

INTT various updates

Cheng-Wei Shih,
National Central University/RIKEN

Dec 6th, 2024
INTT meeting



國立中央大學
National Central University

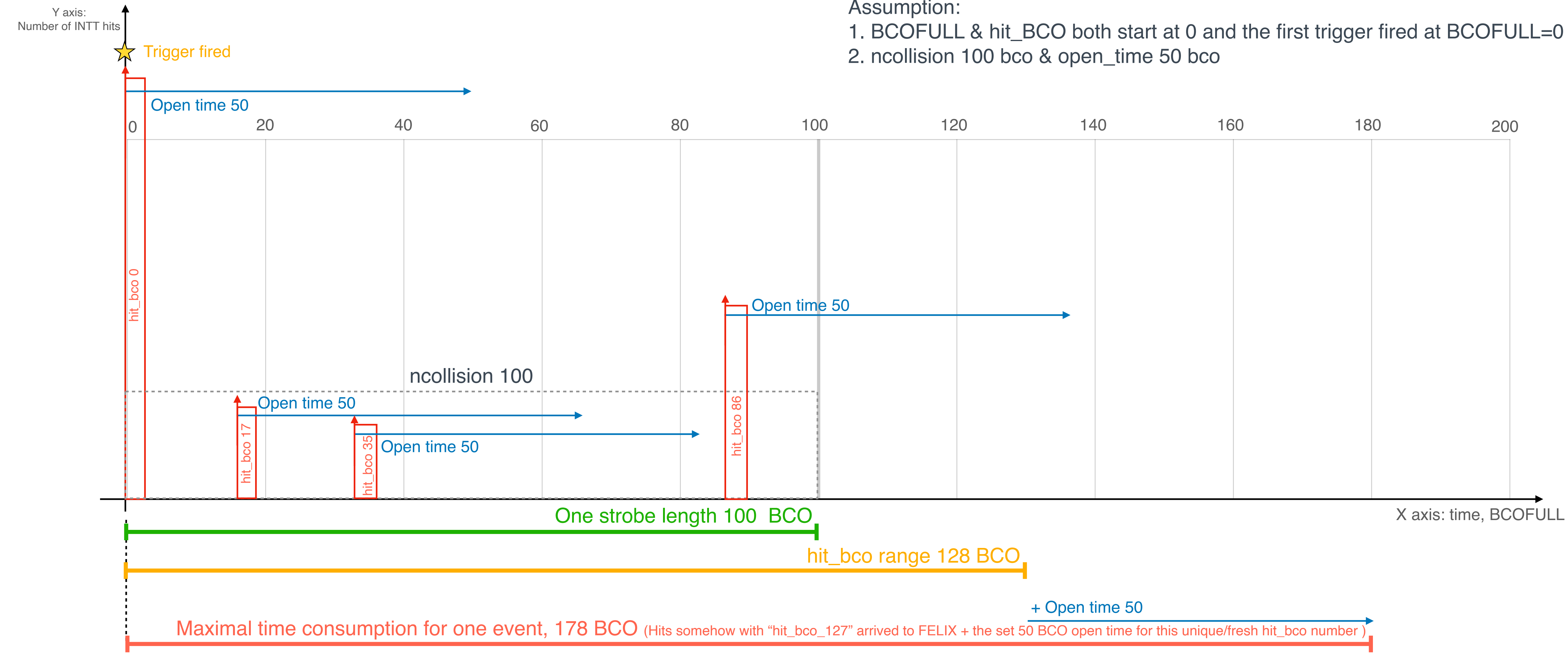


Note: the hit transmission from chip to ROC: 1 hit / 1 bco

In single event

Assumption:

1. BCOFULL & hit_BCO both start at 0 and the first trigger fired at BCOFULL=0
2. ncollision 100 bco & open_time 50 bco



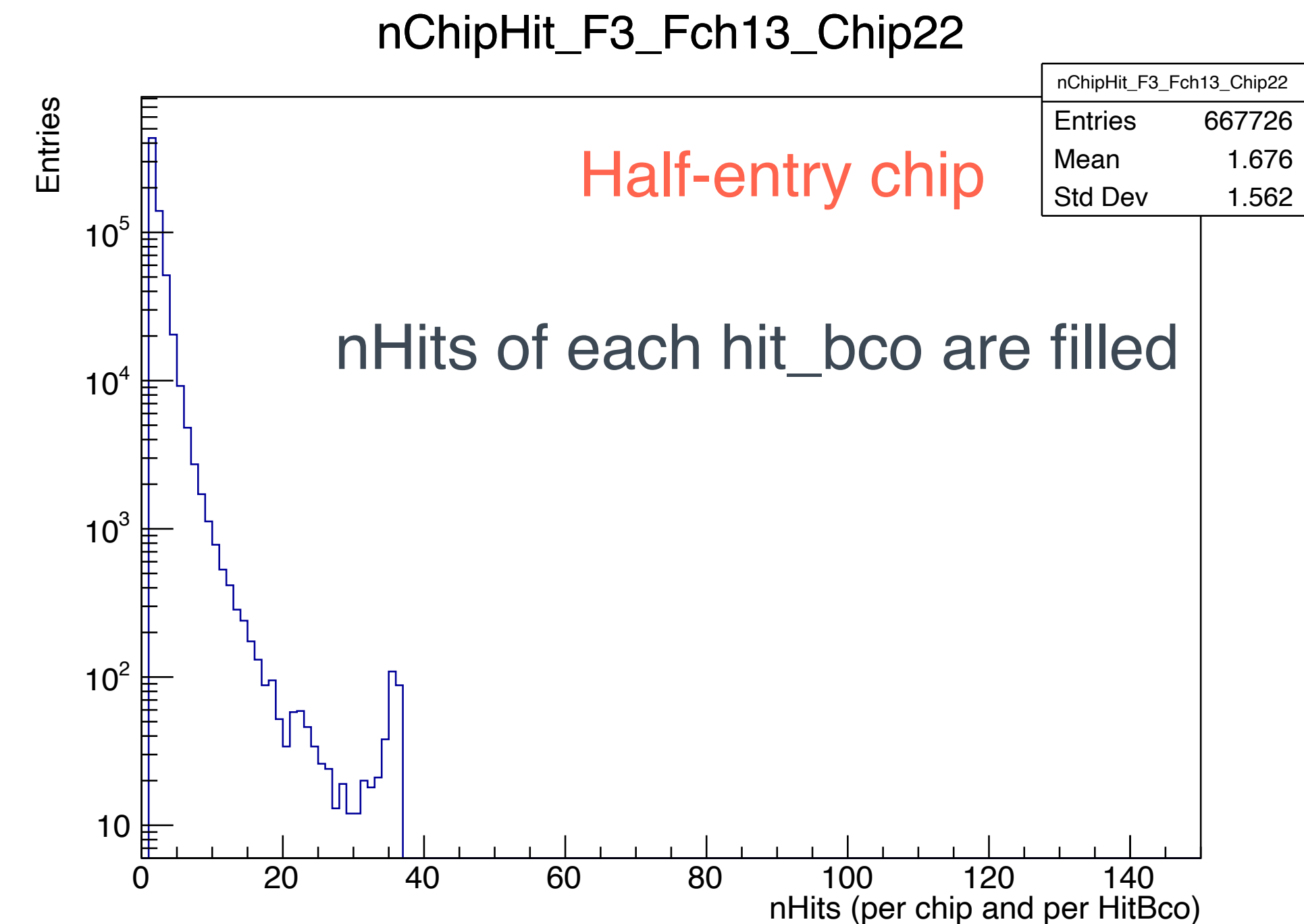
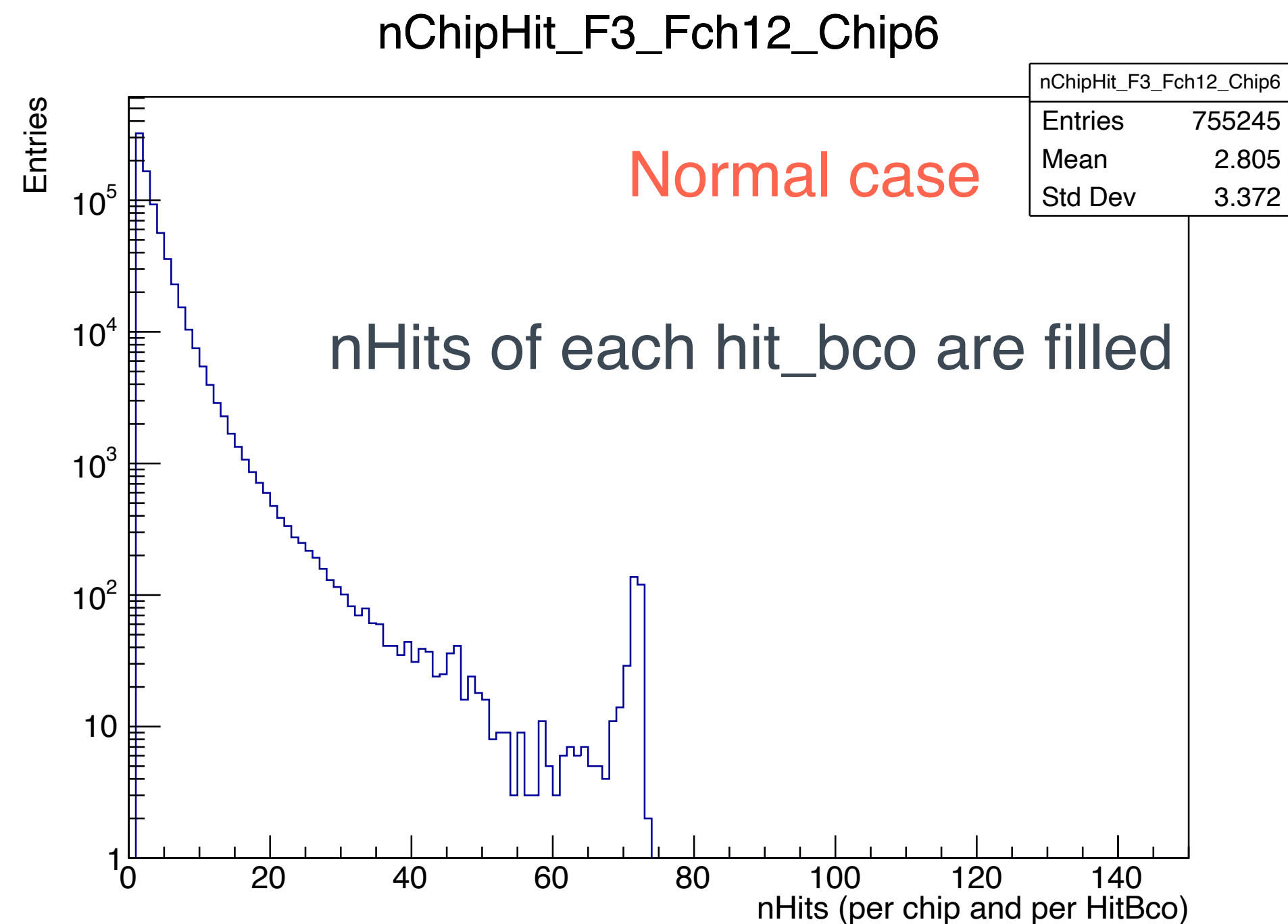
- Current concept: when the first hit with "hit_bco_A" arrives at FELIX, FELIX waits for the other hits with same "hit_bco_A" coming for "50 BCO", and if there is another hit with "hit_bco_B", the FELIX would open another "50 BCO" for the hits with "hit_bco_B"

Code can be found in [GitHub](#)

With HitQA and CloneHit Removal (CloneHit: same FELIX, FELIX_ch, chip_id, chan_id, hit_bco)

