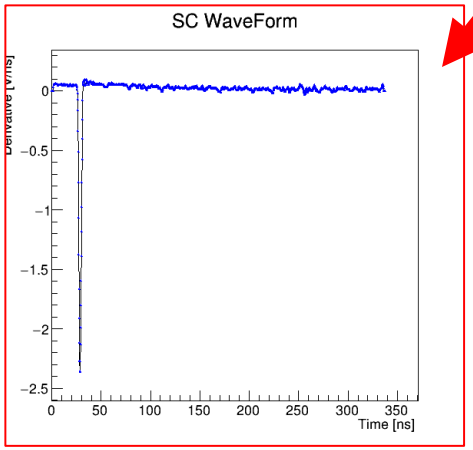
In the graph time vs charge of all the events, the 2 “columns” are due to ***minimum bias trigger*** and ***trigger fragmentation,*** which have 2 different response times.

- Column width are due to the SC resolution of 12 ns.

Results, 2

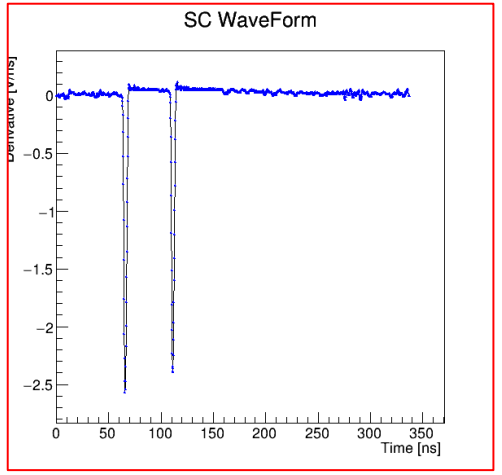
Before and after the “columns”, it is possible to find two different types of events:





“Hidden” Pileup events:
It is visible only a peak because the previous one was recorded before the start of SC time window.

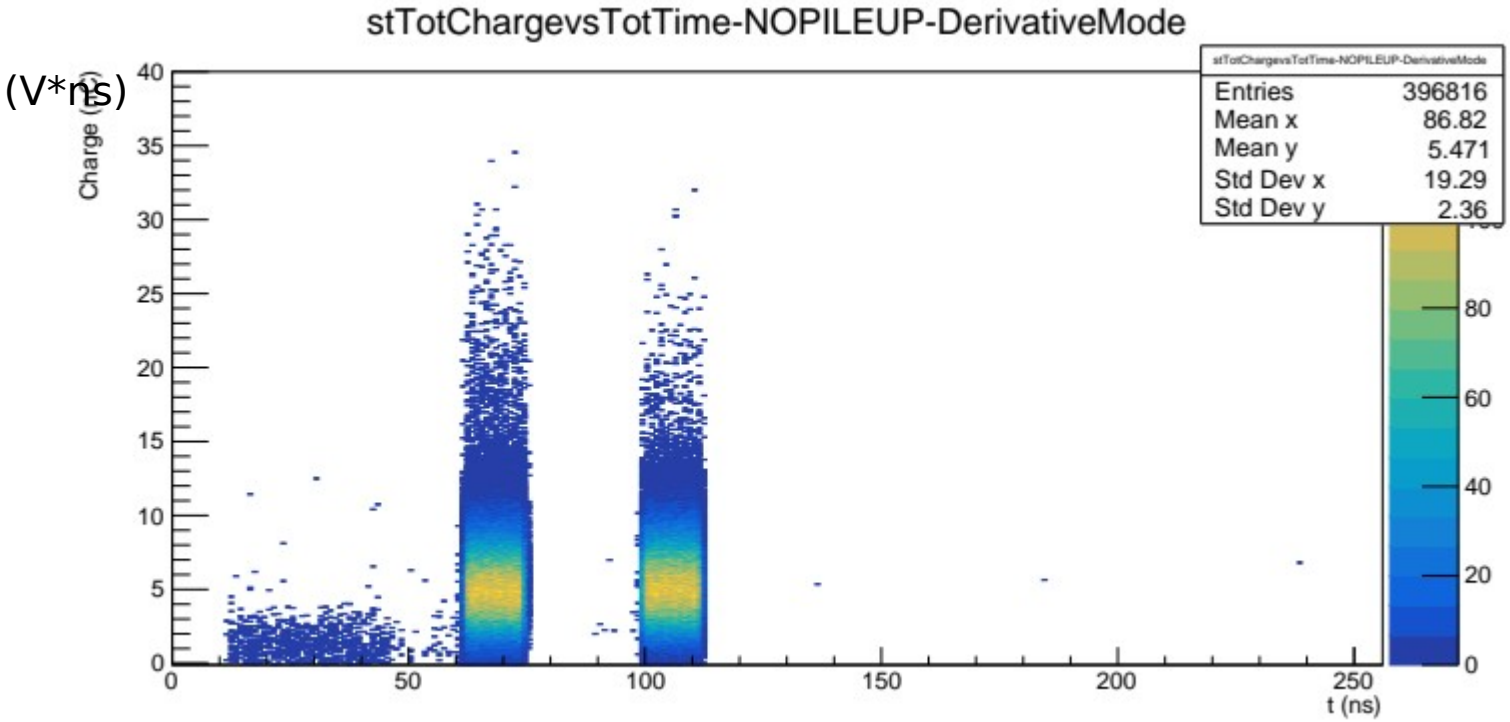
Pileup events:
The presence of two or more peaks makes a wrong reconstruction of time and charge