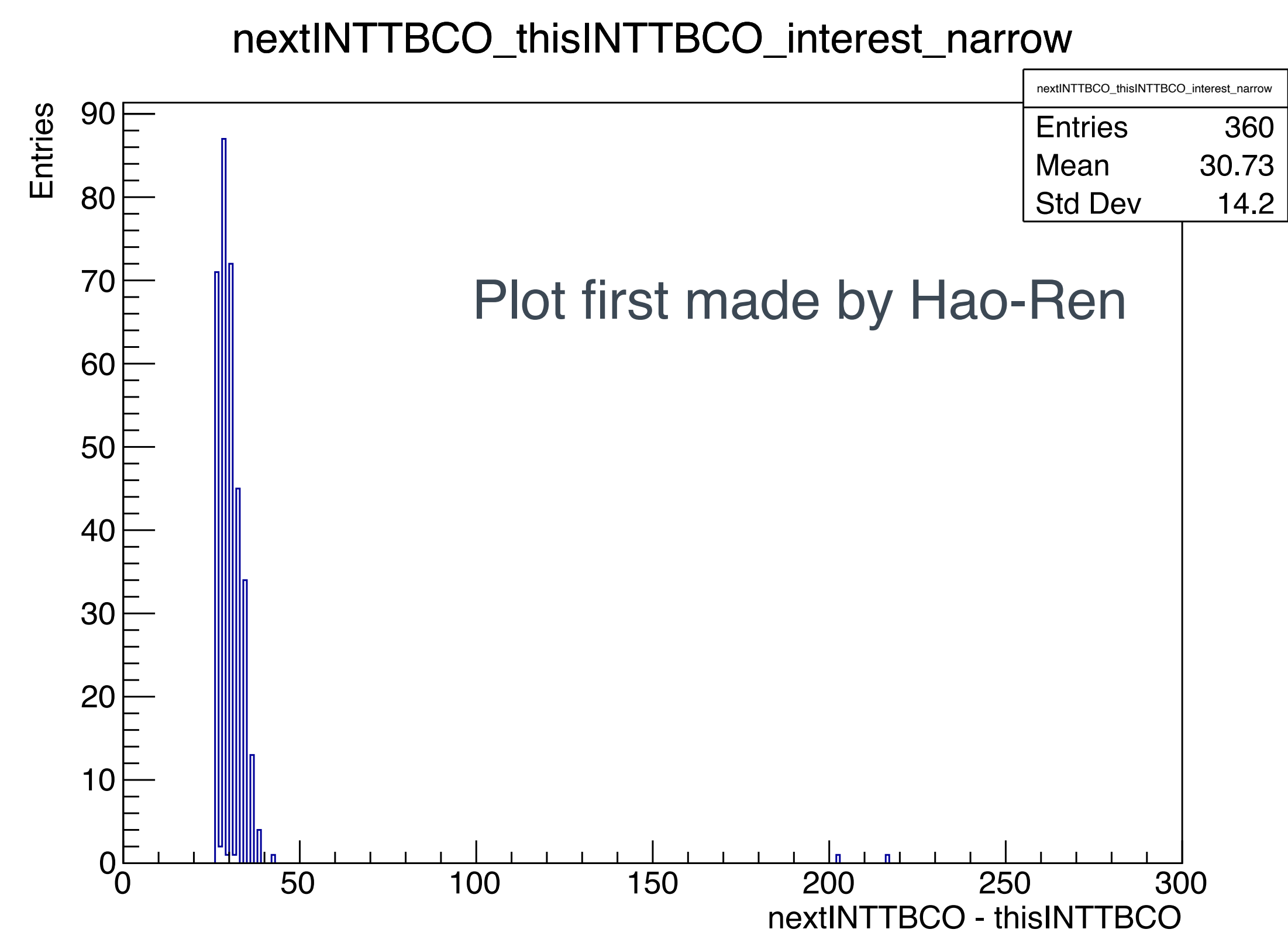
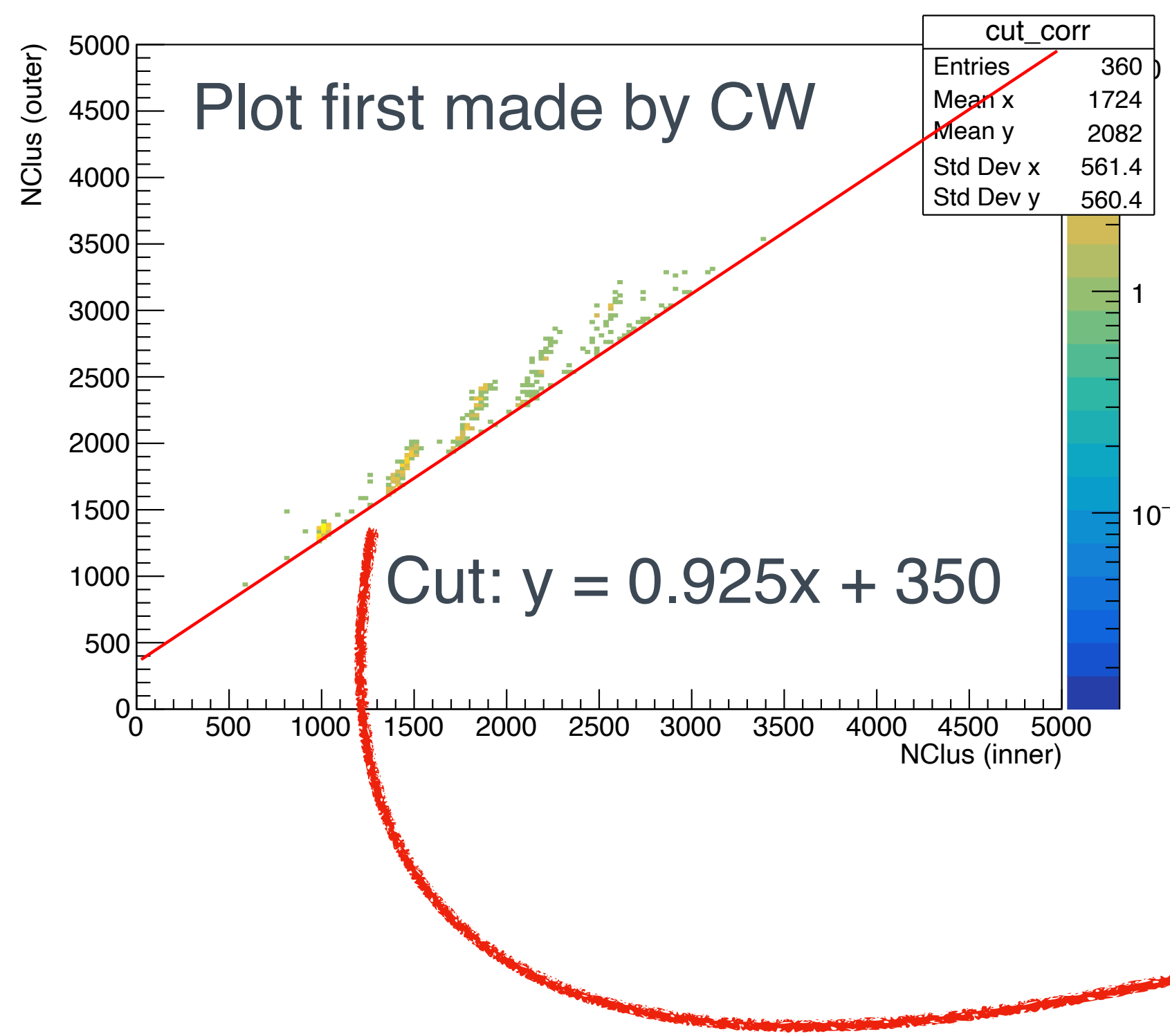
Count the number of hits of each chip, per hit_bco

INTT_open_time: 50 BCO (I assume, where to confirm?)



- The spike at nhits 74: hits been rejected due to the late arrival to the FELIX
- The maximal number of hits of each chip and per hit_bco is 74
- Half-entry chips have similar structures → Hit missing happened before FELIX (at chip)
- May not be relevant to the two spikes in the cluster phi size distribution (43 or 46 channels being adjacent)

Event of interest (EOI)



The very next events of the EOI are very close to EOI in time wise

Hypothesis: Hits in FELIX been assembled with INTTheader (INTT_bcofull) and sent out to the down stream. Since FELIX receives new trigger, the previous INTT_bcofull is overwritten. The hit assembly continues, but with the new INTT_bcofull

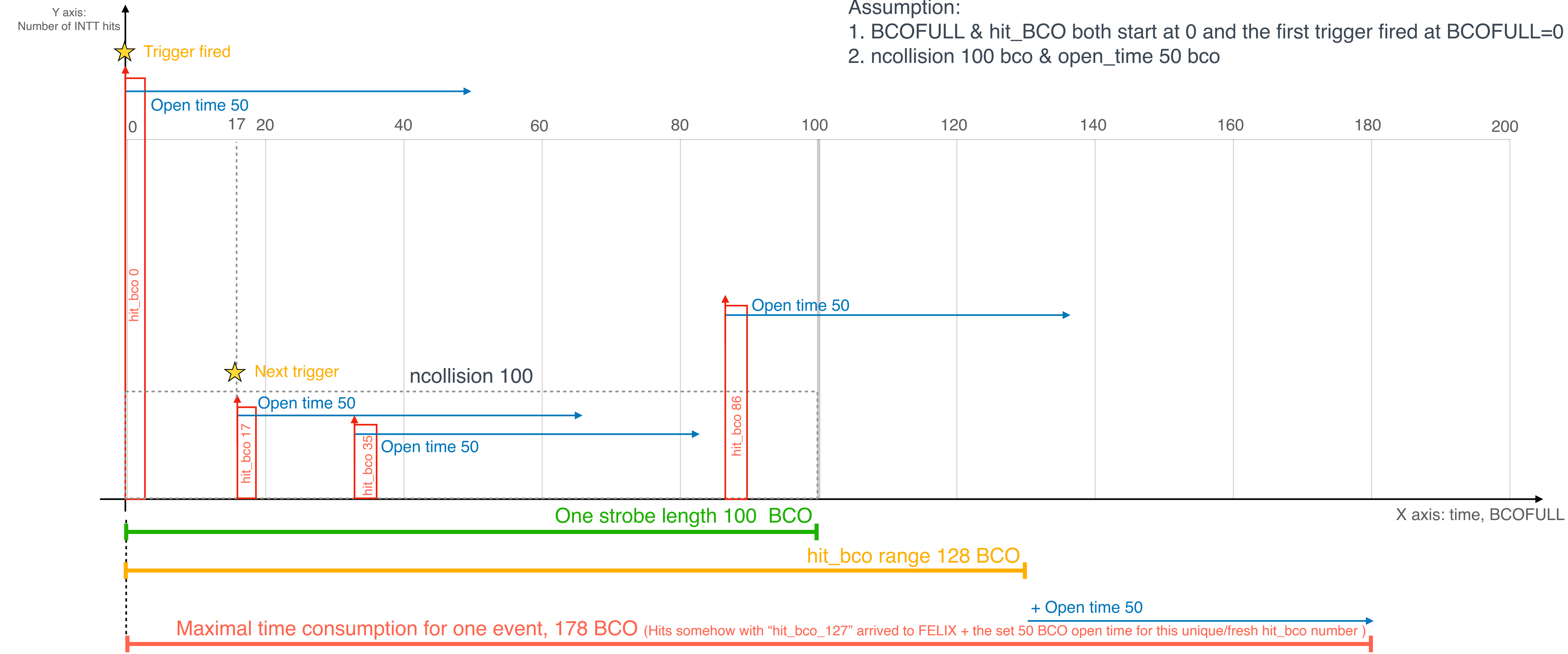
Can we probably just have a simple "BCOFULL_diff" cut?

Note: the hit transmission from chip to ROC: 1 hit / 1 bco

In single event

Assumption:

- 1. BCOFULL & hit_BCO both start at 0 and the first trigger fired at BCOFULL=0
- 2. ncollision 100 bco & open_time 50 bco



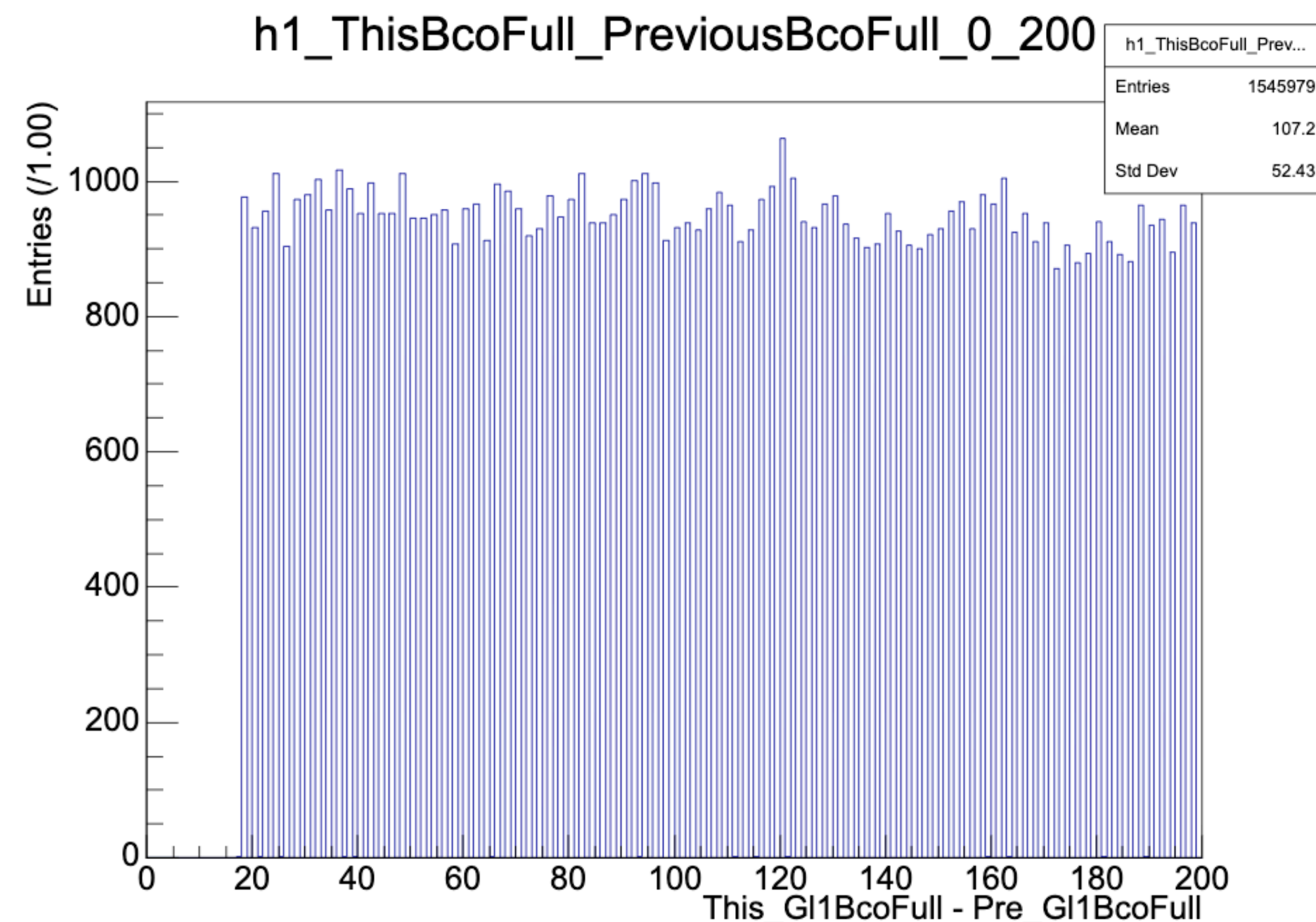
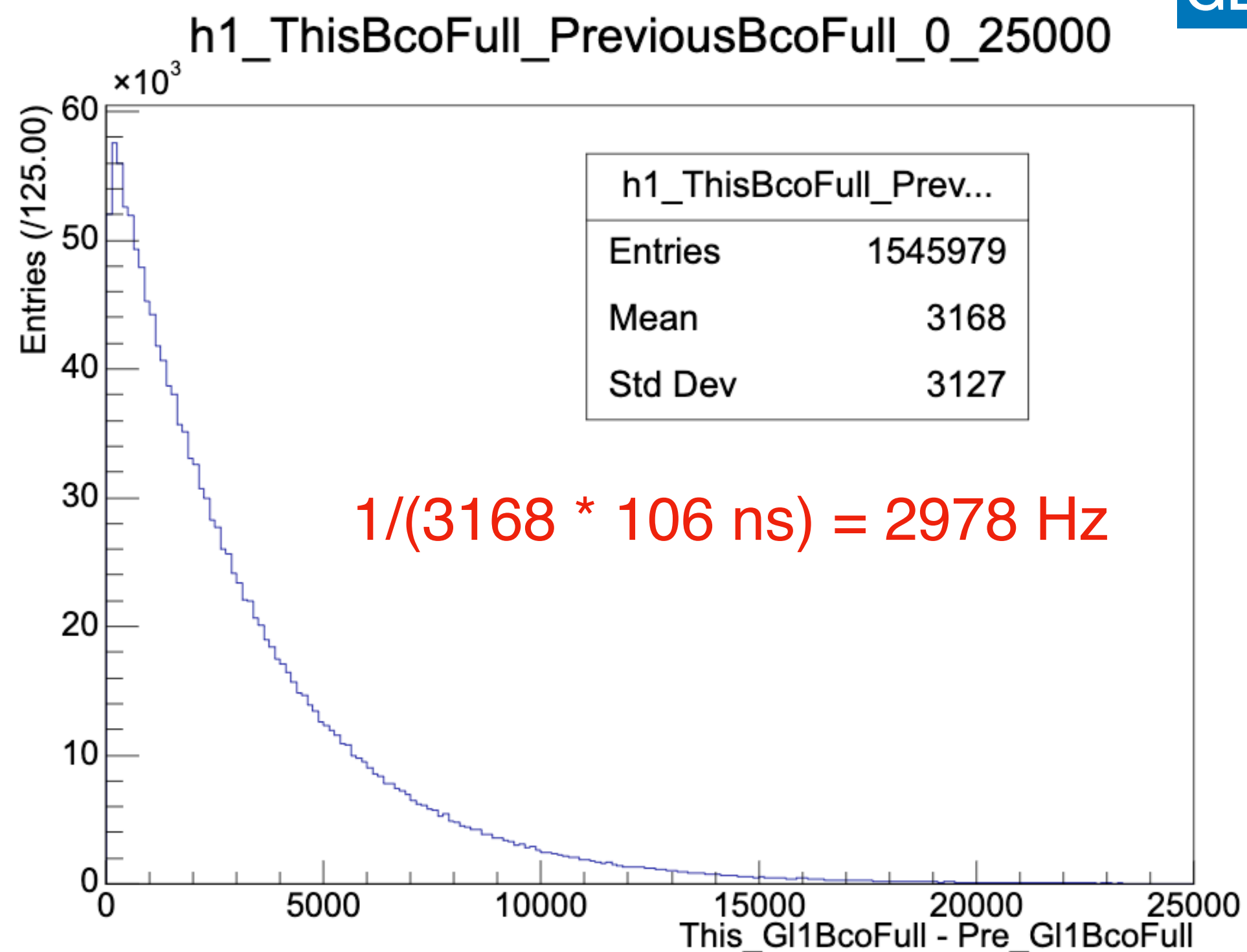
- Question 3.: As shown in cartoon, what if we have “hit_bco_0” in “this_event”, the FELIX is taking the hits with “hit_bco_0”, but the next trigger happened within the “open-time”? what will happen?

Event bco span

Code in [GitHub](#)

| Runnumber | run time (min) | nEvent | Rate (Hz) |
|-----------|----------------|----------|-----------|
| 54279 | 60.133 | 5842231 | 1619.253 |
| 54280 | 60.183 | 10610255 | 2938.331 |

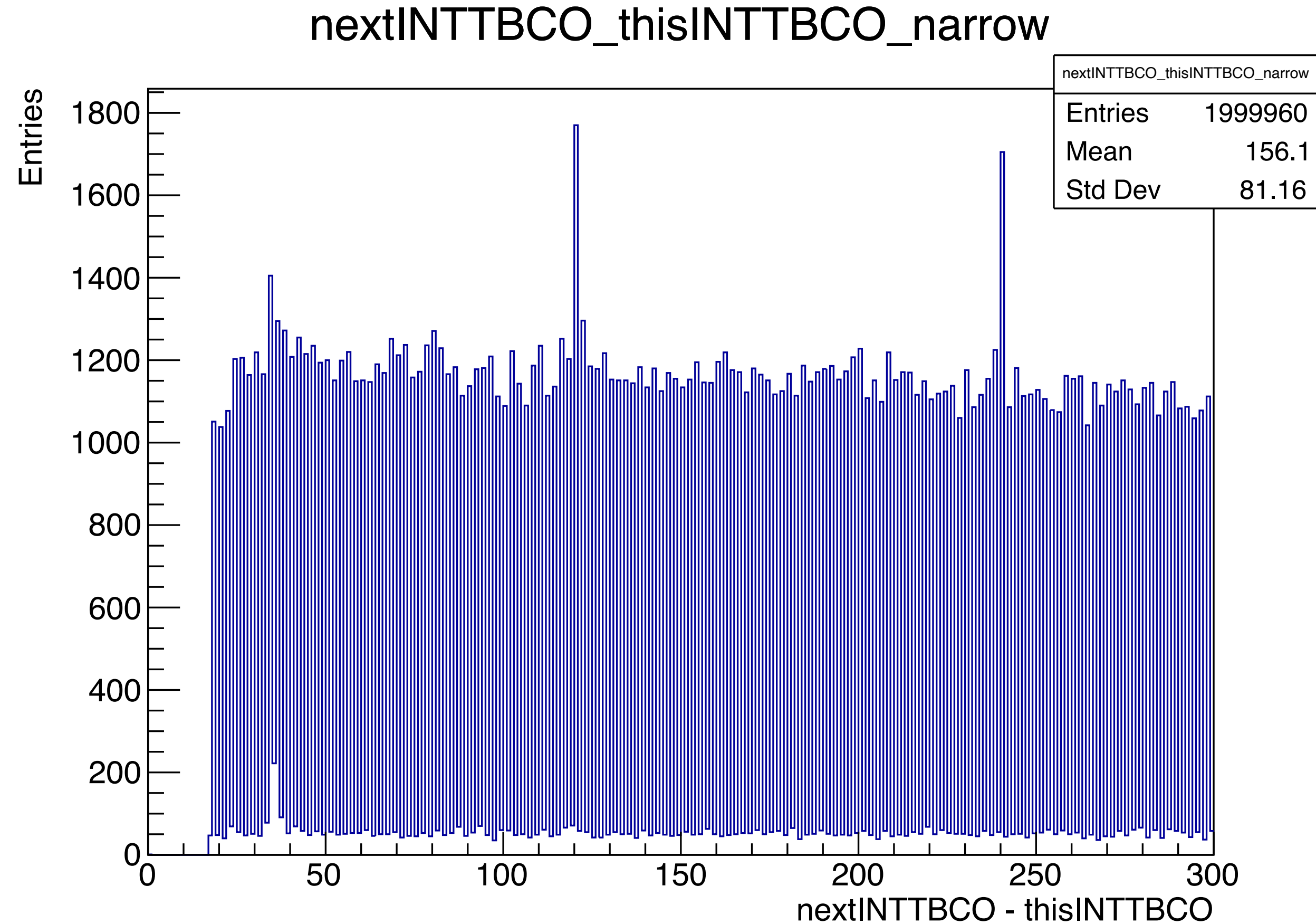
GL1BCO is used



Somehow the distribution event bco is different from what we expected
But it seems to be the case, at least, the average trigger rate is matched

Somehow run54280 has higher trigger rate than the previous run → could possibly by re-tune the scale-down factor

INTT BCOFULL (from "INTTEVENTHEADER->get_bco_full()")