n.b: they can be even in the trigger columns

Results, 3

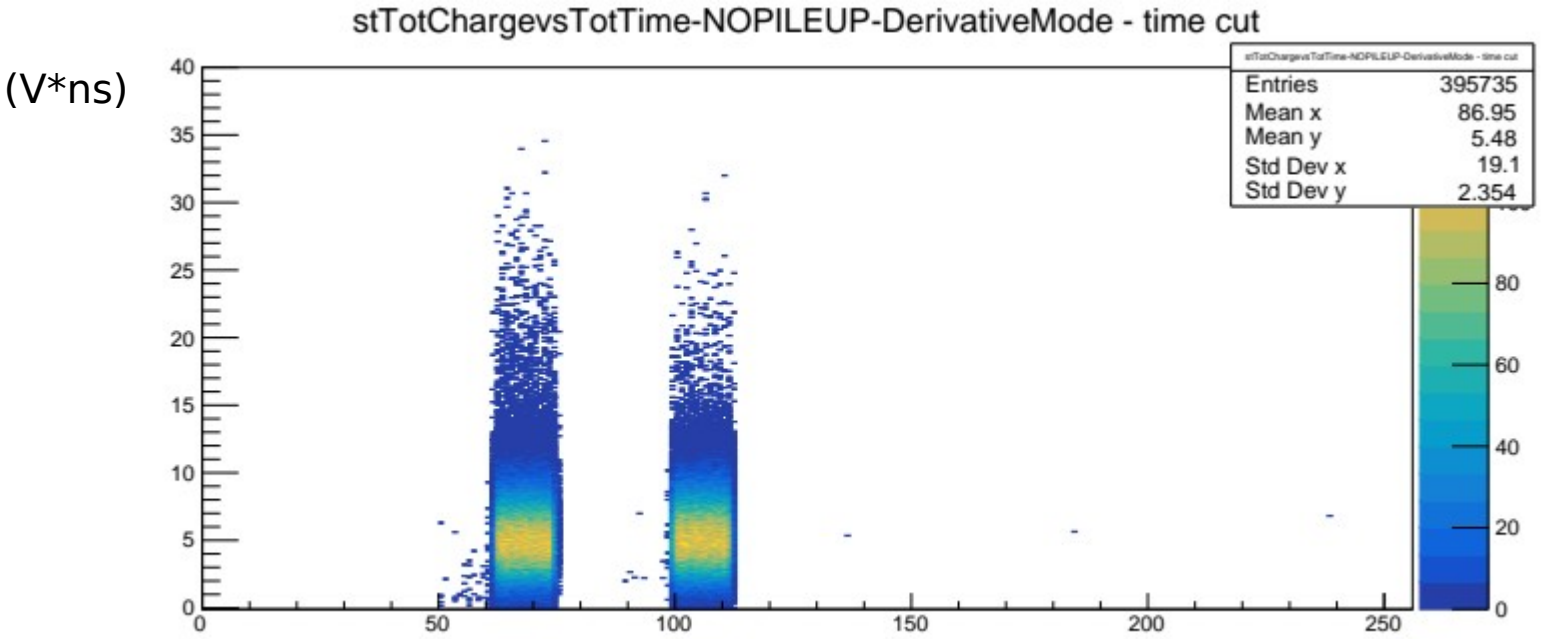
- Applying the **derivative method**, pile up events are filtered out, as it is visible mainly after the “columns”:



With a **removal of 0.8%** of the total events

Results, 4

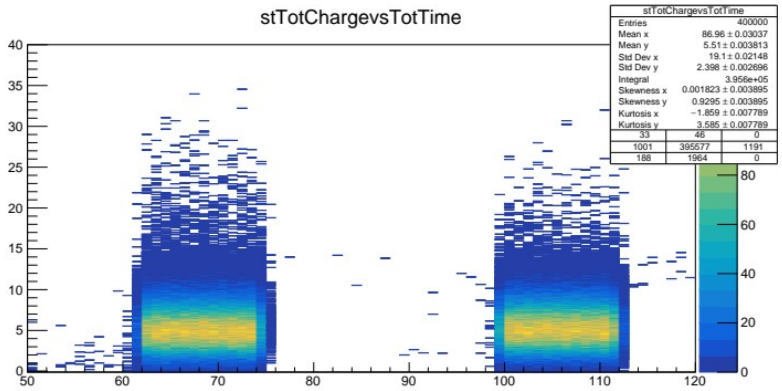
- In first analysis, it is possible to filter out **hidden pileup** applying a cut of **t<50ns**



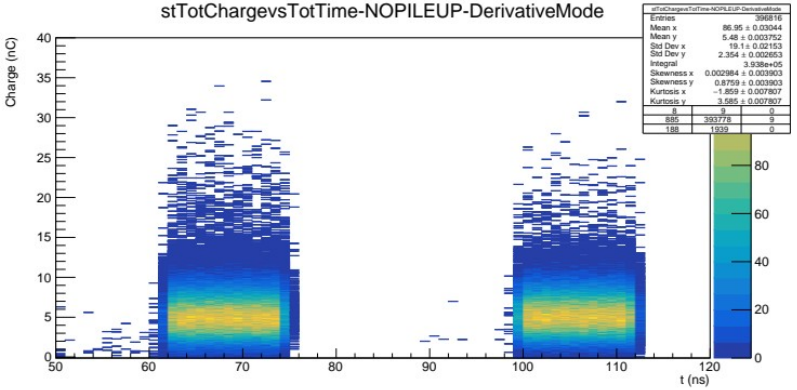
with a **removal (derivative + cut) of~ 1.1 %** of the total events

Results, 5

- The method is also able to filter out pile up events in the trigger “columns”:



No pile up filter,
events with 50 ns < t < 120 ns: 395577



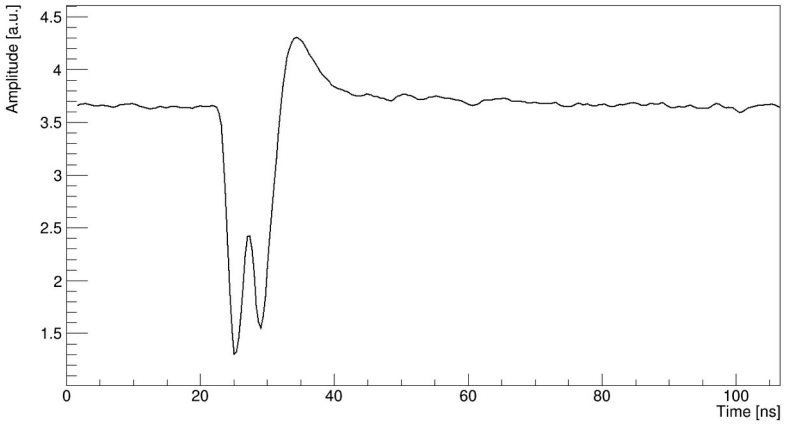
With pile up filter,
events with 50 ns < t < 120 ns: 393778