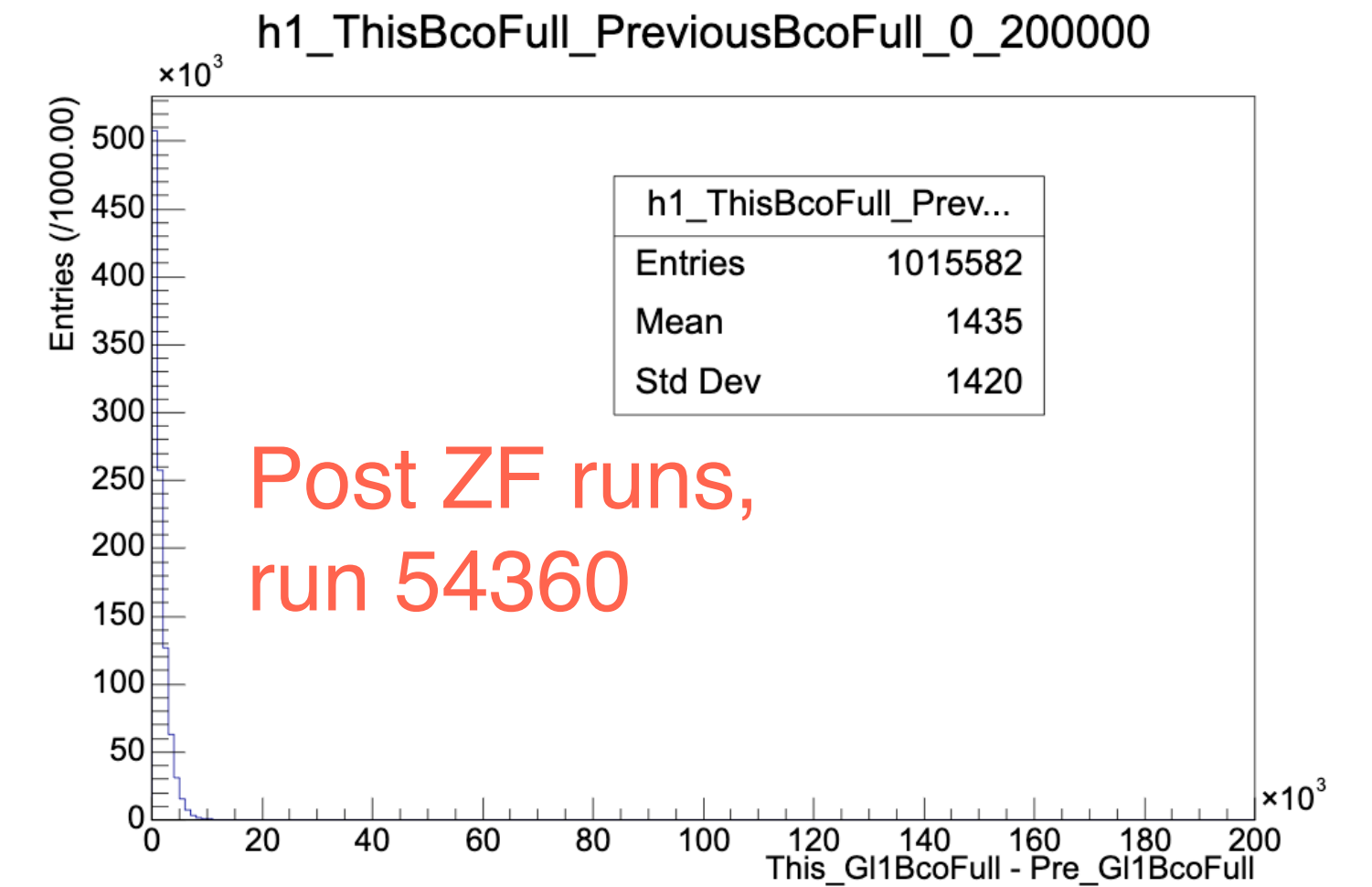
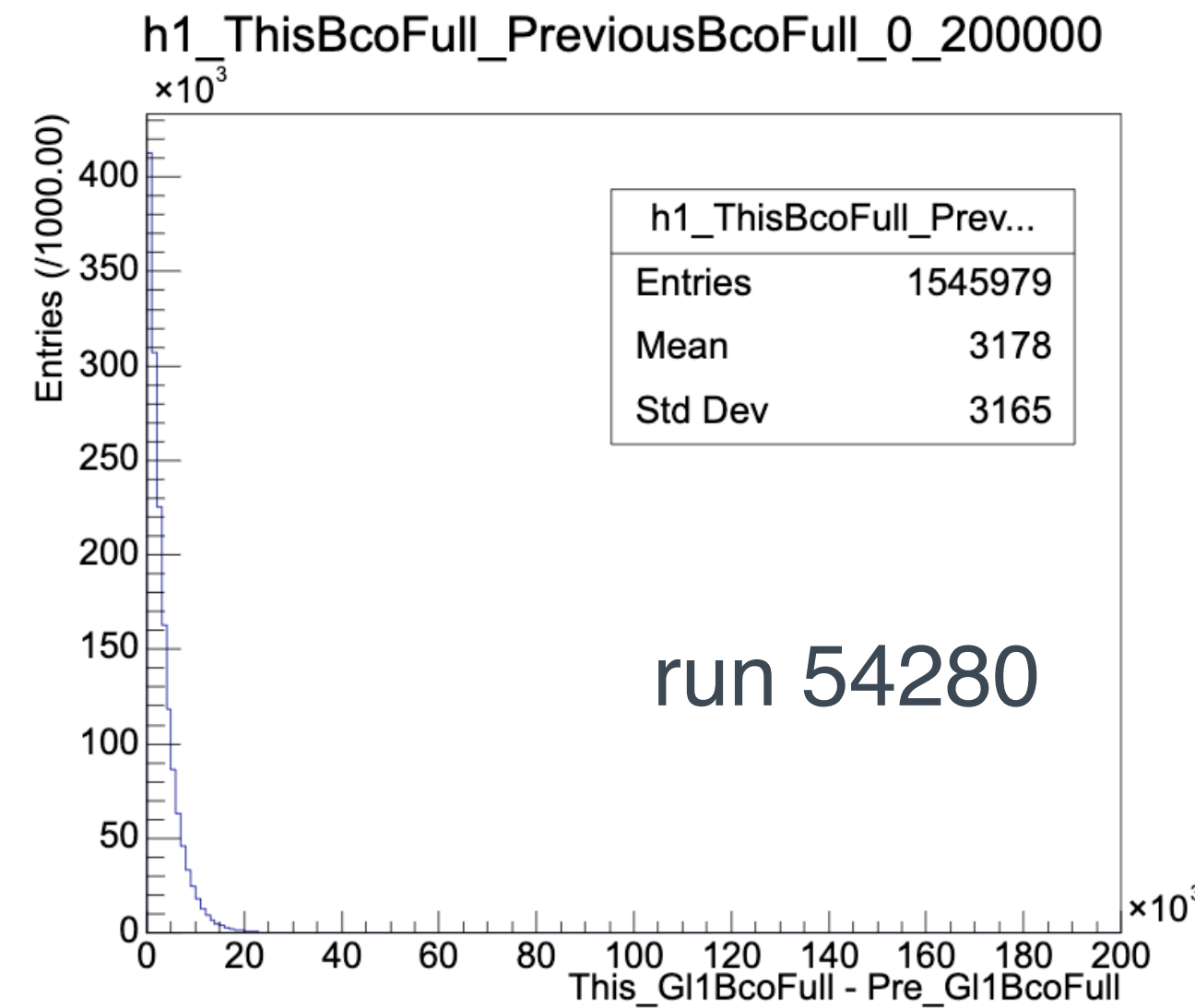
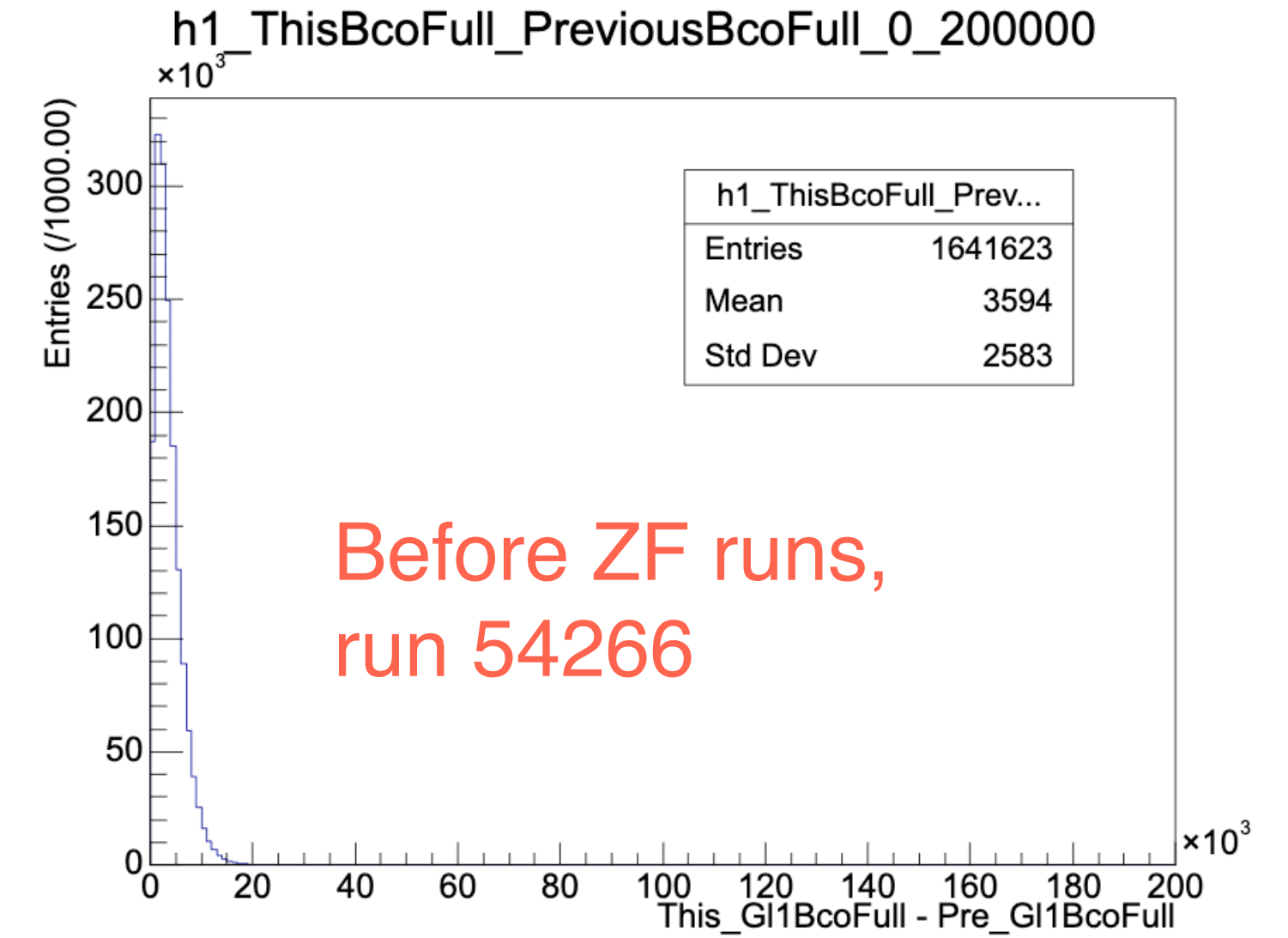