with a **removal in** *50 ns < t < 120 ns* **of~ 0.5 %**

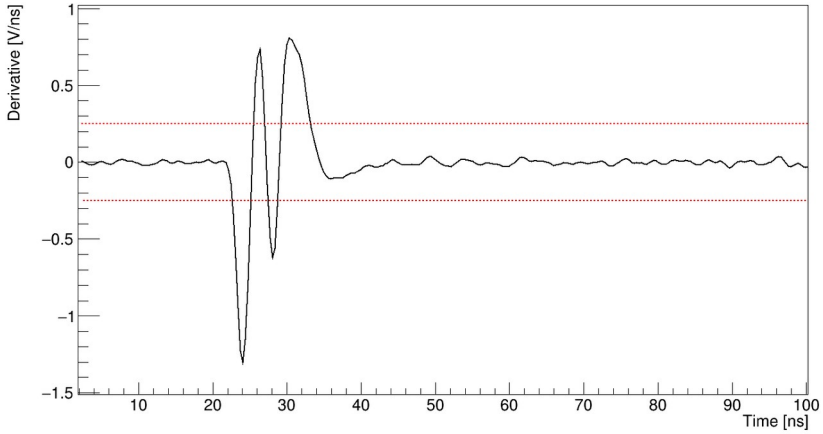
- This method is more powerful than just a time cut.

Comments

Pile up events which are temporarly very close are succesfully discriminated.



derivation

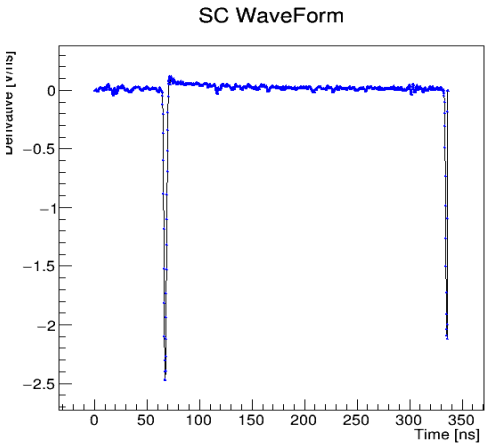


Credits to Roberto Zarrella

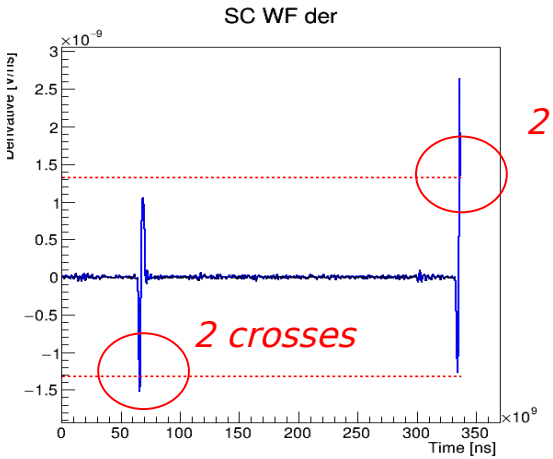
More than **4 crosses**: pile up event

Comments

Some pile up events are not well discriminated when one peak is between the end of the SC time window. The derivative is bad reconstructed and the threshold is too high to have intersections with the signal.



derivation



Only **4 crosses**: recognized as single event

- The solution could be to decrease the value of the threshold.
- However, the presence of this events is very low, **~ 0.004%** of the total