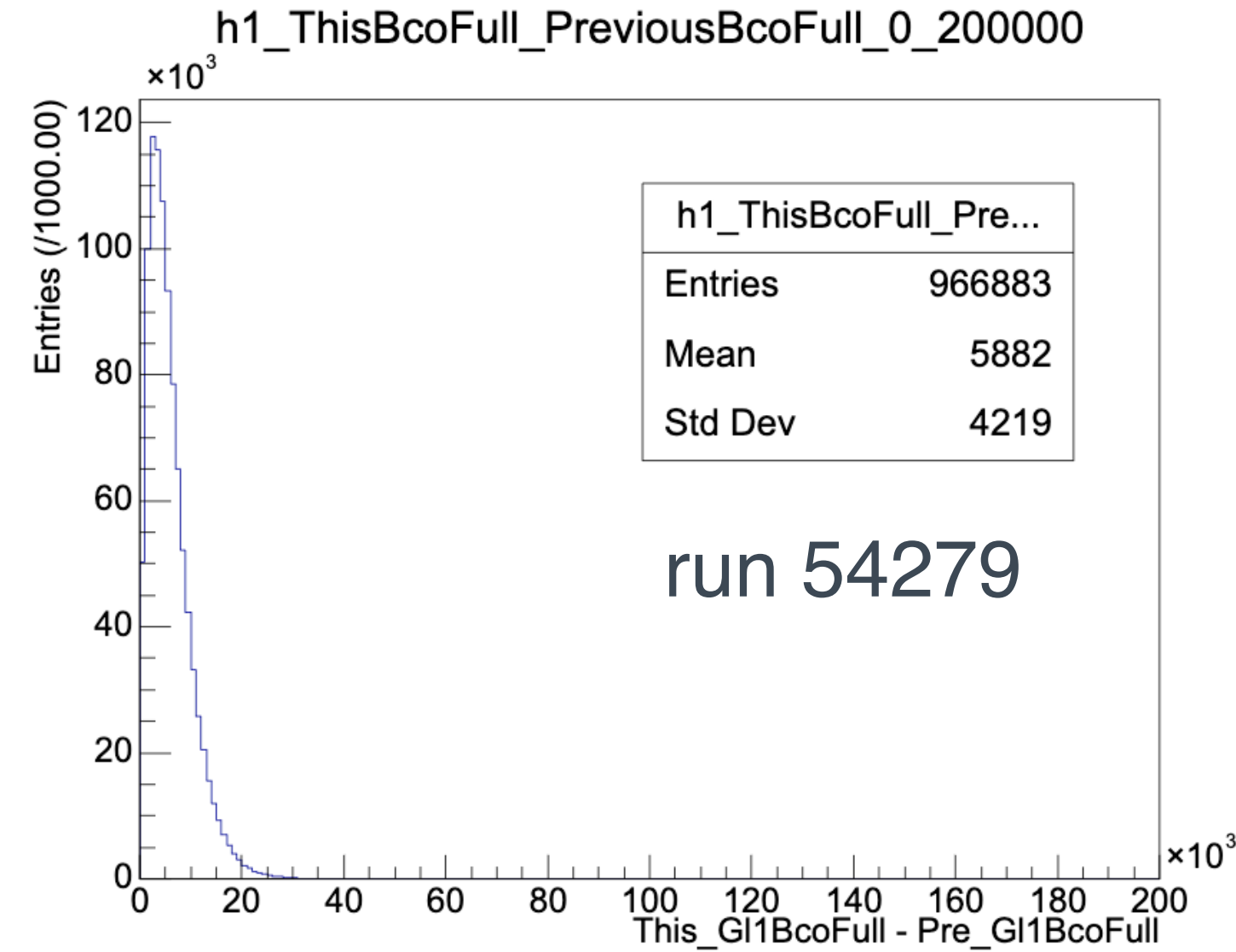
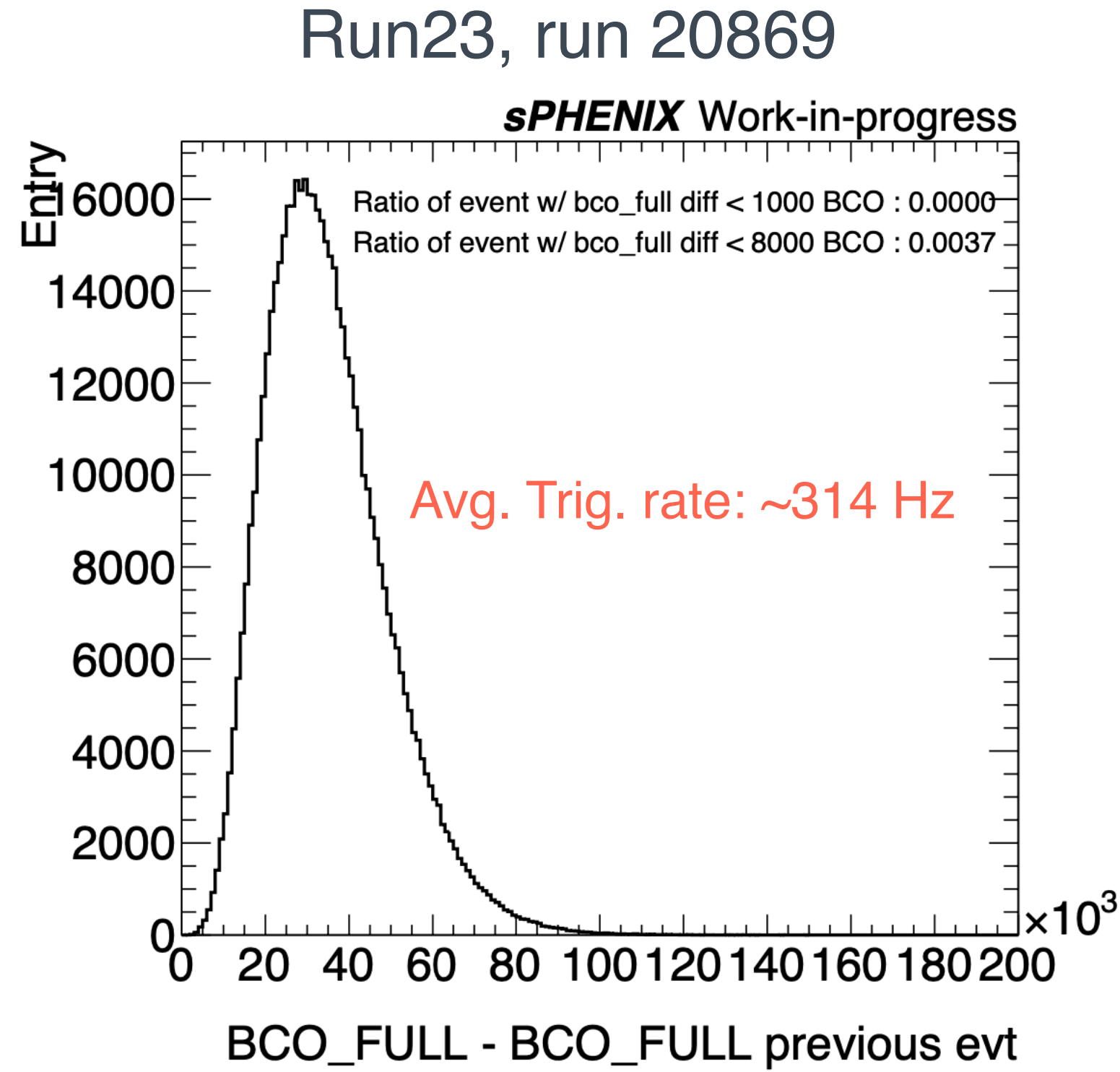


Still similar distribution comparing to that of made of GL1BCO

It seems that INTT FELIX servers don't deny the coming trigger signals even when the data processing is still ongoing

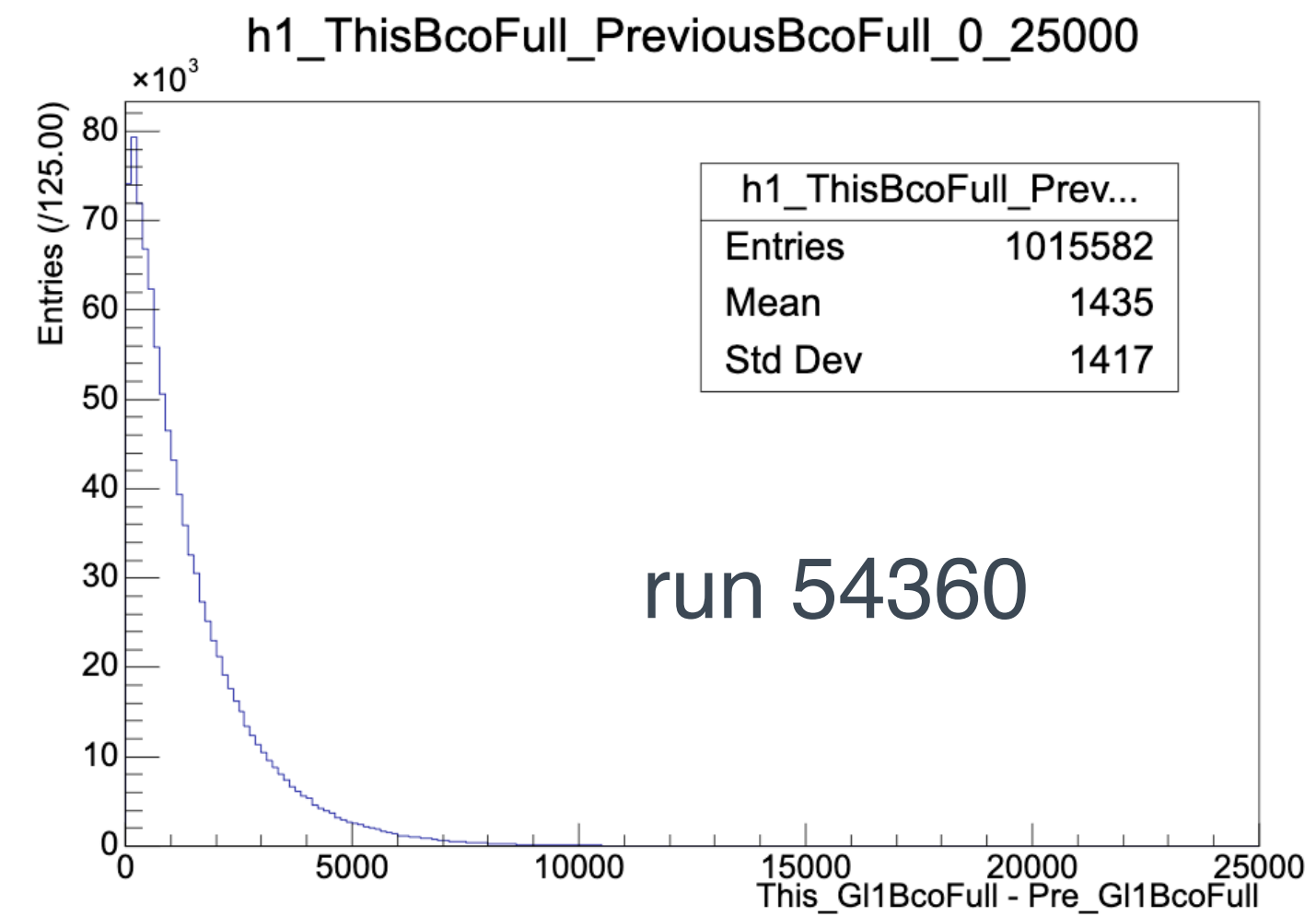
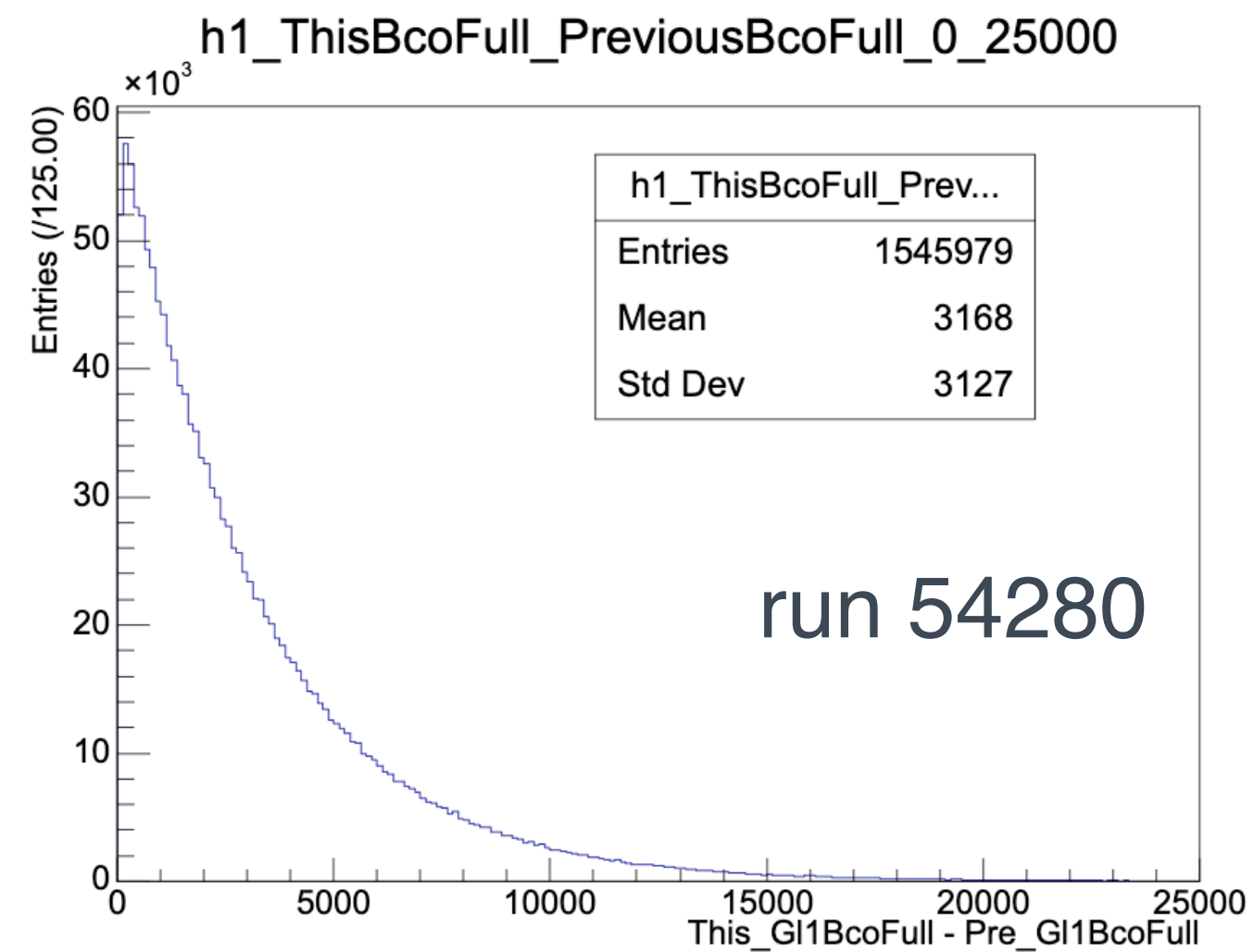
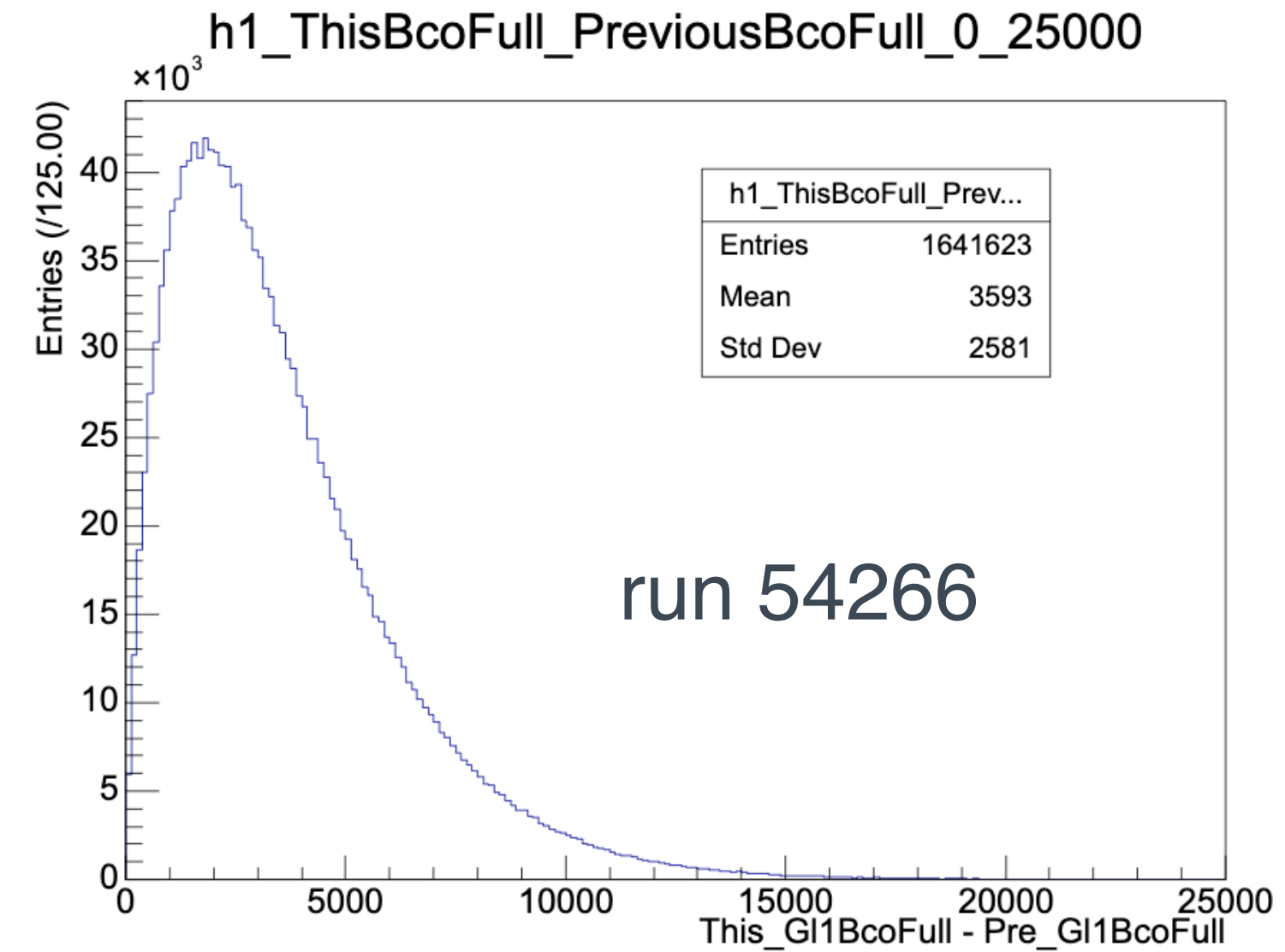
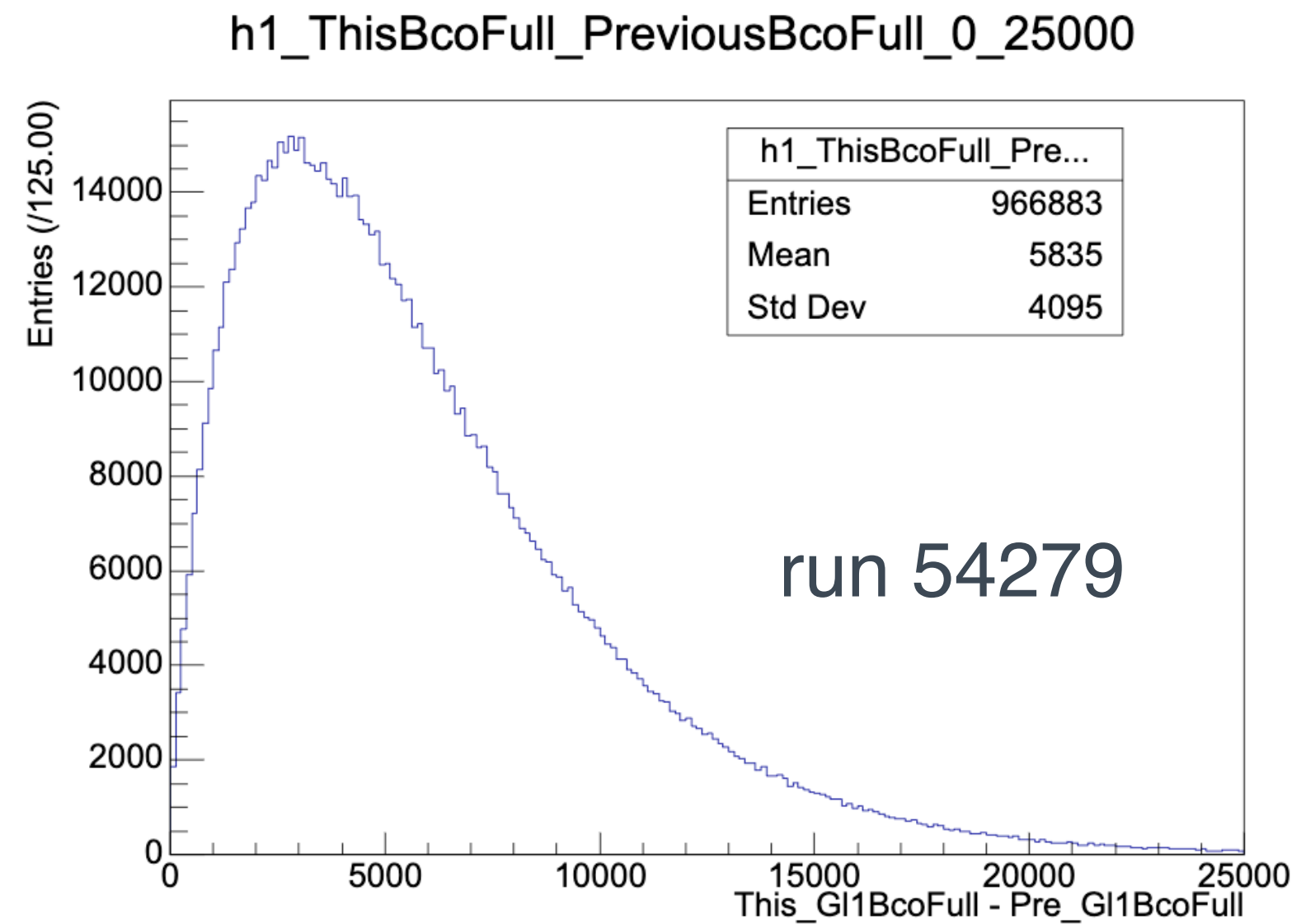
Event bco span, same scale