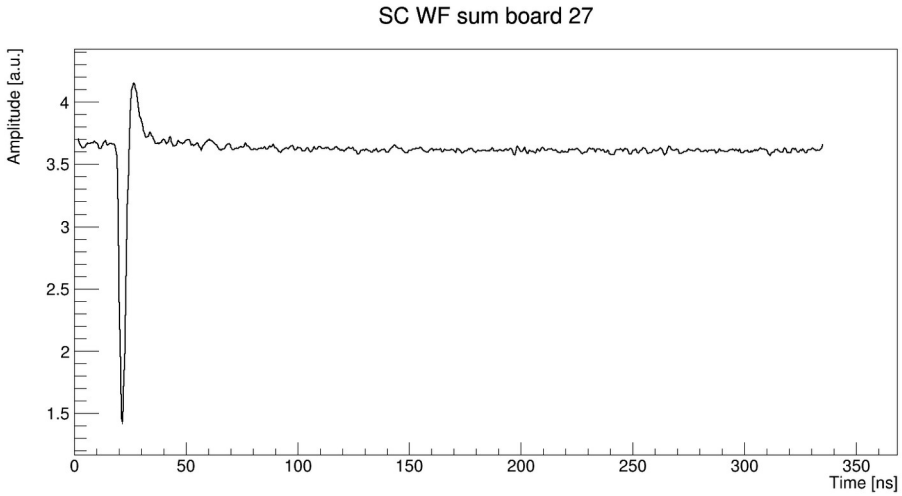
Conclusions

- The derivative method revelead as valid for pile up discrimination, with a removal of **~ 1 % of total events**
- Applying this method on SC, we prevent a bad propagation on further detectors, such as TW
- Improvements on the algoritm could increase the performace, such as new threshold values or integration with other methods

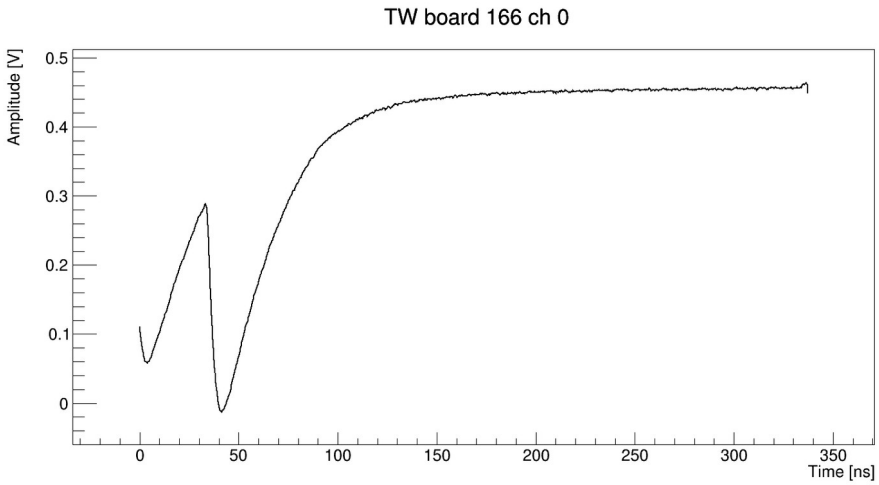
Backup slides

“hidden” pile up events

As we said, there are events which are seen as single in the SC because the previous peak is before the SC time window. Instead, seeing the signal of the same event on the channel of the TW, all peaks can be found.



SC: 1 signal



TW: overimposition of 2 signals

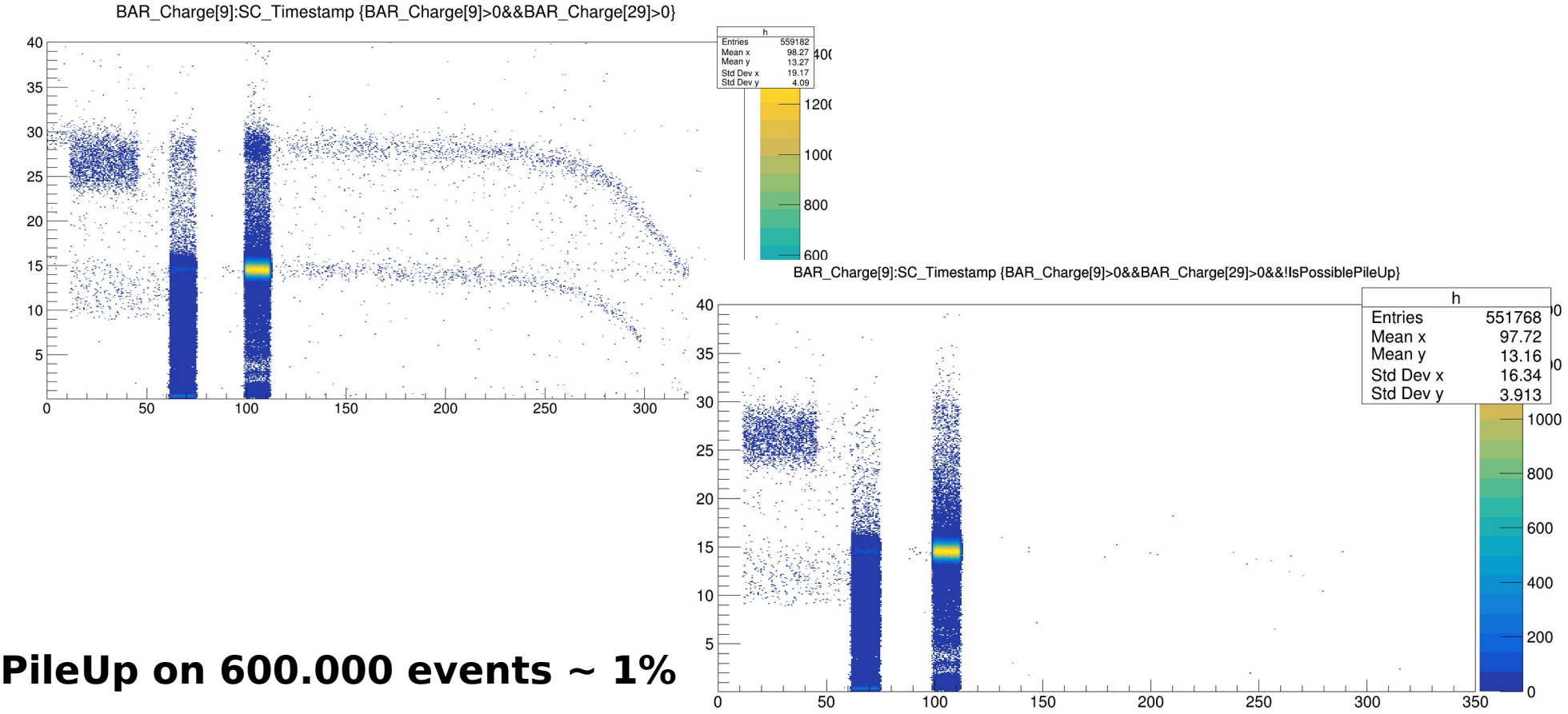
Credits to Roberto Zarrella

- To disentagle this pileup, an algorithm which also includes the signal in the TW can be introduced.
- However, in first analysis, just a temporal cut on SC with t<50 ns could be enough because they rapresents little % of all the events.

example

DataGSI2021/data_test.00004310.physics_foot.daq.RAW._lb0000._FOOT-RCD._0001.data

- Both ***minimum bias trigger*** and ***trigger fragmentation***



PileUp on 600.000 events ~ 1%

Credits to Roberto Zarrella

Implementation

The method was developed by Roberto Zarrella and then implemented in SHOE in the branch “pileUpStudy”

In particular, we introduced the function

Bool_t TASTrawHit::CheckForPileUp(std::vector<double>* w_ptr, std::vector<double>* t_ptr, Int_t event)

which modifies the private attribute bool **pileup** of the classes

TASTrawhit (class of the single waveform signal from SC)

TASTntuRaw (class of vector of waveform signals and “SuperHit” which is the sum of the 8 channel ones)

TASThit (class of the object hit, which takes physical information from “SuperHit” waveform, such as time, charge ...)