GL1BCO is used



Event bco span, same scale

GL1BCO is used



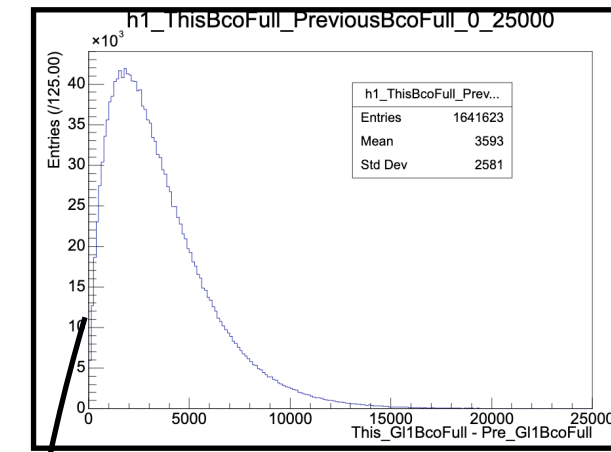
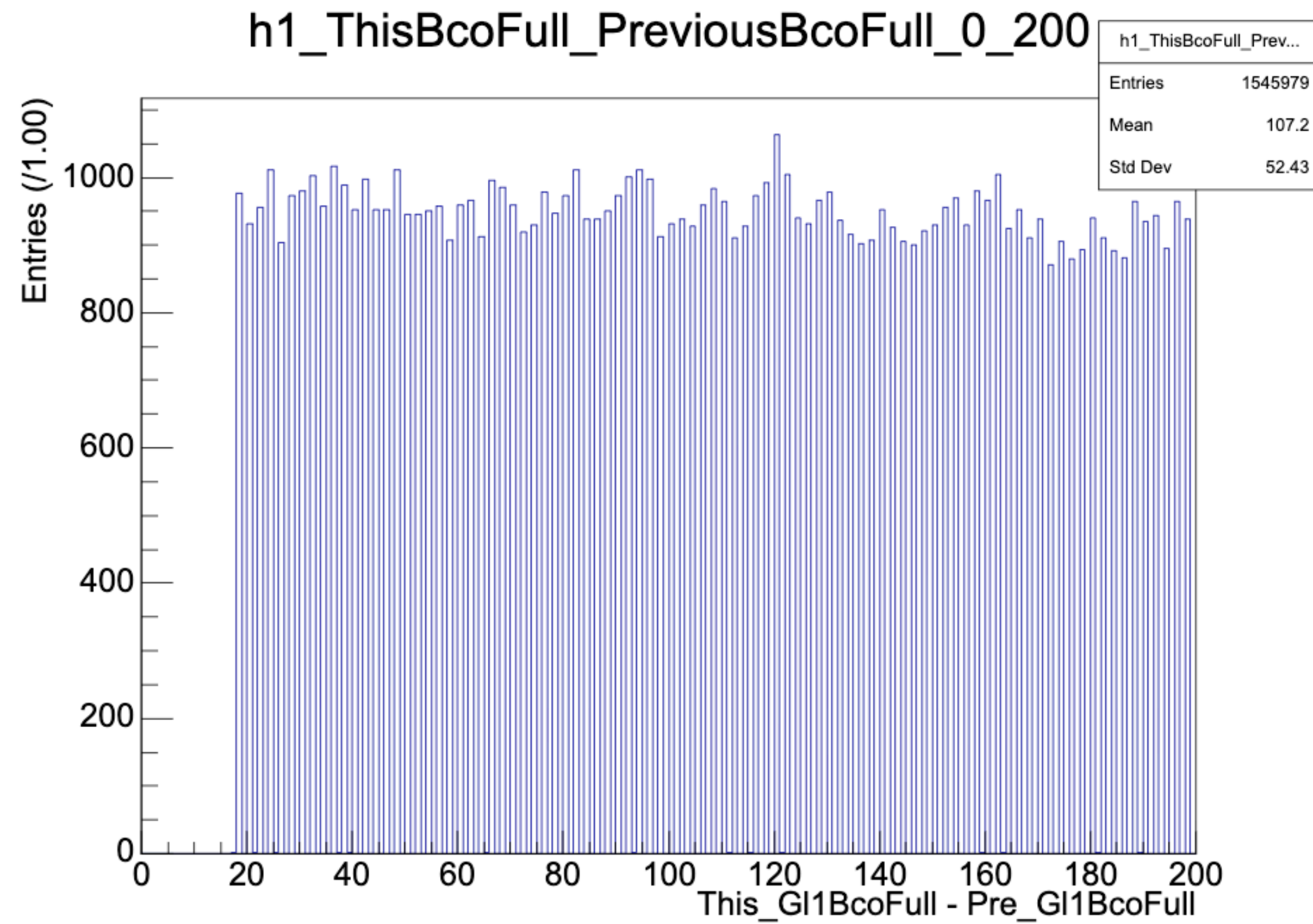
The distributions look reasonable

To have the Poisson distribution with large λ , to trigger rate has to be very low, few hundred Hz

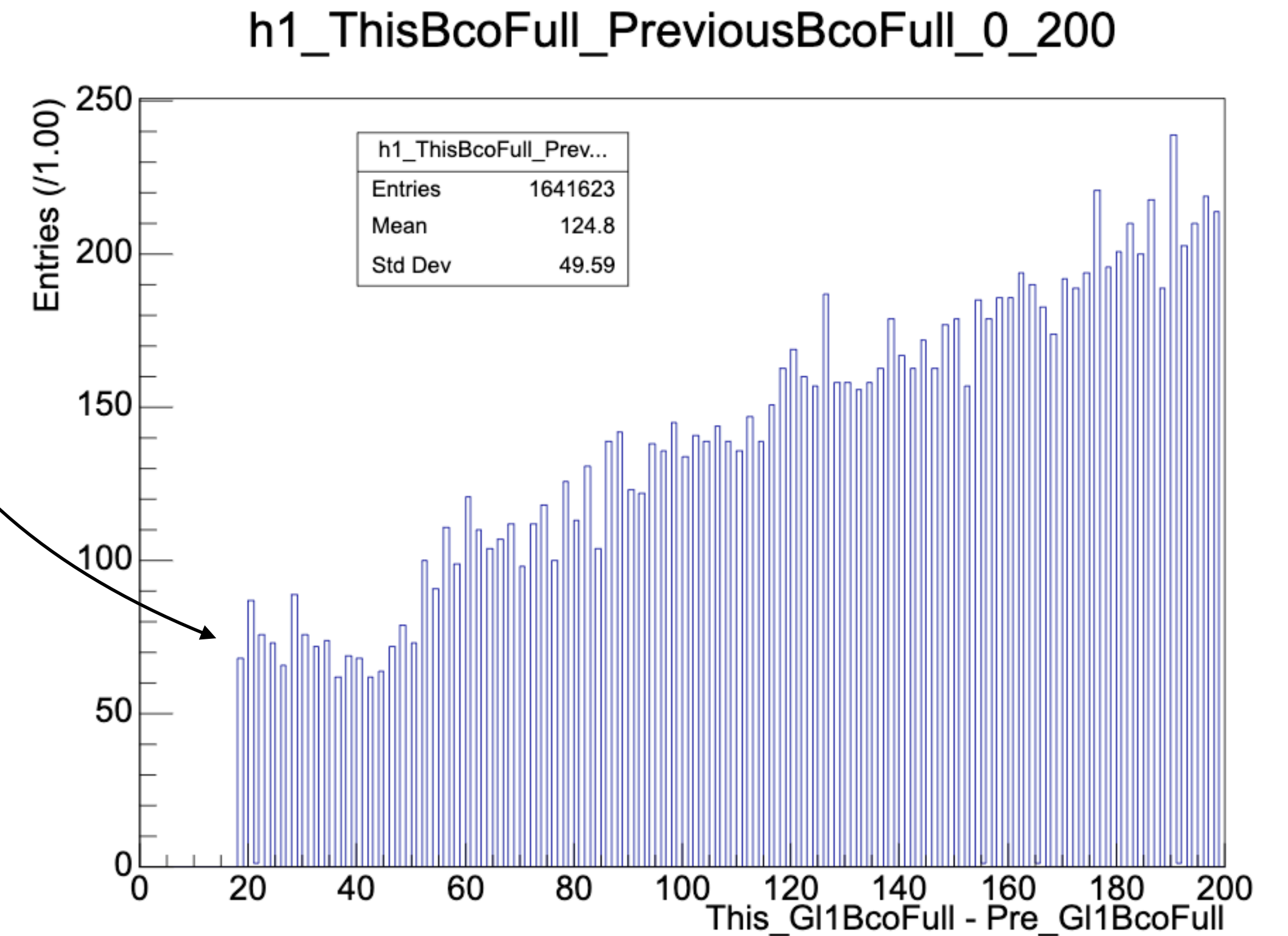
Event bco span (narrow)

GL1BCO is used

run 54280



run 54266



Have the same dead time, 17 BCO (It may be the default set in the GTM? not due to the busy signal?)

- The cut-off can be seen in the chip occupancy distributions, the work principle of open time is confirmed in some level
 - In run 54280, one chip can have up to 74 hits per event and per hit_bco
 - The half-entry chips have similar structures. Half of hits cannot make it be received by ROCs, but the time is still spent
- The very next events of the event of interest (EOI) are very close to EOI in terms of the time span
 - Hypothesis: the INTT_bcofull is overwritten when the next trigger is received by FELIX while FELIX is still proceeding the hit assembly with the rather late arrival hits corresponded to the previous INTT_bcofull
 - Would it be a severe problem in the p+p data?
- We can possibly have a INTT_bcofull_diff cut. Some good events might be cut since the distribution is different from what we expect due to the rather higher collision rate
 - With the check of multiple runs, the distributions of event_bco_span look reasonable