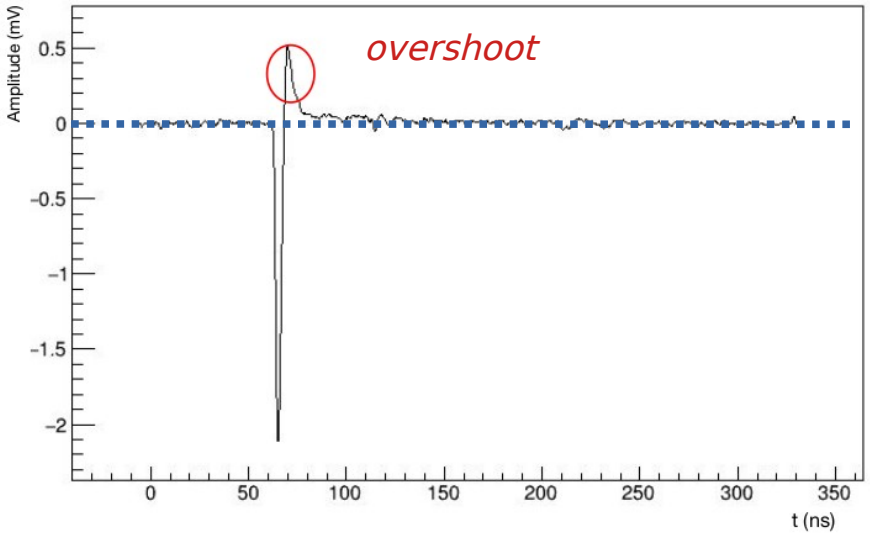
TASTntuHit (class of vectors of hits)

Charge method, 1

Another method to discriminate pile up is the **integral method**.

- Integrating the signal of the SC, it is possible to obtain the value of the **charge** of the crossing particle, from which discriminating single and pile up events.

- The signal integral is computed excluding the samples with ***amplitude > baseline + 2 st.dev***

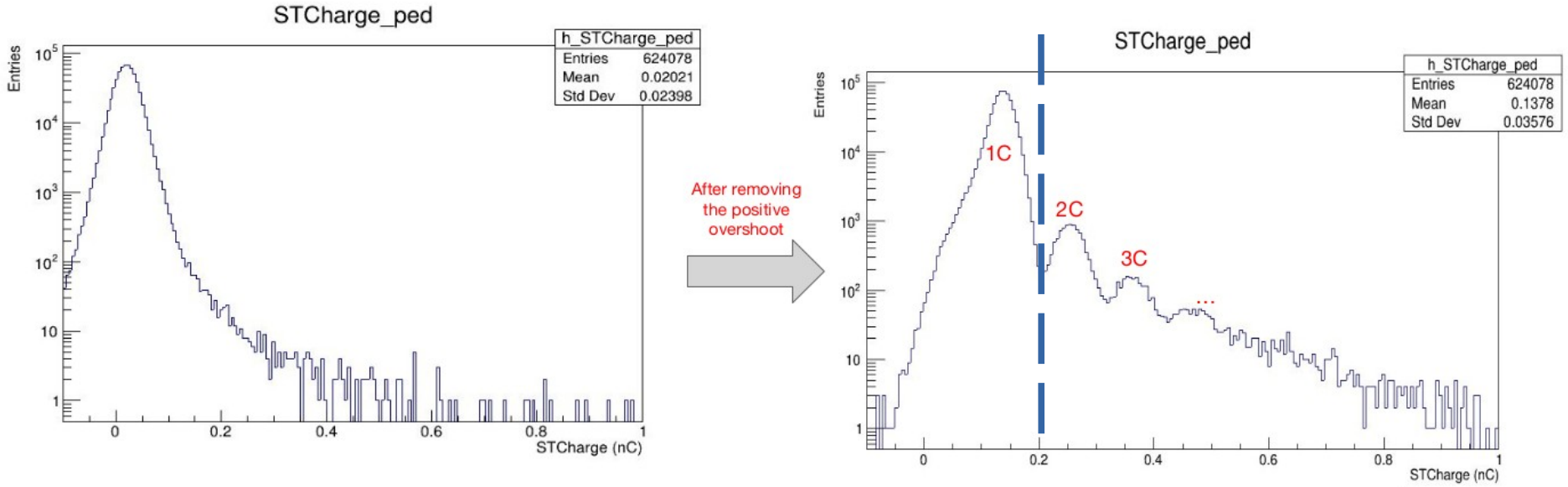


Credits to Marco Bon

- In this way we remove the positive **overshoot**, a positive peak of the signal due to electronics which is different from event to event.

Charge method, 2

Removing overshooting, it is possible to distinguish charges of single events and pileup ones as separate peaks in the histogram of charges.



Credits to Marco Bon

To filter out pileup events, a cut can be introduced in the value of charge. In this case as c < 0.2 nC.

This method is developed by Giacomo Traini et al.

And under implementation on Shoe