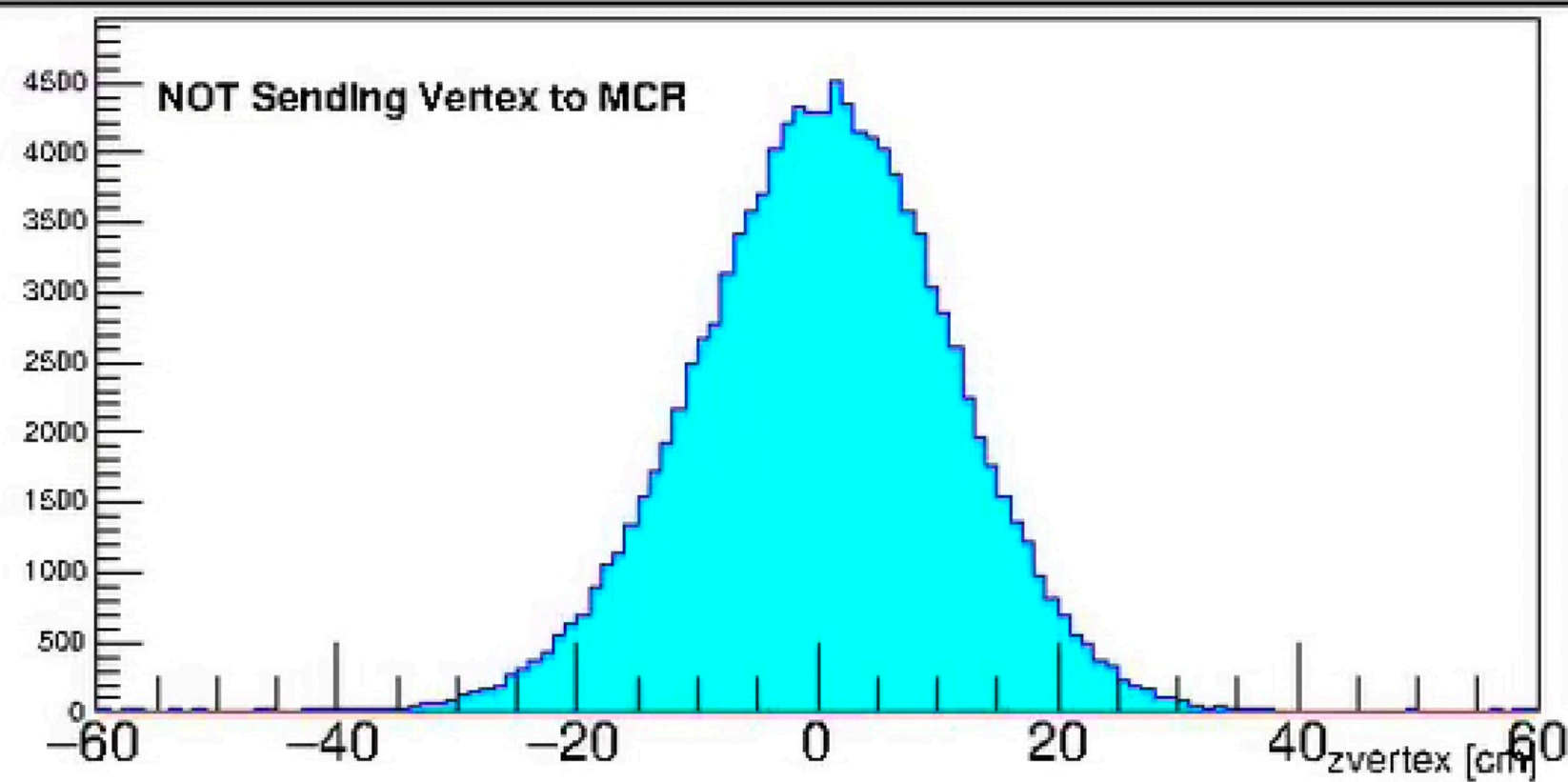
Back up

Run description - 54280

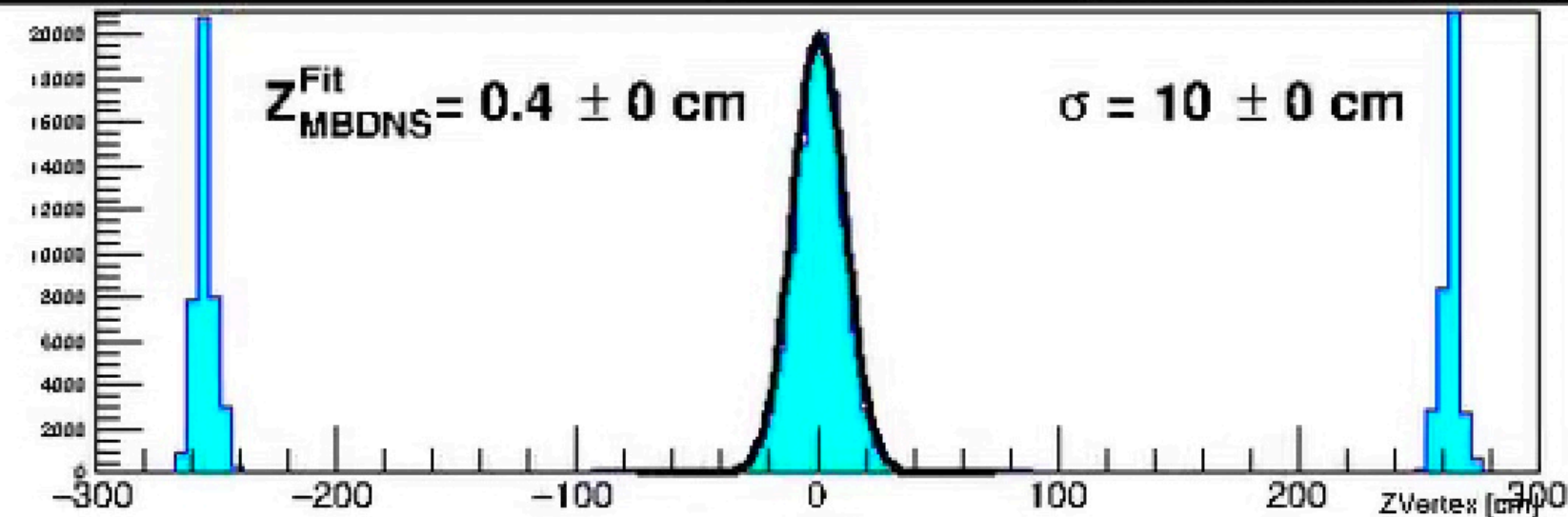
- Spike appears at each end of MBD
- The mini-bias definition is not yet available (as far as I know)
- Live trigger available to constraint the MBD vertex Z

Run #54280 Events: 204357 Date: Thu Oct 10 06:43:31 2011

MBD zvertex



MBD ZVertex (TRIG = MBDNS>=1)



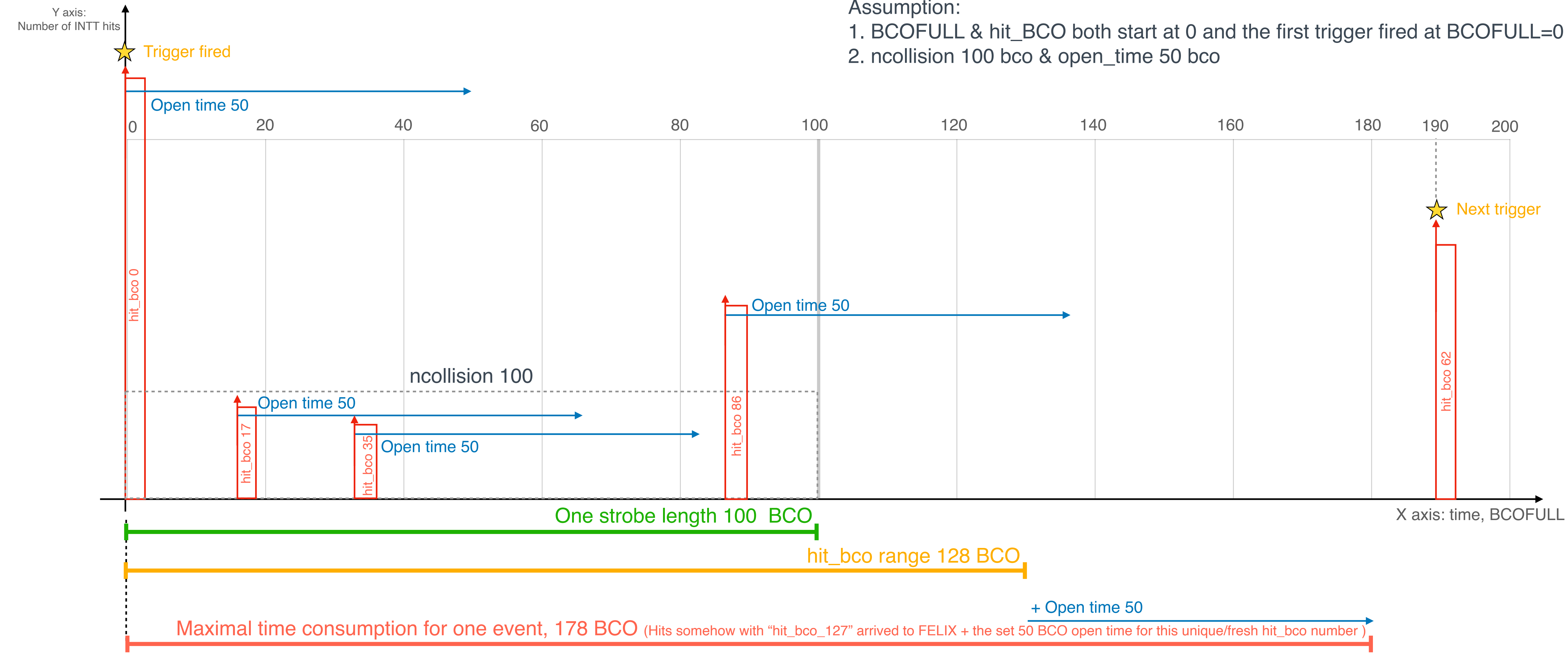
| Trigger input channel | Name | enabled | Scaledown | Raw | Live $\langle \div \rangle$ | Scaled | Live (%) |
|-----------------------|---------------------------|---------|-----------|-------------|-----------------------------|----------|----------|
| 0 | Clock | yes | 93810 | 33836274325 | 33663041357 | 358838 | 99.5 |
| 1 | ZDC South | yes | off | 102829214 | 102308816 | 0 | 99.5 |
| 2 | ZDC North | yes | off | 98430768 | 95872319 | 0 | 97.4 |
| 3 | ZDC Coincidence | yes | 60 | 9417100 | 9370209 | 153672 | 99.5 |
| 4 | HCAL Singles/Coincidence | yes | off | 30282609 | 30125423 | 0 | 99.5 |
| 5 | | yes | off | 33836274325 | 33663041357 | 0 | 99.5 |
| 6 | | yes | off | 0 | 0 | 0 | 0 |
| 7 | | yes | off | 0 | 0 | 0 | 0 |
| 8 | MBD S >= 2 | yes | off | 86958423 | 86380777 | 0 | 99.3 |
| 9 | MBD N >= 2 | yes | off | 85797943 | 85195687 | 0 | 99.3 |
| 10 | MBD N&S >= 2 | yes | 0 | 10242665 | 10187457 | 10187457 | 99.5 |
| 11 | MBD N&S >= 1 | yes | off | 18093659 | 17967450 | 0 | 99.3 |
| 12 | MBD N&S >= 2, vtx < 10 cm | yes | off | 4021509 | 4000602 | 0 | 99.5 |
| 13 | MBD N&S >= 2, vtx < 30 cm | yes | off | 5799143 | 5768655 | 0 | 99.5 |

Note: the hit transmission from chip to ROC: 1 hit / 1 bco

In single event

Assumption:

- 1. BCOFULL & hit_BCO both start at 0 and the first trigger fired at BCOFULL=0
- 2. ncollision 100 bco & open_time 50 bco



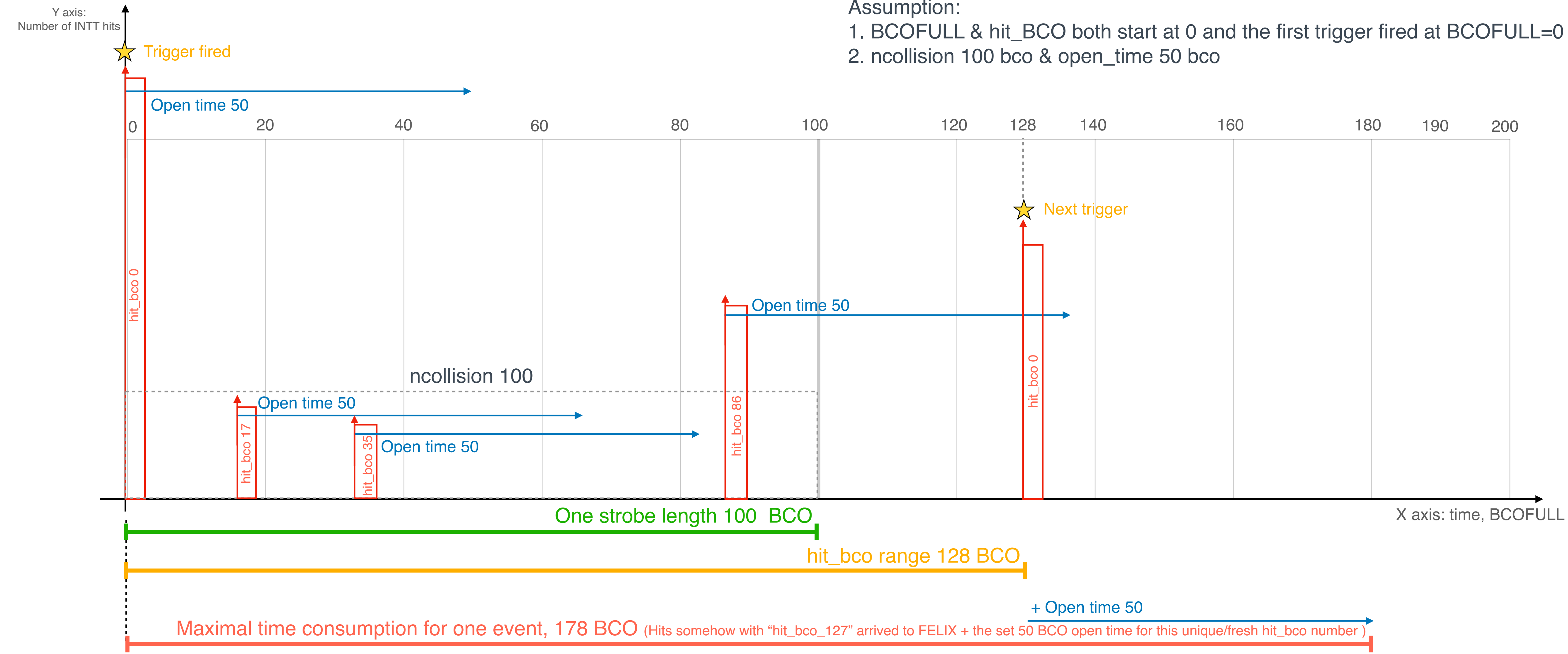
- Question 2.: As shown by cartoon, what if the “next trigger” is > 178 BCO away from the first trigger? (I assume this is the most safe case)

Note: the hit transmission from chip to ROC: 1 hit / 1 bco

In single event

Assumption:

1. BCOFULL & hit_BCO both start at 0 and the first trigger fired at BCOFULL=0
2. ncollision 100 bco & open_time 50 bco



- Question 3.: As shown in cartoon, what if we have hit_bco_0 in "this_event", and the next trigger fired at "BCOFULL_128 (hit_bco_0, again)". In addition, the FELIX is still taking the hits for hit_bco_86 for "this_event". What will happen?